

현대 무기체계 소프트웨어 개발 사례 분석을 통한 품질개선 방안에 대한 연구

최태윤*, 이호진
(주)대한항공 무인기사업부

An Investigation into Improving the Quality of Modern Weapon System Software Products: a Case Study

Taeyun Paul Choi*, Hyo-Jin Lee
Korean Air UAS Business Department

Abstract : As modern weapon systems have evolved into embedded software systems, the capability to develop, install, and maintain high-quality software products is becoming increasingly valuable in today's competitive aerospace and defense industry. It is desirable, especially for government-contracted projects whose end-users are military personnel, that specific types of software quality elements are assured early during product development; namely, functionality and usability. Motivated by this need to approach improving software quality from a user's perspective, this paper presents a case study that analyzes system-level software integration testing results for a weapon system indigenously developed in the Republic of Korea. Quantitative data, such as software defect ratios and configuration change rates, are computed to observe meaningful trends with respect to functionality – represented by tests designed to verify the system's electrical signal interfaces – and usability, which is evaluated in terms of how often the initial human-computer interface had to be re-designed. Insights gained from these exercises are subsequently presented as a set of “lessons learned” to take into account when developing new software items for complex weapon systems.

Key Words : Software Testing, Process and Product Quality Assurance, Configuration Management, Integrated Product and Process Development

* 교신저자 : taeychoi@koreanair.com

1. 서론

현대 무기체계의 형태가 하드웨어에 소프트웨어를 탑재하는 내장형 시스템으로 발전하는 과정에서 전체 기능 중 소프트웨어가 차지하는 비중은 점차 증가해 왔다. 미국의 군용항공기 개발사례를 정리한 표 1을 보면, 1960년대 미국 공군의 주력 항공기였던 F-4(3세대 전투기)에서 SW(Software)가 담당하는 기능 비율이 8% (Ferguson, 2001) 정도였던 반면, 5세대 전투기인 F-35의 경우에는 SW로 구현된 기능 비율이 90%(김호준, 2012)에 육박하고 있음을 확인할 수 있다. 그러므로 고품질의 SW 제품을 개발, 설치 및 유지보수 할 수 있는 역량의 확보는 현대 무기체계의 성공적 구현을 위한 선결 조건이라 할 수 있다.

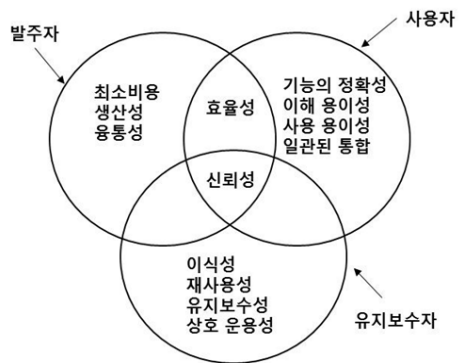
<표 1> 미국 군용항공기 SW 기능비율

| 무기체계 | 도입년도 | SW 기능비율 |
|-------------------|-----------|---------|
| F-4 Phantom | 1960년 | 8% |
| F-111 Aardvak | 1967년 | 20% |
| F-16 Falcon | 1978년 | 45% |
| F-22 Raptor | 2005년 | 80% |
| F-35 Lightning II | 2016년(계획) | 90% |

SW 품질평가에 대한 표준화된 기준을 제시하는 ISO/IEC 9126-1 품질모델(ISO/IEC, 2001)은 SW 제품에 대한 품질특성을 표 2와 같이 6개의 외부특성(External Metrics)과 21개의 내부특성(Internal Metrics)으로 정의한다. 국내 무기체계 개발사업의 경우에는 소요군이라는 내정된 사용자를 대상으로 시제품에 대한 전투적합 판정을 획득하는 것이 사업의 일차적인 목표이기 때문에, 무기체계 개발기관 또는 업체는 6개의 외부특성 중 그림 1(박태순, 2009)의 분류에 따라 기능의 정확성 및 사용 용이성과 같은 사용자 관점의 SW 품질을 중점적으로 확보할 필요가 있다.

<표 2> ISO/IEC 9126-1 SW 제품 품질모델

| 외부특성 | 설명 | 내부특성 |
|-------------------------|---|-----------------------------------|
| 기능성 (Functionality) | 사용자가 요구하는 기능을 충족시키는 정도 | 적합성 보안성 정확성 유연성 상호호환성 |
| 사용성 (Usability) | 사용자로 하여금 쉽게 이해하고, 배우고, 사용하며, 친숙해질 수 있는 능력 | 성숙성 회복성 오류허용성 |
| 신뢰성 (Reliability) | 정확하고 일관성 있게 수행하는 능력 | 이해성 운용성 습득성 |
| 효율성 (Efficiency) | 투입된 자원에 대하여 제공되는 성능의 정도 | 실행효율성 자원효율성 |
| 유지보수성 (Maintainability) | 쉽게 서비스하고, 수리하고, 고칠 수 있는 능력 | 변경성 분석성 안전성 시험성 |
| 이식성 (Portability) | 새로운 환경으로 쉽게 수정할 수 있는 능력 | 일치성 치환성 환경적응성 이식작업성 |



[그림 1] 소프트웨어 품질에 대한 관점 (박태순, 2009)

SW 결함으로 인해 발생하는 비용은 개발 후기 단계로 갈수록 막대하게 증가한다고 알려져 있으며, 최근 F-35 납기 지연 및 비용 증가 사례의 주요 원인 중 하나로 탑재 SW 결함이 지적되었다 (Rosenberg, 2012). 따라서 사업 실패의 위험을 최소화하기 위한 최적의 시나리오는 SW 개발 초기 단계부터 품질에 악영향을 주는 요소들을 조기에

발견하고 제거해 나아갈 수 있도록 SW 개발 프로세스를 개선하는 것이다. 이와 같이 SW 제품 품질을 향상시키고자 하는 현대적 품질관리 패러다임은 해외에서 도입되어 국내 SW 공학/품질관리 분야에서도 주목 받고 있는 프로세스 품질모델(Chrissis, 2011)의 기본 개념인 동시에 국내에서 선행되고 있는 SW 품질보증 활동의 기본 방향이기도 하다 (방위사업청, 2011).

본 논문은 사용자 관점의 무기체계 SW 제품 품질 향상을 목적으로, 국방 SW 개발 프로세스 개선 방향을 모색하고자 하는 동기에서 출발하였다. 우선, SW 중심의 복합체계 특성을 가진 무기체계의 전기적 신호 인터페이스설계와 인간 컴퓨터 인터페이스를 검증한 사례를 바탕으로 SW 형상항목별로 결함분포도와 형상변경율을 도출하였다. 또한 체계 수준에서의 SW 통합시험 수행 시 작성한 소프트웨어 문제점 보고서들을 원인별로 분석하고 분류하였다. 이러한 정량적 분석 결과를 토대로 개선이 필요한 SW 개발 프로세스 요소들을 교훈으로 도출하였고, 정리된 결과를 본 논문을 통해 제시하고자 한다.

2. 사례 연구

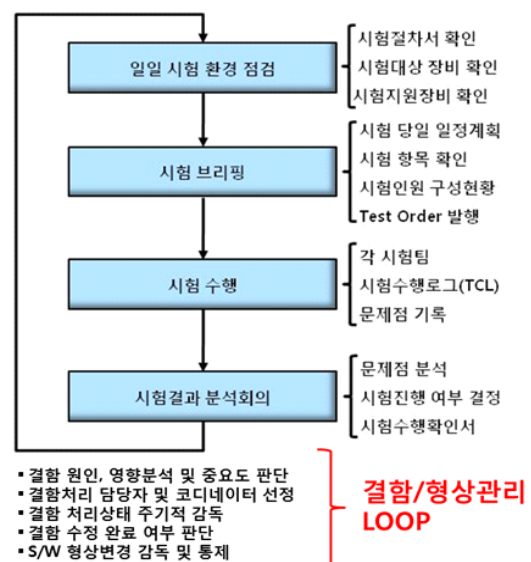
본 논문에 활용된 기초 데이터는 본 연구진이 참여한 무기체계 통합시험의 결과 일부를 발췌한 것이다. 이 통합시험은 시험 대상 무기체계의 체계연동 및 기능적 요구사항을 검증하는 것으로, 시험에 필요한 모든 SW 형상항목, 즉 CSCI(Computer Software Configuration Item)의 SW 통합시험이 종료된 이후에 수행되었다. 시험 대상 CSCI 중 그래픽 사용자 인터페이스가 포함된 SW의 경우에는 사용성 검증을 통해 HCI(Human-Computer Interface)에 대한 개선 사항을 도출하는 절차가 인터페이스/기능 시험과 병행되었다.

2.1 통합시험 개요

모든 시험은 무기체계의 운용을 모의할 수 있는

SIL(System Integration Lab)에서 수행되었다. SIL 환경은 실제 무기체계에 탑재되는 현장교환품목과 동일한 하드웨어에 CSCI를 탑재한 실장비 부분, 실장비는 아니지만 해당 장비의 전기적/SW 인터페이스를 모사해 주는 모의장비 부분, 그리고 실/모의장비간 연동의 실시간 확인을 가능하게 해주는 모니터링장비 부분으로 구성되었다. 실장비 부분은 실제 무기체계의 전원계통이 출력해 주는 것과 유사한 특성의 직류전원을 공급해 주도록 설계되었으며, 모의장비 부분과 모니터링장비 부분은 국내에서 개발된 장비와 이더넷 통신 및 MIL-STD-1553B (DOD, 1978) 통신 등을 모니터링해 주는 상용 장비를 포함하도록 설계되었다.

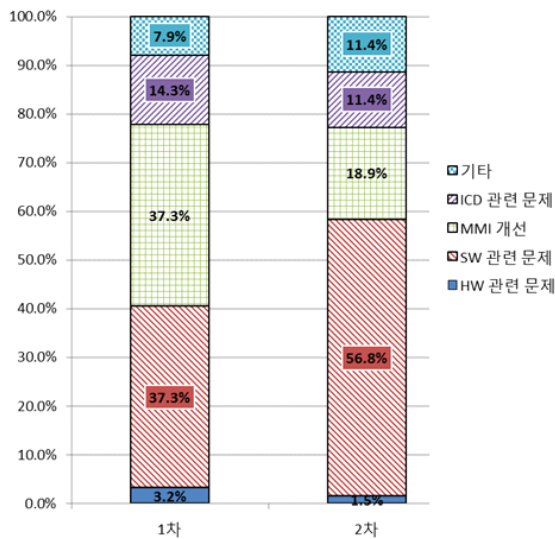
통합시험은 사전에 작성된 시험절차서에 따라 진행되었고, 해당 문서는 SW 규격화 대상 자료 중 하나인 소프트웨어 시험절차서(방위사업청, 2011)의 구성요소를 대부분 포함하고 있었다. 각 시험 항목에는 시험의 성격에 부합하는 고유 시험식별자가 부여되었으며, 각 시험 항목과 체계 요구사항간의 관계를 나타내주는 요구사항 추적표가 시험절차서 및 세부 시험수행 로그에 명시되었다. 시험 중 발견된 결함을 기록하고, 사안의 중요성에 따라 재시험을 수행하는 절차와 SW 형상관리 절차 또한 그림 2와 같이 적용되었다.



[그림 2] 통합시험 결함 처리 및 형상관리 절차

2.2 통합시험 결과 분석

그림 3은 통합시험 수행간에 SPR(Software Problem Report)로 기록된 결함들의 유형별 발생 분포 추세를 보여주며, 표 3은 결함 분류에 사용된 판단기준을 나타낸다. 시험 차수는 체계연동 검증의 기준이 된 인터페이스통제문서의 버전에 따라 구분하였다. 즉, 2차 시험 때 사용한 인터페이스통제문서는 1차 시험의 기준이 된 문서를 개정한 것이다.



[그림 3] 통합시험 SW 결함 유형별 발생 추이

<표 3> SW 결함 분류 기준

| 구분 | 소프트웨어 결함 유형 |
|------------|---|
| HW 문제점 | 소프트웨어 변경 없이 하드웨어 수리를 통해 해결된 결함 |
| SW 문제점 | 소프트웨어 코딩, 설계변경 또는 기능 추가로 해결된 결함 |
| HCI 개선 | HCI의 사용성 향상을 위한 SW 수정 |
| ICD 관련 문제점 | ICD를 잘못 구현하였거나 원천적인 ICD 오류를 수정하기 위한 소프트웨어 변경 |
| 기타 | 간헐적으로 발생하거나 원인규명이 어려워 소프트웨어 변경 없이 일정기간 동안 관찰 후 검토회의를 통해 종결시킨 결함 |

결함분포 데이터가 보여주는 첫 번째 거시적인 경향은 시험 차수가 진행되면서 하나의 결함 유형이 전체 결함에서 차지하는 비율이 감소한 만큼 다른 유형의 결함이 증가하여 결과적으로 결함 감소에 대한 효과가 상쇄된 점이다. 우선, 2차 시험에서 HCI 개선을 요구한 SPR의 비율이 1차 시험과 비교해서 18.4% 감소(37.3% → 18.9%)한 값과 비슷하게 SW 문제점이 전체 결함에서 차지하는 비중이 19.5% 증가(37.3% → 56.8%)하였다. 인터페이스 통제문서, 즉 ICD(Interface Control Document) 관련 문제점 비율의 경우에는 1차 시험 결과와 비교하여 2차 시험에서 2.9%의 결함 감소(14.3% → 11.4%)를 달성했으나 '기타'로 분류되는 유형의 결함이 3.4% 증가(7.9% → 11.4%)하였다.

두 번째로 주목할 만한 경향은 결함 발생 당시에 SPR로 처리되었으나, 이후 원인 분석을 통해 하드웨어 수리로 조치가 완료된 결함이 존재하였다는 점이다. 통합시험이 단위장비 수준에서의 수락시험 또는 인증시험 종료 후에 수행된 것을 감안했을 때, 1/2차 시험에서 보고된 HW(Hardware) 결함비율은 타 결함유형에 비해 수치상으로는 낮아 보이지만, HW 신뢰성 보증 담당자에게는 다소 우려스러운 결과일 수 있다.

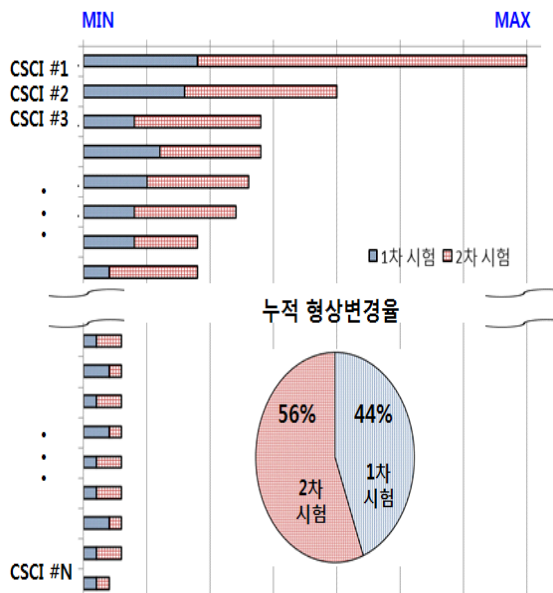
셋째, ICD 개정과 함께 ICD 관련 문제점 및 HCI 개선 비율이 하락한 것은 고무적인 반면, SW 문제점 비율이 1차 시험의 37.3%에서 2차 시험 때 56.8%로 크게 상승한 것은 바람직하지 않은 현상이다. (2차 시험 시 검출된 SW 문제점 개수는 1차 시험 때 발견된 SW 문제점 개수보다 1.6배 높았다.) 이는 표 4의 세부 유형별 결함분포 결과가 보여주듯이, ICD 개정은 인터페이스 설계 오류와 ICD 자체의 품질을 향상시키는데 분명한 기여를 하였지만 이로 인한 SW 코딩 오류와 설계변경의 증가를 유도한 것으로 보인다.

이러한 경향은 통합시험 기간 중 축적한 SW 형상변경 데이터의 분석을 통해서도 관찰할 수 있다. 그림 4 내부의 원 도표가 나타내는 것과 같이 1차 시험 때 누적된 CSCI 형상변경의 횟수보다 2차 시

험 시기에 발생한 형상변경 횟수의 합계가 더 큰 것으로 파악되었다. 특히 7개의 CSCI에 대한 형상변경 횟수가 전체 변경 건수의 과반 이상에 기여하였고, 증가율을 해당 CSCI들에 한정하여 계산할 경우 1차 시험과 비교한 2차 시험에서의 형상변경 증가율이 20%(40% → 60%)가 되어 그림 4에 나타난 평균 증가율인 12%(44% → 56%)보다 높아진다.

<표 4> 통합시험 SW 결함 상세유형별 발생율

| 구분 | 1차 시험 | 2차 시험 |
|-------------|-------|-------|
| HW 문제점 | 3.2% | 1.5% |
| SW 문제점 | 37.3% | 56.8% |
| - SW 코딩 오류 | 12.7% | 16.7% |
| - SW 설계 변경 | 22.2% | 38.6% |
| - SW 기능 추가 | 2.4% | 1.5% |
| HCI 개선 | 37.3% | 18.9% |
| ICD 관련 문제점 | 14.3% | 11.4% |
| - ICD 설계 오류 | 12.7% | 8.4% |
| - ICD 미적용 | 1.6% | 3.0% |
| 기타 | 7.9% | 11.4% |



[그림 4] 통합시험 SW 형상변경을 추이

3. 교훈

사용자가 SW 제품의 품질이 높다고 느낄 수 있으려면, 해당 SW 제품이 배우기 쉽고, 조작이 간단하며, 다양한 기능과 우수한 성능을 갖추었다는 인상을 받아야 한다(박태순, 2009). 국방 무기체계 SW 개발자가 최종 사용자인 소요군의 만족도를 높이기 위해서 여러 접근 방법을 고려할 수 있겠으나, 본 논문에서는 그림 3에 분류된 유형의 결함을 최소화함으로써 표 2에 나타나 있는 기능성 및 사용성을 향상시킬 수 있는 방안을 교훈으로 도출하였다. 그 결과를 우선순위가 높은 순서대로 정리하면 다음과 같다.

3.1 선제적 사용자 인터페이스 설계

사람이 컴퓨터의 터치스크린, 키보드, 마우스 등의 입력장치를 조작하여 통제명령을 주고, 이에 대응하는 상태정보를 모니터와 같은 출력장치를 통해 인지한 후 다시 입력장치를 통해 반응하는 기능을 구현한 SW에서 사용자 인터페이스 설계는 매우 중요하다. 특히 무기체계 SW 설계는 사용자의 실수나 오동작과 같은 휴먼 에러의 발생 가능성에 대비하여 안전장치의 다중화 설계를 고려해야 하며, 개발자에게 생소하더라도 사용자에게는 익숙한 형태의 인터페이스를 제공해야 한다. 이처럼 완성도가 높은 사용자 인터페이스를 설계하기 위해서는 개발자가 HCI 설계 모형(김행근, 2006)에 따라 사용자 모델(User Model), 시스템 인식(System Perception) 등과 같은 최종 사용자를 고려한 설계를 선제적으로 수행할 필요가 있으며, 개발 초반부터 사용자 그룹의 적극적인 참여와 피드백을 유도할 수 있어야 한다. 이를 위해서는 SW의 요구분석 단계에서부터 기능/성능 요구사항과 함께 사용자 인터페이스에 대한 요구사항을 명세화 하고, SW 요구사항 검토회의와 같은 공식 의사결정 회의체를 통하여 개발자 및 고객(사용자) 상호가 그 내용에 대해 검토/합의하는 과정을 수행해야 한다. 그리고 이러한 합동검토 체제는 해당 SW의 기본설계, 상세설계, 구현/단

위시험, 통합시험, 그리고 기술개발시험평가와 운용 시험평가 단계까지 유지하는 것이 바람직하다.

3.2 통합팀 기반의 ICD 워킹그룹 운영

체계연동 설계가 확정된 이후에 수행된 통합시험에서 ICD 관련 문제점이 발생한 주 원인은 연동설계 내용이 원천적으로 잘못되었기 때문인 것으로 분석되었다. 표 4에서 볼 수 있듯이, SW 개발자가 ICD에 기술된 사항을 잘못 구현한 경우보다 통합시험 과정에서의 연동설계상 오류인 경우가 상대적으로 높았다. 이러한 유형의 문제점을 방지하기 위해서는 체계기능/연동 설계를 수행하는 시스템엔지니어들과 분야별로 할당된 기능을 구현하는 SW 엔지니어들 사이에 충분한 의사소통과 협업을 유도하는 업무환경이 조성되어야 하겠다. 이러한 맥락에서 본 연구진은 ICD의 작성, 검토 및 배포를 전담하는 워킹그룹 형태의 조직이 무기체계 개발사업의 상세설계검토회의 시점에 구성되는 것이 적절하다고 본다. 아울러 해당 조직에는 IPPD(Integrated Product and Process Development) 기법의 통합팀 개념을 도입하여, 소수의 분야별 전문가들과 체계연동규격을 개발한 시스템엔지니어가 하나의 유기적이고 역동적인 팀으로써 ICD를 작성하고 유지해 나갈 수 있는 여건을 마련해야 하겠다. 잘못 설계된 인터페이스 내용이 체계에서 부체계로 할당되고, 결과적으로 SW에서 결함처리를 하는 것은 계속해서 소를 잃으면서 그 때마다 외양간을 고치는 것과 크게 다를 바가 없다.

3.3 소스코드 변경관리 전문도구 도입

표 4에 'SW 코딩 오류'로 기록된 유형의 결함 중 상당수가 여러 명의 SW 개발자가 소스코드 변경관리 전문도구를 사용하지 않고 각기 다른 시점에 수정한 소스코드를 급하게 배포하는 과정에서 발생한 것으로 분석되었다. 사업의 특성에 적합한 전문도구의 도입은 개발자의 휴먼 에러로 인한 실패비용과 개발관리에 필요한 인력 소모를 감소시켜 줄 수 있다. 다행히 최근에는 오픈소스로 획득할 수 있는

SW 형상관리 도구 중에서도 주목할 만한 제품이 많아지는 추세이다(Wikipedia, 2012).

3.4 소프트웨어 신뢰성 확보활동 강화

본 연구 사례에서는 무기체계의 운용 중에 간헐적으로 발생하고 블랙박스(기능) 시험을 통해서 원인규명이 어려운 유형의 SW 결함을 표 3에 '기타'로 분류하였다. 이러한 유형의 SW 결함을 줄이기 위해서는 정부 지침(방위사업청, 2011)에 따른 코딩규칙 준수 및 자동화 도구를 통한 정적/동적시험 수행이 이상적이라 할 수 있겠다. 비용 또는 일정상의 제약으로 인해 전문도구 기반의 SW 시험 인프라를 구축하지 못하더라도, 기존 인력을 활용해서 개발 중인 SW 제품의 잠재적 결함을 최소화할 수 있는 방법은 여러 가지가 있다. 예를 들어, 프로덕트 및 프로세스 품질보증 활동의 일환으로 워크스루(Walk-through) 또는 인스펙션(Inspection)과 같은 동료검토 활동을 소스코드 및 산출물에 대해서 시행을 할 수 있다. 또한 주기적인 감리(Audit)를 통해 SW 개발자들의 프로세스 이행 실태를 점검함으로써 내부적으로 품질에 대한 경각심을 고취시킬 수 있다. 그러므로 국내 무기체계 개발기관에서는 SW 시험 도구 확보 및 관련 인력의 양성에 대한 장기적인 전략을 수립하고, 이와 함께 전문도구의 사용 없이 자체적으로 수행할 수 있는 다양한 신뢰성 확보 기법의 적용을 고려할 필요가 있다.

4. 결 론

본 논문에서는 국내에서 진행된 무기체계 통합시험 사례를 연구하여 국방 SW 개발 과정에서 SW 제품의 기능성 및 사용성 향상에 긍정적인 영향을 미칠 수 있는 교훈을 도출하였다. 이 중 대표적인 방안으로 그래픽 사용자 인터페이스 형태의 사용자 인터페이스 설계에 대해 SW 요구사항명세 검토 단계에서부터 지속적으로 사용자 의견을 수렴하고, 사용자와 개발자가 함께 상세설계 내용을 검토할 것

을 제안하였다. 또한 IPPD 기법의 핵심 중 하나인 통합팀의 개념을 ICD 워킹그룹에 적용하여 시스템 엔지니어와 SW 개발자간 의사소통을 보다 원활하게 해 주는 동시에 원천적으로 잘못된 연동설계 내용이 SW 결함으로 기록되는 악순환을 방지할 필요가 있음을 피력하였다. 형상관리 활동 및 신뢰성 시험은 정부에서 발행한 최신 지침에 정의된 프로세스를 따르되, 각 도메인별로 특화된 자동화 도구의 사용을 통해 휴먼 에러로 인한 코딩/디버깅/빌드 변경 문제점과 실행시간 오류와 같이 실제 체계 운용에 치명적인 문제점의 발생 가능성을 최소화할 것을 권고한다. 최근 방위사업청에서 국방 SW 개발사업 참여업체가 지식경제부의 SW 프로세스 품질인증을 획득할 경우에는 이에 대한 비용 지원과 함께 가산점을 부여하겠다고 선언한 만큼(정용철, 2012), 향후 본 논문과 유사한 주제의 연구가 지속적으로 수행되어 국방 무기체계 SW 품질개선에 기여할 수 있는 우수 사례들이 더 많이 발굴되길 바란다.

참 고 문 헌

1. J., Ferguson. Crouching dragon, hidden software: software in DoD weapon systems. IEEE Software, 18(4), 105-107, 2001
2. 김호준. 방사청, 첨단무기 S/W 개발에 430억원 투입. 연합뉴스, 2012년 9월 9일
3. ISO/IEC. ISO/IEC 9126-1 : Software Engineering - Product Quality - Part 1 : Quality Model. ISO Copyright Office, 2001
4. 박태순. TopSpot 소프트웨어 공학. 도서출판 탑스팟, 2009
5. Z., Rosenberg. Senior F-35 official warns on software breakdowns, relationship crisis. Flightglobal, September 17, 2012
6. M.B., Chrissis, M., Konrad, and S. Sandy. CMMI : Guidelines for Process Integration and Product Improvement. 3rd Edition,

Boston: Addison-Wesley, 2011

7. 방위사업청. 방위사업청 지침 제2011-26호, 무기체계 소프트웨어 개발 및 관리 지침. 2011년 8월 26일
8. DOD. MIL-STD-1553B : Digital Time Division Command/Response Multiplex Data Bus. Department of Defense, 1978
9. 김행곤. 임베디드 소프트웨어 공학. 도서출판 그린, 2006
10. Wikipedia. Comparison of open source configuration management software. September 19, 2012 http://en.wikipedia.org/wiki/Comparison_of_open_source_configuration_management_software
11. 정용철. 국방 SW 개발사업 'SP 인증제' 도입. 디지털타임스, 2012년 9월 25일