1991-03

A micro computer based procurement system: an application of reverse engineering techniques

Skrtich, George T.; Delaney, Daniel E.

Monterey, California. Naval Postgraduate School

https://hdl.handle.net/10945/43745

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A MICRO COMPUTER BASED PROCUREMENT SYSTEM: AN APPLICATION OF REVERSE ENGINEERING TECHNIQUES

by

George T. Skrtich
and
Daniel E. Delaney

March 1991

Thesis Advisor: Magdi Kamel

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) Code AS	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification) A MICRO COMPUTER BASED PROCUREMENT SYSTEM: AN APPLICATION OF REVERSE ENGINEERING TECHNIQUES

12 PERSONAL AUTHOR(S)
Skrtich, George T., and Delaney, Daniel E.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM _____ TO _____	1991 March	134

16 SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Small purchase, Reverse Engineering, Flat Files, APADE, Algorithm

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The Department of the Navy has developed a system called the Automation of Procurement and Accounting Data Entry (APADE), which automates the procurement of nonstandard materials. Small Navy Field contracting locations, however, cannot afford to utilize this service, and the Navy currently has no standard micro computer software for such procurement. This thesis analyzes and reviews the Navy's APADE procurement system using a reverse engineering approach. It establishes an entity relationship model from the existing APADE flat files. This entity relationship model is then used to design and implement a prototype of the APADE system small procurement module on micro computers. The prototype micro computer version emulates the small procurement functions of the mainframe system.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL Magdi Kamel	22b TELEPHONE (Include Area Code) (408) 646-2494 22c OFFICE SYMBOL AS/Ka

DD Form 1473, JUN 86 Previous editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

S/N 0102-LF-014-6603

Approved for public release; distribution is unlimited

A Micro Computer Based Procurement System: An Application of Reverse Engineering Techniques

by

George T. Skrtich
Lieutenant, United States Navy
B.S., West Liberty State, 1976

and

Daniel E. Delaney
Lieutenant, United States Navy
B.S., New Hampshire College, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MANAGEMENT
MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
March 1991

Author: [redacted]
George T. Skrtich
[redacted]
Daniel E. Delaney

Approved By: [redacted]
Magdi N. Kamel, Thesis Advisor
[redacted]
Martin J. McCaffrey, Second Reader
[redacted]
David R. Whipple, Chairman,
Department of Administrative Sciences

ABSTRACT

The Department of the Navy has developed a system called the Automation of Procurement and Accounting Data Entry (APADE), which automates the procurement of nonstandard materials. Small Navy Field contracting locations, however, cannot afford to utilize this service, and the Navy currently has no standard micro computer software for such procurement. This thesis analyzes and reviews the Navy's APADE procurement system using a reverse engineering approach. It establishes an entity relationship model from the existing APADE flat files. This entity relationship model is then used to design and implement a prototype of the APADE system small procurement module on micro computers. The prototype micro computer version emulates the small procurement functions of the mainframe system.

TABLE OF CONTENTS

I. THESIS RESEARCH

A. BACKGROUND

The United States Navy has implemented the Navy's Automated Procurement and Accounting Data Entry (APADE) System throughout the Navy Field Contracting System (NFCS) to provide support to Naval Regional Contracting Centers, the Contracting departments of the Naval Supply Centers, and their supporting, or satellite, activities. APADE is a standardized procurement system that applies automated data processing to the procurement process. The services of APADE are, however, limited due to cost justifications and proximity to larger contracting and procurement facilities. Many smaller facilities lack computer assisted operation and are not supported by APADE.

B. OBJECTIVES

The research objective is to formulate and design a prototype micro computer application for small Naval Field Contracting System (NFCS) procurement activities (five personnel or less). The prototype will facilitate the performance and management of the procurement process, emulating the Navy's Automated Procurement and Accounting Data Entry (APADE) System.

C. THE RESEARCH QUESTIONS

The primary research question is: Can a modified Automation of Procurement and Accounting Data Entry (APADE) micro computer application be designed for small Naval Field Contracting locations?

Research evolves around providing the main functionality of the APADE procurement system in a micro computer based system to be used by small procurement activities.

Secondary research questions in this area include: What relations can be derived from the APADE flat files; what key elements of APADE should be extracted for the model. Research activities include: Review and evaluation of the APADE small procurement modules and database structure; review of reverse engineering literature; formulation of an efficient reverse engineering methodology; application of a reverse engineering methodology to the APADE system; evaluation, design, and implementation of the prototype micro computer application.

D. SCOPE, LIMITATIONS AND ASSUMPTIONS

The scope of this thesis entails establishing an entity relationship model from mainframe system data. This model is then used in the development of a prototype micro computer application to be used by small procurement activities with five buyers or less.

Since the objective is to emulate the mainframe version of APADE on a micro computer, all research efforts are

localized within the structure and functionality of the APADE system.

E. LITERATURE REVIEW AND METHODOLOGY

A 1986 thesis asserts the benefits of incorporating the APADE system into small procurement activities through communications links [Ref.1]. Our proposal is to step away from the mainframe and/or telecommunication processes and use stand alone micro computers and application software tailored to perform the desired APADE functions.

Initial review on the subject has indicated that:

* Providing APADE mainframe computer services to small NFCS activities through telecommunications is expensive, both in procurement and maintenance costs.
* Off the shelf commercial procurement software is expensive and requires extensive tailoring.

Recent improvements in micro computer hardware and software have greatly enhanced their speed, storage, and networking capabilities. Implementing micro computer procurement systems at small NFCS activities supplants the need for mainframe, or mini computer systems.

Reverse Engineering is a relatively recent software technology which strives to produce well structured, easily maintained programs from existing source code. The overall process generally considers three separate aspects of the problem; restructuring, re-engineering, and reverse engineering [Ref. 2:p. 60].

Restructuring involves conversion of unstructured source code into a cleanly structured version to reduce maintenance costs, improve system quality, and enable program enhancements without changing the functionality of the code. Re-engineering analyzes the source code, defines source structure and identifies data redundancies, non-standard names and unused code. Re-engineering produces documentation, including structure charts, cross reference indexes, and file, record and field relationships. Reverse engineering techniques analyze the original code, synthesize the unstructured data definitions and produce structured data definitions at the specification level. Of the three aspects, reverse engineering is the only one that applies to this research.

"Reverse engineering", as described by James Martin is: ". . . a software technology that has as its goal the migration of old systems into a cleanly engineered form that can be enhanced easily." [Ref. 3:p. 52]

Reverse engineering methods should be capable of synthesizing unstructured data and processing definitions to environment-independent specifications. At this point, forward engineering processes can be evoked which will re-implement the software application. To date this process has only been successful on the data side, and methodologies used for our research include only that side.

Initial research entails a review and analysis of the APADE procurement system through evaluation of relevant APADE program modules. A planned methodology includes examination of: program flowcharts, program data dictionary, user's manuals, training programs, on-line operating APADE program use, and established small procurement guidelines. Reverse engineering techniques are applied establishing an entity relationship diagram from the flat file format presently used by APADE. This diagram is used to design and implement a workable small procurement application.

F. ORGANIZATION OF STUDY

Our approach to the study involves four general areas: A review of DoD purchasing procedures; a description of the APADE system and its functionality; the formation and application of a sound reverse engineering strategy to the APADE process; and the design of a micro computer based prototype which emulates the small procurement modules of the mainframe system.

An outline of the chapters is as follows. Chapter II reviews the historical background to procurement in the Department of Defense and describes current small procurement procedures in detail. Chapter III investigates the APADE system, giving a description of the processes, files and data structure. Reverse Engineering methodologies are reviewed in Chapter IV, and an algorithm is proposed and

applied to the APADE data structure. This process produces an Entity Relation Diagram (ERD) for the mainframe system data. In Chapter V, modules of the APADE ERD, applied to small purchase procedures, are used to create data flow diagrams which model the small purchase functional requirements of the system. These requirements form the basis for the detailed design specifications used in the prototype. This is described in Chapter VI. Finally, Chapter VII reviews the course of the study, commenting on both reverse engineering methodology and aspects of the prototype developed, and indicating additional applications and follow-on work which may expand the effectiveness of the prototype.

II. DoD PURCHASE PROCEDURES

A. HISTORICAL BACKGROUND TO PURCHASE PROCEDURES

The heritage of the Department of Defense (DoD) procurement has been unfolding since the beginning of the Revolutionary War. The concept of the purchase order as a means of procuring material and services was not new. The basic premise of relying on agreements between two parties with regard to future conduct evolved more fully into practice during the 16th century. The birth of federal purchasing regulations began with the authorization to construct and provision ships for the Revolutionary War. Since then, the legislative powers have grown, especially in the direction and oversight of the government procurement process.

B. SMALL PURCHASE CONCEPTS AND APPLICATION

"Small purchase" is defined as ". . . procurement of supplies, non-personal services, and construction in the amount of $25,000 or less." [Ref. 4:Subchapter C, Section 13.101]

Means of providing for small purchase incorporate the use of imprest funds, purchase orders, and blanket purchase agreements.

The importance of the role that small purchasing takes is critical to the operation of DoD activities and forces.

Approximately 90 percent of all DoD acquisition actions involve procurements valued less than the $25,000 threshold. Since these funds are expended through "simplified", less rigid procedures, emphasis must be made to ensure that overpricing and excessive administrative costs for small purchases are controlled.

The small purchase buyer is a skilled and knowledgeable professional. As an official agent of the U.S. Government, buyers are legally and morally obligated to execute their authority to ensure each purchase executed against U.S. Government funds is legally correct and properly obligated. The professional small purchase buyer ensures that all written government and contractor agreements meet all contract legal requirements. Small purchase activity buyers have the tacit responsibility of translating the customer's request for goods and services into a purchase action. These actions require buyers to understand the variety of elements associated with the purchase request format. This procurement expertise is coupled with an ability to accurately evaluate the adequacy of the requirement description. Buyers perform a very important mission as an extension of the customer, turning their requests into the delivery of material or services vital to their mission.

The procurement action begins with the customer request for material. The customer's responsibility involves submitting current, accurate, and complete purchase request

data. This allows the small purchase buyer to accurately identify the requirement in the purchase action. Interpreting the customer's request accurately is critical to any form of procurement. The purchase request submitted by the customer is the vehicle used to identify the material or service desired. It is important to ensure that the customer needs match the buyer's perception of the requirement. Sharing information and providing for free-flowing communication between customers and purchase activity buyers is important for proper purchase action.

C. PROCUREMENT ACTION WORKFLOW

Procurement action workflow evolves around a five step process: receipt and entry of a customer requisition, technical review, assignment to buyer, buyer procurement action, and award. The flow of these processes is shown in Appendix A.

1. Requisition Receipt and Entry

Procurement processing begins with the receipt of a customer's purchase request at the procurement office. Requisitions can be received in the form of hard copy documents or through electronic data transfer. An internal control routing system, overseeing the requisition process, is important to any purchasing activity. This system must provide requisition status information and assign priority action to urgent requests. Routing systems vary depending on the type of activity involved.

Initial screening is another factor in the receipt process. This step is critical to the internal control and routing procedures. Requisitions are sorted by priority and screened for completeness. Preliminary requisition screening entails the evaluation of the following requisition line elements:

* Authorization signature
* Accounting data information
* Priority designator
* Clearances and approvals as required
* Attached specifications, drawings or blueprints

These screening checks may be performed quickly by personnel not directly associated with the formal procurement process, since the action is a repetitive routine. Procurement actions are then consolidated, assigned a Control Number and placed in folders awaiting further action.

2. Technical Review

Following the screening process, the requirement document is submitted to the technical department for review. This review verifies that the required item cannot be crossed to a Naval Stock Number (NSN), checks mandatory screening requirements, or provides additional technical information so the buyer can intelligently interpret the customer's needs.

If the needs are not clear, or the purchase request is incomplete, the requisition is suspended and returned to the customer. Some actions may be corrected through phone conversation negating the requirement for suspension. Again, local policies dictate when, how, and for what reasons correction action must be followed. It is important to understand that this function should be performed as early as possible in the procurement process, allowing the buyers to perform unimpeded.

Technical screening should determine if material and service requirements can be provided through the following sources (by priority):

a. Materials

* Agency inventories
* Excess from other agencies
* Federal Prison Industries, Inc.
* Procurement list of products available from the Committee for Purchase from the Blind and Other Severely Handicapped
* Government wholesale supply sources such as stock programs of the General Services Administration, Defense Logistics Agency, Veterans Administration and military Inventory Control Points (ICPs)
* Mandatory Federal Supply Schedules
* Optional use of Federal Supply Schedules
* Commercial sources

b. Services

* Procurement lists of services available from the Committee for Purchases from the Blind and Other Severely Handicapped

* Mandatory Federal Supply Schedules and mandatory General Services Administration term contracts for personal property rehabilitation

* Optional use of Federal Supply Schedule when mandatory

* Federal Prison Industries, Inc.

* Other commercial sources

c. Products requiring mandatory specified sources

* Jewel bearings and related items

* Public utility services

* Printing and related services

* Automatic data processing and telecommunications acquisitions

* Leased motor vehicles

* Strategic and critical materials from excess General Services Administration inventories

* Helium

This review determines whether or not supplies or services are available through the agency's inventories, General Service Administration, Federal Prison Industries, Inc., Blind or Other Severely Handicapped Industries, Defense Logistics Agency, the military's own inventory control points, and commercial sources. In many instances, this process is streamlined through the use of micro computer data files containing this information. The use of optical disk systems such as "Partsmaster" dramatically improved the ability of the Technical activity to research this information. Should material not meet the mandatory

sources, the customer requisition is passed to the buyer's supervisor for buyer assignment.

3. Assignment To Buyer

A centralized control desk will screen purchase requests for assignment to the appropriate buyer for further procurement action. Procurement assignments are made to a particular buyer or organizational area group. These area groups are identified by Federal Stock Category (FSC) commodity codes. This allows specialization and inherent expertise in specific material procurement categories. It is the responsibility of the buyer's supervisor to assign the requisition workload accordingly. Other important elements include: Workload of each buyer, priority of requisitions and buyer availability (considering buyer personal leave, sick leave, etc.).

4. Procurement Action

This action begins with the assigned buyer reviewing the requisition for completeness and accuracy. Requests with incorrect or incomplete information, that cannot easily be corrected, may be returned to the control desk for action. This action may lead to suspension of the requisition for corrections. If no corrections are required, the requisition process continues.

The buyer selects the optimal method of procurement. Four methods of small purchasing action are available for the buyer. They are: blanket purchase agreement (BPA),

imprest fund, purchase order, and delivery order actions. A brief description of these methods of procurement action is provided below:

a. Blanket Purchase Agreement (BPA)

The Blanket Purchase Agreement is an extremely valuable tool of the small purchase activity buyer. This is a simplified method of meeting repetitive requirements for goods or services by establishing "charge accounts" with various vendors. These vendors must "pre-qualify" and agree to conduct business under the terms of the specified BPA contract. BPA procurement is controlled by establishing a dollar threshold limitation. Procurement dollar threshold limitations are designed to ensure that proper competition guidelines are followed in the procurement process.

Blanket Purchase Agreement files are maintained by the procurement activity. Competition must still be solicited and vendors must be rotated equitably.

b. Imprest Funds

The objective of the imprest fund is to provide for cash payment at the time of purchase for inexpensive items or services. This eliminates the administrative processing time, reduces costs, and ensures prompt receipt of material or service. This action usually involves transactions that do not exceed $150.00 ($300.00 under emergency conditions). Imprest fund orders may be placed

orally or without competition if the prices are considered fair and reasonable.

c. Purchase Orders

The basic objective of the purchase order is to contract for goods and services which are over the dollar limitation of the BPA or if the final price of the procurement is not yet determined. Because of the flexibility of the purchase order, it is frequently used. The purchase order is undoubtedly the safest small purchase method available. Items or services requested are fully identified in writing, and if the price is known, fully priced. Unpriced purchase orders require that estimates be realistically provided and reviewed upon receipt of the invoice.

d. Delivery Orders

Delivery orders allow the buyer to use existing contracts established by the General Service Administration and other contracting agencies. Existing contracts provide a ready source of materials or services by pre-establishing possible sources, prices, and delivery terms. Delivery Orders are most commonly encountered when using the Federal Supply Schedules (FSS) as a mode of supply.

The second major step in the buyer action is the issuance of a Request for Quotations (RFQ)/Solicit Oral Quotations (SOQ). Sources identified through the technical review process or by the customer, may be combined with

additional sources gathered by the buyer. These may be used to solicit oral price quotations. Other information obtained during this process are: the responsiveness of the source to a Government procurement, delivery dates of material, available vendor discounts, shipping information, and special handling requirements. This information is used in the evaluation and award process.

Although small purchasing regulations invoke simplified procedures for administering a small purchase contract, documentation of buyer action in the solicitation process is paramount to proper procurement action. The simplified procedures in no way reduce the necessity for comprehensive documentation.

Documentation contained in the buyer's worksheet, detailing all transaction information, is used in preparation of the final procurement contract.

5. Award

After the solicitation and review in the procurement action process, buyers prepare draft sheets containing all necessary information used for contract preparation. The documentation information is used in the formal requisition document preparation. The formal requisition document contains customer requirements and information gathered during the solicitation process.

This requisition contract is printed and submitted for review to the designated authority. This individual is

knowledgeable of all aspects of the requisition process, including regulatory requirements. Proper use of this review process is necessary to ensure that procurement award documents are accurate, complete, and correct. This process maintains oversight, ensuring that fair and reasonable prices are obtained through competitive means.

Following review, the requisition award document is signed by the authorized authority and distributed. Distribution generally includes: the vendor awarded, the paying activity, the administering activity, the customer, and document file copy. The document file copy is filed together with all supporting documentation until all actions under the contract are complete. This action completes the buying activity procurement process.

D. PROCUREMENT AUTOMATION IN DoD

New computer and information technologies are now available to take the burden of massive paperwork from the shoulders of contracting officers. This gives buyers and contract administrators the ability to share information rapidly and inexpensively between themselves and with material suppliers. Buying can become efficient and effective.

Over the past 20 years, the DoD has automated many procurement functions in an attempt to reduce procurement costs, reduce Procurement Action Lead Time (PALT), and improve material receipt time. These efforts, however, were

developed separately by different services, with little coordination between them, and directed to high volume purchasing activities.

> However, its (DoD) automation projects are usually local or command-level initiatives that support unique requirements.....they do so in isolation and thus neither benefit from previous experience nor share their capabilities with others. [Ref. 5:pp. 2-5]

Major DoD automated procurement systems are listed in Table 1.1. The Army's SAACONS is the only attempt to create a Service-wide automated procurement system. All other DoD branches have developed and implemented their own activity level contracting system. Only the Defense Logistics Agency (DLA) lacks a totally modernized installation level contracting system.

The Navy has two major procurement systems, APADE and PED, and numerous local activity and smaller command procurement systems.

> The large number of diverse automated procurement systems in the Navy is a function of its decentralized procurement structure. [Ref. 5:pp. 3-13]

Automation of Procurement and Accounting Data Entry (APADE) is the Naval Supply Systems Command (NAVSUP) automation initiative for major contracting activities of the Navy Field Contracting System (NFCS). Today's APADE has evolved from previous designs, beginning with small purchase systems. It now supports large contracting capabilities. An overview of APADE is given in the following chapter.

TABLE 1.1

MAJOR AUTOMATED PROCUREMENT SYSTEMS

TYPE OF CONTRACTING	SYSTEM	SERVICE/ AGENCY
Operational Contracting (base/regional)	BCA	Air Force
	SAACONS	Army
	APADE	Navy
CENTRAL SUPPLY (wholesale)	CDMS	Air Force
	IPS	Army
	DPACS	DLA
	PED	Navy
WEAPON SYSTEM	AMIS	Air Force
	IPS	Army
CONTRACT ADMINISTRATION	AMIS	AIR FORCE
	MOCAS	DLA

DoD's automation of procurement is redundant even though features are comparable, since all must adhere to the same basic procurement concepts and legislative requirements. Benefits resulting from sharing technology are not being realized and standardization of data requirements, processes, and formats are needed.

> The Services are emphasizing future development in system integration instead of system wide standardization. [Ref. 5:pp. 2-5]

III. APADE SYSTEM

A. GENERAL DESCRIPTION

APADE provides preaward processing of requirements and contract actions, solicitation and purchase/delivery order preparation, postaward contract management, status inquiry, report generation, and system support including help and Computer Aided Instruction (CAI). APADE uses micro computer terminals with preformatted menu-driven screens and is co-hosted to a mini computer cluster within NAVSUP's logistics information network called Stock Point Logistics Integrated Communications Environment (SPLICE). The system provides interoperability and interconnectivity with other Naval logistics systems through SPLICE.

The U.S. Navy has implemented APADE throughout the Navy Field Contracting System (NFCS) to provide support to Naval Regional Contracting Centers, the Contracting Departments of the Naval Supply Centers, and their supporting, or satellite, activities. APADE is a standardized procurement system that applies automated data processing to the procurement process.

Table 1.2 shows the capabilities of the APADE system. The APADE system is divided into four distinct areas: Small purchase, system interfaces, contract management, and large purchase capabilities. Since the Navy centralizes its

contracting operations, APADE is used at the large regional contracting and supply centers. ". . . APADE is scheduled for the 35 largest activities in the NFCS, each with unlimited contracting authority." [Ref. 5:pp. 4-14]

TABLE 1.2

APADE CAPABILITIES

CAPABILITIES	STATUS
CONTRACT PLACEMENT:	
Requirement Processes:	
Receipt -	Automatic and CRT
Validation -	Yes
Coordination -	Yes
Consolidation -	Yes
Procurement File Preparation:	
Electronic File -	Planned
Automatic Referral -	Yes
Planning:	
Buy Price/History -	Yes - local only
Sources -	Yes - local only
Expert Advice -	Planned
Synopsis Preparation -	Word Processing
Synopsis Transmission -	Yes
Solicitation/Contract Preparation:	
Model Contract Preparation -	No
FAR (on line) -	Planned
Clause Selection -	Planned

CAPABILITIES	STATUS
CONTRACT PLACEMENT:	
Requirement Processes:	
Bidder's Mailing List -	Planned
Electronic Bid Board -	No - local plans
Electronic Solicitation -	No
Bid/Proposal/Negotiation Analysis:	
Proposal Evaluation -	No
Source Selection -	No
Price Analysis -	Spreadsheet
Past Performance -	No
Bid Abstract -	Yes
Negotiation Memo -	Word Processing
DD Form 1547/1861 Preparation -	Planned
Award:	
Business Clearance Review -	Yes
Debarred/Suspended Check -	Yes
DD Form 350/1057 Preparation -	Yes
DD Form 350/1057 Edit -	Yes
DD Form 350/1057 Transmission -	Yes
Electronic Distribution -	No
CONTRACT MANAGEMENT:	
Action Tracking -	Yes
Workload Management -	Yes
Contract Summaries	Yes
Undefinitized Actions	No
Contract Closeout -	Planned
Delivery/Performance -	Yes
MILSCAP -	Planned

APADE provides document control, buyer support services, automated document preparation, information storage and retrieval, automated interface capabilities, and a wide range of current and accurate management information.

B. FUNCTIONAL PROCESSES IN APADE

The APADE system is divided into six discrete functional processes:

* Requisition Input Processing
* Buyer Support Processing
* Contract Administration Processing
* Contract Completion Processing
* Inquiry Processing
* Report Processing

While the first four processes occur sequentially with regard to any particular requirement, all six processes are, or can be, in process at any time that the system is in use. Small purchase procedures require, or may require, the use of the first, second, fifth and sixth processes. The following paragraphs provide a summary of the main functions of the APADE system.

1. Requisition Input Processing

The procurement process begins with the receipt of a customer requisition. This requirement may be in the form of any of six recognized Naval requisition documents or may also be submitted in machine-processable form. Requisitions which contain more than one line item are entered and

processed as a series of single line records. This feature does not preclude customer follow-up by requisition number. Records are then edited for correctness and completeness. Items which cannot be corrected immediately by the clerk are stored in a suspense file until additional or corrected information is received. Items are then screened to determine if they can be satisfied by an Indefinite Delivery Type Contract (IDTC), a form of delivery order. If such a source is available, the system completes the action by generating a delivery order. If the item cannot be obtained by an automated delivery order, the system then collects all data pertinent to the item such as: item description, price history, ship-to-data, accounting data, and a list of potential sources. The system then assigns a purchase request (PR) number, makes a preliminary recommendation as to whether large or small purchase procedures should be followed, and assigns the requirement to a specific buyer. The PALT clock begins at this time, registering the start of the total time required to complete the purchase request.

2. Buyer Support Processing

The buyer's first task is to review the data accumulated during the input process and determine its adequacy. In the event the buyer notes a deficiency in the information, he or she may refer the requirement, as appropriate, to the in-house technical activity or to the requesting activity. All data remains intact and the system

reflects the status of the referral, including the reasons for referral, the date referred, and the date removed from referral status. These dates are used at a later point to adjust the PALT, if necessary. Referrals may occur at any time during the procurement process. Next, the buyer must determine how the requirement will be satisfied. At this point a delivery order may be initiated against an IDTC. The buyer may also place an order against a Federal Supply Schedule (FSS). Existing contracts may also be modified to satisfy the requirement. If the requirement does not exceed the large purchase threshold, the buyer may have the system generate a purchase order, record a blanket purchase agreement (BPA) call, or record an imprest fund buy. Information with regard to the existence of IDTCs, FSSs, and BPA's is available within the system and may be requested by the buyer.

Large purchase requirements necessitate a more formalized approach. For large purchases which can not be satisfied by an IDTC or an FSS order, the buyer determines whether a milestone plan should be established for the purpose of monitoring the buying process. Milestone plans have standard checkpoint dates. These dates may be revised, as necessary, and the system will maintain the planned date as well as all revisions. The system also records the number of times any milestone date is revised and the completion date of each of the milestones. There are 24

established milestone plans, each of which establishes a purchase path for the requirement. Large purchase procedures can be quite lengthy. Items may be solicited orally or in writing, or negotiated. For either negotiations or solicitations, additional worksheet information must be gathered. If the solicitation is oral, the buyer need only note the Federal Supply Codes for Manufacturers (FSCMs) of the sources called and dates of the solicitations. In other cases it is necessary for the buyer to input additional information. The buyer will then perform a formal solicitation to the general public or to identified potential bidders. The system, through the word processing functions, produces notifications such as Commerce Business Daily (CBD) synopses. The system also maintains a bidders mailing list which contains information on interested contractors as well as flagging the debarred, ineligible, and suspended contractors. The system records the bidders to whom the solicitation was provided. In the event that a solicitation must be modified, the buyer inputs the necessary information and the system produces the amendment and develops the mailing list based on the distribution applicable to the solicitation.

The preparation of award documents parallels that of solicitations. The buyer inputs the required data, selects the appropriate clauses, if required, and the system

prepares the full range of documents for the award and for information to all concerned.

3. Contract Administration Process

Information entered into the APADE system during the requisition input process and during the buying support process remains available on-line to facilitate the contract administration process. During this process, the system will accept information with regard to performance and/or payments, prepare or record modifications, and will provide contract monitoring at any level selected by the activity.

4. Contract Completion Process

The contract completion process includes the recording of the contract completion statement, DD Form 1594, when one is required or desired by the activity; the preparation of locally-designed completion statements for all other contracts; the preparation of a skeletonized APADE record; the transfer of the total APADE record to the activity archives; and the final deletion of the skeletonized record from the active system.

5. Inquiry Processing

APADE has been designed as an on-line system in which users have immediate access to stored information. Access is provided to all work in process and all support files, and facilitates the flow of information for several essential purposes:

* Procedures are provided so that inquiries made by or for customers result in prompt and accurate status information.

* Procedures are provided so that inquiries made by or for buyers or contract administration personnel result in information on which to base operational decisions.

* Procedures are provided so that inquiries made by or for supervisors result in information on which to plan work and assign resources.

While file security requires that the update capability be carefully restricted, inquiry capability is restricted only in specifically defined circumstances, such as abstracted bid data prior to award or limitations placed on terminals installed in customer activities.

6. Report Processing

The APADE system includes procedures for the preparation of all reports required by higher, external, authorities. Internal reports, required by each activity to monitor and manage its operations, are not explicitly defined. The APADE system provides each activity the capability to design, produce and update a wide range of reports whose content and frequency may be tailored to local requirements.

All external reports are produced in hard copy. Selected reports are also prepared in machine-processable form for transmission to the Naval Material Command (NAVMAT) headquarters. The report generator feature of the APADE system allows the local activity to design real-time or repetitive reports. Repetitive reports may be run on a

scheduled basis, upon request, or upon the occurrence of a defined condition.

C. FILES AND DATA STRUCTURE

The APADE system uses ten files, containing 152 data elements and six "look-up tables". The tables are simple structures, accessed by a key, which provide only one or two elements of data.

The elements used in the APADE system have standard structures. They are either character strings or numeric. We provide in the remainder of this chapter a description of the files used in APADE and their data elements. Appendix B provides a detailed listing of these files and their data elements.

1. APADE Support Files

a. Record File

The APADE Record file is used as a catchall file. During the life of a transaction, all data pertinent to the requisition is stored in the Record file. It is a shell which contains virtually all of the elements utilized in the system as they pertain to a given transaction.

Input of any of the six standard Navy documents initiates a record in the Record file. Each of these documents has a unique format. All input documents, however, contain several mandatory elements. Some of these elements are:

* Document Number or Requisition Number
* Unit Identification Code (UIC)
* Priority
* Fund Code
* Line Item Number
* Quantity
* Unit of Issue
* Unit Cost

The growth of the Record file record during the buying process will vary greatly depending on the type of purchase action. If the requirement can be satisfied simply, such as by placing a delivery order under an IDTC, only the information needed to prepare and release the delivery order would be added to the record. At the other extreme, the record could grow to contain some or all of the following:

* Information on consolidations with another PR
* The number, reason, and duration of any referrals
* The establishment and accomplishment of up to 50 milestone events during the buying process
* The drafting and approval of a Determination and Findings (D&F)
* The preparation of a Request for Authority to Negotiate (RAN)
* The preparation of Commerce Business Daily synopses (CBD)
* The preparation and distribution of a Pre-Invitation Notice (PIN)

* The preparation and distribution of a solicitation, including applicable clauses
* The reason for, and content of, any amendment of the solicitation
* Bid abstract information
* The request for, and results of, Pre-Award Surveys
* Data regarding all necessary approvals
* The award document including applicable clauses.

Since the field names from all six types of input documents are common and the field structures are similar, elements of the record file in Appendix B are a composite of the fields in all of the available input documents.

b. Clause File

The clause file has two purposes. It automatically provides the clauses required for solicitations and contracts, and establishes a source of information. The development of any procurement document requires certain standard information. The data provided in this file cover a wide range of elements, all of which define the circumstances of the specific procurement. As the system processes requisition data to generate a procurement document, it notes those elements identified in the system as affecting the clauses required. After having noted all elements that affect clause selection, the system will identify the pattern of these factors to a selection indicator. The selection indicator identifies all standard,

mandatory clauses which must be included in the document. These clauses are extracted from the file and printed in the document, or incorporated by reference, as applicable. Buyers may also append non-mandatory clauses to the record file. Clauses are keyed by the clause number, and input of any appropriate clause number will append the correct information.

c. Ship To File

This file provides an address or information to be used to obtain an address for the shipment of material being purchased. Specific shipping instructions, if any, are also available from this file. The Ship To file utilizes the Unit Identification Code (UIC) as a key.

d. Price History File

This file provides information to the buyer concerning prices paid for the same item purchased at an earlier date and identified by the key, NSN_LSN (National or Local Stock Number), which identifies an item description in the record file. The Price History file contains all fields necessary to describe a historical contract.

e. IDTC File

The IDTC file provides the data necessary for the production of delivery orders against the contracts contained in the file and information in support of the procurement process.

The key for this file is the commodity indicator (Commod_ind) which matches the Federal Stock Category (FSC) field in the record file. The IDTC file contains all contract data necessary to produce a delivery order through the Autobuy file.

f. Commercial Source File

This file has three purposes:

* to provide Bidder's Mailing Lists for solicitations, etc., including proper rotation procedures,
* to provide information to buyers and management personnel,
* as a safeguard to preclude the automatic issuance of a solicitation, processing of an offer, or award of a contract to a firm that is on the Joint Consolidated List of Debarred, Ineligible, or Suspended Contractors.

Sources are identified by a concatenated key of commodity indicator and FSCM, which uniquely matches a source to a commodity.

g. Accounting File

The purpose of the Accounting file is to translate fund codes provided by the requisitioner to the accounting information required in the preparation of an award document.

The Accounting file is accessed through the key cost code, and contains all fields necessary to provide full accounting data for the requisition.

h. Buyer File

This file provides in-the-clear identity of the buyer whose buyer code appears in the acquisition

responsibility field of a purchase request and information which may be copied when a solicitation or award document is printed.

The Buyer file contains the key, buyer code, and all data which fully identifies a specific buyer.

i. Automated Buy File

The purpose of the Automated Buy file is to provide the data necessary to produce an automated delivery order for certain customer requirements, during the requisition input process. The key field for this file is a concatenation of the contract number, which matches the number in the IDTC file, and the NSN_LSN, which specifies the unique item in the IDTC contract. All data fields necessary to complete a delivery order are included in this file.

j. Purchase Item Description (PID) File

The file has three purposes:

* to provide the buyer with as complete a description of the item as is reasonably possible
* for inclusion or extraction for inclusion in solicitation and/or contractual documents
* to reduce the number of items which require extensive and possibly repetitive technical review

The key field for this file is the NSN_LSN.

2. General Observations

The files of the APALE system do not lend themselves to a relational database system. Many of these files are not normalized, therefore creating serious update problems

if moved without modification to a relational database environment. In order to engineer an effective and efficient relational database system, the APADE file structure will have to be modified. This will be addressed in Chapter IV.

It should be noted that the size of the APADE system and its capabilities does allow these data structures to effectively and efficiently process huge volumes of purchase requests. Volume alone is not adequate justification for the redundancy inherent in this system, but the speed and capacity of the mainframe eliminates foreseeable delays in processing.

There are several common field names used throughout the support files. In most cases, these elements also share a common field structure; however, there are some distinctive differences:

a. Many data elements in different files have different names, such as, Descr or Item_descr.

b. The structure (length and type) of elements in different files varies. An example would be the description file which ranges in size from 50 to 2000 characters, or is a memo field of indeterminate length.

c. The level of abstraction in different files could differ also. For example, one field can contain both the name and address, or names and addresses can be in separate fields.

The next chapter will investigate the data in the above files. By an application of reverse engineering methodology, this data will be transformed into set of relations which can be used to construct a high level entity relationship diagram.

IV. REVERSE ENGINEERING AND THE APADE SYSTEM

A. BACKGROUND

In most automated organizations, there exists systems which were developed before the advent of today's principles of good database design. These systems usually contain large amounts of unstructured old code. Traditional attempts at maintaining such systems have been unsatisfactory and expensive, since most of the data is redundant and unnormalized.

One way to overcome these problems is to "reverse-engineer" these systems into a cleanly structured form. This is accomplished by converting old code into new, more evolutionary software which will have the capability to be continually modified and its design improved. With a new system, built using good design principles, maintenance problems are greatly reduced.

Our research analyzes the data side of the existing APADE system. Using reverse engineering techniques, the existing data structure is converted into a high level conceptual model which is then used in the design of a micro computer database application.

B. REVERSE ENGINEERING APPLIED TO APADE

1. Justification

The APADE system consists of flat files. These files are accessed in a non-hierarchical manner, that is, they are used as needed and not in any particular order. Critical identifying fields are "keyed", although the mainframe "keys" are simple record identifiers and do not operate as keys in a relational database sense. As shown in the preceding discussion of the APADE files, many fields provide identical information, but are named and structured differently. Changes to data in this system would require an exhaustive search of all files to ensure that the update is completely processed.

The intent of our investigation is to apply APADE-like processing in a new environment, adapting the data to an application which will run efficiently and effectively on micro computers.

2. A Methodology for Reverse Engineering

Our question then, is how to apply current theories of reverse engineering? Our intent is to design a relational database system which will emulate APADE's functionality. Our goal is to create a viable conceptual view of a micro computer based APADE system through an application of reverse engineering principles. Only data is analyzed within the scope of this thesis. To accomplish this, we construct a high level conceptual model (Entity

Relationship Model) from existing data file definitions, identify data redundancies and non-standard names, and restructure the existing files into a normalized set of relations. This Entity Relationship diagram (ERD) is the basis for the design of a prototype for the small purchase portion of the APADE system. Standard "forward engineering" design techniques are applied to the resulting ERD to produce a relational schema for the prototype.

In order to reverse engineer the APADE data in a systematic manner, an algorithm was formulated which transforms flat files into a high level ERD.

The algorithm consists of two phases. Phase I constructs a relational schema (map) from the existing flat file system. Phase II transforms the relational map into a high level ER diagram.

Phase I: Derive a Relational Schema

Step 1. Isolate All Files and Identify Data Elements

Review the mainframe functional documents and extract all file information. For each file, identify every attribute and its structure (length, type). Identify data elements that represent the same fact and reconcile naming and structural differences, if any.

Step 2. Group Related File Attributes Into a Preliminary Set of Relations

For each existing file, identify preliminary groupings of attributes. Synthesize data

attributes based on a common theme or belonging to the same entity, such as, accounting data, item description data, etc., to produce a preliminary set of relations. Identify preliminary primary and foreign keys.

Step 3. Identify Data Dependencies in the Preliminary Relations

Within each preliminary relation, identify functional, multi-valued, partial and transitive dependencies between attributes.

Step 4. Normalize Relations

Normalize each preliminary relation, creating as many new relations as necessary. Identify and isolate all candidate and foreign keys and then select an appropriate primary key for each relation.

Step 5. Establish a Relational Map and the Degree of Relationship

Review the resultant relations and establish a logical relational mapping for the system. This is accomplished by explicitly indicating the relationship (by drawing a line) between relations which are implicitly related through foreign keys. Additional foreign keys are identified to connect the new relations. Duplications of data, if present, should be eliminated, or data should be renamed. The relational map is then examined and preliminary degrees of relationship are established (one to one, one to many, many to many). Foreign keys linking the relations are identified in the following manner:

* Where one to many relationships are suggested, the primary key from the relation on the "one" side is identified as a foreign key in the relation on the "many" side, if it is not included already.

* For one to one relationships, the primary key for either relation is included as a foreign key in the other relation.

* If a logical link between relations suggests a many to many relationship, an additional relation is created with both primary keys of the original relations as a concatenated primary key for the new relation. This situation could occur if there is missing information in the flat files.

Phase II. Construct An Entity Relationship Diagram

Construct an Entity Relationship Diagram utilizing a reverse approach to the ERD-to-Relational mapping described in "Fundamentals of Database Design" [Ref. 6, pp. 329-330]. This phase is implemented in eight steps as follows:

Step 1. For a relation R, with a primary key which is a concatenation of the primary keys of "n" relations, n >2, transform relation R into an "n"-ary relationship. Include non-key attributes of R as attributes of the relationship.

Step 2. For a relation R_1, which has a composite primary key consisting of a primary key of relation R_2 and an attribute A, which is multi-value dependent on the primary key of R_2, include A as a multi-valued attribute of the entity corresponding to R_2.

Step 3. For a relation R_1, whose primary key is a concatenation of the primary keys of two other

relations, transform the relation into a relationship. Include non-key attributes of R_1 as attributes of the relationship.

Step 4. For each one to many (1:M) relationship between two relations, create a binary 1:M relationship type. Ignore this step if one of the relations has been transformed into a relationship in a preceeding step.

Step 5. For each one to one (1:1) relationship between two relations, create a binary 1:1 relationship type. Ignore this step if one of the relations has been transformed into a relationship.

Step 6. For relation R, if all or part of its primary key, K_1, consists of a primary key, K_2, of another relation, transform the relation into a "weak" entity, that is, an entity whose existence fully depends on another entity.

Step 7. For each remaining relation, transform the relation into a regular entity.

Step 8. Examine each relationship to determine total or partial participation. Total participation of an entity in a relationship means that the existence of the entity depends on its participation in the relationship. Partial participation implies that the entity may or may not participate. For our ER diagram, total and partial participation is indicated by representing a minimum

and maximum range of participation of each entity in a given relationship. An entity with partial participation will have a minimum constraint of "0", while an entity with total participation will have a minimum of "1", meaning that at least one instance of participation must exist. Maximums may be indicated by a fixed integer, such as "6" which means that at most six instances of participation may occur, or by utilizing the indefinite "M" constraint meaning that an unspecified maximum exists.

The above steps provide a systematic approach to defining a high level ER model from an existing flat file system. In the following section, we apply them to the APADE system.

3. Application of the Methodology to APADE

For the purposes of continuity during the analysis, as files are split into new relations during normalization, new file names will simply be concatenations to the original file name. At step (5) of Phase I, new names will be assigned to these relations.

Phase I. Derive a Relational Schema

Step 1. Isolate all APADE files and Identify Data Elements

All APADE files were isolated and defined in Chapter III and a complete breakdown of the data elements of each file is shown in Appendix B. Where data elements seem to conflict, i.e., identical elements are structured differently, or elements used for the same

purpose are named differently, they are restructured or renamed appropriately. For instance, the item description fields throughout APADE are intended for the same purpose, however, they have different structures within each file, or are named differently.

Step 2. Group Related Attributes into a Preliminary Set of Relations

The first file discussed is the APADE Record file. The analysis of following files will be accomplished in the same manner as shown for the Record file. For purposes of brevity, only the Record file will be analyzed in detail here. The remaining files will be discussed and detailed results of the analysis of all files will be included in Appendix C.

(1) Record. Taken from the data displayed in Appendix B, the APADE record is structured in the following format. Attributes are listed alphabetically and keys are underlined.

RECORD(<u>Doc_Req_No</u>, ACRN, APL_No, Appropriat, Auth_Acct, Bur_Cont_No, COG, Color, Cost_Code, Descr, Dist, DMS, EIA, Est_price, FSC, FSCM, Fund_code, Fund_expdt, Line_Item_no, Mf_cat, Mod, NSN_LSN, Obj_class, Pr_no, Priority, Proj_code, Prop_acct, Pt_no, Qty, RDD, Serial, Series, Sig_Code, Size, SOS, Sub_allot, Subhead, Supp_add, Tech_man, Tech_ord, Transactio, UIC, Unit_iss)

The first task in this portion of the analysis is to decide on the groupings of related attributes. The attributes of this file describe three entities: the requisition itself, the funding for the

requisition, and the items requisitioned. Once we establish these preliminary groups, we separate the Record file into three preliminary relations. The resultant new relations are:

RECORD(<u>Doc_Req_No</u>, Dist, Fund_expdt, Fund_code, Pr_no, Priority, Proj_code, Sig_code, SOS, Supp_add, RDD, UIC)

RECORD/item(ACRN, APL_No, COG, Color, Descr, DMS, <u>Doc_Req_No</u>, EIA, Est_price, FSC, FSCM, <u>Line_item_no</u>, Mf_cat, Mod, NSN_LSN, Pt_no, Qty, Serial, Series, Size, Tech_man, Tech_ord, Unit_iss)

RECORD/acct(Appropriat, Auth_acct, Bur_Contr_no, Cost_code, <u>Doc_Req_No</u>, <u>Fund_code</u>, Obj_class, Prop_acct, Sub_allot, Subhead, Transactio)

Preliminary primary keys are underlined. The preliminary foreign key for the last two relations is Doc_Req_No.

(2) Clause File. The Clause file contains only three related attributes, with the Clause number as the key.

(3) Shipto File. The Shipto file contains five attributes with UIC as the key. These attributes are unique to each receiving command and the UIC uniquely identifies them.

(4) Price History. The PRICEHI file provides background information and pricing data for the buyer. There are two basic groupings of information available from this file;

* The information on the contract
* The information on the items in the contract

The key for this file is the NSN_LSN, which uniquely identifies an item, however, it will not uniquely identify a contract. Also, in APADE, the NSN_LSN is an optional field. Small purchase actions are for non-standard merchandise. The NSN_LSN is a government assigned stock designator and is used for standard requisitioning. The NSN_LSN is not always available for an item. For the data which describes the contract, we have chosen Contra_No as the preliminary primary key for the contract information and a concatenation of Contra_No and Nomenclatu as a better preliminary primary key for the data which describes item information. The item group should also include Contra_No as a preliminary foreign key. It should be noted for further analysis that the Contract number specified here is a historical number and bears no relationship to the Contract number to be awarded to the active purchase transaction.

(5) IDTC file. The IDTC file is a reference contract for current items. If a buyer wishes to procure an item from the IDTC, an automated buy is generated. The automated buy (discussed later) uses data from the IDTC for a new purchase. The preliminary review of the IDTC file indicates the following groupings:

* Information on the IDTC contract; and
* Information on the items in the contract.

The preliminary primary keys are Contr_nr for the data which provides contract information; and a concatenation of Contr_nr and Nomenclatu for the item information.

(6) Source file. This file contains information on potential sources for requisitioned items. The data contained in this file suggests two divisions of information:

* Information on potential sources for required items; and
* Information on the current status of the contractor.

Sources are identified by the contractor's name and address, a single attribute in the Source file, which is the ideal preliminary primary key. The other aggregation of information concerns the contractor's status. This information determines whether the vendor is eligible for consideration. Status is the preliminary primary key for this relation and should be retained in the source relation as a foreign key.

(7) ACCT File. The ACCT file describes the accounting data which obligates funds for payment of the contract. The file contains eleven elements to describe this transaction, with UIC and Costcode as the key. An initial review of the file indicates no reasonable

division of information. This file uniquely describes the accounting function.

(8) Buyer file. The Buyer file contains information about the buyers assigned to the purchase shop. It has contact information as well as buyer background information. The buyer background information, consisting of the buyer's occupational series and paygrade, is contained in only two data elements. Since all data is keyed to the buyer code, as the preliminary primary key, no further sub-groupings are necessary.

(9) Autobuy file. The Autobuy file is activated whenever an IDTC file provides a source for a purchase action. When this happens, elements from the IDTC file and the Record file will be combined with the elements of the Autobuy file to produce a delivery order. The Autobuy file is keyed to the Contract number. In all respects this file emulates the IDTC file and should therefore be restructured in the same manner.

(10) PID file. The last file in the APADE system provides additional descriptive data to the record file. The key for this file is the NSN_LSN element. This key uniquely describes the remaining attributes.

Step 3. Identify Data Dependencies

Here, we review the preliminary relations to ensure that the functional dependencies are logical and that no partial or transitive dependencies

remain. Five relations are not reviewed below: Autobuy, Buyer, Clause, Accounting, and PID. The Autobuy relation contains the same information as the IDTC relation. Any functional dependencies described in the IDTC relation apply to Autobuy. The Buyer, Clause, Accounting, and PID files are small, already normalized, files.

(1) Record File. The new RECORD relation attribute, Doc_Req_No, becomes the primary key. Each of the other attributes are dependent upon this key. No partial or transitive dependencies exist. In the RECORD/acct relation, the attributes, Fund_code and Doc_Req_No constitute the primary key. This concatenation could create several update anomalies since all other attributes can be determined by fund code alone. RECORD/item, with the primary key of Doc_Req_no and Line_item_no, should determine all items requisitioned. All functional dependencies for this relation are shown in Appendix C.

(2) Shipto File. In the Shipto relation, the primary key for the new Shipto model remains UIC. This uniquely determines all information on the receiving activity.

(3) PRICEHI File. The Price_hi relation contains the primary key, Contra_no and Nomenclatu which uniquely determines all other attributes. Similarly, for Price_Hi_Contr, the primary key Contra_nr, uniquely

determines all other attributes. No further partial or transitive dependencies exist for these two relations.

(4) IDTC File. For the IDTC relation, the primary key is Contr_nr. A transitive dependency exists as shown in Appendix C. The IDTC_Item relation has a concatenation of Nomenclatu and Contr_nr as a primary key. This concatenation produces no partial transitive dependencies.

(5) Source. In the Source relation, Commod_ind is multi-value dependent on the primary key, Name_add. Also, the non-key attribute, Status, uniquely determines Deb_in_sus and Explanatio.

Step 4. Normalize Relations

Based on the functional dependencies of the previous step, we review the proposed relations and determine the correct normal form. The goal for the this normalization is to achieve third normal form. In the final relational schema, where transitive or partial dependencies remain, the decision to retain an un-normalized relation is based on the frequency of expected change in the data. If data is not expected to change, or rarely changes, the potential for anomalies will be minimal.

(1) Record File. The Record relation is already in third normal form and Doc_Req_No is its primary key. The Record/Acct relation is also in third normal form. However, the data in this relation exactly

duplicates that of the Acct File. This relation is unnecessary and is eliminated. The Record/Item relation, as analyzed in the previous step, contains several partial dependencies. Decomposing this relation into new relations, based on the dependencies shown in Appendix C, eliminates the problem. Therefore, the Record/Item relation is split into three additional relations as shown below.

RECORD/item(Doc_req_no, Line_item_no, Descr, Qty, Unit_iss, Est_price, Color, Size)

RECORD/item_descr(Descr, FSC, APL_No, COG, FSCM, NSN_LSN, Mod, Pt_no, Serial, Series, Tech_man, Tech_ord)

RECORD/mfr_info(FSCM, Mf_cat)

RECORD/mfr_item(Descr, FSCM, FSC, Mod, Pt_no, Serial, Series, Tech_man, Tech_ord)

(2) Clause. The Clause file needs no alteration. Since there is only one key which uniquely identifies the other two attributes, there are no transitive or partial dependencies and the relation is in third normal form.

(3) Shipto. The Shipto relation has a single, un-concatenated key and is in third normal form.

(4) PRICEHI. Both the Price History relations are in third normal form. For the Price_hi relation, the primary key is a concatenation of the Nomenclatu and Contra_no attributes. The primary key for

Price_Hi_Contr, Contra_no, is a primary key in this relation and a foreign key in the Price History relation.

(5) IDTC. The IDTC relation is not normalized. Removing the transitive dependency shown in Appendix C normalizes the relation. The transitive dependency is used to create a new relation. However, Record/mfr_info was created during the Record file decomposition. Since the key for Record/mfr_info also determines Contr_name/add, it is added to the Record/mfr_info relation. This removes any partial or transitive dependencies. Contr_nr becomes the primary key for IDTC and a concatenation of Contr_nr and Nomenclatu becomes the primary key for the IDTC_Item relation.

(6) Source. Both Source relations are normalized. There are no partial or transitive dependencies. Therefore, the candidate keys become our choice for primary keys. The attribute Status is also a foreign key in the Source relation.

(7) ACCT. The ACCT file is also in third normal form, since the primary key, Fund_Code and UIC, uniquely determines the remaining attributes.

(8) Buyer. The Buyer relation is in third normal form. The primary key, Buyer_Code fully determines the remaining attributes. No partial or transitive dependencies exist.

(9) Autobuy. The Autobuy relations are normalized in exactly the same manner as the IDTC relations described above.

(10) PID. The selected primary key, Nomenclatu, determines all the non-key attributes, and presents no partial or transitive dependencies. Nomenclatu is a valid primary key and is used as a foreign key, matching the Descr attribute in the Item relation.

Step 5. Establish a Relational Map and the Degree of Relationship

Once the existing files are normalized and primary and foreign keys for each new relation have been determined, the next step is to review the newly structured relations to eliminate redundancy and establish relationships between the normalized realtions. Our original ten files have been restructured to 19 normalized relations. These are shown in Appendix C. In a flat file system, data redundancy is a necessary evil because of the lack of relational structure. Therefore, it is possible that several files used in the APADE system could contain identical data. For example, the APADE Record file contains accounting data which can be accessed in the same manner as the ACCT file data. In the final schema for the system, only one accounting file is chosen. Similarly, the IDTC model contains identical data as that held in the Autobuy model. In fact, most of the autobuy data is taken directly from the IDTC model. All these elements are not

necessary in the Autobuy model. Autobuy is again restructured to reduce the data duplication. To do this, we combine the item description data into a common relation and rename it IDTC/Autobuy_Item. This eliminates the second Autobuy model.

This results in a system data structure with 17 relations representing the data files. These are listed in Appendix D. At this point, we have renamed the relations to better describe their contents. The original names used during the initial breakdown are shown in parentheses. The look up tables mentioned in Chapter III have also been included in Appendix D as fully normalized relations. Note that the Autobuy relations have been restructured to one relation with only autobuy data.

Utilizing the relations from Appendix D, we then prepare a relational map by identifying the degree of relationships between relations. For example, the attributes Buyer_Code and Pr_No are added to the Requisition relation as foreign keys, linking the Buyer, Contract and Solicitation relations.

The final relational schema includes six new relations: Amendment_Contract, Clause_Contract, Cancellation_Rqn, Referral_Rqn, Srce_Commodity, and Contract. The first five relations are created to eliminate the many to many relationship suggested. The last relation, Contract, is added to create

a logical link between relations. On inspection of the Amend and Clause relations, no logical connection presented itself which preserved the "real world" model of the data. Amendments and Clauses are physically entered onto a Contract, not a requisition. Also, a Contract will have additional, necessary, data which is not represented in any of the identified relations. Therefore, a Contract relation is created. This relation has Contr_no as a primary key, and additional attributes of Date of Contract (Dte_of_contrct) and Delivery Date. All other aspects of a contract are available in the other relations. Contract has a degree of relationship of one to many with both the Clause and Amendment relations as well as a one to one relationship with Requisition and Solicitation. The addition of this relation ties all relations together logically.

A mapping of the relations, showing degrees of relationship, is shown in Appendix E and a listing of each relation, with all key and non-key attributes, is shown in Appendix F.

Phase II. Construct an Entity Relationship Diagram

The last phase of this procedure is to transform the relational map into a useable conceptual diagram. Following the eight steps for this phase, the relations from Appendix E are transformed as follows:

Step 1. There is one relation which contains the primary keys of more than two relations as its

primary key. The IDTC/Autobuy_Item relation, with the key Contr_Nr and Nomenclat contains the primary keys of Autobuy, IDTC and Description. Therefore, IDTC/Autobuy_Item is transformed into an "n"-ary relationship and is renamed "Contained".

Step 2. Commod_ind, from the Srce_Commodity relation, is multi-value dependent on the primary key of the Source relation. This attribute is assigned to the Source entity and is identified as a multi-valued attribute.

Step 3. There are seven relations whose primary keys are concatenations of the primary keys of other relations: Price_Hi, Referral_Rqn, Cancellation_Rqn, Clause_Contract, Amendment_Contract, Srce_Commodity and Mfr_Supply. These relations become relationships in our conceptual view. They are renamed as follows:

* Price_Hi -> Supplies
* Referral_Rqn -> Suspends
* Cancellation_Rqn -> Cancels
* Clause_Contract -> Defines
* Amendment_Contract -> Changes
* Srce_Commodity -> Supplied_by
* Mfr_Supply -> Produced_by

Step 4. Seven relationships are created during this step. They are named by the convention that

relationships are read from left to right or top to bottom in an ER diagram. The relationships are as follows:

* Source "Has" Srce_Status
* Item "Described_By" Description
* FSS_Sked "Lists" Description
* Requisition "Contains" Item
* Shipto "Receives" Requisition
* Buyer "Works_On" Requisition
* Description "Detailed_By" PID

Step 5. Four binary 1:1 relationships are created here and named according to the link between relations as follows:

* Requisition "Advertised_On" Solicitation
* Requisition "Purchased_On" Contract
* Contract "Generated_By" Soliciation
* ACCTFile "Funds" Requisition

Step 6. The Item relation can only be identified by the key, Doc_Req_No and Line_Item_Nr. Since Doc_Req_No is the primary key of the Requisition relation, and is part of the required key of Item, Requisition is the identifying owner of Item and Item is transformed into a "weak" entity. The "Contains" relationship becomes an identifying relationship. Both the weak entity and the identifying relationship are identified by a double lined figure in the ER diagram.

Step 7. The remaining relations can now be transformed into regular entities and added to the diagram. The final ER data model, showing the entities and relationships, is shown in Appendix G.

Step 8. The degrees of relationship identified on the relational map are now examined to determine participation constraints for the ER model. We have identified only one "weak" entity: Item. Weak entities almost always have total participation with their identifying relationships. Other entities also require total participation:

* Description must have Item
* Contract must have Requisition
* Shipto must have Requisition
* Buyer must have Requisition

The remaining entities and relationships have partial participation constraints. The identified constraints are added to the conceptual view. With the completion of this step, final relational diagram of Appendix F is transformed into an Entity Relationship Diagram, as shown in Appendix H. For the sake of clarity, the non-key attributes have not been included in the diagram.

The resultant ERD contains all necessary information from the APADE system. It includes the data requirements for all six of the APADE functional areas represented in a high level conceptual diagram.

V. MICRO-APADE FUNCTIONAL REQUIREMENTS

In this chapter we utilize the ERD and relational schema developed in the previous chapter and apply them to the small purchase requirements described in Chapter II. A modified ERD is produced which encompasses only those relations which apply to small purchase procedures. Utilizing this new ERD, we apply "forward engineering" techniques, formulating functional and design requirements for a prototype of the new relational system.

A. THE PROTOTYPE ERD

The ERD derived from the APADE flat file system is utilized as a basis for the design for a micro computer relational database. For the purpose of the prototype, we have decided to emulate the contracting of small purchases as it is currently being done in the mainframe system. As stated earlier, small purchases are those purchases which do not exceed $25,000.00. The ERD in Appendix G encompasses all data represented in the APADE system and, obviously, some of the data does not apply to small purchase processing. Additionally, some of the process applications within the small purchase purview will also not be addressed in our prototype. For instance, we will not address automated cancellation of requisitions, the Federal Supply Schedule file and the PID file. Our intent is to provide a

working model for a buyers daily transactions. With this in mind, the following entities have been eliminated from the prototype ERD:

* Cancellation
* IDTC_Ctr
* Autobuy_Ctr
* Amendment
* Solicitation
* PID
* FSS_Sked

The resultant, modified ERD for small purchases is shown in Appendix I. The functional requirements of the prototype will be based on this representation of APADE data.

The functions desired in the prototype fully emulate, and where possible, enhance their corresponding mainframe processes. The enhancements include a fully automated, paper-less process, where APADE requires a substantial amount of manual input, and a more efficient approach to the buyer worksheet source review process.

B. FUNCTIONAL REQUIREMENTS

The APADE system operates as described in Chapter III. For the input process, a clerk processes the documents. Requisitions with multiple line items have each item entered as a separate record. The requisition number is the same on each record, allowing traceability. Next, requisitions go to a technical review section, a portion of APADE which is

done manually. Based on the nature of the requisition and the estimated total cost of the requested items, a decision as to large or small purchase procedures is then made. At this point the supervisor will assign purchase request numbers and a buyer to the requisition. This completes the input process. The next phase of the small purchase procedure is the Buyer Support Process. Here the buyer accumulates additional technical or descriptive data, if required, and begins to process the requisition through available sources, such as internal schedules and contracts (FSS, IDTC, etc.), or external sources. The external sources include commercial sources, price history files and blanket purchase agreements (blanket purchase agreements are included in the commercial source file under the Agree_ind attribute). The potential sources are appended to the record in the form of a "buyer's worksheet". This worksheet is handled manually at most APADE sites. The buyer then selects the most appropriate source and awards the contract.

We use a series of data flow diagrams to describe the functional requirements of the existing system as a basis for designing the new system. The first diagram is a decomposition diagram which provides an outline of the processes in a hierarchical structure. Next, a context diagram is prepared which defines the scope of the system. The next view is a system diagram which defines the interactions of the major functions of the system. Finally,

mid-level and primitive Data Flow Diagrams (DFDs) are prepared. These diagrams outline the processes required to complete the functionality of the system. They are developed in levels, from a broad view of the system to a detailed view of each process that encompasses a single function. The DFD's for the prototype are included as Appendix J. In our DFD's, we include four system processes: Input, Assign, Pre-Award/Award and Output. The functional requirements for these processes are described below. Each functional requirement includes all aspects of the overall function. That is, all sub-processes are included in the general requirement.

1. Input Process

The input process incorporates all error checks and edits which occur in the mainframe system. If input errors occur, the user then has the opportunity to: correct the error and continue processing; cancel the requisition; or refer the requisition. Requisitions are cancelled if the requisition number duplicates that of an existing requisition. Cancelled requisitions are assigned a cancellation code which identifies the existance of a duplicate requisition. A cancellation code generates a cancellation notice to the requesting activity. Referred requisitions are assigned a referral code identifying the nature of the referral. No further purchase action is taken on referred requisitions until the referral code has been

rescinded. Assignment of a referral code generates a referral memorandum which is sent to the requesting activity. This memorandum identifies the requisition, cites the problem and requests corrective action.

For the duration of the referral, the requisition is kept on the buyer's worklist as a suspended item. Once the corrections have been made, the requisition becomes active and is re-prioritized for workload planning.

2. Assign Process

Once requisitions have successfully passed through the input process, they are queued for review in the assign process. At this point, they are reviewed by the Supervisor, assigned to a particular buyer and then assigned a purchase request number. The Supervisor makes the assignment to a buyer based on current buyer workload. The purchase request number is assigned serially from a list of authorized purchase numbers.

3. Pre-Award/Award Process

When the requisition has been assigned a purchase request number and listed on a buyer's workload, it is ready for small purchase processing. Assigned requisitions go to individual buyer queues and are sorted by date of receipt and priority. Buyers have the ability to select which requisition they will work on based on this prioritized list. When a buyer selects a requisition for processing, it

is appended to a "Buyer Worksheet". Worksheets separate line items on a requisition and collect data for each one, while maintaining the requisition as a whole. An initial buyer review determines if additional item information is required. Buyers access the various description files to append any additional item information. Next, the Buyer begins gathering appropriate sources of supply. If a source has been suggested on the input document, the Buyer may, or may not, accept this source. If the Buyer elects not to use the suggested source, the information is deleted from the worksheet. Additional sources are accumulated from the various support files, such as the Price History file, the Manufacturer file and the Source file. The Buyer accesses any or all of these files, as necessary, according to local policy. Once sufficient sources have been collected, the worksheet contains the information necessary for the buyer to contact all sources. Information specific to each line item is solicited from each source and input by the buyer. The Buyer also inputs; whether the source contacted was responsive, i.e., whether they were available or interested in the contract; whether they had the required item in stock and in the quantities required; the quoted price for each item; and the promised delivery date should the contract be awarded.

Once all item data has been accumulated, the buyer reviews the data and selects a final source based on local

activity criteria, such as lowest price, quickest delivery, a combination of both, or any other criteria mandated by management. An "award" flag is attached to the selected source data and all reviewed source information is stored in the Contract file. Buyers then append mandatory clauses to the Contract, if any apply.

At any time during the Buyer Worksheet process, a buyer may cancel or refer a requisition. Cancellation or referral codes initiate processing for: Cancellation, including generating a cancellation memorandum, if it is to be canceled; or suspends the requisition, as described above, if it is to be referred. The referral code also generates a referral memorandum. This completes the Buyer Worksheet process.

4. Output Process

Awarded requisitions are kept in the Contract file awaiting the supervisor's final review. The supervisor reviews all requisitions in the contract file and processes them for individual or batch printing. Once the supervisor has approved the requisition, a Contract number, a Date of contract and a Delivery Date are added to the requisition and the requisition becomes a formal contract. Contracts queue to the printer for final processing.

Completed contracts are printed in the form of a Navy Standard Order for Supplies and Services (DD Form 1155), and includes all appropriate data for that document,

including approval signature blocks. In addition to the first document (DD Form 1155), an abstract of the final contract is produced for in-house reporting purposes. This document contains statistical data on the specific transaction.

Cancellation and referral notices are also printed. These notices are in a standard format, with prescribed information included. The information on each notice is keyed to the specific cancellation or referral code and fully describes the reasons for the action. Notices include all appropriate requisition data, including, but not limited to, activity name and address, requisition number, date of requisition, and the line items involved. (Cancellations do not require the line item information.)

C. ADDITIONAL REQUIREMENTS

1. Access Security. The mini-APADE prototype should include measures to protect the integrity and security of all data. Security measures include the use of a hierarchical password system which allows only authorized users access to separate levels of the application. The prototype contains, at a minimum, three levels of record security. These levels include:

- Supervisory - The highest level of access. The Supervisor has access to all files and data in the system as well as access granting authority.
- Buyer - Buyers are assigned buyer codes and individual passwords for access to those purchase requests assigned to their code only.

* Maintenance - Maintenance personnel have access to the system for maintenance only. A password and code authorizes maintenance personnel access to perform structural corrections and system backups.

Each of the functional requirements described above are now tailored to a prototype of the mini-APADE ER diagram. Details of the prototype design are described in the following chapter.

VI. MICRO-APADE PROTOTYPE DESIGN

The functional requirements above, combined with the ERD in Appendix I, provide the basis for a prototype design. Our task is to use the high level ER diagram, modified from the mainframe system and the data flow diagrams designed in Chapter V and generate specific design requirements for the prototype. The design of the prototype must incorporate all required small purchase functions in a relational database.

A. PROTOTYPE DESIGN SPECIFICATIONS

1. Prototype Data Design

The Entity Relationship Diagram in Appendix I can now be forward engineered [Ref 6, pp. 329-330] into a relational map. The resultant relations represent the file and data structure for the prototype. The only addition made to the relational map is a further separation of the Source relation into both Source and BPA relations. The attributes, Agree_ind and Status, are removed from the Source relation and used, along with the key, Name_add, to create the BPA relation. This is to accommodate the procedural differences between the two files. Commercial sources, price history documents and manufacturer information documents are appended singly. Blanket Purchase Agreement records are appended in groups of three to comply with current policies. Commercial and Blanket Purchase

sources are selected on an automatically rotating basis to allow a fair review of all available sources. This precludes favoritism and streamlines the buyer search process. The BPA relation has a one to one relationship to the current Source relation. A relational map of the Micro-APADE system is shown in Appendix K. The Buyer Worksheet mentioned below is a system generated skeleton used for relational operations required to complete the processing.

2. Prototype Process Design

The APADE prototype is composed of four distinct process areas (as shown in Appendix J); Input, Assignment, Pre-Award/Award, and Output. Processes are accessed via a menu hierarchy. This hierarchy closely follows the system diagram in Appendix J, although menus do not appear at the primitive levels. A menu hierarchy, showing access paths to each process is included as Appendix L. The following paragraphs detail the specifications for each process and sub-process by the process numbers given in Appendix J.

a. Input Process (1)

For each instance of a requisition, the input process:

* Accepts manual entry of a requisition. Input is made to a generic screen which includes all possible fields from any recognized requisition document.
* Query the user for the number of line items on the requisition.
* Accept multiple line item requisitions as a single transaction.

* Include a suggested source of supply as an optional field. If a source of supply is submitted, the vendor information is stored in a temporary file which is accessed during the Buyer Worksheet process, when sources are being gathered for review.

b. Edit sub-process (1.1)

Once requisitions are entered into the system, they are subjected to a field edit to determine validity. This process is further decomposed into five sub-processes.

c. Validate Fields (1.1.1)

For each instance of a requisition, and for each line item on the requisition, the system:

* Ensures that the Priority, Nomenclature, Unit of Issue and Quantity fields are not blank.
* Ensures that the Unit of Issue field for each line item is a valid unit of issue.
* Ensures that the FSC field is valid.
* If any of the fields above are in error, invokes a screen prompt to inform the user of the type and location of the error.

d. Correct Requisition (1.1.2)

On receipt of an error message, the user attempts to manually enter corrections to the requisition fields. Corrected, validated requisitions are sent to the Validate Requisition Number sub-process (1.1.4). Uncorrectable requisitions are forwarded to the Refer Requisition sub-process (1.1.3).

e. Refer Requisition (1.1.3)

If the user is unable to make corrections on the requisition, the user:

* Flags the requisition number as a referred requisition. (Flagged requisitions are in a state of "suspense" and are not processed further until corrective action has been taken.)

The system:

* Appends the appropriate referral code from the Referral file to the requisition.

* Generates an Invalid Requisition Report notification for the output process.

f. Validate Requisition Number (1.1.4)

The system:

* Scans the requisition number for completeness.

* Searches the Requisition File for matching requisition numbers.

* Notifies the user with a screen prompt that the current requisition number has been previously entered.

* If a matching requisition number is already in the Requisition File, a Cancellation Report notification is generated for the output process.

* Valid requisitions are forwarded to the Assign Process.

g. Assign Process (2)

For each valid requisition entered into the

Assign Process, The Small Purchase supervisor:

* Assigns a Purchase Request Number. The supervisor obtains the next available Purchase Request number (Pr_no) from an external Purchase number file. These numbers are system generated.

* Assigns requisitions to Buyers by Buyer code and based on current buyer workload.

h. Assign Purchase Number (2.1)

The system:

* Obtains the next Purchase Request number from the System Supervisor and appends it to the requisition.

i. Assign Buyer (2.2)

The system:

* Appends the Buyer Code to the requisition.

Fully assigned requisitions are forwarded to the Pre-Award/Award process.

j. Pre-Award/Award Process (3)

Once requisitions are validated and assigned, they enter the Pre-Award/Award process. During this process, buyers collect additional item information and specifications, accounting data for the contract, and any additional address information required for the receiving activity (Ship to file) or the requesting activity. Next, the buyer gathers source information, solicits bids from potential sources and awards the contract to the best source. Once a source has been selected, the requisition becomes a Contract and is sent to the Contract file to await output processing. The information collected on unsuccessful sources is also sent to the Contract File for historical and audit purposes.

k. Update Requisition (3.1)

The user:

* Gathers any missing or additional item information from the description support files.
* Verifies or obtains shipping addresses and information through the Shipto file.

l. Gather Sources (3.2)

For each updated requisition, a maximum of 99 potential sources may be appended to the Buyer Worksheet. The user does any, or all of the following:

* Screens and selects historical data for review from the Price History File.

* Screens and selects commercial source data for potential vendors. For fairness in solicitation, vendors in this file are reviewed sequentially. No vendor may be selected for review a second time until all vendors have been made available at least once.

* Screens the BPA File for potential sources. Due to Acquisition regulations, users must select a minimum of three sources from this file.

The system:

* Notifies the user if no sources are available which meet the criteria.

If no sources are available, the user then:

* Flags the requisition as unprocessible.

* Obtains the appropriate Referral code from the Referral File.

* Generates a No Sources Available Report Notification for the Output process.

m. Solicitation Process (3.3)

Once all desired sources are collected in a requisition source list, the user begins soliciting price and availability information. During this process, the user verifies the responsiveness of each source. If the source is not available for solicitation, or if the item requested is not carried, an appropriate entry is appended to the worksheet. During this process, the user also verifies

source data. Updated source data is stored temporarily on the worksheet and reviewed by the supervisor at a later date.

n. Verify Vendor Info (3.3.1)

During the off line solicitation, all vendor information is verified. For each source appended to the worksheet, the user:

* Verifies vendor information.
* Enters modifications to the vendor information on the worksheet.

o. Solicit Vendor Quotes (3.3.2)

For each instance of requisition, for each line item on the requisition and for each potential source on the worksheet, the user:

* Determines source responsiveness.
* Flags the source with the appropriate responsiveness code.
* Appends comments to the worksheet concerning source responsiveness.
* Obtains unit price, unit availability and delivery information from the source.
* Appends source bid information to the worksheet.

The system:

* Accepts source prices, determines extended prices based on the desired quantity.
* Generates a cumulative bid total based on the extended prices for each line item requested.

The user:

* Notes if the cumulative total exceeds the Estimated Price on the requisition.

If the cumulative total exceeds the Estimated price, the user:

* Flags the source as Non-Responsive.
* Appends the appropriate referral code from the Referral File.
* Generates an Ineligible Price Report Notification for the Output process.

p. Award Contract (3.3.3)

After all potential sources have been contacted, the user:

* Reviews quotes received.
* Selects a source for the contract which best meets locally determined criteria.
* Flags the selected source with an appropriate award indicator.

The system:

* Accepts the award indicator and appends the requisition with the selected source to the Contract file.
* Generates a Contract Number and appends it to the awarded requisition. (Contract numbers are system generated, pre-formatted, numbers based on local activity requirements. The last generated number is stored in memory and updated when required for a new contract.)
* Appends the remaining unselected sources and solicitation information to the Contract file.

q. Output Process (4)

Once a source has been selected for a requisition and the requisition has been stored in the Contract file, the requisition becomes a potential contract. The last process is product output. During the output process, three separate functions occur: The contracts and

reports are reviewed and approved by the supervisor; Contracts and contract file abstracts are printed, all report notifications are printed.

r. Supervisor Review (4.1)

Upon receipt of valid contracts and error, or referral, reports, the supervisor:

* Ensures all contract data is accurate.
* Approves contracts for final printing.
* Ensures all required data is entered on the reports.
* Ensures data on reports is accurate.
* Elects to print contracts individually or in contract batches.
* Elects to print reports individually or in report batches.

s. Output Contract (4.2)

For each contract approved by the supervisor, the system:

* Produces a signable contract document in standard DD Form 1155 format.
* Produces a file copy of the contract.

t. Output Reports (4.3)

For each error or referral report approved by the supervisor, the system:

* For a cancellation, obtains the appropriate cancellation text from the cancellation file, based on the cancellation code.
* For a referrral, obtains the appropriate referral text from the referral file, based on the referral code. (Invalid requisition reports and no source available reports are included in the referral file.)

* Produces a signable memorandum for the appropriate activity.

VII. CONCLUSIONS

A. REVERSE ENGINEERING

The use of reverse engineering theory on existing flat file systems can effectively produce useable relational database designs. The steps outlined in phase one of our two phase algorithm analyze attributes of each file, synthesizing them based on functional dependency into a set of relations. The second phase of our algorithm provides a set of rules for creating entities and relationships from this relational schema. These entities and relationships are then used to construct a high level Entity Relationship Diagram. This ER diagram is then "forward engineered" to produce a viable database design.

When attempting to reverse engineer a system manually, care must be taken to ensure that all data is included in the review. Reviewing all data structures in a major system is a long and tedious process and the probability of overlooking some data is high. Functional manuals for the system are invaluable aids in this process. We were able to obtain copies of the functional descriptions of the APADE system from the Naval Supply Center, Oakland, California. The manuals provided not only the file and data structures, but also the higher level links between the separate processes and the files used in them. A firm understanding

of the overall process and the organizational goals streamlines the search for data in the system. This information was obtained both through the functional manuals and by frequent visits to the Supply Center. Interviews with managers and users alike gave us invaluable insight into the system structure and functionality. Without such aids, the research would have been extremely difficult, if not impossible in the time frame for completing the thesis.

Automated tools are capable of generating system structure information based on manual file input or system code analysis. The use of such tools would greatly enhance this research process, resulting in more sophisticated products.

B. PROTOTYPE

The prototype was written using the Developer's edition of DBASE IV. Again, visits to the Supply Center in Oakland proved invaluable. Our goal for this stage of the research was to gather as much user input as possible, match this information to the small purchase structure of APADE, and incorporate all this information into our design. As a result of these visits, the input structure of our prototype is more effective and efficient than that of the mainframe. It is almost a paperless system. Where APADE requires the user to manually prepare data for technical descriptions and the buyer worksheet, the prototype contains all files necessary to accomplish these processes automatically.

The prototype runs on a standard PC with 80286 or 80386 processors. The prototype, written within the DBASE IV database management system, requires a minimum of 1 Megabyte of memory. Even in this uncompiled version, the prototype operates with no discernable delays.

The prototype was demonstrated to our thesis sponsor, the APADE Functional Management Division of NAVSUP, in September and again in November, 1990. Four cosmetic and one operational change were requested. The cosmetic changes involved screen displays. The operational change involved re-indexing the source databases to provide data retrieval sorted by several different fields. These changes were incorporated into the prototype and the system has been accepted by the sponsor without further reservation. The prototype code has been delivered to NAVSUP and the production model of our prototype is to be introduced to the Navy's small purchase shops on a trial basis within the next three months. A fully matured version of the prototype is expected within the next six months.

During the demonstration, a discussion of efficiency ensued, resulting in contact with a software developer who promised, and delivered an updated, DBASE IV compatible, compiling program. It is estimated that the efficiency of the prototype will increase by thirty percent when compiled. This should allow a production version of the prototype to operate on virtually any PC machine.

DBASE IV has limitations when applied to a truly relational schema. For instance, the number of files available for update cannot exceed three during any join operation. Using the QBE capability of the program enables a programmer to join up to eight files at a time, however. This is a read only structure and, as such, is good only for data retrieval. The prototype required more flexibility. Extensive use of private memory variables enabled us to overcome this obstacle without seriously impairing the relational structure. We were able to join files, store pertinent data to memory, manipulate the memory variables, and then replace the data in each appropriate file. This is not the operational format of a relational system. For the prototype, DBASE IV was the medium of choice since it is a widely used software in the military and, as such, presents a familiar format for untrained users. A compiled version of the prototype will present a familiar DBASE IV format.

The final version of the prototype included a BPA system which actually improved upon the APADE system. Naval Contracting methodology requires at least three BPA vendors to be screened for each purchase. Also, all vendors for a particular commodity must receive an equal share of the workload, or, at least an opportunity for an equal share. The prototype routines for collecting BPA sources fully support the regulations for ensuring fairness in solicitations. The prototype sequentially selects three BPA

sources and appends them to each worksheet. The user cannot decrease this number, but can add additional BPA sources, in groups of three, if desired. The mainframe system leaves these requirements to the users' discretion.

C. FOLLOW-ON WORK

Three follow-on issues will be discussed here;

(1) Small purchase areas not addressed in the prototype;

(2) Additional areas for potential follow-on work, and

(3) System maintenance.

1. Small Purchase Areas not addressed in the Prototype

As mentioned in Chapter V, all areas of the APADE system which pertain to the large purchase operation were intentionally eliminated from the prototype. Some areas within the small purchase purview were also not addressed. The small purchase areas not addressed include: security, IDTC/Autobuy processing, FSS-Schedule and PID files, cancellation processing, and price history processing.

a. Security

Security is an important aspect in any system. The basic security requirements of the mini-APADE prototype were detailed in Chapter V. The prototype includes a primitive password system which grants access based on a buyer code and password. This was included for demonstration purposes only. A production version of this prototype must have all three access levels included in its security system.

b. IDTC/Autobuy Processing

The IDTC files and the Autobuy file were intentionally left out of the prototype. The process was not considered a routine purchase transaction. Our intent was to develop a program which addressed a purchase requiring a full range of source screening and selection. The IDTC/Autobuy process automatically generates a Delivery Order with no such screening. In order to produce a working model of an APADE-like small purchase system, we kept our focus on a small purchase process which encompassed the entire range of transactions. The IDTC/Autobuy process does not include all the routine transactions. In fact, it would be a distinct process of a system. Future remodelling and refinement of the prototype could include the IDTC/Autobuy process as a separate module.

c. FSS-Schedule and PID Files

The FSS_Sked file was not included in the prototype. Use of the FSS_Sked file would enlarge the available source files for the buyer worksheet. However, the Federal Supply Schedule (FSS) is quite large and even a representative sample may have impaired the supposed effectiveness of the prototype. This file would not constitute a new process module for the prototype, rather, it would only enlarge the Gather Sources function. This file would better be added during the compilation of the

production model. The PID file was not included for the same reasons as given for the FSS_Sked file.

d. Cancellation Processing

The Cancellation file includes various types of cancellations with explanations. For our prototype, we included only one variation for canceling a requisition, a duplicate requisition number. This was included in the code and not adapted as a file with one record.

e. Price History Processing

The Price History file in the prototype is a read only file. It is imperative to incorporate a design to allow for the appending of information to this file. This will provide routine updating of the historical database, keeping contract numbers and items prices current.

2. Additional Areas for Potential Follow-on Work

a. Database Backup System

An effective method to provide systematic back up of program files has not been incorporated in the prototype. Off line backup is the only method at the present time. Procedures should be developed to routinely, and automatically, back up system files on a fixed schedule.

b. Management Report Development

A management report system, tailored to small procurement activities, should be incorporated into the

prototype. A wide variety of management reports are needed for management review and control. Necessary data is available in the files to provide the reports needed.

c. File Maintenance

A process needs to be developed to purge file data as necessary in conjunction with file backup maintenance. This process is required to control growth in file records. Controlled file record growth will maintain optimum efficiency of the system.

d. Network Capability

Presently, the prototype is a stand alone system, capable of individual user use. Networking the program would allow an expanded capability for the activity and centralization of archive and source files. The functional requirements require supervisory input at various places in the process. They also require a security system which allows only assigned buyers access to assigned requisitions. These functions would be better served on a network. A PC-LAN network would be extremely economical and ideal for the intended purpose of this system; i.e., a five to eight person operation. The average installation cost of such a network is approximately $450.00 per station, not including hardware. Compared with the lease costs of the APADE mainframe system, this would be extremely affordable.

e. Modem Capability

The use of modems, particularly with software which would auto-dial sources could increase the efficiency of this system. Also, the Supply Center in Jacksonville, Florida has experimented with software which placed un-awarded requisitions on a computer billboard. The experiments have met with a moderate success and indicates an additional use for automated computer purchasing.

3. System Maintenance

The Micro-APADE prototype requires constant maintenance to remain current. This task is to be absorbed by the APADE functional office at NAVSUP whose staff programmers are fine tuning the system during the test phase and will centrally maintain it after implementation.

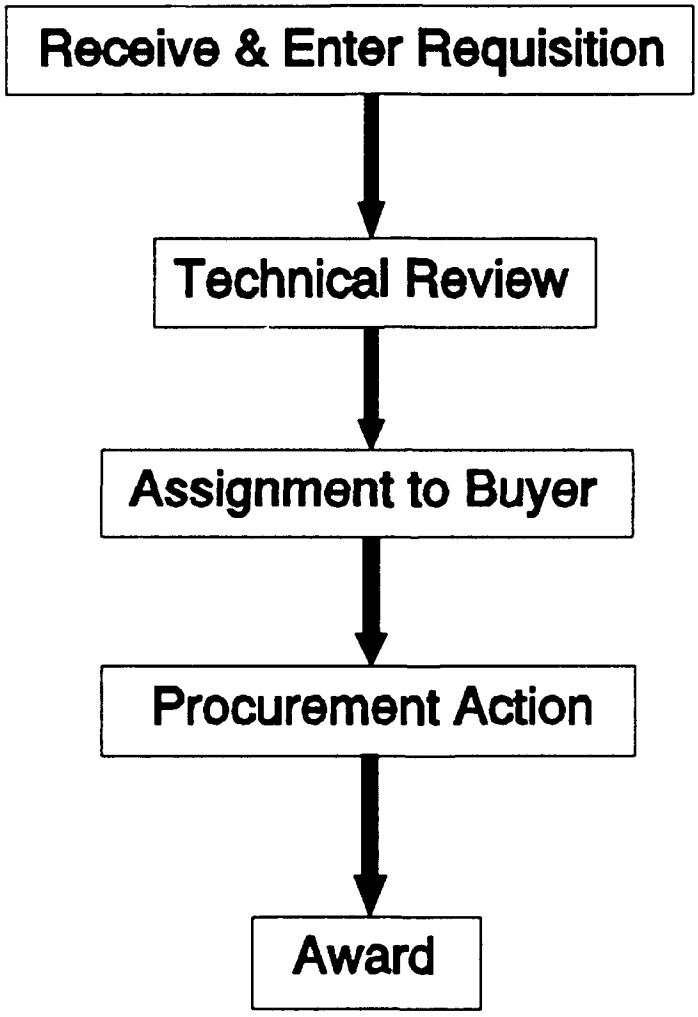
D. SUMMARY

This thesis uses a reverse engineering theory to effectively produce a high level, conceptual view of a DoD mainframe system. The conceptual view provides the basis for the functional and design requirements of a relational database. A micro computer prototype has been developed from this design which retains all the functionality of the small procurement modules of the original system. Areas of potential follow on work and areas not covered by the prototype have been identified and discussed.

The prototype has been accepted for testing by NAVSUP. Current plans are for possible implementation in July, 1991.

APPENDIX A

REQUISITION FLOW CHART



APPENDIX B

APADE FILES

File Name	Element Name	Data Structure	Description	Mandatory /Optional
RECORD				
	Doc_Req_No	15 Char.	Document/Requisition Number	M
	ACRN	2 Char.	Allowance Cross Ref Nr	O
	APL_No	10 Char.	Allowance Parts List Number	O
	Appropriat	7 Char.	Appropriation	O
	Auth_Acct	6 Char.	Authorization Accounting Activity	O
	Bur_Cont_No	5 Char.	Bureau Control Number	O
	COG	2 Char.	Cognizance Symbol	O
	Color	5 Char.	Item color	O
	Cost_Code	12 Char.	Activity Cost Code	O
	Descr	500 Char.	Item description	O
	Dist	2 Char.	Distribution code	O
	DMS	4 Char.	DMS rating	O
	EIA	20 Char.	End item application	O
	Est_price	8 Num.	Estimated Price	O
	FSC	4 Char.	Federal Stock Category	O
	FSCM	5 Char.	Manufacturer Federal source code	O
	Fund_code	2 Char.	Activity Fund code	M
	Fund_expdt	6 Num.	Funds expenditure date	O
	Line_Item_no	3 Num.	Line item number	M
	Mf_cat	20 Char.	Mfr catalog number	M
	Mod	14 Char.	Model number	M
	NSN_LSN	13 Char.	National or Local stock number	O
	Obj_class	3 Char.	Object class	O
	Pr_no	6 Char.	Purchase Request number	M
	Priority	2 Num.	Priority	M
	Proj_code	3 Char.	Project code	O
	Prop_acct	6 Char.	Property accounting activity	O
	Pt_no	17 Char.	Mfr part number	O
	Qty	8 Num.	Quantity	M

File Name	Element Name	Data Structure	Description	Mandatory /Optional
RECORD (cont'd)				
	RDD	3 Num.	Required delivery date	O
	Serial	12 Char.	Item serial number	O
	Series	12 Char.	Item series	O
	Sig_Code	1 Char.	Signal code	O
	Size	15 Char.	Item size	O
	SOS	20 Char.	Source of Supply	O
	Sub_allot	1 Char.	Sub allotment code	O
	Subhead	4 Char.	Appropriation subhead	O
	Supp_add	6 Char.	Supplementary address	O
	Tech_man	20 Char.	Technical manual number	O
	Tech_ord	15 Char.	Technical order number	O
	Transactio	2 Char.	Transaction type	O
	UIC	6 Num.	Unit Identification code	M
	Unit_iss	2 Char.	Unit of issue	M
CLAUSE				
	Clause_no	20 Char.	Clause identification number	M
	Title_date	100 Char.	Title and date	O
	Text	Var. Memo	Clause text	M
SHIPTO				
	Customer	40 Char.	Customer name	M
	Ship_add	60 Char.	Shipping Address	O
	Ship_code	2 Num.	Shipping information code	O
	Ship_info	16 Char.	Shipping information code definition	O
	UIC	6 Num.	Unit identification code	M
PRICE_HI				
	Contr_no	17 Char.	Contract number	M
	Cust_uic	6 Num.	Customer UIC	O
	Date_of_kt	4 Num.	Date of contract	M
	K_size	1 Num.	Contract size	M
	Nomenclatu	50 Char.	Nomenclature	M
	NSN_LSN	13 Char.	National or Local stock number	O
	Prod_leadt	3 Num.	Production lead time	O
	Quantity	8 Num.	Quantity	O

File Name	Element Name	Data Structure	Description	Mandatory /Optional
PRICE_HI (cont'd)				
	Ref_no	32 Char.	Reference number	O
	Unit_iss	2 Char.	Unit of issue	M
	Unit_pr	8 Num.	Unit price	M
	Vendor	5 Num.	FSCM of the vendor	M
	Vendor_siz	1 Char.	Vendor size	M
IDTC				
	Admin_name_ add	100 Char.	Administering activity name and address	O
	Clin_subclin	6 Char.	Contract line item or sub line item number	M
	Commod_ind	6 Char.	Commodity indicator	M
	Contr_name_ add	100 Char.	Contractor name and address	M
	Contr_no	17 Char.	Contract number	M
	Delivery	3 Char.	Delivery code	O
	Disc_term	1 Char.	Discount terms	O
	Eff_ord_ dte_stp	6 Num.	Stop effective order date	M
	Eff_ord_ dte_stt	6 Num.	Start effective order date	M
	FOB	5 Char.	F.O.B. point	O
	FSCM	5 Char.	Manufacturer federal source code	O
	Item_descr	100 Char.	Item description	O
	Last_de_or	4 Num.	Last delivery order number	M
	Max_ord_q	4 Num.	Maximum order quantity	O
	Min_ord_q	2 Num.	Minimum order quantity	O
	PAA_name_ add	100 Char.	Paying activity name and address	O
	Provisions	2000 Char.	Special provisions	O
	Unit_iss	2 Char.	Unit of issue	M
*	Unit_pr	20 Num.	Unit price	M

* - Unit_pr element is repeated ten times in this file for additional pricing.

SOURCE				
	Agree_ind	5 Char.	Existing agreement indicator	O
	Bus_cat	2 Char.	Business category	M
	Commod_ind	6 Char.	Commodity indicator	M

File Name	Element Name	Data Structure	Description	Mandatory /Optional
SOURCE (cont'd)				
	Deb_in_sus	7 Char.	Debarred, ineligible, suspended code	O
	Explanatio	200 Char.	Explanation	O
	FSCM	5 Char.	Manufacturer federal source code	M
	Name_add	100 Char.	Name and address	O
	Status	1 Char.	Status indicator	M
	Use_indica	1 Char.	use indicator	O
ACCTFILE				
	Appropriat	7 Char.	Appropriation	M
	Auth_acct	6 Char.	Authorization accounting activity	M
	Bur_ctr_no	5 Char.	Bureau Control number	M
	Cost_code	12 Char.	Cost Code	M
	Fund_code	2 Char.	Fund code	M
	Obj_class	3 Char.	Object class	M
	Prop_acct	6 Char.	Property accounting activity	M
	Sub_allot	1 Char.	Suballotment number	M
	Subhead	4 Char.	Appropriation subhead	M
	Transactio	2 Char.	Transaction Type	M
	UIC	6 Num.	Unit identification code	M
BUYER				
	Buyer_code	3 Char.	Buyer code	M
	Buyer_grad	2 Char.	Buyer pay grade	O
	Buyers_nam	25 Char.	Buyer's name	O
	Corresp_co	7 Char.	Buyer's correspondence code	O
	Occupat_se	5 Char.	Occupational series	O
	Phone_ext	4 Num.	Buyer's telephone extension	O
	Phone_nr1	10 Num.	Buyer's telephone number	M
	Phone_nr2	10 Num.	Buyer's alternate telephone number	O
AUTOBUY				
	Admin_name_ add	100 Char.	Administering activity name and address	O
	Clin_subclin	6 Char.	Line item or sub-line item number	M

File Name	Element Name	Data Structure	Description	Mandatory /Optional
AUTOBUY (cont'd)				
	Contr_name_ add	100 Char.	Contractor name and address	M
	Contr_no	17 Char.	Contract number	M
	Delivery	3 Char.	Delivery	O
	Disc_term	1 Char.	Discount terms	O
	Dte_lst_use	6 Num.	Date of last use	M
	Eff_ord_ dte_stp	6 Num.	Stop effective order date	M
	Eff_ord_ dte_stt	6 Num.	Start effective order date	M
	FOB	4 Char.	F.O.B. point	O
	Item_descr	100 Char.	Item description	M
	Last_de_or	4 Num.	Last delivery order number	M
	Max_ord_q	4 Num.	Maximum order quantity	O
	Min_ord_q	2 Num.	Minimum order quantity	O
	NSN_LSN	13 Char.	National or local stock number	M
	PAA_name_ add	100 Char.	Paying activity name and address	O
	Ref_no	37 Char.	Reference number	O
	Unit_iss	2 Char.	Unit of issue	M
*	Unit_pr	20 Num.	Unit price	M

* - The unit price element is repeated ten times in this file for additional pricing.

PIDFILE				
	Descr_data	500 Char.	Descriptive data	M
	Dte_lst_use	4 Num.	Date of last use	O
	Nomenclatu	50 Char.	Nomenclature	M
	NSN_LSN	13 Char.	National or Local stock number	M
	Ref_no	32 Char.	Reference number	O

APPENDIX C

APADE FILE DEPENDENCY ANALYSIS

The following analysis begins with the APADE data structure by file. The preliminary synthesis represents the initial division of each file into a more functionally dependent schema. The final breakdown shows the ultimate relational model for each file considering entity integrity. Data attributes which are noted as foreign keys for other entities are listed in brackets, [], while primary keys for each schema are underlined. The functional dependencies for each schema are listed below each initial synthesis but not in the final model.

File	Elements
1. RECORD(	Doc_Req_No, ACRN, APL_No, Appropriat, Auth_Acct, Bur_Cont_No, COG, Color, Cost_Code, Descr, Dist, DMS, EIA, Est_price, FSC, FSCM, Fund_code, Fund_expdt, Line_Item_nr, Mf_cat, Mod, NSN_LSN, Obj_class, Pr_no,Priority, Proj_code, Prop_acct, Pt_no, Qty, RDD,Serial, Series, Sig_Code, Size, SOS, Sub_allot,Subhead, Supp_add, Tech_man, Tech_ord, Transactio, UIC, Unit_iss)

PRELIMINARY SYNTHESIS

a. Describes the Requisition

RECORD(Doc_Req_No, Dist, Fund_code, Fund_expdt, Priority, Pr_no, Supp_add, Proj_code, RDD, Sig_code, SOS, UIC)

```
Doc_Req_No -> Dist
           -> Fund_code
           -> Fund_expdt
           -> Pr_no
           -> Priority
           -> Supp_add
           -> Proj_code
           -> RDD
           -> Sig_code
           -> SOS
           -> UIC
```

b. Describes the Item

```
RECORD/item(APL_No, COG, Color, Descr, ACRN,
            Est_price, FSC, FSCM, Line_item_nr,
            Mf_cat, Mod, NSN_LSN, Pt_no, Qty,
            Serial, Series, Size, Tech_man,
            Tech_ord, Unit_iss, DMS, EIA,
            Doc_Req_no)

  Doc_Req_no, Line_item_nr, -> Descr
                            -> Qty
                            -> Unit_iss
                            -> Est_price
                            -> Color
                            -> Size

                       Descr -> FSC
                             -> APL_No
                             -> COG
                             -> NSN_LSN
                             -> ACRN
                             -> DMS
                             -> EIA

                        FSCM -> Mf_cat

                 Descr, FSCM -> Mod
                             -> Pt_no
                             -> Serial
                             -> Series
                             -> Tech_man
                             -> Tech_ord
```

c. Describes the Requisition
   Accounting Data

```
RECORD/acct(Appropriat, Auth_acct, Bur_Contr_no,
            Cost_code, Fund_code, Obj_class,
            Prop_acct, Sub_allot, Subhead,
            Transactio, Doc_Req_no)

                   Fund_code -> Appropriat
                             -> Auth_acct
                             -> Bur_Contr_no
                             -> Obj_class
                             -> Prop_acct
                             -> Sub_allot
                             -> Subhead
                             -> Transactio

                  Doc_Req_No -> Cost_code
```

FINAL BREAKDOWN

a. Describes the Requisition

RECORD(Doc_Req_No, Dist, [Fund_code],
Fund_expdt, Pr_no, Priority,
Supp_add, Proj_code, RDD, Sig_code, SOS,
UIC)

b. Describes the Accounting Data

RECORD/acct([Doc_Req_no], Appropriat, Auth_acct,
Bur_Contr_no, Cost_code, Fund_code,
Obj_class, Prop_acct, Sub_allot,
Subhead, Transactio)

c. Describes the Requisition Item

RECORD/item(Doc_req_no, Line_item_nr, [Descr],
Qty, Unit_iss, Est_price)

d. Describes a Generic Item

RECORD/item_descr(Descr, FSC, APL_No, COG, ACRN,
DMS, EIA, [FSCM], NSN_LSN,
Color, Size)

e. Describes the Items Supplied By a Manufacturer

RECORD/descr_mfr([Descr], [FSCM], Mod,
Pt_no, Serial, Series,
Tech_man, Tech_ord)

f. Describes the Manufacturer

RECORD/mfr_info(FSCM, Mf_cat)

File	Elements
2. CLAUSE(Clause_nr, Title_Date, Text)	

PRELIMINARY SYNTHESIS

a. Describes the Clause

CLAUSE(Clause_nr, Title_Date, Text)

Clause_nr -> Title_Date
-> Text

FINAL BREAKDOWN

a. Describes the Clause

CLAUSE(Clause_nr, Title_Date, Text)

File	Elements
3. SHIPTO(UIC,	Ship_cod, Ship_info, Customer, Ship_add)

PRELIMINARY SYNTHESIS

a. Describes the Receiving Activity.

SHIPTO(UIC, Customer, Ship_add,
Ship_code, Ship_info)

UIC -> Ship_code
-> Customer
-> Ship_add
-> Ship_info

FINAL BREAKDOWN

a. Describes the Receiving Activity.

SHIPTO(UIC, Ship_code, Customer, Ship_add,
Ship_info)

File	Elements
4. PRICE_HI(NSN_LSN,	Ref_nr, Contra_nr, Date_of_kt, Vendor, Vendor_siz, Quantity, Unit_issue, Cust_uic, Nomenclatu, Unit_price, Prod_leadt, K_size)

PRELIMINARY SYNTHESIS

a. Describes the Items in the Historical Contract.

PRICE_HI(NSN_LSN, [Contra_nr], [Nomenclatu],
Quantity, Unit_issue, Unit_pr,
Prod_leadt)

Contra_nr, Nomenclatu -> NSN_LSN
-> Quantity
-> Unit_issue
-> Unit_pr
-> Prod_leadt

b. Describes the Historical Contract.

PRICE_HI_CONTR(Contra_nr, Date_of_kt,
Vendor, Vendor_siz, K_size,
Cust_uic, Ref_nr)

Contra_nr -> Date_of_kt
-> Vendor
-> Vendor_siz
-> K_size
-> Cust_uic
-> Ref_nr

FINAL BREAKDOWN

a. Describes the Items in the Historical Contract.

PRICE_HI(NSN_LSN, [Contra_nr], [Nomenclatu],
Quantity, Unit_issue, Unit_pr,
Prod_leadt)

b. Describes the Historical Contract.

PRICE_CONTR(Contra_nr, Date_of_kt, Vendor,
Vendor_siz, K_size, Cust_uic,
Ref_nr)

File	Elements
5. IDTCFILE	(Commod_ind, Nomenclatu, Contr_nr, FSCM, Last_de_or, Eff_ord_dte_stt, Eff_ord_dte_stp, Cont_name/add, Admin_name/add, Disc_term, Clin_subcl, Item_descr, Unit_issue, Unit_pr, Min_ord_q, Max_ord_q, Delivery, FOB, Provisions)

PRELIMINARY SYNTHESIS

a. Describes the IDTC Contract.

IDTC(Contr_nr, FSCM, Commod_ind, Last_de_or,
Eff_ord_dte_stt, Eff_ord_dte_stp,
Contr_name/add, Admin_name/add,
Disc_term, Delivery, FOB, Provisions)

Contr_nr -> Commod_ind
-> Last_de_or
-> Eff_ord_dte_stt
-> Eff_ord_dte_stp
-> Admin_name/add
-> Disc_term

-> Delivery
-> FOB
-> Nomenclatu
-> Provisions

FSCM -> Contr_name/add

b. Describes the IDTC Contract Items.

IDTC_ITEM([Contr_nr], [Nomenclatu],
Clin_subcl, Item_descr,
Unit_issue, Unit_pr,
Min_ord_q, Max_ord_q)

Contr_nr, Nomenclatu -> Clin_subcl
-> Item_descr
-> Unit_issue
-> Unit_pr
-> Min_ord_q
-> Max_ord_q

FINAL BREAKDOWN

a. Describes the IDTC Contract.

IDTC(Contr_nr, FSCM, Commod_ind, Last_de_or,
Eff_ord_dte_stt, Eff_ord_dte_stp,
Contr_name/add, Admin_name/add,
Disc_term, Delivery, FOB, Nomenclatu,
Provisions)

b. Describes the IDTC Contract Items.

IDTC_ITEM([Contr_nr], [Nomenclatu], Clin_subcl,
Item_descr, Unit_issue, Unit_pr,
Min_ord_q, Max_ord_q)

File	Element
6. SOURCE	(Commod_ind, FSCM, Name_add, Deb_in_sus, Explanatio, Bus_cat, Status, Agree_ind, Use_indica)

PRELIMINARY SYNTHESIS

a. Describes the Source.

SOURCE(FSCM, Commod_ind, Name_add,
Deb_in_sus, Explanatio, Bus_cat,
Status, Agree_ind, Use_indica)

Name_add -> Bus_cat
-> Agree_ind
-> Use_indica
-> Status
-> FSCM

Name_add ->> Commod_ind

Status -> Deb_in_sus
-> Explanatio

FINAL BREAKDOWN

a. Identifies the Source.

SOURCE(FSCM, Commod_ind, Name_add, Bus_cat, Agree_ind, Use_indica, [Status])

b. Identifies the Commodities Supplied by a Source.

SRCE_COMMOD(Name_add, Commod_ind)

c. Describes the Contractor Status.

SOURCE_STAT(Status, Deb_in_sus, Explanatio)

File	Element

7. ACCTFILE(UIC, Fund_code, Appropriat, Subhead, Bur_ctr_nr, Auth_acct, Obj_class, Sub_allot, Transactio, Prop_acct, Cost_code)

PRELIMINARY SYNTHESIS

a. Describes the Accounting Data.

ACCTFILE(UIC, Fund_code, Appropriat, Subhead, Bur_ctr_nr, Auth_acct, Obj_class, Sub_allot, Transactio, Prop_acct, Cost_code)

UIC, Fund_code -> Appropriat
-> Subhead
-> Bur_ctr_nr
-> Auth_acct
-> Obj_class
-> Sub_allot
-> Transactio

-> Prop_acct
-> Cost_code

FINAL BREAKDOWN

a. Describes the Accounting Data.

ACCTFILE(UIC, Fund_code, Appropriat,
Subhead, Bur_ctr_nr, Auth_acct,
Obj_class, Sub_allot, Transactio,
Prop_acct, Cost_code)

File Element

8. BUYER(Buyer_code, Buyers_nam, Buyer_grad, Occupat_se,
Corresp_co, Phone_nr1, Phone_nr2, Phone_ext)

PRELIMINARY SYNTHESIS

a. Describes the Buyer Information.

BUYERINFO(Buyer_code, Buyers_name,
Corresp_co, Phone_nr1, Phone_nr2,
Phone_ext)

Buyer_code -> Buyers_name
-> Corresp_co
-> Phone_nr1
-> Phone_nr2
-> Phone_ext
-> Buyer_grad
-> Occupat_se

FINAL BREAKDOWN

a. Describes the Buyer Information.

BUYERINFO(Buyer_code, Buyers_name,
Corresp_co, Phone_nr1, Phone_nr2,
Phone_ext, Buyer_grad, Occupat_se)

File Element

9. AUTOBUY(Commod_ind, Nomenclatu, Contr_nr, FSCM,
Last_de_or, Eff_ord_dte_stt, Eff_ord_dte_stp,
Cont_name/add, Admin_name/add, Disc_term,
Clin_subcl, Item_descr, Unit_issue, Unit_pr,
Min_ord_q, Max_ord_q, Delivery, FOB,
Provisions)

PRELIMINARY SYNTHESIS

a. Describes the Autobuy Contract.

```
AUTOBUY(Contr_nr, FSCM, Last_de_or,
        Eff_ord_dte_stt, Eff_ord_dte_stp,
        Contr_name/add, Admin_name/add,
        Disc_term, Delivery, FOB, Provisions)

        Contr_nr -> FSCM
                 -> Last_de_or
                 -> Eff_ord_dte_stt
                 -> Eff_ord_dte_stp
                 -> Contr_name/add
                 -> Admin_name/add
                 -> Disc_term
                 -> Delivery
                 -> FOB
                 -> Provisions
```

b. Describes the Autobuy Contract Items.

```
AUTOBUY_ITEM([Contr_nr], Nomenclatu,
             Clin_subcl, Item_descr,
             Unit_issue, Unit_pr, Commod_ind,
             Min_ord_q, Max_ord_q)

             Contr_nr, Nomenclatu -> Clin_subcl
                                  -> Item_descr
                                  -> Unit_issue
                                  -> Unit_pr
                                  -> Min_ord_q
                                  -> Max_ord_q

                          Contr_nr ->> Nomenclatu
```

FINAL BREAKDOWN

a. Describes the AUTOBUY Contract.

```
AUTOBUY(Contr_nr, FSCM, Last_de_or,
        Eff_ord_dte_stt, Eff_ord_dte_stp,
        Contr_name/add, Admin_name/add,
        Disc_term, Delivery, FOB,
        Provisions)
```

b. Describes the AUTOBUY Contract Items.

```
AUTOBUY_ITEM([Contr-nr], Nomenclatu, Clin_subcl,
             Item_descr, Unit_issue, Unit_pr,
             Min_ord_q, Max_ord_q, Commod_ind)
```

File	Elements
10. PIDFILE	(NSN_LSN, Nomenclatu, Ref_no, Descr_data, Dte_lst_use)

PRELIMINARY SYNTHESIS

a. Describes the Item Descriptions

PID(NSN_LSN, Nomenclatu, Ref_no, Descr_data, Dte_lst_use)

Nomenclatu -> Ref_no
-> Descr_data
-> Dte_lst_use
-> NSN_LSN

FINAL BREAKDOWN

a. Describes the Item Descriptions.

PID(NSN_LSN, Nomenclatu, Ref_no, Descr_data, Dte_lst_use

APPENDIX D

APADE RELATIONAL DATA MODEL

The files listed below are the result of the final data analysis of the mainframe system. The names in parentheses are the names used during the preliminary breakdown. They have been changed to more fully describe the data representations.

1. (RECORD)

REQUISITION(Doc_Req_No, Dist, Fund_expdt,
[Pr_no], Priority, Supp_add,
[Fund_code], Proj_code, RDD,
Sig_code, SOS, [UIC],
[Buyer_code])

2. (ACCTFILE)

ACCTFILE(Appropriat, Fund_code, UIC,
Bur_Contr_no, Cost_code,
Auth_acct, Obj_class, Prop_acct,
Sub_allot, Subhead, Transactio)

3. (RECORD/item)

ITEM([Doc_req_no], Line_item_nr,
[Descr], Qty, Unit_iss, Est_price)

4. (RECORD/item_descr)

DESCRIPTION(Descr, FSC, APL_No, COG, Color,
[FSCM], NSN_LSN, ACRN, DMS, Size,
EIA)

5. (RECORD/descr_mfr)

MFR_SUPPLY([Descr], [FSCM], Mod,
Pt_no, Serial, Series,
Tech_man, Tech_ord)

6. (RECORD/mfr_info)

MFR(FSCM, Mf_cat, Contr_name/add)

7. (CLAUSE)

CLAUSE(Clause_nr, Title_Date, Text)

8. (SHIPTO)

SHIPTO(UIC, Ship_code, Customer,
Ship_add, Ship_info)

9. (PRICE_HI)

PRICE_HI(NSN_LSN, [Contra_nr], [Nomenclatu],
Quantity, Unit_issue, Unit_pr,
Prod_leadt)

10. (PRICE_CONTR)

PR_HI_CONTR(Contra_nr, Date_of_kt, Vendor,
Vendor_siz, K_size, Cust_uic,
Ref_nr)

11. (IDTC)

IDTC_CTR(Contr_nr, FSCM, Commod_ind,
Last_de_or, Eff_ord_dte_stt,
Eff_ord_dte_stp, Contr_name/add,
Admin_name/add, Disc_term,
Delivery, FOB, Provisions)

12. (IDTC_ITEM)

IDTC/AUTOBUY_ITEM([Contr_nr], [Nomenclatu],
Clin_subcl, Item_descr, Unit_issue,
Unit_pr, Min_ord_q, Max_ord_q)

13. (SOURCE)

SOURCE(FSCM, Commod_ind, Name_add,
Bus_cat, Agree_ind, Use_indica,
[Status])

14. (SOURCE_STAT)

SRCE_STATUS(Status, Deb_in_sus,
Explanatio)

15. (SOURCE_COMMOD)

SRCE_COMMOD(Name_add, Commod_ind)

16. (BUYERINFO)

BUYER(Buyer_code, Buyers_name,
Corresp_co, Phone_nr1, Phone_nr2,
Phone_ext, Buyer_grad, Occupat_se)

17. (AUTOBUY)

AUTOBUY_CTR(Contr_nr, Last_de_or, Qty,
Commod_ind, Nomenclatu)

18. (PID)

PID(NSN_LSN, Nomenclatu, Ref_no,
Descr_data, Dte_lst_use)

APADE MAINFRAME TABLES

1. (REFERRAL)

REFERRAL(Refer_no, Dte_of_Ref, Dte_compl)

2. (FSS_SKED)

FSS_SKED(Nomenclatu, Ref_no, Descr_data)

3. (CANCELLATION)

CANCELLATION(Doc_req_no, Canc_code, text)

4. (AMENDMENT)

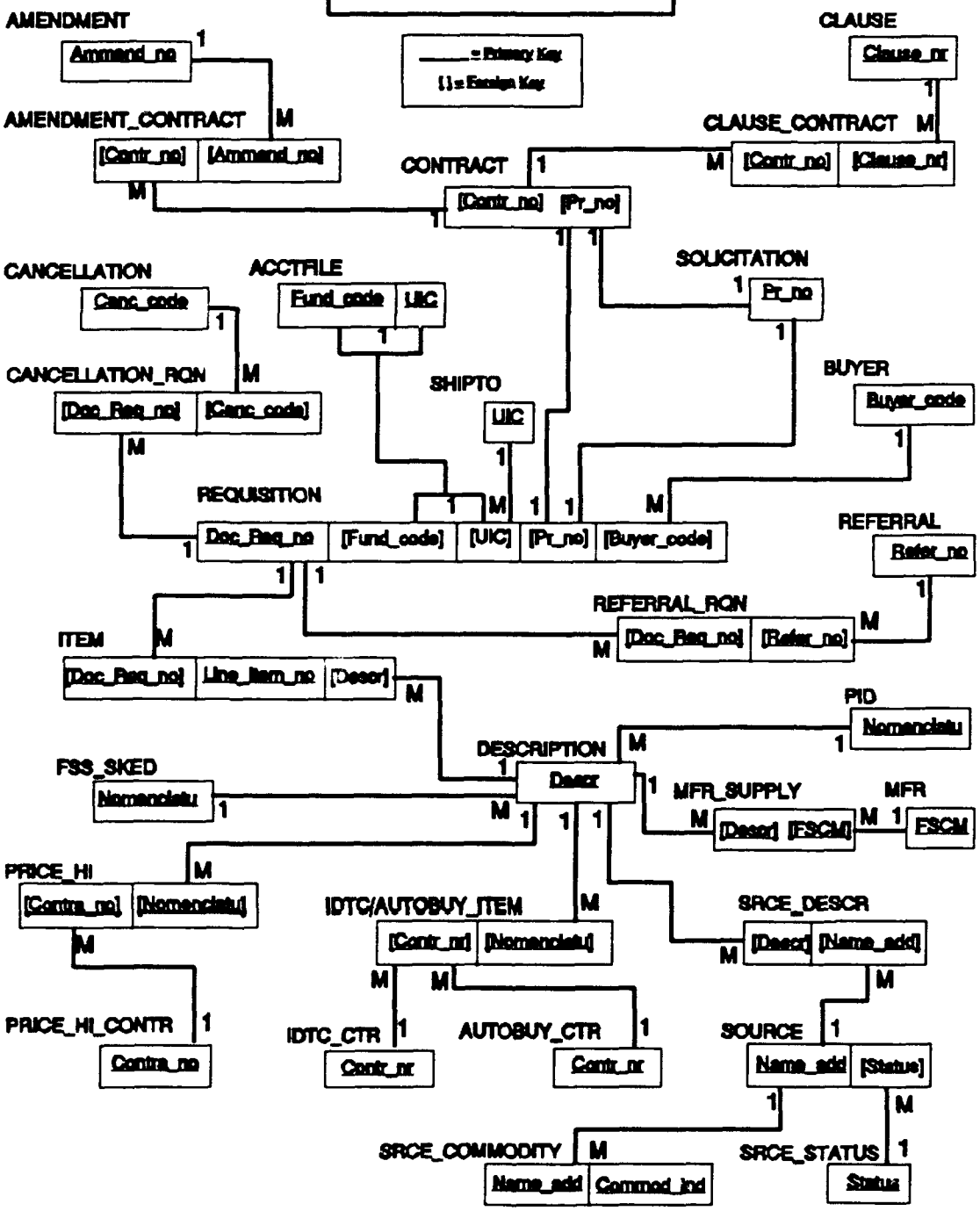
AMENDMENT(Contr_no, Ammend_no, text)

5. (SOLICITATION)

SOLICITATION(Pr_no, text)

Appendix E

Relational Mapping



APPENDIX F

FINAL RELATIONAL MODEL

1. REQUISITION(Doc_Req_No, Dist, Fund_expdt, Priority, Supp_add, Proj_code, RDD, Sig_code, SOS)

2. ACCTFILE(Appropriat, Fund_code, UIC, Bur_Contr_no, Cost_code, Auth_acct, Obj_class, Prop_acct, Sub_allot, Subhead, Transactio)

3. ITEM(Line_item_nr,Doc_req_no, Qty, Unit_iss, Est_price)

4. DESCRIPTION(Descr, APL_No, COG, ACRN, NSN_LSN, Color, Size, DMS, EIA)

5. MFR(FSCM, Mf_cat, Contr_name/add)

6. CLAUSE(Clause_no, Title_Date, Text)

7. SHIPTO(UIC, Ship_code, Customer, Ship_add, Ship_info)

8. PR_HI_CONTR(Contra_no, Date_of_kt, Vendor, Vendor_siz, K_size, Cust_uic, Ref_nr)

9. IDTC_CTR(Contr_no, FSCM, Last_de_or, Eff_ord_dte_stt, Eff_ord_dte_stp, Contr_name/add,Contr_name/add, Disc_term, Delivery, FOB, Provisions)

10. SOURCE(FSCM, Commod_ind, Name_add, [Status], Bus_cat, Agree_ind, Use_indica)

11. SRCE_STATUS(Status, FSCM, Deb_in_sus, Explanatio)

12. SRCE_COMMOD(Name_add, Commod_ind)

13. BUYER(Buyer_code, Buyers_name, Corresp_co, Phone_nr1, Phone_nr2, Phone_ext, Buyer_grad, Occupat_se)

14. AUTOBUY_CTR(Contr_no, Last_de_or, Nomenclatu, Qty, Commod_ind)

15. PID(NSN_LSN, Nomenclatu, Ref_no, Descr_data, Dte_1st_use)

16. REFERRAL(Refer_no, Dte_of_Ref, Dte_compl)

17. FSS_SKED(Ref_no, Descr_data)

18. CANCELLATION(Canc_no, text)

19. AMENDMENT(Ammend_no, text)

20. SOLICITATION(Pr_no, text)

21. CONTRACT(Contr_no, Date_of_cont, Delivery_dte, [Pr_no])

22. PRICE_HI(NSN_LSN, Contra_nr, Nomenclatu, Quantity, Unit_issue, Unit_pr, Prod_leadt)

23. IDTC/AUTOBUY_ITEM(Contr_no, Nomenclatu, Clin_subcl, Item_descr, Unit_issue, Unit_pr, Min_ord_q, Max_ord_q)

24. REFERRAL_RQN(Refer_no, Doc_Req_no)

25. CANCELLATION_RQN(Doc_Req_no, Canc_Code)

26. CLAUSE_CONTRACT(Contr_no, Clause_nr)

27. AMENDMENT_CONTRACT(Contr_no, Amend_no)

28. MFR_SUPPLY(FSCM, Descr, Mod, Pt_no, Serial, Series, Tech_man, Tech_ord)

29. SRCE_DESCR(Name_add, Descr)

APPENDIX G

FINAL ENTITY RELATIONSHIP DATA MODEL

Entities

1. REQUISITION(Doc_Req_No, Dist, Fund_expdt, Priority, Supp_add, Proj_code, RDD, Sig_code, SOS)

2. ACCTFILE(Appropriat, Fund_code, UIC, Bur_Contr_no, Cost_code, Auth_acct, Obj_class, Prop_acct, Sub_allot, Subhead, Transactio)

3. [WEAK] ITEM(Line_item_nr,Doc_req_no, Qty, Unit_iss, Est_price)

4. DESCRIPTION(Descr, APL_No, COG, ACRN, NSN_LSN, Color, Size, DMS, EIA)

5. MFR(FSCM, Mf_cat, Contr_name/add)

6. CLAUSE(Clause_no, Title_Date, Text)

7. SHIPTO(UIC, Ship_code, Customer, Ship_add, Ship_info)

8. PR_HI_CONTR(Contra_no, Date_of_kt, Vendor, Vendor_siz, K_size, Cust_uic, Ref_nr)

9. IDTC_CTR(Contr_no, FSCM, Last_de_or, Eff_ord_dte_stt, Eff_ord_dte_stp, Contr_name/add,Contr_name/add, Disc_term, Delivery, FOB, Provisions)

10. SOURCE(FSCM, Commod_ind, Name_add, Status, Bus_cat, Agree_ind, Use_indica)

11. SRCE_STATUS(Status, FSCM, Deb_in_sus, Explanatio)

12. SRCE_COMMOD(Name_add, Commod_ind)

13. BUYER(Buyer_code, Buyers_name, Corresp_co, Phone_nr1, Phone_nr2, Phone_ext, Buyer_grad, Occupat_se)

14. AUTOBUY_CTR(Contr_no, Last_de_or, Nomenclatu, Qty, Commod_ind)

15. PID(NSN_LSN, Nomenclatu, Ref_no, Descr_data, Dte_lst_use)

16. REFERRAL(Refer_no, Dte_of_Ref, Dte_compl)

17. FSS_SKED(Ref_no, Descr_data)

18. CANCELLATION(Canc_no, text)

19. AMENDMENT(Ammend_no, text)

20. SOLICITATION(Pr_no, text)

21. CONTRACT(Contr_no, Date_of_cont, Delivery_dte, [Pr_no])

Relationships

1. SUPPLIES(NSN_LSN, Quantity, Unit_issue, Unit_pr, Prod_leadt)

2. CONTAINED(Clin_subcl, Item_descr, Unit_issue, Unit_pr, Min_ord_q, Max_ord_q)

3. SUSPENDS

4. CANCELS

5. DEFINES

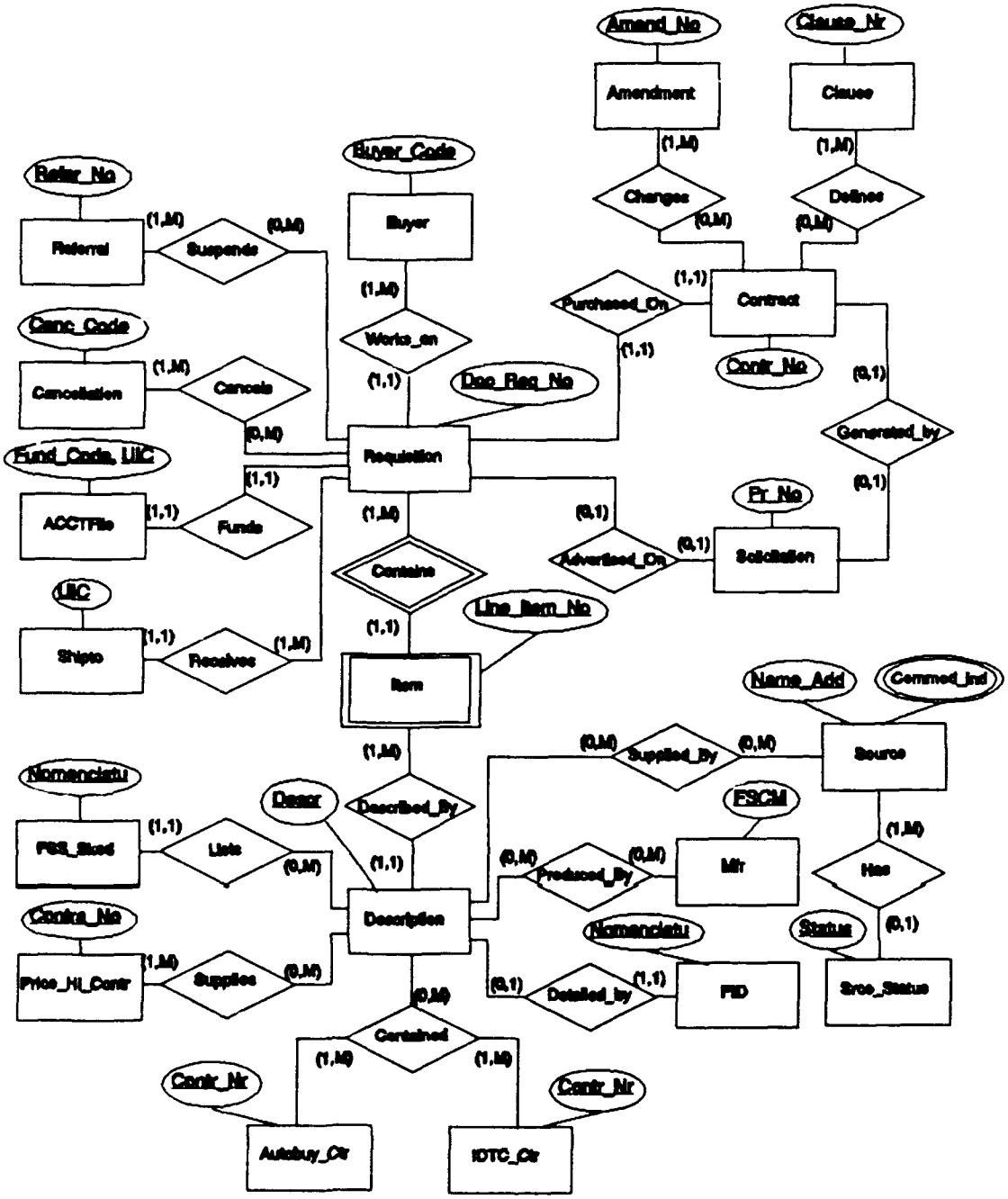
6. CHANGES

7. PRODUCED_BY(Mod, Pt_no, Serial, Series, Tech_man, Tech_ord)

8. SUPPLIED_BY

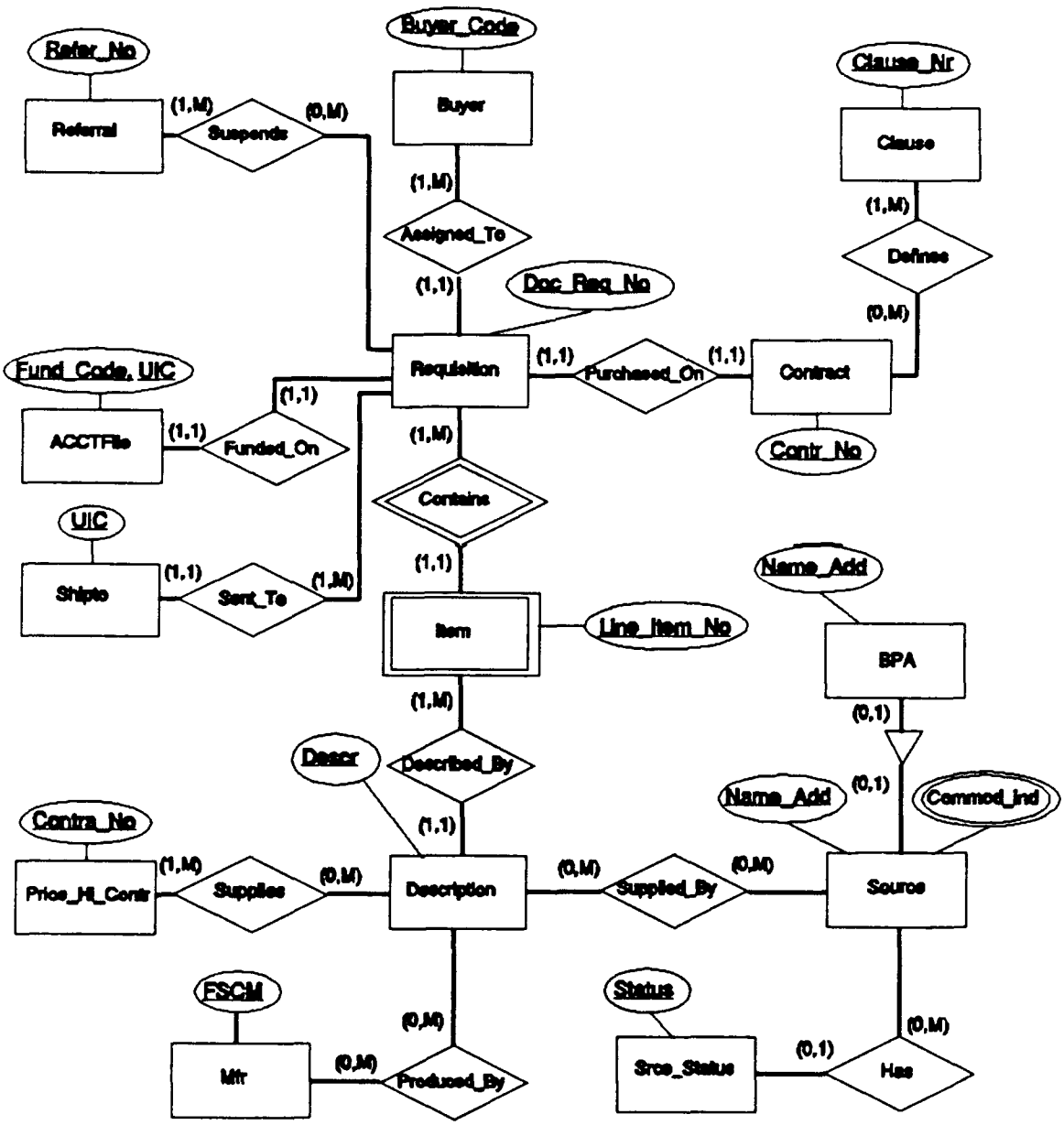
Appendix H

APADE ERD



Appendix I

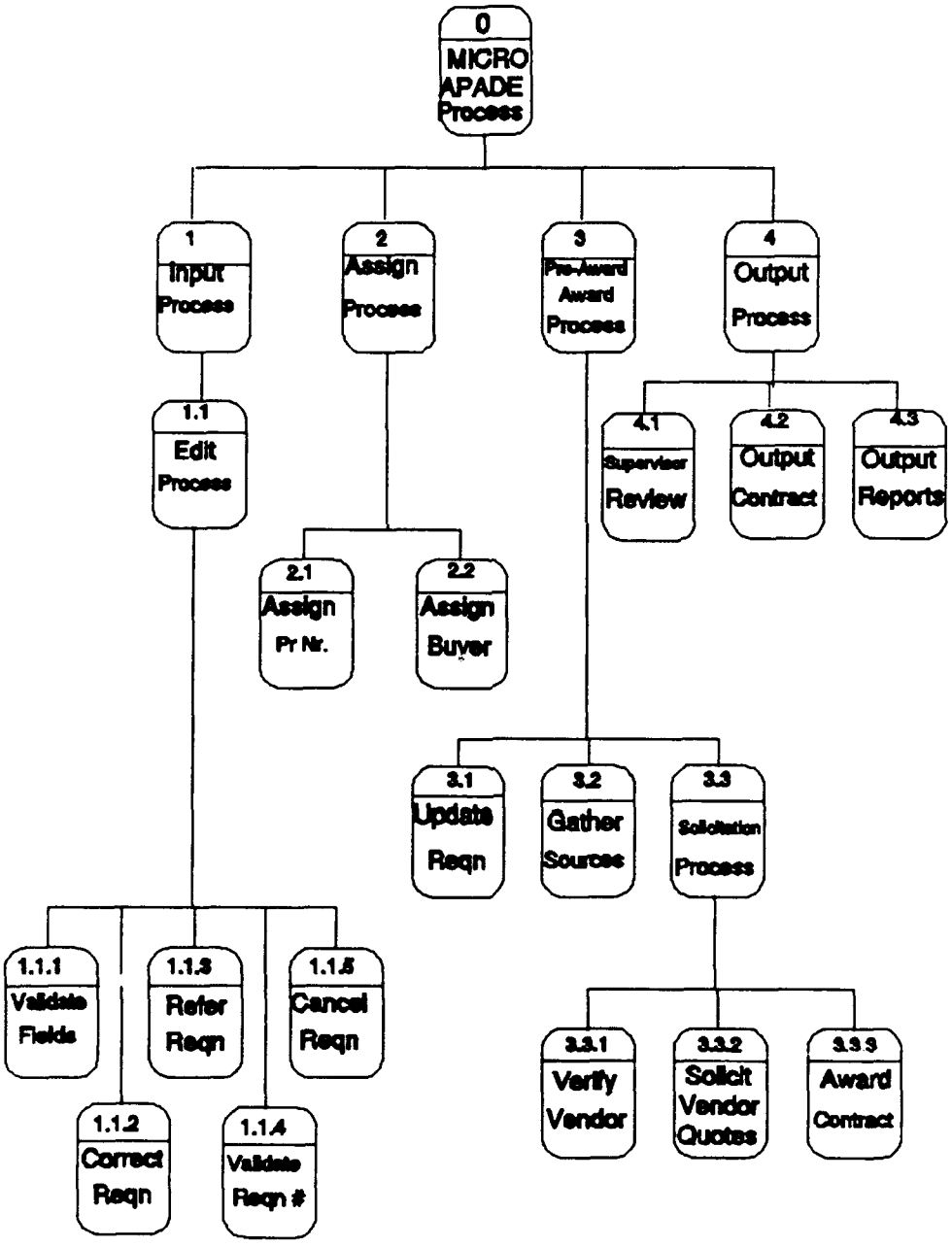
MICRO-APADE ERD



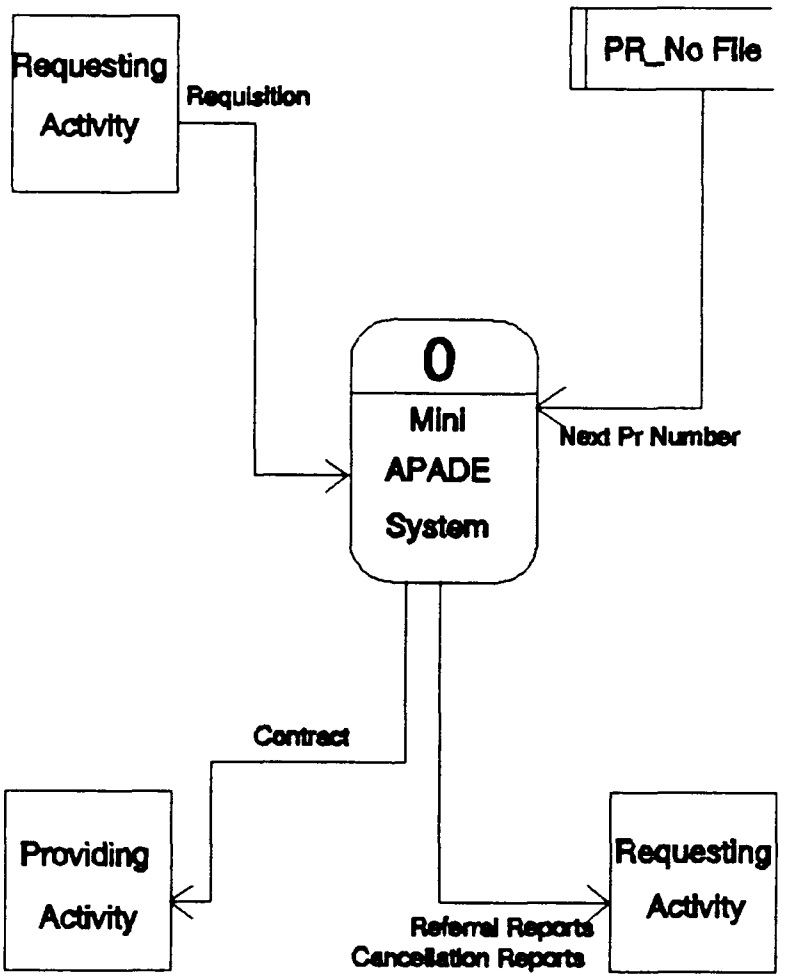
Appendix J

MICRO-APADE Data Flow Diagrams

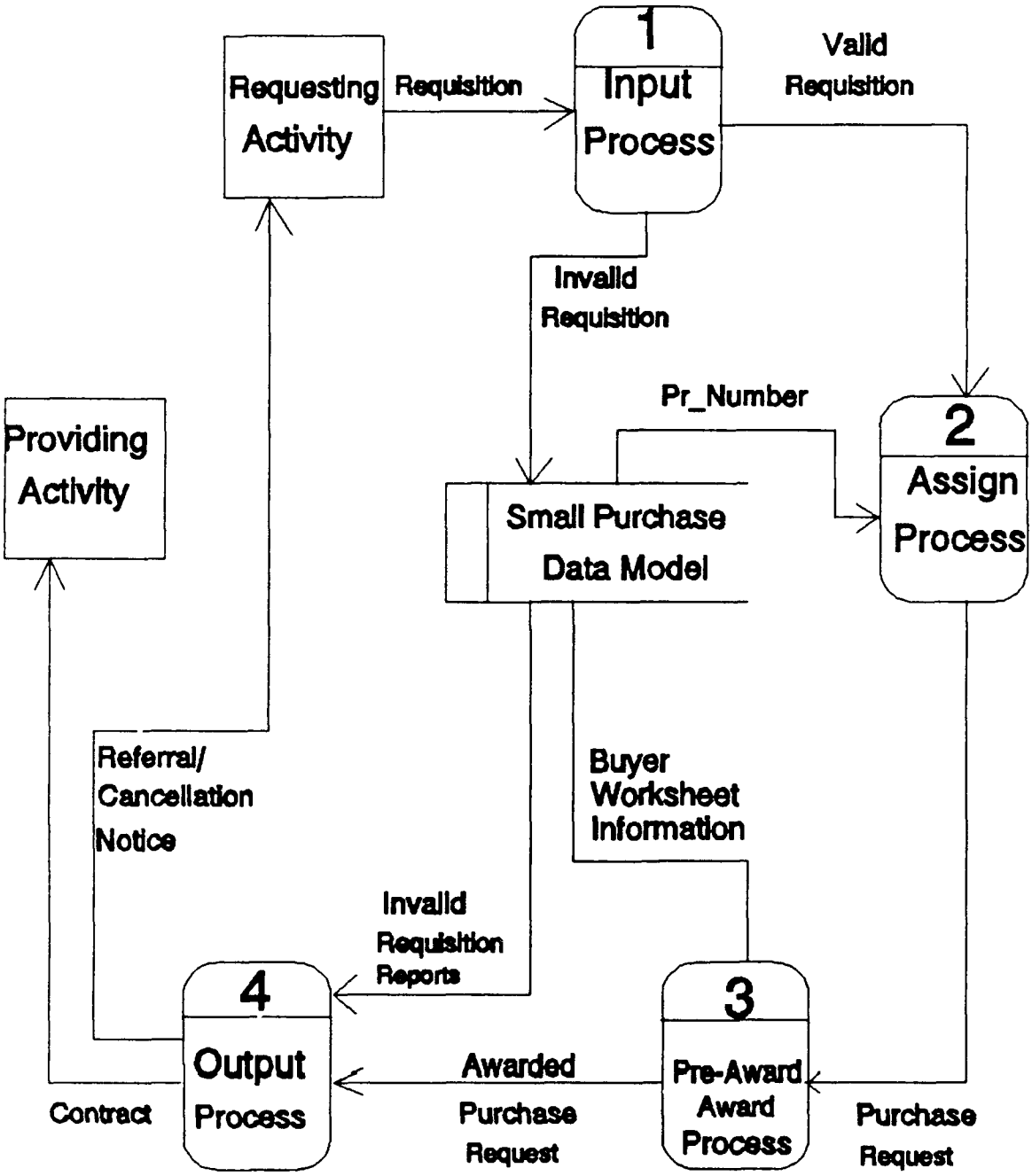
Decomposition Diagram



Context Diagram

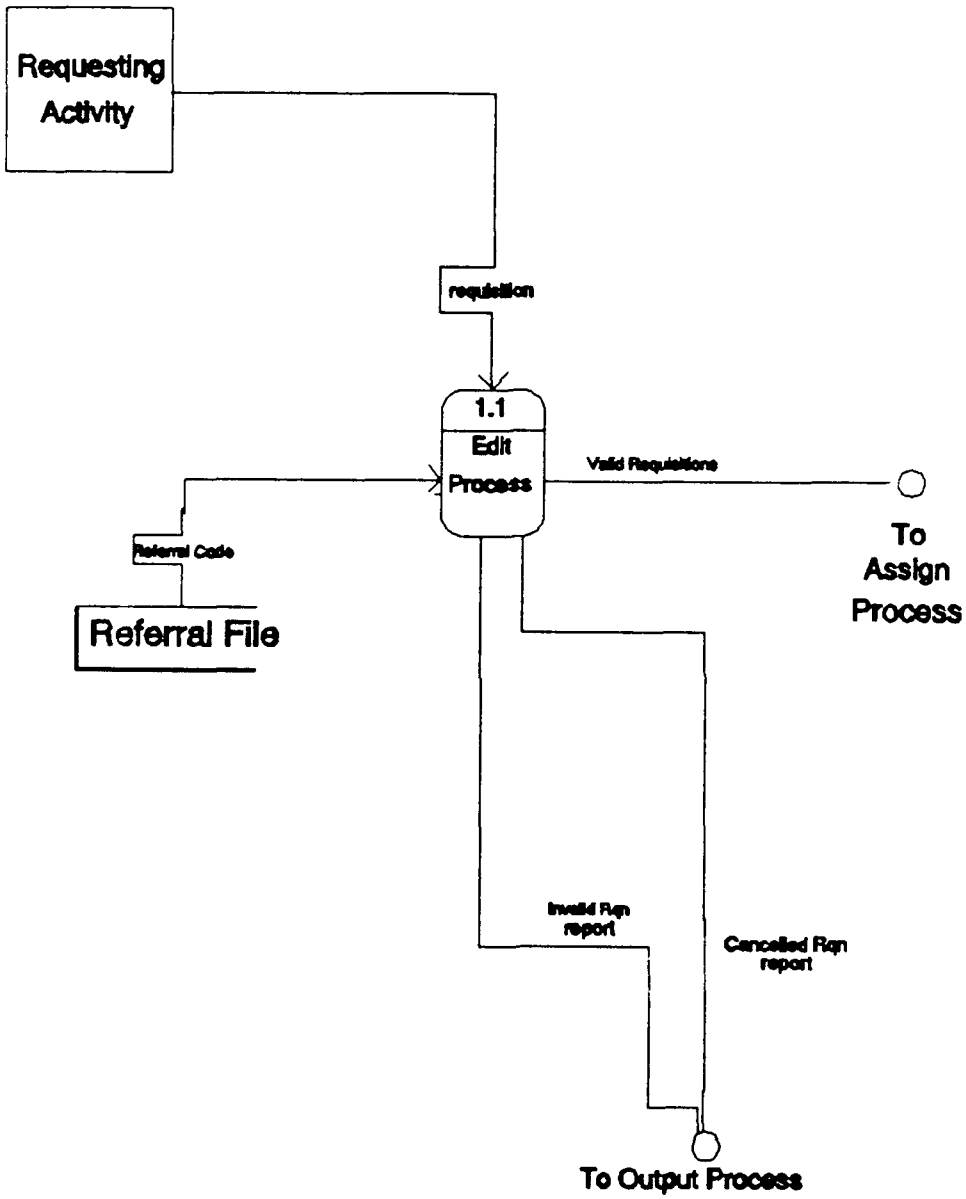


Systems Diagram

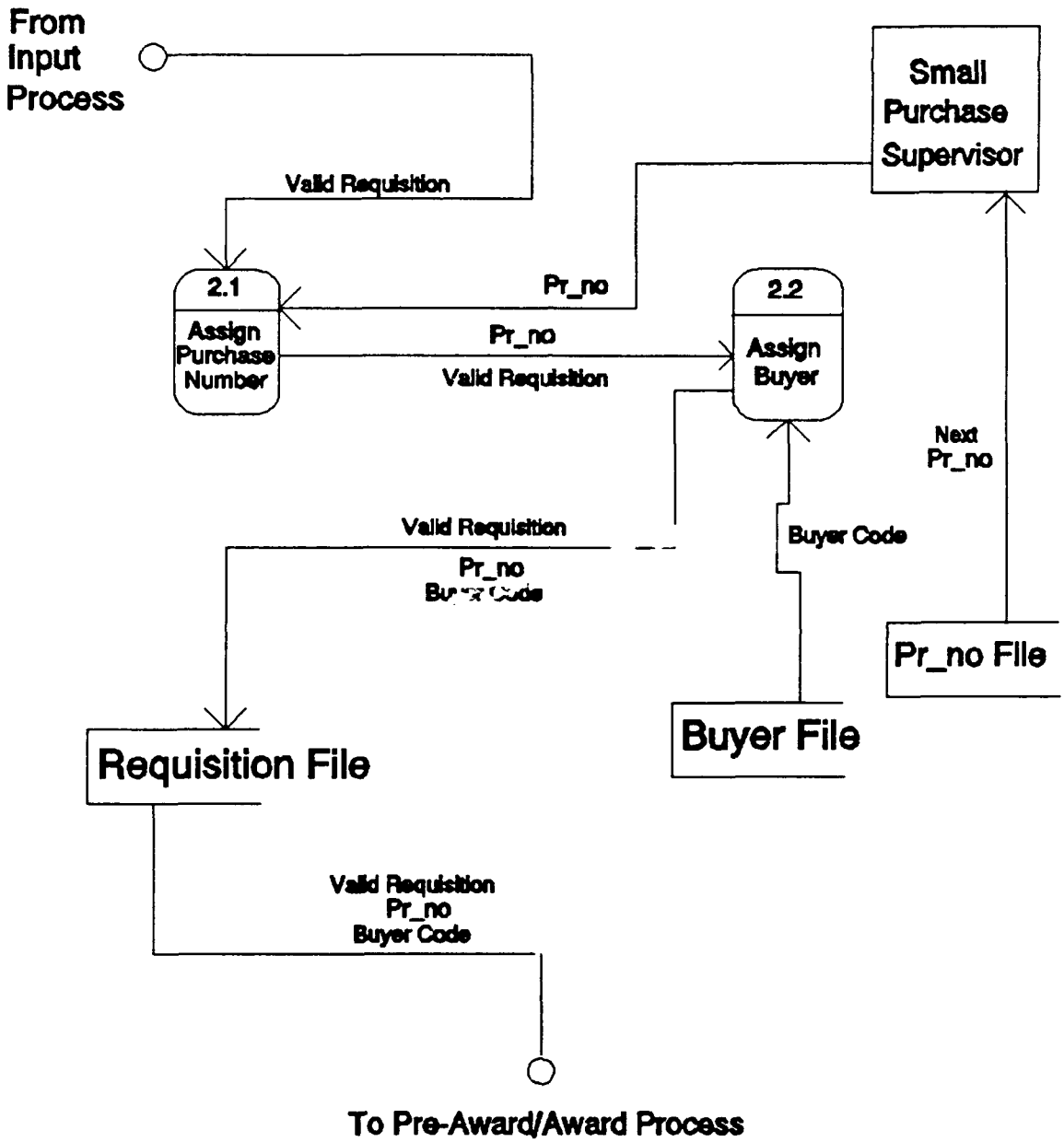


Input Process DFD

(Mid-Level)

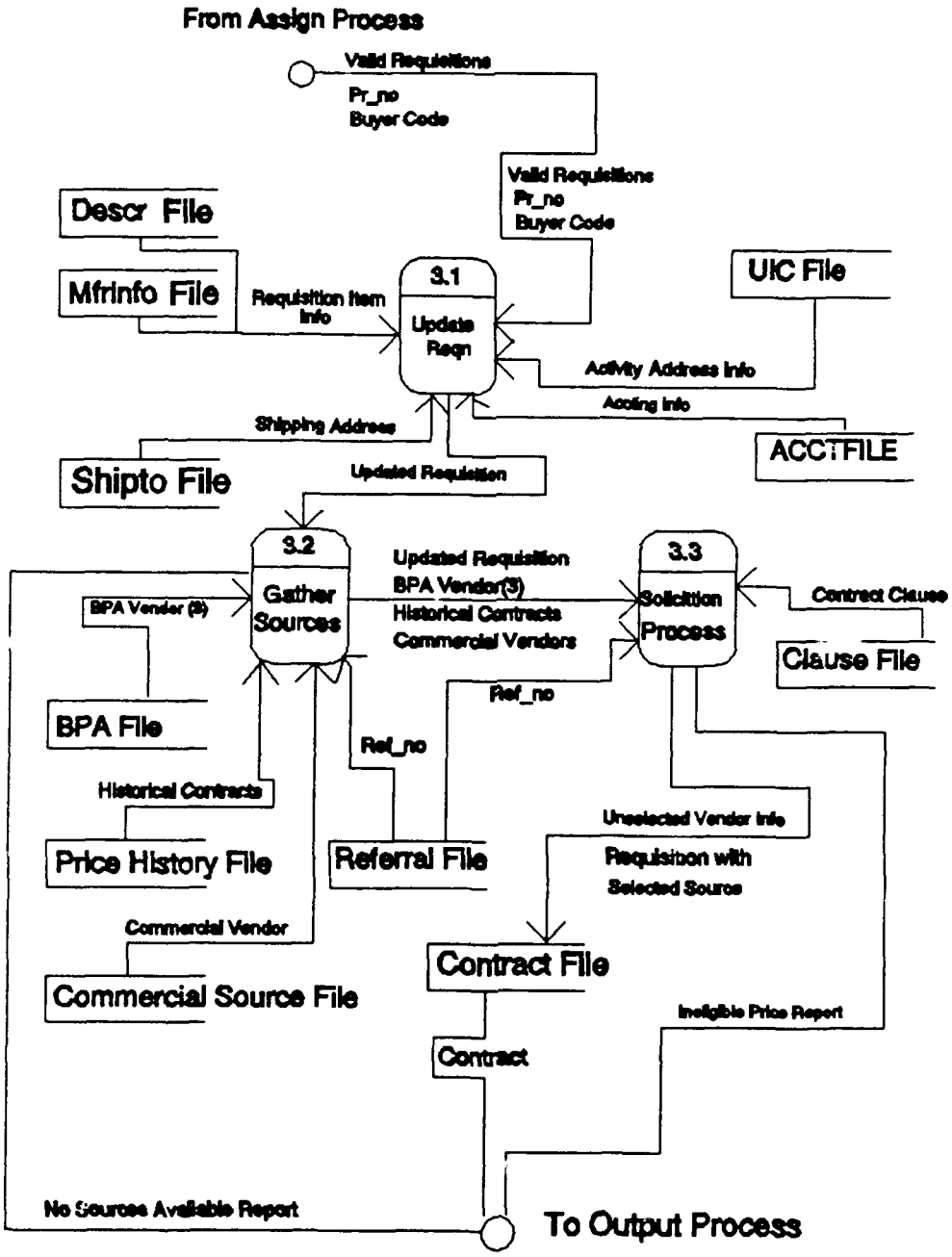


Assign Process DFD

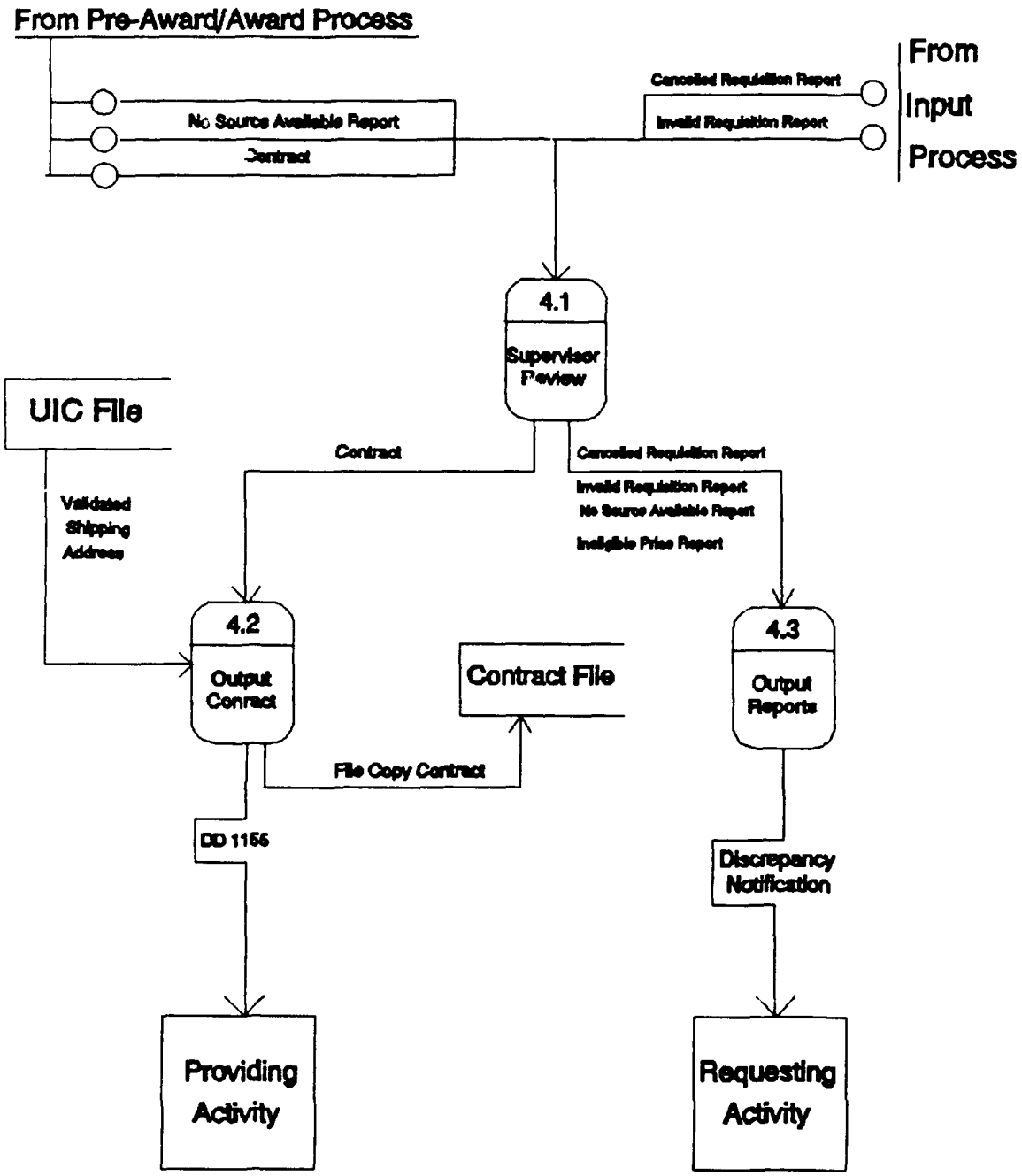


Pre-Award/Award Process DFD

(Mid-Level)

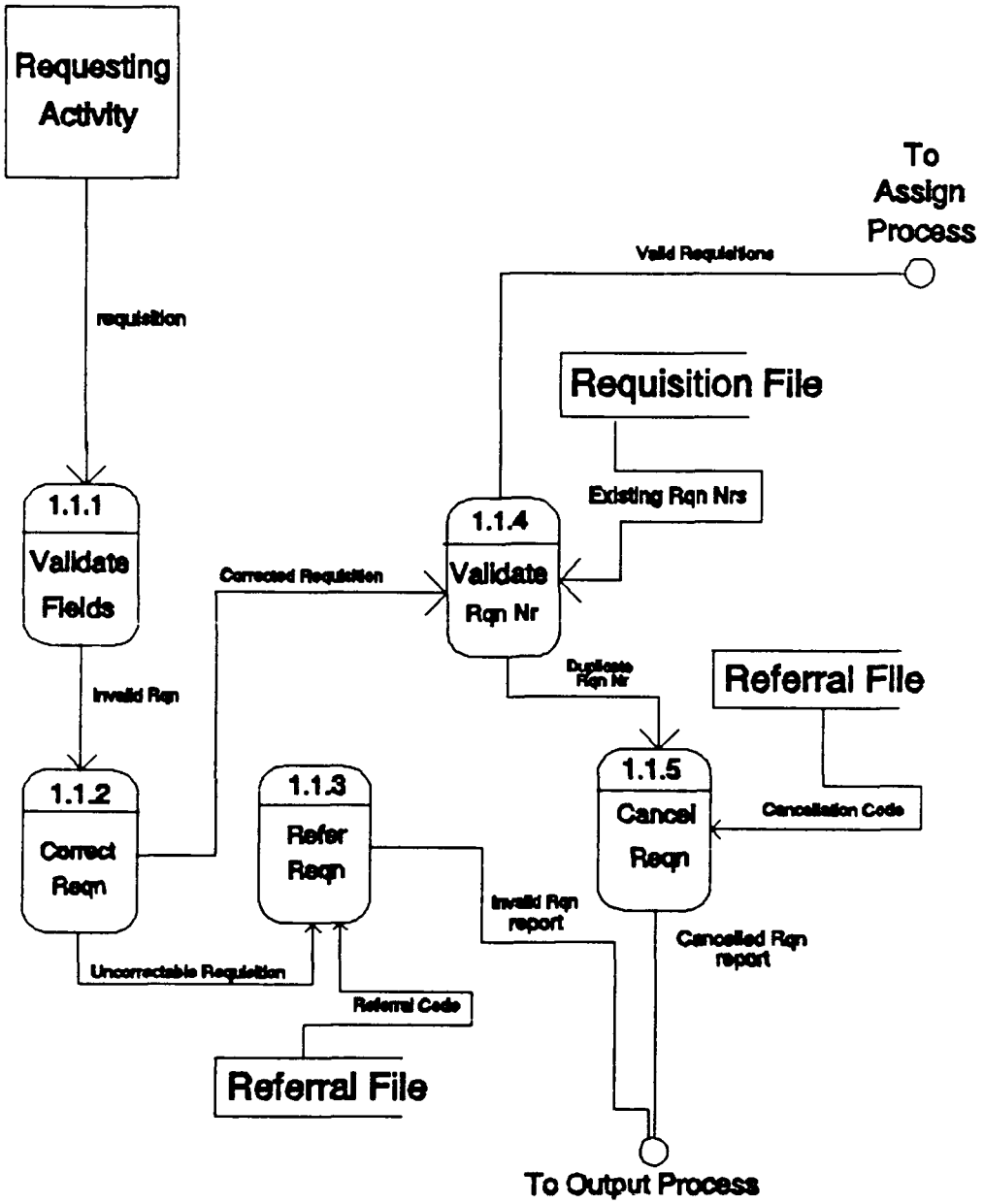


Output Process DFD



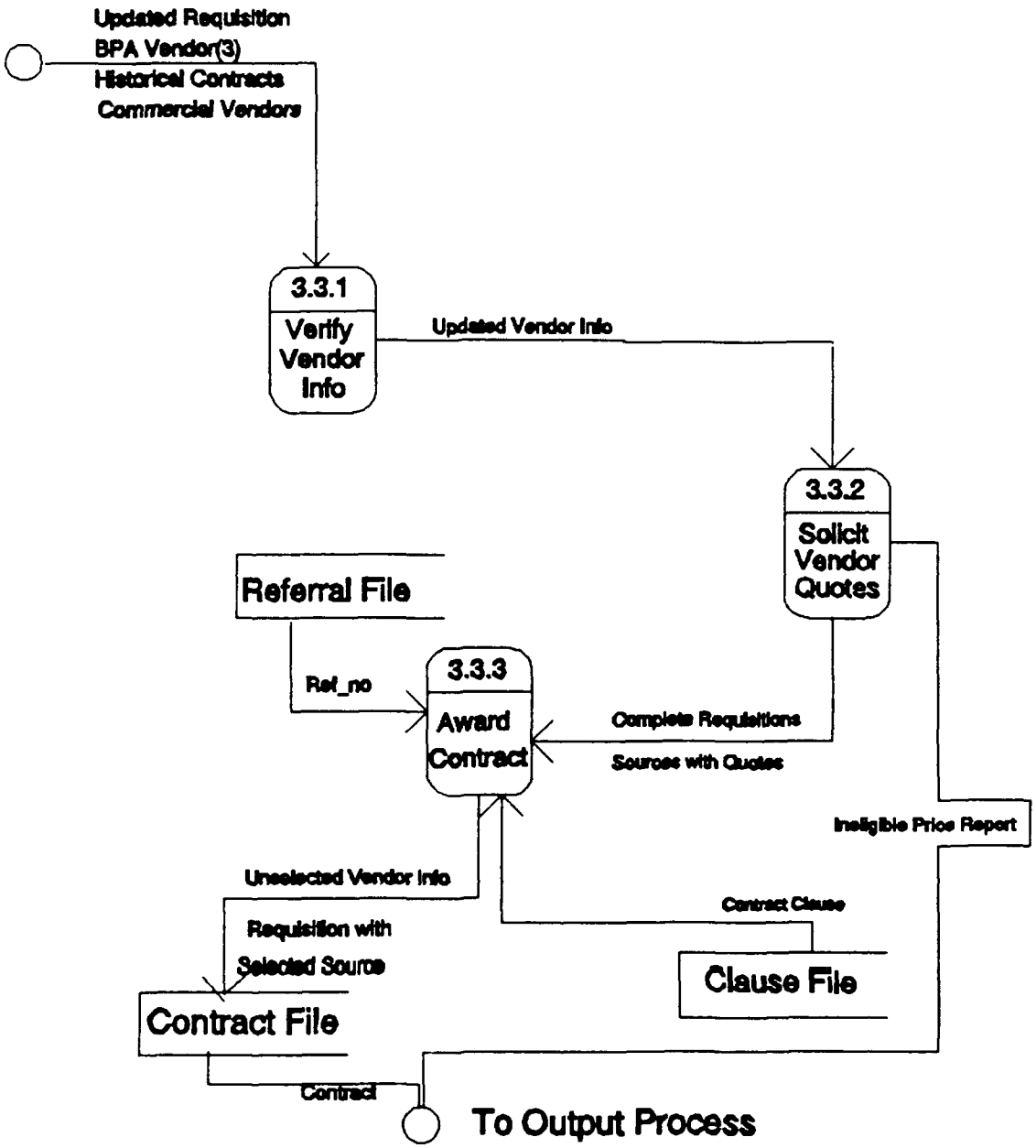
Input Process DFD

(Lower Level)



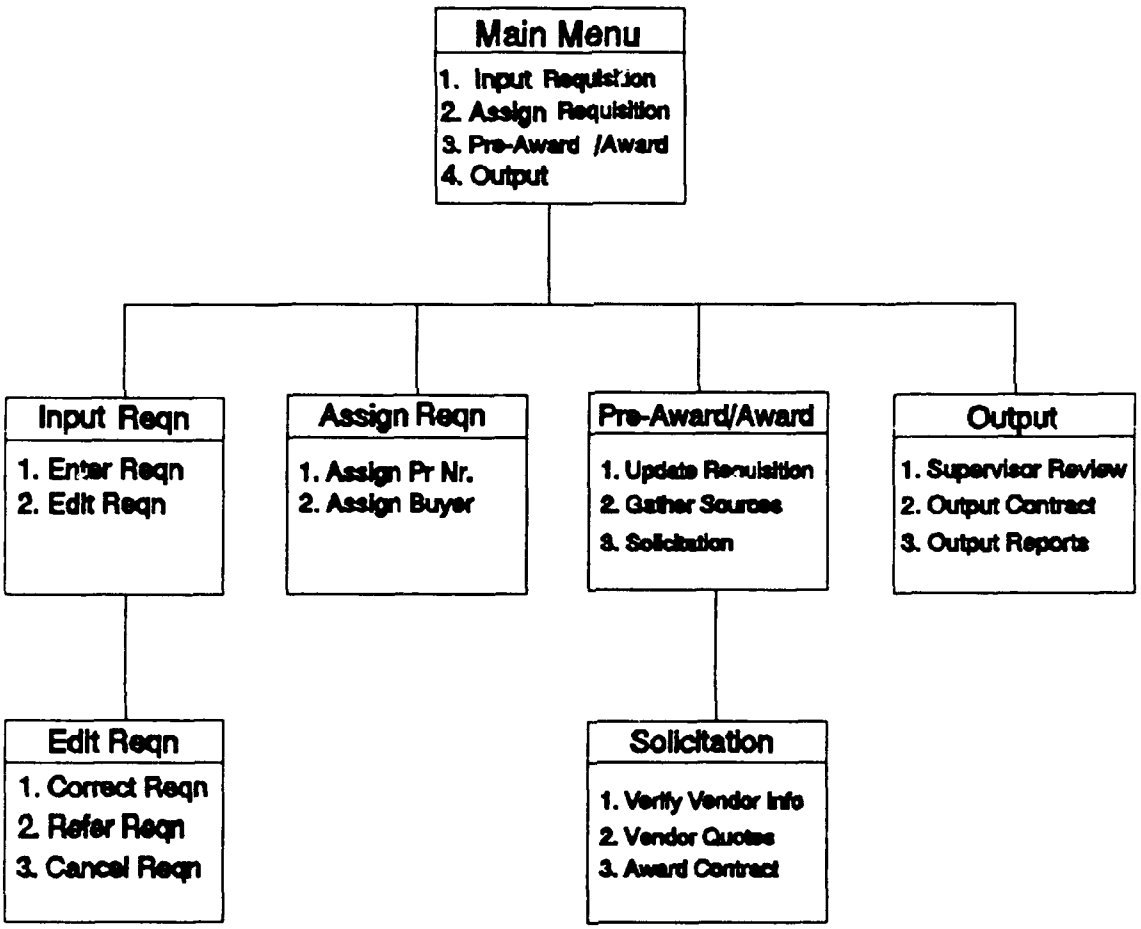
Pre-Award/Award Process DFD

(Lower Level)



Appendix K

Menu Hierarchy



APPENDIX L

MICRO-APADE RELATIONAL MODEL

1. REQUISITION(Doc_Req_No, Dist, Fund_expdt, Priority, Supp_add, Proj_code, RDD, Sig_code, SOS)

2. ACCTFILE(Appropriat, Fund_code, UIC, Bur_Contr_no, Cost_code, Auth_acct, Obj_class, Prop_acct, Sub_allot, Subhead, Transactio)

3. ITEM(Line_item_nr,Doc_req_no, Qty, Unit_iss, Est_price)

4. DESCRIPTION(Descr, APL_No, COG, ACRN, NSN_LSN, Color, Size, DMS, EIA)

5. MFR(FSCM, Mf_cat, Contr_name/add)

6. CLAUSE(Clause_no, Title_Date, Text)

7. SHIPTO(UIC, Ship_code, Customer, Ship_add, Ship_info)

8. PR_HI_CONTR(Contra_no, Date_of_kt, Vendor, Vendor_siz, K_size, Cust_uic, Ref_nr)

9. SOURCE(FSCM, Commod_ind, Name_add, [Status], Bus_cat, Agree_ind, Use_indica)

10. SRCE_STATUS(Status, FSCM, Deb_in_sus, Explanatio)

11. SRCE_COMMOD(Name_add, Commod_ind)

12. BUYER(Buyer_code, Buyers_name, Corresp_co, Phone_nr1, Phone_nr2, Phone_ext, Buyer_grad, Occupat_se)

13. REFERRAL(Refer_no, Dte_of_Ref, Dte_compl)

14. SOLICITATION(Pr_no, text)

15. CONTRACT(Contr_no, Date_of_cont, Delivery_dte, [Pr_no])

16. PRICE_HI(NSN_LSN, Contra_nr, Nomenclatu, Quantity, Unit_issue, Unit_pr, Prod_leadt)

17. BPA(Name_add, Agree_ind, Staus)

18. REFERRAL_RQN(Refer_no, Doc_Req_no)

19. CLAUSE_CONTRACT(Contr_no, Clause_nr)

20. MFR_SUPPLY(FSCM, Descr, Mod, Pt_no, Serial, Series, Tech_man, Tech_ord)

21. SRCE_DESCR(Name_add, Descr)

REFERENCES

1. Christopher Drake and Steven Carver, Alternatives in Automating Small Procurement Field Activities; A Cost-Benefit Analysis, Master's Thesis, Naval Postgraduate School, Monterey, California, June, 1986.

2. James Martin, "Reverse-Engineering Gives Old Systems New Lease On Life," PCWEEK, April 16, 1990.

3. James Martin, "Making The Case For True Reverse Engineering Tools," PCWEEK, April 23, 1990.

4. United States Government, Federal Acquisition Regulation (FAR), Commercial Clearing House, Inc., 1986.

5. Logistics Management Institute Report PL804R1, Greater Buyer Effectiveness Through Automation, by Daniel J. Drake, January 1989.

6. Elmasri, R. and Navathe, S., Fundamentals of Database Systems, Benjamin/Cummings Publishing Co., Inc., 1989.

BIBLIOGRAPHY

Bachman, Charlie, "A Case For Reverse Engineering", Datamation, 1 July 1988.

Dobler, D.W., Lee, L., and Burt, D.N., Purchasing and Materials Management, 4th ed., McGraw-Hill Book Co., 1984.

Elmasri, R. and Navathe, S., Fundamentals of Database Systems, Benjamin/Cummings Publishing Co., Inc., 1989.

Jones, Edward, DBASE IV, Programmer's Reference Guide, Howard W. Sams & Company, 1989.

Simpson, Alan, DBASE IV, User's Desktop Companion, SYBEX Publishing Co., 1989.

U. S. Department of the Navy, Fleet Material Support Office, Automation of Procurement and Accounting Data Entry System, Functional Description, Mechanicsburg, Pennsylvania, 21 June 1985.

U. S. Department of the Navy, Naval Supply Systems Command, System Decision Paper Milestone II for the Automation of Procurement and Accounting Data Entry (APADE) System, Washington, D. C., 7 March 1986.

U. S. Department of the Navy, Regional Contracting Department, Naval Supply Center Oakland, Defense Small Purchase Course, February 1986.

U. S. Department of the Navy, Supply Department, Naval Ordnance Station, Indian Head, Maryland, Acquisition Handbook, 1988.