AD-A243 191

Technical Document 2174
August 1991

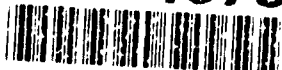# Information Management Engineering (IME) Laboratory: A Database Development Facility

Dr. Marion G. Ceruti
Richard A. Auclair

91-16797

91 11 29 04

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152-5000

J. D. FONTANA, CAPT, USN
Commander

R. T. SHEARER, Acting
Technical Director

RBT

CONTENTS

CONTENTS (continued)

1.0 INTRODUCTION

The Information Management Engineering (IME) Laboratory Facility is located at the Naval Ocean Systems Center (NOSC), Building 600, Room 257W, in San Diego, CA. Development of the IME Menu System, including its associated software, was initiated at NOSC by government personnel in the Advanced Command, Control, Communications, and Intelligence (C³I) Systems Branch and their contractors. The IME Menu System is a collection of software tools for building, maintaining, verifying, and analyzing various Department of Defense database systems. For ease of access, these tools have been made available from one central source, the IME Menu System.

Documentation of release 1.0, entitled "Information Management Engineering Facility (IME)" was completed in July of 1988 but remains unpublished. This present document supersedes it, because numerous improvements have been made. The present document reflects further development of the system to respond to evolving requirements of the various projects for which the IME lab was the primary database support facility. These projects include the following: the OSIS Baseline Upgrade (OBU); the Maritime Defense Zone (MDZ); the Operations Support Group Prototype (OSGP); the Fleet Command Center Battle Management Program (FCCBMP) for CINCPACFLT at Pearl Harbor, HI; the CINCPACAF Integrated Decision Support System (CIDSS) testing for CINCPACAF at Hickam Air Force Base, HI; the Mobile Systems Tactical Data Facility (MSTDF); and the Operations Support System (OSS).

This document is a guide for the database administrator and programmer to use for accessing the IME Menu System. Because of the multitude of similar, yet distinct programming projects, common user files had to be consolidated under one application menu. The IME Menu System provides easy access to project database screens and utilities required for program development.

2.0 SYSTEM DESCRIPTION

2.1 HARDWARE

The IME lab is equipped with a Digital Equipment Corporation (DEC) MicroVAX 3500; four RA81 disk drives, each with a 456-MB capacity; a Kennedy Model-9610 nine-track tape drive; a System Industries Model-SI-59 8-mm tape drive; and a Dataproducts Model-TB-1000/80482 line printer. The system console consists of a DEC VT340 video monitor and keyboard. Seven VT220 terminals and one MacIntosh II access the MicroVAX, which also is connected through a DEC DELNI device to a SUN SPARCstation 330 for file transfer.

2.2 SOFTWARE

2.2.1 Overview

This section briefly describes each software tool accessible through the IME Menu System and provides instructions on how to use the IME menu.

The IME Menu System was designed to provide a user-friendly interface for a set of database tools. Menu selections include options such as data loading facilities and other database applications, tape and disk management, database referential integrity checking, percent fill function, data dictionary, and classified document and magnetic tape inventory functions. The inventory function is based on the Library Inventory Tracking and Reporting (LITAR) System (reference 1), but has been modified extensively. Reports can be generated to produce data dictionary hard copies and inventory lists.

2.2.2 Preconditions for Use

Users of the IME Menu System need a background in database management, particularly in Standard Query Language (SQL), the Oracle Relational Database Management System (references 2, 3), and a working knowledge of the VAX/VMS Operating System. Another requirement for users is to set the VMS default directory to a default that grants them write privileges for many of the functions to work. The Oracle tables associated with the specific databases under development must be available to enable all of the reports and Oracle screen forms to work properly for both data updates and retrievals.

2.2.3 Menu Function and Directory Structure

The IME Menu System is a set of tools designed for building, maintaining, and retrieving databases, tapes, and related documents. It allows users easy access to any of the utilities available in the system. The VAX/VMS IME_MENU directory and sub-

directories contain all source code necessary to implement the IME Menu System. When new code is added to the system, it is placed into—and referenced from—a subdirectory of the main IME_MENU directory.

2.2.3.1 How to Use the IME Menu System. To access the system from the VAX/ VMS prompt, type IME <return>. A logical name defining IME does not need to be in your LOGIN.COM file.

The main menu, entitled IME Facility Management Main Menu, will appear on the screen. The main menu currently has 10 categories, including a NOTICE option. Eight of the 10 categories provide access to additional menus that are submenus containing the available applications options. The options in this menu are as follows:

    0. Exit to VMS

    1. Table of Contents

    2. Database applications

    3. Database build

    4. Database forms

    5. Database tools

    6. Inventory management

    7. Tape and disk tools

    8. Experimental test

    9. Timed reminder message

    10. Notice(s) to users

    <return> Return to VMS

Each menu features options to return to the previous menu or to VMS. Enter <return> to exit from the current menu to the previous menu, and 0 to return to VMS. In the main menu, entering either 0 or <return> will cause an exit to VMS, because the main menu is called directly from VMS. The other exception to the exit rule occurs when accessing an IAP screen developed from the Oracle SQL*FORMS Utility. In this case, exit to the previous menu is accomplished by entering the PF4 key (reference 2).

Menu options also can be selected using the Table of Contents from the main menu. To access this option, type 1<return>. This function displays a list of all

4

options available from the main menu and their submenus (table 1). Each submenu can be chosen directly from this screen by entering the appropriate number, without returning to the main menu. Some submenus are listed here in an abbreviated form to save space on the screen.

Table 1. Display of Table of Contents menu.

TABLE_OF_CONTENTS

(2) DATABASE APPS.	(3) DATABASE BUILD	(4) DATABASE FORMS	(5) DATABASE TOOLS
CIDB	OB fill	NWTDB Screens	Percent Fill Report
NWTDB	DMA fill	IME Screens	Free Blks in partitions
IME		FCCBMP Screens	Table Description Reports
OSGP		CIDSS Screens	Table Record Count Report
FCCBMP			Data Dictionary Report
OBU			Partition Reports
LAB257			Referential Integrity Rep.
			Duplicate Keys Report
			Unload Oracle to Flat File
			Load Oracle from Flat File

(6) INVENTORY MANAGEMENT	(7) TAPE/DISK TOOLS	(8) EXPERIMENTAL TEST
Inventory Screen	Tape to Disk	Domain Checking
Inventory Reports	Split It Up	
Export Inventory Table	Read WWMCCS Tape	

Type desired menu number, or 0 for Exit to VMS, or <return> only for main menu.

5

2.2.3.2 Database Applications. To access this option, type 2<return>.

This option, though it provides only Oracle screens, is called applications (vs. forms), because these screens are not limited to allowing only data entry. They were designed to identify errors and to verify data, thus facilitating the quality control of database fill.

This option allows you to access Oracle Interactive Application Process (IAP) screens for the following databases: CIDSS Integrated Database (CIDB), IME, FCCBMP, Naval Warfare Tactical Database (NWTDB), OSGP, OBU, and Lab 257. IAP is an Oracle utility that allows you manually to insert, delete, update, and query data in the database through a unique screen (or screens) developed for each database. These screens were created with the Oracle SQL*FORMS package. The IAP screens include those for the data dictionary for each database and fill plans. Presently, Lab 257 is the only one of these applications for which an associated Oracle database is on line.

2.2.3.2.1 Data Dictionary Screens. These IAP screens allow you to update, insert, delete, and select information from the Data Dictionary.

Because a Data Dictionary is a document containing descriptions of all the attributes and their definitions in the existing database, it is a helpful tool in understanding the purpose and usage of the tables. The Data Dictionary contains the null status of each attribute in the database. For example, because fill for primary keys is mandatory, these attributes are designated as "not null." This is necessary to comply with C. J. Date's entity integrity rule that states that no attribute participating in the primary key of a base relation (a table) is allowed to accept null values (reference 4). Thus, all primary key attributes must be filled with correct values during manual record insertion, prior to exiting the screen.

2.2.3.2.2 Fill Plan Screens. The Fill Plan is an important database fill tool. Before each database can be filled from the intermediate tables, an analysis must be performed to identify each intermediate data field that will fill each final data field. The Fill Plan IAP screens show the relationship between each attribute in the intermediate tables and the corresponding source at the attribute level.

2.2.3.3 Database Build. To access this option, type 3<return>.

This option displays a submenu for accessing the database building tools for filling tables in the Defense Mapping Agency Aerospace Center (DMAAC) and Order of Battle (OB) databases. These databases have been or will be used to fill the integrated databases for programs such as FCCBMP, CIDSS, and OSS. The options in this submenu are as follows:

1. OB Fill

2. DMA Fill

<return> – Exit Menu

0 – Exit to VMS

The OB fill and DMA fill menu selections provide access to submenus that pertain to that particular database.

2.2.3.4 Database Forms. To access this option, type 4<return>.

Choosing this option leads to another submenu, entitled "Forms Menu," that allows you to access the following IAP screens for several databases.

0. Exit to VMS

1. NWTDB Screens

2. IME Screens

3. FCCBMP Screens

4. CIDSS Screens

<return> – Return to Main Menu

The forms submenus are described below.

2.2.3.4.1 NWTDB Screens. NWTDB Screens allow you to perform queries on NWTDB tables only. Fifty-two screens are accessible through the NWTDB submenu, "NWTDB RELATION SCREENS." These screens are divided into five categories: equipment, surface ships, ship and submarine classes, fixed-wing aircraft, and helicopters.

2.2.3.4.2 IME Screens. Choosing the "IME Screen" option calls another submenu, entitled "IME RELATION SCREENS." This submenu allows you to access IAP screens in six categories of tables: equipment (EQUIP), ships (SHIP), squadrons (SQUAD), submarines (SUB), aircraft (AIR), and GENERAL tables, for a total of 51 tables.

2.2.3.4.3 FCCBMP Screens. The FCCBMP screens allow you to access a submenu of predefined database queries. Options include queries on world port index, major equipment, equipment identification codes, and fuel consumption.

2.2.3.4.4 CIDSS Screens. The CIDSS Oracle Forms Directory allows you to access a submenu of CIDSS's IAP screens in seven categories. These categories consist of

military fixed-wing aircraft, helicopters, missiles, airfields, sortie, Jane's aircraft, and commercial aircraft.

2.2.3.5 Database Tools. To access the Database Tools Menu, type 5<return>.

The options in this menu are as follows:

0. Exit to VMS

1. Write Percent Fill Report

2. Display Free Blocks in Partitions

3. Write Table Descriptions Report

4. Write Table Record Count Reports

5. Write Data Dictionary Reports

6. Write Partition Reports

7. Write Referential Integrity Report

8. Write Duplicate Keys Report

9. Delete Duplicate Keys

10. Transfer Data from Oracle Table to Flat File

11. Transfer Data from Flat File to Oracle Table.

<return> – return to Main Menu

You may select from these functions to obtain information about the databases. These tools are command procedures written in the Digital Command Language (DCL) that call other procedures and functions written in Oracle SQL or FORTRAN. Therefore, these menu selections are sometimes called options or command procedures.

These tools are helpful for fill plans and integrity checking. Most tools can be used with any database accessible to a user. Each tool will prompt you to enter the database name and password, if needed.

All submenu selections generate disk .LOG files. Selections that produce reports write an additional .OUT file on the disk containing the results for the report requested. All reports are written to disk files that are named by a concatenation of the database names and the report names. An example of a percent fill report filename is NWTDB_PERCENT_FILL.OUT, where NWTDB is the database. You can print or delete each report or .LOG file outside of the IME Menu System, in the appropriate VMS file directory.

2.2.3.5.1 Write Percent Fill Reports. This command procedure creates a report showing the percentage of attributes that are not null for each table of the selected database. To select a particular database, type the username and password at the appropriate prompts. The percent fill software will submit the report job to the batch queue and will notify the user's terminal when the job is completed. The percent fill report and .LOG file will be placed in the [IME_MENU.DB_TOOLS.PERCENT_FILL] directory.

2.2.3.5.2 Display Free Blocks in Partitions. This command procedure creates a screen display showing the number of free VAX blocks remaining in the currently defined partitions. Unlike the report functions that write results to a disk, the Free Blocks function spools the report directly to your terminal screen. When the display is complete, you will be prompted to press <return> to return to the Database Tools Menu.

2.2.3.5.3 Write Table Descriptions Report. This command procedure will create a report containing alphabetically ordered descriptions of all tables in a specified database. This tool will submit the table-descriptions job to the batch queue and will notify the your terminal when the job is completed.

The report will be placed in the [IME_MENU.DB_TOOLS.DESCRIPTIONS] directory.

2.2.3.5.4 Write Table Record Count Report. This command procedure will list the tables and the number of records in each table for a specified database. The program will prompt you for the username and password of the desired database. The database table count option will submit the report to the batch queue and will notify your terminal when the job is completed. The report will be placed in the [IME_MENU.DB_ TOOLS.TABLE_COUNT] directory.

2.2.3.5.5 Write Data Dictionary Reports. This command procedure provides menu access to 13 reports used to compare data in the Oracle Table Definitions (OTBLDEFN) with data in a LAB257 Data Dictionary. A help facility is available with this option. This part of the user interface works differently from most of the IME Menu System because it was developed separately.

As an aid to the user, selected options become highlighted when chosen. This is a way of selecting multiple options for consecutive execution from a single menu presentation. Most menus in the IME Menu System allow only one option to be selected each time the menu is displayed.

2.2.3.5.6 Write Partition Reports. This option produces a report on the Oracle partitions. The report includes information on table names, table users, Oracle storage, and Oracle extents. A help facility describing each menu choice is available with the

9

Partition Report option. As with Data Dictionary Reports, selected options become highlighted when chosen.

2.2.3.5.7 Write Referential Integrity Report. C. J. Date defines referential integrity as follows: "If base relation R2 includes a foreign key FK matching the primary ke PK of some base relation R1, then every value of FK in R2 must either (a) be equal to the value of PK in some tuple (record) of R1 or (b) be wholly null..." (reference 4). Therefore, the Referential Integrity Report function requires a special table, REFERENTIAL_INTEGRITY_RULES, to be created and/or filled for each database. The REFERENTIAL_INTEGRITY_RULES table contains the following attributes, each of which is 30 characters long: DATABASE, REFERRING_RELATION, REFER-RING_ATTRIBUTE, REFERENCE_ATTRIBUTE, REFERENCE_RELATION. The last four attributes correspond to R2, FK, R1, and PK as defined above.

This option lists data values that are possible sources of error in the referential integrity of a given database. It was designed as an aid for data analysts for improving the quality of databases.

The function will work only if the REFERENTAIL_INTEGRITY_RULES table in the Oracle user database <username>/<password> has been filled for the specified database referential integrity check. This can be accomplished by the database administrator. To select a particular database, enter the username and password at the appropriate prompts.

A referential integrity check is performed by executing queries such as the following:

> Select ATTRIBUTE_FK from TABLE_R2
> minus
> Select ATTRIBUTE_PK from TABLE_R1

Here, ATTRIBUTE_FK could have the same name as ATTRIBUTE_PK. If no records are selected, no referential integrity errors are detected in the fill for ATTRIBUTE_FK in TABLE_R2, because this is a subset of the fill for ATTRIBUTE_PK in TABLE_R1. If records from TABLE_R2 are returned, these are errors detected because no values exist in the fill for ATTRIBUTE_PK in TABLE_R1 that correspond to values detected in ATTRIBUTE_FK in TABLE_R2.

2.2.3.5.8 Write Duplicate Keys Report. This option produces a report listing duplicate keys detected in the entire specified database. To select a particular database, enter the username and password at the prompt.

2.2.3.5.9 Delete Duplicate Keys. This option deletes duplicate keys in individual tables only, or in the entire specified database, depending upon your input. Duplicate

keys are undesirable in a database because the key is assumed to be a unique identifier of the record (reference 4). To select a particular database, enter the username and password at the prompt.

2.2.3.5.10 Transfer Data from Oracle Table to Flat File. This option unloads data from an Oracle table into a flat file. In this specific context, a flat file is an ASCII file containing records of equal length. In each record, the fields (or attributes) are fixed in length and in the same order. Fields are contiguous because no delimiters such as commas are inserted between them.

The flat file will be in a format required by the Oracle Host Language Interface. For example, all numeric fields are 32 characters long and are left-justified. Character fields contain the same number of characters as defined in the Oracle table definition and are left-justified. Date fields must have the standard Oracle Date format.

2.2.3.5.11 Transfer Data from Flat File to Oracle Table. This option loads data from a flat file into an Oracle table "similar" to the table from which the data were unloaded. Accurate results are obtained only if either the flat file was unloaded from Oracle previously by selecting the Unload Oracle Table to Flat File option described above, or if the flat file is in a format identical to that of a flat file produced with the Unload option. The attribute order, length, and data types must be identical to the original, but table and attribute names may be different.

This tool was not designed to load just any flat file from any database. However, this may be possible if the flat file has the correct attribute lengths and if the table to receive the data has been designed carefully to match the attribute ordering, length, and data types.

An Oracle table must be created to receive the data before using this tool. (The user is assumed to know how to create an Oracle table.) The receiving table must contain the same number and length of attributes in the same order as the table from which the data were loaded originally. The loaded data will be appended to the data that already exist in the receiving table. The order of execution of the steps in this procedure is listed below:

1. Enter file name
2. Enter username/password
3. Enter Oracle table name
4. After the table is loaded, you will be prompted to press the <return> key to continue. Control will be returned to the Database Tools Menu.

2.2.3.6 Inventory Management. To access this option, type 6<return>.

The Inventory Management functions available as options in this submenu are as follows:

0. Exit to VMS

1. Inventory Screen

2. Inventory Reports

3. Export Inventory Table

<return> Return to Main Menu

The operation of this function is essentially the same as that of the LITAR System (reference 1). The IME_INVENTORY database contains information regarding classified documents and tapes. The Inventory Management functions are described below.

2.2.3.6.1 Inventory Screen. This Oracle IAP screen allows you to update, insert, delete, and query information about the classified tape and document inventory.

2.2.3.6.2 Inventory Reports. Selecting this option, which is similar to the reporting function of the LITAR System, allows the user to obtain printed or screen listings from the document and tape inventory system. This option reports the results of a query to your terminal and creates an ASCII file containing the results in your default directory. The LITAR System has been modified to run on a VAX 3600. Changes include the substitution of the terms "inventory" or "tape/doc" for "tape"; and "257 area" for "263 rack."

2.2.3.6.3 Export Inventory Table. Submenu selection 3 produces a display entitled "Exporting Inventory Table" and automatically exports the inventory table, thus creating an export file, IME_INVENTORY.EXP, in the [IME_MENU.INVENTORY] directory. This database export is quick and easy to perform each time a change is made to the inventory database.

2.2.3.7 Tape and Disk Tools. To access this option, type 7<return>.

Because electronic data transfer is more efficient and less prone to error than manual data entry, most data from outside agencies are received on magnetic tapes. The tape and disk utilities help you to load the data from a tape onto a disk for subsequent transfer into a database.

Selecting this menu option displays the following submenu choice, Tape and Disk Tools, which allows you to access these tape and disk FORTRAN utilities:

0. Exit to VMS

1. Tape To Disk

2. Split It Up

3. Read WWMCCS Tape

<return> Return to Main Menu

2.2.3.7.1 Tape To Disk. This is a utility, written in FORTRAN, designed to copy a tape file to a disk file. It inputs data from magnetic tape files of various formats and writes the data to a disk file in user-specified format. Data can be input subsequently into a relational database management system.

The program provides an internal submenu to facilitate option selection and to assist you by prompting for all inputs when required. Before copying the data to the disk, perform a trial run to determine the correct tape format, unless you are absolutely sure of the format and record size of the data contained on the tape. A trial run with 80 characters or fewer (as the disk output record size) will show all of the data in a tape block wrapped around on the screen without any truncation. A trial record size greater than 80 characters is accepted, but each record will be truncated beyond 80 to allow the beginning of each record to be viewed easily on the screen. When the actual copy is performed, the output record size will be as you specified with no truncation.

When initiated, the program will attempt to determine the tape block size. You are expected to determine by observation whether each tape block contains a single record or multiple records.

If the tape block contains a single record, the record size will equal the tape block size. If the tape block contains multiple records, you must determine the record size. This can be accomplished easily by observing where data appear to repeat and by counting the number of characters between repeats. In either case, you must enter the determined record size when requesting the output DISK FILE record size.

If the tape data are in BCD or EBCDIC code, the tape-to-disk program provides for conversion to ASCII code. Best results are obtained by performing a trial with no conversion requested. Most tapes have one or more header blocks that the program will display in the following formats: unconverted, converted to ASCII from EBCDIC, and converted to ASCII from BCD. Observing the header blocks can help you to decide whether a conversion is needed. BCD and EBCDIC conversions to ASCII are similar, the main difference being that EBCDIC allows lowercase letters.

To abort a trial, type ABORT <return> at a programmed reading break. In that case, the tape will rewind and the program will prompt you to press the return key, returning the user to the Tape-to-Disk Menu.

The program allows you to skip files. Files remaining after being skipped also may be read or transferred to a disk file. A double end-of-file mark on the tape is assumed to be the logical end of the tape.

13

2.2.3.7.2 Split It Up. This program takes as input from a disk file, records of length greater than 80 characters, divides them into records of length 80 characters or less, and outputs them wrapped around to a second disk file. This allows you to view or edit files with large records that otherwise may not be as readily accessible through the VAX editor.

The program prompts you for the input file and for the name of the output file. The output file will be written in your current directory.

2.2.3.7.3 Read WWMCCS Tape. Read WWMCCS is a tape utility used to convert Honeywell (BCD)-formatted tapes from the World Wide Military Command and Control System (WWMCCS) to a VAX ASCII disk file.

This program will read any Honeywell standard BCD tape and write out the logical (converted to ASCII) records to a disk file. The program was written in FORTRAN to operate under VAX/VMS. To use the program, load a tape onto a drive and run the READ_WWMCCS.COM file automatically under the IME Menu System, which precludes the need to type in the MOUNT commands for mounting the tape. The tape will be mounted automatically by running the READ_WWMCCS.COM DCL command program from the IME Menu System.

2.2.3.8 Experimental Test. To access this option, type 8<return>.

This section is dedicated to software under development. It currently contains a domain checking system.

2.2.3.9 Timed Reminder Messages. To access this option, type 9<return>.

This option alerts personnel in the IME lab of meetings, deliverables, deadlines, etc. Alerts are initiated by submitting a dummy batch job with completion notification requested and also by sending email messages to a user list. The name of the dummy batch job is the short message.

2.2.3.10 Notice(s) to Users. To access this option, type 10<return>.

This option displays text files concerning general announcements, including those to inform personnel of changes to the IME system that affect the usage of one or more menu items. Text files to be displayed are created by the system manager or the database administrator.

2.2.4 Modularity of IME_MENU

If the IME_MENU and subdirectories need to be moved as a block and installed into a different directory or even a different computer system, all of the source code can be backed up starting at the IME_MENU directory. Although the programs are unclassified, be careful when generating backup files from directories that also contain

classified data files. Magnetic tapes or disks containing classified data must be marked and handled appropriately for that particular level of classification. The following DCL example captures files of all classifications residing in the specified directory:

```
$ SET DEFAULT DEVICE:[MAIN_DIR.IME_MENU]
$ BACKUP [...]*.*;* DEVICE:IME_MENU.BCK
```

The system is backed up into the new location by the following VMS command:

```
$ SET DEFAULT <New directory location>
$ BACKUP DEVICE:IME_MENU.BCK DEVICE:[...] *.*;*
```

After the BACKUP is loaded into the new location, two items need to be entered into the VAX SYLOGIN.COM file, or modified as follows:

```
$ IME :==@DEVICE:[MAIN_DIR.IME_MENU]IME_MENU.COM
$ IME_ROOT_WS :== DEVICE:[MAIN_DIR.IME_MENU.
```

where DEVICE represents the logical name for the disk, and MAIN_DIR is a higher-level directory, if any, that will contain the IME_MENU directory and its subdirectories. IME_MENU itself could be a top-level directory. One additional item is required in the VAX SYSTARTUP.COM file (or SYSTARTUP_V5.COM file for VMS 5.0 or higher):

```
$ DEFINE/SYS/TRANS=CONCEALED IME_ROOT-
        DEVICE:[MAIN_DIR.IME_MENU.]
```

These definitions and symbols make referring to a directory under the IME_MENU System convenient and generic. For example, if IME_ROOT is defined as above, IME_ROOT:[DB_TOOLS] is equivalent to DEVICE:[MAIN_DIR.IME_MENU.DB_TOOLS], and ' 'IME_ROOT_WS'DB_TOOLS] placed in a VAX DCL write-to-screen statement would be equivalent to DEVICE:[MAIN_DIR.IME_MENU.DB_TOOLS]. Here, ' 'IME_ROOT_WS' is a VMS symbol.

The logical name, IME_ROOT, is used extensively within the IME_MENU System to specify the location of source code and output files. The symbol name, IME_ROOT_WS, is used within the IME_MENU System for printing output file locations to the screen. The use of IME_ROOT and IME_ROOT_WS facilitates moving the entire IME_MENU System to new locations, because changing only these two items automatically will change all of the locations in the system where they appear as references.

3.0 REFERENCES

1. M. G. Ceruti, R. A. Auclair, J. P. Schill, and K. Yarnell. 1991. "On-Line Magnetic Tape Library Inventory Tracking and Reporting (LITAR) System." NOSC TD 2057 (March). Naval Ocean Systems Center, San Diego, CA.

2. Oracle Corporation. 1987. "*ORACLE SQL*Plus User's Guide*," Version 2.0, part number 3201-V2.0, Oracle Corp., Beimont, CA.

3. Oracle Corporation. 1987. "*ORACLE SQL*Forms Operator's Guide*," Version 2.0, part number 3301-V2.0, Oracle Corp., Belmont, CA.

4. C. J. Date. 1986. "*Relational Database: Selected Writings*," pp. 25 – 26. Addison-Wesley Publishing Company, Menlo Park, CA.

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1 AGENCY USE ONLY (Leave blank)	2 REPORT DATE August 1991	3 REPORT TYPE AND DATES COVERED Final FY 85 — FY 91

4 TITLE AND SUBTITLE INFORMATION MANAGEMENT ENGINEERING (IME) LABORATORY: A DATABASE DEVELOPMENT FACILITY	5 FUNDING NUMBERS Proj No. AS64 CD40 CC37 CD52 CC62 CD53 CD28 SX15
6 AUTHOR(S) Dr. Marion G. Ceruti and Richard A. Auclair	

7 PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Ocean Systems Center San Diego, CA 92152–5000	8 PERFORMING ORGANIZATION REPORT NUMBER TD 2174

9 SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Naval Security Group Command Washington, DC 20393–5210 Space and Naval Warfare Systems Command Washington, DC 20363–5100	10 SPONSORING/MONITORING AGENCY REPORT NUMBER

11 SUPPLEMENTARY NOTES

12a DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	12b DISTRIBUTION CODE

13 ABSTRACT (Maximum 200 words)

An Information Management Engineering (IME) database development laboratory is described that features a VAX/Macintosh hardware suite and a menu-driven user interface. The software system was created for—and is used by—database developers and database administrators who work with the Digital Equipment Corporation Virtual Memory Operating System (VMS) and the Oracle Database Management System. It was designed to accomplish a variety of database-related tasks, including data input from magnetic tape, disk file management, database fill, data entry, data dictionary report generation, data quality analysis and validation, data reference inventory management, and user communications. This system was used to develop and enhance several command and control databases for the following projects: the OSIS Baseline Upgrade (OBU); the Maritime Defense Zone (MDZ); the Operations Support Group Prototype (OSGP); the Fleet Command Center Battle Management Program (FCCBMP) for CINCPACFLT at Pearl Harbor, HI; CINCPACAF Integrated Decision Support System (CIDSS) testing for CINCPACAF at Hickam Air Force Base, HI; the Mobile Systems Tactical Data Facility (MSTDF); and the Operations Support System (OSS).

14 SUBJECT TERMS data dictionary database fill database administration foreign key information management inventory management magnetic tape primary key referential integrity relational database	15 NUMBER OF PAGES 23
	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 SAME AS REPORT

21a. NAME OF RESPONSIBLE INDIVIDUAL	21b TELEPHONE *(include Area Code)*	21c OFFICE SYMBOL
Dr. Marion G. Ceruti	(619) 553–4068	Code 423

21a. NAME OF RESPONSIBLE INDIVIDUAL | 21b TELEPHONE *(include Area Code)* |

INITIAL DISTRIBUTION