1. STOMP/eSTOMP Software Quality Assurance Plan, Rev 2.2 2
1.1 1. SQAP Purpose 3
1.1.1 1.1 Documentation & References 4
1.1.2 1.2 Roles and Responsibilities 4
1.2 2. STOMP/eSTOMP Descriptions 4
1.2.1 2.1 Operational Modes 5
1.2.2 2.2 Software Background 6
1.3 3. Software Grading 6
1.4 4. Requirements for STOMP and eSTOMP Use by Other Projects 7
1.5 5. Software Quality Assurance 7
1.6 6. Software Project Management & Quality Planning 9
1.7 7. Software Risk Management 11
1.8 8. Software Safety 11
1.9 9. Software Requirements Identification & Management 12
1.10 10. Design 12
1.11 11. Software Configuration Management 13
1.12 12. Problem Reporting & Corrective Action 13
1.13 13. Verification & Validation 14
1.14 14. Training of Safety Software 15
1.15 15. Reviews & Assessments 16
1.15.1 15.1 Independent Assessments 16
1.15.2 15.2 Management Assessments 16
1.15.3 15.3 Software Document Review 16
1.16 16. Release Plan 16
1.17 17. Records 16
1.18 18. Procurement and Supplier Management 17
1.19 19. Operation, Maintenance and Retirement 17
1.20 20. SQAP References 17
1.21 APPENDIX A - Staff Roles 18
1.22 APPENDIX B - Document Review Record (DRR) 19
1.23 APPENDIX C - Software Acceptance Review 19

STOMP/eSTOMP Software Quality Assurance Plan, Rev 2.2

Michael Fayer

Vicky Freedman

May 2016

PNNL-24118

Prepared for
the U.S. Department of Energy
under Contract DE-AC06-76RL0 1830

Pacific Northwest National Laboratory
Richland, Washington 99352
This document maintained electronically at

https://spcollab.pnnl.gov/sites/stompshare/Shared%20Documents/Forms/AllItems.aspx

Approvals

This document is electronically approved and signed via the PNNL Electronic Records & Information Capture Architecture (ERICA) system.

Revision History

Revision	Date	Summary of Changes
1.0	December 11, 2012	Original
2.0	February 2015	Updated to add eSTOMP, update HDI references,...
2.1	July 2015	Minor changes: • Added ASME to *Abbreviations* section • Section 1: Changed from NQA-1-2000 reference to NQA-1-2008/NQA-1a-2009 and Parts I & II • Section 6 *System Software*: Clarified the external libraries used by STOMP/eSTOMP as *system software* • Section 9: Added clarification of *security* requirements. • Section 10.3: Added reference to *system software* (section 6) as well as V&V reference • Section 18: Added clarification and reference to Commercial Grade Dedication
2.2	May 2016	Added clarification on testing

Abbreviations

ASME	American Society of Mechanical Engineers

EED	Energy and Environment Directorate
HDI	How do I?...
OTD	Office of Technology Development
PL	Product Line
PMOD	Project Management Office Director
PMP	Project Management Plan
RIDS	Record Inventory and Disposition Schedule
SCM	Software Configuration Manager
SCMP	Software Configuration Management Plan
SCR	Software Change Request
SDD	Software Design Document
SQA	Software Quality Assurance
SQAP	Software Quality Assurance Plan
SQP	Software Quality Practitioner
SRD	Software Requirements Document
SSC	Safety-related Structure, Systems and Components
STOMP	Subsurface Transport Over Multiple Phases
STOMP*share*	STOMP SharePoint site
STP	Software Test Plan
TGM	Technical Group Manager
ZERT	Zero Emissions Research and Technology

1. SQAP Purpose

The purpose of this Software Quality Assurance Plan (SQAP) is to define and control the software quality assurance (SQA) activities that will be followed during the course of continued software development, deployment and maintenance of Subsurface Transport Over Multiple Phases (STOMP) software application and its scalable implementation (eSTOMP). The plan defines the SQA activities, identifies the documentation that will be created and maintained during the entire software engineering process, and covers all software development and processes throughout the applicable life cycles of the software.

The STOMP/eSTOMP software is a versatile engineering simulator that is used by many projects to examine issues regarding multiphase flow and transport through porous media. Because the software is used by many projects, its software life cycle is significantly longer than the duration of any single project that could contribute to its maintenance or development. This SQAP was developed specifically to address the maintenance and future development of the software apart from any single project that funds such work. Any project that implements, maintains and/or develops the STOMP/eSTOMP capability may reference this plan in that project's controlling documents, rather than addressing the software on a project by project basis. This approach will assure continuity of software configuration control throughout the remainder of the STOMP/eSTOMP software life cycle.

The goal of the plan is to maintain specific operational modes of STOMP and eSTOMP as qualified Safety Software, Level C, per the DOE Order (O) 414.1-D *Quality Assurance* definition for safety software and American Society of Mechanical Engineers (ASME) *Quality Assurance Requirements for Nuclear Facility Applications* NQA-1-2008 w/NQA-1a-2009 addenda, Part I requirements applicable to software (Requirements: 3 *Design Control*, 7 *Control of Purchased Items and Services*, and 11 *Test Control*) and Part II, Subpart 2.7 *Quality Assurance Requirements for Computer Software for Nuclear Facility*

Applications. This specifically refers to how the software is managed, and how it adheres to documentation requirements. Because of the extensive number of options available in the STOMP and eSTOMP codes, only a subset of all options are tested. The software User Guide (available through stomp.pnnl.gov) and the Software Requirements Document (available through the STOMP/eSTOMP Sharepoint https://spcollab.pnnl.gov/sites/stompshare/Pages/Home.aspx) document the options that have been tested under the quality program.

- Safety software is required to follow How do I?...(HDI) Exhibit: *Safety Software Requirements* and *Develop Software for Delivery* workflow.
- For those projects that plan on utilizing STOMP/eSTOMP in a *non-safety related* manner, please refer to the *Grading Software* section for more guidance.

1.1 Documentation & References

- ASME *Quality Assurance Requirements for Nuclear Facility Applications* NQA-1-2008 w/NQA-1a-2009 addenda
- DOE Order 414.1D, *Quality Assurance*
- PNNL Software Quality Assistant (PNNL internal site)
- PNNL STOMP website (http://stomp.pnnl.gov)
- PNNL SharePoint Site **STOMP*share*** (https://spcollab.pnnl.gov/sites/stompshare/Pages/Home.aspx)
- STOMP Theory, User, and Application Guides
- STOMP/eSTOMP Software Quality Assurance Plan (this document) (SQAP)
- STOMP and eSTOMP Software Requirements Documents (SRDs)
- STOMP and eSTOMP Software Design Documents (SDDs)
- STOMP and eSTOMP Software Configuration Management Plan (SCMPs)
- STOMP and eSTOMP Software Test Plans (STPs)
- STOMP and eSTOMP Software Test Results (STRs)

1.2 Roles and Responsibilities

The Laboratory's standard roles, responsibilities, accountabilities, and authorities (R2A2s) for project managers and staff apply to this team. Table 1 defines the responsibilities for the key roles. APPENDIX A - Staff Roles identifies the staff members assigned to each role. The assignment of roles will be reviewed and updated annually.

Table 1- Roles and Responsibilities for STOMP Quality Assurance Activities

Role	Responsibilities
Software Configuration Manager (SCM)	Decision authority to approve software changes; software preservation (backup); software configuration management and record keeping; assignment of training requirements; identification of risks/hazard exposure; assignment of commitments and performance of software development and testing staff; investigation of the impact of errors; communication of errors and impacts to the affected Project Manager(s) and Product Line Manager(s); communication of software changes and status
Software Developers	Software source code development
Technical Group Manager	Project management; assurance (along with PMOD) that work associated with products, solutions, services, and processes are conducted in a manner that protects the overall quality of deliverables, integrity of the Laboratory, repeatability of the results, and customer satisfaction with the product.
Software Testers	Conduct STOMP and eSTOMP software testing
Software Quality Practitioner with R&D Safety Software and High Assurance endorsement (SQP-RHE)	Guidance and oversight of SQA activities

2. STOMP/eSTOMP Descriptions

The purpose and intended use of the STOMP and eSTOMP software is to produce numerical predictions of subsurface flow and transport phenomena in variably saturated subsurface environments. This may include sites that are contaminated with volatile or nonvolatile organic compounds or radionuclides, or sites that require quantifying the permanent storage capacity, leakage rates, and public risks associated with geologic sequestration of CO2.

STOMP is custom-developed software that has been under development and in use by PNNL-since 1993. The eSTOMP software has been in development since 2009. The software is versatile, and has been used by numerous projects inside PNNL as well as by external agencies. Short courses have been offered in North America and Europe for users of STOMP outside PNNL. The STOMP software is not a single program. Instead, it is comprised of several "operational modes" that are constructed by connecting specific code modules available in the STOMP source repository to couple specific governing equations. For example, if single-phase water flow will be simulated, then STOMP operational mode 1 (STOMP-W) is the appropriate operational mode and is constructed with the source code necessary to solve the partial differential equations that govern the conservation of water mass and dilute species transport. If two-phase flow with energy transport in both aqueous and gas phases is to be simulated, then STOMP operational mode -WAE is the appropriate operational mode and is constructed with the source code necessary to solve the partial differential equations that govern the conservation of water mass, air mass, and dilute species transport. The STOMP-CO2 and STOMP-CO2e simulators solve conservation equations for component mass (i.e., water, CO2, and salt) and energy. Other operational modes cover coupling conservation equations for salt mass, energy, ice, oil, etc.

eSTOMP is based on STOMP. eSTOMP differs from eSTOMP inasmuch that it has incorporated global arrays to make the code scalable so that it can run on multiple processor cores. To date, only three modes of STOMP have been made scalable:

1. eSTOMP-W-(R)
2. eSTOMP-WAE
3. eSTOMP-CO2e.

These eSTOMP operational modes solve the same partial differential equations as STOMP, but use Global Arrays to execute the software on multiple processor cores.

2.1 Operational Modes

The STOMP simulator has been written with a variable source code that allows the user to choose the solved governing equations (e.g., water mass, air mass, dissolved-oil mass, oil mass, salt mass, thermal energy, CO2 gas). Depending on the chosen operational mode, the governing transport equations will be written over one to five phases (e.g., aqueous phase, gas phase, (nonaqueous phase liquid) NAPL phase, ice phase, solid phase). Solute transport, radioactive decay, and chemical reactions are solved using a direct solution technique (e.g., Patankar's power-law formulation, (total variation diminishing) TVD scheme) following the solution of the coupled flow equations. Input is directed through semi-formatted text files and output is available through a variety of user-directed formats. The simulator recognizes a number of boundary condition types and allows their specification both internally and externally to the computational domain.

The decision to use modes arose from the desire to improve efficiency by minimizing overall code size and operational overhead. One example of an operational mode is STOMP –W, which simulates single-phase water flow and transport using a serial solution technique. The STOMP-W mode is assembled using the source code modules necessary to solve the partial differential equations that govern the conservation of water mass and dilute species transport.

Another example is eSTOMP-W-R, which simulates single-phase water flow and reactive transport using a parallel solution technique. The eSTOMP-W-R mode is assembled using the source code modules necessary to solve the partial differential equations that govern the conservation of water mass, species transport, and reaction chemistry.

The operational modes of the serial version that follow this SQAP are:

- STOMP-W (water; serial solution)
- STOMP-W-R (water-reactive transport using Eckechem; serial solution)
- STOMP-WAE (water-air-energy; serial solution)
- STOMP-WAE-B (water-air-energy-surface barrier; serial solution)
- STOMP-WO (water-oil; serial solution)
- STOMP-WOA (water-oil-air; serial solution)
- STOMP-CO2 (CO2; serial solution)
- STOMP-CO2e (CO2-energy; serial solution)

The operational modes of the parallel version that follow this SQP are:

- eSTOMP-W (water; parallel solution)
- eSTOMP-W-R (water-reactive transport with Eckechem; parallel solution)
- eSTOMP-WAE (water-air-energy; parallel solution)
- eSTOMP-CO2e (CO2-energy; parallel solution)

More information on the serial and parallel versions of STOMP can be found in the SCMP.

2.2 Software Background

PNNL began development of STOMP in 1993 in the framework of the VOC-Arid Soils Integration Demonstration Program (Arid-ID) and funded by the U.S. Department of Energy (DOE), Office of Technology Development (OTD). STOMP was designed to be a general-purpose tool for simulating subsurface flow and transport. The simulator was specifically designed to provide scientists and engineers from varied disciplines with multidimensional analysis capabilities for modeling subsurface flow and transport phenomena. STOMP's target capabilities were guided by proposed or applied remediation activities at sites contaminated with volatile organic compounds and/or radioactive material. Developed with the support of the U.S. DOE and Office of Environmental Restoration and Waste Management, the simulator's modeling capabilities address a variety of subsurface environments, including nonisothermal conditions, fractured media, multiple-phase systems, nonwetting fluid entrapment, soil freezing conditions, nonaqueous phase liquids, reactive transport, radioactive decay, solute transport, dense brines, nonequilibrium dissolution, and surfactant-enhanced dissolution and mobilization of organics.

STOMP-CO2 and STOMP-CO2e began development in 2001 in response to the U.S. DOE's request for the development of numerical simulation capabilities for quantifying permanent storage capacity for CO2, leakage rates, and public risks associated with geologic sequestration of CO2. In conjunction with this request the Zero Emissions Research and Technology (ZERT) Center has been created with the mission of conducting basic and applied research that supports the development of new technologies for minimizing emissions of anthropogenic carbon dioxide and other greenhouse gases that impact global climate change. As a member of the ZERT Center, PNNL is conducting research associated with geologic sequestration of CO2 that includes the thermochemistry of supercritical CO2-brine mixtures, mineralization kinetics, leakage and microseepage of CO2, and new materials for CO2 capture.

eSTOMP modes began development in 2009 in response to a need for faster execution times when applying multiphase flow and multicomponent reactive transport to two- or three-dimensional problem domains. Supercomputers provide the the computing power and storage facilities needed to execute large and complex simulations in the subsurface. eSTOMP takes advantage of this architecture to to leverage high-performance computation.

Both the STOMP and eSTOMP simulators solve the partial-differential equations that describe the conservation of mass or energy quantities by employing integrated-volume finite-difference discretization to the physical domain and backward Euler discretization to the time domain. Except for certain geometries, boundary conditions, and simplified property functions, there are no analytic solutions to these equations, leaving numerical solutions as the only option for general problems. To solve these conservation equations numerically they are first converted to algebraic form using the integral finite difference approach applied to structured orthogonal grids and Euler-backward time differences. The resulting equations are nonlinear coupled algebraic equations, which are solved using Newton-Raphson iteration.

Both STOMP and eSTOMP are written in FORTRAN. STOMP has been executed on a variety of platforms at national laboratories, government agencies, private companies, and universities. Full optimization of the simulator has been successful on Convex, Cray, Hewlett Packard, IBM, Silicon Graphics, and Sun workstations and mainframe computers. The theoretical and numerical approaches applied in the simulator have been documented in a published theory guide. Application and use of the simulator have been documented in two guide manuals. Battelle maintains the copyright to the coding and intellectual property associated with STOMP. eSTOMP is currently being prepared for its first public release in 2015.

3. Software Grading

The software being developed for STOMP and eSTOMP meets the safety software definitions defined in the PNNL HDI Exhib it: Safety Software Classifications, and has been graded as Safety Software, Level C in HDI Exhibit: Safety Software Grading Levels. The following grading information has also been documented in the PNNL software inventory tool.

- **Software Product:** The purpose and intended use of the STOMP software is to produce numerical predictions of subsurface flow and transport phenomena in variably saturated subsurface environments, which are contaminated with volatile or nonvolatile organic compounds or radionuclides including radioactive chain decay processes.
- **Software Classification:** System Management and Administrative Controls Software. Software that performs a hazard control function in support of nuclear facility or radiological safety management programs or technical safety requirements or other software that performs a control function necessary to provide adequate protection from nuclear facility or radiological hazards. This software supports eliminating, limiting, or mitigating nuclear hazards to workers, the public, or the environment as addressed in 10 CFR 830, 10 CFR 835, and the DEAR ISMS clause.
- **PNNL Clarification:** This is software that is typically used to classify, design, or analyze a DOE nuclear or radiological facility or site involving radiological hazards. This software helps to confirm the proper accident or hazards analysis of a facility. In this definition, the term facility has a broader definition than some may assume; it

can refer to an entire DOE site, large operating areas within a site, individual buildings or laboratories, waste storage areas, and other types of locations that may contain radiological materials. This type of safety software is defined as Safety and Hazard Analysis Software and Design Software. (Examples include GENII, MCNP, etc.)

The rationale for identifying STOMP a priori both as Safety Software and as Safety and Hazard Analysis (Level C grading) software is that it has been and is expected to continue to be used for prediction of migration of radiological contaminants at DOE owned facilities. This makes the Safety Software Subject Area applicable for such applications. However, the official grading level will be determined by the project applying the software. Please see section Grading Software for Project section (below) for more guidance.

- **Software Type:** Custom developed
- **Grade:** Level C

Because STOMP and eSTOMP comprise multiple operational modes with both shared and unique code, qualification will be performed on an operational mode basis. The qualified operational modes will be distinguishable from those that are not yet qualified.

This grading level includes software applications that do not meet Level B criteria but meet one or more of the following criteria.

- Software failure that could cause a potential violation of regulatory permitting requirements.
- Software failure that could affect environment, safety, health monitoring or alarming systems.
- Software failure that could affect the safe operation of a Safety-related Structure, Systems and Components (SSC).

3.1 Grading Software for Projects

Each project is required to grade their use/development of the STOMP/eSTOMP software. If the use/development of the software is:

- determined to be at a safety software level C, this plan must be followed.
- determined to be at a higher safety software level (i.e., A or B), then it is the project's responsibility to meet the additional Lab software requirements for the applicable grade. This includes, the project engaging an SQP-RH for guidance.
- determined to *not* be safety or high assurance software, this plan is not required. At a minimum, the project must follow either the HDI *Use or Develop Software for Analysis* or *Develop Software for Delivery* workflows.

4. Requirements for STOMP and eSTOMP Use by Other Projects

This SQAP is applicable for the remainder of both the STOMP and eSTOMP software life cycles. PNNL supports the STOMP/eSTOMP maintenance and qualification as safety software to provide the software as a general resource to multiple projects. Any project that implements, maintains, and/or develops STOMP or eSTOMP may reference this plan in that project's controlling documents, rather than creating a completely new set of software QA documentation specific to the project.

While both STOMP and eSTOMP have been generically classified as graded as System Management and Administrative Controls Software, Safety Software Level C, this classification cannot be guaranteed for all projects. The following steps must be taken for projects:

1. The first requirement for a project that will use STOMP or eSTOMP is to conduct a grading of software risk. Please refer to the *Grading Software for Projects* section for guidance on grading.
2. Once grading has been completed, the project manager (or delegate) that adopts this STOMP SQAP into a project PMP, shall identify the appropriate level of quality control needed for each design, development, integration, or maintenance task. This must be done in collaboration with the project SQAP.
3. All documentation used to support development or application activities shall be maintained as files for that project.
4. Each project that uses, maintains, and/or expands the STOMP/eSTOMP software will address their project-specific software use in the EPR form.
5. If the software development is not incorporated into the controlled version of the software, then the support for the project specific version must be maintained by the project.

5. Software Quality Assurance

STOMP/eSTOMP software maintenance and development activities shall comply with the applicable requirements in HDI using a graded approach outlined in this plan as the method to meet the requirements outlined in the Laboratory's DOE

approved program. Details of the Lab's approach to assuring quality are contained in the following subsections of this plan.

Safety software requirements are broken out into SQA work activities, per DOE G 414.1-4, Table 4. Mapping Safety Software Types and Grading Levels to SQA Work Activities. The software grading has determined that the following SQA work activities are required for this project:

Table 2 – SQA Work Activities

SQA Work Activity & SQAP Section	**Safety Software Grade Level: C**
Software Project Management & Quality Planning	Grade
Software Risk Management	Grade
Software Configuration Management	Grade
Procurement & Supplier Management	Full
Software Requirements Identification & Management	Full
Software Design & Implementation	Full
Software Safety	Grade
Verification & Validation	Grade
Problem Reporting & Corrective Action	Full
Training of Safety Software	Full

Other required activities include:

- Operation, Maintenance and Retirement
- Reviews & Assessments
- Release Plan
- Records

Orders, Standards, Guides, and PNNL Requirements

The following required Orders, Guides, Standards, and PNNL Requirements are referenced in this plan:

- DOE Order (O) 414.1D *Quality Assurance*
- DOE Guide (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*
- American Society of Mechanical Engineers (ASME) Nuclear Quality Assurance (NQA)-1-2008/NQA-1a-2009 addenda *Quality Assurance Requirements for Nuclear Facility Applications*, Subpart 2.7 *Quality Assurance Requirements for Computer Software for Nuclear Facility Applications*
- PNNL HDI Work Controls, Workflows & Exhibits are listed in the table below:

Table - HDI Work Controls, Workflows & Exhibits

Work Control	**Workflow**	**Exhibit**	**SQAP Section**
Quality Controls for Project Work			All
	Acquire Product or Service Acquire Product or Service via P-Card		Procurement & Supplier Management
	Capture and Protect Intellectual Property		Software Project Management & Quality Planning

	Design, Develop and Evaluation Training		Training of Safety Software
	Develop Corrective or Preventative Actions		Problem Reporting & Corrective Action
	Develop Software for Delivery		All
	Deliver Information Product		Release Plan
	Document Control		Records
	Complete Corrective Actions and Evaluate Effectiveness		Problem Reporting & Corrective Action
	Manage Project Records		Records
	Participate in External Audit or Assessment		Reviews & Assessments/Independent Assessments
	Technology Commercialization		Software Project Management & Quality Planning
		Safety Software Requirements	SQA Work Activities
		Safety Software Classifications	Software Grading
		Safety Software Grading Levels	Software Grading

6. Software Project Management & Quality Planning

This SQAP establishes the management processes, including planning, scheduling, execution, assessing corrective actions, and providing resources for work to provide project deliverables based on risk, safety, software life cycle, complexity, and quality requirements. The established organizational structure, functional responsibilities, levels of authority, and interfaces for those managing, performing, and assessing work are identified in Roles and Responsibilities section of this SQAP.

Changes to this SQAP will be reflected in a revision; a revision history will be maintained with this SQAP. Copies of this SQAP are available to all staff via **STOMP*share***, a PNNL external SharePoint site (see Documentation & References section for more details), for the duration of the software life cycles.

The following project quality controls will be used to address key project risks and assure the repeatability of the results and integrity of the project deliverables and analyses:

- Protection of intellectual property (IP) in accordance with the HDI Cap*ture and Protect Intellectual Property* and *Tech nology Commercialization* workflows.
- All staff involved in software development and maintenance shall follow the Laboratory requirements for quality problems (e.g., deficiency or nonconformance) as defined in the Problem Reporting & Corrective Action section of this plan.
- Identify your customer and/or customer advocate.
- Identify customer specific procedures, specific plans, or standards that need to be applied to the project.
- If software must be acquired, and is safety related, engage the SQAP and follow the follow the HDI *Acquire Product or Service* workflow
- If 3rd party licensing is needed:Engage Technology Commercialization (HDI POC) and customer for IP purposes
    - Describe all 3rd party software in the Design section of this plan
    - Review software licenses for suitability within the software product Notify the PM, PMO director (PMOD) and

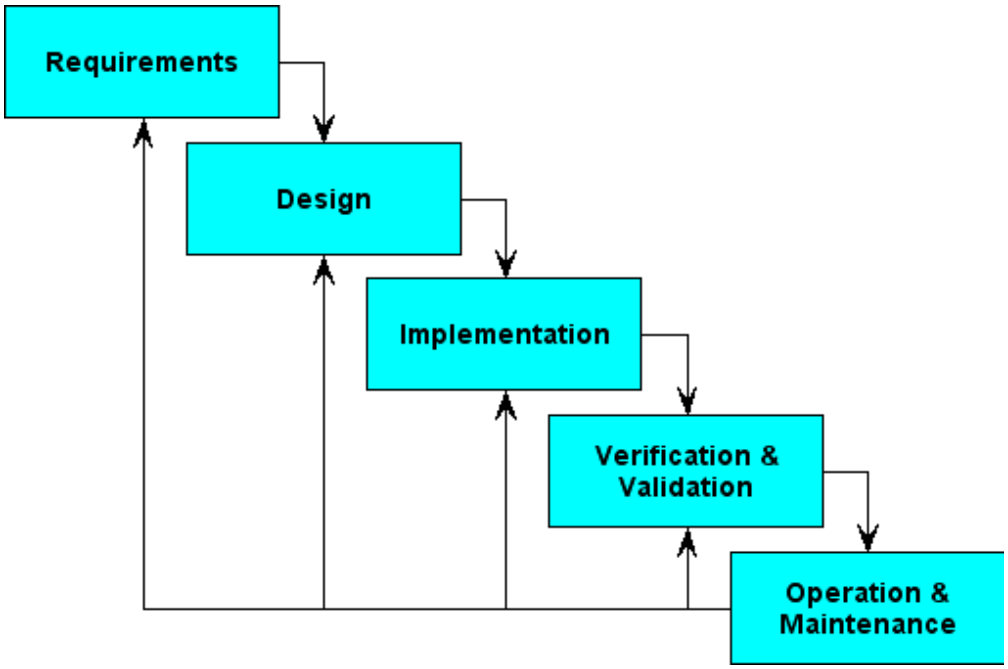
customer of the intent to use 3rd party software
    - Is there any risk of using the software evaluated including the potential for the publisher to drop support for the software and the potential for changes to the software license?
    - Discuss issues or concerns related to 3rd party software with IP, Legal, Export Control, the customer, and others at the discretion of the PM?
- Establish a software development methodology
- Establish configuration management (CM)
- Plan for software support and maintenance

When the determination is made that the software life cycle is completed, all software maintenance and development records will be turned over to DOE records retention. Other closeout activities may be identified at the time the software life cycle ends. The Laboratory's intellectual property office will be notified that the code will no longer be supported. See the Records section of this plan for more details.

Software Development Methodology

The design of STOMP and eSTOMP is described in the waterfall model life cycle shown in the figure below. In brief, the requirements and specifications are first determined, followed by the design of the software prototype. This is followed by implementation, then testing and maintenance. This software life cycle demonstrates that any problems in maintenance can be resolved by returning to the necessary step in the life cycle.

Waterfall Model



Support Software

Software Tools

The following software tools are used to facilitate SQA activities:

COTS Software

The following list of COTS software is maintained and controlled by PNNL. None of these tools affect performance of the safety software.

- Software Configuration Management: Subversion (SVN)
- Document Configuration Management: eSTOMP Confluence site
- JIRA: Issue tracking
- STOMPshare SharePoint site: SCR process as described in this plan and the SCMP

eSTOMP Required Software

The following list of software is required for STOMP and eSTOMP:

- eSTOMP and STOMP utilities as described in the SDD.

This software will be evaluated, reviewed, tested, and accepted for use, and placed under configuration control as part of the life cycle or revised software product. If one of these required tools need to be updated, prior to the update, any impacts to the safety software will be evaluated to determine level of effort (required reviews and testing).

System Software

System software includes operating systems, lower level software layers, assemblers, interpreters, diagnostics and utilities that affect the software product. STOMP and eSTOMP use required, external libraries which are considered *system software* as described in the design section of this plan and in the STOMP SDD and eSTOMP SDD.

The safety software will not be used in an operational environment/capacity. Due to this, the following conditions specify *system software*:

1. The developers' computers are considered the *system software* during development (design & implementation).
2. The PNNL high performance computing (HPC) system will be considered the *system software* for the initial verification of eSTOMP code. This system will most likely also be utilized for future analysis work for eSTOMP.

If the operating system or other major changes to the *system* are required and/or planned, it is required to evaluate, review, test, and accept system software for use, and place under configuration control as part of the life cycle or revised software product. A set of regression tests, as described in the STP, will be run to verify the safety software is working as intended.

7. Software Risk Management

The specific risks for this project, including corporate risk, customer quality risk, environment safety & health risk, and safeguards and security risk will be more specifically addressed by the PMP of each project that uses, maintains, or further develops the STOMP software.

There are specific risks and hazards that pertain to the maintenance and development of the STOMP and eSTOMP software that are identified here, along with the planned means to manage and mitigate these risks and hazards. The primary risk posed by use of this software is that a mistake in the software design or implementation could result in the calculation of an erroneous result, resulting in one or more of the following undesirable outcomes:

1. For projects in progress, adverse impacts to project budget and schedule as corrections are made and calculations repeated to correct the mistake.
2. For completed projects, invalidation of regulatory products (e.g., Environmental Impact Statements, License Applications, or Composite Analysis) that rely on the calculations performed with the software.
3. Damage to the reputation of the Laboratory.

STOMP software is widely adopted and applied both in and out of the Laboratory, at Hanford and at other sites worldwide. Therefore the quality of the software is expected to be above reproach. Every effort must and will be undertaken to minimize the adverse outcomes identified above.

The primary means to minimize the risk of a software error of consequence are:

- Strict adherence to both the STOMP and eSTOMP Software Configuration Management Plans,
- Strict adherence to both the STOMP and eSTOMP Software Test Plans, and
- Timely identification, response, and communication regarding software errors and anomalies discovered by recognized users of the software and reported using the SCR process described in the Software Configuration Management section.

8. Software Safety

Per the Software Grading section of this SQAP, STOMP has been graded as Safety Software, Level C. The intended use of the STOMP and eSTOMP software is to produce numerical predictions of subsurface flow and transport phenomena in variably saturated subsurface environments that may contain radionuclides or other contaminants. In this role, STOMP supports eliminating, limiting, or mitigating nuclear hazards to workers, the public, or the environment. Therefore, STOMP and eSTOMP can be used for Safety and Hazard Analysis, Facility Design, and Safety Management and Administrative Controls. STOMP and eSTOMP are not part of a SSC.

Hazards that have the potential for defeating the STOMP/eSTOMP safety function are software-oriented and user-oriented.

- Software-oriented hazards include faulty algorithms, lack of proper handling of incorrect data or error conditions, buffer overflow, and incorrect sequence of operations due to either logic or timing faults. All of these software-oriented hazards are minimized through the use of extensive testing. Identification and resolution of remaining hazards are handled using the SCR process.
- User-oriented hazards include incorrect problem definition, inappropriate model setup, faulty input data, and incorrect interpretation of STOMP/eSTOMP results. These hazards must be addressed by the individual application projects. Items that could be software-related must by identified and resolved using the SCR process.

9. Software Requirements Identification & Management

The Software Requirements Document (SRD) provides information on the requirements as outlined in the User Guide. The SRD considers requirements related to performance; database; security, including user access control; interface and safety; installation; and design. All safety-related requirements are addressed in the SRD. For security requirements, STOMP and eSTOMP is not considered operational software nor does the code access the internet or external networks; therefore, restrictions to outside users is not applicable. During development, testing and analyses activities PNNL controls the network that the safety software is run on as well as each workstation; therefore, user access control is maintained.

As new capabilities and functionality are added to the software, those requirements are required to be added to the SRD for 1) guiding software development and 2) testing against the new requirements. Requirements for new capabilities will be identified per the Configuration Management process. Each requirement will have its own unique identification number and be traceable to applicable tests. Traceability will be maintained throughout the software life cycle.

The SRD will be reviewed per the Reviews & Assessments section.

Details can be found in the STOMP SRD and the eSTOMP SRD.

10. Design

10.1 Hardware

No specific hardware design is anticipated to support software maintenance and development activities.

10.2 Software Design and Implementation

The Software Design Documents (SDDs) describe how the software will satisfy the design software requirements; how the software functions internally; and how the operating system, structure, interfaces, design inputs and sources, design constraints, timing between components, and data structures are considered and integrated into the code. The control flow, control logic, data flow model (structure, rules), and relationship between data and process are detailed. Methods to mitigate the consequences of software failure are an integral part of the software design.

The SDD will be reviewed per the Reviews & Assessments section.

Details can be found in the STOMP SDD and the eSTOMP SDD.

As mentioned in the Software Requirements Identification & Management section, as new capabilities and functionality are added to the software, those requirements will be updated to the SRS and the same will occur for design. New design details will be added to the SDD for 1) guiding software development and 2) testing against the new design. New design will be interwoven with new requirements as applicable. These new capabilities will be identified per the Configuration Management process. This design will be traceable to the applicable, newly added and/or existing requirements. Traceability will be maintained throughout the software life cycle~~s~~.

10.3 Third Party Software

No third party, stand-alone software is used in the design or interface of STOMP/eSTOMP. External libraries are utilized which are considered *system software* (see section 6) and verified during STOMP and eSTOMP V&V execution. The external libraries are described as follows:

> The serial implementation of STOMP is written in FORTRAN. The direct banded linear system solver is fully integrated into the simulator, but the iterative linear system solver requires linking the simulator with the SPLIB libary that was developed at Indiana University (Bramley and Wang, 1995). The SPLIB solver library (Bram and Wangley 1995) is freely available and may be downloaded at ftp://ftp.cs.indiana.edu/pub/bramley/splib.tar.gz.
>
> The scalable implementation of STOMP (eSTOMP) is also written in FORTRAN, but requires linkage to two external libraries that are freely available:
>
> - Global Arrays (GA) (Nieplocha et al, 2006), available at http://www.emsl.pnl.gov/docs/global/download.shtml
> - PETSc (Balay et al., 2012), available at http://www.mcs.anl.gov/petsc/download/index.html.
> - Message Passing Interface library, such as Gropp et al. (1999)
>
> eSTOMP can be executed with any proprietary version of MPI, or the freely available MPICH, which can be downloaded at mpich.org.
>
> Detailed design and implementation information on these libraries can be found in the SDDs.

11. Software Configuration Management

The Software Configuration Management Plan (SCMP) provide information on the requirements and procedures necessary for adequate control of the software. The SCMP identifies the software, hardware, and documentation requirements and establishes the methodology to generate configuration identifiers; manage engineering, scientific and input/output formatting changes; maintain status accounting; and perform assessments and reviews during requirements analysis, design, development, and maintenance of software release.

The SCMP addresses configuration a) identification, b) control, c) status accounting, and d) audits and reviews. The SCMP will meet the following requirements:

- maintain configuration items under CM in a configuration repository
- release software at the completion of each key activity of the software design process
- establish a baseline configuration
- implement a labeling system for configuration items
- review, verify, and validate baseline changes as appropriate, and approve before implementation
- identify and maintain records of design and development changes
- include evaluation of the effect of the design and development changes on constituent parts and product already delivered in the review
- control and document the unique identification of the product
- maintain user list of the software (as applicable)

The SCMP will be reviewed per the Reviews & Assessments section.

Details can be found in the STOMP SCMP and the eSTOMP SCMP.

12. Problem Reporting & Corrective Action

This section provides a description of how issue reporting will be implemented and documented. Problem reporting and corrective action is coupled with the CM of the software. At a minimum, this process addresses the practices and procedures to be followed for reporting, tracking, and resolving problems or issues identified during the software development and mainte nance process to include,

- How issues/defects/problems/change reports will be reported/submitted.
- How subsequent fixes (corrective actions) will be implemented.
    - The HDI *Develop Corrective or Preventative Actions* and *Complete Corrective Actions and Evaluate Effectiveness* workflows address overall corrective action resolution.
    - Methods for notifying stakeholders of a new version release.
    - States the specific organizational responsibilities concerned with their implementation.

12.1 Design & Implementation Phase

During the Design & Implementation phase of the life cycle, problem reporting and corrective actions are addressed using the Software Change Request (SCR) process that is described in the STOMP and eSTOMP SCMPs and in the Software Configuration Management section this SQAP. The SCR process is used to identify potential problems and request new capabilities. Any user can submit an SCR. The actual SCR steps are managed on **STOMP*share***.

Briefly, the SCR process provides:

- Documentation and tracking to closure of any problems reported for the software and authorization to perform the corrective action.
- Identification in the form of:
    - a unique identifier for each issue
    - detailed steps to recreate the issue
    - solution to the problem and closure
- Evaluation of the severity of the issue to determine if the issue is a defect, error, or new feature request. The issue will be:
    - assigned to a person on the team for resolution
    - updated accordingly throughout its life until closure.
- Communication of the issue and its resolution to the team.

More details on this SCR process can be found in the STOMP SCMP and the eSTOMP SCMP.

12.2 Released Software Phase

Defects found after the release of STOMP or eSTOMP software are managed under the HDI *Quality Controls for Project Work* work control requirement *Monitor work to identify conditions adverse to quality*.

- If a defect has the potential for impacting decisions (conditions adverse to quality) the cause of the condition must be identified and corrective action taken to preclude recurrence. The HDI *Develop Corrective or Preventative Actions* an d *Complete Corrective Actions and Evaluate Effectiveness* workflows will be utilized to manage corrective actions.
- If a defect does not have the potential to be adverse to quality, the same process during the Design & Implementation phase will be implemented.

13. Verification & Validation

This section provides a description of the test process which includes how Verification & Validation (V&V) will be implemented and documented. More specifically, this process involves required test planning, test cases, test results, results analysis, and acceptance.

Verification is performed throughout the life cycle of the software; verifying if the software is working as intended per the requirements (have we built the product right?). Validation activities are performed at the end of the software development process to ensure the software meets the requirements as requested by the customer (have we built the right product?).

At a minimum, this process will verify and validate that the software:

- meets the requirements that guided its design and code development,
- fulfills the intended use and user expectations,
- works as expected,
- can be implemented with the same characteristics,
- satisfies the needs of stakeholders,
- relevant abnormal conditions (defects) have been evaluated for mitigating unintended functions through testing, observation, or inspection techniques. These abnormal conditions (defects) are tracked to resolution,
- traceability of software requirements to software design and acceptance testing has been performed for software based on SQRL requirements.

V&V activities will be performed by competent staff other than those who developed the item being verified or validated. In those instances where this level of independence may not be achieved, an individual associated with the development of the software may perform these activities with justification and approval by the SCM.

Test records, test planning, reviews, test cases, and test results will be maintained as records.

13.1 Test Planning

The Software Test Plans (STPs) provide details of the V&V testing. These plans include requirements to be tested and the test methods and evaluation metrics to be employed. At a minimum, the STPs will include, but is not limited to:

- Software requirements to test
- Test methods
- Characteristics to be tested
- Testing approach
- Pass/fail criteria
- Test deliverables
- Test setup
- Test environments
- Test tracking
- Issue reporting

Details can be found in the STOMP STP and the eSTOMP STP.

13.2 Test Cases

Software test cases will be documented for integration and acceptance testing. No unit tests are performed since integration testing adequately isolates and addresses the software functionality. The test cases identified in the STP exercise key features of the software to demonstrate reliability, functionality, predictability, economy, repeatability, and efficiency. At a minimum, the following will be addressed:

- Each test case will have a unique identifier (for example, incremental numbering) for traceability to software requirements.
- Develop one or more test cases for each requirement and to include, at a minimum:
    - Acceptance criteria
    - Test ID (unique identifier as described above)
    - Description
    - Related requirement(s)
    - Test steps with expected results
    - Tester name

13.3 Test results

At a minimum, test results criterion will be described (pass/fail) for the software testing conducted according to test planning. A Software Test Results (STR) document will be generated to document test results for each release.

Reviews

At a minimum, the following software reviews will be conducted:

- Team members may review and approve software test cases at the CM's discretion.
- CM is required to review and accept test results for *internal acceptance* of the software product.
- Any reviews that are conducted should be independent.

The STP and STR will be reviewed per the Reviews & Assessments section.

14. Training of Safety Software

Training requirements for all staff involved in designing, developing, testing, using, and evaluating Level C custom developed safety software will be documented and reviewed periodically. The SCM shall ensure that only personnel who are knowledgeable and possess adequate technical skills to perform all their assigned tasks may fulfill software development or maintenance tasks. The SCM will identify any additional specific software maintenance tasks that will require staff training in accordance with the HDI *Design, Develop and Evaluation Training* workflow. All training activities will be documented on **STO MP*share***. Training activities may include:

- Execution of one or more of the STOMP/eSTOMP test problems
- Demonstration of Code Development

- Completion of Reading Assignments (e.g., the SQAP; the SCMP)

Individual projects are responsible for maintaining training documentation, including project-required coursework or on-the-job training that is not tracked on **STOMP*share***. This responsibility includes assigning the appropriate training, assuring that the training has been completed prior to project staff conducting work that requires the training, and maintaining project records of training assignments and successful completion.

15. Reviews & Assessments

Reviews and assessments fall into three categories: software document review, management assessments, and independent assessments. Each is described in the following sections.

15.1 Independent Assessments

If an issue is found, the PM for the project using the STOMP/eSTOMP software will use the PNNL Issue Tracking System (ITS/TRACS; https://reliance.pnl.gov) to capture any inspections, project reviews, or audits conducted by the customer of the project or regulatory and oversight agencies. Audits and inspections of the project conducted by regulatory and oversight agencies shall be in accordance with the HDI *Participate in External Audit or Assessment* workflow.

At the request of PM, the SQP-RH who are independent of the software development efforts, will evaluate required software documentation to verify that the operational modes have qualified as Safety Software, Grading Level C, under the conditions identified in the HDI *Safety Software* subject area. When such independent evaluation confirms the operational modes in question are qualified, the finding of software acceptance shall be documented in the project record. APPENDIX C. Software Acceptance Review lists the elements required in the documentation of a Software Acceptance review.

The code version and operational modes of STOMP and eSTOMP that are currently qualified shall be identified on STOMPshare project website.

15.2 Management Assessments

Management assessments will be conducted on an annual basis and the results saved as a project record. Projects that incorporate this SQAP as part of their software management will abide by the instructions found in the Software Grading secti on.

15.3 Software Document Review

The HDI Exhibit: *Safety Software Requirements* requires a regular review of software documentation when the software is determined to be safety software. Software documents will be reviewed on an annual basis and updated as needed. The documents to be reviewed are:

- SQAP (this document)
- Software Requirements Document (SRD)
- Software Design Document (SDD)
- Software Configuration Management Plan (SCMP)
- Software Test Plan (STP)
- Software Test Results (STR)

The above documents shall be reviewed in accordance with the HDI *Exhibit: Graded Approach for Determining Reviewers of Scientific and Technical Information*. The results of each review will be saved as a project record in an appropriate format that can include an email for minor changes, ERICA approvals, or a Document Review Record (see Appendix B for an example) when significant comments are generated and addressed.

Reviews of software change requests and problem reporting are addressed in the SCMP.

16. Release Plan

All software and software documentation, including presentations, that are to be released outside PNNL, including to a project's customer, will be released in accordance with the HDI Deliver Information Product workflow.

17. Records

The SCM, or SCM delegate, shall review, approve, issue, and revise appropriate software documentation to a sufficient detail necessary to qualify the software at the level specified in the Software Grading section of this plan. All such software documentation will be identified as records. Controlled documents are handled in accordance with the HDI *Document Control* workflow.

Records pertaining to code development and maintenance will be maintained as part of the Hydrology Technical Group records, apart from any funding project, to preserve active records throughout the software life cycle. Records include all SQA documents, SCRs and their resolution, and training. Copies of critical records may also be included in the Record Inventory and Disposition Schedule (RIDS) of any funding project, if deemed appropriate in the pertinent project PMP.

Records are defined and managed in accordance with the HDI Project Records workflow. The SCM will appoint a Software Custodian to maintain and control the software documentation records. The Software Custodian is responsible for creating and maintaining the software records inventory. At the end of the software life cycle, the software records will be transferred to another project or archived.

18. Procurement and Supplier Management

Safety software procurement is not planned for STOMP and eSTOMP; therefore, commercial grade dedication of software is not required to be implemented. Non-safety commercial off-the-shelf (COTS) software may be used for test interpretation and related purposes (e.g., Igor Pro, TecPlot, Microsoft Excel, etc.). Use of such software for these purposes will comply with the HDI *Acquire Product or Service Acquire Product or Service via P-Card* and *Use or Develop Software* workflows.

There is no equipment-related software (e.g., firmware) associated with STOMP or eSTOMP.

19. Operation, Maintenance and Retirement

Operational Support

The safety software will not be used in an operational environment/capacity; therefore this section is not applicable.

Maintenance and Support

This section describes the support and maintenance of the delivered software, as applicable. The following is applicable:

- Enhancement requests from the user community
- Revisions to software based on software design requirements
- Reported software problems must be evaluated and corrected if necessary

The SCR process, as described in this plan and the SCMP, will be utilized for maintenance and support.

During the life cycle, maintenance and support will be maintained by the SCM.

Retirement

This section describes the plan for software retirement. This includes the plan for prevention of routine software use once the software product is terminated. Retirement will be implemented by the SCM.

As mentioned previously, this software will not be used in an operational environment/capacity. However, for retirement, the following shall be implemented by the SCM for the safety software:

1. Utilizing the user list, contact all users via email, informing them that the product is no longer in use, it's no longer supported and to retire its use.
2. There is no way to verify that access to the code is restricted from further use as it will be delivered to the customer.
3. Contact the PNNL intellectual property office, informing them that the code will no longer be supported.
4. Confirmation of retirement on PNNL computers where the software may reside.

20. SQAP References

Balay S, J Brown, K Buschelman, V Eijkhout, WD Gropp, D Kaushik, MG Knepley, LC McInnes, BF Smith, and H Zhang, 2012. PETSc Users Manual, ANL-95/11 - Revision 3.3, Argonne National Laboratory, Chicago, Illinois.

Bramley, R and X Wang, 1995. SPLIB: A Library of Iterative Methods for Sparse Linear Systems. Department of Computer Science, Indiana University, Bloomington, Indiana.

Gropp W, E Lusk, and A Skjellum, 1999. Using MPI: Portable Parallel Programming with the Message-Passing Interface, MIT Press, Cambridge, Massachusetts.

Ward AL, MD White, EJ Freeman, JR, and ZF Zhang. 2005. STOMP Sparse Vegetation Evapotranspiration Model for the Water-Air-Energy Operational Mode . PNNL-15465, Pacific Northwest National Laboratory, Richland, WA.

White, MD, and M Oostrom. 2000. STOMP Subsurface Transport Over Multiple Phases: Theory Guide PNNL-12030 (UC-2010), Pacific Northwest National Laboratory, Richland, Washington.

White, MD, and M Oostrom. 2006. STOMP Subsurface Transport Over Multiple Phases, Version 4: User's Guide PNNL-15782 (UC-2010), Pacific Northwest National Laboratory, Richland, Washington.

White MD, and BP McGrail. 2005. STOMP Subsurface Transport Over Multiple Phases Version 1.0 Addendum: ECKEChem Equilibrium-Conservation-Kinetic Equation Chemistry and Reactive Transport. PNNL-15482, Pacific Northwest National Laboratory, Richland, Washington.

White MD, DH Bacon, BP McGrail, DJ Watson, SK White, and ZF Zhang. 2012. STOMP Subsurface Transport Over Multiple Phases: STOMP-CO2 and STOMP-CO2e Guide: Version 1.0 . PNNL-21268, Pacific Northwest National Laboratory, Richland, Washington.

APPENDIX A - Staff Roles

Name	Role				
	Software Configuration Manager	Software Developer	Software Tester	Manager or Administrator	SQP-RH
Bacon, Diana		yes	yes		
Brouns, Tom				Yes (Sector)	
Chen, Xingyuan			yes		
Fang, Yilin		yes	yes		
Fayer, Michael				Yes (TGM)	
Freedman, Vicky	eSTOMP	yes	yes		
Freshley, Mark				Yes (PMOD)	
Hou, Zhangshuan			yes		
Johnson, Tim C			yes		
Johnson, Wayne				Yes (DD)	
Li, Xinya			yes		
Oostrom, Mart		yes	yes		
Rockhold, Mark		yes	yes		
Schlahta, Steve				Yes (PMOD)	
Swannack, Russell					yes
Tartakovsky, Guzel			yes		
Truex, Mike				Yes (Program)	
White, Mark	STOMP	yes	yes		
White, Signe			yes		

Yabusaki, Steve		yes	yes		
Yonkofski, Catherine		yes	yes		
Zhang, Fred			yes		

APPENDIX B - Document Review Record (DRR)

The Document Review Record (DRR) form can be used by project personnel to document technical or management reviews that occur during required review activities.

<Project Name>			**Document Review Record (DRR)**	
The referenced form is submitted for your review. Please return the completed form to respective author or tester: _____ Comments Due: ___/__/___				
Reviewer (Print/Sign)			**Date**	
The Reviewer(s) should not be an originator of the content they review. In the case where a Reviewer is also a content originator, justification and approval is needed.				
Justification			**Project Management Approval (Print/Sign)**	
Comment			**Comments and Recommendations**	
#	**Type** (R=required; S=suggested)			
Add more rows as needed				
Concur with comment resolution, review complete. _____ Reviewer Signature Date				Comments resolved by: _____ Originator Signature Date

APPENDIX C - Software Acceptance Review

This example shows the minimum elements required in a software acceptance review.

- name of the SQP-RH reviewer
- name of the requestor
- date of the review
- software name and version identifier
- description of the items reviewed
- outcome of the review (either Approved or Rejected; if rejected, provide an explanation)