U.S. Marine Corps

TEST
PLAN

DEPARTMENT OF THE NAVY
HEADQUARTERS UNITED STATES MARINE CORPS
WASHINGTON, D.C. 20380

5231/14
CCIS-45
20 MAR 1987

From: Commandant of the Marine Corps

Subj: INFORMATION RESOURCES MANAGEMENT (IRM) TEST PLAN

Ref: (a) MCO 1510.37
        (b) MCO P5231.1
        (c) MCO 5271.1
        (d) MCO P5600.31

Encl: (1) IRM-5231-14

1. PURPOSE. To provide guidance and instructions on the development of Test Plans as required by references (a) and (b).

2. AUTHORITY. This publication is published under the auspices of reference (c).

3. APPLICABILITY. The guidance contained in this publication is applicable to all contractors and Marine Corps personnel responsible for the preparation of a Test Plan. This standard is applicable to the Marine Corps Reserve.

4. DISTRIBUTION. This technical publication will be distributed as indicated. Appropriate activities will receive updated individual activity Table of Allowances for Publications. Requests for changes in allowance should be submitted in accordance with reference (d).

5. SCOPE

    a. Compliance. Compliance with the provisions of this publication is required unless a specific waiver is authorized.

    b. Waivers. Waivers to the provisions of this publication will be authorized only by CMC (CC) on a case by case basis.

6. RECOMMENDATIONS. Recommendations concerning the contents of this technical publication should be forwarded to CMC (CCI) via the appropriate chain of command. All recommended changes will be reviewed upon receipt and implemented if appropriate.

Subj: INFORMATION RESOURCES MANAGEMENT (IRM) TEST PLAN

7. <u>SPONSOR</u>. The sponsor of the technical publication is CMC
(CCI).

Brigadier General, U.S. Marine Corps
Director, Command, Control, Communications
and Computer (C4) Systems Division

DISTRIBUTION: L1P
                 L1Q
                 L1R
                 7000006   (2)
                 7000007   (2)
                 7000012   (2)
                 7000014   (2)
                 7000015   (2)
                 7000017   (2)
                 7000045   (2)
                 7000050   (2)
                 7000142   (2)
                 7000148   (2)
                 7000071   (5)

Copy to: 8145001

UNITED STATES MARINE CORPS

Information Resources Management (IRM) Standards and Guidelines Program

Test Plan
IRM-5231-14

2 0 MAR 1987

<u>RECORD OF CHANGES</u>

Log completed, change action as indicated.

Change Number	Date of Change	Date Received	Date Entered	Signature of Person Entering Change

PUBLICATION TABLE OF CONTENTS

LIST OF FIGURES

Chapter Table of Contents

Chapter 1

GENERAL

(This page intentionally left blank)

Chapter 1

GENERAL

1.1. INTRODUCTION. The objective of this Test Plan Standard is to define the format and content of all formal Test Plans to be developed in support of the project. By adhering to this standard, both development and user acceptance personnel will be assured of producing a uniform, applicable plan which provides the required and necessary planning, specification, and procedural information to meet the goals of the various test types and levels.

1.1.1. Objectives. The primary objectives of the Test Plans described herein are to provide the necessary management and technical guidance for performing the test functions, and to provide a schedule of events and activities. The Test Plans should also document the personnel and equipment resources necessary to support the test functions, and describe the system functions or capabilities to be tested. Also, the specific test inputs and expected results, and the means for documenting test result should be provided.

1.2. SCOPE. This standard will be the basis for all test plans required in support of applications software development and implementation. The standard is structured to enable the application developer or user to develop plans for conducting pre-delivery testing or acceptance testing, and to report the results. It provides for the development of three separate document components: Pre-Delivery Test Plan, Acceptance Test Plan, and Test Analysis Report. Pre-delivery testing is done by the developer while acceptance testing is the responsibility of the user or agency accepting the system. The Test Analysis Report is the formal document for describing the results of acceptance testing. The planning and reporting to be accomplished through pre-delivery and acceptance testing covers several different levels and types of tests. The following paragraphs briefly describe each of the three documents, and the testing that will be involved.

1.2.1. Documentation Components. This standard will be the basis for the preparation of all test plans and reports for the project implementation. The following paragraphs provide a brief description of each document listed above. The detailed description and content requirements are presented in paragraph 2.1., "Documentation Standards."

  a. Pre-Delivery Test Plan. This document describes the plans and procedures for the unit/module, integration, function, and system testing required to be performed by the developing organization. Also included should be a description of the technical report of test results.

b. <u>Acceptance Test Plan</u>. This document describes the activities associated with the formal acceptance testing to be conducted by the user and personnel responsible for formal system acceptance following the completion of the development activities.

c. <u>Test Analysis Report</u>. This document reports the results of acceptance testing.

1.2.2. <u>Test Activity Definitions</u>. The testing activities are defined in terms of the levels of development they are related to, and in terms of the types of testing to be conducted. To provide for consistent usage of the terms, all test-related documentation for the project should conform to the definitions and descriptions presented in the following paragraphs.

a. <u>Unit/Module Test Level</u>. Unit or module testing is the process of determining that the individual application software units or modules, by themselves and in isolation from other units, perform as designed. This testing is done by the application programmer responsible for implementing the detail design. The recommended approach to this level of testing is for the programmer to bring into the structured walkthrough some representation of the implementation, such as Pseudo-code or Structured English, along with the test cases by which the unit should be tested. During the walkthrough each test case should be played through the logic paths of the unit as would be expected after implementation. Upon approval to continue, the unit would be coded in the implementation language and the test cases applied through actual computer runs.

b. <u>Component Test Level</u>. An extension of unit testing is component testing, which consists of testing a sub-structure of software, less than a full function, as determined from the structure chart. A number of units can be integrated and tested by the same programmer, perhaps using stubs for not yet completed code. It is at the point where the components are being integrated into higher levels of structure, such as functions, that integration may exceed the scope of a single programmer and becomes an issue for a test organization.

c. <u>Integration Test Level</u>. Integration testing is the testing of the total system including the interfaces between and among the system parts. This entails different techniques and different philosophies depending on whether the parts to be tested are units of a single function, especially when designed and developed by a single programmer, or if the products of more than one person or elements of different functions are to be integrated. At one level, component testing can be a form of integration testing. The testing of higher structures of software resulting from combining components is also integration testing. The planning and control of integration testing for any particular development effort should be documented in the Pre-Delivery Test Plan. Like unit or module testing, these tests are to determine

if the software, as designed (or as-built), performs as the developer intended it to.

   d.   System Test Level.

      (1) System testing is the comparison of the developed system to its original objectives. This should be the last level of testing to be conducted by the developing organization, and should be planned and conducted by an independent test team, rather than the programmers who unit tested the software.

      (2) Because of the similarity between this testing and acceptance testing, there may be an economy realized by having the system test planned and executed either jointly by the development organization and the ultimate system user, or having the users monitor and approve the system tests.

      (3) System testing can be carried out in an environment similar to but different from the operational or production environment, but does not necessarily need to be done in the actual production environment. Also, these tests can be carried out using "artificial" test data generated especially for the test.

   e.   Acceptance Test Level. Acceptance testing should be conducted and controlled by the functional agency that will use and be responsible for the system. It should be carried out in the environment in which it is to operate in production, or as close as possible to it, and should use live data. This data can be captured from previous run cycles if there is an existing system to be replaced or improved. The tests should include the input transaction processing, edit checking, and validation checking of the data used to create master files rather than relying on the use of "converted" data files.

   f.   Functional Testing.

      (1) Functional testing concentrates on attempting to locate errors in the performance of the functions which the software is required to provide. It is called "black box" testing because all that needs to be known is what goes in and what should come out. A specific stimulus, usually in the form of an input transaction or terminal entry, should provide some result, usually an output on a screen, a report, or a data file.

      (2) This type of test is used for system and acceptance testing, and is especially suitable to requirements validation performed to determine if all of the user-required functions have been provided.

      (3) Functional testing attempts to find errors between the requirements and the implementation, and assumes that if the function performs for a nominal number of test cases, it can reasonably be expected to perform as stated in the requirements.

g.   Structural Testing.

(1) Structural testing is concerned with the implementation details and attempts to detect errors in the code structure, the statement, condition, and decision coverage. The tests are designed to exercise all of the conditional logic, with both true and false conditions, and to execute every statement of the code at least once. This is the type of testing employed at the lower levels of testing, where a variety of tools and techniques can be used to introduce test case values to exercise paths.

(2) Structural testing attempts to detect errors in implementation, and can only demonstrate that the code being tested does what it was built to do, regardless of whether or not it was built to perform the wrong function.

1.2.3. Test Categories. The following test categories are all accepted types of tests applicable to one or more of the test levels described above. The test planners preparing the pertinent plans defined by this standard should specify the types or categories of tests to be used in performing the testing.

a.   Facility Testing. Facility testing is the determination of whether each facility in the objectives was actually implemented. Facility in this context is identical with function, but is used to avoid confusion with functional testing as defined in Paragraph 1.2.2.f., "Functional Testing."

b.   Volume Testing. Volume testing is the determination of whether the system can handle the volume of data specified in the objectives.

c.   Stress Testing. Stress testing is the determination of whether the system can perform using a peak volume of data over a short period of time.

d.   Usability Testing. Usability testing is the determination of whether the system provides for the human factors and ease of use specified in the objectives.

e.   Security Testing. Security testing is the process of attempting to devise test cases that subvert the system's security checks as specified in the objectives.

f.   Performance Testing. Performance testing is the determination of whether the system performs within the response times and throughput rates specified in the objectives.

g.   Storage Testing. Storage testing is the determination of whether the system meets the storage requirements as specified in the objectives.

h. **Recovery Testing.** Recovery testing is the determination of whether the system recovers from programming errors, hardware failures, and data errors according to the objectives.

i. **Documentation Testing.** Documentation testing is the determination of whether the user documentation is accurate, clear, and useful. It is also the determination of whether the procedures outlined in the objectives and user documentation are met, such as procedures to be followed by the system operator, data-base administrator, terminal user, and functional user. Any examples illustrated in the documentation should be included in test cases and executed by the program.

j. **Equipment/Hardware Testing.** Equipment/hardware testing is the determination of whether the system operates on the hardware minimum and maximum configuration as specified in the objectives.

k. **On-line System Reliability Testing.** On-line system reliability testing is the determination of whether the on-line functions can accept and correctly process any input without causing total on-line system failure.

1.3. **APPROACH.** All applications software development efforts for the project will require one or more levels of testing to ensure that errors are detected and corrected prior to implementation. Test activities will be conducted by the developing organization prior to delivery of the software for acceptance testing. The data requirements for every effort will include the test plan of the appropriate type, and will specify the use of the pertinent standard component. One deliverable all development organizations will be required to develop is a Pre-Delivery Test Plan.

1.3.1. **Pre-Delivery Test Plan.**

a. **Requirements.** The Pre-Delivery Test Plan should contain the material identified and described in Chapter 2, "Content and Format", and should document the plans, schedules, and procedures for conducting module level tests, integration tests, functional tests, or system tests, and the responsibilities for the performing and monitoring of the tests. The test plan developer should document the plans and procedures for all levels and types of tests to be conducted during the development effort. The anticipated application of the test categories to the different levels of testing should be described. The choice of the testing methods at the lower levels are largely at the discretion of the development organization. However, the system level testing should be accomplished in a manner commensurate with the descriptions presented in Section 4 of the Acceptance Test Plan, "Test Specification and Evaluation."

b. **Technical Report.** Furthermore, the developer should document the results of pre-delivery testing in a technical report. The form and content of this technical report should vary according to the testing activities being performed, but should be

determined prior to the approval of the test plan. Results of
testing such as stress testing, performance testing, and system
testing should be described in this report. Particular attention
should be given to discussing deficiencies, their impact, and the
proposed corrective action to be taken. This report should be
processed in accordance with the Inspection and Acceptance
Standard.

1.3.2. Acceptance Test Plan. The Acceptance Test Plan will be
developed by the government and will be required prior to the Test
Stage of the System Development Phase of the life cycle. The
testing should be conducted under the control of the government
with the user personnel providing approval of the test results.
The Test Analysis Report is the vehicle for acceptance test
approval and should also be processed in accordance with the
Inspection and Acceptance Standard.

Chapter Table of Contents

Chapter 2

CONTENT AND FORMAT

(This page intentionally left blank)

Chapter 2

CONTENT AND FORMAT

2.1. DOCUMENTATION STANDARDS. The test plan components as described in paragraph 1.2., "Scope," will be developed as described in the following paragraphs. The components are intended to be stand alone; that is, the user of this standard can develop an Acceptance Test Plan and not be required to produce a Pre-Delivery Test Plan. In the following paragraphs, the components of each test plan will be described. The descriptions will consist of a required table of contents, a description of the contents, and the criteria for evaluating the completeness and correctness of the targeted document.

2.1.1. Pre-Delivery Test Plan Deliverable. One deliverable that should be produced in accordance with this standard is a Pre-Delivery Test Plan. The plan should be structured as specified in Appendix B and C. Guidelines for evaluation are included which will enable reviewing activities to properly evaluate the plan.

    a. Pre-Delivery Test Plan Table of Contents. All Pre-Delivery Test Plans to be developed for the project will be formatted as shown in Appendix B.

    b. Pre-Delivery Test Plan Content Description. The text materials to be included in each paragraph will be developed by following the content descriptions as shown in Appendix C.

2.1.2. Pre-Delivery Test Plan Evaluation Criteria. Criteria through which to evaluate the Pre-Delivery Test Plan for completeness and accuracy are as follows:

    a. All sections and paragraphs contained in Appendix C must be included as presented.

    b. Any section or paragraph deemed not applicable to the plan must appear with a statement to that effect and a justification for the exclusion.

    c. The scope and approach of the activities covered by the plan must be consistent with those presented in Chapter 1 of this standard.

    d. The test activities and methods must be consistent and applicable to the level of the development effort.

2.1.3. Acceptance Test Plan Deliverable. The second deliverable to be produced through the use of this standard should be an Acceptance Test Plan. The plan should be structured as specified in Appendix D and E.

a. Acceptance Test Plan Table of Contents. The required
format of the acceptance test plan is shown in Appendix D.

b. Acceptance Test Plan Content Description. A description
of the required subject matter for each paragraph of the plan is
contained in plan format in Appendix E.

2.1.4. Acceptance Test Plan Evaluation Criteria. Criteria through
which to evaluate the Acceptance Test Plan for completeness and
accuracy are as follows:

a. All sections and paragraphs contained in Appendix D must
be included as presented.

b. Any section or paragraph deemed not applicable to the plan
must appear with a statement to that effect and a justification
for the exclusion.

c. The scope and approach of the activities covered by the
plan must be consistent with those presented in Chapter 1,

d. The test activities and methods must be consistent and
applicable to the level of the development effort.

2.1.5. Test Analysis Report Deliverable. The deliverable that
will be produced through the use of this standard will be a Test
Analysis Report. This report will be structured as specified in
Appendix F and G.

a. Test Analysis Report Table of Contents. All test analysis
reports to be developed for the project should be formatted as
shown in Appendix F.

b. Test Analysis Report Contents Description. The test
materials to be included in each paragraph should be developed by
following the content descriptions as shown in Appendix G.

2.1.6. Test Analysis Report Evaluation Criteria. The Test
Analysis Report should be documented in accordance with Appendix F
and G. If deviations from this standard are felt by the
acceptance test agency to be necessary or desirable, a waiver must
be requested from the Project Manager. Attached to the waiver
must be the rationale for the deviation.

2.2. DOCUMENTATION DEPENDENCIES. The documentation governed by
this standard may also rely on the content of other project
deliverables and/or standards. Figure 2-01, "Precedence
Relationship," shows those project deliverables and standards
which impact the Test Plan deliverables.

2.2.1. Preceding Documents. The boxes that precede the Test Plan
as shown by a connected line with an arrow, are those project
deliverables that must be completed before the Test Plan. The
preceding documents for any one development effort are:

a. Data Base Plan Deliverables.

    (1) Data Base Plan Deliverables
    (2) Detailed Design Specification Deliverables

2.2.2. Consultation Documents. The boxes and bars that are in line vertically with the Test Plan show the concurrent documents that may be consulted at that time. The boxes are other project deliverables governed by standards, and the bars are particular conventions described by standards. The deliverables and standards used for consultation are:

a. Data Base Conversion Plan Deliverables.

    (1) Data Base Conversion Plan Deliverables
    (2) ADPE Support Plan Deliverables
    (3) Telecommunications Support Plan Deliverables
    (4) Project Deliverables Style Manual (IRM-5230-02)
    (5) Inspection and Acceptance (IRM-5231-17)
    (6) Data Dictionary (IRM-5235-01)
    (7) Library Management System (IRM-5233-06)
    (8) Man-Machine Dialogue (IRM-5234-02)
    (9) Programming Standard (IRM-5234-01)
   (10) Prototyping Standard (IRM-5231-18)

2.2.3. Change Requirements. Since the SDM is an integrated methodology, there exists a relationship between documents in that preceding documents provide information to the follow-on documents. During the development of the Test Plan new issues may arise that will require changes to preceding documents. These changes must be documented and approved in accordance with the quality assurance and configuration management procedures. Externally imposed milestones that are unrealistic to accomplish should not be used as an excuse to defer or eliminate the documentation requirements.

SYSTEM REQTS. REVIEW ◇   PRELIM. DESIGN REVIEW ◇   CRITICAL DESIGN REVIEW ◇   ACCEPTANCE REVIEW ◇   SYSTEM EFFECTIVENESS REVIEW ◇

MISSION ANALYSIS/ PROJECT INITIATION	CONCEPT DEVELOPMENT	DEFINITION AND DESIGN			SYSTEM DEVELOPMENT			DEVELOPMENT AND OPERATIONS
		GENERAL		DETAILED DESIGN	CODE	DEVELOPMENT AND INTEGRATION	TEST	EVAL.
		DEF.	DESIGN					

○ MENS   ○ SDP I   ○ SDP II   ○ SDP III   ○ SDP IV

SPECIFICATIONS

RS → EA → FRD → GDS → DDS

DBP → DBC

TP

ADP

TSP

PLANS

CONVENTIONS

SM

IA

OD

LMS

PS

PRS

MDS

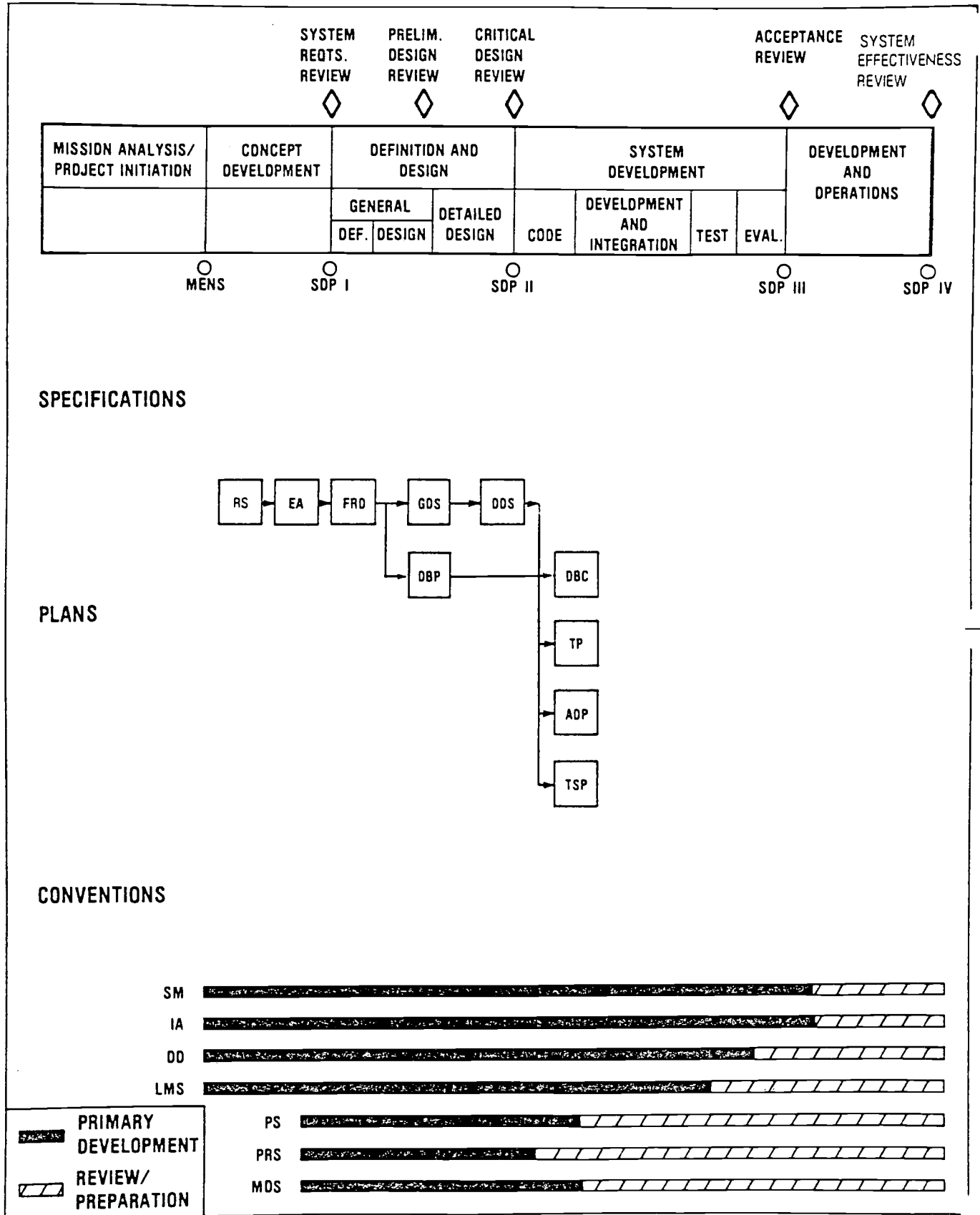▬ PRIMARY DEVELOPMENT

▨ REVIEW/ PREPARATION

FIGURE 2-01
Precedence Relationship

2-6

Appendix A

GLOSSARY

ADP: ADP   is an acronym for "ADPE Support Plan"

DBC: DBC   is an acronym for "Data Base Conversion Plan"

DBP: DBP   is an acronym for "Data Base Plan"

DD: DD   is an acronym for "Data Dictionary"

DDS: DDS   is an acronym for "Detailed Design Specification"

EA: EA   is an acronym for "Economic Analysis"

FRD: FRD   is an acronym for "Functional Requirements Definition"

GDS: GDS   is an acronym for "General Design Specification"

IA: IA   is an acronym for "Inspection and Acceptance"

LMS: LMS   is an acronym for "Library Management System"

MDS: MDS   is an acronym for "Man-Machine Dialogue"

MENS: MENS   is an acronym for "Mission Element Need Statement"

PRS: PRS   is an acronym for "Prototyping Standard"

PS: PS   is an acronym for "Programming Standard"

RS: RS   is an acronym for "Requirements Statement"

SDP: SDP   is an acronym for "System Decision Paper"

TP: TP   is an acronym for "Test Plan"

TSP: TSP   is an acronym for "Telecommunications Support Plan"

(This page intentionally left blank)

Appendix B

PRE-DELIVERED TEST PLAN TABLE OF CONTENTS

Pre-Delivery Test Plan

(This page intentionally left blank)

Appendix C

PRE-DELIVERED TEST PLAN CONTENT DESCRIPTION

SECTION 1 INTRODUCTION

This section will describe the purpose of the Pre-Delivery Test
Plan, a brief summary of applicable references, terms and
abbreviations, and security considerations.

1.1 PURPOSE OF THE PRE-DELIVERY TEST PLAN

This paragraph will describe the purpose of the Pre-Delivery Test
Plan and introductory information on the use of this document.
This introductory material should include the scope,
responsibility, and organization of the document, and references
to applicable policy documents. The Test Plan should be written
to fulfill the following objectives:

    a.   Provide a document for communication and control during
the development activity

    b.   Provide guidance for the management and technical effort
necessary throughout the test activities

    c.   Establish a comprehensive Pre-Delivery Test Plan and
communicate to the user the nature and extent of the testing
necessary to provide a basis for implementation of the system

    d.   Coordinate with the user an orderly schedule of events, a
specification of equipment (both hardware and software) and
organizational requirements (to include personnel), methodology of
testing, and a list of materials to be delivered

    e.   Provide the specification of system requirements and
functions to be evaluated, the test constraints to be adhered to,
and the test methodology to be used

    f.   Provide descriptions of the test, the method to control
the test, and procedures to be followed when executing the test

    g.   Provide a written record of the actual test inputs, the
instructions to permit execution of the test by the user staff and
operator personnel, and the expected outputs

    h.   Provide form and content descriptions for the Technical
Pre-Delivery Test Results Report which will communicate the test
results, known deficiencies, impacts, and recommended solutions

1.2 PROJECT REFERENCES

This paragraph should provide a summary of the references
applicable to the history and development of the project.

Documentation describing systems or procedures which supplement or provide for interaction with the project should be specified. All references should be identified with at least the author, title, agency, and the approval agency and date. All project references can be provided in an appendix.

1.3 TERMS AND ABBREVIATIONS

This paragraph should provide a listing, or make reference to an appendix containing all terms, definitions, or acronyms unique to this document and subject to interpretation by the user of the document. It will make reference to the Volume II for all terms defined there.

1.4 PRIVACY AND PROTECTION OF TEST PLAN

This paragraph should describe any privacy and security considerations associated with this Test Plan in accordance with Marine Corps Order (MCO) P5510.14.

SECTION 2 RESPONSIBILITIES

This section should describe the responsibilities of the developer, whether contractor or in-house, and the responsibilities of the project organizations having authority for Pre-Delivery Test approval.

2.1 DEVELOPER'S RESPONSIBILITIES

This paragraph will describe the responsibilities of the members of the development organization, including but not limited to, the following testing activities:

a. Conduct of any unit/component testing required during system development

b. Conduct of pre-delivery testing in the garrison hardware configuration

c. Preparation of a written technical report itemizing the results of pre-delivery testing

d. Submission of technical reports for approval authority review and approval

e. Attendance at meetings associated with test activities

2.2 TEST APPROVAL AUTHORITY RESPONSIBILITIES

This paragraph will describe the responsibilities of the approval authority. The approval authority's responsibilities will be assigned by the Project Manager. The approval responsibilities include but are not limited to the following:

a. Review and approval of the Test Plan, specifications, and procedures

b. Validation of both functional and operational characteristics of the project

c. Monitoring of test activities, to include review of test conditions and results.

d. Approval of the technical reports submitted by either the contractor or the in-house developer

e. Attendance at meetings associated with test activities

2.2.1 Project Development Office Responsibilities

A brief description of the responsibilities of the Project Manager for the activities listed in Section 2.2, "Test Approval Authority Responsibilities," should be provided, if applicable.

2.2.2 CDPA/RASC/DFASC Responsibilities

A brief description of the responsibilities of Central Design and Programming Activities (CDPA's), Regional Automated Service Centers (RASC's), or Deployable Force Automated Service Centers (DFASC's), as appropriate, for the activities in Section 2.2, "Test Approval Authority Responsibilities," should be provided, if applicable.

2.2.3 Functional User Responsibilities

A brief description of the responsibilities of functional users for the activities in Section 2.2, "Test Approval Authority Responsibilities," should be provided, if applicable.

SECTION 3 PRE-DELIVERY TESTING ACTIVITIES

The conduct of the Pre-Delivery Test activities required during development should be accomplished in accordance with the task description or contract governing the effort, and in accordance with accepted software testing practices. The primary objectives of pre-delivery testing are:

a. To detect any errors within the modules, subsystems, or system

b. To ensure all parts and the system perform their intended functions correctly

c. To ensure that the modules, subsystems, and system do what they are supposed to do

In planning for the tests, the levels and categories described within this standard should be used as the basis, but should not limit the test activities from broadening the scope of tools and methodologies based on the developer's experience. This section should describe the pre-test and test activities, and should cover as appropriate the subjects addressed below.

3.1 PRE-TEST ACTIVITIES

This paragraph should present the activities to be carried out in preparation for the actual testing before it begins. Examples of the types of activities include:

a. Training of testers

b. Scheduling of resources, including human, equipment, and environmental

c. Preparation of test data

d. Obtaining support or test software

e. Planning for handling and storage of test output

This is not intended to be an exhaustive list. The test plan developer must ensure that contingencies for the subject testing have been provided for and described in detail in the Test Plan.

3.2 TESTING METHODOLOGIES

Pre-delivery testing should be planned and conducted by the development organization. The Test Plan and related specifications and procedures should be reviewed and approved by the government prior to plan implementation, as appropriate, and the results should be reviewed upon completion of testing. The monitoring of actual test activities may be done as a Quality Assurance (QA) function by the organization designated to perform QA.

The test activities to be carried out should be commensurate with the scope and nature of the development effort, but should normally consist of the following, as defined in Paragraph 1.2.2., "Test Activity Definitions," of this standard.

a. Unit/Module Testing

b. Component Testing

c. Integration Testing

d. Function Testing

e. System Testing

This paragraph should contain a description of the pre-delivery testing which will be provided by the developer. The developer should prepare and submit to the approval authority a written plan that itemizes the proposed pre-delivery testing. Because of the similarity between the system testing and acceptance testing activities, test plan developers should review paragraph 2.1.3., "Acceptance Test Plan Deliverables," of this standard, and employ the methods specified, as applicable. Included in this section should be the subjects described in the following paragraphs.

3.2.1 System Description

This paragraph should provide a general description of the system being tested. A brief system data flow diagram (DFD) or similar overview chart showing inputs and outputs should be included to provide a frame of reference for the description of the tests to be conducted. Also, a system organizational chart should be provided that will include the testing requirements at each level.

3.2.2 Testing Schedule

This paragraph should provide a listing or chart identifying the test types, the location where the testing will be done, and the time frame related to each test.

3.2.3 Location

This paragraph should identify the location at which the testing will be completed and the participating organizations involved in the tests. This paragraph should also list the testing to be performed, the types of test required, the approval authority for each type of test, and the plan for attaining the various functions.

3.2.4 Milestone Chart

This paragraph should provide a chart that correlates the test events with the time frames established for their completion. When preparing this chart, consideration should be given to all tests scheduled. The chart should be in chronological order with supporting narrative that depicts the activities listed below:

    a. Overall on-site test time by calendar date

    b. Pre-test on-site time required for system debugging, orientation, and familiarization

    c. Time assigned for the collection of data base values, input values, and other operational data required for a system test

    d.   Time assigned for user orientation and familiarization with system documentation

    e.   Time assigned for user training, operator training, maintenance and control group training, and orientation briefings for non-ADP personnel

    f.   Time assigned for preparation, review, and approval of the Technical Pre-Delivery Test Results Report.

3.2.5 Equipment Requirements

This paragraph should provide a list or chart identifying any special or temporary equipment needed for the test. The quantity required and the period of use for each type of equipment should be specified. The location of the equipment and the organization having operational control of the equipment should also be specified. Examples of such equipment are extra peripherals (tape drives, printers, plotters), test message generators, and test timing devices.

3.2.6 Software Requirements

This paragraph should list the support software used during the test which is not part of the system being tested. Programs such as operating system sort programs and one-time application test programs should be listed with a brief description.

3.2.7 Personnel Requirements

This paragraph should provide a chart or a listing of the number and names of required personnel, skills required, and duration of involvement during the test periods. It should indicate special requirements, such as multi-shift operation and assignment, or the retention of key skills to ensure continuity and consistency in extensive test programs. This chart or listing should be related to the Milestone Chart defined in Paragraph 3.2.4.

3.2.8 Responsibilities

This paragraph should identify the test team leaders and personnel responsible for preparing test data, calculating predetermined results, organizing the overall test, and repairing discovered errors. The paragraph should identify the person to be contacted in the event of problems.

3.2.9 Test Materials

The following paragraphs itemize those articles associated with conducting the test.

3.2.9.1 Deliverable Materials

This paragraph should itemize all materials that will be delivered
as part of the test system. Included should be the quantity and
full identification of the following:

a. Computer Operations Manual, Test Plan, QA Plan, and
Implementation Plan

b. Program source code, listings, procedures, and Job Control
Language (JCL)

c. Data base source code (i.e., ADAMINTS, direct calls),
listings, procedures, and JCL

3.2.9.2 Site-Supplied Materials

All items that are expected to be provided by the test site should
be listed in this paragraph. Examples of these include:

a. Data base input

b. Other application inputs

c. Test control auditing programs

d. Test worksheets and other forms or instructions prepared
to control and expedite the test activity, including type, layout,
and quantity

3.2.10 Security and Privacy

This paragraph describes any security considerations associated
with the Test Plan in addition to any privacy restrictions
associated with the data being handled. Security restrictions
should be provided in a three column list specifying the system
component (for example, system, program, input, and outputs), the
security classification of that component, and any special
consideration necessary for handling that component. Privacy
restrictions should be provided in a two column list specifying
the system component and any special considerations necessary for
handling that component.

3.2.11 Test Environment

The responsible individual at the test site should be identified
as well as a list of items required by the test team at the test
site such as physical facility, furniture, and telephones. Access
requirements should be identified and proper procedures followed
in advance of the test.

3.2.12 Waivers of Standards

This paragraph should describe any waivers from established
project, USMC, or DOD Standards.

SECTION 4   TECHNICAL PRE-DELIVERY TEST RESULTS REPORT

This section describes the responsibilities, format, and content for creating the subject report.

4.1   RESPONSIBILITIES

The system development organization is responsible for creating the Technical Pre-Delivery Test Results Report. Within the development organization, the authors and approval authorities for each test should be indicated. The final approval authority should also be identified. For in-house developers, this individual should be designated by the Project Manager. For application contractors, this approval authority should be the individual having signatory authority to submit contract deliverables to the government Contracting Officer.

Once this report is submitted for final approval, the procedures in the Inspection and Acceptance Standard should be followed. Final approval authority for in-house developers should be the Project Manager, and for application contractors the final approval authority should be the Contracting Officer. This final approval should be obtained before acceptance testing begins.

4.2   FORMAT

The format of this report should be outlined in a Contract Data Requirements List (CDRL) DD Form 1423, or equivalent document. While the format may vary, it should be in accordance with the Style Manual.

4.3   CONTENT

The content of the Technical Pre-Delivery Test Results Report should be organized to relate directly to the Pre-Delivery Test Plan. Any deviations of actual testing compared to planned testing should be indicated. The content of this report should be defined and approved in the Pre-Delivery Test Plan and should be organized by test type such as unit/module test, component test, integration test, function test, or system test. For each test type, the following should be included in the report:

(Note: For unit/module testing, the results information may be summarized at the discretion of the test plan approval authority.)

a.   Test Description - What the test was, where it was conducted, when it was conducted, the personnel involved in the test should be listed, and the test approval authority should be included.

b.   Milestone Chart - Include deviations of "actual" versus "planned" test activities and dates for retest if applicable.

c. Test Scenario Description - Describe the actual test scenario. Was all hardware, software, and material in place for the test? Was the environment ready for the test? What problems were encountered during the conduct of the test and how were they resolved? Was the test successful from the standpoint of the test scenario?

d. Summary Test Results - Each specification of the system requirements and each function that was tested, as described in Section 1.1, "Purpose of the Pre-Delivery Test Plan," should be listed and the test results should be indicated. Test results normally would be Pass, Fail, or Pass with Modifications. Deficiencies should be noted with corrective action taken if the result was Pass with Modification. If the result was Fail, the deficiency and associated impact should be stated along with proposed corrective action and a retest schedule as required. A matrix showing the specification or function that was tested mapped against results should be used to graphically depict the test results. Narrative comments, deficiencies, impacts, and retest schedules should accompany the matrix.

e. Detailed Test Descriptions - A form should be used to record test results as the test is run. This form should contain the following items:

    (1) Date of Test

    (2) Location of Test

    (3) Recorder's Name

    (4) Type of Test - unit/module, function, etc.

    (5) Specification/Function to be Tested

    (6) Expected Results

    (7) Actual Results

    (8) Pass, Fail, or Pass with Modifications

    (9) Deficiencies/Impact

    (10) Signature of Test Approval Authority

Copies of these forms should be maintained by the developer in the project files and, at the discretion of the approving authority, attached to the Technical Pre-Delivery Test Results Report.

(This page intentionally left blank)

Appendix D

ACCEPTANCE TEST PLAN TABLE OF CONTENTS

Acceptance Test Plan

Appendix E

ACCEPTANCE TEST PLAN CONTENT DESCRIPTION

SECTION 1   INTRODUCTION

This section should describe the scope, responsibility, and
organization of the document, reference material, terms and
abbreviations, and privacy and security considerations.

1.1   PURPOSE OF THE ACCEPTANCE TEST PLAN

This paragraph should describe the purpose of the Acceptance Test
Plan and introductory information on the use of this document.
This introductory material should include the scope,
responsibility, and organization of this document and references
to applicable policy documents. This paragraph should accomplish
the following objectives:

    a.   Provide a document for communication and control during
the acceptance testing activity

    b.   Provide guidance for the management and technical effort
necessary throughout the test period

    c.   Establish a comprehensive Acceptance Test Plan and
communicate to the user the nature and extent of the testing
necessary to provide a basis for implementation of the system

    d.   Coordinate with the user an orderly schedule of events, a
specification of equipment (both hardware and software) and
organizational requirements (to include personnel), methodology of
testing, a list of materials to be delivered, and a schedule of
user orientation and training

    e.   Provide the specification of system requirements and
functions to be evaluated, the test constraints to be adhered to,
and the test methodology to be used

    f.   Provide descriptions of the tests, the method to control
the tests, and procedures to be followed when executing the tests

    g.   Provide a written record of the actual test inputs to
exercise system limits and critical capabilities, the instructions
to permit execution of the test by the user staff and operator
personnel, and the expected outputs

1.2   PROJECT REFERENCES

This paragraph should provide a summary of the references
applicable to the history and development of the project.
Documentation describing systems or procedures which supplement or
provide for interaction with the project should be specified.

All references should be identified with at least the author, title, agency, and the approval agency and date. All project references can be provided in an appendix.

1.3 TERMS AND ABBREVIATIONS

This paragraph should provide a listing, or make reference to an appendix containing all terms, definitions, or acronyms unique to this document and subject to interpretation by the user of the document. A reference should be made to the glossary in Volume II for the terms defined there.

1.4 PRIVACY AND PROTECTION OF TEST PLAN

This paragraph should describe any privacy and security considerations associated with this Test Plan in accordance with MCO P5510.14.

SECTION 2 RESPONSIBILITIES

This section should describe the responsibilities and approval authority of Acceptance Test Plan development, and of the acceptance testing activities. It should identify the organizations that are to prepare the Test Plan and test data, to conduct tests, to monitor testing, and to approve the results of the testing.

2.1 DEVELOPER'S RESPONSIBILITIES

This paragraph should describe the responsibilities of any development organization required to participate or assist in the acceptance testing process.

2.2 TEST APPROVAL AUTHORITY RESPONSIBILITIES

This paragraph should describe the responsibilities of the test approval authority. The approval authority's responsibilities will be assigned by the Project Manager. The responsibilities include, but are not limited to, the following testing activities:

    a. Development of a detailed Acceptance Test Plan delineating all required procedures associated with the acceptance testing of the project

    b. Preparation of a test data base, test input data, and the associated expected results for use during acceptance testing

    c. Conduct of the acceptance test, possibly with the developer's assistance

    d. Validation of both functional and operational characteristics of the project

e. Conduct of meetings associated with the acceptance test

f. Preparation of the Test Analysis Report

2.2.1 Project Development Office Responsibilities

A brief description of the responsibilities of the Project Manager for the activities listed in Section 2.2, "Test Approval Authority Responsibilities," will be provided, if applicable.

2.2.2 CDPA/RASC/DFASC Responsibilities

A brief description of the responsibilities of CDPAs, RASCs, and DFASCs as appropriate, for the activities in Section 2.2, "Test Approval Authority Responsibilities," will be provided.

2.2.3 Functional User Responsibilities

A brief description of the responsibilities of the functional user for the activities in Section 2.2, "Test Approval Authority Responsibilities," will be provided, if applicable.

SECTION 3 TEST PLAN

The Acceptance Test Plan is the vehicle by which the system user defines the test environment, conditions, and criteria necessary to ensure a valid, accurate acceptance test of the system being implemented. The test should be conducted by an organization distinct and separate from the agency or contractor which led the application system development, or provided the hardware, telecommunications equipment, or system software. Although the acceptance test may be similar to the system test conducted during pre-delivery testing, it differs because the acceptance test should be conducted in a "live" environment, using developer-trained user and operator personnel, and controlled by the user agency. This section should present a description of the system being tested, a schedule for the test, a milestone chart, the location of the test, and resources required for the test process, such as the personnel, software, and equipment.

3.1 System Description

This paragraph should present a general description of the system being tested. A brief system data flow diagram (DFD), or similar overview chart showing inputs and outputs should be included to provide a frame of reference for the description of the tests to be conducted. Also, a system organizational chart should be provided including the macro-level testing requirements at each functional level.

3.2 Testing Schedule

This paragraph should provide a listing or chart identifying the test types, the locations to accomplish the test, and the time frame related to each test.

3.3 Testing Locations

This paragraph should identify the location at which the testing will be completed, and the participating organizations. This paragraph should also list the tests to be performed. The management plan for the types of tests required, the approval authority for each type of test, and the plan for bringing the various functions to completion should also be included.

3.4 MILESTONE CHART

This paragraph should provide a milestone chart which correlates the test events with the time frames established for their completion. When preparing this chart, consideration should be given to all tests scheduled. The chart should be in chronological order with supporting narrative depicting the activities listed below.

    a. Overall on-site test period by calendar date

    b. Pre-test on-site period required for system debugging, orientation, and familiarization

    c. Period assigned for the collection of data base values, input values, and other operational data required for system test

    d. Period assigned for user orientation and familiarization with system documentation

    e. Period assigned for user training, operator training, maintenance and control group training, and orientation briefing for non-ADP personnel

    f. Period assigned for preparation, review, and approval of the Test Analysis Report

3.5 EQUIPMENT REQUIREMENTS

The paragraph should provide a list or chart identifying any special or temporary equipment needed for the test. The quantity required and period of usage for each type of equipment should be specified. The location of the equipment and the organization having operational control of the equipment will also be specified.

3.6 SOFTWARE REQUIREMENTS

This paragraph should list the support software used during the test which is not part of the system being tested. Programs such as operating system sort programs and one-time application test programs should be listed with a brief description.

3.7 PERSONNEL REQUIREMENTS

This paragraph should provide a chart or a listing of the number and names of personnel with indicated skill types required during the entire test period. It should indicate special requirements, such as multi-shift operation and assignment, or the retention of key skills to ensure continuity and consistency in extensive test programs. This chart or listing should be related to the milestone chart.

3.8 PERSONNEL RESPONSIBILITIES

This paragraph should identify personnel responsibilities. The responsibilities may include who will prepare the test data, calculate the predetermined results, or organize the overall test. Also identified should be the person to be contacted if problems arise during the test.

3.9 TRAINING ORIENTATION

This paragraph should describe the number of personnel being provided training prior to testing and the types of training to be undertaken. This information should be related to the personnel requirements in Paragraph 3.7, "Personnel Requirements." This plan should include user instruction (classroom and computer), operator instruction, maintenance and control group instruction (if applicable), and the orientation briefing of non-computer-oriented staff personnel. Reference should be made to the Training Plan Standard.

3.9.1 User Training

This paragraph should describe the user training in the use and operation of the system. Included in this training is the use of system documentation and manuals. The organization responsible for providing user training and the scope of the training should also be provided. The Users Manual Standard should be referenced.

3.9.2 Operator Training

This paragraph should describe training in the operation of the system. Included in this training is the use of system documentation, manuals, procedures, restore/restart procedures, and backup/recovery procedures. The organization responsible for providing operation training should also be provided. The Computer Operations Manual Standard should be referenced.

3.9.3 Maintenance Training

This paragraph should describe the training for the maintenance of the system. Included in this training is the use and maintenance of system documentation, manuals, and procedures. The organization responsible for providing maintenance training and the scope of the training should also be provided.

3.10 TEST MATERIALS

This paragraph should itemize the articles associated with conducting the test, such as program source, listings, test control program, worksheets, and extra peripheral equipment required to perform the tests.

3.10.1 Deliverable Materials

This paragraph should itemize all materials that will be delivered as part of the accepted system. Included should be the quantity and full identification of the following:

    a. Users Manual, Computer Operations Manual, QA Plan, and Implementation Plan

    b. Program source code, listings, procedures, and Job Control Language (JCL)

    c. Data base source code (i.e., ADAMINT, direct calls), listings, procedures, and JCL

3.10.2 Site-Supplied Materials

All items that are expected to be provided by the test site should be listed in this paragraph. Examples of these include:

    a. Data base input

    b. Other application inputs

    c. Test control programs or other special test programs and their form

    d. Test worksheets and other forms or instructions prepared to control and expedite the test activity

    e. Equipment required during or in support of the test, when it is not normally part of the equipment configuration or is not being delivered as part of the installation effort; identify the equipment by name, type, and quantity required. Examples of this equipment are tape drives, printers, plotters, test message generators, test timing devices, and test event records.

3.11 SECURITY AND PRIVACY

This paragraph should describe any security considerations
associated with this test in addition to any privacy restrictions
associated with the data being handled. In addition, governing
documentation should be identified.

3.11.1 Security Restrictions

Security restrictions should be provided specifying the system
component (for example, system, program, input, outputs), the
security classification of that component, and any special
considerations necessary for handling that component.

3.11.2 Privacy Restrictions

Privacy restrictions should be provided specifying the system
component and any special considerations necessary for handling
that component.

3.12 TEST ENVIRONMENT

The person responsible for the test environment at the site should
be identified by name, organization, address, and phone number.
Also included should be a list of items required at the site such
as telephones, furniture, and adequate space. Access requirements
should be identified and arrangements made for the test personnel
to have physical and operational access.

3.13 WAIVERS OF STANDARDS

This paragraph describes any waivers from established project,
Marine Corps, or DOD Standards.

SECTION 4 TEST SPECIFICATION AND EVALUATION

This section should describe the requirements of the acceptance
test, functions that will be exercised, test methods and
constraints, test progression, and evaluation criteria.

4.1 TEST SPECIFICATIONS

This paragraph should describe the test specifications as required
by the Functional Requirements Definition (FRD).

4.1.1 Requirements

This paragraph should describe the requirements and goals of the
system as outlined in the FRD. The requirements should specify
why the system is needed, what the system should do, and how well
the system should do it.

4.1.2 System Functions

This paragraph should provide a detailed list of the system functions which will be exercised during the overall acceptance testing. This list, derived from the FRD, must be ordered in such a way that the functions are related to the requirements given in Paragraph 4.1.1, "Requirements." The structure of the testing is designed by specifying all functions, the hierarchical structure of all the functions, and the interfaces between functions.

4.1.3 Test/Function Relationships

This paragraph should provide a list of the tests which, taken as a whole, constitute the overall test activity. It will also provide, as applicable, a test/function relationship chart summarizing the overall allocation of the system functions to the test. E-01, "Example of Test/Function Relationship Chart," shows an example of this type of chart. The equivalence partitioning, boundary-value analysis, cause-effect graphing, and error-guessing methods are especially pertinent to function testing and are described in the following paragraphs.

4.1.3.1 Equivalence Partitioning

This paragraph should describe the equivalence classes which are identified by taking each input condition (usually a sentence or phrase in the specification) and partitioning it into two or more groups. A table similar to Figure E-02, "Example of Equivalence Classes," should be used to list all external (input) conditions and the two types of equivalence classes - valid and invalid. The valid equivalence classes represent valid inputs to the system function, or program whereas the invalid equivalence classes represent all other possible states of the condition. The following set of guidelines may be used to identify input conditions for equivalence classes:

a. Range of values (for example, 1-999)

b. Number of values (for example, 1, 2, 3, 4, 5)

c. Set of values (for example, E1, E2, E4, E5, E9)

d. Classes of values (for example, alpha, numeric)

FUNCTION (Reference)	Generate and maintain the data base (3.2.1)		Selectively retrieve data (3.2.2)											Produce special catalog (3.2.3)			
Program short name Number/test																	
1. Add new record	X																
2. Add new SORU	X																
3. Add a card	X																
4. Change a card	X																
5. Delete a card	X																
6. Delete a record	X																
7. Delete a SORU		X															
8. Create index file			X														
Retrieve records with a requested:																	
9. SORU				X													
10. Command designator					X												
11. Functional area						X											
12. Record ID code (RIC)							X										
13. Several RICs							X										
14. Index set and SORU								X									
15. Index set									X								
16. 2 different index sets									X								
17. 3 different index sets									X								
18. RIC and associated records									X								
19. Print full data base														X			
20. Extract record directory															X		
21. Extract record names/RICs																X	
22. Extract keywords, permutated																	X

FIGURE E-01
Example of Test/Function Relationship Chart

	EXTERNAL CONDITION	VALID EQUIVALENCE CLASSES	INVALID EQUIVALENCE CLASSES
	Vehicle can be car, truck, bus	car truck bus	trailer

FIGURE E-02
Example of Equivalence Classes

4.1.3.2 Boundary-Value Analysis

This paragraph should describe the boundary-value analysis which requires one or more elements be selected such that each edge of this equivalence class is the subject of a test. The following set of general guidelines may be used to identify test cases:

    a. Minimum and maximum number (for example, minimum 1 and maximum 999)

    b. Beneath minimum and beyond maximum (for example, range 1-999; 0 and 1000)

    c. Blank and zero values (for example, 0 or " ")

4.1.3.3 Cause-Effect Graphing

This paragraph should describe the cause-effect graphing that identifies a systematic method of generating test cases representing combinations of conditions. The cause-effect graphing requires the translation of specifications into Boolean logic network. The following process should be used to derive test cases:

    a. Divide the specifications into workable pieces.

    b. Identify the causes and effects in the specifications. A cause is a distinct input condition. An effect is an output condition. Assign each cause and effect a unique number.

    c. Transform the cause and effect into a Boolean graph.

    d. Identify the constraints describing the combination of causes and effects.

    e. Convert the graph into a limited-entry decision table.

    f. Convert the decision table into test cases.

4.1.3.4 Error Guessing

This paragraph should describe error guessing. The basic idea is to enumerate a list of possible errors or error-prone situations and then write test cases based on the list.

4.2 TEST METHODS AND CONSTRAINTS

The following paragraphs should describe the methods used and any limitation imposed on acceptance testing requirements.

4.2.1 Acceptance Test Conditions

This paragraph should indicate whether the test is to be made using the normal system input and data base, or whether a special set of inputs and the data base is to be used.

4.2.2 Extent of Acceptance Test

This paragraph should indicate the extent of the testing to be employed. Where total testing is not to be employed, the test requirements should be presented either as a percentage of some well defined total quantity or as a number of samples of discrete operating conditions or values. Also indicated should be the rationale for adopting limited testing.

4.2.3 Data Recording

This paragraph should indicate data recording requirements, including those data types not normally recovered from system operation. This paragraph should describe guidance for retention, accumulation, and dissemination of all normal system operation data; for example, input data, output reports, machine lists. Also include special test requirements such as utility prints, manual counts, and interim files.

4.2.4 Acceptance Test Constraints

This paragraph should indicate the anticipated limitations imposed on the test due to system or test conditions, such as limitations on timing, interfaces, equipment, personnel, and the data base.

4.3 TEST PROGRESSION

In case of progressive or cumulative test, an explanation should be included concerning the manner in which progression is made from one test to another so that the cycle or activity for each test is completely accomplished.

4.4 TEST EVALUATION

The following paragraphs should describe the criteria and techniques to be used in evaluating the test requirements.

4.4.1 Test Data Criteria

This paragraph should describe the rules by which test results will be evaluated; for example:

    a. Tolerances - The range over which a data value output by a system performance parameter can vary and still be considered acceptable.

b.   Samples - The minimum number of combinations or alternatives of input conditions and output conditions that can be exercised to constitute an acceptable test of the parameters involved.

c.   Counts - The maximum number of interrupts, halts, or other system breaks which may occur due to nontest conditions.

4.4.2 Test Data Reduction

This paragraph should describe the technique to be used for manipulation of the raw test data into a form suitable for evaluation, if applicable. The available techniques could include:

a.   Manual - The manual collection and collation of system test outputs into test sequence order followed by visual inspection of the results.

b.   Semiautomatic - The semiautomatic inspection of test results as obtained by data recording means using a test data reduction program followed by manual (visual) inspection of selected test results which do not lend themselves to complete reduction by automatic means.

c.   Automatic - The automatic inspection of test results specifically recorded for manipulation by the test data reduction program. Test results, as recorded, include all items of test significance. The test data reduction program contains an image of correct data output for an item-by-item comparison of data and provides a summary of an evaluated test as output

4.5 ACCEPTANCE SYSTEM INTERFACES

This paragraph should describe the interfaces with other systems including systems external to the project. For each system, the following should be specified:

a.   Description of operational considerations of data transfer, such as security considerations

b.   General description of data transfer requirements to and from the subject system and characteristics of communications media/systems used for transfer

c.   Formats and volumes of data to be interchanged

d.   Type of anticipated interface, such as manual or automatic

e.   Anticipated interface procedures, including telecommunications considerations

4.6   TEST METHODS AND DOCUMENTATION

This paragraph should provide a description of the methods and documentation being employed to satisfy the test requirements. A Test Analysis Report will be produced following the completion of the acceptance test activities.

SECTION 5   TEST DESCRIPTION

This section should describe an overall description of the test to be performed, methods of control, and step-by-step procedures to accomplish each test.

5.1   TEST DESCRIPTION

This paragraph should provide a general description of the testing to be performed. Acceptance testing is the testing of a complete system or subsystem, and is usually conducted using live data in an operational environment.

5.2   TEST CONTROL

The following paragraphs should describe the methods used to perform the tests and the manner in which the inputs and outputs are controlled.

5.2.1   Acceptance Test Means of Control

This paragraph should indicate what means the test is to be controlled by:

   a.   Manual Means - Manual inspection of necessary inputs and manual control of test sequence

   b.   Semiautomatic Means - Manual insertion of necessary inputs and automatic (test program) control of test sequence

   c.   Automatic Means - Preparation and use of a special test program to provide necessary input, conduct tests, monitor and record test results

5.2.2   Test Data

In each of the following paragraphs identify all test data, including inputs, commands, and outputs. Any security considerations should be included.

5.2.2.1   Input Data

This paragraph should describe the manner in which input data are controlled in order to:

a. Test the system with a minimum number of data types and values

b. Exercise the system with a range of bona fide data types and values which test for overload, saturation, and other "worst case" effects

c. Exercise the system with bogus data types and values which test for rejection of irregular inputs

5.2.2.2 Input Commands

This paragraph should describe the manner in which input commands are used to control:

a. Initialization of test

b. Halt or interrupt of test

c. Repeat of unsuccessful or incomplete test

d. Alternate modes of operation as required by test

e. Termination of test

5.2.2.3 Output Data

This paragraph should describe the manner in which output data are analyzed in order to:

a. Detect occurrence (or ultimate nonoccurrence) of output data

b. Record or identify media and location of all output data for indication of test performance

c. Evaluate output as a basis for continuation of test sequence

d. Evaluate test output against required output to assess the performance of the test

5.2.2.4 Output Notification

This paragraph should describe the manner in which output notification messages by the system concerning status or limitations on internal performance are controlled in order to:

a. Indicate readiness for test (normal operation condition)

b. Provide indications of irregularities in input test data or test data base due to normal or erroneous test procedures

c. Provide indications of irregularities in internal operations on test data due to normal or erroneous test procedures

d. Provide indications on the control, status, and results of the test as available from an auxiliary test supervisor program (if used)

5.3 TEST PROCEDURES

This paragraph should contain the step-by-step procedures to accomplish each test of the system. Each step should be assigned a test step number. This number, along with critical test data and test procedures information, should be tabulated for test control and the recording of test results.

5.3.1 Test Setup

If not stated elsewhere or by standing operating procedures, this paragraph should itemize the activities associated with setup of the computer facilities to conduct the test, including all routine machine activities from "power on" through "console setup" to "data input." Included should be the distribution of test documents, worksheets, and other forms.

5.3.2 Test Data Development

This paragraph should describe the development of data for testing. If there has been good planning and organization, the test data developed for unit, integration, or system testing often may be used for acceptance testing. Several considerations are involved in developing test data: program type (format, edit, process, update, select, report), complexity (program, system, calculations, update methods), and unique system requirements or constraints (allowing or disallowing certain normal or special conditions). All test data should be clearly labeled and stored for future use when program or system modifications are required.

Certain conditions must be tested independently, while others may be tested together. Several conditions must be created to test the system's capability to identify and react to the situation as it was designed. Refer to Appendix A, "Test Conditions," for a listing of test conditions.

This paragraph should describe the development of test data for acceptance testing. The test data used for the acceptance test must be comprehensive. All master files should be built from input transactions (edit exceptions for each type of error condition included), and transactions against all changeable fields. If the system is designed for multiple organizations, the test should demonstrate the system's ability to process both single and multiple entities. Restart procedures must correctly function any number of times per run. It should be possible to

restart in the same control group more than once, do several restarts one after another using different starting points, restart across different control groups, and restart on the correct record when control groups match on minor control (for example, same check number for two different companies). The forms alignment process for preprinted forms and restart procedures should be demonstrated. If selective processing is a feature of the system, it should be included in the test. The acceptance testing should be closely examined for the system's compliance with standards, and where applicable, perform certain benchmark checks. A sample checklist has been provided in Appendix I, "Acceptance Testing Checklist."

5.3.3 Test Initialization

This paragraph should itemize in test sequence order the activities associated with establishing the conditions of the first test starting with the equipment in the setup condition. Initialization may include such functions as:

a. Readout of control function locations and critical data from indicators and storage locations for reference purposes

b. Queuing of data input values for first test

c. Queuing of test support programs, if used

d. Coordination of personnel actions associated with test

5.3.4 Test Steps

This paragraph should itemize the test into test steps in test sequence order. It shall also include special operations, such as:

a. Visual inspection of test conditions

b. Data dumps

c. Instructions for data recording

d. Modifications of data base

e. Interim evaluation of test results

5.3.5 Test Termination

This paragraph should itemize in test sequence order, the activities associated with termination of the test such as:

a. Readout of critical data from indicators and location for reference purposes

b. Termination of operation of time-sensitive test-support programs and test apparatus

c. Collection of system and operator records of test results

5.3.6 Restart Procedures

This paragraph should provide a description of the restart procedures to ensure effective and efficient recovery from a temporary problem within the hardware or software systems. The description should include information to the operations center personnel concerning restart procedures that should be followed in the event of a system failure.

Appendix F

TEST ANALYSIS REPORT TABLE OF CONTENTS

Test Analysis Report

(This page intentionally left blank)

Appendix G

TEST ANALYSIS REPORT CONTENT DESCRIPTION

SECTION 1 GENERAL

This section should describe the purpose of the report, objectives and references, terms and abbreviations, and security considerations.

1.1 PURPOSE OF THE TEST ANALYSIS REPORT

This paragraph should describe the purpose of the Test Analysis Report as appropriate. The following issues should be addressed.

    a. Document the results of the testing.

    b. Provide a basis for allocating responsibility for deficiency correction, and follow-up.

    c. Provide a basis for the preparation of the statement of test completion.

    d. Establish user approval in the operation of the system.

1.2 PROJECT REFERENCES

This paragraph should provide a brief summary of the project objectives and identify the project sponsor and user. Also provided should be a list of applicable documents by author or source, reference number, title, date, and security classification. This paragraph should include, but is not limited to, the following:

    a. Users Manual

    b. Computer Operations Manual

    c. Acceptance Test Plan

    d. Training Plan

    e. Quality Assurance Plan

    f. Configuration Management Plan

1.3 TERMS AND ABBREVIATIONS

This paragraph should provide a listing or include a reference to an appendix containing all terms, definitions, or acronyms unique to this document and subject to interpretation by the user of the document. It should make reference to the Volume II glossary as

appropriate. This listing should not include item names or data codes.

1.4 SECURITY

This paragraph should describe any security considerations associated with the test analysis and the data being handled. In addition, governing documentation should also be identified.

SECTION 2 TEST ANALYSIS

This section should identify each test run, its function, performance, parameter comparison, and data performance as described in its specific test plan documentation.

2.1 TEST (IDENTIFY)

Each test should be separately identified in an individual paragraph and must be related to those presented in the test plan.

2.1.1 System Function (Identify)

Each system function, corresponding to the functions presented in Paragraph 4.1.2, "System Functions," of the Acceptance Test Plan, should be separately described.

2.1.1.1 Function Performance

This paragraph should describe the system capability as it has been demonstrated in one or more tests. It should also assess the manner in which the environment may be different from the operational environment and the effect of this difference on the capability, if any.

2.1.1.2 Parameter Performance

This paragraph should compare the parameter performance of the test with the parameter performance described in applicable documentation.

2.1.1.3 Data Performance

This paragraph should compare the I/O performance of the test with the I/O capabilities as described in applicable documentation.

2.n n-th TEST

This paragraph will describe, for "n" cases, subsequent tests in a manner similar to that used in Paragraph 2.1, "Test (Identify)."

2.1.n n-th System Function

This paragraph should describe, for "n" cases subsequent functions in a manner similar to Paragraph 2.1.1, "System Function (Identify)."

SECTION 3   SUMMARY AND CONCLUSIONS

This section should contain a summary of the capability of the system tested and a discussion of the deficiencies and their impact on the system.

3.1   DEMONSTRATED CAPABILITY

This paragraph should provide a general statement of the capability of the system as demonstrated by the test, compared with the performance requirements contained in the system Functional Requirements Definition. An individual discussion of conformance with specific requirements may be included on complex systems.

3.2   SYSTEM DEFICIENCIES

As required by the results of the testing, an individual statement should be provided for each deficiency in system operations, as measured against the test plan. Accompanying each deficiency should be a discussion of the impact:

    a. On system performance if the deficiency is not corrected

    b. On the system design if the deficiency is corrected, along with the assignment of organizational responsibility for the correction

3.3   SYSTEM REFINEMENTS

An itemization of improvements which can be realized in system design or operation, as determined during the test period, should be given. Accompanying each improvement should be a discussion of the added capability it provides the system and the impact on the system design.

(This page intentionally left blank)

Appendix H

TEST CONDITIONS

INTRODUCTION

This appendix provides a breakdown of test conditions involved in developing test data. While the following conditions are not exhaustive, they can serve as guidelines for test data preparation. The following letter items provide a listing of these test conditions.

    a. For the Data Base, the following are required:

        (1) Duplicate master records.

        (2) Out of sequence records.

        (3) Wrong master file.

    b. For New Records:

        (1) Set up a new record before the first record now on the master file.

        (2) Set up a new record after the last record now on the master file.

        (3) Set up three or four new records with consecutive record keys (with no existing records in between).

        (4) Set up a record for nonexistent organization.

        (5) Set up a new organization identification record (header record) at the beginning, end, and middle of the file. This step is applicable only if header identification records are used in the file(s).

        (6) Set up two or more new header records, one immediately after the other.

        (7) Set up a new record with record keys of zeros (0000).

        (8) Set up a new record with record keys of nines (9999).

        (9) Set up a new, but incomplete, record (only one or two fields of a possible ten).

    c. For Transactions:

        (1) Post transactions to the first record in the file.

(2) Post transactions to the last record in the file.

(3) Post transactions to records other than the first and last records in the file.

(4) Post transactions to a new record set up on the same run (if allowed).

(5) Post transactions to several consecutive records.

(6) Post various combinations of transactions to one record.

(7) Attempt to post transactions to nonexistent records that would be lower in sequence than the first existing record, higher in sequence than the last existing record, and between existing records as well as to several consecutive nonexistent records.

(8) Post transactions to a record to make balances or totals (inventory, deposits, etc.) go negative.

(9) Verify the effect on other fields in the record.

(10) Post several large amounts or quantities to a record to create an arithmetic overflow in total or balance fields. Examine results.

(11) If a header record followed by detail records is being used, create detail records for the first record in the file, the last record in the file, two consecutive records, one existing record, and several nonexistent records.

(12) Check for results of transactions with mispunched data in significant fields.

    d. For Inactive and Deleted Records:

(1) Drop the first record from each file.

(2) Drop the last record from each file.

(3) Drop three or four consecutive records from each file.

(4) Attempt to drop a nonexistent record.

(5) Code a record as "inactive" (no more data to be posted), and attempt to post data to that record in the same run.

(6) Attempt to post data to a record that was coded "inactive" in a previous run.

(7) Reclassify an "inactive" record as "active" and post transactions in the same run.

e. For Dates:

(1) Ensure that all date fields on records are being updated properly. Check data records, control records, and header identification records.

(2) Omit the date constant card from the input and examine results.

(3) Input a date with 00 and 13 month, 00 and 32 day, and an invalid year.

(4) Where appropriate, enter a date that exceeds the maximum interval between posting runs. Try a date more than seven days old, for example, and another date more than seven days in the future in a weekly update system.

(5) Make two posting runs using the same date.

f. For Logic and Processing Tests:

(1) Check all calculations that produce averages or percents with small, average, and large numbers (for example, 0,1, 5000, 9999).

(2) Create a condition for all division routines with a zero divisor.

(3) Create data for all exceptions and errors.

(4) Create test data for the minimum values for each field, such as minimum hourly rate and maximum salary.

(5) Enter data that are below the minimum value and above the maximum value for each field.

(6) Create data that includes multiple exceptions and errors in the same transaction. Enter several combinations.

g. For program type: Each type of program requires unique and different test data considerations. While the following suggestions are not exhaustive, they can serve as guidelines for test data preparation and expectations.

h. For Edit Programs, test data for an alphabetic field will include:

(1) Completely full field

(2) Blank Field

(3) First position in the field only

(4) First position in the field blank

(5) Mixed alphabetic and numeric characters

i. Test data for each field for a quantity or amount will include:

(1) Field of nine

(2) Field of zeros

(3) Field of blanks

(4) Exact lower limit of field, if any

(5) Exact upper limit of field, if any

(6) Typical quantity or amount, between any limit

(7) Value above limit, if any (not all nines)

(8) Value below limit, if any (not all zeros)

(9) Value with wrong sign (plus or minus)

(10) Alphabetic data in each field

j. For Update Program:

(1) Create data to set up several complete master file records.

(2) Develop change data for a nonexistent master file record.

(3) Create data to set up a new record when a record with the same key is already in the master file.

(4) Create data with a record key of zeros.

(5) Create data with a record key of nines.

(6) Create one or two items to establish a new but incomplete master file record.

(7) Create data to set up a new record and post changes in the same run.

k. For Process Program:

(1) Enter data that produce calculation results of low, average, and high values.

(2) Enter data that will create a condition for multiplication and/or division by zero.

(3) Enter data that will cause an out-of-balance condition on the file control record. Examine results.

(4) Enter data that will cause an arithmetic overflow.

(5) Enter data that will cause an out-of-balance crossfoot condition.

1. For Report Programs:

(1) Includes test data with minus values to ensure that minus signs are printed for each field, on each type of detail, and on total lines.

(2) Create test data with all nines in the fields to see if all digits are printed and not being overlaid by other data.

(3) Enter test data with all zeros in the fields to check zero suppression.

(4) Check all sum and calculation results.

(This page intentionally left blank)

Appendix I

ACCEPTANCE TESTING CHECKLIST

INTRODUCTION

This appendix provides a sample checklist to be used in performing acceptance testing. The test data used for the Acceptance Test must be comprehensive. The group responsible for acceptance testing should maintain a log of all exceptions taken. The final acceptance should be reviewed with programmers, analysts, and other participants involved in the development process. See Table I-1, "Acceptance Testing Checklist," for a sample checklist.

Acceptance Testing Checklist

	FUNCTION	YES	NO

File Controls:

Does the system provide an audit trail from program to program?

Are control totals on money fields and record counts by type of transaction?

Is the process control summary easy to identify, read, and understand?

Are any totals or messages printed on the console typewriter?

Are all input/output file sequences checked?

Are control records included in each master file? Transaction files?

Edit:

Is there a single edit program, or is editing performed throughout the system?

Are all "raw" input data edited?

Are all fields edited for all possible errors?

Are errors designated so that an error condition or error field will be quickly and easily identified in the printed output?

Are errors dropped or passed on into the system?

Does the system provide for reentry of error transaction?

Table I-1 (Page 1 of 4)

Acceptance Testing Checklist

| FUNCTION | YES | NO |

Update:

Does the system provide for additions, deletions, and changes?

Is the update calendar dependent? If so, what are the results of missing a cycle?

Will the system properly handle add records that are lower in sequence and higher in sequence than existing records?

Is the update process needed even if no file updating is required?

Can the update programs create initial files (without using a "dummy" file)?

Are the first and last records properly update?

If selective processing is allowed, is an updated master file written out?

Do all sequential files contain a control record? Is it update?

Processing:

Are the first and last records properly processed?

Are calculation results correct for low and high values?

Do division routines check for a zero divisor?

Table I-1 (Page 2 of 4)

Acceptance Testing Checklist

FUNCTION   YES   NO

Reports:

Do reports include proper headings?

Are fields headed? Properly aligned?

Are all numeric fields sign-controlled?

Are cents or units positions zero-suppressed?

If page headings change on control break, are
headings correct where page overflow occurs during
total break? Following control breaks?

Is the page number sequence correct?

Are totals identified and easy to read and
understand?

Is the end-of-report printout obvious?

Is "Current Date" on all reports?

Are there provisions for preprinted forms
alignment?

Are controls included for prenumbered documents
(rations, registers, I.D. cards, and invoices)?

Are the zero balance or negative checks printed?

Are documents counted? Is the ending number
maintained in control records?

Are minor totals rolled to major totals?

Table I-1 (Page 3 of 4)

Acceptance Testing Checklist

FUNCTION   YES   NO

Console Messages:

Are any console messages printed?

Are console messages absolutely necessary?

Are console messages in numeric code?   Narrative?
Easily understood?   Short and concise?

Are responses to console messages required?   Easily
entered?

Special Features:

Do programs have externally controlled options?

Is each option independent or does it need other
options for its execution?

Are program options controlled via console
typewriter?   Control card?

Do all long programs (30 to 40 minutes processing
time) contain the check-point feature?

Do any programs feature selective starting points?
Selective processing?   Selective total breaks?

Where a selective starting point is allowed, are
grand (final) totals correct?   Can the page number
be set?   Is the starting point identified?

When selective processing in a master file occurs,
is an updated master written out?

Where selective total breaks are allowed, are
totals on all selections correct?   Are totals
rolled?

Table I-1 (Page 4 of 4)

(This page intentionally left blank)

Appendix J

BIBLIOGRAPHY

DeMarco, Tom, Concise Notes on Software Engineering, New York, N.Y., Yourdon Press, 1979.

DeMarco, Tom, Controlling Software Projects, New York, N.Y., Yourdon Press, 1982.

Martin, James, Programming Real-Time Computer Systems, Englewood Cliffs, N.J., Prentice-Hall, Inc., 1965.

Myers, Glenford, The Art of Software Testing, New York, N.Y., John Wiley and Sons, 1979.

Wolberg, John R, Conversion of Computer Software, Englewood Cliffs, N.J., Prentice-Hall, Inc., 1983.

(This page intentionally left blank)

COMMENTS/REVISIONS

Technical publications under the Information Resources Management
(IRM) Standards and Guidelines Program (MCO 5271.1) are reviewed
annually. Your comments and/or recommendations are strongly
encouraged.

IRM Tech Pub Name: _____

IRM-_____-____ (Number)   Date of Tech Pub: _____

COMMENTS/RECOMMENDATIONS:_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Name/Rank:_____(optional)

Unit:_____(optional)

Mail To: Commandant of the Marine Corps (Code CCID-40)
          Headquarters, United States Marine Corps
          Washington, D.C.   20380-0001