철도 소프트웨어 개발 및 검증을 위한 지침

The guideline for development and verification of railway software

이영준 최종균 차경호 천세우 이장수 권기춘* 정의진**

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

ABSTRACT

The Railroad Safety Act's regulation reads as follows. “The Minister of Construction and Transportation may qualify and authorize the product to guarantee performance and safety of parts, machine, and device used in Railway fields.” Another regulation reads as follows. “The guidelines about targets, standards, and procedures of Quality and Authority in first provision are decided as Ministry of Construction and Transportation Decree.” The software used in rail cars and facilities is considered as a railway product. Therefore, it is qualified and authorized for acquiring the safety of rail cars and facilities. The software businesses shall again a Quality and Authority for applying a software to the rail cars and facilities.

This paper regulates some guidelines that are needed to develop a software. The procedures that a software developer performs are divided by plan, requirement, design, implementation, and maintenance. The procedures that a software verification person performs are classified by verification plan, requirement verification, design verification, implementation verification, testing verification, maintenance verification, and safety activity. The entire processes and detailed activities to develope and verify a software are described as new guidelines.

---

1. 서 론

철도 안전법 제27조 1항에는 “건설교통부장관은 철도에 사용되는 부품, 기기 또는 장치 등의 성능 및 안전성을 확보하기 위하여 철도용품에 대한 품질인증을 할 수 있다”라고 규정하고 있으며, 제27조 3항에는 “제1항의 규정에 의한 품질인증의 대상, 기준 및 절차 등에 관하여 필요한 사항은 건설교통부령으로 정한다”라고 규정하고 있다. 이러한 규정을 해석할 때 철도차량 및 시설에 사용하는 소프트웨어는 철도용품의 하나로 볼 수 있으며, 철도차량 및 시설의 안전성을 확보하기 위한 품질인증의 대상이다. 따라서 소프트웨어 사업자는 소프트웨어를 철도차량 또는 시설에 사용하기 위해서 품질인증을 받아야 한다.

본 논문에서는 소프트웨어가 철도 차량 및 시설에 사용하기 위해서 지켜야 할 지침에 대해 기술한다. 소프트웨어의 안전등급을 분류하고 품질보증을 위한 활동들을 정의하며 형상관리에 대한 지침도 규정한다. 소프트웨어의 개발자가 수행하여야 하는 절차를 계획, 요구사항, 설계, 구현, 유지보수지침으로 규정하고 검증자가 수행하여야 하는 각 단계별 검증에 대해서도 규정한다.

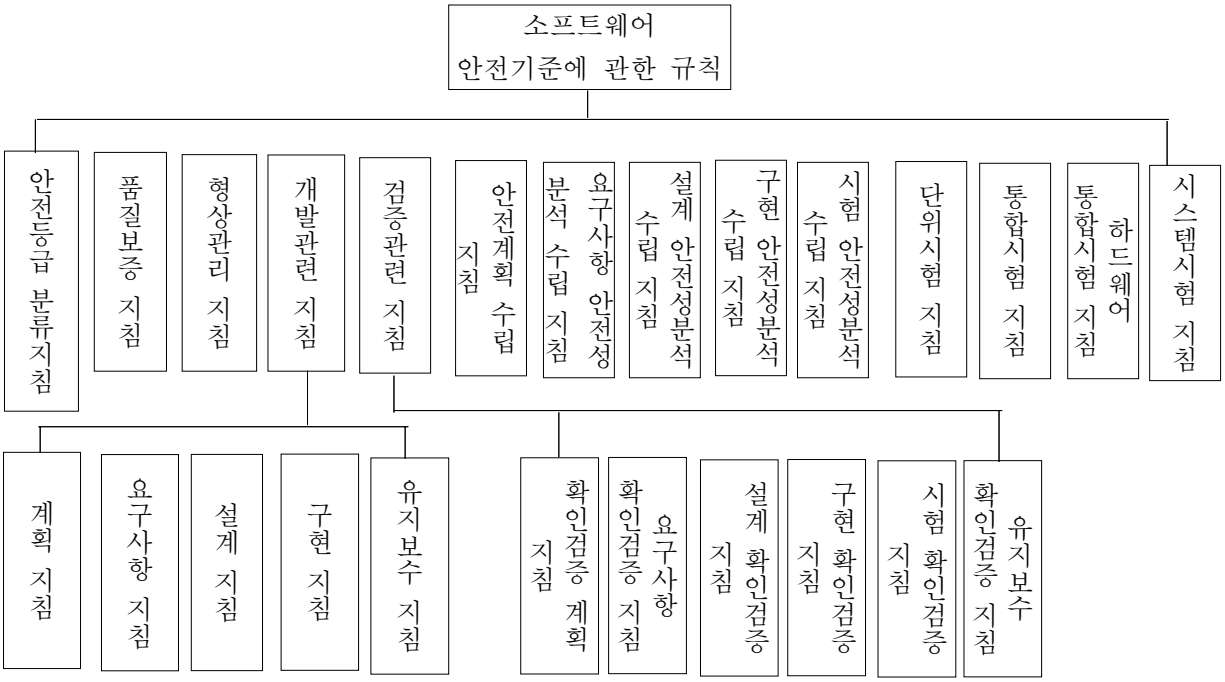
---

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

본 논문에서는 철도 차량 및 시설에 사용하는 안전 소프트웨어에 대한 지침을 제시하고 그 내용을 간략하게 살펴본다. 본 논문의 구조는 다음과 같다. 2장에서는 철도 소프트웨어의 지침 구조에 대해서 살펴보고 안전등급 분류지침, 품질보증지침, 형상관리지침의 내용을 정의하고 3장에서는 소프트웨어의 개발에 관련된 지침을 살펴본다. 4장에서는 소프트웨어의 검증에 관련된 지침을 살펴보고 5장에서 결론을 맺는다.

2. 철도 소프트웨어의 지침 구조

철도 소프트웨어에 대한 지침은 개발 및 검증을 위한 생명주기 단계마다 수행하여야 하는 일련의 활동들을 정의한다. 이러한 지침들에 대한 상위 규칙을 마련하여 지침에 대한 근거를 제시하였는데 이 규칙 역시 철도안전법 제27조를 바탕으로 마련된 것이다.
철도 소프트웨어의 규칙 및 지침에 대한 구조는 그림1 과 같은 형태로 표현할 수 있다.



[그림 1] 철도 소프트웨어 규칙 및 지침 구조

이 지침들은 크게 안전등급 분류지침, 품질보증 지침, 형상관리 지침과 개발 관련 지침, 검증 관련 지침, 안전성 관련 지침, 시험 관련 지침으로 나눌 수가 있다.
안전등급 분류지침은 IEC62279 및 IEC15026을 바탕으로 소프트웨어의 안전등급을 분류하고 있으며 소프트웨어의 안전등급 분류체계, 소프트웨어 안전등급 결정에 대한 사항들을 기술한다. 또한 안전등급을 결정하기 위한 선행 조건과 안전등급 결정을 위한 방법에 대해 세분화하여 기술한다.
품질보증 지침은 품질보증활동에 대한 상세기준을 규정하고 있다. 품질문제 제기, 품질문제에 대한 해결방안 제시, 해결방안의 이행여부 확인을 위해 권한과 독립성이 있는 조직을 구성하도록 규정하며 사업

자가 품질보증 계획을 수립할 때 고려하여야 하는 사항들을 기술하고 있다.
형상관리 지침에서 규정하는 사항은 형상관리 업무를 수행하는 조직 및 책임, 계획활동, 형상항목을 선정하기 위한 기준 수립, 형상 항목들에 대한 통제, 사업자가 수행하야야 하는 형상상태 업무에 관한 내용들이다. 이러한 기준을 만족하도록 형상관리를 수행할 때 소프트웨어의 품질을 향상시킬 수 있게 된다.

3. 철도 소프트웨어 개발 지침

철도 소프트웨어 개발 지침은 소프트웨어의 개발자를 위한 규정이라 할 수 있다. 그림 1에서 보는 바와 같이 개발관련 지침은 계획지침, 요구사항지침, 설계지침, 구현지침, 유지보수지침으로 구성되어 있다. 철도 소프트웨어를 개발하기 위한 사업자나 개인은 이러한 규정들을 준수하여야 소프트웨어를 철도분야에 적용할 수 있는 근거를 마련할 수 있는 것이다. 구현된 소프트웨어에 대한 시험활동은 시험관련 지침에서 개발자의 역할을 정의하고 있으므로 해당 시험을 수행하여야 한다. 소프트웨어에 대한 시험도 개발활동의 일부분이므로 개발 관련 지침으로 그룹화 할 수 있으나 시험에 관련된 활동은 개발자 뿐 아니라 검증자도 수행하여야 하는 활동이므로 시험관련 활동을 별도의 지침으로 규정하고 있다. 이는 개발절차와 검증절차보다도 시험활동이 더욱 중요하기 때문에 별도로 규정한 것이다. 소프트웨어의 궁극적인 목표는 완전하고 오류없는 제품을 만드는 것이다. 따라서 구현된 제품에 대한 기능 및 성능을 보장하기 위해서는 모든 사례에 대한 시험을 수행하여야 한다. 개발 관련 지침에 대한 내용을 간략하게 살펴보면 다음과 같다.

● 계획 지침

계획 지침은 소프트웨어를 개발하기 전에 수행하는 활동이다. 계획을 수립한대로 개발이 이루어지고 있는 지를 확인하는 것이 검증활동의 일부분이므로 정확한 계획을 세워야 한다. 계획 지침에서 요구하는 계획 활동의 종류는 관리계획, 개발계획, 통합계획, 유지보수계획이다.
관리계획에서는 과제관리를 통해 체계적인 공정계획 활동들을 관리할 것을 명시하고 있고 개발계획에서는 생명주기 단계 목표와 역할, 소프트웨어 생명주기 모델, 기술적 개발 노력의 관리 전략, 과제의 위험요소 파악, 설계 결과물과 개발공정의 관리 척도 정의, 개발에 적용된 생명주기 행위 구분, 개발 일정 파악, 개발동안 사용할 방법 및 도구 파악, 개발에서 준수하는 국내외 표준들을 규정할 것을 명시한다. 통합계획에서는 단위 기기 통합, 시스템 통합과 같이 통합의 수준을 파악하고 통합할 대상들이 무엇인지 분류하는 것이다. 유지보수 계획은 소프트웨어 유지보수에 관한 지침이 별도로 존재하므로 계획 지침에서는 세부 내용을 명시하지는 않는다. 다만 유지보수 계획을 세워야 함을 정의하고 있다.

● 요구사항 지침

소프트웨어의 요구사항은 컴퓨터에 설치될 소프트웨어가 가져야 하는 기능으로써 시스템의 요구사항을 반영하여야 한다. 이러한 요구사항은 소프트웨어의 컴포넌트는 물론 소프트웨어가 운용될 환경에 대한 특정한 요구사항 및 소프트웨어의 기능적인 특징, 요구되는 특성 등을 고려하여야 하는 것이다. 요구사항 지침에서 요구하는 규정은 소프트웨어의 요구사항에 대한 개요를 기술하고 시스템과 소프트웨어의 경계를 구별하는 것이다. 또한 입력변수와 출력변수를 정의하여 시스템의 연계에 대한 분석을 수행하고 컴퓨터 시스템의 기능적 행위를 정의하는 것이다. 컴퓨터 시스템의 시간적 요구사항을 정의하여야 하며 소프트웨어 설계 제약사항에 대해서도 정의하여야 한다. 요구사항 지침에서는 소프트웨어 요구사항 명

세서의 목차를 부록으로 제시하고 있다. 그러나 강제적인 사항이 아니므로 소프트웨어의 특성에 따라 다른 목차를 사용할 수도 있다.

● **설계 지침**

요구사항 분석은 소프트웨어의 개념에 초점을 맞추지만 소프트웨어 설계는 컴퓨터의 개념으로 초점이 이동된다. 요구사항은 무엇을 설계할 것인가에 대한 사용자의 요구사항이고 설계는 이러한 목표를 컴퓨터에서 어떻게 실현할 것인가를 결정하는 과정이라 할 수 있다. 설계지침에서 정의된 규정은 소프트웨어의 구조를 파악하고 소프트웨어와의 연계에 대해 설계하는 것이다. 또한 모든 설계 개체의 내부적인 사항들을 기술하는 상세설계를 요구하고 있다. 상세설계는 구현에 필요한 개체들의 속성과 행위들을 제공하는 것으로써 구조적인 언어나 도식적인 방법을 사용하여 기술할 수 있다. 설계 지침에서는 소프트웨어 설계 명세서의 목차를 부록으로 제시하고 있다. 물론 제시된 목차는 설계명세서를 기술하기 위한 일반적인 사항들로 구성되어 있으므로 필요한 경우 다른 목차를 사용할 수 있다.

● **구현 지침**

소프트웨어 구현은 소프트웨어 설계를 실제 코드로 변환하는 과정이다. 따라서 설계 개체가 코드로 구현되어 오류없이 수행되도록 하기 위해서는 코드에 대한 규칙을 준수하여야 한다. 따라서 구현 지침은 코드의 신뢰성, 견고성, 추적성, 유지보수성, 응집성, 유연성, 이식성을 강조하고 있다.

● **유지보수 지침**

유지보수는 소프트웨어 결과물의 인수 후 발생한 결함을 고치거나, 성능을 개선 또는 변경된 환경에 적응하도록 소프트웨어를 변경하는 행위를 말한다. 유지보수 지침에서 규정한 조항은 유지보수에 대한 계획내용과 유지보수를 수행하기 위한 절차와 방법을 기술한다. 유지보수 계획 문서에 대한 목차를 제시하고 있다.

4. 철도 소프트웨어 검증 지침

철도 소프트웨어의 검증에 관련된 지침의 종류는 확인검증 계획 지침, 요구사항의 확인검증 지침, 설계의 확인검증 지침, 구현의 확인검증 지침, 유지보수의 확인검증 지침이 있다. 이러한 검증 지침의 규정은 개발된 소프트웨어가 개발관련 지침을 준수하였는가를 확인하는 것부터 시작한다. 또한 각 단계별로 검증을 수행하는 방법과 절차들을 정의하고 있고 개발 관련 지침과 마찬가지로 검증 활동을 위한 목차를 제시하고 있다. 검증의 방법은 주로 추적성 분석과 적합성 분석을 통해 수행된다. 각 단계별로 수행하여야 하는 역할 또한 정의되어 있으며 검증문서를 작성하기 위한 목차도 제시하고 있다. 검증 지침은 IEEE1012 와 IEC62279에서 권고한 절차들도 준수하도록 작성되어 있다. 안전성 관련 지침은 크게 보면 검증의 일환이라고 할 수 있지만 안전성분석은 안전등급 분류에 따라 구별되어 수행될 수 있는 과정이므로 별도의 지침으로 존재한다. 또한 검증자가 수행하여야 하는 시험도 시험관련 지침에 포함되어 있으므로 검증 지침의 항목으로 정의되어 있지 않다. 다만 검증단계별로 시험에 대한 계획들을 세울 것을 명시하고 있다.

5. 결 론

현재 철도 소프트웨어를 개발할 때의 사용되는 국제 규격으로는 EN 50128과 IEC 62279의 절차들이 있다. 그러나 국내의 철도 소프트웨어의 개발에서 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. 따라서 이러한 세부사항들을 지침형태의 문서로 규정하여 철도 소프트웨어에 적용하는 것이 필요하다.

철도안전법에서 차량용품에 대한 품질인증을 강제하고 있으며 소프트웨어는 차량용품의 범위에 속하는 것으로 볼 수 있으므로 소프트웨어도 품질인증을 획득하여야 한다. 그러나 소프트웨어에 대한 품질인증은 개발 및 검증의 프로세스를 준수하고 제품의 기능 및 성능을 시험하여 획득할 수 있는 것이다. 따라서 이러한 과정들이 어떤 방법과 절차를 가지고 진행되는 지를 정의하기 위하여 각 지침들이 필요한 것이다. 소프트웨어에 대한 등급을 분류하고 품질보증의 절차를 정의하며 생명주기 모델에 따라 개발을 수행하고 개발된 제품에 대한 검증을 수행하는 방법들에 대해서 지침서에 기술하고 있다. 또한 안전성을 확보하기 위한 안전성분석, 개발자 및 검증자가 독립적으로 수행하는 시험에 대해서도 지침서에 기술되고 있다. 이러한 지침들을 정확하게 준수하는 것만이 소프트웨어에 대한 전체품질을 보장할 수 있는 것이다.

6. 참고문헌

[1] IEEE Std. 1012-2004, IEEE Standard for Software Verification and Validation.

[2] IEC 62279, Railway Applications - Communications Signalling And Processing Systems - Software For Control And Protection Systems, 2002.

[3] IEC 15026, Information technology -- System and software integrity levels.

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

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

[6] 이영준 외, 철도 안전 소프트웨어를 위한 개발기준 연구, 한국철도학회논문집, 2007