

# 무기체계 소프트웨어 신뢰성 향상을 위한 프로세스 : 사례연구

## Process for Software Reliability Improvement of Weapon System : Empirical Study

\*장주수<sup>1</sup>, #강명택<sup>2</sup>, 백종민<sup>3</sup> 신주환<sup>4</sup>

\*J. S. Jang<sup>1</sup>, #M. M. Kang(mmkang@kaist.ac.kr)<sup>2</sup>, J. M. Baik(jmbaik@kaist.ac.kr)<sup>3</sup>, J. H. Sin<sup>4</sup>

<sup>1</sup> 모아소프트, <sup>2</sup>한국과학기술원, <sup>3</sup>한국과학기술원 <sup>4</sup>국방과학연구소

### 1. 서론

최근 들어 컴퓨터 시스템이 다양한 기능들을 탑재하고 환경 적응적인 지능적이고 자동화된 시스템으로 진화됨에 따라 시스템에서의 소프트웨어 비중이 점차 커지고 있다. 무기체계 시스템의 경우에도 과거에는 시스템에서의 하드웨어 비중이 소프트웨어에 비해 컸으나 점차적으로 소프트웨어의 비중이 증대되고 많은 하드웨어를 제어함으로써 소프트웨어에 대한 중요성이 점차 증대되고 있다. 소프트웨어 신뢰성은 주어진 조건 및 환경에서 소프트웨어가 주어진 시간 동안 고장 없이 동작할 확률을 뜻하는 것으로 소프트웨어의 신뢰성을 향상시키는 것이 무기체계의 품질을 높이는 데 있어 가장 중요한 요소가 된다. 이를 위해 소프트웨어 개발 프로세스에 따라 소프트웨어의 신뢰성 측정을 위한 데이터를 지속적으로 수집하는 활동을 수행하여야 할 필요성이 있으며 이러한 데이터를 기반으로 개발 초기 단계에서부터 소프트웨어의 신뢰성을 측정하고 분석하는 활동을 수행하여야만 최종적으로 높은 신뢰성을 가지는 소프트웨어를 개발할 수 있게 된다.

신뢰성 측정을 위해 선행되어야 하는 중요한 활동이 데이터 수집 활동이다. 데이터 수집 활동이 제대로 수행되지 못한다면 현재 개발 중인 소프트웨어의 신뢰성을 측정할 수 없으며 높은 신뢰성의 소프트웨어를 개발 하는데 있어서 큰 문제를 초래할 수 있다. 그렇기 때문에, 개발 프로세스에 따라서 각 단계에서 수집 가능한 척도에 따라 소프트웨어가 가지는 신뢰성 척도들을 수집하는 활동을 지속적으로 수행하고 관리함으로써 현재 개발 중인 소프트웨어의 신뢰성이 측정되고 분석되어 소프트웨어의 신뢰성을 향상시키는데 큰 도움이 된다.

본 논문에서는 현재 개발 진행 중인 무기체계 소프트웨어의 신뢰성을 향상시키는 것이 주목적이며 이를 만족시키기 위해 각 단계를 세분화하여 진행하였다. 특히, 초기단계에서는 소프트웨어 신뢰성 측정에 필요한 척도 정의 및 기존 시스템에 대한 신뢰성 분석을 수행하였으며 이를 기반으로 하여 현재 개발 중인 무기체계 소프트웨어의 신뢰성을 향상시키는데 중점을 두고 진행되었다. 초기 소프트웨어 개발 프로세스 단계에서는 소프트웨어의 신뢰성을 예측하기 위해 RL-TR-92-52 [1] 모델을 제시하였으며 테스트 단계 이후에서의 신뢰성 추정을 위해서는 Schneidewind [2] 모델을 제시하였다. 이러한 모델을 통해서 소프트웨어의 초기 신뢰성을 예측하고 앞으로의 신뢰성을 추정함으로써 보다 높은 신뢰성을 가지기 위한 노력을 수행할 수 있을 뿐만 아니라 개발 완료 시점을 결정할 수 있게 된다.

### 2. 소프트웨어 신뢰성에 대한 통찰

소프트웨어 신뢰성에 대한 관심은 1970년대 중반부터 시작되었다 [3]. 대부분의 소프트웨어 신뢰성 측정을 위한 모델들이 개발되기 시작되었으며 현재까지 많은 모델들이 등장하고 소프트웨어 개선을 위한 많은 활동들이 여러 산하 기관에서 수행되고 있다.

미 국방성은 신뢰성 연구 센터 (RAC; Reliability Analysis Center)를 운영하여 국방 시스템뿐만 아니라 일반 산업 시스템에서의 신뢰성 향상을 위한 많은 연구를 진행하고 있다. RAC는 제품에 대한 신뢰성과 품질 고려에 있어서 관련 있는 데이터들을 수집하고 분석하며 유용한 정보들을 배포하고 있다. RAC는 또한

정보의 평가 및 배포뿐만 아니라 훈련 및 교육 그리고 컨설팅을 제공하고 있다.

### 3. 신뢰성 향상을 위한 프로세스

1단계에서는 단계별 신뢰도 평가 척도 정의에 대한 연구를 수행하였다. 소프트웨어 척도는 소프트웨어 측정 기술을 기반으로 소프트웨어 생명 주기 동안에 소프트웨어의 특징 또는 특성을 객관적이고 과학적인 수치로 정량화할 수 있도록 하는 기술이다. 소프트웨어 신뢰도를 예측하고, 추정하기 위해 신뢰성과 관련된 데이터의 수집은 중요하기 때문에 소프트웨어 신뢰성에 영향을 미치는 단계별 척도는 반드시 측정되어야 한다. 다음 표 1은 소프트웨어 개발 과정에서 신뢰도에 영향을 미치는 척도들을 고려하여 12개의 항목으로 분류하였다.

표 4 소프트웨어 개발과정에서의 척도

	척도
규모	- 요구사항관련 측정 (요구사항의 수, FP 등) - 설계단계 및 구현단계 관련 측정
결함	- 검사 대상 및 단계, Staff - 유형, severity, 수, 원인 - 삽입, 발견, 제거시간 (단계/시간 측정)
실패	- 실패 대상, Staff - 유형, severity, 수, 원인 - 발생시간, 복구시간, 삽입/제거시간 (시간측정) - 프로젝트 활동들 소요시간 (단계별/ 대상별)
시간	- Review / Inspection 시간 (단계/ 시간 측정) - Test 시간 (단계/ 시간 측정), 릴리즈 시간
노력	- 프로젝트 관련자의 유형구분, 도메인 경험, 투입 수, 및 각 활동에 대한 인력 수 - 각 프로젝트 활동 지원하는 도구 및 자원의 내역
검사	- 수행단계, 대상, 설명 - 결점 유형, severity, 수, 원인 - 결점 삽입/발견/제거 단계 및 시간 - Inspection 준비/미팅/수행/제작업 시간, 횟수, 인력
복잡도	- Cyclomatic, Halsted's 기준 측정 - Information Flow 측정 - 로직의 유형 및 수 - LDRA를 이용한 복잡도 Report이용
완결성	- 상위단계의 내용이 하위단계에 반영 여부를 측정
Test Coverage	- 단위, 통합, 시스템 Coverage - LDRA 이용한 Report이용
성숙도 & 환경	- CMMI level 및 프로젝트 팀 구성 - 개발환경(개발언어, 개발툴) 및 테스트 환경 - 문서표준, 코딩표준 등 표준화 관련
추적성 & 형상관리 & 일정관리	- 요구사항 추적성 (툴 지원여부) - 형상관리 여부 - 전체프로젝트 일정 및 계획대비 진척사항관리
성능	- 시스템 이용률, 처리량, 응답시간, 대기시간, 서비스 시간, CPU 사용율, 메모리 사용율, 서버 busy 시간,

2단계에서는 현재 개발 진행 중인 소프트웨어의 신뢰성 확보를 위하여 기존 시스템에 대한 신뢰도 분석 및 평가를 수행하여 향후 신뢰도 향상 방안에 대한 제안을 하였다.

3단계에서는 대상시스템의 소프트웨어 신뢰성 보증을 위해 소프트웨어 개발 프로세스의 특성을 반영하고 소프트웨어 신뢰성의 예측 및 추정을 통해 대상시스템의 신뢰성 평가를 수행하기 위한 평가 프로세스를 개발 하였다. 소프트웨어 신뢰성 평가는 개발 프로세스의 활동들과 연계되어 수행될 필요성이 있으며 지속적으로 개선사항을 수집하고 이를 현재의 프로세스에 반영함으로써 조직에 적합한 형태로 테일러링 되어야 한다.

4단계에서는 소프트웨어 신뢰도 모델 정의에 대한 내용으로 소프트웨어 신뢰성 예측 모델과 추정 모델로 분류하고 소프트웨어 신뢰도 평가 프로세스에서의 적용 범위와 활용 영역을 분석하였다. 소프트웨어 신뢰도 평가 방법에는 크게 초기 신뢰도 예측과 사후 신뢰도 추정 방법으로 나누어 생각할 수 있다. 초기 신뢰도 예측 모델은 과거 경험적 자료를 입력으로 사용하며 코드가 완료된 시점 또는 시험 단계 이전에 사용되어지며 소프트웨어의 시험 방향을 결정하는 용도로 사용된다. 사후 신뢰도 추정 모델은 주로 시험단계에서 발생하는 데이터를 입력으로 시험이 종료된 시점에 사용될 수 있으며 시험을 통한 신뢰도 확보 평가를 위한 용도로 사용된다.

5단계에서는 4단계에서 제시한 RL-TR-92-52 모델과 Schneidewind 모델을 적용하기 위한 소프트웨어를 개발하여 현재 개발 중인 소프트웨어의 신뢰성을 측정하고 평가할 수 있도록 하였다. 즉, 모델을 적용하기 위해서는 각 모델에 맞는 데이터를 수집하고 계산하여 신뢰성을 측정해야 하기에 이를 자동화 시킬 필요성이 있어 신뢰성 측정 분석 도구를 개발하였다. 현재 CASRE와 SMERFS [9] 같은 많은 소프트웨어 신뢰성 측정 도구들이 존재하지만 이 도구를 사용하는데 있어 불편함과 단점들이 존재하여 이를 개선하고 사용자 편의성을 제공하기 위해 새로운 신뢰성 측정 분석 도구를 개발하였다. 이 도구의 도움으로 초기 신뢰성 예측과 사후 신뢰성 추정 및 분석 활동이 보다 쉽게 수행되었다.

6단계에서는 현재 개발 진행 중인 시스템의 신뢰성 분석을 수행하였다. 신뢰성 분석을 수행하기 이전에 신뢰성 분석에 필요한 데이터를 수집하였으며 수집된 데이터를 정리하여 초기 신뢰성을 예측하였다. 개발 진행 중인 시스템의 개발이 완료되지 못한 시점이기 때문에 사후 신뢰성 추정 모델을 적용하지 못하였으며 요구사항 명세서 및 설계 명세서를 통한 초기 신뢰성 예측만을 수행하였다. 신뢰성 측정 결과 기존 시스템 A 보다 현재 개발 중인 시스템 B의 결함 밀도가 낮음을 확인 할 수 있었다.

**4. 결론**

무기체계 시스템은 인간의 생명과 매우 밀접한 시스템이며 많은 하드웨어들은 소프트웨어에 의해 제어되고 있다. 이러한 소프트웨어는 동작함에 있어서 높은 신뢰성을 확보하는 것이 중요시 되고 있다. 높은 신뢰성의 소프트웨어를 개발하기 위해서는 개발 프로세스 상에서 지속적으로 신뢰성이 측정되고 분석되어 개선사항이 반영될 필요가 있으며 이를 위해 6단계에 걸친 신뢰성 향상 프로세스들이 수행되었고 수행된 프로세스에 의해 보다 높은 신뢰성을 확보할 수 있었으며 보증 활동 또한 수행이 되었다. 본 논문에서 제시하는 단계별 프로세스들이 현재 개발 프로젝트에 맞게 테일러링하여 적용된다면 개발 조직에서 목표로 하는 신뢰성에 도달하는데 많은 도움이 될 것이다.

**참고문헌**

[1] James A. McCall, William Randell, Janet Dunham, Linda Lauterbach, RL-TR-92-52, Final Technical Report, Rome Laboratory, April 1992.  
 [2] N.F. Schneidewind, Reliability Modeling for Safety-Critical

Software, IEEE Transactions on Reliability, Vol. 46, No. 1, March 1997  
 [3] RAC, Introduction to Software Reliability: a State of the Art Review, Reliability Analysis Center, pp. 1-10, 1996  
 [4] Alan B. Salisbury, Software 2015: A National Software Strategy to Ensure U.S. Security and Competitiveness, Report of the 2nd National Software Summit, April 29, 2005  
 [5] 원상호. 미국 소프트웨어산업 추진 전략 : 2015 미국 안보 및 경쟁력 확보를 위한 SW 추진전략, 6월 29일, 2005년  
 [6] Peter B. Lakey, McDonnell Douglas Corporation, St. Louis, System and Software Reliability Assurance Notebook, Rome Laboratory, pp. 7-10, 7-11  
 [7] Deun Lee, SunitaChulani, Barry W. Boehm, COQUALMO, USC/CSE, 2001  
 [8] Allen Nikora, CASRE-A Computer-Aided Software Reliability Estimation Tool, [http://www.openchannelfoundation.org/project/view\\_support.php?group\\_id=250](http://www.openchannelfoundation.org/project/view_support.php?group_id=250), 2002  
 [9] William Farr, Oliver Smith, SMERFS - Statistical Modeling and Estimation of Reliability Functions for Systems, <http://www.slingcode.com/smerfs/drffarr.php>, 1996..