



10809
J.J. Kramer

**CAPRI
TECHNOLOGY
INCORPORATED**

50 Curtner Ave., Suite 7, Campbell, California 95008

61 FR 46839
Sept. 5, 1996
①
OCT 21 AM 7:14
USIN Phone: (408) 559-5996 Fax: (408) 559-5998

October 11, 1996

Rules Review and Directives Branch
DFIPS, Office of Administration
U.S. Nuclear Regulatory Commission
Washington DC 20555

SUBJECT: Comments on Draft Regulatory Guides:

DG-1054. Verification, Validation, Reviews and Audits for ...
DG-1055. Configuration Management Plans for ...
DG-1056. Software Test Documentation for ...
DG-1057. Software Unit Testing for ...
DG-1058. Software Requirements Specifications for ...
DG-1059. Developing Software Life Cycle Processes for ...

...Digital Computer Software Used in Safety Systems of Nuclear Power Plants

The original "safety model" for nuclear power plants, developed in the 1960's and 1970's, does not include provisions for programmable logic (e.g., software). The technology was too new to be considered "safe" for use in safety system designs. Instead, hardwired analog I&C systems were used which depended on quality, diversity, and defense-in-depth to ensure safe and reliable operation.

The proposed reg guides attempt to backfit software technology into the existing safety model, design criteria, and standards. This add-on approach can introduce or mask safety problems because the underlying model does not address software. For example, diversity is one of the three cornerstones in the existing model. Diversity includes redundancy, isolation, and separation criteria in the design of safety systems to achieve high reliability. However, for software-based systems, these same criteria are not appropriate. Redundancy is useless in a common-mode software failure (e.g., the Ariane-5 failure). Even new ideas, such as software reuse for high reliability, may not be safe (again, look at the Ariane-5 experience). Therefore, I believe that the nuclear power safety model must be revisited and revised to include software-based systems. The draft reg guides are a good start in this direction because they reference some of the high level criteria and attempt to relate a general-purpose software standard to those criteria, but the overall safety model perspective is missing.

All of the proposed reg guides deal with the quality aspect of software, but where is the focus on safety? Quality is good and the process that is used to produce a software product needs to be of proven quality.

Unfortunately, the set of IEEE standards endorsed by the reg guides does not guarantee safety (or, in my opinion, quality), so it will still be up to the licensee to defend the quality of any software-based safety system. For example, the IEEE 830 standard on software requirements specifications (SRS) describes attributes of a good SRS and defines the section headings that a good SRS might have. I believe it is possible to write an inadequate SRS from a safety viewpoint that meets the IEEE 830 standards. Since IEEE 830 was not written to cover system safety and possibly may not cover software safety, then why include it in a reg guide? The same comment applies to the other general-purpose standards endorsed in these draft reg guides.

What was the basis for choosing the standards referenced in the draft reg guides and for not including other standards? For example, ASME NQA-2A-1990, Part 2.7 prescribes a plan for software quality that includes the development process, requirements, V&V, and configuration management. Why were the eight IEEE standards chosen and this one was not?

I must also question the usability of the proposed reg guides. They explicitly reference eight different IEEE standards plus some other implicit ones (i.e., IEEE Std. 603-1991 and IEEE Std. 7-4.3.2-1993). This set of interacting reg guides and standards has too many interfaces, which leads to inconsistencies and ambiguities, which makes them difficult to use. Why not have one reg guide for software quality that has a minimum number of referenced IEEE standards? Immediately, this minimalist approach reduces the number of inconsistencies.

There are a number of software-related research projects sponsored by EPRI that have produced reports that are more appropriate than the references included in the reg guides. Why have they not been included?

There is a very active Internet email discussion group on system safety and software that has been debating the value and use of standards in this area. This group is loosely organized by Dennis Lawrence of LLNL with the objective to recommend a set of software safety standards to be developed by the IEEE. As a result of the group discussions, a minority position has been posted that recommends studying and documenting the use of existing standards in a number of application domains, including nuclear power. [The minority position statement is appended to this note.] Once this knowledge base is available, then standards can be developed where needed. Many of the participants in this discussion group are associated with the nuclear power industry, and represent both the regulatory and the licensee sides. At this time, all of the work is on a volunteer basis and there is no official project with a purpose, schedule, deliverables, etc.

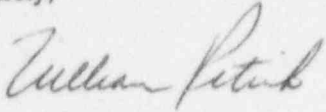
RECOMMENDATIONS

1. Support a nuclear power study group recommended by the minority report to the IEEE WGSC by sponsoring or co-funding a project to establish the knowledge base for nuclear power applications of existing safety standards dealing with software.
2. Assist in the development of a safety model for nuclear power plants that maintains the focus on safety (rather than software QA) and incorporates world-wide thinking on system safety with embedded software (including the work on ISO/IEC Standards 12207 and 15288).
3. Develop a minimum number of reg guides (i.e., ONE) that establishes a minimum number of standards that are consistent with the new safety model from recommendation 2. To do this, one must question the need for and the use of every standard or reference. As a starting point, I would recommend a derivative of MIL STD 882C for a safety plan, ASME NQA-2A-1990, Part 2.7 for a software quality program, and IEEE-603 (and its adjunct IEEE 7-4.3.2) for requirements consistent with the current safety model (i.e., no programmable logic). Any of the other standards listed in the

proposed reg guides that do not have specific acceptance criteria are useful as background information, but they should not be included in the reg guide (e.g. IEEE 830).

4. Delay imposing any new reg guides for software-based safety systems until there is some consensus on recommendation 3.

Sincerely,



William Petrick
Capri Technology Inc.
50 Curtner Ave., Suite 7
Campbell, CA 95008

Tel. 408.559.5996
Fax. 408.559.5998
E-mail. caprit@ix.netcom.com
URL. <http://www.netcom.com/~caprit>

ATTACHMENT TO DRAFT REG GUIDE COMMENTS

September, 1996

Dr. Dennis Lawrence
LLNL
Livermore, CA

Subject: Minority Position on New Software Safety Standards

Dr. Lawrence:

After many fruitful and thought-provoking online discussions regarding the charter and recommendations for new software safety standards, it became clear that there were two dominant positions taken by the posting members. The purpose of this letter is to document the minority position so the SESC committee can make an informed decision.

The minority position is that it is premature to write new software safety standards, and that the effort will be better spent to establish a solid knowledge base on which to build future safety system standards. The following paragraphs provide an explanation for our position and a recommendation that is both practical and efficient, considering the scarce resources available.

First, the concept of software safety must be carefully defined.. Software is not unsafe by itself, so any approach to building safer software must be part of and consistent with an overall safety engineering program. Separating software safety from the system can only lead to misunderstandings and potential unsafe system conditions. This is precisely what the standards are trying to prevent. We must be very careful to maintain the focus on safety, not software.

Next, there are important differences in approaches to safety between various industries, including the underlying safety models, accident models, regulatory requirements, engineering disciplines, and assumptions about the causes of accidents. We believe that any attempt to provide one standard for software safety that adequately covers these areas for all industries is not practical and would not be effective.

Finally, new safety standards are proliferating from a number of industries and standards groups around the world. We do not see any reason to expect that another standard in this general area will result in the ultimate or fundamental safety standard. Furthermore, if there is not enough real knowledge included in a standard and that standard is used inappropriately in a real system, then the standard becomes part of the safety problem, not its solution. This is particularly important in court cases where IEEE standards are referenced as protection from charges of negligence. At the present time, it is more important to understand the basis for the existing standards and learn from the experiences that come from using these standards to develop safe systems.

Therefore, the primary emphasis of the minority position is to establish a foundation of knowledge based on existing standards related to system and software safety and on general software engineering practices that are already used for safety applications. We recommend that a multi-industry volunteer group be established to develop a guidance document based on these existing standards. This group can operate in the same way that IEEE standards are written and approved. The guidance document would identify the standards and related handbooks, provide descriptions and contextual information, and include pointers

and comparisons to other standards, where appropriate. The guidance would not create more definitions or terms, it would not develop new approaches to software safety, and it would not try to enforce one set of terms or engineering discipline on all industries or all problems.

Since worldwide standards are constantly changing, the guidance developed by the working group must also be dynamic. As part of the initial effort of the working group, we also recommend that a procedure be established to update and maintain the guidance information. Using the ubiquitous Internet, access to the guidance information and updates from new standards can be readily accommodated. In addition, a web-based repository could also contain pointers to relevant books, papers, and experience reports that have been used to develop safe systems.

Sincerely,

(A list of endorsers that is now over 30)
