ELSEVIER

Reliability Engineering and System Safety 69 (2000) 421–427

RELIABILITY ENGINEERING & SYSTEM SAFETY

www.elsevier.com/locate/ress

Software quality assurance in the 1996 performance assessment for the Waste Isolation Pilot Plant

G.K. Froehlich[a,*], H.C. Ogden[b], K.A. Byle[c]

[a]*Regulatory Compliance Department, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185, USA*
[b]*Technical Integration Department, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185, USA*
[c]*Quality Assurance Department, Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87185, USA*

Abstract

The US Department of Energy (DOE) Waste Isolation Pilot Plant (WIPP), located in southeast New Mexico, is a deep geologic repository for the permanent disposal of transuranic waste generated by DOE defense-related activities. Sandia National Laboratories (SNL), in its role as scientific advisor to the DOE, is responsible for evaluating the long-term performance of the WIPP. This risk-based Performance Assessment (PA) is accomplished in part through the use of numerous scientific modeling codes, which rely for some of their inputs on data gathered during characterization of the site. The PA is subject to formal requirements set forth in federal regulations. In particular, the components of the calculation fall under the configuration management and software quality assurance aegis of the American Society of Mechanical Engineers (ASME) Nuclear Quality Assurance (NQA) requirements. This paper describes SNL's implementation of the NQA requirements regarding software quality assurance (SQA). The description of the implementation of SQA for a PA calculation addresses not only the interpretation of the NQA requirements, it also discusses roles, deliverables, and the resources necessary for effective implementation. Finally, examples are given which illustrate the effectiveness of SNL's SQA program, followed by a detailed discussion of lessons learned. Published by Elsevier Science Ltd.

Keywords: Software quality assurance; Software QA; Software testing; Software verification; Software validation; Software life-cycle; Quality assurance; Performance assessment; Waste Isolation Pilot Plant; Transuranic waste; Radioactive waste

1. Introduction

The Waste Isolation Pilot Plant (WIPP) is a deep geologic repository located in southeast New Mexico, which has been licensed for the permanent disposal of transuranic waste generated by US Department of Energy (DOE) defense-related activities [1]. The scientific advisor to DOE, Sandia National Laboratories (SNL), is responsible for evaluating the long-term (10,000-year) performance of the WIPP. This risk-based Performance Assessment (PA) is accomplished in part through the use of numerous scientific modeling codes, which rely for some of their inputs on data gathered during characterization of the site. These calculations depend in large part on computer codes used to simulate processes within the repository system, as well as transport and retardation of radionuclides from the repository through surrounding hydrogeologic formations, and simulation of releases due to possible future human intrusion into the repository. Probabilistic modeling and analysis codes are also used to characterize both the uncertainty of physical parameters and the unpredictability of future events. In such a regulatory environment as nuclear-waste disposal, SNL's work must be held to high standards of accountability. For SNL, this means that the PA codes must comply with rigorous software quality-assurance (SQA) requirements.

The origins of the formal SQA requirements are given, and the status of SQA at the time the requirements were imposed is described. Interpretation of the requirements and their applicability to WIPP PA is discussed. This is followed by a detailed description of SNL's implementation of SQA, including a description of procedures and roles. Examples are cited which demonstrate the effectiveness of our implementation of SQA. Finally, there is a discussion of lessons learned.

2. Origin of SQA requirements

In 1992, the Land Withdrawal Act [2] named the US Environmental Protection Agency (EPA) as the regulator for WIPP. As such, EPA became responsible for developing disposal regulations, and for certifying the long-term safety of the repository. In late 1993, federal regulation 40CFR191

* Corresponding author. Fax: + 1-505-844-8558.
E-mail address: gkfroeh@sandia.gov (G.K. Froehlich).

0951-8320/00/$ - see front matter Published by Elsevier Science Ltd.
PII: S0951-8320(00)00041-7

[3] set forth the disposal regulations and release limits. In effect, this regulation outlined *what* needed to be done to demonstrate compliance with the release limits, without specifying *how* to do it. Then, in early 1996, 40CFR194 [4] established criteria for demonstrating compliance with 40CFR191, in effect specifying *how* to do so. This latter regulation invoked the American Society of Mechanical Engineers' (ASME) Nuclear Quality Assurance (NQA) Standards, i.e. ASME NQA-1-1989 edition, NQA-2a-1990 addenda (Part 2.7) to ASME NQA-2-1989 edition, and ASME NQA-3-1989 edition [5].

The NQA standards were originally developed by the ASME at the request of the American National Standards Institute (ANSI). ASME formed a committee on Nuclear Quality Assurance in 1975, which developed NQA-1 and NQA-2 from the ANSI/ASME N45.2 series of standards and initially issued them in 1979. The NQA standards define QA program requirements for siting, design, construction, operation, and decommissioning of nuclear facilities. The standards also define QA requirements for planning and executing tasks during fabrication, construction, modification, repair, maintenance, and testing of the systems, components, and structures of nuclear facilities. Part 2.7 defines quality assurance requirements for the development, procurement, maintenance, and use of computer codes. The NQA standard is recognized as the *de facto* standard for the nuclear industry, largely because it is maintained by an active organization that periodically updates the standard to reflect both state-of-the-art technologies and real world experience. This fact, and the maturity of the standard, led EPA to select it over other standards that were considered.

3. Applicability of NQA to WIPP

As described above, the NQA standards were developed for application to nuclear facilities. While WIPP qualifies as a nuclear facility in a certain sense, the PA software clearly has no real-time performance requirements. In a nuclear power plant, there are time-critical issues for software, such as restart capability following a software "crash". Another example is unintended functionality that, by itself or in combination with other unintended functionality, could degrade the entire system (possibly with serious consequences). PA codes, on the other hand, do not share these concerns. If a code "crashes", the cause is found, and the job is merely resubmitted. Unintended functionality, while undesirable, at most results in an incorrect result, which will be discovered in a subsequent review; there is no real-time safety issue. Using this rationale, we interpreted the requirements of Part 2.7 for application to WIPP PA codes. It is important to note that our interpretation was not unilateral—concurrence was obtained from both the customer (DOE) and the regulator (EPA), who were engaged early in the interpretation process.

The majority of the WIPP software that required qualification to Part 2.7 standards was already either partially or completely written before promulgation of 40CFR194. These codes had undergone various degrees of earlier SQA, but in all cases, the earlier QA did not fulfill Part 2.7 requirements (primarily in the area of documentation). SNL applied Part 2.7, Section 10.2, "Software Developed Not Using This Standard", to address this condition. Section 10.2 essentially permits the necessary software requirements, testing, and user documentation to be created after most of the development phases have been completed. Furthermore, Section 10.2 requires that once Part 2.7 is implemented, configuration management and change control per Part 2.7 be implemented as well [6]. Evaluation of our earlier SQA activities showed that it would be more effective to re-do (versus supplement) qualification for previously qualified computer codes, to ensure consistency and to enable uniform application of testing tools and methods.

4. Establishment of existing SQA program

Promulgation of 40CFR191 led to development of the SNL Quality Assurance Program Document, Rev. R, 7/31/95, which was the basis for the SNL QA program that was in place for the Compliance Certification Application (CCA) submitted to EPA [7]. Prior to that time, the SNL SQA program was based on good scientific practice rather than regulatory requirements. Based on the nature of the software and its intended use, a three-level approach for software qualification was developed [8]. The levels were progressive, and were defined as X (eXperimental), C (Candidate), and A (Adjudicated). An X level code was one that was still in the developmental stages. At the X level, conceptual models were being implemented for evaluation. Testing was conducted, but not formally documented. The testing was conducted and reviewed by the code developer. At this stage the code team consisted of the code sponsor, responsible for guiding the code through the defined QA process, and the code consultant, who was responsible for the theoretical basis (conceptual models, physics, etc.) of the code. In cases where the software was a utility code, this was the same person.

At the point where the conceptual model(s) represented by a code were determined to be applicable for WIPP, the code moved to the next level of QA (level C). At this stage the code was a candidate for full quality assurance. A review team (one or more individuals, depending on the complexity of the software being reviewed) was assigned to the code, and a code-qualification package was assembled for review. When the code was determined to be stable and ready for final qualification, the code and its accompanying documentation (test cases, user's manual, theoretical manual) were assembled for consideration as an A level package. Rather than following specific criteria, as the codes were of many different types (utility, modeling, etc.), code sponsors and