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

DRAFT REGULATORY GUIDE

August 1996
Division 1
Draft DG-1055

Contact: J. Kramer (301)415-5891

DRAFT REGULATORY GUIDE DG-1055

CONFIGURATION MANAGEMENT PLANS 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, that systems and components be designed, tested, and inspected to quality standards commensurate with the safety function to be performed.[1] Criterion 1, "Quality Standards and Records," of Appendix A, "General Design Criteria for Nuclear Power Plants," of 10 CFR Part 50 requires, in part,[1] that appropriate records of the design and testing of systems and components important to safety be maintained by or under the control of the nuclear power unit licensee throughout the life of the unit. Appendix B, "Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants," to 10 CFR Part 50 describes criteria that must be met by a quality assurance program for systems and components that prevent or mitigate the consequences of postulated accidents. 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

[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, such 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 the configuration management activity. Criterion III, "Design Control," of Appendix B requires measures for design documentation and identification and control of design interfaces. The same criterion also requires that design changes be subject to design control measures commensurate with those used in the original design. Criterion VI, "Document Control," requires that all documents that prescribe activities affecting quality, such as instructions, procedures, and drawings, be subject to controls that ensure that documents, including changes, are reviewed for adequacy and approved for release by authorized personnel. Criterion VIII, "Identification and Control of Materials, Parts, and Components," requires, in part, that parts and components be identified to prevent the use of incorrect or defective parts or components. Criterion XVI, "Corrective Action," requires that conditions adverse to quality, such as failures, malfunctions, deficiencies, and others, be identified, and that the cause be determined, the condition be corrected, and the entire process be documented. Criterion XVII, "Quality Assurance Records," requires in part that sufficient records be maintained so that data that is closely associated with the qualification of personnel, procedures, and equipment be identifiable and retrievable.

This regulatory guide, which endorses IEEE Std 828-1990, "IEEE Standard for Software Configuration Management Plans,"[3] and ANSI/IEEE Std 1042-1987, "IEEE Guide to Software Configuration Management,"[3] with the clarifications provided in the Regulatory Position, describe methods acceptable to the NRC staff for complying with the NRC's regulations for promoting high functional reliability

---

[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.

and design quality in software used in safety systems.[4] In particular, the methods are consistent with the previously cited General Design Criteria and the criteria for quality assurance programs of Appendix B as they apply to the maintenance of appropriate records of, and control of, software development activities. 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 Commission'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.

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

[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."

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, identification, control, and documentation of the software must be accomplished as part of the effort to achieve compliance with the NRC's requirements. In addition to the record maintenance requirement of Criterion 1 of Appendix A, Appendix B to 10 CFR Part 50 provides detailed quality assurance criteria, including criteria for administrative control, design documentation, design interface control, design change control, document control, identification and control of parts and components, and control and retrieval of qualification information associated with parts and components. For software, these activities are often called, in aggregate, "software configuration management" (SCM). SCM is identified as a safety system criterion by the industry standard, IEEE Std 7-4.3.2-1993, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations,"[3] which is endorsed by Revision 1 of Regulatory Guide 1.152, "Criteria for Digital Computers in Safety Systems of Nuclear Power Plants."

Configuration management is a significant part of high quality engineering activities and is already required by the NRC staff for structures, systems, and components important to safety. While the principles and intentions of traditional configuration management apply equally to software, there is also a significant change in emphasis for which traditional hardware configuration management systems might not be sufficient. This is because with software there is a greater emphasis on design process and the deliverable product is more like a design output. In the production of engineered hardware, design outputs are inputs to a manufacturing process, and configuration management activities focus on ensuring that design outputs and manufacturing process variables are traceable to identifiable manufactured products. In contrast, with engineered software, a large amount of design process information and many intermediate design outputs are associated with the final design output. Relatively many software engineering changes are expected and encountered. Consequently, although similar in intent to hardware configuration management, software configuration management requires a change in emphasis, with expansion of the importance of intermediate

design baselines and associated design process information. The needs for robust change management and identification and control of product versions are also substantially increased.

One consensus standard on software engineering, IEEE Std 828-1990, "IEEE Standard for Software Configuration Management Plans," and one guide, ANSI/IEEE Std 1042-1987, "IEEE Guide to Software Configuration Management" (reaffirmed in 1993), describe software industry approaches to SCM that are generally accepted in the software engineering community. They provide guidance for planning and executing an SCM program. Together, this standard and guide elaborate on the important features required in an SCM program that may be under-emphasized in traditional hardware configuration management programs.

The relationship of IEEE Std 1042-1987 to IEEE Std 828-1990 is important. IEEE Std 1042-1987 is a tutorial guide that explains how to comply with IEEE Std 828-1990. Section 1.1, "Scope," of IEEE Std 1042-1987 states that the guide provides suggestions and examples and "presents an interpretation of how IEEE Std 828-1983 [since updated to IEEE Std 828-1990] can be used for planning the management of different kinds of computer program development and maintenance activities." The actual criteria to be met for compliance with the standard are contained in IEEE Std 828-1990. Both the standard and the guide apply to general purpose software development and maintenance efforts, and they do not have specific criteria for safety system software.

C. REGULATORY POSITION

IEEE Std 828-1990, "IEEE Standard for Software Configuration Management Plans," provides an approach acceptable to the NRC staff for meeting the requirements[5] of 10 CFR Part 50, as applied to software, in planning configuration management of safety system software, subject to the provisions listed below.

ANSI/IEEE Std 1042-1987, "IEEE Guide to Software Configuration Management," subject to the provisions listed below, provides guidance acceptable to the NRC staff for carrying out software configuration management plans produced under the auspices of IEEE Std 828-1990. IEEE Std 1042-1987 should be used with the definitions of IEEE Std 828-1990 to implement the details of plans prepared

[5] 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.

pursuant to IEEE Std 828-1990. In the provisions listed below, reference is made to explanatory sections of IEEE Std 1042-1987 where appropriate to clarify SCM concepts.

To meet the cited requirements of Appendix A of 10 CFR Part 50 by complying with the cited criteria of Appendix B, the following exceptions are necessary and will be considered by the NRC staff in the review of applicant submittals as described by the Standard Review Plan, which is currently under revision. (In this Regulatory Position, the cited criteria are in Appendix B to 10 CFR Part 50 unless otherwise noted.)

1. IEEE Std 828-1990 refers to IEEE Std 610.12-1990, "IEEE Standard Glossary of Software Engineering Terminology," for definitions of the technical terms that are enumerated in section 1.3 of IEEE Std 828-1990. These definitions are acceptable with the following clarifications and additions:

   1.1 Baseline

   Meaning (1) of baseline is to be used in IEEE Std 828-1990. Formal review and agreement is taken to mean that responsible management has reviewed and approved a baseline. Baselines are subject to change control. Acceptable baseline change approval authority is described in Exception 2 below.

   1.2 Promotion

   The term "promotion," as defined in Section 1.4 of IEEE Std 1042-1987, is added to the list of defined terms.

   1.3 Interface

   All four variations of the meaning of interface are to be used in IEEE Std 828-1990, depending upon the context. Meaning (1), "A shared boundary across which information is passed," is interpreted broadly according to Criterion III to include design interfaces between participating design organizations.

   1.4 Configuration Audit

   IEEE Std 610.12-1990 refers the definition of configuration audit to two other audits without specifying whether one or both definitions

are meant. In the context of an audit for delivery of a product, a configuration audit includes both a functional configuration audit and a physical configuration audit.

2. Section 2.2.4 of IEEE Std 1042-1987 is modified to permit hierarchies of change approval authority levels, as described in that section and discussed in section 3.3.2.1, provided the required authority level is commensurate with life cycle stage (nearness to release) and product importance to safety. The promotion of a software product might involve a change in level of control and responsible individual.

3. Criterion II, "Quality Assurance Program," states that activities affecting quality are to be accomplished under suitably controlled conditions. Criterion V, "Instructions, Procedures, and Drawings," states that instructions, procedures, or drawings are to include appropriate quantitative or qualitative acceptance criteria for determining that important activities have been satisfactorily accomplished. Section 1.3 of IEEE Std 828-1990 defines "control point." The software configuration management (SCM) plan should describe the criteria for selecting control points and establish the correspondence between control points identified in the plan and baselines, project milestones, and life cycle milestones.

4. Section 2.3 of IEEE Std 828-1990 describes four functional areas under which configuration management activities are grouped: configuration identification, configuration control, status accounting, and configuration reviews and audits. However, while IEEE Std 828-1990 requires that SCM plans describe provisions for these activities, it has no minimal set of activities for safety system software. Criterion II, "Quality Assurance Program," states that activities affecting quality are to be accomplished under suitably controlled conditions. Criteria III, "Design Control"; VI, "Document Control"; VII, "Control of Purchased Material, Equipment, and Services"; VIII, "Identification and Control of Materials, Parts, and Components"; XVII, "Quality Assurance Records"; and XVIII, "Audits," address various aspects of the need for controlling designs, documentation, and materials. For safety system software, the minimal set of activities must accomplish the following: identification and control of all software

designs and code, identification and control of all software design interfaces, control of all software design changes, control of software documentation (user, operating, and maintenance documentation), control of software vendors supplying safety system software, control and retrieval of qualification information associated with software designs and code, software configuration audits, and status accounting. Some of these functions or documents may be performed or controlled by other quality assurance activities; in this case the SCM plan should describe the division of responsibility.

5. Criterion XVI, "Corrective Action," requires that conditions adverse to quality, such as failures, malfunctions, deficiencies, and others, be identified, the cause be determined, the condition be corrected, and the entire process be documented. In software development or maintenance, the responsibility for these activities is often distributed among several organizations, potentially leading to a fragmented view of the correction process. Section 2.3.2 of IEEE Std 828-1990 requires a partial description of the correction process, including change requests, change evaluation, change approval, change implementation, change verification, and changed-version release. The preliminary steps leading to a change request should also be described, including responsibility for executing and documenting anomaly reports, problem analyses, and statistical monitoring of software performance. If these activities are described by other documents, the descriptions may be included by reference.

6. Section 2.3.1.1 of IEEE Std 828-1990 requires, as a minimum, that all configuration items that are to be delivered be listed in the SCM plan. This fulfills the intent of Criterion VIII, "Identification and Control of Materials, Parts, and Components," with regard to safety system software if all software deliverables are identified and controlled as configuration items. Criterion III, "Design Control," requires measures for design documentation, identification and control of design interfaces, and control of design changes. Criterion VI, "Document Control," requires that all documents that prescribe activities affecting quality, such as instructions, procedures, and drawings, be subject to controls that ensure that documents, including changes, are reviewed for adequacy and approved for release by

authorized personnel. Criterion XVII, "Quality Assurance Records," requires in part that sufficient records be maintained so that data that is closely associated with the qualification of personnel, procedures, and equipment will be identifiable and retrievable. For safety system software, configuration items or controlled documents should include the following:

- Requirements, designs, and code
- Exact versions of support software used in development
- Libraries of software components essential to safety
- Software plans that could affect quality
- Test software requirements, designs, or code used in testing
- Test results used to qualify software
- Analyses and results used to qualify software
- Software documentation
- Databases and software configuration data
- Commercial software items that are safety system software
- Software change documentation

Items that could change because of design changes, review, or audit should be configuration items subject to formal change control. Other items that may not change but are necessary to ensure correct software production, such as compilers, should also be configuration items. Items that are retained for historical or statistical purposes may be controlled documents.

7. Criterion VII, "Control of Purchased Material, Equipment, and Services," requires measures to ensure that purchased material conforms to procurement documents. Criterion VIII, "Identification and Control of Materials, Parts, and Components," requires measures to be established for the identification and control of materials, parts, and components. Contractually developed or qualified commercial software products that are safety system software must be taken under control by an SCM program that complies with IEEE Std 828-1990 as endorsed by this regulatory guide. This means, for example, that the exact version of the product is identified and controlled according to the change control procedures applied to other configuration items and that its usage is tracked and reported.

8. Tools used in the development of safety system software should be handled according to IEEE Std 7-4.3.2-1993, "Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations," as endorsed by Revision 1 of Regulatory Guide 1.152. In particular, tools must be taken under control (i.e., treated as a configuration item) by an SCM program operated by the using organization that complies with IEEE Std 828-1990 as endorsed by this regulatory guide.

9. Criterion V, "Instructions, Procedures, and Drawings," requires that instructions, procedures, and drawings include appropriate quantitative or qualitative acceptance criteria for determining that important activities have been satisfactorily accomplished. In addition, Criterion VIII, "Identification and Control of Materials, Parts, and Components," requires measures to be established for the identification and control of materials, parts, and components; and Criterion II, "Quality Assurance Program," states that activities affecting quality must be accomplished under suitably controlled conditions. In order to maintain acceptance criteria established in accordance with Criterion V and suitably controlled conditions (in accordance with Criteria II and VIII) for safety system software, section 3.2 of IEEE Std 828-1990 is not endorsed by this regulatory guide.

10. IEEE Std 828-1990, in paragraph 2.3.2(4), requires a definition of the verification, implementation, and release of a change. The criteria for verification must be consistent with Criterion III, "Design Control," which requires that design changes be subject to design control measures commensurate with those applied to the original design. This encompasses the re-examination of any appropriate safety analysis related to the change.

11. IEEE Std 828-1990, in paragraph 2.1(7), requires the scope of the SCM plan to address the assumptions upon which the plan is based, including assumptions that might have an impact on cost and schedule. Assumptions about cost and schedule must not diminish safety considerations; any use of these criteria must be consistent with the requirement of 10 CFR 50.57(a)(3) that there be reasonable assurance that the activities authorized by the operating license can be conducted without endangering the health and safety of the public.

12. IEEE Std 828-1990 and IEEE Std 1042-1987 reference other industry codes and standards. These references to other 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 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, 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 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 NRC 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-

[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.

6294, "Design 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 configuration management (SCM) has been identified in many studies, including most of those referenced above, as a key activity for the production of high-integrity systems. Effective control of materials and design activities is fundamental in Appendix B to 10 CFR Part 50, and SCM is fundamental to the control of software products and development activities. NUREG/CR-6294, in discussing mandatory factors in the development of safety-critical software, states that "Configuration management is crucial and is absolutely necessary to have confidence that the correct product is built, and that change occurs in an orderly way. Configuration errors are among the simplest and also the most prevalent made in the software industry." NUREG/CR-6263 discusses the relation of SCM to safety and references additional standards and studies stressing the importance of SCM to high-integrity systems. In the Foreword to IEEE Std 828-1990, it is stated that "SCM activities, whether planned or not, are performed on all software development projects; planning makes these activities more effective." This is a recognition that one cannot develop software without SCM of some form and that planning helps to ensure that SCM is effectively executed.

2. ALTERNATIVE APPROACHES

Based on the studies mentioned above, an alternative was identified in which consensus in the software engineering community is sufficient to ensure

[2]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.

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 attraction 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 NRC 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 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 will propose measures it deems necessary, and these measures will then be reviewed by the staff and discussed with the applicant to reach a resolution that is 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 U.S. 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

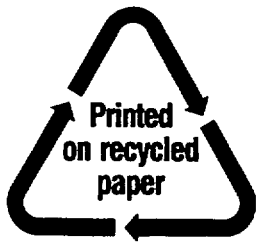
4. CONCLUSIONS

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

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]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.

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