A Method to Improve the Software Acceptance Criteria for Nuclear Power Plants

Yong Suk Suh,[a] Heui Youn Park,[a] Ki Sung Son,[b] Ki Hyun Lee,[b] Hyeon Soo Kim [c]
a I&C and HF Div., KAERI, 150 Dukjin-dong, Yuseong-gu, Daejon, Korea, 305-353, yssuh@kaeri.re.kr
b Control Tech. Research Inst., SEC Co., Ltd.,974-1 Goyeon-ri Woongchon-myon, Ulju-gun, Ulsan, Korea, 689-871
c Dept. of Computer Science and Eng., Chungnam Nat'l Univ., 220 Gung-dong, Yuseong-gu, Daejon, Korea, 305-764

1. Introduction

The license is a mandatory process required by a governmental authority and the certification is a voluntary process administrated by a professional community. A software certification is a result of an assessment that the certified software conforms to required criteria or standards. The certification is used as a committed promise to produce a high quality software, so software acquirers are requiring it from their suppliers. For example, US DoD (Department of Defense) requires an achievement of CMMI-SW (Capability Maturity Model Integration-Software) certification for participating in a major military software project [1]. It is commonly said that the purpose of achieving a certification is to improve the product quality. In the nuclear area, a software certification has been rarely concerned with or required for the software used in a safety function of NPPs (Nuclear Power Plants).

The safety critical software for NPPs is accepted by the nuclear regulators when the following three criteria are met: 1) acceptable plans should be prepared to control the software development activities, 2) the plans should be followed in an acceptable software life cycle, and 3) the process should produce acceptable design outputs [2]. The acceptance criteria are so abstractive that the nuclear regulators may assess the software development plans, activities, outputs based on their subjective engineering judgments. This is inevitable because a software has invisible or intangible characteristics. It is hard to assess the totality of a software prior to running it. These have caused the judgments to be biased. The regulators may want some objectiveness in assessing how much capability for software development the supplier possesses. In that case, the software certification can assist them for such an assessment.

This paper proposes a method to improve the software acceptance criteria by applying the software certification to the criteria. This will assist the regulators to assess the supplier's software development capability. For this, the criteria should include the following certification requirements: 1) software process, 2) software engineers, and 3) software product.

2. Application of a Software Certification

A well-defined process, well-educated engineers, and a well-tested product are major factors that impact on the software quality. This paper surveys the certifications for the three factors and suggests how to apply them to the software acceptance criteria.

2.1 Certification of a Software Process

IEEE 7-4.3.2-2003 specifies that software shall be developed, modified, and accepted in accordance with IEEE/EIA 12207.0-1996. The IEEE/EIA 12207.0 is so well-defined that the DoD announced that MIL-STD-498, which had been used for military software development, was cancelled and instead the IEEE/EIA 12207 has replaced [3]. The IEEE/EIA 12207.0 adopted ISO/IEC 12207. The ISO/IEC 12207 establishes a common framework for software life cycle processes that can be referenced by the software industry. This paper focuses on the ISO/IEC 12207 as a basis for establishing the software process. There are three well-known certifications which assess a conformance to the ISO/IEC 12207: 1) SPICE (Software Process Improvement and Capability dEtermination), 2) CMMI-SW, and 3) ISO 9001.

The SPICE, which was developed by the ISO (International Organization for Standardization) as ISO 15504, is a standard focusing on a software process assessment and improvement, as well as a supplier's software development capability determination. It provides a framework for the assessment of the processes and categorizes six capability levels: 0) incomplete process, 1) performed, 2) managed, 3) established, 4) predictable, and 5) optimizing process. The certified SPICE assessors appraise an organizational capability as one of the six levels. The process model in the SPICE is based on the ISO/IEC 12207. There was a study which suggested that the safety critical software processes should achieve a third capability level or above of the SPICE [4].

The CMMI-SW which was developed by SEI (Software Engineering Institute) at U.S. CMU (Carnegie Mellon University) provides twenty-five key process areas and classifies the capability and the maturity of a software development organization into six and five levels respectively. The capability levels are classified as: 0) incomplete process, 1) performed, 2) managed, 3) defined, 4) quantitatively managed, and 5) optimizing process. The maturity levels are classified as: 1) initial, 2) managed, 3) defined, 4) quantitatively managed, and 5) optimizing process. The SEI authorized lead assessors appraise an organizational maturity as one of the levels. The CMMI-SW uses many ideas from the SPICE and now it is compatible with the SPICE. So, the CMMI-SW is based on the ISO/IEC 12207. The U.S. DoD requires an applicant who wishes to participate in a major military software project to achieve at a minimum CMMI-SW level 3 or its equivalent [1].

The ISO 9001 standard provides requirements of a quality management system applicable to the overall industries. The ISO has an ISO 90003 standard which helps in the implementation of ISO 9001 for the software industry: The ISO 90003 provides guidance for organizations in the application of ISO 9001 to the acquisition, supply, development, operation and the maintenance of a software and the related support services. The ISO 90003 is strongly related to the ISO/IEC 12207. So, in order to achieve the ISO 9001 certification in the software industry, it should be considered to implement the ISO/IEC 12207.

The three certifications are not really that different in terms of their software engineering principles. They are consistently developed on the basis of ISO/IEC 12207. For the software acceptance criteria for NPPs, it is not necessary to require achieving all of the three certifications but instead one of the three is enough to assess a supplier's capability.

2.2 Certification of a Software Engineer

In Korea, when referring to the MIC (Ministry of Information and Communication), there exists plenty of various certifications for software engineers such as private, public, and foreign certifications. It is hard to establish a guideline for the qualification of safety critical software engineers. There is no agreement among the software professionals on the basis of a software engineer's qualification. However, as a result of a safety critical software disaster, a qualification will be inevitably required by the legislatures in the future. KEPIC (Korea Electric Power Industry Code) QAP, which specifies the criteria of the nuclear quality assurance plan, requires the qualifications of inspectors, auditors, and mechanics and civil professional engineers. These requirements can be applied to the qualification of safety critical software engineers for NPPs.

2.3 Certification of a Software Product

It is not easy to certify a software product due to the lack of standardized software metrics and the difficulties in software measurements. There exists requirements, in the acceptance criteria, that an IV&V (Independent Verification and Validation) should be performed throughout the software process. However, it does not formally state the qualification of an IV&V organization. It is necessary to require that a certified organization should perform the IV&V. In Korea, TTA (Telecommunications Technology Association), which is a governmental organization, has an IT (Information Technology) testing laboratory. The laboratory provides third-party independent testing and certification services for IT products, to ensure that the IT products meet the user requirements and conform to the standards. The laboratory uses international standardized quality models and metrics from the ISO/IEC 9126 and the ISO/IEC 14598 standards. The ISO/IEC 9126 provides, in a structured manner, six quality characteristics for the evaluation of a software: functionality, reliability, usability, efficiency, maintainability, and portability. Each characteristic has several sub-characteristics and each sub-characteristic also has several attributes. The ISO/IEC 14598 consists of a six-part standard for a software product assessment. TTA is a eligible organization to certify a software product, so TTA can be considered as a third-party IV&V organization for NPPs.

3. Conclusion

This paper pointed out that a nuclear regulator may accept a safety critical software based on subjective judgments. This can be a hazardous factor for the safety of NPPs. To overcome this problem, this paper proposed to adopt software certifications as a part of the software acceptance criteria. Although the software certifications are currently premature, they can ensure that software engineering professionals appraise the software supplier's capability to produce a high quality software.

This paper surveyed the certification of a software process, a software engineer, and that of a software product and discussed a method to adopt them into the software acceptance criteria. The adoption can reduce the nuclear regulators' burden when assessing a supplier's capability and also eliminate regulators' subjective view during an assessment. This can have an effect of increasing the software quality, thus it results in increasing the safety of NPPs. This paper does not insist on the software certification policy as a sole means to assess an supplier's software development capability but encourages the adoption of the certification as a part of the software acceptance criteria for NPPs. The certification should be renewed periodically.

REFERENCES

[1] Memorandum for Component Acquisition Executives Director of Ballistic Missile Defense Organization, Software Evaluations for ACAT 1 Programs, U.S. DoD, 1999.

[2] Software Review Plan: Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems, NUREG-0800 BTP HICB-14, U.S. NRC, Rev. 4, 1997.

[3] Notice of Cancellation, Software Development and Documentation, MIL-STD-498, NOTICE 1, May 27, 1998

[4] O Benediktsson, R B Hunter, A D McGettrick, Processes for Software in Safety Critical Systems, *Software Process: Improvement and Practice*, John Wiley and Sons Ltd., 2001, Vol. 6, issue 1, pp. 47-62.