

**Final**

# **ELECTRONIC RECORDS ARCHIVES**

## **TESTING MANAGEMENT PLAN (TSP v4.0)**

**(WBS # 1.8.1.16.1)**

**for the**

**NATIONAL ARCHIVES AND  
RECORDS ADMINISTRATION**

**ELECTRONIC RECORDS ARCHIVES  
PROGRAM MANAGEMENT OFFICE  
(NARA ERA PMO)**

Final  
September 1, 2006

PREPARED BY:

*Integrated Computer Engineering (ICE) Directorate  
of  
American Systems*

Contract Number: GS-35F-0673K  
Delivery Order Number: NAMA-01-F-0031/06-010

Final

Electronic Records Archives (ERA)

Testing Management Plan (TSP)

ERA Program Management Office (ERA PMO)

Final

ERA Testing Management Plan (TSP)

Signature Page

Program Director,

I recommend approval of the Testing Management Plan (TSP).

Approved,

  
\_\_\_\_\_  
Dyung Le  
ERA Director of Systems Engineering Division

10/3/2006  
Date

Approved,

  
\_\_\_\_\_  
Kenneth Thibodeau,  
ERA Program Director

10.5.06  
Date

Final

### Document Change Control Sheet

Document Title: Testing Management Plan (TSP)

Date	Filename/Version #	Author	Revision Description
7/2/02	ERA.DC.TSP.1.0.doc	Ursula Parker	Baseline Testing Management Plan
5/1/03	ERA.DC.TSP.2.0.doc	Ursula Parker	Changes made per Government's comments
5/23/03	ERA.DC.TSP.2.1.doc	Ursula Parker	Changed made per CR-ERA-PMO-DCMT-47
08/31/05	ERA.DC.TSP.3.0.doc	Penny Ha	Changed made per CR# - ERA00000729 and Consolidate Review Comment Form dated 8/11/05.
07/24/06	ERA.DC.TSP.4.0.doc	David Leistner	Change made per CR# - ERA00001039. This is Version 4.0 DRAFT
09/01/06	ERA.DC.TSP.4.0.doc	David Leistner	Updates made per Government comments moving this to Version 4.0 FINAL

# TABLE OF CONTENTS

<b>SIGNATURE PAGE</b> .....	ERROR! BOOKMARK NOT DEFINED	<b>i</b>
<b>DOCUMENT CHANGE CONTROL SHEET</b> .....		<b>III</b>
<b>1.0 INTRODUCTION</b> .....		<b>1</b>
1.1 PURPOSE .....		1
1.2 ERA PROGRAM OVERVIEW .....		2
1.3 MISSION DESCRIPTION.....		2
1.4 SYSTEM DESCRIPTION .....		2
1.4.1 Key Features.....		2
1.4.2 Interfaces .....		2
1.5 SECURITY ASSESSMENT .....		3
1.6 MEASURES OF EFFECTIVENESS AND SUITABILITY.....		3
<b>2.0 ACRONYMS AND DEFINITIONS</b> .....		<b>3</b>
2.1 ACRONYMS .....		3
2.2 DEFINITIONS .....		5
<b>3.0 REFERENCED DOCUMENTS</b> .....		<b>9</b>
3.1 ERA PMO REFERENCES .....		9
3.2 INDUSTRY STANDARDS AND REFERENCES.....		9
3.3 LMC REFERENCES .....		10
<b>4.0 ERA TESTING MANAGEMENT STRUCTURE</b> .....		<b>10</b>
4.1 ROLES AND RESPONSIBILITIES .....		11
4.2 STAFFING.....		13
4.3 INCREMENTAL APPROACH.....		14
4.4 STATUS REPORTS .....		14
4.5 MANAGEMENT AND SOFTWARE QUALITY ASSURANCE (SQA) REVIEWS.....		15
4.6 TEST READINESS REVIEWS (TRRS) .....		15
4.7 OPERATIONAL READINESS REVIEW (ORR).....		15
<b>5.0 TEST AND EVALUATION (T&amp;E)</b> .....		<b>15</b>
5.1 DEVELOPMENT TEST (DT) OVERVIEW .....		16
5.1.1 Unit and String Testing.....		16
5.1.2 Integration Testing.....		17
5.1.3 System Testing .....		17
5.2 DEVELOPMENT TEST (DT) ENTRANCE AND EXIT CRITERIA .....		17
5.3 ACCEPTANCE TEST (AT) OVERVIEW.....		17
5.3.1 Production Acceptance Tests (PAT).....		19
5.3.2 Operational Acceptance Tests (OAT).....		19
5.3.3 Installation Acceptance Tests (IAT).....		19
5.4 INSTALLATION, TESTING, AND CONTROL.....		20
5.5 ACCEPTANCE TEST (AT) ENTRANCE AND EXIT CRITERIA.....		21
<b>6.0 TEST REPORTING</b> .....		<b>21</b>
<b>7.0 INDEPENDENT VERIFICATION AND VALIDATION (IV&amp;V)</b> .....		<b>22</b>
<b>8.0 TEST AND EVALUATION RESOURCE SUMMARY</b> .....		<b>22</b>

Final

8.1 TEST ITEMS ..... 22

8.2 TEST ENVIRONMENTS AND FACILITIES ..... 22

8.3 TEST SUPPORT EQUIPMENT ..... 23

8.4 TEST BEDS ..... 23

8.5 SPECIAL REQUIREMENTS ..... 23

8.6 STAFFING AND PERSONNEL TRAINING ..... 24

**9.0 RISKS AND CONTINGENCIES..... 24**

**10.0 PLAN MAINTENANCE..... 24**

**APPENDIX A: TESTING EXECUTION..... A-1**

**APPENDIX B: TEST METHODOLOGY OVERVIEW .....B-1**

**APPENDIX C: ACCEPTANCE CRITERIA DEVELOPMENT GUIDELINES ..... C-1**

**APPENDIX D: ACCEPTANCE TEST SCENARIO DEVELOPMENT ..... D-1**

**APPENDIX E: ACCEPTANCE TEST SCENARIO/TEST CASE GUIDELINES AND LAYOUT .....E-1**

**APPENDIX F: ACCEPTANCE TEST LOGS .....F-1**

**APPENDIX G: TEST & EVALUATION TEST PLANS..... G-1**

**APPENDIX H: ACCEPTANCE TEST REPORT ..... H-1**

**APPENDIX I: PROGRAM TROUBLE REPORT GUIDELINES .....I-1**

**APPENDIX J: ACCEPTANCE TEST PROCESS ..... J-1**

**1.0 INTRODUCTION ..... J-1**

**2.0 PURPOSE ..... J-1**

**3.0 OBJECTIVES AND GOALS ..... J-1**

**4.0 METHODOLOGY ..... J-1**

4.1 APPROACH ..... J-2

4.2 VERIFICATION..... J-7

4.3 AT PROTOCOL..... J-7

4.4 AT ENTRY CRITERIA ..... J-8

4.5 EVALUATION AND RETEST..... J-8

**5.0 MANAGEMENT..... J-8**

5.1 TEST CONTROL..... J-9

5.2 TEST RESULTS RECORD KEEPING..... J-9

**6.0 ACCEPTANCE PROCESS STRUCTURE..... J-9**

6.1 STEPS FOR ACCEPTANCE PROCESS ..... J-9

6.2 AT EXIT CRITERIA..... J-12

**7.0 ORGANIZATIONAL INTERFACES ..... J-12**

Final

7.1 DEVELOPMENT CONTRACTOR ..... J-12

7.2 CONFIGURATION MANAGEMENT (CM) ..... J-12

7.3 QUALITY MANAGEMENT (QM) ..... J-12

7.4 INDEPENDENT VERIFICATION AND VALIDATION (IV&V)..... J-12

**8.0 REPORTING..... J-13**

8.1 PROGRAM TROUBLE REPORTS..... J-13

8.2 PROGRESS ..... J-13

8.3 TEST METRICS..... J-13

**APPENDIX K: INCREMENT 1 ACCEPTANCE TEST OVERVIEW..... K-1**

**TABLE OF FIGURES**

Figure 4-1: PMO Test Organization Chart ..... 11

Figure A-1: ERA Testing Execution..... A-1

Figure C-1: Defining Acceptance Criteria & Tests..... C-1

Figure D-2: Acceptance Test Development Approach ..... D-3

Figure I-1: PTR Defect Life Cycle Task Flow..... I-2

Figure J-1: Acceptance Process Diagram ..... J-9

Figure K-1: Increment 1 Acceptance Test Flow ..... K-2

Figure K-2: Increment 1 Test Activities ..... K-6

Figure K-2: Increment 1 Test Activities (continued)..... K-7

**LIST OF TABLES**

Table 2-1: Acronyms List ..... 5

Table 4-1: Test Organization Roles and Responsibilities..... 13

Table 4-2: Skill of Personnel by Type and Test Phase ..... 14

Table J-1: Pre-Acceptance Checklist ..... J-3

Table J-2: Acceptance Checklist..... J-6

Table K-1: Increment 1 Acceptance Test Milestones ..... K-5

# Testing Management Plan (TSP)

## 1.0 INTRODUCTION

This Testing Management Plan (TSP) addresses and provides guidance for the testing management activities to be performed in support of the National Archives and Records Administration (NARA) Electronic Records Archives (ERA) program. It is designed to capture and convey the overall structure and objectives of the ERA Test and Evaluation (T&E) activities.

### 1.1 Purpose

This document provides a basis for planning, performing, managing, monitoring, and measuring the ERA system testing activities. Specifically, this plan documents the following:

- References that will be used as the basis for test management, planning, development, and documentation;
- The organizations responsible for planning, management, and test execution;
- Management of a testing strategy that addresses the evolution of the design, incremental delivery, testing efficiency, and testing coverage as well as the system's known areas of risk;
- An overview of the testing process to include testing phases and processes for evaluating test adequacy;
- Test facility, test equipment, and test support requirements;
- Approach for documenting, tracking, and resolving issues found during testing;
- Measurement and reporting of test work products and test results; and
- The approach for developing acceptance criteria.

The TSP is a program level document and is applicable to ERA testing activities in the development lifecycle phases. This TSP focuses on the overall test management approach used to ensure a high quality system that meets user acceptance criteria. Thus, this TSP is analogous to a master test and evaluation plan. ERA testing will include the full system (i.e., application, distributed infrastructure, middleware, and supporting system services). During the development of a release product and the subsequent development efforts leading to Initial Operating Capability (IOC), the ERA Testing Team will be responsible for overseeing and monitoring the Development Contractor's test efforts, ensuring the product is tested against the requirements, and ensuring the deliverables derived from their test efforts comply with requirements. The ERA Testing Team also has responsibilities for conducting testing during the acceptance process for each delivery (i.e., releases and increments). Further ERA Testing Team duties, activities, and responsibilities will be discussed later in this document.

Credible sources such as the IEEE Std 12207.1, *Standard for Information Technology: Software Life Cycle Processes – Life Cycle Data* and the Software Engineering Institute's (SEI) Software Capability Maturity Model (SW-CMM) address and provide guidance for a managed test process but do not recommend a format or framework for a document detailing testing management.

**Final**

Consequently the methodology found in this TSP is based on guidance from the Department of Defense (DoD) Standard 5000.2-R, *Test and Evaluation Master Plan (TEMP)*, April 2002 and IEEE Std 829-1998, *Standard for Software Test Documentation*. The TSP has been modified and updated to fit ERA's system requirements as well as industry "Best Practices" for testing large systems.

**1.2 ERA Program Overview**

ERA will be a comprehensive, systematic, and dynamic means for storing, preserving, and accessing virtually any kind of electronic record, free from dependence on any specific hardware or software. The ERA system, when operational, will make it easy for NARA customers to find the records they want and easy for NARA to deliver those records in formats suited to customers' needs. The success of the ERA Program Management Office (PMO) in building and deploying the ERA system will depend on professional program and project management with an emphasis on satisfying NARA's requirements for a viable system.

**1.3 Mission Description**

The testing management methodology and activities depicted in this TSP will ensure that the ERA system meets NARA's strategic goals by addressing the deficiencies identified in the *ERA Mission Needs Statement (MNS)*.

**1.4 System Description**

ERA will be an agency-wide system that is capable of managing the entire lifecycle of the electronic records that NARA receives. The system will be developed to satisfy a core set of requirements that address the entire lifecycle of electronic holdings, and the needs of the system's users. When fully operational, ERA will authentically preserve and provide access to any kind of archived electronic record, free from dependency on any specific hardware or software.

**1.4.1 Key Features**

ERA system key features are described in the *ERA Requirements Document (RD)*.

**1.4.2 Interfaces**

The system will be capable of interfacing and interacting with other systems as needed. The ERA system will interface with four (4) classes of systems:

- Financial Systems,
- Non-Electronic Records Tracking Systems,
- Help Desk System, and
- Transferring Entity Systems.



**Final**

These interfaces are described in the *NARA ERA Interface Requirements Specification (IRS)* and the *NARA ERA Interface Control Document (ICD) Transferring Entity System Interface*. An ICD will be created for each system that will interface with ERA.

**1.5 Security Assessment**

Early in the development lifecycle, threats to the ERA system will be minimal. As the ERA system matures from the IOC state to the Full Operational Capability (FOC) state, threats will increase. The vast amount of information stored, processed, and transferred by ERA could make it a likely target of diverse threats, compromise of data, disruption of service, or loss of information. Testing of the ERA system will be performed to establish a high degree of confidence in the security of ERA and to minimize system threats. The ERA Security Team performs all testing related to security of ERA with support from the ERA Test Team. Final security certification and accreditation will be performed by an independent organization.

NARA and the ERA PMO will determine the level of security required for the ERA system, Development Contractor - Lockheed Martin Corporation (LMC), ERA Testing Team, test environments, test facilities, and proprietary components.

The *System Security Plan (SSP – CDRL 11)* for ERA provides more detail on the anticipated ERA system security scope and activities.

**1.6 Measures of Effectiveness and Suitability**

The ERA system must meet the requirements set forth in the *ERA RD*. The *ERA RD* presents the ERA requirements and reflects the critical components of the ERA. The requirements baselines that are developed will provide detailed and system level criteria that will be the basis for the testing and evaluation of the design and performance of the system and its components.

The ERA Testing Team will ensure that the ERA system meets the system performance objectives identified in the *ERA RD*.

**2.0 ACRONYMS AND DEFINITIONS**

The terms used in this document are defined in IEEE Std. 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology* and in the Joint Publication 1-02, “*DoD Dictionary of Military and Associated Terms*.”

**2.1 Acronyms**

ACRONYM	DESCRIPTION
16 CSP	16 Critical Software Practices
ANSI	American National Standards Institute
APB	Acquisition Program Baseline

## Final

ACRONYM	DESCRIPTION
AS	Acquisition Strategy
AT	Acceptance Test
BP	Business Practices
CCB	Configuration Control Board
CI	Configuration Item
CM	Configuration Management
CMM	Capability Maturity Model
CMP	Configuration Management Plan
CMTP	Contractor's Master Test Plan
CO	Contracting Officer
COTS	Computer Off-the-Shelf
CR	Change Request
CSCI	Computer Software Configuration Items
DoD	Department of Defense
DT	Development Test
ERA	Electronic Records Archives
ERB	Engineering Review Board
FAR	Federal Acquisition Regulation
FCA	Functional Configuration Audit
FOC	Full Operational Capability
GOTS	Government Off-the-Shelf
HWCI	Hardware Configuration Items
IAT	Installation Acceptance Tests
ICD	Interface Control Document
ICE	Integrated Computer Engineering
IEEE	Institute of Electrical and Electronics Engineers
IOC	Initial Operational Capability
IRD	Interface Requirements Document
IV&V	Independent Verification & Validation
IVVP	Independent Verification and Validation Plan
KPP	Key Performance Parameters
LM	Lockheed Martin
LMC	Lockheed Martin Corporation
MNS	Mission Needs Statement
MP	Metrics Plan
MTP	(Contractor's) Master Test Plan
NARA	National Archives and Records Administration
NIST	National Institute of Standards and Technology
OAT	Operational Acceptance Tests
ORR	Operational Readiness Review

ACRONYM	DESCRIPTION
PaT	Production Acceptance Tests (informal)
PAT	Production Acceptance Tests (formal)
PCA	Physical Configuration Audit
PD	Program Director
PMO	Program Management Office
PMP	Program Management Plan
PRP	Peer Review Process
PTR	Problem Tracking Report
PWS	Performance Work Statement
QA	Quality Assurance
QC	Quality Control
QM	Quality Management
QMP	Quality Management Plan
RD	Requirements Document
RFP	Request for Proposal
RKM	Risk Management Plan
SED	Systems Engineering Division
SEI	Software Engineering Institute
SME	Subject Matter Expert
SOW	Statement of Work
SPMN	Software Program Manager Network
SQA	Software Quality Assurance
SSP	System Security Plan
SW-CMM	Software Capability Maturity Model
T&E	Test and Evaluation
TEMP	Test and Evaluation Master Plan
TOMP	Task Order Management Plan
TRR	Test Readiness Review
TSP	Testing Management Plan
VS	Vision Statement

Table 2-1: Acronyms List

## 2.2 Definitions

**Acceptance criteria:** The criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity. (**Appendix C, Acceptance Criteria Development Guidelines**)

**Acceptance testing:** (1) Formal testing conducted to determine whether a system satisfies its acceptance criteria and enables the customer to determine whether to accept the system. (2)

**Final**

Formal testing conducted to enable a user, customer, or other authorized entity to determine whether to accept a system or component.

**Development testing:** Formal or informal testing conducted during the development of a system or component, usually in the development environment by the developer.

**Functional testing:** (1) Testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions. Contrast with: **Structural testing**. (2) Testing conducted to evaluate the compliance of a system or component with specified functional requirements. See also: **Performance testing**.

**Independent Verification and Validation (IV&V):** Verification and validation performed by an organization that is technically, managerially, and financially independent of the development organization.

**Installation and checkout phase:** The period of time in the software lifecycle during which a software product is integrated into its operational environment and tested in this environment to ensure that it performs as required.

**Integration testing:** Testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them. See also: **System testing; Unit testing**.

**Load testing:** Testing that studies the behavior of the system when it is working at its limits. See also: **Stress Testing**.

**Operational Readiness Review (ORR):** A review conducted to verify that the test procedures for Operational Acceptance Testing (OAT) are complete, comply with test plans and descriptions, and satisfy test objectives. Verify that a project is prepared to proceed to the next step of formal testing.

**Operational testing:** Testing conducted to evaluate a system or component in its operational environment.

**Path testing (coverage):** Testing that is designed to execute all or selected paths through a computer program.

**Pass/Fail criteria:** Decision rules used to determine whether a software item or software feature passes or fails a test.

**Performance testing:** Testing conducted to evaluate the compliance of a system or component with specified performance requirements. See also: **Functional testing**.

**Program Trouble Report (PTR):** A document reporting on any event that occurs during the testing process that requires investigation.

**Quality Assurance (QA):** (1) The process of evaluating overall project performance on a regular basis to provide confidence that the project will satisfy the relevant quality standards. (2) The organizational unit that is assigned responsibility for quality assurance. (*A Guide to the Project Management Body of Knowledge (PMBOK Guide), 2000 Edition*)

**Quality Control (QC):** (1) The process of monitoring specific project results to determine if they comply with relevant quality standards and identifying ways to eliminate causes of unsatisfactory performance. (2) The organizational unit that is assigned responsibility for quality control. (*A Guide to the Project Management Body of Knowledge (PMBOK Guide), 2000 Edition*)

**Quality Management (QM):** Ensures that planning, performing, managing, monitoring, and measuring the ERA quality management activities are accomplished.

**Regression testing:** Selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements.

**Scenario:** (1) A description of a series of events that could be expected to occur simultaneously or sequentially. (2) An account or synopsis of a projected course of events or actions. (*IEEE Std. 1362-1998, Guide for Information Technology – System Definition – Concept of Operations (ConOps) Document*)

**Software item:** Source code, object code, job control code, control data, or a collection of items.

**Stress testing:** Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements. See also: **Load testing**.

**String Testing:** The testing of interfaces between individual software units or groups of related units (i.e., component, modules).

**Structural testing:** Testing that takes into account the internal mechanism of a system or component. Types include branch testing, path testing, statement testing. Contrast with: **Functional testing**.

**System testing:** Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. See also: **Integration testing; Unit testing**.

**Test:** An activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component.

**Test case specification:** A document specifying inputs, predicted results, and a set of execution conditions for a test item (also called **Test case**).

**Final**

**Test design specification:** Documentation specifying the details of the test approach for a software feature or combination of software features and identifying the associated tests.

**Test item:** A software item that is an object of testing.

**Test log:** A chronological record of relevant details about the execution tests.

**Test phase:** The period of time in the lifecycle during which components of a system are integrated, and the product is evaluated to determine whether or not requirements have been satisfied.

**Test plan:** A document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning.

**Test procedure:** (1) Detailed instructions for the set-up, execution, and evaluation of results for a given test case. (2) A document containing a set of associated instructions as in (1). (3) Documentation specifying a sequence of actions for the execution of a test.

**Test Readiness Review (TRR):** A review conducted to evaluate preliminary test results for one (1) or more configuration items and verify that the test procedures for each configuration item are complete, comply with test plans and descriptions, and satisfy test requirements. Verify that a project is prepared to proceed to formal testing of the configuration item. (Also see **ORR**)

**Test summary report:** A document summarizing testing activities and results. It also contains an evaluation of the corresponding test items.

**Test script:** The steps of a test case that have been automated are now in the scripting language of the automated functional test tool.

**Testability:** (1) The degree to which a system or component facilitates the establishment of test criteria and the performance of tests to determine whether those criteria have been met. (2) The degree to which a requirement is stated in terms that permit establishment of test criteria and performance of tests to determine whether those criteria have been met.

**Testing:** (1) The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component. (2) The process of analyzing a software item to detect the differences between existing and required conditions (i.e., bugs) and to evaluate the features of the software items. See also: **Acceptance testing; Development testing; Integration testing; Operational testing; Performance testing; Regression testing; System testing; Unit testing.**

**Unit Testing:** The testing of individual hardware or software units or groups of related units (i.e., component, modules). See also: **Integration testing; System testing.**

### 3.0 REFERENCED DOCUMENTS

This section lists the industry standards, references, and documents that provide guidance in the development of the TSP.

#### 3.1 ERA PMO References

The following ERA PMO documentation was used to support the generation of this document. Please note that the documents referenced were current at the time of reference and publication, and remain so unless superseded by a subsequent version.

- ERA Configuration Management Plan (CMP), version 2.3
- ERA Quality Management Plan (QMP), version 3.0
- ERA Peer Review Process (PRP), version 1.1
- ERA Program Management Plan (PMP), version 3.0
- ERA Metrics Plan (MP), version 4.0
- ERA Risk Management Plan (RKM), version 3.0
- ERA Systems Security Plan (SSP), version 4.0
- ERA Vision Statement (VS), version 1.0
- ERA Missions Needs Statement (MNS), version 1.2
- ERA Independent Verification and Validation Plan (IVVP), version 1.1

#### 3.2 Industry Standards and References

The following industry standards and references were used in the creation of this document.

- IEEE Std 829-1998, Standard for Software Test Documentation
- IEEE Std 12207.1, Standard for Information Technology: Software Life Cycle Processes – Life Cycle Data
- American National Standards Institute (ANSI)/IEEE Std 1008-1987, Standard for Software Unit Testing
- DoD Standard 5000.2-R, Test and Evaluation Master Test Plan (TEMP), April 2002
- Testing Computer Software (Second Edition), Cem Kaner et. al., Wiley Computer Publishing, 1999
- Software Testing and Continuous Quality Improvement, William E. Lewis, CRC Press LLC, 2000
- Software Program Manager Network (SPMN), Road to Performance-Based Management, Based on 16 Critical Software Practices (16 CSP)
- IEEE Std 610.12-1990, Standard Glossary of Software Engineering Terminology
- Integrated Computer Engineering (ICE) Integration & Test Process Guidance
- Software Engineering Institute, “Capability Maturity Model, Version 1.1”
- Joint Publication 1-02, “DoD Dictionary of Military and Associated Terms”
- IEEE Std 1362-1998, Guide for Information Technology – System Definition – Concept of Operations (ConOps) Document

**Final**

- A Guide to the Project Management Body of Knowledge (PMBOK Guide), 2000 Edition
- National Institute of Standards and Technology (NIST) System Assessment Questionnaire

**3.3 LMC References**

The following LMC documentation was used to support the generation of this document. Please note that the documents referenced were current at the time of reference and publication, and remain so unless superseded by a subsequent version.

- System Security Plan (SSP) CDRL 11, version 1.1.0
- Interface Requirements Specification (IRS), December 08, 2005
- Interface Control Document (ICD) Transferring Entity System Interface CDRL 50, version 1.2.0
- System Requirements Specification (SyRS) CDRL 01, December 08, 2005
- System Integration Plan (SIP) CDRL 56, version 1.1.0

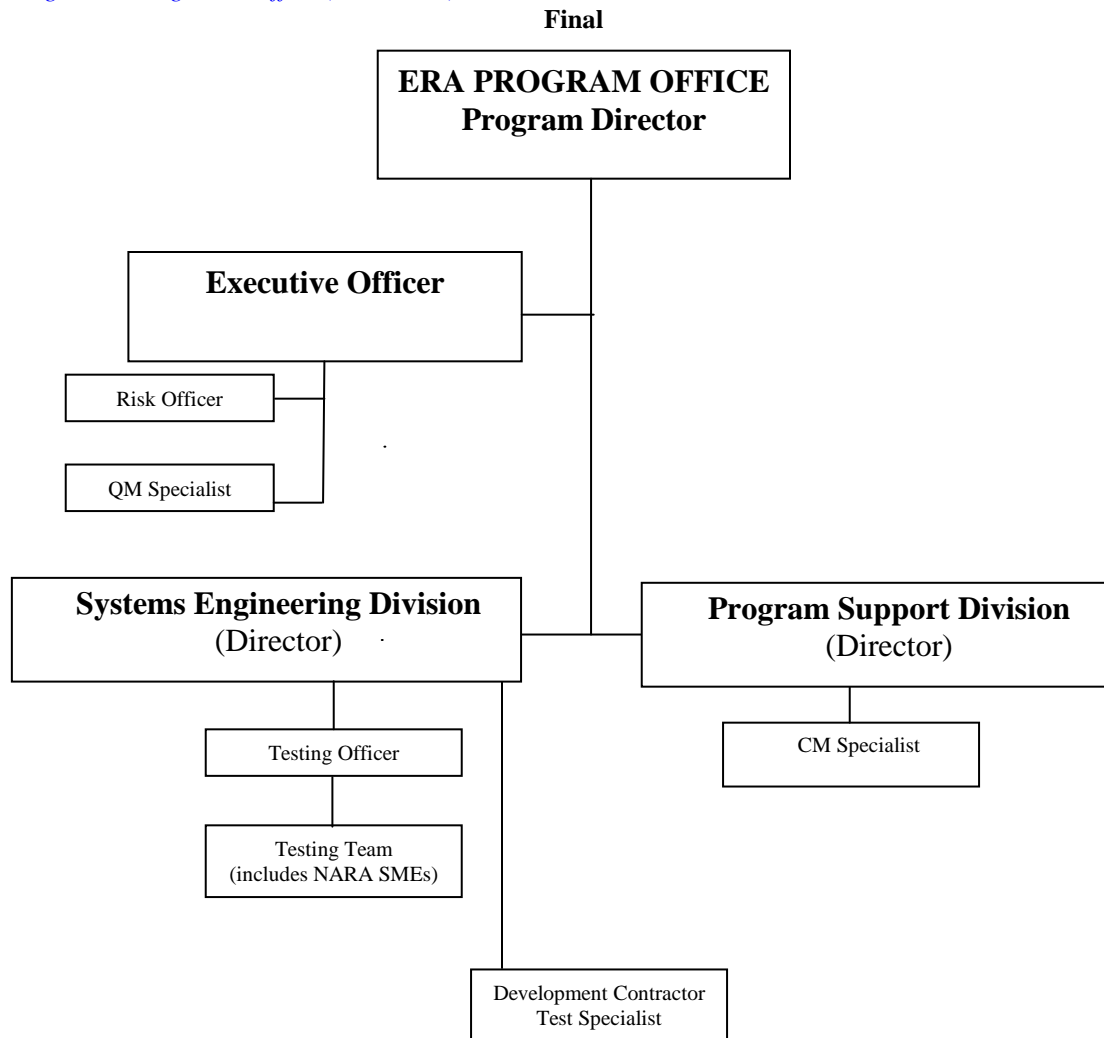
**4.0 ERA TESTING MANAGEMENT STRUCTURE**

The ERA Program Management Organization (PMO) Test Organization consists of representatives from the ERA PMO in each of the following roles.

- Program Director (PD)
- Systems Engineer Director
- Testing Officer
- Testing Team
- Configuration Management (CM) Specialist
- Development Contractor Test Specialist
- Quality Management (QM) Specialist
- Risk Officer

**Figure 4-1, PMO Test Organization Chart**, defines the organizational chart and categories of personnel who participate in testing management and the test process.





**Figure 4-1: PMO Test Organization Chart**

**4.1 Roles and Responsibilities**

**Table 4-1, Test Organization Roles and Responsibilities**, lists the primary responsibilities of each role of the Test Organization.

Roles	Required Responsibilities
Program Director (PD)	<ul style="list-style-type: none"> <li>• Ensure testing resources are assigned early enough to provide for adequate test preparation</li> <li>• Periodically review test results to ensure the software satisfies its requirements</li> <li>• Define a project level software testing organization and identify the responsibilities of the testing organization</li> </ul>

## Final

Roles	Required Responsibilities
Systems Engineer Director	<ul style="list-style-type: none"> <li>• Review the TSP and provide feedback to the Testing Officer</li> <li>• Provide the Testing Officer with the standards, policies, tools, and procedures applicable to the project</li> <li>• Support testing activities by confirming the Testing Officer's and Testing Team's responsibilities and authority</li> </ul>
Testing Officer	<ul style="list-style-type: none"> <li>• Responsible for development and execution of acceptance tests</li> <li>• Responsible for identifying any formal testing standards, then ensuring that those standards are being followed</li> <li>• Identify what to test and determine when testing resources are needed to ensure adequate test preparation</li> <li>• Determine and acquire needed test environment and support, if required</li> <li>• Monitor and control test work products and test results</li> <li>• Oversee overall testing effort</li> <li>• Develop acceptance test plan and revise the plan, as needed</li> <li>• Review test work products to ensure that they are complete and are developed according to plan</li> <li>• Review test scripts and scenarios to ensure they satisfy acceptance criteria</li> <li>• Review and validate test plans, procedures, scripts, and scenarios</li> <li>• Review test report templates</li> <li>• Review test results to determine whether software satisfies ERA objectives</li> <li>• Identify and configuration manage testing tools</li> <li>• Attend test related peer reviews</li> <li>• Support or conducts Readiness Reviews (TRRs/ORRs)</li> </ul>
Testing Team	<ul style="list-style-type: none"> <li>• Execute test plan(s)</li> <li>• Develop and execute test design specifications, procedures, scenarios, cases, and scripts</li> <li>• Attend peer reviews of requirements and software to ensure in depth knowledge of the functionality of the software</li> <li>• Peer review test plan(s), procedures, test cases, test scripts, and scenarios</li> <li>• Analyze each requirement to verify it can be tested</li> <li>• Document and monitor test issues and track to closure</li> <li>• Perform tool administration</li> <li>• Participate in test related peer reviews</li> <li>• Participate in TRRs/ORRs</li> </ul>

## Final

Roles	Required Responsibilities
NARA Subject Matter Expert (SME)	<ul style="list-style-type: none"> <li>• Witness test execution during development testing</li> <li>• Execute and validate tests during acceptance testing</li> </ul>
CM Specialist	<ul style="list-style-type: none"> <li>• Monitor software builds conducted by Developer Contractor</li> <li>• Conduct Physical and Functional audits following the successful completion of Acceptance Testing</li> </ul>
QM Specialist	<ul style="list-style-type: none"> <li>• Ensure testing is conducted per the test plan or procedures</li> <li>• Conduct QA audits of testing process</li> <li>• Perform QA inspections</li> </ul>
Risk Officer	<ul style="list-style-type: none"> <li>• Uncover and assess technical risks during testing phases</li> <li>• Assist in developing risk mitigation plans</li> <li>• Track and report risk information to the PD</li> </ul>
Development Contractor Test Specialist- LMC	<ul style="list-style-type: none"> <li>• Ensure program development documentation maps to the PMO documents</li> <li>• Develop and execute test design specifications, procedures, scenarios, cases, and scripts</li> <li>• Review and validate all development test plans, test design specifications, procedures, scenarios, cases, and scripts</li> <li>• Develops test reports</li> <li>• Support or conducts Readiness Reviews (TRRs/ORRs)</li> <li>• Schedule and conduct software builds</li> </ul>

Table 4-1: Test Organization Roles and Responsibilities

## 4.2 Staffing

Table 4-2, **Skill of Personnel by Type and Test Phase**, lists the ERA PMO personnel types required to adequately test the ERA System.

Type	Skill	Test Phase
Testing Officer	Management of testing knowledge and issue resolution management	All
Test Engineers (Test Team)	<ul style="list-style-type: none"> <li>• Monitor testing at all levels</li> <li>• Prepare test scripts and data</li> <li>• Execute and validate all test scripts and scenarios, as needed</li> <li>• Analyze technical test results</li> <li>• Resolve technical issues as they arise</li> </ul>	All
CM Specialist	Confirm and control changes over approved configuration items from all phases of acceptance testing	All

## Final

Type	Skill	Test Phase
QM Specialist	Review testing documentation and perform audits, as required	All
Risk Officer	Manage risk assessment strategy and report risk information	All
System Test Administrator	Manage the problem report database, system software and hardware configuration items, database loading/refresh scheduling, and test tracking tool for Customer Acceptance Test Lab	All
NARA Subject Matter Expert (SME)	Execute tests and review results during Operational Acceptance Testing	System, Acceptance

Table 4-2: Skill of Personnel by Type and Test Phase

### 4.3 Incremental Approach

The ERA Testing Team will use an incremental approach to T&E. This approach will provide usable, operational outputs at the completion of each increment and/or release. There will be three (3) releases within the first increment of ERA, the second release being the IOC release. There will be two (2) releases for each subsequent increment, the second release within each increment being the operational release. At preliminary releases, the Development Contractor – LMC, will be subject to Test Readiness Reviews (TRRs). At operational releases TRRs will be conducted, as well as Operational Readiness Reviews (ORRs). TRRs are discussed in **Section 4.6**. ORRs are discussed in **Section 4.7**. The operational releases or increments are contractually binding milestones:

- **Increment #1** – Provides IOC and incorporates testing of the core system functionality by the third release.
- **Later Increments** – Incorporates the testing of improvements and additions to Increment #1. The final increment will complete the FOC.

Each increment is divided into two (2) or three (3) releases. The ERA testing strategy will support the strategy with monitoring Unit, Integration, and System testing for each release. The strategy includes informal Product acceptance Tests (PaTs; little a) for preliminary releases. For these preliminary releases a TRR will be conducted as well as a post-test briefing on the results. For operational releases, a more formal Product Acceptance Test (PAT) will be executed with the addition of Operational Acceptance Tests (OATs) and Installation Acceptance Tests (IATs). A TRR will also be conducted before the IAT.

### 4.4 Status Reports

Testing activities will be reported on a regular basis. During the development testing activities, the LMC will provide testing activity status reports to the ERA PMO and any other affected

**Final**

groups. During the acceptance testing activities, the ERA Testing Team will provide testing activity status reports to ERA senior program management.

**4.5 Management and Software Quality Assurance (SQA) Reviews**

All testing activities (i.e., Development Test (DT) and Acceptance Test (AT)) will be reviewed with senior management in ERA Program Management Meetings, and with Development Contractor project management in review meetings. In addition, the testing process will be subject to QA reviews and audits. Refer to the *ERA Quality Management Plan (QMP)* for information on the role of QM in SQA reviews.

**4.6 Test Readiness Reviews (TRRs)**

TRRs are technical in nature and will be conducted by the Development Contractor (e.g., development engineers, testing engineers, QM specialists, CM specialists) with the ERA PMO in attendance. The goal of the review is to ensure that all related test items and materials have been completed and are ready for turnover to the next test phase. Additionally, the TRR provides ERA PMO and LMC management with the assurance that the developed ERA system has undergone a thorough test process. Reviews will be held for each operational increment at the completion of system testing for that increment. The *ERA QMP* provides guidance on review activities and process.

**4.7 Operational Readiness Review (ORR)**

An ORR is designed to provide an understanding of the status of ERA and the readiness for OAT. The state of the system, status of the associated PAT, acceptance test procedures, and any issues are presented. In addition the test schedule and activities are reviewed to ensure that all parties involved are in-sync with responsibilities and expectations during OAT. ORRs will be conducted by the ERA PMO with the support of the Development Contractor. The *ERA QMP* provides guidance on review activities and process.

**5.0 TEST AND EVALUATION (T&E)**

ERA T&E involves DT and AT. Details on DT and AT activities are provided within the DT and AT sections of this document.

In general, T&E is structured to:

- Provide essential information to support decision-making,
- Provide essential information for assessing technical risk,
- Determine the technical performance parameters of the design and architecture,
- Verify the attainment of technical performance specifications and objectives, and
- Verify that systems are operationally effective and suitable for their intended use.

The ERA PMO Testing Team will monitor test items, features, methods, processes, and documentation for compliance with standards and testing adequacy. To improve the testing

**Final**

process, the ERA PMO and LMC Testing Teams will capture metrics and will use metric reports to analyze and report on the status of testing. Refer to the *ERA Metrics Plan (MP)* for comprehensive metric activities.

DT and AT will be oriented toward demonstrating system performance as listed in the Key Performance Parameters (KPPs). Assessments will be conducted by QM and CM during the DT effort in order to determine programmatic risk, to support TRRs, and subsequent AT. The AT effort will collect data to support overall test objectives. Testing Team leader will conduct TRRs to ensure that the software, hardware, test environments, test facilities, and test engineers are ready to begin testing.

The ERA Testing Team will establish the necessary discipline, rigor, and structure to achieve the objectives of T&E by implementing and managing the testing strategy, assigning resources, witnessing testing, and monitoring results.

An overview of T&E is illustrated in **Appendix A, Testing Execution**.

**Section 5.0** presents an overview of acceptance testing (along with **Appendix J, Acceptance Test Process**) for the entire life of the project. **Appendix K, Increment 1 Acceptance Testing Overview**, presents testing activities and test coverage for Increment 1.

## **5.1 Development Test (DT) Overview**

The objectives of DT are to verify the status of development, verify that design risks have been minimized, demonstrate that all technical and performance requirements specified in the contract are met, and certify readiness for AT. DT is structural in nature and will consist of Unit, Integration, and System Testing. DT will be performed by the ERA Development Contractor and can be witnessed by the ERA Testing Team, Independent Verification and Validation (IV&V), QM, and any other designated representatives. The Development Contractor - LMC has prepared a Master Test Plan (MTP), which is the highest level development test plan and describes testing that will be conducted to demonstrate that the technical and performance requirements specified in the contract have been met. The MTP also identifies lower level development test plans that will be prepared to describe tests such as Unit, Integration, and System tests.

The Development Contractor will prepare test reports following the completion of each phase of testing (i.e., Unit, Integration, and System). Refer to **Section 6.0** for information on test reporting.

### **5.1.1 Unit and String Testing**

This phase of testing is considered the basic level of testing that focuses on the smaller building blocks of a program (e.g., components, modules) or system separately. Unit Testing is the earliest phase of testing and is the most cost-effective phase in removing defects. Unit testing permits the testing and debugging of small units, thereby providing a better way to manage the integration of the units into larger units. The detailed unit design is used as a basis to compare

**Final**

how and what the unit is able to perform. Unit Testing will be conducted by the Development Contractor and can be witnessed by the ERA Testing Team, IV&V, and QM. Unit Test documentation can be reviewed by QM, CM, IV&V, and the ERA Testing Team but will need to be requested.

**5.1.2 Integration Testing**

Following Unit Testing and prior to the beginning of System Testing, groups of units are fully tested. Units are systematically added one (1) or more units at a time to the core of already integrated modules. The goals of integration testing are to verify that services and methods interact correctly and hardware and software are integrated adequately. Integration Testing will be conducted by the Development Contractor to demonstrate accurate operation of the integrated units. The Integration documentation can be reviewed by QM, CM, IV&V, and the ERA Testing Team, but will need to be requested.

**5.1.3 System Testing**

This phase of testing occurs prior to formal acceptance testing. Its purpose is to test the system as a whole for functionality and fitness for use based on the system test plan. The goals of System Testing are to verify that the requirements and services are implemented correctly and include usability testing, performance testing, functional testing, and error checking. System Testing will be conducted by LMC and can be witnessed by the ERA Testing Team, IV&V, and QM. System test plans will be generated. The System Test Plan is subject to review by the ERA Testing Team, QM, CM, and IV&V.

**5.2 Development Test (DT) Entrance and Exit Criteria**

The DT entrance criteria include baseline requirements, a completed and approved MTP, and approved test cases and test procedures. Exit criteria or successful completion of DT testing requires that:

- All test documentation has been completed (e.g., test plans and test procedures),
- All test scripts have been executed and Program Trouble Reports are generated for each failure or anomaly,
- All Program Trouble Reports with a severity level of 1 or 2 have been resolved,
- All changes made as a result of trouble reports have been tested,
- The test report has been reviewed and approved, and
- All documentation associated with the ERA system has been updated to reflect changes made during testing.

The Development Contractor is responsible for the documentation associated with DT activities.

**5.3 Acceptance Test (AT) Overview**

AT is a contractual decision point where the ERA system and documentation is handed from the Development Contractor to the ERA PMO for T&E from a user's perspective. AT is functional

**Final**

in nature and will consist of PaT, PAT, OAT, and Installation Acceptance Testing (IAT). Though not part of the formal testing process, two (2) CM audits, Physical Configuration Audit (PCA) and Functional Configuration Audit (FCA) will be performed following the completion of AT. Guidance for conducting audits is found in the *ERA CMP*.

The objectives of AT are to demonstrate that the ERA system is operationally effective and operationally suitable for use, to assess and determine the extent to which the ERA increments have met the A-level requirements, and to determine that NARA's infrastructure is ready to accept the system in a realistic environment before deployment. During OAT, the involvement of NARA users and SMEs will be encouraged to ensure that operational system issues are identified early.

A frequent perception of testing is that the principal goal is program verification; however, several other goals exist. The main AT goals are to:

- Ensure the software satisfies the user requirements and expectations;
- Stress the software at all levels by identifying discrepancies, discovering deficiencies, determining limitations, and verifying interfaces;
- Demonstrate and integrate capabilities by proving the software's ability to handle a wide spectrum of data values and demonstrate requirements satisfaction;
- Demonstrate system usefulness by demonstrating operational capabilities and proving adequacy of documentation; and
- Gaining user acceptance.

AT activity confirms that the software system satisfies all the requirements. **Appendix B, Test Methodology Overview**, shows an overview of the test categories and preparation for verifying the requirements and functionality. AT will not be performed until the software has successfully completed development testing. AT will involve trained users exercising production representative ERA system configurations in a realistic manner to determine the degree to which the system satisfies the stated operational requirements in the *System Requirements Specification (SyRS)*. For Increment 1 Release 1 the LMC test lab will be used for acceptance testing (PaT) since this is an informal activity and the designated operational site will be used for Increment 1 Release 2 acceptance testing (PAT, OAT, and IAT). Specific AT facilities for future deliveries have yet to be identified and established.

AT objectives provide insight into the ERA increment's operational effectiveness and suitability, along with its state of maturity, integration, stability, and readiness for formal acceptance. In determining each increment's readiness to proceed to formal acceptance, through AT, the status of each increment will be judged against the DT exit criteria and the AT entrance criteria.

Throughout AT testing techniques such as stress, regression, performance, and load/volume tests will be used. AT activities will be carried out in accordance with this TSP, the CMP, the QMP, and the ERA Acceptance Test Plans. The development process for creating test scenarios and



**Final**

the content of those test scenarios is shown in **Appendix D, Acceptance Test Scenario Development** and **Appendix E, Acceptance Test Scenario/Test Case Guidelines and Layout**.

The AT plan will be prepared by the ERA Test Team based on the SyRS, as well as on any related design documents. The AT plan is subject to review by QM and CM. **Appendix G, Test and Evaluation Test Plans**, shows a recommended format for Acceptance Test Plans.

AT test results form the basis of the ERA Testing Team's recommendation to the Contracting Officer (CO) and the PD regarding acceptance and deployment of the product.

### **5.3.1 Production Acceptance Tests (PAT)**

TRRs will be conducted prior to PAT. The primary goal of the PAT will be to complete a thorough test to ensure functional robustness of the delivered ERA system. **Appendix J, Acceptance Test Process**, provides an overview of this formal effort. The ERA system will be evaluated for technical accuracy, functionality, correctness, and usability. PAT will be performed in a test environment by the ERA Testing Team and witnessed by QM and IV&V. During acceptance testing (including OAT and IAT) test logs will be kept on the testing being performed – test runs, test status (pass/fail) and issues encountered. The recommended layout for these logs is in **Appendix F, Acceptance Test Logs**. The issues encountered will be entered into the defect management tool. These issues are called Program Trouble Reports (PTRs) and an overview of the lifecycle of these defects/issues is contained in **Appendix I, Program Trouble Report Guidelines**. The Development Contractor will support PAT, as well as OAT and IAT.

### **5.3.2 Operational Acceptance Tests (OAT)**

Operational readiness is the state of system preparedness to perform the missions and functions for which ERA is designed. An ORR will occur prior to OAT.

The primary goal of the OAT will be to ensure that normal production operations sustain the level of required performance documented in the SyRS. ERA system documentation (e.g., Operations Manual, online help, online tutorial) will also be tested (i.e., compare documentation and system keystroke by keystroke) and evaluated for technical accuracy, conformance, and usability. OAT testing will involve a limited number of users at the test facility performing normal business functions. OAT will be performed by a group of NARA functional end users (i.e., SMEs) in conjunction with the ERA Testing Team and can be witnessed by QM and IV&V. The Development Contractor will support OAT, as needed. Results from OAT will be documented and provided to the PD for evaluation.

### **5.3.3 Installation Acceptance Tests (IAT)**

After the first increment of ERA is tested, accepted, and declared operational at the first site, copies of that increment may be produced and installed at other facilities as necessary. Following delivery to the site, each subsequent increment will undergo installation and testing (e.g., communication, interoperability, and connectivity). IAT will be performed at every NARA

**Final**

ERA system installation facility by the ERA Testing Team along with LMC to ensure that the system is installed and functioning properly. LMC will perform an Installation and Checkout (I&C) and then a TRR prior to the ERA Test Team performing testing for the IAT (the ERA Security Team's security testing can be performed with IAT or after IAT is completed). Even though this test activity is intended for installing ERA at additional sites, the ERA system will go through an IAT for the first facility prior to IOC.

**5.4 Installation, Testing, and Control**

Installation, Testing, and Control are all integral elements of the testing environment. All three (3) elements need to work in an effectively cohesive manner so that the ERA testing effort can accurately locate, correct, and track requirements, defects, and enhancements. Since the test environments will emulate a normal operational facility, the procedures for the test environment operation and management are similar. Installation and inspection of the testing environment occurs at the test facilities prior to the start of software testing. The ERA System Test Administrator is responsible for the management, control, scheduling, and maintenance of the testing environment.

LMC will be required to use a CM Tool for checking-in and checking-out such things as source code files, installation scripts, test scripts, and documentation so that revision history information can be monitored and tracked. Migration checklists will be developed to assist in the compilation of components for testing. The checklists detail the execution of migration procedures in sequence throughout the testing levels and provide useful information in the TRR.

The incorporation of application software and test elements into the test environments is highlighted as follows.

- Execute the migration checklist form throughout the migration process. This checklist ensures all elements in the migration from Unit Test to System Test take place. CM, QM, and the ERA Testing Team are responsible for this task.
- Create/modify the needed test database files and tables. LMC and the ERA System Test Administrator coordinate this task.
- Identify and assemble the elements of the application software for testing. CM initiates this task.
- Review and identify any new procedure(s) used for installing the test software. QM review will be performed on new procedure(s) before CM performs its review.
- Conduct or Participate in TRR/ORR. This step is performed prior to moving from one testing level to another. The ERA Test Officer is responsible for this task. The PD or a designated representative chairs the review.

**Final**

- Check the testing environment. This step ensures that the migration is successfully executed in the test environment and everything is ready for System Testing. CM confirms proper operation of the application software. The ERA System Test Administrator checks the database operations.

Once all or part of the ERA system is fielded as an operational system, it will be necessary to have a maintenance test environment or staging area where problems can be replicated and resolved without impact on the production or development environments.

## 5.5 Acceptance Test (AT) Entrance and Exit Criteria

Entrance criteria for AT include successful completion of DT along with baselined CM controlled documentation, software, and hardware.

Upon the completion of AT, an Acceptance Test Report will be prepared by the ERA Test Team. Refer to **Appendix H, Acceptance Test Report** for a recommended test report format.

Only critical or “show-stopper” issues (level 1 or 2 PTRs) found in AT of new functionality or regressions from prior functionality are fixed prior to system acceptance. See **Appendix I, Program Trouble Report Guidelines** for more details on defect levels and tracking.

## 6.0 TEST REPORTING

Issues and results will be documented in testing logs (**Appendix F, Acceptance Test Logs**) and the ERA Issue Tracking database (**Appendix I, Program Trouble Report Guidelines**). All test plans, test procedures, and test cases or other test work products produced by the ERA Test Team will not be considered complete until the work products undergo peer reviews. The test product peer review procedures are documented in the *ERA Peer Review Process (PRP)*.

Problem Reports and Test Reports are required. Problem Reports will be used to document discovered anomalies, deficiencies, or discrepancies. Ideally the problem report also referred to as the issue or bug report (LMC refers to these as PTRs – Program Trouble Reports), captures how to reproduce the problem and an analysis of the error.

Minimally, the problem report will include:

- Tester (name of tester);
- Problem report number (unique identifier assigned to the problem);
- Severity;
- Problem summary/description (briefly describe the problem);
- Steps to reproduce the problem (describe steps, symptoms, and error messages); and
- Module/program/functional area where error occurred (identify where the problem exists).

**Final**

Test reports will be used to document the results of a test and will recommend a course of action based on those results. Test reports for each phase of DT and AT will be produced. When testing for an increment is complete, a test report will be generated. The test report describes the testing performed and evaluates the results.

The test report should include:

- Test Report Identifier (unique identifier assigned to the report);
- Summary of Tests (summarize the evaluation of test items);
- Variances (report any inconsistencies of test items from their design specifications);
- Comprehensiveness Assessment (evaluate comprehensiveness of the testing process);
- Summary of Results (summarize the results of testing);
- Evaluation (provide an overall evaluation of each test item, e.g., impact of any deviation from goals);
- Summary of Activities (summarize major testing activities and events); and
- Approvals (specify names and titles of persons who must approve the report).

## **7.0 INDEPENDENT VERIFICATION AND VALIDATION (IV&V)**

The ERA IV&V Team will be monitoring and reviewing LMC testing activities throughout the ERA system lifecycle. Specific IV&V activities are detailed in the *ERA IV&V Plan (IVVP)*.

## **8.0 TEST AND EVALUATION RESOURCE SUMMARY**

This section describes test and evaluation resources that will be used during the course of the ERA acquisition program.

### **8.1 Test Items**

All testable items that comprise the ERA system will be tested. The versions to be tested will be placed in the appropriate libraries by the Development Contractor. The Development Contractor will also control changes to the versions under test, perform system builds, and notify the ERA Testing Team when new versions are available. All configuration management activities performed by the Development Contractor will be monitored and approved by the ERA PMO CM Specialist.

Specific items (e.g., hardware and software) and associated details within these configuration areas will be addressed in an updated version of this document.

### **8.2 Test Environments and Facilities**

Test environments will be established to perform test preparation, build verification, and unit, integration, system, and acceptance tests prior to deploying the ERA system. The test environments will be separate from the development environment and identical, to the extent possible, to the operational or production environment. During AT, testing will not be conducted

**Final**

using the development environment. All test environments will be approved by the ERA PMO and placed under CM control. Test environment and facilities support resources will be coordinated through the ERA PMO and key representatives at each facility.

To establish the operational test environment, the following steps will be taken.

- **Review and expand technical environment** -The purpose of this step is to ensure that adequate computer hardware and the appropriate system software has been installed and is available through the testing phase.
- **Inspect the test environment** -The purpose of this step is to ensure that an effective test environment has been established for the testing phase. The ERA Systems Engineers, Testing Officer, and CM will review the test environments to make certain that HWCI's needed to support the testing are available and operating properly.
- **Prepare system software to support testing** -The purpose of this step is to ensure that the system software in the test environment is ready for the testing effort. The ERA Testing Team will confirm proper operation of the following types of system software: operating systems, utilities, network software, network management software, Local Access Network (LAN) utilities, and testing tools by physically observing every configurable item in the test environment.

For Increment 1 the Development Contractor's test facility will be used for Release 1 testing and Release 2 acceptance testing will occur at the designated operational site. Before Release 3, it is anticipated as of the development of this document that the Customer Acceptance Test (CAT) lab should be available.

### **8.3 Test Support Equipment**

Specific equipment or tools and associated details will be addressed in an updated version of this document. Various test support equipment may be used during each of the testing phases. Current analysis of test tools for regression and load testing as well as test management are being performed. The updated version of this document will contain a listing and description of the selected tools.

### **8.4 Test Beds**

Specific requirements for test beds (test data) are currently being defined and the data gathered. These are joint activities with LMC with NARA NWME providing assistance in providing the requested data.

### **8.5 Special Requirements**

Federal Acquisition Regulations (FARs) require that the ERA system comply with Section 508 of the Rehabilitation Act of 1973. The Development Contractor has set up a Human Factors lab

**Final**

to conduct reviews and tests for usability and accessibility. LMC will use AccVerify and JAWs (speech tool for the blind). The ERA Test Team will use the IBM Home Page Reader (another speech tool for the blind) and will be reviewing the testing performed in the Human Factors lab to determine that accessibility standards are being met.

**8.6 Staffing and Personnel Training**

Training on the ERA system will be provided, as required to all test and end user personnel prior to the start of AT. In addition, training will be given to all test personnel on how to conduct testing to ensure familiarity with any special requirements, forms, and reporting methods.

As ERA continues to mature in its development, test resource requirements will be reassessed, and refined, with subsequent TSP updates reflecting any changed system concepts or resource requirements.

**9.0 RISKS AND CONTINGENCIES**

A system of ERA's magnitude will not be void of risk and associated mitigations. Similarly, there will be risks that ERA Testing Team will encounter. A solid test management strategy; the involvement of IV&V, QM, and CM; various reviews; and reporting methods will prove beneficial to the ERA Testing Team and may help lessen the impact of realized risks. When risks and contingencies arise, they will be handled using formal risk management as is discussed in the *ERA Risk Management Plan (RKM)*.

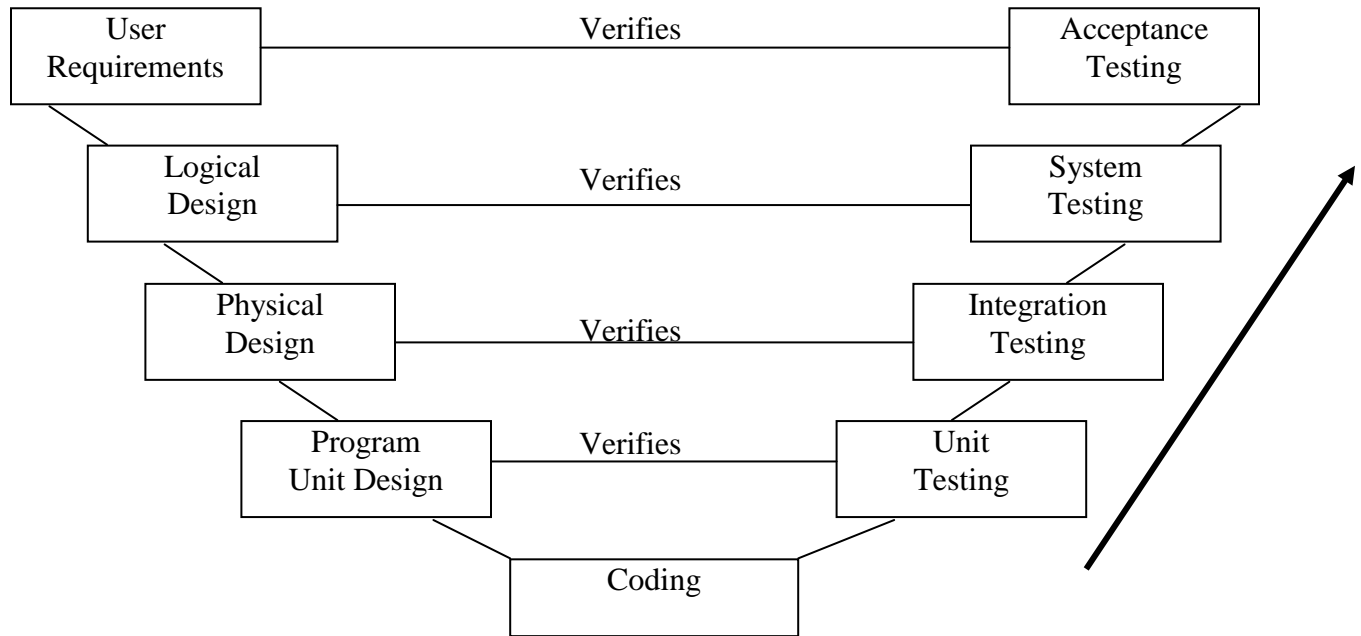
**10.0 PLAN MAINTENANCE**

The ERA Testing Officer is responsible for this plan. As a part of process improvement (e.g., IV&V assessments, lessons learned, QM assessments), the TSP and the overall testing management approach will continue to be adapted for use in future releases of the ERA System. The TSP will be updated as needed to maintain current and sufficient testing management activities and will be maintained under CM control. Any update to the TSP will be controlled by the Configuration Control Board (CCB) as defined in the *ERA CMP*.

Final

## APPENDIX A: Testing Execution

**Figure A-1, ERA Testing Execution**, presents an overview of each testing phase and its relationship to the system design. This is a notional diagram that does not prescribe a classic waterfall development approach for the entire ERA system, but rather is intended to convey levels of testing that may be conducted in an iterative manner.



**Figure A-1: ERA Testing Execution**

**Final**

## **APPENDIX B: Test Methodology Overview**

The testing methodology implemented at ERA incorporates both test strategy and testing tactics. In addition, the testing approach is very flexible to accommodate for changing schedules of software deliveries and functionality. The testing methodology encompasses verification of implemented requirements and functionality, ascertaining usability of the Graphical User Interface (GUI), and reliability of the ERA application, in addition to reviewing Development Contractor documentation, analyzing requirement traceability, monitoring testing activities and test coverage, and providing support to the Development Contractor to facilitate their testing process.

The testing strategy addresses or determines the testing that needs to be performed and the most efficient approaches to accomplish these tests. The test approach includes the following steps:

- Learning and understanding the domain that the system will operate in,
- Learning and understanding the system itself,
- Analyzing requirement traceability and coverage,
- Identifying risks,
- Determining the type of testing that should be performed,
- Determining when testing should occur,
- Developing tests,
- Executing tests,
- Reporting on test results, and
- Re-executing tests to correct problems/issues.

The ERA system is an evolving application and therefore the testing approach is also evolving and flexible. The steps to this approach are iterative and are constantly being applied and improved to ensure that compliance to the requirements and objectives are achieved.

### **Process Development**

Part of a Test Methodology is setting up an acceptance testing process (the details of the test methodology and test process are contained in the following appendices) that should begin early in the development of the application and continually be updated as the system matures. Even though acceptance testing cannot begin until the system has reached a reasonable level of stability and enough capabilities are present to ensure that this will be an effective effort, it is necessary to begin the implementation of acceptance testing as early as possible. Besides allowing time to be fully prepared for the first acceptance test effort, the acceptance test process will have had time to become fully integrated into the project's management practices.

Developing a process is not sufficient no matter how good it is. Everyone associated with the project needs to be cognizant of the process as well as the steps involved in performing the process. Expectations should be set and communicated to personnel to ensure what their roles encompass and the goals required for successful execution of this task. In addition, early development allows time for those involved to provide input on improving the process. The



**Final**

ultimate goal is that no one should be surprised as to what will be done and what the expectations for the delivered application are.

Implementing any process begins with preparation - a building up of a foundation so that acceptance testing can be performed from which feedback and evaluation can be made.

**Preparation**

The acceptance process is continuous, always repeating until completion of the project. Therefore preparation evolves from developing a process, test documentation, and software evaluation to process improvement based on the lessons learned from the last iteration of the process. Preparation for an acceptance test requires an in-place process and completed/updated documentation, which includes an Acceptance Test Plan, Acceptance Criteria, and Test Scenarios.

Preparation activities for continuous acceptance testing contain the following:

- Developing the Process,
- Developing Documentation,
- Defining Acceptance Criteria,
- Acceptance Test Development,
- Evaluating the Application, and
- Lessons Learned.

**Test Categories**

Some of the categories of tests that will be used in verifying the ERA system include but are not limited to the following.

**Functional Testing**

Functional testing ensures that the requirements are properly satisfied by the application. Functional testing is not concerned with how processing occurs but with the results of the processing. The goal of functional testing is to confirm that all of the application's capabilities are present, available, and function properly. This testing will include string and end-to-end tests.

**Requirement Testing**

Requirements testing must verify that the system can perform its functions correctly and that the correctness can be sustained over a period of time. Successfully implementing user requirements is only one (1) aspect of requirements testing. The objectives that need to be addressed are:

- User requirements are implemented,
- Correctness is maintained over extended processing periods, and
- Application processing complies with NARA policies and procedures.

**Final****Visual Inspection**

Throughout the verification of the GUI, visual inspection (demonstration) will also be incorporated with the test conditions. Requirements for the ERA GUI cover how the application should look or what information should be present including the general requirement to meet Section 508.

**Testing Ranges/Boundaries**

For requirements that specify ranges or boundaries for input parameters the testing cannot just cover a single input whose value falls within the stated range. Testing for ERA will include verifying both lower and upper values of the range as well as a value within the range or boundary.

**Negative Testing**

In verifying requirements, not only the conditions of the requirement must be tested but also the opposite of those conditions must be tested. Negative testing involves using the same tests used for verifying that requirements were satisfied but with input parameters that were not indicated in the requirements. Sometimes this type of testing is called error checking.

An example would be that an input field should accept an input of a “Y” or an “N.” The requirement does not state that a different value is an error, so the test determines what the developers implemented. Negative testing determines what happens if a user were to enter a value not indicated by the requirement. If the application were to fail or produce incorrect results then the application needs to be updated to not allow for values outside of the range.

The individuals involved with the development of the requirements and the application become so focused that they might never consider that a user might create input outside of what the application is intended for. Also, developers might be so concerned with the schedule that they might skip putting in checks for inputs because the requirement does not specifically state that this should be done.

Negative testing is a technique used to make sure that what ever a user can enter either by design or accident is addressed, so that the ERA application will be able to handle those situations gracefully and the system will not lock up.

**Defect Verification Testing**

If an issue or problem is encountered during any testing activity a defect will be entered into the Defect Management Tool (see **Appendix I**). Once a defect has been corrected by the development team, and this correction is included in an ERA delivery to NARA, the application can be retested to determine if the update was implemented correctly. If the defect or problem no longer exists, the defect can be closed. However, if the problem has not been corrected, the defect is reassigned back to the Development Contractor point of contact for further investigation and analysis.

**Final****Regression Testing**

One (1) of the aspects of software development that can affect the operational status of an application is the snowballing or cascading effect of making changes to a software system. For example, one (1) segment of the system is developed and thoroughly tested. Then a change is made to another segment, which has a disastrous effect on the thoroughly tested portion. Either the incorrectly implemented change caused a problem, or the change introduced problems in a previously tested segment. Regression testing retests previously tested functions to ensure that they still function properly after a change or update has been made. With the ERA application, regression testing is performed after each new delivery to NARA.

Regression testing involves rerunning tests that have been previously executed to ensure that the same results can be achieved currently as were achieved when the system was last tested. Included in ERA regression testing is executing the test scenarios from previous acceptance test efforts.

**Repeatability Testing**

A precursor to regression testing is repeatability testing. Where regression testing involves verifying that the same results are obtained from the test cases/scenarios between different deliveries of the application, repeatability testing verifies that the same results are generated from the same test case/scenario being executed several times for the same delivery of the application. This effort determines whether or not that the application will consistently generate the same results over a period of time.

Final

## APPENDIX C: Acceptance Criteria Development Guidelines

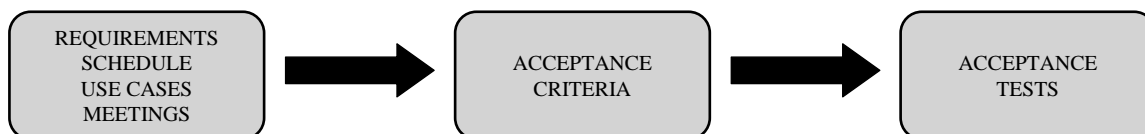
When an iteration of acceptance testing occurs, determinations are made about the application. These determinations are based on a set of guidelines regarding the expectations for the developers and the status of the application. These guidelines are the Acceptance Criteria. These criteria are based on requirements, schedule, contract obligations, and the status of the application and compiled by NARA with input from LMC.

The following is a list of general criteria that can be used to establish more specific criteria.

- Determine Goals
- Demonstrate Specific Capabilities
- Demonstrate Specific Processes
- Review Status of Defects
- Review Waivers
- Review Deliverables/Documents
- Review Status of Requirement Coverage
- Review of Contractor's System Test Results and Output
- Source Code Quality
- Issues from Previous Acceptance Test

The defining/updating of criteria will occur for each acceptance test. Not all criteria deal with testing the application but do involve evaluating the software. During acceptance testing the overall status of the application has to be determined so that NARA and LMC can make informed decisions for the project. Besides testing, the status is determined by reviewing deliverables and documentation, defects, waivers, and requirement coverage, as well as any other guidelines that might be appropriate for each Increment or Release delivery.

Demonstration of specific capabilities and processes goes into determining what type of acceptance testing should be developed. **Figure C-1, Defining Acceptance Criteria & Tests**, shows the connection or flow from the basis for the criteria to the acceptance test themselves. Also included in the Acceptance Criteria are any issues or failures from the previous acceptance test.



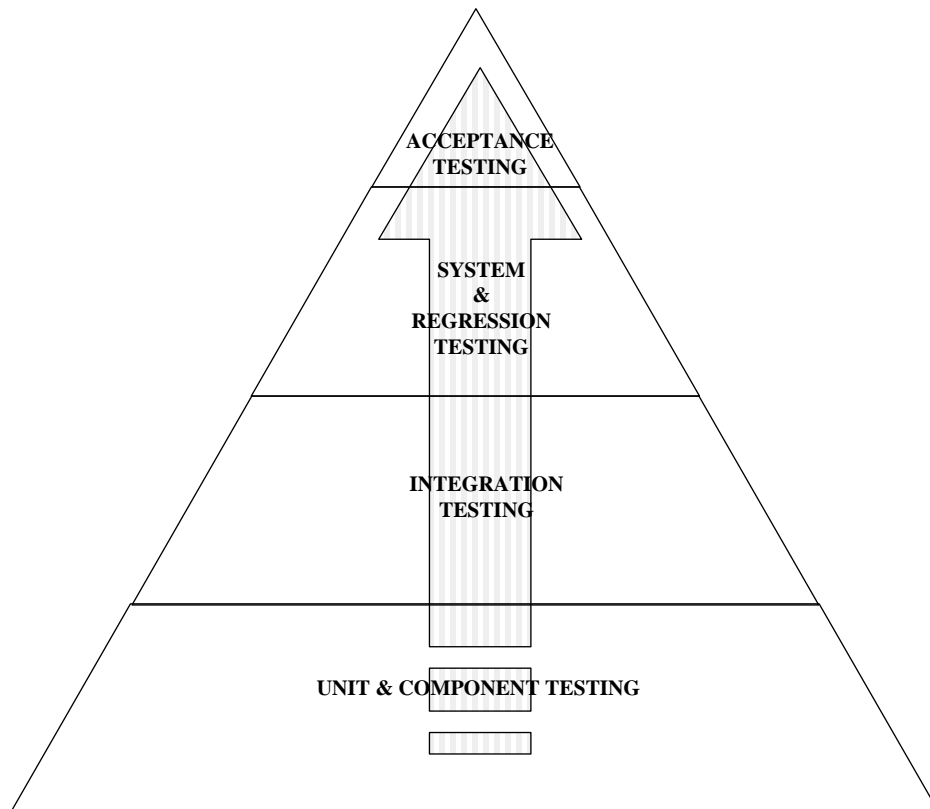
**Figure C-1: Defining Acceptance Criteria & Tests**

Final

## APPENDIX D: Acceptance Test Scenario Development

Acceptance testing is not meant to retest every aspect of an application. Acceptance tests are designed to provide an additional level of confidence to the customer and demonstrate that the application is operating as desired. Remember that other testing is being performed to provide the verification that the application is meeting requirements.

Testing is predicated on an accumulative approach. That is, each phase of testing builds on the last one while focusing on different aspects (**Figure D-1, Testing Foundation** - Regression testing will occur during Acceptance as well as System testing). If acceptance testing were to try to focus on every requirement or logic path, the effort would require additional time and funding.



**Figure D-1: Testing Foundation**

Acceptance tests are designed to demonstrate the most common functional features of a system in a setting that is as close to operational as possible. The specific acceptance tests should be broader use-case scenarios that encompass system work task goals as opposed to specific atomic

**Final**

bits of functionality. Usually these scenarios cover end-to-end processing. With this intent, acceptance tests cover more functionality or aspects of the application than lower level testing.

Acceptance test scenarios should be created with as much detail as possible; a detailed step-by-step how-to of the testing to be performed. Scenarios should explain the tests to be performed and how those tests should be conducted. This is an important area to cover in the documentation. The layout and rules for creating a test scenario helps to ensure that everyone follows the same standards and formats.

As mentioned earlier, the one (1) aspect of determination as to what the acceptance test scenarios should demonstrate is established with the definition of the Acceptance Criteria. **Figure D-2, Acceptance Test Development Approach**, illustrates the entire flow of the scenario development process. Once an acceptance test scenario has successfully demonstrated the specified capability, the scenario will then be used as a regression test to verify that the introduction of new capabilities and functions into the application does not alter previously verified capabilities. This test scenario then does not need to be included in the next acceptance test unless new or updated criteria require an updated version of this test scenario.

The first steps in developing tests are to determine what features need to be developed and how those tests are going to be written. **Appendix E** contains the guidelines and layout for documenting the test/scenario.

Final

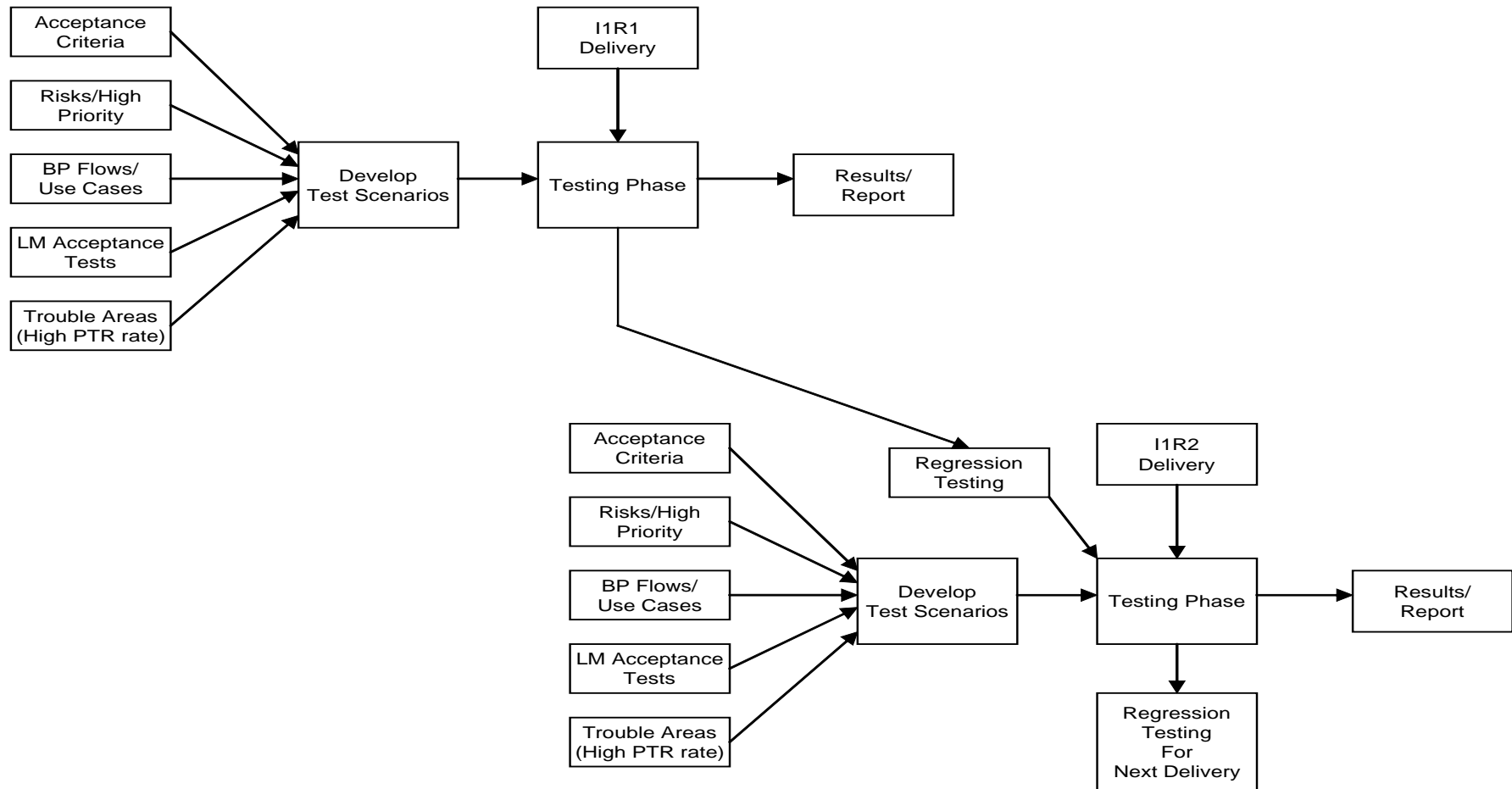


Figure D-2: Acceptance Test Development Approach

Legend: BP- Business Process, LM- Lockheed Martin, PTR – Problem Trouble Report

Final

## **APPENDIX E: Acceptance Test Scenario/Test Case Guidelines and Layout**

### **Acceptance Test Scenario Layout**

The actual creation of a test scenario is a two (2) part process. The first part is the writing of the test description.

For each delivery that goes through acceptance testing, a separate test plan will be developed describing test scenarios that will be performed to verify the functionality of that Increment/Release for acceptance. A test scenario is made up of a series of tests that cover all aspects of a capability. The first step in creating a test scenario is the writing of the test description for each scenario. These test descriptions then go through a peer review and then updated. Each Test Scenario description overview contains:

- 1) Description,
- 2) Objective,
- 3) Requirement Coverage,
- 4) Entrance Criteria,
- 5) Artifacts (produced),
- 6) Basic Test Flow, and
- 7) Post Test Analysis.

After the test descriptions are defined and the system is available, the second part in the test scenario creation process is the development of detailed step-by-step procedures to be performed for each test.

### **Test Description Example:**

#### **Form R1 Scenario**

The Form scenario deals with managing online Forms. This covers online Form definitions and online Form instances. Where the Form definition is the Form layout/template and the Form instance is the Form containing user supplied data (a filled out Form).

This scenario contains two (2) tests that cover the requirements for creating, updating, and deleting both definitions and instances of Forms along with the validation that is part of creation and updating process. The two (2) tests encompass the following capabilities:

- 1) Creating, modifying, and validating online Form definitions
- 2) Accessing, creating, updating, validating, submitting, and approving online Form instances along with deletion of both a Form definition and instance

#### **1) Form Acceptance Test 1 (ERA-R1-Form-001)**



**Final**

Description:	This test demonstrates the basic functionality of creating, updating, and deleting a Form definition. Along with creating a Form definition are the system capabilities of validating the new Form definitions for spelling and declared type attributes as well as checking the form for correctness.
Objective:	Verify creating a Form definition Verify updating a Form definition Verify validation of a Form definition Verify log/audit files
Requirement Coverage:	ERA2.6, ERA2.6.1, ERA2.6.2, ERA2.6.3, ERA2.6.3.1, ERA2.6.3.2, ERA2.6.4
Entrance Criteria:	Portal (User Interface for Forms) Default Business Rules (need to be defined)
Artifacts:	Form definitions Logs/Audit files
Basic Test Flow:	<ol style="list-style-type: none"><li>a) Create Form1 (definition)<ul style="list-style-type: none"><li>- Should cover all basic entry fields</li><li>- Errors should be included in with these types</li><li>- Text should include spelling errors</li><li>- Errors should be included so the Form is not correct (based on business rules?)</li></ul></li><li>b) System should not create Form1 definition and should ask the user to correct before submitting for creation</li><li>c) Make corrections and submit for creation</li><li>d) Verify that Form1 definition was created and saved</li><li>e) Repeat steps a – d for Form2 (definition) but use different format, layout, and entry fields</li><li>f) Retrieve Form1 definition</li><li>g) Update Form1 definition<ul style="list-style-type: none"><li>- Errors should be included in updating entry fields</li><li>- Updated text should include spelling errors</li><li>- Errors should be included in the update so the Form is not correct (based on business rules?)</li></ul></li><li>h) System should not update Form1 definition and should ask the user to correct before submitting for update</li><li>i) Make corrections and submit for update</li></ol>

**Final**

j) Verify that the Form definition was updated and saved

Post Test

Analysis:

Review that the Form definitions were created/updated correctly and that the system performed validation of the content while creating and updating. Also review all log files for errors.

## 2) Form Acceptance Test 2 (ERA-R1-Form-002)

Description:

Creating/updating Form instances (filling out Forms) and deleting a Form definition will be tested. During Form instance creation/updating the verification of the user supplied data is verified.

Objective:

Verify accessing a Form definition  
 Verify listing available Form definitions  
 Verify deleting a Form definition  
 Verify creating a Form Instance (filling out a Form)  
 Verify validate user supplied data  
 Verify submitting a Form  
 Verify approval of a Form  
 Verify log/audit files

Requirement

Coverage:

ERA2.6, ERA2.6.2, ERA2.6.4, ERA2.6.5, ERA2.6.6, ERA2.6.7, ERA2.8.1, ERA2.6.8.2, ERA2.6.8.3, ERA2.6.9

Entrance Criteria:

Portal (User Interface for Forms)  
 Default Form definitions  
 Form1 and Form2 definition from ERA-R1-Form-001  
 Default Business Rules (need to be defined)

Artifacts:

Form Instances  
 Logs/Audit files

Basic Test Flow:

- a) List available default Form definitions
- b) Select Form1 definition and fill it out
  - Errors should be included in the entered data types
  - Out of bound data should be included in fields with data ranges
  - Errors should be included for fields that contain pre-defined constraints
  - Errors should be entered for all user supplied data

**Final**

- c) System should not create Form1 instance and should ask the user to correct before submitting for creation
- d) Make corrections and submit for creation
- e) Verify that Form1 instance was created and saved
- f) Repeat steps b – e for one of the system supplied default Form definitions (the business rules should indicate that once this Form instance is created it should be submitted for Approval)
- g) Verify that the Release 1 default form definitions are present and correct
- h) Delete the Form2 definition
- i) Verify that the Form2 definition was deleted.
- j) Update Form1 instance
  - Errors should be included in the entered data types
  - Out of bound data should be included in fields with data ranges
  - Errors should be included for fields that contain pre-defined constraints
- k) System should not update Form1 instance and should ask the user to correct before submitting for update
- l) Make corrections and submit for update
- m) Verify that Form1 definition was updated
- n) Delete the Form1 instance
- o) Verify that the Form1 instance was deleted
- p) Approve default Form instance from step f

Post Test  
Analysis:

Review that the Form instances were created/updated correctly and that the system performed validation of the content while creating and updating. In addition the selected Form definition should have been deleted. Also, review all log files for errors.

## Acceptance Test Case Guidelines and Layout

Below is the layout of a test case/scenario for the detailed step-by-step procedures. Included in the test description (the first part of the test case) are the entries for the tester's name, the date the test was executed, and the Increment/Release it was tested. As test cases are created more rules for the layout or style can be determined. Currently only the test case name, description, objective, and entrance criteria are included in this layout. Other test descriptions could be included. Not every test case/scenario has to be limited to just these entries. If required, an additional entry that is only needed by a single test case/scenario can be included.

**Final****Test Case Rules:**

- 1) The header row should appear on all pages of the test case/scenario. By selecting “Table Properties” and selecting the “Row” tab you can make the first (or first few lines) repeated. This way if a test case requires several pages the columns will always be titled. (Because the structure of the test case/scenario is made of two (2) tables it is the first rows of the second table that this rule applies to.)
- 2) A row should never break when the end of the page is reached. The entire “Description” or “Comment” should be on the same page and not be split between two (2) pages. By selecting “Table Properties” and selecting the “Row” tab you can make sure the rows do not break.
- 3) Test steps should be created for each action required by a user. The description should not have the user do several things. This could cause the user to miss something. Also by making each action a separate step makes the test procedures more readable.
- 4) Every option selected should be in bold (e.g., click **Apply** button)
- 5) Test scenarios/cases need to be created so that they are very readable/understandable and as detailed as time allows. These test cases are documentation of how testing will be performed and then contain the results of the testing performed. The more information there is (especially using the comments – for giving the tester more information or providing more during testing) the better it is.

Final

**Test Case/Scenario Layout**

<b>Test Scenario/Case Name</b>	
Description:	
Objective:	
Entrance Criteria:	
Tester:	
Increment/ Release:	
Date:	
Pass/Fail:	
Comment:	
<b>Procedure to run the xxxx test</b>	

Step	Requirements	Description	Actual Results	Expected Results	P/F	Comment
1.0	Section Title/Description:					
1.1						
1.2						
2.0						

Final

Step	Requirements	Description	Actual Results	Expected Results	P/F	Comment

**Final**

## **APPENDIX F: Acceptance Test Logs**

During testing, three (3) types of objective evidence are maintained reflecting the results of testing: 1) hardcopy logs and test procedures, 2) defects (PTRs) entered into LMTSS's defect Management Tool, and 3) actual output obtained (could include screen shots, generated reports, log files, or data files).

The logs (templates are below) are completed with information about the test being performed along with any issues/errors encountered during the test. After a test has been executed both the tester and witness sign the log. As the test is proceeding, the test procedures steps are annotated as pass or fail along with any appropriate comment about the actual result for the step. These documents are also signed and dated by both the tester and witness. Copies of these signed documents are included with the distribution of the Acceptance Test Report. The originals are given to the ERA Configuration Manager.

Any defects identified during testing are entered into the defect Management Tool (see **Appendix I**) and the defect ID is added to the issue in the log files.

The output from testing is downloaded and also given to the ERA Configuration Manager.

Once all testing and evaluation have been completed, the ERA Test Team creates a report on the results and the status of the Acceptance Test. The outline for this document is in **Appendix H**.

### **Acceptance Test Scenario Log**

For each execution of an Acceptance Test Scenario a test log is created using the following template:

Final

<b>1.0</b>		
<b>Date:</b> XX/XX/XXXX – XX/XX/XXXX <b>Location:</b> XXXX <b>Increment/Release:</b> I#R#	<b>Test Executors:</b> Print Name _____ Sign Name _____	<b>Test Witnesses:</b> Print Name _____ Sign Name _____
<b>Scenario Name:</b>	<b>Scenario Description:</b>	
<b>Overall Status (Pass/Fail):</b>	<b>Remarks:</b>	
<b>#</b>	<b>Acceptability Criteria</b>	<b>Status (Pass/Fail)</b>
<b>1</b>		
<b>2</b>		
<b>3</b>		
<b>4</b>		
<b>#</b>	<b>Incident</b>	
<b>1</b>		
<b>2</b>		
<b>3</b>		
<b>4</b>		
<b>5</b>		
<b>6</b>		
<b>7</b>		



Final

### Acceptance Test Summary Log

During post acceptance testing evaluation a summary of all the logs is compiled using the following template:

Date:

Location: CAT Lab

#	Test Scenario	Testers	Witnesses	Configuration	Issues (number - severity)	Evaluation (P/F)
1.0	Acceptance Test 1					
2.0	Acceptance Test 2					
3.0	Acceptance Test 3					

Final

## APPENDIX G: Test & Evaluation Test Plans

### Format of Acceptance Test Plans

*Acceptance Test Plans will be prepared in accordance with IEEE 829 Std-1998, Standard for Software Test Documentation. Refer to the standard for content requirements.*

- 1) Test plan identifier
- 2) Introduction
- 3) Test items
- 4) Features to be tested
- 5) Features not to be tested
- 6) Approach
- 7) Item pass/fail criteria
- 8) Suspension criteria and resumption
- 9) Test deliverables
- 10) Testing tasks
  - Test Descriptions
  - Detailed Test Procedures
- 11) Environmental needs
- 12) Responsibilities
- 13) Staffing and training needs
- 14) Schedule
- 15) Risks and contingencies
- 16) Approvals

Final

## APPENDIX H: Acceptance Test Report

### Content and Format of Acceptance Test Report

*Acceptance Test Reports will be prepared in accordance with IEEE 829 Std-1998, Standard for Software Test Documentation. Refer to the standard for content requirements.*

- 1) Test Report Identifier
- 2) Introduction
  - Scope
  - Definitions, Acronyms, and Abbreviations
  - Objectives
  - Roles and Responsibilities
- 3) Summary of Tests
- 4) Variances
- 5) Comprehensiveness Assessment  
(Includes Requirements Coverage)
- 6) Summary of Results
- 7) Evaluation
- 8) Summary of Activities  
(Includes Schedule)
- 9) Approvals

## Final

## APPENDIX I: Program Trouble Report Guidelines

### Overview

The LMC process and tool for tracking and reporting the status of defects will be used for defects identified during Acceptance Testing. These defects are referred to as PTRs. The generated PTRs are entered into the Borland Star Team tool, which is being used as the PTR lifecycle software asset management tool. The information in this section was obtained from CDRL 56 (System Integration Plan).

### Defect Management Lifecycle and Goals

In order to resolve issues PTRs are created for:

- Documenting defects so that the situations can be resolved in a timely manner,
- Assigning responsibility for defect investigation and fixes,
- Ensuring that the defect is tracked until it has been completely fixed,
- Capturing defects from all phases of test which can be used for metrics and reporting purposes, and
- Tracking all defects until resolved.

### Defect Tracking During Test Phases

Each PTR will go through a “lifecycle” that tracks the various states as well as the status; the most important being creation and verification. The Borland Star Team tracking tool will contain information regarding the history of the PTR, including when it is transitioned to different states, what each state means, and the organization and resource responsible for actions at that point.

### Defect Categories and Severity

PTR defect Categories are rank ordered from one (1) through five (5) with Category One (1) being the most important. Each of these numeric Categories has an associated Severity which describes the extent and importance of the problem. Categories and their associated Severities are listed below:

**Category 1 (Test Critical)** – Major test case(s) are blocked from successfully executing without an available workaround. During the conduct of dry-runs and formal test executions, a Test Critical PTR should have an acceptable workaround or fix within 72 hours.

**Category 2 (High)** – Significant degradation in major operational functions or performance/stability. No workaround available. During the conduct of dry-runs and formal test executions, a High PTR requires a fix or acceptable workaround within 10 Days.

**Category 3 (Medium)** – Workaround available for total or partial loss of major operational functions. Marginal impact to major operational functions.

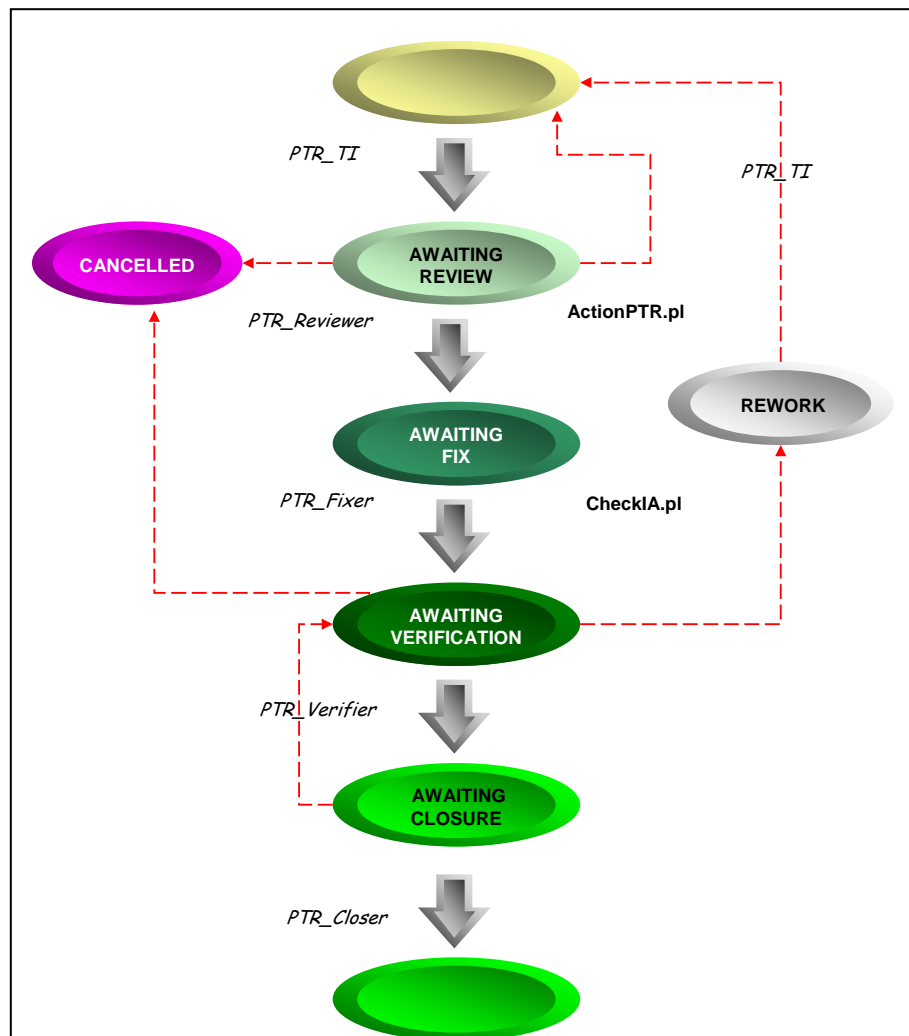
**Category 4 (Low)** – A system problem that does not prohibit the successful completion of a test. No significantly noticeable impact to system operations.

**Category 5 – Minor** annoyance or imperfection.

Final

### PTR Life Cycle Task Flow

A PTR moves to different states throughout its lifecycle. Some of these movements occur through manual transitions and some through automatic transitions depending on triggers within the Borland Star Tool. **Figure I-1, PTR Defect Lifecycle Task Flow**, illustrates the lifecycle of a PTR.



**Figure I-1: PTR Defect Life Cycle Task Flow**

**Open State:** Initial state of the PTR

PTR that has been created and submitted.

**Awaiting Technical Investigation (TI) State:** PTR is pending completion of technical investigation.

The PTR is currently being assessed by the supporting Integrated Product Team (IPT) leads where it will be assigned to the correct subsystem.

**Final**

**Awaiting Review State:** *PTR is pending approval to be implemented after technical investigation or PTR may be cancelled.*

The PTR is currently being reviewed and is awaiting approval to implement. This can generate three (3) possibilities.

1. After being reviewed it is determined that the PTR is invalid therefore it is moved to the cancelled state.
2. After being reviewed it is determined that the PTR is valid therefore it is moved to Awaiting fix.
3. After being reviewed it is determined that more information is required and it is moved back to the awaiting TI state. All corresponding Work Orders are returned to the impact assessment state.

**Cancelled State:** *PTR is found to be invalid.*

The PTR is not valid therefore is cancelled. There are many options to cancel a PTR such as duplicate, user error, incorrect data, etc.

**Awaiting Fix State:** *The problem is accepted by the owner, is being analyzed by Development or other originating department and is in the process of being fixed.*

Once the PTR is moved to awaiting fix, all the work orders in the impacted status are moved to work authorized. The affected supporting IPT corrects the defect. The solution has been identified, fixed, implemented, and the fix is available for deployment. Once deployment is completed the PTR is moved to the awaiting verification status.

**Awaiting Verification State:** *PTR fix is complete and the solution is to be verified by the test team.*

The solution has been implemented and is ready to be tested. After testing the solution there are two (2) categories.

1. It is determined that the solution did not fix the problem and the PTR is sent back to rework. All impacted work orders are returned to impact assessment and the PTR is moved back to awaiting TI.
2. It is determined that the PTR has corrected the original defect and the PTR is moved to awaiting closure, once there is approval between I&T test lead and TAC.

**Awaiting Closure State:** *PTR is reviewed by QA and TAC.*

The QA group reviews the PTR for accuracy and completeness. After revision there can be two (2) categories:

1. It is determined that the PTR needs to be corrected and it is returned to the awaiting verification state.
2. It is determined that the PTR is complete and accurate and is moved to the PTR Closed state.

**Rework State:** *Testing of the solution was not successful and the original problem still exists.*

**PTR Closed State:** *Defect has been corrected and tested.*

Final

## APPENDIX J: Acceptance Test Process

### 1.0 INTRODUCTION

The AT process defines quality control activities that are identified to ensure that the integrated ERA system meets user requirements and performs, as per the “design” in the system documentation. The AT process sets the stage for the *ERA Acceptance Test Plans (ATPs)* that are to be developed and other work products (e.g., test procedures, test cases). Initiation of the AT activities begins with successful completion of system testing by LMC.

### 2.0 PURPOSE

The purpose of the ERA AT process is to describe the methodology, goals, objectives, and strategy that will be employed to accomplish AT for the ERA system. It also provides guidance to all parties involved. The AT process is used as the primary means for evaluating deliverables and describing the AT methodology. This process details the ERA AT activities, deliverables, and entry/exit criteria.

### 3.0 OBJECTIVES AND GOALS

The primary objective and goal of the AT process is to verify successful execution of the ERA operational characteristics and interfaces for a range of customer loads and configurations. The AT process objectives and goals are the following.

- AT activities are planned and scheduled
- LMC deliverables and activities adhere to the applicable standards, procedures, and requirements
- Affected groups and individuals are informed of their roles and responsibilities as well as the AT activities and results
- AT techniques, criteria, and methods are established
- Each integration Configuration Item (CI) works properly in the system test environment
- Merged integration CIs successfully work together
- Users can recognize system errors and know how to handle them
- The system can survive any kind of invalid input or invalid processing actions and exhibits graceful terminations (e.g., informs user via messages, does not lock, and saves process before ending)
- Non-conformances, anomalies, and defects are documented and addressed

### 4.0 METHODOLOGY

ERA AT will focus on the successful execution of ERA business processes as well as the verification of the A-Level requirements Functional, Operational, and Performance requirements as discussed in the *SyRS*. The ERA Testing Officer and Acceptance Testing Team may witness lower level tests and use data from these tests as analytical data to verify the A-Level requirements.

## Final

The most effective way to perform AT is to evaluate end-to-end operational functionality of a system or component. The objective is to invoke the overall functionality and not necessarily to focus on all the minute or individual system features that were the focus during development testing (unit, integration, system) that occurred prior to AT. The primary testing method to be employed for ERA AT is operational scenarios. Operational scenarios allow the tester/user to create a sequence of events as they would occur in NARA business situations. For example, an AT scenario may take a records processor through administering an accessioning, verification, arrangement, and description of electronic records. The *ERA Concept of Operations (ConOps)* and *ERA Use Case Document* offer good samples of archival scenarios.

### 4.1 Approach

To accomplish the AT objectives and goals in the timeframe specified in the ERA schedule, the following strategy will be employed.

1. Review all LMC system test cases.
2. Review and analyze executed tests and open problem reports from ERA development testing to determine all required test cases and test scenarios, including the sequencing of the tests for future test bed/data setup, volume testing, and regression testing as needed.
3. Develop AT test cases and test scenarios.
  - Identify each AT test case by a descriptive name and number according to the accepted naming conventions. Document functional requirements to be verified.
  - Describe test objectives for each test case relative to the system requirements and module definition.
  - Describe test initialization requirements.
  - Develop required scripts for validating data, regression testing, and any other test scenarios.
  - Conduct a dry run of the AT test cases to ensure that scripts work and give expected results.
  - Provide the estimated duration of the tests based on successful dry runs already completed.
4. Complete the Pre-Acceptance Checklist. **Table J-1, Pre-Acceptance Checklist**, provides guidance in preparation for the acceptance activities. The Pre-Acceptance Checklist helps to ensure that all necessary preparatory activities have been completed and that required operating documents were developed and approved.



Final

CHECKLIST	COMMENTS
<input type="checkbox"/> An <i>ERA ATP</i> has been reviewed to ensure the plan reflects the current version of the software and system requirements.	
<input type="checkbox"/> The <i>ERA ATP</i> is approved by the ERA PMO and other project stakeholders prior to conducting any acceptance tests.	
<input type="checkbox"/> Support staff has been identified for the project.	
<input type="checkbox"/> Copies of ERA system documents have been provided to the support staff.	
<input type="checkbox"/> The security checklist has been completed by the ERA Security Officer and forwarded to each installation site, if applicable.	
<input type="checkbox"/> The approved <i>ERA ATP</i> is placed under configuration management.	
<input type="checkbox"/> Operational procedures and other test materials have been provided to the Acceptance Testing Team prior to the start of acceptance test training.	
<input type="checkbox"/> The Acceptance Testing Team is trained, if necessary.	
<input type="checkbox"/> The <i>ERA Program Management Plan (PMP)</i> and <i>Work Breakdown Structure (WBS)</i> have been updated to include any revised estimates of resources, cost, and schedule.	
<input type="checkbox"/> A TRR has been conducted indicating exit from system testing.	
<input type="checkbox"/> The acceptance test environment has been properly setup and configured.	

**Table J-1: Pre-Acceptance Checklist**

- Execute the test procedures as written. Due to issues such as test dependencies (i.e., testing sequence and precedence) and constraints, no deviations will be allowed in the actual procedure, unless specifically approved by the ERA Testing Officer. The ERA

**Final**

Testing Officer will adjust the test procedures to reflect the change, or to support a work-around solution to facilitate further testing. Tests will be executed by the ERA Acceptance Testing Team and monitored by the ERA Testing Officer and other observers (e.g., IV&V) for AT integrity.

6. At the conclusion of each test procedure, the tester assigns a PASS/FAIL to the test activity based solely on the expected results. If the expected results are not obtained, the ERA Testing Officer will immediately be advised and a problem report generated.
7. Complete the Acceptance Checklist. **Table J-2, Acceptance Checklist**, provides guidance for the acceptance activities. The Acceptance Checklist helps to ensure that all necessary activities have been completed and that required operating documents were developed and approved.

CHECKLIST	COMMENTS
<input type="checkbox"/> User Training has been conducted.	
<input type="checkbox"/> Software testing tools have been calibrated, if necessary.	
<input type="checkbox"/> Software configuration has been verified.	
<input type="checkbox"/> At each installation site, the facility has been inspected to ensure that the site preparation is complete and in accordance with the installation plan.	
<input type="checkbox"/> The installation has been coordinated with the ERA PMO, operations staff, support staff, and other affected organizations.	
<input type="checkbox"/> Any necessary modifications to the physical installation environment are complete.	
<input type="checkbox"/> The hardware has been inventoried and tested.	
<input type="checkbox"/> If the software product requires an initial data load or data conversion, the tested programs are installed and executed.	
<input type="checkbox"/> The software product has been installed on the hardware platform and tested according to the installation plan.	
<input type="checkbox"/> Problems and corrective actions are documented.	

## Final

CHECKLIST	COMMENTS
<input type="checkbox"/> All hardware and software was retested after maintenance or replacement.	
<input type="checkbox"/> A copy of all installation test materials has been placed under configuration management.	
<input type="checkbox"/> A copy of training materials has been submitted to the ERA Training Officer for review and approval and placed under configuration management.	
<input type="checkbox"/> The test environment is subject to strict, formal configuration control to maintain the stability of the environment and to assure the validity of all tests.	
<input type="checkbox"/> All acceptance test activities have been coordinated with the Testing Officer, NARA SMEs, operations staff and other affected organizations.	
<input type="checkbox"/> Acceptance testing has been conducted in an environment that functions like the production environment using acceptance test data and test procedures established in the <i>ERA ATP</i> .	
<input type="checkbox"/> All tests have been executed correctly within the acceptable threshold for defects.	
<input type="checkbox"/> Any tests that failed have been documented, corrected, and retested.	
<input type="checkbox"/> An Acceptance Test Report has been created.	
<input type="checkbox"/> A copy of all acceptance test materials has been placed under configuration management.	
<input type="checkbox"/> At the completion of acceptance testing an ORR is conducted.	
<input type="checkbox"/> After successful completion of ORR, the updated system documentation was established as a new baseline.	
<input type="checkbox"/> Complete operating documentation describing the ERA system has been approved and delivered.	

## Final

CHECKLIST	COMMENTS
<input type="checkbox"/> This acceptance checklist is approved and completed.	
<input type="checkbox"/> A formal written acceptance of the software product is generated by the ERA PD to verify that the software product is accepted and ready for production.	
<input type="checkbox"/> The software product is deployed to a full operational status according to the <i>ERA Deployment Plan</i> that is to be developed.	
<input type="checkbox"/> Stress and other operational tests have been conducted.	
<input type="checkbox"/> Any training activities are completed.	
<input type="checkbox"/> The maintenance support has begun as planned.	
<input type="checkbox"/> At the end of deployment, a formal transfer of all responsibilities to the support staff is conducted.	
<input type="checkbox"/> A formal announcement of deployment to production has been done.	
<input type="checkbox"/> Access rules have been modified to provide access to the ERA system by the support staff and remove the Acceptance Testing Team and other temporary user access from further access to the system.	
<input type="checkbox"/> All project file materials, operating documents, and other pertinent system materials have been turned over to the maintenance staff.	

Table J-2: Acceptance Checklist

8. Defects generated as a result of AT will be distributed to LMC for investigation and corrective action. At this point, the developers' responsibilities include:
- Check out module components using software version control tool,
  - Correct and unit test software modules,
  - Check in module components using the software version control tool, and
  - Notify the Configuration Management (CM) Specialist to migrate the module components to the appropriate test sites and compile them for execution.

**Final**

9. Retest the effected application components by executing the failed AT test case(s). Also perform regression testing of all affected modules during the retest.

**4.2 Verification**

Each ERA requirement is analyzed to determine how it can be confirmed during AT. The four (4) verification methods used during the AT process include the following, per *IEEE Std. 610.12-1990, Software Engineering Terminology*.

1. **Test** is an activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component.
2. **Demonstration** is a dynamic analysis technique that relies on observation of system or component behavior during execution, without need for post-execution analysis, to detect errors, violations of development standards, and other problems.
3. **Analysis** is the process of evaluating a system or component based on its form, structure, content, or documentation.
4. **Inspection** is a static analysis technique that relies on visual examination of development products to detect errors, violations of development standards, and other problems. Types include code inspection and design inspection.

In some cases, more than one (1) verification method may need to be applied for an adequate evaluation.

**4.3 AT Protocol**

The ERA Acceptance Testing Team has the overall responsibility for the preparation, execution, and review of AT activities. As stated earlier, AT begins when LMC successfully completes system testing. Once LMC completes system testing and conducts the Test Readiness Review (TRR), they will coordinate the delivery of test items (e.g., code, documentation, test scripts) with the ERA Contracting Officer (CO)/ Contracting Officer's Representative (COR), the ERA Configuration Manager, and the ERA Testing Officer. The test items are transferred to the ERA CM library as described in the *ERA Configuration Management Plan (CMP)*. After the delivery of test items to the ERA PMO has been completed, the CO/COR is given notice by the ERA Testing Officer at least two (2) weeks in advance of the planned commencement for AT. Attendance to witness the tests is at the option of the CO/COR.

To alleviate sudden revelations during AT, a weekly status meeting is conducted by the ERA Testing Officer. The ERA Testing Officer meets with the Acceptance Testing Team to review the testing activities for the week, disclose notable non-conformances (i.e., Severity 1 (Fatal) and Severity 2 (Serious)), and prepare for the next week of tests. Prior to the start of each meeting, a log report is generated that will inventory the number of test executed and the recorded number of test failures. In addition, reports will be generated from the defect tracking system. The minutes of the meeting will record any significant discussions and decisions.

### **Final**

Ultimately, after the AT, the ERA Testing Officer along with the ERA PMO and the user representatives decides that even though there may be open AT defects remaining, there is an acceptable risk level for migrating the system into the production environment.

## **4.4 AT Entry Criteria**

The AT entry criteria requires the following actions be completed before acceptance testing activities can begin.

- The ERA system has been successfully installed and migrated to the test environment(s)
- All modules have been successfully executed at least once in system test cases
- The TRR has been conducted
- Hardware/Software is available for test configurations
- All Severity 1 (Fatal) and Severity 2 (Serious) defects from the system test have been documented, fixed, verified, and closed in the defect tracking database
- The checklist in **Table J-1, Pre-Acceptance Checklist**, is completed

## **4.5 Evaluation and Retest**

Once the ERA system is in the AT process, the retest of modifications or corrections to the ERA system involves ensuring that system alterations work properly and do not cause other deficiencies elsewhere in the system. Once the AT (which includes regression tests of the ERA system) is done, the ERA Acceptance Testing Team provides a recommendation to the ERA Testing Officer to accept or reject the modifications.

The ERA Testing Officer is responsible for coordinating the review of all test cases and test results, as well as resolving conflicts between the ERA Acceptance Testing Team and the developers concerning retesting.

## **5.0 MANAGEMENT**

Management of the AT process includes all the tasks necessary to manage the personnel and to administer tasking and deliverables of the AT. A non-inclusive list of the tasks includes:

- Preparing and updating schedule for AT activities;
- Monitoring status, completeness, and completion of tasks;
- Identifying resources needed for AT testing activities;
- Documenting AT status and progress;
- Reviewing and evaluating the quality of test cases;
- Reviewing and evaluating the quality of test execution;
- Reviewing and evaluating the results of text execution;
- Monitoring the status of problem report resolution; and
- Monitoring the status of testing and retest activities.

Final

5.1 Test Control

AT activities will be controlled through adherence to the AT process, management oversight by the ERA Testing Officer. The execution of tests and discovery and resolution of problems will be tracked in an electronic format. In addition, the work products developed during AT such as the testing plans, test cases, test data, test reports, and meeting minutes will be preserved in soft and hard copies and placed under CM control.

5.2 Test Results Record Keeping

The results of executing each Acceptance Test Scenario/Case will be stored in a Test Log associated with the PTRs in the defects tracking system. The test log tracks the execution of each test case by Test Case ID, Test Phase, and Test Record or Sequence Number. Each log will record either a failed execution of a test case or the successful execution of a test case. In addition, failed test cases will result in the generation of a corresponding defect record. The LMC defect tracking tool will track the status of problems and their resolution and facilitate the flow of pertinent information between all the parties responsible for configuration management, software development, documentation, and AT.

6.0 ACCEPTANCE PROCESS STRUCTURE

The Acceptance process structure comprises the steps for acceptance. **Figure J-1, Acceptance Process Diagram**, depicts the seven (7) steps with their respective inputs (↓) and outputs (↑).

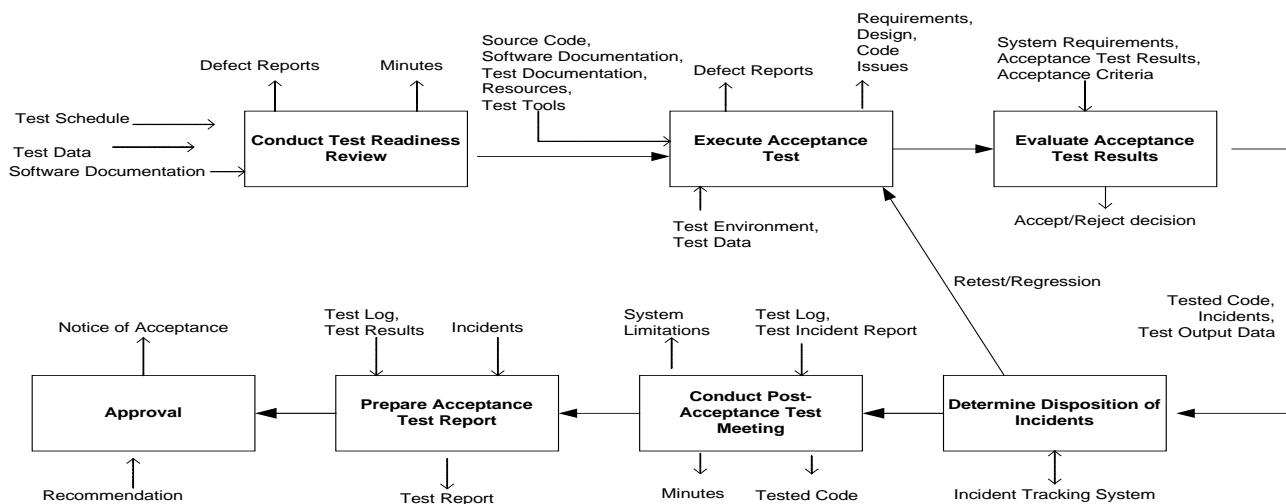


Figure J-1: Acceptance Process Diagram

6.1 Steps for Acceptance Process

The seven (7) steps for the acceptance process follow.

**Final**

- **Conduct Test Readiness Review (TRR).** At each TRR LMC will describe the testing performed, disclose system testing results, and identify areas of risk. Often the information to be handed off for acceptance is contained in a turnover package. Additional information on TRRs is provided in the *ERA Quality Management Process (QMP)* and the *ERA TSP*.

Inputs and outputs for this step include:

Inputs: Test schedule, Test data, Software documentation, Anomalies from prior testing, Acceptance Criteria

Outputs: Defect reports, Minutes

- **Execute Acceptance Test.** Acceptance tests are conducted for every release and increment. The ERA PMO and NARA SMEs will create separate ATPs for each release and increment.

Inputs and outputs for this step include:

Inputs: Source code, Software documentation, Test documentation (i.e., Acceptance Test Plan, Design, Cases, Procedures), Resources, Test tools, Test data, Test environment

Outputs: Logs, Defect reports

- **Evaluate Acceptance Test Results.** Non-conformances are tracked via a defect tracking system. The defect tracking system is used to capture defects, anomalies, discrepancies, and corrective actions. The tool allows control of the identified non-conformances. Anomalies and defects that cannot be resolved within the acceptance process are addressed by the ERA PD.

If upon evaluation the release/increment has passed its AT, the release/increment is deemed accepted. The ERA PMO can then proceed with the deployment process for the release/increment.

If after evaluation the Acceptance Testing Team finds that the release/increment has failed its specified AT, the ERA Testing Officer is notified and provided with a description of deficiencies. The deficiencies are handed off to the CO/COR. Upon direction from the CO/COR, LMC then proceeds with further development and refinement to produce a revised version of the release. The revised release is retested and re-evaluated.

Inputs and outputs for this step include:



**Final**

Inputs: System requirements, Acceptance test results, Acceptance criteria

Outputs: Accept/reject decision

- **Determine Disposition of Incidents.** The ERA Testing Officer along with others in the ERA PMO (e.g., Engineering Review Board (ERB)) will decide the nature of the defect.

Inputs and outputs for this step include:

Inputs: Tested code, Defects, Defect Tracking System, Test output data

Outputs: Defect Status Report from Defect Tracking System

- **Conduct Post Acceptance Test Meeting.** The ERA PMO and NARA SMEs will conduct the Post Acceptance Test Meeting.

Inputs and outputs for this step include:

Inputs: Test log, Test incident report

Outputs: Minutes, Tested code

- **Prepare Acceptance Test Report.** The ERA Testing Officer will create the Acceptance Test Report and submit it to the ERA PD.

Inputs and outputs for this step include:

Inputs: Incidents, Test log, Test results

Outputs: Acceptance Test Report

- **Approval.** The ERA PD receives recommendation on the acceptance or rejection of the release/increment based on the Acceptance Test Report.

Inputs and outputs for this step include:

Inputs: Recommendation

Outputs: Notice of Acceptance

## Final

## 6.2 AT Exit Criteria

The AT exit criteria requires the following actions be completed before acceptance testing activities can end.

- 100% of AT test cases attempted; 95% concluded successfully.
- All Severity 1 (Fatal) and Severity 2 (Serious) AT defects have been documented, fixed, verified, and closed in defect tracking database.
- The checklist in **Table J-2, Acceptance Checklist**, is completed.

## 7.0 ORGANIZATIONAL INTERFACES

The following subsections identify each organization with which the ERA Acceptance Testing Team interfaces.

### 7.1 Development Contractor

The ERA Acceptance Testing Team monitors and evaluates development test activities that occur at releases and increments. The testing team reviews release and increment test documentation, witnesses testing, analyzes test results, and reviews test reports. These activities allow the testing team to gain an overall understanding of the software and potential risk areas that may warrant additional attention when the software is promoted to the next level of testing.

### 7.2 Configuration Management (CM)

The ERA Acceptance Testing Team coordinates with the CM Specialist in identifying all acceptance test items placed under configuration control. These include, but are not limited to, test plans and procedures, test scripts, test data sets, and the software and hardware used to perform acceptance testing. The CM Specialist conducts configuration audits and supports the testing team.

### 7.3 Quality Management (QM)

The ERA Acceptance Testing Team communicates with the Quality Management (QM) Specialist concerning acceptance test plans, procedures, schedules, and for the purpose of providing QM with information regarding acceptance testing activities and issues. QM may witness acceptance tests and conducts quality audits.

### 7.4 Independent Verification and Validation (IV&V)

The ERA Acceptance Testing Team ensures that Independent Verification and Validation (IV&V) has access to all ERA test activities and technical information (e.g., test documentation) for review and analysis.

**Final****8.0 REPORTING**

The ERA Testing Officer generates various reports for the ERA PD. Reports such as Nonconformance, Progress, and Test Metrics cover AT activities and their subsequent results.

**8.1 Program Trouble Reports**

When an AT test procedure completes successfully, the results are reported as a success (PASS). However, when an AT test procedure is not successful it is considered a nonconformance. Non-conformances come in many forms – anomalies, discrepancies, problems, incidents, and defects. No matter which term is used, each connotes noncompliance with expected results and the inability to meet requirements. **Appendix I, Program Trouble Report Guidelines**, presents more detail on the handling of defects.

**8.2 Progress**

The Acceptance Testing Team provides testing activity status reports informing the Testing Officer of such matters as, but not limited to, the number of unresolved, resolved, and deferred problems. The team also provides status on the preparation and update of the AT test matrix.

**8.3 Test Metrics**

The ERA Testing Officer is responsible for maintaining the following AT Test statistics and reporting them at the ERA Project Status Meeting. The statistics include, but are not limited to:

- Total Number of Test Cases Developed,
- Total Number of Test Cases Executed,
- Total Number of Test Case Failures,
- Total Number of Problem Reports Generated,
- Total Number of Open Problem Reports, and
- Total Number of Open Problem Reports by Priority.

**Final****APPENDIX K: Increment 1 Acceptance Test Overview**

Increment 1 will consist of three (3) Releases, which will go through acceptance testing. The first Release will deal with infrastructure and only go through an informal acceptance test effort. The second and third Releases will go through a formal test effort which includes PAT, OAT, and IAT. This appendix presents the overview of the test approach, test flow, test coverage, test tasks and activities, as well as a table showing the milestones for Increment 1 testing.

**Release 1 Acceptance Test Approach****Product acceptance Test (PaT)**

- a. Will be conducted over a three (3) week period
- b. First seven (7) Business Days: Perform a subset of LM delivered system tests (delivered in Acceptance Test Procedure CDRL)
- c. Second eight (8) Business Days: Perform NARA ERA PMO developed acceptance tests
- d. Will be conducted on the LM test lab

**Release 2 Acceptance Test Approach****A. Product Acceptance Test (PAT)**

- a. Will be conducted over a four (4) week period
- b. Week 1: Regression testing of a subset of acceptance tests from Release 1
- c. Week 2: Perform a subset of LM delivered system tests (delivered in Acceptance Test Procedure CDRL)
- d. Week 3 – 4: Perform NARA ERA PMO developed acceptance tests
- e. LM external simulator will be used to simulate external agencies for data transfers
- f. Will be conducted on the operational site

**B. Operational Acceptance Test (OAT)**

- a. Will be conducted over a one (1) week period
- b. A subset of Release 2 acceptance tests will be used
- c. NARA users will participate
- d. LM external simulator will be used to simulate external agencies for data transfers
- e. Will be conducted on the operational site

**C. Installation Acceptance Test (IAT)**

- a. Performed after LM Installation and Checkout (I&C)
- b. Will be conducted over a one (1) week period
- c. One (1) or two (2) external agencies will participate by transferring data to be ingested
- d. Support C&A testing
- e. Will be conducted on the operational site

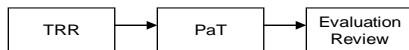
**Release 3 Acceptance Test Approach****A. Product Acceptance Test (PAT)**

- a. Will be conducted over a four (4) week period

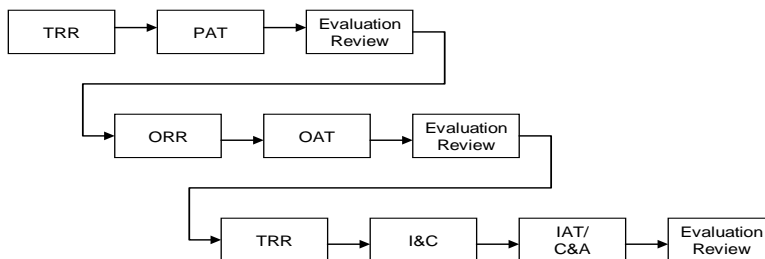
**Final**

- b. Week 1: Regression testing of a subset of acceptance tests from Release 2
  - c. Week 2: Perform a subset of LM delivered system tests (delivered in Acceptance Test Procedure CDRL)
  - d. Week 3 – 4: Perform NARA ERA PMO developed acceptance tests
  - e. LM external simulator will be used to simulate external agencies for data transfers
  - f. Will be conducted in the Customer Acceptance Lab
- B. Operational Acceptance Test (OAT)
- a. Will be conducted over a one (1) week period
  - b. A subset of Release 3 acceptance tests will be used
  - c. NARA users will participate
  - d. LM external simulator will be used to simulate external agencies for data transfers
  - e. Will be conducted in the Customer Acceptance Lab
- C. Installation Acceptance Test (IAT)
- a. Performed after LM I&C
  - b. Will be conducted over a one (1) week period
  - c. One (1) or two (2) external agencies will participate by transferring data to be ingested
  - d. Support C&A testing
  - e. Will be conducted on the operational site

**I1R1**



**I1R2**



**I1R3**

Same as I1R2

**Figure K-1: Increment 1 Acceptance Test Flow**

**Final**

## **Emphasis of Test Scenarios/Test Coverage**

### **Release 1**

- 1) Workflow Functionality
- 2) Form Functionality
- 3) Workbench Functionality
- 4) Storage Functionality
- 5) User Registration/Account Functionality
- 6) Configuration Management Functionality
- 7) Service Management Functionality

### **Release 2**

- 1) Partial Disposition Functionality
- 2) Transfer
- 3) Ingest
- 4) System Management/System Monitoring
- 5) Software Deployment/Test Data Management
- 6) Manage Units of Work
- 7) Task Management
- 8) Template Functionality
- 9) Accessibility/Usability
- 10) Backup/Recovery

### **Release 3**

- 1) Full Disposition Agreement Capabilities
- 2) Initial Dissemination Capability
- 3) Template Management Functionality
- 4) Additional Transfer Capabilities
- 5) Sample Record Functionality
- 6) Legal Custody Processing
- 7) Original Order Creation
- 8) Records Management/Lifecycle Data
- 9) Preservation Planning
- 10) Full end-to-end processing

The following highlights the tasks and activities that will be performed as related to the test phases during Increment 1.

### **Test Planning & Support Tasks**

- Update the ERA Testing Management Plan
- Implement Test Tools and Approach

**Final**

- Witness LM Testing and provide reports
- Review LM documentation and provide feedback
- Review requirement coverage provided by LM tests
- Plan and prepare for OAT and IAT (also develop and maintain use cases)
- Coordinate with LM Test Team and ERA PMO Security Test Team

**PaT Tasks**

- Participate in the TRR presentation
- Perform PaT tests
- Perform a subset of the LM System Tests
- Document/Analyze Test Results
- Create a Report on the Test Results and Activities

**PAT Tasks**

- Participate in the TRR presentation
- Perform PAT tests
- Document/Analyze Test Results
- Give a presentation on the Test Results
- Create a Report on the Test Results and Activities

**OAT Tasks**

- Conduct the ORR presentation
- Coordinate user participation and perform OAT tests
- Document/Analyze Test Results
- Give a presentation on the Test Results
- Create a Report on the Test Results and Activities

**IAT Tasks**

- Participate in the TRR presentation
- Perform IAT tests
- Document/Analyze Test Results
- Give a presentation on the Test Results
- Create a Report on the Test Results and Activities

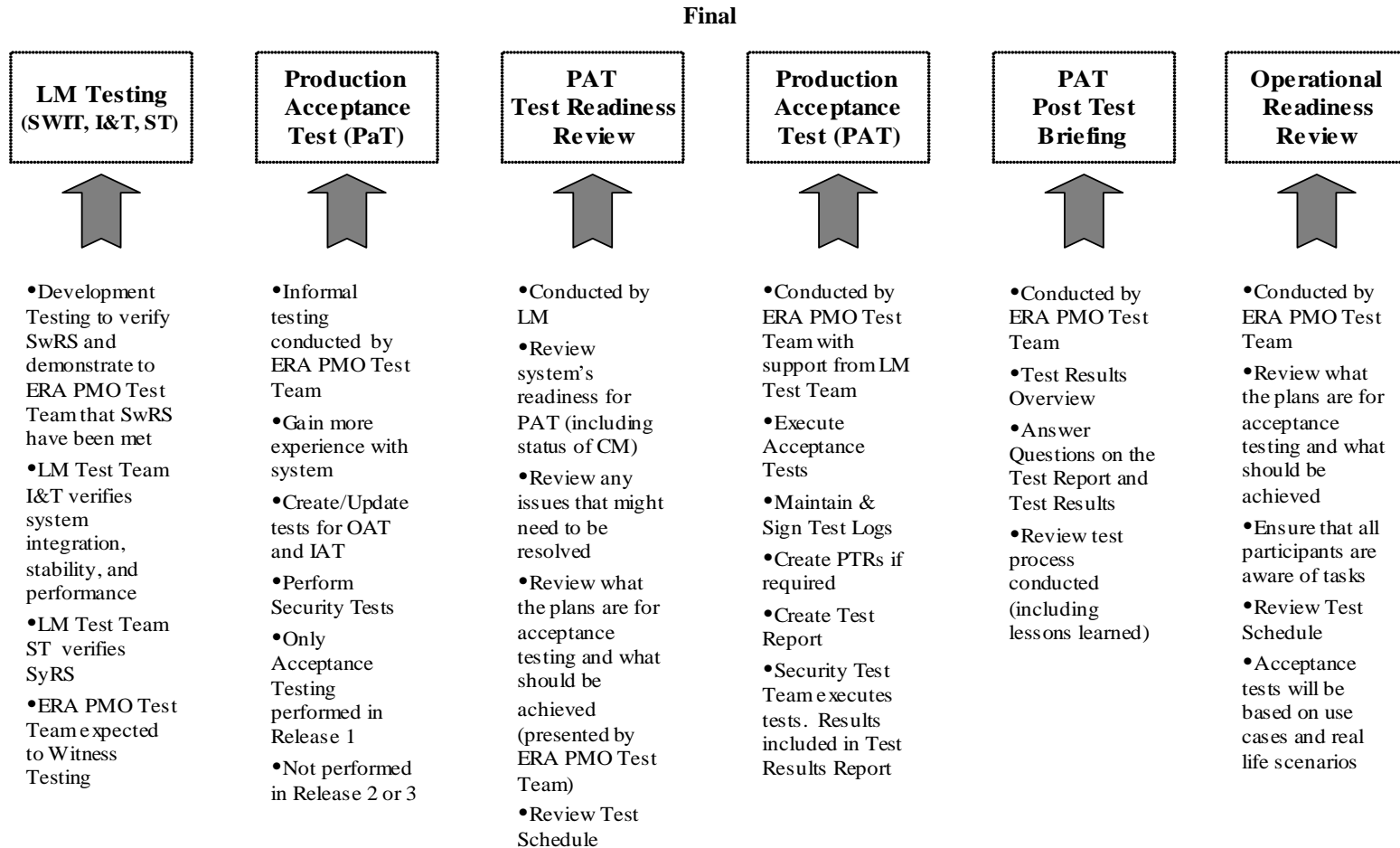
Final

ACCEPTANCE TEST MILESTONES

	PaT	PAT	OAT	IAT
R1	Duration: 3 weeks Roles: Tester – NARA Test Team Witness – LM Test Team Support – LM Activities: TRR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: LM Test Site	Not Performed	Not Performed	Not Performed
R2	Not Performed	Duration: 4 weeks Roles: Tester – NARA Test Team Witness – LM Test Team Support – LM Activities: TRR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: Operational Site Other: External Simulator	Duration: 1 week Roles: Tester – Expert Users/Trainers Witness – NARA Test Team Support – LM Activities: ORR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: Operational Site Other: External Simulator	Duration: 2 weeks Roles: Tester – NARA Test Team Witness – LM Test Team Support – LM Activities: TRR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: Operational Site Other: Agencies
R3	Not Performed	Duration: 4 weeks Roles: Tester – NARA Test Team Witness – LM Test Team Support – LM Activities: TRR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: CAT Other: External Simulator	Duration: 1 week Roles: Tester – Expert Users/Trainers Witness – NARA Test Team Support – LM Activities: ORR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: CAT Other: Agencies	Duration: 2 weeks Roles: Tester – NARA Test Team Witness – LM Test Team Support – LM Activities: TRR Perform Tests Collect/Remove Test Artifacts Test Result Report Post-Test Briefing Test Artifacts: Test Result Report Test Scenarios & Logs Captured Data Test Site: Operational Site Other: Agencies

Table K-1: Increment 1 Acceptance Test Milestones





**Figure K-2: Increment 1 Test Activities**

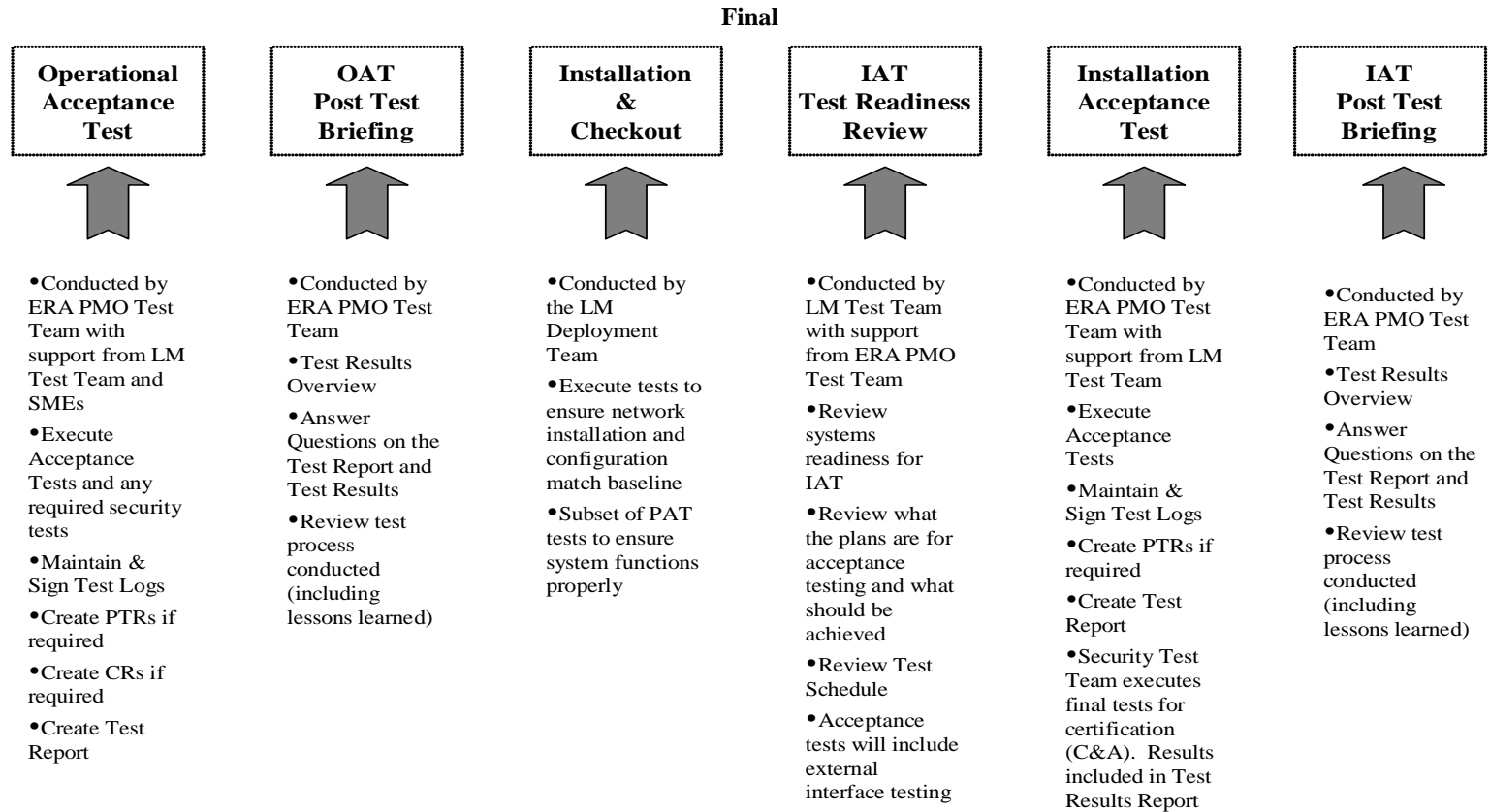


Figure K-2: Increment 1 Test Activities (continued)