LAW ENFORCEMENT AND VEHICLE REGISTRATION ADMINISTRATION SYSTEM

by

Henry Mark McGibbon
Mark Stephen Nault

September 1995

Thesis Advisor: Shu S. Liao
Co-Advisor: William B. Short

Approved for public release; distribution is unlimited.
REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, 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
2. REPORT DATE
   September 1995
3. REPORT TYPE AND DATES COVERED
   Master's Thesis
4. TITLE AND SUBTITLE
   LAW ENFORCEMENT AND VEHICLE REGISTRATION ADMINISTRATION SYSTEM [U]
5. FUNDING NUMBERS
6. AUTHOR(S)
   McGibbon, Henry, M. and Nault, Mark, S.
7. Performing ORGANIZATION NAME(S) AND ADDRESS(ES)
   Naval Postgraduate School
   Monterey CA 93943-5000
8. Performing ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
10. SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES
   The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.
12a. DISTRIBUTION/AVAILABILITY STATEMENT
    Approved for public release; distribution is unlimited.
12b. DISTRIBUTION CODE
13. ABSTRACT
   The Computer On-Line Police System (COPS) is a vehicle registration and ticket management system used at the Naval Postgraduate School (NPS) Security Department, originally designed by the Naval Computer and Telecommunications Station, San Diego, in 1991. A Baseline Assessment of COPS revealed the following weaknesses: severely limited query capabilities, outdated system hardware, software design errors, functionality gaps, antiquated graphical user interfaces (GUI), and no computerized data archiving capability. This thesis alleviates these deficiencies. Using the System Development Methodology (SDM), the authors will provide NPS, and potentially other Department of Defense (DoD) security forces, with a significantly improved vehicle registration database system.
   A new IS, called the Law Enforcement and Vehicle Registration Administration System (LEVRAS), was designed, programmed, and developed. The fully operational LEVRAS met all of the requirement specifications, and replaced COPS after a parallel conversion. Users were trained, and the NPS Security Department accepted the new database system for its daily operations. Fully supporting the LEVRAS lifecycle management, maintenance will be performed by the NPS Management Information System (MIS) Department.
14. SUBJECT TERMS
   Information System and Database Design
15. NUMBER OF PAGES 212
16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified
18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified
19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified
20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18 298-102
Approved for public release; distribution is unlimited.

LAW ENFORCEMENT AND VEHICLE REGISTRATION ADMINISTRATION SYSTEM

Henry M. McGibbon
Lieutenant, United States Navy
B.S., University of Utah, 1986

and

Mark Stephen Nault
Commander, United States Navy
B.S., United States Naval Academy, 1978

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN
INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
September 1995

Authors:

Henry M. McGibbon

Mark S. Nault

Approved by:

Shu S. Liao, Thesis Advisor

William B. Short, Thesis Co-Advisor

Reuben Harris, Chairman, Department of Systems Management
ABSTRACT

The Computer On-Line Police System (COPS) is a vehicle registration and ticket management system used at the Naval Postgraduate School (NPS) Security Department, which was designed by the Naval Computer and Telecommunications Station, San Diego, in 1991. COPS is an inadequate information system (IS) possessing the following weaknesses: severely limited query capabilities, outdated system hardware, software design errors, functionality gaps, antiquated graphical user interfaces (GUI), and no computerized data archiving capability. This thesis will try to alleviate these deficiencies. Using the System Development Methodology (SDM), the authors hope to provide NPS, and potentially other Department of Defense (DoD) security forces, with a significantly improved vehicle registration database system.

A Baseline Assessment of COPS verified that a new IS was necessary. A new IS, called the Law Enforcement and Vehicle Registration Administration System (LEVRAS), was designed, programmed, and developed. The fully operational LEVRAS met all of the requirement specifications, and replaced COPS after a parallel conversion was conducted. Users were trained, and the NPS Security Department accepted the new database system for its daily operations. Fully supporting the LEVRAS lifecycle, maintainance will be performed by the NPS Management Information System (MIS) Department.
### TABLE OF CONTENTS

I. INTRODUCTION ........................................................................................................... 1
   A. OBJECTIVE ........................................................................................................... 1
   B. BACKGROUND ..................................................................................................... 1
   C. SCOPE ................................................................................................................ 2
   D. METHODOLOGY .................................................................................................. 3
   E. CHAPTER OUTLINE ............................................................................................... 5

II. COMPUTER ON-LINE POLICE SYSTEM (COPS) BASELINE ASSESSMENT ........ 9
   A. EXISTING VEHICLE REGISTRATION SYSTEM .................................................. 9
   B. NEW VEHICLE REGISTRATION SYSTEM ASSESSMENT
      (SDM PHASE I - SYSTEM ANALYSIS) ............................................................... 13

III. DATABASE DEVELOPMENT PROCESS .................................................................. 19
   A. BASIC DATABASE CONCEPTS ........................................................................ 19
   B. DATABASE DEVELOPMENT METHODOLOGY ................................................. 20
   C. TECHNICAL DATABASE CONSIDERATIONS ................................................... 22
      1. Data Flow Diagrams ....................................................................................... 22
      2. Data Dictionary .............................................................................................. 24
      3. Semantic Object Modeling .......................................................................... 25
      4. Database Application Design ...................................................................... 28
      5. System Conversion and Acceptance Test .................................................... 29
      6. Documentation, Training, and Maintenance ............................................... 31

IV. SDM PHASE II - SYSTEM DESIGN ....................................................................... 33
   A. DATA FLOW DIAGRAMS .................................................................................. 33
   B. DATA DICTIONARY ............................................................................................ 38
LIST OF TABLES AND FIGURES

1. COPS Hardware, Software, and Ancillary Equipment (Table 1) .................. 9

2. Proposed Characteristics of LEVRAS vs. COPS (Table 2) ....................... 17

3. Generic Data Flow Diagram (Figure 1) ................................................. 23

4. Generic Semantic Object (Figure 2) ..................................................... 26
# LIST OF ACRONYMS, ABBREVIATIONS, AND TERMS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPS</td>
<td>Bits Per Second</td>
</tr>
<tr>
<td>CASE</td>
<td>Computer-Aided Systems Engineering</td>
</tr>
<tr>
<td>CHP</td>
<td>California Highway Patrol</td>
</tr>
<tr>
<td>CIM</td>
<td>Corporate Information Management (initiative)</td>
</tr>
<tr>
<td>CLATS</td>
<td>California Law Enforcement Terminal System</td>
</tr>
<tr>
<td>COPS</td>
<td>Computer On-Line Police System</td>
</tr>
<tr>
<td>COTS</td>
<td>Commercial-Off-The-Shelf</td>
</tr>
<tr>
<td>DBMS</td>
<td>Database Management System</td>
</tr>
<tr>
<td>DFD</td>
<td>Data Flow Diagram</td>
</tr>
<tr>
<td>DoD</td>
<td>Department of Defense</td>
</tr>
<tr>
<td>GB</td>
<td>Gigabyte</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>INLETS</td>
<td>Interstate Law Enforcement Terminal System</td>
</tr>
<tr>
<td>IS</td>
<td>Information System</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>LEVRAS</td>
<td>Law Enforcement and Vehicle Registration Administration System</td>
</tr>
<tr>
<td>MB</td>
<td>Megabyte</td>
</tr>
<tr>
<td>MHZ</td>
<td>Megahertz (processor clock speed)</td>
</tr>
<tr>
<td>MIS</td>
<td>Management Information System</td>
</tr>
<tr>
<td>NCTS</td>
<td>Naval Computer and Telecommunications Station, San Diego</td>
</tr>
<tr>
<td>NIC</td>
<td>Network Interface Card</td>
</tr>
<tr>
<td>NPS</td>
<td>Naval Postgraduate School</td>
</tr>
<tr>
<td>OJT</td>
<td>On-the-Job-Training</td>
</tr>
<tr>
<td>RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>RS</td>
<td>Requirements Specification</td>
</tr>
<tr>
<td>SDLC</td>
<td>System Development Life Cycle</td>
</tr>
<tr>
<td>SOP</td>
<td>Standard Operating Procedure</td>
</tr>
<tr>
<td>UPS</td>
<td>Uninterrupted Power Supply</td>
</tr>
<tr>
<td>VIRO</td>
<td>Vehicle Identification and Registration Office (at NPS)</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENT

The authors sincerely thank Professor Shu S. Liao and Commander William B. Short for their guidance during this thesis development. Their expeditious, yet thorough, proof reading and feedback of the following eight thesis chapters were invaluable, especially toward the completion of this thesis. The authors thank the NPS Security Officer, Mr. Gregg Caughran, and selected staff, Mr. Alex Wills and Ms. Marilyn Owens for their cooperation and assistance in the making of the Law Enforcement and Vehicle Registration Administration System (LEVRAS). Also, the authors thank Commander Robert Wilcox for his hard work during several group projects with the authors, which set the stage for this thesis. Finally, the authors would like to extend our utmost thanks to our wonderful wives and understanding children for their support during the many weekends when the dads were not available for family affairs due to thesis work.
I. INTRODUCTION

A. OBJECTIVE

This thesis reviews the existing Naval Postgraduate School (NPS) vehicle registration system. It also designs, develops, and implements a new computerized administrative system supported by a relational database. The existing system is used for vehicle registration management and traffic ticket processing. The system is inadequate due to the following weaknesses: severely limited query capabilities, outdated system hardware, software design errors and functionality gaps, antiquated graphical user interfaces (GUI), and lengthy computer processing time. Our new system improves upon or eliminates the deficiencies identified above. It also includes anti-viral protection, minimizes data entry errors, and provides for system back-ups.

B. BACKGROUND

In October 1993, students living in Naval family quarters (La Mesa Village) at NPS, Monterey, California, were concerned over alleged child abduction attempts made in the housing area by a woman in a tan colored van. The vehicle was identified as having an official Department of Defense (DoD) Registered Vehicle Sticker, along with a red enlisted sticker (possibly issued by Fort Ord) on the front left corner of the windshield. Two numbers from a Texas license plate were also identified by La Mesa residents and reported to NPS Security. NPS detectives, in turn, contacted the NPS Vehicle Identification and Registration Office (VIRO) and Fort Ord Security, and requested a list of tan colored vans with Texas license plates and registered to military enlisted members.

Neither the NPS VIRO nor Fort Ord Security could perform this simple task in a timely manner, because every enlisted vehicle registration card had to be checked by hand. The computerized information system used to store vehicular information was ineffective and unable to perform the requested query. Thus, the search for time-sensitive critical
vehicle registration information came to a virtual standstill. After a week of manually sorting through NPS and Fort Ord vehicle registration cards, the target vehicle registration card still had not been located.

In the meantime, NPS Security had received another report of a child abduction attempt. Residents of La Mesa Village were deeply troubled by the slow security force response. The vehicle was still at large due to the ineffective query process in these security database systems. Finally, after a three week intensive search, Fort Ord's database administrators found the registration card that matched the description of the vehicle. This thesis will attempt to substantially improve the ability of NPS police to respond when dealing with database queries and should bring peace-of-mind to the residents of La Mesa Village.

C. SCOPE

This thesis focuses on the comparison of the current NPS VIRO system, the Computer On-Line Police System (COPS), with a proposed alternative system, the Law Enforcement and Vehicle Registration Administration System (LEVRAS). The authors feel confident that the proposed system, LEVRAS, will be a vast improvement over COPS. We envision NPS and other military commands using LEVRAS as a standardized base/post vehicle management system via the DoD's Corporate Information Management (CIM) initiative.

Two basic administrative functions of NPS Security are vehicle and traffic ticket management. The VIRO handles incoming personnel and contractor vehicle registration for database entry, and vehicle decal or temporary pass issue. All vehicle management functions are conducted within building 211. Ticket management includes the disposition and processing of traffic tickets, which is conducted in building 200. The main functions of ticket processing include input of traffic ticket data, correlation of ticket data to the individual, and point violation assignment for traffic violation infractions (after adjudication at traffic court).
Supervisory reports from VIRO and the ticket management office are required on a "demand pull", as well as a routine basis. Weekly reports list ticket violations and decal issue and expiration from predominately newly reporting and graduating NPS students. Ad hoc reports are generated for Security Officer supervisory decision making and for NPS detectives criminal investigation work.

Intra-networking between buildings 200 and 211 includes information exchange via hardcopy paper and walking 5.25 inch floppy diskettes between the buildings. To streamline this process, the authors suggested that a local area network (LAN) be installed between the two respective workstations. This suggestion produced a work request generated by Mr. Gregg Caughran, NPS Security Officer, and was approved expeditiously. The LAN, along with the authors' new database administration system, will be developed and installed concurrently by management information system (MIS) contractors and the authors, respectively.

D. METHODOLOGY

A COPS Baseline Assessment will determine the existing architecture of COPS in terms of hardware, software and organizational structure. After the COPS Baseline Assessment is completed, the authors, working with the Security Officer, will verify that LEVRAS should replace COPS. A methodical approach will then be taken in the systematic design and construction of our product. LEVRAS system development will use the five-phase System Development Methodology (SDM), as discussed by CDR William B. Short's (1993) Introduction to Computer Management (IS-2000) classroom discussion.

The five SDM phases are:

- Phase I - System Analysis
- Phase II - System Design
- Phase III - Programming
- Phase IV - Conversion and Implementation
- Phase V - Post-Implementation.
Each phase will be developed with a quality product in mind. The Systems Analysis Phase will be carefully scrutinized to ensure that an accurate Requirements Specification (RS) document is developed. A quality RS document will help reduce future costs and errors. The System Design Phase will assist the authors in fully understanding LEVRAS requirements by developing Data Flow Diagrams (DFD's). Programming will be done with a state-of-the-art database software package to ensure production longevity in the system development life cycle (SDLC). Testing LEVRAS modules as they are being programmed will help reduce time spent debugging the completed and conglomerated module sums.

Once LEVRAS is fully developed and tested, careful consideration will be given to how to implement the conversion of COPS to LEVRAS (Conversion and Implementation Phase). As the LEVRAS system developers, we will play an integral part in the conversion process. Training will also be a factor to consider prior to, during, and after the conversion. The LEVRAS Thesis will have to be made readily available to interested parties after the authors depart NPS to answer questions pertaining to LEVRAS system development; the Dudley Knox Library will therefore maintain a copy as part of their thesis inventory. Although SDM is a methodical approach to system development, a hybrid system development approach (using a prototype system) may be implemented to ensure that LEVRAS managers and users are involved throughout the entire process. The hybrid system development approach will help correct errors in the early phases of system development which could prove costly as the development process evolves into later phases. Prototyping will also let LEVRAS managers and users express their information/database processing needs more fully.

The LEVRAS requirements definition, database design, and database application development software will use proprietary software otherwise known as Commercially-Off-the-Shelf (COTS) general-purpose software. EXCELERATOR is a requirements definition software package that will be used to develop LEVRAS DFD's. SALSA is a semantic object modeling software package that will be used to develop the LEVRAS database design.
Paradox is a software database application package that will be used to develop a user interface in a windows environment. A LEVRAS User's Manual will be developed to familiarize managers, as well as users, in system basics and detailed system procedures.

E. CHAPTER OUTLINE

The chapters of this thesis will be organized as follows:

I. Introduction. This chapter will discuss the objective, background, scope, and methodology. The objective states the main purpose for this thesis. The background discusses why the authors chose this thesis topic, in addition to the overall weaknesses of the current NPS vehicle management system. The scope focuses on the vehicle management system's functionalities and its physical layout. The methodology describes how the authors will attack the problem of system design, development, and implementation. Finally, this chapter outline section modularizes and describes each chapter in a succinct manner.

II. Computer On-Line Police System (COPS) Baseline Assessment (including SDM Phase I - System Analysis). This chapter will provide an in-depth look at the existing hardware, software, and administrative procedures used for NPS vehicle management operations. User inputs for improving current system functionalities and additional non-existing functionalities will be identified to produce a requirement specification (RS).

III. Database Development Process. This chapter will address the key areas of a MIS and its administrative data that will be manipulated to produce desired information. This process is a generic heuristic for the development of any relational database and its applications. Specifically, these key areas include: database concepts, database development methodology, requirements analysis and specifications, database design, and
programming. Finally, this chapter will close with system conversion and implementation, as well as post-implementation issues.

IV. **SDM Phase II - System Design.** This chapter will build upon the foundation addressed in Chapter III. Data requirements will be researched and subsequently established with the concurrence of the NPS Security. Data flow diagrams will display the core processes involved with the new Law Enforcement and Vehicle Registration Administration System. These DFDs will assist in identifying the data dictionary specifications used to actually develop the database and its applications.

V. **SDM Phase III - Programming.** This chapter will employ semantic object modeling as the methodology used for modeling the LEVRAS specifications for its data dictionary. Semantic objects and their attributes will be created using "SALSA" semantic object modeling software. SALSA will assist in making the LEVRAS schema, which will be transformed into Paradox format. Once the database tables are constructed, the GUIs will be built from user specifications gathered during the new vehicle registration system assessment as described in Chapter II. The GUIs will be tied together using relational database concepts in the ObjectPal programming language. Thorough testing will be performed before providing LEVRAS to the users. To further support NPS Security personnel, a LEVRAS User's Manual will be written specifically for NPS Security use.

VI. **SDM Phase IV - Conversion and Implementation.** This chapter will describe the actual conversion from COPS to LEVRAS, and the extensive training provided to all users and supervisors, which offset any anxiety inherent in the change process. A conversion from COPS to the new database system will be executed. Once LEVRAS is fully operational, the new system will allow the users to efficaciously query on demand, perform reliable back-ups, and conduct numerous other new or improved functions.
VII. SDM Phase V - Post-Implementation. System maintenance requirements will be addressed in the LEVRAS User's Manual. After fulfilling all system installation requirements, the NPS Security Officer will sign the "System and Acceptance Test" document, which will signify his approval of LEVRAS.

VIII. Conclusions and Lessons Learned. This final chapter will summarize the System Development Methodology process and project team concepts to be employed by the authors during this thesis. It will further expand on the author's interpretations of the overall system design and implementation process that will greatly enhance future system developers' efforts.
II. COMPUTER ON-LINE POLICE SYSTEM (COPS) BASELINE ASSESSMENT

A. EXISTING VEHICLE REGISTRATION SYSTEM

This chapter will provide an in-depth look at the existing hardware, software, and administrative procedures used for NPS vehicle management operations. COPS is a mid-1980's computer system that reduced man-hours for filing and retrieving records, and enhanced organizational clarity in security administration procedures. Training was also minimal since the COPS program GUIs are displayed in a lucid fashion and the procedures to operate COPS are mechanized. The existing system hardware and software is outlined in Table 1 below:

<table>
<thead>
<tr>
<th>ITEM TYPE</th>
<th>QUANTITY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>Two</td>
<td>Zenith ZWX-248-62</td>
</tr>
<tr>
<td>Monitor</td>
<td>Two</td>
<td>14&quot; Black and White</td>
</tr>
<tr>
<td>Secondary Storage</td>
<td>Two</td>
<td>5.25&quot; Floppy Disk Drive</td>
</tr>
<tr>
<td>Backup Storage</td>
<td>One</td>
<td>Irwin Tape Cartridge Drive</td>
</tr>
<tr>
<td>Printer</td>
<td>Two</td>
<td>Alps P2000GDot Matrix Printer</td>
</tr>
<tr>
<td>Surge Protector</td>
<td>Two</td>
<td>15 Amperes, Six Plug Outlet</td>
</tr>
<tr>
<td>Software</td>
<td>One</td>
<td>COPS (Dbase III)</td>
</tr>
<tr>
<td>Manual Card File</td>
<td>One</td>
<td>11,000 Active 5x8 Cards and Archive over 50,000 Cards</td>
</tr>
<tr>
<td>Consumables</td>
<td>Varies</td>
<td>Fanfold Paper, Floppies, Tape Cartridges, Printer Ribbons, and Index Cards</td>
</tr>
<tr>
<td>Controlled Consumables</td>
<td>Varies</td>
<td>Vehicle Decals and Temporary Passes</td>
</tr>
<tr>
<td>Office Equipment</td>
<td>One Each</td>
<td>Desk, Chair, and File Cabinet</td>
</tr>
</tbody>
</table>

Table 1. COPS Hardware, Software, and Ancillary Equipment
Presently, users of COPS enter, modify, delete, store, and display vehicular information on vehicles registered at NPS using a primitive flat-file database technology. All COPS hardware and software are located within the VIRO, building 211, and building 200, the home of the NPS security forces. The present system cannot ensure accurate information, resulting in errors that are inexcusable and embarrassing for the NPS Security Department. These errors result primarily from the data entry phase (data entry clerk typographic errors) and the possibility of loss of information due to system failures in between the infrequent backups (backups are conducted anywhere from one week to one month). Also, there is not an adequate procedure to track vehicles that are no longer registered at NPS. The Administrator scans over the entire vehicle registration database by pulling each record (one-by-one), checking vehicle registration expiration status. This Standard Operating Procedure (SOP) is excruciatingly slow. The many outdated records remaining in the database cause an increase in the mean time to respond to database queries.

Another major problem is data archiving. Once a vehicle leaves the system, only a hard-copy file is maintained in the VIRO's file cabinet. Data retrieval under this system is also archaic, as clearly exemplified in the "La Mesa child abduction attempt" previously cited in the Background portion of the Introduction. Manually sorting through thousands of archival records can take virtually hours, days, or even weeks to locate a record of concern.

Several other deficient areas were noted during our assessment. The major deficiencies include:

- Local area-networking is nonexistent. COPS consists of two similar stovepipe subsystems, which communicate via physically transporting floppy diskettes between the VIRO site and the Security site.

- Inter-networking is nonexistent. No computer data-link exists between COPS and other military installations or any outside law enforcement agencies such as the local police force, California Highway Patrol (CHP), and other police forces nationwide.
COPS hardware and software is obsolete. COPS no longer supports the functionalities required by the NPS Security Department (as discussed in this list).

Data entry is manual and subject to inaccuracies. Data entry clerks are prone to typographical errors and input data in wrong formats. COPS software was not programmed to prevent simple referential data errors.

Database archiving and backups are inadequate. VIRO's Irwin Tape Drive Cartridge Backup System is capable of backing-up COPS data; however, it is a single point of failure that has failed. This last point ties in directly with the next item in this list - system maintenance.

System maintenance is nonexistent. Presently, a COPS maintenance contract does not exist. Data entry clerks are reluctant to notify management of COPS subcomponent failure (due to physical barriers such as different buildings or different offices within the same building), thereby causing degraded system functionalities.

COPS is vulnerable to virus penetration and other security breaches. Exchanging floppy diskettes between computers is a very dangerous practice since this can spread a virus from an infected computer to an unprotected computer. COPS is also vulnerable to computer infection and privacy act violations by any subreptitious virus program or other intrusions whenever operators leave their terminals.

Query capability is limited. The query process involves manually looking through index cards or performing a computerized query on only one of the
following fields: name, social security number, license plate, decal, and ticket.

□ Administrative procedures are weak. The sole document for governing COPS keyboard entries is the COPS User's Manual, which is written and distributed by the Naval Computer and Telecommunications Station (NCTS), San Diego. This manual only delineates the type of data that needs to be entered in a specific field on a specific screen. For example, on the Decal Entry Screen, this manual states:

Enter the required information in the blank field provided. Validation of data entered is done for "DECAL NUMBER", "EXPIR YR", "EXPIRMO", "LICENSE NUMBER", "VEHICLE BODY", VEHICLE MAKE", and "VEHICLE COLOR". (NCTS San Diego, 1991, p. 13)

Although the COPS User's Manual provides data entry instructions, it lacks a standard operating procedure (SOP) that could be used throughout DoD. When the VIRO Administrator is absent from the COPS workstation, the vehicle registration process ceases. An SOP would alleviate this major problem, as well as other security and customer related topics.

It is clearly evident that COPS is riddled with numerous problems which include data field ambiguities, severe query limitations, and weak security safeguards. Some of these discrepancies are small, but unfortunately many are large and are inherent in the present method of the COPS database operations. Although the paper filing system is a somewhat effective way to archive and backup data, it is extremely slow and inefficient. The technology of today offers several solutions to the deficiencies described above, as delineated in the following sections.
B. NEW VEHICLE REGISTRATION SYSTEM ASSESSMENT
(SDM PHASE I - SYSTEM ANALYSIS)

All the problems described above were discussed in detail with the NPS Security Officer and his staff. User inputs for improving the current system functionalities, as well as additional non-existing functionalities were identified during these discussions. The format used in this section will first list the previously identified problem, and then provide the authors' recommendations for improvement based on the Security Department personnel inputs.

Problem: Local area-networking is nonexistent.
Solution: The authors suggested that a local area network (LAN) be designed to connect the Vehicle Registration Office (building 211) and the NPS Base Police Station (building 200). As a result, Security Department generated MIS contract was approved for connection of the two independent workstations. These workstations are now located within the same building, the NPS Police Station. This strategy improved the entire intra-communication process among the VIRO and Security personnel. The probability of complete, accurate, and timely data transfer within the department is dependent on hardware and software exchange which now minimizes the negative effects of the human intervention process.

Problem: Inter-networking is nonexistent.
Solution: A modem and communication software should be incorporated into the LEVRAS system to facilitate communication with other law enforcement agencies, DoD security forces at other locations, and on-campus officials not included on the LEVRAS LAN. This will provide
rapid police data exchange between these agencies, and further assist in timely police response.

**Problem**: COPS hardware and software is obsolete.

**Solution**: A new suite of hardware and software will be provided with LEVRAS, gaining speed, reliability, and increased functionality. The new hardware components will most likely include:

- (2) IBM Compatible 486, 66 Megahertz (MHZ) computers
- (2) 8 Megabyte (MB) Random Access Memory (RAM)
- (2) 1 Gigabyte (GB) Hard Drives
- (2) Network Interface Cards (NIC)
- (2) Laser Printers
- Coaxial cabling and connectors
- Modem, 14.4 Bits Per Second (BPS)
- Uninterrupted Power Supply (UPS).

The new software components will most likely include:

- Communications (Modem) Software
- Network Software
- Anti-Virus Software
- Paradox Database Software
- Semantic Object Modeling Software.

**Problem**: Data entry is manual and subject to inaccuracies.

**Solution**: The LEVRAS database entry screens will substantially reduce data entry errors through validity checks that will ensure that data is correct and
appropriate. "Validity checks help to minimize data errors by checking the
data before it is placed in the table [database]." (Rock, 1993, p. 277)

Problem: Database archiving and backups are inadequate.
Solution: The new LEVRAS system will include two, one gigabyte hard
drives, for complete archival and database backup as noted above in the new
hardware component list.

Problem: System maintenance is nonexistent.
Solution: A comprehensive MIS system hardware maintenance plan was
included in the LAN installation contract. Database software maintenance
can be accomplished by in-house personnel using the LEVRAS User's
Manual and other references, or outside contractors.

Problem: COPS is vulnerable to virus penetration and other security
breaches.
Solution: The LEVRAS will use commercial-off-the-shelf (COTS) anti-virus
software, which will help reduce numerous security vulnerability problems.
The software should be a reputable COTS product, such as Norton Anti-Virus
or McAfee Anti-Virus software. Other security improvements include the
relocation of VIRO computer to NPS Security Police Headquarters, which
is manned continuously. This improves security by providing 24 hour
supervision of the entire LEVRAS system.

Problem: Query capability is limited.
Solution: The LEVRAS is based on Paradox, which provides a complete and
flexible relational database management system with a comprehensive query
capability. Paradox users can query on anything from a simple question
about the information in one table, to a complex question about the
information in several tables. In a Paradox query you can specify: tables to ask questions about; fields you want to see in the answer; records you want to select; calculations you want to perform; and answers to 'what if' questions. Queries can also perform operations that: insert new records; delete records; change values; and create new fields. This enhanced query function of the vehicle registration database is the primary reason why the authors and the NPS Security Officer decided to upgrade COPS.

□ Problem: Administrative procedures are weak.
Solution: The authors will provide a comprehensive LEVRAS User's Manual, which will assist in the development of a SOP specifically for vehicle registration, security and other relevant procedures.

The proposed LEVRAS system will be specifically designed to support the NPS Security Force. Its primary function will be to provide a state-of-the-art relational database management system with a comprehensive query capability, providing accurate, timely, and complete vehicular, owner, decal, and ticket violation information to NPS Law Enforcement Officials. Functional improvements are included in the table on the following page. The solutions to the problems described above and the proposed LEVRAS characteristics presented on the following page in Table 2, will be the foundation for this project's requirements specification (RS).
<table>
<thead>
<tr>
<th>CHARACTERISTICS</th>
<th>LEVRAS</th>
<th>COPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific content of information outputs</td>
<td>Increased content to meet user needs</td>
<td>Basic vehicular registration data</td>
</tr>
<tr>
<td>Selectivity</td>
<td>Extensive data query capability</td>
<td>Limited data query capability</td>
</tr>
<tr>
<td>Time lags</td>
<td>Fast 80486 Central Processor Unit (CPU), 66 MHZ</td>
<td>Slow 80286 CPU</td>
</tr>
<tr>
<td>Accuracy of outputs</td>
<td>Increased accuracy using validity checks</td>
<td>Susceptible to data entry errors</td>
</tr>
<tr>
<td>Reliability</td>
<td>Increased reliability using UPS and internal hard drive backups</td>
<td>No UPS &amp; backups unreliable due to a malfunctioning Irwin tape drive</td>
</tr>
<tr>
<td>Generality</td>
<td>General enough for both ticket and VIRO processing, &amp; understandable information for supervisor reporting</td>
<td>Not general enough: ticket processing done on two separate systems and the information is too narrow in scope</td>
</tr>
<tr>
<td>Flexibility</td>
<td>Easy to modify Paradox's ObjectPal code</td>
<td>Moderately difficult to change program dBase code</td>
</tr>
</tbody>
</table>

Table 2. Proposed Characteristics of LEVRAS vs. COPS

This section discussed the existing vehicle registration and security information system with the proposed new LEVRAS system. The NPS Security Officer, as well as the authors, feel that there is an overwhelming need to improve the existing system's hardware and software due to the inherent COPS functionality deficiencies. The SDM Phase I - System Analysis is now complete. The following chapters will discuss the authors' design and implementation of the new system.
III. DATABASE DEVELOPMENT PROCESS

This chapter will begin with a conceptual view of what the new information system will necessitate. The scope of this thesis will be narrowed to identify what kind of information system is appropriate for the new system. Once it has been determined what kind of information system LEVRAS will require, then the database design will be developed in order to properly manipulate the data to meet all functional requirements.

A. BASIC DATABASE CONCEPTS

LEVRAS will be a management information system (MIS). The reason for this is:

An MIS is an integrated structure of databases and information flow that optimizes the collection, transfer, and presentation of information throughout a multilevel organization whose component groups perform a variety of tasks to accomplish a united objective. (Long, 1993, p. 441)

Although a typical MIS may be incorporated into large corporations with many departments, LEVRAS is narrower in scope and breadth. LEVRAS still fulfills the definition of a MIS, since it contains the core elements, such as an integrated database structure that pulls together many data elements of the organization in a relational manner.

The key to the proper operation of the envisioned new system is its relational database. A relational database encompasses the following:

A two-dimensional array containing single-valued entries and no duplicate rows. The meaning of the columns is the same in every row. The order of the rows and columns is immaterial. (Kroenke, 1992, p. 640)
A database is:

A self-describing collection of integrated records... it contains, in addition to the user's source data, a description of its own structure. This description is called the data dictionary... A database is more than a collection of files. A database includes files of source data plus a description of the relationships among the records in the files. These relationship descriptions are stored and recalled during database processing. (Kroenke, 1992, pp. 12-14)

The relational database described above will be beneficial to LEVRAS for three reasons. First, the organizational layout of the database will be conceptually easy to understand due to its two-dimensional format of columns and rows. Second, the relationships between the data elements will be easily comprehended by users. For example, a VIRO customer may have multiple vehicles. Third, this database structure can be used to support many functional applications such as transaction processing, supervisor reports, and assistance to decision-making.

B. DATABASE DEVELOPMENT METHODOLOGY

During the LEVRAS database development process, the five phases of the System Development Methodology (SDM) will capture the essence of the intended application functionalities. The phases are described below:

- **SDM Phase I - System Analysis** (determine the project objective):
  - Form project team: team leader, programmers, system specialists.
  - Define the problem: team members consolidate their respective views to establish a consensus definition of the project objective.
  - Establish scope: prioritize the required functions and choose the functions above a set threshold.
  - Assess feasibility: determine and evaluate political, economic and technical constraints.
- Develop requirement specifications (RS) to meet user's needs: includes the desired inputs, outputs, and process constraints.

☐ SDM Phase II - System Design (determine what the system must do):
- Determine functional components: update, display, and control mechanisms.
- Create user's process model: describes the organizational processes and the objects to be stored in the database.
- Reassess requirements: finalize system requirements.
- Reassess feasibility: determine if the chosen architecture remains feasible and present to systems sponsor for review and approval.

☐ SDM Phase III - Programming (determining how the system will operate):
- Use prototypes: Working model of forms, reports, and menus for user review and feedback.
- Select systems architecture: choose the best data model that meets user's requirements.
- Develop database design: objects, attributes, and relationships.
- Construct database: create database schema based on object design.
- Develop application design: menus, data entry screens, query schemes, reports, and control mechanisms.
- Build applications: create user interfaces with a programming language to link the menu and data screens to required system tasks.

☐ SDM Phase IV - Conversion and Implementation (construct system in accordance with its design):
- Choose best conversion method: abrupt cutover, parallel, location, or staged.
- Train users and maintenance personnel: User's manual, other references, and hands-on.
- Install database and applications: insert new programmed database software and its relevant data into the desired system hardware.
- Conduct System and Acceptance Test: system sponsor approve new system operation.

☐ SDM Phase V - Post-Implementation Phase (support for long term operations).
- Complete and deliver the user's manual: include operator help and maintenance guidance.
- Execute system maintenance plan: routine and corrective.

C. TECHNICAL DATABASE CONSIDERATIONS

This section will address project team concerns regarding the fabrication of the database. The project team will then take these concerns and determine the functional components of each application that will be used in the database. There are two ways to fulfill the user's requirements and build the user's database. Specifically, these development methods are the top-down and bottom-up styles. The top-down approach takes a broad view and narrows the scope to specific functionalities. The bottom up approach first takes a myopic look at the organizational tasks, and then broadens its view outward to the strategic goals. Combining both the top-down and bottom-up styles is known as the hybrid database development methodology.

1. Data Flow Diagrams

Data flow diagrams (DFDs) will be used to provide the project team members with a conceptual view of the data to be used throughout the database and its related applications.
DFDs will also be used to identify the data flow through a system and determine the work processes required to implement system functions. Figure 1 depicts a generic DFD. Descriptions of the DFD components will follow the diagram.

Figure 1. Generic Data Flow Diagram

DFDs are comprised of the following components:

- **Process**: performs a transformation on the incoming data flows, that is, the outgoing data flows will contain data that has been altered from the original incoming data. There must be at least one incoming and one outgoing data flow for each process.

- **Data Flow**: is equivalent to a "pipe" that carries information from one source to another. One of these sources must be a process. The data flow can also be coming from an entity or a data store.

- **Entity**: also known as sources or destinations. Entities provide inputs an outputs to the system. These inputs and outputs can be located inside processes or outside the system.
Data Store: synonyms include file and database. The data store does exactly what its name implies, it stores the data for the system. New data can be entered in the data store, and then retrieved, manipulated, or deleted.

The data flow diagrams are exploded, or decomposed, until primitive DFD levels display the basic process and data flows, which will help define the system data requirements. Computer-Aided Systems Engineering (CASE) tools, such as EXCELERATOR by Intersolv, can be used to transform the DFDs into their respective database applications. Another alternative is to manually convert this information into semantic object models, as discussed in the previous section. In either case, a data dictionary will be created that will contain the descriptions of the primitive data requirements derived from the DFDs.

2. Data Dictionary

The data dictionary will capture the DFDs themselves, as well as the requirements and specifications definitions that the DFDs provided.

A database is self-describing: it contains, in addition to the user’s source data, a description of its own structure. This description is called the data dictionary (or data directory, or meta data). It is the data dictionary that makes program/data independence possible. (Kroenke, 1992, p.13)

A customer receiving a receipt at the check-out counter, for example, would provide a picture of data flowing from a check-out process to a customer entity. In addition, the customer attributes, the data elements contained in the data flow, and the process definition would all be described and stored in the data dictionary.
3. Semantic Object Modeling

Developing the database will require the project team having a good grasp of what is needed in the form of data. This data will be represented in the form of objects (using semantic object modeling) and their respective attributes. Figure 2, on the following page, provides an example of a semantic object and its characteristics. Definitions that explain the semantic object and its characteristics precede the figure.

The overall view of semantic object modeling can be described as follows:

- **Semantic Object**: a person, place, or thing. Relevant data about these objects is stored within the database. Example - CUSTOMER (person), VIRO (place), TICKET (thing).

- **Attribute**: describes the semantic object in these forms - simple, group, formula, or semantic object link. Example - CustomerID (simple), CustomerName (group), TicketPoints (formula), SocialSecurityNumber (semantic object link).

- **Identifier**: attributes that are used to distinguish an instance of a semantic object, which can be unique or nonunique. Example- CustomerID (unique) and CustomerName (nonunique).

- **Subtype Semantic Object**: a parent semantic object broken down into children, and are more specialized or specific about the parent. Example - AUTOMOBILE (semantic object or parent) specialized to MOTORCYCLE and TRUCK (subtype semantic objects or children).
- **Cardinality:** The minimum or maximum amount an attribute can characterize one instance of a semantic object or of a group attribute. Example - one CUSTOMER to many AUTOMOBILES.

- **Domain:** describes the set of possible data types and formats available for use in the database. Example - SocialSecurityNumber 123-45-6789.

![Diagram of OBJECT1](image)

**Figure 2.** Generic Semantic Object
The database will be developed using more than one requirement. Using multiple requirements will cause these requirements to overlap, and will create more complexity in the overall database. Requirements that are identified properly from the outset of the project will greatly deter the dreaded occurrence of *analysis-paralysis*.

The database design is an important part of the relational model because it will be used to delineate database management system (DBMS) independent designs. In other words, semantic objects defined in the data dictionary and their characteristics will be more clearly solidified. A social security number, for example, will be formatted to contain the proper domain of physical properties to include data type (positive whole numbers), field size (11 elements including the hyphens between the numbers), and value constraints (zero to nine). Semantic object modeling will assist in capturing the meaning of the data to be modeled. It deals with objects and their characteristics to determine which data is to maintained and how to handle the relationships between the objects. This information is further used to create a database that will eventually be transformed into relational tables. These tables will allow the users of the system to maintain a relevant database that may provide a wide-range of services including multi-table: forms, reports, screens, and queries.

The following heuristic can be used to develop the semantic object model (refer to Figure 2. Generic Semantic Object):

a. **Step 1.** Retrieve semantic objects from the data dictionary. If applicable, determine which semantic objects will be the parent and their respective children. Also, links can be made between objects.

b. **Step 2.** List all of the attributes for each semantic object to be modelled. There are simple and group attributes. Group attributes can be uniquely defined by grouping list items (sub-attributes) under the group attribute heading. If applicable, determine any formulas relevant to the object.
c. **Step 3.** Choose one semantic object attribute that tags the selected object. This will uniquely identify the object to distinguish itself from other like objects.

d. **Step 4.** Determine the value and format for each of the attributes. The physical properties of the attribute can be described as a data type, field size, or value constraint.

e. **Step 5.** Assign the cardinality of an attribute that describes one instance of a group attribute or semantic object. An instance is an actual person, place, or thing rather than an abstract object.

f. **Step 6.** Generate schema. A schema describes the database to the DBMS; however, it will not cause any of the actual data elements to be entered.

After the schema has been generated, the information system designers can commence work on the generated tables within the respective database software program. This schema provides the foundation upon which the applications, such as forms, screens, and reports can be built.

4. **Database Application Design**

Designing the applications for the database will require the programmers to know how to make the software capture the essential functional and data requirements. "An application is an integrated collection of related features that permit you to perform a task." (Jensen and Anderson, 1994, p. 4) At this point in the system development methodology, the software moves from logical to physical structures (programs and data files). These structures define programs to carry out specific system functions such as entering data and printing multi-table reports. In presenting the design graphically, the programmers will be able to use different representations within Paradox's ObjectPal programming language. Some graphic design options include: radio buttons, scroll bars, and dialog boxes.
When developing the software components, programmers should create the program in sections from smallest to largest: first program units (tables), then modules (groups of tables) supporting the major processes, the interfaces among modules and with external systems (LAN nodes). The program will be debugged (tested) as the program is being built to prevent small errors from expanding into large and potentially costly errors. Testing includes the following: unit, module, and system integration testing. When programs are debugged and the errors are removed, documentation should be written for future reference to support training and maintenance. Also, programmers should ensure that when an error is detected and corrected that it does not affect other program parts. Documentation will also be written in incremental stages to provide the users and maintenance personnel with a well written and complete user's manual (finalized in the implementation phase).

Now that the program code has been written to meet the system requirements, the user's will be able to see the finished product. The finished product includes all of the functionalities as they pertain to their physical attributes. These physical attributes include: how the user will open the program; what the opening presentation screen will display; how to enter data into input screens and how these screens will appear to the user; what kind of effects feature buttons and pull-down menus will have on the database and its applications; what kind of output data will be presented on reports; and how to exit the program during a session. This system is now ready to be delivered.

5. System Conversion and Acceptance Test

System conversion can be implemented by a variety a methods: abrupt cutover, parallel conversion, location conversion, and staged conversion. The following paragraphs will address each of the conversion approaches individually. The remainder of this section will address the system acceptance test.

The abrupt cutover uses a rapid approach to system conversion. On a specific date, the old system will be tuned-off and the new system will be turned-on. This is considered a very high-risk approach to system conversion. If the new system crashes, then there is a
strong possibility of having a long recovery time, which would be unacceptable for mission critical operations. If an organization cannot afford any system down time, then it is recommended that another conversion approach be used.

Parallel conversion simultaneously uses both the new and old system during the conversion stage. This is considered a low-risk conversion process since both systems are being used. If one system crashes then the other system is used without losing any information. This approach is best utilized on mission critical systems that cannot afford to sustain any downtime. A drawback of this approach is that it takes more manpower and resources to run two systems simultaneously.

Location conversion is used when an organization is going to replace many of the same systems. The organization will choose one of several departments that will be converting their information systems. This one department will be the test bed for all other departments undergoing future information system conversions. Lessons learned can be gathered and used to smooth all other future conversions. This method is also considered a low-risk approach since only one department will sustain a temporary loss of operations if the new system should crash.

Staged conversion can best be summarized as follows:

Like location conversion, staged conversion is a variation on the abrupt and parallel conversions. A staged conversion is based on the version concept introduced earlier. Each successive version of the new system is converted as it is developed. Each version may be converted using the abrupt, parallel, or location strategies. (Barlow, Bentley, and Whitten, 1994, p. 740)

The systems acceptance test is the major and final activity that occurs during system conversion. The systems acceptance test can be defined as the final system test, which will be performed by the end-users. The users will use real data during this test. After successful completion of the systems acceptance test, the sponsor(s) will take custody of the system.
6. Documentation, Training, and Maintenance

A key factor in the successful support and maintenance of software is complete and comprehensive documentation. The manuals should be written to support users and maintenance personnel. Meeting requirements for system documentation is the objective factor; however, the subjective factor of quality is actually the more crucial part of this document. In other words, the user's manual should aid both the users and maintenance personnel in the performance of their jobs.

Training the end-users to effectively operate the new system is also an important factor in ensuring that the systems is fully used. Training should take place during the implementation phase, and if possible even earlier, to make the conversion process run more smoothly. This training entails reading the user's manual, as well as hands-on training with portions of the new system.

Another technical consideration is the maintenance plan. The project team should deliver a well thought out method for the information system sponsor to employ a routine preventive maintenance plan, as well as pursue equipment repairs when needed. This maintenance plan should also include information and references, in case users desire to upgrade system functionalities. Three avenues for maintenance personnel include: the user's manual, other reference manuals and user's guides (for example, Borland Paradox for Windows User's Guide and the Guide to ObjectPal), and as a last resort, use hired contractors.
IV. SDM PHASE II - SYSTEM DESIGN

In Chapter II, SDM Phase I - System Analysis, the authors identified inherent problems with the current security Information System (IS). Potential solutions to these deficiencies were also presented. In Chapter III, Section C, Technical Database Considerations, the authors discussed other factors that will assist in the development and design of the improved IS. The result of these efforts is the Requirements Specification (RS) delineated in Appendix A. This RS is a conglomeration of functional and non-functional requirements including: the input of data, the processing and storage of this data, and system outputs. For example, a vehicle model is entered into the database, and the NPS Security Officer is presented a summary listing of all the Ford Pintos currently on file.

The remainder of this chapter will be devoted to the development of specific data objects and processes using the RS. Data flow diagrams will be constructed to help define each of these system elements. The final product should be a data dictionary containing element descriptions to be used in the next phase of IS development.

A. DATA FLOW DIAGRAMS

Chapter III presented the basics of data flow diagram (DFD) theory, as it applies to the development of the new Law Enforcement and Vehicle Registration Administration System (LEVRAS). In this section, DFDs were actually created for LEVRAS, using a computer aided systems engineering (CASE) tool. These DFDs are contained in Appendix B, and are discussed below. As mentioned previously, these DFDs will be used to transform the RS into data dictionary definitions.

The first step in developing DFDs is the construction of the decomposition diagram. The LEVRAS Decomposition Diagram, located in Appendix B, provides an overview of LEVRAS DFDs.
LEVRAS is composed of three primary levels, and includes the following processes:

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>PROCESSES</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Level</td>
<td>LEVRAS (Overall System)</td>
</tr>
<tr>
<td>Second Level</td>
<td>Process 1 - Process Customer Checkin</td>
</tr>
<tr>
<td></td>
<td>Process 2 - Process Customer Checkout</td>
</tr>
<tr>
<td></td>
<td>Process 3 - Generate Query Response</td>
</tr>
<tr>
<td></td>
<td>Process 4 - Process Tickets</td>
</tr>
<tr>
<td></td>
<td>Process 5 - Generate Reports</td>
</tr>
<tr>
<td>Third Level (Primitive)</td>
<td>Process 1.1 - Input Customer Data</td>
</tr>
<tr>
<td></td>
<td>Process 1.2 - Modify Customer Data</td>
</tr>
<tr>
<td></td>
<td>Process 1.3 - Generate Vehicle Decal</td>
</tr>
<tr>
<td></td>
<td>Process 1.4 - Generate Temporary Pass</td>
</tr>
<tr>
<td></td>
<td>Process 2.1 - Process Data Transfer</td>
</tr>
<tr>
<td></td>
<td>Process 2.2 - Archive Customer Data</td>
</tr>
<tr>
<td></td>
<td>Process 3.1 - Validate Requested Data</td>
</tr>
<tr>
<td></td>
<td>Process 3.2 - Generate Query Response</td>
</tr>
<tr>
<td></td>
<td>Process 4.1 - Input Ticket Data</td>
</tr>
<tr>
<td></td>
<td>Process 4.2 - Process Ticket Disposition</td>
</tr>
<tr>
<td></td>
<td>Process 4.3 - Archive Ticket Data</td>
</tr>
<tr>
<td></td>
<td>Process 5.1 - Generate Ticket Report</td>
</tr>
<tr>
<td></td>
<td>Process 5.2 - Generate Decal Report</td>
</tr>
<tr>
<td></td>
<td>Process 5.3 - Generate Customer Report</td>
</tr>
<tr>
<td></td>
<td>Process 5.4 - Generate Custom Report</td>
</tr>
</tbody>
</table>

The next step in the DFD process is to display how LEVRAS relates to its environment, using a context diagram. The LEVRAS Context Diagram (First Level) consists of one process (LEVRAS, the overall system), three entities (CUSTOMER, SECURITY OFFICER, POLICEMAN), and six associated data flows (see Context Diagram in Appendix A). The primary utilization of the system is centered around the ability of the SECURITY OFFICER (and his staff) to access customer (NPS student, staff and contractors) vehicular
and supporting data on a real-time basis. Information from the CUSTOMER and the POLICEMAN entities populates and updates the database to provide current information to the SECURITY OFFICER at all times.

The context diagram is exploded (decomposed) to reveal the next level of granularity. The LEVRAS Second Level Diagram consists of five processes (PROCESS CUSTOMER CHECKIN, PROCESS CUSTOMER CHECKOUT, GENERATE QUERY RESPONSE, PROCESS TICKETS, and GENERATE REPORTS), three entities (CUSTOMER, SECURITY OFFICER, and POLICEMAN), three data stores (CUSTOMER/VEHICLE DATA, TICKET DATA and ARCHIVE DATA), and the associated data flows (see Appendix A Second Level Diagram).

Referring to the Second Level Diagram, the SECURITY OFFICER is centrally located and is able to submit query requests and receive query responses for all system data through Process 3, the "Generate Query Response" process. In addition, the SECURITY OFFICER utilizes Process 4, the "Process Tickets" process, to receive information on and dispose of tickets. Finally, the SECURITY OFFICER uses Process 5, the "Generate Reports" process, to request and receive system data printouts.

The following describes the Second Level Data Flows:

- The CUSTOMER provides data to the system through Process 1, the "Process Customer Checkin" process and Process 2, the "Process Customer Checkout" process. The system responds with decal or pass assignment (not shown as a data flow since it is implicit) and a checkout complete report.

- The POLICEMAN interfaces only through Process 4, "Process Tickets", which he or she uses to enter ticket information and in turn receives a completion report from the system.
The three data stores previously discussed compose the system database, and store system information. The Second Level Diagram also contains numerous data flows that accomplish the interfaces between the data stores and the five Second Level processes.

The final step of the DFD process is to decompose the Second Level DFD into its most primitive form. Descriptions of the five primitive DFDs are as follows:

- **Third Level DFD (Processes 1.1P - 1.4P).** Consists of four processes (INPUT CUSTOMER DATA, MODIFY CUSTOMER DATA, GENERATE VEHICLE DECAL and GENERATE TEMPORARY PASS), one entity (CUSTOMER), one data store (CUSTOMER AND VEHICLE DATA) and associated data flows. Data is input and displayed in Process 1.1P and displayed and modified in Process 1.2P. Once the data is input, transactions to the CUSTOMER include "Decal Assignment" and "Temporary Pass Assignment".

- **Third Level DFD (Processes 2.1P - 2.2P).** Consists of two processes (PROCESS DATA TRANSFER and ARCHIVE CUSTOMER DATA), two entities (CUSTOMER and SECURITY OFFICER), two data stores (CUSTOMER AND VEHICLE DATA and ARCHIVE DATA) and associated data flows. CUSTOMER data is displayed and deleted from the CUSTOMER AND VEHICLE DATA data store and transferred to the ARCHIVE DATA data store via Processes 2.1P and 2.2P.

- **Third Level DFD (Processes 3.1P - 3.2P).** Consists of two processes (VALIDATE REQUEST DATA and GENERATE QUERY RESPONSE), one entity (SECURITY OFFICER), three data stores (CUSTOMER AND VEHICLE DATA, ARCHIVE DATA, and TICKET DATA) and associated data flows. The SECURITY OFFICER entity represents the initiation of
control processes in the system rather than an item to maintain data on. Transactions at this level include a "Query Request" via processes 3.1P and 3.2P, which result in a properly formatted "Query Response" back to the SECURITY OFFICER.

Third Level DFD (Processes 4.1P - 4.3P). Consists of three processes (INPUT TICKET DATA, PROCESS TICKET DISPOSITION, and ARCHIVE TICKET DATA), two entities (POLICEMAN and SECURITY OFFICER), two data stores (TICKET DATA and ARCHIVE DATA) and associated data flows. The SECURITY OFFICER entity represents the initiation of control processes in the system. The POLICEMAN entity inputs ticket data. Once the ticket data is input and displayed in Process 4.1P, Process 4.2P is utilized to process the ticket disposition.

Third Level DFD (Processes 5.1P - 5.4P). Consists of four processes (GENERATE TICKET REPORT, GENERATE DECAL/PASS REPORT, GENERATE CUSTOMER/VEHICLE REPORT, and GENERATE CUSTOM REPORT), one entity (SECURITY OFFICER), three data stores (TICKET DATA, CUSTOMER AND VEHICLE DATA, and ARCHIVE DATA) and associated data flows. The SECURITY OFFICER entity represents the initiation of control processes to specifically request and receive reports. Transactions consist of the generation of four types of reports via the following four processes: Process 5.1P (GENERATE TICKET REPORT), Process 5.2P (GENERATE DECAL AND TEMPORARY PASS REPORT), Process 5.3P (GENERATE CUSTOMER AND VEHICLE REPORT), and Process 5.4P (GENERATE CUSTOM REPORT). The CUSTOM REPORT can be tailored according to the specific needs of the SECURITY OFFICER on a real-time basis. Note the
presence of all three data stores at this level, facilitating complete access to all system information.

The proposed LEVRAS system is being specifically designed to support the NPS Security Force. The above processes compose the core of LEVRAS. Its primary function will be to provide a state-of-the-art relational database management system with a comprehensive query capability providing accurate and complete customer, vehicular, and ticket violation information to NPS law enforcement officials in a timely manner. All RS Primary Functional Requirements contained in Appendix A, are included in the primitive level DFDs above.

B. DATA DICTIONARY

In Chapter III, Section 2, a generic data dictionary was defined. This section, along with Appendix C, specifically elaborates on the LEVRAS Data Dictionary. The LEVRAS Data Dictionary includes: LEVRAS External Entities (CUSTOMER, SECURITY OFFICER, and POLICEMAN), LEVRAS Processes (1.1P INPUT CUSTOMER DATA, 1.2P MODIFY CUSTOMER DATA, 1.3P GENERATE VEHICLE DECAL, 1.4P GENERATE TEMP PASS, 2.1P PROCESS DATA TRANS, 2.2 ARCHIVE CUSTOMER DATA, 3.1P VALIDATE REQUEST DATA, 3.2P GENERATE QUERY RESPONSE, 4.1P INPUT TICKET DATA, 4.2 PROCESS TICKET DISPO, 4.3 ARCHIVE TICKET DATA, 5.1P GENERATE TICKET REPORT, 5.2P GENERATE DCL/PASS REPORT, 5.3P GENERATE CUST/VEH REPORT and 5.4P GENERATE CUSTOM REPORT), LEVRAS Data Stores (CUSTOMER AND VEHICLE, TICKET DATA, and ARCHIVE DATA), and their associated LEVRAS Data Flows. Finally, the LEVRAS Data Dictionary will be used as the foundation for SDM Phase III - Programming.
V. SDM PHASE III - PROGRAMMING

In Chapter III, Database Development Process, the authors discussed the many theoretical facets of Information System (IS) design. The initial thrust of the programming portion of the database development process was shown to focus on the quick generation of a prototype system. This was done, and in fact provided much information which assisted in the development and design of the improved IS. First, prototype graphical user interfaces (GUIs) were built and shown to the users for their review. Second, sample forms and reports were customized to meet the requirements of the NPS Security Officer. The result of these efforts was specific design features to be used during programming, including the fact that the semantic modelling method would adequately support the construction of an effective database.

A. SEMANTIC OBJECT MODELING

Chapter III provides a summary of semantic object modelling theory, including a heuristic for their development. The implementation of this heuristic will now be discussed. Appendix D contains the semantic objects and an expansion of the Data Dictionary in Appendix C, which describes the object attributes and relationships.

a. Step 1 The LEVRAS Data Dictionary was reviewed, and all appropriate data objects and their elements were established. The seven objects were: Customer (any person requiring vehicle registration services, such as a Naval Postgraduate School student or a private contractor requiring access to government grounds), Vehicle Registration (a customer's valid vehicle registration), Drivers License (a customer's valid drivers license), Insurance Certificate (a customer's valid vehicle insurance policy card or statement), Vehicle (a customer's vehicle requiring registration), Vehicle Decal (the appropriate official
decal presented to a customer upon proper entry and validation of information using LEVRAS), and Traffic Ticket (a ticket issued by NPS Security Department policemen when a customer is in violation of NPS traffic regulations).

b. **Step 2** Using the *SALSA* semantic object modelling software package, the objects were individually entered. These semantic objects directly relate to the physical objects described in step 1, and were named: CUSTOMER, REGISTRATION, DRIVERLICENSE, INSURANCE, VEHICLE, DECAL, and TICKET. Then, all the attributes addressed in step 1 were named, individually entered using *SALSA*, and assigned to the appropriate semantic object. For example, the CUSTOMER semantic object contains many attributes, including LastName, FirstName, HomeAddress, and DutyStation.

c. **Step 3** One semantic object attribute, which uniquely tags the selected object, was chosen to distinguish it from the other objects. These identifiers are listed first in each semantic object, and are coded by a preceding "ID" symbol. The CUSTOMER object is uniquely identified by the SocialSecurityNumber attribute, while VEHICLE can be distinguished by its LicensePlateNumber attribute.

d. **Step 4** The value and format for each of the attributes were determined. The physical properties of the attributes including data type, field format and length, and a description were entered. The Attribute Report, which directly follows the LEVRAS Semantic Objects in Appendix D, shows these entries in detail. This step also involved establishing semantic object links. These links are shown by the inclusion of one or more object's names at the bottom of the listed

40
attributes for that object. For example, the CUSTOMER and VEHICLE object names appear as the last two items listed within the DECAL semantic object.

e. **Step 5** The cardinality of each attribute, which describes one instance of a group attribute or semantic object, was then assigned. An instance is an actual person, place, or thing rather than an abstract idea. For example, the TICKET object contains the attribute BaseJudgeName. When the ticket is first written, a judge may not have been assigned, hence a minimum cardinality of zero is required. NPS Security Department regulations require a single judge to disposition each ticket. This establishes a maximum cardinality of one.

f. **Step 6** The semantic objects and their attributes were presented to the users for final review, since these data models would determine the actual database structure. The final touches were entered, and then the schema was generated for our database software package, PARADOX. The database structure will be discussed in the following section of this chapter.

The semantic objects, and descriptions of their attributes and relationships are contained in Appendix D. The database tables are part of the schema which was generated using SALSA, and are contained in Appendix E.
B. DATABASE APPLICATION DESIGN

The next step in the development of LEVRAS was to build "user friendly" GUIs upon the database table structures, which were generated in the final phase of semantic object modelling (see Appendix E). As discussed in the opening paragraph of this chapter, a prototype was used to verify user GUI preferences. These ideas were blended with the many GUI construction tools provided in the PARADOX database software package to finalize the application interfaces.

The first two application screens provide access to the other many program functions. When initially starting the LEVRAS program, a welcome screen provides the name of the system, as well as the "feeling" that the program has successfully loaded. The user can read the program title slide, and then jump to the main menu by clicking on the START pushbutton. (These two opening screens are contained in Appendix F). Here the system pauses, waiting for the user to select one of the following modes:

1. **Customer Data Mode.** Allows the VIRO Clerk to input, modify, or archive customer data. The program steps the user through the following data input forms: Customer Personal Data, Vehicle Data, Drivers License Data, Vehicle Registration Data, and Vehicle Insurance Data. Once the appropriate information has been entered into LEVRAS and verified by the VIRO Clerk, the proper decal is issued to the customer. The Decal Data Input Form is then used to enter the decal information. Refer to Appendix G for examples of these data input forms.

2. **Ticket Data Mode.** Allows the Ticket Administrator to input and update ticket information. After the customer and decal data has been entered, the user can enter data into the Ticket Data Entry Form. The program presents the user with a form similar to those previously discussed. In addition, a Ticket Administration screen has been included to allow later entry of ticket information, which is not
normally available during initial ticket data processing. This screen provides much of the customer, vehicle and decal data used to assist the Ticket Administrator in his or her duties. Refer to Appendix G for an example of the Ticket Data Input Form.

3. **Reports Mode.** Allows the VIRO Clerk, Ticket Administrator, or other NPS Security Department staff member to print out routine and customized reports. These reports include a Customer and Vehicle Status Report, a Decal Status Report, a Ticket Status Report, and a Customized Report (per the directions of the NPS Security Officer). Refer to Appendix H for examples of these reports.

4. **Query Program Mode.** Allows the NPS Security Department staff to enter any amount of customer, vehicle, or ticket data and obtain a list of customers or other desired output data matching the description of the input information.

5. **Exit Program Mode.** Allows LEVRAS users to exit the database program so that their computers can be used for other functions, such as word processing.

Once the various forms and reports were constructed, each of them had to be linked to the appropriate database tables using the structuring tools in **PARADOX**. This was done to ensure that data entered or modified using LEVRAS GUIs would be properly maintained in the LEVRAS database. Next, these GUIs had to be properly linked to the Main Menu, to allow logical and "user friendly" flow through the program functionalities. This was accomplished using the ObjectPAL programming language, also contained in the **PARADOX** software package. Software driven pushbuttons were also programmed using ObjectPAL, to assist in the easy execution of LEVRAS commands. Such commands include "Add", to enter a new customer record into the database, and "Back", to launch the program back to the previous data entry screen. Program scripts, which drive the screen linkages and pushbuttons, are contained in Appendix I.
The final step in application programming was to provide for referential integrity and record data validity checks. The software features contained in PARADOX provide for these valuable functions to be added while programming. Validity checks are specifications which define the range of acceptable values for database table fields. These checks prevent improper data from being added to a record. Numbers, for example, can be prevented from being entered into letter only fields. The referential integrity feature enforces the relationships between data stored in separate, yet related tables. This ensures that like data, such as a social security number, is exactly the same in any table of the database in which it appears. These two powerful functionalities embody the key elements which make multitable queries possible.

In this chapter, the authors have reviewed the key elements of the LEVRAS Data Dictionary to construct a prototype. This prototype provided several GUIs and some of the functionalities set forth in the LEVRAS Requirements Specification (RS). Potential LEVRAS users previewed their new IS through the program "look and feel" provided by the prototype. The authors took the constructive comments from the NPS Security Department staff, revisited the LEVRAS RS and Data Dictionary once again, and created the final system semantic objects and their attributes. A database structure was produced from these objects, and the needed database functions were programmed. The authors have included many powerful features into the finished product, including single and multitable forms and reports, automatic validity checks and continuous referential integrity service, and the most significant RS element, the ability to query any data maintained in the LEVRAS database, with even the smallest amount of input data.
VI. SDM PHASE IV - CONVERSION AND IMPLEMENTATION

This chapter will elaborate on Chapter III's discussion on the conversion and implementation process. Chapter III discussed several different conversion and implementation methodologies, such as abrupt cutover, parallel, and staged. The Law Enforcement and Vehicle Registration Administration System (LEVRAS) was installed using the parallel conversion methodology. The remainder of this chapter will present the conversion process that the authors and NPS Security personnel experienced.

A. TRAINING

The first step the authors used during the training process was actually letting the end-users provide their inputs for form and menu development. This process was iterative and the end-users gained more knowledge during each session. These inputs are now part of the LEVRAS program. The second step employed was to conduct a LEVRAS briefing for the NPS Security Officer and his staff. This briefing covered the LEVRAS User's Manual, LEVRAS program functionalities, data conversion, local area network operations (the LEVRAS program source code was revised to implement a read-only lock when in a multiuser mode), and maintenance points of contact. This briefing was conducted in an open forum setting to best facilitate questions and user enthusiasm for the new system.

At the end of the LEVRAS briefing, the LEVRAS User's Manual was presented to NPS Security personnel for their review, which would enhance the user acceptance during the conversion from COPS to LEVRAS. The authors provided individualized on-the-job-training (OJT) to the end-users during system installation with continuous reference to the LEVRAS User's Manual. The installation process will be discussed in the next section.
B. CONVERSION

The Law Enforcement and Vehicle Registration Administration System used a parallel conversion methodology to convert from the existing system (COPS) to LEVRAS.

Parallel conversion. In parallel conversion, the existing system and the new system operate simultaneously, or in parallel, until the project team is confident that the new system is working properly. Parallel conversion has two important advantages. First, the existing system serves as a backup if the new system fails to operate as expected. Second, the results of the new system can be compared to the results of the existing system. (Long, 1993, p.536)

During the parallel conversion, we reviewed the proposed characteristics of LEVRAS versus COPS as discussed in Chapter II of this thesis. These characteristics explained proposed LEVRAS improvements to combat COPS security vulnerabilities and functionality weaknesses. Referring back to Chapter II, Table 2, the authors inspected the hardware and software upgrades as follows:

- **Specific content of information outputs.** LEVRAS now includes customized and standard reports, as well as hard copy query responses.

- **Selectivity.** The new system now has an advanced query capability to include a selection of desired fields, and use of wildcards (if incomplete data is available).

- **Time lags.** The NPS Security Officer was made aware of his department's slow computing processing time. He upgraded all hardware to 80486 CPUs, with laser printers. Software was also upgraded to include MicroSoft Windows 3.11 and Paradox 4.5 for Windows, which both increased the speed of the old system software - a MS-DOS based Dbase product.
Accuracy of outputs. LEVRAS increased the accuracy of outputs by including referential integrity and validity checks during the source code programming. The referential integrity linked all tables together so that all like information is standard throughout the database. Validity checks guard against improper field value entry, such as preventing letters from being entered into number only fields.

Reliability. The new system hardware includes a 340 MB internal hard drive on each workstation (two workstations are presently being used with plans to expand), which increases storage space for on-line processing and data backups. In addition, LEVRAS provides a means to save old customer data in its archive mode. The authors reminded the NPS Security Officer to set aside fiscal year 96 appropriations for UPS equipment. An UPS will provide NPS Security more reliability during power outages.

Generality. Basically, COPS is comprised of two "stovepipe" systems, which can not share its computerized information without physically transferring data via a 5.25" floppy diskette. LEVRAS provides continuous on-line sharing of all data. Also, any system function can be performed at any workstation. LEVRAS was designed to be general enough for any level of administration, from the NPS Security Officer to the newest Security Department Staff member.

Flexibility. The LEVRAS source code was provided to the MIS maintenance group located in the NPS Herrmann Hall building. This source code can be modified to accommodate for LEVRAS future growth.
After agreeing that the current facilities were ready to support conversion, the authors and the users commenced LEVRAS installation in accordance with LEVRAS User's Manual (See Appendix J) procedures. All system files were successfully loaded. COPS functions were checked to ensure availability during the parallel conversion. Live data entry will be discussed in the following section.

C. SYSTEM ACCEPTANCE TEST

Prior to the NPS Security Department performing their system testing, the authors tested and debugged program errors. The authors also selected two unbiased people to test the LEVRAS program for useability and system flaws. In addition, the authors' thesis advisors reviewed and critiqued the final product. Minor modifications were included as a result of this review and critique.

The LEVRAS program was then tested by the end-users (after their initial training session). Actual customer records were entered and manipulated by the users under the supervision of the authors. New customers arrived, and LEVRAS successfully supported all required data storage and decal issue procedures. During lulls in the action, COPS data was retrieved and entered into LEVRAS. The NPS Security Officer monitored this process and was satisfied with the system testing results. The system met all of Security Department requirements to include: password security, GUI friendliness, data accuracy, and query capability. With the concurrence of the NPS Security Officer, LEVRAS was turned over to the NPS Security Department for normal VIRO and Ticket Administration Office operations.
VII. SDM PHASE V - POST IMPLEMENTATION

This chapter will discuss how LEVRAS documentation and system maintenance will be handled during its system life cycle. The authors wrote a comprehensive LEVRAS User's Manual (Appendix J) to support the NPS Security Department. In addition, the authors made arrangements with NPS Management Information System (MIS) computer support personnel to maintain LEVRAS upon the authors departure from NPS.

A. DOCUMENTATION

As previously stated, a LEVRAS User's Manual was written to support daily operations, as well as providing references for maintenance support. The LEVRAS User's Manual specifically addresses the following:

Points of contact,
Software development,
Deliverables,
Diskette files list,
Hardware and software requirements,
Installing LEVRAS,
System callup and passwords,
Log-out and power-down procedures,
Housekeeping procedures,
Error messages,
Tutorial for menus, data entry forms, reports, queries, and archive,
LEVRAS codes, and
References.
The references mentioned above refer the LEVRAS users to *Paradox 4.5 for Windows* and *MicroSoft Windows* software manuals for support of these products. In addition, other references will provide support to *Paradox* for any future upgrades to the LEVRAS program.

**B. SYSTEM MAINTENANCE**

Keeping within the standardized SDM procedures and lifecycle management process, maintenance support is available for LEVRAS. Specifically, the authors will maintain the LEVRAS software program until September 1995. Thereafter, LEVRAS will be maintained by the Management Information System computer specialists located in Herrmann Hall, room E-204, phone (408) 656-2195.

The LEVRAS program script was provided to the above computer specialists for future upgrades and program system maintenance. For example, a LEVRAS menu could be modified to include another pushbutton for a new report. The actual LEVRAS menu would be redesigned by using the *.FSL* master file associated with that particular menu to be modified. Then the pushbutton would be built and added onto the menu in the designated area. Once the pushbutton is built and its label applied, the source code would be attached to the pushbutton (use a previously built pushbutton within the LEVRAS source code as a guide). The source code would direct the pushbutton to the new report (*.RSL* master file). The report would then be built to accommodate the user's data requirements. This report would be built by selecting New Report in *Paradox*, and following *Paradox* manuals referred to in the LEVRAS User's Manual.

Once the new functionalities are constructed, system maintainers would need to document all of their code (they can also document their source code as it is being built, which is a better way of writing code) within the source code itself. Once the source code is documented, the LEVRAS system maintainers would then print this modification or upgrade and add it to the original source code (provided by the authors).
LEVRAS system files that were provided to the NPS Security Department and the MIS computer specialists, would then need to be updated to reflect the new changes or additions. These files would then need to be installed within the operating LEVRAS program and tested as discussed in Chapter VI of this thesis. The files should be maintained on a 3.5" floppy diskette and stored in a safe location.

In the case of system crashes, the stored 3.5" LEVRAS program diskette can be used to re-install the program. Refer to LEVRAS User's Manual for program installation (Appendix J). Once the LEVRAS software program is loaded, the LEVRAS user would need to install his or her data (located in the LEVRAS tables, *.DB working files). If by chance the working data tables are destroyed (*.DB working files) the daily backup files would be used to recover all of the destroyed data. This is why it is crucial for users to routinely backup their LEVRAS files, as part of their system maintenance program.
VIII. CONCLUSIONS AND LESSONS LEARNED

The Law Enforcement and Vehicle Registration Administration System (LEVRAS) was developed for the Naval Postgraduate School (NPS), Monterey, California. This management information system (MIS) replaces the Computer Online Police System (COPS), which was plagued by typical legacy computer system problems, such as old, slow hardware and software riddled with functionality gaps. The NPS Security Department staff now performs its daily vehicle registration and ticket administration operations using LEVRAS. The authors' original interest in designing and developing LEVRAS began back in October 1993, when a child abduction attempt in LA Mesa Village (military housing) was reported to NPS Police. As concerned parents living in La Mesa, we thought that the lengthy process to find the culprit was unsatisfactory. Therefore, the authors offered their information system (IS) services to the NPS Security Officer to help alleviate this problem. The result of the authors' nearly two year effort was this thesis, and a fully operationally relational database program, LEVRAS. In addition, the authors provided extremely beneficial suggestions to improve the Security Department's IS. Most of these suggestions were funded and acted upon by the NPS Security Officer. A casual visit to the NPS Police Headquarters will reveal a dramatic change from past IS operations to include today's computer hardware and software technology.

All of the Naval Postgraduate School Security Department's primary functional requirement specifications were met by LEVRAS including:

- Capability to input, store, retrieve, update and delete appropriate law enforcement and vehicular information.
- Provide basic and selective query capability for active and archive data.
- Provide reports containing accurate and complete vehicular, owner, decal and ticket information.
- Input and display system data for decal and temporary pass issue.
- Input and display system data for ticket disposition.
LEVRAS was installed and tested by the users. The NPS Security Officer and his staff gladly accepted LEVRAS as a replacement for their antiquated existing IS, since it proved to be a dramatic improvement especially in the areas of query capability and archiving old customer data.

In the first quarter of the Information Technology Management (ITM) curriculum at NPS, the *Introduction to Computer Management* (IS-2000) course was instrumental in providing the basics of the System Development Methodology (SDM). This methodology provided the project team foundation that was a crucial element in making this thesis process a success. Acting as an actual software development team, the authors learned many lessons along the way. Some of the major lessons learned include:

- The Semantic Object Modeling with Salsa book used in the thesis was incomplete. Specifically, the authors did not know how to generate a schema using the element known as "data" vice the "_ID" element. This was a crucial part of making the entire database schema operate properly. Just by a stroke of luck, the authors discovered this flaw (after two full days of work), enabling them to generate a fully functional schema.

- Another two days of work was lost due to the inadvertent omission of a single required data field (with approximately a hundred semantic object attributes, this type of oversight is easy to make). This caused the entire LEVRAS database schema to be re-generated in *SALSA*, and required the application development in *Paradox* to start from scratch.

- ObjectPal (the fourth generation language provided in *Paradox*) was much harder to program than originally expected. This lengthened the anticipated project software development schedule by 25%.
The manuals provided with Paradox and after market ObjectPal programming books did not clearly present the coding methods needed to program LEVRAS. This resulted in code that did not function as expected, and many iterative programming re-writes were needed to make the functionalities work properly.

Another two days was spent commenting the source code for future maintenance or upgrades. The authors did not comment the source code as LEVRAS was being developed, but instead went back after the program was fully operational.

After developing many LEVRAS menus and forms, the authors discovered that similar pushbuttons were located in different places on the different screens. This even made the authors confused to the point of vertigo. Realignment of these pushbuttons to the exact location on each screen solved this problem.

Leaving this thesis on a positive note, the authors suggest that software developers choose project team members carefully. Project team members should be compatible both professionally and personally, so that an approach such as the Delphi Method can be successfully implemented without any hard feelings or grudges.

Finally, the authors effectively used the knowledge gained in their NPS graduate ITM program, by creating a MIS focusing around a relational database. The bottom line result of these efforts is improved NPS Security force that can immediately respond to the smallest amount of data on military installation intruders. La Mesa Village residents now feel more secure knowing that LEVRAS is on their side!
APPENDIX A. REQUIREMENTS SPECIFICATION

This appendix summarizes the top level performance requirements for LEVRAS in quantifiable terms. It contains functional and nonfunctional requirements. A functional requirement is defined as a detailed description of data inputs, processes, and outputs that must be made. A nonfunctional requirement is not a full required function, but rather a system characteristic that would enhance the existing functions.

Primary Functional Requirements

1. Input, store, retrieve, update and delete appropriate law enforcement and vehicular information (complete vehicular, owner, decal, and ticket violation information).

2. Provide basic and advanced query capability as follows:
   a. Basic - Queries involving entire fields (e.g., last name).
   b. Advanced - Queries involving portions of fields (e.g., last name with the second letter of "t").

3. Provide reports (hard-copy output) containing accurate and complete vehicular, owner, decal and ticket information on demand.

4. Input and display system data for decal and temporary pass issue.

5. Input and display system data for ticket disposition.

Secondary Functional Requirements

1. Import data from the COPS system via 5.25 inch floppy disk.

2. Import data from off-campus sources via 5.25 or 3.5 inch floppy disk or modem (e.g., California Law Enforcement Terminal System (CLATS) and Interstate Law Enforcement Terminal System (INLETS)).

3. Be capable of photographic data entry (e.g., a scanning device).

4. Have anti-virus software protection.
5. Generate a daily security violator(s) listing to be distributed to Security Officer and NPS gate guards.

6. Provide password protection in accordance with NPS Security Department regulations.

7. Have a backup method. The backup medium will be kept physically separate from the main secondary storage device, allowing for backups of all data and schema each eight hour shift (i.e., three times daily). The backup media should be able to be removed from the system for transport or storage away from the system.

Non-Functional Requirements

1. Provide a database size to accommodate a minimum of 10,000 vehicles (accompanied by all support data such as owner, decal and ticket information).

2. Provide a response time of a maximum of 30 seconds for basic queries and a maximum of 60 seconds for advanced queries.

3. Update a single vehicle record with a maximum response time of 10 seconds.

4. Provide a user-friendly interface that an individual with only minimum computer experience can effectively use. Also, training will be provided by the authors.

5. Operate on-line 24-hours a day due to real-time access required by NPS law enforcement.

6. Provide high quality print capability for reports (hard-copy outputs).

7. Implement a scheduled periodic and corrective system maintenance plan.

8. Install a LAN for VIRO and NPS Police.

9. Use state-of-the-art COTS hardware and software.

10. Provide a system user's manual to address system administrative procedures.
APPENDIX B. DATA FLOW DIAGRAMS

This appendix provides LEVRAS Data Flow Diagrams (DFDs), starting with the LEVRAS Decomposition Diagram. The Decomposition Diagram is exploded (decomposed) into the overall LEVRAS Context Diagram (First Level). This First Level Diagram is exploded into a Second Level Diagram with five processes (PROCESS CUSTOMER CHECKIN, PROCESS CUSTOMER CHECKOUT, GENERATE QUERY RESPONSE, PROCESS TICKETS, and GENERATE REPORTS). The Second Level Processes are further exploded into five primitive Third Level Diagrams with 15 processes (INPUT CUSTOMER DATA, MODIFY CUSTOMER DATA, GENERATE VEHICLE DECAL, GENERATE TEMPORARY PASS, PROCESS DATA TRANS, ARCHIVE CUSTOMER DATA, VALIDATE REQUEST DATA, GENERATE QUERY RESPONSE, INPUT TICKET DATA, PROCESS TICKET DISPO, ARCHIVE TICKET DATA, GENERATE TICKET REPORT, GENERATE DCL/PASS REPORT, GENERATE CUST/VEH REPORT and GENERATE CUSTOM REPORT). LEVRAS External Entities (CUSTOMER, SECURITY OFFICER, and POLICEMAN) along with LEVRAS Data Stores (CUSTOMER AND VEHICLE DATA, TICKET DATA, and ARCHIVE DATA) are appropriately placed throughout LEVRAS DFD Levels, as displayed in the following pages.
Second Level DFD
APPENDIX C. DATA DICTIONARY

This LEVRAS Data Dictionary contains descriptions of the components of the primitive DFDs contained in Appendix B, and as discussed in Chapters III and IV. This appendix will document the external entities, processes, and data stores. Data flows are connections between the external entities, processes, and data stores; furthermore, they are "common sense" connections and will be presented within the context of the other aforementioned DFD components.

EXTERNAL ENTITIES

CUSTOMER:  An NPS student, staff, or a contractor provides data to the system through Process 1 (PROCESS CUSTOMER CHECKIN) and Process 2 (PROCESS CUSTOMER CHECKOUT). CUSTOMER information populates and updates two of the data stores (CUSTOMER AND VEHICLE DATA and ARCHIVE DATA) to provide current information to the SECURITY OFFICER.

SECURITY OFFICER:  The NPS Security Officer (or his staff) represents the initiation of control processes in the system to specifically request and receive reports, rather than an item to maintain data on. LEVRAS can provide current information to the SECURITY OFFICER upon request. The SECURITY OFFICER interfaces with the system through Process 3 (GENERATE QUERY RESPONSE), Process 4 (PROCESS TICKETS), and Process 5 (GENERATE REPORTS).

POLICEMAN:  An NPS Police Officer represents the input of ticket data, rather than an item to maintain data on. The POLICEMAN interfaces with the system only through Process 4 (PROCESS TICKET). Ticket information from the POLICEMAN populates and updates the TICKET DATA data store.
PROCESSES

1.1P INPUT CUSTOMER DATA: Handles the input of CUSTOMER data during checkin. Interfaces with one external entity (CUSTOMER), and one data store (CUSTOMER AND VEHICLE DATA).

1.2P MODIFY CUSTOMER DATA: Handles the modification of CUSTOMER data during checkin. Interfaces with one external entity (CUSTOMER), and one data store (CUSTOMER AND VEHICLE DATA).

1.3P GENERATE VEHICLE DECAL: Handles the decal assignment during CUSTOMER checkin, when directed by Process 1.1P. Interfaces with one external entity (CUSTOMER), and one data store (CUSTOMER AND VEHICLE DATA).

1.4P GENERATE TEMP PASS: Generates a temporary pass during CUSTOMER checkin, when directed by Process 1.1P. Interfaces with one external entity (CUSTOMER), and one data store (CUSTOMER AND VEHICLE DATA).

2.1P PROCESS DATA TRANS: Processes data transfer information during CUSTOMER checkout. Interfaces with two external entities (CUSTOMER and SECURITY OFFICER), and one data store (CUSTOMER AND VEHICLE DATA).

2.2P ARCHIVE CUSTOMER DATA: Archives CUSTOMER data during checkout. Interfaces with one external entity (SECURITY OFFICER), and one data store (ARCHIVE DATA).

3.1P VALIDATE REQUEST DATA: Validates requested data during query response. Interfaces with one external entity (SECURITY OFFICER), and all three data stores (CUSTOMER AND VEHICLE DATA, TICKET DATA, and ARCHIVE DATA).
3.2P GENERATE QUERY RESPONSE: Generates query response when directed by process 3.1P. Interfaces with one external entity (SECURITY OFFICER), and no data stores.

4.1P INPUT TICKET DATA: Facilitates the input of ticket data for ticket processing. Interfaces with one external entity (POLICEMAN), and one data store (TICKET DATA).

4.2P PROCESS TICKET DISPO: Processes ticket disposition. Interfaces with one external entity (SECURITY OFFICER), and one data store (TICKET DATA).

4.3P ARCHIVE TICKET DATA: Archives ticket data at the completion of ticket processing as directed by Process 4.2P. Interfaces with no entities, and one data store (ARCHIVE DATA).

5.1P GENERATE TICKET REPORT: Generates routine ticket reports. Interfaces with one external entity (SECURITY OFFICER), and one data store (TICKET DATA).

5.2P GENERATE DCL/PASS REPORT: Generates routine decal and pass reports. Interfaces with one external entity (SECURITY OFFICER), and one data store (CUSTOMER AND VEHICLE DATA).

5.3P GENERATE CUST/VEH REPORT: Generates routine CUSTOMER and vehicle reports. Interfaces with one external entity (SECURITY OFFICER), and one data store (CUSTOMER AND VEHICLE DATA).

5.4P GENERATE CUSTOM REPORT: Generates customized security reports for the SECURITY OFFICER. Interfaces with one external entity (SECURITY OFFICER), and all three data stores (CUSTOMER AND VEHICLE DATA, TICKET DATA, and ARCHIVE DATA).
DATA STORES

CUSTOMER AND VEHICLE DATA: Contains a group of attributes concerning CUSTOMER and vehicle data (including but not limited to customer's name and address, vehicle ID number and model, and insurance policy number).

TICKET DATA: Contains a group of attributes concerning ticket data (including but not limited to ticket number, date, and traffic violation code).

ARCHIVE DATA: Contains a combination of the data attributes of the two above data stores. The active data is contained within CUSTOMER AND VEHICLE DATA and TICKET DATA data stores while the inactive data is stored within the ARCHIVE DATA data store. Data is stored in the ARCHIVE DATA data store only after being removed from one or both of the other two data stores aforementioned.
APPENDIX D. SEMANTIC OBJECTS

This appendix contains printouts of the seven semantic objects designed for the LEVRAS database. The attributes, and the cardinality of each attribute, are listed for each object. Chapter five of this thesis discusses the process used to construct these data models. Descriptions of the attributes and object relationships follow the two pages of semantic object models.
# LEVRAS SEMANTIC OBJECTS

## VEHICLE

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LicensePlateNumber</td>
<td>1.1</td>
</tr>
<tr>
<td>LicensePlateState</td>
<td>1.1</td>
</tr>
<tr>
<td>VehicleIdentificationNumber</td>
<td>1.1</td>
</tr>
<tr>
<td>VehicleMake</td>
<td>1.1</td>
</tr>
<tr>
<td>VehicleColor</td>
<td>1.1</td>
</tr>
<tr>
<td>VehicleType</td>
<td>1.1</td>
</tr>
<tr>
<td>VehicleYear</td>
<td>1.1</td>
</tr>
<tr>
<td>CUSTOMER</td>
<td>1.1</td>
</tr>
<tr>
<td>REGISTRATION</td>
<td>1.1</td>
</tr>
<tr>
<td>INSURANCE</td>
<td>1.1</td>
</tr>
<tr>
<td>DECAL</td>
<td>1.1</td>
</tr>
<tr>
<td>TICKET</td>
<td>0..N</td>
</tr>
</tbody>
</table>

## DECAL

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecalNumber</td>
<td>1.1</td>
</tr>
<tr>
<td>DecalColor</td>
<td>1.1</td>
</tr>
<tr>
<td>DecalIssueDate</td>
<td>1.1</td>
</tr>
<tr>
<td>DecalExpirationMonth</td>
<td>1.1</td>
</tr>
<tr>
<td>DecalExpirationYear</td>
<td>1.1</td>
</tr>
<tr>
<td>CUSTOMER</td>
<td>1.1</td>
</tr>
<tr>
<td>VEHICLE</td>
<td>1.1</td>
</tr>
</tbody>
</table>

## TICKET

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TicketNumber</td>
<td>1.1</td>
</tr>
<tr>
<td>PolicemanName</td>
<td>1.1</td>
</tr>
<tr>
<td>ViolationCode</td>
<td>1.1</td>
</tr>
<tr>
<td>ViolationDate</td>
<td>1.1</td>
</tr>
<tr>
<td>BaseCourtDate</td>
<td>1.1</td>
</tr>
<tr>
<td>BaseDisposition</td>
<td>0..1</td>
</tr>
<tr>
<td>BaseDispositionDate</td>
<td>0..1</td>
</tr>
<tr>
<td>BaseJudgeName</td>
<td>0..1</td>
</tr>
<tr>
<td>Points</td>
<td>0..1</td>
</tr>
<tr>
<td>TransactionDate</td>
<td>1.1</td>
</tr>
<tr>
<td>CUSTOMER</td>
<td>1.1</td>
</tr>
<tr>
<td>VEHICLE</td>
<td>0..1</td>
</tr>
</tbody>
</table>
APPENDIX E. DATABASE TABLES

This appendix contains the LEVRAS database tables which were created in the schema generation process. Chapter five of this thesis discusses generation of database schema, and the semantic modelling steps that occur prior to it. One database table was created for each of the LEVRAS semantic objects, which can be found in Appendix F.
# Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: BaseCourtDate</th>
<th>Contained in: TICKET</th>
<th>Caption: CrtDate</th>
<th>Description: BaseCourtDate</th>
<th>ID Status: None</th>
<th>Minimum Required: 1</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: Date</th>
<th>Length:</th>
<th>Format: DD/MM/YY</th>
<th>Initial Value:</th>
</tr>
</thead>
</table>

| Attribute          | Type: Simple Value | Profile: BaseDisposition | Contained in: TICKET | Caption: Disposn | Description: BaseDisposition | ID Status: None | Minimum Required: 0 | Maximum Allowed: 1 | Value Type: Text | Length: 30 | Format: Warning | Initial Value: |

| Attribute          | Type: Simple Value | Profile: BaseDispositionDate | Contained in: TICKET | Caption: DispDt | Description: BaseDispositionDate | ID Status: None | Minimum Required: 0 | Maximum Allowed: 1 | Value Type: Date | Length: | Format: MM/DD/YY | Initial Value: |

| Attribute          | Type: Simple Value | Profile: BaseJudgeName | Contained in: TICKET | Caption: JudgeName | Description: BaseJudgeName | ID Status: None | Minimum Required: 0 | Maximum Allowed: 1 | Value Type: Text | Length: 10 | Format: McGibbon | Initial Value: |
### Attribute Report

#### Album: LEVRAS.ALB

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Profile</th>
<th>Contained in</th>
<th>Caption</th>
<th>Description</th>
<th>ID Status</th>
<th>Minimum Required</th>
<th>Maximum Allowed</th>
<th>Value Type</th>
<th>Length</th>
<th>Format</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curric/Staff</td>
<td>Simple Value</td>
<td>Curric/Staff</td>
<td>CUSTOMER</td>
<td>Curric/Staff</td>
<td>Curric/StaffCode</td>
<td>None</td>
<td>0</td>
<td>1</td>
<td>Text</td>
<td>3</td>
<td>123</td>
<td></td>
</tr>
</tbody>
</table>

#### CUSTOMER

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Profile</th>
<th>Contained in</th>
<th>Caption</th>
<th>Description</th>
<th>ID Status</th>
<th>Minimum Required</th>
<th>Maximum Allowed</th>
<th>Value Type</th>
<th>Length</th>
<th>Format</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Object Link</td>
<td>CUSTOMER</td>
<td>DECAL</td>
<td>CUSTOMER</td>
<td></td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Object Link</td>
<td>CUSTOMER</td>
<td>INSURANCE</td>
<td>CUSTOMER</td>
<td></td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Object Link</td>
<td>CUSTOMER</td>
<td>TICKET</td>
<td>CUSTOMER</td>
<td></td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Object Link</td>
<td>CUSTOMER</td>
<td>DRIVERLICENSE</td>
<td>CUSTOMER</td>
<td></td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Attribute Report
Album: LEVRAS.ALB

CUSTOMER
Type: Object Link
Profile: CUSTOMER
Contained in: VEHICLE
Caption: CUSTOMER
Description: CUSTOMER
ID Status: None
Minimum Required: 1
Maximum Allowed: 1

CUSTOMER
Type: Object Link
Profile: CUSTOMER
Contained in: REGISTRATION
Caption: CUSTOMER
Description: CUSTOMER
ID Status: None
Minimum Required: 1
Maximum Allowed: 1

DatabaseEntryDate
Type: Simple Value
Profile: TransactionDate
Contained in: CUSTOMER
Caption: DatabaseEntryDate
Description: DatabaseEntryDate
ID Status: None
Minimum Required: 1
Maximum Allowed: 1
Value Type: Date
Length:
Format: MM/DD/YY
Initial Value:

DECAL
Type: Object Link
Profile: DECAL
Contained in: VEHICLE
Caption: DECAL
Description: DECAL
ID Status: None
Minimum Required: 1
Maximum Allowed: 1

DECAL
Type: Object Link
Profile: DECAL
Contained in: CUSTOMER
Caption: DECAL
Description: DECALINFO
ID Status: None
Minimum Required: 1
Maximum Allowed: N (No Limit)
### Attribute Report
**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Profile</th>
<th>Contained in</th>
<th>Caption</th>
<th>Description</th>
<th>ID Status</th>
<th>Minimum Required</th>
<th>Maximum Allowed</th>
<th>Value Type</th>
<th>Length</th>
<th>Format</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecalColor</td>
<td>Simple Value</td>
<td>DecalColor</td>
<td>DECAL</td>
<td>DclClr</td>
<td>DecalColor</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DecalExpirationMonth</td>
<td>Simple Value</td>
<td>DecalExpirationMonth</td>
<td>DECAL</td>
<td>DotExpMo</td>
<td>DecalExpirationMonth</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>2</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>DecalExpirationYear</td>
<td>Simple Value</td>
<td>DecalExpirationYear</td>
<td>DECAL</td>
<td>DllExpYr</td>
<td>DecalExpirationYear</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>2</td>
<td>95</td>
<td></td>
</tr>
<tr>
<td>DecalIssueDate</td>
<td>Simple Value</td>
<td>DecalIssueDate</td>
<td>DECAL</td>
<td>DclIssueDt</td>
<td>DecalIssueDate</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: DecalNumber</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecalNumber</td>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: DSN#</td>
<td>Description: DecalNumber</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 6</td>
<td>Format: ABC123</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: DecalNumber</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecalNumber</td>
<td></td>
<td>Contained in: DECAL</td>
</tr>
<tr>
<td></td>
<td>Caption: DSN#</td>
<td>Description: DecalNumber</td>
</tr>
<tr>
<td></td>
<td>ID Status: Unique</td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 6</td>
<td>Format: ABC123</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Object Link</th>
<th>Profile: DRIVERLICENSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVERLICENSE</td>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: DRIVER LICENSE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: DutyStation</th>
</tr>
</thead>
<tbody>
<tr>
<td>DutyStation</td>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: DutyStation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
<td>Minimum Required: 0</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 10</td>
<td>Format: NPS</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
<td></td>
</tr>
</tbody>
</table>
Attribute Report
Album: LEVRAS.ALB

EmployeeType
Type: Simple Value
Profile: EmployeeType
Contained in: CUSTOMER
Caption: EmployeeType
Description: EmployeeType
ID Status: None
Minimum Required: 1
Maximum Allowed: 1
Value Type: Text
Length: 10
Format: ABCDEFGHIJ
Initial Value:

ExpirationDate
Type: Simple Value
Profile: ExpirationDate
Contained in: REGISTRATION
Caption: RegExpDt
Description: RegistrationExpDate
ID Status: None
Minimum Required: 1
Maximum Allowed: 1
Value Type: Date
Length:
Format: MM/DD/YY
Initial Value:

ExpirationDate
Type: Simple Value
Profile: ExpirationDate
Contained in: INSURANCE
Caption: ExpDt
Description: ExpirationDate
ID Status: None
Minimum Required: 1
Maximum Allowed: 1
Value Type: Date
Length:
Format: MM/DD/YY
Initial Value:

ExpirationDate
Type: Simple Value
Profile: ExpirationDate
Contained in: DRIVERLICENSE
Caption: ExpDt
Description: LicenseExpirationDate
ID Status: None
Minimum Required: 1
Maximum Allowed: 1
Value Type: Date
Length:
Format: MM/DD/YY
Initial Value:
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Profile</th>
<th>Contained in</th>
<th>Caption</th>
<th>Description</th>
<th>ID Status</th>
<th>Minimum Required</th>
<th>Maximum Allowed</th>
<th>Value Type</th>
<th>Length</th>
<th>Format</th>
<th>Initial Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faculty</td>
<td>Simple Value</td>
<td>Faculty</td>
<td>CUSTOMER</td>
<td>Faculty</td>
<td>FacultyDepartment</td>
<td>None</td>
<td>0</td>
<td>1</td>
<td>Text</td>
<td>2</td>
<td>AB</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>Simple Value</td>
<td>FirstName</td>
<td>CUSTOMER</td>
<td>FirstName</td>
<td>FirstName</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>10</td>
<td>ABCDEFGHIJ</td>
<td></td>
</tr>
<tr>
<td>Grade</td>
<td>Simple Value</td>
<td>Grade</td>
<td>CUSTOMER</td>
<td>Grade</td>
<td>EmployeeGrade</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>4</td>
<td>GS13</td>
<td></td>
</tr>
<tr>
<td>HomeAddress</td>
<td>Simple Value</td>
<td>HomeAddress</td>
<td>CUSTOMER</td>
<td>HomeAddress</td>
<td>StreetNumberAndStreetName</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>30</td>
<td>1234567890</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Type: Simple Value</td>
<td>Profile:</td>
<td>Contained in: CUSTOMER</td>
<td>Caption:</td>
<td>Description:</td>
<td>ID Status: None</td>
<td>Minimum Required: 1</td>
<td>Maximum Allowed: 1</td>
<td>Value Type: Text</td>
<td>Length: 17</td>
<td>Format: ABCDEFGHIJKLMNOPQ</td>
<td>Initial Value:</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------</td>
<td>-------------------</td>
<td>------------------------</td>
<td>-------------------</td>
<td>---------------------</td>
<td>-----------------</td>
<td>------------------</td>
<td>------------------</td>
<td>------------------</td>
<td>-----------</td>
<td>----------------------------</td>
<td>----------------</td>
</tr>
</tbody>
</table>

| HomeCity      |                    | City              |                       | City              | City                | None            | 1                | 1                | Text             | 17        | ABCDEFGHIJKLMNOPQ            |                |
| HomePhone     |                    | HomePhone         | CUSTOMER              | HomePhone         | HomePhoneNumber     | None            | 1                | 1                | Text             | 10        | 4086551234                  |                |
| HomeState     |                    | State             | CUSTOMER              | State             | 2DigitStateCode     | None            | 1                | 1                | Text             | 2         | CA                         |                |
| HomeZip       |                    | Zip               | CUSTOMER              | HomeZipCode       | HomeZipCode         | None            | 1                | 1                | Text             | 5         | 93940                       |                |
# Attribute Report

## Album: LEVRAS ALB

### INSURANCE

**Type:** Object Link  
**Profile:** INSURANCE  
**Contained in:** CUSTOMER  
**Caption:** INSURANCE  
**Description:** INSURANCE  
**ID Status:** None  
**Minimum Required:** 1  
**Maximum Allowed:** N (No Limit)

### INSURANCE

**Type:** Object Link  
**Profile:** INSURANCE  
**Contained in:** CUSTOMER  
**Caption:** INSURANCE  
**Description:** INSURANCE  
**ID Status:** None  
**Minimum Required:** 1  
**Maximum Allowed:** 1

### InsuranceCompanyNam

**Type:** Simple Value  
**Profile:** InsuranceCompanyNam  
**Contained in:** INSURANCE  
**Caption:** InsCo  
**Description:** InsuranceCompanyNam  
**ID Status:** None  
**Minimum Required:** 1  
**Maximum Allowed:** 1  
**Value Type:** Text  
**Length:** 15  
**Format:** Abcdefghijklmno  
**Initial Value:**

### InsurancePolicyNumber

**Type:** Simple Value  
**Profile:** InsurancePolicyNumber  
**Contained in:** CUSTOMER  
**Caption:** InsPo#  
**Description:** InsurancePolicyNumber  
**ID Status:** None  
**Minimum Required:** 1  
**Maximum Allowed:** 1  
**Value Type:** Text  
**Length:** 20  
**Format:** Abcdefghij0123456789  
**Initial Value:**

### InsurancePolicyNumber

**Type:** Simple Value  
**Profile:** InsurancePolicyNumber  
**Contained in:** INSURANCE  
**Caption:** InsPo#  
**Description:** InsurancePolicyNumber  
**ID Status:** Unique  
**Minimum Required:** 1  
**Maximum Allowed:** 1  
**Value Type:** Text  
**Length:** 20  
**Format:** Abcdefghij0123456789  
**Initial Value:**

86
### Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile:</th>
<th>Contained in:</th>
<th>Caption:</th>
<th>Description:</th>
<th>ID Status:</th>
<th>Minimum Required:</th>
<th>Maximum Allowed:</th>
<th>Value Type:</th>
<th>Length:</th>
<th>Format:</th>
<th>Initial Value:</th>
</tr>
</thead>
<tbody>
<tr>
<td>LastName</td>
<td></td>
<td>LastName</td>
<td>CUSTOMER</td>
<td>LastName</td>
<td>FirstName</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>15</td>
<td>ABCDEFGHIJKLMNOP</td>
<td></td>
</tr>
<tr>
<td>LicenseNumber</td>
<td></td>
<td>LicenseNumber</td>
<td>DRIVERLICENSE</td>
<td>License#</td>
<td>DriverLicenseNumber</td>
<td>Unique</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>10</td>
<td>ZZ28123ABD</td>
<td></td>
</tr>
<tr>
<td>LicenseNumber</td>
<td></td>
<td>LicenseNumber</td>
<td>CUSTOMER</td>
<td>License#</td>
<td>DriverLicenseNumber</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>10</td>
<td>ZZ28123ABD</td>
<td></td>
</tr>
<tr>
<td>LicensePlateNumber</td>
<td></td>
<td>LicensePlateNumber</td>
<td>CUSTOMER</td>
<td>Plate#</td>
<td>LicensePlateNumber</td>
<td>None</td>
<td>1</td>
<td>1</td>
<td>Text</td>
<td>10</td>
<td>ABCDE12345</td>
<td></td>
</tr>
</tbody>
</table>
## Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: LicensePlateNumber_786434</th>
<th>Contained in: VEHICLE</th>
<th>Caption: Plate#</th>
<th>Description: LicensePlateNumber</th>
<th>ID Status: Unique</th>
<th>Minimum Required: 1</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: Text</th>
<th>Length: 10</th>
<th>Format: ABCDE12345</th>
<th>Initial Value:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: LicensePlateState</th>
<th>Contained in: VEHICLE</th>
<th>Caption: PlateST</th>
<th>Description: LicensePlateState</th>
<th>ID Status: None</th>
<th>Minimum Required: 1</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: Text</th>
<th>Length: 2</th>
<th>Format: MA</th>
<th>Initial Value:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: LicenseState</th>
<th>Contained in: DRIVERLICENSE</th>
<th>Caption: LicenseST</th>
<th>Description: 2DigitStateCode</th>
<th>ID Status: None</th>
<th>Minimum Required: 1</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: Text</th>
<th>Length: 2</th>
<th>Format: CA</th>
<th>Initial Value:</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: MiddleInitial</th>
<th>Contained in: CUSTOMER</th>
<th>Caption: MI</th>
<th>Description: MiddleInitial</th>
<th>ID Status: None</th>
<th>Minimum Required: 0</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: Text</th>
<th>Length: 1</th>
<th>Format: A</th>
<th>Initial Value:</th>
</tr>
</thead>
</table>
## Attribute Report
### Album: LEVRAS ALB

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points</td>
<td>Profile: Points</td>
</tr>
<tr>
<td></td>
<td>Contained in: TICKET</td>
</tr>
<tr>
<td></td>
<td>Caption: Pt</td>
</tr>
<tr>
<td></td>
<td>Description: DrivingPoints</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 0</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 2</td>
</tr>
<tr>
<td></td>
<td>Format: 12</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PolicemanName</td>
<td>Profile: TicketOfficerName</td>
</tr>
<tr>
<td></td>
<td>Contained in: TICKET</td>
</tr>
<tr>
<td></td>
<td>Caption: PolicemanName</td>
</tr>
<tr>
<td></td>
<td>Description: PolicemanName</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 10</td>
</tr>
<tr>
<td></td>
<td>Format: Jones</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Profile: Rank</td>
</tr>
<tr>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: Rank</td>
</tr>
<tr>
<td></td>
<td>Description: MilitaryRank</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 0</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 3</td>
</tr>
<tr>
<td></td>
<td>Format: E02</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RegistrationNumber</td>
<td>Profile: RegistrationNumber</td>
</tr>
<tr>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: Reg#</td>
</tr>
<tr>
<td></td>
<td>Description: RegistrationNumber</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 10</td>
</tr>
<tr>
<td></td>
<td>Format: ABCDE12345</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>
### Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RegistrationNumber</td>
<td>Profile: RegistrationNumber</td>
</tr>
<tr>
<td></td>
<td>Contained in: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Caption: Reg#</td>
</tr>
<tr>
<td></td>
<td>Description: RegistrationNumber</td>
</tr>
<tr>
<td></td>
<td>ID Status: Unique</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 10</td>
</tr>
<tr>
<td></td>
<td>Format: ABCDE12345</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REGISTRATION</th>
<th>Type: Object Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Profile: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Contained in: VEHICLE</td>
</tr>
<tr>
<td></td>
<td>Caption: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Description: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REGISTRATION</th>
<th>Type: Object Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Profile: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Description: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: N (No Limit)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RegistrationState</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Profile: RegistrationState</td>
</tr>
<tr>
<td></td>
<td>Contained in: REGISTRATION</td>
</tr>
<tr>
<td></td>
<td>Caption: RegST</td>
</tr>
<tr>
<td></td>
<td>Description: 2DigitStateCode</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 2</td>
</tr>
<tr>
<td></td>
<td>Format: CA</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sex</th>
<th>Type: Simple Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Profile: Sex</td>
</tr>
<tr>
<td></td>
<td>Contained in: CUSTOMER</td>
</tr>
<tr>
<td></td>
<td>Caption: Sex</td>
</tr>
<tr>
<td></td>
<td>Description: MaleOrFemale</td>
</tr>
<tr>
<td></td>
<td>ID Status: None</td>
</tr>
<tr>
<td></td>
<td>Minimum Required: 1</td>
</tr>
<tr>
<td></td>
<td>Maximum Allowed: 1</td>
</tr>
<tr>
<td></td>
<td>Value Type: Text</td>
</tr>
<tr>
<td></td>
<td>Length: 6</td>
</tr>
<tr>
<td></td>
<td>Format: Abedef</td>
</tr>
<tr>
<td></td>
<td>Initial Value:</td>
</tr>
</tbody>
</table>
# Attribute Report

**Album: LEVRAS(ALB)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMC#</td>
<td>Simple Value</td>
<td>StudentPostOfficeBox#</td>
</tr>
<tr>
<td>SocialSecurityNumber</td>
<td>Simple Value</td>
<td>SocialSecurityNumber</td>
</tr>
<tr>
<td>TICKET</td>
<td>Object Link</td>
<td>TICKET</td>
</tr>
<tr>
<td>TicketNumber</td>
<td>Simple Value</td>
<td>TicketNumber</td>
</tr>
</tbody>
</table>

### SMC#
- **Type:** Simple Value
- **Profile:** SMC#
- **Contained in:** CUSTOMER
- **Caption:** SMC#
- **Description:** StudentPostOfficeBox#
- **ID Status:** None
- **Minimum Required:** 0
- **Maximum Allowed:** 1
- **Value Type:** Text
- **Length:** 4
- **Format:** 1234
- **Initial Value:**

### SocialSecurityNumber
- **Type:** Simple Value
- **Profile:** SocialSecurityNumber
- **Contained in:** CUSTOMER
- **Caption:** SSN
- **Description:** SocialSecurityNumber
- **ID Status:** Unique
- **Minimum Required:** 1
- **Maximum Allowed:** 1
- **Value Type:** Text
- **Length:** 9
- **Format:** 111223333
- **Initial Value:**

### TICKET
- **Type:** Object Link
- **Profile:** TICKET
- **Contained in:** CUSTOMER
- **Caption:** TICKET
- **Description:** TICKET
- **ID Status:** None
- **Minimum Required:** 0
- **Maximum Allowed:** N (No Limit)

### TICKET
- **Type:** Object Link
- **Profile:** TICKET
- **Contained in:** VEHICLE
- **Caption:** TICKET
- **Description:** TICKET
- **ID Status:** None
- **Minimum Required:** 0
- **Maximum Allowed:** N (No Limit)

### TicketNumber
- **Type:** Simple Value
- **Profile:** TicketNumber
- **Contained in:** TICKET
- **Caption:** TKT#
- **Description:** TicketNumber
- **ID Status:** Unique
- **Minimum Required:** 1
- **Maximum Allowed:** 1
- **Value Type:** Text
- **Length:** 9
- **Format:** A12345678
- **Initial Value:**
## Attribute Report

**Album: LEVRAS.ALB**

### TicketNumber

- **Type:** Simple Value
- **Profile:** TicketNumber
- **Contained in:** CUSTOMER
- **Caption:** TKT#
- **Description:** TicketNumber
- **ID Status:** None
- **Minimum Required:** 1
- **Maximum Allowed:** 1
- **Value Type:** Text
- **Length:** 9
- **Format:** A12345678
- **Initial Value:**

### TransactionDate

- **Type:** Simple Value
- **Profile:** TransactionDate
- **Contained in:** TICKET
- **Caption:** TmDt
- **Description:** TransactionDate
- **ID Status:** None
- **Minimum Required:** 1
- **Maximum Allowed:** 1
- **Value Type:** Date
- **Length:**
- **Format:** MM/DD/YY
- **Initial Value:**

### VEHICLE

- **Type:** Object Link
- **Profile:** VEHICLE
- **Contained in:** CUSTOMER
- **Caption:** VEHICLE
- **Description:** VEHICLE
- **ID Status:** None
- **Minimum Required:** 1
- **Maximum Allowed:** N (No Limit)

### VEHICLE

- **Type:** Object Link
- **Profile:** VEHICLE
- **Contained in:** DECAL
- **Caption:** VEHICLE
- **Description:** VEHICLE
- **ID Status:** None
- **Minimum Required:** 1
- **Maximum Allowed:** 1

### VEHICLE

- **Type:** Object Link
- **Profile:** VEHICLE
- **Contained in:** REGISTRATION
- **Caption:** VEHICLE
- **Description:** VEHICLE
- **ID Status:** None
- **Minimum Required:** 1
- **Maximum Allowed:** 1
# Attribute Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type:</th>
<th>Profile:</th>
<th>Contained in:</th>
<th>Caption:</th>
<th>Description:</th>
<th>ID Status:</th>
<th>Minimum Required:</th>
<th>Maximum Allowed:</th>
</tr>
</thead>
<tbody>
<tr>
<td>VEHICLE</td>
<td>Object Link</td>
<td>VEHICLE</td>
<td>INSURANCE</td>
<td>VEHICLE</td>
<td></td>
<td>None</td>
<td>1</td>
<td>N (No Limit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VEHICLE</td>
<td>Object Link</td>
<td>VEHICLE</td>
<td>TICKET</td>
<td>CUSTOMER</td>
<td></td>
<td>None</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VehicleColor</td>
<td>Simple Value</td>
<td>VehicleColor</td>
<td>VEHICLE</td>
<td>VehColor</td>
<td>VehicleColor</td>
<td>None</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VehicleIdentificationNumber</td>
<td>Simple Value</td>
<td>VehicleIdentificationNumber</td>
<td>VEHICLE</td>
<td>VIN</td>
<td>VehicleIdentificationNumber</td>
<td>None</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VehicleMake</td>
<td>Simple Value</td>
<td>VehicleMake</td>
<td>VEHICLE</td>
<td>Make</td>
<td>VehicleMake</td>
<td>None</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Attribute Report
#### Album: LEVRAS.ALB

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type: Simple Value</th>
<th>Profile: AttributeProfile</th>
<th>Contained in: AlbumProfile</th>
<th>Caption: AttributeCaption</th>
<th>Description: AttributeDescription</th>
<th>ID Status: None</th>
<th>Minimum Required: 1</th>
<th>Maximum Allowed: 1</th>
<th>Value Type: AttributeType</th>
<th>Length:</th>
<th>Format: AttributeFormat</th>
<th>Initial Value:</th>
</tr>
</thead>
<tbody>
<tr>
<td>VehicleType</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VehicleYear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ViolationCode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ViolationDate</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WorkPhoneNumber</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Relationship Report

**Album: LEVRAS.ALB**

<table>
<thead>
<tr>
<th>Table</th>
<th>Relationship Type</th>
<th>Mandatory</th>
<th>Related Table</th>
<th>Foreign Key Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMER</td>
<td>1:N</td>
<td>Yes</td>
<td>REGISTRA</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK1</td>
</tr>
<tr>
<td></td>
<td>1:N</td>
<td>Yes</td>
<td>VEHICLE</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK3</td>
</tr>
<tr>
<td></td>
<td>1:N</td>
<td>Yes</td>
<td>DECAL</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK4</td>
</tr>
<tr>
<td></td>
<td>0:N</td>
<td>No</td>
<td>TICKET</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK6</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>DRIVERLI</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK8</td>
</tr>
<tr>
<td></td>
<td>1:N</td>
<td>Yes</td>
<td>INSURANC</td>
<td>SocialSecurityNumber matches SocialSecurityNumber_FK9</td>
</tr>
<tr>
<td>DECAL</td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK4 matches SocialSecurityNumber</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>VEHICLE</td>
<td>VE_LicensePlateNumber_FK5 matches LicensePlateNumber</td>
</tr>
<tr>
<td>DRIVERLI</td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK8 matches SocialSecurityNumber</td>
</tr>
<tr>
<td>INSURANC</td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK9 matches SocialSecurityNumber</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>VEHICLE</td>
<td>V_LicensePlateNumber_FK10 matches LicensePlateNumber</td>
</tr>
<tr>
<td>REGISTRA</td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK1 matches SocialSecurityNumber</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>VEHICLE</td>
<td>VE_LicensePlateNumber_FK2 matches LicensePlateNumber</td>
</tr>
<tr>
<td>TICKET</td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK6 matches SocialSecurityNumber</td>
</tr>
<tr>
<td></td>
<td>0:1</td>
<td>No</td>
<td>VEHICLE</td>
<td>VE_LicensePlateNumber_FK7 matches LicensePlateNumber</td>
</tr>
<tr>
<td>VEHICLE</td>
<td>1:1</td>
<td>Yes</td>
<td>REGISTRA</td>
<td>LicensePlateNumber matches VE_LicensePlateNumber_FK2</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>CUSTOMER</td>
<td>SocialSecurityNumber_FK3 matches SocialSecurityNumber</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>DECAL</td>
<td>LicensePlateNumber matches VE_LicensePlateNumber_FK5</td>
</tr>
<tr>
<td></td>
<td>0:N</td>
<td>No</td>
<td>TICKET</td>
<td>LicensePlateNumber matches VE_LicensePlateNumber_FK7</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>Yes</td>
<td>INSURANC</td>
<td>LicensePlateNumber matches V_LicensePlateNumber_FK10</td>
</tr>
</tbody>
</table>
# LEVRAS DATABASE

## Table Report

**Album: LEVRAS.ALB**

### CUSTOMER Table

**DBMS Type:** PARADOX for Windows/DOS 4.0+

**Source Object or Attribute:** CUSTOMER Object

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value Type</th>
<th>Length</th>
<th>Null Allowed</th>
<th>Object Attribute Name</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curric/Staff</td>
<td>A</td>
<td>3</td>
<td>Yes</td>
<td>CUSTOMER.Curric/Staff</td>
<td></td>
</tr>
<tr>
<td>DatabaseEntryDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>CUSTOMER.DatabaseEntryDate</td>
<td></td>
</tr>
<tr>
<td>DecalNumber</td>
<td>A</td>
<td>6</td>
<td>No</td>
<td>CUSTOMER.DecalNumber</td>
<td></td>
</tr>
<tr>
<td>DutyStation</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>CUSTOMER.DutyStation</td>
<td></td>
</tr>
<tr>
<td>EmployeeType</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER.EmployeeType</td>
<td></td>
</tr>
<tr>
<td>Faculty</td>
<td>A</td>
<td>2</td>
<td>Yes</td>
<td>CUSTOMER.Faculty</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER.FirstName</td>
<td></td>
</tr>
<tr>
<td>Grade</td>
<td>A</td>
<td>4</td>
<td>No</td>
<td>CUSTOMER.Grade</td>
<td></td>
</tr>
<tr>
<td>HomeAddress</td>
<td>A</td>
<td>30</td>
<td>No</td>
<td>CUSTOMER.HomeAddress</td>
<td></td>
</tr>
<tr>
<td>HomeCity</td>
<td>A</td>
<td>17</td>
<td>No</td>
<td>CUSTOMER.HomeCity</td>
<td></td>
</tr>
<tr>
<td>HomePhone</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER.HomePhone</td>
<td></td>
</tr>
<tr>
<td>HomeState</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>CUSTOMER.HomeState</td>
<td></td>
</tr>
<tr>
<td>HomeZip</td>
<td>A</td>
<td>5</td>
<td>No</td>
<td>CUSTOMER.HomeZip</td>
<td></td>
</tr>
<tr>
<td>InsurancePolicyNumber</td>
<td>A</td>
<td>20</td>
<td>No</td>
<td>CUSTOMER.InsurancePolicyNumber</td>
<td></td>
</tr>
<tr>
<td>LastName</td>
<td>A</td>
<td>15</td>
<td>No</td>
<td>CUSTOMER.LastName</td>
<td></td>
</tr>
<tr>
<td>LicenseNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER_LICENSENUMBER</td>
<td></td>
</tr>
<tr>
<td>LicensePlateNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER_LICENSEPLATENUMBER</td>
<td></td>
</tr>
<tr>
<td>MiddleInitial</td>
<td>A</td>
<td>1</td>
<td>Yes</td>
<td>CUSTOMER.MiddleInitial</td>
<td></td>
</tr>
<tr>
<td>Rank</td>
<td>A</td>
<td>3</td>
<td>Yes</td>
<td>CUSTOMER.Rank</td>
<td></td>
</tr>
<tr>
<td>RegistrationNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>CUSTOMER.RegistrationNumber</td>
<td></td>
</tr>
<tr>
<td>Sex</td>
<td>A</td>
<td>6</td>
<td>No</td>
<td>CUSTOMER.Sex</td>
<td></td>
</tr>
<tr>
<td>SMC#</td>
<td>A</td>
<td>4</td>
<td>Yes</td>
<td>CUSTOMER.SMC#</td>
<td></td>
</tr>
<tr>
<td>SocialSecurityNumber</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>CUSTOMER_SocialSecurityNumb er</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>TicketNumber</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>CUSTOMER.TicketNumber</td>
<td></td>
</tr>
<tr>
<td>WorkPhoneNumber</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>CUSTOMER.WorkPhoneNumber</td>
<td></td>
</tr>
</tbody>
</table>
## VEHICLE Table

**DBMS Type:** PARADOX for Windows/DOS 4.0+

**Source Object or Attribute:** VEHICLE Object

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value Type</th>
<th>Length</th>
<th>Null Allowed</th>
<th>Object Attribute Name</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>LicensePlateNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>VEHICLE.LicensePlateNumber</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>LicensePlateState</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>VEHICLE.LicensePlateState</td>
<td></td>
</tr>
<tr>
<td>SocialSecurityNumber_FK3</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>VehicleColor</td>
<td>A</td>
<td>7</td>
<td>No</td>
<td>VEHICLE.VehicleColor</td>
<td></td>
</tr>
<tr>
<td>VehicleIdentificationNumber</td>
<td>A</td>
<td>20</td>
<td>No</td>
<td>VEHICLE.VehicleIdentificationNumber</td>
<td></td>
</tr>
<tr>
<td>VehicleMake</td>
<td>A</td>
<td>5</td>
<td>No</td>
<td>VEHICLE.VehicleMake</td>
<td></td>
</tr>
<tr>
<td>VehicleType</td>
<td>A</td>
<td>11</td>
<td>No</td>
<td>VEHICLE.VehicleType</td>
<td></td>
</tr>
<tr>
<td>VehicleYear</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>VEHICLE.VehicleYear</td>
<td></td>
</tr>
</tbody>
</table>
LEVRAS DATABASE

Table Report
Album: LEVRAS.ALB

DRIVERL Table

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value Type</th>
<th>Length</th>
<th>Null Allowed</th>
<th>Object Attribute Name</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpirationDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>DRIVERLICENSE. ExpirationDate</td>
<td></td>
</tr>
<tr>
<td>LicenseNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>DRIVERLICENSE. LicenseNumber</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>LicenseState</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>DRIVERLICENSE. LicenseState</td>
<td></td>
</tr>
<tr>
<td>SocialSecurityNumber_FK8</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>Column Name</td>
<td>Value Type</td>
<td>Length</td>
<td>Null Allowed</td>
<td>Object Attribute Name</td>
<td>Index</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------</td>
<td>--------</td>
<td>--------------</td>
<td>----------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>ExpirationDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>REGISTRATION.ExpirationDate</td>
<td></td>
</tr>
<tr>
<td>RegistrationNumber</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>REGISTRATION.RegistrationNumber</td>
<td></td>
</tr>
<tr>
<td>RegistrationState</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>REGISTRATION.RegistrationState</td>
<td></td>
</tr>
<tr>
<td>SocialSecurityNumber_FK1</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>VE_LicensePlateNumber_FK2</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>Column Name</td>
<td>Value Type</td>
<td>Length</td>
<td>Null Allowed</td>
<td>Object Attribute Name</td>
<td>Index</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------------</td>
<td>--------</td>
<td>--------------</td>
<td>------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>ExpirationDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>INSURANCE.ExpirationDate</td>
<td></td>
</tr>
<tr>
<td>InsuranceCompanyName</td>
<td>A</td>
<td>15</td>
<td>No</td>
<td>INSURANCE.InsuranceCompany</td>
<td></td>
</tr>
<tr>
<td>InsurancePolicyNumber</td>
<td>A</td>
<td>20</td>
<td>No</td>
<td>INSURANCE.InsurancePolicy</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>SocialSecurityNumber_FK9</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>V_LicensePlateNumber_FK10</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>ForeignKey</td>
<td></td>
</tr>
</tbody>
</table>
**LEVRAS DATABASE**

**Table Report**

**Album: LEVRAS.ALB**

**DECAL Table**

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value Type</th>
<th>Length</th>
<th>Null Allowed</th>
<th>Object Attribute Name</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>DecalColor</td>
<td>A</td>
<td>5</td>
<td>No</td>
<td>DECAL.DecalColor</td>
<td></td>
</tr>
<tr>
<td>DecalExpirationMonth</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>DECAL.DecalExpirationMonth</td>
<td></td>
</tr>
<tr>
<td>DecalExpirationYear</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>DECAL.DecalExpirationYear</td>
<td></td>
</tr>
<tr>
<td>DecalIssueDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>DECAL.DecalIssueDate</td>
<td></td>
</tr>
<tr>
<td>DecalNumber</td>
<td>A</td>
<td>6</td>
<td>No</td>
<td>DECAL.DecalNumber</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>SocialSecurityNumber_FK4</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>VE_LicensePlateNumber_FK5</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>ForeignKey</td>
<td></td>
</tr>
</tbody>
</table>
**TICKET Table**

DBMS Type: PARADOX for Windows/DOS 4.0+
Source Object or Attribute: TICKET Object

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value Type</th>
<th>Length</th>
<th>Null Allowed</th>
<th>Object Attribute Name</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>BaseCourtDate</td>
<td>D</td>
<td>30</td>
<td>No</td>
<td>TICKET.BaseCourtDate</td>
<td></td>
</tr>
<tr>
<td>BaseDisposition</td>
<td>A</td>
<td></td>
<td>Yes</td>
<td>TICKET.BaseDisposition</td>
<td></td>
</tr>
<tr>
<td>BaseDispositionDate</td>
<td>D</td>
<td></td>
<td>Yes</td>
<td>TICKET.BaseDispositionDate</td>
<td></td>
</tr>
<tr>
<td>BaseJudgeName</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>TICKET.BaseJudgeName</td>
<td></td>
</tr>
<tr>
<td>Points</td>
<td>A</td>
<td>2</td>
<td>Yes</td>
<td>TICKET.Points</td>
<td></td>
</tr>
<tr>
<td>PolicemanName</td>
<td>A</td>
<td>10</td>
<td>No</td>
<td>TICKET.PolicemanName</td>
<td></td>
</tr>
<tr>
<td>SocialSecurityNumber_FK6</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>ForeignKey</td>
<td>PrimaryKey</td>
</tr>
<tr>
<td>TicketNumber</td>
<td>A</td>
<td>9</td>
<td>No</td>
<td>TICKET.TicketNumber</td>
<td></td>
</tr>
<tr>
<td>TransactionDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>TICKET.TransactionDate</td>
<td></td>
</tr>
<tr>
<td>VE_LicensePlateNumber_FK7</td>
<td>A</td>
<td>10</td>
<td>Yes</td>
<td>ForeignKey</td>
<td></td>
</tr>
<tr>
<td>ViolationCode</td>
<td>A</td>
<td>2</td>
<td>No</td>
<td>TICKET.ViolationCode</td>
<td></td>
</tr>
<tr>
<td>ViolationDate</td>
<td>D</td>
<td></td>
<td>No</td>
<td>TICKET.ViolationDate</td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX F. APPLICATION MENUS

This appendix contains copies of the LEVRAS Application Menus. Included are the WELCOME TO LEVRAS introduction screen, and the MAIN MENU. The welcome screen is shown upon entering the LEVRAS program, and provides title slide type information. The Main Menu then allows LEVRAS users to proceed to the desired application in order to conduct required business.
APPENDIX G. APPLICATION INPUT FORMS

This appendix contains examples of all of the LEVRAS Data Input Forms which can be used in the system. In addition, a copy of the Ticket Administrator screen is also included. The construction of these forms is discussed in chapter five of this thesis, and instructions on their use are contained in the LEVRAS User's Manual (see Appendix J of this thesis).
<table>
<thead>
<tr>
<th>Customer Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SBN</strong></td>
</tr>
<tr>
<td><strong>NAME</strong></td>
</tr>
<tr>
<td><strong>STATE</strong></td>
</tr>
<tr>
<td><strong>ZIP</strong></td>
</tr>
<tr>
<td><strong>GENDER</strong></td>
</tr>
<tr>
<td><strong>EMAIL</strong></td>
</tr>
<tr>
<td><strong>PHONE</strong></td>
</tr>
<tr>
<td><strong>ADDRESS</strong></td>
</tr>
<tr>
<td><strong>CITY</strong></td>
</tr>
<tr>
<td><strong>COUNTY</strong></td>
</tr>
<tr>
<td><strong>PHONE</strong></td>
</tr>
<tr>
<td><strong>DATE</strong></td>
</tr>
<tr>
<td><strong>ADDRESS</strong></td>
</tr>
<tr>
<td><strong>CITY</strong></td>
</tr>
<tr>
<td><strong>ZIP</strong></td>
</tr>
<tr>
<td><strong>STATE</strong></td>
</tr>
<tr>
<td><strong>COUNTRY</strong></td>
</tr>
</tbody>
</table>
## Driver License Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>License Num</td>
<td>026-44-0020</td>
</tr>
<tr>
<td>State</td>
<td>CA</td>
</tr>
<tr>
<td>Agency</td>
<td>EZGOING</td>
</tr>
<tr>
<td>Expiration Date</td>
<td>5/13/95</td>
</tr>
</tbody>
</table>
Registration Information

- Telephone: 011-46-3463
- Name: NH
- Address: REG4
- Identi#1c: CHEF1
- Entry Date: 4/12/96

ADD
MODIFY
DELETE
FWD
BACK
HOME
HELP
<table>
<thead>
<tr>
<th>Security Number</th>
<th>026-44-0020</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insurance Company Name</td>
<td>USAA</td>
</tr>
<tr>
<td>Policy Number</td>
<td>23456789</td>
</tr>
<tr>
<td>Agent Name</td>
<td>WTLFT</td>
</tr>
<tr>
<td>Effective Date</td>
<td>4/14/96</td>
</tr>
<tr>
<td>Decal Information</td>
<td></td>
</tr>
<tr>
<td>---------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>SSN</strong></td>
<td>011-46-3483</td>
</tr>
<tr>
<td><strong>Center Plate Number</strong></td>
<td>CHEF1</td>
</tr>
<tr>
<td><strong>Decal Color</strong></td>
<td>BLUE</td>
</tr>
<tr>
<td><strong>Decal Number</strong></td>
<td>AA2683</td>
</tr>
<tr>
<td><strong>Decal Issue Date</strong></td>
<td>5/7/85</td>
</tr>
<tr>
<td><strong>Decal Expiration Month</strong></td>
<td>09</td>
</tr>
<tr>
<td><strong>Decal Expiration Year</strong></td>
<td>35</td>
</tr>
<tr>
<td>SSN</td>
<td>026-44-0020</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Ticket Number</td>
<td>10085380</td>
</tr>
<tr>
<td>Policeman Name</td>
<td>JONES</td>
</tr>
<tr>
<td>Violation Code</td>
<td>96</td>
</tr>
<tr>
<td>Violation Date</td>
<td>12/12/95</td>
</tr>
<tr>
<td>Case Court Date</td>
<td>1/19/95</td>
</tr>
<tr>
<td>Case Disposition</td>
<td>WARNING</td>
</tr>
<tr>
<td>Case Disposition Date</td>
<td>1/19/95</td>
</tr>
<tr>
<td>Case Judge Name</td>
<td>MARILYN</td>
</tr>
<tr>
<td>Points</td>
<td>02</td>
</tr>
<tr>
<td>License Plate Number</td>
<td>RUNNER</td>
</tr>
<tr>
<td>DECAL EXP MONTH</td>
<td>DECAL EXP YEAR</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>09</td>
<td>95</td>
</tr>
<tr>
<td>09</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX H. APPLICATION OUTPUT REPORTS

This appendix contains examples of the following LEVRAS application output reports: Customer and Vehicle Status Report, Decal Status Report, Ticket Status Report, and a Customized Report (designed per NPS Security Officer direction). These reports are printed out routinely to provide supervisory information concerning LEVRAS operations. In addition, the Customized Report can be reformatted anytime to maximize the quality of information provided to the NPS Security Officer. The construction of these reports is discussed in chapter five of this thesis. Instructions on the use of these reports is contained in the LEVRAS User's Manual (see Appendix J).
<table>
<thead>
<tr>
<th>State</th>
<th>Type</th>
<th>Year</th>
<th>Model</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>RUNNER</td>
<td>90</td>
<td>VW</td>
<td>GOLF</td>
</tr>
<tr>
<td>CA</td>
<td>WTLFT</td>
<td>86</td>
<td>VW</td>
<td>QUANTUM</td>
</tr>
<tr>
<td>Decal #</td>
<td>Decal Exp Month</td>
<td>Decal Exp Year</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>----------------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AA2770</td>
<td>09</td>
<td>95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AB2770</td>
<td>09</td>
<td>95</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
NAVAL POSTGRADUATE SCHOOL CUSTOMER TICKET VIOLATION REPORT

SSN: 026-44-0020
Last Name: MCGIBBON
First Name: HENRY
Middle Initial: M
Rank: O3
Employee Type: 
Duty Station: NPS
SMC#: 
Work Phone: 
Curric/Staff: 
Home Phone: 
Database Entry Date: 

<table>
<thead>
<tr>
<th>License</th>
<th>Location</th>
<th>Violency</th>
<th>Vehicle</th>
<th>Ticket Type</th>
<th>Date Accepted</th>
<th>Date Discharged</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>RUNNER</td>
<td>90</td>
<td>VW</td>
<td>GOLF</td>
<td>1/19/95</td>
<td>SILVER</td>
</tr>
<tr>
<td>CA</td>
<td>WTLFT</td>
<td>86</td>
<td>VW</td>
<td>QUANTUM</td>
<td>1/19/95</td>
<td>GOLD</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ticket ID</th>
<th>Date</th>
<th>Time</th>
<th>Location</th>
<th>Reason</th>
<th>Date</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>10085380</td>
<td>96</td>
<td>12/12/95</td>
<td>1/19/95</td>
<td>WARNING</td>
<td>1/19/95</td>
<td>02</td>
</tr>
<tr>
<td>10085383</td>
<td>23</td>
<td>5/7/95</td>
<td>5/7/95</td>
<td>DETAINED</td>
<td>5/7/95</td>
<td>10</td>
</tr>
<tr>
<td>100999999</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

120
MEMORANDUM

From: Greg Caughran, Security Officer, Code 44
To: MCGIBBON, HENRY, M, 026-44-0020, NPS, 36

Subj: TEMPORARY SUSPENSION OF DRIVING PRIVILEGES

Ref: (a) NAVPGSCOLINST 5560.5
     (b) OPNAVINST 11200.5C

Encl: (1) Letter of Acknowledgement of Temporary Suspension of Driving Privileges

1. In accordance with references (a) and (b), you are hereby notified that your driving privileges on the Naval Postgraduate School and temporarily suspended, (and two points have been assessed to your driving privileges, effective immediately. The suspension is issued because you failed to acknowledge the following citation number(s) on the following dates within the three working days as specified on the citation(s):

<table>
<thead>
<tr>
<th>Citation Number</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>10085380</td>
<td>12/12/95</td>
</tr>
<tr>
<td>10085383</td>
<td>5/7/95</td>
</tr>
</tbody>
</table>

During this suspension period, you are not to drive any privately owned motor vehicles on NPS property, including motorcycles and mopeds.

2. This suspension is mandatory. An exception may be granted only by a request for court appearance with the traffic hearing officer.

3. You are to report to the Security Department and submit enclosure (1) no later than FIVE DAYS after receipt of this later. At this time you driving privileges will be suspended for 10 WORKING DAYS from the date you report to the Security Department. Point of contact is Ms. Marilyn Owens (Traffic Administrator) at extension 2556.

Greg C. Caughran

Copy To:
07

121
APPENDIX I. PROGRAM SCRIPTS

This appendix contains the unique LEVRAS Program Scripts. These program scripts are copies of the actual lines of source code, using the Paradox ObjectPal programming language. The functions of the source code include the pushbutton execution of commands presented on the LEVRAS forms and menus. The source code also displays the linkage between the various LEVRAS data input forms and menus. Chapter five of this thesis contains information on program script generation. The LEVRAS User's Manual, Chapter V., contained in Appendix J, refers maintainers to Paradox publications if the need to modify the source code arises.
method run(var eventInfo Event)

;declares userInput String and Form to be used to provide
;credit to LEVRAS designers and developers in a UserInput
;information box, and then takes the user to the first form
;(welcome) of the LEVRAS program (after pushing the script
;pushbutton on the Paradox speed bar).

var
    userInput String
    formVar Form
endVar

userInput = "LAW ENFORCEMENT VEHICLE
REGISTRATION ADMINISTRATION
SYSTEM (LEVRAS)
designed & developed by:

CDR Mark S. Nault, USN
&
LT Henry M. McGibbon, USN"
userInput.View("About LEVRAS")
formVar.open("welcome")
close()
endmethod

method pushButton(var eventInfo Event)

;declares the Main Menu form

var
    mainForm Form; declaring MainMenu form.
endVar

;opens the Main Menu form and closes the Welcome
;form

mainForm.open("mainmenu"); opens Main Menu.
mainForm.show()
close()
endmethod
;COMMENT: ADVANCES THE MAIN MENU FORM TO
; CUSTOMER INFORMATION FORM

method pushButton(var eventInfo Event)

; declares Customer table and form

var
custTbl Table
custForm Form
endVar

; opens Customer Information form and closes Main Menu
; form
custTbl.attach("Customer.DB")

; ensures that only one user writes to a table in a
; multiuser mode
if not lock(custTbl, "Write", "customer.db", "Write") then
  endif
  custForm.open("Customer")
custForm.show()
close()
endmethod

;COMMENT: DEPARTS LEVRAS PROGRAM ALTOGETHER
; FROM THE MAIN MENU

method pushButton(var eventInfo Event)

; closes the Main Menu and leaves the
; LEVRAS program

close()
endmethod
COMMENT: INSTRUCTIONS ON HOW TO PERFORM QUERIES ALONG WITH
AN EXAMPLE OF AN EXECUTED QUERY (CUSTOMER.QBE)
AND ITS RESULTS (PRIVATE.DB).

method pushButton(var eventInfo Event)

; declares contextHelp String for info
; boxes and tv TableView for query

var

    contextHelp String
    tv TableView
endVar

beep()

; quote used name used in info box

contextHelp = "Queries can be performed
by selecting a pre-made *.QBE file.
Select File->Open->Query. Choose
pre-made query fn Select File box.
Push Lightning Bolt on Speedbar
to execute query. Further Questions?
Refer to LEVRAZ User's Manual."
contextHelp.view("LEVRAZ QUERIES")

beep()

; quote used and name of info box

contextHelp = "An example of a
successful Customer.QBE query
is to follow. Ensure to select
Close in the top left
corner of the Priv:answer.DB
window after viewing."
contextHelp.view("LEVRAZ QUERY EXECUTION EX.")

; executes customer.qbe query and stores answer
; in a table called :priv:answer.db. If the
; query has execution errors, a magstop box
; will appear stating that a query execution
; error has occurred.

beep()
if executeQBEFile("customer.qbe", ":priv:answer.db") then
    tv.open("*:priv:answer.db")
else msgStop("QUERY EXECUTION ERROR", "The query
eexample called CUSTOMER.QBE was
not executed! Select
CUSTOMER.QBE and try again!")
endIf
endmethod
;COMMENT: ADVANCES THE MAIN MENU FORM TO
; PROCесс TICKET FORM

method pushButton(var eventInfo Event)

;declares Process Ticket form

var
    procForm Form
    tcOne, tcTwo, tcThree, tcFour TCursor
endVar

;Write lock is enforced if another user tries to access
;while being used

tcOne.open("Customer")
tcTwo.open("Decal")
tcThree.open("Vehicle")
tcFour.open("Ticket")
lock(tcOne, "Write", tcTwo, "Write",
    tcThree, "Write", tcFour, "Write")

;opens Process Ticket form and closes the Main Menu
;form

procForm.open("processt")
procForm.show()

close()

endmethod
method pushButton(var eventInfo Event)

; declares Report form

var
  repoForm Form
endVar

; opens Report form and then closes
; the Main Menu form

repoForm.open("reports")
repoForm.show()
close()

endmethod
;COMMENT: INSTRUCTIONS HOW TO ARCHIVE DATA

method pushButton(var eventInfo Event)

; declares contextHelp string to be used in info box

var
contextHelp String
endVar

; opens up an info box and displays quoted info in box and names the box LEVRAS ARCHIVE

beep()
contextHelp.view("LEVRAS ARCHIVE")

endmethod
;COMMENT: ADDS NEW RECORD TO CUSTOMER TABLE

method pushButton(var eventInfo Event)

;declare Customer table

var
custTbl Table
endVar

;inserts a record into Customer.DB table

   beep()
   action(DataBeginEdit)
   action(DataInsertRecord)
   custTbl.attach("Customer.DB")
   action(DataPostRecord)

endmethod

;COMMENT: MODIFIES FIELD(S) IN A RECORD OF CUSTOMER TABLE

method pushButton(var eventInfo Event)

;declares the Customer table.

   var
custTbl Table
endVar

;edits a record in the Customer.DB table

   beep()
   action(DataBeginEdit)
   custTbl.attach("Customer.DB")
   action(DataPostRecord)
   action(DataEndEdit)

endmethod
method pushButton(var eventInfo Event)

; declare string to be used in a dialog box

var
    response String
endVar

; dialog box ensures user truly desires to delete a record

response = msgQuestion("Delete Record?", "Delete this selected record?")
if (response = "Yes") then
    action(DataDeleteRecord)
    if (response = "no") then
        action(DataPostRecord)
    endif
endif
beep()
endmethod

method pushButton(var eventInfo Event)

; declares the contextHelp String to create a help box.

var
    contextHelp String
endVar

beep()
contextHelp = "Refer to LEVRAS User's Manual for help and/or push F1."; info to be displayed in the LEVRAS HELP Box.
contextHelp.view("LEVRAS HELP"); title of the HELP box.
endmethod
;COMMENT: RETURNS CUSTOMER FORM TO MAIN FORM
method pushButton(var eventInfo Event)
;
defines the Main Menu form

var
    mainForm Form; declaring MainMenu form.
endVar

;opens the Main Menu form and then closes the
;Customer Information form

mainForm.open("mainmenu"); opens Main Menu.
mainForm.show(); shows Main Menu on CRT.
close(); closes the Customer Information screen.
endmethod

;COMMENT: ADVANCES CUSTOMER INFORMATION FORM TO
; DRIVER LICENSE INFORMATION FORM

method pushButton(var eventInfo Event)
;
defines driver license form and its table

var
    drv1Form Form;
    drv1Tbl Table
endVar

;fowards Customer Information screen to Driver License
;screen by opening Driver License table and form, and
;then closes the Customer Information Screen.
driv1Tbl.attach("Driverli.DC")

if not lock(drv1Tbl, "Write", "Driverli.DC", "Write") then
    endif; for multiuser use...only one user can use table

drv1Form.open("Dvrlisen");
drv1Form.show()
close()
endmethod
method arrive(var eventInfo MoveEvent)

; lists rank selections to be selected in a scroll box (in sequential order)

TheList.list.selection = 1
TheList.list.value = "E1"
TheList.list.selection = 2
TheList.list.value = "E2"
TheList.list.selection = 3
TheList.list.value = "E3"
TheList.list.selection = 4
TheList.list.value = "E4"
TheList.list.selection = 5
TheList.list.value = "E5"
TheList.list.selection = 6
TheList.list.value = "E6"
TheList.list.selection = 7
TheList.list.value = "E7"
TheList.list.selection = 8
TheList.list.value = "E8"
TheList.list.selection = 9
TheList.list.value = "E9"
TheList.list.selection = 10
TheList.list.value = "W1"
TheList.list.selection = 11
TheList.list.value = "W2"
TheList.list.selection = 12
TheList.list.value = "W3"
TheList.list.selection = 13
TheList.list.value = "W4"
TheList.list.selection = 14
TheList.list.value = "W5"
TheList.list.selection = 15
TheList.list.value = "O1"
TheList.list.selection = 16
TheList.list.value = "O2"
TheList.list.selection = 17
TheList.list.value = "O3"
TheList.list.selection = 18
TheList.list.value = "O4"
TheList.list.selection = 19
TheList.list.value = "O5"
TheList.list.selection = 20
TheList.list.value = "O6"
TheList.list.selection = 21
TheList.list.value = "O7"
TheList.list.selection = 22
TheList.list.value = "O8"
TheList.list.selection = 23
TheList.list.value = "O9"
TheList.list.selection = 24
TheList.list.value = "O10"
endmethod
method arrive(var eventInfo MoveEvent)

; lists choices for selection by a LEVRA
; user (list in sequential order as shown
; below)

TheList.list.selection = 1
TheList.list.value = "CO"
TheList.list.selection = 2
TheList.list.value = "ACDU"
TheList.list.selection = 3
TheList.list.value = "MILRES"
TheList.list.selection = 4
TheList.list.value = "REMTIL"
TheList.list.selection = 5
TheList.list.value = "CIVGOV"
TheList.list.selection = 6
TheList.list.value = "COMMERCIAL"
TheList.list.selection = 7
TheList.list.value = "MILDEP"
TheList.list.selection = 8
TheList.list.value = "DISVET"
endmethod

method arrive(var eventInfo MoveEvent)

; lists two choices: either male or female

TheList.list.selection = 1
TheList.list.value = "MALE"
TheList.list.selection = 2
TheList.list.value = "FEMALE"
endmethod
method arrive(var eventInfo MoveEvent)

;lists grade selections to be selected in a scroll box (and in sequential order, as shown below)

TheList.list.selection = 1
TheList.list.value = "A81"
TheList.list.selection = 2
TheList.list.value = "A82"
TheList.list.selection = 3
TheList.list.value = "A83"
TheList.list.selection = 4
TheList.list.value = "A84"
TheList.list.selection = 5
TheList.list.value = "A85"
TheList.list.selection = 6
TheList.list.value = "A86"
TheList.list.selection = 7
TheList.list.value = "A87"
TheList.list.selection = 8
TheList.list.value = "A88"
TheList.list.selection = 9
TheList.list.value = "A89"
TheList.list.selection = 10
TheList.list.value = "A90"
TheList.list.selection = 11
TheList.list.value = "A91"
TheList.list.selection = 12
TheList.list.value = "A92"
TheList.list.selection = 13
TheList.list.value = "A93"
TheList.list.selection = 14
TheList.list.value = "A94"
TheList.list.selection = 15
TheList.list.value = "A95"
TheList.list.selection = 16
TheList.list.value = "A96"
TheList.list.selection = 17
TheList.list.value = "A97"
TheList.list.selection = 18
TheList.list.value = "A98"
TheList.list.selection = 19
TheList.list.value = "A99"
TheList.list.selection = 20
TheList.list.value = "A100"
TheList.list.selection = 21
TheList.list.value = "A101"
TheList.list.selection = 22
TheList.list.value = "A102"
TheList.list.selection = 23
TheList.list.value = "GS13"
TheList.list.selection = 24
TheList.list.value = "GS14"
TheList.list.selection = 25
TheList.list.value = "GS15"
TheList.list.selection = 26
TheList.list.value = "GS16"
TheList.list.selection = 27
TheList.list.value = "GS17"
TheList.list.selection = 28
TheList.list.value = "GS18"
TheList.list.selection = 29
TheList.list.value = "NA1"
TheList.list.selection = 30
TheList.list.value = "NA2"
TheList.list.selection = 31
TheList.list.value = "NA3"
TheList.list.selection = 32
TheList.list.value = "NA4"
TheList.list.selection = 33
TheList.list.value = "NA5"
TheList.list.selection = 34
TheList.list.value = "NA6"
TheList.list.selection = 35
TheList.list.value = "NA7"
TheList.list.selection = 36
TheList.list.value = "NA8"
TheList.list.selection = 37
TheList.list.value = "NA9"
TheList.list.selection = 38
TheList.list.value = "NA10"
TheList.list.selection = 39
TheList.list.value = "NA11"
TheList.list.selection = 40
TheList.list.value = "NA12"
TheList.list.selection = 41
TheList.list.value = "NA13"
TheList.list.selection = 42
TheList.list.value = "NA14"
TheList.list.selection = 43
TheList.list.value = "NA15"
TheList.list.selection = 44
TheList.list.value = "NL1"
TheList.list.selection = 45
TheList.list.value = "NL2"
TheList.list.selection = 46
TheList.list.value = "NL3"
TheList.list.selection = 47
TheList.list.value = "NL4"
TheList.list.selection = 48
TheList.list.value = "NL5"
TheList.list.selection = 49
TheList.list.value = "NL6"
TheList.list.value = "NL7"
TheList.list.value = "NL8"
TheList.list.value = "NL9"
TheList.list.selection = 50
TheList.list.value = "NL10"
TheList.list.selection = 51
TheList.list.value = "NL11"
TheList.list.selection = 52
TheList.list.value = "NL12"
TheList.list.selection = 53
TheList.list.value = "NL13"
TheList.list.selection = 54
TheList.list.value = "NL14"
TheList.list.selection = 55
TheList.list.value = "NL15"
TheList.list.selection = 56
TheList.list.value = "PS1"
TheList.list.selection = 57
TheList.list.value = "PS2"
TheList.list.selection = 58
TheList.list.value = "PS3"
TheList.list.selection = 59
TheList.list.value = "PS4"
TheList.list.selection = 60
TheList.list.value = "PS5"
TheList.list.selection = 61
TheList.list.value = "PS6"
TheList.list.selection = 62
TheList.list.value = "PS7"
TheList.list.selection = 63
TheList.list.value = "WD1"
TheList.list.selection = 64
TheList.list.value = "WD2"
TheList.list.selection = 65
TheList.list.value = "WD3"
TheList.list.selection = 66
TheList.list.value = "WD4"
TheList.list.selection = 67
TheList.list.value = "WD5"
TheList.list.selection = 68
TheList.list.value = "WD6"
TheList.list.selection = 69
TheList.list.value = "WD7"
TheList.list.selection = 70
TheList.list.value = "WD8"
TheList.list.selection = 71
TheList.list.value = "WD9"
TheList.list.selection = 72
TheList.list.value = "WD10"
TheList.list.selection = 76
TheList.list.value = "WD11"
TheList.list.selection = 77
TheList.list.value = "WD12"
TheList.list.selection = 78
TheList.list.value = "WD13"
TheList.list.selection = 79
TheList.list.value = "WD14"
TheList.list.selection = 80
TheList.list.value = "WD15"
TheList.list.selection = 81
TheList.list.value = "WD16"
TheList.list.selection = 82
TheList.list.value = "WD17"
TheList.list.selection = 83
TheList.list.value = "WD18"
TheList.list.selection = 84
TheList.list.value = "WD19"
TheList.list.selection = 85
TheList.list.value = "WG1"
TheList.list.selection = 86
TheList.list.value = "WG2"
TheList.list.selection = 87
TheList.list.value = "WG3"
TheList.list.selection = 88
TheList.list.value = "WG4"
TheList.list.selection = 89
TheList.list.value = "WG5"
TheList.list.selection = 90
TheList.list.value = "WG6"
TheList.list.selection = 91
TheList.list.value = "WG7"
TheList.list.selection = 92
TheList.list.value = "WG8"
TheList.list.selection = 93
TheList.list.value = "WG9"
TheList.list.selection = 94
TheList.list.value = "WG10"
TheList.list.selection = 95
TheList.list.value = "WG11"
TheList.list.selection = 96
TheList.list.value = "WG12"
TheList.list.selection = 97
TheList.list.value = "WG13"
TheList.list.selection = 98
TheList.list.value = "WG14"
TheList.list.selection = 99
TheList.list.value = "WG15"
TheList.list.selection = 100
TheList.list.value = "WL1"
TheList.list.selection = 101
TheList.list.value = "WL2"
TheList.list.selection = 102
TheList.list.value = "WL3"
TheList.list.selection = 103
TheList.list.value = "WL4"
TheList.list.selection = 104
TheList.list.value = "WL5"
TheList.list.selection = 105
TheList.list.value = "WL6"
TheList.list.selection = 106
TheList.list.value = "WL7"
TheList.list.selection = 107
TheList.list.value = "WL8"
TheList.list.selection = 108
TheList.list.value = "WL9"
TheList.list.selection = 109
TheList.list.value = "WL10"
TheList.list.selection = 110
TheList.list.value = "WL11"
TheList.list.selection = 111
TheList.list.value = "WL12"
TheList.list.selection = 112
TheList.list.value = "WL13"
TheList.list.selection = 113
TheList.list.value = "WL14"
TheList.list.selection = 114
TheList.list.value = "WL15"
TheList.list.selection = 115
TheList.list.value = "WP1"
TheList.list.selection = 116
TheList.list.value = "WP2"
TheList.list.selection = 117
TheList.list.value = "WP3"
TheList.list.selection = 118
TheList.list.value = "WP4"
TheList.list.selection = 119
TheList.list.value = "WP5"
TheList.list.selection = 120
TheList.list.value = "WP6"
TheList.list.selection = 121
TheList.list.value = "WP7"
TheList.list.selection = 122
TheList.list.value = "WP8"
TheList.list.selection = 123
TheList.list.value = "WP9"
TheList.list.selection = 124
TheList.list.value = "WP10"
TheList.list.selection = 125
TheList.list.value = "WP11"
TheList.list.selection = 126
TheList.list.value = "WP12"
TheList.list.selection = 127
TheList.list.value = "WP13"
TheList.list.selection = 128
TheList.list.value = "WP14"
TheList.list.selection = 129
TheList.list.value = "WP15"
TheList.list.selection = 130
TheList.list.value = "WP16"
TheList.list.selection = 131
TheList.list.value = "WP17"
TheList.list.selection = 132
TheList.list.value = "WP18"
TheList.list.selection = 133
TheList.list.value = "WP19"
TheList.list.selection = 134
TheList.list.value = "WP20"
TheList.list.selection = 135
TheList.list.value = "WP21"
TheList.list.selection = 136
TheList.list.value = "WP22"
TheList.list.selection = 137
TheList.list.value = "WP23"
TheList.list.selection = 138
TheList.list.value = "WP24"
TheList.list.selection = 139
TheList.list.value = "WP25"
TheList.list.selection = 140
TheList.list.value = "WP26"
TheList.list.selection = 141
TheList.list.value = "WP27"
TheList.list.selection = 142
TheList.list.value = "WP28"
TheList.list.selection = 143
TheList.list.value = "WP29"
TheList.list.selection = 144
TheList.list.value = "WP30"
TheList.list.selection = 145
TheList.list.value = "WP31"
TheList.list.selection = 146
TheList.list.value = "WP32"
TheList.list.selection = 147
TheList.list.value = "WP33"
TheList.list.selection = 148
TheList.list.value = "WP34"
TheList.list.selection = 149
TheList.list.value = "WP35"
TheList.list.selection = 150
TheList.list.value = "WP36"
TheList.list.selection = 151
TheList.list.value = "WP37"
TheList.list.selection = 152
TheList.list.value = "WP38"
TheList.list.selection = 153
TheList.list.value = "WP39"
TheList.list.selection = 154
TheList.list.value = "WP40"
TheList.list.selection = 155
TheList.list.value = "WP41"
TheList.list.value = "WP42"
TheList.list.value = "WP43"
TheList.list.value = "WP44"
TheList.list.value = "WP45"
TheList.list.value = "WP46"
TheList.list.value = "WP47"
TheList.list.value = "WP48"
TheList.list.value = "WP49"
TheList.list.value = "WP50"
TheList.list.value = "WP51"
TheList.list.value = "WP52"
TheList.list.value = "WP53"
TheList.list.value = "WP54"
TheList.list.value = "WP55"
TheList.list.value = "WP56"
TheList.list.value = "WP57"
TheList.list.value = "WP58"
TheList.list.value = "WP59"
TheList.list.value = "WP60"
TheList.list.value = "WP61"
TheList.list.value = "WP62"
TheList.list.value = "WP63"
TheList.list.value = "WP64"
TheList.list.value = "WP65"
TheList.list.value = "WP66"
TheList.list.value = "WP67"
TheList.list.selection = 182
TheList.list.value = "WP68"
TheList.list.selection = 183
TheList.list.value = "WP69"
TheList.list.selection = 184
TheList.list.value = "WP70"
TheList.list.selection = 185
TheList.list.value = "WP71"
TheList.list.selection = 186
TheList.list.value = "WP72"
TheList.list.selection = 187
TheList.list.value = "WP73"
TheList.list.selection = 188
TheList.list.value = "WP74"
TheList.list.selection = 189
TheList.list.value = "WP75"
TheList.list.selection = 190
TheList.list.value = "WP76"
TheList.list.selection = 191
TheList.list.value = "WP77"
TheList.list.selection = 192
TheList.list.value = "WP78"
TheList.list.selection = 193
TheList.list.value = "WS1"
TheList.list.selection = 194
TheList.list.value = "WS2"
TheList.list.selection = 195
TheList.list.value = "WS3"
TheList.list.selection = 196
TheList.list.value = "WS4"
TheList.list.selection = 197
TheList.list.value = "WS5"
TheList.list.selection = 198
TheList.list.value = "WS6"
TheList.list.selection = 199
TheList.list.value = "WS7"
TheList.list.selection = 200
TheList.list.value = "WS8"
TheList.list.selection = 201
TheList.list.value = "WS9"
TheList.list.selection = 202
TheList.list.value = "WS10"
TheList.list.selection = 203
TheList.list.value = "WS11"
TheList.list.selection = 204
TheList.list.value = "WS12"
TheList.list.selection = 205
TheList.list.value = "WS13"
TheList.list.selection = 206
TheList.list.value = "WS14"
TheList.list.selection = 207
TheList.list.value = "WS15"
TheList.list.selection = 208
TheList.list.value = "WS16"
TheList.list.value = "WS17"
TheList.list.value = "WS18"
TheList.list.value = "WS19"
TheList.list.value = "WS21"
TheList.list.value = "WS22"
TheList.list.value = "WS23"
TheList.list.value = "WS24"
TheList.list.value = "WS25"
TheList.list.value = "WS26"
TheList.list.value = "WS27"
TheList.list.value = "WS28"
TheList.list.value = "WS29"
TheList.list.value = "WS31"
TheList.list.value = "WS32"
TheList.list.value = "WS33"
TheList.list.value = "WS34"
TheList.list.value = "WS35"
TheList.list.value = "WS36"
TheList.list.value = "WS37"
TheList.list.value = "WS38"
TheList.list.value = "WS39"
TheList.list.value = "WS41"
TheList.list.value = "WS42"
TheList.list.value = "WS43"
TheList.list.value = "WS44"
TheList.list.value = "WS45"
TheList.list.value = "WS46"
TheList.list.value = "WS47"
TheList.list.value = "WS48"
TheList.list.value = "WS49"
TheList.list.value = "WS51"
TheList.list.value = "WS52"
TheList.list.value = "WS53"
TheList.list.value = "WS54"
TheList.list.value = "WS55"
TheList.list.value = "WS56"
TheList.list.value = "WS57"
TheList.list.value = "WS58"
TheList.list.value = "WS59"
TheList.list.value = "WS61"
TheList.list.value = "WS62"
TheList.list.value = "WS63"
TheList.list.value = "WS64"
TheList.list.value = "WS65"
TheList.list.value = "WS66"
TheList.list.value = "WS67"
TheList.list.value = "WS68"
TheList.list.value = "WS69"
TheList.list.value = "WS71"
TheList.list.value = "WS72"
TheList.list.value = "WS73"
TheList.list.value = "WS74"
TheList.list.value = "WS75"
TheList.list.value = "WS76"
TheList.list.value = "WS77"
TheList.list.value = "WS78"
TheList.list.value = "WS79"
TheList.list.value = "WS81"
TheList.list.value = "WS82"
TheList.list.value = "WS83"
TheList.list.value = "WS84"
TheList.list.value = "WS85"
TheList.list.value = "WS86"
TheList.list.value = "WS87"
TheList.list.value = "WS88"
TheList.list.value = "WS89"
TheList.list.value = "WS91"
TheList.list.value = "WS92"
TheList.list.value = "WS93"
TheList.list.value = "WS94"
TheList.list.value = "WS95"
TheList.list.value = "WS96"
TheList.list.value = "WS97"
TheList.list.value = "WS98"
TheList.list.value = "WS99"
TheList.list.value = "WS101"
TheList.list.value = "WS102"
TheList.list.value = "WS103"
TheList.list.value = "WS104"
TheList.list.value = "WS105"
TheList.list.value = "WS106"
TheList.list.value = "WS107"
TheList.list.value = "WS108"
TheList.list.value = "WS109"
TheList.list.value = "WS111"
TheList.list.value = "WS112"
TheList.list.value = "WS113"
TheList.list.value = "WS114"
TheList.list.value = "WS115"
TheList.list.value = "WS116"
TheList.list.value = "WS117"
TheList.list.value = "WS118"
TheList.list.value = "WS119"
TheList.list.value = "WS121"
TheList.list.value = "WS122"
TheList.list.value = "WS123"
TheList.list.value = "WS124"
TheList.list.value = "WS125"
TheList.list.value = "WS126"
TheList.list.value = "WS127"
TheList.list.value = "WS128"
TheList.list.value = "WS129"
TheList.list.value = "WS131"
TheList.list.value = "WS132"
TheList.list.value = "WS133"
TheList.list.value = "WS134"
TheList.list.value = "WS135"
TheList.list.value = "WS136"
TheList.list.value = "WS137"
TheList.list.value = "WS138"
TheList.list.value = "WS139"
TheList.list.value = "WS141"
TheList.list.value = "WS142"
TheList.list.value = "WS143"
TheList.list.value = "WS144"
TheList.list.value = "WS145"
TheList.list.value = "WS146"
TheList.list.value = "WS147"
TheList.list.value = "WS148"
TheList.list.value = "WS149"
TheList.list.value = "WS151"
TheList.list.value = "WS152"
TheList.list.value = "WS153"
TheList.list.value = "WS154"
TheList.list.value = "WS155"
TheList.list.value = "WS156"
TheList.list.value = "WS157"
TheList.list.value = "WS158"
TheList.list.value = "WS159"
TheList.list.value = "WS161"
TheList.list.value = "WS162"
TheList.list.value = "WS163"
TheList.list.value = "WS164"
TheList.list.value = "WS165"
TheList.list.value = "WS166"
TheList.list.value = "WS167"
TheList.list.value = "WS168"
TheList.list.value = "WS169"
TheList.list.value = "WS171"
TheList.list.value = "WS172"
TheList.list.value = "WS173"
TheList.list.value = "WS174"
TheList.list.value = "WS175"
TheList.list.value = "WS176"
TheList.list.value = "WS177"
TheList.list.value = "WS178"
TheList.list.value = "WS179"
TheList.list.value = "WS181"
TheList.list.value = "WS182"
TheList.list.value = "WS183"
TheList.list.value = "WS184"
TheList.list.value = "WS185"
TheList.list.value = "WS186"
TheList.list.value = "WS187"
TheList.list.value = "WS188"
TheList.list.value = "WS189"
TheList.list.value = "WS191"
TheList.list.value = "WS192"
TheList.list.value = "WS193"
TheList.list.value = "WS194"
TheList.list.value = "WS195"
TheList.list.value = "WS196"
TheList.list.value = "WS197"
TheList.list.value = "WS198"
TheList.list.value = "WS199"
TheList.list.value = "WS201"
TheList.list.value = "WS202"
TheList.list.value = "WS203"
TheList.list.value = "WS204"
TheList.list.value = "WS205"
TheList.list.value = "WS206"
TheList.list.value = "WS207"
TheList.list.value = "WS208"
TheList.list.value = "WS209"
TheList.list.value = "WS210"
TheList.list.value = "WS211"
TheList.list.value = "WS212"
TheList.list.value = "WS213"
TheList.list.value = "WS214"
TheList.list.value = "WS215"
TheList.list.value = "WS216"
TheList.list.value = "WS217"
TheList.list.value = "WS218"
TheList.list.value = "WS219"
TheList.list.value = "WS221"
TheList.list.value = "WS222"
TheList.list.value = "WS223"
TheList.list.value = "WS224"
TheList.list.value = "WS225"
TheList.list.value = "WS226"
TheList.list.value = "WS227"
TheList.list.value = "WS228"
TheList.list.value = "WS229"
TheList.list.value = "WS231"
TheList.list.value = "WS232"
TheList.list.value = "WS233"
TheList.list.value = "WS234"
TheList.list.value = "WS235"
TheList.list.value = "WS236"
TheList.list.value = "WS237"
TheList.list.value = "WS238"
TheList.list.value = "WS239"
endmethod
method arrive(var eventInfo MoveEvent)

;lists choices to be selected by user
;(listed in sequential order)

TheList.list.selection = 1
TheList.list.value = "AL"
TheList.list.selection = 2
TheList.list.value = "AK"
TheList.list.selection = 3
TheList.list.value = "AR"
TheList.list.selection = 4
TheList.list.value = "AS"
TheList.list.selection = 5
TheList.list.value = "AZ"
TheList.list.selection = 6
TheList.list.value = "BC"
TheList.list.selection = 7
TheList.list.value = "CA"
TheList.list.selection = 8
TheList.list.value = "CN"
TheList.list.selection = 9
TheList.list.value = "CO"
TheList.list.selection = 10
TheList.list.value = "CT"
TheList.list.selection = 11
TheList.list.value = "DE"
TheList.list.selection = 12
TheList.list.value = "DC"
TheList.list.selection = 13
TheList.list.value = "DG"
TheList.list.selection = 14
TheList.list.value = "DM"
TheList.list.selection = 15
TheList.list.value = "FL"
TheList.list.selection = 16
TheList.list.value = "FR"
TheList.list.selection = 17
TheList.list.value = "GA"
TheList.list.selection = 18
TheList.list.value = "GM"
TheList.list.selection = 19
TheList.list.value = "GR"
TheList.list.selection = 20
TheList.list.value = "GU"
TheList.list.selection = 21
TheList.list.value = "HI"
TheList.list.selection = 22
TheList.list.value = "IA"
TheList.list.selection = 23
TheList.list.value = "IC"
TheList.list.selection = 24
TheList.list.value = "ID"
TheList.list.selection = 25
TheList.list.value = "IL"
TheList.list.selection = 26
TheList.list.value = "IN"
TheList.list.selection = 27
TheList.list.value = "IT"
TheList.list.selection = 28
TheList.list.value = "KY"
TheList.list.selection = 29
TheList.list.value = "KS"
TheList.list.selection = 30
TheList.list.value = "LA"
TheList.list.selection = 31
TheList.list.value = "IX"
TheList.list.selection = 32
TheList.list.value = "MA"
TheList.list.selection = 33
TheList.list.value = "MD"
TheList.list.selection = 34
TheList.list.value = "ME"
TheList.list.selection = 35
TheList.list.value = "MI"
TheList.list.selection = 36
TheList.list.value = "MN"
TheList.list.selection = 37
TheList.list.value = "MO"
TheList.list.selection = 38
TheList.list.value = "MS"
TheList.list.selection = 39
TheList.list.value = "MT"
TheList.list.selection = 40
TheList.list.value = "NC"
TheList.list.selection = 41
TheList.list.value = "ND"
TheList.list.selection = 42
TheList.list.value = "NE"
TheList.list.selection = 43
TheList.list.value = "NH"
TheList.list.selection = 44
TheList.list.value = "NJ"
TheList.list.selection = 45
TheList.list.value = "NM"
TheList.list.selection = 46
TheList.list.value = "NV"
TheList.list.selection = 47
TheList.list.value = "NY"
TheList.list.selection = 48
TheList.list.selection = 49
TheList.list.value = "NY"
TheList.list.selection = 50
TheList.list.value = "OH"
TheList.list.selection = 51
TheList.list.value = "OK"
TheList.list.selection = 52
TheList.list.value = "OR"
TheList.list.selection = 53
TheList.list.value = "PA"
TheList.list.selection = 54
TheList.list.value = "PR"
TheList.list.selection = 55
TheList.list.value = "PT"
TheList.list.selection = 56
TheList.list.value = "RI"
TheList.list.selection = 57
TheList.list.value = "SC"
TheList.list.selection = 58
TheList.list.value = "SD"
TheList.list.selection = 59
TheList.list.value = "TN"
TheList.list.selection = 60
TheList.list.value = "TX"
TheList.list.selection = 61
TheList.list.value = "UT"
TheList.list.selection = 62
TheList.list.value = "VT"
TheList.list.selection = 63
TheList.list.value = "VI"
TheList.list.selection = 64
TheList.list.value = "UK"
TheList.list.selection = 65
TheList.list.value = "UT"
TheList.list.selection = 66
TheList.list.value = "VA"
TheList.list.selection = 67
TheList.list.value = "VE"
TheList.list.selection = 68
TheList.list.value = "WA"
TheList.list.selection = 69
TheList.list.value = "WI"
TheList.list.selection = 70
TheList.list.value = "WV"
TheList.list.selection = 71
TheList.list.value = "WY"
TheList.list.selection = 72
TheList.list.value = "XK"
;COMMENT: LISTS LAST TWO DIGITS OF YEAR OF VEHICLE
; STARTING WITH 50 TO 01 (1950 TO 2001) ON
; THE VEHICULAR INFORMATION FORM IN THE VEHICLE
; YEAR FIELD

method arrive(var eventInfo MoveEvent)

;lists last two digits of year in order shown below.
;the information is displayed in a drop down box with
;a vertical scroll box.

TheList.list.selection = 1
TheList.list.value = "50"
TheList.list.selection = 2
TheList.list.value = "51"
TheList.list.selection = 3
TheList.list.value = "52"
TheList.list.selection = 4
TheList.list.value = "53"
TheList.list.selection = 5
TheList.list.value = "54"
TheList.list.selection = 6
TheList.list.value = "55"
TheList.list.selection = 7
TheList.list.value = "56"
TheList.list.selection = 8
TheList.list.value = "57"
TheList.list.selection = 9
TheList.list.value = "58"
TheList.list.selection = 10
TheList.list.value = "59"
TheList.list.selection = 11
TheList.list.value = "60"
TheList.list.selection = 12
TheList.list.value = "61"
TheList.list.selection = 13
TheList.list.value = "62"
TheList.list.selection = 14
TheList.list.value = "63"
TheList.list.selection = 15
TheList.list.value = "64"
TheList.list.selection = 16
TheList.list.value = "65"
TheList.list.selection = 17
TheList.list.value = "66"
TheList.list.selection = 18
TheList.list.value = "67"
TheList.list.selection = 19
TheList.list.value = "68"
TheList.list.selection = 20
TheList.list.value = "69"
TheList.list.selection = 21
TheList.list.value = "70"
TheList.list.selection = 22
TheList.list.value = "71"
TheList.list.selection = 23
TheList.list.value = "72"
TheList.list.selection = 24
TheList.list.value = "73"
TheList.list.selection = 25
TheList.list.value = "74"
TheList.list.selection = 26
TheList.list.value = "75"
TheList.list.selection = 27
TheList.list.value = "76"
TheList.list.selection = 28
TheList.list.value = "77"
TheList.list.selection = 29
TheList.list.value = "78"
TheList.list.selection = 30
TheList.list.value = "79"
TheList.list.selection = 31
TheList.list.value = "80"
TheList.list.selection = 32
TheList.list.value = "81"
TheList.list.selection = 33
TheList.list.value = "82"
TheList.list.selection = 34
TheList.list.value = "83"
TheList.list.selection = 35
TheList.list.value = "84"
TheList.list.selection = 36
TheList.list.value = "85"
TheList.list.selection = 37
TheList.list.value = "86"
TheList.list.selection = 38
TheList.list.value = "87"
TheList.list.selection = 39
TheList.list.value = "88"
TheList.list.selection = 40
TheList.list.value = "89"
TheList.list.selection = 41
TheList.list.value = "90"
TheList.list.selection = 42
TheList.list.value = "91"
TheList.list.selection = 43
TheList.list.value = "92"
TheList.list.selection = 44
TheList.list.value = "93"
TheList.list.selection = 45
TheList.list.value = "94"
TheList.list.selection = 46
TheList.list.value = "95"
TheList.list.selection = 47
TheList.list.value = "96"
TheList.list.selection = 48
TheList.list.value = "97"
TheList.list.selection = 49
TheList.list.value = "98"
TheList.list.selection = 50
TheList.list.value = "99"
TheList.list.selection = 51
TheList.list.value = "00"
TheList.list.selection = 52
TheList.list.value = "01"
endmethod
METHOD: TheList::arrive

;COMMENT: LISTS VEHICLE MAKE ON THE VEHICULAR INFORMATION
;SCREEN

method arrive(var eventInfo MoveEvent)

;lists vehicle makes in the order shown below in a
drop down window with a scroll box

TheList.list.selection = 1
TheList.list.value = "ACURA"
TheList.list.selection = 2
TheList.list.value = "ALPHA"
TheList.list.selection = 3
TheList.list.value = "AMC"
TheList.list.selection = 4
TheList.list.value = "AUDI"
TheList.list.selection = 5
TheList.list.value = "BENZ"
TheList.list.selection = 6
TheList.list.value = "BMW"
TheList.list.selection = 7
TheList.list.value = "BUICK"
TheList.list.selection = 8
TheList.list.value = "CAD"'
TheList.list.selection = 9
TheList.list.value = "CHEV"
TheList.list.selection = 10
TheList.list.value = "CHRY"
TheList.list.selection = 11
TheList.list.value = "DATSU"
TheList.list.selection = 12
TheList.list.value = "DODGE"
TheList.list.selection = 13
TheList.list.value = "EAGLE"
TheList.list.selection = 14
TheList.list.value = "FIAT"
TheList.list.selection = 15
TheList.list.value = "FORD"
TheList.list.selection = 16
TheList.list.value = "GEO"
TheList.list.selection = 17
TheList.list.value = "GMC"
TheList.list.selection = 18
TheList.list.value = "HONDA"
TheList.list.selection = 19
TheList.list.value = "HRLY"
TheList.list.selection = 20
TheList.list.value = "HYUN"
TheList.list.selection = 21
TheList.list.value = "INFI"
TheList.list.selection = 22
TheList.list.value = "ISUZU"
TheList.list.selection = 23
TheList.list.value = "JAGU"
TheList.list.selection = 24
TheList.list.value = "JEET"
TheList.list.selection = 25
TheList.list.value = "KAWA"
TheList.list.selection = 26
TheList.list.value = "LEXUS"
TheList.list.selection = 27
TheList.list.value = "LINC"
TheList.list.selection = 28
TheList.list.value = "MAZDA"
TheList.list.selection = 29
TheList.list.value = "MERC"
TheList.list.selection = 30
TheList.list.value = "M3"
TheList.list.selection = 31
TheList.list.value = "MITSU"
TheList.list.selection = 32
TheList.list.value = "NISS"
TheList.list.selection = 33
TheList.list.value = "OLDS"
TheList.list.selection = 34
TheList.list.value = "PEUGO"
TheList.list.selection = 35
TheList.list.value = "PLYM"
TheList.list.selection = 36
TheList.list.value = "PONTI"
TheList.list.selection = 37
TheList.list.value = "POR5"
TheList.list.selection = 38
TheList.list.value = "RENAU"
TheList.list.selection = 39
TheList.list.value = "SAAB"
TheList.list.selection = 40
TheList.list.value = "SATUR"
TheList.list.selection = 41
TheList.list.value = "SUBA"
TheList.list.selection = 42
TheList.list.value = "SUZI"
TheList.list.selection = 43
TheList.list.value = "TOYO"
TheList.list.selection = 44
TheList.list.value = "VOLVO"
TheList.list.selection = 45
TheList.list.value = "VW"
TheList.list.selection = 46
TheList.list.value = "YAMA"
TheList.list.selection = 47
TheList.list.value = "YUGO"
TheList.list.selection = 48
TheList.list.value = "OTHER"

endmethod
; COMMENT: LISTS VEHICLE COLOR ON THE VEHICULAR INFORMATION
; SCREEN

method arrive(var eventInfo MoveEvent)

; lists vehicle colors in the vehicle color field (in the
; order shown below)
    TheList.list.selection = 1
    TheList.list.value = "BEIGE"
    TheList.list.selection = 2
    TheList.list.value = "BLACK"
    TheList.list.selection = 3
    TheList.list.value = "BLUE"
    TheList.list.selection = 4
    TheList.list.value = "BRONZE"
    TheList.list.selection = 5
    TheList.list.value = "BROWN"
    TheList.list.selection = 6
    TheList.list.value = "GOLD"
    TheList.list.selection = 7
    TheList.list.value = "GRAY"
    TheList.list.selection = 8
    TheList.list.value = "GREEN"
    TheList.list.selection = 9
    TheList.list.value = "MAROON"
    TheList.list.selection = 10
    TheList.list.value = "MUSTARD"
    TheList.list.selection = 11
    TheList.list.value = "ORANGE"
    TheList.list.selection = 12
    TheList.list.value = "PEACH"
    TheList.list.selection = 13
    TheList.list.value = "PINK"
    TheList.list.selection = 14
    TheList.list.value = "PURPLE"
    TheList.list.selection = 15
    TheList.list.value = "RED"
    TheList.list.selection = 16
    TheList.list.value = "RUST"
    TheList.list.selection = 17
    TheList.list.value = "SILVER"
    TheList.list.selection = 18
    TheList.list.value = "TEAL"
    TheList.list.selection = 19
    TheList.list.value = "WHITE"
    TheList.list.selection = 20
    TheList.list.value = "YELLOW"
    TheList.list.selection = 21
    TheList.list.value = "OTHER"

endmethod
;COMMENT: GOES BACK TO CUSTOMER INFORMATION FORM
; FROM DRIVER LICENSE FORM

method pushButton(var eventInfo Event)
; declares customer table and form

    var
custTbl Table
custForm Form
endVar

; reverts back to customer form fm driver license
; form by opening Customer table and form, and
; then closes the Driver License Information Screen

custTbl.attach("Customer.DB")
if not lock(custTbl, "FULL", "customer.db", "FULL") then
endif
custForm.open("Customer")
custForm.show()
close()
endmethod
;COMMENT:  RETURNS DRIVER LICENSE FORM TO MAIN FORM

method pushButton(var eventInfo Event)

;declares Main Menu Form

    var
    mainForm Form; declaring MainMenu form.
endVar

;returns to the Main Menu form from the Driver License form and then closes the Driver License Information Screen

    mainForm.open("mainmenu"); opens Main Menu.
    mainForm.show()
    close()
endmethod
method arrive(var eventInfo MoveEvent)

; lists year 95 thru 01 (1995 - 2001) in the order
; shown below. the list is displayed in a drop
; down scroll box.

TheList.list.selection = 1
TheList.list.value = "95"
TheList.list.selection = 2
TheList.list.value = "96"
TheList.list.selection = 3
TheList.list.value = "97"
TheList.list.selection = 4
TheList.list.value = "98"
TheList.list.selection = 5
TheList.list.value = "99"
TheList.list.selection = 6
TheList.list.value = "00"
TheList.list.selection = 7
TheList.list.value = "01"

endmethod

method arrive(var eventInfo MoveEvent)

; lists decal colors in the decal color field (in the
; order shown below)

TheList.list.selection = 1
TheList.list.value = "BLUE"
TheList.list.selection = 2
TheList.list.value = "GREEN"
TheList.list.selection = 3
TheList.list.value = "RED"
TheList.list.selection = 4
TheList.list.value = "WHITE"
TheList.list.selection = 5
TheList.list.value = "BLACK"

endmethod
method arrive(var eventInfo MoveEvent)

; lists month from 01 thru 12 in the order shown below.
; the list is shown in a drop down scroll box.

TheList.list.selection = 1
TheList.list.value = "01"
TheList.list.selection = 2
TheList.list.value = "02"
TheList.list.selection = 3
TheList.list.value = "03"
TheList.list.selection = 4
TheList.list.value = "04"
TheList.list.selection = 5
TheList.list.value = "05"
TheList.list.selection = 6
TheList.list.value = "06"
TheList.list.selection = 7
TheList.list.value = "07"
TheList.list.selection = 8
TheList.list.value = "08"
TheList.list.selection = 9
TheList.list.value = "09"
TheList.list.selection = 10
TheList.list.value = "10"
TheList.list.selection = 11
TheList.list.value = "11"
TheList.list.selection = 12
TheList.list.value = "12"

endmethod
method arrive(var eventInfo MoveEvent)

; lists ticket violation code (01-99) in the order shown ; below. the list is displayed in a drop down scroll box.

TheList.list.selection = 1
TheList.list.value = "01"
TheList.list.selection = 2
TheList.list.value = "02"
TheList.list.selection = 3
TheList.list.value = "03"
TheList.list.selection = 4
TheList.list.value = "04"
TheList.list.selection = 5
TheList.list.value = "05"
TheList.list.selection = 6
TheList.list.value = "06"
TheList.list.selection = 7
TheList.list.value = "07"
TheList.list.selection = 8
TheList.list.value = "08"
TheList.list.selection = 9
TheList.list.value = "09"
TheList.list.selection = 10
TheList.list.value = "10"
TheList.list.selection = 11
TheList.list.value = "11"
TheList.list.selection = 12
TheList.list.value = "12"
TheList.list.selection = 13
TheList.list.value = "13"
TheList.list.selection = 14
TheList.list.value = "14"
TheList.list.selection = 15
TheList.list.value = "15"
TheList.list.selection = 16
TheList.list.value = "16"
TheList.list.selection = 17
TheList.list.value = "17"
TheList.list.selection = 18
TheList.list.value = "18"
TheList.list.selection = 19
TheList.list.value = "19"
TheList.list.selection = 20
TheList.list.value = "20"
TheList.list.selection = 21
TheList.list.value = "21"
TheList.list.selection = 22
TheList.list.value = "22"
TheList.list.selection = 23
TheList.list.value = "23"
TheList.list.selection = 24
TheList.list.value = "24"
TheList.list.selection = 25
TheList.list.value = "25"
TheList.list.selection = 26
TheList.list.value = "26"
TheList.list.selection = 27
TheList.list.value = "27"
TheList.list.selection = 28
TheList.list.value = "28"
TheList.list.selection = 29
TheList.list.value = "29"
TheList.list.selection = 30
TheList.list.value = "30"
TheList.list.selection = 31
TheList.list.value = "31"
TheList.list.selection = 32
TheList.list.value = "32"
TheList.list.selection = 33
TheList.list.value = "33"
TheList.list.selection = 34
TheList.list.value = "34"
TheList.list.selection = 35
TheList.list.value = "35"
TheList.list.selection = 36
TheList.list.value = "36"
TheList.list.selection = 37
TheList.list.value = "37"
TheList.list.selection = 38
TheList.list.value = "38"
TheList.list.selection = 39
TheList.list.value = "39"
TheList.list.selection = 40
TheList.list.value = "40"
TheList.list.selection = 41
TheList.list.value = "41"
TheList.list.selection = 42
TheList.list.value = "42"
TheList.list.selection = 43
TheList.list.value = "43"
TheList.list.selection = 44
TheList.list.value = "44"
TheList.list.selection = 45
TheList.list.value = "45"
TheList.list.selection = 46
TheList.list.value = "46"
TheList.list.selection = 47
TheList.list.value = "47"
TheList.list.selection = 48
TheList.list.value = "48"
TheList.list.selection = 49
TheList.list.value = "49"
TheList.list.selection = 50
TheList.list.value = "50"
TheList.list.selection = 51
TheList.list.value = "51"
TheList.list.selection = 52
TheList.list.value = "52"
TheList.list.selection = 53
TheList.list.value = "53"
TheList.list.selection = 54
TheList.list.value = "54"
TheList.list.selection = 55
TheList.list.value = "55"
TheList.list.selection = 56
TheList.list.value = "56"
TheList.list.selection = 57
TheList.list.value = "57"
TheList.list.selection = 58
TheList.list.value = "58"
TheList.list.selection = 59
TheList.list.value = "59"
TheList.list.selection = 60
TheList.list.value = "60"
TheList.list.selection = 61
TheList.list.value = "61"
TheList.list.selection = 62
TheList.list.value = "62"
TheList.list.selection = 63
TheList.list.value = "63"
TheList.list.selection = 64
TheList.list.value = "64"
TheList.list.selection = 65
TheList.list.value = "65"
TheList.list.selection = 66
TheList.list.value = "66"
TheList.list.selection = 67
TheList.list.value = "67"
TheList.list.selection = 68
TheList.list.value = "68"
TheList.list.selection = 69
TheList.list.value = "69"
TheList.list.selection = 70
TheList.list.value = "70"
TheList.list.selection = 71
TheList.list.value = "71"
TheList.list.selection = 72
TheList.list.value = "72"
TheList.list.selection = 73
TheList.list.value = "73"
TheList.list.selection = 74
TheList.list.value = "74"
TheList.list.selection = 75
TheList.list.value = "75"
TheList.list.selection = 76
TheList.list.value = "76"
TheList.list.selection = 77
TheList.list.value = "77"
TheList.list.selection = 78
TheList.list.value = "78"
TheList.list.selection = 79
TheList.list.value = "79"
TheList.list.selection = 80
TheList.list.value = "80"
TheList.list.selection = 81
TheList.list.value = "81"
TheList.list.selection = 82
TheList.list.value = "82"
TheList.list.selection = 83
TheList.list.value = "83"
TheList.list.selection = 84
TheList.list.value = "84"
TheList.list.selection = 85
TheList.list.value = "85"
TheList.list.selection = 86
TheList.list.value = "86"
TheList.list.selection = 87
TheList.list.value = "87"
TheList.list.selection = 88
TheList.list.value = "88"
TheList.list.selection = 89
TheList.list.value = "89"
TheList.list.selection = 90
TheList.list.value = "90"
TheList.list.selection = 91
TheList.list.value = "91"
TheList.list.selection = 92
TheList.list.value = "92"
TheList.list.selection = 93
TheList.list.value = "93"
TheList.list.selection = 94
TheList.list.value = "94"
TheList.list.selection = 95
TheList.list.value = "95"
TheList.list.selection = 96
TheList.list.value = "96"
TheList.list.selection = 97
TheList.list.value = "97"
TheList.list.selection = 98
TheList.list.value = "98"
TheList.list.selection = 99
TheList.list.value = "99"
endmethod
method arrive(var eventInfo MoveEvent)

; lists points awarded for traffic violations from 01-99
; as shown below. the list is displayed in a drop down
; scroll box.

TheList.list.selection = 1
TheList.list.value = "01"
TheList.list.selection = 2
TheList.list.value = "02"
TheList.list.selection = 3
TheList.list.value = "03"
TheList.list.selection = 4
TheList.list.value = "04"
TheList.list.selection = 5
TheList.list.value = "05"
TheList.list.selection = 6
TheList.list.value = "06"
TheList.list.selection = 7
TheList.list.value = "07"
TheList.list.selection = 8
TheList.list.value = "08"
TheList.list.selection = 9
TheList.list.value = "09"
TheList.list.selection = 10
TheList.list.value = "10"
TheList.list.selection = 11
TheList.list.value = "11"
TheList.list.selection = 12
TheList.list.value = "12"
TheList.list.selection = 13
TheList.list.value = "13"
TheList.list.selection = 14
TheList.list.value = "14"
TheList.list.selection = 15
TheList.list.value = "15"
TheList.list.selection = 16
TheList.list.value = "16"
TheList.list.selection = 17
TheList.list.value = "17"
TheList.list.selection = 18
TheList.list.value = "18"
TheList.list.selection = 19
TheList.list.value = "19"
TheList.list.selection = 20
TheList.list.value = "20"
TheList.list.selection = 21
TheList.list.value = "21"
TheList.list.selection = 22
TheList.list.value = "22"
TheList.list.selection = 23
TheList.list.value = "23"
TheList.list.selection = 24
TheList.list.value = "24"
TheList.list.selection = 25
TheList.list.value = "25"
TheList.list.selection = 26
TheList.list.value = "26"
TheList.list.selection = 27
TheList.list.value = "27"
TheList.list.selection = 28
TheList.list.value = "28"
TheList.list.selection = 29
TheList.list.value = "29"
TheList.list.selection = 30
TheList.list.value = "30"
TheList.list.selection = 31
TheList.list.value = "31"
TheList.list.selection = 32
TheList.list.value = "32"
TheList.list.selection = 33
TheList.list.value = "33"
TheList.list.selection = 34
TheList.list.value = "34"
TheList.list.selection = 35
TheList.list.value = "35"
TheList.list.selection = 36
TheList.list.value = "36"
TheList.list.selection = 37
TheList.list.value = "37"
TheList.list.selection = 38
TheList.list.value = "38"
TheList.list.selection = 39
TheList.list.value = "39"
TheList.list.selection = 40
TheList.list.value = "40"
TheList.list.selection = 41
TheList.list.value = "41"
TheList.list.selection = 42
TheList.list.value = "42"
TheList.list.selection = 43
TheList.list.value = "43"
TheList.list.selection = 44
TheList.list.value = "44"
TheList.list.selection = 45
TheList.list.value = "45"
TheList.list.selection = 46
TheList.list.value = "46"
TheList.list.selection = 47
TheList.list.value = "47"
TheList.list.selection = 48
TheList.list.value = "48"
TheList.list.selection = 49
TheList.list.value = "49"
TheList.list.selection = 50
TheList.list.value = "50"
TheList.list.selection = 51
TheList.list.value = "51"
TheList.list.selection = 52
TheList.list.value = "52"
TheList.list.selection = 53
TheList.list.value = "53"
TheList.list.selection = 54
TheList.list.value = "54"
TheList.list.selection = 55
TheList.list.value = "55"
TheList.list.selection = 56
TheList.list.value = "56"
TheList.list.selection = 57
TheList.list.value = "57"
TheList.list.selection = 58
TheList.list.value = "58"
TheList.list.selection = 59
TheList.list.value = "59"
TheList.list.selection = 60
TheList.list.value = "60"
TheList.list.selection = 61
TheList.list.value = "61"
TheList.list.selection = 62
TheList.list.value = "62"
TheList.list.selection = 63
TheList.list.value = "63"
TheList.list.selection = 64
TheList.list.value = "64"
TheList.list.selection = 65
TheList.list.value = "65"
TheList.list.selection = 66
TheList.list.value = "66"
TheList.list.selection = 67
TheList.list.value = "67"
TheList.list.selection = 68
TheList.list.value = "68"
TheList.list.selection = 69
TheList.list.value = "69"
TheList.list.selection = 70
TheList.list.value = "70"
TheList.list.selection = 71
TheList.list.value = "71"
TheList.list.selection = 72
TheList.list.value = "72"
TheList.list.selection = 73
TheList.list.value = "73"
TheList.list.selection = 74
TheList.list.value = "74"
TheList.list.selection = 75

163
TheList.list.value = "75"
TheList.list.selection = 76
TheList.list.value = "76"
TheList.list.selection = 77
TheList.list.value = "77"
TheList.list.selection = 78
TheList.list.value = "78"
TheList.list.selection = 79
TheList.list.value = "79"
TheList.list.selection = 80
TheList.list.value = "80"
TheList.list.selection = 81
TheList.list.value = "81"
TheList.list.selection = 82
TheList.list.value = "82"
TheList.list.selection = 83
TheList.list.value = "83"
TheList.list.selection = 84
TheList.list.value = "84"
TheList.list.selection = 85
TheList.list.value = "85"
TheList.list.selection = 86
TheList.list.value = "86"
TheList.list.selection = 87
TheList.list.value = "87"
TheList.list.selection = 88
TheList.list.value = "88"
TheList.list.selection = 89
TheList.list.value = "89"
TheList.list.selection = 90
TheList.list.value = "90"
TheList.list.selection = 91
TheList.list.value = "91"
TheList.list.selection = 92
TheList.list.value = "92"
TheList.list.selection = 93
TheList.list.value = "93"
TheList.list.selection = 94
TheList.list.value = "94"
TheList.list.selection = 95
TheList.list.value = "95"
TheList.list.selection = 96
TheList.list.value = "96"
TheList.list.selection = 97
TheList.list.value = "97"
TheList.list.selection = 98
TheList.list.value = "98"
TheList.list.selection = 99
TheList.list.value = "99"
endmethod
method pushButton(var eventInfo Event)
;
declares Custom report

    var
custRptReport
endVar

custRpt.open("Custom")
endmethod


method pushButton(var eventInfo Event)
;
declares vehicle report

    var
    vehiRptReport
endVar

;see comment above

    vehiRpt.open("Vehicle")
endmethod
method pushButton(var eventInfo Event)

; declares Decal Report

var
    decaRptReport
endVar

; see comment above

decaRpt.open("Decal")

endmethod

method pushButton(var eventInfo Event)

; declares Ticket report

var
    tickRptReport
endVar

; see comment above

tickRpt.open("ticket")

endmethod
APPENDIX J. LEVRAS USER'S MANUAL

This appendix contains instructions on how to use the Law Enforcement and Vehicle Administration Registration System. The following pages are designed to be a ready reference manual for user's that need additional LEVRAS knowledge. A copy of Appendix J will be provided to the Naval Postgraduate School Security Department during system installation.
LEVRAS USER'S MANUAL

THE NAVAL POSTGRADUATE SCHOOL
SECURITY DEPARTMENT

THE LAW ENFORCEMENT AND VEHICLE REGISTRATION
ADMINISTRATION SYSTEM (LEVRAS)

DESIGNERS
CDR MARK S. NAULT
LT HENRY M. MCGIBBON

MONTEREY, CALIFORNIA
SEPTEMBER 1995
# TABLE OF CONTENTS

I. GENERAL .................................................. 1

   1.1 - Purpose of the User's Manual ....................... 1
   1.2 - Points of Contact .................................. 1
   1.3 - Software Development ............................... 2
   1.4 - Deliverables ......................................... 2
   1.5 - Diskette Files List ................................ 2

II. SYSTEM INFORMATION ..................................... 5

   2.1 - Hardware/Software Requirements ...................... 5
   2.2 - Installing LEVRAS .................................. 5
   2.3 - System Callup/Passwords ............................. 5
   2.4 - Log-Out and Power-down Procedures .................. 6
   2.5 - Housekeeping ....................................... 6
   2.6 - Error Message ...................................... 7

III. DETAILED PROGRAM DESCRIPTION ....................... 8

   3.1 - Tutorial for Welcome Screen ......................... 8
   3.2 - Tutorial for Main Menu ............................... 8
   3.3 - Tutorial for Customer Data Processing ............... 9
   3.4 - Tutorial for Driver License Data Processing ....... 9
   3.5 - Tutorial for Insurance Data Processing ............. 10
   3.6 - Tutorial for Registration Data Processing .......... 10
   3.7 - Tutorial for Vehicle Data Processing ............... 10
3.8 - Tutorial for Decal Data Processing ............................................. 10
3.9 - Tutorial for Ticket Data Processing ............................................ 10
3.10 - Tutorial for Ticket Administration ........................................... 11
3.11 - Tutorial for Displaying and Printing Reports ............................. 11
3.12 - Tutorial for Archive Data Processing ....................................... 11

IV. LEVRAS CODES ............................................................................ 13

4.1 - State and Country Codes ......................................................... 13
4.2 - Decal Color Codes ................................................................. 14
4.3 - Employee Type Codes ............................................................. 14
4.4 - Grades Codes ......................................................................... 15
4.5 - Rank Codes ............................................................................. 15
4.6 - Ticket Violation Codes ............................................................. 15

V. USER'S MANUAL REFERENCES ....................................................... 19
I. GENERAL

1.1 Purpose of the User's Manual. This manual is to provide users of the Law Enforcement and Vehicle Registration Administration System (LEVRAS) with the necessary information required to operate its program. After review of this manual, the user will have the required knowledge to add, modify, delete, query and print vehicle registration and ticket information with regards to a vehicle registration customer at the Naval Postgraduate School, Monterey, California. The LEVRAS User's Manual will assume that the user has no previous knowledge of the LEVRAS program. Thus, this manual will become a valuable resource/training aid for all LEVRAS program users.

1.2 Points of Contact. All questions/concerns pertaining to LEVRAS will be forwarded to the individuals listed below (address of the LEVRAS designers remains current until September, 1995):

Superintendent, Code 370
Naval Postgraduate School
Monterey, CA 93943-5000
(408) 656-2536

Attention: LEVRAS Designers
CDR M. S. Nault
LT H. M. McGibbon

After September 1995, contact the Naval Postgraduate School's Management Information System (MIS) Office at (408) 656-2195. The MIS office is located in Herrmann Hall room E-204. For specific LEVRAS related questions, contact the Computer Specialist, Renee A. Lightcap or successor at extension 1066 (e-mail address: rlighcap@nps.navy.mil).
1.3 Software Development. The LEVRAS system program was developed by two Naval Postgraduate students: CDR Mark S. Nault and LT Henry M. McGibbon for their Information Technology Management (ITM) thesis. SALSA (a semantic object modeling methodology tool) by Wall Data, Incorporated and Paradox, version 4.5 for Windows (a relational database language) by Borland International, Incorporated were extensively used for the LEVRAS program design and development. The LEVRAS system application program and documentation are the exclusive property of the Naval Postgraduate School and the U.S. Navy.

1.4 Deliverables. The deliverables for the LEVRAS program include:

a. One (1) 3.5 floppy diskette containing the LEVRAS Program.
b. One (1) LEVRAS Users Manual.

Note: Saved code generation data files (*.SSL, *.FSL and *.RSL) will be retained by the NPS Security Officer for back-up purposes.

1.5 Diskette Files List. The 68 LEVRAS files listed below, when interfaced with Paradox, will enable the user to operate LEVRAS program.

What makes that LEVRAS operate? Files...and more files!
Refer to the following page for the LEVRAS file listing.

```
C:\
ARChive.DB
ARChive.FDL
ARChive.PX
ARChive.QBE
ARChive.VAL
BEGIN.SDL
CUSToMER.RSL
CUSToMER.FDL
CUSToMER.QBE
DECAL.FDL
DECAL.RDL
DRIVERL1.VAL
DvRLISeN.FDL
GEMSETup.DB
GEMSETup.TXT
INSURANCE.FDL
MAINMENU.FDL
PDoxwoRK.INI
PROCESST.FDL
REGISTER.FDL
REPORTS.FDL
TICKet.FDL
TICKet.RDL
VEHICLE.FDL
VEHICLE.RDL
WELCOME.FDL

C:\SALSA\n
CUSToMER.DB
CUSToMER.PX
CUSToMER.VAL
DECAL.DB
DECAL.PX
DECAL.VAL
DECAL.X06
DECAL.X07
DECAL.Y06
DECAL.Y07
DRIVERL1.DB
DRIVERL1.PX
DRIVERL1.VAL
DRIVERL1.X04
DRIVERL1.Y04
INSURANCE.DB
INSURANCE.PX
INSURANCE.VAL
INSURANCE.X04
INSURANCE.X05
INSURANCE.Y04
INSURANCE.Y05
REGISTRA.DB
REGISTRA.PX
REGISTRA.VAL
REGISTRA.X04
REGISTRA.X05
REGISTRA.Y04
REGISTRA.Y05
TICKET.DB
TICKET.PX
TICKET.VAL
TICKET.XOB
TICKET.XOC
TICKET.YOB
TICKET.YOC
USAENGL.RSL
VEHICLE.DB
VEHICLE.PX
VEHICLE.VAL
VEHICLE.X08
VEHICLE.Y08
```
The LEVRAS file extensions are correlated to their respective file types:

<table>
<thead>
<tr>
<th>Extension</th>
<th>Type of File</th>
</tr>
</thead>
<tbody>
<tr>
<td>*.DB</td>
<td>Table</td>
</tr>
<tr>
<td>*.FDL</td>
<td>Delivered Form</td>
</tr>
<tr>
<td>*.FSL</td>
<td>Saved Form</td>
</tr>
<tr>
<td>*.FTL</td>
<td>Temporary Form</td>
</tr>
<tr>
<td>*.INI</td>
<td>Initializing Configuration File</td>
</tr>
<tr>
<td>*.PX</td>
<td>Primary Index of File</td>
</tr>
<tr>
<td>*.QBE</td>
<td>Saved Query</td>
</tr>
<tr>
<td>*.RDL</td>
<td>Delivered Report</td>
</tr>
<tr>
<td>*.RSL</td>
<td>Saved Report</td>
</tr>
<tr>
<td>*.SDL</td>
<td>Delivered Script</td>
</tr>
<tr>
<td>*.SSL</td>
<td>Saved Script</td>
</tr>
<tr>
<td>*.VAL</td>
<td>Validity Table Check</td>
</tr>
<tr>
<td>*.XOn</td>
<td>Secondary Single-Field Index for a Table</td>
</tr>
<tr>
<td>*.YOn</td>
<td>Secondary Single-Field Index for a Table</td>
</tr>
</tbody>
</table>
II. SYSTEM INFORMATION

2.1 Hardware/Software Requirements. The following microcomputer hardware configuration should support the minimum requirements for using the LEVRAS software program (assuming that the user does not have a LEVRAS Information System): a) An IBM compatible 386, 25 MHZ, microcomputer with a 3.5" floppy disk drive and at least 4 MB of RAM and 22 MB free hard disk space (for both Paradox software and LEVRAS program files). b) Microsoft Windows (version 3.1 or higher). c) A 132-character width printer that uses the standard 8.5" x 11" paper. d) A mouse. e) An VGA or higher video card.

2.2 Installing LEVRAS. Power up your microcomputer using normal power up procedures. Start Windows and choose File Manager. Create a C:\SALSA directory. Using File Manager copy B:\ files to the C:\ directory. Also, copy the B:\SALSA files to the C:\SALSA directory. Exit File Manager. Start Paradox for Windows. Your microcomputer is now ready to callup the LEVRAS program.

To prevent loss of LEVRAS program files, it is strongly recommended that you make a backup of all LEVRAS files onto another 3.5" diskette. One diskette will be the master and the other a working copy. From the DOS C:\> prompt, copy all LEVRAS files by typing, "DISKCOPY A: A:" (a: is assumed as the 3.5" floppy drive, if b: is the 3.5" disk drive, use b: in lieu of a:). Insert the LEVRAS SOURCE diskette into drive A. Push the return key. After your microcomputer prompts you to insert the target diskette, remove the LEVRAS source diskette and insert the target diskette. When prompted, name your volume label: WORKINGCOPY. Now you have a LEVRAS program master and working copy.

2.3 System Callup/Passwords. Once the LEVRAS system software is installed, callup the LEVRAS program by clicking on the SCRIPT pushbutton (located on the speedbar). Select BEGIN.SDL and click on OK. The "ABOUT LEVRAS" dialogue box will appear and when finished reading the box contents, click on OK. The Welcome screen will appear (where you will see two handsome LEVRAS designers). Press start to begin the LEVRAS program.
The LEVRAS program software is password protected (in compliance with the Privacy Act Law of 1974); therefore, when prompted at the "PASSWORD" dialogue box for the system password, type> POLO (be aware, Paradox passwords are case sensitive and would not normally be disclosed in a user's manual). Each table in LEVRAS is password protected; however, you need only type the password once to get into the LEVRAS program when prompted for the password in the PASSWORD Window. If successful at entering in the correct password, all system screens will become fully functional. For further instructions on the LEVRAS program, refer to the DETAILED PROGRAM DESCRIPTION in Part III of this manual.

To change the LEVRAS password (which is recommended on a routine basis) select File -> Utilities -> Restructure (select the first table), under Table Properties scroll to Password Security and select Modify. Type in the new password and click OK then Verify and Save. To test the new LEVRAS password, exit Paradox. Restart Paradox to activate the new Password. Repeat the procedure above to change all system table passwords. It is recommended that the same password be used for each table (*.DB file).

2.4 Log-out and Power-down Procedures. To log-out of the LEVRAS program, you must be in the "Main Menu" window. Once there, click on the EXIT pushbutton. This action will exit you out of the LEVRAS program altogether. To exit Paradox select File, then Exit. Follow the normal procedures to exit MicroSoft Windows completely (refer to MicroSoft Windows User's Manual). To power-down the microcomputer, refer to the local SOP instructions or your hardware user's manual.

2.5 Housekeeping. LEVRAS is intended to retain a large amount of information that pertains to Naval Postgraduate School (NPS) affiliated customers that register their vehicles at the NPS Vehicle Registration Office. Since there is a substantial amount of information, it is strongly recommended that the user periodically save his/her data at intervals that would
require no more than one half-hour worth of work to reconstruct, if temporary loss of power or some other mishap should occur.

2.6 Error Message. If LEVRAS should produce an error message during program operation, a number of user and/or computer software related problems might have occurred. The user should discontinue the activity that is causing the error and save data that has been previously processed before continuing on with any further computer related process. Numerous Message stop dialogue boxes and status line messages have been programmed in the LEVRAS computer code to assist the user if an error should occur.
III. DETAILED PROGRAM DESCRIPTION

3.1 Tutorial for Welcome Screen. The Welcome Screen is called, "Welcome to the Law Enforcement Vehicle Registration and Administration System". The Welcome screen has one pushbutton (and two smiling faces). Click on the Start pushbutton to commence the LEVRAS program.

3.2 Tutorial for Main Menu. The Main Menu has seven pushbuttons (CUSTOMER DATA, PROCESS TICKET, QUERIES, REPORTS, ARCHIVE, HELP, and EXIT).

CUSTOMER DATA: Clicking on this pushbutton takes the user to the Customer Information form for data insertion. The remaining data entry forms follow the Customer Information form to make a complete customer record.

PROCESS TICKET: Clicking on this pushbutton takes the user to the Process Ticket Information form, which is used by the NPS Ticket Administrator. This screen is normally used when a complete customer record has been entered in by the Vehicle Registration Administrator.

QUERIES: Clicking on this pushbutton provides instructions on how to perform a Customer.qbe (query) via a dialog box. The dialog box specifically states how to perform a query. When the "Select File Box" appears, choose the desired query (*.qbe file). After you choose the desired query file, place a check on the desired fields by clicking on the box next to field name. For specifics on the powerful Paradox query capabilities refer to page 323, table 16-2 of the Guide to ObjectPal Paradox 4.5 for Windows software literature.

REPORTS: Clicking on this pushbutton takes the user to the LEVRAS Reports Menu, to select one of the four output reports.

ARCHIVE: Clicking on this pushbutton takes the user to the Archive Information form to allow archive data entry and queries.

HELP: Clicking on this pushbutton takes the user to the LEVRAS Help dialog box.
3.3 Tutorial for Customer Data Processing. This form has six pushbuttons (ADD, MODIFY, DELETE, FWD, HOME, and HELP). Each of these pushbuttons are self-explanatory; however, use the F9 key on your keyboard to get into edit mode (if prompted). To enter a new customer into the LEVRAS database, click on the ADD pushbutton and start entering in the fields. In the SSN field dashes will automatically appear to separate the social security numbers, and only numbers can be entered. All fields that require letters or alphanumeric data entries will capitalize the letters automatically. Scroll box fields require the user to either enter in their own data or use the scroll box information (to maintain a standard database, the designers recommend using the scroll box information). The phone number fields require that the user first push the spacebar on their keyboard to produce the open parenthesis bracket for the area code. The other area code parenthetical bracket and dash will automatically appear without any user intervention. The SMC#, Curric/Staff and Faculty fields require no data or the appropriate number of data characters to completely fill the fields. The Database Entry Date field requires the following format MM/DD/YY. The month and day data elements require a zero in front of a single number month or day (for example, April will require the user enter 04).

For CUSTOMER INFORMATION form fields that request for "Home State, Employee Type, Rank, Sex, and Grade" refer to Part IV - LEVRAS CODES of this User's Manual. To delete customer information, push the delete button (data entered on other forms will not be affected). To enter other vehicle registration information press FWD (takes you to the Driver License Information form) or press HOME (takes you back to the Main Menu).

3.4 Tutorial for Driver License Data Processing. The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. To go back to the Customer
3.5 **Tutorial for Insurance Data Processing.** The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. To go back to the Driver License Information form press BACK. To go forward to the Registration Information form press FWD.

3.6 **Tutorial for Registration Data Processing.** The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. To go back to the Insurance Information form press BACK. To go forward to the Vehicular Information form press FWD.

3.7 **Tutorial for Vehicle Data Processing.** The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. Additional scroll box fields, which operate similar to the Customer Information scroll box fields are: Vehicle Year, Vehicle Make, and Vehicle Color. To go back to the Registration Information form press BACK. To go forward to the Decal Information form press FWD.

3.8 **Tutorial for Decal Data Processing.** The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. Additional scroll box fields, which operate similar to the Customer Information scroll box fields are: Decal Color, Decal Expiration Date, and Decal Expiration Year. To go back to the Vehicular Information form press BACK. To go forward to the Ticket Information form press FWD.
3.9 **Tutorial for Ticket Data Processing.** The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. Additional scroll box fields, which operate similar to the Customer Information scroll box fields are Violation Code and Points. To go back to the Decal Information form press BACK. To go forward to the Process Ticket Information form press PROC TKT.

3.10 **Tutorial for Ticket Administration.** This form was intended for the Ticket Administrator. Basically, it provides an all encompassing view of a customer and any tickets. It should be used to locate a customer, and update the ticket status (for information that was previously entered on the other data forms). To get the most out of this form, first locate a record that needs a ticket be updated by pushing the LOCATE pushbutton. Follow the instructions in the LOCATE dialog box to locate the intended customer record. The other pushbuttons operate similar to the Customer Information form. The Decal, Vehicle, and Ticket scroll boxes can be modified by pressing the MODIFY pushbutton or press F9 on the computer keyboard to edit. Press FWD to return to the Ticket Information form. Press HOME to return to the Main Menu.

3.11 **Tutorial for Displaying and Printing Reports.** The LEVRAS Reports menu lists four types of reports: CUST/VEH (Customer/Vehicle), DECAL, TICKET, and CUSTOM. Upon selection of a particular report, the report will display one customer record. The user can use the speedbar to locate a particular record (see Ticket Administration, Section 3.10) or scroll through each record. To leave a called up report, go to the upper left corner of the report window and choose Close. The CUST/VEH report displays a customer and his or her vehicles. The DECAL report displays a customer and his or her decals. The TICKET report displays a customer with his or her decals, vehicles, and tickets. The CUSTOM report displays a temporary suspension of driving privileges memo from the NPS Security Officer. This report may be customized by pressing the design pushbutton on the report speedbar. Then manipulate the report as required. Press the lightning bolt pushbutton when done.
editing the CUSTOM report. To leave the LEVRAS Report menu press the HOME pushbutton to return to the Main Menu.

3.12 Tutorial for Archive Data Processing. The data entry procedures for this form are similar to the Customer Information form. Ensure that the social security number (SSN field) matches the Customer Information form data. Scroll box fields, which operate similar to other data entry form scroll box fields are: Vehicle Year, Vehicle Make, Vehicle Color, License Plate State, and Decal Color. Archive queries are similar to the active database Customer queries discussed in Section 3.2 with the exception of the selection of Archive.qbe file for these queries. To go back to the Main Menu press HOME.
IV. LEVRAS CODES

4.1 State and Country Codes. The following codes are used in the fields that require a state/country within the respective INFORMATION forms are listed below:

<table>
<thead>
<tr>
<th>Code</th>
<th>State/Country</th>
<th>Code</th>
<th>State/Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>AL</td>
<td>Alabama</td>
<td>MO</td>
<td>Missouri</td>
</tr>
<tr>
<td>AK</td>
<td>Alaska</td>
<td>MS</td>
<td>Mississippi</td>
</tr>
<tr>
<td>AR</td>
<td>Arkansas</td>
<td>MT</td>
<td>Montana</td>
</tr>
<tr>
<td>AS</td>
<td>American Samoa</td>
<td>NC</td>
<td>North Carolina</td>
</tr>
<tr>
<td>AZ</td>
<td>Arizona</td>
<td>ND</td>
<td>North Dakota</td>
</tr>
<tr>
<td>BG</td>
<td>Belgium</td>
<td>NE</td>
<td>Nebraska</td>
</tr>
<tr>
<td>CA</td>
<td>California</td>
<td>NH</td>
<td>New Hampshire</td>
</tr>
<tr>
<td>CN</td>
<td>Canada</td>
<td>NJ</td>
<td>New Jersey</td>
</tr>
<tr>
<td>CO</td>
<td>Colorado</td>
<td>NL</td>
<td>Netherlands</td>
</tr>
<tr>
<td>CT</td>
<td>Connecticut</td>
<td>NM</td>
<td>New Mexico</td>
</tr>
<tr>
<td>CZ</td>
<td>Canal Zone</td>
<td>NV</td>
<td>Nevada</td>
</tr>
<tr>
<td>DC</td>
<td>District of Columbia</td>
<td>NW</td>
<td>Norway</td>
</tr>
<tr>
<td>DE</td>
<td>Delaware</td>
<td>NY</td>
<td>New York</td>
</tr>
<tr>
<td>DM</td>
<td>Denmark</td>
<td>OH</td>
<td>Ohio</td>
</tr>
<tr>
<td>FL</td>
<td>Florida</td>
<td>OK</td>
<td>Oklahoma</td>
</tr>
<tr>
<td>FR</td>
<td>France</td>
<td>OR</td>
<td>Oregon</td>
</tr>
<tr>
<td>GA</td>
<td>Georgia</td>
<td>PA</td>
<td>Pennsylvania</td>
</tr>
<tr>
<td>GM</td>
<td>Germany</td>
<td>PR</td>
<td>Puerto Rico</td>
</tr>
<tr>
<td>GR</td>
<td>Greece</td>
<td>PT</td>
<td>Portugal</td>
</tr>
<tr>
<td>GU</td>
<td>Guam</td>
<td>RI</td>
<td>Rhode Island</td>
</tr>
<tr>
<td>HI</td>
<td>Hawaii</td>
<td>SC</td>
<td>South Carolina</td>
</tr>
<tr>
<td>IA</td>
<td>Iowa</td>
<td>SD</td>
<td>South Dakota</td>
</tr>
<tr>
<td>IC</td>
<td>Iceland</td>
<td>TN</td>
<td>Tennessee</td>
</tr>
<tr>
<td>ID</td>
<td>Idaho</td>
<td>TK</td>
<td>Turkey</td>
</tr>
<tr>
<td>IL</td>
<td>Illinois</td>
<td>TT</td>
<td>Trust Territories</td>
</tr>
<tr>
<td>IN</td>
<td>Indiana</td>
<td>TX</td>
<td>Texas</td>
</tr>
<tr>
<td>IT</td>
<td>Italy</td>
<td>VI</td>
<td>Virgin Islands</td>
</tr>
<tr>
<td>KY</td>
<td>Kentucky</td>
<td>UK</td>
<td>United Kingdom</td>
</tr>
<tr>
<td>KS</td>
<td>Kansas</td>
<td>UT</td>
<td>Utah</td>
</tr>
<tr>
<td>LA</td>
<td>Louisiana</td>
<td>VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>LX</td>
<td>Luxembourg</td>
<td>VT</td>
<td>Vermont</td>
</tr>
<tr>
<td>MA</td>
<td>Massachusetts</td>
<td>WA</td>
<td>Washington</td>
</tr>
<tr>
<td>MD</td>
<td>Maryland</td>
<td>WI</td>
<td>Wisconsin</td>
</tr>
<tr>
<td>ME</td>
<td>Maine</td>
<td>WV</td>
<td>West Virginia</td>
</tr>
<tr>
<td>MI</td>
<td>Michigan</td>
<td>WY</td>
<td>Wyoming</td>
</tr>
<tr>
<td>MN</td>
<td>Minnesota</td>
<td>XX</td>
<td>Others</td>
</tr>
</tbody>
</table>
4.2 Decal Color Codes. The following decal colors distinguish between the categories of a customer's job related positional status and should be entered in the "Decal Color" field on the respective INFORMATION forms as listed below:

- Blue: Officer
- Black: Commercial
- Green: Civilian
- Red: Enlisted
- White: Temporary

4.3 Employee Type Codes. Employee types are used in the Employee Type field on the respective INFORMATION forms. Fill-in the appropriate Employee Type by entering in one of the following codes:

- CO: company
- ACDU: active duty military
- MILRES: military reservist
- RETMIL: retired military
- CIVGOV: civilian government
- COMMERCIAL: commercial
- MILDEP: military dependent
- DISVET: disabled veteran

4.4 Grade Codes. Select one alphanumeric code which determines a customer's grade (civilian) on the respective INFORMATION forms as listed below:

- AS1 - AS7  Admin Support
- E1 - E9   Enlisted
- GM13 - GM15 Merit Pay System
- GS1 - GS18 General Schedule
- NA1 - NA15 Non-supervisory
- NL1 - NL15 Leader
- PS1 - PS7  Patron Service
- SES1 - SES9 Senior Executive Service
- WD1 - WD19 Production
- WG1 - WG15 Wage Grade

184
WL1 - WL15 Leader
WP1 - WP78 Printing
WS1 - WS19 Supervisor
WT1 - WT12 Apprentice
WT13 - WT18 Shop Trainee
YV - YV3506 Summer Employee
YW - YW3506 Winter Employee
ZZ Other

4.5 Rank Codes. Select one alphanumeric code which determines a customer's rank (military) on the respective INFORMATION forms as listed below:

E1 - E-9 Enlisted Rank
W1 - W5 Warrant Officer Rank
O1 - O10 Officer Rank

4.6 Ticket Violation Codes. The following ticket violation codes are used on the respective INFORMATION forms. Place the appropriate two digit code in the Violation Code field in lieu of the "Meaning" phrases:

<table>
<thead>
<tr>
<th>CODE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Illegal Parking</td>
</tr>
<tr>
<td>02</td>
<td>Speeding</td>
</tr>
<tr>
<td>03</td>
<td>Expired/mutilated DECAL</td>
</tr>
<tr>
<td>04</td>
<td>Improper equipment</td>
</tr>
<tr>
<td>05</td>
<td>Blocking traffic</td>
</tr>
<tr>
<td>06</td>
<td>No inspection sticker; rejected sticker; expired sticker</td>
</tr>
<tr>
<td>07</td>
<td>Blocked railroad tracks (5 ft. fm tracks)</td>
</tr>
<tr>
<td>08</td>
<td>Expired state license</td>
</tr>
<tr>
<td>09</td>
<td>Reckless driving</td>
</tr>
<tr>
<td>10</td>
<td>Hit and run</td>
</tr>
<tr>
<td>11</td>
<td>Abandoned vehicle</td>
</tr>
<tr>
<td>12</td>
<td>Failure to obey yield sign</td>
</tr>
<tr>
<td>13</td>
<td>Failure to keep right</td>
</tr>
<tr>
<td>14</td>
<td>Improper turn</td>
</tr>
<tr>
<td>15</td>
<td>Failure to obey traffic light</td>
</tr>
<tr>
<td>16</td>
<td>Failure to obey posted sign</td>
</tr>
</tbody>
</table>

185
<table>
<thead>
<tr>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>Illegal traveling or being in a restricted area</td>
</tr>
<tr>
<td>18</td>
<td>Traveling through a &quot;NO THOROUGHFARE&quot;</td>
</tr>
<tr>
<td>19</td>
<td>Illegal U-Turn</td>
</tr>
<tr>
<td>20</td>
<td>Failure to have vehicle under control</td>
</tr>
<tr>
<td>21</td>
<td>Driving while intoxicated (alcohol or drugs)</td>
</tr>
<tr>
<td>22</td>
<td>No vehicle registration</td>
</tr>
<tr>
<td>23</td>
<td>Child left unattended in vehicle</td>
</tr>
<tr>
<td>24</td>
<td>Failure to obey stop sign</td>
</tr>
<tr>
<td>25</td>
<td>Failure to obey traffic officer's signal</td>
</tr>
<tr>
<td>26</td>
<td>Drag racing</td>
</tr>
<tr>
<td>27</td>
<td>Following too close</td>
</tr>
<tr>
<td>28</td>
<td>Illegal use of license plates</td>
</tr>
<tr>
<td>29</td>
<td>Improper passing</td>
</tr>
<tr>
<td>30</td>
<td>Improper backing</td>
</tr>
<tr>
<td>31</td>
<td>Failure to obey aircraft warning lights</td>
</tr>
<tr>
<td>32</td>
<td>Improper driving</td>
</tr>
<tr>
<td>33</td>
<td>Leaving the scene of an accident</td>
</tr>
<tr>
<td>34</td>
<td>Failure to give proper signal</td>
</tr>
<tr>
<td>35</td>
<td>Operating vehicle on-base during a suspension</td>
</tr>
<tr>
<td>36</td>
<td>Failure to yield to pedestrians in a crosswalk</td>
</tr>
<tr>
<td>37</td>
<td>Impeding the flow of traffic</td>
</tr>
<tr>
<td>38</td>
<td>Improper use of traffic lanes</td>
</tr>
<tr>
<td>39</td>
<td>Failure to yield to emergency vehicle</td>
</tr>
<tr>
<td>40</td>
<td>Unsafe vehicle</td>
</tr>
<tr>
<td>41</td>
<td>Leaving vehicle unattended</td>
</tr>
<tr>
<td>42</td>
<td>Allowing unlicensed person to operate vehicle</td>
</tr>
<tr>
<td>43</td>
<td>Illegal use of visitor's pass</td>
</tr>
<tr>
<td>44</td>
<td>High flagging</td>
</tr>
<tr>
<td>45</td>
<td>Operating motorcycle without helmet or safety glasses</td>
</tr>
<tr>
<td>46</td>
<td>Improper towing of vehicle</td>
</tr>
<tr>
<td>47</td>
<td>Operating vehicle without license plates</td>
</tr>
<tr>
<td>48</td>
<td>Operating loaded truck w/o material properly secured</td>
</tr>
<tr>
<td>49</td>
<td>Using vehicle in commission of crime</td>
</tr>
<tr>
<td>50</td>
<td>Refusal to allow breath/blood test</td>
</tr>
<tr>
<td>51</td>
<td>No insurance</td>
</tr>
<tr>
<td>52</td>
<td>Unnecessary noise</td>
</tr>
<tr>
<td>53</td>
<td>Driving on revoked license</td>
</tr>
<tr>
<td>54</td>
<td>Not having a valid operating license</td>
</tr>
<tr>
<td>55</td>
<td>Operating vehicle underage (16 years old)</td>
</tr>
<tr>
<td>56</td>
<td>Operating vehicle without lights in the dark</td>
</tr>
<tr>
<td>57</td>
<td>Littering</td>
</tr>
</tbody>
</table>

186
CODE MEANING

58 Overloading vehicle
59 Illegal display of base decal
60 Illegal display of state license plates
61 No base pass or decal
62 Violating conditions of restricted privilege
63 Manslaughter/Negligent homicide by op of vehicle
64 Incompetent to op vehicle when physically impaired
65 Unauth use of vehicle belonging to another (nofelony)
66 Mandatory recovation is req upon conviction
67 Offense in another state; suspension or recovation would have been required if occurring on base
68 Attempting to elude police officer
69 Violation of curfew laws
70 Owner permitting another to operate vehicle when physically impaired
71 Driving vehicle while impaired (alcohol level more than .05% and less than .10%)
72 Failure to stop for school bus or school crossing signal
73 Failure to yield (no official sign involved)
74 Driver involved in an accident deemed responsible (add to specific offense)
75 Operating motorcycle without helmet chinstrap fastened
76 Operating motorcycle without headlights and/or taillights
77 Operating motorcycle on sidewalk, lawn, seeded areas or other areas
78 Perjury or making false affidavit or statement under oath
79 Selling or disposing of vehicle with Naval Base Sticker
80 Illegal use of decal issued to another person
81 Permitting an unlawful or fraudulent use of an official's driver's license
82 Failure to submit a written accident report when required by regulation and/or law
83 Failure to maintain current registration record
84 Repair of vehicle (non-emergency) on roadways, streets, parking areas/spaces
85 Failure to register vehicle with police department
86 Picking up or discharging passengers in other than designated areas
<table>
<thead>
<tr>
<th>CODE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>87</td>
<td>Infraction of trfc codes or regulations not provided for</td>
</tr>
<tr>
<td>88</td>
<td>Lending vehicle without giving borrower written permission and/or registration</td>
</tr>
<tr>
<td>89</td>
<td>Driving another's vehicle without written permission or registration</td>
</tr>
<tr>
<td>90</td>
<td>Carpool violations</td>
</tr>
<tr>
<td>91</td>
<td>Use of subterfuge to gain selective parking in carpool program</td>
</tr>
<tr>
<td>92</td>
<td>Unauthorized use of decal issued for military purpose</td>
</tr>
<tr>
<td>93</td>
<td>Violation of U.S. criminal codes</td>
</tr>
<tr>
<td>94</td>
<td>Controlled substance</td>
</tr>
<tr>
<td>95</td>
<td>Weapons violation</td>
</tr>
<tr>
<td>96</td>
<td>Selling vehicle and not removing decal</td>
</tr>
<tr>
<td>97</td>
<td>Jaywalking</td>
</tr>
<tr>
<td>98</td>
<td>For future use</td>
</tr>
<tr>
<td>99</td>
<td>For future use</td>
</tr>
</tbody>
</table>
V. USER'S MANUAL REFERENCES

For further Paradox information, the designers of LEVRAS recommend the following publications:


LIST OF REFERENCES


BIBLIOGRAPHY


INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Cameron Station
   Alexandria, Virginia 22304-6145

2. Library, Code 052
   Naval Postgraduate School
   Monterey, California 93943-5101

3. CDR William B. Short, Code SM/SH
   Department of Systems Management
   Naval Postgraduate School
   Monterey, California 93943-5002

4. Shu Liao, Code SM/LC
   Department of Systems Management
   Naval Postgraduate School
   Monterey, California 93943-5002

5. Renee A. Lightcap
   Herrmann Hall, Office E-204
   Management Information Systems
   Naval Postgraduate School
   Monterey, California 93943

6. Gregg Caughran, Code 44
   Security Department, Building 200
   Naval Postgraduate School
   Monterey, California 93943

7. CDR Mark S. Nault
   10 Mervine Street
   Monterey, California 93940

8. LT Henry M. McGibbon
   379D Bergin Drive
   Monterey, California 93940