U.S. NUCLEAR REGULATORY COMMISSION
OFFICE OF NUCLEAR REGULATORY RESEARCH

August 1996
Division 1
Draft DG-1057

DRAFT REGULATORY GUIDE

Contact: J. Kramer (301)415-5891

DRAFT REGULATORY GUIDE DG-1057

SOFTWARE UNIT TESTING FOR DIGITAL COMPUTER SOFTWARE USED IN SAFETY SYSTEMS OF NUCLEAR POWER PLANTS

A. INTRODUCTION

In 10 CFR Part 50, "Domestic Licensing of Production and Utilization Facilities," paragraph 55a(a)(1) requires, in part,[1] that systems and components be designed, tested, and inspected to quality standards commensurate with the safety function to be performed. Criterion 1, "Quality Standards and Records," of Appendix A, "General Design Criteria for Nuclear Power Plants," to 10 CFR Part 50 requires, in part,[1] that a quality assurance program be established and implemented in order to provide adequate assurance that systems and components important to safety will satisfactorily perform their safety functions. Appendix B, "Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants," to 10 CFR Part 50 describes criteria that a quality assurance program for systems and components that prevent or mitigate the consequences of postulated accidents must meet. In particular, besides the systems and components that directly prevent or mitigate the consequences of postulated accidents, the criteria of Appendix B also apply to all activities affecting the

DF02
D/1

[1]In this draft regulatory guide, many of the regulations have been paraphrased; see 10 CFR Part 50 for the full text.

safety-related functions of such systems and components as designing, purchasing, installing, testing, operating, maintaining, or modifying. A specific requirement is contained in 10 CFR 50.55a(h), which requires that reactor protection systems satisfy the criteria of IEEE Std 279-1971, "Criteria for Protection Systems for Nuclear Power Generating Stations."[2] Paragraph 4.3 of IEEE Std 279-1971[3] states that quality of components is to be achieved through the specification of requirements known to promote high quality, such as requirements for design, inspection, and test.

Many of the criteria in Appendix B to 10 CFR Part 50 contain requirements closely related to testing activities. Criterion I, "Organization," requires the establishment and execution of a quality assurance program. Criterion II, "Quality Assurance Program," requires, in part, that the program take into account the need for special controls, processes, test equipment, tools, and skills to attain the required quality, as well as the need for verification of quality by inspection and test. Criterion III, "Design Control," requires, in part, that measures be established for verifying and checking the adequacy of design, such as by the performance of a suitable testing program, and that design control measures be applied to items such as the delineation of acceptance criteria for inspections and tests. Criterion V, "Instructions, Procedures, and Drawings," requires activities affecting quality to be prescribed by documented instructions, procedures, or drawings of a type appropriate to the circumstances and that these activities be accomplished in accordance with these instructions, procedures, or drawings. Criterion V further requires that instructions, procedures, and drawings include appropriate quantitative or qualitative acceptance criteria for determining that important activities have been satisfactorily accomplished. Criterion XI, "Test Control," requires establishment of a test program to ensure that all testing required to demonstrate that structures, systems, and components will perform satisfactorily in service is identified and performed in accordance with written test procedures that incorporate the requirements and acceptance limits contained in applicable design documents. Test procedures must include provisions for ensuring that all prerequisites for the given test have been met, that adequate test

---

[2]Revision 1 of Regulatory Guide 1.153, "Criteria for Safety Systems," endorses IEEE Std 603-1991, "Criteria for Safety Systems for Nuclear Power Generating Stations," as a method acceptable to the NRC staff for satisfying the NRC's regulations with respect to the design, reliability, qualification, and testability of the power, instrumentation, and control portions of the safety systems of nuclear power plants.

[3]IEEE publications may be obtained from the IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854.

instrumentation is available and used, and that the test is performed under suitable environmental conditions. Criterion XI also requires that test results be documented and evaluated to assure that test requirements have been satisfied. Finally, Criteria VI, "Document Control," and XVII, "Quality Assurance Records," provide for the control of the issuance of documents, including changes thereto, that prescribe all activities affecting quality and provide for the maintenance of sufficient records to furnish evidence of activities affecting quality. The latter requires test records to identify the inspector or data recorder, the type of observation, the results, the acceptability of the results, and the action taken in connection with any deficiencies noted.

This regulatory guide endorses ANSI/IEEE Std 1008-1987, "IEEE Standard for Software Unit Testing,"[3] as amended in the Regulatory Position. IEEE Std 1008-1987 describes a method acceptable to the NRC staff for complying with parts of the NRC's regulations for promoting high functional reliability and design quality in software used in safety systems.[4] In particular, the method is consistent with the previously cited General Design Criteria and the criteria for quality assurance programs of Appendix B as they apply to software unit testing. The criteria of Appendices A and B apply to systems and related quality assurance processes, and if those systems include software, the requirements extend to the software elements.

In general, information provided by regulatory guides is reflected in the Standard Review Plan (NUREG-0800, currently under revision), which is used by the Office of Nuclear Reactor Regulation in the review of applications to construct and operate nuclear power plants. This regulatory guide will apply to Chapter 7 of that document.

Regulatory guides are issued to describe and make available to the public such information as methods acceptable to the NRC staff for implementing specific parts of the NRC's regulations, techniques used by the staff in evaluating specific problems or postulated accidents, and guidance to applicants. Regulatory guides are not substitutes for regulations, and compliance with regulatory guides is not required. Regulatory guides are issued in draft form for public comment to involve the public in the early stages of developing the regulatory positions. Draft regulatory guides have not received complete staff review and do not represent official NRC staff positions.

---

[4]The term "safety systems" is synonymous with "safety-related systems." The General Design Criteria cover systems, structures, and components "important to safety." The scope of this draft regulatory guide is, however, limited to "safety systems," which are a subset of "systems important to safety."

The information collections contained in this draft regulatory guide are covered by the requirements of 10 CFR Part 50, which were approved by the Office of Management and Budget, approval number 3150-0011. The NRC may not conduct or sponsor, and a person is not required to respond to, a collection of information unless it displays a currently valid OMB control number.

B. DISCUSSION

The use of industry consensus standards is part of an overall approach to meeting the requirements of 10 CFR Part 50 when developing safety systems for nuclear power plants. Compliance with standards does not guarantee that regulatory requirements will be met. However, compliance does ensure that practices accepted within various technical communities will be incorporated into the development and quality assurance processes used to design safety systems. These practices are based on past experience and represent industry consensus on approaches used for development of such systems.

Software incorporated into instrumentation and control systems covered by Appendix B will be referred to in this regulatory guide as safety system software. For safety system software, software testing is an important part of the effort to achieve compliance with the NRC's requirements. Software engineering practices rely, in part, on software testing to meet general quality and reliability requirements consistent with Criteria 1 and 21 of Appendix A to 10 CFR Part 50, as well as Criteria I, II, III, V, VI, XI, and XVII of Appendix B.

The consensus standard, IEEE Std 1008-1987 (reaffirmed in 1993), defines a method for planning, preparing for, conducting, and evaluating software unit testing. The method described is consistent with the previously cited regulatory requirements as they apply to safety system software.

Current practice for the development of software for high-integrity applications includes the use of a software life cycle process that incorporates software testing activities, e.g., IEEE Std 1074-1991, "IEEE Standard for Developing Software Life Cycle Processes."[3] Software testing, including software unit testing, is a key element in software verification and validation activities, as indicated by IEEE Std 1012-1986, "IEEE Standard for Software Verification and Validation Plans,"[3] and IEEE Std 7-4.3.2-1993, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations." A common approach to software testing [NUREG/CR-6101, "Software Reliability and Safety in Nuclear Reactor Protection Systems" (November 1993);

NUREG/CR-6263, "High Integrity Software for Nuclear Power Plants: Candidate Guidelines, Technical Basis and Research Needs" (June 1995)][5] utilizes a three-level test program to help ensure quality in a complex software product or complex set of cooperating software products, i.e., unit-level testing, integration-level testing, and system-level testing such as system validation tests or acceptance tests. IEEE Std 1008-1987 delineates an approach to the unit testing of software that is based on the assumption of a larger context established by verification and validation (V&V) planning as well as general planning for the full range of testing activities to be applied. Therefore, software unit testing performed in accordance with IEEE Std 1008-1987 should be consistent with planning information established in V&V plans and higher-level software test plans, although that planning information is not within the scope of IEEE Std 1008-1987.

C. REGULATORY POSITION

The requirements[6] in ANSI/IEEE Std 1008-1987, "IEEE Standard for Software Unit Testing," provide an approach acceptable to the NRC staff for meeting the requirements of 10 CFR Part 50 as they apply to the unit testing of safety system software, subject to the provisions listed below. The appendices to IEEE Std 1008-1987 are not endorsed by this regulatory guide except as noted below. Appendix A to this standard provides guidance regarding the implementation of the software unit testing approach, and Appendix B to the standard provides context regarding software engineering information and testing assumptions that underlie the software unit testing approach.

To meet the requirements of 10 CFR 50.55a(h) and Appendix A to 10 CFR Part 50 as assured by complying with the criteria of Appendix B to 10 CFR Part 50 applied to the unit testing of safety system software, the following provisions are necessary and will be considered by the NRC staff in the review of applicants' submittals. (In this section, the cited criteria are in Appendix B to 10 CFR Part 50 unless otherwise noted.)

---

[5]Copies are available at current rates from the U.S. Government Printing Office, P.O. Box 37082, Washington, DC 20402-9328 (telephone (202)512-2249); or from the National Technical Information Service by writing NTIS at 5285 Port Royal Road, Springfield, VA 22161. Copies are available for inspection or copying for a fee from the NRC Public Document Room at 2120 L Street NW., Washington, DC; the PDR's mailing address is Mail Stop LL-6, Washington, DC 20555; telephone (202)634-3273; fax (202)634-3343.

[6]In this regulatory guide, the term "requirements" refers to requirements imposed by the NRC's regulations as well as to requirements that must be met in order to comply with a standard.

1. Criterion XI, "Test Control," requires that a test program be established to ensure that all testing required to demonstrate that systems and components will perform satisfactorily in service is identified and performed in accordance with written test procedures that incorporate requirements and acceptance limits contained in applicable design documents. Criterion I, "Organization," Criterion II, "Quality Assurance Program," Criterion III, "Design Control," Criterion V, "Instructions, Procedures, and Drawings," Criterion VI, "Document Control," and Criterion XVII, "Quality Assurance Records," contain requirements bearing on information associated with testing. IEEE Std 1008-1987, in section 1.1, mandates the use of the Test Design Specification and the Test Summary Report defined by ANSI/IEEE Std 829-1983, "IEEE Standard for Software Test Documentation." In addition, IEEE Std 1008-1987 either incorporates additional information into these two documents or indicates the need for additional documents. Regardless of whether these two documentation formats are used, the documentation used to support software unit testing (either documentation used directly in the software unit testing activity or documentation of the overall testing effort) must include information necessary to meet regulatory requirements as applied to software test documentation. As a minimum, this information includes:

   - Qualifications, duties, responsibilities, and skills required of persons and organizations assigned to testing activities,
   - Environmental conditions and special controls, equipment, tools, and instrumentation needed for the accomplishment of testing,
   - Test instructions and procedures incorporating the requirements and acceptance limits in applicable design documents,
   - Test prerequisites and the criteria for meeting them,
   - Test items and the approach taken by the testing program,
   - Test logs, test data, and test results,
   - Acceptance criteria,
   - Test records indicating the identity of the tester, the type of observation, the results and acceptability, and the action taken in connection with any deficiencies.

   Any of the above information items that are not present in the documentation selected to support software unit testing must be incorporated as additional items.

2. Criterion XI, "Test Control," requires establishment of a test program to ensure that all testing required to demonstrate that structures, systems, and components will perform satisfactorily in service is identified and performed in accordance with written test procedures that incorporate the requirements and acceptance limits contained in applicable design documents. The two aspects of test coverage that are particularly important for the unit testing of safety system software are coverage of requirements and coverage of the internal structure of the code.

2.1 Coverage of Requirements

For safety system software, those requirements identified as essential to the safety determination[7] must be tested. Section 3.2.2(5) of IEEE Std 1008-1987 suggests consideration of expected use of the unit in the determination of features to be tested. All features and associated procedures, states, state transitions, and associated data characteristics essential to the safety determination must be included in the testing.

2.2 Coverage of Internal Structure

Section 3.1.2(2) of IEEE Std 1008-1987 specifies statement coverage (covering each source language statement with a test case) as a criterion for measuring the completeness of the software unit testing activity. Statement coverage is a very weak criterion for measuring test completeness [See Beizer[8] and NUREG/CR-6263[9]]. Therefore, the staff does not endorse statement coverage as a sufficient coverage criterion for software unit testing. For safety system software, the unit test coverage criteria to be employed should be identified and justified.

3. Criteria VI, "Document Control," and XVII, "Quality Assurance Records," as well as 10 CFR 21.51, require the control and retention of documents and records affecting quality. In addition, Criterion III, "Design Control," requires that design changes be subject to design control measures

---

[7] Draft Regulatory Guide DG-1058, "Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants," is under development; it proposes to endorse IEEE Std 830-1993, "IEEE Recommended Practice for Software Requirements Specifications."

[8] Boris Beizer, *Software Testing Techniques*, Van Nostrand Reinhold, 1990.

[9] S. Seth et al., "High Integrity Software for Nuclear Power Plants: Candidate Guidelines, Technical Basis and Research Needs, NUREG/CR-6263, June 1995.

commensurate with those applied to the original design. Preservation of testing products is discussed in section 3.8.2(4) of IEEE Std 1008-1987. Since design control measures must be applied to acceptance criteria for tests and since some software testing materials are frequently re-used and evolve during the course of software development and software maintenance (for example, regression test materials), such materials should be configuration items under change control of a software configuration management system.[10] Additional information on this topic is provided in section A6 of Appendix A to IEEE Std 1008-1987.

4. Criterion III, "Design Control," imposes an independence requirement for the verification and checking of the adequacy of the design, requiring that those persons who verify and check be different from those who accomplish the design. Therefore, independence is an additional requirement for software unit testing. Those persons who establish the requirements-based elements for a software unit test must be different from those who designed or coded the software. The guidance in section A7 of Appendix A to IEEE Std 1008-1987 provides acceptable ways to meet this requirement for software unit testing. These persons must also be as proficient in software engineering as the designer or coder; Criterion II, "Quality Assurance Program," states that the program must provide for indoctrination and training of personnel performing activities affecting quality as necessary to ensure that suitable proficiency is achieved and maintained.

5. Section 1.3 of IEEE Std 1008-1987 references ANSI/IEEE Std 729-1983, "IEEE Standard Glossary of Software Engineering Terminology," and ANSI/IEEE Std 829-1983, "IEEE Standard for Software Test Documentation." These referenced standards should be treated individually.

   If a referenced standard has been incorporated separately into the NRC's regulations, licensees and applicants must comply with that standard as set forth in the regulation. If the referenced standard has been endorsed in a regulatory guide, the standard constitutes a method acceptable to the NRC staff of meeting a regulatory requirement as described in the regulatory

---

[10]NRC endorsement of IEEE Std 828-1990, "IEEE Standard for Software Configuration Management Plans," and IEEE Std 1042-1987, "IEEE Guide to Software Configuration Management," is proposed in Draft Regulatory Guide DG-1055 to provide guidance for general software configuration management plans and their implementation.

guide. If a referenced standard has been neither incorporated into the NRC's regulations nor endorsed in a regulatory guide, licensees and applicants may consider and use the information in the referenced standard, if appropriately justified, consistent with current regulatory practice.

D. IMPLEMENTATION

The purpose of this section is to provide information to applicants and licensees regarding the NRC staff's plans for using this regulatory guide. No backfitting is intended or approved in connection with the issuance of this proposed guide. Any backfitting that may result from applying this new guidance to operating plants would be justified in accordance with established NRC backfitting guidance and procedures.

This draft guide has been released to encourage public participation in its development. Except in those cases in which an applicant proposes an acceptable alternative method for complying with specified portions of the NRC's regulations, the method to be described in the active guide reflecting public comments will be used in the evaluation of submittals in connection with applications for construction permits, standard design certifications and design approvals, and combined operating licenses. The active guide will also be used to evaluate submittals from operating reactor licensees that propose modifications that go beyond the current licensing basis, if those modifications are voluntarily initiated by the licensee and there is a clear connection between the proposed modifications and this guidance. This guide will be used in conjunction with, and will eventually be reflected in, the Standard Review Plan, which is currently under revision.

REGULATORY ANALYSIS

1. PROBLEM

Because traditional and well-understood methods of design and quality assurance for developing and manufacturing hardware apply imperfectly to software design and development, additional guidance beyond standard approaches for hardware is necessary if the intent of the NRC's regulations is to be achieved. This problem is faced in many industries where computers and software are replacing traditional hardware-only instrumentation and control (I&C) designs. To this extent, the nuclear industry is not very different from any industry associated with high-consequence hazards. While additional guidance is necessary to help prevent failures of digital I&C safety systems, the potential benefits of these systems make their use highly desirable.

The use of computers and software in safety-related I&C designs is part of the larger problem of ensuring long-term safety of nuclear power plants, and it is seen as part of the solution as well. It is not just digital systems themselves that give rise to concerns about design verification and quality assurance, but the increase in the complexity of the system designs (including software) being attempted is also a factor. The NRC staff discussed its concerns in SECY 91-292, "Digital Computer Systems for Advanced Light Water Reactors,"[1] and again in parts of SECY 93-087, "Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactor (ALWR) Designs."[1] Subsequently, the staff sponsored studies that resulted in characterization of design factors, guidelines, technical bases, and practices generally considered appropriate for high-integrity software [See NUREG/CR-6101, "Software Reliability and Safety in Nuclear Reactor Protection Systems" (November 1993); NUREG/CR-6113, "Class 1E Digital Systems Studies" (October 1993); NUREG/CR-6263, "High Integrity Software for Nuclear Power Plants: Candidate Guidelines, Technical Basis and Research Needs" (June 1995); NUREG/CR-6293, "Verification and Validation Guidelines for High Integrity Systems" (March 1995); and NUREG/CR-6294, "Design

[1]Copies are available for inspection or copying for a fee from the NRC Public Document Room at 2120 L Street NW., Washington, DC; the PDR's mailing address is Mail Stop LL-6, Washington, DC 20555; telephone (202)634-3273; fax (202)634-3343.

Factors for Safety-Critical Software" (December 1994)[2]]. These studies identified software design control techniques that are currently being used in "best practice" software development efforts. While it is possible to simply list the criteria covered, the problem still remains of reaching a common understanding between the NRC staff and industry practitioners regarding what constitutes acceptable software engineering practice for safety systems. An agreed-upon collection of standards, established practice, and engineering techniques for software engineering methods is needed to complement the collection that already supports traditional hardware engineering methods, such as statistical quality control, testing standards, and quality assurance techniques used on design and manufacturing processes for hardware components.

Software testing is a key element in software verification and validation (V&V) activities and is fundamental to the assurance of software quality. Software unit testing, or component testing, is the first of several levels of testing typically applied as part of the overall testing effort. Its purpose is to examine consistency with software unit requirements and design and to discover implementation errors. Subsequent levels of testing focus on interactions among system components and consistency with system requirements. This is analogous to hardware testing in which components are tested prior to integration into assemblies. The importance of software unit testing for the development of high-integrity software is reflected by the fact that it is one task in the minimum set of V&V activities required for critical software by the consensus standard, IEEE Std 1012-1986, "IEEE Standard for Software Verification and Validation Plans." The use of software unit testing in the nuclear context is discussed in NUREG/CR-6101, NUREG/CR-6113, and NUREG/CR-6263. Since software unit testing is an important part of the overall testing effort for high integrity software, the use of a systematic and documented approach to this testing is an appropriate subject for staff review.

---

[2]Copies are available at current rates from the U.S. Government Printing Office, P.O. Box 37082, Washington, DC 20402-9328 (telephone (202)512-2249); or from the National Technical Information Service by writing NTIS at 5285 Port Royal Road, Springfield, VA 22161. Copies are available for inspection or copying for a fee from the NRC Public Document Room at 2120 L Street NW., Washington, DC; the PDR's mailing address is Mail Stop SS-6, Washington, DC 20555; telephone (202)634-3273; fax (202)634-3343.

2. ALTERNATIVE APPROACHES

Based on the studies cited above, an alternative was identified in which consensus in the software engineering community is sufficient to ensure widespread familiarity and reasonable levels of agreement. There are two additional approaches, taking no action and prescribing a detailed approach built from staff selections of best practice. In all, three approaches were identified:

1. Take no action,
2. Prescribe a detailed approach,
3. Endorse one or more software engineering standards.

The first alternative, taking no action, has the advantage that its initial cost is low since there are no "start-up" activities. It has flexibility, since each applicant would develop its own technical basis demonstrating that its digital system, and the quality assurance measures applied to it, complied with the NRC's regulations. However, this could have adverse effects on the level of staff effort required to conduct reviews or to ensure consistency among reviews. In the absence of an identified set of commonly accepted guidelines, practices, and quality assurance measures applicable to software engineering, NRC staff reviews would take longer and require greater effort to ensure consistent staff safety evaluations. From the applicant's perspective, this flexibility also has associated potential costs because there could be more unknowns associated with demonstrating compliance with regulations. Although the initial cost would apparently be low, taking no action could result in greater total costs, to both the NRC staff and the applicant, during the safety evaluation process.

Prescribing a detailed approach could have significant preliminary costs involved in formulating the approach and dealing with the public comment that would inevitably result. The staff has been reluctant in the past to take this approach. Such an approach places the staff in the position of designer, and compromises, or appears to compromise, the staff's independence as design safety reviewers; this is not the role of the regulator.

Consensus standards on software development are available and represent current good practice as agreed upon by responsible professionals in the software industry. Many organizations issuing standards, such as the IEEE and ANSI, provide for review and revision of standards at regular intervals to ensure the

consensus positions are current. In the United States, the Institute of Electrical and Electronic Engineers (IEEE), the American Nuclear Society (ANS), the Electronic Industry Association (EIA), the Instrument Society of America (ISA), the American Society of Mechanical Engineers (ASME), and the American National Standards Institute (ANSI) are the standards bodies issuing software engineering standards, computer standards, or related quality standards. In Europe, the International Electrotechnical Commission (IEC), the International Organization for Standardization (ISO), the International Atomic Energy Agency (IAEA), and the Comité Consultatif International Télégraphique et Téléphonique (CCITT) fill the same roles. The European Committee for Electrotechnical Standardization (CENELEC), a regional standardization body, adopts national and international standards. The overall collection of standards issued by these bodies covers a variety of subjects considered important to software quality. The standards specific to the nuclear industry issued by the U.S.-based organizations are, in general, compatible with the NRC's regulations. The software engineering standards issued by these organizations, notably the IEEE software engineering standards, are in general compatible with nuclear-industry-specific standards. Together, these standards form a framework for addressing the use of software within nuclear systems in the U.S. nuclear regulatory environment. Selected international standards can complement this framework; however, they tend to be organized differently and do not map directly into the U.S. industry-specific framework.

3. VALUES AND IMPACTS

Values and impacts for each of the three identified approaches are analyzed below. In this analysis, the probability of an alternative approach having a positive effect on software quality and the probability of the effect of software quality on the achievement of overall safety goals are not known quantitatively. Although the current state of the art does not support quantitative estimates, the results of poor software quality are evident in notable instances of software failure in various industries. Therefore, a positive correlation between software quality and the achievement of safety goals is inferred from the instances of negative effects of poor software quality, i.e., software quality is a necessary but insufficient factor in achieving safety goals. In the summary below, an impact is a cost in schedule, budget, or staffing or an undesired

property or attribute that would accrue from taking the proposed approach. Both values and impacts may be functions of time.

3.1 Alternative 1 - Take No Action

If no action is taken, retaining the status quo, the NRC staff will continue to receive applications or requests to review safety questions that are prepared with no clear guidance on what the staff considers to be acceptable methods of ensuring that safety-related software meets the requirements of the NRC's regulations. Each applicant would propose such measures as it deems necessary, and these measures would be reviewed by the staff and discussed with the applicant to reach a resolution acceptable to the staff and the applicant. This preserves the value of flexibility, but at the impact of additional staff and applicant effort and potential schedule extension. It is possible that a de facto staff position would develop from the accumulation of successful applications, but the amount of time and effort required to reach this condition is unknown.

Value - No value beyond the status quo

Impact - Schedule, budget, and staffing cost, to the staff and applicant, associated with regulatory uncertainty

3.2 Alternative 2 - Prescribe a Detailed Approach

If the staff prescribed a detailed approach, applicants would enjoy regulatory certainty at the expense of reduced flexibility. Tangible immediate impacts would include staff time to specify and defend the approach in a public forum. Intangible impacts would include a potential compromise of staff effectiveness as impartial safety reviewers and a loss of input from innovative applicants. Future impacts would include maintenance of the approach as newer software engineering methods were developed by the technical community.

Values - Probable improvement in the likelihood of achieving safety goals as a consequence of staff expertise and specialized knowledge derived during the development of the prescribed approach

- Common understanding of regulatory view of software practice

Impacts - Cost of staff effort to develop the approach

- Potential compromise of staff objectivity

- Innovative approaches discouraged as a result of increased cost
- Cost of evolving, maintaining, and communicating the approach

3.3 Alternative 3 - Endorse One or More Software Engineering Standards

If the staff endorses selected consensus software engineering standards, the staff and applicants obtain the benefit of the work of responsible software engineering professional standards committee volunteers. The value in this is the common understanding between the staff and applicants of an approach that has acceptance as good practice in the technical community. The standards usually permit tailoring to meet the needs of particular situations, so that a medium level of flexibility is retained. Additional staff effort is minimal, since members of the staff are already active in standards committees that the staff considers important to safety. Because detailed standards that address specific software engineering practices are available, the staff may select standards that address topics of particular importance regarding safety system software. Many standards, including IEEE software engineering standards, are reviewed and updated periodically, which acquaints the staff with changing practices. Coordination of standards efforts for standards used widely in the United States with international standards efforts is increasing, but the outcome of this is still unpredictable.

Values
- Probable improvement in the likelihood of achieving safety goals as a consequence of improvement in software practices
- Consideration of relevant topics
- Common understanding of good software practice, as defined by consensus processes in the software industry
- Maintenance and evolution of the definition of good software practice by the software industry

Impact
- Cost of endorsing the selected standards

4. CONCLUSIONS

There are a number of potential benefits associated with the use of digital I&C safety systems in nuclear power plants. Implementations of these systems must be consistent with the NRC's regulations. Three approaches to providing additional guidance for software were examined. Taking no action may result in

accumulating regulatory expense as applicants submit proposed methods to assure the staff that safety-related software meets the requirements of the NRC's regulations. A de facto acceptable method would probably evolve, but the time and effort required for this to happen are unknown. A detailed staff prescription has unacceptable impacts and would involve the staff directly in the applicant's solution of technical problems. Endorsing selected software engineering standards has good value with minimal impact and addresses the stated problem. Note that none of these approaches presents new regulatory requirements; they define acceptable approaches for meeting existing requirements.

5. DECISION RATIONALE

Based on the lowest impact and highest value for problem solution capability, the third alternative, endorsing selected software engineering standards, has been chosen. The highest value will be achieved by selecting standards that address software engineering processes that have a high potential for ensuring that safety system software meets the requirements of the NRC's regulations as applied to software. Standards should be selected based upon relevance and maturity.

BIBLIOGRAPHY

Beizer, Boris, *Software Testing Techniques*, Van Nostrand Reinhold, 1990.

Hecht, H., A.T. Tai, K.S. Tso, "Class 1E Digital Systems Studies, NUREG/CR-6113, USNRC, October 1993.[1]

Hecht, H., et al., "Verification and Validation Guidelines for High Integrity Systems," NUREG/CR-6293, USNRC, March 1995.[1]

Institute of Electrical and Electronics Engineers, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations," IEEE Std 7-4.3.2, 1993.

Lawrence, J.D., "Software Reliability and Safety in Nuclear Reactor Protection Systems," NUREG/CR-6101 (UCRL-ID-117524, Lawrence Livermore National Laboratory), USNRC, November 1993.[1]

Lawrence, J.D, and G.G. Preckshot, "Design Factors for Safety-Critical Software," NUREG/CR-6294, USNRC, December 1994.[1]

Seth, S., et al., "High Integrity Software for Nuclear Power Plants: Candidate Guidelines, Technical Basis and Research Needs," NUREG/CR-6263, USNRC, June 1995.[1]

USNRC, "Criteria for Digital Computers in Safety Systems of Nuclear Power Plants, Regulatory Guide 1.152, Revision 1, January 1996.[2]

USNRC, "Standard Review Plan," NUREG-0800, February 1984.[1]

---

[1]Copies may be purchased at current rates from the U.S. Government Printing Office, P.O. Box 37082, Washington, DC 20402-9328 (telephone (202)512-2249); or from the National Technical Information Service by writing NTIS at 5285 Port Royal Road, Springfield, VA 22161. Copies are available for inspection or copying for a fee from the NRC Public Document Room at 2120 L Street NW., Washington, DC; the PDR's mailing address is Mail Stop LL-6, Washington, DC 20555; telephone (202)634-3273; fax (202)634-3343.

[2]Single copies of regulatory guides may be obtained free of charge by writing the Office of Administration, Attention: Distribution and Services Section, U.S. Nuclear Regulatory Commission, Washington, DC 20555-0001; or by fax at (301)415-2260. Copies are available for inspection or copying for a fee from the NRC Public Document Room at 2120 L Street NW, Washington, DC; the PDR's mailing address is Mail Stop LL-6, Washington, DC 20555; telephone (202)634-3273; fax (202)634-3343.

Federal Recycling Program

UNITED STATES
NUCLEAR REGULATORY COMMISSION
WASHINGTON, DC 20555-0001

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

FIRST CLASS MAIL
POSTAGE AND FEES PAID
USNRC
PERMIT NO. G-67