THESIS

A DATA-BASED FINANCIAL MANAGEMENT INFORMATION SYSTEM (FMIS) FOR ADMINISTRATIVE SCIENCES DEPARTMENT

by

Neil S. Ford and Nicholas W. Zimmon

December, 1990

Thesis Advisor: Shu Liao
Co-Advisor: Tung Bui

Approved for public release; distribution is unlimited
**Title:** A DATA BASED FINANCIAL MANAGEMENT INFORMATION SYSTEM (FMIS) FOR ADMINISTRATIVE SCIENCES DEPARTMENT

**Authors:** Ford, Neil Spencer and Zimmon, Nicholas Waldon

**Abstract:**

The Administrative Sciences (AS) Department of the Naval Postgraduate School (NPS) is placing an increasing emphasis on keeping departmental expenses at minimum levels requiring the AS Department to carefully monitor a large number of complex financial accounts. It becomes necessary to develop a Financial Management Information System that would result in improved management of financial assets, better use of clerical skills, and more detailed, accurate, and up-to-date reporting within the AS Department. Based on the requirement analysis and prototypes performed by previous work, this thesis develops and implements a personal computer-based Management Information System for the management of the many funding accounts controlled by the Administrative Sciences Department. The central objective was to integrate accounting transactions performed in several different offices, currently using different software programs, into a singe all-comprising Management Information System. The system was written using dBASE IV and is currently operational.

**DISTRIBUTION AVAILABILITY OF ABSTRACT**

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unrestricted</td>
</tr>
</tbody>
</table>

**NAME OF RESPONSIBLE INDIVIDUAL**

Shanita Lam

**TELEPHONE (include Area code)**

408-646-2505
A Data-Based
Financial Management Information System (FMIS)
For Administrative Sciences Department

by

Neil Spencer Ford
Lieutenant Commander, United States Navy
B.S., Oklahoma State University, 1975

and

Nicholas Waldon Zimmon
Lieutenant, Supply Corps, United States Navy
B.S., University of Colorado, 1972

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN FINANCIAL MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
December 1990

Authors:

Neil S. Ford

Nicholas W. Zimmon

Approved by:

Shu Liao, Thesis Advisor

Tung Bui, Second Reader

David Whipple, Chairman
Department of Administrative Sciences
ABSTRACT

The Administrative Sciences (AS) Department of the Naval Postgraduate School (NPS) is placing an increasing emphasis on keeping departmental expenses at minimum levels requiring the AS Department to carefully monitor a large number of complex financial accounts. It becomes necessary to develop a Financial Management Information System that would result in improved management of financial assets, better use of clerical skills, and more detailed, accurate, and up-to-date reporting within the AS Department. Based on the requirement analysis and prototypes performed by previous work, this thesis develops and implements a personal computer-based Management Information System for the management of the many funding accounts controlled by the Administrative Sciences Department. The central objective was to integrate accounting transactions performed in several different offices, currently using different software programs, into a single all-encompassing Management Information System. The system was written using dBASE IV and is currently operational.
# TABLE OF CONTENTS

I. INTRODUCTION ............................................ 1  
   A. REVIEW OF CURRENT SYSTEMS ....................... 1  
   B. PURPOSE .................................................. 3  

II. SYSTEMS ANALYSIS AND DESIGN ......................... 5  
   A. SYSTEM REQUIREMENTS AND ANALYSIS .................. 5  
      1. Review of Current System ......................... 5  
      2. Survey of User's Requirements .................... 6  
      3. Framework Development ............................. 7  
   B. DATABASE AND REPORT SPECIFICATIONS ............... 7  
      1. Database Specifications ........................... 7  
      2. Report Specification ............................... 10  
   C. DEVELOPMENT AND IMPLEMENTATION ................... 10  
      1. Database Development .............................. 10  
      2. Database Characteristics .......................... 11  
      3. Database Generation ............................... 14  
      4. System Implementation ............................. 14  

III. CONCLUSIONS ............................................. 17  

APPENDIX A: DATA DICTIONARY ............................ 19
A. TABLE 1: DATA FILES ........................................... 19
B. TABLE 2: ACCTS.DBF DATA ELEMENTS ...................... 20
C. TABLE 3: DACCTS.DBF DATA ELEMENTS ...................... 21
D. TABLE 4: PERSONNE.DBF DATA ELEMENTS .................... 22
E. TABLE 5: LABOR1.DBF DATA ELEMENTS ....................... 24
F. TABLE 6: TEMPLAB.DBF DATA ELEMENTS ..................... 25
G. TABLE 7: TRAVEL.DBF DATA ELEMENTS ...................... 26
H. TABLE 8: SUPPLY.DBF DATA ELEMENTS ....................... 28
I. TABLE 9: DATABASE INDEXES ................................. 30
J. TABLE 10: REPORTS ........................................... 31

APPENDIX B: FMIS PROGRAMS ....................................... 32
A. EXPSUM.PRG .................................................. 32
B. DIRECT.PRG .................................................. 46
C. ACCTPROC.PRG ............................................... 50

APPENDIX C: APPLICATION DOCUMENTATION ..................... 65

APPENDIX D: USER'S GUIDE ........................................ 84
A. INTRODUCTION ................................................ 84
B. GETTING STARTED ............................................ 84
C. MENUS ......................................................... 86
   1. The Accounts Module ..................................... 89
      a. Add New Accounts .................................... 89
      b. View/Edit Accounts .................................. 91
      c. Remove Marked Records ............................. 93

V
d. Reports .............................. 94

e. Direct Fund Allocation .............. 94

2. The Personnel Module .............. 96
   a. Add New Personnel ................ 97
   b. View/Edit Personnel ................ 98
   c. Remove Marked Personnel .......... 98
   d. Reports .............................. 99

3. The Supply Module .................. 99
   a. Add Supply Transactions .......... 100
   b. View/Edit Transactions .......... 101
   c. Remove Marked Transactions .... 101
   d. Reports .............................. 101

4. The Labor Module ................... 102
   a. Add Payroll Records ................ 102
   b. View/Edit/Delete Payroll Records . 102
   c. Reports .............................. 103

5. The Travel Module ................... 103
   a. Add New Travel Orders ............ 104
   b. View/Edit Travel Orders .......... 105
   c. Remove Marked Travel Orders ..... 106
   d. Reports .............................. 106

6. Tools Module ........................ 106
   a. Backup Data to Drive A ............ 107
   b. Import Supply Data ............... 108
   c. Rebuild Corrupted Indexes ....... 108
   d. Restore Database Files from Drive A 109

vi
e. Export Acct\Pers Data ............ 109

7. Exit Menu ....................... 109

LIST OF REFERENCES ............. 111

INITIAL DISTRIBUTION ............ 112
I. INTRODUCTION

The purpose of this thesis was to design a menu-driven data-base oriented decision support system to manage the various resource accounts controlled by the Administrative Science Department, Naval Postgraduate School. The myriad management systems currently in use will be consolidated into one comprehensive system, eliminating much of the manual manipulation of data presently required to generate financial reports. This computerized system will be capable of producing reports to meet existing internal reporting requirements while providing the ability to generate additional reports and queries as needed. The menu-driven format will require minimal computer knowledge by the user to operate the data-based system. The system application is written in dBASE IV, implemented on a MS-DOS-based personal computer.

A. REVIEW OF CURRENT SYSTEMS

With the dwindling resources available to all DOD activities, the need to effectively use all available dollars is more important than ever. In order to manage resources effectively, a properly designed Financial Management Information System (FMIS) will be indispensable.
The need for a resource management information system in the AS Department has been examined by Renner [Ref. 1]. The study conducted a structured analysis to determine the computing needs of the AS Department. This analysis defined three different sub-systems:

- Financial
- Personnel
- Property

to keep track of all information pertaining to the management and control of departmental activities. Booker [Ref. 2] developed and implemented a prototype system in 1986 in part as a feasibility study, implementing some of the features of each subsystem outlined in the previous analysis. The prototype proved the system feasible by showing that many of the manual procedures could be automated. Sexton [Ref. 3] developed and implemented the Property Management System (PMS) to support the management and accountability of the departmental property on a microcomputer.

Other information systems currently in use were created by the operators and users of the information using personal computer (PC) based software that they were familiar with. The result is an assortment of different information systems for tracking financial transactions and producing reports. Because both database and spreadsheet programs are in use, there is much duplication of data entry to create the reports required by management. Users familiar with one software
program are typically not familiar with the other software program, limiting the ability for any user to access the information he or she requires.

B. PURPOSE

The purpose of this thesis was to develop a personal computer-based Resource Management Information System for the management of the many funding accounts controlled by the Administrative Sciences Department. The central objective was to integrate accounting transactions currently performed in several different offices using different software programs, into a single all-encompassing Resource Management Information System.

The main requirement of the system was to maintain financial accountability for the numerous fund accounts entrusted to the department. The ability of the system to provide management reports, geared toward both internal and external reporting requirements, was crucial to the development of the system. The FMIS had to be efficient and responsive to requests for information. The FMIS would track financial transactions involving labor, travel, and CAPTAR accounts.

The project was divided into five phases. The first phase consisted of a survey of the user's requirements. This included understanding the accounting system, both in terms of internal processes and external requirements.
The second phase was the selection of a software package that met three criteria: (1) readily available at NPS, (2) an inherent capability to store and process large amounts of data and, (3) would provide an easy-to-learn and user-friendly interface. Ashton-Tate's dBASE IV, a database management system, meet the preceding requirements and was chosen as the project's software package.

Phase three involved database development and report definition. Phase four involved installing the prototype FMIS on AS Department PC's for testing. Both phases required close interaction with the users.

The fifth phase involved actual implementation of the system at the beginning of FY 91. It commenced with entry of the new fiscal year's data and manual comparison to assure the accuracy and reliability of the system. Close monitoring of the operational effectiveness continued through the first quarter of the fiscal year.
II. SYSTEMS ANALYSIS AND DESIGN

Prior to development of the FMIS an understanding of the current practice was required. The informational needs of the users were surveyed and the goals of the FMIS were defined. A framework was then developed and a software package chosen. Once the databases were defined, the desired reports to be generated were designed. The final requirement was the preparation of an application to implement the FMIS.

A. SYSTEM REQUIREMENTS AND ANALYSIS

1. Review of Current System

The AS Department is responsible for maintaining approximately 35 separate fund accounts. The nature of these funds include the day-to-day support of the department, including supplies, staff/faculty salaries, funding from external sources in support of scholarly research performed by faculty members, and other staff/faculty support funding. Within each account, all obligations including salaries, travel and OPTAR expenditures, must be tracked to meet the external financial reporting requirements, and just as importantly, to enable financial decision making.

The management of all accounts was being done separately, by those in charge of maintaining direct and indirect funds and reimbursable account status along with
supply requisition status, at several desks using personal computers and various software products. While the processes were "computerized", there had been no integration of the various functions. In effect, it was necessary to maintain lists of transactions for manual compilation into several "master" computer spreadsheets to produce summary reports. This process was extremely labor intensive in that data was repetitively processed at more than one point to generate reports for managerial use.

2. Survey of User's Requirements

To begin designing the Administrative Sciences (AS) Department's Management Information System (FMIS), an extensive survey was conducted to determine specific user needs, to get an understanding of the financial accounting system within which the FMIS was to operate, to assess the current system in use by the department, and to select a software package considered both powerful and flexible enough to meet the system requirements. This was accomplished through numerous interviews with key personnel tasked with the financial management responsibility within the department.

The main requirement of the FMIS was established: To maintain financial accountability for the numerous fund accounts entrusted to the department from sources both internal to NPS and from outside the school, while providing an efficient, labor-saving means to do so.
3. Framework Development

The survey resulted in the development of a framework by which the FMIS was to be designed, and was defined as follows:

- The system would employ a Data Base Management System (DBMS) to enable data accumulation and to provide efficient report generation.
- A user-friendly menu-driven system requiring little technical knowledge of the software and computer operation.
- Adaptable for future installation on a Local Area Network and implementation of additional user specified modules.
- Developed using a software package that was readily available in the market, and as previously mentioned, provide the power and flexibility required for the defined system. dBASE IV was chosen for those reasons.
- Use PC/MS DOS based personal computers.

B. DATABASE AND REPORT SPECIFICATIONS

Once an understanding of the financial management function of the AS Department was obtained, the system framework defined, and a software package selected, the next step was to begin development of the FMIS.

1. Database Specifications

Databases were developed and their relationships established. This was accomplished through user interviews in which six logical database files and their specific data elements were defined. It was also necessary to develop one additional temporary database file to enable batch updating. Each was to be developed as a separate module, with an eye
towards ensuring that relationships could be established as necessary to take advantage of the power afforded by a relational database structure.

As depicted in Figure 2.1, the data files constitute a hierarchy. The Accounts database is considered to be the "master" database file because all fund accounts and their respective initial balances are established there. Subordinate to the Accounts file is the Direct Accounts file. This was required to establish specific funding allotments to individual accounts controlled by principal investigators. Subordinate to both the Accounts and Direct Accounts file; the Labor, Supply and Travel databases were established to accumulate transactional processing data for later report preparation drawing on account information located in both the Account and Direct Account databases. A Personnel file was established to maintain personal staff/faculty information, in particular, salary information for use in payroll report processing. Further, relationships were established between the Accounts and Personnel data files and the Labor, Supply and Travel files to enable account and personnel validation during transaction data entry. This was accomplished through use of user-defined functions, a feature of dBASE IV. Finally, as mentioned above, a temporary database file was created to allow batch updating of the Labor database so that actual payroll amounts could be stored with each record.
Figure 2.1 depicts the various relationships that were established to take advantage of the relational structure afforded by dBASE IV. Specifics regarding the above data files and their elements are contained in the Data Dictionary, Appendix A.

In conjunction with the definition of databases, and to facilitate data entry into the system, a formatted data entry screen was created for each data file. They were based on the format in which raw data was obtained in an effort to facilitate data entry.
2. Report Specification

Once the database files and their attributes were defined, report definition began. This was also accomplished with direct user input. Three main categories of reports were developed:

- Summary reports - management reports aimed toward providing bottom-line fund status.
- Audit reports - reports supporting the Summary Reports and providing specific transaction information.
- Status/Tickler reports - reports geared toward supporting individual transactions, e.g., supply status listings, delinquent travel claims listings, etc.

The report generating feature of dBASE IV enabled development of all report requirements with the exception of the summary reports. Because this feature precludes the simultaneous use of multiple database files, it was necessary to use the dBASE IV programming language to create these reports.

A listing of all reports and their specific file names and related views is provided in Appendix A, Reports. Additionally, dBASE programming code documentation for the summary reports is provided in Appendix B, Program Listings.

C. DEVELOPMENT AND IMPLEMENTATION

1. Database Development

Developing a database requires three steps. Data requirements for the organization as a whole must be identified. The key entities, attributes, and relationships
that constitute the organization's data must be identified. The next step is to define the logical relationships between the data, a process known as data modeling. These relationships must be represented in the database. The third step involves organizing the database in such a way as to optimize access, flexibility, and efficiency.

2. Database Characteristics

A database is a set of data organized to serve many applications efficiently by centralizing the data and minimizing redundant data. dBASE IV, a database management system (DBMS), is a popular PC-based software package that accomplishes this task.

The data resides physically on a computer disk in the form of data elements. dBASE IV acts as the interface between application programs and the physical data files. Application programs request data elements from the database. When the application program calls for a data element like travel advances, dBASE IV finds this element in the database and presents it to the application program. The software package incorporates all of the computer languages required to execute the requirements of the user in a user-friendly application. The end user is relieved of the task of understanding where and how the data is actually stored. This permits the end user to issue a few simple queries to produce reports.
Every information system must keep track of certain entities (e.g., faculty, supply requisitions) that have certain attributes (e.g., address, cost) and certain relations among them (e.g., temporary faculty, overdue requisitions). Previously this could have been a 3 x 5 card file system. There are three different ways in which the DBMS keeps track of entities, attributes, and relations.

Hierarchical DBMSs present data to users in a tree-like structure. To the user, each record looks like an organizational chart with an upper segment connected logically to a lower segment in a parent-child or one-to-many relationship, one parent and possibly many children. The data is physically linked to one another by a series of pointers that form chains of related data segments.

Network DBMSs are a variation of hierarchical DBMSs. Network structures depict data logically as many-to-many relationships. Network structures reduce redundancy and are often faster than hierarchical DBMSs but require many more pointers. Advantages to both types of DBMSs are organization and access to data, efficient processing of transactions, and the ability to have a large number of variable-length records. This makes them very good for structured routine request, but the access paths, directories, and indices must be specified in advance. But, this design has low flexibility and is programming intensive, time consuming, and difficult to install.
A relational model can relate any piece of information stored in one file to any piece of information stored in another file as long as the two tables share a common data element. In each file, the rows or tuples are unique records and the columns are fields. A relational database is highly flexible in regard to ad hoc inquiries, power to combine information from different sources, simplicity of design and maintenance, and the ability to add new data and records without disturbing existing programs and applications. A significant weakness of a relation database is processing efficiency. They are slow because they typically require many accesses to the data stored on the disk. This problem is significant only for large-volume transaction processing systems like airline reservation systems, and should not be significant in our application.

Logically defined, dBASE IV incorporates a relational database structure. Each database file or table consist of records (or rows) of information. Each record is divided into separate fields (or columns) of information. Forms are used to input data into files and reports are the familiar output. A one-to-one relationship among forms, database files, and reports is not required. dBASE IV can easily pull together information from separate database tables into a single report. Information may be added, deleted, or changed effortlessly.
3. Database Generation

The final step before system installation was to tie the various modules of the FMIS together into a menu-driven system. This was accomplished using the dBASE IV Applications Generator. This powerful feature enabled the creation of pull-down menus for all aspects of the system's use.

Within the application, each database file and the various operations associated with it are accessed through individual menus. Additionally, a "Tools" menu is included to provide several utilities for data file maintenance, e.g., database backup and restoration, rebuilding of corrupted indexes, etc. Application documentation is provided in Appendix C, Application Documentation.

4. System Implementation

The FMIS was installed on two personal computers in the AS Department two weeks prior to the beginning of FY 91. This provided the opportunity to test the system with sample data while at the same time, giving the user hands-on operating experience. Several minor errors were detected during the period and corrected. As to be expected, this also resulted in additional user requirements (additional reports) as the power of the FMIS became apparent.

Because data entry to the system was accomplished in two different locations during the test period, it was necessary to produce two different versions of the system (a
temporary measure until installation of the proposed local area network (LAN)). The main system was installed in the AS Department office; more easily accessible to key supervisory personnel. All data entry, with the exception of supply transactions, and report generation is accomplished at this location. A second system, limited to supply support, was installed at another location where supply services are provided to staff and faculty personnel. This system differed in that complete access to the Supply database and limited access to the Accounts database (to verify existence Job Order Numbers for validation purposes) was provided. The user can perform data entry and print various requisition status reports from this location. Also, because report generation is performed at the main location, and account and personnel validation is performed at the other using updated account and personnel information, it is necessary to share data between the two systems. Utilities were provided in each system's "Tools" menu for the import/export of updated data files.

The installation phase proved very successful. With the exception of the before mentioned errors, the program performed as designed. Extensive review of all reports by supervisory personnel indicated that all transactions were being processed correctly; balances, total obligations and individual transactions were accurately reflected.

The FMIS was well received by the user. A presentation of the nearly finalized version was made to the
Department Chairman and other key supervisory personnel and was accepted with suggestions for future development in subsequent thesis research. As a result, the system was put into use beginning with the new fiscal year and monitored closely during the first quarter for unforeseen processing errors. No significant problems were encountered.

Finally, a user's manual was prepared along with two backup copies of the system. With empty database files, the system will fit on a single 1.2 Megabyte floppy diskette. The user's manual is contained in Appendix D.
III. CONCLUSIONS

The Financial Management Information System is presently being utilized as a means of accounting for the many funded accounts the AS Department Chairman must keep track of. The users have indicated their satisfaction with the system and look forward to being able to track financial data with improved accuracy and timeliness.

Since the implementation of the FMIS at the beginning of Fiscal Year 1991, very small changes have been required to accomplish all of the goals set forth in the beginning. With the assistance of the user's manual, this thesis, and a basic knowledge of dBASE IV, this data based application could be easily adapted for other departmental financial records.

Financial management was only one of three areas where PC-based computer applications would be beneficial. The Property Management System is currently providing a useful means of maintaining accountability for AS Department property. The users have expressed an interest in improving the property management system with more detailed reports and possibly integrated with the supply and personnel databases. Weaknesses in this system were not explored in this thesis and are left as possible areas for future study.

Although this system does have a limited personnel database, an improved or enhanced personnel database with
additional reports may provide potentially useful information to the department's decision makers.

Finally, as the users becomes more familiar with the reports available, new reports will be identified. This might entail adding new records to the existing databases but should not require significant modifications to this FMIS. With the implementation of a local area network tying together the different databases and their users, additional problems may surface requiring a person with knowledge of dBASE IV to correct.

In conclusion, the goal of this thesis was to deliver a satisfactory system that the users would be comfortable with and have this system installed to support the new fiscal year reporting requirements. This goal was accomplished with the Financial Management Information System.
APPENDIX A: DATA DICTIONARY

A. TABLE 1: DATA FILES

<table>
<thead>
<tr>
<th>FILE NAME</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCTS.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT AN ACCOUNT</td>
</tr>
<tr>
<td>DACCTS.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT A DIRECT ACCOUNT</td>
</tr>
<tr>
<td>PERSONNE.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT AN EMPLOYEE</td>
</tr>
<tr>
<td>LABOR1.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT INDIVIDUAL PAYROLL TRANSACTIONS</td>
</tr>
<tr>
<td>TEMPLAB.DBF</td>
<td>DATA</td>
<td>ENABLES CALCULATIONS OF INDIVIDUAL PAYROLL TRANSACTIONS FOR UPDATING LABOR1.DBF</td>
</tr>
<tr>
<td>SUPPLY.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT INDIVIDUAL SUPPLY TRANSACTIONS</td>
</tr>
<tr>
<td>TRAVEL.DBF</td>
<td>DATA</td>
<td>ATTRIBUTES ABOUT INDIVIDUAL TRAVEL ORDERS</td>
</tr>
</tbody>
</table>
B. TABLE 2: ACCTS.DBF DATA ELEMENTS

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>TYPE</th>
<th>WIDTH</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>FUNDTYPE</td>
<td>CHAR</td>
<td>1</td>
<td>TYPE OF ACCOUNT</td>
</tr>
<tr>
<td>MIPRNR</td>
<td>CHAR</td>
<td>15</td>
<td>MIPR NUMBER</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>TITLE</td>
<td>CHAR</td>
<td>50</td>
<td>NAME AND DESCRIPTION OF ACCOUNT</td>
</tr>
<tr>
<td>SPONSOR</td>
<td>CHAR</td>
<td>30</td>
<td>SPONSORING ACTIVITY OF ACCOUNT</td>
</tr>
<tr>
<td>DATERECVD</td>
<td>DATE</td>
<td>8</td>
<td>DATE FUNDING AUTHORIZED</td>
</tr>
<tr>
<td>EXPDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE FUNDING AUTHORITY EXPIRES</td>
</tr>
<tr>
<td>SERIAL1</td>
<td>CHAR</td>
<td>11</td>
<td>SERIAL NUMBER BLOCKS ASSIGNED TO ACCOUNT</td>
</tr>
<tr>
<td>SERIAL2</td>
<td>CHAR</td>
<td>9</td>
<td>ADDITIONAL SERIAL NUMBER BLOCK</td>
</tr>
<tr>
<td>PROJORDER</td>
<td>LOG</td>
<td>1</td>
<td>IS ACCOUNT PROJECT ORDER?</td>
</tr>
<tr>
<td>BGTPGDATE</td>
<td>DATE</td>
<td>8</td>
<td>BUDGET PAGE DATE</td>
</tr>
<tr>
<td>CONSUP</td>
<td>LOG</td>
<td>1</td>
<td>IS CONTRACT SUPPORT AUTHORIZED?</td>
</tr>
<tr>
<td>CONSUPAMT</td>
<td>NUM</td>
<td>12</td>
<td>AMOUNT OF CONTRACTOR SUPPORT AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>LABOR</td>
<td>NUM</td>
<td>12</td>
<td>LABOR FUNDING AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>OPTAR</td>
<td>NUM</td>
<td>12</td>
<td>SUPPLY SUPPORT FUNDING AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>TRAVEL</td>
<td>NUM</td>
<td>12</td>
<td>TRAVEL FUNDING AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>INTTLFUNDS</td>
<td>NUM</td>
<td>12</td>
<td>INITIAL TOTAL FUNDING AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>TTLFUNDS</td>
<td>NUM</td>
<td>12</td>
<td>CURRENT AUTHORIZED TOTAL FUNDING AUTHORIZED BY ACCOUNT</td>
</tr>
<tr>
<td>ELEMENT</td>
<td>TYPE</td>
<td>WIDTH</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
<td>-------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>NAME</td>
<td>CHAR</td>
<td>15</td>
<td>NAME OF PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PRINCIPAL INVESTIGATOR CODE</td>
</tr>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>SERIAL1</td>
<td>CHAR</td>
<td>11</td>
<td>SERIAL NUMBER BLOCKS ASSIGNED TO ACCOUNT</td>
</tr>
<tr>
<td>SERIAL2</td>
<td>CHAR</td>
<td>9</td>
<td>ADDITIONAL SERIAL NUMBER BLOCK</td>
</tr>
<tr>
<td>TRAVEL</td>
<td>NUM</td>
<td>9</td>
<td>TRAVEL FUNDING AUTHORIZED TO PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>LABOR</td>
<td>NUM</td>
<td>9</td>
<td>LABOR FUNDING AUTHORIZED TO PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>OPTAR</td>
<td>NUM</td>
<td>9</td>
<td>SUPPLY SUPPORT FUNDING AUTHORIZED TO PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>CONSUPAMT</td>
<td>NUM</td>
<td>9</td>
<td>CONTRACTOR SUPPORT AUTHORIZED TO PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>TOTAL</td>
<td>NUM</td>
<td>12</td>
<td>TOTAL FUNDING AUTHORIZED TO PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>ELEMENT</td>
<td>TYPE</td>
<td>WIDTH</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------------</td>
<td>-------</td>
<td>-------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>LASTNAME</td>
<td>CHAR</td>
<td>15</td>
<td>LAST NAME OF INDIVIDUAL</td>
</tr>
<tr>
<td>FIRSTNAME</td>
<td>CHAR</td>
<td>10</td>
<td>FIRST NAME OF INDIVIDUAL</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PRINCIPAL INVESTIGATOR CODE</td>
</tr>
<tr>
<td>IDCODE</td>
<td>CHAR</td>
<td>2</td>
<td>IDENTIFICATION CODE</td>
</tr>
<tr>
<td>PLOFBIRTH</td>
<td>CHAR</td>
<td>15</td>
<td>PLACE OF BIRTH</td>
</tr>
<tr>
<td>ADDRESS</td>
<td>CHAR</td>
<td>20</td>
<td>STREET ADDRESS OF RESIDENCE</td>
</tr>
<tr>
<td>CITY</td>
<td>CHAR</td>
<td>15</td>
<td>CITY OF RESIDENCE</td>
</tr>
<tr>
<td>STATE</td>
<td>CHAR</td>
<td>2</td>
<td>STATE OF RESIDENCE</td>
</tr>
<tr>
<td>ZIP</td>
<td>CHAR</td>
<td>5</td>
<td>ZIP CODE</td>
</tr>
<tr>
<td>EMCONTACT</td>
<td>CHAR</td>
<td>20</td>
<td>EMERGENCY CONTACT</td>
</tr>
<tr>
<td>FACSTAFF</td>
<td>CHAR</td>
<td>1</td>
<td>FACULTY OR STAFF</td>
</tr>
<tr>
<td>HPHONE</td>
<td>CHAR</td>
<td>13</td>
<td>HOME PHONE NUMBER</td>
</tr>
<tr>
<td>SSN</td>
<td>CHAR</td>
<td>11</td>
<td>SOCIAL SECURITY NUMBER</td>
</tr>
<tr>
<td>BDATE</td>
<td>DATE</td>
<td>8</td>
<td>BIRTH DATE</td>
</tr>
<tr>
<td>OPHONE</td>
<td>CHAR</td>
<td>4</td>
<td>OFFICE PHONE</td>
</tr>
<tr>
<td>GRADESTEP</td>
<td>CHAR</td>
<td>8</td>
<td>GRADE AND STEP</td>
</tr>
<tr>
<td>ODOOR</td>
<td>CHAR</td>
<td>4</td>
<td>OFFICE NUMBER</td>
</tr>
<tr>
<td>TITLE</td>
<td>CHAR</td>
<td>20</td>
<td>TITLE OF POSITION HELD</td>
</tr>
<tr>
<td>STATUS</td>
<td>CHAR</td>
<td>15</td>
<td>EMPLOYMENT STATUS</td>
</tr>
<tr>
<td>PDNO</td>
<td>CHAR</td>
<td>4</td>
<td>POSITION DESCRIPTION NUMBER</td>
</tr>
<tr>
<td>BILLETNR</td>
<td>CHAR</td>
<td>10</td>
<td>BILLET NUMBER OCCUPIED</td>
</tr>
<tr>
<td>BASESAL</td>
<td>NUM</td>
<td>9</td>
<td>BASE SALARY</td>
</tr>
<tr>
<td>PROMODATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE OF LAST PROMOTION</td>
</tr>
<tr>
<td>ELEMENTS</td>
<td>TYPE</td>
<td>WIDTH</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
<td>-------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>APPTSTART</td>
<td>DATE</td>
<td>8</td>
<td>APPOINTMENT START DATE</td>
</tr>
<tr>
<td>APPTEND</td>
<td>DATE</td>
<td>8</td>
<td>APPOINTMENT END DATE</td>
</tr>
<tr>
<td>CISBEG</td>
<td>DATE</td>
<td>8</td>
<td>CIS BEGIN DATE</td>
</tr>
<tr>
<td>CISEND</td>
<td>DATE</td>
<td>8</td>
<td>CIS END DATE</td>
</tr>
<tr>
<td>REMARKS</td>
<td>MEMO</td>
<td>10</td>
<td>REMARKS</td>
</tr>
</tbody>
</table>
### E. TABLE 5: LABOR1.DBF DATA ELEMENTS

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>TYPE</th>
<th>WIDTH</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>IDCODE</td>
<td>CHAR</td>
<td>2</td>
<td>ID CODE OF PAYEE</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PI CODE OF PAYEE</td>
</tr>
<tr>
<td>LASTNAME</td>
<td>CHAR</td>
<td>15</td>
<td>LAST NAME OF PAYEE</td>
</tr>
<tr>
<td>FIRSTNAME</td>
<td>CHAR</td>
<td>10</td>
<td>FIRST NAME OF PAYEE</td>
</tr>
<tr>
<td>PPE</td>
<td>DATE</td>
<td>8</td>
<td>PAY PERIOD END DATE</td>
</tr>
<tr>
<td>REGHRS</td>
<td>NUM</td>
<td>4</td>
<td>REGULAR HOURS WORKED</td>
</tr>
<tr>
<td>OTHRS</td>
<td>NUM</td>
<td>4</td>
<td>OVERTIME HOURS WORKED</td>
</tr>
<tr>
<td>ANLEAVE</td>
<td>NUM</td>
<td>4</td>
<td>ANNUAL LEAVE HOURS TAKEN</td>
</tr>
<tr>
<td>SKLEAVE</td>
<td>NUM</td>
<td>4</td>
<td>SICK LEAVE HOURS TAKEN</td>
</tr>
<tr>
<td>HOLIDAY</td>
<td>NUM</td>
<td>4</td>
<td>HOLIDAY HOURS TAKEN</td>
</tr>
<tr>
<td>ACCRATE</td>
<td>NUM</td>
<td>6</td>
<td>ACCELERATION RATE</td>
</tr>
<tr>
<td>TOTPAY</td>
<td>NUM</td>
<td>10</td>
<td>TOTAL PAY OBLIGATION</td>
</tr>
</tbody>
</table>
**F. TABLE 6: TEMPLAB.DBF DATA ELEMENTS**

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>TYPE</th>
<th>WIDTH</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>IDCODE</td>
<td>CHAR</td>
<td>2</td>
<td>ID CODE OF PAYEE</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PI CODE OF PAYEE</td>
</tr>
<tr>
<td>LASTNAME</td>
<td>CHAR</td>
<td>15</td>
<td>LAST NAME OF PAYEE</td>
</tr>
<tr>
<td>FIRSTNAME</td>
<td>CHAR</td>
<td>10</td>
<td>FIRST NAME OF PAYEE</td>
</tr>
<tr>
<td>PPE</td>
<td>DATE</td>
<td>8</td>
<td>PAY PERIOD END DATE</td>
</tr>
<tr>
<td>REGHRS</td>
<td>NUM</td>
<td>4</td>
<td>REGULAR HOURS WORKED</td>
</tr>
<tr>
<td>OTHRS</td>
<td>NUM</td>
<td>4</td>
<td>OVERTIME HOURS WORKED</td>
</tr>
<tr>
<td>ANLEAVE</td>
<td>NUM</td>
<td>4</td>
<td>ANNUAL LEAVE HOURS TAKEN</td>
</tr>
<tr>
<td>SKLEAVE</td>
<td>NUM</td>
<td>4</td>
<td>SICK LEAVE HOURS TAKEN</td>
</tr>
<tr>
<td>HOLIDAY</td>
<td>NUM</td>
<td>4</td>
<td>HOLIDAY HOURS TAKEN</td>
</tr>
<tr>
<td>ACCRATE</td>
<td>NUM</td>
<td>6</td>
<td>ACCELERATION RATE</td>
</tr>
<tr>
<td>TOTPAY</td>
<td>NUM</td>
<td>10</td>
<td>TOTAL PAY OBLIGATION</td>
</tr>
</tbody>
</table>
### G. TABLE 7: TRAVEL.DBF DATA ELEMENTS

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>TYPE</th>
<th>WIDTH</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREPDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE ORDERS PREPARED</td>
</tr>
<tr>
<td>TRAVTYPE</td>
<td>CHAR</td>
<td>13</td>
<td>TYPE OF TRAVEL</td>
</tr>
<tr>
<td>PURPOSE</td>
<td>CHAR</td>
<td>15</td>
<td>PURPOSE OF TRAVEL</td>
</tr>
<tr>
<td>TRAVDATE</td>
<td>DATE</td>
<td>8</td>
<td>PROCEED DATE</td>
</tr>
<tr>
<td>ESTRTNDATE</td>
<td>DATE</td>
<td>8</td>
<td>ESTIMATED RETURN DATE</td>
</tr>
<tr>
<td>DOCNR</td>
<td>CHAR</td>
<td>9</td>
<td>DOCUMENT NUMBER ASSIGNED TO TRAVEL ORDERS</td>
</tr>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PRINCIPAL INVESTIGATOR CODE</td>
</tr>
<tr>
<td>FLAGAPP</td>
<td>CHAR</td>
<td>1</td>
<td>FLAG APPROVAL REQUIRED?</td>
</tr>
<tr>
<td>LASTNAME</td>
<td>CHAR</td>
<td>15</td>
<td>LAST NAME OF TRAVELER</td>
</tr>
<tr>
<td>FIRSTNAME</td>
<td>CHAR</td>
<td>10</td>
<td>FIRST NAME OF TRAVELER</td>
</tr>
<tr>
<td>PLACE</td>
<td>CHAR</td>
<td>60</td>
<td>TRAVELER'S ITINERARY</td>
</tr>
<tr>
<td>PEREXP</td>
<td>NUM</td>
<td>7</td>
<td>ESTIMATED PER DIEM CHARGE</td>
</tr>
<tr>
<td>ACTPERD</td>
<td>NUM</td>
<td>7</td>
<td>ACTUAL PER DIEM CHARGE</td>
</tr>
<tr>
<td>TRANSEXPE</td>
<td>NUM</td>
<td>7</td>
<td>ESTIMATED TRANSPORTATION CHARGE</td>
</tr>
<tr>
<td>ACTTRANS</td>
<td>NUM</td>
<td>7</td>
<td>ACTUAL TRANSPORTATION CHARGE</td>
</tr>
<tr>
<td>REGEXP</td>
<td>NUM</td>
<td>7</td>
<td>ESTIMATED REGISTRATION CHARGE</td>
</tr>
<tr>
<td>ACTREG</td>
<td>NUM</td>
<td>7</td>
<td>ACTUAL REGISTRATION CHARGE</td>
</tr>
<tr>
<td>ESTPRICE</td>
<td>NUM</td>
<td>7</td>
<td>ESTIMATED TOTAL TRAVEL CHARGE</td>
</tr>
<tr>
<td>ACTAMT</td>
<td>NUM</td>
<td>7</td>
<td>ACTUAL TRAVEL CHARGE</td>
</tr>
<tr>
<td>ADPD</td>
<td>DATE</td>
<td>8</td>
<td>ADVANCE PICKUP DATE</td>
</tr>
<tr>
<td>TICPD</td>
<td>DATE</td>
<td>8</td>
<td>TICKET PICKUP DATE</td>
</tr>
<tr>
<td>REGPD</td>
<td>DATE</td>
<td>8</td>
<td>REGISTRATION PICKUP DATE</td>
</tr>
<tr>
<td>ELEMENT</td>
<td>TYPE</td>
<td>WIDTH</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>ADVCODE</td>
<td>CHAR</td>
<td>5</td>
<td>ADVANCE CODE</td>
</tr>
<tr>
<td>ADVDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE OF TRAVEL ADVANCE</td>
</tr>
<tr>
<td>ADVAMT</td>
<td>NUM</td>
<td>7</td>
<td>AMOUNT OF TRAVEL ADVANCE</td>
</tr>
<tr>
<td>CLAIMSUB</td>
<td>LOG</td>
<td>1</td>
<td>TRAVEL CLAIM SUBMITTED?</td>
</tr>
<tr>
<td>CLAIMDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE TRAVEL CLAIM SUBMITTED</td>
</tr>
<tr>
<td>CANC</td>
<td>LOG</td>
<td>1</td>
<td>TRAVEL ORDERS CANCELED?</td>
</tr>
<tr>
<td>CANCDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE TRAVEL ORDERS CANCELED</td>
</tr>
</tbody>
</table>
## Table 8: Supply.dbf Data Elements

<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>TYPE</th>
<th>WIDTH</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DORDER</td>
<td>DATE</td>
<td>8</td>
<td>DATE OF ORDER</td>
</tr>
<tr>
<td>DOCNR</td>
<td>CHAR</td>
<td>9</td>
<td>DOCUMENT NUMBER ASSIGNED</td>
</tr>
<tr>
<td>JON</td>
<td>CHAR</td>
<td>5</td>
<td>JOB ORDER NUMBER</td>
</tr>
<tr>
<td>EXPELMNT</td>
<td>CHAR</td>
<td>9</td>
<td>EXPENSE ELEMENT</td>
</tr>
<tr>
<td>PI</td>
<td>CHAR</td>
<td>5</td>
<td>PRINCIPAL INVESTIGATOR</td>
</tr>
<tr>
<td>VENDOR</td>
<td>CHAR</td>
<td>20</td>
<td>SUPPLIER OF REQUIREMENT</td>
</tr>
<tr>
<td>ESTPRICE</td>
<td>NUM</td>
<td>9</td>
<td>ESTIMATED PRICE</td>
</tr>
<tr>
<td>DESC</td>
<td>CHAR</td>
<td>35</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>CATEGORY</td>
<td>CHAR</td>
<td>5</td>
<td>CATEGORY OF REQUIREMENT</td>
</tr>
<tr>
<td>REQUESTOR</td>
<td>CHAR</td>
<td>2</td>
<td>REQUESTOR CODE</td>
</tr>
<tr>
<td>PRI</td>
<td>CHAR</td>
<td>3</td>
<td>PRIORITY OF REQUIREMENT</td>
</tr>
<tr>
<td>RDD</td>
<td>DATE</td>
<td>8</td>
<td>REQUIRED DELIVERY DATE</td>
</tr>
<tr>
<td>PONR</td>
<td>CHAR</td>
<td>12</td>
<td>PURCHASE ORDER NUMBER</td>
</tr>
<tr>
<td>PODATE</td>
<td>DATE</td>
<td>8</td>
<td>PURCHASE ORDER DATE</td>
</tr>
<tr>
<td>EDD</td>
<td>DATE</td>
<td>8</td>
<td>ESTIMATED DELIVERY DATE</td>
</tr>
<tr>
<td>RECDATE</td>
<td>DATE</td>
<td>8</td>
<td>RECEIPT DATE</td>
</tr>
<tr>
<td>PART</td>
<td>LOG</td>
<td>1</td>
<td>PARTIAL SHIPMENT?</td>
</tr>
<tr>
<td>PKUPDATE</td>
<td>DATE</td>
<td>8</td>
<td>DATE MATERIAL PICKED UP</td>
</tr>
<tr>
<td>CANC</td>
<td>LOG</td>
<td>1</td>
<td>REQUIREMENT CANCELED?</td>
</tr>
<tr>
<td>CANCDATE</td>
<td>DATE</td>
<td>8</td>
<td>CANCELLATION DATE</td>
</tr>
<tr>
<td>RTNREQDATE</td>
<td>DATE</td>
<td>8</td>
<td>RETURN REQUEST DATE</td>
</tr>
<tr>
<td>ACTPRICE</td>
<td>NUM</td>
<td>12</td>
<td>ACTUAL PRICE</td>
</tr>
<tr>
<td>ADP</td>
<td>CHAR</td>
<td>1</td>
<td>ADP CODE</td>
</tr>
<tr>
<td>ELEMENT</td>
<td>TYPE</td>
<td>WIDTH</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>----------</td>
<td>------</td>
<td>-------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>DEPTEXP</td>
<td>LOG</td>
<td>1</td>
<td>DEPARTMENT EXPENSE ITEM?</td>
</tr>
</tbody>
</table>
## I. TABLE 9: DATABASE INDEXES

<table>
<thead>
<tr>
<th>DATABASE</th>
<th>PRODUCTION INDEX FILE</th>
<th>INDEX</th>
</tr>
</thead>
</table>
| ACCTS.DBF | ACCTS.MDX             | DATERECVD  
|          |                        | JONPI (JON+PI)  
|          |                        | PI  
|          |                        | REPORT (FUNDTYPE+JON)  
|          |                        | DIRECT (JON+PI)  
|          |                        | JON  
|          |                        | NAME  
|          |                        | PI  
| DACCTS.DBF | DACCTS.MDX             | DIRECT (JON+PI)  
| PERSONNE.DBF | PERSONNE.MDX      | FACSTAFF  
|          |                        | IDCODE  
|          |                        | LASTNAME  
|          |                        | PI  
| LABOR1.DBF | LABOR1.MDX             | DIRECT (JON+PI)  
|          |                        | IDCODE  
|          |                        | JON  
|          |                        | LASTNAME  
|          |                        | PAYROLL  
|          |                        | (DTOC(PPE)+JON)  
|          |                        | PI  
|          |                        | PPE  
| TEMPLAB.DBF | TEMPLAB.MDX             | IDCODE  
|          |                        | JON  
|          |                        | LASTNAME  
|          |                        | PAYROLL  
|          |                        | PI  
| TRAVEL.DBF | TRAVEL.MDX             | DIRECT (JON+PI)  
|          |                        | DOCNR  
|          |                        | ESTRNDATE  
|          |                        | JON  
|          |                        | LASTNAME  
|          |                        | PI  
|          |                        | PREPDATE  
|          |                        | TRAVDATE  
|          |                        | CANCDATE  
|          |                        | CATEGORY  
|          |                        | DIRECT (JON+PI)  
|          |                        | DOCNR  
|          |                        | DORDER  
|          |                        | EXPELMNT  
|          |                        | JON  
|          |                        | PI  
|          |                        | PKUPDATE  

30
### J. Table 10: Reports

<table>
<thead>
<tr>
<th>Report File</th>
<th>View File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGING.FRGR</td>
<td>AGING.QBE</td>
<td>Requisition Aging History</td>
</tr>
<tr>
<td>APPST30.FRGR</td>
<td>APP30ST.QBE</td>
<td>Flags Faculty/Staff whose appointment date is within 30 days of expiration</td>
</tr>
<tr>
<td>APPSTATU.FRGR</td>
<td>APPSTATU.QBE</td>
<td>Appointment Status</td>
</tr>
<tr>
<td>DELQTRAV.FRGR</td>
<td>DELQTRAV.QBE</td>
<td>Flags Personnel with delinquent travel claims</td>
</tr>
<tr>
<td>FLAGAPP.FRGR</td>
<td>FLAGAPP.QBE</td>
<td>Flags Travel Orders Requiring Command Approval</td>
</tr>
<tr>
<td>INDPAY.FRGR</td>
<td>PAYREC.QBE</td>
<td>Cumulative Payroll Data by Employee</td>
</tr>
<tr>
<td>LABCHGS.FRGR</td>
<td>LABCHGS.QBE</td>
<td>Cumulative Labor Charges by Account</td>
</tr>
<tr>
<td>OTHERPAY.FRGR</td>
<td>OTHERPAY.QBE</td>
<td>Cumulative Labor Charges by Account for funds managed by other depts</td>
</tr>
<tr>
<td>PERSON.FRGR</td>
<td>PERSON.QBE</td>
<td>Personnel Roster</td>
</tr>
<tr>
<td>SUPCHG.FRGR</td>
<td>SUPCHG.QBE</td>
<td>Cumulative Optar Charges by Account</td>
</tr>
<tr>
<td>SUPSTAT.FRGR</td>
<td>SUPSTAT.QBE</td>
<td>Supply Requisition Status of All Requisitions</td>
</tr>
<tr>
<td>TRAVPKUP.FRGR</td>
<td>TRAVPKUP.QBE</td>
<td>Flags Travel Orders with pickup action due</td>
</tr>
<tr>
<td>TRAVSTAT.FRGR</td>
<td>TRAVSTAT.QBE</td>
<td>Cumulative Travel Charges by Account</td>
</tr>
<tr>
<td>EXPSUM.PRG</td>
<td>N/A</td>
<td>Summary Account Status of Reimbursable, Direct and Indirect Funds</td>
</tr>
<tr>
<td>DIRECT.PRG</td>
<td>N/A</td>
<td>Faculty allocations of &quot;NA&quot; Funding and Financial Status of Each</td>
</tr>
</tbody>
</table>
APPENDIX B: FMIS PROGRAMS

A. EXPSUM.PRG

*:******************************************************************************
*: Program: EXPSUM.PRG
*: System: Financial Management Information System
*: Authors: LCDR N. S. Ford & LT N. W. Zimmon
*: Uses: ACCTS.DBF
*: PERSONNE.DBF
*: LABOR1 DBF
*: SUPPLY.DBF
*: TRAVEL.DBF
*: MDX files: ACCTS.MDX
*: PERSONNE.MDX
*: LABOR1.MDX
*: SUPPLY.MDX
*: TRAVEL.MDX
*:******************************************************************************

SET TALK OFF
CLEAR

*---Open databases
SELECT A
USE accts ORDER jon
SELECT B
USE personne ORDER idcode
SELECT C
USE labor1 ORDER jon
SELECT D
USE supply ORDER jon
SELECT E
USE travel ORDER jon
*---Set relationships
SELECT A
SET RELATION TO jon INTO labor1, jon INTO supply, jon INTO; travel
SELECT C
SET RELATION TO idcode INTO personne

*---Print the report
ON ESCAPE RETURN

32
PLINENO = 0
SELECT A
GO TOP

***--Allow for selection of reporting period.
SET ORDER TO REPORT
enddate = { / / }
fiscal = " "
@ 8,20 SAY "Period ending date for report: " GET enddate
READ
@ 10,20 SAY "Fiscal year: " GET fiscal PICTURE "99"
READ

***--Establish grand total and other variables.
ggttlfunds = 0
ggtravel = 0
ggttlchgtg = 0
ggtbal = 0
ggoptar = 0
ggttlchgtg = 0
ggobal = 0
ggconsupamt = 0
ggconsupchg = 0
ggconsupbal = 0
gglobal = 0
ggsjumplnay = 0
global = 0
ggtchgtg = 0
ggendbal = 0
advance = 0
pgno = 1
CLEAR
?
?
?

***--Notify user of abort option.
?SPACE(5),"Press the End key to abort print job..."

SET CONSOLE OFF
SET PRINT ON
DO WHILE .NOT. EOF() .AND. INKEY() # 2

***--Advance page at end of Reimbursable Account section.
   IF advance = 1
       EJECT
       _PLINENO = 0
       advance = 0
   ENDIF
?"Page No. ", pgno PICTURE "99"
?DATE()
?

33
---Report Heading
SPACE(60),"ADMINISTRATIVE SCIENCES
DEPARTMENT",SPACE(30),;
"REVISED ",enddate
SPACE(68),"FY ",fiscal," FUND STATUS"
?"REIMBURSABLE FUND STATUS"
?
?"ACCT ",","TOTAL"," ","TRAVEL";
" ","TRAVEL"," ","TRAVEL";
" ","OPTAR"," ","OPTAR";
" ","OPTAR"," ","CONSUP"," ","CONSUP";
" ","CONSUP"," ","LABOR"," ","LABOR";
" ","LABOR"," ","TOTAL"," ","ACCT";
" ","EXP"
?"NR ","PI"," AUTH"," ",&","AUTH";
" ","EXP"," ","BAL"," ",&","AUTH";
" ","EXP"," ","BAL"," ",&","AUTH";
" ","EXP"," ","BAL"," ",&","AUTH";
" ","EXP"," ","BAL"," ",&","EXP";
"
"BAL"," ",&","DATE"
?REPLICATE("-",210)

---Establish subtotal variables.
gttlfunds = 0
gtravel = 0
gttltchg = 0
gtbal = 0
gooptar = 0
gtlochg = 0
gobal = 0
gconsupamt = 0
gconsupchg = 0
gconsupbal = 0
glabor = 0
gsumpay = 0
glbal = 0
gtchg = 0
gendbal = 0

---Search Account database for reimbursable type accts
---and sum individual account charges.
SCAN FOR fundtype = "R"
typefund = fundtype
DO WHILE fundtype = typefund .AND. .NOT. EOF()
   job = jon
   sumpay = 0
ttlochg = 0
tttlchg = 0
tconsup = 0
   ---Sums Labor charges.
   DO WHILE jon = job .AND. a->fundtype = typefund;

34
.AND. .NOT. EOF()
SELECT C
paydate = ppe
DO WHILE ppe = paydate .AND. jon = job:
   .AND. a->fundtype = typefund .AND. .NOT.;
   EOF()
   IF ppe <= enddate
      sumpay = sumpay + totpay
   ENDF
   SKIP
ENDDO
ENDDO
SELECT A
*---Sums Supply charges.
DO WHILE jon = job .AND. a->fundtype = typefund;
   .AND. .NOT. EOF()
SELECT D
docdate = dorder
DO WHILE dorder = docdate .AND. jon = job .AND.;
a->fundtype = typefund .AND. .NOT. EOF()
   IF dorder <= enddate .AND. CANC # .T.
      IF category = "CS"
         tconsup = tconsup +;
         IIF(actprice=0, estprice, actprice)
      ELSE
         ttlochg = ttlochg +;
         IIF(actprice=0, estprice, actprice)
      ENDF
     SKIP
E N D D O
SELECT A
*---Sums Travel charges.
DO WHILE jon = job .AND. a->fundtype = typefund;
   .AND. .NOT. EOF()
SELECT E
pdate = prepdate
DO WHILE prepdate = pdate .AND. jon = job .AND.;
a->fundtype = typefund .AND. .NOT. EOF()
   IF prepdate <= enddate .AND. CANC # .T.
      priceest = perdexp + transexp + regexp
      ttltchg = ttltchg + IIF(actamt=0,;
                                 priceest, actamt)
   ENDIF
   SKIP
ENDDO
*---Prints individual account totals and increments
*---subtotal amounts.
SELECT A
?jon, PI, TRANSFORM(ttlfunds,"9,999,999.99"),;
    TRANSFORM(travel,"9,999,999.99"),TRANSFORM;
    (ttltchg,"999,999.99"),TRANSFORM;
    (travel-ttltchg,"9,999,999.99"),;
    TRANSFORM(optar,"9,999,999.99"),TRANSFORM;
    (ttlochg,"999,999.99"),TRANSFORM;
    (optar-ttlochg,"9,999,999.99"),;
    TRANSFORM(consupamt,"999,999.99"),;
    TRANSFORM(tconsup,"999,999.99"),
    TRANSFORM(consupamt-tconsup,"999,999.99"),;
    TRANSFORM(labor,"9,999,999.99"),TRANSFORM;
    (sumpay,"999,999.99"),TRANSFORM;
    (labor-sumpay,"9,999,999.99"),;
    TRANSFORM(sumpay+ttlochg+ttltchg,;
    "9,999,999.99"),TRANSFORM(tltfunds-;
    sumpay-ttlochg-ttltchg,"9,999,999.99"),;
    SPACE(3),expdate
gttlfunds = gttlfunds + ttlfunds
gtravel = gtravel + travel
gttltchg = gttltchg + ttltchg
gtbal = gtbal + (travel-ttltchg)
goptar = goptar + optar
gtlochg = gtlochg + ttlochg
gobal = gobal + (optar - ttlochg)
gconsupamt = gconsupamt + consupamt
gconsupchg = gconsupchg + tconsup
gconsupbal = gconsupbal + (consupamt-tconsup)
glabor = glabor + labor
gsumpay = gsumpay + sumpay
glbal = glbal + (labor-sumpay)
gtchg = gtchg + (sumpay+ttlochg+ttltchg)
gendbal = gendbal + (ttlfunds-sumpay-ttlochg-;
    ttltchg)

SKIP

ENDDO
ENDSCAN

*---Prints total account charges and increments grand totals.
*---REPLICATE("-",210)
"TOTAL:" ,TRANSFORM(gttlfunds,"9,999,999.99"),;
    TRANSFORM(gtravel,"9,999,999.99"),TRANSFORM;
    (gtltchg,"9,999,999.99"),TRANSFORM;
    (gtbal,"9,999,999.99"),TRANSFORM(goptar,;
    "9,999,999.99"),TRANSFORM(gtlochg,;
    "9,999,999.99"),TRANSFORM(gobal,"9,999,999.99"),;
    TRANSFORM(gconsupamt,"999,999.99"),;
    TRANSFORM(gconsupchg,"999,999.99"),;
    TRANSFORM(gconsupbal,"999,999.99"),;

36
TRANSFORM(glabor,"9,999,999.99") ,TRANSFORM;
 (gsumpay,"9,999,999.99") ,TRANSFORM(glbal,;
 "9,999,999.99") ,TRANSFORM(gtchg,"9,999,999.99") ,;
 TRANSFORM(gendbal,"9,999,999.99")

?REPLICATE("-",210)
ggttlfunds = ggttlfunds + gttlfunds
ggttravel = ggttravel + gtravel
ggttlitcghg = ggttlitcghg + gttlitcghg
 ggtbal = ggtbal + gtbal
ggoptar = ggoptar + goptar
ggttloch = ggttloch + gttloch
 ggbal = ggbal + gbal
ggconsupamt = ggconsupamt + gconsupamt
 ggcconsupcghg = ggcconsupcghg + gconsupcghg
 ggcconsupbal = ggcconsupbal + gconsupbal
 gglabor = gglabor + glabor
ggsumpay = ggsumpay + gsunpay
gglbal = gglbal + glbal
 ggtchghg = ggtchghg + gtchghg
ggendbal = ggendbal + gendbal
 advance = advance + 1
 pgno = pgno + 1
ENDDO
IF INKEY() = 2
 SET PRINT OFF
 EJECT
 SET CONSOLE ON
 RETURN
ENDIF

GO TOP
*---Allows aborting of print job.
DO WHILE .NOT. EOF() .AND. INKEY() # 2

 IF advance = 1
 EJECT
 _PLINENO = 0
 advance = 0
ENDIF
 ?"Page No. ", pgno PICTURE "99"
*---Page heading.
 ?DATE()
 ?
 ?SPACE(60), "ADMINISTRATIVE SCIENCES
 DEPARTMENT", SPACE(30),;
 "REVISED ", enddate
 ?SPACE(68), "FY ", fiscal, " FUND STATUS"
 ?"DIRECT/INDIRECT FUND STATUS"
 ?

37
"ACCT","","INITIAL","","ACTUAL","","
"TRAVEL","","TRAVEL","","TRAVEL",";
"OPTAR","","OPTAR","","OPTAR",";
"OPTAR","","CONSUP","","CONSUP",";
"","CONSUP","","TOTAL"," 
"ACCT","'',"FINAL"

"NR","PI"," AUTH"," AUTH",";
"","AUTH","","EXP","","BAL",";
"","AUTH","","EXP","","BAL",";
"","AUTH","","EXP","","BAL",";

REPLICATE(",",185)
gintlfunds = 0
gttlfunds = 0
gtravel = 0
gttlochg = 0
gtbal = 0
goptar = 0
gttlochg = 0
gobal = 0
gtchg = 0
gconsupamt = 0
gconsupchg = 0
gconsupbal = 0
gendbal = 0
gfinbal = 0

*---Sums account charges for Direct/Indirect accounts.
SCAN FOR fundtype = "0"
    typefund = fundtype
    DO WHILE fundtype = typefund .AND. .NOT. EOF() 
        job = jon
        ttlochg = 0
        ttiltchg = 0
        regch = 0
        tconsup = 0
*---Sums Supply charges.
    DO WHILE jon = job .AND. a->fundtype = typefund ; .AND. .NOT. EOF() 
        SELECT D 
        docdate = dorder
        DO WHILE dorder = docdate .AND. jon = job .AND.; 
            a->fundtype = typefund .AND. .NOT. EOF() 
            IF dorder <= enddate .AND. CANC # .T.
                IF category = "CS"
                    tconsup = tconsup +; 
                    IIF(actprice=0,estprice,actprice)
                ELSE 
                    ttlochg = ttlochg +; 
                    IIF(actprice=0,estprice,actprice) 
            ENDIF 
        ENDDO 
    ENDWHILE 
ENDDO 

38
SKIP
ENDDO
ENDDO
SELECT A

*--Sums Travel charges.
DO WHILE jon = job .AND. a->fundtype = typefund .AND. .NOT. EOF()
SELECT E
pdate = prepdate
DO WHILE prepdate = pdate .AND. jon = job .AND. a->fundtype = typefund .AND. .NOT. EOF()
IF prepdate <= enddate .AND.:
   CANCE # . T.
   priceest = perdexp+transexp
   regchg = regchg + IIF(actreg=0, regexp, actreg);
   ttltchg = ttltchg + IIF(actamt=0, priceest, actamt-IIF(actreg=0, regexp, actreg))
ENDIF
ENDDO
SKIP
ENDDO
ENDDO

*--Prints individual account charges and
*--increments subtotals.
SELECT A

?jon, PI, TRANSFORM(inttlfunds,"9,999,999.99"),;
TRANSFORM(travel,"9,999,999.99"), TRANSFORM;
( ttltchg,"9,999,999.99" ), TRANSFORM;
(travel-ttltchg,"9,999,999.99"),;
TRANSFORM(optar,"9,999,999.99"),;
TRANSFORM(ttlochg+regchg,"999,999.99"),;
TRANSFORM(optar-(ttlochg+regchg),"999,999.99"),;
TRANSFORM(consumpt,"999,999.99"),;
TRANSFORM(tconsup,"999,999.99"),;
TRANSFORM(consumpt-tconsup,"999,999.99"),;
TRANSFORM((ttlochg+regchg)+ttltchg+;
tconsup,"9,999,999.99"), TRANSFORM(ttlfunds-;
(ttlochg+regchg)-ttltchg-tconsup,;
"9,999,999.99"), TRANSFORM(inttlfunds-;
(ttllochg+regchg)-ttltchg-;
tconsup,"9,999,999.99")
ginttlfunds = ginttlfunds + inttlfunds
gttlffunds = gttlfunds + ttfunds
gtravel = gtravel + travel
gttltchg = gttltchg + ttltchg
gtbal = gtbal + (travel-ttltchg)
goptar = goptar + optar
gtlochg = gtlochg + (ttlochg+regchg)
gobal = gobal + (optar - (ttlochg+regchg))
gconsupamt = gconsupamt + consupamt
gconsupchg = gconsupchg + tconsup
gconsupbal = gconsupbal + (consupamt-tconsup)
gtchg = gtchg + (ttlochg+regchg+ttltchg+tconsup)
gendbal = gendbal + (ttlfunds-(ttlochg+regchg)-
ttltchg-tconsup)
gfinbal = gfinbal + (intttlfunds-(ttlochg+regchg)-
ttltchg-tconsup)

SKIP

ENDDO
ENDSCAN

*---Prints account charges subtotals and increments grand
*---totals.

?REPLICATE("-",185)

?"TOTAL:",SPACE(5),TRANSFORM(gintttlfunds,"9,999,999.99"),;
TRANSFORM(gtravel,"9,999,999.99"),;TRANSFORM(gttltchg,
"9,999,999.99"),TRANSFORM(gtbal,"9,999,999.99"),;
TRANSFORM(goptar,"9,999,999.99"),TRANSFORM(gttlochg,
"999,999.99"),TRANSFORM(gobal,"9,999,999.99"),;
TRANSFORM(gconsupamt,"999,999.99"),TRANSFORM;
(gconsupchg,"999,999.99"),TRANSFORM(gconsupbal,
"999,999.99"),TRANSFORM(gtbal,"9,999,999.99"),;
TRANSFORM(ggendbal,"9,999,999.99"),TRANSFORM;
(gfinbal,"9,999,999.99")

?REPLICATE("-",185)
ggttlfunds = ggttlfunds + gttlfunds
ggtravel = ggtravel + gtravel
ggtltchg = ggtltchg + gttltchg
ggtbal = ggbal + gtbal
ggoptar = ggoptar + goptar
ggttlochg = ggttlochg + gtlochg
ggobal = ggobal + gobal
ggconsupamt = ggconsupamt + gconsupamt
ggconsupchg = ggconsupchg + gconsupchg
ggconsupbal = ggconsupbal + gconsupbal
ggtch = ggtch + gtch
gggendbal = ggendbal + gendbal
advance = advance + 1
pgno = pgno + 1

ENDDO

*---Allows aborting of print job.

IF INKEY() = 2
SET PRINT OFF
EJECT
SET CONSOLE ON
RETURN
ENDIF
?
SELECT A
GO TOP
*---Sets index for detailed Direct/Indirect account charges
*---report.
SET ORDER TO REPORT
*---Establishes required variables.
tregtrav = 0
titototrav = 0
tpcctrav = 0
tedototrav = 0
tttgmtrav = 0
gttltlchg = 0
gtbal = 0
tsupchgl = 0
thonorchg = 0
tregchg = 0
gconsupchgl = 0
gconsupbal = 0
gttlochg = 0
gobal = 0
*---Report heading.
? "DETAILED DIRECT/INDIRECT FUND STATUS"
? "ACCT", "REGULAR", "ITO", "PCC";
"EDO", "TQM", "TOTAL";
"TRAVEL", "SUPPLY";
"REG";
"SUP EXP", "OPTAR", "CONSUP", "CONSUP"
? "NR", "TRAVEL", "TRAVEL", "TRAVEL", ";
"TRAVEL EXP", "BALANCE", "EXP";
"HONOR", "FEE", "TOTAL";
"BALANCE", "EXP", "BALANCE"
?REPLICATE(" ", 180)
*---Sum individual account charges.
DO WHILE .NOT. EOF() .AND. INKEY () # 2
job = jon
ttlochg = 0
ttltchg = 0
regchg = 0
tconsup = 0
*---Select Direct/Indirect accounts.
DO WHILE jon = job .AND. .NOT. EOF()
IF fundtype = "O"
SELECT E
pcctrav = 0
edototrav = 0

tqmtrav = 0
itotrav = 0
SCAN FOR jon = job WHILE a->fundtype = "0"
   *--Sum Travel charges.
   IF prepdate <= enddate .AND. CANC # .T.
      priceest = perexp+transexp
   IF travtype = "ITO"
      itotrav = itotrav + IIF(actamt=0,;
      priceest,actamt - IIF(actreg=0,;
      regexp,actreg))
   ELSE
      IF travtype = "PCC"
         pccctrav = pccctrav + IIF(actamt=0,;
         priceest,actamt-IIF(actreg=0,;
         regexp,actreg))
      ELSE
         IF travtype = "EDO"
            edotrav = edotrav + IIF(actamt=0,;
            priceest,actamt-IIF(actreg=0,;
            regexp,actreg))
         ELSE
            IF travtype = "TQM"
               tqmtrav = tqmtrav;
               + IIF(actamt=0,priceest,;
               actamt-IIF(actreg=0,;
               regexp,actreg))
               ENDIF
            ENDIF
         ENDIF
      ENDIF
   ENDIF
   ENDIF
ENDSCAN
SELECT D
honorchg = 0
regfee = 0
SCAN FOR jon = job WHILE a->fundtype = "0"
   *--Sum Supply charges.
   IF dorder <= enddate .AND. CANC # .T.
      IF category = "HO"
         honorchg = honorchg +;
         IIF(actprice=0,estprice,actprice)
      ELSE
         IF category = "CS"
            tconsup = tconsup +;
            IIF(actprice=0,estprice,actprice)
         ELSE
            ttlochg = ttlochg +;
            42
IF (actprice=0, estprice, actprice)
ENDIF

ENDIF

ENDSCAN

SELECT A

*---Print individual account charges and increment subtotal amounts.

?j on, SPACE (3), TRANSFORM (ttltchgl-itotrav-pcctrav-;
edotrav-tqmtrav, "999,999.99"), SPACE (1),
TRANSFORM (itotrav, "999,999.99"),
SPACE (1), TRANSFORM;
(pcctrav, "999,999.99"), SPACE (1), TRANSFORM;
(edotrav, "999,999.99"), TRANSFORM;
(tqmtrav, "999,999.99"), SPACE (1), TRANSFORM;
(tttlchgl, "999,999.99"), SPACE (1),
TRANSFORM (travel-t ttlchgl,;
"999,999.99"), SPACE (1), TRANSFORM (ttlochg,;
"999,999.99"), SPACE (6), TRANSFORM;
(honorchg, "999,999.99"), SPACE (1), TRANSFORM;
(regchg, "999,999.99"), TRANSFORM;
(ttlochg+honorchg+regchg,"999,999.99"),;
TRANSFORM (optar-(ttlochg+regchg+honorchg),;
"999,999.99"), TRANSFORM (tconsup,"999,999"),
TRANSFORM (consupamt-tconsup,"999,999.99")

 tregtrav = tregtrav + (ttltchgl-itotrav-pcctrav-;
edotrav-tqmtrav)
titotrav = titotrav + itotrav
tpcctrav = tpcctrav + pcctrav
tedotrav = tedotrav + edotrav
ttmqtrav = ttqmtrav + tmqtrav
gttltchgl = gttltchgl + tttltchgl
gtbal = gtbal + (travel-tttltchgl)
ttsupchg = ttsupchg + (ttlochg-honorchg)
thonorchg = thonorchg + honorchg
tregchg = tregchg + regchg
gtlochg = gtlochg + (ttlochg+regchg+honorchg)
gobal = gobal + (optar-(ttlochg+regchg+honorchg))
gconsupchg = gconsupchg + tconsup
gconsupbal = gconsupbal + (consupamt-tconsup)

ENDIF

SKIP

ENDDO

*---Enables aborting of print job.

IF INKEY () = 2
SET PRINT OFF
EJECT
SET CONSOLE ON
RETURN
ENDIF

*---Print detailed total account charges.
?REPLICATE("-",180)
?"TOTAL:",SPACE(2),TRANSFORM(tregtrav,"999,999.99"),;
    SPACE(1), TRANSFORM(titototrav,"999,999.99"),;
    SPACE(1),TRANSFORM(tpcctorav,"999,999.99"),;
    SPACE(1),TRANSFORM(tedotrav,;
"999,999.99"),SPACE(1),TRANSFORM(ttqmtrav,"999,999.99"),;
TRANSFORM(gttltchg,"999,999.99"),SPACE(1),TRANSFORM;
(gtbal,"999,999.99"),SPACE(1),TRANSFORM;
(tsupchg,"999,999.99"),SPACE(7),TRANSFORM;
(thonorchg,"99,999.99"),;
SPACE(1),TRANSFORM(tregchg,"99,999.99"),TRANSFORM;
(gttlochg,"999,999.99"),SPACE(2),TRANSFORM(gobal,;
"99,999.99"),TRANSFORM(gconsupchg,"999,999.99"),;
TRANSFORM(gconsupbal,"999,999.99")
?REPLICATE("-",180)
*---Eject page
EJECT
_PLINENO = 0

*---Page heading.
?"Page No. ", pgno PICTURE "99"
?DATE()
?
?SPACE(64),"ADMINISTRATIVE SCIENCES DEPARTMENT",SPACE(30),;
"REVISED ",enddate
?SPACE(72),"GRAND TOTALS"
?SPACE(62),"REIMBURSEABLE/DIRECT/INDIRECT FUNDS"
?
?
?"ACCT ",",",",TOTAL",","",",TRAVEL";"
","",",TRAVEL"",","",",OPTAR";"
","",",OPTAR"",","",",OPTAR"",","",",CONSUP";"
","",",CONSUP"",","",",LABOR";"
","",",LABOR"",","",",LABOR";
"TOTAL",","",",ACCT"
?"NR ","PT",","",",AUTH","",",AUTH","",",AUTH","",",AUTH","",",AUTH","",",AUTH","",",AUTH","",",AUTH";
" "
?"BAL",","",",BAL"","",",BAL",","",",BAL","",",BAL","
"EX","",",EX","",",EX","",",EX","",",EX","",",EX";
" "
?"BAL",","",",BAL","",",BAL",","",",BAL","",",BAL","
"EX","",",EX","",",EX","",",EX","",",EX","",",EX";
"
?*---Print grand total of all account charges.
?REPLICATE("-",190)
?"GRAND"
?"TOTAL:",TRANSFORM(ggttlfunds,"9,999,999.99"),;
TRANSFORM(ggttrav,"9,999,999.99"),;
TRANSFORM(gttltltchg,"9,999,999.99")

TRANSFORM(ggtbal,"9,999,999.99"),
TRANSFORM(ggoptar,"999,999.99"),TRANSFORM;
(ggtnlochg,"999,999.99"),TRANSFORM(ggobal,"999,999.99"),
TRANSFORM(ggconsupamt,"999,999.99"),TRANSFORM;
(ggconsupchq, "999,999.99"),TRANSFORM;
(ggconsupbal,"999,999.99"),; 
TRANSFORM(gglabor,"9,999,999.99"),;
TRANSFORM(ggsumpay,"9,999,999.99"),TRANSFORM;
(ggllbal,"9,999,999.99"),TRANSFORM;
(ggtchq,"9,999,999.99"),; 
TRANSFORM(ggendbal,"9,999,999.99")
?REPLICATE("=",190)

*---Close databases and clear printer.
CLOSE DATABASES
EJECT
SET PRINT OFF
SET CONSOLE ON
*: EOF: EXPSUM.PRG
B. DIRECT.PRG

***:*******************************************************
***: Program: DIRECT.PRG
***: System: Financial Management Information System
***: Authors: LCDR N. S. Ford & LT N. W. Zimmon
***: Calls: D_PGBRK.PRG
***: Uses: DACCTS.DBF
***: : PERSONNE.DBF
***: : SUPPLY.DBF
***: : TRAVEL.DBF
***: MDX files: DACCTS.MDX
***: : PERSONNE.MDX
***: : SUPPLY.MDX
***: : TRAVEL.MDX
***:

SET TALK OFF
CLEAR
***---Open databases

SELECT A
USE daccts ORDER PI
SELECT B
USE personne ORDER idcode
SELECT C
USE supply ORDER DIRECT
SELECT D
USE travel ORDER DIRECT
***---Set relationships

SELECT A
SET RELATION TO jon INTO supply, jon INTO travel
***---Print the report
ON ESCAPE RETURN
_PLINENO = 0
ON PAGE AT LINE 57 DO d_pgbkrk
enddate = ( / / )
fiscal = " "
@ 10,20 SAY "Period ending date for report: " GET enddate
READ
@ 12,20 SAY "Fiscal year: " GET fiscal PICTURE "99"
READ
SET STATUS OFF
SET SCOREBOARD OFF

46
CLEAR
? 
?
?SPACE(5), "Press the End key to abort print job..."
SET CONSOLE OFF
SET PRINT ON
pgno = 1

?"Page No. ", pgno PICTURE "99"
?DATE()

?SPACE(60),"ADMINISTRATIVE SCIENCES DEPARTMENT",SPACE(30),;
"REVISED ",enddate
?SPACE(68),"FY ",fiscal," NA FUND STATUS"
?
?"", "TRAVEL"," ", "TRAVEL";
" ", "TRAVEL"," ", "OPTAR"," ", "OPTAR"," ";
"REG"," ", "OPTAR +"," ", "OPTAR"," ", "CONSUP";
" ", "CONSUP"," ", "CONSUP"," ", "TOTAL";
" ", "ENDING"
?"NAME "," PI"," AUTH"," EXP"," 
" ", "BAL"," ", "AUTH"," ", "EXP";
" ", "FEES"," ", "REG FEES"," ", "BAL";
" ", "AUTH"," ", "EXP"," ", "BAL";
" ", "EXP"," ", "BAL"
?
?REPLICATE("--",180)
gttlfunds = 0
gtravel = 0
gttltchgd = 0
goptar = 0
gttlochg = 0
gregchg = 0
gobal = 0
gconsupamt = 0
gconsupchg = 0
gconsupbal = 0
gopreg = 0
gtchg = 0
gendbal = 0

*---Sum "NA" charges for each Principal Investigator.
DO WHILE .NOT. EOF() .AND. INKEY() # 2

pir = PI
ttlochg = 0
tttltchgd = 0
regchg = 0
tconsup = 0
DO WHILE PI = pir .AND. .NOT. EOF()

*---Sums Optar and Contract Support charges.
SELECT C
SCAN FOR PI = pir WHILE jon = "NA"
    IF dorder <= enddate .AND. CANC # .T.
        IF category = "CS"
            tconsup = tconsup;
            + IIF(actprice=0,estprice,actprice)
        ELSE
            ttlochg = ttlochg;
            + IIF(actprice=0,estprice,actprice)
        ENDIF
    ENDIF
ENDSCAN
* ---Sums Travel charges.
SELECT D
SCAN FOR PI = pir WHILE jon = "NA"
    IF prepdate <= enddate .AND. CANC # .T.
        priceest = perdexp + transexp
        regchg = regchg + IIF(actreg=0,regexp,;
            actreg)
        ttltchg = ttltchg + IIF(actamt=0,priceest,;
            actamt- IIF(actreci=0,regexp,actreg))
    ENDIF
ENDSCAN
* ---Prints Travel, Optar and Contract Support charges
* ---for each Principal Investigator.
SELECT A
?name,PI,TRANSFORM(travel,"999,999.99"),TRANSFORM;
    (ttltchg,"999,999.99"),TRANSFORM(travel-ttltchg,;
        "999,999.99"),TRANSFORM(optar,"999,999.99"),;
    TRANSFORM(ttltchg,"999,999.99"),;
    TRANSFORM(regchg,"999,999.99"),TRANSFORM;
    (ttlochg+regchg,"9,999,999.99"),;
    TRANSFORM(optar-(ttlochg+regchg),"999,999.99"),;
    TRANSFORM(consupamt,"999,999.99"),TRANSFORM;
    (tconsup,"999,999.99"),TRANSFORM(consupamt-;
        tconsup, "999,999.99"),TRANSFORM(ttltchg+ttlochg+;
        regchg+tconsup,"9,999,999.99"),;
    TRANSFORM(optar+travel+consupamt-(ttltchg+ttlochg+;
        regchg+tconsup),"9,999,999.99")
gtravel = gtravel + travel
gttltchg = gttltchg + ttltchg
gtbal = gtbal + (travel-ttltchg)
goptar = goptar + optar
gttlochg = gttlochg + ttlochg
gregchg = gregchg + regchg
gopreg = gopreg + (ttlochg + regchg)
gobal = gobal + (optar - (ttlochg+regchg))
gconsupamt = gconsupamt + consupamt
gconsupchg = gconsupchg + tconsup

48
gconsupbal = gconsupbal + (consupamt - tconsup)
gtchg = gtchg + (ttlochg + regchg + ttltchg + tconsup)
gendbal = gendbal + ((travel + optar + consupamt) - ttlochg; 
    - regchg - ttltchg - tconsup)

SKIP
ENDDO
ENDDO

*--- Allows aborting of print job when End key is pressed.
IF INKEY() = 2
    EJECT
    SET PRINT OFF
    SET CONSOLE ON
ENDIF

*--- Prints total "NA" charges by charge type.
?REPLICATE("=", 180)
"TOTAL: ", SPACE(10), TRANSFORM(gtravel, "9,999,999.99"), ;
    TRANSFORM(gttltchg, "9,999,999.99"), TRANSFORM;
    (gtbal, "9,999,999.99"), TRANSFORM;
    (goptar, "9,999,999.99"), TRANSFORM;
    (gttlochg, "999,999.99"), TRANSFORM;
    (gregchg, "9,999,999.99"), ;
    TRANSFORM(gopreg, "999,999.99"), ; TRANSFORM;
    (gobal, "9,999,999.99"), TRANSFORM;
    (gconsupamt, "999,999.99"), ;
    TRANSFORM(gconsupchg, "999,999.99"), TRANSFORM;
    (gconsupbal, "999,999.99"), TRANSFORM;
    (gtchg, "9,999,999.99"), TRANSFORM(gendbal, "9,999,999.99")

CLOSE DATABASES
EJECT
SET PRINT OFF
SET CONSOLE ON
SET STATUS ON
*: EOF: DIRECT.PRG
C. ACCTPROC.PRG

*:*******************************************
*: Program: ACCTPROC.PRG
*: System: Financial Management System
*: Author: LCDR N. S. Ford & LT N. W. Zimmon
*: Procs & Fncts: ISJON()
*: : ISPI()
*: : ISIDCODE()
*: : PAYROLL
*: : PAYEDIT
*: : SUPIMP
*: : SUPUP
*: : RECRENDEX
*: : D_PGBRK
*: : RESTORE
*: : SRCHPER
*: : SRCHJON
*: : SRCHDNR
*: : SRCHALOT
*: : SRCHTRAV
*: : SRCHLAB
*:-------------------------------------------
*:This overall procedure file is named ACCTPROC

-------------------Validate Job Order Number
!*---------------------------------------------------------------
!* Function: ISJON()
!*---------------------------------------------------------------
!* Called by: SUPPLY.FMT
!* : DACCTS.FMT
!* : TRAVEL.FMT
!* : LABOR.FMT
!*---------------------------------------------------------------
FUNCTION isjon
PARAMETERS mjon

DO CASE

   *---If user is exiting, do nothing
CASE mjon = " 
   ok = .T.

   *---JON was entered
CASE SEEK(mjon,"Accts")
   ok = .T.
otherwise
ok =.F.
endcode
return (ok)

--------------------Validate Principal Investigator
!*-------------------------------------------------------------------
!* Function: ISPI()
!*
! Called by: SUPPLY.FMT
!* : DACCTS.FMT
!* : TRAVEL.FMT
!*
!*-------------------------------------------------------------------
function ispi
parameters mpi

do case
  *---If user is exiting, do nothing
case mpi = ""
  ok = .T.
  *
  *---PI was entered
case seek(mpi,"Personne")
  ok = .T.
  *
  otherwise
  ok = .F.
  *
endcode
return (ok)

--------------------Validate IDCODE
!*-------------------------------------------------------------------
!* Function: ISIDCODE()
!*
! Called by: LABOR.FMT
!*
!*-------------------------------------------------------------------
function isidcode
parameters midcode

do case
  *---If user is exiting, do nothing
case midcode = ""
  ok = .T.
*---IDCODE was entered
CASE SEEK(midcode,"Personne")
  ok = .T.

OTHERWISE
  ok = .F.

ENDCASE
RETURN (ok)

*---Calculates individual payroll amounts using a temporary
*---database, then adds new records to permanent database
*---file.
!*---------------------------------------------------------------
!*     Procedure: PAYROLL
!*---------------------------------------------------------------
!*     Uses: TEMPLAB.DBF
!*          : PERSONNE.DBF
!*          : LABOR1.DBF
!*---------------------------------------------------------------
!*     MDX files: TEMPLAB.MDX
!*          : PERSONNE.MDX
!*          : LABOR1.MDX
!*---------------------------------------------------------------

PROCEDURE payroll

SET TALK OFF
SELECT A
USE templab ORDER idcode
SELECT B
USE personne ORDER idcode
SELECT C
USE labor1 ORDER idcode
SELECT A
SET RELATION TO idcode INTO personne
GO TOP
DO WHILE .NOT. EOF()
    REPLACE totpay WITH ROUND(((b->basesal/2080) *;
                                 (reghrs+(1.5*othrs))*(l+accrate)),2)
    REPLACE lastname WITH b->lastname
    REPLACE firstname WITH b->firstname
    SKIP
ENDDO
CLOSE DATABASES
USE labor1
APPEND FROM templab
CLOSE DATABASES

SET SAFETY OFF
USE templab
ZAP
CLOSE DATABASES
RETURN

*---Enables editing/updating of payroll records using a temporary
*---database, then adds edited records back to permanent database.

**************************************************************************************************************
*! Procedure: PAYEDIT
*!
*! Uses: TEMPLAB.DBF
*! : LABOR1.DBF
*! : PERSONNE.DBF
*! : ACCTS.DBF
*! MDX files: TEMPLAB.MDX
*! : LABOR1.MDX
*! : PERSONNE.MDX
*! : ACCTS.MDX

PROCEDURE payedit

SELECT A
USE templab
SET DELETED ON
APPEND FROM labor1
REINDEX
SET ORDER TO payroll
GO TOP
SELECT B
USE labor1
SET SAFETY OFF
ZAP
SET SAFETY ON
SELECT C
USE personne ORDER PI
SET RELATION TO PI INTO templab
SELECT D
USE accts ORDER jon
SET RELATION TO jon INTO templab
SELECT A

RETURN
---Enables importing of supply information for supply
---database from remotely located computer.
*------------------*------------------*------------------*------------------*------------------*
!* Procedure: SUPIMP
!*!* Uses: SUPPLY.DBF
!*!* MDX files: SUPPLY.MDX
!*!*------------------*------------------*------------------*------------------*------------------*
PROCEDURE supimp

SET TALK ON && Show progress
USE supply
SET SAFETY OFF
ZAP
APPEND FROM B:\supply & Append from low-density drive
REINDEX
SET SAFETY ON
CLOSE DATABASE
SET TALK OFF && Suppress progress messages
RETURN

---Procedure for importing Account and Personnel databases
---into remotely located computer. Enables validation of
---Account/Personnel information for entering/editing of
---supply transactions.
*------------------*------------------*------------------*------------------*------------------*
!* Procedure: SUPUP
!*!* Uses: ACCTS.DBF
!*!* : DACCTS.DBF
!*!* : PERSONNE.DBF
!*!* MDX files: ACCTS.MDX
!*!* : DACCTS.MDX
!*!* : PERSONNE.MDX
!*!*------------------*------------------*------------------*------------------*------------------*
PROCEDURE supup

SET TALK ON & Show progress
SET SAFETY OFF
USE accts
ZAP
APPEND FROM A:\accts
REINDEX
USE daccts
ZAP
APPEND FROM A:\daccts

54
REINDEX
USE personne
ZAP
APPEND FROM A:\personne
REINDEX
SET TALK OFF && Suppress progress messages
RETURN

*---Procedure for re-indexing databases.
*!****************************************************************************************************
!* Procedure: RECRENDEX
!* *
!* Uses: ACCTS.DBF
!* : DACCTS.DBF
!* : PERSONNE.DBF
!* : LABOR1.DBF
!* : SUPPLY.DBF
!* : TRAVEL.DBF
!* : TEMPLAB.DBF
!* *
!* MDX files: ACCTS.MDX
!* : DACCTS.MDX
!* : PERSONNE.MDX
!* : LABOR1.MDX
!* : SUPPLY.MDX
!* : TRAVEL.MDX
!* : TEMPLAB.MDX
!* *
*!****************************************************************************************************
PROCEDURE recrendx

SET TALK ON && Show progress
USE accts
REINDEX
USE daccts
REINDEX
USE personne
REINDEX
USE labor1
REINDEX
USE supply
REINDEX
USE travel
REINDEX
USE templab
REINDEX
SET TALK OFF && Suppress progress messages
RETURN
*---Page break procedure for "NA Summary Report".
!*--------------------------------------------------------------------------------------
!* Procedure: D_PGBRK
!*--------------------------------------------------------------------------------------
PROCEDURE d_pgbkrk

EJECT PAGE
pgno = pgno + 1
?"Page No. ", pgno PICTURE "99"
?DATE()
?
?SPACE(60), "NA Summary - Continued"
?
?RETURN

*---Procedure for restoring backed-up database files
!*--------------------------------------------------------------------------------------
!* Procedure: RESTORE
!*--------------------------------------------------------------------------------------
!* Uses: ACCTS.DBF
!* : DACCTS.DBF
!* : PERSONNE.DBF
!* : LABOR1.DBF
!* : SUPPLY.DBF
!* : TRAVEL.DBF
!* : TEMPLAB.DBF
!* MDX files: ACCTS.MDX
!* : DACCTS.MDX
!* : PERSONNE.MDX
!* : LABOR1.MDX
!* : SUPPLY.MDX
!* : TRAVEL.MDX
!* : TEMPLAB.MDX
!*--------------------------------------------------------------------------------------
PROCEDURE RESTORE

SET TALK ON && Show progress
SET SAFETY OFF
USE accts
ZAP
APPEND FROM A:\accts
REINDEX
USE daccts
ZAP
APPEND FROM A:\daccts
REINDEX
USE personne
ZAP
APPEND FROM A:\personne
REINDEX
USE labor1
ZAP
APPEND FROM A:\labor1
REINDEX
USE supply
ZAP
APPEND FROM A:\supply
REINDEX
USE travel
ZAP
APPEND FROM A:\travel
REINDEX
USE templab
ZAP
APPEND FROM A:\templab
REINDEX
SET TALK OFF && Suppress progress messages
RETURN

*---The following are procedures for searching databases for
*---editing records.
*!**********************************************************
PROCEDURE srchper
!* Procedure: SRCHPER
!* Calls: PERSONNE.FMT
!* Uses: PERSONNE.DBF
!* MDX files: PERSONNE.MDX
!* Formats: PERSONNE.FMT
*!
*!**********************************************************
PROCEDURE srchper

SET TALK OFF
USE personne ORDER lastname
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
    CLEAR
memlname = SPACE(15)
@10,2 SAY "Enter last name of person to look for: " GET
memlname
READ

*---If nothing entered, exit.
IF memlname = " "
SET FORMAT TO personne
EDIT NOAPPEND
CLOSE FORMAT
searching = .F.
LOOP
ENDIF

*---Try to find that person.
SEEK UPPER(memlname)

IF FOUND()
SET FORMAT TO personne
EDIT NOAPPEND
CLOSE FORMAT
searching = .F.
LOOP
ELSE
@22,0 CLEAR
?"Can't find", memlname
WAIT
ENDIF
ENDDDO
RETURN
*!************************************************************************************************************
*! Procedure: SRCHJON
*!
*!   Calls: ACCTS.FMT
*!
*! Uses: ACCTS.DBF
*!
*! MDX files: ACCTS.MDX
*!
*! Formats: ACCTS.FMT
*!
*!************************************************************************************************************

PROCEDURE srchjon

SET TALK OFF
USE accts ORDER jon
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
    CLEAR
    memjon = SPACE(5)
    @10,2 SAY "Enter Job Order No. to look for: " GET memjon
    READ

    *---If nothing entered, exit.
    IF memjon = " "
        SET FORMAT TO accts
        EDIT NOAPPEND
        CLOSE FORMAT
        searching = .F.
        LOOP
    ENDIF

    *---Try to find that account.
    SEEK UPPER(memjon)

    IF FOUND()
        SET FORMAT TO accts
        EDIT NOAPPEND
        CLOSE FORMAT
        searching = .F.
        LOOP
    ELSE
        @22,0 CLEAR
        ?"Can't find", memjon
        WAIT

59
ENDIF

ENDDO

RETURN

!*-----------------------------------------------------------------------
!*                Procedure: SRCHDNR
!*                Calls: SUPPLY.FMT
!*                Formats: SUPPLY.FMT
!*                *-----------------------------------------------------------------------
PROCEDURE srchdnr
*---Supply database opened through embedded code in the
*---application program

SET TALK OFF
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
CLEAR
memdoc = SPACE(9)
@10,2 SAY "Enter Document No. to look for: " GET memdoc
READ

*---If nothing entered, exit
IF memdoc = " ">
    SET FORMAT TO supply
    EDIT NOAPPEND
    CLOSE FORMAT
    searching = .F.
LOOP
ENDIF

*---Try to find that document number.
SEEK UPPER(memdoc)

IF FOUND()
    SET FORMAT TO supply
    EDIT NOAPPEND
    CLOSE FORMAT
    searching = .F.
LOOP
ELSE
   @22,0 CLEAR
   ?"Can't find", memdoc
   WAIT
ENDIF
ENDDO
RETURN

*!*********************************************************
!*     Procedure: SRCHALOT
!*     Calls: DACCTS.FMT
!*     Formats: DACCTS.FMT
!*     *********************************************************
PROCEDURE srchalot

SET TALK OFF
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
   CLEAR
   memname = SPACE(15)
   @10,2 SAY "Enter Lastname to look for: " GET memname
   READ

   *---If nothing entered, exit.
   IF memname = ""
      SET FORMAT TO daccts
      EDIT NOAPPEND
      CLOSE FORMAT
      searching = .F.
      LOOP
   ENDIF

   *---Try to find that person.
   SEEK UPPER(memname)

   IF FOUND()
      SET FORMAT TO daccts
      EDIT NOAPPEND
      CLOSE FORMAT
      searching = .F.
LOOP

ELSE
    @22,0 CLEAR
    ?"Can't find", memname
    WAIT
ENDIF
ENDDO
RETURN

!* **************************************************************************
!* Procedure: SRCHTRAV
!* *
!* Calls: TRAVEL.FMT
!* *
!* Formats: TRAVEL.FMT
!* *
!* **************************************************************************
PROCEDURE srchtrav
---Travel database opened through embedded code in the
---application program

SET TALK OFF
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
    CLEAR
    memlname = SPACE(15)
    @10,2 SAY "Enter last name of traveler to look for: ";
        GET memlname
    READ

    *---If nothing entered, exit.
    IF memlname = " "
        SET FORMAT TO travel
        EDIT NOAPPEND
        CLOSE FORMAT
        searching = .F.
        LOOP
    ENDF

    *---Try to find that person.
    SEEK UPPER(memlname)

    IF FOUND()
SET FORMAT TO travel
EDIT NOAPPEND
CLOSE FORMAT
searching = .F.
LOOP

ELSE
@22,0 CLEAR
?"Can't find", memlname
WAIT
ENDIF

ENDDO

RETURN

*!******************************************************************************
*! Procedure: SRLCHLAB
*!
*! Calls: LABOR.FMT
*!
*! Formats: LABOR.FMT
*!
*!******************************************************************************
PROCEDURE srchlab

***Templab opened from embeded code in the application
***program.
SET TALK OFF
SET ORDER TO lastname
SET ESCAPE OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET NEAR ON
searching = .T.

DO WHILE searching
    CLEAR
    memlname = SPACE(15)
    @10,2 SAY "Enter Last name of person to look for: ";
    GET memlname
    READ

    ***If nothing entered, exit.
    IF memlname = " "
        SET FORMAT TO labor
        GO TOP
        EDIT NOAPPEND
        CLOSE FORMAT
        searching = .F.
        LOOP

*
**---Try to find that person.**
SEEK UPPER(memlname)

IF FOUND()
    SET FORMAT TO labor
    EDIT NOAPPEND
    CLOSE FORMAT
    searching = .F.
    LOOP
ELSE
    @22,0 CLEAR
    ?"Can't find", memlname
    WAIT
ENDIF

ENDDO
RETURN

*: EOF: ACCTPROC.PRG
Application Documentation for System: RMS.PRG

Application Author: LCDR N. S. Ford and LT N. W. Zimmon

dBASE Version.....: 1.1

Display Application Sign-On Banner: Yes

Main Menu to Open after Sign-On: RMSMAIN.BAR

Sets for Application:

<table>
<thead>
<tr>
<th>Set</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bell</td>
<td>ON</td>
</tr>
<tr>
<td>Carry</td>
<td>OFF</td>
</tr>
<tr>
<td>Centry</td>
<td>OFF</td>
</tr>
<tr>
<td>Confirm</td>
<td>OFF</td>
</tr>
<tr>
<td>Delimiters</td>
<td>OFF</td>
</tr>
<tr>
<td>Display Size</td>
<td>25 lines</td>
</tr>
<tr>
<td>Drive</td>
<td></td>
</tr>
<tr>
<td>Escape</td>
<td>ON</td>
</tr>
<tr>
<td>Path</td>
<td></td>
</tr>
<tr>
<td>Safety</td>
<td>ON</td>
</tr>
</tbody>
</table>

Starting Colors for Application:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Color Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>W+/B</td>
</tr>
<tr>
<td>Heading</td>
<td>W+/B</td>
</tr>
<tr>
<td>Highlight</td>
<td>GR+/BG</td>
</tr>
</tbody>
</table>
Box : GR+/BG  
Messages : W+/B  
Information : B/W  
Fields : N/BG

Database/View: accts  
Index Order: jon

Layout Report for Horizontal Bar Menu: RMSMAIN

Setup for RMSMAIN follows:

Colors for Menu/Picklist:

Color Settings:
Text : W+/N  
Heading : W+/N  
Highlight : GR+/B  
Box : W+/R  
Messages : W+/N  
Information : B/W  
Fields : N/BG

Before Menu dBASE Code RMSMAIN:

SET PROCEDURE TO ACCTPROC

Bar actions for Menu RMSMAIN follow:

Bar: 1  
Prompt: ACCOUNTS
Action: Open a Popup Menu Named: ACCTMENU

Bar: 2  
Prompt: PERSONNEL  
Action: Open a Popup Menu Named: PERSMENU

Bar: 3  
Prompt: SUPPLY  
Action: Open a Popup Menu Named: SUPPMENU
Bar: 4
Prompt: LABOR
Action: Open a Popup Menu Named: LABMENU

Bar: 5
Prompt: TRAVEL
Action: Open a Popup Menu Named: TRAVMENU

Bar: 6
Prompt: TOOLS
Action: Open a Popup Menu Named: TOOLMENU

Bar: 7
Prompt: EXIT
Action: Open a Popup Menu Named: EXITMENU

Layout Report for Popup Menu: ACCTMENU

Setup for ACCTMENU follows:

Use database/view and index file(s) in effect at run time.

Colors for Menu/Picklist:

Color Settings:
- Text: W+/N
- Heading: W+/N
- Highlight: GR+/B
- Box: W+/R
- Messages: W+/N
- Information: B/W
- Fields: N/BG

Before Menu dBASE Code ACCTMENU:

- SET STATUS OFF
- SET SCOREBOARD OFF
Bar actions for Menu ACCTMENU follow:

Bar: 1
Prompt: ADD NEW ACCOUNTS
Action: APPEND
Format File: accts.fmt

Before dBASE Code for this item:
*---Open Accounts database
USE personne ORDER pi
USE accts

After dBASE Code for this item:
*---Close Accounts database
CLOSE DATABASES

Bar: 2
Prompt: VIEW/EDIT ACCOUNTS
Action: Run dBASE Program: DO SRCHJON

Before dBASE Code for this item:
*---Open Accounts database
USE personne ORDER pi
GO TOP

After dBASE Code for this item:
*---Close Accounts database
CLOSE DATABASES

Bar: 3
Prompt: REMOVE MARKED ACCOUNTS
Action: Pack Current File
Window WINDOW1 FROM 10,10 TO 20,60 Double

Before dBASE Code for this item:
*---Pack Accounts database
USE accts

After dBASE Code for this item:
*---Save changes and close database
CLOSE DATABASES
Bar: 4
Prompt: PRINT EXPENSE SUMMARY
Action: Run dBASE Program: DO EXPSUM

After dBASE Code for this item:
-----------------------------------
SET PRINT OFF
CLOSE DATABASES
-----------------------------------

Bar: 5
Prompt: PRINT OTHER LABOR REPORT
Action: Run Report Form OTHERPAY.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: OTHERPAY.QBE

After dBASE Code for this item:
-----------------------------------
SET CONSOLE ON
-----------------------------------

Bar: 6
Prompt: ---------------------------
Action: Text only defined for this option - NO ACTION
---------------------------

Bar: 7
Prompt: DIRECT FUND ALLOCATION
Action: Open a Popup Menu Named: DACCMENU

Layout Report for Popup Menu: PERSMENU

Setup for PERSMENU follows:
-----------------------------------
Use database/view and index file(s) in effect at run time.
Colors for Menu/Picklist:
------------------------
Color Settings:
Text : W+/N
Heading : W+/N
Highlight : GR+/B
Box : W+/R
Messages : W+/N
Information : B/W
Fields : N/BG

Bar actions for Menu PERSMENU follow:
--------------------------------------
Bar: 1
Prompt: ADD NEW PERSONNEL
Action: APPEND
Format File: personne.fmt

Before dBASE Code for this item:
------------------------------
*---Open Personnel database
USE personne

After dBASE Code for this item:
------------------------------
*---Close Personnel database
CLOSE DATABASES

Bar: 2
Prompt: VIEW/EDIT PERSONNEL
Action: Run dBASE Program: DO SRCHPER

After dBASE Code for this item:
------------------------------
*---Close Personnel database
CLOSE DATABASES

Bar: 3
Prompt: REMOVE MARKED PERSONNEL
Action: Pack Current File
Window WINDOW3 FROM 10,10 TO 20,60 Double

Before dBASE Code for this item:
------------------------------
*---Open Personnel database
USE personne
After dBASE Code for this item:
-----------------------------------
*---Save changes and close database
CLOSE DATABASES

Bar: 4
Prompt: PRINT PERSONNEL REPORT
Action: Run Report Form PERSON.frm
Command Options:
  PLAIN
  NOEJECT
Print Mode: Send to Default Printer
New Database/View: PERSON.QBE

After dBASE Code for this item:
-----------------------------------

Bar: 5
Prompt: PRINT APPT STATUS REPORT
Action: Run Report Form APPSTATU.frm
Command Options:
  PLAIN
  NOEJECT
Print Mode: Send to Default Printer
New Database/View: APPSTATU.QBE

After dBASE Code for this item:
-----------------------------------

Bar: 6
Prompt: PRINT 30 DAY APPT STATUS REPORT
Action: Run Report Form APPST30.frm
Command Options:
  PLAIN
  NOEJECT
Print Mode: Send to Default Printer
New Database/View: APPSTATU.QBE

After dBASE Code for this item:
-----------------------------------
Layout Report for Popup Menu: SUPPMENU

Setup for SUPPMENU follows:

Use database/view and index file(s) in effect at run time.

Colors for Menu/Picklist:

Color Settings:

- Text: W+/N
- Heading: W+/N
- Highlight: GR+/B
- Box: W+/R
- Messages: W+/N
- Information: B/W
- Fields: N/BG

Bar actions for Menu SUPPMENU follow:

Bar: 1
Prompt: ADD NEW TRANSACTIONS
Action: APPEND
Format File: supply.fmt

Before dBASE Code for this item:

*---Open Supply database
SELECT A
USE personne ORDER pi
SELECT B
USE accts ORDER jon
SELECT C
USE supply
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:

*---Close Supply database
CLOSE DATABASES
Bar: 2
Prompt: VIEW/EDIT TRANSACTIONS
Action: Run dBASE Program: DO SRCHDNR

Before dBASE Code for this item:
-----------------------------------------------
*---Open Supply database
SELECT A
USE personne ORDER pi
GO TOP
SELECT B
USE accts ORDER jon
GO TOP
SELECT C
USE supply ORDER docnr
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:
-----------------------------------------------
*---Close Supply database
CLOSE DATABASES

Bar: 3
Prompt: REMOVE MARKED TRANSACTIONS
Action: Pack Current File
Window WINDOW4 FROM 10,10 TO 20,60 Double

Before dBASE Code for this item:
-----------------------------------------------
*---Open Supply database
USE supply

After dBASE Code for this item:
-----------------------------------------------
*---Save changes and close database
CLOSE DATABASES

Bar: 4
Prompt: PRINT AGING REPORT
Action: Run Report Form AGING.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: AGING.QBE

After dBASE Code for this item:
-----------------------------------------------
SET CONSOLE ON
Bar: 5
Prompt: PRINT OUTSTANDING REQN REPORT  
Action: Run Report Form SUPSTAT.frm  
Command Options:  
PLAIN  
NOEJECT  
Print Mode: Send to Default Printer  
New Database/View: SUPSTAT.QBE  
After dBASE Code for this item:  
-----------------------------------  
SET CONSOLE ON  
-----------------------------------

Bar: 6
Prompt: PRINT SUPPLY OBLIGATION REPORT  
Action: Run Report Form SUPCHG.frm  
Command Options:  
PLAIN  
NOEJECT  
Print Mode: Send to Default Printer  
New Database/View: SUPCHG.QBE  
After dBASE Code for this item:  
-----------------------------------  
SET CONSOLE ON  
-----------------------------------

Bar: 7
Prompt: PRINT REQN STATUS REPORT  
Action: Run Report Form SUPRQNST.frm  
Command Options:  
PLAIN  
NOEJECT  
Print Mode: Send to Default Printer  
New Database/View: SUPRQNST.QBE  
After dBASE Code for this item:  
-----------------------------------  
SET CONSOLE ON  
-----------------------------------

Layout Report for Popup Menu: LABMENU  
-----------------------------------

Setup for LABMENU follows:  
-----------------------------------

Use database/view and index file(s) in effect at run time.
Colors for Menu/Picklist:

Color Settings:

Text : W+/N
Heading : W+/N
Highlight : GR+/B
Bo* : W+/R
Messages : W+/N
Information : B/W
Fields : N/BG

Bar actions for Menu LABMENU follow:

Bar: 1

Prompt: ADD PAYROLL RECORDS
Action: APPEND
Format File: labor.fmt

Before dBASE Code for this item:

*----Open Temporary Labor database
SELECT A
USE personne ORDER idcode
SELECT B
USE accts ORDER jon
SELECT C
USE templab
SET RELATION TO idcode INTO personne, jon INTO accts

After dBASE Code for this item:

*----Update Labor database and save
CLOSE DATABASES
*----Calls Payroll procedure from Acctproc file
DO PAYROLL

Bar: 2

Prompt: VIEW/EDIT/DELETE PAYROLL RECORDS
Action: Run dBASE Program: DO SRCHLAB
Use database/view and index file(s) in effect at run time.

Before dBASE Code for this item:

*----Open databases
DO PAYEDIT
After dBASE Code for this item:
*---Close Labor database
CLOSE DATABASES
*---Call Payroll procedure from Acctproc file
DO PAYROLL

---

Bar: 3
Prompt: PRINT LABOR EXPENSE REPORT
Action: Run Report Form LABCHGS.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: LABCHGS.QBE

After dBASE Code for this item:
-----------------------------
SET CONSOLE ON

---

Bar: 4
Prompt: PRINT PAYRECORD REPORT
Action: Run Report Form INDPAY.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: PAYREC.QBE

After dBASE Code for this item:
-----------------------------
SET CONSOLE ON

---

Layout Report for Popup Menu: TRAVMENU

Setup for TRAVMENU follows:
-----------------------------

Use database/view and index file(s) in effect at run time.
Colors for Menu/Picklist:

---------------------------------------

Color Settings:

Text : W+/N
Heading : W+/N
Highlight : GR+/B
Box : W+/R
Messages : W+/N
Information : B/W
Fields : N/BG

Bar actions for Menu TRAVMENU follow:
---------------------------------------

Bar: 1
Prompt: ADD NEW ORDER
Action: APPEND
Format File: travel.fmt

Before dBASE Code for this item:
---------------------------------------

*---Open Travel database
SELECT A
USE personne ORDER pi
GO TOP
SELECT B
USE accts ORDER jon
GO TOP
SELECT C
USE travel
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:
---------------------------------------

*---Close Travel database
CLOSE DATABASES

---------------------------------------

Bar: 2
Prompt: VIEW/EDIT ORDERS
Action: Run dBASE Program: DO SRCHTRAV

Before dBASE Code for this item:
---------------------------------------

*---Open Travel databases
SELECT A
USE personne ORDER pi
GO TOP
SELECT B
USE accts ORDER jon
GO TOP
SELECT C
USE travel ORDER lastname
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:
----------------------------------------------
*----Close Travel database
CLOSE DATABASES
----------------------------------------------

Bar: 3
Prompt: REMOVE MARKED ORDERS
Action: Pack Current File
Window WINDOW6 FROM 10,10 TO 20,60 Double

Before dBASE Code for this item:
----------------------------------------------
*----Open Travel database
USE travel

After dBASE Code for this item:
----------------------------------------------
*----Close and save changes
CLOSE DATABASES
----------------------------------------------

Bar: 4
Prompt: PRINT TRAVEL OBLIGATION REPORT
Action: Run Report Form TRAVSTAT.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: TRAVSTAT.QBE

After dBASE Code for this item:
----------------------------------------------
SET CONSOLE ON
----------------------------------------------

Bar: 5
Prompt: PRINT TRAVEL TICKLER REPORT
Action: Run Report Form TRAVPKUP.frm
Command Options:
PLAIN
NOEJECT
Print Mode: Send to Default Printer
New Database/View: TRAVPKUP.QBE
After dBASE Code for this item:

SET CONSOLE ON

---

Bar: 6
Prompt: PRINT DELINQUENT TRAVEL CLAIM
Action: Run Report Form DELQTRAV.frm
Command Options:
    PLAIN
    NOEJECT
Print Mode: Send to Default Printer
New Database/View: DELQTRAV.QBE

After dBASE Code for this item:

SET CONSOLE OFF

---

Bar: 7
Prompt: REPORT
Action: Text only defined for this option - NO ACTION

---

Bar: 8
Prompt: PRINT FLAG APPROVAL STATUS
Action: Run Report Form FLAGAPP.frm
Command Options:
    PLAIN
    NOEJECT
Print Mode: Send to Default Printer
New Database/View: FLAGAPP.QBE

After dBASE Code for this item:

SET CONSOLE ON

---

Bar: 9
Prompt: REPORT
Action: Text only defined for this option - NO ACTION

---

Layout Report for Popup Menu: TOOLMENU

Setup for TOOLMENU follows:

Use database/view and index file(s) in effect at run time.
Colors for Menu/Picklist:

_____________________
Color Settings:
    Text        : W+/N
    Heading     : W+/N
    Highlight   : GR+/B
    Box         : W+/R
    Messages    : W+/N
    Information : B/W
    Fields      : N/BG

Bar actions for Menu TOOLMENU follow:
__________________________
Bar: 1
    Prompt: BACKUP DATA TO DRIVE A
    Action: Run Dos Program - COPY *.DBF A:

Bar: 2
    Prompt: IMPORT SUPPLY DATA
    Action: Run dBASE Program: DO SUPIMP
            Window IMP FROM 10,5 TO 20,75 Single

Bar: 3
    Prompt: REBUILD CORRUPTED INDEXES
    Action: Run dBASE Program: DO RECRENDX
            Window RENDX FROM 10,5 TO 20,75 Double

Bar: 4
    Prompt: RESTORE DATABASES FROM
    Action: Run dBASE Program: DO RESTORE
            Window REST FROM 10,5 TO 20,75 Single

Bar: 5
    Prompt: DRIVE A
    Action: Text only defined for this option - NO ACTION

Bar: 6
    Prompt: EXPORT ACCT/PERS DATA
    Action: Run Dos Program - COPY *.DBF B:
            Window EXP FROM 10,5 TO 20,75 Single
Layout Report for Popup Menu: EXITMENU

Setup for EXITMENU follows:

Use database/view and index file(s) in effect at run time.

Colors for Menu/Picklist:

Color Settings:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>W+/N</td>
</tr>
<tr>
<td>Heading</td>
<td>W+/N</td>
</tr>
<tr>
<td>Highlight</td>
<td>GR+/B</td>
</tr>
<tr>
<td>Box</td>
<td>W+/R</td>
</tr>
<tr>
<td>Messages</td>
<td>W+/N</td>
</tr>
<tr>
<td>Information</td>
<td>B/W</td>
</tr>
<tr>
<td>Fields</td>
<td>N/BG</td>
</tr>
</tbody>
</table>

Bar actions for Menu EXITMENU follow:

Bar: 1
Prompt: QUIT TO DOS
Action: Quit to DOS:

Layout Report for Popup Menu: DACCMENU

Setup for DACCMENU follows:

Colors for Menu/Picklist:

Color Settings:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>W+/N</td>
</tr>
<tr>
<td>Heading</td>
<td>W+/N</td>
</tr>
<tr>
<td>Highlight</td>
<td>GR+/B</td>
</tr>
<tr>
<td>Box</td>
<td>W+/R</td>
</tr>
<tr>
<td>Messages</td>
<td>W+/N</td>
</tr>
<tr>
<td>Information</td>
<td>B/W</td>
</tr>
<tr>
<td>Fields</td>
<td>N/BG</td>
</tr>
</tbody>
</table>

Bar actions for Menu DACCMENU follow:

Bar: 1
Prompt: ADD NEW ALLOCATIONS
Action: APPEND
Format File: daccts.fmt
Before dBASE Code for this item:

*---Open Direct Account database
SELECT A
USE personne ORDER pi
SELECT B
USE accts ORDER jon
SELECT C
USE daccts
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:

*---Close Direct Account database
CLOSE DATABASES

Bar: 2
Prompt: VIEW/EDIT ALLOCATIONS
Action: Run dBASE Program: DO SRCHALOT

Before dBASE Code for this item:

*---Open Direct Account database
CLOSE DATABASES
SELECT A
USE personne ORDER pi
GO TOP
SELECT B
USE accts ORDER jon
GO TOP
SELECT C
USE daccts ORDER name
SET RELATION TO pi INTO personne, jon INTO accts

After dBASE Code for this item:

*---Close Direct Account database
CLOSE DATABASES

Bar: 3
Prompt: REMOVE MARKED ALLOCATIONS
Action: Pack Current File
Window WINDOW2 FROM 10,10 TO 20,60 Double

Before dBASE Code for this item:

*---Open Direct Account database
USE daccts
After dBASE Code for this item:

*---Save changes and close database
CLOSE DATABASES

Bar: 4
Prompt: PRINT NA SUMMARY
Action: Run dBASE Program: DO DIRECT

After dBASE Code for this item:

*---
SET PRINT OFF
CLOSE DATABASES

End of Application Documentation
APPENDIX D: USER'S GUIDE
TO THE ADMINISTRATIVE SCIENCE DEPARTMENT'S
FINANCIAL MANAGEMENT INFORMATION SYSTEM

A. INTRODUCTION

The purpose of this user's manual is to familiarize the user with the AS Department's Financial Management Information System (FMIS), as well as to provide some background information which may be used by more experienced personal computer users. Access to the various databases and reports is gained by selecting the desired action from pull-down menus. The system is very easy to use but does require some knowledge with regard to specific requirements at various data entry points. Wherever possible, these requirements are treated by the program as "multiple choice" entries in formatted data entry screens; that is, only allowed choices are available to the user. Also, the user should note that the built-in validation procedures also put limitations on data entry. These topics are addressed in this manual where appropriate. Finally, sample program "screens" are presented to aid in familiarization.

B. GETTING STARTED

Before running FMIS, the dBASE IV system program must be installed on a hard disk as specified in the dBASE IV
installation guide. Next, the FMIS system application files must be loaded onto the hard disk and stored in a sub-subdirectory of the dBASE IV directory. The hard disk storage space requirements are as follows: (1) dBASE program - 3.5 to 4.5 MB, and (2) FMIS - 1.5 MB. Therefore, a minimum of 10 Megabytes of available hard disk storage space is required. If the system is used by other departments with large databases, more hard disk storage space is recommended.

To run the application, simply type "FMIS" at the DOS prompt. The FMIS.BAT file will automatically access the appropriate subdirectories, start dBASE IV and execute the application. The "Welcome Screen" (Figure D.1) will then

![Figure D.1](image)

Welcome to the Administrative Science Dept's
Financial Management Information System

Press 4-J to continue.
appear. A path command in the DOS autoexec.bat file should include the subdirectory in which the dBASE IV program files are stored. Figure D.2 provides an example startup batch file.

Another way to start the application is from the dBASE Control Center. This is done by highlighting the application name "RMS" under the Application column, pressing enter and responding appropriately to the on screen prompts.

Finally, the system can also be started from the dBASE "dot prompt". This is done by entering "DO RMS" and pressing the Enter key.

C. MENUS

Each functional area or module, i.e., Accounts, Personnel, Supply, etc., is displayed across the top of the screen. Through each menu, all operations of the system, from data

```
FMIS.BAT

ECHO OFF
CD C:\DBASE
CD C:\DBASE\FMIS
DBASE RMS

NOTE: THE DBASE IV PROGRAM FILES ARE STORED IN THE "DBASE" SUBDIRECTORY. THE FMIS SYSTEM FILES ARE STORED IN THE "FMIS" SUBDIRECTORY. THE "DBASE RMS" COMMAND EXECUTES BOTH THE DBASE PROGRAM AND THE APPLICATION.
```

Figure D.2  Startup Batch File
entry to report preparation, can be performed. As can be seen on the screen (Figure D.3), the menu for accessing the Accounts module is displayed. Other modules may be accessed by pressing the left or right arrow keys. As these keys are pressed, each newly selected module’s menu will pop-up. When the desired menu is displayed, item selection may be made by pressing the up or down arrow keys. When the desired selection is highlighted, pressing the Enter key will initiate the action. Table D.1 is a summary of the keys used to navigate the pull-down menus.

Table D.1 Keys Used to Navigate Pull-Down Menus.

<table>
<thead>
<tr>
<th>Key</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left arrow</td>
<td>Moves to the module option to the left.</td>
</tr>
<tr>
<td>Right arrow</td>
<td>Moves to the module option to the right.</td>
</tr>
<tr>
<td>Down arrow</td>
<td>Moves down to the next available menu option on the current pull-down menu.</td>
</tr>
<tr>
<td>Up arrow</td>
<td>Moves up to the next available menu option on the current pull-down menu.</td>
</tr>
<tr>
<td>PgDn</td>
<td>Moves to the last available option on the current pull-down menu.</td>
</tr>
<tr>
<td>End</td>
<td>Same as PgDN.</td>
</tr>
<tr>
<td>PgUp</td>
<td>Moves to the first available option on the current pull-down menu.</td>
</tr>
<tr>
<td>Home</td>
<td>Same as PgUp.</td>
</tr>
<tr>
<td>Enter</td>
<td>Selects the currently highlighted option.</td>
</tr>
<tr>
<td>First letter</td>
<td>Selects the currently highlighted option.</td>
</tr>
<tr>
<td>Esc</td>
<td>Exits the system or backs up to the Accounts menu when in the Direct Accounts menu.</td>
</tr>
</tbody>
</table>
As mentioned above, access to each module is accomplished through its menu. Specific operations include adding,

![Accounts Menu](image)

**Figure D.3 Accounts Menu**

editing, and deleting data, report generation and general database maintenance. Data entry and editing commands are the same for each module. For example, making an "Add Data" selection in any module will call up a formatted data entry screen for inputing data. The data entry process, that is, keying in the data, is performed in the same manner for each.

It is important to note that two methods are employed to help alleviate input errors during data entry and editing. The first is a validation process. This occurs when entering
data into the Job Order Number (JON) and Principal Investigator (PI) fields of the data entry formats of the Direct Account, Supply, Travel and Labor modules. The system will not allow the user to create a new record unless a valid JON and PI code is entered in the appropriate field. The second method is that of creating "multiple choice" entry fields. This allows the input of only predetermined entries. A record cannot be established with other than one of those choices. The following sections will discuss each module and point out specific data entry requirements or processes.

1. The Accounts Module

The Accounts module is used for management of the various fund accounts. Within this module accounts and their related authorized funding levels are established and maintained. Also, summary obligation reports are generated from this menu. Lastly, this menu also allows entry to the Direct Accounts Module, a subsidiary of the Accounts Module. Figure D.3 depicts the system with the Accounts menu active.

a. Add New Accounts

To add new accounts, select "Add New Accounts" from the Accounts menu above. Making this selection will display a formatted data entry screen (see Figure D.4). Keyboard commands for data entry are listed in Table D.2.

It is important to note two aspects of entering data in this screen. First, the second data entry field asks
### Navigation and Editing keys for Data Entry/Editing Screens

- Enter key in the first data field without entering data will result in the return to the Account menu without saving the record.
- Enter key after data is entered completes entry and moves to next field.
- Down arrow key moves cursor down.
- Up arrow key moves cursor up.
- Right arrow key moves cursor right one character.
- Left arrow key moves cursor left one character.
- PgDn key moves down one record on edit screen or one screenful where annotated in screen format.
- PgUp key moves up on record on edit screen or one screenful where annotated in screen format.
- Del key deletes character over cursor.
- Backspace key moves left one character, erasing along the way.
- Ctrl-End keys pressed simultaneously saves the new record or editing changes.
- Ctrl-Y keys pressed simultaneously deletes all characters to right of cursor.
- Ctrl-U keys pressed simultaneously marks record for deletion.
- Ins key switches between Insert and Overwrite modes.
- Home key moves to first character in current field.
- End key moves to last character in current field.
- Tab key moves to next field.
- Shift-Tab keys pressed simultaneously moves to previous field.
- Esc key leaves current record without saving record or changes if editing.

for account type. Enter "R" for Reimbursable, "O" for Direct/Indirect, and "X" for labor accounts not directly managed but charged by the department. Signifying "X" will segregate all transactions citing the account's JON from summary reporting.
Figure D.4 Add New Accounts Screen

Total obligations for these labor charges are reported in the "Other Labor Report", a menu option in this module. Second, the JON entry field is used as a validation check by the Direct Account, Supply, Labor and Travel modules. An incorrect JON may prohibit transaction entry in one of those modules.

b. View/Edit Accounts

To view or edit account information, select "View/Edit Accounts". Once this selection is made, the search screen displayed in Figure D.5 will appear. This screen allows you to specify the JON you wish to view/edit, and will cause the system to go directly to that record and display its
Enter Job Order No. to look for: □□□

Figure D.5 Search Screen for Viewing/Editing Account Data File

data. If the JON entered is not found during the search, a message will appear and ask you to try again. Pressing Enter with no entry will result in the display of the first sorted JON.

Each record will be displayed by a view/edit screen as shown in Figure D.6. New records may not be added from this or any edit screen in the system. Commands listed in Table D.2 are also used for editing records.

Deleting a record is a two-step operation. First, the Ctrl-U command is used in the editing screen to mark the record(s) for deletion. Once the record has been marked, other records can then be viewed or edited. The Ctrl-End key
Figure D.6 Account View/Edit Screen

is then used when all editing is complete. This saves the deletion mark with the record. The record can be unmarked by returning to the edit screen, selecting the record and using the Ctrl-U command. The second step in the deletion process is described below.

**c. Remove Marked Records**

The final step in deleting a record(s) is by selecting the "Remove Marked Records" option from the Accounts menu. This will permanently remove each marked record from the database and automatically perform any necessary data file maintenance.
d. Reports

Two report options are available in the Account module. Either report may be generated by highlighting the desired report and pressing the Enter key.

s. Direct Fund Allocation

The purpose of the Direct Accounts module is to set up fund authorization to Direct-Funded research project accounts for summary financial reporting purposes. Selecting the "Direct Fund Allocations" option will gain access to the Direct Accounts Module. An additional menu, as displayed in Figure D.7 will appear.

![Direct Accounts Menu](image)

Figure D.7 Direct Accounts Menu
Selecting the "Add New Allocations" option will display the data entry screen shown in Figure D.8. This allows the establishment of new allocations and related funding authorization. Of note, the JON and PI field entries are validated against the Accounts and Personnel databases. This ensures that the account and faculty member have been established in the system thus assuring correct accrual of obligations.

Selecting "View/Edit Allocations" will result in first a search screen requesting the name of the faculty member to whom the allocation was made, and then an edit...
screen. Once again, deleting a record is a two step process. The record(s) to be deleted must be marked. Then the "Remove Marked Allocations" option should be selected to remove the record from the data file.

One report is available from within the Direct Accounts Module. It is generated by selecting the "Print NA Summary" option.

2. The Personnel Module

The Personnel module (Figure D.9) provides the ability to maintain personal information about AS Department personnel (faculty and staff) while storing data used by the other system modules for both relational and validation purposes.
a. Add New Personnel

Creating a new record in this module requires use of two screens for data entry. Figures D.10 and D.11 show the data entry screens to establish a new employee record. When in screen one, access screen two by pressing the PgDn key. The new screen will appear and the cursor will be positioned in the "Remarks" field. To return to screen one, press the PgUp key, or to add additional new records, press the PgDn key.

The "Remarks" field in the second screen makes use of a Memo field. The memo field is a free-format area to enter a block of text. No specific entry format is required. To enter a remark, press the F9 key. Information can be typed directly into the field, however, it will start to "scroll" out of the "Remarks" window as the space is filled. Pressing the F9 key again will open the Memo field so that all data entered may be viewed or edited. Pressing F9 again will close the field. To save the entered information, press the Ctrl and End keys simultaneously. To add additional records press the PgDn key, or to save the new record and return to the Personnel menu, press the Ctrl and End keys a second time.

There are two data entry fields that warrant special attention. The "Principal Investigator" (PI) field is used by the Direct Accounts, Supply, Travel and Labor modules for validation purposes. Also, the "ID Code" field is used by the Labor module to extract information such as name and base
salary when processing payroll transactions.

b. View/Edit Personnel

To view or edit personnel records requires the use of two screens as discussed above. Selecting "View/Edit Personnel" will result in first a search screen requesting the last name of the person whose record is to be viewed, and then an edit screen in which changes may be made.

c. Remove Marked Personnel

Selecting this option will remove from the Personnel data file all records previously marked for deletion.
d. Reports

Three report options are available and may be selected from the Personnel menu by highlighting the desired report option and pressing the Enter key.

3. The Supply Module

The purpose of the Supply module is to record transaction information for each procurement or requisition action carried out by the department. In addition to the financial data collected, it also provides tools for the supply clerk to facilitate requisition processing, follow-up and validation.
a. Add Supply Transactions

Selecting "Add New Transactions" from the menu as displayed in Figure D.12 will allow recording of each supply requirement. The data entry screen, as shown in Figure D.13, will validate the JON and PI field entries against the Account and Personnel data file to ensure that the account and the principal investigator have been established in the system.

In the "Category" field, there are two categories that are currently tracked. They are honorariums and contract support. To enable summary reporting of honorariums and contract support, enter "HO" or "CS" respectively. Other category codes as required may be entered in this field.
b. View/Edit Transactions

Selecting "View/Edit Transactions" from the Supply module menu will allow viewing or editing of individual transactions. A search screen allows a search by document number. Editing may then be accomplished.

c. Remove Marked Transactions

Selecting this option will remove from the data file those records previously marked for deletion.

d. Reports

Several reports are available in this module and may be selected by highlighting the desired option and pressing the Enter key.
4. The Labor Module

The purpose of the Labor module is to allow for recording of all payroll actions for financial reporting purposes. The module also maintains payroll data for individual employees including pay-to-date, hours worked, leave taken, etc.

a. Add Payroll Records

Selecting the "Add Payroll Records" from the Payroll module menu displayed in Figure D.14 will allow entry of payroll data into the system. Using the input from the "ID Code" field, the data entry screen (Figure D.15) will automatically extract name and base salary information from the Personnel module, make all required calculations, and store the name and total pay information into their respective fields. The name and total pay calculation will not appear on the screen until the user leaves the record.

b. View/Edit/Delete Payroll Records

Selecting "View/Edit/Delete Payroll Records" from the Labor module menu allows viewing, editing or deletion of individual payroll records. Again, a search screen based on last name is provided to allow direct access to specific records. The deletion process is different in this module in that it's not the same two step process as described previously. When records are marked for deletion, they are permanently removed from the data file when the edit screen is
Figure D.14 Labor Module Menu

exited.

c. Reports

There are two reports available in this module and may be selected by highlighting the desired option and pressing the Enter key.

5. The Travel Module

The purpose of the Travel module is to allow the management of travel orders both for financial and processing purposes. This includes tracking/tickler reports for delinquent travel claim tracking, "pickup" dates, and tracking orders requiring "Flag" approval.
Selecting "Add New Travel Orders" from the menu shown in Figure D.16 allows entry of new orders into the system. This data entry format uses two screens for data input as was the case in the Personnel module, and moving between screens is the same. Figures D.17 and D.18 show the two data entry screens used in the Travel module.

There are several fields that have specific data input requirements. The "Type of Travel" field provides a "multiple choice" format for inputting data. The options represent specific travel types that are tracked separately and are selected by pressing the Space bar. The selections
are "Regular", "ITO", "TQM", "PCC", and "EDO". Next, when canceling a travel order, entering a "Y" in the "Canceled (Y/N)" field will automatically exclude it from financial obligation totals without the need to zero out any previously entered dollar amounts. Finally, the "Claim Submitted? (Y/N)" will automatically set to "N" and must be changed to "Y" to clear the travel order from delinquency tracking.

b. View/Edit Travel Orders

Selecting the "View/Edit Travel Orders" option from the Travel module menu will allow viewing or editing of travel orders. A search screen is initially displayed which allows a search by traveler's last name.

Figure D.16 Travel Module Menu
Figure D.17 Add Travel Orders - Screen 1

c. Remove Marked Travel Orders

Selecting this option from the Travel menu will permanently remove all records marked for deletion from the data file.

d. Reports

Several reports are available from the Travel menu and may be selected by highlighting the desired option and pressing the Enter key.

6. Tools Module

The purpose of the Tools module is to provide various file maintenance utilities needed to effectively manage the FMIS. These include database backup and restoration,
Enter/Edit Travel Information (Page 2)

<table>
<thead>
<tr>
<th>Est. Travel Expenses</th>
<th>Actual Travel Expenses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Per Diem: 0.00</td>
<td>Per Diem: 0.00</td>
</tr>
<tr>
<td>Transportation: 0.00</td>
<td>Transportation: 0.00</td>
</tr>
<tr>
<td>Registration: 0.00</td>
<td>Registration: 0.00</td>
</tr>
<tr>
<td>Total Est. Price: $9.00</td>
<td>Total Actual Amount: $0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pickup Dates</th>
<th>Travel Advance Code:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advance:</td>
<td>Advance Date:</td>
</tr>
<tr>
<td>Ticket:</td>
<td>Advance Amount: 0.00</td>
</tr>
<tr>
<td>Registration:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Claim Submitted (Y/N):</th>
<th>Canceled (Y/N):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Claim Date:</td>
<td>Cancellation Date:</td>
</tr>
</tbody>
</table>

Figure D.18 Add Travel Orders - Screen 2

reindexing of data files and exporting/importing database files to and from the personal computer where supply transactions are processed (these are interim procedures while awaiting the network installation of the system). The various Tool options are selected from the menu displayed in Figure D.19.

a. Backup Data to Drive A

Selecting this option backups all database files. It does not, however, backup the memo field contained in the Personnel module. Prior to initiating this option, be sure that a formatted data disk is placed in the A drive.
b. Import Supply Data

Selecting "Import Supply Data" allows the importing of Supply data files from the desk where supply processing is conducted. Place the disk containing the data in the B drive and select the menu option. The old data files will be erased and the new data will be appended to the Supply database file. Reindexing will be accomplished automatically.

c. Rebuild Corrupted Indexes

This selection should be run whenever there is an unexpected loss of power as this might cause the indexes to become corrupted. This will cause the system to fail and display an error message specifying a problem with an index.
d. **Restore Database Files from Drive A**

Selecting this option will restore a previous backup of all system databases. Place the disk containing the files in drive A and make the selection. Any previous data file records will be deleted, the backup files read into the databases, and the database files reindexed.

e. **Export Acct\Pers Data**

This option allows for the exporting of updated account and personnel data file records required by the Supply module for validation of JON and PI data fields during transaction entry. Place a disk in the B drive and select menu item. The account and personnel database files will be copied to the disk.

7. **Exit Menu**

Selecting "Quit to DOS" from the Exit menu (Figure D.20) will leave the system and return you to the DOS prompt.
Figure D.20  Exit Menu
LIST OF REFERENCES

