

A Study on the Software Standardization and Simulator Design for Efficient Reliability Test in Combat System

Hwan-Jun Choi*

*Engineer, Naval R&D Center, Hanwha Systems, Pangyo, Korea

[Abstract]

In this paper, we propose the standardization architecture and weapon-sensor simulator for efficient reliability test in combat system. To reduce man-month of reliability test, application with high dependency on other module is selected and apply FORM. The proposed standardization architecture extracts common, variable elements and design patterns, S.O.L.I.D principles were applied. The proposed weapon-sensor simulator implements essential functions by identifying highly dependent element of other modules and the information from equipment can be directly received without processing by using communication middleware. As a result, it can replace actual ship-mounted equipment. In addition, it is possible to reduce the consumption rate of human resources when perform reliability test and modification time can be shorted.

▶ **Key words:** Naval Combat System, Standardization, FORM, Simulator, Reliability Test, DSS

[요 약]

본 논문은 함정 전투체계에서 신뢰성 시험 효율화를 위한 표준 아키텍처 및 무장-센서 시뮬레이터의 설계 방법을 제안한다. 제안된 표준화 아키텍처는 함정 전투체계 소프트웨어의 수정 공수 및 신뢰성 시험 수행 시간 단축을 위한 방법으로 함정 전투체계 소프트웨어에서 타 모듈의 의존도가 높은 응용을 선별하여 FORM 기법을 적용하였다. 이를 기반으로 공통요소와 가변요소를 추출, 추상화하였으며, 디자인 패턴 및 S.O.L.I.D 원칙을 고려한 아키텍처 디자인을 설계하였다. 제안된 무장-센서 시뮬레이터는 타 모듈의 의존성이 높은 요소를 식별하여 필수 기능을 구현하였고 통신 미들웨어에 기반 하여 장비의 정보를 가공 없이 직접 수신 받을 수 있도록 처리하였다. 그 결과 실제 함정 탑재장비를 대체하는 효과를 가진다. 또한 신뢰성 시험 수행 효율을 높이고 수정 사항 발생 시 수정시간을 단축할 수 있다.

▶ **주제어:** 함정전투체계, 표준화, FORM, 시뮬레이터, 신뢰성 시험, DSS

-
- First Author: Hwan-Jun Choi, Corresponding Author: Hwan-Jun Choi
 - *Hwan-Jun Choi (hwanjun627@hanwha.com), Naval R&D Center, Hanwha Systems
 - Received: 2022. 11. 23, Revised: 2022. 12. 08, Accepted: 2022. 12. 15.

I. Introduction

함정전투체계 시스템은 국내 도입된 이래로 지속적으로 함정 내에 탑재되고 있으며 필수적인 시스템으로 자리매김하고 있다. 이에 지속적인 업그레이드 및 추가적인 함정이 도입됨에 따라 소프트웨어의 추가 및 변경 또한 수시로 일어나고 있으며 시스템의 규모는 점차 대형화 되고 있는 추세이다[1-2].

특히, 국방 소프트웨어의 특성으로 인해 소프트웨어의 품질과 안정성에 대한 요구 수준은 일반 소프트웨어에 비해 월등히 높으며 이에 따른 품질 및 안정성 검증을 위한 절차 또한 무수히 존재하며 초기 개발 기간보다 소프트웨어 안정 및 기능 확인을 위한 시험 기간이 월등히 긴 상황이다[3-4]. 그 중 소프트웨어 신뢰성 시험은 개발 비용 중 상당 부분을 차지 할 정도로 많은 시간과 노력을 요구하는 절차이다[5]. 따라서 함정 전투체계 내의 수정사항이 발생하게 되면 추가적인 신뢰성 시험절차를 거쳐야 하고 이어지게 되고 추가적 자원 소모로 이어지게 된다.

또한 최근 시행하고 있는 동적 신뢰성 시험은 함정전투체계 요구사항에 기반 한 시험 절차를 통해 검증 되어야 하고 이에 필요한 시험 환경 구성 시간 또한 증가하고 있다.

본 논문에서는 함정 전투체계 시스템에서 신뢰성 시험 효율화를 위한 소프트웨어 표준화 및 무장-센서 시뮬레이터 설계에 대해 제안 하고자 한다. 제안한 표준화 소프트웨어는 함정 내 탑재되어있는 무장-센서의 의존성을 때어냄으로서 향후 수정요소를 최소화 하였고 제안한 무장-센서 시뮬레이터는 기존 장비 연동단과 타 전투체계 소프트웨어의 역할을 대체하여 신뢰성 시험의 인적 자원 소모를 대폭 감소 시켰다.

본 논문의 구성은 다음과 같다. 2장에서는 함정 전투체계 및 장비 연동단에 대한 개념과 시스템 구조에 대해서 설명하고 구조 개선을 위해 적용하는 방법론 및 기술들에 대해 서술한다. 3장에서는 장비 의존성이 높은 전투체계 응용인 교전 소프트웨어의 새로운 아키텍처에 대해 설명하고 신뢰성 수행 환경 개선을 위한 무장-센서 시뮬레이터를 제안한다. 4장에서는 제안된 소프트웨어와 무장-센서 시뮬레이터를 기존의 환경과 비교 후 개선 효과에 대해 분석하고 마지막으로 5장에서는 연구 결과에 대한 결론과 기 대효과에 대해 서술하며 마무리 한다.

II. Preliminaries

1. Naval Combat System

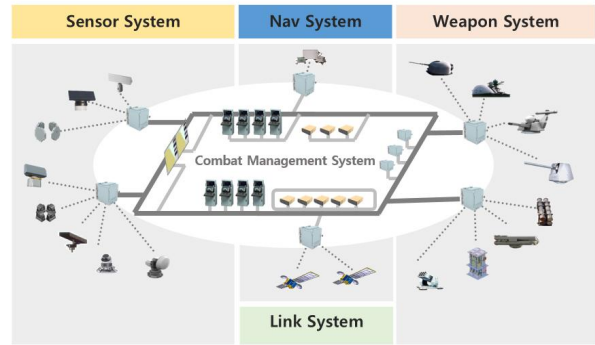


Fig. 1. System Architecture of Naval Combat System

함정 전투체계 시스템은 함정에 탑재된 모든 센서, 무장, 항해 지원 장비 등을 네트워크로 연결하여 통합된 전술 상황 정보를 만들어 공유하고 표적의 탐지, 추적에서부터 위협평가, 무장 할당, 교전 및 명중 여부 평가 분석에 이르기까지 지휘 및 무장통제를 자동화함으로써 위협에 대한 전투 효과를 극대화시키기 위한 통합체계로 지휘통제, 무장통제, 전술자료 교환 및 전시 등을 수행하는 시스템이다[6-7]. 함정 전투체계의 구성요소는 Fig. 1과 같으며, 함정의 특성에 따라 장비 및 구조는 변경이 가능하다.

2. Interface Control Unit System

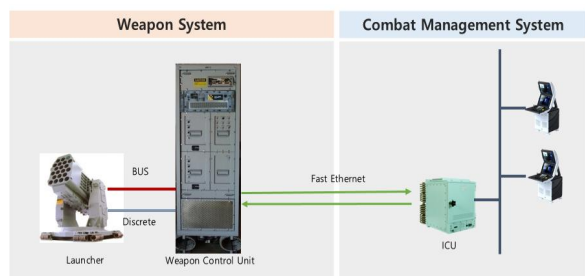


Fig. 2. Example of Interface Control Unit System

연동단 시스템은 함정 전투체계 시스템 내 장비 간 통신을 위해 합의된 연동 사양(물리적 체결, 프로토콜, 통신설정, 주기, 데이터 형식, 연동 개시/종료 조건 등)을 규정한 ICD(Interface Control Document)에 따라 서로 다른 두 체계 간 데이터를 교환하는 기능을 가진 시스템이다[8]. 같은 장비라도 제조사 및 모델에 따라 하드웨어 및 소프트웨어 통신 방식은 달라지기 때문에 타 체계 간 정보 공유를 위해 장비에서 보낸 데이터를 연동단 시스템에서

CMS(Combat Management System) 메시지로 변환하여 다른 체계 간 정보 공유 작업을 수행 한다. 또한 연동 사항에 위배 되는 정보를 장비에서 수신 시 연동을 단절해 데이터 무결성, 상호 체계 간 간섭을 피할 수 있으며 CMS에 생성된 정보를 종합해 장비를 컨트롤 할 수 있다. Fig. 2는 연동단 시스템의 구성 예시를 보여준다.

3. FORM (Feature-Oriented Reuse Method)

휘처(Feature)란 시스템에서 식별 가능한 추상화의 단위를 의미 하며 소프트웨어 아키텍처 재사용성 증가를 위해 발명된 휘처 중심의 모델링 기법을 휘처 모델이라 한다. 휘처 모델 중 요구공학을 위한 분석법인 FODA(Feature-Oriented Domain Analysis)[9]를 설계와 구현 까지 확장한 개발 방법을 FORM이라 한다[10]. FORM 기법을 사용하여 공통요소와 가변요소를 도출 하게 된다면 기존의 클래스에서 가변요소를 분리 가능하도록 설계 가능하게 되며 이는 소프트웨어 재사용성의 증가를 의미하게 된다.

4. Design Pattern

디자인 패턴은 소프트웨어 설계 시 재사용성을 늘리기 위해 고안 된 방법이며 워드 커닝햄(Ward Cunningham)과 켄트 벡(Kent Beck)이 1980년대 말 스피토크(Smalltalk) 언어를 사용한 프로젝트에서 패턴을 개발하여 사용하게 된 것이 소프트웨어 디자인 패턴의 시초이다. 이후 에릭 감마(Erich Gamma), 리처드 헬름(Richard Helm), 존 블리지드(John Vlissides), 랄프 존슨(Ralph Johnson)에 의하여 디자인 패턴을 체계적으로 정리되어 책으로 발표되었는데 이들 팀의 이름을 따서 'GoF(Gang of Pattern) 패턴' 이라고 불리며 GoF 디자인 패턴은 크게 생성패턴, 구조패턴, 행위패턴으로 분류되어 지며 총 23개의 패턴으로 정의 되어 있다[11].

5. DSS(Data Sharing Service)

DSS 통신이란 DDS(Data Distribution Service)[12-14] 기반의 통신기법이며 응용 간 물리적 위치에 제약 없이 데이터 공유 기능을 제공하여 데이터 중심 설계가 가능하도록 하는 서비스를 말한다. DSS 통신을 사용하게 되면 데이터 공유를 기반으로 규격에 상관없이 응용 간 필요한 메시지 공유가 가능하다는 장점이 있다.

III. The Proposed Scheme

본 절에서는 함정 전투체계 시스템에서 신뢰성 시험 효율화를 위한 소프트웨어 표준화 및 장비 시뮬레이터 설계 과정 대해 설명 하고자 한다. 표준화 대상 소프트웨어는 함정 내 탑재 된 연동장비의 의존도가 높은 교전 소프트웨어로 선정 하였다. 제안된 무장-센서 시뮬레이터는 교전 소프트웨어와 정보 교환하는 소프트웨어를 식별하고 필수적 요소를 구분 한 뒤 자유로운 데이터 공유가 가능한 시스템 구성에 초점을 맞추어 설계 하였다.

1. Naval Combat System Software Design

교전 소프트웨어는 함정 내 설치된 무장, 센서 장비의 자원을 전술 정보화 하여 전술 상황을 탐지하고 위협평가, 무장 할당 및 명중평가 등 교전 통제를 자동화하는 응용이다. 교전 소프트웨어는 함정 내 장비의 변경과 신규 장비 연동 등의 변화가 생기게 되면 연동장비 간 의존성이 높아 타 응용에 비해 시간과 비용이 추가적으로 발생하게 된다. 이 문제를 해결하기 위해 FORM 기법을 통해 가변요소를 식별하고 디자인 패턴 적용을 통한 새로운 교전 소프트웨어를 설계하였다.

1.1 Classification of Variable Factor

교전 소프트웨어의 장비 의존도를 낮추기 위해 휘처 모델 기반 분석법인 FORM 기법을 사용하였다. 그 일환으로 기존 사업을 진행하면서 식별 되었던 모듈들의 요구사항에 대해 분석하였고 분석한 정보에 기반 하여 휘처 요소를 분류 하였다. 그 결과 위협 표적 목록, 교전 제어, 표적 정보와 같은 기능은 모든 사업에 대해 공통적으로 들어가는 부분으로 식별하였다. 다음으로 위협 평가, 교전 채널 권고, 교전 자원 및 정보 관리, 통신 방법과 같은 경우는 사업에 따라 변경 될 수 있는 요소로 식별하였다. 위의 결과를 세분화 하여 필수적으로 구현되어야 하는 부분은 Mandatory, 사업에 따라 선택되거나 선택되지 않는 부분은 Optional, 여러 요소 중 하나만 선택되어야 하는 속성은 Alternative로 세분화 하여 휘처 모델을 설계 하였고 Fig. 3은 위의 결과를 바탕으로 휘처 모델로 표현한 그림이다.

- ThreatMgr : 위협표적 관리 클래스는 위협 기준에 대한 정보를 관리하고 이 정보를 기반으로 작전 성분별 위협 목록을 생성 및 관리하는 클래스이다. 위협표적 관리 기능은 작전성분 및 표적의 특성에 따라 별도의 평가가 필요하지만 평가를 하는 기준의 대부분이 특성에 상관없이 공통적인 절차를 가지고 있고 특성에 따른 일부의 변화만 존재하기 때문에 이에 가장 적합한 디자인 패턴 방식인 Template Method Pattern을 적용하였다.
- CDataManager : CMS간 메시지 통신을 수행하기 위한 클래스이다. 통신 클래스는 메시지를 구성하고 응용간 메시지 송수신 기능을 제공한다. 메시지 기반의 소프트웨어로 구성되어있는 전투체계의 특성 상 통신 클래스는 가장 중요한 부분 중 하나이며 통신 방식의 변화에 따른 영향성을 최소화시켜야 한다. 따라서 ISP(Interface Segregation Principle) 원칙에 기반한 Adaptor Pattern을 적용하여 통신 방식이 변경되어도 타 독립된 인터페이스 구조상 타 클래스에 영향이 없도록 설계하였다.
- CRscMgr : 센서, 무장 스펙 및 교전 가용성 등 교전에 필요한 리소스에 대한 정보를 관리하는 클래스이다. 그 중 정보 변경에 따른 실시간 처리가 필요한 데이터의 경우 즉시 반영이 필요하기 때문에 Observer Pattern을 적용하였고 옵저버 클래스인 CEmployable 클래스를 별도 구성하여 실시간 정보 처리가 가능하게 설계하였다.
- COsd : 자함에 대한 운동 및 위치 정보를 관리하는 클래스이다. 자함 정보는 오직 하나의 정보로 관리 되어야하고 자함 정보는 타 클래스와의 데이터 공유가 간편해야하기 때문에 Singleton Pattern을 적용하여 정보의 독립성 및 접근성을 높였다.

특정 정보의 경우 정보의 특성상 정보의 변경이 발생하게 되고 코드 수정이 필요한 경우가 존재하게 된다. 이러한 경우 클래스 설계만으로 적용이 어려우며 신뢰성 시험 관점에서 코드 수정은 신뢰성 시험 공수에 영향을 주게 된다. 이러한 문제를 해결하기 위해 추가로 가변 정보를 식별하였으며 해당 정보들의 코드 영향성을 줄이기 위해 파일 관리를 통한 정보를 읽은 후 적용하는 방식을 사용하였고 클래스를 별도 분리하여 정보 수정 시 코드 수정이 없도록 설계하였다. 아래는 파일 관리가 필요한 가변 정보의 목록이다.

Table 3. List of Variable Information

No	Data
1	Observer Position for Naval Surface Fire Support
2	Target Position for Naval Surface Fire Support
3	Weapon Restriction Rule
4	Weapon Restriction Area
5	Weapon, Sensor Spec Information
6	Fire Channel Information

Table 3.의 정보를 토대로 하드코딩 되어있던 정보는 파일로 분리 처리하여 반영 되도록 변경 하였고 추가적인 정보 수정 시 파일의 정보만 변경하여 반영되도록 적용 하였다.

2. Weapon-Sensor Simulator based on DSS

본 절에서는 제안된 무장-센서 시뮬레이터의 설계 과정에 대해 서술한다. 교전 소프트웨어의 요구사항 기반 동적 신뢰성 시험을 수행하기 위해서는 타 체계로부터 수신되어야 하는 필수 정보들이 존재하게 된다. 그 중 연동 장비에서 수신되어야 하는 정보의 경우 장비 간 연동 사양에 따라 서로 다른 통신방식으로 데이터를 전송하게 되고 이를 위해 별도의 연동단에서 장비 데이터를 가공 후 CMS와 공유하게 된다. 이러한 이유로 교전 소프트웨어의 요구사항 기반 동적 신뢰성 시험 수행 시 함 내에 탑재된 각 장비의 연동단 과 장비 시뮬레이터가 추가적으로 필요하게 되고 이는 신뢰성 시험 환경 구성을 위한 추가적인 인적 자원 소모를 의미한다. 이러한 문제를 해결하기 위해 교전 소프트웨어의 필수 반영 요소를 고려한 센서-무장 시뮬레이터에 대해 제안하고자 한다.

2.1 Classification of Necessary Information

제안된 연동단 시뮬레이터 설계를 위해 교전 소프트웨어 운용에 필요한 타 소프트웨어로 부터 수신 받는 필수 정보를 분류해 보았으며 그 결과는 아래와 같다.

Table 4. List of Necessary Info for Warfare SW

Information	from
Equipment Status	SM, Weapon, Sensor, ICU
Equipment Engagement Available Status	Weapon, Sensor, ICU
Engagement Information	Weapon, Sensor, ICU
Allocation, Deallocation Response	Weapon, Sensor, ICU
Point/Bearing Track Information	Sensor, ICU, TM
Torpedo/Missile Information	Sensor, ICU, TM
Reverse Track Information	Sensor, ICU, TM
Own Ship Information	OSD, ICU

Table 4.에서 TM(Track Management)은 표적관리, SM(Support Management)은 체계관리를 의미한다. 분류된 정보를 종합하면 교전 소프트웨어의 신뢰성 시험 수행을 위해서 추가적으로 무장, 센서, 각 장비 연동단, 표적 관리, 체계관리 소프트웨어가 지원되는 환경이 필요하다는 것을 알 수 있다.

2.2 Architecture Design

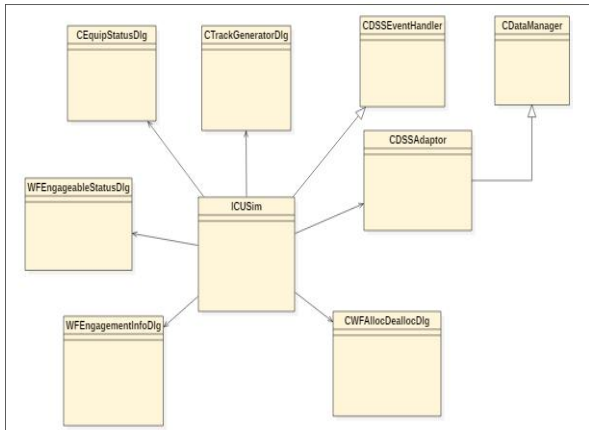


Fig. 6. Weapon-Sensor Simulator Class Diagrams Design

2.1 절을 통해 분류된 정보를 기반으로 신뢰성 시험 진행에 필요한 기능 및 시험 구성을 위한 인적 자원 소모를 절감시킬 수 있는 방안을 고려하여 아키텍처 디자인을 수행하였다. 설계 시 신뢰성 시험 환경 구성에 가장 문제가 되었던 통신 방식에 대한 문제를 우선순위로 고려하였다. 그 결과 DSS 통신 방식을 적용하여 장비의 정보를 직접 전달하게 된다면 불필요한 자원 소모율을 줄이게 되고 교전 소프트웨어와 정보 공유가 자유로워 즉각적인 데이터 처리가 가능해진다. 또한 타 소프트웨어에서 동작하는 기능은 제안된 무장-센서 시뮬레이터를 통해 동작하도록 구성하여 효율을 높였다. Fig. 6은 제안된 무장-센서 시뮬레이터의 클래스 다이어그램을 보여준다.

2.3 Simulator Implementation Result

본 절에서는 제안된 무장-센서 시뮬레이터에 대한 구현 결과 및 시뮬레이터를 통해 생성된 결과 화면을 보여준다.



Fig. 7. Equipment Status Interface

- 장비 상태 정보 처리 : 장비 상태 정보는 장비에서 전송한 정보를 연동단 소프트웨어 수신 및 가공하여 체계관리 소프트웨어에 전달하면 그 결과를 교전 소프트웨어에 송신하여 처리하는 구조였으나 시뮬레이터에서 그 역할을 대체하여 교전 소프트웨어에 결과처리를 직접 받을 수 있도록 개선하였다.



Fig. 8. Track Info Generator Interface

- 표적정보 생성 처리 : 표적 정보는 센서 장비에서 객체를 탐지하면 연동단 소프트웨어에 탐지 정보를 전달하고 해당 정보를 가공하여 표적관리 소프트웨어에 전달한다. 이후 표적관리는 CMS 표적 정보로 가공하여 그 결과를 교전 소프트웨어에 송신한다. 제안된 무장-센서 시뮬레이터에서는 표적 생성 처리를 교전 소프트웨어에 직접 가공하여 전송하도록 변경하였다.

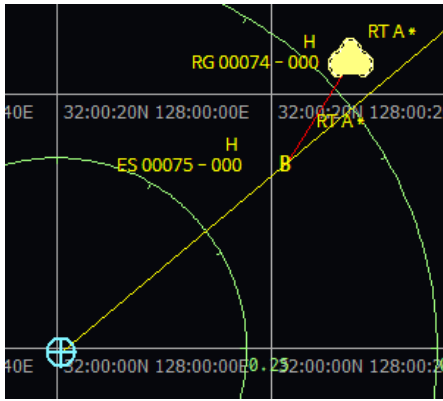


Fig. 9. Track Generate Result in CMS

- 무장-센서 할당 및 해제 처리 : 교전 소프트웨어에서 센서 추적 명령 또는 무장 할당 요청을 하는 경우 요청 대기 시간 내에 장비는 요청에 대한 응답 메시지를 전송해야 한다. 이러한 과정을 제안된 무장-센서 시뮬레이터를 통해 모사하여 설계 하였다.



Fig. 10. Alloc/Dealoc Response Interface

제안된 무장-센서 시뮬레이터를 통해 무장, 센서, 장비 연동단, 표적관리 및 체계지원 소프트웨어의 역할을 대체 수행하게 됨으로서 신뢰성 시험 환경 구성에 대한 자원 효율성을 증가 시키고 나아가 로컬 환경에서 테스트 가능한 환경을 제공해 준다. 이로 인해 기능 오류 발생 시 증상 재현에 따른 소모 시간을 감소시키는 효과를 가져다 줄 것이다.

IV. Performance Analysis

본 절에서는 함정 전투체계 운용 중 장비 및 기능 변경에 따른 인적 공수, 신뢰성 시험 수행 시간을 통해 제안된 소프트웨어 및 시뮬레이터의 실효성 및 자원 효율성을 검

토하였다. 비교 대상으로는 이전 사업에 사용 되었던 함정 전투체계 교전 소프트웨어를 기준으로 비교하였다.

1. Changeability Test

제안된 소프트웨어의 장비 및 기능에 대한 변경사항 발생 시 소요되는 작업량을 측정하기 위해 2가지 상황을 가정하였으며 이를 기반으로 수정 변화율을 측정했다.

- Scenario 1 : 운용중인 함정에 신규로 무장과 센서가 탑재 되었다. 무장은 대공전과 대함전에 대응 가능하다. 센서는 대공 표적에 대한 탐색 및 추적기능을 탑재 하고 있으며 맹목각이 존재 한다.

첫 번째 상황의 경우 기존의 응용에서는 교전 환경 중심의 클래스 구조로 인해 대공전, 대함전에 대한 모든 클래스 변경이 이루어지게 되며 무장, 센서의 정보를 처리하는 메시지에 대한 수정이 추가적으로 이루어지게 된다. 반면 제안된 응용은 교전 환경에 기반 하지 않은 교전 기능 중심 및 표준화 설계를 통해 무장, 센서에 관련된 클래스만 수정이 발생하며 추가적인 메시지는 파일 처리를 통해 수정이 발생하지 않는다.

- Scenario 2 : 교전 수칙이 추가되었다. 무장 A가 이미 교전중일 때 미사일 표적이 탐지 된다면 기존 교전에 대한 교전을 취소하고 미사일 표적에 대해 대응하도록 수정이 필요하다.

두 번째 상황의 경우 두 소프트웨어 모두 추가적인 메시지 수정이 이루어지지 않지만 기존 응용은 무장 A가 대응할 수 있는 모든 교전 환경에 대한 코드 수정이 발생하게 되고 교전 수칙에 대한 수정으로 클래스 내 의존성이 높은 클래스인 Funcc 클래스의 변경이 이루어져 코드 영향성이 높아지게 된다. 반면 제안된 응용은 기능별 클래스 구분으로 인해 기능이 변경된 클래스에서만 수정이 발생하게 된다.

수정 변화율(C_{rate})은 수정되어야 할 메시지와 클래스 수를 기반으로 산정하였으며 수식 (1)과 같이 정의된다.

$$C_{rate} = ((C_m / C_t) + (M_m / w)) * 100 \quad (1)$$

여기서 C_m 는 수정된 클래스의 수, C_t 는 전체 클래스의 수, M_m 은 수정된 메시지의 수를 의미하며 w 는 가중치를 의미한다. 변화율 산출 수식에서 총 메시지의 수는

수정 공수에 직접적인 영향을 미치지 않기 때문에 정규화를 통한 응용별 고정 상수 값을 적용하였으며 메시지 수정 공수에 기반 하여 가중치를 30으로 산정하였다[16].

Table 5. Changeability Test Result - Scenario 1

-	Modified Msg	Total Class	Modified Class	Change rate(%)
Existing SW	5	53	23	60.06
Proposed SW	0	34	3	8.82

Table 6. Changeability Test Result - Scenario 2

-	Modified Msg	Total Class	Modified Class	Change rate(%)
Existing SW	0	53	23	19.98
Proposed SW	0	34	3	5.88

Table 5-6의 결과를 살펴보면 제안된 소프트웨어를 사용하는 경우 기존 사업에 사용하는 소프트웨어보다 변화율이 14.1~51.24% 차이 나는 것을 확인 할 수 있고 이는 인적 자원 소모율을 줄이는 것을 의미한다.

2. Reliability Test Result

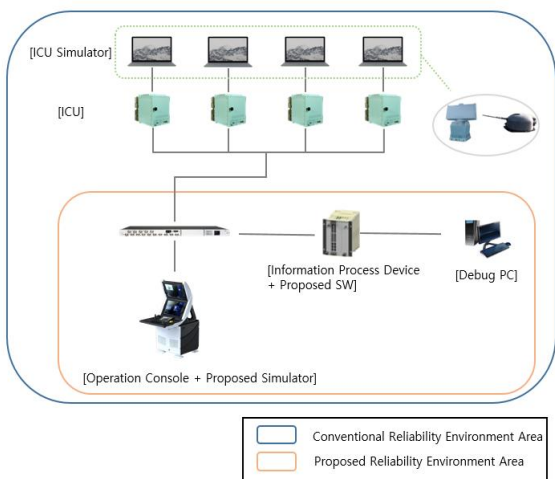


Fig. 11. Reliability Test Environment Site

제안된 소프트웨어와 무장-센서 시뮬레이터를 사용하여 요구사항 기반 동적 신뢰성 시험을 수행 하였을 때 소요되는 작업 공수 시간을 산정하였다.

Fig. 11은 신뢰성 시험 구성 환경을 보여 준다. 기존의 환경은 장비 모의를 위한 추가적 장치가 무장, 센서의 수 만큼 필요하게 된다. 반면 제안된 시뮬레이터를 적용한 험

환경에서는 운용 콘솔에 제안된 시뮬레이터를 설치 및 실행이 가능한 환경이 가능하다. 이로 인해 장비 모의 환경 구성을 위한 시간이 절감 되는 효과를 가질 수 있다.

신뢰성 시험은 사전 설계를 통해 식별된 소프트웨어 기능 요구사항에 모두 충족하는 수행 절차를 작성 후 준비된 신뢰성 시험장에서 시험을 수행하였을 때 해당 응용의 모든 구문이 수행되는지 확인 되면 수행 완료의 기준으로 판단한다. 본 시험의 결과를 측정하기 위한 신뢰성 시험 도구는 Quality Scroll Cover를 사용하였으며 Fig. 12은 운용화면을 보여준다[17]. 신뢰성 시험 수행 후 수행 결과 파일을 등록하면 COVER를 통해 수행결과를 확인 할 수 있다.

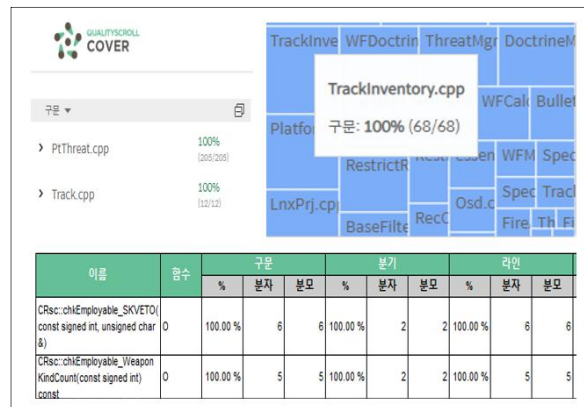


Fig. 12. Reliability Test Tool - COVER

작업 공수 시간 산정 기준은 신뢰성 시험을 수행하는 환경을 구축하는데 소요되는 시간과 신뢰성 시험 수행 후 해당 응용의 모든 구문이 수행되는데 까지 소요되는 시간의 합으로 산정하였으며 그 결과는 Table 7.과 같다.

Table 7. Execution Time to Perform Reliability Test

-	Prepare Env Set	Execute Reliability Test	Total Days
Existing SW	5	15	20
Proposed SW & Sim	0.5	5	5.5

결과를 살펴보면 제안된 응용은 기존의 응용보다 14.5 일 더 빠른 수행 시간을 보여준다. 그 이유는 운용 콘솔과 정보처리 장치만 사용하면 신뢰성 시험을 수행할 수 있는 환경이 구성되고 기능 중심의 아키텍처 설계로 인해 불필요한 반복 수행이 없어졌기 때문이다. 결과적으로 제안된 응용 및 시뮬레이터를 통해 자원 효율관점에서 기존의 환경보다 실효성이 있다고 판단 할 수 있다.

V. Conclusions

본 논문에서는 신뢰성 시험 효율을 위한 함정 전투체계 표준화 아키텍처와 무장-센서 시뮬레이터를 제안하였다. 제안된 함정 전투체계 아키텍처는 FORM 분석법에 기반한 클래스 디자인과 가변 데이터 의존성을 줄여주는 데이터 처리를 통해 신규 사업 및 수정사항 발생 시 변화를 차이가 기존 응용 대비 14.1~51.24% 낮은 결과를 보여준다. 제안된 무장-센서 시뮬레이터는 무장, 센서, 장비 연동단, 표적관리 및 체계지원 소프트웨어의 역할을 대체 수행함으로써 기존 시험 환경 대비 14.5일 빠른 동적 신뢰성 시험 수행 결과를 보여준다. 이는 환경 구성에 대한 자원 효율성을 증가 시키고 나아가 로컬 환경에서 테스트 가능한 환경을 기반으로 기능 오류에 대한 처리 속도 또한 상승시키는 효과를 가져다준다. 본 연구는 실 운용 환경에 유용하게 쓰일 것이며 나아가 함정전투체계 개선에 대한 연구 자료로 효과적으로 활용 될 것이다.

REFERENCES

- [1] Arciszewski H. F, De Greef T. E, Van Delft J. H, "Adaptive automation in a naval combat management system" *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, Vol. 39, No. 6, pp. 1188-1199, Nov. 2009. DOI: 10.1109/TSMCA.2009.2026428
- [2] Soon-Joo K, "An Examination on Overseas Technology Trend and Domestic Development Pattern of the Naval Combat Management System", *Korea Association of Defence Industry Studies*, Vol. 16, No. 9, pp. 237-258, Jan. 2009.
- [3] Ki-Chang Kim, Yong-Hwan Kim, Ju-Hwan Shin, Ki-Jun Han, "A Case study on Application for Software Reliability Model to Improve Reliability of the Weapon System" *Journal of KISS: Software and Application*, Vol. 38, No. 8, pp. 405-418, Aug. 2011.
- [4] Seung-Young Lee, Dong-Hwan Kim, "Policy Trends of Defense Software Quality Improvement" *Communications of KIISE*, Vol. 38, No.9, pp. 23-35, Jan 2020.
- [5] Sang-Gyu Jeong, Young-Jin Seo, Min-Uk Jang, Yoon-Woo Lee, "Research on Application of Requirements-based Software Reliability Verification to Domestic Military Aircraft" *Journal of Advanced Navigation Technology*, Vol. 22, No. 4, pp. 264-270, Aug. 2018. DOI: 10.12673/JANT.20.18.22.4.264
- [6] Young-Dong Heo, "A Study on the Standardization of System Support Software in the Combat Management System" *Journal of the Korea Society of Computer and Information*, Vol. 25, No. 11, pp. 147-155, Nov. 2020. DOI: 10.9708/JKSCI.2020.25.11.147
- [7] Ki-Tae Kwon, Ki-Pyo Kim, Hwan-Jun Choi, "Design of the Scalable Naval Combat System Software using Abstraction and Design Pattern" *Journal of the Korea Society of Computer and Information*, Vol. 24, No. 7, pp. 101-108, Jul. 2019. DOI: 10.9708/JKSCI.2019.24.07.101.
- [8] Chi-Sun Back, Jin-Hyang Ahn, "A Study of the Standard Interface Architecture of Naval Combat Management System" *Journal of the Korea Society of Computer and Information*, Vol. 26, No. 1, pp. 147-154, Jan. 2021. DOI: 10.9708/JKSCI.2021.26.01.147.
- [9] C. Kang, et al., "Feature-oriented domain analysis(FODA) feasibility study" *Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst*, 1990.
- [10] C. Kang, S. Kim, "FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures" *Annals of Software Engineering*, Vol. 5, No. 1, pp. 143-168, Jan. 1998. DOI: 10.1023/A:1018980625587
- [11] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns : Elements of Reuseable Object-Oriented Software" *Person Education, Inc*, 1995
- [12] DDS, <https://www.omg.org/omg-dds-portal/>
- [13] OMG, *Data Distribution Service for Real-time Systems Version 1.4*, Apr. 2015
- [14] OMG, *The Real-time Publish-Subscribe Protocol(RTPS) DDS Interoperability Wire Protocol Specification*, Sep. 2014
- [15] Martin R. C, "Design Principle and design patterns" *Object Mentor*, Vol. 1, No. 34, pp. 597, Nov. 2000.
- [16] Hyoung-Kweon Kim, "A study for the reduction of the SW reliability test time and human errors using the SW reliability test automation" *Journal of the Korea Society of Computer and Information*, Vol. 20, No. 10, pp. 45-51, Oct. 2015.
- [17] *Quality Scroll Cover*, http://www.suresofttech.com/html/tool/quality_cover/

Authors



Hwan-Jun Choi received the B.S and M.S. degrees in Information and communication system from Sejong University, Korea, in 2013 and 2015 respectively. He is interested in Combat System Software, sensor, weapon interface control unit information processing.