ornl

ORNL/TM-13155

RECEIVED
JUN 2 1 1996
OSTI

OAK RIDGE
NATIONAL
LABORATORY

LOCKHEED MARTIN

Testing and Transition: The Final Days of System Development

L. F. Truett
J. P. Rollow

MANAGED AND OPERATED BY
LOCKHEED MARTIN ENERGY RESEARCH CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

ORNL-27 (8-95)

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P. O. Box 62, Oak Ridge, TN 37831; prices available from (423) 576-8401, FTS 626-8401.

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA 22161.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government of any agency thereof.

ORNL/TM-13155

Energy Division

TESTING AND TRANSITION: THE FINAL DAYS OF SYSTEM DEVELOPMENT

L. F. Truett J. P. Rollow

May 1996

Prepared for the
Product Management Office
Office of the Deputy Chief of Staff for Information Management
MILITARY TRAFFIC MANAGEMENT COMMAND
Falls Church, Virginia 22041-5050
under
Interagency Agreement DOE No. 1405-1351-A1

Prepared by
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORPORATION
for the
U.S. DEPARTMENT OF ENERGY
under contract No. DE-AC05-96OR22464

CONTENTS

ABSTRACT v

1. INTRODUCTION 1
   1.1 Background 1
   1.2 Overview of This Task 1
   1.3 Acronyms and Initialisms 2
   1.4 References 3

2. TESTING 5
   2.1 Schedules for Testing 5
   2.2 Functional Testing and Test Conditional Reports (TCRs) 6
      2.2.1 Purpose of TCRs 6
      2.2.2 Format of TCRs 7
      2.2.3 TCR Usage 11
      2.2.4 Technical (MAISRC) TCRs 12
   2.3 Data for Testing Functional TCRs 13
   2.4 Test Order 15
   2.5 Problem Reports and Code Uploads 15
   2.6 Performance Testing 16
   2.7 Results of Testing 17

3. TRANSITION 19
   3.1 Implementation 19
   3.2 Transition to a Maintenance Contractor 19

4. LESSONS LEARNED 21

5. CONCLUSIONS 23

APPENDIX A A-1
APPENDIX B B-1
APPENDIX C C-1
APPENDIX D D-1
APPENDIX E E-1
APPENDIX F F-1
APPENDIX G G-1

LIST OF FIGURES

2.1 Front page of TCR format used for testing 8
2.2 Reverse side of a TCR 9
2.3 Sample of the first page of an actual TCR 10

LIST OF TABLES

2.1 Formal testing schedule of the WPS Regional ICDB 6

ABSTRACT

As part of existing tasking, the Military Traffic Management Command (MTMC) requested that Oak Ridge National Laboratory (ORNL) assist with writing test scenarios for the formal testing of the Worldwide Port System (WPS) Regional Integrated Cargo Database (ICDB). In collaboration with MTMC, ORNL wrote almost 600 Test Conditional Reports (TCRs), which were used to test specific functional processes. In addition, ORNL prepared the overall test order, managed tracking of problem reports and code uploads, and interacted with the testers throughout the entire testing period.

Because ORNL provided analysis and design for ICDB and because ORNL was intimately involved in development, it was unusual to be so deeply involved in system testing. This document reports on the testing process and on lessons learned.

ORNL also assisted MTMC during the initial implementation period and during transition from a developmental to a production system. A maintenance contractor was hired for ICDB, and ORNL assisted this contractor in preparing for system maintenance responsibilities. This document reports on this transition period also.

1. INTRODUCTION

1.1 BACKGROUND

The Worldwide Port System (WPS) Regional Integrated Cargo Database (ICDB) was designed and developed by personnel from the Military Traffic Management Command (MTMC) and personnel from Oak Ridge National Laboratory (ORNL). ICDB provides data integration and worldwide tracking capability for cargo that passes through common-user ocean cargo ports. ICDB is a data repository for the WPS terminal-level system, is a primary source for queries and cargo traffic reports, receives data from and provides data to other MTMC and non-MTMC systems, provides capabilities for processing booking records and cargo shipment documentation, distributes manifests, and provides regional managers with capabilities for managing cargo and monitoring cargo information. ICDB provides an integrated database for efficient and reliable data management, data manipulation, and data distribution.

Conforming to the requirements of Autostrad-2000 (A-2000), which implies an open systems design, ICDB resides on a Unix platform with Oracle7 as the relational database management system (RDBMS). The basic ICDB architectural design consists of a single worldwide central database server and multiple regional processing hubs. In the Continental United States (CONUS), there is a hub on each coast. ORNL was involved only in CONUS development. Port sites are connected to either a west or east coast hub. The primary source of data to ICDB is from the WPS terminal-level system, located at the ports.

System development started in early 1993. The WPS terminal-level system for CONUS functionality was developed by WPS personnel located at MTMC Western Area, Oakland, California. ICDB developers included ORNL personnel and MTMC personnel located at MTMC Information Management, Eastern Area (IME). Formal testing began in early 1995. Testing was completed, the system was accepted, and implementation began on the West Coast in August of 1995.

1.2 OVERVIEW OF THIS TASK

The WPS terminal-level system was implemented Outside the United States (OCONUS) prior to development of the CONUS WPS and ICDB. The OCONUS regional database for the

WPS terminal-level system is a data repository with ad hoc query capabilities, but the CONUS regional database (ICDB) is a sophisticated database with extensive state-of-the-art capabilities. Although the OCONUS WPS had successfully completed Major Army Information System Review Council (MAISRC) testing, the ICDB was significantly different; thus, it was required to undergo MAISRC review as a new system.

Tasking was assigned to ORNL to assist with writing test scenarios for the formal MAISRC testing of the functional requirements of ICDB. ORNL was provided a format for Test Conditional Reports (TCRs) and was assigned responsibility for writing TCRs to test the functionality of the system. Additional TCRs were written to test other portions (e.g., background processes, performance) of ICDB. Some of these TCRs were prepared by ORNL and some by the MAISRC test representative.

ORNL was asked to assist the maintenance contractor during the transition from a development mode to a working system. Because of the demands of development and testing, there was very little time to meet with and train the maintenance contractor. ORNL prepared a booklet of training materials and hosted two training sessions toward the end of ORNL's involvement in the project.

1.3 ACRONYMS AND INITIALISMS

A-2000	Autostrad-2000
CONUS	Continental United States
DBA	Database Administration
ECP	Engineering Change Proposal
FD	Functional Description
ICDB	Integrated Cargo Database
IME	Information Management, Eastern Area
MAISRC	Major Army Information System Review Committee
MILSTAMP	Military Standard Transportation and Movement Procedures
MTMC	Military Traffic Management Command
OCONUS	Outside the Continental United States
ORNL	Oak Ridge National Laboratory
PMO	Product Management Office
PR	Problem Report
RDBMS	Relational Database Management System
SA	System Administration
SDT	System Development Test
SQT	System Qualification Test

TCN	Transportation Control Number
TCR	Test Conditional Report
WPS	Worldwide Port System

1.4 REFERENCES

Oak Ridge National Laboratory. Database Specifications for the Worldwide Port System (WPS) Regional Integrated Database (ICDB). UNCLASSIFIED. ORNL/TM-12725. March 1994.

Oak Ridge National Laboratory. Architectural Analysis for the Worldwide Port System (WPS) Regional Integrated Database (ICDB). UNCLASSIFIED. ORNL/TM-12357. April 1993.

Oak Ridge National Laboratory. Functional Description for the Worldwide Port System (WPS) Regional Integrated Database (ICDB). UNCLASSIFIED. DRAFT November 1992, revised January 1993, revised November 1993; ORNL/TM-13118. November 1995.

Oak Ridge National Laboratory. Standards and Conventions for the Worldwide Port System (WPS) Regional Integrated Cargo Database (ICDB). UNCLASSIFIED. ORNL/TM-12799. February 1995.

Oak Ridge National Laboratory. System/Subsystem Specifications for the Worldwide Port System (WPS) Regional Integrated Cargo Database (ICDB). UNCLASSIFIED. FINAL DRAFT, September 1995; ORNL/TM-13119. November 1995.

2. TESTING

Testing occurred throughout the development period at individual development sites for specific applications and for integration of the code among WPS, IME, and ORNL developers. These tests were limited by the state of development and the architectural platform at any particular point in time. In December 1993, a platform of production hardware was made available. In 1994, ORNL designed and developed a multi-threaded approach to database and code maintenance on the production architecture (the central server and two hubs). The three database threads, which are defined below, provided excellent opportunities for testing ICDB functionality and for managing changes to the developmental code.

- DEMO. This database was used by the MTMC ICDB Product Management Office (PMO) to demonstrate ICDB menus and screens to end users.
- TEST. This database was for testing of new code.
- PROD. After code and/or schema changes were tested (in the TEST thread), they were uploaded to PROD.

In late October until early December of 1994, PMO conducted intensive testing of the system as far as it could be tested at that time. This testing was limited by the fact that code to load data from ICDB into WPS had not been developed at that point in time, and, therefore, this major component could not be tested. During PMO testing, PMO wrote problem reports, and the responsibility for correction was assigned to either IME or ORNL. Most of the problems, though not all, were corrected prior to the end of 1994.

The primary purpose of this document is to report on the more official testing which occurred toward the end of the project (throughout the first nine months of 1995). These tests were monitored by MAISRC testers. The required tests were established by MAISRC and included the System Development Test (SDT) and System Qualification Test (SQT).

2.1 SCHEDULES FOR TESTING

Tests included semiformal (pre-SDT), formal (SDT-I and SDT-II), and very formal (SQT-I and SQT-II) formats. Table 2.1 gives the schedule for these tests.

Table 2.1. Formal testing schedule of the WPS Regional ICDB

Test	Dates	Place
pre-SDT	Jan 9-Jan 20, 1995	Oakland, California
SDT-I	Jan 30-Mar 3, 1995	Oakland, California
SQT-I	Apr 24-Jun 1, 1995	Oakland, California
SDT-II	Jul 10-Jul 21, 1995	Falls Church, Virginia
SQT-II	Aug 7-Aug 25, 1995	Oakland, California

When testing began with WPS, representatives of the ICDB development sites (ORNL, IME), PMO, and WPS met at Western Area, along with testers who were selected from the pool of future system users. WPS personnel made all arrangements for the test sites and for testing personnel.

The testing scenarios (TCRs) and the test plan established during SDT were used by MAISRC personnel for the SQT. In addition, MAISRC personnel developed a separate set of TCRs for testing performance — both capacity levels and timing. The TCRs for evaluating performance (see also Section 2.2.4) were used in SQT-I, SDT-II, and SQT-II only.

2.2 FUNCTIONAL TESTING AND TEST CONDITIONAL REPORTS (TCRs)

ORNL was a primary developer of ICDB and expected that an independent testing organization would be employed to prepare the testing procedures. MAISRC personnel were in charge of the testing process, developed the TCR format, and established the testing procedures as discussed in the following sections. However, ORNL was assigned the responsibilities of writing individual functional TCRs and developing the test order. WPS and PMO also wrote functional TCRs.

2.2.1 Purposes of TCRs

The purpose of the functional TCRs was to test specific requirements. Because the final test acceptance was determined by MAISRC personnel, a very clearly defined set of tests which

identified and tested each functional process was necessary. In addition, different testers were used in the formal tests (e.g., SDT, SQT); thus, the testers were unfamiliar with the system and had to learn how the system worked at the same time that they were trying to test it. The TCRs were a mechanism to provide sufficient control to both train and test at the same time.

2.2.2 Format of TCRs

The TCR format provided a very structured scenario for testing. The TCRs were written using a predetermined format established by MAISRC. A diskette with the established format using a word-processing software was provided to ORNL. The format is shown in Figure 2.1 (the front of a TCR form) and Figure 2.2 (the reverse side of a TCR). The front side of a sample TCR, as given to a tester, is shown in Figure 2.3. The reverse side was completed in hand writing by the tester during testing and retained by MAISRC personnel for a permanent record of test results.

In the TCR header block (see Figure 2.1), the following conventions were followed. "TCR No." was a shorthand identifier. TCRs initiated by ORNL used a numbering scheme based on the Functional Description (FD) requirements in the WPS FD, Appendix A, combined with the numbering scheme in the ICDB FD. TCRs initiated by WPS used a numbering scheme based on the cargo status as export or import. The "Date Prepared" and "Preparer's Name" were completed by the preparer. The next entry, "ECP No.," which stood for Engineering Change Proposal, was left blank during writing of the TCR. An ECP number was assigned if a routine problem was identified which did not require changes during the SQT. The ECP number allowed the problem to be tracked for future correction. The entry for "Input Medium" was always *screen* for the functional TCRs.

In the body of the TCR, the "Test Purpose" stated the functionality to be tested; "FD Reference" was a reference to the pages in the WPS FD, Appendix A, which provided a very brief explanation of the primary functions of ICDB; "Related Programs" listed particular preceding or following TCRs, technical TCRs (written by ORNL) to load required data, or technical TCRs (written by ORNL) to test background processes to ensure that the functionality was successful; "Condition to be Evaluated" was similar to "Test Purpose," although more concise.

TEST CONDITION REQUIREMENT (TCR)
WORLDWIDE PORT SYSTEM (WPS)

TCR NO.:	DATE PREPARED:	PREPARER'S NAME:	ECP NO.:	INPUT MEDIUM: Screen

TEST PURPOSE:

FD REFERENCE:

RELATED PROGRAMS:

CONDITION TO BE EVALUATED:

TEST PROCEDURES:

EXPECTED OUTPUT:

GENERATED TRANSACTIONS/REPORTS:

DATABASE CHANGES EXPECTED:

Figure 2.1. Front page of TCR format used for testing.

TEST CONDITION REQUIREMENT (TCR)
WORLDWIDE PORT SYSTEM (WPS)

TCR NO.: RUN NO.:	DATE PREPARED:	PREPARER'S NAME:	ECP NO.:	INPUT MEDIUM: Screen

TEST RESULTS:

SUMMARY OF TEST RESULTS/OUTPUT GENERATED:

TEST CONTROL INFORMATION

DATE TESTED: _____ _____
PRINTED NAME/ORGANIZATION

_____
SIGNATURE

( ) ACCEPTED ( ) NOT ACCEPTED

EXPLAIN:

( ) PR _____ ( ) ECP GENERATED _____

DATE TCR ACCEPTED: _____

ACCEPTANCE SIGNATURE: _____

Figure 2.2. Reverse side of a TCR.

ADSM-25-L35-F01-WPS-HPC-PT
30 June 1993

TEST CONDITION REQUIREMENT (TCR)
WORLDWIDE PORT SYSTEM (WPS)

TCR NO.: G4-112a2	DATE PREPARED: June 1995	PREPARER'S NAME: Truett	ECP NO.:	INPUT MEDIUM: Screen

TEST PURPOSE: To ensure that manifest records reported on a hard-copy report of manifests failing quality validation routines can be corrected

FD REFERENCE: WPS FD Appendix A, pp. 36-37

RELATED PROGRAMS: G4-tech-112a1, WPS TCR to ensure that the corrected record is transmitted to the appropriate WPS site.

CONDITION TO BE EVALUATED: Error correction

TEST PROCEDURES:
Note: Tester will be supplied with a list of userids and passwords for specific roles (Attachment 1).

1. Log on as puser2.
2. Select Regional Processes (5); press Enter.
3. Select Print Reject Listing (4); press Enter.
4. Select Print Manifest Reject Listing (2); press Enter. When the pop-up window appears for identifying a printer, press F9 to obtain a list of available printers. Use the UP/DOWN arrow keys to identify the printer you desire to use; press Enter three times (to select the printer name, to continue, and to return to the application screen).
5. Obtain the printout of rejected records. Note the reject reason on the printout for rejected records. You will be correcting various records.
6. Use the UP/DOWN arrow keys to select Return to Regional Processes Menu (4); press Enter.
7. Select Edit Import Manifests (2); press Enter.
8. The cursor is in the Control NBR field. Enter the Control NBR from the listing printed earlier. Enter the Control NBR that matches TCN = MREJ00XXXXXXXXXKK2. Press F8. Notes the reject reason on the listing invalid POD). Tab to POD and change the POD to 3DK. Press F10 to submit the revised POD and its associated trailer data.
9. Press S-F10 to exit the screen.
10. Repeat Steps 7-9 for TCN = MREJ00XXXXXXXXXZX. Change POD to 3DK. Press F10 to commit.
11. Repeat Steps 7-9 for TCN = MREJ10BBXXXXXXXXX. Change POD to 3DK. Commit.
12. Repeat Steps 7-9 for TCN = MREJ10XXXXXXXXXKK2. Change PK to ZB. Commit.
13. Repeat Steps 7-9 for TCN = MREJ20 XXXXXXKK2. Change the TCN to MREJ20FINXXXXXXKK2. Commit.
14. Repeat Steps 7-9 for TCN = MREJ20BBXXXXXXXXX. Change PK to BX. Commit.
10. Exit.

EXPECTED OUTPUT: None
GENERATED TRANSACTIONS/REPORTS: None
DATABASE CHANGES EXPECTED: Corrected record is sent to WPS

Figure 2.3. Sample of the first page of an actual TCR.

In the next block, the "Test Procedures" were clearly written, step by step. Each procedure began with a verb, and no action (either a choice of data to be entered or a key to be pressed) was left to random choice. Every act of the tester was very carefully orchestrated in advance.

The last section on the front page of the TCR listed the "Expected Output," "Generated Transactions/Reports," and "Database Changes Expected." The tester had the responsibility to ensure that the expected results documented in this section of the TCR matched the actual results demonstrated during the test. If data was passed to WPS, a follow-on TCR checked the conditions at the terminal.

2.2.3 TCR Usage

Each TCR tested a separate specific function of ICDB. A listing of all functional TCRs is provided in Appendix A. These TCRs were sorted into "test sets" which are discussed in Section 2.4, "Test Plan." All functional TCRs were written as if the user were accessing the functional menu and application screens of the ICDB user interface. This method was ideal for querying or for producing reports, and it was also ideal for the WPS terminal-level functionality of data entry. However, it was not adequate for testing any background processes (e.g., distributing code tables to the hubs and the WPS sites, resending data, and creating and sending data files to other organizations). These processes were initiated through screen applications and might appear to be successful to the functional user, but they actually occurred in the background and could not be checked through the user interface. To check these processes, the tester needed to be able to log into Oracle's SQL*Plus and create SQL*Plus queries to check local and remote Oracle tables. Therefore, ORNL wrote additional technical TCRs which were required for performing and/or testing the background processes. Actions required by the technical TCRs written by ORNL were performed by an ICDB or a WPS System Administrator (SA) or Database Administrator (DBA). A listing of these TCRs is provided in Appendix B.

Because certain kinds of records could enter the system either at the hubs or at WPS terminals, TCRs had to be written to test data flows from all data entry points. Therefore, the TCRs could be either "initiating" or "follow-on" depending on where data entered the system. In addition, records could enter the system either via on-line-build screens (manual data entry) or via flat files that were then processed through validation routines (batch data entry).

For manual data entry, testers were given precise instructions on the data to be entered. For batch data, the flat files were also prepared in advance in order to maintain control. Printouts of the flat file layout of records in Military Standard Transportation and Movement Procedures (MILSTAMP) 80-column format were provided to testers. (Additional information about the data used for testing is supplied in Section 2.3.)

If data entered the system at one of the ICDB hubs, the ORNL TCRs were written from a point of view which tested data initiating at the hubs. WPS wrote follow-on TCRs to test the receipt of that data at the appropriate terminal. The TCRs normally instructed the user to view a specific record on both ICDB and WPS screens and in the flat file printout (for batch-loaded records). The TCRs for these tests are listed in Appendix A, Test Sets 1-1 through 1-24. When the data displayed at ICDB did not equal the data displayed at WPS or when the data at either site did not agree with the flat file layout, a problem was noted. (Additional information about problem reporting is given in Section 2.5.)

If data entered the system at the WPS sites (either as a new record or as status changes), WPS personnel wrote the initiating TCRs, and the follow-on TCRs (written by ORNL and PMO) tested the receipt of that data at the central database. The primary purpose of the follow-on TCR was to ensure that data displayed on the WPS terminal-level screen could also be viewed correctly in ICDB, if required. (Some fields were specific to terminal-level processing and did not appear in ICDB.) A list of these TCRs is given in Appendix A, Test Sets 2-1 through 2-10.

In addition, some ORNL TCRs tested functionality that did not require WPS follow-on testing. And some WPS TCRs did not require follow-on testing at ICDB. These TCRs are also included in the Test Sets listed in Appendix A.

2.2.4 Technical (MAISRC) TCRs

A totally separate set of technical TCRs was written and tested by the MAISRC test administrator using the same format as was used for the functional TCRs. These technical TCRs tested the following processes: (1) communication processing and the automatic switching from network access to modem access when a normal communication line was unavailable, (2) performance times for queries based on number of users logged in, (3) performance times for transferring specified numbers of records based on the total

number of records in the database, (4) performance times for transferring specified numbers of records from several sites almost simultaneously, (5) ability of users to log in via a 1-800 number, (6) data extracts production of and acceptance by appropriate receiving agencies, (7) security, and (8) backup and restore capabilities.

ORNL reviewed the technical TCRs prepared by MAISRC. During the technical testing, ORNL also supported the testing agency (MAISRC) and responded to issues and problems. The technical testing was very significant because it tested major components of the system (e.g., communications, performance, and automated data extraction). This report, however, emphasizes the functional testing because ORNL did not write the TCRs for the technical tasks. The results of the performance testing are addressed in Section 2.6.

2.3 DATA FOR TESTING FUNCTIONAL TCRs

Obtaining a large volume of data for testing was a major problem. Although MTMC, through the Area Commands, provided certain test data sets, these did not cover all conditions. The PMO established a plan for constructing data sets that would ensure a sample of data within each commodity category (e.g., hazardous, personal property, ammunition). MTMC transportation managers use a Transportation Control Number (TCN) to identify cargo shipments. The PMO plan was to alter this 17-position TCN identifier in such a way that it would provide the following information for test purposes: port indicator (for those ports being used in the test), type of cargo (i.e., containerized or breakbulk), commodity type, container number or piece count, and content number. The remaining positions in the revised TCN were usually filled with "X"'s except in special cases (e.g., personal property shipments and unit moves). This TCN structure was provided to the testers as an attachment to the TCRs and was cross-referenced in the TCRs. The PMO also determined that a certain number of each type of record would be required. This information was also provided to the testers. The TCN structure for the test data sets and the list of TCNs in each category are provided in Appendix C.

The original intention of PMO was to obtain sufficient "real" data for each of the appropriate categories and then change the TCNs to match the format shown in Appendix C. However, because of problems on the IME hardware and because cargo in every record category was not shipped during the time frame for obtaining records, actual data for all cargo types was not obtained. Because the TCRs were written in such a precise manner, the exact TCNs were

needed to complete the TCRs. As time grew closer to actual testing and ORNL could not complete the TCRs, ORNL personnel constructed the test data sets from the available data.

ORNL selected files of "real" data, either cargo shipment records or manifests. Because of the desire to have example data from multiple cargo "types," the raw data files were occasionally modified from the "real" data sets. For example, some cargo shipment records were used to make manifest records and some manifest records were modified as cargo shipment records. In addition, the preliminary booking data was adapted from cargo shipment records to ensure that the records would match.

These data sets were used as input to the system (to ensure that certain records with specific TCNs would be available to the testers). The flat files were also printed out, and this hard-copy of the record layouts was provided to the testers. This flat-file record layout was lengthy and is not reproduced in this document. Other data sets were also formulated by ORNL as needed. A list of all of the types of data that was reproduced is given in Appendix D.

Because of the way the data was derived, sometimes the data was "inaccurate." For example, the flat file of data for a cargo shipment might contain a field that was actually a manifest field (e.g., stow location). Although the system was intelligent enough to know during the processing of this file not to insert data into a stow location field in the database for a shipment record of cargo that had not actually been stowed, the appearance of this piece of data on the printout of the flat file was confusing to the testers. Therefore, the testers were instructed to ask the Test Coordinator for assistance if they were unsure about the data in a field.

Sometimes the testers were confused because of system functionality with which they were unfamiliar. For example, ICDB was designed to re-compute the pieces, weight, and cube for a van based on a sum of all of the content records. Therefore, the total summed value for the pieces, weight, and cube of a van's contents on the flat file printout might not agree with the data contained in the database and displayed on the screen. The tester was invited to test the computational powers of the system by summing the pieces, weight, and cube of each content record for the van.

The TCRs that tested the on-line build functionality of the system also had certain limitations, primarily based on the tester's data entry abilities. (For example, the tester performing data entry might type "abcdefg" rather than "mnopqrs" for a van number. When a tester using a

separate TCR queried for van number "mnopqrs" using the instructions on the TCR, that van number would not exist because it had been mistyped.) Caveats were built into user instructions for this occurrence. Testers were reminded that records built via the on-line build screens did not always conform to the instructions provided in the TCR. Therefore, if there was a discrepancy, the tester was instructed to check with the Test Coordinator.

It is important to note that the data sets described above were developed and used ONLY for the formal tests described in this document. "Real" data was used throughout ICDB development for site and integration testing. It is only when using actual real-world data that real-world problems can be identified.

2.4 TEST ORDER

Prior to formal testing, PMO designed the format for a form that could be used to provide order to the testing. ORNL was assigned the responsibility to further develop this test order to ensure that all functionality would be tested. Because some of the intelligence in ICDB is to match records based on certain criteria (e.g., cargo shipment records to booking records), the data had to be loaded in a certain order to test all conditions. The sequence in which data entered the database affected how the information was incorporated into the cargo record, and a careful control of the sequencing was required to test this logic. ORNL developed a two-phase approach as the most efficient method. This allowed testing of the WPS terminal-level data entry to proceed at the same time that data was being entered at the Hubs. This dual process of adding data at two sources concurrently was possible because different record identifiers were used. It helped to speed up the data entry part of the test. The test order (a blank copy is reproduced in Appendix E) allowed the test coordinator to maintain control of the processes and to ensure that they occurred in the appropriate sequence. The test sets listed in Appendices B and A, respectively, are referenced in the second and third columns of the test order (Appendix E). Each day, primary contacts from ORNL, IME, PMO, and WPS met with the MAISRC tester to update the test order.

2.5 PROBLEM REPORTS AND CODE UPLOADS

During SQT, when testers identified problems, they initiated a problem report (see Appendix F, Figure F.1). At the end of each day, personnel from WPS, ORNL, PMO, and IME, met

with the MAISRC coordinator to determine if a problem report was really a problem or a misunderstanding on the tester's part. If a problem actually existed, a priority was assigned,[1] and the responsibility for correction was determined. Then the problem was assigned a Problem Report (PR) number and was logged in as needing correction. If the responsibility for correction fell to WPS, usually the problem report was not entered into the log because the functionality of the terminal-level system was outside of the purview of testing for the WPS Regional ICDB.

A procedure was established for uploading code to fix the problems (see Appendix F, Figure F.2). After a correction had been made by the responsible developing agency, ORNL performed all uploads to ICDB, whether the problem was fixed by ORNL, by IME, or by WPS. The terminal-level system had already been approved through the formal MAISRC testing process. Therefore, if the problem was strictly a terminal-level issue, then WPS controlled code uploads to the WPS test system and the problem report was not a formal part of the testing documentation.

Once code had been revised and the fix tested on the development system, ORNL would request a code upload. The MAISRC test coordinator would approve the upload, and the TCR which was being used for testing when the problem report had been written would be used for retesting. In some cases an entire test set of functional TCRs would need to be retested in order to retest one TCR. Based on the results of retesting, the problem log would be updated.

2.6 PERFORMANCE TESTING

Performance testing was of two types: inquiry response time (including reports production) and data transfers/loads with a full database (9 million rows). These performance requirements are given in Appendix G.

Prior to testing with a fully loaded database, all performances easily met required timeframes. Based on ORNL experience, performance tuning is best done with a large amount of data;

---

[1]These priorities were "Emergency" conditions, requiring 24-hour turnaround time for corrections; "Urgent" conditions, requiring correction as quickly as possible; and "Routine" changes, which did not require changes prior to implementation. These labels were assigned according to DA Pamphlet 25-6, "Configuration Management for Automated Information Systems."

therefore, ORNL had done no performance tuning with the limited data available. Because of other commitments and the prolonged functional testing schedule, MTMC was unable to obtain 9 million rows of cargo data early enough to do performance testing in advance. Therefore, no performance testing was conducted until the end of the first SQT.

Thus, because the system had never been tested with a significant amount of data, ORNL expected that performance tuning would be required, and, when a full database was loaded, performance was identified as a significant problem for both (1) inquiries and (2) data transfers.

The inquiries were performed from the user interface screens (see Appendix G, first, second, fourth, and sixth activities listed). When the system could not meet performance requirements for inquiries, ORNL personnel analyzed the problems and added some additional indexes. This solved the majority of the performance problems for inquiries.

When the system could not meet performance requirements for data transfer processes (see Appendix G, seventh through tenth activities listed), it was a much larger problem and more difficult to solve. The transfer and conversion processes were complicated; usually a single transfer process called many different scripts and procedures.

ICDB was required to poll each WPS terminal site every five minutes and pull all changed records (up to a maximum of 1000 records). This requirement was tested after the requirements for the user interface inquiries had been passed. The data transfer and loading processes had always been very rapid, until the database contained 9 million rows. The procedures for inserting and/or updating 1000 records into a database of this size totally failed the 5-minute requirement. Analyzing the performance problems (while still testing functional ECPs) and finding and implementing solutions took considerable time during the last week of SQT-I. ORNL could not analyze and tune all of the data transfer procedures. Therefore, SDT and SQT had to be repeated after a one-month interruption for performance tuning.

During this month, MTMC personnel, the Oracle corporation, ORNL, and other MTMC contractors worked to tune database parameters, scripts, and procedures. The month was adequate to identify primary performance problems and determine the best solutions. Also, an upgrade to the next version of Oracle's RDBMS was required to use the Trace utility. Oracle's Trace showed exactly which statements were slowing down performance for known

problems. ORNL was able to rewrite the data transfer procedures in order to significantly improve performance.

During the month allotted for performance tuning and the Oracle upgrade, ORNL also rewrote scripts for performing backups and assisted MTMC in establishing guidelines for ensuring that MTMC personnel responsible for backup and restore procedures had an established protocol for completing these responsibilities.

2.7 RESULTS OF TESTING

After the first SQT, ORNL and PMO spent one month upgrading and tuning the system. Then SDT-II retested functionality and performance to ensure that everything was functioning as expected. SQT-II began shortly after SDT-II and was completed in only half the time that had been spent on SQT-I. The system was accepted and implementation began the next weekend!

3. TRANSITION

There is a charged atmosphere when transitioning a development software system into production. The exercise is no longer a practice; it is no longer in testing; it is being put into use. ORNL was not present at the WPS sites or the ICDB hubs or the central server when the system "went live," but ORNL personnel were available to provide assistance and/or advice when needed. This section is a brief overview of the first few weeks of implementation, during which developmental control of the software and maintenance of the software passed from ORNL to MTMC and the maintenance contractor.

3.1 IMPLEMENTATION

Implementation of the system began on the weekend following the successful conclusion of SQT II. As real-world users began using the system, unforeseen problems related primarily to (1) data and (2) performance tuning occurred. After less than three months of actual operation, most CONUS terminals were transitioned into WPS sites, and the data feeds to/from ICDB continued to grow in number and size. Performance tuning situations continued to show up. Although the performance problems that were identified and fixed prior to SQT-II remained "fixed," as new sites were implemented and the volume of data increased, other performance problems arose. ORNL worked with the PMO as requested to resolve these problems during the transition period.

3.2 TRANSITION TO A MAINTENANCE CONTRACTOR

The ORNL team was gradually phased out as the maintenance contractor was brought on. ORNL prepared a "training" document and hosted maintenance contractor personnel for two visits. The training document covered the major components of the system as well as methods for trouble-shooting and performance tuning. The sections of the training document included the following:

- an overview of how/why the system was designed,
- system administration/database administration functions,
- an overview of the "will to live" and communications,

- data transfers to other organizations,
- data transfers from/to the server, hubs, and sites,
- user interface processes,
- trouble-shooting strategies, and
- performance tuning strategies.

The section on trouble shooting used a philosophy that was very valuable to ORNL during the analysis of problems identified during testing. The process included saving data into temporary tables at various stages during data transfer and conversion. Saving data (even temporarily) becomes a performance issue in a real-world situation, and this strategy is no longer used for analyzing problems.

ICDB is an extremely complicated system, and the transition was not altogether smooth. Telephone conferences were valuable. Also, two subcontractors who had been with the ORNL development team for a lengthy period of time were put on contract by the maintenance contractor during the transition. This decision has proved to be very useful.

Because the schedule for testing, fixing, and retesting the code was so short, ORNL staff had very little time when they could be available to the maintenance contractor. During the spring and summer of 1995, some ORNL staff members were often on the West Coast for formal testing. The remaining staff members were supporting the testing effort by addressing those problems that were written up as emergency or critical problem reports.

During SQT II, the maintenance contractor was still staffing positions for the group that would be maintaining the system. They were also getting hardware and software installed to "imitate" the PMO system. They were still working on this task during the last month that ORNL was on the project, so many of the emergency situations that occurred as new sites were added to the "live" system were handled by ORNL and the PMO. Because of these situations and time constraints, the maintenance contractor had little experience with dealing with system problems before they took over maintaining the system.

4. LESSONS LEARNED

ORNL participated actively in the ICDB formal testing process, both in writing the TCRs and in on-site assistance. As a means to an end, the process was worth the time and effort expended. However, from the point of view of resource utilization (ORNL personnel spent the better part of a year on the West Coast), ORNL generally felt that there might have been a better way to accomplish the same results. It certainly would have been more convenient for East Coast residents to have conducted more of the tests on the East Coast; however, the testing was performed in connection with the terminal system whose developers were located on the West Coast. The following list represents the "lessons learned" on both the testing process and the transition period **from an ORNL perspective.**

* The most important lesson learned is that all testing, including all functionality, integration of code developed at different sites, and technical testing, should have been completed successfully prior to ANY formal, controlled testing. Although each site was responsible for testing the code developed by that site, ORNL was responsible for integrating all modules. Much of the ORNL code written for communications and data transfers could not be tested until development activities at other sites were complete. Thus, it was impossible to test all of the ORNL code in advance. Additional time should have been provided for integration testing.

* In-process reviews of code and functionality were conducted; however, deficiencies were not always corrected in a timely manner. The easy solutions seemed to be accomplished quickly, but the hard problems continued to plague integration testing for much longer than reasonable.

* Less time spent in repetitive testing of the same functions and more time for fixing problems would have been a more efficient use of time. There was hardly time to fix problems before another test (more travel) was scheduled.

* TCRs were not an effective way to test a system because they listed specific steps and used canned data to produce expected results. Nothing was random, and many functional inaccuracies were noticed only when testers stopped using TCRs and just tried to "break" the system.

* The TCRs might have been more effective if someone other than the developers had written them and if they had been prepared earlier. For example, TCRs for some modules were written almost at the last minute because the modules were incomplete or were being revised.

* There were a lot of TCRs written for the functional requirements, but one whole subsystem (SA/DBA) was never tested in this manner. It seems to be rather robust; therefore, perhaps the system personnel who used it did a better job of testing than was done through the TCRs.

* End-to-end testing before all components were operational was a mistake. It cost a tremendous loss of resources (time and money) to fly the testers and the developers who were supporting the testing to the West Coast for lengthy periods of time, especially when they had to wait for fixes to code that had never previously been tested.

* For the developer to provide test data and test scenarios was not wise. PMO determined that TCRs required data sets with certain characteristics and specific TCNs and planned to develop these data sets. Because ORNL did not have the functional knowledge of MTMC data to make all appropriate changes, ORNL's conversion of cargo shipment records to manifest-type records and vice versa caused a lot of unnecessary confusion for the testers.

* Large data sets (9,000,000 database rows) should have been available for performance testing prior to SQT. Performance testing for the first time during SQT caused the failure of the first SQT.

* New requirements were reassigned from IME to ORNL and to WPS just prior to and during SDT and the first SQT. This functionality was not ready for informal integration testing prior to formal testing, and a lot of the functionality failed SQT.

* Testing of data that originated at the hubs was difficult because the WPS process for loading data from the hubs into the terminal system was inconsistent and undependable. This particular process continued to cause problems through SDT-II. The ability of WPS to accept data from the hubs should have been available much earlier than it was, with sufficient time to test and retest this process prior to any formal testing.

* The testing process was one method of convincing the end-users of the system that the system was effective. Additional training was also provided.

* Either the maintenance contractor should have been on board much sooner or ORNL should have remained on active status much later. The many small problems which occurred upon implementation would have been much easier for ORNL personnel to identify and fix than they have been for the maintenance contractor, who was unfamiliar with the system.

5. CONCLUSIONS

Frequently research and development organizations are immersed in the research, but never reach the end of the development stage. The ORNL team worked from initial requirements gathering, through requirements analysis, and through design activities; they assisted with development tasks, integrated the system components (both software and hardware), participated in testing, provided assistance during initial implementation, and provided partial training materials to the maintenance contractors.

The system testing period was a time filled with a great deal of stress brought about by the tight deadlines and the formal testing environment. Some tasks that were assigned to other organizations were completed by ORNL in order to ensure that they were completed. Though having ORNL write the TCRs for testing was unexpected, it provided a valuable experience with this highly-structured testing methodology.

In the real world, both government and industry must be cost-conscious. It is imperative to accomplish as much as possible within as short a time as possible with whatever resources are available. The very formal SQT with structured TCRs that provided documented evidence and were repeatable as often as necessary provided this environment.

The ORNL team derived substantial satisfaction from the success of ICDB. The project ended with goals met.

APPENDIX A

LIST OF FUNCTIONAL TCRs SORTED BY TEST SET

LIST OF TCRs SORTED BY TEST SETS

PHASE 1: RECORDS ORIGINATING AT THE REGIONAL DATABASE AND PASSED TO WPS VIA ONE HUB

Note: Sets 1-1 through 1-9 test processing of booking and ATCMD records from the Regional to WPS sites

Pass/Fail

Set 1-1: Test booking data load

R1-113-BKG-1	Data loads of batch loads of booking data (from METS)
R1-113-BKG-2	Data loads of batch loads of booking data (from METS)
R1-113-BKG-3	Data loads of batch loads of booking data (from METS)

Set 1-2: Test matchup of booking and ATCMD data

R1-113-BKG-4	Matches of batch booking and batch ATCMDs (booking loaded first)
R1-113-BKG-5	Data loads of batch loads of booking data with no matches to ATCMDs

Set 1-3: Build ATCMDs on-line at the Regional

R1-121-exp02	Add export van records with single consignee
R1-121-exp03	Add export van HHG records
R1-121-exp04	Add export van POV records
R1-121-exp05	Add export van ammo records
R1-121-HAZ	Add export van hazardous records
R1-121-exp06	Add export refrigerated van records
R1-121-exp07	Add export van records with three stopoffs
R1-121-exp08	Add export breakbulk records
R1-121-exp10	Add export HHG breakbulk records
R1-121-exp11	Add export POV breakbulk records
R1-121-exp12	Add export ammo breakbulk records
R1-121-exp13	Add export unaccompanied baggage breakbulk records
R1-121-exp14	Add export government vehicle breakbulk records
R1-121-exp14A	Add export outsize breakbulk records
R1-121-exp15	Add export mail breakbulk records

Set 1-4: Remaining TCRs to test load of ATCMD data

R1-113-ATCMD-1	Data loads of batch loads of ATCMD data
R1-113-ATCMD-2	Data loads of batch loads of ATCMD data
R1-113-ATCMD-3	Data loads of batch loads of ATCMD data
R1-113-ATCMD-4	Data loads of batch loads of ATCMD data
R1-113-ATCMD-5	Data loads of batch loads of ATCMD data
R1-113-ATCMD-6	Data loads of batch loads of ATCMD data
R1-113-ATCMD-6A	Data loads of batch loads of ATCMD data
R1-113-ATCMD-7	Data loads of batch loads of ATCMD data
R1-113-ATCMD-8	Data loads of batch loads of ATCMD data
R1-113-ATCMD-9	Data loads of batch loads of ATCMD data
R1-113-ATCMD-12	Data loads of batch loads of ATCMD data
R1-113-ATCMD-13	Data loads of batch loads of ATCMD data
R1-113-ATCMD-14	Data loads of batch loads of ATCMD data
R1-113-ATCMD-15	Data loads of batch loads of ATCMD data

Pass/Fail

R1-113-ATCMD-16	Data loads of batch loads of ATCMD data
R1-113-ATCMD-17	Data loads of batch loads of ATCMD data
R1-113-ATCMD-18	Data loads of batch loads of ATCMD data
R1-113-ATCMD-19	Data loads of batch loads of ATCMD unit data for Army
R1-117	Exploding TCNs received via batch load processes

Set 1-5: Building remaining ATCMDs on-line at Regional

R1-121-exp16	Add export classified breakbulk records
R1-121-exp17	Add export sensitive breakbulk records
R1-121-exp18	Add export controlled breakbulk records
R1-121-exp19	Add export hazardous cargo breakbulk records
R1-121-exp21	Add export empty van records
R1-122-BKG-VAN1	Recall existing container booking data for populating the online build screen
R1-122-BKG-VAN2	Add container booking data to test matchups with batch-loaded booking records
R1-122-refuse	Refusal to allow changes via on-line build to an existing ATCMD record

Set 1-6: Test matchup of booking and ATCMD data

R1-113-BKG-6	Matches of batch-loaded booking data and on-line-built ATCMDs
R1-113-BKG-7	Matches of batch-loaded booking data and on-line-built ATCMDs
R1-113-BKG-8	Matches of batch-loaded booking data and ATCMDs (ATCMD loaded first)

Set 1-7: Resend/divert ATCMDs

T6-118a1	Resend ATCMDs by container nbr
T6-118a3	Resend and Divert ATCMDs by container no.
T6-118b1	Resend ATCMDs by TCN
T6-118b2	Divert ATCMDs by container no.
T6-118b2-ext	Divert van with CONEX
T6-118b3	Resend and Divert ATCMDs by container no.
T6-118-WPS	View Divert to Seattle

Set 1-8: Correct rejected ATCMD record

G4-112a1	ATCMD error correction and resubmission

Set 1-9: Correct invalid TACs

G4-112a3	TAC correction

Set 1-10: View corrections on WPS

G4-112-WPS	View corrections to ATCMDs

Note: Sets 1-22 through 1-24 test processing of import manifest records from the Regional to WPS sites

Pass/Fail

Set 1-22: Test load of import manifest data

R3-21-import-1	Receipt of import manifest data from OCONUS sources
R3-21-import-2	Receipt of import manifest data from OCONUS sources
R3-21-import-3	Receipt of import manifest data from OCONUS sources
R3-21-import-4	Receipt of import manifest data from OCONUS sources
R3-21-import-5	Receipt of import manifest data from OCONUS sources
R3-21-import-6	Receipt of import manifest data from OCONUS sources
R3-21-import-6A	Receipt of import manifest data from OCONUS sources
R3-21-import-7	Receipt of import manifest data from OCONUS sources
R3-21-import-9	Receipt of import manifest data from OCONUS sources
R3-21-import-9A	Receipt of import manifest data from OCONUS sources
R3-21-import-10	Receipt of import manifest data from OCONUS sources
R3-21-import-11	Receipt of import manifest data from OCONUS sources
R3-21-import-12	Receipt of import manifest data from OCONUS sources
R3-21-import-13	Receipt of import manifest data from OCONUS sources
R3-21-import-14	Receipt of import manifest data from OCONUS sources
R3-21-import-15	Receipt of import manifest data from OCONUS sources
R3-21-import-16	Receipt of import manifest data from OCONUS sources
R3-21-import-17	Receipt of import manifest data from OCONUS sources
R3-21-import-18	Receipt of import manifest data from OCONUS sources

Set 1-23: Resend and divert manifests

T2-24a1	Resend entire import manifest by VOYDOC to same port
T2-24a2	Divert entire import manifest by VOYDOC to a changed port
T2-24b1	Resend individual manifested container records by van nbr to same WPS POD
T2-24b2	Divert individual manifested container records by van nbr to changed WPS POD
T2-24c1	Resend individual manifested records by TCN to same WPS POD
T2-24c2	Divert individual manifested records by TCN to a changed WPS POD

Set 1-24: Correct rejected manifest record

G4-112a2	Import manifest error correction and resubmission

Note: Sets 1-50 through 1-59 test processing of general requirements

Set 1-50: Test all intransit visibility requirements

G1-41a1	Retrieval of cargo status based on full TCN
G1-41a2	Retrieval of cargo status based on partial TCN
G1-41b1	Retrieval of cargo records based on container number
G1-41c1	Retrieval of cargo records based on SSN
G1-41d3	Retrieval based on last name

Pass/Fail

G1-41e1	Count of records retrieved
G1-41e2	Count of records retrieved
G1-41e3	Count of records retrieved
G1-42a1	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a2	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a3	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a5	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a7	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a11	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a12	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42a16	Retrieval of cargo records (Mult Ship-Detail/All Cargo Types)
G1-42b1	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b2	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b3	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b4	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b5	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b6	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b7	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b8	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b9	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b10	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42b11	Retrieval of cargo records (Mult Ship-Detail/Hazardous/Explo)
G1-42c1	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c2	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c3	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c4	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c5	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c7	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c8	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42c9	Retrieval of cargo records (Mult Ship-Detail/HHG/POV)
G1-42d1	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d2	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d3	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d4	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d6	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d8	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d9	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42d10	Retrieval of cargo records (Mult Ship-Detail/Unit/Vehicle)
G1-42e1	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-42e2	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-42e3	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-42e5	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-42e6	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-42e7	Retrieval of cargo records (Mult Ship-Detail/Commodity CD)
G1-43a	Retrieval of cargo records (Mult Ship-Summary)
G1-43b	Retrieval of cargo records (Mult Ship-Summary)
G1-43d	Retrieval of cargo records (Mult Ship-Summary)
G1-43e	Retrieval of cargo records (Mult Ship-Summary)
G1-43g	Retrieval of cargo records (Mult Ship-Summary)

Pass/Fail

G1-43h	Retrieval of cargo records (Mult Ship-Summary)
G1-43m	Retrieval of cargo records (Mult Ship-Summary)
G1-43n	Retrieval of cargo records (Mult Ship-Summary)
G1-43o	Retrieval of cargo records (Mult Ship-Summary)
G1-44a1	User's roles to control data access
G1-44a2	User's roles to control data access
G1-44a3	User's roles to control data access
G1-44a4	User's roles to control data access
G1-44a5	User's roles to control data access
G1-44a6	User's roles to control data access
G1-44a7	User's roles to control data access
G1-44a8	User's roles to control data access
G1-46a1	List of values on queryable columns screens.
G1-46a2	List of values on queryable columns screens.
G1-46a3	List of values on queryable columns screens.
G1-46a4	List of values on queryable columns screens.
G1-46a5	List of values on queryable columns screens.
G1-46b1	List of values on response screens
G1-46b2	List of values on response screens
G1-46b3	List of values on response screens
G1-46b4	List of values on response screens
G1-46b5	List of values on response screens
G1-46b6	List of values on response screens
G1-46b7	List of values on response screens
G1-46b8	List of values on response screens
G1-46b9	List of values on response screens

Set 1-51: Resend GTN files; change frequency of transmittal for GTN files

T3-gtn3	Resending data extracts to GTN
T3-gtn4	Change timeframes for sending files to GTN

Set 1-51A: Change frequency of transmittal for GTN files

T3-gtn4A	Ensure that frequency for sending files to GTN is set to 6 hours

Set 1-52: Resend non-GTN files

T1-datax3	Resend capabilities for non-GTN recipients

Set 1-53: Test Reports production

G5-exp-onhand	Functionality to produce export onhand shippable cargo list
G5-exp-hhg/pov-onhand	Functionality to produce export hhg/pov onhand shippable cargo list
G5-exp-haz/exp-onhand	Functionality to produce export hazardous/explosive shippable cargo list
G5-imp-onhand	Functionality to produce import cargo list
G5-imp-hhg	Functionality to produce import hhg cargo list
G5-imp-pov	Functionality to produce import pov cargo list
G5-imp-haz/exp	Functionality to produce import hazardous/explosive shippable cargo list
G5-tcmd-eff	Functionality to produce the TCMD Shipper Error Listing
G5-command-ton	Functionality to produce command tonnage report
G5-mnfst-recap	Functionality to produce manifest recap M/T by POE report

Pass/Fail

Set 1-54: Test code table maintenance to add codes

G6-code1	Identification of code tables centrally maintained and distributed to WPS
G6-code2	Identification of code tables centrally maintained, not distributed to WPS
G6-comm-add	Functionality to add a new code (centrally distributed functionality).
G6-commg1-add	Functionality to add a new code (centrally distributed functionality).
G6-commg2-add	Functionality to add a new code (centrally distributed functionality).
G6-cdist-add	Functionality to add a new code (centrally distributed functionality).
G6-damage1-add	Functionality to add a new code (centrally distributed functionality).
G6-damage2-add	Functionality to add a new code (centrally distributed functionality).
G6-damage3-add	Functionality to add a new code (centrally distributed functionality).
G6-delreason-add	Functionality to add a new code (centrally distributed functionality).
G6-flag-add	Functionality to add a new code (centrally distributed functionality).
G6-goods-add	Functionality to add a new code (centrally distributed functionality).
G6-hndlng-add	Functionality to add a new code (centrally distributed functionality).
G6-numconv-add	Functionality to add a new code (centrally distributed functionality).
G6-ocarr-add	Functionality to add a new code (centrally distributed functionality).
G6-opcode-add	Functionality to add a new code (centrally distributed functionality).
G6-pkg-add	Functionality to add a new code (centrally distributed functionality).
G6-port-add	Functionality to add a new code (centrally distributed functionality).
G6-shipstatus-add	Functionality to add a new code (centrally distributed functionality).
G6-tac-add	Functionality to add a new code (centrally distributed functionality).
G6-typecd-add	Functionality to add a new code (centrally distributed functionality).
G6-typevslcd-add	Functionality to add a new code (centrally distributed functionality).
G6-vanownr-add	Functionality to add a new code (centrally distributed functionality).
G6-vstat-add	Functionality to add a new code (centrally distributed functionality).
G6-cancel-add	Functionality to add a new code (non-distributed functionality).
G6-delay-add	Functionality to add a new code (non-distributed functionality).
G6-dodaac-add	Functionality to add a new code (non-distributed functionality).
G6-gbl-add	Functionality to add a new code (non-distributed functionality).
G6-ldgterms-add	Functionality to add a new code (non-distributed functionality).
G6-mode-add	Functionality to add a new code (non-distributed functionality).
G6-ship-add	Functionality to add a new code (non-distributed functionality).
G6-add-WPS	View additions to Edit Tables in WPS

Set 1-55: Test code table maintenance to modify a code

G6-comm-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality).
G6-commg1-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality).
G6-commg2-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality).
G6-cdist-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality).
G6-damage1-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-damage2-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)

Pass/Fail

G6-damage3-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-delreason-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-flag-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-goods-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-hndlng-mod1	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-numconv-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-ocarr-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-opcode-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-pkg-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-port-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-shipstatus-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-tac-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-typecd-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-typevslcd-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-vanownr-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-vstat-mod	Functionality to change a code and/or the description of a code (centrally distributed functionality)
G6-cancel-mod	Functionality to change a code and/or the description of a code (non-distributed functionality).
G6-delay-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-dodaac-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-ldgterms-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-mode-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-ship-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-gbl-mod	Functionality to change a code and/or the description of a code (non-distributed functionality)
G6-mod-WPS	View modifications to Edit Tables in WPS

Pass/Fail

Set 1-56: Test code table maintenance to delete a code

G6-comm-del	Functionality to delete a code (centrally distributed functionality).
G6-commg1-del	Functionality to delete a code (centrally distributed functionality).
G6-commg2-del	Functionality to delete a code (centrally distributed functionality).
G6-cdist-del	Functionality to delete a code (centrally distributed functionality).
G6-damage1-del	Functionality to delete a code (centrally distributed functionality).
G6-damage2-del	Functionality to delete a code (centrally distributed functionality).
G6-damage3-del	Functionality to delete a code (centrally distributed functionality).
G6-delreason-del	Functionality to delete a code (centrally distributed functionality).
G6-flag-del	Functionality to delete a code (centrally distributed functionality).
G6-goods-del	Functionality to delete a code (centrally distributed functionality).
G6-hndlng-del	Functionality to delete a code (centrally distributed functionality).
G6-numconv-mod2	Functionality to modify a code (centrally distributed functionality).
G6-ocarr-del	Functionality to delete a code (centrally distributed functionality).
G6-opcode-del	Functionality to delete a code (centrally distributed functionality).
G6-pkg-del	Functionality to delete a code (centrally distributed functionality).
G6-port-del	Functionality to delete a code (centrally distributed functionality).
G6-shipstatus-del	Functionality to delete a code (centrally distributed functionality).
G6-tac-del	Functionality to delete a code (centrally distributed functionality).
G6-typecd-del	Functionality to delete a code (centrally distributed functionality).
G6-typevslcd-del	Functionality to delete a code (centrally distributed functionality).
G6-vanownr-del	Functionality to delete a code (centrally distributed functionality).
G6-vstat-del	Functionality to delete a code (centrally distributed functionality).
G6-cancel-del	Functionality to delete a code (non-distributed functionality).
G6-delay-del	Functionality to delete a code (non-distributed functionality).
G6-dodaac-del	Functionality to delete a code (non-distributed functionality).
G6-gbl-del	Functionality to delete a code (non-distributed functionality).
G6-ldgterms-del	Functionality to delete a code (non-distributed functionality).
G6-mode-del	Functionality to delete a code (non-distributed functionality).
G6-ship-del	Functionality to delete a code (non-distributed functionality).
G6-del-WPS	Verify deletions to Edit Tables in WPS

Set 1-57: Reviewing outstanding discharge dates; entering/checking discharge dates

R6-disch-dts	Reviewing vessels with outstanding discharge dates
R6-voy4	Dates entered by functional manager
R6-voy5	Dates received from OCONUS via TTU data extract

Set 1-58: Checking discharge dates

R6-voy5	Dates received from OCONUS via TTU data extract

Pass/Fail

Set 1-59: Utilities

G6-util-unix1	Printing UNIX files (not IQ) to user's PC
G6-util-unix2A	Downloading UNIX files (not IQ) to user's PC drive A
G6-util-unix2B	Downloading UNIX files (not IQ) to user's PC drive B
G6-util-unix2C	Downloading UNIX files (not IQ) to user's PC drive C
G6-util-iq1	Printing IQ files to user's PC
G6-util-iq2A	Downloading IQ files to user's PC drive A
G6-util-iq2B	Downloading IQ files to user's PC drive B
G6-util-iq2C	Downloading IQ files to user's PC drive C

Set 1-60: Export Manifest distribution

Reg-G-mfst-001	Distributing export manifest files

Set 1-61: Export Manifest distribution

Reg-G-mfst-002	Printing export manifest at Hub

Set 1-62: Export Manifest distribution

Reg-G-mfst-003	Prepare export manifest distribution tables at Hub

Set 1-63: IQ

RG-IQ-001	Use of IQ for ad hoc queries

Set 1-64: IQ

Reg-G-Fort-to-Port	Process to add/change/delete unit cargo

Set 1-70: WPS autoload of a dataset with multiple data conditions

GTR-01	Datesset to test interfaces, reports, etc.

Set 1-71: WPS tally transactions on selected records and vessels

Import tally transaction 01	Discharge records on import manifests
Import tally transaction 02	Discharge vessel on import manifests
Import tally transaction 03	Disposition records on import manifests
Export tally transaction 01	Receipt records on export cargo

Set 1-72: WPS loading of records to test reject processing

Export T06-rej	Rejected duplicate records

Set 1-73: WPS manifest call (export manifest)

MFST-Call-WPS	Call of manifest at WPS terminal

PHASE 2: RECORDS ORIGINATING AT WPS AND PASSED TO THE REGIONAL DATABASE VIA ONE HUB

Pass/Fail

Set 2-1: Perform export data loads (R2-Export-xxx on Regional)

Export 01 Add - Vessel Control Register and export autoload from floppy
Export 02 Add - Van records with single consignee
Export 03 Add - Van HHGS records
Export 04 Add - Van POV records
Export 05 Add - Van AMMO records
Export 06 Add - Refrigerated Van records
Export 07 Add - Breakbulk records
Export 08 Add - Breakbulk records
Export 09 Add - Breakbulk records
Export 10 Add - General Cargo Breakbulk records
Export 11 Add - HHGS Breakbulk records
Export 12 Add - POV Breakbulk records
Export 13 Add - AMMO Breakbulk records
Export 14 Add - Unacc Baggage Breakbulk records
Export 15 Add - Govn't Vehicle Breakbulk records
Export 16 Add - Mail Breakbulk records
Export 17 Add - Classified Breakbulk records
Export 18 Add - Sensitive Breakbulk records
Export 19 Add - Controlled Breakbulk records
Export 20 Add - Hazardous Cargo Breakbulk records
Export 21 Add - Unitized/CONEX records
Export 22 Add - Empty Van records
Export 23 Add - Loaded Ro/Ro Van records
Export 23A Assign booking number to breakbulk records and run query
Export B01 Create records to test Tally, Regression, Manifest Call, Manifest Adjustment and History

Set 2-2: Perform import data loads (R3-Import-xxx on Regional)

Import 01 Create - Vessel Control Register
Import 01F Upload of floppy disk to test import tally
Import 01FF Create van with multiple commodities
Import T01 Autoload to test Import Tally and Regression

Set 2-3 Perform export status changes (R5-Export-xxx on Regional)

Export 24 Correct - Van Records by RECNO
Export 25 Correct - Van Records by TCN
Export 26 Correct - Van HHGS Records by RECNO
Export 27 Correct - HHGS Van Records by TCN
Export 28 Correct - POV Van Records by RECNO
Export 29 Correct - POV Van Records by TCN
Export 30 Correct - AMMO Van Records by RECNO
Export 31 Correct - AMMO Van Records by TCN
Export 32 Correct - Refrigerated Van Records by RECNO(Add second seal)
Export 33 Correct - Refirgerated Van Records by TCN(Add second seal)
Export 34 Correct - Breakbulk Records by RECNO

Pass/Fail

Export 35	Correct - Breakbulk Records by TCN
Export 36	Correct - Breakbulk Records by RECNO
Export 37	Correct - Breakbulk Records by TCN
Export 38	Correct - Breakbulk Records by RECNO
Export 39	Correct - Breakbulk Records by TCN
Export 40	Correct - Breakbulk General Cargo Records by RECNO
Export 41	Correct - Breakbulk general cargo records by TCN
Export 42	Correct - HHGS Breakbulk records by RECNO
Export 43	Correct - HHGS Breakbulk Records by TCN
Export 44	Correct - POV Breakbulk records by RECNO
Export 45	Correct - POV Breakbulk Records by TCN
Export 46	Correct - AMMO Breakbulk records by RECNO
Export 47	Correct - AMMO Breakbulk Records by TCN
Export 48	Correct - Unaccompanied Baggage Breakbulk records by RECNO
Export 49	Correct - Unaccompanied Baggage Breakbulk Records by TCN
Export 50	Correct - Government Vehicle Breakbulk Records by RECNO
Export 51	Correct - Government Vehicle Breakbulk Records by TCN
Export 52	Correct - Mail Breakbulk Records by RECNO
Export 53	Correct - Mail Breakbulk records by TCN
Export 54	Correct - Classified Breakbulk Records by RECNO
Export 55	Correct - Classified Breakbulk Records by TCN
Export 56	Correct - Sensitive Breakbulk Records by RECNO
Export 57	Correct - Sensitive Breakbulk Records by TCN
Export 58	Correct - Controlled Breakbulk Records by RECNO
Export 59	Correct - Controlled Breakbulk Records by TCN
Export 60	Correct - Hazardous Cargo Breakbulk Records by RECNO
Export 61	Correct - Hazardous Cargo Breakbulk Records by TCN
Export 62	Correct - Unitized/CONEX Records by RECNO
Export 63	Correct - Unitized/CONEX Records by TCN
Export 64	Correct - Empty Van Records by RECNO
Export 65	Correct - Empty Van records by TCN
Export 66	Correct - Loaded Ro/Ro Van by RECNO
Export 67	Correct - Loaded Ro/Ro Van by TCN
Export 68	Change - TCN by RECNO
Export 69	Change - TCN by TCN
Export 70	Delete - Records by TCN
Export 71	Delete - Records by RECNO
Export 72	Delete - Records by Container Numbers
Export 73	Mode to Lift - by RECNO
Export 74	Mode to Lift - by TCN
Export 75	Mode to Lift - by Container Number
Export 76	Change - STOW by TCN
Export 77	Change - STOW by RECNO
Export 78	SPLIT/STOW Location - by RECNO
Export 79	SPLIT/STOW Location - by TCN
Export 80	SPLIT/LIFT - by RECNO
Export 81	SPLIT/LIFT - by TCN
Export 82	MASS CORRECtION - by TCN
Export 83	MASS CORRECTION - by BVOY

Pass/Fail

Export 84 MASS CORRECTION - by VOYDOC

Set 2-4 Perform export status changes (R5-Export B-xx on Regional)

Export B02 Tally - Source stuff by RECNO
Export B03 Tally - Source stuff by TCN
Export B04 Tally - Source Stuff by container #
Export B05 Tally - Receipt by RECNO
Export B06 Tally - Receipt by TCN
Export B07 Tally - Receipt by container #
Export B08 Tally - Receipt by BVOY
Export B09 Tally - Stuff single stop van by RECNO
Export B10 Tally - Stuff single stop van by TCN
Export B11 Tally - Stuff single stop van by container #
Export B12 Tally - Stuff multi stop van by RECNO
Export B13 Tally - Stuff multi stop van by TCN
Export B14 Tally - Stuff multi stop van by container #
Export B15 Tally - Split stuff single stop van by RECNO
Export B16 Tally - Split stuff single stop van by TCN
Export B17 Tally - Split stuff single stop van by container #
Export B18 Tally - Transfer by RECNO
Export B19 Tally - Transfer by TCN
Export B20 Tally - Transfer by container #
Export B22 Tally - Lift by RECNO
Export B23 Tally - Lift by TCN
Export B24 Tally - Lift by container #
Export B26 Regress - Lifted to previous status by TCN
Export B27 Regress - Lifted to previous status by RECNO
Export B28 Regress - Lifted to previous status by container #
Export B30 Regress - Lifted to previous status by VOYDOC
Export B31 Regress - Tml stuffed contents to onhand BB by TCN
Export B32 Regress - Tml stuffed contents to onhand BB by RECNO
Export B33 Regress - Tml stuffed contents to onhand BB by container #
Export B34 Regress - Received to previous status by TCN
Export B35 Regress - Received to previous status by RECNO
Export B36 Regress - Received to previous status by container #
Export B38 Regress - Deleted to previous status by TCN
Export B39 Regress - Deleted to previous status by RECNO
Export B40 Regress - Deleted to previous status by container #
Export B42 Regress - Deleted to previous status by VOYDOC
Export B43 Manifest Export Cargo Records
Export B44 Adjust Manifested Export Cargo Records/Deletions, Corrections and Call Manifest Adjustments By TCN, RECNO or CNTNR
Export B45 Inactivate Export Vessel

Set 2-5 Perform import status changes (R5-Import-xxx on Regional)

Import 13 Correct - by TCN
Import 14 Correct - by VSNR and POSTNO
Import 15 Correct - STOW
Import 16 Correct - TCN

Pass/Fail

Import 17	Correct - Breakbulk baggage
Import 18	Correct - Breakbulk Ammo
Import 19	Correct - Breakbulk Mail
Import 20	Correct - Breakbulk HHG
Import 21	Correct - Breakbulk HazMat
Import 22	Correct - Breakbulk POV
Import 23	Correct - Breakbulk Government vehicle
Import 24	Correct - HHG van
Import 25	Correct - Baggage van
Import 26	Correct - Ammo van
Import 27	Correct - Mail van
Import 28	Correct - HazMat van
Import 29	Correct - POV van
Import 30	Correct - Government vehicle van
Import 30a	Correct - Empty Van by TCN
Import 31	Stuff - Baggage in a van by TCN
Import 32	Stuff - Ammo in a van by TCN
Import 33	Stuff - Mail in a van by TCN
Import 34	Stuff - HHG in a van by TCN
Import 35	Stuff - HazMat in a van by TCN
Import 36	Stuff - POV in a van by TCN
Import 37	Stuff - Govn't vehicle in a van by TCN
Import 38	Correct - Baggage contents by TCN
Import 39	Correct - Ammo contents by TCN
Import 40	Correct - Mail contents by TCN
Import 41	Correct - HHG contents by TCN
Import 42	Correct - HazMat contents by TCN
Import 43	Correct - POV by TCN
Import 44	Correct - Govn't veh contents by TCN
Import 45	Correct - Van prime w content by TCN
Import 46	Correct - Van prime w content by POSTNO
Import 47	Correct - Baggage contents by POSTNO
Import 48	Correct - Ammo contents by POSTNO
Import 49	Correct - Mail contents by POSTNO
Import 50	Correct - HHG contents by POSTNO
Import 51	Correct - HazMat contents by POSTNO
Import 52	Correct - POV contents by POSTNO
Import 53	Correct - Govn't veh content by POSTNO
Import 54	Mass Change - By TCN
Import 55	Delete - Van by TCN
Import 56	Delete - Breakbulk by TCN
Import 57	Delete - Van with contents by TCN
Import 58	Delete - Records by range
Import 59	Delete - Record by container #
Import 60	Delete - Record by POSTNO
Import 61	Regress - Record by POSTNO
Import 62	Regress - Record by container #
Import 63	Regress - Van by TCN
Import 64	Regress - Breakbulk by TCN

Pass/Fail

Import 65 Regress - Van with contents by TCN
Import 66 Regress - By POSTNO range
Import 67 Delete - By mass
Import 68 Regress - By mass
Import 69 Add Van with stop-offs and Contents
Import 70 Mass/Other Changes - By TCN
Import 71 Mass Correction by TCN
Import 72 Mass/Other Changes by VSNR & POSTNO
Import 75 TCN Change by TCN
Import 76 STOW Change by TCN
Import 77 Add contents to Unitized/CONEX
Import 78 Add additional contents Unitized/CONEX
Import 79 Correct contents Unitized/CONEX by TCN
Import 80 Correct contents in Unitized/CONEX by VSNR & POSTNO
Import 81 Add contents to a Loaded Ro/Ro by TCN
Import 82 Correct Contents in Loaded Ro/Ro by TCN
Import 83 Correct Contents in Loaded Ro/Ro by VSNR & POSTNO

Set 2-6 Perform import status changes (R5-Import-T-xx on Regional)

Import T-02 Review IQ Reports
Import T-03 Tally/Discharge - Container # & TCN
Import T-03a Transfer & Regress - Container # & TCN
Import T-04 Tally/Discharge - By Single Postno
Import T-04a Transfer and Regress - By Single POSTNO
Import T-05 Tally/Discharge - By Postno Range
Import T-05a Tranfer and Regress - By Postno Range
Import T-06 Tally/Discharge - By Consignee
Import T-06a Transfer by Consignee - Regress by Mass
Import T-07 Tally/Discharge - By VSNR
Import T-07a Transfer and Regress - By Mass
Import T-08 Tally/Discharge to Mode & Regress - By Container # & TCN
Import T-09 Tally/Discharge to Mode & Regress - By Single Postno
Import T-10 Tally/Discharge to Mode & Regress - By Postno Range
Import T-11 Tally/Discharge to Mode by Consignee Regress by Mass
Import T-12 Tally/Discharge to Mode by VSNR - Regress by Mass
Import T-13 Tally/Discharge - By VSNR
Import T-14 Tally/Disposition - By Container # and TCN
Import T-14a Regress - By Container # and TCN
Import T-15 Tally/Disposition - By Single Postno
Import T-15a Regress - By Single Postno
Import T-16 Tally/Disposition - By Postno Range
Import T-16a Regress - By Postno Range
Import T-17 Tally/Disposition - By Consignee
Import T-17a Regress - By Mass
Import T-18 Tally/Disposition - By VSNR
Import T-18a Regress - By Mass
Import T-19 Unstuff Container - by single post number
Import T-20 Unstuff Container - by TCN
Import T-21 Unstuff Container - by container #

Pass/Fail

Import T-22 Unstuff Container - by Post number range
Import T-23 Unstuff Contents - by single post number
Import T-24 Unstuff Contents - by TCN
Import T-25 Split Disposition - change TCN and split disposition by single post number
Import T-26 Unstuff Contents - by post number ranges

Set 2-7 Perform WPS export autoload from diskette (R2-Export 85-Reg)

Export 85 Autoload from Floppy

Set 2-8 Perform Interterminal transfer (R2-Export-xxx)

Export 87 Interterminal Transfer - by RECNO
Export 88 Interterminal Transfer - by TCN
Export 89 Interterminal Transfer - by Container Number

Set 2-9 Add POV records (R2-Export-xxx on Regional)

Export 91 Add - POV
Export 92 Add - Advance POV
Export 93 Add - POV Van Prime

Set 2-10 Change POV records (R5-Export-xxx on Regional)

Export 94 Change - TCN by RECNO
Export 95 Change - TCN by TCN
Export 96 Change - STOW by RECNO
Export 97 Change - STOW by TCN
Export 98 Delete - by TCN
Export 99 Delete - by RECNO
Export 100 Regress - by TCN
Export 101 Regress - by RECNO

Set 2-11 Automatic Purge (R5-TML-Exp-001 on Regional)

TML-Exp-001 Automatic purge on terminal system

Set 2-12 Flat File Transmittals

R1-flatfiles-to-WPS Transmittal of flat files (booking files and contractor pay) from Area Command to WPS terminal

Set 2-13 Flat File Transmittals

R1-fmsfiles-from-WPS Transmittal of flat files (FMS files) through Hub to FMS

Set 2-14 Flat File Transmittals

R1-atcmds-from-WPS-to-recipients Transmittal of flat files (ATCMDs) through Hub to ETADs and NAOMIS

Set 2-15 History

History-WPS Reading from WPS History

APPENDIX B

LIST OF TECHNICAL TCRS PREPARED BY ORNL TO SUPPORT FUNCTIONAL TCR TESTING

TECHNICAL TCRS
PREPARED FOR TESTING THE WPS CONUS REGIONAL DATABASE

Technical TCRs to support "Receive Data" test requirements

R1-dataload-BKG1	To load booking data prior to batch-loading ATCMDs
R1-dataload-BKG2	To load booking data after batch-load of ATCMDs
R1-dataload-ATCMD	To load ATCMD/ASPUR data
R1-dataload-ATCMD-2	To load ATCMD data to check correction of rejected atcmd records and invalid TACs
R1-dataload-arejrecords	To load records that will be rejected during preprocessing
R1-tech-areject	To check for rejected ATCMD/ASPUR records (formerly R1-dataload-areject)
R1-dataload-unit	To load unit move data with "A" in first position of TCN
R1-dataload-correct	To load data to change atcmds (add/delete contents, change LWH, change TAC)
R1-dataload-shipeff	To load data for checking all 39 errors on the TCMD Daily Error Report
R2-load-exp01-VSL-REG	To load vessel register data from the WPS xvoydoc table
R3-dataload-IMPORT-MAN	To load import manifest data
R3-dataload-IMPORT-MAN-2	To load import manifest data to check correction of rejected records
R3-dataload-IMPORT-MAN-3	To load import manifest data to check correction of rejected records
R3-dataload-mrejrecords	To load records that will be rejected during preprocessing
R3-tech-mreject	To check for rejected import manifest records (formerly R3-dataload-mrejects)
R3-load-imp01-VSL-REG	To load data from the WPS voydoc table
R4-tech-expmandist	To check export manifest distribution processes
R5-dataload-ACI	To load ACI data
R6-dataload-VSL-REGIST	To accept the vessel register from METS and send it to WPS
R6-dataload-TTU	To load TTU data from OCONUS sites
R6-tech-voy4	To test logic for updating discharge dates from screen
R6-tech-disch-dts	To ensure that manifest records exist which outstanding discharge dates

Technical TCRs to support the "Transmit Data" test requirements

T1-tech-datax1	To ensure that data files are generated, transmitted, and moved to a storage directory
T1-tech-datax3	To test resend capability for specific data file
T1-tech-datax6	To ensure that CDCP data files are generated, transmitted, and moved to a storage directory
T2-tech24a	To test for resend/divert entire manifest
T2-tech24b	To check that individual manifest records were resent or diverted
T2-tech-portdodaac	To check transfer when consignee = dodaac of a port
T3-tech-gtn1	To ensure that GTN data files are generated, transmitted, and moved to a storage directory
T3-tech-gtn4	To check changed frequency for GTN
T3-tech-gtn3	To test resend capability for specific GTN data file(s)
T6-tech118	To check that individual atcmd records were resent or diverted
T6-tech-rejects	To check records sent from ICDB that are rejected by WPS

Technical TCRs to support "General Requirements" test requirements

G4-tech-112a1	To check screen corrections of rejected ATCMDs
G4-tech-112a2	To check screen corrections of rejected manifests
G4-tech-112a3	To check screen corrections of invalid TACs
G6-tech-code1-add	To check addition of distributed code table data
G6-tech-code1-mod	To check modification of distributed code table data
G6-tech-code1-del	To check deletion of distributed code table data
G6-tech-code2-add	To check addition of non-distributed code table data
G6-tech-code2-mod	To check modification of non-distributed code table data
G6-tech-code2-del	To check deletion of non-distributed code table data
GX-tech-WTL1	To check that, when the server is down, the data is saved on the Hub and then, when the server comes back up, the data automatically goes to the server.
GX-tech-WTL2	To check that, when a site connection is down, the data is transmitted via SLIP

APPENDIX C

DESCRIPTION OF THE TEST DATA SCHEMA

This handout was provided to the testers during all tests as Attachment 4 to their test packet. It includes a list of TCNs sorted by category and the total number of records in each set. The test data schema assisted testers in using the flat-file printout of individual records. TCRs cross-referenced this handout for record counts within each cargo type category.

Attachment 4

Test Data Schema for WPS/WPS CONUS Regional Testing

The first 6 positions of the TCN for individual shipment unit records will be changed to enable testers to easily identify the specific cargo type and the source of data. The following is the schema for numbering:

Position
- 1 = Port Indicator
- 2 = Source of data/Container vs. Breakbulk
- 3-4 = Commodity Type
- 5 = Container Number or piece count
- 6 = Content Number

Port Indicator (TCN position 1)

0, 4	=	oakland (ports beginning 3xx; all ports not otherwise assigned to a termac)
1	=	socal
2, 5	=	seattle (ports beginning 4xx)
3	=	new orleans (ports beginning 2xx)

Source of data: Container vs. Breakbulk and batch vs. online (TCN position 2)

1 = ATCMD container batch	Data Set 1
2 = ATCMD container online	Data Set 2
3 = ATCMD breakbulk batch	Data Set 3
4 = ATCMD breakbulk online	Data Set 4
5 = import manifest container batch	Data Set 5
7 = import manifest breakbulk batch	Data Set 7

Commodity type codes (TCN positions 3-4)

00 Ammo/explosives	11 reefer cargo	19 empty van
01 Hazardous	12 general cargo with stopoffs	20 reserved
02 POV	13 mail	21 general cargo
03 HHG	14 loaded RO/RO	22 CONEX-related records
07 Unit cargo	15 unaccompanied baggage	23-29 reserved
08 Outsize	16 classified	30-31 booking records (with matches to batch ATCMDs)
09 Gvmt vehicles	17 sensitive	32-33 booking records (with matches to on-line builds)
10 General cargo	18 controlled	36 booking records (with no matches to ATCMDs)

Container number, piece count, or CONEX indicator where CONEX is not in a van (TCN position 5)
- 0-9 (container nbr, piece count)
- A, B, C (CONEX indicator, where CONEX is not in a van)

Contents indicator (TCN position 6)
- 0 (container record or a breakbulk prime record)
- 1-9 (number of specific content record)
- A, B (CONEX contained in a van)

Remainder of TCN will be filled with "X"s (except as noted, e.g., S in position 8 for unit cargo; ssn in positions 6-14 for HHG/POV cargo; booked voydoc in positions 7-10 for booking data sets only); also last three positions will be as given in the TCN of the original dataset.

Data Set – Booking (batch data load):

Cargo type category	TCN beg. No.	TCN end. No.	Tot. records	Comments
Booking Set 1A (Container)	0130008888V	0130908888V	10 vans	Match to batch ATCMDs, loaded before ATCMDs
Booking Set 2A (Container)	0131008888V	0131908888V	10 vans	Match to batch ATCMDs, loaded after ATCMDs
Booking Set 1C (Container)	0232008888V	0232908888V	10 vans	Match to ATCMDs built online, loaded before ATCMDs
Booking Set 2B (Container)	0233008888V	0233908888V	10 vans	Match to ATCMDs built online, loaded after ATCMDs
Booking Set 1B	0136008888V	0136908888V	10 vans	No matches to any ATCMD records

Total of 50 booking records loaded to database, 40 of which will be matched to ATCMDs

Data Set 1 – ATCMD Container (batch data load):

Cargo type category	TCN beg. No.	TCN end. No.	Act'l records loaded*	Comments
Ammo/Explos: Container	010000	010060	7 vans	
Ammo/Explosive: Contents	010001	010061	15 contents	
Hazardous: Container	010100	010190	10 vans	
Hazardous: Contents	010101	010199	77 contents	orig. 57 haz & 21 gen; 1 reject (ftnt †)
POV: Containers	010200	010290	10 vans	
POV: Contents	01020(ssn)	01029(ssn)	20 contents	Usu 2 per container
HHG: Container	010300	010390	10 vans	
HHG: Content	01030(ssn)	01039(ssn)	28 contents	Max of 9 per container
General: Container Set 1	011000	011090	10 vans	
General: Contents	011001	011099	14 contents	Max of 9 per container
General: Reefer Container	011100	011190	10 vans	
General: Reefer Contents	011101	011199	40 contents	Max of 9 per container
Loaded RO/RO	011400	011410	2 Ro/Ros	
RO/RO Contents	011401	011411	3 contents	
General: Empty Container	011900	011990	10 emp.vans	Actually breakbulk cargo
General: Container Set 2	012100	012180	9 vans	
General: Contents	012101	012189	63 contents	Orig. 64 contents; 1 reject
Van with CONEX	012200	012200	1 van	
CONEX in van	01220A	01220A	1 CONEX	CONEX in van (TCN = 012200)
Content in CONEX in van	012201	012201	1 content	Content in CONEX (TCN = 01220A) in van
CONEX not in van	0122A0	0122B0	2 CONEXs	2 CONEXs not in a van
CONEX contents	0122A1	0122B1	2 contents	2 contents in CONEXs
Booking Set 1A	0130008888V	0130908888V	10 vans	Matches to booking records
Booking Set 2A	0131008888V	0131908888V	10 vans	Matches to booking records

*A total of 365 ATCMD prime records (vans and contents) are actually loaded to the database. Two records from the original flat files are rejected.

†This rejected record is corrected during testing.

Data Set 2 – ATCMD Container (online build):

Cargo type category	TCN beg. No.	TCN end. No.	Tot. records	TCR & attachment No.
Ammo/Explos: Container	020000		1 van	R1-121-EXP05
Ammo/Explosive: Contents	020001	020003	3 contents	R1-121-EXP05-C1, C2, C3
Hazardous: Container	020100		1 van	R1-121-HAZ
Hazardous: Contents	020101	020103	3 contents	R1-121-HAZ-C1, C2, C3
POV: Containers	020200		1 van	R1-121-EXP04
POV: Contents	02020(ssn)	02020(ssn)	2 contents	R1-121-EXP04-C1, C2
HHG: Container	020300		1 van	R1-121-EXP03
HHG: Contents	02030(ssn)	02030(ssn)	5 contents	R1-121-EXP03-C1
General: Container	021000		1 van	R1-121-EXP02
General: Contents	021001	021005	5 contents	R1-121-EXP02-C1
General: Reefer van	021100		1 van	R1-121-EXP06 (and trailers)
General: Reefer contents	021101	021102	2 contents	R1-121-EXP06-C1, C2
General: Van with stopoffs	021200		1 van	R1-121-EXP07
General: Stp van contents	021201	021202	2 contents	R1-121-EXP07-C1, C2
Booking Set 1C	0232008888V	0232908888V	10 vans	10 matches to booking; R1-122-BKG-VAN1
Booking Contents	0232018888V	0232918888V	10 contents	1 content built for each booked van
Booking Set 2B	0233008888V	0233908888V	10 vans	10 matches to booking; R1-122-BKG-VAN2
Booking Contents	0233018888V	0233918888V	10 contents	1 content built for each booked van

*A total of 69 ATCMDs are built online and loaded to the database.

Data Set 3 – ATCMD Breakbulk (batch data load):

Cargo type category	TCN beg. No.	TCN end. No.	Act'l tot. records	Comments
Ammo/Explosives	030000	030010	2 pieces	
Hazardous	030100	030190	10 pieces	
POV	03020(ssn)	03029(ssn)	10 pieces	
HHG	030300	030390	0 pieces	
Unit	030700X$	030790X$	10 pieces	
Outsize	030800	030830	4 pieces	
Government Vehicles	030900	030990	19 pieces	10 orig. pieces, exploded to 19; see last 3 positions of TCN for TCN beginning 030900

*A total of 55 ATCMDs are loaded to the database.

Data Set 4 – ATCMD Breakbulk (online build):

Cargo type category	TCN beg. No.	TCN end. No.	Act'l tot. records	TCR & attachments No.; comments
Ammo/Explosives	040000	040020	3 pieces	R1-121-EXP12
Hazardous	040100	040120	3 pieces	R1-121-EXP19
POV	04020(ssn)	04022(ssn)	3 pieces	R1-121-EXP11
HHG	04030(ssn)	04032(ssn)	3 pieces	R1-121-EXP10
Outsize	040800	040820	3 pieces	R1-121-EXP14A
Government Vehicles	040900	040920	3 pieces	R1-121-EXP-14
General BB Cargo	041000	041020	3 pieces	R1-121-EXP08
Mail	041300	041320	3 pieces	R1-121-EXP15
Unaccomp'd Baggage	04150(ssn)	04152(ssn)	3 pieces	R1-121-EXP13
Classified	041600	041620	3 pieces	R1-121-EXP16
Sensitive	041700	041720	3 pieces	R1-121-EXP17
Controlled	041800	041820	3 pieces	R1-121-EXP18
Empty Van	041900	041920	3 pieces	R1-121-EXP21

*A total of 42 ATCMDs are built online and loaded to the database.

Data Set 5 -- Import Manifest Containers (batch data load):

Cargo type category	TCN beg, No.	TCN end. No.	Act'l records loaded*	Comments
Ammo/Explo: Container	050000	050090	7 vans	
Ammo/Explosive:Contents	050001	050061	15 contents	Max of 9 per container
Hazardous: Container	050100	050190	10 vans	
Hazardous: Contents	050101	050199	77 contents	orig. 57 haz + 21 gen; 1 reject
POV: Containers	050200	050290	10 vans	
POV: Contents	05020(ssn)	05029(ssn)	20 contents	Usu 2 per container
HHG: Container	050300	050390	10 vans	
HHG: Content	05030(ssn)	05039(ssn)	34 contents	26 HHG and 8 general
General: Container Set 1	051000	051090	10 vans	
General: Contents	051001	051099	10 contents	Max of 9 per container
General: Reefer Container	051100	051190	10 vans	2 of these are reefer RO/ROs
General: Reefer Contents	051101	051199	28 contents	Max of 9 per container
Loaded RO/RO	051400	051410	2 Ro/Ros	
RO/RO Contents	051401	051411	3 contents	
General: Empty Container	051900	051990	10 vans	Actually considered breakbulk
General: Container Set 2	052100	052180	9 vans	
General: Contents	052101	052189	9 contents	
Van with CONEX	052200	052210	1 van	
CONEX in van	05220A	05220A	1 CONEX	CONEX in van (TCN = 052200)
Content in CONEX	052201	052201	1 content	Content in CONEX (TCN = 05220A) in van
CONEX not in van	0522A0	0522B0	2 CONEXs	CONEXs not in van
CONEX contents	0522A1	0522B1	2 contents	Contents in CONEX

*A total of 281 actual cargo records loaded to the database.
†This rejected record is corrected during testing.

Data Set 7 – Import Manifest Breakbulk (batch data load):

Cargo type category	TCN beg. No.	TCN end. No.	Act'l tot. records	Comments
Ammo/Explosives	070000	070010	2 pieces	
Hazardous	070100	070190	10 pieces	
POV	07020(ssn)	07029(ssn)	10 pieces	
HHG	07030(ssn)	07039(ssn)	0 pieces	
Unit	070700X$	070790X$	10 pieces	
Outsize	070800	070830	4 pieces	
Government Vehicles	070900	070990	10 pieces	

A total of 46 actual cargo records are loaded to the database.

APPENDIX D

A LIST OF HANDOUTS OF FLAT FILE LAYOUTS OF DATA LOADED VIA BATCH INPUT

This handout was provided to the testers during all tests. It provided a list to the testers of all of the flat-file printouts they were receiving. Attachments 7A-7E were printouts of individual records that matched the criteria described in Appendix C of this document.

Attachment 7A: ATCMD Containers and Breakbulk --PRIME DATA ONLY. These are the data sets defined in Attachment 4 as Data Sets 1 (container batch input) and 3 (breakbulk batch input)

Attachment 7B: ATCMD Containers and Breakbulk -- PRIMES AND TRAILERS. These are the data sets defined in Attachment 4 as Data Sets 1 (container batch input) and 3 (breakbulk batch input)

Attachment 7C-1: BOOKING DATA. These are the data sets defined in Attachment 4 as Booking Sets 1A, 1B, and 1C

Attachment 7C-2: BOOKING DATA. These are the data sets defined in Attachment 4 as Booking Sets 2A and 2B

Attachment 7D: Import Manifest Containers and Breakbulk --PRIME DATA ONLY. These are the data sets defined in Attachment 4 as Data Sets 5 and 7

Attachment 7E: Import Manifest Containers and Breakbulk -- PRIMES AND TRAILERS. These are the data sets defined in Attachment 4 as Data Sets 5 and 7

Attachment 7F: ACI Data

Attachment 7G: Vessel Register Data

Attachment 7H: Discharge and Disposition Data (TTU Records) from OCONUS

Attachment 7I: Unit Move Records for Army, with an "A" in the first position of the TCN (10 records)

Attachment 7J: ATCMD batch changes/corrections (add/delete content, change LWH, change TAC) to previously submitted records

Attachment 7K: Data for testing the TCMD Effectiveness Report

Attachment 7L: Duplicate Records, testing reject processing

APPENDIX E

TEST ORDER

This handout was provided to the test leaders to assist in maintaining control over the order of the testing.

TEST ORDER FOR TCRS (WPS Regional Database)

PHASE 1: RECORDS ORIGINATING AT THE REGIONAL DATABASE AND PASSED TO WPS VIA ONE HUB

Note: these tasks may be run concurrently with Phase 2 tasks

Approval Date	Regional Technical Tasks	Regional Functional Tasks	WPS Tasks
		THIS SECTION TESTS PROCESSING OF ATCMD DATA FROM REGIONAL TO WPS SITES (3DK, 4DK) THROUGH WAXX	
	Load 1st set of booking data (R1-dataload-BKG1)	1-1 Test bkg data load (R1-113-BKG-1, -2, -3)	
	Load vessel register from METS (R6-dataload-VSL-REGIST)		Load METS vessel register
	Load ATCMDs (R1-dataload-ATCMD)	1-2 Test matchup of booking and ATCMD data (R1-113-BKG-4, -5)	
	Load set of Army unit move data (R1-dataload-unit)		
		1-3 Build ATCMDs on-line (R1-121-EXP02 thru R1-121-EXP15)	WPS "follow-on" TCRs to R1-121-EXP02 thru R1-121-EXP15 (note: first TCR contains manual autoload)
		1-4 Remaining TCRs to test load of ATCMD data (all R1-113-atcmd-xxx; R1-117)	WPS "follow-on" TCRs to R1-113-atcmd-xxx and R1-117 (R1-atcmd-xxx)
		1-5 Build remaining ATCMDs on-line (R1-121-EXP16 thru R1-122-refuse)	Allow nightly autoload to load these ATCMDs and check for receipt on WPS; WPS "follow-on" TCRs to R1-121-EXP16 thru R1-121-refuse)
	Load 2nd set of booking data (R1-dataload-BKG2)	1-6 Test matchup of booking and ATCMD data (R1-113-BKG-6, -7, -8)	
		1-7 Resend/Divert ATCMDs (T6-118a1, a3, b1, b2's, b3)	
	Check for resent/diverted ATCMDs on WPS work tables (T6-tech118)		WPS follow-on TCRs to T6-118xxx (T6-118-WPS; note: TCR contains manual autoload)
	Load ACI data (R5-dataload-ACI)		Print and reprint ACI data and enter data into WPS
	Load batch correction records (R1-dataload-correct)		

	Load batch set of records for testing reject processing (R1-dataload-arejrecords)		
	Check ATCMD reject list (R1-tech-areject)		
		1-8 Correct rejected ATCMD record (G4-112a1)	
		1-9 Correct invalid TACs (G4-112a3)	
	Load dataset to initiate automated correction processes (R1-dataload-ATCMD-2); check correction of rejected ATCMD (G4-tech-112a1)		
	Check correction of invalid TACs (G4-tech-112a3)		1-10 Process batch corrections, on-line ATCMD corrections, and on-line TAC corrections on WPS (G4-112-WPS -- note: TCR contains manual autoload)
		THIS SECTION TESTS PROCESSING OF IMPORT MANIFEST DATA FROM REGIONAL TO WPS SITES (3DK, 4DK) THROUGH WAXX	
	Load import manifests (R3-dataload-IMPORT-MAN and R3-dataload-IMPORT-MAN-2)		
	Load data for testing manifest reject processing (R3-dataload-mrejrecords)		
		1-22 Test load of import manifest data (all R3-21-import-xxx)	WPS "follow-on" TCRs to R3-21-import-xxx (note: first TCR contains manual autoload)
		1-23 Resend and divert manifests after receipt of documentation; resend and divert by cntr nbr; resend and divert by TCN (all T2-24ax, bx, and cx)	
	Check for resent/diverted manifests on WPS work tables (T2-tech24a and 24b)		WPS follow-on TCRs to all T2-24's (note: first TCR contains manual autoload for 4DS)
	Check manifest reject list (R3-dataload-mreject)	1-24 Correct rejected manifest record (G4-112a2)	
	Load data to initiate automated correction processing (R3-dataload-IMPORT-MAN-3); check correction of rejected mmfst record (G4-tech-112a2)		

		THIS SECTION TESTS "OTHER" TYPES OF PROCESSING ORIGINATING AT THE REGIONAL DATABASE – THESE TESTS ARE AIMED PRIMARILY AT "GENERAL" REQUIREMENTS	
		1-50 Test all intransit visibility requirements (all G1-xxx)	
			1-70 WPS autoload of data sets with multiple data conditions (GTR-01)
			1-71 WPS performs tally transaction on selected records and vessels (Import tally transactions-01 - 03 and Export tally transactions-01)
	Check data extracts for GTN (T3-tech-gtn1)		
	Check non-GTN data extracts (T1-tech-datax1, 6)		
		1-51 Resend GTN files; change frequency of transmittal for GTN files (T3-gtn3, 4)	
		1-52 Resend non-GTN files (T1-datax3)	
	Check resent GTN and non-GTN files (T3-tech-gtn3 and T3-tech-gtn4; T1-tech-datax3)		
		1-51A Ensure that GTN frequency is set to 6 hours after testing is complete (T3-gtn4a)	
	Load dataset for testing ATCMD shipper effectiveness conditions (R1-dataload-shipeff)		
		1-53 Test reports production (G5-xxx)	
		1-54 Test code table maintenance and distribution (all G6-xxx-add); *note: do only the ones to add a code here*	
	Check for code table distribution on WPS work tables (G6-tech-code1-add and G6-tech-code2-add)		WPS "follow-on" TCR (G6-add-WPS)
		1-55 Test code table maintenance and distribution (all G6-xxx-mod); *note: do only the ones to modify a code here*	

	Check for code table distribution on WPS work tables (G6-tech-code1-mod and G6-tech-code2-mod)		WPS "follow-on" TCR (G6-mod-WPS)
		1-56 Test code table maintenance and distribution (all G6-xxx-del); *note: do only the ones to delete a code here*	
	Check for code table distribution on WPS work tables (G6-tech-code1-del and G6-tech-code2-del)		WPS "follow-on" TCR (G6-del)
	Ensure that outstanding discharge dates are available for updating (R6-tech-disch-dts)	1-57 Reviewing outstanding discharge dates (R6-disch-dts)	
	Monitor functional process for updating discharge dates (R6-tech-voy4)	1-57a Entering/checking discharge dates (R6-voy4)	
	Load of discharge dates via TTU (R6-dataload-TTU)	1-58 Checking discharge dates updated via TTU process (R6-voy5)	
	Check records that were rejected by WPS (T6-tech-rejects)		1-72 WPS TCR for testing reject processing (Export-T06-Rej)
		1-59 Utilities for printing unix and IQ files and saving to a disk (G6-util-xxx)	
			1-73 WPS call the manifest processing (MFST-Call-WPS
		1-60 Export Manifest dist. processes (Reg-G-mfst-001)	
	Check export manifest distribution processes (R4-tech-expmandist)	1-61 Export Manifest print process (Reg-G-mfst-002)	
		1-62 Export manifest distribution table preparation (Reg-G-mfst-003)	
		1-63 IQ (RG-IQ-001)	
	1-64 Fort-to-port processing (Reg-G-Fort-to-Port)	Check on Regional	Check on WPS

PHASE 2: RECORDS ORIGINATING AT WPS AND PASSED TO THE REGIONAL DATABASE VIA ONE HUB

Note: these tasks may be run concurrently with Phase 1 tasks

Approval Date	Regional Technical Tasks	Regional Functional Tasks	WPS Tasks
		THESE TASKS TEST LOADING OF DATA AT THE REGIONAL FROM WPS	
			2-1 Perform all export data loads (Nita's and Kathy's data entry -- Export 01 - Export 23 and Export B01)
	Test transfer of voyage data (R2-load-EXP01-VSL-REG)	All req's related to receiving WPS ATCMD data (all R2-Export-xxx)	
			2-2 Perform all import data loads (Georgia and Rod's data entry -- Import 01 - Import 12 and Import T01)
	Test transfer of voyage data (R3-load-IMP01-VSL-REG)	All req's related to receiving WPS import-manifest-loaded-at-port data (all R3-Import-xxx)	
			2-3 Perform export status changes (Export 24 - Export 84)
		All req's related to receiving WPS status changes (all R5-Export-24-Reg - R5-Export-84-Reg)	
			2-4 Perform export status changes (Export B02 - Export B45)
		All req's related to receiving WPS status changes (R5-Export B02-Reg - R5-Export B45-Reg)	
			2-5 Perform import status changes (Import 13 - Import 83)
		All req's related to receiving WPS status changes (R5-Import 13 -Reg - R5-Import 83-Reg)	
			2-6 Perform import status changes (Import T-02 - Import T-26)
		All req's related to receiving WPS status changes (R5-Import-T-02-Reg - R5-Import-T-26-Reg)	

			2-7 Perform WPS export autoload from diskette (Export 85)
		All req's related to receiving WPS data (R2-Export 85-Reg)	
			2-8 Perform Interterminal transfer (Export 87 - Export 89)
		Receiving interterminal transfer (R2-Export-87-Reg - R2-Export 89-Reg)	
			2-9 Add POV records (Export 91 - Export 93)
		Receive new POV records (R2-Export 91-Reg - R2-Export-93-Reg)	
			2-10 Change POV records (Export 94 - Export 101)
		Receive changes to POV records (R5-Export 94-Reg - R5 -Export-101-Reg)	
			2-11 Automatic purge (TML-EXP-001)
		Follow-on TCRs to automatic purge (TML-Exp-001-Reg)	2-12 Flat file transmittals of booking files and contractor pay to WPS [Bob Watts to coordinate]
			2-13 Flat file transmittals of FMS files from WPS thru Hub [Bob Watts to coordinate]
			2-14 Flat file transmittals of ATCMDs to ETADs and NAOMIS [Bob Watts to coordinate]
			2-15 Accessing records in WPS History

APPENDIX F

FORMS THAT WERE USED DURING MAISRC TESTING OF ICDB

Figure F.1 DA Form 5005-R. Engineering Change Proposal -- Software. This form was used for writing up problems that were identified during testing. Whoever identified the problem described it in sufficient detail that the developers could then assign priority, responsibility, and a recommended solution.

Figure F.2 SQT - WPS Regional Code Upload Log. This form was used to keep track of the status of code uploads to correct problem reports. It was constantly updated, sometimes multiple times a day. This form was developed and maintained by ORNL.

ENGINEERING CHANGE PROPOSAL—SOFTWARE (ECP-S)

For use of this form, see AR 25-3; the proponent agency is ODISC4

(Check one)
☐ PROBLEM REPORT ☐ ECP-S

1. TO:	2. FROM:

3. ORIGINATOR NUMBER	4. POINT OF CONTACT (Name and telephone no.)	5. PRIORITY (Check one if ECP-S) ☐ EMERGENCY ☐ URGENT ☐ ROUTINE

6. APPLICATION CI BASELINE/VERSION	7. EXECUTIVE SW BASELINE/VERSION	8. PROBLEM DATE (YYMMDD)

9. JOB/CYCLE/PROGRAM ID

10. TITLE OF PROBLEM/CHANGE

11. DESCRIPTION OF PROBLEM/CHANGE (List all attachments and referenced documents) (If additional space is needed, use Item 15, Remarks)

12. EFFECT ON USER (If additional space is needed, use Item 15, Remarks)

13. RECOMMENDED SOLUTION/JUSTIFICATION (If additional space is needed, use Item 15, Remarks)

14. DATE (YYMMDD)	NAME AND TITLE OF SUBMITTING AUTHORITY	SIGNATURE

DA FORM 5005-R, NOV 81 REPLACES DA FORM 4157-R, 1 FEB 78, WHICH IS OBSOLETE.

15. REMARKS *(If additional space is needed, use separate sheet of paper)*

USER/MACOM ACTION *(ECP-S Only)*

16. MACOM *(Check one and include any comments)*

☐ APPROVE

☐ DISAPPROVE

17. DATE *(YYMMDD)*	NAME AND TITLE	SIGNATURE

ASSIGNED RESPONSIBLE AGENCY *(Problem Report Only)*

18. PROBLEM REPORT ACTION TAKEN *(Check one)*

☐ RESOLVED BY CUSTOMER ASSISTANCE

☐ IDENTIFIED AS URGENT OR ROUTINE

☐ EMERGENCY ECP FORMALIZED

☐ DUPLICATE OF EXISTING ECP: NO. _____

☐ CANCELED BY ORIGINATOR

☐ CANCELED FOR INSUFFICIENT IDENTIFICATION

☐ CANCELED FOR INSUFFICENT DOCUMENTATION

19. DATE *(YYMMDD)*	NAME AND TITLE	SIGNATURE

PROPONENT AGENCY *and/or* **ASSIGNED RESPONSIBLE AGENCY** *(ECP-S Only)*

20. CLASS OF ECP *(Check one)*	21. JUSTIFICATION CODE	22. ECP NUMBER
☐ I ☐ II		

23. ECP TYPE *(Check one)*	24. ESTIMATED COSTS/SAVINGS
☐ PRELIMINARY ☐ FORMAL	

25. OTHER SYSTEM/CI AFFECTED

26. CHANGE IDENTIFICATION *(Check one in each column)*

☐ FUNCTIONAL/ALLOCATED	☐ MAJOR	☐ MAINTENANCE
☐ TECHNICAL/PRODUCT	☐ MINOR	☐ MODIFICATION

27. PROJECTED IMPLEMENTATION

28. APPROVAL AUTHORITY *(Check agency and action taken)*

☐ PROPONENT AGENCY ☐ ASSIGNED RESPONSIBLE AGENCY ☐ APPROVED ☐ DISAPPROVED

29. DATE *(YYMMDD)*	NAME AND TITLE	SIGNATURE

REVERSE OF DA FORM 5005-R, NOV 81

SQT - WPS Regional Code Upload Log

Date/Time Changed	Code	Functionality	PR Nbr	Priority	Notes	Regression Test Set
			001			
			002			
			003			
			004			
			005			
			006			
			007			
			008			
			009			
			010			
			011			

INTERNAL DISTRIBUTION

1. M. S. Bronzini
2. G. E. Courville
3. S. G. Hildebrand
4. S. Moore
5. C. I. Moser
6-8. J. P. Rollow
9. R. B. Shelton
10. P. C. Shipe
11. B. E. Tonn
12-14. L. F. Truett
15. Central Research Library
16. Document Reference Section
17-18. Laboratory Records
19. Laboratory Records - RC
20. ORNL Patent Office

EXTERNAL DISTRIBUTION

21. Ms. Betty Childers, ICDB Product Management Office, Military Traffic Management Command, 5611 Columbia Pike, Falls Church, VA 22041-5050.
22. Mr. Herb Kaskoff, WPS Project Management Office, Military Traffic Management Command, 5611 Columbia Pike, Falls Church, VA 22041-5050.
23. Dr. Thomas E. Drabek, Professor, Department of Sociology, University of Denver, Denver, CO 80208-0209.
24. Mr. Jack Hetman, ICDB Product Management Office, Military Traffic Management Command, 5611 Columbia Pike, Falls Church, VA 22041-5050.
25. Mr. Calvin D. MacCracken, President, Calmac Manufacturing Corporation, 101 West Sheffield Avenue, Englewood, New Jersey 07631.
26. Ms. Ruth Olgivie, ICDB Product Management Office, Military Traffic Management Command, 5611 Columbia Pike, Falls Church, VA 22041-5050.
27. Mr. George F. Sowers, Senior Vice President, Law Companies Group, Inc., 114 Townpark Drive, Suite 250, Kennesaw, GA 30144-5599.
28. Dr. C. Michael Walton, Paul D. and Betty Robertson Meed Centennial, Professor and Chairman, Department of Civil Engineering, College of Engineering, The University of Texas at Austin, Cockrell Hall, Suite 4.2, Austin, TX 78712.
29. ORNL Site Manager, U.S. Department of Energy, Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6269.
30-31. Office of Scientific and Technical Information, U.S. Department of Energy, P.O. Box 62, Oak Ridge, Tennessee 37831.