

## 개발단계 무기체계의 소프트웨어 품질검증체계 연구

윤경환\*, 박만춘, 주진천, 엄원용, 오진우, 윤재형, 김종규  
국방기술품질원

### Research on SW Verification System for Weapon System in Development Stages

Gyeonghwan Yoon\*, Manchun Park, JinChun Ju, Wonyong Eom,  
Jin-Woo Oh, Jae-Hyeong Yun, Jong-Kyu Kim  
Defense Agency for Technology and Quality

**요약** 국방기술품질원(이하 기품원)에서는 무기체계 소프트웨어가 개발되는 과정에서부터 품질보증 활동을 수행하고 있다. 개발과정마다 연구개발주관기관이 약속된 활동을 수행하고 개발과정에서의 산출물이 적합한지 점검하는 것이다. 이러한 개발단계의 품질보증 활동을 고도화하면 무기체계의 운용과정에서 발생할 수 있는 문제점들을 미리 예방하여 소프트웨어 품질을 전반적으로 향상시킬 수 있다.

한편 기품원의 SW 기술지원부서에서는 무기체계의 운용단계에서 발생하는 SW 결함을 종합하여 데이터베이스(이하 DB)로 관리하고 있다. 무기체계 운용 시 발생했던 SW 결함사례들을 현재 개발 중인 무기체계에서도 발생 가능한지 미리 점검할 수 있다면 유사한 SW 결함이 재발하는 것을 개발단계에서 방지할 수 있을 것이다.

본 논문에서는 무기체계 개발단계에서 SW 결함을 예방하기 위한 체계적인 업무 방법과 절차를 연구하였다. 무기체계 소프트웨어의 개발단계에서부터 기존에 발생했던 SW 결함을 검증하는 체계를 구축한다면 조직적으로 SW 결함을 예방할 수 있다. 체계적인 검증업무 절차를 통해 SW 기술지원 담당자의 개인역량이나 우연에 따른 품질관리 편차를 줄이고 SW 품질을 전반적이고 지속적으로 향상시키는 것이다.

본 논문에서는 SW 결함 DB에서 점검 항목들을 추출한 SW 품질개선 체크리스트를 품질검증체계에서 활용하였다. 또 SW 결함이 발생했던 기존의 소스 코드 구현 패턴과 현재 개발 중인 소스 코드의 구현 패턴을 비교하는 소스코드 구조분석도 품질검증체계에서 활용하였다. 본 논문에서 소프트웨어 품질검증체계를 통해 SW 결함을 실제로 예방한 사례들을 다루었으며 SW 품질검증체계의 효과성을 확인하였다.

**Abstract** The Defense Agency of Technology and Quality (DTaQ) conducts software quality assurance activities from the process of developing weapon system software. By upgrading software quality assurance activities at this stage of development, problems that may occur during the operation of a weapon system can be prevented in advance and the quality of weapon system software could be improved effectively.

Meanwhile, DTaQ data on collects SW defects that occur during the operation stage of weapon systems and manages them as a database. Advance checks on whether cases of SW defects that occurred during the operation of the weapon system can also occur in the weapon system currently under development would enable the prevention of the recurrence of similar SW defects in the development stage.

In this paper, we present our findings of research on operations and procedures aimed at the prevention of SW defects in development stages of a weapon system. The establishment of a Software Quality Verification System to verify SW defects enables systematic prevention of SW defects is possible.

In this paper, SW Quality Management Checklist and Source Code Structural Analysis method were implemented. Additionally, we dealt with a real case of SW defect prevention using the Software Quality Verification System, confirming the effectiveness of the system.

**Keywords** : Weapon System Development Process, Software Quality Assurance Activity, SW Defect Database, Software Quality Management Checklist, Source Code Structural Analysis

\*Corresponding Author : Gyeonghwan Yoon(Defense Agency for Technology and Quality)

email: sunshine@dtaq.re.kr

Received February 22, 2023

Accepted April 7, 2023

Revised March 16, 2023

Published April 30, 2023

## SW Quality Assurance Activity

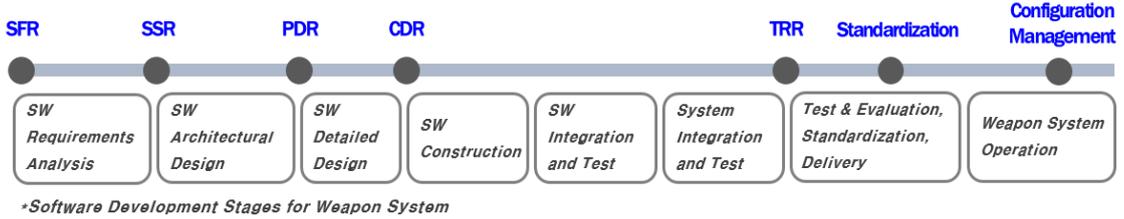


Fig. 1. Software quality assurance activity according to software development stages for weapon system

## 1. 서론

현재 방위력 개선을 위해 방위사업청 주관으로 도입하는 무기체계 소프트웨어는 「무기체계 소프트웨어 개발 및 관리 매뉴얼」에 따라 개발된다[1]. 사업추진 단계에 따라 Fig. 1과 같이 소프트웨어 요구사항 분석(SW Requirements Analysis) 소프트웨어의 구조설계(SW Architectural Design), 소프트웨어 상세설계(Software Detailed Design), 구현(SW Construction), 소프트웨어 통합 및 테스트(SW Integration and Test), 시스템 통합 및 테스트(System Integration and Test) 규격화(Test & Evaluation, Standardization, Delivery), 운용(Weapon System Operation)의 단계에 따라 체계적으로 개발된다. 그리고 이러한 개발단계마다 소요군, 기술지원기관, 연구개발주관기관 등 무기체계의 사용과 관리 주체가 소프트웨어의 개발과정과 산출물을 조직적으로 관리하고 있다.

체계적이고 조직적인 무기체계 소프트웨어의 개발 관리에도 불구하고 소요군이 무기체계를 실전에서 운용 시 많은 문제점이 발생하고 있다. 무기체계를 개발하는 과정에서의 부정확한 소프트웨어 개발 요구사항, 잘못된 설계로 인한 연동 기능 오류, 기능 동작 시 예외 상황 대처 부족 등 여러 가지 사유가 문제점의 원인이다.

이러한 문제점들을 해결하기 위해 운용단계에서 무기체계 소프트웨어나 하드웨어를 수정하는 기술변경(Engineering change)이 발생한다. 기술변경은 소스코드 수정, 해당 기술자료 묶음의 수정을 포함하며 기능 테스트, 신뢰성 시험의 재수행 등 연구개발주관기관의 무기체계 재개발 비용뿐만 아니라 방위사업청, 기술지원기관의 형상관리 업무 증가, 소요군의 전력 공백과 같은 비용도 발생시킨다. 최근 3년간 이러한 SW 기술문서의 변경 건수는 국방표준중합정보시스템의 규격제안서 현황('23.1.6) 기준으로 '20년도 206건, '21년도 260건, '22

년도 329건으로 계속 증가하였고 기술변경에 따른 사용주체, 관리 주체의 품질비용도 계속 상승하였다.

무기체계 운용단계에서 발생하는 이러한 기술변경을 미리 예방할 수 있다면 불필요한 비용을 줄이고 소요군의 만족도를 향상시킬 수 있을 것이다. 따라서 기술변경의 원인을 파악한 뒤 사전에 기술변경의 원인을 제거하는 것이 필수적이다.

본 논문은 운용단계에서 발생한 무기체계의 여러 가지 SW 변경 원인을 분석하고 이를 바탕으로 반복적으로 발생하는 SW 변경을 무기체계 개발단계에서 체계적으로 줄이기 위한 절차와 방법, 기법에 관한 연구이다. 또 SW 기술지원 담당자의 개인역량이나 우연에 따른 품질관리 편차를 줄여 SW 품질을 전반적이고 지속적으로 향상시키기 위한 연구이다.

본 논문은 무기체계 소프트웨어의 개발단계 품질보증 방법, SW 결함 DB, 소스코드 구조 분석 등 개발단계 SW 품질검증체계의 구성 요소들을 먼저 서술하고 이후 개발단계 SW 품질검증체계의 구축과 적용에 관한 내용을 다룬다.

## 2. 이론적 배경

### 2.1 무기체계 소프트웨어 정의

무기체계란 유도무기, 항공기, 함정 등 전장(戰場)에서 전투력을 발휘하기 위한 무기와 이를 운영하는데 필요한 장비, 부품, 시설, 소프트웨어 등 제반 요소를 통합한 것으로서 대통령령이 정하는 것을 말한다[2]. 대통령령으로 현재 무기체계는 지휘통제·통신 무기체계, 감시·정찰 무기체계, 전차·장갑차, 전투함, 전투기 등의 무기체계부터 모의분석·모의훈련 소프트웨어, 전투력 지원을 위한 필수장비까지 폭넓게 지정하고 있다[3]. 무기체계 소프트웨어란 이러한 무기체계에 포함된 소프트웨어를 말한다[4].

## 2.2 개발단계 품질보증과 기품원 역할

소프트웨어 품질보증은 사업기간 동안 소프트웨어 산출물과 사업에 적용되는 수명주기 프로세스가 계약에 명시된 요구사항을 따르고 있고, 정의된 계획을 준수하는지를 보증하기 위한 활동으로서 크게 개발 프로세스 보증, 산출물 보증으로 나눌 수 있으며 검토회의, 검증, 확인, 감리 등을 통해 수행된다[1].

기품원은 무기체계의 개발단계에서 프로세스 보증, 산출물 보증의 활동을 지원하며 무기체계 소프트웨어 개발 프로세스는 이전 페이지 Fig. 1과 같다. 기품원의 SW 개발단계 프로세스 품질보증 활동으로는 체계요구사항 검토회의(SFR, System Requirement Review), 소프트웨어요구사항 검토회의(SSR, Software Specification Review), 기본설계 검토회의(PDR, Preliminary Design Review), 상세설계 검토회의(CDR, Critical Design Review), 시험준비 검토회의(TRR, Test Readiness Review)가 있다. 각 개발단계의 활동과 산출물이 계약으로 계획된 사항들과 일치하며 각각의 결과물은 품질목표에 부합하는지를 소프트웨어 검토회의(Software Review) 활동을 통해 점검하는 것이다.

각각의 개발단계 산출물은 Table 1과 같으며 각각의 검토회의 전이나 규격화(Standardization) 시 산출물의 적합성을 점검한다. 이후 무기체계의 운용단계에서는 무기체계 SW의 버전과 HW, 도면 등의 변경을 점검하는데 이를 형상관리(Configuration Management)라 하며 무기체계의 기술변경은 형상관리를 구성하는 업무로 구분할 수 있다.

Table 1. Standard document outputs of each software development process stage

Process stage	Standard document output
SW Requirements Analysis	Software Requirement Specification(SRS)
SW Architectural Design	Software Design Description(SDD) Interface Design Description(IDD)
SW Detailed Design	
SW Construction	Software Test Plan(STP) Software Test Description(STD)
SW Integration and Test	Software Test Result(STR)
System Integration and Test	Software Installation Plan(SIP) Software Catalog Specification(SCS)

소프트웨어 검증(Software Verification)은 각 개발 단계에서 작성된 소프트웨어 산출물이 요구사항을 올바

르게 반영하는지 점검하는 활동으로 요구사항 검증, 설계 검증, 코드 검증, 통합 검증, 문서화 검증 등으로 나눌 수 있다. 현재 기품원은 소프트웨어 검토 회의와 소프트웨어 검증 활동을 통해 무기체계 SW의 품질을 관리하고 있다. 이러한 기품원의 품질관리 활동을 고도화하면 무기체계의 소프트웨어의 품질을 체계적으로 향상시킬 수 있을 것이다.

## 2.3 기술변경과 SW 결함 DB

결함은 CMMI, TMMi, SWEBOK, IEEE 1044에서 다양하게 정의된다. 그렇지만 방위산업에서 결함은 시스템의 오류뿐만 아니라 고객이 요구하고 의도한 것과 산출물을 다르게 만듦으로써 재작업을 요구하는 모든 원인으로 정의할 수 있다[5].

위와 같이 결함에 대한 넓은 정의를 바탕으로 무기체계의 운용단계에서 발생하는 기술변경 원인을 결함으로 정의할 수 있다. 그리고 이에 따라 기술변경 원인 중 SW가 문제였던 기술변경 사례들은 SW 결함사례로 분류할 수 있다.

SW 결함 DB는 8대 무기체제로 구분된 모든 무기체계 결함을 포함한다. 기존의 연구개발주관 기관에서 관리하는 결함 DB의 경우 결함이 발생한 연구개발주관기관 내부에서만 제한적으로 관리되었다. 그렇지만 국방기술품질원에서 관리하는 SW 결함 DB는 모든 연구개발 주관기관들이 만든 결함을 통합적으로 관리하는 특징이 있다.

기품원의 SW 기술지원 부서에서는 위의 SW 결함사례가 발생한 상황, 결함의 원인, 연구개발주관기관의 개선안 등을 조사하여 관리한다[6]. 또 이러한 결함이 SW 연구개발 단계 중 어느 단계에서 원인이 발생했는지 분석하고 있다[7]. SW 결함의 원인이 발생한 연구개발 단계를 파악하면 유사한 무기체계의 개발 시 해당 연구개발 단계에서 SW 결함을 미리 예방할 수 있기 때문이다. 결함 예방뿐만 아니라 기존의 무기체계 SW 결함을 해결한 개선 방안을 유사 무기체계 개발 시 선제적으로 적용하는 SW 품질관리 방안도 연구 중이다[8]. 기술변경 사례에서 SW 결함을 추출하여 관리하는 SW 결함 DB는 이처럼 활용성이 높으며 SW 품질검증체계에서도 SW 결함 DB를 활용한다.

## 2.4 SW 품질개선 체크리스트와 SW 결함 DB

SW 결함을 예방하기 위해 현재 개발 중인 무기체계 사업을 분석하면 기존에 개발되었던 유사 무기체계를 찾을 수 있다. 기존의 무기체계에서 발생했던 SW 결함사례

를 SW 결함 DB에서 추출하여 이러한 SW 결함이 현재 개발 중인 무기체계에서 발생 가능한지 체계적으로 점검하는 것이 필요하다.

Table 2. Software Quality Management Checklist for weapon system

Perspective	Keywords of SW defects	Count
Input	User input, Electronic analog/digital signal	7
Output	Output display, Electronic analog/digital signal output	3
Communications	Wrong message, Fault frame, Buffer overflow, Transmitter/receiver SW module review	10
Data	Initialization condition, False data	12
Diagnostic functions	Built in test, Booting, Fault condition	15
Object control	Engine, motor, Machine gun	13
Logic	Unit of calculation, User mode, Multi-thread	17
Interface	Interface Control Document(ICD) change review	11
User	Report results, Missing information	18
HW	HW change history review	4
Standardization	Missing computer files in KDSIS, Different computer file information between KDSIS and documents	13
Total	-	123

이러한 점검 항목들을 구체화한 것이 SW 품질개선 체크리스트(Software Quality Management Checklist)이다. SW 결함은 Table 2와 같이 11가지 관점에서 점검할 수 있으며 기동화력분야의 21'년도 기술변경을 바탕으로 도출하였다.

각 점검 항목은 입력, 출력, 통신, 데이터, 고장, 제어, 로직, 연동, 사용자, 하드웨어, 규격화 등으로 구분한다. 각각의 구분된 점검 항목은 세부적인 점검 항목으로 나뉘며 세부 항목은 기존의 SW 결함사례와 연결되어 있다. 그래서 SW 결함 발생상황, SW 결함 원인, 개선안을 상세히 확인할 수 있으며 현재 세부항목은 Table 2와 같이 총 123개로 구성되어 있다.

SW 품질개선 체크리스트는 기존의 무기체계 결함사례와 현재 개발 중인 무기체계의 상황을 체계적으로 비교하여 SW 결함 발생 가능성을 검토할 수 있게 해주며 기존 SW 결함의 개선안도 찾아볼 수 있는 추적성도 갖고 있다. SW 품질검증체계에서는 반복되는 SW 결함을 찾기 위해 SW 품질개선 체크리스트를 활용하게 된다.

## 2.5 소스코드 구조 분석(SW Source code Structural Analysis)

소스코드 구조 분석이란 현재 개발 중인 무기체계의 소스 코드에서 SW 결함을 직접 확인하기 위해 기품원 내부적으로 시도하는 점검 방법이다. 무기체계 운용 중에 발생한 SW 결함을 분석해보면 파급력이 큰 결함의 경우 소스 코드에서 구조적인 문제점을 가지고 있는 경우가 많았다. 가장 대표적인 문제는 무기체계의 운용 중에 예상하지 못한 예외적, 간헐적인 상황에서 무기체계의 제어가 누락되거나 잘못된 방식으로 제어되는 경우이다. 구현된 소스 코드를 구체적으로 확인해보면 C언어의 경우 "if 제어문"의 작성 중 "else 구문"이 누락되거나 "else 구문"에서 잘못된 제어가 된 경우이다. 결국 소스 코드 구조 분석은 이러한 예외적인 상황의 동작을 소스 코드에서 중점적으로 리뷰하는 것이다.

또 소스코드 구조분석은 데이터 처리 방식과 로직을 소스코드에서 리뷰하여 결함이 발생했던 소스코드의 구현방식과 유사한 패턴으로 구현되었는지 비교해 보는 것도 포함한다. SW 품질검증체계에서는 소스코드를 점검할 수 있는 단계에서 소스코드 구조분석을 수행하게 된다.

## 3. 개발단계 SW 품질검증체계 구축

### 3.1 검증 프로세스 정립

개발단계 SW 품질검증체계는 무기체계의 SW 결함을 예방하는 체계적인 업무 절차와 방법으로 Fig. 2와 같이 도식화할 수 있다.

개발단계 SW 품질검증체계는 가장 먼저 Fig. 2의 형상관리(Configuration Management) 업무를 통해 시작된다. 제품의 형상을 변경하고 관리하는 무기체계의 기술변경 업무를 통해 SW 결함사례를 식별하고 이러한 내용을 SW 결함 DB로 종합한다. 이때 사용자 불만(User Complaints), 기능 결함(Functional Faults), 비기능 결함(Non Functional Faults)과 같이 SW 결함의 주요 특징이 Fig. 2와 같이 구별되어 저장된다. 또 결함의 발생상황, 결함 원인, 결함의 유입 시기 등도 함께 SW 결함 DB로 저장된다.

SW 결함 DB를 통해 이전 페이지의 Table 2의 항목으로 구성된 SW 품질개선 체크리스트를 선별한다. 개발단계 SW 품질검증체계에서는 이러한 SW 품질개선 체크리스트를 가장 중점적으로 활용하며 개발단계 시점과 무기체계 특성에 맞춰 SW 품질개선 체크리스트를 재구성

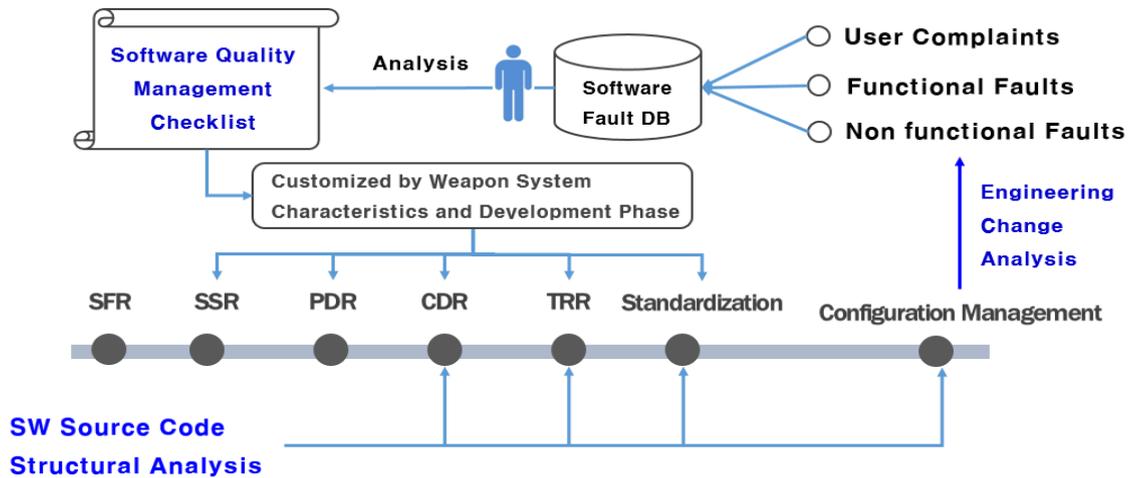


Fig. 2. SW Quality Verification System in weapon software development process

(테일러링)한 뒤 사용한다. SW 품질개선 체크리스트를 통해 유사 결함이 식별되는 경우 각 개발단계의 특성에 맞는 결함 예방 활동을 수행하게 된다.

개발단계의 CDR 이후부터는 산출물로써 구현된 소스 코드의 점검이 가능하다. 이런 단계에서는 위의 Fig. 2와 같이 소스코드 구조 분석을 통해 SW 결함을 직접 점검할 수 있다.

## 4. SW 품질검증체계 적용

### 4.1 요구사항 검토단계에서 결함 예방

다음 페이지의 Fig. 3은 개발단계 SW 품질검증체계를 적용하여 전장모의훈련체계에서 결함 예방 사례를 도식화한 것이다.

전장모의훈련체계는 실제 전장 상황과 유사한 상황을 컴퓨터 프로그램으로 구현하여 지휘관, 참모의 전투지휘 능력을 배양하고 관련 담당자들의 전쟁 수행 절차를 숙달하기 위한 모의훈련 시스템이다. 소요군이 운용 중인 A 전장모의훈련체계에서는 유사한 전장모의훈련체계 개발 시 요구사항 검토단계에서 예방이 가능한 결함이 발생했고 해당 결함이 SW 결함 DB에서 관리되고 있었다. A 전장모의훈련체계에서는 훈련 영상 재생 중 음향과 음성이 포함되지 않은 영상이 저장되어 개선 요구가 있었지만 음향과 음성 녹화가 불가능하도록 설계되어 개선이 불가능했던 것이다. B 전장모의훈련체계 개발 시 SW 결함 DB에서 이러한 결함사례를 SW 품질점검 체크리스트로 추출하고 해당 무기체계의 개발단계에서 이러한 결함

을 예방하는 조치를 수행하였다. B 전장모의훈련체계의 요구사항을 검토할 때 훈련 영상에 음향과 영상이 필요한지 미리 점검하여 SW 결함을 방지한 것이다.

### 4.2 소스코드 구조분석으로 결함 예방

지휘정찰 분야의 무기체계의 경우 신호처리 부분이 가장 핵심적이다. 무기체계의 신호처리 모듈에서는 FFT(Fast Fourier Transform)의 연산식을 이용해 시간 영역의 신호를 주파수 영역의 신호로 변환하기도 한다. 이후 주파수 신호를 필터링하거나 신호의 크기를 이용해 신호를 변환하여 사용하기도 한다. 이러한 연산의 특징은 한 수식의 결과가 다른 수식에 입력되고, 다시 연산의 결과가 다음 수식으로 입력되어 연결되는 특징이 있다. 이렇게 연속적인 신호의 연산 중 하나의 수식에서 오버플로우(Overflow, 정수 연산의 결과가 허용 범위를 초과)가 발생하면 잘못된 신호가 생성되고 연쇄적으로 잘못된 값이 전달되며 최종 결과값에서 오류가 발생한다. 이러한 오류는 디버깅 시 어느 수식에서 잘못되었는지 찾기가 어렵다. 이러한 결함은 SW 결함 DB에서 관리되고 있으며 신호를 처리하는 것이 중요한 무기체계 개발 시 CDR 단계에서 해당 결함을 미리 방지하기 위해 이 내용을 점검하고 있다. 지휘정찰분야의 C 무기체계의 개발 시 소스코드 구조분석으로 위와 유사한 형태의 결함이 존재할 것으로 식별되었으며 해당 무기체계의 신호처리 모듈에서 정수 연산 시 오버플로우 방지 함수를 사용하여 오버플로우 문제를 원천적으로 차단하기를 권고하였다. 이러한 내용은 CDR 단계에서 반영되었으며 발생 가능한 SW 결함을 예방할 수 있었다.

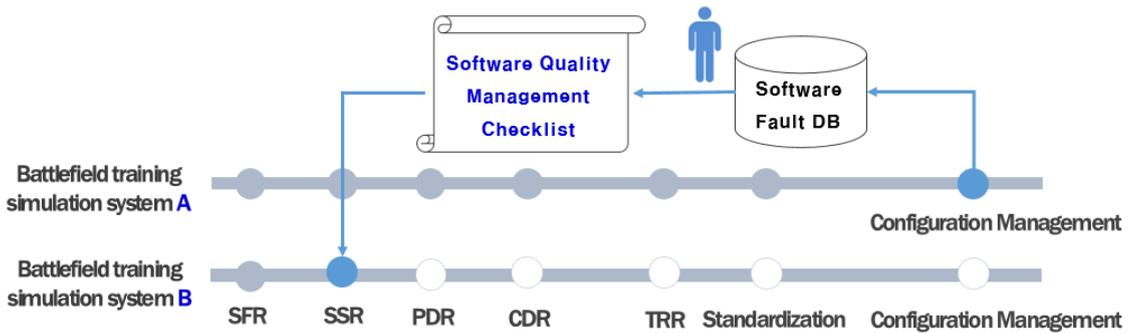


Fig. 3. Application of SW Quality Verification System

### 5. 결론

무기체계의 운용단계에서 발생하는 기술변경을 예방하는 것은 불필요한 재개발 비용을 제한하고 형상통제를 수행하는 기술지원기관의 업무량을 감소시키며 무기체계의 신뢰성과 품질을 향상시키는 것이라 할 수 있다. 따라서 무기체계의 품질을 효과적으로 향상시키기 위해서는 무기체계의 개발단계에서 발생 가능한 결함들을 미리 식별하여 예방하는 조치가 필요하다.

이런 관점에서 개발단계 SW 품질검증체계는 SW 결함 DB를 활용한 SW 품질개선 체크리스트, 소스코드 구조 분석을 활용하여 무기체계의 개발단계에서 체계적이고 효과적으로 SW 결함을 예방한다. 이런 업무 방식은 개인 역량이나 우연에 따른 품질관리 편차를 줄이고 무기체계 품질을 조직적으로 향상시킨다. 무기체계 소프트웨어의 품질을 지속적으로 향상시키기 위해 SW 결함 DB와 SW 품질개선 체크리스트, 소스코드 구조분석 방법을 더욱 고도화하는 것이 필요할 것으로 생각된다.

### References

- [1] Defense Acquisition Program Administration(DAPA) Manual "Weapon System Software Development and Management Manual", DAPA, Korea, 2022.
- [2] Defense Acquisition Act, Act, Korea, 2022.
- [3] Defense Acquisition Instructions, Instructions, Korea, 2022.
- [4] Defense Acquisition Program Administration(DAPA), Regulations for Supporting the Development of Weapon System Software, Regulation, DAPA, Korea, 2020, pp. 12.
- [5] J. Y. Lee, "Application case of Defect management

system in Defense weapon development process", *Proceedings of Korean Institute of Industrial Engineers Joint Conference*, Gyeongju, Korea, pp.689-708, May 2012.

UCI(KEPA): I410-ECN-0101-2013-530-003197882

- [6] J. W. Oh, "Research on the Strengthen Quality Management of the Analysis of the Causes of Software Engineering Change by System Company", *Journal of DQS*, Vol. 4, No. 2, pp. 2-8, Dec. 2022.
- [7] J. K. Kim, J. H. Yun, J. S. Yu, "Analysis of Engineering Change Cause and Defect Inflow Stage of Weapon System Software", *Journal of the Korea Academia-Industrial cooperation Society*, Vol. 23, No. 3, pp. 131-137, Mar. 2022.  
DOI: <https://doi.org/10.5762/KAIS.2022.23.3.131>
- [8] J. W. Oh, J. K. K.im, J. S. Yu, J. H. Yun, C. H. Song, "Research on DB Construction and Utilization Measure to Analyze the Cause of Weapon System Software Engineering Change and Derive Improvement Plan", *Journal of the Korea Academia-Industrial cooperation Society*, Vol. 22, No. 4, pp. 331-337, Apr. 2021.  
DOI: <https://doi.org/10.5762/KAIS.2021.22.4.331>

윤 경 환(Gyeonghwan Yoon)

[정회원]



- 2013년 2월 : 서울시립대학교 전자전기컴퓨터공학부 (학사)
- 2013년 1월 ~ 2016년 8월 : 현대오트론 연구원
- 2018년 12월 ~ 현재 : 국방기술품질원 연구원

<관심분야>

무기체계 소프트웨어, 소프트웨어 품질관리, 테스트

박 만 춘(Manchun Park)

[정회원]



- 1993년 2월 : 충남대학교 전산학과 (학사)
- 2013년 2월 : 한성대학교 국방경영학 (석사)
- 1993년 4월 ~ 현재 : 국방기술품질원 연구원

<관심분야>

무기체계 소프트웨어, 소프트웨어 품질관리, 테스트

오 진 우(Jin-Woo Oh)

[정회원]



- 2018년 2월 : 경상대학교 정보과학과 (정보과학석사)
- 2014년 1월 ~ 현재 : 국방기술품질원 선임연구원

<관심분야>

무기체계 소프트웨어, 소프트웨어공학

주 진 천(JinChun Ju)

[정회원]



- 1996년 2월 : 포항공과대학교 컴퓨터공학과 (석사)
- 1996년 1월 ~ 2006년 1월 : 국방과학연구소 선임연구원
- 2006년 2월 ~ 현재 : 국방기술품질원 책임연구원

<관심분야>

무기체계 소프트웨어, 소프트웨어 품질관리, 테스트

윤 재 형(Jae-Hyeong Yun)

[정회원]



- 2017년 2월 : 건국대학교 전자공학부 (전자공학석사)
- 2017년 2월 ~ 현재 : 국방기술품질원 연구원

<관심분야>

국방, 무기체계 소프트웨어, 소프트웨어 품질

엄 원 용(Wonyong Eom)

[정회원]



- 2009년 2월 : 충북대학교 전자공학전공
- 2011년 2월 : 한국과학기술연구원 전기및전자공학 (공학석사)
- 2012년 12월 ~ 현재 : 국방기술품질원 선임연구원

<관심분야>

무기체계 소프트웨어, 무기체계 품질관리

김 종 규(Jong-Kyu Kim)

[정회원]



- 2019년 8월 : 동국대학교 정보통신공학과(정보통신공학석사)
- 2019년 12월 ~ 현재 : 국방기술품질원 연구원

<관심분야>

국방, 무기체계 소프트웨어, 소프트웨어공학