한국시뮬레이션학회 논문지
Vol. 24, No. 4, pp. 1-9 (2015. 12)
http://dx.doi.org/10.9709/JKSS.2015.24.4.001
ISSN 1225-5904

원전 안전-필수 소프트웨어의 품질향상을 위한 최적화된 확인 및 검증 방안

구서룡* · 유영제

An Optimized V&V Methodology to Improve Quality for Safety-Critical Software of Nuclear Power Plant

Seo-Ryong Koo* · Yeong-Jae Yoo

ABSTRACT

As the use of software is more wider in the safety-critical nuclear fields, so study to improve safety and quality of the software has been actively carried out for more than the past decade. In the nuclear power plant, nuclear man-machine interface systems (MMIS) performs the function of the brain and neural networks of human and consists of fully digitalized equipments. Therefore, errors in the software for nuclear MMIS may occur an abnormal operation of nuclear power plant, can result in economic loss due to the consequential trip of the nuclear power plant. Verification and validation (V&V) is a software-engineering discipline that helps to build quality into software, and the nuclear industry has been defined by laws and regulations to implement and adhere to a through verification and validation activities along the software lifecycle. V&V is a collection of analysis and testing activities across the full lifecycle and complements the efforts of other quality-engineering functions. This study propose a methodology based on V&V activities and related tool-chain to improve quality for software in the nuclear power plant. The optimized methodology consists of a document evaluation, requirement traceability, source code review, and software testing. The proposed methodology has been applied and approved to the real MMIS project for Shin-Hanul units 1&2.

Key words : Safety-Critical Software, Verification & Validation, Quality, Traceability, Testing

요 약

원자력 분야에서 안전관련(safety-related) 소프트웨어의 활용이 점차 확대됨에 따라서, 그에 상응하는 소프트웨어 안전과 신뢰도 향상을 위한 방안 연구가 지난 10여년 전부터 활발히 진행되고 있다. 원전 계측제어시스템(MMIS)은 원자력 발전소의 두뇌와 신경망에 해당하는 기능을 수행하고 있고 첨단 디지털 장비들로 구성된다. 따라서 원전 계측제어시스템의 소프트웨어 오류는 원자력 발전소 운전에 지장을 초래할 수 있고, 오동작으로 인한 발전소 정지로 경제적 손실을 초래할 수 있다. 소프트웨어 확인 및 검증(verification and validation, V&V)은 소프트웨어 품질을 향상시킬 수 있는 소프트웨어 공학의 분야로 알려져 있고, 원자력 산업계에서는 소프트웨어 생명주기에 따른 철저한 V&V 활동을 이행하고 준수할 것을 법규로 규정하고 있다. V&V 활동은 소프트웨어 전 생명주기에 따라 분석과 시험 활동들의 조합으로 다른 품질관련 공학 업무를 보완하는 역할을 한다. 본 논문에서는 명세 평가, 요건 추적, 소스코드 리뷰, 및 소프트웨어 시험을 통한 최적화된 안전관련 소프트웨어 V&V 방법론에 기반한 소프트웨어 품질 향상 방안과 단계별로 적합한 도구를 활용하여 효율성을 확보할 수 있는 방안을 제시하고자 한다. 제안된 방법론은 실제 신한울 1,2호기 원자력발전소 MMIS 시스템에 적용되어 입증되었다.

주요어 : 안전-필수 소프트웨어, 확인 및 검증, 품질, 추적성, 테스팅

Received: 23 November 2015, **Revised:** 3 December 2015, **Accepted:** 10 December 2015

***Corresponding Author:** Seo-Ryong Koo
E-mail: seoryong.koo@doosan.com
Doosan Heavy Industries & Construction Co., Ltd.
Nuclear I&C Test/V&V Team

1. 서 론

대표적인 안전관련 소프트웨어인 원자력발전소 계측제어(instrumentation and control) 시스템의 소프트웨어는 오류 발생 시 원자력발전소 운전의 심각한 피해를 초래할

수 있기 때문에 소프트웨어에 대한 안전성 및 신뢰성 등이 반드시 고려되어야 한다. 이와 유사한 시스템을 안전-필수(safety-critical) 시스템이라고 하며, 항공, 위성, 국방 제어시스템 등이 그 좋은 예가 된다[1, 2, 3]. 최근 원자력 발전소는 기존 아날로그 기반 기기들의 부품들이 노후화되고 단종 되어 더 이상 수급이 어려워짐에 따라서 디지털 기반 기기로의 업그레이드가 이루어지고 있다. 대표적으로 국내에서는 원전 계측제어시스템(man-machine interface systems, MMIS)에 대한 디지털 국산화 개발을 2001년부터 준비하여 현재 신한울1,2호기 발전소에 최초 국산화를 성공하여 납품이 진행되고 있다. 원전 MMIS는 약 2,000여 개의 프로세서 및 컴퓨터로 구성되어 각 계통들은 응용 소프트웨어 기반으로 기능이 구현되어 있고, 이런 응용 소프트웨어의 안전성 및 신뢰성 확보를 위해 철저한 V&V 절차가 이행되어야 한다.

일반적으로 소프트웨어의 개발 생명주기는 계획, 요구사항 분석, 설계, 구현, 테스팅 및 유지보수 등의 단계로 진행된다. 소프트웨어 개발의 품질 및 신뢰성 향상을 위한 확인 및 검증은 확인(verification)과 검증(validation)의 단계로 구성되는데, 소프트웨어 확인 작업은 소프트웨어 요구사항 및 설계의 적정성을 평가하는 과정으로 단계별 산출물 간의 일치성을 확인하는 과정이다. 그리고, 소프트웨어 검증 작업은 소프트웨어가 지정된 기능을 정확히 수행하는가 여부를 밝히는 과정으로 사용자의 요구사항이 충족되었음을 시험을 통해 객관적 증거를 확보하는 과정이다. 따라서, 안전-필수 시스템 기능 수행의 성패를 결정적으로 좌우하는 소프트웨어에 대한 철저한 확인 및 검증(V&V)은 매우 중요하다. 시스템의 작동을 결정하기 위한 논리 수행을 소프트웨어로 처리하는 위성 및 방사선 의료기기 등의 시스템은 점점 소프트웨어에 대한 의존성이 증가하고 있으며, 이러한 안전-필수 시스템의 경우 소프트웨어 오작동은 곧바로 치명적인 손실을 야기하게 된다. 하지만, 소프트웨어 생명주기에 따른 철저한 V&V 활동은 대부분 노동 집약적인 업무로써 사업 일정 및 비용에 업무의 범위가 조정되는 경우가 많다. 따라서, 원전에 적용 가능하고 인허가 및 품질의 영향을 최소화하기 위한 V&V 방법론에 대한 연구가 진행되고 있다[4, 5, 6].

본 연구에서는 대표적인 안전-필수 시스템인 원전 계측제어 시스템 소프트웨어의 안전성 및 신뢰성을 향상하기 위한 소프트웨어 개발 생명주기별 최적화된 V&V 방법을 제안하고, 각 단계별 방법을 지원하는 도구 체계를 함께 구현함으로써 검증에 대한 신뢰성 확보 및 업무의 효율성을 도모하고 이를 통한 국제적 품질 경쟁력을 갖춘 원전 소프트웨어를 만드는 기초가 되고자 한다. Fig. 1은 이와 같은 소프트웨어 수명주기별 소프트웨어 설계 활동과 소프트웨어 V&V 방법론을 도식화 한 것이다. 제안하고 있는 V&V 방법론은 원전 인허가 및 표준에서 제안하는 최소한의 V&V 업무를 문서 평가와 추적성 분석으로 정의하고, 3단계의 소프트웨어 시험을 포함하며 각 단계별 활동들은 효율적인 지원도구들의 활용을 통해서 최적화된 V&V 방법론이라고 판단한다.

본 논문의 구성은 다음과 같다. 2장에서는 최적화된 생명주기별 소프트웨어 확인 및 검증 방법론을 소개하고, 3장에서는 상세한 수명주기 단계별 명세평가, 요건 추적, 소스코드 리뷰, 소프트웨어 시험, 및 소프트웨어 결함관리에 대해서 설명한다. 마지막으로 4장에서는 결론 및 향후 연구 계획을 제시한다.

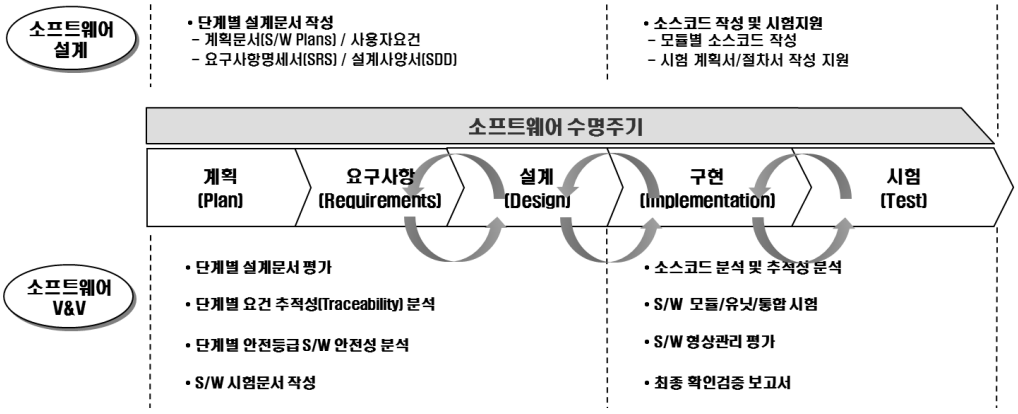

Fig. 1. Software Lifecycle

2. 소프트웨어 확인 및 검증 절차

소프트웨어 V&V 코드 요건인 IEEE-1012에 따르면 고유한 목적을 갖는 단계들로 구성된 수명주기에 따라 각 단계별 특징적인 설계 활동과 V&V 활동이 수행되어야 한다[7]. Fig. 1에서 보는 것과 같이 소프트웨어 수명주기는 각 업무별 특성에 따라서 크게 전반부/후반부의 두 부분으로 분류될 수 있는데, 전반부는 계획, 요구사항, 설계 단계까지로 문서 산출물 기반의 설계와 V&V 활동이 이루어지는 단계이고, 후반부는 구현, 시험 단계로 소프트웨어 자체 소스코드에 대한 분석 및 시험이 이루어지는 단계로 나눌 수 있다.

원전에 사용되는 계측제어 소프트웨어는 향후 인허가 적합성 제고를 위해 다양한 국제 표준들(NUREG, IEEE Standard 등)에 부합할 수 있도록 수명주기별로 철저하게 검증되어야 하고, 원전 소프트웨어의 특성에 맞추어 최적화된 V&V 방법론의 제안이 필요하다. 이에 본 연구에서는 수명주기 전반부 V&V 활동을 위해서 규제기관 및 국제 표준에서 제시하는 체크리스트(checklist)기반의 문서평가와 소프트웨어 문서에 대한 정방향 및 역방향 요건 추적 분석 방법론을 적용하였다. 후반부에서는 소프트웨어 명세 및 커버리지(coverage)에 근거한 3단계 Test를 통해 소프트웨어 품질향상을 도모할 수 있다. 이러한 V&V 활동들은 최적화된 CASE(computer aided software engineering) 도구들을 사용하여 인적 실수 저감을 통해서 검증 결과의 완전성, 정확성, 일관성을 향상시킬 수 있고, V&V 활동 수행의 자동화를 통해 사업의 공수 절감과 결과의 재사용성 향상을 도모할 수 있다.

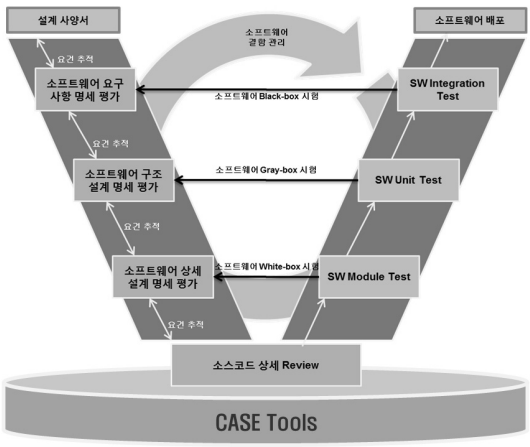

Fig. 2. Methodology for Software V&V

Fig. 2는 본 연구에서 제시하고 있는 원전 계측제어 소프트웨어를 위한 최적화된 V&V 방법론이다. 소프트웨어 공학 분야의 V-모델을 기반으로 수명주기 단계별 최적화된 활동을 제안하며, 수명주기 전반의 결함 또는 이슈관리와 CASE 도구의 지원을 통해 관리의 효율성을 높일 수 있을 것으로 기대한다. 소프트웨어 수명주기 전반부에서는 최상위 설계사양서를 기초로 소프트웨어 요구사항 명세 또는 설계명세서에 대한 평가와 추적성 분석을 수행할 수 있도록 방안을 제안하고, 소스코드 생성 후 IEEE-1012에 따른 점진적인 3단계 소프트웨어 시험을 모듈시험, 유닛시험, 소프트웨어 통합시험으로 진행하는 방안을 제안한다.

3. 수명주기 단계별 상세 확인 및 검증

본 장에서는 2장에서 제시한 최적화된 소프트웨어 확인 및 검증 방법론에 대한 상세한 내용을 기술한다. 3.1절에서는 소프트웨어 명세 평가 및 요건 추적 분석 방법을 제안하고, 3.2절에서는 소스코드 리뷰와 소프트웨어 시험에 대해서 제안하며, 3.3절에서는 소프트웨어 수명주기 단계 전반에 걸친 결함관리 절차를 제시한다. 각 절에서는 제안하는 방법론을 지원하기 위한 효율적인 CASE 도구를 제시하고자 한다.

3.1 소프트웨어 명세 평가 및 요건 추적

2장에서 제시한 Fig. 2의 전반부에 해당하는 대표적인 V&V 활동으로 체크리스트 기반의 소프트웨어 명세 평가와 양방향 요건 추적성 분석을 제안한다. 설계 부서에서 작성하는 소프트웨어 개발 산출물인 소프트웨어 요건명세서(software requirement specification, SRS)와 소프트웨어 설계명세서(software design specification, SDD)에 대한 문서평가를 위해서 NUREG-0800을 기준으로 원자력 규제기관의 규제 입장을 기준으로 평가할 수 있는 체크리스트를 수립하였고, 국제 표준에서 요구하는 소프트웨어 명세의 상세 특성을 IEEE-1012 기준으로 수립하여 제안한다.

Fig. 3은 SRS 및 SDD 문서 평가/검증을 위한 각 체크리스트별 항목을 나타낸다. SRS/SDD 등의 문서기반의 평가를 위한 NUREG-0800 인허가 적합성 검토 측면에서는 문서에서 기술하는 기능(function) 특성 및 공정(process) 특성에 대한 평가를 수행할 수 있는 상세 체크리스트를 수립하였다. 그리고 문서의 상세 검증을 위해서 IEEE-1012 기준의 추적성, 명세 평가, 연계(interface) 분석의 특성을

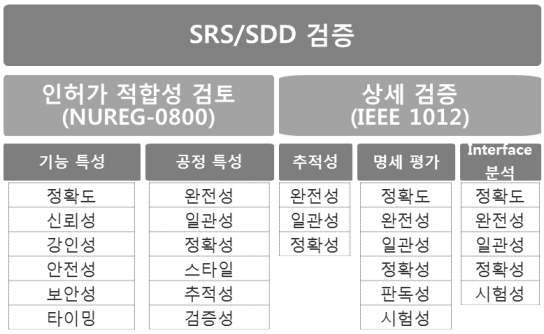

Fig. 3. Checklists for SRS/SDD

평가할 수 있는 상세 체크리스트를 수립하여 제안하고 있다. 소프트웨어 수명주기 전반의 문서기반 평가를 위해서 Fig. 3으로 구성된 상세 체크리스트를 활용하여 검증 특성별 검증항목을 만족하는지 문서 평가를 단계별로 진행하고 오류사항에 대해서는 설계팀으로 피드백을 하면서 문서의 품질을 향상 시킬 수 있다.

수명주기 전반부의 또 다른 V&V 주요 활동은 정방향/역방향 추적성 분석이다. 추적성 분석은 최상위 설계요건이 수명주기에 따라 하위 세부 단계별로 적절하게 반영이 되어 있는지 여부를 확인할 수 있는 활동으로 정의할 수 있다. 정방향 추적성 분석은 수명주기 단계 간 상위 단계의 요건 및 산출물이 하위 단계의 요건 및 산출물로 모두 구체화 되어 반영되었는지 확인하는 방안으로 정의하며, 역방향 추적성 분석은 모든 하위 요건들이 상위 요건을 근거로 도출되었는지 확인하는 방안으로 정의한다. 수명주기 단계별로 정방향 및 역방향 분석을 모두 수행함으로써 단계별 요건의 누락을 방지하고 상/하위 요건 간의 추적성 정보를 확인할 수 있다. 이와 같은 추적성 분석 매우 노동 집약적인 업무가 대부분이므로 CASE 도구인 IBM의 DOORS 활용을 통해서 인적 오류를 저감하고 업무의 효율성을 높일 수 있다.

Fig. 4는 DOORS를 활용한 소프트웨어 요건 추적성 분석을 수행하는 예를 보여 준다. 요건단계 문서를 기준으로 하위 설계단계 문서와 시험 사례까지의 요건 추적분석 결과를 DOORS를 통해서 연결하고 단계별 V&V 보고서 작성시 활용할 수 있다.

3.2 소스코드 리뷰 및 소프트웨어 시험

2장에서 제안한 최적화된 소프트웨어 V&V 방법론의 후반부는 전반부의 문서 검증이 완료된 설계 사항에 대해 설계 부서에서 작성하는 소스코드 기반의 코드 리뷰와 모듈시험, 유닛시험, 소프트웨어 통합시험의 점진적인 3단계 소프트웨어 시험을 통한 검증으로 구성된다. 소스코드 리뷰는 NUREG/CR-6463 표준에 기반한 체크리스트를

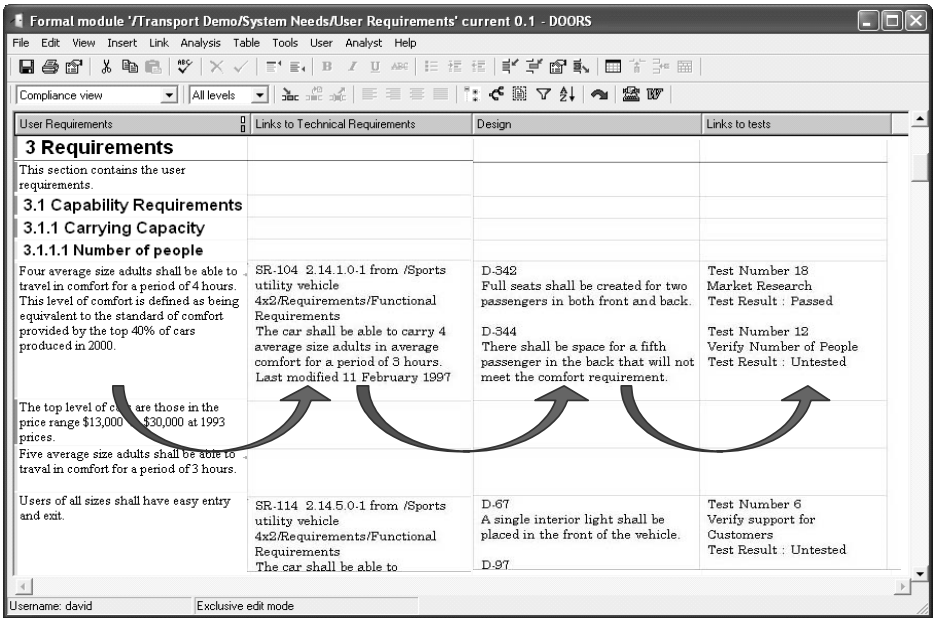

Fig. 4. Traceability Analysis using DOORS

Table 1. Checklist for Source Code Review

Quality Characteristics	
5.1 Reliability	5.1.1 Predictability of Memory Utilization
	5.1.2 Predictability of Control Flow
	5.1.3 Predictability of Timing
5.2 Robustness	5.2.1 Transparency of Functional Diversity
	5.2.2 Exception Handling
	5.2.3 Error Containment
5.3 Traceability	5.3.1 Use of Built-in Functions
	5.3.2 Use of Compiled Libraries
5.4 Maintainability	5.4.1 Readability
	5.4.2 Data Abstraction
	5.4.3 Functional Cohesiveness
	5.4.4 Malleability
	5.4.5 Portability
5.5 Security	

수립하여 소프트웨어의 품질 특성을 정량적 및 정성적으로 평가하고 코딩을 위한 권고사항을 설계부서에 제시한다. 즉, 소스코드가 만족해야 할 품질 특성을 평가하고, 소프트웨어 오작동을 유발할 수 있는 잠재적 오류를 검토하여 수정하도록 설계 부서에 검증의 결과로 피드백을 한다. 또한, 소프트웨어 코딩에 대한 전문 지식을 가진 설계자들이 부족한 한계로 인해, 시스템 고장을 일으킬 수 있는 잠재적 오류가 포함될 수 있는데, 소스코드 리뷰 과정을 통해 이와 같은 잠재적 오류를 발견하고 정확한 코딩 가이드라인을 제시한다. 본 연구에서는 수명주기 전반부 문서검증을 위한 체크리스트와 유사하게 Table 1과 같이 NUREG/CR-6463 기반의 소스코드 리뷰 체크리스트를 수립하여 활용하는 것을 제안한다.

이와 같은 소스코드 리뷰 및 분석의 효율성을 위해서는 국내 테스팅 상용 소프트웨어 CASE 도구인 ㈜슈어소프트의 CodeScroll을 활용할 것을 제안한다. Fig. 5와 같이 CodeScroll은 기본적인 소스코드 품질을 위한 가이드라인 및 코딩 규칙을 제공하여 자동으로 검사할 수 있도록 지원하며, Table 1과 같은 원자력 사업에 특화된 코딩 규칙을 사용자 규정 규칙으로 수립하여 적용할 수 있도록 소프트웨어 V&V 업무를 지원할 수 있다. 효율적인 CASE 도구의 지원을 통해서 수만 라인에 이르는 소스코드에 대한 평가 및 검토 업무를 자동화할 수 있을 것으로 기대한다.

소프트웨어 수명주기 단계 중 소프트웨어 V&V의 마지막 단계는 소프트웨어에 대한 철저한 검증시험이다. 본 연구에서는 표준에 따른 3단계 검증시험을 Fig. 6과 같이 제시하고자 한다. 소프트웨어 검증시험의 충실도를 향상시키기 위해 소프트웨어 최소 단위로부터 최종 시스템 단위까지 시험 범위를 중첩시켜 테스팅을 수행하는 점진적인 검증시험 방안을 수립하였다. 첫 번째 모듈시험(Module Test)에서는 소프트웨어 최소 단위가 상세 설계와 정확하게 부합하고 구현되었는지 검증하고, 두 번째 유닛시험(Unit Test)에서는 소프트웨어가 구조 설계 및 요구사항에 부합하는지 검증을 수행한다. 마지막으로 소프트웨어 통합시험(Software Integration Test)은 작성된 응용 소프트웨어가 최초 사용자의 시스템 측면의 요구사항을 만족하는지 검증을 하는 단계이다. Fig. 6에 표현한 모듈시험은 시스템 소프트웨어의 최소 단위인 기능 모듈이 해당하고, 각 모듈에 대해서 소스코드 내 존재하는 모

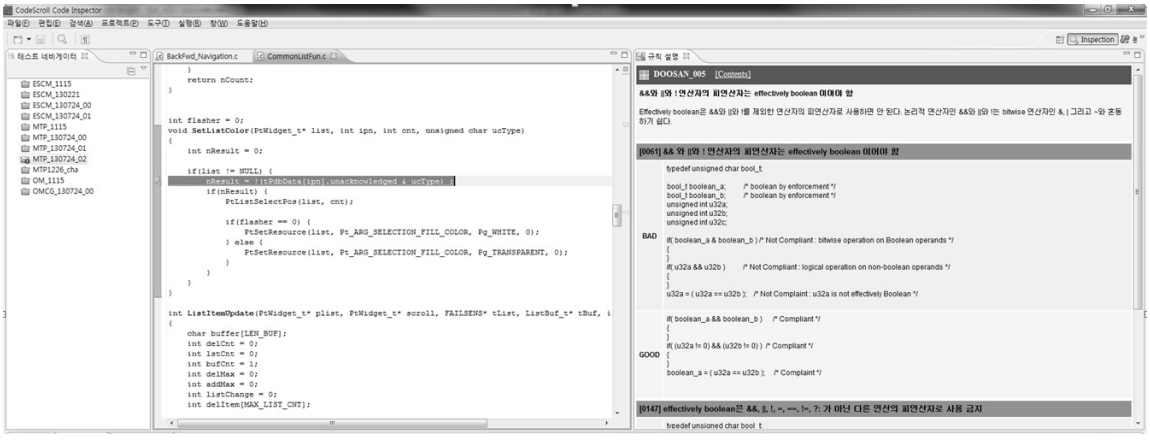

Fig. 5. Source Code Review using CodeScroll

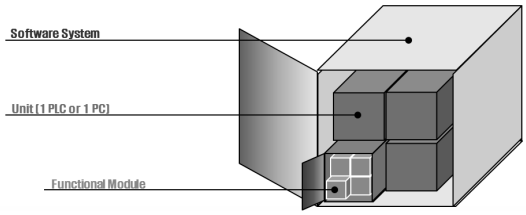

Fig. 6. Scope of Software Validation Test

든 분기를 커버하는지 확인하는 white-box 테스팅을 수행하는 단계이다. 다음 단계인 유닛시험은 시스템 소프트웨어 내의 서브 시스템을 이루는 제어기 또는 컴퓨터 수준의 단위로 정의하며, 여러 개의 기능 모듈들이 통합된 형태의 소프트웨어를 대상으로 요건 및 기능 커버리지를 만족하는지 확인하는 gray-box 테스팅의 형태로 수행된다. 마지막 단계인 소프트웨어 통합시험은 최종 시스템 수준의 소프트웨어에 대한 기능요건을 정확하게 만족하게 구현되었는지 확인하는 black-box 테스팅을 수행하는 단계로써, 최초 정의한 소프트웨어 요구사항 명세서 상의 모든 요건들이 시험사례로 입력되어 만족하는지 검증을 수행한다.

Fig. 6과 같이 각 단계별 검증시험들은 시험 범위를 중첩시켜 신뢰도 향상을 도모하고, 각 단계별로 충족해야 하는 커버리지 요건을 만족할 때까지 검증시험을 수행함으로써 소프트웨어의 잠재적인 결함을 최대한 발견하여 품질 높은 원전 소프트웨어를 개발할 수 있도록 한다. 소프트웨어 검증시험에서 역시 인적 오류를 방지하고 효율성을 높이기 위해 시험 설계 및 실행 CASE 도구인 ㈜슈어소프트의 CodeScroll을 적용하는 것을 제안한다. 기존의 임베디드 소프트웨어 시험 수행 시 시험 사례들을 수작업으로 입력하며 결과를 받는 방식이었으나, 도구를 활용하여 시험사례 세트를 일괄적으로 입력하여 시험 수행을 자동화 하는 방식으로 많은 수행 시간 절감을 도모할 수 있다. 그리고 Fig. 7에서 보는 것처럼 시험 수행 결과 분석 시 필요한 요건 및 분기 커버리지 측정 및 분석을 도구를 활용하면 매우 수월하게 결과를 얻을 수 있는 장점이 있다. 각 요건 및 분기 커버리지를 만족하지 못하면 그에 상응하는 시험사례를 제안해 주는 기능도 제공되어 검증자가 기준이 되는 커버리지 만족을 위한 분석을 용이하게 한다.

이렇듯 소프트웨어 검증시험에서 CASE 도구를 활용하면 반복적인 소프트웨어 검증시험 수행을 일부분 자동화할 수 있고, 각 단계별로 충족해야 하는 커버리지 요건에 대한 분석을 자동으로 실행시켜 주어 검증자들을 지원

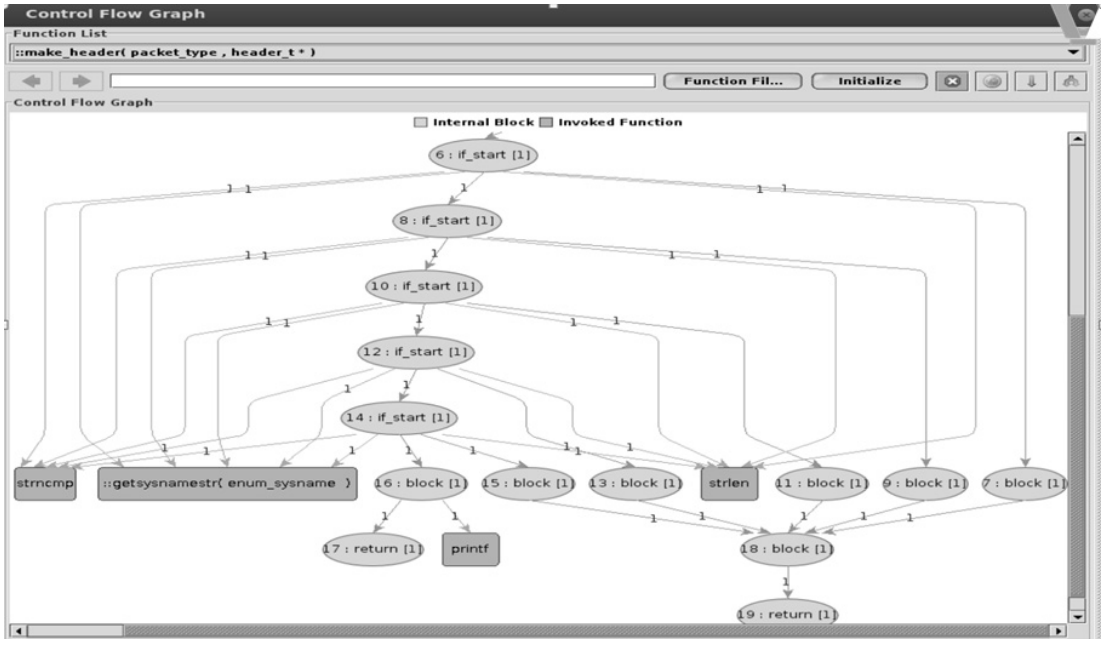

Fig. 7. Coverage Analysis using CodeScroll

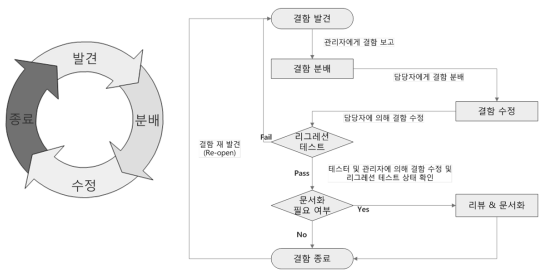

Fig. 8. Issue Management Process

해 줄 수 있다. 또한, 시험 수행 중 사용한 시험사례와 시험 수행 결과들이 모두 데이터베이스로 저장되어 향후 회귀분석 및 시험이나 타 프로젝트에서의 재사용성에 매우 유용하게 활용될 수 있는 장점이 있다.

3.3 소프트웨어 결함 관리

소프트웨어공학 측면에서 모든 소프트웨어에 대한 결함 관리는 매우 중요한 항목이다. 특히, 안전-필수 시스템 소프트웨어는 개발 초기부터 유지보수 단계까지 그 결함에 대한 이력(history) 확보 및 검증이 핵심적으로 필요하고 수명주기별 V&V 활동의 품질유지 및 향상을 위해서는 정확한 결함 이력의 유지/관리 및 향후 경험지식(Lessons Learned)로 활용하는 것이 소프트웨어 재사용성 향상의 측면에서도 매우 중요하다. 따라서 본 연구에서는 소프트웨어 개발과 확인 및 검증 참여자 간의 명확한 의사소통 및 결함 이력 관리를 위해서 Fig. 8과 같은 결함관리 절차를 구축하여 제안한다.

소프트웨어 결함관리 절차는 소프트웨어 수명주기 전 단계에 걸쳐서 문서평가, 추적성 분석, 소스코드 리뷰 및 검증시험에서 발생하는 이슈 및 결함 사항들을 발견하고, 각 항목별 담당자에게 분배하여 수정을 할 수 있도록 조치하며, 최종적으로 수정된 이슈 및 결함에 대해서 확인하여 종료하는 단계를 갖는다. 이와 관련한 상세한 수행 절차는 Fig. 8의 순서도에 따른다. V&V 수행 중 발견된 결함들에 대해서는 회귀시험 (regression test)을 통해서 결함이 정확하게 수정되어 확인될 때까지 시험을 반복하여 수행하며, 최종적으로 결함이 재발견 되지 않으면 해당 회귀시험을 종료하고 문서화하여 결함에 대한 이력 및 결과를 기록한다. 이러한 결함 기록은 이후 수명주기 단계에서 새롭게 발생하는 결함들에 대한 원인 분석 및 시험결과의 재사용성 측면에서 매우 중요하게 활용될 수 있다.

생명주기별 모든 소프트웨어 공정에서 그렇듯, 소프트웨어 결함 관리 역시 CASE 도구의 지원이 중요하다. 본 연구에서는 체계적인 소프트웨어 결함 관리를 위해서 Fig. 9와 같이 산업계의 무료 소프트웨어인 Mantis Bug Tracker를 CASE 도구로 활용하도록 제안한다. 도구 사용을 통해

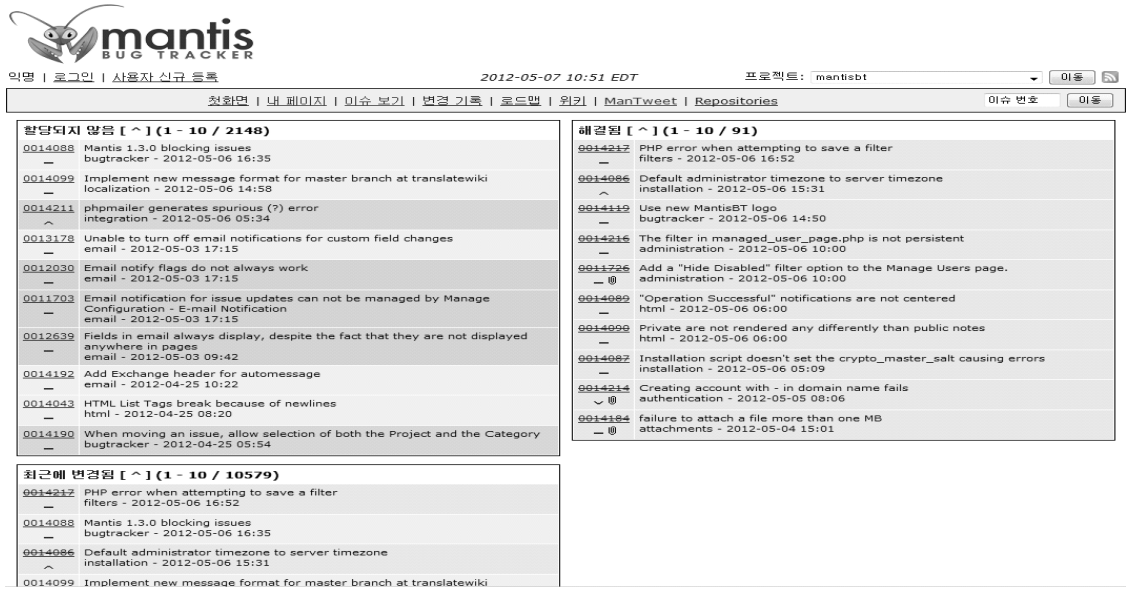

Fig. 9. CASE Tool for Issue Management

서 결함 및 이슈에 대한 이력(history) 관리가 수월해 지며 소프트웨어 개발 / 확인 및 검증자간의 명확한 의사소통을 가능하게 해 줄 수 있다.

4. 결론 및 향후 연구

본 논문에서는 원전 안전-필수 소프트웨어의 전반적인 품질을 향상시키고 업무의 효율성을 증대시키기 위한 방안으로써 최적화된 소프트웨어 확인 및 검증(V&V) 방법론에 기반 한 원전 안전관련 소프트웨어 안전성 및 신뢰도 향상 방안을 제시하고, 수명주기별로 상세한 검증 방법론 및 활용 CASE 도구를 제안하였다. 원전에서 요구하는 산업계 표준과 인허가 기준을 만족하기 위해서 수명주기 전반부에서는 체크리스트 기반의 문서 평가 방법과 양방향 추적성 분석 방법론을 제안하였고, 수명주기 후반부에서는 체크리스트 기반 소스코드 검토 및 점진적 3단계 소프트웨어 검증시험 방법론을 제안하였다. 마지막으로 소프트웨어 생명주기 전 단계에 걸쳐 단계별 방법들을 지원할 수 있는 도구 세트를 마련하여 제안하고, 업무 중 발생하는 결함관리를 위한 절차 및 도구까지 제안하였다. 이와 같이 본 연구에서 제안된 하나의 최적화된 통합 확인 및 검증 프레임은 향후 원전 뿐만 아닌 다른 안전-필수 소프트웨어 응용 생명주기 관리(application lifecycle management, ALM) 프레임워크를 구축하는데 기반이 될 것으로 전망한다.

본 연구에서 제안된 V&V 방법론은 신한울 1,2호기 원자력발전소 MMIS 최초 국산화 사업에 적용되었고, 성공적으로 1호기 MMIS 소프트웨어 개발을 수명주기 단계별로 최종 완료하여 납품함으로써 그 방법론의 효용성을 입증할 수 있었다. 현재는 동일한 2호기 MMIS 소프트웨어에 대한 검증시험을 진행 중에 있으며, 향후 건설 계획 중인 신고리 5,6호기 및 신한울 3,4호기 원자력발전소 사업에서도 활용될 계획이다.

향후 사업 적용 및 연구를 통해서 품질향상 및 소프트웨어 신뢰도 향상을 위한 시뮬레이션 기반의 방안을 모색해 볼 계획이며, ALM 측면에서의 CASE 도구들의 연계를 통합하여 좀 더 효율적이고 검증자에게 용이할 수 있는 도구사슬(tool-chain)에 대한 연구를 계획 중이다.

References

1. N.G. Leveson (1995), “Safeware – System Safety and Computers,” Addison-Wesley.
2. Storey N (1996), “Safety-Critical Computer Systems,” Addison-Wesley.
3. Dyer M (1992), “The Cleanroom Approach to Quality Software Development,” John Wiley & Sons.
4. S. Koo, P. Seong, J. Yoo, S. Cha, and Y. Yoo (2005), “An Effective Technique for the Software Requirements Analysis of NPP Safety-Critical Systems, Based on Software Inspection, Requirement Traceability, and Formal Specification”, *Reliability Engineering and System Safety*, Vol. 89, No. 3, pp. 248-260.
5. S. Koo, P. Seong (2006), “Software Design Specification and Analysis Technique(SDSAT) for the Development of Safety-Critical Systems Based on a Programmable Logic Controller(PLC)”, *Reliability Engineering and System Safety*, Vol. 91, Issue 6, pp. 648-664.
6. S. Koo, P. Seong, J Yoo, S Cha, C Youn, H Han (2006), “NuSEE: an integrated environment of software specification and V&V for NPP safety-critical systems,” *Nuclear Engineering and Technology*, Vol. 38, No. 3, pp. 259-276.
7. IEEE (1998), IEEE Standard 1012 for software verification and validation, an American National Standard.

구 서 룡 (seoryong.koo@doosan.com)

1998 한국과학기술원(KAIST) 원자력공학과 학사
2000 한국과학기술원(KAIST) 원자력 및 양자공학과 석사
2005 한국과학기술원(KAIST) 원자력 및 양자공학과 박사
2006～현재 두산중공업(주) 원자력I&C BU 재직 중

관심분야 : SW엔지니어링, SW 확인 및 검증(V&V), 애플리케이션 생명주기 관리(ALM)

유 영 제 (yeongjae.yoo@doosan.com)

2000 한국과학기술원(KAIST) 원자력공학과 학사
2003 한국과학기술원(KAIST) 원자력 및 양자공학과 석사
2007 비엔에프테크놀로지(주)
2007～현재 두산중공업(주) 원자력I&C BU 재직 중

관심분야 : SW엔지니어링, SW 확인 및 검증(V&V), CMMI, 애플리케이션 생명주기 관리(ALM)