



Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Software Management Control

Revision: 0

Effective Date is 3 days after the date of approval

Prepared By: _____ *Signature on File* _____ *12/16/15*
Robert Hasson **Date**
Ash Fall Project QA Lead

Approved By: _____ *Signature on File* _____ *12/16/15*
Kelly Ebert **Date**
ORP Ash Fall Project Engineer

Concurrence: _____ *Signature on File* _____ *12/17/15*
Ken Armstrong **Date**
EMCBC Assistant Director,
Office of Technical Support and Asset Management

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

1.0 PURPOSE

This procedure describes the requirements and responsibilities for the standard approach for software controls based on its intended use and relationship to safety. This includes requirement and design control, development, software test control, configuration management, and retirement for the Ash Fall Project software supporting the Office of River Protection (ORP) Program.

2.0 SCOPE

The scope of this procedure is to describe the process for controlling and maintaining software as implemented by the EMCBC Ash Fall Project supporting the ORP Program.

3.0 APPLICABILITY

This procedure applies to EMCBC personnel and contractors that participate in the Ash Fall Project activities implementing the Ash Fall Project Quality Assurance Project Plan (QAPP) AFP-QAPP-01.

The software controls described in this procedure apply to activities within the scope of the QAPP applicable to the Ash Fall Project. This procedure is to be used in conjunction with other documents controlling quality impacting work.

The Ash Fall Project utilizes three styles of software. The three styles of software used are:

- Vendor Embedded (E) software – software part of an instrumentation system, and calibrated as part of the system. (Addressed in the procedure AFP-AP-17, *Control of Measuring and Test Equipment*.)
- Acquired (A) software – software that is supplied by an external source, on external media (e.g., disc).
- Developed (D) software – software developed by Ash Fall Project participants.

Per the QAPP, software or firmware that meets any of the following conditions is EXEMPT from this procedure.

- Personal Productivity software, application software dedicated to producing information, such as documents, presentations, project scheduling, worksheets, databases, charts, graphs, digital paintings, electronic music and digital video, that has no impact on safety, security, or business risk. If the software fails, is not available or if the software produces invalid information, these conditions will not impact safety, security, or business risk.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

4.0 REQUIREMENTS and REFERENCES

4.1 Requirements

4.1.1 EM-QA-001, *EM Quality Assurance Program (QAP)*

4.1.2 ASME NQA-1-2008/2009a, *Quality Assurance Requirements for Nuclear Facility Applications*

4.2 References

4.2.1 AFP-QAPP-01, *Quality Assurance Project Plan (QAPP)*

4.2.2 AFP-AP-01, *Personnel Qualification and Training*

4.2.3 AFP-AP-09, *Problem Reporting and Corrective Action*

4.2.4 AFP-AP-11, *Procurement Document Control*

4.2.5 AFP-AP-20, *Quality Assurance Records*

5.0 DEFINITIONS and ACRONYMS

- 5.1 Acceptance Testing – the process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specified requirements and to identify differences between expected and actual results in the operating environment. **Note:** in the DRI PI-SWERL® system, the initial baseline acceptance testing is comparison of visual displays on the GUI to those of calibrated instrument displays. Acceptance Testing falls under Test Control.
- 5.2 Acquired Software (A) – software supplied by an external source, on external media (e.g. disc, USB flash drive, etc).
- 5.3 Adaptive Maintenance – software maintenance performed to make existing software usable in a changed environment.
- 5.4 Attribute – non-time-related issues of software operation such as portability, acceptance criteria, access control, and maintainability.
- 5.5 Automated Computational Applications – this term is defined to encompass commercially available software packages used to automate repetitive hand calculations (e.g., Excel), and provide graphical representations of test data (e.g.,

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Sigma Plot). It is NOT used to describe applications written in computer languages/programming by the staff to perform standard algebraic or geometric calculations (e.g., surface area) that will be used repetitively by staff.

- 5.6 Baseline – a specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved change control process. **Note:** the initial baseline for embedded software in the DRI PI-SWERL® system is comparison of display to calibrated instrument read-outs. Any basic calculated values are verified by comparing displays to hand calculations and/or spread sheet calculations.
- 5.7 Baseline Document (Baseline Item) – a document necessary to the software qualification process that has been entered into configuration control, reviewed, approved, and submitted to records.
- 5.8 Benchmarking – a testing and review process that includes comparing software results to published examples or other proven solutions.
- 5.9 Change Control – activities associated with a change to existing baseline documents or software. This phase is entered to maintain configuration status accounting of a baseline document. **Note:** for the DRI PI-SWERL® embedded software, change control is initiated when a new image is transferred from the development computer to the operational system.
- 5.10 Computer Program – a combination of computer instructions and data definitions that enable computer hardware to perform computational or control functions.
- 5.11 Configuration Control (Software) – the process of identifying and defining the configuration items in a system, controlling the release and change of these items throughout the system’s life cycle, recording and reporting the status of configuration items and change requests.
- 5.12 Configuration Item (Software) – a collection of hardware or software elements treated as a unit for the purpose of configuration control.
- 5.13 Configuration Status Accounting – the control of the status of items resulting from software design. These include requirements documents (approved Test Plan), design documents, test plans, source code.
- 5.14 Control Point – a point in the software life cycle at which specified agreements or control (typically a test or review) are applied to the software configuration items

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

being developed, e.g., an approved baseline or release of a specified document or computer program.

- 5.15 Corrective Maintenance – maintenance performed to correct errors in hardware or software.
- 5.16 Design – the part of the software life cycle at which technical and quality requirements for the software (change) are specified, reviewed, and approved. Design based testing controls are planned. Design is based on the approved Test Plan.
- 5.17 Developed Software (D) – software source code that is developed in house. This software is under complete control of this procedure and subject to all life-cycle stages. **Note:** Spread Sheets used for repetitive use data analysis and reduction fall under D software and are addressed in this procedure.
- 5.18 Document – in this procedure, a software baseline component in hard copy or electronic form that provides objective evidence of compliance with software quality requirements.
- 5.19 Embedded Software (E) – software that is part of an instrumentation system and calibrated as part of the system. This is purchased “material”, and is addressed through the procurement and calibration requirements of the QAPP. The Ash Fall Project **does not develop or make revisions** to this style of software.
- 5.20 Error (Software) – a condition deviating from a requirement or not properly performing a required function.
- 5.21 Firmware – a combination of computer program and data that resides as read-only software on a hardware device.
- 5.22 Functionality – the functions the software is to perform.
- 5.23 Independent Review (Software Design Verification) – technical review performed by qualified personnel who are independent of those who performed the work, but that may be a part of the same organization.
- 5.24 Operating Environment – a collection of software, firmware, and hardware elements that provide for the execution of computer programs.
- 5.25 Maintenance – activities associated with a supported in-use version of an embedded software image.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- 5.26 Regression Testing – selective retesting to detect errors introduced during modification of the computer program or to verify that the computer program still meets specified requirements.
- 5.27 Software – computer programs (including supporting operating systems, compilers, etc.), procedures, rules and associated documentation and data pertaining to the operation of a computer system.
- 5.28 Software Design Verification – the process of determining if the product of the software design activity fulfills the software design requirements.
- 5.29 Software Development Cycle – the activities that begin with the decision to develop a software product and end when the software is delivered. The software development cycle typically includes the following activities: (a) software design requirements, (b) software design, (c) implementation, (d) test, (e) sometimes installation.
- 5.30 Software Engineering – (a) the application of systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software: that is, the application of engineering to software, (b) the study of approaches as in (a).
- 5.31 Software Life Cycle – the period of time that begins when a software product is conceived and ends when the software is no longer available for use. The life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase. These phases may overlap or be performed iteratively, depending on the software development approach used.
- 5.32 Software Tool – a computer program used in the development, testing, analysis, or maintenance of a program or its documentation. Examples include comparators, cross-reference generators, compilers, CASE (Computer Aided Software Engineering) tools, configuration and code management software, decompilers, disassemblers, editors, flowcharters, monitor test case generators, and timing analyzers.
- 5.33 Software (DRI PI-SWERL®) – software developed to display instrument readings and acquire data. The software does not control the test equipment. DRI PI-SWERL® falls under the scope of this procedure as E software.
- 5.34 SQAP – Software Quality Assurance Plan.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- 5.35 System Software – software designed to enable the operation and maintenance of a computer system and its associated computer programs.
- 5.36 Technical Review – technical reviews are documented and traceable, in-depth critical reviews, analyses, and evaluations of software, documentation, or data that requires verification for applicability, correctness, adequacy, and completeness, and performed by qualified personnel.
- 5.37 Test Case (Software) – a set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement.
- 5.38 Test Plan (Software) – a document that describes the approach to be followed for testing a system or component. Typical contents identify the items to be tested, tasks to be performed, and responsibilities for the testing activities.
- 5.39 Testing Software – the process of (a) operating a system (i.e., software and hardware) or system component under specified conditions, (b) observing and recording the results, (c) making an evaluation of some aspect of the system (i.e., software and hardware) or system component in order to verify that it satisfies specified requirements and to identify errors.
- 5.40 Unintended Functions – behavioral change in software modules not directly modified also referred to as collateral impact.
- 5.41 Version ID – the alpha numeric designator used to identify a collection of controlled baseline documents (software code, associated requirement documentation, design documentation, test cases, and user manual). The format includes the associated Test Plan ID (TPID) (if applicable) as the major designator, and a minor designator (of the Format TPID.XXXX). The major designator indicates the controlled element that constitutes the baseline. A complete baseline must have the same major designator (TPID), but may differ in the minor designator (XXXX). This is to allow changes to one element of the baseline that do not require changes in other elements of the baseline.

6.0 RESPONSIBILITIES

6.1 Ash Fall Project Engineer

- 6.1.1 Ensuring this software QA program is established in accordance with the QAPP, implemented, and maintained for the Ash Fall Project.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

6.1.2 Assigning personnel capable of performing the work in the design, development, use, and evaluation of software, based on their role.

6.2 Ash Fall Project Quality Assurance Lead

6.2.1 Ensuring a software QA program is established in accordance with the QAPP, implemented, and maintained in accordance with this procedure.

6.2.2 Reviewing and approving a SQAP.

6.2.3 Ensuring independent reviews have been completed when required.

6.3 Software Developer

6.3.1 Ensuring software development process is documented per this procedure.

6.3.2 Ensuring SQAP and other required documents are developed, reviewed and approved as required.

6.3.3 Ensuring compliance with requirements and reviews as defined in SQAP.

6.4 Independent Reviewer

6.4.1 Reviewing and approving software lifecycle documents, performed by competent individual(s) (as assigned by the Ash Fall Project Engineer) other than those who performed the original design but who may be from the same organization.

6.5 EMCBC Software Custodian

6.5.1 Maintaining the software inventory list.

6.5.2 Controlling the use of software.

7.0 GENERAL INFORMATION

7.1 Users must be trained to the level appropriate, as determined by the software developer, in accordance with AFP-AP-01, *Personnel Qualification and Training*.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.0 PROCEDURE

8.1 General

8.1.1 The Ash Fall Project Engineer ensures that the software QA program is in accordance with the QAPP and properly implemented.

8.1.2 The Ash Fall Project Engineer designates individuals or organizations responsible for implementation and defining the interfaces with other organizations. The Ash Fall Project Engineer also designates individuals or organizations responsible for reviewing SQAPs, procedures, and life cycle activities. The Ash Fall Project Engineer ensures that these individuals receive training, in accordance with AFP-AP-01, *Personnel Qualification and Training* procedure that is commensurable with the scope, complexity, and importance of the task as defined in the QAPP.

8.2 SQAP

8.2.1 The software developer prepares a SQAP based on the nature, complexity, intended use and effect of the software, based on Attachment F which provides an example of the SQAP. These plans identify the following:

- A description of the overall nature and purpose of the software
- The software products to which it applies
- The organizations responsible for performing the work and achieving software quality, including their tasks and responsibilities
- The software engineering methods, software lifecycle phases and requirements for each phase in accordance with this procedure
- A risk and safety analysis
- The required documentation to be maintained
- The standards, conventions, techniques, or methodologies that shall be used to guide the software development, as well as methods to ensure compliance to the same
- The required software reviews
- The methods for error reporting and corrective action
- Software configuration control requirements per this procedure and methodologies used
- The security capabilities to be implemented.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.2.2 These SQAPs may be prepared individually for each software project, or may exist as a generic document to be applied to developed software, procured software, or software used by each organization. If generic plans are used any unique items for software covered by the generic plan must be documented.

8.2.3 SQAPs shall be reviewed and approved by the Quality Assurance Lead.

8.3 Software Engineering Life Cycle

8.3.1 Introduction

8.3.1.1 Software shall be controlled throughout its life cycle. The control methodology specified in the SQAP shall encompass, at a minimum, the activities, documentation, reviews, and approvals required by this procedure.

8.3.1.2 Software shall be developed or acquired in a traceable, planned, and orderly manner. The software life cycle defined in this section provides the basis for planning and implementing a software development or maintenance project. This section identifies the specific software activities, documentation, and reviews associated with each life cycle phase. This provides control points that when reached, shall ensure specified software is documented, reviewed and base lined.

8.3.1.3 The number of phases and relative emphasis placed on each phase of software development or maintenance shall be defined in the SQAP and will be dependent on the nature and complexity of the software. As each life cycle document is approved, it shall be placed under configuration control in accordance with Section 8.4, Software Configuration Control.

8.3.1.4 The TPID of each piece of software shall be assigned in accordance with Section 8.4 of this procedure.

8.3.1.5 Software Engineering General Overview.

8.3.1.5.1 The scope of software engineering activities includes the following elements, as appropriate:

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

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

8.3.1.6 Verification and Validation (V&V).

8.3.1.6.1 The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements.

8.3.1.7 Risk/Safety Analysis Requirements.

8.3.1.7.1 The SQAP will identify the activities to be performed to ensure the assessment of risk and corresponding mitigations. The SQAP will identify when these activities will be performed and where/how they will be documented.

8.3.1.7.2 The risk/safety mitigation strategies shall be implemented throughout the lifecycle and documented as appropriate in the lifecycle documents. The Software Developer and Independent Reviewer are responsible to make sure this is completed.

8.3.1.7.3 The risk/safety analysis for software is documented as appropriate in the lifecycle documents.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.3.2 Requirements Phase (See Attachment A)

- 8.3.2.1 During this phase, the software developer shall specify and document the requirements that the software must satisfy, and the independent reviewer shall review, and approve the requirements. These requirements shall define the functions to be performed by the software and shall provide the detail and information necessary to design the software.
- 8.3.2.2 Software requirements shall be verifiable and traceable throughout all stages of the software development cycle.
- 8.3.2.3 Documentation: Software requirements documentation shall define requirements for, functionality, performance, design inputs, design constraints, installation considerations, operating systems (if applicable), and external interfaces necessary to design the software.
- 8.3.2.4 Acceptance criteria shall be established in the software requirements documentation for each of the identified requirements. Such criteria shall be used for verification/validation planning and performance as defined in each related life cycle phase.

NOTE

Functionality – The functions the software is to perform.

Performance – The time-related issues of software operation such as speed, recovery and response time.

Design – Constraints imposed on implementation phase activities – Elements that will restrict design operations.

Attributes – Non-time related issues of software operation such as portability, acceptance criteria, access control, and maintainability.

External Interfaces – Interactions with people, hardware, and other software.

- 8.3.2.5 Review and Approval: The Software Developer shall ensure that the software requirement documentation is reviewed and approved by the Independent Reviewer at the completion of this phase. This review shall ensure that the requirements are complete, verifiable, consistent, and technically feasible. The review shall also ensure that the requirements will result in a

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

feasible and usable final product. The review shall be shown complete as denoted by their signature on the form.

8.3.3 Design Phase (See Attachment A)

8.3.3.1 During this phase, the software developer shall develop, document, review, and control a software design on a timely basis. The software developer shall prescribe and document in the SQAP the design activities to the level of detail necessary to permit the design process to be carried out and to permit verification that the design meets requirements.

8.3.3.2 Design Elements:

- The design shall specify the interfaces, overall structure (control and data flow) and the reduction of the overall structure into physical solutions (algorithms, equations, control logic, and data structures).
- The design shall consider the computer operating environment. Computer programs are designed as an integral part of an overall system.
- Measures to mitigate the consequences of problems shall be an integral part of software design. These potential problems include external and internal abnormal conditions and events that can affect the computer program.
- Software shall be designed so as not to fail into an unsafe or unsecured state.

8.3.3.3 Documentation: Software design documentation shall contain:

- A description of the major components of the software design as they relate to the software requirements
- A technical description of the software with respect to the theoretical basis, mathematical model, control flow, data flow, control logic, data structure, numerical methods, physical models, process flow, process structures, and applicable relationship between data structure and process standards
- A description of the allowable or prescribed ranges for inputs and outputs
- The design described in a manner that can be translated into code

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- A description of the approach to be taken for intended test activities based on the requirements and design that specify the hardware and software configuration to be used during test execution
- Applicable reference drawings, specifications, codes, standards, regulations, procedures, or instructions that establish software design requirement test, inspection, and acceptance criteria.

8.3.3.4 Review and Approval

8.3.3.4.1 The software developer shall ensure that an Independent Review (IR) is performed, documented, and approved as acknowledged by the independent reviewer's signature in the approval box of the Design document. This review shall evaluate the technical adequacy of the design approach; ensure internal completeness, consistency, clarity, and correctness of the software design; and verify the software design is traceable to the requirements.

8.3.3.4.2 The IR shall be performed by competent individual(s) that is familiar with the design detail and the intended use, other than those who developed and documented the original design. The individual(s) may be from the same organization. The results of the IR shall be documented. The identification of the verifier shall be documented. 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.

8.3.3.4.3 Software design documentation shall be completed prior to finalizing the IR.

8.3.3.4.4 The extent of the IR and the methods chosen are a function of:

- The importance to risk and safety analysis
- The complexity of the software
- The degree of standardization

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- The similarity with previously proven software
- The importance to the business.

8.3.4 Implementation Phase (See Attachment B)

8.3.4.1 During this phase products such as computer program listings, source code, executables and instructions for computer program use shall be completed by the software developer. They shall adhere to specified coding standards, conventions, and design specifications. The implemented software shall be analyzed to identify and correct errors. The source code for development of software shall be placed under configuration control in accordance with Section 8.4, Software Configuration Control, prior to commencement of the Test Phase.

8.3.4.2 Documentation: Implementation documentation shall include a copy of the software, test cases and associated criteria that are traceable to the software requirements, and design documentation.

8.3.4.3 Review and Approval: The Software Developer shall ensure that specified design constraints, standards, and conventions are implemented per the approved SQAP, Requirements and Design documents. In addition, the Software Developer shall ensure that reviews of the test cases are performed and the test cases are approved by the Independent Reviewer at the completion of this phase. The review shall be shown complete as denoted by their signature on the form.

8.3.5 Test Phase (See Attachments D and E)

8.3.5.1 During this phase, testing shall be planned and performed by the software developer for all software design requirements for software. The software shall be validated by executing the test cases. Failure to successfully execute the test cases shall be reviewed to determine if modification of the requirements, the design, the implementation, or the test plans and test cases is required. Testing shall demonstrate the capability of the software to produce valid results for test cases encompassing the range of permitted usage defined by the program documentation. Such activities shall ensure that the software adequately and correctly performs all intended functions.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- 8.3.5.2 Testing shall demonstrate, as appropriate, that the computer program:
- Properly handles abnormal conditions and events as well as credible failures
 - Does not perform adverse unintended or unexpected functions
 - Does not degrade the system either by itself, or in combination with other functions or configuration items
 - Complies with the risk and safety requirements.
- 8.3.5.3 Test Phase activities shall consist of the testing of the software to ensure adherence to requirements, and to ensure that the software produces correct results for the test cases specified. Test procedures shall provide for evaluating the adequacy of the software test case results using one of the following methods:
- Analysis without computer assistance
 - Other validated computer program(s)
 - Experiments and tests
 - Standard problems with known solutions, or
 - Confirmed published data and correlations.
- 8.3.5.4 Documentation: Test Phase documentation shall include test procedures or plans and the results of the execution of test cases. The test results documentation shall demonstrate successful completion of all test cases or the resolution of unsuccessful test cases and provide direct traceability between the test results and specified software requirements.
- 8.3.5.5 Tests performed in support of a review can be used to complement acceptance testing. The tests and test results shall be included in the acceptance testing documentation. Such tests shall be subjected to the same criteria as the acceptance tests. These tests do not substitute for performing the comprehensive, end of development, acceptance test.
- 8.3.5.6 Testing to an approved plan or process and on a different computer with the same operating environment in which the software will be used shall be the primary method to ensure adherence to the requirements and to ensure the software produces correct results for the test cases.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- 8.3.5.7 Test procedures or plans shall specify the following, as applicable:
- Required tests and test sequence
 - Required range of input parameters
 - Criteria for establishing test cases
 - Requirements for testing logic branches
 - Requirements for hardware integration
 - Anticipated output values
 - Acceptance criteria
 - Reports, records, standard formatting, and conventions
 - Identification of operating environment, support software, software tools or system software
 - Requirements or limits for Hardware Operating System(s)
 - Cyber security testing criteria
 - Risk and safety testing criteria.
- 8.3.5.8 Test Results shall include the following, as applicable:
- Computer program tested including system software used
 - Computer software and hardware configuration used
 - Test equipment and calibrations, where applicable
 - Date of test
 - Tester or data recorder
 - Type of observation
 - Simulation models used, where applicable
 - Test problems
 - Results of reviews and tests and applicability
 - Actions taken in connection with any deviations noted
 - Person evaluating test results
 - Organized to allow traceability to requirements and design
 - Test failures shall be reviewed and actions required documented
 - Acceptability.
- 8.3.5.9 Review and Approval: The Software Developer shall ensure that a technical review of the test procedures/plans and test results is performed. The technical review of the test results shall ensure that the test requirements have been satisfied. When software design is required, Independent Review (IR) of the test results is required in accordance with the Design Phase Section. Documentation of

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

review comments and their disposition shall be retained until they are incorporated into the updated software. Comments not incorporated and their disposition shall be retained until the software is approved for use. The review shall be shown complete as denoted by the independent reviewer's signature in the approval box of the document.

8.3.6 Installation and Acceptance Phase (Attachment E)

8.3.6.1 During this phase the software becomes part of a system incorporating applicable software components, hardware, and data and is accepted for use. The following activities shall be completed during the installation and acceptance phase.

- The Software Developer shall determine the acceptance testing to be performed prior to approval of the computer program for use.
- Acceptance testing shall include a comprehensive test in the operating environment for all critical software design requirements.
- Acceptance testing shall be performed prior to approval of the computer program for use.
- The tests performed shall obtain the necessary data with sufficient accuracy for evaluation and acceptance.
- Configuration items shall be under configuration change control prior to starting acceptance testing.
- Software validation shall be performed to ensure that the installed software product satisfies the specified software requirements.
- The Software Developer shall ensure the acceptance test includes the cyber security requirements verification.
- The Software Developer shall ensure the acceptance test includes the risk and safety requirements verification.
- The Software Developer shall ensure User Instructions are complete.

8.3.6.2 Documentation: Installation and Acceptance Phase documentation shall include results of the test case execution for system installation and integration, User Instructions, and documentation of the software acceptance for operational use.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- 8.3.6.3 User Instructions shall include:
- The approved operating systems
 - A description of the user's interaction with the software
 - A description of any required training necessary to use the software
 - Input and output specifications
 - Input and output formats
 - A description of software and hardware limitations
 - A description of user messages initiated as a result of improper input and how the user can respond, including anticipated errors and how users can respond
 - Information for obtaining user and maintenance support
 - Applicable installation procedures and available sample/examples of execution
 - In-use test that shall be performed, documented, and verified to provide confirmation of acceptable performance of software
 - Risk and safety specifications.
- 8.3.6.4 Review and Approval: The Software Developer shall ensure that design verification activities and an independent review of the acceptance testing, installation results, and user instructions is performed and completed prior to computer program use.
- 8.3.6.5 Installation testing shall be performed by an independent reviewer, and shall ensure the integrity of the software and its interfaces (for example, associated system and memory resident software, associated run-time libraries, and the hardware configuration).
- 8.3.6.5.1 If any of these dependent features are changed in any way, installation testing shall be re-performed.
- 8.3.6.5.2 If an error is discovered during testing, the testing personnel shall report the error to the software custodian and the Quality Assurance Lead, who shall determine whether testing shall continue.
- 8.3.6.5.3 If the error is so severe as to require a new software version, the testing shall not continue. In this case, the Quality Assurance Lead shall document that the

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

testing was terminated (e.g., on the Test Plan), and determine whether other documents (e.g., SQAP, Requirements Document, Design Document) require revision. The documentation of the acceptance of the software for operational use shall ensure that configuration baselines, documentation, and reviews have been completed. The review shall be shown complete as denoted by the independent reviewer's signature in the approval box of the document.

8.3.7 Operations and Maintenance Phase

- 8.3.7.1 During this phase, software shall be controlled to remove latent errors (corrective maintenance), to respond to new or revised requirements (enhancements/preventive maintenance), or to adapt the software to changes in the operating environment (adaptive maintenance). Software modifications shall be approved, documented, verified and validated, and controlled in accordance with the related life cycle phases.
- 8.3.7.2 The validation of modifications shall be subject to selective regression testing to detect errors introduced during the modification of software or operating system components to verify that the modifications have not caused unintended adverse effects and to verify that the modified software still meets its specified requirements.
- 8.3.7.3 Test cases shall be developed and documented by the software developer to permit confirmation of acceptable performance of the software in the environment in which the software is used. Test cases shall be run whenever the software is installed on a different computer, or when hardware or operating system configuration changes are made.
- 8.3.7.4 Periodic in-use manual or automatic self-check in-use tests shall be prescribed and performed for those computer programs where computer program errors, data errors, computer hardware failures, or instrument drift can affect required performance.
- 8.3.7.5 Software modifications shall be tested to ensure cyber security requirements are not impacted. Cyber security

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

testing requirements shall be included in the change process.

8.3.7.6 Software modifications shall be tested to ensure the risk and safety requirements are not impacted. Risk and safety testing requirements shall be included in the change process.

8.3.8 Retirement Phase

8.3.8.1 During the retirement phase, the support for a software product shall be terminated and the routine use of the software prevented. Cyber security, risk and safety impacts will be verified. The software inventory list will be used to document the software as retired.

8.4 Software Configuration Control

8.4.1 Introduction

8.4.1.1 The methods to be used to control, uniquely identify, describe, and document the configuration of each version or update of a computer program (for example, source, object, back-up files) and its related documentation (for example, software design requirements, instructions for computer program use, test plans, and results) shall be described in the SQAP. The SQAP shall meet the following criteria for configuration identification, change control and configuration status accounting.

8.4.1.2 Unique Identification Number: A Unique Identification Number shall be assigned to each piece of software being implemented. Unique Identification Number (UID) specifications: Each item of software will be assigned a UID number that will be associated with the item for the life of the software. At the end of the life of the item the UID will not be utilized further. The UID number shall be associated with all categories of software and data records. The numbers shall be assigned in sequence and in chronological order. The recorder version number of the software, as well as the implemented date, will be utilized to track the changes of the software.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.4.2 Configuration Identification

8.4.2.1 A configuration baseline shall be defined at the completion of each major phase of the software life cycle. Approved changes created subsequent to a baseline shall be added to the baseline. A baseline shall define the most recently approved software configuration.

8.4.2.2 A baseline labeling system shall be implemented that:

- Uniquely identifies each configuration item
- Identifies changes to configuration items by revision
- Provides the ability to uniquely identify each configuration of the revised software available for use.

8.4.2.3 Configuration items shall include:

- Documentation as required by procedures and the SQAP (e.g., software design requirements, instructions for computer program use, test plans and results)
- Cyber security items required
- Risk and safety items as required
- Computer programs (e.g., source, object, back-up files)
- Support software.

8.4.3 Configuration Change Control

8.4.3.1 Control of the baseline includes describing check-in/out of the source code. Software backups shall use a proven method to ensure recovery capability.

8.4.3.2 Ensure cyber security requirements are defined and included in the SQAP. Ensure risk and safety requirements are defined and included in the baseline.

8.4.3.3 Users may request changes to software requirements, user, and/or design documentation to correct errors, to add new features and models, or to modify existing features. Proposed changes to software shall be formally documented per Attachment G, including the initiation, evaluation and disposition of the change. The user shall document the software identification, a description of the change requested, the rationale for the change, and if known, the requirements for re-testing and acceptance of the test results, and

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

the identification of other baseline configuration items/documentation that will require modification to address the proposed change. For example, a change in the design documentation may require a change in the requirements, implementation, and user documentation.

8.4.3.4 Upon receipt of the change request, the software developer shall evaluate the change, and proceed as follows:

- If a problem is being reported, proceed to Section 8.7 of this procedure.
- If the information submitted is incomplete (e.g., lacks test case input) or illegible, the software developer shall request clarification or reject the change request and return to the originator for additional action.

8.4.3.5 When the change request is accepted for action, the software developer shall assign it a change request number and ensure the documentation is included in, or traceable to, the software's UID per Attachment G. The software developer shall make the changes requested to the software, ensure the software is re-tested, ensure the test results are acceptable, and ensure the life cycle documentation is revised to reflect the changes. Modification to the life cycle documentation will require the same level of review and approval as the original. Only approved changes shall be made to software baselines. Software verification activities shall be performed for the change as necessary to ensure the change is appropriately reflected in software documentation, and to ensure that document traceability is maintained. Software validation shall be performed as necessary to ensure that the change does not adversely affect the performance of the software. The software custodian will update the disposition of the change request to "Accepted" upon completion of the independent review.

8.4.3.6 Configuration items shall be maintained under configuration management until the software is retired.

8.4.4 Configuration Status Accounting and Control

8.4.4.1 The information that is needed to manage a configuration baseline shall be documented. This information shall identify the approved configuration baseline, the status of proposed changes

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

to the configuration baseline, the status of approved changes, the history of changes and information to support the functions of configuration identification and configuration control. Configuration control is to include configuration status notification of organizations affected by configuration changes. The status of configuration items resulting from software design shall be maintained current. Configuration item changes shall be controlled until they are incorporated into the approved product baseline. The controls shall include a process for maintaining the status of changes that are proposed and approved, but not implemented.

8.4.5 Support Software

8.4.5.1 Support software includes software tools and system software.

8.4.5.2 Software tools that could impact the performance of the configuration item shall be evaluated, reviewed, tested, accepted for use, and placed under configuration control as part of the software development cycle of a new or revised software product. In cases involving modifications of software products using the software tools, the configuration of the support software associated with that modification shall be managed. 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.

8.4.5.3 System software consists of the on-line computer programs used to provide basic or general functionality and facilitate the operation and maintenance of the application computer program. Examples include: lower level software layers, assemblers, interpreters, diagnostics, and utilities. System software shall be evaluated, reviewed, tested, and accepted for use as part of the software development cycle of a new or revised software product. System software shall be placed under configuration change control, as appropriate. Changes to the system software shall be evaluated for impact on the software product.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.5 Evaluation (Existing/Acquired Software)

8.5.1 Software, which was not developed in accordance with this procedure, shall be evaluated and validated in accordance with Section 8.3.5, and placed under configuration control and controlled in accordance with Section 8.4 of this procedure. The Ash Fall Project Engineer shall ensure that the evaluation schedule for affected software is developed. The Software Developer shall perform and document an evaluation of existing software which:

- Determines the adequacy of software documentation to support testing, operation, and maintenance
- Identifies activities to be performed throughout the applicable life cycle of the software including preparation of required documentation and performance of required reviews and/or tests
- Determines the software's capabilities and limitations for intended use
- Specifies test plans and test cases required to validate the capabilities within the stated limitations
- Identifies instructions for software use within the limits of its capabilities
- Identifies any exceptions to the life cycle documentation and its justification
- Identifies adequacy of cyber security requirements for use
- Identifies adequacy of the risk and safety requirements for use.

8.5.2 This evaluation may be documented in a work request, plan, procedure, project level instruction, or other method, as appropriate. Exceptions to the life cycle documentation shall be approved by the QA Lead.

8.5.3 Revisions to previously base lined software received from organizations not required to follow Software QA Requirements, shall be evaluated in accordance with the requirements.

8.5.4 The results of the above documentation and the performance of the actions necessary to accept the software shall be reviewed and approved by an Independent Reviewer. The documentation and associated computer program(s) shall establish the current baseline.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.6 Software Procurement

8.6.1 Procurement

8.6.1.1 Software and software services shall be procured in accordance with procurement procedure, AFP-AP-11, *Procurement Document Control*. Procurement documents shall identify requirements for Supplier's reporting of software errors to the Purchaser and, as appropriate, the Purchaser's reporting of software errors to the Supplier.

8.6.1.2 Cyber security requirements shall be defined in the procurement specifications.

8.6.1.3 Risk and safety requirements shall be defined and included in the procurement specifications.

8.7 Problem Reporting and Corrective Action

8.7.1 The methods to be used for reporting site developed or site maintained code software problems and taking appropriate action shall be done in accordance with AFP-AP-09, *Problem Reporting and Corrective Action*, unless otherwise described in the SQAP.

8.7.2 Conditions adverse to quality shall be identified promptly and corrected as soon as practicable. In the case of a significant condition adverse to quality the cause of the condition shall be determined and corrective action taken to preclude recurrence. The identification, cause, and corrective action for significant conditions adverse to quality shall be documented and reported to appropriate levels of management. Completion of corrective actions shall be verified. The problem reporting and corrective action process shall address the appropriate requirements of the QA program corrective action system and the following elements:

- Method(s) for documenting, evaluating, and correcting software problems, including cyber security risk and safety, shall:
 - Describe the evaluation process for determining whether a reported problem is an error; and
 - Define the responsibilities for disposition of the problem reports, including notification to the originator of the results of the evaluation.

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- When the problem is determined to be an error, the method shall provide, as appropriate, for:
 - How the error relates to appropriate software engineering elements
 - How the error impacts past and present use of the computer program
 - How the corrective action impacts previous development activities
 - How the users are notified of the identified error, its impact; and how to avoid the error, pending implementation of corrective actions.

8.7.3 If a problem is being reported, a description of the problem and a listing of input for a test case that demonstrates the error (or sufficient information to allow a test case to be developed) shall be documented in a formal request per Attachment G. The software developer shall confirm that the computer program is run using input supplied with or described in the request and that the appropriate action below is accomplished:

- If the error cannot be duplicated with the input provided or if it is found that the supplied input is in error, the software developer shall note this information with the request to the user and in the project records. No further action shall be required for the change request.
- If the error occurs as described by the user, or if any other errors occur in the run, the software developer shall complete the Nonconformance Report, sign and date it and forward it and the input and output to the appropriate personnel as defined in the Nonconformance procedure.
 - The Ash Fall Project Engineer (or designee) shall review the change request, nonconformance report, and input and output documentation to assess the impact of the problem, to determine disposition of the request, and to determine if users shall be notified. The Ash Fall Project Engineer (or designee) shall indicate the disposition of the request through documentation traceable to the request and return the information to the software developer.
 - The software developer shall ensure the appropriate information is traceable to the UID and logged on a Software Change Request (Attachment G) and shall take the following actions as appropriate on the disposition as specified by the Ash Fall Project Engineer (or designee):

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- If the change request is approved, the software developer shall confirm changes to the software, requirements documentation, and design documentation are developed in accordance with the appropriate sections of this procedure
- If the requested change is disapproved, the software developer shall send notification to the originator. (Note: Notification may be sent electronically.) A copy is to be maintained as a project record.
- If users are to be notified of an error, the software developer shall send information on the error to the department. (Note: Notification may be sent electronically.)
- The software custodian may notify users of the software and provide user access to interim corrections for errors that have been processed before formal release of a new version of the software. The software developer shall include a transmittal letter or memo stating that users assume all risks and responsibilities for use of software containing such interim corrections and that users shall modify the version identifier (UID) accordingly to reflect the changes installed and shall include the verification disclaimer.

8.8 Software Inventory List (SIL)

8.8.1 The Software Inventory List contains software that will be controlled in accordance with this procedure. The SIL will be maintained by the Software Custodian. The SIL should include the following information (at a minimum):

- Software Description
- Software Name
- Version Identifier
- Safety Software Designation (Safety and Hazard Analysis Software, M&TE, etc.)
- Responsible Individual
- Date Implemented
- Retirement Date (as applicable)

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

8.9 Automated Calculational Applications Software

8.9.1 The types of computer programs covered by this procedure are developed and used to eliminate the need for repetitive hand calculations or graphing activities. These types of computer programs (i.e., automated calculational applications – ACA), shall be treated as hand calculations. For this reason, the need for processes inherent to other computer program (e.g., simulation) activities, such as the development and documentation of requirements specifications and design reviews are not required. The processes required for these automated calculational applications (ACA) are design and development, testing, application, and control of the ACA – as defined in this procedure.

8.9.2 Automated Calculational Applications

8.9.2.1 ACA Design and Development

8.9.2.1.1 The ACA design and development process is the operation of transferring the concept of an ACA into an ACA. This is performed at the computer by the developer, and results in the development of the ACA for operation (i.e., use). It **does not** require the development of support documentation, such as requirement specifications and flow charts. The ACA package shall be prepared for each ACA application that will be used to provide reportable results. The ACA shall include:

- ACA listing,
- a brief description of the purpose of the ACA,
- sufficient instruction for utilizing the ACA,
- operational environment specifics,
- a name, and date or revision for the ACA.

8.9.2.1.2 The ACA package shall be transmitted to project records in accordance with record transmittal requirements (e.g., Records Inventory Disposition Schedule).

8.9.2.2 When cell protection (lock-down) is deemed necessary to assure control of the ACA application (e.g., due to multiple users) the cells will be identified in the ACA description and the protection

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

password shall be determined, applied, and controlled by the developer.

8.9.3 Test of ACA

8.9.3.1 When the ACA has been developed for its intended application, it shall be tested prior to any application. The developer shall perform and document test cases sufficient to accept that the ACA is performing the actions it has been developed to perform (i.e., calculations or graphical presentation of data).

8.9.3.2 Testing can be through the performance of hand calculated over-check(s) of portions of the application, or the development and use of test cases to produce results for review and acceptance, or a combination of both. The mode of testing shall be defined and maintained with the ACA package. The criteria used to determine the acceptability (e.g., results agree within ± 0.01) of the ACA shall also be documented and maintained with the ACA package.

8.9.4 Application of ACA

8.9.4.1 Prior to use of an ACA, the user shall review the ACA package to assure that the ACA has been tested and can be used for the intended application. This review shall be documented (e.g., notation in the documentation).

8.9.4.2 When an ACA is used to produce results for the project, the user shall assure that documentation of the ACA application is prepared. The documentation shall include; the acceptance review notation of the user, the data used, the ACA used, and the results obtained.

8.9.4.3 Review of the ACA results for use on the project shall be performed as part of the document review process of the project milestone (e.g., report).

8.9.5 ACA Control

8.9.5.1 Each user of the ACA is responsible for the control of the ACA. The user shall assure that a copy of the ACA package is maintained with the ACA. The access controls necessary for the ACA shall consider the potential for loss or damage (e.g., unintended change)

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

to the ACA. The access control process shall be documented and maintained as part of the ACA package.

8.9.5.2 Configuration Control of ACA

8.9.5.2.1 The user should periodically review the ACA to assure that no unintended changes have occurred to the accepted ACA. These checks shall be noted in project documentation.

8.9.6 ACA Errors

8.9.6.1 When errors are encountered during the operation of the ACA – the developer shall notify the Project Engineer to determine the impact on activities and the necessary corrective action measures.

9.0 RECORDS

9.1 The approved document in its entirety shall be submitted by the EMCBC Coordinator to records in accordance with AFP-AP-20, *Quality Assurance Records*.

9.2 The following are considered Lifetime QA Records:

- Requirements and Design Life Cycle Document/Phase Criteria
- Implementation Life Cycle Document/Phase Criteria
- Software Test Control Plan and Review Form
- Software Test Document Criteria and Review Form
- Installation and Acceptance Life Cycle Document/Phase Criteria
- The approved Controlled Document SQAP
- Software Change Request

10.0 FORMS USED

Form 6-1 – Requirements and Design Life Cycle Document/Phase Criteria
Form 6-2 – Implementation Life Cycle Document/Phase Criteria
Form 6-3 – Software Test Control Plan and Review Form
Form 6-4 – Software Test Document Criteria and Review Form
Form 6-5 – Installation and Acceptance Life Cycle Document/Phase Criteria
Form 6-6 – Software Quality Assurance Plan (SQAP)
Form 6-7 – Software Change Request

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

11.0 ATTACHMENTS

- Attachment A – Requirements and Design Life Cycle Document/Phase Criteria
- Attachment B – Implementation Life Cycle Document/Phase Criteria
- Attachment C – Software Test Control Plan and Review Form
- Attachment D – Software Test Document Criteria and Review Form
- Attachment E – Installation and Acceptance Life Cycle Document/Phase Criteria
- Attachment F – Software Quality Assurance Plan (SQAP)
- Attachment G – Software Change Request

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment A Requirements and Design Life Cycle Document/Phase Criteria

Ash Fall Project AFP-AP-06, Form 6-1	Requirements and Design Life Cycle Document/Phase Criteria	Form Version: 0 Total Pages:
<p>This document is created for the initial baseline and then for each major development item. It is the cover page to the life cycle document that describes the overall translation of the design from the requirements (Approved Test Plan). It describes how the software requirements are translated into design, test implications, how it should be documented, and what it impacts. This is used to produce and approve life-cycle documents during the Requirements and Design Phase.</p>		
<p>Unique Identification Number (UID):</p>		
<p>Description: <i>(High level description of the major components of the software design)</i> Include:</p> <ul style="list-style-type: none"> • Functionality <i>(the functions the software is to perform)</i> • Performance <i>(the time-related issues of software operation such as speed, recovery and response time)</i> • Design <i>(constraints imposed on implementation phase activities – elements that will restrict design operations)</i> • Attributes <i>(non-time related issues of software operation such as portability, access control, and maintainability)</i> • External Interfaces <i>(interactions with people, hardware, and other software)</i> 		
<p>Technical Description: <i>(Theoretical basis, mathematical model, control flow, data flow, control logic, data structure, numerical methods, physical models, process flow, process structures, and applicable relationship between data structure and process standards)</i></p>		
<p>Design: <i>(The design described in a manner that can be translated into code)</i> Include:</p> <ul style="list-style-type: none"> • A description of the allowable or prescribed ranges for inputs and outputs • Applicable reference drawings, specifications, codes, standards, regulations, procedures, or instructions that establish software design requirement test, inspection and acceptance criteria 		
<p>Implementation Details: <i>(How the component should be implemented. Include test cases and associated criteria that are traceable to the software requirements and design documentation.)</i></p>		
<p>Collateral Impact: <i>(Other components of the software that will need to be changed – not needed for initial baseline)</i></p>		
<p>Unit Testing: <i>(A description of the approach to be taken for intended test activities based on the requirements and design that specify the hardware and software configuration to be used during test execution. Basic tests to be performed at the “development level” before the source code is transferred into production. Include acceptance criteria.)</i></p>		
<p>Testing Impact: <i>(How will this change affect existing working tests? Note design changes that will impact existing regression testing. If this is a change, are any new tests required to effectively test this design change?)</i></p>		
<p>User Documentation: <i>(How/where will the user documentation need to be updated, for example, a new example, screens or additions to existing displays on the production system?)</i></p>		
<p>Related Design: <i>(Reference related design dependencies, for example the System Tool Architecture Specification)</i></p>		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated the design plan; needed configuration control items are identified, the translation of requirements into design is appropriate, and traceability is established.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment B Implementation Life Cycle Document/Phase Criteria

Ash Fall Project AFP-AP-06, Form 6-2	Implementation Life Cycle Document/Phase Criteria	Form Version: 0 Total Pages:
<p>This document is created for the initial baseline and then for each major development item. It is the cover page to the life cycle document that describes the overall implementation of the software design. Implementation documentation includes a copy of the software, test cases and associated criteria that are traceable to the software requirements and design documentation (Attachment A). This is used to produce and approve life-cycle documents during the Implementation Phase.</p>		
<p>Unique Identification Number (UID):</p>		
<p>Implementation Details: <i>(How the component should be implemented)</i></p>		
<p>Copy of Software: <i>(Include a copy of computer program listings, source code, executables and instructions for computer program use)</i></p>		
<p>Test Cases: <i>(Include test cases and associated criteria that are traceable to the software requirements and design documentation)</i></p> <ul style="list-style-type: none"> • Required test and test sequence • Required range of input parameters • Identification of the stages at which testing is required • Criteria for establishing test cases • Requirements for testing logic branches • Requirements for hardware integration • Anticipated output values • Acceptance criteria • Reports, records, standard formatting, and conventions • Identification of operating, support software, software tools or system software • Requirements or limits for Hardware Operating System(s) • Risk and safety criteria 		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated the implemented software; the specified design constraints, standards, and conventions are implemented, and the test cases and associated criteria are traceable to the software requirements and design documentation.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment C Software Test Control Plan and Review Form

Ash Fall Project AFP-AP-06, Form 6-3	Software Test Control Plan and Review Form	Form Version: 0 Total Pages:
<p>A Software Test Control Plan (STCP) form shall be submitted to document and explain the software testing scope and strategy. Successful completion of the STCP shall demonstrate that the software will adequately and correctly perform all its intended functions. Attach additional supporting information as needed. This form is used as a cover page.</p>		
<p>Unique Identification Number (UID):</p>		
<p>1. Summary: <i>(Provide a brief and clear statement that concisely describes the goals of the overall testing strategy. Explain requirements and expectations. Provide an overview of the testing scope, strategy, schedule, and risks.)</i></p>		
<p>2. Test Procedures, Test Scripts and/or Test Cases: <i>(Describe test procedures, test scripts, and/or test cases that are planned for use with the current release. Identify any known deficiency in any existing or missing tests and the actions planned to correct or mitigate the condition.)</i></p>		
<p>3. Test Coverage: <i>(Describe the test coverage planned. Reference or attach mapping and justification of adequacy.)</i></p>		
<p>4. Test Schedule and Execution: <i>(Specify the milestones of the test schedule planned and the strategy of execution for the current release. Reference or attach schedule. Identify any known deficiency in schedule e.g.; resource allocation, special test configurations unavailable, etc. and the actions planned to correct or mitigate the condition.)</i></p>		
<p>5. Test Results: <i>(Describe the process and the requirements for documenting test results that will be followed for the current release.)</i></p>		
<p>6. Problem Reporting and Disposition: <i>(Describe how problems identified during testing of the software code are reported, tracked, and resolved. Note: These problems may need to be documented through the Problem Reporting and Corrective Action, AFP-AP-09.)</i></p>		
<p>7. Acceptance Testing Metrics: <i>(Detail what metrics will be collected, monitored, trended, and reported for acceptance testing. Describe how these metrics will be used in decision-making.)</i></p>		
<p>8. Test Records: <i>(Describe the process and the requirements for documenting test records that will be followed for the life-cycle of the software.)</i></p>		
<p>9. Risks to the successful completion of the test plan. <i>(List major assumptions and identify significant constraints on testing such as resource availability, schedule changes, and time constraints. Identify the level of risk associated with each assumption and specify a contingency plan for each risk factor with details of the mitigating actions that are to be implemented should the potential risk occur.)</i></p>		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated this life-cycle document. Needed configuration control items are identified, and configuration management has been maintained.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment D Software Test Document Criteria and Review Form

Ash Fall Project AFP-AP-06, Form 6-4	Software Test Document Criteria and Review Form	Form Version: 0 Total Pages:
<p>This document is prepared for testing for acceptance of baseline products and for final acceptance documentation. This is used as the cover page for testing documentation (life-cycle document). This documentation shall include the results of the execution of test cases.</p>		
<p>Unique Identification Number (UID):</p>		
<p>Criteria for this Phase:</p> <ul style="list-style-type: none"> • Computer program tested, including system software used • Computer software and hardware configuration used • Test equipment and calibration, where applicable • Date of Test • Tester or data recorder • Type of observation • Test problems 		
<p>Test Case Results Evaluated Using One of the Following Methods: <i>(Simulation models used, where applicable)</i></p> <ul style="list-style-type: none"> • Manual calculations (Hand, Excel, MathCAD, etc.)? • Manual inspection (for example captures of user interface screens?) • Calculations using comparable proven problems or benchmarks? • Other optional methods (explanations required)? 		
<p>Did the executed tests satisfy acceptance criteria as identified in applicable requirements and design documents?</p>		
<p>Are tests organized in such a way to allow traceability to requirements and design?</p>		
<p>Are final tests documented and reproducible?</p>		
<p>Specify any deviations from the approved acceptance test plan. Provide justification that the deviation(s) do not prevent the software product from adequately and correctly performing all its intended functions. Deviations from approved test plans should be reviewed and approved prior to the resumption of testing: (if any)</p>		
<p>Test failures and actions required for correction are documented and reviewed.</p>		
<p>Are the test results acceptable?</p>		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated the test requirements and test results of this life-cycle document. This signature verifies that the test requirements have been met and that comments of the independent review have been incorporated. The test results of the software ensure the software is acceptable for use.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment E Installation and Acceptance Life Cycle Document/Phase Criteria

Ash Fall Project AFP-AP-06, Form 6-5	Installation and Acceptance Life Cycle Document/Phase Criteria	Form Version: 0 Total Pages:
<p>This document is prepared for incorporating the software into a system. This document shall include results of the test case execution for system installation and integration, user instructions, and documentation of the software acceptance for operational use. This is used as the cover page for installation and acceptance documentation (life-cycle document).</p>		
<p>Unique Identification Number (UID):</p>		
<p>Test Case Results for Installation and Integration are Documented</p>		
<p>Did the executed tests satisfy acceptance criteria as identified in applicable requirements and design documents?</p>		
<p>Are the test results acceptable?</p>		
<p>User Instructions Include:</p> <ul style="list-style-type: none"> • The approved operating systems • A description of the user’s interaction with the software • A description of any required training necessary to use the software • Input and output specifications • Input and output formats • A description of software and hardware limitations • A description of user messages initiated as a result of improper input and how the user can respond, including anticipated errors and how users can respond • Information for obtaining user and maintenance support • Applicable installation procedures and available sample/examples of execution • In-use test that shall be performed, documented, and verified to provide confirmation of acceptable performance of software • Cyber security specifications • Risk and safety specifications 		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated the design verification activities and performed a technical review of the acceptance testing, installation results, and user instructions. This signature verifies acceptance of the software for operational use, and configuration baselines, documentation, and reviews have been completed.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment F Software Quality Assurance Plan (SQAP)

(This is an example of items that need to be included in the SQAP, at a minimum.)

Change Log

Revision	Description	Author	Date

Definitions and Acronyms

1.0 Scope

This SQAP was developed in accordance with... This document defines the software quality assurance requirements for the following software: ...

2.0 Software Description

The purpose of this software is ... The hardware will consist of ...

3.0 Roles and Responsibilities

3.1 Quality Assurance Lead

- Reviewing and approving SQAP
- Ensuring independent reviews have been completed when required

3.2 Software Developer

- Ensuring software development process is documented per AFP-AP-06 and this plan
- Ensuring Software QA Plans (SQAP) and other required documents are developed, reviewed and approved as required
- Ensuring compliance with requirements and reviews as defined in SQAP

3.3 Independent Reviewers

- Reviewing and approving Software Lifecycle Documents, as required

3.4 Software Custodian

- Maintaining the Software Inventory List
- Maintaining Software Configuration Management Library
- Controlling the use of software

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

4.0 Software QA Requirements

4.1 Lifecycle Documentation

The <software> will be developed using ...

The Software Developer shall maintain draft documents, source code, and other configuration items on the Configuration Status Accounting system until they have been accepted or approved and are placed under configuration control.

4.1.1 Requirements Phase

The software requirements shall be identified in the <software> Requirements Document (Filename: XXXXX-Requirements-Vn.n). The document shall satisfy the Requirements Phase documentation requirements as defined in Section 8.3.2 of AFP-AP-06. Upon review and approval, this document shall be placed under configuration control (Section 4.2).

4.1.2 Design Phase

The software design shall be documented in the <software> Design Document (Filename: XXXXX-Design-Vn.n). This document shall satisfy the Design Phase documentation requirements as defined in Section 8.3.3 of AFP-AP-06. Upon review and approval, this document shall be placed under configuration control (Section 4.2).

4.1.3 Implementation Phase

The software implementation shall be documented in the <software> Implementation Document (Filename: XXXXX-Implementation-Vn.n). This document shall satisfy the Implementation Phase documentation requirements as defined in Section 8.3.4 of AFP-AP-06. Upon review and approval, this document shall be placed under configuration control (Section 4.2).

4.1.4 Test Phase

The Software Test Plan (Filename: XXXXX-TestPlan-Vn.n) shall be developed and include test cases to verify conformance to all software acceptance criteria enumerated in the Requirements Document. This test plan shall include a pass/fail checklist to facilitate tracking of test results. Upon review and approval, this document shall be placed under configuration control (Section 4.2).

Environmental Management Consolidated Business Center

Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

The Software Developer shall perform the procedure described by the Software Test Plan and record the results, noting any deviation from expected results. Upon review and approval, the Software Test Plan Results (Filename: XXXXX-TestPlanResults-Vn.n) shall be placed under configuration control (Section 4.2) and the software shall be considered approved for use.

4.1.5 Installation and Acceptance

4.1.6 Operations and Maintenance

Software notifications shall be approved and implemented according to Section 8.3.7 of AFP-AP-06. Errors or other conditions adverse to quality shall be reported according to Section 4.5.

4.1.7 Retirement Phase

During this phase, the Software Custodian shall remove the software or otherwise prevent its use on any installed systems. The software inventory list shall be updated to indicate the software is retired.

4.2 Software Configuration Control

The Software Custodian shall maintain a Software Configuration Management Library on a file server (the contents of this server are regularly backed up to an offsite location). A project folder shall be created under the library to contain baseline folders for baselines that are under development and baselines that have been approved for use. The baseline folder may be organized further at the discretion of the Software Custodian, but it must contain all configuration items, the configuration item list (4.2.1), and the configuration status log (4.2.3).

When the software has been approved for use, the Software Custodian shall enter any outstanding Configuration Items into the baseline and archive a read-only copy of the baseline folder as an approved release.

4.2.1 Configuration Identification

The Software Custodian shall maintain a list of all baseline Configuration Items in a spreadsheet (Filename: XXXXX-ConfigurationItems.xls) contained in the top level of each baseline folder. Each Configuration Item entry shall contain the following information:

- Unique Identification
- Name

Environmental Management Consolidated Business Center

Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

- Description
- Version/Revision
- Configuration Item Type (Document, Source Code, Support Software, etc.)

4.2.2 Configuration Change Control

Changes to an approved baseline shall meet the requirements of AFP-AP-06.

4.2.3 Configuration Status Accounting and Control

The Software Custodian shall maintain a log of configuration status in a spreadsheet (Filename: XXXXX-ConfigurationStatus.xls) in the top level of each baseline folder. Dated entries should be recorded for the addition or modification of configuration items during development and proposed and approved changes during change control. These entries shall include the following information:

- Date
- Configuration Item UID (if applicable)
- Entry Description
- Status

4.2.4 Support Software

Software components included in the <software> that are classified as support software shall be evaluated and reviewed as part of the design phase and tested and accepted during test phase. Distribution files or media for all support software shall be controlled as part of Section 4.2, Software Configuration Control.

4.3 Evaluation (Existing/Acquired Software)

All software that is not developed for the Ash Fall Project will be considered Support Software and is covered under 4.2.4 of this document and AFP-AP-06, Section 8.4.5.

4.4 Software Procurement

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

4.5 Problem Reporting and Corrective Action

Any errors or failure to meet specified requirements that are discovered in the <software> should be reported according to AFP-AP-09, *Problem Reporting and Corrective Action*.

4.6 Software Inventory List (SIL)

The Software Custodian shall maintain the SIL according to AFP-AP-06, Section 8.8. This list shall be located on the file server.

4.7 Records

During retirement phase of the software, records produced according to AFP-AP-06 and this plan that require review and approval shall be submitted to project records according to AFP-AP-20, *Quality Assurance Records*.

Approvals

Author (signature)

Author (print)

Date

I have reviewed this Software Quality Assurance Procedure/Plan and found it to be acceptable.

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center
Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Attachment G – Software Change Request

Ash Fall Project AFP-AP-06, Form 6-7	Software Change Request	Form Version: 0 Total Pages:
Software Identification		
Software UID		
Initiated by/Date:		
Description of the Requested Change:		
Rationale for the Requested Change:		
Evaluation (Changes should be made or not):		
Change Request Number Assigned:		
Requirements for Retesting and Acceptance of Test Results:		
Documentation Revision Required:		
Disposition:		

Environmental Management Consolidated Business Center Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Approvals

Author (signature)

Author (print)

Date

I have evaluated the software change request and performed a technical review of the acceptance testing and documentation revision. This signature verifies acceptance of the software for operational use, and configuration baselines, documentation, and reviews have been completed.

Independent Reviewer
(signature)

Independent Reviewer (print)

Date

Quality Assurance Lead
(signature)

Quality Assurance Lead (print)

Date

Environmental Management Consolidated Business Center
Ash Fall Project

Software Management Control

Procedure: AFP-AP-06
Revision 0, 12/20/15

Form 12-1 – Record of Revision

DOCUMENT: AFP-AP-06, *Software Management Control*

Revision Number	Description of Changes	Revision on Pages	Effective Date
0	Initial Issue	All	12/20/2015