

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 1 of 23
-----------------------------------	--

Companywide	Program Requirements Document	For Additional Info: http://EDMS	Effective Date: 01/08/14
-------------	-------------------------------	---	--------------------------

Manual: 13—Quality Assurance Program

Change Number: 340543

*The current revision can be verified on EDMS.

1. PURPOSE

This program requirements document (PRD) identifies requirements and responsibilities for controlling the quality of computer *software* (see def.) and *safety software* (see def.). Unless otherwise noted in text below, requirements for safety software fall under the term software in this PRD.

2. APPLICABILITY

The requirements contained in this PRD shall be implemented per the approved company graded approach process and in compliance with the graded approach requirements specified in PRD-5071, “Quality Assurance Program.” Safety software is designated Quality Level 2 (QL-2) unless the software is part of a system or component determined to be QL-1. Safety software is further categorized as custom-developed, acquired, configurable, commercial design and analysis (D&A), or utility calculation software. Other quality-affecting software is designated QL-3.

3. RESPONSIBILITIES

3.1 Quality Assurance

- 3.1.1 The Quality Assurance organization is responsible for providing direction and assessment of the company software Quality Assurance Program, which includes, but is not limited to the following:
- A. Defining and disseminating company software Quality Assurance Program requirements and instituting *processes* (see def.) to satisfy those requirements
 - B. Preparing, implementing, and maintaining this PRD
 - C. Ensuring the contractual requirements, as specified in the appropriate contract clauses, are reflected in this PRD
 - D. Providing qualified *quality engineers* (see def.) to support the implementation and maintenance of the company software Quality Assurance Program
 - E. Conducting surveillances of user organizations to *verify* (see def.) implementation of the software Quality Assurance Program
 - F. Providing independent oversight of the software quality implementation to ensure compliance with applicable regulations and requirements

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 2 of 23
-----------------------------------	--

- G. Providing *guidance* (see def.) to facilities in establishing and implementing software Quality Assurance Program requirements
- H. Assigning a software quality assurance subject matter expert to assist the Idaho Cleanup Project with the implementation of software quality assurance requirements.
- I. Maintaining the Safety Software Inventory List.

3.2 Quality Engineer

3.2.1 The quality engineer is responsible for the following:

- A. Reviewing and concurring with department or user organization *software engineering* (see def.) documentation that implements the software Quality Assurance Program and processes
- B. Assisting Information Technologies (IT) and user organizations in identifying and resolving software quality assurance *issues* (see def.)
- C. Reviewing and approving software *procurement documents* (see def.).

3.3 Information Technology Organization

3.3.1 The IT organization is responsible for the following:

- A. Maintaining IT *procedures* (see def.)
- B. Maintaining the company's networks
- C. Providing software development and *testing* (see def.) support to projects
- D. Coordinating IT processes with the software owners and users
- E. Providing IT *Software Inventory Portfolio* (SIP) (see def.) and standard software support
- F. Providing cyber-security policies and procedures.

3.4 Project-Managed Software

3.4.1 The project-managed organizations are responsible for the following:

- A. Maintaining project-managed software procedures
- B. Developing software requirements and providing software development and testing
- C. Providing support for project-managed software.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 3 of 23
-----------------------------------	--

3.5 User Organization

3.5.1 User organizations are responsible for the following:

- A. Implementing the applicable software requirements as directed in company procedures in accordance with the requirements of this PRD
- B. Resolving software quality issues and implementing timely *corrective action* (see def.).

3.6 Facility Design Authority

3.6.1 The Facility Design Authority is responsible for identifying software specification, acquisition, design, development, *verification* (see def.) and validation (including inspection and test), *configuration management* (see def.), maintenance, and retirement for safety software.

3.7 Engineering

3.7.1 The Idaho Cleanup Project (ICP) chief engineer will:

- A. Establish and maintain ICP-wide engineering and configuration management procedural requirements
- B. Assess ICP engineering for performance per applicable procedures, codes and standards, and practices
- C. Manage ICP central design engineering and analysis functions.

3.7.2 The Area/Project chief engineer will:

- A. Direct, lead, and manage Operations Engineering at assigned Site areas
- B. Oversee all engineering and modification activities in assigned areas, including changes to new and existing configuration management structures, systems, or components (SSCs), to ensure that activities are performed per this PRD
- C. Identify computer systems within assigned areas. Assign system engineers to each computer system by evaluating system support needs and personnel resources and capabilities
- D. Ensure that the assigned system engineer has all the necessary technical training for the system to which he or she is assigned

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 4 of 23
-----------------------------------	--

- E. Recommend and manage modifications for multiple large projects, small projects, small facility modifications, and other tasks from initiation to completion.

3.7.3 The system engineer (or delegate) will:

- A. Determine the computer system configuration control requirements based on the quality level
- B. Develop a software management plan and maintain configuration management of *configuration items* (see def.).

3.8 Software Owners

3.8.1 Software owners will:

- A. Retain overall responsibility for assigned software and associated *software documentation* (see def.) and *software records* (see def.), which include:
 - 1. Project software management and software risk management, if applicable
 - 2. Providing software functional requirements
 - 3. Complying with procured software licenses
 - 4. Developing and maintaining software documentation, records, and SIP data
 - 5. Performing *acceptance testing* (see def.)
 - 6. Training
 - 7. Managing software operations and maintenance
 - 8. Ensuring unneeded software is retired
- B. Authorize use of software under their ownership
- C. Semiannually review the data in the SIP available on the CH2M-WG Idaho, LLC, (CWI) homepage under the IT link and document review using the “Perform Semi-annual Review” radio button.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 5 of 23
-----------------------------------	--

4. REQUIREMENTS

4.1 Companywide Applications

The requirements identified in this subsection meet the requirements in the following:

- A. Department of Energy (DOE) order, DOE O 414.1D, “Quality Assurance,” Attachment 5, “Safety Software Quality Requirements”
- B. “Quality Assurance Requirements for Nuclear Facility Application,” American Society of Mechanical Engineers (ASME) NQA-1-2008 with Addenda through NQA-1a-2009, Subpart 2.7, “Quality Assurance for Computer Software for Nuclear Facility Applications”
- C. DOE/RW-0333P, “Quality Assurance Requirements and Descriptions,” Revisions 10 and 20.

The requirements identified in this subsection apply to the entire company as defined by FWD-7, “Foreword.”

The requirements contained in this PRD shall be implemented per the approved company graded approach process and in compliance with the graded approach requirements specified in PRD-5071. Safety software is designated QL-2 unless the SSC quality level determination drives the quality level to 1. Safety software is further categorized as custom-developed, acquired, configurable, commercial D&A, or utility calculation software. Other quality-affecting software is designated QL-3.

4.1.1 Software engineering activities includes the following elements, as appropriate:

- A. Software acquisition method(s) for controlling the acquisition process for software and *software services* (see def.)
- B. Software engineering method(s) used to manage the *software life-cycle* (see def.) activities
- C. Application of standards, conventions, and other work practices that support the software life-cycle
- D. Controls for support software used to develop, operate, and maintain *computer programs* (see def.).

4.1.2 Software engineering method(s) shall be documented.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 6 of 23
-----------------------------------	--

- 4.1.3 The selected software engineering method shall ensure that software life-cycle activities are planned and performed in a traceable and orderly manner.
- 4.1.4 Work processes involving safety software shall be developed and implemented using this PRD and shall include the following elements:
- A. Facility design authority involvement in identifying software specification, acquisition, design, development, verification, and validation (including inspection and test), configuration management, maintenance, and retirement
 - B. Identification, documentation, and maintenance of a safety software inventory that contains the following information, as a minimum: software description, software name, version identifier, safety software designation (for example, safety system software, safety and hazard analysis software and design software, safety management and administrative controls software), quality level designation, specific nuclear facility where application is used, and the responsible individual
 - C. Established quality level for safety software as documented in the Quality Assurance Program
- NOTE:** *Software determined to be QL-1 or QL-2 is considered safety software. Safety software is designated QL-2 unless the SSC quality level determination drives the quality level to 1.*
- D. Established software types as documented in the Quality Assurance Program.
- NOTE:** *Five types of software have been defined: (1) custom developed software, (2) configurable software, (3) acquired software, (4) utility calculation software, (5) commercial D&A software.*
- 4.1.5 Using the safety software type established and approved in Item 4.1.4.D (above), select and implement the applicable software activities from the following list to ensure that safety software performs its intended functions:
- A. Software project management and quality planning
 - B. Software risk management
 - C. Software configuration management
 - D. Procurement and supplier management

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 7 of 23
-----------------------------------	--

- E. Software requirements identification and management
- F. Software design and implementation
- G. Software safety
- H. Verification and validation
- I. Problem reporting and corrective action
- J. Training of personnel in the design, development, use, and evaluation of safety software.

4.2 Software Acquisition

4.2.1 Procured Software and Services

- 4.2.1.1 Procured software and software services shall be procured in accordance with requirements in PRD-5075, “Procurement Document Control.”
- 4.2.1.2 Upon acceptance of a procured software application, the required elements of this PRD shall be determined and performed.
- 4.2.1.3 Procurement documents shall identify requirements for supplier’s reporting of software *errors* (see def.) to the *purchaser* (see def.) and, as appropriate, the purchaser’s reporting software errors to the supplier.

4.2.2 Otherwise Acquired Safety Software

- 4.2.2.1 PRD-5078, “Control of Purchased Items and Services,” and PRD-5069, “Dedication of Purchased Commercial Grade Items and Services,” shall be applied to the acquisition of safety software that has not been previously approved under a program consistent with this PRD for use in its intended application (for example, freeware, shareware, procured *commercial off-the-shelf* (see def.), or otherwise acquired software), shall be evaluated in accordance with the requirements of this PRD.
- 4.2.2.2 The acquired software shall be identified and controlled during the dedication process.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 8 of 23
-----------------------------------	--

- 4.2.2.3 The dedication process shall be documented and include the following:
- A. Identification of the capabilities and limitations for intended use as critical characteristics
 - B. Utilization of software management plans, *test plans* (software) (see def.), and *test cases* (software) (see def.) as the methods of acceptance to demonstrate the capabilities within the limitations
 - C. Instructions for use (such as, user manual) within the limits of the dedicated capabilities.
- 4.2.2.4 The dedication process shall be documented and the performance of the actions necessary to accept the software shall be reviewed and approved.
- 4.2.2.5 The resulting documentation and associated computer program(s) shall establish the current *baseline* (see def.).
- 4.2.2.6 Subsequent revisions of accepted software received from organizations not required to follow this PRD shall be dedicated in accordance with this section.

4.2.3 Otherwise Acquired Non-Safety Software

- 4.2.3.1 PRD-5078, “Control of Purchased Items and Services,” shall be applied to the acquisition of software.
- 4.2.3.2 The approval process shall be documented and include the following:
- A. Identification of the capabilities and limitations for intended use
 - B. Utilization of software management plans, test plans (software), and test cases (software) as the methods of acceptance as applicable
 - C. Instructions for use (such as, user manual).

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 9 of 23
-----------------------------------	--

4.3 Software Design Process

The software *design process* (see def.) shall be documented, approved by the responsible design organization, and controlled. This process shall include the following activities.

4.3.1 Identification of Software Design Requirements

- 4.3.1.1 Software design requirements shall be identified and documented and their selection reviewed and approved.
- 4.3.1.2 Software design requirements shall be traceable throughout the software life-cycle.
- 4.3.1.3 The software design requirements shall specify technical and software engineering (for example, software acquisition method, software engineering methods, application of standards, conventions, and other work practices, and the controls for support software used to develop, operate, and maintain computer programs) requirements, including security features (for example, vulnerability protection and cyber security). Identify applicable reference drawings, specifications, codes, standards, regulations, procedures, or instructions that establish software design requirement test, inspection, and acceptance criteria. Security requirements shall be specified commensurate with the risk from unauthorized access or use.
- 4.3.1.4 Software requirements shall identify the operating system, function, interfaces, performance requirements, installation considerations, *design inputs* (see def.), and any design constraints of the computer program.

4.3.2 Software Design

- 4.3.2.1 The software design shall be documented and shall define the computational sequence necessary to meet the software requirements.
- 4.3.2.2 The software design shall consider the computer program's *operating environment* (see def.).

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 10 of 23
-----------------------------------	--

- 4.3.2.3 Measures to mitigate the consequences of the problems, as identified through analysis, shall be an integral part of the design. These potential problems include external and internal abnormal conditions and events that can affect the computer program.
- 4.3.2.4 Software design documentation shall include (as applicable):
- A. Numerical methods
 - B. Mathematical models
 - C. Physical models
 - D. Control flow
 - E. Control logic
 - F. Data flow
 - G. Process flow
 - H. Data structures
 - I. Process structures
 - J. Applicable relationships between data structures and process structures.

NOTE: *This documentation may be combined with the documentation of the software design requirements or the computer program listings resulting from implementation of the software design.*

4.3.3 Software Design Verification

- 4.3.3.1 *Software design verification* (see def.) shall evaluate the technical adequacy of the design approach and ensure internal completeness, consistency, clarity, and correctness of the software design and shall verify that software design is traceable to the software design requirements.
- 4.3.3.2 Software design verification shall include review of test results.
- 4.3.3.3 The requirements for the software design verification activity shall be documented in the software engineering method.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 11 of 23
-----------------------------------	---

4.3.3.4 Software design verification shall be performed by competent individual(s) or group(s) other than those who developed and documented the original design, but who may be from the same organization.

4.3.3.5 Verification may be performed by the originator's supervisor, provided:

A. The supervisor did not specify a singular design approach or rule out certain design considerations and did not establish the design inputs used in the design

OR

B. The supervisor is the only individual in the organization competent to perform the verification.

NOTE: *Cursory supervisory reviews do not satisfy the intent of this requirement.*

4.3.3.6 The results of the verification shall be documented with the identification of the verifier indicated.

4.3.3.7 Software verification methods shall include any one or a combination of *design reviews* (see def.), alternate calculations, and tests performed during computer program development.

4.3.3.8 The extent of verification and the methods chosen shall be a function of the complexity of the software, the degree of standardization, the similarity with the previously proved software, and the importance to safety.

4.3.3.9 Simple and easily understood computer programs (for example, computer programs whose results can be easily confirmed through hand calculations) that are used in the design of plant SSCs, may be excluded from the controls of this PRD if designs using these computer programs are individually verified. Design verification documentation should include design inputs, the computer-program-generated results, and computer-generated evidence of the programmed algorithms or equations (for example, computer program listings, spreadsheet cell contents). However, frequent use of the software may justify the application of this PRD in order to simplify future use of the software.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 12 of 23
-----------------------------------	---

4.3.3.10 The software design verification shall be completed prior to approval of the computer program for use.

4.4 Computer Program Testing

4.4.1 Test Requirements

4.4.1.1 Computer program tests, including, as appropriate, software design verification, factory acceptance tests, site acceptance tests, and in-use tests shall be controlled. (from NQA-1-2008 with Addenda through NQA-1a-2009, Requirement 11, Section 200)

4.4.1.2 Test requirements and acceptance criteria for computer programs shall be provided by the organization responsible for the use of the computer program and shall include the following, as applicable.

4.4.1.2.1 Software design verification testing shall demonstrate the capability of the computer program(s) to provide valid results for test problems encompassing the range of documented permitted usage.

4.4.1.2.2 Computer program acceptance testing shall consist of the process of exercising or evaluating a system or system component by manual or automatic means to ensure that it satisfies the specified requirements and to identify differences between expected and actual results in the operating environment.

4.4.1.2.3 In-use computer programs testing shall demonstrate required performance over the range of operation of the controlled function or process.

4.4.2 Test Procedures

4.4.2.1 Computer program test procedures shall provide for demonstrating the adherence of the computer program to documented requirements.

4.4.2.2 For those computer programs used in design activities, test procedures shall provide for assuring that the computer program produces correct results.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 13 of 23
-----------------------------------	---

- 4.4.2.3 For those computer programs used for operational control, computer program test procedures shall provide for demonstrating required performance over the range of operation of the controlled function or process.
- 4.4.2.4 The procedures shall also provide for evaluating technical adequacy through comparison of test results from alternative methods, such as hand calculations, calculations using comparable proven programs, or empirical data and information from technical literature.

4.4.3 In-Use Test Procedures

- 4.4.3.1 In-use test procedures shall be developed and documented to permit confirmation of acceptable performance of the computer program in the operating system.
- 4.4.3.2 In-use test procedures shall be performed after the computer program is installed on a different computer, or when there are significant changes in the operating system.
- 4.4.3.3 Periodic in-use manual or automatic self-check in-use tests shall be prescribed and performed for those computer programs in which computer program errors, data errors, computer hardware failures, or instrument drift can affect required performance.
- 4.4.3.4 Test procedures or plans shall specify the following, as applicable:
- A. Required tests and test sequence
 - B. Required ranges of input parameters
 - C. Identification of the stages at which testing is required
 - D. Criteria for establishing test cases
 - E. Requirements for testing logic branches
 - F. Requirements for hardware integration
 - G. Anticipated output values
 - H. Acceptance criteria
 - I. Reports, records, standard formatting, and conventions.

SOFTWARE QUALITY ASSURANCE

Identifier: PRD-5092

Revision*: 19

Page: 14 of 23

4.4.4 Test Results

- 4.4.4.1 Test results shall be documented and maintained. Test results shall be evaluated by the responsible authority to ensure that test requirements have been satisfied.

4.4.5 Test Records

- 4.4.5.1 Test records shall be established and maintained to indicate the ability of the computer program to satisfactorily perform its intended function or to meet its documented requirements.

- 4.4.5.1.1 Computer program test records will include:

- A. Computer program tested, including *system software* (see def.) used
- B. Computer hardware used
- C. Test equipment and calibrations, where applicable
- D. Date of test
- E. Tester or data recorder
- F. Simulation model used, where applicable
- G. Test problems
- H. Results and applicability
- I. Action taken in connection with any deviations noted
- J. Person evaluating test results
- K. Acceptability.

4.5 Implementation

- 4.5.1 The software design shall be translated into computer program(s) using the programming organization's or design organization's programming standards and conventions.
- 4.5.2 The implementation process shall result in software products such as computer program listings and instructions for computer program use.
- 4.5.3 A review shall be performed in accordance with Subsection 4.6 of this PRD.

SOFTWARE QUALITY ASSURANCE

Identifier: PRD-5092

Revision*: 19

Page: 15 of 23

4.6 Review

- 4.6.1 The appropriate software engineering elements as described in Subsection 4.1.1 shall define the *control points* (see def.) and associated review.
- 4.6.2 Reviews of software shall assure compliance with the approved software design requirements.
- 4.6.3 Reviews may be performed and documented separately or combined, as appropriate, to the defined software engineering method.
- 4.6.4 The following two reviews are required:
 - 4.6.4.1 One review shall consider the requirements related to the activities of preparing the computer program for acceptance testing. This review can be combined with or be part of the software design verification.
 - 4.6.4.2 One review shall provide assurance of the satisfactory completion of the *software development cycle* (see def.), including acceptance testing. This review can be combined with or be part of software design verification. Individual(s) familiar with the design detail and the intended use of the computer program shall be included in the review.
- 4.6.5 Reviews shall identify the participants and their specific review responsibilities.
- 4.6.6 Documentation of review comments and their disposition shall be retained until they are incorporated into the updated software.
- 4.6.7 Comments not incorporated and their disposition shall be retained until the software is approved for use.
- 4.6.8 When review alone is not adequate to determine if requirements are met, alternate calculations shall be used, or tests shall be developed and integrated into the appropriate activities of the software development cycle.
- 4.6.9 Tests performed in support of review can be used to complement acceptance testing.
 - 4.6.9.1 The tests and test results shall be included in the acceptance testing documentation.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 16 of 23
-----------------------------------	---

4.6.9.2 Such tests shall be subjected to the same criteria as the acceptance tests.

4.6.9.3 These tests do not substitute for performing the comprehensive, end of development, acceptance test.

4.7 Acceptance Testing

- 4.7.1 The acceptance testing activity shall demonstrate that the computer program adequately and correctly performs all intended functions (that is, specified software design requirements).
- 4.7.2 Acceptance testing shall demonstrate, as appropriate, that the computer program:
- A. Properly handles abnormal conditions and events as well as credible failures
 - B. Does not perform adverse unintended functions
 - C. Does not degrade the system, either by itself or in combination, with other functions or configuration items.
- 4.7.3 Acceptance testing shall be performed prior to approval of the computer program for use.
- 4.7.4 Configuration items shall be under configuration change control prior to starting acceptance testing.
- 4.7.5 Acceptance testing shall be planned and performed for all software design requirements. Acceptance testing ranges from a single test of all software design requirements to a series of tests performed during computer program development. Performance of a series of tests provides assurance of correct translation between activities and proper function of individual modules.
- 4.7.6 Testing shall include a comprehensive acceptance test performed in the operating environment prior to use.
- 4.7.7 The test plans, test cases, and test results shall be documented, reviewed, and approved prior to use of the computer program in accordance with PRD-5082, "Test Control."
- 4.7.8 Observations of unexpected or unintended results shall be documented and dispositioned prior to test result approval.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 17 of 23
-----------------------------------	---

- 4.7.9 The acceptance testing of changes to the computer program shall be subjected to selective retesting to detect unintended adverse effects introduced during the change. Such testing shall provide assurance that the changes have not caused unintended adverse effects in the computer program, and to verify that the modified system(s) or system component(s) still meets specified software design requirements.

4.8 Operation

- 4.8.1 After the software is approved for use and installed in the operating environment, the use of the software shall be controlled in accordance with approved procedures and instructions, including the following, as applicable:
- A. Application documentation (for example, application log)
 - B. Access control specifications
 - C. Problem reporting and corrective action
 - D. In-use tests
 - E. Configuration change control process
 - F. Computer system vulnerability protection.

4.9 Maintenance

- 4.9.1 The appropriate software engineering elements, as described in Subsection 4.1.1 of this PRD, shall identify how changes to the software are controlled. Typically, changes are in response to:
- A. Enhancement requests from the user community
 - B. Revisions to software based on software design requirements
 - C. Changes to the operating environment and changes to the computer system vulnerability protections
 - D. Reported software problems that must be corrected.

4.10 Software Configuration Management

NOTE: *Software configuration management includes, but is not limited to, configuration identification, change control, and status control.*

- 4.10.1 The appropriate software engineering elements as described in Subsection 4.1.1 of this PRD shall identify when configuration baselines are to be established.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 18 of 23
-----------------------------------	---

- 4.10.2 Configuration items (software) to be controlled shall include, as appropriate:
- A. Documentation (for example, software design requirements, instructions for computer program use, test plans, and results)
 - B. Computer program(s) (for example, source, object, back-up files)
 - C. Support software.
- 4.10.3 Configuration items shall be maintained under configuration management until the software is retired.

4.10.4 Configuration Identification

- 4.10.4.1 A software baseline shall be established at the completion of each activity of the software design process.
- 4.10.4.2 Approved changes created subsequent to a baseline shall be added to the baseline.
- 4.10.4.3 A baseline shall define the most recently approved software configuration.
- 4.10.4.4 A labeling system for configuration items shall be implemented that:
- A. Uniquely identifies each configuration item
 - B. Identifies changes to configuration items by revision
 - C. Provides the ability to uniquely identify each configuration of the revised software available for use.

4.10.5 Configuration Change Control

- 4.10.5.1 Changes to software shall be formally documented.
- 4.10.5.2 The software configuration change control process shall include:
- A. Initiation, evaluation, and disposition of a change request
 - B. Control and approval of changes prior to implementation
 - C. Requirements for retesting (such as, *regression testing* [see def.]) and acceptance of the test results.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 19 of 23
-----------------------------------	---

- 4.10.5.3 The software change documentation shall include:
 - A. A description of the change
 - B. The rationale for the change
 - C. The identification of the affected software baselines.
- 4.10.5.4 The changes shall be formally evaluated and approved by the organization responsible for the original design, unless an alternate organization has been given the authority to approve the changes.
- 4.10.5.5 Only authorized changes shall be made to software baselines.
- 4.10.5.6 Appropriate verification activities shall be performed for the change.
- 4.10.5.7 The change shall be appropriately reflected in documentation and traceability of the change to the software design requirement shall be maintained.
- 4.10.5.8 Appropriate acceptance testing shall be performed for the change.

4.10.6 Configuration Status Control

- 4.10.6.1 The status of configuration items resulting from software design shall be maintained current.
- 4.10.6.2 Configuration item changes shall be controlled until they are incorporated into the approved project baseline.
- 4.10.6.3 The controls shall include a process for maintaining the status of changes that are proposed and approved, but not implemented. The controls shall also provide for notification of this information to impacted organizations.

4.11 Retirement

- 4.11.1 During retirement, support for the software product is terminated, and the routine use of the software shall be prevented.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 20 of 23
-----------------------------------	--

4.12 Standards, Conventions, and Other Work Practices

- 4.12.1 As appropriate, the software engineering method, software acquisition method, or both shall establish the need for standards, conventions, and other required work practices to facilitate software life-cycle activities (for example, software design and implementation activities). Standards, conventions, and other required work practices shall be documented.

4.13 Support Software

- 4.13.1 The software engineering method, software acquisition method, or both shall establish the need for *software tools* (see def.) as appropriate.
- 4.13.1.1 Software tools shall be evaluated, reviewed, tested, and accepted for use, and placed under configuration control as part of the software development cycle of a new or revised software product.
- 4.13.1.2 Software tools that do not affect the performance of the software need not be placed under configuration control.
- 4.13.1.3 In cases involving modifications of software products using the software tools, the configuration of the support software associated with that modification shall be managed.
- 4.13.1.4 Changes to the software tool shall be evaluated for impact on the software product to determine the level of reviews and retesting that will be required.
- 4.13.2 System software shall be evaluated, reviewed, tested, and accepted for use as part of the software development cycle of a new or revised safety software product.
- 4.13.2.1 System software shall be placed under configuration control.
- 4.13.2.2 Changes to the system software shall be evaluated for impact on the software product to determine the level of reviews and retesting that will be required.

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 21 of 23
-----------------------------------	---

4.14 Problem Reporting and Corrective Action

- 4.14.1 Method(s) for documenting, evaluating, and correcting software problems shall:
- A. Describe the evaluation process for determining whether a reported problem is an error or other type of problem (for example, user mistake).
 - B. Define the responsibilities for disposition of the problem reports, including notification to the originator of the results of the evaluation.
- 4.14.2 When the problem is determined to be an error, the method shall provide, as appropriate, for:
- A. How the error relates to appropriate software engineering elements
 - B. How the error impacts past and present use of the computer program
 - C. How the corrective action impacts previous development activities
 - D. How the users are notified of the identified error, its impact, and how to avoid the error, pending implementation of corrective actions.

5. RECORDS

Implementing procedures will define the baseline documents that are to be maintained as records in accordance with PRD-5088, "Quality Assurance Records." These documents can be provided as separate or as combined documents.

6. DEFINITIONS

Refer to LST-199, "Quality Assurance Program Requirements Document Definitions," for the definitions of the following terms:

acceptance testing

baseline

commercial off-the-shelf

computer program

configuration item

configuration management

control point

SOFTWARE QUALITY ASSURANCE

Identifier: PRD-5092

Revision*: 19

Page: 22 of 23

*corrective action**design input**design process**design review**error**guidance**issues**operating environment**procedure**process**procurement document**purchaser**quality engineer**regression testing**safety software**software**software design verification**software development cycle**software documentation**software engineering**Software Inventory Portfolio**software life-cycle**software records**software services**software tool**system software**testing**test case**test plan**verification**verify*

SOFTWARE QUALITY ASSURANCE	Identifier: PRD-5092 Revision*: 19 Page: 23 of 23
-----------------------------------	---

7. REFERENCES

ASME NQA-1-2008 with Addenda through NQA-1a-2009, Subpart 2.7, “Quality Assurance Requirements for Computer Software for Nuclear Facility Application,” American Society of Mechanical Engineers

DOE O 414.1D, “Quality Assurance”

DOE/RW-0333P, *Quality Assurance Requirements and Description*, Revisions 10 and 20, Office of Civilian Radioactive Waste Management

FWD-7, “Foreword”

LST-199, “Quality Assurance Program Requirements Document Definitions”

PRD-5069, “Dedication of Purchased Commercial Grade Items and Services”

PRD-5071, “Quality Assurance Program”

PRD-5075, “Procurement Document Control”

PRD-5078, “Control of Purchased Items and Services”

PRD-5082, “Test Control”

PRD-5088, “Quality Assurance Records”