AD-A246 762

DTIC
ELECTE
MAR 0 3 1992
S B D

SMALL CONTRACT AWARD: IMPROVING THE VENDOR SELECTION PROCESS

THESIS

Daniel E. Hagmaier, B.S.
Captain, USAF

AFIT/GLM/LSC/91S-26

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

92-04843

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

92 2 25 121

AFIT/GLM/LSC/91S-26

SMALL CONTRACT AWARD: IMPROVING THE VENDOR SELECTION PROCESS

THESIS

Daniel E. Hagmaier, B.S.
Captain, USAF
AFIT/GLM/LSC/91S-26

Approved for public release; distribution unlimited

The opinions and conclusions in this paper are those of the author and are not intended to represent the official position of the DOD, USAF, or any other government agency.

DTIC
COPY
INSPECTED
5

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

AFIT/GLM/LSC/91S-26

SMALL CONTRACT AWARD: IMPROVING THE VENDOR SELECTION PROCESS

THESIS

Presented to the Faculty of the School of Systems and Logistics

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Logistics Management

Daniel E. Hagmaier, B.S.

Captain, USAF

September 1991

Approved for public release; distribution unlimited

Acknowledgments

The document you are now reading would not have been produced without the contributions made by many other individuals.

First, I would like to acknowledge the immense cooperation provided by the people that comprise DESC-P and DESC-Z. Without their timely response to requests for information and the time invested to teach me the acquisition world's unique language, successful completion of this effort would not have been possible.

They invested countless hours conveying the nuances and subtleties of their acquisition language so I might have a clearer understanding of their requirements. Not only did they satisfy my thirst for the knowledge of their procedures, but also saw that I did not hunger for data when the time came. Each request for information was handled expeditiously. Again, to all the people at DESC - Thank you!

Secondly, my gratitude and appreciation is extended to my thesis advisor, Lt Col Larry W. Emmelhainz. His guidance and inspiration is woven throughout the pages of this document. He constantly motivated me to strive for excellence in my work. His advice proved to be invaluable, and I highly regard my experiences in working with him.

I would also like to take this opportunity to thank my family. 1991 will be remembered as the year of the lost summer. This project would not be possible if it were not for their unselfish relinquishing of trips to the park, days at the zoo and weekend camping trips.

Finally, most of all, thanks to my typing pool, editor-in-chief, layout department and paste-up crew of one, my wife Dianne. Here's to a new year.

Daniel E. Hagmaier

Table of Contents

Page

Acknowledgments ii

List of Figures ix

List of Tables xii

Abstract xiii

I. Introduction 1-1

Overview 1-1
Background 1-1
About DESC 1-1
Small Contract Procurement Process 1-3
Problems 1-10
Inefficiencies 1-10
Guidance 1-10
VASPP Concept 1-11
Scope of Research 1-12
Specific Problem 1-12
Research Objective 1-12
Research Questions 1-12
Areas of Study 1-13
Method of Organization 1-13

II. Literature Review 2-1

Overview 2-1
Prototyping 2-1
Computer Assistant Systems 2-2
Systems Reviewed 2-2
Database Management Systems (DBMSs) 2-2
Decision Support Systems (DSSs) 2-3
Expert Systems (ESs) 2-4
Applicability 2-7
Synopsis 2-9
System Verification 2-10
Testing vs Debugging 2-10
Testing 2-11
Program Functions 2-11

Page

Test Data 2-11
Identification 2-11
Application 2-13
Approach 2-13
Conclusions 2-14

III. Methodology 3-1

Overview 3-1
Problem Identification 3-1
Methodology 3-1
Decision Criteria 3-2
Proposal Development 3-4
Methodology 3-4
The DSS 3-5
Verification 3-7
Validation 3-7
Phase I. 3-7
Overview 3-7
Procedure 3-8
Phase II. 3-13
Overview 3-13
Procedure 3-13
Phase III. 3-17
Overview 3-17
Procedure 3-17
Conclusions 3-18

IV. Development 4-1

Overview 4-1
Investigative Efforts 4-1
User Interview Process 4-1
Results 4-2
Inputs 4-2
Outputs 4-3
Paper Prototype 4-3
Components 4-3
Inputs 4-3
Outputs 4-4
Review 4-5
Initial Prototype Development 4-5
Design Considerations 4-5
Data Requirements 4-5
Data Files 4-6

Page

Data Structure 4-6
Software Development 4-8
Methodology 4-8
Environment Selection 4-8
System Description 4-9
The Model Component 4-17
Review 4-18
Full Prototype Development 4-18
Requirements Re-evaluation 4-18
Modifications 4-18
Embellishments 4-19
Enhancements 4-21
Customer Depot Complaint File (CDCF) . . 4-21
Required Delivery Date (RDD) 4-21
Data Files 4-24
Design Problems 4-25
Data Structure 4-25
Program Coding 4-26
Verification 4-26
Focus 4-26
Desk-top Review 4-27
Path Verification 4-28
User Review 4-28
Conclusion 4-28

V. Validation 5-1

Overview 5-1
Phase I 5-1
Synopsis 5-1
Testing 5-1
Results 5-3
Analysis 5-5
Phase II 5-7
Synopsis 5-7
Testing 5-7
Phase II - Part 1 5-8
Manual Processing 5-8
Automated Processing 5-8
Phase II - Part 2 5-12
Results 5-12
Error Rate 5-13
Analysis 5-18
Processing Time 5-20

Page

Analysis 5-20
Phase III 5-24
Synopsis 5-24
Testing 5-24
Results 5-24
Error Rate 5-24
Analysis 5-25
Processing Time 5-27
Analysis 5-29
User Comments 5-30
Initial Conclusions 5-32
Short Comings 5-33
PR Testing 5-33
Problem 5-33
Correction 5-33
Results 5-37
Buyers Selected (Phase II Impact) 5-38
Problem 5-38
Correction 5-38
Error Rate 5-38
Processing Time 5-39
Results 5-41
Buyers Selected (Phase III Impact) 5-52
Problem 5-52
Correction 5-52
Error Rate 5-52
Processing Times 5-55
Results 5-57
Conclusions 5-58

VI. Summary, Findings, and Recommendations 6-1

Overview 6-1
Summary of Research 6-1
Findings 6-2
Summary Of Findings 6-5
Recommendations for Future Research 6-5
Recommendation for Future Modification 6-6
Lessons Learned 6-7
Final Notes 6-7

Appendix A: Program Code A-1

AVSA.PRG A-1
PREPVEN.PRG A-7

Page

SCREENS.PRG A-16
SELCTVEN.PRG A-46

Appendix B: Variable Cross Reference Table B-1

Appendix C: Data Base Structure C-1

PR_TEMP.DBF C-1
HOLD.DBF C-2
NSN.DBF C-2
DCRLCODE.DBF C-2
VENDOR.DBF C-3
DCRL.DBF C-4
MODEL.DBF C-4
PRICE.DBF C-5
CDCF.DBF C-6
QUALITY.DBF C-6
HISTORY.DBF C-6
Data Dictionary C-7

Appendix D. Other Program Documentation D-1

File List D-1
Index Parameter Summary D-2
Procedures Summary D-4
Tree Diagram for databases and program files D-5

Appendix E: Variable Descriptions E-1

Appendix F: Data Field Descriptions F-1

Appendix G: Questionnaire Responses G-1

Panel Questionnaire Responses G-1
Buyer Questionnaire Responses G-3

Appendix H: User's Guide . . . . . UG

Table Of Contents . . . . . UG-i
Disclaimer . . . . . UG-ii
Introduction UG-1
System Requirements UG-2
Software Installation UG-3
Starting AVSA UG-4
User's Screens UG-5
Welcome Screen UG-6

Page

Program Information Screen UG-7
Initializing Screen UG-8
NSN Screen UG-9
Quantity Screen UG-11
RDD Screen UG-12
Set-A-Side Screen UG-13
Selecting Vendors Screen UG-14
No Qualified Vendor Screen UG-15
Analyzing Vendor Screen UG-16
Organizing Data Screen UG-17
Unit Pricing Screen UG-18
Extended Pricing Screen UG-20
Vendor Information Screen UG-21
DCRL Screen UG-23
CDCF Screen UG-24
Award Screen UG-26
Exiting The System UG-27
The Model Component UG-28

Bibliography BIB-1

Vita VIT-1

List of Figures

Figure		Page
1-1.	Trends in Item Management	1-2
1-2a.	Sample Purchase Request	1-4
1-2b.	Sample Purchase Request (Continued)	1-5
1-2c.	Sample Purchase Request (Continued)	1-6
1-3a.	Sample Vendor Price Lists	1-7
1-3b.	Sample Vendor Price Lists (Continued)	1-8
1-3c.	Sample Vendor Price Lists (Continued)	1-9
1-4.	The VASPP Concept	1-11
2-1.	Components of an Expert System	2-5
3-1.	Initial Model - The VASPP Concept	3-2
3-2.	Structure of Development Process	3-4
3-3.	Information Flow	3-6
3-4.	Panel Selection Form	3-9
3-5.	DESC Form 701	3-10
3-6.	Panel Questionnaire	3-12
3-7.	Buyer Selection Form	3-14
3-8.	Buyer Questionnaire	3-15
3-9.	Test Consolidation Form	3-16
4-1.	Initial Flow Diagram	4-4
4-2.	Welcome Screen	4-9
4-3.	Program Information Screen	4-10
4-4	NSN Input Screen	4-10
4-5.	NSN Not On File Screen	4-11

Figure Page

4-6. Quantity Input Screen 4-11

4-7. Set-A-Side Screen 4-12

4-8. No Qualified Vendor Screen 4-13

4-9. Extended Price Screen 4-15

4-10. Vendor Screen 4-16

4-11. Unit Price Screen 4-16

4-12. Problem Vendor Screen 4-19

4-13. Award Screen 4-20

4-14. 'Final' Extended Price Screen 4-21

4-15. CDCF Screen 4-22

4-16. Required Delivery Date Input Screen 4-23

4-17. 'Final' Vendor Screen 4-24

4-18. System Diagram 4-27

5-1. Completed DESC Form 701 5-2

5-2. Completed Panel Selection Form 5-3

5-3. Sign Test For Panel Preference 5-6

5-4. Buyer Selection Form 5-9

5-5. Purchase Request (Identifying NSN and Quantity) 5-10

5-6. Form 36 Trailer (Identifying RDD) 5-11

5-7. Phase II Manual vs Automated Error Rate 5-17

5-8. Phase II Error Rate ANOVA Test 5-19

5-9. Phase II Error Rate 5-20

5-10. Phase II Timing Descriptive Statistics 5-22

5-11. Phase II Processing Times 5-22

5-12. Phase II Processing Time ANOVA Test 5-23

Figure Page

5-13. Phase II & III Error ANOVA Test 5-27

5-14. Phase II Automated vs Phase III Error Rates 5-28

5-15. Phase II vs Phase III Processing Time Descriptive Statistics 5-29

5-16. Average Phase II/III Processing Time 5-30

5-17. Phase II/III Time Comparison ANOVA Test 5-31

5-18. Purchase Request Degree of Difficulty Form 5-34

5-19. Purchase Request Degree of Difficulty ANOVA Test 5-37

5-20. Errors By Buyer 5-39

5-21. ANOVA Test Buyer Error Rate Comparison 5-40

5-22. Tukey Comparison Of Buyer Error Rates 5-41

5-23. Phase II SubGroup One ANOVA Comparison 5-42

5-24. Phase II SubGroup Two ANOVA Comparison 5-43

5-25. Phase II SubGroup Three ANOVA Comparison 5-44

5-26. Tukey Comparison Of Processing Time Means 5-45

5-27. Phase II Subset One Processing Times ANOVA 5-46

5-28. Phase II Subset Two Processing Times ANOVA 5-47

5-29. Phase II Subset Three Processing Times ANOVA 5-48

5-30. Phase II Subset Four Processing Times ANOVA 5-49

5-31. Phase II Subset Five Processing Times ANOVA 5-50

5-32. Phase II Subset Six Processing Times ANOVA 5-51

5-33. Tukey's Comparison Of Low Error Buyers 5-53

5-34. Phase II Automated (Buyer 4)/Phase III Comparison Of Errors ANOVA 5-53

5-35. Phase II Manual (Buyer 4)/Phase III Processing Errors ANOVA 5-54

5-36. Phase II Automated (Buyer 5&8)/Phase III Processing Times ANOVA . . . 5-55

5-37. Phase II Automated (Buyers 5&8)/Phase III Processing Time Descriptive Statistics 5-56

5-38. Phase II Manual (Buyers 5&8)/Phase III Comparison Of Processing Times ANOVA 5-57

List of Tables

Tables Page

2-1. Comparison of DSS and ES 2-6

5-1. Phase I Results 5-4

5-2. Panel Preference Data 5-5

5-3. Phase II Manual Error Results 5-14

5-4. Phase II Automated Error Results 5-15

5-5. Phase II Manual Summary of Responses 5-16

5-6. Phase II Automated Summary of Responses 5-16

5-7. Phase II Error Data 5-18

5-8. Phase II Processing Time 5-21

5-9. Phase III Results 5-25

5-10. Phase III Summary of Responses 5-26

5-11. Phase II & III Error Data 5-26

5-12. Phase II/III Processing Times 5-28

5-13. Purchase Request Degree of Difficulty Responses 5-35

5-14. Purchase Request Average Degree of Difficulty 5-36

AFIT/GLM/LSC/91S-26

Abstract

The objective of this research was to design and validate a methodology that would enhance the productivity of the buyers who perform the vendor selection function at the Defense Electronic Supply Center.

The approach utilized computer automation and software developed specifically for this application.

A prototype was developed and tested.

Comprised of three phases and eighteen buyers, the testing evaluated the prototype in four areas: 1) accuracy of the presented information, 2) thoroughness of the presented information, 3) ability of the user to use the presented information, and, 4) usability by inexperienced personnel.

Examination of the data generated by the test phase confirms the approach used to enhance the productivity of the buyers was valid.

As a result of findings from this research, the recommendations derived include the integration of this methodology into the development of future buyer assistance programs.

SMALL CONTRACT AWARD: IMPROVING THE VENDOR SELECTION PROCESS

I. Introduction

Overview

The goal of this thesis is to show that improvement is possible in the current small contract award process of the Defense Electronics Supply Center (DESC). To reach this goal, the vendor selection process must first be understood. This includes not only the tedious mechanical method of vendor selection; but also the intuition and insight brought to the problem by human intervention. The procedure used to complete this investigation began with an analysis of the current selection process. After the analysis, a literary review was conducted, searching for the proper technology to apply. Finally, a prototype system was developed to test the theories that evolved through research. The following pages document the process performed in this quest to improve the small contract vendor selection process at DESC.

Background

About DESC. The Defense Electronics Supply Center (DESC), is a major supply center for the Defense Logistics Agency (11:6). It is the principal Department of Defense activity for the procurement and management of electronic spare parts (11:7). In 1989, DESC managed almost one million electronic items (Figure 1-1) (12). Their involvement in this area has continued to grow over the last two decades.

ITEMS MANAGED BY DESC BY YEAR

1989	972,479
1988	964,800
1987	962,174
1986	923,205
1985	924,011
1984	896,806
1983	867,393
1982	838,351
1981	770,600
1980	755,700
1979	764,100
1978	734,200
1977	729,300

FIGURE 1-1 -- TRENDS IN ITEM MANAGEMENT

Last year, in performing its mission, one hundred fifty one buyers at DESC entered into contract with some four thousand vendors, resulting in the award of one hundred twenty-five thousand separate contracts, worth six hundred four million dollars (11). Eighty-seven percent of these contracts were given to small and/or disadvantaged businesses (21).

As numerous as DESC's past efforts were, their workload is about to increase.

> On November 11, 1989, the Secretary of Defense directed the OSD staff to review selected Defense Management Report Decisions (DMRD), and where applicable, develop detailed implementation plans. One of the DMRDs encompassed in this review was DMRD 926, "Consolidation of Inventory Control Points (ICPs)." (28:iii).

On July 3, 1990, the Deputy Secretary of Defense announced the approval of several recommendations submitted in the study team's report. Among the teams' approved recommendations was to "transfer item management responsibility for approximately one million consumable items from the Military Services to the Defense Logistics Agency" (5).

As a result, DESC will gain authority for an additional three hundred forty-eight thousand separate contract items (20).

Small Contract Procurement Process. DESC has several different methods for selecting the proper supplier of a product. The method used depends upon the specific requirements of the customer and the item itself. The dollar value of the contract is a major influence on the method selected. Low value contracts comprise a significant portion of DESC's activities. To better control the ever increasing volume of small contract awards, management sees a need to improve the vendor selection process.

For each item inventoried, there is a person responsible for assuring an adequate supply exists to meet the users' needs. This person is referred to as the 'Item Manager' (IM). The item manager informs the 'buyer' at DESC how many units of the item must be ordered to satisfy the demand. The document identifying this requirement is the 'Purchase Request', also known as the 'PR'. Figure 1-2, a through c, illustrates an example of this document.

Each buyer at DESC is responsible for a specific federal stock class of item. All items in a federal stock class have similar characteristics. For example, stock class 5905 contains resistors, while stock class 5910 contains capacitors. The buyer receives the PR identifying the part or product required and is responsible for selecting the appropriate vendor for contract award. To reach this decision, the buyer must determine which vendor provides the item at the lowest cost. However, this is not the only decision factor. Delivery time, past performance and other government guidelines are also considered (20).

To accomplish this, the buyer researches price and vendor information to compile a comparative analysis. This research involves examining hard copy price lists (in non-standardized formats) (Figure 1-3, a through c) and obtaining vendor performance information from several sources. Finally, the buyer must consider such issues as vendor size and ownership before making the final selection.

CONTINUATION SHEET	REFERENCE NO. OF DOCUMENT BEING CONTINUED	PAGE 1 OF PAGES

NAME OF OFFEROR OR CONTRACTOR

ITEM NO	SUPPLIES/SERVICES	QUANTITY	UNIT	UNIT PRICE	AMOUNT
	PR YPE911510003 52				
	NSN 5905-01-265-5245				
	ITEM DESCRIPTION:				
	RESISTOR,FIXED,FILM I/A/W SPEC NR MILR55182F QP DASH 0105 BASIC DTD 90 JAN 26 AMEND NR DTD TYPE NUMBER RNC55H8161DS				
	PRLI 000100	3	EA		
	QTY VARIANCE: PLUS % MINUS % INSP/ACCEPT POINT:				
	PREP FOR DELIVERY:				
	PACKAGING DATA - MIL-STD-2073 (1A 16 JUL 84) (2B 14 MAR 86) QUP = 001: PRES MTHD = C2: CLNG/DRY = X: PRESV MAT = XX: WRAP MAT = XX: CUSH/DUNN MAT = XX: CUSH/DUNN THKNESS = X: UNIT CONT = XX: LEVEL PRESV = A: INTRMDTE CONT = XX: INTRMDTE CONT QTY = XXX: PACK = U: PACKING LEVEL = C: MARKING SHALL BE IN ACCORDANCE WITH MIL-STD-129L SPECIAL MAR KING CODE: 99 - NO CODES IN THIS TABLE ONLY MIL-STD-129				
	DOD LOGMARS BAR CODE MARKING REQUIRED IAW MIL-STD-129L, APPENDIX H, DATED 15 OCT 90 AND MIL-STD-1189B, DATED 10 AUG 89.				
	DELIVER FOB: BY:				
	PARCEL POST / FREIGHT ADDRESS:				
	SW0400 DEFENSE DEPOT RICHMOND DEFENSE GENERAL SUPPLY CENTER RICHMOND VA 23297-5000				
	CONTINUED ON NEXT PAGE				

NSN 7540-01-152-8064 36-109 STANDARD FORM 36

PREVIOUS EDITION USABLE

FIGURE 1-2A -- SAMPLE PURCHASE REQUEST

CONTINUATION SHEET	REFERENCE NO. OF DOCUMENT BEING CONTINUED	PAGE 2 OF PAGES

NAME OF OFFEROR OR CONTRACTOR

ITEM NO	SUPPLIES/SERVICES	QUANTITY	UNIT	UNIT PRICE	AMOUNT
	PR YPE91151000352 PRLI 000100 CONT'D				
	NON-MILSTRIP RDD 2012 PROJ				
	END OF PR				

NSN 7540-01-152-8067 36-109 STANDARD FORM 36

PREVIOUS EDITION USABLE

FIGURE 1-2B -- SAMPLE PURCHASE REQUEST (CONTINUED)

```
F 27 P    DESC                                    PURCHASE REQUEST TRAILER LISTING                    AS OF 31 MAY 91   PAGE   0001

PURCHASE NUMBER  STOCK/PART NR    PGC  S/P PDC  STK UI   UNIT PRICE  UNIT WT  UNIT CUBE  LNGTH  WIDTH  DEPTH SQD/RB      AAC   PDLT
YPE91151000352  5905-01-265-5245            1     EA          5.00      0.00      0.000                          91143       3    150

TPA  CI  AMC  AMSC  ORC  SSR  WS   SPC   PAC   TMC   VEP   B/O   OSD   DMS  RC  CONV FACTOR  TECH DATA REV CODES P E C Q T PMIC GFM
          1     T   BT        Y    1                              91143                                                          N


        PLACE OF INSPECTION    BRANCH-BUYER CODE  DB000              CAUSE              QUALITY CONTROL CODE/CLAUSE
              ORIGIN                                                                                  204

WS ITEMS LINE NR 00 23N

                                        SUGGESTED SOURCES AND REFERENCES                  MORE REFERENCES AVAILABLE IN DD 635 NO

81349 2                        RNC55H8161DS                                               NO ADDRESS AVAILABLE


BASIC NUMBER   MILR55182F        QP DASH 0105  QPL /RDC  Q

 PRLI   LOC  DESTINATION  RDD   QUANTITY  FCC  SEA  PROJ  IPD  N/E

000100  SRE     SW0400   92012          3 EA

                  PR TOTAL QUANTITY              3          TOTAL PRICE                15 00
```

FIGURE 1-2c -- SAMPLE PURCHASE REQUEST (CONTINUED)

QUANTITY BREAKS

R E S I S T O R S

PART NUMBER	QTY 500	QTY 1000
RNC50H(10R0-1503)FS	.236	.227
RNC50J(10R0-1503)FS	.318	.276
RNC50K(10R0-1503)FS	.236	.227
RNC55H(10R0-1004)FS	.192	.181
RNC55K(10R0-1004)FS	.192	.181
RNC55H(10R0-1004)BS	.24	.23
RNC55K(10R0-1004)BS	.24	.23
RNC55J(10R0-1004)FS	.232	.216
RNC55J(10R0-1004)BS	.288	.267
RNC60H(10R0-1004)FS	.193	.187
RNC60K(10R0-1004)FS	.193	.187
RNC60J(10R0-1004)FS	.238	.221
RNC60H(10R0-1004)BS	.26	.25
RNC60J(10R0-1004)BS	.301	.281
RNC60K(10R0-1004)BS	.26	.25
RNC65H(10R0-1004)FS	.272	.255
RNC65J(10R0-1004)FS	.326	.314
RNC65K(10R0-1004)FS	.272	.255
RNC65H(10R0-1004)BS	.306	.301
RNC65J(10R0-1004)BS	.394	.36
RNC65K(10R0-1004)BS	.306	.301

UPDATE EFFECTIVE NOVEMBER 1, 1990...VALID TIL FURTHER NOTICE.....

FIGURE 1-3A -- SAMPLE VENDOR PRICE LISTS

FSCM: 6S313
PHONE: 800-358-8708
DATE: 4-01-90

PRODUCT: RESISTORS
MIL-SPEC: MIL-R-55182
MILITARY TYPE: RNC90Y

CONFIDENTIAL RESALE PRICE LIST FOR USE BY DESC PROCUREMENT
ESTABLISHED RELIABILITY

RANGE		25-49	50-99	100-249	250-499	500-999	1000 & UP
RNC90Y							
50.1 TO 49.9K	F(1%)	5.60	5.22	4.31	4.09	4.01	3.92
50.1 TO 49.9K	D(.5%)	6.30	5.88	4.35	4.60	4.51	4.41
50.1 to 49.9K	B(.1%)	7.01	6.54	5.39	5.12	5.01	4.90
50.1 TO 49.9K	A(.05%)	8.06	7.52	6.20	5.89	5.76	5.64
50.1 TO 49.9K	T(.01%)	9.11	8.50	7.01	6.66	6.51	6.37
50.1 TO 49.9K	V(.005%)	11.91	12.12	9.15	8.69	8.52	8.33
50K TO 59.9K	F(1%)	6.87	6.16	4.74	4.51	4.41	4.32
50K TO 59.9K	D(.5%)	7.74	6.94	5.33	5.06	4.95	4.85
50K TO 59.9K	B(.1%)	9.37	8.40	6.47	6.14	6.01	5.89
50K TO 59.9K	A(.05%)	10.77	9.67	7.44	7.06	6.92	6.76
50K TO 59.9K	T(.01%)	11.16	10.02	7.71	7.32	7.17	7.01
50K TO 59.9K	V(.005%)	14.61	13.10	10.07	9.57	9.37	9.17
60K TO 99.9K	F(1%)	8.13	7.28	5.60	5.32	5.21	5.10
60K TO 99.9K	D(.5%)	9.14	8.19	6.30	5.98	5.86	5.74
60K TO99.9K	B(.1%)	10.15	9.10	7.01	6.66	6.51	6.37
60K TO 99.9K	A(.05%)	11.68	10.46	8.06	7.65	7.49	7.33
60K TO 99.9K	T(.01%)	13.20	11.84	9.09	8.65	8.46	8.29
60K TO 99.9K	V(.005%)	17.27	15.47	11.91	11.31	11.07	10.83

** MINIMUM ORDER 25 PCS **

M, P, OR R LEVEL TOLERANCE
S LEVEL TOLERANCE ADD 40%

FIGURE 1-3B -- SAMPLE VENDOR PRICE LISTS (CONTINUED)

FIGURE 1-3c -- SAMPLE VENDOR PRICE LISTS (CONTINUED)

TYPE MANUFACTURER	TEMP. COEFF.	TOL	FAIL RATE	OHMIC RANGE	PKG.	PRICE APPLIES IN 50 PIECE INCREMENTS					
						50	100	150	200	300	500
RNR65 ANGSTROHM	E	B	S	24.9-48.7	1 per	4.35	3.79	3.70	3.68	3.66	3.64
				49.9-1M	1 per	3.84	3.33	3.25	3.23	3.20	3.19
				1.01M-4.99M	1 per	6.55	5.79	5.69	5.66	5.61	5.59
RNR65 ANGSTROHM	E	D	S	24.9-48.7	1 per	3.54	3.06	2.98	2.96	2.93	2.92
				49.9-1M	1 per	3.45	2.97	2.89	2.87	2.85	2.84
				1.01M-4.99M	1 per	6.03	5.32	5.24	5.22	5.17	5.13
RNR70 ANGSTROHM	C	F	S	24.9-48.7	1 per	5.73	5.05	4.97	4.93	4.88	4.86
				49.9-1M	1 per	2.96	2.52	2.44	2.42	2.40	2.39
				1.02M-1.5M	1 per	3.67	3.17	3.12	3.09	3.06	3.04
				1.52M-7.5M	1 per	5.73	5.05	4.97	4.93	4.88	4.86
RNR70 ANGSTROHM	C	B	S	24.9-48.7	1 per	7.59	6.74	6.62	6.59	6.55	6.51
				49.9-1M	1 per	4.04	3.51	3.43	3.42	3.37	3.36
				1.02M-1.5M	1 per	4.46	3.89	3.81	3.78	3.75	3.73
				1.52M-7.5M	1 per	7.59	6.74	6.62	6.59	6.55	6.51

June 26, 1990
T.T.I., INC. (0BTU6)

********** SEE COVER PAGE FOR INSPECTION **********

Problems. The following problems have been identified with the current process.

Inefficiencies. The current process appears to house inefficient procedures. As an example, each buyer maintains separate price lists provided by each vendor. There is no standardization between vendors regarding the format in which the information is portrayed. There is no consistency in the arrangement of the part numbers, quantity price breaks, or lot size offered (Figure 1-3, a through c). The 'uniqueness' of each price list examined by the buyer leads to needless delay in retrieving the required information (20).

The vendor submits price lists to DESC for each parts class offered. Each time the prices change, the vendor submits an updated list. DESC routes these lists to the proper buyers for their use. Should the buyer complete a vendor award using outdated information, (i.e., before receiving and posting the current prices), a delay in item shipment may result until resolution of the differences is reached.

Guidance. In awarding small contracts, the buyers consult several government guidelines before determining which vendor will receive the contract. These guidelines are not binding. Rather, the guidelines suggest what characteristics the vendor should possess to receive a contract.

With these many inputs into the decision process, management has voiced a concern regarding the accuracy of the decisions being made. Not only is the correctness of the decision an issue, but the latitude inherent to the selection process makes it difficult to justify why a given decision was made (18).

The absence of structure makes maintenance of the needed information a challenging task. Standardizing the presentation of the data could accelerate the selection process. Furthermore, it would ease the task of the buyers as well as reduce processing time if they were not required to calculate rudimentary figures such as the extended price from the unit price for each vendor.

VASPP Concept. Improvement in the small contract award process is only part of a greater vision of Col. Hewett and Mr. Vicars from DESC-P (18). VASPP (Vendor Automated Supplied Pricing Program) (Figure 1-4) is an encompassing program concept that will focus on competitive small purchases under twenty-five thousand dollars (16:2). Under this concept, the manufacturers and distributors (vendors) of an item will submit and update their prices to DESC via electronic means for inclusion into a centralized database. Once received, the buyer would have access to the latest revisions of the vendor pricing information.

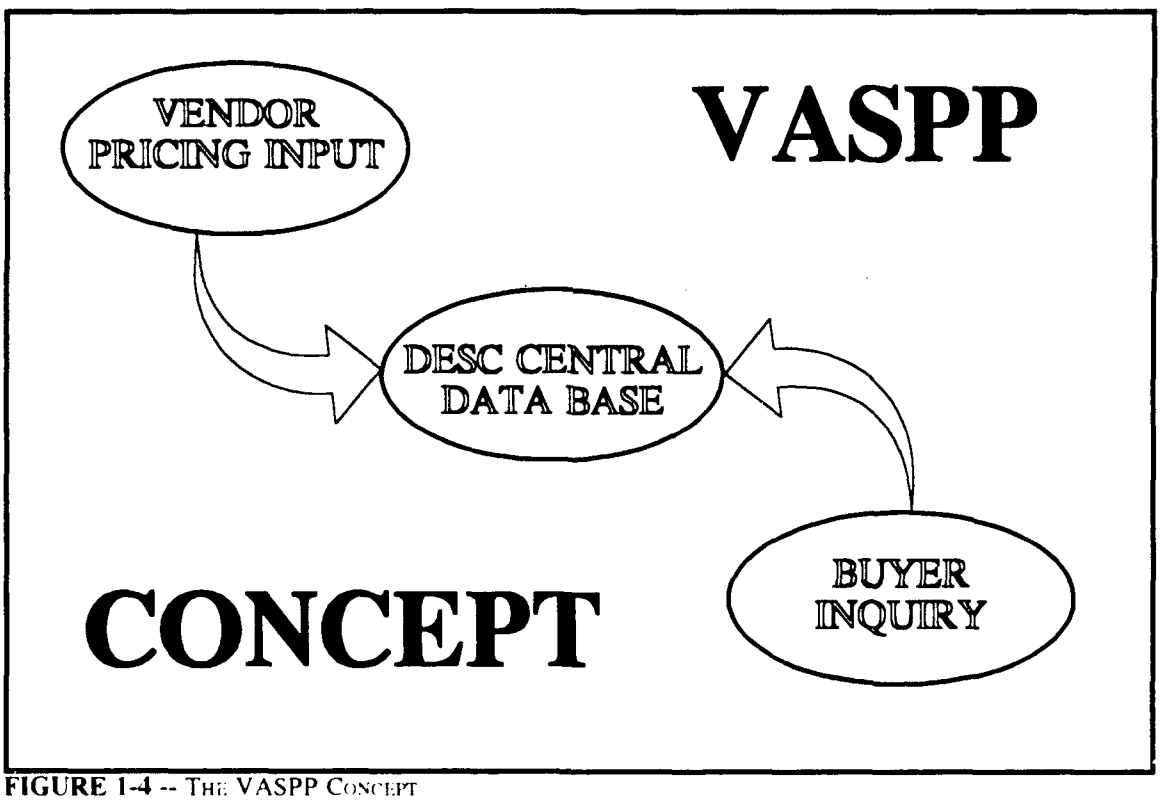


FIGURE 1-4 -- THE VASPP CONCEPT

DESC expects VASPP will aid the organization through enhancements in several areas. Among them:

- Potential to realize significant ALT [Administrative Lead Time] savings with reasonable investment of resources for development and maintenance of the program.

- Resource savings from reduced administrative efforts; for example, avoid clerical function of inputting quotes/alternate bids.

- Focuses on small purchase arena which represents ninety-eight percent of procurement actions and sixty percent of obligated dollars.

- Maintains ready source of supply without some of the disadvantages (e.g., pricing, exclusivity, and resource demands) of long term contracts. (16)

Scope of Research

Specific Problem. The current system used for small contract award determination requires a significant amount of labor to acquire the most basic of data. In addition, to maintain the ability of making the appropriate decisions, the buyer must be ever vigilant for changing information from many sources. As a result, there is degradation in the award process and doubts have arisen concerning the quality of the resulting decision (18).

Research Objective. The overall goal of this project is to determine whether improvements in the current small contracting process are possible. To reach this end, the first objective is to identify the information requirements of the current award process. The next objective is to develop a 'tool' for the buyer. To fulfill the needs of the primary users (the buyers), it should be responsive, and identify those vendors best meeting cost, performance and other governmental guidelines. The third objective is to confirm whether the designed system actually enhances the current process.

Research Questions. To meet the objectives of this research project an answer is needed for the following questions:

1. What information must the buyer obtain before selecting the proper vendor?
2. What information does the buyer generate while awarding a contract to the vendor?
3. What automated management systems are available, and, of these systems, which ones could satisfy the needs of DESC, given the type of data available and the results required?
4. Can an effective automated system be designed, developed and employed to assist the buyer's vendor selection process at DESC?

Areas of Study. The bounds of this study are limited to actions directly related to improving the small contract vendor selection process at DESC. The proposed solution shall take a purchase request input by the buyer and identify the vendor(s) that is(are) competitive on that product. Efforts will focus on the development of a fully functional computer based prototype system. To aid in future integration into the current data processing environment, the prototype will maximize the use of data already available from the computer systems at DESC.

Method of Organization. This paper documents the research conducted using six chapters. Chapter One identifies the problem as described by DESC, and provides background information directing this research. Chapter Two contains the literature review conducted for this project. It focuses on the various methods of computer based management systems and software verification. Chapter Three describes the methodology used to develop a solution to the research problem. Chapter Four describes the development and verification of the system software. Chapter Five includes the analysis of the prototype validation process. Finally, Chapter Six summarizes the research findings and provides recommendations for future actions.

II. Literature Review

Overview

In support of this research, Chapter I identified the following research question: 'What automated management systems are available, and, of these systems, which ones could satisfy the needs of DESC, given the type of data available and the results required?' Required to address this question is the examination of two supporting questions:

1. What type of computer assistant systems can satisfy DESC's requirements?
2. What are the strengths and weaknesses of the systems under consideration?

Once a system is selected and designed, the program coding must be verified. Additional research was conducted in this area to answer the following question: 'Once developed, how can the system be verified?'

The findings from these questions can be used to answer Research Question number three. The information obtained will affect the structure of the proposed system, and consequently how the system will be tested.

Prototyping

The total VASPP concept, (explained in Chapter I) which this research supports, extends well beyond the scope of this project. There is little guidance regarding the final structure VASPP will assume. As a result, the author views these efforts as a prototype from which future developments will spawn. Initial prototyping is an effective method for dealing with ideas that have yet to solidify.

> This design strategy, known as prototyping, has proven to be useful across a wide range of informational systems' applications. In general, prototypes have been shown to:

(1) improve the likelihood of developing systems desired by users,

(2) shorten the overall development period,

(3) reduce management risk, and

(4) serve as specifications for further (later) system development. (7:94)

Computer Assistant Systems

This portion of the literature review addresses the first set of supporting questions. For the purposes of this research, the phrase 'Computer Assistant System' refers to an application of computer technology that aids the user in the decision making process.

Before proceeding with the review, it is helpful to summarize what is known thus far concerning DESC's requirements. First, DESC would like to simplify the small vendor selection process. Areas appearing to have latitude for improvement are: standardizing the vendor price lists, providing the buyer with past procurement information, and reducing the need to perform routine calculations.

Secondly, DESC would like to use the results of this project as a baseline for the VASPP program. If successful, this research will lay the foundation on which to build follow-on development efforts.

Systems Reviewed.

Database Management Systems (DBMSs). Database management systems are a means of keeping current information in a readily accessible format available for convenient review. "... *a data base management system* (DBMS) is generally defined as a collection of computer programs used to create, maintain, access, update, and protect one or more data bases" (30:222).

Some advantages of this type of system include:

1. It offers rapid access to and flexible use of information. A DBMS uses sophisticated methods of organization and retrieval.

2. The incidence of redundancy (repetition) is limited and information kept current. This is critical, because there is a direct relationship between the efficiency of a computer program and its ability to avoid storing unnecessary information and to keep the information it does store up-to-date.

3. The cost/benefit ratio is good. The cost of setting up and operating a DBMS is low compared to the value of the benefits it affords.

4. Storage of information is compact, compared to paper storage.

5. Mundane, repetitive tasks such as searching for information and preparing reports can be automated.

6. A DBMS imposes an organized structure that would be difficult to attain manually. Once a DBMS has been established, its maintenance encourages efficiency in office procedures. (31:8)

These benefits are not without their corresponding drawbacks. Some disadvantages of using Database Management Systems are:

1. Operation and programming requires skill in the use of the system as well as a knowledge of DBMS concepts.

2. Because information is stored in a complex way, it can be difficult to back up or reconstruct.

3. Information is centralized, and it requires maintenance. Someone must assume responsibility for administering the DBMS.

4. As the power and features of the DBMS are utilized more complex information management is required, and this generates new administrative problems. (31:8)

Decision Support Systems (DSSs). There are many variants to the definition of a Decision Support System offered in current literature. M. J. Ginzberg and E. A. Stohr offer one that seems particularly applicable. Their proposal reads: "a DSS is a computer-based

information system used to support decision making activities in situations where it is not possible or not desirable to have an automated system perform the entire decision process'' (17:12).

The components of the Decision Support System are: a) the database, b) the database management facilities, c) the quantitative modeling component, d) the report generator, and e) the human interface (10:75). These elements combine to provide the user with the information required to base a decision.

The key characteristics of effective DSS are:

1. Support for semi-structured (underspecified decisions)
2. Support for all phases of decision making (intelligence, design, choice, implementation)
3. Combination of modeling (analytic) techniques with data base and data presentation techniques
4. Emphasis on ease of use and flexibility/adaptability (compared to execution efficiency)
5. An interaction with transaction processing (EDP) and other information systems, such as MIS and office systems (30:300)

Expert Systems (ESs). ''An Expert System captures and stores . . . knowledge, such as rules, policies and logic, in a knowledge base in much the same way as a conventional computer program stores numeric information in a database'' (3:25). It is comprised of the following components; an inference mechanism, a knowledge base, a database management component, a report generator, and finally a user interface, (10:65). (Figure 2-1)

Following is a list of properties . . . common to many expert systems:

- Explicit representation of domain knowledge.
- A general-purpose inference mechanism providing control.
- Provision for reasoning with uncertain evidence and knowledge.
- Provision of justification, explanation and other run-time user support (8:7)

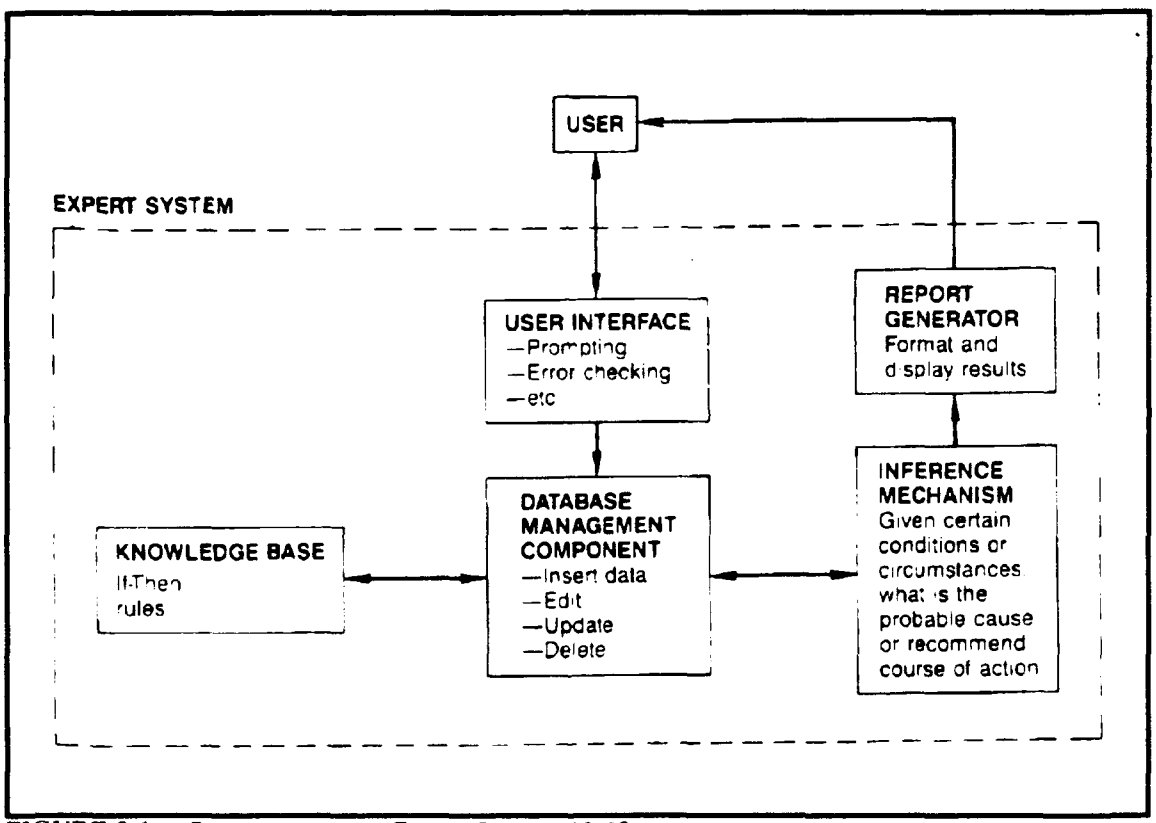


FIGURE 2-1 -- COMPONENTS OF AN EXPERT SYSTEM (10:65)

There are several advantages inherent to the development process of expert systems. Frequently, the act of developing an expert system provides the first documented record of the knowledge contained in an area. The existence of such a system provides consistency that is usually not present with humans. And, knowledge captured in these systems is available to a larger audience than a finite number of human experts (1:20-22).

However, expert systems are not without their limitations. "Currently expert systems can address only very narrow areas of expertise and have limited capability to encode common sense. . . . Expert Systems also have only limited capability to explain their reasoning" (3:45).

Table 2-1 depicts the difference between Decision Support Systems and Expert Systems. To summarize the table, the focus of the decision support system is to aid the manager in identifying the best alternatives. The expert system, however, seeks to find the single best solution to a problem. Because of this difference between the systems, the background of each user typically is different. A manager typically uses a decision support system to identify the range of possible solutions, and then adapts the solutions to the real

TABLE 2-1

Comparison of DSS and ES (29:49)

	Decision Support System	Expert System
Paradigm	Management decision making	Problem Solving
Goal of system	Support of intuition	"Complete" solution
Goal type	"Ill-specified"	"Well-specified"
User	Manager	Educated layperson
Factors of influence	Not predictable From many domains	Predictable Restricted
Representation problem solving	Sparse representation	Dense representation
Control	With the user	With the system
Techniques	Tools in formalized subdomains	Artificial Intelligence Knowledge represe.

world problem. The user of an expert system generally has little background knowledge of the problem and acts on the single answer provided.

Applicability. The literature review provided the following key characteristics regarding each system considered.

1) Data Base Management System
   a) Stores, maintains and retrieves data.
   b) Presents all stored data whether relevant or not.

2) Decision Support System
   a) Excludes information irrelevant to the question.
   b) Supports ill-defined problem analysis.

3) Expert System
   a) Provides a single answer to an inquiry.
   b) Requires highly structured problem definition.

The vendor selection process at DESC involves more than the storage and retrieval of data (the focus of the database management system). The information requirements extend beyond simple reporting of stored information. Vendor pricing information is the data of primary interest. This information however, is not used in isolation. To be useful, the pricing data must be reviewed with vendor performance and market reasonableness data. For these reasons, an approach using a pure database management system ideology is unsatisfactory.

An expert system's purpose is to arrive at a single conclusion given a well defined set of constraints. All inputs to the vendor selection process are not yet succinctly defined. The decision process at DESC involves a synthesis of empirical data and buyer experience. Without a solid understanding of how all the inputs interact, making a successful expert system is unlikely. While this is a worthwhile project for future research, it extends beyond the timeframe available for this developmental effort. Therefore, the expert system approach is rejected.

The decision support system appears to be able to satisfy the research criteria. It possesses the features of incorporating data file structure with a set of 'intelligent' rules, thereby screening the data presented to the user.

> The concept of DSS requires that the data base(s) and these modeling techniques be brought together in an interactive way to enable multiple alternatives to be evaluated and to ensure that the best decision is made.
>
> Helping the . . . manager through the decision-making process does not mean that the DSS will produce THE answer, The more correct focus is to interpret the DSS result as a suggestion. The [manager] is still the decision-maker and needs to think of the outputs of the DSS as result which should be considered with other variables . . ..(13:2)

With the philosophy of a decision support system closely paralleling the direction of this project, an examination of the decision support system's components is in order.

Allen and Emmelhainz identify three fundamental elements of the decision support system as: *the dialog subsystem, the data base subsystem*, and the models subsystem (2:132).

The following compares the characteristics of each subsystem with the problems identified in the vendor selection process.

> The dialog subsystem establishes the degree, format, and method of interface with the user. Many DSS experts consider this the most important subsystem since the power, flexibility, and usability characteristics of the entire DSS are determined by the dialog subsystem. The two components of this subsystem are the communication methods (software) and the equipment (terminals, etc.). Nearly all dialog subsystems include interactive terminals as the interface equipment (2:3).

For the prototype to communicate with the buyers, some form of a dialog subsystem must be in place. The proposed method capitalizes on the versatility of the personal computer as the input/output device.

> The data base subsystem is the storehouse of knowledge for the DSS. It records and manipulates data from both internal and external sources, This subsystem usually has the capability of combining data from a number of sources, adding or deleting data quickly, and presenting it in user-understandable terms. Most data base subsystems allow for interactive input of data. The output of the data from the data base subsystem is often used as input to the models subsystem (2:3).

'Combing data from several sources' is crucial to this project. The prototype will be asked to track data maintained in several different data files and present only the information that is relevant to the buyers inquiry.

> The models subsystem contains the analytical techniques used to evaluate data and to determine "Solutions." This subsystem catalogs and maintains a wide range of models to support all levels and functions of users. In many DSS, the models subsystem is imbedded in the information (dialog) subsystem to allow easy, interactive access to the models by the user (2:3).

In the approach applied by this research, the models subsystem is perhaps the least autonomous of the three systems. The model coding lies dispersed throughout the prototype. Portions of the model function in tandem with the data base manager. Other functions are not called upon until the screen displays are presented to the user. The model used in the prototype performs both analytical (i.e. performing extended price calculations) and discriminatory (i.e. screening debarred vendors from the user) manipulation of the data.

Through the data review and discrimination process, the system should provide the user with only the data relevant to the decision making process, and inform the user of any peculiarities existing in the data set. Providing the user with 'just the facts' should provide a faster, more precise, and ultimately superior solution than is obtainable using current methods.

Synopsis. This review examined the characteristics of three types of automated assistant systems. Those considered were: database management systems, decision support systems,

and expert systems. The method displaying the most promise to satisfy the needs of DESC is the decision support system.

System Verification

Having identified the basic characteristics the system requires, attention is now turned to software testing for the system. This section addresses the third supporting question, 'Once developed, how can the system be verified?'

Testing vs Debugging. It is interesting to note a difference exists between software testing and software debugging. "The purpose of testing is to show that bugs exist. The purpose of debugging is to find the error or misconception that led to the program's failure and to define the program changes that correct the error" (6:5). Beizer lists the following differences between testing and debugging:

1. Testing starts with known conditions, uses predefined procedures, and has predictable outcomes. Only whether or not the program passes the test is unpredictable. Debugging starts from possibly unknown initial conditions, and the end cannot be predicted, except statistically.

2. Testing can and should be designed and scheduled beforehand. The procedures for, and duration of, debugging cannot be so constrained.

3. Testing is a demonstration of error or apparent correctness. Debugging is a deductive process.

4. Testing proves a programmer's failure. Debugging is the programmer's vindication.

5. Testing should strive to be predictable, dull, constrained, rigid, and inhuman. Debugging demands intuitive leaps, conjectures, experimentation, intelligence, and freedom.

6. Testing, to a large extent, can be designed and accomplished in ignorance of the design. Debugging is impossible without detailed design knowledge.

7. Testing can be done by an outsider; debugging must be done by an insider.

8. While it is possible to establish theoretical limits to what testing can and cannot do, debugging, so far, has not been amenable to theoretical treatment. (6:5-6)

As alluded to, debugging is a very inexact art performed by the programmer. Testing, on the other hand, is more of a science, and may be performed by anyone.

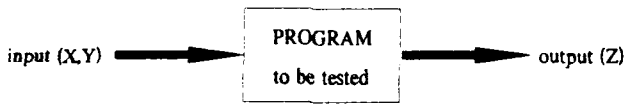
Testing. "There are two steps in functional testing. The first involves the identification of the functions that are implanted in a program. The second involves the selection of test data that can be used to check that the program implements the functions correctly" (22:281).

Program Functions. One method for verification involves functional analysis of the software. But how does one identify a function? Howden describes a function with the following: "The most important feature of a function is that it can be independently tested. The input and output domains for each of the functions . . . can be completely specified" (23:282). This brings to mind the concept of modular programming, the process of designing the software in discrete but cooperative units. "To avoid . . . difficulties every large program should be divided into a series of modules or procedures (subroutines and functions) so designed that each does a clearly defined task, is a logical part of the original problem, and so far as possible uses only its own, locally defined variable" (27:65). Each module in the program has its own unique input/output criteria and can be developed apart from the rest of the system.

Test Data.

Identification. The test data comprises the other major element of the testing process. One might believe if a thorough test is to be conducted, it would be necessary to

submit data elements for all possible inputs. As J. C. Huang points out in his paper, this is an impossible quest.

> Suppose the program to be tested has two input variables and one output variable as depicted below:
>
> 
>
>
> If, for an assignment of values to the input variables X and Y, the output variable Z will assume a correct value upon execution of the program, then we can assert that the program is correct for this particular test case. And if we can test the program for all possible assignments to X and Y, then we will be able to determine its correctness. The difficulty here is that, even for a program with only two input variables, the number of possible assignments will be prohibitively large. To see why this is so, let us assume that X and Y are integer variables. Furthermore, let us assume that the program is to be run on a computer with 32-bit registers. There are $2^{32} \times 2^{32} = 2^{64}$ possible assignments to the input pair (X, Y). Now suppose this program is relatively small, and on the average it takes one milli-second to execute the program once. Then it will take more than 50 billion years for us to complete the test! (24:289)

There is an alternative to absolute testing. "The two most important kinds of functional test data are *extremal values and special values*. Extremal values lie on the "edges" or "boundaries" of sets of data, Special values have special algebraic or computational properties" (22:184). These two data types may be defined further by the following:

> The identification of extremal values for unstructured numeric variables is relatively simple. If the domain of the variables is an interval of the form $[a, b]$, then a and b are the extremal values. If the variable is of type integer, then $a + 1$ and $b - 1$ can also be considered extremal. Each element of a small finite set of elements can be thought of as an extremal value. If a numeric variable is used in a function that carries out arithmetic computations, then the special values for the variable include zero, $\pm e$ (for e small) and $\pm E$ (for E large). Similar rules can be used to identify important test data values for non-numeric, unstructured variables (23:284).

Application. With the tools in hand, attention is turned to their application. To apply the variables, we look not at the program modules, but analyze the program logic, seeking to describe the program paths. A program path is "the sequence of instructions which is performed for a given set of inputs. If this works correctly, then all other sets of inputs which cause the program to follow the same path also yield the correct result" (27:88).

> Path testing is a structural test technique that focuses on control structures rather than processing. A process has one entry and one exit. It performs one or more operations on data. It can consist of one instruction or a long sequence of instructions unbroken by program branches or junctions. From the point of view of path testing, a one-instruction process and a 1000-instruction process are equivalent - they are both processes (6:38).

The application of these concepts as described below will be useful:

> It is convenient to abstract the notion of path further and to deal with a graph representation of a program. Junctions and decisions are replaced with the more abstract and simpler notion of node. A node is any point in the program where the control flow either merges or diverges or both. Nodes are joined by links. Processes, as defined above, are examples of links. However, a link may do no actual processing. For example, a conditional branch instruction consists of a node (the decision instruction) and two links (the flowchart lines that depict the branch alternatives.) The graph representation is convenient because it depicts only labels or addresses and the path segments that join them (6:38).

These paths may not necessarily correspond to the developmental program modules defined in the above section. They may be a subset of, or an amalgamation of those modules. In most cases, the result is a simpler, easier to comprehend representation of the programming logic (6:38).

Approach. To verify the prototype, path identification and testing is a viable method. The verification process can be simplified through the use of modular software design techniques. As such, a modular development approach is adopted. Under this concept,

prototype testing is accomplished by first identifying the program paths. Once the paths are defined, they are examined to identify their specific extremal and special values, as well as the associated results. After software analysis identifying pertinent inputs and expected outputs is completed, system performance can be tested using this anticipatory information as judgmental criteria.

Conclusions

This research conducted in support of the chapter focused on three questions. The first being, 'What type of computer assistant systems can satisfy DESC's requirement?' Three systems were examined, each with its own strengths. Those systems examined were: data base management system, decision support systems, and expert systems. It is believed that a decision support system can best fulfill the DESC's requirements. The attributes of each system was reviewed as required by the second question, 'What are the strengths and weaknesses of the systems under consideration?'. A decision support system was selected based on the constraints imposed by the problem. It was nether required nor desired by DESC to have the system provide 'a' solution. Buyer analysis of the decision criteria will still be accomplished. As such, simplification of the data reviewed by the buyers was sought. The final question, 'Once developed, how can the system be verified?', was addressed next. Software verification will be accomplished through extremal and special variable application through program paths. The details of these procedures can be found in the following chapters.

III. Methodology

Overview

This chapter describes the approach used to identify the bounds of the research problem, and describes the development and evaluation processes that will follow. Problem identification was achieved by conducting personal interviews at DESC with the management and those workers directly affected. After prototype development was completed, an experiment was conducted to test the effectiveness of the resulting design.

Problem Identification

Methodology. The efforts of this development will be integrated into an encompassing program (VASPP). Therefore, it was first necessary to become familiar with the larger system and how the development efforts of this research will integrate into it. This was completed through a series of interviews with the DESC management. "This is the stage when knowing who, what, where, when, how and how much is important. The most effective means of obtaining this information is by interviewing. One of the advantages of interviewing is: 'in the depth and detail of information that can be secured'" (15:60).

An introductory meeting was held with the Chief, PPS (Procurement and Policy), to gain a better understanding of the VASPP concept and DESC's expected benefits from this development effort (9). Through these interviews, information was gathered concerning the scope of the VASPP project. As a result of information extracted from this meeting, Figure 3-1 was constructed as a simplistic, visual representation of VASPP. This was presented to COL Hewett (DESC-P) and his staff (18). The concepts portrayed by this model were accepted by DESC with minor changes.

The VASPP system, as envisioned, will receive inputs concerning bid and pricing information from the vendor. The inputs will enter the system through an electronic or

telecommunication medium. These inputs will be checked for validity and integrity by the translator module. After passing validity checks, the information is formatted for inclusion into the central vendor pricing database. The buyers at DESC may then interrogate the database through the decision support system to cull out the vendors appropriate for a given request.

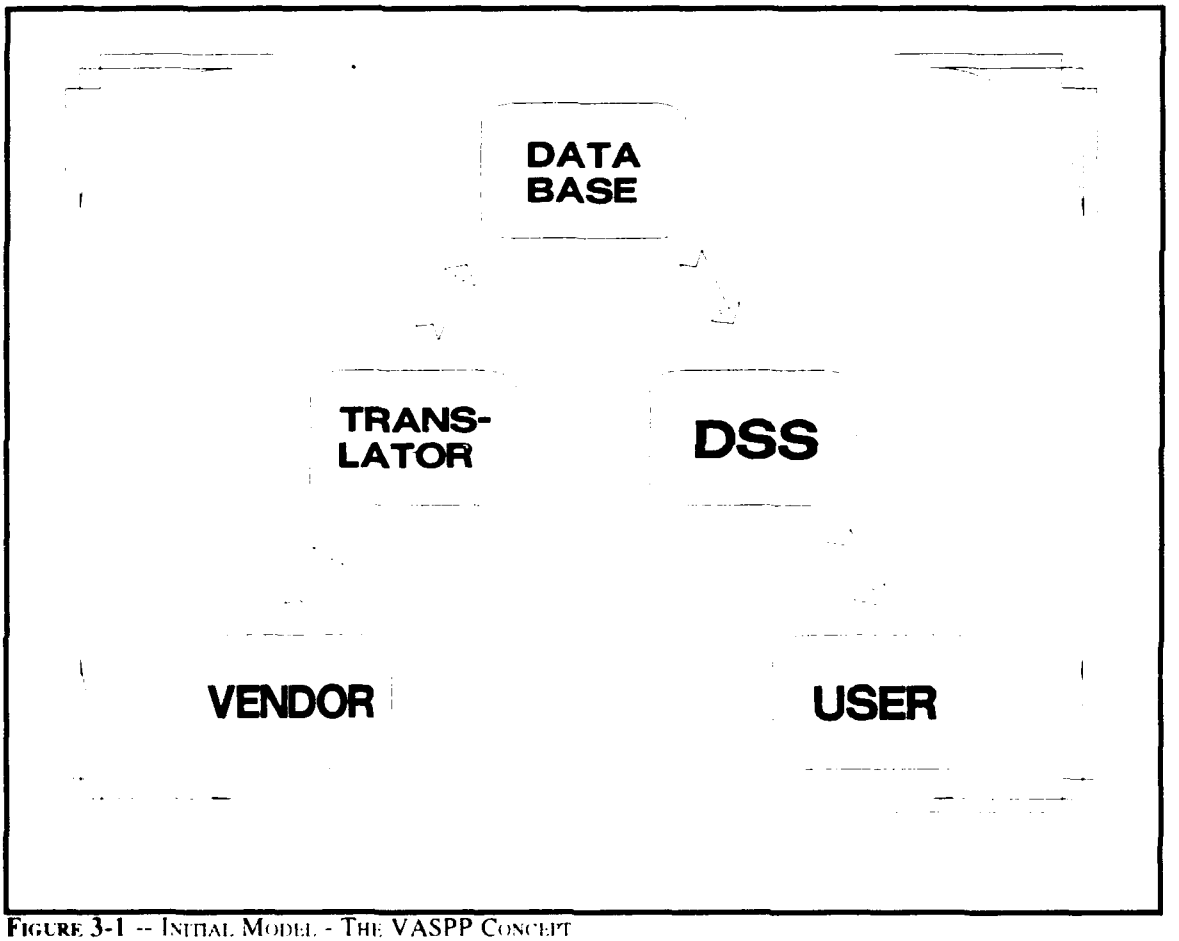


FIGURE 3-1 -- INITIAL MODEL - THE VASPP CONCEPT

Decision Criteria. The VASPP concept is a composite of several operations that must interact with one another. The development of the entire VASPP system extends beyond the scope of a single thesis. Therefore, the author chose to explore a single aspect of the program. Focus was placed on the end user requirements of the system. By having the

destination clearly identified, it will be easier to orient the development efforts of future system modules in the proper direction.

Examination of the vendor selection and award processes was the first step in identifying the requirements of this system. This was accomplished through personal interviews with the buyers and management at DESC (20). The details of those interviews can be found in Chapter IV.

Once the concept of the award process was understood, the next area explored was the identification of the data used in the buyer's decision process. This information sprang from several different sources. The identification of those sources was accomplished through interviews with Mr. M. Corelis and Mr. D. Dickman (9).

Several data elements were identified relative to the decision process. They include the following components:

a) most economical quantity pricing,

b) existence of DESC-identified quality vendors,

c) existence and degree of DESC-identified problem vendors,

d) existence and degree of customer complaints toward the vendors, and;

e) existence of excessive overdue orders from the vendor.

Also, consideration must be given to other information where guidance is less formalized. These data, alone, cannot be used as the sole criteria from which a decision is made. However, they can influence the final decision when viewed with other factors previously mentioned. These elements are:

a) size of vendor business;

b) ownership of vendor business; and,

c) freedom of the buyer to contract beyond the requested quantity.

Having completed the process of identifying the decision criteria, the next step was to learn how to apply that criteria. This was achieved through working directly with the buyers on the floor. First, the vendor selection process was observed by the researcher. To verify the process was understood, the researcher processed several purchase requests under the scrutiny of the buyer. The buyer observed the researcher's actions to assure consistency and completeness with the established procedures.

Proposal Development

Methodology. Figure 3-2 outlines the process used in identifying the characteristics of the problem and its transformation into the Decision Support System.

DEVELOPMENT PROCESS

- Examine Buying Process
- Determine Inputs and Outputs
- Validate Information Requirements
- Develop Logic Flow
- Produce Program Code
- Demonstrate the Prototype

FIGURE 3-2 -- STRUCTURE OF DEVELOPMENT PROCESS

As stated earlier, the first task in the development process was to gain an understanding of the current buying procedures. Once understanding of the procurement process was gained, the input and output requirements of the buyer were analyzed. Accomplishment of the above was achieved through, and confirmed by, personal interviews with the management and buyers at DESC.

Now, being knowledgeable in the fundamental process used in small contract procurement process, a detailed logic flow diagram was developed to capture the concepts needed for software development. (This flowchart is detailed in Chapter IV.) In designing the prototype, the researcher's goal was to incorporate a structure that could be expanded to manage the procurement of thousands of items. High consideration was given to system design to lessen the impact of data maintenance overhead. As a result, the identification, transformation and utilization of data already collected and maintained at DESC, was given the utmost consideration.

After essential core elements of the prototype system were coded, it was examined by DESC for consistency with their conceptual requirements (19).

The DSS. Figure 3-3 depicts the informational flow to/from the user and supporting data bases, through the developed DSS. It is comprised of three sections, the dialog, the database, and the model subsystems.

A request for information is entered by the user into the input/output subsystem (dialog subsystem). The system compares the request against the data stored in the price data file. Bidding vendors are examined for past performance information by the database subsystem. The model subsystem reviews the results obtained thus far. It removes any extraneous data and alerts the buyer to unusual circumstances. The filtered information is passed onto the input/output subsystem, where it is displayed on the terminal for user review.

Having the potential to be used by many users, it could not be assumed all users would have a high degree of computer experience. Thus, an effort was made to keep the dialog system as modest and direct as viable.

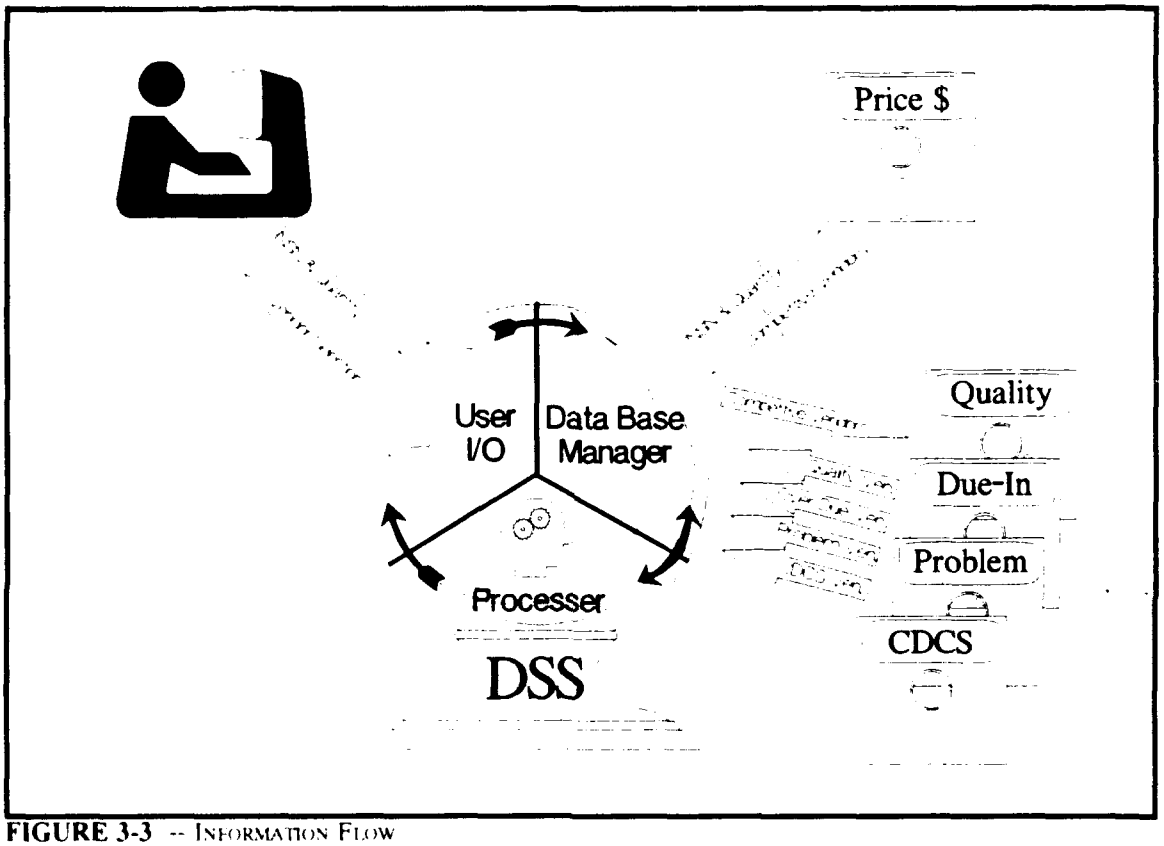


FIGURE 3-3 -- INFORMATION FLOW

An effort was made to reduce the space required for additional data storage and maintenance tasks for the data processing center at DESC. The database subsystem was designed to maximize the use of file information currently in existence on line. For efficiency of data interrogation, the database encompasses several small data files containing related fields. This approach, as opposed to the use of a few large, encompassing files, enhances the system analysis of the data, enabling faster data searches and retrieval.

To provide the management at DESC the ability to tailor the prompts provided to the buyer, a model subsystem was incorporated. Selected outputs of the system can be

changed based on the contents of this user model, thereby influencing the final award. Criteria for selecting which outputs to modify were based on the DPAC's information system now in use at DESC. DPAC is a computer system that is used for other type of vendor awards.

The model subsystem is a separate file that contains configuration parameters controlled by the system manager. These parameters influence the range of 'acceptable' bids and the presentation of informational prompts to the user.

Verification

Knowledge gained through the literature review was applied in the verification process. Logic diagrams were constructed identifying the activities the prototype was required to perform. Independent tasks were isolated to assist in modular development. The operation of the DSS was verified after the addition of each software function. After completing the development process, the prototype was tested to assure inter-module compatibility. Using the technique of flow path identification, the model and data files were modified as required and the system tested to insure all paths were functioning. Any unexpected results were analyzed and if appropriate, corrected.

Validation

The goal of this research project is to design and develop an effective automated system to enhance the contract award process at DESC. This system must be easy to use and provide the correct information, enabling the buyer to select an appropriate vendor for each purchase request. To determine if the developed system meets these criteria, a three step testing process was carried out.

Phase I.

Overview. A panel compared the information provided by the automated system against that provided by the current process.

Procedure. A panel of 'experts' was formed to rate the system. This panel consisted of the following individuals:

1) An experienced buyer. This member will be knowledgeable with the 5905 award criteria and be selected by the '5905' supervisor.

2) An experienced contracting officer. This person should be responsible for insuring the daily accuracy of the 5905 contract awards. He/she will be nominated from management overseeing the buyer floor.

The panel selected thirty purchase requests for MilSpec 55182 from the 5905 stock class input stream. (MilSpec 55182 was the data subset available to the researcher for testing purposes.) The prototype testing was accomplished through the following process.

A purchase request was arbitrarily selected from the sample set. It was processed using the existing manual method for vendor selection. Special attention was given to the specified data files interrogated and the information provided by those files. These data were recorded on a form attached to the purchase request (Figure 3-4).

A standard abstract, DESC Form 701, was also prepared to record all vendor pricing information for the item identified on the purchase request (Figure 3-5).

After these steps, the panel members determined which vendor should receive the contract award. If it was unclear which vendor should receive the award, those under consideration were recorded.

Having completed the manual process, the same purchase request was entered into the prototype system by the panel. The panel recorded any deviations or omissions of the resulting information provided by the system. This information was placed on the form identified in Figure 3-4 as well. Using the information provided by the automated system, the panel again determined which vendor was most qualified to receive the contract award. The selected vendor (or vendors) were recorded on the same form.

The vendor selected, the quantity ordered, and the total contract value obtained from the manual system was compared to that from the automated system. The panel documented

PANEL SELECTION

PR: _____

Problem Vendor

Problem Product

Purchase History

Other	

Award Information

Preferred ☐ Tie ☐ Preferred ☐

Manual System			
	Vendor Selected:	Quantity:	Extended Price:
1.			
2.			
3.			

Automated System			
	Vendor Selected:	Quantity:	Extended Price:
1.			
2.			
3.			

Comments

FIGURE 3-4 -- PANEL SELECTION FORM

ABSTRACT OF PROPOSALS QUOTATIONS
(NEGOTIATED PROCUREMENT) MIL-R-55182*

METHOD OF SOLICITATION: ☐ ORAL ☒ WRITTEN

☐ Prices from Unsolicited Quotes
☒ Prices from Price Book
☒ Prices from Solicited Quotes

NAME AND ADDRESS OF PROPOSER

1	2	3
1G379 PIONEER M05 dale/bradford	54991 ASCHBACHER M05 mepco/	7R545 HAMILTON AVNET M05 dale/trw/itc x-3 add .04¢ ea.
4	5	6
56856 VAMISTOR min. of $45 per line direct ship subtract 10¢ ea.	6S313 G & A SALES M02 ultronix/mepco/corning	

PROPOSAL NO.	1	2	3	4	5	6
PROPOSED DELIVERY	280	180	220	120	220	
PROPOSED CASH DISCOUNT	N 30	N 30	N 30	2% 10	2% 10	
SB-SMALL BUSINESS LB-LARGE BUSINESS	LB	SB	LB	SB	SB	
F.O.B. DATA	D	D	D	D	D	

ITEM NO.	UNIT	QUANTITY						

NAME AND ADDRESS OF PROPOSER

7	8	9

10

MEMO: Prices are considered fair and reasonable due to adequate dealer and/or manufacturer competition.

PROPOSAL NO.	7	8	9	10
PROPOSED DELIVERY				
PROPOSED CASH DISCOUNT				
SB-SMALL BUSINESS LB-LARGE BUSINESS				
F.O.B. DATA				

ITEM NO.	UNIT	QUANTITY				

PRICE CERTIFICATION

☐ No buy last 12 mths.
☐ No increase.
☐ Increase less than 10% (see tape).
☐ Increase over 10% (see F-774D).
☐ First time buy.

I have solicited proposals and recommend that award be made as indicated on this abstract.

DATE | SIGNATURE OF BUYER

DESC Form 701 MAR 82 Previous Edition May Be Used

FIGURE 3-5 -- DESC FORM 701

any deviations between the systems. They then determined which system provided the more appropriate answer, or, documented the existence of equally correct vendor selections.

This process was repeated for the remaining twenty-nine purchase requests. Upon conclusion, the researcher totaled the number of times each of the systems provided the superior answer and the number of times the two systems resulted in equivalent answers. A sign Test was used to analyze the results. A Sign Test was chosen because. . .

> "The **Sign Test** is a nonparametric alternative to the **Paired T Test**. It requires virtually no assumptions about the paired samples other than that they are random and independent. On the negative side, it is not as powerful as the **Paired T Test** or the **Wilcoxon Signed Rank Test**. It is especially useful for situations where quantitative measures are difficult to obtain, but where a member of the pair can be judged 'greater than' or 'less than' the other member of the pair (4:207).

The panel members from DESC were asked to provide a narrative of their comments regarding the automated system performance and effectiveness compared to the manual system. This was accomplished on the form depicted in Figure 3-6.

This completes the first phase of the validation process. By analyzing the data that this phase generates, a determination was made regarding whether: 1) the system presented the correct information to the buyer for an award decision; and 2) the system performed in a manner consistent with the expectations of DESC?

Because of the importance of the decisions this system will influence, a high degree of confidence in the system must exist. Accordingly, the minimum acceptable level of accuracy for the initial prototype was set (somewhat arbitrarily) at ninety percent confidence. If this level of certainty cannot be met, the validity of the succeeding phases would be questionable.

The second criteria that must be met before advancing to the next phase of testing is the panel's expectations in the system. If the system fails to meet the panel's expectations,

or if the panel believes the system fails to perform within acceptable standards, they may elect to cancel further testing.

PANEL QUESTIONNAIRE

Now that you've had a chance to work with the Automated Vendor Selection System, please take a few minutes to answer the following questions regarding the system's performance.

Describe any problems you incurred while using the system.
What information presented by the system, if any, is irrelevant to the award selection process?
What other information should the system provide to aid in the award process?
Do you have any suggestions for future enhancements to this system?
Do you have any other comments or suggestions regarding the design or usefulness of this system?
As presented today, does the system assist the buyer in the vendor selection process?

FIGURE 3-6 -- PANEL QUESTIONNAIRE

Phase II.

Overview. A panel of eight buyers processed the thirty purchase requests in the sample set using a combination of the manual and the automated systems. The manual run was compared to the automated run with respect to processing time. The vendors selected using the automated run for each purchase request were compared to those selected by the 'expert' panel, to determine whether the buyers arrived at the correct answer.

An Analysis of Variance (ANOVA) Test was used wherever appropriate to compare samples. A randomized block design was used to analyze the data. ". . .the **randomized block design** utilizes experimental units that are *matched sets*, assigning one from each set to each treatment" (26:878).

Procedure. A pool of eight buyers was formed from the buyer floor. These buyers were to have experience in the 5905 stock class items. The software was loaded on the eight personnel computers belonging to the buyers. Eight copies of each purchase request in the test set was produced, each with a blank results form attached.

When this phase of the testing begins, four of the buyers were given half of the purchase requests (fifteen) to process manually. The other buyers were given the remaining requests to process on the automated system The buyer noted the time processing of that request began on the form attached to each request (Figure 3-7). Once a vendor was selected, the buyer recorded the chosen vendor, quantity ordered, and total price of the award. After completion, the buyer recorded the current time, and indicated if they experienced any external delays (i.e., phone calls) while processing the transaction.

This process was repeated until all fifteen (half of the complete set) purchase requests were completed. After completion, the buyer returned the purchase requests to the researcher and received the remaining fifteen requests for processing. If the buyer used the manual system to process the first set, he/she processed the second set using the automated

BUYER SELECTION

PR: _____ Buyer No: _____ A

Timing Criteria

Time Started: _____ Delayed ☐

Time Finished: _____ _____

Award Information

Vendor Selected:	Quantity:	Extended Price:
_____	_____	_____

(Place Comments On Reverse)

FIGURE 3-7 -- BUYER SELECTION FORM

system. Conversely, if the first set was processed using the automated system, he/she processed the second set manually. After the buyers complete both manual and automated processing phases, they were asked to complete a system evaluation form (Figure 3-8). This also was returned to the panel upon completion.

The times required to process the purchase requests were summed for both the manual and the automated sets. Any purchase request that showed a delay in processing occurred will not be included in the totals. The average processing time of the remaining requests will then be calculated for each of the two methods.

Using a consolidation form (Figure 3-9), the researcher recorded the buyer's selection for each purchase request processed using the automated system. Also, it was noted whether the buyer arrived at the same award decision as the panel. To qualify as a match, the vendor, quantity and price must agree. If these three criteria did not match the

BUYER QUESTIONNAIRE

Now that you've had a chance to work with the Automated Vendor Selection System, please take a few minutes to answer the following questions regarding the system's performance.

Describe any problems you incurred while using the system.
What information presented by the system, if any, is irrelevant to the award selection process?
What other information should the system provide to aid in the award process?
Do you have any suggestions for future enhancements to this system?
Do you have any other comments or suggestions regarding the design or usefulness of this system?

FIGURE 3-8 -- BUYER QUESTIONNAIRE

panel's, the consolidation form was marked accordingly. Those purchase requests that deviated, were reviewed by the panel. If the panel determined the buyer (though not in agreement with their first choice) has made a reasonable alternative selection, then the response form was so noted.

Once all purchase requests were reviewed from all buyers, the number of matching transactions were summed with the number of reasonable transactions. The result was compared to the number of discrepancies minus the number of reasonable transactions. A reasonable transaction is defined as, 'an award selection, differing from that agreed by the

TEST CONSOLIDATION

Manual ☐ Automated ☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

PR: _____

	Agree	Dis-Agree	Reason-able		Agree	Dis-Agree	Reason-able
1.	☐	☐	☐	5.	☐	☐	☐
2.	☐	☐	☐	6.	☐	☐	☐
3.	☐	☐	☐	7.	☐	☐	☐
4.	☐	☐	☐	8.	☐	☐	☐

TOTALS

___	___	___	___	___	___	___	___
(1-4)			(5-8)	(1-4)			(5-8)

(SUM 1-4) (SUM (5-8)

FIGURE 3-9 -- TEST CONSOLIDATION FORM

panel as 'best', in vendor, quantity, and/or price; but still satisfies the intent of the purchase request without an increase in unit cost'.

The narratives collected from the buyers after the test were reviewed and trends documented in the final report.

Phase III.

Overview. Phase III was similar to Phase II. The difference being the eight buyers performing the testing did not routinely work with 5905 products, and they processed the requests using only the automated system.

Procedure. As before, the new group of buyers was given a set of purchase requests with a form to record the results attached to each. This time however, they were given only a complete set of fifteen purchase requests. Each request was processed using the automated system, and the results recorded on the attached form. When the set of requests was completed, the average time to process the requests was calculated, and the award information compared to the panel's selections. The percentage of reasonable responses was compared to the results of the first buyer group.

The error rate and average time to process of Phase II was compared to those of Phase III, looking for a significant difference in test results. Such a difference may suggest a lack of objectivity in awarding contracts brought to the evaluation by the buyers from the 5905 group.

For example, by working with the same vendors over an extended period of time, a buyer could 'know' certain traits of the vendors. Perhaps one vendor always quotes a lower price than another vendor, thus the buyer may improperly make the award decision without examining all information on file. Another example of bias that could develop as a result of prior knowledge is described as follows. A vendor has been historically poor in meeting

scheduled delivery dates. The vendor finally identifies the cause for the poor performance and corrects the situation. The problem vendor files maintained at DESC are been updated reflecting this change in performance. However, the buyer, aware of the past problems, awards to another vendor quoting a higher price. In this situation, the award was made without proper justification.

Conclusions.

Chapter III introduces the methodology followed in the research and development of this project. Specifically, it describes the method of development for the Decision Support System and the approach used for testing its utility. Chapters IV and V contain the details regarding the verification and validation of the results of this effort.

IV. Development

Overview

This chapter recounts the design and verification process used in the prototype development process. A multi-step development process was used to arrive at the 'final' system design. Those steps consisted of: user interviews, paper prototype development, initial prototype development and full prototype development. To insure the prototype would perform as intended, it was subjected to coding verification prior to validation at DESC.

The reader should be alerted to the following before proceeding. It is the researcher's belief that software development is as much art as it is science. The development process detailed in the following pages includes techniques developed and refined by the researcher through several years of personal programming and computer related experience.

It is not the intent of this project to identify or suggest 'the' proper method for software development. The intent is to document a successful transformation of user requirements into an effective system. The results obtained from validation will determine if this effort was successful.

Investigative Efforts

User Interview Process. To identify the expectations developed for the completed prototype, several interviews were conducted with the personnel at DESC. Meetings with DESC-P and other management level personnel were useful in identifying their desires for the system. Perhaps the most important outcome from these meetings was an understanding of VASPP and the relationship this development effort with it. (The VASPP concept was discussed in an earlier chapter and will not be repeated here.)

Once comfortable with management's views regarding the VASSP program, attention was directed to the buyers' needs of the system. Before a successful system could be designed, the buyers process for vendor selection had to be understood. Again, the interview technique was used to identify these requirements. Information was obtained by talking with several buyers and observing the vendor selection process. The researcher obtained further insight by actually performing the mechanics of the vendor selection process. The buyers provided 'real world' purchase requests and in-turn guided the researcher through the steps necessary to arrive at an award decision. This exercise assisted in clarifying the buyers data requirements and its useful presentation.

Results. Through this series of interviews and exercises, a better understanding of the vendor selection process was obtained, and, of how these efforts would later merge with a larger system. The following items influenced the prototype development efforts.

Inputs. Two pieces of information are required to identify the price offered by a vendor for a specific product. The first is the 'Type number'. The second is the quantity requested. With the Type number, the buyer can consult the vendor price list to identify if, one, a particular vendor offers the product for sale, and two, if it is for sale, the price per unit for a given quantity. The buyer can next compare the quantity requested with the quantity price breaks offered to obtain the best value for the customer.

It should be noted, the Type number identifies a specific component, the price lists however, are 'grouped'. A range of similar products carries the same pricing information. It is the product grouping that the vendors must identify in their price lists. As a result, the buyer looks not for a specific Type number in the price lists, but must identify the proper price group.

A third piece of information is also required before making the final award decision. 'Set-A-Side' is a term DESC uses to show only small businesses will be considered to receive

the award. As a result, vendors carrying 'large vendor' status are ineligible for selection consideration.

Outputs. A series of screens was designed to provide the user with the relevant award information. The buyers make their award decision on DESC Form 701. As this is the format they are accustomed to seeing, design of the prototype output screens was based on this form. The intent of this decision was improved user acceptance. It was felt the buyers would be less resistive to a new system if the system manifested itself in a form familiar to them. Details on the user screens will be covered later.

Paper Prototype

With the primary inputs and outputs of the system identified, a paper prototype was developed. This 'Desk-top' model consisted of flow charts identifying major logic concepts and sketching of the display screens.

Components. Figure 4-1 depicts the introductory flow chart developed. The purpose of these high flow charts is to bring structure to the software design. The detail in these charts is only sufficient to identify the major inputs to the system, its major processing blocks and the outputs provided to the buyer. It provides a functional view of the system's primary components and its major decision points.

Inputs. The inputs to the system were identified as follows:

a) NSN of the item requested;

b) The Quantity requested; and,

c) Identification of a Set-A-Side procurement. (In the form of Yes or No).

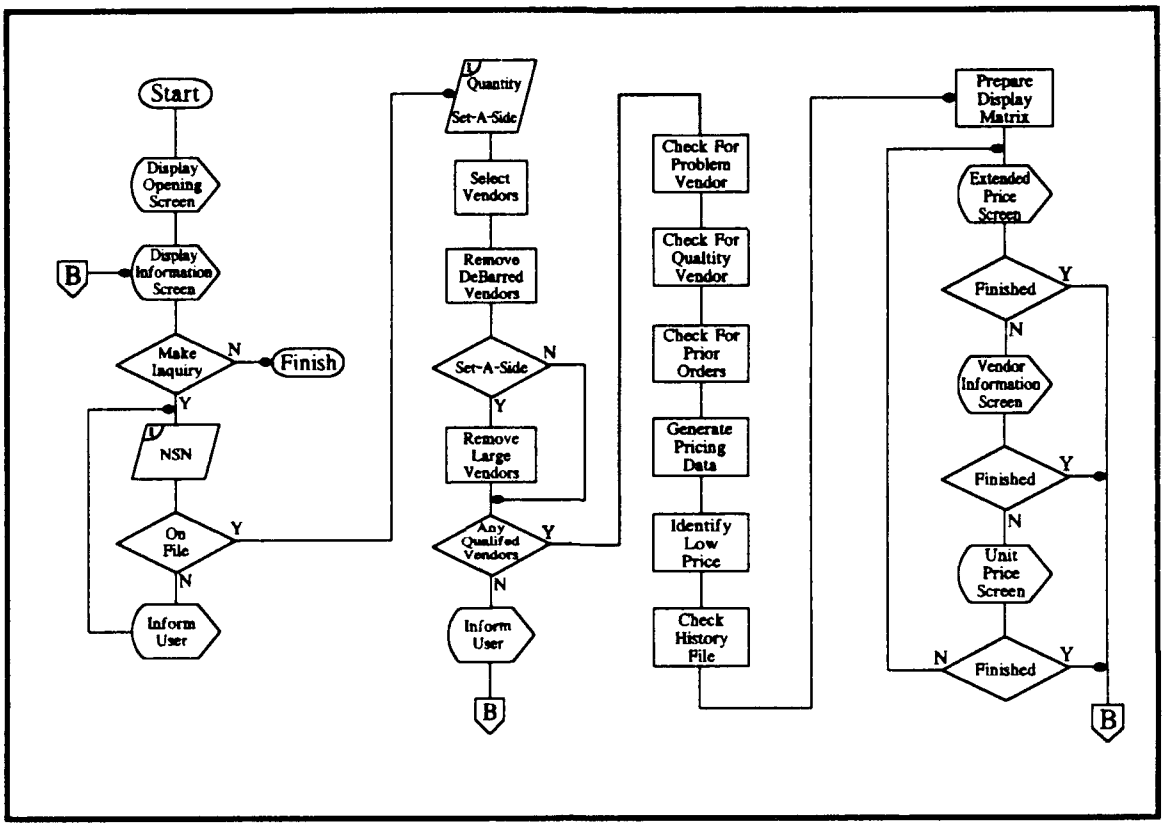


FIGURE 4-1 --INITAL FLOW DIAGRAM

Outputs. The outputs the buyers required from the system to make the award decision were identified as follows:

a) Vendors who offer the item for sale,

b) Minimum quantity the vendor will sell that satisfies the requirements of the purchase request,

c) The price of that quantity,

d) Whether the vendor offers an attractive price reduction for a larger order,

e) Total price of the purchase request,

f) Early payment discounts,

g) Freight Charges (FOB Origin/Destination), and

h) Past vendor performance data.

The user screens were developed using grid paper. The grids were representative of CRT screen size the buyer would be using. From these drawings, a programmer can identify the required coordinates of specific display data. This greatly eases later software coding.

Because of constraints of the computer CRT (Cathode Ray Tube), (sometimes referred to as the monitor) the information required is presented on three screens. The first screen is the Unit pricing screen. This screen will inform the buyer of the vendors bidding on a component, the quantity breaks offered by the vendor, and the price per unit for a specified quantity. The second screen is the extended price screen. Its design is based on the unit pricing screen. It differs from the unit pricing screen in that the prices displayed in the matrix represent unit cost times the quantity. The final screen is the detailed vendor information screen. This screen identifies the vendor by name, specific shipping information, discounts offered for prompt payment, and a record of occurrence in supporting data files.

Review. The Desk-top Model was presented to DESC management for their review and comments. Details of the proposed prototype operation were narrated. This included the identification of primary data files indigenous to the prototype and the data requirements from supporting systems. Screen descriptions were presented in the same sequence as the proposed prototype would generate them. Since DESC users offered no significant changes to the model, transition into the next phase of software development began.

Initial Prototype Development

Design Considerations.

Data Requirements. To be useful, the prototype must interrogate several data files for information. Some of these files reside on other computer systems, others reside on

printed paper tucked in a drawer. For those files that currently exist in an electronic format, the data were extracted and used without manipulation. For those files yet to be created, arrangement of the data elements to simplify integration with existing prototype software modules was emphasized. The major data files considered for use in the initial prototype are identified as follows:

1) NSN file. Lists all items for which the prototype contains pricing information.
2) Price file. Contains all pricing data for the items identified in the NSN file.
3) Vendor file. Contains, by cage code, vendor specific information, i.e., delivery time, type of vendor, and cage code for those vendors providing bids on the items in the NSN file.
4) DCRL file. Contains, by vendor, specific details of past performance problems.
5) Due-In file. Contains, by NSN, information on products ordered but not yet delivered.
6) History file. Contains, by NSN, past procurement information for a specific product.
7) Quality file. Contains, by cage code, those vendors identified in DESC's quality vendor program.

Data Files. The data files used in the initial prototype were for developmental purposes only. They were not complete. Some data files contained only a few representative records from the real world data files. Other data files were constructed before the actual data files became available. In this instance, the necessary data element was contrived based on the information that would be required for successful implementation. This short coming will be discussed further in the next section.

Data Structure. Certain characteristics of the raw data were exploited to simplify prototype design.

For example, there is a one for one relationship of Type number to National Stock Number (NSN). The NSN appears at the top of each purchase request. The NSN is also a key field used to interrogate other data files currently maintained at DESC, for example, the History and Due-In data files. The prototype was designed to request the NSN instead of the Type number. This decision was made as the NSN is readily available to the buyer, and it would eliminate a cross-referencing step by the system.

A second code appears in virtually every vendor related operation in the current system. That code is the Cage code. The cage code is a five position alpha-numeric element that uniquely identifies a vendor. This alias becomes a shorthand the buyers use to refer to a specific vendor. The function of the cage code in the prototype will be covered later.

To reduce the amount of data storage space required for each item the following procedure was adopted. Instead of storing a price schedule with each item, a code was devised to identify a unique set of prices. All products from the same vendor with the same pricing scheme are assigned the same code. This technique saved one hundred ninety-four bytes of storage space for each part on file. The resulting NSN data file record length is only thirty-four bytes long. When the prototype integrates into VASPP, it must rely on vendor pricing information stored in a central data file. The data contained within this data file will be submitted and maintained by the vendor. The structure of this database is not yet determined. An outcome of this research will be the minimum data elements the vendors must supply for successful implementation. The complete details of the pricing data structure used and a description of each data element used can be found in Appendix C. (This appendix contains the data description for all data bases used.)

The prototype must search, without intolerable delay, a data file containing thousands of records (assuming at least one record per item). For example, the MilSpec 55182 items, a single subset of the items in Stock Class 5905, contains over 50,000 entries. To expedite this process, two design features were incorporated into the system. The first

was to minimize the elements contained in the larger data files. Reducing the size of the data file, reduces the number of bytes the system must transfer between the storage area and the processing unit where it can analyze the information.

The second technique makes use of indexed files wherever possible. Indexing is essentially a refinement to minimizing file size. The concept of an index file is as follows. A separate file is created containing two elements. The first element is called the key field. In this example, it is the NSN. The second field contains the position (the record number) in the main database that contains the Key element. The system rapidly searches the smaller index file for the Key (the NSN). Once located, it can make a direct request for the data record of interest in main data file.

Software Development.

Methodology. The software was designed in modular format, taking care to make each unit as independent from the other modules as possible. This technique leads to easier testing and modification (27:62). As each module was developed, it was checked for proper operation; examining both extremal and special values. Unexpected results were corrected prior to continuing with the next stage of program development.

Beyond generating the program code, internal documentation was concurrently produced. With the task of the software manager in mind, these programming notes were placed in the code to assist in future debugging or program modifications.

Environment Selection. Through interviews, it was learned personnel in DESC's automation department, DESC-Z, were familiar with Ashton-Tates software known as *dBase III Plus*. One of this program's main strengths is its ability to assist the user in performing complex database manipulations and retrievals. The programming approach to the problem, being heavily reliant on data retrieval (the final prototype integrates eleven

separate data files), and the author's own acquaintance with the program, made *dBase III Plus* a natural choice for use on this project.

System Description. The following narrative describes the operational process designed into the prototype. Only major actions performed by the prototype are covered. (The reader may find it useful to refer to the initial flow charts of Figure 4-1).

The first thing the user sees when starting the system is a welcome screen (Figure 4-2). This screen simply identifies the software and asks the user to proceed when ready. The second screen, Figure 4-3, provides a brief description of the software and informs the user of the inputs required to use the system successfully. The user has the opportunity to exit the system at this point or continue to the next screen.

The third screen, Figure 4-4, is the first of the input screens. Prompts for information are presented sequentially. The first item requested is the NSN. Once entered, the system accesses the NSN data base. If the NSN input by the user is not on file, the user is informed and allowed to reenter the requirement (Figure 4-5). Once the user enters an NSN contained in the data base, the system prompts for the quantity required (Figure 4-6). The system

```
Welcome To The

A U T O M A T E D
V E N D O R
S E L E C T I O N
A S S I S T A N T

Beta Version 2.5

Press Any Key To Continue
```

FIGURE 4-2 -- WELCOME SCREEN

verifies a numeric value was entered and presents the final prompt, Set-A-Side (Figure 4-7). If the purchase request is identified to be set-a-side for small business, the user enters a 'Y'. If not, the user enters an 'N'. If the user is unsure, the system will accept a '?', and treats it as an 'N'. This provides the user with all qualified vendors.

```
The Automated Vendor Selection Assistant
selects the vendor(s) who have competitively bid
on the item of interest.

To proceed, you must know the item's NSN
and the quantity required.

Do you wish to continue? <Y/N>Y
```

FIGURE 4-3 -- PROGRAM INFORMATION SCREEN

```
Enter the NSN of the item to be procured

5905-01-009-5543

(Press <CR> when complete)

Press <ESC><ESC> to Quit the Assistant
```

FIGURE 4-4 -- NSN INPUT SCREEN

```
Enter the NSN of the item to be procured

5905-01-009-5555

This NSN is not on file

(Press <CR> when complete)

Press <ESC><ESC> to Quit the Assistant
```

FIGURE 4-5 -- NSN NOT ON FILE SCREEN

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90    EA.

(Press <CR> when complete)

Enter <0><CR> To Quit
```

FIGURE 4-6 -- QUANTITY INPUT SCREEN

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90 EA.

(Press <CR> when complete)

Is this procurement Set-Aside for small business? <Y/N/?>
```

FIGURE 4-7 - SET-A-SIDE SCREEN

The system now has all the information required for processing. It scans the NSN data base to locate all vendors who have bid on the item. Each vendor's pricing data for the item are transferred to a temporary data file. The vendors in the temporary data file are then compared to the DCRL file. If a vendor is identified in the DCRL file as 'DeBarred' it is removed from the temporary data file. (A DeBarred vendor is ineligible to receive any contract awards.)

If it is a set-a-side procurement, the temporary file is scanned again, this time looking for vendors coded as 'large vendors'. Those vendors are removed from the file. After this two step process, the only vendors remaining in the temporary file are those that are eligible to receive the contract award.

If, after completing these two procedures, there are no vendors qualified to receive the award, the buyer is informed (Figure 4-8) and returned to the information screen. The buyer can either fail to make the award or can relax the requirements and reprocess the request.

```
No qualified vendors
are on file matching
your requirements.

Press Any Key To Continue
```

FIGURE 4-8 -- NO QUALIFIED VENDOR SCREEN

Now that the vendors bidding on the item are known, the system makes several background checks. The DCRL and quality data files are scanned. The vendor's cage code is the Key element used to perform the look-up. The Due-In file is checked, using the NSN of the items. The results of these searches are recorded by setting specific flags assigned to each vendor. If a 'hit' was made, the appropriate flag is set to 'True'.

The system now focuses on the pricing data. It calculates the minimum quantity of items that can be ordered, while still satisfying the Purchase Request. Each vendor has different minimum quantity requirements. Some vendors will only sell in specific lot sizes. In this case, the system increases the order quantity to the value of the next lot size. Sometimes the lot size varies with the quantity ordered. The system can adjust the order quantity accordingly. One final check made for each vendor is the minimum order dollar amount. Again, if the requested quantity times the unit price of the item is below the minimum dollar amount, the order quantity is increased to the minimum quantity that will meet the minimum dollar order threshold.

Once the appropriate minimum order quantity for each vendor is determined, that and any additional relative pricing data are transferred into a second temporary storage file. In this file, the order quantities are sorted in ascending order, and the lowest price offered is identified. The low price is compared to past purchases. Appropriate flags are tripped if any deviations are found. These flags will be used to trigger the display of appropriate warning messages for the buyer.

From this temporary file, the data are transferred into a series of memory variables. These memory variables are organized to form a two-dimensional table. The information presented in the user pricing screens comes directly from this memory table. The columns of the table represent the various quantities of the product that can be purchased. The rows identify the vendors that offer the item for sale. At the intersection of a given row and column is the pricing data related to that specific vendor/quantity intersection.

The eligible vendors offering the item for sale have been identified. Performance records have been checked. The *minimum* quantity the vendor is willing to sell, that meets or exceeds the quantity requested, is identified. Once the lowest total price that satisfies the purchase request has been identified, the data are now ready for display.

The next step is for the system to present the output screens to the user. The first screen presented is the extended price screen (Figure 4-9). This screen informs the buyer which vendors sell the product, and their total price for a given quantity of the product. The vendor cage code is color coded corresponding to its appearance in the DCRL file, the Due-In file, or the Quality file. The logic governing the color code assigned has a designated order of hierarchy. Color coding for a vendor found in the Due-In file will override an appearance in the Quality file. Also, appearance in the DCRL file will override all other color coding

Pricing information is also color coded. The lowest total price to satisfy the purchase request is highlighted bright green. If there is a tie between vendors, both low quotes will be highlighted. If the low price is 'considerably' lower than the next lowest vendor's price,

the low price is highlighted yellow. (This feature alerts the buyer that the price may be unrealistic). What identifies a price as considerably lower is controlled by the Model data file. Additional elements of the model subsystem will be discussed later.

Pricing Data For: 5905-01-009-5543

	90	100	200	250	300	500
56856	128.70	114.00		255.00		450.00
0BTU6		25.50				
6S313		92.00	170.00		234.00	385.00
00001						136.00

VENDOR: Problem Vendor
Items Due-In From Vendor
Quality Vendor

PRICE: Price May Be To Low
Low Price

<SPACE BAR> To Toggle Screens

<ESC> When Finished

FIGURE 4-9 -- EXTENDED PRICE SCREEN

Pressing the space bar toggles to the vendor screen (Figure 4-10). This screen displays the vendor delivery information and informs the buyer if the vendor is identified in any of the supporting historical performance files.

Pressing the space bar again brings up the unit pricing screen (Figure 4-11). This screen is identical with the extended pricing screen except the prices in the matrix are 'per piece'.

The buyer can continue toggling between these screens until the award decision is made. Pressing < ESC > will return the system to the program information screen. Once back to that screen, the buyer can either enter a 'Y' to make another inquiry, or an 'N' to terminate the session.

Vendor Data For: 5905-01-009-5543

CAGE	VENDOR	DISC	NET	DEL	FOB	SM	PROB	CDCS	D-IN	QUAL
56856	Vamistor Corp.		/		D		X			
0BTU6	T. T. I., Inc.	0.5%	20/30	120	D				X	
6S313	G & A Sales	1.0%	10/	220	O	Y				X
00001	Hamilton Avnet Electronics		/30	250	D					

<SPACE BAR> To Toggle Screen <N> Net Price <ESC> When Finished

FIGURE 4-10 -- VENDOR SCREEN

Pricing Data For: 5905-01-009-5543

	90	100	200	250	300	500
56856	1.4300	1.1400		1.0200		0.9000
0BTU6		0.2550				
6S313		0.9200	0.8500		0.7800	0.7700
00001						0.2720

VENDOR: Problem Vendor
Items Due-In From Vendor
Quality Vendor

PRICE: Price May Be To Low
Low Price

<SPACE BAR> To Toggle Screens <ESC> When Finished

FIGURE 4-11 -- UNIT PRICE SCREEN

The Model Component. In addition to the discrimination and mathematical calculations already discussed, the model data file controls how and when specified information is presented on the screen. The values contained within the model can be changed at the request of the management. At this stage of development, the model controls the following display attributes:

a) Low Price Flag. This element alerts the buyer to the fact that the vendor is quoting a price that is significantly lower than the competitors. When tripped, the low price will be displayed in yellow on the pricing screens.

b) No History Flag. The number stored in this element represents a dollar threshold value. If the unit price of an item exceeds this amount, and there is no historical purchase information on file, a message is printed on the output screens.

c) Exceeds History Price. The prototype compares the item's current unit price with the unit price of the item when last ordered. If the current unit price exceeds the last unit purchase price by more that the percentage contained in the element, a message is presented to the buyer.

d) Excessive Contract Value. If the total value of the award exceeds the dollar amount stored in this element, a warning is printed on the screen informing the buyer the limit for small contract award has been exceeded.

e) Variation. On the price list the vendor identifies any variations in shipping quantity. The vendors claim authorization to ship a quantity within a stated percentage of the contract quantity. For example, a vendor may claim a variation of two percent. If the contract was written for one hundred units, the vendor could ship only ninety-eight units and still satisfy the contract. The prototype checks this variation, internally increments the quantity to account for the variation, and computes the resulting award value of the contract. If the award value exceeds the excessive contract value, (defined above), a warning is provided on the user screens.

Review. A formal presentation of the prototype was given to the DESC-P, supporting management, and selected buyers. The purpose of this review was to insure the overall design of the prototype conformed to DESCs expectations. This pre-inspection was necessary to avoid the possibility of extensive programming hours consumed in unproductive areas. This however, was not so. The initial prototype was reviewed with great enthusiasm. The design met or exceeded their anticipations for this first review. Minor modifications, discussed below, were suggested. Without reservation, the initial prototype was accepted and plans were made to proceed.

Full Prototype Development

Having gained approval of the basic design, attention was turned to developing a complete working prototype.

Requirements Re-evaluation. To pin down the exact characteristics the next prototype required, a meeting with several buyers and management personnel was scheduled for the following week. At this meeting comments were solicited regarding the current system design. A detailed examination of each screen was made. Attention was given to the data presented, making sure all information required to make the award decision was accounted for. Also critiqued was the presentation format of for each screen. Any changes suggested were recorded. Documented in the next section are those changes.

Modifications. Unless otherwise noted, the fully developed prototype maintains all the operational characteristics described for the initial prototype (see Initial Prototype Development for details). Changes to the prototype fell into two categories, embellishments of existing features and enhancements of new features suggested by the review panel.

Embellishments. Several features of the initial prototype were not yet functional prior to its review. Two user screens had yet to be developed. The first was the DCRL information screen.

The prototype syntax refers to this screen as the 'Problem Vendor Screen'. If a bidding vendor appears in the DCRL data file, the cage code is highlighted red. To see the information contained in the file, the buyer enters 'P' from any of the user screens and the discrepancy details for that vendor appears on a new screen (Figure 4-12). When the buyer finishes reviewing the file, he/she is returned to the previous user screen.

```
                                6S313

SECOM ELECTRONICS CORP          | 89/11/15 D  Pre-Award Survey Required
12 PROGRESS PLACE               | **/**/**
JACKSON NJ    08527-3002        | **/**/**
                                | **/**/**
                                | **/**/**
                                | **/**/**

        SEE P IOM 20 OCT 89 RE UNSATISFACTORY PERFORMANCE
        RECOMMEND DETERMINATION OF NONRESPONSIBILITY,
        PREAWARD SURVEY, DCAS ADMINISTRATION

Press any key to continue...
```

FIGURE 4-12 -- PROBLEM VENDOR SCREEN

The second screen not developed for the initial prototype was the Due-in Screen. The Due-In data file was not available for review when the initial prototype was developed. Request to DESC for a copy of their actual file was unsuccessful in providing a product that was usable for this project. A file was available that identified the items Due-In, but there was no linkage made to the vendor responsible for filling the order. Because of the inability to track an order to the vendor providing it, this portion of the prototype became

dysfunctional. This problem was reported to DESC. After discussion with management and buyers, the determination was made that this specific information was not critical to the award decision process. It was information that would be useful if available to the buyers, but its absence would not critically impede the decision process.

Yet another screen required in the system was an award screen. Once the buyer selects the best vendor to receive the award, a DESC Form 800 must be filled out. The Award Screen, (Figure 4-13), pulls together all the information required to completed this form.

Vendor:	Remit To:
G & A Sales 2854 Blue Rock Road Cincinnati, Ohio 45239	Same

Cage: 6S313 State Code: 39 Source Type: A

Discount: 1.000% In 10 Days Variance: + 0% - 2%

Delivery Time: 220 Days FOB: O RFCC Code: Z

	90	100	200	250	300	500
Unit Price		0.9200	0.8500		0.7800	0.7700
Ext. Price		92.00	170.00		234.00	385.00

Press <P> For Previous Screen
Any Other Key When Finished

FIGURE 4-13 -- AWARD SCREEN

In the fully functional prototype, historical information regarding past buys was not only examined, as in the initial prototype; but also displayed for the buyers' use. If procurement information for the item is found in the Historical data file, the most recent purchase information is displayed in the upper right hand corner of either Pricing Screen (see Figure 4-14). This provides the buyer with not only the vendor and price of the last order, but gives the buyer an estimate regarding the rate of consumption.

*** UNIT PRICE EXCEEDS HISTORY ***

Last Purchased On 80302
From 91637 For $0.27

Extended Pricing Data For: 5905-01-009-5543

	90	100	200	250	300	500
56856	128.70	114.00		255.00		450.00
6S313		92.00	170.00		234.00	385.00
7K545						136.00

VENDOR: Problem Vendor Info
CDCF Vendor Info
Quality Vendor

PRICE: Price May Be To Low
Low Price

<U> Unit Pricing <A> Award Screen <C> CDCF Vendor Detail
<V> Vendor Information <Q> Quit <P> Problem Vendor Detail

FIGURE 4-14 -- 'FINAL' EXTENDED PRICE SCREEN

Enhancements. The next paragraphs identify changes made to the prototype as suggested by the review panel.

Customer Depot Complaint File (CDCF). A second 'problem' file was identified. The CDCF was a listing by NSN of items that have had complaints registered. The complaints can be anything from substandard product performance to mismarked packaging. The prototype incorporates this data file using the following method. First, it checks for the existence of the NSN in the CDCF data file. If the NSN exists, a search is conducted within the NSN for a cage code matching any of the bidding vendors. If a bidding vendor is found to have a complaint filed on the product in question, the CDCF flag is set for that vendor. When the cage codes are displayed on the user screens, the cage is color-coded violet. The buyer can review the contents of he relevant CDCF records using the CDCF screen, Figure 4-15.

Required Delivery Date (RDD). It was suggested the buyer make an additional input to the prototype and enter the RDD date. The Required Delivery Date

is the Julian date the item is required for use. It can be found on the last page of the purchase request.

```
                                6S313

DISC --> Q5
CAUSE -> CN CONTRACTOR NONCOMPLIANCE (PRIME CONTRACTOR)
DISP --> AD DALE - CAT I - DAC FROM C/C "K" TO C/C "H" W/MGMT CODE "
CORR --> AO POC BETTY GEBELE/OSIB/AV986-6486.

Press any key to continue...
```

FIGURE 4-15 -- CDCF SCREEN

The input screen was modified to accommodate this additional input (Figure 4-16). After the buyer enters the quantity of the item required and before he/she indicates the Set-A-Side status, the system now asks for the RDD date. The prototype performs a validation check on the buyer's input. The input is a five digit numeric. The first two positions represent the last two digits of the year. The next three positions represent the day of the year. Because DESC habitually receives purchase requests with required delivery dates prior to the day of receipt, the system will accept one year prior to the current year. The system will accept the day input if it is a number between one and three hundred sixty-five inclusive. (Three hundred sixty-six is accepted if the year entered is a leap year.)

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90 EA.

(Press <CR> when complete)

What is the RDD date? 92105
```

FIGURE 4-16 -- REQUIRED DELIVERY DATE INPUT SCREEN

The vendor information screen and the model were modified to take advantage of this information (see Figure 4-17). An additional element was added to the model for administrative lead time. This is the in-house time required to process the award paper work. The system calculates the current Julian date. To that, the administrative lead time is added. Also added is the vendor's stated delivery time. The current date, plus the Administrative Lead Time, plus the Delivery Time is the Projected Delivery Date. The delivery projection is compared against the required delivery date. If a vendor can deliver on or before the required delivery date, the projected delivery date is displayed as green. If a vendor cannot meet the required delivery date, the projected delivery date is displayed in red.

Also modified, was the coding of the fully developed prototype to give the buyer better control and access to the user screens. In the initial prototype the user toggled through the screens using the space bar. The order of presentation was fixed. The prototype now held six user screens, and toggling was unsatisfactory. A menu structure was developed allowing the user to move directly to the menu of choice.

Vendor Data For: 5905-01-009-5543

CAGE	VENDOR	DISC	NET	DEL	FOB	SPEC	PROB	CDCS	QUAL
56856	Vamistor Corp.		/	91356	D				
6S313	G & A Sales	1.0%	10/30	92091	O				
7K545	Hamilton Avnet Electronics		/	92121	D				

User's Options:
<U> Unit Pricing
<E> Extended Pricing
<A> Award Screen
<Q> Quit
<C> CDCF Vendor Detail
<P> Problem Vendor Detail

FIGURE 4-17 -- 'FINAL' VENDOR SCREEN

Data Files. The data files used with this prototype were extracted or created from the actual files found at DESC.

1) NSN. To create the NSN data file, The programmers at DESC-Z Generated an extract from their master file. The extract contained only those NSNs associated with MilSpec 55182 item. Even with this reduced subset, it took nine diskettes to transfer the data.

2) Price. This data file was created from the hard copy price lists provided by the vendors. Only those vendors who submitted requests for MilSpec 55182 items were included. However, each vendor's list was entered in its entirety.

3) Vendor. All vendor specific information required by the system is stored in this file. All vendors bidding on MilSpec 55182 items are included.

4) DCRL. The DCRL data file is an image of the complete master data file at DESC. Thus, it contained all vendors DESC recognizes as 'problem vendors', and identifies their transgressions.

5) CDCF. Because the size of the master file inhibited transfer to floppy diskettes, a subset was used. Again, NSNs associated with MilSpec 55182 were extracted.

6) History. Because the size of the master file inhibited transfer to floppy diskettes, a subset was used. Again, NSNs associated with MilSpec 55182 were extracted.

7) Quality. This data file was created from the Quality vendor list maintained at DESC and entered in its entirety.

Design Problems. Two design problems surfaced while developing the prototype. One problem dealt with the data structure and one problem dealt with the program coding.

Data Structure. The most challenging aspect of the development efforts rested with the pricing data itself. For a relational data structure to work, the data must be organized in a standardized format. That is not so with the pricing information provided by the vendors.

There was no commonality within the product groups. The quantity at which price changes occurred were inconsistent. Some vendors had a minimum order quantity, other vendors had minimum dollar amounts. Some vendors would sell individual units, while others would only sell individual units over a certain quantity. Still other vendors would only sell in specified lot sizes.

Consistency had to be brought to these variances. The design of the pricing data file achieved most of this goal. It uses three fields to identify a price: the minimum quantity for a grouping, the maximum quantity for a grouping, and the unit price for that grouping. There are ten sets of these price groupings. Therefore, a vendor can provide up to ten different quantity price breaks for a product.

The vendor information file is used to solve the problem of lot size and minimum order quantity. Elements were added to the file structure for these two values. When calculating the pricing information, the prototype checks these two elements and responds according to their contents.

Program Coding. Most of the coding required to produce the prototype was conventional in nature. The use of indexes, and linking several data files with key fields, are typical data file procedures. The most challenging feature of the coding was the display matrix.

Following are the steps developed to organize the matrix. First, vendor pricing groups that contain insufficient quantities to satisfy the purchase request are eliminated. The remaining pricing data are transferred to a temporary data file. This process is repeated for all bidding vendors. With all pricing information in the temporary file, it is arranged by ascending order quantity. The first six quantities are copied into the memory display matrix. The remaining, if any, additional quantities are removed from further processing.

Next, the pricing information is transferred into the display matrix. This does not have to be such a challenge, but *dBase* does not provide for array variable identification. As a result, each cell in the matrix must be uniquely identified and addressed individually. The prototype examines the pricing information in the temporary data base, locates the proper vendor row in the matrix and finally finds the proper column to place the price.

The program coding required to perform the above steps can be found in the program PrepVen, line numbers 124 through 215, and 301 through 384 (Appendix A).

Verification

Focus. Once all desired functions and features of the prototype were coded, the official verification phase could begin. Some additional comments on the software development are in order at this time.

It is worthwhile to revisit the idea of software debugging. While it is a noble gesture to strive for error-free coding, proving it is so, is another matter. "If the objective of testing were to *prove* that a program is free of bugs, then not only would testing be practically impossible, but it would also be theoretically impossible" (6:12).

Verification of the system was a multi-step process. The first phase involved desk-top review of the program code. The second step incorporated was path verification procedures.

Desk-top Review. Prior to conducting the review, the program code was analyzed by *Snap*, a public domain documenting program for *dBase* source files. By informing *Snap* of the first program module in the series, it is able to analyze the entire program structure. Assuming there are no logic errors located, *Snap* continues with the documenting process. Through a series of user selectable switches, it can convert the case of the variables and reserved words (i.e., forces *dBase III Plus* reserve words to be printed in capital letters), tab indentured code, number the program lines and create a variable cross reference table.

With these enhancements, it was possible to perform an in-depth desk-top review of the program code. Desk-top review consists of manually examining each line of code, looking for peculiarities. Some details examined were: submodule sequencing, redundant variables, and documentation completeness. Discrepancies were corrected and a final copy of the program code produced. (Appendix A) (Figure 4-18 depicts the final system design).

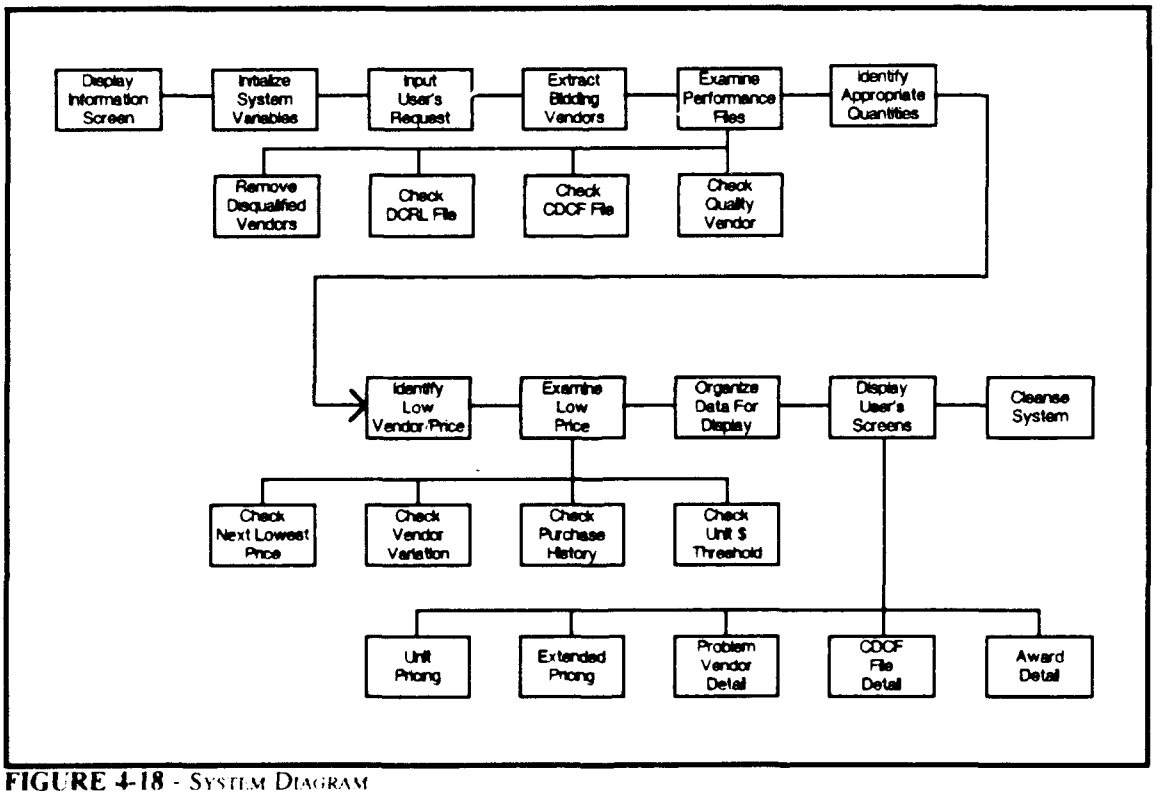


FIGURE 4-18 - SYSTEM DIAGRAM

Path Verification. After the desk-top review, the code rested in its 'final' format. The code was inspected by hand and all paths through the system identified. Through inspection, the extremal and special variables were identified for each path.

With the program paths identified, each path was tested for proper execution by means of altering data input (either through the user prompts or by altering the contents of the database). Where necessary, values were artificially assigned to program variables. Any unexpected results in program execution were examined to determine their origin, and as necessary, corrections made to the program code.

User Review. With the prototype complete and errors checked, a meeting was scheduled with personnel at DESC. A demonstration of the system was provided along with a description of the enhancements incorporated since the last formal review. As before, the prototype was well received. While some 'future' enhancements were identified, none offered would affect the functionality of the prototype (i.e., color changes on the screen), or retard the validation phase.

With DESC management satisfied with the fully developed prototype, the design was 'frozen'. Efforts now turned to the formal validation process. Details of this step can be found in the next chapter.

Conclusion

This chapter recounts the development and resulting verification process of the prototype. A multi-stage development approach was used. The basic process included defining the requirements, designing a system to match the requirements and, building the design. The importance of close coordination with the user cannot be over emphasized. Without user input, the development process could have easily fallen short of expectations.

V. Validation

Overview

This chapter details the sequence of events used to test the prototype and analyzes the data generated from those tests. As described in Chapter III, the test plan incorporated three phases. That scheme was adhered to without modification. Two problems were discovered after reviewing the data, requiring further analysis beyond that described in Chapter III. Recorded in the pages that follow are the details of all testing and analysis.

Phase I

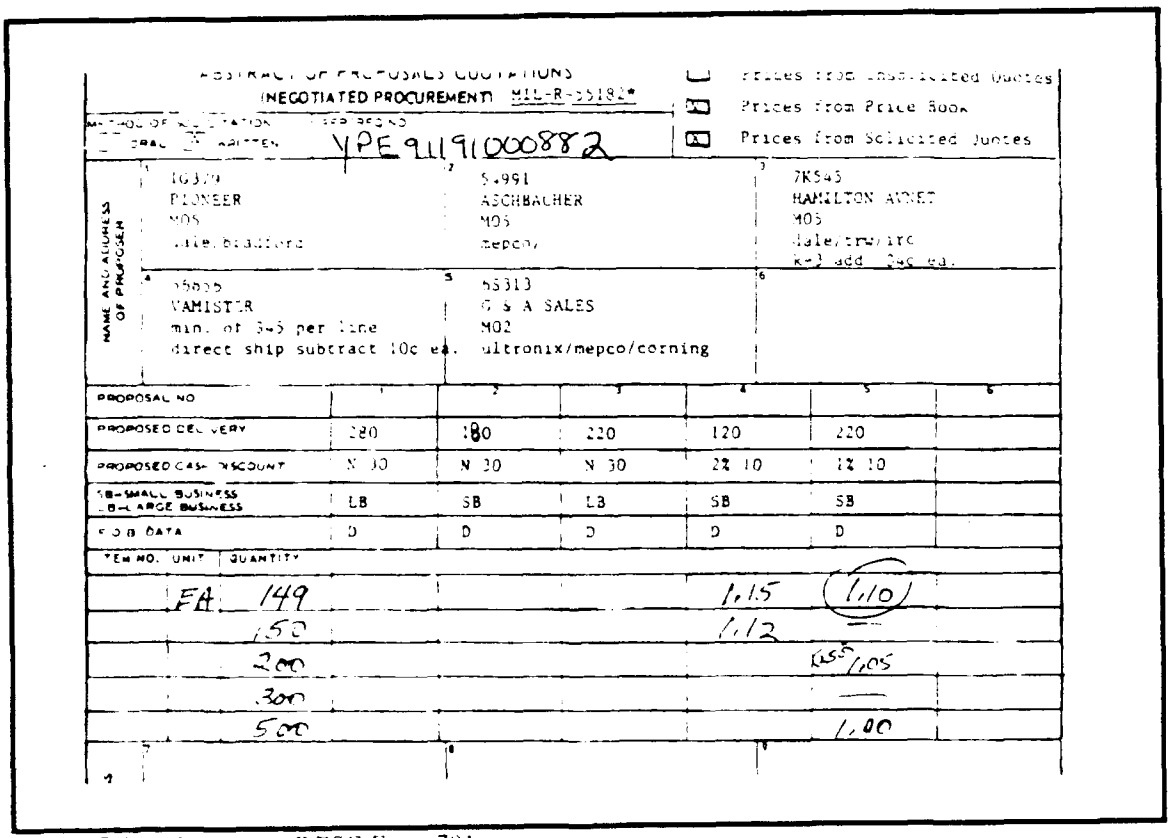
Synopsis. The purpose of Phase I is to determine the completeness and accuracy of the information presented by the prototype. The question is asked, 'Does the system provide the correct information?' This phase of testing was completed on July 22, 1991.

As requested, DESC provided two people with expertise in the award selection process of MilSpec 55182 items. The buyer chosen, Ms. Racine Taylor, has worked in this area for five years. Ms. Carol Vance is the contracting officer for MilSpec 55128 items and was the second member selected to serve on the Expert panel.

The researcher provided the panel with approximately forty minutes of background information and prototype training. This included outlining the procedure used to process the purchase requests using the prototype and how to complete the forms developed for this test. It was stressed that time was not being measured in this phase of testing. The only criteria of interest was the accuracy of the information presented by the prototype and the correct vendor selection information for each purchase request.

Testing. Thirty purchase requests were provided by DESC for testing. The purchase requests used were selected from those awaiting buyer processing.

The panel began the testing process by selecting a purchase request from those provided. First, the request chosen was processed using the existing manual system. The panel used DESC Form 701 to document this process (See Figure 5-1). Provided with each purchase request was a Panel Selection Form. The panel annotated the vendor chosen in the Selection section of this form (Figure 5-2).



ABSTRACT OF PROPOSALS QUOTATIONS
(NEGOTIATED PROCUREMENT) MIL-R-55182*

Prices from Price Book
Prices from Solicited Quotes

METHOD OF SOLICITATION: ORAL / WRITTEN
REQ SEQ NO: YPE9119100088 2

NAME AND ADDRESS OF PROPOSER

1	2	3
16379 PIONEER MOS [illegible]	54991 ASCHBACHER MOS mepco/	7K543 HAMILTON AVNET MOS dale/ctr/irc kel add 2sc ea.
4	5	6
55005 VAMISTOR min. of 3+5 per line direct ship subtract 10c ea.	65313 G & A SALES MO2 ultronix/mepco/corning	

PROPOSAL NO	1	2	3	4	5	6
PROPOSED DELIVERY	280	180	220	120	220	
PROPOSED CASH DISCOUNT	N 30	N 30	N 30	2% 10	2% 10	
SB-SMALL BUSINESS LB-LARGE BUSINESS	LB	SB	LB	SB	SB	
F.O.B. DATA	D	D	D	D	D	
ITEM NO. / UNIT / QUANTITY						
EA 149				1.15	1.10	
150				1.12	—	
200					1.05	
300					—	
500					1.00	

FIGURE 5-1 -- COMPLETED DESC FORM 701

Having selected a vendor using the current manual process, the panel next used the prototype to choose the appropriate vendor. After entering the National Stock Number (NSN) of the part required, the quantity requested, and the Required Delivery Date (RDD), the prototype interrogated its various databases. It then presented the panel with the Net Price Screen. (Refer to Chapter IV for a discussion of the various user screens.)

Award Information

Preferred ☐	Manual System			Tie ☐	Preferred ☐	Automated System		
	Vendor Selected:	Quantity:	Extended Price:			Vendor Selected:	Quantity:	Extended Price:
1.	65313	149	163.90		1.	65313	149	163.90
2.					2.			
3.					3.			

FIGURE 5-2 -- COMPLETED PANEL SELECTION FORM

From this point, the panel could consult the net price screen and other prototype screens as required. To review, the remaining screens are: extended pricing data, vendor delivery data, problem vendor data, customer complaint data, and award detail data. These screens, in concert, provide the buyer information on which to base the award decision.

Again, the panel documented the award information on the attached Panel Selection Form. This two-step process of vendor selection was repeated for the remaining twenty-nine purchase requests in the test set. Discussion of the results of this test follows in the next section.

Once processing of all thirty purchases was complete, each panel member received a questionnaire. The responses provided on the completed questionnaires can be found in Appendix G.

Results. Table 5-1 lists the data obtained from the first phase of testing. As documented in the table, the information from the prototype system provided the same results as the current manual process in all but two cases. On those two occasions, the pricing data base contained an error. The researcher, in reviewing the vendors pricing data, misinterpreted the vendors price list. It should be noted that the prototype displayed the pricing information

TABLE 5-1

PHASE I RESULTS

PHASE I RESULTS			
Purchase Request	Manual Preferred	Tie	ProtoType Preferred
YPE91195001054		X	
YPE91195001053		X	
YPE91191000882		X	
YPE91191000881		X	
YPE91191000877		X	
YPE91191000876		X	
YPE91191000875		X	
YPE91188000919		X	
YPE91191000874		X	
YPE91188000914		X	
YPE91188000894		X	
YPE91188000893		X	
YPE91188000892		X	
YPE91188000890		X	
YPE91188000887		X	
YPE91188000885		X	
YPE91188000883		X	
YPE91188000881		X	
YPE91151000352		X	
YPE91188000880		X	
YPE91188000879	X		
YPE91188000878		X	
YPE91188000877		X	
YPE91177000268		X	
YPE91175000178		X	
YPE91148000183		X	
YPE91195001056		X	
YPE91157000145		X	
YPE91146000673		X	
YPE91151000115	X		

as intended by the researcher. However, the experienced panel quickly revealed this misunderstanding.

Analysis. A Sign Test was used to perform statistical analysis on the data for this phase (Figure 5-3). Table 5-2 contains the data set used in the analysis. A 1 in both the manual and automated columns indicates a tie. A 1 in one column and a 0 in the other indicates

TABLE 5-2

PANEL PREFERENCE DATA

PANEL PREFERENCE		
CASE	MANUAL	AUTOMATED
1	1	1
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	1
9	1	1
10	1	1
11	1	1
12	1	1
13	1	1
14	1	1
15	1	1
16	1	1
17	1	1
18	1	1
19	1	1
20	1	1
21	1	0
22	1	1
23	1	1
24	1	1
25	1	1
26	1	1
27	1	1
28	1	1
29	1	1
30	1	0

the panel preferred one method over the other. The question being tested is, 'Is there a difference in the award selection using the manual process versus using the prototype?' ''The null hypothesis tested by the sign test is that the median of the differences is zero'' (4:208).

```
STATISTIX 3.5
ID: Panel Preference
SIGN TEST FOR MANUAL - AUTOMATED
NUMBER OF NEGATIVE DIFFERENCES              0
NUMBER OF POSITIVE DIFFERENCES              2
NUMBER OF ZERO DIFFERENCES (IGNORED)       28
PROBABILITY OF A RESULT ASOR MORE EXTREME THAN OBSERVED          0.2500
A VALUE IS COUNTED AS A ZERO IF ITS ABSOLUTE VALUE IS LESS THAN  1.0E-0005
CASES INCLUDED 30    MISSING CASES 0
```

FIGURE 5-3 -- SIGN TEST FOR PANEL PREFERENCE

With a computed one tailed p-value of .2500, the null hypothesis cannot be safely rejected. In other words, there is insufficient statistical evidence to show the vendor selections differ between the two methods. Therefore, the prototype is believed to be providing the panel with sufficiently correct information on which to base the award decision.

The panel was asked to respond to a questionnaire. The final question the panel responded to dealt with the utility of the prototype. Both panel members indicated the system did have the potential to aid the buyer in the vendor selection process. Thus, the decision was made to proceed with Phase II testing.

The vendors selected for each purchase request in this phase of testing are considered to be the correct answers against which to judge the vendor selection responses of follow-on testing.

Phase II

Synopsis. Testing in the second phase is designed to answer the question, 'Is the buyer able to select the correct vendor using the prototype system?' To this end, DESC selected eight buyers whose daily assignments include procurement of MilSpec 55182 items. However, after the testing was complete, the researcher discovered not all eight buyers selected were familiar with the procurement of MilSpec 55182 items. More details on this deviation can be found later in the chapter. These buyers processed the same purchase requests that the panel examined in Phase I. Their selections were compared to that of the panels'. The details of the test follow.

Testing Phase II testing commenced on 23 July 1991. On that morning, the buyers received an hour briefing concerning this phase of testing. The briefing covered the overriding VASPP concept (provided by Mr. Bill Gates) and a presentation of the prototype software (provided by the researcher). The researcher also described the testing procedure that would begin that afternoon.

The researcher loaded the prototype software onto the computer at each buyer's desk. The prototype was started tested with a trial entry and returned to the welcome screen. Placed on each desk was a set of fifteen purchase requests. Four of the buyers, selected arbitrarily, were provided the purchase requests a set of vendor price lists, and DESC 701 forms, used for manual processing.

The other four buyers were provided a set of purchase requests (without the price lists or Form 701) for processing on the prototype system. Each of the four members of a group was given the same purchase requests to process. All purchase requests being processed by a group of four were different from those in the other group.

Phase II - Part 1.

Manual Processing. To perform the vendor selection process, the buyers repeat the process used in Phase I for manual processing. Once the part number of the item requested is located on the purchase request, the buyer is able to consult the vendor price list. If the part number appears in the price list, the appropriate information is transcribed onto DESC Form 701.

This process is completed for all known vendors. Having identified the vendors listing the product for sale, the buyer computed the extended price (price of each item, times the quantity required).

The buyer, now knowing which vendor(s) can supply the parts at the lowest cost, must decide which vendor is best qualified to receive the contract award. Before this decision can be made, the buyer must consult several historical files maintained at DESC regarding each vendor. Once the file review is completed, the buyer possesses the information provided by the vendor, the data stored on file, and knowledge gained through experience. The buyer can now make the final award decision.

The selected vendor, the quantity ordered, and the extended price of the award were then recorded on the Buyer Selection Form attached to the purchase request (Figure 5-4). In addition, the buyers recorded the purchase request processing start time and completion time. If an interruption occurred during the analysis, the buyer marked the appropriate block on the attached form. Each buyer (using the manual system) processed all fifteen purchase requests in this manner.

Automated Processing. The four buyers using the automated system received a set of fifteen purchase requests. Each set contained identical purchase requests. These purchase requests were unique from those provided to the buyer performing the manual process.

BUYER SELECTION

PR: YPE91191000882 Buyer No: 4 M

Timing Criteria

Time Started: 2:04 Delayed ☐

Time Finished: 2:08 4 m.r.

Award Information

Vendor Selected:	Quantity:	Extended Price:
65313	149	1.10

(Place Comments On Reverse)

FIGURE 5-4 -- BUYER SELECTION FORM

Using the prototype, the buyer was prompted to enter the NSN of the item requested. Once entered, the prototype verified the validity of the NSN and asked the buyer to input the quantity required. These two data elements can be found on the front page of the purchase request (Figure 5-5). Next, the prototype asked the buyer for the required delivery date; found on the last page of the purchase request (called the trailer) (Figure 5-6). Finally, the user indicated whether the award was to be given to a disadvantaged business (Set-A-Side).

Having entered all required information, the system interrogated its data files and displayed the unit cost screen. This screen informs the buyer which vendors supply the item required as well as the minimum quantity of the product (and the price at which the vendor offered it for sale) that meets or exceeded the quantity requested on the purchase request. The buyer was now able to switch to any of the user screens, examining the data presented, to arrive at an award decision. As with the manual process; the buyer recorded the vendor

CONTINUATION SHEET	REFERENCE NO. OF DOCUMENT BEING CONTINUED	PAGE 1 OF PAGES

NAME OF OFFEROR OR CONTRACTOR

ITEM NO	SUPPLIES/SERVICES	QUANTITY	UNIT	UNIT PRICE	AMOUNT
	PR YPE91191000882				
	NSN 5905-00-784-3209 NSN				
	ITEM DESCRIPTION:				
	ITEM NAME RESISTOR,FIXED,FILM I/A/W SPEC NR MILR55182F QP DASH 0108 BASIC DTD 90 JAN 26 AMEND NR 01 DTD 90 AUG 24 TYPE NUMBER RNR55C4020FS I/A/W SPEC NR MILR55182F/1M QP DASH 0108 BASIC DTD 85 DEC 06 AMEND NR 02 DTD 90 JAN 26 TYPE NUMBER RNR55C4020FS				
	PRLI 000100	86	EA		
	QTY VARIANCE: PLUS % MINUS % INSP/ACCEPT POINT:				
	PREP FOR DELIVERY:				
	PACKAGING DATA - MIL-STD-2073 (1A 16 JUL 84) (2B 14 MAR 86) QUP = 001: PRES MTHD = JK: CLNG/DRY = 1: PRESV MAT = 00: WRAP MAT = XX: CUSH/DUNN MAT = XX: CUSH/DUNN THKNESS = X: UNIT CONT = XX: LEVEL PRESV = A: INTRMDTE CONT = D3: INTRMDTE CONT QTY = XXX: PACK = U: PACKING LEVEL = C: MARKING SHALL BE IN ACCORDANCE WITH MIL-STD-129L SPECIAL MAR KING CODE: 39 - SENSITIVE ELECTRONIC DEVICE MIL-STD-129				
	DOD LOGMARS BAR CODE MARKING REQUIRED IAW MIL-STD-129L, APPENDIX H, DATED 15 OCT 90 AND MIL-STD-1189B, DATED 10 AUG 89.				
	DELIVER FOB: BY:				
	CONTINUED ON NEXT PAGE				

NSN 7540-01-152-8067 36-109 STANDARD FORM 36

PREVIOUS EDITION USABLE * GPO 1983 O - 381-526 (90501) Prescribed by GSA FAR (48 CFR) 53.

Quantity

FIGURE 5-5 -- PURCHASE REQUEST (IDENTIFYING NSN AND QUANTITY)

```
F-27-P    DESC                                PURCHASE REQUEST TRAILER LISTING                    AS OF 10 JUL 91   PAGE   0002

PURCHASE NUMBER  YPE91191000882       STOCK/PART NR   5905-00-784-3209     PGC
                                                        BASE PRICE
DLA90089P7734       7K545 A  N   B EA          1.02000  89044       100 00000   D  Y  1    AVNET INC
                                                                                            HAMILTON/AVNET ELECTRONICS DIV
                                                                                            7760 WASHINGTON VILLAGE DR
                                                                                            DAYTON OH   45459-4017
                                                                                            E K N N
                                                                                            5065
                                                                                            513-439-6750

                                        SUGGESTED SOURCES AND REFERENCES                    MORE REFERENCES AVAILABLE IN DD 635 NO
81349 2                       RNR55C4020FS                                                  NO ADDRESS AVAILABLE

BASIC NUMBER   MILR55182F        QP DASH 0108  QPL /RDC  Q
BASIC NUMBER   MILR55182F/1M     QP DASH 0108  QPL /RDC  Q

 PRLI   LOC  DESTINATION  RDD   QUANTITY  FCC  SEA  PROJ  IPD  N/E
000100  SRE     SW0400   92093        86 EA         DSB
000200  SUE     SW3400   92093        63 EA         DSB
                PR TOTAL QUANTITY          149          TOTAL PRICE              220.52
END ITEM APPLICATION                                                  SMCC F
```

RDD

WEAPONS SYSTEM

FIGURE 5-6 -- FORM 36 TRAILER (IDENTIFYING RDD)

selected, the quantity procured, the extended price of the contract, and the time required to reach an award decision, on the Buyer Selection Form attached to each purchase request.

Throughout this process, the researcher remained in the area of the buyers to answer any questions they may have had.

Phase II - Part 2. As each buyer completed their respective set of fifteen purchase requests, the researcher provided them with a second set of fifteen requests. This new set was identical with those being processed by the other group. The buyers using the manual system for their first set, now used the prototype system to process the new set. Those buyers that used the prototype system, now used the manual system to process the new set. As in the first round, the buyers recorded the vendor selected, the quantity procured, extended price of the award, and the processing start and stop times.

After each buyer completed processing the second set of purchase requests, they received a questionnaire. The questionnaire tried to capture the buyers impression of the prototype, as tested, and the course that future developments should take. The buyers were instructed to take their time in filling out the questionnaire and return it the following day. The comments provided by the buyers can be found in Appendix G.

Results. The data obtained from this phase of testing are consolidated and presented in the following tables and graphs. The data are divided into two components, the first being the results obtained from the current manual system for processing purchase requests. The second contains the data obtained from processing the purchase requests using the prototype.

The first column in each table identifies the purchase request that was processed. Following that is the cage code (Vendor Identification Code) of the vendor selected, the quantity ordered, the price paid, and the number of people who made this selection. The final column, Type of Error, is discussed in detail later in this chapter. It should be noted,

the first row of each section (the row containing the purchase request number) is the correct response, as determined by the panel.

In Table 5-3, the # indicates where the buyers misinterpreted the vendor price list. The vendor selected does not offer the exact part as requested on the purchase request. Thus, these entries are counted as errors. Table 5-5 is a summary of the errors identified in this portion of the testing.

Table 5-4 lists the responses obtained from the automated portion of the testing. Its format is the same as that for Table 5-3, Phase II Manual Error Results.

The Type of Error symbols found in Table 5-4 consist of the following:

1) * - this entry matches the panel selection.

2) & - this error is a result of transcribing the data incorrectly. It counts as a reasonable choice.

3) # - indicates the purchase requests effected by the incorrect vendor information entered in the pricing data base.

Table 5-6 is a summary of the errors identified in this portion of the testing.

Error Rate. In Tables 5-3 and 5-4, the final column indicates the type of error made. The panel reviewed each selection that did not match exactly in cage, quantity, and price. They made the determination of whether the selection annotated was a reasonable alternate selection or if it was an error. If the panel felt an error had been made, they tried to decide what led to the incorrect response.

Figure 5-7 depicts the relationship of reasonable responses to the error responses. In the manual phase, the buyers matched the panel exactly fifty percent of the time. Twenty-eight point three percent of the responses were reasonable alternate choices. The manual method of making the vendor selection resulted in a twenty-one point seven percent error rate.

TABLE 5-3

PHASE II MANUAL ERROR RESULTS

PHASE II - MANUAL ERROR RESULTS					
Purchase Request Number	Cage	Qnty	Extended Price	n	Type Of Error
YPE91191000882	6S313	149	$163.90	5	*
		500	$500.00	1	Reasonable
	56856	150	$168.00	1	Error
	0BTU6	149	$178.80	1	Error
YPE91188000883	7K545	1000	$230.00	1	*
		528	$126.72	7	Reasonable
YPE91195001056	6S313	100	$110.00	5	*
		100	$153.00	1	Math Error
	7K515	100	$115.00	1	Error
	0BTU6	100	$120.00	1	Error
YPE91195001053	7K545	1000	$181.00	3	*
		809	$155.33	5	Reasonable
YPE91188000892	56856	100	$135.00	1	*
		66	$118.80	1	Reasonable
	6S313	100	$134.00	4	Error (#)
	0BTU6	100	$151.00	1	Error
		100	$389.00	1	Math Error
YPE91191000875	6S313	200	$90.00	1	*
		200	$45.00	1	Math Error
	7K545	500	$119.00	5	Reasonable
		500	$110.50	1	Math Error
YPE91188000885	6S313	100	$110.00	5	*
	56856	64	$102.40	1	Reasonable
		100	$115.00	1	Error
	7K545	100	$115.00	1	Error
YPE91188000881	7K545	500	$197.00	5	*
		165	$65.01	1	Math Error
		500	$144.00	1	Math Error
	6S313	165	$198.00	1	Error
YPE91188000894	7K545	500	$159.00	6	*
		300	$95.40	1	Math Error
		500	$116.00	1	Math Error
YPE91188000893	6S313	100	$60.00	3	*
		100	$110.00	1	Math Error
	7K545	500	$120.00	3	Reasonable
	56856	100	$144.00	1	Error
YPE91151000115	7K545	4191	$783.72	7	*
		4191	$808.86	1	Math Error
YPE91188000914	6S313	100	$54.00	6	*
	7K545	500	$159.00	2	Reasonable
YPE91188000919	6S313	100	$45.00	5	*
	7K545	500	$118.00	3	Reasonable
YPE91188000877	7K545	500	$197.00	6	*
		331	$107.91	1	Math Error
		331	$130.41	1	Math Error
YPE91191000877	7K545	1000	$216.00	1	*
		1000	$260.00	1	Math Error
		642	$148.94	5	Reasonable
	6S313	642	$148.94	1	Transcribe

TABLE 5-4

PHASE II AUTOMATED ERROR RESULTS

PHASE II - AUTOMATED ERROR RESULTS					
Purchase Request Number	Cage	Qnty	Extended Price	n	Type Of Error
YPE91177000268	6S313	100	$53.00	5	*
	56856	31	$48.98	3	Reasonable
YPE91175000178	7K545	500	$120.00	6	*
	6S313	300	$135.00	2	Error
YPE91148000183	6S313	100	$160.00	7	*
		31	$160.00	1	Transcribe(&)
YPE91157000145	7K545	6500	$1,404.00	0	* (#)
		7000	$1,512.00	8	Reasonable
YPE91146000673	6S313	100	$45.00	8	*
YPE91195001054	6S313	131	$144.10	7	*
	6S313	500	$500.00	1	Reasonable
YPE91191000881	7K545	500	$96.50	2	*
		1000	$187.00	1	Error
	6S313	500	$90.00	4	Reasonable
		268	$90.00	1	Transcribe(&)
YPE91191000876	7K545	1000	$187.00	4	*
	6S313	541	$156.89	4	Reasonable
YPE91191000874	6S313	100	$110.00	3	*
		11	$47.50	1	Error
	56856	25	$47.50	3	Reasonable
		12	$46.20	1	Error
YPE91188000890	7K545	500	$144.00	7	*
		1000	$267.00	1	Error
YPE91188000887	7K545	1000	$187.00	8	*
YPE91151000352	6S313	100	$53.00	7	*
	56856	29	$45.82	1	Reasonable
YPE91188000880	6S313	200	$120.00	4	*
	7K545	200	$120.00	1	Transcribe(&)
		500	$130.00	2	Reasonable
		1000	$250.00	1	Error
YPE91188000879	7K545	2567	$654.59	0	*
		3000	$765.00	8	Reasonable
YPE91188000878	56856	58	$92.80	3	* (#)
	6S313	100	$110.00	5	Reasonable

TABLE 5-5

PHASE II MANUAL SUMMARY OF RESPONSES

PHASE II MANUAL SUMMARY OF RESPONSES	
Response Categories	Number Observed
Matched Panel	60
Reasonable	33
Transcription	1
Math Errors	13
Other Errors	13

TABLE 5-6

PHASE II AUTOMATED SUMMARY OF RESPONSES

PHASE II AUTOMATED SUMMARY OF RESPONSES	
Response Categories	Number Observed
Matched Panel	71
Reasonable	39
Transcription	3
Errors	7

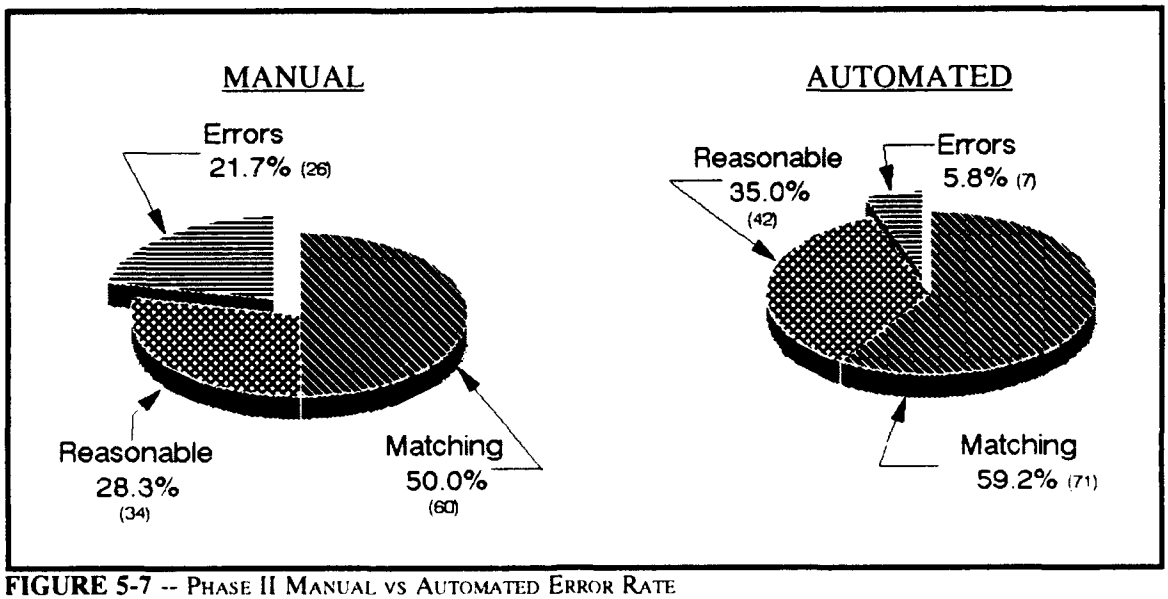


FIGURE 5-7 -- PHASE II MANUAL VS AUTOMATED ERROR RATE

In the automated phase, the buyers matched the panel fifty-nine point two percent of the time. Thirty-five percent of the responses were reasonable alternate choices. This method of making the vendor selection resulted in a five point eight percent error rate.

Using the prototype, there was a nine point two percent increase in buyer selections that matched the panel exactly. Conversely, the prototype offered a fifteen point nine percent reduction in errors.

The most common source of error appears to be math related. This could result from either misreading the vendor price lists or from failing to perform the extended price calculations correctly. As the prototype performs all calculations, math errors were only identified as occurring in the manual process.

A second problem identified was transcribing the data onto the response form. If two of the three categories matched the panel selection, it was sometimes possible to deduce the remaining data element was copied wrong. For example, if using the prototype system, the buyer identified the correct cage code and quantity but the price was incorrect, it could be surmised the price was copied incorrectly from the monitor.

Analysis. Using Statistix 's AOV procedure, an ANOVA was performed on the error data set for Phase II (Figure 5-8). The data examined are found in Table 5-7. The null hypothesis being examined is that the mean of the differences is zero (26:882). With a computed p-value of 0.0002, the null hypothesis can be rejected at the ninety-nine percent confidence level. In other words, there is strong statistical evidence to suggest there is a difference in the number of errors produced by the buyers using the two methods of vendor selection.

TABLE 5-7

PHASE II ERROR DATA

Phase II Errors									
	1 = Man	Buyer							
PR	2 = Auto	1	2	3	4	5	6	7	8
1	1			1				1	
2	1	1		1					
3	1		1	1	1		1	1	1
4	1			1				1	
5	1					1	1		
6	1	1		1				1	
7	1					1		1	
8	1		1	1		1			
9	1								1
10	1			1					
11	1								
12	1								
13	1								
14	1	1		1					
15	1								
16	2							1	
17	2								
18	2	1		1					
19	2							1	
20	2								
21	2							1	
22	2								
23	2								
24	2								
25	2								
26	2								
27	2								
28	2								
29	2								
30	2	1		1					

```
STATISTIX 3.5                                                    23 AUG 91, 17:35
ID: PHASE II PROCESSING ERRORS

ONE WAY AOV FOR ERROR = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------  --------  ------  ------
BETWEEN    1   1.667      1.667      14.42  0.0002
WITHIN   238   27.52      1.156E-01
TOTAL    239   29.18

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   37.51     1    0.0000

COCHRAN'S Q                  0.7604
LARGEST VAR / SMALLEST VAR   3.174

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  1.293E-02
EFFECTIVE CELL SIZE                       120.0

                        SAMPLE     GROUP
  SYSTEM      MEAN       SIZE     STD DEV
--------  ----------  ------  ----------
    1       2.250E-01   120     4.193E-01
    2       5.833E-02   120     2.354E-01

 TOTAL      1.417E-01   240     3.400E-01

CASES INCLUDED 240    MISSING CASES 0
```

FIGURE 5-8 -- PHASE II ERROR RATE ANOVA TEST

Figure 5-9 shows the number of errors resulting from using the manual system compared to the errors that resulted from using the prototype. The horizontal axis shows the number of errors made on a given purchase request. The vertical axis shows the number of purchase requests that contained the X-axis quantity of errors. From this graph, it can be seen the highest error rate for any purchase request processed using the prototype is two. (Two of the eight buyers recorded incorrect information.) This contrasts to the manual system. There was one outlier purchase request with seven errors, and two with three errors. The graph also shows using the prototype, ten purchase requests were processed without errors by all buyers, while the manual system could only claim four error-free requests.

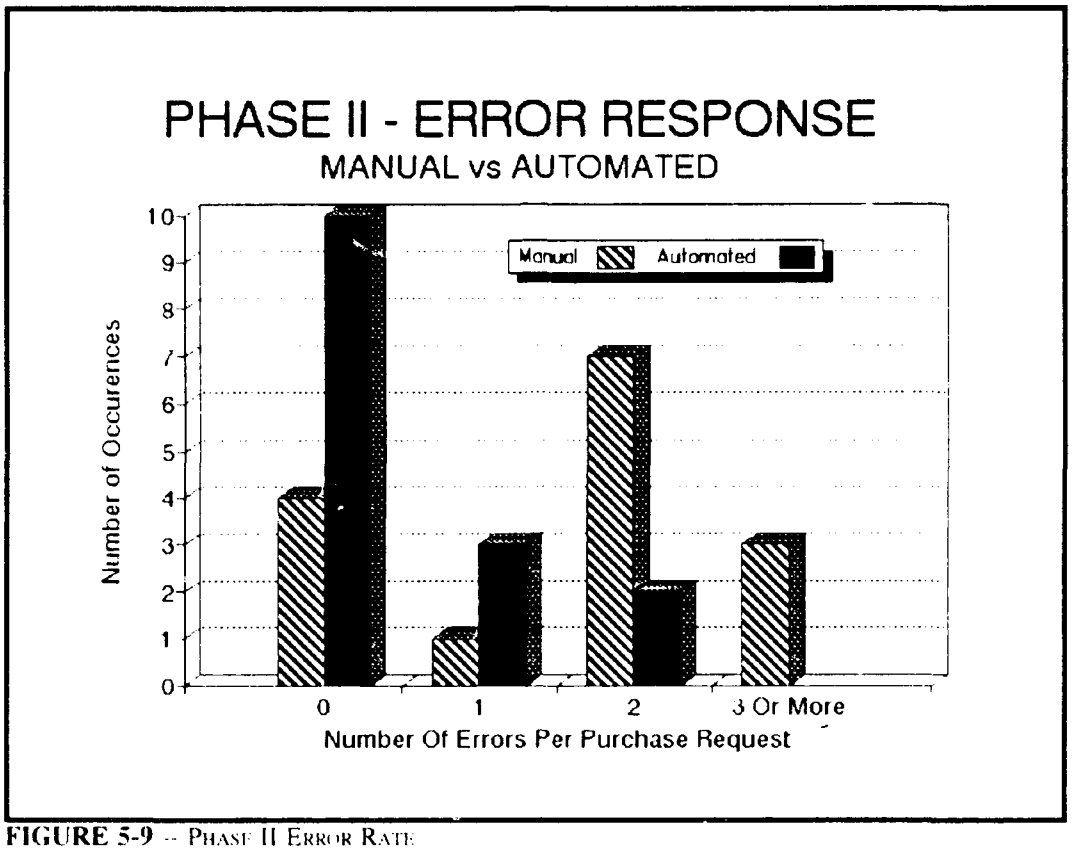


FIGURE 5-9 -- PHASE II ERROR RATE

Processing Time. The buyers documented the start and stop times for each purchase request processed. If a delay occurred during processing, the response form was annotated by checking the delayed box. Those requests marked delayed were not computed in the total processing time. The total time taken to process all non-delayed purchase requests was divided by the number of non-delayed requests, to attain the average time required to process each request.

Analysis. Figure 5-10 shows the descriptive statistics (derived from Table 5-8) for the processing time reported by the buyers to process the manual versus the automated portions of Phase II testing. PHASE2M identifies the statistics for the time required to process the request using the manual process. The mean processing time is 4.4 minutes. There is ninety-five percent confidence that the manual processing time will lie between 3.9 and 4.9 minutes, with a standard deviation of 2.3 minutes.

The statistics for PHASE2A was generated by the same buyers, but this time using the prototype. The results of this phase of testing are as follows. The mean processing time is 2.4 minutes. There is ninety-five percent confidence that the manual processing time will lie between 2.2 and 2.7, with a standard deviation of 1.2 minutes.

TABLE 5-8

PHASE II PROCESSING TIME

Phase II Processing Time									
	1=Man	Buyer							
PR	2=Auto	1	2	3	4	5	6	7	8
1	1	5		3		3	5	3	3
2	1	8	5	10	4		5		4
3	1	4	5	4	4	2	3	5	3
4	1	5	5	5	2	4	4	5	8
5	1	2	10	5	3	2	4	9	2
6	1	7	10	3	2	2		5	3
7	1	3	5	2	3	2	2	5	2
8	1	4	10	8		3	3		
9	1			8		2	5		4
10	1		5		3	2	3		2
11	1	[illegible]	10	11	5	4	5	2	4
12	1	5	5		2	2	2		2
13	1	3	10			3	5	5	2
14	1	7	5	10	4	2		2	5
15	1	5	5	5	4	2	5		3
16	2		1	5	2	1	3		2
17	2	1	1	2		2	3	5	2
18	2	2	2	3	3	2	4	5	3
19	2	2	5	4	2	2	2	4	1
20	2	2	5			2	3	5	3
21	2	3		5			3		3
22	2	1	1	4	1	1	2		1
23	2	1	1	2	2	2	2		2
24	2	2	1	4	2	3	3	5	2
25	2	2	1	2	1	2			2
26	2	2	1	4	3	2	3	5	
27	2	2	1	3	1	2	4		1
28	2	2	2	3	2	2	3	5	2
29	2	2	1	2	3	2	3	5	1
30	2	1	1	3	1	2	3		2

STATISTIX 3.5 30 AUG 91, 21:00

ID: PHASE II (MANUAL) VS PHASE II (AUTOMATED) PROCESSING TIMES

DESCRIPTIVE STATISTICS

	PHASE2M	PHASE2A
CASES	99	104
LOWER 95.0% C.I.	3.940	2.186
MEAN	4.404	2.423
UPPER 95.0% C.I.	4.868	2.660
S.D.	2.325	1.220
S.E. (MEAN)	2.337E-01	1.197E-01
C.V.	52.80	50.36
MINIMUM	2.000	1.000
MEDIAN	4.000	2.000
MAXIMUM	11.00	5.000

FIGURE 5-10 -- PHASE II TIMING DESCRIPTIVE STATISTICS

The typical processing time was reduced by 2.0 minutes using the prototype. This represents an approximate forty-five percent reduction in processing time. Figure 5-11 illustrates the processing times of the two systems.

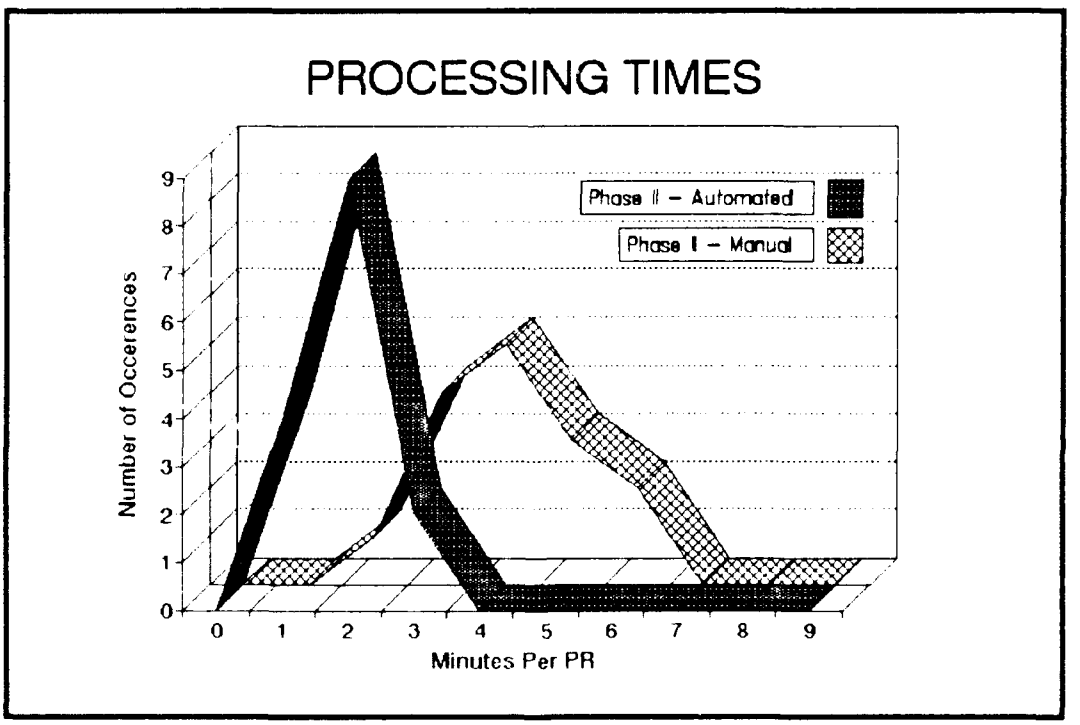


FIGURE 5-11 --PHASE II PROCESSING TIMES

An ANOVA test was performed comparing the processing time of the manual process against the time required to use the prototype (Figure 5-12). The data used are found in Table 5-8.

The null hypothesis being examined is that the mean of the differences is zero (4:206). With a computed p-value of 0.000, the null hypothesis can be rejected. In other words, there is very strong statistical evidence to suggest there is a significant difference in the processing times of the two systems.

```
STATISTIX 3.5                                                    30 AUG 91, 21:23
ID: PHASE II PROCESSING TIMES

ONE WAY AOV FOR TIME = SYSTEM

SOURCE     DF      SS        MS        F       P
------  ----  --------  --------  ------  ------
BETWEEN     1   199.0     199.0    58.55  0.0000
WITHIN    201   683.2     3.399
TOTAL     202   882.3

                      CHI SQ    DF      P
BARTLETT'S TEST OF   ------  ------  ------
   EQUAL VARIANCES    39.33     1    0.0000

COCHRAN'S Q                  0.7840
LARGEST VAR / SMALLEST VAR    3.631

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  1.929
EFFECTIVE CELL SIZE                       101.4

                        SAMPLE    GROUP
 SYSTEM      MEAN        SIZE    STD DEV
--------  ----------  ------  ----------
    1       4.404        99     2.325
    2       2.423       104     1.220

 TOTAL      3.389       203     1.844

CASES INCLUDED 203    MISSING CASES 37
```

FIGURE 5-12 -- PHASE II PROCESSING TIME ANOVA TEST

Phase III

Synopsis. The third and final phase of testing was conducted on July 25th. This test sought to answer the question 'Is the system designed such that, a person unfamiliar with the items being procured, is able to make a valid vendor selection decision?'

Testing. As before, eight buyers, selected by DESC, received an orientation briefing in the morning. The content of the briefing was the same as that presented to the buyers in the previous phase. The prototype software was loaded onto the buyers computer systems during the lunch break. Placed on each buyers desk, was a set of fifteen purchase requests. Each set was identical, and was the same as used in the automated portion of Phase II testing.

The differences in this phase of testing versus the automated portion of Phase II lie in two areas. One, the buyers chosen to participate were not familiar with the vendors and products of the MilSpec 55182 items. Two, the buyers only processed fifteen purchase requests, using the prototype to process all requests.

As before, The researcher was available to assist the buyers during the testing portion.

Results. The data obtained from this phase of testing are found in Table 5-9 and the types of errors encountered are summarized in Table 5-10.

Error Rate. The panel reviewed each selection that did not match exactly in cage, quantity, and price. They determined if the selection annotated was a reasonable alternate selection or if it was an error. If the panel felt an error existed, they tried to decide what led to the incorrect response. A '&' indicates the response was reasonable. A '!' indicates the response was unreasonable. A '*' indicates a perfect match to the panel selection.

TABLE 5-9

PHASE III RESULTS

PHASE III RESULTS					
Purchase Request Number	Cage	Qnty	Extended Price	n	Type Of Error
YPE91177000268	6S313	100	$53.00	5	*
	56856	31	$48.98	3	Reasonable
YPE91175000178	7K545	500	$120.00	4	*
	6S313	300	$135.00	3	Error
		30	$135.00	1	Transcribe(!)
YPE91148000183	6S313	100	$160.00	8	*
YPE91157000145	7K545	6500	$1404.00	8	*
YPE91146000673	6S313	100	$45.00	7	*
	7K545	500	$118.00	1	Reasonable
YPE91195001054	6S313	131	$144.10	7	*
	56856	131	$150.65	1	Wrong Vendor
YPE91191000881	7K545	500	$96.50	2	*
	6S313	300	$90.00	6	Reasonable
YPE91191000876	7K545	1000	$187.00	3	*
	6S313	541	$156.89	5	Reasonable
YPE91191000874	6S313	100	$110.00	2	*
	56856	25	$47.50	5	Reasonable
		11	$47.50	1	Transcribe(&)
YPE91188000890	7K545	500	$144.00	6	*
	6S313	332	$146.08	2	Wrong Vendor
YPE91188000887	7K545	1000	$187.00	7	*
	6S313	729	$211.41	1	Wrong Vendor
YPE91151000352	6S313	100	$53.00	4	*
	56856	29	$45.82	3	Reasonable
		19	$45.60	1	Transcribe(!)
YPE91188000880	6S313	200	$120.00	7	*
	7K545	500	$130.00	1	Reasonable
YPE91188000879	7K545	2567	$654.59	8	*
YPE91188000878	56856	58	$92.80	4	*
	6S313	100	$110.00	4	Reasonable

Analysis. An ANOVA Test was performed on the data obtained in this phase and the data from the automated portion of Phase II (Figure 5-13). The data examined is found in Table 5-11. As before, the null hypothesis is that the mean of the differences is zero. The

TABLE 5-10

PHASE III SUMMARY OF RESPONSES

PHASE III SUMMARY OF RESPONSES	
Response Categories	Number Observed
Matched Panel	82
Reasonable	29
Transcription Errors	2
Errors	7

TABLE 5-11

PHASE II & III ERROR DATA

PHASE II vs PHASE III

Errors Per Purchase Request

PR	Buyer 1		Buyer 2		Buyer 3		Buyer 4		Buyer 5		Buyer 6		Buyer 7		Buyer 8	
	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3
1													1			
2																
3	1				1											
4								1				1	1			
5								1								
6													1			
7																
8																
9												1				
10																
11										1						
12																
13																
14																
15	1	1		1	1			1				1				

```
STATISTIX 3.5                                                    30 AUG 91, 22:47
ID: PHASE II/III PROCESSING ERRORS

ONE WAY AOV FOR ERROR = PHASE

SOURCE   DF      SS          MS          F       P
------  ----   --------   --------   ------  ------
BETWEEN    1   1.667E-02  1.667E-02    0.27  0.6066
WITHIN   238   14.92      6.268E-02
TOTAL    239   14.93

                     CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES    1.61     1    0.2044

COCHRAN'S Q                  0.5581
LARGEST VAR / SMALLEST VAR   1.263

COMPONENT OF VARIANCE FOR BETWEEN GROUPS -3.834E-04
EFFECTIVE CELL SIZE                       120.0

                        SAMPLE     GROUP
  PHASE        MEAN      SIZE     STD DEV
```

FIGURE 5-13 -- PHASE II & III ERROR ANOVA TEST

computed p-value is 0.6066, therefore the null cannot be rejected. There is no significant difference in the error data obtained from those people not familiar with the MilSpec 55122 items when compared to those who are familiar with the items.

Figure 5-14 compares the error rate of the automated portion of Phase II against the results of Phase III. The pie graph labeled Phase III is derived from Table 5-9.

Interestingly, the buyers without prior experience in this area showed a nine point one percent increase in buyer selections that matched the panel exactly. The Phase III buyers also experienced an increase in errors, which, as shown by the above ANOVA, was not statistically different.

Processing Time. As before, only the non-delayed times were used in calculations. The data set comparing the automated times from Phase II and those from Phase III are presented in Table 5-12.

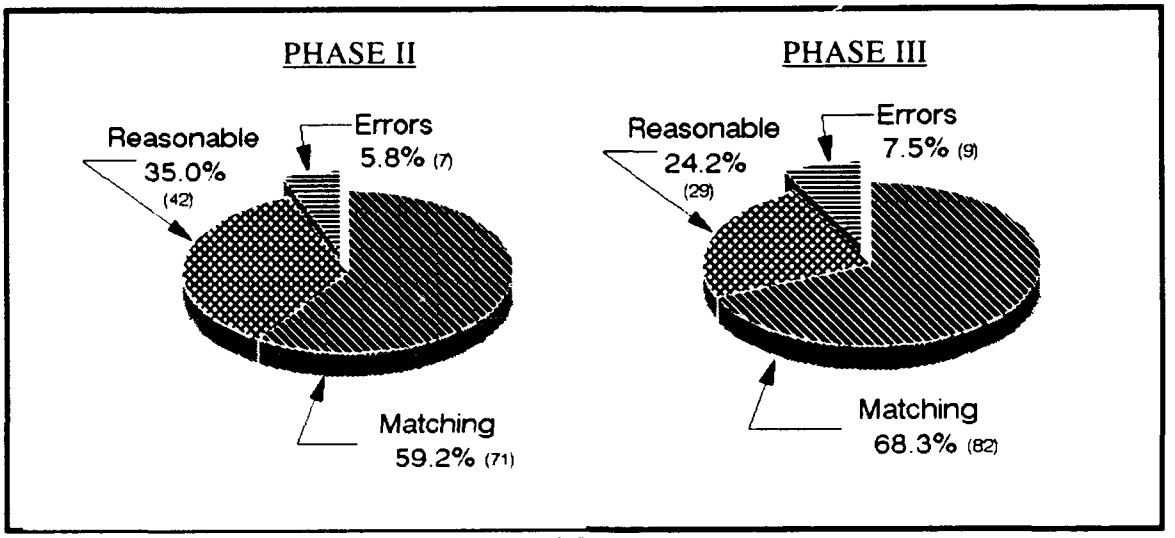


FIGURE 5-14 -- PHASE II AUTOMATED VS PHASE III ERROR RATES

TABLE 5-12

PHASE II/III PROCESSING TIMES

PHASE II vs PHASE III

Processing Time

PR	Buyer 1		Buyer 2		Buyer 3		Buyer 4		Buyer 5		Buyer 6		Buyer 7		Buyer 8	
	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3	Ph 2	Ph 3
1		3	1		5	5	2	1	1	3	3	4			2	2
2	1	3	1	4	2	1		2	2	1	3	7	5	3	2	4
3	2		2	4	3	2	3	1	2	2	4	6	5	5	3	2
4	2	2	5	3	4	2	2	1	2	2	2		4	5	1	3
5	2	2	5			1		1	2	2	3	8	5	4	3	4
6	3			2	5	1		1		2	3	6		7	3	8
7	1		1	2	4	2	1	2	1	2	2	5		3	1	4
8	1	3	1	5	2	3	2	2	2	3	2	4		3	2	5
9	2		1	4	4	2	2	2	3	2	3		5	3	2	3
10	2		1	3	2	3	1	1	2			3		3	2	5
11	2	1	1	2	4	2	3	1	2	2	3		5			3
12	2	3	1	4	3	5	1	2	2	2	4	4			1	4
13	2	2	2	3	3	3	2	2	2	2	3		5	5	2	2
14	2	2	1	3	2	2	3	2	2	2	3	5	5	3	1	3
15	1	3	1	3	3	3	1	1	2	2	3			3	2	4

Analysis. Figure 5-15 shows the descriptive statistics for the time required for the buyers to process the Phase II automated versus the Phase III automated requests. Phase2 indicates the statistics of the time required to process the requests by users familiar with the MilSpec 55182 procurement process. Phase3 indicates the statistics of the time required by the users unfamiliar with this process. The results obtained from the people who were not familiar with the products and vendors used in the evaluation showed similar results compared to those who were.

The mean processing time for the buyers who were familiar with the procurement process is 2.4 minutes. There is ninety-five percent confidence that the processing time of those familiar with the products and vendors will lie between 2.2 and 2.7 minutes. The standard deviation is 1.2 minutes. The mean processing time for the buyers who were unfamiliar with the information is 3.0 minutes. There is ninety-five percent confidence that their processing time will lie between 2.7 and 3.3 minutes. The standard deviation is 1.5 minutes. Figure 5-16 illustrates the processing times of the two groups.

STATISTIX 3.5 23 AUG 91, 14:18

ID: PHASE II (AUTOMATED) VS PHASE III (AUTOMATED) PROCESSING TIMES

DESCRIPTIVE STATISTICS

	PHASE2	PHASE3
CASES	104	104
LOWER 95.0% C.I.	2.186	2.665
MEAN	2.423	2.962
UPPER 95.0% C.I.	2.660	3.258
S.D.	1.220	1.526
S.E. (MEAN)	1.197E-01	1.496E-01
C.V.	50.36	51.53
MINIMUM	1.000	1.000
MEDIAN	2.000	3.000
MAXIMUM	5.000	8.000

FIGURE 5-15 -- PHASE II VS PHASE III PROCESSING TIME DESCRIPTIVE STATISTICS

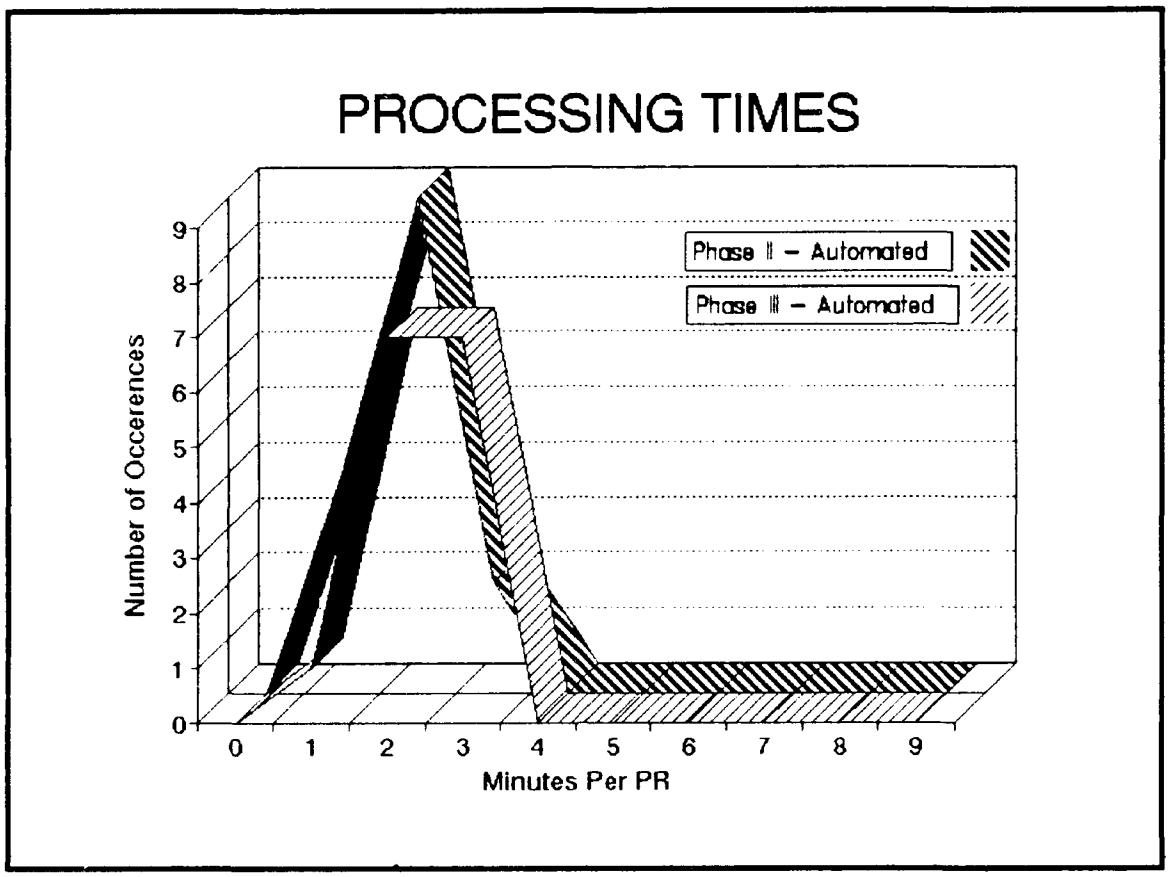


FIGURE 5-16 -- AVERAGE PHASE II/III PROCESSING TIME

An ANOVA test was performed comparing the processing times of the automated portion of Phase II against the time required by the buyers in Phase III (Figure 5-17). This time the computed p-value is 0.0054. There is evidence to indicate a difference exists in the processing times of the two groups.

User Comments

The comments provided by the users were very encouraging. The major responses to each question are summarized below. The reader is invited to refer to Appendix G for a complete listing of all comments provided.

Problems incurred while using the system were few. The most significant rests in the keys that are active to the user at the Award screen. The system was designed to enable

```
STATISTIX 3.5                                                    23 AUG 91, 14:25
ID: PHASE II (AUTOMATED) VS PHASE III (AUTOMATED) PROCESSING TIMES

ONE WAY AOV FOR: PHASE2 PHASE3

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------    ------  ------
BETWEEN    1   15.08      15.08       7.90  0.0054
WITHIN   206   393.2      1.909
TOTAL    207   408.3

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES    5.08     1    0.0242

COCHRAN'S Q                  0.6099
LARGEST VAR / SMALLEST VAR   1.564

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  1.266E-01
EFFECTIVE CELL SIZE                       104.0

                       SAMPLE     GROUP
VARIABLE      MEAN      SIZE     STD DEV
--------  ----------  ------  ----------
PHASE2      2.423      104      1.220
PHASE3      2.962      104      1.526

 TOTAL      2.692      208      1.382

CASES INCLUDED 208    MISSING CASES 32
```

FIGURE 5-17 -- PHASE II/III TIME COMPARISON ANOVA TEST

the user to press virtually any key when completed with this screen and return to the System Information screen. It is apparently 'too easy' to exit the Award screen prematurely.

Irrelevant information provided by the system was the next area discussed. A majority of the buyers thought the required delivery date was over emphasized. It appears the RDD date given on the purchase request provides little influence in the award decision.

Several different comments were offered for additional information that the prototype could/should provide. The most common suggestions are: identifying the manufacture of the part being offered by the vendor; packing data; location and terms of item inspection; vendor points of contact; and the inclusion of the quantity of the last buy in the historical information section. (The quantity of the last procurement is tracked by the prototype. It was an oversight that it was not included on the user screens.)

Responses to system future enhancements paralleled those for additional information. One forward looking individual suggested the system be designed to accommodate the automatic printing of the DESC Form 800 after the award decision is made.

The last question allowed the user to provide comments on the usefulness of the system. The responses here range from cautious optimism to full endorsement of the system. It is quite evident the buyers view the prototype as a significant improvement over the current method of vendor selection.

Initial Conclusions

There is strong evidence implying the prototype can present the correct information to the buyer and the buyer can successfully use the prototype to make a responsive award decision. To arrive at this inference, prototype testing was accomplished in three phases. The first addressed whether the prototype presented the correct information. The second phase demonstrated the prototype produced quicker and more consistent results. The third phase examined its usability to buyers unfamiliar with the products and/or vendors.

The comments provided by the buyers regarding the utility of the prototype are very positive. There is commonality in their replies that leads one to believe the prototype significantly enhances the current vendor selection method.

The positive results thus far must be tempered as the composition of the Phase II test group was not as intended. This caveat is discussed at length in the next section.

Short Comings

After review of the data, two problems were identified in the testing process.

PR Testing.

Problem. A weakness was recognized involving the purchase requests. The fifteen requests processed manually were never processed using the prototype. The possibility exists that the improvements observed in the error rate and processing times of the prototype could be explained by the accumulated difference in complexity of the purchase requests in each group.

Correction. In an attempt to correct this deficiency, each panel member was asked to rank each purchase request according to its complexity. The panel members were provided with a copy of the thirty purchase requests, the vendor price lists, and the 701 forms they completed in Phase I of the testing. Using a five level scale, the buyers indicated their opinions regarding the complexity of the purchase requests. The form found in Figure 5-18 was used to record their responses. No purchase request was scored more difficult than 'Easy'. Table 5-13 displays the results of their efforts. The first fifteen purchase requests listed on the form were processed in Phase II using the manual method of vendor selection. The last fifteen purchase requests were processed using the prototype. The panel was not informed of this grouping.

Table 5-14 shows the average difficulty assigned to each purchase request. 'Method' defines the system used to process the purchase requests in Phase II of the testing. A '1' represents the manual process was used, and a '2' represents the prototype system was used. 'Purchase Request' tracks the thirty requests processed on either system. 'Panel Member 1' identifies the responses provided by one panel member regarding the degree of difficulty of each purchase request evaluated. A '1' corresponds to 'Very Easy', '2'

Purchase Request

Degree of Difficulty

Purchase Request	Very Easy	Easy	Average	Difficult	Very Difficult
YPE91191000882					
YPE91188000883					
YPE91195001056					
YPE91195001053					
YPE91188000892					
YPE91191000875					
YPE91188000885					
YPE91188000881					
YPE91188000894					
YPE91188000893					
YPE91151000115					
YPE91188000914					
YPE91188000919					
YPE91188000877					
YPE91191000877					
YPE91177000268					
YPE91175000178					
YPE91148000183					
YPE91157000145					
YPE91146000673					
YPE91195001054					
YPE91191000881					
YPE91191000876					
YPE91191000874					
YPE91188000890					
YPE91188000887					
YPE91151000352					
YPE91188000880					
YPE91188000879					
YPE91188000878					

FIGURE 5-18 -- PURCHASE REQUEST DEGREE OF DIFFICULTY FORM

TABLE 5-13

PURCHASE REQUEST DEGREE OF DIFFICULTY RESPONSES

Purchase Request Degree of Difficulty				
	Panel Member 1		Panel Member 2	
Purchase Request	Very Easy	Easy	Very Easy	Easy
YPE91191000882	X			X
YPE91188000883	X			X
YPE91195001056	X			X
YPE91195001053	X			X
YPE91188000892	X		X	
YPE91191000875		X		X
YPE91188000885	X			X
YPE91188000881	X			X
YPE91188000894	X		X	
YPE91188000893	X		X	
YPE91151000115	X		X	
YPE91188000914	X		X	
YPE91188000919	X		X	
YPE91188000877	X			X
YPE91191000877	X			X
YPE91177000268	X		X	
YPE91175000178	X		X	
YPE91148000183	X		X	
YPE91157000145	X		X	
YPE91146000673	X		X	
YPE91195001054	X			X
YPE91191000881	X			X
YPE91191000876	X			X
YPE91191000874	X			X
YPE91188000890	X			X
YPE91188000887	X			X
YPE91151000352	X		X	
YPE91188000880	X		X	
YPE91188000879	X		X	
YPE91188000878	X			X

TABLE 5-14

PURCHASE REQUEST AVERAGE DEGREE OF DIFFICULTY

Purchase Request Average Degree of Difficulty				
Method	Purchase Request	Panel Member 1	Panel Member 2	Average
1	1	1	2	1.5
1	2	1	2	1.5
1	3	1	2	1.5
1	4	1	2	1.5
1	5	1	1	1.0
1	6	2	2	2.0
1	7	1	2	1.5
1	8	1	2	1.5
1	9	1	1	1.0
1	10	1	1	1.0
1	11	1	1	1.0
1	12	1	1	1.0
1	13	1	1	1.0
1	14	1	2	1.5
1	15	1	2	1.5
2	16	1	1	1.0
2	17	1	1	1.0
2	18	1	1	1.0
2	19	1	1	1.0
2	20	1	1	1.0
2	21	1	2	1.5
2	22	1	2	1.5
2	23	1	2	1.5
2	24	1	2	1.5
2	25	1	2	1.5
2	26	1	2	1.5
2	27	1	1	1.0
2	28	1	1	1.0
2	29	1	1	1.0
2	30	1	2	1.5

corresponds to 'Easy', and so on to '5' which corresponds to 'Very Difficult'. 'Panel Member 2' identifies the responses provided by the other panel member. The final column, 'Average' represents the average degree of difficulty assigned to each purchase request. It was derived by combining the points assigned to each request and dividing the result by two.

Results. Figure 5-19 depicts the results of the ANOVA test performed on the table data. The p-value of the between samples errors is 0.3987. There is not significant statistical evidence to indicate the purchase requests processed in each group differ in complexity. The null hypothesis (the difficulty level of the two samples are equal) cannot be rejected above

```
STATISTIX 3.5                                                    31 AUG 91, 0:01
ID: PURCHASE REQUEST DEGREE OF DIFFICULTY

ONE WAY AOV FOR SCORE = METHOD

SOURCE    DF      SS         MS         F       P
------ ----  --------  --------  ------  ------
BETWEEN    1   1.500E-01  1.500E-01    0.72  0.3987
WITHIN    58   12.03      2.075E-01
TOTAL     59   12.18

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES    0.33     1    0.5629

COCHRAN'S Q                  0.5540
LARGEST VAR / SMALLEST VAR   1.242

COMPONENT OF VARIANCE FOR BETWEEN GROUPS -1.916E-03
EFFECTIVE CELL SIZE                       30.0

                        SAMPLE     GROUP
  METHOD      MEAN       SIZE     STD DEV
--------  ----------  ------  ----------
    1       1.333        30     4.795E-01
    2       1.233        30     4.302E-01

 TOTAL      1.283        60     4.555E-01

CASES INCLUDED 60    MISSING CASES 0
```

FIGURE 5-19 -- PURCHASE REQUEST DEGREE OF DIFFICULTY ANOVA TEST

the sixty percent confidence level. Therefore the difference between the manual and automated results cannot be explained by differences in the complexity of the purchase requests. The purchase requests can be considered equal for both the manual and automated approaches.

Buyers Selected (Phase II Impact).

Problem. For the Phase II testing, DESC was requested to provide eight buyers familiar with processing MilSpec 55182 price listed items. After the testing was completed, it was learned by the researcher that not all eight buyers were familiar with the items as requested. A buyer not being familiar with the current vendor selection process could explain some of the improvements observed in the error rate and processing times of the prototype. Due to the anonymity granted to the buyers, it was not possible to identify which buyer generated which set of data.

Correction.

Error Rate. The data obtained from this Phase of testing was re-evaluated, examining the errors made by each buyer. Figure 5-20 shows the errors made by each buyer for both portions of the test. It appears buyers one, three, and seven made significantly more errors than the other buyers. This theory was tested using the *Statistix* one-way AOV test. A computed p-value of 0.0015 confirms a significant difference exists in the error rates of the buyers (Figure 5-21).

To determine which buyers were significantly different, Tukey's comparison of means test was used. Tukey was selected because "It controls the experimentwise error rate yet still retains good power" (4:144). The results of this test show three buyer groups in which the means are not significantly different from one another at the 0.05 level (Figure

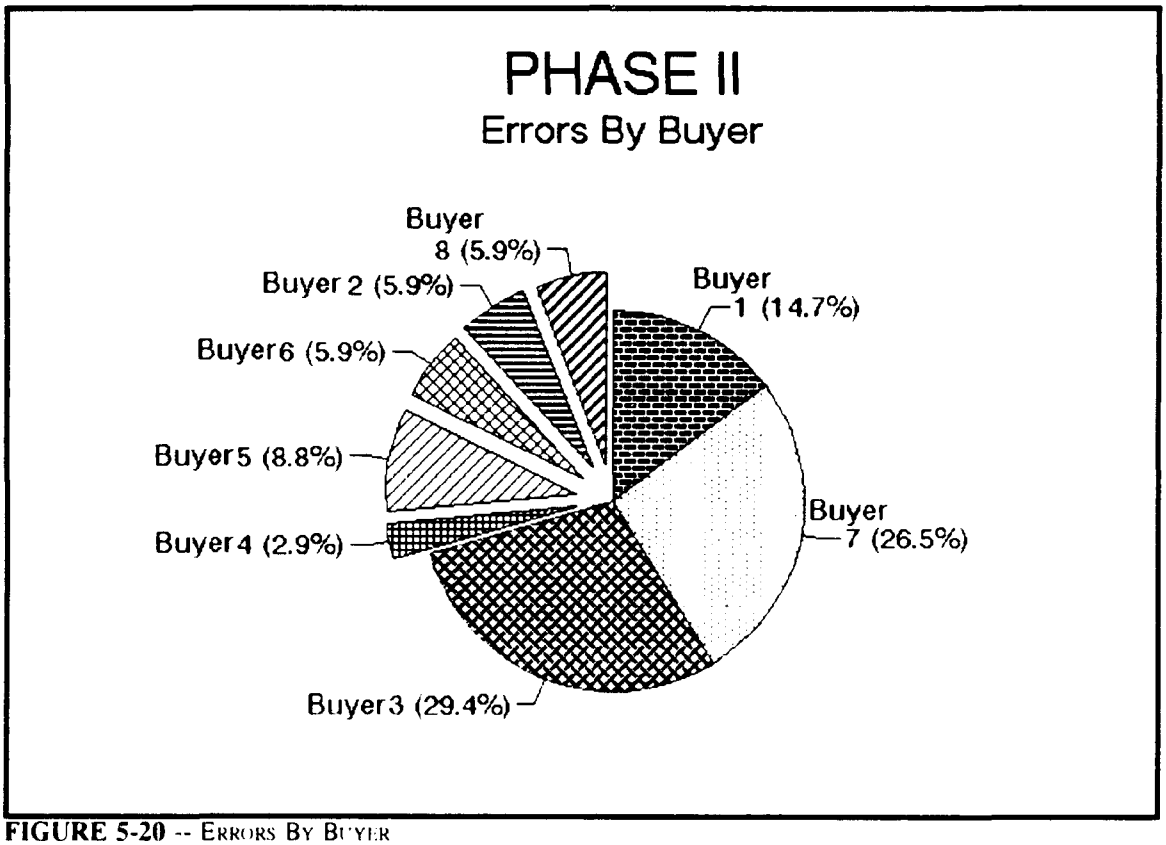


FIGURE 5-20 -- ERRORS BY BUYER

5-22). An ANOVA test was performed for each group, using the Phase II error data from Table 5-7. This series of tests is looking for a group of buyers which do not experience a significant reduction in error rates by using the prototype. After examination of the ANOVA results (Figures 5-23 through 5-25) it can be concluded, that each group, whether experienced or not, each group is experiencing a significant reduction (at the ninety-nine percent confidence level) in error rates by using the prototype.

Processing Time. The timing data from Phase II were re-evaluated, excluding buyers based on Tukey's test comparing the buyers mean time required to process the purchase requests (Figure 5-26). Based on mean processing time, there is no group of buyers that did not experience a significant improvement in the purchase request processing time when using the prototype. Statistically, there is over a ninety-nine percent confidence level that the processing time of the two methods are different (Figure 5-27 through 5-32).

```
STATISTIX 3.5                                                    23 AUG 91, 16:41
ID: PHASE II PROCESSING ERRORS

ONE WAY AOV FOR ERROR = BUYER

SOURCE    DF      SS         MS         F       P
------  ----  --------  --------  ------  ------
BETWEEN    7   2.783     3.976E-01   3.49  0.0015
WITHIN   232   26.40     1.138E-01
TOTAL    239   29.18

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   44.51     7    0.0000

COCHRAN'S Q                 0.2525
LARGEST VAR / SMALLEST VAR   6.897

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  9.461E-03
EFFECTIVE CELL SIZE                          30.0

                        SAMPLE     GROUP
  BUYER       MEAN      SIZE     STD DEV
--------  ----------  ------  ----------
    1       1.667E-01    30     3.790E-01
    2       6.667E-02    30     2.537E-01
    3       3.333E-01    30     4.795E-01
    4       3.333E-02    30     1.826E-01
    5       1.000E-01    30     3.051E-01
    6       6.667E-02    30     2.537E-01
    7       3.000E-01    30     4.661E-01
    8       6.667E-02    30     2.537E-01

 TOTAL      1.417E-01   240     3.373E-01

CASES INCLUDED 240    MISSING CASES 0
```

FIGURE 5-21 -- ANOVA Test Buyer Error Rate Comparison

```
STATISTIX 3.5                                                    31 AUG 91, 0:48
ID: PHASE II PROCESSING ERRORS (ANALYSIZED BY BUYER)

TUKEY (HSD) PAIRWISE COMPARISONS OF MEANS OF ERROR BY BUYER

                          HOMOGENEOUS
  BUYER        MEAN       GROUPS
--------  ----------  ----------
     3      3.333E-01   I
     7      3.000E-01   I I
     1      1.667E-01   I I I
     5      1.000E-01   I I I
     2      6.667E-02  .. I I
     6      6.667E-02  .. I I
     8      6.667E-02  .. I I
     4      3.333E-02  .... I

THERE ARE 3 GROUPS IN WHICH THE MEANS ARE
NOT SIGNIFICANTLY DIFFERENT FROM ONE ANOTHER.

CRITICAL Q VALUE                4.285   REJECTION LEVEL    0.050
CRITICAL VALUE FOR COMPARISON  2.6391E-01
STANDARD ERROR FOR COMPARISON  8.7099E-02
```

FIGURE 5-22 -- TUKEY COMPARISON OF BUYER ERROR RATES

Results. The Phase II data was re-examined looking for any indication that inexperienced buyers could have affected the test results. An analysis of the buyer's mean scores was used to categorize them in further testing. (Inexperienced buyers should show a statistically different mean error rate and processing from the experienced buyers.) An ANOVA test was performed on each group. No evidence was produced to indicate the differences in error rates and processing times recorded using the prototype, was due to the different experience levels of the buyers.

```
STATISTIX 3.5                                                    31 AUG 91, 1:11
ID: PHASE II PROCESSING ERRORS (BUYERS 2, 4, 6, AND 8 REMOVED)

ONE WAY AOV FOR ERROR = SYSTEM

SOURCE    DF      SS         MS          F       P
------   ----  --------   --------    ------  ------
BETWEEN    1   1.408      1.408        8.51  0.0042
WITHIN   118   19.52      1.654E-01
TOTAL    119   20.92

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES    8.43     1    0.0037

COCHRAN'S Q                  0.6832
LARGEST VAR / SMALLEST VAR   2.156

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  2.072E-02
EFFECTIVE CELL SIZE                        60.0

                         SAMPLE     GROUP
  SYSTEM      MEAN        SIZE     STD DEV
 --------  ----------   ------   ----------
    1       3.333E-01    60      4.754E-01
    2       1.167E-01    60      3.237E-01

 TOTAL      2.250E-01   120      4.067E-01

CASES INCLUDED 120    MISSING CASES 0
```

FIGURE 5-23 -- PHASE II SUBGROUP ONE ANOVA COMPARISON

```
STATISTIX 3.5                                                    31 AUG 91, 1:08
ID: PHASE II PROCESSING ERRORS (BUYERS 3 AND 4 REMOVED)

ONE WAY AOV FOR ERROR = SYSTEM

SOURCE    DF      SS         MS         F       P
------   ----  --------   --------   ------  ------
BETWEEN    1   9.389E-01  9.389E-01    8.74  0.0035
WITHIN   178   19.12      1.074E-01
TOTAL    179   20.06

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   26.19     1    0.0000

COCHRAN'S Q                  0.7531
LARGEST VAR / SMALLEST VAR   3.049

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  9.238E-03
EFFECTIVE CELL SIZE                        90.0

                        SAMPLE     GROUP
  SYSTEM      MEAN       SIZE     STD DEV
 --------  ----------  ------  ----------
    1       2.000E-01    90     4.022E-01
    2       5.556E-02    90     2.303E-01

 TOTAL      1.278E-01   180     3.278E-01

CASES INCLUDED 180    MISSING CASES 0
```

FIGURE 5-24 -- PHASE II SUBGROUP TWO ANOVA COMPARISON

```
STATISTIX 3.5                                                    31 AUG 91, 1:05
ID: PHASE II PROCESSING ERRORS (BUYERS 3 AND 7 REMOVED)

ONE WAY AOV FOR ERROR = SYSTEM

SOURCE    DF      SS         MS         F       P
------   ----   --------   --------   ------  ------
BETWEEN    1   6.722E-01  6.722E-01    9.15  0.0029
WITHIN   178   13.08      7.347E-02
TOTAL    179   13.75

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   59.82     1    0.0000

COCHRAN'S Q                  0.8505
LARGEST VAR / SMALLEST VAR   5.688

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  6.653E-03
EFFECTIVE CELL SIZE                        90.0

                         SAMPLE     GROUP
  SYSTEM      MEAN        SIZE     STD DEV
 ---- ---  ----------   ------  ----------
    1       1.444E-01    90     3.535E-01
    2       2.222E-02    90     1.482E-01

 TOTAL      8.333E-02   180     2.711E-01

CASES INCLUDED 180    MISSING CASES 0
```

FIGURE 5-25 -- PHASE II SUBGROUP THREE ANOVA COMPARISON

STATISTIX 3.5 31 AUG 91, 1:33

ID: Phase II Processing Times

TUKEY (HSD) PAIRWISE COMPARISONS OF MEANS OF TIME BY BUYER

BUYER	MEAN	7	3	2	6	1	8
7	4.706						
3	4.615	0.21					
2	4.222	1.15	1.05				
6	3.407	3.08	3.23	2.20			
1	3.222	3.52	3.72	2.70	0.50		
8	2.643	4.92*	5.31*	4.30*	2.08	1.58	
4	2.565	4.91*	5.26*	4.28	2.18	1.70	0.20
5	2.214	5.95*	6.47*	5.46*	3.25	2.74	1.18

BUYER	MEAN	4
4	2.565	
5	2.214	0.92

CRITICAL Q VALUE 4.285 REJECTION LEVEL 0.050

STANDARD ERRORS AND CRITICAL VALUES OF DIFFERENCES

VARY BETWEEN COMPARISONS BECAUSE OF UNEQUAL SAMPLE SIZES.

FIGURE 5-26 -- TUKEY COMPARISON OF PROCESSING TIME MEANS

```
STATISTIX 3.5                                                    31 AUG 91, 1:41
ID: Phase II Processing Times (Buyers 4, 5, and 8 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS        MS        F       P
------  ----  --------  --------  ------  ------
BETWEEN    1   203.8     203.8     53.02  0.0000
WITHIN   122   469.1     3.845
TOTAL    123   672.9

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   18.50     1    0.0000

COCHRAN'S Q                  0.7555
LARGEST VAR / SMALLEST VAR   3.089

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  3.229
EFFECTIVE CELL SIZE                        61.9

                        SAMPLE     GROUP
  SYSTEM      MEAN      SIZE     STD DEV
 --------  ----------  ------  ----------
    1       5.300       60     2.431
    2       2.734       64     1.383

 TOTAL      3.976      124     1.961

CASES INCLUDED 124    MISSING CASES 26
```

FIGURE 5-27 -- PHASE II SUBSET ONE PROCESSING TIMES ANOVA

```
STATISTIX 3.5                                                    31 AUG 91, 1:44
ID: Phase II Processing Times (Buyers 4, 5, 7, and 8 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------    ------  ------
BETWEEN    1   240.7      240.7       65.77  0.0000
WITHIN   105   384.2      3.659
TOTAL    106   624.9

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   26.29     1    0.0000

COCHRAN'S Q                  0.8127
LARGEST VAR / SMALLEST VAR   4.338

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  4.440
EFFECTIVE CELL SIZE                        53.4

                        SAMPLE     GROUP
  SYSTEM      MEAN      SIZE     STD DEV
 --------  ----------  ------  ----------
    1       5.431       51      2.476
    2       2.429       56      1.189

 TOTAL      3.860       107     1.913

CASES INCLUDED 107    MISSING CASES 13
```

FIGURE 5-28 -- PHASE II SUBSET TWO PROCESSING TIMES ANOVA

```
STATISTIX 3.5                                                    31 AUG 91, 1:51
1   Phase II Processing Times (Buyers 3, 5, 7, and 8 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------   ------  ------
BETWEEN    1   187.5      187.5      65.69  0.0000
WITHIN   102   291.1      2.854
TOTAL    103   478.6

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   26.56     1    0.0000

COCHRAN'S Q                  0.8178
LARGEST VAR / SMALLEST VAR   4.488

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  3.556
EFFECTIVE CELL SIZE                        51.9

                        SAMPLE     GROUP
  SYSTEM      MEAN       SIZE     STD DEV
 --------  ----------  ------  ----------
    1       4.780        50     2.188
    2       2.093        54     1.033

 TOTAL      3.385       104     1.689

CASES INCLUDED 104    MISSING CASES 16
```

FIGURE 5-29 -- PHASE II SUBSET THREE PROCESSING TIMES ANOVA

```
STATISTIX 3.5                                                    31 AUG 91, 2:03
ID: Phase II Processing Times (Buyers 2, 3, and 7 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------    ------  ------
BETWEEN    1   69.96      69.96       50.26  0.0000
WITHIN   131   182.3      1.392
TOTAL    132   252.3

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   28.68     1    0.0000

COCHRAN'S Q                  0.7976
LARGEST VAR / SMALLEST VAR   3.940

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  1.032
EFFECTIVE CELL SIZE                        66.5

                        SAMPLE     GROUP
  SYSTEM      MEAN       SIZE     STD DEV
 --------  ----------  ------  ----- ----
    1       3.554        65     1.500
    2       2.103        68     7.559E-01

 TOTAL      2.812       133     1.180

CASES INCLUDED 133    MISSING CASES 17
```

FIGURE 5-30 -- PHASE II SUBSET FOUR PROCESSING TIMES ANOVA

```
STATISTIX 3.5                                                    31 AUG 91, 2:15
ID: Phase II Processing Times (Buyers 1, 2, 3, 6, and 7 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------   ------  ------
BETWEEN    1   23.92      23.92      24.32  0.0000
WITHIN    77   75.75      0.984
TOTAL     78   99.67

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   14.75     1    0.0001

COCHRAN'S Q                 0.7830
LARGEST VAR / SMALLEST VAR   3.608

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  5.808E-01
EFFECTIVE CELL SIZE                          39.5

                        SAMPLE     GROUP
  SYSTEM      MEAN       SIZE     STD DEV
 ----  ---  ----------  ------  ----------
    1       3.026        39     1.246
    2       1.925        40     6.558E-01

 TOTAL      2.468        79     0.992

CASES INCLUDED 79    MISSING CASES 11
```

FIGURE 5-31 -- PHASE II SUBSET FIVE PROCESSING TIMES ANOVA

```
STATISTIX 3.5                                                    31 AUG 91, 2:17
ID: Phase II Processing Times (Buyers 1, 2, 3, 4, 6, and 7 Removed)

ONE WAY AOV FOR TIME = SYSTEM

SOURCE    DF      SS         MS         F       P
------  ----  --------   --------    ------  ------
BETWEEN    1   14.00      14.00       13.10  0.0007
WITHIN    54   57.71      1.069
TOTAL     55   71.71

                    CHI SQ    DF       P
BARTLETT'S TEST OF  ------  ------  ------
   EQUAL VARIANCES   15.07     1    0.0001

COCHRAN'S Q                 0.8292
LARGEST VAR / SMALLEST VAR   4.855

COMPONENT OF VARIANCE FOR BETWEEN GROUPS  4.618E-01
EFFECTIVE CELL SIZE                        28.0

                        SAMPLE     GROUP
  SYSTEM      MEAN      SIZE     STD DEV
 --------  ----------  ------  ----------
    1       2.929       28     1.331
    2       1.929       28     6.042E-01

 TOTAL      2.429       56     1.034

CASES INCLUDED 56    MISSING CASES 4
```

FIGURE 5-32 -- PHASE II SUBSET SIX PROCESSING TIMES ANOVA

Buyers Selected (Phase III Impact).

Problem. The purpose of Phase III testing was to determine if the system could assist buyers, not familiar with the items being procured, in making a better and more timely award decision. Without the benefit of having a control group, comprised in its entirety of individuals knowledgeable in the procurement of the items being examined, the intended goal of Phase III cannot be directly reached.

Correction. The researcher, through earlier interviews, knows at least two of the Phase II buyers are knowledgeable in the procurement of MilSpec 55182 items. However, because of anonymity, the control number assigned to those buyers during testing is unknown. Conventional wisdom dictates those buyers most familiar with the items, should generate the best scores. This reasoning will also provide the most stringent criteria against which to compare the buyers participating in Phase III testing.

Error Rate. The number of errors made by each buyer in Phase II testing were reviewed. The two buyers having the fewest errors, a composite of manual and automated scores, were used in a nonparametric analysis with the eight buyers of Phase III. A statistical test providing a nonparametric ANOVA test is the Kruskal-Wallis One Way AOV (4:222).

From the combined stages of Phase II testing, buyer Four had one error and buyers Two, Six, and Eight had two errors. Tukey's comparison of means of errors for Phase II buyers was performed with buyers Two, Six, and Eight (Figure 5-33). All three of these buyers error rates were not significantly different from each other. Therefore, as it is statistically impossible to differentiate between the three buyers, the Phase III results will be compared only to buyer Four. Figure 5-34 shows the results of this ANOVA test. With a p-value of 0.2757, there can be no more than seventy-two percent confidence that the inexperienced buyers performed as well as the experienced buyer.

```
STATISTIX 3.5                                                    31 AUG 91, 9:24
ID: Phase II Processing Errors  (Buyers 1, 3, 4, 5, and 7 Removed)

TUKEY (HSD) PAIRWISE COMPARISONS OF MEANS OF ERROR BY BUYER

                          HOMOGENEOUS
  BUYER        MEAN       GROUPS
--------  ----------  ----------
     2      6.667E-02   I
     6      6.667E-02   I
     8      6.667E-02   I

THERE ARE NO SIGNIFICANT PAIRWISE DIFFERENCES AMONG THE MEANS.

CRITICAL Q VALUE                3.373   REJECTION LEVEL    0.050
CRITICAL VALUE FOR COMPARISON  0.0000
STANDARD ERROR FOR COMPARISON  0.0000
```

FIGURE 5-33-- TUKEY'S COMPARISON OF LOW ERROR BUYERS

```
STATISTIX 3.5                                                    31 AUG 91, 9:38
ID: Phase II/III Processing Errors (Include Only Buyer 4 From Phase II)

KRUSKAL-WALLIS ONEWAY NONPARAMETRIC AOV FOR ERROR = PHASE

              MEAN    SAMPLE
  PHASE       RANK     SIZE
--------    ------   ------
    2         63.5      15
    3         68.6     120
 TOTAL        68.0     135

KRUSKAL-WALLIS STATISTIC                        1.1964
P VALUE, USING CHI-SQUARED APPROXIMATION        0.2740

PARAMETRIC AOV APPLIED TO RANKS

SOURCE    DF      SS         MS         F       P
------  ----  --------  --------     ------  ------
BETWEEN    1   341.7      341.7        1.20  0.2757
WITHIN   133   3.793E+04  285.2
TOTAL    134   3.827E+04

TOTAL NUMBER OF VALUES WHICH WERE TIED  135
MAX. DIFF. ALLOWED BETWEEN TIES     1.0E-0005

CASES INCLUDED 135    MISSING CASES 0
```

FIGURE 5-34 -- PHASE II AUTOMATED (BUYER 4) / PHASE III COMPARISON OF ERRORS ANOVA

A second nonparametric ANOVA was performed. This time comparing the performance of the Phase III buyers with the errors recorded by buyer Four from the Phase II manual testing. A low p-value will indicate a significant difference in the number of errors recorded by the Phase III buyers. The ANOVA test computed a p-value of 0.9078 (Figure 5-35). This is a very negligible indication that the Phase III buyers' performance was statistically different to the 'best' Phase II buyer using the manual method.

From this series of testing, it is demonstrated an inexperienced buyer, using the prototype, can perform at least as well as an experienced buyer using the current manual system for vendor selection (when comparing error rates). There remains the possibility that an experienced buyer can out perform an inexperienced buyer when they are both using the prototype.

```
STATISTIX 3.5                                                    31 AUG 91, 10:00
ID: Phase II (Manual) (Buyer 4) / Phase III Processing Errors

KRUSKAL-WALLIS ONEWAY NONPARAMETRIC AOV FOR ERROR = PHASE

              MEAN    SAMPLE
  PHASE       RANK     SIZE
--------    ------   ------
    2         67.5      15
    3         68.1     120
 TOTAL        68.0     135

KRUSKAL-WALLIS STATISTIC                          0.0134
P VALUE, USING CHI-SQUARED APPROXIMATION          0.9078

PARAMETRIC AOV APPLIED TO RANKS

SOURCE    DF      SS          MS          F       P
------   ----  --------   --------    ------  ------
BETWEEN    1   4.219       4.219        0.01  0.9084
WITHIN   133   4.218E+04   317.2
TOTAL    134   4.219E+04

TOTAL NUMBER OF VALUES WHICH WERE TIED  135
MAX. DIFF. ALLOWED BETWEEN TIES      1.0E-0005

CASES INCLUDED 135    MISSING CASES 0
```

FIGURE 5-35 -- PHASE II MANUAL (BUYER 4) / PHASE III PROCESING ERRORS ANOVA

Processing Times. The processing time recorded by each buyer in Phase II testing were reviewed. The two buyers with lowest composite manual and automated times, were used in a nonparametric analysis with the eight buyers of Phase III. The Kruskal-Wallis One Way AOV nonparametric ANOVA test was used to compare the results.

From the combined Phase II testing the best two processing times were 2.214 and 2.643 minutes per purchase request. These times were recorded by buyers Five and Eight respectively. An ANOVA was performed with only buyers Five and Eight from Phase II automated and with all buyers included from Phase III. As before, a small p-value indicates a difference exists in the mean processing times recorded by the two groups. With a p-value of 0.0002 (Figure 5-36), there is very strong statistical evidence indicating there is a difference in the mean processing times of the two groups.

```
STATISTIX 3.5                                                    31 AUG 91, 11:22
ID: Phase II Automated (Buyers 5 and 8) / Phase III Processing Times

KRUSKAL-WALLIS ONEWAY NONPARAMETRIC AOV FOR TIME = PHASE

              MEAN    SAMPLE
  PHASE       RANK     SIZE
--------    ------   ------
    2         43.9      28
    3         72.6     104
 TOTAL        66.5     132

KRUSKAL-WALLIS STATISTIC                          13.4903
P VALUE, USING CHI-SQUARED APPROXIMATION           0.0002

PARAMETRIC AOV APPLIED TO RANKS

SOURCE    DF      SS          MS          F       P
------  ----   --------   --------   ------  ------
BETWEEN    1   1.816E+04  1.816E+04   14.92  0.0002
WITHIN   130   1.582E+05  1.217E+03
TOTAL    131   1.764E+05

TOTAL NUMBER OF VALUES WHICH WERE TIED  132
MAX. DIFF. ALLOWED BETWEEN TIES     1.0E-0005

CASES INCLUDED 132    MISSING CASES 18
```

FIGURE 5-36 -- PHASE II AUTOMATED (BUYER 5 & 8) / PHASE III PROCESSING TIMES ANOVA

The question remains, which group, the experienced or the inexperienced buyers, had the lowest processing times. By examining the means reported by the descriptive statistics of the two groups (Figure 5-37), the two fastest Phase II individuals are almost a full minute per purchase request faster than the Phase III buyers.

STATISTIX 3.5 31 AUG 91, 11:36

ID: Phase II Automated (Buyers 5 and 8) / Phase III Processing Times

DESCRIPTIVE STATISTICS

	PHASE2	PHASE3
CASES	28	104
LOWER 95.0% C.I.	1.694	2.675
MEAN	1.929	2.971
UPPER 95.0% C.I.	2.163	3.267
S.D.	6.042E-01	1.523
S.E. (MEAN)	1.142E-01	1.493E-01
C.V.	31.33	51.26
MINIMUM	1.000	1.000
MEDIAN	2.000	3.000
MAXIMUM	3.000	8.000

FIGURE 5-37 -- PHASE II AUTOMATED (BUYERS 5 & 8) / PHASE III PROCESSING TIME DESCRIPTIVE STATISTICS

A second nonparametric ANOVA was performed. This time comparing the performance of the Phase III buyers with the processing time recorded by the two fastest Phase II buyer's manual time. A low p-value will indicate a significant difference in the processing time recorded by the Phase III buyers. The ANOVA p-value of 0.9747 (Figure 5-38), is a very weak indication that the Phase III buyers' performance was statistically different from the 'best' Phase II buyers using the manual method.

From this series of testing, it is demonstrated an inexperienced buyer, using the prototype can perform at least as well as an experienced buyer using the current manual system for vendor selection (when comparing processing time). However, the experienced buyer can out perform an inexperienced buyer when they are both using the prototype.

```
STATISTIX 3.5                                                    31 AUG 91, 11:52
ID: Phase II Manual (Buyers 5 and 8) / Phase III Processing Times

KRUSKAL-WALLIS ONEWAY NONPARAMETRIC AOV FOR TIME = PHASE

              MEAN    SAMPLE
  PHASE       RANK     SIZE
--------    ------   ------
    2         66.3      28
    3         66.6     104
 TOTAL        66.5     132

KRUSKAL-WALLIS STATISTIC                          0.0010
P VALUE, USING CHI-SQUARED APPROXIMATION          0.9747

PARAMETRIC AOV APPLIED TO RANKS

SOURCE    DF      SS         MS         F       P
------  ----   --------   --------   ------  ------
BETWEEN    1   1.371      1.371       0.00  0.9748
WITHIN   130   1.781E+05  1.370E+03
TOTAL    131   1.781E+05

TOTAL NUMBER OF VALUES WHICH WERE TIED  132
MAX. DIFF. ALLOWED BETWEEN TIES     1.0E-0005

CASES INCLUDED 132    MISSING CASES 18
```

FIGURE 5-38 -- PHASE II MANUAL (BUYERS 5 & 8) / PHASE III COMPARISON OF PROCESSING TIMES ANOVA

Results. Because only two buyers in Phase II where known to possess experience in MilSpec 55182 items, the results from that phase had to be re-examined. The two best scores were identified from each portion of the Phase II testing. Whether or not these scores represents the efforts of the experienced buyers, is inconsequential. If the scores do belong to the experienced buyers, the true level of buyer performance which is being used as a reference, is properly established. If they do not belong to the experienced buyers, the level of buyer performance being used as a reference, is raised by the unknown difference in performance between the experienced and the inexperienced buyers. The result is a higher performance level the Phase III buyers have to achieve before their performance can be considered comparable to Phase II, the norm.

The results obtained from this phase of testing demonstrates that the Phase III buyers (buyers without prior experience) can use the prototype and perform the vendor selection process, at least as well as the experienced buyers using the manual method.

Conclusions

Phase I testing sought an answer to the question, 'Does the system provide the correct information?' The data obtained clearly indicates the prototype does provide the correct information on which to base an award decision. Because not all purchase requests were processed using both the automated and manual systems, and all of the buyers participating in Phase II were not familiar with the vendor selection process of MilSpec 55182 items, conclusions to the remaining phases cannot be as succinct.

The objective of Phase II testing was to address the question 'Is the buyer able to select the correct vendor using the prototype system?' There was a significant improvement in the error rate experienced by the buyers when using the prototype, as well as improvement in the processing time. A panel review the complexity of the purchase requests. Their analysis indicates the improvements demonstrated by the prototype could not be explained by the purchase requests being processed by the automated system were 'easier'.

Phase III testing wanted to provide an answer to the question 'Is the system designed such that, a person unfamiliar with the items being procured, is able to make a valid vendor selection decision?' Without all buyers in Phase II being familiar with the items, the results obtained are not as strong as they could have been. Regardless, it was demonstrated that using the prototype the novice has the capability to make an award decision, that is at least as accurate and timely as the manual decisions made by the 'best' of the those individuals familiar with the items being considered.

VI. Summary, Findings, and Recommendations

Overview

The process followed to reach the conclusions drawn from this paper is outlined below. A summary of the research methodology, is presented. After which, the research findings, and recommendations for prototype enhancements and follow-on research are offered.

Summary of Research

The current small purchase vendor selection process at DESC relies on a manual system to generate the award decision. The current process of small contract award determination requires a significant amount of labor to acquire the most basic of data. In addition, to assure a proper decision is made, the buyer must maintain constant surveillance on dynamic information, stemming from many sources. As a result, the award process is subject to degradation, and doubts have arisen concerning the quality of those decisions. The primary objective of this research project was to determine whether improvements in the current small contracting process were possible.

To this end, a series of meetings was held with DESC to investigate two preliminary questions. The first asked was, 'What was the user's perception of the problem?' The second asked, 'How was the current vendor selection process at DESC conducted?' These questions were addressed, and the methodology for this research devised. This approach comprised the design and development of a computer based decision support system. A prototype running on a personal computer, capable of analyzing data obtained from actual data files, resulted. The prototype coding was verified and a formal validation plan was developed. Using the testing procedure documented in Chapter V, the prototype was validated with the help of DESC personnel. The results of the testing were analyzed, and they are summarized for the reader in the Findings Section below.

Findings

The research questions presented in Chapter I are repeated below along with each conclusion.

Research Question 1. What information must the buyer obtain before selecting the proper vendor?

Conclusion 1. To answer this question a series of interviews was conducted with the buyers at DESC. The minimum information the buyer requires to make an award decision are:

a) the identity of the item required (either Type Number or NSN)

b) the quantity of the item required

c) the identity of the vendors offering the item for sale

d) the vendor's selling price for the item

e) the identity of DeBarred vendors

The following information enables the buyer to make a better informed decision regarding the vendor award:

a) quantity price reduction for the item of interest

b) FOB origin or destination

c) delivery time

d) performance problems with the vendors

e) performance problems with the products

f) past purchasing information for the item

The above items were incorporated into the prototype and tested in the validation process. The result of testing suggests the prototype did incorporate the items necessary for the buyers to make an intelligent vendor selection.

Research Question 2. What information does the buyer generate while awarding a contract to the vendor?

Conclusion 2. This question was answered by interviews with the buyers. The DESC Form 800 is generated by the buyers after making the award decision. The significant elements of this form were identified. Information required to complete the form was assembled on the award screen in the prototype. Buyer interviews confirmed the information presented on the award screen was sufficient to process the required DESC Form 800.

Research Question 3. What automated management systems are available, and, of these systems, which ones could satisfy the needs of DESC, given the type of data available and the results required?

Conclusion 3. A literary review was conducted. The review focused on the various automated management systems commonly used today. Three types were reviewed: a data base management system, a decision support system, and an expert system. Of these management systems, the decision support system appeared to be the closest match for DESC's problem.

DESC sought a system that would assist their buyers in performing the vendor selection process. They were looking for a system that would organize information relevant to specific requests, thereby enabling the making of timely, informed decisions. A decision support system supports this open-ended decision analysis. The user provides the constraints of the problem and the decision support system generates possible alternative solutions. The user then employs personal insights to select the best solution from the alternatives presented.

The validation results, and feedback from the user questionnaire, confirmed developing the prototype in the vein of a decision support system was sound. By using the

prototype, the buyers tested were able to achieve a significant reduction in errors. The percentage of errors decreased from 21.7% using the current process, to 5.8% using the prototype. Processing time was almost cut in half. The prototype reduced the time required for each request by two minutes.

In the questionnaires completed by the buyers, not one indicated the approach used by the prototype was incorrect. Acceptance of the system was unilateral. They are willing to adopt this system into their working environment, and are eager to do so.

Because of the above results, structuring the prototype design based on a decision support system, proved to be both theoretically and functionally correct.

Research Question 4. Can an effective automated system be designed, developed and employed to assist the buyer decision process at DESC?

Conclusion 4. Yes, without reservation. As stated in conclusion three, when the buyers used the prototype, there was a significant reduction in errors produced in processing the purchase request. Not only were there fewer errors, but it took less time to process the requests as well.

The system can also be successfully used by personnel who are unfamiliar with the products. The test results confirm that an inexperienced buyer can perform the vendor selection process at least as well as the best buyers using the manual system today. The implications of this finding bear directly on the department managers. Flexibility in personnel utilization can be enhanced. No longer will the work have to wait on 'Mary' or 'Joe' to return from vacation. The workload can be effectivly shared by all buyers.

There is overwhelming evidence indicating this prototype system is a valuable tool in the vendor selection process. With DESC's desire to bring cohesiveness to the award process, and the ever shrinking pool of resources in which to operate, it is clear the current methods of doing business must be re-examined. Developing and implementing the

prototype is a proven solution that will enhance the productivity of the small purchase, vendor selection process. A process which consumes 87% of the contracting workload at DESC.

Summary Of Findings

The goal of this research was to demonstrate improvements in the current small contract vendor selection process were possible. Through personal interviews, knowledge of the current process was obtained. Further investigations identified deficiencies in the methods used for vendor selection. A system was designed striving to reduce the number of obstacles to the process.

Simplicity for the user was the primary concern in system design. A balance was sought between too little and too much information on the user screens. Maximizing the utility of the system with a minimum of user inputs was the design goal.

The prototype that evolved from this effort was tested at DESC, by the very buyers who the system was designed to assist.

The results of the prototype testing showed it is possible to achieve a significant reduction in purchase request processing time while increasing the accuracy of the award decisions. Usability of the system by those unfamiliar with the items being procured was demonstrated in the third phase of testing. The timeliness and quality of the decisions made by this group were equivalent to those made by experienced buyers using the manual process.

From the analysis of the test data, and the responses provided by the users, the researcher is confident the system developed improves the vendor selection process.

Recommendations for Future Research

Before the seed sown by this research will bear fruit, it must be nurtured by other research. VASPP is still primarily a concept. This research has established a point of

departure for further development of the VASPP system, however, there is still much undone. Before VASPP can be realized, an interface for the vendor to enter VASPP, must be designed. Along with this, the logic required to govern vendor data input verification must be examined.

The prototype, with its dependence on data from many sources, is very reliant on the integrity of its support files. Structure and control of the vendor pricing data file must be developed to assure its integrity. Data maintenance and transfer from all supporting data files needs to be addressed. Without accurate information available to the system, inferior performance can be expected.

The vendor selection process can be enhanced beyond that demonstrated by the prototype. Both the upstream and downstream activities are automated. Purchase request transmittal to and from the buyer should be examined to take advantage of a computer to computer information transfer. Achievement of this interface will reduce the generation of paper products and personnel overhead, while increasing throughput and improving the accuracy of the products produced.

Recommendation for Future Modification

The prototype was developed using a decision support system as a model. The development of an expert system went beyond the time limits constraining this paper. However, now having established a solid foundation that identifies the requirements of the buyers, it seems possible an expert system can be developed. The question, 'Can the vendor selection process be defined with sufficient depth to develop an expert system?', needs to be re-addressed. If this is possible, an expert system overlay for the prototype could completely automate the vendor selection process.

An automated system already produces purchase request information, and the buyers submit their award decisions to another automated system. With an expert system

performing the award decision, a seamless transition could take place between the requirement identification and contract award. This could result in a completely automated small vendor award process, increasing decision integrity and decreasing lead time.

Lessons Learned

A significant portion of the success of this project rests with the cooperation afforded to the researcher by DESC. Prior knowledge with the acquisition process was minimal. The personnel eagerly answered questions and patiently reiterated the vendor selection process as necessary. The significant lesson learned from these efforts is the importance of maintaining an open line of communication between the user and the developer. In this research it was doubly important. Not only did the expectations of management have to be satisfied, but also the needs of the system user had to be carefully cultivated. Without constant communication with the customer, a successful system could not have been developed.

Final Notes

The use of an automated system has been shown to increase the effectiveness of the vendor selection process. This is but one of countless areas where productivity could be improved with the judicious use of automated techniques. With government being forced to accept an ever increasing work load, while, simultaneously, resources are being denied, productivity must be improved wherever possible. Managers should not overlook the benefits of properly applied computer support.

Appendix A: Program Code

Copyright, United States Air Force, 1991
Automated Vendor Selection Assistant

```
*:*********************************************************************
*:
*:      Program: AVSA.PRG
*:
*:       System: Automated Vendor Selection Assistant
*:       Author: Capt Daniel E. Hagmaier
*:    Copyright (c) 1991, United States Air Force
*:
*:        Calls: TITLESCR--procedure
*:             : INFO_SCR--procedure
*:             : INITLSCR--procedure
*:             : INPUTSCR--procedure
*:             : SELCTSCR--procedure
*:             : SELCTVEN.PRG
*:             : ANALZSCR--procedure
*:             : PREPVEN.PRG
*:             : PRICESCR--procedure
*:             : VENDRSCR--procedure
*:             : CDCFSCR--procedure
*:             : PROBMSCR--procedure
*:             : AWARDSCR--procedure
*:             : NOVENSCR--procedure
*:
*:         Uses: PR_TEMP.DBF
*:             : HOLD.DBF
*:
*: Documented: 7/30/91        11:34                    SNAPI version 1.73
*:*********************************************************************



***    INITIALIZE    ***********************************************
*                                                                   *
*   Establishes the configuration the system.  It defines the      *
*   way the display screen appears to the user.  It also           *
*   defines some of the operating parameters for the program.      *
*                                                                   *
*********************************************************************

SET BELL OFF                                  && Suppresses the 'Beep'
SET CENTURY OFF                               && Allows input of 2 digit year
SET COLOR TO G/B,RB/N,BG                      && Sets the display colors
SET DECIMALS TO 4                             && Numbers displayed w/ 4 decimals
SET DELETED ON                                && Ignores deleted records
SET ESCAPE OFF                                && Inhibits the <ESC> key
SET PROCEDURE TO screens                      && Opens the procedure file
SET SCOREBOARD OFF                            && Inhibits the line 0 prompts
SET STATUS OFF                                && Inhibits the status line
```

```
SET TALK OFF                                    && Inhibits command responses



***   Program Control   ****************************************
*                                                              *
*   Controls the execution of all programming routines, up to  *
*   user termination.                                          *
*                                                              *
****************************************************************



STORE " " TO mchoice                            && User selection
STORE .T. TO mnew_nsn                           && Program control flag
DO TitleScr                                     && Display opening screen
STORE .F. TO mend                               && Program termination flag
DO WHILE .NOT. mend                             && Run program until MEND = .T.
    DO Info_Scr                                 && Display information screen
    IF mchoice = "Y"                            && User's response


*       >>>   DECLARE SYSTEM VARIABLES   <<<

        Do InitlScr                             && User information screen
        IF mnew_nsn                             && This will be a new NSN


*           >>>   INITIALIZE MEMORY VARIABLES   <<<

            STORE SPACE(16) TO mnsn             && NSN of item
            STORE ' ' TO mreturn                && Last displayed user screen
            STORE ' ' TO msetaside              && Set-aside procurement
            STORE ' ' TO mhist_cage             && Most recent contracted vendor
            STORE ' ' TO mhist_date             && Most recent purchase date
            STORE 0 TO mhist_pr                 && Most recent purchase price
            STORE 0 TO mquantity                && Amount of item desired
            STORE 0 TO mlow_price               && Lowest cost to procure item
            STORE 0 TO mrdd                     && Required delivery date
            STORE 0 TO mday                     && The number of today's date
            STORE .F. TO mlow                   && Price may be to low flag
            STORE .F. TO munit_pr               && Display unit price flag
            STORE .F. TO mvariation             && Variation exceeds limit flag
            STORE .F. TO mhistory1              && Price greater than recent histo
ry
            STORE .F. TO mhistory2              && No prior NSN history flag


```

Copyright, United States Air Force, 1991
Automated Vendor Selection Assistant

```
*           >>>   INITIALIZE DISPLAY MATRIX VARIABLES   <<<

            STORE 1 TO mcounter                 && mcage1 through mcage9
            DO WHILE mcounter < 10
                STORE 'MCAGE'+LTRIM(STR(mcounter)) TO mcage
                STORE '      ' TO &mcage
                STORE mcounter + 1 TO mcounter
            ENDDO

            STORE 1 TO mcounter                 && morder1 through morder6
            DO WHILE mcounter < 7
                STORE 'MORDER'+LTRIM(STR(mcounter)) TO morder
                STORE 0 TO &morder
                STORE mcounter + 1 TO mcounter
            ENDDO

            STORE 1 TO mrow                     && mext_1_1 through mext_9_6
            DO WHILE mrow < 10
                STORE 1 TO mcolumn
                DO WHILE mcolumn < 7
                    STORE 'MEXT_'+LTRIM(STR(mrow))+'_'+LTRIM(STR(mcolumn)) TO mep
                    STORE 0 TO &mep
                    STORE mcolumn + 1 TO mcolumn
                ENDDO
                STORE mrow + 1 TO mrow
            ENDDO
        ENDIF


*       >>>   SET JULIAN DATE   <<<

        IF DAY(DATE()) <> mday                  && mday contain the current day?


*           >>>   GET THE SYSTEM DATE   <<<

            STORE YEAR(DATE()) TO myear
            STORE MONTH(DATE()) TO mmonth
            STORE DAY(DATE()) TO mday


*           >>>   CALCULATE THE DAYS IN THE PAST MONTHS   <<<

            DO CASE
            CASE mmonth = 1
                STORE 0 TO mj_date
            CASE mmonth = 2
                STORE 31 TO mj_date
```

```
            CASE mmonth = 3
                STORE 59 TO mj_date
            CASE mmonth = 4
                STORE 90 TO mj_date
            CASE mmonth = 5
                STORE 120 TO mj_date
            CASE mmonth = 6
                STORE 151 TO mj_date
            CASE mmonth = 7
                STORE 181 TO mj_date
            CASE mmonth = 8
                STORE 212 TO mj_date
            CASE mmonth = 9
                STORE 243 TO mj_date
            CASE mmonth = 10
                STORE 273 TO mj_date
            CASE mmonth = 11
                STORE 304 TO mj_date
            CASE mmonth = 12
                STORE 334 TO mj_date
            ENDCASE


*           >>>   ADD THE DAYS OF THE CURRENT MONTH   <<<

            STORE mj_date + mday TO mj_date


*           >>>   CORRECT FOR LEAP YEAR   <<<

            STORE IIF(MOD(myear,4) = 0,.T.,.F.) TO mleap_yr
            IF mleap_yr .AND. mj_date > 59
                STORE mj_date + 1 TO mj_date
            ENDIF
        ENDIF


*       >>>   INPUT AND ANALYZE USER REQUEST   <<<

        SET CONFIRM ON                          && Must use <CR> to terminate inpu
t
        DO InputScr                             && Get users inputs
        IF mchoice = 'Q'    && Has the input been aborted
            LOOP     && Return to beginning of Do While
        ENDIF
        DO SelctScr                             && Display information screen
        DO SelctVen                             && Get bidding vendors

```

```
        IF RECCOUNT() > 0                       && If bidding vendors exist
            DO AnalzScr                         && Display information screen
            DO PrepVen                          && Prepare vendors for display


*           >>>   DISPLAY USER SCREENS   <<<

            STORE ' ' TO mchoice                && Reset user's choice
            DO WHILE UPPER(mchoice) <> 'Q'
                DO CASE
                CASE UPPER(mchoice) = 'U'
                    STORE .T. TO munit_pr
                    DO PriceScr
                CASE UPPER(mchoice) = 'E'
                    STORE .F. TO munit_pr
                    DO PriceScr
                CASE UPPER(mchoice) = 'V'
                    DO VendrScr
                CASE UPPER(mchoice) = 'C'
                    DO CdcfScr
                CASE UPPER(mchoice) = 'P'
                    DO ProbmScr
                CASE UPPER(mchoice) = 'A'
                    DO AwardScr
                OTHERWISE
                    STORE .T. TO munit_pr
                    DO PriceScr
                ENDCASE
            ENDDO

            STORE .T. TO mnew_nsn

        ELSE                                    && If no vendors qualify
            DO NoVenScr
        ENDIF
    ELSE                                        && If user is finished
        STORE .T. TO mend                       && Set MEND = .T.
    ENDIF


*   >>>   PREPARE DATA FILES FOR NEXT USE   <<<

    CLOSE DATABASES
    SET SAFETY OFF                              && Allow unprompted deletion
    USE pr_temp
    ZAP                                         && Remove records from pr_temp.dbf
    USE hold
    ZAP                                         && Remove records from hold.dbf
```

```
    SET SAFETY ON

ENDDO


***   CLEAN-UP   ***********************************************
*                                                               *
*   This section closes open files, releases the memory         *
*   variables, and restores dBase to its default operating      *
*   environment.                                                *
*                                                               *
*****************************************************************

CLEAR ALL                                   && Closes all files & memory
SET BELL ON                                 && Enables the 'Beep'
SET CONFIRM OFF                             && Enables Auto Advance
SET DECIMALS TO 2                           && Numbers displayed w/ 2 decimals
SET DELETED OFF                             && Activates deleted records
SET ESCAPE ON                               && Enables the <ESC> key
SET SCOREBOARD ON                           && Enable line 0 display
SET STATUS ON                               && Enables the status bar
SET TALK ON                                 && Enables command responses

*: EOF: AVSA.PRG
```

```
*:*********************************************************************
*:
*:      Program: PREPVEN.PRG
*:
*:       System: Automated Vendor Selection Assistant
*:       Author: Capt Daniel E. Hagmaier
*:    Copyright (c) 1991, United States Air Force
*:
*:    Called by: AVSA.PRG
*:
*:         Uses: CDCF.DBF
*:             : QUALITY.DBF
*:             : VENDOR.DBF
*:             : HOLD.DBF
*:             : MODEL.DBF
*:             : HISTORY.DBF
*:
*:      Indexes: CDCF_N_C.NDX
*:             : Q_CAGE.NDX
*:             : V_C_MIL.NDX
*:             : H_EXT_PR.NDX
*:             : H_ORD_Q.NDX
*:             : HIST_N_D.NDX
*:
*: Documented: 7/30/91       11:37                    SNAP! version 1.73
*:*********************************************************************



***   CHECK VENDOR PERFORMANCE   ******************************
*                                                              *
*   This section of the program check each vendor remaining in  *
*   the data file PR_TEMP.  They are checked for past perform-  *
*   ance problems as well as outstanding performance. Flags are *
*   set for each vendor indicating the results of this search.  *
*                                                              *
****************************************************************

*   >>>   CHECK FOR PROBLEM VENDOR INFORMATION   <<<

SELECT pr_temp                                  && Activate PR_TEMP.DBF
GO TOP                                          && Set pointer to first record
DO WHILE .NOT. EOF()                            && Scan entire file
    SELECT dcrl                                 && Activate DCRL.DBF
    SEEK pr_temp->cage                          && See if cage exists in DCRL.DBF
    IF FOUND()                                  && If cage is in the DCRL.DBF
        IF restrict1 <> ' ' .OR. restrict2 <> ' ' .OR. restrict3 <> ' ';
            .OR. restrict4 <> ' ' .OR. restrict5 <> ' '
```

```
            REPLACE pr_temp->prob WITH .T.
        ENDIF
    ENDIF
    SELECT pr_temp                          && Activate PR_TEMP.DBF
    SKIP                                    && Move pointer to next record
ENDDO                                       && Repeat until End-Of-File


*   >>>   CHECK CDCF FILE   <<<

SELECT C                                    && Establish 2nd work area
USE cdcf INDEX cdcf_n_c                     && Open CDCS file

SELECT pr_temp
GO TOP
DO WHILE .NOT. EOF()
    SELECT cdcf
    SEEK mnsn+pr_temp->cage                 && Look to see if exists
    IF FOUND()
        REPLACE pr_temp->cdcf WITH .T.
    ENDIF
    SELECT pr_temp                          && Activate PR_TEMP.dbf
    SKIP                                    && Advance pointer to check next r
ecord
ENDDO


*   >>>   CHECK QUALITY VENDOR FILE   <<<

SELECT C                                    && Establish alternate work area
USE quality INDEX q_cage                    && Open quality file

SELECT pr_temp                              && Activate PR_TEMP.DBF
GO TOP                                      && Set pointer to first record
DO WHILE .NOT. EOF()                        && Scan the entire file
    SELECT quality                          && Activate QUALITY.DBF
    SEEK pr_temp->cage                      && See if cage exists in QUALITY.D
BF
    IF FOUND()                              && If cage is in QUALITY.DBF
        REPLACE pr_temp->quality WITH .T.   && Set the quality flag for the ve
ndor
    ENDIF
    SELECT pr_temp                          && Activate PR_TEMP.DBF
    SKIP                                    && Move pointer to next record
ENDDO                                       && Repeat until End-Of-File
```

```
***   ORGANIZE INFORMATION   ************************************
*                                                                *
*   This section of the program organizes the selected vendors  *
*   on minimum quantity offered which satisfies the requirement.*
*   A matrix of memory variables are filled, which will latter *
*   be displayed by the user screens.                            *
*                                                                *
******************************************************************

*   >>>   ESTABLISH DATA FILES   <<<

SELECT B                                        && Select an alternate work area
USE vendor INDEX v_c_mil                        && Activate VENDOR.dbf
SELECT C                                        && Select an alternate work area
USE hold                                        && Open a temporary storage db fil
e

SELECT pr_temp                                  && Activate primary work area
SET RELATION TO cage+mil_spec INTO vendor       && Link datafiles together
GO TOP                                          && Set pointer to the first record



******************************************************************
*                                                                *
*   The following code finds the first column in the temporary  *
*   vendor file who's quantity is equal to, or exceeds the      *
*   requirement.                                                 *
*                                                                *
******************************************************************

*   >>>   SCAN VENDOR PRICES   <<<

DO WHILE .NOT. EOF()                            && Examine all vendors
    STORE 1 TO mseries                          && Field pointer = 1
    STORE 'QMAX'+LTRIM(STR(mseries)) TO mmax    && Create pointer to QMAX1
    IF &mmax < mquantity .AND. &mmax <> 0       && Test QMAX1
        STORE .T. TO mnextcol                   && Set program control flag
    ELSE
        STORE .F. TO mnextcol
    ENDIF
    DO WHILE mnextcol .AND. mseries < 11        && Examine up to QMAX10
        STORE mseries+1 TO mseries              && Add one to series
        STORE 'QMAX'+LTRIM(STR(mseries)) TO mmax
        IF &mmax >= mquantity .AND. &mmax * mquantity >= vendor->min_order
            STORE .F. TO mnextcol
        ENDIF
        IF &mmax = 0                            && No further pricing information
```

```
            STORE .F. TO mnextcol                && Set program control flag
            STORE mseries - 1 TO mseries         && Correct mseries
        ENDIF
    ENDDO                                        && Repeat until all prices examine
d



**************************************************************
*                                                            *
*   Having found the minimum amount the vendor will sell that  *
*   meets the requirements, that and all subsequent price      *
*   breaks are transferred to a temporary data base named HOLD.*
*                                                            *
**************************************************************

*   >>>   MOVE PRICING INFORMATION TO HOLD.DBF   <<<

    STORE 'A->QMAX'+LTRIM(STR(mseries)) TO mmax
    DO WHILE mseries < 11
        IF &mmax = 0
            EXIT
        ENDIF
        SELECT hold
        APPEND BLANK
        REPLACE cage WITH a->cage
        STORE 'A->QMIN'+LTRIM(STR(mseries)) TO mmin
        STORE 'A->PRICE'+LTRIM(STR(mseries)) TO mprice
        REPLACE unit_price WITH &mprice

*       >>>   CALCULATE THE ORDER QUANTITY AND EXTENDED PRICE   <<<

        IF mquantity < &mmin
            REPLACE ext_price WITH &mmin * &mprice
            REPLACE ord_quant WITH &mmin
        ELSE
            IF &mmax < mquantity
                STORE (INT(mquantity/&mmax)+1)*&mmax TO mquant
                REPLACE ext_price WITH mquant * &mprice
                REPLACE ord_quant WITH mquant
            ELSE
                REPLACE ext_price WITH mquantity * &mprice
                REPLACE ord_quant WITH mquantity
            ENDIF
        ENDIF

*       >>> CHECK AND ADJUST LOT SIZE   <<<

```

```
        IF vendor->lot_size > 1                 && Is an adjustment required?
            IF INT(ord_quant/vendor->lot_size) <> ord_quant/vendor->lot_size
                STORE 1 TO munits
                DO WHILE vendor->lot_size*munits < ord_quant
                    STORE munits + 1 TO munits
                ENDDO
                REPLACE ord_quant WITH vendor->lot_size*munits
                REPLACE ext_price WITH vendor->lot_size*munits*unit_price
            ENDIF
        ENDIF

*       >>>   CHECK FOR MINIMUM VENDOR ORDER QUANTITY   <<<

        IF ext_price < vendor->min_order
            STORE vendor->min_order/unit_price TO munits
            IF INT(munits) <> munits
                STORE INT(munits + 1) TO munits
            ENDIF
            REPLACE ord_quant WITH munits
            REPLACE ext_price WITH munits * unit_price
        ENDIF

        STORE mseries + 1 TO mseries
        STORE 'A->QMAX'+LTRIM(STR(mseries)) TO mmax
    ENDDO
    SELECT pr_temp
    SKIP
ENDDO



*****************************************************************
*                                                               *
*   Now, in the HOLD data file, is a list of all qualified      *
*   vendors who have bid on the item.  Along with the cage      *
*   code, the associated quantity and extended price are        *
*   stored.  The next instructions identifies the lowest pur-   *
*   chase price and sets specific data flags concerning the     *
*   lowest price.                                               *
*                                                               *
*****************************************************************

*   >>>   INFORM THE USER OF THE PROGRAM STATUS   <<<

CLEAR
@ 6,27 TO 10,49 DOUBLE
@ 8,30 SAY 'Organizing Vendors'
```

```


* >>>  FIND THE LOWEST COST  <<<

SELECT D                                    && Select an alternate work area
USE model                                   && Open the management MODEL.DBF
SELECT hold                                 && Activate the HOLD.DBF
SET INDEX TO h_ext_pr, h_ord_q              && Activate the indexes for HOLD.D
BF
REINDEX                                     && Update the indexes
GO TOP                                      && Move pointer to the first recor
d


*  >>>  COMPARE LOW PRICE TO NEXT LOWEST  <<<

STORE ext_price TO mlow_price               && Record #1 ext_price is lowest p
rice
STORE unit_price TO mnet_price              && Transfer unit price to memory
STORE cage TO mcage                         && Transfer cage to memory
LOCATE FOR cage <> mcage                    && Look for the next lowest vendor
IF FOUND()                                  && If another vendor exists
    IF mnet_price * ((model->low/100)+1) < unit_price
        STORE .T. TO mlow                   && If price too low, set flag
    ENDIF
ENDIF                                       && End of comparison


*  >>>  CHECK FOR VARIATION COSTS  <<<

SELECT pr_temp                              && Activate PR_TEMP.DBF
LOCATE FOR cage = mcage                     && Locate vendor with lowest price
IF mlow_price * ((vendor->qty_var_m/100)+1) > model->up_limit
    STORE .T. TO mvariation                 && Set flag
ENDIF                                       && End of variation check


*  >>>  CHECK HISTORY  <<<

SELECT C                                    && Select alternate work area
USE HISTORY INDEX hist_n_d                  && Activate HISTORY.DBF
SEEK mnsn                                   && Look for NSN
IF FOUND()                                  && If it is on file

*  >>>  FIND MOST RECENT PURCHASE  <<<

    DO WHILE nsn = mnsn                     && Go one record beyond matching N
SN
```

```
        SKIP                                    && Advance record pointer
    ENDDO
    SKIP -1                                     && Backup one record

    STORE date TO mhist_date                    && Transfer to memory variables
    STORE price TO mhist_pr
    STORE cage TO mhist_cage

*   >>>  COMPARE UNIT PRICE   <<<

    IF mnet_price > price * ((model->history1/100)+1)
        STORE .T. TO mhistory1                  && Set history flag
    ENDIF
ELSE                                            && If NSN is not on file
    IF mnet_price > model->history2             && If unit price exceeds limits
        STORE .T. TO mhistory2                  && Set history flag
    ENDIF
ENDIF                                           && [illegible] of history check



***   FILL MEMORY VARIABLES   ********************************
*                                                              *
*   The data contained in the hold data file is next organized *
*   for display.  This is accomplished by loading a matrix of  *
*   memory variables.                                          *
*                                                              *
****************************************************************

*   >>>   PLACE DATA IN MEMORY 'MATRIX'   <<<

SELECT C                                        && Select alternate work area
USE hold INDEX h_ord_q                          && Activate HOLD.DBF
GO TOP                                          && Set pointer to first record

STORE ord_quant TO morder1                      && Fill first matrix unit
STORE ord_quant TO mlast_ord                    && Store for program control
STORE cage TO mcage1                            && Fill first matrix unit
STORE ext_price TO mext_1_1                     && Fill first matrix unit
STORE ext_price TO mext_price                   && Store for program control

STORE 1 TO mrow                                 && Initialize pointer variable
STORE 1 TO mcolumn                              && Initialize pointer variable

SKIP                                            && Move pointer to next record

DO WHILE .NOT. EOF()                            && Fill matrix until EOF is reache
d
```

```
    IF ord_quant <> mlast_ord                   && Compare order quantities

*       >>>   MOVE TO NEXT COLUMN  <<<    && if not the same

        STORE mcolumn + 1 TO mcolumn            && Advance column
        IF mcolumn = 7                          && End of display?
            EXIT                                && if so, terminate
        ENDIF
        STORE 'MORDER'+LTRIM(STR(mcolumn)) TO mcell
        STORE ord_quant TO &mcell
        STORE ord_quant TO mlast_ord            && Update order quantity
    ENDIF

*   >>>  FIND PROPER ROW   <<<

    STORE 1 TO mrow                             && Reset row
    STORE .F. TO mflag                          && Program control
    DO WHILE .NOT. mflag                        && Look for row with matching cage
        STORE 'MCAGE'+LTRIM(STR(mrow)) TO mcage
        IF &mcage = cage .OR. &mcage = "    "
            STORE .T. TO mflag                  && Set flag when found
        ELSE
            STORE mrow + 1 TO mrow              && Advance to next row
        ENDIF
    ENDDO

    STORE cage TO &mcage
    STORE 'MEXT_'+LTRIM(STR(mrow))+'_'+LTRIM(STR(mcolumn)) TO mep
    STORE ext_price TO &mep
    SKIP                                        && Advance record pointer
ENDDO                                           && End of filling memory matrix



***   REMOVE HIGH QUANTITY VENDORS   **************************
*                                                              *
*   This code examines the quantity offered by the vendors and *
*   removes those vendors who's lowest quantity offered was so *
*   large, they did not make it into the memory matrix.        *
*                                                              *
****************************************************************

*     >>>   REMOVE HIGH QUANTITY VENDORS   <<<

IF morder6 > 0                                  && Not needed if matrix is not fil
led
    SELECT pr_temp                              && Activate PR_TEMP.DBF
    GOTO TOP                                    && Set pointer to first record
```

```
    DO WHILE .NOT. EOF()                    && Examine entire file
        IF qmin1 > morder6                  && QMIN1 greater than largest in t
he matrix
            DELETE                          && Mark vendor for deletion
        ENDIF
        SKIP                                && Advance pointer to next record
    ENDDO                                   && Repeat until End-Of-File is rea
ched
ENDIF                                       && End high quantity test



RETURN                                      && Return control to calling progr
am

*: EOF: PREPVEN.PRG
```

Copyright, United States Air Force, 1991
Automated Vendor Selection Assistant

```
*:*********************************************************************
*:
*:      Program: SCREENS.PRG
*:
*:       System: Automated Vendor Selection Assistant
*:       Author: Capt Daniel E. Hagmaier
*:    Copyright (c) 1991, United States Air Force
*:
*:    Called by: AVSA.PRG
*:
*:         Uses: NSN.DBF
*:             : VENDOR.DBF
*:             : DCRL.DBF
*:             : DCRLCODE.DBF
*:             : MODEL.DBF
*:             : CDCF.DBF
*:
*:      Indexes: N_NSN.NDX
*:             : V_C_MIL.NDX
*:             : DCR_CAGE.NDX
*:             : CDCF_N_C.NDX
*:
*: Documented: 7/30/91       11:05                    SNAP! version 1.73
*:*********************************************************************



*****************************************************************
*                                                               *
*   This is a procedure file containing the display screens for *
*   user.  The coding herein obtains the user's inputs, and     *
*   performs the necessary validation on those inputs.          *
*                                                               *
*****************************************************************



***   TITLE SCREEN   ********************************************
*                                                               *
*   This screen is the log-on screen for the program.           *
*                                                               *
*****************************************************************

PROCEDURE TitleScr                          && Labels this block of code
CLEAR                                       && Erases the screen


*   >>>   CREATE THE SCREEN   <<<
```

```
@ 0,0 TO 22,79 DOUBLE                          && Draws box around screen
@ 2,32 SAY "Welcome To The"                    && Print text to the screen
@ 6,30 SAY "A U T O M A T E D"
@ 8,33 SAY "V E N D O R"
@ 10,30 SAY "S E L E C T I O N"
@ 12,30 SAY "A S S I S T A N T"
@ 16,31 SAY "Beta Version 2.5"
@ 23,26 SAY "Press Any Key To Continue"


*   >>>   WAIT FOR USER'S RESPONSE   <<<

WAIT ""                                        && Wait for keypress

RETURN                                         && Return control to calling progr
am



***   PROGRAM INFORMATION SCREEN   ******************************
*                                                                *
*   This screen describes the program and allows the user to    *
*   exit the program if desired.                                 *
*                                                                *
******************************************************************

PROCEDURE Info_Scr                             && Labels this block of

CLEAR                                          && Erases the screen
SET COLOR TO G/B                               && Insures colors are set properly


*   >>>   CREATE THE SCREEN   <<<

TEXT                                           && Following is sent to the screen

                    The Automated Vendor Selection Assistant

                 selects the vendor(s) who have competitively bid

                          on the item of interest.



                    To proceed, you must know the item's NSN

```

```
                          and the quantity required.
```

```
                        Do you wish to continue? <Y/N>
ENDTEXT                                          && End text sent to the screen
@ 0,0 TO 23,79 DOUBLE                            && Draws box around the screen


*   >>>   GET THE USER'S INPUT   <<<

SET COLOR TO B/B                                 && Hide prompt
SET INTENSITY OFF                                && Hide prompt
STORE 'Y' TO mchoice                             && Make 'Yes' the default
@ 24,55 GET mchoice PICTURE "Y"                  && Accept only <Y> or <N>
READ                                             && Activate the GET
SET INTENSITY ON                                 && Enable highlighted prompt
SET COLOR TO G/B                                 && Restore screen to normal
CLEAR                                            && Clear the screen

RETURN                                           && Return control to calling progr
am



***   INPUT SCREEN   ********************************************
*                                                              *
*   This section prompts the user for the NSN, quantity desired,*
*   and set-aside information.  The inputs are validated and   *
*   returned to the master program via the memory variables    *
*   'MNSN', 'MQUANTITY', and 'MSETASIDE' respectively.  UI,     *
*   (Unit of Issue), is a field from the VENDOR>DBF.  This     *
*   screen may be terminated before entering quantity by       *
*   pressing <ESC><ESC>.                                       *
*                                                              *
*****************************************************************

PROCEDURE InputScr                               && Labels this block of code


*   >>>   DECLARE LOCAL VARIABLES   <<<
```

```
143
144 STORE .F. TO mvalid                          && True when NSN is valid
145 STORE 0 TO mtime                             && Current system time
146 STORE 0 TO mstop                             && Stop time for delay
147 STORE 0 TO mcurrent                          && Current time
148 STORE 0 TO mquantity                         && Amount of product requested
149
150
151 *   >>>   PREPARE THE WORK AREA   <<<
152
153 CLEAR                                        && Erases the screen
154 SELECT A                                     && Activate primary work area
155 USE nsn INDEX n_nsn                          && Used to validate the NSN
156 SELECT B                                     && Activate alternate work area
157 USE vendor INDEX v_c_mil                     && Used to obtain Unit Of Issue
158 SELECT A                                     && Activate primary work area
159 SET RELATION TO cage+mil_spec INTO vendor    && Link VENDOR.DBF with PRICE.DBF
160 SET ESCAPE ON                                && Enable the <ESC> key
161 ON ESCAPE DO return                          && Returns control to calling prog
162 ram
163
164 *   >>>   CREATE THE SCREEN   <<<
165
166 @ 23,20 SAY "Press <ESC><ESC> to Quit the Assistant"
167 @ 0,0 TO 14,79                               && Draw Box
168 @ 14,26 SAY "(Press <CR> when complete)"     && Print on screen
169
170
171 *   >>>   ENTER & VALIDATE THE NSN   <<<
172
173 STORE .F. TO mvalid                          && Set program control flag
174 DO WHILE .NOT. mvalid                        && Do until NSN is correct
175     @ 2,19 SAY "Enter the NSN of the item to be procured"
176     @ 5,31 GET mnsn PICTURE "9999-99-999-9999"  && Enter the NSN
177     READ                                     && Activate GET command
178     SEEK mnsn                                && Search for NSN in PRICE.DBF
179     IF FOUND()                               && If NSN is valid
180         STORE .T. TO mvalid                  && Set the control flag
181     ELSE                                     && If NSN is not valid
182
183 *       >>>   DISPLAY WARNING   <<<
184
185         ? CHR(7)                             && Ring the bell
186         SET COLOR TO R*/B                    && Blinking red
187         @ 9,28 SAY "This NSN is not on file" && Print on screen
188         SET COLOR TO G/B                     && Return screen to normal color
189
```

```
*       >>>   TIMER LOOP   <<<

        STORE TIME() TO mtime                   && Current system time
        mstop = VAL(SUBSTR(mtime,1,2))*3600+;
        VAL(SUBSTR(mtime,4,2))*60+;
        VAL(SUBSTR(mtime,7,2))+5                && Time + 5 seconds
        DO WHILE mcurrent < mstop               && Repeat loop until stop time
            STORE TIME() TO mtime               && Check current time
            mcurrent = VAL(SUBSTR(mtime,1,2))*3600+;
            VAL(SUBSTR(mtime,4,2))*60+;
            VAL(SUBSTR(mtime,7,2))
        ENDDO                                   && End timing loop
        @ 9,5 CLEAR TO 9,75                     && Remove blinking message
    ENDIF                                       && End warning routine
ENDDO                                           && End input NSN routine



*   >>>   CANCEL ESCAPE KEY   <<<

@ 23,0 CLEAR                                    && Remove <ESC> message
ON ESCAPE                                       && Deactivate on escape
SET ESCAPE OFF                                  && Disable escape key



*   >>>   ENTER & VALIDATE THE QUANTITY   <<<

@ 23,29 SAY 'Enter <0><CR> To Quit'             && Display message on screen
@ 9,26 SAY "Enter the quantity required"
@ 12,42 SAY vendor->ui + "."                    && Unit of issue
@ 12,36 GET mquantity PICTURE '@Z 99999'        && Get quantity
READ                                            && Activate GET command

IF mquantity = 0                                && Check for 'Quit' input
    STORE 'Q' TO mchoice                        && Set memory variable
    RETURN                                      && Return to calling program
ENDIF

@ 23,0 CLEAR                                    && Remove 'Quit' message



*   >>>   ENTER THE RDD DATE   <<<

@ 18,26 SAY 'What is the RDD date?'             && Prompt for the RDD date
STORE .F. TO mvalid                             && Reset mValid flag
DO WHILE .NOT. mvalid
```

```
    STORE .T. TO mvalid                          && Set mValid flag
    @ 18,48 GET mrdd PICTURE '@Z 99999'          && Get the date from the user
    READ                                         && Activate the GET command

*   >>>  VALIDATE THE ENTRY   <<<

    IF INT(mrdd/1000) < (myear-(INT(myear/100)*100))-1
        STORE 0 TO mrdd                          && Clear the mRDD variable
        STORE .F. TO mvalid                      && Reset the mValid flag
    ENDIF
    IF mrdd-((INT(mrdd/1000))*1000) > 366 .OR. mrdd-((INT(mrdd/1000))*1000) < 1
        STORE 0 TO mrdd                          && Clear the mRDD variable
        STORE .F. TO mvalid                      && Reset mValid flag
    ENDIF
ENDDO

*   >>>   ENTER & VALIDATE SET-ASIDE INFO   <<<

SET CONFIRM OFF                                  && Enable auto advance
STORE ' ' TO msetaside                           && Reset variable
@ 18,10 SAY "Is this procurement Set-Aside for small ;
business? <Y/N/?>"                               && Print prompt on screen
DO WHILE msetaside <> "Y" .AND. msetaside <> "N"
    @ 18,69 GET msetaside PICTURE "!"            && Convert input to
    READ                                         && Activate GET command
    msetaside = IIF(msetaside = "?","N",msetaside)
ENDDO

SET INTENSITY OFF                                && Disable highlighted

RETURN                                           && Return control to calling progr
am



***   SELECTING VENDOR SCREEN   ********************************
*                                                               *
*   This screen alerts the user to the fact that the system is  *
*   in the process of selecting vendors from the database.  *
*                                                               *
*****************************************************************

PROCEDURE SelctScr                               && Label this block of code

CLEAR                                            && Clear the screen

@ 6,27 TO 10,49 DOUBLE                           && Draw box
@ 8,30 SAY 'Selecting Vendors'                   && Print message
```

```
285
286 RETURN                                          && Return control to calling progr
286 am
287
288
289
290 ***   NO QUALIFIED VENDER SCREEN   ******************************
291 *                                                                *
292 *   This screen alerts the user of the condition in which no    *
293 *   qualified vendors exist.                                     *
294 *                                                                *
295 ******************************************************************
296
297 PROCEDURE NoVenScr                              && Label this block of code
298
299
300 *   >>>   INFORM USER   <<<
301
302 CLEAR                                           && Clear the screen
303 ? CHR(7)                                        && Ring the bell
304 SET COLOR TO R+/B                               && Set color to blinking red
305 @ 5,26 TO 13,53 DOUBLE                          && Draw box
306
307 SET COLOR TO G/B                                && Return color to normal
308 @ 7,30 SAY 'No qualified vendors'               && Print message on screen
309 @ 9,30 SAY 'are on file matching'
310 @ 11,31 SAY 'your requirements.'
311 @ 14,27 SAY 'Press Any Key To Continue'
312
313
314 *   >>>   WAIT FOR USER'S INPUT   <<<
315
316 WAIT ""                                         && Wait for user to acknowledge
317
318 STORE .F. TO mnew_nsn                           && Set program control flag
319
320 RETURN                                          && Return control to calling progr
320 am
321
322
323
324 ***   PRICE SCREEN   ********************************************
325 *                                                                *
326 *   This screen displays the vendor(s) and their price(s) for   *
327 *   the item requested by the user.  It is used for both unit   *
328 *   pricing as well as extended pricing based on the mUNIT_PR   *
329 *   flag.                                                        *
330 *                                                                *
```

```
331 ****************************************************************
332
333 PROCEDURE PriceScr                          && Labels this block of code
334
335 CLEAR                                       && Clear the screen
336 SET COLOR TO G/B                            && Set colors to standard values
337
338
339 *   >>>   DRAW GRID   <<<
340
341 @ 4,3 TO 22,75 DOUBLE                       && Draw boxes and lines
342 @ 6,3 TO 6,75 DOUBLE
343 @ 10,3 TO 10,75
344 @ 14,3 TO 14,75
345 @ 18,4 TO 18,75 DOUBLE
346 @ 4,9 TO 18,9
347 @ 4,20 TO 18,20
348 @ 4,31 TO 18,31
349 @ 4,42 TO 22,42
350 @ 4,53 TO 18,53
351 @ 4,64 TO 18,64
352
353 @ 4,9 SAY CHR(209)                          && Special characters at intersect
353 ions
354 @ 4,20 SAY CHR(209)
355 @ 4,31 SAY CHR(209)
356 @ 4,42 SAY CHR(209)
357 @ 4,53 SAY CHR(209)
358 @ 4,64 SAY CHR(209)
359
360 @ 18,9 SAY CHR(207)
361 @ 18,20 SAY CHR(207)
362 @ 18,31 SAY CHR(207)
363 @ 18,42 SAY CHR(216)
364 @ 22,42 SAY CHR(207)
365 @ 18,53 SAY CHR(207)
366 @ 18,64 SAY CHR(207)
367
368 @ 6,3 SAY CHR(204)
369 @ 10,3 SAY CHR(199)
370 @ 14,3 SAY CHR(199)
371 @ 18,3 SAY CHR(204)
372
373 @ 6,75 SAY CHR(185)
374 @ 10,75 SAY CHR(182)
375 @ 14,75 SAY CHR(182)
376 @ 18,75 SAY CHR(185)
377
```

```
@ 6,9 SAY CHR(216)
@ 6,20 SAY CHR(216)
@ 6,31 SAY CHR(216)
@ 6,42 SAY CHR(216)
@ 6,53 SAY CHR(216)
@ 6,64 SAY CHR(216)

@ 10,9 SAY CHR(197)
@ 10,20 SAY CHR(197)
@ 10,31 SAY CHR(197)
@ 10,42 SAY CHR(197)
@ 10,53 SAY CHR(197)
@ 10,64 SAY CHR(197)

@ 14,9 SAY CHR(197)
@ 14,20 SAY CHR(197)
@ 14,31 SAY CHR(197)
@ 14,42 SAY CHR(197)
@ 14,53 SAY CHR(197)
@ 14,64 SAY CHR(197)

IF mhist_date <> ' '
    @ 0,49 TO 1,49 DOUBLE
    @ 2,49 TO 2,74 DOUBLE
    @ 0,75 TO 1,75 DOUBLE

    @ 2,49 SAY CHR(200)
    @ 2,75 SAY CHR(188)
ENDIF

*   >>>   DISPLAY CONSTANT ITEMS   <<<

IF .NOT. munit_pr                                  && List user options
    @ 23,3 SAY '< > Unit Pricing'
ELSE
    @ 23,3 SAY '< > Extended Pricing'
ENDIF

@ 24,3 SAY '< > Vendor Information'
@ 23,30 SAY '< > Award Screen'
@ 24,30 SAY '< > Quit'
@ 23,55 SAY '< > CDCF Vendor Detail'
@ 24,55 SAY '< > Problem Vendor Detail'


*>>>  FILL IN KEY CODES   <<<
```

```
SET COLOR TO BG+/B                          && Change screen color
IF .NOT. munit_pr                           && Print key codes
    @ 23,4 SAY 'U'
ELSE
    @ 23,4 SAY 'E'
ENDIF
@ 24,4 SAY 'V'
@ 23,31 SAY 'A'
@ 24,31 SAY 'Q'
@ 23,56 SAY 'C'
@ 24,56 SAY 'P'

*>>>  LABEL SCREEN   <<<

SET COLOR TO G/B                            && Change screen color
IF munit_pr                                 && Label screen
    @ 3,0 SAY "Unit Pricing Data For: "
ELSE
    @ 3,0 SAY "Extended Pricing Data For: "
ENDIF
SET COLOR TO BG+/B                          && Change screen color
@ 3,$ SAY mnsn                              && Print NSN

*>>>  PRINT HISTORY DATA   <<<

IF mhist_date <> ' '                        && If history data on file
    SET COLOR TO G/B                        && Set screen colors
    @ 0,51 SAY 'Last Purchased On '         && Display data
    @ $,$ SAY mhist_date PICTURE 'XXXXX'
    @ $+1,51 SAY 'From '
    @ $,$ SAY mhist_cage
    @ $,$+1 SAY 'For $'
    @ $,$ SAY mhist_pr PICTURE '@B 9999.99'
ENDIF

*>>>  PRINT COLOR CODES   <<<

SET COLOR TO G/B                            && Change screen color
@ 19,5 SAY 'VENDOR:'                        && Print legend
@ 19,44 SAY 'PRICE:'
SET COLOR TO R/B
@ 19,12 SAY CHR(219)+' Problem Vendor Info'
SET COLOR TO GR+/B
@ 19,50 SAY CHR(219)+' Price May Be To Low'
SET COLOR TO RB/B
@ 20,12 SAY CHR(219)+' CDCF Vendor Info'
SET COLOR TO G+/B
@ 20,50 SAY CHR(219)+' Low Price'
```

```
@ 21,12 SAY CHR(219)+' Quality Vendor'
SET COLOR TO G/B                                && Return to normal color


*   >>>   DISPLAY ORDER QUANTITIES   <<<

@ 5,12 SAY morder1 PICTURE '@Z 99,999'          && Print order quantities
@ 5,23 SAY morder2 PICTURE '@Z 99,999'
@ 5,34 SAY morder3 PICTURE '@Z 99,999'
@ 5,45 SAY morder4 PICTURE '@Z 99,999'
@ 5,56 SAY morder5 PICTURE '@Z 99,999'
@ 5,67 SAY morder6 PICTURE '@Z 99,999'


*   >>>   DISPLAY MEMORY MATRIX   <<<

*>>>   INITIALIZE VARIABLES   <<<

STORE 1 TO mcounter                             && 1 to 9
STORE 1 TO mcount                               && 1 to 3
STORE 7 TO mrow                                 && 7 to 22
STORE 11 TO mcolumn                             && 11 to 66 step 11
STORE 1 TO mcol                                 && 1 to 6
STORE .T. TO mcontinue                          && Program control flag
STORE 'MCAGE1' TO mcage                         && First matrix item to be display
ed

SELECT pr_temp                                  && Activate PR_TEMP.DBF
DO WHILE mcontinue                              && Do for all matrix cages
    LOCATE FOR cage = &mcage                    && Find matrix cage in PR_TEMP.DBF

*   >>>  COLOR CODE VENDORS

    DO CASE                                     && Check flags
    CASE prob
        SET COLOR TO R/B                        && Change displayed color
    CASE cdcf
        SET COLOR TO RB/B
    CASE quality
        SET COLOR TO G+/B
    ENDCASE

    @ mrow,4 SAY &mcage                         && Print cage to screen
    SET COLOR TO G/B                            && Restore color to normal

*   >>>  DISPLAY PRICES   <<<

    DO WHILE mcol < 7                           && Fill the 6 screen columns
```

```
        STORE 'MEXT_'+LTRIM(STR(mcounter))+'_'+LTRIM(STR(mcol)) TO mprice

*       >>>   COLOR CODE PRICES   <<<

        DO CASE                                 && Test price flags
        CASE mlow .AND. &mprice = mlow_price
            SET COLOR TO GR+/B
        CASE &mprice = mlow_price
            SET COLOR TO G+/B
        ENDCASE

        IF munit_pr                             && Display price on screen
            STORE 'MORDER'+LTRIM(STR(mcol)) TO morder
            STORE &mprice/&morder TO mnet_price
            @ mrow,mcolumn-1 SAY mnet_price PICTURE '@Z 9,999.9999'
        ELSE
            @ mrow,mcolumn SAY &mprice PICTURE '@Z 99,999.99'
        ENDIF

        SET COLOR TO G/B

*       >>>   ADVANCE COUNTERS   <<<

        STORE mcolumn + 11 TO mcolumn           && Increment screen position count
er
        STORE mcol + 1 TO mcol                  && Increment column counter
    ENDDO                                       && Finish one row

    STORE 11 TO mcolumn                         && Reset screen position counter
    STORE 1 TO mcol                             && Reset column counter
    STORE mrow + 1 TO mrow                      && Advance screen position counter
    STORE mcounter + 1 TO mcounter              && Advance matrix counter

*   >>>  CHECK FOR GRID LINE   <<<

    IF mcount = 3                               && Don't print on a grid line
        STORE mrow + 1 TO mrow                  && Advance screen position counter
        STORE 1 TO mcount                       && Advance counter
    ELSE
        STORE mcount + 1 TO mcount              && Advance counter
    ENDIF

    STORE 'MCAGE'+LTRIM(STR(mcounter)) TO mcage
    IF &mcage = '      '
        STORE .F. TO mcontinue                  && .F. if all cages displayed
    ENDIF
ENDDO
```

```


*****************************************************************
*                                                               *
*   The following code checks for conditions the buyer should   *
*   be aware of before making an award.                         *
*                                                               *
*****************************************************************

*   >>>   CHECK VENDOR VARIATION   <<<

STORE 0 TO mline                                && Display row counter
IF mvariation                                   && Check for variation problem
    SET COLOR TO R+*/B                          && Display warning
    @ mline,0 SAY '***'
    SET COLOR TO G/B
    @ mline,$+3 SAY 'LOW QUOTE PLUS VARIATION EXCEEDS $'
    @ mline,$ SAY LTRIM(STR(model->up_limit,10,2))
    SET COLOR TO R+*/B
    @ mline,$+3 SAY '***'
    SET COLOR TO G/B
    STORE mline + 1 TO mline                    && Advance row counter
ENDIF


*   >>>   CHECK HISTORICAL DATA   <<<

IF mhistory1                                    && Check high price for history
    SET COLOR TO R+*/B                          && Display warning
    @ mline,0 SAY '***'
    SET COLOR TO G/B
    @ $,$+3 SAY 'UNIT PRICE EXCEEDS HISTORY'
    SET COLOR TO R+*/B
    @ $,$+3 SAY '***'
    STORE mline+1 to mline
ENDIF

IF mhistory2                                    && Check no history on NSN
    SET COLOR TO R+*/B                          && Display warning
    @ mline,0 SAY '***'
    SET COLOR TO G/B
    @ mline,$+3 SAY 'UNIT PRICE OVER $'
    @ mline,$+1 SAY LTRIM(STR(model->history2,10,2))
    @ mline,$+1 SAY 'WITH NO HISTORY'
    SET COLOR TO R+*/B
    @ mline,$+3 SAY '***'
    SET COLOR TO G/B
ENDIF
```

```


*   >>>   GET USER'S RESPONSE   <<<

STORE ' ' TO mchoice                          && Reset user's choice variable
SET COLOR TO B/B
IF munit_pr
    STORE 'U' TO mreturn                      && Flag for user to return to this
 screen
    DO WHILE .NOT. UPPER(mchoice)$'ACEPQV'    && Limit user's response choices
        WAIT '' TO mchoice                    && Get response
    ENDDO
ELSE
    STORE 'E' TO mreturn                      && Flag for user to return to this
 screen
    DO WHILE .NOT. UPPER(mchoice)$'APQUV'     && Limit user's response choices
        WAIT '' TO mchoice                    && Get response
    ENDDO
ENDIF
SET COLOR TO G/B

STORE .F. TO munit_pr                         && Reset program control flag

RETURN                                        && Return to calling program



***   VENDOR SCREEN   ******************************************
*                                                              *
*   This screen displays the vendor data for those qualified   *
*   vendors competing on the item requested by the user.       *
*                                                              *
*****************************************************************

PROCEDURE VendrScr                            && Labels this block of code

CLEAR                                         && Clear the screen
SET COLOR TO G/B                              && Set color to standard value


*   >>>   INSERT TEXT   <<<

@ 2,50 SAY 'D                 S P C Q'        && Place text on the screen
@ 3,50 SAY '[illegible]      N    D   F P R D U'
@ 4,50 SAY 'S      E    E   O E O C A'
@ 5,4 SAY 'CAGE  VENDOR'
@ 5,50 SAY 'C      T    L   B C B S L'
```

```


*   >>>   DRAW GRID   <<<

@ 4,3 TO 4,47 DOUBLE                          && Draw boxes, lines
@ 1,47 TO 18,47 DOUBLE
@ 1,47 TO 1,75 DOUBLE
@ 1,75 TO 18,75 DOUBLE
@ 4,3 TO 18,3 DOUBLE
@ 18,3 TO 18,75 DOUBLE

@ 6,3 TO 6,75 DOUBLE
@ 10,3 TO 10,75
@ 14,3 TO 14,75

@ 4,9 TO 18,9
@ 6,53 TO 18,53
@ 1,59 TO 18,59 DOUBLE
@ 1,65 TO 18,65
@ 1,67 TO 18,67 DOUBLE
@ 1,69 TO 18,69
@ 1,71 TO 18,71
@ 1,73 TO 18,73

@ 4,3 SAY CHR(201)                            && Place special characters at int
ersections
@ 4,9 SAY CHR(209)
@ 4,47 SAY CHR(185)
@ 1,47 SAY CHR(201)
@ 1,59 SAY CHR(203)
@ 1,65 SAY CHR(209)
@ 1,67 SAY CHR(203)
@ 1,69 SAY CHR(209)
@ 1,71 SAY CHR(209)
@ 1,73 SAY CHR(209)
@ 1,75 SAY CHR(187)

@ 18,3 SAY CHR(200)
@ 18,9 SAY CHR(207)
@ 18,47 SAY CHR(202)
@ 18,53 SAY CHR(207)
@ 18,59 SAY CHR(202)
@ 18,65 SAY CHR(207)
@ 18,67 SAY CHR(202)
@ 18,69 SAY CHR(207)
@ 18,71 SAY CHR(207)
@ 18,73 SAY CHR(207)
@ 18,75 SAY CHR(188)
```

```
709
710  @ 6,3 SAY CHR(204)
711  @ 10,3 SAY CHR(199)
712  @ 14,3 SAY CHR(199)
713
714  @ 6,75 SAY CHR(185)
715  @ 10,75 SAY CHR(182)
716  @ 14,75 SAY CHR(182)
717
718  @ 6,9 SAY CHR(216)
719  @ 6,47 SAY CHR(206)
720  @ 6,53 SAY CHR(209)
721  @ 6,59 SAY CHR(206)
722  @ 6,65 SAY CHR(216)
723  @ 6,67 SAY CHR(206)
724  @ 6,69 SAY CHR(216)
725  @ 6,71 SAY CHR(216)
726  @ 6,73 SAY CHR(216)
727
728  @ 10,9 SAY CHR(197)
729  @ 10,47 SAY CHR(215)
730  @ 10,53 SAY CHR(197)
731  @ 10,59 SAY CHR(215)
732  @ 10,65 SAY CHR(197)
733  @ 10,67 SAY CHR(215)
734  @ 10,69 SAY CHR(197)
735  @ 10,71 SAY CHR(197)
736  @ 10,73 SAY CHR(197)
737
738  @ 14,9 SAY CHR(197)
739  @ 14,47 SAY CHR(215)
740  @ 14,53 SAY CHR(197)
741  @ 14,59 SAY CHR(215)
742  @ 14,65 SAY CHR(197)
743  @ 14,67 SAY CHR(215)
744  @ 14,69 SAY CHR(197)
745  @ 14,71 SAY CHR(197)
746  @ 14,73 SAY CHR(197)
747
748
749  *   >>>   DISPLAY OPTIONS   <<<
750
751  @ 22,3 SAY "User's Options:"                    && Display user's choices
752  @ 23,3 SAY '< > Unit Pricing'
753  @ 24,3 SAY '< > Extended Pricing'
754  @ 23,30 SAY '< > Award Screen'
755  @ 24,30 SAY '< > Quit'
756  @ 23,55 SAY '< > CDCF Vendor Detail'
```

```
@ 24,55 SAY '< > Problem Vendor Detail'

SET COLOR TO BG+/B                          && Display key codes
@ 23,4 SAY 'U'
@ 24,4 SAY 'E'
@ 23,31 SAY 'A'
@ 24,31 SAY 'Q'
@ 23,56 SAY 'C'
@ 24,56 SAY 'P'

SET COLOR TO G/B                            && Return color to normal
@ 2,0 SAY "Vendor Data For: "               && Title screen
SET COLOR TO BG+/B                          && Change color
@ 2,$ SAY mnsn                              && Print NSN


*   >>>   FILL SCREEN   <<<


*>>>  INITIALIZE COUNTERS   <<<

STORE 1 TO mcounter
STORE 1 TO mcount
STORE 7 TO mrow

*>>>  PLACE DATA ON SCREEN   <<<

SELECT pr_temp                              && Activate PR_TEMP.DBF
STORE 'MCAGE'+LTRIM(STR(mcounter)) TO mcage && Create cage pointer variable
STORE .T. TO mcontinue                      && Set program control flag
SET COLOR TO G/B                            && Insure normal screen color
DO WHILE mcontinue                          && Print data
    LOCATE FOR cage = &mcage                && Find mCAGE1 in the database
        DO CASE                             && Check for highlights
    CASE prob
        SET COLOR TO R/B
    CASE cdcf
        SET COLOR TO RB/B
    CASE quality
        SET COLOR TO G+/B
    ENDCASE
        @ mrow,4 SAY &mcage                 && Print cage
    SET COLOR TO G/B                        && Reset color to normal
    @ mrow,10 SAY vendor->name              && Print vendor data
    @ mrow,48 SAY vendor->disc PICTURE '@Z 99.9%'
    @ mrow,54 SAY vendor->days PICTURE '@Z 99'
    @ mrow,56 SAY '/'
    @ mrow,57 SAY vendor->net PICTURE '@Z 99'
```

```
    STORE mj_date + vendor->delivery + model->alt TO mdelivery
    STORE VAL(RIGHT(STR(myear),2)) TO myear

    IF (myear*1000)+mdelivery > mrdd            && Compare delivery date to RDD
        SET COLOR TO R/B                        && If unable to meet
    ENDIF                                       && Change colors
    DO CASE                                     && Calculate delivery date
    CASE .NOT. mleap_yr .AND. mdelivery <= 365
        @ mrow,60 SAY myear PICTURE '@Z 99'
        IF mdelivery < 100
            @ mrow,62 SAY '0'
            @ mrow,63 SAY mdelivery PICTURE  '@Z 99'
        ELSE
            @ mrow,62 SAY mdelivery PICTURE '@Z 999'
        ENDIF
    CASE .NOT. mleap_yr .AND. mdelivery > 365
        @ mrow,60 SAY myear+1 PICTURE '@Z 99'
        IF mdelivery - 365 < 100
            @ mrow,62 SAY '0'
            @ mrow,63 SAY mdelivery - 365 PICTURE  '@Z 99'
        ELSE
            @ mrow,62 SAY mdelivery - 365 PICTURE '@Z 999'
        ENDIF
    CASE mleap_yr .AND. mdelivery <= 366
        @ mrow,60 SAY myear PICTURE '@Z 99'
        IF mdelivery < 100
            @ mrow,62 SAY '0'
            @ mrow,63 SAY mdelivery PICTURE  '@Z 99'
        ELSE
            @ mrow,62 SAY mdelivery PICTURE '@Z 999'
        ENDIF
    CASE mleap_yr .AND. mdelivery > 366
        @ mrow,60 SAY myear+1 PICTURE '@Z 99'
        IF mdelivery - 366 < 100
            @ mrow,62 SAY '0'
            @ mrow,63 SAY mdelivery - 366 PICTURE  '@Z 99'
        ELSE
            @ mrow,62 SAY mdelivery - 366 PICTURE '@Z 999'
        ENDIF
    ENDCASE

    SET COLOR TO G/B

*   >>>  PRINT FLAGS   <<<

    @ mrow,66 SAY vendor->fob PICTURE '!'       && Display F.O.B.
    IF vendor->size_code <> 'A'                 && Small/large vendor flag
        @ mrow,68 SAY 'r'
```

```
    ENDIF
    IF prob                                   && Problem vendor flag
        @ mrow,70 SAY 'X'
    ENDIF
    IF cdcf                                   && CDCF Flag
        @ mrow,72 SAY 'X'
    ENDIF
    IF quality                                && Quality flag
        @ mrow,74 SAY 'X'
    ENDIF

*   >>>  ADVANCE COUNTERS   <<<

    STORE mcounter + 1 TO mcounter            && Advance mCAGE# counter
    STORE mrow + 1 TO mrow                    && Advance print row counter

*   >>>  CHECK FOR GRID LINES   <<<

    IF mcount = 3                             && If three lines have been printe
d
        STORE mrow + 1 TO mrow                && Advance row counter
        STORE 1 TO mcount                     && Reset counter
    ELSE
        STORE mcount + 1 TO mcount            && Advance counter
    ENDIF

    STORE 'MCAGE'+LTRIM(STR(mcounter)) TO mcage
    IF &mcage = '    '                        && Check for last cage
        STORE .F. TO mcontinue                && .F. if all have been printed
    ENDIF
ENDDO


*   >>>   WAIT FOR USER RESPONSE   <<<

STORE 'V' TO mreturn                          && Flag for user to return to this
 screen
STORE ' ' TO mchoice                          && Reset user's choice
SET COLOR TO B/B                              && Hide response
DO WHILE .NOT. UPPER(mchoice)$'ACEPQU'        && Limit user's responses
    WAIT '' TO mchoice                        && Get user's response
ENDDO
SET COLOR TO G/B                              && Return screen to normal

RETURN                                        && Return control to calling progr
am
```

```
898
899 ***   ANALYZING VENDOR SCREEN   ********************************
900 *                                                               *
901 *   This screen alerts the user to the fact that the system is  *
902 *   in the process of analyzing vendors in the temporary db.    *
903 *                                                               *
904 *****************************************************************
905
906 PROCEDURE AnalzScr                          && Labels this block of code
907
908 CLEAR                                       && Clear the screen
909
910 @ 6,27 TO 10,49 DOUBLE                      && Draw box
911 @ 8,29 SAY 'Analyzing Vendor(s)'            && Print message
912
913 RETURN                                      && Return control to calling progr
913 am
914
915
916
917 ***   INITIALIZING SYSTEM SCREEN   ******************************
918 *                                                               *
919 *   This screen alerts the user to the fact that the system is  *
920 *   in the process of initializing the system.                  *
921 *                                                               *
922 *****************************************************************
923
924 PROCEDURE InitlScr                           && Labels this block of code
925
926 CLEAR                                       && Clear the screen
927
928 @ 6,26 TO 10,51 DOUBLE                      && Draw box
929 @ 8,28 SAY 'Initializing The System'        && Display message
930
931 RETURN                                      && Return control to calling progr
931 am
932
933
934
935 ***   PROBLEM VENDOR SCREEN   ***********************************
936 *                                                               *
937 *   This screen displays the information on file in DCRL data   *
938 *   base for a selected vendor.                                 *
939 *                                                               *
940 *****************************************************************
941
942 PROCEDURE ProbmScr                          && Labels this block of code
943
```

```
@ 23,0 CLEAR                                    && Clear the screen
SET COLOR TO G/B                                && Set color to normal

@ 23,26 SAY 'Display Details For:'              && Display user instructions
@ 24,10 SAY 'Press <SPACE BAR> For Next Choice - Any Other Key To Accept'


*   >>>   DISPLAY CAGE CODES   <<<

SELECT pr_temp                                  && Activate PR_TEMP.DBF
LOCATE FOR prob                                 && Initialize the locate command
GOTO TOP                                        && Return to the first record
SET COLOR TO BG/B                               && Change screen color
STORE ' ' TO mchoice                            && Clear user's choice

DO WHILE mchoice = ' '                          && Display vendors with problem f
ag
    IF EOF()                                    && If BOF,
        GOTO TOP                                && Go to Top of File
    ENDIF
    CONTINUE                                    && Look for next problem vendor
    IF EOF()                                    && If all problem vendors displaye
d
        @ 23,47 SAY 'No One'                    && Print 'No One'
    ELSE
        @ 23,47 SAY cage + ' '
    ENDIF
    WAIT '' TO mchoice                          && Get user's input
ENDDO


*   >>>   DISPLAY USER'S CHOICE   <<<

IF EOF()                                        && User picked 'No One'
    STORE mreturn TO mchoice                    && Prepare to return to previous s
creen
ELSE                                            && Otherwise,
    CLEAR                                       && Clear the screen
    SET COLOR TO G/B                            && Set color to normal

*   >>>   DRAW THE GRID   <<<

    @ 4,3 TO 19,77 DOUBLE                       && Draw boxes, lines
    @ 6,4 TO 6,76 DOUBLE
    @ 13,4 TO 13,76
    @ 7,36 TO 12,36

    SELECT C                                    && Use alternate work area
```

```
989     USE dcrl INDEX dcr_cage                    && Activate DCRL.DBF
990     SELECT D                                   && Use alternate work area
991     USE dcrlcode                               && Activate DCRLCODE.DBF
992     SELECT DCRL                                && Activate DCRL
993     SEEK pr_temp->cage                         && Look for cage
994
995     @ 5,37 SAY cage                            && Display vendor address
996     @ 7,4 SAY LEFT(name1,32)
997     @ $+1,4 SAY LEFT(name2,32)
998     @ $+1,4 SAY LEFT(name3,32)
999     @ $+1,4 SAY LEFT(name4,32)
1000
1001    SELECT dcrlcode                            && Display problem codes
1002    @ 7,38 SAY dcrl->date1 + ' ' + dcrl->category1
1003    LOCATE FOR code = LEFT(dcrl->category1,1)
1004    IF FOUND()
1005        @ $,50 SAY title
1006    ENDIF
1007    @ $+1,38 SAY dcrl->date2 + ' ' + dcrl->category2
1008    LOCATE FOR code = LEFT(dcrl->category2,1)
1009    IF FOUND()
1010        @ $,50 SAY title
1011    ENDIF
1012    @ $+1,38 SAY dcrl->date3 + ' ' + dcrl->category3
1013    LOCATE FOR code = LEFT(dcrl->category3,1)
1014    IF FOUND()
1015        @ $,50 SAY title
1016    ENDIF
1017    @ $+1,38 SAY dcrl->date4 + ' ' + dcrl->category4
1018    LOCATE FOR code = LEFT(dcrl->category4,1)
1019    IF FOUND()
1020        @ $,50 SAY title
1021    ENDIF
1022    @ $+1,38 SAY dcrl->date5 + ' ' + dcrl->category5
1023    LOCATE FOR code = LEFT(dcrl->category5,1)
1024    IF FOUND()
1025        @ $,50 SAY title
1026    ENDIF
1027    @ $+1,38 SAY dcrl->date6 + ' ' + dcrl->category6
1028    LOCATE FOR code = LEFT(dcrl->category6,1)
1029    IF FOUND()
1030        @ $,50 SAY title
1031    ENDIF
1032
1033    SELECT dcrl                                && Print restriction verbiage
1034    @ 14,14 SAY restrict1
1035    @ $+1,14 SAY restrict2
1036    @ $+1,14 SAY restrict3
```

```
    @ $+1,14 SAY restrict4
    @ $+1,14 SAY restrict5

    STORE mreturn TO mchoice                    && Return user to previous screen
    @ 23,0 SAY ' '                              && Position cursor
    WAIT                                        && Wait for user to respond
ENDIF

SELECT D                                        && Return MODEL.DBF to area D
USE MODEL                                       && Activate MODEL.DBF

RETURN                                          && Return control to calling progr
am



***   CDCF VENDOR SCREEN   ************************************
*                                                              *
*   This screen display the information on file in CDCF data   *
*   base for a selected vendor.                                *
*                                                              *
****************************************************************

PROCEDURE CdcfScr                               && Labels this block of code

@ 23,0 CLEAR                                    && Clear the screen
SET COLOR TO G/B                                && Set color to normal

@ 23,26 SAY 'Display Details For:'              && Display user instructions
@ 24,10 SAY 'Press <SPACE BAR> For Next Choice - Any Other Key To Accept'


*   >>>   DISPLAY CAGE CODES   <<<

SELECT pr_temp                                  && Activate PR_TEMP.DBF
LOCATE FOR cdcf                                 && Initialize the locate command
GOTO TOP                                        && Return to the first record
SET COLOR TO BG/B                               && Change screen color
STORE ' ' TO mchoice                            && Clear user's choice

DO WHILE mchoice = ' '                          && Display vendors with problem fl
ag
    IF EOF()                                    && If BOF,
        GOTO TOP                                && Go to Top of File
    ENDIF
    CONTINUE                                    && Look for next problem vendor
    IF EOF()                                    && If all problem vendors displaye
d
```

```
        @ 23,47 SAY 'No One'                    && Print 'No One'
    ELSE
        @ 23,47 SAY cage + ' '
    ENDIF
    WAIT '' TO mchoice                          && Get user's input
ENDDO


*   >>>   DISPLAY USER'S CHOICE   <<<

IF EOF()                                        && User picked 'No One'
    STORE mreturn TO mchoice                    && Prepare to return to previous s
creen
ELSE                                            && Otherwise,
    CLEAR                                       && Clear the screen
    SET COLOR TO G/B                            && Set color to normal

*   >>>   DRAW THE GRID   <<<

    @ 4,3 TO 13,77 DOUBLE                       && Draw boxes, lines
    @ 6,4 TO 6,76 DOUBLE

*   >>>   PRINT THE CONSTANTS   <<<

    SET COLOR TO G+/B                           && Change screen color
    @ 8,5 SAY 'DISC -->'
    @ $+1,5 SAY 'CAUSE ->'
    @ $+1,5 SAY 'DISP -->'
    @ $+1,5 SAY 'CORR -->'

*   >>>   PRINT THE DATA   <<<

    SET COLOR TO W/B                            && Change screen color
    @ 5,37 SAY cage                             && Display cage
    SELECT C                                    && Use alternate work area
    USE cdcf INDEX cdcf_n_c                     && Activate CDCF.DBF
    SEEK mnsn+pr_temp->cage                     && Look for cage
    STORE .T. TO mflag
    DO WHILE .NOT. EOF() .AND. (nsn = mnsn .AND. cage = pr_temp->cage)
        @ 8,14 SAY disc_code
        @ $,$+1 SAY LEFT(disc,60)
        @ $+1,14 SAY cause_code
        @ $,$+1 SAY LEFT(cause,60)
        @ $+1,14 SAY disp_code
        @ $,$+1 SAY LEFT(disp,60)
        @ $+1,14 SAY corr_code
        @ $,$+1 SAY LEFT(corr,60)
        @ 23,0 SAY ' '                          && Position curser
```

```
1129        WAIT                                && Wait on key press
1130        SKIP                                && Find next occurrence
1131    ENDDO
1132 ENDIF
1133
1134 RETURN                                     && Return to calling program
1135
1136
1137
1138 ***   AWARD SCREEN   ******************************************
1139 *                                                             *
1140 *   After the buyer makes a decision as to who will receive the *
1141 *   contract, this screen will show the information needed to   *
1142 *   complete the resulting paperwork.                          *
1143 *                                                             *
1144
1145 ***************************************************************
1146
1147 PROCEDURE AwardScr                         && Labels this block of code
1148
1149 @ 23,0 CLEAR                               && Clear the screen
1150 SET COLOR TO G/B                           && Set normal screen colors
1151
1152 @ 23,26 SAY 'Display Details For:'         && Display user instructions
1153 @ 24,10 SAY 'Press <SPACE BAR> For Next Choice - Any Other Key To Accept'
1154
1155 *   >>>   DISPLAY CAGE CODES   <<<
1156
1157 SELECT pr_temp                             && Activate PR_TEMP.DBF
1158 GOTO TOP                                   && Set pointer to first record
1159 SET COLOR TO BG/B                          && Change display color
1160 STORE ' ' TO mchoice                       && Reset user's selection
1161
1162 DO WHILE mchoice = ' '                     && Display cage codes
1163     IF .NOT. EOF()
1164         @ 23,47 SAY cage + ' '
1165     ELSE                                   && IF End Of File
1166         @ 23,47 SAY 'No One'               && Display 'No One'
1167     ENDIF
1168
1169 *   >>>  GET USER'S SELECTION   <<<
1170
1171     WAIT '' TO mchoice                     && Get user's choice
1172     IF mchoice = ' '                       && If space bar
1173         IF .NOT. EOF()
1174             SKIP                           && Move pointer to next record
1175         ELSE
1176             GOTO TOP                       && Position pointer at record one
```

```
1177         ENDIF
1178     ENDIF
1179 ENDDO                                          && End of display cage codes
1180
1181
1182 *   >>>   DISPLAY USER'S CHOICE   <<<
1183
1184 IF EOF()
1185     STORE mreturn TO mchoice                   && Prepare to return to last scree
1185 n
1186 ELSE
1187
1188     CLEAR                                      && Clear screen
1189     SET COLOR TO G/B                           && Set color to normal
1190
1191
1192 *   >>>   DRAW GRIDS   <<<
1193
1194
1195     @ 2,0 TO 21,79 DOUBLE                      && Draw boxes/lines
1196     @ 8,1 TO 8,78 DOUBLE
1197     @ 16,1 TO 16,78 DOUBLE
1198     @ 18,1 TO 18,78
1199     @ 3,39 TO 7,39
1200     @ 17,13 TO 20,13
1201     @ 17,24 TO 20,24
1202     @ 17,35 TO 20,35
1203     @ 17,46 TO 20,46
1204     @ 17,57 TO 20,57
1205     @ 17,68 TO 20,68
1206
1207     @ 8,0 SAY CHR(204)                         && Place special characters at int
1207 ersections
1208     @ 16,0 SAY CHR(204)
1209     @ 18,0 SAY CHR(199)
1210
1211     @ 8,79 SAY CHR(185)
1212     @ 16,79 SAY CHR(185)
1213     @ 18,79 SAY CHR(182)
1214
1215     @ 2,39 SAY CHR(209)
1216     @ 8,39 SAY CHR(207)
1217
1218     @ 16,13 SAY CHR(209)
1219     @ 16,24 SAY CHR(209)
1220     @ 16,35 SAY CHR(209)
1221     @ 16,46 SAY CHR(209)
1222     @ 16,57 SAY CHR(209)
```

```
    @ 16,68 SAY CHR(209)

    @ 21,13 SAY CHR(207)
    @ 21,24 SAY CHR(207)
    @ 21,35 SAY CHR(207)
    @ 21,46 SAY CHR(207)
    @ 21,57 SAY CHR(207)
    @ 21,68 SAY CHR(207)

    @ 18,13 SAY CHR(197)
    @ 18,24 SAY CHR(197)
    @ 18,35 SAY CHR(197)
    @ 18,46 SAY CHR(197)
    @ 18,57 SAY CHR(197)
    @ 18,68 SAY CHR(197)


*   >>>   FILL IN CONSTANTS   <<<

    @ 0,0 SAY 'Award Information For:'          && Print static text

    @ 3,2 SAY 'Vendor:'
    @ 3,41 SAY 'Remit To:'
    @ 10,2 SAY 'Cage:'
    @ 10,31 SAY 'State Code:'
    @ 10,64 SAY 'Source Type:'
    @ 12,2 SAY 'Discount:      % In    Days'
    @ 12,59 SAY 'Variance: +  % -  %'
    @ 14,2 SAY 'Delivery Time:     Days'
    @ 14,38 SAY 'FOB:'
    @ 14,66 SAY 'RFCC Code:'
    @ 19,2 SAY 'Unit Price'
    @ 20,2 SAY 'Ext. Price'

    @ 23,25 SAY 'Press <P> For Previous Screen'
    @ 24,26 SAY 'Any Other Key When Finished'

    SET COLOR TO W/B                            && Change screen colors
    @ 0,23 SAY mnsn                             && Print NSN


*   >>>   FILL IN VENDOR SPECIFIC DATA   <<<

    SELECT VENDOR                               && Activate VENDOR.DBF

    @ 4,4 SAY address1                          && Relation was set from
    @ 5,4 SAY address2                          && PR_TEMP
    @ 6,4 SAY address3
```

```
1271     @ 7,4 SAY address4
1272
1273     @ 4,43 SAY remit1                          && Print billing address
1274     @ 5,43 SAY remit2
1275     @ 6,43 SAY remit3
1276     @ 7,43 SAY remit4
1277
1278     @ 10,8 SAY cage                            && Print other vendor information
1279     @ 10,43 SAY state
1280     @ 10,77 SAY size_code
1281     @ 12,12 SAY disc PICTURE '9.999'
1282     @ 12,22 SAY days
1283     @ 12,70 SAY qty_var_p
1284     @ 12,75 SAY qty_var_m
1285     @ 14,17 SAY delivery
1286     @ 14,43 SAY fob
1287     @ 14,77 SAY rfcc
1288
1289
1290 *   >>>   FILL IN PRICING DATA   <<<
1291
1292     SET COLOR TO G/B                           && Set standard colors
1293
1294     @ 17,16 SAY morder1 PICTURE '@Z 99,999'    && Print order quantities
1295     @ 17,27 SAY morder2 PICTURE '@Z 99,999'
1296     @ 17,38 SAY morder3 PICTURE '@Z 99,999'
1297     @ 17,49 SAY morder4 PICTURE '@Z 99,999'
1298     @ 17,60 SAY morder5 PICTURE '@Z 99,999'
1299     @ 17,71 SAY morder6 PICTURE '@Z 99,999'
1300
1301 *   >>>  SEARCH FOR SELECTED VENDOR IN PRICING MATRIX   <<<
1302
1303     STORE 1 TO mcounter                        && Initialize counter
1304     STORE mcage1 TO mcage                      && Store first cage in matrix
1305     IF TYPE('mcage') = 'N'                     && If it is all numeric,
1306         STORE STR(mcage,5) TO mcage            && Convert to string
1307     ENDIF
1308
1309     DO WHILE cage <> mcage                     && Search for proper cage in matri
1309 x
1310         STORE mcounter+1 TO mcounter           && Advance counter by 1
1311         STORE 'MCAGE'+LTRIM(STR(mcounter)) TO mcage
1312         STORE &mcage TO mcage                  && Store next cage in matrix
1313         IF TYPE('mcage') = 'N'                 && If cage is numeric,
1314             STORE STR(mcage,5) TO mcage        && Convert to string
1315         ENDIF
1316     ENDDO                                      && End searching for cage
1317
```

```
    SET COLOR TO W/B                          && Enhance screen colors
    STORE 15 TO mcolumn                       && Initialize screen pointer
    STORE 1 TO mcol                           && Initialize pointer

    DO WHILE mcol < 7                         && Print the prices
        STORE 'MEXT_'+LTRIM(STR(mcounter))+'_'+LTRIM(STR(mcol)) TO mprice
        STORE 'MORDER'+LTRIM(STR(mcol)) TO morder
        STORE &mprice/&morder TO mnet_price
        @ 19,mcolumn-1 SAY mnet_price PICTURE '@Z 9,999.9999'
        @ 20,mcolumn SAY &mprice PICTURE '@Z 99,999.99'

        STORE mcol+1 TO mcol                  && Advance the counters
        STORE mcolumn+11 TO mcolumn
    ENDDO

    SET COLOR TO G/B                          && Return color to normal
ENDIF


*   >>>   GET USER RESPONSE   <<<

@ 23,0 SAY ' '                                && Position curser
WAIT ' ' TO mchoice                           && Get user's input
IF UPPER(mchoice) = 'P'                       && User pressed 'P'
    STORE mreturn TO mchoice                  && Prepare to return to prior scre
en
ELSE
    STORE 'Q' TO mchoice                      && Prepare to quit
ENDIF

RETURN                                        && Return control to calling progr
am



***   RETURN TO OPENING SCREEN   ******************************
*                                                              *
*   If the users presses <ESC><ESC> while entering the NSN,    *
*   control is directed to this program code.  The user will be *
*   returned to the program information screen.                *
*                                                              *
****************************************************************

PROCEDURE RTURN                               && Label this block of code

STORE 'Q' TO mchoice                          && Prepare to quit program
RETURN TO MASTER                              && Return to master program
```

```
1364  *: EOF: SCREENS.PRG
```

```
*:*****************************************************************
*:
*:      Program: SELCTVEN.PRG
*:
*:       System: Automated Vendor Selection Assistant
*:       Author: Capt Daniel E. Hagmaier
*:    Copyright (c) 1991, United States Air Force
*:
*:    Called by: AVSA.PRG
*:
*:         Uses: PR_TEMP.DBF
*:             : NSN.DBF
*:             : PRICE.DBF
*:             : DCRL.DBF
*:             : VENDOR.DBF
*:
*:      Indexes: N_NSN.NDX
*:             : P_C_CODE.NDX
*:             : DCR_CAGE.NDX
*:             : V_C_MIL.NDX
*:
*: Documented: 7/30/91       11:36                    SNAP! version 1.73
*:*****************************************************************



***   SELECT QUALIFIED VENDORS   ********************************
*                                                                 *
*   This procedure file selects the qualified vendors bidding     *
*   on the item identified in the Screens Procedure.  Memory      *
*   variables mNSN, mQUANTITY, and mSETASIDE from the input       *
*   screen, are used in the selection process.                    *
*                                                                 *
*******************************************************************



***   VENDOR SELECTION   ****************************************
*                                                                 *
*   This portion of the code creates a temporary datafile,        *
*   PR_TEMP.DBF.  In it,  records from the price database that    *
*   contain bidding vendors, will be copied.                      *
*                                                                 *
*******************************************************************



*   >>>   LOAD TEMPORARY FILE   <<<
```

```
SELECT A                                        && Select primary work area
USE pr_temp                                     && Activate PR_TEMP.dbf
SELECT B                                        && Select alternate work area
USE nsn INDEX n_nsn                             && Activate NSN.dbf
SELECT C                                        && Select alternate work area
USE price INDEX p_c_code                        && Activate PRICE.dbf w/ CAGE & PR
_CODE
SELECT B                                        && Select alternate work area
SET RELATION TO cage+price_code INTO price      && Tie the two db files together
SEEK mnsn                                       && Locate NSN
DO WHILE nsn = mnsn
    SELECT A                                    && Select primary work area
    APPEND BLANK                                && Generate blank record
    REPLACE cage WITH nsn->cage, price_code WITH nsn->price_code,mil_spec WITH nsn
->mil_spec
    REPLACE qmin1 WITH price->qmin1, qmax1 WITH price->qmax1, price1 WITH price->p
rice1
    REPLACE qmin2 WITH price->qmin2, qmax2 WITH price->qmax2, price2 WITH price->p
rice2
    REPLACE qmin3 WITH price->qmin3, qmax3 WITH price->qmax3, price3 WITH price->p
rice3
    REPLACE qmin4 WITH price->qmin4, qmax4 WITH price->qmax4, price4 WITH price->p
rice4
    REPLACE qmin5 WITH price->qmin5, qmax5 WITH price->qmax5, price5 WITH price->p
rice5
    REPLACE qmin6 WITH price->qmin6, qmax6 WITH price->qmax6, price6 WITH price->p
rice6
    REPLACE qmin7 WITH price->qmin7, qmax7 WITH price->qmax7, price7 WITH price->p
rice7
    REPLACE qmin8 WITH price->qmin8, qmax8 WITH price->qmax8, price8 WITH price->p
rice8
    REPLACE qmin9 WITH price->qmin9, qmax9 WITH price->qmax9, price9 WITH price->p
rice9
    REPLACE qmin10 WITH price->qmin10, qmax10 WITH price->qmax10, price10 WITH pri
ce->price10
    SELECT B                                    && Select alternate work area
    SKIP                                        && Move pointer to next record
ENDDO                                           && End of load PR_TEMP.dbf


***   REMOVE UNQUALIFIED VENDORS   ****************************
*                                                              *
*   Of the vendors in the temporary database, vendors which    *
*   have been de-barred are 'Deleted'.  Also, if the procure-  *
*   ment is set aside for small business, the large vendors    *
*   will be 'Deleted'.  The remaining vendors will latter be   *
*   checked for other problems, or excellence.                 *
```

```
85  *                                                          *
86  ************************************************************
87
88
89
90  *   >>>   IDENTIFY DE-BARRED VENDOR(S)   <<<
91
92  SELECT C                                    && Establish alternate work area
93  USE dcrl INDEX dcr_cage                     && Open DCRL.DBF file
94
95  SELECT pr_temp                              && Activate PR_TEMP.DBF
96  GO TOP                                      && Set pointer to the first record
97  DO WHILE .NOT. EOF()                        && Check entire file
98      SELECT dcrl                             && Activate DCRL.DBF
99      LOCATE FOR  (pr_temp->cage = cage)      && Look for first cage in PR_TEMP.
99  DBF
100     IF FOUND()
101         IF category1 = 'A' .OR. category2 = 'A' .OR. category3 = 'A';
102             .OR. category4 = 'A' .OR. category5 = 'A' .OR. category6 = 'A'
103             SELECT pr_temp
104             DELETE                          && Marks current rec for deletion
105         ENDIF
106     ENDIF
107     SELECT pr_temp                          && Activate PR_TEMP.DBF
108     SKIP                                    && Advance to next record
109 ENDDO                                       && Repeat until end of PR_TEMP.DBF
110
111
112
113 *   >>>   SET-A-SIDE   <<<
114
115 IF msetaside = "Y"                          && If PR is for small business
116
117
118     *>>>  LINK PR_CAGE.DBF WITH VENDOR.DBF   <<<
119
120     SELECT B                                && Select alternate work area
121     USE vendor INDEX v_c_mil                && Open VENDOR.DBF for use
122     SELECT A                                && Select primary work area
123     SET RELATION TO cage+mil_spec INTO vendor   && Link datafiles together
124
125
126 *   >>>  REMOVE VENDORS CODED AS LARGE   <<<
127
128     SELECT pr_temp                          && Activate PR_TEMP.DBF
129     GO TOP                                  && Set pointer to first record
130     DO WHILE .NOT. EOF()                    && Check entire file
131         IF vendor->size_code = 'A'          && 'A' equals large vendor
```

```
132             DELETE                          && Mark current record for deletio
132 n
133         ENDIF
134         SKIP                                && Move pointer to next record
135     ENDDO                                   && Repeat until end of file
136
137
138
139 *  >>>  REMOVE LINK WITH VENDOR.DBF   <<<
140
141     SET RELATION TO                         && Removes relation
142
143
144
145 ENDIF                                       && End of Set-A-Side coding
146
147
148
149 *>>>  REMOVE DELETED FILES   <<<
150
151 SELECT pr_temp                              && Insure PR_TEMP.DBF is active
152 PACK                                        && Remove any deleted records
153
154 RETURN                                      && Return control to calling progr
154 am
155
156 *: EOF: SELCTVEN.PRG
```

Appendix B: Variable Cross Reference Table

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Cross-Reference Report
Date: 7/30/91
Time: 11:38

AVSA.PRG	40	41	43	44	45	46	47	48	49	62
	63	64	65	66	67	68	73	74	79	80
	81	82	83	84	85	86	88	89	90	91
	92	98	105	127	183	185	186	191	193	223
	226	227	234	236	238	252	254	255	256	257
	258	259	260							
SCREENS.PRG	50	51	64	75	78	83	108	109	114	115
	116	117	118	119	120	121	123	144	145	146
	147	148	154	155	156	157	158	159	161	167
	173	174	176	177	178	179	180	181	186	187
	188	192	195	196	197	201	202	203	204	210
	211	212	218	220	221	222	224	225	226	229
	235	236	238	239	240	245	246	250	256	257
	261	262	266	268	279	283	284	286	297	303
	305	307	308	316	318	320	335	336	341	353
	410	426	427	440	441	446	447	451	452	453
	463	464	475	480	492	493	494	495	496	497
	498	502	508	515	516	520	525	532	544	545
	546	548	549	550	551	555	556	557	559	564
	579	580	581	595	596	605	606	620	623	624
	628	629	635	637	648	650	651	656	686	751
	759	767	770	784	785	786	787	788	789	790
	798	799	800	808	809	810	811	850	851	854
	857	860	866	867	871	873	875	880	887	888
	889	890	891	895	908	910	911	913	926	928
	929	931	944	945	953	954	955	959	961	964
	969	975	976	978	979	983	988	989	990	991
	992	993	995	1001	1033	1040	1041	1042	1045	1046
	1048	1061	1062	1070	1071	1072	1076	1078	1081	1086
	1092	1093	1095	1096	1100	1105	1113	1114	1115	1116
	1117	1128	1129	1130	1134	1147	1149	1150	1152	1157
	1158	1160	1162	1165	1166	1171	1172	1174	1176	1179
	1185	1188	1189	1195	1207	1242	1260	1261	1266	1268
	1269	1273	1278	1292	1294	1303	1304	1305	1306	1309
	1310	1313	1314	1316	1318	1320	1322	1329	1333	1340
	1341	1342	1344	1347	1359	1361	1362			
SELCTVEN.PRG	50	51	52	54	55	57	58	60	61	73
	75	92	93	95	96	97	98	104	107	108
	109	115	120	121	122	123	128	129	130	131
	132	134	135	145	151	152	154			

PREPVEN.PRG	41	42	43	44	45	46	53	54	59	60
	66	70	71	78	80	81	84	85	86	89
	90	105	107	108	110	111	112	126	127	128
	129	134	135	140	141	142	144	188	240	241
	242	243	244	245	250	252	253	256	263	264
	266	267	272	273	274	275	279	280	282	284
	291	293	294	295	297	311	312	313	315	316
	317	318	319	321	322	324	326	331	332	333
	337	342	343	344	347	349	356	357	371	374
	375	376	378	379	380	384				

10

AVSA.PRG	99	113	158							
SCREENS.PRG	54	258	283	343	343	369	374	385	386	387
	388	389	390	585	610	674	674	711	715	728
	729	730	731	732	733	734	735	736	800	910
	928	948	1065	1153	1246	1247	1248	1278	1279	1280
PREPVEN.PRG	233									

100

SCREENS.PRG	244	244	814	822	830	838
PREPVEN.PRG	255	265	290			

1000

SCREENS.PRG	244	248	248	248	248	808

11

AVSA.PRG	160				
SCREENS.PRG	310	495	544	548	1330
PREPVEN.PRG	134	159			

12

AVSA.PRG	162									
SCREENS.PRG	55	220	221	467	471	474	480	986	1249	1250
	1281	1281	1282	1283	1284					

120

AVSA.PRG	149

13

SCREENS.PRG	305	985	985	1100	1200	1200	1218	1225	1232

14

SCREENS.PRG	167	168	311	344	344	370	375	392	393	394
	395	396	397	675	675	712	716	738	739	740
	741	742	743	744	745	746	1034	1034	1035	1036
	1037	1038	1120	1122	1124	1126	1251	1252	1253	1285
	1286	1287								

15

SCREENS.PRG	1319

```
151
  AVSA.PRG        151

16
  AVSA.PRG         79
  SCREENS.PRG      56 1197 1197 1208 1212 1218 1219 1220 1221 1222
                 1223 1294

17
  SCREENS.PRG    1200 1201 1202 1203 1204 1205 1285 1294 1295 1296
                 1297 1298 1299

18
  SCREENS.PRG     235  239  258  261  345  345  346  347  348  350
                  351  360  361  362  363  365  366  371  376  667
                  669  670  671  671  677  678  679  680  681  682
                  683  684  698  699  700  701  702  703  704  705
                  706  707  708 1198 1198 1209 1213 1232 1233 1234
                 1235 1236 1237

181
  AVSA.PRG        153

182
  SCREENS.PRG     374  375  715  716 1213

185
  SCREENS.PRG     373  376  688  714 1211 1212

187
  SCREENS.PRG     696

188
  SCREENS.PRG     405  708

19
  SCREENS.PRG     175  464  465  467  469  983 1254 1326

197
  SCREENS.PRG     385  386  387  388  389  390  392  393  394  395
                  396  397  728  730  732  734  735  736  738  740
                  742  744  745  746 1232 1233 1234 1235 1236 1237

199
  SCREENS.PRG     369  370  711  712 1209

2
  AVSA.PRG        142  255
  SCREENS.PRG      51  175  193  194  195  198  199  200  401  401
                  404  405  585  610  656  768  770  806 1195 1215
                 1244 1246 1249 1251 1254 1255
```

20
SCREENS.PRG 166 347 347 354 361 379 386 393 471 473
1200 1201 1202 1203 1204 1205 1255 1327

200
SCREENS.PRG 404 698

201
SCREENS.PRG 686 689

202
SCREENS.PRG 700 702 704

203
SCREENS.PRG 690 692

204
SCREENS.PRG 368 371 710 1207 1208

206
SCREENS.PRG 719 721 723

207
SCREENS.PRG 360 361 362 364 365 366 699 701 703 705
706 707 1216 1225 1226 1227 1228 1229 1230

209
SCREENS.PRG 353 354 355 356 357 358 687 691 693 694
695 720 1215 1218 1219 1220 1221 1222 1223

21
SCREENS.PRG 474 1195 1225 1226 1227 1228 1229 1230

212
AVSA.PRG 155

215
SCREENS.PRG 729 731 733 739 741 743

216
SCREENS.PRG 363 378 379 380 381 382 383 718 722 724
725 726

219
SCREENS.PRG 467 469 471 473 474

22
SCREENS.PRG 50 341 349 364 751 1282

```
23
   SCREENS.PRG      57  109  166  210  218  229  411  413  417  419
                   428  430  433  435  481  752  754  756  760  762
                   764  944  947  965  967 1041 1061 1064 1082 1084
                  1128 1149 1152 1164 1166 1257 1261 1339

24
   SCREENS.PRG     117  416  418  420  432  434  436  753  755  757
                   761  763  765  948 1065 1153 1201 1201 1219 1226
                  1233 1258

243
   AVSA.PRG        157

25
   SCREENS.PRG    1257

26
   SCREENS.PRG      57  168  219  235  305  928  947 1064 1152 1258

27
   SCREENS.PRG     283  311  910 1295
   PREPVEN.PRG     233

273
   AVSA.PRG        159

28
   SCREENS.PRG     187  929

29
   SCREENS.PRG     218  911

3
   AVSA.PRG        144
   SCREENS.PRG     341  342  343  344  368  369  370  371  411  413
                   416  442  444  447  555  584  587  599  601  609
                   613  657  666  670  670  671  673  674  675  686
                   698  710  711  712  752  753  871  983 1100 1199
                  1244 1245

30
   SCREENS.PRG      52   54   55  284  308  309  417  418  754  755
   PREPVEN.PRG     234

304
   AVSA.PRG        161

31
   AVSA.PRG        143
   SCREENS.PRG      56  176  310  348  348  355  362  380  387  394
                   433  434  762  763 1247
```

```
32
   SCREENS.PRG      51  996  997  998  999

33
   SCREENS.PRG      53

334
   AVSA.PRG        163

34
   SCREENS.PRG     482

35
   SCREENS.PRG    1202 1202 1220 1227 1234

36
   SCREENS.PRG     221  986  986

3600
   SCREENS.PRG     193  198

365
   SCREENS.PRG     812  820  822  824  826

366
   SCREENS.PRG     248  828  836  838  840  842

37
   SCREENS.PRG     995 1114

38
   SCREENS.PRG    1002 1007 1012 1017 1022 1027 1252 1296

39
   SCREENS.PRG    1199 1199 1215 1216

4
   AVSA.PRG         43  146  174
   SCREENS.PRG     194  199  341  345  346  347  348  349  350  351
                   353  354  355  356  357  358  428  430  432  515
                   658  659  666  666  670  677  686  687  688  760
                   761  798  983  984  985  996  997  998  999 1100
                  1101 1268 1268 1269 1270 1271 1273

41
   SCREENS.PRG    1245

42
   SCREENS.PRG     220  349  349  356  363  364  381  388  395

43
   SCREENS.PRG    1273 1274 1275 1276 1279 1286
```

```
44
   SCREENS.PRG     465

45
   SCREENS.PRG     483

46
   SCREENS.PRG    1203 1203 1221 1228 1235

47
   SCREENS.PRG     666  667  667  668  688  689  700  719  729  739
                   965  967 1082 1084 1164 1166

48
   SCREENS.PRG     239  801

49
   SCREENS.PRG     283  400  400  401  404  910 1297
   PREPVEN.PRG     233

5
   AVSA.PRG        148
   SCREENS.PRG     176  195  202  305  464  480  481  482  483  484
                   485  659  660  995 1106 1107 1108 1109 1114 1269
                  1274 1306 1314

50
   SCREENS.PRG     469  473  656  657  658  660 1005 1010 1015 1020
                  1025 1030

51
   SCREENS.PRG     453  455  928

53
   SCREENS.PRG     305  350  350  357  365  382  389  396  678  678
                   701  720  730  740

54
   SCREENS.PRG     802

55
   SCREENS.PRG     117  419  420  756  757

56
   SCREENS.PRG     435  436  484  764  765  803

57
   SCREENS.PRG     804 1204 1204 1222 1229 1236

59
   AVSA.PRG        145  175
   SCREENS.PRG     679  679  690  702  721  731  741 1250
```

```
6
   AVSA.PRG        150
   SCREENS.PRG      52  283  342  342  368  373  378  379  380  381
                   382  383  673  673  678  710  714  718  719  720
                   721  722  723  724  725  726  910  928  984  984
                  1101 1101 1270 1275
   PREPVEN.PRG     233

60
   SCREENS.PRG     194  199  813  821  829  837 1121 1123 1125 1127
                  1298

62
   SCREENS.PRG     815  818  823  826  831  834  839  842

63
   SCREENS.PRG     816  824  832  840

64
   SCREENS.PRG     351  351  358  366  383  390  397 1248

65
   SCREENS.PRG     680  680  691  703  722  732  742

66
   SCREENS.PRG     850 1253

67
   SCREENS.PRG     485  681  681  692  704  723  733  743

68
   SCREENS.PRG     852 1205 1205 1223 1230 1237

69
   SCREENS.PRG     261  682  682  693  705  724  734  744

7
   AVSA.PRG        106  115  152
   SCREENS.PRG     185  195  200  303  308  494  520  780  986  996
                  1002 1199 1271 1276 1322
   PREPVEN.PRG     332

70
   SCREENS.PRG     855 1283

71
   SCREENS.PRG     683  683  694  706  725  735  745 1299

72
   SCREENS.PRG     858
```

```
73
   SCREENS.PRG     684  684  695  707  726  736  746

74
   SCREENS.PRG     401  861

75
   SCREENS.PRG     202  341  342  343  344  345  373  374  375  376
                   402  402  405  668  669  669  671  673  674  675
                   696  708  714  715  716 1284

76
   SCREENS.PRG     984  985 1101

77
   SCREENS.PRG     983 1100 1280 1287

78
   SCREENS.PRG    1196 1197 1198

79
   SCREENS.PRG      50  109  167 1195 1211 1212 1213

8
   AVSA.PRG        154
   SCREENS.PRG      53  284  911  929 1106 1120 1196 1196 1207 1211
                  1216 1278
   PREPVEN.PRG     234

9
   AVSA.PRG        156
   SCREENS.PRG     187  202  202  219  309  346  346  353  360  378
                   385  392  677  677  687  699  718  728  738

90
   AVSA.PRG        147

A
   SCREENS.PRG     154  158
   SELCTVEN.PRG     50   60  122
   PREPVEN.PRG     165

ADDRESS1
   SCREENS.PRG    1268

ADDRESS2
   SCREENS.PRG    1269

ADDRESS3
   SCREENS.PRG    1270
```

ADDRESS4
SCREENS.PRG 1271

ALT
SCREENS.PRG 805

ANALZSCR
AVSA.PRG 192
SCREENS.PRG 906

AND
SCREENS.PRG 96

APPEND
SELCTVEN.PRG 61
PREPVEN.PRG 164

ASSISTANT
SCREENS.PRG 85

AUTOMATED
SCREENS.PRG 85

AWARDSCR
AVSA.PRG 214
SCREENS.PRG 1147

B
AVSA.PRG 42
SCREENS.PRG 78 114 114 120 156 186 188 304 307 336
426 440 446 452 463 466 468 470 472 475
508 510 512 516 527 529 540 581 583 586
588 596 598 600 606 608 612 614 621 621
633 651 759 767 769 787 792 794 796 799
809 846 889 889 893 945 956 979 1062 1073
1096 1105 1113 1150 1159 1189 1260 1292 1318 1333
SELCTVEN.PRG 52 56 73 120
PREPVEN.PRG 105

BELL
AVSA.PRG 40 253

BG
AVSA.PRG 42
SCREENS.PRG 426 446 759 769 956 1073 1159

BID
SCREENS.PRG 87

BLANK
SELCTVEN.PRG 61
PREPVEN.PRG 164

```
BUSINESS
   SCREENS.PRG     259

C
   SCREENS.PRG     988 1115
   SELCTVEN.PRG     54   92
   PREPVEN.PRG      59   77  107  272  311

CAGE
   SCREENS.PRG     159  502  789  967  993  995 1084 1114 1117 1119
                  1119 1164 1278 1309
   SELCTVEN.PRG     57   62   62   99   99  123
   PREPVEN.PRG      45   66   84  111  165  165  252  253  264  286
                   317  346  353

CATEGORY1
   SCREENS.PRG    1002 1003
   SELCTVEN.PRG    101

CATEGORY2
   SCREENS.PRG    1007 1008
   SELCTVEN.PRG    101

CATEGORY3
   SCREENS.PRG    1012 1013
   SELCTVEN.PRG    101

CATEGORY4
   SCREENS.PRG    1017 1018
   SELCTVEN.PRG    102

CATEGORY5
   SCREENS.PRG    1022 1023
   SELCTVEN.PRG    102

CATEGORY6
   SCREENS.PRG    1027 1028
   SELCTVEN.PRG    102

CAUSE
   SCREENS.PRG    1123

CAUSE_CODE
   SCREENS.PRG    1122

CDCF
   SCREENS.PRG     509  793  857 1071 1116
   PREPVEN.PRG      60   65   68

CDCFSCR
   AVSA.PRG        210
   SCREENS.PRG    1059
```

```
CDCF_N_C
   SCREENS.PRG    1116
   PREPVEN.PRG      60

CENTURY
   AVSA.PRG         41

CHR
   SCREENS.PRG     185  303  353  354  355  356  357  358  360  361
                   362  363  364  365  366  368  369  370  371  373
                   374  375  376  378  379  380  381  382  383  385
                   386  387  388  389  390  392  393  394  395  396
                   397  404  405  467  469  471  473  474  686  687
                   688  689  690  691  692  693  694  695  696  698
                   699  700  701  702  703  704  705  706  707  708
                   710  711  712  714  715  716  718  719  720  721
                   722  723  724  725  726  728  729  730  731  732
                   733  734  735  736  738  739  740  741  742  743
                   744  745  746 1207 1208 1209 1211 1212 1213 1215
                  1216 1218 1219 1220 1221 1222 1223 1225 1226 1227
                  1228 1229 1230 1232 1233 1234 1235 1236 1237

CLEAR
   AVSA.PRG        252
   SCREENS.PRG      45   77  121  153  202  210  229  281  302  335
                   650  908  926  944  978 1061 1095 1149 1188
   PREPVEN.PRG     232

CLOSE
   AVSA.PRG        233

CODE
   SCREENS.PRG    1003 1008 1013 1018 1023 1028

COLOR
   AVSA.PRG         42
   SCREENS.PRG      78  114  120  186  188  304  307  336  426  440
                   446  452  463  466  468  470  472  475  508  510
                   512  516  527  529  540  581  583  586  588  596
                   598  600  606  608  612  614  621  633  651  759
                   767  769  787  792  794  796  799  809  846  889
                   893  945  956  979 1062 1073 1096 1105 1113 1150
                  1159 1189 1260 1292 1318 1333

COMPETITIVELY
   SCREENS.PRG      87

CONFIRM
   AVSA.PRG        183  254
   SCREENS.PRG     256
```

```
CONTINUE
   SCREENS.PRG      107  963 1080

CORR
   SCREENS.PRG     1127

CORR_CODE
   SCREENS.PRG     1126

D
   SCREENS.PRG      990 1045
   PREPVEN.PRG      240

DATABASES
   AVSA.PRG         233

DATE
   AVSA.PRG         127  132  133  134
   PREPVEN.PRG      284

DATE1
   SCREENS.PRG     1002

DATE2
   SCREENS.PRG     1007

DATE3
   SCREENS.PRG     1012

DATE4
   SCREENS.PRG     1017

DATE5
   SCREENS.PRG     1022

DATE6
   SCREENS.PRG     1027

DAY
   AVSA.PRG         127  134

DAYS
   SCREENS.PRG      802 1282

DCRL
   SCREENS.PRG      989  992 1002 1002 1003 1007 1007 1008 1012 1012
                   1013 1017 1017 1018 1022 1022 1023 1027 1027 1028
                   1033
   SELCTVEN.PRG      93   98
   PREPVEN.PRG       44
```

```
DCRLCODE
   SCREENS.PRG     991 1001

DCR_CAGE
   SCREENS.PRG     989
   SELCTVEN.PRG     93

DECIMALS
   AVSA.PRG         43  255

DELETE
   SELCTVEN.PRG    104  132
   PREPVEN.PRG     376

DELETED
   AVSA.PRG         44  256

DELIVERY
   SCREENS.PRG     805 1285

DISC
   SCREENS.PRG     801 1121 1281

DISC_CODE
   SCREENS.PRG    1120

DISP
   SCREENS.PRG    1125

DISP_CODE
   SCREENS.PRG    1124

DOUBLE
   SCREENS.PRG      50  109  283  305  341  342  345  400  401  402
                   666  667  668  669  670  671  673  679  681  910
                   928  983  984 1100 1101 1195 1196 1197
   PREPVEN.PRG     233

ELSE
   AVSA.PRG        223  226
   SCREENS.PRG     181  412  429  443  536  558  627  817  825  833
                   841  874  966  977 1083 1094 1165 1175 1186 1343
   PREPVEN.PRG     131  175  180  293  348

ENDCASE
   AVSA.PRG        164  218
   SCREENS.PRG     513  530  797  844
```

```
ENDDO
   AVSA.PRG        103  110  119  121  219  241
   SCREENS.PRG     201  204  252  264  546  560  626  631  882  892
                   970 1087 1131 1179 1316 1331
   SELCTVEN.PRG     75  109  135
   PREPVEN.PRG      54   72   90  144  193  212  215  281  351  357
                   379

ENDIF
   AVSA.PRG        122  177  178  187  225  228
   SCREENS.PRG     203  227  247  251  406  414  431  445  459  538
                   560  565  590  603  615  632  810  819  827  835
                   843  853  856  859  862  876  881  962  968 1006
                  1011 1016 1021 1026 1031 1043 1079 1085 1132 1167
                  1177 1178 1307 1315 1334 1345
   SELCTVEN.PRG    105  106  133  145
   PREPVEN.PRG      50   51   69   87  133  139  143  162  183  184
                   196  197  205  208  257  258  267  292  296  297
                   334  338  350  377  380

ENDTEXT
   SCREENS.PRG     108

EOF
   SCREENS.PRG     960  964  975 1077 1081 1092 1119 1163 1173 1184
   SELCTVEN.PRG     97  130
   PREPVEN.PRG      43   64   82  126  326  374

ESCAPE
   AVSA.PRG         45  257
   SCREENS.PRG     160  161  211  212

EXIT
   PREPVEN.PRG     161  333

EXT_PRICE
   PREPVEN.PRG     173  178  181  195  201  207  250  318  319  355

FOB
   SCREENS.PRG     850 1286

FOR
   SCREENS.PRG     502  789  954 1003 1008 1013 1018 1023 1028 1071
   SELCTVEN.PRG     99
   PREPVEN.PRG     253  264

FOUND
   SCREENS.PRG     179 1004 1009 1014 1019 1024 1029
   SELCTVEN.PRG    100
   PREPVEN.PRG      46   67   85  254  275
```

```
G
   AVSA.PRG          42
   SCREENS.PRG       78  120  188  307  336  440  452  463  472  475
                    512  516  529  540  583  588  598  608  614  633
                    651  767  787  796  799  846  893  945  979 1062
                   1096 1105 1150 1189 1292 1333

GET
   SCREENS.PRG      117  176  221  239  261

GO
   SELCTVEN.PRG      96  129
   PREPVEN.PRG       42   63   81  112  245  313

GOTO
   SCREENS.PRG      955  961 1072 1078 1158 1176
   PREPVEN.PRG      373

GR
   SCREENS.PRG      468  527

HAVE
   SCREENS.PRG       87

HISTORY
   PREPVEN.PRG      273

HISTORY1
   PREPVEN.PRG      290

HISTORY2
   SCREENS.PRG      610
   PREPVEN.PRG      294

HIST_N_D
   PREPVEN.PRG      273

HOLD
   AVSA.PRG         237
   PREPVEN.PRG      108  163  242  312

H_EXT_PR
   PREPVEN.PRG      243

H_ORD_Q
   PREPVEN.PRG      243  312
```

IF

AVSA.PRG	68	74	127	175	185	191				
SCREENS.PRG	179	224	244	248	399	410	427	441	451	532
	555	563	580	595	605	622	808	814	822	830
	838	851	854	857	860	871	879	960	964	975
	1004	1009	1014	1019	1024	1029	1077	1081	1092	1163
	1172	1173	1184	1305	1313	1341				
SELCTVEN.PRG	100	101	115	131						
PREPVEN.PRG	46	47	67	85	129	137	140	160	172	176
	188	189	201	203	254	255	265	275	290	294
	327	332	346	371	375					

IIF

AVSA.PRG	174
SCREENS.PRG	263

INDEX

SCREENS.PRG	155	157	989	1116		
SELCTVEN.PRG	53	55	93	121		
PREPVEN.PRG	60	78	106	243	273	312

INFO_SCR

AVSA.PRG	67
SCREENS.PRG	75

INITLSCR

AVSA.PRG	73
SCREENS.PRG	924

INPUTSCR

AVSA.PRG	184
SCREENS.PRG	139

INT

SCREENS.PRG	244	244	248	248
PREPVEN.PRG	177	189	203	204

INTENSITY

SCREENS.PRG	115	119	266

INTEREST.

SCREENS.PRG	89

INTO

SCREENS.PRG	159	
SELCTVEN.PRG	57	123
PREPVEN.PRG	111	

ITEM

SCREENS.PRG	89	94

```
KNOW
   SCREENS.PRG      94

LEFT
   SCREENS.PRG     996  997  998  999 1003 1008 1013 1018 1023 1028
                  1121 1123 1125 1127

LOCATE
   SCREENS.PRG     502  789  954 1003 1008 1013 1018 1023 1028 1071
   SELCTVEN.PRG     99
   PREPVEN.PRG     253  264

LOOP
   AVSA.PRG        186

LOT_SIZE
   PREPVEN.PRG     188  189  189  191  194  195

LOW
   PREPVEN.PRG     255

LTRIM
   AVSA.PRG        100  107  116  116
   SCREENS.PRG     521  521  533  562  585  610  785  878 1311 1323
                  1323 1324
   PREPVEN.PRG     128  136  158  166  167  211  335  345  354  354

MASTER
   SCREENS.PRG    1362

MCAGE
   AVSA.PRG        100  101
   SCREENS.PRG     498  502  515  562  563  785  789  798  878  879
                  1304 1306 1306 1309 1311 1312 1312 1314 1314
   PREPVEN.PRG     252  253  264  345  346  346  353

MCAGE1
   SCREENS.PRG    1304
   PREPVEN.PRG     317

MCELL
   PREPVEN.PRG     335  336

MCHOICE
   AVSA.PRG         62   68  185  198  199  201  204  207  209  211
                   213
   SCREENS.PRG     116  117  225  620  624  625  629  630  888  890
                   891  957  959  969  976 1040 1074 1076 1086 1093
                  1160 1162 1171 1172 1185 1340 1341 1342 1344 1361
```

```
MCOL
   SCREENS.PRG     496  520  521  533  545  545  549 1320 1322 1323
                  1324 1329 1329

MCOLUMN
   AVSA.PRG        114  115  116  118  118
   SCREENS.PRG     495  535  537  544  544  548 1319 1326 1327 1330
                  1330
   PREPVEN.PRG     322  331  331  332  335  354

MCONTINUE
   SCREENS.PRG     497  501  564  786  788  880

MCOUNT
   SCREENS.PRG     493  555  557  559  559  779  871  873  875  875

MCOUNTER
   AVSA.PRG         98   99  100  102  102  105  106  107  109  109
   SCREENS.PRG     492  521  551  551  562  778  785  866  866  878
                  1303 1310 1310 1311 1323

MCURRENT
   SCREENS.PRG     147  196  198

MDAY
   AVSA.PRG         88  127  134  169

MDELIVERY
   SCREENS.PRG     805  808  812  814  816  818  820  822  824  826
                   828  830  832  834  836  838  840  842

MEND
   AVSA.PRG         65   66  227

MEP
   AVSA.PRG        116  117
   PREPVEN.PRG     354  355

MEXT_1_1
   PREPVEN.PRG     318

MEXT_PRICE
   PREPVEN.PRG     319

MFLAG
   SCREENS.PRG    1118
   PREPVEN.PRG     343  344  347

MHISTORY1
   AVSA.PRG         92
   SCREENS.PRG     595
   PREPVEN.PRG     291
```

```
MHISTORY2
   AVSA.PRG        93
   SCREENS.PRG    605
   PREPVEN.PRG    295

MHIST_CAGE
   AVSA.PRG        82
   SCREENS.PRG    456
   PREPVEN.PRG    286

MHIST_DATE
   AVSA.PRG        83
   SCREENS.PRG    399  451  454
   PREPVEN.PRG    284

MHIST_PR
   AVSA.PRG        84
   SCREENS.PRG    458
   PREPVEN.PRG    285

MIL_SPEC
   SCREENS.PRG    159
   SELCTVEN.PRG    62   62  123
   PREPVEN.PRG    111

MIN_ORDER
   PREPVEN.PRG    137  201  202

MJ_DATE
   AVSA.PRG       141  143  145  147  149  151  153  155  157  159
                  161  163  169  169  175  176  176
   SCREENS.PRG    805

MLAST_ORD
   PREPVEN.PRG    316  327  337

MLEAP_YR
   AVSA.PRG       174  175
   SCREENS.PRG    812  820  828  836

MLINE
   SCREENS.PRG    579  582  584  585  587  589  589  597  602  602
                  607  609  610  611  613

MLOW
   AVSA.PRG        89
   SCREENS.PRG    526
   PREPVEN.PRG    256
```

```
MLOW_PRICE
   AVSA.PRG          86
   SCREENS.PRG      526  528
   PREPVEN.PRG      250  265

MMAX
   PREPVEN.PRG      128  129  129  136  137  137  140  158  160  176
                    177  177  211

MMIN
   PREPVEN.PRG      166  172  173  174

MMONTH
   AVSA.PRG         133  140  142  144  146  148  150  152  154  156
                    158  160  162

MNET_PRICE
   SCREENS.PRG      534  535 1325 1326
   PREPVEN.PRG      251  255  290  294

MNEW_NSN
   AVSA.PRG          63   74  221
   SCREENS.PRG      318

MNEXTCOL
   PREPVEN.PRG      130  132  134  138  141

MNSN
   AVSA.PRG          79
   SCREENS.PRG      176  178  447  770 1117 1119 1261
   SELCTVEN.PRG      58   59
   PREPVEN.PRG       66  274  279

MOD
   AVSA.PRG         174

MODEL
   SCREENS.PRG      585  610  805 1046
   PREPVEN.PRG      241  255  265  290  294

MONTH
   AVSA.PRG         133

MORDER
   AVSA.PRG         107  108
   SCREENS.PRG      533  534 1324 1325

MORDER1
   SCREENS.PRG      480 1294
   PREPVEN.PRG      315
```

```
MORDER2
   SCREENS.PRG     481 1295

MORDER3
   SCREENS.PRG     482 1296

MORDER4
   SCREENS.PRG     483 1297

MORDER5
   SCREENS.PRG     484 1298

MORDER6
   SCREENS.PRG     485 1299
   PREPVEN.PRG     371  375

MPRICE
   SCREENS.PRG     521  526  528  534  537 1323 1325 1327
   PREPVEN.PRG     167  168  173  178  181

MQUANT
   PREPVEN.PRG     177  178  179

MQUANTITY
   AVSA.PRG         85
   SCREENS.PRG     148  221  224
   PREPVEN.PRG     129  137  137  172  176  177  181  182

MROO
   AVSA.PRG         87
   SCREENS.PRG     239  244  245  248  248  248  248  249  808

MRETURN
   AVSA.PRG         80
   SCREENS.PRG     623  628  887  976 1040 1093 1185 1342

MROW
   AVSA.PRG        112  113  116  120  120
   SCREENS.PRG     494  515  535  537  550  550  556  556  780  798
                   800  801  802  803  804  813  815  816  818  821
                   823  824  826  829  831  832  834  837  839  840
                   842  850  852  855  858  861  867  867  872  872
   PREPVEN.PRG     321  342  345  349  349  354

MSERIES
   PREPVEN.PRG     127  128  134  135  135  136  142  142  158  159
                   166  167  210  210  211

MSETASIDE
   AVSA.PRG         81
   SCREENS.PRG     257  260  260  261  263  263  263
   SELCTVEN.PRG    115
```

```
MSTOP
   SCREENS.PRG     146  193  196

MTIME
   SCREENS.PRG     145  192  193  194  195  197  198  199  200

MUNITS
   PREPVEN.PRG     190  191  192  192  194  195  202  203  203  204
                   204  206  207

MUNIT_PR
   AVSA.PRG         90  202  205  216
   SCREENS.PRG     410  427  441  532  622  635

MUST
   SCREENS.PRG      94

MVALID
   SCREENS.PRG     144  173  174  180  236  237  238  246  250

MVARIATION
   AVSA.PRG         91
   SCREENS.PRG     580
   PREPVEN.PRG     266

MYEAR
   AVSA.PRG        132  174
   SCREENS.PRG     244  244  806  806  808  813  821  829  837

N
   AVSA.PRG         42
   SCREENS.PRG     107  259

NAME
   SCREENS.PRG     800

NAME1
   SCREENS.PRG     996

NAME2
   SCREENS.PRG     997

NAME3
   SCREENS.PRG     998

NAME4
   SCREENS.PRG     999

NET
   SCREENS.PRG     804
```

```
NOVENSCR
   AVSA.PRG        224
   SCREENS.PRG     297

NSN
   SCREENS.PRG     155 1119
   SELCTVEN.PRG     53   59   62   62   62
   PREPVEN.PRG     279

N_NSN
   SCREENS.PRG     155
   SELCTVEN.PRG     53

OF
   SCREENS.PRG      89

ORD_QUANT
   PREPVEN.PRG     174  179  182  189  189  191  194  206  315  316
                   327  336  337

OTHERWISE
   AVSA.PRG        215

PACK
   SELCTVEN.PRG    152

PREPVEN
   AVSA.PRG        193

PRICE
   SELCTVEN.PRG     55   57   63   63   63   64   64   64   65   65
                    65   66   66   66   67   67   67   68   68   68
                    69   69   69   70   70   70   71   71   71   72
                    72   72
   PREPVEN.PRG     285  290

PRICE1
   SELCTVEN.PRG     63   63

PRICE10
   SELCTVEN.PRG     72   72

PRICE2
   SELCTVEN.PRG     64   64

PRICE3
   SELCTVEN.PRG     65   65

PRICE4
   SELCTVEN.PRG     66   66
```

```
PRICE5
   SELCTVEN.PRG      67   67

PRICE6
   SELCTVEN.PRG      68   68

PRICE7
   SELCTVEN.PRG      69   69

PRICE8
   SELCTVEN.PRG      70   70

PRICE9
   SELCTVEN.PRG      71   71

PRICESCR
   AVSA.PRG         203  206  217
   SCREENS.PRG      333

PRICE_CODE
   SELCTVEN.PRG      57   62   62

PROB
   SCREENS.PRG      507  791  854  954
   PREPVEN.PRG       49

PROBMSCR
   AVSA.PRG         212
   SCREENS.PRG      942

PROCEDURE
   AVSA.PRG          46
   SCREENS.PRG       44   75  139  279  297  333  648  906  924  942
                   1059 1147 1359

PROCEED
   SCREENS.PRG       94

PR_TEMP
   AVSA.PRG         235
   SCREENS.PRG      500  784  953  993 1070 1117 1119 1157
   SELCTVEN.PRG      51   95   99  103  107  128  151
   PREPVEN.PRG       41   45   49   52   62   66   68   70   80   84
                     86   88  110  213  263  372

P_C_CODE
   SELCTVEN.PRG      55

QMAX1
   SELCTVEN.PRG      63   63
```

QMAX10
SELCTVEN.PRG 72 72

QMAX2
SELCTVEN.PRG 64 64

QMAX3
SELCTVEN.PRG 65 65

QMAX4
SELCTVEN.PRG 66 66

QMAX5
SELCTVEN.PRG 67 67

QMAX6
SELCTVEN.PRG 68 68

QMAX7
SELCTVEN.PRG 69 69

QMAX8
SELCTVEN.PRG 70 70

QMAX9
SELCTVEN.PRG 71 71

QMIN1
SELCTVEN.PRG 63 63
PREPVEN.PRG 375

QMIN10
SELCTVEN.PRG 72 72

QMIN2
SELCTVEN.PRG 64 64

QMIN3
SELCTVEN.PRG 65 65

QMIN4
SELCTVEN.PRG 66 66

QMIN5
SELCTVEN.PRG 67 67

QMIN6
SELCTVEN.PRG 68 68

QMIN7
SELCTVEN.PRG 69 69

```
QMIN8
   SELCTVEN.PRG     70   70

QMIN9
   SELCTVEN.PRG     71   71

QTY_VAR_M
   SCREENS.PRG    1284
   PREPVEN.PRG     265

QTY_VAR_P
   SCREENS.PRG    1283

QUALITY
   SCREENS.PRG     511  795  860
   PREPVEN.PRG      78   83   86

QUANTITY
   SCREENS.PRG      96

Q_CAGE
   PREPVEN.PRG      78

R
   SCREENS.PRG     186  304  466  508  581  586  596  600  606  612
                   792  809

RB
   AVSA.PRG         42
   SCREENS.PRG     470  510  794

READ
   SCREENS.PRG     118  177  222  240  262

RECCOUNT
   AVSA.PRG        191

REINDEX
   PREPVEN.PRG     244

RELATION
   SCREENS.PRG     159
   SELCTVEN.PRG     57  123  141
   PREPVEN.PRG     111

REMIT1
   SCREENS.PRG    1273

REMIT2
   SCREENS.PRG    1274
```

REMIT3
SCREENS.PRG 1275

REMIT4
SCREENS.PRG 1276

REPLACE
SELCTVEN.PRG 62 63 64 65 66 67 68 69 70 71
72
PREPVEN.PRG 49 68 86 165 168 173 174 178 179 181
182 194 195 206 207

REQUIRED.
SCREENS.PRG 96

RESTRICT1
SCREENS.PRG 1034
PREPVEN.PRG 47

RESTRICT2
SCREENS.PRG 1035
PREPVEN.PRG 47

RESTRICT3
SCREENS.PRG 1036
PREPVEN.PRG 47

RESTRICT4
SCREENS.PRG 1037
PREPVEN.PRG 48

RESTRICT5
SCREENS.PRG 1038
PREPVEN.PRG 48

RETURN
SCREENS.PRG 64 123 226 268 286 320 637 895 913 931
1048 1134 1347 1362
SELCTVEN.PRG 154
PREPVEN.PRG 384

RFCC
SCREENS.PRG 1287

RIGHT
SCREENS.PRG 806

RTURN
SCREENS.PRG 161 1359

S
SCREENS.PRG 87

```
SAFETY
   AVSA.PRG        234  239

SCOREBOARD
   AVSA.PRG         47  258

SCREENS
   AVSA.PRG         46

SEEK
   SCREENS.PRG     178  993 1117
   SELCTVEN.PRG     58
   PREPVEN.PRG      45   66   84  274

SELCTSCR
   AVSA.PRG        188
   SCREENS.PRG     279

SELCTVEN
   AVSA.PRG        189

SELECT
   SCREENS.PRG     154  156  158  500  784  953  988  990  992 1001
                  1033 1045 1070 1115 1157 1266
   SELCTVEN.PRG     50   52   54   56   60   73   92   95   98  103
                   107  120  122  128  151
   PREPVEN.PRG      41   44   52   59   62   65   70   77   80   83
                    88  105  107  110  163  213  240  242  263  272
                   311  372

SELECTION
   SCREENS.PRG      85

SELECTS
   SCREENS.PRG      87

SIZE_CODE
   SCREENS.PRG     851 1280
   SELCTVEN.PRG    131

SKIP
   SCREENS.PRG    1130 1174
   SELCTVEN.PRG     74  108  134
   PREPVEN.PRG      53   71   89  214  280  282  324  356  378

SPACE
   AVSA.PRG         79

STATE
   SCREENS.PRG    1279
```

```
STATUS
   AVSA.PRG          48  259

STR
   AVSA.PRG         100  107  116  116
   SCREENS.PRG      521  521  533  562  585  610  785  806  878 1306
                   1311 1314 1323 1323 1324
   PREPVEN.PRG      128  136  158  166  167  211  335  345  354  354

SUBSTR
   SCREENS.PRG      193  194  195  198  199  200

TALK
   AVSA.PRG          49  260

TEXT
   SCREENS.PRG       83

THE
   SCREENS.PRG       85   87   89   94   96

TIME
   SCREENS.PRG      192  197

TITLE
   SCREENS.PRG     1005 1010 1015 1020 1025 1030

TITLESCR
   AVSA.PRG          64
   SCREENS.PRG       44

TOP
   SCREENS.PRG      955  961 1072 1078 1158 1176
   SELCTVEN.PRG      96  129
   PREPVEN.PRG       42   63   81  112  245  313  373

TYPE
   SCREENS.PRG     1305 1313

UI
   SCREENS.PRG      220

UNIT_PRICE
   PREPVEN.PRG      168  195  202  207  251  255

UPPER
   AVSA.PRG         199  201  204  207  209  211  213
   SCREENS.PRG      624  629  890 1341

UP_LIMIT
   SCREENS.PRG      585
   PREPVEN.PRG      265
```

```
USE
   AVSA.PRG        235  237
   SCREENS.PRG     155  157  989  991 1046 1116
   SELCTVEN.PRG     51   53   55   93  121
   PREPVEN.PRG      60   78  106  108  241  273  312

VAL
   SCREENS.PRG     193  194  195  198  199  200  806

VENDOR
   SCREENS.PRG      85   87  157  159  220  800  801  802  804  805
                   850  851 1266
   SELCTVEN.PRG    121  123  131
   PREPVEN.PRG     106  111  137  188  189  189  191  194  195  201
                   202  265

VENDRSCR
   AVSA.PRG        208
   SCREENS.PRG     648

V_C_MIL
   SCREENS.PRG     157
   SELCTVEN.PRG    121
   PREPVEN.PRG     106

W
   SCREENS.PRG    1113 1260 1318

WAIT
   SCREENS.PRG      62  316  625  630  891  969 1042 1086 1129 1171
                  1340

WHILE
   AVSA.PRG         66   99  106  113  115  199
   SCREENS.PRG     174  196  237  260  501  520  624  629  788  890
                   959 1076 1119 1162 1309 1322
   SELCTVEN.PRG     59   97  130
   PREPVEN.PRG      43   64   82  126  134  159  191  279  326  344
                   374

WHO
   SCREENS.PRG      87

WISH
   SCREENS.PRG     107
```

WITH

SELCTVEN.PRG	62	62	62	63	63	63	64	64	64	65
	65	65	66	66	66	67	67	67	68	68
	68	69	69	69	70	70	70	71	71	71
	72	72	72							
PREPVEN.PRG	49	68	86	165	168	173	174	178	179	181
	182	194	195	206	207					

Y

SCREENS.PRG	107	259

YEAR

AVSA.PRG	132

YOU

SCREENS.PRG	94	107

ZAP

AVSA.PRG	236	238

Appendix C: Data Base Structure

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Database Structure Summary
Date: 7/30/91
Time: 11:37

Structure for database : PR_TEMP.DBF
Number of data records : 0
Date of last update : 7/21/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	PRICE_CODE	Character	7	
3	MIL_SPEC	Character	5	
4	QMIN1	Numeric	5	
5	QMAX1	Numeric	5	
6	PRICE1	Numeric	10	4
7	QMIN2	Numeric	5	
8	QMAX2	Numeric	5	
9	PRICE2	Numeric	10	4
10	QMIN3	Numeric	5	
11	QMAX3	Numeric	5	
12	PRICE3	Numeric	10	4
13	QMIN4	Numeric	5	
14	QMAX4	Numeric	5	
15	PRICE4	Numeric	10	4
16	QMIN5	Numeric	5	
17	QMAX5	Numeric	5	
18	PRICE5	Numeric	10	4
19	QMIN6	Numeric	5	
20	QMAX6	Numeric	5	
21	PRICE6	Numeric	10	4
22	QMIN7	Numeric	5	
23	QMAX7	Numeric	5	
24	PRICE7	Numeric	10	4
25	QMIN8	Numeric	5	
26	QMAX8	Numeric	5	
27	PRICE8	Numeric	10	4
28	QMIN9	Numeric	5	
29	QMAX9	Numeric	5	
30	PRICE9	Numeric	10	4
31	QMIN10	Numeric	5	
32	QMAX10	Numeric	5	
33	PRICE10	Numeric	10	4
34	PROB	Logical	1	
35	CDCF	Logical	1	
36	QUALITY	Logical	1	
37	HISTORY	Logical	1	
** Total **			222	

Used by: AVSA.PRG
Used by: SELCTVEN.PRG

Structure for database : HOLD.DBF
Number of data records : 0
Date of last update : 7/21/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	ORD_QUANT	Numeric	5	
3	UNIT_PRICE	Numeric	10	4
4	EXT_PRICE	Numeric	8	2
** Total **			29	

Used by: AVSA.PRG
Used by: PREPVEN.PRG

Structure for database : NSN.DBF
Number of data records : -15550
Date of last update : 7/20/91

Field	Field name	Type	Width	Dec
1	NSN	Character	16	
2	CAGE	Character	5	
3	MIL_SPEC	Character	5	
4	PRICE_CODE	Character	7	
** Total **			34	

Used by: SCREENS.PRG
Used by: SELCTVEN.PRG

Structure for database : DCRLCODE.DBF
Number of data records : 10
Date of last update : 6/12/91

Field	Field name	Type	Width	Dec
1	CODE	Character	1	
2	TITLE	Character	25	
** Total **			27	

Used by: SCREENS.PRG

Structure for database : VENDOR.DBF
Number of data records : 13
Date of last update : 7/21/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	MIL_SPEC	Character	5	
3	NAME	Character	30	
4	SIZE_CODE	Character	1	
5	DELIVERY	Numeric	3	
6	QTY_VAR_P	Numeric	2	
7	QTY_VAR_M	Numeric	2	
8	FOB	Character	1	
9	INSPECT	Numeric	6	2
10	DISC	Numeric	5	3
11	DAYS	Numeric	2	
12	NET	Numeric	2	
13	LOT_SIZE	Numeric	3	
14	MIN_ORDER	Numeric	6	2
15	UI	Character	2	
16	STATE	Character	2	
17	RFCC	Character	1	
18	ADDRESS1	Character	35	
19	ADDRESS2	Character	35	
20	ADDRESS3	Character	35	
21	ADDRESS4	Character	35	
22	REMIT1	Character	35	
23	REMIT2	Character	35	
24	REMIT3	Character	35	
25	REMIT4	Character	35	
** Total **			359	

Used by: SCREENS.PRG
Used by: SELCTVEN.PRG
Used by: PREPVEN.PRG

Structure for database : DCRL.DBF
Number of data records : 318
Date of last update : 7/19/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	NAME1	Character	35	
3	NAME2	Character	35	
4	NAME3	Character	35	
5	NAME4	Character	35	
6	ADDED	Character	8	
7	CHANGED	Character	8	
8	DEL_IND	Character	7	
9	CATEGORY1	Character	15	
10	DATE1 ATE1	Character	8	
11	CATEGORY2	Character	15	
12	DATE2	Character	8	
13	CATEGORY3	Character	15	
14	DATE3	Character	8	
15	CATEGORY4	Character	15	
16	DATE4	Character	8	
17	CATEGORY5	Character	15	
18	DATE5	Character	8	
19	CATEGORY6	Character	15	
20	DATE6	Character	8	
21	RESTRICT1	Character	50	
22	RESTRICT2	Character	50	
23	RESTRICT3	Character	50	
24	RESTRICT4	Character	50	
25	RESTRICT5	Character	50	
** Total **			557	

Used by: SCREENS.PRG
Used by: SELCTVEN.PRG

Structure for database : MODEL.DBF
Number of data records : 1
Date of last update : 7/ 2/91

Field	Field name	Type	Width	Dec
1	LOW	Numeric	2	
2	UP_LIMIT	Numeric	6	
3	HISTORY1	Numeric	2	
4	HISTORY2	Numeric	7	2
5	ALT	Numeric	3	
** Total **			21	

Used by: SCREENS.PRG
Used by: PREPVEN.PRG

Structure for database : PRICE.DBF
Number of data records : 405
Date of last update : 7/20/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	PRICE_CODE	Character	7	
3	QMIN1	Numeric	5	
4	QMAX1	Numeric	5	
5	PRICE1	Numeric	10	4
6	QMIN2	Numeric	5	
7	QMAX2	Numeric	5	
8	PRICE2	Numeric	10	4
9	QMIN3	Numeric	5	
10	QMAX3	Numeric	5	
11	PRICE3	Numeric	10	4
12	QMIN4	Numeric	5	
13	QMAX4	Numeric	5	
14	PRICE4	Numeric	10	4
15	QMIN5	Numeric	5	
16	QMAX5	Numeric	5	
17	PRICE5	Numeric	10	4
18	QMIN6	Numeric	5	
19	QMAX6	Numeric	5	
20	PRICE6	Numeric	10	4
21	QMIN7	Numeric	5	
22	QMAX7	Numeric	5	
23	PRICE7	Numeric	10	4
24	QMIN8	Numeric	5	
25	QMAX8	Numeric	5	
26	PRICE8	Numeric	10	4
27	QMIN9	Numeric	5	
28	QMAX9	Numeric	5	
29	PRICE9	Numeric	10	4
30	QMIN10	Numeric	5	
31	QMAX10	Numeric	5	
32	PRICE10	Numeric	10	4
** Total **			213	

Used by: SELCTVEN.PRG

Structure for database : CDCF.DBF
Number of data records : 9458
Date of last update : 7/ 9/91

Field	Field name	Type	Width	Dec
1	NSN	Character	16	
2	CAGE	Character	5	
3	DISC_CODE	Character	2	
4	DISC	Character	56	
5	CAUSE_CODE	Character	2	
6	CAUSE	Character	56	
7	DISP_CODE	Character	2	
8	DISP	Character	56	
9	CORR_CODE	Character	2	
10	CORR	Character	56	
** Total **			254	

Used by: SCREENS.PRG
Used by: PREPVEN.PRG

Structure for database : QUALITY.DBF
Number of data records : 24
Date of last update : 7/19/91

Field	Field name	Type	Width	Dec
1	CAGE	Character	5	
2	VENDOR	Character	25	
3	FSC	Numeric	4	
** Total **			35	

Used by: PREPVEN.PRG

Structure for database : HISTORY.DBF
Number of data records : 117
Date of last update : 7/20/91

Field	Field name	Type	Width	Dec
1	NSN	Character	16	
2	DATE	Character	5	
3	CAGE	Character	5	
4	PRICE	Numeric	6	2
5	QUANTITY	Numeric	5	
** Total **			38	

Used by: PREPVEN.PRG

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Data Dictionary
Date: 7/30/91
Time: 11:38

Field Name	Type	Len	Dec	Database
ADDED	C	8	0	DCRL.DBF
ADDRESS1	C	35	0	VENDOR.DBF
ADDRESS2	C	35	0	VENDOR.DBF
ADDRESS3	C	35	0	VENDOR.DBF
ADDRESS4	C	35	0	VENDOR.DBF
ALT	N	3	0	MODEL.DBF
CAGE	C	5	0	PR_TEMP.DBF
				HOLD.DBF
				NSN.DBF
				VENDOR.DBF
				DCRL.DBF
				CDCF.DBF
				PRICE.DBF
				QUALITY.DBF
				HISTORY.DBF
CATEGORY1	C	15	0	DCRL.DBF
CATEGORY2	C	15	0	DCRL.DBF
CATEGORY3	C	15	0	DCRL.DBF
CATEGORY4	C	15	0	DCRL.DBF
CATEGORY5	C	15	0	DCRL.DBF
CATEGORY6	C	15	0	DCRL.DBF
CAUSE	C	56	0	CDCF.DBF
CAUSE_CODE	C	2	0	CDCF.DBF
CDCF	L	1	0	PR_TEMP.DBF
CHANGED	C	8	0	DCRL.DBF
CODE	C	1	0	DCRLCODE.DBF
CORR	C	56	0	CDCF.DBF
CORR_CODE	C	2	0	CDCF.DBF
DATE	C	5	0	HISTORY.DBF
DATE1ATE1	C	8	0	DCRL.DBF
DATE2	C	8	0	DCRL.DBF
DATE3	C	8	0	DCRL.DBF
DATE4	C	8	0	DCRL.DBF
DATE5	C	8	0	DCRL.DBF
DATE6	C	8	0	DCRL.DBF
DAYS	N	2	0	VENDOR.DBF
DELIVERY	N	3	0	VENDOR.DBF
DEL_IND	C	7	0	DCRL.DBF
DISC	N	5	3	VENDOR.DBF
DISC	C	56	0	CDCF.DBF
DISC_CODE	C	2	0	CDCF.DBF

DISP	C	56	0	CDCF.DBF
DISP_CODE	C	2	0	CDCF.DBF
EXT_PRICE	N	8	2	HOLD.DBF
FOB	C	1	0	VENDOR.DBF
FSC	N	4	0	QUALITY.DBF
HISTORY	L	1	0	PR_TEMP.DBF
HISTORY1	N	2	0	MODEL.DBF
HISTORY2	N	7	2	MODEL.DBF
INSPECT	N	6	2	VENDOR.DBF
LOT_SIZE	N	3	0	VENDOR.DBF
LOW	N	2	0	MODEL.DBF
MIL_SPEC	C	5	0	PR_TEMP.DBF
				NSN.DBF
				VENDOR.DBF
MIN_ORDER	N	6	2	VENDOR.DBF
NAME	C	30	0	VENDOR.DBF
NAME1	C	35	0	DCRL.DBF
NAME2	C	35	0	DCRL.DBF
NAME3	C	35	0	DCRL.DBF
NAME4	C	35	0	DCRL.DBF
NET	N	2	0	VENDOR.DBF
NSN	C	16	0	NSN.DBF
				CDCF.DBF
				HISTORY.DBF
ORD_QUANT	N	5	0	HOLD.DBF
PRICE	N	6	2	HISTORY.DBF
PRICE1	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE10	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE2	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE3	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE4	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE5	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE6	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE7	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE8	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE9	N	10	4	PR_TEMP.DBF
				PRICE.DBF
PRICE_CODE	C	7	0	PR_TEMP.DBF
				NSN.DBF
				PRICE.DBF
PROB	L	1	0	PR_TEMP.DBF
QMAX1	N	5	0	PR_TEMP.DBF
				PRICE.DBF

QMAX10	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX2	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX3	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX4	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX5	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX6	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX7	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX8	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMAX9	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN1	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN10	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN2	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN3	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN4	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN5	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN6	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN7	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN8	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QMIN9	N	5	0	PR_TEMP.DBF
				PRICE.DBF
QTY_VAR_M	N	2	0	VENDOR.DBF
QTY_VAR_P	N	2	0	VENDOR.DBF
QUALITY	L	1	0	PR_TEMP.DBF
QUANTITY	N	5	0	HISTORY.DBF
REMIT1	C	35	0	VENDOR.DBF
REMIT2	C	35	0	VENDOR.DBF
REMIT3	C	35	0	VENDOR.DBF
REMIT4	C	35	0	VENDOR.DBF
RESTRICT1	C	50	0	DCRL.DBF
RESTRICT2	C	50	0	DCRL.DBF
RESTRICT3	C	50	0	DCRL.DBF
RESTRICT4	C	50	0	DCRL.DBF
RESTRICT5	C	50	0	DCRL.DBF
RFCC	C	1	0	VENDOR.DBF

SIZE_CODE	C	1	0	VENDOR.DBF
STATE	C	2	0	VENDOR.DBF
TITLE	C	25	0	DCRLCODE.DBF
U1	C	2	0	VENDOR.DBF
UNIT_PRICE	N	10	4	HOLD.DBF
UP_LIMIT	N	6	0	MODEL.DBF
VENDOR	C	25	0	QUALITY.DBF

Appendix D: Other Program Documentation

System: Automated Vendor Selection Ass tant
Author: Capt Daniel E. Hagmaier
File List
Date: 7/30/91
Time: 11:39

Programs and procedures:
- ANALZSCR--procedure
- AVSA.PRG
- AWARDSCR--procedure
- CDCFSCR--procedure
- INFO_SCR--procedure
- INITLSCR--procedure
- INPUTSCR--procedure
- NOVENSCR--procedure
- PREPVEN.PRG
- PRICESCR--procedure
- PROBMSCR--procedure
- SCREENS.PRG
- SELCTSCR--procedure
- SELCTVEN.PRG
- TITLESCR--procedure
- VENDRSCR--procedure

Procedure files:
- SCREENS.PRG

Databases:
- CDCF.DBF
- DCRL.DBF
- DCRLCODE.DBF
- HISTORY.DBF
- HOLD.DBF
- MODEL.DBF
- NSN.DBF
- PRICE.DBF
- PR_TEMP.DBF
- QUALITY.DBF
- VENDOR.DBF

Index files:
- CDCF_N_C.NDX
- DCR_CAGE.NDX
- HIST_N_D.NDX
- H_EXT_PR.NDX
- H_ORD_Q.NDX
- N_NSN.NDX
- P_C_CODE.NDX

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Index Parameter Summary
Date: 7/30/91
Time: 11:38

N_NSN.NDX -- Indexed on: NSN

Used in: SCREENS.PRG
Used in: SELCTVEN.PRG

V_C_MIL.NDX -- Indexed on: cage+mil_spec

Used in: SCREENS.PRG
Used in: SELCTVEN.PRG
Used in: PREPVEN.PRG

DCR_CAGE.NDX -- Indexed on: CAGE

Used in: SCREENS.PRG
Used in: SELCTVEN.PRG

CDCF_N_C.NDX -- Indexed on: NSN+CAGE

Used in: SCREENS.PRG
Used in: PREPVEN.PRG

P_C_CODE.NDX -- Indexed on: CAGE+PRICE_CODE

Used in: SELCTVEN.PRG

Q_CAGE.NDX -- Indexed on: CAGE

Used in: PREPVEN.PRG

H_EXT_PR.NDX -- Indexed on: ext_price

Used in: PREPVEN.PRG

H_ORD_Q.NDX -- Indexed on: ORD_QUANT

Used in: PREPVEN.PRG

HIST_N_D.NDX -- Indexed on: NSN+DATE

Used in: PREPVEN.PRG

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Procedures Summary
Date: 7/30/91
Time: 11:38

SCREENS.PRG
   Contains: TITLESCR
      Called by: AVSA.PRG
   Contains: INFO_SCR
      Called by: AVSA.PRG
   Contains: INPUTSCR
      Called by: AVSA.PRG
   Contains: SELCTSCR
      Called by: AVSA.PRG
   Contains: NOVENSCR
      Called by: AVSA.PRG
   Contains: PRICESCR
      Called by: AVSA.PRG
   Contains: VENDRSCR
      Called by: AVSA.PRG
   Contains: ANALZSCR
      Called by: AVSA.PRG
   Contains: INITLSCR
      Called by: AVSA.PRG
   Contains: PROBMSCR
      Called by: AVSA.PRG
   Contains: CDCFSCR
      Called by: AVSA.PRG
   Contains: AWARDSCR
      Called by: AVSA.PRG
   Contains: RTURN
      No calls to this procedure

System: Automated Vendor Selection Assistant
Author: Capt Daniel E. Hagmaier
Tree Diagram for databases and program files.
Date: 7/30/91
Time: 11:39

```
AVSA.PRG
     TITLESCR--procedure
     INFO_SCR--procedure
     INITLSCR--procedure
     INPUTSCR--procedure
     SELCTSCR--procedure
     SELCTVEN.PRG
       -->PR_TEMP.DBF
       -->NSN.DBF
       -->PRICE.DBF
       -->DCRL.DBF
       -->VENDOR.DBF
     ANALZSCR--procedure
     PREPVEN.PRG
       -->CDCF.DBF
       -->QUALITY.DBF
       -->VENDOR.DBF
       -->HOLD.DBF
       -->MODEL.DBF
       -->HISTORY.DBF
     PRICESCR--procedure
     VENDRSCR--procedure
     CDCFSCR--procedure
     PROBMSCR--procedure
     AWARDSCR--procedure
     NOVENSCR--procedure
```

Appendix E: Variable Descriptions

Variable Name	Type	Description
ANALZSCR	Procedure name.	Informs the user of program status
AWARDSCR	Procedure name.	Displays the user Award screen
B	dBase work area.	
C	dBase work area.	
CDCFSCR	Procedure name.	Displaying CDCF information
CDCF_N_C	CDCF index file.	Indexed on NSN and Cage
D	dBase work area.	
DCRL	Data file.	Contains problem vendor info.
DCRLCODE	Data file.	Contains DCRL code descriptions
DCR_CAGE	DCRL index file.	Indexed on Cage code
HIST_N_D	History index file.	Indexed on NSN and Date
HOLD	Data file.	Temporary, holding price info
H_EXT_PR	Hold index file.	Indexed on extended price
H_ORD_Q	Hold index file.	Indexed on order quantity
INFO_SCR	Procedure name.	Program information screen
INITLSCR	Procedure name.	Informs the user of program status
INPUTSCR	Procedure name.	Controls the user input screens
MCAGE	Memory variable.	Contains cage code
MCAGE1	Memory variable.	Contains first cage in mem matrix
MCELL	Program pointer.	Used to point to current cell
MCHOICE	Memory variable.	Contains user's response
MCOL	Program pointer.	Tracks the current matrix column
MCOLUMN	Program pointer.	Tracks memory matrix column
MCONTINUE	Program flag.	Controls internal looping
MCOUNT	Counter variable.	Controls matrix development
MCOUNTER	Counter variable.	Controls matrix development
MCURRENT	Memory variable.	Contains current time
MDAY	Memory variable.	Numeric value of today's date
MDELIVERY	Memory variable.	The total days for vendor delivery
MEND	Program flag.	Controls internal looping

MEP	Memory variable.	Contains extended price
MEXT_1_1	Matrix variable.	Extended price, row 1, column 1
MEXT_PRICE	Memory variable.	Contains extended price
MFLAG	Program flag.	General purpose control
MHISTORY1	Program flag.	Controls Price Exceeds Hist. msg.
MHISTORY2	Program flag.	Controls 'No Hist. On File' msg.
MHIST_CAGE	Memory variable.	Most recent vendor contracted.
MHIST_DATE	Memory variable.	Most recent purchase date.
MHIST_PR	Memory variable.	Most recent purchase price.
MJ_DATE	Memory variable.	Contains the Julian date.
MLAST_ORD	Memory variable.	Use for matrix development.
MLEAP_YR	Program flag.	Set if current year is leap year.
MLINE	Memory variable.	Counter for matrix development.
MLOW	Program flag.	Controls Price May Be To Low flag.
MLOW_PRICE	Memory variable.	Contains lowest purchase price.
MMAX	Memory variable.	Contains 'QMAXn' for matrix.
MMIN	Memory variable.	Contains 'QMINn' for matrix.
MMONTH	Memory variable.	Contains current month.
MNET_PRICE	Memory variable.	Contains net price for display.
MNEW_NSN	Program flag.	Cleared while NSN current.
MNEXTCOL	Program flag.	Controls search for vendor price.
MNSN	Memory variable.	Contains the current NSN.
MODEL	Data file name.	
MORDER	Memory variable.	Contains the value of 'ORDERn'
MORDER1	Memory variable.	Quantity of column 1 in matrix.
MORDER2	Memory variable.	Quantity of column 2 in matrix.
MORDER3	Memory variable.	Quantity of column 3 in matrix.
MORDER4	Memory variable.	Quantity of column 4 in matrix.
MORDER5	Memory variable.	Quantity of column 5 in matrix.
MORDER6	Memory variable.	Quantity of column 6 in matrix.
MPRICE	Memory variable.	Contains displayed extended price

MQUANT	Memory variable.	Contains quantity being sold.
MQUANTITY	Memory variable.	Amount requested by user.
MRDD	Memory variable.	Contains required delivery date.
MRETURN	Memory variable.	Contains the last viewed screen.
MROW	Memory counter.	Tracks the current matrix row.
MSERIES	Program counter.	Tracks the vendors price breaks.
MSETASIDE	Program flag.	Set if procurement is Set-A-Side.
MSTOP	Program timer.	Time which warning messages end.
MTIME	Program timer.	Current system time.
MUNITS	Memory variable.	Contains number of lots required.
MUNIT_PR	Program flag.	Controls pricing screen.
MVALID	Program flag.	Set when NSN is in the data file.
MVARIATION	Program flag.	Controls display of 'Exceeds' msg.
MYEAR	Memory variable.	Contains current year.
NOVENSCR	Procedure name.	Displays 'No vendor available'.
N_NSN	Index file.	Used by NSN, indexed on NSN.
PREPVEN	Procedure file.	*Prepares vendor* data for display.
PRICESCR	Procedure name.	Displays unit and extended prices.
PROBMSCR	Procedure name.	Displays problem vendor info.
PR_TEMP	Data file.	Contains temp vendors and $ data.
P_C_CODE	Price index file.	Indexed on cage code.
Q_CAGE	Quality index file.	Indexed on cage code.
RTURN	Procedure name.	Called when escape key pressed.
SCREENS	Procedure file.	Contains screen display programs
SELCTSCR	Procedure name.	Informs user of program status.
SELCTVEN	Procedure file.	Selects bidding vendors.
TITLESCR	Procedure name.	Displays title screen.
VENDRSCR	Procedure name.	Display vendor info screen.
V_C_MIL	Vendor index file.	indexed on cage, mil_spec.

Appendix F: Data Field Descriptions

Field Name	Database	Description
ADDED	DCRL.DBF	Date when vendor added to the DCRL file
ADDRESS1	VENDOR.DBF	First line of vendors business address
ADDRESS2	VENDOR.DBF	Second line of vendors business address
ADDRESS3	VENDOR.DBF	Third line of vendors business address
ADDRESS4	VENDOR.DBF	Forth line of vendors business address
ALT	MODEL.DBF	Administrative Lead Time for award paper work
CAGE	PR_TEMP.DBF HOLD.DBF NSN.DBF VENDOR.DBF DCRL.DBF CDCF.DBF PRICE.DBF QUALITY.DBF HISTORY.DBF	Vendors cage code, unique to each vendor
CATEGORY1	DCRL.DBF	First vendor problem
CATEGORY2	DCRL.DBF	Second vendor problem
CATEGORY3	DCRL.DBF	Third vendor problem
CATEGORY4	DCRL.DBF	Forth vendor problem
CATEGORY5	DCRL.DBF	Fifth vendor problem
CATEGORY6	DCRL.DBF	Sixth vendor problem
CAUSE	CDCF.DBF	Reason for discrepancy
CAUSE_CODE	CDCF.DBF	Code identifying discrepancy
CDCF	PR_TEMP.DBF	Flag set if vendor found in CDCF file
CHANGED	DCRL.DBF	Date the record was updated
CODE	DCRLCODE.DBF	The code letters found in the DCRL file
CORR	CDCF.DBF	Correction description
CORR_CODE	CDCF.DBF	Correction Code
DATE	HISTORY.DBF	Julian date of item purchase
DATE1	DCRL.DBF	Date first vendor problem entered
DATE2	DCRL.DBF	Date second vendor problem entered
DATE3	DCRL.DBF	Date third vendor problem entered
DATE4	DCRL.DBF	Date fourth vendor problem entered

DATE5	DCRL.DBF	Date fifth vendor problem entered
DATE6	DCRL.DBF	Date sixth vendor problem entered
DAYS	VENDOR.DBF	Number of days to qualify for payment discount
DELIVERY	VENDOR.DBF	Number of days to deliver an order
DEL_IND	DCRL.DBF	
DISC	VENDOR.DBF	Discount offered prompt Payment
DISC	CDCF.DBF	Discrepancy description
DISC_CODE	CDCF.DBF	Discrepancy code
DISP	CDCF.DBF	Disposition description
DISP_CODE	CDCF.DBF	Disposition code
EXT_PRICE	HOLD.DBF	Extended price of a quantity of product
FOB	VENDOR.DBF	Vendor identified FOB point
FSC	QUALITY.DBF	Federal Stock Class vendor qualified on
HISTORY	PR_TEMP.DBF	Flag identifying historical problems
HISTORY1	MODEL.DBF	Limit current price can exceed historical price
HISTORY2	MODEL.DBF	Price limit for item not on file
INSPECT	VENDOR.DBF	Reserved for vendor inspection informa tion
LOT_SIZE	VENDOR.DBF	Purchase requirements
LOW	MODEL.DBF	Controls the low price highlight
MIL_SPEC	PR_TEMP.DBF NSN.DBF VENDOR.DBF	MilSpec of item
MIN_ORDER	VENDOR.DBF	Dollar amount of vendor minimum order
NAME	VENDOR.DBF	Name of vendor
NAME1	DCRL.DBF	First line of vendor address
NAME2	DCRL.DBF	Second line of vendor address
NAME3	DCRL.DBF	Third line of vendor address
NAME4	DCRL.DBF	Fourth line of vendor address
NET	VENDOR.DBF	Number of days payment is due to the vendor

NSN	NSN.DBF CDCF.DBF HISTORY.DBF	NSN of the item
ORD_QUANT	HOLD.DBF	Quantity of product being analyzed
PRICE	HISTORY.DBF	Previous purchase price of item
PRICE1	PR_TEMP.DBF PRICE.DBF	Extended price of first price block
PRICE10	PR_TEMP.DBF PRICE.DBF	Extended price of tenth price block
PRICE2	PR_TEMP.DBF PRICE.DBF	Extended price of second price block
PRICE3	PR_TEMP.DBF PRICE.DBF	Extended price of third price block
PRICE4	PR_TEMP.DBF PRICE.DBF	Extended price of forth price block
PRICE5	PR_TEMP.DBF PRICE.DBF	Extended price of fifth price block
PRICE6	PR_TEMP.DBF PRICE.DBF	Extended price of sixth price block
PRICE7	PR_TEMP.DBF PRICE.DBF	Extended price of seventh price block
PRICE8	PR_TEMP.DBF PRICE.DBF	Extended price of eighth price block
PRICE9	PR_TEMP.DBF PRICE.DBF	Extended price of ninth price block
PRICE_CODE	PR_TEMP.DBF NSN.DBF PRICE.DBF	Code linking an item to a price group
PROB	PR_TEMP.DBF	Flag set if vendor found in DCRL file
QMAX1	PR_TEMP.DBF PRICE.DBF	Max Purchase quantity for price block one
QMAX10	PR_TEMP.DBF PRICE.DBF	Max Purchase quantity for price block ten
QMAX2	PR_TEMP.DBF PRICE.DBF	Max Purchase quantity for price block two
QMAX3	PR_TEMP.DBF PRICE.DBF	Max Purchase qnty for price block three
QMAX4	PR_TEMP.DBF PRICE.DBF	Max Purchase qnty for price block four
QMAX5	PR_TEMP.DBF PRICE.DBF	Max Purchase qnty for price block five

QMAX6	PR_TEMP.DBF PRICE.DBF	Max Purchase quantity for price block six
QMAX7	PR_TEMP.DBF PRICE.DBF	Max Purchase qnty for price block seven
QMAX8	PR_TEMP.DBF PRICE.DBF	Max Purchase qnty for price block eight
QMAX9	PR_TEMP.DBF PRICE.DBF	Maximum Purchase quantity for price block nine
QMIN1	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block one
QMIN10	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block ten
QMIN2	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block two
QMIN3	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block three
QMIN4	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block four
QMIN5	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block five
QMIN6	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block six
QMIN7	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block seven
QMIN8	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block eight
QMIN9	PR_TEMP.DBF PRICE.DBF	Minimum Purchase quantity for price block nine
QTY_VAR_M	VENDOR.DBF	Percent the vendor can ship under requested amt
QTY_VAR_P	VENDOR.DBF	Percent the vendor can ship over request ed amt
QUALITY	PR_TEMP.DBF	Flag indicating vendor was found in Quality file
QUANTITY	HISTORY.DBF	Number of items purchased
REMIT1	VENDOR.DBF	Vendor's billing address, line 1
REMIT2	VENDOR.DBF	Vendor's billing address, line 2
REMIT3	VENDOR.DBF	Vendor's billing address, line 3
REMIT4	VENDOR.DBF	Vendor's billing address, line 4
RESTRICT1	DCRL.DBF	First line of vendor restrictions

RESTRICT2	DCRL.DBF	Second line of vendor restrictions
RESTRICT3	DCRL.DBF	Third line of vendor restrictions
RESTRICT4	DCRL.DBF	Forth line of vendor restrictions
RESTRICT5	DCRL.DBF	Fifth line of vendor restrictions
RFCC	VENDOR.DBF	RFCC code used by the vendor
SIZE_CODE	VENDOR.DBF	Code indicating vendor's status (See DESC Form 800)
STATE	VENDOR.DBF	Government state code for vendor's resid ance
TITLE	DCRLCODE.DBF	Long description of DCRL codes
UI	VENDOR.DBF	Unit of issue
UNIT_PRICE	HOLD.DBF	Unit price of an item
UP_LIMIT	MODEL.DBF	Maximum amount of small contract awards
VENDOR	QUALITY.DBF	Cage code of vendor

Appendix G: Questionnaire Responses

PANEL QUESTIONNAIRE RESPONSES

1. Describe any problems you incurred while using the system?

   - None

2. What information presented by the system, if any, is irrelevant to the award selection process?

   - Required delivery, although not irrelevant, is not looked at as closely as low bidder.

   - The input of a RDD date; awards are not usually based on this.

3. What other information should the system provide to aid in the award process?

   - Should provide quantity in past procurement history. This has a direct bearing on award process when making total comparison of unit prices that exceed 10%.

   - Designate vendors who have minimum by quantities. Add quantity purchased in last buy block info.

4. Do you have any suggestions for future enhancements to this system?

   - None

5. Do you have any other comments or suggestions regarding the design or usefulness of this system?

   - It certainly saves time and effort. The overall view of the extension screens is great. Program is very well written. Computer instructions are easy to follow.

- It will be very beneficial and useful to all buyers using price lists as we now have. The major concern would be pricing updates and how they would be done.

6. As presented today, does the system assist the buyer in the vendor selection process?

   - Yes, definitely.

   - Yes.

BUYER QUESTIONNAIRE RESPONSES

1. Describe any problems you incurred while using the system?

- None.
- None.
- So far, none.
- None.
- None.
- One was hitting a wrong key which put me "back" temporarily on a couple of PR's. Also, noticed that QPL (Qualified Sources) sources were not indicated and noticed that there was no indication that government source inspection was acceptable to the contractor(s) for supplying the parts.
- Need P.O.C. & phone numbers for the contractors. At award step, need a definitive key that restarts the system due to accidently hitting a key, besides "P", and not being finished. Switching from U screen to E screen comparing low bid to delivery, the U screen should have the info on meeting RDD also.
- None.
- N/A.
- None.
- None.
- None.
- Didn't have any problems.

2. What information presented by the system, if any, is irrelevant to the award selection process?

- None.

- None.

- No Change.

- Request for Required Delivery Date (RDD). Not that the RDD is not important. I just don't think we use it to determine the awardee over another, under normal situations.

- None.

- Although delivery is important, I think RDD info. is not that relevant to this situation. If delivery is urgent, would not be bought as price listed item.

- None.

- Thought all the information was relevant.

- Set-aside, if a large business is low, the set-aside should be dissolved and not continued.

- None.

- None.

- None.

- None.

- The RDD has not been a priority when deciding what contractor receives the award. Delivery is important however, price is mostly the determining factor. This is not irrelevant information just over emphasized in the system.

3. What other information should the system provide to aid in the award process?

- DCAS information.

- More infor. on past history.

- Combined PR.

- None.

- Last buy qty.

- FOB origin should designate city & state.

- Pre-award information - Are there any problems with a certain vendor(s) - They should be identified.

  Technical information - The QPL items should have the qualified sources identified along with the current QPI info. (specs). In addition, the system could indicat whether a contractor(s) has accepted government source inspection (Y or N) and if there is any lot charge associated.

  Packaging information - The system should indicate whether a certain company can comply with Mil packaging requirements & bar coding requirements & whether it is done at its facility or farmed out. (Important if GSI is implemented).

- When several P/N's are acceptable for a particular NSN, how do the buyers know which part dealers are quoting. Same is true of MIL-SPEC items; the sample PR's used dealers as vendors given. To write up the award buyers need to know which mfg. they are quoting.

- P.O.C. for contractors. Phone numbers. If there is alternate bids that due to dollar savings should be evaluated, to enhance competition. How long are the quotes valid for.

- RFQ s have other requirements than NSN, Qty, & delivery date req'd - specifically: 1) FOB point request, 2) inspection & acceptance point, 3) packaging & marking reqmt's

all vary. Also, we must know (if awarding to dealers) whose mfg part will be supplied, as there is a great possibility that more than one mfgr. is approved.

- Prices if place of Inspection and Acceptance is Origin, as well as the U/P if the place of inspection is destination. Phone # and contact point for each vendor. The inclusion of the Contractor's/Vendor's phone # might help aid the buyer if he/she needs to contact C/V for any reason. Could be included with address of vendor (screen).

- U/P w/GSI - if there's a charge for GSI. Where insp-/accep is to be performed (i.e. contractor's plant, pkgr's plant, name and address of pkgr). Previous buy - "Last Purchased On . . . ." should include qty and P.O./Contract #.

- Somehow interaction time between the contract specialist and the contractor must be accounted for in the system as well as time spent for inner office communications between the buyer and item manager or technician. The buyer needs some type of authority to change for example FOB point/inspection qty variance to tailor each quote to each award.

- One of the QPL source price list was not written on abstract. All of the vendors should have been on there.

4. Do you have any suggestions for future enhancements to this system?

- Not at this time.

- Not yet.

- None.

- No.

- Yes. A company's certs and reps could be input for those buys between $10,000 and $25,000 by company officials when they "feed DESC price/del." info.

- Need P.O.C. from contractors. Quote expiration date. Have a definitive key that will restart the system at award stage in lieu of just hitting any key besides "P". If low bidder on U screen does not meet RDD there should be a method of annotating such in lieu of needing to go to the E screen.

- Whose mfg. part will be supplied, as there is a great possibility that more that one mfgr. is approved.

- The screen that shows last buy info might state the quantity bought (as of now it just states when the last buy was and the price paid).

- Have Form 800 in the system where information can be transferred and then print Form 800.

- As described in previous block, it would be beneficial for the buyer to be able to make unilateral changes to only change certain contents of the contractor's quotation so the terms of each quotation would apply and serve the governments needs i.e. INSP/ACC pt, qty variance.

- No, I really don't know that much about it yet.

5. Do you have any other comments or suggestions regarding the design or usefulness of this system?

- No.

- This will be a good thing to have.

- Not yet.

- This system is a leadtime saver. It deletes the solicitation leadtime and enhances the award process all at the same time.

- Great idea. Very helpful.

- It's a <u>great</u> improvement over price lists.

- The usefulness of the system is very good and has numerous possibilities. Good idea!

- Think the system could be very useful to buyers in most cases. Is there a way to include low offerers quoting alternate part numbers, which happens every so often. The price screens showed min buy qty for several vendors, how would minimum dollar amount per line item be reflected.

- Would be excellent for price listed QPL's, however due to the nature of the beast (shady contractors & reps) these should be followed up in some manner so that there would be some written backup to ensure the quoter's could not repeatedly claim typo errors which would in turn create a nightmare for the post-award personnel. Could make these all bilateral contracts/purchase orders since we would have it all in the computer system and the time involved would be offset by the results of the written, legal, obligations.

- Very useful tool! This would greatly decrease PALT.

- Could be very useful - depends on how often computer is up-and-running.

- Having three years experience with DPACS (DPACS is a step up from manual buying when it works) and only a short time with the Automated Vendor Selection Assistant,

it looks like DPACS may have some competition. Hope this system works.

- I think it is a step in the right direction, however it will take time to improve and perfect. Their must be a way to monitor the accuracy of the user as well as an allotment built in to the system for the time spent for the extra steps and unique situations that arise on each procurement. Their must also be allowances made for computer down time. Overall at least at its inception this process needs to be monitored closely by management to assure fairness.

- The program seems to be really easy. And that helped a lot. The program could be very useful because could save on the buyer' time & mind. Manual written QPL'a are very boring and monotonous.

USER'S GUIDE

For The

AUTOMATED VENDOR SELECTION ASSISTANT

Software

Table Of Contents

Disclaimer ii

Introduction 1

System Requirements 2

Software Installation 3

Starting AVSA 4

User Screens 5
- Welcome Screen 6
- Program Information Screen 7
- Initializing Screen 8
- NSN Screen 9
- Quantity Screen 11
- RDD Screen 12
- Set-A-Side Screen 13
- Selecting Vendors Screen 14
- No Qualified Vendor Screen 15
- Analyzing Vendor Screen 16
- Organizing Data Screen 17
- Unit Pricing Screen 18
- Extended Pricing Screen 20
- Vendor Information Screen 21
- DCRL Screen 23
- CDCF Screen 24
- Award Screen 26

Exiting The System 27

The Model Component 28

Disclaimer

This software was developed for the exclusive use of the Defence Electronic Supply Center (DESC), for demonstration poroses only. It, in its current configuration, is not intended to be used in the in the actual vendor selection decision process. The user assumes responsibility of any such employment.

Introduction

This guide is designed to pilot the user through the use of the Automated Vendor Selection Assistant Program. The software, at the time of this writing, is not intended for use as a stand alone program. Its purpose was to establish validity for the concepts presented in the VASPP program. As a result, links to the actual supporting data files have been simulated.

As the VASPP system evolves, it is envisioned only the ideas generated by this prototype will survive. The tasks accomplished by the Automated Vendor Selection Assistant software are a subset of those required by VASPP. As such, it is expected this code will be re-written in the native language of VASPP, when that stage of VASPP development is reached.

System Requirements

The Automated Vendor Selection Assistant program is designed to run on a stand alone personal computer system. It was developed on a 286, AT class machine. A hard disk drive is required. The supporting program and data files consume six megabytes of disk space. In addition to the program files, *dBase III Plus*, must reside on the system. A color monitor is recommended, but not required. There are no provisions in the system to produce printed images, therefor a line printer is not required.

Due to the system dependance on data files for information, performance of the hard drive will directly affect software performance. As such, it is suggested a 'File Defragmation' utility be used on the hard drive before installing the software.

Software Installation

The Automated vendor selection Assistant contains program and database files. They should be installed in their own directory, on the same disk drive that *dBase* resides. There is no requirement to keep the program files separate from the data files. See your DOS manual for information on creating subdirectories and copying files.

Starting AVSA

The Automated Vendor Selection Assistant (AVSA) must run in conjunction with *dBase III Plus*. Earlier versions of *dBase* are incompatible. To begin program execution, *dBase* must first be running on the computer system. Please refer to your program manual for instructions regarding the installation and operating of *dbase III Plus*.

At the *dbase* dot prompt, the following command need to be entered:

SET PATH TO useroption

where user option is the full directory path to the AVSA files. For example, if the files are stored in the subdirectory 'AVSA' on disk drive 'C', the command would be entered as follows:

SET PATH TO C:\AVSA

With the path set, AVSA can be started. To start AVSA, the command;

DO AVSA

is entered. This will bring up the welcome screen. The following pages will describe the program operation.

User's Screens

Welcome Screen

```
Welcome To The

A U T O M A T E D
V E N D O R
S E L E C T I O N
A S S I S T A N T

Beta Version 2.5

Press Any Key To Continue
```

This is the opening screen providing program identification. The user strikes any key to proceed to the next screen.

Program Information Screen

```
The Automated Vendor Selection Assistant
selects the vendor(s) who have competitively bid
on the item of interest.

To proceed, you must know the item's NSN
and the quantity required.

Do you wish to continue? <Y/N>
```

This screen explains the purpose and identifies the information required from the user for successful program execution. The user decides whether to continue on into the program or terminate and return to the DOS prompt.

Initializing Screen

```
Initializing The System
```

This is a program status screen. After the user informs the system to proceed, this screen is displayed while memory variables are being initialized. It will appear briefly prior to entering the NSN each time. The duration that this screen is displayed is dependant upon the speed of the computer system.

NSN Screen

```
Enter the NSN of the item to be procured

5905-01-009-5543

(Press <CR> when complete)

Press <ESC><ESC> to Quit the Assistant
```

This is the first of the input screens. Prompts for information are presented sequentially. The first item requested is the NSN. The user enters the thirteen digits, the system supplies the '-'. Only numerics are accepted. The user can use the arrow keys to make corrections in the entry. When the NSN is complete, pressing a carriage return <CR> enters it into the system. The system then checks to see if the NSN matches an entry in the pricing data file. If no match is found, a warning message is displayed and the user is allowed to re-enter the NSN. If the NSN is on file, the system prompts the user for the quantity required.

To terminate the entry, the escape key <ESC> is pressed twice. The entry can be terminated at any time while the user is imputing the NSN. If the user elects to end

the session prematurely, the system resets to the Program Information Screen. At the information screen, the user can start another inquiry or exit to the *dBase* prompt.

Quantity Screen

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90    EA.

(Press <CR> when complete)

Enter <0><CR> To Quit
```

After a NSN is entered that the system recognizes, the user is prompted for the number of item required. Only numerics entries are accepted. When the correct value is entered, the user presses the carriage return <CR>. If the user wishes to end the session prematurely, a zero <0> may be entered. Entering a zero will return the system to the Program Information Screen where the user can either restart the inquiry or exit the system entirely.

RDD Screen

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90 EA.

(Press <CR> when complete)

What is the RDD date? 92105
```

The Required Delivery Date (RDD) is the next item requested by the system. A numeric value is entered. The first two digits (in this case '92') represent the year that the items are required. The next three digits (105) indicates the day of the year the item is required. Any year is valid from one year prior to the current year to ninety-nine. Valid day entries range from zero to 365 (366 during a leap year).

This date is used a target date for vendor delivery. For more information on the use of the RDD, refer to the Vendor Detail Screen.

Set-A-Side Screen

```
Enter the NSN of the item to be procured

5905-01-009-5543

Enter the quantity required

90 EA.

(Press <CR> when complete)

Is this procurement Set-Aside for small business? <Y/N/?>
```

The system uses the information provided in this entry to exclude large vendors from consideration if a 'Y' is entered. Otherwise, all vendors bidding on the item of interest are examined. If the user is unsure if the request has been identified as a Set-A-Side, a '?' can be entered. This is functionally equivalent to entering a 'N', as all vendors bidding on the item are examined.

This is the last of the input screens. From this point, the system assumes control, displaying status screen as the processing evolves. Depending on the relative speed of the system and the size of the data files in use, processing may take from one to several minutes.

Selecting Vendors Screen

```
Selecting Vendors
```

This program status screen informs the user the program is in the process of interrogating the NSN data file looking for qualified vendors that have bid on the item of interest. Vendors that are currently in a 'DeBarred' status are removed from consideration. 'Large' vendors are also removed if the procurement is designated as 'Set-A-Side' If the system fails to locate a qualified vendor, a message to that effect is displayed. If the system finds at least one qualified vendor the program status screen is updated.

No Qualified Vendor Screen

```
No qualified vendors
are on file matching
your requirements.
```

Press Any Key To Continue

If there are no qualified vendors bidding on the item, the user is informed with the presents of the above screen. When the user presses a key, the system returns to the Program Information Screen. From this point, the user can either fail to make the award or can revise the requirements (i.e. specifying that the procurement is not limited to small vendors) and reprocess the request.

Analyzing Vendor Screen

```
Analyzing Vendor(s)
```

The system informs the user it has successfully located at least one qualified vendor for further consideration by displaying this screen. Any vendors remaining after the selection process are analyzed for past contract performance. The DCRL and Quality data files are scanned. If any irregularities are found, the system sets internal flags to display appropriate messages on the following user screens.

After reviewing the vendors background, the system focuses on pricing information. The minimum order quantity is calculated. This check considers vendor lot size and minimum order dollar amount. Price breaks for larger purchases are identified as well.

Organizing Data Screen

```
Organizing Vendors
```

Once background checks are made for each vendor, and price information is recorded, the system moves into the final phase of processing. The Organizing Vendor screen is displayed at this time. The system is preparing the data for display in the following screens.

Unit Pricing Screen

*** UNIT PRICE EXCEEDS HISTORY ***

Last Purchased On 80302
From 91637 For $0.27

Unit Pricing Data For: 5905-01-009-5543

	90	100	200	250	300	500
56856	1.4300	1.1400		1.0200		0.9000
6S313		0.9200	0.8500		0.7800	0.7700
7K545						0.2720

VENDOR: Problem Vendor Info
CDCF Vendor Info
Quality Vendor

PRICE: Price May Be To Low
Low Price

<E> Extended Pricing <A> Award Screen <C> CDCF Vendor Detail
<V> Vendor Information <Q> Quit <P> Problem Vendor Detail

The first user screen presented is the Unit Pricing Screen. This screen informs the user which vendors sell the item, and the unit price they charge for each unit. The vendor's cage code is color coded, corresponding the vendor's appearance in the DCRL data file, the CDCF file or the Quality file. The logic governing the color code assigned to each vendor has a designated order of hierarchy. The Coding for a vendor found in the CDCF file will override an appearance in the Quality file. Accordingly, an appearance in the DCRL data file will override all other color coding.

Pricing information is also color coded. The lowest total price to satisfy the purchase request is highlighted in bright green. If there is a tie between vendors, both low quotes will be highlighted. If the low price is 'considerably' lower than the next

lowest vendor's price, the low price is highlighted yellow. This feature alerts the user that the price may be unrealistic. The threshold of the low price is controlled by the model database.

Historical information regarding past procurement of the item are displayed in the upper right hand corner of the screen. This feature provides the user with not only the vendor and price of the last order, but provides a estimate regarding the rate of item consumption.

The upper left hand corner is a message area. Any irregularities identified by the system during its analysis of the data are displayed here. These are cautionary messages, alerting the user to potential problems with the requirement. (Refer to the Model Component for further explanation of these messages.)

An option menu is located at the bottom of the screen. While this screen appears on most of the user screens, the options available change, depending on the user screen currently being displayed. From the unit price screen the user may transfer to the following screens: Extended Pricing screen, Vendor Detail screen, CDCF Vendor Detail screen, Problem Vendor Detail Screen, and the Award screen, or return to the Program Information Screen.

Extended Pricing Screen

*** UNIT PRICE EXCEEDS HISTORY ***

Last Purchased On 80302
From 91637 For $0.27

Extended Pricing Data For: 5905-01-009-5543

	90	100	200	250	300	500
56856	128.70	114.00		255.00		450.00
6S313		92.00	170.00		234.00	385.00
7K545						136.00

VENDOR: Problem Vendor Info
CDCF Vendor Info
Quality Vendor

PRICE: Price May Be To Low
Low Price

<U> Unit Pricing <A> Award Screen <C> CDCF Vendor Detail
<V> Vendor Information <Q> Quit <P> Problem Vendor Detail

This is the extended pricing screen. All information contained on it is the same as for the unit pricing screen with two exceptions. First, the numbers contained in the matrix now represent the extended price information. It is calculated by taking the unit price for a given quantity and multiplying it by the quantity shown in the column headings. This results in the total purchase price for the items. The second change is in the user options section. The option for Extended Price screen has been changed to Unit Price screen.

Vendor Information Screen

Vendor Data For: 5905-01-009-5543

CAGE	VENDOR	DISC	NET	DEL	FOB	SPEC	PROB	CDCF	QUAL
56856	Vamistor Corp.		/	91356	D				
6S313	G & A Sales	1.0%	10/30	92091	O				
7K545	Hamilton Avnet Electronics		/	92121	D				

User's Options:

<U> Unit Pricing <A> Award Screen <C> CDCF Vendor Detail

<E> Extended Pricing <Q> Quit <P> Problem Vendor Detail

The vendor screen display the vendor delivery information and informs the user if the vendor was found in any of the supporting data files. On the left side of the screen, the cage code is located. Next to that is the name of the vendor. After the vender identification section, discount information is given and delivery information after that.

'DEL' is the projected delivery date. The vendor quotes a delivery time for his products. That time, in days, is added to the current Julian date. In addition to the delivery time an administrative lead time is also added. The projected delivery date is now compared to the required delivery date entered by the user at the input screen. If the projected delivery date is prior to the RDD, the delivery date is displayed in green. If the vendor cannot meet the RDD, the delivery date is displayed in red.

The final section of the vendor information display area identifies what data files the vendor is reported in. 'SPEC' indicates the vendor is coded as something other than large. 'PROB' is marked if the vendor appears in the DCRL, problem vendor file. 'CDCF' reports the existence of the vendor in the Customer Depot Complaint File, and 'QUAL' identifies this vendor as being on the quality vendor list.

As before, abnormalities identified by the system are indicated in the upper left hand corner of the screen. All valid user options are indicated at the bottom of the screen.

DCRL Screen

```
                                  6S313
SECOM ELECTRONICS CORP          | 89/11/15 D  Pre-Award Survey Required
12 PROGRESS PLACE               | **/**/**
JACKSON NJ    08527-3002        | **/**/**
                                | **/**/**
                                | **/**/**
                                | **/**/**

        SEE P IOM 20 OCT 89 RE UNSATISFACTORY PERFORMANCE
        RECOMMEND DETERMINATION OF NONRESPONSIBILITY,
        PREAWARD SURVEY, DCAS ADMINISTRATION

Press any key to continue...
```

This is the DCRL screen. If a bidding vendor appears in the DCRL file, the cage code is highlighted red. To see the information contained in the file, the user enters 'P' from any of the user screens and the discrepancy details for that vendor appears on a this screen. This screen displays all the information on file for that vendor. When the user finishes reviewing the data, striking any key will return the program to the previous screen.

CDCF Screen

```
6S313

DISC --> Q5
CAUSE -> CN CONTRACTOR NONCOMPLIANCE (PRIME CONTRACTOR)
DISP --> AD DALE - CAT I - DAC FROM C/C "K" TO C/C "H" W/MGMT CODE "
CORR --> AO POC BETTY GEBELE/OSIB/AV986-6486.
```

```
Press any key to continue...
```

The Customer Depot Complaint File is a listing by NSN of items that have had complaints registered. The complaint can be anything from substandard product performance, to mis-marked packaging. The software incorporates this data file using the following method. First, the system checks for the existence of the NSN in the CDCF data file. If the NSN exists, a search is conducted within the NSN for a cage code matching any of the bidding vendors. If a bidding vendor is found to have a complaint filed on the product in question, the system color codes the vendors cage code in the display screens. There will also be a mark in the 'CDCF' column for that vendor on the Vendor Information screen.

By selecting 'C' from the options menu the user can call up the above screen for the affected vendors. Pressing any key will continue to call up multiple entries. When

all information has been displayed, the system return the user to the screen that the user entered the 'C' option.

Award Screen

Award Information For: 5905-01-009-5543

Vendor:	Remit To:
G & A Sales 2854 Blue Rock Road Cincinnati, Ohio 45239	Same

Cage: 6S313 State Code: 39 Source Type: A

Discount: 1.000% In 10 Days Variance: + 0% - 2%

Delivery Time: 220 Days FOB: O RFCC Code: Z

	90	100	200	250	300	500
Unit Price		0.9200	0.8500		0.7800	0.7700
Ext. Price		92.00	170.00		234.00	385.00

Press <P> For Previous Screen
Any Other Key When Finished

The final screen available to the user is the Award screen. Through the use of the other screens, the user makes a determination as to which vendor should receive the contract award. Once the determination is made, entering an 'A' for the user option allows the user to enter the cage code of the vendor receiving the award. Once entered, the system displays this screen. On it, is all the information the user needs to complete the DESC Form 800. This includes the vendors business and billing addresses, vendor type code, discount information, delivery data, and quoted prices for this vendor.

The user can either press a 'P' if he wishes to return to the information screens or any other key will return the system to the Program Information screen.

Exiting The System

The user can terminate use of the system at several points along the way. At the data entry screens two escape key presses <ESC><ESC> will interrupt execution while entering the NSN. Entering a zero Quantity will return the program to the Program Information Screen as well.

When the user advances to the information screens, entering a 'Q' from the options menu will return the system to the Program Information Screen. The program automatically sends the user to the Program Information Screen after the award screen is selected, unless told to do otherwise.

Once the user arrives at the Program Information Screen, he has the option of either entering a 'Y' and reinitializing the system for another inquiry, or a 'N' can be entered to return to the *dBase* dot prompt. Entering 'QUIT' at the *dBase* prompt will return the computer system to the DOS prompt.

The Model Component

The model data file controls how and when specified information is presented on the screen. The values contained within the model can be changed by the system administrator. For information on updating data files, consult your *dBase* reference manual. The following is controlled by the contents of the model data base.

a) Low Price Flag. This element alert the user when the vendor is quoting a price that is significantly lower than the competitors. When set, the low price will be displayed in yellow on the pricing screens.

b) No History Flag. The number stored in this element represents a dollar threshold value. If the unit price of an item exceeds this amount, and there is no historical purchase information on file, a message is printed on the output screens.

c) Exceeds History Price. The prototype compares the item's current unit price with the unit price of the item when last ordered. If the current unit price exceeds the last unit price by more that the percentage contained in this element, a message is presented to the buyer.

d) Excessive Contract Value. If the total value of the award exceeds the dollar amount stored in this element, a warning is printed on the screen informing the user the limit for small contract award has been exceeded.

e) Variation. On the price lists, the vendor identifies any variation in shipping quantity. The vendors claim authorization to ship a quantity within a stated percentage of the contract quantity. For example, a vendor may claim a variation of two percent. If the contract was written for one hundred units, the vendor could ship only ninety-eight units an still satisfy the contract. The prototype check this variation, internally increments the quantity ordered to account for the variation, and computes the resulting award value of the contract. If the award value exceeds the excessive contract value, defined above, a warning is displayed on the user screens.

Bibliography

1. Allen, Maj Mary K. *The Development of an Artificial Intelligence System for Inventory Management*. PhD Dissertation. The Ohio State University, Columbus OH, 1986.

2. Allen, Mary Kay and Margaret A. Emmelhainz. *Decision Support Systems: An Innovative Aid To Managers*. Class handout distributed in LOGM 615, Logistics Decision Support Systems. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, Spring Quarter 1991.

3. Allen, Mary Kay, Ph.D. and Omar Keith Helferich, Ph.D. *Putting Expert Systems to Work in Logistics*. Oak Brook IL: Council of Logistics Management, 1990.

4. Analytical Software. *Statistix 3.5*. St. Paul: Analytical Software, June 1991.

5. Atwood, D. L. The Deputy Secretary Of Defense. *DMRD 926 Inventory Control Point Consolidation Study Report*. Memorandum. July 3, 1990.

6. Beizer, Boris. *Software System Testing and Quality Assurance*. New York: Van Nostrand Reinhold Company, 1984.

7. Belardo, Salvatore and Kirk R. Karwan. *Tools for the Development of Decision Support Systems: Prototyping, Microcomputers and O. R. Models*. as referenced in Gass, Saul I. and others. *Impacts of Microcomputers on Operations Research*. New York: Elsevier Science Publishing Company, 1986.

8. Black, W.J. *Intelligent Knowledge Based Systems: An Introduction*. Berkshire, England: Van Nostrand Reinhold (UK) Co. Ltd, 1986.

9. Corelis, Mike, DESC-PPS, Chief, Systems Branch. Personal interview. Defense Electronics Supply Center, Dayton OH, April 1991.

10. Davis, Michael W. *Applied Decision Support*. Englewood Cliffs NJ: Prentice Hall, 1988.

11. *dBase III Plus*. Ashton-Tate, 1986. Database Software Package.

12. *DESC Defense Electronics Supply Center; Gentile Station; Dayton OH*. Personnel Orientation Brochure. Winkler Company, Dayton OH, undated.

13. *DESC*. Information Pamphlet. Defense Electronics Supply Center, Gentile Station, Dayton OH, undated.

14. Emmelhainz, Maj Larry W. *Decision Support Systems: Substituting Logistics Brains For Logistics Brawn*. Class handout distributed in LOGM 615, Logistics Decision Support Systems. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, Spring Quarter 1991.

15. Emory, C. William. *Business Research Methods*. Homewood IL: Richard D. Irwin Inc., 1985.

16. Gates, William F. III, Draft briefing for DLA-P for presentation 16 April 1991. Contract Review Office, DESC-PT, Defense Electronics Supply Center, Dayton OH, 1 April 1991.

17. Ginzberg, Michael J. and Edward A. Stohr. *Decision Support Systems: Issues and Perspectives*. New York: New York University, 1982. as referenced in Ginzberg, Michael J. and others. *Decision Support Systems*. New York: North-Holland Publishing Company, 1982.

18. Hewett, Col Gary W., Director of the Directorate of Contracting and Production, DESC-P, and others. Personal interviews. Defense Electronics Supply Center, Dayton OH, April 1991.

19. Hewett, Col Gary W., Director of the Directorate of Contracting and Production, DESC-P, and others. Personal interviews. Defense Electronics Supply Center, Dayton OH, May 1991.

20. Holland, Eleanore, DESC-PPR, Chief, Policy Branch and Larry Beatly, DESC-PPS, and others. Personal interview. Defense Electronics Supply Center, Dayton OH, April 1991.

21. Holland, Eleanore, DESC-PPS, Chief, Policy Branch. Telephone interview. Defense Electronics Supply Center, Dayton OH, 29 May 1991.

22. Howden, William E. *A Survey of Dynamic Analysis Methods*. Victoria Canada: University of Victoria, as referenced in Miller, Edward and William E. Howden. *Tutorial: Software Testing and Validation Techniques*. New York: The Institute of Electrical and Electronics Engineers, 1981.

23. Howden, William E. *Empirical Studies of Software Validation*. Victoria Canada: University of Victoria, as referenced in Miller, Edward and William E. Howden. *Tutorial: Software Testing and Validation Techniques*. New York: The Institute of Electrical and Electronics Engineers, 1981.

24. Huang, J.C. *An Approach to Program Testing*. Houston: University of Houston, as referenced in Miller, Edward and William E. Howden. *Tutorial: Software Testing and Validation Techniques*. New York: The Institute of Electrical and Electronics Engineers, 1981.

25. Kennamer. Walter J. *SNAP! 1.73*. 1986. dBase Documentation Software Package.

26. McClave, James T. and P. George Benson. *Statistics For Business and Economics* (Fourth Edition). San Francisco: Dellen Publishing Company, 1988.

27. Meek, Brian et al. *Guide to Good Programming Practice*. New York: Halsted Press, 1983.

28. Office of the Secretary of Defense. *DMRD 926 Inventory Control Point Consolidation Study Report*. Final Report, July 1990, attachment to The Deputy Secretary of Defense Memorandum, dated July 3, 1990.

29. Pfeifer, Rolf and Hans-Jakob Luthi. *Decision Support Systems and Expert Systems: A Complementary Relationship?*. Zurich Switzerland: University of Zurich and Swiss Federal Institute of Technology, as referenced in Sol, Henk G. and others. *Expert Systems and Artificial Intelligence in Decision Support Systems: Proceedings of the Second Mini Euroconference, Lunteren, The Netherlands, 17-20 November 1985*. Dordrecht Holland: D. Reidel Publishing Company, 1987.

30. Sprague, Ralph H. and Eric D. Carlson. *Building Effective Decision Support Systems*. Englewood Cliffs NJ: Prentice-Hall, 1982.

31. Wray, Robert A. *dBase III Plus Programming*. Boston: Boyd & Fraser Publishing Company, 1988.

VITA

Captain Daniel E. Hagmaier was born in Warren OH on 19 January 1958. He entered Uniform Service in January 1975 by enlisting in the United States Coast Guard Reserve. Upon graduation from Bristol High School, Bristolville OH in 1976, he transitioned to the active duty force of the United States Coast Guard. From July 1976 to July 1977 he served on the Coast Guard Cutter USS Woodrush, based in Duluth MN. Upon completion of Aviation Electronic School, at Elizabeth City, NC in February 1978, he was stationed at Barber's Point NAS HI. Ground duty entailed repair of airframe and component electronic systems. Flight duties consisted of membership on Search and Rescue teams as Radioman and Navigator aboard C-130 Hercules aircraft. He was honorably discharged at the rank of E-5 in July 1980.

In May 1982, he graduated from Kent State University, Kent OH, with an Associates of Applied Science degree in *Electrical and Electronic Engineering Technology*. In May 1984, he received a Bachelor of Science degree in Industrial Technology. Captain Hagmaier was commissioned in the USAF after graduating from Officer Training School in March 1985.

He was first stationed at Headquarters, Engineering and Installation Division, Tinker AFB OK as Logistics Officer for the Secure Voice Program Management Office. In April 1988 he transferred to the 39th Tactical Group, Incirlik AB Turkey serving as Deputy Chief, Logistics Plans and Programs. He served in this position until entering the School of Systems and Logistics, Air Force Institute of Technology, in May 1990.

Permanent Address: 1825 Maplewood N.E.
Warren OH 44483

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1991	Master's Thesis

4. TITLE AND SUBTITLE
SMALL CONTRACT AWARD: IMPROVING THE VENDOR SELECTION PROCESS

5. FUNDING NUMBERS

6. AUTHOR(S)
Daniel E. Hagmaier, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GLM/LSC/91S-26

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The objective of this research was to design and validate a methodology that would enhance the productivity of the buyers who perform the vendor selection function at the Defense Electronic Supply Center. The approach utilized computer automation and software developed specifically for this application. A prototype was developed and tested. Comprised of three phases and eighteen buyers, the testing evaluated the prototype in four areas: 1) accuracy of the presented information, 2) thoroughness of the presented information, 3) ability of the user to use the presented information, and, 4) usability by inexperienced personnel. Examination of the data generated by the test phase confirms the approach used to enhance the productivity of the buyers was valid. As a result of findings from this research, the recommendations derived include the integration of this methodology into the development of future buyer assistance programs.

14. SUBJECT TERMS
Decision-Making; Artificial Intelligence; Management; Logistics Computer Program; Integrated System; Management Information Systems

15. NUMBER OF PAGES
304

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

AFIT Control Number AFIT/GLM/LSC/91S-26

AFIT RESEARCH ASSESSMENT

The purpose of this questionnaire is to determine the potential for current and future applications of AFIT thesis research. Please return completed questionnaires to: AFIT/LSC, Wright-Patterson AFB OH 45433-6583.

1. Did this research contribute to a current research project?

   a. Yes b. No

2. Do you believe this research topic is significant enough that it would have been researched (or contracted) by your organization or another agency if AFIT had not researched it?

   a. Yes b. No

3. The benefits of AFIT research can often be expressed by the equivalent value that your agency received by virtue of AFIT performing the research. Please estimate what this research would have cost in terms of manpower and/or dollars if it had been accomplished under contract or if it had been done in-house.

   Man Years _____ $_____

4. Often it is not possible to attach equivalent dollar values to research, although the results of the research may, in fact, be important. Whether or not you were able to establish an equivalent value for this research (3 above), what is your estimate of its significance?

   a. Highly Significant b. Significant c. Slightly Significant d. Of No Significance

5. Comments

_____ _____
Name and Grade Organization

_____ _____
Position or Title Address