# Software Configuration Management Plan – Basic Phase

**November 8, 2010**

ASCEM-PM-QA/CMP-110810-Rev. 0

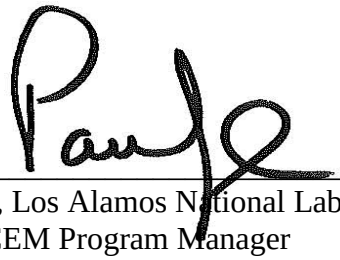
ASCEM

**United States Department of Energy**

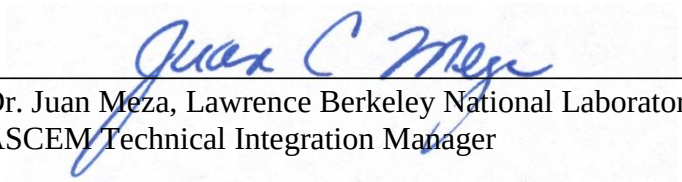
EM Environmental Management

safety ❖ performance ❖ cleanup ❖ closure

## Approval:

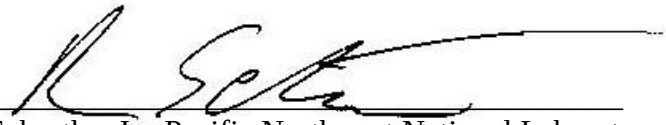


Dr. Paul Dixon, Los Alamos National Laboratory — 11/09/2010 Date
Multi-Lab ASCEM Program Manager



Dr. Juan Meza, Lawrence Berkeley National Laboratory — 11/09/2010 Date
ASCEM Technical Integration Manager

## Concurrence:



Ronald Schrotke, Jr., Pacific Northwest National Laboratory — 11/09/2010 Date
ASCEM Project Quality Engineer

# Table of Contents



## Figures

# 1. INTRODUCTION

This document describes the general software configuration management (SCM) processes that will enable the Advanced Simulation Capability for Environmental Management (ASCEM) team to maintain control of all software work products that are produced during the ASCEM Basic Phase. The general software configuration management concepts that will be adopted by ASCEM include:

- Configuration Identification
- Configuration Change Control and Defect Tracking
- Configuration Status Accounting
- Configuration Audits/Reviews.

## *1.1. Scope*

The scope of this document is to deploy only an ASCEM software configuration management infrastructure to meet the needs of a defined set of users and to manage a clearly defined computing environment. This will include software (newly developed software, acquired software, COTS and/or GOTS), documentation and records management.

As with other ASCEM software quality tasks, a risk-based graded approach (RBGA) is utilized. What this means in practice is that the rigor required for the software configuration management processes will be a function of whether the ASCEM software work products are being developed for the Basic, Applied or Implementation Phases. This Configuration Management Plan only contains the configuration management requirements for the Basic Phase.

This Plan uses a tailored IEEE Std 828-1995[1] as a guide to ensure the salient configuration management topics are addressed.

[1] IEEE Std 828-1995, IEEE Standard for Software Configuration Management plans

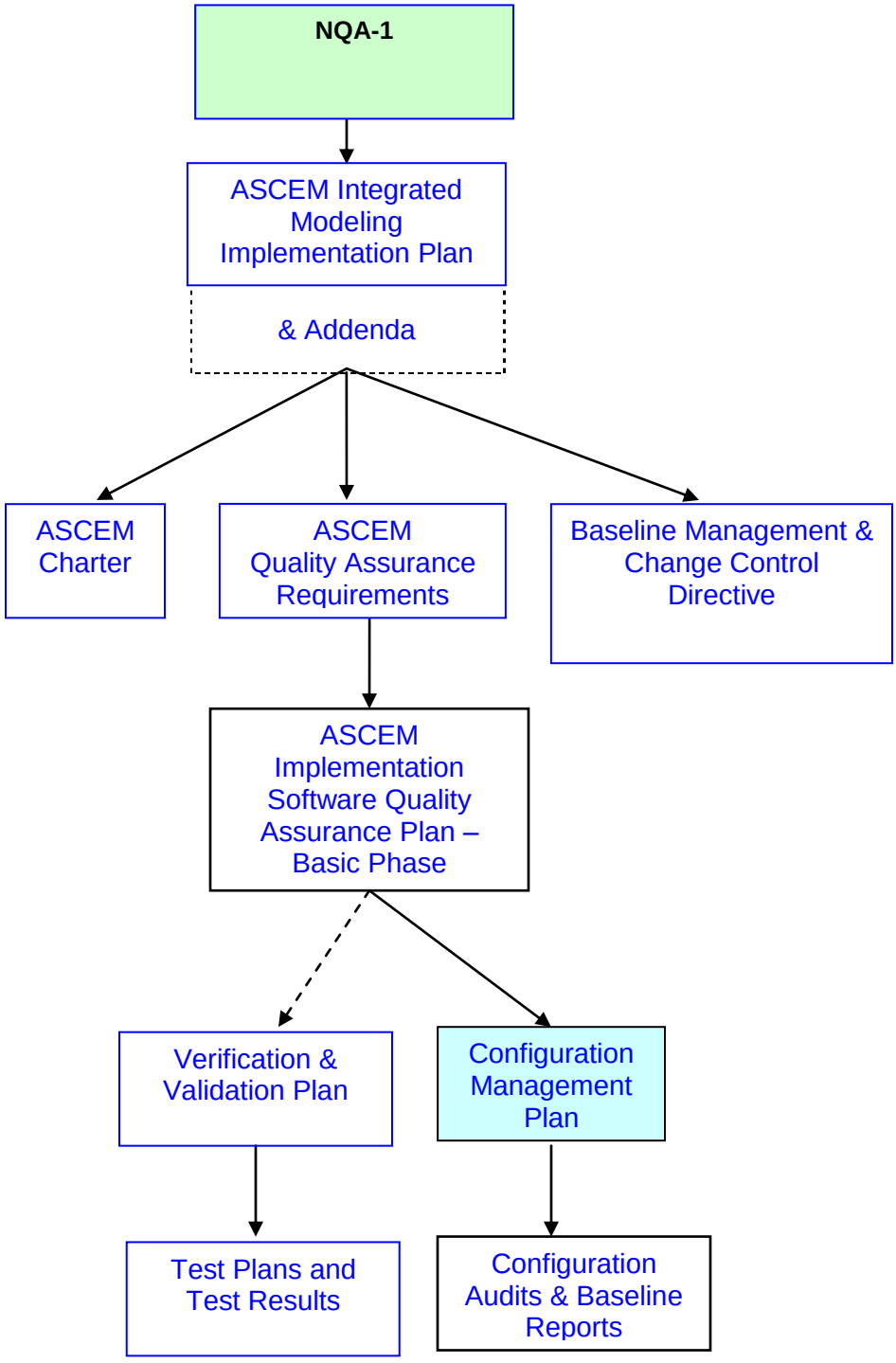



**Figure 1.** Relationship of configuration management plan to other ASCEM documents.

The authority and requirements for this ASCEM Software configuration Management Plan are derived from the superordinate documents as noted in the document hierarchy in *Figure 1* below. Consequently, if there are any disagreements within or among the quality assurance (QA) artifacts supporting ASCEM, the higher tier document has precedence.

## *1.2. Terms*

1. **Baseline** – A specification or product that has been formally reviewed and agreed upon that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures [1].
2. **Change Control Board –** Board established to assess and provide expert support to the Change Control Board Chairperson in the review and disposition of those Baseline Change Requests specified for Change Control Board action [2].
3. **Configuration Control** – An element of configuration management consisting of the evaluation, coordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification [1].
4. **Configuration Identificatio**n – An element of configuration management, consisting of selecting the configuration items for a system and recording their functional and physical characteristics in technical documentation [1].
5. **Configuration Management -** A discipline applying technical and administrative direction and surveillance to a) identify and document the functional and physical characteristics of a configuration item, b) control changes to those characteristics, c) record and report change processing and implementation status, and d) verify compliance with specified requirements [1].
6. **Functional Configuration Audit (FCA**) – An audit conducted to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the performance and functional characteristics specified in the functional or allocated configuration identification, and that its operational and support documents are complete and satisfactory [1].
7. **Physical Configuration Audit (PCA)** – An audit conducted to verify that a configuration item, as built, conforms to the technical documentation that defines it [1].
8. **Record** -Records include all books, papers, maps, photographs, machine readable materials, or other documentary materials, regardless of physical form or characteristics, made or received by an agency of the United States Government under Federal law or in connection with the transaction of public business and preserved or appropriate for preservation by that agency or its legitimate successor as evidence of the organization, functions, policies, decisions, procedures, operations or other activities of the Government or because of the informational value of the data in them [3].
9. **Requirements Traceability Matrix (RTM)** – A matrix that records the relationships between products of the development process (e.g., a matrix that records the relationship

between the requirements and the design of a given software component, between a design component and the source code for a module, and between the source code and test cases) [1].

10. **Risk-Based Graded Approach -** A Department of Energy mandated approach that means the process of ensuring that the level of analysis, documentation, and actions used to comply with a requirement are commensurate with:
    - The relative importance to safety, safeguards, and security (Level of Risk)
    - The magnitude of any hazard involved
    - The relative importance of radiological and non-radiological hazards
    - Any other relevant factors.
11. **Status accounting** – An element of configuration management consisting of the recording and reporting of information needed to manage a configuration effectively. This information includes a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes [1].
12. **Variant** - The variant descriptor is a designator to the right of the decimal point, i.e., ***version.variant*** and indicates minor textual or formatting edits.
13. **Version** – Functional or "major" changes to a software product or documents that necessitate an increment of the version identifier.
14. **Work Product -** Any artifact produced by a process, including files, documents, parts of the product, services, processes, specifications, and invoices.

# 2. ORGANIZATION AND MANAGEMENT

*Figure 2* below shows the ASCEM organizational structure. The quality and configuration management responsibilities are allocated across responsibilities.

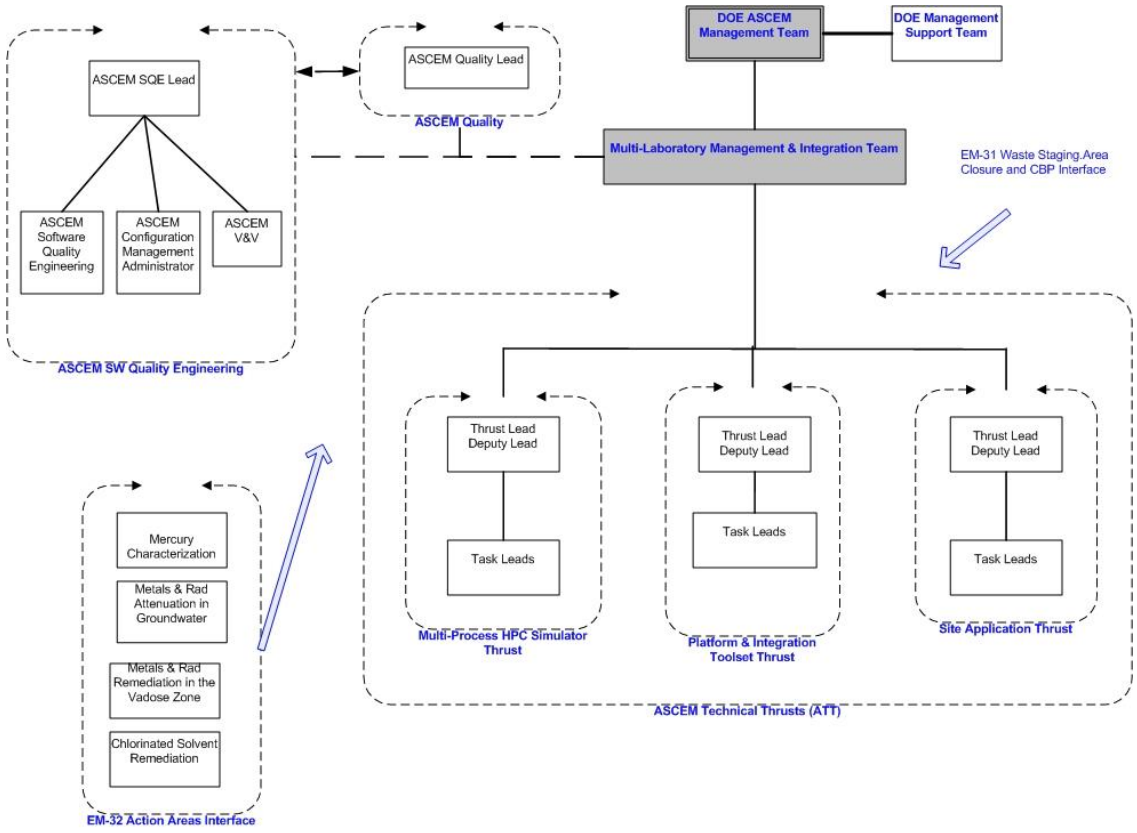



**Figure 2.** ASCEM organization.

## *2.1. Responsibilities*

### 2.1.1. ASCEM Change Control Board

***ASCEM Baseline Management, Risk and Change Control Directive, July 1, 2010,*** addresses the change control process and will not be replicated herein.

### 2.1.2. ASCEM Software Configuration Management Administrator

The ASCEM Software Configuration Management Administrator is responsible for a) configuration identification, b) configuration management status accounting to the ASCEM Quality Lead, and c) configuration management audits.

### 2.1.3. ASCEM Program Manager

The ASCEM Program Manager assures the integrity of all ASCEM work products via appropriate configuration management, and sees that the requisite configuration audits are appropriately conducted, documented, and reported.

### 2.1.4. ASCEM Quality Lead

The ASCEM Quality Lead has the oversight responsibility to ensure that the ASCEM software configuration management objectives are met, and that the ASCEM software configuration management complies with NQA-1-2004 (2005, 2007 Addenda) Nuclear Quality Assurance criteria [4].

### 2.1.5. ASCEM Software Quality Engineering Lead

The ASCEM Software Quality Engineering Lead reviews the ASCEM software configuration management processes and this configuration management plan to assure compliance with ASCEM's quality objectives, as well as compliance with NQA-1 (2004, 2005, 2007 Addenda)

### 2.1.6. ASCEM Thrust Leads

The ASCEM Thrust Leads work with the ASCEM Software Configuration Management Administrator to ensure that all work products under their purview have appropriate software configuration identifiers assigned. Other Thrust Lead configuration management responsibilities include notifying ASCEM Program Management when there are new configuration items or version changes of configuration items (i.e., *changes in functionality)*, and seeing that all of the work products within their purview are being managed under version control. If there are any necessary corrective actions for either the ASCEM software application or ASCEM documents resulting from functionality changes, the Thrust Leads must ensure that corrective actions are managed to a consensus resolution.

## *2.2. Applicable Policies, Directives and Procedures*

The ASCEM Quality Assurance Plan and its implementation software quality plans have precedence over this ASCEM Software Configuration Management Plan.

# 3. SCM ACTIVITIES

## *3.1. A Definition for RBGA for ASCEM Configuration Management*

Since an RBGA is appropriate for managing all ASCEM software configuration management activities, the following configuration management activities are defined for the defined NQA-1 (2004, 2005, 2007 Addenda) Basic Phase:

- Configuration Identification (*Section 3.2)*
- Configuration Control (*Section 3.3)*
- Status Accounting (*Section 3.4)*
- Functional Configuration Audit (*Section 3.5)*

The Applied and Implementation Phases will require additional rigor and will consequently have additional configuration processes and activities.

## *3.2. Configuration Identification*

The configuration identification schemas enable the identification and tracking of all ASCEM work products. The following section will describe the details how these configuration items are identified and tracked.

### 3.2.1. Document Configuration Identification Schema

Every document created during the development of this project is considered a configuration identification item and will have a unique control number associated with it. The document control numbers are issued by the Configuration Management Quality Assurance Custodian.

The document numbering will follow the pattern:

**Project Name – D.DocID-mmddyy-x.xx**

**Project Name – D.DocID – Issuance Date – Version.Variant Number.**

Where the "D" indicates the work product is a "document," the "DocID" indicates the type of document, the mmddyy indicates the month/day/year and the x.xx indicates the version and variant, beginning at 1.00 for the particular document. The version number is incremented every time major or minor changes are approved.

For example, a document schema may look something like this: *ASCEM-D.CMP-070110-1.00.*

The "D.CMP" Identifier" is a unique label identifier to indicate a document with the abbreviation "CMP" (for Configuration Management Plan).

### 3.2.2. Software and Data Configuration Identification Schema

As with documents, all software, data, and database configuration items that are created must have unique control numbers associated. The granularity of these configuration identifiers will be different from documentation since the ASCEM Software Configuration Management Administrator, with consensus from the pertinent Thrust and Task Leads, must define at what modularity level of software to assign a configuration identifier.

The software-identifying schema will follow the pattern:

**Project Name – S.Software Identifier – Issuance Date – Version.Variant Number.**

A software configuration identifier might look like the following: *ASCEM-S.MPHPCS-083011-1.20*.

The "*S.MPHPCS*" designation indicates a software item associated with "MPHPCS," the Multi-Process High Performance Computing Simulator," while the "083011-1.20" indicate the release dates and ***version.variant*** designations.

## *3.3. Configuration Control*

It is essential to have an RBGA for configuration control of the ASCEM QA work products. Without proper configuration control, retrievability of the correct ASCEM work product would be very difficult, at best. The following ASCEM QA artifacts must be managed, retained, and controlled as non-permanent records:

- Software requirements specification
- Design
- Test cases and test results
- Verification and validation
- ASCEM source, object and executable code
- Non-conforming and corrective action work products.

When "Configuration Management" is nominally addressed, configuration control is considered the only necessary component. Although configuration control is essential, it is only one discipline within configuration management. The other three configuration management disciplines are:

1. Configuration identification
2. Status accounting
3. Configuration audits.

How ASCEM will address the other configuration management processes are discussed below.

Section 2 of the *ASCEM Baseline Management, Risk and Change Control Directive* discusses the "Baseline Change Control Process." "Configuration Control" in this context means the management and prioritization of any requests for ASCEM functionality changes, whether additions, deletions, or modifications. Any changes to ASCEM functionality will obviously impact the ASCEM scope and/or schedule, and so will trigger the ASCEM Baseline Change Control Process.

### 3.3.1. Requesting, Evaluating and Approving Changes

*Figure 3*, originally in the *ASCEM Baseline Management, Risk and Change Control Directive*, indicates the process for ASCEM scope, schedule, and budget changes. For the software configuration management process, only a change in scope (i.e., a change in functionality) is applicable for this process.

Any ASCEM project member can submit a change request using the form provided in Appendix C.

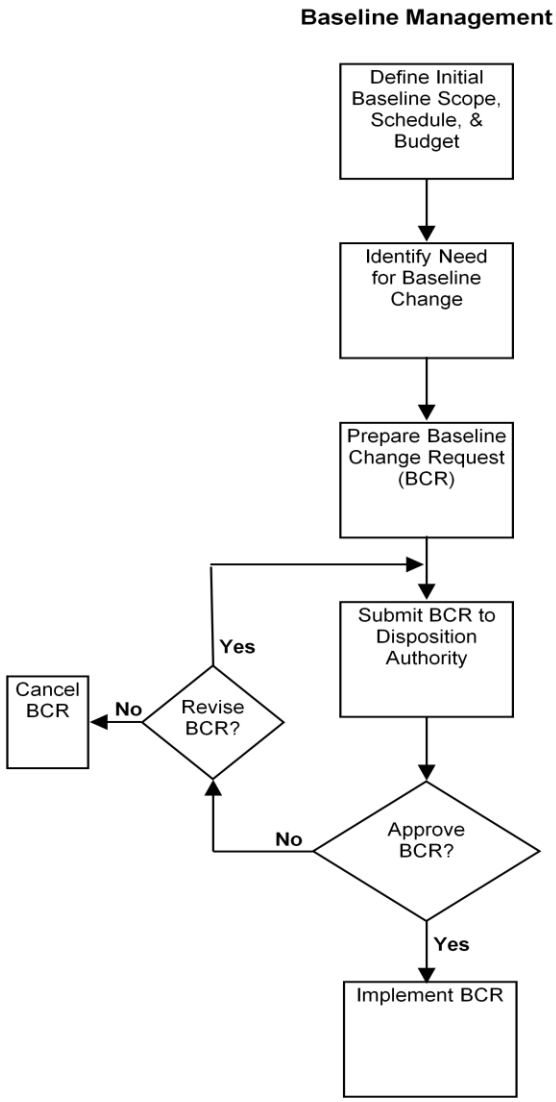



**Figure 3**. Requesting, evaluating, and approving changes.

### 3.3.2. Defect Tracking

ASCEM uses the *Trac* and *Subversion* for version control issues. *Trac* has a "Ticket" system that can be used for noting and tracking defects within ASCEM documents or software.

### 3.3.3. Version Control

ASCEM uses the *MercurialHg,* and *Subversion* software tools to control and manage versioning of documentation and software.

### 3.3.4. Defining Baselines

The ASCEM Project Manager, with the consensus of the Thrust Leads, will define each ASCEM baseline. No changes will be accepted after the established baseline definition without approval from the ASCEM Change Control Board.

### 3.3.5. Managing Baselines

With the consent of the appropriate Thrust Lead, the ASCEM Program and Deputy Program Manager, and the ASCEM Quality Lead, the ASCEM Software Configuration Management Administrator can release a baseline to designated recipients.

Upon release of any ASCEM "baseline," the ASCEM Software configuration Management Administrator is responsible for documenting and tracking that baseline and appropriately updating the configuration identifiers.

### 3.3.6. Records Management

Reference the "**Baseline Management, Risk and Change Control Directive," dated July 1, 2010,** for records management.

## *3.4. Status Accounting*

The ASCEM Software Configuration Management Administrator is responsible for ascertaining the status of all ASCEM software configuration items, whether documents or software. Use of the *Mercurial* configuration item control system and the *Trac* Ticket system will enable a quick determination and reporting of the status of all configuration items.

## *3.5. Software Configuration Audits*

The ASCEM Software Configuration Management Administrator is responsible for conducting configuration audits as noted below. The purpose of the functional configuration and physical configuration audits is to ensure that the ASCEM baseline is complete, correct, known, and reproducible. As with other ASCEM software configuration management activities, a risk-based graded approach is appropriate. Appendix A provides checklists to support the implementation of configuration audits.

### 3.5.1. Baseline Definition

Upon exiting the Basic Phase with intent to deliver the initial ASCEM application to the Applied Phase, the Software Configuration Management Administrator must perform an FCA within 30 days of the baseline release. The FCA results must be documented and provided to the Thrust Leads, the ASCEM Quality Lead, and the ASCEM Program Management.

### 3.5.2. Functional Configuration Audit (FCA)

The purpose of the ASCEM FCA is to ensure that the audited configuration item, both software and its documentation, are consistent with their documented specifications. Appendix A contains a checklist guide to help with the FCA.

The outputs of the software verification and validation activities and the creation of the Requirements Traceability Matrix are key components of a FCA. Appendix B contains a RTM template.

# 4. SCM RESOURCES

ASCEM uses the Open Source *TortoiseHg*, the *Mercurial! TortoiseH*g, and *Trac* to support the SCM objectives that address the requisite defect tracking and version control.

A brief description of the functionality is provided below.

1. Get a *copy* of the document to work on, this is called a *clone* and includes both a working copy as well as a local copy of the entire repository
2. Work on the document by editing, adding, renaming files, etc.
3. Save your changes in your local copy of the repository
4. *hg commit* creates a *changeset* and puts it in your local repository
5. When you *commit*, reference a *Trac* ticket (e.g., refs #2), in your commit message.
6. If you have more changes to make, then continue the changes as noted in Step 2 above.
7. Make sure your local repository is up to date with the *Master* repository.

8. *hg pull* gets *changesets* from the *Master* and puts them in the local repository.
9. Use *hg merge* or *hg update* to bring these into your working copy
10. Share your contributions with the team
11. Make sure your local repository is up to date with the *Master* (see 4).
12. *hg push* your *changesets* to the Master repository.

## 5. Maintenance of SCM Plan

This ASCEM Software Configuration Management Plan will be annually reviewed by the ASCEM Quality Lead to ensure the described software configuration management process is correct and complete. If any changes are needed, the ASCEM Quality Lead will ensure the necessary revisions are made and then submitted.

## 6. REFERENCES

[1] IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology.

[2] ASCEM Baseline Management, Risk and Change Control Directive, July 1, 2010.

[3] 36 CFR 1220, Federal Records.

[4] ASME NQA-1-2004 (including addenda 2005 and 2007), Subpart 2.7 (*Quality Assurance Requirements for Computer Software for Nuclear Facility Applications),* and 4.2 (*Guidance on Graded Application of Quality Assurance for Nuclear-Related Research and Development).*

# APPENDIX A – ACRONYMS AND ABBREVIATIONS

| | |
|---|---|
| ASCEM | Advanced Simulation Capacity for Environmental Management |
| FCA | Functional Configuration Audit |
| NQA | Nuclear Quality Assurance |
| QA | Quality Assurance |
| RBGA | Risk-Based Graded Approach |
| SCM | Software Configuration management |

# APPENDIX B: FUNCTIONAL CONFIGURATION AUDIT CHECKLIST

## *6.1. Functional Software Configuration Audit (FCA)*

**CI Nomenclature:** _____

**Date:** _____

CI/CSCI Identifier: _____

Release # _____

| **Requirements** | **Yes** | **No** | **NA** |
|---|---|---|---|
| 1. Facilities for Conducting FCA Available | | | |
| 2. Audit Team members have been identified and informed of audit | | | |
| 3. Audit Team members are aware of their responsibilities | | | |
| 4. Software Requirements Specification (SRS) | | | |
| 5. Design Documentation (SDD) | | | |
| 6. Acceptance Test Procedures Reviewed and Approved | | | |
| 7. Acceptance Test witnessed | | | |
| 8. Completed Acceptance Test with Results | | | |

**Signature of FCA Team Members:** **Date:**

_____ _____

_____ _____

_____ _____

_____ _____

**Check one:**

☐ Results reviewed satisfy the requirements and are accepted (See attached comments).

☐ Results reviewed do not satisfy requirements (See attached comments and list of deficiencies).

*Approved by:* _____*Date:* _____

# APPENDIX C: REQUIREMENTS TRACEABILITY MATRIX

## Traceability Matrix

### Requirements Traceability Matrix for Software Package

| Requirement Source | Requirement Number | Design Document Reference (*if applicable*) | Test Case | Hand Calculation Reference (*if applicable*) | Acceptance Criteria | Pass/Fail | NOTES |
|---|---|---|---|---|---|---|---|
| | **SR.01** | | | | | | |
| | **SR.02** | | | | | | |
| | **SR.03** | | | | | | |
| | **SR.04** | | | | | | |

# APPENDIX D: CHANGE REQUEST

## Change Request

| INFORMATION TO BE PROVIDED BY CONFIGURATION MANAGEMENT OFFICER | | | | |
|---|---|---|---|---|
| **1. CR #:** | **2. Date Request Logged:** | | **3. CCB Review Date:** | |
| **4. CR Status:** ____ Open ____ Closed | | | **5. Date Status was revised:** | |
| **INFORMATION TO BE PROVIDED BY ORIGINATOR** | | | | |
| **6. Name** | **7. Phone No.:** | | **8. Organization:** | **9. Date Created:** |
| **10. Associated CR # /Trouble Ticket#:** | **11. Category:** ____ System Mod ____ Data Fix ____ Deviation ____ Waiver | **12. Priority** ____ Routine ____ Urgent ____ Mandated ____ EMERGENCY | **13. Severity:** ____ Low ____ Medium ____ High ____ Critical | **14. Change Type:** ____ System Setup ____ COTS ____ Hardware ____ OS ____ Software ____ Document |
| **15. System:** | | | **16. Requested Implementation Date:** | |
| **17. Title:** | | | | |
| **18. Description of Proposed Change:** | | | | |
| **19. Benefit and/or Justification:** (Detailed Cost Savings, # of Transactions Saved, # of FTEs Saved, Cycle Time Reduction, Legislation or Legal Reference for Mandated Changes): | | | | |
| **20. Impact if Not Approved:** | | | | |
| **INFORMATION TO BE PROVIDED BY THE TECHNICAL LEAD/DEVELOPMENT MANAGER** | | | | |
| **21. Technical Lead / Development Manager:** | **22. Date:** | | **23. Phone No.:** | |
| **24. Person Assigned:** | **25. Requirements Analyst:** | | **26. System Version/Release:** | |
| **27. Impact Statement** *(Note any/all system downtime)*: | | | | |
| **28. Level of Effort/Time Estimate:** | **29. Projected Start Date:** | | **30. Projected End Date:** | |

| | | | |
|---|---|---|---|
| **31. Configurable Items (CIs) Affected:** | | | |
| **ENGINEERING REVIEW BOARD (ERB) RECOMMENDATION (Optional)** | | | |
| **32. Disposition:** ___Approved ___Conditionally Approved ___Disapproved ___Withdrawn ____ Deferred | | | |
| **33. ERB or Technical Comments:** | | | |
| **34. Authorizing Signatures** | | | |
| | | | **Date:** |
| | | | **Date:** |
| **CONFIGURATION CONTROL BOARD (CCB) RECOMMENDATION** | | | |
| **35. Disposition:** ___Approved ___Conditionally Approved ___Disapproved ___Deferred ___ Withdrawn | | | |
| **36. Target Release:** | **37. Release/Implementation Date:** | **38. Approved Priority:** ___Routine ___ Urgent ___Mandated ___EMERGENCY | |
| **39. Comments:** | | | |
| **40. Authorizing Signature(s)** | | | |
| | **Date:** | | **Date:** |
| | **Date:** | | **Date:** |