TABLE OF CONTENTS

1.0 PURPOSE, APPLICABILITY, AND ORGANIZATION....2
1.1 Chapter 21 Purpose and Applicability....2
1.2 Chapter 21 Use, Training, and Organization....4
1.3 Section SOFT-GEN Purpose and Applicability....9
2.0 SOFTWARE OWNERSHIP, IDENTIFICATION, AND DETERMINATION....12
2.1 Software Ownership/Software Identification (ML-1 through ML-3 only)....12
2.2 Software Determination....13
3.0 SOFTWARE DATA SHEET (SWDS) (ML-1 through ML-3 only)....15
4.0 CONFIGURATION MANAGEMENT (CM)....15
4.1. Develop and Maintain a Software Baseline (SWBL) (ML-1 through ML-3 only)....15
4.2. Control Software Changes....18
5.0 PERFORM PROBLEM REPORTING & CORRECTIVE ACTION (PR&CA)....20
6.0 PERFORM SOFTWARE (PROJECT) RISK MANAGEMENT....21
7.0 APPENDICES, ATTACHMENTS AND FORMS....22
APPENDIX A: CHAPTER 21 DEFINITIONS, RESPONSIBILTIES, AND ACRONYMS....23
APPENDIX B: CHAPTER 21 SO AND SRLM DECISION DIAGRAM (GUIDANCE)....32
APPENDIX C: CHAPTER 21 REFERENCE LISTING....34

ATTACHMENTS:
ATTACHMENT 1: CHAPTER 21 SUMMARY TABLE

FORMS:
SOFT-GEN-FM01: SOFTWARE DATA SHEET FORM (SWDS)
SOFT-GEN-FM02: SOFTWARE BASELINE FORM (SWBL)
SOFT-GEN-FM03: NON-SSC SOFTWARE CHANGE PACKAGE FORM (SWNCP)

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue as provisional document.	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Made chapter mandatory. GEN forms renumbered. Added template 3046, *Software Risk Register*; DOE-STD-1073, *Configuration Management.* Clarified applicability and added Nuclear Criticality Safety, review of supplier software error reports, modified file numbering scheme, Less-Than-Minor Change definition, other minor changes throughout.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

As with all LANL Engineering Standards, but especially with this new chapter, please contact the chapter POC with comments, issues, etc.

1.0 PURPOSE, APPLICABILITY, AND ORGANIZATION

1.1 Chapter 21 Purpose and Applicability

A. Purpose: Chapter 21 is the software plan for the Facility Conduct of Engineering program (FAC-COE), safety basis, and nuclear criticality safety activities. It provides reasonable assurance that in-scope software will consistently, compliantly, and efficiently satisfy its intended use. It describes the "how, when, and who" for implementing the software quality assurance (SQA) requirements[1] of:

1. DOE Order 414.1D, *Quality Assurance,* (hereafter O 414.1D),
2. ASME NQA-1-2008/NQA-1A-2009, *Quality Assurance Requirements for Nuclear Facility Applications, Part I and Part II* (hereafter NQA-1), and
3. DOE-STD-1073, *Configuration Management.*

B. Applicability:

1. For new and "modified" software, this chapter is required when software is within the scope of:

   a. PD340, *Conduct of Engineering for Facility Work*,

   b. PD110, *Safety Basis*, or

   c. SD130, *Nuclear Criticality Safety Program*

   "Modified" software includes Major Computer Program Changes (see Definitions). Both Major Computer Program Changes (including upgrades from non-safety to safety software) and software considered non-compliant[2] are subject to this chapter.

   Also included is Research and Development (R&D), programmatic and other engineering software that affects: (a) structures, systems, or components (SSCs) described in Chapter 3 of a Documented Safety Analysis (DSA)[3] and/or (b) adjacent SSCs[4].

2. The chapter defines and applies to two major types of software:

   a. System, structure, and component (SSC) software; and,

   b. Non-SSC software (see definitions), including "Simple and Easily Understood" software used in the design of SSCs.[5]

---

[1] Requirements are clarified for LANL application in SD330 *LANL Quality Assurance Program*. Chapter 21 differences relative to P1040 *Software Quality Management* (SQM) and P840-1 *Quality Assurance for Procurements* are authorized by QPA-VAR-2016-005.1 (EMRef-75).

[2] Compliant means meeting applicable DOE O 414.1D and NQA-1 SQA requirements, thus compliance with LANL P1040 or prior chapter revisions is also acceptable. When in doubt about compliance, assess against relevant requirements; Quality and Performance Assurance (QPA) may be able to assist.

[3] Criteria are from PD340. If safety/technical baseline is not affected and the work is R&D engineering work, then the work is conducted to PD370, *Conduct of Engineering for Research and Development [R&D].* If safety/technical baseline is not affected and the work is other than R&D engineering work, then this chapter may be used if use is consistent with the governing quality assurance program. Safety SSCs described in Chapter 3 of the DSA are Safety Class, Safety Significant, or Other Hazard Controls (OHCs).

[4] Adjacent SSCs are SSCs that are located adjacent to the safety or OHC SSCs in a nuclear facility such that changes to these SSCs could negatively impact the safety or mission of the activity. (adapted from DOE-STD-1073-2003, *Configuration Management).*

[5] See definitions in SOFT-GEN Appendix A, *Chapter 21 Definitions and Acronyms*. Includes safety and non-safety software. Also, see applicability of quality assurance controls summarized in SOFT-GEN Attachment 1, *Chapter 21 Summary Table* and further details in Chapter 21 sections.

C. Exclusions: This chapter does not apply to:

1. Personal productivity and collaboration software that does not provide calculation output.

2. Software covered by existing, compliant software quality management plans/processes.[6] The issuance of this chapter does not necessitate immediate replacement of those controls (grandfathering applies) unless/until a Major Change.

3. Uncomplicated software tools, including productivity aids that are adequately controlled through the design process. *Examples include unit conversion software, spike programs, equipment selection software, diagnostic software, and interpolation calculators.*

D. Implementation:

1. LANL: Once the need to adopt this chapter for a given software is identified, initiate performance feedback in the Performance Feedback and Improvement Tracking System (PFITS) per P322-4. The PFITS action must require the software to be compliant with this chapter by a due date agreed to by both Software Owner Responsible Line Manager (SRLM) and chapter Point of Contact (POC). [7]

   Should this chapter conflict with other LANL Conduct of Engineering direction, contact the Chapter 21 POC for direction.

2. Subcontractors (architect-engineers, consultants, etc.): Chapter 21 provides minimum requirements for software quality assurance (SQA) including both SSC and Non-SSC software.

   a. For Non-SCC software (e.g., safety design or analysis), the chapter includes basic reporting requirements to facilitate LANL's oversight of project-specific determinations, etc. *It may also be used as an example of a comprehensive approach when assessing Subcontractor SQA programs.*

   b. For SSCs with software being specified or designed, and for any software being delivered for LANL ownership and use, Chapter 21 requires the same documentation deliverables that LANL personnel provide[8] (to the extent the information is known in the design and construction phases; when not known, insert TBDs and deliver as "Draft").

      Unless otherwise indicated in subcontract documents, the construction Subcontractor serves as the Software Owner (SO) up to the point of turnover. Upon turnover, the SO responsibility transfers to the LANL system engineer. The FDAR is the SRLM and retains that responsibility throughout the software life cycle however may delegate activities through subcontract to the Subcontractor.

---

[6] However, a software summary as described in SOFT-GEN Section 2.1 is required for new projects, including when compliant SQA programs other than this chapter are used.

[7] Six months is believed appropriate for most software while some may justify longer. Example wording: "For [software application], develop, issue and follow the SQA documents required by Engineering Standards Manual Chapter 21 – Software."

[8] For SSC software: Use of chapter forms/document numbers provides standardization and retrievability of documents that LANL must maintain going forward. As necessary, documentation will be updated or produced by construction subcontractors and verified by LANL in the start-up/commissioning process in conjunction with the system engineer function.

c. Subcontractor quality assurance programs must meet the requirements of this chapter, comply with O 414.1D, and—where required by subcontract—NQA-1, including Part I and Part II, Subparts 2.7 and 2.14.

E. Associated Management levels (MLs)—rather than software type, category, or Software Risk Level (SRL) terms—are used throughout.[9] Where ML-1, ML-2, ML-3 or ML-4 software is written, this is the associated software ML.

See Form 2033 for an ML crosswalk (guidance table) to software types, categories, and software risk levels (SRLs).

F. Several of the concepts discussed above are depicted in the following figure:

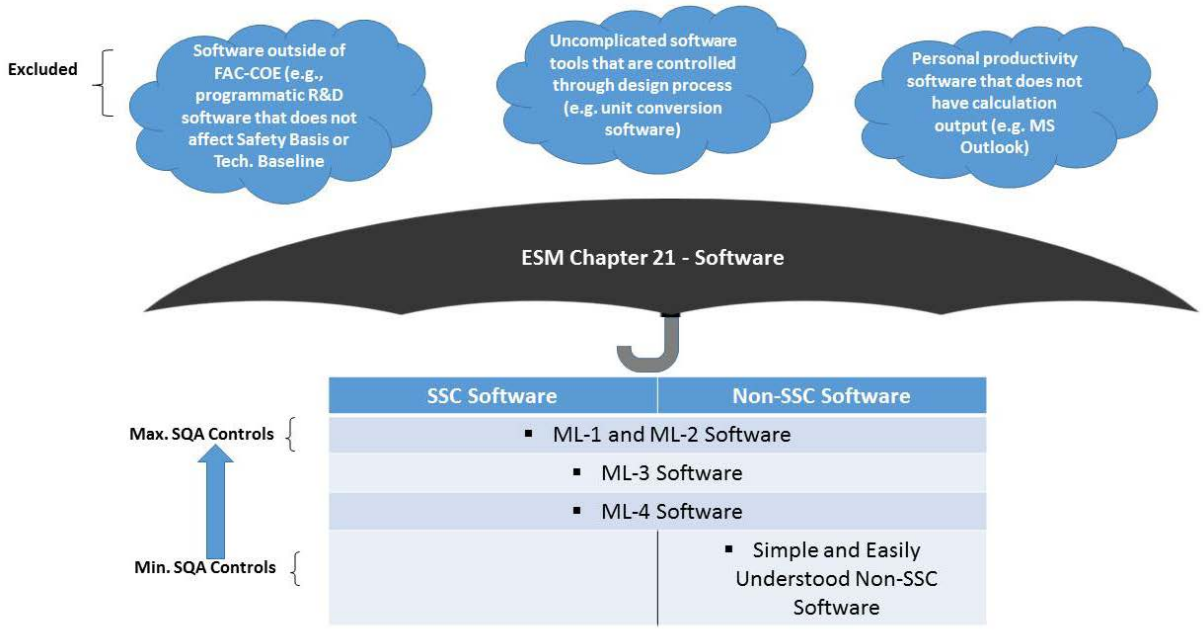


Figure 21.1-1 Chapter 21 Overview

1.2 Chapter 21 Use, Training, and Organization

A. Concept: Chapter 21 implements the SQA life cycle as shown in Figure 21.1-2, SQA Process Life Cycle.

[9] See Appendix A for definition. Promotes a common vernacular for users.

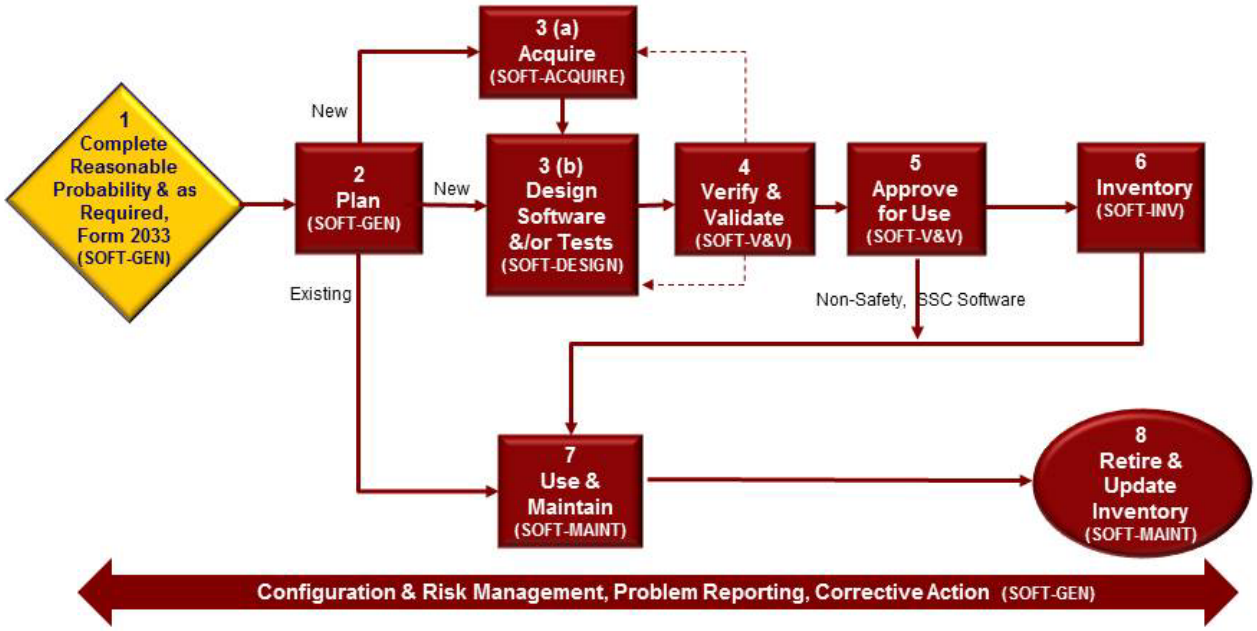


Figure 21.1-2 SQA Process Life Cycle

B. This chapter uses an umbrella concept to implement the SQA life cycle; previously presented Figure 21.1-1 depicts this concept at a high level, and there is more detail in Figure 21.1-3, *Chapter 21 Concept, Sections, and Deliverables*. That is, Chapter 21 serves as the software plan for software within its scope. Implementation of this chapter/plan will produce software-specific SQA implementation documentation (e.g., software baseline, software data sheet) for multiple software items. The software-specific documentation includes information that is unique[10] to each software program. Examples of software-specific information include the software technical requirements, software ownership, and software test plans.

C. Training:

1. Testing: For final SSC software testing in the operating environment that is used to verify the quality of ML-1 and ML-2 SSCs in Hazard Category II or III nuclear facilities, certify test personnel per applicable NQA-1 requirements.

2. LANL: Training requirements will be controlled by ESD Training Program Plan (ESD-PLAN-001, kept on the CoE T&Q webpage).[11] *Guidance: Live training has been developed for classroom training on this chapter.* For NQA-1 test personnel certification and details of maintaining certification, see P330-8, *Inspection and Test* and ESM Chapter 15 – *Commissioning*.

[10] Unique in that it cannot otherwise readily be addressed on a higher level (in this chapter proper). This centralized + distributed model ensures consistency of approach/rigor within user organizations while eliminating the production of standalone SQA plans that repeat boilerplate; it also utilizes a living, stand-alone documentation approach for those documents that are subject to change, easing revision of same without necessitating revision of an entire plan. Also, revision of Chapter 21 does not invalidate the approach or documentation associated with a previous chapter revision unless the revised chapter material specifically states this.

[11] Based on the JTA for Chapter 21 in ESD-SAT-006. For these and additional training requirements for implementing this chapter, use PD781-1, *Conduct of Training* and P343, Facility Engineering Training and Qualification Manual to establish and maintain them.

D. Organization: Chapter 21 is divided into six (6) sections that span the software life cycle: SOFT-GEN, SOFT-ACQUIRE, SOFT-DESIGN, SOFT-V&V, SOFT-INV, and SOFT-MAINT. These sections and a summary of deliverables are depicted in Figure 21.1-3, *Chapter 21 Concept, Sections, and Deliverables*.[12] Note, not all six sections will apply to every software used, and section usage may not be in sequential order. That is:

1. For all software, use SOFT-GEN, SOFT-V&V, SOFT-INV, and SOFT-MAINT.

2. For software that is acquired, also use SOFT-ACQUIRE.

3. For software that is designed (used interchangeable with developed), also use SOFT-DESIGN.

E. Definitions and Acronyms: See SOFT-GEN Appendix A, *Chapter 21 Definitions, Responsibilities, and Acronyms* for key definitions and acronyms. Defined terms are normally capitalized, set in quotes, or both in this chapter.

F. Work Summary: Refer to SOFT-GEN Attachment 1, *Chapter 21 Summary Table*, for a summary of deliverables for this entire chapter. Refer to the table(s) at the beginning of each section for a work summary of that section. These tables address both SSC software and Non-SSC software.

G. Italics: Throughout the ESM, italics are one method used to indicate guidance. Exception: When italics are used to highlight document titles.

[12] The 6 phases encompass all life cycle phases described by other industry documents in a simplified approach.

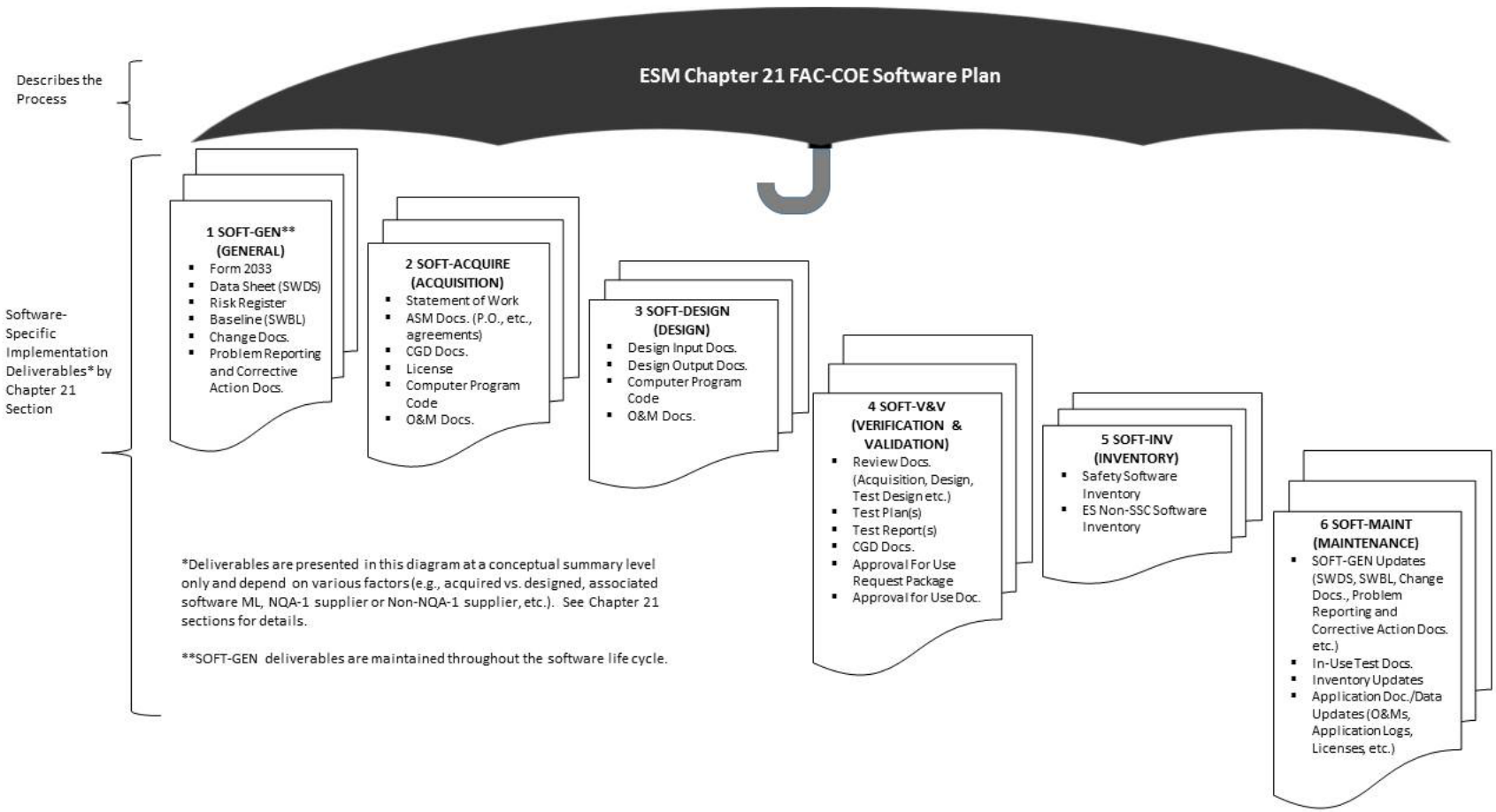


Figure 21.1-3 Chapter 21 Concept, Sections, and Deliverables

H. Roles, Responsibilities, Authorities, and Accountabilities: See SOFT-GEN Appendix B, *SO and SRLM Decision Diagram (Guidance)* when determining these key positions.[13]

I. "Or Equivalent": When this term is used, the SRLM determines equivalency. If questions arise regarding equivalency, the Chapter 21 Point of Contact (POC) will resolve.

J. Chapter References: See SOFT-GEN, Appendix C, *Chapter 21 Reference Listing* for hyperlinked references. Additional hyperlinks are provided in key locations throughout the chapter for convenience.

K. Forms:

1. LANL personnel: Endeavor to use chapter forms as-is and report issues and improvement ideas to the Chapter 21 POC.[14]

2. LANL subcontractors: For SSC software, use chapter forms or POC-approved Subcontractor forms to satisfy chapter requirements. For Non-SSC software, use either Subcontractor's own forms or integrate, adapt, and reformat the chapter forms. As determined by the Chapter 21 POC, either approach is acceptable so long as key functions, data, and approvals are retained.

3. In all cases, LANL Form 2033 must be used as-is.

4. Forms usage is summarized in Table 21.1-1 below:

Table 21.1-1 Form Usage Summary Guidance
See Appendix A for definitions of acronyms

Form	When Form is Initially Developed	When Form is Revised
2033	At earliest practical time but before acquisition and/or design	When the ***application*** of the software program significantly changes
SWDS	Earliest practical time but before acquisition and/or design as applicable	As required by SRLM to adequately manage software and/or per doc. control process[15]
SWBL	At earliest practical time but before acquisition and/or design	▪ Before acceptance testing ▪ After acceptance testing/SWAU ▪ After/with approved changes
SCP or SWNCP	When changes are needed	NA (usually not revised)
SWAU Docs. (e.g., SCP, Ch 15 Cx, PD115 Readiness Docs.)	▪ Prior to initial release (SWAU Form or Readiness Docs.) ▪ With changes (Change Docs.)	NA (usually not revised)

[13] The LANL Facility Design Authority (DA) is always authorized to act or sign for one or more DA Reps (FDARs).

[14] Forms are designated as samples which means minor aspects may be modified so long as all functions, key data, and approvals are retained. Alternatively, SRLMs may direct that other, comparable forms be used (with prior written permission of the chapter POC).

[15] Initial revision may have TBD for some parameters (e.g. use and maintain items) that are not known in the early planning stage and will be determined at a future date.

L. Chapter Revision: So long as SD330 compliance is maintained, the Chapter 21 POC is authorized to approve and issue revised forms, references, and SOFT-GEN Att. 1 independently of associated sections.[16]

1.3 Section SOFT-GEN Purpose and Applicability

A. Purpose: SOFT-GEN provides general software information and requirements to implement Chapter 21. These general requirements include how to: (a) identify software and make software determinations (e.g., Form 2033), (b) implement configuration management (e.g., baselines and software changes), (c) report problems/take corrective actions, and (d) perform software risk management.

B. See Table 21.1-2 for a summary of SOFT-GEN.

C. Applicability of SOFT-GEN: See the Chapter 21 applicability subsection above and the following:

1. For "Simple and Easily Understood" software that is used in the design of SSCs and that is individually verified per SOFT-V&V subsection 9, only the software identification and determination requirements of SOFT-GEN apply. If not individually verified per SOFT-V&V but rather pre-verified per SOFT-V&V subsection 6 or 7, then all requirements of SOFT-GEN apply.

   See other sections for applicability of the respective section for this software (e.g., SOFT-INV requires this type of software to be inventoried).

2. SOFT-GEN applies to subcontractors providing design agency or analysis services to LANL[17].

[16] Chapter 21 compliance to source document is determined by the Chapter 21 POC and periodically independently assessed to ensure compliance is retained. Other chapter section revisions follow the revision process described in ESM Chapter 1 – General, Section 100.

[17] SSC and Non-SSC software resulting from and/or used in the SSC design must be identified and software determinations completed as stated herein.

Table 21.1-2 Section SOFT-GEN Summary for SSC and Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		1, 2, 3	4	How	When	Who[2,3]	ESM Ch. 21 Ref.[4]
1	Identify software ownership/software and complete software determinations	R	Gr	▪ Identify SO and SRLM ▪ Identify software and develop initial and updated software summaries ▪ Determine reasonable probability ▪ Complete and retain Form 2033 ▪ *RLMs issue data call for software*	▪ Earliest practicable time ▪ Before software design, acquisition ▪ *Identify software at least annually for software data call*	▪ SO (D) ▪ SRLM (R, A) ▪ FDAR (R, A) ▪ *RLM (D) for data call*	SOFT-GEN, 2
2	Prepare Software Data Sheet (SWDS)	R	-	▪ Develop Software Data Sheet (SWDS), SOFT-GEN-FM01	▪ Earliest practical time ▪ Before design, acquisition as applicable ▪ Maintain throughout software lifecycle	▪ SO (D) ▪ SRLM (R, A)	SOFT-GEN, 3.0
3	Develop and maintain software baseline	R	-	▪ Develop software baseline (SWBL) using SOFT-GEN-FM02 ▪ Implement a baseline labeling system ▪ Maintain baseline to ensure the most recent approved configuration	▪ Earliest practical time ▪ Before/after acceptance testing ▪ Before software approval for use (SWAU) ▪ After approved changes ▪ For ML-1/ML-2 designed software, at major design activity	▪ SO (D) ▪ SRLM (R, A)	SOFT-GEN, 4.0

Table 21.1-2 Section SOFT-GEN Summary for SSC and Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		1, 2, 3	4	How	When	Who[2,3]	ESM Ch. 21 Ref.[4]
4	Control software changes (change control)	R	Gr	▪ For SSC software, prepare changes per AP-341-519 and/or AP-341-507-FM01; ▪ For Non-SSC software, prepare SWNCP per SOFT-GEN-FM03 ▪ Document changes (including comments in computer program code where possible) ▪ Make doc. changes per governing document control process ▪ Revise software baseline	▪ In a timely manner ▪ When required to ensure proper performance ▪ Per Table 21.1-4	▪ Per Table 21.1-4 or equivalent	SOFT-GEN, 4
5	Perform problem reporting and corrective action	R	Gr	▪ Before approval for use (SWAU), use P330-6, P322-4, less formal methods (e.g., bug lists, comments in program) Or Equivalent ▪ After SWAU, use P330-6, P322-4 or Equivalent	▪ Throughout software lifecycle	▪ SO (D) ▪ SRLM (R, A)	SOFT-GEN, 5
6	Perform software risk management	R	Gr	▪ Implement Chapter 21 ▪ Develop and maintain risk register (Template 3046 or equivalent); preferably as part of the SWDS	▪ Earliest practical time ▪ Before design, acquisition as applicable ▪ Maintain throughout software lifecycle	▪ SO (D) ▪ SRLM (R, A)	SOFT-GEN, 6

Notes:
[1] ML = Associated Management Level as per Form 2033. R = Required. Gr = Required but graded. "-" = Not required.
[2] D = Develop; R = Review; A = Approve.
[3] SO = Software Owner. RLM = Responsible Line Manager; SRLM = Software Responsible Line Manager. FDAR = Facility Design Authority Representative.
[4] Ref. = ESM Chapter 21 section name and subsection number (e.g., Section SOFT-GEN, Subsection 3, Software Identification and Determination).

2.0 SOFTWARE OWNERSHIP, IDENTIFICATION, AND DETERMINATION

2.1 Software Ownership/Software Identification (ML-1 through ML-3 only)

A. How

1. Using Appendix B, *SO and SRLM Decision Diagram (Guidance)*, identify Software Owners (SOs) and a Software Owner Responsible Line Managers (SRLMs).

Note: For ML-4 software, this chapter does not require software identification/summary; however, it may be invoked by the SRLM.

2. For Non-SSC (e.g., used for design and/or analysis), identify and summarize the software.[18]

   Develop a software summary includes the following: (a) software name, (b) software functional description (what it does), (c) software application (when/where/how used), (d) whether the computer program can be changed other than through replacement of the software (replacement includes software upgrades), and (e) whether the software is "Simple and Easily Understood" software that will be individually verified each time the software is used.

Note: **SOFT-GEN webpage references include an example software summary template**.

Guidance: SRLMs or Divisions should broadcast a software data call on a periodic (at least annual) basis to ensure software is identified and managed as required by this chapter.

3. For SSC software, identify and summarize the software as described above with the following clarification:

   a. Analyze the SSC to determine whether the SSC has software. Visually inspect and/or review vendor information (product specifications, catalog data, operations and maintenance manuals, etc.). Identify to a practical extent and to a level that the software can be controlled. *For example, for a control system module that has multiple pieces of software on a single module and the software cannot be modified other than through replacement of the entire module, then identifying the software to the module part number is appropriate.* Do not introduce a potential for adversely affecting manufacturer warranties, SSC damage, or unintended software changes by disassembling, etc.

4. Provide the summary to the LANL Software Responsible Line Manager (SRLM).[19]

B. When.

1. For Non-SSC software, provide the summary to the LANL SRLM at the earliest practicable time and prior to software acquisition or design (or use, if already on hand).

2. For SSC software, provide draft summaries no later than the 60% project design deliverables. When a required field on the software summary is not known, indicate TBD or similar term. Deliver the final summary (complete without TBDs)

[18] The software summary is not a controlled document or quality record. The responsible LANL SRLM uses this information to ensure that software determinations and subsequent required actions in this chapter are completed.
[19] LANL project engineer function must ensure review by appropriate SMEs and/or FDAR.

with the 90% detailed design deliverables. For a deferred[20] SSC design, include the final summary in the deferred design submittal(s).

C. Who. The Software Owner (SO) is responsible for software identification. See Appendix B for guidance on determining the SO. During the design phase, the design agency shall act for the SO. *SRLMs or Divisions should broadcast a software data call on a periodic (at least annual) basis to ensure software is identified and managed as required by this chapter.*

2.2 Software Determination

A. How. Use Table 21.1-3 and the text that follows it to complete software determinations.

1. Ask the following question as shown in Table 21.1-3:

Table 21.1-3 The Reasonable Probability Software Question
Excluding personal productivity and collaboration software that does not provide calculation output (e.g., e-mail software, presentation software)[1], is or will the software be used in connection with[3] the design, analysis, and/or operation of: ▪ a nuclear (including radiological) facility (Ref. LANL Nuclear Facility List and Conduct Of Operations Resources websites); ▪ an accelerator, live-firing range, biological hazard facility, high explosive facility, or moderate- or high- chemical hazard facility[2] as determined using Safety Basis Procedure (SBP)111-1, Facility Hazard Categorization and Documentation; or ▪ LANL's Essential Functions (EFs) as described in SEO-COOP-006, *Continuity of Operations (COOP) Plan.*
Notes: [1]Personal productivity and collaboration software that does not provide calculation output is non-safety, commercially controlled software and is excluded from this chapter. [2] From SBP111-1, a facility is defined as equipment, structure, system, process, or activity that fulfills a specific purpose except for utility distribution structures. [3]Use must be associated with ML-1, ML-2 or ML-3 items. The term "in connection with" broadly captures software that directly or indirectly is used in relation to design, analysis or operation. Operation includes SSC operation (e.g., valve actuation) and administrative operation (e.g., combustible-load-tracking software).

If the answer to the question is "yes" or "not sure" and the associated SSC ML is ML-1 through ML-3 or unknown, then complete Form 2033. If the associated SSC is ML-4, then do not complete Form 2033 unless required by the SRLM or Subcontract.

By default, all software is non-safety and commercially controlled (analogous to ML-4) unless there is a reasonable probability that the software could be safety software or risk significant software.

Guidance: Examples of commercially controlled, non-safety software include: Microsoft PowerPoint® presentation graphics program, Microsoft Outlook® messaging software and, when used in ML-4 applications, Caesar II, SASSI.

[20] Deferred refers to design performed after permit is granted (ref. IBC-2015 107.3.4 and 202 and ESM Chapter 16, IBC-GEN.

2. Complete Form 2033 using the instructions on the form. See form instructions for which revisions of the form may be used. If the software is planned to be used as "Simple and Easily Understood" software used in the design of SSCs (see definitions), then note it on the form.

Guidance on MLs, SRLs, and Type.: For Non-SSC software, use the highest ML of the SSC(s) being designed/analyzed. For SSC software, use ML of the SSC unless the software has a lower associated ML level than the SSC's highest ML level (In this latter case, the reason needs to be documented. This may necessitate an ML determination using AP-341-502 when the software may be a lower ML level than the system it supports.[21] Example: Flowmeter credited for confinement while its embedded software is not). In addition, in both cases, the 2033 software SRL and Type can be at a lower level than the associated SSC in some instances

Guidance on bundling: Form 2033 allows multiple software items to be included (often it is the support software), and SOFT-INV allows SSC software to be inventoried as a software collection (bundled); see SOFT-INV Section 3.0, A. 5. Bundling may work better if all software is the same ML level relative to Form 2033 and subsequent Ch. 21 requirements (may cause confusion if not). Example of where bundling might be considered: A control panel that has a PLC and a touchscreen and separate programs for these components, those separate programs, both ML-2, could be given a single SWID (and 2033, etc.)

3. Obtain a software identification number (SWID) in accordance with AP-341-402, *Engineering Document Management in Operating Facilities. A link to the SharePoint numbering utility is on the Ch 21 webpage.* Ensure the SWID is on the Form 2033. (Unless a SWID location is indicated on the form, place the SWID in the "Reference No." location on the form along with the record number). Ensure the SWID is on subsequent software documentation. Retain the completed Form 2033 as a record in accordance with the SRLM's governing records management process.

B. When.

1. Complete at the earliest practical time and before software design or acquisition. For SSC software, submit draft Form 2033 with the 60% detail design submittal. When a required field on the Form 2033 is not known, indicate TBD or similar term. Provide Form 2033 forms (complete without TBDs) with the 90% detailed design submittals. For a deferred SSC design, include the final Form 2033 in the deferred design submittal(s).

2. If the application of the software significantly changes, review the original determination and complete a new Form 2033 as required. *Guidance: It is recommended that Form 2033s be reviewed at least every three years or other time period that is commensurate with the associated risk.*

   Note: Changes made to the software (e.g., software version, SO, SRLM) do not require a new Form 2033 to be completed unless the change significantly affects the application. The software inventory must be reviewed and updated as required. See SOFT-INV.

[21] Current AP-341-502 excludes software but revision is planned; proceed with determination in the meantime if the associated ML cannot be inferred.

C. Who. The SRLM is responsible for software determinations. Form 2033 instructions require the SO to develop, and the SRLM and FDAR to review and approve the form.

3.0 SOFTWARE DATA SHEET (SWDS) (ML-1 THROUGH ML-3 ONLY)

A. How. Prepare a Software Data Sheet (SWDS). See SOFT-GEN-FM01, *Software Data Sheet Form (SWDS)*. One SWDS can be used for multiple installations. Process and retain the data sheet in accordance with the SRLM's governing document control/records management processes *(e.g., EDMS, and for ES-Div, associated Desktop Instruction).*

Ensure the correct SWID (obtained when completing Form 2033) is part of the SWDS record number and subsequent software documentation.

Note: Software data sheets are not required for ML-4 software; however, they may be used at the SRLM's discretion.

B. When. Complete a SWDS at the earliest practical time and before design and/or acquisition as applicable (e.g., at end of planning).[22] Revise the SWDS as required to maintain sufficient accuracy as determined by the SRLM, to adequately manage the software.

Note: *SRLM should review the data sheet at least every two years.*

C. Who. At a minimum, the SO develops and the SRLM reviews and approves the SWDS.

4.0 CONFIGURATION MANAGEMENT (CM)

A. Maintain software configuration management (CM) in accordance with this Subsection[23]. This section includes developing a baseline of configuration controlled items and the change control processes to make controlled changes to the baseline.

4.1. Develop and Maintain a Software Baseline (SWBL) (ML-1 through ML-3 only)

A. How.

1. Develop and maintain a software baseline.

From App. A definition, a baseline is a specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved change control process. (Ref. NQA-1).

Use SOFT-GEN-FM02, *Software Baseline Form (SWBL)*.

Note: A software baseline includes the computer program (code and [configuration] data) and the computer program documentation. For a mature

[22] Ref. IEEE 12207. At the end of the planning phase, SWDS is proof of a plan and management commitment to the project. Initial revision may have TBD for some parameters (e.g. use and maintain items) that are not known in the early planning stage and will be determined at a future date.

[23] See ANSI/IEEE Std. 828, *IEEE Standard for Configuration Management in Systems and Software Engineering* for guidance.

baseline, one may think of this as the information and computer program files that are needed to run the software for a specific application.

ML-4 software does not require a software baseline unless it is required by the SRLM.

2. For SSC software: Ensure the software baseline is consistent with the SSC technical baseline documents, as applicable. See *Engineering Standards Manual Chapter 1, Section Z10, General*; AP-341-616, *Technical Baseline Change During Design;* and AP-341-405, *Identification and Control of Technical Baseline, Variances, Alternate Methods, and Clarifications in Operating Facilities.*

   a. As applicable, ensure configurations (e.g., bounding set points) are established in accordance with AP-341-613, *Instrumentation Set Point Control.*

3. Define the baseline configuration items needed to run the program. Include, as applicable[24]:

   a. **Documentation** - software design requirements; instructions for computer program use including installation and test requirements; system description; test plans and results; characteristics derived from regulatory requirements and commitments; calculations and analyses; support software documentation.

   b. **Computer program(s)** - source, object, back-up, data,[25] configuration, and support software files needed to run the program. Configuration files include limits, settings, bounding set points, etc. As applicable, include the file/directory attributes (e.g., for Window systems and servers) or authorities (e.g., for Unix/Linus systems). These are the required access settings for reading, writing and executing for reading, writing and executing for users, owners, etc.

4. Develop and control the software baseline in accordance with the SRLM's document/electronic document control process *(e.g., EDMS)*. The SRLM's approved document control process is used, as a minimum, to control both the documents and the computer program files that comprise the baseline. The document control process must ensure only authorized access and changes are made to software baselines. Ensure the correct SWID (obtained when completing Form 2033) is part of the SWBL record number and subsequent software documentation.

5. Update the software baseline after software changes. Ensure that the baseline defines the most recent approved software configuration.[26] Ensure work is performed on computers, servers, etc. using documentation and computer

[24] Applicability is the determination that the activities and processes are actually used. For example, if the work scope does not include or contribute to design, then design requirements do not apply and the related design procedures are not used [Ref. SD330, *LANL Quality Assurance Program*].

[25] This is the data needed to run the program. This can include data such as limits, bounding set points, and databases.

[26] In the event the in-use computer program fails, becomes corrupted, or other problems occur, the baseline may be used to re-establish operation

programs that match (mirror) the baseline as described and controlled in the SRLM's document control process.

6. Identify the associated hardware needed to run the computer program in the system design description document(s) or directly in the baseline.

7. When "Support Software" (including software tools) is used for new or modified software, one must evaluate, review, accept for use and place it under configuration control as well. Note however, that software tools that do not affect the performance of the software need not be placed under configuration control. Identify the software/tool in the baseline and manage the configuration. Changes to software tools must be evaluated for impact on the software product to determine the level of reviews and retesting that will be required. Vendor-supplied software used with hardware to configure control systems is an example of support software.

8. Implement a baseline labeling system in accordance with the following:

   a. Uniquely identify each configuration item (documentation or computer program).

   b. Identify changes to configuration items by revision.

   c. Provide the ability to uniquely identify each configuration of the revised software available for use.

   d. For baseline document labeling, use the SRLM's approved document numbering system (e.g., AP-341-402, *Engineering Document Management in Operating Facilities).*

   e. Include baseline label information within the computer source code where possible. For each logical block or class, include a brief description of its function, the name of the person writing the description and the date the description was added. For code changes following SWAU, update the header comments with the revision number (e.g., 1.1, 1.2...) and the SCP number driving the change.

   f. For off-the-shelf software (OTS) or other (existing) software with a compliant labeling configuration, use the OTS or existing software labeling configuration.

   g. For other software not described in (f), use Table 21.1-4, *Computer Program File Labeling System*, or as approved by the Chapter 21 POC.

   h. Ensure the correct SWID (normally obtained when completing Form 2033) is integrated into the baseline labeling where possible.

Table 21.1-4 Computer Program File Labeling System			
Prior to Approval for Use[1]		After Approval for Use	
Major Change[2]	Minor Change[2]	Major Change	Minor Change
A, B, C…	A.1, A.2, A.3…	1, 2, 3…	1.1, 1.2, 1.3…

[1] Software approval for use (SWAU) constitutes that the software requirements have been satisfied (including testing, user's manuals, etc.) and the software is ready to be used in it intended operating environment.

[2] See definitions of Major Change and Minor Change in App A, Definitions.

B. When.

1. For ML-1 and ML-2 designed software, establish software baselines at the completion of each major software design activity.

2. Establish software baselines as soon as practical. At the latest, establish before formal acceptance testing/commissioning. Pre-acceptance testing may be performed to support verification of the baseline prior to (final) acceptance testing. Maintain baselines (configuration items) under configuration management until software retirement.

C. Who. The SO prepares the baseline. The SRLM and others as required in the governing document control process, review and approve the software baseline.

4.2. Control Software Changes

A. How.

1. ML-1 and ML-2 Safety Software: For computer program changes (revisions) to software, use SOFT-V&V to V&V the software change and/or commercial grade dedication (CGD), e.g., per *AP-341-703*, to dedicate the change. Unless a more conservative approach is required by the LANL SRLM, use the following graded approach for dedicating software changes:

   a. Dedicate Major Changes (see Definitions).

   b. Dedicate Minor Changes (see Definitions) if the changes affect the critical characteristic(s) that pertain to the functionality of the computer program and as applicable, the associated SSC.

2. For all ML levels, see Table 21.1-5 and apply the following to Major and Minor software changes:

   a. For designed software, institute design control measures commensurate with those applied to the original design.

   b. Document changes and include the following in the change documentation:

      1) a description of the change;

      2) rationale for the change;

      3) identification of affected software baselines, including documents to be updated; and

      4) evaluation and approval of the change (see SOFT-V&V for evaluation methods/criteria).

Table 21.1-5 SOFT-GEN Software Change Summary by Software Management Level (ML)
(This is a summary only and does not include all requirement details. See text for details.)

For the following types of software changes	Provide the following deliverable	SSC Software ML[1]				Non-SSC Software ML[1]			
		1	2	3	4	1	2	3	4
Any change made before or after approval for use (SWAU). Includes Less than Minor, Minor, and Major changes.	Documented comment in the computer program code where feasible	R	R	R	R	R	R	R	R
Minor and Major Changes made during design implementation (e.g., fielding or startup) and before SWAU	AP-341-519 change documents[2]	R	R	Gr	Gr	-	-	-	-
Minor and Major Changes made once software is baselined	Revised baseline[3]	R	R	R	-	R	R	R	-
Minor and Major Changes made after SWAU	SSC: Software Change Package (SCP) Form AP-341-507-FM01[4]	R	R	Gr	-	-	-	-	-
	Non-SSC: SWNCP Form, SOFT-GEN-FM03	-	-	-	-	R	R	Gr	-

[1] See Form 2033 for a crosswalk between MLs and software type, category and software risk levels. R = Required. Gr = Required but graded. "-" = Not required.

[2] Includes Field Change Notices (FCNs), Field Change Requests (FCRs), and Design Revision Notices (DRNs) as described in AP-341-519 unless explicitly superseded by another administrative procedure.

[3] Use SOFT-GEN-FM02, *Software Baseline Form (SWBL)*. Include documents and computer program files.

[4] See ESM Chapter 15, *Commissioning* and AP-341-801, *Post Modification/Post Maintenance Testing* for related testing requirements of SSC software changes.

c. Perform appropriate verification activities and validation (V&V) of changes (see SOFT-V&V). Perform appropriate acceptance testing for the change.

d. Incorporate the changes in documentation, and maintain traceability of the change to the software design requirements.

   1) Include the (1) initiation, evaluation, and disposition of a change request, (2) control and approval of changes prior to implementation, and (3) requirements for retesting (e.g., regression testing) and acceptance of the test results.

e. Perform the following:

   1) maintain a current status of configuration items and control configuration changes until they are incorporated into the approved product baseline;

   2) maintain the status of changes that are proposed and approved, but not implemented; and

3) provide notification of this information to affected organizations.

f. For changes to documentation only, use the governing document control process.

g. Implement the processes described and/or referenced in this chapter to ensure that interfaces are controlled/managed such that unintended consequences do not occur.

h. As applicable, ensure bounding configuration changes are established (Ref. AP-341-613, *Instrumentation Set Point Control)*.

i. Update the software baseline.

3. For less than minor computer program changes, ensure they are made by a competent individual knowledgeable in the software. If feasible, add a brief description, name of who is making, justification that the change does not modify performance functions (all MLs), and the date that the less than minor change is made in the computer program code at the time the change is made. *If it is not feasible to document the less than minor change in the code, then document in associated operational logbooks, etc.*

B. When. Control changes per Table 21.1-5 in a timely manner and when required to ensure proper performance.

C. Who.

1. For SSC software, see AP-341-519 for design phase and AP-341-507 for operations phase unless explicitly superseded by another procedure. For Non-SSC software, see SOFT-GEN-FM03, Non-SSC Software Change Package Form (SWNCP).

2. The organization responsible for the original software design and others as deemed necessary by the SRLM must evaluate, and approve software changes unless an alternate organization has been given the authority by the SRLM to approve the changes.

3. As determined by the SRLM, only those that are knowledgeable in the computer program code may make changes to the computer program code.

5.0 PERFORM PROBLEM REPORTING & CORRECTIVE ACTION (PR&CA)

A. How.

1. For ML-1 through ML-4 software, perform software problem reporting and corrective action throughout the software life cycle. For software where suppliers provide/post errors, review errors/notifications at a frequency commensurate with the risk and frequency of use. *A review of problem reports prior to use in calculations and/or a minimum of once per year is recommended.*

2. For ML-4 software, less-formal methods (e.g., bug lists, use of comment fields within the computer program, software-tailored tools such as TeamForge and trac) or formal methods (see below) may be used throughout the software lifecycle.

3. For ML-1 through ML-3 software, less-formal methods may be used through Approval for Use (SWAU). After (SWAU), formal methods must be used. Individually or in combination, P330-6, *Nonconformance Reporting*, P322-4,

Issues Management, or equivalent process as determined by the SRLM, must address the following:[27]

a. Describe the evaluation process for determining whether a reported problem is an error (see definitions) or other type of problem (e.g., user mistake).

b. Define the responsibilities for disposition of problem reports, including notification to the originator of the results of the evaluation.

c. When a problem is determined to be an error (see definitions), provide, as appropriate, for

1) how the error relates to appropriate software engineering elements;

2) how the error impacts past and present use of the computer program;

3) how the corrective action impacts previous development activities; and

4) how the users are notified of the identified error, its impact, and how to avoid the error, pending implementation of corrective actions.

d. Problem reporting and corrective action must include methods for documenting, evaluating, and correcting software problems. *This should include an estimate of the level of effort and corrected software release date.*

e. Conditions adverse to quality shall be identified promptly and corrected as soon as practicable.

f. In the case of a significant condition adverse to quality, the cause of the condition shall be determined and corrective action taken to preclude recurrence.

g. The identification, cause, and corrective action for significant conditions adverse to quality shall be documented and reported to appropriate levels of management.

h. The verification of completed corrective actions.

B. When. Perform throughout the software life cycle.

C. Who. The SO performs software problem reporting and corrective action on behalf of the SRLM. The SRLM is responsible to ensure it is performed as required by the associated procedure (e.g., P330-6 on NCRs).

6.0 PERFORM SOFTWARE (PROJECT) RISK MANAGEMENT

A. How.

1. Software risk management as required by O414.1D focuses on the risks to successfully complete the software project; it does not focus on the risks of potential failure of the software.[28] Software risk management applies to all phases of the software project life cycle. Achieve overall software risk management through implementation of the NQA-1 based, systematic processes of this chapter and the supporting quality assurance program. *That is, successful implementation of this chapter should result in successful completion of a software project. The*

[27] The SWDS may be used to describe an equivalent as long as the PR&CA requirements are satisfied.
[28] Based on DOE G 414.1-4, *Safety Software Guide for Use with 10 CFR 830 Subpart A, Quality Assurance Requirements, and DOE O 414.1C, Quality Assurance.*

SRLM should determine if additional software project-level software Risk Management is needed for ML-4 software (e.g., large and important projects).

2. For ML-1 through ML-3 software, address software-specific risks (risks specific or unique to the software implementation). Manage software-specific risks by (a) assessing and (b) controlling the risks.

   a. With a graded approach approved by the SRLM and Chapter POC, implement risk management based on industry accepted methods such as those described in DOE SQAS21.01.00-1999*, Software Risk Management–A Practical Guide;* DOE G413.3-7, *Risk Management Guide; and/or* DOE G 414.1-4, *Safety Software Guide for Use with 10 CFR 830 Subpart A, Quality Assurance Requirements, and DOE O 414.1C, Quality Assurance*.

3. *Risk assessment includes risk identification, analysis, and prioritization to ensure that the necessary resources are available to mitigate them. Risk control includes risk resolution and tracking.*

4. Resolve risks using risk avoidance, mitigation, and/or transference.

5. Document and maintain software-specific risks on a risk register (also referred to as risk list) in the SWDS of if desired, in a separate document. Use Template 3046, Software Risk Register from the LANL Forms Center or equivalent. Attach the risk register to the SWDS is the best practice; create a unique SWRR number only if not attaching.

B. When. Develop a SWRR at the earliest practical time, before design/acquisition as applicable, and perform risk management throughout the software life cycle.

C. Who. The SO performs software risk management on behalf of the SRLM. The SRLM is responsible to ensure it is performed, and reviews and approves the risk register as part of the SWDS.

7.0 APPENDICES, ATTACHMENTS AND FORMS

Appendices:
- APPENDIX A: Chapter 21 Definitions, Responsibilities, and Acronyms
- APPENDIX B: Chapter 21 SO and SRLM Decision Diagram (Guidance)
- APPENDIX C: Chapter 21 Reference Listing

Attachments:
- ATTACHMENT 1: Chapter 21 Summary Table

Forms:
- SOFT-GEN-FM01: Software Datasheet Form (SWDS)
- SOFT-GEN-FM02: Software Baseline Form (SWBL)
- SOFT-GEN-FM03: Non-SSC Software Change Package Form (SWNCP)

APPENDIX A: CHAPTER 21 DEFINITIONS, RESPONSIBILTIES, AND ACRONYMS

Only key definitions are listed. See PD340, *Conduct of Engineering for Facility Work*; SD330, *Los Alamos National Laboratory Quality Assurance Program*; and documents referenced therein for additional definitions.

Table 21.1-A1 Chapter 21 Definitions and Responsibilities

Item	Definition/Responsibility
acceptance testing, also known as software validation	The process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specified requirements, and, to identify differences between expected and actual results in the operating environment. (Ref. NQA-1).
Acquired software	Software that is generally supplied through basic procurements, two-party agreements, or other contractual arrangements. Acquired software includes off-the-shelf (OTS) software such as operating systems, database management systems, compliers, software development tools, and commercial calculational software and spreadsheet tools. Downloadable software that is available at no cost to the user (referred to as freeware) is also considered acquired software. (Based on DOE G 414.1-4).
Acquired Non-SSC software	Non-SSC software that is acquired software where the code cannot be changed other than through replacement. This may also be referred to as Read-Only Non-SSC software. Replacement includes replacement with a subsequent software version or upgrade. (Definition developed for this chapter).
Administrative controls	Administrative controls mean the provisions relating to organization and management, procedures, record keeping, assessment, and reporting necessary to ensure safe operation of a facility. (Ref. 10CFR 830)
Associated Management Level (ML)	The highest management level (ML) of an SSC that is associated with the software function. The software function must be required for, and/or the software failure will have an adverse effect on the SSC. (adaptation of Form 2033 guidance)
baseline	A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved change control process. (Ref. NQA-1). ***Note:*** See SOFT-GEN-FM02, *Software Baseline Form (SWBL)*. A software baseline includes the computer program (code and [configuration] data) and the computer program documentation. In layman's terms, one may think of this as the information and computer program files that are needed to run the software for a specific application.
Bounding set points	Bounding set points are those that are bounding or limiting values required by or needed to satisfy safety basis requirements, protect equipment/systems from operational damage, or other limiting values for the proper intended operation of the software. Changes to operational set points within the minimum or maximum values would not constitute a change, but rather would be considered operational use of the software. Example: Bounding set points may be minimum or maximum pressure or tank level values.
Commercially Controlled (CC) software	As determined using Form 2033, *Safety/Non-Safety Software Determination, Categorization, and Software Risk Level (SRL)*, software that is not, or will not be safety software or risk significant software. Such software may be acquired (including off the shelf (OTS) software, freeware, or designed software. Examples: Personal productivity software (e.g., Microsoft PowerPoint, Oracle Project Primavera, MS Outlook, etc.); typical business accounting systems, facility personnel comfort temperature control systems. (Based on P1040).
computer program	A combination of computer instructions and data definitions that enables computer hardware to perform computational or control functions. (Ref. NQA-1).

Table 21.1-A1 Chapter 21 Definitions and Responsibilities	
Item	**Definition/Responsibility**
computer program listings	A printout [e.g., pdf] or other human readable display of the source and, sometimes, object statements that make up a computer program. (Ref. ISO/IEC/IEEE 24765).
configuration item	A collection of hardware or software elements treated as a unit for the purpose of configuration control. (Ref. NQA-1).
configuration management	The process of identifying and defining the configuration items in a system (i.e., software and hardware), controlling the release and change of these items throughout the system's life cycle, and recording and reporting the status of configuration items and change requests. (Ref. NQA-1).
control point	A point in the software life cycle at which specified agreements or control (typically a test or review) are applied to the software configuration items being developed, e.g., an approved baseline or release of a specified document or computer program. (Ref. NQA-1).
Critical software	Software whose proper performance is critical to the expected performance of a safety SSC, a defense-in-depth SSC, or the safety of the nuclear facility. (Ref. DOE-STD-1073).
design analyses	Calculations and/or experiments associated with design. (Based on NQA-1).
designed (or developed) software	Software that is designed or developed for a specific (custom) application. It may be developed by DOE or one of its Management and Operator contractors or contracted with a software company through the procurement process. ***Note:*** Includes the following from DOE G 414.1-4. (a) custom-developed software, (b) configurable software, and (c) utility calculation software. (Definition developed for this chapter and based on DOE G 414.1-4).
Designed Non-SSC software	Non-SSC software where the computer program can be changed other than through replacement. Replacement includes replacement with a subsequent software version or upgrade. (Definition developed for this chapter).
Designed SSC software	SSC software where the computer program can be changed other than through replacement of the computer program and/or the associated SSC. (Definition developed for this chapter).
engineered (engineering) control	Controls that eliminate or reduce exposure to a hazard through use or substitutions of engineered machinery or equipment. (Based on DOE G 450.4-1C).
Engineering Services Software Coordinator	Individual assigned to maintain the software inventory for ES Division, and to perform other duties as assigned (e.g., per desktop instruction).
error	An error is a condition deviating from an established baseline including deviations from the current approved computer program and its baseline requirements. (Ref. NQA-1). An error is something which requires a software change (major or minor). Examples of errors include (a) if a computer program used for design of a structural member provides incorrect design output, and (b) if a computer program turns on heating instead of cooling at high temperature settings.
Firmware	The combination of a hardware device, computer programs, and data that reside as read-only software on that device. The firmware (sometimes referred to as embedded software) can perform very limited functions such as keypad controls, or can provide significant function and control capabilities for control rod drives or safety systems. (Ref. NQA-1).
freeware	Software that is available for use at no cost or for a nominal, usually voluntary fee. (Ref. Merriam-Webster Dictionary).

Table 21.1-A1 Chapter 21 Definitions and Responsibilities	
Item	**Definition/Responsibility**
function	A function is a task that must be performed. A function statement describes the capability necessary for a facility, system, or component to fulfill its mission. Describe a function using verb/noun combination such as "filter particulate." A function transforms inputs to desired outputs. In the example of "filter particulate," the function transforms an input fluid containing particulates into two outputs, the fluid without particulates and the particulates. A function describes what must be done, not how. Every function has at least one requirement associated with it.
hazards controls	Hazard controls mean measures to eliminate, limit, or mitigate hazards to workers, the public, or the environment, including: (1) Physical design, structural, and engineering features; (2) Safety structures, systems, and components; (3) Safety management programs; (4)Technical safety requirements; and (5) Other controls necessary to provide adequate protection from hazards. (Ref. 10CFR830).
Less-Than-Minor Computer Program Change	A change that is not a major or minor computer program change and: ▪ adds, deletes, and/or modifies ML-4 performance function code, ▪ adds, deletes, and/or modifies code that does not modify a Performance Function (all MLs), or ▪ imparts changes without adding, deleting or modifying design and/or analysis output values (all MLs). Examples: Modify code to increase the ramp time on an ML-4 softstart pump. Install security patch/service pack updates. An OTS software patch that includes a code change to prevent a screen from "freezing" or loading slowly (all MLs). Add/modify code clarifying notes (all MLs). Modify code to produce multiple reporting formats (all MLs). (Definition developed for this chapter).
Major Computer Program Change	A change that: ▪ the Software Responsible Line Manager (SRLM) or computer program supplier designates as a Major Change, ▪ adds or deletes an ML-1, ML-2 or ML-3 SSC "Performance Function" (including bounding set point changes), ▪ modifies ML-1 or ML-2 SSC performance function code, excluding clarifying notes, ▪ adds, deletes or modifies design and/or analysis output values of ML-1, ML-2 or ML-3 calculations, ▪ recodes to another language, or ▪ modifies a significant number of lines of code. ***Note:*** A Major OTS computer program change is often indicated with increment increase in version number (e.g., change from version 1 to 2). An evaluation of the software however, is required to determine whether the version release is a Major Change. Examples: A change from Delta V control system software from version 7.0 to version 8.0. A change that adds code to implement an interlock functional performance requirement that an ML-3 laser system cannot be activated until area doors are locked. A change that modifies code on ML-2 ventilation system backdraft damper so that damper closure does not slam shut and potentially damage the damper assembly. A change in the algorithm or databased used for calculating the water flow rate in an ML-3 fire protection piping system design. A change in coding language from C to C++. A version change where 40% of the lines of code were modified. (Definition developed for this chapter).
Major Document Change	A document change that is not a minor document change. A major document change includes revisions, changes, or modifications to a document (e.g., procedure, work instruction, drawing, etc.) which impact the effective implementation of the requirement(s). (Based on P1020-2).

Table 21.1-A1 Chapter 21 Definitions and Responsibilities	
Item	**Definition/Responsibility**
Minor Computer Program change	A change that is not a major computer program change and: ▪ adds or deletes an ML-4 SSC Performance Function (including bounding set point changes), ▪ modifies ML-3 SSC Performance Function code, excluding clarifying notes, or ▪ adds, deletes or modifies design and/or analysis output values of ML-4 calculations ***Note:*** A minor OTS software change is often indicated with a fractional increase in version number (e.g., 1.1 or 1.01). An evaluation of the software however, is required to determine whether the version release is a Minor Change. Examples: Add code to implement automatic pump shut-off performance requirement on ML-4 sump low-level alarm. Modify code to fix a coding error on an ML-3 heating/cooling system so that cooling, rather than heating activates at high temperatures. Change the algorithm for calculating the current that flows in an ML-4 electric power system under abnormal conditions. (Definition developed for this chapter).
Minor Document Change	A document change, as defined by the governing document control program, that includes but is not limited to inconsequential editorial corrections, grammatical and spelling changes, organizational name and acronym changes, and similar type changes. (Based on P1020-2).
Non-NQA-1 qualified supplier	A supplier that did not develop and maintain the software in accordance with an NQA-1 quality assurance program. (Definition developed for this chapter).
model	Simplifications of the real world constructed to gain insights into select attributes of a particular physical, biological, economic, engineered, or social system. (Ref. EPA/100K-09/003, *Guidance on the Development, Evaluation, and Application of Environmental Models*.)
Non-safety software	As determined using Form 2033, software that is not otherwise determined to be safety software. Non-safety software includes risk significant and commercially controlled software. (Ref. P1040).
Non-SSC software	Software used in design, analysis and/or for administrative control. This software does not physically monitor and/or control SSCs. Examples: Piping system design/analysis software (CAESAR II®), fire protection system design software (SprinkCAD), area lighting calculation software, spreadsheets used to perform structural load calculations, safety analysis software used to perform dispersion modeling, software used to track facility combustible loading, and software used to track Technical Safety Requirement (TSR) implementation. (Definition developed for this chapter).
Otherwise-Acquired Software	Software that was not acquired, developed and/or maintained in accordance with an NQA-1 quality assurance program. This software may be from entities internal to LANL entities external to LANL (e.g., other DOE sites, U.S. EPA, etc.) This includes existing software (also referred to as in-use or legacy software).
operating environment	A collection of software, firmware, and hardware elements that provide for the execution of computer programs. (Ref. NQA-1)). It is also the location and conditions (environment) where the software will be used or operated to meet is intended function. (Based on P330-8).
performance function	A function that is required to satisfy item performance. (Definition developed for this chapter).
performance function code	The computer program language (code) that is required to satisfy item performance. (Definition developed for this chapter). The performance function code is only those lines of code that affect the performance function.

Table 21.1-A1 Chapter 21 Definitions and Responsibilities	
Item	**Definition/Responsibility**
Read-Only SSC software	SSC software where the computer program cannot be changed other than through replacement of the computer program and/or the associated SSC. (It is sometimes referred to as embedded software or firmware; however, for this Chapter, the term Read-Only SSC software is used.) ***Note:*** Read-only software includes software where limits and/or set points can be configured (e.g., via keypad entries) without modifying the computer program (code). (Definition developed for this chapter).
Regression testing	Selective retesting to detect errors introduced during modification of the computer program or to verify that the modified computer program still meets its specified requirements. (Ref. NQA-1).
Risk Significant (RS) software	Software that is, or will be, used for any of the purposes for which safety software is used (see Form 2033), only such purposes are in or for an accelerator, live-fire range, biological hazard facility, explosive hazard facility, or a moderate- or high- hazard chemical facility; or failure of the software would prevent LANL from performing Essential Functions as described in SEO-COOP-006, *Continuity of Operations (COOP) Plan*. Commercially controlled software is not risk significant software. (Ref. P1040).
Safety and Hazard Analysis and Design Software (SHADS)	Safety software that is used, or will be used, to classify, design, or analyze nuclear (including radiological) facilities. This software is not part of an SSC, but helps to ensure the proper accident or hazards analysis of nuclear (including radiological) facilities or an SSC that performs a safety function. This is Non-SSC software, safety software, and is categorized as SHADS. (Ref. P1040).
Safety Management and Administrative Controls Software (SMACS)	Safety software that performs, or will perform, a hazard control function in support of nuclear (including radiological) facility radiological Safety Management Programs (SMPs) or TSRs; and/or this is software that performs, or will perform, a control function in support of a nuclear (including radiological) facility necessary to provide adequate protection from nuclear (including radiological) facility radiological hazards. It supports eliminating, limiting, or mitigating nuclear hazards to workers, the public, or the environment as addressed in 10 CFR 830, 10 CFR 835, *Occupational Radiation Protection*, and the Department of Energy Acquisition Regulation (DEAR) Integrated Safety Management System (ISMS) clause 48 CFR 970.5223-1, *Integration of Environment, Safety, and Health into Work Planning and Execution*. (Ref. O414.1D).
Safety software	Software that includes any of the following: SSS, SHADS, or SMACS. Both SSC software and Non-SSC software can be safety software. (Ref. P1040).
Safety System Software (SSS)	Safety software for a nuclear (including radiological) facility that performs, or will perform, a safety function as part of an SSC and is cited in either (a) a DOE-approved documented safety analysis, or (b) an approved hazard analysis per DOE P 450.4A, *Integrated Safety Management Policy* and 48 CFR 970-5223-1, *Integration of Environment, Safety, and Health into Work Planning and Execution.* This is SSC safety software and is categorized as SSS. ***Note:*** References implemented at LANL as described in PD110, *Safety Basis*. See DOE-approved documented safety analyses at LANL Safety Basis Document Lists (SBDLs). Analyses include Documented Safety Analyses (DSAs), Preliminary Documented Safety Analyses (PDSAs), Bases for Interim Operations (BIOs), etc. (Based on P1040).
Simple and Easily Understood (Non-SSC) software	Software that satisfies the following criteria: a. The software is used in the design of SSCs; b. The results of the computer program can be easily confirmed through hand calculations; c. A person technically qualified in the subject can review and understand the program and the supporting calculations; and, d. The software can be individually verified with each use (e.g., calculation). (Based on NQA-1).

Table 21.1-A1 Chapter 21 Definitions and Responsibilities	
Item	**Definition/Responsibility**
software	Computer programs and associated documentation and data pertaining to [needed for] the operation of a computer system. (Ref. NQA-1).
software approval/ approved for use (SWAU)	An approval that constitutes that the software requirements have been satisfied (including installation and operating instructions), and the software is ready to be used in the intended operating environment. (Ref. P1040).
software change	A software change is an addition, deletion and/or modification to software. (Definition developed for this chapter).
Software Coordinator	Individual(s), designated by division management, providing coordinating and/or administrative functions in support of chapter compliance (e.g., inventory and associated reporting). *ES Division has an ES-Div Software Coordinator. ES-Software@lanl.gov*
software design requirement	A requirement that impacts or constrains the design of a software system or software system component. (Based on ISO/IEC/IEEE 24765).
software engineering	(a) the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software (b) the study of approaches as in (a) (Ref. NQA-1)
software life cycle	The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and, sometimes, retirement phase. These phases may overlap or be performed iteratively, depending on the software development approach used. (Ref. NQA-1)
software design verification:	The process of determining if the product of the software design activity fulfills the software design requirements. (Ref. NQA-1).
software engineering elements	(a) software acquisition method(s) for controlling the acquisition process for software and software services; (b) software engineering method(s) used to manage the software life-cycle activities; (c) application of standards, conventions, and other work practices that support the software life cycle; (d) controls for support software used to develop, operate, and maintain computer programs. (Ref. NQA-1).
Software Owner (SO)	Responsibilities (performs or causes to be performed, see others in chapter): • Provides the software information and Form 2033, *Safety/Non-Safety Software Determination, Categorization, and Software Risk Level (SRL)* and obtains review and concurrence of the form in accordance with this document. • Reviews and approves the software project planning documentation. • Owns the software and supports the SRLM in complying with the requirements of this document. • Prepares the approval for use documentation that describes the intended use and any associated limitations, access controls, etc., for using the software.
Software Point of Contact (SPOC)	That individual selected by division management to act as software owner for specific software or multiple Non-SCC software programs. Ideally it's the main or super-user of each program, but can be another user or even a single individual for a group or division.
Software Responsible Line Manager (SLRM)	Responsibilities (performs or causes action below to be performed, see others in chapter): • Manages and maintains software in accordance with this document to ensure it operates as intended. • Determines reasonable probability, and as applicable, • As applicable, acquires software and/or software services. • Except for SQA associated with using the software, is responsible for the SQA of the software. • V&Vs the software.

	reviews, signs and retains Form 2033 as a record. • Identifies, documents, approves, controls, and maintains safety and risk significant software owned by the SRLM that is new software or in-use software at LANL nuclear (including radiological) facilities. • Provides software inventory information and any changes thereto to QPA-IQ and ES. • Applies the appropriate amount of SQA rigor in software planning and implementation. • Reviews and approves the software project planning documentation. • Identifies and ensures reviews are performed by competent individuals or groups other than those who developed and documented in the original software design (but who may be from the same organization). • Ensures software owning organization personnel managing or working to this document are adequately trained and as required, qualified. • Approves software for use. • Completes in-use tests in the operating environment. • Retires software, including removal of safety software from software inventories.
software tool	A computer program used in the development, testing, analysis or maintenance of a program or its documentation. Examples include vendor-supplier configuration tools, conversion tables, comparators, cross-reference generators, compilers, CASE (Computer-Aided Design Software Engineering) tools, configuration and code management software, decompilers, disassemblers, editors, flowcharters, monitor test case generators, and timing analyzers. (Based on NQA-1).
Software User (SU)	Responsibilities (performs or causes to be performed, see others in chapter): • Reports software errors and problems. • Uses software within software limitations and in accordance with this document.
Software User Responsible Line Manager (SU RLM)	Responsibilities (performs or causes actions below to be performed, see others in chapter): • Supports completion of in-use tests in the operating environment. • Ensures software users and software user organization personnel managing or working to this document are adequately trained, and as required, qualified.
SSC software	Software that controls and/or monitors system, structures and components (SSCs) and is running and interacting with its environment in real time. SSC software may be safety or non-safety software. Examples: Building Automation Control System (BAS) software, process gas monitoring and control system software, fire alarm control panel (FACP) software, continuous air monitor (CAM) software, seismic switch software, and uninterrupted power supply (UPS) software. (Based on TR. No. 397, *Quality Assurance for Software Important to Safety*, IAEA, 2000).
support software	Software or a program that aides in the development, maintenance, or use of other software or provides general application-independent capability (Ref. ISO/IEC/IEEE 24765). Support software includes software tools and system software (Ref. NQA-1). ***Note***: SSC and Non-SSC software may have support software.
system software	An element of support software, the computer programs used to provide basic or general functionality and facilitate the operation and maintenance of the application computer program. Examples include lower level software layers, assemblers, interpreters, diagnostics, and utilities. (Based on NQA-1).
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. (Ref. NQA-1).

testing (software)	The process of: (a) operating a system (i.e., software and hardware) or system component under specified conditions (b) observing and recording the results (c) making an evaluation of some aspect of the system (i.e., software and hardware) or system component in order to verify that it satisfies specified requirements and to identify errors (Ref. NQA-1)
test plan (procedure)	A document that describes the approach to be followed for testing a system or component. Typical contents identify the items to be tested, tasks to be performed, and responsibilities for the testing activities. (Ref. NQA-1).
toolbox code	Software that is listed on the DOE Safety Software Quality Assurance Central Registry (Registry). (Ref. P1040).
validation (software)	The process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specified requirements and to identify differences between expected and actual results in an operating environment (Ref. NQA-1); and providing evidence that the software, and its associated products, satisfies system requirements allocated to software at the end of each life cycle activity, solves the right problem (e.g., correctly models physical laws, implements business rules, uses the proper system assumptions), and satisfies the intended use and user needs (Ref. DOE O 414.1D).
verification	The act of reviewing, inspecting, testing, checking, auditing, or otherwise determining and documenting whether items, processes, services, or documents conform to specified requirements (Ref. NQA-1); and providing objective evidence that the software and its associated products conform to requirements (e.g., for correctness, completeness, consistency, and accuracy) for all life cycle activities during each life cycle process (e.g., acquisition, supply, development, operation, and maintenance); satisfy standards, practices, and conventions during life cycle processes; successfully complete each life cycle activity; and satisfy all the criteria for initiating succeeding life cycle activities (e.g., building the software correctly) (Ref. O 414.1D).

Table 21.1-A2 Chapter 21 Acronyms	
Acroynm[1]	**Definition**
A	Approve
AFU	Approval for Use
ASCE	American Society of Civil Engineers
AP	Administrative Procedure
ASM	Acquisition Services Management
ASME	American Society of Mechanical Engineers
BAS	Building Automation System
BIO	Basis for Interim Operations
CAM	Continuous Air Monitor
CGD	Commercial Grade Dedication (Dedicated)
CM	Configuration Management
CoE	Conduct of Engineering
COOP	Continuity of Operations Plan
Cx	Commissioning
D	Develop
DA	Design Authority
DAG	Design Agency
DCF	Design Change Form
DOE	(United States) Department of Energy
DRN	Design Revision Notice
DSA	Documented Safety Analysis
EF	Essential Function
ES	Engineering Services
ES-Div	Engineering Services Division
ESM	Engineering Standards Manual
ESSC	Engineering Services Software Coordinator
FAC-COE	Facility Conduct of Engineering
FACP	Fire Alarm Control Panel
FCR	Field Change Request
FDAR	Facility Design Authority Representative
FDD	Facility Design Description
G	Guide
Gr	Grade
IEEE	Institute of Electrical and Electronics Engineers
IESL	LANL Institutional Evaluated Supplier List
IWR	Interim Work Request
LANL	Los Alamos National Laboratory
LCxA	LANL Commissioning Authority
ML	Management Level
NA	Not Applicable
NQA-1	ASME NQA-1-2008/NQA-1A-2009, Quality Assurance Requirements for Nuclear Facility Applications, Part I and Part II
OCIO	Office of Chief Information Officer

Table 21.1-A2 Chapter 21 Acronyms	
Acroynm[1]	**Definition**
O&M	Operations and Maintenance
OTS	Off the Shelf
PDSA	Preliminary Documented Safety Analysis
PFD	Process Flow Diagram
P&ID	Process and Instrumentation Diagram
POC	Point of Contact
R	Required or review (see context)
RE	Responsible Engineer
RLM	Responsible Line Manager
SB	Safety Basis Division
SBP	Safety Basis Procedure
S/CI	Suspect/Counterfeit Item
SDD	System Design Description
SHADS	Safety and Hazard Analysis Software
SMACS	Safety Management and Administrative Controls Software
SME	Subject Matter Expert
SO	Software Owner
SOO	Sequence of Operations
SOW	Statement of Work
SPOC	Software Point of Contact
SRLM	Software Responsible Line Manager
SSC	Structure, System, or Component
SSS	Safety System Software
SRL	Software Risk Level
SU	Software User
SWAU	Software Approve (Approval) for Use
SWDD	Software Design Documentation (or Document)
SWBL	Software Baseline
SWDS	Software Data Sheet
SWHA	Software Hazards Analysis
SWID	Software Identification Number
SWNCP	Non-SSC Software Change Package
SWRS	Software Requirements Specification
SWYRS	Software System Requirements Specification
SWTM	Software Traceability Matrix
SWTP	Software Test Plan
SWTR	Software Test Report
TA	Technical Area
UPS	Uninterrupted Power Supply
V&V	Verify and Validate (or verification and validation)

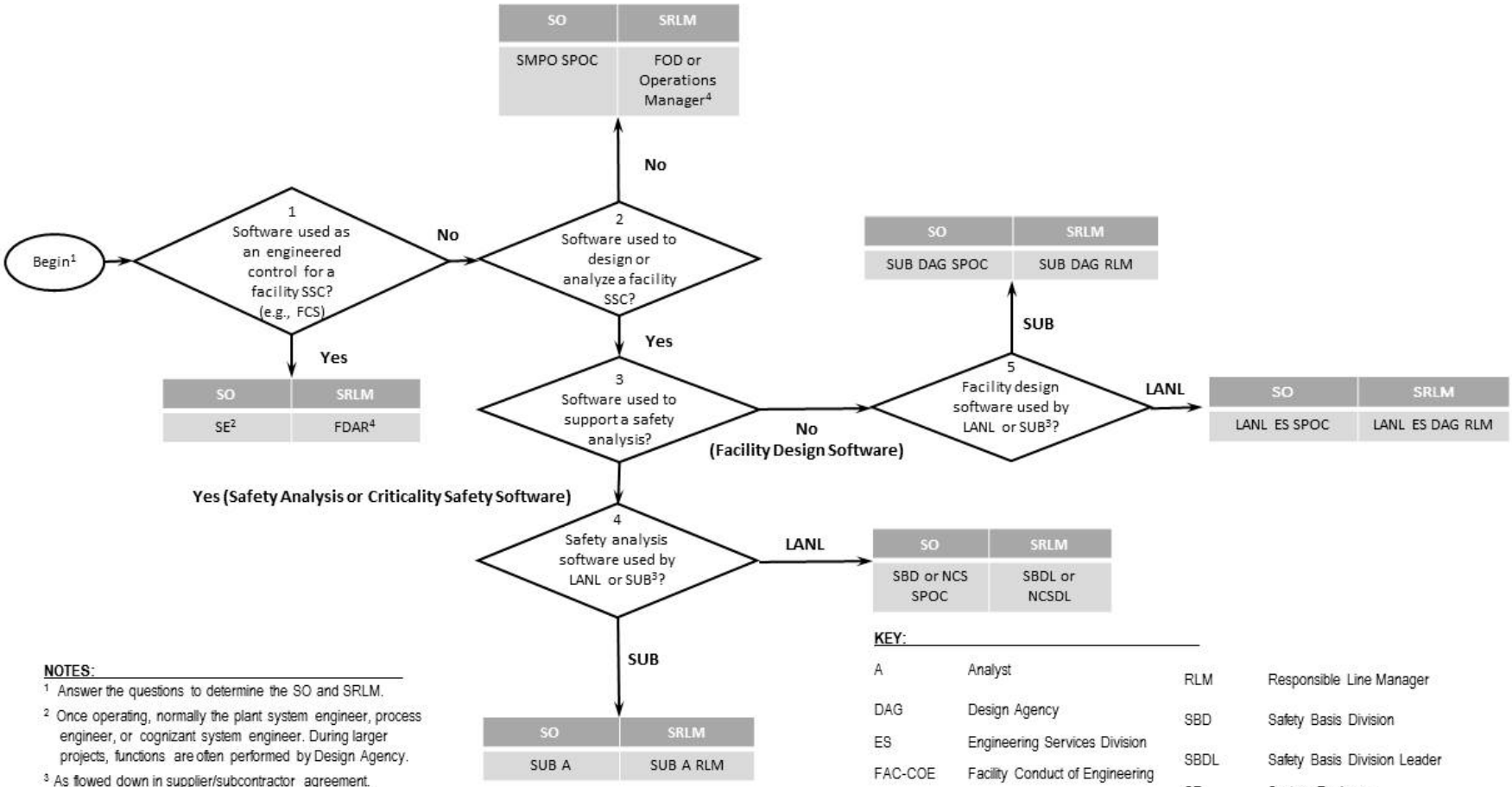
[1] Only key acronyms are listed. See PD340, *Conduct of Engineering for Facility Work* and documents referenced therein for additional meanings.

APPENDIX B: CHAPTER 21 SO AND SRLM DECISION DIAGRAM (GUIDANCE)

Use Figure 21.1-B1 to identify the SO and SRLM. For clarification, contact the Facility Design Authority Representative (FDAR).

- For software used within a single facility/application and/or under a single FDAR, the associated FDAR has the ultimate authority to determine the appropriate SO and SRLM.
- For software used among multiple facilities/applications where there may be multiple FDARs, the LANL Facility Design Authority (DA) has the ultimate authority to make the determination.
- *For fire alarm control panel (FACP) software, the system engineer is the SO. Fire Protection Division Office (FP-DO) provides support to the SO to promote chapter compliance.*

Figure 21.1-B1 Chapter 21 SO and SRLM Decision Diagram (Guidance)



NOTES:

1 Answer the questions to determine the SO and SRLM.

2 Once operating, normally the plant system engineer, process engineer, or cognizant system engineer. During larger projects, functions are often performed by Design Agency.

3 As flowed down in supplier/subcontractor agreement. Criticality Safety decisions use same general logic as SBD.

4 For software used across multiple FODs, the SMP Program Owner should be the SRLM.

5 For clarification, contact the FDAR who has the ultimate authority to determine the appropriate SO and SRLM.

6 See SOFT-GEN App. A, Definitions for administrative and engineered control definitions.

KEY:

A	Analyst	RLM	Responsible Line Manager
DAG	Design Agency	SBD	Safety Basis Division
ES	Engineering Services Division	SBDL	Safety Basis Division Leader
FAC-COE	Facility Conduct of Engineering	SE	System Engineer
FCS	Facility Control System	SMPO	Safety Management Program Owner
FDAR	Facility Design Authority Representative	SO	Software Owner
FOD	Facility Operating Director	SPOC	Software Point of Contact
MAR	Material at Risk	SRLM	Software Responsible Line Manager
NCS	Nuclear Criticality Safety	SSC	System, Structure, Component
NCSDL	Nuclear Criticality Safety Division Leader	SUB	Subcontractor

APPENDIX C: CHAPTER 21 REFERENCE LISTING

Most recent document revision (including revised document numbering and/or titles) applies unless otherwise stated. *Most national standards are available to LANL personnel via links; others must purchase.*

10 CFR 830, *Nuclear Safety Management*

10 CFR 835, *Occupational Radiation Protection*

48 CFR 970-5223-1, *Integration of Environment, Safety, and Health into Work Planning and Execution*

ANSI/ANS-10.7, *Non-Real-Time, High-Integrity Software for the Nuclear Industry – Development Requirements*

ASME NQA-1-2008/NQA-1A-2009, *Quality Assurance Requirements for Nuclear Facility Applications*, Part I and Part II

ASME NQA-1-2015 Part III, Subpart 3.2-2.14, *Quality Assurance Requirements for Commercial Grade Items and Services, Commercial Grade Computer Programs, and Software Services*

ASQ *Certified Software Quality Engineer Handbook*, L. Westfall, Am. Soc of Quality Pres.

DOE G 413.3-21, *Cost Estimating Guide*

DOE O 414.1D, *Quality Assurance*

DOE Safety Software Quality Assurance Central Registry

DOE G 414.1-4, *Safety Software Guide for Use with 10 CFR 830 Subpart A, Quality Assurance Requirements, and DOE O 414.1C, Quality Assurance* (Revision A pending)

DOE G 450.4-1C, *Integrated Safety Management System Guide*

DOE P 450.4A, *Integrated Safety Management Policy*

DOE SQAS21.01.00-1999, *Software Risk Management–A Practical Guide*

DOE-STD-1073, *Configuration Management*

DOE-STD-1195, *Design of Safety Significant Safety Instrumented Systems Used at DOE Nonreactor Nuclear Facilities*

EPA/100/K-09/003, *Guidance on the Development, Evaluation, and Application of Environmental Models, www.epa.gov/crem*

IAEA TR. No. 397, *Quality Assurance for Software Important to Safety*, IAEA, 2000

IEEE

ANSI/IEEE Std 7-4.3.2-2010, *IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations*

ANSI/IEEE Std 828, *IEEE Standard for Configuration Management in Systems and Software Engineering*

ANSI/IEEE Std 26514, *Systems and Software Engineering—Requirements for Designers and Developers of User Documentation*

ANSI/IEEE Std 29148, *Systems and Software Engineering – Life Cycle Processes – Requirements Engineering*

IEEE Std 344, *Standard for Seismic Qualification of Equipment for Nuclear Power Generating Stations*

IEEE Std 830, *IEEE Recommended Practice for Software Specifications*

IEEE Std 1012, *IEEE Standard for System and Software Verification*

IEEE Std 1016, *IEEE Standard for Information Technology – System Design – Software Design Descriptions*

IEEE Std 1016.1, *IEEE Guide to Software Design Descriptions*

IEEE Std 1028, *IEEE Standard for Software Reviews and Audits*

IEEE Std 1666, IEEE *Standard for Standard System C Language Reference Manual*

IEEE Std 12207, *Systems and Software Engineering – Software Life Cycle Processes*

ISO/IEC/IEEE 24765, *Systems and Software Engineering – Vocabulary*

ISO/IEC/IEEE 29119, *Software and systems engineering—Software testing*

ISO/IEC/IEEE 29148, *Systems and Software Engineering-Life Cycle Processes-Requirements Engineering*

ANSI/ISA S84.00.01, *Functional Safety: Safety Instrumented Systems for the Process Industry Sector*

LANL (internal-only unless noted)

AP-341-402, *Engineering Document Management in Operating Facilities*

AP-341-405, *Identification and Control of Technical Baseline, Variances, Alternate Methods, and Clarifications in Operating Facilities*

AP-341-502, *Management Level Determination and Identification of Quality Assurance and Maintenance Requirements*

AP-341-504, *Temporary Modification Control*

AP-341-507, *SSC Control Software Change Package*

AP-341-510, *Field Walk-down, Data Gathering, and Inspections*

AP-341-519, *Design Revision Control*

AP-341-605, *Calculations*

AP-341-601, *Functions and Requirements Document*

AP-341-602, *Requirements and Criteria Document (RCD)*

AP-341-611, *System Design Descriptions*

AP-341-613, *Instrumentation Set Point Control*

AP-341-616, *Technical Baseline Change During Design*

AP-341-620, *Review of LANL Produced Design Documents*

AP-341-621, *Design Authority Technical Review*

AP-341-622, *LANL Review of Designs Produced by External Agencies*

AP-341-703, *Commercial Grade Dedication*

AP-341-801, *Post Modification/Post Maintenance Testing*

AP-341-802, *System Health Reporting*

AP-341-901, *Performing Vital Safety System Assessments*

AP-350-406, *Startup and Commissioning*

ASM 3400.00.0410, *Goods or Services Requiring Special Review/Approval*

ASM Website

Conduct of Operations Resources

CIO-P100, *Cyber Security Risk Management*

SEO-COOP-006, *Continuity of Operations (COOP) Plan*

ES Division Office (ES-Div) Website

ESD Training Program Plan, ESD-PLAN-001

ESDO-AP-002, *Engineering Services (ES) Software Inventory Instructions*

Form 2033, *Safety/Non-Safety Software Determination, Categorization, and Software Risk Level (SRL)*

Information System Security Officer (ISSO)

LANL Institutional Supplier List (IESL)

LANL Nuclear Facility List

LANL Safety Basis Document List

LA-14167-MS, Los Alamos National Laboratory, *Concepts of Model Verification and Validation,* Thacker et al., 2004

P204-1, *Controlled Unclassified Information*

P218, *Cyber Security Controls*

P322-4, *Issues Management*

P330-2, *Control and Calibration of Measuring and Test Equipment (M&TE)*

P330-3, *Quality Audits*

P330-5, *Special Processes*

P330-6, *Nonconformance Reporting*

P330-8, *Inspection and Test*

P330-9, *Suspect/Counterfeit Items (S/CI)*

P341, *Facility Engineering Processes Manual*

P342, *Engineering Standards*

P343, *Facility Engineering Training and Qualification Manual*

P781-1, *Conduct of Training*

P805, *Export Control*

P840-1, *Quality Assurance for Procurements*

P1020-2, *Laboratory Document Control*

P1040, *Software Quality Management*

P1011, *Computer Hardware, Software, and Services: Exceptions to Acquisition, Maintenance, and Support*

PD110, *Safety Basis*

PD115, *LANL Readiness Program*

PD210, *Cyber Security Program*

PD328, *LANL Assessment Program*

PD340, *Conduct of Engineering for Facility Work*

PD370, *Conduct of Engineering for Research and Development (R&D)*

QPA-DO-FSD-001, *LANL Safety Software Inventory Procedure*

SBP-15-351, *Design Basis or Safety Basis Change Review*

SBP111-1, *Facility Hazard Categorization and Documentation*

SBP112-3, *Unreviewed Safety Question (USQ) Process*

SBP113-3, *Unreviewed Safety Issue (USI) Process*

SD130, *Nuclear Criticality Safety Program*

SD330, *Los Alamos National Laboratory Quality Assurance Program*

STD-342-100, *Engineering Standards Manual* (available externally)

Merriam-Webster Dictionary

TeamForge (LANL-internal)

Trac

ATTACHMENT 1: CHAPTER 21 SUMMARY TABLE

Rev	Date	Description	POC
0	06/23/16	Initial issue	Tobin Oruch, ES-DO
1	05/25/17	Updated for Rev. 1 of chapter	Tobin Oruch, ES-DO

This table is summary of the entire chapter, addressing both SSC software and Non-SSC software. Every attempt is made to keep it in sync with the requirements in each Section; however:

- Section requirements are also more detailed, and
- In the case of a conflict, contact the Chapter POC (Section requirements will generally supersede those in this summary).

Table 21.1-ATT1 Chapter 21 Summary by Software Management Level (ML)
This table is a summary only and does not include all requirement details. See text of Section(s) for details.

Section Name/ Subsection[1]	Activity Description	Deliverable	SSC Software ML[2]			Non-SSC Software ML[2]		
			1, 2	3	4	1, 2	3	4
SOFT-GEN, 2	Identify software, evaluate and complete software determination	Software summary (Also required for subcontractors working to compliant programs other than this chapter)	R	R	-	R	R	R
		Form 2033	R	R	Gr[3]	R	R	Gr[3]
SOFT-GEN, 3	Plan software-specific activities	Software data sheet form SOFT-GEN-FM01 (SWDS)	R	R	-	R	R	-
SOFT-GEN, 4	Develop and maintain a software baseline (SWBL)	Software baseline form SOFT-GEN-FM02 (SWBL)[4]	R	R	-	R	R	-
SOFT-GEN, 4	Control software changes:							
	Less than minor change, minor and major changes made before or after software approval for use (SWAU)	Documented comment in the computer program where feasible	R	R	R	R	R	R
	Minor or major changes made before SWAU	AP-341-519 change documents[5]	R	Gr	Gr	-	-	-
	Minor or major changes made once software is baselined	Revised SWBL[4]	R	R	-	R	R	-
	Minor or major changes made after SWAU	SSC Software Change Form AP-341-507-FM01	R	Gr	-	-	-	-
		Non-SSC Software Change Form SOFT-GEN-FM03 (SWNCP)	-	-	-	R	Gr	-
SOFT-GEN, 5	Report problems and take corrective action:							
	Before SWAU	P330-6, P322-4, bug lists, documented comment field in the computer program	R	R	Gr	R	R	Gr
	After SWAU	P330-6, P322-4 documents (e.g., Nonconformance Reports [NCRs])	R	R	Gr	R	R	Gr
SOFT-GEN, 6	Manage software (project) risks	Risk register (or list) as part of SWDS	R	R	-[12]	R	R	-[12]
SOFT-ACQUIRE, 3, 4	Acquire existing LANL software if suitable	Update authorized users/SWDS as required; generate an interface agreement where desired	-	-	-	R	R	Gr

Table 21.1-ATT1 Chapter 21 Summary by Software Management Level (ML)
This table is a summary only and does not include all requirement details. See text of Section(s) for details.

Section Name/ Subsection[1]	Activity Description	Deliverable	SSC Software ML[2]			Non-SSC Software ML[2]		
			1, 2	3	4	1, 2	3	4
SOFT-ACQUIRE, 3, 4	Acquire new software	▪ Statement of Work (SOW)	R	R	-	R	R	-
		▪ SWDS	R	R	-	R	R	-
		▪ Commercial Grade Dedication (CGD) Documentation per AP-341-703	R[6]	R[6]	-	R[6]	R[6]	-
		▪ P-card/Purchase Order (PO)	R	R	R[7]	R	R	R[7]
		▪ Exhibit H	R	R	R[8]	R	R	R[8]
		▪ Inter-/Intra- LANL Agreements[9]	R[9]	R[9]	R[9]	R[9]	R[9]	R[9]
SOFT-DESIGN, 3, 6	Develop software design requirements (input)	▪ Software design inputs: o SSC drawings o SSC performance specification o FDD or SDD ▪ Other SSC design inputs/outputs as required	R	Gr	Gr	-	-	-
		▪ Software system requirements specification (SWYRS)	--	-	-	R	Gr	-[13]
SOFT-DESIGN, 4, 7	Design the software (output)	▪ Software design outputs: o software specification (SWRS) o software design (SWDD) o software system hazard analysis and mitigation documentation (SWHA) o software design traceability documentation (SWTM) o SOFT-GEN deliverables o Other software design deliverables as required o Operations and maintenance (O&M) instructions o computer model mathematical terms[10]	R	Gr	Gr	R	Gr	-[13]

Table 21.1-ATT1 Chapter 21 Summary by Software Management Level (ML)
This table is a summary only and does not include all requirement details. See text of Section(s) for details.

Section Name/ Subsection[1]	Activity Description	Deliverable	SSC Software ML[2]			Non-SSC Software ML[2]		
			1, 2	3	4	1, 2	3	4
SOFT-DESIGN, 5, 8	Translate the software design into computer program language (code)	▪ Computer program language (code): o data structures/files, o source code (where feasible), o executable code ▪ Computer program listings (e.g., pdf printout of source code)	R	Gr	Gr	R	Gr	Gr
SOFT-V&V, 4	Review (input) requirements	▪ Dispositioned review comments	R	Gr	Gr	R	Gr	Gr
SOFT-V&V, 4	Review acquisition	▪ Acquisition review document	-	-	-	R	R	-
SOFT-V&V, 4	Develop and review the test plan (SWTP)	▪ Test plan (e.g., Form 3055 (future)) ▪ Dispositioned review comments	R	Gr	Gr	R	Gr	Gr
SOFT-V&V, 4	Review the software design (SWDD)	▪ Software design ▪ Dispositioned review comments	R	Gr	Gr	R	Gr	Gr
SOFT-V&V, 4	Review computer program code where feasible	▪ Dispositioned code review comments	R	-	-	R	-	-
SOFT-V&V, 4	Test and review test results	▪ Interim tests and test (qualification) reports ▪ Simulated acceptance test and acceptance test report ▪ Acceptance test and test (qualification) report (SWTR) ▪ Dispositioned review comments	R	Gr	Gr	R	Gr	Gr
SOFT-V&V, 4	Review and approve software for use (SWAU)	▪ SWAU request package ▪ Dispositioned review comments ▪ Software approval for use form SOFT-V&V-FM01, SWAU	R	Gr	-	R	Gr	-
SOFT-INV, 3, 4	Inventory safety software	▪ Updated institutional-level safety software inventory	R	R[11]	-	R	R[11]	-
		▪ ES-Div: Updated ES Non-SSC software inventory	-	-	-	R	R[11]	-

Table 21.1-ATT1 Chapter 21 Summary by Software Management Level (ML)
This table is a summary only and does not include all requirement details. See text of Section(s) for details.

Section Name/ Subsection[1]	Activity Description	Deliverable	SSC Software ML[2]			Non-SSC Software ML[2]		
			1, 2	3	4	1, 2	3	4
SOFT-INV, 3, 4	Inventory non-safety software	▪ ES-Div: Updated ES non-SSC software inventory ▪ ES-Div: Form 2033 to ESSC	-	-	-	R	R	R
SOFT-MAINT, 3, 4	Use and maintain the software	▪ Operation and maintenance (O&M) documentation ▪ Operational event documentation ▪ Application logs ▪ Licenses and registrations ▪ Access controls ▪ Computer system vulnerability protections ▪ Problem reports and corrective actions per SOFT-GEN ▪ In-use test plans, tests, review documentation, and test reports per SOFT-V&V ▪ Updated, configuration management, SWDS and deliverables as described on SWDS and SWBL per SOFT-GEN ▪ Accurate software inventory per SOFT-INV ▪ Other quality assurance control documentation (e.g., training records, assessments)	R	Gr	Gr	R	Gr	Gr
SOFT-MAINT, 3, 4	Retire the software	▪ Software removed to prevent unintended or routine use ▪ Terminated licenses/agreements ▪ Updated software inventory(ies) ▪ Cancelled or revised affected documents ▪ Software record	R	R	R	R	R	R

Notes:

[1] ESM Chapter 21 section name and subsection number.

[2] See Form 2033 for a crosswalk between MLs and software type, category and SRL. R = Required. Gr = Required but graded. "-" = Not required.

[3] Complete Form 2033 for ML-4 software only if there is a reasonable probability that it could be ML-1, ML-2, or ML-3 software, or, the highest function of the SSC is greater than ML-4.

[4] See SOFT-GEN-FM02, *Software Baseline Form (SWBL)*; the SWBL may be included as part of a system design description (SDD) or other controlled document. The SWBL must include documents and computer program files.

[5] Includes Field Change Notices (FCNs), Field Change Requests (FCRs), Interim Work Requests (IWRs), Design Revision Notices (DRNs), Design Change Forms (DCFs) as described in AP-341-519 unless specifically superseded by another administrative procedure.

[6] Required only for ML-1 or ML-2 safety software that is not procured from an NQA-1 qualified supplier.

[7] A PO or a Purchase Card (P-card) can be used for ML-4 software; a P-card cannot be used for ML-1 through ML-3 software.

[8] Required if the supplier comes onsite (at LANL) to install and/or maintain software.

[9] Required, or should be required, if software is acquired from LANL organizations, DOE sites or government agencies. See P840-1 for details.

[10] Required for computer program models only.

[11] Required if safety software.

[12] Following Chapter 21 provides adequate risk management.

[13] May be addressed in associated software calculation.

SWDS No.:	Rev.:

1.0 SWDS PURPOSE, AUTHORITY AND APPLICABILITY

1.1 Introduction:	The purpose of this software data sheet (SWDS) is to provide the software-specific, software quality management implementing details for the subject software in accordance with STD-342-100, *LANL Engineering Standards Manual, Chapter 21 – Software* (hereafter *Chapter 21*). Use of *Chapter 21*, this SWDS and the documents referenced therein, provide reasonable assurance that the software will consistently, compliantly and efficiently satisfy its intended use. This SWDS is issued under the authority of the software owner responsible line manager (SRLM). The SRLM is responsible for managing and maintaining this document in accordance with *Chapter 21*. This document applies to LANL employees, and as specified in subcontracts, subcontractors.

1.2 Software Name:				1.3 SWID No.:	
1.4 TA Number(s):		1.5 Facility Number(s):		1.6 Facility Name(s):	
1.7 Software Description:					
1.8 Software Need:					
1.9 SSC Software/Non-SSC Software:	SSC ☐ Non-SSC ☐			1.10 Equipment ID:	
1.11 Highest Management Level (ML) Associated with the Software:	ML-1 ☐	ML-2 ☐	ML-3 ☐	ML-4 ☐	
1.12 Software Designation:	Select from drop down menu				

2.0 ROLES AND DOCUMENT APPROVALS

2.1 The names of those filling roles at the time this SWDS was generated are indicated below. This SWDS was developed **(D)** by the SO and reviewed and approved **(R, A)** by the SRLM (and others if indicated below). See the Engineering Services Division Office (ES-DO) software inventory for any subsequent name changes to key roles.

2.2 Name (Last, First):	2.3 Z Number:	2.4 Role:										2.5 SWDS Approval and Approval Date:
		SO (**D**)	SRLM (**R, A**)	SD	SD RLM	SU	SU RLM	FDAR or DA	Independent Design V&V	Other	Other	

SWDS No.: Rev.:

2.6 Role Comments/Clarifications:

3.0 REQUIREMENTS	
3.1 Software Requirements, Limitations and Hierarchy (See SWBL for latest revisions)::	
3.2 Key Planning Assumptions/Constraints:	
4.0 SCOPE/DELIVERABLES	
4.1 Deliverables:	
5.0 ACQUISITION/DESIGN STRATEGY	
5.1 Acquisition/Design Strategy:	
6.0 SOFTWARE PROJECT MANAGEMENT (PM)	
6.1 Rough Estimate of Software Project Cost, K$	

Fiscal Year (FY):					Total, K$ (5 FY):
Current FY (FY1)	FY2	FY3	FY4	FY5	

6.2 Other PM Considerations:	
7.0 USE AND MAINTENANCE	
Use and maintain software in accordance with the requirements and limitations in Section 3.0 and the following:	
7.1 Software User/Administrator Authorization Process:	
7.2 Authorized Software User Requirements:	
7.3 Authorized Administrator Requirements:	

SWDS No.: Rev.:

7.4 Software Risk Register Required?	Yes ☐	No ☐	7.5 Risk Register Minimum Review/Frequency:	
7.6 Minimum Assessment Frequency & Method(s):				
7.7 In-Use Testing Required?	Yes ☐	No ☐	7.8 Minimum In-Use Test Frequency & Method(s):	
7.9 Problem Reporting and Corrective Action (PR&CA)	☐ P330-6, *Nonconformance Reporting* and/or P322-4, *Laboratory Performance Feedback and Improvement Process*			
	☐ Other (Describe): _____			
7.10 Other				

8.0 ATTACHMENT LIST

8.1 Attachment No.	8.2 Attachment Title

9.0 SWDS REVISION HISTORY

9.1 Revision	9.2 Revision Date	9.3 Revision Description and Reason for Revision
0	Click here to enter a date.	Initial issue

GENERAL

This software data sheet, and the documents referenced herein, provide the *software-specific* information necessary to implement the requirements of STD-342-100, *LANL Engineering Standards Manual, Chapter 21 – Software* (hereafter *Chapter 21*). This form promotes the successful and consistent implementation of the software data sheet (SWDS) requirements of *Chapter 21*. For definitions, general requirements, and processes, see *Chapter 21*, SOFT-GEN.

This form is developed (D) by the Software Owner (SO); and at a minimum, reviewed and approved (R, A) by the Software Responsible Line Manager (SRLM).

If entering sensitive information (e.g., Unclassified Controlled Nuclear Information -UCNI), ensure proper Derivative Classifier/Reviewing Official reviews are performed and appropriate markings applied.

Establishing use and maintenance requirements (Field 7) using SOFT-MAINT: The Initial SWDS revision may have TBD for some use and maintenance parameters that are not known in the early planning stage and will be determined at a future date.

LANL personnel: Endeavor to use Chapter forms as-is and report issues and improvement ideas to the Chapter 21 POC. POC may authorize other methods equivalent to chapter forms in writing.

LANL subcontractors must use Chapter forms to satisfy Chapter requirements for SSC software. For Non-SSC software, subs may either use their own forms or integrate, adapt, and reformat the forms; either approach is acceptable so long as key functions, data, and approvals are retained.

HEADER

Field	Entry Information
Software Name	Enter the name of the software in the upper right hand corner of the header after the title (e.g., Software Data Sheet Form (SWDS) *for Ceasar II Software*).
SWDS No.	The numbering scheme of this form follows that of the SRLM's document control/records management system. For software used in the operation of existing or new LANL facilities, use numbering scheme and process specified in AP-341-402, *Engineering Document Management in Operating Facilities*.
Rev.	Enter the SWDS revision number. Initial issue is 0.

1. SWDS PURPOSE, AUTHORITY AND APPLICABILITY

Field	Entry Information
1.1	Review, and as required, revise introductory text to ensure accuracy for the subject software with respect to SWDS purpose, authority, and applicability.
1.2	Enter the name of the software. Do not enter the software revision as revisions may change over time. Use the Software Baseline (SWBL) to track the software revision.
1.3	Enter the software identification number (SWID). *The SWID should normally be the same SWID as on the Form 2033. Obtain per AP-341-402, Engineering Document Management in Operating Facilities and the CoE SharePoint Document Numbering Site*. Ensure the SWID is part of subsequent software documentation.
1.4	Enter the two-digit Technical Area (TA) number(s) where the software is used. If used sitewide, enter 99.

Field	Entry Information
1.5	Enter the facility number(s) where the software is used. Follow AP-341-402 conventions for utilities, multiple, etc. (e.g., i*f multiple, enter MULT…)*
1.6	Enter the facility name(s) where the software is used.
1.7	Describe what the software does.
1.8	Describe why the software is needed (justification).
1.9	Indicate whether the software is SSC Software or non-SSC software by checking the appropriate box. Check only one box.
1.10	For SSC software, enter the equipment identification number associated with the software per ESM Chapter 1, Section 200 (e.g., 03-0216-HVAC-BAS-1). For new SSC software where the equipment identification number is not yet established, enter TBD. For Non-SSC software, enter NA.
1.11	Indicate the highest management level (ML) associated with the software (where ML-1 is highest, ML-4 lowest). Reference AP-341-502. If the software is associated with multiple MLs, check the highest ML box. Check only one box.
1.12	Select the software designation (safety software, risk significant software [RS] or commercially controlled software [CC]) from the drop down menu. Select only one designation.

2. ROLES AND DOCUMENT APPROVALS

Field	Entry Information
2.1	Revise the example text as needed to indicate how future name changes are addressed in the SWDS. (e.g., Modify reference to ES-DO software inventory if needed. One may indicate "Changes to the SRLM will be documented in a revision to this document; for other role name changes, contact the SRLM" or similar text.)
2.2	Enter the name of the person(s) filling the Chapter 21 role(s) at the time the SWDS is generated. For assistance in determining the SO and SRLM, see SOFT-GEN Appendix C: *SO and SRLM Decision Diagram for FAC-COE.* One person—i.e., one software user (SU) etc.—is a sufficient representative sample for SUs. Reference to a role listing outside of the SWDS is acceptable (e.g., for SUs, "see Safety Basis Analysts as identified on Safety Basis Division Website"). ***Note:*** The SWDS does not need to be revised if persons in the required roles change as long as the person filling the role is adequately communicated through other means.
2.3	Enter the Z number (if available). Enter NA if not available.
2.4	Enter an "x" for each role that the person is filling. Note that one person may serve multiple roles. For any roles that the SRLM wishes to electively require SWDS review and approval, enter "(R, A)" after the role.

2.5	Enter the date and either initials or signature that the SWDS was reviewed (R) and approved (A). The effective date of the document is the date of the last acquired signature unless otherwise indicated. Electronic approvals are acceptable. Enter for those roles in block 2.4 where R, A is indicated. At a minimum, the SLRM must review and approve the SWDS. For roles that do not require review and approval of the SWDS, leave blank or enter NA.
2.6	Enter text as necessary to clarify Chapter 21 roles. If roles are added that are in addition to those in Chapter 21, specify the roles and responsibilities in this section. If no clarifications are necessary, enter "NA".

3. REQUIREMENTS

Field	Entry Information
3.1	Enter the requirements in the order of precedence. (i.e., Requirement document 1 takes precedence over requirement document 2.) Include the document number and name and revision. Include any limitations as applicable. The requirements may also be entered directly rather than in separate requirements documents. For example, "For LANL applications, software should only be used with ASME B31.3-2016 Ver. 1.1 database file"; "Use only modules 2 and 3; do not use other modules". ***Note:*** Subsequent changes to the requirements in this section are managed using the Software Baseline (SWBL).
3.2	Enter high-level key planning assumptions/constraints. Planning assumptions/constraints are those that may affect the overall design/acquisition approach, scope, schedule or cost (e.g., software, operating system, and computer hardware upgrades are assumed every three years starting in the current year and will require full software verification and validation as well as revisions to most software quality documentation.)

4. SCOPE/DELIVERABLES

Field	Entry Information
4.1	Enter the software deliverables required throughout the software life cycle. See SOFT-GEN section, Attachment B for a summary of Chapter 21 deliverables. These deliverables summarize the scope of the software project.

5. ACQUISITION/DESIGN STRATEGY

Field	Entry Information
5.1	Indicate if the software is existing or new. Describe the approach that was, and/or will be used, to acquire/design the software; address support software and associated hardware. If the software is safety software (ML-1 or ML-2 only) indicate whether the software will be acquired from an NQA-1 qualified supplier or whether it will acquired from a Non NQA-1 qualified supplier using commercial grade dedication (CGD). See acquisition factors such as cost, supplier availability in SOFT-ACQUIRE. Provide a summary level description of what work was, and/or will be performed by LANL personnel and what work was, and/or will be performed by supplier(s). Indicate whether the software will be for used on a server (generally preferred) or standalone computer. Describe licenses, registrations, problem reporting, and/or services that must be maintained.

6. SOFTWARE PROJECT MANAGEMENT (PM)

Field	Entry Information
6.1	Provide estimated software costs for the next five years. Use the high-level Work Breakdown Structure (WBS) below or other desired WBS as an aid. Attach supporting detail as required. See SWDS example. **WBS 1.1 - Design/Acquire Software.** This is the software design and/or acquisition. It includes planning, documentation (for hardware and software), software quality assurance activities, Verification and Validation (V&V) and all other activities associated with software design and/or acquisition. It includes labor, equipment and materials. **WBS 1.2 – Design/Acquire Hardware.** This is the associated hardware. It includes computer servers, computer workstations, etc. It includes labor, equipment and materials. **WBS 1.3 – Install/Approve for Use.** This is software and hardware installation; associated installation V&V; and approval for use. It includes labor, equipment and materials. **WBS 1.4 – Use and Maintain**. This is the use and maintenance of the integrated software/hardware system. It includes software license fees, training costs, problem reporting, corrective action, configuration management, in-use testing, assessments etc. ***Note:*** Cost and schedule information are planning level estimated provided to help management allocate sufficient resources to successfully manage the software throughout its lifecycle. Detailed cost estimating is not required.
6.2	As determined by the SRLM, add software specific project management (PM) detail here for those software projects where such detail is necessary to successfully manage the software. (An example of this would be target schedule milestone dates for completion of software requirements definition, software design/acquisition, software final acceptance testing, software approval for use.) If not necessary, enter NA. ***Note:*** Other PM considerations (e.g., schedule, reporting, etc.,) are normally addressed as part of the SLRM's work control and/or integrated hardware project management protocols.

7. USE AND MAINTENANCE

Field	Entry Information
7.1	Describe the person(s) and process for authorizing software users/administrators. (For ES-Div Non-SSC ML-1 through 3 software, this should be the ES Division Non-SSC Software Inventory), also linked from the Ch. 21 webpage under references.
7.2	Enter the minimum training and access controls/vulnerability protections a person must follow to be an authorized software user (e.g., required reading, Cryptocard access level, training courses, certifications, etc.). An authorized user is authorized to use the software for its intended purpose. An authorized user does not have the authority to make changes to the computer program code and/or perform software maintenance activities. Required reading of the SWDS and associated user documentation should typically be specified, particularly when there are known software limitations or reported errors.
7.3	Enter the minimum training-related activities that a person must fulfill to be an authorized software administrator (required reading, training courses, certifications, etc.). An authorized administrator is an authorized user who also has authority to make changes to the computer program code and/or perform software maintenance activities.
7.4	Indicate whether a software risk register is required by checking either the yes or no box. (A register is required for ML-1 and ML-2 software). If required, obtain Form 3046, *Software Risk Register (SWRR) Template* from the LANL Forms Center, complete it and attach it to the SWDS. (Providing a reference (e.g., title, document number, hyperlink) to a stand-alone risk register is also acceptable.)
7.5	Indicate the minimum frequency that the risk register must be reviewed and as required, revised to ensure accuracy of the list. A minimum of once per year is recommended. If a risk register is not required, enter "NA".
7.6	Indicate the minimum frequency that a software quality assessment must be performed, and the assessment methods (e.g., surveillance, audit, etc.). See P328-1, *Performance Assurance Planning Cycle and Integrated Assessment Schedule Maintenance* for scheduling guidance. Multiple assessment methods are acceptable (e.g., P330-3, *Quality Audits* per PD328, *LANL Assessment Program,* AP-341-901, *Performing Vital Safety System Assessments* per P341, *Facility Engineering Processes Manual*). Software quality may be assessed as part of broader assessments.
7.7	Indicate whether in-use testing is required by checking either the yes or no box. (In-use testing is required for ML-1 and ML-2 software and recommended for ML-3 software). See SOFT-MAINT. In-use testing must be performed when software is installed on a different computer or after significant software changes.
7.8	Indicate the minimum frequency that in-use tests must be performed, and the acceptable test method(s). Enter "NA" if in-use tests are not required. Note that in-use software tests may be an element of broader tests as long as they satisfy the software in-use test requirements.
7.9	Indicate the problem reporting and corrective action (PR&CA) processes to be used. Check all that apply. If other is checked, either describe the process in the space and/or reference the process. Include "who", "when", and "how" to report problems and take corrective action. Ensure the process satisfies Chapter 21 (SOFT-GEN) PR&CA requirements. ***Note:*** Less formal methods may be used through development and testing (e.g., bug lists); however, once the software is approved for use, formal methods apply. If more than one method is checked, clarify which method applies and when it applies.

Field	Entry Information
	Note: *For acquired software, check other and indicate the minimum frequency the SRLM should review supplier and/or other pertinent organization (e.g., NRC) problem reports, notifications, knowledge bases, websites or other information sources that could affect software management and/or use. For non-SSC, ES-Div software, each review should be documented in the ES-Div, software inventory. For SSC software, each review should be documented in the associated SSC operations documents (e.g., operator logs etc.)*
7.10	As required, provide any other software-specific protocols required to promote proper software use, maintenance and retirement (e.g., operational event documentation review and retention protocols, application log protocols, etc.). See SOFT-MAINT for additional information. Enter NA if not applicable.

8.0 ATTACHMENT LIST

Field	Entry Information
8.1	List attachments as appropriate. Enter the attachment number.
8.2	Enter the attachment title.

9.0 SWDS REVISION HISTORY

Field	Entry Information
9.1	Enter the revision number and update the revision number in the form header.
9.2	Enter the date of the revision from the date drop down menu for past revisions. For current revision, enter “Last approval date in Section 2.0” or similar text.
9.3	Enter a summary description of what was revised and the reason for the revision.

SWDS No.:	SWDS-16-03-2333-0089	Rev.:	0

1.0 SWDS PURPOSE, AUTHORITY AND APPLICABILITY

1.1 Introduction:	The purpose of this software data sheet (SWDS) is to provide the software-specific, software quality management implementing details for the subject software in accordance with STD-342-100, *LANL Engineering Standards Manual, Chapter 21 – Software* (hereafter *Chapter 21*). Use of *Chapter 21*, this SWDS and the documents referenced therein, provide reasonable assurance that the software will consistently, compliantly and efficiently satisfy its intended use. This SWDS is issued under the authority of the software owner responsible line manager (SRLM). The SRLM is responsible for managing and maintaining this document in accordance with *Chapter 21*. This document applies to LANL employees, and as specified in subcontracts, subcontractors.

1.2 Software Name:	Combustible Material Tracker (CBT) Software			1.3 SWID No.:	03-2333-045
1.4 TA Number(s):	03	1.5 Facility Number(s):	2333	1.6 Facility Name(s):	Nonproliferation Security Facility (SF)
1.7 Software Description:	The software is used as a tool to track combustible material				
1.8 Software Need:	The software is needed because tracking combustible material manually (without software) is less efficient and error-prone.				
1.9 SSC Software/Non-SSC Software:	SSC ☐ Non-SSC ☒			1.10 Equipment ID:	NA
1.11 Highest ML Associated with the Software:	ML-1 ☐ ML-2 ☐ ML-3 ☒ ML-4 ☐				
1.12 Software Designation:	Safety Software				

2.0 ROLES AND DOCUMENT APPROVALS

2.1 The names of those filling roles at the time this SWDS was generated are indicated below. This SWDS was developed **(D)** by the SO; and, reviewed and approved **(R, A)** by the SRLM (and others if indicated below). See the Engineering Services Division Office (ES-DO) software inventory for any subsequent name changes to key roles.

2.2 Name (Last, First):	2.3 Z Number:	2.4 Role:										2.5 SWDS Approval and Approval Date:
		SO **(D, R, A)**	SRLM **(R, A)**	SD	SD RLM	SU	SU RLM **(R, A)**	FDAR or DA	Independent Design V&V	ISSO Rep. **(R,A)**	Other	
Trujillo, Robin	124268	x										R. Trujillo signature, 9/25/16
Lopez, Pete	121987		x				x					P. Lopez signature, 9/27/16
Walker, John	123495									x[1]		J. Walker signature, 09/27/16

Obsolete Data

SWDS No.:	SWDS-16-03-2333-0089	Rev.:	0

Vigil, Mary	115970			x								NA
Thompson, Tom	143987				x							NA
Bradley, Mike	114987					x						NA
Heitkamp, Heidi	121987							x	x			NA

2.6 Role Comments/Clarifications:

[1] The ISSO Representative (Rep.) is responsible for: (a) ensuring applicable PD210, *Cybersecurity Program,* CIO-P100, *Cyber Security Risk Management* cybersecurity/security requirements are identified and reported to the SRLM; and, for (b) supporting implementation of such requirements.

3.0 REQUIREMENTS

3.1 Software Requirements, Limitations and Hierarchy:	1. ESM Chapter 21, Software 2. SRS-15-03-NSF-0002 CBT Software Requirements Specification 3. Limitation: Use only software modules 2 and 3; do not use other modules.
3.2 Key Planning Assumptions/Constraints:	1. The upgrade must be completed by July 1, 2017. If not completed by that date, Los Alamos Field Office authorization is required to operate with existing software. 2. Software, software operating system, and supporting computer hardware will required upgrade/replacement approximately every four years.

4. SCOPE/DELIVERABLES

4.1 Deliverables:	1. Form 2033 and revisions. 2. Documents (e.g., user manual) and computer programs as per SBL-15-03-2333-0234 (and subsequent revisions), Software Baseline for CBT with software replacement (including operating system) frequency estimated every 4 years. 3. Hardware as listed in SD-15-03-NSF-0006, CBT Software Design Document; Programmable Logic Controllers and associated devices (e.g., firewalls) replacement frequency estimated every 4 years. 4. Approval for use documentation (Section SOFT-V&V, Att. B, as required) 5. Non-SSC Software Change Packages (Section SOFT-GEN, Att. E, as required) 6. Problem reporting & corrective action documentation (as required) 7. Institutional-Level safety software inventory and ES-DO inventory updates as required. 8. Software quality assessment reports, bi-annual or as required. 9. Software risk register updates, as required. 10. Software data sheet (this document) revisions, as required. 11. Operational event documentation

Fictitious Data

SWDS No.:	SWDS-16-03-2333-0089	Rev.:	0

5. ACQUISITION/DESIGN STRATEGY

5.1 Acquisition/Design Strategy:	▪ **Primary Software:** The primary software is an existing C++ code. This code will undergo re-design by qualified LANL programmers and include software code/deliverables as required by ESM Chapter 21 for ML-3 software. ▪ **Support Software:** Support software is off-the-shelf software (e.g., MS 2003 and MS Windows XP) that will be acquired as ML-4 software from LANL Electronic Software Distribution (ESD). ▪ **Hardware:** Hardware is conventional PLC related hardware and peripherals that will be procured as ML-4 hardware.

6.0 SOFTWARE PROJECT MANAGEMENT (PM)

6.1 Rough Estimate of Software Project Cost, K$ (See attached supporting detail)

Fiscal Year (FY):					Total, K$ (5 FY):
Current FY (FY1)	FY2	FY3	FY4	FY5	
540	215	200	230	655	1,840

6.2 Other PM Considerations:	NA

7. USE AND MAINTENANCE

Use and maintain software in accordance with the requirements and limitations in Section 3.0 and the following:

7.1 Software User/Administrator Authorization Process:	▪ The SRLM will identify, evaluate and authorize software users and administrators based on need and the knowledge, skills and abilities required as indicated in this document via the SWDS or referenced webpage.
7.2 Authorized Software User Requirements:	▪ Online UTrain Course No. 45412, SWDS-16-03-2333-0089, *Software Data Sheet (SWDS) for CBT Software* ▪ Online UTrain Course No. 45458, UM-15-03-NSF-001, CBT *User Manual,* and ▪ Online UTrain Course No. 52346, AP-15-03-NSF-0007, *CBT Administrative Procedure.* ▪ Must have LANL Unclassified (Yellow) Cryptocard to access
7.3 Authorized Software Users:	▪ As listed on the ES Division Non-SSC Software Inventory
7.4 Authorized Administrator Requirements:	▪ Authorized Software User Requirements, plus ▪ Online UTrain Course No. 12341, MM-15-03-NSF-0009, CBT *Maintenance Manual*, and ▪ Minimum of five (5) years of experience in C++ programming
7.5 Authorized Administrators:	SO, SD (see Section 2)

7.6. Software Risk Register Required?	Yes ☐	No ☒	7.7 Risk Register Minimum	NA

Electronic Copy

SWDS No.:	SWDS-16-03-2333-0089	Rev.:	0

			Review/Frequency:	
7.8 Minimum Assessment Frequency & Method(s):			Bi-annual; P330-3, *Quality Audits*	
7.9 In-Use Testing Required?	Yes ☒	No ☐	7.10 Minimum In-Use Test Frequency & Method(s):	Annual, when installed on a different computer, and when significantly changed; SWTP-15-03-NSF-0008, *CBT Software Test Plan*
7.11 Problem Reporting and Corrective Action (PR&CA)	☒ P330-6, *Nonconformance Reporting* and/or P322-4, *Laboratory Performance Feedback and Improvement Process*			
	☐ Other (Describe): NA			
7.12 Other	▪ The CBT software produces various operational event documentation, including self-diagnostic test results. As part of assessments review operational event documentation for issues or issue precursors and take actions as required. ▪ The operational event documentation should be retained for a minimum of four years within the computer program and backed up to the DCS02 server at least annually. The documentation is not a quality record.			

8.0 ATTACHMENT LIST

8.1 Attachment No.	8.2 Attachment Title
A1	CBT Software Cost Estimate Supporting Detail, Rev. 0

9.0 SWDS REVISION HISTORY

9.1 Revision	9.2 Revision Date	9.3 Revision Description and Reason for Revision
0	Last approval date in Section 2.0	Initial issue

Fictitious Data

SWDS No.:	SWDS-16-03-2333-0089	Rev.:	0

ATTACHMENT

Attachment A1: 09/17/2016, CBT Cost Estimate Supporting Detail, Rev. 0

Item No:	Item Name:	Rough Estimated Cost, K$:					
		Fiscal Year (FY):					Total, K$ (FY17 – FY21):
		2017	2018	2019	2020	2021	
1	Design/Acquire Software	200	0	0	0	235	435
2	Design/Acquire Hardware	0	0	0	0	25	25
3	Install/Approve for Use	150	0	0	0	175	325
4	Use and Maintain	190	215	200	230	220	1,055
Total		**540**	**215**	**200**	**230**	**655**	**1,840**

Key Cost & Schedule Clarifications:

1. Costs are fully burdened at 25% of direct costs and escalated at 4% per year.
2. Design/Acquire Software costs include documentation update costs.
3. Software upgrade costs includes cost to upgrade existing version of Combustible CBT server and workstation support software (e.g., Windows Server 2016 and MS Office Outlook) to newer versions.
4. Maintenance cost includes 0.5 FTE SO Level of Effort (LOE) at $180 /hour (burdened); and a bi-annual software quality assessment at an average $20k per assessment. Maintenance costs do not include annual supplier service contract maintenance, notifications knowledge base updates.
5. Hardware costs include costs for servers, workstations, field scanners, firewall, and router see hardware listing in SWDD-15-03-NSF-0006.
6. All work is planned to begin in 1st quarter of shown FY and end in third-quarter of same FY to allow for a three month schedule contingency within each FY.
7. The SO will report issues of concern/deviations in scope, schedule or cost that adversely affect the software plan to the SRLM as part of the reporting processes associated with use of the software (e.g., issues with software scope, schedule or cost for software used in the facility design/analysis of facilities is reported as part of the project/modification that requires the design/analysis.
8. FY17 hardware costs covered by ES Division Office as part of ES Server 0294 software FY17 upgrade.

Fictitious Do

Software Baseline Form (SWBL)

SWBL No.: Rev.:

1.0 SOFTWARE INFORMATION					
1.1 Software Name and Version:					
1.2 TA No.:		1.3 Facility No.:		1.4 Facility Name:	
1.5 SWID No.:			1.6 Equipment ID:		
1.7 Safety Basis Document(s) and Control(s) Supported:					

2.0 SOFTWARE BASELINE – DOCUMENTS			
2.1 Document Number (including revision)	2.2 Document Title	2.3 Document Description	2.4 Document File Location:

3.0 SOFTWARE BASELINE – COMPUTER PROGRAMS		
3.1 Computer Program File/Folder Name (including revision)	3.2 Computer Program Description	3.3 Computer Program File Location:

4.0 PREPARED BY/REVIEWED BY/APPROVED BY
4.1 Prepared By – Software Owner (SO), *(Name, Z number, Organization, Signature and Date):*
4.2 Reviewed and Approved By – Software Responsible Line Manager (SRLM), *(Name, Z number, Organization, Signature and Date):*
4.3 Reviewed and Approved By – Others as required by SRLM, *(Name, Z number, Organization, Signature and Date):*

5.0 ATTACHMENT LIST	
5.1 Attachment No.	5.2 Attachment Title

6.0 REVISIONS		
6.1 Revision	6.2 Date	6.3 Change Description and Reason for Change

GENERAL

This form is provided to promote the successful and consistent implementation of the software baseline (SWBL) requirements of STD-342-100, *LANL Engineering Standards Manual, Chapter 21, Software*.

A software baseline (SWBL) is a specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved change control process (Ref. NQA-1). A software baseline includes the computer program (code and [configuration] data) and the computer program documentation. The computer programs include the computer program code, and as applicable, configuration/data files and support software.

For software that has been approved for use, the SWBL may be considered the information and computer program files that are needed to run the software for a specific application (a complete backup). See attached SWBL example for items that may be listed as part of a SWBL.

If entering sensitive information (e.g., Unclassified Controlled Nuclear Information -UCNI), ensure proper Derivative Classifier/Reviewing Official reviews are performed and appropriate markings applied.

For additional detail on developing a software baseline (SWBL), see *Chapter 21, SOFT-GEN.*

LANL personnel: Endeavor to use Chapter forms as-is and report issues and improvement ideas to the Chapter 21 POC. POC may authorize other methods equivalent to chapter forms in writing.

LANL subcontractors must use Chapter forms to satisfy Chapter requirements for SSC software. For Non-SSC software, subcontractors may either use their own forms or integrate, adapt, and reformat the forms; either approach is acceptable so long as key functions, data, and approvals are retained.

HEADER

Field	Entry Information
Software Name	Enter the name of the software in the upper right hand corner of the header as shown in the example form (e.g., Software Baseline for CBT).
SWBL No.	Enter the document number assigned to the SWBL. The number follows that of the SRLM's document control/records management process. For software used in the operation of existing or new LANL facilities, use the numbering scheme and process specified in AP-341-402, *Engineering Document Management in Operating Facilities,* e.g., SWBL-03-0216-1234. *The CoE SharePoint Document Numbering Site currently supplies the final, unique, sequential number for the string.*
Rev.	Enter the SWBL revision number. Initial approved issue of the software baseline is 0.

1.0 SOFTWARE INFORMATION

Field	Entry Information
1.1	Enter the name and version of the software.
1.2	Enter the two-digit Technical Area (TA) number(s) where the software is used. If used sitewide, enter 99.
1.3	Enter the facility number(s) where the software is used. Follow AP-341-402

Field	Entry Information
	conventions for utilities, multiple, etc. (*e.g., if multiple, enter MULT…)*
1.4	Enter the facility name(s) where the software is used.
1.5	Enter the software identification number (SWID). The SWID should be the same SWID as on the Form 2033. Keep SWID the same through minor revisions, and most major revisions unless the major revision is so different that the SRLM wishes to obtain a new SWID. SWIDs are obtained in accordance with AP-341-402, *Engineering Document Management in Operating Facilities.* Ensure the SWID is part of the SWBL and subsequent software documentation.
1.6	For SSC software, enter the equipment identification (CLI) number associated with the software per ESM Chapter 1, Section 200 (e.g., 03-0216-HVAC-BAS-1). For Non-SSC software, enter NA.
1.7	As applicable, identify the safety basis documents (e.g., Documented Safety Analysis; Safety Analysis Document) and controls therein that the software supports. Provide section references. If not applicable, indicated NA.

2.0 SOFTWARE BASELINE - DOCUMENTS

Field	Entry Information
2.1	Enter the document number of the document that must be controlled as part of a baseline. Examples include software design requirements, instructions for computer program use, test plans, documents that describe the system hardware configuration necessary to run the software, etc. Obtain and enter the number in accordance with the SRLM's document control records/management system. Include revision number. ***Note***: It is not necessary to reenter safety basis documents identified in Field 1.7.
2.2	Enter the document title.
2.3	Enter a description of the document.
2.4	Enter the location of the files. If attaching the files to the SWBL, enter the attachment name and attach electronic copies of the document files (e.g., on a compact disc or other media). Ensure the file names match the names listed in the SWBL. ***Note***: Methods other than attaching the files (e.g., hyperlinking to documents in an electronic data management system [EDMS]), including a screenshot of the file location etc., may be used as long as they conform to the governing document control/records management process and ensure file confidentiality, integrity and availability/retrievability. For Read Only SSC computer programs that cannot be obtained, indicate that the computer program file location is within the SSC and/or the associated SSC spare part.

3.0 SOFTWARE BASELINE – COMPUTER PROGRAMS

Field	Entry Information
3.1	Enter or attach (e.g., screenshot of the file directory) the computer program name, revision, file extension (e.g., exe) of the computer program that must be controlled as

Field	Entry Information
	part of the baseline. If the file date is part of the revision number/unique file identifier, then include the file date. If the file date is not part of the unique file identifier (e.g., some file dates change when copied even when no changes to the file are made), then do not include the file date. Include the computer program code files and as applicable, configuration files and support software. Examples include source files, object files, database files, etc.
3.2	Enter the computer program description. Describe what it does/how it supports the overall use of the software.
3.3	Enter the location of the files. If attaching the files to the SWBL, enter the attachment name and attach electronic copies of the computer program files (e.g., on a compact disc or other media). Ensure the file names match the names listed in the SWBL. See note in 2.4 for alternate methods.

4.0 PREPARED BY/REVIEWED BY/APPROVED BY

Field	Entry Information
4.1	Enter the SWBL preparer's (SO) name, Z number, organization, signature, and date. For assistance in determining the SO, see SOFT-GEN Appendix C: *SO and SRLM Decision Diagram for FAC-COE.*
4.2	Review and if acceptable, approve by entering SRLM name, Z number if applicable, organization, signature, and date. For assistance in determining the SRLM, see SOFT-GEN Appendix C: *SO and SRLM Decision Diagram for FAC-COE.*
4.3, 4.n+1…	Add additional reviewers/approvers as required by the SRLM and/or the SRLM's document control process. Review and as appropriate, approve by entering additional reviewer/approver name, Z number, organization, signature, and date. Enter none or not applicable (NA) if there are no additional reviewers/approvers.

5.0 ATTACHMENTS

Field	Entry Information
5.1	List attachments as appropriate. Enter the attachment number.
5.2	Enter the attachment title.

6.0 REVISIONS

Field	Entry Information
6.1	Enter the SWBL revision number and update the revision number in the form header.
6.2	Enter the date of the SWBL revision.
6.3	Enter a summary description of the SWBL change and the reason for the change.

1.0 SOFTWARE INFORMATION					
1.1 Software Name and Version:	Combustible Material Tracker (CBT) Software, V5.2				
1.2 TA No.:	03	1.3 Facility No.:	2333	1.4 Facility Name:	Nonproliferation Security Facility (NSF)
1.5 SWID:	03-2333-0047	1.6 Equipment ID:	NA		
1.7 Software Description:					
The software is used as a tool to track combustible material as part of the NSF combustible material tracking system.					
1.8 Safety Basis Document(s) and Control(s) Supported:					

2.0 SOFTWARE BASELINE – DOCUMENTS		
2.1 Document Number (including revision)	2.2 Document Title	2.3 Document Description
SWDS-15-03-NSF-0001	CBT Software Data Sheet (SWDS)	Documents software-specific information to supplement ESM Chapter 21
SWRS-15-03-NSF-0002	CBT Software Requirements Specification	Requirements document
SWTM-15-03-NSF-0003	CBT Software Requirements Traceability Matrix	Traceability document that traces requirements from the SRS through the software lifecycle.
SWRL-15-03-NSF-0004	CBT Software Risk List (Register)	Listing and ranking of risks for the software throughout its lifecycle.
SWHA-15-03-NSF-0005	CBT Software Hazard Analysis	Hazard analysis of software and integrated software/hardware system.
SWDD-15-03-NSF-0006	CBT Software Design Document	Software design. Includes system configuration including associated hardware and facility interfaces and key support software configuration
SWTP-15-03-NSF-0007	CBT Software Acceptance Test and In-Use Test Plan	Test plan document that describes acceptance and in-use tests.
SWTR-15-03-NSF-0008	CBT Software Acceptance Test Report	Describes acceptance test results and acceptance.

Fictitious Data

2.0 SOFTWARE BASELINE – DOCUMENTS		
2.1 Document Number (including revision)	2.2 Document Title	2.3 Document Description
SWMM-15-03-NSF-0009	CBT Software Maintenance Manual	Detailed technical manual for software subject matter expert (SME) to use for troubleshooting and maintaining the software.
SWUM-15-03-NSF-0010	CBT Software User Manual	User manual. Includes general instructions for use, including limitations.
AP-15-03-NSF-0007	CBT Software Administrative Procedure	Administrative procedure that integrates the software into the facility day-to-day operations.
SWLI-15-03-SNF-1110	CBT Software License/License Key	License and license key required to use the software
2.4 Document file location:		
Attachment 1: CBT SWBL-15-03-2332-0234 Document Files.		

3.0 SOFTWARE BASELINE – COMPUTER PROGRAMS	
3.1 Computer Program File/Folder Name (including revision)	3.2 Computer Program Description
CBT 1.02.exe	Primary executable computer program that provides overall system function
CBT Master Database 1.2. mdb	The primary database of material.
Microsoft Windows XP Professional SP1.1	Off the shelf (OTS) support software that is the operating system on client computers.
Microsoft Server 2003 R2 Folder	OTS support software that is the operating system on computer servers.
Microsoft Access Database 2000 SP1.01 Folder	OTS support software that provides data storage on servers and client computers.
Microsoft Outlook 2013 SP1.0 Folder	OTS support software that provides email notifications of transactions.
3.3 Computer program file location:	
Attachment 2: CBT SWBL-15-03-2332-0234 Computer Program Files.	

4.0 PREPARED BY/REVIEWED BY/APPROVED BY
4.1 Prepared By – Software Owner (SO), *(Name, Z number, Organization, Signature and Date):*
Tom Thompson, 143987, NEN-2, 10/1/2015
4.2 Reviewed and Approved By – Software Responsible Line Manager (SRLM), *(Name, Z number, Organization, Signature and Date):*
Michael Bradley, 114987, NEN-2, 10/2/2015
4.3 Reviewed and Approved By – Others as required by SRLM, *(Name, Z number, Organization, Signature and Date):*
None Required by SRLM (Not Applicable)

5.0 ATTACHMENT LIST

5.1 Attachment No.	5.2 Attachment Title
1	CBT SWBL-15-03-2332-0234 Document Files
2	CBT SWBL-15-03-2332-0234 Computer Program Files

6.0 REVISIONS

6.1 Revision	6.2 Date	6.3 Change Description and Reason for Change
0	10/02/2015	Original Issue

Fictitious Data

Non-SSC Software Change Package Form (SWNCP)

SWNCP No.:	Rev.:	Page 1 of 3

1.0 SOFTWARE CHANGE REQUEST INFORMATION (COMPLETED BY REQUESTOR)

1.1 Software Name and Version:		1.2 Request Date:	Click here to enter a date.	1.3 Requested By Date:	Click here to enter a date.

1.4 Requestor:*(Name, Z Number if applicable, Organization)*	
1.5 Software Owner (SO): *(Name, Z Number if applicable, Organization)*	
1.6 Software Change Request Title:	

1.7 TA No.:		1.8 Facility No(s):		1.9 Facility Name(s):	

1.10 Type of Reason for Change:	☐ Defect	☐ Requirement Change

1.11 Reason for Change Request:

2.0 SOFTWARE CHANGE REQUEST DEVELOPMENT (COMPLETED BY SO)

2.1 SWID No.:		2.2 Software Designation:	Select from drop down menu	2.3 Assoc. Management Level (ML):	Select from drop down menu

2.4 Type of Change:	☐ Major	☐ Minor

2.5 Change Description:

2.6 Software Change Impact/Risk Description:

2.7 Affected Software Baseline - Documents

Document No. (including revision)	Document Title	Document Description

2.8 Affected Software Baseline – Computer Programs

Computer Program File Name (Including revision)	Computer Program Description

2.9 Planned Type of Verification and Validation (V&V) for the Change

☐ Review(s): ☐ Test(s): ☐ Alternate Calculation(s):

2.10 Verification and Validation (V&V) Description:	
2.11 Is this software used for control in a hazard category 2 or 3 nuclear facility, high hazard nonnuclear facility, moderate hazard nonnuclear facility, and/or accelerator facility? If "No" is indicated, go to Step 3.1. If "Yes" is indicated, go to Step 2.8.	Select from drop down menu
2.12 Does this software change require engagement of the evaluation processes of SBP-15-351, *Design Basis or Safety Basis Change Review* or other safety basis (SB) reviews? If "No" is indicated, go to Step 3.1. If "Yes" is indicated, complete safety basis evaluation(s), before approval for use (Section 6.) and go to Step 2.9. Consider completing evaluations before proceeding further.	Select from drop down menu
2.13 Design basis and/or safety basis evaluation(s) document number(s):	

3.0 SOFTWARE CHANGE REQUEST REVIEW AND ACCEPTANCE/REJECTION (COMPLETED BY SRLM AND AS REQUIRED, FDAR OR DA)	
3.1 Software Responsible Line Manager (SRLM) Review and Acceptance/Rejection	☐ Accepted ☐ Accepted with Comments ☐ Rejected
If "Accepted with Comments" or "Rejected" is indicated, then provide comments or justification below:	
3.2 SRLM: *(Name, Z Number if applicable, Organization, Signature, and Date)*	
3.3 FDAR or DA Review and Acceptance (required only for major changes of software used for administrative control in a LANL facility) If "Accepted with Comments" or "Rejected" is indicated, then provide comments or justification below:	☐ NA ☐ Accepted ☐ Accepted with comments ☐ Rejected
3.4 FDAR(s) or DA : *(Name, Z Number, Organization, Signature, and Date)*	

4.0 SOFTWARE CHANGE IMPLEMENTATION (COMPLETED BY SO)	
4.1 Change Implementer(s): *(Name, Z Number if applicable, Organization, and Date of Implementation)*	
4.2 Design Organization Reviewing Software Design Change:	Select from drop down menu

4.3 Implemented Change Description:

5.0 SOFTWARE CHANGE VERIFICATION AND VALIDATION (V&V) & REQUEST FOR APPROVAL FOR USE (COMPLETED BY SO)

5.1 The software change has been V&V'd in accordance with the attached V&V planning documentation. The approval for use requirements of Chapter 21 associated with this change have been satisfied. The revised software baseline (without SRLM signature), V&V results, and as required, other supporting approval for use documentation associated with this change is attached. Approval for use is requested.
SO:*(Name, Z Number if applicable, Organization, Signature, and Date)*

6.0 CHANGE ACCEPTANCE AND APPROVAL FOR USE (COMPLETED BY SRLM AND AS REQUIRED, FDAR OR DA)

I (a) accept the software change; (b) attest that the software change and the approval for use requirements of Chapter 21 have been satisfied; and, (c) attest that the software is ready to be used in its intended operating environment.	
6.1 Software Responsible Line Manager (SRLM) Review and Acceptance	☐ Approved for Use ☐ Rejected
If "Rejected" is checked, then provide reason below:	
6.2 SRLM: *(Name, Z Number if applicable, Organization, Signature, and Date)*	
6.3 FDAR or DA Review and Acceptance (for major changes only)	Not Applicable (NA) ☐ Approved for Use ☐ Rejected ☐
If "Rejected" is checked, then provide reason below:	
6.4 FDAR or DA: *(Name, Z Number, Organization, Signature, and Date)*	

7.0 ATTACHMENTS

7.1 Attachment No.	7.2 Attachment Title

8.0 REVISIONS

8.1 Revision No.	8.2 Date	8.3 Revision Description and Reason for Revision
	Click here to enter a date.	

GENERAL

This form is provided to promote the successful and consistent implementation of the computer program change requirements of Engineering Standards Manual (ESM) STD-342-100, *Chapter 21, Software (Chapter 21).*

These instructions are for completing the form and are not part of the completed form.

LANL personnel: Endeavor to use Chapter forms as-is and report issues and improvement ideas to the Chapter 21 POC. POC may authorize other methods equivalent to chapter forms in writing.

LANL subcontractors must use Chapter forms to satisfy Chapter requirements for SSC software. For Non-SSC software, subcontractors may either use their own forms or integrate, adapt, and reformat the forms; either approach is acceptable so long as key functions, data, and approvals are retained.

If entering sensitive information (e.g., Unclassified Controlled Nuclear Information -UCNI), ensure proper Derivative Classifier/Reviewing Official reviews are performed and appropriate markings applied.

This form does not apply to SSC Software. For SSC software, use AP-341-507, *SSC Software Control Software Change Package* or AP-341-519, *Design Revision Control (see Ch 21 SOFT-GEN for discussion).*

Any person may request changes (Requestor). The Requester completes Section 1.0 and provides the form to the Software Owner (SO) who then further develops the software change request and engages the SRLM and as required, FDAR or DA for review and approvals as indicated on the form.

This form does not apply to Less Than Minor computer program changes as defined in Chapter 21. At time of writing, Chapter 21 (SOFT-GEN) defines these as:

Less Than Minor Computer Program Change. A change that is not a major or minor computer program change and:

- adds, deletes and/or modifies ML-4 performance function code,
- adds, deletes and/or modifies code that does not modify performance function (all MLs), or
- imparts changes without adding, deleting or modifying design and/or analysis output values (all MLs).

Examples: Install security patch/service pack updates. An OTS software patch that includes a code change to prevent a screen from “freezing” or loading slowly (all MLs). Add/modify code clarifying notes (all MLs). Modify code to produce multiple reporting formats (all MLs).

For less than minor computer program changes, ensure they are made by a competent individual knowledgeable in the software. If feasible, add a brief description, name of who is making, justification that the change does not modify performance functions (all MLs), and the date that the less than minor change is made in the computer program code at the time the change is made. *If it is not feasible to document the less than minor change in the code, then document in associated operational logbooks, etc.*

Final Actions: Upon completion of changes, the SRLM is responsible for:

(a) Incorporating changes into the software baseline as soon as practical after changes are approved, and

(b) retaining Non-SSC Software Change packages as a record in accordance with P1020-1, *Laboratory Records Management* or the governing records management process.

HEADER (COMPLETED BY REQUESTOR)

Field	Entry Information
Software Name	Enter the name of the software in the upper right hand corner of the header (e.g., Non-SSC Software Change Package Form (SWNCP) *for Ceasar II Software)*.
SWNCP No.	Number this form in accordance with the governing document control procedure. For LANL owned Non-SSC software in the scope of the Facility Conduct of Engineering (FAC-COE), use the numbering scheme and process specified in AP-341-402, *Engineering Document Management in Operating Facilities*.
Rev.	Enter the SWNCP revision number. Enter "0" for initial revision and 1, 2, etc. for subsequent revisions.

1. SOFTWARE CHANGE REQUEST INFORMATION (COMPLETED BY REQUESTOR)

Field	Entry Information
1.1	Enter the software name and version to be changed. Use the version number before the change is implemented.
1.2	Enter the date the software change request was initiated.
1.3	Enter the target date that the requestor would like to have the software change completed.
1.4	Enter the name, Z number if applicable, and organization of the person requesting the change (Requestor). This should typically be the SO although others may also be Requestors. For assistance in determining the SO, see SOFT-GEN Appendix C: *SO and SRLM Decision Diagram for FAC-COE.*
1.5	Enter the name, Z number if applicable, and organization of the SO. If the Requestor is the SO, indicate "Same as 1.4" or similar text.
1.6	Enter a specific and succinct change request title. If desired, include the version before the change (e.g., "Correct POSTMAX Version 2.0 Test Comparison Executable to Run Test Case 7 Without Aborting").
1.7	Enter the LANL Technical Area (TA) number of the TA associated with the software change. If the software is used at various TAs, enter TA 99.
1.8	Enter the facility number(s) of the facility associated with the software change. Follow AP-341-402 conventions for utilities, multiple, etc. (e.g., i*f multiple, enter MULT…)*
1.9	Enter facility name(s) of the facility associated with the software change. If the software is used at various LANL facilities, enter "multiple".
1.10	Enter the type of reason for the change request. Check all that apply.

Field	Entry Information
	Defect: 1. a problem which, if not corrected, could cause an application to either fail or to produce incorrect results. 2. an imperfection or deficiency in a project component where that component does not meet its requirements or specifications and needs to be either repaired or replaced. (Ref. ISO/IEC/IEEE 24765:2010-12-15, Systems and Software Engineering – Vocabulary). **Requirement Change:** New and/or modified requirements.
1.11	Describe the reason (rationale) for the change request. Describe "why" the change is needed. Enter the Nonconformance Report (NCR) name and number or other problem/error report that may apply. Indicate what could happen if the change is not implemented. If known, provide a high-level description of the desired changes.

2. SOFTWARE CHANGE REQUEST DEVELOPMENT (COMPLETED BY SO)

Field	Entry Information
2.1	Enter the software identification number (SWID). The SWID should be the same SWID as on the Form 2033. Keep SWID the same through minor revisions, and most major revisions unless the major revision is so different that the SRLM wishes to obtain a new SWID. SWIDs are obtained in accordance with AP-341-402, *Engineering Document Management in Operating Facilities*.
2.2	Select the software designation (safety software, risk significant software (RS) or commercially controlled software (CC)) from the drop down menu. Select only one designation.
2.3	Select the associated Management Level (ML- 1, 2, 3 or 4) from the drop down menu. If used for multiple ML applications select the highest risk (lowest number), (e.g., for software used in ML-2 and ML-3 applications, enter ML-2.)
2.4	Indicate whether the change is a Major or Minor change by checking either the major or minor box. If Major and Minor changes are to be made, select major. Note that a change can include a change to the computer program (including data) or changes to documentation. Use Chapter 21 definitions. At the time of writing, Chapter 21 definitions are as follows**:** **Major Computer Program Change:** A change that: ▪ the Software Responsible Line Manager (SRLM) or computer program supplier designates as a major change, ▪ adds or deletes an ML-1, ML-2 or ML-3 SSC performance function, ▪ modifies ML-1 or ML-2 SSC performance function code, excluding clarifying notes, ▪ adds, deletes or modifies design and/or analysis output values of ML-1, ML-2 or ML-3 calculations, ▪ recodes to another language, or ▪ modifies a significant number of lines of code. ***Note:*** A Major OTS computer program change is often indicated with increment increase in

Field	Entry Information
	version number (e.g., change from version 1 to 2). An evaluation of the software however, is required to determine whether the version release is a major change. Examples: A change in the algorithm or databased used for calculating the water flow rate in an ML-3 fire protection piping system design. A change in coding language from C to C++. A version change where 40% of the lines of code were modified. **Minor Computer Program Change:** A change that is not a major computer program change and: ▪ adds or deletes an ML-4 SSC performance function, ▪ modifies ML-3 SSC performance function code, excluding clarifying notes, or ▪ adds, deletes or modifies design and/or analysis output values of ML-4 calculations ***Note:*** A minor OTS software change is often indicated with a fractional increase in version number (e.g., 1.1 or 1.01). An evaluation of the software however, is required to determine whether the version release is a minor change. Example: Change the algorithm in ML-2 software for calculating the current that flows in an ML-4 electric power system under abnormal conditions. **Minor Document Change.** A document change, as defined by the governing document control program, that includes but is not limited to inconsequential editorial corrections, grammatical and spelling changes, organizational name and acronym changes, and similar type changes. A minor document change does not include revisions, changes, or modifications to a document (e.g., procedure, work instruction, drawing, etc.) which impact the effective implementation of the requirement(s). (Ref. P1020-2)**.** ***Note:*** Minor document changes alone (i.e., without an associated major document change and/or a major or minor computer program change) do not require a SWNCP. **Major Document Change.** A document change that is not a minor document change.
2.5	Develop the software change. Describe "what" the change is to sufficient detail to allow a reviewer to understand, review and as appropriate, approve the change request.
2.6	Describe the impacts and potential risks associated with the change. Impacts should include technical as well as estimated commercial impacts (cost, schedule, resources, etc.).
2.7	List affected software baseline documents. Provide document number, revision, document title, and document description.
2.8	List affected software baseline computer programs (including database, configuration files). Provide computer program filename, version, and computer program description.
2.9	Evaluate and indicate the planned type of verification and validation (V&V) for the change by checking the review(s), test(s) and/or alternate calculations(s) box. Check all that apply.
2.10	Describe the V&V to sufficient detail to allow a reviewer to understand, review and approve the change request and the planned approach to V&V. Describe the approach to retesting

Field	Entry Information
	and acceptance of test results. ***Note:*** This description is a high level planned approach/strategy. The final V&V approach and details must be subsequently developed, reviewed and approved.
2.11	Indicate whether the software is used for control in a hazard category 2 before 3 nuclear facility, high hazard nonnuclear facility, moderate hazard nonnuclear facility, and/or accelerator facility by selecting either "Yes" or "No" from the drop down menu. If "No" is indicated, go to Step 3.1. If "Yes" is indicated, go to Step 2.8.
2.12	Indicate if a safety basis (SB) evaluation of the change is required per the applicable SB procedures listed on the SB website by selecting either "Yes" or "No" from the drop down menu. (e.g., SBP-15-351, *Design Basis or Safety Basis Change Review,* SBP113-3, *Unreviewed Safety Issue Process.)* If "Yes" is indicated, then the SO ensures the required actions are taken, attaches evidence of completion prior to submitting the SWNCP for approval for use (Section 6.0) and ultimately, use of the software in its intended (working) environment. Proceed to Step 2.9. If "No" is indicated, then proceed to Step 3.1. As required, contact a Facility Design Authority Representative (FDAR) or a safety basis representative for assistance.
2.13	Enter the attached design basis and/or safety basis document numbers associated with the evaluation(s) (e.g., per SBP-112-3, USQD TA55-16-0234-D). If not applicable, enter "NA".

3. CHANGE REQUEST REVIEW AND ACCEPTANCE (COMPLETED BY SRLM AND AS REQUIRED, FDAR OR DA)

Field	Entry Information
3.1	The SRLM evaluates the change request and indicates "Accepted", "Accepted with Comments", or "Rejected". Provide comments when "Accepted with Comments" is indicated. Provide reason(s) for rejection when "Rejected" is indicated. ***Note:*** **"**Accepted" means that the requested change is acceptable as proposed. "Accepted with Comments" means that the change is authorized as described in comments.
3.2	Enter the SRLM name, Z number if applicable, organization, signature, and date.
3.3	If the change is a major change for Non-SSC software that is used for administrative control in a LANL facility, then the FDAR or DA evaluates the change request and indicates either "NA", "Accepted", "Accepted with Comments", or "Rejected". ***Note:*** Where there are multiple FDARs, the LANL Facility Design Authority, or a representative sampling of FDARS (but no fewer than two) must review and sign the request. ***Note:*** **"**Accepted" means that the requested change is acceptable as proposed. "Accepted with Comments" means that the change is authorized as described in comments. If the software change is a minor change or a change to software that is not used for administrative control in a LANL facility, then the SRLM indicates "NA" and proceeds to Step 4.1.
3.4	Enter the FDAR(s) or DA name, Z number, organization, signature, and date for all

Field	Entry Information
	decisions other than "NA" in 3.3. If "NA" in 3.3, enter "NA". If the change is rejected, retain the rejected change request as a record in accordance with P1020-1, *Laboratory Records Management* or the governing records management process.

4. CHANGE IMPLEMENTATION (COMPLETED BY SO)

Field	Entry Information
4.1	Make the change and enter who implemented the change(s) (Change Implementer(s)). Provide name, Z number if applicable, organization name and date of implementation.
4.2	For software design changes, select from the drop down menu whether the change was reviewed by the "Original Design Organization" or an "Authorized Alternate Design Organization". Section 6 authorizes the alternate organization. If the change did not involve design (e.g., upgrade from version 2.0 to 3.0 of off-the-shelf (OTS) software), enter NA. ***Note:*** The design organization approving the change shall have demonstrated competence in the specific design area of interest and have an adequate understanding of the requirements and intent of the original design.
4.3	Describe the final implementation of the change (e.g., as applicable, provide/attach the relevant computer program listings, including pseudo code). As applicable, include any limitations and changes to use and maintenance processes. For software document changes, provide a detailed description of the document revisions, or attach/hyperlink to the revised documents. Ensure the document revision history describes the associated change(s).

5.0 VERIFICATION AND VALIDATION (V&V) AND READINESS (COMPLETED BY SO)

Field	Entry Information
5.1	Plan and complete V&V, including retesting as required and acceptance of test results. Take actions and provide information to demonstrate conformance with Chapter 21 approval for use requirements. Example information includes test reports, revised SWBL, revised computer program listings, training records, completed Unreviewed Safety Question (USQ) documentation, implementation sequence, etc. Control changes. Attach (or list documents with hyperlinks to) V&V plan documentation; V&V results documentation; and, completed but without the SRLM signature, the revised software baseline (SWBL). For software changes used for ML-1, ML-2 or ML-3 administrative control in a LANL facility, also include the planned implementation sequence. Base the extent of V&V on the complexity of the software, the degree of standardization, the similarity with previously approved software, and the importance to safety. Major changes must be acceptance tested per SOFT-V&V. For minor changes, if review alone is inadequate to determine if requirements are satisfied, use alternate calculations and/or develop and integrate (interim) tests to support the review. For computer programs that were originally commercially dedicated (CGD'd): CGD major changes. CGD minor changes if the change affects the critical characteristic(s) that pertain to the functionality of the computer program and as applicable, the associated SSC.

6.0 CHANGE ACCEPTANCE AND APPROVAL FOR USE (COMPLETED BY SRLM AND AS REQUIRED, FDAR OR DA)

Field	Entry Information
6.1	The SRLM evaluates the change request information to ensure Chapter 21 software change and approval for use requirements have been satisfied and that the software is ready to be used in its intended operating environment. Ensure that interfaces with software systems, organizations and/or activities that the software is based on or otherwise has an interface with, are controlled such that unintended consequences with such systems/organizations do not occur (e.g., coordinate with cybersecurity personnel per PD210, *Cybersecurity Program* or CIO-P100, *Cyber Security Risk Management Plan*). If an alternate design organization reviewed the change, ensure the design organization approving the change has demonstrated competence in the specific design area of interest and has an adequate understanding of the requirements and intent of the original design. Check the appropriate decision box (either "Approved for Use" or "Rejected"). Provide reason(s) for rejection when the "Rejected" box is checked. For assistance in determining the SRLM, see SOFT-GEN Appendix C: *SO and SRLM Decision Diagram for FAC-COE.*
6.2	Enter the SRLM name, Z number if applicable, organization, signature, and date on the SWNCP. If the change is approved for use, enter SRLM information and sign the attached revised SWBL. If rejected, the SRLM retains the rejected SWNCP as a record in accordance with P1020-1, *Laboratory Records Management* or the governing records management process.
6.3	If the software change is a minor change or, a major change for software that is not used for administrative control in a LANL facility, then the SRLM checks the "NA" decision box. If the change is a major change for Non-SSC software used for control in a LANL facility, then the FDAR evaluates the change and implementation sequence of the change in accordance with AP-341-621, *Design Authority Technical Review* and checks the appropriate decision box. ***Note:*** Where there may be multiple FDARs, the LANL Facility Design Authority (DA), or a representative sampling of FDARS (but no fewer than two) must review. Provide reason(s) for rejection when "Rejected" box is checked. If Rejected, the SRLM retains the SWNCP as a record in accordance with P1020-1, *Laboratory Records Management* or the governing records management process.
6.4	If "NA" in 6.3 is checked, then enter "NA". For other than "NA" in 6.3, enter the FDAR or DA name, Z number, organization, signature, and date. The SWNCP is complete. The SRLM retains the SWNCP as a record in accordance with P1020-1, *Laboratory Records Management* or the governing records management process.

7.0 ATTACHMENTS

Field	Entry Information
7.1	Enter the attachment number for each attachment.
7.2	Enter the attachment title.

8.0 REVISIONS

Field	Entry Information
8.1	If the SWNCP requires revision, then revise in accordance with the governing document control program. Enter the revision number: 1, 2, etc. as appropriate.
8.2	Enter the SWNCP revision approval date.
8.3	Describe the revision and why the revision was made.

TABLE OF CONTENTS

1.0 PURPOSE, SCOPE, AND APPLICABILITY 1
2.0 KEY DEFINITIONS AND ACRONYMS 5
3.0 SSC SOFTWARE ACQUISITION 5
3.1 ML-4 SSC Software 5
3.2 ML-1 through ML-3 SSC Software 5
4.0 NON-SSC SOFTWARE ACQUISITION 8
4.1. Existing Non-SSC Software (All MLs) 8
4.2. New ML-4 Non-SSC Software 9
4.3. New Software Used in SSC Design that is "Simple and Easily Understood" 9
4.4. Other New Non-SSC Software 10

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Eliminated CGD for ML-3 associated (SRL3) safety software; clarifications.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

1.0 PURPOSE, SCOPE, AND APPLICABILITY

This section, SOFT-ACQUIRE, describes the "how, when, and who" for acquiring software and software services. See Table 21.2-1 for a summary.

See Chapter 21 applicability in SOFT-GEN. In addition:

1. This section applies to workers that are acquiring software. See SOFT-GEN for definition of acquired software.[1]

2. To apply this section, the software must first be identified and a draft or final software determination (Form 2033) completed. See SOFT-GEN.

3. Freeware: Freeware cannot be acquired without the approval of the Chapter POC or designate(s).[2] Exceptions: Uncomplicated software tools (described in Chapter 21 applicability in SOFT-GEN), toolbox codes, "Simple and Easily Understood" software used in the design of SSCs, software from other organizations within the DOE complex, and/or software acquired from other government agencies (e.g., US EPA, Army Corps of Engineers). Note: P1011, *Computer Hardware, Software, and Services: Exceptions to Acquisition, Maintenance, and Support* has additional requirements for freeware.

[1] Software developed by a LANL owner (self-developed) would not follow this section, but would follow all others.
[2] Should be necessary for LANL mission accomplishment and where no feasible alternative solutions are available.

4. See SOFT-V&V for verification and validation, including commercial grade dedication; see SOFT-MAINT for maintenance of acquired software.

5. Software acquisition includes associated software services throughout this section.

Note on software acquired from an internal-to-LANL supplier (i.e., designed/developed in-house by another LANL organization):
Requirements herein related to contracts and procurements may not apply exactly as written, so user discretion is allowed. Follow applicable concepts and adapt the others as applicable.

Table 21.2-1 SOFT-ACQUIRE Section Summary for SSC and Non-SSC Software
(This is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			Reference
		1, 2	3	4	How	When	Who[2, 3]	ESM Ch. 21 Ref.[8]
I. For SSC Software								
1	Acquire new software	R	Gr	Gr	▪ Evaluate and document whether to acquire from an NQA-1 -- or Non-NQA-1 qualified supplier and commercially dedicate (CGD)[4] ▪ Develop a SWDS, (SOFT-GEN-FM01)[5] ▪ Develop Statement of Work (SOW) [6] ▪ Acquire per P840-1, PD210 and CIO-P100 ▪ Use P-card or Purchase Order (PO) ▪ Develop interface agreement, integrated contract orders, interagency agreements or equivalent[7] ▪ Complete Exhibit H[5]	▪ When needed to perform work	▪ SO (D) ▪ SRLM (R, A)	SOFT-ACQUIRE, 3
II. For Non-SSC Software								
1	Acquire existing LANL software	R	R	Gr	▪ Look for and evaluate existing LANL software ▪ If suitable, acquire existing software from the SRLM and obtain SRLM authorization to use it ▪ Update documentation (e.g. licenses), as applicable to reflect additional authorized user(s)	▪ Look for and evaluate existing LANL software before acquiring new software ▪ When needed to perform work	▪ SU (D) ▪ SRLM (R, A)	SOFT-ACQUIRE, 4

Table 21.2-1 SOFT-ACQUIRE Section Summary for SSC and Non-SSC Software
(This is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			Reference
		1, 2	3	4	How	When	Who[2, 3]	ESM Ch. 21 Ref.[8]
2	Acquire new software	R	Gr	Gr	▪ Determine and document if software is used in SSC design and is "Simple and Easily Understood"; document and verify with each use ▪ Determine if software is a toolbox code ▪ Evaluate and document whether to acquire from NQA-1 --or Non-NQA-1 qualified supplier and commercially dedicate the software (CGD)[4] ▪ Develop SWDS, (SOFT-GEN-FM01)[5] ▪ Develop Statement of Work (SOW) [6] ▪ Acquire per P840-1, PD210 and CIO-P100 ▪ Develop CGD documentation per AP-341-703 ▪ Use P-card or Purchase Order (PO) ▪ Develop inter- or intra- LANL agreements[7] ▪ Complete Exhibit H[5]	▪ When needed to perform work	▪ SO (D) ▪ SRLM (R, A)	SOFT-ACQUIRE, 4

Table Notes:

[1] ML = Associated Management Level per Form 2033. Complete Form 2033 prior to acquisition. R = Required. Gr = Required but graded.

[2] D = Develop or implement; R = Review; A = Approve.

[3] SO = Software Owner. SRLM = Software Responsible Line Manager. SU = Software User

[4] CGD applies to ML-1 and ML-2 safety software only.

[5] Required for ML-1 through ML-3 (safety and risk significant) software or for any ML if the supplier is required to come onsite to install/maintain the software.

[6] Applies to ML-1 through ML-3 software only.

[7] See P840-1 for details. Required for acquisitions from LANL organizations, DOE sites, and/or government agencies.

[8] Ref. = ESM Chapter 21 section name and subsection number (e.g., Section SOFT-GEN, Subsection 3, Software Identification and Determination).

2.0 KEY DEFINITIONS AND ACRONYMS

See SOFT-GEN for key definitions and acronyms.

3.0 SSC SOFTWARE ACQUISITION

3.1 ML-4 SSC Software

A. How.

1. Acquire in accordance with P840-1 *Quality Assurance for Procurements*, PD210 *Cyber Security Program,* CIO-P100, *Cyber Security Risk Management Plan,* and the following:

   a. Procure with a purchase card (P-card) or purchase order (PO) using LANL online systems. Specify software requirements and acceptance criteria.

   b. Complete an Exhibit H if the supplier comes onsite to install and/or maintain the software (ref. P840-1), as required.

   c. Guidance: For software acquired from other organizations within LANL or within the DOE complex, use interface agreements, integrated contracts or equivalent where possible. See P840-1 for details and an example interface agreement.

   d. Protect against suspect/counterfeit items (S/CI) items by procuring from reputable authorized suppliers and ensuring valid software licenses. See P330-9, *Suspect Counterfeit Items (S/CI)* for additional information.

   e. Retain acquisition documentation in accordance with the SRLM's governing records management process.

Note: Ensure the correct SWID (obtained when completing Form 2033) is part of the acquisition documentation. See AP-341-402, *Engineering Document Management in Operating Facilities* for details.

B. When. When software is needed to perform work.

C. Who. The SO develops and the SRLM reviews and approves the acquisition[3].

3.2 ML-1 through ML-3 SSC Software

Acquire software for ML-1 through ML-3 applications in accordance with this Subsection.

[3] Additional review and approval by the requestor's Division Leader and OICO may be required by Acquisition Services Management (ASM). See ASM 3400.00.0410, *Goods or Services Requiring Special Review/Approval* and/or contact an ASM representative.

A. How.

1. ML-1 and ML-2 Safety Software: Acquire from an NQA-1 qualified supplier, or, acquire from a Non-NQA-1 qualified supplier and commercial-grade-dedicate (CGD) the software. See AP-341-703, *Commercial Grade Dedication* for CGD.

   a. Make the acquisition strategy decision above considering the following:

      i. the availability of NQA-1 qualified suppliers (see the LANL Institutional Evaluated Supplier List, (IESL));

      ii. performance history;

      iii. cost, schedule and other commercial factors applicable to the software lifecycle.

2. Develop a software data sheet (SWDS). Use SOFT-GEN-FM01, *Software Data Sheet Form (SWDS)*. Document the NQA-1/Non-NQA-1 acquisition strategy on the SWDS. As required by the SRLM, document supporting rationale on the SWDS or elsewhere.

3. ML-3 and Risk Significant software: May acquire from a Non-NQA-1 qualified supplier, and CGD is not required.[4] *Guidance: Current P840-1 requires receipt inspection; however, use of Exhibit H part number verification (PNV) version is generally adequate because V&V at installation ensures quality.*

4. Develop a Statement of Work (SOW). Use the applicable SOW forms (from ES-EPD, AP-341-702, or equivalent). *Guidance: ASM website*. Address the following requirements:

Note: Throughout SOFT-ACQUIRE**, treat bold text (not headings)** as ML-1 and ML-2 requirements but guidance for ML-3. In this subsection, plain text is a requirement for ML-1 through ML-3.

   a. Identify the software requirements, **including but not limited to specific functionality in specified environments, acceptance criteria, and supplier/LANS interfaces**; for designed software, see SOFT-DESIGN.

   b. Require documentation (and/or computer programs) describing methods used in developing and validating the software (e.g., test cases). See SOFT-DESIGN and SOFT-V&V.

   c. **From NQA-1 qualified suppliers, and when available from Non-NQA-1 qualified suppliers, require the supplier to report software errors to the purchaser, and as appropriate, the ability for the purchaser to report software errors to the supplier**.

   d. Specify software features, including requirements for safety, security functions and performance as applicable.

   e. Specify requirements for supplier notification of new releases or other issues (in addition to errors) that impact the software operation.

[4] Per Alternate Method QPA-VAR-2016-005.1.

f. **As required, specify mechanisms for the users to obtain assistance in operating the software**.

g. Specify quality assurance requirements (*e.g., conform to DOE Order 414.1D or ASME NQA-1-2008/NQA-1A-2009 or* ASME NQA-1-2008/NQA-1A-2009, *Quality Assurance Requirements for Nuclear Facility Applications, Part I and Part II*). Specify quality assurance requirements on an Exhibit H for ML-1 through ML-3 (safety and risk significant) software, and for any ML software if the supplier is required to be onsite to install/maintain the software.

h. *Address the following guidance:*

1) For ML-1/ML-2 safety software that will undergo CGD, specify required supplier support to CGD the software (e.g., access for supplier assessments, surveys, 3rd party dedicators). Also see SOFT-V&V (3.0.N) for additional requirements when performing CGD.

2) Specify training and/or technical support required to promote successful installation and use of the software.

3) Consider specifying minimum acceptable performance history for similar applications.

4) Specify desired method/media for computer program delivery (e.g., electronic download, compact disc).

5) Consider attaching marked-up vendor product information that identifies key software requirements (e.g., 64 bit, Windows 10 operating system, Delta V version 10.2).

6) As needed, contact the Office of Chief Information Officer (OCIO) for assistance implementing PD210 and/or CIO-P100

5. As required, clarify LANL-supplier division of responsibility on the SWDS.

Example: LANL was planning to acquire software for an ML-2 application. The supplier provided notifications of defects, new releases or other issues that may impact the software operation on the supplier's website. The SWDS specified that prior to each use, the SRLM must review the supplier website and/or contact the supplier to ensure the notification requirement was fully satisfied.

6. Acquire in accordance with P840-1, PD210, and CIO-P100. Note the following:

a. For intra-laboratory acquisitions use interface agreements per P840-1. See P840-1 for an example interface agreement. Address SOW requirements in the agreement.

b. For inter-laboratory acquisitions from other organizations within the DOE complex, use integrated contract orders (ICOs) per P840-1. Address SOW requirements in the ICO.

c. Do not use P-cards for purchasing software associated with ML-1 through ML-3 applications (unless a formal exception is obtained per P840-1.

d. Protect against suspect/counterfeit items (S/CI) items by procuring from reputable suppliers and ensuring valid software licenses. See P330-9, *Suspect Counterfeit Items (S/CI)* for additional information.

e. Retain acquisitions documentation in accordance with the SRLM's governing records management process.

Ensure the correct SWID (obtained when completing Form 2033) is part of the acquisition documentation. See AP-341-402 for details.

f. See LANL Institutional Evaluated Supplier List (IESL) for a listing of NQA-1 qualified suppliers.

g. See AP-341-703, *Commercial Grade Dedication* for CGD. (See SOFT-V&V [*e.g., 3.0.N*] for use of performance history for dedicating computer programs, etc.) Identify and control acquired software during the dedication process.

B. When. Acquire when software is needed to perform work.

C. Who. The SO performs acquisition activities. The SRLM must review and approve software acquisitions.[5]

4.0 NON-SSC SOFTWARE ACQUISITION

Before acquiring new Non-SSC software, look for and evaluate whether existing LANL software may be used; if existing LANL software is suitable, then acquire it from the SRLM Subsection 4.1. If existing software is not suitable, then identify the desired new software and acquire it per one of the following subsections:

- Subsection 4.2: New ML-4 Non-SSC Software
- Subsection 4.3: New Software used in SSC Design that is "Simple and Easily Understood"
- Subsection 4.4: Other New Non-SSC Software

4.1. Existing Non-SSC Software (All MLs)

A. How.

1. Look for existing software at LANL that may be suitable. *Review the Engineering Services Division software inventory. Contact the ES-Div Software Coordinator, SRLM, or subject matter expert (SME) that may be using the desired software or similar software. The Coordinator should be listed on ES-Div and Chapter 21 webpages.*

2. Evaluate whether existing software is suitable for the intended application. Evaluate limitations (including licenses), boundary conditions, performance (including issues/risks), and bases. Incorporate the following in the evaluation:

a. Ensure the ML of the existing software is commensurate with the intended use. *For example, if CAESAR II software were acquired and managed as ML-3 software, it can only be used on ML-3 and ML-4 applications. It cannot be used on ML-1 or ML-2 applications unless: (a) it were determined to be "Simple and Easily Understood" software used in design, or (2) it were commercial grade dedicated.*

[5] See previous footnote regarding other approvals such as ASM.

b. Ensure the industry code associated with the software is appropriate for the intended application. *For example, CAESAR II, version 7.0 is based on the 2012 edition of American Society of Mechanical Engineers (ASME) B31.3; the software should be used only for designs where this edition of the code is appropriate.*

3. If the existing software is determined to be suitable, contact the SRLM to acquire the software and obtain authorization to use it per the SRLM's instructions. (Note that use may be limited by license restrictions in some cases.)

4. The SRLM reviews and authorizes his/her software use in the intended application by the additional software users. As applicable, the SRLM updates documentation (e.g., license agreements, etc.) to reflect additional users.

B. When. Before acquiring new software, look for, evaluate and acquire existing LANL software if suitable. Acquire when needed to perform work.

C. Who. The software user (SU) completes required acquisition actions. The SRLM of the existing software reviews and authorizes new users/applications

4.2. New ML-4 Non-SSC Software

See Subsection 3.1 and apply to Non-SSC ML-4 software.

4.3. New Software Used in SSC Design that is "Simple and Easily Understood"

A. How.

1. Determine if the software is in the category of software that is used in SSC design that is "Simple and Easily Understood". The Responsible Engineer (RE) completes the determination using the following criteria. Each criterion must be satisfied.

   a. The software is used in the design of SSCs;

   b. The results of the computer program can be easily confirmed through hand calculations;

   c. A person technically qualified, as determined by the SRLM, in the subject can review and understand the program and the supporting calculations; and,

   d. The software can be individually verified with each use (e.g., calculation). See SOFT-V&V for individual verification requirements.

   If the software is in this category, then acquire the software as ML-4 software in accordance with Subsection 4.2. Such software is excluded from commercial grade dedication (CGD) or the requirement to acquire from an NQA-1 supplier.

2. If the software is not in this category, then acquire in accordance with Subsection 4.4.

Note: If the software is safety software, frequent use of the software may justify CGD or acquisition from an NQA-1 supplier.

B. When. Acquire when software is needed to perform work.

C. Who. The SO develops and the SRLM reviews and approves the acquisition (see footnote for Paragraph 3.1.c regarding ASM).

4.4. Other New Non-SSC Software

Software for ML-1, ML-2, and/or ML-3 applications that is not "Simple and Easily Understood" software used for SSC design must be acquired in accordance with this subsection. This includes toolbox codes.

A. How.

1. Using the requirements of this subsection determine whether to acquire software from:
   - the DOE Safety Software Quality Assurance Central Registry (hereafter Registry);
   - an NQA-1 qualified supplier; or
   - a Non-NQA-1 qualified supplier.

2. Toolbox codes: Determine if the software is listed on the Registry. If listed, it is a toolbox code. [6] Acquire in accordance with the following:
   a. Only the version of the toolbox code listed on the Registry may be acquired without performing CGD or acquiring from an NQA-1 supplier.
   b. *If the software is from other DOE site, acquire using an integrated contract order (ICO) in accordance with P840-1 and this subsection.*
   c. *If the software is from another government agency, acquire using an inter-agency agreement (IAA) in accordance with P840-1 and this subsection.*

3. Excluding toolbox codes, acquire ML-1 and ML-2 safety software from an NQA-1 qualified supplier, or acquire from a Non-NQA-1 qualified supplier and commercial grade dedicate (CGD) the software. See AP-341-703, *Commercial Grade Dedication* for CGD.
   a. Make the acquisition strategy decision above considering the following factors:
      i. the availability of NQA-1 qualified suppliers (see the LANL Institutional Evaluated Supplier List, (IESL)) and/or the feasibility of adding NQA-1 qualified suppliers to the IESL,
      ii. performance history;
      iii. cost, schedule and other commercial factors applicable to the software lifecycle.

4. Develop a SWDS. Use SOFT-GEN-FM01, *Software Data Sheet Form (SWDS)*. Document the NQA-1/Non-NQA-1 acquisition strategy on the SWDS. As required by the SRLM, document supporting rationale on the SWDS or elsewhere.

5. ML-3/Risk Significant Software: May acquire from a Non-NQA-1 qualified supplier; CGD does not apply. G*uidance: Current P840-1 requires receipt*

[6] P840-1, Revision 8 doesn't address the central registry.

inspection; however, use of Exhibit H part number verification (PNV) version is generally adequate because V&V at installation ensures quality.

6. Develop a Statement of Work (SOW). Use the applicable SOW forms (from ES-EPD, AP-341-702, or equivalent). *Guidance: ASM website*. Address the following requirements:

Note: Throughout SOFT-ACQUIRE**, treat bold text (not headings)** as ML-1 and ML-2 requirements but guidance for ML-3. In this subsection, plain text is a requirement for ML-1 through ML-3.

a. Identify the software requirements. Include as specific functionality in specified environments (as applicable), acceptance criteria, **and supplier/LANL interfaces**.

b. Require documentation (and/or computer programs) describing methods used in **developing and** validating the software (e.g., test cases).

c. **From NQA-1 qualified suppliers, and when available from Non-NQA-1 qualified suppliers, require the supplier to report software errors to the purchaser, and as appropriate, the ability for the purchaser to report software errors to the supplier**.

d. Specify software features, including requirements for safety, security functions and performance as applicable.

e. Specify requirements for supplier notification of new releases or other issues (in addition to errors) that impact the software operation.

f. *As required, specify mechanisms for the users to obtain assistance in operating the software.*

g. Specify quality assurance requirements (*e.g., conform to DOE Order 414.1D or ASME NQA-1-2008/NQA-1A-2009, Quality Assurance Requirements for Nuclear Facility Applications, Part I and Part II*). Specify quality assurance requirements on an Exhibit H for ML-1 through ML-3 (safety and risk significant) software and for any ML software if the supplier is required to be onsite to install/maintain the software.

h. For software that includes models (see SOFT-GEN for definition and SOFT-DESIGN for additional information), ensure the appropriateness of the model for the application in addition to the appropriateness of the computer code used to provide the outcome of the model.

i. *Address the following guidance:*

1) *Specify minimum design codes for the software. As a fictitious example, SAP2000 must satisfy ASCE 10-97 for steel frame design and ACI 318-08 for concrete frame design.*

2) *For safety software that will undergo CGD, specify supplier support that may be required to dedicate the software (e.g., access for supplier assessments, surveys, third-party*

dedicators). Also see SOFT-V&V (3.0.N) for additional requirements when performing CGD.

3) *Specify training and/or technical support required to promote successful installation and use of the software.*

4) *Specify whether single user, multiuser server-based, or cloud-based licenses are required (contact the ES-Div Software Coordinator for support).*

5) *Consider specify minimum acceptable performance history for similar applications.*

6) *Specify desired method/media of computer program delivery (e.g., electronic download, compact disc).*

7) *Consider attaching marked-up vendor product information that identifies the key software requirements (e.g., 64 bit, Windows 10, ASME B31.3 2012 Edition).*

8) *As needed, contact the Office of Chief Information Officer (OCIO) for assistance implementing PD210 and/or CIO-P100.*

7. As required, clarify LANL-supplier division of responsibility on the SWDS.

   Example: LANL was planning to acquire software in an ML-2 application. The supplier provided notifications of defects, new releases or other issues that may impact the software operation on the supplier's website, rather than a more direct notification such as email. The SWDS specified that the SRLM will review the supplier website and/or contact the supplier prior to each use for the information and thereby ensure the requirement was satisfied.

8. Acquire in accordance with P840-1, PD210, and CIO-P100. Note the following:

   a. See LANL Institutional Evaluated Supplier List (IESL) for a listing of NQA-1 qualified suppliers

   b. See AP-341-703, *Commercial Grade Dedication* for CGD. (See SOFT-V&V (*e.g., 3.0.N*) for use of performance history for dedicating computer programs and other CGD requirements.)

   c. *Guidance: Intra-laboratory acquisitions should use interface agreements per P840-1. See P840-1 for an example interface agreement. Address SOW requirements in the agreement.*

   d. For inter-laboratory acquisitions from other organizations within the DOE complex, use integrated contract orders (ICOs) per P840-1. Address SOW requirements in the ICO.

   e. Do not use P-cards for purchasing software associated with ML-1 through ML-3 applications (unless a formal exception is obtained per P840-1.

   f. Protect against suspect/counterfeit items (S/CI) items by procuring from reputable authorized suppliers and ensuring valid software licenses. See P330-9, *Suspect Counterfeit Items (S/CI)* for additional information.

g. Retain acquisitions documentation in accordance with the SRLM's governing records management process.

Ensure the correct SWID (obtained when completing Form 2033) is part of the acquisition documentation record number. See AP-341-402 for details.

B. When. Acquire when software is needed to perform work. Complete acquisition prior to approval for use as described in SOFT-V&V.

C. Who. The SO performs acquisition activities. The SRLM must review and approve software acquisitions (See previous footnote regarding other approvals such as ASM.).

TABLE OF CONTENTS

1.0 PURPOSE, SCOPE, AND APPLICABILITY 1
2.0 KEY DEFINITIONS AND ACRONYMS 6
3.0 SSC SOFTWARE DESIGN REQUIREMENTS (INPUT) 6
4.0 SSC SOFTWARE DESIGN (OUTPUT) 8
4.1 General Software Design Requirements 8
4.2 Software System Architectural Design 10
4.3 Software Requirements Specification (SWRS) 11
4.4 Software Architectural Design 11
4.5 Software Detail Design 11
4.6 Software System Hazard Analysis and Mitigation 12
4.7 Software Design Traceability and Testability 13
4.8 Operations and Maintenance (O&M) Instructions 13
4.9 SOFT-GEN Requirements and Deliverables 13
5.0 SSC SOFTWARE COMPUTER PROGRAM LANGUAGE (CODE) 13
6.0 NON-SSC SOFTWARE DESIGN REQUIREMENTS (INPUT) 14
7.0 NON-SSC SOFTWARE DESIGN (OUTPUT) 15
8.0 NON-SSC SOFTWARE PROGRAM LANGUAGE (CODE) 16

REFERENCE: SOFTWARE DESIGN DOCUMENTATION EXAMPLE

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Added design output clarifications; other minor editorial clarifications.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

1.0 PURPOSE, SCOPE, AND APPLICABILITY

This section describes the “how, when, and who” for designing (developing) software. See Tables 21.3-1 and 21.3-2 for a summary of this section for SSC software and Non-SSC software respectively.

See Chapter 21 applicability in SOFT-GEN. In addition:

- This section does not apply to “simple and easily understood” software used in the design of SSCs if individually verified as described in SOFT-V&V.
- See SOFT-V&V for required verification and validation activities and deliverables for designed software.
- See SOFT-GEN for additional requirements that apply during design.

- Also see SOFT-ACQUIRE for acquisition requirements associated with software that is designed for LANL (this section and SOFT-ACQUIRE may apply.)
- Users must tailor SOFT-DESIGN or do not use for some SSC software (e.g., that's embedded) or when acquiring as-is (e.g., off-the-shelf)

Table 21.3-1 SOFT-DESIGN Section Summary for SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		1, 2	3, 4	How	When	Who[2, 3]	ESM Ch. 21 Ref. Section[4]
1	Develop software design requirements (input)	R	Gr	▪ Develop software design input deliverables: o SSC drawings (e.g., PFDs, P&IDs, SOOs, hardware/instrument location drawing, location drawing, instrument/device list). o SSC performance specification o FDD or SDD o Other SSC design inputs/outputs as required	▪ Earliest practical time ▪ 60%, 90%, 100% SSC detail design ▪ Prior to software design as much as practical	▪ SSC DA (D, R, A) ▪ LANL SRLM (R, A) ▪ FDAR (R, A)	SOFT-DESIGN, 3
2	Design the software (output)	R	Gr	▪ Design the software (output) deliverables: o software specification (SWRS) o software design (SWDD) o software system hazard analysis and mitigation documentation (SWHA) o software design traceability documentation (SWTM) o operations and maintenance (O&M) instructions o SOFT-GEN deliverables o Other software design deliverables as required	▪ Earliest practical time ▪ If in scope of SSC design, software requirements (draft) at 30% design submittals, firmed up at 60%. ▪ If not in scope of SSC design, then in detail design submittals ▪ Prior to software coding as much as practical	▪ SD (D, R, A) ▪ SRLM (R, A) ▪ LANL SRLM (R, A)	SOFT-DESIGN, 4
3	Translate the software design into computer program language	R	Gr	▪ Develop computer program language (code): o data structures/files, o source code (where feasible), o executable code ▪ Provide computer program listings (e.g., pdf printout of source code)	▪ Earliest practical time ▪ Prior to final acceptance testing	▪ SD (D, R, A) ▪ SRLM (R, A)	SOFT-DESIGN, 5

Table 21.3-1 SOFT-DESIGN Section Summary for SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		1, 2	3, 4	How	When	Who[2, 3]	ESM Ch. 21 Ref. Section[4]

[1] ML = Associated Management Level per Form 2033. R = Required. Gr = Required but graded.

[2] D = Develop or implement; R = Review; A = Approve.

[3] SSC DA = Design agency for SSC that is associated with the software); SRLM = Software Responsible Line Manager (prior to turnover to LANL). LANL SRLM = SRLM after turnover to LANL. FDAR = Facility Design Authority Representative.

[4] Ref. = ESM Chapter 21 section name and subsection number (e.g., section SOFT-GEN, subsection 3, Software Identification and Determination).

Table 21.3-2 SOFT-DESIGN Section Summary for Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		-1, -2	-3, -4	How	When	Who[2, 3]	ESM Ch. 21 Ref.[5]
1	Develop software design requirements (input)	R	Gr	▪ Develop system requirements (input) specification (SWYRS)	▪ Earliest practical time ▪ Prior to software design as much as practical	▪ SO (D, R, A) ▪ SRLM (R, A) ▪ FDAR (R, A)	SOFT-DESIGN, 6
2	Design the software (output)	R	Gr	▪ Design the software (output) deliverables: o software specification (SWRS) o software design (SWDD) o software system hazard analysis and mitigation documentation (SWHA) o software design traceability documentation (SWTM) o operations and maintenance (O&M) instructions o SOFT-GEN deliverables o Other software design deliverables as required o Computer model mathematical terms[4]	▪ Earliest practical time ▪ Prior to software coding as much as practical	▪ SD (D, R, A) ▪ SRLM (R, A)	SOFT-DESIGN, 7

Table 21.3-2 SOFT-DESIGN Section Summary for Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]		Implementation Detail			Reference
		-1, -2	-3, -4	How	When	Who[2, 3]	ESM Ch. 21 Ref.[5]
3	Translate the software design into computer program language (code)	R	Gr	▪ Develop computer program language (code): o data structures/files, o source code (where feasible), o executable code ▪ Provide computer program listings (e.g., pdf printout of source code)	▪ Earliest practical time ▪ Prior to final acceptance testing	▪ SD (D, R, A) ▪ SRLM (R, A)	SOFT-DESIGN, 8

[1] ML = Associated Management Level per Form 2033. R = Required. Gr = Required but graded.
[2] D = Develop or implement; R = Review; A = Approve.
[3] SO = Software Owner. SRLM = Software Responsible Line Manager. FDAR = Facility Design Authority Representative.
[4] Applicable to computer program models only.
[5] Ref. = ESM Chapter 21 section name and subsection number (e.g., SOFT-GEN, Subsection 3, Software Identification and Determination).

2.0 KEY DEFINITIONS AND ACRONYMS

See Chapter 21, SOFT-GEN for key definitions and acronyms.

3.0 SSC SOFTWARE DESIGN REQUIREMENTS (INPUT)

Define, control and verify the design. This section describes the minimum requirements, activities and deliverables for preparing SSC software design requirements (i.e., inputs). SSC software design requirements are the information provided to the SSC software designer to design the SSC software.

A. How.

1. SSC software design requirement (input) deliverables depend on the nature, complexity and associated ML of the software. Use Table 21.3-3, *SSC Software Design Requirement (Input) Deliverables* and direction from the LANL SRLM as a guide to define software design requirement deliverables.

2. Process and retain input documentation in accordance with the SRLM's governing document control and records management process. Ensure the correct SWID (obtained when completing Form 2033) is part of the documentation record number. See AP-341-402, *Engineering Document Management in Operating Facilities* for details.

Table 21.3-3 SSC Software Design Requirement (Input) Deliverables

No	Deliverable	Apply by ML[1]		
		1, 2	3	4
01	SSC System Drawings (e.g., process flow diagrams (PFDs) process and instrumentation diagrams (P&IDs), Sequence of Operations (SOOs), hardware/instrument location drawing, network drawing[2], instrument/device list.)	R	Gr	Gr
02	SSC performance specification	R	Gr	Gr
03	SSC Facility Design Description (FDD) and/or System Design Description (SDD)	R	Gr	-
04	Other SSC Design inputs/outputs as required to support software design[3]	R	Gr	-

[1] R = As required by ESM Chapter 1, General, Section Z10 or other ESM chapters; Gr = Required but graded per Section Z10 or other ESM chapters. "-" = Not required.

[2] Level of detail as required by FDAR; e.g., a notional or "block" diagram of the network as a minimum, detailed technical drawing later.

[3] As determined by LANL SRLM; may include SSC hazard analyses, Requirements Criteria Document (RCD), identification of commercial grade dedication (CGD) critical characteristics etc.

3. SSC software design requirements (inputs) must:

   a. Be identified and documented using a systems engineering process (Ref. P341, *Facility Engineering Processes Manual* and ESM Chapter 20 (pending publication)).

   b. Be documented as part of the SSC design documents (e.g., Process and Instrumentation drawings (P&IDs) with Sequence of Operations (SOOs), system specifications, system design descriptions (SDDs)).

   c. Be based on upper tier performance and functional requirements.

   d. Identify the operating system, function, interfaces, performance requirements, installation considerations, applicable SSC design

inputs, critical characteristics (as applicable), and design constraints of the computer program.

e. Specify technical and software engineering requirements, including security features.

f. Identify applicable reference drawings, specifications, codes, standards, regulations, procedures, or instructions that establish test, inspection, and acceptance criteria.

g. Be commensurate with the risk of unauthorized use; address security requirements (e.g., vulnerability/cyber-security protections).

h. For ML-1 through ML-3 software, be traceable throughout the software life cycle.

i. For ML-1 through ML-3 software, be based on system/component hazard analyses that identify system/component risks and the means for controlling them.

j. The software design shall consider the computer program's Operating Environment (see Definitions in SOFT-GEN).[1]

k. Identify and address the user human machine interface (HMI) requirements. Factor in existing user operational protocols, conventions and methods by users/operators. As appropriate, specify prototypes/HMI screenshots for review and comment as part of the design deliverables.

l. Ensure software design requirements, as applicable, are consistent with SSC technical baseline documents. For information on SSC software technical baseline documents, see STD-342-100, *Engineering Standards Manual, Chapter 1, Section Z10, General*; AP-341-616, *Technical Baseline Change During Design;* and AP-341-405, *Identification and Control of Technical Baseline, Variances, Alternate Methods, and Clarifications in Operating Facilities*.

m. Ensure the design inputs and sources are identified and documented, specified on a timely basis, translated into design documents, and their selection review and approved.

n. Include user-level input/review of screen shots, prototypes, etc.

4. For guidance, see:

- IEEE Std 1233, *IEEE Guide for Developing System Requirements Specifications*, and
- ISO/IEC/IEEE 29148, *Systems and Software Engineering-Life Cycle Processes-Requirements Engineering*.

B. When. Develop software design requirements at the earliest practical time and as much as practical, prior to software design. As applicable, submit software requirements (draft) at 30% design submittals, firmed up at 60%.

C. Who. For SSC software, the SSC Design Agent (or Design Agency [DA]) that develops the SSC detailed design also develops, reviews, and approves the SSC

[1] An integral part of software design is the design of a computer program that is part of an overall system. ASME NQA-1a-2009, Part II, Subpart 2.7-402

software design requirements. The LANL SRLM and FDAR review and approve the software requirements.

4.0 SSC SOFTWARE DESIGN (OUTPUT)

A. How.

B. When. Design software at the earliest practical time and as much as practical, prior to software coding. Unless the software design is deferred to subcontractors, submit software design deliverables with the 90% and 100% detail designs. If software design is deferred, then specify and submit software design deliverables as part of the required submittal process.

C. Who. The SD develops, reviews and approves the deliverables of this section including the architectural design, software specification, detail design, hazard analysis and mitigation documentation, and design traceability documentation; the SD, SRLM (prior to turnover to LANL), and LANL SRLM review and approve.

4.1 General Software Design Requirements

A. Software design is an expanding and complex area. Accordingly, this section is limited to providing only a high-level description of the software design output requirements. For additional detail, see:

- DOE-STD-1195, *Design of Safety Significant Safety Instrumented Systems Used at a DOE Nonreactor Nuclear Facilities*
- ANSI/ISA 84.00-01, *Functional Safety: Safety Instrumented Systems for the Process Industry Sector*;
- ANSI/IEEE std. 7-4.3.2, *IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations,*
- IEEE Std 1016, *IEEE Standard for Information Technology – System Design – Software Design Descriptions*[2]
- IEEE Std 1016.1, *IEEE Guide to Software Design Descriptions*
- ESM Chapter 8: *Instrumentation and Controls*

B. Design software to meet the software requirements and deliverables of this section, SOFT-DESIGN, and other ESM chapters as applicable. Design deliverables may be combined as needed in one or more documents.

C. Use an accepted design methodology (see IEEE Std 1016.1).

D. As specified by the LANL SRLM for ML-1 and ML-2 software, require and provide software designer and computer programmers (coders) qualifications commensurate with the risk associated with the software. *Refer to P781-1, Conduct of Training for processes to determine qualifications.*

E. Document the software design and the computational sequence necessary to meet the software requirements including as applicable: numerical methods, mathematical models, physical models, control flow, control logic, data flow, process flow, data structures, supporting calculations (Ref. AP-341-605, *Calculations*), software baseline, process structures, and the applicable relationships between data structures and process structures. Combine with the documentation of the software

[2] Flow diagrams, charts and/or pseudo-code should be provided.

design requirements, or the computer program listings (e.g., pdf printout of source code, see definitions) resulting from implementation of the design.

F. Ensure software design outputs, as applicable, are consistent with SSC technical baseline documents.

G. Software design (output) deliverables depend on the nature, complexity and associated ML of the software. See SOFT-DESIGN References for examples. Deliverables consist of drawings and/or other descriptive documents. They may be combined in one or more documents or provided separately. Software design deliverables may be divided into the following categories: (a) software design documentation (system architecture, software architecture, software detail design collectively referred to as SWDD), (b) Software Requirements Specification (SWRS), (c) hazard analysis and mitigation documentation (SWHA), (d) software design traceability documentation (SWTM), (e) O&M documentation, (f) SOFT-GEN deliverables, and (g) other software deliverables.

Note: Verification and Validation (V&V) documentation, including test plans, interim test reports, and design review documentation, are typically produced concurrent with the design. See SOFT-V&V.

H. Provide associated software deliverables as required by other ESM chapters (e.g., Chapter 8 for instrumentation and control systems).

I. Provide design deliverables per Table 21.3-4, *Software Design (Output) Deliverables*.

J. Ensure the design is defined, controlled and verified; the design interfaces are identified and controlled; design adequacy is verified by individuals other than those who designed the software. *Guidance: The eventual system engineer and/or users should work with the design/development team to ensure documentation etc. is done to an appropriate level commensurate with the ML of the software/SSC.*

K. Ensure design documentation and records include not only final design documents, such as drawings and specifications, and revision to those documents, but also documentation that identifies the important steps in the design process, including sources of design inputs.

L. As applicable, address use and maintenance requirements in the design (e.g., access controls and/or computer system vulnerability protections) per SOFT-MAINT.

Table 21.3-4 Software Design (Output) Deliverables				
No	Design Deliverable Element	Apply by ML[1]		
		1, 2	3	4
01	Software system architecture design[2]	R	Gr	Gr
02	Software requirements specification (SWRS)[2]	R	Gr	-
03	Software architecture design[2]	R	Gr	Gr
04	Software detail design[2]	R	Gr	-
05	Software system hazard analysis and mitigation docs. (SWHA)[2]	R	Gr	-
06	Software design traceability documentation (e.g., SWTM)[2]	R	Gr	-
07	Computer Program Listings, including the program (design product)[2]	R	Gr	-
08	Operations and maintenance (O&M) instructions	R	R	R
09	SOFT-GEN deliverables[3]	See SOFT-GEN		
10	Other software design deliverables[4, 5]	See LANL SRLM and other ESM chapters.		

[1] R = Required; Gr = Required but graded; "-" = Not required.
[2] May be combined in one or more documents or provided separately.
[3] Required and graded based on the nature, complexity and associated ML per direction from the LANL SRLM.
[4] Includes software list, Form 2033, software data sheet, software baseline.
[5] As determined by LANL SRLM and/or as required in other ESM chapters (e.g., Human Machine Interface (HMI) graphical layouts/slides). See SOFT-V&V for associated V&V deliverables.
[6] See ESM Chapter 8 for software related deliverables (e.g., logic diagrams, loop diagrams, point list tables, etc.) specific to instrument and control (I&C) system designs.

M. Retain documentation in accordance with the governing records management process. For ES-Div Non-SSC software and SSC software, ensure the correct SWID (obtained when completing Form 2033) is part of the documentation record number. See AP-341-402 for details.

4.2 Software System Architectural Design

A. Develop the software system architectural design. Allocate system-level requirements to hardware, software, and interfaces of the system.

B. For ML-1, ML-2, and some ML-3 systems where software operational complexity and lifecycle maintenance can be disadvantageous, evaluate and as appropriate, employ designs that use no or minimal software.

C. Sufficiently detail the system architectural design such that a person technically qualified in the subject can verify and validate (V&V) that the design satisfies requirements without recourse to the Software Designer (SD).

D. Consider the following concepts, as applicable, in the software system architectural design.

1. **Isolation**: Critical elements should be separated from each other to preclude undefined and/or unintended interactions. For ML-1 through ML-3 software, safety systems should be separated from non-safety systems where possible. Barriers (e.g., separation of safety from non-safety modules) should be used to prevent non-safety functions from interfering with safety functions.

Consider use of different platforms, power supplies, inputs/outputs to physically separate the safety from non-safety elements.

2. **Independence/Diversity**: For ML-1 and ML-2 software, independent, diverse systems should be employed. These are systems where the stimuli originate from and are handled by separate elements with different designs, independent hardware inputs and/or independent software modules.

   Note: Because different manufacturers may use the same processor or software, common mode failures may not be averted simply by acquiring from several manufacturers.

3. **Fail-Safe Design**: Design such that the system remains in a safe mode upon failure without compromising isolation features or other safety functions.

4. **Incompatibility/Longevity**: Design to ensure integrated hardware-software compatibility as much as possible throughout the life-cycle of the system.

4.3 Software Requirements Specification (SWRS)

A. For software system requirements that are allocated to software, further refine the requirements and develop software-level requirements. Document the software requirements. Software requirements are documented in a software requirements specification (SWRS). See IEEE Std 830, *IEEE Recommended Practice for Software Specifications* for guidance.

4.4 Software Architectural Design

A. Transform the software requirements into the software architectural design. The software architecture design specifies the structures of the software and the various software components, interactions and interfaces between the software components.

B. Include the high-level design for the external interfaces between the software components and other components within the system (hardware and human interface) and between the software and entities outside the system.

C. Sufficiently detail the software architectural design such that a person technically qualified in the subject can V&V the design without recourse to the SD.

D. Where appropriate and/or as required by the software requirements document, design software control functions that are performed incrementally rather than in a single step to reduce the potential that a single failure of a software element can cause an unsafe system state.

E. Where appropriate and/or as required by the software requirements document, design built-in fault detection and self-diagnostics that detect and report software faults and failures in a timely manner and allow actions to be taken to avoid an impact on the system's safe operation.

4.5 Software Detail Design

A. For ML-1 through ML-3 software, define the internals of each software component down to individual software modules that can be coded.

B. Specify/document the software coding standards and/or conventions.

C. As applicable, include data structures/databases and external interfaces.

D. Sufficiently detail the detailed software design to: (a) allow a person technically qualified in the subject to V&V the design without recourse to the SD; and (b) allow computer programmers to code the design without undue difficulty.

4.6 Software System Hazard Analysis and Mitigation

A. For ML-1 through ML-3 software, conduct and document a **software** hazard analysis at the system and component level to identify software risks and develop mitigating approaches for controlling them. Potential failures should be identified and evaluated for their consequences of failure and probability of occurrence. Some potential problems may include (1) complex or faulty algorithm, (2) lack of proper handling of incorrect data or error conditions, (3) buffer overflow, and (4) incorrect sequence of operations due to either logic or timing faults.

B. Ensure the software hazard analysis is consistent with the system safety documentation for the associated facility. See SBP111-1, *Facility Hazard Categorization and Documentation* for associated facility safety documentation.

C. For ML-1 and ML-2 software, perform and document the hazard analysis based on recognized consensus standards. See ESM Chapter 8, *Instrument & Controls* for application of hazard analysis/mitigation with respect to the following standards for SSC software:

- ANSI/Instrumentation, Systems, and Automation Society (ISA) S84.00.01, *Functional Safety: Safety Instrumented Systems for the Process industry Sector*; and,
- ANSI/IEEE Std 7-4.3.2, IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations).

D. *For ML-3 software, standard-based methods or less formal methods should be used using a graded approach (e.g., failure modes and effects analysis, fault-tree modeling, event-tree modeling, cause-consequence diagrams, hazard and operability analysis, and interface analysis).*

E. For ML-4 software, a hazard analysis is not required but may be performed.

F. *Multiple/common-cause failures should be evaluated in the hazard analysis.*

Note: Failure mode and effect analysis (FMEA) approaches, when used alone, do not address multiple failures/common-cause failures. (ANSI/ISA 84.00-01-2004-Part 1 and IEEE STD-7-4.3.2-2003 provide guidance.)

G. The hazard analysis and design must include analysis and possible problems with the computer program's operating environment (including security environment) and external and internal abnormal conditions and events that can affect the computer program.

H. In the software design documentation (SWDD), as part of the SWHA, or in a separate deliverable, provide documentation that shows how the consequences of hazards/problems are mitigated. *Mitigation strategies should be included for:*

1. *Software standard hazards (i.e., the basic hazards associated with the software or process);*
2. *Software system failures (i.e., the functionality written into the software itself to protect from failure); and*

3. *Software system overrides (i.e., the functionality written into the software to keep a user or other system from bypassing the safety features within the software item.)*

4.7 Software Design Traceability and Testability

A. For ML-1 and ML-2 software, document traceability from the software design requirements to the software design deliverables. For ML-3 software, a software traceability matrix (SWTM) is recommended to document traceability. Other methods however, such as tracing the requirement within the design deliverable are also acceptable. Requirements tools and/or databases may be used.

B. For large, complex ML-1 and ML-2 software, consider bi-directional traceability (tracing from the requirements to the design as well as from the design to the requirements) to identify functionality that may have been inadvertently added without requirement drivers.

C. Design the software and the associated SSC such that the requirements can be readily tested.

4.8 Operations and Maintenance (O&M) Instructions

A. Develop and provide operations and maintenance (O&M) instructions to use and maintain the computer program(s). See SOFT-MAINT for use and maintenance requirements and address in the O&M instructions and/or the SWDS.

B. Sufficiently detail the O&M instructions to allow an individual trained in the use of such software to follow the instructions and use the program without undue difficulty and/or probability of misuse.

C. *For ML-1 and ML-2 software, O&M instructions should be developed using standard conventions for instructions (e.g., ANSI/IEEE Std 26514, Systems and Software Engineering – Requirements for Designers and Developers of User Documentation).*

4.9 SOFT-GEN Requirements and Deliverables

Satisfy software requirements and provide deliverables as described in SOFT-GEN.

5.0 SSC SOFTWARE COMPUTER PROGRAM LANGUAGE (CODE)

A. How.

1. Translate the software design into computer program language (code).
2. For ML-1 through ML-3 software, use standards and/or conventions as approved in the software design (e.g., IEEE Std 1666, *IEEE Standard for Standard System C Language Reference*).
3. Develop source code suitable for compilation or translation.
4. Provide both source code and/or executable code as deliverables. If source code cannot be provided (e.g., for proprietary reasons), then provide objective evidence that the source code was V&V'd as per IEEE Std. 1012 and/or acceptable methods.
5. For ML-1 and ML-2 software, provide computer program listings (see definitions).
6. Code should be developed to sufficient detail and clarity to allow someone technically qualified in the computer programming language to review,

understand, and verify it meets the software design requirements without recourse to the SD and/or coder.

7. For ML-1 and ML-2 code, ensure most code lines have explanatory notes/comments (pseudo code) to support future troubleshooting, bug fixes, and/or system modifications.

8. Employ configuration management (including computer program code labeling) and problem reporting/corrective management in accordance with SOFT-GEN.

B. When. Develop code at the earliest practical time and prior to final program acceptance testing.

C. Who. The SD develops, reviews and approves the software code; the SRLM reviews and approves the code.

6.0 NON-SSC SOFTWARE DESIGN REQUIREMENTS (INPUT)

Guidance: Unlike SSC software, Non-SSC software typically does not use upper level SSC design documents (e.g., P&IDs, SOOs, SDDs) for design input. Therefore, the design input is communicated through other documentation. A ***system*** *requirements specification (SWYRS) is used to convey upper level design inputs analogous to the upper-level SSC design documents. A software requirements specification (SWRS) is developed for software system requirements that are allocated to software. The SWRS further refines the requirements and develops software-level requirements.* For designs where software is relatively low in complexity, where the software requirements constitute most of the system requirements, and when allowed by the LANL SRLM, the SWYRS and SWRS may be combined.

A. How.

1. Using a graded approach for ML1 through ML-4 associated software, develop a system requirements specification (SWYRS). The SWYRS is a document that communicates the requirements of the customer to the technical community who will further specify and build the software.

   For ML-4 software, the calculation associated with the software may be used to convey the software requirements in lieu of the SWYRS and/or SWRS.

   For ML-1 through ML-3 software, the requirements may be directly included in the SWDS.

   See the following for SWYRS guidance:

   - ISO/IEC/IEEE 29148, *Systems and Software Engineering-Life Cycle Processes-Requirements Engineering*.

2. The SWYRS must:

   a. Be identified and documented using a systems engineering process (Ref. P341, *Facility Engineering Processes Manual* and ESM Chapter 20 [pending publication]).

   b. Identify the operating system and other aspects of Operating Environment, function, interfaces, performance requirements,

installation considerations, applicable design inputs, and design constraints of the computer program.

c. Specify technical and software engineering requirements, including security features.

d. Identify applicable reference drawings, specifications, codes, standards, regulations, procedures, or instructions that establish test, inspection, and acceptance criteria.

e. Be commensurate with the risk of unauthorized use; address security requirements (e.g., vulnerability/cyber-security protections).

f. For ML-1 through ML-3 software, be traceable throughout the software life cycle.

g. For ML-1 through ML-3 software, be based on system/component hazard analysis that identifies system/component risks and the means for controlling them.

B. When. Develop the SWYRS at the earliest practical time and where practical, prior to software design.

C. Who. The SO develops the SWYRS and the SO (e.g., with assistance of SD); SRLM and FDAR review and approve the SWYRS.

7.0 NON-SSC SOFTWARE DESIGN (OUTPUT)

A. How.

1. For ML-4 software, the calculation associated with the software may be used to convey the software design output deliverables.

   For ML-1 through ML-3 software, the design output may be included in the SWDS.

2. See SSC Software design Subsection 4 and note that SSC-related text and/or references do not apply. For example, the following are not applicable:

   a. Integration of design outputs with SSC technical baseline documents (*4.1.F)*

   b. Provide associated software deliverables or use associated software standards as required for SSCs as stated by other ESM chapters (*4.1.B and 4.6.C*).

3. *Non-SSC software can include computer models. There are various definitions for models. From* EPA/100/K-09/003, *models are simplifications of the real world constructed to gain insights into select attributes of a particular*

physical, biological, economic, engineered, or social system.[3] *From LA-14167-MS, The Mathematical Model comprises the conceptual model, mathematical equations, and modeling data needed to describe the Reality of Interest. The Mathematical Model will usually take the form of the partial differential equations (PDEs), constitutive equations, geometry, initial conditions, and boundary conditions needed to describe mathematically the relevant physics. The Reality of Interest represents the physical system for which data is being obtained.*[4] For computer model design, also include the following:

a. A description of the conceptual model for the problem to be solved. Include assumptions, algorithms, relationships and data.

b. The mathematical terms that describe the conceptual model. Include mathematical equations, boundary values, initial conditions, and modeling data as required. Include numerical solution techniques such as finite element and finite difference when equations describing the mathematical model cannot be solved analytically.

c. *Guidance: Use ANSI/ANS-10.7, Non-Real-Time, High-Integrity Software for the Nuclear Industry – Development Requirements as a guide*.

B. When. Design software at the earliest practical time and as much as practical, prior to software coding.

C. Who. The SD develops, reviews, and approves the software design; the SO and SRLM review and approve the design.

8.0 NON-SSC SOFTWARE PROGRAM LANGUAGE (CODE)

See Subsection 5.

REFERENCE: SOFTWARE DESIGN DOCUMENTATION EXAMPLES

Examples provided are for informational purposes only and have not been reviewed for compliance to the requirements of this chapter. Better examples will be posted as they become available.

At time of writing, examples included:
Software Requirements Specification (SRS) Example (Source Tracker)
Design Document Example for Designed Software (Source Tracker)
User Manual (O&M instructions) for Designed Software (Source Tracker)

[3] EPA/100/K-09/003, *Guidance on the Development, Evaluation, and Application of Environmental Models.*
[4] LA-14167-MS, Los Alamos National Laboratory, *Concepts of Model Verification and Validation*, Thacker et al., 2004.

TABLE OF CONTENTS

1.0 PURPOSE AND APPLICABILITY....2
2.0 KEY DEFINITIONS AND ACROYNMS....8
3.0 V&V AND SOFTWARE APPROVAL FOR USE (SWAU) - GENERAL....8
4.0 DESIGNED SSC SOFTWARE V&V and SWAU....10
4.1 Design Requirement (Input) Review....11
4.2 Test Plan Development and Review....11
4.3 Software Design Review....14
4.4 Computer Program Code (Code) Review....15
4.5 Testing/Cx, Review, and Report....15
4.6 Review and Approval for Use....17
5.0 READ-ONLY SSC SOFTWARE V&V AND SWAU....18
5.1 Requirement (Input) Review....18
5.2 Test Plan Development and Review....18
5.3 Design Review....19
5.4 Testing/Cx and Review....19
5.5 Review and Approval for Use....19
6.0 DESIGNED NON-SSC SOFTWARE V&V AND SWAU....19
6.1 Model V&V....19
6.2 Design Requirement (Input) Review....20
6.3 Test Plan Development and Review....20
6.4 Software Design Review....20
6.5 Designed Non-SSC Computer Program Code Review....20
6.6 Testing and Review....20
6.7 Review and Approval for Use....20
7.0 ACQUIRED NON-SSC SOFTWARE V&V AND SWAU....20
7.1 Model V&V....21
7.2 Acquisition Requirement (Input) Review....21
7.3 Test Plan Development and Review....21
7.4 Acquisition Review....22
7.5 Testing and Review....22
7.6 Review and Approval for Use....22
8.0 TOOLBOX CODE V&V AND SWAU....22
9.0 SIMPLE & EASILY UNDERSTOOD NON-SSC SOFTWARE V&V/SWAU....23
APPENDICES, FORMS, AND REFERENCES....24

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Added acquisition document review and an example Request Package Checklist; clarifications.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

1.0 PURPOSE AND APPLICABILITY

Purpose: This section of describes the "how, when, and who" for:

- verifying and validating (V&V) software, and
- approving software for use (SWAU)

See Tables 21.4-1 and 21.4-2 below for a summary of SOFT-V&V for SSC software and Non-SSC software respectively.

Applicability: See Chapter 21 applicability in SOFT-GEN. In addition:

1. Apply the V&V and SWAU subsections that are applicable to your software. For example, for acquired Non-SSC software, Subsections 1.0–3.0 and 7.0 apply.
2. For software changes during V&V, see SOFT-GEN.
3. Use this section for V&V and SWAU of software that is acquired and for software that is designed.
4. Read-Only: This section includes V&V for Read-Only SSC software. This is SSC software where the computer program cannot be modified other than through replacement of the computer program or the associated SSC. It is sometimes referred to as embedded software or firmware; however, for this section, the term Read-Only SSC software is used.
5. For SSC software, use this section in conjunction with ESM Chapter 15, *Commissioning* or AP-341-801, *Post Modification/Post Maintenance Testing.*

   Note: Throughout this chapter, the terms commission, commissioning, Cx documents, etc. refer to either of the above test documents, based on applicability.
6. For Non-SSC software, this section includes V&V and SWAU for: (a) toolbox codes and (b) software used in SSC design that is Simple and Easily Understood. See SOFT-GEN for definitions.
7. For computer programs that are commercially dedicated, the applicable V&V requirements of this section are required for critical characteristics specified in the CGD technical evaluation. SWAU requirements of this chapter also apply.

Table 21.4-1 Software SOFT-V&V Summary for SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[5]
		1, 2	3	4	How	When[2]	Who[3, 4]	
1.0 For Designed SSC Software								
1	Review design (input) requirements	R	Gr	Gr	▪ Review design input documents (docs.) to SOFT-DESIGN design (input) requirement criteria ▪ As applicable to ML-1 and ML-2 software, ensure Commercial Grade Dedication (CGD) requirements are addressed	▪ Prior to software design as much as practical	▪ SD (R) ▪ SRLM (R) ▪ LCxA (R) ▪ FDAR (R)	SOFT-V&V, 4; SOFT-DESIGN, 3
2	Develop/ review test plan (SWTP)	R	Gr	Gr	▪ Develop software test plan (SWTP) and submit 60%, 90% and 100% test plan submittals ▪ Review test plan to SOFT-V&V test plan criteria ▪ Integrate into/with SSC Commissioning (Cx) docs.	▪ After requirements and concurrent with design ▪ Prior to interim testing and/or acceptance testing	▪ SD (D, R, A) ▪ LCxA (R) ▪ SRLM (R, A)	SOFT-V&V, 4
3	Review software design (SWDD)	R	Gr	Gr	▪ Review to SOFT-DESIGN and SOFT-V&V criteria ▪ Review 60%, 90% and 100% software design documents/submittals	▪ Per software design schedule ▪ Prior to acceptance testing	▪ SD (R) ▪ SRLM (R) ▪ IR (R)	SOFT-V&V, 4; SOFT-DESIGN, 3,4
4	Review computer program code	R	-	-	▪ Review computer program code where feasible	▪ Prior to acceptance testing	▪ SD (R) ▪ SRLM (R) ▪ IR (R)	SOFT-V&V, 4
5	Testing/ Commission-ing (Cx) & review	R	Gr	Gr	▪ Test (including interim tests) per SSC design, software test plan, and Cx docs. ▪ Perform acceptance test in simulated test environment (i.e., test bed) ▪ Perform acceptance test in operating environment ▪ Provide, review & approve test (qualification) report (SWTR)	▪ Per software test plan, Cx docs, and SSC design ▪ Prior to approval for use	▪ T (D, R, A) ▪ SD (R, A) ▪ SO (R, A) ▪ LCxA, (R, A) ▪ SRLM (R, A)	SOFT-V&V, 4
6	Review/ SWAU	R	Gr	-	▪ Prepare software approval for use (SWAU) request package ▪ Review to SOFT-MAINT SWAU criteria ▪ Approve and document SWAU	▪ Prior to intended use	▪ SO (D) ▪ SRLM (R, A)	SOFT-V&V, 4; SOFT-MAINT, 3

Table 21.4-1 Software SOFT-V&V Summary for SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[5]
		1, 2	3	4	How	When[2]	Who[3, 4]	
2.0 For Read Only SSC Software								
1	Review (Input) requirements	R	Gr	Gr	▪ Review SSC design/acquisition requirement (input) docs. (e.g. Statement of Work, [SOW]) per governing doc. control process ▪ As applicable to ML-1 and ML-2 software, ensure CGD requirements are addressed	▪ Prior to acquisition	▪ Per governing doc. control process	SOFT-V&V,5; SOFT-DESIGN, 3; SOFT-ACQUIRE, 3
2	Develop and review test plan	R	Gr	Gr	▪ Use SSC Cx documents per ESM Chapter 15 as software test plan documents	▪ Per ESM Chapter 15 ▪ Prior to Cx	▪ Per ESM Chapter 15	SOFT-V&V, 5
3	Review the design	R	Gr	Gr	▪ Review software as part of the SSC design deliverables (ESM Chapter 1)	▪ Per SSC design schedule	▪ Per governing SSC review process	SOFT-V&V, 5
4	Commission-ing and Review	R	Gr	Gr	▪ Test per SSC design and Cx docs. ▪ Provide, review, and approve Cx report per ESM Chapter 15	▪ Per Cx docs. and SSC design ▪ Prior to approval for use	▪ Per ESM Chapter 15	SOFT-V&V, 5;
5	Review and approve for use	R	Gr	-	▪ Prepare SWAU request package ▪ Review to SOFT-MAINT SWAU criteria ▪ Approve and document SWAU	▪ Prior to intended use	▪ SO (D) ▪ SRLM (R, A)	SOFT-V&V, 5; SOFT-MAINT, 3

[1] ML = Management Level; R = Required; Gr = Required but graded. "-" = Not required

[2] "When" constitutes the V&V control points (points in the process where reviews are performed)

[3] D = Develop or Execute; R = Review; A = Approve

[4] SD = Software Designer; SO = Software Owner. SRLM = Software Responsible Line Manager. FDAR = Facility Design Authority Representative; T = Testers; IR = Independent Reviewer. LCxA = LANL Commissioning Authority. See associated ESM Chapter 21, SOFT-GEN for more information

[5] Ref. = ESM Chapter 21 section name and subsection number (e.g., section SOFT-GEN, Subsection 3, Software Identification and Determination)

Table 21.4-2 SOFT-V&V Summary for Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[6]
		1, 2	3	4	How	When[2]	Who[3, 4]	
1.0 For Designed Non-SSC Software								
1	Review design (input) requirements	R	R	R	▪ Review design input docs. to SOFT-DESIGN design (input) requirement criteria ▪ As applicable to ML-1 and ML-2 software, ensure Commercial Grade Dedication (CGD) requirements are addressed	▪ Prior to software design as much as practical ▪ Prior to software acquisition	▪ SD (R) ▪ SRLM (R)	SOFT-V&V, 6; SOFT-DESIGN, 6
2	Model V&V	R	Gr	Gr	▪ As applicable, V&V the appropriateness of the model in addition to the computer code ▪ Document in design and testing reviews	▪ After requirements and concurrent with design	▪ SRLM (R, A)	SOFT-V&V, 6; SOFT-DESIGN, 6
3	Develop/ review test plan	R	Gr	-	▪ Develop software test plan (SWTP) (e.g., Form 3055 (future)) and submit 60%, 90% and 100% test plan submittals ▪ Review test plan to SOFT-V&V test plan criteria	▪ After requirements and concurrent with design ▪ Prior to formal interim testing and/or acceptance testing	▪ SD (D, R, A) ▪ SRLM (R, A)	SOFT-V&V, 6
4	Review software design	R	Gr	Gr	▪ Review to SOFT-DESIGN design and SOFT-V&V criteria ▪ Review 60%, 90% and 100% software design documents/submittals	▪ Per software design schedule ▪ Prior to acceptance testing	▪ SD (R) ▪ SRLM (R) ▪ IR (R)	SOFT-V&V, 6; SOFT-DESIGN, 6, 7
5	Review computer program code	R	-	-	▪ Review computer program code where feasible	▪ Prior to acceptance testing	▪ SD (R) ▪ SRLM (R) ▪ IR (R)	SOFT-V&V, 6, 8
6	Testing and review	R	Gr	Gr	▪ Test per software test plan ▪ Provide, review and approve test report (SWTR)	▪ Per software test plan, ▪ Prior to approval for use	▪ T (D, R, A) ▪ SD (R, A) ▪ SO (R, A) ▪ SRLM (R, A)	SOFT-V&V, 6

Table 21.4-2 SOFT-V&V Summary for Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[6]
		1, 2	3	4	How	When[2]	Who[3, 4]	
7	Review/ SWAU	R	Gr	-	▪ Prepare SWAU request package ▪ Review to SOFT-MAINT SWAU criteria ▪ Approve and document SWAU	▪ Prior to intended use	▪ SO (D) ▪ SRLM (R, A)	SOFT-V&V, 6; SOFT-MAINT, 4
2.0 For Acquired Non-SSC Software								
1	Model V&V	R	Gr	Gr	▪ As applicable, V&V the appropriateness of the model in addition to the computer code ▪ Document in acquisition and testing reviews	▪ Should be completed prior to acquisition; must be completed prior to SWAU	▪ SRLM (R, A)	SOFT-V&V, 7; SOFT-DESIGN, 7 SOFT-ACQUIRE, 4
2	Review acquisition (input) requirements	R	R	R	▪ Review acquisition input docs. (e.g., SOW) to SOFT-ACQUIRE acquisition (input) requirement criteria ▪ As applicable to ML-1 and ML-2 software, ensure CGD requirements are addressed	▪ Prior to software acquisition	▪ SO (R) ▪ SRLM (R)	SOFT-V&V, 7; SOFT-ACQUIRE, 4
3	Develop/ review test plan	R	Gr	-	▪ Develop software test plan (SWTP) (e.g., Form 3055 (future)) ▪ Review test plan to SOFT-V&V test plan criteria	▪ After requirements and concurrent with acquisition ▪ Prior to acceptance testing	▪ SD (D, R, A) ▪ SRLM (R, A)	SOFT-V&V, 7
4	Review acquisition	R	Gr	-	▪ Review and document conformance with acquisition documents	▪ Prior to acceptance testing	▪ SO (D)	SOFT-V&V, 7
5	Testing and review	R	Gr	Gr	▪ Test per software test plan ▪ Provide, review and approve test report (SWTR)	▪ Per software test plan, ▪ Prior to approval for use	▪ T (D, R, A) ▪ SD (R, A) ▪ SO (R, A) ▪ SRLM (R, A)	SOFT-V&V, 7

Table 21.4-2 SOFT-V&V Summary for Non-SSC Software
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[6]
		1, 2	3	4	How	When[2]	Who[3, 4]	
6	Review/ SWAU	R	Gr	-	▪ Prepare SWAU request package ▪ Review to SOFT-MAINT SWAU criteria ▪ Approve and document SWAU	▪ Prior to intended use	▪ SO (D) ▪ SRLM (R, A)	SOFT-V&V, 7; SOFT-MAINT, 4
3.0 For Acquired Non-SSC (OTS) – Toolbox Codes								
1	V&V and SWAU	R	R	R	▪ V&V per the Registry instructions for the toolbox code ▪ Prepare SWAU request package ▪ Review to SOFT-MAINT SWAU criteria ▪ Approve and document SWAU	▪ For each toolbox code installation ▪ Prior to intended use	▪ SO (D, R, A) ▪ SRLM (R, A)	SOFT-V&V, 8; SOFT-MAINT, 4
4.0 For Simple & Easily Understood Non-SSC Software Used in Design of SSCs								
1	V&V as part SSC design V&V	R	Gr	Gr	▪ V&V using the methods previously described for Non-SSC software, or, V&V the software each time it is used for SSC design in accordance with the SSC design process ▪ Provide and review minimum software information (e.g., computer-generated evidence of the programmed algorithms or equations) ▪ Apply appropriate SSC design review rigor to ensure the computer program produces satisfactory results.	▪ Each time the software is used for an SSC design and when required by the SSC design V&V process ▪ Prior to intended use	▪ As per the applicable SSC design V&V process	SOFT-V&V, 9; SOFT-MAINT, 4

[1] ML = Management Level; R = Required; Gr = Required but graded

[2] "When" constitutes the V&V control points (points in the process where V&V activities are performed)

[3] D = Develop or Execute; R = Review or Inspect; A = Approve

[4] SD = Software Designer; SO = Software Owner. SRLM = Software Responsible Line Manager. FDAR = Facility Design Authority Representative; T = Testers; IR = Independent Reviewer

[5] A SD subject matter expert (SME) other than the SD that developed the code shall be used to perform the code review

[6] Ref. = ESM Chapter 21 section name and subsection number (e.g., Section SOFT-GEN, Subsection 3, Software Identification and Determination)

2.0 KEY DEFINITIONS AND ACROYNMS

See SOFT-GEN for definitions and acronyms and note the following: Often the terms verification and validation are confused with each other. In simplified layman's terms: verification indicates whether *"you built the item correctly"* whereas validation indicates whether *"you built the right item."*

3.0 V&V AND SOFTWARE APPROVAL FOR USE (SWAU) - GENERAL

The following requirements apply to both SSC and Non-SSC software:

A. Base the extent of V&V on the complexity of the software, the degree of standardization, the similarity with previously approved software, and the importance to safety.

Note: For the following Paragraphs B–E, the documentation details are specified in subsequent subsections.

B. For ML-1 and ML-2 software, trace, document, and V&V the software from requirements development through design, acquisition, testing and SWAU. Refer to ESM Chapter 20, *Systems Engineering* (future). See SOFT-V&V *References: V&V Examples* for an example software requirements matrix (SWTM).

C. For ML-1 through ML-3 software, document the V&V results, including the identification of the verifier/validator.

D. For ML-1 and ML-2 reviews, document the reviewers, reviewer responsibilities, comments, and dispositions; retain reviewer comments and dispositions as records per the record retention schedule but at least until the software is approved for use.

E. For ML-3 and ML-4 reviews, at a minimum, document the reviewers (names) and their comments, and retain the comments as records.

F. Ensure the limits of use associated with the V&V are clearly defined. If the software is intended to be used outside the original limits of use, evaluate and, as appropriate, perform additional V&V to determine if the software can be used for outside the original limits.

G. For other than acceptance testing reviews, if the review alone is inadequate to determine if requirements are satisfied, use alternate calculations and/or develop and integrate (interim) tests (may include demonstrations) to support the review.

H. Suppliers that design SSC and Non-SSC software for LANL must V&V software based on Subsection 4 and Subsection 6, respectively. The V&V must be performed as part of the supplier's software product development. For ML-1 and ML-2 software, suppliers must provide V&V processes and objective evidence of the supplier's V&V for LANL review and retention.

I. LANL workers must complete and document reviews and retain review comments and dispositions as per this section and the governing document control and records management process. For ES-Div Non-SSC software and SSC software, ensure the correct SWID (obtained when completing Form 2033) is part of the documentation record number. See AP-341-402, *Engineering Document Management in Operating Facilities* for details.

J. Protect against suspect/counterfeit items (S/CI) items. As applicable, review and ensure valid software licenses. See P330-9, Suspect Counterfeit Items (S/CI) for additional information on identifying and processing an S/CI item.

K. Software design verification shall be performed by competent individual(s) or group(s) other than those who developed and documented the original design, but who may be from the same organization. Cursory supervisory reviews do not satisfy the intent of this chapter. Verification may be performed by the originator's supervisor, provided (a) the supervisor did not specify a singular design approach or rule out certain design considerations and did not establish the design inputs used in the design, or (b) the supervisor is the only individual in the organization competent to perform the verification.

L. Verification must include any one or a combination of design reviews, alternate calculations, and tests performed during computer program development. Although multiple reviews are specified herein, the reviews may be performed and documented separately or combined, as appropriate.

M. Software tools must be evaluated, reviewed, tested, accepted for use, and placed under configuration control; software tools that do not affect the performance of the software need not be placed under configuration control.

N. When commercial grade dedication (CGD or dedication) is used to V&V software, satisfy NQA-1, Part II, Subpart 2.14, *Quality Assurance Requirements for Commercial Grade Items and Services* and the following:

1. The dedication process shall be documented and include:

   a. Identification of the capabilities and limitations for intended use as critical characteristics,

   b. Utilization of test plans and test cases as the method of acceptance to demonstrate the capabilities within the limitations, and

   c. Instructions for use (e.g., user manual) within the limits of the dedicated capabilities.

2. The performance of the actions necessary to accept the software shall be reviewed and approved. The resulting documentation and associated computer program(s) shall establish the current baseline.

3. For guidance in dedicating computer programs, see ASME NQA-1-2015 Part III, Subpart 3.2-214, *Quality Assurance Requirements for Commercial Grade Items and Services, Commercial Grade Computer Programs, and Software Services* and the references therein.

4. Subsequent revisions of ML-1 and ML-2 software from non-NQA-1 qualified suppliers that do not follow NQA-1 Subpart 2.7 to V&V the software for acceptance (as described in SOFT-V&V) shall be dedicated. See SOFT-GEN for details. Unless a more conservative approach is required by the LANL SRLM, use the following graded approach for dedicating Ml-1 and ML-2 software revisions (changes):

   a. Dedicate major changes (see Definitions).

   b. Dedicate minor changes (see Definitions) if the changes affect the critical characteristic(s) that pertain to the functionality of the computer program or the associated SSC.

5. Incorporate the following guidance as applicable, when dedicating computer programs (Ref. NQA-1-2015, Part III, Subpart 3.2-2.14):

   a. Guidance: Acceptable data for historical performance should evaluate the industry-monitored performance of the commercial

grade computer program, industry product tests, certification to national codes and standards (nonnuclear-specific), and other industry records or databases. When a computer program has been demonstrated to be reliable based on its historical performance, it should be credited during dedication. Historical performance should be supported by the use of one of the other verification methods listed above.

b. This acceptance method [Method 4: Acceptable Supplier Item or Service Performance Record] should have a greater application for the dedication of computer programs used in design or analysis. Computer programs that are commercially available and that have industrywide application may be used successfully hundreds or even hundreds of thousands of times daily. The results of these uses and engineering judgment associated with the acceptance of the computer program should be considered when dedicating the computer program.

c. Errors reported by the users to the supplier and failures associated with structures, systems, and components may be evaluated as part of the failure analysis investigation. This method is most effective when the supplier provides error reports to the purchaser for applicability and significance evaluation and when the users contact the supplier when computer program errors are suspected. A technical support agreement in the procurement documents provides assurance that there is adequate communication between the supplier and users.

4.0 DESIGNED SSC SOFTWARE V&V AND SWAU

Note: **To minimize text duplication, many of the Non-SSC software subsections that appear later (e.g., 5-7) refer back to this SSC-related subsection with instructions that the SSC-related requirements do not apply. Therefore, to facilitate comprehension, Subsection 4.0 SSC-only requirements are followed by a red "*" at the end of the text and these should be modified or ignored in accordance with direction in those later Subsections. If additional clarification is needed, contact the Chapter POC.**

Table 21.4-3 summarizes the minimum V&V and SWAU activities for designed SSC software. The text that follows it provides additional detail.

Table 21.4-3 Minimum Required V&V Activities - Designed SSC Software					
No	V&V Activity Description	Apply by ML[1]			
		1	2	3	4
1	Design Requirement (Input) Review	R	R	R	Gr
2	Software Test Plan (SWTP) Development & Review[2]	R	R	Gr	Gr
3	Software Design Review[3]	R	R	Gr	Gr
4	Software Code Review[4]	R	R	-	-
5	Testing/Cx & Review*[2]	R	R	Gr	Gr
6	Software Review and Approval for Use (SWAU)	R	R	Gr	-

[1] R = Required; Gr = Required but graded; "-" = Not required

[2] See ESM Chapter 15, *Commissioning* for minimum test/commissioning (Cx) requirements based on ML

[3] See SOFT-DESIGN for design deliverables

[4] Where feasible as determined by the LANL SRLM

4.1 Design Requirement (Input) Review

A. How. In accordance with the governing document control process[1], review and document the review of design (input) requirements to ensure they satisfy the design requirement criteria in SOFT-DESIGN. As guidance for ML-1 through ML-3 software, use the requirement review criteria from IEEE Std 1012-2012, *IEEE Standard for System and Software Verification and Validation*[2]. For CGD of ML-1 and ML-2 software in accordance with AP-341-703, Commercial Grade Dedication or equivalent, ensure that CGD requirements are addressed.

B. When. As much as practical, review prior to software design.

C. Who. The Software Designer (SD), SRLM, LANL Commissioning Authority (LCxA) and FDAR review ML-1 through ML-3 software. * The FDAR review is not required for ML-4 software.

4.2 Test Plan Development and Review

Tests to verify conformance of a computer program to requirements must be planned and executed. The software test plan (SWTP) may include interim testing to support software design, acceptance testing to support commissioning (Cx) and/or in-use testing to support operations after the software is approved for use (SWAU). * Therefore, because the test plan may be broader than acceptance testing, it is not used synonymously with Cx test planning documents. The acceptance testing portion of the test plan however, must be integrated into or with Cx documentation. *

For testing that requires changes to the approved configuration management SSCs of an operating facility, see AP-341-504, *Temporary Modification Control*.*

[1] See AP-341-620, *Review of LANL Produced Design Documents*; AP-341-621, *Design Authority Technical Review*, AP-341-622, *LANL Review of Designs Produced by External Agencies*, and AP-321-624, *Independent External Design Review*, or approved equivalent.

[2] The V&V tasks/criteria in this standard uses system integrity levels (SILs) for grading instead of MLs; use the following approximate correlation: ML-1 = SIL 4; ML-2 = SIL 3; ML-3 = SIL 2; ML-4 = SIL 1.

A. How. Plan tests as required by the SSC design, ESM Chapter 15, *Commissioning (Cx)*[3] and this Subsection. Document in a test plan. * Review and disposition comments in accordance with the governing document control process.

Address the following in the test plan:

1. Perform at least one acceptance test in a simulated environment (i.e., test bed or bench testing) for ML-1 software. * For ML-2 software, perform at least one acceptance testing in a simulated environment when specified by the FDAR. * Perform simulated environment acceptance tests prior to acceptance testing in the actual operating environment. *

2. As determined by the FDAR for ML-1 and/or ML-2 software, where the simulated environment without hardware cannot adequately test the software, test in an environment with hardware. *

3. Perform a comprehensive, end-of-development acceptance test in the operating environment[4] prior to use. Include support software tests as appropriate.

4. As a recommendation and/or as required by other ESM chapters, plan for tests during various points in the software design (interim tests) in addition to the acceptance test. Interim tests may be informal tests or formal tests to support reviews. Test plan documentation for informal testing is recommended, but not required.

5. Test plan documentation for formal interim tests used to support reviews is required and must satisfy the same applicable criteria as the acceptance test[5]. Include the tests and test results in the acceptance testing documentation.

6. Coordinate and integrate the software test plan and testing with Cx and the governing (SSC) work control requirements. *

7. For ML-1 through ML-3 software, include software testing required after approval for use (in-use tests) that confirms continued acceptable performance in the operating system. Include either directly in the test plan or through reference to other documents. Perform in-use tests after the computer program is installed on a different computer, or when there are significant changes in the operating system.

   Note: The in-use test frequency for ML-3 software is typically less frequent than for ML-1 or ML-2 software. ML-3 in-use tests may be simplified tests that address only the key required software functions. Testing of all functional requirements is not required for ML-3 in-use testing.

8. Create a test record form or template that contains, at a minimum, the following:

   a. SSC(s)/item tested, *

   b. date of test,

[3] See AP-350-406, *Start-up and Commissioning* or equivalent for details.

[4] As determined by the LANL SRLM, where testing in the operating environment introduces unacceptable risk or loss, an exception to performing testing in the operating environment may be pursued in accordance with ESM, Chapter 1, General, Z10, *General Requirements for All Disciplines/Chapters*.

[5] From NQA-1. Note that "same criteria" allows one to employ a subset of the full suite of acceptance test criteria for unit testing (i.e., white box testing); program, subsystem, or system testing (i.e., gray-box testing).

c. characteristics tested,

d. tester (name) and/or data recorder,

e. type of observation (i.e., type of test),

f. results, applicability and acceptability (of the test),

g. action taken in connection with any deviations,

h. reference to the test methodology/test plan used with associated revision,

i. person evaluating and accepting test results,

j. a unique test record identifier to allow traceability to the specific item tested (if not subsequently assigned as part of the records management protocols),

k. computer program tested including system software, as applicable[6],

l. computer hardware, as applicable

m. test equipment and calibrations, as applicable, (Include equipment identification, calibration date, and calibration expiration date for M&TE, Ref. P330-2)

n. simulation models used, as applicable, and

o. test problems/cases.

9. Incorporate Appendix A, *Software Test Plan Requirements Table*.

10. See ISO/IEC/IEEE 29119, *Software and systems engineering—Software testing*, for guidance.

11. Provide 60%, 90% and 100% test plan submittals for review and approval. See References, *V&V Examples* for an example test plan.

12. When dedicating software in accordance with AP-341-703, *Commercial Grade Dedication* or equivalent, the acceptance test plan may be a separate test plan or combined with the CGD Technical Evaluation and Acceptance (TEA) Plan. If combined, ensure CGD requirements such as critical characteristics are specified.

13. Identify and coordinate with associated SSC tests (e.g., qualification tests for seismic qualification of electrical equipment, Ref. IEEE Std 344, *Standard for Seismic Qualification of Equipment for Nuclear Power Generating Stations.*) *

14. Review, approve and control the test plan (including test record form) in accordance with the governing document control process (e.g., controlled and accessible in EDMS). Review for adequacy and conformance to this chapter.

15. Maintain test plan accuracy through approval for use.

16. As required, identify hold and/or witness points and associated responsibilities.

17. For ML-1 and ML-2 software, design tests to demonstrate performance requirements are satisfied and relevant unintended defects are detected. In

[6] For read-only software, identify computer program name to the extent possible (See SOFT-GEN).

layman's terms, design tests to try to "break the software"[7]. Incorporate the following:

a. Incorporate hazards analysis and industry accepted test strategies and techniques.[8]

b. Ensure the integrated software/SSC system can handle and recover from errors. * *Ensure testing addresses latent run-time errors that may occur (e.g., an error that causes a program to continually use up Randam Access Memory (RAM) while the program is running that results in a system "crash" after the system is in operation for a time.)*

c. Demonstrate that loss of operational control (i.e., the computer program crashes) does not occur in normal and abnormal conditions and/or that software errors do not damage or otherwise cause SSC failures or unsafe conditions. *

18. For final acceptance tests in the operating environment that are used to verify the quality of ML-1 and ML-2 SSCs in Hazard Category II or II nuclear facilities, include the requirement for testers to satisfy the tester certification requirements of P330-8, *Inspection and Test*. See ESM Ch. 15, and/or AP-341-801, *Post Modification/Post Maintenance Testing* for certification details.

B. When. Develop after software requirements have been established, and concurrent with the software design. Submit draft test plans with draft software design deliverables. Submit final test plan with the final software design and prior to acceptance testing.

C. Who. A SD develops, reviews and approves the test plan (including test record form); the LCxA reviews; * the SRLM reviews and approves. *Review by a quality assurance subject matter expert (QA SME) is recommended for ML-1 and ML-2.*

Note: Where possible, the SD should be a subject matter expert (SME) other than the SD that designed the software.

4.3 Software Design Review

A. How. In accordance with the governing document control process[9], review and document the review to ensure the software design satisfies the software design criteria specified in SOFT-DESIGN. Perform the following as part of the design review:

1. Review for acceptability of the design requirements, the design and conformance to the design process.

2. Evaluate the technical adequacy of the approach;

3. Ensure internal completeness, consistency, clarity, and correctness of the software design.

---

[7] If testing of the completed integrated software/hardware unit adequately tests the software, then software-only testing is not required.

[8] See *Certified Software Quality Engineer Handbook*, L. Westfall, American Society of Quality Press.

[9] See AP-341-620, *Review of LANL Produced Design Documents*; AP-341-621, *Design Authority Technical Review*, AP-341-622, *LANL Review of Designs Produced by External Agencies*, and AP-321-624, *Independent External Design Review* or equivalent.

4. Verify that the software design is compliant with and traceable to the approved software design requirements.

5. Review test results.

As guidance for ML-1 through ML-3 software, use design review criteria from IEEE Std 1012, *IEEE Standard for System and Software Verification and Validation*[10] and IEEE STD 1028, *IEEE Standard for Software Reviews and Audits.*

B. When. Review in accordance with the software design deliverable submittal schedule and prior to acceptance testing.

C. Who. The SD and SRLM review. In addition, for ML-1 and ML-2 software, one or more individuals other than those who designed the software must review (Independent Reviewers, or IRs). IRs may be from the same organization.

4.4 Computer Program Code (Code) Review

A. How. In accordance with the governing document control process, review and document the review to ensure the code satisfies the criteria in SOFT-DESIGN. *Guidance: Use the code review criteria from IEEE Std 1012, IEEE Standard for System and Software Verification and Validation.*[11]

B. When. Review the code prior to acceptance testing.

C. Who. The Software Designer (SD) and SRLM review. In addition, an independent reviewer (IR) SME other than the SD that developed the code shall review.

4.5 Testing/Cx, Review, and Report

A. How. For SSC software (as well as Non-SSC software) test, review and provide test (also commonly referred to as qualification) reports in accordance with the design, software test plan, and as applicable, Cx documents. These reports are referred to as SWTRs. Ensure the following:

1. Complete activities to prepare for testing in accordance with the governing (SSC) work control documents and: *

   a. Ensure required software backups are accessible in the event that a need to “rollback” to original files arises.

   b. Ensure that documentation required prior to testing has been received.

   c. Review results and resolution of nonconformances identified by prior tests (e.g., Nonconformance reports (NCRs), bug/error reports, etc.). Ensure that issues have been sufficiently resolved to proceed with the test.

   d. Ensure suitable environmental conditions exist throughout the testing process.

   e. As required for integrated software/hardware testing, ensure M&TE such as calipers, thermometers, etc. are in calibration, ready for use, and that proper M&TE maintenance, storage and handling is in place

[10] To apply the V&V tasks/criteria in this standard that uses system integrity levels (SILs) for grading instead of MLs, use the following correlation: ML-1 = SIL 4; ML-2 = SIL 3; ML-3 = SIL 2; ML-4 = SIL 1.

[11] As determined by the LANL SRLM, when it is not feasible for LANL to obtain source code (e.g., for proprietary reasons), review the supplier’s code review V&V process and objective evidence.

in accordance with P330-2, *Control and Calibration of Measuring and Test Equipment (M&TE)*, or equivalent. *

f. Ensure readiness of temporal test requirements (e.g., ensuring witnesses are present and ready when required etc.).

g. Verify that those performing and/or supporting testing have the minimum training, qualification, authorization, and as appropriate, certifications. NQA-1 certified testers are required for ML-1 and ML-2 SSC testing in the operating environment in Hazard Category II or III nuclear facilities (Ref. P330-8).

h. Ensure a clear understanding of the roles and responsibilities for those performing the test(s), as well as those that may be affected by the test(s).

i. Perform other activities as may be required by the test plan.

j. Conduct tests under controlled conditions using established acceptance and performance criteria.

k. Ensure specified calibration, environmental conditions, and storage requirements are maintained throughout the test process. *

l. Ensure configuration requirements are satisfied throughout the duration of the test.

m. Indicate the status of the test (i.e. tested, not tested, testing in process). Indicate on the SSC or in documents traceable to the SSC to ensure that required tests are performed and to ensure SSCs that have not passed tests are not inadvertently used or operated. For ML-1 and ML-2 software, use physical location, tags, markings, shop travelers, stamps, inspection records, or other suitable means to indicate test status. For Non-SSC software, requiring software to be maintained in a test environment until acceptance testing is complete is an acceptable method to control the test status. Specify the authority for application and removal of tags, markings, or removal from the test environment etc.

n. Retest, (e.g., regression test) as appropriate, modified and/or replaced software.

o. Ensure the required post-test configuration is in place after completing the test. *

p. Document the test results on a test record form to indicate the ability of the software to satisfactorily perform its intended function/performance criteria. Include the minimum test record information on the form or template (see Section 4.2).

q. As applicable, ensure that CGD requirements are addressed.

r. Review and evaluate the test results to requirements and ensure requirements have been satisfied. Assure that the software produces correct results. Determine and document the acceptability of the test in a test (qualification) report (SWTR). Review in accordance with the test plan. Address the following:

1) Perform preliminary evaluations to determine the validity of the test results and the appropriateness of further testing.

Identify additional test requirements and required changes to the test plan. Retest as required.

2) Analyze and evaluate the data to verify completeness of results, conformance to design (input) requirements, achievement of test objectives, and conformance to the test plan. See References: *V&V Examples* for an example report.

2. Retain test documentation per the governing document control/records management processes. Ensure test records have a unique identifier and are traceable to the software.

B. When. Conduct tests when specified in the SSC design, test plan, and/or Cx documents and prior to approval for use. *

C. Who.

1. For ML-1 – ML-3 software, authorized testers (Ts), as identified in the software plan and/or Cx documents perform tests. * For ML-4 software, Ts as identified in the SSC design and/or Cx documents perform tests. *

2. For ML-1 – ML-3 software, those responsible for review and approval are as indicated in the test plan however must include the following: Lead Tester (T), SD or other SD SME individual familiar with the design detail and intended use of the computer program, SO, LCxA and SRLM. *

3. For ML4, authorized Ts shall be facility operations and/or Cx personnel authorized by facility management. The SO, LCxA and SRLM review and approve acceptance tests. *

4.6 Review and Approval for Use

A. How. For suppliers, perform software product review and release in accordance with the supplier's quality assurance program. For LANL workers that are approving ML-1 through ML-3 software for use at LANL, review and approve for use (SWAU) in accordance with this Subsection. SWAU is not required for ML-4 software unless invoked by the LANL SLRM.

1. Prepare a SWAU request package that demonstrates: (a) satisfactory completion of V&V, (b) the applicable operational and maintenance (O&M) controls in SOFT-MAINT are in place, and (c) the software is ready to be used in its intended environment.

2. Perform a review of the SWAU request package using the criteria from SOFT-V&V and SOFT-MAINT and document the review and SWAU[12]. For ML-1 and ML-2 software, include (either directly or through reference) the intended use and any associated limitations, access controls, etc. for using the software. See SOFT-V&V-FM01, *Software Approval for Use Form (SWAU, Sample) With Example*.

3. For changes to software in LANL facilities, AP-341-507, *SSC Software Change Package* and AP-341-801, *Post Modification/Post Maintenance Testing* may be used to document SWAU. *

[12] In cases where the acquisition predated mandatory use of Chapter 21, use the above and/or criteria from the approved QAP and implementing procedures used to deliver the item. Same logic applies to Read-only and Non-SSC software.

4. For startup and restart of ML-1 through ML-3 software in LANL nuclear facilities, the operational readiness processes may be used to document SWAU. See PD115, *LANL Readiness Program.*

B. When. Review and SWAU prior to intended use (operation).

C. Who. The SO prepares the SWAU request package. The SRLM reviews and approves ML-1 through ML-3 software for use. If AP-341-507, PD115, or other processes are used for approval for use, the reviews and approvers are as specified in the respective process. *

5.0 READ-ONLY SSC SOFTWARE V&V AND SWAU

This Subsection applies to LANL workers that acquire Read-Only SSC software. Read-Only software is software where the computer program cannot be changed other than through replacement of the computer program and/or the associated SSC.[13]

Table 21.4-4 presents the minimum V&V and SWAU activities that must be performed for Read-Only SSC software. The following text provides additional detail.

Table 21.4-4 Minimum Required V&V Activities - Read Only SSC Software

No	V&V Activity Description	Apply by ML[1]		
		-1, -2	3	4
1	Requirement (Input) Review	R	Gr	Gr
2	Test Plan Development and Review[2]	R	Gr	Gr
3	Design Review	R	Gr	Gr
4	Testing/Cx and Review[2]	R	Gr	Gr
5	Review and Approval for Use (SWAU)	R	Gr	-

[1] R = Required; Gr = Required but graded; "-" = Not required.

[2] See ESM Chapter 15, *Commissioning* for minimum test/commissioning (Cx) requirements based on ML.

5.1 Requirement (Input) Review

Read Only SSC software input requirements are specified in the SSC design and/or acquisition requirement (input) documents. Examples include a Requirements Criteria Document (RCD) and Statement of Work (SOW). Prior to acquisition, review the SSC requirement to the requirement criteria of the respective document in accordance with the governing document control process. As applicable, ensure CGD requirements are addressed.

5.2 Test Plan Development and Review

Use the SSC Cx documents and SSC design as the software test plan documents. Apply the General Test Plan Requirements in Table 21.4-A1 to Cx as shown in the table. Use the remaining Subsection 4.2 requirements as Cx guidance. Review and approve the Cx documents per ESM Chapter 15 prior to testing.

[13] Facility/system control software is considered designed SSC software for the purposes of this chapter.

5.3 Design Review

Review the Read-Only software as part of the review of the associated SSC design in accordance with the SSC design deliverable schedule in ESM Chapter 1, Z10. Perform the review in accordance with the governing SSC design review process.[14]

For ML-1 and ML-2 software, review the supplier's V&V documentation of the software product (e.g., as specified in the Scope of Work such as Exhibit D or H). Review based on Subsection 4 criteria and as applicable, commercial grade dedication (CGD) requirements.

5.4 Testing/Cx and Review

Test per the SSC Cx and SSC design documents. Provide, review and approve Cx report per ESM Chapter 15 prior to approval for use.

5.5 Review and Approval for Use

See Subsection 4.6 and apply to Read-Only SSC software.

6.0 DESIGNED NON-SSC SOFTWARE V&V AND SWAU

Table 21.4-5 summarizes the minimum V&V and SWAU activities for designed Non-SSC software.

The following text provides additional detail.

Table 21.4-5 Minimum Required V&V Activities –Designed Non-SSC Software

No	V&V Activity Description	Apply by ML[1]		
		-1, -2	3	4
1	Model V&V[2]	R	Gr	Gr
2	Design Requirement Review	R	R	R
3	Test Plan (SWTP) Development and Review	R	Gr	-
4	Design Review[2]	R	Gr	Gr
5	Code Review[4]	R	-	-
6	Testing & Review	R	Gr	Gr
7	Review and Approval for Use (SWAU)	R	Gr	-

[1] R = Required; Gr = Required but graded; "-" = Not required.

[2] Required for software that includes models.

[3] See SOFT-DESIGN for design deliverables.

[4] Where feasible as determined by the LANL SRLM.

6.1 Model V&V

A. How. For software that includes models (see SOFT-DESIGN), verify and validate the appropriateness of the model in addition to the appropriateness of the computer code used to provide the outcome of the model. In addition to the requirements of this subsection, use industry accepted model V&V methods to quantify confidence and

[14] See AP-341-620, *Review of LANL Produced Design Documents*; AP-341-621, *Design Authority Technical Review*, and AP-341-622, *LANL Review of Designs Produced by External Agencies*.

build credibility in a numerical model for the purpose of making a prediction. [15,16] Example methods include alternate calculations, references to previously established standards and practices, and specialized peer reviews. Document the model V&V in addition to the computer code V&V as part of the software design and testing reviews.

B. When. After requirements and concurrent with design.

C. Who. The SRLM reviews and approves.

6.2 Design Requirement (Input) Review

See Subsection 4.1 and apply the following clarifications:

- Requirements associated with SSCs do not apply.
- FDAR and LCxA review are not required.

6.3 Test Plan Development and Review

See Subsection 4.2 and apply the following clarifications:

- Requirements associated with SSCs, including commissioning (Cx) and governing SSC work controls do not apply.
- Acceptance testing in a simulated environment (i.e., test bed) does not apply.
- See Form 3055, Software Test Plan (future) for an example test plan template

6.4 Software Design Review

See Subsection 4.3.

6.5 Designed Non-SSC Computer Program Code Review

See Subsection 4.4.

6.6 Testing and Review

See Subsection 4.5 and apply the following clarification:

- Requirements associated with SSCs, including testing in simulated environment, commissioning (Cx), governing SSC work controls and M&TE do not apply.

6.7 Review and Approval for Use

See Subsection 4.6 and apply the following clarifications:

- Requirements associated with SSCs, including AP-341-507, do not apply.
- For changes of ML-1 through ML-3 software in LANL facilities, use SOFT-GEN-FM03, *Non-SSC Software Change Package Form (SWNCP)* or equivalent.

7.0 ACQUIRED NON-SSC SOFTWARE V&V AND SWAU

Table 21.4-5 presents the minimum V&V and SWAU activities that must be performed for acquired Non-SSC software. The following text provides additional detail.

[15] LA-14167-MS, Los Alamos National Laboratory, *Concepts of Model Verification and Validation*, Thacker et al., 2004.
[16] ANSI/ANS 10.7-2013, *Non-Real-Time, High-Integrity Software for the Nuclear Industry – Developer Requirements.*

Table 21.4-5 Minimum Required V&V Activities – Acquired Non-SSC Software				
No	V&V Activity Description	Apply by ML[1]		
		1, 2	3	4
1	Model V&V[2]	R	Gr	Gr
2	Acquisition Requirement (Input) Review	R	R	R
3	Test Plan (SWTP) Development and Review[3]	R	Gr	-
4	Acquisition Review	R	Gr	-
5	Testing & Review[3]	R	Gr	Gr
6	Review and Approval for Use (SWAU)	R	Gr	-

[1] R = Required; Gr = Required but graded; * = Not Required.

[2] Required for software that includes models.

[3] For software acquired from an NQA-1 qualified supplier, most of the V&V requirements are satisfied by the supplier.

7.1 Model V&V

See Subsection 6.1. Model V&V should be completed prior to acquisition and must be completed prior to SWAU.

7.2 Acquisition Requirement (Input) Review

A. How.

1. In accordance with the governing acquisition and associated document control processes[17], review the acquisition documents (e.g., Statement of Work, Exhibit H etc.,) to ensure they satisfy the acquisition criteria in the SOFT-ACQUIRE. As applicable, ensure CGD requirements are addressed.

B. When. Review prior to software acquisition.

C. Who. The SO and SRLM review.

7.3 Test Plan Development and Review

See Subsection 4.2 and apply the following clarifications:

- Subsection 4.2 does not apply to ML-4 software unless invoked by the LANL SLRM. Install and test per the supplier instructions.
- For software obtained from an NQA-1 qualified supplier, most of the V&V requirements, including acceptance test plan and testing activities, have been performed by the supplier. Prepare a simplified test plan that uses the applicable supplier-provided test cases (and/or other test cases) to demonstrate the software is adequately installed, suitable for the intended application, and properly working in the intended operating environment. See Form 3055, Software Test Plan (future).
- For ML-1 and ML-2 **safety** software acquired from a non-NQA-1 qualified supplier, apply chapter test plan and testing requirements for the software critical characteristics specified in the CGD technical evaluation.
- For ML-3 safety software and ML-1 through ML-3 non-safety software, CGD is not required; however, prepare a test plan and perform testing in accordance with this

[17] See P840-1, *Quality Assurance for Procurements* or equivalent.

chapter for the key software functional requirements required for the intended application.

- Requirements associated with SSCs, including commissioning (Cx) and governing SSC work controls do not apply
- Acceptance testing in a simulated environment (i.e., test bed) does not apply
- Testing associated with software design (e.g., interim tests) do not apply
- Preparation or incorporation of a hazard analysis is not required
- The test plan is prepared, reviewed and approved in accordance with the governing document control process
- Supplier-provided test cases, documentation and comparative alternate calculations/published data may be used to demonstrate the software meets requirements for its intended use.
- Perform test plan development and review after establishing requirements, concurrent with the acquisition, and prior to acceptance testing.
- The SO or a SD may develop the test plan.

7.4 Acquisition Review

A. How. For ML-1 through ML-3 software, review the acquired software for conformance to the acquisition documents (see SOFT-ACQUIRE*).* Evaluate the technical adequacy and completeness and verify that the software is traceable to the acquisition requirements. As applicable, review in accordance with CGD requirements. As applicable, ensure valid software licenses and markings that indicate original manufacture's product to help prevent acquisition of suspect/counterfeit software. Document the review (checklist recommended) and retain a record of the review in accordance with the governing document control/records management processes. *Guidance: An example checklist is posted below this section on the website (Note: At time of writing, the example was for MACCS2, a safety basis toolbox code, so the checklist does not contain all the tasks/documents that would be appropriate for a non-toolbox program; these would need to be added by the checklist preparer).*

B. When. Review prior to acceptance testing.

C. Who. The SO performs the acquisition review.

7.5 Testing and Review

See Subsection 6.6 with the following clarifications:

- Requirements associated with software design (e.g., interim testing) do not apply.
- Subsection 6.6 does not apply to ML-4 software unless invoked by the LANL SLRM. Install and test per the supplier instructions.

7.6 Review and Approval for Use

See Subsection 6.7.

8.0 TOOLBOX CODE V&V AND SWAU

A. How. V&V toolbox codes (see SOFT-ACQUIRE for definition and information) in accordance with the Registry instructions for the specific toolbox code. Review and SWAU per Subsection 6.7.

B. When. Complete for each installation of a toolbox code.

C. Who. The SO performs and the SO and SRLM review and approve.

9.0 SIMPLE & EASILY UNDERSTOOD NON-SSC SOFTWARE V&V/SWAU

A. How. Perform V&V/SWAU on "Simple and Easily Understood" Non-SSC software used in the design of the SSCs per Subsections 6 or 7 (pre-verify), or V&V/SWAU as part of the SSC design V&V for each time it is used (individually verify).

1. To individually verify:

a. Follow the SSC design process. Document the design analysis (e.g., AP 341-605, *Calculations*, for LANL). Include the following in the calculation:

1) Whether the software is pre-verified or individually verified

2) the objective of the analyses

3) design inputs and their sources

4) results of literature searches or other applicable background data

5) assumptions and indication of those assumptions that must be verified as the design proceeds.

6) identification of any computer calculation, including identification of the computer type, computer program name, and revision, inputs, outputs, evidence or reference to computer program verification, and the bases (or reference thereto) supporting application of the computer program to the specific physical problem.

7) review and approval.

b. The following software information should be included in the SSC design verification documentation and retained as a record: (e.g., calculation documentation): (a) design inputs, (b) the computer-program-generated results, (c) computer-generated evidence of the programmed algorithms or equations (e.g., computer program listings, spreadsheet cell contents).

c. Those performing individual software verifications must be cognizant that software may not have undergone CGD and/or may not have been purchased from an NQA-1 supplier. Accordingly, they must apply the appropriate rigor in the SSC design review to ensure the computer program produces satisfactory results. Show that the software produces correct solutions for the encoded mathematical model within defined limits for each parameter employed. As applicable, show that the encoded mathematical model produces a valid solution to the physical problem associated with the particular application.

2. To SWAU, see Subsection 6.7.

B. When. V&V/SWAU the software each time the software is used for a SSC design, when required by the SSC design V&V process, and prior approval of the associated SSC design.

C. Who. As per the applicable SSC design V&V process.

APPENDICES, FORMS, AND REFERENCES

APPENDICES:

APPENDIX A: SOFTWARE TEST PLAN REQUIREMENTS TABLE

FORMS:

SOFT-V&V-FM01: SOFTWARE APPROVAL FOR USE FORM (SWAU)

REFERENCES:

V&V EXAMPLES

The reference collection associated with SOFT-V&V contains example V&V documentation. Examples initially provided are for informational purposes only and have not been reviewed for compliance to the requirements of this chapter. Better examples will be posted as they are shared with chapter POC. At time of writing, they included:

Example Requirements Traceability Matrix (Source Tracker)
Example No. 1 Test Plan (Source Tracker)
Example No. 1 Test Plan/Report (Source Tracker)
Example No. 2 Test Report
Example No. 3 Test Report
Example SWAU Request Package Checklist (MACCS2)

APPENDIX A: COMPUTER PROGRAM TEST PLAN REQUIREMENTS

Table 21.4-A1 provides computer program test plan requirements. The text in *italicized print* should be used as guidance for all items marked as "Gr".

Table 21.4-A1 Computer Program Test Plan Requirements

No	Requirement	Apply by ML[1]			
		1	2	3	4
1.0 General Test Plan Requirements[6]					
1.01	Include or reference the test name, objective(s) and date.	R	R	R	*
1.02	Include or reference the test configuration *(including pre-test, during testing, and post-test configurations where appropriate).*	R	R	Gr[2]	*
1.03	Satisfy test prerequisites. Prerequisites include the following as applicable: calibrated instrumentation, appropriate and/or special equipment, trained personnel, condition of test equipment and the item to be tested, suitable environmental conditions, and provisions for data acquisition. Ensure suitability and readiness of equipment and/or algorithms used for simulated computer program test inputs/fault seeding.	R	R	R	*
1.04	Assure that suitable environmental conditions are satisfied during testing	R	R	R	*
1.05	Assure adequate instrumentation is available and used.	R	R	R	*
1.06	Assure appropriate tests and equipment are used (e.g., performance of a pneumatic leak test may be required because a hydrostatic leak test could damage the item). Select measuring and test equipment based on the type, range, accuracy, and tolerance needed to accomplish the required measurements for determining conformance to specified requirements.	R	R	R	*
1.07	Provide for traceability of measuring and test equipment to the test.	R	R	R	*
1.08	Assure necessary monitoring is performed.	R	R	R	*
1.09	Specify characteristics to be tested.	R	R	R	*
1.10	Obtain test requirements and acceptance criteria *from, or approved by, the responsible design organization*.	R	R	Gr	*
1.11	Control the test. (This includes controlling changes during testing; includes as appropriate, software design verification, factory acceptance, site acceptance, and in-use tests.)	R	R	R	*
1.12	Control tests under appropriate environmental conditions using the tools and equipment necessary to conduct the test in a manner to fulfill test requirements and acceptance criteria.	R	R	R	*
1.13	Develop/specify test methods to obtain necessary data with sufficient accuracy for evaluation and acceptance.	R	R	R	*
1.14	Base test requirements and acceptance criteria upon specified requirements contained in the applicable design documents, or other pertinent technical documents that provide approved requirements. Ensure acceptance criteria are specific, unambiguous, measurable, and attainable.	R	R	R	*
1.15	Document and maintain test results.	R	R	R	*
1.16	When changes to an approved configuration management SSC of an operating facility are required for testing purposes, obtain approval by the Facility Design Authority (DA) or FDAR, and others as required in the governing work control process (e.g., AP-341-504, *Temporary Modification Control*), prior to performing the test.	R	R	R	*

No	Requirement	Apply by ML[1]			
		1	2	3	4
1.17	When identification and traceability requirements are specified in requirements documents, design the test to satisfy the requirements.	R	R	R	*
1.18	When post-installation testing is planned, mutually establish post-installation test requirements and acceptance documentation with the item purchaser and supplier.	R	R	R	*
1.19	When special process examinations/tests are performed (e.g., Nondestructive Examination (NDE) such as radiographic testing, magnetic particle testing, ultrasonic testing, liquid penetrant testing, electromagnetic testing, neutron radiographic testing, leak testing, acoustic emission testing, and visual testing), address the applicable requirements of STD-342-100, *Engineering Standards Manual,* Chapter 13, *Welding, Joining & NDE*, and P330-5, *Special Processes.*	R	R	R	*
1.20	Perform testing in accordance with the governing work control process and the test plan; address applicable health, safety, environmental and security requirements associated with the testing. Ensure post-configuration conditions meet post-configuration test plan requirements, including approvals as required.	R	R	R	*
1.21	Evaluate conformance of test results with test requirements and acceptance criteria.	R	R	R	*
1.22	Identify those responsible for accepting the test(s).	R	R	R	*
1.23	The process for obtaining a consent to waive specified hold points (testing, inspection and/or witness holds) or, make changes to the test plan, must be as described in applicable work control documents (e.g., field change, design change process, etc.) that govern configuration management for the item. If the waiver and/or change process is not addressed in the governing work control process, then address the waiver process, including obtaining consent from authorized personnel, in the test plan.	R	R	R	*
1.24	Obtain consent to waive hold points prior to continuation of work beyond the designated hold point and ensure documentation of the consent.	R	R	R	*
1.25	As applicable, develop and prepare an approach to sampling. *When used, base sampling on standard statistical methods with approval from the responsible engineer for the item. Obtain support from the responsible engineer and/or quality assurance representative in development of the plan as required.* *When used for accepting commercial grade items and services, base sampling upon standard statistical methods with supporting engineering justification. Consider lot/batch/traceability, homogeneity, and the complexity of the item.* If sampling is performed, document the sampling method *and the approval of the method by the responsible engineer*. When less than 100% sampling is used, draw samples at random.	R	R	Gr	*
1.26	*Demonstrate adequacy of performance under conditions that simulate the most adverse design conditions with operating modes and environmental conditions considered in determining the most adverse conditions.*[4]	R	R	Gr	*
1.27	*Where the test is intended to verify only specific design features, the other features of the design shall be verified by other means. Establish and verify scaling laws when tests are being performed on models or mockups.*[4]	R	Gr	Gr	*
1.28	*Perform error analysis, where applicable, on the results of model test work prior to the use of the final design.*[4]	R	Gr	Gr	*
2.0 Additional Test Plan Requirements for General Computer Program Testing[6]					
2.01	*Involve engineering, in accordance with the governing work control process, as applicable, in safety software inspection, including obtaining concurrence to the acceptability of safety software verification results.*	R	R	Gr	*

No	Requirement	Apply by ML[1]			
		1	2	3	4
2.02	*Obtain test requirements and acceptance criteria by the organization responsible for the use of the computer program.*	R	R	Gr	*
2.03	Include required tests, test sequence and frequency.	R	R	R	*
2.04	Include required ranges of input parameters.	R	R	R	*
2.05	Identify the stages at which testing is required.	R	R	R	*
2.06	*Include criteria for establishing test cases.*	R	R	Gr	*
2.07	*Include requirements for testing logic branches.*	R	R	Gr	*
2.08	Include requirements for hardware integration.	R	R	R	*
2.09	Include anticipated output values.	R	R	R	*
2.10	Include acceptance criteria, reports, records, standard formatting, and conventions.	R	R	R	*
2.11	Demonstrate the adherence of the computer program to documented requirements.	R	R	R	*
2.12	*Evaluate technical adequacy through comparison of test results from alternative methods such as hand calculations, calculations using comparable proven programs, or empirical data and information from technical literature.*[3]	R	R	Gr	*
2.13	*For software design verification testing, demonstrate the capability of the computer program(s) to provide valid results for test problems encompassing a range of documented permitted usage.*[4]	R	R	Gr	*
2.14	For those computer programs used for operational control, demonstrate required performance over the range of operation of the controlled function or process.	R	R	R	*
2.15	*For computer programs in which computer program errors, data errors, computer hardware failures, or instrument drift can affect required performance, prescribe and perform periodic in-use manual or automatic self-check tests.*[5]	R	R	Gr	*
2.16	Provide for system software testing. *For new or revised software, evaluate, review, test, and accept for use. Place system software under configuration change control. Evaluate system software changes for impact on the software product to determine the required level of retesting.*	R	R	Gr	*
3.0 Additional Test Plan Requirements for Computer Program <u>Acceptance</u> Testing[7]					
3.01	Include the process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specified requirements and to identify differences between expected and actual results in the operating environment.	R	R	R	*
3.02	Demonstrate that the computer program adequately and correctly performs all intended functions (i.e., specified software design and operating requirements).[8]	R	R	R	*
3.03	Demonstrate that the computer program properly handles abnormal conditions and events *as well as credible failures. As applicable include cybersecurity vulnerability testing.*	R	R	Gr	*
3.04	Demonstrate that the computer program does not perform adverse unintended functions.	R	R	R	*
3.05	*Demonstrate that the computer program does not degrade the system either by itself, or in combination with other functions or configuration items.*	R	R	Gr	*
3.06	Ensure that acceptance testing is performed and passed prior to approval of the computer program for use.	R	R	R	*

No	Requirement	Apply by ML[1]			
		1	2	3	4
3.07	Ensure that configuration items are identified and are under configuration change control prior to starting acceptance testing.	R	R	R	*
3.08	Ensure that acceptance testing is planned, performed, and passed *for all software design requirements.*	R	R	Gr	*
3.09	Ensure that acceptance testing ranges from a single test of all software design requirements to a series of tests performed during computer program development. (Performance of a series of tests provides assurance of correct translation between activities and proper function of individual modules).	R	R	R	*
3.10	Include a comprehensive acceptance test that is performed in the operating environment prior to use.	R	R	R	*
3.11	*Provide for documentation and disposition of observations of unexpected or unintended results prior to test approval.*	R	R	Gr	*
3.12	For acceptance testing of changes to a computer program, provide for selective retesting to detect unintended adverse effects introduced during changes to the computer program. Provide assurance that the changes have not caused unintended adverse effects in the computer program, and to verify that a modified system (s) or system component (s) still meets specified software design requirements.	R	R	R	*

Notes:

[1] R = Required; Gr = Graded by using *italicized text* as guidance and non-italicized text as required; * = As required by ESM Chapter 15, *Commissioning* and the SRLM.

[2] Not used.

[3] Applies to Non-SSC software only.

[4] Applies to design verification testing only, reference NQA-1, Part I, Requirement 3, *Design Control*.

[5] Apply as required during testing to ensure that test results are valid.

[6] From NQA-1, Part I, Requirement 11, *Test Control.* As applicable, General Test Plan Requirements apply to testing of items as graded (items that include computer programs or do not include computer programs). Additional Test Plan Requirements for General Computer Program Testing apply to formal interim computer program tests and computer program acceptance tests as graded.

[7] From NQA-1, Part II, Subpart 2.7, *Quality Assurance Requirements for Computer Software for Nuclear Facility Applications* and associated guidance. As applicable, additional Test Plan Requirements for Computer Program Acceptance Testing apply to acceptance testing of computer programs in addition to requirements under Headings 1 and 2 in this table as graded in the table.

[8] If the software is required to perform a Technical Safety Requirement (TSR) surveillance, demonstrate satisfactory execution of the surveillance as part of the acceptance test.

Conduct of Engineering

Software Approval for Use Form (SWAU) For:

SWAU No.: Rev.:

1.0 GENERAL INFORMATION					
1.1 Software Name and Revision:					
1.2 TA Number(s):		1.3 Facility Number(s):		1.4 Facility Name(s):	
1.5 SWID No.:			1.6 Equipment ID:		
1.7 Software Intended Use:					
1.8 SSC Software/Non-SSC Software:	SSC ☐ Non-SSC ☐				
1.9 Highest Management Level (ML) Associated with the Software:	ML-1 ☐ ML-2 ☐ ML-3 ☐ ML-4 ☐				
1.10 Software Designation:	Select from drop down menu				

2.0 APPROVALS		
2.1 As the Software Responsible Line Manager (SRLM), I attest that the software approval for use requirements of ESM Chapter 21, *Software* have been satisfied and that the software is ready to be used in its intended operating environment in accordance with approved operating and maintenance instructions and, any limitations as described in Section 3.		
Printed Name/Z No.:	Signature	Date

3.0 USE METHODS/LIMITATIONS
3.1 Use Methods/Limitations

4.0 REVISIONS		
4.1 Revision No.	4.2 Date	4.3 Revision Description and Reason for Revision
0	Click here to enter a date.	Initial issue

5.0 ATTACHMENTS	
5.1 Attachment No.	5.2 Attachment Title

GENERAL

This form is provided to promote the successful and consistent implementation of the software approval for use (SWAU) requirements of Engineering Standards Manual (ESM) STD-342-100, *Chapter 21, Software (Chapter 21).* The software owner (SO) is responsible for preparing this form with the approvals as required by Chapter 21.

SWAU indicates that the software requirements have been satisfied and the software is ready to be used in its intended operating environment including:

a. Required verification and validation (V&V) has been successfully completed.

b. Required use and maintenance processes are in place (e.g., personnel are trained and/or authorized; operations and maintenance (O&M) documents, configuration management, and access controls etc. are in place.).

c. As applicable, safety basis reviews have been performed (e.g., Unreviewed Safety Question (USQ) per PD110, *Safety Basis*).

These instructions are for completing the form and are not part of the completed form.

If entering sensitive information (e.g., Unclassified Controlled Nuclear Information–UCNI), ensure proper Derivative Classifier/Reviewing Official reviews are performed and appropriate markings applied.

LANL personnel: Endeavor to use Chapter forms as-is and report issues and improvement ideas to the Chapter 21 POC. POC may authorize other methods equivalent to chapter forms in writing.

LANL subcontractors must use Chapter forms to satisfy Chapter requirements for SSC software. For Non-SSC software, subs may either use their own forms or integrate, adapt, and reformat the forms; either approach is acceptable so long as key functions, data, and approvals are retained.

HEADER

Field	Entry Information
Software Name	Enter the name of the software in the upper right hand corner of the header as shown in the example form (e.g., Software SWAU for CBT Software).
SWAU No.	The numbering scheme of this form follows that of the SRLM's document control/records management system. For software used in the operation of existing or new LANL facilities, use numbering scheme and process specified in AP-341-402, *Engineering Document Management in Operating Facilities*.
Rev.	Enter the SWAU revision number. Initial issue is 0.

1.0 GENERAL INFORMATION

Field	Entry Information
1.1	Enter the name and revision number of the software.
1.2	Enter the technical area(s) where the software is used. If the software is used site wide enter "site wide".
1.3	Enter the facility number(s) where the software is used. If the software is used site wide enter "site wide".

Field	Entry Information
1.4	Enter the facility (building) name(s) where the software is used. If used site wide enter "site wide".
1.5	Enter the software identification number (SWID). The SWID should be the same SWID as on the Form 2033. Keep SWID the same through minor revs, and most major revisions unless the major revision is so different that the SRLM wishes to get a new SWID. SWIDs are obtained in accordance with AP-341-402, *Engineering Document Management in Operating Facilities.*
1.6	For SSC software, enter the equipment identification number associated with the software per ESM Chapter 1, Section 200 (e.g., 03-0216-HVAC-BAS-1). For Non-SSC software, enter NA.
1.7	Describe the intended use of the software (i.e., what the software does/how it will be used.)
1.8	Indicate whether the software is SSC Software or non-SSC software by checking the appropriate box. Check only one box.
1.9	Indicate the highest management level (ML) associated with the software. Reference AP-341-502. (ML-1 is the highest ML indication). If the software associated with multiple MLs, check the highest ML box. Check only one box.
1.10	Enter the software designation (safety software, risk significant software (RS) or commercially controlled software (CC)). Enter only one designation.

2.0 REVIEW AND APPROVAL

Field	Entry Information
2.1	The SRLM reviews the SWAU request package (See SOFT-V&V) to ensure, satisfactory completion of V&V, applicable operational and maintenance (O&M) controls per SOFT-MAINT are in place, and the software is ready to be used in its intended environment. Ensure that interfaces with software systems, organizations and/or activities that the software is based on or otherwise has an interface with, are controlled such that unintended consequences with such systems/organizations do not occur (e.g., coordinate with cybersecurity personnel per PD210, *Cybersecurity Program* or CIO-P100, *Cyber Security Risk Management Plan*.) The SRLM attests that: (a) the SWAU requirements of ESM Chapter 21, *Software* have been satisfied; and (b) the software is ready to be used in its intended operating environment in accordance with approved operating and maintenance instructions and any limitations as described in Section 3. Enter the SRLM name, Z number if applicable, organization, signature, and date. ***Note***: For assistance in determining the SRLM, see SOFT-GEN Appendix B: *SO and SRLM Decision Diagram for FAC-COE.*

3.0 LIMITATIONS

Field	Entry Information
3.1	Describe the use/maintenance processes, associated limitations, access controls, etc. for using the software. Describe either directly and/or through reference to other documents (e.g., user manual, SWDS).

4.0 REVISIONS

Field	Entry Information
4.1	If the SWAU requires revision, then revise in accordance with the governing document control process. Enter the revision number, 1, 2, etc. as appropriate.
4.2	Enter the revision approval date.
4.3	Describe the revision and why the revision was made.

5.0 ATTACHMENTS

Field	Entry Information
5.1	Attach the SWAU request package and as appropriate, SRLM request package review documentation (e.g., checklists, etc.). Enter the attachment number(s). List and include other attachments as appropriate.
5.2	Enter the attachment title.

SWAU No.:	SWAU-03-2333-12345	Rev.:	1

1.0 GENERAL INFORMATION

1.1 Software Name and Revision:	Combustible Material Tracker (CBT) Software, Revision 2.0				
1.2 TA Number(s):	03	1.3 Facility Number(s):	2332	1.4 Facility Name(s):	Nonproliferation Security Facility (NSF)
1.5 SWID	045	1.6 Equipment ID:	NA		
1.7 Software Intended Use:	The software is used as a tool to track combustible material.				
1.8 SSC Software/Non-SSC Software:	SSC ☐ Non-SSC ☒				
1.9 Assoc. Management Level (ML):	ML-1 ☐ ML-2 ☐ ML-3 ☒ ML-4 ☐				
1.10 Software Designation:	Risk Significant Software (RS)				

2.0 APPROVALS

2.1 As the Software Responsible Line Manager (SRLM), I attest that the software approval for use requirements of ESM Chapter 21, *Software* have been satisfied and that the software is ready to be used in its intended operating environment in accordance with approved operating and maintenance instructions and any limitations as described in Section 3.

Printed Name/Z No.:	Signature	Date
Pete Lopez/121987		06/17/2015

3.0 USE METHODS/LIMITATIONS

3.1 Use Methods/Limitations

1. As stated in SWUM-15-03-NSF-0010, *CBT Software User Manual* and AP-15-03-NSF-0007, *CBT Software Administrative Procedure.*
2. This SWAU is limited to employees and subcontractors working for LANL NEN Division.

4.0 REVISIONS

4.1 Revision No.	4.2 Date	4.3 Revision Description and Reason for Revision
0	07/20/2015	Original Issue, CBT Software, Revision (version) 1.0
1	06/17/2016	Upgraded software to Revision 2.0; Revision 1.0 no longer supported.

5.0 ATTACHMENTS

5.1 Attachment No.	5.2 Attachment Title
A1	CBT Software Revision 2.0 Request for Approval for Use Package, Revision 0

Sensitive Data

TABLE OF CONTENTS

1.0 PURPOSE, SCOPE AND APPLICABILITY 1
2.0 KEY DEFINITIONS AND ACRONYMS.....1
3.0 SSC SOFTWARE INVENTORY 1
4.0 NON-SSC SOFTWARE INVENTORY 3

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Added reference to Engineering Services software inventory and its instructions; clarifications.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

1.0 PURPOSE, SCOPE AND APPLICABILITY

This section describes the "how, when, and who" for inventorying software. See Table 21.5-1 for a summary of this section. See Chapter 21 applicability in SOFT-GEN.

2.0 KEY DEFINITIONS AND ACRONYMS

See SOFT-GEN for key definitions and acronyms.

3.0 SSC SOFTWARE INVENTORY

A. How. Inventory **safety** SSC software in the LANL Institutional-level safety software inventory in accordance with QPA-DO-FSD-001, *LANL Safety Software Inventory Procedure.* **Non-safety** SSC software is not required to be inventoried.

Note: See ESM Chapter 21, SOFT-GEN and Form 2033 to determine if the software is safety software.

1. Provide required information, namely:

   a. Software name,

   b. Description,

   c. Version identifier,

   d. Category,

   e. Software Risk Level (SRL),

   f. Application,

   g. Software Responsible Line Manager (SRLM), and

   h. Other information as required by QPA-DO-FSD-001.

5 Section SOFT-INV: Inventory

Rev. 1, 05/25/17

Table 21.5-1 SOFT-INV Section Summary
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	Software Type[1]		Implementation Detail			ESM Ch. 21 Ref.[5]
		Safety	Non-Safety	How	When[2]	Who[3, 4]	
1. For SSC Software							
1	Inventory **safety** software	R	-	▪ Provide safety software inventory information ▪ Inventory safety software Per QPA-DO-FSD-001, *LANL Safety Software Inventory Procedure*	▪ Provide SSC software inventory information per ESM Chapter 1, Z10 submittal schedule ▪ Inventory as soon as practical after approval for use ▪ Revise as needed to ensure the inventory is complete and accurate	▪ Design Agency provides safety SSC software inventory information in PEL for designed and acquired software; see SOFT-GEN for Subcontractor responsibilities (D) ▪ SRLM inventories safety software (D)	SOFT-GEN, 3 SOFT-INV, 3
1. For Non-SSC Software							
1	Inventory **safety and non-safety** software	R	R	▪ Inventory safety software in the Institutional-level safety software inventory per QPA-DO-FSD-001, *LANL Safety Software Inventory Procedure* ▪ ES-Div: Inventory safety and non-safety software in the ES Non-SSC software inventory per ESDO-AP-002.	▪ Inventory as soon as practical after approval for use ▪ Revise as needed to ensure the inventory is complete and accurate	▪ Design Agency provides safety SSC software inventory information in PEL for designed and acquired software; see SOFT-GEN for Subcontractor responsibilities (D) ▪ SRLM inventories software (D) ▪ ESSC supports ES Non-SSC software inventory additions/changes (S)	SOFT-GEN, 3 SOFT-INV, 4

Notes:

[1] See SOFT-GEN and Form 2033 to determine if software is safety software or non-safety software. R = Required; "-" = Not required

[2] "When" specifies when the activity must be performed

[3] D = Develop or execute. S = Support

[4] SRLM = Software Responsible Line Manager. ESSC = Engineering Services (ES) Software Coordinator as indicated on the ES Division Office (ES-DO) website. See SOFT-GEN (e.g., flow diagram) for assistance in identifying the SRLM

[5] Ref. = ESM Chapter 21 section name and subsection number (e.g., Section SOFT-GEN, Subsection 3, Software Identification and Determination)

Note: Do not enter classified or controlled unclassified information (CUI) into the inventory.

2. Provide changes to the inventory to ensure completeness and accuracy.

3. Subcontractors that provide SSC software to LANL must provide minimum required SSC Safety Software inventory information as part of the Project Equipment List (PEL) or via other submittal per ESM Chapter 1, Section Z10.

4. If LANL has access to and/or configuration control of SSC safety software[1], then it must be inventoried on the Institutional-level safety software inventory.

5. SSC safety software may be inventoried as a software collection. For example, there may be various software elements – control system executable, Human Machine Interface (HMI) software, field element embedded software – associated with a gas handling process control system software. The software may be inventoried as one item, the gas handling process control system software. Note that the various software elements must be baselined and under configuration management per SOFT-GEN.

B. When. Inventory SSC safety software as soon as practical after it is approved for use and as needed to ensure the inventory is complete and accurate. Provide inventory information in accordance with the PEL or other submittals schedule per ESM Chapter 1, Section Z10.

C. Who. Design Agency provides safety SSC software inventory information in PEL for designed and acquired software; see SOFT-GEN for Subcontractor responsibilities. The SRLM[2] is responsible for inventorying safety SSC software.

4.0 NON-SSC SOFTWARE INVENTORY

A. How.

1. Inventory **safety** Non-SSC software in the Institutional-level safety software inventory in accordance with QPA-DO-FSD-001.

   a. Subcontractors: Safety Non-SSC software used by a subcontractor and prescribed or otherwise controlled by LANL or DOE (e.g., toolbox codes) must be inventoried. The Subcontractor shall provide the minimum required safety software information to inventory such software in the institutional-level safety software inventory.

2. ES-Div: Inventory **safety and non-safety** Non-SSC software in the Engineering Services (ES) Non-SSC software inventory in accordance with ESDO-AP-002, Engineering Services (ES) Software Inventory Instructions. *For assistance, contact the ES Software Coordinator (ESSC) at ES-Software@lanl.gov.*

Note: As described in SOFT-MAINT, the software should not be used unless it is listed on the ES inventory or otherwise approved for use (SWAU). The listing of the software on the ES inventory is the primary method to communicate SWAU.

[1] Includes LANL service contracts where suppliers may upgrade/flash new software on SSCs.
[2] The activity may be delegated however the responsibility is retained by the SRLM.

B. When. Inventory Non-SSC software (both safety and, for ES-Div, non-safety) concurrent with SWAU and as needed thereafter to ensure completeness and accuracy.

C. Who. The SRLM is responsible for inventorying Non-SSC software.

TABLE OF CONTENTS

1.0 PURPOSE AND APPLICABILITY....1
2.0 KEY DEFINITIONS AND ACRONYMS....1
3.0 SSC SOFTWARE....1
3.1 Use and Maintain SSC Software....1
3.2 Retire SSC Software....6
4.0 NON-SSC SOFTWARE....7
4.1. Use and Maintain Non-SSC Software....7
4.2. Retire Non-SSC Software....7

Rev	Date	Description	POC	RM
0	06/23/16	Initial issue	Tobin Oruch, *ES-DO*	Mel Burnett, *CENG-OFF*
1	05/25/17	Clarifications for existing (in-use) software, other minor clarifications.	Tobin Oruch, *ES-DO*	Lawrence Goen, *ES-DO*

1.0 PURPOSE AND APPLICABILITY

This section describes the "how, when, and who" for using, maintaining and retiring software.

See Table 21.6-1 for a summary of this section (for both SSC software and Non-SSC software). Applicability is per Chapter 21 applicability in SOFT-GEN.

2.0 KEY DEFINITIONS AND ACRONYMS

See SOFT-GEN for key definitions and acronyms.

3.0 SSC SOFTWARE

3.1 Use and Maintain SSC Software

A. <u>How</u>.

1. Control the software to ensure that the use remains consistent with the software design and approved acceptance test results. For ML-1 through ML-3 software in addition to the applicable controls from other chapter sections[1], ensure the following controls are established; and documented on the SDWS and/or in this chapter prior to software approval for use (SWAU). Use and maintain the software using these controls until the software is retired.

   a. Application documentation,

   b. Access control specifications,

[1] Includes identification and documentation of software in a software summary (ref. SOFT-GEN). For existing software (also referred to as in-use or legacy software) that is non-compliant, initiate a PFITS action to resolve deficiencies (see SOFT-GEN *Article 1.0* for details).

c. Computer system vulnerability protections,

d. Problem reporting and corrective action,

e. In-use tests,

f. Configuration management,

g. Inventory, and

h. Supporting quality assurance program controls.

2. Process and retain documentation in accordance with the SRLM's governing document control and records management process and/or as described in the SWDS. Ensure the correct SWID (obtained when completing Form 2033) is part of the documentation record number. See AP-341-402, *Engineering Document Management in Operating Facilities for details.*

3. Use the following to implement the aforementioned controls. See IEEE Std 12207, *Systems and Software Engineering – Software Life Cycle Processes* and ANSI/ANS-10.8, *Non-Real-Time, High-Integrity Software for the Nuclear Industry – User Requirements* for additional guidance.

a. **Software Application Documentation.** Application documentation for SSC software is a general term comprised of the following main categories.

1) **Operation and Maintenance (O&M) Documentation.** Software shall be controlled in accordance with approved procedures and instructions (i.e., O&M instructions). They are required for all ML levels. The O&M documentation must be sufficiently detailed and usable to allow a competent individual trained in the use of the software to use the program without undue difficulty and/or likely misuse. *O&M manual(s) should be listed on the SWBL.*

2) **Licenses and Registrations**.[2] As applicable, maintain license documentation and other registrations, including maintenance contracts, as appropriate to promote proper software use. As applicable, document licenses on the SWBL. As applicable, document license and registration maintenance protocols on the SWDS.

3) ***Operational Event Documentation****. Operational event documentation should be retained for ML-1 and ML-2 software to support software maintenance, cybersecurity protection and assessments. Operational event documentation is generated as part of the software operation. It is dependent on the software design. Event documentation may include system startup, shutdown, changes, logons, logon failures, logoffs, self-diagnostic test results, tamper attempts, and/or other information. As applicable, indicate event documentation review and retention protocols on the SWDS.*

[2] Licenses and registrations are used to ensure the validity of the software, prevent suspect/counterfeit items (S/CI), and to maintain knowledge of software errors, issues, updates, etc.

Table 21.6-1 SOFT-MAINT Summary
(This table is a summary only and does not include all requirement details. See text for details.)

Activity No.	SQM Activity	ML[1]			Implementation Detail			ESM Ch. 21 Ref.[5]
		1, 2	3	4	How	When	Who[2]	
1	Use and maintain the software	R	Gr	Gr[3]	▪ Use and control the software to ensure that the use remains consistent with the software design and approved acceptance test results ▪ Maintain software application documentation ▪ Licenses and registrations ▪ Use and maintain access controls ▪ Use and maintain vulnerability protections ▪ Report problems and take corrective actions ▪ Perform in-use tests ▪ Maintain configuration management (e.g., maintain software baseline, Software Data Sheet [SWDS] and software change documentation) ▪ Maintain inventory(ies)[4] ▪ Implement supporting quality assurance program controls (e.g., training/qualification and assessments)	▪ Upon software approval for use (SWAU) and until the software is retired ▪ In-use testing and assessments as specified on the SWDS (at a min., when installed on a different computer or after significant changes)	▪ SU (for use) ▪ SRLM (for maintenance)	SOFT-GEN SOFT-V&V SOFT-INV SOFT-MAINT
2	Retire the software	R	R	R	▪ Prevent from unintended or routine use ▪ Terminate software agreements and/or licenses ▪ Remove from the inventory(ies) as applicable[4] ▪ Where practical and/or where a future need may exist, retain as a record	▪ When no longer required for routine use or (AFU)	▪ SRLM	SOFT-INV SOFT-MAINT

[1] ML = Management Level; R = Required; Gr = Required but graded.
[2] SU = Software User; SRLM = Software Responsible Line Manager. The SU and SRLM may be the same person.
[3] In-use testing is not required for ML-4.
[4] See SOFT-INV for details.
[5] Ref. = ESM Chapter 21 section name.

4) ***Application Logs**. Application logs are lists (or logs) that provide information that describes who, what baseline revision, when and where (e.g., facility) the software was used. Application logs should be maintained for ML-1 and ML-2, Non-SSC software when such traceability above that provided by the software inventory is required. Application logs may be retained as part of the ES-Div software inventory (possible future capability) or through other means. As applicable, provide application log protocols on the SWDS.*

b. **Access Control Requirements.** For all ML levels, establish and implement access controls in accordance with P218, *Cyber Security Access Controls*. Address the security of the computer system and the critical data that may reside on the system. For ML-1 through ML-3 software, document access controls on the SWDS and as required, in the governing software procedures. See SOFT-GEN for SWDS details.

c. **Computer System Vulnerability Protections.** For all ML levels, establish and implement computer system vulnerability protections as required by PD210, *Cybersecurity Program* and CIO-P100, *Cyber Security Risk Management Plan.* Base controls on consequence of loss, confidentiality, integrity, and availability. As required, contact an Information System Security Officer (ISSO) to assist with identifying and implementing computer system vulnerability controls. As appropriate, implement the following protections that may be designed into the program, established as part of the O&M procedures, and/or as specified on the SWDS:

1) Access timeouts after a specified time of inactivity,

2) Recovery/contingency protocols, including retention of backups of system data. (The SWBL serves as the backup for the software – computer programs and documentation needed to run the software), and

3) Software tools to verify pedigree authenticity and/or detect tampering associated with suspect/counterfeit items (S/CI).

d. **Problem Reporting and Corrective Action.** For all ML levels, follow the problem reporting and corrective action processes in SOFT-GEN.

e. **In-Use Tests.** Perform in-use testing of ML-1 through ML-3 software as specified in this section and the SWDS. In-use testing is not required for ML-4 software unless otherwise specified by the SRLM.

1) Perform in accordance with procedures to confirm acceptable performance in the operating environment. Integrate governing SSC work controls.

2) Perform in accordance with the applicable test planning and testing requirements of the SOFT-V&V section.

3) Demonstrate required performance over the range of operation of the controlled function or process.

4) Document test procedures and frequency on the SWDS. Specify test frequency using the following:

i. Perform testing after the computer program is installed on a different computer.

ii. Perform testing when there are significant changes, as determined by the SRLM, in the operating system/environment

iii. Perform periodic testing where computer program errors, data errors, computer hardware failures, or instrument drift can affect require performance (consider manual and/or automatic self-check tests).

iv. Perform testing when required by procedure (e.g., AP-341-801, Post Modification/Post Maintenance Testing)

v. Perform testing based on the risk of failure of the computer program. For ML-1 and ML-2 software, at a minimum, an annual test is recommended.

f. **Configuration Management.** Perform the following:

1) Follow configuration management and interface control processes in SOFT-GEN. Maintain completeness and accuracy of the software baseline and SWDS.

2) Where practical, use and maintain software versions that are compatible with prior versions of the software and associated hardware (sometimes referred to as backward compatibility) to minimize interface issues.

3) As required by the SRLM, ensure data produced by the software is adequately retained (backed up) so that it may be readily retrieved if needed.

g. **Inventory.** For all ML levels, maintain complete and accurate software inventories in accordance with the SOFT-INV.

h. **Other Quality Assurance (QA) Controls.**

1) Implement other quality assurance (QA) controls in accordance with the governing quality assurance program (e.g., training, assessments, etc.). These other QA controls must be used in conjunction with the software-specific controls of ESM Chapter 21 to provide a comprehensive software QA program. Selected LANL documents that provide controls for the ESM software QA program are listed below for reference:

i. SD330, *LANL Quality Assurance Program* (e.g., P330-3, and P330-6, P330-8, P840-1), and P805, *Export Control*;

ii. P341, *Facility Engineering Processes Manual* (e.g., AP-341-402, AP-341-502, AP-341-507, AP-341-510, AP-341-605, AP-341-504, AP-341-611, AP-341-613, AP-341-703, AP-341-801, AP-341-802, AP-341-901);

iii. P342, *Engineering Standards* (e.g., ESM Chapters 1, 7, 8, 12, and 15); and,

iv. P343, *Facility Engineering Training and Qualification Manual*.

2) As applicable, document software-specific training/qualification requirements on the SWDS and complete training/qualification to promote proper use of the software.

3) Document software assessment methods and frequency on the SWDS. Various engineering administrative procedures (e.g., AP-341-510, *Field Walk-down, Data Gathering, and Inspections*) or LANL processes (e.g., P330-3, *Quality Audits*) may be used to perform software assessments.

B. When. For ML-4 software, use and maintain software after installation and testing in accordance with the SDWS and this chapter. For ML-1 through ML-3 software, use and maintain after SWAU. For all ML levels, use and maintain until the software is retired. For ML-1 through ML-3 software, perform in-use testing and assessments as specified on the SWDS. At a minimum, perform in-use tests when installed on a different computer or after significant changes.

C. Who. The Software Users (SUs) are responsible for using the software within the software limitations and in accordance with the software SWDS/O&M requirements. The SRLM is responsible for establishing the use and maintenance requirements and maintaining the software.

3.2 Retire SSC Software

A. How. Retire software and prevent it from unintended or routine use. Perform the following, as applicable, to retire all ML levels of software:

1. Terminate software support including registrations, agreements and/or licenses.

2. Remove the software.

3. Remove safety software information from the Institutional-level Safety software inventory per QPA-DO-FSD-001, LANL *Safety Software Inventory Procedure*.

4. Cancel or revise affected documents per the governing document control process. *Guidance: Also, close all open error and software change requests, regardless of their state.*

5. Where practical and/or where a future need for the software may exist, retain the retired software baseline (which includes the computer program files) as a record. Retain in accordance with the governing quality assurance records management procedure. Retired software should be retained such that it is retrievable and can be readily installed in the event it is necessary to “rollback” to the software. Retain support software as required to rollback.

6. Cancel or revise affected training courses/plans.

B. When. Retire software when the software is no longer required for routine use and/or when it is no longer approved for use.

C. Who. The SRLM is responsible for retiring software.

4.0 NON-SSC SOFTWARE

4.1. Use and Maintain Non-SSC Software

Follow Subsection 3.1 with the following clarifications:

A. ES-Div: Before using, verify that the software, including correct version, is listed as approved for use (SWAU) on the ES Division Non-SSC Software Inventory on the ES-Div Website. If not listed, contact the SRLM or Engineering Services Software Coordinator (ESSC). Do not use LANL Non-SSC software unless it is listed as SWAU on the ES-Div Non-SSC software inventory website or SWAU is otherwise provided by the SRLM in writing.

B. ES-Div: Before using, the SU must verify that he/she is authorized to use the software by reviewing the ES-Div software inventory website. If the SU is not shown as an authorized user for the software, contact the SRLM or ESSC. Do not use the software unless authorization is shown on the website or SWAU is otherwise provided by the SRLM in writing.

C. SSC-related administrative procedures/processes do not apply.

D. Use the Non-SSC change process in SOFT-GEN (instead of AP-341-507).

E. For “Simple and Easily Understood” software, document the determination of “Simple and Easily Understood” software in the calculation where the software was used and review and approve the software (individually verify) with each use as part of the calculation per SOFT-V&V.

4.2. Retire Non-SSC Software

Follow Subsection 3.2 with the following addition:

A. ES-Div: Remove safety and non-safety software from the ES Division Non-SSC Software Inventory in accordance with ESDO-AP-002, *Engineering Service (ES) Software Inventory Instructions*.