철도 안전 소프트웨어를 위한 개발 기준 연구

The development standard research for railway safety software

이영준 김장열 차경호 천세우 이장수 권기춘* 정의진**

Lee, Young-Jun Kim, Jang-Yoel Cha, Kyung-Ho Cheon, Se-Woo Lee, Jang-Soo Kwon, Ki-Choon*, Jung, Ui-Jin**

ABSTRACT

The systems such as the railway control system, satellite control system and nuclear power plant control system are the safety critical systems because the failure of them could lead to risk significant events. These softwares of digital systems must follow the life cycle process from the beginning of software development to guarantee their safety and reliability. The NRC(Nuclear Regulatory Commission) Reg Guide of nuclear fields, the RTCA/DO-178B standard which is used to acquire the certification for software in industrial aero field in European Union and United State, the DEF STAN 00-55 standard for the safety of electronic weapon in England, the IEC 601-1-4 for medical equipment and the IEC 62279 for railway system recommended the development life cycle.

This paper introduces the development process and compares each other. Also it indicates applicable development criteria for the software of systems related to railway fields and describes the detailed procedure of development criteria. We describe the procedure to make the software development criteria in nuclear filed. For the software development related to railways, the process from plan phase to maintenance phase must be satisfied. The safety and reliability is guaranteed through these standards.

---

1. 서 론

디지털 컴퓨터 기술의 급속한 발전에 힘입어 산업계 전반에 걸쳐서 아날로그 기술의 쇠퇴와 디지털 기술로의 대전환이 이루어지고 있다. 철도 분야에서도 컴퓨터 기술이 도입되고, 정보통신분야의 온라인 및 실시간 시스템이 적용되기 시작하면서 열차의 고속·고밀도 운행이 가능해지고 많은 주요 철도설비가 소프트웨어기반의 디지털 시스템으로 제어되고 있다. 아날로그 시스템과는 다르게, 디지털 시스템에 있어서 소프트웨어는 시스템이 가져야 할 모든 주요 기능을 담당하고 관리하는 핵심적인 역할을 수행한다. 즉, 디지털 시스템은 소프트웨어에 의해서 프로그래밍 되며, 이 소프트웨어는 용도에 적합하게 설계된 컴퓨터에 설치되어 제어 기능을 수행하게 된다. 디지털 시스템은 기존의 아날로그 시스템에 비해 훨씬 복잡한 제어 기능을 소프트웨어적으로 처리하여 효과적으로 수행할 수 있어서 기존의 RLL(Relay Ladder Logic)과 같은 하드웨어 프로그램 작업에 비해 효율적이다.

---

* 한국원자력연구소, 계측제어.인간공학연구부, 비회원
E-mail : yjlee426@kaeri.re.kr
TEL : (042)868-8769 FAX : (042)868-8916
** 한국철도기술연구원

유럽을 비롯한 일본, 미국 등의 선진 철도기술을 갖고 있는 여러 국가들에서는 소프트웨어 안전성을 확보하기 위한 기술개발의 필요성을 일찍이 인식하고 연구에 착수하였으며, 소프트웨어 개발 기법, 기준 및 체계를 구축하고, 인허가 기관을 설치하여, 기준 및 체계에 적합하게 개발된 소프트웨어만을 인증하여 철도 시스템에 사용할 수 있도록 허가하고 있다. 국내의 철도 시스템 분야도 도시철도 및 고속철도 등이 상용화됨에 따라 컴퓨터에 기반을 둔 디지털 시스템을 다수 사용하고 있다. 따라서 디지털 시스템을 구성하는 소프트웨어의 안전성 및 신뢰성이 매우 중요시 되고 있는 상황이다. 철도 소프트웨어의 안전성 및 신뢰성을 확보하기 위해서는 프랑스나 독일 등의 유럽 국가와 같이 자체적으로 소프트웨어를 개발 및 검증하는 표준을 제정하는 작업이 필요하다.

본 논문에서는 안전필수 소프트웨어를 사용하는 타산업의 기준들과 기존 철도소프트웨어에서 준수하고 있는 기준에 대해 살펴보고 비교분석한다. 이를 바탕으로 국내 철도 산업 환경에 적합한 철도 소프트웨어 안전기준체계를 구축하기 위한 소프트웨어의 개발 및 안전성 확보 기술을 토대를 확립한다. 본 논문의 구조는 다음과 같다. 2장에서는 원자력, 항공우주, 국방, 의료, 철도산업에서 규정하고 있는 개발공정에 대해 살펴보고 3장에서 결론을 맺는다. 3장은 철도 소프트웨어의 개발공정에서 필요한 사항들을 제시하는 것으로 결론을 맺는다.

2. 소프트웨어 안전기준 기술개발 현황

2.1 원자력소프트웨어

원자력 발전소에서 사용되는 안전 소프트웨어는 NRC(Nuclear Regulatory Commission)에서 발행한 Reg. Guide 규제지침과 이 규제지침서들이 승인한 산업표준(IEEE Standard)에 따라 개발된다. NUREG 0800 SRP(Software Review Plan)는 NRC가 어떤 관점에서 제출된 문서를 검토할 것인지를 지시하는 검토 지침서이고 Appendix 부분의 BTP-14는 소프트웨어에 대한 검토지침을 기술하고 있다. 원자력에서 사용되는 소프트웨어는 이 지침에서 제공하는 요건을 만족해야 발전소에 적용할 수 있으므로 개발 프로세스는 BTP-14의 절차에 따라 진행되고 있다.

[그림 1]에서 보는 바와 같이 BTP-14에서는 소프트웨어의 생명주기를 계획, 요구사항, 설계, 구현, 통합, 검증, 초기화, 운영 및 유지보수 단계로 나누고 있다. 각 생명주기별로 수행해야 하는 활동들도 나와 있다.

[그림 1] BTP-14 소프트웨어 생명주기

2.2 항공우주 소프트웨어

항공우주산업은 산업, 군사, 정보, 과학 분야와 같은 다양한 분야에서 서비스를 제공하고 있다. 항공기는 사람들뿐 아니라 우편물이나 산업제품들에 대한 운송을 지원해 주고 있고, 유.무인 우주선과 위성들은 과학적인 실험들과 전 세계의 통신 서비스를 지원해 주고 있다.

이러한 항공우주산업은 점점 안전에 관심을 나타내기 때문에 그 요건들이 규범화 되고 있고 많은 국가에서 서로 협력해서 이에 대한 규칙들을 작성하고 있다. 항공우주 소프트웨어의 안전과 신뢰성을 향상시키기 위하여 4개의 주요 기관들이 표준들을 개발하고 있는데 그 기관들을 다음과 같다.

- Requirements and Technical Concepts in Aviation (RTCA), Inc.
- European Space Agency (ESA)
- U.S. National Aeronautics and Space Administration (NASA)
- American Institute of Aeronautics and Astronautics(AIAA)

이러한 기관들이 서로 발행하고 있는 표준들은 항공우주 시스템의 안전을 보장하기 위해서 만족해야만 하는 요건들과 규범, 그리고 절차들에 대해서 기술되어 있다. 여기서는 RTCA/DO-178B 개발공정을 살펴본다.

이 기준은 유럽연합과 미국에서 산업 항공분야에 사용되는 소프트웨어에 대한 인증을 얻기 위해서 사용되는 표준이다. RTCA/DO-178B에서는 항공시스템에서 가져야 할 두 가지 주요한 목적이 기술되어 있다. 첫 번째로 소프트웨어의 항공운항 시스템과 장비들은 서로 호환되어야 하고 두 번째로 내부 기능들이 안전하게 수행될 수 있도록 소프트웨어가 가져야 하는 지침을 제공해야 한다는 것이다. 이 표준에서 기술하는 소프트웨어의 개발 프로세스는 계획, 개발, 확인, 형상관리, 품질보증, 그리고 인증단계의 여섯 가지 절차이고 각 단계가 만족해야 하는 요건과 내용들이 자세히 기술되어 있다.

2.3 국방 소프트웨어

국방 산업은 안전과 신뢰공학에 있어 긴 역사를 가지고 있는 분야이다. 무기 시스템은 파괴를 목적으로 만들어지지만 특정 목적을 위해서 동작되어야 하고, 안전하고 신뢰를 확보한 운영이 제일 중요하다. 국방 소프트웨어도 여러 표준들이 있고 이 표준에는 소프트웨어의 개발 프로세스에 대한 내용이 권고되고 있다. 국방 소프트웨어의 개발프로세스가 어떠한 과정으로 이루어지고 있는지 DEF STAN 00-55 표준문서를 통해 확인한다.

DEF STAN 00-55는 영국 국방성에서 공포한 표준으로 전자무기 장비의 안전을 위한 체계적인 기준이 확립되어 있다. 이 문서는 안전에 관련된 소프트웨어를 개발하고 분석하여 군사장비에 적용할 수 있도록 도와주는 역할을 하고 있다. 이 문서는 소프트웨어의 안전에 제한되어 있고 하드웨어나 인적요소, 그리고 다른 시스템의 안전에 대해서는 기술하고 있지 않다.

DEF STAN 00-55에서는 안전관련 소프트웨어의 생명주기를 소프트웨어 안전 및 신뢰성 계획, 소프트웨어 요구사항 명세, 소프트웨어 설계개발, 구현, 정형증명 및 정적분석, 동적 분석, 소프트웨어 확인 및 인증의 7가지 단계로 나누어서 개발프로세스를 기술하고 있다.

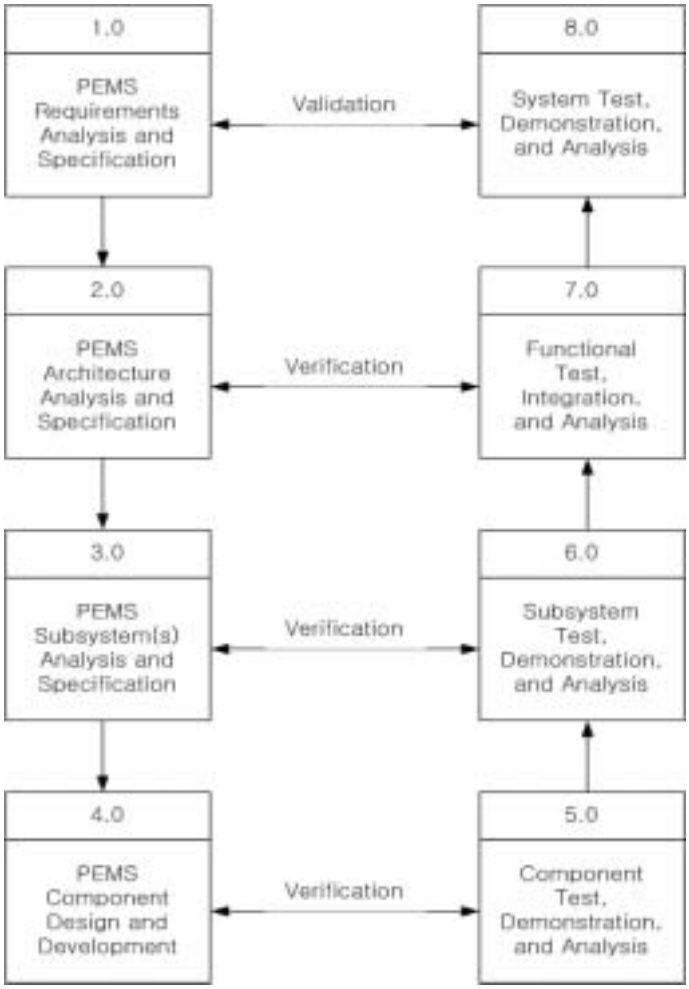
2.4 의료기기 소프트웨어

IEC 601-1-4(Medical Electrical Equipment) 문서는 프로그램이 가능한 전자의료 시스템(Programmable Electrical Medical System - PEMS)과 같은 시스템을 위한 표준이다. 이 표준에서는

장비를 개발하기 위하여 소프트웨어 생명주기 공정을 따라야 할 것을 지침하고 있으나 특별한 생명주기 공정을 요구하지는 않는다. 대신 PEMS의 개발 생명주기가 일관성이 있는지, 시스템의 위험상태에 대하여 엄격한 기준을 가지고 있는지에 대해서 살펴보고 있다. 또한 생명주기 모델이 선택되었을 경우 그 선택된 생명주기 모델은 4가지의 특별한 특성들을 가지고 진행되어야 한다.

- 잘 정의된 입력, 출력, 그리고 출력을 가진 명확한 단계
- 단계와 활동들 사이의 기술적인 정보들의 피드백
- 확인/검증 활동들은 에러의 원인까지 살펴보고 연관된 에러들을 찾기 위해 인접하고 유사한 분야에 대한 검토를 한다.
- 개발 생명주기와 위험 관리 활동들은 완전히 통합되어야 한다.

[그림 2]는 일반적인 PEMS 개발 생명주기 모델을 나타내고 있다.



[그림 2] PEMS 개발 생명주기

이 표준문서에서 제안하고 있는 개발 프로세스는 8가지 단계로 진행하고 있다. 요구사항 분석과 명세, 구조 분석과 명세, 하위 시스템 분석과 명세, 컴포넌트 설계 및 개발, 컴포넌트 시험과 분석, 하위 시스템 시험과 분석, 기능 시험과 분석, 시스템 시험과 분석 등이 이 8가지 단계이다.

2.5 철도 소프트웨어

IEC62279는 철도신호시스템의 소프트웨어측면에서 신뢰성과 안전성을 확보하기 위한 기준으로 소프트웨어의 안전성 요건을 규정하고 있다. 소프트웨어에서 필요한 안전요구사항과 관련 기능은 IEC61508과 IEC62278에 의해 규정되고 IEC62279에서는 이러한 요구사항과 기능을 달성하고자 하는 수단들을 제시한다. 이 규격의 핵심은 소프트웨어 안전 무결성 등급(Safety Integrity Level : SIL)에 있다. 안전 등급을 0에서 4까지 5단계로 구분해서 0등급은 안전과 관련이 없는 수준이며 1에서 4등급으로 갈수록 안전에 치명적인 영향을 미친다는 것을 의미한다.

이 규격에서 설명하고 있는 소프트웨어의 개발 프로세스는 소프트웨어의 요구사항, 구조, 설계 및 구현, 증명 및 시험, 소프트웨어와 하드웨어의 통합, 검증, 평가, 유지보수, 확인, 품질보증의 절차로 나누어져 있고 각 절차마다 수행할 업무와 생산할 항목들에 대해서도 정의하고 있다.

3. 결 론

철도 시스템을 개발하는 데에 있어서 시스템 엔지니어링 방법을 적용한 것은 고속철도 기술개발사업을 통해 처음으로 시도되었고, 이를 통해 프랑스로부터 시스템 엔지니어링 방법론과 유사한 방법론을 습득하였다. 시스템의 요건분석과 관리, 그리고 기능분석을 총체적으로 수행함으로써 문제정의와 설계지식의 모형화를 체계적, 효율적으로 수행하여 왔다. 그러나 현재 철도 소프트웨어를 개발할 때의 프로세스는 EN 50128과 IEC 62279의 절차들이 있으나 실제 소프트웨어의 개발 프로세스에서는 이러한 절차를 따른 경우는 거의 전무한 상태이다. EN 50128의 생명주기별 단계는 요구사항 명세, 구조명세, 설계 및 개발, 통합, 검증, 평가, 유지보수, 확인, 품질보증의 9단계로 나누어서 정의하고 각 단계마다 생산하여야 할 문서들에 대해서 열거되어 있다. 이러한 규격들은 실제 무엇을 요구하고 있는 지 파악할 수 있으나 실제 결과물로 제시하여야 하는 문서들에 대한 목차나 세부사항들에 대한 설명이 부족하다. 가령 IEEE Std 830과 같은 표준들은 요구사항 문서에 기술되어야 하는 내용과 절차, 각 장마다 필요한 사항을 세부적으로 제시하고 있다. 따라서 철도 소프트웨어의 요구사항명세를 작성할 경우 EN 50128을 우선적으로 따르되 IEEE Std. 830 도 참고하면 더욱 자세히 설명할 수 있을 것이다. 생명주기 구조단계의 구조명세는 요구사항을 가지고 소프트웨어의 컴포넌트들을 분할하여 각 컴포넌트들이 독점적인 소프트웨어로 인정받기 위하여 차별화하는 것이다. 그러나 이러한 구조는 이미 요구명세와 설계명세에 녹아들어가 있고 요구사항명세와 설계명세를 IEEE Std.를 가지고 작성하다면 더 이상 필요하지 않을 것이다. 다만 소프트웨어가 객체지향적인 방법론을 가지고 접근하고 있고 그 객체의 크기가 일반소프트웨어의 크기만큼 규모와 역할이 상당하다면 구조명세를 작성하여 객체간의 독립성을 표현하는 것도 좋은 방법이다. 물론 소프트웨어의 크기가 일정이상의 규모와 역할을 하다는 가정을 전제한다. 각 단계마다 생산되는 시험 명세서는 그 내용에 따라 시험 계획서, 시험 절차서 등으로 분류하는 것도 좋은 방법 중의 하나이다. 원자력 분야의 Standard Review Plan 에서는 시험의 중요성을 부각시키기 위해 시험을 위한 계획과 절차로 분류하고 각 단계에서 수행해야 할 시험에 대해서 권고하고 있다. 품질보증 활동은 개발 프로세스에 포함될 하나의 단계라고 하기보다는 더욱 상위의 위치에서 그 활동을 명시하는 것이 효율적이다. 요구사항부터 확인활동까지의 생명주기과정은 품질보증을 위한 하나의 과정에 속한다고도 할 수 있기 때문이다.

4. 참고문헌

[1] IEEE Std. 7-4.3.2-2003, "IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations".

[2] USNRC RG1.173, "Development of Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants", 1997.

[3] USNRC NUREG-0800, "Software Review Plan", Branch Technical Position-14.

[4] RTCA. "Software considerations in airborne systems and equipment certification", DO-178B, Requirements ad Technical Concepts for Aeronautics, 1992.

[5] DEF-STAN 00-55, "Requirements for Safety-Related Software in Defence Equipment"

[6] IEC 601-1-4(1996-06), Medical Electrical Equipment

[7] EN 50128, "Railway Aapplications - Communications, Signalling and Processing Systems - Software for Railway Control and Protection Systems," March 2001.

[8] IEEE Std. 1012-1998(Revision of IEEE Std. 1012-1986), IEEE Std. for Software Verification and Validation.

[9] 유일상 외, 차세대 고속전철 시스템엔지니어링 체계 모델 개발, 한국철도학회논문집 2002.