

Enclosure 2

MFN 09-358

**GEH Nuclear Energy,
“ESBWR – Software Management Program Manual,”**

NEDO-33226,” Revision 4,

May 2009 — Non-Proprietary Version



HITACHI

GE Hitachi Nuclear Energy

NEDO-33226
Revision 4
Class I
DRF 0000-0051-3897
May 2009

Licensing Topical Report

**ESBWR – SOFTWARE MANAGEMENT
PROGRAM MANUAL**

Copyright 2006, 2009 GE-Hitachi Nuclear Energy Americas LLC

All Rights Reserved

PROPRIETARY INFORMATION NOTICE

This enclosure contains General Electric Hitachi Nuclear Energy (GEH) proprietary information and is furnished in confidence solely for the purpose(s) stated in the transmittal letter. No other use, direct or indirect, of the document or the information it contains is authorized. Furnishing this enclosure does not convey any license, express or implied, to use any patented invention or, except as specified above, any proprietary information of GEH disclosed herein or any right to publish or make copies of the enclosure without prior written permission of GEH. The proprietary information is enclosed within double brackets. [[This sentence is an example.^{3}]]. Figures and large equation objects are enclosed in double brackets. The superscript notation {3} refers to Paragraph (3) of the enclosed affidavit, which provides the basis for the proprietary determination.

IMPORTANT NOTICE REGARDING THE CONTENTS OF THIS REPORT

Please Read Carefully

The information contained in this document is furnished for the purpose of supporting the NRC review of the certification of the ESBWR. The only undertakings of GEH with respect to information in this document are contained in contracts between GEH and participating utilities, and nothing contained in this document shall be construed as changing those contracts. The use of this information by anyone other than those participating entities and for any purposes other than those for which it is intended is not authorized; and with respect to any unauthorized use, GEH makes no representation or warranty, and assumes no liability as to the completeness, accuracy, or usefulness of the information contained in this document.

Table of Contents

1.	INTRODUCTION	1-1
1.1	OVERVIEW	1-1
1.2	PURPOSE AND SCOPE.....	1-1
1.3	ACRONYMS, ABBREVIATIONS, AND DEFINITIONS	1-2
2.	APPLICABLE DOCUMENTS	2-1
2.1	SUPPORTING DOCUMENTS	2-1
2.2	REGULATORY DOCUMENTS, CODES AND STANDARDS.....	2-1
2.2.1	NUREG.....	2-1
2.2.2	Code of Federal Regulations (CFR)	2-1
2.2.3	U.S. Nuclear Regulatory Commission (NRC) Regulatory Guides (RG).....	2-1
2.2.4	Institute of Electrical and Electronic Engineers (IEEE) Standards.....	2-2
2.3	SUPPLEMENTAL DOCUMENTS.....	2-3
2.4	ADDITIONAL IEEE STANDARD GUIDANCE	2-8
2.5	INTERNATIONAL STANDARDS	2-8
3.	SOFTWARE MANAGEMENT PLAN.....	3-1
3.1	PURPOSE AND SCOPE.....	3-1
3.2	ORGANIZATION	3-1
3.2.1	I&C Design Engineering	3-1
3.2.2	Software Project Engineering	3-2
3.2.3	Configuration Management Manager	3-2
3.2.4	Software Quality Assurance Manager	3-2
3.2.5	Project Management Team	3-3
3.2.6	Training.....	3-3
3.2.7	Cyber Security Team	3-3
3.3	ORGANIZATIONAL BOUNDARIES AND INTERFACES	3-3
3.4	ORGANIZATIONAL RESPONSIBILITIES.....	3-3
3.4.1	(Deleted)	3-3
3.4.2	(Deleted)	3-4
3.4.3	I&C Design Engineering Manager	3-4
3.4.4	Software Project Engineering Manager	3-4

3.4.5	Software Quality Assurance Manager	3-4
3.4.6	Training Services Lead	3-4
3.4.7	Configuration Management Manager	3-4
3.4.8	Technical Project Engineer	3-4
3.5	SOFTWARE MANAGEMENT PLAN CHANGE CONTROL PROCESS	3-4
3.6	PROJECT MANAGEMENT PRIORITIES, MONITORING, AND CONTROL ..	3-5
3.6.1	Project Initiation	3-5
3.6.2	Project Planning and Scheduling	3-6
3.6.3	Project Execution	3-6
3.6.4	Project Controls	3-6
3.6.5	Post-Delivery Closeout	3-7
3.7	METHODS AND TOOLS FOR PROJECT MANAGEMENT	3-8
3.7.1	Methods	3-8
3.7.2	Tools	3-8
3.8	BUDGET	3-8
3.9	RISK MANAGEMENT	3-9
3.10	SECURITY	3-9
3.11	TRAINING AND QUALIFICATIONS	3-10
4.	MANAGEMENT PROCESS	4-1
5.	SOFTWARE DEVELOPMENT PLAN.....	5-1
5.1	INTRODUCTION	5-1
5.2	PURPOSE AND SCOPE	5-1
5.3	ORGANIZATION OF SOFTWARE LIFE CYCLE PROCESS	5-1
5.4	METHODS	5-3
5.4.1	Configuration Management and Change Control	5-3
5.4.2	Verification and Validation	5-3
5.4.3	Testing	5-3
5.4.4	Software Safety Analysis	5-3
5.4.5	Baseline Review	5-4
5.4.6	Deferred Design Verification	5-4
5.4.7	Cyber Security Assessment	5-4
5.5	TOOLS	5-4
5.5.1	Support Software	5-5

5.5.2	Requirements Traceability Matrix	5-5
5.6	PLANNING PHASE	5-18
5.6.1	Planning Phase Inputs	5-18
5.6.2	Planning Phase Outputs	5-18
5.6.3	Software Safety Analysis Report	5-20
5.6.4	Cyber Security Assessment Report.....	5-20
5.6.5	Planning Phase Baseline Review Record	5-20
5.7	REQUIREMENTS PHASE	5-20
5.7.1	Requirements Phase Inputs	5-20
5.7.2	Requirement Phase Outputs	5-20
5.7.3	Requirements Phase Activities	5-21
5.7.4	Hardware/Software Specification	5-22
5.7.5	Software Requirements Specification.....	5-23
5.7.6	System Requirements Specification	5-25
5.7.7	Data Communications Protocol	5-25
5.7.8	User Interface Specification.....	5-26
5.7.9	Software Support Tools/Documentation for Software Development.....	5-26
5.7.10	Software Safety Analysis Report	5-28
5.7.11	Cyber Security Assessment Report.....	5-28
5.7.12	Requirements Phase Baseline Review Record	5-28
5.8	DESIGN PHASE	5-28
5.8.1	Design Phase Inputs.....	5-28
5.8.2	Design Phase Outputs	5-28
5.8.3	Design Phase Activities	5-29
5.9	IMPLEMENTATION PHASE	5-34
5.9.1	Implementation Phase Inputs.....	5-34
5.9.2	Implementation Phase Outputs	5-34
5.9.3	Implementation Phase Activities	5-35
5.10	TEST PHASE	5-38
5.10.1	Test Phase Inputs	5-38
5.10.2	Test Phase Outputs	5-38
5.10.3	Software Validation Test	5-39
5.10.4	Software Validation Test Report	5-39
5.10.5	Production Release	5-40

5.10.6	Software Release Notes	5-40
5.10.7	Cyber Security Assessment Report.....	5-40
5.10.8	(Deleted)	5-40
5.10.9	Test Phase Baseline Review Record.....	5-40
5.11	INSTALLATION PHASE.....	5-40
5.11.1	Installation Phase Inputs	5-40
5.11.2	Installation Phase Outputs	5-41
5.11.3	System Factory Acceptance Test.....	5-42
5.11.4	Multi-System Factory Acceptance Test.....	5-42
5.11.5	Site Acceptance Test.....	5-43
5.11.6	Software Operations & Maintenance Manuals	5-43
5.11.7	Software Training Manuals	5-43
5.11.8	Software Installation.....	5-43
5.11.9	Cyber Security Assessment Report.....	5-43
5.11.10	Installation Phase Baseline Review Record.....	5-44
5.12	OPERATIONS AND MAINTENANCE PHASE.....	5-44
5.12.1	Operations and Maintenance Phase Inputs	5-44
5.12.2	Operations and Maintenance Phase Outputs.....	5-44
5.12.3	Operations and Maintenance Activities	5-45
5.12.4	(Deleted)	5-46
5.12.5	Operations and Maintenance Phase Baseline Review Record.....	5-46
5.13	RETIREMENT PHASE.....	5-46
5.13.1	Retirement Phase Activities Baseline Review Record	5-47
6.	SOFTWARE INTEGRATION PLAN	6-1
6.1	INTRODUCTION	6-1
6.2	PURPOSE.....	6-1
6.3	SOFTWARE INTEGRATION.....	6-1
6.4	ORGANIZATION AND MANAGEMENT.....	6-2
6.5	MANAGEMENT AND ORGANIZATION INTERFACES	6-2
6.6	SCHEDULING AND PLANNING.....	6-2
6.7	RESOURCES	6-2
6.8	TRAINING	6-2
6.9	REVIEWS.....	6-2

6.10	TEST PERSONNEL ROLES AND RESPONSIBILITIES	6-2
6.10.1	Responsible Technical Project Engineer	6-3
6.10.2	Responsible Test Engineer.....	6-3
6.10.3	Cyber Security Test Engineer	6-3
6.10.4	Test Personnel Qualifications	6-3
6.11	SOFTWARE FUNCTIONAL TEST	6-3
6.11.1	Software Functional Test Guidelines	6-3
6.11.2	Software Functional Methods	6-5
6.11.3	Software Functional Test Documentation	6-10
6.12	SOFTWARE VALIDATION TEST.....	6-12
6.12.1	Software Validation Test Guidelines - Design Team	6-12
6.12.2	Software Validation Test Guidelines - IVVT	6-14
6.12.3	Software Validation Test Documentation.....	6-14
6.13	PROBLEM REPORTING	6-17
6.14	MEASUREMENT AND METRICS	6-17
7.	SOFTWARE INSTALLATION PLAN	7-1
7.1	INTRODUCTION	7-1
7.2	PURPOSE	7-1
7.3	SCOPE	7-1
7.4	ORGANIZATION, MANAGEMENT AND RESPONSIBILITIES	7-1
7.5	INSTALLATION ACTIVITIES	7-1
7.5.1	Software Installation Procedure.....	7-1
7.5.2	Software Installation Reporting.....	7-2
7.5.3	Installation Configuration Tables	7-3
7.5.4	Operations and Maintenance Manual	7-3
7.5.5	Training Manuals.....	7-3
7.6	METHODS AND TOOLS.....	7-3
7.6.1	Installation Methods	7-3
7.6.2	Archive Retrieval.....	7-4
7.6.3	Installation Test	7-4
7.6.4	Installation Documentation and Problem Reporting.....	7-4
7.6.5	Verification and Validation Methods	7-4
7.7	MEASUREMENTS AND METRICS	7-4

8.	SOFTWARE OPERATIONS AND MAINTENANCE PLAN	8-1
8.1	INTRODUCTION	8-1
8.1.1	Purpose	8-1
8.1.2	Scope.....	8-1
8.2	ORGANIZATION, MANAGEMENT AND RESPONSIBILITIES	8-1
8.3	ACTIVITIES.....	8-1
8.3.1	Operation Phase Activities.....	8-1
8.3.2	Maintenance Phase Activities.....	8-2
8.4	(Deleted)	8-2
8.4.1	(Deleted)	8-2
8.4.2	(Deleted)	8-2
8.5	OPERATION AND MAINTENANCE MANUAL	8-2
8.5.1	Software Operation and Maintenance Manuals.....	8-3
8.5.2	Verification and Validation Methods	8-4
8.6	MEASUREMENT AND METRICS	8-4
9.	SOFTWARE TRAINING PLAN.....	9-1
9.1	INTRODUCTION	9-1
9.1.1	Purpose	9-1
9.1.2	Scope.....	9-1
9.2	TRAINING ORGANIZATION.....	9-1
9.2.1	Responsibilities and Qualification	9-1
9.3	TRAINING ACTIVITIES	9-1
9.3.1	Training Plan	9-2
9.3.2	Training Manual and Materials.....	9-2
9.3.3	Training Program and Training Courses.....	9-3
9.3.4	Training Implementation	9-4
9.4	METHODS AND TOOLS.....	9-5
9.5	MEASUREMENT AND METRICS	9-5
10.	APPENDICES	10-1
10.1	APPENDIX A SOFTWARE PLANS CONFORMANCE REVIEW	10-1
10.2	APPENDIX B ACRONYMS AND ABBREVIATIONS.....	10-13
10.3	APPENDIX C DEFINITIONS	10-18

10.4	APPENDIX D SOFTWARE FUNCTIONAL TEST DATA SHEET (EXAMPLE).....	10-27
10.5	APPENDIX E SOFTWARE FUNCTIONAL TEST METRICS SHEET (EXAMPLE)..	10-28
10.6	APPENDIX F SOFTWARE VALIDATION TEST METRICS SHEET (EXAMPLE)..	10-29

List of Tables

Table 5.6-1 Planning Phase Output Documents	5-19
Table 5.7-1 Requirements Phase Output Documents	5-21
Table 5.8-1 Design Phase Output Documents	5-29
Table 5.9-1 Implementation Phase Output Documents	5-35
Table 5.10-1 Test Phase Output Documents.....	5-39
Table 5.11-1 Installation Phase Input Documents	5-41
Table 5.11-2 Installation Phase I Output Documents	5-41
Table 5.11-3 Installation Phase II Output Documents	5-42
Table 5.11-4 Installation Phase III Output Documents	5-42
Table 5.12-1 O&M Phase Output Documents	5-45
Table 5.13-1 Retirement Phase Output Documents.....	5-47

List of Figures

Figure 5-1. Software Life Cycle Process Overview.....	5-5
Figure 5-2. Software Life Cycle Process-Planning Phase.....	5-6
Figure 5-3. Software Life Cycle Process-Requirements Phase.....	5-7
Figure 5-4. Software Life Cycle Process-Design Phase.....	5-8
Figure 5-5. Software Life Cycle Process-Implementation Phase.....	5-9
Figure 5-6. Software Life Cycle Process-Test Phase.....	5-10
Figure 5-7a. Test Software Life Cycle Process-Installation Phase (Software Installation).....	5-11
Figure 5-7b. Software Life Cycle Process-Installation Phase (System Installation).....	5-12
Figure 5-8. Software Life Cycle Process-Installation Phase (Site Installation).....	5-13
Figure 5-9. Software Life Cycle Process-Operations and Maintenance Phase and Retirement Phase	5-14
Figure 5-10. Software Life Cycle Process Notes.....	5-15
Figure 5-11. Hardware and Software Design Overview.....	5-16
Figure 5-12. Cyber Security Interaction Model with SMPM and SQAPM Activities.....	5-17

Summary of Changes From previous revision

Item	Location	Change
1	Entire document	Revised entire document to correct the internal procedure reference to incorporate RAI 7.1-127 comment.
2	Entire document	Revised entire document to correct the references made the to the applicable documents listed in Section 2.0 to incorporate RAI 7.1-128 comment.
3	Entire document	Corrected typographical and grammatical errors, consistent use of document titles and acronyms, list numbering, and paragraph formatting to improve clarity and readability of SMPM.
4	Entire document	Deleted reference to "ESBWR" as appropriate to make this SMPM more generic.
5	Entire document	Changed "Cyber Security Analysis" to "Cyber Security Assessment" and "Cyber Security Analysis Report" to "Cyber Security Assessment Report" to be consistent with CySPP.
6	Entire document	Replaced "Instrument" with "component" as appropriate to be more general.
7	Entire document	Replaced I&C and Electrical Systems Engineering (I&C/ESE) with I&C Design Engineering (I&C) to make the SMPM more generic.
8	Section 2.3	Deleted 2.b as it is not referenced by the SMPM.
9	Sections 2.3(2.m), 3.5, 10.3 (Appendix C), and Subsection 3.6.4,	Changed "Self-Assessment, Corrective Action and Audits" to "Corrective Action Process" as GEH has revised the title of this procedure.
10	Section 3.1	Revised Section 3.1 to clarify the purposes of SMP to incorporate RAI 14.3-402 comment.
11	Subsection 3.2.1	<ul style="list-style-type: none"> • Revised 1st paragraph to clarify Cyber Security team roles. • Moved last paragraph to new Subsection 3.2.7.
12	Subsection 3.2.2	Deleted reference to the Simulation Assisted engineering (SAE) and Human Factors Engineering (HFE) teams as this is outside the scope of this SMPM.

Item	Location	Change
13	Subsection 3.2.7 (New)	Moved from the last paragraph of Subsection 3.2.1. This paragraph was revised to clarify roles and responsibilities of the Cyber Security Team by referencing the CySPP.
14	Subsection 3.4.1	Deleted “New Unit Engineering Manager” subsection to make the SMPM more generic.
15	Subsection 3.4.2	Deleted “ESBWR Engineering Manager” subsection to make the SMPM more generic.
16	Section 3.10	Deleted 3rd paragraph “The Cyber Security Program is defined in ESBWR Cyber Security Program Plan [2.3(1.b)]” as it was duplicated in the 2nd paragraph.
17	Section 5.3	Revised the Installation Phase to clarify the purpose and scope, and clarify assessment is performed in accordance with CySPP.
18	Subsection 5.3.8.5	Deleted Bullets #11 “Ensure that an SSA has been performed as defined in the SQAPM [2.3(1.a) Section 4.0].” and #12 “Ensure that a CySA has been performed in accordance with the CySPP [2.3(1.b)].” as these are out of Previously Developed Software Report scope.
19	Subsection 5.4.1	Added new paragraph to specify that the design team is required to resolve comments, anomalies and discrepancies identified during the change control process.
20	Subsection 5.4.2	<ul style="list-style-type: none"> • Changed subsection heading from “Independent Verification” to “Verification and Validation”. • Revised subsection to provide a consistent description and reference of V&V tasks as defined in the SQAPM. • Added new sentence to specify that the Responsible Engineer is required to resolve V&V comments identified during the V&V.
21	Subsection 5.4.3	Added new sentence to specify that the Responsible Engineer shall resolve the test discrepancy or anomaly detected during testing.
22	Subsection 5.4.4	<ul style="list-style-type: none"> • Added new sentence to specify that the Responsible Engineer shall resolve the anomalies noted during the SSA task • Deleted last sentence in 1st paragraph as it is repeated in 2nd paragraph.

Item	Location	Change
23	Subsection 5.4.5	<ul style="list-style-type: none"> • Revised subsection to clarify and minimize duplicity of the scope and baseline review as described in SQAPM, and included appropriate references to the SQAPM. • Added new sentence to specify that the non-conformances identified during the baseline review shall be resolved by the design team.
24	Subsection 5.4.7	<ul style="list-style-type: none"> • Deleted 3rd sentence in 2nd paragraph as it is outside the scope of this SMPM, and was not consistent with the CySPP. • Added new sentence to specify that the Responsible Engineer shall resolve the comments identified during the Cyber Security Assessment. • Added reference to CySPP to clarify that CDAs are described, identified and evaluated in accordance with CySPP.
25	Subsection 5.6.1	<ul style="list-style-type: none"> • Deleted reference to “Chapter 7” for completeness. • Combined the 3 (three) HFE documents into one “HFE Analysis Report” for completeness. • Added “Change Requests” to be consistent with the process described in the SMPM.
26	Subsections 5.6.3, 5.7.10, 5.8.3.11, 5.9.3.8	Revised these subsections to match the content of the paragraph with the heading.
27	Subsection 5.6.5	Replaced “The design documents have been verified and all system design, MMIS, HFE, and contractual requirements are incorporated” with “The concept documents have been verified” to be consistent with Section 5.6.
28	Subsections 5.6.5, 5.7.12, 5.8.3.13, 5.9.3.10, 5.10.9, 5.11.10	<ul style="list-style-type: none"> • Deleted the following sentence “After open items have been resolved, the results shall be submitted to the BRT as a revision to BRR for approval. The SQA team shall perform configuration audits during BRs.” as they are process related. • Added “and the associated resolution of the open items” to be included in the BRR for completeness.

Item	Location	Change
29	Subsections 5.7.2, 5.8.2, 5.9.2, 5.10.2	Added ““Except for the Baseline Review Record” to the beginning of the 3rd sentence to clarify to responsibilities of the Responsible Engineer, and the scope of phase baseline review.
30	Subsections 5.7.4	Revised Bullets 3, 8 and 9 to clarify boundaries, isolation and data communication requirements.
31	Subsections 5.7.5	Deleted Bullet #4, item #3 “Used of a minimum number of programming languages, compilers, and support packages in the implementation of the software” as this requirement is not verifiable nor a “software testing criteria and quality assurance requirements”.
32	Subsection 5.7.8	Deleted 2nd paragraph list of HFE documents as the HFE requirements are specified in the System Design Specification.
33	Subsection 5.7.9	<ul style="list-style-type: none"> • Replaced “independent verification and validation” and “IV&V” with “V&V” to clarify the V&V requirements for Software Support Tools. • Added new paragraph to clarify tools qualification.
34	Subsection 5.8.3.1	Deleted sentence, “SDD detail has to be sufficient to support V&V by an independent engineer” as it is redundant. Previous sentence already specified that the SDD should be in “sufficient detail”.
35	Subsection 5.8.3.2	Revised to clarify cyber security requirements concerning timing and sizing analysis by combining the last 2 bullets to form new paragraph.
36	Subsection 5.8.3.3	Added “secure coding practices” to include cyber security requirements in Software Coding Conventions and Guidelines Document.
37	Subsection 5.8.3.5	Deleted last paragraph as it the incorporation of PDS (Previously developed software) is outside the scope of PDS evaluation. Incorporation of PDS is discussed in Section 5.9.
38	Subsections 5.8.3.12, 5.9.3.9, 5.10.7, 5.11.9,	Revised these subsections to match the content of the paragraph with the heading.
39	Subsection 5.3.8.10	Revised section to indicate SFAT Test Plan and associated test documentation shall be baselined during the Test Phase.

Item	Location	Change
40	Subsection 5.9.3.3	Revised 2 nd paragraph to clarify that code review may be performed by individual who is responsible for the software development or coding on another project.
41	Subsection 5.9.3.4	Revised last sentence in 1st paragraph to clarify software functional testing.
42	Subsection 5.10.5	Replaced 2nd and 3rd sentence with “as described in the SQAPM [2.3(1.a)]” as Product Release is a QA task and is described in the SQAPM.
43	Subsection 5.10.8	Deleted this section as HFE V&V is outside the scope of the SMPM.
44	Subsection 5.10.6	Added reference to Subsection 7.5.1 where additional details on Software Installation Procedure are described.
45	Section 5.11	Deleted reference to SGI as it is outside the scope of this SMPM.
46	Subsection 5.11.2	Added ““Except for the Baseline Review Record” to the beginning of the paragraph to clarify the scope of Installation Phase baseline review.
47	Subsection 5.11.3	Deleted 1st sentence “The System Factory Acceptance Test (SFAT) plans, test design, test case and test procedure specifications and the Cyber Security SFAT Plan and Procedure Specifications shall be completed.” as it is an incomplete sentence.
48	Subsection 5.11.4	Deleted 3rd paragraph on HFE procedures and training requirements as it was outside the scope of MFAT.
49	Subsection 5.11.8	Replaced “HFE, ISV, V&V, Result Summary Report” with “Software Installation” as HFE activities and results are outside the scope of this SMPM.
50	Subsection 5.12.1	<ul style="list-style-type: none"> • Changed "HFEITS" to "Change Request" and "CAR" to "Problem Report" to be consistent with the SQAPM.

Item	Location	Change
51	Subsection 5.12.3	<ul style="list-style-type: none"> • Replaced HFEITS with Nuclear Customer Issue Resolution (CIR) tool as this is the tool used by GEH to log customer's issues. • Added new bullet (Bullet #1) "Evaluation of change request, reported problem or anomaly" to clarify O&M phase activities. • Revised the Baseline Change Assessment process to improve clarity of requirements. Moved and re-organized Bullets 2-5 in Subsection 5.13.1 as sub-bullets to Bullet #2. • Added new bullet (Bullet #3) "Determination of the extent of the design and implementation tasks to be reiterated" to clarify O&M phase activities. • Moved last bullet "Scheduling of software product retirement" to Subsection 5.13.1 as this is outside the scope of O&M phase.
52	Subsection 5.12.5	Deleted the Subsection as CySA Report will not be produce during the O&M Phase. The scope of the O&M Phase is to evaluate the change request or problem reported.
53	Subsection 5.12.5	<ul style="list-style-type: none"> • Revised paragraph to improve clarity. • Deleted "The SQA team shall perform configuration audits during baseline review" as this task will not be performed during the O&M Phase baseline review.
54	Subsection 5.13.1	<ul style="list-style-type: none"> • 2nd paragraph will be deleted and content move under the activities paragraph as (new) bullets: <ul style="list-style-type: none"> - Assuring the retired software product is properly de-activated. - Assuring data migration of quality data from the software product being retired. - Assuring documentation of the retired software product - Scheduling of software product retirement activities • Deleted paragraph describing Baseline Change Assessment process is it not part of Retirement Phase activities.
55	Entire Section 6.0	Changed "Test designer" and "validation designer" to "RTE".

Item	Location	Change
56	Section 6.1	Revised to delete most the Section as Section 6.3 provides better description of the SIP and to reduce redundancy.
57	Section 6.3	Revised to clarify the SIP requirements and process.
58	Section 6.8	Added more detail to clarify purpose of this section.
59	Subsection 6.10.1	<ul style="list-style-type: none"> • Deleted “software functional” to make test scope more generic. • Revised to clarify responsible Technical Project Engineer responsibilities.
60	Subsection 6.10.2	<ul style="list-style-type: none"> • Changed “Software Functional Test Engineer” to “Responsible Test Engineer” as test is not limited to software functional test. • Replaced “SDD intra-system data communication protocol specification and the SMPM” with “Subsections 6.11.1 and 6.12.1” as these Subsections described the test guidelines.
61	Subsection 6.10.4	Revised to clarify the qualification requirements of the Test Personnel.
62	Subsection 6.11.2.1.2 (old Subsection 6.12.1.2)	Deleted “Problem Reports (PRs)” as it is a repeat of Bullet #3.
63	Sections 6.11, 6.12 and 6.13	These sections are re-organized to group the requirements in logical sections and subsections to improve clarify and readability.
64	Subsection 6.11.1.1 (old Subsection 6.11.1)	Deleted last sentence as SQAPM does not specify software functional test as a hold points.
65	Subsection 6.11.1.2 (old Subsection 6.11.2)	<ul style="list-style-type: none"> • Revised subsection to clarify and improve quality. • Deleted the example for 2nd bullet as it is bad described, and the example is not needed as the meaning of the bullet is clear.
66	Entire Subsection 6.11.2.1.1 (old Subsection 6.12.1.1)	Deleted “/Unit” for consistency.

Item	Location	Change
67	Subsection 6.11.2.1.2 (old Subsection 6.12.1.2)	Revised this section to clarify and improve readability.
68	Subsection 6.11.2.2.1 (old Subsection 6.12.2.1)	Moved description of incremental testing methods from end of 3 rd paragraph to end of 2 nd paragraph.
69	Subsection 6.11.2.2.2 (old Subsection 6.12.2.2)	Changed “HFE” to “HSI”.
70	Subsection 6.11.2.2.4 (old Subsection 6.12.2.4)	<ul style="list-style-type: none"> • Changed heading from “Software Functional” to “Integration”. • Revised section to improve quality and for completeness.
71	Section 6.12 (old Subsection 6.12.3)	Changed “instrument” to “component” to make test scope more generic.
72	Subsection 6.12.1 (old Subsection 6.12.3.1)	Changed heading to “Software Validation Testing - Design Team”.
73	Subsection 6.12.1.2 (old Subsection 6.12.3.1.2)	Revised Subsection to clarify software validation test design requirements and activities.
74	Subsection 6.12.1.2.1 (old Subsection 6.12.3.1.2.1)	Revised Subsection, deleted Bullets #5 and #6 to clarify the Component Level Validation Test Design requirements.
75	Subsection 6.12.1.2.2 (old Subsection 6.12.3.1.2.2)	Revised Subsection, deleted Bullets #4 to #8 to clarify the System Level Validation Test Design requirements.
76	Subsection 6.12.1.4 (New)	Added Subsection 6.12.3.1.4, Validation Test Summary.

Item	Location	Change
77	Subsection 6.12.2 (old Subsection 6.12.3.2)	Changed heading to “Software Validation Testing – IVVT”.
78	Subsection 6.11.3.1 (old Subsection 6.13.1.1)	<ul style="list-style-type: none"> • Replaced “module test” in last paragraph with “software functional test”. • Reorganized 4th paragraph bullets list to improve and clarify the information to be included in the test data sheet.
79	Subsections 6.12.3, 6.12.3.1, 6.12.3.2, 6.12.3.3 and 6.12.3.4 (New)	<p>Added Subsection 6.12.3 Software Validation Test Documentation, and associated subsections.</p> <ul style="list-style-type: none"> • 6.12.3.1 - Software Validation Test Plan • 6.12.3.2 - Software Validation Test Cases and Test Procedure Specification • 6.12.3.3 - Software Validation Test Metrics Sheet • 6.12.3.4 - Software Validation Test Report
80	Subsection 7.5.2	<ul style="list-style-type: none"> • Deleted 2nd paragraph as control turnover to the Licensee is and outside the scope of this SMPM. • Replaced “Any Cyclic Redundancy Code or checksum that may be displayed by the installed software” with "software file integrity checks" to make requirement more generic to all software platform.
81	Subsection 7.6.5	Revised subsection to clarify V&V requirements for phase outputs.
82	Subsection 8.1.1	Revised to clarify the purpose of the SOMP.
83	Subsection 8.1.2	Revised to clarify the Scope of the SOMP.
84	Section 8.3	Revised Section to improve the clarity of O&M activities.
85	Subsection 8.3.1	Revised Section to improve the clarity of Operation Phase activities.
86	Subsection 8.3.2	Revised Section to improve the clarity of Maintenance Phase activities.
87	Section 8.4, Subsections 8.4.1, 8.4.2	Deleted the Procedures Section and Subsections as these are Licensee’s responsibility and is outside the scope of this SMPM.

Item	Location	Change
88	Section 8.5	<ul style="list-style-type: none"> • Changed Section heading from “Methods and Tools” to “Operation and Maintenance Manual” to be consistent with the content described. • Revised the 1st and 2nd paragraph to clarify the requirements necessary to support the development of O&M Manual.
89	Subsection 8.5.2	Changed O&M Phase outputs to O&M Manual and revised to clarify the V&V requirements.
90	Section 9.1	<ul style="list-style-type: none"> • Deleted “for the plant” as the Section 1.2 has defined the overall purpose and scope. • Deleted “Software training is performed prior to delivery of the software (System startup and post turn over) and during the O&M phase of the software life cycle” as training schedule is specified in the Training Plan (Subsection 9.3.1) not in the SMPM. • Deleted last sentence as HFE/MMIS IP does not provide the outline of the training plan. HFE Training Development Implementation Plan provides the strategy to training development, which is described in Subsection 9.3.3 (old Section 9.4).
91	Subsection 9.1.1	Added training program to bullet 1 for completeness.
92	Subsection 9.1.2	<ul style="list-style-type: none"> • Revised to clarify scope and improve readability. • Changed “SMPM” to “STmgP” to be more accurate.
93	Section 9.2	<ul style="list-style-type: none"> • Revised this section to clarify requirements. • Deleted “based on the Licensee needs” in 6th sentence as augmentation of the training staff could be due to other reasons.
94	Section 9.3	<p>Revised 1st bullet list for completeness.</p> <ul style="list-style-type: none"> • Bullet 2. Added “and training materials” • Bullet 3. Added “training program”. • Bullet 4. Added “program and training courses”. • Added new bullet. “Implement the Training Program”.
95	Section 9.3	Deleted 2nd paragraph as it is not accurate. IEEE 1074 defined training as an integral part of the development process.

Item	Location	Change
96	Sections 9.3 – 9.7	These sections are re-organized to group the requirements in logical sections and subsections to improve clarify and readability.
97	Sections 9.3.1 (new)	Added “Training Plan” to specify the requirements for developing the Training Plan.
98	Section 9.3, Subsection 9.3.2 (old 9.3.1)	<ul style="list-style-type: none"> • Moved 2nd part of Section 9.3 to 1st part of Subsection 9.3.2, as it was more appropriate to include this requirement in Subsection 9.3.2. • Added “Identification, operation and maintenance of safety-related software products” to comply with IEEE 1228.
99	Subsection 9.3.2 (old Subsection 9.3.1)	<ul style="list-style-type: none"> • Changed subsection heading from “Software Training Manual Program” to “Training Manual and Materials” to be consistent with the subject matter being discussed within this subsection. • Deleted “The timing of the Licensee’s acceptance shall be specified in the contract” as it is outside the scope of this SMPM. • Added reference to require Training Manuals to be reviewed in accordance with the SQAPM. • Deleted “The Training Manual shall be prepared in accordance with the requirements specified in the STngP and the HFE/MMIS IP” as Section 9.0 specified the requirements for Training Manual.
100	Subsection 9.3.3 (old Section 9.4)	Deleted “programs for operations, maintenance, and support of the software products” in 2nd paragraph as this is outside the scope of the HFE Training Development Implementation Plan.
101	Subsection 9.3.3 (old Section 9.4)	Added new paragraphs to clarify when the training program plan needs to be initiated and completed, and specify who prepare and approve training program and schedule to be compliance with IEEE 1074, and may be revised to incorporate lessons learned, feedback and recommendation provided by the students.

Item	Location	Change
102	Subsection 9.3.4 (old Section 9.6)	Revised to be compliance with IEEE 1074. <ul style="list-style-type: none"> • Changed heading from “Training Facilities” to “Training Implementation”. • Added new paragraph to describe TSL roles and responsibilities in the implementation of the training program, including determine the type of training facility.
103	Section 9.4 (old Section 9.5)	Section re-numbered due to re-organization of Sections 9.3 – 9.7 to improve readability and quality of Section 9.0.
104	Section 9.5 (old Section 9.7)	<ul style="list-style-type: none"> • Revised to clarify the examples of training tool metrics. • Bullet 1. Replaced with “End of course student feedback”. • Bullet 4: Changed “during plant scenarios” to “measures for plant scenarios”. • Added new bullets to improve completeness of training evaluation process. • Exam validity and difficulty indexing • Management observation of training • Evaluation of post-training on-the-job performance
105	Section 10.1 (Appendix A)	<ul style="list-style-type: none"> • Items #26 – 30. Deleted “as stated in Section 2.4” as this is redundant. Already described in Section 2.4. • Item 31 - Deleted “as stated in Section 2.4 except for 4.4” as this is redundant. Already described in Section 2.4 and Section 4.1. of SQAPM.
106	Section 10.2 (Appendix B)	<ul style="list-style-type: none"> • Added CIR - Customer Issue Resolution. • Added RM – Responsible Manager • Added CDA – Critical Digital Asset • Added CyST – Cyber Security Team • Added SVTP – Software Validation Test Plan • Deleted acronyms not used in the SMPM and SQAPM.

Item	Location	Change
107	Section 10.3 (Appendix C)	<ul style="list-style-type: none"> • Revised definition for “code” to be consistent with IEEE 610.12. • Revised definition for “Verification and Validation (V&V)” to be consistent with IEEE 610.12. • Changed “Requirements Traceability Analysis” to “Requirements Analysis” to be consistent with IEEE 610.12. • Added definition for “Requirements Traceability Analysis”. • Added definition for “Critical Digital Asset”.
108	Figure 3-1, Table 5.6-1, Section 3.2, Section 9.2	Deleted Figure 3-1, Organizational Functions and Interfaces, and revised the associated table and Subsections to remove obsolescence issues related to this figure to incorporate RAI 7.1-129 comment.
109	Figures 5-2 to 5-12	<p>Revised figures to be consistent with the design process and quality process described in the SMPM and SQAPM, respectively.</p> <p>Figure 5-7 is divided into two (2) figures, Figure 5-7a and Figure 5-7b so the accurate process can be presented.</p>
110	Figure 5-2	Added “V” to Concept documents quality block to be consistent with the SQAPM.
111	Figure 5-8	Added Customer Witness Point prior to SAT to be consistent with the SQAPM.
112	Figure 5-9	Changed O&M phase process step 2 to “Reapply Software Life Cycle Process”.
113	Figure 5-10	Deleted Note 3. This note is not needed as the SQAPM adequately described the V&V tasks requirements.
114	Figure 5-11	Added “RTA” to Software Class N3 and N2 to be consistent with traceability requirements discussed in the SMPM.
115	Figure 5-11, Figure 5-12	<ul style="list-style-type: none"> • Deleting and replaced the referenced R.G and IEEE standards with the appropriate LTR title. • Deleted the superscripts and associated notes.

Item	Location	Change
116	Table 5.6-1, Table 5.7-1, Table 5.8-1, Table 5.9-1, Table 5.10-1, Table 5.11-2, Table 5.11-3, Table 5.11-4, Table 5.12-1	<ul style="list-style-type: none"> • Changed table heading “Verification Organization” to “V&V Organization” to clarify the V&V organizational responsibilities • Corrected references to SQAPM • Added “CySA in accordance with CySPP” to clarify the scope of CySA.
117	Table 5.6-1	<ul style="list-style-type: none"> • Changed “Preliminary Hazard Analysis Report” to “SSA Report” to be consistent with the SQAPM. • Added note to clarify the I/O, function/setpoint data design process.
118	Table 5.10-1	Added SFAT, MFAT, SAT, Cyber Security SFAT, Cyber Security MFAT and Cyber Security SAT Plan, associated test documentation and design organization.
119	Table 5.11-2	<ul style="list-style-type: none"> • Deleted items 1, 2, 3 and 7 as these items are not Installation Phase I output documents. • Added correct output document 10, Software Safety Analysis Report.
120	Table 5.11-3	<ul style="list-style-type: none"> • Deleted items 1 and 3 as these items are not Installation Phase II output documents. • Added correct output document – item 7, Software Safety Analysis Report.
121	Table 5.11-4	<ul style="list-style-type: none"> • Deleted items 1, 2, 3 and 4 as these items are not Installation Phase III output documents. • Added correct output documents – items 9, 10, 11 and 12, I/O, Function/Setpoint data (Final), Installation Configuration Tables, Installation Report and Software Safety Analysis Report.

Item	Location	Change
122	Table 5.12-1	<p>The following changes are made to be consistent with the SQAPM.</p> <ul style="list-style-type: none"> • Changed Item #1 from "ECN(s)" to "Baseline Change Assessment Report" as described in Subsection 5.13.3. • Deleted Items 2-9 will be deleted as these of out of scope. • Changed Item #9 "Revised Baseline Review Records" to "O&M Phase Baseline Review Record". • Changed Table title from "O&M and Retirement Phase Output Documents" to "O&M Phase Output Documents".
123	Table 5.13-1 (New)	Added Table 5.13-1 to describe the output documents to the Retirement Phase.

1. INTRODUCTION

1.1 OVERVIEW

The Software Management Program Manual (SMPM) governs the design and development activities for the Digital Computer-Based Instrumentation and Controls (I&C) Software. Key planning documents for the I&C design team are contained in this manual.

1.2 PURPOSE AND SCOPE

The purpose of the SMPM is to establish the processes and the technical direction for the planning, design, development and management activities of the Digital Computer-Based I&C Software within the scope of the Man-Machine Interface (MMI) System and Human Factor Engineering (HFE) Implementation Plan (MMIS/HFE IP) [2.1].

The scope of the SMPM includes software products with Software Class Q, N3, and N2 (See Appendix C for definitions). Nonsafety-related systems are referenced as Software Class N.

The software plans, identified in the MMIS/HFE IP [2.1], and included in the SMPM are:

- | | |
|---|---------------|
| 1. Software Management Plan (SMP) | [Section 3.0] |
| 2. Software Development Plan (SDP) | [Section 5.0] |
| 3. Software Integration Plan (SIntP) | [Section 6.0] |
| 4. Software Installation Plan (SIP) | [Section 7.0] |
| 5. Software Operation and Maintenance Plan (SOMP) | [Section 8.0] |
| 6. Software Training Plan (STRngP) | [Section 9.0] |

The Software Quality Assurance Program Manual (SQAPM) [2.3(1.a)] includes the software plans used by Quality Assurance (QA) and the Software Project Engineering (SPE) organizations. The SQAPM governs the same I&C software scope identified in the MMIS/HFE IP [2.1]. The software plans included in the SQAPM are:

1. Software Quality Assurance Plan (SQAP)
2. Software Safety Plan (SSP)
3. Software Verification & Validation Plan (SVVP)
4. Software Configuration Management Plan (SCMP)
5. Software Test Plan (STP)

Together, the SMPM and the SQAPM include the software plans identified in MMIS/HFE IP [2.1] and conform to the guidance provided by NUREG-0800, Standard Review Plan [2.2.1].

The SMPM shall be in force during all phases of the software life cycle process.

The applicable Software Products (software and firmware) covered in the SMPM encompass the I&C systems, as specifically defined in the MMIS/HFE IP [2.1] (Subsection 1.2.4 only), perform the monitoring, control, alarming, and protection functions associated with all modes of plant

normal operation (i.e., startup, shutdown, standby, power operation, and refueling) as well as off-normal, emergency, and accident conditions.

1.3 ACRONYMS, ABBREVIATIONS, AND DEFINITIONS

Acronyms and abbreviations are defined in Appendix B. Definitions for terms used in the SMPM are supplied in Appendix C.

2. APPLICABLE DOCUMENTS

Applicable documents identified in this section are supporting documents, supplemental documents, and codes and standards. Supporting documents provide the input requirements to the SMPM. Supplemental documents are used in conjunction with the SMPM. Applicable codes and standards are also identified in the SMPM.

2.1 SUPPORTING DOCUMENTS

The following supporting documents were used as the controlling input documents in the development of the SMPM. These documents form the design basis for the activities stated in the SMPM. In the event of any differences noted between the SMPM and the ESBWR Composite Design Specification [2.1], the SMPM governs.

- ESBWR Man-Machine Interface System and HFE Implementation Plan (MMIS/HFE IP), NEDO-33217
- ESBWR Composite Design Specification (A11-5299), 26A6007
- ESBWR Composite Design Specification Standard Review Plans and Regulatory Guides (A11-5299), 26A6007AB
- ESBWR Composite Design Specification Industry Codes and Standards (A11-5299), 26A6007AC
- ESBWR DCD, Chapter 7, I&C Systems, 26A6642AW
- ESBWR DCD, Chapter 15, Safety Analysis, 26A6642BP

2.2 REGULATORY DOCUMENTS, CODES AND STANDARDS

The following documents are applicable to the activities specified within the SMPM. The SMPM conforms to planning requirements of these documents except as explicitly noted in Appendix A.

2.2.1 NUREG

- NUREG-0800, Standard Review Plan (SRP), Chapter 7, Branch Technical Position (BTP) HICB-14 R4, Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems

2.2.2 Code of Federal Regulations (CFR)

- 10 CFR 50, Appendix B, Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants

2.2.3 U.S. Nuclear Regulatory Commission (NRC) Regulatory Guides (RG)

- RG 1.152-2006 - Criteria for Use of Computers in Safety Systems of Nuclear Power Plants
- RG 1.168-2004 - Verification, Validation, Reviews, and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants

- RG 1.169-1997 - Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
- RG-1.170-1997- Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
- RG-1.171-1997 - Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
- RG 1.172-1997 - Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
- RG 1.173-1997 - Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants

2.2.4 Institute of Electrical and Electronic Engineers (IEEE) Standards

The following standards are applicable to the activities specified within the SMPM. The SMPM conforms to planning requirements of these standards except as explicitly noted in Appendix A.

The IEEE Standards provide recommended implementation techniques and methods. The SMPM makes specific commitments only to those requirements restated in this document. The Project Work Plan shall capture the detailed implementation attributes in accordance with Work Planning and Scheduling [2.3(2.a)]. Future exceptions or deviations from the recommendations specified in the IEEE standards shall require management approval as defined in the SQAPM [2.3(1.a)] and the SMPM, and are potentially subject to NRC notification in accordance with the MMIS/HFE IP [2.1].

- IEEE 7-4.3.2-2003 IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations
- IEEE 603-1991 and correction sheet dated January 30, 1995 - IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations. This IEEE standard is applicable to the design of safety-related instrumentation and control systems of which software is a part.
- IEEE 828-1990 IEEE Standard for Software Configuration Management Plans
- IEEE-829-1983 IEEE Standard for Software Test Documentation
- IEEE-830-1993 IEEE Recommended Practice for Software Requirements Specifications
- IEEE-1008-1987 IEEE Standard for Software Unit Testing
- IEEE 1012-1998 IEEE Standard for Software Verification and Validation
- IEEE 1028-1997 IEEE Standard for Software Reviews Description
- IEEE-1042-1987 IEEE Guide to Software Configuration Management Description
- IEEE-1074-1995 IEEE Standard for Developing Software Life Cycle Processes

2.3 SUPPLEMENTAL DOCUMENTS

The following supplemental documents are used in conjunction with the SMPM and enable the performance of the activities stated in Appendix A.

Reference Number	Document
1.a	GE Hitachi Nuclear Energy, "ESBWR - Software Quality Assurance Program Manual," NEDE-33245P, Class III (Proprietary), and NEDO-33245, Class I (Non-Proprietary)
1.b	GE Hitachi Nuclear Energy, "ESBWR Cyber Security Program Plan," NEDE-33295P, Class III (Proprietary), and NEDO-33295, Class I (Non-Proprietary)
1.c	GE Hitachi Nuclear Energy, "ESBWR HFE Training Development Implementation Plan," NEDO-33275, Class I (Non-Proprietary)

GE Hitachi Nuclear Energy Procedures and Policies		
Reference Number	Document Title	Abstract
2.a	Work Planning and Scheduling	Defines the process and responsibilities for developing and documenting work plans and schedules for customer-contracted design work and authorized projects. Four key purposes of a Project Work Plan are to define project scope, develop a schedule, monitor progress, and control resources.
2.b	(Deleted)	
2.c	Design Review	Defines responsibilities and procedural requirements for conducting formal design adequacy evaluations. Design Reviews are used to verify that product designs meet customer, functional, contractual, safety, health, environmental, regulatory, industry codes and standards, and corporate requirements.
2.d	Design Record File	Defines the process for the generation of a Design Record File, which is a formal, controlled information record for in-progress and completed engineering work.

GE Hitachi Nuclear Energy Procedures and Policies		
Reference Number	Document Title	Abstract
2.e	Material Request	Details responsibilities and procedural requirements for the release of technical, engineering, customer, and quality requirements that define material, equipment, labor, services and related data to meet GE Hitachi Nuclear Energy (GEH) contract/purchase order, code, and regulatory requirements.
2.f	Independent Design Verification	Details roles and responsibilities for reviewing and substantiating a design to provide independent and documented confirmation that the design meets specified requirements.
2.g	Deferred Design Verification	Defines the process for deferring design verification and for clearing previous deferrals. The process applies to cases where a design, or portion of a design, must be released prior to completion of verification.
2.h	Document Initiation or Change by Engineering Review Memorandum/Engineering Change Notice	Establishes the requirements for the initiation of, or change to, engineering controlled documents by use of the Engineering Review Memorandum/Engineering Change Notice. The process assures traceability, configuration, and quality assurance of engineering documents that are maintained through the current document revision, status, and final disposition.
2.i	Procurement Initiation and Control	Specifies the requirements for procurement of material, equipment, and services, including the application of technical, engineering, customer, and quality requirements on purchase orders. Defines the requirements for establishing and maintaining the Approved Suppliers List.

GE Hitachi Nuclear Energy Procedures and Policies		
Reference Number	Document Title	Abstract
2.j	Field Deviation Disposition Request	Establishes a process to document and disposition the technical position for field deviations to GEH supplied hardware, software, or services. Responsible individuals evaluate Field Deviation Disposition Requests to assure that the proposed field action meets safety, technical, quality, application and commercial requirements.
2.k	Safety-Related Classification	Defines the requirements used to identify structures, systems, components, parts, and technical services that are safety-related. Safety-related structures, systems, components, and parts provide safety-related functions necessary to assure: a. The integrity of the reactor coolant pressure boundary; or b. The capability to shut down the reactor and maintain it in a safe shutdown condition; or c. The capability to prevent or mitigate the consequences of accidents that could result in potential off site exposures comparable to 10CFR50.34(a)(1) or 10CFR100.11 guideline exposures, as applicable.
2.l	Operation and Maintenance Instruction Manuals	Defines requirements applicable to the preparation, review, and approval of Operation and Maintenance instruction manuals.
2.m	Corrective Action Process	Specifies the responsibilities for actions to promptly identify, record and correct Conditions Adverse to Quality to assure that these conditions do not affect the quality of products or services. Defines the requirements and responsibilities for conducting ongoing self-assessments, focused self-assessments, and internal audits of organizations within GEH.

GE Hitachi Nuclear Energy Procedures and Policies		
Reference Number	Document Title	Abstract
2.n	Quality and Technical Training	<p>Defines the roles and responsibilities to assure personnel proficiency in quality and technical related activities. The Quality and Technical Training program:</p> <ul style="list-style-type: none"> a. Assures personnel are trained and proficient in assigned quality and technical tasks. b. Documents qualifications for technical positions, including minimum education, experience, and any special training requirements. c. Records training assignments in a centralized controlled training database.
3.a	Work Authorization	<p>Establishes the requirements and responsibilities within GEH for the preparation and approval of Work Authorizations that communicate requirements to functional components of GEH or Global Nuclear Fuel (GNF).</p>
3.b	Project Risk Management Procedure	<p>Implements the project risk management requirements of GEH Policy. Provides a controlled process for risk management to maintain positive control of work situations, especially during critical tasks or activities.</p>
3.c	Project Management Policy	<p>Provides requirements for the single Project Management process across all GEH. The process components include project initiation, planning, scheduling, execution, controls, and post-delivery closeout.</p>
3.d	Project Financial Management	<p>Establishes specific requirements and describes typical methods that are used to assure project financial management activities are accomplished in compliance with GEH policies.</p>
3.e	Quality Policy and Quality System Requirements	<p>Establishes the requirements of the GEH business quality system. Defines requirements necessary to implement the quality policy and to demonstrate, by performance both inside and outside GEH, total dedication to the attainment of quality leadership and customer satisfaction.</p>

GE Hitachi Nuclear Energy Procedures and Policies		
Reference Number	Document Title	Abstract
3.f	Nuclear Energy Quality Assurance Audit Requirements	Establishes the requirements and processes for a comprehensive audit program to verify the implementation and effectiveness of the GEH Quality System. The audit program requirements apply to hardware, software and service products and to all personnel who perform quality-related activities on them.
3.g	Reporting of Defects and Noncompliance Under 10CFR Part 21	Defines the requirements and responsibilities within GEH for ensuring compliance with the requirements of Part 21 of Title 10 of the Code of Federal Regulations, i.e., 10CFR21, "Reporting of Defects and Noncompliance".
3.h	Hazardous Business Risk and Safety in GEH Services and Products	Establishes the organizational responsibilities and systems within GEH to ensure that services and products are evaluated and controlled for hazardous business risks, safety, and environmental effects
3.i	Hazardous Business Risk Evaluations and Control	Defines the responsibilities and practices for evaluation of new GEH activities (products, services, projects, or processes) or changes to existing activities, review of proposed commercial applications, and implementation of risk mitigation and controls.
3.j	Nuclear Customer Issue Resolution – Nuclear CIR Tool	Nuclear CIR Tool is a customer issue resolution process and tool that is used to identify, track and respond to any customer questions, complaints, actions, or issues unless it relates to matter that is adverse to quality (requiring a corrective/preventive action) hence which is formally tracked in another system.

Reference Number	Document
(Deleted)	
(Deleted)	
(Deleted)	

4.	Electric Power Research Institute (EPRI), "Guidelines on Evaluation and Acceptance of Commercial Grade Digital Equipment in Nuclear Safety Applications," EPRI TR-106439
(Deleted)	

2.4 ADDITIONAL IEEE STANDARD GUIDANCE

The following IEEE Standards provide additional guidance for the implementation activities. Conformance of the SMPM to these activities has been evaluated. Selected sections/topics from these IEEE Standards are excluded from commitment because they either provide conflicting requirements with other Standards or the level of detail is not appropriate for the SMPM. Clarifications and justifications for such exclusions are provided in Appendix A.

- IEEE-730-2002 - IEEE Standard for Software Quality Assurance Plans
- IEEE-610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology
- IEEE-1016-1998 - IEEE Recommended Practice for Software Design Descriptions
- IEEE-1058.1-1987 - IEEE Standard for Software Project Management Plans
- IEEE 1219-1998 - IEEE Standard for Software Maintenance
- IEEE 1228-1994 - IEEE Standard for Software Safety Plans
- IEEE 12207-1996 - IEEE/EIA Standard for Software Life Cycle Processes

2.5 INTERNATIONAL STANDARDS

- ISO 9001:2000, Quality Management Systems-Requirements

3. SOFTWARE MANAGEMENT PLAN

3.1 PURPOSE AND SCOPE

The Software Management Plan (SMP) establishes the managerial process for the design and development activities of the Digital Computer-Based I&C Software within the scope of the MMIS/HFE IP [2.1]. The purposes of the SMP are to:

- Establish project management activities, which include but are not limited to the following activities:
 - Project planning and scheduling
 - Project monitoring and control
 - Project execution
 - Post delivery and closeout
- Define the organization and responsibilities of individuals or groups involved in the various design and V&V activities
- Define risks management process
- Establish the methods and tools for project management
- Define financial (budget) responsibilities and controls
- Define security (including cyber security) requirements
- Define training requirements and qualification of project personnel

3.2 ORGANIZATION

The organization addresses software management control and ensures that independence is maintained between the design organization and the quality assurance, software safety, and V&V organizations.

This section describes the following organization functions:

- I&C Design Engineering (I&C)
- Software Project Engineering (SPE)
- Configuration Management Manager (CMM)
- Software Quality Assurance (SQA) Manager
- Project Management Team (PMT), i.e. Project Control
- Training

3.2.1 I&C Design Engineering

The I&C software development organization comprises the GEH I&C Design Engineering, representation from Cyber Security organization and the GEH and non-GEH software products vendor organization. The GEH I&C Design Engineering (I&C) Organization comprises the I&C Design Engineering Manager (I&C Manager), the platform Technical Project Engineers (TPEs),

and the Responsible I&C Engineers. This organization implements the activities defined in the SMPM.

The I&C Manager is responsible for overall performance and schedule of the software development effort, including work flow to the system TPEs, system engineers, and software products vendors. The platform TPEs are responsible for day-to-day management, coordination, and scheduling of the system design and software development effort. They are responsible for interfacing with the system engineers and software product vendors. The platform TPEs are also responsible for providing status reports to the I&C Manager.

The I&C Engineer is responsible for the design and development of the software products. The I&C Engineer is responsible for reviewing and confirming that the design documentation and outputs produced by the software products vendors meet the technical requirements specified in the contract/purchase order.

The vendors may be internal or external to GEH and shall be organized such that a single Point of Contact (POC) is assigned the responsibility of interfacing with the TPE. Alternative POCs shall be assigned to take over the duties when the Primary POC is unavailable. The Primary POC and alternative POCs shall be determined by the hardware/software vendor organization and may be any individual within the organization who is qualified to act as the organization's agent. Software developed by the vendors shall be in accordance with the SMPM and the SQAPM [2.3(1.a)].

3.2.2 Software Project Engineering

Software Project Engineering (SPE) is independent of the I&C organization to ensure organizational freedom to perform quality tasks without undue pressure or conflict of interest related to budget or schedule.

The SPE organization is responsible for executing the quality tasks as described in the SQAPM [2.3(1.a)] and comprises the following teams as described in SQAPM [2.3(1.a)], Subsection 3.2.3.4:

- Independent Verification and Validation Team (IVVT)
- Software Safety Analysis Team (SST)
- Baseline Review Team (BRT)

3.2.3 Configuration Management Manager

The Configuration Management Manager (CMM) has the overall responsibility for the Configuration Management System (CMS), herein referred to as Product Data Management System (PDMS). The CMM responsibilities are addressed in the SQAPM [2.3(1.a) Section 6.0].

3.2.4 Software Quality Assurance Manager

The Software Quality Assurance (SQA) Manager interfaces with the SPE Manager and has the overall responsibility and authority for the SQA program. The SQA Manager responsibilities are addressed in the SQAPM [2.3(1.a)] Subsection 3.2.3.1.

3.2.5 Project Management Team

The technical management of software products is the responsibility of the TPEs. The Project Management Team (PMT) is responsible for the commercial aspects of the project. A commercial Project Manager (PM) shall be assigned to oversee each of the projects, and shall be responsible for delivering the commitments of a purchase order or contract to the Licensee.

The PMT performs the following activities:

- Project work planning
- Development and maintenance of the integrated project schedule - TPEs shall provide task inputs and support for this activity
- Update of the integrated schedule to show that project tasks are completely and accurately reflected
- Assignment of project resources and skill sets to support the project needs
- Preparation of project progress reports
- Project risk management assessment
- Project budgeting
- Engineering procurement and construction
- Communication with Licensee and vendors

3.2.6 Training

See Section 3.11 of the SMPM for training requirements.

3.2.7 Cyber Security Team

The Cyber Security Team (CyST) is responsible for ensuring cyber security of the design, development and evaluation of the software products throughout the Software Life Cycle phases.

The roles and responsibilities of the CyST are defined in the CySPP [2.3(1.b)].

3.3 ORGANIZATIONAL BOUNDARIES AND INTERFACES

The SMPM and SQAPM specify the organizational structures for the I&C and Electrical Systems Engineering and SPE. This includes boundaries and relationships with the external and internal organizations. The PM provides the Licensee and vendor an interface with I&C/ and SPE organizations.

[[

]]

3.4 ORGANIZATIONAL RESPONSIBILITIES

Organizational responsibilities are defined in the following subsections:

3.4.1 (Deleted)

3.4.2 (Deleted)

3.4.3 I&C Design Engineering Manager

The I&C Manager is responsible for directing the engineering work of the I&C Design Engineering organization. The functional leads for various I&C functions report to the I&C Design Engineering Manager.

3.4.4 Software Project Engineering Manager

The Software Project Engineering (SPE) Manager is responsible for the software quality tasks during the design and development of the software product.

3.4.5 Software Quality Assurance Manager

The Software Quality Assurance (SQA) Manager, who interfaces with the SPE Manger, has the overall responsibility and authority of the SQA Program. The SQA Manager reports to the Quality General Manager.

3.4.6 Training Services Lead

The Training Services Lead (TSL) is responsible for organizing the overall training process, including scheduling, budgeting, and resource allocation. The TSL reports to the Plant Performance and Optimization Manager.

3.4.7 Configuration Management Manager

The Configuration Management Manager (CMM) is responsible for the configuration management of the ESBWR project, including software products.

3.4.8 Technical Project Engineer

The Technical Project Engineer (TPE) has technical responsibility for the software tasks related to software or a group of software products. The TPEs report to the I&C Manager.

3.5 SOFTWARE MANAGEMENT PLAN CHANGE CONTROL PROCESS

The SMPM is applicable for the entire Software Life Cycle of the software product. It is anticipated that the software development cycle shall evolve as software development technology changes. It is acceptable to revise the SMPM to improve quality. The change control process is described in the SQAPM [2.3(1.a)].

The SMPM is a controlled document under configuration control in accordance with the SQAPM [2.3(1.a) Section 6.0].

[[

]]

If a change to the SMPM is warranted, the SPE Manager shall determine if NRC notification is required and shall track the notification process as defined by the MMIS/HFE IP [2.1].

Changes to the SMPM require approvals of the following managers or designated appointees: I&C Manager, SPE Manager, and the SQA Manager.

[[

]]

If changes to the SMPM are made, the I&C Manager shall conduct and document an evaluation indicating that previously completed projects do not have to be reopened to implement the SMPM changes. When changes are made to the SMPM, requirements traceability will be maintained and verified.

3.6 PROJECT MANAGEMENT PRIORITIES, MONITORING, AND CONTROL

The objective of project management is to coordinate the development of project deliverables and to ensure that the deliverables meet the Licensee expectations for nuclear safety, quality, cost, and schedule. The key elements for a successful project delivery by project management are:

- Integrity - All aspects of the project ~~is~~ are performed and practiced with integrity at all times
- Quality - All aspects of the project comply with the software development and quality assurance process defined in the SMPM, the SQAPM [2.3(1.a)], and the applicable industry codes and standards
- Occupational Safety - Safe work habits are practiced at all times
- Outputs - Deliverables meet the quality, schedule, and budget requirements as specified by the project work plans.

[[

]]

The key management processes are listed below and described in the following subsections:

- Project Initiation
- Project Planning and Scheduling
- Project Execution
- Project Controls
- Post-Delivery Closeout

3.6.1 Project Initiation

Project initiation begins after a contract has been awarded or an internal project is authorized. A preliminary schedule is developed that considers project resource availability, and is consistent with the approved project work scope and budget.

3.6.2 Project Planning and Scheduling

[[

]]

The PWP identifies the work scope, design inputs and outputs, deliverables, and QA requirements, as described in the SQAPM [2.3(1.a)]. Timing for these activities shall be consistent with the integrated project schedule.

[[

]]

3.6.3 Project Execution

Project Execution completes the work defined in the PWP by performing the following activities:

- Developing the Work Breakdown Structure (WBS) for the project.
- Initiating material requisitions for services and materials
- Conducting project kickoff meetings with the interfacing organizations (e.g., Licensee and vendors). The frequency of the project meeting is determined by the PM and shall be conducted with the internal organization (e.g., I&C and SPE) and the external organization (e.g., Licensee or vendor).
- Conducting Baseline Review (BR) for each software product or logical group of software products at the end of each Software Life Cycle phase as described in the SQAPM [2.3(1.a)].
- Preparing phase Baseline Review Reports in accordance with the SQAPM [2.3(1.a)].
- Monitoring software product development progress
- Identifying critical path items/activities
- Identifying tools
- Identifying Software Safety Analysis (SSA), Independent Verification and Validation (IV&V), and configuration control activities
- Establishing/reviewing milestone dates for these items/activities
- Identifying and adjusting manpower and resource levels
- Identifying internal and vendor performance problems as early as possible
- Performing Risk Assessment and Risk Management (See Section 3.9)

3.6.4 Project Controls

Project Controls activities include measurement and monitoring of project execution. The metrics that are integral to the Workforce Planning and Scheduling Tools are applied by the PM so that corrective action can be taken when necessary to adjust for schedule delays, unexpected

changes in work scope, or quality issues stemming from design team and vendor performance challenges.

[[

]]

Project performance is monitored using computer-based tools and project reviews. The project review is used to assess the execution of the project.

3.6.4.1 Frequency of Project Review

The frequency of project reviews is commensurate with the complexity of the project. The frequency of project reviews is specified in the PWP.

3.6.4.2 Progress Reports

The PM prepares the progress reports that detail progress and status on a regular basis. The frequency of the progress report is specified in the contract and in the PWP.

3.6.5 Post-Delivery Closeout

Post-Delivery Closeout finalizes the product and completes the delivery in accordance with the contract. Activities include:

- The closure of project paperwork, including Design Record Files (DRFs)
- Closeout of vendor activities, including vendors submittal of required documentation to GEH
- Turnover of the project to the Licensee, including transfer of SQA activities to the Licensee

The following shall also be performed during Post-Delivery Closeout.

3.6.5.1 Project Deliverables

The project deliverables include a combination of hardware, software, design documents, and supporting documents such as test and analysis reports. The project deliverables are identified in the Licensee contracts.

3.6.5.2 Software Developed by Vendors

A software product developed by vendors shall conform to the requirements outlined in the SMPM and SQAPM [2.3(1.a)].

[[

]]

Additionally, Class Q software is approved by the I&C Manager and the SPE Manager and audited by the SQA team.

3.7 METHODS AND TOOLS FOR PROJECT MANAGEMENT

3.7.1 Methods

See Sections 3.6 Project Management Priorities, Monitoring, and Control and Section 3.9 Risk Management.

3.7.2 Tools

The Project Manager shall indicate in the PWP the approved tools required for efficient performance of the project. To ensure efficient and effective execution of the software, the GEH Project Team shall be provided with the tools for project management such as:

- Desktop computers or notebooks/laptops
- [[]]
- High speed printers, copiers, and scanners
- Software programs such as Microsoft® Word, Excel, Outlook and Adobe® Acrobat that are widely and commonly used to ensure efficient communication with Licensee and vendors

Workforce planning and scheduling tool which allows:

- The PMs to plan activities, and develop and maintain project schedules to track project progress
- The assignment of resources to ensure that resource requirements can be met by available resources with appropriate resource skill-sets to support the project
- The ability to ensure the resource requirements can be met with appropriate resource skill-sets to support the project

Product Data Management System is:

- A computer-based data system that stores, retrieves, and reports data relevant to the engineering definition of products and services offered and provided to the Licensee
 - The official configuration control system for engineering controlled documents
- [[]]

3.8 BUDGET

[[]]

The PMT is independent of the engineering teams responsible for the design and quality assurance work on the project. The specific budgetary activities are:

- Accurately allocate resources to each project organization including vendors internal and external to the project organization.
- Assign resources to each project organization to maintain financial independence from each other.

The PM is responsible for generating the project task charge numbers based on the identified and scheduled activities so that expenditures can be monitored at the task level. Unique charge numbers are generated for each activity or for a set of similar activities. For example, the charge numbers assigned for software implementation work are different from IV&V and SSA activities.

Expenses incurred by the project shall be charged to appropriate charge numbers. The expenses include, but are not limited to, labor (including GEH employees and contractors), travel and living expenses, external contract labor, and purchased material.

A quarterly financial review shall be conducted with the New Plant Project (NPP) General Manager to ensure that the costs incurred are consistent with the approved budgets. If the project estimate at completion is different from the approved budgeted cost, the project cost budgets shall be adjusted to match the cost at completion. In order to achieve the correct cost budget for the project, it is imperative that the costs and commitments are considered when analyzing project costs. This may result in a possible adjustment to the current estimate at completion to support the project commitments, especially those related to safety and quality of the software products.

3.9 RISK MANAGEMENT

Risk Management is the process of identifying, controlling, and eliminating or minimizing unpredictable events that may affect the project.

Risk Management shall be implemented in accordance with Project Risk Management Procedure [2.3(3.b)], Hazardous Business Risk and Safety in GEH Services and Products [2.3(3.h)], and Hazardous Business Risk Evaluations and Control [2.3(3.i)].

The Task Leads shall prepare a risk management plan to document responsibilities and actions needed to assess, abate, monitor, and control the identified risks and concerns. It is acceptable to include the risk management plan in the task-specific PWP.

3.10 SECURITY

Planning and testing to ensure compliance with regulatory cyber security requirements is an integral part of the software development. Ensuring an adequate Cyber Security Program requires constant changes to ensure protection from new or emerging threats.

To accommodate the need for frequent enhancements to the Cyber Security Program, cyber security requirements shall be integrated into the Software Life Cycle through implementing procedures. The design and development of software products shall be performed in accordance with Regulatory Guide 1.152 [2.2.3] and ESBWR Cyber Security Program Plan (CySPP) [2.3(1.b)].

3.11 TRAINING AND QUALIFICATIONS

[[

]]

The Engineering and Project training are performed either by classroom or individual study. The SMPM, SQAPM [2.3(1.a)], and applicable tools are needed to support the design work.

[[

]]

In addition, project requirements mandate that personnel receive training on processes, procedures, and tools as required to support the specific project. The use of such tools shall be documented in the PWP.

[[

]]

4. MANAGEMENT PROCESS

Section 4.0 from Rev. 2 was deleted. The Management Process is now discussed in Section 3.0 | as part of the overall Software Management Plan.

5. SOFTWARE DEVELOPMENT PLAN

5.1 INTRODUCTION

The Software Development Plan (SDP) describes the plan for technical project development of the I&C software that performs the monitoring, control, and protection functions for all modes of plant operation.

5.2 PURPOSE AND SCOPE

The SDP describes the software engineering development process for each phase of the Software Life Cycle process. The phases include Planning, Requirements, Design, Implementation, Test, Installation, Operations & Maintenance (O&M), and Retirement. The SDP also addresses the preparation, execution, and documentation of software testing for the software products. The SDP conforms to RG 1.173 [2.2.3] and IEEE 1074 [2.2.4], except as specified in Appendix A.

The purpose of the SDP is to:

- Establish the standards, methods, tools, and procedures for the software design and development process.
- Define the activities performed for each phase of the software development.
- Define how requirements are traced to lower levels of the engineering phases from Planning Phase to Test Phase.
- Specify how the safety-related requirements are documented, evaluated, reviewed, verified, and tested during the design process to minimize unknown, unreliable, and abnormal conditions.
- Describe the organization and responsibilities of individuals or groups involved in the various V&V and review activities.
- Provide a structure for test and review guidance for software functional testing.
- Provide the requirements and guidelines necessary to prepare, execute, and document software tests.
- Address software test documentation.
- Address metrics that include error tracking, cyber security tracking, and resolution.

5.3 ORGANIZATION OF SOFTWARE LIFE CYCLE PROCESS

The software development process follows phase changes in a Software Life Cycle model.

[[

]]

The Software Life Cycle is not based on a pure waterfall model; instead it uses a modified waterfall model that includes the provisions for task iteration. The Software Life Cycle phases

defined in the SMPM conform to and are based on RG 1.152 [2.2.3], RG 1.173 [2.2.3], and IEEE 1074 [2.2.4]. The Software Life Cycle phases are described as follows:

Planning Phase - The definition of the project scope, methodologies, and resources to develop and maintain the deliverable software are determined. The planning activities include evaluation of system and Licensee requirements, identification of resources, and development of schedule projections and risk assessments. The Planning Phase Baseline Review Record (BRR) documents successful completion of this phase.

Requirements Phase - The definition of the detailed functional and performance requirements, security requirements, design constraints, and test criteria are determined. The Requirements Phase BRR documents successful completion of this phase.

Design Phase - Requirements are transformed into architecture and a detailed representation of software. The Design Phase BRR documents successful completion of this phase.

Implementation Phase - The software design is transformed into software source or application codes that include secure coding practices. The Implementation Phase activities include software code review and software functional tests. Utilizing a structured test approach, a software functional test is conducted to validate the software source or application codes. Software-software and software-hardware integration is performed during software functional testing. Typically, prototype hardware is used at this time. The Implementation Phase BRR documents successful completion of this phase.

Test Phase - The software validation testing occurs which tests for potential defects (errors) and verifies security requirements. The results are documented in the Software Validation Test Report. The Test Phase BRR documents successful completion of this phase.

Installation Phase - Installation is performed as a three-phase process, starting with software level installation, then system level installation and concluding with on-site installation.

Operations & Maintenance Phase - This phase involves the operational life of the software product(s). It includes the operation, maintenance, calibration, surveillance, cyber security assessment in accordance with CySPP [2.3(1.b)], and other processes associated with the use of the system. Application of the processes is based on data, documentation, and procedures provided with each system in the O&M manual. The Maintenance section of the O&M manual includes procedures to maintain and resolve operational anomalies.

[[
]]

Retirement Phase - The effect of replacing or removing the existing software product from the operating environment shall be addressed.

The activities include:

- User notification
- Effect on existing software products that are to remain operational in the operating environment
- Disposition of the retired software product including security disposition. This includes:

- Deactivation
- Deletion or the removal of the software product from the operating environment
- Operational comparison of the new and old software products
- Documentation activities, including archiving of records

5.4 METHODS

The following methods are used to support the design and development of the software product.

5.4.1 Configuration Management and Change Control

[[

]]

Discrepancies or deficient conditions detected in a CI shall be resolved by the Design Team in accordance with the Change Control process described in the SQAPM [2.3(1.a) Section 6.0].

5.4.2 Verification and Validation

[[

]]

5.4.3 Testing

Testing is conducted to ensure the correctness of constructed code and completeness of requirements specified in the Requirements Phase and Design Phase documents.

[[

]]

5.4.4 Software Safety Analysis

Software Safety Analysis (SSA) shall be performed to ensure the safety of Class Q and N3 software. Safety is the most important consideration for the safety-related I&C and takes precedence over budget and schedule.

[[

]]

5.4.5 Baseline Review

Baseline review shall be performed at the completion of each Software Life Cycle phase in accordance with the SQAPM [[

]]. Non-conformances identified during the baseline review shall be resolved by the Design Team.

5.4.6 Deferred Design Verification

Conditional release of a design document may be permissible in cases where a design, or portion(s) of a design, must be released prior to completion of independent verification. Independent verification is conducted in accordance with the SQAPM [2.3(1.a), Section 5.0].

[[

]]

5.4.7 Cyber Security Assessment

A Cyber Security Assessment (CySA), as defined in CySPP [2.3(1.b)], shall be performed to ensure the security of Critical Digital Assets (CDAs). Description, identification and evaluation of CDAs are described in CySPP [2.3(1.b)].

[[

]]

5.5 TOOLS

Specific tools that are required for the project, which may include, but are not limited to, materials, prototypes, hardware, simulators, emulators, and support software shall be documented.

[[

]]

5.5.1 Support Software

Support software is a tool used to aid the development of the software product throughout the software development process.

[[

]]

5.5.2 Requirements Traceability Matrix

A Requirements Traceability Matrix shall be prepared for both Software Class Q and Software Class N design outputs.

[[

]]

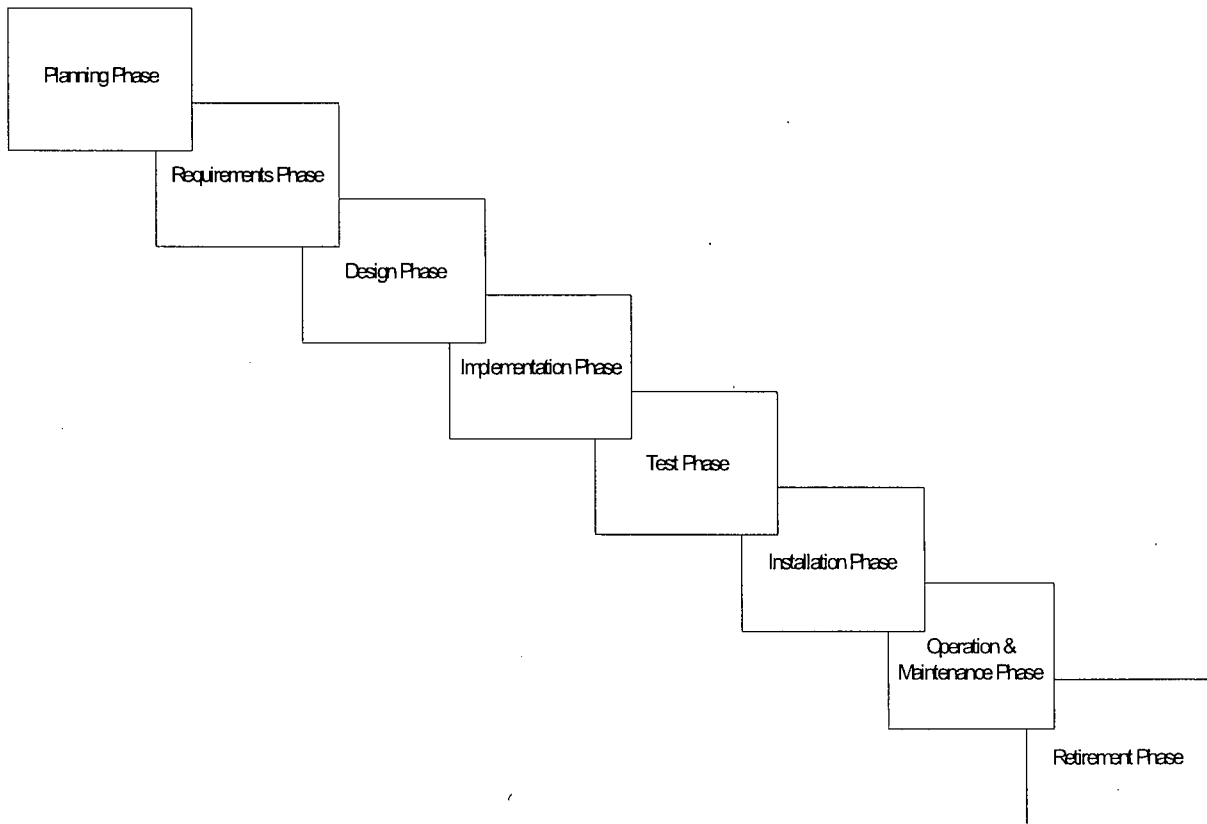


Figure 5-1. Software Life Cycle Process Overview

[[

]]

Figure 5-2. Software Life Cycle Process-Planning Phase

[[

]]

Figure 5-3. Software Life Cycle Process-Requirements Phase

[[

]]

Figure 5-4. Software Life Cycle Process-Design Phase

[[

|

]]

Figure 5-5. Software Life Cycle Process-Implementation Phase

|

[[

]]

Figure 5-6. Software Life Cycle Process-Test Phase

[[]]

Figure 5-7a. Test Software Life Cycle Process-Installation Phase (Software Installation)

[[

]]

Figure 5-7b. Software Life Cycle Process-Installation Phase (System Installation)

[[

]]

Figure 5-8. Software Life Cycle Process-Installation Phase (Site Installation)

]]

]]

Figure 5-9. Software Life Cycle Process-Operations and Maintenance Phase and Retirement Phase

[[

]]

Figure 5-10. Software Life Cycle Process Notes

[[

]]

Figure 5-11. Hardware and Software Design Overview

[[

]]

Figure 5-12. Cyber Security Interaction Model with SMPM and SQAPM Activities

5.6 PLANNING PHASE

[[

]]

5.6.1 Planning Phase Inputs

[[

]]

5.6.2 Planning Phase Outputs

[[

]]

Table 5.6-1 Planning Phase Output Documents

II			

5.6.3 Software Safety Analysis Report

[[

]]

5.6.4 Cyber Security Assessment Report

[[

]]

5.6.5 Planning Phase Baseline Review Record

[[

]]

5.7 REQUIREMENTS PHASE

[[

]]

5.7.1 Requirements Phase Inputs

[[

]]

5.7.2 Requirement Phase Outputs

[[

]]

Table 5.7-1 Requirements Phase Output Documents

[[
]]

5.7.3 Requirements Phase Activities

[[

]]

5.7.4 Hardware/Software Specification

[[

|
|
|

]]

5.7.5 Software Requirements Specification

[[

|

|

|

|

]]

5.7.6 System Requirements Specification

[[

]]

5.7.7 Data Communications Protocol

[[

]]

5.7.8 User Interface Specification

[[

]]

5.7.9 Software Support Tools/Documentation for Software Development

[[



]]

5.7.10 Software Safety Analysis Report

[[

]]

5.7.11 Cyber Security Assessment Report

[[

]]

5.7.12 Requirements Phase Baseline Review Record

[[

]]

5.8 DESIGN PHASE

[[

]]

5.8.1 Design Phase Inputs

[[

]]

5.8.2 Design Phase Outputs

[[

]]

Table 5.8-1 Design Phase Output Documents

[[
]]

5.8.3 Design Phase Activities

5.8.3.1 *Software Design Description*

[[

]]

5.8.3.2 *Intra-System Communication Protocol Specification*

[[

]]

5.8.3.3 *Software Coding Conventions and Guidelines Document*

[[

]]

5.8.3.4 *Software Support Tool Documentation Package*

The Software developer shall evaluate the use of software tools in the new design and document their intended use to ensure they are consistent with the SRS and SDD.

5.8.3.5 *Application of Previously Developed Software*

[[

]]

5.8.3.6 Commercial Off-The-Shelf Software

[[

]]

5.8.3.7 Software Validation Test Documentation Development

[[

]]

5.8.3.8 Site Acceptance Test Documentation Development

[[

]]

5.8.3.9 Multi-System Factory Acceptance Test Documentation Development

[[

]]

5.8.3.10 System Factory Acceptance Test Documentation Development

[[

]]

5.8.3.11 Software Safety Analysis Report

[[

]]

5.8.3.12 Cyber Security Assessment Report

[[

]]

5.8.3.13 Design Phase Baseline Review Record

[[

]]

5.9 IMPLEMENTATION PHASE

[[

]]

5.9.1 Implementation Phase Inputs

[[

]]

5.9.2 Implementation Phase Outputs

[[

]]

Table 5.9-1 Implementation Phase Output Documents

[[
]]

5.9.3 Implementation Phase Activities

5.9.3.1 Software Coding Readiness Review

[[

]]

5.9.3.2 Software Coding

[[

]]

5.9.3.3 Code Review

[[

]]

5.9.3.4 Software Functional Test

[[

]]

5.9.3.5 Software Functional Test Report

[[

]]

5.9.3.6 Software Build Description

[[

]]

5.9.3.7 Software Validation Test Documentation

The Software Validation Test Plan, Software Validation Test Procedures and Test Cases Specification are defined in Subsection 6.12.3, Software Validation Test Documentation.

5.9.3.8 Software Safety Analysis Report

[[

]]

5.9.3.9 Cyber Security Assessment Report

[[

]]

5.9.3.10 Implementation Phase Baseline Review Record

[[

]]

5.10 TEST PHASE

[[

]]

5.10.1 Test Phase Inputs

[[

]]

5.10.2 Test Phase Outputs

[[

]]

Table 5.10-1 Test Phase Output Documents

[[
]]

5.10.3 Software Validation Test

[[

]]

5.10.4 Software Validation Test Report

[[

]]

5.10.5 Production Release

[[

]]

5.10.6 Software Release Notes

[[

]]

5.10.7 Cyber Security Assessment Report

[[

]]

5.10.8 (Deleted)

5.10.9 Test Phase Baseline Review Record

[[

]]

5.11 INSTALLATION PHASE

[[

]]

5.11.1 Installation Phase Inputs

[[

]]

Table 5.11-1 Installation Phase Input Documents

]]

5.11.2 Installation Phase Outputs

[[

]]

Table 5.11-2 Installation Phase I Output Documents

[[
]]

Table 5.11-3 Installation Phase II Output Documents

[[
]]

Table 5.11-4 Installation Phase III Output Documents

[[
]]

5.11.3 System Factory Acceptance Test

[[

]]

5.11.4 Multi-System Factory Acceptance Test

[[

]]

5.11.5 Site Acceptance Test

[[

]]

5.11.6 Software Operations & Maintenance Manuals

[[

]]

5.11.7 Software Training Manuals

[[

]]

5.11.8 Software Installation

Software installation is described in Section 7.0.

5.11.9 Cyber Security Assessment Report

[[

]]

5.11.10 Installation Phase Baseline Review Record

[[

]]

5.12 OPERATIONS AND MAINTENANCE PHASE

[[

]]

5.12.1 Operations and Maintenance Phase Inputs

[[

]]

5.12.2 Operations and Maintenance Phase Outputs

[[

]]

Table 5.12-1 O&M Phase Output Documents

[[

]]

5.12.3 Operations and Maintenance Activities

[[

]]

5.12.4 (Deleted)

5.12.5 Operations and Maintenance Phase Baseline Review Record

[[

]]

5.13 RETIREMENT PHASE

[[

]]

Table 5.13-1 Retirement Phase Output Documents

[[
]]

5.13.1 Retirement Phase Activities Baseline Review Record

[[

]]

6. SOFTWARE INTEGRATION PLAN

6.1 INTRODUCTION

The Software Integration Plan (SIntP) describes the integration process for software modules, and software and hardware, and testing of the integrated product.

6.2 PURPOSE

The purpose of the SIntP is to:

- Describe the organization and responsibilities of individuals or groups involved in the test activities.
- Describe software test management (e.g., scheduling, resource planning, security, risks and contingency planning, anomalies, problem reporting, and training needs).
- Describe the methods for software testing.
- Provide the requirements and guidelines necessary to prepare, execute, and document software tests.
- Define required software test documentation.
- Define measurements and metrics for error tracking and resolution, and assess the success or failure of the software integration and software test effort.

The approach to software integration and testing activities shall be carried out in a deliberate and methodical manner.

For testing activities, deviations from this SIP shall be justified and approved by the RTPE. The justification and approval shall be prepared, reviewed, approved, and maintained in the software project DRF.

6.3 SOFTWARE INTEGRATION

Software integration consists of three steps:

1. Integrating the software modules together to form a single software program. During this step, module tests shall be performed to ensure that the software performs as designed.
2. Integrating the software programs with prototype hardware. During this step, integration tests shall be conducted to ensure the software modules perform as intended when installed in the prototype hardware.
3. Validating the resulting integrated software product. During this step, software validation tests shall be conducted to ensure the integrated software product perform as intended, and does not perform unintended functions.

During software integration, interface analyses, data flow analyses, timing, and sizing analyses shall be performed as appropriate. The results of the analyses shall be documented in the Software Functional Test Data Sheet (See Appendix D).

6.4 ORGANIZATION AND MANAGEMENT

Section 6.10 describes the test personnel roles and responsibilities. SQAPM [2.3(1.a), Section 5.0] describes the IVVT roles and responsibilities. CySPP [2.3 (1.b)] describes the Cyber Security roles and responsibilities.

6.5 MANAGEMENT AND ORGANIZATION INTERFACES

The test results are reported by the RTE to the RTPE through the reports outlined in this document.

The Design Team and Test Team interface with the IVVT and CyST. The CyST is responsible for performing Cyber Security Assessment on the test reports. The IVVT is responsible for performing verification of the class Q Software Functional Test Report (SFTR).

6.6 SCHEDULING AND PLANNING

The RTPE has overall responsibility for scheduling and planning test tasks and activities.

The schedule for software testing activities shall be integrated in the PWP as addressed in Subsection 3.6.2.

6.7 RESOURCES

Resource management includes the determination of the required resources. Resources include the following elements:

- Test facilities
- Test equipment and tools
- Qualified test engineers
- Any special needs for security, including cyber security adherence to CySPP [2.3 (1.b)]

6.8 TRAINING

The CyST Lead, IVVT Task Lead, and the RTPE shall ensure that their staff is trained in the test guidelines and methods described in Subsections 6.11 and 6.12 before the commencement of test activities.

6.9 REVIEWS

The test progress and issues related to the test shall be evaluated on a regular basis (e.g., during weekly review meetings). The progress report data from these meetings shall be used to track and update the project schedule. Special attention shall be given to circumstances indicating deficiencies in the testing process. If needed, the CAR process, as defined in the SQAPM [2.3(1.b)], shall be used to initiate corrective actions to improve the test process.

6.10 TEST PERSONNEL ROLES AND RESPONSIBILITIES

This section defines the test personnel roles and responsibilities.

6.10.1 Responsible Technical Project Engineer

The Responsible Technical Project Engineer (RTPE), is responsible for approving deviations from the SIP, scheduling and planning the test activities.

6.10.2 Responsible Test Engineer

The Responsible Test Engineer (RTE) is responsible for designing, executing, and documenting the test results in accordance with the Subsections 6.11.1 and 6.12.1.

6.10.3 Cyber Security Test Engineer

The Cyber Security Test Engineer (CySTE) is responsible for designing and executing the adversary based test plans, procedures, and test cases. The CySTE shall document the test results.

6.10.4 Test Personnel Qualifications

Test personnel shall be formally trained and qualified in the following:

- SMPM, SQAPM [2.3(1.a)] and CySPP [2.3(1.b)]
- Test guidelines, methods, techniques and tools
- Software Coding and Convention Guidelines
- The target platform in which the software product is undergoing testing
- The software language used by the software platform

6.11 SOFTWARE FUNCTIONAL TEST

The following test guidelines include the key elements that are required for performing test activities:

- Test preparation
- Test design
- Test execution
- Test summary

This test guideline conforms to RG 1.170 [2.2.3] and IEEE 829 [2.2.4].

6.11.1 Software Functional Test Guidelines

6.11.1.1 Test Preparation Guidelines

Test preparation ensures that the required test activities are properly carried out to ensure software quality. This is accomplished by identifying the resources that are required to support the development, execution, and the documentation of the test.

The individual responsible for test preparation shall perform the following tasks:

- Define the scope of the test and identify the software items to be tested.
- Design a detailed test schedule aligned with the project plan.

- Specify test prerequisites.
- Specify the test environment.
- Identify equipment, documentation, tools, and instrumentation needed to accomplish the test.
- Adjust the integrated project schedule to account for equipment, documentation, tool, and instrumentation needs.
- Assign qualified test engineers.
- Ensure the training needs are satisfied.
- Start the test report.

6.11.1.2 Test Design Guidelines

The RTE shall perform the following tasks:

- Specify the software features to be tested for each software item.
- Specify and provide justification for the software features not to be tested.
- Determine the test approach and specify the test techniques.
- Specify the test cases and acceptance criteria for each item.
- Develop the test procedures and instructions.

Structural testing is a test methodology in which test steps are based on knowledge of the internal structure of the software module or a group of software modules. A structural test may execute all the statements or branches in the software module to check how the system is implemented. Test techniques to be used for structural testing include:

- Branch testing - Executes each outcome of each decision point in a computer program.
- Path testing - Exercises every independent execution path through the computer program.
- Statement testing - Executes each statement of a computer program.

Functional testing is a test methodology using requirements external to a feature to derive test cases and test procedures. Functional testing verifies the end results at the feature I/O level. However, functional testing does not check on how the feature is realized, nor does it assume that all statements related to the feature are executed. Test techniques to be used for functional testing include:

- Module interface testing - Evaluates whether the values along the interface are correct as they relate to software modules that call them.
- Interface testing - Detects errors that may have been introduced into the system due to misinterpretation of the interface specification.
- Regression testing - Selectively re-tests software item to ensure that modifications have not caused unintended effects and that the software item subject to the test still complies with its specified requirements.

- Stress testing - Evaluates a system or component at or beyond the limits of its specified requirements.

Reviews, in the form of design walkthroughs, are conducted during the test design process to evaluate the adequacy of the selected test strategy and to ensure that the test features are identified for Class Q software.

6.11.1.3 Test Execution Guidelines

The test execution exposes the software test item to conditions that may reveal potential implementation errors. Test execution includes the following tasks:

- Obtain test items including relevant reference documentation.
- Set the test environment.
- Observe the software and hardware during testing for both the expected and unexpected behaviors.
- Confirm completeness and test termination requirements are satisfied.
- Document the test results while executing the test procedures.
- Initiate the change control process per SQAPM [2.3(1.a), Section 6.0] to resolve design errors encountered during the test.

6.11.1.4 Test Summary Guidelines

The test summary evaluates the test. It shall include the following:

- Test activities
- Test results
- Requirement traceability
- Test issues and the associated resolutions

Requirements traceability demonstrates that every functional requirement, performance requirement, and interface requirement in a SDD and Intra-Systems Data Communication Protocol Specification have been validated by the test. The SQAPM [2.3(1.a)] provides methods for performing traceability analysis.

6.11.2 Software Functional Methods

Methods used to perform software testing are described in the following sections.

6.11.2.1 Software Functional Test (Module Level)

[[

]]

6.11.2.1.1 Module Test Preparation

[[

]]

6.11.2.1.2 Module Test Design

[[

]]

6.11.2.1.2.1 Module Test Design - Class N Software

[[

]]

6.11.2.1.2.2 Module Test Design - Class Q Software

[[

]]

6.11.2.1.3 Module Test Execution

[[

]]

6.11.2.1.4 Module Test Summary

[[

]]

6.11.2.2 *Software Functional Test (Integration Level)*

[[

]]

6.11.2.2.1 Integration Test Preparation

[[

]]

6.11.2.2.2 Integration Test Design

[[

]]

6.11.2.2.3 Integration Test Execution

[[

]]

6.11.2.2.4 Integration Test Summary

[[

]]

6.11.3 Software Functional Test Documentation

6.11.3.1 Software Functional Test Data Sheet

[[

]]

6.11.3.2 Software Functional Test Metrics Sheet

[[

]]

6.11.3.3 Software Functional Test Report

[[

]]

6.12 SOFTWARE VALIDATION TEST

[[

]]

6.12.1 Software Validation Test Guidelines - Design Team

6.12.1.1 Software Validation Test Preparation

[[

]]

6.12.1.2 Software Validation Test Design

[[

]]

6.12.1.2.1 Component Level Validation Test Design

[[

]]

6.12.1.2.2 System Level Validation Test Design

[[

]]

6.12.1.3 Software Validation Test Execution

[[

]]

6.12.1.4 Software Validation Test Summary

[[

]]

6.12.2 Software Validation Test Guidelines - IVVT

[[

]]

6.12.3 Software Validation Test Documentation

6.12.3.1 Software Validation Test Plan

[[

]]

6.12.3.2 Software Validation Test Cases and Test Procedure Specification

[[

]]

6.12.3.3 Software Validation Test Metrics Sheet

[[

]]

6.12.3.4 Software Validation Test Report

[[

]]

6.13 PROBLEM REPORTING

[[

]]

6.14 MEASUREMENT AND METRICS

[[

]]

7. SOFTWARE INSTALLATION PLAN

7.1 INTRODUCTION

The Software Installation Plan (SIP) describes the software installation process and activities performed during the Installation Phase.

7.2 PURPOSE

The purpose of the SIP is to:

- Define the installation phase activities.
- Describe the installation procedures.
- Describe the software installation management, including but not limited to, scheduling, resource planning, security, risks and contingency planning, anomaly and problem reporting, and training needs.
- Provide the requirements and guidelines necessary to prepare, execute, and document software installation.

7.3 SCOPE

The scope of the SIP is to address software installation strategy and techniques. The documentation necessary to support and document the installation activities is also addressed in the SIP.

7.4 ORGANIZATION, MANAGEMENT AND RESPONSIBILITIES

Organization activities are addressed in Section 3.2.

7.5 INSTALLATION ACTIVITIES

The following sections define the activities to be performed during the Installation Phase of the Software Life Cycle.

7.5.1 Software Installation Procedure

A software installation procedure shall be produced for each software package. A combined procedure may be produced for multiple packages within a single system, but each system or logical group of systems shall have an installation procedure.

Installation procedure may be prepared as a standalone document or as part of the O&M Manual. The installation procedure shall include:

- Description of the software installation procedure
- Software installation methods and procedures
- Criteria used to determine the success or failure of the installation effort
- Checklist or sequence of steps that confirms the correctness and completeness of in the installation activities for a specific systems in accordance with the system design

documents. The following list is an example of items to be considered as part of the checklist:

- Affected functions are inoperable and in a safe condition according to the plant's technical specifications before proceeding with installation.
- The computer system is functional.
- The sensors and actuators are functional.
- All cards are present and installed in the correct slots.
- The communication system is correctly installed.
- The correct software versions are installed on the correct computers.
- Appropriate return-to-service testing has been successfully conducted before declaring the modified function operable.
- Installation configuration tables are complete.
- Environmental conditions (e.g., temperature, humidity, vibration, and rack space) are considered and provided for.
- Special tools, methods, or techniques used to accomplish the installation function shall be identified.
- Installation tools shall be qualified with a degree of rigor and level of detail appropriate to the safety significance of the software utilizing the installation tools.
- Security provisions have been satisfied.
- Precautions to ensure personnel and plant safety have been identified.

7.5.2 Software Installation Reporting

A software installation report shall be produced for each software installation procedure. A combined report may be produced for multiple packages within a single system. However, each system or logical group of systems shall have its own installation report.

An installation report for each package or system shall be produced upon the completion of the installation effort. The installation report shall include the following installation activities as a minimum:

- Serial numbers or other identification for the hardware platform on which the software is installed
- Software revisions
- Circuit board revisions
- Software file integrity check
- Test results
- Anomalies discovered during installation
- Any associated data sheets generated during the installation

- Installation test summary
- Any user configurable parameter values
- Indication of Licensee approval and acceptance of the installation activities
- Completed checklist

7.5.3 Installation Configuration Tables

Where applicable, installation configuration tables shall be produced. The tables shall include the functional characteristics defined in Subsection 7.5.1 to ensure the software is correctly configured for the software products being installed.

When applicable, configuration tables shall be developed for each software system or logical group of systems. Each user configurable function shall be defined, along with each configurable mode. Each configurable mode shall include the function, safety, and security of the overall application. The configuration tables shall include the following items:

- Software configuration tables shall include the information necessary for the correct operation of the system and its associated plant functions. This includes any vendor default settings used to test and accept the initial configuration.
- Installation configuration tables shall be consistent with the system specifications.
- Software configuration tables shall contain system specific data.
- Class Q software shall be required to provide traceability for each installed program element backward to the integrated software elements that created that installed program element.

7.5.4 Operations and Maintenance Manual

The O&M Manual shall be produced for each system or logical group of systems. O&M Manual shall include installation details necessary to enable the end user to install the software on the system.

O&M Manuals are described in Subsection 8.5.1.

7.5.5 Training Manuals

The training manuals for each software product or logical group of software products shall be produced. The training manuals are based on design documents and O&M manuals. The training manuals provide the basis for training the Licensee or end-user.

Training manuals are described in Subsection 9.3.2.

7.6 METHODS AND TOOLS

7.6.1 Installation Methods

Installation methods and tools used to support installation shall be defined in each installation procedure.

7.6.2 Archive Retrieval

The software package shall be placed under CM as required by the SQAPM [2.3(1.a) Section 6.0]. Plant-specific methods of archival and retrieval are the responsibility of the Licensee and are beyond the scope of this SMPM. Where applicable, specific backup and recovery procedures shall be included in the maintenance section of the O&M Manual.

7.6.3 Installation Test

An installation test procedure shall be developed as a separate document or as part of the installation procedures for each software package to be installed.

7.6.4 Installation Documentation and Problem Reporting

The problem or issues encountered during the installation process shall be reported in an installation report. The SQAPM [2.3(1.a)] defines a process for problem reporting and corrective action.

7.6.5 Verification and Validation Methods

Documentation produced during the Installation Phase shall be verified and validated in accordance with the SQAPM, [2.3(1.a) Section 5.0].

7.7 MEASUREMENTS AND METRICS

Measurement and metrics shall be developed in accordance with the SQAPM [2.3(1.a)].

8. SOFTWARE OPERATIONS AND MAINTENANCE PLAN

8.1 INTRODUCTION

The Software Operation and Maintenance Plan (SOMP) defines the processes and activities used to operate and maintain the software product during plant operation.

8.1.1 Purpose

The SOMP defines the requirements, methods and considerations for problem reporting, disposition of change request, backup media maintenance and disaster recovery operations during the Operation and Maintenance Phase of the Software Life Cycle.

8.1.2 Scope

The scope of the GEH SOMP addresses the activities required to support the licensee during the Operation and Maintenance phase of the software life cycle.

8.2 ORGANIZATION, MANAGEMENT AND RESPONSIBILITIES

Organization activities are addressed in Section 3.0. Management activities are addressed in Section 5.0. Responsibilities are addressed in Subsection

8.3 ACTIVITIES

The following sections define the O&M Phase activities associated with licensed software support. Changes to software that result from these activities shall be performed in accordance with the software development process described in Section 5.0, and the configuration management process described in SQAPM [2.3(1.a), Section 6].

8.3.1 Operation Phase Activities

GEH shall have a reporting system in place compliant with reference Reporting of Defects and Noncompliance Under 10 CFR Part 21 [2.3(3.g)] to notify the Licensees and the US NRC when a change request raises a condition associated with any Class Q software that may:

- Have the potential to create a Substantial Safety Hazard or performance degradation
- Contribute to exceeding a Technical Specification Safety Limit
- Have generic safety implications
- Result in reporting safety implications to a customer, the NRC, or other regulatory agency

GEH shall have a reporting system in place to ensure prompt notification of software issues, software defects, and software revisions when a change request raises conditions associated with any Class N software that may:

- Generate a revision to an existing version of the software
- Have the potential to result in performance degradation issues
- Have the potential to exercise protection system functions

8.3.2 Maintenance Phase Activities

GEH maintenance support activities shall include:

- Application and data backup and disaster recovery
- Backup media maintenance
- Cyber security support in accordance with the CySPP
- Support for installation of revised software

8.4 (Deleted)

8.4.1 (Deleted)

8.4.2 (Deleted)

8.5 OPERATION AND MAINTENANCE MANUAL

An O&M manual shall be developed for each software product or logical group of software products. Development of the O&M Manual shall be initiated during the Requirements Phase and completed during the SFAT to support the installation of the software product in accordance with Operation and Maintenance Instruction Manuals [2.3.(2.1)].

The operation section of O&M Manual shall include a description of the actions available to the operator/user as listed below:

- The operating modes
- Error messages including description and error recovery methods
- Backup and recovery procedures
- Operator actions shall be specified in terms of inputs supplied by the operator or system
- Actions initiated by the operator
- Responses to the operator or system

The purpose and operation of each function shall be described including interfaces with other systems. The O&M manual shall describe methods, techniques, tools, software, hardware, and associated documentation required to operate the software product.

The O&M Manual shall describe the operational environment within which the software product shall operate. This includes:

- Precautions
- Limitations
- Personnel or plant hazards

- Maintaining the integrity of the Cyber Security Defensive Model including considerations for restricted access to the model itself in accordance with the CySPP [2.3(1.b)]
- Variables in the physical environment that the software must monitor and control
- User interfaces. User interfaces shall be described fully for each category of operator or user.
- Required actions to ensure cyber security protection during O&M, restoration of integrity of cyber security defensive model after O&M activities, and recognition of cyber events in which indication, control, and protection features may have been compromised.

The operations section of the O&M manual shall be consistent with the system operations, system requirements, the system design, documented descriptions, and known properties of the operational environment within which the software shall operate. Individual user instructions shall not contradict other instructions. Uniform and consistent terminology, notation, and definitions shall be used throughout the manuals. Vendor supplied manuals shall adhere to the same requirements.

The maintenance manual section of the O&M manual shall include:

- Precautions
- Limitations
- Personnel or plant maintenance hazards
- Security vulnerabilities
- Trouble shooting and reporting procedures and methods
- A description of or reference to Configuration Management and Change Control procedures. The Configuration Management and Change Control procedures shall:
 - Confirm that changes have been implemented correctly, the changes and a sufficient test overlap have been defined and performed, and that no faults have been introduced in the software by the changes.
 - Ensure that software is functioning properly after the maintenance.
 - Upgrade field procedures. Field upgrade procedures shall be described, including:
 - Installation procedures
 - Installation test procedures
 - Installation test checklists
 - Installation test data sheets

8.5.1 Software Operation and Maintenance Manuals

The Software Operation and Maintenance Manual shall be developed in accordance with the requirements outlined below and may be incorporated into the system O&M Manual.

The Software Operation Manual shall, at a minimum, include:

- Information necessary for all operating modes. This includes normal operation, off normal operation, and emergency operation.
- Start-up and shutdown of the software product. This includes error recovery and backup.
- List(s) of error messages. Error messages shall be listed together and include definitions and corrective action(s) by the operator.
- Description of the operational environment within which the software shall operate. This description shall include precautions and limitations that must be observed during operations to avoid exposing personnel or the plant to hazards or security vulnerabilities.
- Description of each user interface for each category of user, including operators, shift supervisors and, nuclear engineers.
- The Software Maintenance Manual shall, at a minimum, include:
 - Procedures describing how a change to the operational software should be implemented.
 - Identification of the precautions and limitations that must be observed during maintenance to avoid exposing personnel or the plant to hazards or security vulnerabilities.
 - Change control process shall be described or referenced.
 - Software Installation Procedure. This includes regression test steps that confirm the revised/enhanced software is correctly installed and that no faults have been introduced in the revised/enhanced software.
 - Methods used to restore older versions of software and methods used to back up software.
 - Methods to troubleshoot and diagnose the software product.

8.5.2 Verification and Validation Methods

The O&M Manual shall be evaluated in accordance with the SQAPM [2.3(1.a) Section 5.0].

8.6 MEASUREMENT AND METRICS

Measurements and metrics shall be developed in accordance with the SQAPM [2.3(1.a)].

9. SOFTWARE TRAINING PLAN

9.1 INTRODUCTION

The Software Training Plan (STrngP) describes the software training activities to be carried out before and during the operation of software products. The STrngP addresses the management, implementation and resource characteristics as addressed in BTP-14 [2.2.1].

9.1.1 Purpose

The purpose of the STrngP is to define:

- The requirements and methods used to develop the training program and manual.
- The training needs of appropriate plant staff, including operators, I&C engineers, and technicians.
- A general description of the training facilities.
- The organization supporting the training effort including interfaces and responsibilities.

9.1.2 Scope

The scope of the STrngP is to address the development and implementation of the training program and documentation for the software product to ensure proper operation and usage (e.g., safety and security topics) by multiple disciplines of users. The disciplines subject to these training requirements include operators, engineers, maintenance personnel, and management personnel.

9.2 TRAINING ORGANIZATION

This section provides a description of the training organization supporting the software product training effort as well as organizational interfaces and responsibilities. The organizational responsibilities are identified in Subsection 3.4. The Training Services Lead (TSL) is a functional position responsible for assignment of personnel to support training for the software products. The TSL ensures the training requirements are accomplished. The training requirements are established based on the needs to operate and maintain the software products. The TSL augments the training staff to support the required training based on the Licensee needs.

9.2.1 Responsibilities and Qualification

The TSL has overall responsibility for the trainer qualification process. Qualified personnel are selected for the Trainer positions based on work related experience and knowledge in the operation of Nuclear Power Plant and I&C Systems, as detailed in the individual's resume.

9.3 TRAINING ACTIVITIES

This section defines the required training activities including:

- Development and maintenance of training plan.
- Development and review of the training manual and training materials.

- Development of training program and training courses.
- Implementation of the Training Program.

9.3.1 Training Plan

Training Plan shall be prepared by the TSL to define the scope, approaches and implementation of the training program. The Training Plan shall include:

- Objective
 - Describe the goals of the training
- Roles and Responsibilities.
 - Identify roles and responsibilities of staff that will develop the training materials
 - Identify the trainers that will conduct the training
 - Identify organization that will provide reprographic services
- Training Requirements
 - Identify groups and individuals that require training
 - Specify the needs by skill level
 - Identify timeframe for which training is required
- Training Tools and Techniques
 - Describe the training techniques
 - Identify tools needed such as computers, facilities, training manuals and materials
- Training Prerequisites
 - Identify the required prerequisites for the trainees
- Schedule
 - Develop a training schedule to include development of course content, duration, planned dates, trainees, trainer, location, and disposition of post training feedback
- Evaluation
 - Describe how feedback will be obtained and analyzed
 - Describe how metrics will be collected

9.3.2 Training Manual and Materials

The training manual shall be prepared by the trainer and shall address the following:

- Startup
- Shutdown
- Installation
- Backup

- Restoration
- Configuration
- Calibration
- Troubleshooting
- Replacing failed hardware modules
- Plant modes, including alarm and indicator responses
- Operating specific scenarios
- Recommended surveillance testing
- Security, includes Cyber Security
- Identification, operation and maintenance of safety-related software products
- Training assessment

Software training manual may be prepared as a stand alone document or as part of the training manual. Software training manuals shall be prepared in accordance with the following requirements:

- Description of actions available to the operator and the technician for all operating modes, including error recovery.
- Description of operator actions specified in terms of inputs supplied by users and equipment, actions initiated by the operation, and responses to the user input.
- Description of the maintenance environment, including precautions and limitations that must be observed during maintenance to avoid incorrectly configuring, damaging, or otherwise defeating the system's functionality and thus exposing the plant to hazards.
- Description of the operational environment within which the software shall operate, including precautions and limitations that must be observed during operations to avoid exposing personnel or the plant to hazards.
- Description of variables in the physical environment that the software must monitor and control. User interfaces should be fully described for each category of user.

Training materials shall be prepared to supplement the training manuals. Multimedia presentation, lecture course outlines, case studies, laboratory exercises and interactive student quizzes may be used to enhance the training and learning objectives.

The Training Manual and training materials shall be reviewed and controlled in accordance with the SQAPM [2.3(1.a)]. The training manual shall be completed and accepted by the Licensee prior to the start of the training sessions.

9.3.3 Training Program and Training Courses

A comprehensive training program shall be developed to identify the needs for different types of training and the categories of people requiring training for each need with a comprehensive set of

established training modules or programs developed for the software products. Training is provided for the following generic types of system users:

- Plant Operations
- Maintenance
- System Administrator
- General Purpose User
- Engineering

The training program shall adhere to the HFE Training Development Implementation Plan [2.3(1.c)], which describes the processes, methods, and systematic approaches for the development of the training program.

The training program plan shall be initiated during the Requirement Phase and completed prior to the Test Phase when the system users are expected to apply required expertise in witnessing software validation testing during the Test Phase and the systems testing during the Installation Phase. The training program plan may be revised as needed to incorporate training lessons learned, feedback and recommendations provided by the students.

The training program and course outlines, and training schedule shall be prepared by the trainer and approved by the TSL.

When preparing a training course, the trainer shall determine the type of training facility that provides the most effective nuclear training. Examples of such training facilities are:

- Dedicated classroom space (e.g., conference rooms)
- Instructor-led classroom software lab facilities
- Self-study computer lab facilities
- A remote training access tool (e.g., presentation tools) which allow training at a remote training workstation
- Control room simulator

9.3.4 Training Implementation

Training shall be implemented in accordance with an approved Training Plan. The TSL shall:

- Ensure the provision of the necessary materials
- Arrange the locations and facilities for training
- Assign trainers or instructors and, if necessary, train them
- Ensure the enrolling of students
- Monitor course effectiveness via student exams, student feedback, management observations of training, and post-training on-the-job performance
- Collect lessons learned and information from student and management feedback
- Evaluate post-training on-the-job performance

- Update the materials for the next training cycle

9.4 METHODS AND TOOLS

Methods and tools used to perform software training shall be defined in each manual as required for each software product or logical group of software products. The responsible trainer shall determine the content and methods for each training course. The TSL shall approve the course content and methods.

9.5 MEASUREMENT AND METRICS

Metrics provide a basis for determining the effectiveness of the training program. Metrics selected during the development of the training program may be a combination of tools based on the nature of the training program being offered. For example, a training course providing a one-day overview session would utilize a different set of metrics than a four-week course that utilizes extensive use of simulation training tools. The training program should allow for quizzes or practical exams based on course objectives relevant to the task responsibilities. The training program may also allow self-study for certain aspects of the training.

Examples of training tool metrics are:

- End of course student feedback
- Certification exams
- Exam validity and difficulty indexing
- Computer software laboratory tests
- Student performance measures for plant scenarios in a training simulator
- Management observation of training
- Evaluation of post-training on-the-job performance

The test results or training results obtained at the end of the training activities shall be measured, recorded, analyzed, and reported.

10.APPENDICES

10.1 APPENDIX A SOFTWARE PLANS CONFORMANCE REVIEW

The Regulatory Guides and IEEE Standards have been reviewed for conformance. In general, the IEEE Standards provide more detailed guidance for the implementation activities. When requirements derived from the Standards are specifically addressed within this plan, a commitment to the approach is made. Conformance clarification and justification is provided in this Appendix.

[[
]]

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			
II				II				II

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			
Regulatory Guides								
[[
]]	

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			
IEEE Standards from Section 2.2.4								
[[

Appendix A - Software Plans Conformance Review								
Item	Reg. Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			
IEEE Standards from Section 2.4								
[[

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

Appendix A - Software Plans Conformance Review								
Item	Reg Guide	IEEE Stand.	Related Software Plan			Deviation	Conform. Code	Justification
			SMPM	SQAPM	None			

10.2 APPENDIX B ACRONYMS AND ABBREVIATIONS

The following acronyms and abbreviations are used throughout the SMPM.

Acronym	Meaning
ASL	Approved Suppliers List
ASME	American Society of Mechanical Engineers
BR	Baseline Review
BRR	Baseline Review Record
BRT	Baseline Review Team
BTP	Branch Technical Position (see HCIB)
CAQ	Condition Adverse to Quality
CAR	Corrective Action Request
CCB	Change Control Board
CDA	Critical Digital Asset
CEO	Chief Executive Officer
CFR	Code of Federal Regulations
CI	Configuration Item
CIR	Customer Issue Resolution
CM	Configuration Management
CMM	Configuration Management Manager
CMS	Configuration Management System
COTS	Commercial-Off-The-Shelf
CPU	Central Processing Unit
CySA	Cyber Security Assessment
CySPP	Cyber Security Program Plan
CyST	Cyber Security Team
CTS	Commitment Tracking System
DCD	Design Control Document

Acronym	Meaning
DCPS	Data Communication Protocol Specifications
DRF	Design Record File
ECA	Engineering Change Authorization
ECN	Engineering Change Notice
EIA	Electronic Industries Alliance
EMC	Electromagnetic Compatibility
EOP	Engineering Operating Procedure
EPRI	Electrical Power Research Institute
ERM	Engineering Review Memorandum
ESBWR	Economic Simplified Boiling Water Reactor
FDDR	Field Deviation Disposition Request
FDI	Field Disposition Instruction
FMEA	Failure Modes and Effects Analysis
GEH	GE Hitachi Nuclear Energy
HFE	Human Factors Engineering
HFEITS	Human Factors Engineering Issue Tracking System
HICB	Instrumentation and Control Branch, NRC Branch Technical Positions for I&C
HSI	Human System Interface
HSS	Hardware/Software Specification
ISCPS	Intra-System Communication Protocol Specification
I&C	Instrumentation and Controls
IEEE	Institute of Electrical and Electronic Engineers
I/O	Input/Output
IP	Implementation Plan
IR	Inspection Report
ISO	International Standards Organization
IV&V	Independent Verification and Validation
IVVT	Independent Verification and Validation Team

Acronym	Meaning
LD	Logic Diagram
LLC	Limited Liability Corporation
LTR	Licensing Topical Report
MCR	Main Control Room
[[]]
MMI	Man Machine Interface
MMIS	Man Machine Interface System
N/A	Not Applicable
N-DCIS	Nonsafety-related – Distributed Control and Information System
NPP	New Plant Project
NRC	Nuclear Regulatory Commission
O&M	Operation and Maintenance
P&ID	Piping & Instrumentation Diagram
P&P	Policies and Procedure
PDM	Project Design Manual
PDMS	Product Data Management System
PDS	Previously Developed Software
PM	Project Manager
PMT	Project Management Team
POC	Point of Contact
PQC	Product Quality Certification
PR	Problem Report
PRA	Probabilistic Risk Assessment
PRM	Process Radiation Monitor
PWP	Project Work Plan
Q-DCIS	Safety-related - Distributed Control and Information System
QA	Quality Assurance
QCE	Quality Control Engineer

Acronym	Meaning
RCCE	Responsible Configuration Control Engineer
RE	Responsible Engineer
RG	Regulatory Guide
RM	Responsible Manager
RMCN	Review Memorandum Change Notice
RSE	Responsible System Engineer
RTA	Requirements Traceability Analysis
RTE	Responsible Test Engineer
RTM	Requirements Traceability Matrix
RTPE	Responsible Technical Project Engineer
RV	Responsible Verifier
SAE	Simulation Assisted Engineering
SAT	Site Acceptance Test
SATT	Site Acceptance Test Team
SBD	Software Build Description
SCM	Software Configuration Management
SCMP	Software Configuration Management Plan
SDD	Software Design Description
SDP	Software Development Plan
SDS	System Design Specification
SFAT	System Factory Acceptance Test
SFT	Software Function Test
SFTR	Software Functional Test Report
SIntP	Software Integration Plan
SIP	Software Installation Plan
SITT	System Installation Test Team
SMP	Software Management Plan
SMPM	Software Management Program Manual

Acronym	Meaning
SOMP	Software Operations and Maintenance Plan
SPE	Software Project Engineering
SQA	Software Quality Assurance
SQAP	Software Quality Assurance Plan
SQAPM	Software Quality Assurance Program Manual
SRP	Standard Review Plan
SRS	Software Requirements Specification
SSA	Software Safety Analysis
SSP	Software Safety Plan
SST	Software Safety Team
STP	Software Test Plan
STrngP	Software Training Plan
SVT	Software Validation Testing
SVTP	Software Validation Test Plan
SVVP	Software Validation and Verification Plan
SyRS	System Requirement Specification
TBD	To Be Determined
TPE	Technical Project Engineer
TR	Topical Report
TSL	Training Services Lead
UIS	User Interface Specification
V&V	Verification and Validation
WBS	Work Breakdown Structure

10.3 APPENDIX C DEFINITIONS

Term	Definition
Acceptance Criteria	The criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity [IEEE 610.12].
Acceptance Testing	Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system [IEEE 610.12].
Algorithm	A finite set of well-defined rules for the solution of a problem in a finite number of steps [IEEE 610.12].
Anomaly	Anything observed in the documentation or operation of software that deviates from expectations based on previously verified software products or reference documents [IEEE 610.12].
Application software	Software designed to fulfill specific needs of a user [IEEE 610.12].
Application Software Package	A collection of software modules brought together to form a single software application, e.g., an instrument (see also System Software Package and Package).
Assembly code	Computer instructions and data definitions expressed in a form that can be recognized and processed by an assembler.
Baseline	Items that have been formally reviewed and agreed upon, that thereafter serve as the basis for further development, and that can be changed only through formal change control procedures [IEEE 610.12].
Baseline Review	A formal review, conducted at the end of each process step of the software engineering design process, and requested by the Design Team's responsible TPE. The baseline review process is under the control of Software Project Engineering (SPE). The Baseline Review Team (appointed by the BRT Task Lead engineer) performs the review. These reviews are intended to confirm adherence to the project documents. The Baseline Reviews are performed and documented in accordance with the Software Quality Assurance Program Manual.
Branch testing	Testing designed to execute each outcome of each decision point in a computer program [IEEE 610.12].
Build	An operational version of a system or component that incorporates a specified sub set of the capabilities that the final product will provide [IEEE 610.12].

Term	Definition
Certification	A written guarantee that a system or component complies with its specified requirements and is acceptable for operational use [IEEE 610.12].
Code	In software engineering, computer instructions and data definitions expressed in a programming language or in a form output by assembler, compiler, or other translator [IEEE 610.12].
Code review	A meeting at which software code is presented to project personnel, managers, users, customers, or other interested parties for comment or approval [IEEE 610.12].
Coding	In software engineering, the process of expressing a computer program in a programming language [IEEE 610.12].
Commitment Tracking System	System used to manage the Conditions Adverse to Quality (CAQs). A Corrective Action Request (CAR) is used to document a CAQ, or an opportunity for process/product improvement, provide for timely evaluation, and record objective evidence of actions taken. Corrective Action Process [2.3(2.m)] specifies the responsibilities for actions to promptly identify, record and correct, as appropriate, CAQs, and to assure that these conditions do not affect the quality of a product or service.
Component	One of the parts that make up a system. A component may be hardware of software and may be subdivided into other components [IEEE 610.12].
Computer language	A language designed to enable humans to communicate with computers [IEEE 610.12].
Configuration control	An element of configuration management, consisting of the evaluation, coordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification [IEEE 610.12].
Configuration Item	An aggregation of hardware, software, design documents or procedures that is designated for configuration management and treated as a single entity in the configuration management process [IEEE 610.12].
Critical Digital Asset	A digital device or system that plays a role in the operation or maintenance of a critical system and can impact the proper functioning of a critical system.
Criticality Analysis	The degree of impact that a requirement, module, error, fault, failure, or other item has on the development or operation of a system. A method used to determine the impact of the software product on the system & environment as a whole and thereby determine the software importance (i.e. Software Class Q, N3 or N2).

Term	Definition
Design Documentation	Design Documentation is information recorded about a specific life cycle activity. Documentation includes software life-cycle design outputs and software life cycle process documentation. A document may be in written or electronic format, and may contain text, illustrations, tables, computer files, program listings, binary images, and other forms of expression. A document for an activity may be packaged with documents for other activities, or documents for non-software life cycle activities. A document for an activity may be divided into several individual entities.
Design output	Documents, such as drawings and specifications, that define technical requirements of structures, systems, and components. For software, design outputs include the products of the development process that describe the end product that will be installed a nuclear power plant. The design outputs of a software development process include SRS, SDD, hardware and software architecture designs, code listings, system build documents, installation configuration tables, O&M manuals, and training manuals.
Design phase	The phase in the software life cycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements [IEEE 610.12].
Design Record File	A formal controlled information record under GEH procedures for in-progress and completed engineering work which is retained and from which work can be retrieved.
Design Reviews	Formal, design adequacy evaluations which are performed by knowledgeable persons other than those directly responsible and accountable for the design in accordance with GEH Design Review [2.3(2.c)]. Design reviews are used to verify that product designs meet functional, contractual, safety, regulatory, industry codes and standards, and company requirements.
Deviation	A departure from a specified requirement.
Documentation	A collection of documents on a given subject [IEEE 610.12].
Error	An incorrect step, process, or data definition [IEEE 610.12].
Failure Mode and Effects Analysis	A tabular method of providing traceability from the modes by which a system may fail and the effect of that failure on the ability of the system to perform its function, or the ability of a collection of systems to recover from the failure.
Fault Tree	A pictorial method of providing traceability from the modes by which a system may fail and the effect of that failure on the ability of the system to perform its function, or the ability of a collection of systems to recover from the failure.

Term	Definition
Field Deviation Disposition Request	Field Deviation Disposition Request (FDDR) is used for documenting and disposition of the technical position for a deviation required in the field in supplied hardware, software, or services (see GEH Field Deviation Disposition Request [2.3(2.j)]).
Firmware	The combination of a hardware device and computer instructions and data that reside as read-only software on that device [IEEE 610.12].
Functional Testing	A system/software test methodology that is derived from external specifications and requirements of the system. Such testing ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions [IEEE 610.12]. Methods for functional testing include random testing and testing at boundary values. It verifies the end results at the system level, but does not check the implementation techniques, nor does it assume that all statements in the program are executed.
Implementation Phase	The phase in the software life cycle during which a software product is created from design documentation and debugged [IEEE 610.12].
Independent Verification and Validation (IV&V)	Verification and Validation performed by an Organization that is technically managerially and financially independent of the development Organization [IEEE 610.12] and RG 1.168 Section C3 [2.2.3].
Installation Phase	The phase in the software life cycle during which the software product is installed into its operational environment and tested to ensure that it performs as intended [IEEE 610.12].
Instrument	A hardware device used for analytical or control functions and usually containing an embedded microprocessor(s).
Integration Testing	Testing in which software elements, hardware elements, or both are combined and tested to evaluate the interaction between them [IEEE 610.12].
Interface	<ol style="list-style-type: none"> 1) A shared boundary across which information is passed. This definition is interpreted broadly to include design interfaces between participating design organizations. 2) A hardware or software component that connects two or more other components for the purpose of passing information from one to the other. 3) To connect two or more components for the purpose of passing information from one to the other. 4) To serve as a connecting or connected component as in (2). [IEEE 610.12 as modified by RG 1.69

Term	Definition
Metric	A quantitative measure of the degree to which a system, component, or process possesses a given attribute [IEEE 610.12].
Module	A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading; for example, the input to, or output from, and assembler, compiler, linkage editor, or executive routine [IEEE 610.12].
Module Testing	Testing of individual hardware or software units or groups of related units [IEEE 610.12].
Operations and Maintenance Phase	The phase in the software life cycle during which the software product is functioning in its operational environment, monitored for satisfactory performance and modified as necessary to correct problems or to respond to changing requirements [IEEE 610.12].
Package	A separately compilable software component consisting of related data types, data objects and sub-programs [IEEE 610.12].
Path Testing	Testing designed to execute all or selected paths through a computer program [IEEE 610.12].
Planning Phase	The initial phase of a software development project, in which project scope, purpose, strategy, schedule and milestones are established and user needs through documentation (for example, system definition documentation and procedures) are described and evaluated.
Procedure	A course of action to be taken to perform a given task [IEEE 610.12].
Process	A sequence of steps performed for a given purpose, e.g., the software development process [IEEE 610.12].
Project Management Plan	A document that describes the technical and management approach to be followed for a project. The plan typically describes the work to be done, the resources required, the methods to be used, the procedures to be followed, the schedules to be met, and the way that the project will be organized [IEEE 610.12].
Regression Testing	Selective re-testing of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements [IEEE 610.12].

Term	Definition
Requirement	<p>A condition or capability that must be met or possessed by a system or system component to satisfy a contract standard specification or other formally imposed documents [IEEE 610.12].</p> <p>In specifying requirements, the word shall is used to indicate mandatory requirements and from which no deviation is permitted ('shall' and 'required to' are equivalent in meaning).</p> <p>Requirements are not specified with the word should. Instead, it is used to indicate that a recommended course of action and is particularly suitable, without mentioning or excluding other courses of action; Also, a certain course of action is preferred but not necessarily required; Also, that (in the negative form) a certain course of action is not prohibited ('should' and 'recommended' are equivalent in meaning).</p>
Requirements Analysis	The process of studying user needs to arrive at a definition of system, hardware, or software requirements [IEEE 610.12].
Requirements Phase	The phase in the Software Life Cycle during which the requirements for a software product are defined and documented [IEEE 610.12].
Requirements Traceability Analysis	The process of tracing the life of a requirement, in both forward and backward direction, using independent verification and traceability matrix to analyze the identified relationships from its source, through design, development, testing and installation to assure the correctness, completeness and accuracy of the software product.
Responsible Configuration Control Engineer	The person assigned responsibility for the configuration management of the I&C software products.
Responsible Engineer	The person responsible for a given technical item, e.g., the design and development of the documentation.
Responsible Technical Project Engineer	The person with overall technical responsibility for ensuring that the hardware and software design of a software product meets the specified requirements.
Responsible Verifier	The Responsible Verifier(s) is an individual who has the independence described in GEH Independent Design Verification [2.3(2.f)] for verifications, or in GEH Deferred Design Verification [2.3(2.g)] for deferred verifications of design process and the accompanying documents.
Retirement	Permanent removal of a system or component from its operational environment [IEEE 610.12].
Simulation	A model that behaves or operates like a given system when provided a set of controlled inputs [IEEE 610.12].

Term	Definition
Software Class N2	Nonsafety-related system software whose failure cannot adversely affect a safety related function.
Software Class N3	<p>Nonsafety-related systems software whose failure could challenge safety systems as defined below:</p> <ul style="list-style-type: none"> a. Software whose inadvertent response to stimuli, failure to respond when required, response out-of-sequence could result in a accident or transient as defined in the DCD, Chapter 15 [2.1(6)]. b. Software that is intended to mitigate the result of an accident. c. Software that is intended to recover from the result of an accident.
Software Class Q	Software performs functions classified per GEH Safety-Related Classification determination process [2.3(2.k)] as Safety-Related.
Software Development Process	The process by which user needs are translated into a software product. The process involves translating user needs into software requirements, transforming the software requirements into design, implementing the design in code, testing the code, and sometimes, installing and checking out the software for operational use [IEEE 610.12].
Software Feature	A distinguishing characteristic of a software item, such as, performance, portability, or functionality.
Software Item	Source code, object code, job control code, control data, or a collection of these items [IEEE 610.12].
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 [IEEE 610.12].
Software Life cycle Phase	The division of the software life cycle into discrete logical units. The I&C software life cycle is divided into eight phases, namely, Planning, Requirements, Design, Implementation, Test, Installation, Operation & Maintenance and Retirement.
Software Module	See Module
Software Package	See Package
Software Unit	See Module
Source Code	Computer instructions and data definitions expressed in a form suitable for input to an assembler, compiler, or other translator.
Statement testing	Testing designed to execute each statement of a computer program [IEEE 610.12].
Stress testing	Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements [IEEE 610.12].

Term	Definition
Supplemental Document	Controlled documents that are referenced or used in conjunction with this plan. These are the enabling documents that either augment or enable the performance of the activities stated in this plan.
Support software	Software that aids in the development or maintenance of other software; for example, compilers, loaders, and other utilities [IEEE 610.12].
Supporting Document	Controlled documents used in the production of this plan. These documents form the design basis for the activities stated in this plan.
System Testing	Testing conducted on a complete, integrated system to evaluate the systems compliance with its specified requirements [IEEE 610.12].
Technical Project Engineer	The person with overall technical responsibility for ensuring that the hardware and software design of a software product meets the specified requirements.
Test case	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 [IEEE 610.12].
Test Item	A software item that is an object of testing [IEEE 610.12].
Test Log	A chronological record of all relevant details about the execution of a test [IEEE 610.12].
Test Objective	An identified set of software features to be measured under specified conditions by comparing actual behavior with the required behavior described in the software documentation [IEEE 610.12].
Test Phase	The phase in the software life cycle during which the components of a software product are integrated with the hardware and evaluated to determine whether or not performance requirements have been satisfied [IEEE 610.12].
Test Plan	A document describing the scope, approach, resources, and schedule of intended test activities. It identifies test items, the features to be tested, the testing tasks, who will do such task, and any risks requiring contingency planning [IEEE 610.12].
Traceability Matrix	A matrix that records the relationship between two or more product specifications (i.e., design documentation) of the development process (e.g., a matrix that records the relationship between the requirements and the design of a given software component) [IEEE 610.12].
Unit Testing	See Module Testing.
User interface	An interface that enables information to be passed between a human user and hardware or software components of a computer system [IEEE 610.12].

Term	Definition
Verification and Validation (V&V)	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 [IEEE 610.12].

**10.4 APPENDIX D SOFTWARE FUNCTIONAL TEST DATA SHEET
(EXAMPLE)**

SOFTWARE FUNCTIONAL TEST DATA SHEET			
Page:	1 of 1	DRF#	
Responsible Engineer		Date	
Integration Test Engineer		Date	
Application Software Package		Revision	
Location		Software Class	
Instrument			
Test Entity			
Test Description	Finding		Status
Notes:			

10.5 APPENDIX E SOFTWARE FUNCTIONAL TEST METRICS SHEET (EXAMPLE)

SOFTWARE FUNCTIONAL TEST METRICS SHEET			
Page:		1 of 1	DRF#
Application Software Package:			Revision:
Subsystem:			Software Class:
Total Errors		Error Type	SFT
Major	Minor		SECTION
0	0	Data reference errors - errors that occur when data items are referenced improperly,	
0	0	Data declaration errors - errors resulting from conflicts between intended and actual usage,	
0	0	Computation errors - errors resulting from improper analysis or computational precision,	
0	0	Comparison errors - errors resulting from improper or imprecise condition expressions,	
0	0	Control flow errors - errors resulting from incorrect branching targets,	
0	0	Interface errors - errors resulting from improper passage of data between software modules,	
0	0	Input/Output errors - errors resulting from incorrect data formats or invalid interface specification,	
0	0	Hardware Errors, and	
0	0	Hardware/Software interaction Errors.	
0	0	Task Interaction Errors	
0	0	Other Errors	
0	0	Totals	
Notes:			

10.6 APPENDIX F SOFTWARE VALIDATION TEST METRICS SHEET (EXAMPLE)

SOFTWARE VALIDATION TEST METRICS SHEET DRF#

Page: 1 of 1

Application Software Package: Revision:

Subsystem: Software Class:

Total Errors		Error Type	SFT
Major	Minor		SECTION
0	0	Data reference errors - errors that occur when data items are referenced improperly,	
0	0	Data declaration errors - errors resulting from conflicts between intended and actual usage,	
0	0	Computation errors - errors resulting from improper analysis or computational precision,	
0	0	Comparison errors - errors resulting from improper or imprecise condition expressions,	
0	0	Control flow errors - errors resulting from incorrect branching targets,	
0	0	Interface errors - errors resulting from improper passage of data between software modules,	
0	0	Input/Output errors - errors resulting from incorrect data formats or invalid interface specification	
0	0	Hardware Errors, and	
0	0	Hardware/Software interaction Errors.	
0	0	Task Interaction Errors	
0	0	Other Errors	
0	0	Totals	

Notes:

Enclosure 3

MFN 09-358

Affidavit

GE-Hitachi Nuclear Energy Americas LLC AFFIDAVIT

I, **David H. Hinds**, state as follows:

- (1) I am Manager, New Units Engineering, GE Hitachi Nuclear Energy ("GEH"), and have been delegated the function of reviewing the information described in paragraph (2) which is sought to be withheld, and have been authorized to apply for its withholding.
- (2) The information sought to be withheld is contained in Enclosure 1 of GEH's letter, MFN 09-358 Mr. Richard E. Kingston to U.S. Nuclear Energy Commission, entitled "*Licensing Topical Report NEDE-33226P, Revision 4, ESBWR – Software Management Program Manual*," dated May 29, 2009. The proprietary information in Enclosure 1, which is entitled "*GEH Nuclear Energy ESBWR – Software Management Program Manual, NEDE-33226P, Revision 4, May 2009 - Proprietary Version*," The proprietary information is enclosed within double brackets. [[This sentence is an example. ^{3}]]. Figures and large equation objects are enclosed in double brackets. Figures and large equation objects are identified with double square brackets before and after the object. In each case, the superscript notation {3} refers to Paragraph (3) of this affidavit, which provides the basis for the proprietary determination.
- (3) In making this application for withholding of proprietary information of which it is the owner or licensee, GEH relies upon the exemption from disclosure set forth in the Freedom of Information Act ("FOIA"), 5 USC Sec. 552(b)(4), and the Trade Secrets Act, 18 USC Sec. 1905, and NRC regulations 10 CFR 9.17(a)(4), and 2.390(a)(4) for "trade secrets" (Exemption 4). The material for which exemption from disclosure is here sought also qualify under the narrower definition of "trade secret", within the meanings assigned to those terms for purposes of FOIA Exemption 4 in, respectively, Critical Mass Energy Project v. Nuclear Regulatory Commission, 975F2d871 (DC Cir. 1992), and Public Citizen Health Research Group v. FDA, 704F2d1280 (DC Cir. 1983).
- (4) Some examples of categories of information which fit into the definition of proprietary information are:
 - a. Information that discloses a process, method, or apparatus, including supporting data and analyses, where prevention of its use by GEH's competitors without license from GEH constitutes a competitive economic advantage over other companies;
 - b. Information which, if used by a competitor, would reduce his expenditure of resources or improve his competitive position in the design, manufacture, shipment, installation, assurance of quality, or licensing of a similar product;

- c. Information which reveals aspects of past, present, or future GEH customer-funded development plans and programs, resulting in potential products to GEH;
- d. Information which discloses patentable subject matter for which it may be desirable to obtain patent protection.

The information sought to be withheld is considered to be proprietary for the reasons set forth in paragraphs (4)a. and (4)b. above.

- (5) To address 10 CFR 2.390(b)(4), the information sought to be withheld is being submitted to NRC in confidence. The information is of a sort customarily held in confidence by GEH, and is in fact so held. The information sought to be withheld has, to the best of my knowledge and belief, consistently been held in confidence by GEH, no public disclosure has been made, and it is not available in public sources. All disclosures to third parties, including any required transmittals to NRC, have been made, or must be made, pursuant to regulatory provisions or proprietary agreements which provide for maintenance of the information in confidence. Its initial designation as proprietary information, and the subsequent steps taken to prevent its unauthorized disclosure, are as set forth in paragraphs (6) and (7) following.
- (6) Initial approval of proprietary treatment of a document is made by the manager of the originating component, the person most likely to be acquainted with the value and sensitivity of the information in relation to industry knowledge, or subject to the terms under which it was licensed to GEH. Access to such documents within GEH is limited on a "need to know" basis.
- (7) The procedure for approval of external release of such a document typically requires review by the staff manager, project manager, principal scientist, or other equivalent authority for technical content, competitive effect, and determination of the accuracy of the proprietary designation. Disclosures outside GEH are limited to regulatory bodies, customers, and potential customers, and their agents, suppliers, and licensees, and others with a legitimate need for the information, and then only in accordance with appropriate regulatory provisions or proprietary agreements.
- (8) The information identified in paragraph (2) is classified as proprietary because it contains details of GEH's software design and licensing methodology. The development of the software management process was achieved at a significant cost to GEH.
- (9) Public disclosure of the information sought to be withheld is likely to cause substantial harm to GEH's competitive position and foreclose or reduce the availability of profit-making opportunities. The information is part of GEH's comprehensive BWR safety and technology base, and its commercial value extends beyond the original development cost. The value of the technology base goes beyond the extensive physical database and analytical methodology and includes development of the expertise to determine and apply the appropriate

includes development of the expertise to determine and apply the appropriate evaluation process. In addition, the technology base includes the value derived from providing analyses done with NRC-approved methods.

The research, development, engineering, analytical and NRC review costs comprise a substantial investment of time and money by GEH.

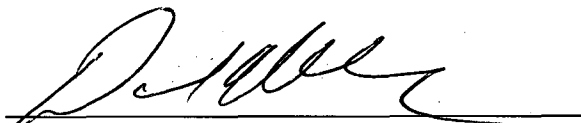
The precise value of the expertise to devise an evaluation process and apply the correct analytical methodology is difficult to quantify, but it clearly is substantial.

GEH's competitive advantage will be lost if its competitors are able to use the results of the GEH experience to normalize or verify their own process or if they are able to claim an equivalent understanding by demonstrating that they can arrive at the same or similar conclusions.

The value of this information to GEH would be lost if the information were disclosed to the public. Making such information available to competitors without their having been required to undertake a similar expenditure of resources would unfairly provide competitors with a windfall, and deprive GEH of the opportunity to exercise its competitive advantage to seek an adequate return on its large investment in developing and obtaining these very valuable analytical tools.

I declare under penalty of perjury that the foregoing affidavit and the matters stated therein are true and correct to the best of my knowledge, information, and belief.

Executed on this 29th day of May 2009.



David H. Hinds
GE-Hitachi Nuclear Energy Americas LLC