철도소프트웨어의 개발을 위한 체계적 접근법 제안

정의진*, 신경호
한국철도기술연구원

Suggestion of Systematic Approach for Developing Railway Software

Eui-Jin Joung*, Kyung-ho Shin
KRRI(Korea Railroad Research Institute)

Abstract - Safety critical systems are those in which a failure can have serious and irreversible consequences. Nowadays digital technology has been rapidly applied to critical system such as railways, airplanes, nuclear power plants, and vehicles. The main difference between analog system and digital system is that the software is the key component of the digital system. The digital system performs more varying and highly complex functions efficiently compared to the existing analog system because software can be flexibly designed and implemented. The flexible design make it difficult to predict the software failures. This paper reviews safety standard and criteria for safety critical system such as railway system and suggests development methodology, ordering management and assessment process for railway software with more detail description.

1. 서 론

철도시스템 운영의 주요 관심사 중의 하나는 안전성 확보를 들 수 있다. 따라서 다중 보호개념을 적용한 fail-safe 특성이 설계에서 중요시되고 있다. 안전성 확보를 위해서는 소자특성상 fail-safe 특성이 강하게 나타나는 릴레이를 주로 사용하여 왔다. 그러나 안전성뿐만 아니라 편의성도 중요한 대중 교통수단이란 점 때문에 여러 가지 새로운 기능들이 요구되고 있으며, 릴레이로 구현하기에는 비효율적인 부분도 많아지게 되었다. 따라서 안전과 직접적으로 관련 없는 설비에 대해서는 소프트웨어로 구현하여 적은 공간에서 빠르게 원하는 기능을 수행하도록 하고 있으며, 안전과 직접적으로 관련되어 있는 설비에 있어서도 차츰 소프트웨어로 구현해 나가는 영역을 넓히고 있는 추세이다.

소프트웨어로 구현할 경우의 기대효과로는 예비부품을 확보할 필요성이 없으며, 따라서 부품 단종에 대한 우려가 필요 없게 된다. 또한 기기 노후화로 인한 설비의 성능 저하도 고려하지 않아도 된다. 소프트웨어의 자기진단 및 자동시험으로 보수 및 정기 시험에 소요되는 인력의 감소 및 작업시간을 단축할 수 있다는 장점이 있다. 그러나 소프트웨어로 시스템의 기능을 구현할 경우에는 소프트웨어 프로세스 처리를 육안으로 확인할 수 없을 뿐만 아니라 한 프로세스를 거친 후 의도하는 다음 프로세스를 정확히 수행한다는 보장할 수가 없다. 따라서 소프트웨어로 구현한 기기의 정확한 기능 수행 보장 및 품질 및 신뢰성 확보가 무엇보다도 중요하다.

현재까지 개발기간, 비용 등의 이유로 철도분야의 경우 소프트웨어의 기능구현에만 중점을 둔 것이 사실이다. 그러나 소프트웨어의 특성상 불확실성이 존재하며, 이러한 불확실성을 염두에 두지 않고, 안전성 검증없이 소프트웨어를 사용할 경우, 만약의 사태로 인해 사고로 이어진다면 그 피해는 매우 엄청나다고 할 수 있다. 이미 선진국 중에서 안전필수 소프트웨어를 다루는 분야에 대하여 소프트웨어의 안전을 확보하기 위한 기준을 제시하고 있으며, 이를 검증하는 체계 또한 갖추어 만일의 사태에 대비하고 있다. 철도 소프트웨어에 있어서도 안전기준을 제시하는 작업을 진행 중에 있으며, 제시된 안전기준에 맞게 철도소프트웨어가 제대로 개발되었는지 검증하고, 인증하는 체계 또한 구축 중에 있다.

무엇보다도 소프트웨어 개발이라는 것은 최종 산출물만을 의미하지는 않는다. Lifecycle 각 단계마다 제시되어야 하는 문서 또한 중요하며, 이 문서를 근거로 검증 및 평가가 이루어져 소프트웨어의 품질을 보증할 수 있기 때문이다. 따라서 본 연구에서는 소프트웨어 수명주기 각 단계마다 수행하여야 하는 업무에 대한 절차, 양식, 기법을 정리한 개발방법론을 제시하였으며, 발주자나 평가자 입장에서 중요한 발주 및 평가 프로세스에 대해 논하고자 한다.

2. 철도소프트웨어 안전기준

소프트웨어 개발과 관련하여서는 일정 지연, 비용 초과, 고객의 불만족 등의 위험요인이 있으며, 철도시스템과 같은 안전필수시스템의 경우, 소프트웨어로 인한 인명손상, 재산상의 손실 등을 고려한 안전성 확보 또한 고려되어야 한다. 품질 좋은 소프트웨어를 만들려는 노력으로 제품자체의 품질을 향상시키는 방법과 제품을 개발하는 프로세스 관리를 통한 문제해결 방안을 생각할 수 있다. 철도소프트웨어의 신뢰성 및 안전성을 확인하고 보증하기 위해서는 제품관점에서 좋은 제품을 만들고, 정확한 시험으로 개발된 제품의 품질이 원하는 수준에 도달했는지를 판단하는 방법이 있을 수 있으며, 이와는 다른 관점으로 좋은 제품은 좋은 조직 체계에서 만들어진다는 프로세스 관점을 생각해 볼 수 있다. 물론 개발 대상 분야에 대한 지식 또한 중요하다. 다음 그림 1은 철도소프트웨어의 품질을 확보하기 위한 각 관점과 관련된 규격들을 분류하여 나타낸 것이다.

철도분야의 안전관련 표준으로는 전기전자 규격인 IEC 61508과 철도관련 규격인 IEC 62278, IEC 62279 규격을 대표적으로 들 수 있다. 이중 IEC 62279는 유럽전기전자표준규격인 CENELEC의 EN 50128에서 국제규격으로 전환된 규격으로 철도분야 소프트웨어에 대해 다루고 있다. 프로세스 관점으로는 미국 SEI (Software Engineering Institute)의 CMMI(Capability Maturity

Model Integration)와 ISO/IEC 15504 (SPICE: Software Process Improvement and Capability dEtermination)를 들 수 있으며, 소프트웨어 관련 프로세스에 대하여 성숙도 등급을 매겨 관리하고 있다. 제품 관점으로의 소프트웨어 품질특성을 정의한 ISO/IEC 9126과 소프트웨어 제품의 품질특성 평가를 다루고 있는 ISO/IEC 14598을 들 수 있다.

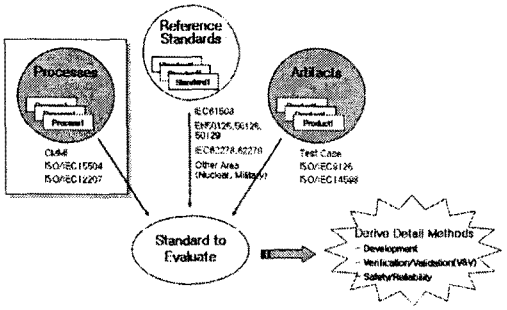

그림 1. 철도소프트웨어 안전기준 관련 표준의 범주

국토해양부 사업인 철도종합안전기술개발사업 중 한국철도기술연구원 주관으로 2004년부터 2008년까지 수행하는 "철도소프트웨어 안전기준 및 체계구축" 과제에서 제시되는 안전기준은 상기 근간이 되는 국제 표준 외에 여러 국제규격(IEC, ISO 등), 국내규격(KS 등) 및 산업체 표준 (IEEE 표준 등) 등과 서로 상충되지 않도록 하며, 국내 환경을 고려하여 안전기준을 제시하고 있다.

본 과제에서 제시하는 안전기준으로는 규칙레벨과 지침레벨로 구성되어 있으며, 규칙레벨로 수명주기별 안전기준(안)을 제시하고 있다. 철도소프트웨어 안전기준에서 제시하는 수명주기로는 개발, 검증, 시험, 안전성 분석의 4가지 수명주기로 분류하고 각각에 대하여 지침레벨로 각각의 수명주기에 대한 세부지침(안)을 개발 중에 있다.

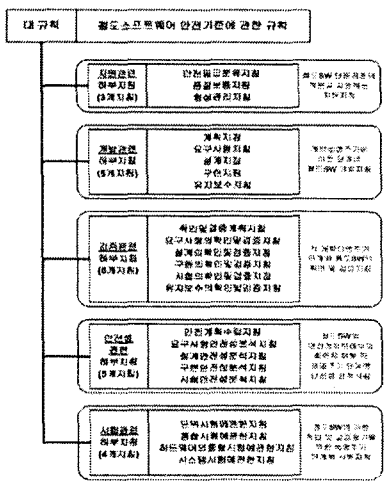

그림 2. 철도소프트웨어 안전기준(안) 구성

그림 2는 제안한 철도소프트웨어 안전기준(안)으로 규칙과 해당 세부지침 목록을 정리하여 나타내었다. 세부지침 중 지원관련 하부지침 그룹은 개발, 검증, 안전성, 시험 관련한 수명주기 전반에 걸쳐 적용해야만 하는 사항을 정한 지침이며, 그 다음에 개발, 검증, 안전성, 시험 관련한 수명주기에 대하여 각각 하부지침 그룹을 두어 세부지침을 기술하였다.

제시한 안전기준중 세부지침(안)의 일부 내용을 발췌한 것으로 구성은 조항 및 해설, 근거기준으로 작성되어 있다. 규칙과 세부지침으로 구성되는 철도소프트웨어 안전기준이 법적인 성격을 갖추고 있다 보니 구체적으로 수행하여야 할 절차가 언급되어 있지 않아 실제로 개발업체나 평가기관에서 안전기준을 적용할 경우에 많은 혼선이 야기될 수 있다. 따라서 철도소프트웨어를 개발, 검증, 시험, 안전성 분석하는 Lifecycle 각 단계에서 수행하여야 하는 업무를 체계적으로 규정한 개발방법론을 마련할 필요가 있다.

3. 철도소프트웨어 안전기준 적용을 위한 세부 절차

3.1 소프트웨어 개발방법론

소프트웨어 개발방법론이란 소프트웨어를 개발하기 위해 개발 조직의 환경과 소프트웨어 및 시스템을 사용할 사용자의 환경에 적합한 소프트웨어 개발방법을 일컫는다. 이러한 개발방법론의 적용은 1970년대 이후 구조적 방법론을 거쳐, 1980년대의 정보공학 방법론, 1990년 이후의 객체지향 방법론이 제기되었다. 각 방법론마다 장단점이 있어 개발하려는 시스템의 특징을 고려하여 개발방법론을 채택하여야 한다.

예를 들어 구조적 방법론은 단순한 업무처리 시스템의 개발에는 효과적이지만 대규모의 복잡한 시스템 개발에는 적합하지 않으며, 정보공학 방법론은 대규모 시스템 개발에는 적합하나 복잡한 시스템의 모델링에 부적합하다. 최근 대두된 객체지향 방법론은 소프트웨어의 확장이나 변화를 용이하게 해줄 뿐만 아니라 기존 소프트웨어의 재사용성을 증가시켜 준다는 장점이 있어 정보공학 방법론을 대체하고 있는 상황이다.

이러한 개발방법론을 적용함으로써 수요자는 균질한 산출물을 얻을 수 있으며, 체계적인 품질보증을 받을 수 있고, 프로젝트 진행 중에 품질을 확인할 수 있다. 공급자는 프로세스 기반의 개발을 할 수 있어 체계적인 품질보증이 가능하고, 표준을 쉽게 만들 수 있으며, 체계적이고 구체적인 계획을 수립할 수 있어서 불필요한 일을 최소화 할 수 있으며, 개발자의 경우 프로세스에 기반한 개발활동이 가능해져 계획에 따라 개발을 할 수 있으며, 결함유발의 가장 큰 원인인 요구사항을 조기에 확정할 수 있어서 시간과 노력을 줄일 수 있으며, 업무의 중복이나 누락을 최소화 할 수 있는 장점이 있다.

3.2 철도소프트웨어 개발방법론

Safety-related 철도소프트웨어 개발방법론은 철도분야에서 특히 강조되는 안전과 관련된 소프트웨어를 개발할 때 적용할 수 있는 접근방법을 제공하기 위해 개발되었다. 본 방법론은 절차서, 양식서, 기법서의 세 부분으로 구성되어 있으며, 절차서는 방법론을 구성하는 각 단계와 각 단계에 포함된 활동을 보여준다. 각 단계는 개발, V&V 및 안전의 3분야의 활동으로 구성되어 있으며, 활동들은 주어진 입력을 받아들여 출력을 생성하기 위한 과정을 나타내었다.

양식서는 절차서에서 정의된 입.출력물을 작성하기 위한 format을 정한 것으로 목차 및 그 구성 내용을 설명하고 있다. 이 양식서를 활용함으로써 사용자는 보다 쉽게 방법론에서 원하는 입.출력물을 확인할 수 있게 된다.

기법서는 절차서에 기술된 활동으로 설명하기에는 보다 기술적인 내용을 포함하고 있는 항목들을 모아둔 것으로 이러한 기법들은 기술이 발전되면서 지속적으로 확대해 나갈 수 있다.

Safety-related 철도소프트웨어에 대한 개발방법론은 다음의 7단계로 구성된다.

1) 철도소프트웨어 계획 수립 단계
2) 철도소프트웨어 요구사항 명세단계
3) 철도소프트웨어 설계 단계
4) 철도소프트웨어 모듈 설계 단계
5) 철도소프트웨어 구축 단계
6) 철도소프트웨어 통합 단계
7) 철도소프트웨어 하드웨어 통합 단계

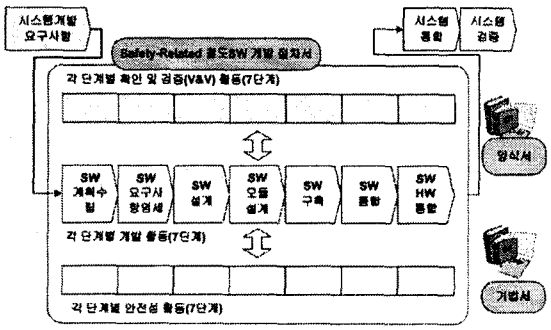

그림 3. 안전관련 철도소프트웨어 개발방법론

본 방법론의 절차서에서는 각 수행단계에 대한 활동 및 각 단계의 담당자, V&V담당자, 안전담당자의 담당자들간의 역할을 제시하고 있다. 또한 구체적으로 해당 단계의 입출력문서 및 수행내용을 기술하였다. 일반 소프트웨어 개발방법론에서는 V&V담당자가 안전담당자의 역할까지 포함하여 품질관리 관점에서 소프트웨어의 확인 및 검증을 수행하는데 대하여 안전을 중요시하는 철도시스템의 특성상 안전담당자 영역을 따로 구분하여 제시하였고, 안전담당자가 작성검토한 문건은 바로 발주기관 또는 제 3 안전성 인증기관의 검토 문건이 되어 최종 검토를 받을 수 있도록 구성하였다.

3.3 철도소프트웨어 발주 및 평가 프로세스

Safety-related 철도소프트웨어 발주관리방법론은 철도분야에서 특히 강조되는 안전과 관련된 소프트웨어를 발주할 때 적용할 수 있는 접근방법을 제공하기 위해 개발되었다. 본 발주관리방법론은 Safety-related 소프트웨어 개발방법론과 연계하여 활용될 수 있도록 작성되었다. 따라서 본 발주관리방법론은 Safety-related 철도소프트웨어를 발주하는 조직에서 활용하고, 선정된 공급자는 Safety-related 철도소프트웨어 개발방법론을 적용하여 소프트웨어를 개발할 때 효과적으로 적용할 수 있을 것이다.

본 발주관리방법론 또한 철도소프트웨어 개발방법론에서와 마찬가지로 절차서, 양식서, 기법서의 세 부분으로 구성되어 있다. 또한 발주 준비, 공급자 선정, 공급자 관리, 검수의 4개의 단계로 구성된다.

본 발주관리방법론의 절차서에서는 각 수행단계에 대한 활동 및 실무부서, 발주부서, 계약부서의 담당자와 안전담당자, 공급자들간의 역할을 제시하고 있다. 또한 구체적으로 해당 단계의 입출력문서 및 수행내용을 기술하고 있다.

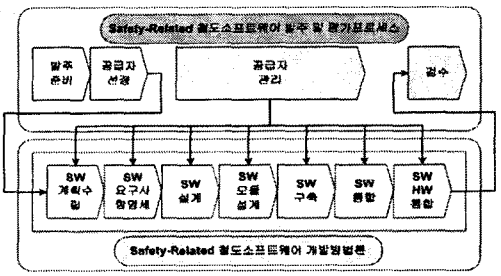

그림 4. 안전관련 철도소프트웨어 발주 및 평가 프로세스

4. 결 론

본래 불확실성이 존재하는 소프트웨어를 철도와 같이 안전성이 중요한 시스템에 적용하기 위해서는 철저한 안전성 검증이 필요하다. 원자력, 항공, 국방분야의 경우에서도 각기 시스템에 맞추어 품질보증 체계를 구축하고 있음을 보아도 알 수 있다. 철도소프트웨어의 경우 안전성을 확보하고 품질 좋은 소프트웨어를 개발하기 위해서 프로세스관점 및 제품관점의 접근이 필요한데 프로세스 성숙도 향상 관점에서는 개발하고자 하는 소프트웨어의 품질을 확보하고자 CMMI나 SPICE(ISO/IEC 15504)에서 제시하는 여러 절차 및 프로세스를 따르도록 함으로써 소프트웨어 개발조직의 성숙도를 향상시키고자 하고 있으며, 제품관점의 접근법으로는 정형기법에 의한 개발 및 검증이나, 개발 초기부터 제시한 도출한 Test Case에 따라 시험을 수행하여 소프트웨어의 품질을 향상시키는 방법을 고려하고 있다.

본 프로젝트에서는 앞서 언급한 프로세스관점 및 제품관점의 소프트웨어 품질향상 방법을 감안하여, 철도소프트웨어에 대한 안전기준을 제시하였으며, 제시된 안전기준의 현장 적용성을 높이기 위하여 절차서, 양식서, 기법서로 구성된 안전관련 철도소프트웨어에 대한 개발방법론을 제시하였다. 또한 철도소프트웨어를 발주하고, 운영하는 기관을 위하여 발주 및 평가프로세스를 제시하였다. 본 연구가 기반이 취약한 철도소프트웨어 산업의 육성에 기여하길 기대한다.

[참 고 문 헌]

[1] IEC 62278, "Railway application - The specification and demonstration of dependability, reliability, availability, maintainability and safety (RAMS)", March, 2002

[2] IEC 62279, "Railway application - Software for railway control and protection system", June, 2002

[3] CENELEC EN50129, "Railway application -Safety related electronic systems for signaling", April, 2000

[4] ISO/IEC 15504 "Information Technology-Process Assessment-Part 1~5"

[5] ISO/IEC 12207 "Information Technology- Software lifecycle processes"

[6] ISO/IEC 9126 "Information Technology-Software Quality Characteristics and Metrics-Part 1,2,3"

[7] ISO/IEC 14598 "Information Technology-Software Product Evaluation-Part 1~6"

[8] 정의진, 철도소프트웨어 안전기준 및 체계 구축 3차년도 보고서, 한국철도기술연구원, 2007