October 1998

RECEIVED
MAY 11 1999
OSTI

Recommendations for a Software Quality Assurance Plan for the CMR Facility at LANL

S. D. Matthews

K. Adams

M. A. McQueen

LOCKHEED MARTIN

For additional copies or information on this document,
please contact

Chris Ischay
Lockheed Martin Idaho Technologies
P.O. Box 1625, MS 3454
Idaho Falls, ID 83415
Ph. (208)526-4382
Fax (208)526-8878
cpi@inel.gov

DISCLAIMER

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

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Recommendations for a Software Quality Assurance Plan for the CMR Facility at LANL

**S. D. Matthews
K. Adams
M. A. McQueen**

Published September 1998

**Idaho National Engineering Laboratory
Defense Program Environmental Surety Program
Lockheed Martin Idaho Technologies Company
Idaho Falls, Idaho 83415**

**Prepared for the
U.S. Department of Energy
Office of Defense Programs
Under DOE Field Office, Idaho
Contract DE-AC07-94ID13223**

CONTENTS

EXECUTIVE SUMMARY	4
1.0 PURPOSE	5
2.0 REFERENCES	6
3.0 MANAGEMENT	8
3.1 Organization	8
3.2 Tasks	9
3.2.1 General	9
3.2.2 Tasks for Chemical Analysis Software	9
3.2.3 Tasks for Radiological Analysis Software	10
3.3 Responsibilities	11
4.0 DOCUMENTATION	12
4.1 Purpose	12
4.2 Minimum Documents Requirements	12
4.2.1 Software Requirements Specification (SRS)	13
4.2.2 Software Design Description (SDD)	14
4.2.3 Software Verification and Validation Plan (SVVP)	15
4.2.4 Software Verification and Validation Report (SVVR)	15
4.2.5 User Documentation	16
4.2.6 Software Configuration Management Plan (SCMP)	16
4.3 Other	17
5.0 STANDARDS, PRACTICES, CONVENTIONS, AND METRICS	17
5.1 Purpose	17
5.2 Content	18
6.0 REVIEWS AND AUDITS	18
6.1 Purpose	18
6.2 Required Reviews	18
6.2.1 Software Requirements Review (SRR)	18

6.2.2 Design Review	18
6.2.3 Verification and Validation Report.....	19
6.2.4 User Manual	19
6.3 Other.....	19
7.0 TEST	20
8.0 PROBLEM REPORTING AND CORRECTIVE ACTION.....	20
9.0 TOOLS, TECHNIQUES, AND METHODOLOGIES	20
10.0 CODE CONTROL	20
11.0 MEDIA CONTROL	20
12.0 SUPPLIER CONTROL.....	21
13.0 RECORDS COLLECTION, MAINTENANCE, AND RETENTION	21
14.0 TRAINING.....	21
15.0 RISK MANAGEMENT	21

LIST OF FIGURES

Figure 1. Recommended Organizational Structure	8
---	---

LIST OF APPENDICES

APPENDIX A. REQUIREMENTS CHECKLIST FORM.....	23
APPENDIX B. SOFTWARE REQUIREMENTS SPECIFICATION REVIEWER'S FORM.....	26
APPENDIX C. SOFTWARE DESIGN DOCUMENT REVIEWER'S FORM.....	28
APPENDIX D. SOFTWARE VALIDATION REPORT REVIEWER'S FORM.....	30
APPENDIX E. USER MANUAL REVIEWER'S FORM.....	33
APPENDIX F. SOFTWARE PROBLEM REPORT AND EVALUATION FORM.....	35
APPENDIX G. CHANGE CONTROL FORM.....	38

EXECUTIVE SUMMARY FOR RECOMMENDATIONS FOR A SOFTWARE QUALITY ASSURANCE PLAN FOR THE CMR FACILITY AT LANL

The Nuclear Materials Technology (NMT) organizations 1 and 3 within the Chemical and Metallurgical Research (CMR) facility at the Los Alamos National Laboratory are working to achieve Waste Isolation Pilot Plant (WIPP) certification to enable them to transport their TRU waste to WIPP. This document is intended to provide not only recommendations to address the necessary software quality assurance activities to enable the NMT-1 and NMT-3 organizations to be WIPP compliant but also is meant to provide a template for the final Software Quality Assurance Plan (SQAP).

The critical software quality assuring activities to be WIPP compliant include:

1. Establishment of an *ad hoc* verification and validation organization, consisting of personnel who did not influence the acquisition or development of the software and are appropriately trained.
2. A graded approach to the software quality assuring activities based upon the appropriate level of risk mitigation defined by the risk associated with incorrect results or incorrect use of the results from the waste characterization and analysis software applications.
3. The minimum documentation requirements include:
 - Software Requirements Specification(s)
 - A Software Verification and Validation Plan
 - Software Verification and Validation Report(s)
 - A Software Configuration Management Plan
 - User Documentation
4. Creation of design documentation for newly developed software
5. Verification and validation activities performed by the verification and validation personnel to include:
 - A review of the Software Requirements Specification to ensure that the “right job is being done and that the job was done right.”
 - A review of the Software Design Description, if appropriate
 - A review of the User Documentation
 - Verification and validation tests to include acceptance tests for software acquired from a third party
6. A mechanism for problem reporting and corrective action
7. Configuration management

All the above software quality assurance activities are accepted as “best industry practice” to ensure that the software applications have the necessary integrity, quality and robustness. If these activities or software products do not currently exist, then a “retrofit” for the waste characterization and analysis software will be required.

RECOMMENDATIONS FOR A SOFTWARE QUALITY ASSURANCE PLAN FOR THE CMR FACILITY AT LANL

1.0 PURPOSE

The Nuclear Materials Technology (NMT) organizations 1 and 3 within the Chemical and Metallurgical Research (CMR) facility at the Los Alamos National Laboratory are working to achieve Waste Isolation Pilot Plant (WIPP) certification to enable them to transport their TRU waste to WIPP. This document is intended to provide not only recommendations to address the necessary software quality assurance activities to enable the NMT-1 and NMT-3 organizations to be WIPP compliant but also is meant to provide a template for the final Software Quality Assurance Plan (SQAP). This document specifically addresses software quality assurance for all software used in support of waste characterization and analysis. Since NMT-1 and NMT-3 currently have several operational software products that are used for waste characterization and analysis, these software quality assurance recommendations apply to the operation, maintenance and retirement of the software and the creation and development of any new software required for waste characterization and analyses. Specifically, this document is applicable to the following software:

“Computer software that manipulates or produces data that are, in turn, used to process, gather or generate information and whose output is relied upon to make design, analytical, operational, or compliance-related decisions with respect to the performance of the waste confinement, waste characterization, waste transportation, or waste acceptance processes. The application of these requirements shall be prescribed in written plan(s), policies, procedures or instructions.”¹

Likewise, the following types of software are exempted from the quality assuring activities recommended by this document:

“software that are considered to be ‘systems software,’ (e.g., operating systems, administrative and management systems, system utilities, compilers, assemblers, translators, interpreters, query languages, word processing programs, spreadsheets, database managers, and graphing programs) or other software that do not generate data that are used in the formulation of conclusions.” However, specific applications written for use within these types of software, e.g., detailed formulas or macros, that can be verified by hand calculations or other means shall meet the following two requirements of the Carlsbad Area Office’s “Quality Assurance Program Document” (QAPD), CAO-94-1012, Revision 1:

1. A listing of the version of the software used, and
2. Documentation that the specific application provides correct results for the specified range of input parameters.”²

¹ Section 6.2, Paragraph A, “U.S. Department of Energy, Carlsbad Area Office, Quality Assurance Program Document”, CAO-94-1012, Revision 1, April 1996

² Section 6.2, Paragraph B, op. cit. 1

2.0 REFERENCES

- ASME NQA-1-1989, "Quality Assuring Activities for Nuclear Facility Applications," The American Society of Mechanical Engineers.
- ASME NQA-2a-1990, "Quality Assurance Requirements of Computer Software for Nuclear Facility Applications," The American Society of Mechanical Engineers.
- Berlack, H. Ronald, 1990, *Software Configuration Management*, New York, NY: John Wiley & Sons
- CAO-94-1012, Revision 1, April 1996, "Quality Assurance Program Document," U.S. Department of Energy Carlsbad Area Office.
- Deutsch, Michael S. and Ronald R. Willis, 1988, *Software Quality Engineering, A Total Technical and Management Approach*, Engelwood Cliffs, NJ: Prentice Hall
- Freedman, Daniel P. and Gerald M. Weinberg, 1990, *Handbook of Walkthroughs, Inspections, and Technical Reviews, Third Edition*, New York, NY: Dorset House
- IEEE 610.12-1990, "Standard Glossary of Software Engineering Terminology," Institute of Electrical and Electronics Engineers.
- IEEE 730-1989, "Standard for Software Quality Assurance Plan," Institute of Electrical and Electronics Engineers.
- IEEE 730.1-1995, "Guide for Software Quality Assurance Planning," Institute of Electrical and Electronics Engineers.
- IEEE 828-1990, "Standard for Software Configuration Management Plans," Institute of Electrical and Electronics Engineers.
- IEEE 829-1983, "Standard for Software Test Documentation," Institute of Electrical and Electronics Engineers.
- IEEE 830-1993, "Recommended Practice for Software Requirements Specifications," Institute of Electrical and Electronics Engineers.
- IEEE 1012-1986, "Standard for Software Verification and Validation Plans," Institute of Electrical and Electronics Engineers.
- IEEE 1016-1987, "Recommended Practice for Software Design Descriptions," Institute of Electrical and Electronics Engineers.
- IEEE 1016.1-1993, "Guide to Software Design Descriptions," Institute of Electrical and Electronics Engineers.
- IEEE 1028-1988, "Standard for Software Reviews and Audits," Institute of Electrical and Electronics Engineers.

- IEEE 1042-1987, "Guide to Software Configuration Management," Institute of Electrical and Electronics Engineers.
- IEEE 1059-1993, "Guide for Software Verification and Validation Plans," Institute of Electrical and Electronics Engineers.
- IEEE 1062-1993, "Recommended Practice for Software Acquisition," Institute of Electrical and Electronics Engineers.
- Implementation Guide for use with 10 CFR Part 830.120, Quality Assurance, G-830.120-Rev. 0, U.S. Department of Energy, April 15, 1994.
- Los Alamos National Laboratory, December 31, 1996, *CST*LIMS Quality Management Plan*, Chemical Science and Technology Division
- Los Alamos National Laboratory, December 31, 1996, *CST*LIMS Quality Management Checklist*, Chemical Science and Technology Division
- Los Alamos National Laboratory, November 14, 1996, *Analytical Chemistry Master Software Quality Management Plan*, Chemical Science and Technology Division
- Pressman, Roger S., 1997, *Software Engineering: A Practitioner's Approach, Fourth Edition*, Engelwood Cliffs, NJ: Prentice-Hall
- TRU Waste Characterization Program, August 8, 1997, *TWCP Quality Procedure: Software Management*, TWCP-QP-1.1-006, R.6
- 10 CFR 830.120, April 1994, "Quality Assurance Requirements," *Code of Federal Regulations*, Office of the Federal Register.
- 40 CFR 194, February 9, 1996, "Criteria for the Certification and Re-Certification of the Waste Isolation Pilot Plant's Compliance with the 40 CFR Part 191 Disposal Regulations," *Code of Federal Regulations*, Office of the Federal Register.

3.0 MANAGEMENT

3.1 Organization

The diagram below recommends the necessary organizational structure for the software quality assuring activities for the waste characterization and analysis software used within the CMR facility.

Figure 1. Recommended Organizational Structure

As can be seen in Figure 1, the NMT-1 and NMT-3 organizations independently acquire, develop and maintain any necessary software products. Also, as indicated in Figure 1, the software quality assuring activities and the Software Configuration Control Board (SCCB) for the NMT-1 and NMT-3 organizations can share personnel as necessary to ensure the integrity and quality of their organization's software products. The managers for NMT-1 and NMT-3 are responsible to ensure that all required software quality activities are performed and that cognizant personnel who did not influence the creation, development or use of the product perform the verification and validation activities and are represented on the SCCB. A minimum of three independent and knowledgeable personnel should perform the verification and validation activities, and it is recommended that the SCCB should have a scientist cognizant of the software, a quality representative, the person responsible for the configuration management, the system administrator, the performing organization task leader, and, optionally, a trained and knowledgeable software engineer.

3.2 Tasks

3.2.1 General

Since this SQAP template is intended to programmatically address all software used by the NMT-1 and NMT-3 organizations, these recommendations will be applicable from inception to retirement to any software applications used for waste characterization and analysis for TRU wastes destined for WIPP. Consequently, the requisite tasks will span the entire software life cycle. The necessary tasks include the creation of a software requirements specification, test cases, and a user manual. Other factors that must be considered by the verification and validation personnel include whether the software is being newly created, been previously developed, vendor supplied software, developed under another QA program or whether the software is used for chemical waste characterization and analysis or for radiological waste characterization and analysis.

3.2.2 Tasks for Chemical Analysis Software

All the quality assuring activities are risk based, i.e., the activities are intended to mitigate the risks associated with incorrect results or the incorrect use of the results, and there are less health and environmental risks associated with the chemical waste characterization and analysis results than from their radiological counterpart. Thus, the requisite SQA tasks for software used for chemical waste characterization and analysis can have more flexibility.

The following subsections define the necessary tasks for software QA used to provide chemical waste characterization and analysis.

3.2.2.1 Tasks for Vendor Software, Tasks for Software Developed Under Other QA Programs, Tasks for Existing Software.

The first and most important task for vendor software, existing software or software developed under another QA program is to document what the requirements for the application are. These requirements should be reviewed by the VVC personnel to ensure that the task is being addressed completely and accurately. Test cases against these requirements should be derived next, followed by documentation of the software verification and validation, which can include acceptance test cases, hand calculations, comparison with other validated software of similar purpose, or empirical data and information from confirmed published data and correlations with technical literature. An independent review of the verification and validation methods is a necessary step to ensure accuracy and completeness. The next task is to ensure that a configuration baseline is established for each software product and that all configuration items are managed and controlled. A configuration status accounting performed by a cognizant individual(s) appointed by the SCCB will provide the status of the configuration baseline and verify that all the configuration items exist which are required to support a software product. The remaining task is the development of user documentation of sufficient quality such that "any qualified user (i.e., one having adequate background) can 'set up' and run the software and properly respond to errors."³

3.2.2.2 Tasks for Newly Developed Software. Software being developed or modified by NMT-1 or NMT-3 specifically to support chemical waste characterization or analysis must first have the software requirements documented and reviewed by the verification and validation personnel. The design developed from the requirements must also be documented and reviewed to ensure the software

³ Paragraph 6.3.6, op. cit. 1

design is requirements driven. The subsequent task is to ensure that the developed software satisfies its requirements by accepted verification and validation techniques in order to ensure that the software is “doing the right job and the job was done right.” The verification and validation techniques include, but are not limited to, test cases, hand calculations, comparison with other validated software of similar purpose, or empirical data and information from confirmed published data and correlations with technical literature. Configuration management to ensure the integrity of the developed software becomes the critical task once the software has satisfied its verification and validation criteria. The final task is the creation of user documentation that includes “as a minimum:

- A. the software name and version identifier;
- B. statement(s) of functional requirements and system limitations, including hardware;
- C. an explanation of the mathematical model(s) and derivation of the numerical methods used in the software design. Physical and mathematical assumptions on which the software is based shall be included along with an explanation of the capabilities and limitations inherent in the software;
- D. user instructions that describe user interaction with the software, user messages initiated as a result of improper input and how the user can respond, the identification and description of input and output specifications and formats, and input parameters.”⁴

3.2.3 Tasks for Radiological Analysis Software

As stated previously, the software quality assuring tasks are specifically meant to mitigate the risks that could occur if the computational results are incorrect or incorrectly used. Since any software result used to provide decision support for radiological waste characterization or analysis inherently has more health and environmental risks associated with its results than its chemical waste counterpart, the quality assuring activities for radiological analysis software are more rigorous than those required for chemical analysis software.

The following subsections provide guidance for the requisite SQA tasks for software used to provide waste characterization and analysis for radiological wastes.

3.2.3.1 Tasks for Vendor Software, Tasks for Software Developed Under Other QA

Programs. Elicitation and documentation of the software requirements are paramount for radiological software acquired via a vendor or another organization. These documented requirements must be reviewed by the verification and validation (V&V) personnel to ensure their accuracy and completeness. In addition, prior to the actual use of the software, the adequacy of the vendor software documentation to support operation and maintenance also must be evaluated by the V&V personnel. After the requirements are reviewed, verification and validation tests with acceptance criteria must be documented by the cognizant scientist and reviewed by the V&V personnel. The results of the acceptance tests, and any necessary regression tests, are documented in a verification and validation report by the V&V team members. All problem reporting and corrective actions for any extant errors must be recorded per paragraph 6.9 of the QAPD. The final tasks are configuration management of the software and documentation of a user manual to permit “any qualified user (i.e., one having adequate background) to ‘set up’ and run the software and properly respond to errors.”⁵ If a vendor or other organization already supplied user documentation, then an independent review by the verification and validation personnel will be necessary to ensure that the user documentation satisfies paragraph 6.8.6 of the QAPD. If the

⁴ op. cit. 1

⁵ op. cit. 1

user documentation is incomplete per the QAPD requirements, then addenda must supplement the existing user documentation.

3.2.3.2 Tasks for Newly Developed Software, Tasks for Existing Software. Documentation of both the functional and non-functional requirements for existing software or software being developed must be the first priority. If no requirements documentation exists, then the requirements may need to be “reverse engineered” from the source code. Similarly, the design must be documented. Like the requirements documentation, the design may have to be reverse engineered. Both the requirements and design must be independently reviewed by the V&V personnel to assure the requirements and the design satisfy the application requirements.

Following the review of the requirements, the test cases with acceptance criteria must be developed against the requirements. The test cases not only must address operational conditions but also cases to ensure that any unintended functions cannot occur. The test results must be documented. A traceability matrix will ensure the existence of an “audit trail” so that the test cases are traceable to the code functions, the code is traceable to the design, and the requirements are the design basis.

After the code has been checked out and installed, implementation of a baseline and change control are necessary. Thereafter, any proposed change must be evaluated and approved or rejected by a SCCB.

3.3 Responsibilities

All software quality assuring activities may be performed by any competent individual(s) within NMT-1 or NMT-3 “other than those who performed the software design, but the individuals may be from the same organization, including the designer’s supervisor, provided the supervisor:

1. did not specify a singular design approach;
2. did not rule out certain design considerations;
3. did not establish the design inputs used; and
4. is the only individual in the organization competent to perform the verification or validation.”⁶

Thus, the software quality assuring activities, including reviews, verification and validation, and configuration management can be performed by any cognizant personnel within NMT-1 or NMT-3 provided they did not either work on or influence the development of any software product. It should be noted that best industry practice recommends that the software quality assurance personnel are organizationally independent from the developer/user organization to assure that any schedule or budget influences are minimized on the quality assuring activities.

⁶ Section 6.6, paragraph B, op. cit. 1

4.0 DOCUMENTATION

4.1 Purpose

This section identifies all software documentation that is required for WIPP certification. The types of software used and how it is used varies at the CMR facility; therefore, the specific documentation required will also vary.

4.2 Minimum Documentation Requirements

The following documents are recommended by the Institute of Electrical and Electronics Engineers in Std. 730.1, "Guide for Software Quality Assurance Planning," to assure quality software development and implementation. Specific requirements for each document for NMT-1 and NMT-3 are as listed.

- Software Requirements Specification (SRS) is required for:
 - 1) Vendor supplied software
 - 2) Software developed by other QA programs
 - 3) Software developed by NMT-1 and NMT-3
 - 4) Existing software will require a retrofitted SRS
- Software Design Description (SDD) is required for:
 - 1) Software developed by NMT-1 and NMT-3
 - 2) Existing software will require a retrofitted SDD
- Software Verification and Validation Plan (SVVP):
 - 1) Software documentation requirements will be included in the INEEL report number INEEL/EXT-98-00706, "Recommended Software Verification and Validation Plan for the Chemical and Metallurgical Research Facility at Los Alamos National Laboratory."
- Software Verification and Validation Report (SVVR):
 - 1) NMT-1 and NMT-3 must complete and document a SVVP according to guidelines provided in INEEL report number INEEL/EXT-98-00706.
- User Documentation:
 - 1) Vendor supplied documentation may have to be supplemented to comply with the QAPD.
- Software Configuration Management Plan (SCMP):
 - 1) Software documentation requirements will be included in the INEEL report number INEEL/EXT-98-00705, "Recommended Software Configuration Management Plan for the Chemical and Metallurgical Research Facility at Los Alamos National Laboratory."

IEEE Standards are recommended as guides to mitigate risk. These guides are used as tools only and should be adapted for specific software as appropriate. The governing documents for software documentation are as follows:

- CAO-94-1012, Rev. 1, "Quality Assurance Program Document," Section 6, Software Requirements, U. S. Department of Energy Carlsbad Area Office
- ASME NQA-2a-1990
- 40 CFR Part 194
- 10 CFR 830.120
- DOE Order 5600.6C
- TWCP-QP-1.1-006, R.6, "TRU Waste Characterization Program; TWCP Quality Procedure"

If there is conflicting information in CAO-94-1012, Rev. 1 and TWCP-QP-1.1-006, R.6, the CAO document prevails.

4.2.1 Software Requirements Specifications (SRS)

Written documentation of requirements is essential for all software. Section 6.8.2 of the QAPD, *Requirements Documentation*, states that "Software requirements documentation shall outline the requirements that the proposed software must satisfy. The documentation shall, as applicable, address the following:

- 1) Functionality - the functions the software is to perform;
- 2) Performance - the time-related issues of software operation such as speed, recovery time, response time, etc.;
- 3) Constraints - those imposed on implementation activities – any elements that will restrict design options;
- 4) Attributes – non time-related issues of software operation such as portability, acceptance criteria, access control, maintainability, etc.; and
- 5) External interfaces - interactions with people, hardware, and other software."

Section 6.8.2 also states that "software requirements shall be traceable throughout the software development cycle." A system for tracking requirements changes should therefore be created and maintained for audit purposes.

Appendix A provides a checklist for the items to consider when developing software requirements.

Documentation for all waste characterization and analyses software are governed by the stringent methods quality assurance methods of ASME NQA-2a-1990, as invoked by 40 CFR Part 194. Requirements documentation should be tailored as appropriate for all software whether it is vendor supplied, developed under other QA programs, or developed by NMT-1 and NMT-3. However, standards may not have to be as stringently applied to vendor software and software created by other organizations. Requirements documentation must be retrofitted for software that is already in existence.

IEEE Standard 830-93, "Recommended Practice for Software Requirements Specifications" suggests that the following issues be considered in writing requirements documentation:

- 1) Nature of the requirements
- 2) Environment of the requirements
- 3) Characteristics of a good requirement
- 4) Joint Preparation of the requirements

- 5) Requirements evolution
- 6) Prototyping
- 7) Ensure that the functional requirements only include the software requirements and do not include either design issues or project requirements.

The standard also suggests that requirements be:

- 1) Correct
- 2) Unambiguous
- 3) Complete
- 4) Consistent
- 5) Ranked for importance and/or stability
- 6) Verifiable
- 7) Modifiable
- 8) Traceable

Again, a strongly tailored subset of IEEE 830 would be acceptable for vendor supplied software and software created by other organizations.

Further explanation and example templates for documentation are provided in Standard 830-1983. Guidance may also be found in IEEE Standard 1233-96, "Guide for Developing System Requirements Specifications."

4.2.2 Software Design Description (SDD)

The Software Design Description should describe the technical design aspects as derived from the Software Requirements Specification. Section 6.8.3 of the QAPD, Design and Implementation Documentation, specifies that "documentation consists of a document or series of documents that:

- 1) Describe the major components of the software design as they relate to the software requirements;
- 2) Describe the theoretical basis, embodied mathematical model, control flow, control logic, and data structure(s) of the software;
- 3) Describe the allowable or prescribed ranges for inputs and outputs; and
- 4) Describe the design in a manner that can be translated into code."

Section 4.3.2 in the TWCP Quality Procedure (TWCP-QP-1.1-006,R.6) on Software Management also provides reference to design documentation.

It should be noted that design documentation is not required for procured software. Design descriptions for waste characterization and analyses are required for new software developed by NMT-1 and NMT-3. If existing software does not have a design description, then a retrofitted design description must be created.

Reference to IEEE Standard 1016-1987, "Recommended Practice for Software Design descriptions," and IEEE Standard 1016.1-1987, "Guide to Software Design Descriptions," can provide helpful assistance in creating technical design descriptions. Standard 1016.1 suggests the following as a minimum "set of design entity attributes" for a design description:

- 1) Identification
- 2) Type
- 3) Purpose
- 4) Function
- 5) Subordinates
- 6) Dependencies
- 7) Interface
- 8) Resources
- 9) Processing
- 10) Data

4.2.3 Software Verification and Validation Plan (SVVP)

A Software Verification and Validation Plan is required by ASME NQA-2a-1990. Section 6.8.4 states that:

- A. "Software verification and validation documentation shall consist of associated plans and describe activities, including the results of reviews and tests, and the criteria for accomplishing the verification of the software throughout the software evolution. The documentation shall also specify the hardware and software configurations pertinent to the software verification and validation.
- B. Software verification and validation documentation shall be organized in a manner that allows traceability from the software requirements to both the software design and to the validated capabilities of the software."

The Idaho National Engineering and Environmental Laboratory will provide recommendations for the SVVP in external report number INEEL/EXT-98-00706, "Recommended Verification and Validation Plan for the Chemical and Metallurgical Research Facility at Los Alamos National Laboratory."

4.2.4 Software Verification and Validation Report (SVVR)

The SVVR is required per CAO-94-1012, Rev. 1, and report activities shall be conducted by verification and validation team members. The SVVR shall summarize observations and tests conducted from execution of the SVVP, report number INEEL/EXT-98-00706. IEEE Standard 730.1, section 3.4.2.4, recommends inclusion of the following information:

- 1) Summary of all life cycle validation and verification tasks
- 2) Summary of task results
- 3) Summary of anomalies and resolutions
- 4) Assessment of overall software quality
- 5) Summary from the verification matrix
- 6) Recommendations such as whether the software is, or is not, ready for operational use

Methods used for the V&V shall include black box testing, coverage analysis, and requirements traceability.

4.2.5 User Documentation

Per IEEE Standard 730, user documentation is that which specifies and describes “the required data and control inputs, input sequences, options, program limitations, and other activities or items necessary for successful execution of the software.” Section 6.8.6 of the QAPD (User Documentation) states that “User documentation should be sufficient to allow any qualified user (i.e. one having adequate technical background) to ‘set up’ and run the software and properly respond to errors. User documentation, as a minimum, shall include:

- A) the software name and version number;
- B) statement(s) of functional requirements and system limitations, including hardware;
- C) an explanation of the mathematical model(s) and derivation of the numerical methods used in the software design. Physical and mathematical assumptions on which the software is based shall be included along with an explanation of the capabilities and limitations inherent in the software;
- D) user instructions that describe user interaction with the software, user messages initiated as a result of improper input and how the user can respond, the identification and description of input and output specifications and formats, and input parameters;
- E) a description of any required training necessary to user the software; and
- F) user information for obtaining user and maintenance support.”

IEEE Standard 1063-87, “Standard for Software User Documentation,” would be a useful guide and provide further information on suggestions for user documentation.

Vendor supplied software documentation must be supplemented to comply with the QAPD.

4.2.6 Software Configuration Management Plan (SCMP)

The Software Configuration Management Plan (SCMP) is a required document. Recommendations for the SCMP will be provided by the Idaho National Engineering and Environmental Laboratory via external report number INEEL/EXT-98-00705. This plan describes the tasks, methodology, and tools required to assure that procedures and controls are being implemented as required.

Section 6.7.1 of the QAPD states that “Software shall be placed under configuration control as each configuration item is approved. A software baseline shall define the most recent approved software configuration. The configuration items and their associated documentation shall be traceable to one another. A labeling system for configuration items shall be implemented that:

- 1) uniquely identifies each configuration item;
- 2) identifies changes to configuration items by revision or version identifier; and
- 3) provides the ability to uniquely identify each approved configuration of the revised software that is available for use.”

Section 4.4 of the TWCP-QP-1.1-006, R.6, provides guidance for “Configuration Identification and Status Accounting.” Step one of the procedure requires maintenance of a baseline software list, which “shall contain the following for each baseline code according to the instructions in the Software Requirements Checklist Form . . .”

- code name and version number,
- code version date,
- code sponsor name,
- code classification
- RD version number,
- VVP version number,
- DD version number
- ID version number,
- UM version number,
- list of approved users (may be listed by name, organization or group, or task),
- list of approved system software/hardware configurations,
- list of outstanding Software Problem Report (SPR) numbers, and
- status of approved changes which are in progress
- periodic review dates
- code version retirement date.

Steps two through five contain guidance for document/form maintenance, revisions, distribution and review for code retirement. Section 4.5, "Change Control," of the TWCP "provides the process for requesting, controlling and implementing changes to software configuration baselines." Section 4.7 lists steps required for access control, and specifies that "access control is not required for:

- cases where execution of production baseline software presents no quality issues, and
- software which is executed for an analysis which contains documentation of the computer and software executable used in the analyses."

However, "For all other situations, controls shall be established to the extent possible to permit authorized and prevent unauthorized access of the software." Section 4.8 contains criteria for problem reporting. Appendix F is the example copy of the "Software Problem Report and Evaluation Form" from the TWCP.

More detailed guidance will be provided in the SCMP.

4.3 Other

Not applicable.

5.0 STANDARDS, PRACTICES, CONVENTIONS

5.1 Purpose

The standards applicable to software quality assurance for the NMT-1 and NMT-3 organizations are:

- TRU Waste Characterization Program, TWCP Quality Procedure, TWCP-QQP-1.1-0006,R.6
- ASME NQA-1-1997, Quality Assurance Requirements for Nuclear Facility Applications
- CAO-94-1012, Revision 1, U.S. Department of Energy, Carlsbad Area Office, Qualit Assurance Program Document, April, 1996

- 10 CFR Part 830.120, Quality Assurance, April, 1994

5.2 Content

Neither the NMT-1 nor the NMT-3 organization currently have any mandated documentation, coding, or testing standards.

6.0 REVIEWS AND AUDITS

6.1 Purpose

Reviews are essential to ensure the necessary quality assurance for NMT-1 and NMT-3 software. As stated in Section 3.3, the reviews may be performed by any competent individual(s) within NMT-1 or NMT-3 “other than those who performed the software design, but the individuals may be from the same organization, including the designer’s supervisor, provided the supervisor:

1. did not specify a singular design approach;
2. did not rule out certain design considerations;
3. did not establish the design inputs used; and
4. is the only individual in the organization competent to perform the verification or validation.”⁷

The presenters at the reviews must be the cognizant engineers responsible for the acquisition or development of the software product. The review chairperson should be the software configuration control board chairperson or his/her designee. The VVC reviewers satisfying the above criteria are recommended to be on the review team for the software requirements, the verification and validation report, and the user’s manual.

6.2 Required Reviews

6.2.1 Software Requirements Review (SRR)

The requirements review must ensure that the requirements are complete, verifiable, consistent, and technically feasible. The review shall also assure that the requirements will result in software that correctly performs its intended function. Appendix B provides a template for the requirements review.

6.2.2 Design Review

A design review is only necessary for software developed or modified by the NMT-1 or NMT-3 organizations. The design review must satisfy the criteria defined by NQA-1, Supplement 3S-1, “Supplementary Requirements for Design Control” in which the reviewers must assure that the following are addressed:

- “Were the design inputs correctly selected?”

⁷ Section 6.6, paragraph B, op. cit. 1

- “Are assumptions necessary to perform the design activity adequately described and reasonable? Where necessary, are the assumptions identified for subsequent reverifications when the detailed design activities are completed?”
- “Was an appropriate design method used?”
- “Were the design inputs correctly incorporated into the design?”
- “Is the design output reasonable compared to the design inputs?”
- “Are the necessary design input and verification requirements for interfacing organizations specified in the design documents or in supporting procedures or instructions?”

The design review also shall evaluate the technical adequacy of the design approach, and assure internal completeness, consistency, clarity, and correctness of the software design and shall verify that the software design is traceable to the requirements.⁸ Appendix C provides guidance for the necessary design review topics.

6.2.3 Verification and Validation Report

The Verification and Validation Report Review not only ensures that all necessary verification and validation tests have been performed but also ensures that all the necessary software components and documentation exist for the establishment of a configuration baseline. Appendix D provides a template for the Verification and Validation Report Review.

6.2.4 User Manual

The User Manual Review shall assure that the user documentation contains, as a minimum:

- “The software name and version identifier;
- “Statement(s) of functional requirements and system limitations, including hardware;
- “An explanation of the mathematical model(s) and derivation of the numerical methods used in the software design. Physical and mathematical assumptions on which the software is based shall be included along with an explanation of the capabilities and limitations inherent in the software;
- “User instructions that describe user interaction with the software, user messages initiated as a result of improper input and how the user can respond, the identification and description of input and output specifications and formats, and input parameters;
- “A description of any required training necessary to use the software; and
- “User information for obtaining user and maintenance support.”⁹
- Guidance to allow any qualified user to effectively and correctly use the system.

Appendix E provides a checklist to aid the review of the User Manual.

6.3 Other

The NMT-1 and NMT-3 management shall have the right to request any other reviews deemed appropriate to assure the quality of the software products being used for waste characterizations and analyses.

⁸ Paragraph 7.2, ASME NQA-2a-1989, Subpart 2.7, “Quality Assurance Requirements of Computer Software for Nuclear Facility Applications.”

⁹ Paragraph 6.8.6, op. cit. 1

7.0 TEST

All the necessary tests are explicitly addressed within the “CMR Verification and Validation Plan”. It should also be noted that all test cases must be traceable to the requirements and should be developed for the purpose of uncovering residual errors within the software.

8.0 PROBLEM REPORTING AND CORRECTIVE ACTION

Both Paragraph 6.9 of the Carlsbad Area Office Quality Assurance Program Document, CAO-94-1012, Revision 1, and Section 4.8 of the TRU Waste Characterization Program “TWCP Quality Procedure”, TWCP-QP-1.1-006,R.6, are applicable to any software errors discovered during use. The QAPD further mandates that all “corrective actions shall ensure that:

1. problems are identified, evaluated, documented, and, if required, corrected;
2. problems are assessed for their impact on past and present uses of the software;
3. changes to software are in accordance with the software configuration management requirements of this section of the QAPD; and
4. results are provided to the affected users along with any revised software documentation.”¹⁰

Appendix F provides the template for the “Software Problem Report and Evaluation Form”.

When a software defect or error is discovered, the response is dependent upon whether the software was acquired from a third party or developed within the NMT-1 or NMT-3 organizations. If the software was acquired from a third party, then the software originator must be notified of the existing error. Otherwise, the Software Configuration Control Board shall be notified to ensure the proper baseline is updated for the corrective or adaptive maintenance. Then the Software Configuration Control Board shall notify the code sponsor that corrective/adaptive maintenance is necessary.

9.0 TOOLS, TECHNIQUES, AND METHODOLOGIES

Not applicable.

10.0 CODE CONTROL

The “CMR Configuration Management Plan” will address all the necessary details for configuration management for NMT-1 and NMT-3 waste characterization and analysis software and the associated data. Appendix G provides guidance for software change control.

11.0 MEDIA CONTROL

Not applicable.

¹⁰ Paragraph B, Section 6.9, op. cit. 1

12.0 SUPPLIER CONTROL

Section 6.3 of the QAPD and Section 4.1 of the TWCP recommends appropriate practices when purchasing or acquiring software from a third party. IEEE Std. 1062-1993, "IEEE Recommended Practice for Software Acquisition" also provides useful guidance for acquiring software.

13.0 RECORDS COLLECTION, MAINTENANCE, AND RETENTION

This Software Quality Assurance Plan shall be configuration controlled and retained as a baseline for all software quality assuring activities for NMT-1 and NMT-3 waste characterization and analysis activities.

14.0 TRAINING

The QAPD mandates that all the NMT-1 and NMT-3 personnel "shall be trained and qualified to ensure that they are capable of performing their assigned tasks."¹¹ This statement is especially applicable to software quality assuring tasks, which are not part of the previously defined job descriptions for NMT-1 and NMT-3 personnel. Sections 1.2.1 and 1.2.2 of the QAPD delineate the specific training requirements.

15.0 RISK MANAGEMENT

Not applicable.

¹¹ Section 1.2, op. cit. 1

APPENDIX A
Requirements Checklist Form

Los Alamos
National
Laboratory

WASTE
ISOLATION
PILOT
PLANT

Software Requirements Checklist Form

Effective: 8/28/97

Procedure: TWCP-QP-1.1-006.R.6

Page 1 of 2

SCM Coordinator maintains this form, initialing and dating changes to sections 7-12

1. Code Sponsor Name: _____
2. Computer Code Name: _____
3. Computer Code Date: _____ 4. Computer Code Version No.: _____
5. Computer Code Classification: _____ (see Appendix A)

Software Classification	Software Document Requirements (version number & date) see Appendix A for guidance notes						
	RD	VVP	DD	ID	UM	VD	IC
Firmware DAS							
Purchased DAS							
Developed DAS							
Commercial (Off-the-shelf) SW							
QA-Vendor SW							
Vendor SW							
LANL SW							

7. Approved users (by name, organization, group, or task):

8. Approved system software/hardware configurations:

9. Software Problem Report (SPR) numbers	date initiated	date cleared

10. _____
Software Configuration Manager's Name (*print*) Signature Date

Software Requirements Checklist Form (Continued)

Effective: 8/28/97
Page 2 of 2

11. Change Control

Status of approved changes which are in progress

12. Periodic Reviews (at least once per year)

Initial baseline date: _____

review date	SCM initials	response	review date	SCM initials	response

13. Code Retirement

Retirement Date for Code Version: _____

14.

Code Sponsor's Name (*print*)

Signature

Date

15.

SCM Coordinator's Name (*print*)

Signature

Date

SCM Coordinator distributes completed form and associated documentation to the Code Sponsor and to the RMDC.

APPENDIX B
Software Requirements Specification Reviewer's Form

WASTE ISOLATION PILOT PLANT <small>Los Alamos National Laboratory</small>	<h2 style="margin: 0;">Requirements Document Reviewer's Form</h2> <p style="margin: 0;">Effective: 8/28/97</p>																																
Procedure: <u>TWCP-OP-1.1-006.R.6</u> Page <u>1</u> of <u>1</u>																																	
<p>Reviewer Instructions:</p> <p>Check "Yes" for each item reviewed and found acceptable.</p> <p>Check "No" for each item which requires further work.</p> <p>Check "N/A" for items which are not applicable.</p> <p>Check "N/R" for items not reviewed (only if there are multiple reviewers).</p> <p>Complete an RD Review Form for each reviewer.</p> <p>Prior to sign-off of the RD, all "No" items shall be appropriately addressed by the code sponsor so that "Yes" or "N/A" may be checked.</p> <p>Include this form as part of the baseline RD.</p> <p>The reviewer may document the rationale of the check marks in a memorandum included with this form in the RD.</p>																																	
<ol style="list-style-type: none"> 1. Functionality Are the functions that the software is to perform adequately identified? 2. Performance Are the time-related issues of software operations, such as speed, recovery time, or response time identified? 3. Design Constraints Are elements that will restrict design options identified? 4. Attributes Are non-time-related issues of software operation such as portability, acceptance criteria, access criteria, and maintainability identified? 5. External Interfaces Are required interactions with people, hardware, and other software identified? 6. Completeness Are the requirements complete? 7. Verifiability Can meeting the requirements be verified? 8. Consistency Are the requirements consistent with each other? 9. Technical Feasibility Are the requirements technically feasible and can they result in a useable code? 	<table style="width: 100%; border: none;"> <tr> <td>☒ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> <tr> <td>☐ Yes</td> <td>☐ No</td> <td>☐ N/A</td> <td>☐ N/R</td> </tr> </table>	☒ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R	☐ Yes	☐ No	☐ N/A	☐ N/R
☒ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
☐ Yes	☐ No	☐ N/A	☐ N/R																														
<p>10. Reviewer's Signature:</p> <table style="width: 100%; border: none; margin-top: 10px;"> <tr> <td style="width: 40%; border-bottom: 1px solid black; height: 20px;"></td> <td style="width: 40%; border-bottom: 1px solid black; height: 20px;"></td> <td style="width: 20%; border-bottom: 1px solid black; height: 20px;"></td> </tr> <tr> <td style="font-size: small;">Reviewer's Name (print)</td> <td style="font-size: small;">Signature</td> <td style="font-size: small;">Date</td> </tr> </table>					Reviewer's Name (print)	Signature	Date																										
Reviewer's Name (print)	Signature	Date																															

APPENDIX C
Software Design Document Reviewer's Form

APPENDIX D
Software Validation Report Reviewer's Form

Validation Document Reviewer's Form(Continued)

Effective: 8/28/97

Page 2 of 2

6. **Understandability of Documentation**

Are the validation methods, test data, results, and conclusions documented in a form that can be understood by an independent, technically competent individual?

☐ Yes

☐ No

☒ N/A

☐ N/R

7. **Reviewer's Signature:**

Reviewer's Name (*print*)

Signature

Date

SAMPLE
CONTACT THE CONTROLLED DOCUMENTATION
CENTER STAFF TO OBTAIN THIS FORM.

APPENDIX E
User's Manual Reviewer's Form

WASTE ISOLATION PILOT PLANT <small>Los Alamos National Laboratory</small>	<h2 style="margin: 0;">User's Manual Reviewer's Form</h2> <p style="margin: 0;">Effective: 8/28/97</p>		
Procedure: <u>TWCP-OP-1.1-006.R.6</u>		Page <u>1</u> of <u>1</u>	

Reviewer Instructions:

Check "Yes" for each item reviewed and found acceptable.

Check "No" for each item which requires further work.

Check "N/A" for items which are not applicable.

Check "N/R" for items not reviewed (only if there are multiple reviewers).

Complete a User's Manual Review Form for each reviewer.

Prior to sign-off of the User's Manual, all "No" items shall be appropriately addressed by the code sponsor so that "Yes" or "N/A" may be checked.

Include this form as part of the baseline User's Manual.

The reviewer may document the rationale of the check marks in a memorandum included with this form in the User's Manual.

Revision ☐ Yes ☐ No	Update ☐ Yes ☒ No	If update, restrict scope of review based on the level of the code change.
---	--	--

Does the user's manual contain as appropriate:

1. A statement(s) of functional requirements and system limitations?	☐ Yes	☐ No	☐ N/A	☐ N/R
2. An explanation of the mathematical model and numerical models?	☐ Yes	☐ No	☐ N/A	☐ N/R
3. Physical and mathematical assumptions?	☐ Yes	☐ No	☐ N/A	☐ N/R
4. The capabilities and limitations inherent in the software?	☐ Yes	☐ No	☐ N/A	☐ N/R
5. Instructions that describe the user's interaction with the software?	☐ Yes	☐ No	☐ N/A	☐ N/R
6. The identification of input parameters, formats, and valid ranges?	☐ Yes	☐ No	☐ N/A	☐ N/R
7. Messages initiated as a result of improper input and how the user can respond?	☐ Yes	☐ No	☐ N/A	☐ N/R
8. The identification and description of output specifications and formats?	☐ Yes	☐ No	☐ N/A	☐ N/R
9. A description of any required training necessary to use the software?	☐ Yes	☐ No	☐ N/A	☐ N/R
10. The identification of components of the code that were not tested?	☐ Yes	☐ No	☐ N/A	☐ N/R

11. Reviewer's Signature:

Reviewer's Name (<i>print</i>)	Signature	Date
----------------------------------	-----------	------

Appendix F
Software Problem Report and Evaluation Form

WASTE
ISOLATION
Los Alamos
National
Laboratory
PILOT
PLANT

Software Problem Report and Evaluation Form

Effective: 8/28/97

Procedure: TWCP-OP-1.1-006.R.6

Page 1 of 2

PART I - ERROR NOTICE CLASSIFICATION

Fields 1, 3-5 to be completed by the Code Sponsor.

Field 2 to be completed by SCM Coordinator

1. Affected Site Project Manager(s): _____
2. SPR No.: _____ 3. Date: _____
4. Computer Code Name: _____ 5. Computer Code Version No.: _____

Fields 6-13 to be completed by Code Sponsor

6. Computer code is installed on the following platform:
- Hardware System: _____ Operating System: _____
7. Dates error was effective: _____ 8. Error Classification: ☐ Major ☐ Minor
9. Does the error still exist? ☐ Yes ☐ No If no, date error was resolved: _____

10. Brief Description of Error:

11. Brief Description of Evaluation/Impact: (Describe how the error would be likely to affect analysis results.)

12. Recommendations: (Include information on how users may work around problem [if appropriate] and briefly describe expected correction actions.)

13. _____
Code Sponsor's Name (print) Signature Date

Note: Attach vendor supplied SPRs and additional documentation if necessary.

Software Problem Report and Evaluation Form (Continued)

Effective: 8/28/97

Page 2 of 2

PART II - ERROR NOTICE EVALUATION - FOR MAJOR ERRORS

To be completed by User

1. SPR No.: _____
2. List all analysis whose results or conclusions were affected by the software problem:

3. List the analyses that document the impact and resolution of the software problem:

Acknowledgment by the Site Project Manager and assigned User(s) that the appropriate action has been taken for all affected analysis

4.

_____ User's Name (<i>print</i>)	_____ Signature	_____ Date
_____ User's Name (<i>print</i>)	_____ Signature	_____ Date
5.

_____ Site Project Manager's Name (<i>print</i>)	_____ Signature	_____ Date
---	--------------------	---------------

Return form to SCM Coordinator. Attach a copy of Part I of this form to any revised analysis. SCM Coordinator to distribute completed form and associated documentation to the Code Sponsor and to the RMDC.

APPENDIX G

Change Control Form

Change Control Form (Software/Hardware/Baseline Document)

Effective: 3/28/97

Procedure: TWCP-QP-1.1-006.R.6

Page 1 of 2

This form is for proposal and approval of changes to production baseline software, and to system software and hardware used by more than one person for running production baseline software.

Code Sponsor and/or Computer Administrator Instructions
Complete form and sign field 8. Forward to SPM.

Site Project Manager Instructions

Check "Approve Changes" or "Disapprove Changes" box. If change is disapproved, provide justification. Forward a copy of approved form to the Code Sponsor and to the SCM Coordinator, and (for changes to system software or hardware) the Computer Administrator.

SCM Coordinator Instructions

Maintain a copy of the form and forward a copy to the RMDC.
For approved changes, notify all affected users of the impending change via memo.

1. Software Name: _____
2. Software Version No.: _____
3. Proposed Software Version No.: _____
4. Hardware/Software Platform: _____
5. Expected effective date of proposed change: _____

6. Proposed Changes: (include the description of the change, the rationale for the change, and (for changes to production software) identification of all affected baseline documents - attach pages as needed)

Requirements Document (RD)	Baseline Affected ☐ Yes ☐ No*	Required Resolution ☐ Revision ☐ Page Change ☐ Addenda
Version No: _____	*Rationale _____	
New Version No: _____		
Verification and Validation Plan (VVP)	☐ Yes ☐ No*	☐ Revision ☐ Page Change ☐ Addenda
Version No: _____	*Rationale _____	
New Version No: _____		
Design Document (DD)	☐ Yes ☐ No*	☐ Revision ☐ Page Change ☐ Addenda
Version No: _____	*Rationale _____	
New Version No: _____		
Validation Document (VD)	☐ Yes ☐ No*	☐ Revision ☐ Page Change ☐ Addenda
Version No: _____	*Rationale _____	
New Version No: _____		

Change Control Form

(Software/Hardware/Baseline Document) (Continued)

Effective: 3/28/97
Page 2 of 2

6. continued Proposed Changes:

Implementation Document (ID) ☐ Yes ☐ No* ☐ Revision ☐ Page Change ☐ Addenda
Version No: _____ *Rationale _____
New Version No: _____

User's Manual (UM) ☐ Yes ☐ No* ☐ Revision ☐ Page Change ☐ Addenda
Version No: _____ *Rationale _____
New Version No: _____

7. Describe the proposed changes to software, hardware, and/or documentation.

Specify verification/validation method to be used to determine impact of proposed change and to detect unintended effects and demonstrate that the modified software still complies with its specified requirements: *(include a discussion of required test cases to demonstrate acceptable performance of new code features and rationale for use of regression testing, if all test cases are not re-run)*

8. _____
Code Sponsor (printed) Signature Date

9. ☐ Approve of Proposed Change
☐ Disapprove of Proposed Change (provide justification, attach pages as needed)

10. _____
Site Project Manager's Name (print) Signature Date

11. _____
Software Configuration Manager Name (print) Signature Date

Change Control Form Instructions

This form is for proposal and approval of changes to production baseline software, changes to software documentation, and/or changes to system software and hardware.

Guidelines for selection of Revisions vs. Addendas

Revisions are always acceptable, and in some cases addenda's may be used. Note that the major version number of the baseline documents need not be the same as the major version number of the code. The major version number of baseline documents which are associates with a particular major version number of a code must be consistent.

Revisions are required when:

- There are any components moving from functionality not tested to functionality tested. In these cases, the document revisions will need to describe what functionality will be impacted and what additional test cases will be needed (describe in Proposed Changes Section of this form).
- A new production build (recompile) is needed.
- A code is changing version numbers.

Addenda:

- Cannot be used if a code has changed version numbers.
- Cannot be used if a prior addenda exists to the document.
- Can be used when the proposed changes only affect a small limited portion of a single document.

Code Sponsor Instructions

Complete items 1-6 and 8. Forward to the Site Project Manager (SPM).

Computer Administrator Instructions

Complete items 7 and 8. Forward to SPM.

Site Program Manager Instructions

Complete items 9 and 10. Forward form to the Code Sponsor and the SCM Coordinator.

SCM Coordinator

Return form to Code Sponsor for completion, or forward a copy of the completed form to the RMDC (after completing item 11). Maintain a copy of the form until proposed baseline changes are completed. For approved changes, notify all users on the approved user list of the impending change.

General Instructions

For each entry listed, additional pages may be attached as needed.

1. Software Name: Enter the name of the software as listed on the Baseline Inventory List.
2. Current Baseline Software Version Number: Enter the current software version number as listed on the Baseline Inventory List.
3. Proposed Software Version Number: Enter the proposed software version number.
4. Hardware/Software Platform: Enter the hardware platform on which the software resides and any applicable system software (required for the execution and use of the production baseline software).
5. Expected date of proposed changes: Enter the approximate date that all associated documentation will be completed and submitted to the SCM coordinator.
6. Proposed Changes: Use this section to describe in detail the changes each document will be undergoing. For documents that will have an addenda rather than a revision, justify why an addenda is being used. For each document, list the current document version number (as it appears on the Baseline Inventory List) and (if applicable) the new document version number.

Implementation Document (ID): check if this document is affected and how it will be updated. In general, all ID changes will be revisions, not addenda's. Describe what aspects of the coding will change.

Requirements Document (RD): Describe any features that are being changed, added, or deleted. Describe if any requirements are moving from not tested to tested. Include a discussion of required test cases to demonstrate acceptable performance of new code features. Provide rationale for regression testing if all existing test cases will not be rerun.

Verification and Validation Plan (VVP): Describe test cases and acceptance criteria that are being changed, added, or deleted. Discuss how these test cases demonstrate that the code adequately performs all tested functions.

Design Document (DD): Describe the extent of changes to the DD. Note how changes will be verifiable through testing or other means.

Validation Document (VD): Describe if the VD will change to reflect changes to the VVP or will be updated for other reasons.

User Manuals (UM): Describe what user instructions will be changed, added, or deleted.

General Instruction (continued)

7. System Software/Hardware Change Section

Describe proposed changes to system software and/or hardware. Describe expected impact, if any, to production baseline software which resides on the system. Describe how changes to system software and/or hardware will be tested. Discuss what regression testing of baseline software will be required or describe why no regression testing of production baseline software will be needed. If testing is needed, it must address the change to the system to verify that the change has been installed properly and works properly.

8. Code Sponsor or Computer Administrator signature

Code Sponsor signs for changes to baseline software.

Computer Administrator signs for changes to system software/hardware.

9. SPM selects a box from this line to indicate whether changes are approved or disapproved. If changes are disapproved, explain why.

10. SPM signature. After signing form, SPM forwards to SCM Coordinator.

11. SCM Coordinator signature. SCM Coordinator signs change control form or returns it to Code Sponsor or Computer Administrator for proper completion. After SCM Coordinator signature, forwards form to approved users and RMDC.