7B Implementation Requirements for Hardware/Software Development

This section defines the requirements to be met by the hardware and software development implementation activities that are to be made available for review by the NRC.

7B.1 Software Management Plan

[*The Software Management Plan shall define:*

(a) the organization and responsibilities for development of the software design; the procedures to be used in the software development; the interrelationships between software design activities; and the methods for conducting software safety analyses.

Within the defined scope and content of the Software Management Plan, accepted methods and procedures for the above activities are presented in the following documents:

(i) [IEEE 730, Standard for Software Quality Assurance Plans, Section 3.4;

(ii) ASME NQA-2a, Part 2.7, Quality Assurance Requirements of Computer Software for Nuclear Facility Application;

(iii) ANSI/IEEE-ANS-7-4.3.2, Application Criteria for Digital Computers in Safety Systems for Nuclear Facilities (to be replaced by the issued version of P 7-4.3.2, "Standard Criteria for Digital Computers Used in Safety Systems of Nuclear Power Generation Stations");

(iv) IEC 880, Software for computers in the safety systems of nuclear power stations, Section 3.1;

(v) IEEE 1228 (draft), Standard for Software Safety Plans;

(vi) IEEE 1012, Standard for Software Verification and Validation Plans, Section 3.5;

(vii) IEEE 830, Guide to Software Requirements Specifications, Section 5;

(viii) IEEE 1042, Guide to Software Configuration Management.]*

Note that within the set of documents listed above, differences may exist regarding specific methods and criteria applicable to the Software Management Plan. In situations where such differences exist, all of the methods and criteria presented within those documents are considered to be equally appropriate and valid and, therefore, any of the above listed documents may be selected as the basis for elements of the SMP.

* See Sections 7A.1(2) and 7A.1(1).

(b) that the software safety analyses to be conducted for safety-related software applications shall:

(i) identify software requirements having safety-related implications;

(ii) document the identified safety-critical software requirements in the software requirements specification for the design;

(iii) incorporate in to the software design the safety-critical software functions specified in the software requirements specification;

(iv) identify in the coding and test of the developed software, those software modules which are safety-critical;

(v) evaluate the performance of the developed safety-critical software modules when operated within the constraints imposed by the established system requirements, software design, and computer hardware requirements;

(vi) evaluate software interfaces of safety-critical software modules;

(vii) perform equipment integration and validation testing that demonstrate that safety-related functions identified in the design input requirements are operational.

(c) the software engineering process, which is composed of the following life-cycle phases:

(i) Planning

(ii) Design Definition

(iii) Software Design

(iv) Software Coding

(v) Integration

(vi) Validation

(vii) Change control

(d) the Planning phase design activities, which shall address the following system design requirements and software development plans:

(i) Software Management Plan

(ii) Software Configuration Management Plan

(iii) Verification and Validation Plan

(iv) Equipment design requirements

(v) Safety analysis of design requirements

(vi) disposition of design and/or documentation nonconformances identified during this phase

(e) the Design Definition phase design activities, which shall address the development of the following implementing equipment design and configuration requirements:

(i) equipment schematic;

(ii) equipment hardware and software performance specification;

(iii) equipment user's manual;

(iv) data communications protocol;

(v) safety analysis of the developed design definition;

(vi) disposition of design and/or documentation nonconformances identified during this phase.

(f) the Software Design phase, which shall address the design of the software architecture and program structure elements, and the definition of software module functions:

(i) Software Design Specification;

(ii) safety analysis of the software design;

(iii) disposition of design and/or documentation nonconformances identified during this phase.

(g) the Software Coding phase, which shall address the following software coding and testing activities of individual software modules:

(i) software source code;

(ii) software module test reports;

(iii) safety analysis of the software coding;

(iv) disposition of nonconformances identified in this phase's design documentation and test results.

(h) the Integration phase, which shall address the following equipment testing activities that evaluates the performance of the software when installed in hardware prototypical of that defined in the Design Definition phase:

(i) integration test reports;

(ii) safety analysis of the integration test results;

(iii) disposition of nonconformances identified in this phase's design documentation and test results.

(i) the Validation phase, which comprises the development and implementation of the following documented test plans and procedures:

(i) validation test plans and procedures;

(ii) validation test reports;

(iii) description of as-tested software;

(iv) safety analysis of the validation test results;

(v) disposition of nonconformances identified in this phase's design documentation and test results;

(vi) software change control procedures.

(j) the Change Control phase, which begins with the completion of validation testing, and addresses changes to previously validated software and the implementation of the established software change control procedures.

7B.2 Configuration Management Plan

The Configuration Management Plan shall define:

(a) the specific product or system scope to which it is applicable, the organizational responsibilities for software configuration management, and methods to be applied to:

(i) identify design interfaces;

(ii) produce software design documentation;

(iii) process changes to design interface documentation and software design documentation;

(iv) process corrective actions to resolve deviations identified in software design and design documentation, including notification to end user of errors discovered in software development tools or other software;

(v) maintain status of design interface documentation and developed software design documentation;

(vi) designate and control software revision status. Such methods shall require that software code listings present direct indication of the software code revision status.

Within the defined scope and content of the Configuration Management Plan, accepted methods and procedures for the above activities are presented in the following documents:

(i) [IEEE 1042, Guide to Software Configuration Management;

(ii) IEEE 828, Standard for Software Configuration Management Plans;

(iii) ANSI/IEEE-ANS-7-4.3.2, Application Criteria for Digital Computers in Safety Systems for Nuclear Facilities (to be replaced by the issued

version of P 7-4.3.2, "Standard Criteria for Digital Computers Used in Safety Systems of Nuclear Power Generation Stations");

(iv) IEC 880, Software for computers in the safety systems of nuclear power stations.]*

Note that within the set of documents listed above, differences may exist regarding specific methods and criteria applicable to the Configuration Management Plan. In situations that such differences exist, all of the methods and criteria presented within those documents are considered to be equally appropriate and valid. Therefore, any of the above listed documents may be selected as the basis for elements of the CMP.

(b) methods for, and the sequencing of, reviews to evaluate the compliance of software design activities with the requirements of the CMP;

(c) the configuration management of tools (such as compilers) and software development procedures;

(d) methods for the dedication of commercial software for safety-related usage;

(e) methods for tracking error rates during software development, such as the use of software metrics;

(f) the methods for design record collection and retention.

7B.3 Verification and Validation Plan

The Verification and Validation Plan shall define:

(a) that baseline reviews of the software development process are to be conducted during each phase of the software development life cycle and the scope and methods to be used in the baseline reviews to evaluate the implemented design, design documentation, and compliance with the requirements of the Software Management Plan and Configuration Management Plan.

Within the defined scope and content of the Verification and Validation Plan, accepted methods and procedures for the above activities are presented in the following documents:

(i) [IEEE 1012, Standard for Software Verification and Validation Plans;

(ii) ANSI/IEEE-ANS-7-4.3.2, Application Criteria for Digital Computers in Safety Systems for Nuclear Facilities (to be replaced by the issued

* See Sections 7A.1(2) and 7A.1(1).

version of P 7-4.3.2, "Standard Criteria for Digital Computers Used in Safety Systems of Nuclear Power Generation Stations");

(iii) IEC 880, Software for computers in the safety systems of nuclear power stations.*]

Note that within the set of documents listed above, differences may exist regarding specific methods and criteria applicable to the Verification and Validation Plan. In situations that such differences exist, all of the methods and criteria presented within those documents are considered to be equally appropriate and valid and, therefore, any of the above listed documents may be selected as the basis for elements of the V&VP.

(b) that verification shall be performed as a controlled and documented evaluation of the conformity of the developed design to the documented design requirements at each phase of baseline review.

(c) that the use of commercial software and commercial development tools for safety-related applications is a controlled and documented procedure.

(d) that validation shall be performed through controlled and documented testing of the developed software that demonstrates compliance of the software with the software requirements specifications.

(e) that for safety-related software, verification reviews and validation testing are to be conducted by personnel who are knowledgeable in the technologies and methods used in the design, but who did not develop the software design to be reviewed and tested.

(f) that for safety-related software, design verification reviews shall be conducted as part of the baseline reviews of the design material developed during the Planning through Integration phases of the software development life-cycle (as defined in Criterion 1b, above), and that validation testing shall be conducted as part of the baseline review of the Validation phase of the software development life-cycle.

(g) that validation testing shall be conducted per a documented test plan and procedure.

(h) that for non-safety-related software development, verification and validation shall be performed through design reviews conducted as part of the baseline reviews completed at the end of the phases in the software development life cycle. These design reviews shall be performed by personnel knowledgeable in the technologies and methods used in the design development.

* See Sections 7A.1(2) and 7A.1(1).

(i) the products which shall result from the baseline reviews conducted at each phase of the software development life-cycle; and that the defined products of the baseline reviews and the V&V Plan shall be documented and maintained under configuration management.

(j) the methods for identification, closure, and documentation of design and/or design documentation nonconformances.

(k) that the software development is not complete until the specified verification and validation activities are complete and design documentation is consistent with the developed software.]*

Completion of Software Development

Software development has been completed as defined in the SMP, CMP, and V&VP.

* See Section 7A.1(1).