ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION STUDIES FOR AFRICA

A COMPUTET-ASSISTED FINANCIAL INFORMATION SUPPORT SYSTEM FOR NATIONAL URBAN PLANNING INSTITUTE (NUPI).

A Thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Information Science.

by

ALEMTSEHAYE MAMO ABE

June, 1996

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE OF STUDIES

SCHOOL OF INFORMATION STUDIES FOR AFRICA

A COMPUTER-ASSISTED FINANCIAL INFORMATION SUPPORT SYSTEM FOR NATIONAL URBAN PLANNING INSTIUTE (NUPI).

By

Alemtsehay Mamo

Name and Signature of Members of the Examining Board

Ato Getachew Birru, Chairman, Examining Board

Ato Tesfaye Biru, Advisor

Dr. G.A. Alabi, Internal Examiner

Prof. P.B. Mangla, External Examiner

DEDICATION

I dedicate this work to W/o Worknesh Bekele (my mother), Ato Getachew Regassa (my husband) and Daniel Getachew (my son).

ACKNOWLEDGEMENT

My deep thanks is extended to those who provided me the necessary support that has made this work possible.

I would like to give sincere thanks to Management of NUPI for providing me the scholarship to join SISA. The contributions of Ato Migbaru Yimer, Ato Teshome Bula, Ato Mujerib Solomon, Ato Mekete Sheferaw and other employees of the institute are acknowledged with sincere thanks.

Special thanks is extended to Ato Tesfaye Biru, my thesis advisor, for his continuous assistance and good advices. Also my particular appreciation goes to Dr G.G Chowdhury, W/o Woinshet Abdella, Ato Sisay Fissha, Ato Worku Alemu and other staff of SISA for their help through out the study.

A sincere note of thanks to Ato Anteneh Begashaw for his help and valuable advices. The advices and moral support of my family and friends esp Ato Zewdineh Mamo and W/t Birke W/ Tensai is acknowledged with special thanks.

ABSTRACT

Ability to produce information that has improved qualities such as relevance, timeliness and accuracy is among the main objectives of financial information systems including NUPI'S finance system.

But the system is facing difficulties in producing such outputs. Most of users requirements are not being met by the system . As a result this study is conducted.

The study documents what and how the existing system carries out its activities. Current system problems are identified. Objectives that should be met by a proposed system are set. Various ways of meeting those objectives are investigated. Finally a solution is proposed. Attempt is also made to demonstrate how the solution over comes the major problems of the existing system. The outcome of this study works not only fir NUPI but also for organisations and institutions that are financed by MOF and using single entry Government accounting system.

TABLE OF CONTENTS

PAGE

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND

Governmental and not-for-profit institutions (unlike business or profit making organizations) are generally characterized by (Hey 1989):

- receipts of significant amounts of resources from resource providers who do not expect to receive either repayment or economic benefits proportionate to the resources provided;

- operating purposes that are other than providing goods or services at a profit or profit equivalent; and

- absence of defined ownership interests that can be sold, transferred or redeemed, or that convey entitlement to a share of a residual distribution of resources in the event of liquidation of the organization.

Likewise, while the main concern of financial/accounting systems in business entities are measurement of net income for a specific period, and determining of costs and performing other analysis useful for managerial control and decision making (Davis and Olson 1985), the focus of financial/accounting

systems in governmental organisations is administering/managing proper utilization of budgets.

Financial resources for Governmental institutions are usually provided by the Government, and taxes collected from the general public (income tax, turn over tax, land use tax, cattle tax etc.) make the main sources.

National Urban Planning Institute (NUPI) of Ethiopia is a Government institution, established in 1987 under proclamation No 317/1987 (Negarit Gazeta) to deal with problems related to urban regional planning. So far it is the only institute in the country concerned with urban regional planning.

The main objectives of the institute, as indicated in the proclamation No. 317/1987 are:

- to carry out research/study for preparation of urban plans;
- to prepare master, development and action plans for urban centres based on the findings of the research/study;
- to train manpower in the field of urban planning.

To carry out these objectives, NUPI is currently structured into functional units as shown in appendix A. The Finance Division of NUPI is charged with the responsibility to properly manage the financial resources of the institute.

Among the major functions performed by the division are the following.

- preparation of recurrent and capital budget proposals of the institute for consideration by Ministry of finance (MOF);
- timely collection/payment of cash;
- timely recording of receipt/expenditure transactions;
- keeping financial records, documents and related reports properly;
- timely generation of required financial reports;
- proper control of expenditures; and
- implementation of relevant Governmental financial rules and regulations as appropriate to NUPI.

1.2 STATEMENT OF THE PROBLEM AND JUSTIFICATION

Proper management of any modern institution requires, among other things, the timely collection, recording, processing and reporting of financial activities of the institution to the management for the purpose of making timely and accurate decisions. This is one of the major requirements of the financial information system at NUPI.

As the institute is totally budgeted from central treasury, it is required to prepare and submit a clear budget utilization report including sound explanation for over or under utilization, to the Ministry of Finance on a regular basis. In fact, approval of the institute's budget proposal for a fiscal year is made on the basis of the actual budget utilization of

the institute in the previous fiscal year which is usually judged from such budget utilization report. Currently, considerable delay and inaccuracy is being observed in producing financial reports for both internal (mainly management of NUPI) and external (MOF, General Auditors etc.) bodies.

Another critical problem area in the existing system lies in the sened transaction management. The institute should keep the amount of sened transactions to the minimum possible. Because at the end of every budget year, on June 30, sened transactions are counted by MOF auditors and the institute is required to return that much of money to central treasury from the coming year approved budget. In the coming year, approved budget less total sened amount of last year is given to the institute to carry out its activities. This greatly affects the activities planned to be performed in that year.

The basic procedures of financial systems of government ministry offices that are budgeted from Ministry of Finance (MOF) is the same. They adopt a single-entry government based accounting system. The output of this study could be used to demonstrate how a computer-assisted systems are better and how to build such financial information system.

Moreover, as mentioned in the proposal, the institute is in the process of buying computer resources. By the time the computers are installed, this study could be used as a starting point for automating the financial activities of NUPI. Since the problem of the division is being felt by management, it is also the management plan to utilize the procured computer resources for

this purpose. To this end, the need for detailed study of the existing activities, prior to embarking to the actual automation, and with the line of identifying the components that could benefit from automation is strongly described by management.

1.3 OBJECTIVES OF THE STUDY

1.3.1 General Objective

The general objective of this study is to propose a computer-assisted Financial Information Support System for the National Urban Planning Institute of Ethiopia in order to enhance attainment of institutional goals.

1.3.2 Specific Objectives

The specific objectives of this study are;

- to analyze the existing financial information system with the aim of identifying detailed user information requirements;
- to assess to what extent these requirements are met by, and identify problems in the existing system;
- to propose a better functioning automated system that addresses the requirements and problems identified;
- to develop a prototype system to demonstrate the database management component of the system;

1.4 SCOPE AND LIMITATIONS OF THE STUDY

Developing a complete and operational computer-assisted financial information support system is a complex and difficult task which requires, among others, expertise in the area of finance, information systems, software engineering as well as financial support and adequate time for the design, development and testing. This, however, is beyond the scope of this study. In particular, the study does not aim at, developing an operational automated system, but rather identify major components of the automated system and demonstrate using a prototype for the feasibility and practicability of the components proposed.

1.5 METHODOLOGY

Information Sources: In analyzing and designing the financial information support system, the major data / information sources used are system users working in the various units of the institute. Also, existing forms, documents, finance manuals, auditor reports, reports generated from various units are studied.

Method of Fact Finding: Both primary and secondary fact gathering methods are employed in the study, namely, examination of records, observation and interview.

Records related to financial activities such as forms, documents manuals,job descriptions, budget and other periodical financial reports produced for Ministry of Finance, external and internal

auditor reports, organizational charts etc are examined to get necessary and sufficient information for the analysis and design work.

Observation method is employed for the study in order to get better understanding of the system. Efforts were made to identify every problem and user requirement by conducting intensive discussions with staff members working at different managerial and operational levels. A list of the people interviewed and discussion/interview guide used during the fact finding activity are attached.

Method of Data Analysis: The structured approach to systems development using structured tools and techniques are used for this purpose, and paying special attention to user participation and use of graphic tools.

Data Dictionaries, Data Flow Diagrams (DFDs) and Entity-Relationship (E-R) modelling are heavily employed in both the analysis of the existing system and design of the proposed system. dBASE IV software is used to develop the prototype system.

1.6 DEFINITION OF TERMS

SYSTEM: A collection of components that work together to realize some objective.

FINANCE: The money resource, income, etc of a nation, organization or person.

FINANCIAL: Pertaining to finance.

REVENUE: Any cash inflow into the finance system of NUPI.

SENED: Payments that are made to various reasons and whose process is not completed. When the process of payment is over, either they are transferred into proper expenditure or the committed amount of money is returned to NUPI's account.

EXPENDITURE: Payments that are made to various reasons and whose process is completed.

1.6 DEFINITION OF TERMS

SYSTEM: A collection of components that work together to realize some objective.

FINANCE: The money resource, income, etc of a nation, organization or person.

FINANCIAL: Pertaining to finance.

REVENUE: Any cash inflow into the finance system of NUPI.

SENED: Payments that are made to various reasons and whose process is not completed. When the process of payment is over, either they are transferred into proper expenditure or the committed amount of money is returned to NUPI's account.

EXPENDITURE: Payments that are made to various reasons and whose process is completed.

1.7 ORGANIZATION OF THE THESIS

The thesis is organised into six Chapters.

The first Chapter, Introduction, provides background information on the study; states the objectives and the scopes of the study; and the methodology employed in conducting the study.

Chapter 2 documents the existing system using Data Flow Diagrams (DFDs), and Entity- Relationship (E-R) Diagram. Problems of the existing system as well as requirements are also presented.

The third Chapter is about the general design of the alternative system proposed. Details of the major objectives of the proposed system, and its components are provided.

The fourth and fifth chapters are about database design and the prototype system. Descriptions of databases designed, programs written, necessary input/output forms and user interface are provided.

The last Chapter presents conclusions drawn from the study and recommendations to further enhance the proposed system.

CHAPTER TWO

ANALYSIS OF THE CURRENT

FINANCIAL SYSTEM

2.1 GENERAL CONTEXT

2.1.1 INFORMATION GATHERING PROCESS

Developing computer-based information systems requires considerable time and effort to be spent in the systems analysis activity which usually involves investigation into the system operation and possible changes to the system (Hawryszievyez 1991).

Developing understanding of a system, which is the main objective that underlie the analysis activity is necessary because it helps:

- to identify correct problems; and
- to propose realistic solutions for the identified problems.

For the purpose of understanding the system, an analyst has to gather enough facts on the existing system through methods such as discussion with system users, examination of documents, observation of system operation, questionnaire.

Aktas (1987) noted that " User participation and consequently user commitment is one of the main contributions of structured

methodologies". Also Atherton (1977) states " the wise system designer recognizes that the user of information must be an active participant in the system whose needs direct the design." Therefore the analyst is required to work hand in hand with users to come up with better solutions.

The fact gathering process took full cognizance of the forgoing. Facts for this study were collected through interviewing users and revising forms, financial rules and regulations, auditor reports etc. Various interviews (planned discussions) were made with selected staff members (both managers and operational personnel) through out the analysis and design process of the finance system. The list of people interviewed and the guide used during the interview are attached in appendix D and E.

In the following sections, attempt is made to document the information obtained as to what the system does and how it does. System documentation graphical tools - DFDs and E-R diagrams are employed. Also, what the users require from a new system is discussed in brief.

2.1.2 MAIN REQUIREMENT OF THE FINANCE SYSTEM

The following is a summary of the major requirements identified during the fact gathering process.

- Generation of various periodical financial reports is required. And the reports should be produced within reasonable period of time.
- Ability to make collection / payment within relatively short period of time is required.

- Ability to retrieve financial information through various access points is expected. Also other information contained in the system should be accessed easily.
- The financial system is expected to hold complete and necessary information about all financial transactions.
- The financial system should be in a position to inform sensitive financial issues to decision makers on time. This could be made through memorandum, statements other than periodical reports.
- The financial system is required to be in a position to implement government financial rules and regulations.
- Well-established project based costing system with a possibility to retrieve financial information with different access points.

2.2 THE EXISTING SYSTEM

As indicated earlier, the major objective of the finance system is to manage financial resources properly. This means specifically to raise finance which is necessary to carry out planned activities and to be able to use it efficiently and effectively. This responsibility is given to the finance division. To meet this, the division is organized into two sections: the Budget Section and the Accounts Section.

The Budget Section is mainly concerned with budget related activities such as : preparation of budget proposals, budget transfer from one code to another when ever required, controlling expenditures not to exceed budgeted amount. The institute has two types of budgets: recurrent and capital. Recurrent budget is used to finance regular activities. Payment for salary of regular employees, for utilities, maintenance of vehicles and office equipments are made from recurrent budget. Capital budget is for financing project activities. Surveying costs, costs for cartography, costs for purchase of equipments and machineries are covered from capital budgets.

> The section is concerned with expenditure budget only. The institute does not generate revenue hence there is no revenue budget. But the term revenue is used through out this paper to represent inflow of finance into the system from both budgetary and non - budgetary sources. Budgetary source (MOF) finances expenditures whereas cash inflow from non - budgetary source (other than MOF) is transferred to central treasury account.

Both capital and recurrent budgets are classified by codes.Revenue codes are assigned to budgets when collected and by the time they are expended, expenditure codes are assigned. For example 4010 is a revenue code and 6101 is expenditure code for salary of regular employees.

The section keeps budget proposals, approved budget documents, budget rules and regulations and any information related to the

institute's budget. All activities of the section are done manually. A separate account is maintained for each budget code to control its expenditure. Every month a budget status report is prepared. Also quarterly, semi - annually and annually reports are generated.

In its interaction with external entities, the Finance Division either pays or receives cash. The Accounts Section of the Finance Division is mainly concerned with this actual finance inflow to and outflow from the division and keeping records of the transactions.

The following sections discuss in detail the existing system in respect of actual revenue, expenditure and sened.

2.2.1 Revenue

Sources of Revenue: There are two sources of finance: budgetary source and non - budgetary source. Central treasury (MOF) is classified as budgetary source. It is the sole financier of the institute. Sources other than MOF fall under the category of non - budgetary source. They include various institutions, organizations & employees. Sell of maps and plans, sell of bid documents, unreturned bid bonds, penalty, etc contribute to such inflow of finance. The institute is not allowed to make use of finance that comes from non - budgetary source. NUPI serves only as intermediary to collect and deposit the money to the central treasury account.

Processing of Revenue: Once recurrent and capital budget proposals are prepared by the Budget Section, the recurrent budget proposal is sent to MOF while the capital budget proposal is given to the Plan and Economy Ministry in required number of copies. By the time of budget hearing, assigned / delegated people from the institute defend the proposals to seek approval. Then the institute is communicated of the approved budgets in two standard forms. For the budgets approved, the accounts section prepares three monthly disbursement request forms;

- One for salary of regular employees of the institute;
- One for recurrent expenditures such as utilities, stationeries, uniform, maintenance ; and
- One for capital expenditures such as Adhoc surveys, cartographies, salary for contract employees .

Information required for preparing these forms is obtained from the following sources.

- Sened file: this file stores all payments in process that are made for various reasons. For example, if payment is made to purchase stationary materials, the payment document remains as sened until the purchasing process is completed. The process is said to be complete only when the purchased items are entered into the store of the institute. This is confirmed when the purchaser brings a goods receiving note (model 19) from store. At this point payment voucher is prepared. Invoice, model 19 and other supporting documents are attached to it. Once this payment voucher is approved and signed, the corresponding sened is cancelled and the payment is

said to be transferred from sened to actual expenditure. All sened payments for which a disbursement request is not made are collected and a request is made to refund the finance to the institute.

- Expenditure file: All expenditures for which a disbursement request is not made are collected from this file and a request is made to refund the expended amount to the institute.

- For certain budget codes, simply one - twelve of the approved annual budget is granted every month. This amount is filled on disbursement request form.

Salaries are requested at the end of every month. The request with necessary supporting document is sent to MOF on the last week of the month. The other two requests are made at the beginning of a month. Rules and regulations provided by MOF in this connection are followed in preparing the forms. They are prepared in seven copies and sent to MOF. Upon approval, the MOF orders the National Bank of Ethiopia (NBE) to transfer only the approved amount of money from central treasury's account to the institute's account. Also one copy of approved disbursement request forms are returned to NUPI. These forms are the basis (source documents) for recording revenue from budgetary source. The forms are stored in disbursement box file. Then they are recorded in revenue receipt (see appendix B). Also NBE sends credit advises to NUPI by the amount of transfer from central treasury to the institute's account. The total credit advice

should equal to the sum of the revenue receipts (or approved disbursement requests). The advises are stored in bank statement box file.

Revenue from non - budgetary sources is collected by the finance division when the Accounts Section gets order in various forms such as cartographic documents sales form, inter - office memo and letter. After the order is verified, a revenue receipt is prepared using the order and other supporting documents as a source document.

Any finance inflow, from budgetary or non - budgetary source, must be recorded in the revenue receipt. The revenue receipt is a standard form prepared and sold by MOF to NUPI and similar organizations. One pad receipt contains fifty leaves, each page in three copies and has consecutive serial numbers. Original receipt always goes to the payer as evidence of payment, the second copy should to the Accounts Section and the third copy remains on the pad. A revenue receipt must be signed. If error is committed in preparing the receipt, all the three copies, stapled together, ruled in double lines and marked "VOID" must remain on the pad.

Even if there are source documents that are used for preparing the receipt, a revenue receipt is the ultimate document required by internal as well as external auditors and others for auditing or other legal purposes. Once the revenue receipt is prepared, it is send to the Cash Office. Cashier collects the money and gives original revenue receipt to payer. The revenue receipt pad having the other two copies are returned to Accounts Section to

be used in recording the transactions on register. All information on revenue receipts are further transferred into registers.

The register used for recording the transactions is also a standard book prepared and sold by MOF. There is one type of register to record all revenues from revenue receipts and all expenditures from payment vouchers. The recording on register is made at the beginning of the following month after bank statement is collected. Expenditure which is not reflected on bank statements is not recorded in the register. It is stored in unrecorded expenditure box file. When recording on register is completed, transactions are further summarized into five standard MOF forms, which are sent to MOF when requested. As indicated earlier, the institute uses revenue from budgetary sources only. Revenue from non - budgetary sources are transferred to MOF account.

2.2.2 Expenditure

Cash outflow from the system are made for reasons such as:

- purchases of goods and services;
- settlement of bills such as telephone, water, electricity;
- contractual agreements such as employee training; and
- salary of employees.

When a payment transaction occurs, order in the form of letter or memorandum is sent to accounts section from the General Manager or Administration and Finance Department Head, depending

on the situation at hand. The first step in the accounts section is to check the validity of the payment order and the attached documents. Some of the issues that are involved in checking are;

- whether the payment order is signed by the right person;
- whether the supporting documents are enough;
- whether the supporting documents are audited;
- whether there is budget allocated for the payment;
- if it is for purchase, whether the supporting documents are signed by relevant authorities; and
- whether it is arithmetically correct.

Then a payment voucher and check (if the payment is made in check) is prepared by the Accounts Section. After they are signed by authorities, all payment documents are sent to cash office. Cashier make payments and hold the documents as sened if the process is not over. If the process is completed, the documents are returned to Accounts Section. Original payment voucher must be attached to the supporting documents. They are stored in expenditures box file and used in recording expenditures in a register.

As stated in section 2.2.1, expenditures that are not reflected on bank statement are stored in separate unrecorded expenditure files until one of the coming bank statements confirm their withdrawal from bank. Once recording is over, a summary report in five standard forms (as with the revenue transaction) is prepared to MOF. Both revenues and expenditures are summarized

in separate columns of the forms. The summary report is sent to MOF when requested.

2.2.3 Sened

In section 1.7, sened is defined as expenditure in process. When a sened transaction occurs, a signed payment document is sent to cashier from the Accounts Section. After payment is made, the cashier holds original payment vouchers of sened and returns copy payment vouchers with necessary supporting documents to Account Section. Since payment is made either in cash or check, there are two sened files - check sened file and cash sened file. They are kept with cashier in a safe. According to the existing financial rules and regulations, the seneds are considered as actual cash until they are converted into expenditure. This is the reason why they are kept in a safe. In addition, any sened transaction must be transferred into expenditure or cash within seven days. It is strictly forbidden to keep sened pending beyond the period. They should be cleared as quickly as possible.

During analysis, it is observed that even though one copy of sened is sent to the Accounts Section, proper sened file is not maintained by the section. There is no register for recording sened transactions. If any information related to sened is required,the accounts section should look for such information from the cashier.

A payment voucher must be prepared for all cash outflow (sened or expenditure). This standard voucher is purchased from MOF. It is prepared in two copies: original for expenditure/sened

file and copy for cashier or Accounts Section. The payment voucher is valid only if it is signed by proper authorities and all information on it must be clearly visible.

2.2.4 System modelling tools

In the previous section, attempt is made to describe what the existing finance system does and how it works. This section presents Data Flow Diagrams (DFDs), Entity - Relationship (E-R) diagrams and Forms Summary Sheets/Tables that parallel/document the existing system described in preceding section. The following conventions are considered in using the tools.

DFDs make use of symbols which represent the above listed system components. Different symbols are employed by various authors. System component symbols that are used in this section are the following.

• Process/ bubble.	Processes are represented by circles. They show what systems do. Processes have data inputs and produce data outputs. A process is identified by a name and number that appears inside a circle that represents the process.
• Data flow	Arrows represent data flows between system components.A data that flows is identified by the label of the arrow.

• External entity Source/Sink	External entities are represented by a square or a rectangle. Such entities either input data into the system or use system output. External entities that supply input data into the system are sometimes termed as sources where as those entities that make use of system output are called sinks. A specific entity is identified by name that is written inside the rectangle.
• Data store	Data stores are represented by an open - ended rectangle. They are repository of data. Data stores have a unique name that is written inside the rectangle.

Figures 2.1, 2.2, 2.3, 2.4, 2.5 & 2.6 show context, top - levelled and levelled DFDs of NUPI finance system. Note that two DFD naming conventions are used in the diagrams. Data flows that store/ read the whole data store record between processes and data store are not labelled. For example in Fig 2.2, 'Process expenditure' process stores the whole of 'Expenditure' record into data store EXPENDITURE hence the data flow is not labelled.

When the data flow is bidirectional between processes and data stores, the flow is labelled with two arrows. For example in fig 2.2 'Prepare disbursement request' process reads the whole of disbursement file and then stores approved disbursement request in the file. Therefore the data flow is represented by unlabelled line with two arrows.

Entity-Relationship Diagram is used to model current system data. At this stage attempt is made to develop the conceptual model of data. This model represents the major data objects and the relationship between them. It describes the essential semantics of system data. Conceptual modelling is discussed using symbols from a modelling method known as entity - relationship analysis. Fig 2.7 shows an Entity - Relationship diagram for NUPI's finance system. Note that the numbers, dot (.) and asterisk (*) in the diagram are used to describe the properties of relationships namely the degree of relationship and the membership class.

The following concepts and symbols are used in preparing the E-R diagrams.

- Entities: are distinct things in the enterprise which are of interest to the analyst or things for which data is maintained. Example :- projects, employees, budgets, cars. A rectangle is used to represent an entity.
- Relationships: are meaningful interactions between the objects. A diamond is used to represent the relationship.
- Attributes: are properties of the entities and relationships.

The various forms and reports that are used and generated within the existing system (together with their contents) summarized in Fig 2.8 and 2.9.

Table 2.10 shows the volume of transaction within the existing system in terms of number and type of source documents received and processed.

The profile of the workforce involved in the financial information processing as well as the newly purchased computer facility for the institute in general are presented in Fig 2.11 and 2.12 respectively.

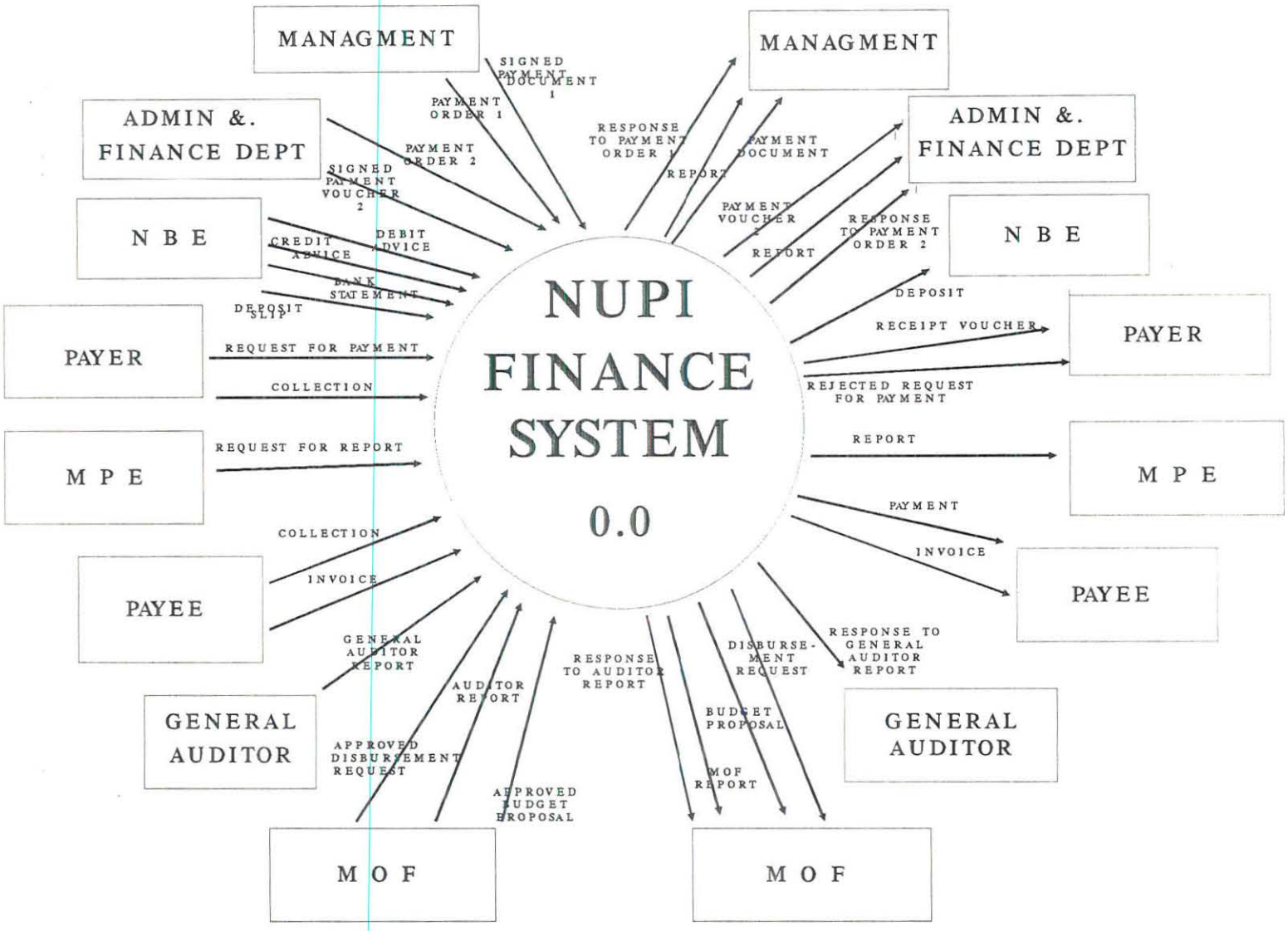


FIG 2.1 CONTEXT DFD OF NUPI FINANCE SYSTEM

NOTE: Two entities with same name are considered as one entity.

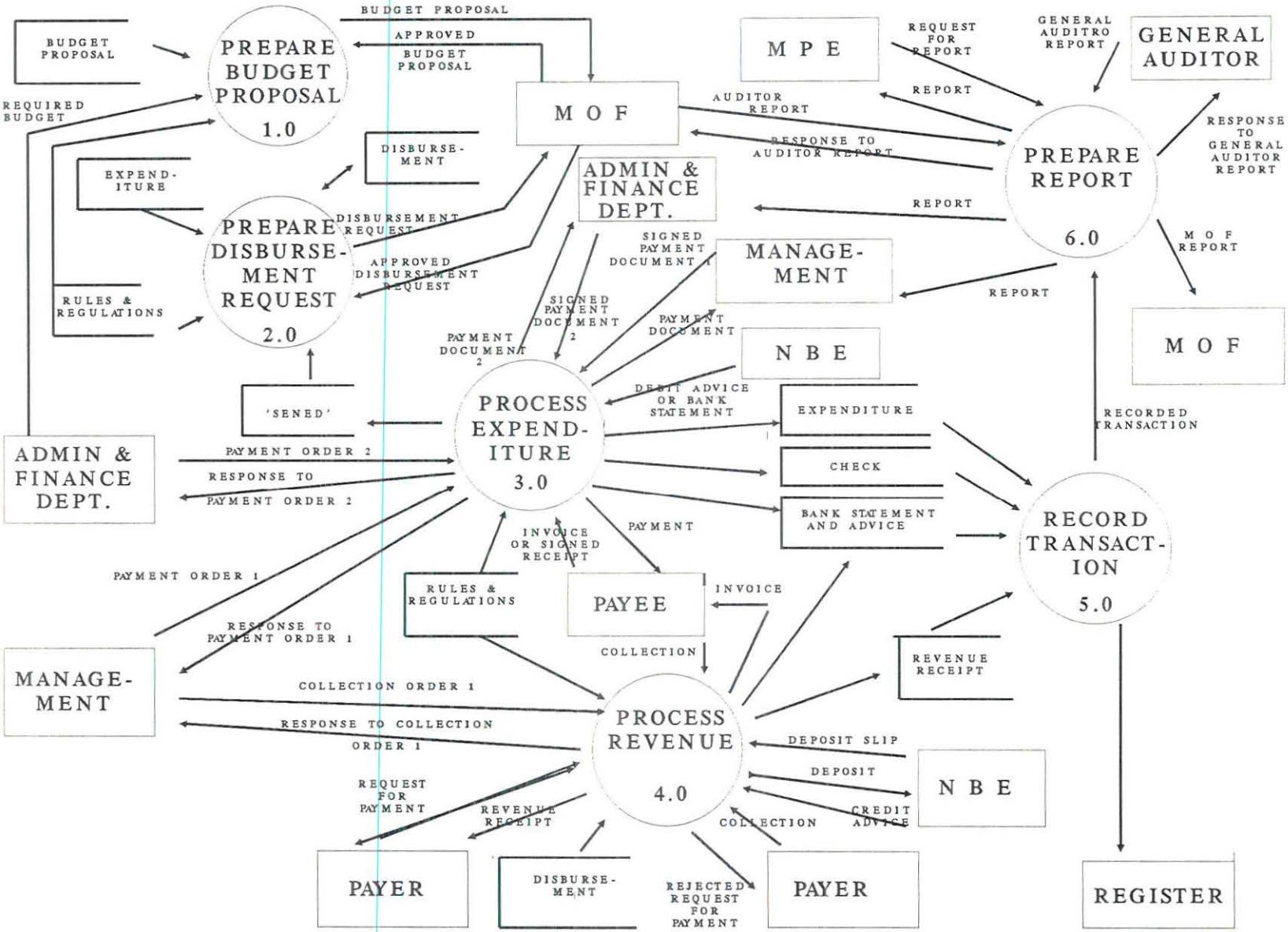


FIG 2.2 A TOP- LEVEL DFD OF THE CURRENT NUPI FINANCE SYSTEM.

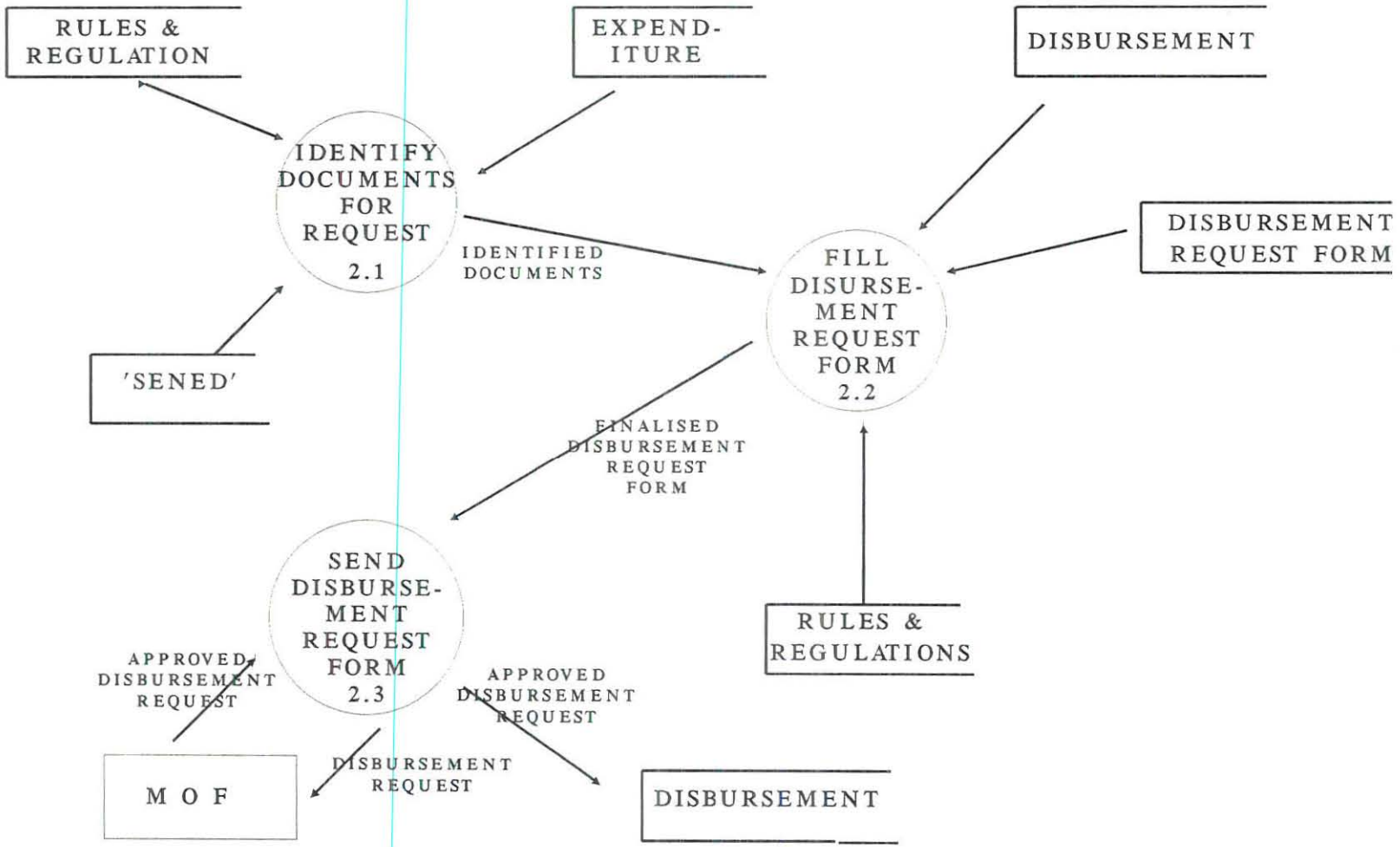


FIG 2.3 DIAGRAM 2 - EXPLOSION OF 'PREPARE DISBURSEMENT REQUEST' PROCESS

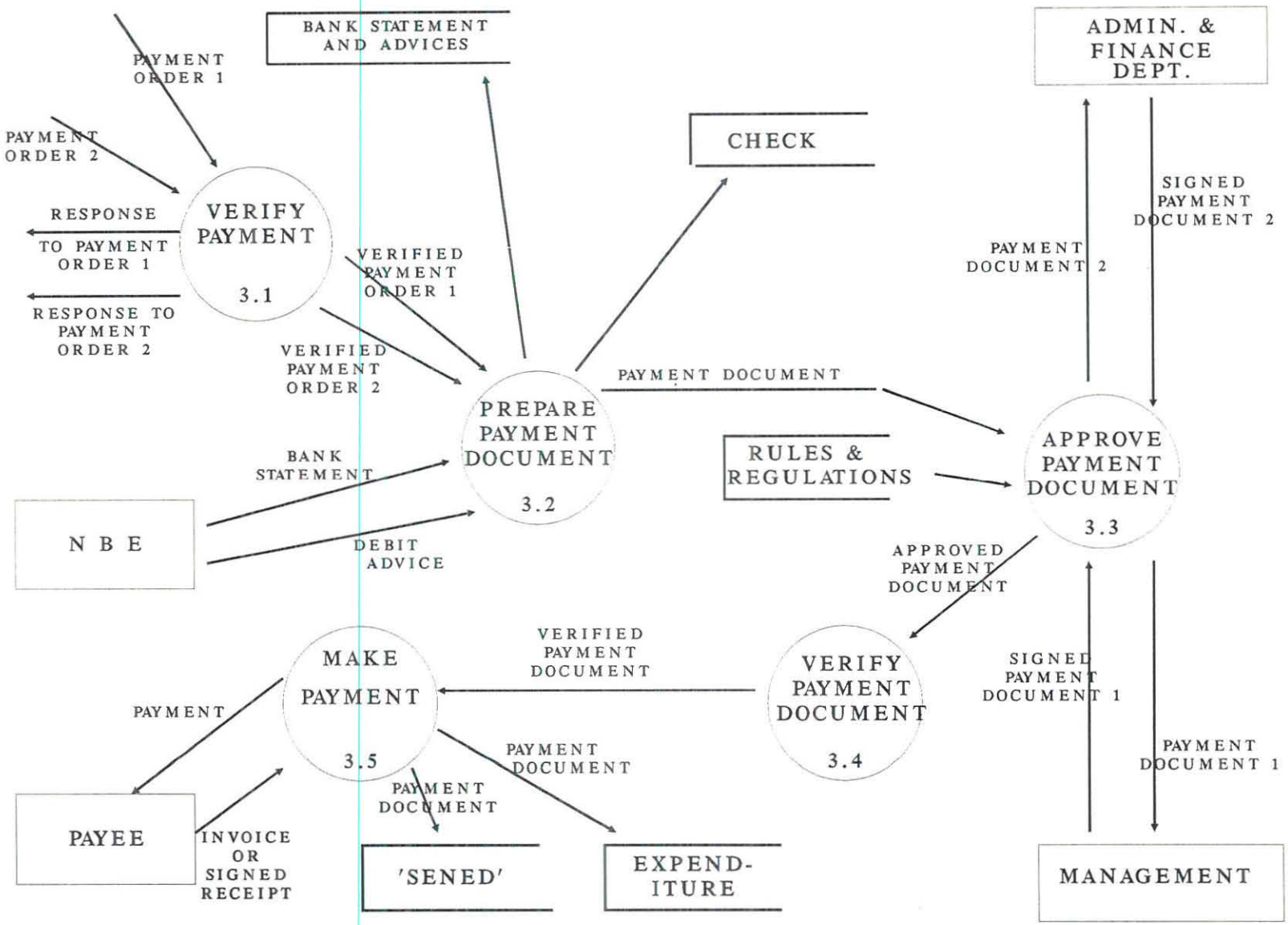


FIG 2.4 DIAGRAM 3- EXPLOSION OF 'PROCESS EXPENDITURE' PROCESS

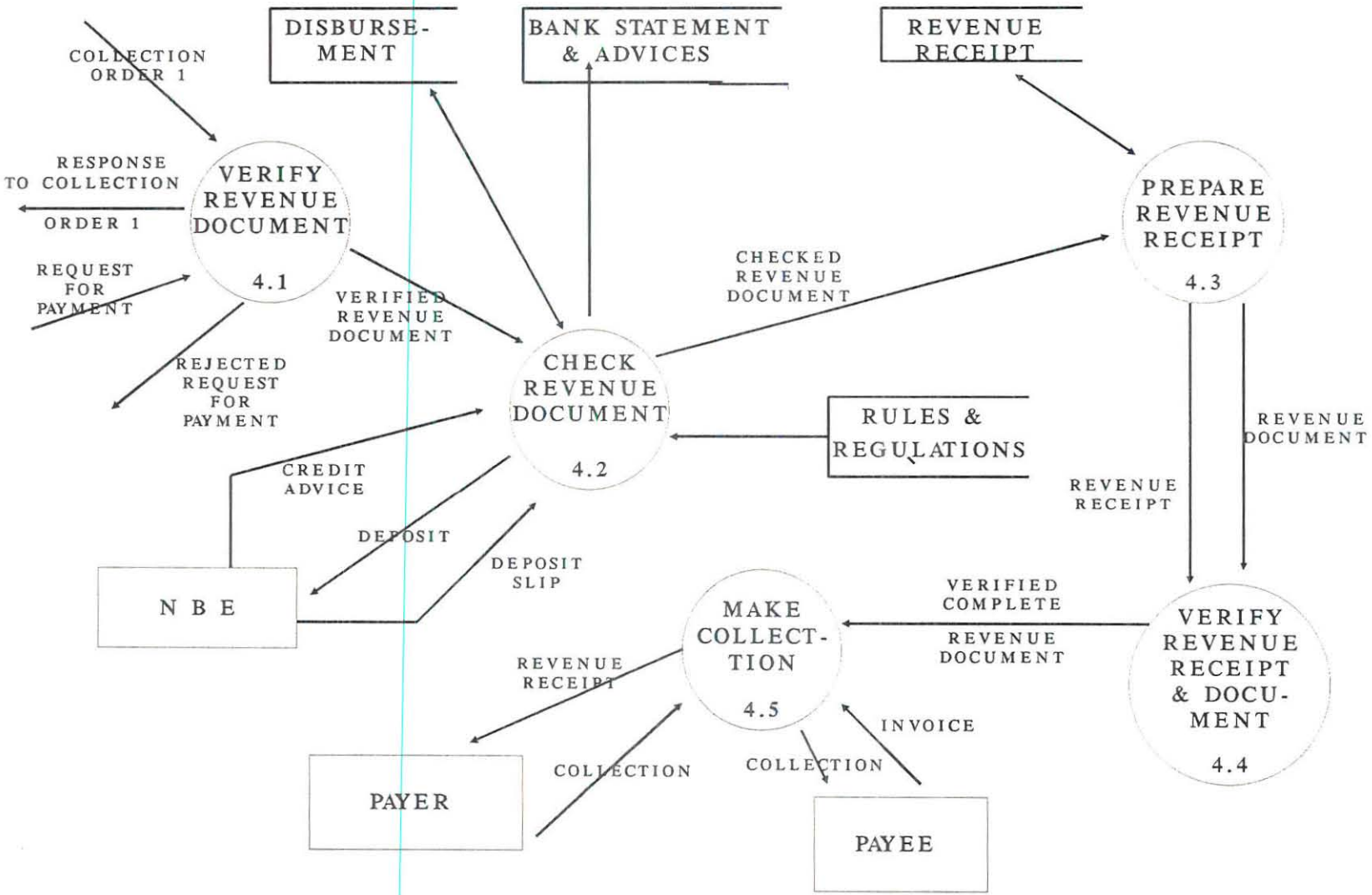


FIG 2.5 DIAGRAM 4 - EXPLOSION OF 'PROCESS REVENUE' PROCESS

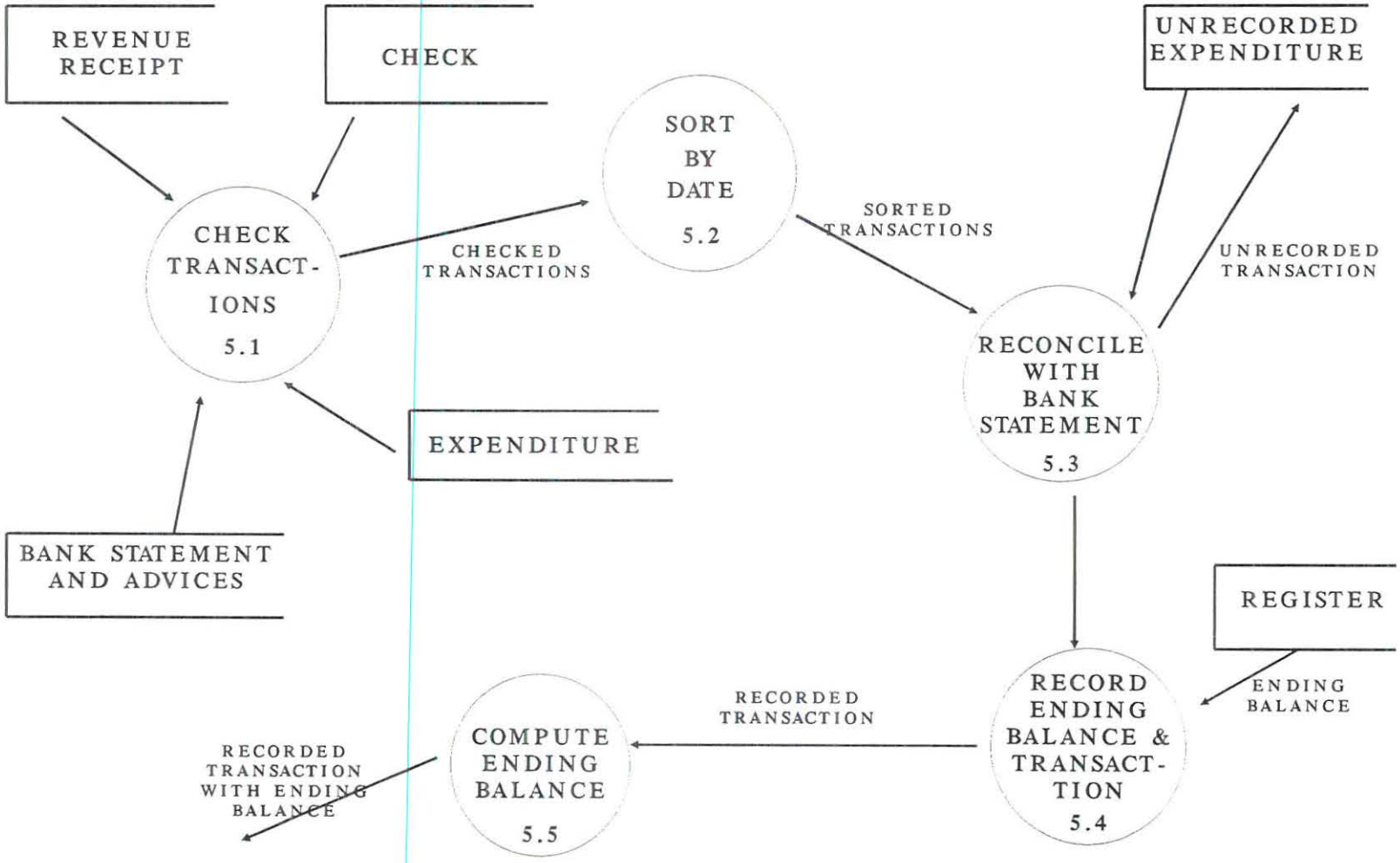


FIG 2.6 DIAGRAM 5 - EXPLOSION OF 'RECORD TRANSACTION' PROCESS

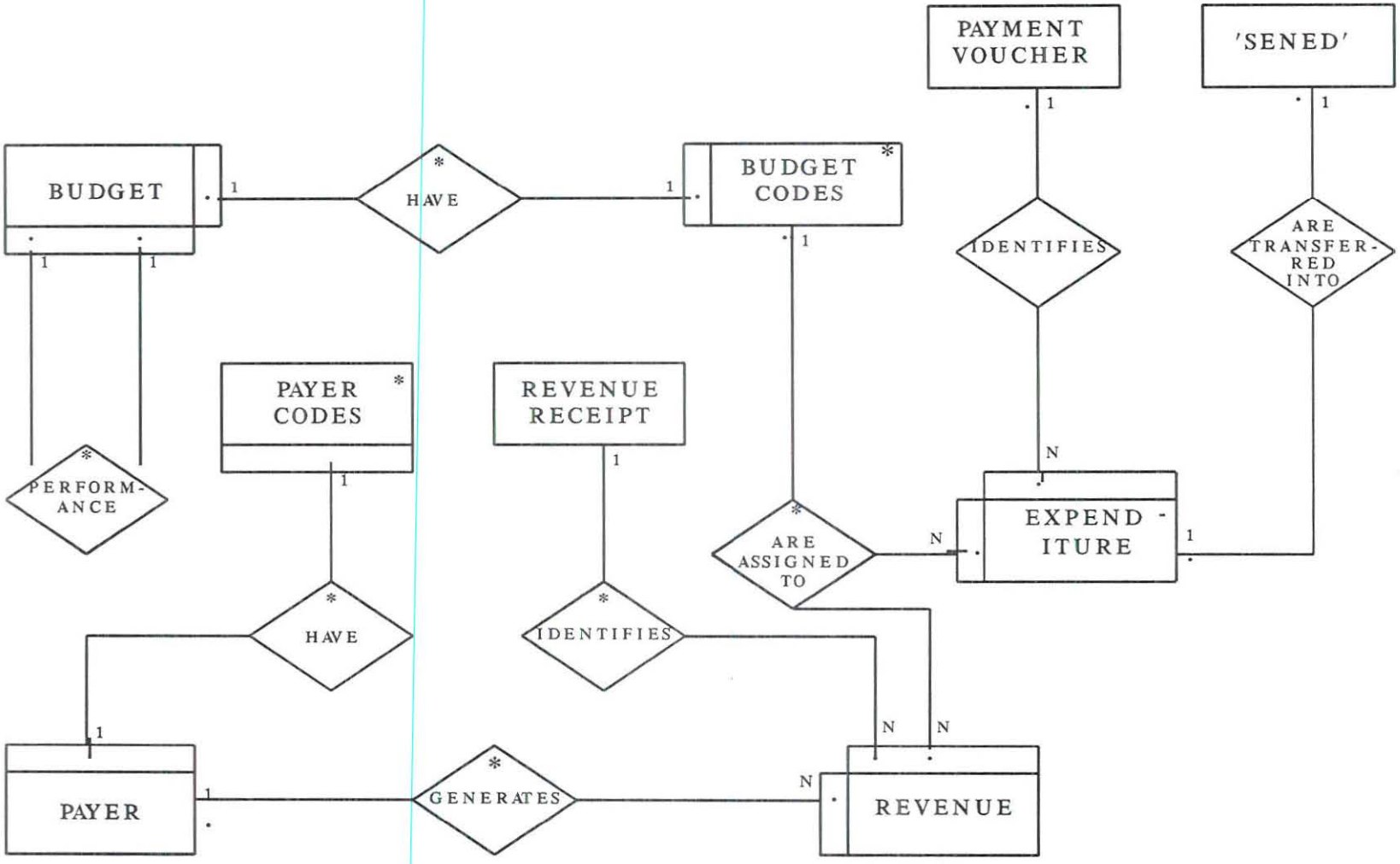


FIG 2.7 AN ENTITY - RELATIONSHIP (E-R) DIAGRAM
THE FUNDAMENTAL COMPONENTS OF NUPI FINANCIAL INFORMATION SYSTEM AND
THE RELATIONSHIP BETWEEN THE COMPONENTS

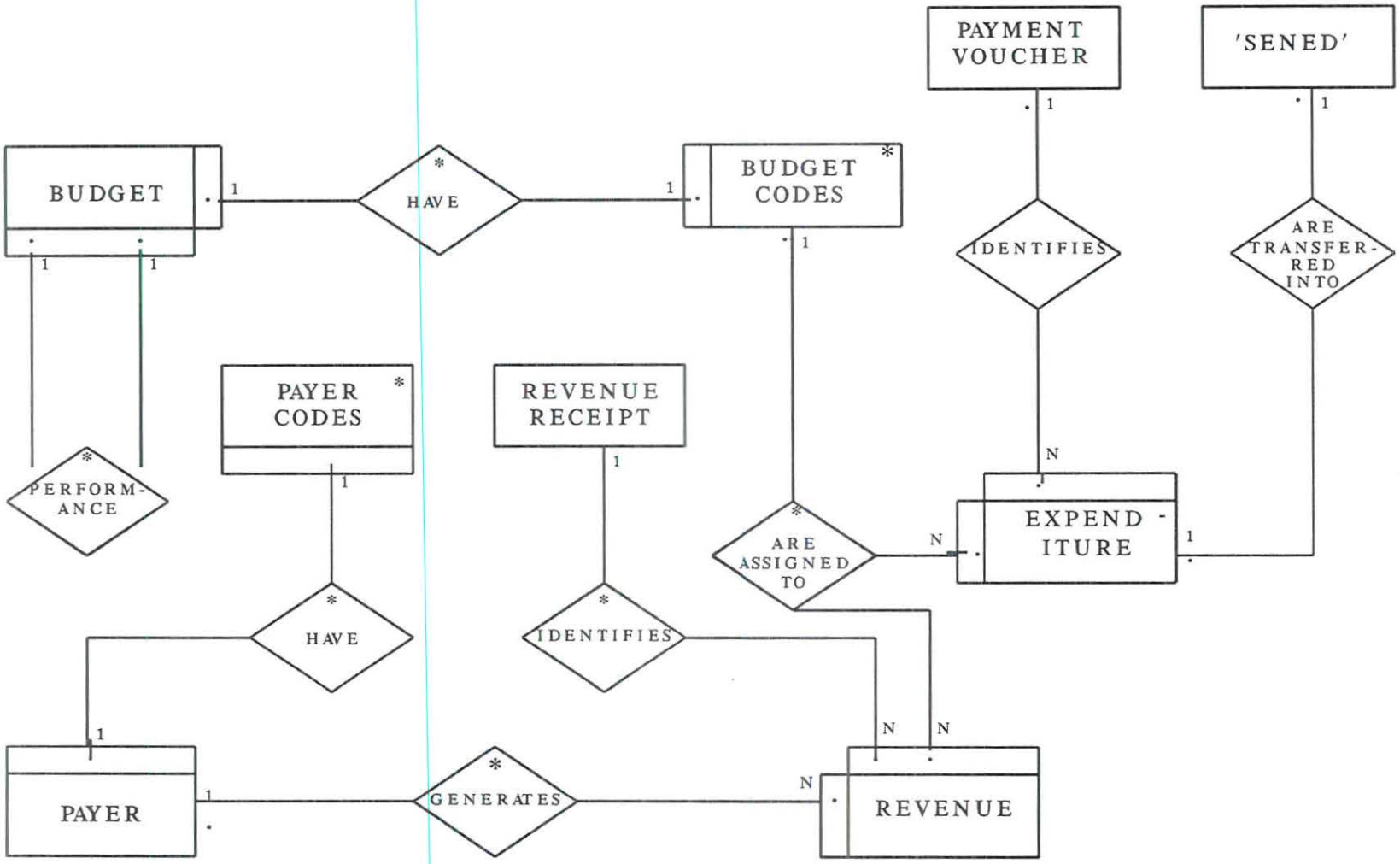


FIG 2.7 AN ENTITY - RELATIONSHIP (E-R) DIAGRAM
THE FUNDAMENTAL COMPONENTS OF NUPI FINANCIAL INFORMATION SYSTEM AND
THE RELATIONSHIP BETWEEN THE COMPONENTS

NATIONAL URBAN PLANNING INSTITUTE
FINANCE DIVISION
EXISTING INPUT/OUTPUT FORMS SUMMARY SHEET

Title	Code	Description	Source	Destination	Frequency	No. of copies	Comments
F_1	Ge/be/ma3/1	Approved recurrent budget notification form	MOF division	Finance division	Annually	1	
F_2	Ge/be/ma3/2	Approved capital budget notification form	MOF	"	"	1	
F_3	Ge/be/We11/1	Salary & allowance disbursement request form	Finance division	MOF	Monthly	7	Original
F_4	Ge/be/We11/2	recurrent budget disbursement request form	"	"	"	7	
F_5	Ge/be/We11/3	Capital budget disbursement request form	"	"	"	7	
F_6	Ge/Hi/29/1	Recurrent budget report	"	"	"	1	
F_7	Ge/Hi/29/2	Recurrent budget expenditure report	"	"	"	1	
F_8	Ge/Hi/29/3	Capital budget expenditure report	"	"	"	1	
F_9	Ge/Hi/29/4	Revenue expenditure report	"	"	"	1	
F_{10}	Ge/Hi/29/5	Summary report	"	"	"	1	
F_{11}	Ge/bema/4	Budget transfer	"	"	Unknown	7	Original
F_{12}	Ge/bema/6	Budget breakdown	"	"	Annually	7	
F_{13}	007	Travel request, authorization & advance	Field travellers	Finance	Unknown	2	
F_{14}	008	Travel & expense report	"	"	"	2	
F_{15}	Model/33	Payroll	Finance	Finance	"	3	
F_{16}	-	Bank statement	NBE	"	Monthly	1	
F_{17}	Ge/Hi/21/1	Revenue receipt	Finance	Finance	Unknown	3	
F_{18}	Model 6	Payment voucher	"	"	"	2	
F_{19}	-	Order 1	Mgt	"	"	1	
F_{20}	-	Order 2	Admin.& Finance	"	"	1	
F_{21}	-	Bank debit advice	NBE	Finance	"	1	
F_{22}	-	Bank credit advice	"	"	"	1	
F_{23}	-	Deposit slip	NBE	Finance	Unknown	1	
F_{24}	-	Maps & plans sales form	Plan & cartography service	"	"	2	
F_{25}	-	Invoice	Payee	"	"	1	
F_{26}	-	Budget status report	Finance	Mgt	Monthly	1	

FIG 2.8 Existing input / output forms summary sheet

NATIONAL URBAN PLANNING INSTITUTE
FINANCE DIVISION
EXISTING INPUT / OUTPUT DATA ITEMS SUMMARY SHEET

Data item	Form Number																								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F
Date	x	x	x	x	x				x		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Number	x	x	x	x	x						x	x			x		x	x	x	x	x	x	x		x
Identification	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		x	x							
Heading (Name)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Code	x	x				x	x	x	x	x		x													
Title	x	x	x	x	x	x	x	x	x	x	x	x													
Sub-title	x	x	x	x	x	x	x	x	x	x	x	x													
Sub-sub-title	x	x	x	x	x	x	x	x	x	x	x	x													
Central govt	x	x	x	x	x	x	x	x	x	x	x	x													
woreda	x	x	x	x	x	x	x	x	x	x	x	x													
Budget description	x	x				x	x	x	x	x		x													
Budget code	x	x	x	x	x	x	x	x	x	x		x						x							
Approved budget	x	x										x													
Financer orgn code			x	x	x																				
Financer orgn			x	x	x																				
Month / Year			x	x	x	x	x	x	x	x															
Approved year			x	x	x																				
Paid till now			x	x	x																				
Balance			x	x	x																				
Requested			x	x	x																				
Approved			x	x	x																				
Total	x	x	x	x	x	x	x	x	x	x	x	x				x									x
Salary & allowance			x																						
Deductions			x																						
Net pay			x																						
Amount in words			x	x	x												x	x							
Cashier name			x	x	x																				
NBE			x	x	x																				
Account #			x	x	x																				
Signature	x	x	x	x	x				x		x	x	x	x	x		x	x	x	x	x	x	x	x	x
Seal	x	x	x	x	x						x	x													
Copies for	x	x	x	x	x						x	x							x	x					

(Cont.)

Data item	Form Number																								
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F
Expenditure list				x	x																				
Revenue						x																			
Retuned revenue						x																			
Expenditure-recurrent							x																		
Returned expenditure recurrent							x																		
Expenditure-capital								x																	
Returned expenditure capital								x																	
Amount received									x																
Amount paid									x																
NB:									x	x		x		x			x								
Configuration									x																
Amount received for salary & allowance										x															
Amount paid for salary & allowance										x															
Amount received for recurrent expenditure										x															
Amount paid for recurrent expenditure										x															
Amount paid for capital expenditure										x							x								
Amount received for capital expenditure										x															

(Cont.)

Data item	Form Number																								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F
Amount paid for capital expenditure										x															
Amount received - 'Fesese'										x															
Amount paid - 'Fesese'										x															
Prepared by										x								x							
Approved by										x			x	x											
From -- budget code											x														
To -- budget code											x														
Transfer budget code description											x														
Account no.													x			x					x	x	x	x	
Requested by / for													x	x											
Unit													x	x											
Job title													x	x											
Trip to													x	x											
Purpose													x												
Date from -- to													x												
Routing														x											
Means of transport													x												
Advance required													x												
No. of passangers													x												
Time of departure													x												
No. of Kms													x												
Place of departure													x												
Serial no.														x	x	x									x
From														x											
To														x											

(Cont.)

Data item	Form Number																								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F
Duration of stay														x											
Total no. of days away form NUPI														x											
Perdium expense														x											
Expenses not covered by perdium														x											
Total expense														x											
Travel advance														x											
Due to Applicant														x											
Due to NUPI														x											
Registered by														x											
Employee Name															x										
Employee post															x										
Salary															x										
Working hours															x										
Gross salary															x										
Tax deductible allowance															x										
Car allowance															x										
Other non-tax deductible allowances															x										
Gross pay															x										
Income tax															x										
Penelity															x										
Credit															x										
Other deductions															x										
Total deduction															x										
Net pay															x										

(Cont.)

Data item	Form Number																								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F
warning															x										
Service charge																x									
Amount																x	x		x	x	x	x	x	x	x
Description																	x		x	x	x	x	x	x	x
Name of orgn																	x							x	x
Payer																	x								
Address																	x	x							
Revenue code																	x								
Purpose of payment							{										x	x							
Paid to																			x	x					
Kind of expenditure																		x							
Checked by																		x							
Check no.																		x							
To Administratin & Finance																			x						
To finance																				x					
Plan in Sq. Km																								x	
Amount / Sq. Km																								x	
Expected expenditure																									
Actual expenditure																									
Variance expenditure																									

. 2.9 Existing input/output Data items summary sheet

NATIONAL URBAN PLANNING INSTITUTE

FINANCE DIVISION

VOLUME OF TRANSACTION

EXPENDITURE

1 1/2 box files of payment vouchers per month.
1 1/2 * 12 = 18 box files payment vouchers per year.
A box file contains 40 payment vouchers.
40 * 18 = 720 payment vouchers per year are processed on average in a given year.

REVENUE

1 pad of revenue receipts are processed on average in a month.
A pad contains 50 leaves of receipts.
50 receipts * 12 = 600 revenue receipts are processed in a given year.

EMPLOYEES CREDIT PROCESSING

On average credit processing is made for 20 employees every month.

PAYROLL PROCESSING

Monthly, salary and deductions of 200 employees is calculated manually.

Table 2.10 Volume of transactions in the existing system

TOTAL STAFF OF THE FINANCE DIVISION

BY QUALIFICATION

SERIAL NO	POSITION	NO	QUALIFICATION
1	FINANCE DIVISION HEAD	1	B.A IN ACCOUNTING AND 6 YEARS OF EXPERIENCE
2	BUDGET SECTION HEAD	1	B.A IN ACCOUNTING AND 4 YEARS OF EXPERIENCE
3	ACCOUNTANT III	1	"
4	" II	1	DIPLOMA IN ACCOUNTING
5	" I	1	"
6	BUDGET CLERKS	2	"
7	CASHIER	1	12 GRADE COMPLETE

	TOTAL	8	

Table 2.11 Staff of the Finance Division

The following are computer system and software specifications the National Urban Planning Institute (NUPI) needs to have to equip its computer center with a LAN system.

1. Microcomputer System for LAN Configuration

1.1 Server

Processor	Intel Pentium 100MHz or Equivalent
Memory	32 MB 70ns xpandble to 136 MB
Hard Disk Drive	2 GB
Floppy Disk Drive	3.5" 1.44MB
Hard Disk controller	PCI Fast/Wide SCSI-2
Accessory Slots	EISA, ISA
Video Adapter	Super VGA, 1MB/2MB
Monitor	14" SVGA 0.28 mm dot pitch
Ports	2 Serial, 1 Parallel, Mouse & Keyboard ports
Case	Floor Stand
Network Interface Card	32 bit LAN Adapter

Pre-loaded Software

DOS 6.2 with original diskettes and manuals

Windows for Workgroups 3.11 with original diskettes and manuals.

1.2 Backup Tape

Capacity	Great than 550 MB
Data Transfer	8 MB / Minute

Physical Cartridge size	Standard
Media Cartridge	20

1.3 Client computers (Qty. 15)

Processor	Intel Pentium 75 MHZ or Equivalent
Memory	8 MB 70ns expandable to 64 MB
Hard Disk Drive	630 MB
floppy disk drive	3.5" 1.44MB and 5.25" 1.2 MB
Hard disk Controller	Enhanced IDE
Accessory Slots	ISA
Video Adapter	Super VGA, 512 KB
Monitor	14" SVGA 0.28 mm dot pitch
Ports	2 Serial, 1 Parallel, Mouse & Keyboard ports
Case	With 230W Power supply

Pre-loaded Software

DOS 6.2 with original diskettes and manuals

Windows for Workgroups 3.11 with original diskettes and manuals

1.4 Networking

Network Operating System	Novel Netware 3.12, 25 users
16 bit Lan adapter card	Qty. 15

With network administrator

Password for configuration protection

1.5 All necessary cables, connectors, terminators, etc. to install 15 work stations and peripherals in a LAN system in a room of dimension 9m X 20m.

2. Printer Configuration

2.1 Laser Printer - (Qty. 2)

Resolution	600dpi
Memory	2 MB
Printing Speed	12 PPM or more
Paper Size	A4
Toner Cartridge	Extra Cartridge
Cables	Parallel and Serial Cables
Software	Driver software in 3.5" diskettes

2.2 Dot Matrix Printer -(Qty. 3)

Number of Pins	24
Paper Loading	Single sheet and continuous feeder features automatic feeding / loading feature for single sheet
cut-sheet feeder	High capacity which holds upto 150 sheets
Printing speed	Letter capacity which holds upto 150 sheets
Number of fonts	12
Cables	Parallel Cables

3. Uninterruptable Power Supply Unit

3.1 Smart UPS

Quantity	One
Maximum Load	600 VA
Backup time	15 Min. minimum

3.2. **Backup UPS** (Qty. 15)

Quantity	Fifteen
Maximum Load	400 VA
Backup time	15 Min. minimum

Should include the following specifications:

Input protection	User resettable breaker
Battery	Spilling proof sealed battery
Battery life	1 year or above
Audible Alarms for	-Battery operation
	- Low battery indicator

With over current and short circuit protection

Lan interface card & Software requirement

Control Panel indicators

4. Other Peripherals

4.1 Plotter

Plotting Speed	Color, < 5 min.
Paper size	A0/E-size
Resolution	600 dpi
Buffer Size	4 MB maximum 68 MB
Network Connection	JetDirect Cards

4.2 Color Scanner

1600 dpi resolution Flat Bed Scanner

24 Bit color

Aldus PhotoStyle SE

Word Scan OCR

With PC Interfacing Kit

5. Software

Word processor:	MS Word for Windows Latest version
Computer Aided Design:	AutoCAD Rel. 12 for Windows
Desktop Publisher:	Aldus Page Maker Ver. 4.0 or latest
	Ventura Publisher Ver. 4.2 or higher
Statistical Analysis Package:	SPSS/PC+ for Windows Ver. 5.0

Note all the hardware and software items and peripherals must be accompanied by technical, installation and operating user manuals.

CHAPTER THREE

PROPOSED BROAD DESIGN

3.1 ASSESSMENT OF THE CURRENT FINANCIAL SYSTEM

Delay: Considerable delay is being observed in most of the functions of the division. Users complained that it is very difficult to obtain information timely and also to get work done on time, in the process of revenue collection or payment of expenditures. Checking process for making collection or payment is slow due to poorly organized paper files. Also there is delay in the recording process of transactions. All these contributes to the delay in the reporting process.

Inadequacy, inconsistency and inaccuracy of financial reports: In section 2.3, it is stated that the Budget Section of the Finance Division produces a budget status report periodically. Again from the interviews conducted and observation made, it is discovered that the said report does not reflect the real financial movement because of the following reasons.

- Only actual expenditures are included in the report. Since sened transactions are not registered, they are excluded from the report. But this gives a misleading picture to users. Seneds are committed resources, and this information should be communicated to users through this report so that it may reflect the real situation.

- The source for producing this report and the reports to MOF is not the same. For example budget status report is prepared from expenditure and other budget records where as monthly MOF report is produced from registers. This brings inconsistency between these two reports.

- The report is not exhaustive. It is not possible to tell from this report how much budget is requested for spending and how much of it is received during a certain period of time.

- Only two reports namely budget status report and summary report to MOF are generated from the system. This is not adequate.

Retrieval problem: difficulties are being experienced in retrieving information through various access points. For example, determining cost by item, cost by department, cost by project etc is too difficult and at times impossible. User expressed the need for a mechanism to know the cost of a project and breakdown of the total project cost.

From the discussion made with the users and from my own observation, the factors that contribute to the inability of the system to meet user requirements are:

Manual methods: In the day-to-day operation of the finance unit, there is a paper-based information involved. The recording, filing, processing, tracking and controlling of such financial

information are all done manually, with the aid of some obsolete (and often breaking) mechanical machines such as typewriters and calculators. Source documents, instructions, inter-office memo, reminders etc. relating to the finance operation are all stored in such manual filing system as table drawers, lateral filing cabinets and suspension files. Retrieval of required information from the existing system usually means manual browsing of papers hence very difficult & time consuming.

What is more, such information extracted from the files are prepared into reports via handwritten drafts, which are then converted to typed text using antiquated (and often breaking) machines.

Lack of a properly organised accounting system : According to my observation, not only the unit uses manual system but also is poorly organized or unsystematic in terms of commonly used standard bookkeeping practices. For instance;

- there is only one journal for common recording of all revenues and expenditures.
- no separate account is maintained for controlling the movement of each item that has a budget code.
- no record is maintained for sened (defined in section 1.7) accounts. Another major area of NUPI finance is management of sened transactions.
- records for staff debtors are not available.
- From the existing records, it is very difficult to obtain such frequently required information as plan preparation cost, total purchase for certain

inventory, and total expenditure of the institute for one quarter of a year.

Use of single-entry accounting systems : All government organizations fully or partially financed by central treasury are required to use almost the same standard forms, registers, models etc. in capturing, storing, recording and reporting transactions based on single entry accounting principles. This method, by its very nature, does not allow to have multiple access points to the accounting books to enable users to obtain the information they need. Usually, additional internal bookkeeping system, complementary to the already existing single entry system, is required to meet such needs.

Shortage of manpower: Manpower is among the essential factors that contribute to the success or failure of a system. Much of employee time goes into the entering, copying, sorting and looking up detailed financial information. The existing system, however does not have enough manpower to carry out these functions. Users also state the lack of strict division of labour even among the existing staff, to contribute to the shortage. An employee of the unit is observed performing many different tasks and this is not in accordance with the rules and regulations governing the unit. As a result of lack of enough manpower, things are not done at reasonable period of time.

From my observation and users opinion all these mean that the existing methods are not only incapable of providing required information for users, but also very slow, over-taxed and prone to error and hence a handicap to the smooth operation of NUPI.

3.2.1 Main Objectives of the Proposed System

On the basis of my findings during the analysis and own experience, the following objectives are identified for consideration in the design of an automated solution in replacement of existing manual system.

- to improve timeliness in the activities of the unit to ensure that;
  a) collection or payment of money may be made timely;
  b) financial reports are generated timely;
  c) fast retrieval of information could be attained;
- to improve quality of financial reports;
- to provide various access points to records; and
- to maintain complete records.

3.2.2 The Proposed Solution

Among the major alternatives put forward by users themselves in overcoming the limitations of the existing system are:

- increasing the staff of the finance division to overcome the delay in processing and reporting, which is the major problem, and to leave things as they are;
- changing the single-entry system into double-entry accounting;
- automating some or all of the processes.

According to my observation and discussion made with officials during my study and own working experience with NUPI, the automation option sums a technically, economically and operationally better feasible solution. Among the main reasons for not considering the first option (increasing the workforce) as a better solution are the following.

- This option may not be economically feasible as it increases costs for salary, space, overhead etc;
- It increases paperwork;
- This decision leads to inefficient utilization of investment on computer resources. As a policy the management does not subscribe to the idea of solving the problem by increasing the size of the work force in the division.

Although previous studies have suggested the replacement of the single-entry system into double-entry accounting, it was not possible to implement mainly because the institute is directed by MOF not to use its own different system than the standard adopted by the ministry for all Government organisations and institutions.

From informal discussions made with employees of the ministry, it is also found out that the existing system will continue as it is and request for such change is not among the priorities - such a change may not be expected in the near future.

Running both the single-entry system per MOF requirements and double-entry system for internal purpose could be possible but

is not feasible as it duplicates time and effort to perform activities as well as complicates things.

Automation is the proposed option because the potential benefits in terms of reduction in clerical operations, space requirement, paperwork, duplication of files, increased accuracy and improved information outweigh the development costs. The development costs include manpower, hardware, software and other additional costs. Computer professionals of the institute stated, computerizing functions of finance increases efficient utilization of the existing computer resources, minimize manual function and increase the possibility of organizing financial information into meaningful reports.

There is a substantial evidence that computer based methods are effective and efficient in handling financial record.

> Because of the similarity in their nature, few business activities lend themselves as well to automated data processing as does accounting. Accounting applications were among the first commercial uses of digital computers. In fact, it was their importance to accounting operations that caused data processing departments in organisations to be initially responsible to accounting and financial vice presidents. ... accounting remains and enormously important function that computer perform very well. (Brightman and Dimsdale 1986).

As Winsten (1990) puts it;

> " Computers, in some respects, are much more reliable than humans. They can carry out a complex set of instructions an infinite number of times without changing a single step or slowing down. This ability to follow instructions to the letter, no matter how long or complex and to work at a constant speed is one of the great features of electronic data processing. Computers are not plagued by fatigue, absenteeism or any of those unexpected human lapses that seem to be inevitable".

Nowadays, computer-based information system have influenced organisations of all types, including the not-for-profit, such as government departments, hospitals and schools. (Hadera 1995). From the above statements, it could safely be concluded that automating financial activities leads to a better information system. And the main reason behind this study is to demonstrate how computer-assisted methods yield an improved financial information system.

As stated in the thesis proposal, the institute has purchased fifteen microcomputers (intel pentium 75 MHZ with memory capacity of 8MB 70ns expandable to 64 MB, with hard disk drive 630 MB, and other necessary LAN configuration. No additional hardware cost is incurred for developing the finance system. Also, system softwares are purchased. The only required software is financial package. Also, it is the view of the management that a computer-based system well interfaced with the manual system could meet its requirements.

The institute has enough system analysts, programmers, qualified users to form a team to manage the automation. At least as a starting point, no additional manpower such as consultants should be hired for the automation work. In undertaking this project NUPI is only utilizing its system people and computer resources efficiently. Other minor costs could incur but the potential benefit is much more than the cost.

Once computer is selected as a feasible option, the next step is to choose among various design alternatives on issues such as which processes should be automated and which processes should remain manual.DFDs showing the selected processes for change , the automation boundary and the new process are presented in fig. 3.1 and 3.2.

It is believed that, automating two of the processes namely recording and reporting processes will overcome most of the identified problems of the existing system. In addition to computerization, the following changes are recommended to improve existing situation.

- Manpower: Even if the system is recommended to be computerized, the division must have the necessary key personnel. Some of the employees in the division are temporarily transferred to finance to handle the problem of manpower. To maintain smooth operation, some additional manpower is demanded.
- The other problem is the existing filing system. A better system should be designed to make possible easy access to information on letters, memorandums, inter-office memos, circulars, etc whenever required.

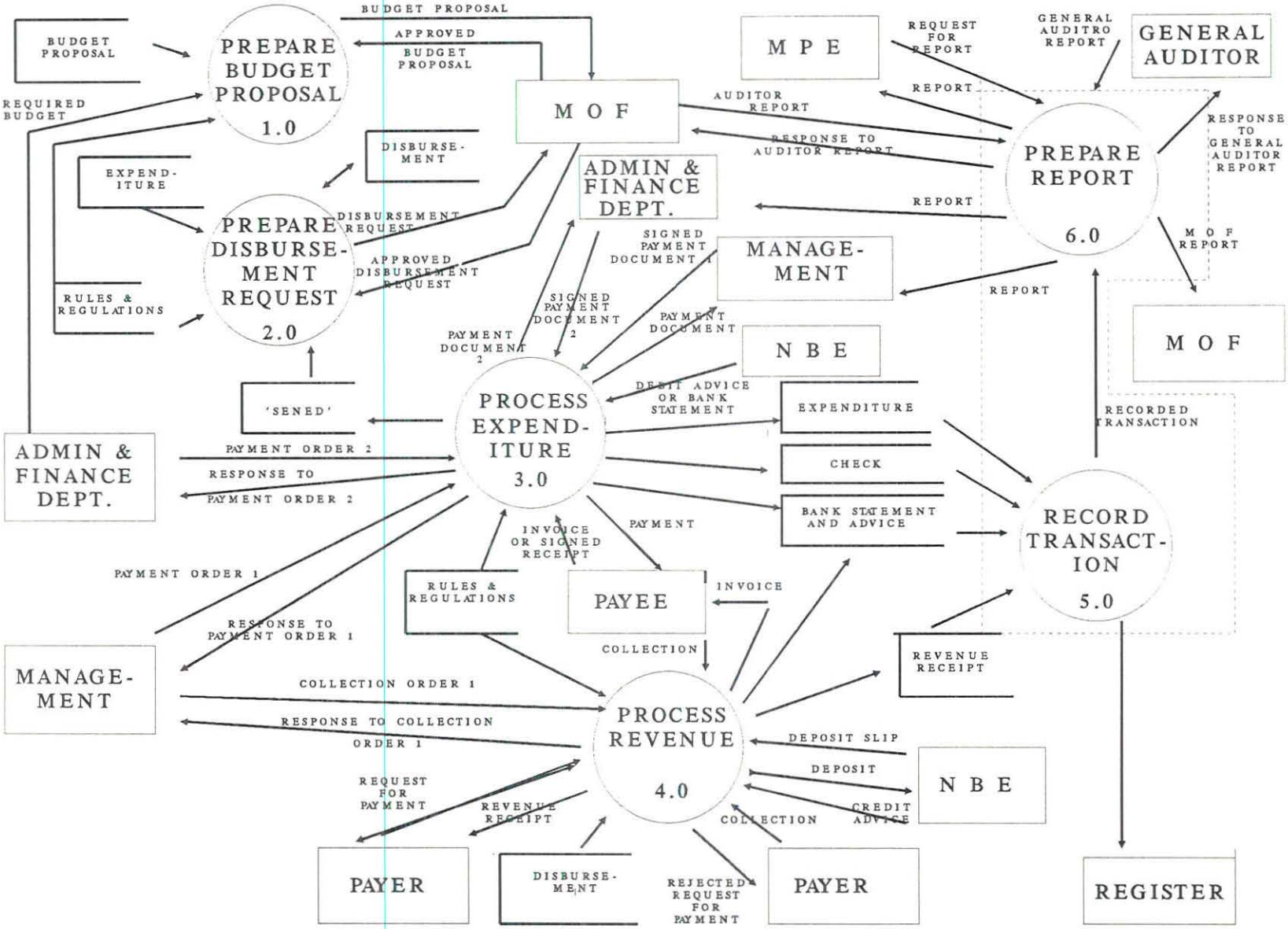


Fig 3.1 : A TOP-LEVEL DIAGRAM SHOWING THE AUTOMATION BOUNDARY

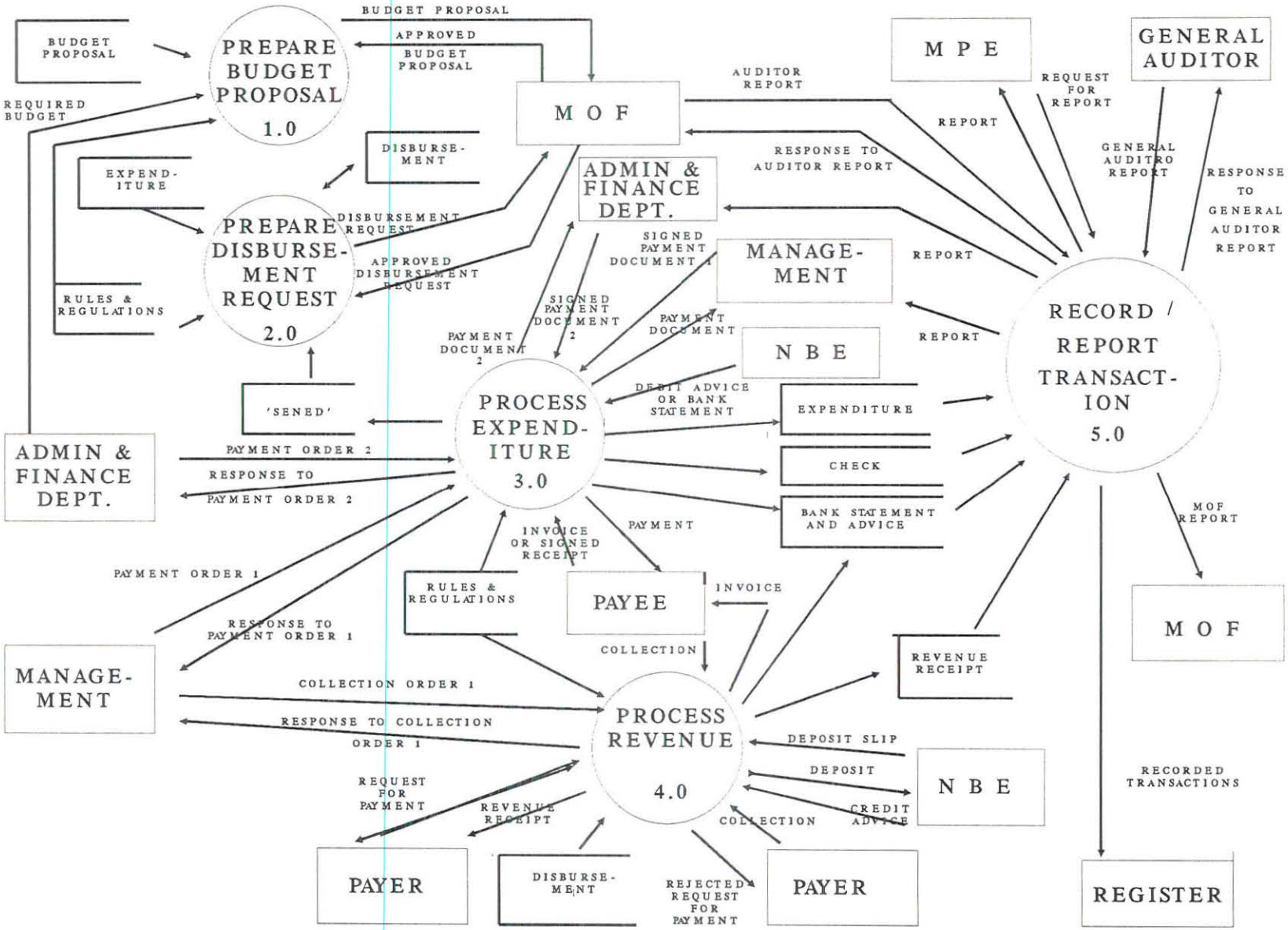


FIG 3.2: A TOP-LEVEL DIAGRAM SHOWING THE NEW FINANCIAL INFORMATION SYSTEM

3.2.3 Components of the proposed computer-based System

In section 3.2.2, it is indicated that computerizing recording and reporting processes is the proposed solution for the existing problems of the finance system. This section describes the components of the proposed system in detail (Refer to Fig 3.1 and Fig 3.2 for the DFD).

Revenue/Expenditure system: Under this system, there are three inter related sub-systems. One for revenue, one for expenditure and one for budget. Any transaction falls into one of these three categories. A transaction may be a revenue transaction, expenditure transaction or budget transaction. Of course a transaction could be actual or sened. But since there is a separate system for handling seneds, this section deals with only actual transactions. Since a transaction falls into one of the three categories, three data entry screens, one for each are designed. The input screen for revenue is the paper revenue receipt as it is. All data items on the revenue receipt are converted into computer screen. The input screen for expenditure is the payment voucher. The data items on payment voucher will be displayed into computer screen. No new input forms are designed so that the system will be easy to use. This is in the sense that the system users are assumed as novice users as it is their first time to use computer systems. Therefore the input form should be as simple, familiar, clear and understandable as possible so that data entry error could be minimized. What is expected from data entry clerk is to simply key-in whatever information he/she has on revenue receipt/payment voucher into computer screen. For budgets, a budget transaction does not

occur frequently. It occurs once in a month or once in a year. A new input form is designed for entering budget transactions. Attempt is made to keep this form simple and clear. All the designed forms are shown in figs 5.3, 5.4, 5.5 & 5.6.

Then the system stores data on the respective databases. In the manual method recording is done only once in a month. This brings delay in other functions of the system. But now, transactions are recorded (entered into computer system) as they occur hence up to date information could be obtained whenever required, also delay for reporting will be minimized.

The system generates various standard financial reports such as monthly report, bank reconciliation and budget report. Also the system allows user to design report formats and produce a report other than those stated when such need arises by management or other users. Facilities to view and print information using various access points such as budget code, department, town, date are also incorporated. This system is expected to solve the major problems of the existing system by allowing;

- preparation of timely financial reports;
- generation of better quality and various financial reports as the system will be integrated with other systems, such as sened to produce improved information;
- consistency in report generation process;
- complete information on financial reports, as the databases contain exhaustive information about each and every transaction;
- various access points to records;

- recording of transactions timely.

Sened system: The institute should keep the amount of sened transactions to the minimum possible. Because at the end of every budget year, on June 30, sened transactions are counted by MOF auditors and the institute is required to return that much of money to central treasury from the coming year approved budget. In the coming year, approved budget less total sened amount of last year is given to the institute to carry out its activities. This greatly affects the activities planned to be performed in that year.

Therefore, whatever information regarding sened should be available, so that action by concerned parties may be taken, in order to minimize the transactions. Under the manual methods, there is no record or report for such transactions. As a result a large amount of money is requested by MOF from the institute. For instance on April 1996, the finance division is informed by people from MOF that the institute is required to return 500,000 Eth birr to central treasury. To tackle sened problems, a sened system is designed. The system allows users:

- to store all sened transactions;
- to make changes on the transactions;
- to delete the transactions when transferred to expenditure;
- to view on screen any information regarding sened and transfers to expenditures;
- to produce various reports to satisfy management needs and to follow up individuals such as purchasers and employees;

- various access points such as sened holder, budget code, name of organisations for whom sened is issued, amount, date.

The system is expected to solve some of the problems of the existing finance system by allowing:

- finance people to keep well informed about movement of sened transactions so that proper action may be taken. For example the division could easily know the total outstanding sened on the purchaser's hand at any point of time. Then a statement can be sent to the purchaser to settle those transactions immediately. Also for larger amounts, finance division will be in a position to report (remind) management to take action to solve problem of outstanding seneds.

- fast checking process. This minimizes the time required to process payments. And also to implement financial rules. For example, an individual is not allowed to hold more than one sened transaction except the purchaser. From the system it is very easy to detect those who are not allowed to take additional money from finance. Therefore they will be forced to settle their outstanding sened. Another advantage of fast checking is to give people clearance as they required.

- information about transfer. From the system, it is possible to detect which sened is transferred into what expenditure, when, by whom etc. for auditing purposes. It is very difficult to find this information on manual system since once sened is transferred into expenditure the sened is destroyed. 'Transfer record' is a new feature of the computer system.

All the above points mean, as a result of the establishment of sened system, NUPI could be in a position to use budget for a given year only for activities of that year. This enhances efficient utilization of budgetary resources.

Payroll: Payroll processing is among the major functions that need automation because of the routine nature of the work. This system keeps a separate account for each employee to keep track of employees financial status as they have many additions and deductions from their salary every month. Also the system calculates gross salary, total deduction and net pay for each employee, produces a summary sheet for each employee and for the total. This system is expected to solve some problem as it;

- allows access to employees records by various access points hence facilitates retrieval of information.
- processes payroll. Every month the staff of finance division will not be busy with calculating salaries manually. This means delay in getting works done will be minimized; and
- allows fast checking process.

Finally, as a result of this system, complete information about employees could be obtained from finance division.

Inventory system: This system is required because there is a difficulty in obtaining information about stock from store. The store is managed manually, and in my opinion there are many problems in the system. For the future the inventory system will be shared between finance and store. But for the time being this system will be maintained by finance to control movement of stock so that financial resources for purchases could be organized. This system provides information about each items of the store and the need by all departments especially technical departments (since they are facing problem in obtaining such items) for these items. It also provides a control mechanism to prevent purchasing of unnecessary materials so that efficient utilization of finance is enhanced.

The two mandatory system not only for NUPI but for all similar organizations that use single - entry governmental accounting systems are the revenue/expenditure system and the sened system. Other systems such as payroll, inventory or other could be established if the organisation does not have already one or if such need arises.

CHAPTER FOUR

LOGICAL DATABASE DESIGN

4.1 RELATIONAL ANALYSIS

" A database is a collection of non-redundant data shareable between different application systems" Hawe (1991). Data should be stored in a common pool so that sharing the resource among application systems is possible at enterprise level. This approach of handling data is termed as database approach to systems design. The advantages of such approach over application - centred approach where a separate data files are maintained for applications. Major advantages of the database approach.

- elimination of unnecessary duplication (redundancy) of data between different master files;
- minimization of data inconsistency; and
- ability of resource sharing.

It is not recommendable to construct a single database for the whole activities of an enterprise as it is impractical to consider all details of the activities within limited period of time. As Howe states,

> "It is often better to construct a number of separate databases, each one covering a single well - defined application area (eg. personnel/ payroll/ manpower planning), but with each database being designed to fit into an overall outline business plan. The scope of the

database should be small enough to be manageable, but large enough to provide adequate benefits from sharing data between the individual applications".

A database is designed for the financial system of NUPI taking the forgoing consideration into account. The following account describes in detail the databases designed for the proposed NUPI finance system.

One component of the database design is the E - R model where data requirements of the new system are defined and subsequently converted into physical database. But in the case of NUPI finance system, the system design process does not produce a new E-R model mainly because the existing standard single entry system is adopted as it is, for the reasons explained in the previous section. As a result the E-R model of the existing system shown on fig 2.7 is considered as it is for the design of the database.

The process of converting the E-R model to a database follows two steps. These are;

- to carry out detail analysis of data in order to get the data model. For this purpose, a relational theory is used and the E-R model is converted into a set of tables or relations; and
- to examine the tables for redundancy and change those relations that are in redundant form into a non - redundant form.

The outcome of these two steps is a non - redundant relational model of the new system.

An E-R diagram is converted into tables/ relations by replacing each entity and relationship on the diagram by a table (The term table and relation are used inter changeably in this study). A relation has a name, columns and rows. The name of the entity/ relationship becomes a relation name. The attributes of the entity/ relationship becomes the relation columns. The occurrence of an entity/ relationship becomes a raw. In this work, for the purpose of brevity (also because the system is not so large), tabular representation of data is preferred to relational notation.

4.1.1 Relationship Properties

Two properties of relationships, namely, degree of relationship and membership class are discussed, as these apply to the current work, in the following.

Degree of relationship: A relationship degree are of three types, namely 1:1 (read as one to one) relationship, 1:many relationship and many:many relationship. If an occurrence of an entity A enters into a relationship with exactly one occurrence of another entity B, and vice versa, the relationship is 1:1. If an occurrence of entity A enters into relationship with many occurrences of entity B, and if an occurrence of entity B enters into a relationship with exactly one occurrence of entity A, such relationship is referred to as 1:many. And if an occurrence of entity A enters into relationship with many occurrences of entity B and vice versa, the situation is referred to as many: many relationship. The relationship degree corresponds to the existing enterprise rules.

The second important property is membership class. Again two situations of relationships could be identified: obligatory and non - obligatory. If every occurrence of an entity must participate in a relationship, then such a relationship is termed as obligatory where as if an occurrence of an entity is allowed by the enterprise rule to exist independently of the relationship, this situation is referred to as non - obligatory membership. These properties in the E - R diagram of fig 2.7 are explained below.

Budgets have Budget codes: Entity Budget and entity Budget Codes are in a 1:1, obligatory relationship of Have. The enterprise rules for the relationship are;

- A budget title have exactly one budget code.
- A budget code is given to exactly one budget title.
- A budget title must have a budget code.
- A budget code must be given to a budget title.

Budget Codes are assigned to Revenue: The entity Budget Codes have a 1:many non - obligatory relationship with revenue, while entity Revenue have an obligatory relationship with Budget Codes. When a revenue is collected, a budget code is assigned to it. For many such occurrences one code may be assigned. For example collection from sell of bid document for many stationeries is given only one budget code where as one revenue is assigned exactly one code. It is not possible to assign two budget codes for one revenue, hence the degree of relationship between them is 1:many. A budget code may / may not enter into relationship with revenue but a revenue must be assigned (must enter into assignment

relationship) with Budget Code. A revenue can not exist without being assigned a Budget Code. Then membership class of Budget Code is non-obligatory and Revenue membership is obligatory. The enterprise rules for the relationship are;

- A budget code may be assigned to many revenues.
- A revenue is assigned to exactly on budget code.
- A budget code need not be assigned to revenues.
- A revenue must be assigned a budget code.

Budget Codes are assigned to Expenditures: The entity Budget Codes have a 1:many, non - obligatory membership with expenditures and entity Expenditures have an obligatory membership. The discussion for relationship Budget Codes are assigned to Revenues holds true for assignment relationship between Budget Codes and Expenditures. The enterprise rules for the relationship are;

- A budget code may be assigned to many expenditure.
- An expenditure is assigned exactly one budget code.
- A budget code need not be assigned to expenditures.
- An expenditure must be assigned a budget code.

Payment Voucher identifies Expenditure: An entity Payment Voucher has a 1;many, non - obligatory identification relationship with entity Expenditure and the membership class between the two entities is obligatory. A one unique payment voucher is assigned to one or more expenditures. Many expenditures could be identified by one payment voucher but an expenditure could not be assigned to more than one payment voucher. Also a Payment Voucher could exist independent of entering into assignment relationship. Therefore the membership class is non-obligatory where as an

expenditure must necessarily attached to a payment voucher hence the relationship is obligatory. The existing enterprise rules for the relationship are;

- A payment voucher may identify many expenditure.
- An expenditure is identified by exactly one payment voucher.
- A payment voucher need not identify any expenditure.
- An expenditure must be identified by a payment voucher.

Revenue Receipt identifies Revenue: An entity Revenue Receipt has a 1:many, non- obligatory relationship with entity Revenue. Explanation for a relationship that exists between entity Payment Voucher and expenditure holds true for this relationship. A revenue receipt exists where it identified revenue or not. It does not necessarily enter into identification relationship therefore the relationship is non-obligatory. But a revenue is obliged to enter into identification relationship with the receipt hence the membership class is obligatory. The enterprise rules for the relationship are;

- A revenue receipt may identify many revenues.
- A revenue is identified by exactly one revenue receipt.
- A revenue receipt need not identify any revenue.
- A revenue must be identified by a revenue receipt.

Sened are transferred into Expenditure: Entity Sened and entity Expenditure are in a 1:1, relationship that has non - obligatory membership. If an occurrence Sened is completed, it may be

transferred to expenditure; if the process is not over, it stays as sened or if payment is returned, it will be cancelled. Also an expenditure may be transferred from sened or it may be a direct expenditure. Therefore both ways the relationship is non - obligatory. As to the degree of relationship, each sened can not be transferred into more than one expenditure and each expenditure can not come from two seneds. In such a situation, there is a 1:1 relationship between then. The enterprise rules for the relationship are:

- A sened is transferred into exactly one expenditure.
- An expenditure is transferred from exactly one sened.
- A sened need not be transferred into expenditure.
- An expenditure need not be transferred from a sened.

Payers have Payer codes: There is a relationship have between entities Payer and Payer Codes. The degree of relationship between them is 1:1 and the membership class is obligatory for both entities. A payer is given only one code and a code is assigned to only one payer. A payer is obliged to have a payer code. There is no payer code which is not assigned to payers. The enterprise rules for the relationship are:

- A payer have exactly one payer code.
- A payer code is given to exactly one payer.
- A payer must have a payer code.
- A payer code must be given to a payer.

Payer generates Revenue : There is a relationship Generates between the two entities Payer and Revenue. Their degree of relationship is 1: many, obligatory membership. A payer could generate many revenues. For example a stationary could pay for

purchasing bid document as well as maps and plans. Where as a certain revenue comes only from one payer. As a result the relationship is 1:many. If an occurrence is identified as a payer, automatically, he/she enters into generation relationship with revenue. Also a revenue does not exist without being generated by a payer, hence the relationship is obligatory for both entities. The enterprise rules for the relationship are:

- A payer may generates many revenues.
- A revenue is generated by exactly one payer.
- A payer must generate revenues.
- A revenue must be generated by a payer.

Budgets have Performance: A 1:1 obligatory relationship exists between the entities Budget and Performance. A budget has one performance. For example approved budget of code 8303 can have only one used budget for the code. Also a certain performance is attached only with one budget. If a budget exists, there is necessarily a performance. A performance can not exist without budget. The enterprise rules for the relationship are:

- A budget have exactly one performance.
- A performance is attached to exactly one budget.
- A budget must have a performance.
- A performance must be attached t a certain budget.

After the entities and relationships are identified, the next step is to convert them into tables or relations as it is mentioned before. The conversion follows certain rules based on the properties of the relationship. For 1:1 relationship the rules followed are:

- If membership is obligatory for both entities, put all attributes into a single table/ relation.
- If membership is non-obligatory for both entities, define three tables, one for each entity and one for the relationship.

For 1:many relationship the rules followed are;

- If membership of the 'many' entity is obligatory, define two tables, one for each entity. Post the identifier of the '1' entity into the 'many' entity's table.

Since only these rules are relevant the diagram on fig 2.7, the other rules are not discussed. An entity or relationship which is not represented by its own table as a result of the above stated rules is marked on the E-R diagram with an asterisk.

The tables for the finance system are presented by the following relational notation which shows the relation name and its attributes.

REVENUE (Revenue Receipt No, Receipt Date, Payer, Budget Code, Budget Title, Amount,Check No, Registered Date, Cashier, Payer Code);

EXPENDITURE(Payment Voucher No, Payment Date, Payee, Budget Code, Amount, Description, Registered Date, Check No, Authorised By, Cashier, Town, Department);

BUDGET (Budget Year, Budget Code, Budget Type, Budget Title, Proposed Budget, Approved Budget, Requested for Spending, Received for Spending, Used, Balance, Returned to MOF, Comment);

SENED (Sened No, Payment Date, Paid To, Payee, Budget Codes, Amount, Description, Check No, Authorized By, Cashier, Town, Department); and

TRANSFER(Payment Voucher No, Sened No, Budget Code, Description, Transferred Amount, Amount Returned, Amount Paid).

The underlined attributes are the relation keys. At this level only the conceptual model of data is developed. The next step is to examine for redundancy and remove the redundancies through a process called normalization. It is the normalized model that is converted into a physical database. For illustration purpose the following transactions of a month in a given budget year are used.

1. MOF releases for NUPI from the capital budget of the institute birr 8000.- (details are shown on tables)
2. A stationary purchased a bid document in birr 50.-
3. Another stationary purchased a bid document in birr 50.-
4. Cash is paid for small expenditures, amounted birr 500.-
5. Cash is paid for utilities, amounted birr 2900.75
6. Cash is paid for field allowance, amounted birr 1500.-
7. Cash is paid for utilities, amounted birr 5000.- and is added to sened,
8. Cash is paid for field allowance, amounted birr 5000.-, sened.
9. Cash is paid for fuel, birr 6000.-, sened.
10. Cash is paid for utilities, birr 2900.-, sened.
11. Cash is paid for field allowance, birr 2100.-, sened.

Budget activities for the year 1994 and 1995 are presented in table BUDGET. The normalization process is presented as follows.

4.1.2 Normalization

The ultimate goal of normalization technique is elimination of redundancy to arrive at tables that have non-redundant attribute values. A data is redundant if its deletion causes no loss of information. It is unnecessary duplication. Tables with non-redundant attribute values are termed as fully-normalised tables. To determine whether a table contains redundant values or not, it must satisfy certain restrictions or constraint. Relations that satisfy such constraints are in higher normal forms. The constraints are defined in terms of functional dependencies and relation keys.

Functional Dependency: Attribute Y is said to be functionally dependent on attribute X, if the value of X in a relation determines a particular value of Y. If the value of X is known, then a unique value of Y could be determined. For each value of X, there is one value of Y. A functional dependency is expressed as:

X ---> Y X determines Y;
          X is a determinant of Y;
          X is determined by X;
          X is functionally dependent on X.

Example: Revenue receipt # ---> Payer

Relation Key: is a column/columns (attribute/attributes) whose value/values select a unique relation raw (tuple). A relation key is an identifier or a label for a raw.

The normalization process includes five stages namely: first, second, third, fourth and fifth normal forms. As one proceeds through the stages, certain undesirable features are eliminated from the un normalized table. For the purpose of this study, only the first three normal forms are discussed and used.

A table is in first normal form if each column and raw contain simple values or no repeating groups.

A table is in second normal form if non - key attributes which are not functionally dependent on the whole of the identifer (relation key) are eliminated. The relations in second normal form (2NF) must satisfy the following constraint:

> All non-key attributes must be functionally dependent on the whole of each relation key.

A rule followed to satisfy the constraint is ' Remove the Offending Functional Dependency'.

A table is in the third normal form if there is no functional dependency between non-key attributes. Therefore the relation in

the third normal form must satisfy the following constraint.

> There must be no dependency between non-key attributes.

The same rule applies to satisfy the constraint: Remove the Offending Functional Dependency.

The tables constructed for the financial system of NUPI which are

presented by their relational notation in previous section are examined using the above rules and converted into higher normal forms as follows.

Revenue Relation:

REVENUE

REVENUE RECEIPT #	RECEIPT DATE	PAYER	BUDGET			CHECK #	REGIS-TERED DATE	CASHIER	PAYER CODE
			CODE	TITLE	AMOUNT				
1	1/5/95	MOF	8101, 8302, 8303.	survey, manpower & allow-ance.	1000.00, 2000.00, 5000.00.	-	1/6/95	Hiwot	001
2	2/5/95	Statio n-ary	5400	Sell of bid document	50.00	00250	2/6/95	"	002
3	2/5/95	"	5400	Sell of bid document	50.00	-	2/6/95	"	002

FIG 4.1 Revenue relation - un normalised form (UNF)

The above relation has repeating values in Code, Title and Amount columns. Therefore it is in non-normal form and it should be converted into first normal form by duplicating the values of the other columns. The resulting table is shown in fig 4.2.

REVENUE

REVENUE RECEIPT #	RECEIPT DATE	RECEIVED FROM	BUDGET CODE	DESCRIP TION	AMOUNT	CHECK #	REGISTE RED DATE	CASHIER	PAYER CODE
1	1/5/95	MOF	8101	Survey	1000.00	-	1/6/95	Hiwot	001
1	1/5/95	MOF	8302	Manpower	2000.00	-	1/6/95	"	001
1	1/5/95	MOF	8303	Allowan-ce	5000.00	-	1/6/95	"	001
2	2/5/95	Station-ary	5400	Sell of bid document	50.00	00250	2/6/95	"	002
3	2/5/95	Station-ary	5400	"	50.00	-	2/6/95	"	002

FIG 4.2 Revenue relation - first normal form (1NF)

Relation key {Revenue receipt #, Budget code}

This table is in its first normal form as the repeating values are removed. Therefore each column and row contain simple values. Relations in 1NF can still contain redundancy. The above Revenue relation has redundant values. For a given revenue receipt, attributes Receipt Date, Received From, Check #, Registered Date, Cashier and Payer Code are stored more than once. In revenue receipt # 1, all the listed attribute values are stored three times instead of once. The above stated rule: 'Remove the offending Functional Dependency' is applied to eliminate the redundancy.

The offending dependency becomes one table. Then all the functionally dependent attributes of the dependency are removed

from the original table. The remaining attributes form the original table. In the process, the revenue table is split ed into two tables. Fig 4.3 and fig 4.4 show the relations.

REVENUE

REVENUE RECEIPT #	RECEIPT DATE	PAYER	PAYER CODE	CHECK #	REGISTERED DATE	CASHIER
1	1/5/95	MOF	001	-	1/6/95	Hiwot
2	2/5/95	Stationary	002	00250	2/6/95	"
3	2/5/95	Stationary	002	-	2/6/95	"
4	2/5/95	MOF	001	-	2/6/95	"

FIG 4.3 Revenue relation - second normal form (2NF)

Relation key { Revenue receipt # }

REVENUE

REVENUE RECEIPT #	BUDGET CODE	DESCRIPTION	AMOUNT
1	8101	Survey	1000.00
1	8302	Manpower	2000.00
1	8303	Allowance	5000.00
2	5400	Sell of bid document	50.00
3	5400	Sell of bid document	50.00

FIG 4.4 Revenue relation - second and third normal form

Relation key {Revenue receipt #, Budget code}

A constraint which states that all non-key attributes must be functionally dependent on the whole of the relation key is satisfied by both tables. Therefore, they are in their 2NF. The tables are tasted by additional constraint to see whether they are in the 3NF or not. The Revenue relation on fig 4.4 is in its 3NF. There is no dependency between non-key attributes. The figure shows the relation it its 2nd as well as 3rd normal forms. But relation in fig 4.3 does not satisfy the additional constraint,

hence contains redundancy.For example MOF with and 001 on Payer name and Payer code columns respectively are stored twice. Therefore the usual rule is applied to the table. In the process, the relation is splitted into two tables namely Revenue 1 and Revenue 2. They are shown in figures 4.5 and 4.6

REVENUE 1

PAYER CODE	PAYER
001	MOF
002	Stationary

FIG 4.5 Revenue 1 relation - third normal form (3NF)

Relation key { Payer code }

REVENUE 2

REVENUE RECEIPT #	RECEIPT DATE	PAYER CODE	CHECK #	REGISTERED DATE	CASHIER
1	1/5/95	001	-	1/6/95	Hiwot
2	2/5/95	002	00250	2/6/95	"
3	2/5/95	002	-	2/6/95	"
4	2/5/95	001	-	2/6/95	"

FIG 4.6 Revenue 2 relation - third normal form (3NF)

Relation key { Revenue receipt # }

The above three revenue relations (Revenue, Revenue 1 & Revenue 2) are now in their third normal form.

Expenditure Relation: The relation and the identified attributes are shown in fig 4.7

EXPENDITURE

PAYMENT VOUCHER #	PAYMENT DATE	PAID TO	BUDGET CODE	AMOUNT	DESCRIP-TION	REGIST-ERED DATE	CHECK #	AUTHOR-ISED BY	CASHIER
009	3/5/95	Post off ice, ETHOF, Shell	6201, 6205, 8303.	100.00, 100.00, 300.00	Postal service, mainten-ance & grease	1/6/95	-	Admin & Finance	Hiwot
010	3/5/95	Tele.	6201	2900.75	Telex of march	2/6/95	00742	General Manager	"
011	4/5/95	Metu	8303	1500.00	Field allow-ance	2/6/95	-	General Manager	"

FIG 4.7 Expenditure relation - UNF

The table is in un normalized form because payment voucher 9 contains three values instead on one, in columns budget code, amount and description. Therefore it should be converted into 1NF.

After conversion, the resulting Expenditure relation in 1NF is shown in fig 4.8.

PAYMENT VOUCHER #	PAYMENT DATE	PAID TO	BUDGET CODE	AMOUNT	DESCRIP-TION	REGIST-ERED DATE	CHECK #	AUTHOR-ISED BY	CASHIER
009	3/5/95	Post office	6201	100.00	postal service	1/6/95	-	Admin. & Finance	Hiwot
009	3/5/95	ETHOF	6205	100.00	Maintenance	1/6/95	-	"	"
009	3/5/95	Shell	8303	300.00	Grease & oil	1/6/95	-	"	"
010	3/5/95	Tele	6201	2900.00	Telex of march	1/6/95	00742	General Manager	"
011	4/5/95	Awassa	8303	1500.00	Field allow-ance	1/6/95	-	"	"

FIG 4.8 Expenditure relation - 1NF

Relation key { Payment voucher #, Budget code }

The above relation contains redundancy because payment date, registered date, check #, authorized by, cashier, town and department are stored three times for payment voucher # 9 instead of once. To convert the relation to 2NF and remove the redundancy the table is splitted into two as shown in fig 4.9 and 4.10.

EXPENDITURE

PAYMENT VOUCHER #	PAYMENT DATE	REGISTERED DATE	CHECK #	AUTHORISED BY	CASHIER
009	3/5/95	1/6/95	-	Admin.& Finance	Hiwot
010	"	"	00742	General Manager	"
011	"	"	-	"	"

FIG 4.9 Expenditure relation - 2NF & 3NF

Relation key { Payment voucher # }

EXPENDITURE 1

PAYMENT VOUCHER #	BUDGET CODE	PAYEE	DESCRIPTION	AMOUNT
009	6201	Post office	Postal service	100.00
009	6205	ETHOF	Maintenance	100.00
009	8303	Shell	Grease	300.00
010	6201	Tele	Telex of march	2900.75
011	8303	Awassa	Field allowance	1500.00

FIG 4.10 Expenditure 1 relation - 2NF & 3NF

Relation key { Payment voucher #, Budget code }

The above two relations are in 2NF as there is no non-key attribute which is not functionally dependent on the whole of the respective relation keys. The tables are examined whether they satisfy additional constraint or not to remain in higher normal form. The restriction is there must be no dependency between non-key attributes. Since in both relations the constrain is satisfied, they are in the second as well as the third normal forms.

BUDGET

BUDGET						ACTUAL				RETURN-ED TO MOF	COMM-ENT
YEAR	CODE	TYPE	TITLE	PROPO-SED	APPROV-ED	REQUEST ED FOR SPEND-ING	RECEIV-ED FOR SPEND-ING	USED	BALANCE		
1994	8101	Capital	Survey	60,000	40,000	30,000	20,000	30,000	10,000	-	over use
	8102	Capital	Cartog-raphy	80,000	80,000	78,000	78,000	78,000	2,000	-	optim-um use
	8302	Capital	Manpow-er	80,000	40,000	30,000	30,000	18,000	22,000	12,000	under use
	6101	Recurr-ent	Salary	250,000	220,000	200,000	200,000	200,000	20,000	-	optim-um use
	6201	"	Utility	50,000	40,000	40,000	40,000	70,000	-30,000	-	over use
1995	8101	Capital	Survey	70,000	50,000	30,000	25,000	25,000	25,000	-	under use
	8102	Capital	Cartog-raphy	80,000	70,000	60,000	60,000	45,000	25,000	15,000	under use
	8302	Capital	Manpow-er	90,000	60,000	55,000	45,000	20,000	40,000	25,000	under use
	6101	Recurr-ent	Salary	260,000	220,000	210,000	210,000	210,000	10,000	-	optim-um use
	6201	"	Utility	55,000	45,000	40,000	40,000	75,000	-30,000	-	over use

FIG 4.11 Budget relation - UNF

The above relation is converted into first normal form by process explained for revenue and expenditure relationships. The resulting table is shown in fig 4.12.

BUDGET

BUDGET YEAR	BUDGET TYPE	BUDGET CODE	BUDGET TITLE	PROPOS-ED BUDGET	APPROV-ED BUDGET	REQUES-TED FOR SPEND-ING	RECEIV-ED FOR SPEND-ING	ACTUAL-LY SPEND	BALANCE	RETURN-ED TO MOF	COMM-ENT
1994	Capital	8101	Survey	60,000	40,000	30,000	20,000	30,000	10,000	-	Over use
1994	Capital	8102	Cartog-raphy	80,000	80,000	78,000	78,000	78,000	20,000	-	opti-mum use
1994	Capital	8302	Man-power	80,000	40,000	30,000	30,000	18,000	22,000	12,000	under use
1994	Recurr-ent	6101	Salary	250,000	220,000	200,000	200,000	200,000	200,000	-	Optim-um use
1994	"	6201	Utility	50,000	40,000	40,000	40,000	70,000	-30,000	-	Over use
1995	Capital	8101	Survey	70,000	50,000	30,000	25,000	25,000	25,000	-	Under use
1995	Capital	8102	Cartog-raphy	80,000	70,000	60,000	60,000	45,000	25,000	15,000	Under use
1995	Capital	8302	Man-power	90,000	60,000	55,000	55,000	20,000	40,000	25,000	Under use
1995	Recurr-ent	6101	Salary	260,000	220,000	210,000	210,000	210,000	10,000	-	Optim-um use
1995	"	6201	Utility	55,000	45,000	40,000	40,000	75,000	-30,000	-	Over use

FIG 4.12 Budget relation - 1NF

Relation key { Budget year, Budget code }

When the 1NF is examined, it contains undesirable feature. Therefore it should be converted into 2NF. As a result the table is splitted into two: Budget and Budget 1. Fig 4.13 and 4.14 show the relations.

BUDGET

BUDGET CODE	BUDGET TYPE	BUDGET TITLE
8101	Capital	Survey
8102		Cartog-
	"	raphy
8302		Man-
	"	power
6101		Salary
6201	"	Utility
	"	

FIG 4.13 Budget relation - 2NF & 3NF

Relation key { Budget code }

BUDGET 1

BUDGET CODE	BUDGET YEAR	PROPOSE BUDGET	APPROVED BUDGET	REQUEST-ED FOR SPENDING	RECEIVED FOR SPENDING	ACTUAL	BALANCE	RETURN-ED	COMMENT
8101	1994	60,000	40,000	30,000	20,000	30,000	10,000	-	Over use
8102	1994	80,000	80,000	78,000	78,000	78,000	20,000	-	Optim-um use
8302	1994	80,000	40,000	30,000	30,000	18,000	22,000	12,000	Under use
6101	1994	250,000	220,000	200,000	20,000	20,000	20,000	-	Optim-um use
6201	1994	50,000	40,000	40,000	40,000	70,000	-30,000	-	Over use
8101	1995	70,000	50,000	30,000	25,000	25,000	25,000	-	Under use
8102	1995	80,000	70,000	60,000	60,000	45,000	25,000	15,000	Under use
8302	1995	90,000	60,000	55,000	45,000	20,000	40,000	25,000	Under use
6101	1995	260,000	220,000	210,000	210,000	210,000	10,000	-	Optim-um use
6201	1995	55,000	45,000	40,000	40,000	75,000	-30,000	-	Over use

FIG 4.14 Relation budget 1 - 2NF

Relation key { Budget code, Budget year }

The tables are tested whether they satisfy additional constraint or not. They satisfied the condition. There is no functional dependency between the non-key attributes of those relations. Therefore the relations are in 3NF.

Sened Relation:

This relation is examined for redundancy and changed into its non-redundant form by applying the given rule and splitting the table as explained before. After the process of normalization is over, the relation is divided into two tables, sened and sened 1. Fig 4.15, 4.16, 4.17 and 4.18 show the situation.

SENED

Sened #	Payment Date	Paid to	Payee	Budget Code	Amount	Description	Check #	Author-ized by	Cashier	Town	Depar-tment
005	1/5/95	Kassahun T.	Tele.	6201	5000	Tele.- April	12520	G M	Hiwot	-	-
006	2/5/95	Bekele T.	Emp.	8303	1000	Allow & oper-	-	"	"	Metu	MPP
					4000	ational cost					
007	3/5/95	Kassahun T.	Shell	6205	6000	Fuel	13750	"	"	-	-
008	3/5/95	"	Tele.	6201	2900	Telex- March	13751	"	"	-	-
009	4/5/95	Tessema B.	Emp.	8303	2100	Allowance	-	"	"	Gore	POP

FIG 4.15 Relation sened - UNF

SENED

Sened#	Payment Date	Paid to	Payee	Budget Code	Amount	Description	Check #	Author-ized by	Cashier	Town	Depar-tment
005	1/5/95	Kassahun T.	Tele.	6201	5000	Tele of	12520	G M	Hiwot	-	-
006	2/5/95	Bekele T.	Emp.	8303	1000	April	-	G M	"	Metu	MPP
006	2/5/95	"	"	8303	4000	Allowance	-	G M	"	Metu	MPP
						Operational					
007	3/5/95	Kassahun T.	Shell	6205	6000	cost	13750	G M	"	-	-
008	3/5/95	Kassahun T.	Tele	6201	2900	Fuel	-	G M	"	-	-
009	4/5/95	Tesseama B.	Emp.	8303	2100	Telex of	-	G M	"	Gore	POP
						march					
						Allowance					

FIG 4.16 Relation sened - 1NF

Relation key { Sened #, Budget code, Description }

SENED

Sen-ed No	Payment Date	Paid to	Payee	Check #	Author-ized by	Cashier	Town	Depar-tment
005	1/5/95	Kassahun T.	Telecommunication	12520	G M	Hiwot	-	-
006	2/5/95	Bekele T.	Employee	-	G M	"	Metu	MPP
007	3/5/95	Kassahun T.	Shell	13750	G M	"	-	-

FIG 4.17 Relation sened - second and third normal form

Relation key { Sened # }

SENED I

SENED #	BUDGET CODE	DESCRIPTION	AMOUNT
005	6201	Telephone	5000
006	8303	Field allowance	1000
006	8303	Operational Cost	4000
007	6205	Fuel	6000
008	6201	Telex	2900
009	8303	Field allowance	2100

FIG 4.18 Relation sened I - second and third normal form

Relation key { Sened #, Budget code, Description }

Transfer relation:

Unlike the other relations, Transfer Relation contains no redundancy. The three constrains discussed above are satisfied by the relation. Therefore it is in 3NF. The relation is shown in Fig 4.18.

TRANSFER

Payment Voucher #	Sened #	Budget Code	Description	Transferred Amount	Amount Returned	Amount paid
010	008	6201	Telex	2900	-	-
011	009	8303	Allowance	1500	600	-

FIG 4.19 Relation transfer - First, second, third normal form

Relation key { Payment voucher #, Sened #, Budget code, Description }

Data about data of the finance system is documented in a data dictionary shown in fig 4.20. It is used to keep track of the data.

NATIONAL URBAN PLANNING INSTITUTE

FINANCE DIVISION

DATA DICTIONARY

DATA ELEMENT IDENTIFIER	DATA ELEMENT NAME	DESCRIPTION	DATA TYPE	DATA LENGTH	INDEX
RECEIPT_NO	Revenue receipt number	A successive number given to a standard receipt prepared by MOF, used in finance division for collecting any cash from any source.	Numeric	6	Y
BUDGT_CODE	Budget code	A code given to capital & recurrent budgets.	Numeric	4	Y
DESCRIPT	Description	Description of a collected revenue or a payment.	Character	50	N
AMOUNT	Amount	The amount of a collected revenue or payment.	Numeric	12	N
PAYER_CODE	Payer code	A code given to one who pays cash for any purpose.	Numeric	3	Y
PAYER_NAME	Payer name	The name of an individual / organisation etc who makes cash payment either for NUPI or to MOF through NUPI	Numeric	30	N
RECPT_DATE	Revenue receipt date	The date on which the revenue is collected or the date on revenue receipt.	Date	8	N
CHECK_NO	Check number	For collections that are made on check, the check number written on revenue receipt or for payments that are made on check, the check number written on payment voucher. This check number is the number on a check issued to NUPI. NUPI has only one Government check with code 65-60-00546.8	Character	30	N

NATIONAL URBAN PLANNING INSTITUTE

FINANCE DIVISION

DATA DICTIONARY

DATA ELEMENT IDENTIFIER	DATA ELEMENT NAME	DESCRIPTION	DATA TYPE	DATA LENGTH	IND-EX
RECEIPT_NO	Revenue receipt number	A successive number given to a standard receipt prepared by MOF, used in finance division for collecting any cash from any source.	Numeric	6	Y
BUDGT_CODE	Budget code	A code given to capital & recurrent budgets.	Numeric	4	Y
DESCRIPT	Description	Description of a collected revenue or a payment.	Character	50	N
AMOUNT	Amount	The amount of a collected revenue or payment.	Numeric	12	N
PAYER_CODE	Payer code	A code given to one who pays cash for any purpose.	Numeric	3	Y
PAYER_NAME	Payer name	The name of an individual / organisation etc who makes cash payment either for NUPI or to MOF through NUPI	Numeric	30	N
RECPT_DATE	Revenue receipt date	The date on which the revenue is collected or the date on revenue receipt.	Date	8	N
CHECK_NO	Check number	For collections that are made on check, the check number written on revenue receipt or for payments that are made on check, the check number written on payment voucher. This check number is the number on a check issued to NUPI. NUPI has only one Government check with code 65-60-00546.8	Character	30	N

Cont.

DATE_REGST	Registered date	The date the revenue receipt is recorded in a register / computer.	Date	8	N
CASHIER	Cashier	The cashier who collects revenue and who makes payment.	Character	15	N
PV_NO	Payment voucher number	A sequential number given to payment vouchers on expenditures. The voucher is a standard form attached with all payments.	Numeric	3	Y
PAYMNT DATE	Payment date	The date on which the payment is made.	Date	8	N
DATE_REGIS	Registered date	The date the payment voucher is recorded in a register / computer.	Date	8	N
PAID_TO	Paid to	A person on whose name expenditure or sened is prepared. A name on the payment voucher.	Character	30	N
AUTHRZD_BY	Authorized by	The mane of a person who authorizes a payment.	Character	20	N
TOWN	Town	A town for which a certain project such as master plan, development plan, preparation is being carried out by the institute.	Character	15	N
DEPARTMENT	Department	A department for which budget is allocated by management.	Character	20	N
PAYEE	Payee	Name of a person / organisation etc for whom / which payment is made.	Character	40	N

Cont.

BUDGT_TYPE	Budget type	The type of the budget such as capital, recurrent.	Character	15	N
BUDGT_ TITLE	Budget title	The description of a certain budget.	Character	40	N
BUDGT YEAR	Budget year	A time period from Hamle 1 - Sene 30 (Ethiopian calender)	Numeric	4	Y
PROPOSED_B	Proposed budget	A capital / recurrent budget proposal of a year given to MOF	Numeric	10	N
APPROVED_B	Approved budget	A capital / recurrent budget approved by MOF for a given year	Numeric	10	N
REQ_SPEND	Requested for spending	Total disbursement requests made to MOF in a given year	Numeric	10	N
REC_SPEND	Received for spending	From disbursement requests of a year, the approved amount	Numeric	10	N
ACTUAL_BUD	Actual budget	The actual amount utilized in a budget year.	Numeric	10	N
COMMENT	Comment	A comment on utilization of budget.	Numeric	10	N
SENED_NO	Sened number	A sequential number given to payment vouchers on seneds.	Numeric	3	Y
TRANS AMNT	Transferred amount	The amount transferred from sened to expenditure.	Numeric	10	N
RETND_AMNT	Amount returned	The excess amount of sened over expenditure.	Numeric	10	N
PAID_AMNT	Amount paid	The excess amount of expenditure over sened.	Numeric	10	N

FIG 4.20 Data dictionary for the computer-assisted financial information system of NUPI.

CHAPTER FIVE

DEVELOPMENT OF THE PROTOTYPE SYSTEM

This section is devoted to demonstrate how the computer-assisted financial information system (the proposed system) solves most of the major problems of the existing system. It is relevant to note, however, that this account is only presented as the documentation of what has been done along this lines, and should not in any way considered a simulation of the proposed system. Source program listing is attached. Before the description of the system, it is also considered in order to briefly state the main strategy of demonstrating the prototype-preference of the making of the software than buying.

5.1 Purchasing off the shelf packages versus in-house development

Meeting user requirements and optimum utilization of available resources are the two major factors taken into consideration for selecting from the MAKE or BUY options. Off the shelf systems are usually more tested, less costly and their implementation requires less time. But they are not tailored to requirements as in-house developed systems are. It is believed that users of the financial system of NUPI are satisfied more by the output of developed systems than purchased once. One of the reasons is that little advancement is done in non-profit single entry governmental accounting system in Ethiopia. As a result the existing packages may not fit for the finance system as they

are. Most of the off the shelf packages such as DACEASY, QUICKBOOKS, PROFIT, Peachtree Accounting are designed for double entry profit making businesses and may not meet user requirements as they should be.

Also it is said that there are system analysts, programmers and information professionals in the institute for the development work. In-house development leads to optimum utilization of institutional resources. In addition, it is expected that development cost will not be expensive because of the size and simplicity of the newly developed system. Therefore, from opinion of users especially system people and my judgment as described above, at this stage, developing in house package using internally available manpower and other resources is the recommended option.

5.2 DATABASES

The prototype system uses dBASE IV. The relations/tables discussed in chapter 4 are directly converted into database files. Each relation become one database file. To convert the conceptual data model developed in chapter 4 into database definition that is supported by a database management system (DBMS) two major steps are required.

These are:

•conversion of the relational model to a set of record types (logical record structure) with each record made up of a number of fields. (This step is independent of the DBMS used);

•conversion of the record types to database definition (a DBMS dependent step).

Database management systems store data in record types. They support link structure between record types. Generally, there are three kinds of DBMSs. These are;

•Relational DBMS (supports relational model);
•Network DBMS (supports network structures) ; and
•Hierarchical DBMS (supports hierarchical structures).

In chapter four, the relational data model is developed. Now the attempt is to convert it into database definition. dBASE IV supports relational modes. Whenever this is the case, there is no need to construct a logical structure. Simply the relational model becomes the dBASE IV data structure. The tables become the relations of dBASE IV. The name of the relation becomes a database file name. The mane of the attributes / columns become field name. The files are listed in fig 5.1

SERIAL NUMBER	FILE NAME	FIELD NAME
1	REVENUE	RECEIPT_NO, BUDGT_CODE, DESCRIPT, AMOUNT
2	REVENUE 1	PAYER_CODE, PAYER_NAME
3	REVENUE 2	RECEIPT_NO, RECPT_DATE, PAYER_CODE, CHECK_NO, DATE_REGST, CASHIER
4	EXPEND	PV_NO, PAYMNTDATE, DATE_REGIS, PAID_TO, CHECK_NO, AUTHRZD_BY, CASHIER, TOWN, DEPARTMENT
5	EXPEND 1	PV_NO, BUDGT_CODE, PAYEE, DESCRIPT, AMOUNT
6	SENED	SENED_NO, PYT_DATE, PAID_TO, PAYEE, CHECK_NO, AUTHOR_BY, CASHIER, TOWN, DEPT.
7	SENED 1	SENED_NO, BUDGT_CODE, DESCRPTION, AMOUNT
8	BUDGET	BUDGT_CODE, BUDGT_TYPE, BUDGT_TITL
9	BUDGET 1	BUDGT_CODE, BUDGT_YEAR, PROPOSED_B, APPROVED_B, REQ_SPEND, REC_SPEND, ACTUAL_BUD, COMMENT
10	TRANSFER	PV_NO, SENED_NO, BUDGT_CODE, DESCRIPT, TRANS_AMNT, RETND_AMNT, PAID_AMNT.

FIG. 5.1 DATABASE FILES OF THE PROTOTYPE SYSTEM

5.3 PROGRAM OF THE PROTOTYPE SYSTEM

A program is written using dBASE IV application facility. The following functions are performed by the program.

The first screen is a welcome screen. Then a main menu appears. The menu has the following five options.

- Revenue / Expenditure system;
- Sened system;
- Payroll system;
- Inventory system;
- Exit.

The user is prompted to select one of the options by pressing Enter key. If he / she wants to terminate at this stage, Exit quits the program. Revenue / Expenditure system is the main system. In fact it is a complete system. The other systems are subsidiary to this system. When this option is selected, a pop-up menu appears. The pop-up menu has three options namely Revenue, Expenditure and Budget. Input to this system is from three sources. Four data entry screens are designed one for revenue, one for expenditure and two for budget transactions. Once data is inputted into the system, it stores the information into ten databases files listed above. Revenue transaction is stored into Revenue, Revenue 1 and Revenue 2 database files. Expenditures transaction is stored into Expend and Expend 1 database files. Budget transaction is stored into Budget and Budget 1 database files. Then whenever required the system outputs various reports. The output is in two forms. The user may view on screen any information from the respective databases or prints on paper using a print option.The first screen is shown below in fig 5.2.

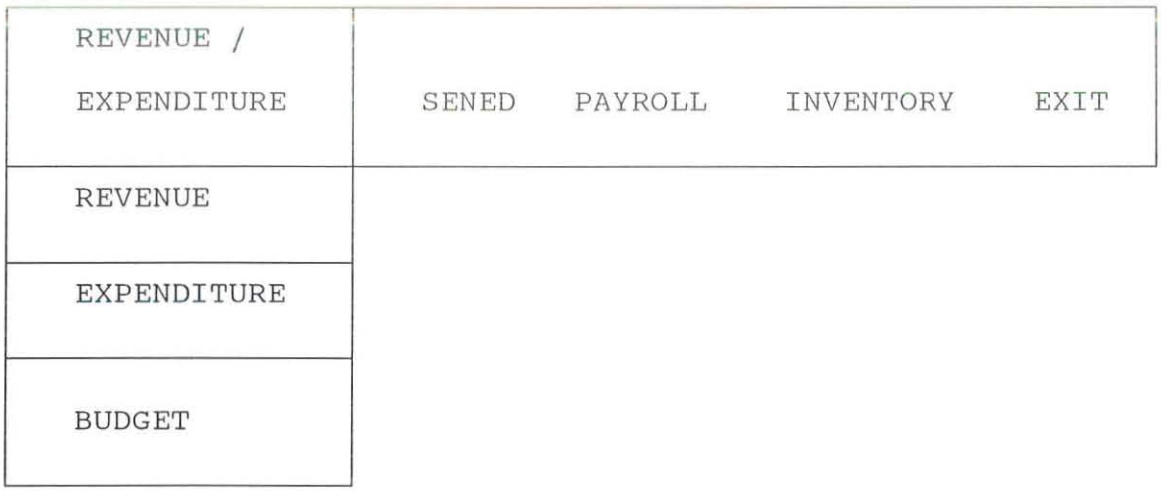


FIG 5.2 Main screen of the prototype system.

When one of Revenue, Expenditure or Budget options is selected, another pop-up menus, one for each option appears on the screen. The menus have four choices : Add, Edit, View, and Print. They are discussed as follows:

Add: This option adds revenue, expenditure or budget information into the system. One of the four data entry screens appears when selection is made. Then a user is prompted to add transactions. After entering each transaction, the system requests for additional transaction. If no, it requests for additional revenue receipt, payment voucher or budget information. If there is one, then the screen repeats otherwise the system returns into the previous screen. At this stage the user may view any information from the respective database files using the standard dBASE IV facility. The input screens are shown in fig 5.3, 5.4 and 5.5, 5.6.

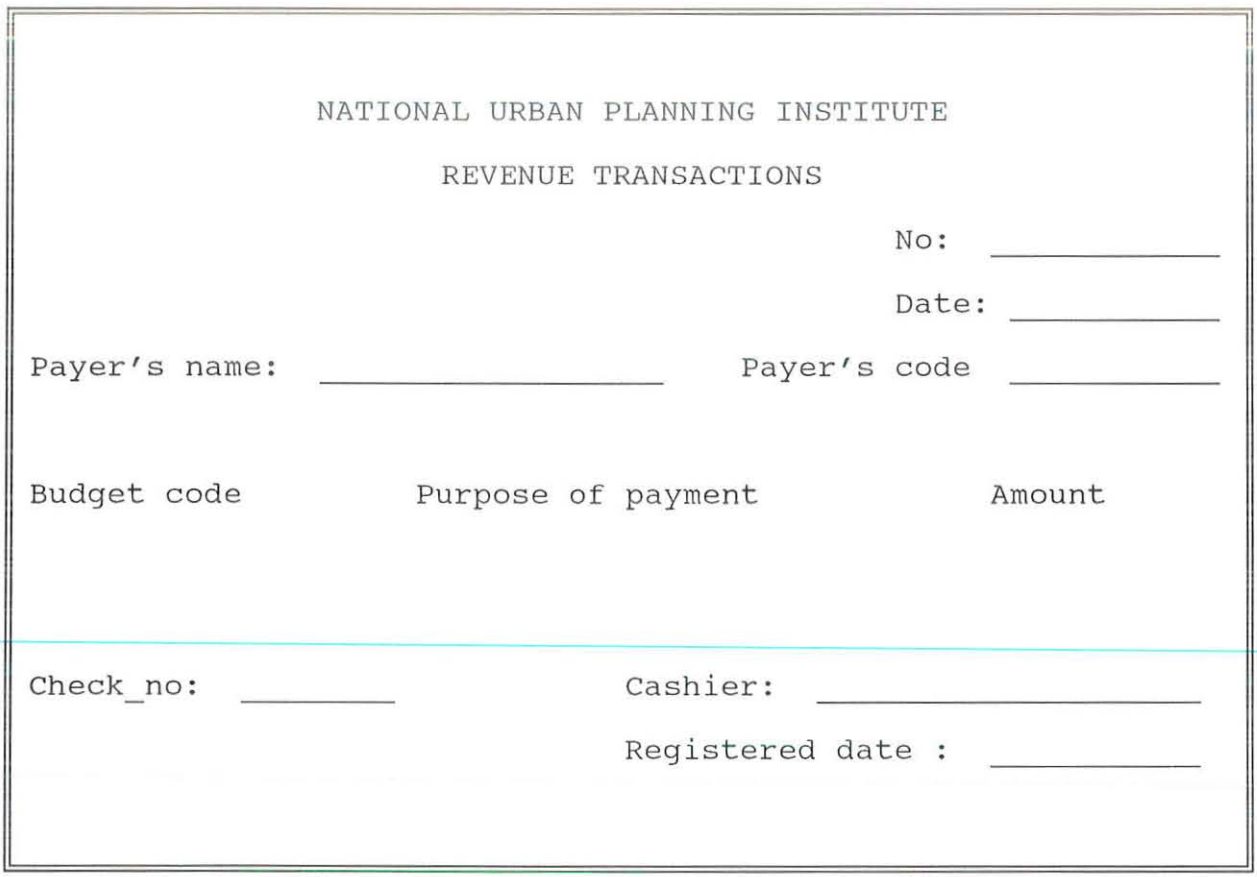


FIG 5.3 Data entry screen for Revenue transactions

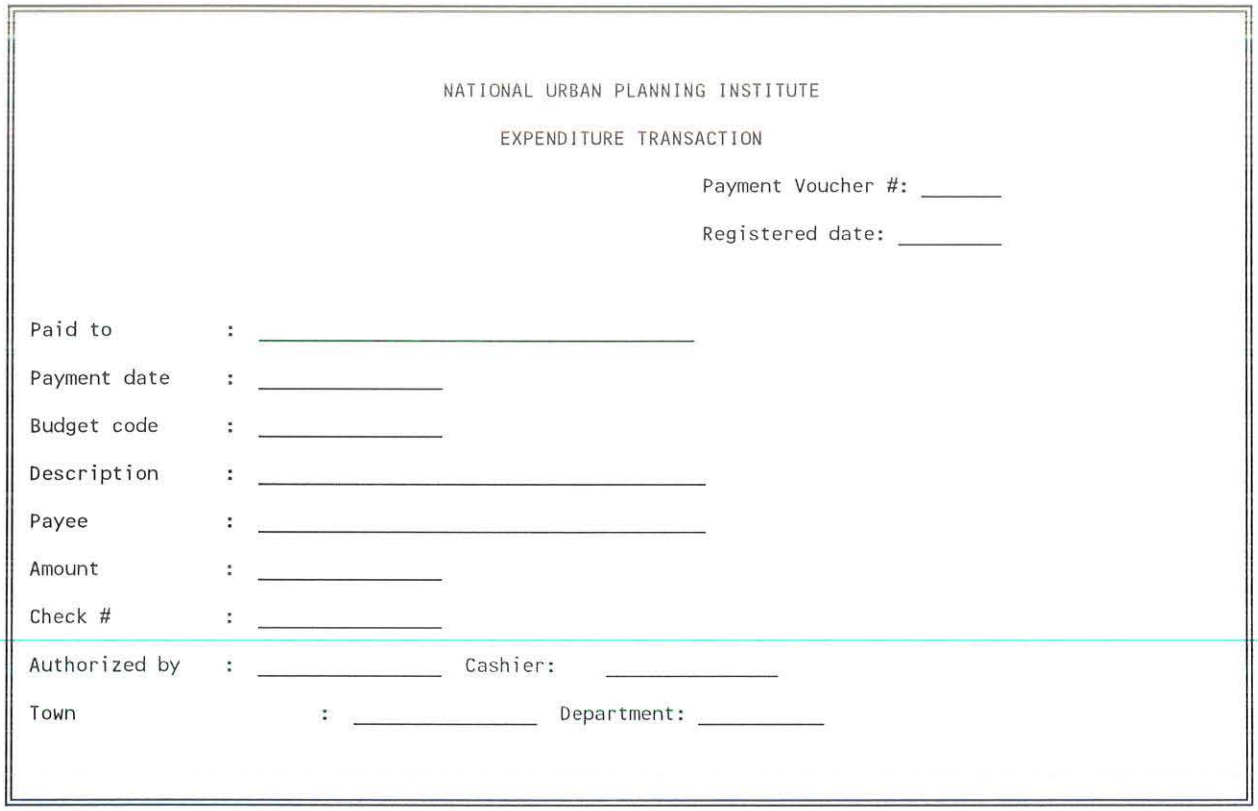


FIG 5.4 :Data entry screen for expenditure transactions.

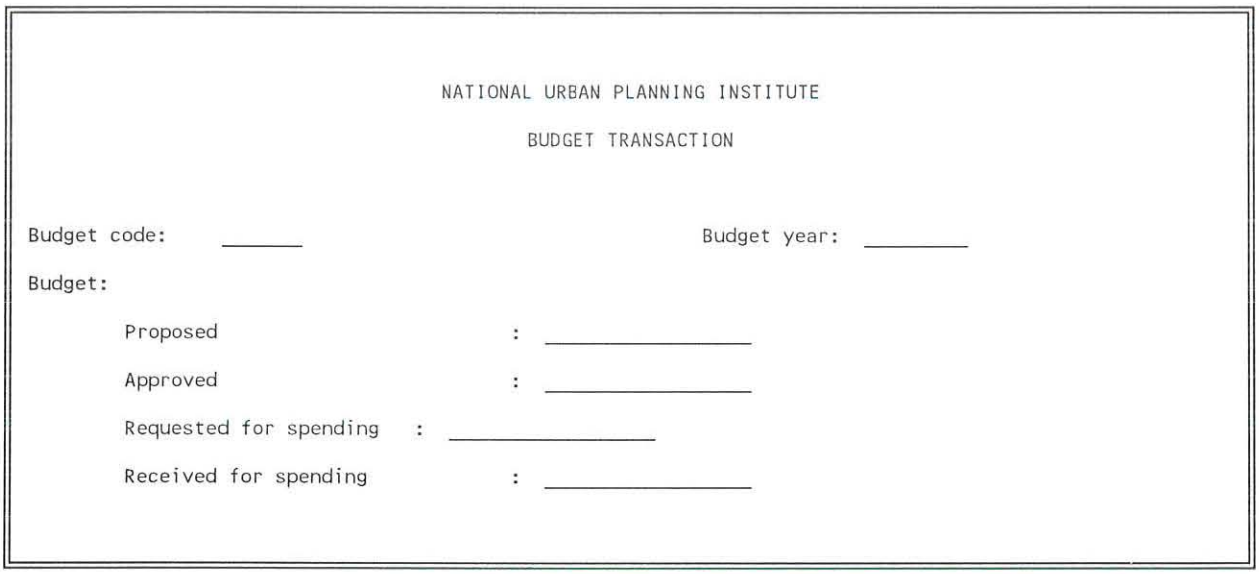


FIG 5.5 Data entry screen for Budget Transaction

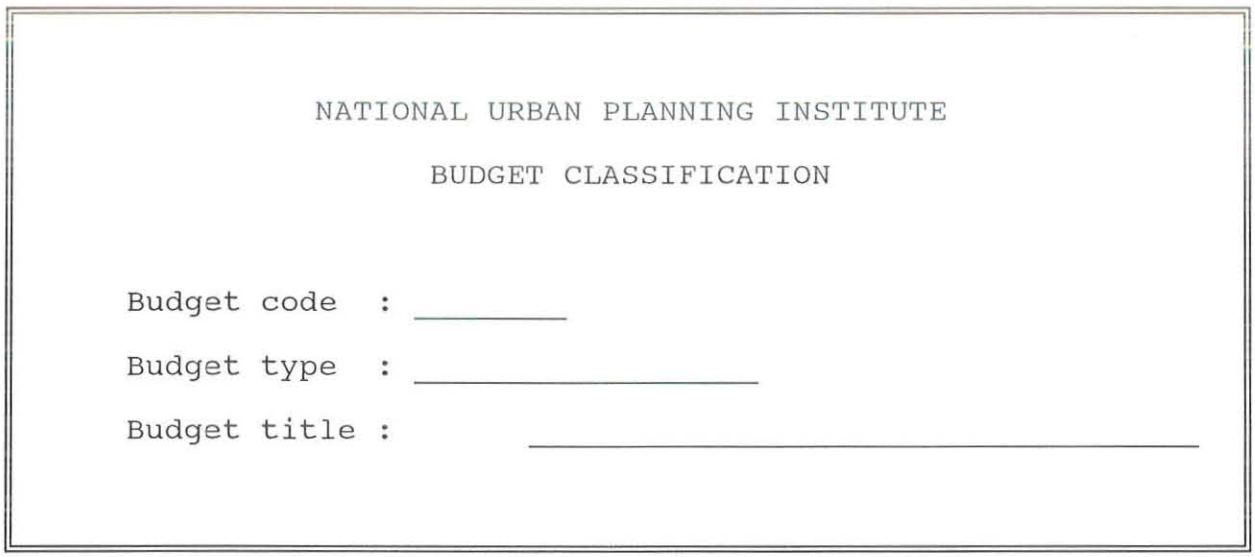


FIG 5.6 Data entry form for Budget Classification.

The budget classification screen is used as chart of account, just to list the budget codes and the corresponding budget titles.

Edit: This option is selected when changing information on the databases is required. Three edit screens are designed, one for each option. To retrieve a record which needs editing, a user is prompted to enter record identifiers. Then the system retrieves the record in an edit screen. After editing the information, the user is requested whether he / she wants to edit again. If no, the system returns back to the options screen. If the user enters record identifier that does not exist into the databases, the system makes search and displays a message like this : " The corresponding record is not found".

If the user wants to display and edit all records, the standard dBASE IV edit mode can be used.

View: When this option is selected, the system displays all records combining all fields the respective databases. It is possible to obtain detail information about each transaction as well as the activities over a certain period of time. A user can not make changes on records from this screen. Changes are made only from edit mode. When a user is through, he/she returns to the options by pressing Esc. key.

Print: This option produces output from the system. For revenue, all revenue transactions of a month is printed as shown in fig 5.7.

Expenditure transactions have one output format. The advantages of such reports is that user could find information on revenue only if he / she requires without being confused by too much information.

The output format for expenditures is shown in fig 5.8. If expenditures of the institute is required over certain period of time, this information is obtained from the report.

NATIONAL URBAN PLANNING INSTITUTE

REVENUE TRANSACTIONS

FOR THE MONTH ENDED MAY 31, 1995

RECEIPT DATE	RECEIPT NUMBER	BUDGET CODE	DESCRIPTION	AMOUNT	
1/5/95	1	8101	Adhoc survey		1000.00
1/5/95	1	8302	Manpower		2000.00
1/5/95	1	8303	Field allowance	5000.00	
2/5/95	2	5400	Sell of bid document		50.00
3/5/95	3	5400	Sell of bid document		60.00
			TOTAL		8110.00

FIG 5.7 Report format for Revenue transactions

NATIONAL URBAN PLANNING INSTITUTE

EXPENDITURE TRANSACTIONS REPORT

FOR THE MONTH ENDED MAY 31, 1995

PAYMENT DATE	PAYMENT VOUCHER #	DESCRIPTION	AMOUNT	
3/5/95	9	Postal services	100.00	
3/5/95	9	Maintenance		100.00
3/5/95	9	Grease and oil		300.00
3/5/93	10	Telex of march		2900.00
4/5/95	11	Field allowance	1500.00	
		TOTAL		4900.00

FIG 5.8 Report format for Expenditure transactions .

The system produces four types of reports namely, Revenue-Expenditure Report, Revenue-Expenditure Report by Budget Code, Revenue-Expenditure Summary Report and Budget Report. The reports are found when print is selected under Budget option. The report formats are shown in figures 5.9, 5.10, 5.11 and 5.12.

NATIONAL URBAN PLANNING INSTITUTE

REVENUE-EXPENDITURE REPORT BY DATE

FOR THE MONTH OF MAY 31, 1995

DATE	RECEIPT/ PV NO	BUDGET CODE	DESCRIPTION	AMOUNT REVENUE	AMOUNT EXPENDITURE
1/5/95	1	8101	Adhoc survey	1000.00	
1/5/95	1	8302	Manpower	2000.00	
1/5/95	1	8303	Field allowance	5000.00	
2/5/95	2	5400	Sell of bid document	50.00	
3/5/95	3	5400	Sell of bid document	60.00	
3/5/95	9	6201	Postal services		100.00
3/5/95	9	6205	Maintenance		100.00
3/5/95	9	8303	Grease and oil		300.00
3/5/95	10	6201	Telex of March		2900.00
4/5/95	11	8303	Field allowance		1500.00
			TOTAL	8110.00	4900.00

FIG 5.9 Revenue - Expenditure Report by date

The above report sorts the information by date. The following table sorts the information by budget code.

NATIONAL URBAN PLANNING INSTITUTE

REVENUE - EXPENDITURE REPORT BY BUDGET CODE

FOR THE MONTH ENDED MAY 31, 1995

DATE	RECEIPT / PV NO	BUDGET CODE	DESCRIPTION	AMOUNT REVENUE	EXPENDITURE
2/5/95	2	5400	Sell of bid document	50.00	
2/5/95	3	5400	Sell of bid document	60.00	
3/5/95	9	6201	Postal services		100.00
3/5/95	10	6201	Telex of March		2900.00
3/5/95	9	6205	Maintenance		100.00
1/5/95	1	8101	Adhoc survey	1000.00	
1/5/95	1	8302	Manpower	2000.00	
1/5/95	1	8303	Field allowance	5000.00	
3/5/95	9	8303	Grease and oil		300.00
3/5/95	11	8303	Field allowance		1500.00
			TOTAL	8110.00	4900.00

FIG 5.10 Revenue-expenditure report sorted by date

NATIONAL URBAN PLANNING INSTITUTE

REVENUE - EXPENDITURE SUMMARY REPORT

FOR THE MONTH ENDED MAY 31, 1995

DATE	RECEIPT/ PV NO	BUDGET CODE	DESCRIPTION	REVENUE	EXPENDITURE	BALANCE
2/5/95	2	5400	Sell of bid document	50.00		50.00
2/5/95	3	5400	Sell of bid document	60.00		110.00
3/5/95	9	6201	Postal services		100.00	-100.00
3/5/95	10	6201	Telex of March		2900.00	-3000.00
3/5/95	9					
		6205	Maintenance		100.00	-100.00
1/5/95	1	8101	Adhoc survey	1000.00		1000.00
1/5/95	1	8302	Manpower	2000.00		2000.00
1/5/95	1	8303	Field allowance	5000.00		5000.00
3/5/95	9	8303	Grease and oil		300.00	4700.00
3/5/95	11	8303	Field allowance		1500.00	3200.00
			TOTAL	8110.00	4900.00	3210.00

FIG 5.11 Revenue - Expenditure Summary Report.

NATIONAL URBAN PLANNING INSTITUTE

BUDGET REPORT

FOR THE MONTH ENDED MAY 31,1995

BUDGET CODE	BUDGET TITLE	APPROVED BUDGET FOR THE YEAR	EXPECTED EXPENDITURE FOR THE MONTH	ACTUAL EXPENDITURE FOR THE MONTH	VARIANCE	BALANCE
6201	Utility	45000.00	5000.00	3000.00	2000.00	42000.00
6205	Maintenance	60000.00	5000.00	100.00	4900.00	59900.00
8101	Adhoc survey	50000.00	4000.00	1000.00	3000.00	49000.00
8302	Manpower	60000.00	12000.00	2000.00	10000.00	58000.00
8303	Operational cost	100000.00	8000.00	1800.00	6200.00	98200.00
	TOTAL	15000.00	34000.00	7900.00	26100.00	30710.00

FIG 5.12 Budget Report

Sened system: In addition to Revenue-Expenditure system, a sened system is developed . Any information related to sened is obtained from the system. When this option is selected, a pop-up menu with Add, Delete, Edit, View and Print option appears. A data entry screen is designed for sened transactions similar to expenditure transactions data entry form with minor difference. The data is stored in Sened, Sened1 and Transfer database files. From these files imformation could be retrieved using various facilities of dBASE IV. When a reord is transferred from sened to expenditure, it is cancelled from sened file using delete option. The option requests the user for confirmation whether that record should be deleted or not. Finally the system produces output as shown in fig 5.10

NATIONAL URBAN PLANNING INSTITUTENATI

SENED STATEMENT

FOR THE MONTH ENDED MAY 31, 1995

SENED #	PAYMENT DATE	PAID TO	PAYEE	CHECK #	DESCRIPTION	ANOUNT
005	1/5/95	ATO KASSAHUN TESFAYE	TELECOMMUNICATION	12520	TELEPHONE OF APRIL	5000
006	2/5/95	ATO BEKELE TESFAYE	EMPLOYEE	-	FIELD ALLOWANCE	1000
006	2/5/95	ATO BEKELE TESFAYE	EMPLOYEE	-	OPERATIONAL COST	4000
007	3/5/95	ATO KASSAHUN TESFAYE	SHELL COMPANY	13750	FUEL	6000
					TOTAL	16000

FIG 5.13 Sened statement

CHAPTER SIX

CONCLUSION AND RECOMMENDATION

6.1 CONCLUSION

- A study to design a computer - assisted financial information support system for NUPI is made with the objective to facilitate provision of right financial information for the right user at the right time for the purpose of making timely and sound decisions.

- The finance division of the institute is given the responsibility to timely collect, record, process financial activities and produce necessary reports. But the system is not carrying out these activities as required by users. To identify why it is not meeting its requirements and to find out a solution to its problems, this study is conducted.

- Users are the main information sources for the thesis work. Enough facts for carrying out this study were gathered from them through interview. Document review and observation methods are also employed. Emphasis is given to users of financial information system of the institute through out the study mainly because the new system is designed for tneir use and its output is used for satisfying their financial information need. The study focuses on them with the

hope of exploiting their knowledge about the system and to come up to solution for the system problems with them. They know better the strength and weaknesses of the current system and hence could propose practical solutions to problems. A structured system analysis and design tools and techniques were used heavily to look into what the existing system does and how it does. They are used to define what and how the new system will do. They were very useful for arriving at the proposed computer assisted financial information support system of NUPI.

- Effort is made to understand the existing financial system. In the process of carrying out this activity main requirement of the finance system and user dissatisfaction area are identified. The existing system is documented using written description as well as graphical and non-graphical documentation tools.

- Assessment is made to the existing system. The major problems of the current financial system include untimeliness in performing most of the functions, poor quality of financial reports and difficulty to retrieve information. Factors that contribute to inability of the system are manual methods, lack of properly organised accounting system, use single-entry accounting systems and shortage of manpower. After problems of current financial information system of NUPI are identified, taking these problems

as a basis, objectives that should be met by the new system are set.As there are many ways of meeting system objectives, two design alternatives are considered. This is to provide users with design choices from which to select one that suits them best. From the investigated alternatives, computerizing some processes, by developing an in house system is proposed. The automation boundary of the processes or domain of change is defined. It is believed that the proposed solution will overcome the existing problems.

- Database design is among the major areas of systems design activity and is included in this study. Data analysis is made. The entities and relationships of the system are identified. Then a conceptual data model is developed. In the process, normalization technique is used to reach at a non-redundant data for the financial information support system of NUPI. Also a data dictionary is compiled.

- Prototype system is developed to demonstrate that the newly designed computer-assisted system could solve the identified problems.

From the out come of this study it could safely be concluded that:

- Single-entry government accounting systems by their very nature, does not provide various access points to records and hence can not satisfy the continually

changing financial information needs of users.

- Using computers for single-entry government accounting systems increases flexibility to the system hence decreases retrieval problem. Also delay in carrying out activities is minimized as a result of using the technology.

- The proposed solution could solve existing system problems by reducing time required to perform an activity, by allowing generation of various periodical financial reports, by increasing accuracy to the information reported, by allowing better information retrieval through many access points such as time period, budget code and by making possible maintainance of complete records to transactions.

- Finally it should be noted that this study aims at identifying the major components of the automated system and demonstrating how the proposed solution could solve the system problems. Further work is expected to make the system operational.

6.2 RECOMMENDATIONS

The following recommendations are made.

- The proposed system should run in parallel with the existing single-entry accounting system because it

is required for reporting and auditing purposes by MOF and General Auditors.

- It is recommended that the proposed solution could be applied to all government offices that are totally financed by central treasury and are using a single-entry government based accounting system.

- Training for users is required if the solution is to be implemented. Users should be trained on how to use computers in general and how to interact with the new financial information support system in particular.

- Management should participate in organizing resources required to proceed towards detail investigation, development and testing of an automated system. Also, even if from fact finding process it is found out that management is willing to commit computer, manpower as well as financial resources to improve the finance system, effort should be made by concerned parties to keep key people aware of the difference such systems would bring on the effectiveness and efficiency of the institute in carrying out its duties and responsibilities so that this activity would become among the priorities of the institute.

- Additional key personnel are required for the Division even if the new system is going to be implemented.

- As it is said earlier detail work on Revenue-Expenditure and Sened system is expected in the future. But the Payroll and Inventory systems should be developed from the scratch.

- From the study it is found out that the existing system is insensitive to users needs. The financial information system of NUPI should be studied and revised periodically to enable it to satisfy users information need by providing right information for the right user at the right time for the purpose of making various decisions.

- Since user participation is an important aspect of systems development, participation of users of the financial system of NUPI through out the development work of the new system is highly recommended.

BIBLIOGRAPHY

Adams, David. R., Wagner, Gerald E and Boyer, Terrence J. 1983. Computer Information Systems: An Introduction. Ohio: South-western publishing co.

Aktas, A. Ziya. 1987. Structured Analysis and Design of Information Systems. New Jersey: Prentice-Hall International.

Atherton, Pauline. Handbook of Information Systems and Services. 1987. Paris: UNESCO.

Brightman, Richard W. and Dimsdale, Jeffery M. 1986. Using Computers in an Information Age. New York: Delmar Publishers Inc.

Burn, Janice and O'Neil, Mike. 1987. Information Analysis. London: Paradigm Publishing Limited.

Brathwaite, Kens. 1988. Analysis, Design, and Implementation of Data dictionaries. New York: McGraw-Hill.

Don, Yeates and Daniels, Alan. 1990. Basic Systems Analysis. 3rd ed. Singapore: Pitman Publishing.

Davis, Gordon B. and Olson, Margrethe H. 1985. Management Information Systems: Conceptual foundations, structure and development. 2nd ed. Singapore: McGraw-Hill.

Ethiopia. Negarit Gazata. No. 15 Addis Ababa, 13th June 1987.

Gore, Marvin R. and Stubbe, John W. 1984. Computers and Information Systems. New York: McGraw-Hill.

Grudnitski, Gary and Burch, John. 1986. Information systems, Theory and Practice. 4th ed. New York: John Wiley & Sons.

Hawe, D. R. 1991. Data Analysis for Database Design. 2nd ed. London: Chapman and Hall.

Hadera, Abera. 1995. Computerized Information Support System for the Immigration and Consular Department of Ethiopia. Masters Thesis. Addis Ababa University. Addis Ababa.

Hey, Leon E. 1989. Accounting for Governmental and Nonprofit entitites. 8th ed. : IRWIN.

Hawryszkiewycz, I.T. 1991. Introduction to Systems Analysis and Design. 2nd ed. New York: Prentice Hall.

Laurie, Peter. 1985. Databases: How to Manage Information on Your Micro. London: Chapman and Hall.

Lucas, JR., Henry C. 1981. The Analysis, Design and Implementation of Information systems. New York; McGraw-Hill.

Smith, Jack L., Keith, Robert M. and Stephens, William L. 1986. Accounting Principles. New York; McGraw-Hill.

Tessema, Geda. 1995. Accounting Information Support System with particular reference to Agency for the Administration of Rented Houses. Master's Thesis. Addis Ababa University. Addis Ababa.

Winsten, Irwin. 1990. Accounting Software and the Microcomputer. A practical guide to evaluation and implementation. New York: John Wiley & Sons.

Webster's New Twentieth Century Dictionary Unabridged. 2nd ed. 1979. New York: Simon and Schuster.

APPENDIX B

REVENUE RECEIPT

TRANSITIONAL GOVERNMENT OF ETHIOPIA

CENTRAL GOVERNMENT

Serial REVENUE RECEIPT No: _____

_____

_____ Department

Payer's name: _____ Payer's code _____

Budget code Purpose of payment Amount

Check_no: _____ Cashier: _____

Registered date : _____

PAYMENT VOUCHER

TRANSITIONAL GOVERNMENT OF ETHIOPIA

CENTRAL GOVERNMENT

Payment Voucher #: _____

Paid to : _____ Registered date : _____

Payment date : _____

Budget code : _____

Description _____

_____

_____

_____

Amount in figures : _____

In words : _____

_____

Check # : _____

Authorized by : _____ Cashier: _____

APPENDIX C

SOURCE CODE OF NUPIFIN PROGRAM

```
*-------------------------------------------------------
* Program.....: NUPIFIN.PRG
* Author.....: This is an APPLICATION OBJECT.
* Date.....: 4-25-96
* Notice.....: Type information here or greetings to your
users.
* Generated by.: dBASE IV version 2.0
* Description..: Main routine for menu system
*-------------------------------------------------------

  *-- Setup environment
  SET CONSOLE OFF
  IF TYPE("gn_ApGen")="U"
    CLEAR WINDOWS
    CLEAR ALL
    CLOSE ALL
    CLOSE PROCEDURE
    gn_ApGen=1
  ELSE
    gn_ApGen=gn_ApGen+1
    IF gn_ApGen > 4
      Do Pause WITH "Maximum level of Application nesting
exceeded."
      RETURN
```

```
    ENDIF
    PRIVATE gn_oldsize
    gn_oldsize=gn_scrsize
    PRIVATE gc_bell, gc_carry, gc_clock, gc_century, gc_confirm,
gc_deli,;
            gc_safety, gc_status, gc_score, gc_talk, gl_leave,
gc_prognum,;
            gc_quit, gc_color, gc_display, gl_color, gl_batch,
gn_scrsize
  ENDIF
  *-- Store some sets to variables
  gc_bell   =SET("BELL")
  gc_carry  =SET("CARRY")
  gc_clock  =SET("CLOCK")
  gc_color  =SET("ATTRIBUTE")
  gc_century=SET("CENTURY")
  gc_confirm=SET("CONFIRM")
  gc_cursor =SET("CURSOR")
  gc_deli   =SET("DELIMITERS")
  gc_display=SET("DISPLAY")
  gc_safety =SET("SAFETY")
  gc_status =SET("STATUS")
  gc_score  =SET("SCOREBOARD")
  gc_talk   =SET("TALK")

  SET CONSOLE ON
  SET BELL ON
  SET CARRY OFF
  SET CENTURY OFF
  SET CLOCK OFF
```

```
  SET CONFIRM OFF
  SET DEFAULT TO C
  SET DELIMITERS TO ""
  SET DELIMITERS OFF
  SET DEVICE TO SCREEN
  SET ESCAPE ON
  SET EXCLUSIVE OFF
  SET LOCK ON
  SET MESSAGE TO ""
  SET PATH TO F:\HOME\ALEM\NUPFIN
  SET PRINT OFF
  SET REPROCESS TO 4
  SET SAFETY ON
  SET TALK OFF

  *-- Initialize global variables
  gl_batch=.F.          && is a batch operation in progress
  gl_color= ISCOLOR() .AND. SET("DISPLAY") <> "CGAMONO"
  gn_error=0             && 0 if no error, otherwise an error
occurred
  gn_scrsize=21         && number of lines on screen
  gn_send=0              && return value from popup of position
menus
  gn_trace=1            && sets trace level, however you need to
change template
  gc_brdr='1'           && border to use when drawing boxes
  gc_dev='CON'          && Device to use for printing - See Proc.
PrintSet
  gl_leave=.f.          && leave the application
  gc_prognum='   '       && internal program counter to handle
```

```
nested menus
  gc_quit=' '          && memvar for return to caller
  gc_scope=''           && scope, for and while of position at
runtime
  listval='NO_FIELD'   && Pick List value

  *-- remove asterisk to turn clock on
  * SET CLOCK TO

  *-- Blank the screen
  SET COLOR TO
  CLEAR
  SET SCOREBOARD OFF
  SET STATUS OFF

  *-- Define menus
  DO MPDEF                                          && Menu Process
DEFinition

  *-- Execute main menu
  DO WHILE .NOT. gl_leave
    DO MAIN WITH "B00"
    IF gc_quit = 'Q'
      EXIT
    ENDIF
    gl_leave = _NodShake( " ;     Do you want to leave this
application?   ", ;
                          13, 18, 2, 44, .T. )
  ENDDO
```

```
*-- Reset environment
DEACTIVATE WINDOW FullScr
?? Color(gc_color)
gn_ApGen=gn_ApGen-1
SET BELL  &gc_bell.
SET CARRY &gc_carry.
SET CLOCK &gc_clock.
SET CENTURY &gc_century.
SET CONFIRM &gc_confirm.
SET CURSOR  &gc_cursor.
SET DELIMITERS &gc_deli.
SET DISPLAY TO &gc_display.
SET STATUS &gc_status.
SET SAFETY &gc_safety.
SET SCOREBOARD &gc_score.
SET TALK   &gc_talk.

IF gn_Apgen < 1
  ON KEY LABEL F1
  CLEAR WINDOWS
  CLEAR ALL
  CLOSE ALL
  CLOSE PROCEDURE
  SET ESCAPE ON
  SET MESSAGE TO ""
  CLEAR
ELSE
  DEFINE WINDOW FullScr FROM 0,0 TO gn_oldsize+3,79 NONE
  DEFINE WINDOW Savescr FROM 0,0 TO gn_oldsize,79 NONE
  DEFINE WINDOW Helpscr FROM 0,0 TO gn_oldsize,79 NONE
```

```
    ACTIVATE WINDOW FullScr
  ENDIF

RETURN
*-- EOP: C:NUPIFIN

*-------------------------------------------------------------
------------------
* Description..: Procedure files for generated menu system.
* The programs that follow are common to main routines
* The last procedure is the Menu Process DEFinition
*-------------------------------------------------------------

PROCEDURE Lockit
PARAMETER pc_ltype
*-------------------------------------------------------------
* Lock the current file or record based on the value of
<pc_ltype>.
*-------------------------------------------------------------
  IF NETWORK()
    gn_error=0
    ON ERROR DO Multerr
    IF pc_ltype = "1"
      ll_lock=FLOCK()
    ENDIF
    IF pc_ltype = "2"
      ll_lock=RLOCK()
    ENDIF
    ON ERROR
  ENDIF
```

```
RETURN
*-- EOP: Lockit WITH pc_ltype

PROCEDURE Info_Box
PARAMETERS pc_say
*-------------------------------------------------------------
* Display the message <pc_say> inside of boxes.
*-------------------------------------------------------------
  ? pc_say
  ? REPLICATE("-",LEN(pc_say))
  ?
RETURN
*--EOP: Info_Box WITH pc_say

PROCEDURE ShowPick
*-------------------------------------------------------------
* Show pick list values.
*-------------------------------------------------------------
  PRIVATE ln_ikey, x1, x2

  listval=PROMPT()
  IF LEFT(entryflg,1)="B"
    lc_file=POPUP()
    DO &lc_file. WITH "A"
    RETURN
  ENDIF
  IF TYPE("lc_window")="U"
    ACTIVATE WINDOW ShowPick
  ELSE
    ACTIVATE WINDOW &lc_window.
```

```
    ENDIF

    STORE 0 TO ln_ikey,x1,x2
    ln_ikey=LASTKEY()
    IF ln_ikey=13
      x1=AT(TRIM(listval)+',',lc_fldlst)
      IF x1 = 0
        lc_fldlst=lc_fldlst+TRIM(listval)+','
      ELSE
        x2=AT(',',SUBSTR(lc_fldlst,x1))
        lc_fldlst=STUFF(lc_fldlst,x1,x2,'')
      ENDIF
      CLEAR
      ? lc_fldlst
    ENDIF
    ACTIVATE SCREEN

RETURN
*--EOP: ShowPick

PROCEDURE Cleanup
*-------------------------------------------------------------
* Test whether report option was selected
*-------------------------------------------------------------

  DO CASE

    CASE gc_dev='CON'
      DO Wait4Key
```

```
    CASE gc_dev='PRN'
      SET PRINT OFF
      SET PRINTER TO

    CASE gc_dev='TXT'
      CLOSE ALTERNATE

  ENDCASE
  gc_dev='CON'

RETURN
*-- EOP: Cleanup

PROCEDURE Pause
PARAMETER pc_msg
*-------------------------------------------------------------
* Procedure to display a message or errors in a window
* Parameters : pc_msg = message line
*-------------------------------------------------------------
  PRIVATE lc_msg, lCursor

  IF TYPE("lc_message")="U"
    gn_error=ERROR()
  ENDIF
  lc_msg = pc_msg
  lc_option='0'

  ACTIVATE WINDOW Pause
  IF gn_error > 0
    IF TYPE("lc_message")="U"
```

```
      @ 0,1 SAY LEFT( [An error has occurred !! - Error message:
]+MESSAGE() , 76 )
    ELSE
      @ 0,1 SAY LEFT( [Error # ]+lc_message , 76 )
    ENDIF
  ENDIF

  @ 1,1 SAY LEFT( lc_msg, 76 )
  lCursor = SET( "CURSOR" ) = "ON"
  SET CURSOR OFF
  WAIT " Press any key to continue..."
  IF lCursor
    SET CURSOR ON
  ENDIF

  DEACTIVATE WINDOW Pause

RETURN
*-- EOP: Pause WITH pc_msg

PROCEDURE Multerr
*-------------------------------------------------------------
* Errors will be handled in one of two ways:
*
* 1) Display message and return to menu
* 2) Dialog box for options Try again and Return to menu
*-------------------------------------------------------------
  PRIVATE lc_opt, lc_erno

  gn_error=ERROR()                    && Set the global error
```

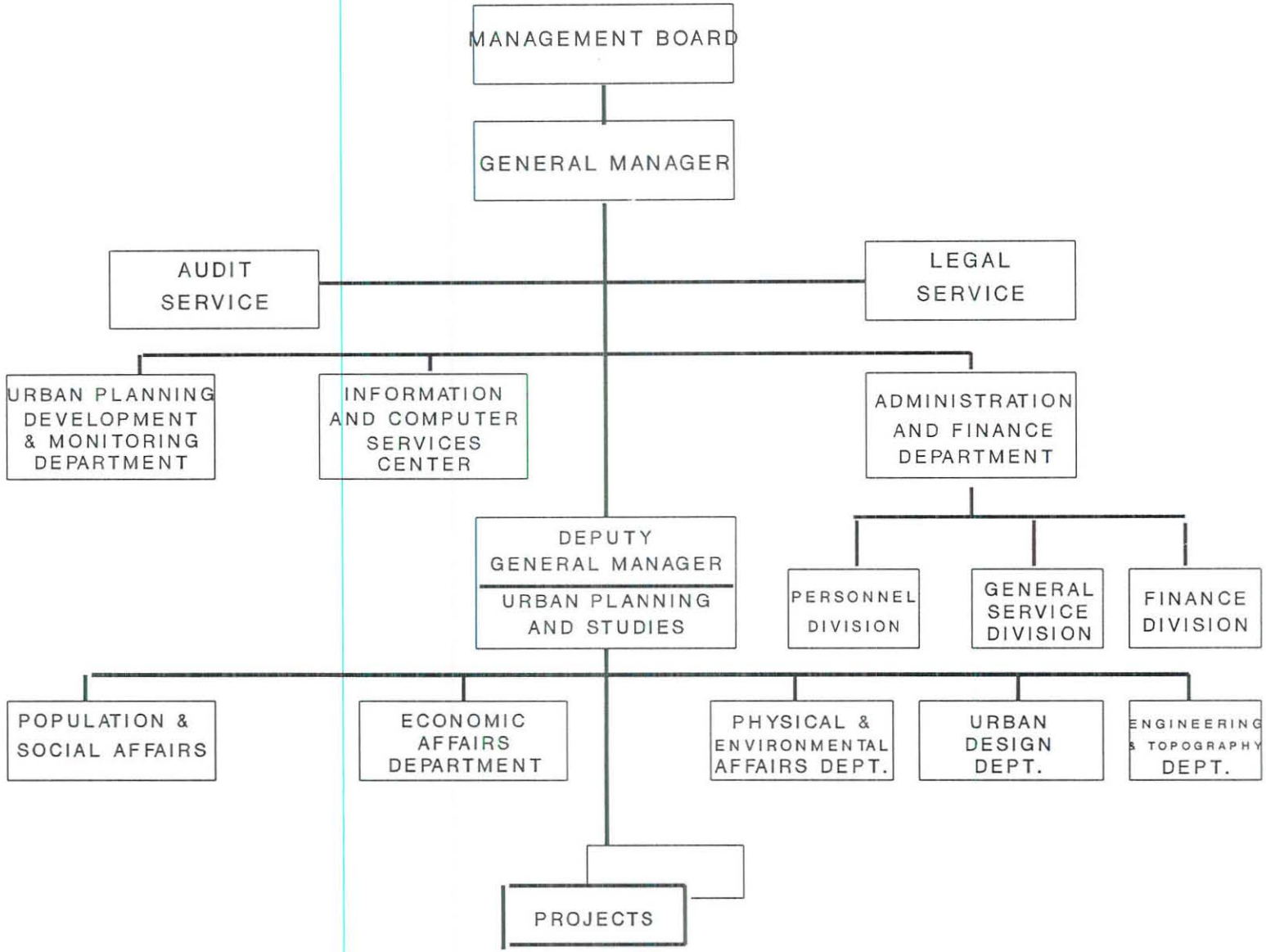
```
variable
  lc_erno=STR(ERROR(),3)+','          && Contains error number
to test
  lc_opt='T'                           && Option variable

  *-- Dialog box for options Try again and Return to menu.
  IF lc_erno $ "108,109,128,129,"
    ACTIVATE WINDOW Pause
    @ 0,2 SAY lc_erno+" "+MESSAGE()
    @ 2,22 SAY "T = Try again, R = Return to menu." ;
           GET lc_opt PICTURE "!" VALID lc_opt $ "TR"
    READ
    DEACTIVATE WINDOW Pause
    IF lc_opt = "R"
      RETURN
    ENDIF
  ENDIF

  *-- Display message and return to menu.
  IF .NOT. lc_erno $ "108,109,128,129,"
    DO PAUSE WITH ERROR()
    RETURN
  ENDIF

  gn_error=0                           && Reset global variable
  RETRY                                && Try the command again

RETURN
*-- EOP: Multerr
```

```
PROCEDURE Trace
PARAMETERS pc_msg, pn_lvl
*-------------------------------------------------------------
* DESCRIPTION
*   Trace procedure to let programmer know what module
*   is about to execute and what module has executed.
*
* PARAMETERS
*   pc_msg    = message line
*   pn_lvl    = trace level
*
*-------------------------------------------------------------
  PRIVATE lc_msg, ln_lvl, lc_trp

  lc_msg = pc_msg
  ln_lvl = pn_lvl
  lc_trp = ' '
  IF gn_trace < ln_lvl                    && Check for trace off
    RETURN
  ENDIF

  DEFINE WINDOW trace FROM 5,0 TO 16,79 DOUBLE
  ACTIVATE WINDOW trace

  DO WHILE lc_trp <> 'Q'
    CLEAR
    @ 2,40-LEN(lc_msg)/2 SAY lc_msg
    @ 4,05 SAY 'S - Set trace level, D - Display status, M -
display Memory'
    @ 5,05 SAY 'P - Turn printer on, Q - to Quit'
```

```
lc_trp = 'Q'
@ 5,38 GET lc_trp PICTURE "@M Q,S,D,M,P"
READ

DO CASE

  CASE lc_trp = 'S'
    @ 2,01 CLEAR
    @ 2,33 SAY 'Set trace level'
    @  4,05  SAY  'Enter  trace  level  to  change  to:'  GET
gn_trace PICTURE '#'
    @ 5,05 SAY [            '0' turns trace off]
    READ

    IF gn_trace=0
      @ 2,01 CLEAR
      @ 3,05 SAY 'Trace is now turned off..To reactivate
Trace - Press [F3]'
      @ 4,05 say 'Press any key to continue...'
      WAIT ''
      ON KEY LABEL F3 gn_trace = 1
    ENDIF

  CASE lc_trp = 'D'
    DISPLAY STATUS
    WAIT

  CASE lc_trp = 'M'
    DISPLAY MEMORY
    WAIT
```

```
      CASE lc_trp = 'P'
        SET PRINT ON

    ENDCASE
  ENDDO

  SET PRINT OFF
  RELEASE WINDOW trace
  @ 24,79 SAY " "

RETURN
*-- EOP: Trace WITH pc_msg, pn_lvl


PROCEDURE PrintSet
*-------------------------------------------------------------
* Procedure for toggling printer output
*-------------------------------------------------------------
  PRIVATE lc_window

  gc_dev='CON'
  lc_choice=' '
  gn_pkey=0
  gn_send=3

  DEFINE WINDOW printemp FROM 08,25 TO 17,56

  DEFINE POPUP SavePrin FROM 10,40
    DEFINE BAR 1 OF SavePrin PROMPT " Send output to ..." SKIP
    DEFINE BAR 2 OF SavePrin PROMPT REPLICATE(CHR(196),24) SKIP
```

APPENDIX A: MASTER ORGANIZATION CHART
NATIONAL URBAN PLANNING INSTITUTE(NUPI)



```
    DEFINE BAR 3 OF SavePrin PROMPT " CON:   Console" MESSAGE
"Send output to Screen"
    DEFINE BAR 4 OF SavePrin PROMPT " LPT1:  Parallel port 1 "
MESSAGE "Send output to LPT1:"
    DEFINE BAR 5 OF SavePrin PROMPT " LPT2:  Parallel port 2"
MESSAGE "Send output to LPT2:"
    DEFINE BAR 6 OF SavePrin PROMPT " COM1:  Serial port 1"
MESSAGE "Send output to COM1:"
    DEFINE BAR 7 OF SavePrin PROMPT " FILE = REPORT.TXT" MESSAGE
"Send output to File Report.txt"
  ON SELECTION POPUP SavePrin DEACTIVATE POPUP

  lc_window = WINDOW()                    && Capture current
window name
  IF .NOT. ISBLANK( lc_window )          && If window was active
    ACTIVATE SCREEN                       && Activate screen for
correct popup posit
  ENDIF

  ACTIVATE POPUP SavePrin
  gn_send = BAR()
  RELEASE POPUP SavePrin

  IF .NOT. ISBLANK( lc_window )          && If window was active
before
    ACTIVATE WINDOW &lc_window.           && Reactivate it
  ENDIF

  IF gn_send <> 0                        && If user madde a popup
selection
```

```
    IF gn_send = 7                          && Output to file
      gc_dev = 'TXT'
      SET ALTERNATE TO REPORT.TXT
      SET ALTERNATE ON
    ELSE
      IF gn_send <> 3                       && Output not to the
screen
        gc_dev = 'PRN'
        ON ERROR DO prntrtry
        DO CASE
          CASE gn_send = 4
            SET PRINTER TO LPT1
          CASE gn_send = 5
            SET PRINTER TO LPT2
          CASE gn_send = 6
            SET PRINTER TO COM1
        ENDCASE
        SET PRINT ON
        ON ERROR
      ENDIF
    ENDIF

  ELSE
    gn_pkey = 27                          && Signal escape pressed
to caller
  ENDIF

  RELEASE WINDOW printemp

RETURN
```

```
*-- EOP: PrintSet

PROCEDURE PrntRtry
*-------------------------------------------------------------
* On error routine for handling printer errors.
*-------------------------------------------------------------
  PRIVATE ll_escape
  ll_escape = SET("ESCAPE") = "ON"

  IF .NOT. PRINTSTATUS()                && If printer not ready

    IF ll_escape                        && If Escape on, set it
off
      SET ESCAPE OFF
    ENDIF

    ACTIVATE WINDOW printemp
    @ 1,0 SAY "Please ready your printer or"
    @ 2,0 SAY "     press ESC to cancel"

    *-- Loop until printer is ready or user presses escape
    gn_pkey = 0
    DO WHILE ( .NOT. PRINTSTATUS()) .AND. gn_pkey <> 27
      gn_pkey = INKEY()
    ENDDO

    DEACTIVATE WINDOW printemp
    IF ll_escape
      SET ESCAPE ON
    ENDIF
```

```
    IF gn_pkey <> 27                    && If user wants to
retry
      RETRY                             && Retry the print
command
    ENDIF

  ENDIF

RETURN
*-- EOP: PrntRtry
```

```
PROCEDURE Position
*-------------------------------------------------------------
* Record positioning (Ask at runtime)
*-------------------------------------------------------------

  IF LEN(DBF()) = 0                     && Make sure a DBF is
open
    DO Pause WITH "Database not in use. "
    RETURN
  ENDIF

  IF EOF()                              && If end of file, go
to top
    GOTO TOP
  ENDIF

  ll_deli = SET("DELIMITERS") = "ON"
  SET DELIMITERS OFF
  ll_space = SET("SPACE") = "ON"
```

```
  SET SPACE ON

  ln_type    = 0                          && Sublevel selection
  ln_rkey    = READKEY()                  && Test for ESC or
Return
  ln_rec     = RECNO()                    && DBF record number
  ln_num     = 0                          && For input of a number
  ll_logic   =.T.                         && For input of a
logical
  ld_date    = DATE()                     && For input of a date
  lc_option = '0'                         && Main option ie. Seek,
Goto and Locate
  lc_ln1     = SPACE(100)                 && For input of a
character

  *-- Scope ie. ALL, REST, NEXT <n>
  STORE SPACE(10) TO lc_scp

  *-- 1 = Character SEEK, 2 = For clause, 3 = While clause
  STORE SPACE(40) TO lc_ln2, lc_ln3
  lc_temp  = ""
  gc_scope = ""
  @ 0,00 SAY "Index order: "+IIF(""=ORDER(),"Database is in
natural order",ORDER())
  @ 1,00 SAY "Listed below are the first 16 fields."
  lc_temp=REPLICATE(CHR(196),19)
                @   2       ,       0   S       A       Y
CHR(218)+lc_temp+CHR(194)+lc_temp+CHR(194)+lc_temp+CHR(194)+l
c_temp
  ln_num=240
```

```
  *-- Build a list of fields in the current DBF
  DO WHILE ln_num < 560
    lc_temp = FIELD( (ln_num-240)/20 +1)
    @ (ln_num/80),MOD(ln_num,80) SAY CHR(179)+;

lc_temp+SPACE(11-LEN(lc_temp))+;
                                    SUBSTR("= Char  = Date  =
Logic = Num   = Float = Memo          ",;

AT(TYPE(lc_temp),"CDLNFMU")*8-7,8)
    ln_num=ln_num+20
  ENDDO
  ln_num=1

  DEFINE POPUP Posit1 FROM 8,30
    DEFINE BAR 1 OF Posit1 PROMPT " Position by " SKIP
    DEFINE BAR 2 OF Posit1 PROMPT REPLICATE(CHR(196),21) SKIP
    DEFINE BAR 3 OF Posit1 PROMPT " SEEK Record" MESSAGE "Search
on index key" SKIP FOR ""=ORDER()
    DEFINE BAR 4 OF Posit1 PROMPT " GOTO Record" MESSAGE
"Position to specific record"
    DEFINE BAR 5 OF Posit1 PROMPT " LOCATE Record " MESSAGE
"Locate record for condition"
    DEFINE BAR 6 OF Posit1 PROMPT REPLICATE(CHR(196),21) SKIP
    DEFINE BAR 7 OF Posit1 PROMPT " Change index order   "
MESSAGE "Reassign current index order"
    DEFINE BAR 8 OF Posit1 PROMPT " Done positioning " ;
      MESSAGE "Press return here, when ready " +;
              "(ESC cancels selections, except index order)"
  ON SELECTION POPUP Posit1 DEACTIVATE POPUP
```

```
SET CONFIRM ON
gn_send = 0
DO WHILE gn_send = 0

  ACTIVATE POPUP Posit1
  gn_send = BAR()
  lc_option = ltrim(str(gn_send))       && Convert bar# to
string

  IF gn_send = 0
    gc_scope=""
    GOTO ln_rec
    EXIT
  ENDIF

  gn_send=0

  DO CASE

    CASE lc_option='3'                  && Seek record

      IF LEN(NDX(1))=0 .AND. LEN(MDX(1))=0
        DO Pause WITH "Can't use this option - No index files
are open."
        LOOP
      ENDIF
      lc_ln1=SPACE(100)
      DEFINE WINDOW Posit2 FROM 8,14 TO 17,67 DOUBLE
      ACTIVATE WINDOW Posit2
      keyexpr=""
```

```
lc_type = Get_Type( ORDER() )
@ 1,1 SAY "Key expression:"
@ 2,1
?? keyexpr FUNCTION "V50"
SET CONFIRM ON
@ 5,1 SAY "Enter the key expression to search for:"
DO CASE
  CASE lc_type = "C"
    @ 6,1 GET lc_ln1 FUNCTION "S50"
  CASE lc_type = "D"
    @ 6,1 GET ld_date FUNCTION "D"
  CASE lc_type = "L"
    @ 6,1 GET ll_logic
  CASE lc_type = "N" .OR. lc_type = "F"
    @ 6,1 GET ln_num PICTURE "################.####"
ENDCASE
SET CURSOR ON
READ
SET CURSOR OFF
SET CONFIRM OFF
RELEASE WINDOWS Posit2

IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
  DO CASE
    CASE lc_type = "C"
      SEEK TRIM(lc_ln1)
    CASE lc_type = "D"
      SEEK ld_date
    CASE lc_type = "L"
      SEEK ll_logic
```

```
            CASE lc_type = "N" .OR. lc_type = "F"
              SEEK ln_num
            ENDCASE
          ELSE
            GOTO TOP
            LOOP
          ENDIF

      CASE lc_option='4'                    && Go to

          ln_type=1
          DEFINE POPUP Posit2 FROM 8,30
            DEFINE BAR 1 OF Posit2 PROMPT " GOTO:" SKIP
            DEFINE BAR 2 OF Posit2 PROMPT REPLICATE(CHR(196),10)
SKIP
            DEFINE BAR 3 OF Posit2 PROMPT " TOP" MESSAGE "GOTO Top
of File"
            DEFINE BAR 4 OF Posit2 PROMPT " BOTTOM" MESSAGE "GOTO
Bottom of File"
            DEFINE BAR 5 OF Posit2 PROMPT " Record # " MESSAGE
"GOTO A Specific Record"
          ON SELECTION POPUP Posit2 DEACTIVATE POPUP

          ACTIVATE POPUP posit2
          gn_send = BAR()
          ln_type = gn_send
          gn_send=0
          IF ln_type <> 0                   && If the user selected
a bar
```

```
          DO CASE

            CASE ln_type=5                    && GOTO A Specific
Record

              DEFINE WINDOW Posit2 FROM 8,26 TO 13,50 DOUBLE
              ACTIVATE WINDOW Posit2
              ln_num=0
                @    3,1    SAY    "Max.    Record    #    =
"+LTRIM(STR(RECCOUNT()))
              @  1,1  SAY  "Record  to  GOTO"  GET  ln_num  PICT
"######" RANGE 1,RECCOUNT()
              SET CURSOR ON
              READ
              SET CURSOR OFF
              IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
                GOTO ln_num
              ENDIF
              RELEASE WINDOWS Posit2

            CASE ln_type=4                 && GOTO Bottom of file
              GOTO BOTTOM

            CASE ln_type=3                 && GOTO Top of file
              GOTO TOP

          ENDCASE

        ENDIF
```

```
      CASE lc_option='5'                    && Locate

        DEFINE WINDOW Posit2 FROM 8,16 TO 14,66 DOUBLE
        ACTIVATE WINDOW Posit2
        @ 1,19 SAY "ie. ALL, NEXT <n>, and REST"
        @ 1,01 SAY "Scope:" GET lc_scp
        @ 2,01 SAY "For:  " GET lc_ln2
        @ 3,01 SAY "While:" GET lc_ln3
        SET CURSOR ON
        READ
        SET CURSOR OFF

        IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
          lc_temp=TRIM(lc_scp)
          lc_temp=lc_temp  +  IIF(LEN(TRIM(lc_ln2))  >  0," FOR
"+TRIM(lc_ln2),"")
          lc_temp=lc_temp + IIF(LEN(TRIM(lc_ln3)) > 0," WHILE
"+TRIM(lc_ln3),"")
          IF .NOT. ISBLANK( lc_temp )
            LOCATE &lc_temp.
            IF .NOT. EOF()
              gc_scope=lc_temp
            ENDIF
          ELSE
            DO Pause WITH "All fields were blank."
          ENDIF
        ENDIF
        RELEASE WINDOW Posit2

      CASE lc_option='7'                    && Change index order
```

```
        DO Showtag
        IF LASTKEY() = 27
          LOOP
        ENDIF
        GOTO TOP
        ln_rec=RECNO()
        @ 0,00
        @ 0,00 SAY "Index order: "+IIF(""=ORDER(),"Database is
in natural order",ORDER())

      CASE lc_option='8'                  && User wants to exit
        gn_send=1                         && Signal end of loop

    ENDCASE

    IF EOF()
      DO Pause WITH "Record not found."
      GOTO ln_rec
      gn_send=0
    ENDIF
  ENDDO
  SET CURSOR ON
  IF ll_deli
    SET DELIMITERS ON
  ENDIF
  IF .NOT. ll_space
    SET SPACE OFF
  ENDIF
  SET CONFIRM OFF
  IF ISBLANK( lc_scp )
```

```
    gc_scope="REST "+gc_scope
  ENDIF

RETURN
*--EOP: Position

PROCEDURE Showtag
*-------------------------------------------------------------
* Display a list of tags for the current file
*-------------------------------------------------------------
  PRIVATE cnt, idxexp, length

  cnt    = 1                              && Count of index tags
  idxexp = ""                             && Index tag expression
  length = 12                             && Minimum tag width

  DEFINE POPUP Showtag FROM 8,10
    DEFINE BAR 1 OF Showtag PROMPT " Index Order" SKIP
    DEFINE BAR 3 OF Showtag PROMPT " Natural Order "

    *-- For each TAG define a menu bar
    DO WHILE .NOT. ISBLANK( TAG( cnt ) )
      scnt = LTRIM(STR(cnt+4))            && Add four to bar
number
      string   =   SUBSTR(TAG(cnt)+SPACE(10),1,11)+CHR(179)+"
"+LEFT(KEY(cnt),47)
      IF length < LEN(string)             && Adjust popup width
as needed
        length = LEN(string)
      ENDIF
```

```
      DEFINE BAR &scnt. OF Showtag PROMPT string
      cnt = cnt + 1
    ENDDO

    DEFINE BAR 2 OF Showtag PROMPT REPLICATE(CHR(205),length+1)
SKIP
    DEFINE BAR 4 OF Showtag ;
                                    P     R     O     M     P     T
REPLICATE(CHR(196),11)+CHR(194)+REPLICATE(CHR(196),length-11)
SKIP
  ON SELECTION POPUP Showtag DEACTIVATE POPUP

  IF cnt > 1                              && If the DBF has tags

    xx=4                                  && Right arrow
    DO WHILE xx = 4 .OR. xx = 19          && Ignore esacpe from
arrow keys
      ACTIVATE POPUP Showtag
      idxexp = PROMPT()
      xx=LASTKEY()
    ENDDO

    IF BAR() <> 0                         && If user selected a
tag
      IF LTRIM(RTRIM(idxexp))="Natural Order"
        SET ORDER TO
      ELSE
        lc_ord = TRIM(LEFT(idxexp,10))
        lc_exp = TRIM(SUBSTR(idxexp,14))
        SET ORDER TO &lc_ord.
```

```
      ENDIF
    ENDIF

  ENDIF
  RELEASE POPUP Showtag

RETURN
*--EOP: Showtag

FUNCTION Get_Type
PARAMETER tagname
*-------------------------------------------------------------
* Determine the type of the tag name.
* Also will set the value of <keyexpr> for the calling routine.
*-------------------------------------------------------------
  PRIVATE ll_exact, ll_talk, keytype, lc_temp, offset, ;
          ln_error

  IF SET( "TALK" ) = "ON"
    SET TALK OFF
    ll_talk = .T.
  ELSE
    ll_talk = .F.
  ENDIF
  ll_exact = SET("EXACT") = "ON"
  SET EXACT ON

  keytype  = "U"                          && Assume undefined tag
  lc_temp  = "U"                          && Assume error occured
with undefined key
```

```
  offset    = 1                          && Counter for tags in
DBF
  ndxflag   = .T.
  ln_error = 0

  DO WHILE .NOT. ISBLANK( TAG( offset ) )

    IF TAG(offset) = tagname
      lc_temp = TYPE( KEY( offset ) )
      ln_error = 0
      ON ERROR ln_error = ERROR()

      DO CASE
        CASE lc_temp="C"
          SEEK "A"
        CASE lc_temp="D"
          SEEK DATE()
        CASE lc_temp="L"
          SEEK .T.
        CASE lc_temp="N" .OR. lc_temp="F"
          SEEK 1
      ENDCASE

      ON ERROR
      IF ln_error = 0
         keytype = lc_temp
         keyexpr = KEY(offset)
         EXIT
      ENDIF
    ENDIF
```

```
    offset = offset + 1

  ENDDO

  IF .NOT. ll_exact
    SET EXACT OFF
  ENDIF
  IF ll_talk
    SET TALK ON
  ENDIF

RETURN( keytype )
*--EOP: Get_Type( Tagname )

PROCEDURE BefAct
*-------------------------------------------------------------
* Save the screen before executing a menu option.
*-------------------------------------------------------------
  SAVE SCREEN TO Browscr&lc_ApGen.
  DEACTIVATE WINDOW Fullscr
  SET SCOREBOARD ON
RETURN
*--EOP: BefAct

PROCEDURE AftAct
*-------------------------------------------------------------
* Restore the screen after executing a menu option.
*-------------------------------------------------------------
  CLEAR
  SET SCOREBOARD OFF
```

```
  ACTIVATE WINDOW Fullscr
  RESTORE SCREEN FROM Browscr&lc_ApGen.
  RELEASE SCREEN Browscr&lc_ApGen.
RETURN
*--EOP: AftAct

PROCEDURE Postnhlp
*-------------------------------------------------------------
* Display help screens for generic menus.
*-------------------------------------------------------------
  DEFINE WINDOW Temphelp FROM 3,12 TO 19,67
  ACTIVATE WINDOW Temphelp
  DO CASE
    CASE "SEEK" $ PROMPT()
    *-- HELP SEEK
      ? " SEEK <exp>"
      ?
      ? " Evaluates a specified expression and attempts to"
      ? " find its value in the master index of the database"
      ? " file.  Returns a logical true (.T.) if the index"
      ? " key is found, and a logical false (.F.) if it is"
      ? " not found."
      ?
      ? " Ex: SEEK CTOD('11/03/87')  -  converts the"
      ? "     expression from character to date and"
      ? "     then searches for the value in the index"
      ?
    CASE LEFT(LTRIM(PROMPT()),4) $ "GOTO TOP BOTT Reco"
    *-- HELP GOTO
      ? " GO/GOTO BOTTOM/TOP [IN <alias>]"
```

```
      ? " or"
      ? " GO/GOTO [RECORD] <record number> [IN <alias>]"
      ? " or"
      ? " <record number>"
      ?
      ? " Positions the record pointer to a specified record"
      ? " or location in the active database file."
      ?
      ? "      TOP moves the pointer to the first record"
      ? "      BOTTOM moves the pointer to the last record"
      ?
      ? " Ex: 4   -   moves the record pointer to record 4"
      ?
    CASE "LOCATE" $ PROMPT()
    *-- HELP LOCATE
      ? " LOCATE FOR <condition> [<scope>]"
      ? "      [WHILE <condition>]"
      ?
      ? " Searches the active database file, sequentially,"
      ? " for the first record that meets the specified"
      ? " criteria.  The function FOUND() returns true (.T.)"
      ? " if LOCATE is successful."
      ?
      ? " Ex: LOCATE FOR Age = '25' NEXT 5"
      ? "     searches for the next five records"
      ? "     containing 25 in the Age field"
      ?
    CASE "Change index order" $ PROMPT() .OR. POPUP() =
"SHOWTAG"
      ?
```

```
      ? [ Select "Change index order" to select the master]
      ? " (controlling) index. You will see a list of indexes"
      ? " from the  stand-alone indexes (.ndx). and mdx"
      ? " file(s) that are activated. The first option in the"
      ? " list, NATURAL ORDER, uses the file in its unindexed"
      ? " state.  Press RETURN to select your choice by which"
      ? " to order the file."
      ?
  ENDCASE

  DO Wait4Key

  DEACTIVATE WINDOW Temphelp
  RELEASE WINDOW Temphelp
RETURN
*--EOP: Postnhlp

PROCEDURE Wait4Key
*-------------------------------------------------------------
-----
* Wait for a key press or mouse click.
*-------------------------------------------------------------
  PRIVATE ll_escape

  ll_escape = SET( "ESCAPE" ) = "ON"
  SET ESCAPE OFF
  WAIT
  IF ll_escape
    SET ESCAPE ON
  ENDIF
```

```
RETURN
*-- EOP: Wait4Key

FUNCTION Color
PARAMETERS pc_scolor
*-------------------------------------------------------------
* Format:
* COLOR( <expC> )
*     <expC>  =  NORMAL,  HIGHLIGHT,  MESSAGES,  TITLES,  BOX,
INFORMATION, FIELDS
*         or a variable with all colors store in it
*  Ver: dBASE 1.1
*
* The COLOR() function either returns or sets colors returned
with the
* SET("attribute") setting
* If <expC> is a color string then null is returned otherwise
the color
* setting is returned for one of dBASE's color options
*
* See Also: SET("attribute")
*
*-------------------------------------------------------------
PRIVATE color_num, color_str, cnt

pc_scolor = UPPER(pc_scolor)
IF pc_scolor = "COLOR"
  *- Return standard, enhanced, border colors only
  RETURN SUBSTR(SET("attr"),1, AT(" &", SET("attr")))
ENDIF
```

```
*- Declare array to parse color options from SET("attr")
PRIVATE color_
DECLARE color_[8]
*- Determine if user is restoring colors vs. saving colors
IF " &" $ pc_scolor
  color_str = ","+pc_scolor+","                    && Restore
color attributes
ELSE
  color_str = ","+SET("ATTRIBUTE")+","             && Save color
attributes
ENDIF

*-- Stuff array with individual color setting
color_str = STUFF(color_str, AT(" &", color_str), 4, ",")
cnt = 1
DO WHILE cnt <= 8
  color_str = SUBSTR(color_str, AT(",", color_str ) +1 )
  color_[cnt] = SUBSTR(color_str, 1, AT(",", color_str ) - 1)
  cnt = cnt + 1
ENDDO

IF " &" $ pc_scolor
  *-- Set color back
  SET COLOR TO ,,&color_[3].                       && Border color
  SET COLOR OF NORMAL TO &color_[1].
  SET COLOR OF HIGHLIGHT TO &color_[2].
  SET COLOR OF MESSAGES TO &color_[4].
  SET COLOR OF TITLES TO &color_[5].
  SET COLOR OF BOX TO &color_[6].
  SET COLOR OF INFORMATION TO &color_[7].
```

```
  SET COLOR OF FIELDS TO &color_[8].
ELSE
  *-- Return color string requested
  DO CASE
    CASE pc_scolor $ "NORMAL"
      color_num =  1
    CASE pc_scolor $ "HIGHLIGHT"
      color_num =  2
    CASE pc_scolor $ "BORDER"
      color_num =  3
    CASE pc_scolor $ "MESSAGES"
      color_num =  4
    CASE pc_scolor $ "TITLES"
      color_num =  5
    CASE pc_scolor $ "BOX"
      color_num =  6
    CASE pc_scolor $ "INFORMATION"
      color_num =  7
    CASE pc_scolor $ "FIELDS"
      color_num =  8
  ENDCASE
ENDIF
RETURN IIF(" &" $ pc_scolor, "", color_[color_num])

FUNCTION _NodShake
PARAMETERS  pc_mssg,  pn_up,  pn_left,  pn_height,  pn_max,
pl_dflt_no
*-------------------------------------------------------------
* NAME
*   _NodShake
```

```
*
* DESCRIPTION
*   Accepts a YES/NO response from user
*
* SYNOPSIS
*    DO _NodShake WITH pc_mssg, pn_up, pn_left, pn_height,
pn_max, pl_dflt_no
*
* PARAMETERS
*   pc_mssg:    dialog box message
*   pn_up:      upper corrdinate of dialog box
*   pn_left:    left coordinate of dialog box
*   pn_height:  height of dialog box
*   pn_max:     maximum width of a line in message
*    pl_dflt_no: flag indicating if default pad highlighted
should be "NO"
*
* EXAMPLE
*    pl_set = _NodShake( pc_vermssg, 13, 25, 2, 28, .T. )
*-------------------------------------------------------------

  PRIVATE ll_ans, ll_console, ll_wrapset, ln_pspset

  ll_console = SET( "CONSOLE" ) = "OFF"
  SET CONSOLE ON
  ll_wrapset = _wrap
  ln_pspset = _pspacing
  _wrap = .F.
  _pspacing = 1
```

```
  DEFINE WINDOW NodShake DOUBLE ;
     FROM pn_up, pn_left TO pn_up + pn_height + 4, pn_left +
pn_max + 1

  DEFINE MENU NodShake
  DEFINE PAD Yes OF NodShake PROMPT "Yes" ;
     AT pn_height + 1, (pn_max - 12) / 2;
     MESSAGE "Select option and press ENTER, or press first
letter" + ;
             " of desired option"

  ON SELECTION PAD Yes OF NodShake DEACTIVATE MENU
  DEFINE PAD No OF NodShake PROMPT "No" ;
     AT pn_height + 1, (pn_max - 12) / 2 + 10 ;
     MESSAGE "Select option and press ENTER, or press first
letter" + ;
             " of desired option"

  ON SELECTION PAD No OF NodShake DEACTIVATE MENU
  ACTIVATE WINDOW NodShake
  CLEAR
  ?
  @ 0, 0
  ?? pc_mssg FUNCTION ";"

  ON KEY LABEL Y KEYBOARD "{Alt-Y}{13}"
  ON KEY LABEL N KEYBOARD "{Alt-N}{13}"

  IF pl_dflt_no
    KEYBOARD "{Alt-N}"
```

```
  ENDIF

  ON KEY LABEL RIGHTARROW
  ON KEY LABEL LEFTARROW

  ACTIVATE MENU NodShake

  ON KEY LABEL Y
  ON KEY LABEL N

  IF PAD() = "YES"
    ll_ans = .T.
  ELSE
    ll_ans = .F.
  ENDIF

  RELEASE WINDOW NodShake
  RELEASE MENU NodShake
  _wrap = ll_wrapset
  _pspacing = ln_pspset

  IF ll_console
    SET CONSOLE OFF
  ENDIF

RETURN ll_ans
*-- EOF: _NodShake( pc_mssg, pn_up, pn_left, pn_height, pn_max,
pl_dflt_no )
```

```
PROCEDURE _Err_Box
PARAMETERS pc_msg
*-------------------------------------------------------------
* NAME
*   _Err_Box - Display an error box
*
* SYNOPSIS
*   DO _Err_Box WITH <pc_msg>
*
* DESCRIPTION
*    _Err_Box will display the <pc_msg> string in a box and
prompt the
*  user to press any key to continue processing.  _Err_Box will
display
*   the message based on the length of <pc_msg>.
*
* PARAMETERS
*   pc_msg - the error message to display in the box.  If the
length is
*           greater than 76, the trailing part is chopped off.
*
* EXAMPLE
*   DO _Err_Box WITH "Incorrect window size"
*  Displays the message in a window as follows at row 9 on the
```

```
screen:
*                        +------------------------------+
*                        |                              |
*                        |    Incorrect window size     |
*                        |                              |
*                        | Press any key to continue... |
*                        |                              |
*                        +------------------------------+
*     Note that the width of the window will increase to
accommodate a longer
*   message string.
*
* LIMITATIONS
*   Truncates the message after 76 characters.  Assumes an 80
character
*   wide screen.  Looks best with SET CURSOR OFF.
*
*-------------------------------------------------------------
  PRIVATE  lc_anykey,  lc_msg,  lc_msglen,  lc_win,  ln_press,
ln_width, ll_trap,;
           ll_escape

  lc_anykey = [Press any key to continue...]
  ln_press  = LEN( lc_anykey )
  lc_win = WINDOW()                      && Currently activated
window if any
  lc_msg = LTRIM( RTRIM( pc_msg ) )      && Trimmed message
  ln_msglen = LEN( lc_msg )               && Trimmed length of
message
  ln_width = 0                          && Width of display area
```

```
in window.
  ll_escape = SET("ESCAPE") = "ON"
  SET ESCAPE OFF

  *-- Determine the width needed for the window:
  IF ln_msglen <= ln_press
    ln_width = ln_press
  ELSE
    *-- Make sure the message fits in the window:
    IF ln_msglen > 76
      lc_msg = LEFT( lc_msg, 76 )
      ln_msglen = 76
    ENDIF
    ln_width = ln_msglen
  ENDIF
  DEFINE WINDOW _err_box FROM 9, ((76 - ln_width) + .5) / 2 ;
                TO 15, (ln_width + 83) / 2 DOUBLE
  ln_width = ( ln_width + 2 )

  *-- Display the message and prompt to the window and wait for
a key press
  ACTIVATE WINDOW _err_box
  @ 1, ( ln_width - ln_msglen ) / 2 SAY lc_msg
  @ 3, ( ln_width - ln_press ) / 2 SAY lc_anykey
  SET CONSOLE OFF                          && For mouse click
recognition
  WAIT
  SET CONSOLE ON

  *-- Clean up the window display and reactivate the previous
```

```
window
  RELEASE WINDOW _err_box
  IF ISBLANK( lc_win )
    ACTIVATE SCREEN
  ENDIF

  IF ll_escape
    SET ESCAPE ON
  ELSE
    SET ESCAPE OFF
  ENDIF

RETURN
*-- EOP: _Err_Box WITH pc_msg



PROCEDURE MPDEF
*-------------------------------------------------------------
* Program.....: MPDEF
* Author......: This is an APPLICATION OBJECT.
* Date........: 4-25-96
* Notice......: Type information here or greetings to your
users.
* Generated by.: dBASE IV version 2.0
* Description..: Defines all menus in the system for C:NUPIFIN
*-------------------------------------------------------------

  IF gl_color
    SET COLOR OF NORMAL TO W+/B
```

```
    SET COLOR OF MESSAGES TO W+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
  CLEAR

  *-- Sign-on banner
  SET BORDER TO
  @ 3,23 TO 12,56 DOUBLE COLOR RG+/GB
  @ 6,24 SAY "            WELCOME"
  @ 8,24 SAY "          TO NUPI'S"
  @ 10,24 SAY "       FINANCIAL SYSTEM"
  @ 4,24 FILL TO 11,55 COLOR W+/N
  @ 24,30 SAY " Press "+CHR(17)+CHR(196)+CHR(217)+" to continue.
"
  SET CONSOLE OFF
  DO Wait4Key
  SET CONSOLE ON
  CLEAR

  *-- Prevents clearing of menus from commands:
  *-- SET STATUS and SET SCOREBOARD
  DEFINE WINDOW FullScr FROM 0,0 TO 24,79 NONE
  *-- Position at runtime and batch process
  DEFINE WINDOW Savescr FROM 0,0 TO 21,79 NONE
  *-- F1 Help
  DEFINE WINDOW Helpscr FROM 0,0 TO 21,79 NONE
```

```
IF gn_ApGen=1
  *-- Pause message box
  DEFINE WINDOW Pause FROM 15,00 TO 19,79 DOUBLE
ENDIF
ACTIVATE WINDOW FullScr
@ 24,00
@ 23,00 SAY "Loading..."
SET BORDER TO DOUBLE

*-- Bar
DEFINE MENU MAIN MESSAGE "Press Enter to select."
  DEFINE PAD PAD_1 OF MAIN PROMPT "REVENUE/EXPENDITURE" AT 1,1

  ON PAD PAD_1 OF MAIN ACTIVATE POPUP REVEXP
  DEFINE PAD PAD_2 OF MAIN PROMPT "SENED" AT 1,30
  ON PAD PAD_2 OF MAIN ACTIVATE POPUP SENED
  DEFINE PAD PAD_3 OF MAIN PROMPT "PAYROLL" AT 1,44
  ON SELECTION PAD PAD_3 OF MAIN DO ACT01
  DEFINE PAD PAD_4 OF MAIN PROMPT "INVENTORY" AT 1,57
  ON SELECTION PAD PAD_4 OF MAIN DO ACT01
  DEFINE PAD PAD_5 OF MAIN PROMPT "EXIT" AT 1,72
  ON PAD PAD_5 OF MAIN ACTIVATE POPUP EXIT
?? "."

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP REVEXP FROM 2,0 TO 6,19 ;
MESSAGE "Use Up/Down Arrows to choose an item. Press Enter to
select."
```

```
  DEFINE BAR 1 OF REVEXP PROMPT "Revenue"
  DEFINE BAR 2 OF REVEXP PROMPT "Expenditure"
  DEFINE BAR 3 OF REVEXP PROMPT "Budget"
ON SELECTION POPUP REVEXP DO ACT02
?? "."

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP SENED FROM 2,26 TO 8,39 ;
MESSAGE "Use Up/Down Arrow keys to choose an item. Press Enter
to select."
  DEFINE BAR 1 OF SENED PROMPT "Add"
  DEFINE BAR 2 OF SENED PROMPT "Delete"
  DEFINE BAR 3 OF SENED PROMPT "Edit"
  DEFINE BAR 4 OF SENED PROMPT "View"
  DEFINE BAR 5 OF SENED PROMPT "Print"
ON SELECTION POPUP SENED DO ACT03
?? "."

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP EXIT FROM 2,58 TO 5,79 ;
MESSAGE "Use Up/Down Arrows to choose an item. Press Enter to
select."
  DEFINE BAR 1 OF EXIT PROMPT "Exit to dot prompt"
  DEFINE BAR 2 OF EXIT PROMPT "Exit to DOS"
ON SELECTION POPUP EXIT DO ACT04
?? "."
```

```
SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP REVENUE FROM 2,12 TO 7,24 ;
MESSAGE "Use Up/Down Arrow keys to choose an item. Press Enter
to select."
   DEFINE BAR 1 OF REVENUE PROMPT "Add"
   DEFINE BAR 2 OF REVENUE PROMPT "Edit"
   DEFINE BAR 3 OF REVENUE PROMPT "View"
   DEFINE BAR 4 OF REVENUE PROMPT "Print"
ON SELECTION POPUP REVENUE DO ACT05
?? "."

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP EXP FROM 3,12 TO 8,24 ;
MESSAGE "Use Up/Down Arrow keys to choose an item. Press Enter
to select."
   DEFINE BAR 1 OF EXP PROMPT "Add"
   DEFINE BAR 2 OF EXP PROMPT "Edit"
   DEFINE BAR 3 OF EXP PROMPT "View"
   DEFINE BAR 4 OF EXP PROMPT "Print"
ON SELECTION POPUP EXP DO ACT06
?? "."

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP BUDGET FROM 4,12 TO 8,30 ;
```

```
  MESSAGE "Use Up/Down Arrow keys to choose an item. Press Enter
to select."
    DEFINE BAR 1 OF BUDGET PROMPT "View"
    DEFINE BAR 2 OF BUDGET PROMPT "Print"
    DEFINE BAR 3 OF BUDGET PROMPT "Copy to file"
  ON SELECTION POPUP BUDGET DO ACT07
  ?? "."

RETURN
*-- EOP: MPDEF.PRG
```

```
PROCEDURE 1HELP1
*-------------------------------------------------------------
* Handle F1 - Help during program execution
*-------------------------------------------------------------
  PRIVATE lc_popmenu, ll_errbox, ll_status

  ON KEY LABEL F1
  lc_popmenu=IIF( "" = POPUP(), MENU(), POPUP() )
  ll_status = SET( "STATUS" ) = "ON"
  IF ll_status
    SAVE SCREEN TO ls_status
    SET STATUS OFF
    RESTORE SCREEN FROM ls_status
  ELSE
    ACTIVATE WINDOW Helpscr
  ENDIF
  SET ESCAPE OFF
  ACTIVATE SCREEN
  @ 0,0 CLEAR TO 21,79
```

```
@ 1,0 TO 21,79 COLOR RG+/GB
@ 24,00
@ 0,0 SAY ""
ll_errbox = .F.
DO CASE

  *-- help for menu MAIN
  CASE "MAIN" = lc_popmenu
    DO _Err_Box WITH "No Help defined."
    ll_errbox = .T.

  *-- help for menu REVEXP
  CASE "REVEXP" = lc_popmenu
    DO _Err_Box WITH "No Help defined."
    ll_errbox = .T.

  *-- help for menu SENED
  CASE "SENED" = lc_popmenu
    DO _Err_Box WITH "No Help defined."
    ll_errbox = .T.

  *-- help for menu EXIT
  CASE "EXIT" = lc_popmenu
    DO _Err_Box WITH "No Help defined."
    ll_errbox = .T.

  *-- help for menu REVENUE
  CASE "REVENUE" = lc_popmenu
    DO _Err_Box WITH "No Help defined."
    ll_errbox = .T.
```

```
    *-- help for menu EXP
    CASE "EXP" = lc_popmenu
      DO _Err_Box WITH "No Help defined."
      ll_errbox = .T.

    *-- help for menu BUDGET
    CASE "BUDGET" = lc_popmenu
      DO _Err_Box WITH "No Help defined."
      ll_errbox = .T.
    OTHERWISE
      DO _Err_Box WITH "Unknown menu name, help was never
defined."
      ll_errbox = .T.
  ENDCASE

  IF .NOT. ll_errbox
    @ 24,26 SAY "Press any key to continue..."
    SET CONSOLE OFF
    DO Wait4Key
    SET CONSOLE ON
  ENDIF

  SET ESCAPE ON
  @ 24,00
  IF ll_status
    SET STATUS ON
    RESTORE SCREEN FROM ls_status
    RELEASE SCREEN ls_status
  ELSE
    DEACTIVATE WINDOW Helpscr
```

```
  ENDIF
  ON KEY LABEL F1 DO 1HELP1
RETURN
*-- EOP: 1HELP1
```

LIST OF INTERVIEWEES

ATO TESHOME BULA: Information and Documentation Centre, Head. and Administration and Finance Department, A/ Head.

ATO MEGBARU YIMER: Administration and Finance Department, A/Head.

ATO MOHAMMED AHMED: Audit and Inspection Service, Head.

W/O MELETE TEKLEMICAEL : Auditor II. Working in the division as Accountant III.

ATO TESFAYE TELAYE: Working in the division as Accountant II.

ATO YARED TOMELESAN: Working in the division as Budget section, Head.

ATO MEKETE SHEFERAW: Computer centre, A/ Head and System analyst III.

ENGINEER MEHERETEAB YEHEDEGO: Plan Preparation Directorate.

W/rt MELETETSEGA BIZEN : Personnel Department, A/Head.

ATO HAILU MASRESHA: General Service, A/Head.

ATO SOLOMON KEBEDE: Plans and Drawings Service, Head

APPENDIX E

DISCUSSION GUIDE

1. what is the main purpose of your unit?
2. In the course of discharging your duties,
    - What problems do you have and what finance-related information is needed for solving them?
    - What decisions do you make and what finance-related information do you need for decision making?
    - What factors are critical to the success of your activity and what information do you need to achieve success in them or monitor progress?
    - What are the outputs from your activity and what information do you need to measure effectiveness in achieving the outputs?
    - What resources are used in producing the outputs and what information is needed to measure effectiveness in use of resources?

DECLARATION

The Thesis is my original work and has not been presented for a degree in any other university.

Alemtsehaye Mamo Abe

23 May, 1996

The Thesis has been submitted for examination with my approval as university advisor.

Ato Tesfaye Biru

24 May, 1996