

## 개발단계 무기체계 소프트웨어의 품질관리 효과분석

윤경환  
국방기술품질원

### The Analysis of Software Quality Management of Weapon System in Development Phase

Gyeonghwan Yoon  
Defense Agency for Technology and Quality

**요약** 본 연구에서는 개발단계와 최초 양산단계에서 무기체계 소프트웨어의 산출물 오류 개수를 종합적으로 분석한다. 그동안 무기체계 소프트웨어의 개발단계 품질관리 방안을 제시한 연구는 있었지만 실제 무기체계 개발사례를 활용한 품질관리의 효과성 분석은 부족했다. 이번 연구에서는 연도별 무기체계 소프트웨어 품질관리 지원사업의 변화, 산출물 오류 개수의 변화, CSCI (Computer Software Configuration Item) 기준의 산출물의 오류 개수 등을 포함한다. 그리고 개발단계의 산출물 오류 개수와 후속단계인 최초 양산단계의 산출물 오류 개수를 비교하고 분석하여 개발단계 무기체계 소프트웨어의 품질관리 효과성을 파악했다. 또 최초 양산단계의 산출물 오류 분류 방식을 활용하여 개발단계의 산출물 오류도 같은 방식으로 분류한 뒤 비교했다. 그래서 개발단계 품질관리 지원이 산출물의 오류 분포 방식에 어떤 영향을 주었는지 분석하였다. 산출물 오류 감소는 연구개발주관기관의 무기체계 소프트웨어 개발 프로세스 이해도 향상, 프로세스 준수를 향상으로 이어져 최종 산출물인 소프트웨어 품질 역시 향상될 것이다. 이번 연구를 통해 개발단계 품질관리 지원 효과를 확인하면 품질관리 지원이 개발단계로 더욱 확대되는 것에 기여할 수 있을 것이다.

**Abstract** This study examined the output errors of the software development process in the development phase and initial production phase for a weapon system. A few studies provided software quality management suggestions for weapon systems in the development phase, but the analyses of their applied suggestions for weapon system development cases are insufficient. This research includes the changes in software quality management cases, output error changes, output errors for each CSCI (Computer Software Configuration Item). The effects of software quality management were confirmed by comparing the output errors in the development phase with those in the initial production phase. In addition, the output errors in the development phase were classified using the error classification method used in the initial production phase and compared output errors in the development phase with those in the initial production phase. The effect of software quality management in the development phase on output error distribution was analyzed. A decrease in output error means a better understanding of the software development process in weapon systems by software engineers and compliance with the software development process. This will improve the quality of the final output, which is software in a weapon system. This study could expand the scope of software quality management in weapon systems more from the initial production phase to the development phase.

**Keywords** : Software Quality, Software Quality Management, Development Phase, Initial Production Phase, Software of Weapon System

---

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

email: sunshine@daq.re.kr

Received May 14, 2021

Accepted October 1, 2021

Revised July 14, 2021

Published October 31, 2021

## 1. 서론

기술이 발전하고 고도화됨에 따라 하드웨어 부품과 소자가 소형화되고 집약되었으며 기존보다 더 많은 하드웨어가 서로 연결되어 동작하게 되었다. 군에 도입되는 무기체계 하드웨어 역시 복잡성이 증대되었고 이를 제어하는 소프트웨어의 총 라인수 또한 증가하였다. 이에 따라 결합 없이 동작해야 하는 무기체계의 소프트웨어 품질 요구사항 또한 높아졌다. 무기체계 소프트웨어의 품질관리 범위도 양산 및 운용유지 단계에서 개발단계 품질관리로 점차 확대되었다.

본 연구에서는 무기체계 소프트웨어의 개발단계 품질관리 지원이 얼마나 효율적이었는지 최초 양산단계 품질관리 지원과 비교하여 분석한다. 그동안 개발단계의 소프트웨어 품질향상을 위해 국방과학연구소, 국방기술품질원에서 신뢰성시험 방법 연구[1,2], 신뢰성 시험 현황 연구[3], 개발단계 품질관리 기술지원의 개선방안 제안[4] 등의 연구는 있었다. 그렇지만 개발단계의 무기체계 소프트웨어 품질관리 지원 실 사례를 활용하여 소프트웨어 기술지원의 효과성을 파악한 연구는 부족했다.

본 연구를 통해 개발단계 품질관리 효과를 확인하면 품질관리 지원이 개발단계로 더욱 확대되는 것에 기여할 수 있을 것이다. 개발단계 품질관리를 통해 양산운영 단계에서 발생하는 소프트웨어 결함을 미리 제거하거나 최소화하여 결함수정 비용 절감, 무기체계 획득기간 단축, 조기 전력화 등을 기대할 수 있다.

또한 소프트웨어 개발 초기단계부터 품질을 관리하면서 연구개발주관기관의 소프트웨어 개발 프로세스 준수 여부도 효율적으로 점검하고 지원할 수 있을 것이다. 결과적으로 무기체계에 내장된 소프트웨어 품질이 향상되어 군이 신뢰성 높은 무기체계를 획득할 수 있을 것이다.

## 2. 이론적 배경

### 2.1 개발단계 소프트웨어의 품질관리

무기체계는 선행연구, 탐색개발, 체계개발, 양산, 운용/폐기 단계에 따라 관리된다. 무기체계의 체계개발 단계는 체계공학 프로세스의 요구사항 정의, 설계, 구현, 통합 검증, 초도생산 등 체계공학 프로세스에서 정의된 단계와 연계된 형태이다. 무기체계 체계개발단계는 「무기체계 소프트웨어 개발 및 관리 매뉴얼」에 따라 세부 단계별 업무와 산출물이 정의되어 있다[4].

이러한 프로세스를 따라 개발된 무기체계는 생산, 양산단계에서 발생한 산출물의 결합 수정비용이 개발단계에서 발생한 결합 수정비용보다 크다. 따라서 결합 수정비용 절감, 일정 단축, 우수 품질 확보를 위해 무기체계 개발 프로세스에서도 개발단계 품질관리 방안이 강조되고 있다. 개발 프로세스의 품질 개선방안, 제품의 품질 평가방안, 개발단계별 소프트웨어 신뢰도 평가방안 등이 연구로 제시되기도 한다[5]. 또한 연구개발주관기관의 관점에서 군수품의 개발 품질보증 프로세스 구축과 적용 사례가 연구되기도 했다[6]. 그렇지만 개발단계 품질관리 방안의 효과분석은 제시되지 않는 한계점이 있었다.

### 2.2 최초 양산단계 소프트웨어의 품질관리

개발단계 이후 최초로 양산된 물량을 대상으로 소프트웨어 품질관리 지원활동이 수행된다. 개발결과 완성된 규격에 따라 개발품질의 양산 실현가능성 및 규격과의 적합성 평가가 목적이다.

기존의 무기체계 최초양산품 품질관리는 하드웨어 중심으로 수행되었다. 그렇지만 소프트웨어 품질관리의 경우 표준 프로세스가 없이 각각의 무기체계 사업별로 이뤄지고 있었다. 이를 개선하기 위해 국방기술품질원에서 최초양산단계의 소프트웨어 품질보증 프로세스가 연구되기도 했다[7]. 컴퓨터 파일의 완전성, 소프트웨어 기술문서의 적합성, 소프트웨어 기술문서와 국방규격(KDS: Korean Defense Specification, 이하 KDS)의 연계성 등을 점검하는 품질보증 프로세스가 그것이다. 최초 양산단계에서 품질보증 방식을 체계적으로 규정하려 시도한 것에서 의의가 있다. 그렇지만 무기체계의 개발 및 규격화 완료 이후의 품질보증 연구이기 때문에 효율적인 품질관리를 위해 개발단계 와 연계된 연구가 필요하다.

### 2.3 개발 프로세스와 최종 성과물

개발 프로세스의 성숙도가 높고 제품의 개발기간 동안 프로세스가 올바르게 준수되면 최종 성과물의 품질 역시 높은 것으로 연구되고 있다.

소프트웨어 제품의 품질을 평가하는 국제표준 ISO/IEC 25023에서도 프로세스의 품질이 소프트웨어의 내부특성, 외부특성에 영향을 주고 최종적으로 사용 품질에 영향을 주는 것으로 본다[8]. 소프트웨어의 내부특성이란 정적 시험으로 검증되고 소스코드의 설계, 구현 단계에서 측정되는 품질을 말하며 외부특성이란 소스코드가 실행되는 테스트링이나 운영을 통해 측정되는 품질을 말한다.

또한 체계적인 프로젝트 관리가 품질을 포함한 제품개발 성과에 긍정적인 영향을 주는 것을 확인하는 연구[9], R&D 프로세스의 성숙도와 제품개발 성과의 영향성 연구도 확인할 수 있었다[10]. 결국 개발 프로세스를 올바르게 이해하고 준수하는 것은 높은 품질의 최종 성과를 얻는 것을 의미한다.

### 3. 이론 및 접근방법

#### 3.1 소프트웨어 품질관리 현황

무기체계 소프트웨어는 2.1 장에서 제시된 것처럼 체계공학 프로세스와 연계된 형태의 소프트웨어 개발 프로세스에 따라 구현된다. 「무기체계 소프트웨어 개발 및 관리 매뉴얼」에 규정된 소프트웨어의 주요 개발 단계는 소프트웨어의 요구사항 분석, 소프트웨어 구조설계, 소프트웨어 상세설계, 소프트웨어 구현, 소프트웨어 통합 및 시험 등으로 이루어져 있다[4]. 각 단계를 진행할 때마다 소프트웨어 개발계획서, 소프트웨어 요구사항 명세서, 소프트웨어 설계 기술서, 소스코드, 단위 및 통합 시험 결과, 소프트웨어 산출물 명세서 등의 필수 산출물이 작성된다. 소프트웨어 기술지원 기관인 국방기술품질원에서는 이러한 산출물을 검토하여 오류를 검출하고 교정한다.

#### 3.2 품질관리 효과분석 방안

본 연구에서는 개발단계와 최초 양산단계에 모두 참여한 사업의 각 단계별 산출물 오류 개수 변화를 파악하여 품질관리 지원효과를 분석한다. 산출물의 오류 개수가 낮아지면 해당 사업은 무기체계 개발 프로세스의 이해도, 준수율이 높은 것이고 최종 산출물인 소프트웨어의 품질도 높은 것으로 간주한다.

또한 개발단계의 산출물 오류는 [7]의 연구에서 제시한 최초 양산단계의 산출물 오류 유형으로 분류하였다. 개발단계 산출물 오류 유형과 최초 양산단계의 산출물 오류를 동일한 유형으로 분류하고 비교하여 개발단계 품질관리 지원이 산출물의 오류 형태에 미치는 영향도 파악할 것이다. 산출물 오류 유형은 Table 1과 같이 4가지 항목으로 분류할 수 있다. 소프트웨어 기술문서의 내용 적합성을 확인하는 항목, 컴퓨터 파일 정보의 정확성을 점검하는 항목, KDS, 품질보증요구서(QAR: Quality Assurance Requirement, 이하 QAR), 도면과 소프트웨어 기술문서의 연계성을 검토하는 항목, 국방표준종합

정보시스템(KDSIS: Korean Defense Standard Information System, 이하 KDSIS)에 소프트웨어 관련 자료들이 적합하게 등록되었는지 확인하는 항목 등이다.

Table 1. Check category for software quality measures

Category	Evaluation standards
SW technical documents	<ul style="list-style-type: none"> <li>○ Appropriateness of contents, procedure, method</li> <li>○ Standard format should be followed</li> </ul>
Computer files	<ul style="list-style-type: none"> <li>○ Appropriateness of computer file information</li> <li>○ Existence of all related computer files</li> </ul>
Document Connectivity	<ul style="list-style-type: none"> <li>○ Connectivity among KDS, Drawing, QAR, SW technical documents</li> </ul>
KDSIS Connectivity	<ul style="list-style-type: none"> <li>○ All software resources should be appropriately registered in KDSIS</li> </ul>

산출물 오류 개수는 CSCI (Computer Software Configuration Item, 이하 CSCI) 단위로 표현이 가능하다. CSCI란 형상관리를 위한 소프트웨어 집합체이며 소프트웨어 품질관리 단위이다. 복잡한 무기체계의 경우 많은 수의 CSCI로 구성되고 산출물 오류 개수가 많으며 작은 무기체계의 경우 적은 수의 CSCI로 구성되어 산출물 오류 개수 역시 적다. CSCI 단위로 산출물 오류 개수를 파악하면 무기체계의 복잡도 영향을 줄여서 품질수준을 더 합리적으로 비교할 수 있을 것이다.

#### 3.3 주요 연구절차

본 논문에서는 먼저 SW 품질관리 현황을 분석한다. 그리고 무기체계 소프트웨어의 품질관리 효과를 분석하기 위한 방안을 제시한다. 그 이후 25개 사업을 대상으로 소프트웨어 품질관리 효과를 분석하기 위한 방안을 적용할 것이다. 마지막으로 소프트웨어의 품질관리 개선 효과를 종합적으로 분석한다. 본 논문의 주요 연구절차 개념도는 Fig. 1과 같다.

## 4. 품질관리 지원현황

#### 4.1 연도별 품질관리 지원현황

2016년부터 2020년까지 품질관리 지원 사업이 꾸준히 확대되었다. 또한 품질관리의 범위도 최초 양산단계 지원에서 개발단계 지원으로 점차 확대되고 있다. 25개의 소프트웨어 품질관리 지원 사업은 개발단계와 최초

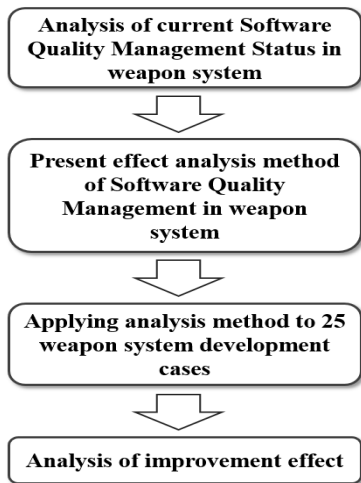


Fig. 1. The schematic diagram of research process

양산단계를 지원한 사업(이하 A) 9개, 최초 양산단계만 지원한 사업(이하 B) 16개로 구분할 수 있다. 개발단계를 지원한 사업은 모두 최초 양산단계를 지원했으며 사업의 분류 개념은 Fig. 2와 같다.

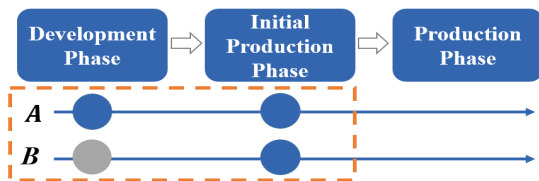


Fig. 2. Comparison of two classification of Weapon System development cases

품질관리 지원 활동은 산출물의 점검 활동을 포함한다. 또한 점검 활동의 결과물인 산출물의 오류 개수를 통해 개발단계 품질관리 효과를 분석할 수 있다. 연도별 A와 B의 산출물 오류 개수는 Fig. 3과 같다.

2019년 무기체계 소프트웨어 품질관리 지원 사업의 수가 9개로 증가하면서 산출물의 오류 개수가 가장 크게 증가했다. 총 산출물의 오류 개수 역시 가장 많았다.

#### 4.2 CSCI 단위당 산출물 오류의 개수

하나의 무기체계 개발 사업은 여러 개의 CSCI로 구성되어 있으며 소프트웨어 품질관리 단위인 CSCI 기준으로 산출물의 오류 개수를 파악하는 것이 적절하다. 5년 동안의 A와 B의 총 CSCI와 산출물 오류 개수는 Fig. 4와 같다.

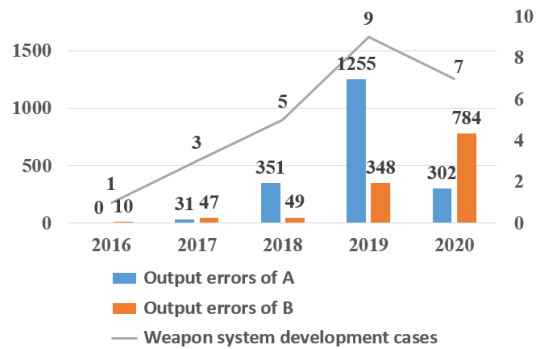


Fig. 3. Number of output errors for A, B and Weapon system development cases 2016-2020

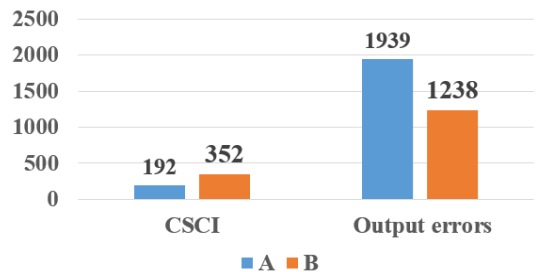


Fig. 4. Number of output errors and CSCI for A, B

소프트웨어 품질관리를 지원한 총 CSCI는 A와 B를 합한 544개 (192개 + 352개) 이고, 총 산출물 오류 개수는 3,177개 (1,939개 + 1,238개) 이다. 따라서 지난 5년 동안 하나의 CSCI 당 산출물 오류 개수는 약 5.84개 인 것을 확인할 수 있다.

#### 4.3 단위 사업별 산출물 오류 분포

산출물 오류는 3.1절에서 제시한 4가지 유형으로 구

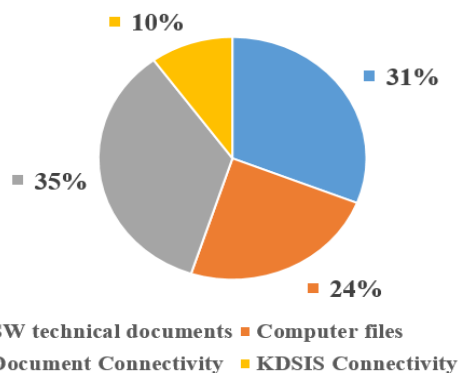


Fig. 5. Classification of output error for unit Weapon System development case

분 가능하다. 총 25개 사업의 산출물 오류를 4가지 유형에 따라 구분했을 경우 평균 1개 무기체계 소프트웨어 개발사업의 오류 분포는 Fig. 5와 같다.

가장 많은 형태의 오류는 소프트웨어 기술문서, QAR, 도면, KDS 등 각 문서의 연결성을 확인하는 문서 연계성 오류이며 전체의 35%에 해당했다. 이후로 소프트웨어 기술문서 오류(31%), 컴퓨터 파일 오류(24%), KDSIS와의 연계성 오류(10%) 순이었다.

## 5. 개발단계 품질관리 지원효과

### 5.1 개발단계 품질지원의 종합 효과

개발단계와 최초 양산단계에서 점검하는 산출물은 차이가 있다. 개발 단계의 경우 소프트웨어 개발 프로세스 전체의 산출물을 검토한다. 그래서 소프트웨어 기술문서를 항목을 검토할 경우 소프트웨어 요구사항 명세서, 소프트웨어 설계 기술서, 소프트웨어 통합시험 절차서, 소프트웨어 시험 결과서, 소프트웨어 산출물 명세서(Software Product Specification, SPS) 등을 확인한다.

그렇지만 최초 양산단계의 경우 소프트웨어의 개발이 완료되었기 때문에 SPS를 주로 검토하게 된다. 개발단계와 최초 양산단계의 공통 점검 항목인 SPS의 오류 개수를 활용하면 개발단계 지원사업의 효과성을 파악할 수 있다. 개발단계를 지원한 사업은 모두 최초 양산단계를 지원했기 때문에 개발단계 지원 사업 9개가 분석 대상이다. Fig. 6은 개발단계와 최초 양산단계의 총 SPS 오류 개수와 CSCI 단위당 산출물 오류 개수를 나타낸다. 개발 단계를 지원한 총 CSCI는 Fig. 4와 같이 192개이다.

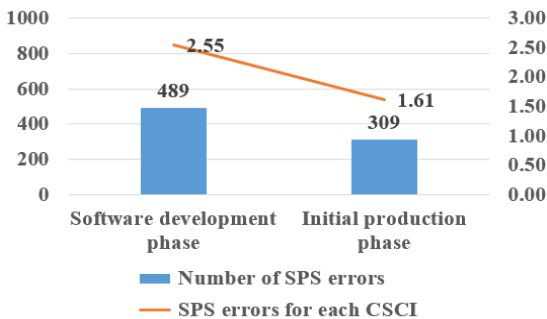


Fig. 6. SPS errors and SPS errors for each CSCI in development phase and initial production phase

개발단계 품질관리 지원 결과 SPS의 오류 개수는 489개이고 최초 양산단계의 SPS 오류 개수는 309개로 36.8% 감소했다. CSCI 당 SPS 산출물의 오류 개수 역시 2.55에서 1.61로 감소했다.

### 5.2 개발단계 품질관리 지원의 사업별 효과

9개의 개발단계 품질관리 지원 사업 중 개발단계 산출물 오류 개수가 가장 많은 3개 사업(이하 C, D, E)을 선정했다. 각 사업의 개발단계와 최초 양산단계의 산출물 오류 개수는 Fig. 7과 같다.

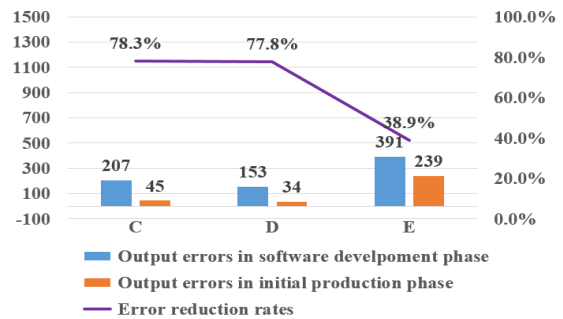


Fig. 7. Output error decreases of C, D, E in initial production phase due to software quality management in development phase

C 사업의 경우 개발단계 품질관리 지원 중 산출물 오류 개수가 최초 양산단계에서 78.3% 감소했다. D 사업의 경우 77.8%, E 사업의 경우 38.3% 감소한 것으로 나타났다. 또한 C, D, E의 CSCI는 각각 26개, 11개, 48개

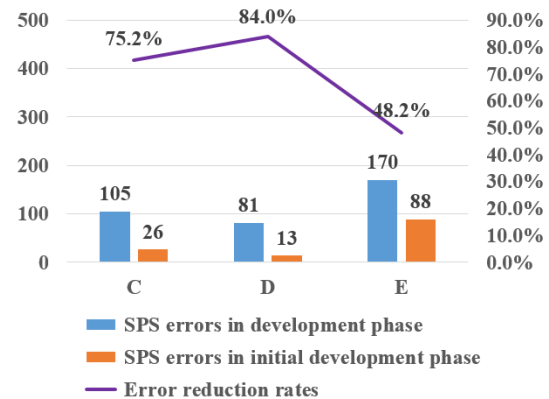


Fig. 8. SPS output error decreases of C, D, E in initial production phase due to software quality management in development phase

이며 최초 양산단계의 단위 CSCI 당 오류 개수는 1.73, 3.09, 4.98이다. 이는 4.2절에서 제시한 단위 CSCI 당 오류 개수인 5.84개 보다 낮은 수이다.

개별 사업의 SPS 산출물 오류의 개수 역시 최초 양산 단계에서 감소했다. Fig. 8은 C, D, E의 SPS 오류 변화를 보여준다.

개발단계 품질관리 지원으로 최초 양산단계의 SPS 산출물 오류는 C는 75.2%, D는 84%, E는 48.2% 감소했다.

### 5.3 기술문서의 연계성 오류 감소

개발단계 품질관리 지원은 오류 분포에도 영향을 주었다. 4.3절에서 제시한 평균 오류 분포와 C, D, E 사업의 오류 분포를 확인하면 Table 2와 같다.

Table 2. Output error category of C, D, E

Category	Mean percentage of output errors	C	D	E
SW technical documents	31%	53%	38%	37%
Computer files	24%	9%	35%	25%
Document Connectivity	35%	33%	24%	19%
KDSIS Connectivity	10%	5%	3%	19%
Total	100%	100%	100%	100%

개발단계 품질관리 지원을 받은 3개 사업은 기술문서의 연계성 항목이 평균보다 낮은 분포를 보이는 것을 확인할 수 있었다.

### 5.4 효과 분석

개발단계 품질관리 지원으로 SPS 오류 개수가 Table 3과 같이 37.8% 감소하였다. 그렇지만 최초 양산단계에서도 SPS 산출물 오류가 완전히 제거되지 않고 계속 발견되고 있다. 그 이유는 개발단계에서 전수검사가 아닌 샘플링을 통한 SPS 오류를 검출하고 있고 아직 개발단계의 산출물 오류 검증 프로세스가 정립되지 않았기 때문이다. 이는 추후에 보완과 발전이 필요한 부분이다.

Table 3. Variation of SPS errors

Type	Development Phase	Initial Production Phase	Decrease ratio
SPS Errors	489	304	37.8%

Table 4. Output error category variation

Type	Mean percentage of output errors	Average of C, D, E	Decrease ratio
Document Connectivity Errors	35%	25.3%	9.7%

또한 개발단계 품질관리 지원은 산출물 오류 분포에도 영향을 주었는데 4가지 항목 중 문서 연계성 항목이 Table 4와 같이 평균 오류 항목보다 9.7% 낮은 분포를 보이는 것을 확인했다. 기술문서의 연계성 항목은 개발단계의 산출물인 소프트웨어 기술문서 사이의 연계성, 소프트웨어 기술문서와 도면, QAR, KDS의 연계성을 점검하는 항목이다. 서로 다른 문서와의 연계성 항목은 무기체계 개발 프로세스에 대한 이해가 반드시 필요한 항목이다. 개발단계 품질관리 지원으로 연구개발주관기관의 무기체계 개발 프로세스에 대한 이해를 향상시키고 결국 산출물의 오류 개수가 줄어드는 것을 확인했다. 무기체계 개발 프로세스를 정확히 준수할수록 최종 산출물인 소프트웨어의 품질이 향상되는 것은 자명하다. 개발단계 품질관리 지원을 통해 사업의 최종 산출물인 무기체계 소프트웨어의 품질을 더 향상시키는 것이다.

## 6. 결론

본 논문에서는 개발단계 소프트웨어 품질관리 지원에 따라 최초 양산단계에서 검출되는 산출물의 오류 개수를 분석하여 개발단계 소프트웨어 품질관리의 중요성과 필요성을 보였다. 또한 개발단계부터 소프트웨어 품질관리를 수행하여 소프트웨어의 품질을 향상시켰음을 확인했다. 이후 국방기술품질원의 개발단계 소프트웨어의 품질관리 지원 확대를 통해 양산단계에서 발생할 수 있는 품질 문제를 선제적으로 개선하고 해결할 수 있도록 지속적인 연구가 필요하다.

## References

- [1] K. Y. Kwon, J. S. Joo, T. S. Kim, J. W. Oh, J. H. Baek, "A Study on Quality Assurance of Embedded Software Source Codes for Weapon Systems by Improving the Reliability Test Process", *Journal of Korea Institute of Information Security and Cryptology*, Vol.42, No.7,

pp.860~867, 2015.

DOI: <http://dx.doi.org/10.5626/JOK.2015.42.7.860>

- [2] M. K Choi, D. U. Bak, S. H. Kook, "An Efficient Method of Test Environment Setup for Weapon System Software Reliability Test", *Journal of Software Engineering Society*, Vol 28, No.1, pp.7~12, 2019.
- [3] T. H. Lee, O. H. Paek, T. H. Kim, "Current Reliability Test Status and Development Direction of Weapon System Software", *Journal of Korea Institute of Information Security and Cryptology*, Vol.28, No.6, pp.76~82, 2018.
- [4] Defense Acquisition Program Administration(DAPA), "Weapon System development and management manual", Notice, DAPA, Korea, 2020.
- [5] I. S. Ryu, B. H. Park, "A Study on improvement method of software quality assurance in development phase", *Communications of the Korean Institute of Information Scientists and Engineers*, Vol.35, No.12, pp.53-60, 2017.
- [6] C. H. Choi, "Application Case Study of R&D Quality Assurance for Defence Products", *Fall Conference of Journal of Korean Society for Quality Management*, Korean Society for Quality Management, Jinju, Korea, Vol.2018, No.0, pp.164-173, October, 2018.
- [7] J. S. Yu, C. H. Song, S. M. Kwon, B. H. Park, J. W. Oh, "The Process of Software Quality Assurance About Initial Product in Weapon System", *The Korea Academia-industrial Cooperation Society*, Vol.22, No.1, pp.285-293, 2021.  
DOI: <https://doi.org/10.5762/KAIS.2021.22.1.285>
- [8] International Standard Organization(ISO), Systems and software engineering — Systems and software Quality Requirements and Evaluation(SQuaRE) — Measurement of system and software product quality, *ISO/IEC 25023*, Geneva, Switzerland, 2016.
- [9] K. S. Shin, S. Y. Hwang, A. Y. Lee, S. C. Kim, "The Effects of Methodical Project Management on the Performance of Product Development", *Project Management Review*, Vol.3 No.2, pp.61-78, 2013.
- [10] K. S. Shin, M. J. Oh, W. K. Kim, S. H. Park, "The Effects of R&D Process Maturity on Product Development Performance: Focused on Mediating Effect of R&D Project Performance", *KIPS Transactions on Computer and Communication Systems*, Vol.7, No.7, pp.165-174, 2018.  
DOI: <https://doi.org/10.3745/KTCCS.2018.7.7.165>

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