DOE/HS-0003

Software Evaluation of Hotspot
and
DOE Safety Software Toolbox Recommendation

U.S. Department of Energy
Office of Health, Safety and Security
1000 Independence Avenue, S.W.
Washington, DC 20585-2040

March, 2007

Foreword

This report documents the outcome of an evaluation of the Safety Software Quality Assurance (SSQA) attributes of Hotspot, a health physics application, relative to the safety software requirements identified in DOE O 414.1C, *Quality Assurance*. This evaluation, a "gap analysis", is performed according to the implementation guide DOE G 414.1-4, and is a requisite for deciding whether Hotspot should be designated as a toolbox code for DOE's safety software Central Registry. Comments regarding this document should be addressed to:

Debra R. Sparkman
U.S. Department of Energy
1000 Independence Avenue, S.W.
Washington, D.C. 20585-2040
(202) 586-3947
debra.sparkman@hq.doe.gov

Contents

Executive Summary

The development and maintenance of a collection, or "toolbox," of high-use, Department of Energy (DOE) Safety Software Quality Assurance (SSQA)-compliant codes is one of the major improvement actions supported under DOE O 414.1C *Quality Assurance* and 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*. Hotspot Health Physics Codes (referred to as Hotspot) V 2.07 and all future minor releases are being considered for the DOE Safety Software Central Registry.

To evaluate Hotspot's compliance with SSQA requirements, a software-specific gap analysis is necessary. SSQA requirements are those documented in DOE O 414.1C. The gap analysis evaluates the SSQA attributes against the identified work activities specified in DOE O 414.1C and DOE G 414.1-4. The evaluation documented herein provides the results of the gap analysis for Hotspot versions specified above and also recommends whether these products and versions should be added to the DOE's Safety Software Central Registry.

Based on the outcome of the gap analysis, Hotspot version V 2.07 and all future minor releases are recommended for inclusion in the DOE Safety Software Central Registry contingent upon the five critical recommendations being implemented by Lawrence Livermore National Laboratory. Of the eleven work activities evaluated for Hotspot, one work activity was fully met, eight were partially met, and two were not met.

Five work activities (software configuration management, verification and validation, problem reporting and corrective action, training, and model validation/performance) include critical recommendations that if implemented properly will increase the level of compliance for those work activities to acceptable quality levels. It is recommended that the following Hotspot improvement actions be taken prior to inclusion into the Central Registry:

CritRec 1. R3-1 Prompt development and implementation of a formal configuration management plan that documents the process to be followed in providing configuration management for the Hotspot program. This includes documentation for the version control system, software storage, software back-up and disaster planning. Critical to the configuration management implementation is a baseline labeling system that addresses major and minor releases and the establishment of a formal change control process that identifies proposed enhancements and potential defects.

CritRec 2. R8-1: Plan, implement, and document the V&V test processes. The test processes should include both developer-level testing (component, integration, and system) as well as the acceptance testing already performed through the QC method.

CritRec 3. R9-1: Establish, implement and documented a problem reporting, evaluation and notification process consistent with the guidance in DOE G 414.1-4 for level B custom software.

CritRec 4. R10-1: Promptly complete and issue the Hotspot User Manual and online help modules for V 2.07 with awareness that these resources are the primary sources for user training.

CritRec 5. R11-1: Implement a method to read meteorological input data files to satisfy the 95^{th}-percentile dose requirement of DOE-STD-3009-94 Change Notice 3 Appendix A, subsection A.3.3 *Dose Estimation / Atmospheric Dispersion*.

The evaluation team has seventeen additional recommendations that should be considered as future improvements for Hotspot and its software processes. These recommendations as well as the critical recommendations are included in each work activity section in this document and summarized in Section 4. Conclusions and Recommended Actions.

1 Introduction

The development and maintenance of a collection, or "toolbox," of high-use, Department of Energy (DOE) Safety Software Quality Assurance (SSQA)-compliant codes is one of the major improvement actions supported under DOE O 414.1C *Quality Assurance* and 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*. Hotspot Health Physics (referred to as Hotspot) V 2.07 and all future minor releases are being considered for the DOE Safety Software Central Registry.

To evaluate Hotspot's compliance with SSQA requirements, a software-specific gap analysis is necessary. SSQA requirements are those documented in DOE O 414.1C. The gap analysis evaluates the SSQA attributes against the identified work activities specified in DOE O 414.1C and DOE G 414.1-4. The evaluation documented herein provides the results of the gap analysis for Hotspot versions specified above and also recommends whether these products and versions should be added to the DOE's Safety Software Central Registry.

1.1 Objectives

The intent of the gap analysis is to evaluate Hotspot specified above and recommend to the DOE Office of Health, Safety and Security (HSS) whether this safety software should be added to the DOE Safety Software Central Registry.

1.2 Description of Hotspot

Hotspot, developed by Lawrence Livermore National Laboratory (LLNL), was created to provide emergency response personnel and emergency planners with a fast, field-portable set of software tools for evaluating incidents involving radioactive material. The software is also used for safety-analysis of DOE facilities handling nuclear material. Hotspot provides a fast and usually conservative means for estimation the radiation effects associated with the short-term (less than 24 hours) atmospheric release of radioactive materials[1].

Hotspot incorporates Federal Guidance Reports 11, 12, and 13 (FGR-11, FGR-12, FGR-13) Dose Conversion Factors (DCFs) for inhalation, submersion, and ground shine. FGR-12 DCF values are used for submersion and ground shine. In addition to the inhalation 50-year Committed Effective Dose Equivalent DCFs, acute (1, 4, 30 days) DCFs are available for estimating deterministic effects. This acute mode can be used for estimating the immediate radiological impact associated with high acute radiation doses (applicable target organs are the lung, small intestine wall, and red bone marrow).

Hotspot was originally developed in 1985 for deployment for a Hewlett Packard HP-41 system. Hotspot V 2.0 through V 8.0 were Microsoft (MS) DOS-based using Borland Turbo Pascal. In 1999, Hotspot (referred to as Hotspot 98) underwent a significant rewrite in MS Visual Basic V6.0 for the MS Windows 95/98/XP environment. In 2002, Hotspot V2.0 was issued. Throughout the development of Hotspot, new functionality and radionuclides were included. The Hotspot development process has been informal and not developed for compliance to nuclear industry consensus standards.

To be considered for inclusion into DOE Safety Software Central Registry as a toolbox code, software must meet basic criteria. Table 1-1 provides a summary of the justification for Hotspot toward meeting

[1] Hotspot Health Physics Codes web site, http://www.llnl.gov/nhi/hotspot//.

these basic criteria.

Table 1-1. Justification for Adding Hotspot to DOE Safety Software Central Registry

Criterion	Justification
Widespread use of the software across DOE complex for safety related applications.	Hotspot is currently used for radiological emergency response planning at most (if not all) DOE sites. Hotspot is routinely used by DOE and contractor personnel "to perform calculations and develop data used to establish the safety basis for DOE facilities and operations, and to support the variety of safety analyses and safety evaluations developed for these facilities."
Meets definition of *safety software* from DOE O 414.1C. (1) Safety System Software. Software for a nuclear facility that performs a safety function as part of a structure, system, or component and is cited in either (a) a DOE approved documented safety analysis or (b) an approved hazard analysis per DOE P 450.4, *Safety Management System Policy,* dated 10-15-96, and the DEAR clause. (2) Safety and Hazard Analysis Software and Design Software. Software that is used to classify, design, or analyze nuclear facilities. This software is not part of a structure, system, or component (SSC) but helps to ensure the proper accident or hazards analysis of nuclear facilities or an SSC that performs a safety function. (3) Safety 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 requirement 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.	Hotspot fits the safety and Hazard Analysis Software and Design Software definition of safety software specified in DOE O 414.1C2.
Demonstrated and quantifiable benefit for designating the software to the Central Registry.	SCAPA has received numerous inquiries from DOE sites about the software quality assurance (SQA) status of Hotspot and the appropriateness of using Hotspot for safety analyses. Currently, none of the codes in the Central Registry is designed to be broadly applicable for radiological safety planning at DOE sites. EPICode and ALOHA are broadly applicable and technically comparable to Hotspot, but these two models only assess non-radiological hazards. Other models are designed to assess routine radiological releases or only releases from specialized facilities (e.g., nuclear reactors). There is therefore a huge gap in the current coverage of DOE's Safety Software Central Registry toolbox

[2] Glantz, Clifford, Justification for Hotspot Inclusion to DOE Safety Software Central Registry, July 21, 2006.

Criterion	Justification
	codes. The cost for DOE sites to independently perform SQA work activities on Hotspot is considerable. The cost-effective course of action is for the LLNL developers and custodians of Hotspot to work with the DOE to perform any required SQA upgrades or testing of Hotspot, add Hotspot to the Central Registry, and allow all of the DOE sites to use Hotspot without having to individually repeat an extensive and expensive SQA testing program.

1.3 Software Type and Grade Level Designation

As specified in DOE G 414.1-4, current and potential safety software Central Registry software is best described under the custom developed category. Criteria for evaluation of Hotspot should be consistent with the graded approach for custom developed software.

Hotspot is a critical component in the safety analysis for DOE nuclear facilities as well as first responder application in formulating protective actions and preparing Emergency Planning Hazards Assessments (EPHAs). On the basis of DOE G 414.1-4 and the information received in the DOE survey on Hotspot use and application, a Hotspot failure could result in incorrect analysis of hazardous exposures to workers or the public. Therefore, as Hotspot is used for DOE safety analysis applications, the Level B software grade level is justified (Table 1-2).

Table 1-2. Software Grade Level Confirmation

Software Level	Check all that apply	Criteria for Grading Level
A. This grading level includes safety software applications that meet one or more of the following criteria.		Software failure that could compromise a limiting condition for operation.
		Software failure that could cause a reduction in the safety margin for a safety system, structure or component (SSC) that is cited in DOE approved documented safety analysis.
		Software failure that could cause a reduction in the safety margin for other systems such as toxic or chemical protection systems that are cited in either: (a) DOE approved documented safety analysis or, (b) an approved hazard analysis per DOE P 450.1 Safety Management System Policy and the DEAR ISMS clause.
		Software failure that could result in non-conservative safety analysis, design or misclassification of facilities or SSCs
B. This grading level includes safety software applications that do not meet Level A criteria but meet one or more of the following criteria.		Safety management databases used to aid in decision making whose failure could impact safety SSC operation.
	√	Software failure that could result in incorrect analysis, design, monitoring, alarming, or recording of hazardous exposures to workers or the public.
		Software failure that could compromise the defense in depth capability for the nuclear facility.
C. This grading level includes software applications that do not meet Level B criteria but meet one or more of the following		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 Level	Check all that apply	Criteria for Grading Level
criteria.		Software failure that could affect the safe operation of an SSC

1.4 Evaluation Process

The evaluation process is initiated by the software sponsor and led by the software evaluator (Table 1-3). Descriptions of these roles and their responsibilities are included in DOE G 414.1-4 Appendix B. The evaluation focuses on 11 work activities. Work Activities 1 - 10 are those defined in DOE O 414.1C. The Central Registry evaluation process adds an eleventh work activity to address model validation/performance. The graded approach, as specified in DOE G 414.1-4, is applied to the work activities (Table 1-4), with work activity 11, required to be fully met. The term *Full* implies that all elements of the work activity must be addressed. The term *Grade* allows some elements of the work activity to be optional or implemented with less rigor. A list of documents reviewed is contained in Appendix A of this report. The roles of individuals interviewed during this evaluation are listed in Appendix B.

Table 1-3. Contact Information for Hotspot Sponsor and Evaluator

Sponsor	Evaluator
Clifford S. Glantz, SCAPA Chair Pacific Northwest National Laboratory PO Box 999 3200 Q Street Richland, WA 99352 (509) 375-2166 cliff.glantz@pnl.gov	Debra R. Sparkman U.S. Department of Energy 1000 Independence Avenue, S.W. Washington, D.C. 20585-2040 (202) 586-3947 debra.sparkman@hq.doe.gov

Table 1-4. Work Activities and Applicability of DOE G 414.1-4 Criteria for Hotspot Products

Work Activity	Applicability
1. Software project management and quality planning	Full
2. Software risk management	Grade
3. Software configuration management	Full
4. Procurement &supplier management	Full
5. Software requirements identification & management	Full
6. Software design & implementation	Full
7. Software safety	Grade
8. Verification and validation	Grade
9. Problem reporting & corrective action	Full
10. Training personnel in the design, development, use, and evaluation of safety software	Grade
11. Model validation/performance	Full

2 Hotspot Summary

The gap analysis of Hotspot considered a body of information that describes the code and its development, characteristics, strengths, operating parameters, and other pertinent information. Detailed below is a general overview of Hotspot (Table 2-1).

Table 2-1. Overview of Hotspot

Type	Specific Information
Version(s) of Hotspot	V2.05, V2.06, and V2.07 (Beta)
Developing Organizations and Sponsor Information	National Atmospheric Release Advisory Center Lawrence Livermore National Laboratory P.O. Box 808, L-103 Livermore, CA 94551, USA
Auxiliary Software Products	FIDLER- a tool for calibrating radiation survey instruments for ground-survey measurements and initial screening of personnel for possible plutonium uptake in the lung. Nuclear Explosion to estimate the effects of a surface-burst nuclear weapon. Radionuclides in the Workplace – Guide for initial planning of experimental and workplace selection.
Software Platform/Portability	MS Windows 95/98/XP
Programming Languages & Tools	MS Visual Basic V6.0, Macromedia RoboHelp X5
Technical Support Point of Contact	Steve Homann Lawrence Livermore National Laboratory shomann@llnl.gov
Code Procurement Point of Contact	Steve Homann Lawrence Livermore National Laboratory homann1@llnl.gov http://www.llnl.gov/nhi/hotspot/
Code Package Label/Title	Hotspot V 2.0x
Contributing Organization(s)	N/A
Recommended Documentation - Supplied with Code Transmittal upon Distribution or Otherwise Available	User documentation is included with software distribution. Hotspot V 2.07 includes a separate users' manual.
Input Data/Parameter Requirements	Source term (material at risk, release fractions, etc.), meteorology (stability class, wind speed, etc), sample time, deposition velocity, receptor distance.
Summary of Output	Dose (CEDE or TEDE), concentration, deposition, ground shine dose rate, and plume arrival time; all as a function of input receptor distance.
Nature of Problem Addressed by Software	Safety analysis and consequence assessment for DOE nuclear facilities.
Significant Strengths of Software	Easy to use, reliable, and conservative. The model is widely used throughout the DOE complex and has an excellent reputation.
Known Restrictions or Limitations	Relies on simple straight-line plume modeling assumptions that do not account for spatial or temporal variations in meteorological conditions or other complex atmospheric dispersion processes. While Hotpot may be simplistic, it produces results that are conservative.
Preprocessing (set-up) time for Typical Safety Analysis Calculation	Minimal
Execution Time	Seconds
Computer Hardware Requirements	Windows capable computers
Computer Operating System Requirements	Windows 95/98/XP

Type	Specific Information
Other Associated Software Products	For emergency response applications, Hotspot interfaces with the National Atmospheric Release Advisory Center (NARAC) models which provide more sophisticated modeling capabilities that include complex terrain and multi-location real-time wind field data.

3 Review of Hotspot Work Activities

Details on the evaluation process relative to requirements and criteria that are met in compliance with DOE G 414.1-4, are covered in sections 3.1 through 3.11 of this report. The review method consisted of reviewing specific work activity criteria against the information contained in documentation as identified in each of the eleven sections.

The work activities for Hotspot should be evaluated based upon the graded level of the safety software and the applicable software type. In the tables that follow, five qualitative values shall be used to evaluate whether a specific criterion is met:

- Yes – evidence is available to confirm that the program, practices, and/or procedures followed in developing the software satisfy the criterion.
- No – sufficient evidence does not exist to demonstrate the criterion is met
- Partial – some evidence exists that the criterion is met, but has not been finalized or is incomplete
- Uncertain – no basis is available to confirm that the criterion is met
- N/A - the requirement is not applicable.

3.1 Software Project Management and Quality Planning

Project management and quality planning establish the foundation to ensure that a quality product is developed and maintained. Software-specific tasks associated with the completion of the software version being developed, including quality assurance tasks such as performing reviews and testing should be identified and tracked to closure. Identification and proper integration of each of the tasks help ensure that tasks are not overlooked and adequate time is allotted to complete the tasks. For small projects such as Hotspot, project management and planning documents may be integrated and combined and can be the focal point to describe the approach to software development. These documents should include software configuration management (SCM) activities, risk identification and mitigation measures, problem reporting and corrective action methodology, reviews, testing, and the graded approach being applied.

The developer documents proposed changes to the software technical content and architecture in log books, but these changes are not integrated with any other LLNL management system. The log books document the more global aspects of what is to be changed in the software but they do not address the specific planning tasks related to this work activity. In addition, occasional high-level discussion of Hotspot activities are documented in the NARAC monthly reports, since Hotspot is embedded in NARAC. More detail in these reports would be desirable.

There is no formal scheduling system for Hotspot updates and version roll-outs as the developer schedules version changes in response to error reports and on the emergence of new available technologies. Development work proceeds and eventually reaches a completion point without a formal schedule for completion. In addition, software upgrades are not formally scoped. As new features arise,

the developer adds these features to the release that is currently being developed. This stretches out the delivery date for the release being developed, thus delaying the availability of new features.

3.1.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.1-1 lists the criteria reviewed for this work activity and summarizes the findings for Hotspot. Of the six criteria evaluated for this requirement, two are not met, and four are partially met. Thus, the requirement is evaluated as partially met.

3.1.2 Sources and Method of Review

The outdated Hotspot User Manual V8.0 was the initial source of information used to evaluate this work practice. This manual identified the organizational structure for Hotspot. Additional sources include occasional NARAC monthly reports, numerous developer logbooks, and the somewhat limited personal technical files of the developer. Appendix A contains a complete list of the documents reviewed.

3.1.3 Software Quality Assurance-Related Issues or Concerns

While formal documentation such as a Software Project Management Plan (SPMP) or a Software Quality Assurance Plan (SQAP) has not been developed, the developer has an informal management system which appears to be adequate. However, a weakness of such an informal management system is that it is not structured to minimize risk of code errors and other undesirable outcomes of the code and its use.

This work activity is fundamental to the overall development of a software application and due to the human resource limitations, is only partially and informally conducted without any checks and balances to ensure software fidelity. These informal software project management and quality planning work activities were not explicitly documented.

3.1.4 Recommendations

R1-1: Document a comprehensive and complete Software Quality Assurance Plan (SQAP), which contains provisions for software project management, software configuration management and other appropriate elements for Hotspot, following the guidance outlined in DOE G 414.1-4.
R1-2: For each software release, develop a simple integrated schedule with appropriate milestones and other measurable performance criteria to ensure the planned release schedule is met. *Note: this was also identified as an opportunity for improvement in the NNSA safety software quality assurance assessment*[3] *that has not been addressed.*

[3] National Nuclear Security Administration, Assessment of Safety Analysis and Design Software, Lawrence Livermore National Laboratory, Hotspot, May 2004.

Table 3.1-1 — Evaluation of Software Project Management and Quality Planning

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.1-1.1	Are the software specific activities and tasks described, identified and documented?	Partial	Software project management and quality assurance activities have not been formally documented. Informal, yet detailed, log books are kept to document software-specific activities, but these are not integrated with any other LLNL management system. Occasional high-level discussion of activities is documented in NARAC monthly reports. Elements of project management (e.g., tasks and schedules with dependencies) have not been developed.
3.1-1.2	Are these activities and tasks sufficient to properly manage and control the software project and produce the required level of quality?	Partial	Given that the documentation is not available, it is not possible to judge whether software project management and quality planning activities and tasks are sufficient. The required level of quality appears to be met, but the risk of undetected errors is unacceptably high.
3.1-1.3	Do these plans identify the organizational structure associated with the project management and quality planning?	Partial	The outdated User's Manual identifies the single developer and his multiple roles. However, it does not include organizational managers associated with the NARAC program, which Hotspot supports.
3.1-1.4	Are these plans initiated early and maintained throughout the software development life-cycle?	No	Formal plans have not been developed. The software development life-cycle is not controlled by a schedule.
3.1-1.5	Are these plans reviewed, approved and controlled?	No	There are no formal plans to approve and control.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.1-1.6	Do these activities and tasks include the following: a. software project schedule? b. software project scope? c. software engineering activities, including software requirements and design? d. software V&V activities, including reviews and test? e. SCM activities? f. software risk management approach? g. software safety analysis and planning? h. supplier control? i. user and software staff training? j. standards, practices, conventions, and metrics? k. records and document collection, maintenance, and retention? l. problem reporting and corrective action methods?	Partial	Software engineering and SQA activities are not performed using a planned approach. Thus many of these activities are not performed. These include: no software project schedule, no identification of project scope, no SQAP, and only limited planning activities.

3.2 Software Risk Management

Software risk management provides a disciplined environment for proactive decision making to continuously assess what can go wrong in the development or acquisition of the software, determine what risks are important to address, and implement actions to address those risks. Typically software risks include those which are either performance or environment related; such as technical and supportability risk, and programmatic risk. Risk management is often a fundamental tool used in conjunction with project management tools (successful completion of a software project, cost, and schedule).

A risk management process would include as a minimum the following elements: risk planning, risk assessment, risk analysis, and an approach for risk handling (avoidance, control, transfer, etc.). As with project management and quality planning, Hotspot's risk management planning and documentation has been very informal. There is no formal risk management process in place to analyze or document any risk associated with the management and development of Hotspot.

Several potential risks have not been effectively addressed include those surrounding testing, project management, and software requirements document. The following examples are provided:

- *Risk associated with lack of independent testing*. With a single person development team, the lack of independence in performing verification and validation activities introduces risks of bias and increases the potential for defects in the software to go unnoticed. Typically this bias can be offset by performing systematic and comprehensive testing by the developer that would include thorough planning and development of test cases and procedures. However without Hotspot implementing a comprehensive developer-based test planning and execution, the risk for releasing software with latent defects increases.
- *Risk associated with an over-committed personnel resource*. With a single person development team that has multiple responsibilities, the risk of not meeting a software release schedule is high. Performing software project management activities including the identification of adequate resources to update, manage and maintain Hotspot within cost and schedule can reduce this risk.

Currently there is a single person who is the scientific expert and code developer for Hotspot. The lack of resources and lack of performing software project management has resulted in delays in the release of the newer version of Hotspot V 2.07 targeted for release in early 2007.

- *Risk associated with undocumented software requirements*. The fact that many of the software requirements used in the development of Hotspot are not documented or maintained creates a potential risk regarding the loss of requirements, assumptions and algorithms used in the creation of Hotspot.

Even though no formal risk analysis has been performed, the developer has informally utilized some risk mitigation techniques to address potential risk, for example:

- One potential risk includes usage of unproven versions of software programming languages. In this case, the risk mitigation technique used was to use a stable and proven software language, Visual Basic, V6.0.
- Another includes the usage of unproven sources or data that may result in the corruption of static scientific input data needed to run Hotspot. In this case the risk mitigation process used was to embed this data in the Hotspot source code.

3.2.1 Work Activity Evaluation and Results

This work activity may be graded. Table 3.2-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the six criteria evaluated for this requirement, four are not met, and two are partially met. Thus the requirement is evaluated as not met.

3.2.2 Sources and Method of Review

Interviews were performed with the developer, Hotspot code was reviewed, and the following documents were reviewed: NARAC monthly reports, developer technical files, several logbooks, Hotspot User Manual for V8.0, online Help, and Hotspot Users Guide version 2.07 Draft. Appendix A contains a complete list of the documents reviewed.

3.2.3 Software Quality Assurance-Related Issues or Concerns

Hotspot uses stable third-party software development tools such as MS Visual Basic V6.0 and Macromedia RoboHelp X5. The use of these stable tools reduces the software risks. This concept should continue.

By establishing and implementing more formal risk management techniques (i.e., project risk management analysis, risk avoidance, mitigation and transference processes) the developer would experience a cost savings in both development and resource allocation.

3.2.4 Recommendations

R2-1: Document and implement a risk management process for Hotspot. This includes performing a risk analysis and identifying any risk mitigation controls.

Table 3.2-1 Evaluation of Software Risk Management

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.2-1.1	Have the risks associated with the successful completion of the software development or procurement been identified and documented?	No	There is no documentation available to determine if software risk management was considered in the development of Hotspot.
3.2-1.2	Do these risks include risks associated with costs, resource availability, schedule, and technical aspects? Examples include: a. Incomplete or volatile software requirements; b. Specification of incorrect or overly simplified algorithms or algorithms that will be very difficult to address within safety software; c. Hardware constraints that limit the design; d. Potential performance issues with the design; e. A design that is based upon unrealistic or optimistic assumptions; f. Design changes during coding; g. Incomplete and undefined interfaces; h. Using unproven computer and software technologies such as programming languages not intended for the target application; i. Use of a programming language with only minimal experience using the language; j. New versions of the operating system; k. Unproven testing tools and test methods; l. Insufficient time for development, coding, and/or testing; m. Undefined or inadequate test acceptance criteria; n. Potential quality concerns with subcontractors or suppliers	Partial	Even though, there is no formal risk management process, no analysis in place and risk have not been clearly documented, the developer has integrated certain risk management techniques in the development of Hotspot.
3.2-1.3	Have risk thresholds been identified and applied?	No	No evidence exists to indicate risk thresholds were identified and applied.
3.2-1.4	Are the risks evaluated for impact and probability of occurrence initially and periodically through the software life cycle?	No	No evidence exists to indicate initial or periodic assessment of risk occurs.
3.2-1.5	Are the risks prioritized and tracked through the software life cycle?	No	
3.-2-1.6	Are actions taken to mitigate the risks using avoidance, risk reduction, and/or transfer of risks approaches?	Partial	Even though, no formal risk management process or analysis is in place and risks have not been clearly documented, the developer has integrated certain risk management techniques in the development of Hotspot.

3.3 Software Configuration Management

Software configuration management (SCM) activities identify all functions and tasks required to manage the configuration of the software system, including software engineering items, establishing the configuration baselines to be controlled, and software configuration change control process.

3.3.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.3-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the nine criteria evaluated for this requirement, four are met and five are partially met. Thus the requirement is evaluated as partially met.

3.3.2 Sources and Method of Review

Information on configuration management was obtained in a series of interviews with the Hotspot program manager/developer and NARAC and LLNL software quality assurance managers. Discussions focused on the methods used to provide configuration management, version control methods, the means of backing up source code and executables, the location of backup material, the frequency of back-ups, and the documentation for configuration management. Appendix A contains a complete list of the documents reviewed.

3.3.3 Software Quality Assurance-Related Issues or Concerns

The Hotspot program manager has an undocumented system for conducting configuration management which appears to be quite rigorous in its application. These methods have been faithfully maintained for many years and have evolved appropriately to keep pace with technological innovations and changes in good business practice expectations for configuration management. Versions of the code are clearly labeled and software modules are routinely backed up. This includes both automatic and manual back-ups. Back-up copies of the software, including those for previous versions of the code, are redundantly stored in a variety of secure locations. Information on the version of the code and a brief list of changes is stored as part of the back-up process.

The process and procedure for configuration management is well understood by the Hotspot program manager/developer. However there is no official configuration management plan. This weakness can be easily remedied. In addition, the current configuration management system does not include the storage of operating system components, developer's documentation, and software quality assurance documentation. The Hotspot program manager/developer has indicated a willingness to document planning documents and other Verification & Validation (V&V) test materials as part of future archival operations

A configuration management software package is not used for Hotspot. This is in large part owing to all the code development work being performed entirely by the Hotspot program manager. With only one person making code modifications, the use of configuration management software has not been deemed necessary.

3.3.4 Recommendations

R3-1: Prompt development and implementation of a formal configuration management plan that documents the process to be followed in providing configuration management for the Hotspot program. This includes documentation for the version control system, software storage, software back-up and disaster planning. Critical to the configuration management implementation is a baseline labeling system that addresses major and minor releases and the establishment of a formal change control process that identifies proposed enhancements and potential defects. *Note: this was also identified as 2 separate*

opportunities for improvement in the NNSA safety software quality assurance assessment[4] *that has not been addressed.* (CritRec 1).

R3-2: Incorporate technical plans, documentation, testing results, and other important project documentation in the configuration management system. Operating system and commercial software used in Hotspot should also be archived along with the Hotspot source code, executables, and key documentation.

R3-3: The Hotspot program should remain cognizant to the potential need for employing configuration management software. As the Hotspot program evolves and as other atmospheric scientists, health physicists, and computer programmers begin to play an active role in code maintenance, testing, and development, the need to employ formal configuration management software will develop.

Table 3-3 -1 Evaluation of Software Configuration Management

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.3-1.1	Are the methods used to control, uniquely identify, describe, and document the configuration of each version or update of software and its related documentation documented?	Partial	The methods used for configuration management are well understood and clearly articulated, though formal documentation does not exist.
3.3-1.2	Is a configuration baseline defined and adequately controlled?	Yes	The configuration baseline appears to be adequately controlled.
3.3-1.3	Does this baseline include operating system components, any associated runtime libraries, acquired software executables, custom-developed source code files, users' documentation, the appropriate documents containing software requirements, software design, software V&V procedures, test plans and procedures, and all software development and quality planning documents?	Partial	The baseline includes custom-developed source code files, data files, and executables. Quality control tests and data sets are archived as part of the software.
3.3-1.4	Has a baseline labeling system been implemented that addresses the following: m. Unique identification of each configuration item? n. Changes to configuration items by revision?	Partial	Although the baseline labeling system is implemented it does not distinguish between major and minor releases.
3.3-1.5	Is the baseline labeling system used throughout the life of the software development and operation?	Yes	The system has evolved over time but has been used throughout the lifetime of the Hotspot program
3.3-1.6	Are proposed changes to the software documented, evaluated, and approved?	Partial	The Hotspot program manager has a well thought out but informal non-documented program for the documentation and evaluation of software changes.

[4] National Nuclear Security Administration, Assessment of Safety Analysis and Design Software, Lawrence Livermore National Laboratory, Hotspot, May 2004.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.3-1.7	Is software baselined prior to approval for use?	Yes	Software source code is archived in the configuration management file structure prior to release to users.
3.3-1.8	Are only approved changes made to the baselined software?	Yes	The program manager carefully reviews all changes; though a documented program for this will be needed; particularly if contributions are to be made by other staff members
3.3-1.9	Are software verification activities performed for the change to baselined software?	Partial	Testing is performed, but a better documented program with the archival of a comprehensive set of V&V information is needed.

3.4 Procurement and Supplier Management

Procurement documentation should include the technical and quality requirements for the safety software. Procurement specifications should include the requirements for supplier notification of defects, new releases, or other issues that impact the operation; and the mechanisms for the users of the software to report defects and request assistance in operating the software.

Hotspot procures two software products that have significant impact on the software development: MS Microsoft Visual Basic V6.0 and Macromedia RoboHelp X5. These products are in use by millions of users. The suppliers have well established SQA programs. Additionally RFF Electronics RFFLow V5 is procured for diagramming the software design. Future activities most likely will replace RFFLow with Microsoft Visio.

3.4.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.4-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the six criteria evaluated for this requirement, four are met and two are partially met. Thus the requirement is evaluated as partially met.

3.4.2 Sources and Method of Review

Information on procurement and supplier management was obtained in a series of interviewees (including question and answer sessions) with the Hotspot program manager. Discussions focused on the procured software used in Hotspot, how it is employed, and what the program manager does to ensure he is aware of potential issues and updates involving this software. Appendix A contains a complete list of the documents reviewed.

3.4.3 Software Quality Assurance-Related Issues or Concerns

In general, the level of quality assurance applied to the vendor software used to develop Hotspot software and provide its “Help” feature appears quite adequate. The only issue is that these quality assurance activities are not documented in any project quality assurance planning document.

3.4.4 Recommendations

R4-1: Develop and maintain technical and quality requirements for acquired software in the project's quality assurance files.

Table 3.4-1 Evaluation of Procurement and Supplier Management

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.4-1.1	Does the procurement and supplier documentation include both the technical and quality requirements including the following categories of software requirements? a. Functionality b. Safety c. Security d. Performance e. Quality	Partial	No formal procurement documentation exists, though the nature of the Microsoft software products being used would require minimal documentation. However, the quality requirements for supplier products are not documented. Technical documentation from Microsoft on Visual Basic is readily available online.
3.4-1.2	Does the procurement and supplier documentation include all documents to be provided to the customer?	Yes	The license agreement and purchase descriptions indicate the documentation provided with the software.
3.4-1.3	Do the procurement and supplier documents include requirements for or the procedures for supplier notification of defects, new releases, and other issues?	Partial	There is no documentation to describe the procedures, but the program manager/developer monitors the supplier for notifications and other issues.
3.4-1.4	Do the procurement and supplier documents include requirements for or the procedures for users to report defects and requests for assistance?	Yes	Supplier documents and web pages for MS Visual Basic and Macromedia Robohelp include problem reporting procedures to the vendor.
3.4-1.5	Has the delivered product been assessed or otherwise validated to ensure requirements have been met? This evidence may be included in the test results, a test summary, supplier site visit reports, or supplier QA program assessment reports.	Yes	Extensive testing by the Hotspot program manager has been conducted to ensure that the Microsoft software is performing in a manner than meets all expectations and is not causing quality assurance problems for Hotspot
3.4-1.6	Has the supplier's QA program been reviewed to ensure it meets or exceeds the procurement specification requirements? This may include review the supplier's QA program through supplier assessment, supplier self-declaration, third-party certification, or other similar methods.	Yes	The extensive use and testing of the software products in question by millions of users and the suppliers well established QA program go a long way to meeting the intent of this requirement

3.5 Software Requirements Identification and Management

Software requirements are the foundation for all software development and maintenance activities. Requirements provide the basis for the features to be implemented. Requirements include not only functional and performance requirements but also security, user access control, interface and safety requirements, along with installation and design constraints. In order to satisfy the criteria below, software requirements must be identified and documented; traceable to the life cycle process, and consistently described.

Hotspot has no current documented set of software requirements. The developer's log book, users' manuals, online help, and validation reports provide information on the features and operation of Hotspot. Logbooks informally document software requirements and for enhancements included in Hotspot V.07, a developer's folder contains the requirements for the new feature. In addition, a handful of software requirements are covered in various documents. For example the user's manual does identify some of the functional requirements used in the software. There is no traceability established forward to the design or test activities.

3.5.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.5-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the nine criteria evaluated for this requirement, two are not met, six are partially met, and one is not applicable. Thus the requirement is evaluated as partially met.

3.5.2 Sources and Method of Review

It is noted that there is no Software Requirement Document (SRD) for Hotspot. The review approach used during this review involved interviews with the developer, reviewing the Hotspot code, and the following documents: NARAC monthly reports, developer technical files, several logbooks, Hotspot User Manual V8.0, Online Help for Hotspot, Hotspot Users Guide version 2.07 Draft. Appendix A contains a complete list of the documents reviewed.

3.5.3 Software Quality Assurance-Related Issues or Concerns

Even though some requirements, including the scientific calculations, are included in the Users Manual V8.0, the developer's logbooks, software development file folders, and within the online help, there is no systematic process or formal documentation in place to determine if all requirements have been identified or adequately described. Without this level of formality traceability of requirements is very difficult, and an uncertainty in knowing what other other requirements have been used in documenting the life cycle stages of Hotspot.

3.5.4 Recommendations

R5-1: From previous versions of Hotspot, identify and document any critical software requirements used in the development of Hotspot.

R5-2: Develop and document software requirements and traceability to those requirements for Hotspot. Requirements documented should include: functional, performance, security, user access control, interface and safety, and installation and design constraints. *Note: this was also identified as an opportunity for improvement in the NNSA safety software quality assurance assessment[5] that has not been addressed.*

[5] National Nuclear Security Administration, Assessment of Safety Analysis and Design Software, Lawrence Livermore National Laboratory, Hotspot, May 2004.

Table 3.5-1 Evaluation of Software Requirements Identification and Management

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.5-1.1	Are the software requirements defined and documented throughout the safety software life cycle?	Partial	As noted, most functional requirements are documented in various documents, reports, log books, and folders.
3.5-1.2	Are the software requirements uniquely identified?	No	There is no identification for each requirement and no determination that each requirement is unique.
3.5-1.3	Are the requirements controlled and maintained throughout the safety software life cycle to minimize conflicting requirements and to maintain accuracy?	No	There is no formal process in place to document, control or maintain requirements for Hotspot. File folders and other documents are not formally controlled.
3.5-1.4	Are the software requirements traceable throughout the software life cycle?	Partial	Although there is no formal traceability matrix, the developer does maintain file folders, and logbooks which shows how software requirements have changed or been altered throughout the genesis of this software.
3.5-1.5	Are changes to the software requirements updated in any and all documents?	Partial	There is no formal process in place to document, control or maintain requirements for Hotspot. Changes however are sometimes captured and documented within the code, and file folders.
3.5-1.6	Are the requirements consistent with the safety system basis?	N/A	The safety system basis is maintained for each DOE facility by site personnel. This is not the responsibility of the Hotspot program manager/developer.
3.5-1.7	Do the software requirements address each type of the following categories? a. Functional b. Performance/timing c. Security, including user access restrictions d. Interface e. Safety	Partial	Some functional requirements are documented in the user's manual, and within file folders. The status of other requirements is unknown.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.5-1.8	Are the software requirements complete, correct, consistent, clear, testable and feasible?	Partial	Software requirements do exist in various documents, Because the algorithms are consistent, infrequently altered and have proven to be correct then the software requirements can be assumed as correct.
3.5-1.9	Can the software requirements be objectively verified and validated?	Partial	The software requirements can be objectively verified and validated if placed into a single document.

3.6 Design and Implementation

During software design and implementation the software is developed, documented, reviewed, and controlled. The software design should be complete and sufficient to meet the software requirements. The design, including interfaces and data structures, should be completely documented; reviews of the design and code should be performed. Additionally, formal developer testing that includes functional, structural, timing, stress, security, and human-factors testing should be planned, performed and the results documented.

RFF Electronics RFFLow V5 is being used to document the software design of V2.05 and beyond through the use of data flow diagrams. MS Visio is being considered to allow for more flexibility and better management of the software design diagrams. A context diagram identifying all 8 major components of the software is available. Only the 3 major components are decomposed to lower level data flow diagrams. The data flow diagrams include the data elements and their flow path between the various components. The data flow diagram function name maps to the source code module. A data dictionary was provided that details the variable type (e.g., integer, double, string) for each of the data elements.

For V2.07, the software development folder concept is being implemented. The development folder includes a description of the requirement, the process flow for the function, the specific test cases that should be created and exercised, as well as the source code.

Hotspot is written in MS Visual Basic V6.0. Macromedia RoboHelp X5 is used to include an embedded help feature. The code is well documented and includes any pit falls to changing portions of the code. The code header includes the module name, change history, purpose and description. The description provides a level of documentation that would normally be included in the design documentation. Hotspot embeds all external scientific data libraries into the source code, eliminating this interface with external organizations for run-time execution.

Visual Basic developer environment is used heavily by the developer. This development environment links the executable function to the specific source code module being executed and provides the ability to step through the code using break points and display variable contents. This development environment provides the call structure for Hotspot. The graphical user interface (GUI) is used to ensure data input ranges are valid. When a data value is entered that is not within the proper data range, Hotspot resets the data input to the appropriate boundary value. No indication of the reset of the data value is provided to

the user. However the display value shows the boundary value being used. The developer manually traces the requirements to the code modules. Each major functional requirement is a major code module.

Hotspot is in maintenance mode. Since the code is well modularized, component testing is accomplished by integrating the updated module with the remaining modules from the current released version. Besides performing the embedded system level test cases (referred to as QC tests) that are released as part of Hotspot, ad-hoc testing is performed by the program manager/developer, an NARAC senior scientist, and is then release as a beta version to key Hotspot users.

3.6.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.6-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the 15 criteria evaluated for this requirement, four are met, two are not met, seven are partially met, and two are not applicable. Thus the requirement is evaluated as partially met.

3.6.2 Sources and Method of Review

The Hotspot program manager/developer and NARAC senior scientist were interviewed. Source code reviewed through the execution of the MS Visual Basic developer environments. Data flow diagrams, flow diagrams, log books, and data dictionary were reviewed. Appendix A contains a complete list of the documents reviewed.

3.6.3 Software Quality Assurance-Related Issues or Concerns

Historically the software design if documented was limited to hand drawn data flow diagrams in the developer's notebook. Beginning with V2.05 the most important functions were captured using an automated drawing tool. V2.06 expands the data flow diagram documentation to begin using flow diagrams for the major enhancement. These are positive actions and would be beneficial to continue. The use of the Visual Basic development environment assists the developer in ensuring the proper software source code modules are updated and provide good navigation to ensure the interfaces with other modules is easily identified. This is a positive practice and should continue.

The use of software development folders for software in maintenance mode is a beneficial practice when only a single developer modifies specific code modules. The use in Hotspot allows all information regarding the addition of an enhancement to be located in a single location. This is a positive practice. However, having a consolidate view of all Hotspot requirements is essential to ensuring that any new requirements are consistent with and do not conflict with existing requirements. The requirements contained in the software development folder should be consolidated into a more global document.

Developer testing is limited to the execution of a set of system level tests that exercise each of the major features in Hotspot. However these tests do not exercise the abnormal inputs or conditions which could occur, especially with an inexperienced Hotspot user. There is no documentation or record that the developer testing, ad hoc or beta testing is performed.

3.6.4 Recommendations

R6-1: Consolidate requirements from software development folder(s) into a more global requirements document.
R6-2: Expand developer testing to include non-normal test cases and document the execution of those test cases.

R6-3: Document the occurrence of ad-hoc and beta testing by others. Include the relationship of the person(s) performing the ad-hoc tests to the development of Hotspot. It is desired to have this relationship to be independent of the development.
R6-4: Enhance the code to highlight input errors and provide more robust notification of input errors.

Table 3.6-1 Evaluation of Software Design and Implementation

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.6-1.1	Does the safety software design describe the control flow, control logic, mathematical model?	Partial	Starting with V2.05, the major components are documented using data flow diagrams. Additionally the control flow for the V2.07 enhancement has been documented.
3.6-1.2	Is the safety software design complete and sufficient to meet the safety software requirements?	Partial	For the major functions the design is complete and sufficient.
3.6-1.3	Does the safety software design fully describe the interfaces with external components or systems?	Yes	The context diagram includes all external interfaces.
3.6-1.4	Does the safety software design describe how the software functions internally?	Yes	The internal functioning of the software is described in the source code header.
3.6-1.5	Does the safety software design describe the inputs and outputs including allowable or their prescribed ranges?	Partial	The data flow diagram includes inputs and outputs. The data dictionary does not include the data ranges. These are embedded into the GUI source code.
3.6-1.6	Does the safety software design describe the data structures and provide layouts of those structures?	Partial	The data dictionary provides the hierarchy but does not show the relationships between data elements.
3.6-1.7	Does the safety software (design) describe error handling strategies and the use of interrupt protocols?	Partial	The source code comments describe the error handling.
3.6-1.8	Has a traceability between safety software requirements and the design been performed and is documented?	Yes	The software development folder contains the requirements and the design associated with those requirements.
3.6-1.9	Have static analyses such as code reviews been performed on safety software code modules?	No	
3.6-1.10	Is the static analysis performed adequate coverage of critical safety software components?	N/A	No static analysis is performed.
3.6-1.11	Was developer unit, (integration and system) testing completed prior to system level testing?	N/A	Hotspot is in maintenance mode. The modularity of the code allows for developer testing to be performed at the same time as system level testing.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.6-1.12	Was developer testing, including unit, integration, and system level testing, planned and documented?	No	Developer testing is informal and only includes a subset of system level functional tests.
3.6-1.13	Does the developer testing include tests to address functions, code structure and logic, stress and load testing, software performance, and human factors?	Partial	Developer testing only addresses functional and human factors testing.
3.6-1.14	Have the results of developer testing been analyzed and documented?	Partial	The results of the testing are compared with previous version releases. However this is not documented.
3.6-1.15	Where appropriate, have reviews and testing been performed by persons independent of the activity or code module being reviewed or tested?	Yes	Ad-hoc testing is performed by key NARAC personnel as well as beta testing by key users.

3.7 Software Safety

Software design and implementation for software critical to safety addresses the impact of the software component's failure on the overall operation and results of the software. That impact is then mitigated through software design concepts such as isolating the source code modules critical to safety, implementing simple rather than complex logic wherever possible, implementing redundancy for key safety functions, and implementing watch dog processes for critical software processes. For safety analysis software the most important design strategies in the work activity is the isolation of safety functions, not using overly complex logic where simpler more straightforward logic could be used, and proper error handling should a component fail.

The Hotspot software is being designed in a very modular fashion with each primary function being a separate module. Inputs, outputs, and termination of program, are all separate modules. Thus isolation of safety analysis functions is being accomplished. As a safety analysis code for DOE's safety software Central Registry, Hotspot does not interface with any other safety software or safety components and thus does not need to consider its failure impact on other safety software components. It does not need to consider redundancy of functions nor ensuring software processes are running through the use of a watch dog program. The software is written in MS Visual Basic and well commented. The logic flow of the source code does not appear to be overly complex.

3.7.1 Work Activity Evaluation and Results

This work activity may be graded. Table 3.7-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the 5 criteria evaluated for this requirement, 2 are met and 3 are not applicable. A hazard analysis was not performed. For Hotspot as with other safety analysis software, the hazard analysis takes the form of a module failure assessment and implementation of error handling. Thus since a hazard analysis was not formally performed and is not applicable for safety analysis software, the 3 criteria associated with the hazard analysis are not applicable. Thus the requirement is evaluated as met.

3.7.2 Sources and Method of Review

The primary sources of information for this work activity were interviews with the program manager/developer, review of the source code, and review of the data flow diagrams. Appendix A contains a complete list of the documents reviewed.

3.7.3 Software Quality Assurance-Related Issues or Concerns

There are no issues or concerns.

3.7.4 Recommendations

There are no recommendations.

Table 3.7-1 Evaluation of Software Safety

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.7-1.1	Has a hazard analysis of the software at the component level been performed and documented?	Yes	No specific hazard analysis of the software component failure has been performed. However the software does include proper isolation of safety functions and adequate error handling should a component fail.
3.7-1.2	Did the hazard analysis identify the potential failures, the consequences of those failures, and the probably of occurrence associated with those failures?	N/A	No hazard analysis was performed.
3.7-1.3	Have actions been taken to eliminate or mitigate the identified failures based upon the consequences of failure and probability of occurrence?	Yes	Error handling has been implemented. Consequences of failure are not applicable for safety analysis software.
3.7-1.4	Was the hazard analysis periodically reviewed and reassessed for possible changes in identified hazards or the addition of new hazards?	N/A	No hazard analysis was performed.
3.7-1.5	Have changes to the hazards analysis been incorporated into the design of the safety software?	N/A	No hazard analysis was performed.

3.8 Verification and Validation

Verification and Validation (V&V) is the largest area within the SQA work activities. Verification is performed throughout the life-cycle of the safety software. Validation activities are performed at the end of the software development or acquisition processes to ensure the software meets the intended requirements. V&V activities include reviews, inspections, assessments, observations, and testing.

Hotspot was originally developed in 1985. Historically the development process has not required using software engineering or software quality assurance standards or guidelines nor the development of software documents. Consistent with the intent of ANS 10.4, *Criteria for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry*, the available documentation, studies and user experience have been reviewed to judge the codes ability to produce valid responses within the specific domain of its intended use.

The developer clearly has an informal process for reviews and testing. This process is not formally linked to overall LLNL policy and guidance at present, but provides no evidence or large gaps, vagueness, or any other systemic fault. No major unintended conditions are currently known or have been discovered to be left unaddressed. This is due to the origin of the code and its continued residence under the supervision of its original author.

NARAC personnel conduct informal reviews and system level testing of Hotspot. NARAC personnel are easily accessible to the Hotspot program manager/developer. Continual informal discussions and reviews of potential design issues are performed with the NARAC staff. Ad hoc testing is performed by NARAC senior scientist prior to the release of the software. Selected users perform beta testing prior to the official release. Hotspot is also equipped with an internal quality control (QC) capability for all major features that checks operation against standard case results upon code installation and at any point desired thereafter.

3.8.1 Work Activity Evaluation and Results

This work activity may be graded. Table 3.8-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the nine criteria evaluated for this requirement, six are met, one is not met, and two are partially met. Thus, for Hotspot the requirement is evaluated as partially met.

3.8.2 Sources and Method of Review

Documentation of the development and its V&V activities exists in a non-structured and informal form. A more complete document review and evaluation may have been possible if this information was accessible to the evaluation team.

3.8.3 Software Quality Assurance-Related Issues or Concerns

Documentation of the development and its V&V activities exists in a non-structured form. A more complete document review and evaluation may have been possible if this information was accessible to the evaluation team.

3.8.4 Recommendations

R8-1: Plan, implement, and document the V&V test processes. The test processes should include both developer-level testing (component, integration, and system) as well as the acceptance testing already performed.
R8-2: Generate or update and review the software documents associated the SSQA activities (e.g., software requirements, SQA planning, test cases and procedures) according to the recommendations in the other work activities.
R8-3: Validate and document the QC test cases that are built into the software with the results from another DOE safety software Central Registry toolbox code or other means appropriate to ensure the results from the test cases are accurate.

Table 3.8-1 Evaluation of Verification and Validation

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.8-1.1	Are V&V activities performed by competent staff independent of the item being verified or validated?	Partial	Ad hoc and beta testing as well as informal design discussions are performed by NARAC personnel and Hotspot users.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.8-1.2	Do management processes exist for performing each of the following? a. V&V activities b. Management reviews c. Independent technical reviews	No	There is no formal management process. However an NNSA assessment was conducted in 2004.
3.8-1.3	Do V&V activities include reviews and/or inspections of the following applicable items? (Note: These items may be combined or included with other system and software documentation.) a. Software requirements specification b. Software design c. Procurement docs d. Code modules e. Training materials f. User documentation g. Test results	Partial	The software requirements are based on user group input. Design reviews are informal discussions with NARAC personnel. A new users' manual is being written. Plans are to have this reviewed.
3.8-1.4	Do the software development and acceptance test cases and procedures include expected results?	Partial	The QC tests results conducted prior to release are compared with previous test results. The previous test results need to be validated and documented.
3.8-1.5	Are the software development and acceptance test cases, procedures, and test results documented?	Partial	The QC acceptance test cases and results are documented at a high level, describing the test scenario and expected results.
3.8-1.6	Are the software development and acceptance test cases, procedures, and test results placed under configuration management?	Yes	The QC tests are the only test cases that exist. They are part of the source code and under the same configuration process.
3.8-1.7	Do the software acceptance tests include the following types of tests? a. Functional b. Software performance c. Security d. Stress e. Load	Yes	The acceptance tests include built in functional test cases. Hotspot is a stand-alone, single-user application that has no software performance or security requirements. Stress and load testing are not applicable.
3.8-1.8	For new software versions, is regression testing performed during development and acceptance testing?	Yes	The QC tests act as a regression test suite.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.8-1.9	For new software versions, is software documentation updated and reviewed?	Yes	All software documents that exist are updated. The user documentation is updated when impacted by a change. The user manual is being updated for V2.07. Data flow diagrams and data dictionary are updated.

3.9 Problem Reporting and Corrective Action

Coupled with the configuration management of the software system, the problem reporting and corrective action process addresses the appropriate requirements of the quality assurance corrective action system. The reporting and corrective action work activity includes (1) methods for documenting, evaluating and correcting software problems; (2) an evaluation process for determining whether a reported problem is indeed a defect or an error; and (3) the roles and responsibilities for disposition of the problem reports, including notification to the originator of the results of the evaluation.

There are no written procedures, policies or software tools that document the process of identifying, tracking and correcting defects in Hotspot. A single program manager/developer has been involved with the program during its entire 20+ year lifetime. Users contact the program manager/developer via phone or email with questions, suspected problems and suggestions for enhancements. The program manager/developer evaluates each communication and responds as deemed appropriate. For suspected errors or unexpected results, the program manager/developer requests the run specification file to document and troubleshoot the run. Emails are saved for future reference and notes for program changes are maintained in developer notebooks and files. New program versions are posted on a web site. They include a summary of changes within the on-board documentation feature.

Error notification to users is problematic since the developer does not maintain a list of registered users. Hotspot is available without fee via download from the Hotspot web site. Users must periodically check the Hotspot web site for new versions and then read the change summary to learn if an error was identified that could affect their previous work.

3.9.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.9-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the eight criteria evaluated for this requirement, four are met, one is not met and three are partially met. Thus the requirement is evaluated as partially met.

3.9.2 Sources and Method of Review

There is no written procedure for problem reporting and corrective actions. Review of this area relied on-site discussions with the program developer and review of example emails. Appendix A contains a complete list of the documents reviewed.

3.9.3 Software Quality Assurance-Related Issues or Concerns

Problem reporting and corrective action is vital to maintaining safety software. The current process is not formally documented although it appears to be effective in evaluating informal problem reports. There is

no process to identify the safety software users and then provide timely notification of errors. This is a critical aspect of problem reporting since users may be required to repeat previous work when the code is used for planning and analysis in safety applications. However it is difficult to maintain a list of current users for no-fee and no registration requirements web-based applications such as Hotspot.

3.9.4 Recommendations

R9-1: Establish, implement and documented a problem reporting, evaluation and notification process consistent with the guidance in DOE G 414.1-4 for level B custom software.

Table 3.9-1 Evaluation of Problem Reporting and Corrective Action

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.9-1.1	Are the practices and procedures for each of the areas below defined and documented? a. Reporting problems or issues b. Tracking those problems or issues c. Resolving those problems or issues	Partial	The process for evaluating, tracking and resolving problems is not documented. However, a consistent process is performed for reporting, tracking and resolving issues.
3.9-1.2	Are the above practices and procedures implemented as defined above?	Yes	User survey did not reveal any unresolved program errors.
3.9-1.3	Does a process exist for evaluating if the reported problem or issue is a software defect, error, or other source?	Partial	Program manager/developer evaluates each email and phone call to determine if an error exists. There is no documentation of this process other than replies to emails.
3.9-1.4	Are responsibilities for the following activities identified? a. Reporting issues b. Approving changes c. Implementing corrective actions	Yes	Program manager/developer has sole responsibility for the program.
3.9-1.5	Are the corrective actions implemented effective?	Yes	
3.9-1.6	Are the defects and errors associated with the safety software defects and errors correlated with software elements?	Partial	Source code modules are updated with comment statements when deemed appropriate.
3.9-1.7	Has the potential impact of those defects and errors been evaluated?	Yes	
3.9-1.8	Have all users of the safety software been notified of the potential impact of the defects and errors?	No	The originator of the defect is notified. Notification of other users depends on them checking the program web site for a new version or learning about the problem via the grapevine.

3.10 Training Personnel in the Design, Development, Use, and Evaluation of Safety Software

The focus of this work activity is on the knowledge and skill levels of staff to perform respective duties, the activity's impact on the quality of the software products, the users' knowledge and skill level, and

the activity's impact on using and interpreting the results of the software properly. This work activity contains three primary areas: 1) training of personnel in the design and development of the Hotspot applications, 2) training of the operations and use, and 3) training of staff performing evaluation of the Hotspot applications. The last is not applicable in this evaluation.

The Hotspot "staff" is an individual who is recognized as an expert in the technical fields addressed by the code. The program manager/developer is active in trade and professional organizations related to the technical areas of the code; authoring peer-reviewed articles and publications in recognized industry journals and periodicals. Since Hotspot is now interfacing with the NARAC code, NARAC program personnel and incident assessors are directly involved in Hotspot usage and application. The program manager/developer has been intensely involved in the development of Hotspot and similar software for more than 20 years. The program manager/developer is the author of EPICODE, which is in the DOE safety software Central Registry. EPICODE has similar technical characteristics and architecture to Hotspot.

There is no formal training available for the Hotspot user community. Users are trained primarily through the use of the online help capability, embedded within the code and through any DOE site-based training programs. There is an outdated Users' Manual which is being updated and it will be available with the release of V2.07. Its availability will enhance the training potential for the user community. In addition, there is a 2005 Russian presentation that could be the nucleus for a formal training program.

Outside sources such as DOE's sites and its Energy Facility Contractors Group (EFCOG) provide training courses in the Hotspot algorithms and in Hotspot application, but there is no direct feedback mechanism to the program manager/developer and the at-large user community.

3.10.1 Work Activity Evaluation and Results

This work activity may be graded. Table 3.10-1 lists the criteria reviewed for this work activity and summarizes the findings. Of the four criteria evaluated for this requirement, associated with Hotspot, three are partially met and one is unknown. Thus, the requirement is evaluated as partially met.

3.10.2 Sources and Method of Review

The outdated Hotspot User Manual was the initial source of information used to evaluate this work practice. Source code, data flow diagrams, and flow diagrams were reviewed for technical skill level. The embedded online help content was also evaluated. Additional sources include NARAC monthly reports, numerous logbooks, and the personal technical files of the developer. Appendix A contains a complete list of the documents reviewed.

3.10.3 Software Quality Assurance-Related Issues or Concerns

The staff knowledge and skills to implement software engineering and software quality assurance methods and practices that impact quality are obtained from job experience and personal improvement goals. The knowledge level in software testing techniques and practices, design structure, error and exception handling is adequate. No reviews were conducted of detailed test cases and procedures, to determine if best software engineering practices were being implemented, since they were not available. A review of the source code and design documentation indicates a reasonable level of knowledge associated with software design and programming.

3.10.4 **Recommendations**

R10-1: Promptly complete and issue the Hotspot User Manual and online help modules for V2.07 with awareness that these resources are the primary sources for user training. (CritRec 4).

R10-2: Implement a formal training program specific for DOE users and their application of Hotspot. This training should utilize the existing site-specific training, DOE EFCOG presentations and other material which is available. This training should be shared with the Hotspot program manager/developer for adaptation and potential use in the more general Hotspot user community. This requires implementation by DOE.

R10-3: Enhance the user training program effectiveness by including several applied problems and solutions to address the full spectrum of Hotspot applications in the appropriate training material.

R10-4: Structure the training program to incorporate provisions for continuing education to ensure users are trained on new features.

Table 3.10-1 Evaluation of Training Personnel in the Design, Development, Use, and Evaluation of Safety Software

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.10-1.1	Does a training or indoctrination program exist for each of the following personnel assignments? a. Safety software analysis b. Software development (concept to retirement) c. Operations and use d. Assessment or evaluation of safety software	Partial	There is no formal LLNL Hotspot indoctrination program. For some DOE safety analysis users, site-specific training is available. NARAC staff is being trained by other LLNL staff familiar with Hotspot. Training for non-LLNL emergency response users is unknown. There is an outdated training manual which provides some information.
3.10-1.2	Does the training or indoctrination program provide for continuing education and training for each of the above personnel?	Partial	There is an online embedded help capability than can be accessed by the user community. This is periodically updated, thus partially meeting the need for continuing education. The program manager/developer receives some continuing education through involvement in technical meetings and seminars.
3.10-1.3	Do continuing education and training improve the performance and proficiency for each of the above personnel?	Unknown	Since there is no formal training programs, metrics are unavailable.

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.10-1.4	Is the training or indoctrination program designed according to the scope, complexity, and importance of the tasks, education and proficiency of the personnel?	Partial	There is an online embedded help capability than can be accessed by the user community. Although it meets some of the user community needs, it was not explicitly designed based on the scope, education and proficiency of the user community.

3.11 Model Validation/Performance

The purpose of this activity is to determine requirements in methodology that must be met for Hotspot to be acceptable for use in Safety Analysis. A significant goal of this activity was to determine that Hotspot is "addressing the right problem" - that it is using methods that, if applied correctly, will yield results that will satisfy dispersion requirements of DOE-STD-3009-94, Change Notice 3, *Preparation Guide for U.S. Department of Energy Nonreactor Nuclear Facility Safety Analysis.*

3.11.1 Work Activity Evaluation and Results

This work activity should be fully met. Table 3.11-1 lists criteria reviewed for this work activity and summarizes the findings. Of the three criteria evaluated for this requirement, two are met and one is partially met. Thus the requirement is evaluated as partially met.

3.11.2 Sources and Method of Review

Review of this area included DOE directives that establish approved methods to be used in atmospheric dispersion modeling for safety documentation. This included directives for Safety Analysis, Change Notice 3, *Preparation Guide for U.S. Department of Energy Nonreactor Nuclear Facility Safety Analysis* in general as well as for Orders and Guides Emergency Planning specifically, DOE O 151.1C, *Comprehensive Emergency Management System*, and the various volumes associated with DOE G 151.1-1 guidance for emergency preparedness and response.

As they are inputs to DOE-STD-3009, NUREG-1140 *A Regulatory Analysis on Emergency Preparedness for Fuel Cycle and Other Radioactive Material Licensees* and NRC Regulatory Guide 1.145 *Atmospheric Dispersion Models for Potential Accident Consequence Assessments at Nuclear Power Plants* were reviewed.

Several standard texts were reviewed on the subject of modeling dispersion of atmospheric releases. These include:

- Hanna, S.R., G.A. Briggs, and R.P. Hosker, 1982: *Handbook on Atmospheric Diffusion*. DOE/TIC-11223, Department of Energy
- DeVaull, G.E., et al, 1995: *Understanding Atmospheric Dispersion of Accidental Releases*, Published by CCPS/AIChE
- Hanna, S.R., P.J. Drivas, and J.C. Chang, 1996: *Guidelines for Use of Vapor Cloud Dispersion Models 2nd Ed*, Published by CCPS/AIChE

- Hanna, S.R., and R.E. Britter, 2002: *Wind Flow and Vapor Cloud Dispersion at Industrial and Urban Sites*, Published by CCPS/AIChE
- Wilson, D.J., 1995: *Concentration Fluctuations and Averaging Time in Vapor Clouds*, Published by CCPS/AIChE

This evaluation also included on-site discussions with the program manager/developer and review of his notes and references. The following documents were reviewed: developer technical files, Hotspot User Manual V8.0 and Online Help for Hotspot V2.06.

The primary method of review was to determine the DOE-approved methodology, review references to determine how that methodology should be implemented and then review Hotspot documentation to determine that an approved methodology was being implemented. Appendix A contains a complete list of the documents reviewed.

3.11.3 Software Quality Assurance-Related Issues or Concerns

Requirements for calculating dose for comparison against evaluation guidelines are found in DOE-STD-3009-94 Change Notice, March 2006. In particular, Appendix A *Evaluation Guideline* includes a section on Atmospheric Dispersion which states: "Accident phenomenology may be modeled assuming straight-line Gaussian dispersion characteristics, applying meteorological data representing a 1-hour average for the duration of the accident". In addition, DOE G 151.1-1, Volume II *Hazards Surveys and Hazards Assessments* states: "Use of a straight line Gaussian model as the atmospheric dispersion portion of the code is acceptable in most cases for emergency planning". Hotspot does use the straight-line Gaussian model so, in that area, it meets the DOE requirement for Safety Analysis codes.

However, DOE-STD-3009 Appendix A also requires use of historical meteorology – a requirement not currently addressed by Hotspot. From Appendix A: "The 95th percentile of the distribution of doses… is the comparison point for assessment against the EG [Evaluation Guide]. The method used should be consistent with the statistical treatment of calculated X/Q values described in regulatory position 3 of NRC Regulatory Guide 1.145 for the evaluation of consequences along the exclusion area boundary". Hotspot does not provide a method for inclusion of statistical/historical meteorology in order to determine the 95th percentile dose at a given receptor. Thus, Hotspot does not currently meet all of the DOE requirements for a Safety Analysis code. The program manager/developer of Hotspot has stated that this capability can be included in Hotspot. Once this is completed, Hotspot should be fully compliant with DOE's Safety Analysis code requirements.

It should be pointed out that use of 95th percentile meteorology is not a requirement for Emergency Planning Hazards Assessments (EPHA). The Guide allows use of either 95th percentile meteorology or a default worst-case condition of F stability with a wind speed of 1 m/s. Therefore Hotspot meets the limited DOE requirements for use in EPHAs.

DOE's EFCOG Safety Analysis Working Group (SAWG) determined that an acceptable safety analysis code must have the ability to handle multiple weather data to meet requirements in DOE-STD 3009-94, Appendix A for direction-independent 95th percentile X/Q. Hotspot does not address Appendix A sampling of site meteorology for 95th percentile statistics. Therefore, the SAWG determined that since

95th percentile capability is required for safety basis documentation, Hotspot does not meet minimum requirements for safety analysis toolbox code[6].

3.11.4 Recommendations

R11-1: Implement a method to read meteorological input data files to satisfy the 95th-percentile dose requirement of DOE-STD-3009-94 Change Notice 3 Appendix A, subsection A.3.3 *Dose Estimation / Atmospheric Dispersion*. (CritRec 5).

Table 3.11-1 Evaluation of Technical Model Adequacy

Criterion Number	Criterion Specification	Compliant	Summary Remarks
3.11-1.1	Are the models and methods used in the safety software based upon industry/science accepted technical practices?	Partial	As Hotspot does not have the capability to use a historical meteorological data set.
3.11-1.2	Is there evidence that output from the code was compared against equivalent output from an independent code and differences resolved?	Yes	Hotspot output has been compared against other Gaussian plume models as well as LODI (a Lagrangian model used by NARAC) with good results.
3.11-1.3	Do the algorithms and numerical or analytical methods used produce valid results?	Yes	Dispersion algorithms used in Hotspot are based upon well-documented equations that yield valid results. Dispersion coefficients are taken from standard references.

4 Conclusions and Recommended Actions

Of the eleven work activities evaluated for Hotspot, one work activity was fully met, eight were partially met, and two were not met. Table 4-1 details the evaluation results for each work activity. Five work activities (software configuration management, V&V, problem reporting and corrective action, training, and model validation/performance) include critical recommendations that if implemented properly will increase the level of compliance for those work activities to acceptable quality levels.

Table 4-1. Work Activity Evaluation Summary

Work Activity	Evaluation
1. Software project management and quality planning	Not Met
2. Software risk management	Not Met
3. Software configuration management	Partial
4. Procurement and supplier management	Partial
5. Software requirements identification and management	Partial
6. Software design and implementation	Partial
7. Software safety	Met
8. Verification and validation	Partial

[6] O'Kula, K.R., D.Y. Chung, P.R McClure, WSRC-MS-2001-00091, *A DOE Computer Code Toolbox: Issues and Opportunities*.

Work Activity	Evaluation
9. Problem reporting and corrective action	Partial
10. Training personnel in the design, development, use, and evaluation of safety software	Partial
11. Model validation/performance	Partial

This evaluation highlighted a previously known weakness in Hotspot complying with DOE's requirements in DOE-STD 3009, Change Notice 3. Enhancements to Hotspot to eliminate this weakness were discussed with the program manager/developer prior to performing this evaluation. The program manager/developer has suggested Hotspot V2.07 (currently under development) could be enhanced to include the necessary functionality to meet the requirements in DOE-STD 3009 Change Notice 3. This enhancement, if implemented properly, allows Hotspot to meet one of the most important criterions.

The configuration management and problem reporting and corrective action work activities have been evaluated as partially met. Two key aspects of these work activities, a baseline labeling system and a formal change control process were discussed with the Hotspot program manager/developer and both the NARAC and LLNL software quality managers during the on-site evaluation activities. At that time, it was indicated a baseline labeling structure that provides definitive identification of major and minor releases would be established. Additionally the NARAC software quality assurance manager provided an approach to include Hotspot problem reporting with a system (i.e., Bugzilla) being used by NARAC. If implemented properly, these improvements would eliminate the two most significant SCM and problem reporting and corrective work activities weaknesses.

Informal testing was the major contributor to concerns in the V&V work activity. Although there have been no identified significant defects in previously released versions to the DOE users, as with any software product, latent defects may exist in Hotspot. To decrease the potential of DOE users encountering these latent defects and safety analysis decisions becoming suspect, for a code such as Hotspot that is in the maintenance mode, the testing process should be formalized and enhanced to be more robust.

Training material available to DOE users is crucial to the proper use of Hotspot and thus the correct safety analysis decisions being applied. Embedded or online help is available with Hotspot. However this method alone may not be adequate to ensure DOE users' are knowledgeable in using Hotspot. Alternative media, such as a user's guide/manual is highly recommended. Prior to initiating this evaluation, the Hotspot program manager/developer had recognized the need for the additional media to assist Hotspot users and is in the process of updating the Hotspot Users' Manual V8.0 (DOS version) for the V2.07 release being planned. Issuance of this users' manual is an important element in compliance with the Training work activity.

Based on the outcome of the gap analysis and contingent upon the acceptable implementation of the five critical recommendations in the gap analysis, Hotspot V2.06 and all future minor releases are recommended for inclusion in the DOE Safety Software Central Registry.

The five critical recommendations were identified that should be successfully implemented, prior to Hotspot V2.0 and all future minor releases being included into DOE's safety software Central Registry. All critical recommendations were discussed with the Hotspot program manager/developer prior to or during this evaluation. These critical recommendations are listed below:

CritRec 1. **R3-1** Prompt development and implementation of a formal configuration management plan that documents the process to be followed in providing configuration management for the Hotspot program. This includes documentation for the version control system, software storage, software back-up and disaster planning. Critical to the configuration management implementation is a baseline labeling system that addresses major and minor releases and the establishment of a formal change control process that identifies proposed enhancements and potential defects.

CritRec 2. **R8-1:** Plan, implement, and document the V&V test processes. The test processes should include both developer-level testing (component, integration, and system) as well as the acceptance testing already performed through the QC method.

CritRec 3. **R9-1:** Establish, implement and documented a problem reporting, evaluation and notification process consistent with the guidance in DOE G 414.1-4 for level B custom software.

CritRec 4. **R10-1:** Promptly complete and issue the Hotspot User Manual and online help modules for V2.07 with awareness that these resources are the primary sources for user training.

CritRec 5. **R11-1:** Implement a method to read meteorological input data files to satisfy the 95th-percentile dose requirement of DOE-STD-3009-94 Change Notice 3 Appendix A, subsection A.3.3 *Dose Estimation / Atmospheric Dispersion*.

The gap analysis identified a total of twenty-two recommendations, including the above five critical priority recommendations. These recommendations are summarized in Table 4-2

Table 4.2 Summary of Recommendations

No.	Work Activity	Recommendation
1.	Software project management and quality planning	**R1-1:** Document a comprehensive and complete Software Quality Assurance Plan (SQAP), which contains provisions for software project management, software configuration management and other appropriate elements for Hotspot, following the guidance outlined in DOE G 414.1-4.
2.	Software project management and quality planning	**R1-2:** For each software release, develop a simple integrated schedule with appropriate milestones and other measurable performance criteria to ensure the planned release schedule is met.
3.	Software risk management	**R2-1:** Document and implement a risk management process for Hotspot. This includes performing a risk analysis and identifying any risk mitigation controls.
4.	Software configuration management	**R3-1:** Prompt development and implementation of a formal configuration management plan that documents the process to be followed in providing configuration management for the Hotspot program. This includes documentation for the version control system, software storage, software back-up and disaster planning. Critical to the configuration management implementation is a baseline labeling system that addresses major and minor releases and the establishment of a formal change control process that identifies proposed enhancements and potential defects. (CritRec 1).
5.	Software configuration management	**R3-2:** Incorporate technical plans, documentation, testing results, and other important project documentation in the configuration management system. Operating system and commercial software used in Hotspot should also be archived along with the Hotspot source code, executables, and key documentation.

No.	Work Activity	Recommendation
6.	Software configuration management	**R3-3:** The Hotspot program should remain cognizant to the potential need for employing configuration management software. As the Hotspot program evolves and as other atmospheric scientists, health physicists, and computer programmers begin to play an active role in code maintenance, testing, and development, the need to employ formal configuration management software will develop.
7.	Procurement and supplier management	**R4-1:** Develop and maintain technical and quality requirements for acquired software in the project's quality assurance files.
8.	Software requirements identification and management	**R5-1:** From previous versions of Hotspot, identify and document any critical software requirements used in the development of Hotspot.
9.	Software requirements identification and management	**R5-2:** Develop and document software requirements and traceability to those requirements for Hotspot. Requirements documented should include: functional, performance, security, user access control, interface and safety, and installation and design constraints.
10.	Software design and implementation	**R6-1:** Consolidate requirements from software development folder(s) into a more global requirements document.
11.	Software design and implementation	**R6-2:** Expand developer testing to include non-normal test cases and document the execution of those test cases.
12.	Software design and implementation	**R6-3:** Document the occurrence of ad-hoc and beta testing by others. Include the relationship of the person(s) performing the ad-hoc tests to the development of Hotspot. It is desired to have this relationship to be independent of the development.
13.	Software design and implementation	**R6-4:** Enhance the code to highlight input errors and provide more robust notification of input errors.
14.	Verification and validation	**R8-1:** Plan, implement, and document the V&V test processes. The test processes should include both developer-level testing (component, integration, and system) as well as the acceptance testing already performed. (CritRec 2).
15.	Verification and validation	**R8-2:** Generate or update and review the software documents associated the SSQA activities (e.g., software requirements, SQA planning, test cases and procedures) according to the recommendations in the other work activities.
16.	Verification and validation	**R8-3:** Validate and document the QC test cases that are built into the software with the results from another DOE safety software Central Registry toolbox code or other means appropriate to ensure the results from the test cases are accurate.
17.	Problem reporting and corrective action	**R9-1:** Establish, implement and documented a problem reporting, evaluation and notification process consistent with the guidance in DOE G 414.1-4 for level B custom software. (CritRec 3).
18.	Training	**R10-1:** Promptly complete and issue the Hotspot User Manual and online help modules for V2.07 with awareness that these resources are the primary sources for user training. (CritRec 4).
19.	Training	**R10-2:** Implement a formal training program specific for DOE users and their application of Hotspot. This training should utilize the existing site-specific training, DOE EFCOG presentations and other material available. This training should be shared with the Hotspot program manager/developer for adaptation and potential use in the more general Hotspot user community. <u>This requires implementation by DOE.</u>
20.	Training	**R10-3:** Enhance the user training program effectiveness by including several applied problems and solutions to address the full spectrum of Hotspot applications in the appropriate training material.
21.	Training	**R10-4:** Structure the training program to incorporate provisions for continuing education to ensure users are trained on new features.

No.	Work Activity	Recommendation
22.	Model validation/performance	**R11-1:** Implement a method to read meteorological input data files to satisfy the 95th-percentile dose requirement of DOE-STD-3009-94 Change Notice 3 Appendix A, subsection A.3.3 *Dose Estimation / Atmospheric Dispersion*. (CritRec 5).

Appendix A. Documents Reviewed

1. Hotspot Data Dictionary for V2.07
2. Hotspot Data Flow Diagrams for V2.05
3. Hotspot Developer Logbooks
4. Hotspot Flow Diagram for V2.07
5. Hotspot Software Development Folder for V2.07
6. Hotspot User Manual V8.0
7. Hotspot Users Guide version 2.07 Draft
8. NARAC monthly reports
9. ANS 10.4, Criteria for the Verification and Validation of Scientific and Engineering Computer Programs for the Nuclear Industry, American Nuclear Society
10. DeVaull, G.E., et al, 1995: *Understanding Atmospheric Dispersion of Accidental Releases*, CCPS/AIChE
11. DOE G 151.1-1 V2 Hazards Surveys and Hazards Assessments, August 21. 1997
12. DOE O 151.1C, *Comprehensive Emergency Management System*, November 2, 2005
13. DOE-STD-3009-94, Change Notice 3, Preparation Guide for U.S. Department of Energy Nonreactor Nuclear Facility Safety Analysis, March 2006
14. Hanna, S.R., and R.E. Britter, 2002: *Wind Flow and Vapor Cloud Dispersion at Industrial and Urban Sites*, CCPS/AIChE
15. Hanna, S.R., G.A. Briggs, and R.P. Hosker, 1982: *Handbook on Atmospheric Diffusion*. DOE/TIC-11223, Department of Energy
16. Hanna, S.R., P.J. Drivas, and J.C. Chang, 1996: *Guidelines for Use of Vapor Cloud Dispersion Models 2nd Ed*, CCPS/AIChE
17. NNSA, *Assessment of Safety Analysis and Design Software, Hotspot*, May 2004
18. NRC Regulatory Guide 1.145 *Atmospheric Dispersion Models for Potential Accident Consequence Assessments at Nuclear Power Plants*
19. NUREG-1140, *A Regulatory Analysis on Emergency Preparedness for Fuel Cycle and Other Radioactive Material Licensees*
20. O'Kula, K.R., D.Y. Chung, P.R McClure, WSRC-MS-2001-00091, *A DOE Computer Code Toolbox: Issues and Opportunities*
21. Wilson, D.J., 1995 *Concentration Fluctuations and Averaging Time in Vapor Clouds*, CCPS/AIChE

Appendix B. Roles of Individuals Interviewed

1. Hotspot program manager/developer
2. LLNL software quality assurance manager
3. NARAC software quality assurance manager
4. NARAC senior scientist

Appendix C. Definitions

This Appendix contains some of the definitions for terms used in this report. Please refer to 10 CFR 830, DOE O 414.1C, and DOE G 414.-4 for additional definitions.

Acceptance Testing. The process of exercising or evaluating a system or system component by manual or automated means to ensure that it satisfies the specified requirements and to identify differences between expected and actual results in the operating environment. Source: ASME NQA-1-2000.

Administrative Controls. The provisions relating to organization and management, procedures, record keeping, assessment, and reporting necessary to ensure safe operation of a facility. Source: 10 CFR 830.

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. Source: ASME NQA-1-2000.

Gap Analysis. Evaluation of the SQA attributes of specific computer software against identified criteria in DOE O 414.1C and DOE G 414.1-4.

Graded Approach. The process of ensuring that the level of analyses, documentation, and actions used to comply with requirements is commensurate with—

- the relative importance to safety, safeguards, and security;
- the magnitude of any hazard involved;
- the life-cycle stage of a facility or item;
- the programmatic mission of a facility;
- the particular characteristics of a facility or item;
- the relative importance to radiological and nonradiological hazards; and
- any other relevant factors.

Source: 10 CFR 830.

Hazard Analysis. The determination of material, system [including software], process, and plant characteristics that can produce undesirable consequences, followed by the assessment of hazardous situations associated with a process or activity. Source: DOE-STD-3009-94.

Hazard Controls. Measures to eliminate, limit, or mitigate hazards to workers, the public, or the environment, including— 10 CFR 830

(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.

Source: 10 CFR 830.

Nuclear Facility. A reactor or a nonreactor nuclear facility where an activity is conducted for or on behalf of DOE and includes any related area, structure, facility, or activity to the extent necessary to ensure proper implementation of the requirements established in CFR, part 10, section 830. Source: 10 CFR 830.

Software Product. The complete set of computer programs, procedures, and possibly associated documentation and data designated for delivery to a user. Source: IEEE STD-610.12-1990.

Quality. The condition achieved when an item, service, or process meets or exceeds the user's requirements and expectations. Source: 10 CFR 830.

Quality Assurance. All those actions that provide confidence that quality is achieved. Source: 10 CFR 830.

Safety. An all-inclusive term used synonymously with environment, safety, and health to encompass protection of the public, the workers, and the environment. Source: DOE O 414.1C.

Safety-class structures, systems, and components (SC SSCs). Structures, systems, or components, including portions of process systems, whose preventive and imitative function is necessary to limit radioactive hazardous material exposure to the public, as determined from the safety analyses. Source: 10 CFR 830.

Safety-significant structures, systems, and components (SS SSCs). Structures, systems, and components which are not designated as safety-class SSCs, but whose preventive or imitative function is a major contributor to defense in depth and/or worker safety as determined from safety analyses [10 CFR 830]. As a general rule of thumb, safety-significant SSC designations based on worker safety are limited to those systems, structures, or components whose failure is estimated to result in a prompt worker fatality or serious injuries (e.g., loss of eye, loss of limb) or significant radiological or chemical exposure to workers. Source: DOE G 420.1-1

Safety and Hazard Analysis Software and Design Software. Software that is used to classify, design, or analyze nuclear facilities. This software is not part of an SSC but helps to ensure the proper accident or hazards analysis of nuclear facilities or an SSC that performs a safety function. Source: DOE O 414.1C.

Safety 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. Source: DOE O 414.1C.

Safety Management Program. A program designed to ensure a facility is operated in a manner that adequately protects workers, the public, and the environment by covering a topic such as: quality assurance; maintenance of safety systems; personnel training; conduct of operations; inadvertent criticality protection; emergency preparedness; fire protection; waste management; or radiological protection of workers, the public, and the environment. Source: 10 CFR 830.

Safety Software. Includes safety system software, safety and hazard analysis software and design software and safety management and administrative controls software. Source: DOE O 414.1C.

Safety Software Central Registry. A virtual repository of safety software applications, called toolbox codes, having widespread application and having a unique purpose in safety-related functions required to support DOE nuclear facilities. This term is synonymous to Central Registry. The Central Registry is managed and maintained by the DOE Office of Health, Safety and Security (DOE/HS).

Safety Structures, Systems, and Components. Both safety class structures, systems, and components and safety significant structures, systems, and components. Source: 10 CFR 830.

Safety System Software. Software for a nuclear facility7 that performs a safety function as part of a structure, system or component and is cited in either DOE approved documented safety analysis or an approved hazard analysis per DOE P 450.4, *Safety Management System Policy*, dated 10-15-96, and the DEAR clause. Source: DOE O 414.1C.

Software. Computer programs, procedures, and associated documentation and data pertaining to the operation of a computer system. Source: NQA-1-2000

Toolbox Code. Safety software that is included in the DOE' Safety Software Central Registry.

Verification and Validation. The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements. Source: IEEE STD-610.12-1990.

7 Per 10 CFR 830, quality assurance requirements apply to all DOE nuclear facilities including radiological facilities (see 10 CFR 830, DOE Std 1120, and the DEAR clause).

Appendix D. Acronyms

ASME	American Society of Mechanical Engineers
CFR	Code of Federal Regulations
DCF	Dose Conversion Factors
DEAR	Department of Energy Acquisition Regulations
DOE	Department of Energy
DOE G	Department of Energy Guide
DOE O	Department of Energy Order
DOE P	Department of Energy Policy
DOE-STD	Department of Energy Standard
EFCOG	Energy Facility Contractors Group
EPHA	Emergency Preparedness Hazards Assessment
FGR	Federal Guidance Report
HSS	Health, Safety and Security (DOE Office of)
IEEE	Institute of Electrical and Electronics Engineers
ISMS	Integrated Safety Management System
LLNL	Lawrence Livermore National Laboratory
MS	Microsoft
NARAC	National Atmospheric Release Advisory Center
NQA	Nuclear Quality Assurance
NUREG	Nuclear Regulation (NRC)
QA	Quality Assurance
QC	Quality Control
SAWG	Safety Analysis Working Group
SCM	Software Configuration Management
SPMP	Software Project Management Plan
SQA	Software Quality Assurance
SQAP	Software Quality Assurance Plan
SRD	Software Requirements Document
SSC	Structure, System, Or Component
SSQA	Safety Software Quality Assurance
V	Version
V&V	Verification And Validation

Appendix E. References

1. ASME, American Society of Mechanical Engineers, *Quality Assurance Requirements for Nuclear Facilities*, NQA-1-2000.
2. CFR Code of Federal Regulations, *10 CFR 830*, *Nuclear Safety Management Rule.*
3. DOE, U.S. Department of Energy, *Quality Assurance*, DOE O 414.1C, (June 17, 2005).
4. DOE, U.S. Department of Energy, *Safety Software Guide for Use with 10 CFR 830 Subpart A, Quality Assurance Requirements*, *and DOE O 414.1C, Quality Assurance*, DOE G 414.1-4, (June 17, 2005).

Appendix F. Evaluation Team Biographies

Campbell, Larry
Senior Engineer
Fluor Hanford Incorporated
Site Emergency Preparedness Program

Education:
M.S., Nuclear Engineering, Carnegie Mellon University, Pittsburgh, PA, 1970
B.S., Engineering Physics, University of Colorado, Boulder, Colorado 1965

Certifications and Licenses:
Professional engineer (PE), Mechanical Engineering

Mr. Campbell has 42 years of experience working for DOE contractors at three DOE sites. He is currently working at the Hanford site where he reviews Emergency Planning Hazard Assessments for DOE, serves as an Emergency Duty Officer, develops procedures and processes for the site emergency consequence assessment center and manages software quality assurance for emergency response and emergency planning computer codes.

Previous experience includes reactor core design and physics parameter trend monitoring, startup testing at several reactors, reactor operations, operator training, aircraft carrier power plant design and engineering problem solving and reactor shield design.

Computer code experience spans the range from usage of numerous codes to writing special purpose programs and includes university classes and continuing education programs.

Davis, Wayne
Fellow Engineer
Washington Safety Management Solutions (WSMS)

Education:
Bachelor of Nuclear Engineering, Georgia Institute of Technology 1980

Mr. Davis has 27 years of nuclear experience and is presently providing senior-level Emergency Planning consultation to the Department of Energy (DOE) at the Savannah River Site (SRS). He has also developed and delivered radiological and chemical accident analysis, dispersion modeling, and consequence assessment training to the DOE community.

After a decade of increasing levels of responsibility in the commercial nuclear sector, including Startup Test Director and Reactor Engineering Supervisor, Mr. Davis came to SRS in 1989.

Since the mid 90's, Wayne has been involved primarily with technical support to Emergency Planning at SRS. He has also provided technical support to other DOE sites including Lawrence Livermore National Laboratory, Los Alamos National Laboratory, and the Nevada Test Site.

Glantz, Clifford
Senior Staff Scientist
Pacific Northwest National Laboratory

Education:
M.S., Atmospheric Sciences, University of Washington, 1982
B.S., Physics and Atmospheric Sciences, State University of New York at Albany, 1979

Relevant Experience:
Mr. Glantz has been a scientist and project manager at Pacific Northwest National Laboratory (PNNL) since 1982. His research involves work in the fields of emergency response and preparedness, consequence assessment modeling, risk assessment and risk management, critical infrastructure protection, applied atmospheric sciences, and environmental assessment. Mr. Glantz has authored over 50 publications and presented his work at scores of technical conferences.

Mr. Glantz is the Chair of the Department of Energy's (DOE) Subcommittee on Consequence Assessment and Protective Actions (SCAPA). He is also Chair of SCAPA's Consequence Assessment Modeling Working Group. In addition, Mr. Glantz is a member of the Temporary Emergency Exposure Level (TEEL) advisory group (TAG), DOE Meteorology Coordinating Council, and the American Nuclear Society working group that is developing new standards for atmospheric dispersion modeling for emergency response applications.

Mazzola, Carl
Environmental Program Manager/Environmental Technology Specialist
Shaw Environmental Incorporated
Project Management Division

Education:
M.S., Meteorology, Pennsylvania State University, University Park, PA, 1970
B.S., Meteorology, City College of New York, New York, NY, 1968
A.A., Mathematics, Kingsborough Community College, Brooklyn, NY, 1966

Certifications and Licenses:
Certified Consulting Meteorologist #381 — American Meteorological Society (1985) [Past member of the AMS Board of Certified Consulting Meteorologists]
Who's Who in Environmental
Who's Who in Science and Engineering

Relevant Experience:
Mr. Mazzola has 36 years of experience and is presently providing senior-level environmental safety and health (ES&H) consultation to the Department of Energy (DOE) at various locations, focusing on environmental management, risk management, chemical safety and emergency preparedness issues. These facilities mainly include Savannah River Site (SRS), Oak Ridge Reservation (ORR), Sandia National Laboratory (SL), Los Alamos National Laboratory (LANL) and the Nevada Test Site (NTS). He has also been developing and delivering environmental compliance, radiological consequence assessment, and chemical dispersion and consequence assessment training to the DOE community.

Since March 1999, Carl has supported the effort to license, construct and operate the Mixed Oxide Fuel Fabrication Facility (MFFF) to be located in the F-Area of SRS, in the areas of environmental permitting, environmental monitoring, chemical safety, chemical and radiological consequence assessment, licensing documentation, and public relations.

He has been involved in the voluntary consensus process for the past 10 years and is presently the chairman of the Nuclear Facilities Standards Committee (NFSC) of the American Nuclear Society (ANS).

He has published more than 35 technical papers, and is nationally recognized as a subject matter expert in atmospheric transport phenomena. He has testified as an expert witness in Federal and State hearings on several occasions over the past 4 decades.

Nevarez, Johnnie
General Engineer, Nuclear Operations
Department of Energy/National Nuclear Security Administration

Education:
BS Electrical Engineering

Certifications and Licenses:
DOE Std-1172-2003 Safety Software Quality Assurance Qualified
Facility Representative Qualified

Relevant Experience:
Mr. Nevarez' has over fifteen years of work experience within the Department of Energy and NNSA. His technical expertise is focused on Facility Operations. He has completed three functional technical qualification standards. Two as a Facility Representative, and the other in the area of Safety Software Quality Assurance.

As a Facility Representative, he has gained operational knowledge and work experience from various assignments in both nuclear, non-nuclear, construction, and accelerator facilities while employed at the Los Alamos and Sandia Site Offices. His operational expertise covers several functional areas such as Conduct of Operations, Configuration and Maintenance Management, Safety Systems, Occupational and Construction Safety, and Quality Assurance.

He has served as both a team leader and as a subject matter expert on several Readiness Reviews. In addition, Mr. Nevarez currently manages the Readiness Review Training program for the NNSA, and teaches the course as requested.

Mr. Nevarez serves as a member to the Software Quality Assurance Committee for the NNSA. He has participated and led several Software Quality Assurance assessments within the NNSA. In addition, Mr. Nevarez has contributed several articles which were published within the NNSA SQA Handbook Part I and Part II. He developing the training modules associated with Part I of this Handbook which was used to instruct NNSA SQA Members. These efforts have recently been acknowledged by Thomas P. D'Agostino, in a Certificate of Appreciation.

He chairs the Course Advisor Group for the development of the DOE Oversight Course for the implementation of DOE O 226.1.

Previous experiences include the management of the Nuclear Safety Support Division, formally known as the Albuquerque Operations Office, Independent Safety Review Division, which had several delegated authorities regarding the Readiness Review Program, Nuclear Facility Safety Analysis, the Accident Investigation Program, Integrated Safety Management, Quality Assurance, and the Employees Concern Program,

Other work experiences includes two years of work as a Work For Others Program/Project Manager, as an Electrical Engineer for Holmes and Narver at the Nevada Test Site, and as an Electrical Engineer at the New Mexico White Sands Missile Range, Atmospheric Science Laboratory.

Schrader, Bradley Dr.
Vulnerability Assessment Analyst
Battelle Energy Alliance
Idaho National Laboratory
Homeland and National Security Directorate

Education:
Ph.D. Health Physics
M.S. Industrial Safety
B.S. Nuclear Engineering

Certifications and Licenses:
ANSI/ASME NQA-1 Certified Auditor
USNRC qualified Radiation Safety Officer
USNRC/NASA/USDOE certified MORT/Accident Investigator
National Registered Radiation Protection Technician (NRRPT)
American Board of Health Physics Certified Health Physicist (CHP)
National Society of Professional Engineers, licensed professional engineer (PE), Nuclear Engineering

Relevant Experience:
Dr. Schrader provides senior advisory level engineering and technical Health Physics support to the DOE complex. Dr. Schrader is the current developer of the accident consequence software RSAC and the company SME for accidental and radiological sabotage consequence assessment.

Sparkman, Debra
Software Quality Engineer
Department of Energy
Office of the Under Secretary for
Energy, Science and Environment
Chief Nuclear Safety

Education:
B.S. Computer Science, University of the Pacific

Certifications and Licenses:
American Society for Quality Certified Software Quality Engineer
American Society for Quality Certified Quality Auditor
DOE Std-1172-2003 Safety Software Quality Assurance Qualified

Relevant Experience:
Ms. Sparkman is currently a member of the Energy, Science and Environment Chief Nuclear Safety staff. Previously Ms. Sparkman was the lead technical expert for DOE's nuclear safety software quality assurance initiatives on software safety. Recent work has included developing criteria for assessment of software in DOE's nuclear facilities and development of DOE order, guidelines, and assessment criteria for the development of nuclear safety software. Ms. Sparkman provides guidance and assistance to various DOE facilities regarding implementing software quality and assessment practices. She has participated in audits/reviews associated with conformance to DOE's nuclear safety software quality requirements and weapons quality program requirements.

In over 30 years at LLNL, Ms. Sparkman's prior positions have included software quality consultant for weapons surveillance, test manager for the National Ignition Facility; software quality manager for Safeguards and Security Engineering and Computations Division; test manager and operations coordinator for the Argus Security System; quality assurance/test coordinator for the Controlled Material Tracking System; and staff member for the Fission Energy and Systems Safety Computer Safety and Reliability group. She has conducted several internal software process assessments, consulted with numerous programs and projects on software quality assurance practices, and established five software test programs at LLNL.

Ms. Sparkman is the secretary for the Nuclear Weapons Complex Software Quality Assurance Subcommittee, a member of the IEEE Software and Systems Engineering Standards Committee, a member of the ASME NQA-1 Engineering Procurement Process subcommittee, ANS 10.4 working group and has authored and co-authored several technical papers on quality practices including quality practices for safety critical systems. She established an LLNL technical interchange committee to promote software quality knowledge sharing, which has been instrumental in the establishment of a common foundation for DOE's and LLNL's software quality engineers. In addition to these activities, Ms. Sparkman was instrumental in establishing a software quality certification through the American Society for Quality for LLNL software engineers.