61 FR 46834
9/5/96
S.S. Kramer Contact

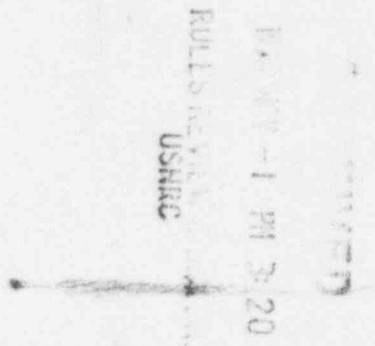
DS09 #3

Nebraska Public Power District

COOPER NUCLEAR STATION
P.O. BOX 98, BROWNVILLE, NEBRASKA 68321
TELEPHONE (402)825-3811
FAX (402)825-5205

NLS960202

October 28, 1996



U.S. Nuclear Regulatory Commission
Rules Review and Directive Branch
DFIPS
Office of Administration
Washington, D.C. 20555

Subject: Draft Regulatory Guides DG-1054, DG-1055, DG-1056, DG-1057, DG-1058, and DG-1059

Gentlemen:

Cooper Nuclear Station has completed its review of the subject documents and the following comments are offered for your information and use.

Draft Regulatory Guide DG-1054 "Verification, Validation, Reviews, and Audits for Digital Computer Software used in Safety Systems of Nuclear Power Plants"

	Page/Paragraph	Comment
1.	p.7/para 3	"The independent verifiers must also be as proficient in software engineering as the software developer." This may not be a reasonable requirement. Consider that there are no two software engineers who are *exactly* as proficient as each other. This means that a developing entity, whether it be a utility or a vendor, cannot put their best software engineer on safety-related software. As written, the guide requires that the best must be held in reserve as a verifier. This is probably not the intent. Suggested reword to following or similar: "The independent verifiers must be experienced with and competent in the operating systems, compilers, and other tools used by the developer to produce the software."
2.	p.8/para 5	"The use of this guidance for the acceptance of pre-existing (e.g., commercial off-the-shelf) critical software not verified during development to the provisions of this regulatory guide or its equivalent is not endorsed." This limitation is understood. However, DG-1059, p.6, para 1.3 states

IEP-11 Guides & Manuals

Rtn

"...an acceptance process must be included at an appropriate point in the life cycle model to establish the suitability of the pre-existing software for its intended use." Based on these two requirements, two separate conclusions could be drawn: (1) Safety-related off-the-shelf software is forbidden, or (2) the use of such software is not forbidden, but there is no regulatory guide which provides an acceptable means of testing such software. Clarification is needed, as these two guides seem to be inconsistent on this issue.

Draft Regulatory Guide DG-1055 "Configuration Management Plans for Digital Computer Software used in Safety Systems of Nuclear Power Plants"

Section	Comment
1. Format	To improve the clarity of the document, the following recommendations are made. Each section should start with the topic of the section, followed by the regulatory criteria applicable. Sample sections with suggested changes are: Section A, Introduction, move third paragraph to be the first paragraph and rewrite as follows: This regulatory guide describes methods acceptable to the NRC staff for complying with the NRC's regulations for promoting high functional reliability and design quality in software used in safety systems. The NRC staff endorses IEEE Standard 828-1990, "IEEE Standard for Software Configuration Management, Plans," and ANSI/IEEE Standard 1042-1987, "IEEE Guide to Software Configuration Management" with the clarifications provided in Section C, Regulatory Position. In particular, the methods are consistent with the General Design Criteria cited below and the... Section B, Discussion, move first two paragraphs to the end of this section. Section C, Regulatory Position, certain provisions start with the discussion of specific sections of a Standard followed with regulatory criteria. Others start with criteria first and follow with the Standards section. Reorganize the following item numbers to start with the Standards section first, then regulatory criteria: item 3 (page 7), item 7 (page 9), item 9 (page 10).

	Page/Paragraph	Comment
2.	p.6/para 1.3	"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." The Criterion III requirement, "Measures shall be established for the identification and control of design interfaces and for coordination among participating design organizations," is met through other language in IEEE 828. While the first two requirements in IEEE 828 (definitions for "the nature of the interface" and the "affected organizations") meet the broad interpretation, the last two (define "the interface code, documentation, and data to be controlled," and "how the interface control documents are approved and released into a specified baseline") are a poor fit. Recommend deletion of this requirement. If the IEEE language regarding organizational interaction is not specific enough in this regard, the regulatory guide should address the issue.
3.	p.9/para 6	"Other items that may not change but are necessary to ensure correct software production, such as compilers, should also be configuration items." This seems to imply that the actual compiler software be retained as a configuration item, as opposed to the "exact version" of the compiler as required in the preceding part of the paragraph. There are indications that this requirement may be unreasonable. (1) Compiler software is controlled by outside organizations. Making it a configuration item wrongly implies that the licensee can control quality, version, etc. (2) There is no guarantee that the compiler can be put to use when the computer platform changes. For example, if this requirement exists as an event reconstruction issue, then the licensee would also have to retain every old computer, peripheral device, and chip set. (3) If the requirements, design, and acceptance testing are deemed adequate, retention of the compiler adds no value. Recommend clarification of this requirement.

Draft Regulatory Guide DG-1056 "Software Test Documentation for Digital Computer Software used in Safety Systems of Nuclear Power Plants"

	Page/Paragraph	Comment
1.	p.8/para 4	"Each feature in safety system software is to be formally tested under at least one test design." This requirement also seems to reduce the possibility of the use of commercial off-the-shelf software. See discussion under DG-1054, comment #2.

Draft Regulatory Guide DG-1057 "Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

	Page/Paragraph	Comment
1.	p.8/para 4	"These persons must also be as proficient in software engineering as the designer or coder." See discussion under DG-1054, comment #1.

Draft Regulatory Guide DG-1058 "Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

	Page/Paragraph	Comment
1.	p.5/para 1.2	"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." See discussion under DG-1055, comment #3.
2.	p.7/para 2.7	"Both types of traceability are required." This refers to the IEEE 830 traceability options of backward and forward traceability. Backward traceability requires that a feature built into software must reference a requirement. This is desirable. Forward traceability requires that each requirement has a reference for each item it spawns in the future. Since Software Requirement Specifications (SRS) are in the first stage of a life cycle, comprehensive forward traceability may only be practically achieved by making each SRS an "as-built," reducing the value of the SRS as a life cycle tool. If this is not the intent, change the requirement to read, "Backward traceability is required." Also, this paragraph includes the phrase "Each identifiable requirement should be written so that it is also 'forward traceable' to subsequent design outputs, e.g., from SRS to software design and from software design to SRS." This seems to represent the relationship between software design and SRS as a forward trace, when it is actually a backward trace. Reword to clarify.
3.	p.9/para 6.3	"Software requirements for handling both hardware and software failures should be provided, including requirements for analysis of and recovery from computer system failures." DG-1054 states that "...software failures that are not the consequence of hardware failures are caused by design errors..." It is not practical to require that software detect and handle its own design flaws. Recommended rewording: "Software requirements for handling hardware failures should be provided, including requirements for analysis of and recovery from computer system failures. Input arguments to each software module shall be validated where applicable. Return status

checking, with error handling where necessary, is required after each function or procedure call."

Draft Regulatory Guide DG-1059 "Developing Life Cycle Processes for Digital Computer Software used in Safety Systems of Nuclear Power Plants"

	Page/Paragraph	Comment
1.	p.6/para 1.3	"...an acceptance process must be included at an appropriate point in the life cycle model to establish the suitability of the pre-existing software for its intended use." This statement seems to be in conflict with the apparent disapproval of pre-existing software for safety use. See discussion under DG-1054, comment #2.

If you have any questions regarding these comments, please contact me.

Sincerely

P D Graham

P. D. Graham
Vice President of Nuclear Energy

/nr

Correspondence No: NLS960202

The following table identifies those actions committed to by the District in this document. Any other actions discussed in the submittal represent intended or planned actions by the District. They are described to the NRC for the NRC's information and are not regulatory commitments. Please notify the Licensing Manager at Cooper Nuclear Station of any questions regarding this document or any associated regulatory commitments.

COMMITMENT	COMMITTED DATE OR OUTAGE
None	