Conf-920428--5

WSRC-MS--91-413

DE92 013069

CERTIFICATION PROCESS OF SAFETY ANALYSIS AND RISK MANAGEMENT AT THE SAVANNAH RIVER SITE (U)

by

M. J. Ades[1], H. Toffer[2], C. J. Lewis[2], and R. D. Crowe[2]

[1] Westinghouse Savannah River Company
Savannah River Site
Aiken, SC 29808

[2] Westinghouse Hanford Company
Mail Stop HO-38
P. O. Box 1970
Richland, WA 99352

A paper proposed for presentation at the
1992 Simulation MultiConference - Visualization, Validation, and Verification of Computer Simulations
Orlando, FL
April 6-9, 1992

and for publication in the proceedings

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

This article was prepared in connection with work done under Contract No. DE-AC09-89SR18035 with the U. S. Department of Energy. By acceptance of this article, the publisher and/or recipient acknowledges the U. S. Government's right to retain a nonexclusive, royalty-free license in and to any copyright covering this article, along with the right to reproduce and to authorize others to reproduce all or part of the copyrighted article.

CERTIFICATION PROCESS OF SAFETY ANALYSIS AND RISK MANAGEMENT COMPUTER CODES AT THE SAVANNAH RIVER SITE

by

M. J. Ades[1], H. Toffer[2], C. J. Lewis[2], and R. D. Crowe[2]

[1] Westinghouse Savannah River Company
Savannah River Site
Aiken, SC 29808

[2] Westinghouse Hanford Company
Mail Stop HO-38
P. O. Box 1970
Richland, WA 99352

INTRODUCTION

The commitment by Westinghouse Savannah River Company (WSRC) to bring safety analysis and risk management codes into compliance with national and sitewide software quality assurance requirements necessitated a systematic, structured approach. As a part of this effort, WSRC, in cooperation with the Westinghouse Hanford Company, has developed and implemented a certification process for the development and control of computer software. The process followed complies with the basic requirements outlined in References 1 and 2, and the sitewide quality assurance manual[3]. Safety analysis and risk management computer codes pertinent to reactor analyses were selected for inclusion in the certification process. As a first step, documented plans were developed for implementing verification and validation of the codes, and establishing configuration control. User qualification guidelines were determined. The plans were followed with an extensive assessment of the codes with respect to certification status. Detailed schedules and work plans were thus determined for completing certification of the codes considered.

Although the software certification process discussed is specific to the application described, it is sufficiently general to provide useful insights and guidance for certification of other software.

CODE CERTIFICATION

Since most of the computer codes considered are existing software, the process of certification must be tailored to meet each particular situation. For these codes, certification is the act of determining, verifying, validating, and attesting in writing to the qualification of software used in important applications. Assurance is provided that the software has been properly reviewed and documented for verification and validation, configuration control has been established, and user qualifications have been determined. Certified software would thus be in compliance with sitewide applicable software quality assurance requirements.

The software certification process as applied to the safety analysis and risk management codes consists of writing a specific verification and validation plan, a configuration control plan, and user qualifications. Subsequent to the plans, applicable literature is reviewed to establish the verification and validation and configuration control status for each code. Compliance with the various action items noted in the plans is documented. Resource commitment plan schedules are drawn up for each individual code to complete certification in accordance with software quality assurance requirements. Since the plans and the assessment call for extensive resource commitment, careful review and signoff by appropriate management levels had to be performed.

THE PLANS

Verification and Validation Plan

The purpose of the verification and validation (V&V) plan document is to establish the key elements of the V&V activity, to identify the necessary documentation, and to suggest review processes. Verification includes the process that establishes correctness of theory, proper coding, and interaction of code modules to process information as required. Validation is the process that establishes how well a computer code meets specified requirements such as reproducing experimental data obtained from specific measurements or from controlled experiments in operating facilities, from normal operations or from benchmarking activities. Benchmarking refers to the process of evaluating the performance of one computer code relative to another code or relative to an exact solution. Since the codes considered were mostly existing codes, the V&V plan stresses verification and validation by demonstrating successful application of the code to predict measurements, benchmark analyses, conceptual solutions, and code-to-code comparisons. Such an approach is consistent with industry accepted practices and guidance provided by national standards. For new codes, the V&V follows the established software quality assurance (SQA) procedures developed.

The plan contains the following important sections: compliance to SQA requirements, code descriptions, and an action matrix. The SQA requirements section discusses how the plan meets department and site requirements.

The brief code descriptions cover the theory, methods of solution, types of problems solved, input information, and historic development of the code.

An action matrix for the V&V plan is then developed (Figure 1). The matrix identifies the different types of information that needs to be gathered for each code to establish documented code verification and validation. The matrix lists and defines the activities that are required for V&V and also serves as a useful tracking tool for monitoring the completion of these activities.

The matrix includes the names of the various codes and a series of action items. The action items are divided into groups. The BASICS group covers common requirements for each code. The topics in the THEORY group pertain to confirmation of code verification. Items included under EXPERIMENTS AND BENCHMARKS focus on the validation effort. Finally, the CONCLUSIONS category establishes the completion of the tasks identified in the verification and validation action matrix (V&VAM).

The boxes shown on the form are filled in as items are completed. Not all the boxes are relevant for each computer code. For example, the Probabilistic Risk Assessment (PRA) codes will not have experimentally validated results.

For all codes, there are two ways for a box to be completed and checked off: A box can be either completed by performing the tasks identified for each box or by establishing that this specific box is not relevant for the particular code. In either case, this information is to be included in the documentation for the particular action matrix box.

Detailed instructions and guidance are provided to establish sign-off of each of the boxes. An example is provided below.

User Manuals in Place

Is adequate documentation in place for a person to use the code?

Adequate user documentation should exist so that a person with technical familiarity and a background in the appropriate field would be able to gain sufficient knowledge of the computer code's input parameters to execute calculations on a controlled code version. A knowledge of the specific computer operating system is assumed.

Configuration Control Plan

The purpose of the configuration control plan is to identify the actions that need to be taken to produce a certified version of the code and maintain it for the qualified and approved users. Configuration control, as defined in the plan document, allows to manage and control computer software by providing certified code versions to the qualified users. Configuration control prevents unauthorized changes or use of the software coding and establishes a method of handling discrepancies, including correction of errors. It also governs retention of supporting documentation such as certification records, user documentation, installation instructions, benchmark testing results, and V&V records.

The plan contains an action matrix providing steps to be taken and required documentation to achieve configuration control. Furthermore, the plan includes a checklist of procedures needed to operate and maintain a code under configuration control. The specialized procedures would address topics such as transferring a code into code control, operating under code control, archiving responsibilities, backup disaster control, error identification and correction procedures, and periodic code testing with test problems.

The action matrix for achieving configuration control is shown in Figure 2. The matrix identifies the different types of activities that need to be completed. The matrix serves as a useful tracking tool for monitoring the completion of tasks. In the matrix, the names of the various codes are listed, as well as a series of action items required under configuration control. The activities are grouped by topics. The first group covers activities in the user community. The topics in the second group pertain to activities in the computer administration organization and user community. The last category deals with the completion process for configuration control.

Not all the boxes are relevant for each computer code. For example, special shell scripts may be required for only a few codes while others may use a standard script. Consequently, each box may be completed and checked off by performing the special tasks identified for each box, referencing why a general situation is applicable for the particular code, or by explaining why the box is not applicable to the specific code. In every case, this information is to be included in the documentation for the particular box. Each item in the action matrix is discussed in detail. As an example

Code Proprietors

Has a code proprietor been identified for a specific code?

A technically knowledgeable person must be assigned to each computer code. This individual would be a member of the user community. The code proprietor would be an experienced individual familiar with the specific computer code. The person must be able to perform the functions explained under the proprietor responsibilities, as well as meet the qualifications in the user qualification sections. The proprietor must be intimately familiar with software quality control requirements and procedures. Considerations should be given for identifying a backup proprietor to ensure continuity of code-related expertise.

When the "Code Proprietors" box is checked off as completed, supporting documentation related to the code proprietor, as defined above, must be available.

User Qualification

The third part of the certification plan consists of establishing user qualifications. A user of a particular computer code will fall into one of three broad categories: the novice or apprentice user, the experienced or cognizant user, or the expert or proprietor for the code. Personnel are designated by management to these levels based on the individual's qualifications and training.

The apprentice user is a person who is in training to use the code and would work under the tutelage of a cognizant user or proprietor. He or she is trained in software quality assurance. This individual generally is in the process of learning to understand the purpose of the calculations, how to follow the input logic and select certain variables, and understand all or specific results from the output.

The apprentice user may have a significant understanding of the physical processes being evaluated by the code. He or she may perform independent noncritical analyses, but code input and output must be checked by a more experienced cognizant user. The apprentice user does not perform critical parameter analyses.

The cognizant user is a person with experience in executing the computer code. As such, the person understands the purpose of the code and is familiar with the code's input requirements and output. The cognizant user is aware of the quality assurance requirements and procedures used for documenting computer calculations required for critical applications, and is qualified to perform such applications.

The cognizant user is sufficiently familiar with the code input and output to perform independent analyses, but may not possess sufficient understanding of the software coding to make changes to the code. He or she must be familiar enough with the code to identify when it is not functioning properly. In the case of an error, the cognizant user notifies the code proprietor or custodian of a possible error condition and requests the proprietor to make any necessary changes.

The code proprietor has an in-depth understanding of the software's coding structure and how the code performs. The code proprietor understands the models used, the logical flow of data through the code, and the numerical methods used by the code, both with respect to theory and range of applicability. The code proprietor is sufficiently familiar with the code to make coding changes, verify the programming, and validate the code with test problems. The code proprietor is aware of the quality assurance requirements for verification, validation, and configuration control. The code proprietor maintains a notebook in which code evolution is tracked. Changes to the code and the impact of these changes are recorded. Validation calculations following alterations to the code are performed by the code proprietor. Test results and validation records are appropriately maintained.

In addition, the code proprietor is considered to be a cognizant user. There is only one code proprietor and a code proprietor backup individual assigned for each computer code.

ASSESSMENT REPORT

The assessment report creates a baseline for the certification effort for a specific computer code. In particular, it establishes applicability and depth of existing technical documentation in the area of code verification and validation, and configuration control. The results of the assessment report serve as an outline for the additional effort required to bring each code into full compliance with software quality assurance requirements.

As part of the assessment activity, the status of each code is derived from two sources. First, each code proprietor provides his or her estimation of the code status and accessibility, and quality of various documentation pertaining to the certification efforts. Second, an extensive review of the Savannah River Site and offsite technical documentation applicable to the certification of each computer code is performed. The documentation review serves to supplement any documentation not identified in the first step. To implement this review, library searches are made, keying on the computer codes of interest. Relevant documents are reviewed and a record of the document review is assembled. A review form, structured according to the subsections in the verification and validation plan action matrix, is used. The form is completed for each document reviewed and an assessment is made to determine how well a document satisfies a particular action matrix requirement. The compiled information is then entered into a database (dBASE III Plus).

The information compiled in the database is used to assess how well the action matrix items are met. Multiple shadings indicate the status of compliance. A blacked-out box indicates that the information on hand is sufficient for closeout. Shadings of grey indicate partial completion. For each code, descriptive information is provided as to what additional work is needed.

The technical information database developed as part of the assessment activity represents a unique resource for code certification. For a specific code, the database can be queried as to how many documents address any given topic in the verification and validation action matrix. Documents containing test problems can be readily identified, reports suitable for training packages can be selected, and summary status reports on how well existing documentation supports the verification and validation effort can be created.

CONCLUSIONS

A systematic approach has been developed to bring safety analysis and probabilistic risk assessment codes under configuration control. The approach consists of developing and using documented plans for code verification and validation, and code configuration control, and establishing guidelines for user qualification. The plans and guidelines contain detailed instructions for what is essential to achieve certification of existing codes. As part of the certification effort, an assessment of the present status of the codes and the pertinent code documentation is made to establish a baseline for completing code certification.

Although the process described applies to safety and risk management codes at the Savannah River Site, the material provided is general enough to be applied to other situations that involve the challenges of certifying existing computer codes.

ACKNOWLEDGEMENT

The information contained in this article was developed during the course of work done under Contract No. DE-AC09-89SR18035 with the U. S. Department of Energy.

REFERENCES

1. "Quality Assurance Requirement of Computer Software for Nuclear Applications", ANSI/ASME NQA-1 Part 2.7, American Society of Mechanical Engineers.

2. "Standard for Software Quality Assurance Plans", ANSI/IEEE Std-730-1984, Institute of Electrical and Electronics Engineers.

3. Savannah River Site, Quality Assurance Manual,1Q, (1990).

	Code Number								
	1	2	3	4	5	6	7	8	
Software Development Plan	■	■	□	■	□	□	■	□	Basics
User Manuals in Place	■	■	■	▨	▨	■	▨	■	
Configuration Control Plan	■	■	■	■	■	■	■	■	
Code Portability	□	□	▨	▨	▨	■	▨	▨	
Appropriate Theory	■	▨	■	■	▨	■	▨	■	Theory
Theory Documented	■	▨	■	■	▨	■	▨	■	
Coding Consistency	■	▨	■	■	▨	■	▨	■	
Theory Verified Conceptually	■	▨	■	■	▨	■	▨	■	
Theory Verified by Experiments	■	▨	■	■	▨	■	▨	■	
Theory Documentation Adequate	■	▨	■	■	▨	■	▨	■	
Tests in Experimental Facilities	■	▨	▨	■	▨	□	▨	▨	Experiments
Tests in Operating Facilities	■	▨	▨	■	▨	■	▨	▨	
Data from Operating Facilities	■	▨	▨	■	▨	▨	▨	▨	
Test Data Documented	■	▨	▨	■	▨	▨	▨	▨	
Appropriate Data Quality	■	▨	▨	■	▨	▨	▨	▨	
Validation Performed	■	▨	▨	■	▨	▨	▨	▨	
Validation Documentation Adequate	■	▨	▨	■	▨	▨	▨	▨	
Benchmark Requirements Identified	■	▨	▨	■	▨	□	▨	■	Benchmarks
Similar Code Comparison	■	▨	▨	■	▨	□	▨	▨	
Exact Solution Comparison	■	▨	▨	■	▨	□	▨	▨	
Industry Benchmark Comparison	■	▨	▨	■	▨	□	▨	▨	
Comparisons Documented	■	▨	▨	■	▨	□	▨	▨	
Benchmark Documentation Adequate	■	▨	▨	■	▨	□	▨	▨	
RequirementsVerification Review	■	▨	■	■	▨	□	▨	▨	Conclusions
Verification Completed	■	▨	■	■	▨	▨	▨	▨	
Standard Set of Test Problems	▨	▨	▨	▨	▨	▨	▨	▨	
Validation/Benchmarking Review	■	▨	▨	■	▨	▨	▨	▨	
Validation/Benchmarking Completed	■	▨	▨	■	▨	▨	▨	▨	

KEY

□ Actions to complete item have not been identified

▨ Actions to complete item have been identified or are in progress

■ Actions on this item are completed

Figure 1. Verification and Validation Action Matrix

	Code Proprietors	Proprietor's Code Log Book	Cognizant User Qualifications	User Manual Availability	Training Package	Code Source Listing Location	Test Problem And Results Location	Controlled Tracking System	System Dependent Code Documentation	Code Interaction Impact	Range of Applicability	Back Up Versions	Shell Scripts and Banners	Configuration Control Review.	Configuration Control In Place.
Safety Analysis															
Code Number															
1	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
2	■	□	□	■	□	■	■	■	■	■	■	■	■	■	■
3	■	□	□	■	□	□	□	□	□	□	□	□	□	□	□
4	■	□	□	■	□	□	□	▨	□	▨	□	□	□	▨	□
5	■	□	□	■	□	□	■	□	□	□	□	□	□	▨	□
6	■	■	□	■	□	□	□	□	▨	□	□	▨	□	▨	□
7	■	□	□	■	□	□	□	□	▨	□	□	□	□	□	□
8	■	□	□	■	□	□	■	□	▨	□	□	□	□	□	□
PRA															
Code Number															
1	■	□	□	■	□	□	□	□	▨	□	□	▨	□	□	□
2	■	□	□	■	□	□	▨	□	□	□	▨	□	□	□	▨
3	■	□	□	■	□	□	▨	□	□	□	□	□	□	□	□
4	■	□	□	■	□	□	▨	□	□	▨	□	□	□	□	□
5	■	□	□	■	□	□	▨	□	□	□	□	□	▨	□	□

KEY

□ Actions to complete item have not been identified

▨ Actions to complete item have been identified or are in progress

■ Actions on this item have been completed

Figure 2. Configuration Control Action Matrix

END

DATE FILMED

6/24/92