

NAVAL POSTGRADUATE SCHOOL Monterey, California

AD-A186 902



DTIC
ELECTE
DEC 28 1987
S D
H

THESIS

SHIPBOARD AMMUNITION MANAGEMENT SYSTEM:
A DATABASE APPROACH

by

Steven L. Smith

September 1987

Thesis Advisor
Co-Advisor

N. R. Lyons
Y. Mortagy

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

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		5 MONITORING ORGANIZATION REPORT NUMBER(S)	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)		7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b OFFICE SYMBOL (if applicable) 54	7b ADDRESS (City, State and ZIP Code) Monterey, California 93943-5000	
8a NAME OF FUNDING SPONSORING ORGANIZATION		9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8b 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) SHIPBOARD AMMUNITION MANAGEMENT SYSTEM: A DATABASE APPROACH			
12 PERSONAL AUTHOR(S) Smith, Steven L.			
13a TYPE OF REPORT Master's Thesis	10 TIME COVERED FROM	14 DATE OF REPORT (Year, Month, Day) 1987 September	15 PAGE COUNT 187
16 SUPPLEMENTARY NOTES			
17 COSAT CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB GROUP	
		Ammunition Management, Ammunition Inventory Management, Database Inventory System, Automated Inventory System	
19 ABSTRACT (Continue on reverse if necessary and identify by block number)			
<p>This thesis concerns the analysis, design, and partial implementation of a software package to automate the present manual system of conventional ammunition management onboard most ships of the U.S. Navy. Structured analysis and design techniques are utilized in the development and approximately one quarter of the application programs have been implemented.</p> <p>The system is designed for stand alone operation on an IBM compatible microcomputer using the relational database package dBase III Plus by Ashton-Tate.</p> <p>Follow-on work would consist of completing the application programs, select a pilot vessel and install the system, collect user comments, and modify the system as necessary. <i>(Key words)</i></p>			
20 DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS		21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL Professor Norman R. Lyons		22b TELEPHONE (include Area Code) (408) 646-2666	22c OFFICE SYMBOL 54Lb

Approved for public release; distribution is unlimited.

Shipboard Ammunition Management System:
A Database Approach

by

Steven L. Smith
Lieutenant, United States Navy
B.S., Washington State University, 1979

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1987

Author:

Steven L. Smith
Steven L. Smith

Approved by:

N.R. Lyons
N.R. Lyons, Thesis Advisor

Y. Mortagy
Y. Mortagy, Co-Advisor

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

James M. Fremgen
James M. Fremgen
Acting Dean of Information and Policy Sciences

ABSTRACT

This thesis concerns the analysis, design, and partial implementation of a software package to automate the present manual system of conventional ammunition management onboard most ships of the U.S. Navy. Structured analysis and design techniques are utilized in the development and approximately one quarter of the application programs have been implemented.

The system is designed for stand alone operation on an IBM compatible microcomputer using the relational database package dBase III Plus by Ashton-Tate.

Follow-on work would consist of completing the application programs, select a pilot vessel and install the system, collect user comments, and modify the system as necessary.



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

TABLE OF CONTENTS

I.	INTRODUCTION	9
A.	PURPOSE	10
B.	MOTIVATION	11
C.	CHAPTER OUTLINE	12
II.	PRESENT MANUAL AMMUNITION MANAGEMENT / PROBLEMS	14
A.	INVENTORY RECORDS	16
1.	Master Stock Record Card	16
2.	Lot/Location Card	19
3.	Serial/Location Card	19
4.	Discussion of Inventory Record	22
B.	REQUISITION, TURN-IN, TRANSFER, RECEIPT	24
1.	Background	24
2.	Requisitioning	25
3.	Turn-in/Transfer	29
4.	Receipt	30
5.	Discussion	30
C.	AMMUNITION TRANSACTION REPORTING	30
D.	COMMENTS	32
III.	AN AUTOMATED SYSTEM DEVELOPEMENT	36
A.	METHODOLOGY	36
B.	ANALYSIS	38
1.	Data Flow Diagrams	39
2.	Subsystem Functional Description	40
3.	System Data	43
C.	DATA ORGANIZATION	46
1.	File Processing	46

2.	Database Processing	46
IV.	SAMS DESIGN	51
A.	SAMS PHYSICAL RELATION DESCRIPTION	52
1.	Transaction File (ATR File)	52
2.	Requisition File	56
3.	Turn-in File	57
4.	Ammunition Data	57
5.	Inventory File	58
6.	NAR Action File	58
7.	NAR Serial File	59
8.	Allowances File	59
9.	Address File	59
10.	Static Data File	60
B.	SAMS STRUCTURE CHARTS	60
V.	IMPLEMENTATION	68
A.	IMPLEMENTATION PROGRESS TO DATE	68
B.	CODING STYLE	69
VI.	CONCLUSIONS AND RECOMMENDATIONS	70
APPENDIX A:	DATA FLOW DIAGRAMS	72
APPENDIX B:	DATA ELEMENT DICTIONARY	107
APPENDIX C:	SAMS PROGRAM DIRECTORY	122
APPENDIX D:	SAMS STRUCTURE CHARTS	127
APPENDIX E:	SAMS PROGRAM LISTINGS	137
1.	AMSMAN.PRGM	137
2.	BCKUPREQ.PRGM	138
3.	COMPLREQ.PRGM	139
4.	CRENEWREQ.PRGM	141
5.	DATAELEM.PRGM	154
6.	EDITREQ.PRGM	156
7.	MISCREQ.PRGM	158

8.	PRINTREQ.PRG	159
9.	PROGFILE.PRG	171
10.	REQ1348.PRG	172
11.	REQMENU.PRG	174
12.	REVVADD.PRG	175
13.	REVVREQ.PRG	176
14.	SPELLIT.PRG	177
15.	STRUCCRT.PRG	177
LIST OF REFERENCES		182
BIBLIOGRAPHY		184
INITIAL DISTRIBUTION LIST		186

LIST OF TABLES

1. MISCELLANEOUS REQUISITION CODES	28
2. SAMS OUTPUT DOCUMENT DATA ELEMENTS	44
3. DATA ELEMENT DICTIONARY FORMAT	47
4. SAMS FILE NAMING CONVENTIONS	53
5. SAMS FILE GROUPS	55
6. SAMS PHYSICAL FILE STRUCTURES	61

LIST OF FIGURES

2.1	National Stock Number breakdown	14
2.2	Department of Defense Ammunition Code	15
2.3	Identification Number Composition	15
2.4	Ammunition Master Stock Record Card	17
2.5	Document Number	18
2.6	Ammunition Lot Location Card	20
2.7	Ammunition Serial Location Card	21
2.8	Ammunition Stock Card Relationships	23
2.9	Manual Requisition Document	26
2.10	DAAS Message Requisition Format	27
2.11	Naval Message Requisition Format	29
2.12	Manual Turn-in Document	31
2.13	Ammunition Transaction Report (example)	33
2.14	Ammunition Transaction Report Formats	34
3.1	SAMS Context Diagram	39
3.2	File Processing System for SAMS	48
3.3	Database System for SAMS	49

I. INTRODUCTION

A continuous chain of ammunition, fuel and equipment is the life blood of a military organization in wartime. The incredible speed and attrition of modern warfare requires the proper placement, both in type and quantity, of ammunition in peacetime to meet the expected demands of a major conflict.

The world-wide commitments of the United States and its allies dictate a logistics system which is exceedingly complex. The global stockpiling of conventional ammunition is perhaps one of the more complex problems military planners must solve. Not only does this stock require very accurate accounting and physical security, but it is perishable in nature and its serviceability must be continually reviewed. During the Vietnam war, Navy ammunition procurement reached a high of \$988 million [Ref. 1: p. 24] with world-wide inventories valued at approximately \$7 billion dollars. It was during this period that the Chief of Naval Operations directed the establishment of a single point of reference within the Navy for conventional ammunition management. The Chief of Naval Material was given the responsibility for the establishment of a Conventional Ammunition Integrated Management System (CAIMS). One of the prime operational purposes of CAIMS [Ref. 2], was to improve the quality of ammunition stock status reaching higher echelon logistics planners. CAIMS is the point of reference regardless of inventory management or ownership responsibilities. Perhaps most important to the end-user, it was the stated policy of CAIMS to minimize the reporting burden of field activities. For example, ships were to report expenditure and inventory information to CAIMS only, and CAIMS would further distribute the information as necessary to other interested parties.

CAIMS was established and is directed by the Navy Ship's Parts Control Center (SPCC) at Mechanicsburg, PA. Program guidance is promulgated by SPCC [Ref. 3] and is further defined with specific implementing and reporting instructions in Fleet Commander instructions [Refs. 4,5].

The program has been successful in many respects, however several audits in the early 1980's conducted by the US General Accounting Office (GAO) and the Naval Audit Service found significant discrepancies between on-site local records and CAIMS data. This brought into question the Navy's ability to maintain accountability for its

57 billion plus ammunition inventory [Refs. 6,7,8]. Future negotiations for ammunition appropriations depend on the mutual assurance of a credible inventory management system. The timely and accurate reporting from the hundreds of field activities and ships is critical therefore to the success of the overall system.

Although CAIMS as a whole has become highly automated, linking SPCC, the major stock points, and other organizations in the logistics hierarchy, the majority of end-users enjoy no such capabilities.

A. PURPOSE

This thesis will suggest a method to automate the present manual record keeping, report writing, and inventory control procedures in use on nearly all US Navy surface and submerged combatants. It is felt that the present procedures contribute to the high error rate in transaction reporting and inventory management. For activities that can dedicate an individual or individuals to the sole task of properly keeping the necessary records and generating the required reports, automation may seem unnecessary. However, the proposed system will increase the accessibility of ammunition and inventory information, greatly reduce storage requirements, and reduce the time required to manage the system. Therefore any activity should benefit and avail itself of a properly designed system that satisfies the functional requirements. The fact is, most ships do not have the luxury of assigning an individual to study this system and become an expert. The list of important administrative and record keeping tasks on a typical naval vessel often exceeds the crew size by a large margin. This thesis then will also attempt to lessen the administrative burden on weapons department, and in some cases supply department, personnel who are charged with the responsibility of conventional ammunition management.

This automated software solution shall be called the Shipboard Ammunition Management System (SAMS) and shall encompass all areas of routine ammunition management. The scope of this thesis will carry the project through the analysis, design, and partial implementation of critical areas of the application programs. Complete implementation and additional research shall be discussed in Chapters 5 and 6. It was particularly desired to start the implementation in this thesis because many good ideas never seem to bridge the void between paper and code on a project as restricted in time as a thesis necessarily is.

B. MOTIVATION

The impetus for this project came from the observation of the need by this author during several tours of duty on submarines of the Atlantic and Pacific Fleets. Although the quantities of conventional ammunition carried onboard submarines is considerably less, in quantity and variety, than most surface combatants, it was still obvious that an automated system could greatly improve the efficiency of the system. As discussed earlier, the many necessary administrative functions onboard a vessel generally do not decrease in proportion to crew size, so many smaller vessels suffer more in this respect. Additionally, enlisted rate training is particularly brief in the areas of conventional ammunition management [Ref. 9: p. 12-1]. Officer training is essentially nil in this area also. Therefore, the accurate and timely reports that CAIMS requires to maintain an accounting of Navy assets is being furnished by people with little time or training to become proficient in yet another administrative task. Now obviously, shipboard personnel are using the available publications and are supplying reasonably accurate information to the CAIMS system otherwise there would be much higher level attention to this problem. This author contends that SAMS will make more time available for important operational and weapons employment training.

Automating a previously manual task requires consideration of the operator's ability to operate the system by back up methods when necessary. The automated system must not be so abstract and "automatic" that the manual skills and knowledge of the underlying procedures are forgotten. Therefore, any system of this type must be instructive as well as functional. Such a system has elements of an expert system. It is developed by previous users who have acquired the proper education and training to implement a software solution, and pass on that expertise to the current users while satisfying the functional requirements. This type of user interface could become extremely tedious if the system is used frequently and so a balance must be struck between efficiency of data entry and the degree of explanatory information.

Lastly, it is unfortunate that the very personnel who require relief from administrative burdens often have neither the time or training to affect the change to automation. Thus it is particularly appropriate that shore commands and the Naval Postgraduate School solve these types of problems in addition to more basic research. The making available of time to train in relevant warfare topics is at the heart of current drives to reduce administrative and paperwork tasks within the Navy.

C. CHAPTER OUTLINE

Chapter 1 has discussed the importance of accountability and inventory management of conventional ammunition within the Navy. A broad discussion of the logistics hierarchy has highlighted the necessity of accurate and timely information flowing up from the hundreds of field activities and ships. The purpose of this thesis was stated to be a software application package to automate the tedious and complex manual record keeping required at the fleet end-user level.

Chapter 2 will describe the present manual system of ammunition management in detail and point out the problems inherent in these procedures. The references for the manual system will be examined to illustrate the difficult task of interpreting this large volume of often overlapping documentation. Examples of data redundancy will be examined to help understand why internal record keeping is more complex than it should be. Finally, the lack of standardized procedures for the end-user management of ammunition will be discussed in light of the improvements available from a standardized software application.

Chapter 3 describes the methodology for the development of a software solution to the problem. The analysis phase is considered in detail and system data elements are identified. A tailored Data Element Dictionary (DED) is developed and Data Flow Diagrams (DFDs) illustrate the proposed system as levels of process descriptions. Finally, the advantages of a database solution and normalization theory are discussed.

Chapter 4 provides the technical details of the relations, fields, indexes and keys with respect to relational database design and normalization. The transition from logical to physical process descriptions takes place with the development of structure charts for the system. The qualities of good modular design are discussed and the effects programming in dBase III Plus has on these qualities.

Chapter 5 discusses the application code that has been implemented for this thesis and the reasons for selecting these programs. Coding design decisions and style are discussed in light of the potential end-users and frequency of use. System expandability and flexibility are highlighted.

Chapter 6 contains recommendations for future research and comments and conclusions.

The Appendices will consist of:

- a. Data Flow Diagrams
- b. Data Element Dictionary
- c. Program Directory

- d. Structure Charts
- e. Program Code Listings
- f. Programs Diskette

II. PRESENT MANUAL AMMUNITION MANAGEMENT / PROBLEMS

Naval units are required to maintain records and submit reports which provide accountability of conventional ammunition and allow reconciliation of Navy ammunition assets world-wide.

As a preface to a discussion of the inventory control procedures, it is worthwhile to review the item control methods used within the Navy. The military services participate in the Federal Cataloging System [Ref. 10: sec. 2032]. Most NATO countries also participate in the United States Item Identification Code, for military standardization purposes, under NATO Standardization Agreement 3151 [Ref. 10: sec. 2035]. All conventional ammunition items at the end-user level should conform to the Federal Cataloging System. These items are assigned a National Stock Number (NSN) by the Defense Logistics Services Center (DLSC) at Battle Creek, Michigan.

An NSN is a 13-digit stock number, see Figure 2.1, and is composed of a 4-digit Federal Supply Classification (FSC) followed by a 9-digit National Item Identification Number (NIIN). An FSC describes a "family" of items of supply sharing such common characteristics as nomenclature, end application, or physical construction. For example, the FSC 1345 is the bomb "family" of conventional ammunition.

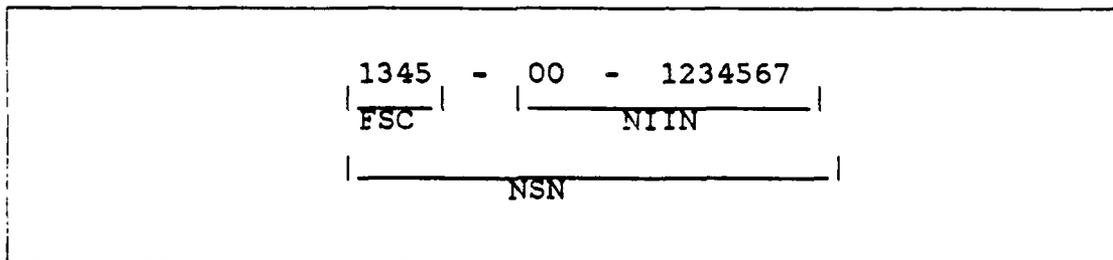


Figure 2.1 National Stock Number breakdown.

The first two digits of the NIIN are the National Codification Bureau (NCB) code, and are essentially a country code (and in some publications referred to as that), within the NATO framework. The United States is assigned the NCB "00" and since it is always part of the NIIN we will no longer distinguish it from the NIIN. A NIIN uniquely identifies an item in the Federal Cataloging System.

Item identification unfortunately gets a bit more complicated. The Department of Defense (DoD) has established a Department of Defense Ammunition Code (DODAC), which is an 8-digit code consisting of a 4-digit FSC (same as previously mentioned), and a 4-digit DoD Identification Code (DODIC). See Figure 2.2 .

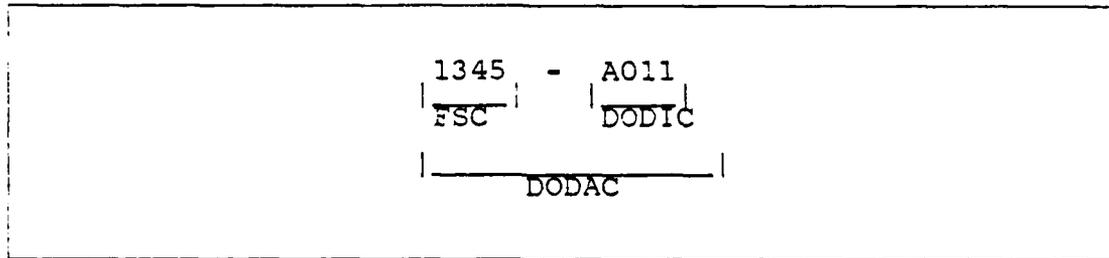


Figure 2.2 Department of Defense Ammunition Code.

To proceed further, the Navy Ship's Parts Control Center (SPCC) has assigned a 4-digit Navy Ammunition Logistics Code (NALC) to certain end round missiles and torpedoes. The NALC is similar to the DODIC except that it is assigned by SPCC rather than DLSC. See [Ref. 11]. NALC's are listed in the Stock List of Navy Ammunition, TWO10-AA-ORD-010 [Ref. 12], along with the DODIC if a NALC has not been assigned.

A particular item will have a unique FSC, a unique NIIN, but may have either a NALC or a DODIC. Figure 2.3 may help illustrate this point.

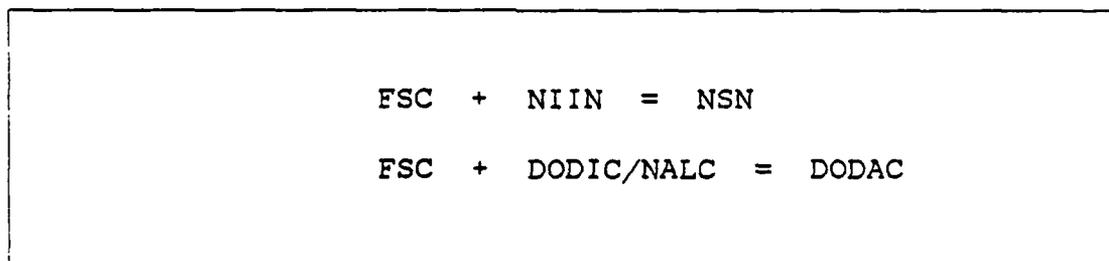


Figure 2.3 Identification Number Composition.

Until recently, Navy ammunition items were tracked by NALC vice NIIN, and it is obvious that since a NALC does not uniquely describe an item but a group of very similar items, accurate stock knowledge was not possible for the end-user accounts.

Items of the same NALC/DODIC are normally functionally compatible, but may have small differences. For example, NALC A015 is for 12 gauge shotgun shells, 00 buck, but one NIIN is for a shell with paper cases and one has plastic cases. Apparently recognizing the lack of accuracy that results from NALC reporting, SPCC is now emphasizing NIIN reporting on all documents. An automated system should enforce this more logical approach, while still maintaining the NALC identity because it is still widely used for referring to an ammunition type. The Stock List of Navy Ammunition, mentioned earlier, contains cross referencing for this purpose, as well as being the prime source of generic information about conventional ammunition within the Navy, Coast Guard, and Marine Corps. It is updated regularly on microfiche and held by all activities with Navy ammunition.

To conclude this discussion of item identification, two additional terms are important. Ammunition Lot Numbers (ALN's) or just Lot Numbers, are assigned by loading, manufacturing, or assembly activities to identify homogeneous material that should function in a "near identical" manner. The lot number is also important to allow tracking, to maintain performance and surveillance records, and allow suspension or recall if necessary. New items are assigned ALN's in accordance with MILSTD-1168A, however much older ammunition is still in the inventory with less uniform lot number formats. Finally, serial numbers are assigned to high value or special interest items to allow individualized tracking.

A. INVENTORY RECORDS

Three types of record cards are used onboard ships for inventory control.

1. Master Stock Record Card

The Ammunition Master Stock Record Card, NAVSUP Form 1296, Figure 2.4, is kept for each NALC/DODIC carried onboard. The card provides a history of the transactions that have occurred effecting the quantity or status of that item. Changes to the quantities in the inventory can take place by receipts, issues, or expenditures. Expenditures can occur due to combat operations, training, test and evaluation, exercises, and other reasons. Each form of transaction has a one character code, the Transaction Code, that describes it. These codes are explained and listed in the SPCC CAIMS manual [Ref. 3: p. 8-5-35].

If material is received or issued it will have a corresponding document number, which is composed of the activity's Unit Identification Code (UIC), the 4-digit julian

date, and a 4-digit serial number. The serial numbers are assigned sequentially and the recommended range is 8000 to 8999. They may then repeat, however used in conjunction with the julian date, a requisition can be uniquely identified. See Figure 2.5

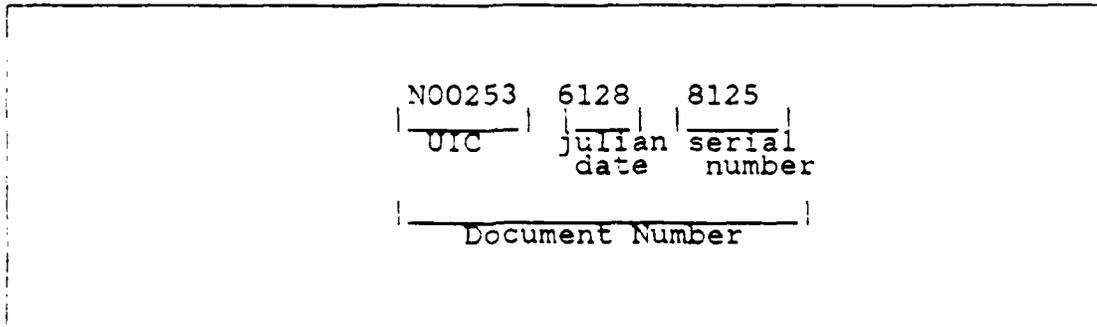


Figure 2.5 Document Number.

Reclassification of onboard inventory items may occur when SPCC, or the Inventory Manager or Technical Manager, determines that a particular lot of ammunition should be suspended, used in a limited fashion, or considered unserviceable. Fleet users are notified of such changes by Notices of Ammunition Reclassification (NAR) messages. Approximately annually, all effective NAR's are incorporated into NAVSEA Publication TWO24-AA-ORD-010, Ammunition - Unserviceable, Suspended, and Limited Use [Ref. 13]. A particular item's degree of serviceability is described by its one character Condition Code, all of which are fully discussed in Appendix C to Reference 13.

On the Master Stock Record Card, the on-hand balances are subdivided among the various condition codes that the activity holds for a particular NALC. Condition Code Alpha, unrestricted, is naturally the most common.

The Naval Sea Systems Command (NAVSEA) publishes allowance lists for each vessel which depend on its type and configuration. The lists contain ammunition type and quantity authorized, as well as the quantity of certain NALCs that may be used for training. The master record card is used to record these numbers and the computed unexpended training allowance remaining for the fiscal year.

Any changes in quantities or condition codes require that an Ammunition Transaction Report (ATR) be submitted. This report will be more fully described later.

but the Master Stock Record Card is used to associate the ATR number with the particular transaction line item on the card.

Finally, various other codes are recorded on the card which are obtained from one of several references:

1. Packaging Remarks: References 12 or 14
2. Logistics Code-NALC: References 12 or 14
3. COG-Cognizance Symbol: Reference 12
4. MCC-Material Control Code: Reference 12
5. DOT-Department of Transportation Hazardous Material Code: Reference 14
6. NIEEW-Net Explosive Weight: References 12 or 14
7. C.G. Haz. Cl.-Coast Guard Hazardous Material Class : Reference 14

2. Lot/Location Card

The Ammunition Lot Location Card, NAVSUP Form 1297, Figure 2.6, is completed for each lot number within a NIIN. There are only a few new concepts on this card that have not been previously explained, so the explanation will be brief. The consignor consignee is the shore activity or operating unit to which the issue was made or from which the item component was received. This card contains much information previously entered on the Master Stock Record Card.

3. Serial/Location Card

The Ammunition Serial Location Card, NAVSUP Form 1356, Figure 2.7, is completed for items that require SLIT tracking. Serial Lot Item Tracking (SLIT) is a system whereby certain ammunition items are designated for increased tracking and surveillance. These items may require identification by lot number, serial number, or both when reporting transactions. This distinction is indicated by the Material Control Code (MCC):

<i>MCC</i>	<i>Reports</i>
B	Lot Number
C	Item Serial Number
E	Lot and Item Serial Number

Items that do not require SLIT reporting will not have an MCC assigned. MCC Bravo ammunition is adequately documented on the Lot Location Card. Items with MCC Charlie and Echo require individualized tracking with the Ammunition Serial Location Card.

The only new item here is the Maintenance Due Date (MDD), which is the month and year of the component's next scheduled maintenance. MDD's are assigned to MCC Charlie and Echo items only. Figure 2.8 may help to illustrate the relationships between the various stock cards.

4. Discussion of Inventory Record

As Figure 2.8 demonstrates, normally 2 and occasionally 3 of the stock record card types are required to describe an ammunition item. With no other device available, the Master Stock Record Card must hold all the transaction information for every transaction (i.e., document number, type, quantity, etc.). It must hold all of the allowance information, or you would have to reference the Allowance List each time you needed that information. It holds much generic data about the item which does not change regardless of transactions or quantity in the inventory. The alternative however would be to look up information, each time you were interested, in at least four very voluminous publications.

The Lot Location card subdivides each NIIN by lots, and much information is repeated. The only new data items are the lot numbers. The Serial Location cards likewise duplicate data.

It is quite obvious that maintaining all the required records for even a small ship's inventory, say at least 40 NALCs, would be extremely tedious. A large vessel with perhaps hundreds would demand full time attention. It is this author's experience that much of the repetitive information would not get entered properly, and the initial construction of a set of cards would be an onerous task. Multiplying those man-hours for all the ships involved results in considerable non-operational training time.

Ideally the inventory records should contain information that deals only with that particular batch of ammunition, namely:

1. NIIN - what is it?
2. Condition Code - what can it be used for?
3. Activity Classification Code - who is it for?
4. Quantity - how many are there?
5. Storage Location - where is it?
6. MDD - when does it need maintenance?
7. Lot Number - what lot is it?
8. Serial Number - which one is it?

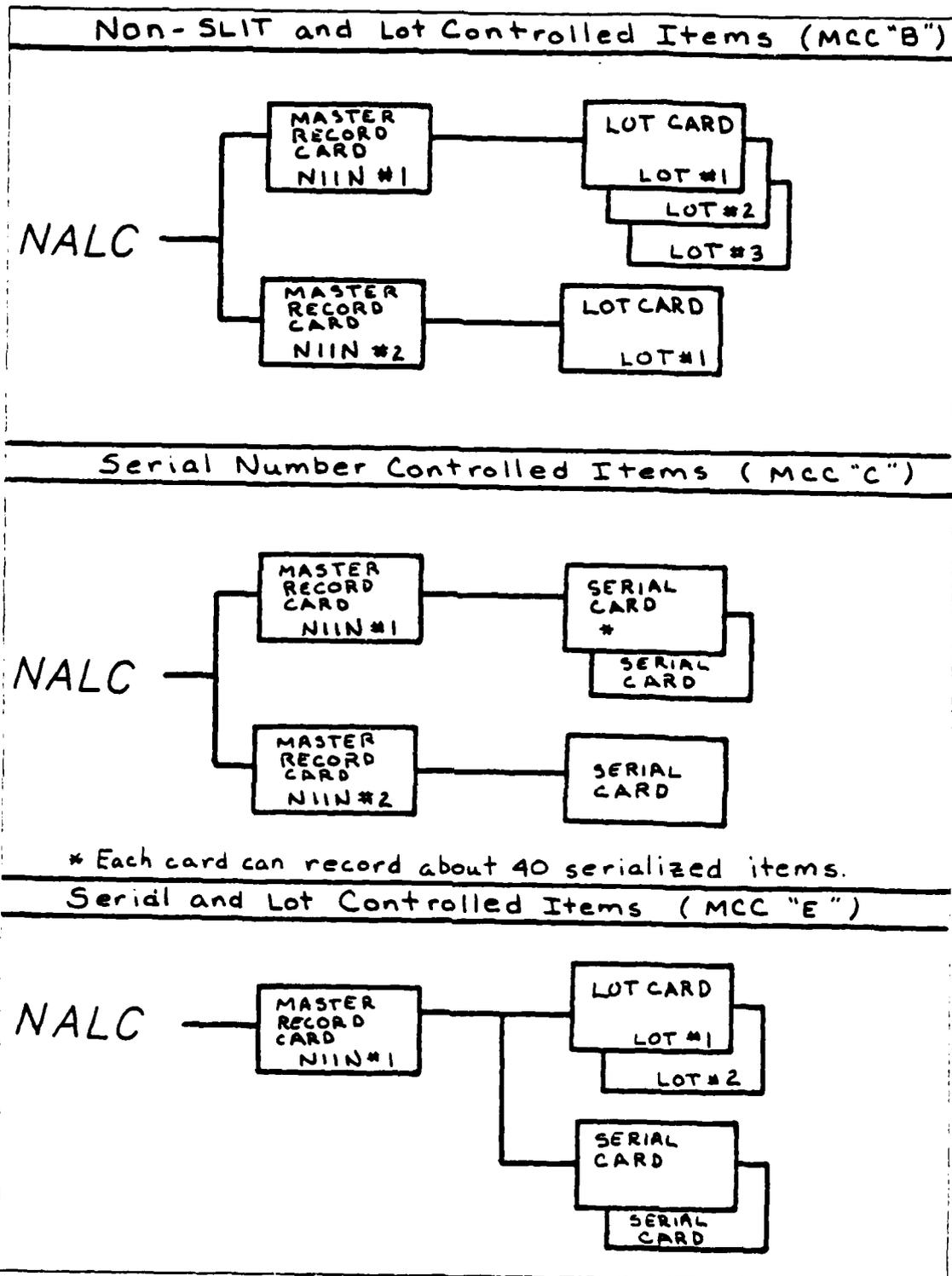


Figure 2.8 Ammunition Stock Card Relationships.

This information can *not* be looked up in a publication because it deals with the particular articles at hand. Other information listed on the stock cards is of general nature and can be looked up, indicating that we could locate it in one central place and reference it when needed. Therefore we would not have to repeat it on every card. This is one of the important concepts of automation that can be used in many ways as chapter 3 will demonstrate.

B. REQUISITION, TURN-IN/TRANSFER, RECEIPT

1. Background

Naval vessels are required to maintain 100% of their authorized allowance on board or on order, so as to be ready for immediate combat operations. Certain exceptions apply which are spelled out in the SPCC CAIMS manual [Ref. 3: p. 8-2-2], and may be modified by fleet commander instructions [Ref. 4,5]. Reasonableness dictates how small an order should be placed to satisfy these requirements, of course, where ammunition is concerned, too much is better than not enough!

A few more definitions are appropriate at this point. Activity Classification Codes (ACC) describe basically who the ammunition is for. It may be carried by a ship to satisfy its own offensive and defensive armament needs, in which case it is ACC Alpha ammunition. However, other ammunition could be carried for embarked Marines, aviation units, or for underway replenishment of other ships. Each of these other recipients causes the ACC to be different. Most medium to small ships and submarines only carry ammunition for their own use, but most large ships have ammunition on board for other purposes and thus they must account for it separately (ie- a separate deck of inventory cards). This makes the ACC an essential data element for each ammunition record. Chapter 8 of the CAIMS manual [Ref. 3: p. 8-5-34], lists and describes all of the ACC codes.

Associated with the concept of ACC, are the three types of allowance lists a ship may have. A Shipfill Allowance List authorizes the quantities and types of ammunition for own ship's use. A Mission Load Allowance List is that ammunition carried to support associated ships or aircraft squadrons; usually aircraft carriers and tender type ships. A Cargo Load Allowance List is normally held by designated cargo or logistic type ships (AE, AOE, AOR, etc.) for replenishment of other vessels.

All naval vessels, as well as the other services and many government agencies, submit requisitions and transaction reports in Military Standard Requisitioning and Issue Procedures (MILSTRIP) format [Ref. 3: Chapter 8]. This system serves to

standardize procedures for efficiency and economy. It allows machine readable logistics traffic by specifying data elements, codes, and formats. Until recently, transaction reporting was done in normal narrative message format which had to be transcribed to MILSTRIP format at a shore activity with ADP capability. New instructions from SPCC [Ref. 3], and those still in production at fleet headquarters, now direct a format that is machine readable, reducing the chance of error in transcription or key punching. SPCC is linked, via ADP equipment and telecommunications with all major stock points, many minor stock points, and other logistics organizations. Ships submit manual requisitions or send messages which are entered into the CAIMS at a shore activity.

The Defense Automatic Addressing System (DAAS), with primary location in Dayton, Ohio, is a telecommunications system which was designed to effectively route logistics traffic from all the services. DAAS computers can receive messages, perform some format error checking, determine the addresses, and route them over the quickest path to their destination. DAAS functions over the Automatic Digital Network (AUTODIN). The message format that is acceptable to DAAS is slightly different, and the requirements for use more restrictive, than a normal naval message. However when applicable, it is the most efficient way to route logistics traffic. The SAMS system should allow for transmission in either format as well as manual requisitioning.

2. Requisitioning

Ships and other naval units may submit requisitions for conventional ammunition in one of three formats as previously alluded to. A manual requisition, DD Form 1348, Figure 2.9 , may be completed and physically delivered or mailed to a shore weapons facility. The shore activity enters the requisition into the CAIMS with their ADP equipment, SPCC routes the requisition to the appropriate Inventory Manager (SPCC,NAVAIR,NAVSEA,NMWEA,JCMPO), and an appropriate stock point is selected to deliver the material.

A message requisition may be prepared in DAAS format, Figure 2.10 , and electrically transmitted. A requisition line item on a DAAS message is printed on one horizontal line or 66 characters, with no separations. A DAAS message may also contain followup actions, modifications, cancellations and other logistics actions besides requisitions. The type of action is determined by the document identifier, column 1-3. The Routing Identifier (R I), column 4-6, is analyzed and the line item sent to its addressee. In this format the requisition is fully machine readable.

EDITION OF 1 AUG 61 MAY BE USED UNTIL EXHAUSTED

DOC NO		STOCK NUMBER		QUANTITY		EDITING DATA		DOC NO		QUANTITY		ROUTING NO		STOCK NUMBER		QUANTITY		UNIT OF ISSUE	
REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK	REQ	STOCK
1	101	1	101	1	101	1	101	1	101	1	101	1	101	1	101	1	101	1	101
2	102	2	102	2	102	2	102	2	102	2	102	2	102	2	102	2	102	2	102
3	103	3	103	3	103	3	103	3	103	3	103	3	103	3	103	3	103	3	103
4	104	4	104	4	104	4	104	4	104	4	104	4	104	4	104	4	104	4	104
5	105	5	105	5	105	5	105	5	105	5	105	5	105	5	105	5	105	5	105
6	106	6	106	6	106	6	106	6	106	6	106	6	106	6	106	6	106	6	106
7	107	7	107	7	107	7	107	7	107	7	107	7	107	7	107	7	107	7	107
8	108	8	108	8	108	8	108	8	108	8	108	8	108	8	108	8	108	8	108
9	109	9	109	9	109	9	109	9	109	9	109	9	109	9	109	9	109	9	109
10	110	10	110	10	110	10	110	10	110	10	110	10	110	10	110	10	110	10	110
11	111	11	111	11	111	11	111	11	111	11	111	11	111	11	111	11	111	11	111
12	112	12	112	12	112	12	112	12	112	12	112	12	112	12	112	12	112	12	112
13	113	13	113	13	113	13	113	13	113	13	113	13	113	13	113	13	113	13	113
14	114	14	114	14	114	14	114	14	114	14	114	14	114	14	114	14	114	14	114
15	115	15	115	15	115	15	115	15	115	15	115	15	115	15	115	15	115	15	115
16	116	16	116	16	116	16	116	16	116	16	116	16	116	16	116	16	116	16	116
17	117	17	117	17	117	17	117	17	117	17	117	17	117	17	117	17	117	17	117
18	118	18	118	18	118	18	118	18	118	18	118	18	118	18	118	18	118	18	118
19	119	19	119	19	119	19	119	19	119	19	119	19	119	19	119	19	119	19	119
20	120	20	120	20	120	20	120	20	120	20	120	20	120	20	120	20	120	20	120
21	121	21	121	21	121	21	121	21	121	21	121	21	121	21	121	21	121	21	121
22	122	22	122	22	122	22	122	22	122	22	122	22	122	22	122	22	122	22	122
23	123	23	123	23	123	23	123	23	123	23	123	23	123	23	123	23	123	23	123
24	124	24	124	24	124	24	124	24	124	24	124	24	124	24	124	24	124	24	124
25	125	25	125	25	125	25	125	25	125	25	125	25	125	25	125	25	125	25	125
26	126	26	126	26	126	26	126	26	126	26	126	26	126	26	126	26	126	26	126
27	127	27	127	27	127	27	127	27	127	27	127	27	127	27	127	27	127	27	127
28	128	28	128	28	128	28	128	28	128	28	128	28	128	28	128	28	128	28	128
29	129	29	129	29	129	29	129	29	129	29	129	29	129	29	129	29	129	29	129
30	130	30	130	30	130	30	130	30	130	30	130	30	130	30	130	30	130	30	130

REQUISITION IS FROM:

REQUISITION SYSTEM DOCUMENT (MANUAL)

DD FORM 1388 (6-67) 1 MAR 74

D O SINGLE 1388 1388

Figure 2.9 Manual Requisition Document.

		INDEX			
			TT	DMON	
<p>FROM: USS YOUR SHIP</p> <p>TO: DAAS DAYTON OH</p> <p>INFO: CINCLANTFLT NORFOLK VA</p> <p>WPNSTA CHARLESTON SC</p> <p>SUBJ: AMMC MILSTRIP REQ</p> <p>ADDN: BW1320009333321 EA00575V2345611008009HN123456Y6R2T3761321900</p>					

Figure 2.10 DAAS Message Requisition Format.

DAAS has restrictions however. There can be no narrative remarks, the classification must be UNCLASSIFIED, and the R/I must be a continental United States (CONUS) activity connected to AUTODIN. Usage of DAAS by afloat units is somewhat limited by fleet commander instructions which require narrative remarks and primary addressee other than DAAS, Dayton, OH on some ammunition items.

The last requisition format is the normal narrative naval message, Figure 2.11. It should be noted that information and codes present on any of the three formats are almost identical. Fleet commander instructions require some minor format changes. The Pacific Fleet Conventional Ordnance Management Manual [Ref. 4: p. 1-1-A-5] for example, requires that the quantity and NSN be spelled out on naval message requisitions to prevent confusion in the event of a garbled message. In any case, this format is not machine readable and must be entered into CAIMS at a shore activity. It is more flexible however than a DAAS formatted message. Remarks may be included, it may be addressed to any activity, and it may contain classified information (the remarks generally are the only classified information).

The three formats are reasonably well described [Ref. 3: Chapter 8]. The codes are rather cryptic and their full names are shown below. Explanations may be found in Appendix B, the Data Dictionary.

TABLE 1
MISCELLANEOUS REQUISITION CODES

<i>Code</i>	<i>Full Name</i>
D/I (Doc. Ident.)	Document Identifier
R/I	Routing Identifier
M&S	Media and Status Code
Serv	Service Code
Dem	Demand Code
Sig	Signal Code
Fund	Fund Code
Dist	Distribution Code
Proj	Project Code
Pri	Priority Code
RDD	Required Delivery Date
Adv	Advice Code

3. Turn-in/Transfer

Ships often have to turn-in ammunition that is excess, reclassified by NARs, or requiring periodic maintenance ashore. The DD Form 1348-1 is used for this purpose. Figure 2.12 . Somewhat fewer coded items are necessary on this form because the ship is physically delivering the material to another activity. Most commands also select another group of serial numbers for turn-ins, in order to differentiate them from requisitions. A separate log is maintained. The CAIMS manual [Ref. 3: p. 8-3-1], assists the user in completing the form.

4. Receipt

Ammunition is also received on DD Form 1348-1. The document number on the receipt document corresponds to the receiving ship's requisition document number, of course the possibility of being sent material that was not ordered exists. The only new data items that have not been previously discussed are the unit price and the unit of issue, both of which are listed in the Stock List of Navy Ammunition [Ref. 12].

5. Discussion

The requisition, turn-in/transfer, and receipt documents require some form of logs to be kept. A Requisition Log may contain a history of requisitions and receipts by document number (serial number). A Turn-in log would be similar. Retention of these documents provides a source of information for the future, instead of having to look up the information all over again. This is not necessarily a good practice as certain data elements may have changed in the interim. But this is just the kind of thing a busy sailor might do to save time, not recognizing the potential problems involved. Recall of the many different data items from publications can be time-consuming.

C. AMMUNITION TRANSACTION REPORTING

Transaction reporting is required for any action, event, or procedure that results in the receipt, issue, transfer, expenditure, loss, gain, reconfiguration or change in material condition of reportable material. [Ref. 3: p. 8-4-3].

The format of Ammunition Transaction Reports (ATR's) has recently been changed to allow optical scanning of the data elements which are separated by from one to four slashes, "/". ATR's have multiple formats depending on the type of transaction, the MCC of the material (i.e., SLIT or non-SLIT), and the actual type of the material. This extreme variability has traditionally caused the greatest problem, resulting in a

high error rate for these documents. Although the new format is more efficient in that it can be optically read and entered into the CAIMS by shore activities, it is now incomprehensible to the fleet user. Therefore each ATR must be created in a painstaking step-by-step manner, minding the format and content of each data element. ATR's are transmitted as naval messages to SPCC, with information addressees and classification depending on the type of ordnance and the type of sending platform (i.e., submarine, surface vessel). Figure 2.13 illustrates one format of ATR. The serial numbers run from 001 to 999 and then repeat, with a ship's most recent ATR indicated in the current ATR's reference line. This is to ensure that SPCC receives all ATR's from a ship without omission, and in the correct sequence. Again, multiple transactions can be listed on one message as long as the common information in the header applies to all the transaction line items. Figure 2.14 attempts to illustrate the various ATR formats, the variability is evident. Each vessel keeps an ATR log to form the primary history of the ship's ammunition transactions, from overhaul to overhaul, and to allow correction of any ATRs that were submitted with incorrect data. This log may consist of all the actual messages and a summary of each ATR with its effect on the running total of each item involved.

Properly submitted ATRs are critical for the overall functioning of CAIMS. The old adage "garbage-in, garbage-out" aptly applies, and it is sincerely hoped, by all levels, that higher echelons are not making procurement and allocation decisions based on incorrect data.

D. COMMENTS

In addition to comments made throughout this chapter concerning data redundancy, multitudes of codes, and more than a few reference publications, mention should be made of the lack of any standardized record keeping procedures. Requisitions, turn-in documents, and ATRs must be submitted, and inventory cards maintained, but no system or procedures exist for the maintenance of logs or records.

It is generally accepted that some sort of auditable system must exist to resolve discrepancies and maintain accountability. The procedures that each individual ship creates may range from excellent to poor. Some ships may have even written ship's instructions for this purpose, detailing the manual methods to be used. A software package, such as the proposed SAMS, will have several benefits besides more accurate and timely ATRs. It will keep all logs and records for the user, it will standardize procedures without requiring the user to create new logs and binders, and finally it will

JOINT MESSAGE CENTER		Unclassified Example	
			NBAT
<p>FROM: USS SARATOGA</p> <p>TO: SPOC MECHANICSBURG PA</p> <p>INFO: AS REQUIRED BY SPOC AND FLEET INSTRUCTIONS</p> <p>IDENTIFICATION: N0801517</p> <p>SUBJ: AMMO TRANS RPT SPOC 8010-12</p> <p>///033607084/D1660974577</p> <p>///A659009356171 A B10000 F5000 T5000///</p> <p>///A6900054204057A/B20000 F1000 T18800///</p>			
Unclassified Example			

Figure 2.13 Ammunition Transaction Report (example).

all fit on a disk or diskettes. It might even save a few nervous breakdowns when new Weapons Officers report onboard to find no system in existence at all.

III. AN AUTOMATED SYSTEM DEVELOPEMENT

The purpose of Chapter II was to give the reader a basic knowledge of the records and reports presently used onboard non-automated ships of the Navy for conventional ammunition inventory control and accountability. Problems of data duplication, lack of standardized record keeping procedures, data inaccessibility, and late and inaccurate external reports as a result of these were discussed. This chapter shall develop a database management system (DBMS) solution to alleviate those problems as well as reduce the administrative burden on ship's force personnel.

A. METHODOLOGY

It is now generally accepted that systems analysis and design must follow an orderly, logical process to arrive at a system that is reliable, maintainable, efficient, and manageable (i.e. within cost and time). The Shipboard Ammunition Management System (SAMS) development has followed such a process with certain modifications that reflect the new technologies of the DBMS and its easy to use programming language.

Structured analysis and design evolved in the 1970's to bring a disciplined approach to computer system development. The 1950's and 1960's were characterized by haphazard analysis and design techniques and what Meilir Page-Jones referred to as the "Instant Karma" approach of computer system development [Ref. 15: p. 27]. Many excellent books now exist on the subject of structured techniques, [Refs. 15,16,17], so the treatment here will be brief and only as it relates to the SAMS development. Davis [Ref. 17: p. 8] defines the steps involved in a structured development, or life cycle as:

1. Problem Definition
2. Feasibility Study
3. Analysis
4. System Design
5. Detailed Design
6. Implementation
7. Maintenance

The relative weight that each step carries and the time spent in each area, of course depend on the project at hand. Most organizations use, or have used, some derivative of this basic life cycle in their system development process. A reasonable approach is to study the available structured development philosophies, modify these where advisable to accommodate new technologies and tools, and apply the best composite plan. There have been worries among structured technique advocates that the new technologies, i.e. fourth generation languages, prototyping packages, and personal computers, will compromise advances made in systems analysis and design. But, as Edward Yourdon, a prime advocate of the structured techniques, points out:

The major philosophical concepts used to build reliable, maintainable information systems, which is what the structured techniques are all about, can continue to embrace new technologies without destroying the concepts themselves. [Ref. 16: p. 6]

Chapters I and II went into some detail on the nature of the problem and defined the scope of the solution. A formal feasibility study was not conducted, however the major points of such a study were considered [Ref. 17: p. 274]:

1. Technical: Can the system be implemented with current technology?
2. Economic: Do the benefits outweigh the costs?
3. Operational or Organizational (*Political*): Can the system be implemented in this organization?

Technical feasibility is well assured. Inventory control systems are not new, the data and reports required are well defined, and the DBMS products make the task less difficult.

Economic feasibility, as usual, is hard to quantify. How much is a 10 per cent increase in system accuracy worth to Navy planners responsible for ammunition procurement and allocation? Also, how much is the reduction in administrative burden, and the subsequent increase in operational readiness, worth? These are difficult questions to answer. In light of the relatively minor expense of the SAMS development however, it only makes sense to pursue an alternative to the present manual system. If the project progresses beyond the research stage to fleet implementation, the increased costs would warrant a more formalized cost benefit study.

Political feasibility should pose no problems. Weapons and operations personnel on most Navy ships are already heavy users of computers in their daily routines. Any labor saving device, which reduces records as well, would most likely be welcomed.

B. ANALYSIS

The analysis phase of a project must fully determine what the system must do. How this is physically accomplished is the subject of the next section on design. For this author, the analysis phase began in 1983, unknowingly, with experience operating the manual system, which is largely unchanged today. However, under the pressure of multiple jobs, most users generally never acquire the big picture that is necessary to design an automated system. Normally the analyst will interview many people involved in different phases of the operation, plus review all the applicable documentation, to gain this big picture. This project required a slightly different approach. The author acted as an expert user analyst and based on experience and intense study, the system was designed and partially implemented. The pitfall here is that the expert user is too "expert" to really appreciate user difficulties. These difficulties would have to be resolved by a period of "normal" user interaction in the fleet following complete implementation. It will be shown that a DBMS makes such modifications, as may be necessary to suit the bulk of the users, much easier to perform.

A complete knowledge of the existing system was the first major job in the SAMS development. Information, publications, instructions, reports, and forms were collected from all the commands and agencies involved in the flow of conventional ammunition to the fleet. The intent, policies, and procedures of the CAIMS program were studied to gain this big picture, even though many details did not directly affect the end user environment. There are several "layers" of instructions that deal with similar, or the same topics, and the semantic inconsistencies between these often necessitated phone call and extensive cross referencing to resolve. This overlapping of instructions is seen by this author as a definite problem. It can thwart a clear end user understanding of the system, since they have little time for phone calls or extensive research.

The object of SAMS is to automate the end-user inventory and control procedures while remaining compatible with requirements of higher level Navy instructions. External report formats must be as described in Chapter II, however internal stock cards would not be used in an automated system naturally. This would require modifications in those instructions deleting the requirements for those cards for the ships that are automated.

1. Data Flow Diagrams

One of the major tools of structured analysis is the use of graphical representations to depict information flow. Data Flow Diagrams (DFDs) are a network representation of a system which is generally much easier to understand, at least by the end-user, than the functional specifications of the past. The DFD is an analysis tool, and as such, is logical in nature and does not depend on hardware, software, data structure, or file organization. A logical representation is one that reflects a user's view of the system and its processes. Several references [Refs. 15,17: p. 60,281], give similar descriptions of the constituent parts and development of DFDs. Figure 3.1 shows the Fundamental System Model or the Context Diagram for the SAMS. This is the highest level of abstraction and shows the entire system as a single node. The user inputs data to the system or requests processing at the source square, the system node performs the necessary processing, and outputs data and/or reports at the sink square. This context diagram serves as a starting point but is not particularly useful or informative. Subsequent levels of refinement form a complete set of Data Flow Diagrams for the SAMS and are included as Appendix A.

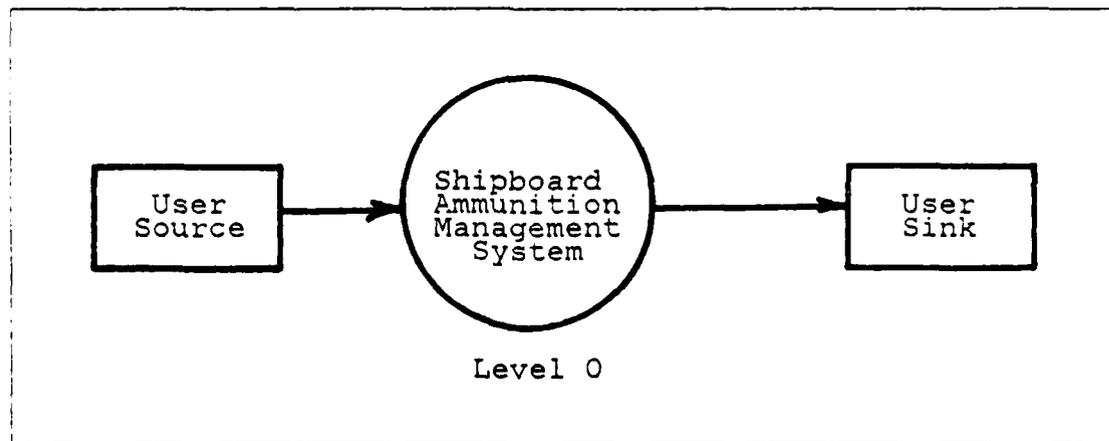


Figure 3.1 SAMS Context Diagram.

Data Flow Diagrams do not show flow of control or sequence of execution of the processes. Also, the representation of files are purely logical and may be implemented quite differently in the final design.

2. Subsystem Functional Description

Experience and research provided the logical divisions for the conventional ammunition management problem. Nine subfunctional areas were identified.

1. User Access Validation
2. General Information Documentation
3. Inventory Allowance Ammunition Data
4. Transaction Management
5. Requisition Management
6. Turn-in Document Management
7. NARS Management
8. System Management
9. Generate Internal Reports

This system breakdown seems obvious on the surface to anyone familiar with the manual system, as well it should be. The more closely an automated system follows the logical, or user's view, of the processes, the higher probability there is of the user feeling comfortable with the system. However, the system breakdown also considers the system goals and improvements desired in the automated system over the manual one.

a. User Access/Validation

The User Access/Validation function should ensure that only authorized users of the system may gain access. The SAMS is intended for use by Weapons Department personnel charged with the responsibility for ammunition, and as such, they should be the only people manipulating the system. Requests for system data from superiors can quickly be accessed by these people. The system should be capable of controlling authorized user's privileges within the system also. The SAMS only requires two basic levels of access. The lower level should allow access to the day-to-day processing options, two through seven and nine. The higher level will allow the SAMS administrator or manager to access all modules for system initialization, infrequent data changes, user changes, or to correct unforeseen system processing problems. This degree of power places great responsibility on the administrator and requires that he be fully knowledgeable of the system and the consequences of his actions. The risk in this approach is necessary however, because there will be no technical representative or outside help if the system develops a problem while at sea.

b. General Information/Documentation

The second option provides general information and documentation that the user may find useful in getting acquainted with the system. This may include a system description, explanation of the options available, what the system does not do, hardware requirements, basic keyboard operations, etc.. Also, this option should give the codes and definitions that are frequently used by the SAMS and within the CAIMS system as a whole. A data dictionary will not only explain the many acronyms but will increase the knowledge of the user and assist the manager in system management.

Finally, a help program can be resident within this option and called from various places throughout the application programs. It is very difficult however, to give adequate help to a user with generic help messages about a particular question he may have. The author feels that giving more, rather than less information in the user interface is generally less frustrating than needing to frequently use a help program. Now, if the program was in constant use by a dedicated operator the extra verbage would become annoying, however that should not be the case here. The data entry to the SAMS would probably slow down an operator who has all of the many codes memorized, but for those less gifted (the majority), the system will minimize the time referencing manuals. Also, the instructive goal of the system can be more easily accomplished in this manner.

c. Inventory/Allowance/Ammunition Data

This is basically a review option. The user would frequently want to review the current status of his onboard inventory, perhaps with a quick overall perspective in mind or a detailed review of a particular item. The SAMS inventory information should make the present record cards unnecessary. Since the change in an inventory level can only occur as the result of a transaction of one kind or another, the inventory is not manipulated directly, but is updated as a result of transaction processing.

Allowance information can be conveniently stored in the system where training expenditures can be updated and monitored. Again, any update occurs as a result of transaction processing.

Ammunition data is that generic information about a particular item that does not change as a result of transactions and previously would have been looked up in one of several publications. See references 12 and 14 for examples. Any amount of information, about any kind of ammunition, could be loaded into the system, however to save disk space, only those items the particular ship would likely carry should be entered. The system manager can add or delete items as necessary.

d. Transactions

Transaction Management is the major processing function of the system. A unit can receive items through a receipt, condition code change, or a few other infrequent occurrences. It can reduce stock by an issue, expenditure, condition code change, or other reason. This subsystem should recognize the type of transaction, update inventories, create the information that will go on the ATR, and update other files when necessary. For example, a receipt transaction should update the requisition file if the receipt occurred as a result of a requisition the ship submitted. An expenditure transaction should update the Allowance File if that item had a training allowance.

e. Requisitions

Requisition Management involves the creation of requisition documents in the various allowable formats. The MILSTRIP system, discussed earlier, is heavily code oriented, and this option should guide the user through the selection of the proper codes for the particular item at hand. This module should also store the information as a permanent record, eliminating the need for manual records. A user may edit these records during creation and prior to their being submitted, however the integrity of the data would be violated if he were allowed to do so afterward. Therefore, as with ATR's and Turn-in documents, there is a point where the data must be committed, and not changeable by the user. Any special cases could only be adjusted by the system manager.

f. Turn-in Documents

The Turn-in Document Management function must create and store the turn-in record and print the DD Form 1348-1, the Single Line Item Release Receipt Document. The processes involved are very much like the requisition management function: one representing imminent issue and one representing imminent receipt. Later, when the transaction actually occurs, an issue transaction can be processed which updates the inventory. Naturally, the information already recorded by the Turn-in processing need not be collected again, rather the Turn-in file is linked to the Transaction file and the information transferred.

g. NAR Messages

Naval Ammunition Reclassification (NAR) Management serves to process these messages as they are received. A stock check must be conducted to determine if the ship holds any of the ammunition lots specified in the message. If the check is negative, the only action required is to log the serial number of the NAR message. If

the stock check shows that action is required, then the proper reclassification transaction must be processed. The information pertinent to the NAR must also be saved for future reference and historical records. There should also be a link between a reclassification transaction and the NAR message that precipitated it.

h. System Management

The System Management module will be extremely powerful, allowing the manager to globally edit all files and initialize the system. He will also be able to manage the security system, archive old records, recover the system when necessary from backup files, and have access to the operating system and DBMS dot prompt. System documentation, of no real value to the normal user, will be available to the manager, to further his understanding of the system.

i. Internal Reports

Lastly, the Generate Internal Reports function will produce those reports that the typical users, and their supervisors, would find useful in management of their conventional ammunition program. Of course, the information would also be available on-line, but it is often useful to have hard-copies. Microcomputers will still not fit in one's back pocket! The SAMS should be flexible enough to allow future report requirements to be incorporated without system redesign. The data flow diagram for this function describes several of the anticipated reports, however expansion is probable as new requirements come to light. User experience would suggest other useful reports.

3. System Data

As the name would suggest, a data flow diagram shows the data that interfaces the active processing areas of the system. The data is stored in files for later recall or processing. A convenient way to aid the analysis of a system is to identify the data that is require at the "sink" and work backwards through the system to the source of the data. The SAMS has certain data elements that it must be able to produce to construct ATRs, requisitions, and turn-in documents. Also, it must record the inventory data to replace the manual stock cards. This data then forms the minimum set for the system. Table 2 identifies the data that is included in these output documents and the stock cards.

These data elements thus form the initial inputs to a Data Element Dictionary (DED). A DED is basically a collection of data about data. The idea is to provide information on the definition, structure, and use of each data element in the system

TABLE 2
SAMS OUTPUT DOCUMENT DATA ELEMENTS

Transaction Report Data Elements

From:	Document Number:
Name	Serv. Desig. Code
Info:	UIC
Info Addresses	Julian Date
Date of last ATR/ or last ATR DTG	Serial
Own ship UIC	Ending Balance
ATR Serial	Lot Number
Activity Class. Code	Component Serial Number(s)
Julian Date	Maintenance Due Date
NIIN	Type of Maintenance Due Code
Beginning Balance	Old Condition Code
Quantity	New Condition Code
User ID/Source ID Code	Transaction Code

Turn-in Document Data Elements

Stock Number:	Ship to:
FSC	Serv. Desig. Code
NIIN	UIC
Unit-of-issue	Name
Quantity	Location/Hull Number
Document Number:	Total Price
Serv. Desig. Code	Security Risk Code
UIC	Material Condition Code
Julian Date	DOT Class. Code
Serial	C.G. Hazard Code
Distribution Code:	Lot Number
Monitoring Activity	Component Serial Number(s)
Cognizance Symbol	NALC
Project Code	Noun Name
Unit Price	NAR Number:
Shipped from:	NAR Serial
Serv. Desig. Code	NAR Year
UIC	Date Shipped or turned in
Name	Freight Class. Nomenclature
Hull Number	Type of Container
	No. of Containers

TABLE 2
SAMS OUTPUT DOCUMENT DATA ELEMENTS (CONT'D.)

Requisition Data Elements

Requisitionee: Serv. Desig. Code UIC Name Location/ Hull Number Requisitioner: Serv. Desig. Code UIC Name Hull Number Document Identifier Routing Identifier Media and Status Code Stock Number: FSC NIIN Unit-of-issue Quantity	Document Number: Serv. Desig. Code UIC Julian Date Serial Demand Code Supplemental Address: Serv. Desig. Code UIC Signal Code Fund Code Distribution Code: Monitoring Activity Cognizance Symbol Project Code Priority Code Required Delivery Date Advice Code Info. Addresses
--	--

Inventory Record Data Elements

Transaction Jul. Date Document Number: Serv. Desig. Code UIC Julian Date Serial Transaction Code Quantity Condition Code ATR Serial Unexp. Trng. Allowance Allowance 90% of Shipfill Allow. Annual Trng. Allowance	NALC Cognizance Symbol NIIN Material Control Code Activity Class. Code DOT Class. Code Net Explosive Weight Stowage Location C. G. Hazard Code Consignor/Consignee UIC Lot Number Component Serial No.(s) Maintenance Due Date
---	--

[Ref. 17: p. 296]. Table 3 lists the format of the information that will be stored in the DED for each data element. By exactly defining the meaning and structure of the data elements, standardization among the application programs is facilitated. Also, as will be discussed, relationships between data can be utilized to greatly increase the amount of useful information the system can produce. This can only be accomplished if consistent data definitions are observed. The DED for the SAMS will be an on-line

facility for user reference and may also be printed for hard copy documentation. Appendix B is the SAMS Data Element Dictionary.

C. DATA ORGANIZATION

1. File Processing

A final consideration, and one of primary importance, is the manner in which the system's data is stored and represented. Traditionally, file processing systems organize all of the data for a particular specialized application into a single file and the program operates on that file to produce the output. In this manner, the file is really just a storage medium for potentially very dissimilar data. No data organization is implied other than sequencing. Other specialized application programs would operate on files that contain all of the data necessary for that application. In terms of the SAMS, such a system might be structured as in Figure 3.2 .

There are some immediate problems with this method of file and data organization for an ammunition management system. First, since each file contains all of the data needed for it's program, there is much redundant data in the system. This of course requires much redundant data entry which the operator resents because he logically questions the necessity for doing it. For example, each of the files in Figure 3.2 , and their associated program, require the name of the ship producing the documents, and thus the files must contain that data. This type of redundancy has traditionally been accepted because the programs that would be required to share data among files were complex. Unless all the application programs and files are constructed with compatibility in mind, a conversion process between different record formats might be necessary. A one-of-a-kind request necessitating shared data would have to be vitally important to merit such effort.

This lack of integration capability limits the information that can be obtained from the data. However, the redundancy of data also presents problems other than the multiple data entry required. It wastes storage space in the files, which is important when considering a microcomputer application. The larger files also cause slower processing times for the system as a whole. Finally, when data elements change, the different programs must all be updated or inconsistencies will result.

2. Database Processing

Database processing is quickly gaining ground on file processing in most ADP applications and has several major advantages over file processing techniques. First and foremost, Data Base Management Systems (DBMS) allow the sharing of data

TABLE 3
DATA ELEMENT DICTIONARY FORMAT

- Longtitle: The long title of the data element.
It may contain up to 30 characters.
- Name: A short hand version of the long title
to be used in programming and is compat-
ible with the DBMS variable representation
and length for record and field names.
(10 characters for field names and 8
characters for record names.)
- DEN: (Data Element Number) For those data
elements that are cataloged in the
Standard Data Element Dictionary(SDED),
a DEN is assigned. The DEN consists of
an alphanumeric character followed by
three or four numerics. The DEN acts as
a means for controlling data elements and
as a short hand name.
- Picture: The data type of the element, i.e.
character, numeric, logical, etc., and
the width of the field.
- Desc.: The narrative description explains
what data or information the element
represents.
- Refs.: Publications and other documentation
that give additional information about
the data element.
- Codes: If the data element has codes associated
with it, the reference that lists all
those codes and their meanings is given.
(Most codes are given in the General
Information option 1.)
- Used in: All relations in which the data element
appears. (SAMS Relations)

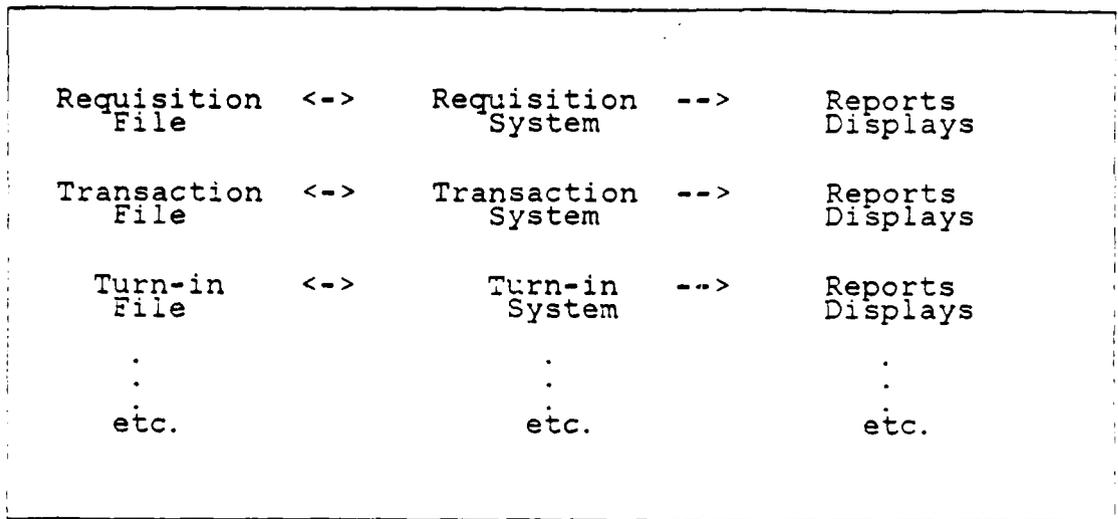


Figure 3.2 File Processing System for SAMS.

between files quickly and easily. This is accomplished by the DBMS program itself, which is usually quite complex, but fortunately requires the user to know nothing about its intricacy. Thus the system files are established under DBMS control in a compatible format. Application programs only communicate with the DBMS, as far as data retrieval is concerned, and it "fetches" the desired data by its own means. Sharing data means that a reduction in redundant data is now possible through intelligent file design. In contrast to Figure 3.2, Figure 3.3 is a representation of how the SAMS might be constructed using database technology.

A DBMS also creates program/data independence since the programs do not need to know anything about the data structure of the files. We can add application programs that use any of the files and this gives much flexibility in system design. Fields can even be added or deleted from the files without any effect on the programs that do not explicitly use the affected data.

There are some disadvantages to database systems, however for medium-to-small applications, with much interaction, they are very insignificant [Ref. 18: p. 6].

The SAMS will therefore use a modern commercial database package, dBase III Plus, by Ashton-Tate of Torrence, California. This is a moderately priced package for use on microcomputers and has gained much popularity. DBase III Plus is a relational database which simply means that data is represented in the form of tables, or relations. Readers interested in a full discussion of relational database theory should refer to Date [Ref. 19], Ullman [Ref. 20], Kroenke [Ref. 18], and Brodie [Ref. 21].

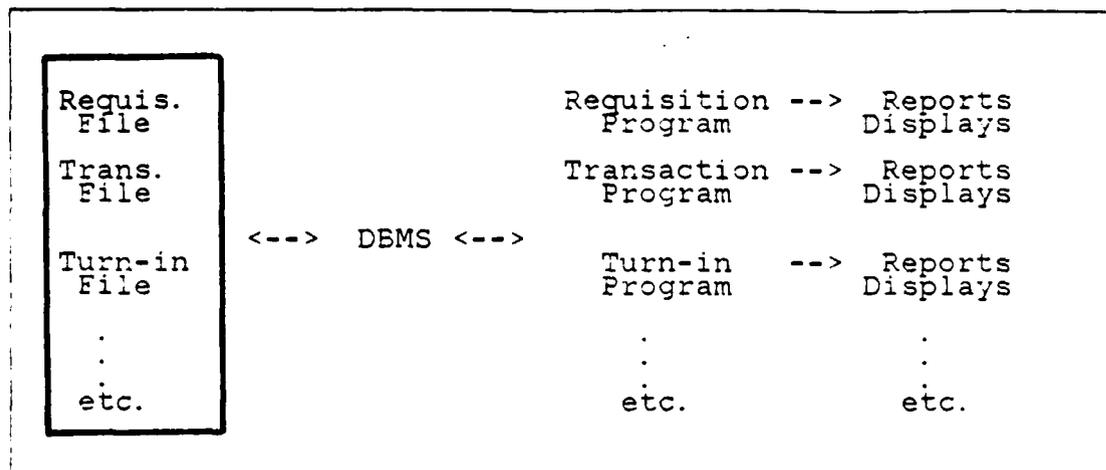


Figure 3.3 Database System for SAMS.

It was previously mentioned that a reduction in redundant data was possible through intelligent file, or relation design. Good design consists of properly matching the data elements, or attributes, of a relation so as to prevent data integrity problems. Wetherbe and Dickson [Ref. 22: p. 213] describe the common integrity problems as:

1. Difficulty in accurately identifying, locating, or updating all records given a specific set of attributes.
2. Needlessly repeating fields in records, therefore requiring redundant storage and updates.
3. Inconsistencies among data.
4. No records within which to store certain fields.

The process of normalization is a set of rules or guidelines which help a database designer build relations that minimize update anomalies and data inconsistencies. For larger systems with a high transaction rate, strict compliance with all of the normalization rules can lead to conflict with retrieval performance because normalization generally results in more relations. This then requires more effort to retrieve all the necessary data. The SAMS however, will be a relatively small system with a low transaction rate, and therefore the design should strive for maximum reasonable normalization. Excellent descriptions and examples of the normalization rules are given in Kroenke [Ref. 18: p. 286] and Kent [Ref. 23].

There are seven normal forms identified, and in consideration of the type of data involved in the SAMS, it may be difficult to apply higher than the Boyce-Codd

normal form (BCNF) in most cases. Briefly, the guidelines, or rules up to this point are:

1. All records in a relation must contain the same number of fields and none must repeat. This is basically the definition of a relation in relational database theory.
2. The second normal form requires that all non-key attributes are dependent on the key. A key is one or more attributes that determine a record.
3. Third normal form is satisfied when no non-key attribute provides a fact about another non-key attribute.
4. Boyce-Codd normal form requires that every attribute that provides a fact about another attribute (determinant), must be capable of being a key for the relation (i.e. a candidate key).

Thus, analysis concludes with the generation of DFDs, the functional descriptions, identification of the data and its structure, and consideration of the normalization policies for the database relations.

IV. SAMS DESIGN

The design of the database relations for the SAMS is a critical juncture in the development process. For the reasons discussed in the previous chapter, thoughtful design of these relations will ensure the flexibility and integrity of the system.

Relation design is not an easy task, even for a relatively small system like SAMS with only a hundred or so data elements. A starting point is to logically group data elements as was done for the external reports and stock records in Table 2 . We must also then consider the data elements that deal with generic ammunition information, allowance lists, NAR messages, addresses of other Navy units, and constant data referring to a particular ship (static data). If we assemble these lists independently of each other, there is much data duplication which must be minimized for the normalization we require.

To store data elements in one relation, yet link them to other relations, there must be common attributes between the two relations. However, these attributes must be unique so that the DBMS can locate the correct record. This normally requires that the relationship be established to a key attribute. DBase III Plus requires that the attribute that is being searched for be an "index" of the relation. An index is a file that is created for an associated database relation which contains a particular attribute in alphabetical, or alphanumeric order and its associated record number in the database. This allows the DBMS to quickly search for a particular value of an attribute in the index file and obtain the corresponding record number in the database file. When records are added or deleted from a relation, only the indexes are reorganized which could save considerable time in a large database. Moreover, many indexes can exist, on different attributes, for the same relation.

With these considerations in mind, the SAMS data elements are separated into appropriate relations and indexed as deemed appropriate to accomplish the file relationships desired. The relation structures will be fully described with regard to normalization in the next section.

Although compatibility in data elements or software is not required of the SAMS, as it is a stand alone system, it was desired that data elements used in the CAIMS be used by SAMS where possible. This resemblance would minimize any

difficulties in user understanding when referring to supply system or CAIMS publications. Also, since CAIMS already has a Data Element Dictionary (SDED), it is a question of not "reinventing the wheel". If data elements in this dictionary were directly applicable to the SAMS, they were used. Minor modifications in data element name or picture were necessary in some cases (COBOL vs. dBase III Plus), but the meanings were close enough to assign the CAIMS Data Element Number (DEN) to the SAMS element.

Table 4 lists the conventions used in selecting the SAMS database and index names. It is easy to see that some sort of pattern is necessary in selecting file names, their number quickly grows beyond human memory. Table 5 conveniently lists the database files and their associated index and format files. A format file is constructed by the DBMS, with user interaction, to create custom data entry screens. Even this relatively small system is composed of over fifty programs and fifty files, therefore program and file directories are also needed. A complete program directory is included as Appendix C.

A. SAMS PHYSICAL RELATION DESCRIPTION

Table 6 lists the physical structure for all of the SAMS database files. It would be helpful to the reader to refer to this table throughout this discussion.

1. Transaction File (ATR File)

This file contains the data elements that relate specifically to a particular transaction. The key in this file is the ATR serial number (ATR SERIAL). To normalize to the Boyce-Codd Normal Form (BCNF), every determinant must be a candidate key, and the only key here is the ATR serial number. Every other data element should furnish a fact about the key, and only the key, and this is satisfied with one exception mentioned later.

As mentioned in Chapter II, ATR numbers run from 001 to 999 and then repeat. It would take quite a few years for the typical ship to go through 999 numbers. Repeating key elements can not be permitted so if there was a chance of this, the old records would be archived. Old instructions required that ships restart their ATR numbers when "chopping" (Change of Operational Command) to another fleet. If this situation existed on a ship that SAMS was to be installed, the old fleet numbering sequence should not be included in the initialization to prevent repeating key field values.

TABLE 4
SAMS FILE NAMING CONVENTIONS

A. Database Files (.dbf)

1. First two letters - system prefix- Ammunition Management "AM"
2. Third-Eighth letter - description of file contents

B. Index Files (.ndx)

1. First two letters - system prefix - Ammunition Management "AM"
2. Third letter - .dbf file description
 - a. Requisition - "R"
 - b. Ammunition Data - "A"
 - c. Transaction - "T"
 - d. Inventory - "I"
 - e. Turn-in - "U"
 - f. Allowances - "W"
 - g. NAR Action - "N"
 - h. NAR Serial - "E"
 - i. Static Data - none
 - j. Address - "D"
3. Fourth-Eighth letter - index file description

C. Format Files (.fmt)

1. First-Third letter - "ADD"
2. Fourth-Eighth letter - description of .dbf file contents

TABLE 4
SAMS FILE NAMING CONVENTIONS (CONT'D.)

D. Exceptions

1. DBSYSTEM.db - User's File (encrypted)
2. CONTFILE.dbf - Contains File - Programs <-> .dbf files
3. DATAELEM.dbf - Data Element File
4. DICFILES.dbf - Files Directory
5. PROGFILF.dbf - Programs File

The ATR Status is a local SAMS data element which serves to indicate the condition of a particular record. A transaction can be incomplete, ready for submission, or submitted. An "incomplete transaction" is one in which the user has not completed entering the necessary data or he is not ready to say the data is correct. This is strictly a user assigned status and no inventory update takes place based on an incomplete or blank entry. When the user is satisfied that the data is correct, he changes the status to "ready for submission", and inventory update takes place. At this point he may print the ATR for submission to the Radio Room or Communications Center, however he may not edit or delete the record. When the message is routed back from the communications personnel, verifying broadcast, the status can be changed to "submitted", and the message Date Time Group (DTG) entered. The DTG may be used on subsequent ATRs and thus must be retained.

The National-Item-Identification-Number (NIIN) uniquely identifies a type of ammunition and can therefore be the link to the Ammunition Data file to obtain the Type of Maintenance Due Code (TMDC). The TMDC will be required on ATRs that involve Serial/Lot Item Tracking (SLIT) material.

To uniquely identify an ammunition item that is or was physically in the ship's inventory, not only the NIIN, but the Condition Code (CC), Activity Classification Code (ACC), Lot Number, and Component Serial Numbers (if applicable), are required. Multiple items with serial numbers can be included on one ATR, so up to ten serial numbers may be included in this record. Unfortunately, in one regard, variable

TABLE 5
SAMS FILE GROUPS

DB File (.dbf)	Index File (.ndx)	Format Files (.fmt)
AMREQUIS	AMRSERUP AMRSERDW AMRREQDD	ADDREQUI
AMMODATA	AMANALC AMACOGSY AMAMCONC AMANIIN	ADDAMMO AMMODATA.dbt (memo field)
AMTRANS	AMTSERUP AMTSERDW AMTTRCD AMTNIIN	ADDTRANS
AMINVEN	AMINIIN AMICONCD AMIACCL	ADDINVEN
AMTURNIN	AMUSERUP AMUSERDW AMUNIIN	ADDTURN
AMALLOW	AMWNALC AMWALTP	ADDALLOW
AMNARACT	AMNNIIN AMNSERUP AMNSERDW	ADDNARAC
AMNARSER	AMENARSR	ADDNARSE
AMSTDATA	none	ADDSDATA
AMDADDR	AMDUIC AMDACTNM	ADDADDRE
DBSYSTEM.db	none	none
CONTFILE	CONTNAME	none
DATAELEM	DATANAME DATAELSR	DATAELDIC
DICFILES	DICFILTP	none
PROGFILE	PROGNAME	none

record lengths are not defined in a relational data base system, so some storage space will be wasted because most transactions do not involve items with serial numbers. Only rarely would a transaction involve more than ten serial numbers. In this case, a second transaction report could be prepared.

The ATR Julian Date (ATRJULDAT) is the three-digit julian day of the year that the ATR is prepared and appears in the header line of the message. It will not necessarily be the same as the message DTG.

The beginning and ending balance must be maintained in the Transaction file because the Inventory file is dynamic in nature having the "instantaneous" stock levels. Thus the Transaction file is the historical record of stock levels.

Quantity seems an obvious data element in the ATR, however if the transaction is a receipt or issue, the document number of the associated receipt or turn-in document is also included. This case would violate the Third normal form because quantity is part of these documents. However, Quantity must be included because expenditure transactions have no corresponding document.

The User Source Identification Code (IDCODE) is a similar situation. For receipts and issues the IDCODE would be the "ship to" or "received from" on the shipping document, however in other cases it can take on other values. Thus it must be included as part of the record.

Finally, any narrative message (ATRREMARKS) must be saved and this is naturally the only place to do that.

The preceding description of the Transaction file and its structure may have seemed rather tedious, however the reader should take heart that commonalities in the other files will not be repeated.

2. Requisition File

Serial number and julian date uniquely identify the requisition, while the other two components of a requisition document number are accessed from the Static Data file. The NIIN again uniquely identifies the item that is being ordered.

The requisitioner and supplemental address UIC and service codes are included in the requisition file. Ninety per cent of the time, an activity's UIC will determine its service designator code, however if that activity is a ship and it changes fleets, then its service designator code will change. The supplemental address is normally the activity at which the material will be received, or loadout activity.

Requisition Status (REQUISSTAT) serves a similar purpose to that in the ATR file except that it must also be able to represent partially filled and cancelled requisitions.

The remaining data elements are multi-valued codes which are all independent and describe the circumstances of the requisition, urgency of need, required delivery

date, etc.. See Appendix B, the Data Element Dictionary, for a complete description of these items. This file also satisfies the Boyce-Codd Normal Form.

3. Turn-in File

The turn-in file may seem to be a redundant file in itself because it will eventually represent a turn-in transaction (issue transaction). However, two considerations help to clarify the need for the file. First, a turn-in document may be prepared well ahead of the actual transaction date and the timing of the inventory update would become a problem. Certainly the inventory should not be updated until the physical stock levels change. Secondly, a turn-in document requires three or four data items that are not needed on a transaction report and thus the ATR file would be made needlessly larger just to accommodate these items.

The turn-in file record must be able to uniquely identify an item of inventory, just as the ATR file must. Therefore the same five data elements: NIIN, ACC, CC, lot number, and serial number(s), must be included to accomplish this.

The ATR serial number is provided to link the record to the transaction record in the ATR file. This data would be automatically appended to the record when the transaction document was prepared.

Finally, a link to the NAR Action file is necessary if the turn-in is in response to a NAR message. The NAR Serial member and the NAR Year accomplish this.

4. Ammunition Data

This file contains data elements that provide facts about specific items (types) of ammunition. The unique element is of course NIIN. This file essentially replaces the need to reference two or three large publications and is accessed many times in the application programs. The Data Element Dictionary, Appendix B, contains complete explanations of these independent descriptors.

One "confusion factor" still remains however. Although NIIN reporting, vice NALC reporting, is logical and consistent with good inventory control, NALC is still used as an identifier in NAR messages, allowance lists, and other documents. For years, NALC has been the defacto unique identifier of ammunition items, and until all organizations begin using NIIN as such, the Ammunition Data file will serve as the NAR-to-NIIN converter in application programs.

It is questionable if the NALC in this relation could be considered a candidate key. Many of the data elements realistically do provide a fact about the NALC, a non-key field, so the Third Normal Form would be violated. This seems unavoidable in this case however.

5. Inventory File

This file is surprisingly simple. The five identifiers of an individual item are there along with the MDD which will be associated with serial number controlled items. If the item is serial number or lot serial number controlled (MCC "C" or MCC "E"), the quantity for the record will be one. Items that are only lot controlled or have no MCC will have a quantity that reflects all the items in the lot. This relation has such a large key (five data elements), that there are only two others that are not included, quantity and storage location. Boyce-Codd Normal Form is easy to attain in this case.

6. NAR Action File

This file contains reclassification data from a NAR message that affects inventory held on board. Only those NARs that are applicable are entered here. The NAR Serial File records all NAR messages received so that any missed messages can be noted.

Naval Ammunition Reclassification messages identify affected ammunition by NALC, lot number, and sometimes serial number. Since lot numbers are not necessarily unique, and a NALC may contain several NIINs, we must first determine the NIINs associated with a particular NALC. The inventory can then be searched for those NIINs, which are a key element of the Inventory file as well as an index. Upon locating an applicable NIIN, the lot number is compared with that contained in the NAR message. This procedure would be much simplified if NAR messages used NIIN as the identifier.

The NAR Action File also contains the ATR Serial Number which took the action the NAR called for. This would be automatically appended when a reclassification ATR was processed.

Finally, this file contains a character field to record the reason for reclassification, which is normally included in the message itself. Also a field, TAGLABEL, to record the label or tag that must be attached to the actual ammunition to explain its status. This label would be a statement like, "For Training Use Only", or "For Emergency Combat Use Only".

7. NAR Serial File

This file's purpose is to provide a convenient location to record all the NAR messages received, whether applicable or not. This method allows the NAR Action File to be much smaller, since non-applicable NARs are excluded. Thus we minimize the size of a relation with nineteen fields by creating one with only three. This file is in Fifth Normal Form, which means that it is in BCNF and it contains no transitive dependencies (fourth), and it can not be subdivided in any way.

8. Allowances File

This file contains the authorized types of ammunition and quantities that a ship may carry. Allowance Lists, promulgated by the Naval Sea Systems Command (NAVSEA), again use the NALC as an ammunition identifier [Ref. 3: Chapter 7]. Remembering that the various NIINs within a NALC group are functionally equivalent, it is clear why NAVSEA publishes the lists in this manner. A ship may carry any NIIN item within the NALC group to satisfy the Allowance List requirements.

The inclusion of the Activity Classification Code (ACC) in the Allowance file allows ammunition for other end uses to be shown in this file rather than creating a different file for each ACC material carried onboard.

Finally, this file contains a computed quantity, USED FY, which will be automatically updated during expenditure ATR processing to show the quantity of ammunition, which has a training allowance, that has been used in the fiscal year. The system manager will need to reinitialize this quantity each year. Performing the computation at this time and storing the value is a departure from strict normalization, however it is far more efficient than doing all of the calculations just prior to printing a training expenditure report.

9. Address File

The Address file contains pertinent data about other supply activities and commands with which a ship may conduct ammunition business. The UIC and Activity Name (ACTIVNAME) are both candidate keys, however UIC is less easily confused than two similar names and it is also only five characters. This makes it better as an index, decreasing search time. There are transitive dependencies in this file: Activity Name and Location can determine Service Designator Code, however Boyce-Codd Normal Form is still attained.

10. Static Data File

This file is unique in that it only contains one record, namely data about the ship that the SAMS is installed in. It furnishes information to application programs that need the ship's name, hull number, etc. to print documents and reports. The Fund Code is needed on requisitions and the Monitoring Activity (MONITACTIV) is combined with the Cognizance Symbol to form the Distribution Code, used on several documents.

The remaining database files are for system documentation and will not affect the operational performance of the system.

B. SAMS STRUCTURE CHARTS

The last step in the design stage is to determine how the system will be partitioned into modules, or programs. These programs will operate on the database relations, via the DBMS, and interactively with the user to perform their function. Unlike a Data Flow Diagram (DFD), the structure chart presents system hierarchy, control, and communication. Appendix D presents the SAMS structure charts.

Structured design has several measures with which to gauge the quality of a system's structure chart. The first is coupling, which is the degree of independence between modules, and the objective is to minimize this. The second is cohesion, which evaluates how closely the activities within a module relate to one another. A module should be considered a "block box", in that it performs its designated function with the surrounding modules knowing little or nothing about its internal code. [Ref. 15: p. 101,117]

A database management system tends to make structured module design an easier process than in the past. The problems of data storage are removed from the application programs and handled by the DBMS, thus traditional input output modules are not necessary. This fact allows programs to flow from one logical process to another without the traditional housekeeping chores. Previously, structured design techniques responded to this by making modules very small with single cohesive functions. A DBMS with a good programming language can produce programs that remain understandable and cohesive while allowing a chain of logical thoughts to be contained in one program.

The SAMS is menu-drive when the selections are unambiguous, and almost conversational when the selections require a more detailed knowledge. Referring to the

TABLE 6
SAMS PHYSICAL FILE STRUCTURES

Relation: Transactions					
DBMS Name: AMTRANS.dbf				key (ATRSERIAL)	
Field	Field Name	Type	Width	Dec.	DEN
1.	* ATRSERIAL	Num	3	0	C089
2.	ATRSTATUS	Char(A)	1		
3.	NIIN	Char	9		D046D
4.	ACTCLCODE	Char(A)	1		E303
5.	CONDCODE	Char(A)	1		C003E
6.	TRANSCODE	Char(A)	1		D219
7.	ATRJULDAT	Num	3	0	K002B
8.	LOTNUMBER	Char	16		C301
9.	OSERNUM/	Char	16		D330
10.	9SERNUM	Char	16		D330
11.	MDD	Char	3		C026
12.	BEGBALANCE	Char	5		
13.	ENDBALANCE	Char	5		
14.	QAUNTITY	Num	5	0	
15.	DOCSVCCOD	Char(A)	1		K048
16.	DOCUIC	Char	5		A002
17.	DOCJULDAT	Num	4	0	K002C
18.	DOCSENUM	Num	4	0	K002B
19.	IDCODE	Char	5		I200/I602B
20.	MESSAGEDTG	Char	14		C076
21.	ATRREMARKS	Memo	10		

Index Files:	Field	Name
	1	AMTSERUP (increasing)
	3	AMTSERDW (decreasing)
	6	AMTNIIN
		AMTTRCD

Relation: Requisitions					
DBMS Name: AMREQUIS.dbf				key(SERIAL, JULIANDATE)	
Field	Field Name	Type	Width	Dec	DEN
1.	NIIN	Char	9		D046D
2.	QUANTITY	Char	5		
3.	* SERIAL	Num	4	0	K002C
4.	* JULIANDATE	Char	4		K002B
5.	PROJCODE	Char	3		K024
6.	SENDTOSERC	Char	1		K048
7.	SENDTOUIC	Char	5		A002
8.	DOCIDENTIF	Char	3		K001
9.	MEDIASTAT	Char	1		K082
10.	DEMANDCODE	Char(A)	1		K020
11.	SUPADDSERC	Char	1		K048
12.	SUPADDUIC	Char	5		A002
13.	SIGNALCODE	Char(A)	1		K021
14.	PRIORITYCD	Char	2		K025
15.	REODELDATE	Char	3		K018
16.	ADVCECODE	Char	2		K026
17.	REQUISSTAT	Char(A)	1		

Index Files:	Field	Name
	3	AMRSERUP (increasing)
	3	AMRSERDW (decreasing)
	15	AMRREQDD

TABLE 6

SAMS PHYSICAL FILE STRUCTURES (CONT'D.)

Relation: Inventory
 DBMS Name: AMINVEN.dbf key(NIIN, LOTNUMBER,
 SERNUMBER, CONDCODE, ACTCLCODE)

Field	Field Name	Type	Width	Dec	DEN
1.	* NIIN	Char	9		D046D
2.	* LOTNUMBER	Char	16		C301
3.	* SERNUMBER	Char	16		D330
4.	MDD	Char	3		C026
5.	* CONDCODE	Char(A)	1		C003E
6.	* ACTCLCODE	Char(A)	1		E303
7.	QUANTITY	Num	5		
8.	STORAGE LC	Char	30		

Index Files:

Field	Name
1	AMINIIN
6	AMICONCD
7	AMIAACCL

Relation: NAR Action
 DBMS Name: AMNARACT.dbf key(NARSERIAL, NARYEAR)

Field	Field Name	Type	Width	Dec	DEN
1.	NIIN	Char	9		D046D
2.	LOTNUMBER	Char	16		C301
3./	OSERNUM	Char	16		D330
12.	9SERNUM	Char	16		D330
13.	* NARSERIAL	Num	3	0	C084
14.	* NARYEAR	Num	2	0	C083
15.	OLDCONDCD	Char(A)	1		C003E
16.	NEWCONDCD	Char(A)	1		C003E
17.	ATRSERIAL	Num	3	0	C089
18.	NARREMARKS	Char	40		
19.	TAGLABEL	Char	30		

Index Files:

Field	Name
1	AMNNIIN
4	AMNSERUP { increasing }
4	AMNSERDW { decreasing }

Relation: NAR Serial
 DBMS Name: AMNARSER.dbf key(NARSERIAL, NARYEAR)

Field	Field Name	Type	Width	Dec	DEN
1.	* NARSERIAL	Num	3	0	C084
2.	* NARYEAR	Num	2	0	C083
3.	NARDTG	Char	14		C078

Index Files:

Field	Name
1	AMENARSR

TABLE 6
SAMS PHYSICAL FILE STRUCTURES (CONT'D.)

Relation: Data Elements
DBMS Name: DATAELEM.dbf

Field	Field Name	Type	Width	Dec	DEN
key(NAME, SOURCE_FIL)					
1.	*	NAME	Char	10	
2.		PICTURE	Char	6	
3.	*	SOURCE_FIL	Char	50	
4.		DESCRIPIO	Char	240	
5.		DEN	Char	6	
6.		LONGTITLE	Char	55	
7.		CODES	Memo	10	
8.		REFERENCE	Char	70	
Index Files:					
		Field	Name		
		1	DATANAME		
		3	DATAELSR		

Relation: Contains File
DBMS Name: CONTFILF.dbf

Field	Field Name	Type	Width	Dec	DEN
key(NAMEOFPROG)					
1.	*	NAMEOFPROG	Char	8	
2.		CONTAINS	Char	50	
Index Files:					
		Field	Name		
		1	CONTNAME		

Relation: Systems File
DBMS Name: DICFILES.dbf

Field	Field Name	Type	Width	Dec	DEN
key(FILE_NAME)					
1.	*	FILE_NAME	Char	8	
2.		FILE_TYPE	Char	4	
3.		DESCRIPIO	Char	40	
Index Files:					
		Field	Name		
		2	DICFILTP		

Relation: System Programs
DBMS Name: PROGFILF.dbf

Field	Field Name	Type	Width	Dec	DEN
key(PROG_NAME)					
1.	*	PROG_NAME	Char	8	
2.		CALLS	Char	100	
3.		PURPOSE	Char	70	
4.		CALLED_BY	Char	80	
Index Files:					
		Field	Name		
		1	PROGNAME		

TABLE 6
SAMS PHYSICAL FILE STRUCTURES (CONT'D.)

Relation: User's File		key(GROUP_NAME, LOG_IN_NAM, PASSWORD)				
DBMS Name: DBSYSTEM.db						
Field	Field Name	Type	Width	Dec	DEN	
1.	* GROUP_NAME	Char	8			
2.	* LOG_IN_NAM	Char	8			
3.	* PASSWORD	Char	16			
4.	ACCCUNT_NM	Char	24			
5.	ACCESS_LVL	Num	1	0		

NOTE: This is a special relation established within the PROTECT program of the dBase III+ and is encrypted. It may only be modified by the DB Administrator. Fields one, two, and three are mandatory. The access level may be one through eight, with one giving the most priveledges.

major subsystem diagram in Appendix D, we see that the main menu allows the user to select one of nine options, each is completely independent of the other. This is an example of low coupling which is desireable. Each subsystem then presents another menu which will determine what the user desires to do. The SAMS generally has four levels of hierarchy in it's application programs:

1. The main menu
2. The subsystem menu
3. The application desired
4. Utility programs (infrequent)

Examples of utility programs can be seen in the Requisition Management subsystem under the Print Requisition application module.

In general, very few parameters are passed in the system. Facilitating this is the fact that in dBase III Plus , a called program can manipulate variables in the caller program without parameters being passed. This can be convenient, but it can also have dangerous implications. The programmer must ensure that the called program does not inadvertently alter variables in the main program that were not intended. Of course, by not using parameters, a programmer limits the usefulness of his programs by making them dependent on a particular application. For further background on parameter passing and implicit and explicit inheritance the reader is referred to MacLennon [Ref. 24].

Structure charts provide the programmer with a framework for the implementation of the various subsystems and individual modules, while the Data Flow Diagrams provide a guide to the logical processes of the subsystems. Neither graphical representation should be regarded as "cast in concrete" however, the physical implementation may suggest modifications.

V. IMPLEMENTATION

A. IMPLEMENTATION PROGRESS TO DATE

Early in the development of the SAMS, it was obvious that complete implementation and field testing would be quite impossible in the available research time. Therefore the author selected the portions of the implementation that would be the most beneficial to anyone continuing the effort, and provide a framework for that implementation.

The design stage identified all of the programs necessary for the SAMS. They are listed in the PROGFILE database and may be printed with their associated data by running PROGFILE.prg. Appendix C contains this listing. Approximately one quarter of the systems programs have been implemented.

The Requisition Management subsystem was fully implemented because its programs demonstrate a myriad of programming techniques to accomplish their tasks. There are modules to review databases, edit/delete, create documents, print documents, backup files, and various forms of interactive programming are demonstrated. Most of the other SAMS subsystems use many of these processes, so a close review of the Requisition Management programs can significantly decrease the learning curve for future programmers. Completed program listings are contained in Appendix E.

In addition, the database, index, and format files necessary to operate the Requisition Management subsystem were created and loaded with sample data. Testing was performed throughout the implementation of these modules to show correct operation, although not exhaustive or as a dedicated separate evolution. System testing is obviously a critical phase of the development process not to be slighted. Module independence will greatly facilitate testing and make rapid debugging possible.

Many of the system documentation modules were also completed during the analysis and design stages and should greatly assist future programmers. These include the Data Element Dictionary, structure charts, program files, files directory, and cross reference between programs and relations.

B. CODING STYLE

Without repeating at length the ideas that have been presented throughout this paper concerning the interactive style of the SAMS, it might be worthwhile to bring together in one spot the essential points.

First, SAMS must be easy to use. Although larger ships may be able to dedicate a single person to the responsibility for system operation as a primary duty, this will normally not be the case. Therefore the slant should be toward more information, rather than less, in the interactive process. This assumes the operator will not be so familiar with the system that the extra information will be annoying. The Requisition Management option presents the user with most of the information that could be gleaned from several publications. The difference is that the publications contain a whole range of logistics information, and the SAMS will present pertinent information that fleet users need and no more. Therefore, the key phrase is "complete, selective information".

The system administrator, or manager, must be knowledgeable of the instructions and publications that explain the present system, and the operators must have a working knowledge. The manager should understand the basics of dBase III Plus, not to the degree of being able to program, but enough to be able to manipulate relations should a problem arise. Moreover, he should understand the implications of his ability to alter those relations.

Finally, code documentation should primarily be contained within the listing itself. Placing it here will increase the chances of it being useful to a future maintenance programmer, and it will be easier to maintain of its own accord. Some authors feel that source code comments are perhaps the easiest form of documentation to maintain, [Ref. 17: p. 251 and 341], and this author would tend to agree.

VI. CONCLUSIONS AND RECOMMENDATIONS

The Shipboard Ammunition Management System presented in this thesis is one viable alternative to the present system. It automates much of the manual record keeping, can prepare useful management reports, and produce external documents compatible with the supply system's ADP equipment.

There are several factors that make a stand alone system difficult to design. First, it must be compatible with different processing systems. It must automate manual procedures on one hand and on the other hand it must produce documents that are compatible with machine readable capabilities of the shore establishment. This dichotomy has in part been a reason why the system has been getting more difficult for the "customer", the fleet user, to maintain. Machine readable ATR's may eliminate key punching errors at the SPCC level, however they are incomprehensible at the user level except to the preparer. Every message that leaves a ship should be checked for accuracy by at least two people prior to the commanding officer signing his permission to transmit, and this could run into considerable manhours if everyone must dig through publications to look up "funny codes and formats".

Thus, a second alternative to solve the problem is to extend automation capabilities to all ships from the shore based ammunition management systems. A unified system, with one set of rules, would definitely be superior to a fragmented system. The professional managers of ammunition ashore would then be able to extend their expertise to the fleet, in order to allow the professional users of the ammunition to practice that more, and management less.

A second problem, as this author perceives it, is the overlapping of directives between the supply system and the fleet logistics agents. The fleet user should deal with one set of rules, perferably one definitive reference. For example, in the Pacific Fleet, a submarine weapons officer has three levels of guidance concerning conventional ammunition management [Refs. 3,4] and operational force commander instructions. Now, there may be good reasons for modifications depending on the theatre of operations or weapon type, but the users guidance should come from one place, to minimize confusion. Consistently following this policy would probably reduce by one third the number of publications that a typical ship is required to carry, in weight if not in number.

The Shipboard Non-tactical ADP System (SNAP) is currently being installed on Navy ships in increasing numbers and is designed to handle many supply, maintenance, and administrative functions previously done manually. This could provide an excellent vehicle to automate the ammunition management function. This thesis and others could provide functional, if not design descriptions of such a subsystem to SNAP.

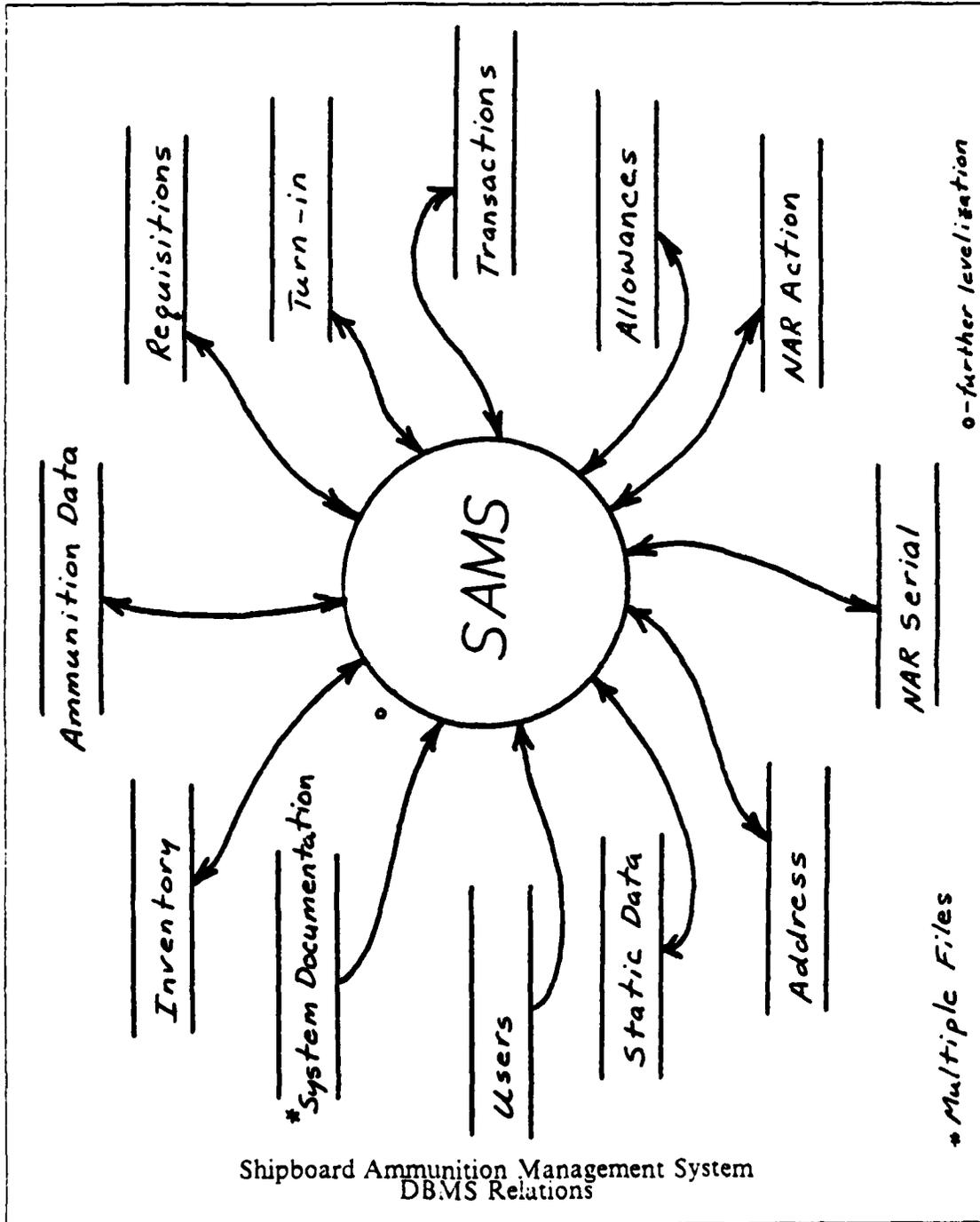
Finally, if the preceding alternatives can not be economically or politically justified, then a stand alone system, the SAMS, should be pursued. The author estimates two to four months of programming would complete the SAMS application programs, then a pilot ship should be selected for the initial installation. The SAMS should run in parallel with the existing manual system to gather user experience, document "bugs", and adjust user interface. This phase would be an extremely interesting follow-on thesis in itself. Ultimately there must be a program office for the SAMS, logically as part of CAIMS at SPCC, to provide guidance and installation assistance to the fleet.

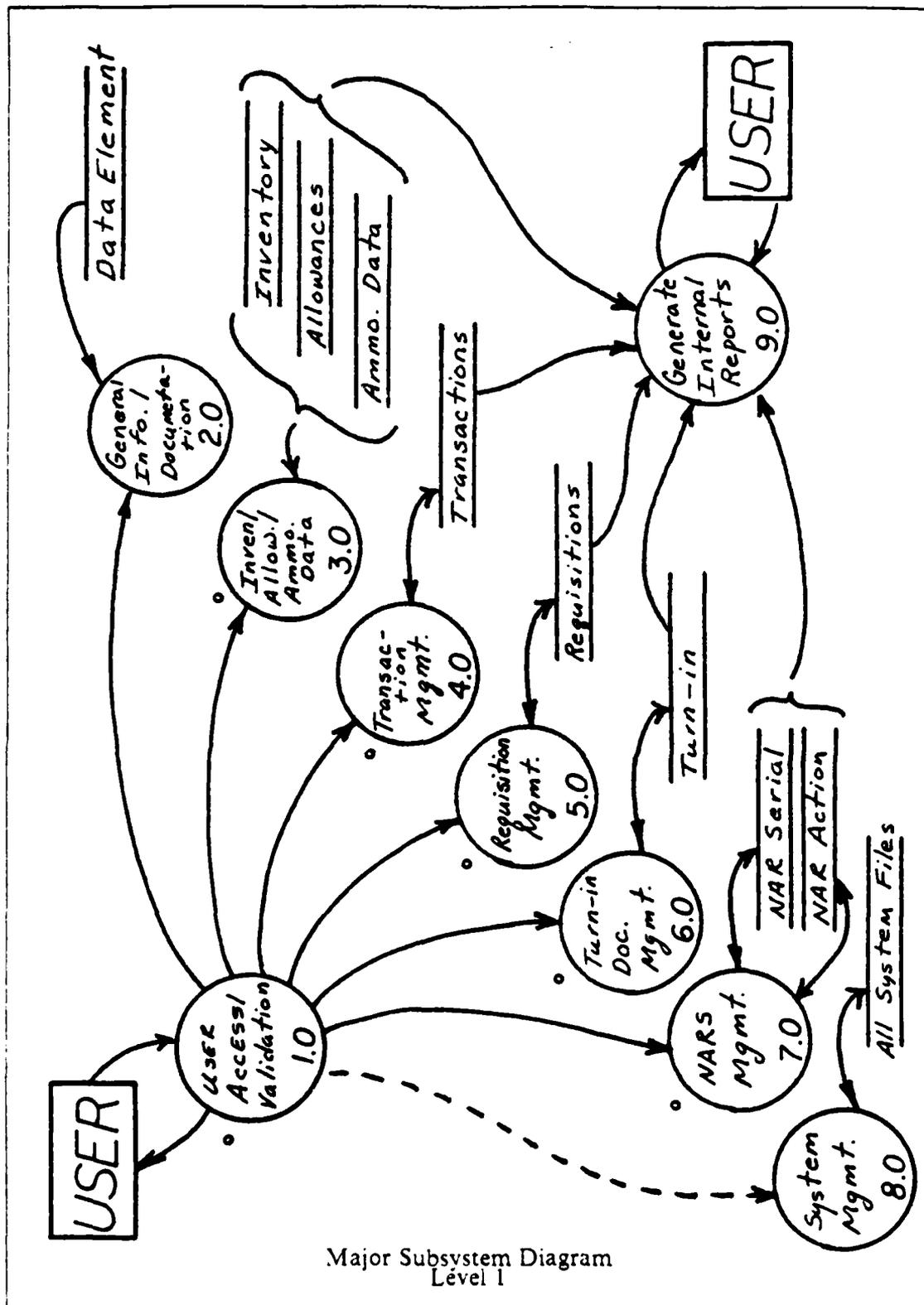
Therefore, the problems of conventional ammunition management can be handled in one of three ways:

1. An automated system within the CAIMS from SPCC down to all shipboard users.
2. Inclusion of ammunition management in SNAP.
3. Stand alone ammunition management - SAMS

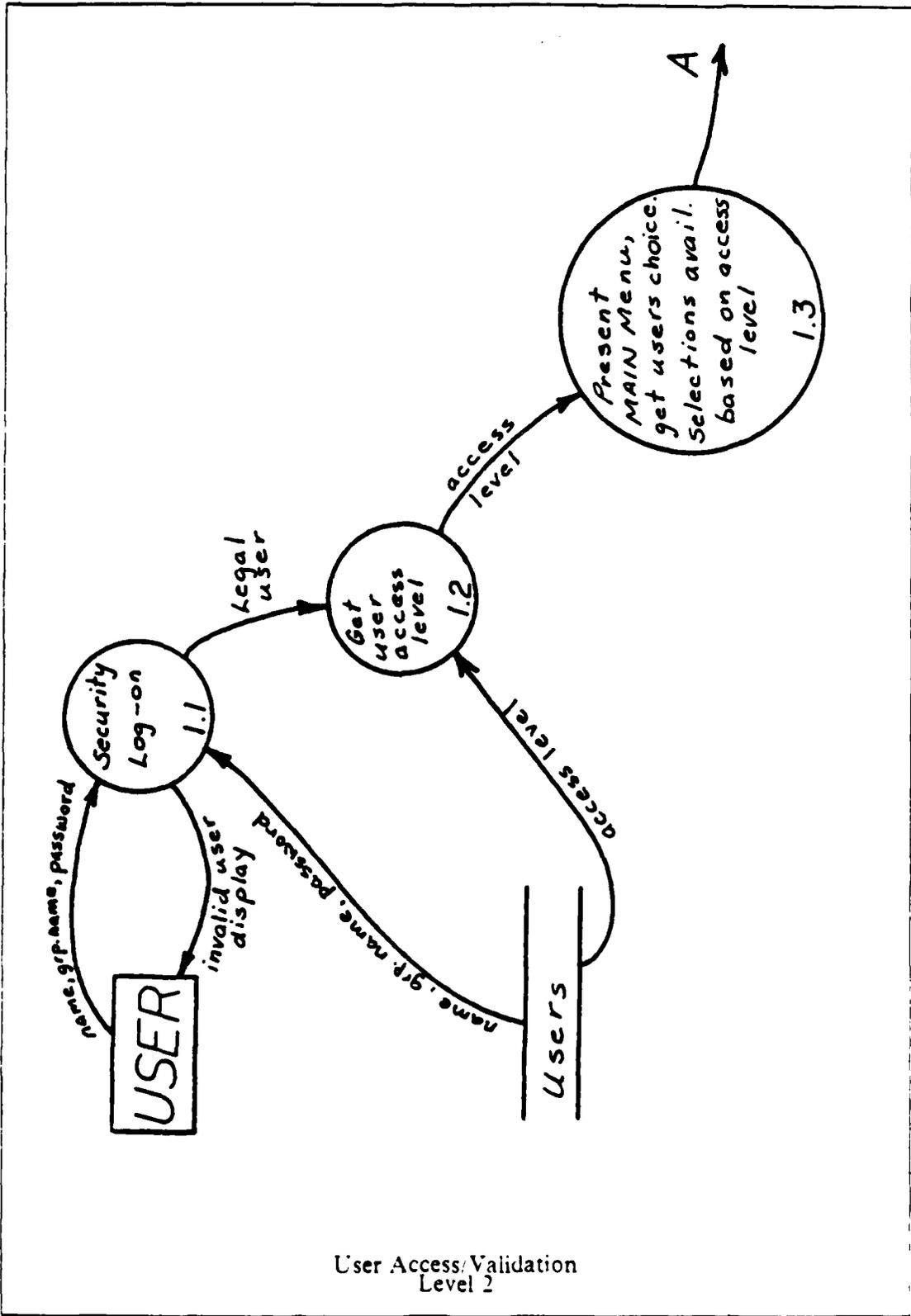
The goal in selecting any one of the three is to improve and unify the system, and to reduce the administrative burden on shipboard personnel. The status quo will not be suitable in the long term if we desire an increase in fleet readiness.

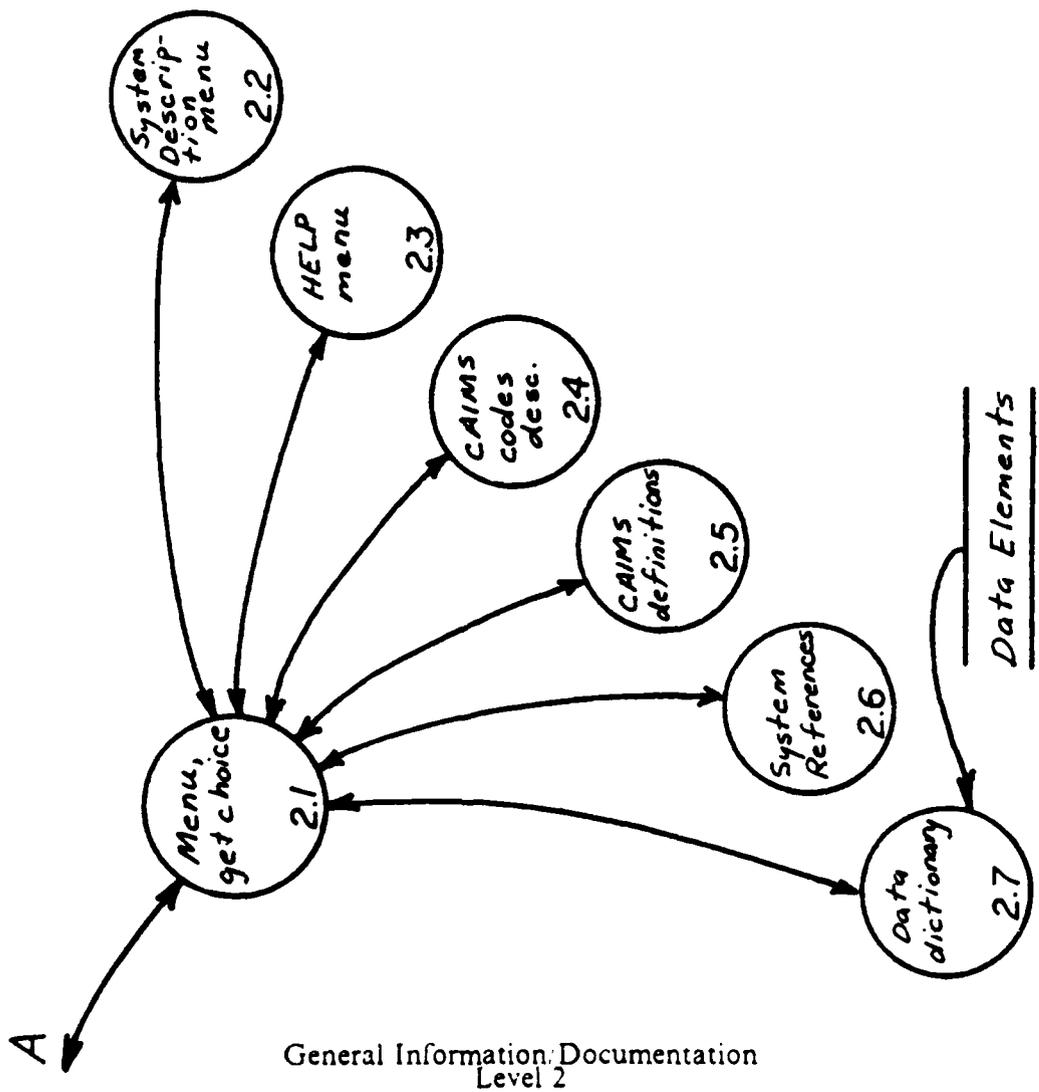
APPENDIX A
DATA FLOW DIAGRAMS



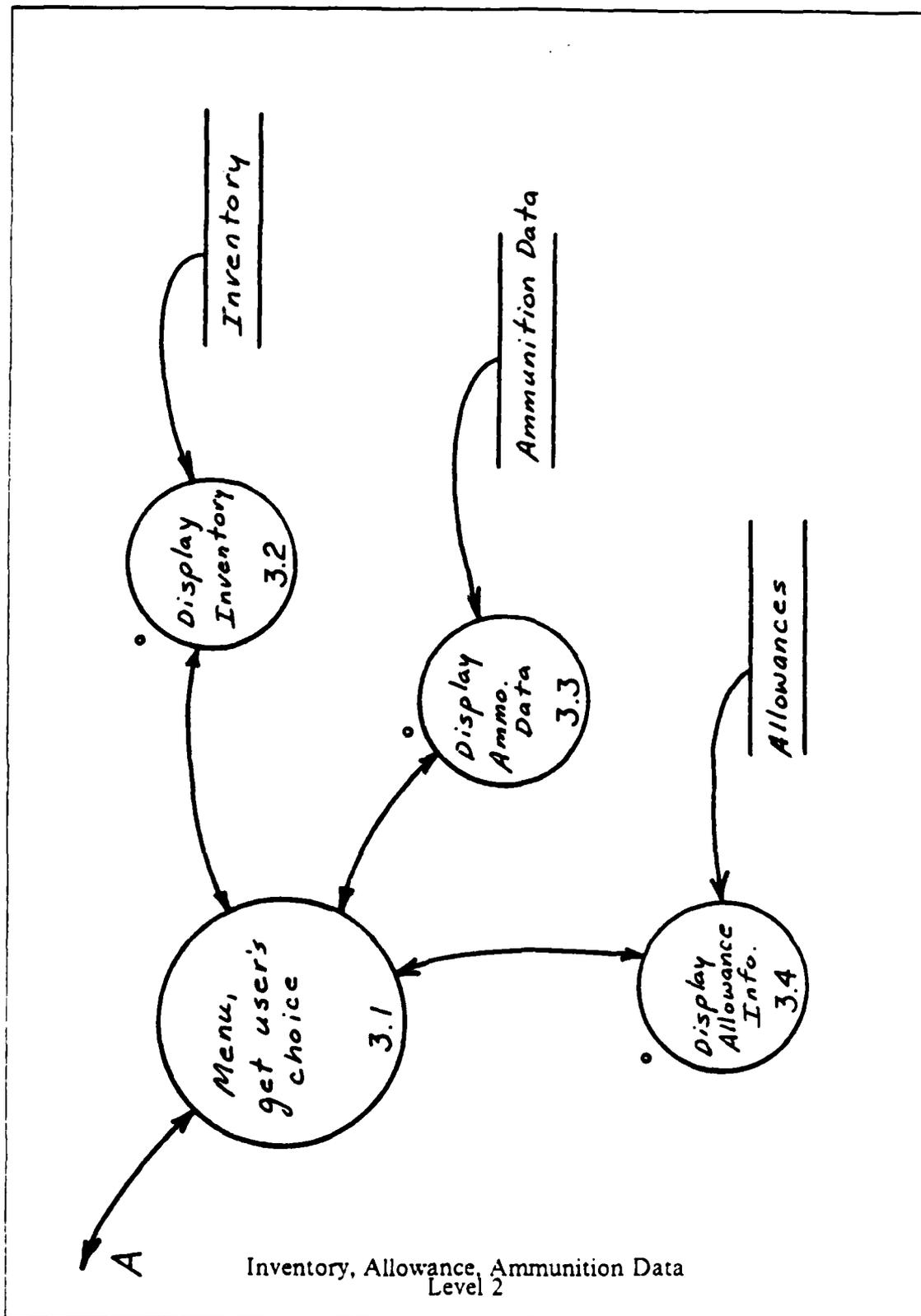


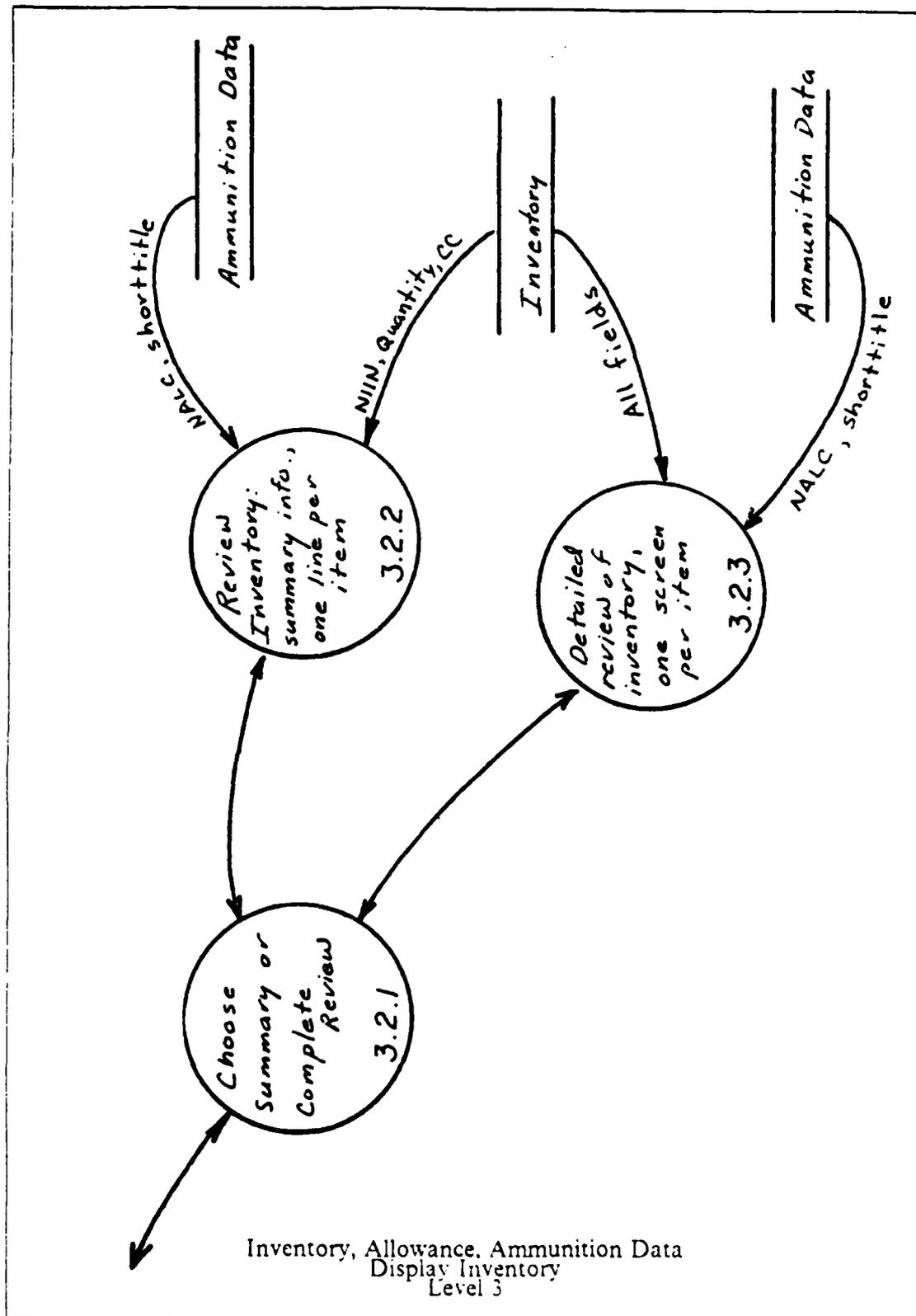
Major Subsystem Diagram
Level 1

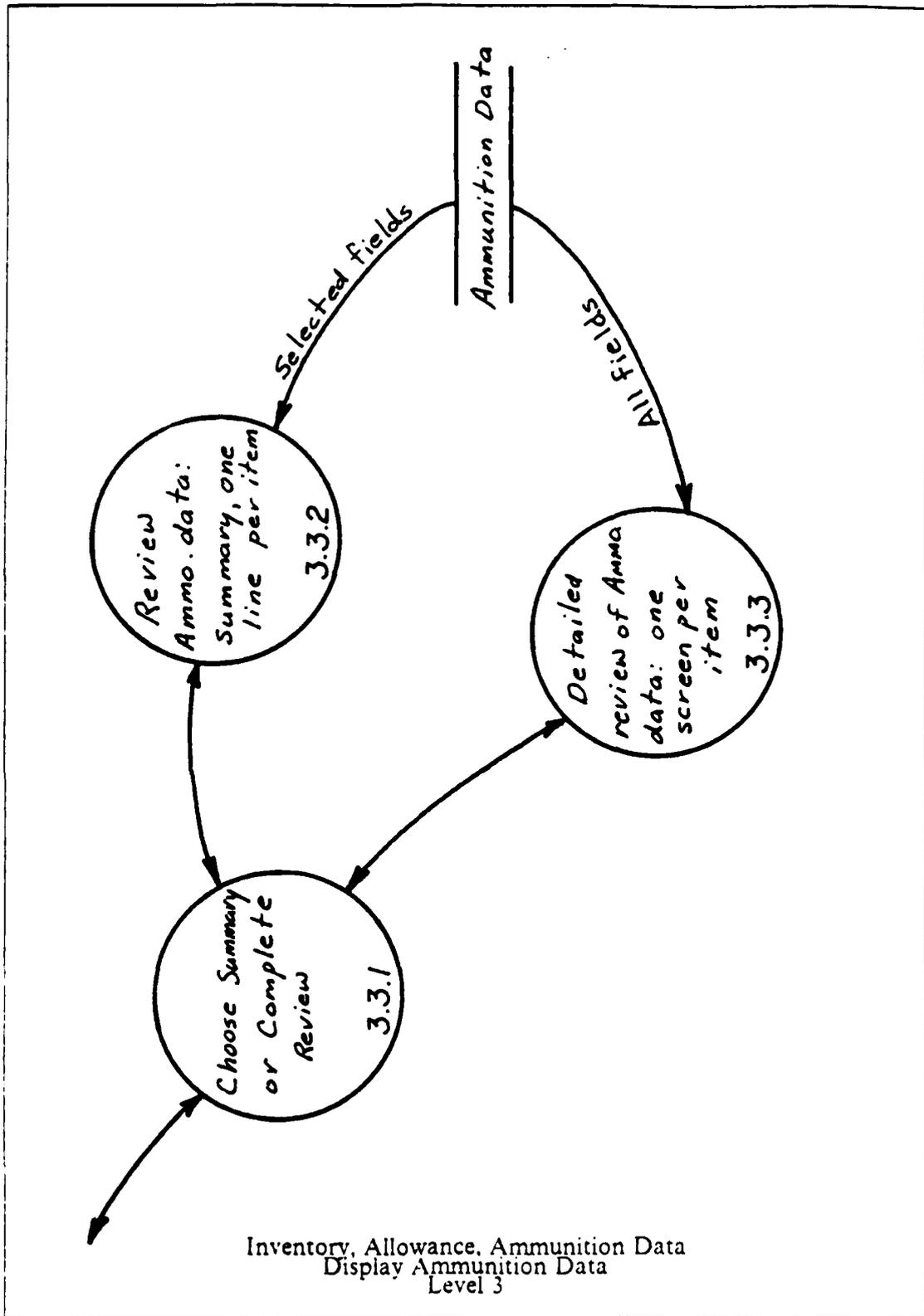


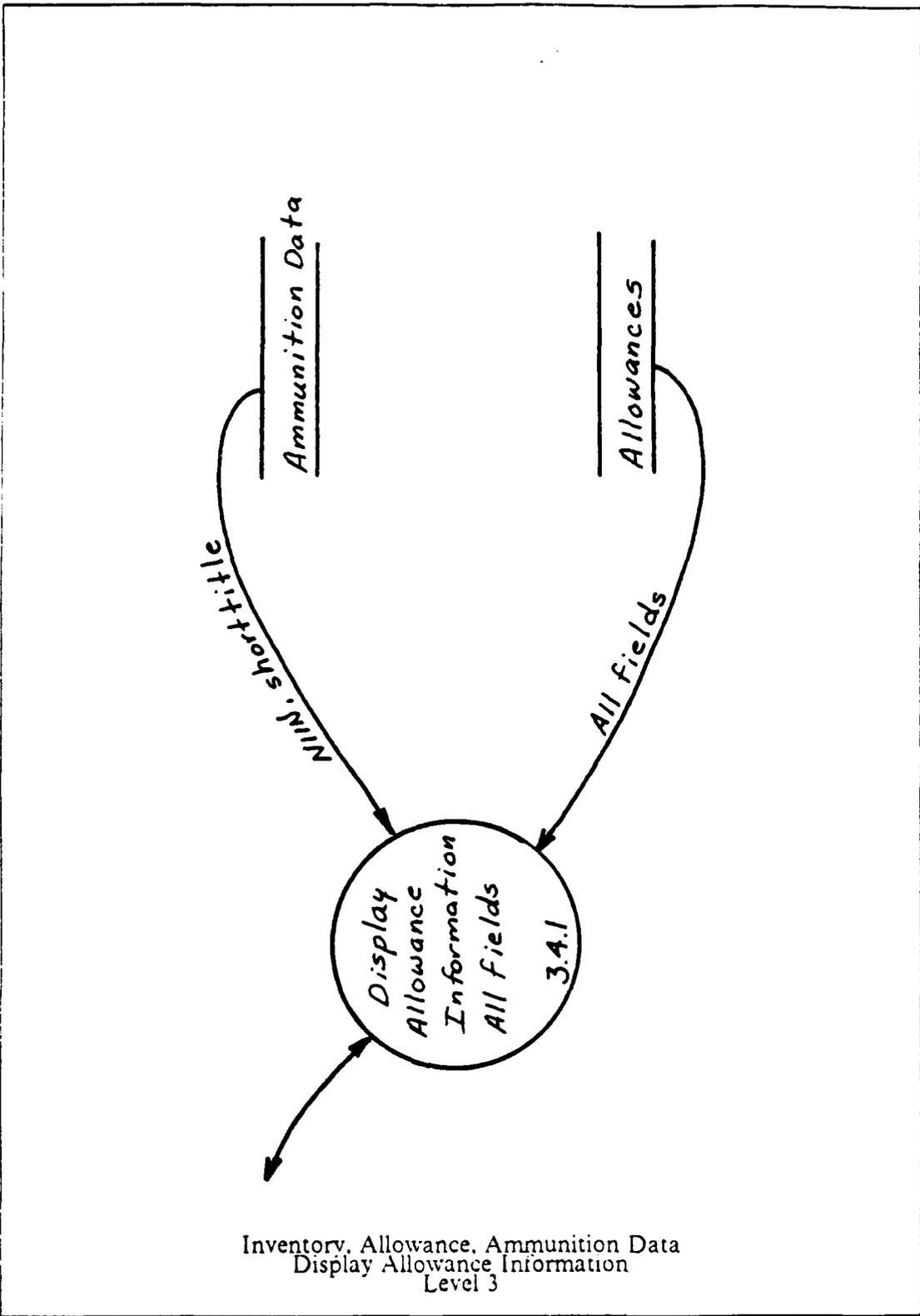


General Information: Documentation
Level 2

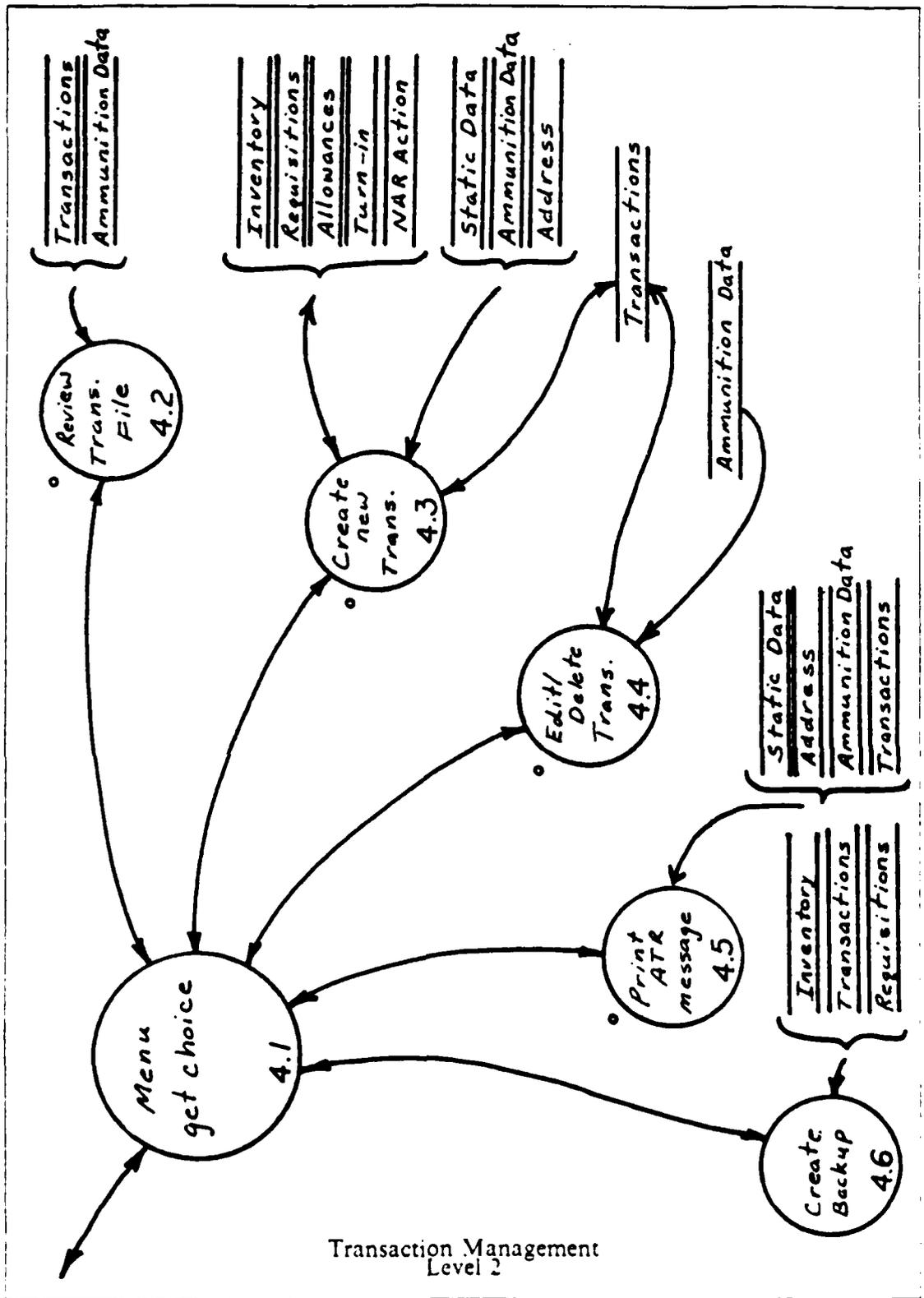


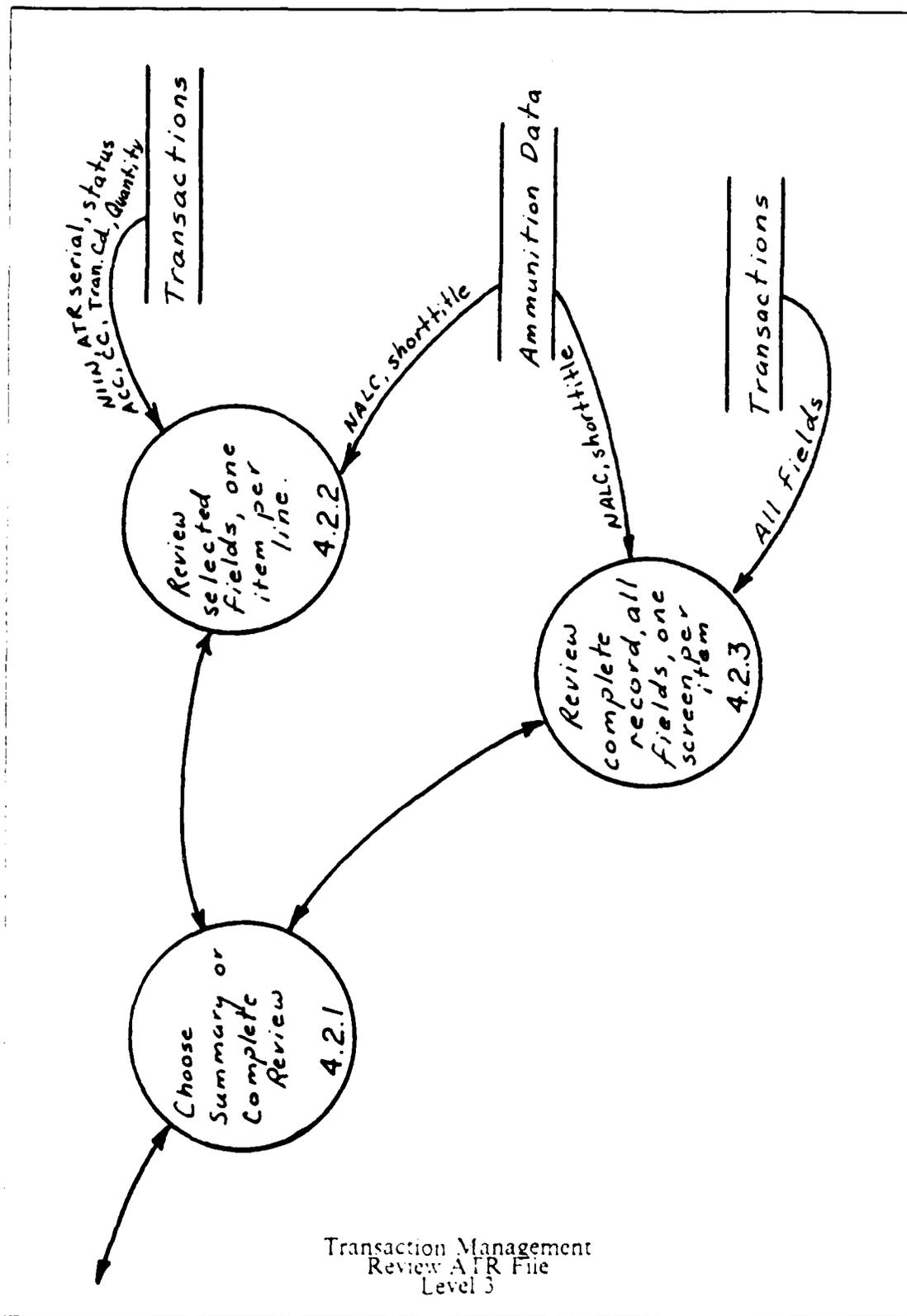


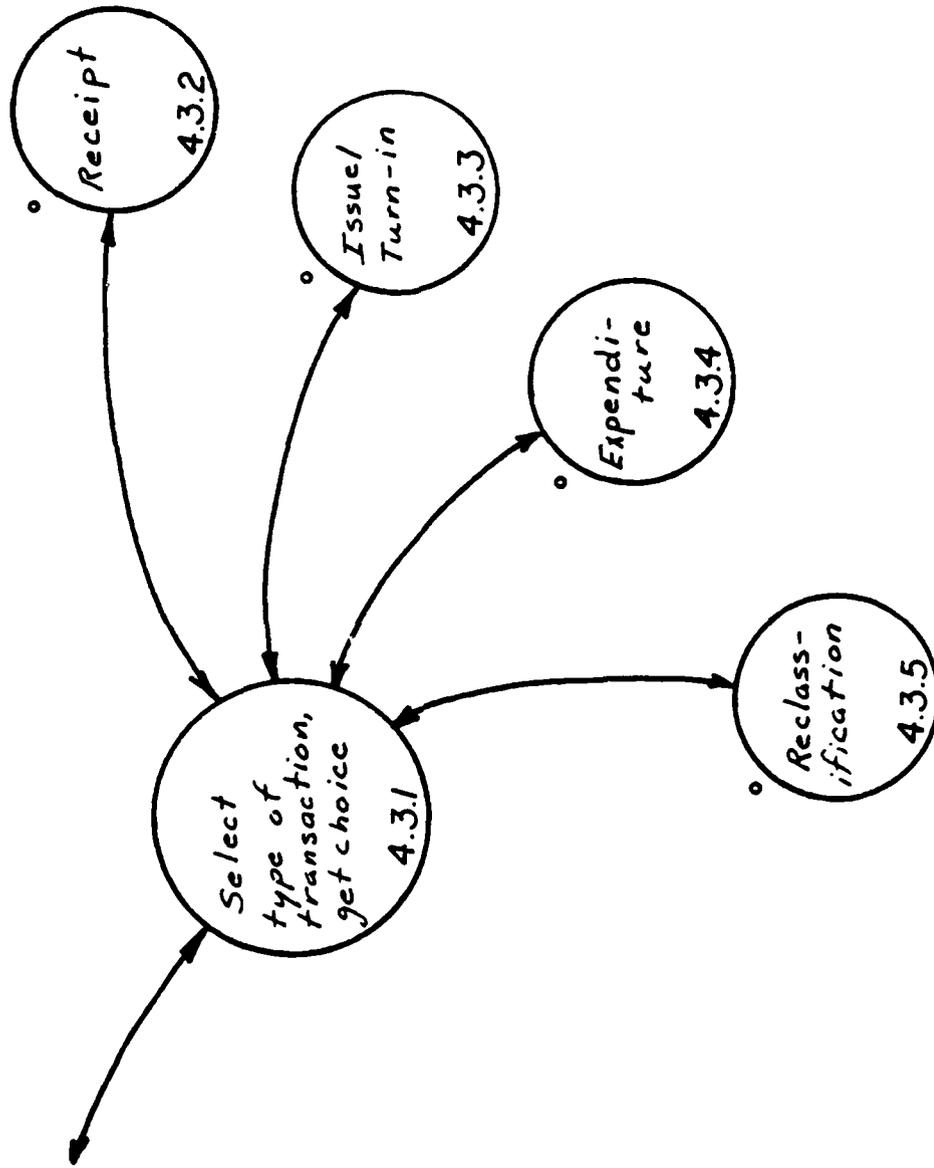




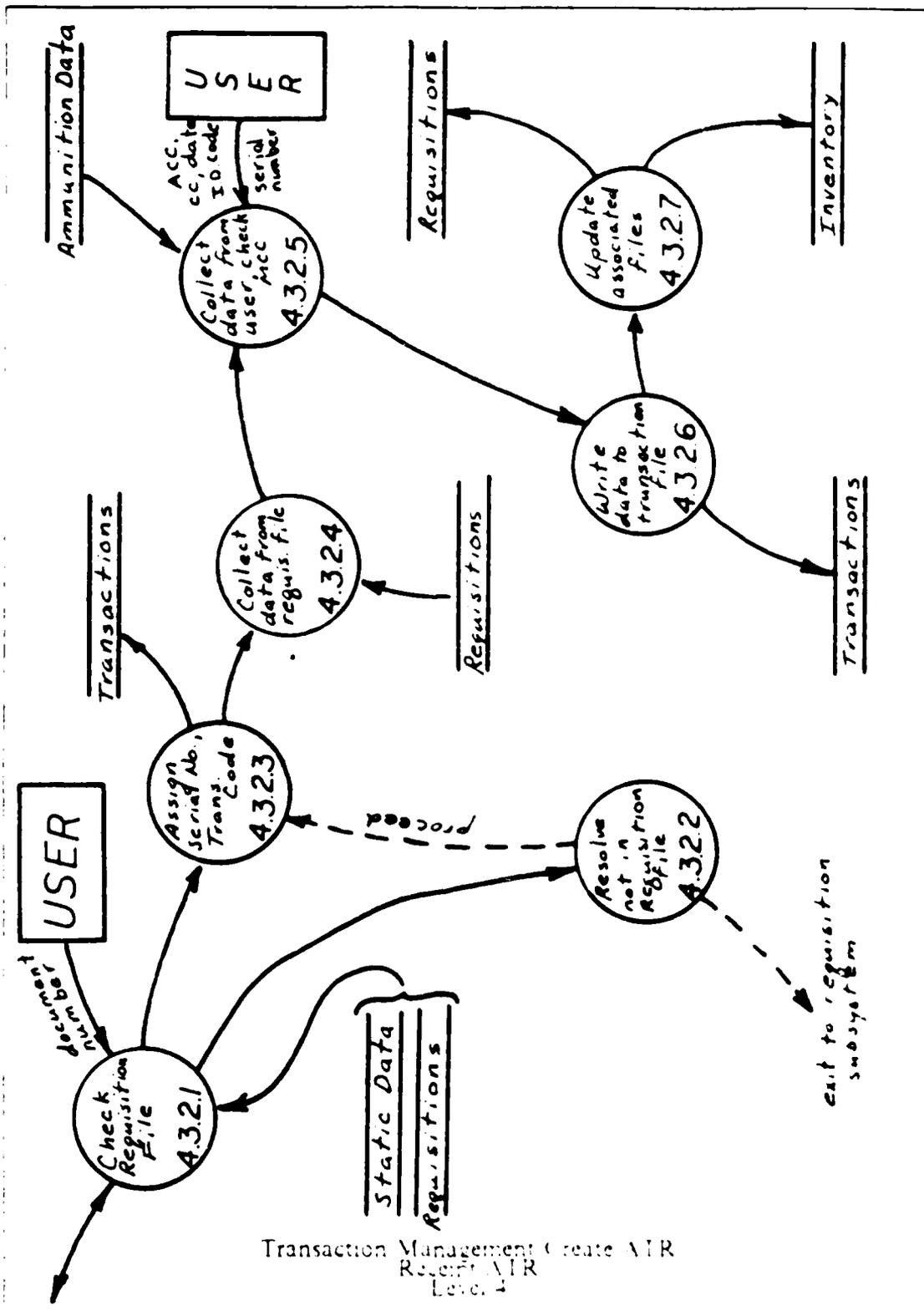
Inventory, Allowance, Ammunition Data
 Display Allowance Information
 Level 3

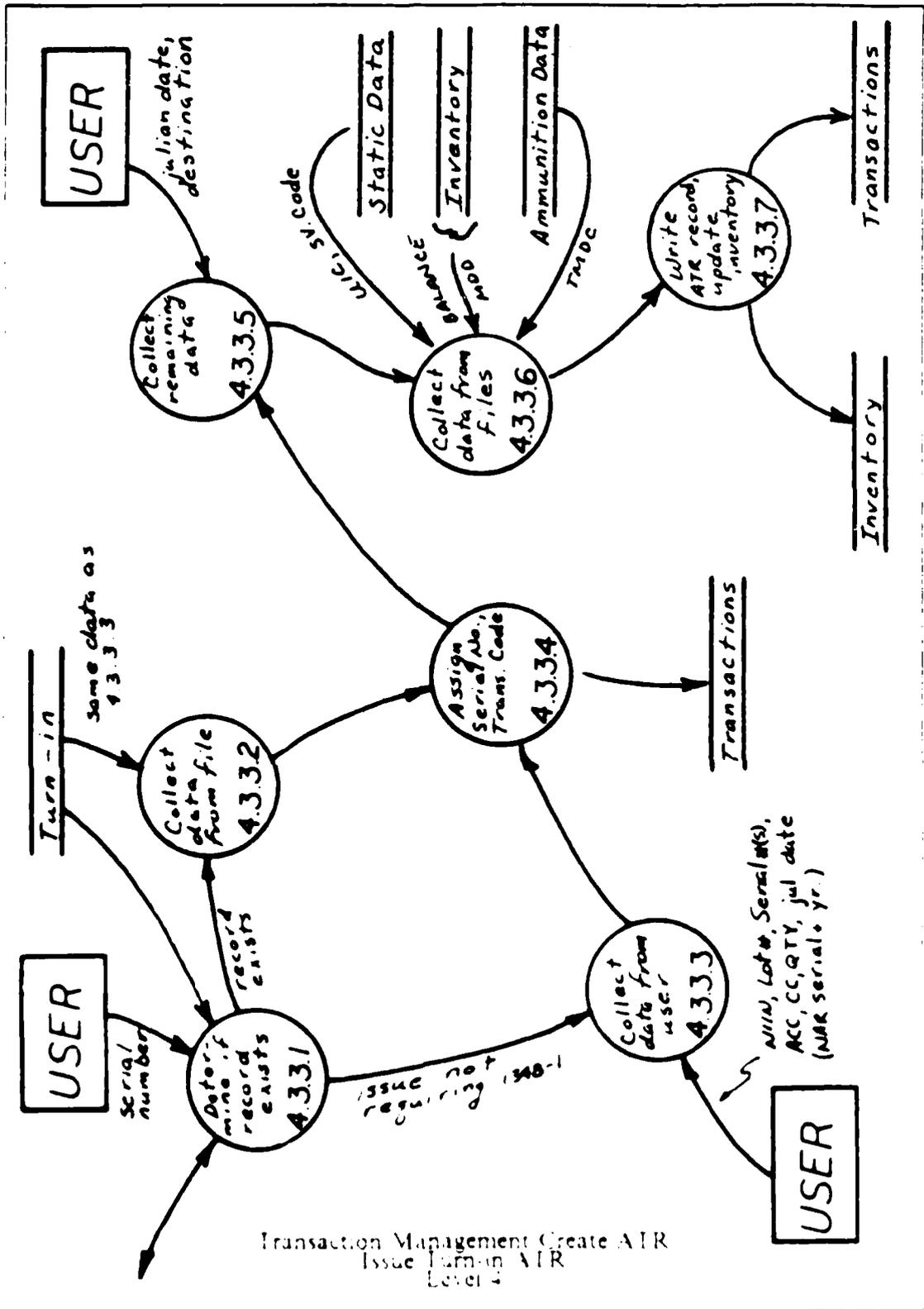


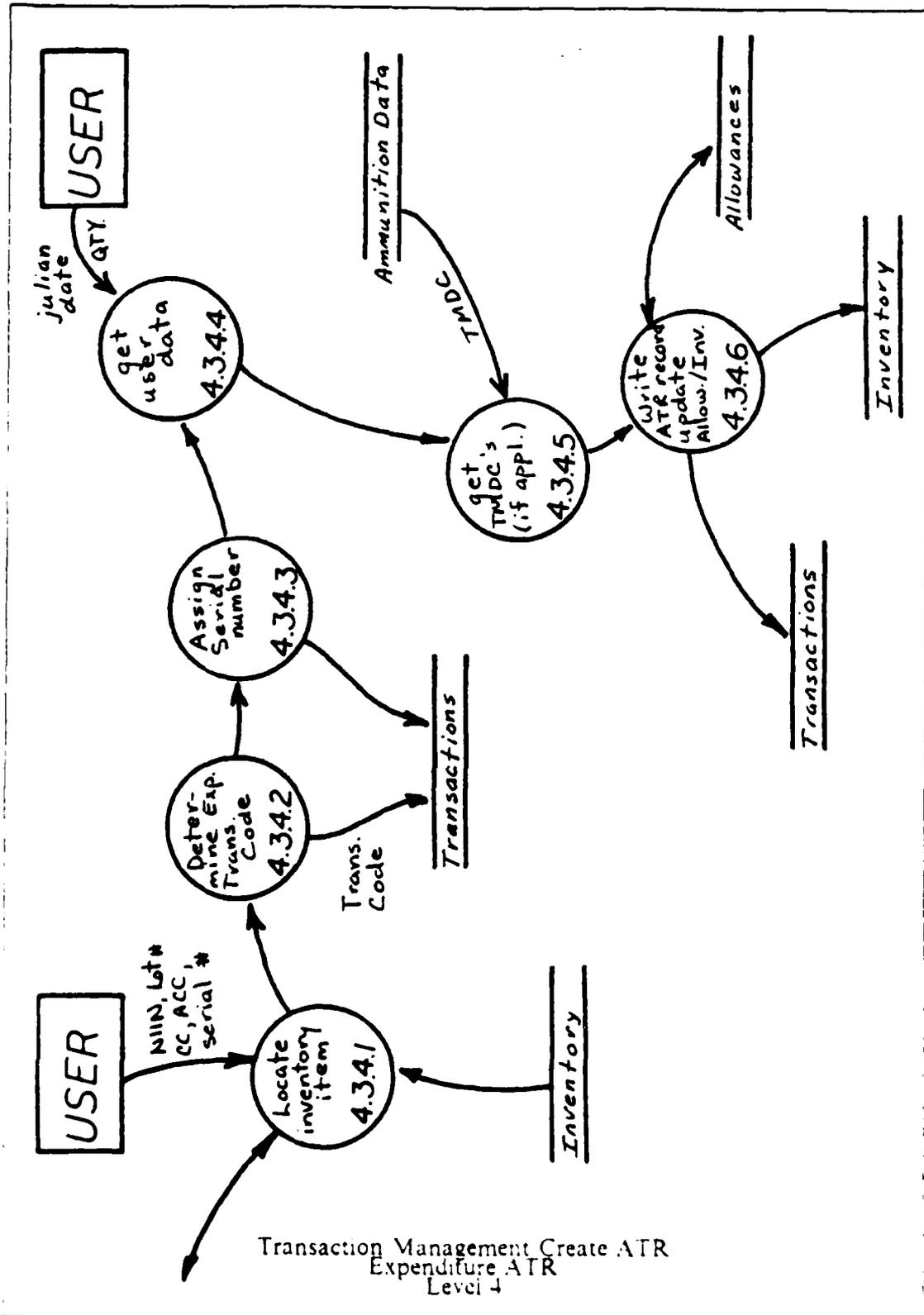


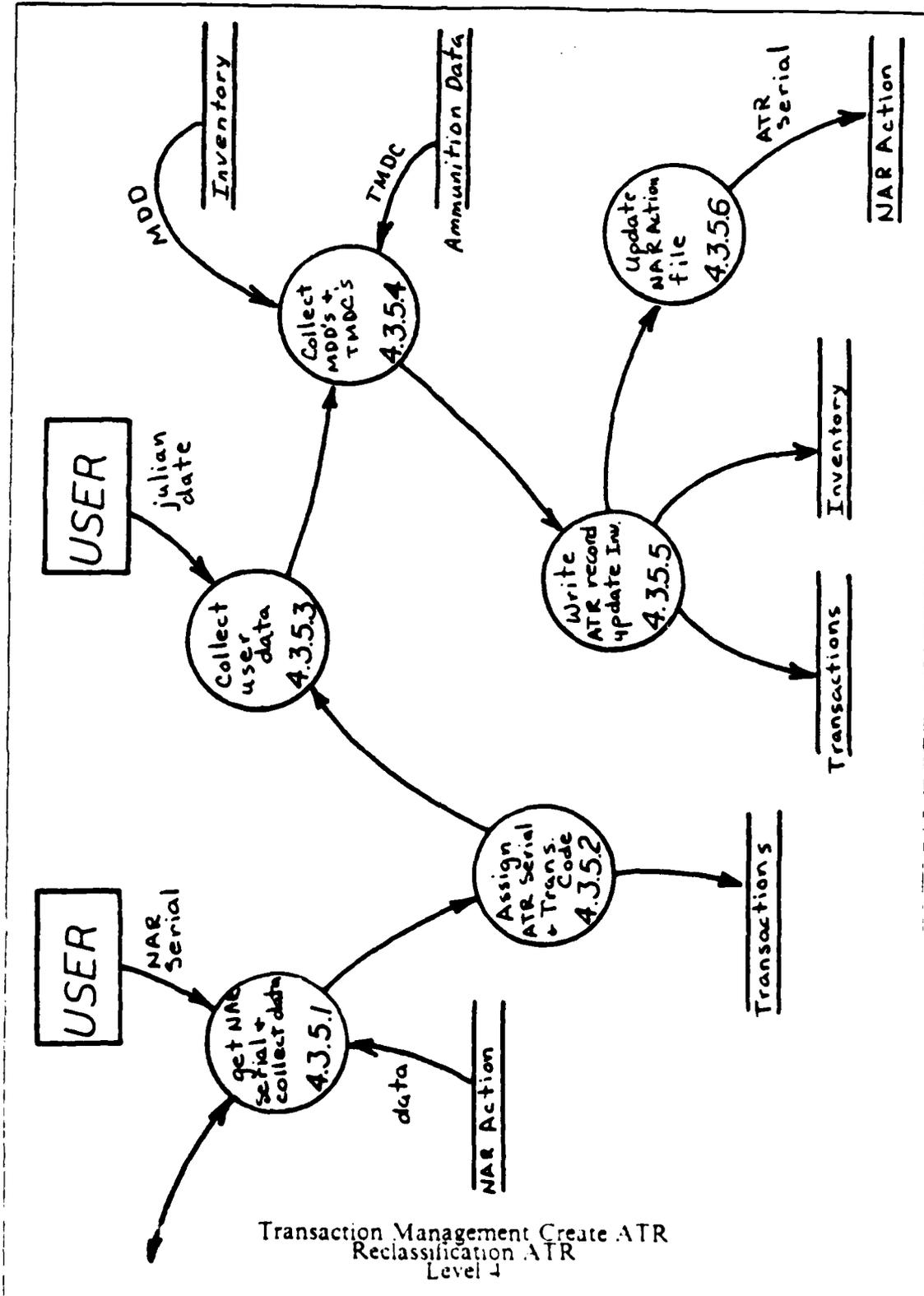


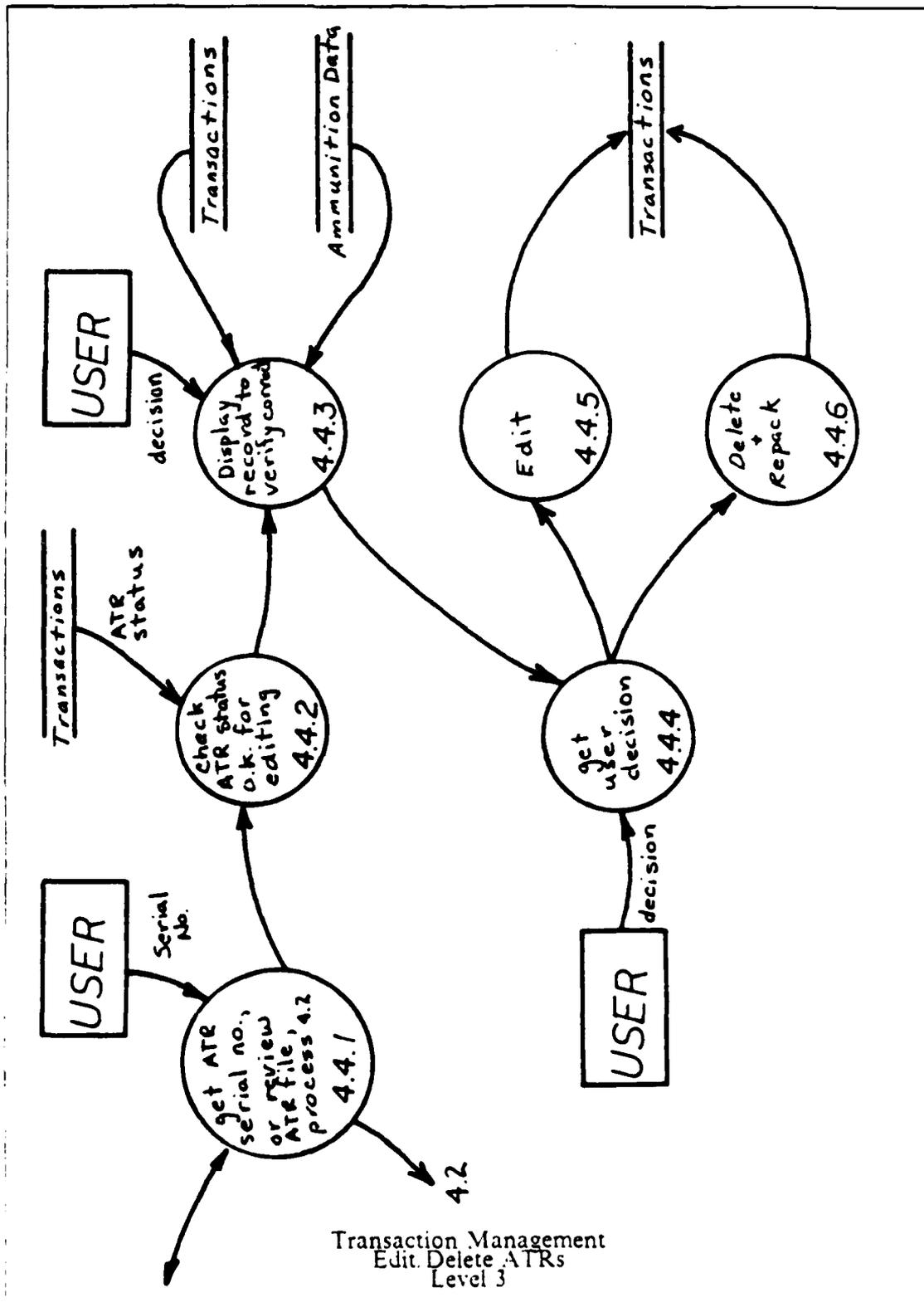
Transaction Management
Create New ATR
Level 3

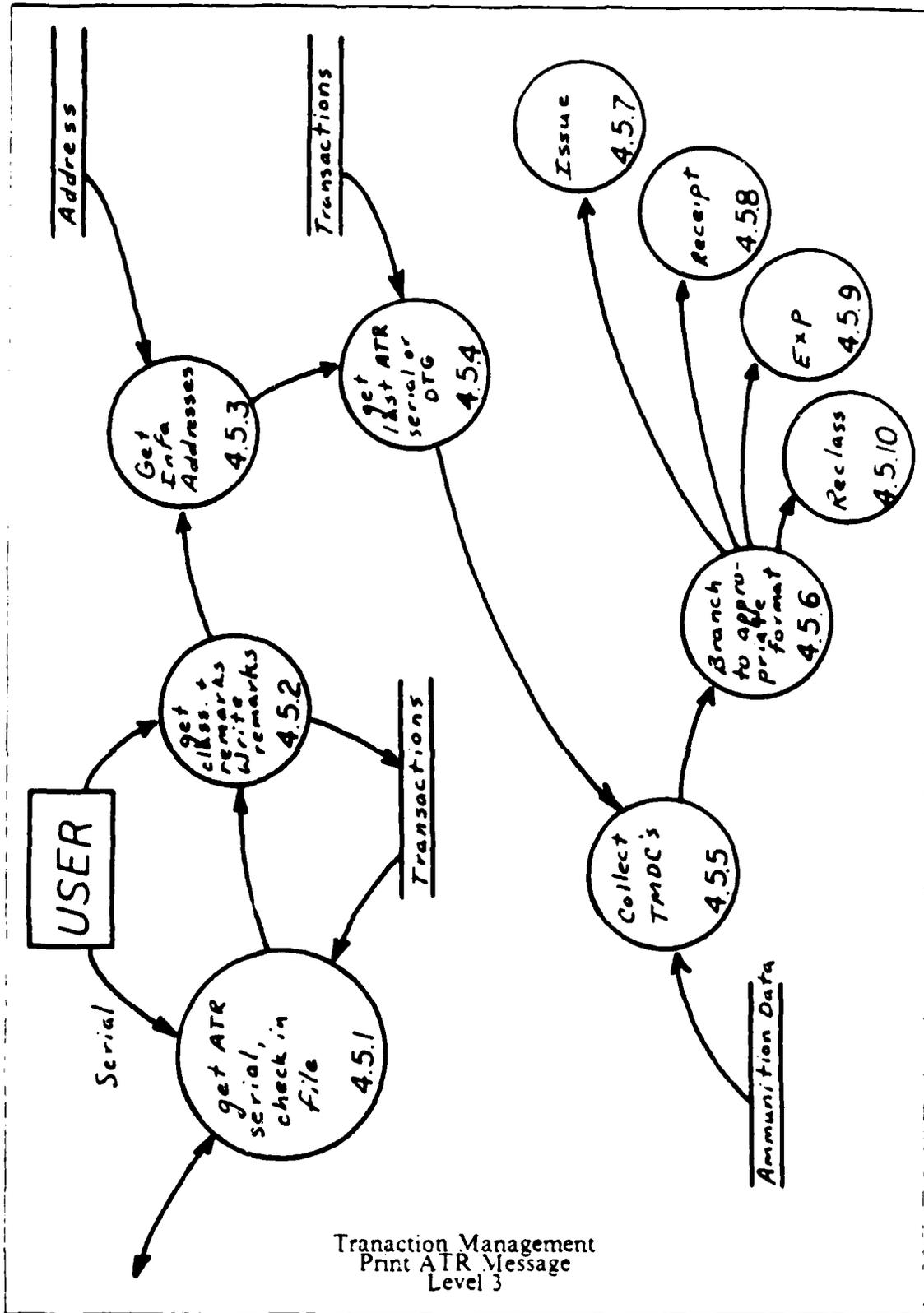


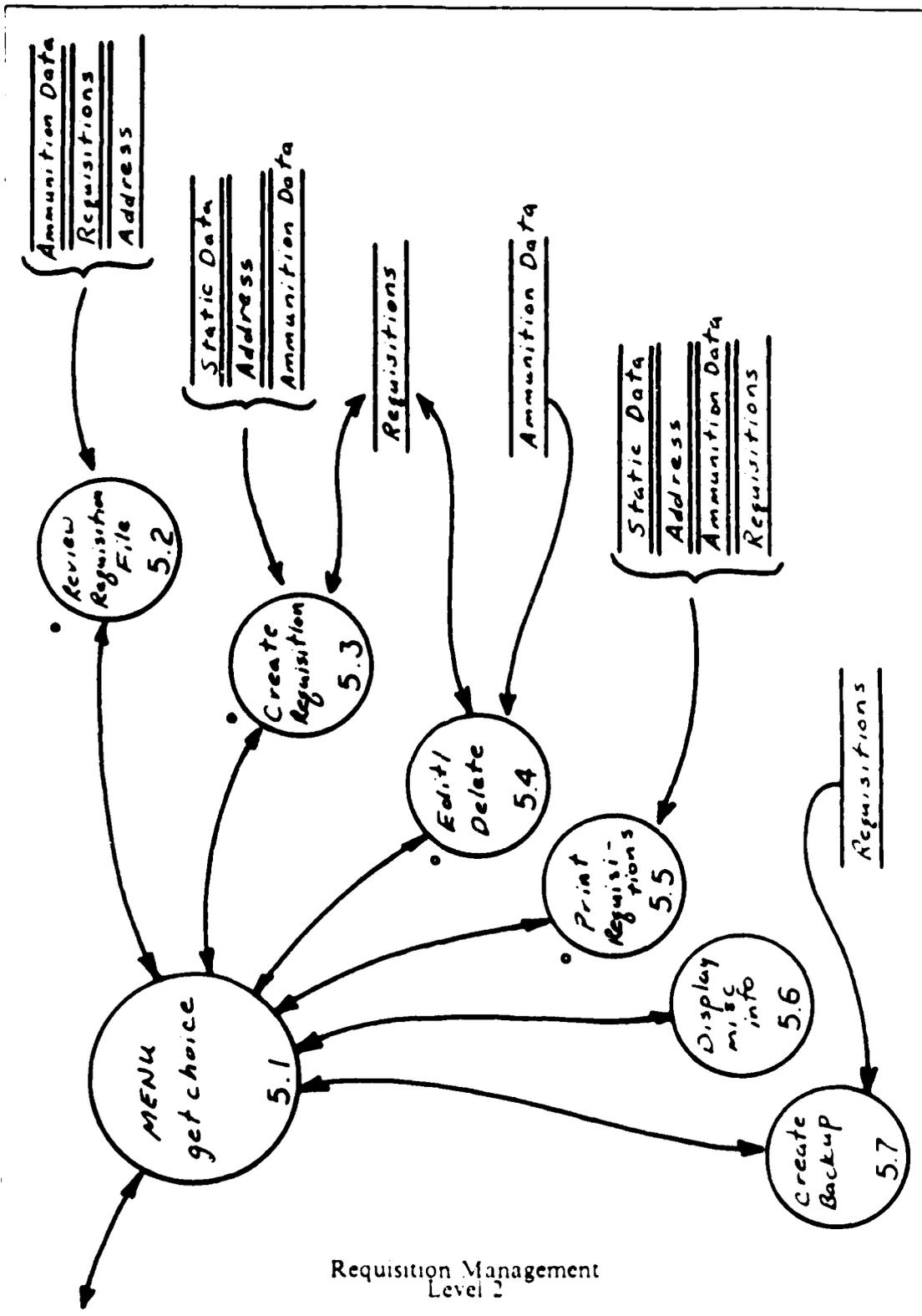




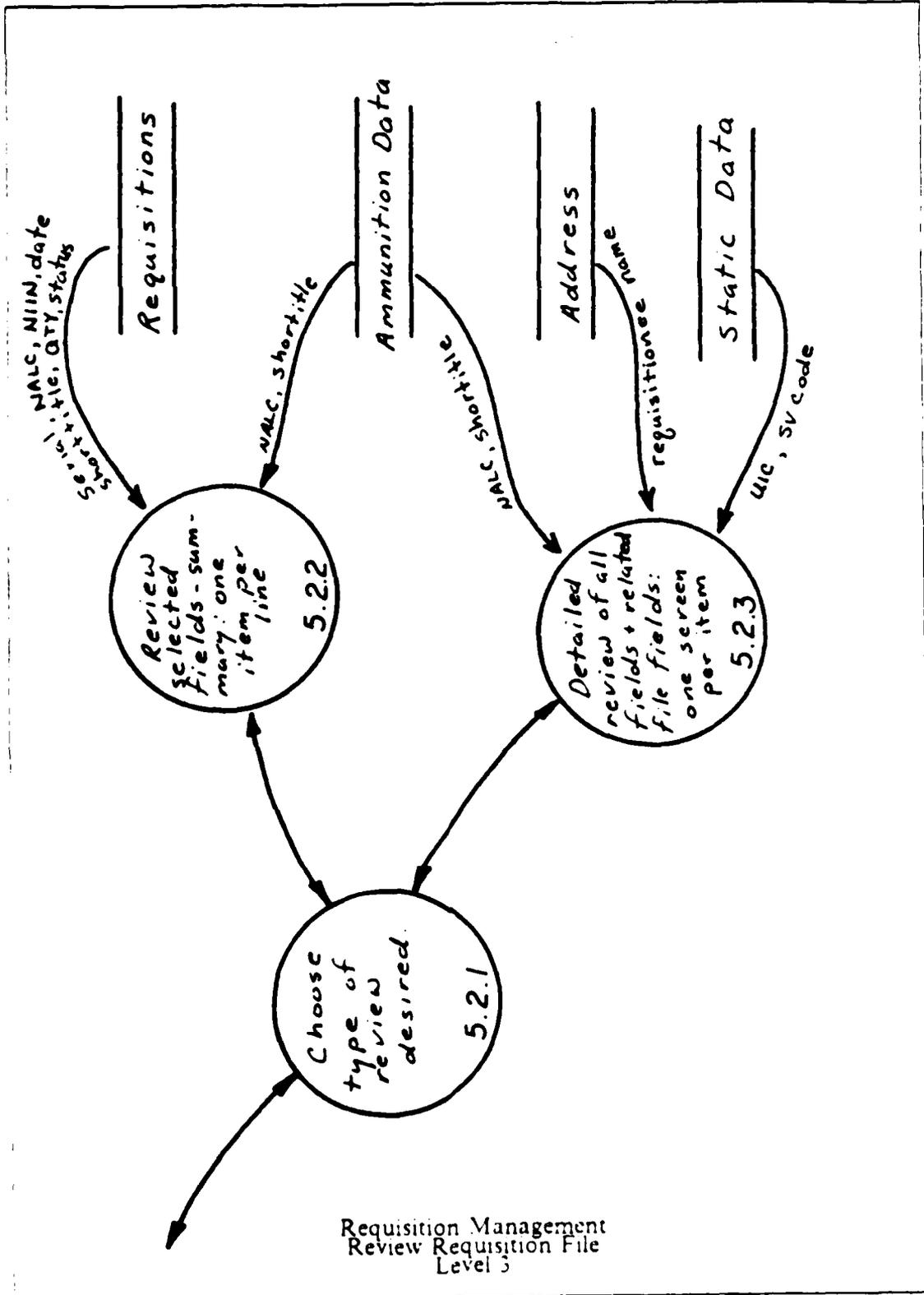


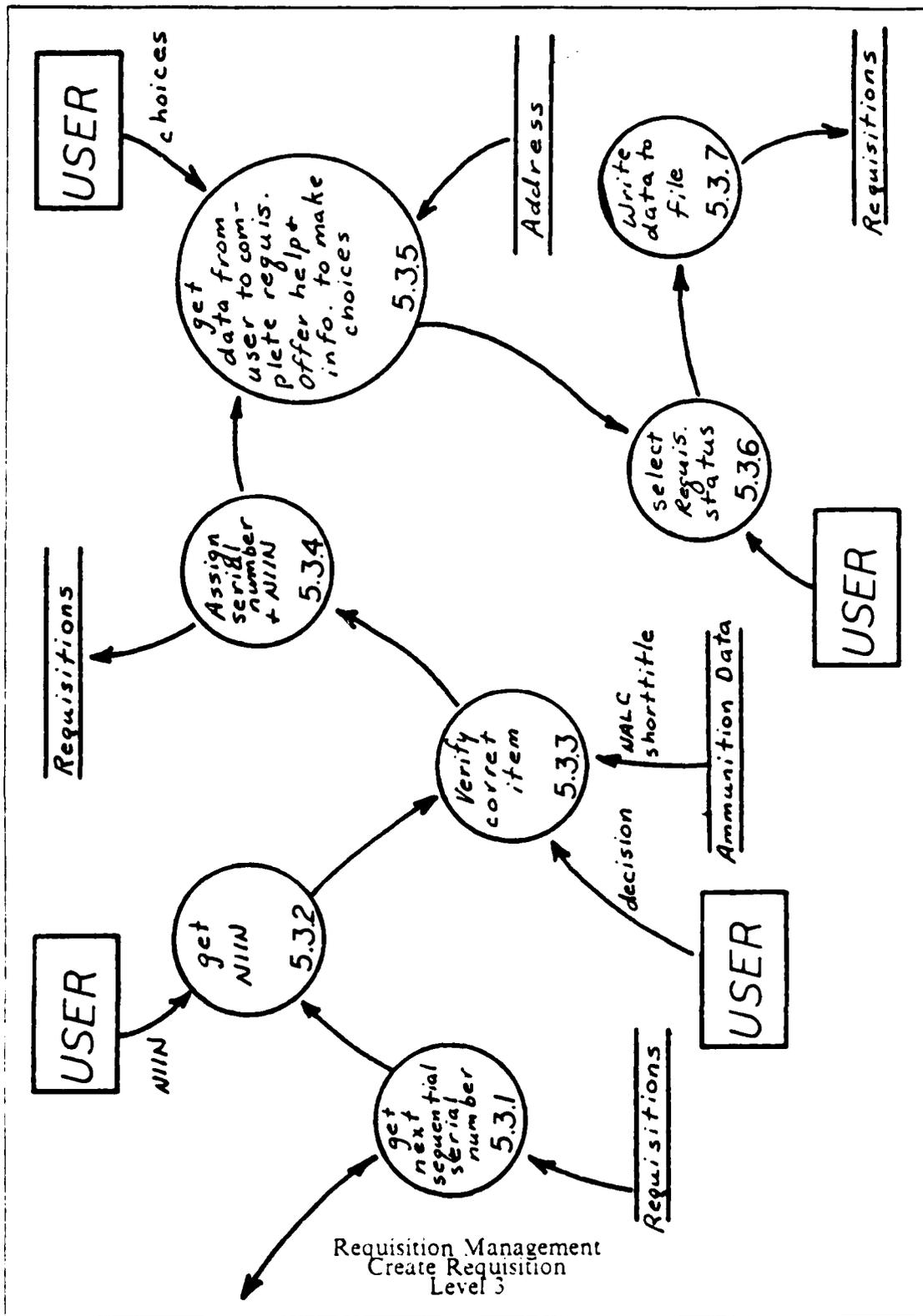


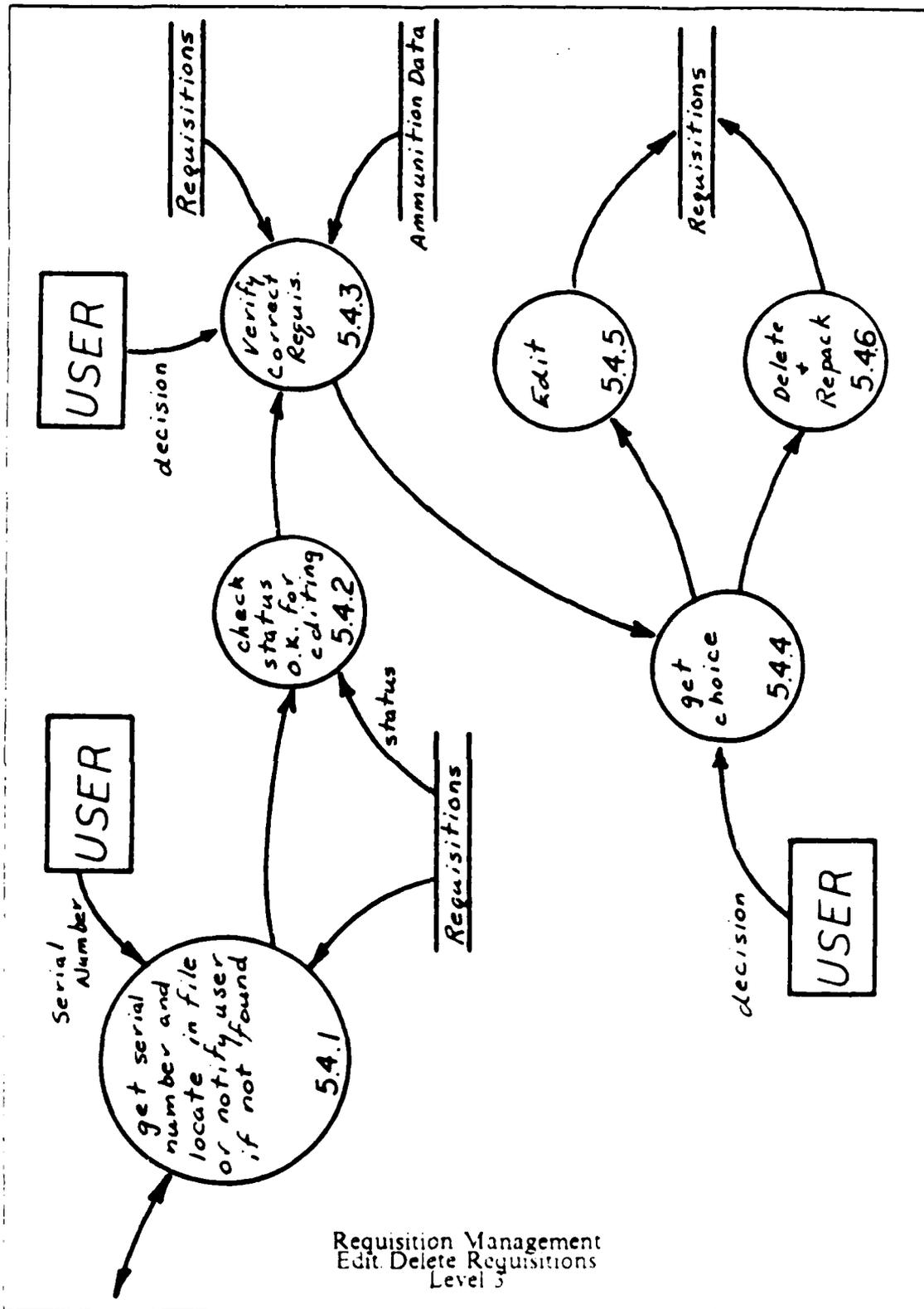


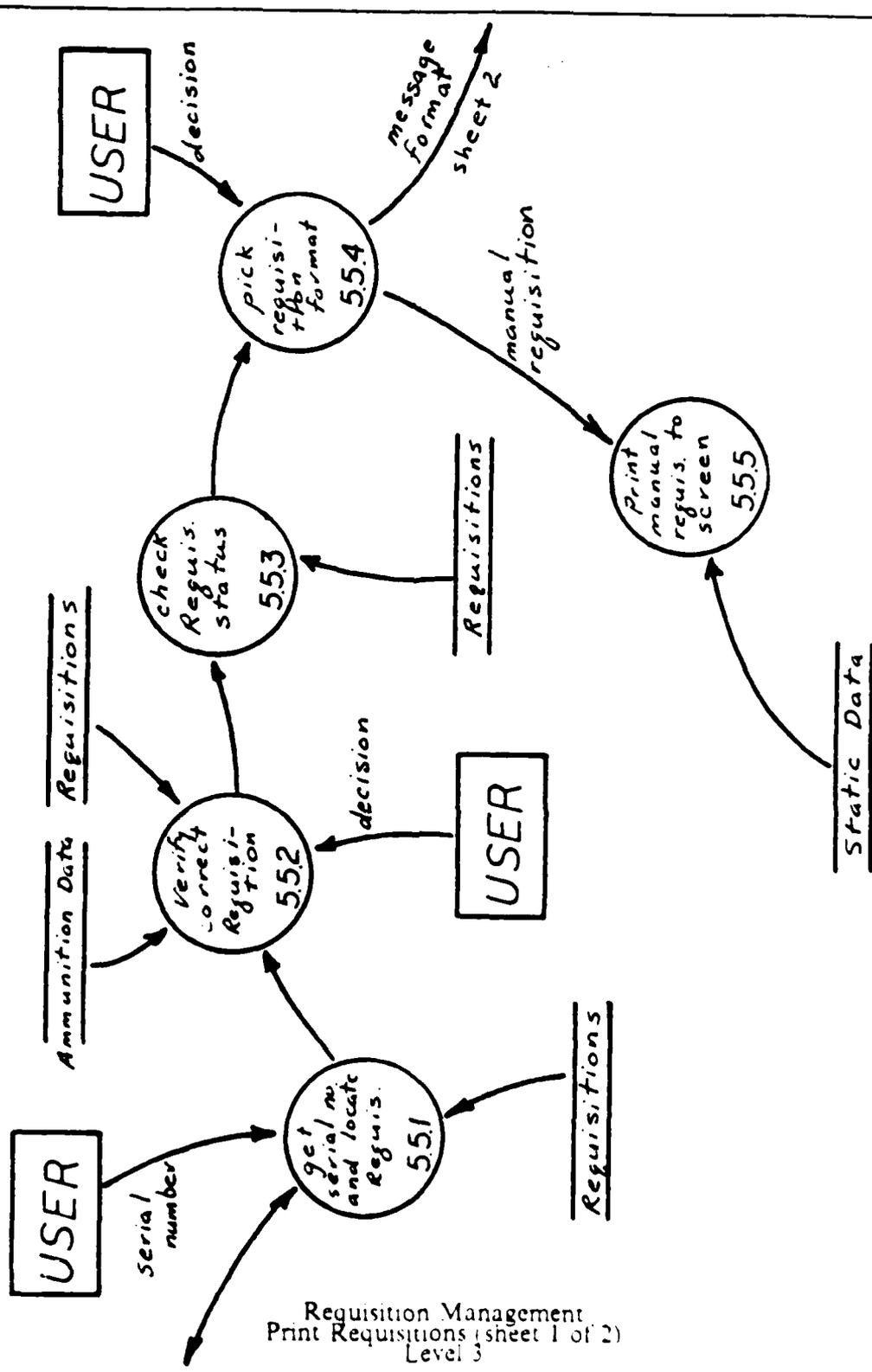


Requisition Management
Level 2

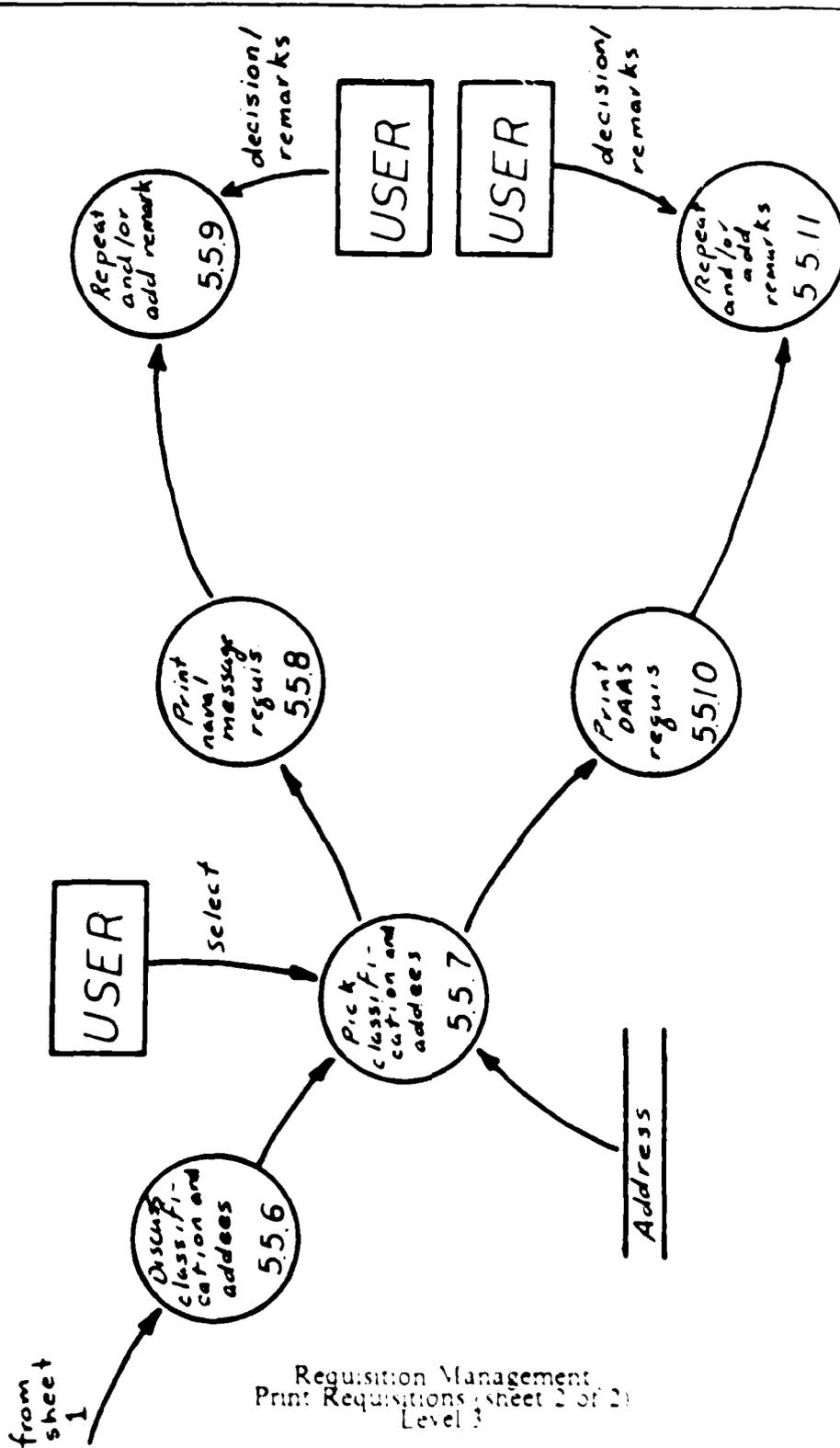






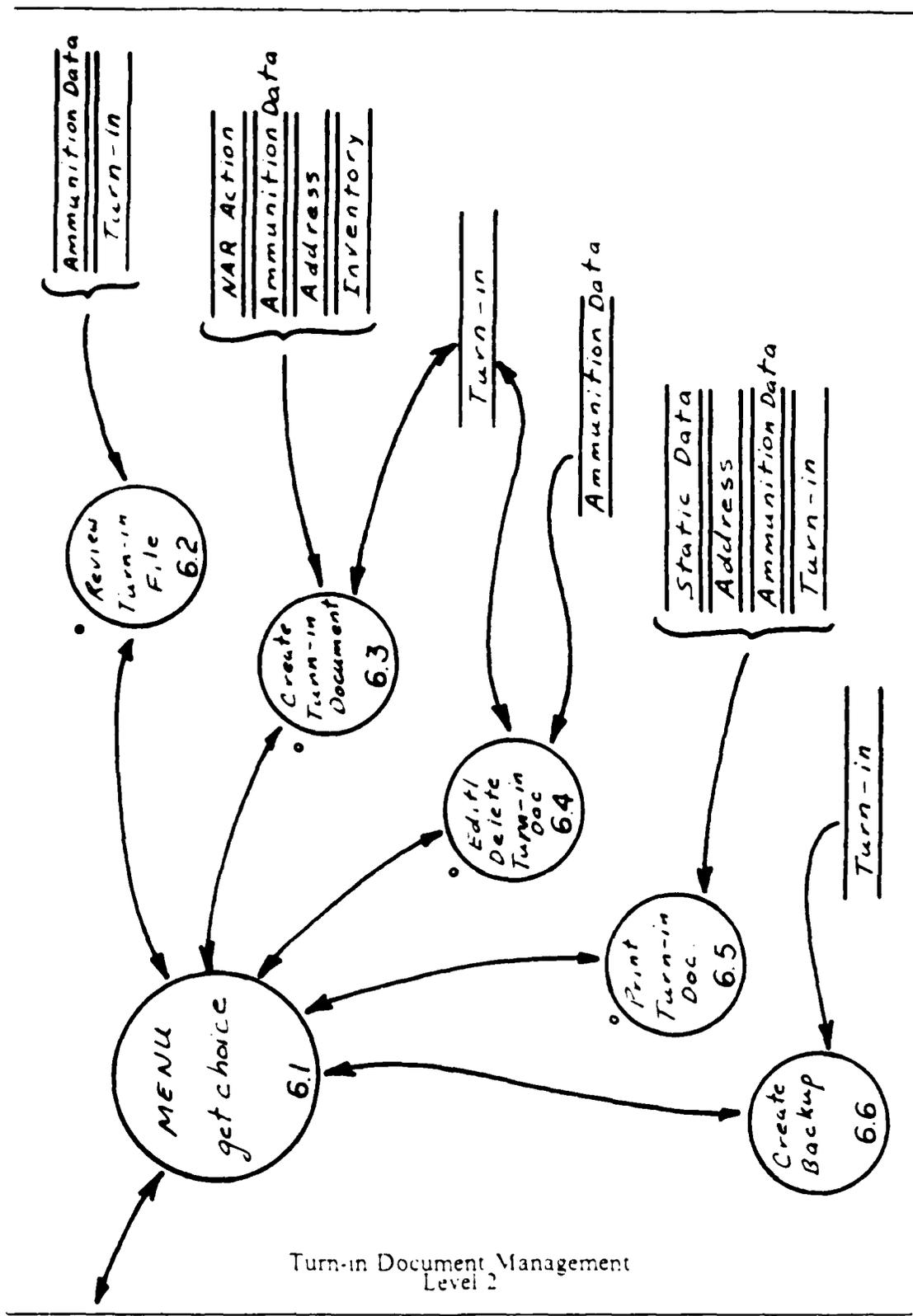


Requisition Management
 Print Requisitions (sheet 1 of 2)
 Level 3

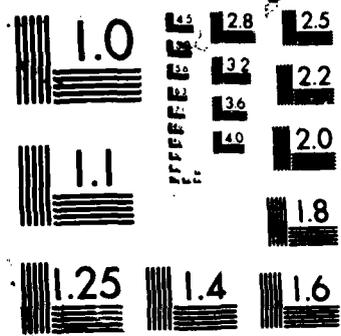


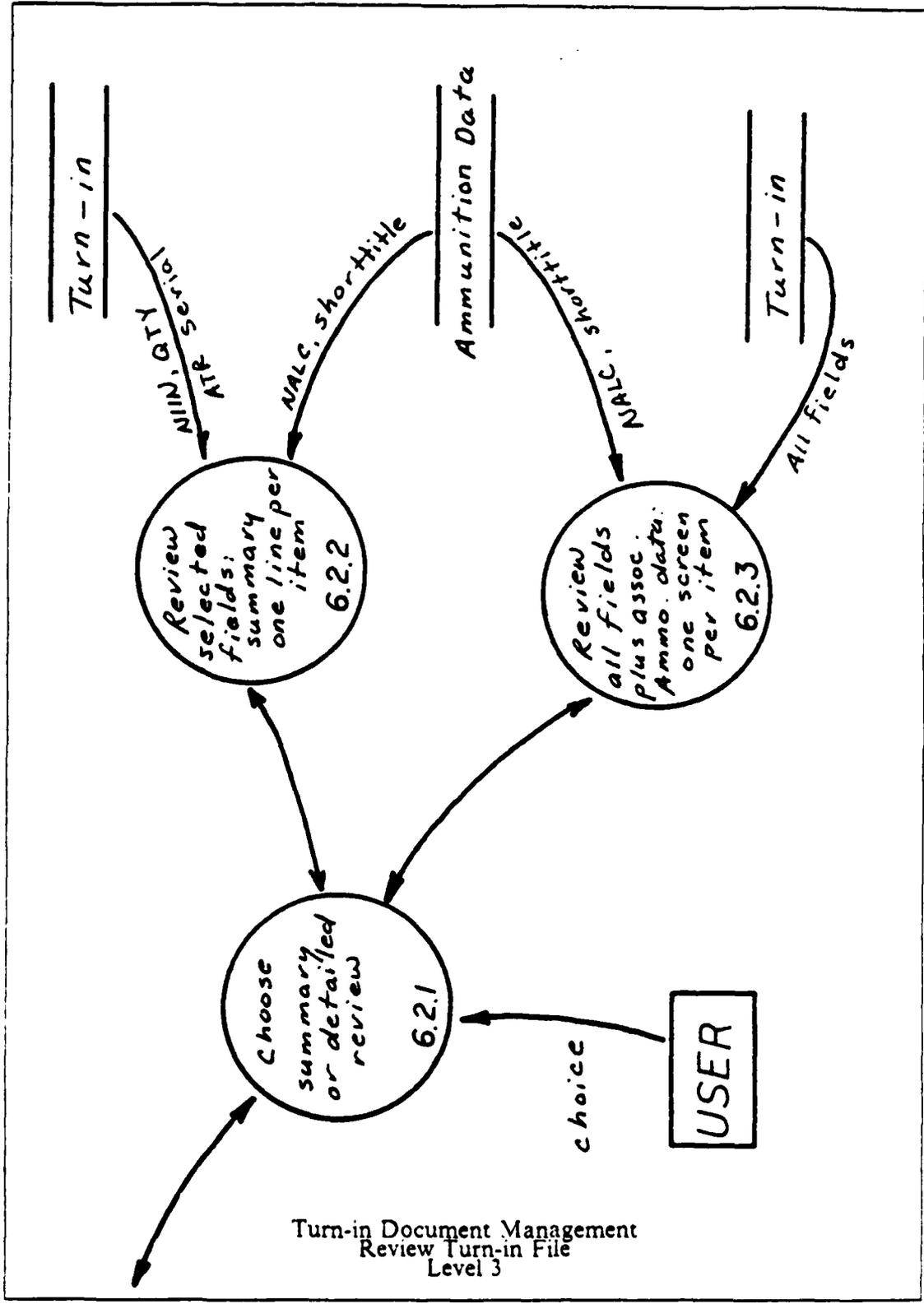
from sheet 1

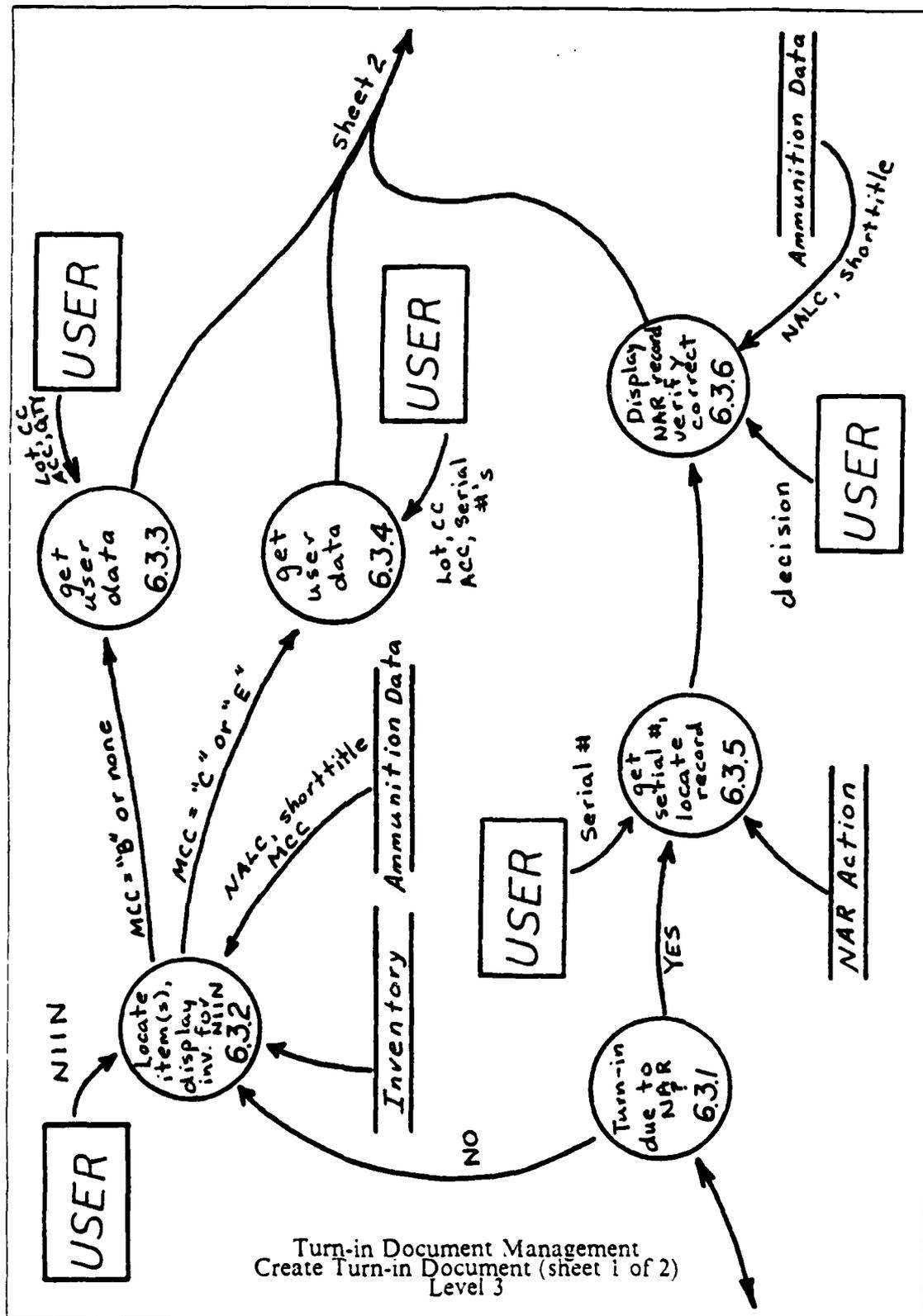
Requisition Management
 Print Requisitions (sheet 2 of 2)
 Level 3

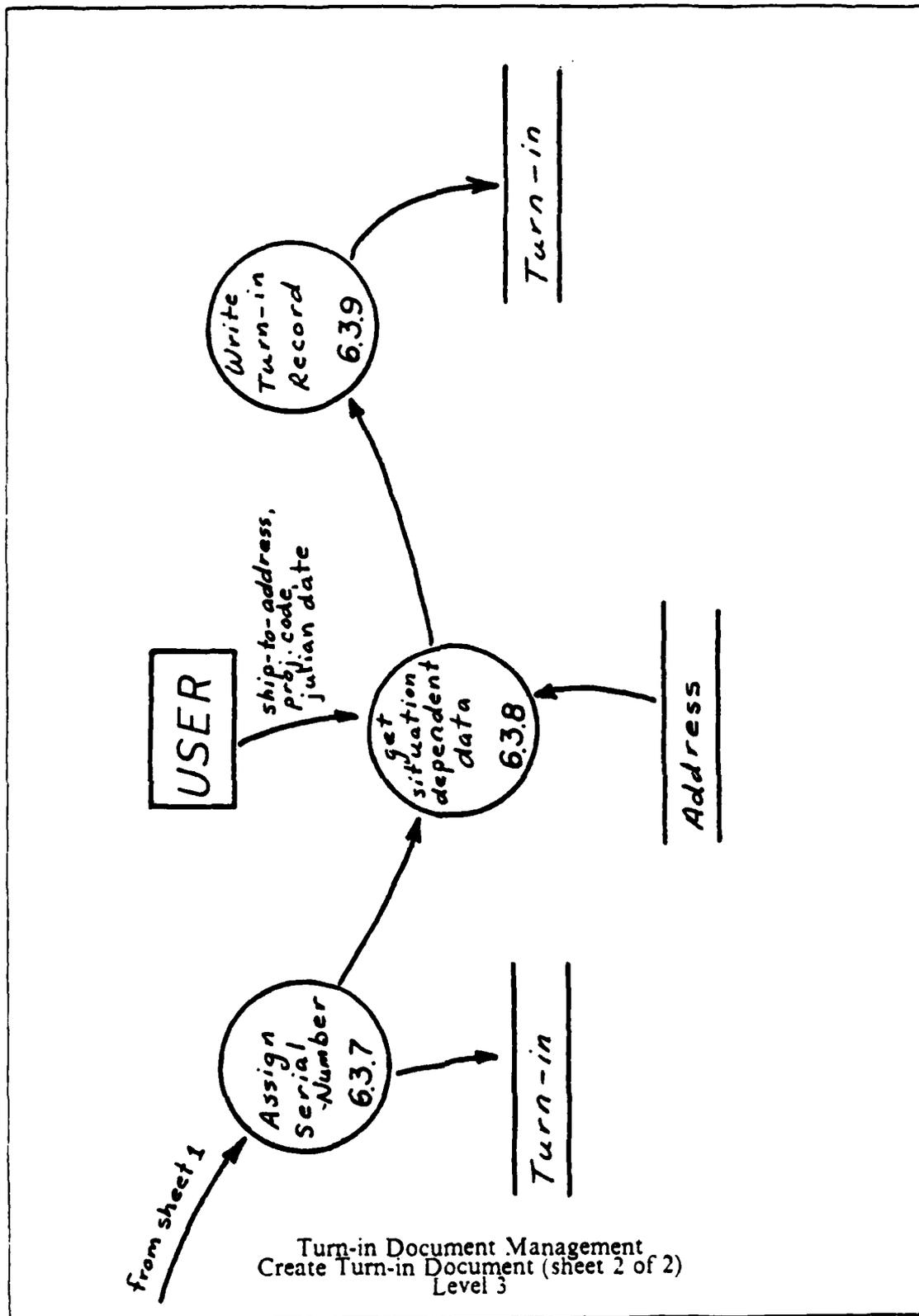


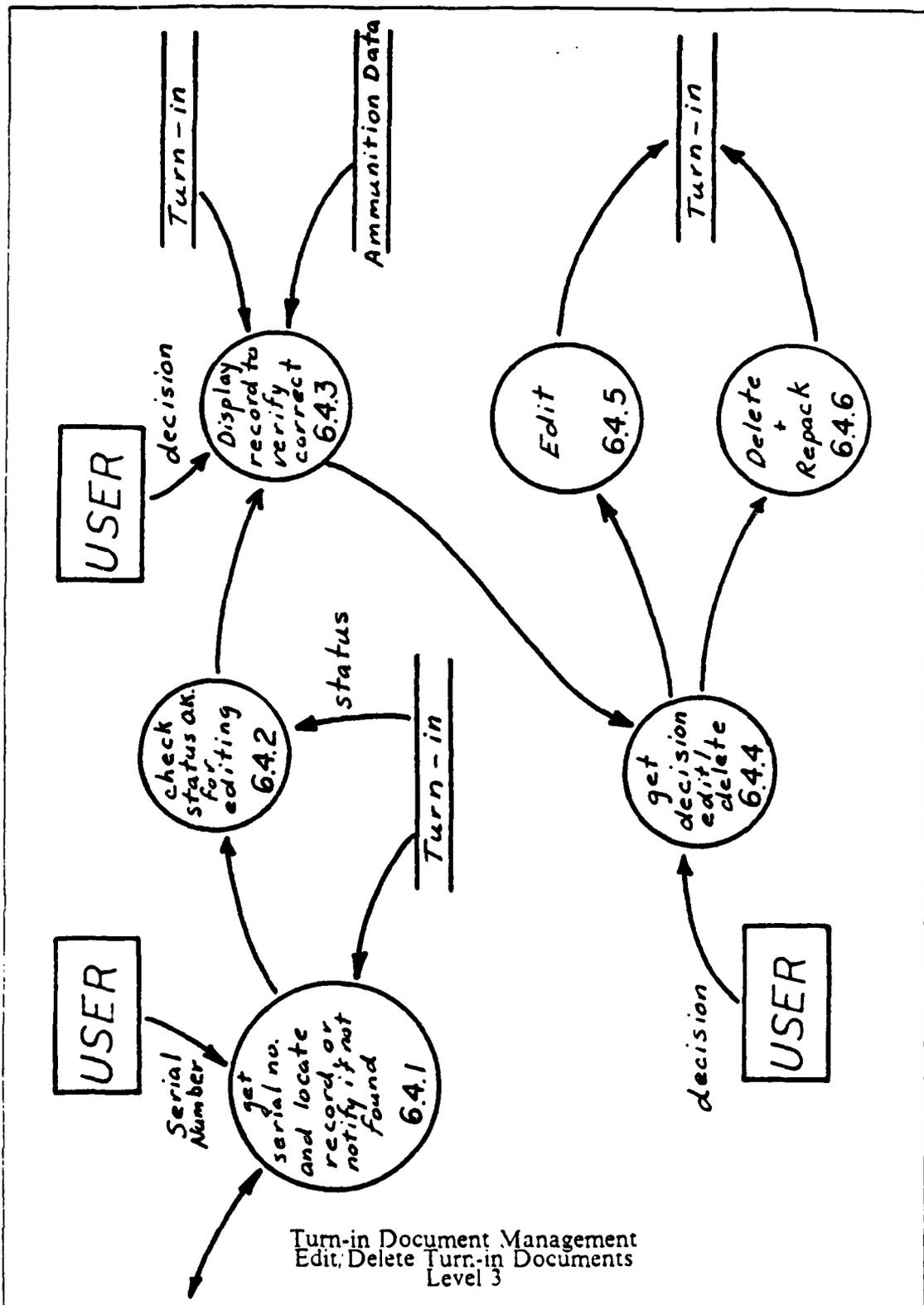
Turn-in Document Management
Level 2

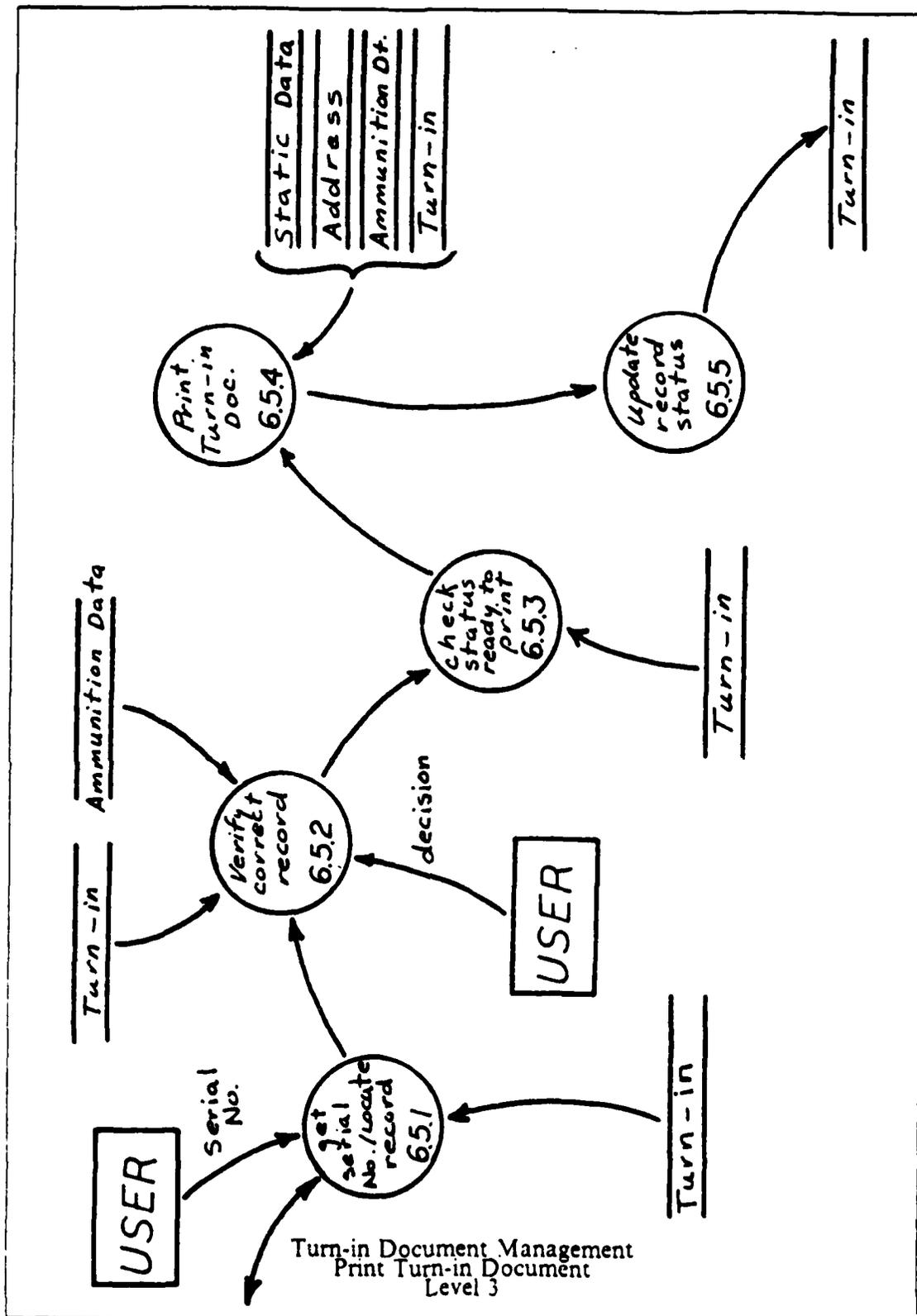


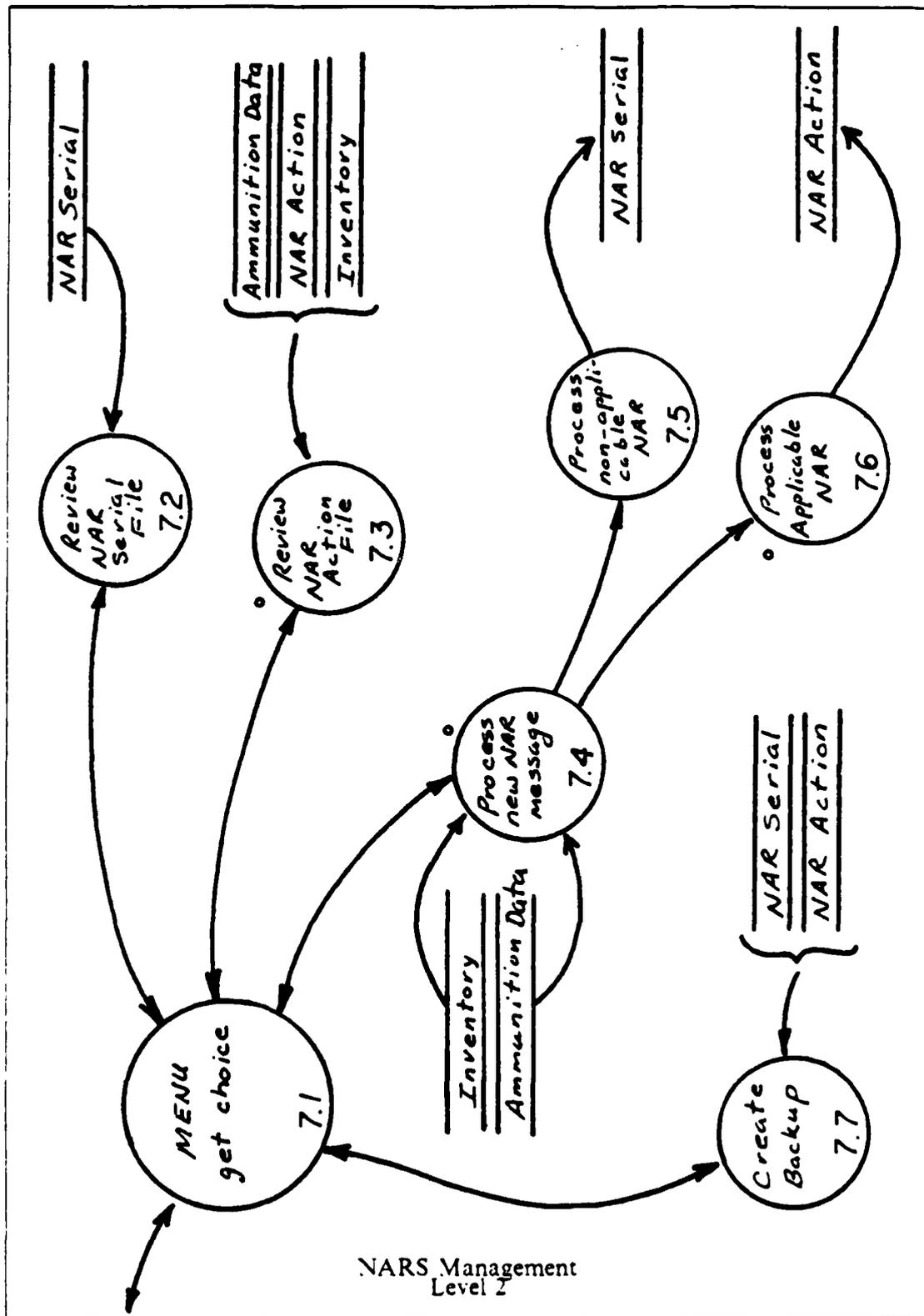




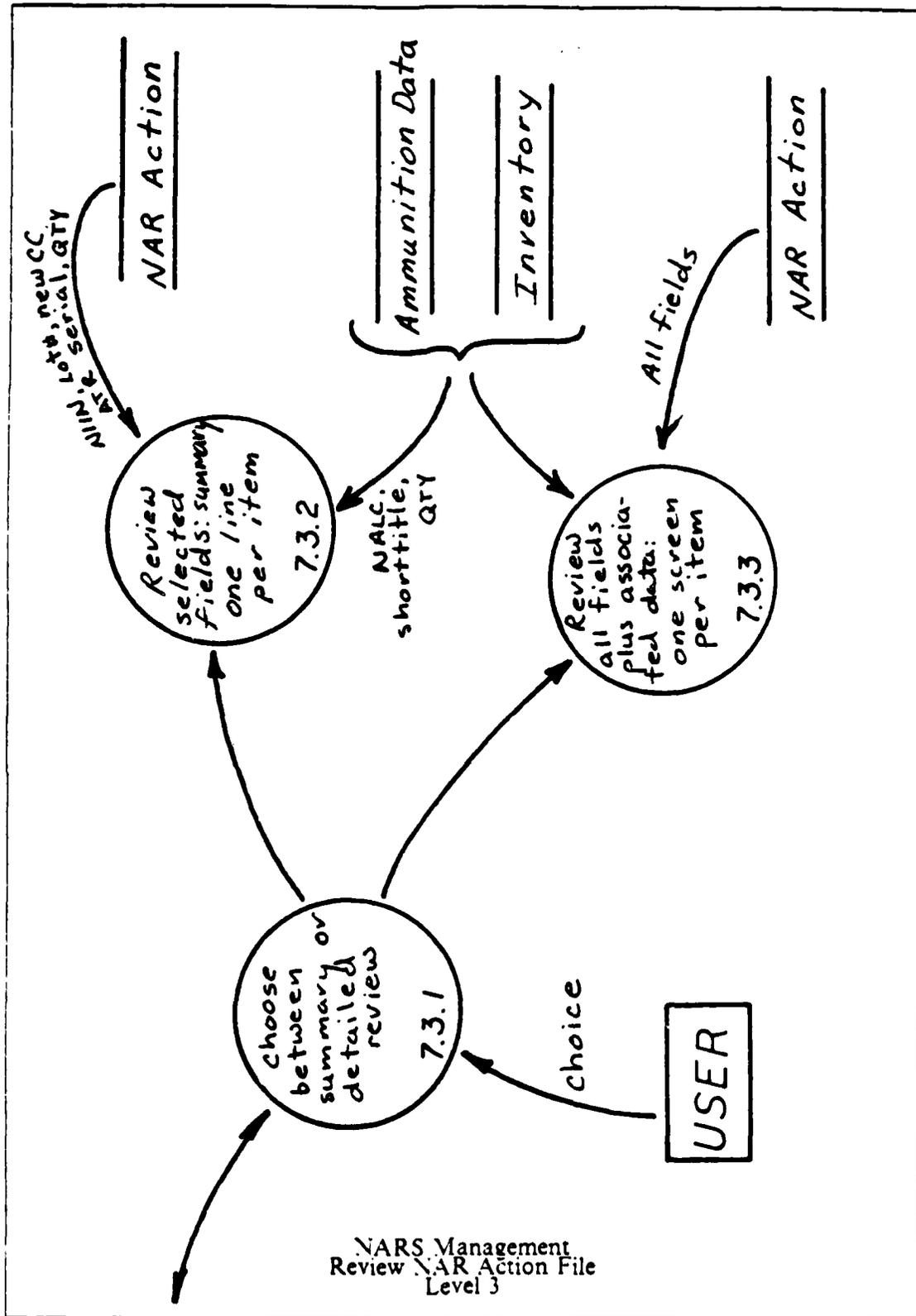


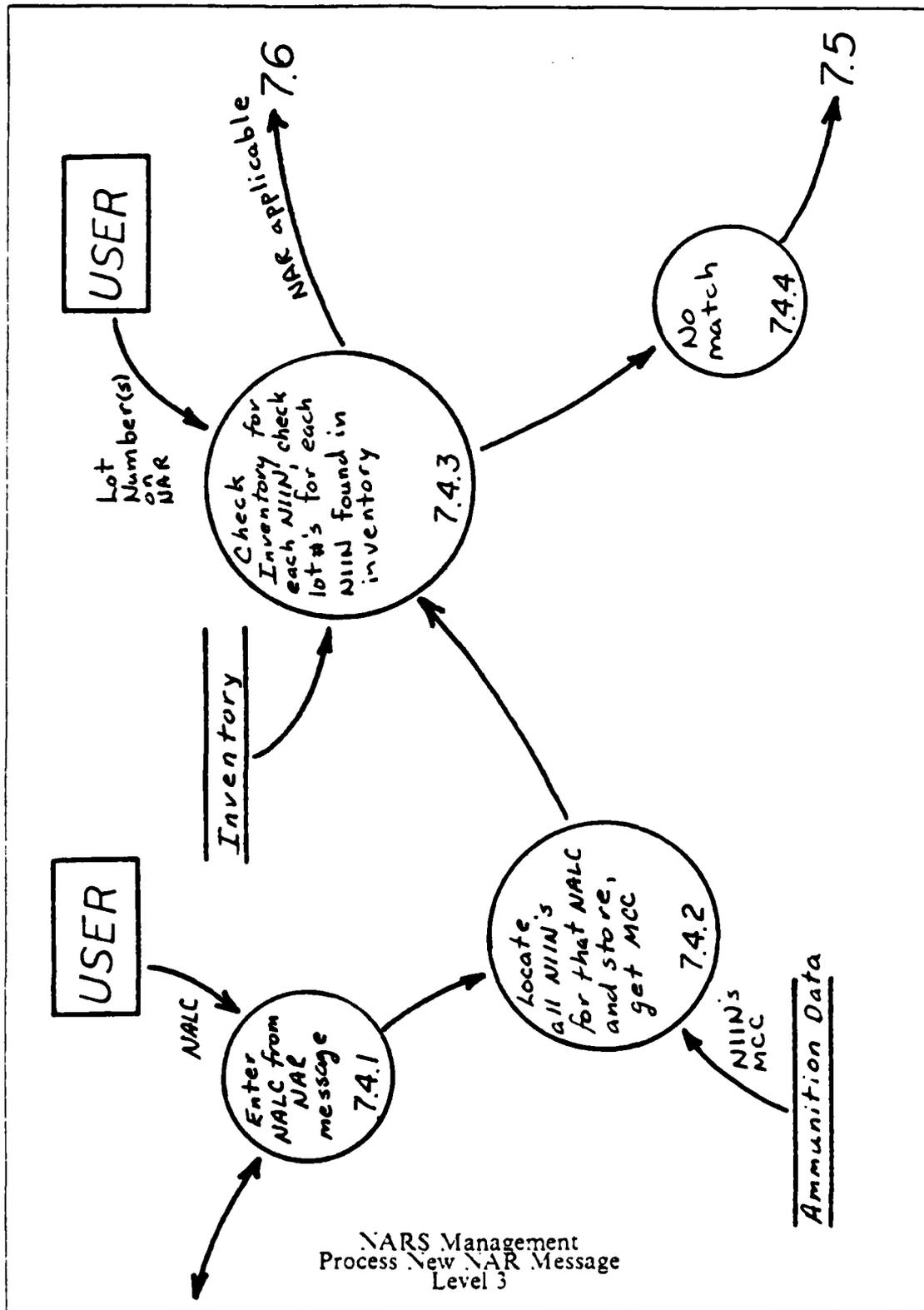


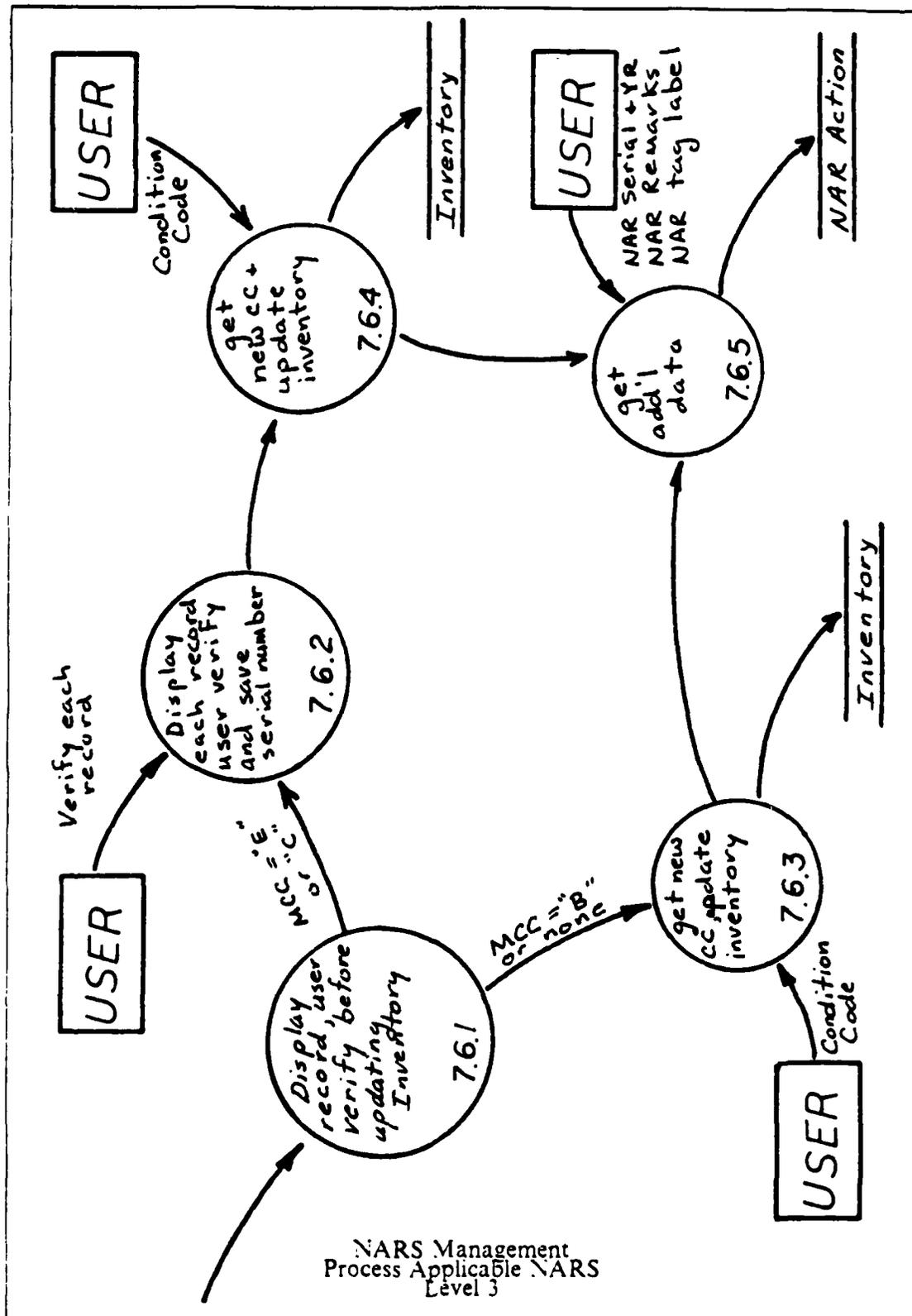




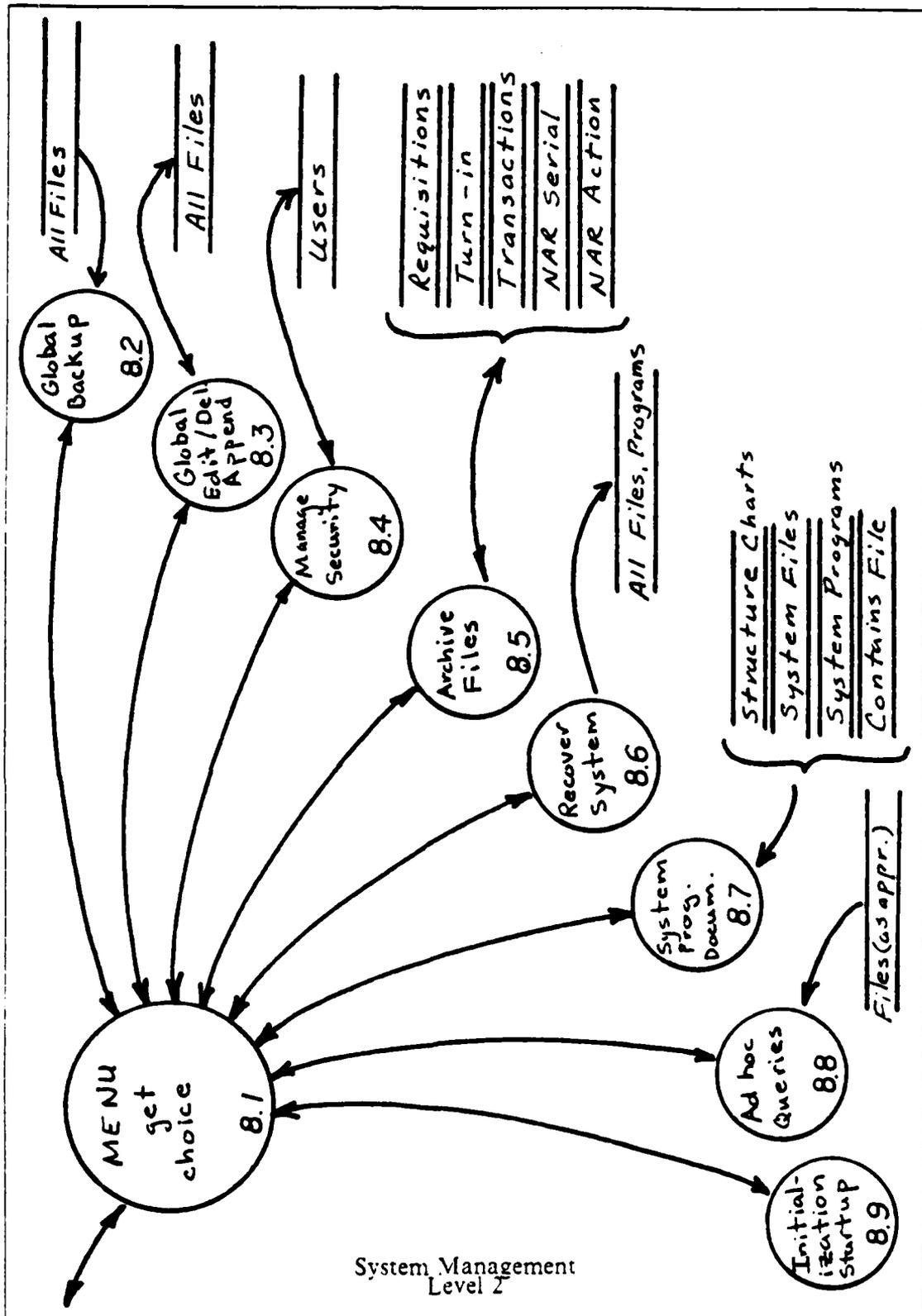
NARS Management Level 2

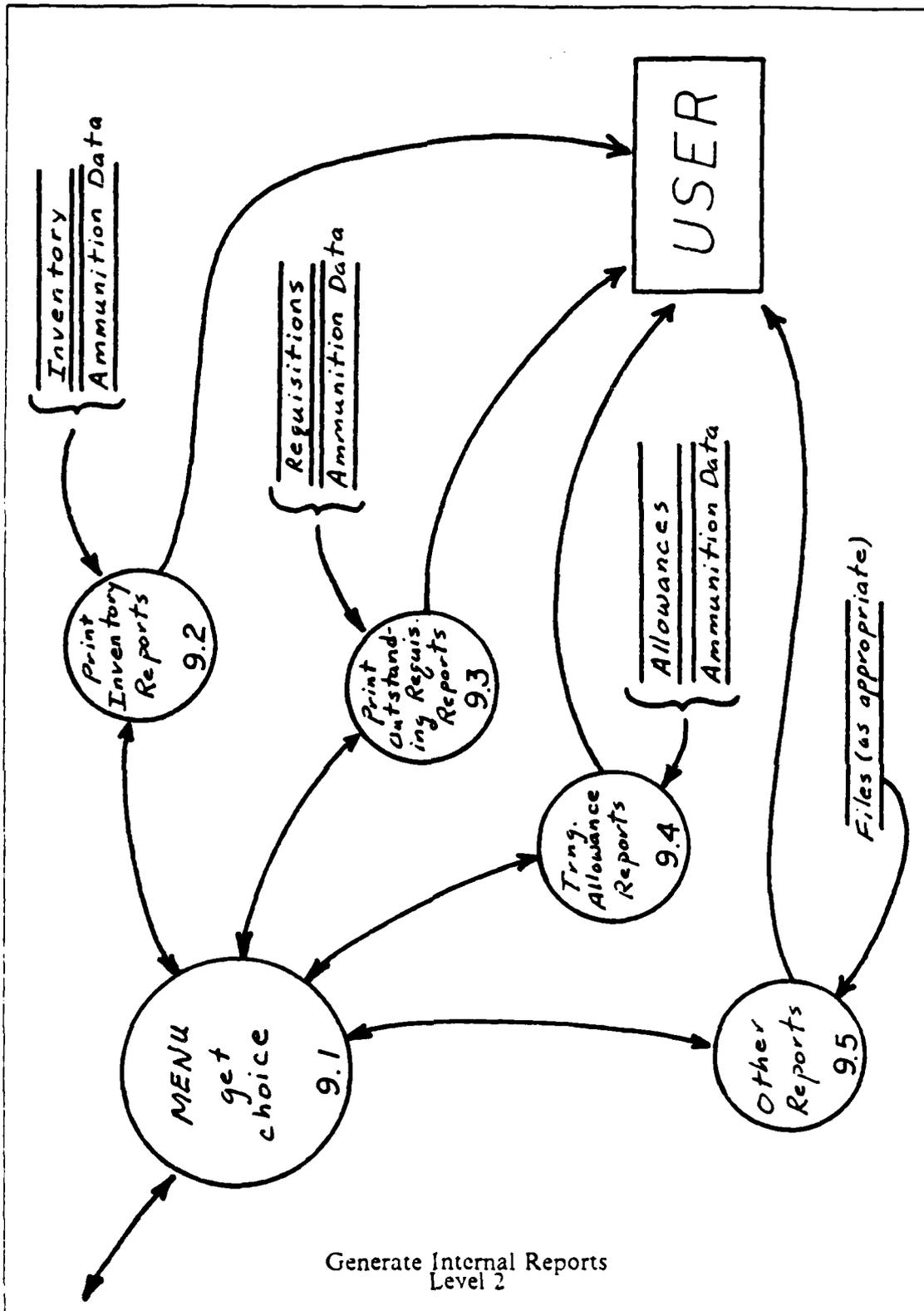






NARS Management
Process Applicable NARS
Level 3





APPENDIX B DATA ELEMENT DICTIONARY

DEN: D330
NAME: CSERNUM
LONG TITLE: Component Serial Number (First)
PIC: X(16)
DESC: The first instance of a serialized component in a transaction report, turn-in document, or a NAR action file entry. The definition of component serial number is the same data element SERNUMBER.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 1SERNUM
LONG TITLE: Component Serial Number (Second)
PIC: X(16)
DESC: Same as OSERNUM, except second instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 2SERNUM
LONG TITLE: Component Serial Number (Third)
PIC: X(16)
DESC: Same as OSERNUM, except third instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 3SERNUM
LONG TITLE: Component Serial Number (Fourth)
PIC: X(16)
DESC: Same as OSERNUM, except fourth instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 4SERNUM
LONG TITLE: Component Serial Number (Fifth)
PIC: X(16)
DESC: Same as OSERNUM, except fifth instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 5SERNUM
LONG TITLE: Component Serial Number (sixth)
PIC: X(16)
DESC: Same as OSERNUM, except sixth instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 6SERNUM
LONG TITLE: Component Serial Number (Seventh)
PIC: X(16)
DESC: Same as OSERNUM, except seventh instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 7SERNUM
LONG TITLE: Component Serial Number (Eight)
PIC: X(16)
DESC: Same as OSERNUM, except eight instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 8SERNUM
LONG TITLE: Component Serial Number (ninth)
PIC: X(16)
DESC: Same as OSERNUM, except ninth instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN: D330
NAME: 9SERNUM
LONG TITLE: Component Serial Number (tenth)
PIC: X(16)
DESC: Same as OSERNUM, except tenth instance.
USED IN: AMNARACT,AMTRANS,AMTURNIN
REFS:

DEN:
NAME: ACCESSLEV
LONG TITLE: Access Level
PIC: 9
DESC: Determines the level of user access or priveledges within the SAMS system; ranging from one to eight with one the most powerful and eight the lowest. Used to regulate access to various menu options from the main menu.
USED IN: DBSYSTEM
REFS:

DEN:
NAME: ACCOUNTNAM
LONG TITLE: Account Name
PIC: X(24)
DESC: An optional 24 character field in the users file that may be used by the SAMS manager to record the users full name or other information.
USED IN: DBSYSTEM
REFS:

DEN: E303
NAME: ACTCLCODE
LONG TITLE: Activity Classification Code
PIC: A
DESC: Indicates the intended use of ammunition stocks carried by a ship or activity.
USED IN: AMINVEN,AMTURNIN,AMTRANS,AMALLOW
REFS:

DEN: D192
NAME: ACTIVNAME
LONG TITLE: Activity Long Name
PIC: X(30)
DESC: Name of the activity as listed in the Plain Language Address Directories (PLAD) if available, otherwise in the clear name.
USED IN: AMDADDR
REFS:

DEN: KC26

NAME: ADVICCODE
LONG TITLE: Advice Code
PIC: X(2)
DESC: Advice Codes are numeric/alphabetic and flow from the requisition originators to initial processing points. The purpose is to provide coded instructions to supply sources when such data is considered essential to the supply action.
USED IN: AMREQUIS
REFS:

DEN:
NAME: ATRJULDAT
LONG TITLE: Ammunition Transaction Report Julian Date (SAMS LDE)
PIC: 999
DESC: The julian day of the year on which the ATR was prepared. Consists of the three number equivalent of the day of the year.
USED IN: AMTRANS
REFS:

DEN:
NAME: ATRREMARKS
LONG TITLE: Ammunition Transaction Report Remarks (SAMS LDE)
PIC: X(150)
DESC: Any narrative remarks that may be necessary to place on an ATR to clarify a transaction or provide loadout or RDD information.
USED IN: AMTRANS
REFS:

DEN: C089
NAME: ATRSERIAL
LONG TITLE: Ammunition Transaction Report Serial Number
PIC: 9(3)
DESC: A three digit number assigned to the sequence of an ATR report. The numbers run from 001 to 999 and then repeat for a particular unit.
USED IN: AMTURNIN,AMTRANS,AMNARACT
REFS:

DEN:
NAME: ATRSTATUS
LONG TITLE: Ammunition Transaction Report Status (SAMS local elem.)
PIC: A
DESC: A one character alphabetic code that indicates the status of an ATR document.
USED IN: AMTRANS
REFS:

DEN:
NAME: BEGBALANCE
LONG TITLE: Beginning Balance (SAMS local data element)
PIC: X(5)
DESC: The inventory balance of a particular ammunition item prior to a transaction occurring.
USED IN: AMTRANS
REFS:

DEN:
NAME: CALLED BY
LONG TITLE: Program called by
PIC: X(80)
DESC: The program that calls the program in question.
USED IN: PROGFILE
REFS:

DEN:
NAME: CALLS
LONG TITLE: Program Calls
PIC: X(100)
DESC: The name of the program(s) that a program calls.
USED IN: PROGFILE
REFS:

DEN:
NAME: CODES
LONG TITLE: Data Element Codes
PIC: memo
DESC: The CAIMS codes associated with a particular data element stored in a separate memo field.
USED IN: DATALEM
REFS:

DEN: C003
NAME: COGSYMBOL
LONG TITLE: Cognizance Symbol
PIC: X(2)
DESC: The Cognizance Symbol is a two position code prefix to National Stock Number to identify and designate the Inventory Control Point, office, or agency which exercises Supply Management of the item.
USED IN: AMMODATA
REFS:

DEN: C003E
NAME: CONDCODE
LONG TITLE: Condition Code
PIC: A
DESC: A single alphabetic character which classifies material in terms of readiness for issue and use, or to identify action underway to change the status of the material.
USED IN: AMINVEN,AMTURNIN,AMTRANS
REFS:

DEN:
NAME: CONTAINS
LONG TITLE: Relations used in a program
PIC: X(50)
DESC: The relations that are referenced or used by a particular program in the SAMS.
USED IN: CONTFILE
REFS:

DEN: K020
NAME: DEMANDCODE
LONG TITLE: Demand Code
PIC: X(A)
DESC: Demand Code, (R)recurring, (N)non-recurring
USED IN: AMREQUIS
REFS: (a),(b),(c)

DEN: G436
NAME: DEN
LONG TITLE: Data Element Number
PIC: X(6)
DESC: A six-digit alphanumeric data field used to identify data elements resident in the system data base. Obtained from NAVSUP Pub 508, Supply Management Program Standard Data Element Dictionary and keyword index.
USED IN: DATALEM

REFS:

DEN:

NAME: DESCRIPTIO

LONG TITLE: Description

PIC: X(40)

DESC: The meaning of a data element or the content of a file.

USED IN: DICFILES, DATAELEM

REFS:

DEN: K001

NAME: DOCIDENTIF

LONG TITLE: Document Identifier

PIC: XXX

DESC: The document identifier code provides identification of each document type (i.e. requisition, cancellation, follow-up, transfer, etc.)

USED IN: AMREQUIS

REFS: (A),(B),(c),(d)

DEN: K002B

NAME: DOCJULDAT

LONG TITLE: Document Julian Date

PIC: 9(4)

DESC: Identifies the date that the document or requisition was established. Consists of the units digit of the calender year and numeric values equivalent to the day of the year (julian date).

USED IN: AMTRANS

REFS:

DEN: K002C

NAME: DOCSERNUM

LONG TITLE: Document Serial Number

PIC: 9(4)

DESC: The portion of the Document Number which applies to the serial number of the document. Used in the Transaction File it can indicate the serial number of either a requisition or a Turn-in document, differentiated by the transaction code.

USED IN: AMTRANS

REFS:

DEN: K048

NAME: DOCSVCCOD

LONG TITLE: Document Service Code (SAMS local data element)

PIC: A

DESC: The service designator portion of the Document Number which is required on an issue or receipt ATR. Obtained from the requisition or turn_in file.

USED IN: AMTRANS

REFS:

DEN: A002

NAME: DOCUIC

LONG TITLE: Document Unit-Identification-Code (SAMS LDE)

PIC: X(5)

DESC: The UIC portion of a Document Number that is required on an issue or receipt ATR. Obtained from the requisition or turn-in file as appropriate.

USED IN: AMTRANS

REFS:

DEN: P255

NAME: DOTCLASCOD

LONG TITLE: Department of Transportation Class Code
PIC: XX
DESC: A code assigned to the classification assigned by the Department of Transportation to indicate the type of hazard involved when shipping the ammunition item.
USED IN: AMMODATA
REFS:

DEN:
NAME: ENDBALANCE
LONG TITLE: Ending Balance (SAMS local data element)
PIC: X(5)
DESC: The inventory balance of a particular ammunition item after a transaction has occurred.
USED IN: AMTRANS
REFS:

DEN: C042
NAME: FEDSUPCLAS
LONG TITLE: Federal Supply Classification
PIC: 9(4)
DESC: A code assigned by the government to designate various groups of common use, commercial type items.
USED IN: AMMODATA
REFS:

DEN:
NAME: FILE_NAME
LONG TITLE: File Name
PIC: X(8)
DESC: Name of the file and may be a database file(.dbf), or an index file(.ndx), or a format file(.fmt).
USED IN: DICFILES
REFS:

DEN:
NAME: FILE_TYPE
LONG TITLE: File Type
PIC: X(4)
DESC: Type of file: database(.dbf), index file(.ndx), format(.fmt)
USED IN: DICFILES
REFS:

DEN:
NAME: FRECLASNM
LONG TITLE: Freight Classification Nomenclature
PIC: X(32)
DESC: The proper shipping name prescribed for the material as required by Title 49 CFR, Part 172.101, and the DOT hazard class(spelled out). Given under the GBL Description column in NAVSEA OP2165, usually 2-5 word desc. and Class " " Explosive.
USED IN: AMMODATA
REFS:

DEN: K022
NAME: FUNDCODE
LONG TITLE: Fund Code
PIC: XX
DESC: A two character code which is used to cite accounting data on Navy requisitions. Fleet units use fund code Y6 and shore units use fund code 26.
USED IN: AMSTDATA
REFS:

DEN:
NAME: GROUP_NAME
LONG TITLE: Group Name
PIC: X(8)
DESC: An eight character word that must be entered by the
user of the SAMS in order for the protection
functions of the system to allow him access
to the system.
USED IN: DBSYSTEM
REFS:

DEN: D196
NAME: HAZARDMTCO
LONG TITLE: Coast Gaurd Hazardous Material Code
PIC: X(4)
DESC: Classification of military explosives and hazardous
munitions as determined by the US Coast Gaurd and
set forth in NAVSEA OP 2165 Vol.2. Ammunition aboard
all classes of ships must be loaded
in accordance with the guidance contained in CG108.
USED IN: AMMODATA
REFS:

DEN:
NAME: HULLNUMBER
LONG TITLE: Hull Number of vessel (SAMS local data element)
PIC: X(8)
DESC: Hull Number of ship; SSN685, FF1096, etc.
USED IN: AMDADDR, AMSTDATA
REFS:

DEN:
NAME: IDCODE
LONG TITLE: I.D. Code (SAMS local data element)
PIC: X(5)
DESC: The UIC of the issuing unit or a code indicating the
ACC or other source of the material involved in the
transaction; or, the UIC of the receiving unit or a
code indicating the ACC changed to or
other destination of the material.
USED IN: AMTRANS
REFS:

DEN: K002B
NAME: JULIANDATE
LONG TITLE: Document Julian Date
PIC: X(4)
DESC: Identifies date document or requisition was established.
The left most digit is the units digit of the current
year and the right three are the numeric equivalent of
the day of the year.
USED IN: AMREQUIS, AMTURNIN
REFS: (a), (b)

DEN: A045
NAME: LOCATION
LONG TITLE: Activity Long Name Location
PIC: X(30)
DESC: Plain Language Address of activity or ship in overhaul
or a precommissioned vessel.
USED IN: AMDADDR
REFS:

DEN:
NAME: LOGIN_NAME
LONG TITLE: Log-in Name
PIC: X(8)

DESC: An eight character name that must be entered properly by the user of SAMS for the security functions of the system to recognize him as a legal user and allow access to the system.

USED IN: DBSYSTEM

REFS:

DEN:

NAME: LONGTITLE

LONG TITLE: Long Title (Local SAMS data element)

PIC: memo

DESC: Official description of ammunition data as described in Stock List of Navy Ammunition, TWO10-AA-ORD-010 (SPCC)

USED IN: AMMODATA

REFS:

DEN:

NAME: LONGTITLE

LONG TITLE: Data Element Long Title

PIC: X(55)

DESC: The full title of the data element as used in the SAMS.

USED IN: DATAELEM

REFS:

DEN: C301

NAME: LOTNUMBER

LONG TITLE: Ammunition Lot Number

PIC: X(16)

DESC: A number assigned at the time of manufacture or assembly to identify a group of rounds of ammunition, each component of which is manufactured by one manufacturer under uniform conditions and which is expected to perform in a uniform way.

USED IN: AMINVEN,AMTURNIN,AMTRANS,AMNARACT

REFS:

DEN: C003A

NAME: MATLCONCOD

LONG TITLE: Material Control Code

PIC: A

DESC: The Material Control Code(MCC) is a single alphabetic character assigned by the Inventory Manager to indicate to field activities that special reporting or control requirements may be necessary. Used in CAIMS to indicate SLIT controlled item.

USED IN: AMMODATA

REFS:

DEN: C026

NAME: MDD

LONG TITLE: Maintenance Due Date

PIC: X(3)

DESC: The month and the year of the next scheduled maintenance on the item of record(MYY). MDD is assigned to serial number and serial and lot number controlled items only.

USED IN: AMINVEN,AMTRANS

REFS:

DEN: K082

NAME: MEDIASTAT

LONG TITLE: Media and Status Code

PIC: X

DESC: The Media and Status Code is a single character indicating the type of supply status required and the method in which it is to be furnished.

USED IN: AMREQUIS

REFS: (a),(b),(c),(d)

DEN: C076
NAME: MESSAGEDTG
LONG TITLE: Naval Message Date-Time-Group
PIC: X(14)
DESC: The standard means of identifying a naval message:
The DTG identified the day, the hour(24 hour clock),
Z(Zulu time), the month, and decade/year of transmittal.
USED IN: AMTRANS
REFS:

DEN:
NAME: MONITACTIV
LONG TITLE: Monitoring Activity (SAMS local data element)
PIC: X
DESC: The shore logistic or operational command which monitors
a ships logistics traffic and transaction reports. It is
used in conjunction with the Cognizance Symbol to form
the Dis tribution Code.
USED IN: AMSTDATA
REFS:

DEN: C003C
NAME: NALC
LONG TITLE: Naval Ammunition Logistics Code
PIC: X(4)
DESC: The NALC is a four character alphanumeric assigned by
SPCC to conventional ammunition items which do not
meet the established DoD criteria for DODAC assignment.
Application of NALC and DODAC are identical.
USED IN: AMMCDATA, AMALLOW
REFS:

DEN:
NAME: NAME
LONG TITLE: Name of data element
PIC: X(10)
DESC: The name of a data element in the SAMS.
USED IN: DATALEM
REFS:

DEN:
NAME: NAMEOFPROG
LONG TITLE: Name of Program
PIC: X(8)
DESC: The name of a program used in the SAMS.
USED IN: CONTFILE
REFS:

DEN: C078
NAME: NARDTG
LONG TITLE: Notice of Ammunition Reclassification Date-Time-Group
PIC: X(14)
DESC: The Date-Time-Group of the NAR message
USED IN: AMNARSER
REFS:

DEN:
NAME: NARREMARKS
LONG TITLE: Notice of Ammunition Transaction Remarks (SAMS LDE)
PIC: X(40)
DESC: Statement concerning the condition of affected
ammunition following a NAR action and/or the reason for
the NAR action; normally explained on the NAR itself.
USED IN: AMNARACT

REFS:

DEN: C084
NAME: NARSERIAL
LONG TITLE: Notice of Ammunition Reclassification Serial Number
PIC: 9(3)
DESC: One of two sub-elements comprising the NAR Number.
NAR serial serves the two-fold purpose of collecting
all items reclassified by a NAR action and identifying
the number of reclassification actions released
during a given year.
USED IN: AMTURNIN, AMNARSER, AMNARACT
REFS:

DEN: C083
NAME: NARYEAR
LONG TITLE: Notice of Ammunition Reclassification Year
PIC: 99
DESC: One of two sub-elements comprising the NAR Number.
NARYEAR identifies the decade and year in which the
Notice of Ammunition Reclassification(NAR) was released.
USED IN: AMTURNIN, AMNARSER, AMNARACT
REFS:

DEN: C304E
NAME: NETEXPLWT
LONG TITLE: Net Explosive Weight
PIC: X(10)
DESC: The total weight of all active explosive components of
an explosive device which includes primary explosives,
secondary explosives, pyrotechnics, and propellants.
Data should be expressed in whole numbers
with the units. (50 LB., 25 KG.)
USED IN: AMMODATA
REFS:

DEN: C003E
NAME: NEWCOND CD
LONG TITLE: New Condition Code
PIC: A
DESC: The Condition Code of ammunition items involved in a
transaction after their change in Condition Code as a
result of the transaction.
USED IN: AMNARACT
REFS:

DEN: D046D
NAME: NIIN
LONG TITLE: National Item Identification Number
PIC: X(9)
DESC: A nine-position non-significant number assigned by the
Defense Logistic Services Center to each approved item
identification under the Federal Cataloging Program.
USED IN: AMREQUIS, AMMODATA, AMINVEN, AMTURNIN, AMTRANS, AMNARACT
REFS: (a), (b)

DEN: C003E
NAME: OLDCOND CD
LONG TITLE: Old Condition Code (SAMS local data element)
PIC: A
DESC: The Condition Code of ammunition items that are involved
in a transaction prior to their change in Condition Code
as a result of the transaction.
USED IN: AMNARACT
REFS:

DEN:
NAME: PASSWORD
LONG TITLE: Password
PIC: X(16)
DESC: A sixteen character password of any case that must be entered properly by the user of SAMS in order for the security functions of the system to recognize him as a legal user and allow access to the system.
USED IN: DBSYSTEM
REFS:

DEN:
NAME: PICTURE
LONG TITLE: Picture of a data element
PIC: X(6)
DESC: The type of the data element, ie (X)character, (9)numeric, etc. and the length of the data element.
USED IN: DATAELEM
REFS:

DEN: K025
NAME: PRIORITYCD
LONG TITLE: Priority Code (Issue-Priority-Designator)
PIC: 99
DESC: A series of two-digit numeric codes assigned by the originator of the document which expresses the relative importance of the requisitioned material movement and the military urgency of the material movement and issue transactions.
USED IN: AMREQUIS
REFS:

DEN:
NAME: PROG_NAME
LONG TITLE: Program Name
PIC: X(8)
DESC: Name of a program in the SAMS.
USED IN: PROGFILE
REFS:

DEN: K024
NAME: PROJCODE
LONG TITLE: Project Code
PIC: X(3)
DESC: A code assigned by the military services or DoD to identify a specific project of a general or special program nature for recognition through out the distribution system.
USED IN: AMREQUIS,AMTURNIN
REFS: (a),(b),(c)

DEN:
NAME: PURPOSE
LONG TITLE: Purpose of a program
PIC: X(70)
DESC: The purpose of a program in the SAMS.
USED IN: PROGFILE
REFS:

DEN:
NAME: QUANTITY
LONG TITLE: Quantity
PIC: X(5)
DESC: Quantity in per unit-of-issue amounts. For quantities greater than 99999, use an M in the rightmost digit to indicate thousands. Normally all positions should be filled in, including leading zeros.

USED IN: AMREQUIS, AMINVEN, AMTURNIN, AMTRANS, AMALLOW
REFS:

DEN:
NAME: REFERENCE
LONG TITLE: Data Element References
PIC: X(70)
DESC: The Navy or Supply publications which fully describe
the purpose and use of a data element.
USED IN: DATAELEM
REFS:

DEN: K018
NAME: REQDELDATE
LONG TITLE: Required Delivery Date
PIC: 999
DESC: When used on a requisition, this element consists of
a three-digit julian date which indicates the date
that the material is required by the requisitionee.
USED IN: AMREQUIS
REFS:

DEN:
NAME: REQUISSTAT
LONG TITLE: Requisition Status (SAMS local data element)
PIC: X(A)
DESC: A local system code to indicate the status of a
requisition action, ex.incomplete, ready,
submitted-not-filled, etc.
USED IN: AMREQUIS
REFS:

DEN: A001
NAME: ROUTIDENT
LONG TITLE: Activity Routing Identifier
PIC: X(3)
DESC: A three digit alphanumeric or alphabetic code
assigned to Inventory Control Point, Inventory
Managers, Distribution Points, and designated storage
points. Used to indicate the intended recipient, the
actual shipper, or action orig. activity
USED IN: AMDADDR
REFS:

DEN: C017
NAME: SECRISKCOD
LONG TITLE: Security Classification Code
PIC: X
DESC: This code designates the degree of physical security
assigned to an item of supply.
USED IN: AMMODATA
REFS:

DEN: K048
NAME: SENDTOSERC
LONG TITLE: Service Designator Code (of requisitionee)
PIC: X
DESC: Service Designator Code of requisitionee, PAC, LANT,
etc.. A code that identifies a service or element of
a service.
USED IN: AMREQUIS
REFS: (a),(b),(c),(d)

DEN: A002
NAME: SENDTOUIC
LONG TITLE: Unit Identification Code of requisitionee

PIC: X(5)
DESC: UIC of requisitionee. Identifies a ship or shore activity uniquely in the manner specified by individual military services for accounting and other purposes.
USED IN: AMREQUIS
REFS: (a), (b)

DEN: K002C
NAME: SERIAL
LONG TITLE: Requisition/Turn-in Serial Number
PIC: 9(4)
DESC: The portion of the Document Number (DEN K002) which applies to the serial number of the document. Under CAIMS, ships use sequential numbers 8000-8999 and then repeat when necessary.
USED IN: AMREQUIS, AMTURNIN
REFS: (a), (b), (c)

DEN: D330
NAME: SERNUMBER
LONG TITLE: Component Serial Number
PIC: X(16)
DESC: An identification number given to each item manufactured within a particular lot of a given stock number.
USED IN: AMINVEN
REFS:

DEN: K048
NAME: SERVCODE
LONG TITLE: Service Code
PIC: X
DESC: A code that identifies a service or element of a service.
USED IN: AMDADDR, AMSTDATA
REFS:

DEN: A002
NAME: SHIPTOUC
LONG TITLE: Unit Identification Code of receiving activity
PIC: X(5)
DESC: Identifies a ship, shore activity, operational unit, agency, contractor, or other organized entity in the manner specified by the individual military service/agency for accounting or other purposes.
USED IN: AMTURNIN
REFS:

DEN:
NAME: SHORTTITLE
LONG TITLE: Short Title of ammunition item (SAMS data element)
PIC: X(20)
DESC: Short description of ammunition item for quick reference.
USED IN: AMMODATA
REFS:

DEN: K021
NAME: SIGNALCODE
LONG TITLE: Signal Code
PIC: X(A)
DESC: This code designates the fields (card columns) which contain the intended consignee (ship to) and the activity to receive the bills and effect payment (bill to).
USED IN: AMREQUIS
REFS:

DEN:
NAME: SOURCR_FIL
LONG TITLE: Source File of the data element
PIC: X(50)
DESC: The files (relations) in which the data element
is used.
USED IN: DATALEM
REFS:

DEN:
NAME: STORAGELC
LONG TITLE: Storage Location (SAMS local data element)
PIC: X(30)
DESC: The location onboard a ship where particular
ammunition items are stored, ex. small arms locker,
ready service locker, Magazine (compt. number), etc.
USED IN: AMINVEN
REFS:

DEN: K048
NAME: SUPADDSERC
LONG TITLE: Service Designator Code of loadout activity
PIC: X
DESC: Supplemental Address Service Designator Code, loadout
point. A single alpha code that designates a service
or element of a service.
USED IN: AMREQUIS
REFS:

DEN: A002
NAME: SUPADDUIC
LONG TITLE: Unit Identification Code of loadout activity
PIC: X(5)
DESC: Supplemental Address UIC, loadout point UIC.
Identifies a ship or shore activity uniquely in
the manner specified by its service for accounting
and other purposes.
USED IN: AMREQUIS
REFS:

DEN:
NAME: TAGLABEL
LONG TITLE: Tag Label (SAMS local data element)
PIC: X(30)
DESC: A short narrative that is placed on the label that
must be attached to NAR affected ammunition to
explain the conditions under which it may be used
or if it may be used and to segregate it from other
unrestricted use ammunition.
USED IN: AMNARACT
REFS:

DEN: C010
NAME: TMDC
LONG TITLE: Type of Maintenance Due Code
PIC: A
DESC: A code indicating the type of maintenance to be
performed on the item of record. Applies to
torpedo MK 46 and ALM (Air launched Missiles).
USED IN: AMMODATA
REFS:

DEN:
NAME: TRNGALLOW
LONG TITLE: Training Allowance (SAMS local data element)

PIC: 9(5)
DESC: The unit-of-issue quantity of a particular ammunition item that a ship or unit is authorized to expend in a fiscal year for training. Promulgated in the 30000 series NAVSEA Shipfill Allowance List.
USED IN: AMALLOW
REFS:

DEN:
NAME: TURNINSTAT
LONG TITLE: Turn-in Document Status (SAMS local data element)
PIC: A
DESC: The status of a turn-in record line item, ex.-incomplete, ready to print or use as an input to an ATR, or ATR action complete.
USED IN: AMTURIN
REFS:

DEN: A002
NAME: UIC
LONG TITLE: Unit Identification Code
PIC: X(5)
DESC: Identifies a ship, shore activity, operational unit, agency, contractor, or other organized entity in the manner specified by the individual military service/agencies for accounting or other purposes.
USED IN: AMDADDR,AMSTDATA
REFS:

DEN: D192
NAME: UNITNAME
LONG TITLE: Activity Long Name (own ship)
PIC: X(45)
DESC: Name of own ship as listed in the Plain Language Address Directory (PLAD) if available, otherwise in the clear name.
USED IN: AMSTDATA
REFS:

DEN: C005
NAME: UNITOFISSU
LONG TITLE: Unit of Issue
PIC: AA
DESC: An abbreviation which represents a determinate amount or quantity and serves as a unit of measurement when issuing the item, ex. BX,EA,etc.
USED IN: AMMODATA
REFS:

DEN: B053
NAME: UNITPRICE
LONG TITLE: Unit Price
PIC: 9(10)
DESC: The price of the individual item of supply per unit of issue.
USED IN: AMMODATA
REFS:

DEN:
NAME: USED/FY
LONG TITLE: Training Allowance used in current fiscal year (LDE)
PIC: 9(5)
DESC: The quantity of a particular ammunition item that has a training allowance that has been expended in the current fiscal year for that purpose.
USED IN: AMALLOW

APPENDIX C

SAMS PROGRAM DIRECTORY

Program: AMSMAIN
Calls: INFOMENU, INVMENU, TRANMENU, REQMENU, NARSMENU, REPTMENU,
MANGMENU, TURNMENU
Purpose: Main prg. that branches to all subsystem menus,
intro. screen.
Called By: PROTECT (dBase III+ security program)

Program: BCKUPATR
Calls: none
Purpose: Backups ATR file to another storage device for system
protection.
Called By: TRANMENU

Program: BCKUPNAR
Calls: none
Purpose: Backups NAR Serial and Action File for system
protection.
Called By: NARSMENU

Program: BCKUPREQ
Calls: none
Purpose: Backups Requisition File to another storage device
for sys. protection.
Called By: REQMENU

Program: BCKUPSYS
Calls: none
Purpose: Allows system manager to bacup all system files.
Called By: MANGMENU

Program: BCKUPTUR
Calls: none
Purpose: Backups Turn-in File to another storage device for
system protection.
Called By: TURNMENU

Program: COMPLREQ
Calls: none
Purpose: Offers detailed review of requis. with all pertinent
data.
Called By: REQMENU

Program: CRENEWRO
Calls: none
Purpose: Creates new requis. with detailed user instructions
Called By: REQMENU

Program: CRENWATR
Calls: none
Purpose: Allows user to create an ATR record, collects all data
Called By: TRANMENU

Program: CRENWATID
Calls: REVWADD
Purpose: Creates a turn-in/issue line-entry.
Called BY: TURNMENU

Program: DATALEM
 Calls: none
 Purpose: Allow the user to review the Data Element Dictionary
 of the SAMS.
 Called By: INFOMENU

Program: DOCCODES
 Calls: none
 Purpose: Presents system and CAIMS codes.
 Called By: INFOMENU

Program: DOCDEFIN
 Calls: none
 Purpose: Presents system and CAIMS definitions.
 Called By: INFOMENU

Program: DOCFILES
 Calls: none
 Purpose: Presents SAMS files and their purpose.
 Called By: MANGDOC

Program: DOCPRGS
 Calls: none
 Purpose: Presents SAMS programs and their purpose.
 Called By: MANGMENU

Program: DOCREFS
 Calls: none
 Purpose: Presents system and CAIMS reference documents.
 Called: INFOMENU

Program: EDITATR
 Calls: none
 Purpose: Allows use to edit or delete non-committed ATR's.
 Called By: TRANMENU

Program: EDITREQ
 Calls: none
 Purpose: Allows editing and deleting of non-submitted
 requisitions.
 Called By: REQMENU

Program: EDITTURN
 Calls: none
 Purpose: Allows editing or deleting of non-submitted turn-in/
 issue document.
 Called By: TURNMENU

Program: INFOHELP
 Calls: none
 Purpose: Provide system operating help accessible throughout
 program.
 Called By: INFOMENU

Program: INFOMENU
 Calls: INFOTEXT, INFOHELP, DOCCODES, DOCDEFIN, DOCREFS, DATALEM
 Purpose: Gives general system info. and help programs.
 Called By: AMSMAIN

Program: INFOTEXT
 Calls: none
 Purpose: Give general system purpose, operation, and other
 information.

Called By: INFOMENU

Program: INVALLOW
Calls: none
Purpose: Displays Allowance List information to SAMS user.
Called By: INVMENU

Program: INVAMMO
Calls: none
Purpose: Allows user to review generic ammunition data
Called By: INVMENU

Program: INVMENU
Calls: INVVIEW, INVAMMO, INVALLOW
Purpose: Allows user to review generic ammo data, inventory
status, allowance.
Called By: AMSMAIN

Program: INVREPT
Calls: none
Purpose: Prints various inventory reports depending on user
desires.
Called By: REPTMENU

Program: INVVIEW
Calls: none
Purpose: Allow user to review onboard inventory with various
field items.
Called By: INVMENU

Program: MANGADHC
Calls: none
Purpose: Allows administrator to create ad hoc queries and
views.
Called By: MANGMENU

Program: MANGARCH
Calls: none
Purpose: Allows administrator to archive old records.
Called By: MANGMENU

Program: MANGDOC
Calls: DOCFILES, PROGFILE, STRUCRT
Purpose: Allows system manager to select various documentation
files for view.
Called By: MANGMENU

Program: MANGEDIT
Calls: none
Purpose: Allows administrator to globally edit and delete
system records.
Called By: MANGMENU

Program: MANGINIT
Calls: none
Purpose: Start-up and initialization instructions for
administrator.
Called By: MANGMENU

Program: MANGMENU
Calls: MANGEDIT, MANGSEC, MANGARCH, MANGRECV, MANGADHC,
MANGINIT, BCKUPSYS, DOCFILES, DOCPRGS
Purpose: Selects type of system management desired
by system administrator.

Called By: AMSMAIN

Program: MANGRECV
Calls: none
Purpose: Allows administrator to recover system from failure.
Called By: MANGMENU

Program: MANGSEC
Calls: PROTECT (dBase III+ security program)
Purpose: Allows administrator to manage security and
access system.
Called By: MANGMENU

Program: MISCREQ
Calls: none
Purpose: Refers user to other doc. for abnormal requis.
operations
Called By: REQMENU

Program: NARSMENU
Calls: REVNARSF, REVNARAF, PROCNAR, BCKUPNAR
Purpose: Allows review of NAR file, stockcheck, action
file update, backup
Called By: AMSMAIN

Program: OSRQREPT
Calls: none
Purpose: Prints outstanding requisition internal reports.
Called By: REPTMENU

Program: PRINTATR
Calls: none
Purpose: Allows printing of the various types of ATR's.
Called By: TRANMENU

Program: PRINTREQ
Calls: REQ1348, REVWADD
Purpose: Allows printing of different requisition formats
Called By: REQMENU

Program: PRINTTUR
Calls: none
Purpose: Prints turn-in document.
Called By: TURNMENU

Program: PROCNAR
Calls: none
Purpose: Processes NAR message: stock check, update inventory,
retain data
Called By: NARSMENU

Program: PROGFILE
Calls: none
Purpose: Allows user to review and print system program file.
Called By: MANGDOC

Program: REPTMENU
Calls: INVREPT, OSRQREPT, TRALREPT
Purpose: Prints various internal system reports.
Called By: AMSMAIN

Program: REQ1348
Calls: none
Purpose: Prints screen display of DD Form 1348 with

data filled in.
Called By: PRINTREQ

Program: REOMENU
Calls: REWVREQ, COMPLREQ, CRENEWRO, EDITREQ, MISCREQ, PRINTREQ,
BCKUPREQ
Purpose: Branches to all requisition processing sub-functions.
Called By: AMSMAIN

Program: REVNARAF
Calls: none
Purpose: Allows user to review the NAR Action File.
Called By: NARSMENU

Program: REVNARSF
Calls: none
Purpose: Allows user to review the NAR Serial File.
Called By: NARSMENU

Program: REWADD
Calls: none
Purpose: Allows user to review address file.
Called By: PRINTREQ, PRINIATR, CRENWTID

Program: REWVREQ
Calls: none
Purpose: Offers summary(exec.) review of all requis. with
most imp. data.
Called By: REQMENU

Program: STRUCCRT
Calls: none
Purpose: Allows system manager to review the SAMS design
structure charts.
Called By: MANGDOC

Program: TRALREPT
Calls: none
Purpose: Prints Training Allowance internal reports.
Called By: REPTMENU

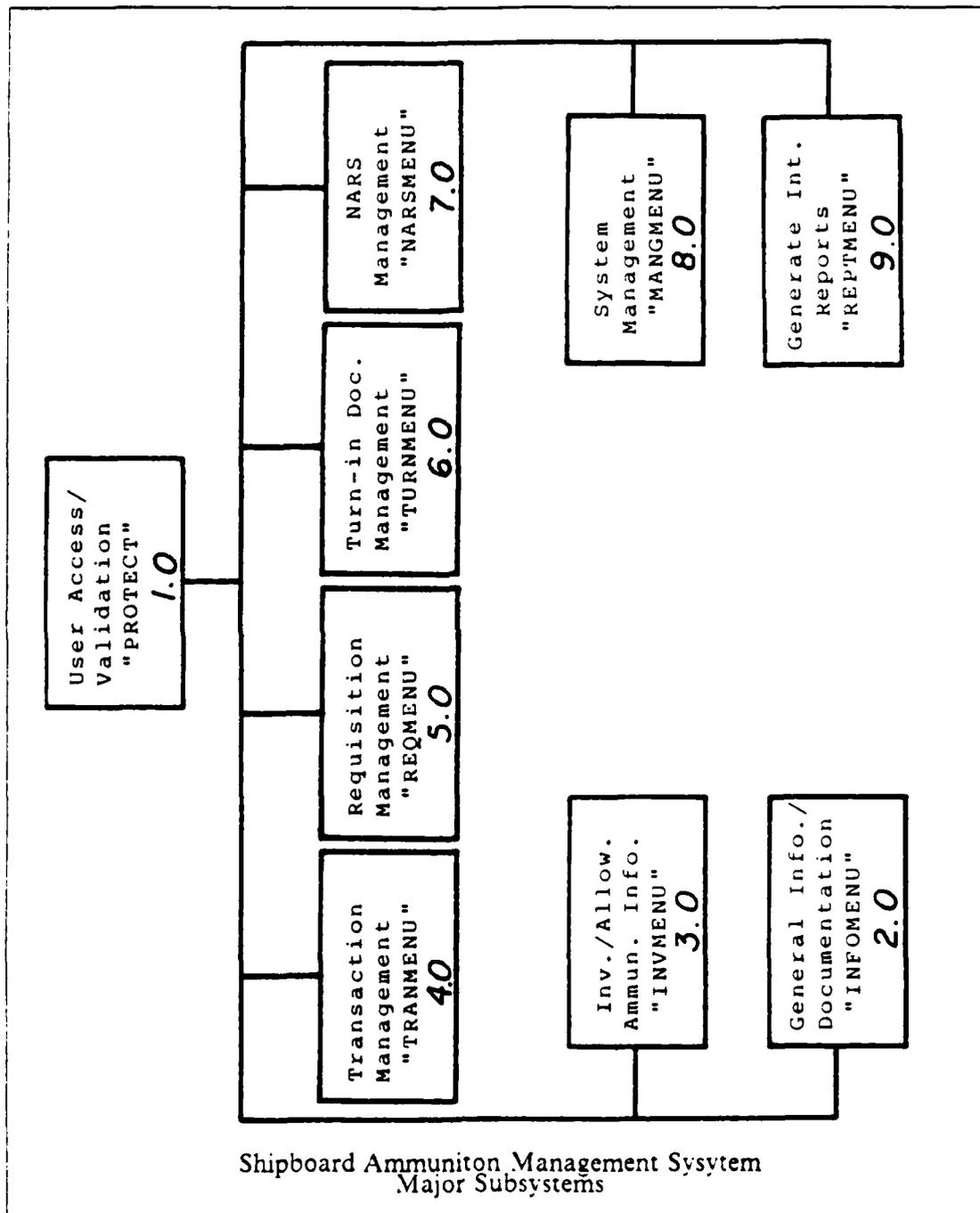
Program: TRANMENU
Calls: VIEWATR, CRENWATR, EDITATR, PRINTATR, BCKUPATR
Purpose: Selects type of ATR management desired.
Called By: AMSMAIN

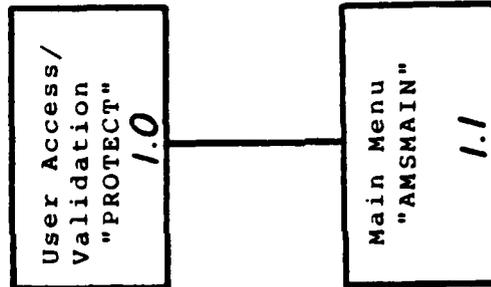
Program: TURNMENU
Calls: TURNREV, CRENWTID, EDITTURN, PRINTTUR, BCKUPTUR
Purpose: Selects type of turn-in/issue management desired.
Called By: AMSMAIN

Program: TURNREV
Calls: none
Purpose: Allows review of turn-in file.
Called By: TURNMENU

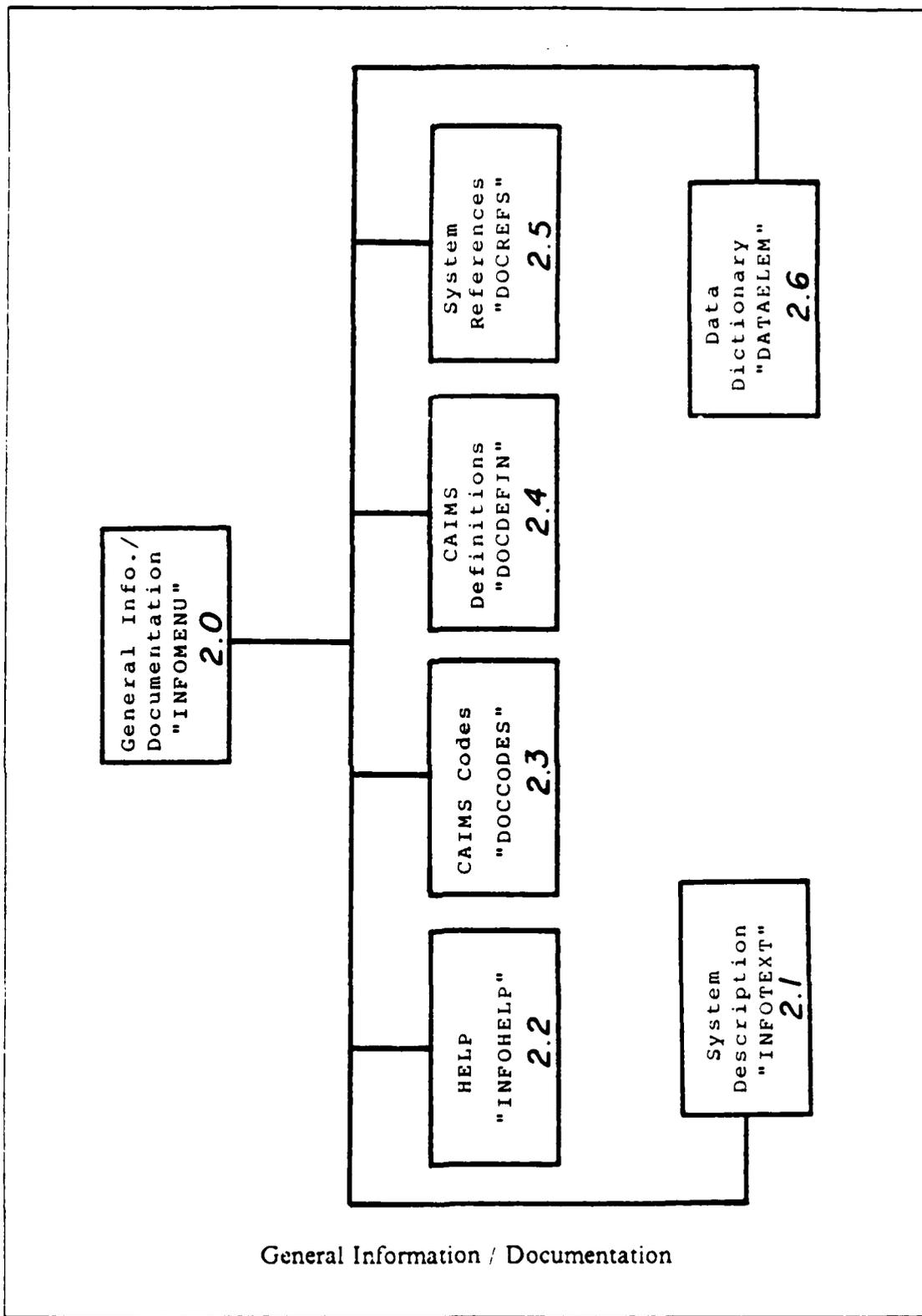
Program: VIEWATR
Calls: none
Purpose: Allow user to review the ATR file.
Called By: TRANMENU

APPENDIX D SAMS STRUCTURE CHARTS

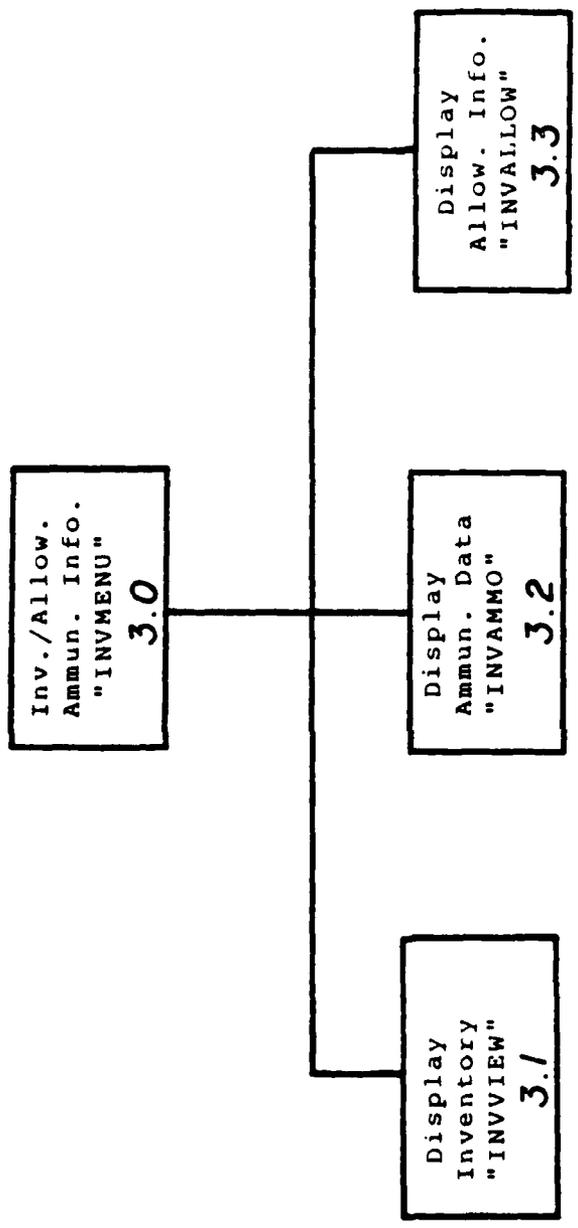




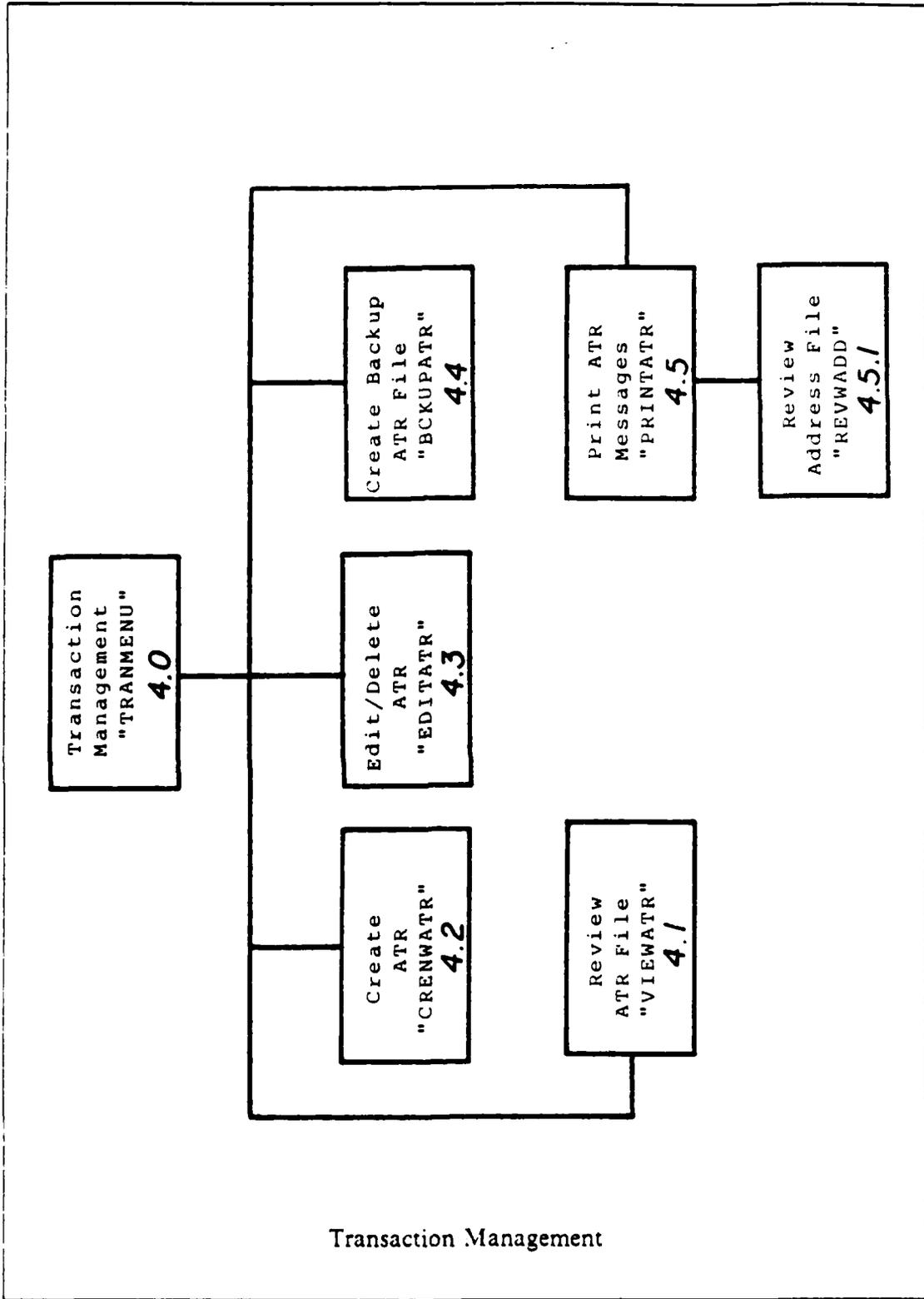
User Access / Main Menu



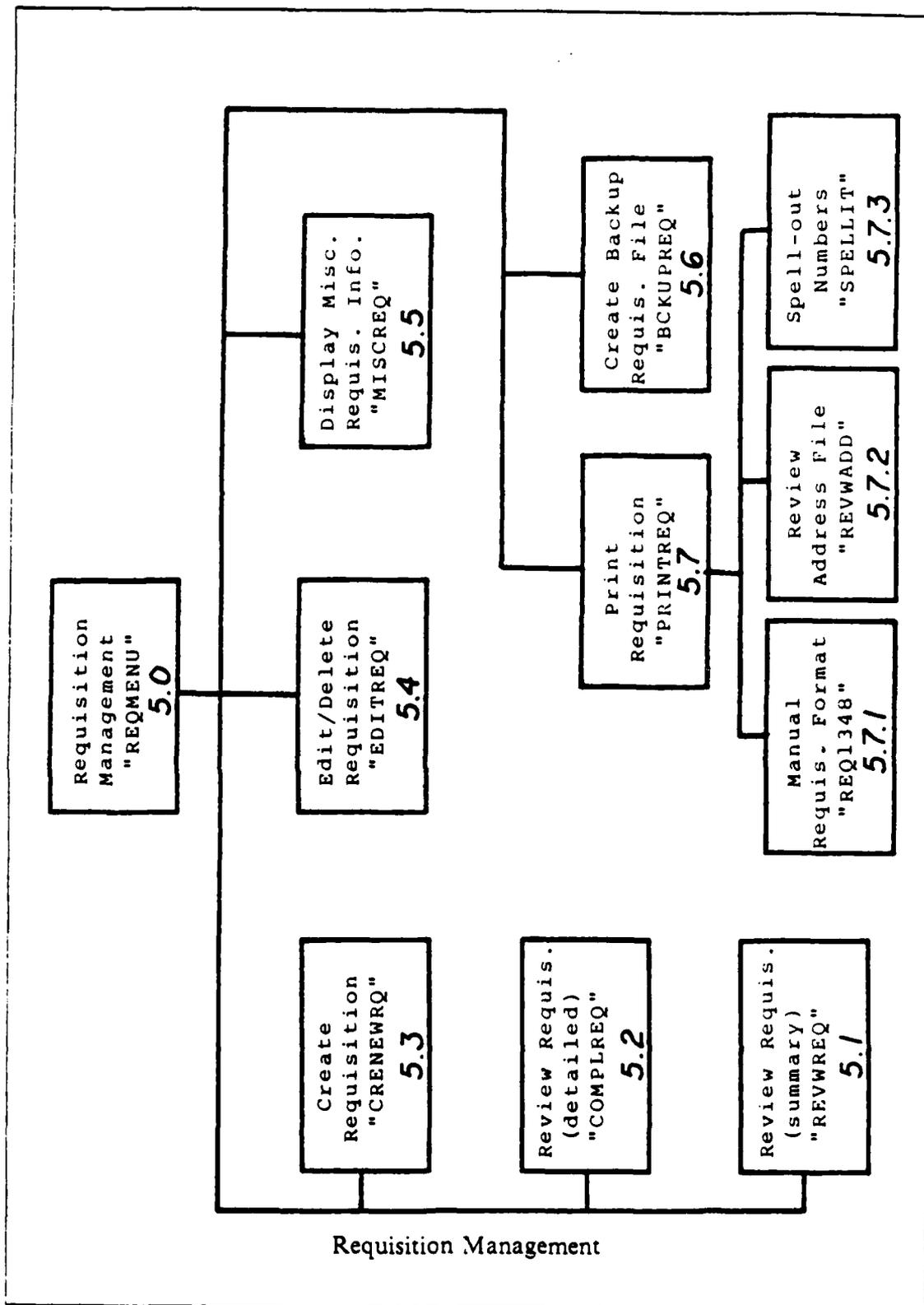
General Information / Documentation

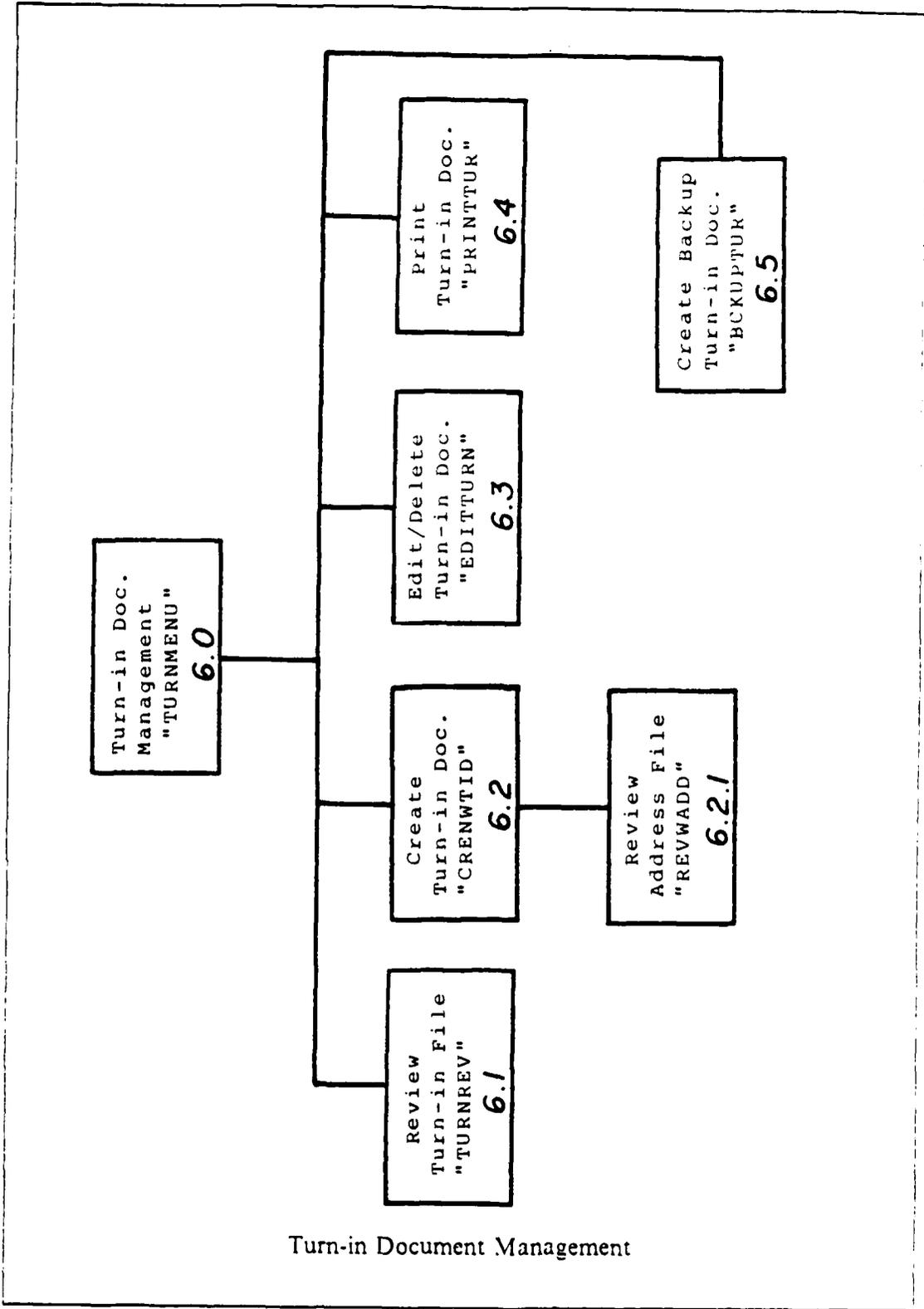


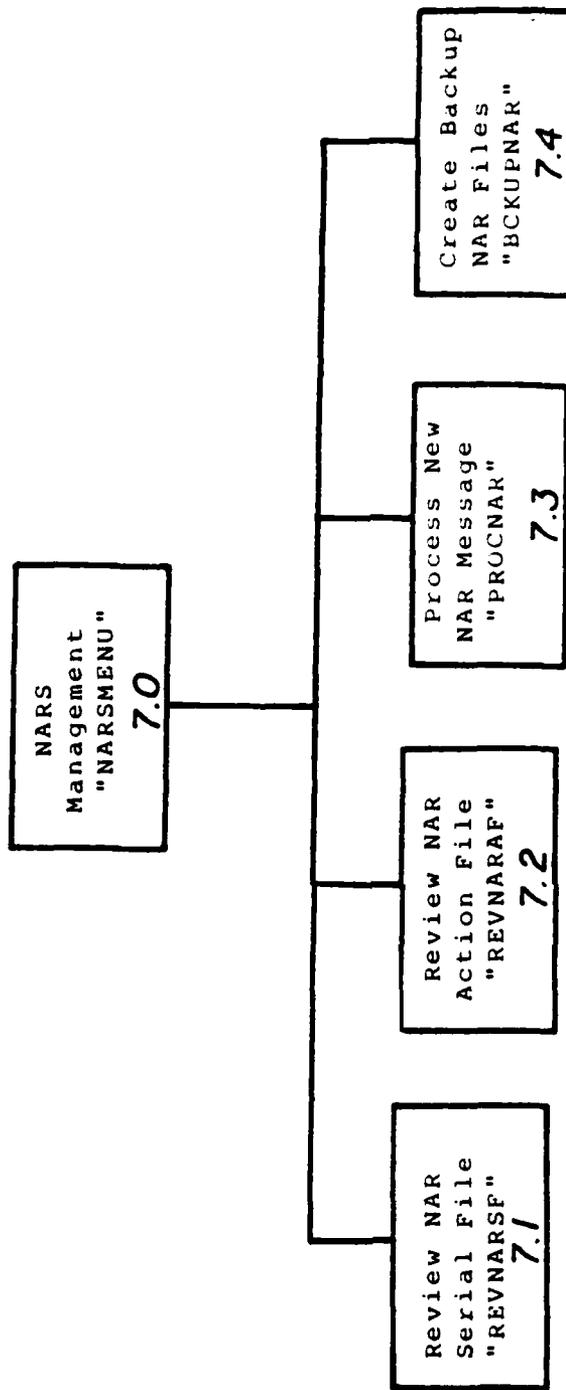
Inventory, Allowance, Ammunition Data



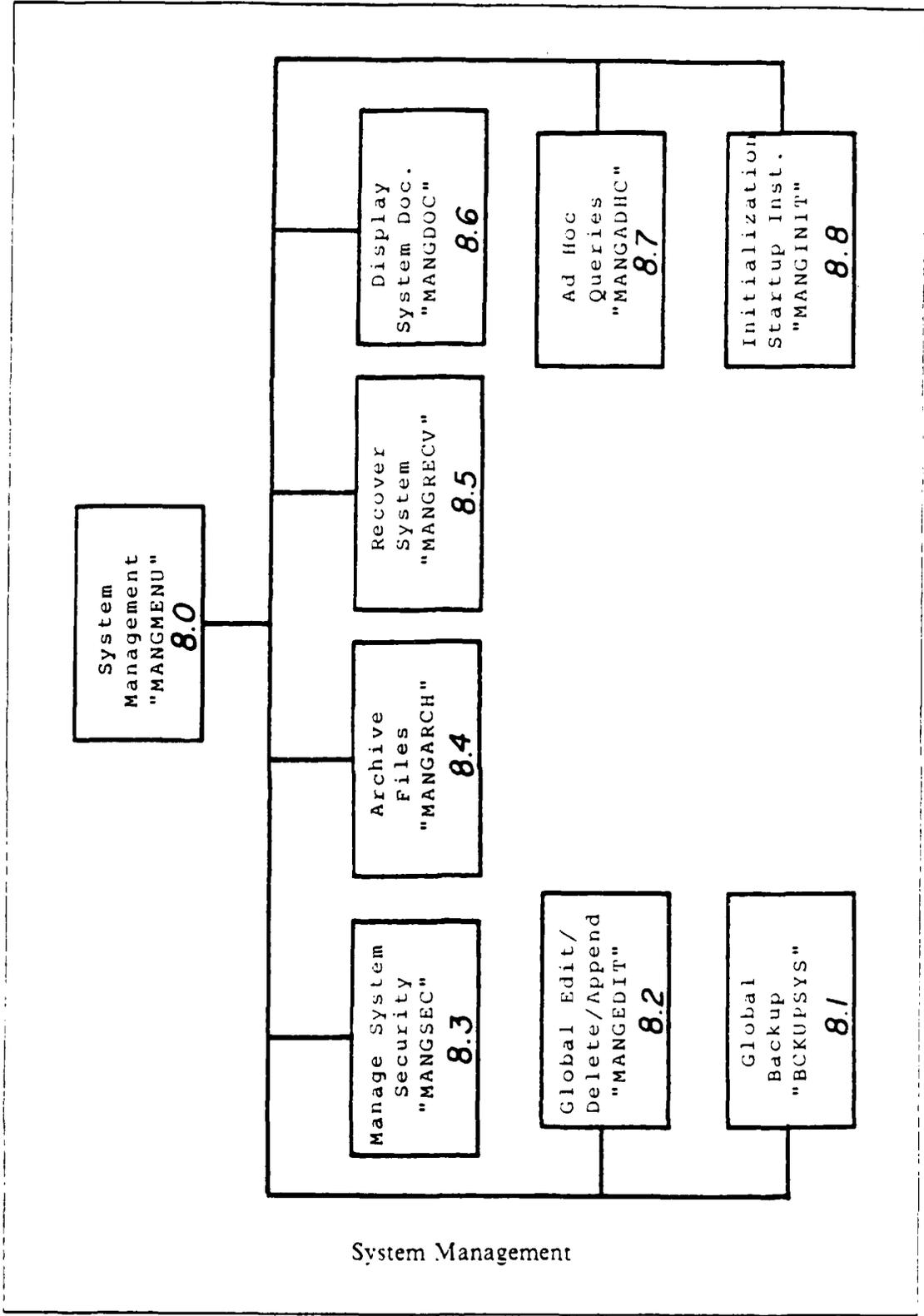
Transaction Management







NARS Management



Generate Int.
Reports
"REPTMENU"
9.0

Print
Inventory Rpt.
"INVREPT"
9.1

Print Outstand-
ing Req. Rpt.
"OSRQREPT"
9.2

Training
Allow. Rpt.
"TRALREPT"
9.3

Other
Reports
9.4

Generate Internal Reports

APPENDIX E

SAMS PROGRAM LISTINGS

1. AMSMAIN.PRG

*AMSMAIN.PRG
*First Version of AMS main program
*Written by LT. Steven L. Smith, USN June 4, 1987

```
clear all
set talk off
set status off
set safety off
set deleted on
set confirm off
set delimiters on
set delimiters to "{}"
```

```
clear
@ 4,10 to 10,70 double
@ 6,22 say " U.S.S. Navy Ship"
@ 8,22 say " Shipboard Ammunition Management System"
@ 1,1 to 20,78 double
@ 12,27 say "Today is " + cday(date()) + ", " + cmonth(date()) + ";
" + str(day(date()),2) + ", " + str(year(date()),4)
@ 18, 10 say "Author: LT. Steven L. Smith, USN "
@ 22, 10 say " Press any key to display the Main Menu "
wait " "
do while .T.
clear
text
```

SAMS MAIN MENU

1. General Information/Documentation
2. Inventory/Ammunition/Allowance Data
3. Transaction Management
4. Requisition Management
5. Turn-in Document Management
6. NARS Management
7. System Management
8. Generate Internal Reports

88. Exit to dBase III Plus Command Level
99. Exit to MS-DOS Operating System

```
endtext
@ 1,0 to 24,79 double
@ 3,1 to 3,78
@ 18,1 to 18,78
store 0 to MSELECT
@ 19,22 say " Enter your selection: " get MSELECT picture "99"
read
do case
case MSELECT = 1
do INFOMENU
case MSELECT = 2
do INVMENU
case MSELECT = 3
do TRANMENU
case MSELECT = 4
```

```

do REQMENU
case MSELECT = 5
do TURNMENU
case MSELECT = 6
do NARSMENU
case MSELECT = 7
do MANGMENU
case MSELECT = 8
do REPTMENU
case MSELECT = 88
clear
set talk on
set status on
set safety on
set deleted off
return
case MSELECT = 99
clear
quit
otherwise
@ 22,16 say "Not a valid selection--"
@ 23,16 say "Press any key to try again."
wait " "
endcase
enddo [.T.]
clear all
return

```

2. BCKUPREQ.PRG

```

* BCKUPREQ.PRG
* This program backs up the requisition file to another storage
* device for system protection.
* Written by LT. Steven L. Smith, USN 1 Sept., 1987

clear
@ 5,15 say "Requisition File Backup"
?
?
text
This selection will copy the current up-to-date contents
of the Requisition File to your backup floppy disk. It is
important to do this after each session in which you change
the contents of the file, just in case something catastrophic
happens to the hard disk.
endtext
@ 18,5 say "Place the backup floppy disk with the Requisition"
@ 19,5 say "File on it in drive A:"
store " " to MCONFIRM
@ 22,10 say "Are you ready to backup the file? (Y/N)";
get MCONFIRM picture "!"
read
if MCONFIRM = "Y"
* run BACKUP AMREQUIS.dbf A:
run BACKUP AMREQUIS.dbf A:/A
wait "Backup Complete - Press any key to return"
else
return
endif
return

```

3. COMPLREQ.PRG

```
* COMPLREQ.PRG
* Program to view detailed requisition data
* Written by LT. Steven L. Smith, USN 15 June, 1987

clear
clear all
do while .T.
@ 1, 10 to 3, 60 double
@ 2, 15 say "View Complete Requisition Data"
?
?
text
    1. Specific Requisition (* must know serial number *)
    2. All Requisitions (* newest to oldest *)
    3. QUIT
endtext
@ 0,0 to 24,78 double
store " " to CHOICE
@ 18, 15 say "Enter choice: " get CHOICE picture "!"
read
do case
    case CHOICE = "1"
        clear
        store 0 to SELECTION
        @ 10,10 say "Enter the requisition serial number:";
        get SELECTION picture "9999" range 8000,8999
        read
        select D
        use AMREQUIS index AMRSERUP,AMRSERDW
        seek SELECTION
            if found()
                clear
                exit
            else
                @ 14,10 say "That serial number is not in the file."
                wait
                clear
            endif
        case CHOICE = "2"
            select D
            use AMREQUIS index AMRSERUP,AMRSERDW
            goto bottom
            clear
            exit
        case CHOICE = "3"
            clear
            close databases
            return
        otherwise
            @ 14,10 say "Not a valid selection-"
            @ 15,10 say "Press any key to try again"
            wait " "
            clear
    endcase
enddo
select C
    use AMMODATA index AMANIIN
select D
do while .not. bof()
    set relation to NIIN into AMMODATA
```

```

store C->NALC to AMARK
store C->SHORTTITLE to BMARK
store C->UNITOFISSU to CMARK
store C->COGSYMBOL to DMARK
set relation to
store SENDTOUIC to EMARK
store SUPADDUIC to FMARK
select B
use AMDADDR index AMDUIC
goto top
seek EMARK
    if found()
        store ACTIVNAME to GMARK
        store LOCATION to HMARK
    else
        store "          " to GMARK
        store "          " to HMARK
    endif
goto top
seek FMARK
    if found()
        store ACTIVNAME to IMARK
        store LOCATION to JMARK
    else
        store "          " to IMARK
        store "          " to JMARK
    endif
select A
use AMSTDATA
goto top
store SERVCODE to KMARK
store UIC to LMARK
select D
@ 2, 5 say "Document Number: " +KMARK+LMARK+" ";
        (ltrim(str(SERIAL))+" "+ JULIANDATE
@ 4, 5 say "NALC: "+AMARK
@ 4, 20 say "NIIN: "+NIIN
@ 4, 40 say "Short title: "+BMARK
@ 6, 5 say "Quantity: "+QUANTITY
@ 6, 25 say "Unit-of-issue: "+CMARK
@ 6, 48 say "COG Symbol: "+DMARK
@ 8, 5 say "Send to: "+SENDTOSERC+SENDTOUIC+", "+;
        rtrim(GMARK)+"", "+ rtrim(HMARK)
@ 10, 5 say "Supplemental Address: "+SUPADDSERC+SUPADDUIC+;
        ", "+ rtrim(IMARK) +", "+ rtrim(JMARK)
@ 12, 5 say "Project Code: "+PROJCODE
@ 12, 25 say "Doc. Ident.: "+DOCIDENTIF
@ 12, 47 say "Media/Status Code: "+MEDIASTAT
@ 14, 5 say "Signal Code: "+SIGNALCODE
@ 14, 24 say "Demand Code: "+DEMANDCODE
@ 14, 45 say "Advice Code: "+ADVCECODE
@ 16, 5 say "Priority Code: "+PRIORITYCD
@ 16, 32 say "Required Delivery Date: "+REQDELDATE
@ 19, 45 say "Requisition Status: "+REQUISSTAT
@ 18, 43 to 20, 68
    skip-1
        if CHOICE = "1"
            @ 22, 5 say "Press any key to return"

```

```

        wait " "
        close databases
        return
    endif
    if bof()
        @ 22, 50 say "End of File"
    endif
    do while .T.
        store " " to MCHOICE
        @ 22,5 say "(C)Continue, (R)Repeat, (X)Exit ";
        get MCHOICE picture "!"
        read
        do case
            case MCHOICE = "C" .OR. MCHOICE = "R"
                if bof()
                    goto bottom
                    clear
                    exit
                else
                    clear
                    exit
                endif
            case MCHOICE = "X"
                clear
                close databases
                return
            otherwise
                @ 23, 20 say "Not a valid selection, press any key:"
                wait " "
                @ 23, 1 clear
                @ 24, 1 clear
            endcase
        enddo
    enddo
enddo
close databases
return

```

4. CRENEWREQ.PRG

```

* CRENEWREQ.PRG
* Program to create a new requisition
* Written by LT. Steven L. Smith, USN 8 June, 1987

```

```

clear
set bell off
clear all
use AMREQUIS index AMRSERDW,AMRSERUP,AMRREQDD

```

```

@ 2,15 say " CREATING A NEW REQUISITION"
@ 1,10 to 3,47 double

```

```

?
?
text

```

Requisitions require some coded information that is very difficult to remember. Each screen in this subprogram will ask you for information and offer a description of the information and the choices available to you. Some items are mandatory on a requisition, and the screen will tell you so, but it will not prevent you from leaving the item blank. It is important then that you check to make sure the requisition is complete before submitting it.

```

endtext

```

```

?
?
wait "      Press any key to start entering data"
@ 5,0 clear
***** Serial Number Decision *****
goto top
store 1 to COUNT
  if ( COUNT + SERIAL ) = 9000
    text
      NOTE: The requisition serial numbers have reached
            8999 and should be restarted at 8000 IAW
            SPCCINST 8010.12D. You'll have the
            opportunity to do that by continuing.
    endtext
    @ 6,5 to 12,70 double
    wait
    clear
  endif

do while .T.
@ 0,0 clear to 5,50
@ 2,15 say "Requisition Serial Number "
@ 1,10 to 3,47 double
store " " to CHOICE
store 0 to VSERIAL
@ 9,8 say "The next sequential requisition serial number is: ";
      +ltrim(str(COUNT + SERIAL))
@ 10,8 say " Is that serial number O.K.? (Y,N)" get CHOICE ;
      picture "@!"
      read

  if CHOICE = "Y"
    VSERIAL = SERIAL + COUNT
    exit
  else
    if CHOICE = "N"
      @ 5,0 clear
      text
        WARNING: A non-sequential serial number should only
        be used when you reach requisition 8999 and need to
        reset the numbers back to 8000 IAW SPCCINST 8010.12D.
        The system will alert you when this happens. A serial
        number less than the one reported above could be a
        duplicate; the system also attempts to check for
        this.
      endtext
      @ 6,5 to 15,75 double
      do while .T.
        store " " to MCHOICE
        @ 17,6 say "Do you still wish to choose a serial";
        +" number other than " + ltrim(str(COUNT + SERIAL));
        +" (Y/N)?";
        get MCHOICE picture "@!"
        read

          if MCHOICE = "N"
            clear
            exit
          endif

          if MCHOICE <> "Y" .AND. MCHOICE <> "N"
            @ 22,8 say "That is not a valid response-- "
            @ 23,8 say "Press any key to try again: "
            wait " "
            clear
          endif
        endif
      endwhile
    endif
  endif
endif

```

```

endif
  if MCHOICE = "Y"
    do while .T.
      store 0 to VSERIAL
      @ 22,8 say "Enter the requisition number:";
      get VSERIAL picture "9999" range 8000,;
      8999
      read
      set index to amrserup,amrserdw,amrreqdd
      seek VSERIAL
      if found()
        @ 5, 0 clear
        @ 16,20 say "That is a duplicate serial number--"
        @ 17,20 say " Press any key to try again:"
        wait " "
      else
        exit
      endif
    enddo
    if MCHOICE = "Y"
      exit
    endif
  endif
enddo
else
  if CHOICE <> "Y" .AND. CHOICE <> "N"
    @ 12,8 say "That is not a valid response--"
    @ 13,8 say "Press any key to try again "
    wait " "
    clear
  endif
endif
endif
  if MCHOICE = "Y"
    exit
  endif

enddo
close databases
***** End -- Serial Number Decision *****
***** Enter NIIN *****
clear
store " " to VNIIN
@ 5,10 say "National Item Identification Number --NIIN"
@ 4,5 to 6,70 double
do while .T.
  store " " to CHOICE
  @ 8, 15 say "Do you know the NIIN of the item you wish to"
  @ 9, 15 say "order?(Y/N):" get CHOICE picture "!"
  read
  do case
  case CHOICE = "N"
  ?
  text
  Go to the Inventory Management program from the main
  menu and find the NIIN of the item you wish to order.
  If this is a new item not yet in the database, be
  sure to add it. Contact the SAMS coordinator or
  Weapons Officer if you do not have the access to
  do this.
  endtext
  ?
  wait

```

```

clear
close databases
return to master
case CHOICE = "Y"
  @ 11, 10 say "Enter the NIIN (leave out the blanks)";
  get VNIIN picture "@"
  read
  use AMMODATA index AMANIIN, AMANALC
  seek VNIIN
  if found()
    do while .T.
      store " " to MCHOICE
      @ 13, 5 say "That NIIN is for: "
      set heading on
      display NIIN,NALC,SHORTTITLE
      @ 18, 5 say "Is this the correct item?(Y/N) ";
      get MCHOICE picture "!"
      read
      do case
        case MCHOICE = "Y"
          @ 7, 0 clear
          exit
        case MCHOICE = "N"
          ?
          text
          Go to the Inventory Management program
          from the main menu to verify the NIIN.
          endtext
          ?
          wait
          clear
          close databases
          return to master
        otherwise
          @ 20, 10 say "That is not a valid response"
          @ 21, 10 say "Press any key to try again-"
          wait
          @ 7, 0 clear
      endcase
    enddo
    if MCHOICE = "Y"
      exit
    endif
  else
    @ 15, 10 say "That NIIN is not in the General"
    @ 16, 10 say "Ammunition Information file."
    ?
    text
    Go to the Inventory Management program from the
    main menu and verify that NIIN.
    endtext
    ?
    wait
    clear
    close databases
    return to master
  endif
otherwise
  @ 10, 10 say "That is not a valid response--"
  @ 11, 10 say "Press any key to try again: "
  wait

```

```

      @ 7, 0 clear
    endcase
  enddo
  use AMREQUIS index AMRSERUP,AMRSERDW,AMRREQDD
  append blank
  replace SERIAL with VSERIAL, NIIN with VNIIN
  @7, 0 clear
  @ 10, 10 say "Requisition serial number " + str(SERIAL)
  @11, 10 say "created for NIIN " + NIIN
  wait
  ***** End NIIN Decision *****
  ***** QUANTITY *****
  close databases
  clear
  @ 1, 10 to 3, 50 double
  @ 2, 25 say "Quantity"
  use AMHODATA index AMANIIN
  seek VNIIN
  @ 5, 10 say " The unit-of-issue for NIIN " + NIIN + ", ";
             + SHORTTITLE + "is: " + UNITOFISSU

  use
  use AMREQUIS index AMRSERUP,AMRSERDW,AMRREQDD
  seek VSERIAL
  store "      " to VQUANTITY
  ?
  ?
  text

```

The quantity must be given as five digits. So if your quantity is less than that, fill in the positions to the left with zeros. (Example: quantity 325 would be given as 00325, 5 would be given as 00005. If you are ordering more than 99999, use an "M" in the rightmost position for thousands and adjust the other numbers accordingly.
 Ref: SPCCINST 8010.12D Para 8-208f.

```

  endtext
  ?
  ?
  @ 22, 15 say "Enter the quantity you desire: " get VQUANTITY;
             picture "!!!!!"
             read
  replace QUANTITY with VQUANTITY
  ***** End Quantity Decision *****
  ***** Julian Date *****
  clear
  @ 1, 10 to 3,50 double
  @ 2, 23 say "Julian Date"
  ?
  ?
  text

```

NOTE: The 4 digit julian date on a requisition that will be transmitted by electronic media (radio) must be equivalent to the date-time-group on the message. so if your radio room or communications center can not transmit the message on the day you think they will, it might be best to leave the date blank below (by just pressing <RETURN>). Or you could put the date in and change it later.

On a manual requisition, DD form 1348, this requirement does not exist, so just put down the date you expect to turn in the requisition.

Julian Date Format: (YDDD)

Y - last digit of current year (ex. 1987 - 7)

DDD - julian day of year (use 3M calendar)

Ref: SPCCINST 8010.12D Para 8-208g(3)

```

endtext
?
store " " to VDATE
@ 22, 15 say "Enter the Julian Date or press <RETURN>: ";
      get VDATE picture "XXXX"
      read

replace JULIANDATE with VDATE
*****End Julian Date Decision *****
***** Requisition and Loadout Point *****
clear
@ 1, 10 to 3, 60 double
@ 2, 15 say "Requisition Routing and Loadout Point"
?
?
text
  Naval message requisitions always go to SPCC Mechanicsburg,
  PA., N00104. Requisitions sent via DAAS format always go to
  DAAS (Defense Automated Addressing System), Dayton, OH.
  (Ref. SPCCINST 8010.12D)
  However, fleet instructions vary on who should receive
  requisitions if submitted by message (DAAS incl). Therefore you
  should indicate below who will receive the requisition or be
  the primary action addree on a message. This is necessary to
  assign the proper routing identification code.

  The supplemental address is the location where you plan to
  actually receive the material and load it out. You need to
  fill this in so that the supply system will know where to
  send the material and from which supply point to fill the
  order.

  The next screen will show you the most common supply
  activities and vessels in your area ( and beyond ).
  A complete list is in NAVSUP Pub. 485, App. 7.

endtext
wait
@ 4,0 clear
close database

use AMDADDR index AMDUIC
store " " to VSERVCOD
store " " to VUIC
store " " to VSUPSRCD
store " " to VSUFUIC

goto top

do while .not. eof()
  List next 6 SERVCODE,UIC,ACTIVNAME,LOCATION
  if eof()
    @ 17,10 say "That's all the activities on file."
  endif
  @ 18,5 say "Enter service code of requisition receipt:";
      get VSERVCOD picture "!"
      read
  @ 19,5 say "Enter the UIC of requisition receipt:";
      get VUIC picture "XXXXX"
      read
  @ 20,5 say "Enter service code of supplemental address: ";
      get VSUPSRCD picture "!"
      read
  @ 21,5 say "Enter UIC of supplemental address: ";
      get VSUFUIC picture "XXXXX"
      read
  do while .T.
    store " " to CHOICE
  @ 22,5 say "Want to see more locations or review again?(Y/N)";
      get CHOICE picture "!"
      read

```

```

do case
  case CHOICE = "N"
    exit
  case CHOICE <> "N" .AND. CHOICE <> "Y"
@ 23,5 say "Not a valid choice, press any key:"
  wait " "
  @ 23,1 clear to 23,70
  case CHOICE = "Y"
    @ 4, 0 clear
    exit
  endcase
enddo
if CHOICE = "N"
  exit
endif
if CHOICE = "Y" .AND. eof()
  goto top
@ 4, 0 clear
endif

enddo
use
use AMREQUIS index AMRSERUP,AMRSERDW,AMRREQDD
seek VSERIAL

replace SENDTOSERC with VSERVCOD, SENDTOUIC with VUIC,;
SUPADDSERC with VSUPSRCD, SUPADDUIC with VSUPUIC
*****End Requisition Routing/Loadout Point*****
***** Project Codes *****
clear
@ 1, 10 to 3, 60 double
@ 2, 15 say "      Project Code  "
?
?
text
  Project Codes basically tell what you are requisitioning the
  ammunition for. The most common codes for fleet units are
  shown below, and more complete lists can be found in SPCCINST
  8010.12D, NAVSUP Pub 485 App. 6, and in the System
  Documentation section of this program.
endtext
?
?
wait" Press any key to show project codes:"
@ 4, 0 clear
do while .T.
text
  Code      Project Title
  835      Ammunition requisitioned for the replacement of
           service ammunition that was used during annual or
           fleet exercise training.
  837      Load adjust(shipfill)- Onload/offload of shipfill
           ordnance to/from combatants or MLSF to facilitate
           offload/onload of training ordnance.
  876      Training. Ammunition requisitioned for or turned in
           following annual training or fleet exercise.
  877      Ship Loadout. Ammunition requisitioned to fill ship
           service allowance for ship deployment.
  878      Ammunition Exchange. Ammunition requisitioned and/or
           turned in for exchange due to NAR's, overage
           components, obsolescence, etc.
endtext
wait
@ 4, 0 clear
text

```

```

Code          Project Title
890          New Construction. Initial onload(reqn.) of
             ammunition for newly constructed or activated ships.
891          Ship Overhaul. Down load(turn-in) of ammunition
             prior to entering yard for overhaul and onload
             (reqn.) of such ammunition upon leaving the yard.
892          Ships Restricted Availability. Ammunition off-loaded
             (turned-in) required by entering a restricted
             availability period and the subsequent onload(reqn.)
             of ammunition upon completion.

endtext
wait
@ 4, 0 clear
do while .T.
  store " " to CHOICE
  @ 15,10 say "Do you need to review the codes again?(Y/N)";
  get CHOICE picture "!"
  read
  do case
    case CHOICE = "N"
      exit
    case CHOICE = "Y"
      exit
    case CHOICE <> "N" .AND. CHOICE <> "Y"
      @ 17,10 say "Not a valid choice, try again:"
      wait
      @ 17,1 clear to 17, 70
      @ 18,1 clear to 18, 70
    endcase
  enddo
  if CHOICE = "N"
    exit
  endif
  @ 4, 0 clear
enddo
store " " to VPROJCOD
@ 20, 10 say "Enter the appropriate project code: ";
get VPROJCOD picture "!!!"
read

replace PROJCODE with VPROJCOD
***** End Project Code Decision *****
***** Document Identifier *****
clear
@ 1,10 to 3, 50 double
@ 2, 15 say " Document Identifier"
?
?
text
The document identifier identifies the purpose of the
document rapidly to the data entry personnel at the supply
activities and can be recognized by automated machinery which
processes many supply documents these days. By purpose it is
meant requisition, issue, status, cancellation, etc..
More information and complete lists of document identifiers
are contained in: SPCCINST 8010.12D Para. 8-208 2(a) and
NAVSUP Pub 485 App. 4.      (* MANDATORY ITEM *)

endtext
?
?
wait" Press any key to view choices and make selection:"
@ 4, 0 clear
?
text
Doc. Iden.          Meaning
A01                For delivery outside CONUS with NSN

```

```

A04          For delivery outside CONUS with NALC-NIIN
AOA          For delivery within CONUS with NSN
AOD          For delivery within CONUS with NALC-NIIN
* A05        For delivery outside CONUS w/exception data
* AOE        For delivery inside CONUS w/exception data
* -- Can't be used for DAAS   NSN = FSC + NIIN
endtext
?
store "      " to VDOCIDEN
@ 22, 10 say " Enter the appropriate document identifier: ";
           get VDOCIDEN picture "!!!"
           read

replace DOCIDENTIF with VDOCIDEN
***** End - document Identifier *****
***** Media and Status Code *****
clear
@ 1, 10 to 3,60 double
@ 2,15 say " Media and Status Codes "
?
?
text
The Media and Status Code is used by the supply system to
determine what type of status should be given regarding the
processing of the requisition; who should receive this status
and by what kind of communications transmission this status
should come. Complete information on this code is contained
in NAVSUP Pub 485 App. 16. (* MANDATORY *)

endtext
?
?
wait" Press any key to show more amplifying information:"
@ 4, 0 clear
do while .T.
  text
  Definitions:
  (a) EXCEPTION STATUS will be used to request information
  relative to any action taken by the supply source other
  than issue of the material.
  (b) 100% SUPPLY STATUS will be used to request information
  relative to any action taken by the supply source
  including release of material for shipment, but not
  including bill of lading numbers or mode of shipment.
  (c) SHIPMENT STATUS may be used in conjunction with
  exception status or 100% supply status to request positive
  information of shipment, including date of shipment, mode,
  bill of lading, or airway bill number, as applicable.
  Supply status for ammunition requisitions will be provided
  to both the requisitioner and the supplemental addressee,
  provided one is given.
endtext
?
wait" Press any key to view choices: "
@ 4, 0 clear
text
M&S Code          Definition
0                 No status provided
2                 Provide exception status via AUTODIN
3                 Provide exception status by mail
* B               Provide 100% supply and exception status
                  by AUTODIN
C                 Provide 100% supply and exception status
                  by mail

```

```

* K          Provide exception and shipment status
              by AUTODIN
  L          Provide exception and shipment status
              by mail
  M          Provide exception and shipment status
              by message
* S          Provide 100% supply and shipment status
              by AUTODIN
* T          Provide 100% supply and shipment status
              by mail
* -- Required for priorities 01 - 08

endtext
wait
@ 4, 0 clear
  do while .T.
    store " " to CHOICE
    @ 6, 10 say "Do you need to see the definitions again?(Y/N)";
      get CHOICE picture "!"
      read
      do case
        case CHOICE = "Y"
          @ 4, 0 clear
          exit
        case CHOICE = "N"
          exit
        case CHOICE <> "N" .AND. CHOICE <> "Y"
          @ 10,10 say "Not a valid choice:"
          wait
          @ 10,5 clear to 10,70
          @ 11,0 clear to 11,70
        endcase
      enddo
    if CHOICE = "N"
      exit
    endif
  enddo
  store " " to VMEDSTAT
  @ 15, 10 say " Enter the appropriate M&S Code: ";
    get VMEDSTAT picture "!"
    read
  replace MEDIASTAT with VMEDSTAT
  ***** End Media and Status Code Decision *****
  ***** Demand Code *****
clear
@ 1, 10 to 3,50 double
@ 2,15 say " Demand Code "
text
  At last a simple code!
      R - Recurring demands. Use when item requisitioned
          is for shipfill ammunition.
      N - Non-recurring demands. Use when item
          requisitioned is clearly a "one-time" request,
          or is the initial loadout of the ship when
          commissioned.
      Ref: NAVSUP Pub 485 sec. 3023, 3024

endtext
?
?
store " " to VDEMCOD
@ 20,10 say " Enter the appropriate letter: ";
  get VDEMCOD picture "!"
  read
  replace DEMANDCODE with VDEMCOD

```

***** End Demand Code *****

***** Signal Code *****

```
clear
@ 1,10 to 3,50 double
@ 2,15 say " Signal Code"
```

text

The signal code is used to identify the activity to which the material is to be shipped and/or billed.

Ref: NAVSUP Pub 485 App 14 (* MANDATORY *)

Signal Code	Meaning
A	Ship and bill to requisitioner
B	Ship to requisitioner and bill to supplemental address
J	Ship to supplemental address and bill to requisitioner. Use Signal Code J when when the supplemental address is used to denote the loadout activity.
K	Ship and bill to supplemental address

endtext

```
store " " to VSIGCOD
@ 22, 10 say " Enter the Signal Code: " get VSIGCOD picture ""
read
```

replace SIGNALCODE with VSIGCOD

***** End Signal Code *****

***** Priority Code (Designator) *****

```
clear
@ 1,10 to 3,60 double
@ 2,15 say " Priority Code (Designator) "
```

text

The Priority Code (0-15) expresses the relationship between the requisitioner's assigned force/activity designator and the selected urgency of need designator, and determines the time frame within which the requisition will be processed. Basically it determines how big a wig you are and how fast they have to fill your order! If you don't know your unit's assigned force/activity designator check with the Supply Officer. The chart on the next screen will help you determine your priority code. Priority Codes are fully explained in NAVSUP Pub 485 sec. 3045-3049, it is worth reading once. Your unit can generally only use 3 of the Priority Codes since you have one assigned F/AD (Force/Activity Designator).

endtext

wait " Press any key to see the PD table:"

```
@ 4, 0 clear
```

text

Priority Code Table

Urgency of Need Designator |

Force/Activity Designator

		I	II	III	IV	V
A (Unable to perform)	01	02	03	07	08	
B (Performance Impaired)	04	05	06	09	10	
C (Routine)	11	12	13	14	15	

endtext

```

store " " to VPRIRCOD
@ 20 , 10 say "Select an appropriate priority code: ";
      get VPRIRCOD picture "XX"
      read

      replace PRIORITYCD with VPRIRCOD
***** End Priority Code *****
*****Required Delivery Date (RDD) *****
clear
@ 1, 10 to 3, 60 double
@ 2, 15 say " Required Delivery Date (RDD) "
?
?
text
  The RDD is the date that you require the material onboard
  and/or the date of the loadout. It is the 3 digit Julian
  day of the year (DDD). They figure out the year from the
  rest of the requisition!      (* MANDATORY *)
endtext
?
store " " to VJDATE
@ 20, 10 say " Enter the Required Delivery Date: ";
      get VJDATE picture "XXX"
      read

      replace REQDELDATE with VJDATE
*****End Required Delivery Date *****
***** Advice Codes *****
clear
@ 1,10 to 3,50 double
@ 2,15 say " Advice Codes "
?
?
text
  An advice code may be entered on the requisition to provide
  the supply source with special insrtuctions to ensure
  appropriate supply action is taken. The codes listed on
  the next screen are the only advice codes that may normally
  be used by Navy units to order ammunition. SPCCINST
  8010.12D lists a few others that may be used when
  authorized by higher authority. NAVSUP Pub 485 App. 1 gives
  complete information on Advice Codes, but note that only
  those listed in the SPCC inst. may be used for ammunition.
  Advice code may be left blank.
endtext
@ 20, 10 say "Press any key to view choices-- "
wait " "
@ 4, 0 clear
text
  Code                Description
  2B                  Do not substitute/interchange. Requested
                    item only will suffice.
  2C                  Do not back order. Reject all unfilled
                    quantities not available to meet RDD.
                    Suitable substitute acceptable.
  2D                  Furnish exact quantity requested (do not
                    adjust to unit pack quantity)
  2J                  Do not substitute or backorder.
  2T                  Deliver to the requisitioner by the RDD
                    or cancel requirement.
endtext
?
store " " to VADVCOD
@ 22, 10 say " Enter Advice Code (if desired): " get VADVCOD;
      picture "!!"

```

```

read
replace ADVICCODE with VADVCOD
***** End Advice Code *****
***** Requisition Status *****
clear
@ 0,10 to 2,55 double
@ 1,15 say " Requisition Status "
?
text
Requisition Status indicates the degree to which a
requisition is complete, ready for submission, submitted but
unfilled, etc.. It is a code established by this system
(SAMS), and not the Navy supply system. The Navy supply
system has many status codes of its own. This requisition
status is for your use in managing your conventional
ammunition requisitions, receipts, etc.. It will be used in
SAMS in various ways.
NOTE: The only item that may be left blank on a requisition
you submit is the Advice Code. Therefore it is a good
habit to simply assign appropriate values to all items
that this program has requested.
The next couple of screens will show you all the selections
that you have made, the exact meaning of the status codes,
and ask you to assign one. All the values shown have been
written to the file, so in order to change any values we'll
finish this part of the program and select the edit option
from the "Requisition Management" menu, if desired.
endtext
wait
@ 3, 0 clear
@ 4,10 say "Requisition Values Assigned"
@ 7, 10 say "Serial Number = " + str(SERIAL)
@ 8,10 say "NIIN = " + NIIN
@ 9, 10 say "Quantity = " + QUANTITY
@ 10, 10 say "Julian Date = " + JULIANDATE
@ 11,10 say "Send to service code = " + SENDTOSERC
@ 12,10 say "Send to UIC = " + SENDTOUIC
@ 13,10 say "Supplemental address service code = " + SUPADDSERC
@ 14,10 say "Supplemental address UIC = " + SUPADDUIC
@ 15, 10 say "Project Code = " + PROJCODE
@ 16, 10 say "Document Identifier = " + DOCIDENTIF
@ 17,10 say "Media and Status Code = " + MEDIASTAT
@ 18, 10 say "Demand Code = " + DEMANDCODE
@ 19, 10 say "Signal Code = " + SIGNALCODE
@ 20, 10 say "Priority Code = " + PRIORITYCD
@ 21, 10 say "Required Delivery Date = " + REQDELDATE
@ 22, 10 say "Advice Code = " + ADVICCODE
?
wait "Press any key to see codes and assign one: "
@ 3, 0 clear
text
Code           Meaning
I             Incomplete. Some mandatory fields are missing
              or you are not ready submit it.
R             Ready. Requisition is ready for submission.
U             Unfilled. Requisition has been submitted but
              is unfilled.
P             Partial. Requisition has been submitted and is
              partially filled.
F             Filled. Requisition fully filled.
C             Cancelled.
endtext
store " " to VREQSTAT

```

```

@ 22, 10 say " Enter Requisition Status code: " get VREQSTAT;
           picture "!!!"
           read

           replace REQUISSTAT with VREQSTAT
wait"Press any key to return to Requisition Management menu:"
***** End Requisition Status *****
***** End CRENEWRO.PRG *****
close databases
set bell on
return

```

5. DATAELEM.PRG

```

* DATAELEM.PRG
* This program allows the user to review the Data Element
* Dictionary and print it if desired.
* Written by LT. Steven L. Smith, USN 31 July, 1987
* Activate next two items if program used alone.
set talk off
set status off
set bell off
do while .T.
clear
@ 5,15 say "SAMS Data Element Dictionary"
@ 4,10 to 6,48
?
?
text
           What would you like to do?
           1. Review the Data Element Dictionary
           2. Print the Data Element Dictionary
           3. Quit
endtext
store 0 to GITEM
@ 20,10 say "Enter Choice: " get GITEM picture "9" range 1,3
           read
do case
case GITEM = 1
@ 7,0 clear
use DATAELEM index DATANAME
do while .T.
@ 8,2 say "DEN:           "+DEN
@ 9,2 say "NAME:          "+NAME
@ 10,2 say "LONG TITLE:   "+LONGTITLE
@ 11,2 say "PIC:             "+PICTURE
@ 12,2 say "DESC:           "+trim(DESCRIPTIO)
@ 18,2 say "USED IN:        "+trim(SOURCE_FIL)
@ 19,2 say "REFS:           "+trim(REFERENCE)
skip
if eof()
@ 23,60 say "End of File"
endif
do while .T.
store " " to XCH
@ 22,5 say "(C)Continue (R)Repeat (X)Exit: ";
get XCH picture "!"
read
do case
case XCH = "C" .OR. XCH = "R"

```

```

        if eof()
            goto top
        endif
        @ 7,0 clear
        exit
        case XCH = "X"
            use
            @ 7,0 clear
            exit
        otherwise
            @ 23,5 say "Invalid selection, Press a key--"
            wait" "
            @ 22,0 clear
        endcase
    enddo
    if XCH = "X"
        exit
    endif
enddo

case GITEM = 2
    clear
    @ 10,10 say "Ensure printer is on, and press a key to start:"
    wait" "
    clear
    @ 12,20 say "Printing, do not disturb"
    @ 11,15 to 13,50 double
    use DATAELEM index DATANAME
    set device to print
    @ 1,10 say "SAMS Data Element Dictionary"+chr(10)
    @ prow(),0 say "_____";
    @ prow(),0 say "_____";
    @ prow(),0 say "_____";
    do while .NOT. eof()
        @ prow()+2,2 say "DEN:      "+DEN+chr(10)
        @ prow(),2 say "NAME:      "+NAME+chr(10)
        @ prow(),2 say "LONG TITLE: "+(LONGTITLE);
        +chr(10)
        @ prow(),2 say "PIC:       "+PICTURE +chr(10)
        @ prow(),2 say "DESC:      "+trim(DESCRPTIO);
        +chr(10)
        @ prow()+6,2 say "USED IN:   "+trim(SOURCE_FIL);
        +chr(10)
        @ prow(),2 say "REFS:      "+trim(REFERENCE);
        +chr(10)
        @ prow(),0 say "_____";
        +chr(10)
    enddo
    skip
enddo
use
set device to screen

case GITEM = 3
    clear
    set talk on
    set status on
    set bell on
    return
endcase
enddo

```

return

6. EDITREQ.PRG

```
* EDITREQ.PRG
* This program allows editing and deleting of certain
* requisitions
* Written by LT. Steven L. Smith, USN 16 June, 1987
do while .T.
  clear
  close databases
  @ 1, 10 to 3, 60 double
  @ 2, 15 say "Editing and Deleting Requisitions"
  ?
  ?
  text
      What would you like to do?
      1. Edit or Change
      2. Delete
      3. Return to previous menu
      NOTE: For obvious reasons you can only delete or edit a
      requisition that has not been submitted. Therefore this
      program will check to make sure that the requisition
      you select has a status code of (I)Incomplete or
      (R)Ready.
  endtext
  store 0 to CHOICE
  @ 20, 20 say "Enter your choice: " get CHOICE picture "9";
  range 1,3
  read
  if CHOICE = 3
    return
  else
    @ 4, 0 clear
    store 0 to NUMBER
    @ 8, 10 say "What is the serial number of the"
    @ 9, 10 say "requisition that you want to edit"
    @ 10,10 say "or delete?" get NUMBER picture "9999";
    range 8000,8999
    read
    select A
      use AMREQUIS index AMRSERUP,AMRSERDW,AMRREQDD
    select B
      use AMMODATA index AMANIIN
    select A
    set relation to NIIN into AMMODATA
    seek NUMBER

    if found()
      @ 12,10 say "Requisition "+ltrim(str(NUMBER))+ " is for:"
      @ 13,10 say B->SHORTTITLE
      @ 14,10 say " NALC: "+B->NALC+" NIIN: "+NIIN+" QUANTITY: ";
      +QUANTITY
      do while .T.
        store " " to MCHOICE
        @ 16,15 say "Is this the correct item?(Y/N)";
        get MCHOICE picture "!"
        read
        do case
```

```

        case MCHOICE = "Y"
            exit
        case MCHOICE = "N"
@ 18,15 say "Return to Requisition Management main"
@ 19,15 say "menu and check serial number with"
@ 20,15 say "option 1 or 2."
?
            wait
            clear
            close databases
            return
        otherwise
@ 18,15 say "Not valid selection, press "
@ 19,15 say "any key to try again:"
            wait " "
            @ 16,0 clear
        endcase
    enddo
    if REQUISSTAT <> "I" .AND. REQUISSTAT <> "R";
        .AND. REQUISSTAT <> " "
@ 18,15 say "That item may not be edited or "
@ 19,15 say "deleted!"
?
        wait
        clear
        close databases
        return
    endif
    if CHOICE = 1
        clear
    text
    WARNING: Do not change the serial number on this
    requisition because this program does not check for
    duplicate numbers like the program that originally
    created it. ( option 3 )
    endtext
    ?
    wait
        clear
        set format to ADDREQUI
        read
        close format
    endif
    if CHOICE = 2
        delete record recno()
        set talk on
        pack
        set talk off
    endif
    else
@ 12,10 say "Requisition "+ltrim(str(NUMBER))+ " is not found "
@ 13,10 say "in the file. Return to the Requisition"
@ 14,10 say " Management main menu and check serial"
@ 15,10 say " number with option 1 or 2."
?
        wait
        clear
        close databases
        return
    endif
endif
enddo

```

7. MISCREQ.PRG

*MISCREQ.PRG

* This program gives miscellaneous information about
*requisitioning.

* Written by LT. Steven L. Smith, USN 14 June, 1987

clear

close databases

@ 1, 3 to 3, 76 double

@ 2, 8 say "Cancellations, Follow-up, Modifications, Misc. Info"

?

text

I. General.

Cancellations, follow-ups and modifications to requisitions are infrequent events that you may someday have to do. Since these are not common, this program only refers you to the publications that deal with them. The SAMS program will still be useful to you in creating these documents.

IMPORTANT ** Note that the fleet commanders and operational commanders promulgate specific instructions concerning ammunition requisitioning and reporting. As weapons personnel you should keep current on these because they deal with your ship's particular area of operations.

Specifically:

endtext

wait

@ 4, 0 clear

?

text

Pacific Fleet - (a) CINCPACFLTINST 8010.12
"Pacific Fleet Conventional Ordnance
Management Manual"
section 1 and appendices 1 - 10
(b) COMSUBPACINST C8500.1
"COMSUBPAC Ordnance Notes"

Atlantic Fleet - (a) CINCLANTFLTINST 8010.4
"Atlantic Fleet Reporting and
Requisitioning Guide"
(b)

endtext

wait

@ 4, 0 clear

?

text

II. Requisition Follow - Up

(a) SPCCINST 8010.12D, Section 8 - 215
(b) NAVSUP Pub. 485, chap. 3, part D, section II,
subsection 1, 3530 - 3537

III. Requisition Modifications

(a) SPCCINST 8010.12D, section 8 - 213, 214
(b) NAVSUP Pub. 485, Chap. 3, part D, section II,
subsection 2, 3550 - 3552

IV. Requisition Cancellation

(a) SPCCINST 8010.12D, section 8 - 216
(b) NAVSUP Pub. 485, chap. 3, part D, section II,
subsection 3, 3565 - 3573

endtext

wait

@ 4, 0 clear

?
text

V. MILSTRIP Status Documents

- (a) SPCCINST 8010.12D, section 8 - 212
- (b) NAVSUP Pub. 485, chap. 3, part D, section I,
3506 - 3511

These documents or messages that the supply system provides you are in response to the status you requested via the Media and Status code on your requisition.

VI. Supplemental Requisition Procedures

- (a) SPCCINST 8010.12D, section 8 - 217

1. 8E cog material - Air launched missile material
(HARPOCN)
2. 8T cog material - Surfaced launched guided
missile material

endtext
wait
@ 4, 0 clear
?
text

3. *4T cog material - Torpedoes and components
ASROC
*8S cog material - SUBROC material
*8U cog material - Sonobouys
* Refers you to Fleet instructions
4. 2D cog material - Tomahawk

endtext
wait "Press any key to return to Requisition Management menu "
return

8. PRINTREQ.PRG

* PRINTREQ.PRG
* Program to print requisition documents
* Written by LT. Steven L. Smith, USN 16 June, 1987

store 1 to REQNO
*set talk on
*set echo on
*set step on
do while .T.
clear
close databases
@ 1, 10 to 3, 60 double
@ 2, 15 say "Printing Requisitions"
?
?
text

What kind of requisition format would you like?

1. Manual (DD Form 1348)
2. Naval message
3. DAAS message
4. Return to previous menu

endtext
store 0 to CHOICE
@ 20, 20 say " Enter your choice: " get CHOICE picture "9";
range 1,4

```

read
  if CHOICE = 4
    return
  endif

```

***** Look up requisition and load data into variables****

```

@ 4, 0 clear
store 0 to NUMBER
@ 8, 10 say "What is the serial number of the"
@ 9, 10 say "requisition that you wish to print?";
      get NUMBER picture "9999" range 8000,8999
read
select A
  use AMREQUIS index AMRSERUP
select B
  use AMMODATA index AMANIIN
select A
set relation to NIIN into AMMODATA
  seek NUMBER
  if found()
@ 12, 10 say "Requisition "+ltrim(str(NUMBER))+" is for:"
@ 13, 10 say B->SHORTTITLE
@ 14, 10 say " NALC: "+B->NALC+" NIIN: "+NIIN+";
      " QUANTITY: "+QUANTITY
      do while .T.
        store " " to MCHOICE
@ 16, 15 say "Is this the correct requisition?(Y/N)";
        get MCHOICE picture "!"
        read
        do case
          case MCHOICE = "Y"
            exit
          case MCHOICE = "N"
@ 18, 15 say "Return to Requisition Management main"
@ 19, 15 say "menu and check serial number with"
@ 20, 15 say " option 1 or 2."
?
            wait
            clear
            close databases
            return
        otherwise
@ 18, 15 say "Not a valid selection, press "
@ 19, 15 say "any key to try again: "
            wait " "
            @ 16, 0 clear
        endcase
      enddo
      if REQUISSTAT = "I"
@ 18, 10 say "WARNING: This item's requisition status "
@ 19, 10 say "indicates it is not ready for submission. "
@ 20, 10 say "Recommend checking an/or updating status."
        wait
        clear
        close databases
        return
      endif

store B->FEDSUPCLAS to APROD
store B->COGSYMBOL to BPROD

```

```

store B->UNITOFISSU to CPROD
store B->SECRISKCOD to XPROD..
set relation to
store SENDTOUIC to DPROD
store SUPADDUIC to EPROD
select C
use AMDADDR index AMDUIC
goto top
seek DPROD
    if found()
        store ACTIVNAME to FPROD
        store LOCATION to GPROD
        store HULLNUMBER to YPROD
        store ROUTIDENT to ZPROD
    else
        store " " to FPROD
        store " " to GPROD
        store " " to YPROD
        store " " to ZPROD
    endif
goto top
seek EPROD
    if found()
        store ACTIVNAME to HPROD
        store LOCATION to IPROD
    else
        store " " to HPROD
        store " " to IPROD
    endif

use
select D
use AMSTDATA
store SERVCODE to JPROD
store UIC to KPROD
store ltrim(UNITNAME) to LPROD
store HULLNUMBER to MPROD
store FUNDCODE to NPROD
store MONITACTIV to OPROD
use
select A
endif
if .not. found()
clear
@ 12,15 say "Requisition "+ltrim(str(NUMBER))+" is not"
@ 13,15 say "found in the file. Return to the "
@ 14,15 say "Requisition Management main menu and "
@ 15,15 say "check the serial number with options "
@ 16,15 say " 1 and 2."
?
wait
clear
close databases
return
endif
*****Finished loading requisition data into variables***
***** Process manual requisition DD Form 1348 *****
    if CHOICE = 1
        clear
        @ 1, 15 to 3, 50
        @ 2, 20 say "Manual Requisition"
        ?
text

```

Notes: A DD Form 1348 can not be produced on most common computer printers because of its size and lack of tractor feed holes. In any case you would have to remove the normal paper.

This program will produce a near replica of the completed DD Form 1348 on the screen and then you will have to transfer the exact data to the DD Form 1348(6-part) card.

You should use a black ball point pen, pressing firmly, or use a typewriter set at 10 pitch. Attempt to keep the characters within the "tick-marks". Use 0 (with a slash through) for the number zero.

```
endtext
?
wait
do REQ1348.PRG
@ 20,10 say "Press any key to return-"
wait " "

endif
***** Finished processing manual requisition *****

****Process information common to DAAS and narrative messages *
    if CHOICE = 2 .OR. CHOICE = 3
        clear
        @ 0,15 to 2,60 double
        @ 1,25 say "Special message information"
        ?
        ?
        text
        NOTE: During periods of restricted communications ( ie
        MINIMIZE), message requisitions shall only be transmitted
        for priorities 01-08 requirements.
        endtext
        @ 3,2 to 8,75
        ?

***** special addressing info. for certain COG material
        text
        Comments: The action and info addrees on naval message
        requisitions vary widely depending on the type of material
        (COG), and the theatre of operations. Due to the frequency of
        change and the variability of these addresses, any attempt to
        automate the choice of these would quickly be in error.
        If your requisition involves COG material that falls in
        special categories, the program will warn you and direct you
        to one of the references.
        endtext
        ?
        wait
        @ 3,0 clear

do case
    case (BPROD = "4E") .OR. (BPROD = "8E")
        text
        ** Your requisition is for Air Launched Missile Material
        (COG 4E or 8E), refer to CINCPACFLTINST 8010.12, section
        1, App. 4 or App. 9(HARPOON) or CINCLANTFLTINST 8010.4H
        endtext
        case BPROD = "8T"
            text
            ** Your requisition is for Surfaced Launched Guided
            Missile Material (COG 8T), refer to CINCPACFLTINST
```

8010.12 section 1, App. 7 or CINCLANTFLTINST 8010.4H

```

endtext
case BPROD = "4T"
text
** Your requisition is for torpedoes and components or
ASROC and components (COG 4T), refer to
CINCPACFLTINST 8010.12 section 1, App. 5 or
CINCLANTFLTINST 8010.4H
endtext
case BPROD = "8S"
text
** Your requisition is for SUBROC material (COG 8S),
refer to CINCPACFLTINST 8010.12 section 1, App. 5
or CINCLANTFLTINST 8010.4H
endtext
case BPROD = "8U"
text
** Your requisition is for sonobouy material (COG
8U), refer to CINCPACFLTINST 8010.12 section 1,
App. 8 or CINCLANTFLTINST 8010.4H
endtext
case BPROD = "2D"
text
** Your requisition is for Tomahawk material (COG
2D), refer to CINCPACFLTINST 8010.12, section 1,
App. 10 or CINCLANTFLTINST 8010.4H
endtext
case BPROD = "6T"
text
** Your requisition is for mine material (COG 6T),
refer to CINCPACFLTINST 8010.12 section 1, App. 6
or CINCLANTFLT 8010.4H
endtext
case (BPROD = "2T") .OR. (BPROD = "2E")
do while .T.
store " " to BCHOICE
@ 5,5 say "Is your requisition for mine material?(Y/N)";
get BCHOICE picture "!"
read
do case
case BCHOICE = "Y"
@ 7,0 clear
text
** Refer to CINCPACFLTINST 8010.12 section 1,
App.6 or CINCLANTFLTINST 8010.4H
endtext
exit
case BCHOICE = "N"
@ 4,0 clear
text
** Your requisition is for normal conventional
ammunition (Cog 2T or 2E(air)). Your "normal"
requisition action and info addrees are:

```

	EastPac	Pacific Fleet MidPac	WestPac	Atlantic Fleet
A		Conus Ord. Activity	Ord. Activity	
C		SPCC or DAAS	SPCC or DAAS	
T		(message)	(message)	
I		TYCOM	TYCOM	
N		COMNAVLOGPAC	COMNAVLOGPAC	
F		ISIC	CTF SEVEN THREE	
O		LOADOUT ACTIVITY	ISIC	

LOADOUT ACTIVITY

Ref: CINCPACFLTINST 8010.12 section 1, App. 3
 CINCLANTFLTINST 8010.4H

```

    endtext
    exit
    otherwise
    wait "Not valid selection, press a key"
    @ 5,0 clear
  endcase
enddo
endcase
wait
***** End of special addressing info.

***** Collecting Addresses *****
if REQNO = 1
  store space(35) to ADDRESS1, ADDRESS2, ADDRESS3, ADDRESS4, ;
  ADDRESS5, ADDRESS6, ADDRESS7, ADDRESS8, ADDRESS9

do while .T.
  clear
  @ 1,10 to 3,60 double
  @ 2,15 say "Message Action and Info. Addees"
  ?
  text
  Enter the appropriate addresses in the order you wish them to
  appear on the message. Use the cursor keys to move around and
  edit. Address format: ACTIVITY LOCATION
  endtext
  @ 7,0 to 7,79
    if CHOICE = 2
      ADDRESS1 = "SPCC Mechanicsburg, PA."
    endif
    if CHOICE = 3
      ADDRESS1 = "DAAS Dayton OH."
    endif
    @ 9,15 say "Action Addressee: 1. " get ADDRESS1
    @ 10,15 say " 2. " get ADDRESS2
    @ 11,15 say " 3. " get ADDRESS3
    @ 13,15 say "Info. Addressee: 1. " get ADDRESS4
    @ 14,15 say " 2. " get ADDRESS5
    @ 15,15 say " 3. " get ADDRESS6
    @ 16,15 say " 4. " get ADDRESS7
    @ 17,15 say " 5. " get ADDRESS8
    @ 18,15 say " 6. " get ADDRESS9
  @ 19,0 to 19,78

do while .T.
  store " " to DCHOICE
  @ 20,2 say "(R)Review address file (S)Save (M)Modify;
  addees (X)Exit"
  @ 21,10 say "Enter choice: " get DCHOICE picture "!"
  read
  do case
    case DCHOICE = "R"
      do REWADD
      select A
      exit
    case DCHOICE = "M"
      exit
    case DCHOICE = "S"

```

```

        exit
        case DCHOICE = "X"
        exit
        otherwise
        @ 22,10 say "Invalid selection.."
        wait
        @ 20,0 clear
        endcase
    enddo
    if DCHOICE = "S" .OR. DCHOICE = "X"
        exit
    endif
enddo

    if DCHOICE = "X"
        loop
    endif
endif

*****End Collecting Addresses*****
*****Classification Determination*****
    if REQNO = 1
        clear
        store " " to CLASS
@ 0,10 to 2,60 double
@ 1,15 say "Message classification information"
?
text
    A DAAS formatted message is always UNCLASSIFIED because none
of the MILSTRIP data elements contain classified
information.
    NOTE: Other technical or operational information about a
particular item may be classified however, such as
for torpedoes, some missiles and rockets,etc..

    A narrative message is likewise UNCLASSIFIED unless you
must add a REMARKS paragraph that contains classified
information such as ship's schedule or other operational
information.
    NOTE: CINCLANTFLT does not permit classified requisitions.
A separate classified message is required.
endtext

do while .T.
    store " " to ECHOICE
    @ 19,5 say "Will you require a classified REMARKS paragraph;
?(Y/N)" get ECHOICE picture "!"
    read
        if ECHOICE = "Y"
            @ 22,10 say "Enter the classification of the remarks:";
            get CLASS picture "!!!!!!!!!!!!!"
            read
            exit
        else
            if ECHOICE = "N"
                CLASS = "UNCLAS"
                exit
            else
                @ 22,10 say "Invalid entry, press any key.."
                wait " "
                @ 19,0 clear
            endif
        endif
    enddo
endif
***** End Classification Determination *****

```

```
***** End of common information *****
endif
```

```
***** Printing the message worksheet *****
if CHOICE = 2
```

```
  if REQNO = 1
```

```
  clear
```

```
  @ 5,5 say "Ensure the printer is on and the paper aligned to"
```

```
  @ 6,5 say "start printing at the top of a sheet"
```

```
  ?
```

```
  wait"Press any key to start printing"
```

```
  clear
```

```
  @ 10,10 say "Printing message requisition"
```

```
  @ 11,10 say "Please do not disturb until return to menu.."
```

```
  @ 9,5 to 12,60 double
```

```
  set device to print
```

```
  @ 1,5 say "Ammunition Requisition Message Worksheet"
```

```
  @ 2,5 say "Security classification = "+CLASS
```

```
  @ 3,10 say "LMF = TT          CIC = ZYUW"
```

```
  @ 4,0 say "-----"
```

```
  @ 6,12 say "From: "+LPROD
```

```
  @ 7,14 say "To: "+ADDRESS1
```

```
    if ADDRESS2 <> " "
```

```
      @ prow()+1,18 say ADDRESS2
```

```
    endif
```

```
    if ADDRESS3 <> " "
```

```
      @ prow()+1,18 say ADDRESS3
```

```
    endif
```

```
  @ prow()+1,12 say "Info: "+ADDRESS4
```

```
    if ADDRESS5 <> " "
```

```
      @ prow()+1,18 say ADDRESS5
```

```
    endif
```

```
    if ADDRESS6 <> " "
```

```
      @ prow()+1,18 say ADDRESS6
```

```
    endif
```

```
    if ADDRESS7 <> " "
```

```
      @ prow()+1,18 say ADDRESS7
```

```
    endif
```

```
    if ADDRESS8 <> " "
```

```
      @ prow()+1,18 say ADDRESS8
```

```
    endif
```

```
    if ADDRESS9 <> " "
```

```
      @ prow()+1,18 say ADDRESS9
```

```
    endif
```

```
  @ prow()+2,0 say trim(CLASS)
```

```
  @ prow(),pcol()+1 say "//8012//"
```

```
  @ prow()+1,0 say "Subj: AMMO MILSTRIP REQUISITION"
```

```
  endif
```

```
  if REQNO > 1
```

```
    clear
```

```
    @ 10,15 say "Printing, do not disturb"
```

```
    @ 9,5 to 11,40 double
```

```
    set device to print
```

```
  endif
```

```
  @ prow()+2,0 say ltrim(str(REQNO)) +". "
```

```
    if CLASS <> "UNCLAS"
```

```
      @ prow(),pcol() say "(U) "
```

```
    endif
```

```
  @ prow(),pcol() say DOCIDENTIF+"/"+"ZPROD +"/"+"MEDIASAT +"/"
```

```
    store 1 to MARK
```

```
    store " " to temp1
```

```
    store " " to temp2
```

```
    store " " to POSIT
```

```

templ = ltrim(str(APROD))
do while MARK < 5
  temp2 = substr(templ ,MARK, 1)
  do SPELLIT with temp2, POSIT
  @ prow(),pcol() say ltrim(POSIT) + " "
  MARK = MARK + 1
enddo
store 1 to MARK
do while MARK < 10
  temp2 = substr(NIIN, MARK, 1)
  do SPELLIT with temp2, POSIT
  if MARK = 7
    @ prow()+1,0 say ltrim(POSIT) + " "
  endif
  @ prow(),pcol() say ltrim(POSIT) + " "
  MARK = MARK + 1
enddo
@ prow(),pcol()-1 say "/" + CPROD + "/"
store 1 to MARK
do while MARK < 6
  temp2 = substr(QUANTITY, MARK, 1)
  do SPELLIT with temp2, POSIT
  @ prow(),pcol() say ltrim(POSIT) + " "
  MARK = MARK + 1
enddo
@ prow(),pcol()-1 say "/" + JPROD + KPROD + "/" + JULIANDATE + "/" + ;
ltrim(str(SERIAL)) + "/" + DEMANDCODE + "/" + chr(10)
@ prow(),0 say SUPADDSERC + SUPADDUIC + "/" + ;
+ SIGNALCODE + "/" + NPROD + "/" + OPROD + BPROD + ;
"/" + PROJCODE + "/" + PRIORITYCD + "/" + REQDELDATE + "/" + ADVICECODE + chr(10)
set device to screen
clear
**** Determine if more messages will be printed
text
  Choose one:  1. More message requisitions to print with
               same addresses and classification.
               2. More message requisitions to print with
               different addresses and/or classification.
               3. Add narrative remarks paragraph
               4. Finished message requisitions
endtext
store 0 to GCHOICE
@ 12,15 say "Enter Choice: " get GCHOICE picture "9" range 1,4
read
do case
  case GCHOICE = 1
    REQNO = REQNO + 1
    clear
    @ 5,5 say "Do not advance printer, next"
    @ 6,5 say "requisition will print as paragraph"
    @ 7,5 say "2,3,etc.."
    ?
    wait
  case GCHOICE = 2
    clear
    set device to print
    @ prow()+2,0 say "BT"+chr(10)
    set device to screen
    @ 5,5 say "Remove previous message worksheet"
    @ 6,5 say "from printer, set up for the next"
    @ 7,5 say "printed message worksheet"
    ?

```

```

        REQNO = 1
        wait
        case GCHOICE = 3
            store space(254) to COMMENTS
            clear
            @ 5,5 say "Enter remarks: " get COMMENTS
            read
            set device to print
            @ prow()+2,0 say ltrim(str((REQNO +1)))+". "
            if CLASS = "CONFIDENTIAL"
                @ prow(),pcol() say "(C) "
            endif
            if CLASS = "SECRET"
                @ prow(),pcol() say "(S) "
            endif
            @ prow(),pcol() say "Remarks: "+ trim(COMMENTS)
            @ prow()+2,0 say "BT"+chr(10)
            set device to screen
            REQNO = 1
            clear

        case GCHOICE = 4
            REQNO = 1
            clear
            set device to print
            @ prow()+2,0 say "BT"+chr(10)
            set device to screen

    endcase
endif

**** Process DAAS formatted message ****
if CHOICE = 3
    if REQNO = 1
        clear
        @ 0,15 to 2,50 double
        @ 1,20 say "DAAS message information"
        ?
        text
        The Defense Automated Addressing System(DAAS) is a real-time
        telecommunications system located in Dayton, OH. which is
        designed to effectively route logistics traffic to supply
        sources. DAAS messages are submitted in a fixed, machine -
        readable format which does not have to be transcribed for
        entry into the CAIMS sytem as do narrative messages or manual
        requisitions.
        ***** DAAS MESSAGES *****
        1. Must be UNCLASSIFIED.
        2. Must not require REMARKS.
        3. Must be to CONUS activities only.
        4. Must not be for CV loadouts from AOE/AE.
        endtext
        @ 10,5 to 16,62 double
        text
        For more information on DAAS read: NAVSUP Pub 485,section
        3028, SPCCINST 8010.12D, section 8-207, CINCPACFLTINST,
        section 1-5 or CINCLANTFLTINST 8010.4H, section
        endtext
    do while .T.
        store " " to HCHOICE
        @ 21,5 say "Is DAAS format still O.K.?(Y/N)" get HCHOICE;
        picture "!"
        read
        do case

```

```

        case HCHOICE = "Y"
            clear
            exit
        case HCHOICE = "N"
            clear
            exit
        otherwise
            @ 23,10 say "Invalid entry, press any key.."
            wait" "
            @ 21,0 clear
        endcase
    enddo
    if HCHOICE = "N"
        loop
        endif
    @ 5,5 say "Ensure the printer is on and the paper aligned"
    @ 6,5 say "to start printing at the top of a sheet."
    ?
    wait"Press any key to start printing"
    clear
    @ 10,10 say "Printing DAAS message requisition--"
    @ 11,10 say "Please do not disturb until return to menu."
    @ 9,5 to 12,60 double
    set device to print
    @ 1,5 say "DAAS Requisition message worksheet"
    @ 2,5 say "Security classification = UNCLAS"
    @ 3,10 say "LMF = TT          CIC = ZYUW "
    @ 4,0 say " _____ "
    @ 6,12 say "From: "+LPROD
    @ 7,14 say "To: "+ADDRESS1
        if ADDRESS2 <> " "
            @ prow()+1,18 say ADDRESS2
        endif
        if ADDRESS3 <> " "
            @ prow()+1,18 say ADDRESS3
        endif
    @ prow()+1,12 say "Info: "+ADDRESS4
        if ADDRESS5 <> " "
            @ prow()+1,18 say ADDRESS5
        endif
        if ADDRESS6 <> " "
            @ prow()+1,18 say ADDRESS6
        endif
        if ADDRESS7 <> " "
            @ prow()+1,18 say ADDRESS7
        endif
        if ADDRESS8 <> " "
            @ prow()+1,18 say ADDRESS8
        endif
        if ADDRESS9 <> " "
            @ prow()+1,18 say ADDRESS9
        endif
    @ prow()+1,0 say "Subj: AMMO MILSTRIP REQON."
endif
    if REQNO > 1
        clear
        @ 10,15 say "Printing, do not disturb."
        @ 9,10 to 11,45 double
        set device to print
    endif
    if REQNO = 1
        @ prow()+4,5 say DOCIDENTIF+ZPROD+MEDIASAT+;
        ltrim(str(APROD))+NIIN+" "+CPROD+QUANTITY+JPROD+KPROD+;

```

```

JULIANDATE+ltrim(str(SERIAL))+DEMANDCODE+SUPADDSERC+;
SUPADDUIC+SIGNALCODE+NPROD+OPROD+BPROD+PROJCODE+;
PRIORITYCD+REQDELDATE+ADVICECODE+chr(10)
endif
if REQNO > 1
    @ prow()+1,5 say DOCIDENTIF+ZPROD+MEDIASAT+;
ltrim(str(APROD))+NIIN+" "+CPROD+QUANTITY+JPROD+KPROD+;
JULIANDATE+ltrim(str(SERIAL))+DEMANDCODE+SUPADDSERC+;
SUPADDUIC+SIGNALCODE+NPROD+OPROD+BPROD+PROJCODE+;
PRIORITYCD+REQDELDATE+ADVICECODE+chr(10)
endif

set device to screen
clear
***** Determine if more requisitions on same message**
text
Choose one:
    1. More requisitions to print with same
        action and info. addresses.
    2. More requisitions to print with different
        action and/or info. addresses.
    3. Finished DAAS message requisition.
endtext
store 0 to ICHOICE
@ 12,15 say "Enter choice: " get ICHOICE picture "9" ;
                                range 1,3
read
do case
    case ICHOICE = 1
        REQNO = REQNO + 1
        clear
        @ 5,5 say "Do not advance printer, next"
        @ 6,5 say "requisition will print below"
        @ 7,5 say "previous one."
        ?
        wait
    case ICHOICE = 2
        clear
        @ 5,5 say "Remove worksheet from printer,"
        @ 6,5 say "set up paper for next message."
        ?
        REQNO = 1
        wait
    case ICHOICE = 3
        clear
        REQNO = 1
endcase
endif

***** End DAAS formatted message *****
enddo
close databases
clear all
return

```

9. PROGFILE.PRG

```
* PROGFILE.PRG
* This program reviews the system program file and prints it if
* desired.
* Written by LT. Steven L. Smith, USN 13 July, 1987
* Activate next two items if program used alone.
set talk off
set status off
set bell off

clear
@ 5,15 say "SAMS Program File"
@ 4,10 to 6,45 double
?
?
text
        What would you like to do?
                1. Review the program file
                2. Print the program file
                3. Quit

endtext
store 0 to ITEM
@ 20,10 say "Enter choice: " get ITEM picture "9" range 1,3
        read

do case
        case ITEM = 1
                clear
                use PROGFILE index PROGNAME
                do while .T.

                        store 1 to MLINE
                        store 1 to MCOUNT
                        do while (MCOUNT <=3) .AND. (.NOT. eof())
                                @ MLINE,5 say "Program Name: "+PROG_NAME
                                @ MLINE+1,5 say "Calls: "+rtrim(CALLS)
                                @ MLINE+3,5 say "Purpose: "+rtrim(PURPOSE)
                                @ MLINE+5,5 say "Called by: "+rtrim(CALLED_BY)
                                @ MLINE+6,0 to MLINE+6,78
                                MLINE = MLINE + 7
                                MCOUNT = MCOUNT + 1
                                skip
                        enddo
                        if eof()
                                @ 23,60 say "End of File"
                        endif
                        do while .T.
                                store " " to CHY
                                @ 22,5 say "(C)Continue (R)Repeat (X)Exit: ";
                                get CHY picture "!"
                                read
                                do case
                                        case CHY = "C" .OR. CHY = "R"
                                                if eof()
                                                        goto top
                                                endif
                                                exit
                                        case CHY = "X"
                                                use
                                                clear
                                                set talk on
                                                set status on
                                                set bell on
                                                return
                                enddo
                        enddo
                enddo
        enddo
```

```

        otherwise
        @ 23,5 say "Invalid selection, press a key-"
            wait " "
            @ 22,0 clear
        endcase
    enddo
    clear
enddo
    case ITEM = 2
    clear
    @ 10,10 say "Printing, do not disturb"
    @ 9,5 to 11,37 double
    use PROGFILE index PROGNAME
    set device to print
    @ 1,15 say "SAMS Program File"+chr(10)
    @ prow(),0 say "_____";
    @ prow(),0 say "_____";
    @ prow(),0 say "_____";
    do while .NOT. eof()
    @ prow()+2,5 say "Program Name: "+PROG_NAME+chr(10)
    @ prow(),5 say "Calls: "+CALLS+chr(10)
    @ prow(),5 say "Purpose: "+PURPOSE+chr(10)
    @ prow(),5 say "Called by: "+CALLED_BY+chr(10)
    @ prow(),0 say "_____";
    skip
    enddo
    use
    set device to screen

    case ITEM = 3
    clear
    @ 10,10 say "Quit this program"
    use
    wait
    set talk on
    set status on
    set bell on
    return

    endcase
set talk on
set status on
set bell on
return

```

10. REQ1348.PRG

```

* REQ1348.PRG
* Program to display replica of DD Form 1348 filled in
* Written by LT. Steven L. Smith, USN 16 June, 1987
clear

```

```

@ 1, 3 SAY SENDTOSERC
@ 1, 4 SAY SENDTOUIC
@ 1, 10 SAY FPROD
if SENDTOSERC = "V" .OR. SENDTOSERC = "R"
    @ 2, 13 say YPROD
else

```

```

@ 2, 12 say GPROD
endif
@@ 1, 40 SAY JPROD
@@ 1, 41 SAY KPROD
@@ 1, 48 SAY LPROD
@@ 2, 45 SAY MPROD
@@ 5, 2 SAY "XXXXXXXXXXXXXXXXXXXXXXXXX"
@@ 5, 25 SAY AMREQUIS->DOCIDENTIF
@@ 5, 30 SAY ZPROD
@@ 5, 36 SAY AMREQUIS->MEDIASAT
@@ 5, 40 SAY ltrim(str(APROD))
@@ 5, 49 SAY AMREQUIS->NIIN
@@ 5, 62 SAY "XX"
@@ 5, 67 SAY CPROD
@@ 5, 71 SAY AMREQUIS->QUANTITY
@@ 8, 2 SAY JPROD
@@ 8, 3 SAY KPROD
@@ 7, 40 SAY "Remarks:"
@@ 8, 11 SAY AMREQUIS->JULIANDATE
@@ 8, 17 SAY AMREQUIS->SERIAL
@@ 8, 23 SAY AMREQUIS->DEMANDCODE
@@ 8, 27 SAY AMREQUIS->SUPADDSERC
@@ 8, 28 SAY AMREQUIS->SUPADDUIC
@@ 8, 36 SAY AMREQUIS->SIGNALCODE
@@ 11, 2 SAY NPROD
@@ 11, 6 SAY OPROD
@@ 11, 7 SAY BPROD
@@ 11, 10 SAY AMREQUIS->PROJCODE
@@ 11, 14 SAY AMREQUIS->PRIORITYCD
@@ 11, 17 SAY AMREQUIS->REQDELDATE
@@ 11, 21 SAY "XXXXXXXXXXXXXXXXXXXXXXXXX"
@@ 14, 2 SAY AMREQUIS->ADVCECODE
@@ 14, 6 SAY "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@@ 16, 16 SAY "DD Form 1348 - Manual Requisition"
@@ 3, 2 TO 3, 77
@@ 6, 2 TO 6, 77
@@ 9, 2 TO 9, 77
@@ 12, 2 TO 12, 77
@@ 0, 1 TO 15, 78 DOUBLE
@@ 1, 33 TO 5, 38
@@ 7, 38 to 11, 38 double
@@ 4, 35 TO 5, 35
@@ 4, 29 TO 5, 29
@@ 4, 24 TO 5, 24
@@ 4, 15 TO 5, 15
@@ 4, 70 TO 5, 70
@@ 4, 65 TO 5, 65
@@ 4, 45 TO 5, 45
@@ 4, 61 TO 5, 61
@@ 7, 35 TO 8, 35
@@ 7, 24 TO 8, 24
@@ 7, 21 TO 8, 21
@@ 10, 20 TO 11, 20
@@ 10, 16 TO 11, 16
@@ 10, 13 TO 11, 13
@@ 10, 9 TO 11, 9
@@ 10, 5 TO 11, 5
@@ 13, 5 TO 14, 5
@@ 13, 33 TO 14, 33 double
@@ 13, 29 TO 14, 29
return

```

11. REQMENU.PRG

*REQMENU.PRG

*Program to present the Requisition Management Menu

*Written by LT. Steven L. Smith, USN , 6 June, 1987

Clear all

do while .T.

clear

text

Requisition Management Menu

1. Review all requisitions - Summary
2. Review complete requisition data
3. Create a new requisition
4. Edit and delete requisitions
5. Cancellation, Follow-up, Modifications, Info.
6. Print requisition documents
7. Backup Requisition File

99. Return to Main Menu

endtext

@ 1,0 to 21,79 double

@ 3,1 to 3,78

@ 18,1 to 18,78

store 0 to MSELECT

@ 19,22 say "Enter your selection: " get MSELECT picture "99"

read

do case

case MSELECT = 1
do REVWREQ.PRG

case MSELECT = 2
do COMPLREQ.PRG

case MSELECT = 3
do CRENEWREQ.PRG

case MSELECT = 4
do EDITREQ.PRG

case MSELECT = 5
do MISCREQ.PRG

case MSELECT = 6
do PRINTREQ.PRG

case MSELECT = 7
do BCKUPREQ.PRG

case MSELECT = 99
return

otherwise

@ 22,16 say "Not a valid selection--"

wait "

Press any key to try again-- "

endcase

enddo [.T.]

clear all

return

12. REWADD.PRG

```
* REWADD.PRG
* This program reviews the address file (library module)
* Written By LT. Steven L. Smith, USN 20 June, 1987
* NOTE: If used external to AMSMAIN, activate next two lines.
* set status off
* set talk off

clear
@ 1,15 to 3,55 double
@ 2,28 say "Address File"
@ 5,0 say "S/C UIC          ACTIVITY          LOCATION          ;
          HULL NO.        R/I"

@ 6,0 to 6,79
select H
use AMDADDR index AMDACTNM

do while .T.
  store 1 to MCOUNT
  store 8 to MLINE
  do while (MCOUNT <= 10) .AND. (.NOT. eof())
    @ MLINE ,1 say SERVCODE
    @ MLINE ,3 say UIC
    @ MLINE ,10 say ACTIVNAME
    @ MLINE ,35 say LOCATION
    @ MLINE ,62 say HULLNUMBER
    @ MLINE ,73 say ROUTIDENT
    MLINE = MLINE + 1
    MCOUNT = MCOUNT + 1
    skip
  enddo
  if eof()
    @ 20,50 say "End of File"
  endif
  do while .T.
    store " " to ZCHOICE
    @ 21,5 say "(C)Continue, (R)Repeat, (X)Exit:" get ZCHOICE;
    picture "!"
    read
    do case
      case ZCHOICE = "C" .OR. ZCHOICE = "R"
        if eof()
          goto top
        endif
        exit
      case ZCHOICE = "X"
        use
        clear
        return
    otherwise
      @ 22,20 say "Invalid selection, press any key to try again-"
        wait " "
        @ 21,0 clear
    endcase
  enddo
  @ 7,0 clear
enddo
use
return
```

13. REVWREQ.PRG

```
* REVWREQ.PRG
* Program to quickly review the requisition file
* Written by LT. Steven L. Smith, USN 13 June, 1987

clear all
select A
  use AMMODATA index AMANIIN
select B
  use AMREQUIS index AMRSERDW,AMRSERUP,AMRREQDD
  set relation to NIIN into AMMODATA

clear

@ 1, 22 to 3, 49 double
@ 2, 27 say "Requisition File"

@ 5,4 say "SERIAL  NALC  NIIN          SHORT TITLE
QUANTITY STATUS  J/DATE "

set heading off
goto top

do while .T.
  @ 1, 0 to 24, 79 double
  @ 6, 1 to 6,78
  store 7 to mline
  store 0 to xcount
  do while (.not. eof()) .AND. (xcount < 10)
    @ mline, 5 say SERIAL
    @ mline, 13 say A->NALC
    @ mline, 19 say NIIN
    @ mline, 30 say A->SHORTTITLE
    @ mline, 55 say QUANTITY
    @ mline, 65 say REQUISSTAT
    @ mline, 73 say JULIANDATE
    mline = mline + 1
    xcount = xcount + 1
  skip
  if eof()
    @ 18,5 say "That's all the requisitions on file:"
  endif
enddo

do while .T.
  store " " to CHOICE
  @ 20,5 say "Want to see more or review again?(Y/N)";
  get CHOICE picture "!"
  read
  do case
    case CHOICE = "N"
      set heading on
      clear all
      return
    case CHOICE = "Y"
      if eof()
        goto top
        @ 6, 0 clear
        exit
      else
        @ 6, 0 clear
        exit
```

```

endif
case CHOICE <> "N" .AND. CHOICE <> "Y"
@21,5 say "Not a valid choice--"
wait"      Press any key to try again-"
@ 19, 1 clear to 23,78
endcase
enddo
enddo
clear all
return

```

14. SPELLIT.PRG

* Procedure SPELLIT
 * This program returns a spelled out character string for the
 * character number
 * Written by LT. Steven L. Smith, USN 20 June, 1987

```

procedure SPELLIT
parameters temp2, POSIT
do case
case temp2 = "9"
store "NINE" to POSIT
case temp2 = "8"
store "EIGHT" to POSIT
case temp2 = "7"
store "SEVEN" to POSIT
case temp2 = "6"
store "SIX" to POSIT
case temp2 = "5"
store "FIVE" to POSIT
case temp2 = "4"
store "FOUR" to POSIT
case temp2 = "3"
store "THREE" to POSIT
case temp2 = "2"
store "TWO" to POSIT
case temp2 = "1"
store "ONE" to POSIT
case temp2 = "0"
store "ZERO" to POSIT
endcase
return

```

15. STRUCCRT.PRG

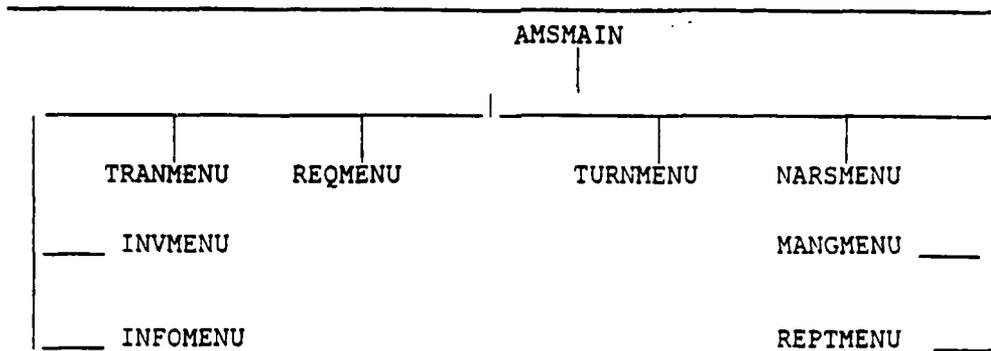
* STRUCCRT.PRG
 * This program displays the SAMS structure charts
 * Written by LT. Steven L. Smith, USN 13 July, 1987

```

clear
set talk off
set status off
@ 10,20 say "SAMS Structure Charts"
@ 8,15 to 12,46 double
@ 20,10 say "Press any key to start viewing charts--"
wait" "
clear
do while .T.
text

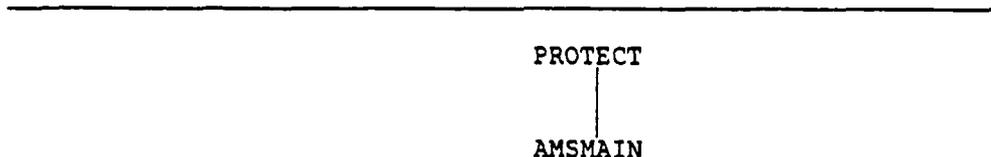
```

Shipboard Ammunition Management System
 Major Sub-system Structure Chart



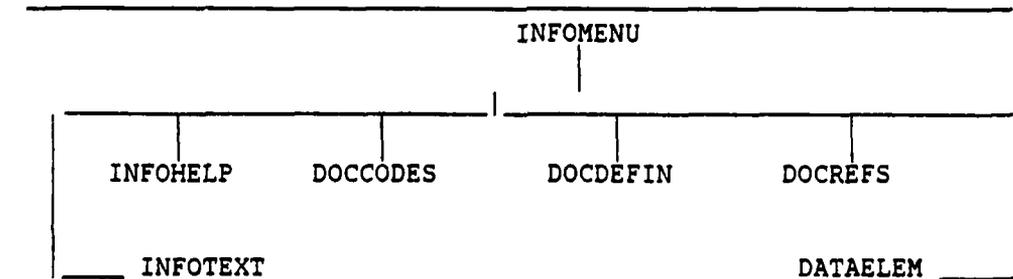
endtext
wait
clear
text

User Access/Main Menu



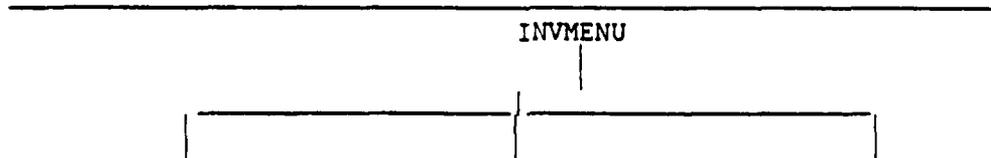
endtext
wait
clear
text

General Information/Documentation



endtext
wait
clear
text

Inventory/Allowance/Ammunition Information



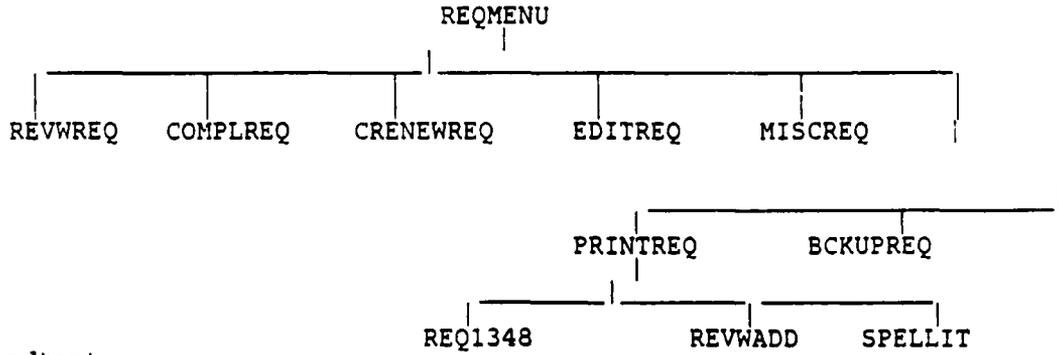
INVVIEW

INVAMMO

INVALLOW

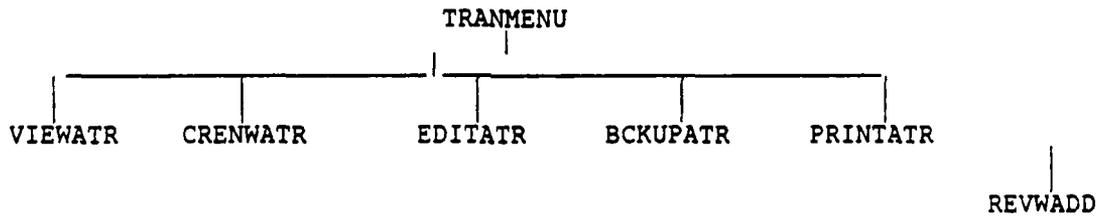
endtext
wait
clear
text

Requisition Management



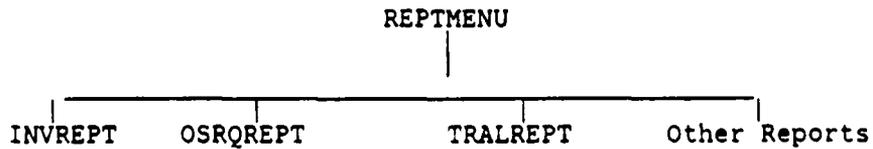
endtext
wait
clear
text

Transaction Management



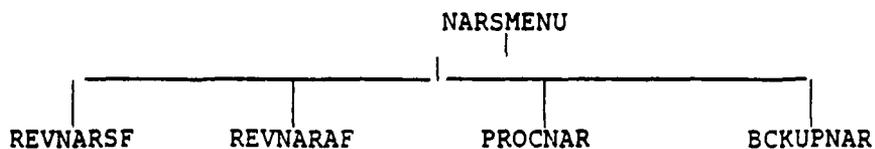
endtext
wait
clear
text

Generate Internal Reports



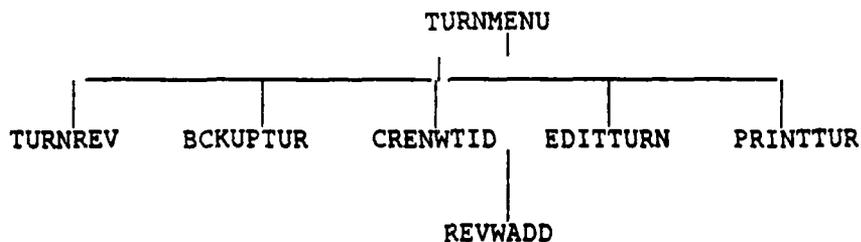
endtext
wait
clear
text

NARS Management



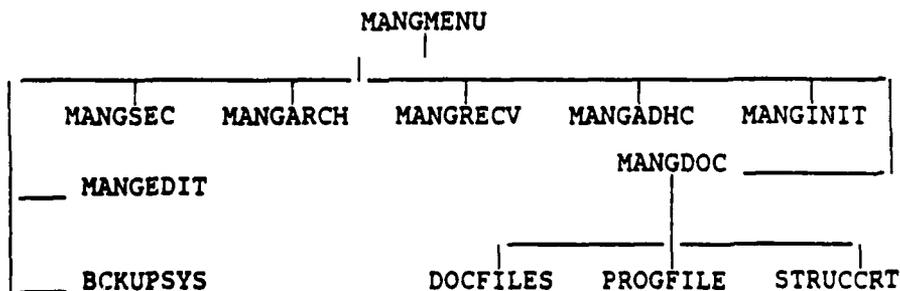
endtext
wait
clear
text

Turn-in Document Management



endtext
wait
clear
text

System Management



endtext
wait
clear

```
do while .T.  
  store " " to XYZ  
  @ 5,5 say "Do you wish to review the charts again?(Y/N)";  
  get XYZ picture "!"
```

```
        read
    do case
        case XYZ = "Y"
            clear
            exit
        case XYZ = "N"
            clear
            set talk on
            set status on
            exit
        otherwise
            @ 10,5 say "Invalid entry, press a key-"
            wait " "
            @ 5,0 clear
        endcase
    enddo
    if XYZ = "N"
        exit
    endif
enddo
return
```

LIST OF REFERENCES

1. Swanson, J.L., "Ammunition Management at SPCC," *Navy Supply Corps Newsletter*, v. 40, n. 7, July 1977.
2. Office of the Chief of Naval Operations, OPNAVINST 8000.13, *Conventional Ammunition Integrated Management System (CAIMS) Objectives and Policies* for, February 1, 1971.
3. Commander, Naval Ships Parts Control Center, SPCCINST 8010.12D, *Conventional Ammunition Integrated Management System (CAIMS)*, November 2, 1984 through change 7, December 17, 1986.
4. Commander in Chief, U.S. Pacific Fleet, CINCPACFLTINST 8010.12, *Pacific Fleet Conventional Ordnance Management Manual*, August 30, 1985.
5. Commander in Chief, U.S. Atlantic Fleet, CINCLANTFLTINST 8010.4H, *Atlantic Fleet Requisitioning and Reporting Guide*, September 25, 1985.
6. U.S. General Accounting Office, PLRD-81-54, *The Navy Must Improve It's Accountability for Conventional Ammunition*, July 29, 1981.
7. U.S. General Accounting Office, PLRD-82-27, *DoD Has Serious Problems With Care and Maintenance of Conventional Ammunition*, February 9, 1982.
8. Naval Audit Service, Audit Report Number T20040, *Multi-Location Audit of the Management of Small Arms and Ammunition Programs in the Navy*, April 1981.
9. Chief of Naval Education and Training, NAVEDTRA 10186-D1, *Gunners Mate G 1&C*, 1980.
10. Commander, Naval Supply Systems Command, NAVSUP Publication 485, *Afloat Supply Procedures*, Revision 1, February 1, 1984 through change 5, February 17, 1987.
11. Department of Defense Handbook H3, *DoD Ammunition Code Cataloging Handbook*, January, 1987.
12. Commander, Naval Ships Parts Control Center, *Stock List of Navy Ammunition*, (microfiche) TWO 10-AA-ORD-010, April 1, 1986.

13. Commander, Naval Sea Systems Command, TWO24-AA-ORD-010, *Ammunition-Unserviceable, Suspended, Limited Use*, October 1, 1986.
14. Commander, Naval Sea Systems Command, NAVSEA OP 2165, Vol. 2, Revision 11, *Navy Transportation Safety Handbook for Ammunition, Explosives, and Related Hazardous Material*, July 1, 1986.
15. Page-Jones, M., *The Practical Guide To Structured Systems Design*, Yourdon Press, Englewood Cliffs, NJ, 1980.
16. Yourdon, E., *Managing the Structured Techniques*, Yourdon Press, New York, NY, 1986.
17. Davis, W.S., *Systems Analysis and Design - A Structured Approach*, Addison - Wesley, Reading, MA, 1983.
18. Kroenke, D., *Database Processing*, Science Research Associates, Chicago, IL, 1983.
19. Date, C.J., *An Introduction to Database Systems Volume 2*, Addison - Wesley, Reading, MA, 1983.
20. Ullman, J.D., *Principles of Database Systems*, Computer Science Press, Potomac, MD, 1980.
21. Brodie, M.L. and Schmidt, J. (editors), *Relational Database Systems*, Springer - Verlag, Berlin, Germany, 1983.
22. Dickson, G.W. and Wetherbe, J.C., *The Management of Information Systems*, McGraw Hill, New York, NY, 1985.
23. Kent, W., "A Simple Guide to Five Normal Forms in Relational Database Theory", *Communications of the ACM*, February, 1983.
24. MacLennan, B.J., *Principles of Programming Languages, Second Edition*, Holt, Rhinehart, and Winston, New York, NY, 1987.

BIBLIOGRAPHY

Government Instructions and Publications:

- Chief of Naval Education and Training, NAVEDTRA 10867-D, *The Weapons Officer*, November, 1985.
- Commander, Naval Supply Systems Command, NAVSUP Publication 508, *Supply Management Program Standard Data Element Dictionary and Keyword Index*, November 25, 1986.
- Commander, Naval Ships Parts Control Center, TWO 10-AA-ORD-010, *Stock List of Navy Ammunition (supplement)*, April 1, 1986.
- Commander, Naval Supply Systems Command, NAVSUP Publication 437, *MILSTRIP/MILSTRAP*, May, 1986 through change 2, November 6, 1986.
- Commander, Naval Supply Systems Command, NAVSUP Publication 553, *Inventory Management*, no date.
- Office of the Chief of Naval Operations, OPNAVINST 4614.1F, *Uniform Material Movement and Issue Priority System (UMMIPS)*, April 15, 1983.
- Office of the Chief of Naval Operations, OPNAVINST 8000.15, *Accountability for Expendable Ordnance*, June 21, 1983.

Other Government Documents and Forms

- Ammunition Lot/Location Card, NSN 0108-LF-501-2970.
- Ammunition Master Stock Record Card, NSN 0108-LF-501-2960.
- Ammunition Serial/Location Card, NSN 0108-LF-501-3560.
- Commander, Naval Supply Systems Command, NAVSUP Form 1353-1, *MILSTRIP Message Transmittal Worksheet*, NSN 0108-LF-501-3638.
- Department of Defense, DD Form 173/2 (OCR), *Joint Messageform*, NSN 0102-LF-000-1735.
- Department of Defense, DD Form 1348, *DoD Single Line Item Requisition System Document*, NSN 0102-LF-001-3490.
- Department of Defense, DD Form 1348-1, *DoD Single Line Item Release/Receipt Document*, NSN 0102-LF-013-1040.

Other Books, Articles, Publications, and Papers

Alderman, B.A., *Software Requirement Specifications for an Ammunition Management System*, Master's Thesis, Naval Postgraduate School, Monterey, CA, September, 1986.

King, Richard Allen. *The MS-DOS Handbook*, SYBEX, Inc., Alameda, CA, 1986.

Liskin, Miriam., *Advanced dBase III Plus Programming and Techniques*, McGraw-Hill, Berkely, CA, 1987.

Simpson, Alan., *Understanding dBase III Plus*, SYBEX, Inc., Alameda, CA, 1986.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Chief of Naval Operations Director, Information Systems (OP-945) Navy Department Washington, D.C. 20350-2000	1
4.	Computer Technology Programs Code 37 Naval Postgraduate School Monterey, Ca 93943-5000	1
5.	Associate Professor Norman R. Lyons Code 54Lb Department of Administrative Sciences Naval Postgraduate School Monterey, Ca 93943-5000	1
6.	Associate Instructor Yehia Mortagy Code 54My Department of Administrative Sciences Naval Postgraduate School Monterey, Ca 93943-5000	1
7.	LT. Steven L. Smith, USN Submarine Officer Advanced Course Naval Submarine School Naval Submarine Base Groton, CT. 06340	2

END

FEB.

1988

DTIC