



**Department of Energy**

Idaho Operations Office  
1955 Fremont Avenue  
Idaho Falls, ID 83401

September 30, 2004

Ms. T. H. Hobbes, Director  
Facility Hazards Identification/Control  
Bechtel BWXT Idaho, LLC  
P.O. Box 1625, MS 3425  
Idaho Falls, ID 83415-3387

**SUBJECT: Results of the Safety Analysis and Design Software Quality Assurance Assessment  
QSD 2004-93 - (TS-QSD-04-095)**

**REFERENCE: Letter, Spencer Abraham to J. T. Conway, Subject: Implementation Plan for  
Defense Nuclear Facilities Safety Board Recommendation 2002-1, dated  
March 13, 2003**

Dear Ms. Hobbes:

Enclosed is the assessment report of the software Quality Assurance processes associated with the Radiological Safety Analysis Computer Program.

The report concludes that the software system assessed has deficiencies in required documentation, and configuration management.

The conclusions drawn by our investigation of RSAC may be an indicator of similar problems with other INEEL software. Please include in your corrective action plan an extent of condition analysis for similar discrepant conditions in other software under your control.

The assessment team thanks your technical staff for their prompt and open support. Please call Bob Blyth at 526-1181 or me if you have any questions.

Sincerely,



G. C. Bowman, Assistant Manager  
Technical Support

Enclosure

cc w/enc:

R. Lagdon, EH-31, GTN  
T. Hathaway, BBWI, MS 3133  
R. Daniels, DNFSB

B. Schrader, BBWI, MS 3428  
G. Clarke, BBWI, MS 3428  
H. Stromberg, BBWI, MS 3133

R. Miller, BBWI, MS 3133  
L. Vaughan, EM-3.2  
R. Quirk, DNFSB

**United States Department of Energy  
Nuclear Energy – Idaho Operations Office**



**Assessment of Safety Analysis and Design Software**

**Idaho National Engineering and Environmental Laboratory**

**Radiological Safety Analysis Computer Program  
(RSAC)**

## Assessment of Analysis and Design Software

Idaho National Engineering and Environmental Laboratory

### Radiological Safety Analysis Computer Program (RSAC)

Prepared by:



---

R.L. Blyth, NE-ID

9/15/2004

## Table of Contents

Table of Contents.....	3
Acronyms.....	4
Introduction.....	5
Assessment Results.....	6
Findings.....	7
QSD 2004-93-01.....	7
QSD 2004-93-02.....	10
QSD 2004-93-03.....	11
QSD 2004-93-04.....	12
Lessons Learned.....	13
Tailoring.....	13
Detailed Results.....	15
Documents and References.....	33
People Contacted.....	36
Software Quality Assurance Data.....	36
Biography of Assessor.....	36

## Acronyms

BBWI	Bechtel BWXT Idaho
CRAD	Criteria Review and Approach Documents
DDS	Design description for software
DNFSB	Defense Nuclear Facility Safety Board
DOE	Department of Energy
EDF	Engineering Design File
INEEL	Idaho National Environmental Engineering Laboratory
NE-ID	Office of Nuclear Energy, Idaho Operations Office
M & O	Management and Operating
MCP	Management Control Procedure
QA	Quality Assurance
RSAC	Radiological Safety Analysis Computer Program
RTM	Requirements traceability matrix
SCMP	Software configuration management plan
SDD	System Design Description
SMP	Software management plan
SQA	Software Quality Assurance
SQAP	Software quality assurance plan
SRS	Software requirements specification
SSCs	Structures, Systems and Components
STP	Software test plan
URL	Uniform Resource Locator or Identifier
V&V	Verification and Validation

## Introduction

This report presents the results of a DOE NE-ID Operations Office assessment of software quality assurance processes of its M & O contractor, Bechtel BWXT Idaho for safety analysis and design software. The assessment was conducted during July 6 – 15, 2004.

### Background

The purpose of the assessment was to assess quality assurance processes at the Idaho National Engineering and Environmental Laboratory for design and analysis software. The assessment addressed two objectives:

1. Meet Commitment 4.2.4.3 of “Implementation Plan for Defense Nuclear Facilities Safety Board Recommendation 2002-1”;
2. Assess the reliability and robustness of a Safety Class or Safety Significant design and analysis software system at the INEEL.

In 1949 the U.S. Atomic Energy commission established the National Reactor Testing Station for testing various types of nuclear reactors and associated equipment. It is located in southeastern Idaho, containing roughly 890 square miles at an average elevation of 4850 feet above sea level. Several name changes later it is now known as the INEEL.

The DOE Implementation Plan for Defense Nuclear Facilities Safety Board (DNFSB) Recommendation 2002-1, Quality Assurance for Safety-Related Software, March 13, 2003, defines the actions and processes that are being taken to ensure the quality of safety software at defense nuclear facilities. Commitment 4.2.4.3 of the Implementation Plan is to complete the assessment of design and analysis software.

The assessment focused on the application of quality assurance processes applied to the Radiological Safety Analysis Computer Program (RSAC).

RSAC is a DOE complex and internationally used software program that calculates the consequences of a release of radionuclides to the atmosphere. It was first coded in 1968. Since then the software has been maintained and upgraded. The most recent version operates on the MS Windows XP operating system. Verification, validation and system testing appear thorough. RSAC was included in the list of applicable toolbox codes submitted in NE-ID’s response to “Implementation Plan for Defense Nuclear Facilities Safety Board Recommendation 2002-1”, element 4.2.1.5.

### Assessment Criteria

The assessment was based on the criteria and review approach document CRAD 4.2.4.1<sup>1</sup>, “Assessment Criteria and Guidelines for Determining the Adequacy of Software Used in the Safety Analysis and Design of Defense Nuclear Facilities”, which was developed to fulfill Commitment 4.2.4.1 of the

---

<sup>1</sup> A copy of this CRAD is available at <http://www.eh.doe.gov/sqa/assessments/crads/SQA%20CRAD%204.2.4.pdf>.

Implementation Plan (IP) for Defense Nuclear Facilities Safety Board (DNFSB) Recommendation 2002-1. It was also based on requirements established by the INEEL M&O contractor for software quality assurance, in particular MCP-550, Software Management.

### **Assessment Scope**

This assessment was limited to the safety analysis and design software program Radiological Safety Analysis Computer Program (RSAC).

## **Assessment Results**

The assessor reviewed objective evidence and interviewed key personnel to evaluate the level of compliance with the assessment criteria. These information sources are listed in report sections Documents and References, and People Contacted.

RSAC is a DOE complex and internationally used software program that calculates the consequences of a release of radionuclides to the atmosphere. RSAC is copyrighted, but available without charge to requesters. Other software with comparable functionality exists that is also available without charge.

It was first coded in 1968. Since then the software has been maintained and upgraded. The most recent version operates on the MS Windows XP operating system. Verification, validation and system testing appear thorough.

Documentation has not kept pace with the software industry, DOE requirements, and versions of the software code.

The RSAC corrective action system lacks documentation of responsibilities and results.

Configuration management of the software is inadequate.

### **Findings Summary**

1. RSAC does not comply with MCP-550, **QSD 2004-93-01**.
2. Responsibilities for the RSAC corrective action system are not documented, **QSD 2004-93-02**.
3. Effectiveness of the RSAC corrective action system cannot be determined, **QSD 2004-93-03**.
4. Items are not identified and controlled to ensure their proper use, **QSD 2004-93-04**.

## Findings

QSD 2004-93-01

RSAC does not comply with MCP-550

### Applicability Requirement:

MCP-550, Software Management

2.0 Scope

This procedure applies to all software applications except:

- A. Software applications and associated data that, as determined by the responsible manager, are **not** used for the purposes of analysis, design, or operation of a *structure, system or component* (SSC; see def.) and are **not** shared with or used by other organizations, projects, or programs.

4.11 Evaluation of Existing Software Applications

4.11.2 Software Technical Lead and/or System Administrator: Determine if software application meets **any** of the following criteria:

- B. Software application is being modified or enhanced for continued use.

### Discussion

RSAC is considered Analysis Software as defined by MCP-3039, Analysis Software Control.

Because RSAC has been distributed outside the INEEL and modified for continued use, it meets the requirements of MCP-550 sections 2.0 A. and 4.11.2 B., and is therefore subject to the balance of MCP-550.

### Finding

#### Part A

The Enterprise Architecture information system contains information that RSAC is classified as Safety Significant.

No objective evidence that RSAC has been classified in accordance with MCP-550, Section 4.3, was observed.

#### Part B

Requirement

- 4.11.3.2 Upgrade existing documentation applicable to the system component(s) affected by the change to meet the content specified in the graded approach process of this procedure (upgrade of format and structure conventions not required) (see, Appendix C Documentation Requirements).

Condition

Part 1

No objective evidence was observed that the following exist for RSAC:

1. Software quality assurance plan<sup>2</sup>,
2. Software configuration management plan,
3. Software requirements specification<sup>3</sup>,
4. Requirements traceability matrix<sup>4</sup>,
5. Design description for software<sup>5</sup>,
6. Software test plan (STP; includes test cases and test report)<sup>6</sup>.

Part 2

Engineering Design File INEEL-2000-003, "Radiological Engineering Software Management Plan" lists and describes INEEL software codes that require a Software Management Plan. RSAC is included in this list.

RSAC is classified as Safety Significant, which leads to an MCP-550 Level A classification. MCP-550 requires that Level A software have a software management plan that complies with IEEE 1058 "Standard for Software Project Management Plans". Engineering Design File INEEL-2000-003, "Radiological Engineering Software Management Plan" does not contain the information required in IEEE 1058.

Part 3

The current version of the user's manual lists Windows 95/98 or NT as the validated operating systems. RSAC 6.2, with WinRp 1.7 operates on Windows XP.

The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.

---

<sup>2</sup> At this stage in the software life cycle, a full SQAP effort is of questionable value

<sup>3</sup> The user's manual could functionally meet this requirement

<sup>4</sup> Because RSAC has been in use for many years, a case can be made that this requirement is unnecessary.

<sup>5</sup> Because RSAC has been in use for many years, a case can be made that this requirement is functionally met or unnecessary.

<sup>6</sup> A comprehensive compilation of existing testing and V & V could meet most of this requirement

The user's manual is not current.

**Part C**

Requirement

4.10.7      Resolve production software application errors per the SMP and/or SCMP.

Condition

Documented procedures for reporting, tracking and resolving RSAC problems or issues do not exist.

**RSAC does not comply with MCP-550.**

**QSD 2004-93-02**

**Responsibilities for the RSAC corrective action system are not documented**

**Requirement:**

Implementation Plan for Defense Nuclear Facilities Safety Board Recommendation 2002-1, Quality Assurance for Safety Software at Department of Energy Defense Nuclear Facilities, March 13, 2003

CRAD - 4.2.4.1, Revision 3,

Assessment Criteria and Guidelines for Determining the Adequacy of Software Used in the Safety Analysis and Design of Defense Nuclear Facilities

4.8 Software Problem Reporting and Corrective Action

2.0 Organizational responsibilities for reporting issues, approving changes, and performing corrective actions are identified and effective.

**Finding:**

RSAC Organizational responsibilities for reporting, issues, approving changes and performing corrective actions are not documented.

**Responsibilities for the RSAC corrective action system are not documented.**

**QSD 2004-93-03**

**Effectiveness of the RSAC corrective action system cannot be determined**

**Requirement:**

Implementation Plan for Defense Nuclear Facilities Safety Board Recommendation 2002-1,  
Quality Assurance for Safety Software at Department of Energy Defense Nuclear Facilities,  
March 13, 2003

CRAD - 4.2.4.1, Revision 3,  
Assessment Criteria and Guidelines for Determining the Adequacy of Software Used in  
the Safety Analysis and Design of Defense Nuclear Facilities

4.8 Software Problem Reporting and Corrective Action

2. Organizational responsibilities for reporting issues, approving changes and performing corrective actions are identified and effective.

**Finding:**

There is no objective evidence of issues reporting and tracking for RSAC.

**Effectiveness of the RSAC corrective action system cannot be determined.**

**QSD 2004-93-04**

**Items are not identified and controlled to ensure their proper use**

**Requirement:**

10 CFR 830.122, Quality Assurance Criteria

(e) Criterion 5-Performance/Work Processes.

(2) Identify and control items to ensure their proper use.

**Finding:**

There is no methodology to track who has downloaded program upgrades.

The RSAC web site identifies RSAC versions RSAC 6.2 with WinRp 1.7 and WinRp 1.7.7 ( <http://www.inel.gov/rsac/productinfo/rsac6.2wrp1.7/main.asp> ).

When using the most current version of RSAC, version number 2.0 appears on at least one of the start up screens.

**Requirements for identifying and controlling items to ensure their proper use are not met.**

## Lessons Learned

Observing a demonstration of the software early in the field investigation phase of assessments is essential to an efficient assessment.

Including URLs for documents reviewed or referenced is a potential time saving methodology.

## Tailoring

The assessor saw little need to adjust the assessment criteria and guidelines contained in Assessment Criteria and Guidelines for Determining the Adequacy of Software Used in the Safety Analysis and Design of Defense Nuclear Facilities.

The following changes, designated by underlined text, were implemented in this assessment.

Topical Area:

Software Requirement Description

Objective:

Analysis and design software functions, requirements, and their bases are defined, documented and controlled

Software Design Description

Objective:

The Software Design Description (SDD) depicting the major components of the software design is defined, documented and controlled.

Software User Documentation

Objective:

Software documentation is available and controlled to guide the user in installing, operating, managing, and maintaining the software.

## Software Verification and Validation

### Objective:

- (b) V & V is performed by persons not directly involved in generating the software code;

## Software Configuration Management

### Criteria:

1. All software components and products to be managed are identified and controlled.
3. Procedures for modifications to those components and products are followed and controlled.

## Software Procurement

### Criteria:

1. Agreements for the acquisition of software programs or components identify the functional, operational and quality requirements appropriate for their use.

## Detailed Results

1. <b>Prepared by:</b> R.L. Blyth	2. <b>Date Prepared:</b> 7/1/2004	3. <b>Type of Checklist:</b> Software, DNFSB <input type="checkbox"/> External <input type="checkbox"/> Internal	
4. <b>Organization Evaluated:</b> INEEL  <b>System Evaluated:</b> RSAC	5. <b>Evaluation Dates:</b> 7/5-14/2004	6. <b>Source/Requirements Document:</b> DNFSB Recommendation 2002-1 Implementation Plan CRAD - 4.2.4.1, Rev 3, <i>RLB additions</i>	
7. <b>Checklist Completed by:</b>  Assessor: _____ R.L. Blyth <span style="float: right; text-align: center;"></span> _____ 7/14/2004			
<b>Topical Area:</b>		<b>Objective:</b>	
4.1 Software Requirement Description		Analysis and design software functions, requirements, and their bases are defined, documented <i>and controlled</i> .	
<b>Criteria</b>		<b>Comments/Notes/ Results</b>	
1. The functional and performance requirements for the analysis and design software are complete and detailed to perform software design.		RSAC is an existing software program that was originally coded in 1968.  The user's manual is a functional equivalent to a software requirements document (SRD), for software at this stage of the software life cycle process.  Due to the stage in the software lifecycle of this system, generating a SRD is not a value adding activity.  <b>A functional equivalent meets this requirement.</b>	
2. The <i>Software Requirement Description</i> (SRD) is reviewed, controlled, and maintained.		RSAC is an existing software program that was originally coded in 1968.  The user's manual is a functional equivalent to a SRD, for software at this stage of the software life cycle process.  The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.  The user's manual is not current.	

<p>3. Each requirement should be uniquely identified and defined such that it can be objectively verified and validated.</p>	<p>RSAC is an existing software program that was originally coded in 1968. A formal Software Requirements Description at this stage in the life cycle of the software is not a value adding activity.</p> <p>A major V &amp; V effort of the entire code was undertaken in 1988.</p> <p>As system functions were added they were tested or V &amp; V'd.</p> <p>There is no unique identifier for each requirement. It appears that every system function has been thoroughly tested.</p> <p><b>A functional equivalent meets this requirement.</b></p>
--	--

## Software Requirement Description

### Approach:

Determine the existence of SRD documentation, either as a standalone document or embedded in another document, and ensure that it specifies, as applicable, the following:

- Functionality - the functions the software is to perform,
- Performance - the time-related issues of software operation such as speed, recovery time, and response time,
- Design constraints imposed on implementation-phase activities - any elements that will restrict design options,
- Attributes - non-time-related issues of software operation such as portability, acceptance criteria, access control, and maintainability, and
- External interfaces - interactions with people, hardware, and other software.

Determine whether the documents containing the SRD are controlled under configuration change control and document control processes. Verify that the SRD is reviewed and updated as necessary for completeness, consistency, and feasibility for developing a usable code.

Identify the standards and guidelines from applicable site/facility procedures, Federal, or industry standards that are applied to the development of the software. Determine their appropriateness and adequacy for the specific analysis and design software under assessment.

If the above requirements are not available, the perceived software requirements may be identified through available documentation and discussions with the program developer, users, and sponsor. These perceived requirements would then be used as the basis for other topical area assessment activities.

<p style="text-align: center;"><b>Topical Area:</b></p> <p><b>4.2 Software Design Description</b></p>	<p style="text-align: center;"><b>Objective:</b></p> <p>The <i>Software Design Description</i> (SDD) depicting the major components of the software design is defined, documented <i>and controlled</i>.</p>
<p style="text-align: center;"><b>Criteria</b></p>	<p style="text-align: center;"><b>Comments/Notes/ Results</b></p>
<p>1. All software-related requirements are implemented in the design.</p>	<p>RSAC is an existing software program that was originally coded in 1968. Generating a formal Software Design Description at this stage in the life cycle of the software is not a value adding activity.</p> <p>As system functions were added they were tested or V &amp; V'd. A distinct verification of requirements inclusion at this point in the life cycle of the software is not a value adding activity.</p> <p><b>A functional equivalent meets this requirement.</b></p>
<p>2. All design elements are traceable to the requirements.</p>	<p>RSAC is an existing software program that was originally coded in 1968.</p> <p>As system functions were added they were tested or V &amp; V'd. A distinct tracing of design elements to requirements at this point in the life cycle of the software is not a value adding activity.</p> <p><b>A functional equivalent meets this requirement.</b></p>
<p>3. The SDD is reviewed, controlled, and maintained.</p>	<p>RSAC is an existing software program that was originally coded in 1968.</p> <p>Generating a formal Software Design Description at this stage in the life cycle of the software is not a value adding activity.</p> <p>The user's manual, INEEL/EXT-01-00540, is a suitable equivalent to a SRD. It is controlled using the EDMS system.</p> <p>The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.</p> <p>The user's manual is not current.</p>

## Software Design Description

### Approach:

Review the appropriate documents, such as vendor specifications for analyzing and designing software, a description of the components and subcomponents of the software design, including databases and internal interfaces, etc. The design may be documented in a standalone document such as an SDD or embedded in other documents. The SDD should contain the information listed below:

- A description of the major safety components of the software design as they relate to the software requirements
- A technical description of the software with respect to control flow, control logic, mathematical model, and data structure and integrity
- A description of the allowable or prescribed ranges for inputs and outputs
- A description of error handling strategy and use of interrupt protocols
- The design should be described in a manner suitable for translating into computer codes

Determine whether the documents containing the software requirement description are controlled under configuration change control and document control processes. Verify that these documents are reviewed and updated as necessary for completeness, consistency, technical adequacy, and correctness.

In instances where the software the design is not available, the contractor may be able to construct a design summary on the basis of available program documentation, review of the source code (if applicable), and information from the facility staff. Care should be taken to ensure that such a design summary is consistent with the complexity and importance of the software to the safety functions.

<p align="center"><b>Topical Area:</b></p> <p><b>4.3 Software User Documentation</b></p>	<p align="center"><b>Objective:</b></p> <p>Software documentation is available <i>and controlled</i> to guide the user in installing, operating, managing, and maintaining the software.</p>
<p align="center"><b>Criteria</b></p>	<p align="center"><b>Comments/Notes/ Results</b></p>
<p>1. The system requirements and constraints, installation procedures, and maintenance procedures such as database fine-tuning are clearly and accurately documented.</p>	<p>The user's manual addresses the system requirements, constraints, and installation.</p> <p>The current version of the controlled user's manual, INEEL/EXT-01-00540, lists Windows 95/98 or NT as the validated operating systems. RSAC 6.2, with WinRp 1.7 operates on Windows XP.</p> <p>The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.</p> <p>The user's manual is not current.</p>
<p>2. Any operational data system requirements and limitations are clearly and accurately documented.</p>	<p>Operational requirements, limitations and parameters are addressed in the user's manual, INEEL/EXT-01-00540.</p> <p>The user's manual lists Windows 95/98 or NT as the validated operating systems. RSAC 6.2, with WinRp 1.7 operates on Windows XP.</p> <p>The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.</p> <p>The user's manual is not current.</p>
<p>3. Documentation exists to aid the users in the correct operation of the software and to provide assistance for error conditions.</p>	<p>See user's manual and Help system.</p> <p><b>This requirement is met.</b></p>
<p>4. Appropriate software design and coding documentation to assist in any future software modifications is defined and documented.</p>	<p>Code appears to be self-documenting using comment lines.</p> <p><b>This requirement is met.</b></p>

## **Software User Documentation**

### **Approach:**

The team will review the user's manual and related documents. These documents may exist either as a standalone document or embedded in other documents. The user documentation should contain:

- User instructions that contain an introduction, a description of the user's interaction with the software, and a description of any required training necessary to use the software
- Input and output specifications appropriate for the function being performed
- A description of error messages or other indications as a result of improper input or system problems and user response
- Information for obtaining user and maintenance support
- A description of system requirements and limitations such as operating system versions, minimum disk and memory requirements, and any known incompatibilities with other software
- A description of any system requirements or limitations for operational data, such as file sizes
- Recommendations for routine database maintenance and instructions for performing this maintenance
- Design diagrams, structure or flow charts, pseudo code, and source code listings necessary for performing future modifications of custom software

<p style="text-align: center;"><b>Topical Area:</b></p> <p><b>4.4 Software Verification and Validation</b></p>	<p style="text-align: center;"><b>Objective:</b></p> <p>The software V&amp;V process is defined and performed, and related documentation is maintained to ensure that:</p> <ul style="list-style-type: none"> <li>(a) the software adequately and correctly performs all intended functions;</li> <li>(b) <i>V &amp; V is performed by persons not directly involved in generating the software code;</i></li> <li>(c) the software does not perform any unintended function.</li> </ul>
<p style="text-align: center;"><b>Criteria</b></p>	<p style="text-align: center;"><b>Comments/Notes/ Results</b></p>
<p>1. All analysis and design software requirements and design have been verified and validated for correct operation using testing, observation, or inspection techniques.</p>	<p>A major V &amp; V effort of the entire code was undertaken in 1988. Since then, as system functions were added they were tested or V &amp; V'd.</p> <p><b>This requirement is met.</b></p>
<p>2. Relevant abnormal conditions have been evaluated for mitigating unintended functions through testing, observation, or inspection techniques.</p>	<p>A major V &amp; V effort of the entire code was undertaken in 1988. Since then, as system functions were added they were tested or V &amp; V'd.</p> <p><b>This requirement is met.</b></p>

## **Software Verification and Validation**

### **Approach:**

Review the software V&V documentation, either as a standalone document or embedded in another document, to determine if:

- The tasks and criteria are documented for verifying the software in each development phase and validating it at completion,
- The hardware and software configurations pertaining to the software V&V are specified,
- Traceability to both software requirements and design exists,
- Results of the V&V activities, including test plans, test results, and reviews are documented,
- A summary of the status of the software's completeness is documented,
- Changes to software are subjected to appropriate V&V,
- V&V is complete, and all unintended conditions are dispositioned before software is approved for use and,
- V&V is performed by individuals or organizations that are sufficiently independent.

<p style="text-align: center;"><b>Topical Area:</b></p> <p><b>4.5 Software Configuration Management</b></p>	<p style="text-align: center;"><b>Objective:</b></p> <p>The Software Configuration Management (SCM) process and related documentation for safety analysis and design software, including calculational software, are defined, maintained, and controlled.</p>
<p style="text-align: center;"><b>Criteria</b></p>	<p style="text-align: center;"><b>Comments/Notes/ Results</b></p>
<p>1. All software components and products to be managed are identified <i>and controlled</i>.</p>	<p>The software code is controlled using WINCVS.</p> <p>Existing documentation is controlled in EDMS and within the software.</p> <p>The user's manual was last updated in 2001. The most current version of RSAC was issued on January 28, 2003.</p> <p>The user's manual is not current.</p> <p>There is no methodology to track who has downloaded program upgrades.</p> <p>The RSAC web site identifies RSAC versions RSAC 6.2 with WinRp 1.7 and WinRp 1.7.7 ( <a href="http://www.inel.gov/rsac/productinfo/rsac6.2wrp1.7/main.asp">http://www.inel.gov/rsac/productinfo/rsac6.2wrp1.7/main.asp</a> )</p> <p>When using the most current version of RSAC version number 2.0 appears on at least one of the start up screens.</p>
<p>2. For those components and products, procedures exist to manage the modification and installation of new versions.</p>	<p>The software code is controlled using WINCVS controlled software configuration management on server Devlinux.</p> <p>A verification and validation plan exists for RSAC, version 6.</p> <p>RSAC does not have a configuration management plan.</p>
<p>3. Procedures for modifications to those components and products are followed <i>and controlled</i>.</p>	<p>RSAC does not have a configuration management plan.</p>

## Software Configuration Management

### Approach:

Review appropriate documents, such as applicable procedures related to software change control, to determine if a SCM process exists and is effective. This determination is made based on the following actions.

- Verify the existence of an SCM plan, either in standalone form or embedded in another document
- Verify that a configuration baseline is defined and that it is being adequately controlled
- Verify that configuration items such as operating systems, source code components, any associated runtime libraries, acquired software executables, custom-developed source code files, users' documentation, documents containing software requirements, software design, software V&V procedures, test plans, and procedures have been identified and placed under configuration control
- Review procedures governing change management, including installation of new versions of the software components and new releases of acquired software
- Review software change packages and work packages to ensure that:
  - (1) possible impacts of software modifications are evaluated before changes are made,
  - (2) various software system products are examined for consistency after changes are made,
  - (3) software is tested according to established standards after changes have been made
- Verify by sampling that documentation affected by software changes accurately reflects all safety-related changes that have been made to the software
- Interview a sample of cognizant line, engineering, and QA managers and other personnel to verify their understanding of the change control process and commitment to manage changes affecting design, safety basis, and software changes in a formal, disciplined, and auditable manner.

<p align="center"><b>Topical Area:</b></p> <p><b>4.6 Software Quality Assurance</b></p>	<p align="center"><b>Objective:</b></p> <p>SQA activities are evaluated for applicability to the analysis and design software, defined to the appropriate level of rigor, and implemented.</p>
<p align="center"><b>Criteria</b></p>	<p align="center"><b>Comments/Notes/ Results</b></p>
<p>1. SQA activities and software practices for requirements management, software design, software configuration management, procurement controls, V&amp;V (including reviews and testing), and documentation have been evaluated and established at the appropriate level for proper applicability to the analysis and design software under assessment.</p>	<p>Other than V &amp; V, and a Software Lifecycle Plan and Report, form 562.19, there is no objective evidence that MCP-550 is being implemented for RSAC.</p> <p>Due to RSAC's stage in the software life cycle, a complete SQAP is of questionable value.</p>
<p>2. SQA activities have been effectively implemented.</p>	<p>Other than V &amp; V, and a Software Lifecycle Plan and Report, form 562.19, there is no objective evidence that MCP-550 is being implemented for RSAC.</p> <p>Due to RSAC's stage in the software life cycle, a complete and fully implemented SQAP is of questionable value.</p>

## Software Quality Assurance

### Approach:

Determine if an appropriate SQA plan exists, either as a standalone document or embedded in another document, as well as related procedures, QA assessment reports, test reports, problem reports, corrective actions, supplier control, and training. Determine the effectiveness of the SQA program by reviewing the SQA plan. The assessment may also include interviewing managers, engineers, and software users. The SQA plan should identify:

- The software products to which it applies,
- The organizations responsible for maintaining software quality, along with their tasks and responsibilities,
- Required documentation: SRD, SDD, software user documentation, SCM plan, and software V&V plans and results,
- Standards, conventions, techniques, or methodologies that guide software development, as well as methods to ensure compliance to the same,
- Methods for error reporting and developing corrective actions, and
- Provisions for controlling software supplier activities for meeting established requirements.

<b>Topical Area:</b> <b>4.7 Software Procurement</b>	<b>Objective:</b> Vendor-supplied software, either COTS software, custom-developed or modified, requires the appropriate levels of QA commensurate with the level of risk introduced by their use.
<b>Criteria</b>	<b>Comments/Notes/ Results</b>
<b>1.</b> Procurement documents for acquisition of software programs identify the <i>functional, operational and</i> quality requirements appropriate for the level of risk introduced by their use.	N / A  Software was developed at INEEL and it predecessor organizations.  <b>This requirement is met.</b>
<b>2.</b> Acquired software is verified to meet the identified quality requirements.	N / A  Software was developed at INEEL and it predecessor organizations.  <b>This requirement is met.</b>

## **Software Procurement**

### **Approach:**

Vendors that supply COTS and other software are evaluated to ensure that they develop software under an appropriate QA program and are capable of providing software that satisfies the specific requirements. The volume of commercial use for vendor software, especially with COTS software, should be considered in determining the adequacy of the vendor's QA program. The assessment of software procurements shall include the following:

- Determine the existence of acquired software QA requirements. These requirements may be embedded in the DOE contractor's or subcontractor's procurement requirements, SRD, SDD, or an SQA plan.
- Review the methods the site uses to verify that vendor software meets the specified QA requirements, and determine if these methods accomplish those requirements. These methods may be included in an SQA plan or software test plan.
- Review evidence that the vendor software was evaluated for the appropriate level of quality. This evidence may be included in test results, a test summary, vendor site visit reports, or vendor QA program assessment reports.

<p style="text-align: center;"><b>Topical Area:</b></p> <p><b>4.8 Software Problem Reporting and Corrective Action</b></p>	<p style="text-align: center;"><b>Objective:</b></p> <p>Formal procedures for software problem reporting and corrective action for software errors and failures are established, maintained, and controlled.</p>
<p style="text-align: center;"><b>Criteria</b></p>	<p style="text-align: center;"><b>Comments/Notes/ Results</b></p>
<p>1. Practices and procedures for reporting, tracking, and resolving problems or issues identified in both software items and software development and maintenance processes are documented and implemented.</p>	<p>Practices for reporting and resolving RSAC software problems exist. See <a href="http://www.inel.gov/rsac/suggestions.asp">http://www.inel.gov/rsac/suggestions.asp</a>, <a href="http://www.inel.gov/rsac/bugreporting.asp">http://www.inel.gov/rsac/bugreporting.asp</a> and <a href="http://www.inel.gov/rsac/updates/">http://www.inel.gov/rsac/updates/</a></p> <p>Documented procedures for reporting, tracking, and resolving RSAC problems or issues do not exist.</p>
<p>2. Organizational responsibilities for reporting issues, approving changes, and performing corrective actions are identified and effective.</p>	<p>Documented procedures for reporting, tracking, and resolving RSAC problems or issues do not exist.</p> <p>Organizational responsibilities for reporting, issues, approving changes and performing corrective actions are not documented for RSAC.</p> <p>There are no documented unresolved issues or problems reported with respect to RSAC.</p> <p>Program updates indicate that problems with RSAC have been discovered or reported and fixed. <a href="http://www.inel.gov/rsac/updates/">http://www.inel.gov/rsac/updates/</a></p> <p>There is insufficient objective evidence to verify the effectiveness of RSAC's corrective action system.</p>

### **Software Problem Reporting and Corrective Action**

#### **Approach:**

Review documents and interview facility staff responsible for problem reporting and notification to determine if:

- A formal procedure exists for software problem reporting and corrective action development that addresses software errors, failures, and resolutions
- Corrections and changes are executed according to established change control procedures
- The problems that impact the software's operation are promptly reported to affected organizations
- Corrections and changes are evaluated for impact and approved before being implemented
- Corrections and changes are verified for correct operation and to ensure that no side effects were introduced before being implemented
- Preventive measures and corrective actions are provided to affected organizations in a timely manner commensurate with the impact of the original defect
- The organizations responsible for problem reporting and resolution are defined



## Documents and References

Identifying No.	Title	Rev. #	Date	URL
Tracking No. 169104	Software Lifecycle Plan and Report	N / A	1/28/2004	<a href="http://xena/edm04b/10099/2524656.tif">http://xena/edm04b/10099/2524656.tif</a>
N / A	WinRAP Software Development Testing and Validation Plan	N / A	8/20/2003	<a href="http://xena/edm04b/10099/2524661.tif">http://xena/edm04b/10099/2524661.tif</a>
EDF-INEEL-99001	Engineering Design File, Verification and Validation Plan for RSAC-6	N / A	3/22/1999	
INEEL/EXT-01-00540	The Radiological Safety Analysis Computer Program (RSAC-6) Users' Manual	1	1/1/2001	
N / A	Software V & V Plans	N / A		<a href="http://edms/pls/edms/toto.dmx_31?f_doc=69104">http://edms/pls/edms/toto.dmx_31?f_doc=69104</a>
N / A	Software Test Record #13	N / A	2/5/2001	
N / A	Version Management with CVS, for CVS 1.9	N / A	© 1992, 1993	
N / A	Software Test Record #1	N / A	5/1/1999	
N / A	Software Test Record #2	N / A	1/9/2001	
N / A	Software Test Record #3	N / A	5/1/1999	
N / A	Software Test Record #4	N / A	5/5/1999	
N / A	Software Test Record #5	N / A	5/5/1999	
N / A	Software Test Record #7	N / A	1/6/2001	
N / A	Software Test Record #10	N / A	1/16/2001	
N / A	Software Test Record #8	N / A	4/20/1999	
N / A	Software Test Record #9	N / A	4/7/1999	

Identifying No.	Title	Rev. #	Date	URL
N / A	Software Test Record #11	N / A	1/16/2001	
N / A	RSAC History	N / A	7/6/2004	<a href="http://www.inel.gov/rsac/history/rsachist.asp">http://www.inel.gov/rsac/history/rsachist.asp</a>
N / A	List of RSAC users	N / A	7/12/2004	
N / A	Program Updates	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/updates/">http://www.inel.gov/rsac/updates/</a>
N / A	What's New	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/whatsnew.asp">http://www.inel.gov/rsac/whatsnew.asp</a>
N / A	RSAC History	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/history/rsachist.asp">http://www.inel.gov/rsac/history/rsachist.asp</a>
N / A	Product Information	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/productinfo/">http://www.inel.gov/rsac/productinfo/</a>
N / A	Downloads	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/downloads/downloads.asp">http://www.inel.gov/rsac/downloads/downloads.asp</a>
N / A	Suggestions	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/suggestions.asp">http://www.inel.gov/rsac/suggestions.asp</a>
N / A	Bug Reporting	N / A	7/12/2004	<a href="http://www.inel.gov/rsac/bugreporting.asp">http://www.inel.gov/rsac/bugreporting.asp</a>
IAS04495	Independent Audit of Software Quality Assurance Legacy Program/applications	N / A	6/17/2004	
	DOE Implementation Plan for Defense Nuclear Facilities Safety Board (DNFSB) Recommendation 2002-1, <i>Quality Assurance for Safety-Related Software</i>		2/28/2003	

Identifying No.	Title	Rev. #	Date	URL
CRAD - 4.2.4.1	Assessment Criteria and Guidelines for Determining the Adequacy of Software Used in the Safety Analysis and Design of Defense Nuclear Facilities	3	10/24/2003	<a href="http://www.eh.doe.gov/sqa/assessments/crads/SQA%20CRAD%204.2.4.pdf">http://www.eh.doe.gov/sqa/assessments/crads/SQA%20CRAD%204.2.4.pdf</a>
MCP-550	SOFTWARE MANAGEMENT	7	7/1/2003	
MCP-3039	ANALYSIS SOFTWARE CONTROL	4	7/1/2003	
INEEL-2000-003	Radiological Engineering Software Management Plan	N / A	11/13/2000	
N / A	RSAC information contained in the Enterprise Archicture database	N / A	N / A	<a href="http://itarchitecture:1081/earch/start.html">http://itarchitecture:1081/earch/start.html</a> - Enterprise Archicture home page URL

## People Contacted

Name	Title	Email address
B.J. Schrader	SR ADVISORY ENGR/SCI	BSCHRADE@inel.gov
R.K. Smith	ADVISORY ENGR/SCI	RS6@inel.gov
T. Hathaway	ADVISORY ENGR/SCI	<u>THATHAWA@inel.gov</u>
M.H. Johnson	PR BUSINESS/OPS SPEC	<u>MHJ@inel.gov</u>
R. C. Aeschbacher	SR OFFICE SPECIALIST	AESCRC@inel.gov
P.I. Loo	STAFF ENGINEER/SCIENTIST	SIP@inel.gov

## Software Quality Assurance Data

Not applicable

## Biography of Assessor

Bob Blyth is the NE-ID designated subject matter expert for software quality assurance. He has certifications as a Professional Engineer, Quality Manager, Professional Mediator and Lead Quality Assurance Auditor. He is currently the Program Manager for the National Spent Nuclear Fuel program Quality Assurance Program. In the last 5 years he has lead 5 major QA audits of DOE facilities and participated as an auditor in 12 other major DOE QA program audits.