

안전을 고려한 철도소프트웨어 개발방법론 도출방안 연구

정의진*, 신경호
한국철도기술연구원

Development of Software Development Methodology with Aspect of Railway Safety

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, 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 software development methodology for 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 구현하기에 비효율적인 부분도 많아지게 되었다. 따라서 안전과 직접적으로 관련 없는 설비에 대해서는 소프트웨어로 구현하여 적은 공간에서 빠르게 원하는 기능을 수행하게 하려는 상황이며, 안전과 직접적으로 관련되어 있는 설비에 있어서도 차츰 소프트웨어로 구현해 나가고 있는 추세이다.

소프트웨어로 구현할 경우의 기대효과로는 예비부품의 확보 어려움이 없으며, 부품 단종에 대한 우려가 없다. 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. 소프트웨어로 구현한 기기의 정확한 기능 수행 보장 및 품질 및 신뢰성 확보가 중요하다.

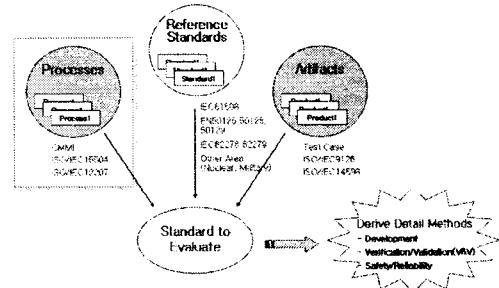
현재까지 개발기간, 비용 등의 이유로 철도분야의 경우 소프트웨어의 기능구현에만 중점을 둔 것이 사실이다. 그러나 소프트웨어의 특성상 불확실성이 존재하며, 이러한 불확실성을 염두에 두지 않고, 안전성 검증없이 소프트웨어를 사용할 경우, 만약의 사태로 인해 사고로 이어진다면 그 피해는 매우 엄청나다고 할 수 있다. 이 미 선진국 중에서 안전필수 소프트웨어를 다루는 분야에

서는 소프트웨어의 안전을 확보하기 위한 기준을 제시하고, 검증하는 체계를 갖추어 만일의 사태에 대비하고 있다. 철도 소프트웨어에 있어서도 안전기준을 제시하는 작업을 진행 중에 있으며, 제시된 안전기준에 맞게 철도 소프트웨어가 제대로 개발되었는지 검증하고, 인증하는 체계 또한 구축 중에 있다.

소프트웨어 개발에 있어서는 최종 산출물도 중요하지만 Lifecycle 각 단계마다 제시되어야 하는 문서 또한 중요하며, 이 문서를 근거로 검증 및 평가가 진행되기 때문이다. 현재 안전기준에 제시되어 있는 사항은 법률적인 용어 및 표현이 쓰이고 있어서 이에 대해 곧바로 개발업체에서 적용하기에는 어려움이 있다. 따라서 본 연구에서는 소프트웨어 Lifecycle 각 단계마다 수행하여야 하는 업무에 대한 절차, 양식, 기법을 정리한 개발방법론을 제시하고자 하였으며, 이에 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 향상시키기 위해서는 제품관점에서 좋은 제품을 만들고, 정확한 시험으로 개발된 제품의 품질이 원하는 수준에 도달했는지를 판단하는 경우가 있으며, 이와는 다른 관점에서 좋은 제품은 좋은 조직 체계에서 만들어진다는 프로세스 관점이 있다. 또한 개발하려는 분야에 대한 지식 또한 중요하다. 다음은 각 관점에서의 관련 규격을 나타낸 그림이다.



<그림 1> 철도소프트웨어 안전기준 관련 표준의 범주

철도분야의 안전관련 표준으로는 전기전자 규격인 IEC 61508과 철도관련 규격인 IEC 62278, IEC 62279 규격을 대표적으로 들 수 있다. 이중 IEC 62279는 유럽전기전자 표준규격인 CENELEC의 EN 50128에서 국제규격으로 전환된 규격으로 철도분야 소프트웨어에 대해 다루고 있다. 프로세스와 관점에서는 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 개발 관련 프로세스에 대하여 성숙도 정도를 등급을 매겨 관리하고 있다. 제품 관점으로의 접근법으로는 소프트웨어 품질특성을 정의한 ISO/IEC 9126과 소프트웨어 제품의 품질특성 평가를 다루고 있는 ISO/IEC 14598이 있다.

전교부 사업인 철도종합안전기술개발사업 중 한국철도기술연구원 주관으로 2004년부터 2008년까지 수행하는 “철도소프트웨어 안전기준 체계구축” 과제에서 제시되는 안전기준은 상기 근간이 되는 국제 표준 외에 여러 국제 규격(IEC, ISO 등), 국내규격(KS 등) 및 산업체 표준(IEEE 표준 등)과 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.

다음은 제시한 안전기준 중 지침레벨의 일부 내용을 발췌한 것이다. 구성은 조항 및 해설, 근거기준으로 구성되어 있다. 아래 예시 조항으로는 구체적으로 수행하여야 하는 절차가 언급되어 있지 않아 많은 애로사항이 발생할 수 있으며, 본 연구에서는 Lifecycle 각 단계에서 수행하여야 하는 업무를 정한 개발방법론을 도출하였다.

제4조 철도 소프트웨어 설계 활동	
소프트웨어 개발자는 소프트웨어 생명주기에 따라 소프트웨어를 개발하기 위해 소프트웨어에 대한 설계 활동을 수행하여야 한다.	
【해설】 본 조항은 IEC 62279 및 IEEE Std 7-4.3.2, IEEE 1074의 내용을 참조하여 작성하였다. 소프트웨어 설계명세는 소프트웨어의 요구사항을 소프트웨어 구조, 소프트웨어 경포트트 및 인터페이스 및 구현단계에서 필요한 데이터와 관련된 사항으로 바꾸어 기술하는 명세이다.	
【근거기준】	
<ul style="list-style-type: none"> ● IEEE Std 62279 ● IEEE Std 7-4.3.2 IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generation Stations ● IEEE Std 1074 IEEE Standard for Developing Software Life Cycle Processes 	

<그림 2> 철도소프트웨어 안전기준의 구성 예

3. 철도소프트웨어 개발방법론

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 1970년대로 거슬러 올라간다.

구조적 방법론은 1970년대 이후 현재까지 가장 널리 사용되어오던 방법론으로 구조적 분석, 구조적 설계, 구조적 프로그래밍으로 이뤄져 있다. 단순한 업무처리 시스템의 개발에는 효과적이지만 대규모의 복잡한 시스템 개발에는 적합하지 않은 특징이 있다. 구조적 방법론을

지원하고 있는 기법은 대표적으로 DFD(Data Flow Diagram), DD(Data Dictionary), Mini-Spec. (Mini-Specification)이 있다.

두 번째로 대두된 방법론은 정보공학 방법론이다. 1980년대 후반 조직 전반에 걸쳐서 제시된 방법론으로 전사적인 차원에서 정보시스템 기획을 통해 단위시스템을 식별하고 이를 CASE 도구를 이용해 분석, 설계, 구현하는 방법론으로 기획 단계가 대폭 강화되고 CASE를 이용한 생산성 향상 등의 장점으로 인해 대규모의 시스템 개발에 적합하다. 그러나 복잡한 시스템의 모델링에 부적합해 현재는 객체지향 방법론으로 대체되고 있는 추세이다. 정보공학 방법을 지원하고 있는 기법은 대표적으로 ERD(Entity Relation Diagram), PHD(Process Hierachy Diagram), 프로세스 대 엔티티 Matrix 등이 있다.

최근에 제시된 방법론으로 객체지향 방법론이 있는데 1990년 이후 제기된 개발방법론으로 소프트웨어의 확장이나 변화를 용이하게 해줄 뿐만 아니라 기존 소프트웨어의 재사용성을 증가시켜 준다는 장점이 있다. 객체지향 시스템에서는 데이터를 다른 프로시저를 하나로 묶어 객체라는 개념을 사용해 실세계를 표현하고 모델링한다. 객체지향의 개념은 객체(object), 메시지(message), 상속(inheritance)의 세 가지 요소가 근간이 된다. 대표적인 객체지향 방법론으로 RUP(Rational Unified Process)가 있다.

이러한 개발방법론을 적용함으로써 얻게 되는 기대효과를 수요자, 공급자, 개발자 측면에서 살펴보면 다음과 같다.

수요자 측면:

- 균질한 산출물을 얻을 수 있다.
- 체계적인 품질보증을 받을 수 있다.
- 프로젝트 진행 중에 품질을 확인할 수 있다.
- 개발자들의 역할과 책임을 식별할 수 있다.
- 제시된 요구사항의 수용여부 확인이 용이해 진다.
- 산출물의 인도가 매끄러워 진다.
- 개발조직이 보유한 최고의 서비스를 기대할 수 있다.

공급자 측면:

- 경험의 재활용이 용이해 진다.
- 프로세스 기반의 개발을 할 수 있다.
- 표준을 쉽게 만들 수 있고, 적용이 용이하다.
- 체계적인 품질보증이 가능하다.
- 체계적이고 구체적인 계획을 수립할 수 있다.
- 개발자들의 책임과 역할이 분명해 진다.
- 불필요한 일을 최소화 할 수 있다.
- 관리의 번거로움을 덜 수 있고, 합리적인 관리가 가능하다.

개발자 측면:

- 합리적인 업무관행을 익힐 수 있다.
 - 프로세스에 기반한 개발활동이 가능해진다.
 - 계획에 따라 통제가 가능하여 개선의 여지가 있다.
- 결합에 대한 두려움을 줄일 수 있다.
 - 결합유발의 가장 큰 원인인 요구사항이 통제될 수 있다.
 - 시간과 노력을 줄일 수 있다.
- 이중작업 최소화
 - 업무의 중복과 누락으로 초래되는 혼란을 최소화 할 수 있다.

3.2 철도소프트웨어 개발방법론

Safety-related 소프트웨어 개발방법론은 철도분야에서 특히 강조되는 안전과 관련된 소프트웨어를 개발할 때 적용할 수 있는 접근방법을 제공하기 위해 개발되었다.

본 방법론은 안전과 관련된(safety-related) 철도분야의 소프트웨어를 개발하고자 할 때 활용할 수 있으며, 절차서, 양식서, 기법서의 세 부분으로 구성되어 있다.

절차서는 방법론을 구성하는 각 단계와 각 단계에 포함된 활동을 보여준다. 각 단계는 개발, V&V 및 안전과 관련된 활동으로 구성되어 있으며, 활동들은 주어진 입력을 받아들여 출력력을 생성하기 위한 과정이다.

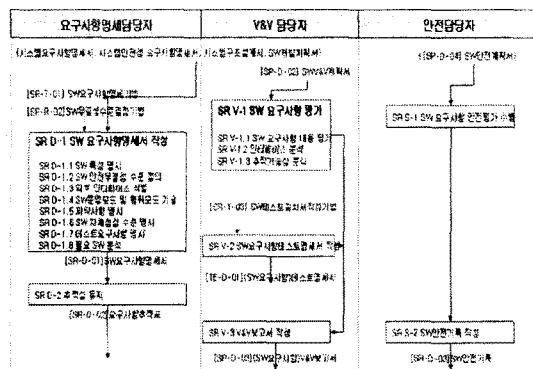
양식서는 절차서에서 정의된 입출력물에 대해 복차 및 그 구성 내용을 설명하고 있다. 이 양식을 활용함으로써 사용자는 보다 쉽게 방법론에서 원하는 입출력물을 활용할 수 있다.

기법서는 절차서의 활동 내역으로 설명하기에는 보다 기술적인 내용을 포함하고 있는 사항들을 모아둔 것이다. 이러한 기법들은 기술이 발전되면서 지속적으로 확대해 나갈 수 있다.

Safety-related 소프트웨어에 대한 개발방법론은 다음의 7단계로 구성된다.

- 1) 소프트웨어 계획 수립 단계
 - 2) 소프트웨어 요구사항 명세 단계
 - 3) 소프트웨어 설계 단계
 - 4) 소프트웨어 모듈 설계 단계
 - 5) 소프트웨어 구축 단계
 - 6) 소프트웨어 통합 단계
 - 7) 소프트웨어 하드웨어 통합 단계

이중 소프트웨어 요구사항 단계에 대한 절차서 예를 나타내면 다음과 같다. 절차서는 각 수행단계에 대한 활동 내역과 역할에 대하여 기술하고 있으며, 해당 단계의 입출력문서 및 수행내용이 기술되어 있다. 안전을 중요시하는 철도시스템의 특성상 안전담당자 영역을 따로 구분하여 강조하였다. V&V 담당자는 요구사항명세 담당자가 수행하는 업무에 대한 확인 및 검증 작업을 진행한다.



<그림 3> 철도소프트웨어 개발방법론 절차서(예)

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 알 수 있다. 철도소프트웨어의 경우 프로세스 성숙도 향상으로 관리 관점에서 소프트웨어의 품질을 확보하고자 하는 방법이 있으며, 정형기법에 의한 개발 및 검증이나, 적절히 도출한 Test Case에 따라 시험을 수행하여 소프트웨어 자체의 오류를 줄이고자 하는 제품관점의 접근법이 있다.

제기된 프로세스 관점 및 제품관점의 소프트웨어 품질 향상 방법을 강안하여, 철도소프트웨어에 대한 안전기준을 제시하였으며, 제시된 안전기준의 현장 적용성을 높이기 위하여 본 논문에서는 절차서, 양식서, 기법서로 구성된 안전관련 철도소프트웨어에 대한 개발방법론을 제시하였다. 특히 안전성을 중요시하는 철도시스템의 특성을 고려하여 안전담당자 영역을 활용하여 제안하였다.

[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