

논문 2020-2-10 <http://dx.doi.org/10.29056/jsav.2020.12.10>

CMMI와 TMMi를 이용한 소프트웨어 Safety 성숙도 모델에 대한 연구

이승목*† , 김영곤* , 안경수*

The Study on the Software Safety Maturity Model using CMMI and TMMi

Seung-Mok Lee*† , Young-Gon Kim* , Kyung-Soo An*

요 약

최근 IoT, 인공지능, 클라우드, 빅데이터, 모바일 분야가 융합되어 4차 산업혁명이라는 새로운 산업시대가 도래하였고 전 산업군으로 확대 되었다. 이 중심에는 소프트웨어가 중요한 역할을 담당하고 있지만 다양한 산업군으로 적용, 보편화됨에 따라 소프트웨어 안전에 대한 이슈가 부각되고 있다. 하지만 현재 소프트웨어 Safety는 개발 관점에서만 활동이 집중적으로 맞춰져 있고, 조직적인 개선 활동과 평가 체계 수준에 대해서는 다소 미진하다.

본 연구에서는 이러한 미진사항을 보완하고자 소프트웨어 Safety 성숙도 모델을 정의하고, 각 성숙도 레벨의 Process Area를 명시하였다. 본 연구에서 고안된 Safety 성숙도 모델을 기반으로 조직관점의 Safety 확립과 함께 체계적인 소프트웨어 Safety 개선 활동에 기여할 것으로 기대한다.

Abstract

Recently, IoT, artificial intelligence, cloud, big data, and mobile fields have converged, leading to a new industrial era called the 4th industrial revolution. This 4th industrial revolution has been expanded to all industry area and Software has been taken as important role in this revolution. Thus Software Safety is the huge factor because Software is highly relevant to human safety in accordance with Software expansion. However this Software Safety has been focused on not organization improvement activities but current design/development,

In this paper, to solve this issue, Software Safety Maturity level and relevant Process Area is defined. This study is expected to contribute to systematic software safety activities.

한글키워드 : 소프트웨어, 안전성, 성숙도, CMMI, TMMi

keywords : Software, Safety, Maturity, CMMI, TMMi

* 한화시스템 항공시스템팀

† 교신저자: 이승목(email: seungmok23.lee@hanwha.com)

접수일자: 2020.11.12. 심사완료: 2020.12.05.

게재확정: 2020.12.21.

1. 서론

IoT, 인공지능, 클라우드, 빅데이터, 모바일 기술은 현재 주요 기술 트렌드로 자리잡았고 그에

따른 다양한 서비스 및 편의성을 제공하고 있다. 이러한 최신 기술이 융합되어 4차 산업혁명이라는 새로운 산업혁명이 도래하였고 이 산업혁명의 파급력은 빠르게 전 산업군에 걸쳐서 대중화, 정착화 되어가고 있다. 이러한 4차 산업혁명의 핵심인 기술 융합을 위해서 소프트웨어는 중요 요소로 부각되었으며 정부는 중고등학교 필수 과목으로 코딩교육 커리큘럼을 새로이 개설하였다. 하지만 4차 산업혁명이 전 사업군으로 적용되면서 방위산업 등에 국한되어 적용되었던 소프트웨어 Safety에 대한 이슈들이 제조, 자동차, 의료, 항공 등의 전 분야에 중요한 이슈가 되고 있다. 제조공장의 기기 오동작에 따른 현장 인력의 인명 피해와 자동차의 오동작으로 인한 탑승자 및 보행자의 인명 피해, 의료용 치료기기의 오동작에 따른 환자 및 기기 조작자의 인명 피해가 충분히 발생될 수 있으므로 소프트웨어의 Safety는 그 어느 때보다도 중요하다. 특히 2000년도 도요타 자동차의 전자제어장치(ECU)에 내장된 SW의 결함[1]과 보잉 737 MAX 기종의 받음각(AOA) 센서의 SW 오류[1]는 SW Safety에 대한 중요성을 일깨워주기 충분한 사건이었다.

구소에서는 SW 안전 분야 선정기준 및 SW 안전 공통기준 도출 연구[4], 소프트웨어안전 확보를 위한 개발프로세스 적용 확산 방안 연구[5], 소프트웨어 안전(Safety) 산업 동향 조사[6] 등의 활동을 수행했다. 또한 법률적으로 소프트웨어산업 진흥법 전부개정안에 SW안전기준 및 안전전문기관 지정이 추가[7] 되었다.

이러한 활동과 함께 소프트웨어 Safety와 관련된 표준 및 가이드 등이 존재하나 이는 엔지니어링 관점에서만 집중되어 있고 조직적 차원의 Safety 활동은 미진하다. 또한 조직적 Safety 활동 수행을 위해 해당 조직의 Safety 관리에 대한 평가도 필요하다. 이를 위해 해당 조직의 업무 역량을 나타내며 이러한 역량을 정량적으로 평가를 할 수 있는 성숙도 모델을 활용, 소프트웨어 Safety에 대한 성숙도를 정의하는 것이 매우 중요하다.

본 논문은 소프트웨어 Safety 성숙도 모델을 위해 프로세스 품질과 제품 품질 관점에서 분석, 모델을 도출한다. 프로세스 품질을 위해 CMMI를, 제품 품질을 위해 TMMi를 활용해 Safety를 위한 성숙도 모델을 제시한다.

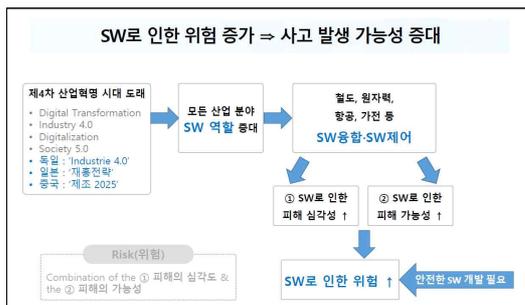


그림 1. SW로 인한 위험 증가[2]
Fig 1. The increasing danger of Software

이러한 중요 이슈에 따라 정보통신산업진흥원에서 중소기업의 SW안전기술 역량 강화를 위한 실무 가이드를 발간하였고[2], 소프트웨어정책연

2. 관련 연구

2.1 CMMI(Capability Maturity Model Integration)

CMMI를 언급하기 위해서는 CMM에 대한 접근이 먼저 필요하다. CMM(Capability Maturity Model)은 1987년에 미 국방성(DoD)이 외부 소프트웨어 개발 조직에게 발주하는 개발 프로젝트에서 발생하는 납기일 지연이나 품질 문제를 해결하기 위해 미국의 카네기 멜론 대학교의 소프트웨어 공학 연구소인 SEI(Software Engineering Institute)를 설립하여 만든 모델이다[8]. CMM은

분야별 목적에 맞게 SE-CMM, SW-CMM, P-CMM 등으로 파생되었고. 이에 따른 중복과 혼란이 야기되었다. 이러한 사유로 CMMI는 CMM 모델의 통합, 모델 사이의 불일치성 제거, 중복 제거, 비용감소, 명확성 및 이해가능성 증가, 일관성 있는 형태, 작성 방법 통일 등을 개선하였다[8]. 즉, CMMI는 여러 분야로 파생된 CMM를 통일성을 위해 하나로 통합된 (Integration) 모델이다.

현재 CMMI는 V2.0으로 2018년부터 정착화되고 있고 이전 버전인 V1.3과 비교해 볼 때 차이점은 표 1과 같다. 하지만 2.2절의 TMMi 관련연구를 통해 확인 할 수 있듯이 TMMi가 CMMI V1.3 이하 버전과 동일한 구성요소 및 호환성을 가지고 있고 CMMI와 TMMi를 동시에 활용하는 Safety 성숙도 모델을 위해서는 CMMI V1.3을 활용하는 것이 타당하다고 판단되므로 CMMI V1.3을 기준으로 설명하도록 한다.

표 1. CMMI V1.3과 V2.0 비교[9]
Table 1. CMMI Compare V1.3 with V2.0

구분	구성요소	설명
영역	프로세스 영역 (Process Area)	프랙티스 영역 (Practice Area)
영역 수	22개	20개
범주	프로세스관리, 프로젝트관리, 엔지니어링, 지원	Doing, Managing, Enabling, Improving
구조	프로세스 영역별 Practice 구조	프랙티스 레벨별 프랙티스 구조

CMMI는 프로세스 영역, 목적, 활동으로 구성되며 각 프로세스 영역 별 목적 달성을 위해 관련 활동들로 정의 된다.

CMMI는 2가지의 Representation Model이 존재하며 Representation Model을 기준으로 조직의 목표와 현 수준을 참조할 수 있다.

표 2. CMMI 구성요소
Table 2. CMMI Components

구성요소	설명[10]
프로세스 영역 (Process Area)	해당 프로세스를 위해 수행되어야 하는 프랙티스들 (Practices)의 모음으로 구성
목적 (Goal)	- 해당 프로세스 영역(PA)이 조직내에서 수행되고 있음을 판단하는 기준으로 사용 - Specific Goal(SG)과 Generic Goal(GG) 두 가지 종류
활동 (Practice)	프로세스 영역(PA)에서 정의하고 있는 목적(SG, GG)들을 달성하기 위해 필요한 구체적인 활동

표 3. CMMI Representation Model[11]
Table 3. CMMI Representation Model

구분	Staged	Continuous
개념	한 단계씩 프로세스 개선	- 조직필요에 따른 프로세스별 개선 - 프로세스관리, 프로젝트 관리, 공학, 지원 등
적용 방식	CMM 과 같은 단계별 PA 적용	- 관련되는 PA만 적용
평가 범위	성숙수준 (Maturity Level), 1 ~ 5단계	능력수준 (Capability Level), 0 ~ 5단계

구분	Staged	Continuous
수행방법	- 점진적 A수행 (PA Grouping) - 조직성숙도 중심으로 프로세스 개선	- 선택적 PA단계 수행 (Practice Grouping) - PA중심으로 업무 프로세스 개선
특징	CMM 에서 이전 용이	SPICE에서 이전용이
성숙도 수준	Maturity Level 이용 조직간 비교 가능	프로세스 영역별로 조직의 성숙도 평가 가능 (Capability Level)

CMMI Representation Model에서 조직의 성숙도 확인 및 조직간 비교를 위해서 Staged Representation을 많이 활용하는데 Staged Representation에 따르는 성숙도 단계는 초기단계, 관리단계, 정의단계, 정량적 관리단계, 최적화 단계 총 5단계로 구분된다.

표 4에서와 같이 각 Level 별 보유한 Process Area를 근거로 인증기관에서 심사 조직의 성숙도를 측정하고, Level을 부여한다. Process Area는 Level 3단계까지 대부분의 정의가 되므로 일반적으로 조직에서는 Level 3단계 인증을 우선으로 고려하고 있다.

2.2 TMMi(Test Maturity Model Integration)

TMMi는 TMMi 재단에서 개발한 테스트 프로세스 성숙도 진단 모델로 개발 능력 성숙도 모델인 CMMI와 서로 상호 보완 관계에 있다[12]. CMMI는 개발 조직의 성숙도 진단 및 개선에는 널리 사용되어 왔으나 SW 테스트 업무에 대한 평가가 취약하다는 단점이 지적되어 TMMi는 지적된 CMMI의 취약점을 보완하기 위해 개발되었다[13].

표 4. Stage of maturity CMMI
Table 4. Stage of maturity CMMI

성숙도 레벨	Process Area[11]
Level 5 (최적화단계)	- 조직적 혁신과 배치 - 분석과 해결
Level 4 (정량적 관리단계)	- 조직적 프로세스 성과 - 정략적 프로젝트 관리
Level 3 (정의단계)	- 요구사항개발 - 기술적 해결 - 제품통합 - 조직적 프로세스 초점 - 통합된 프로젝트 관리 - 통합된 공급자 관리 - 위험결정분석 및 해결 - 통합을 위한 조직환경
Level 2 (관리단계)	- 요구사항 관리 - 프로젝트 계획 - 프로젝트 감시 제어 - 공급자 합의 관리 - 측정과 분석 - 프로세스/제품품질보증 - 형상관리
Level 1 (수행단계)	없음

TMMi의 구성요소는 2.1절의 CMMI 관련 연구에서 언급한 바와 같이 V1.3과 동일하게 각 테스트 프로세스 영역으로부터 해당 프로세스를 통해 달성해야 하는 Specific Goals, Generic Goals, 특정목표를 달성하기 위한 Specific Practices와 일반목표를 달성하기 위한 Generic Practices로 구성되어 있다[14].

레벨 1단계인 Initial 단계에는 관련 프로세스 영역이 정의되어 있지 않으며, 그에 따라 레벨 2 단계부터 5단계까지 관련 영역이 정의되어 있다.

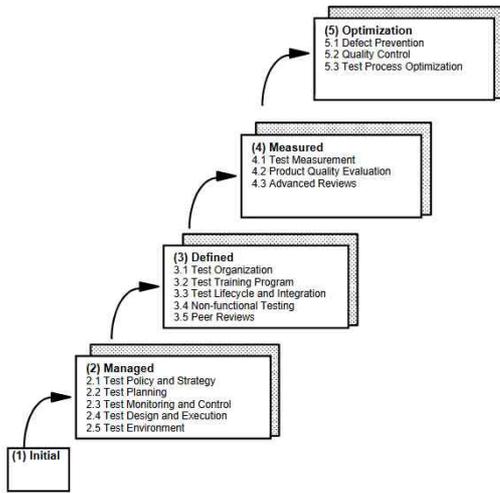


그림 2. TMMi 성숙도 레벨과 프로세스 영역[14]
Fig 2. TMMi maturity levels and process areas

3. SW Safety 성숙도 모델

3.1 CMMI를 활용한 프로세스 성숙도 접근

Safety를 위해 준수되어야 할 품질은 프로세스 품질과 제품품질이다. 그 중 프로세스 품질은 Safety가 크게 요구되는 산업안전, 자동차, 비행기 등의 소프트웨어 생명주기는 V 모델 즉, Verification과 Validation을 기반으로 하고 있다.

하지만 이러한 접근은 소프트웨어 엔지니어링 접근이며 조직의 Safety 성숙도를 위해서는 공학적인 접근뿐만 아니라 추가 고려사항이 요구된다. 이러한 고려사항은 관련연구의 CMMI 특히 CMMI-DEV(제품개발)의 PA(Process Area)를 활용해 참고해 볼 수 있다.

표 5. TMMi 테스트 성숙도 레벨[12]
Table 5. TMMi Test Maturity Level

테스트 성숙도 레벨	설명	인증 필요
레벨 1 (초기 수준)	체계를 갖추지 않고 테스트 수행. 테스트 완성도는 개인에 의존함	필요 없음
레벨 2 (관리 수준)	프로젝트 중심의 테스트 체계를 갖추. 테스트와 디버깅 구분. 기능 중심의 테스트 수행	필요
레벨 3 (정의 수준)	조직 차원의 테스트 체계를 갖추. 테스트 전문 조직 있음. 기능 외 비 기능 테스트 수행	필요
레벨 4 (측정 수준)	모든 테스트 활동을 측정하여 테스트 프로세스의 성과를 평가함. 제품 품질을 평가하는 프로세스를 갖추며, 동료 검토를 통해 프로젝트 초기부터 품질을 평가하고 예측함	필요
레벨 5 (최적화 수준)	조직의 테스트 프로세스를 지속적으로 개선하여 결함을 예방하고 통계적 기법을 적용해 테스트프로세스를 관리하고 품질을 제어함	필요

표 6. Safety 성숙도를 위한 프로세스 고려사항
Table 6. The Process consideration for Safety Maturity

고려사항	설명	V모델
프로세스 관리	엔지니어링 프로세스가 아닌 조직관리 프로세스에 초점	X
프로젝트 관리	프로젝트를 위한 - 계약관리 - 프로젝트 계획 수립 - 프로젝트 모니터링/통제	X
엔지니어링	설계프로세스 활동을 위한 - 요구사항 관리 - 검증 및 확인	○
지원	조직 관점의 - 원인분석 및 해결 - 의사결정 - 프로세스/제품 품질 보증	X

즉, CMMI의 PA에 기반하여 분석해 볼 때 엔지니어링은 V 모델을 통해서 Safety를 위한 성숙도 모델을 정의할 수 있지만 프로세스 관리, 프로젝트 관리, 지원은 Safety를 위한 성숙도 모델을 설명하기 부족하므로 관련 영역들이 Safety 성숙도 모델에 정의되어야 한다.

3.2 TMMi를 활용한 제품 성숙도 접근

Safety를 위해 또 하나 준수되어야 할 품질은 제품품질이다. 이는 V 모델 중 Validation을 통해서 단일 제품단위의 SW로 확인 가능한 항목이다. 하지만 성숙도 모델 측면에서는 단일 제품관점이 아닌 조직 관점으로 Validation을 접근해야 하며 이에 따라 테스트의 정형화와 프로세스 성숙도 연계가 추가적으로 고려되어야 한다.

표 7. Safety 성숙도를 위한 제품 고려사항
Table 7. The Product consideration for Safety Maturity

고려사항	설명	V모델
테스트 정형화	조직의 테스트를 정형화/표준화	X
	진담 테스트 조직을 활용한 전문적 테스트 수행	△
	정형화를 통한 테스트 신속화	X
프로세스 성숙도와 연계	프로세스 성숙도와 연계된 제품성숙도 필요	△

표 7과 같이 V 모델의 Validation에 통해서 성숙도를 일부 정의할 수 있지만, 테스트 정형화 및 정형화를 통한 테스트 신속성 그리고 프로세스 성숙도와 연계성에 대해서 Safety를 위한 성숙도 모델에 반영할 필요가 있다.

3.3 SW Safety 성숙도 모델

Safety 성숙도를 위해서는 프로세스 품질과 제품 품질에 대한 성숙도 반영과 CMMI와 TMMi의 구성요소를 종합적으로 고려해 볼 때 그림 3과 같이 성숙도 모델 확립이 진행되어야 한다.

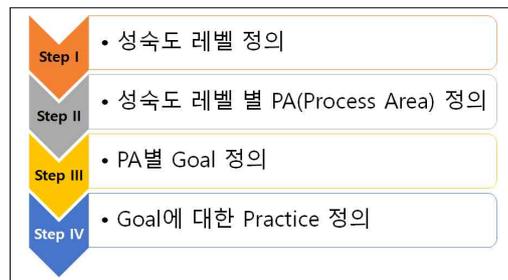


그림 3. 성숙도 모델 확립 절차
Fig 3. The procedure to define Maturity Level

표 8. SW Safety 성숙도 레벨
Table 8. The SW Safety Maturity Level

구분	성숙도 설명	
레벨별 성숙도	<p>레벨이 높을수록 성숙도가 높아지도록 정의</p>	
레벨	1	Safety 관련 프로세스 및 활동이 정의되어 있지 않고, 고객의 요청 등에 의해 임시적으로 활동 수행
	2	Safety에 대한 활동은 수행하나 특정 Project에 의존적으로 수행에 따른 제약 사항 존재
	3	SW Safety에 대한 프로세스 및 활동에 대해 정의가 되어 있고 조직별 R&R이 부여됨
	4	SW Safety에 대한 프로세스의 준수 여부를 정량적으로 할당해 평가 및 관리 통제가 가능함
	5	레벨 4에서 정량적으로 측정된 항목에 대해 지속적 개선 활동을 수행

[Step I 성숙도 레벨 정의]

Step I 성숙도 레벨의 정의는 CMMI의 Staged representation과 TMMi에서 정의된 성숙도 레벨과 마찬가지로 초기단계(레벨 1), 관리단계(레벨 2), 정의단계(레벨 3), 정량적 관리단계(레벨 4), 최적화 단계(레벨 5)의 총 5단계로 SW Safety 성숙도 레벨 정의한다. PA의 범주는 CMMI-DEV와 유사하게 프로세스 관리, 프로젝트

트 관리, 엔지니어링 그리고 지원으로 정의하되 제품품질을 위해 시험 부분이 반영되어야 한다.

[Step II 성숙도 레벨별 PA정의]

먼저 레벨 1인 초기단계에서는 Safety에 연관된 활동이 정의 되어 있지 않다. 따라서 관련 PA도 정의되지 않는다.

레벨 2인 관리 단계에서는 제한적인 활동으로 관리가 되어야 한다. 따라서 프로세스 관리 범주에는 관련 PA는 해당사항이 없지만 프로젝트 관리, 엔지니어링, 자원, 시험에 대해서 PA가 정의 되어야 한다.

표 9. SW Safety 성숙도 레벨 2 PA
Table 9. The PA for Safety Maturity Level 2

구분	상세 PA
프로세스 관리	해당사항 없음
프로젝트 관리	<ul style="list-style-type: none"> - 프로젝트 내 SW Safety 등급 부여 및 등급별 활동 계획 관리 - 프로젝트 내 SW Safety 활동 모니터링 및 통제 - 공급자의 SW Safety 계약 관리
엔지니어링	SW Safety 요구사항 관리
지원	<ul style="list-style-type: none"> - 형상관리 - SW Safety 품질활동 - SW Safety 품질활동 측정과 평가
시험	<ul style="list-style-type: none"> - SW Safety 등급별 시험 정책 및 전략 수립 - SW Safety 등급별 시험 계획 수립, 모니터링 및 통제 - SW Safety 등급별 시험 환경, 시험 설계 및 시험 시행 - SW Safety Test 환경 정의

레벨 3인 정의 단계에서는 SW Safety에 대한 프로세스 및 활동에 대해 정의가 필요하며 관련 역할과 책임이 정의가 되어야 한다. 또한 CMMI와 마찬가지로 조직의 높은 성숙도를 위해서 최소한의 레벨이므로 이에 따라 가장 많은 PA가 할당이 된다.

표 10. SW Safety 성숙도 레벨 3 PA
Table 10. The PA for Safety Maturity Level 3

구분	상세 PA
프로세스 관리	<ul style="list-style-type: none"> - SW Safety의 조직적 교육활동 정의 - SW Safety 조직 구성 및 활동에 대한 R&R과 프로세스 정의 - SW Safety 프로세스 개선 활동
프로젝트 관리	<ul style="list-style-type: none"> - SW Safety 등급별 위험관리 - SW Safety 통합 관리
엔지니어링	<ul style="list-style-type: none"> - SW Safety 개발 정의 - SW Safety 등급별 기술 정의 - SW Safety를 고려한 통합 - SW Safety 요구항목 검증 - SW 유지보수 활동 내 SW Safety 정의
지원	<ul style="list-style-type: none"> - SW Safety 수행 절차에 대한 대안과 의사결정
시험	<ul style="list-style-type: none"> - SW Safety를 위한 조직 구성 - SW Safety 시험 훈련 정의 - 시험생명주기 정의 및 Safety 별 생명주기 상세 정의 - SW Safety 비기능 시험 - 동료 검토

또한, SW 유지보수 시 SW Safety는 주요한 요소 중 하나이므로 “SW 유지보수 활동 내 SW Safety 정의” PA가 추가되었다.

레벨 4인 정량적 관리 단계는 SW 성숙도 활동에 대해 정량적인 측정으로 관리하는 단계이다. CMMI의 4 레벨과 마찬가지로 엔지니어링,

지원 범주에는 관련 PA가 없다.

레벨 5인 최적화 단계는 지속적 개선을 위해 조직 프로세스 및 시험 프로세스의 개선활동에 초점이 맞춰진 활동이므로 프로젝트 관리와 엔지니어링 범주에는 관련 PA가 해당사항이 없다.

표 11. SW Safety 성숙도 레벨 4 PA
Table 11. The PA for Safety Maturity Level 4

구분	상세 PA
프로세스 관리	Safety 조직 활동에 대한 정량적 측정
프로젝트 관리	프로젝트 Safety 활동의 정량화
엔지니어링	해당사항 없음
지원	해당사항 없음
시험	<ul style="list-style-type: none"> - SW Safety 시험 측정 - SW Safety 품질 평가 - 심화된 동료 검토

표 12. SW Safety 성숙도 5단계 PA
Table 12. The PA for Safety Maturity Level 5

구분	상세 PA
프로세스 관리	Safety 활동의 조직적 개선 및 프로세스 개선
프로젝트 관리	해당사항 없음
엔지니어링	해당사항 없음
지원	<ul style="list-style-type: none"> - SW Safety 관련 자발적인 결함 발견과 원인 식별 - SW 재사용에 대한 Safety 영향성 개선
시험	<ul style="list-style-type: none"> - SW Safety 결함 방지 - SW Safety 품질 통제 - SW Safety 시험 절차 최적화

[Step III PA별 Goal 정의]

Step III은 각 PA별로 2개의 Goal인 Specific Goal(SG)과 Generic Goal(GG)을 정의한다. 해당 Goal과 Practice를 언급함에 있어 본 논문의 지면 제약사항을 고려해 CMMI/TMMi와 동일한 PA의 대해서는 생략하고 Step II를 통해 식별된 2개의 추가 PA인 “SW 유지보수 활동 내 SW Safety 정의”와 “SW 재사용에 대한 Safety 영향성 개선”에 대해서만 Goal을 정의한다. 또한 Goal 중 Generic Goal에 대해서 CMMI/TMMi 내 성숙도 레벨별로 정의된 Generic Goal을 활용함으로써 Generic Goal에 대한 언급은 생략한다.

표 13. 추가 PA에 대한 Specific Goal(SG)
Table 13. The SG for Additional PA

구분	SG No.	Goal 내용
SW 유지보수 활동 내 SW Safety 정의	SG 1	유지보수 활동 내 SW Safety 계획
	SG 2	유지보수 간 SW Safety 활동
	SG 3	유지보수 간 SW Safety 활동 평가
SW 재사용에 대한 Safety 영향성 개선	SG 1	개선활동 선정
	SG 2	개선활동 배치

SW 유지보수 활동 내 SW Safety 정의는 해당 SW에 대한 계획을 바탕으로 유지보수간 SW Safety 실제 활동을 수행하고, 수행한 결과에 대해서 활동의 적절성을 평가하는 것이 중요한 Goal이다. SW 재사용에 대한 Safety 영향성 개선은 개선 필요사항을 식별, 선정한 다음, 개선 활동을 실제 적용/배치하여 지속적 개선활동을

수행하는 것이 Goal 이다.

[Step IV Goal에 대한 Practice 정의]

해당 Step은 각 PA별 Specific Goal과 Generic Goal에 대한 Specific Practice과 Generic Practice를 정의한다. Step IV 도 Step III와 마찬가지로 추가 PA인 “SW 유지보수 활동 내 SW Safety 정의”와 “SW 재사용에 대한 Safety 영향성 개선”에 대해서 Practice를 정의한다. Generic Goal에 대한 Generic Practice 또한 CMMI/TMMi 내 성숙도 레벨별로 정의된 Generic Practice를 활용함으로써 별도 언급은 생략한다.

SW 유지보수 활동 내 SW Safety 정의 영역의 주요 Practice는 SW 유지보수 시 수정 SW에 대한 Safety 계획과 환경을 설정하고 Safety 활동을 수행하며 관련된 활동에 대한 평가 및 분석을 정의하는 것이 각 주요 내용이다.

SW 재사용에 대한 Safety 영향성 개선 영역 내 주요 Practice 는 개선을 위한 정보 수집과 함께 개선활동 식별 후 관련 활동을 수행하고 개선사항을 적용/배치하는 것이 주요 내용이다.

표 14. SW 재사용에 대한 Safety 영향성 개선 Specific Practice(SP)

Table 14. The SP for Safety impact improvement for SW Reusability

구분	SP No.	Goal 내용
(SG 1) 개선활동 선정	SP 1.1	개선을 위한 정보 수집 및 분석
	SP 1.2	개선활동 식별 및 수행
(SG 2) 개선활동 배치	SP 2.1	개선사항 배치 계획
	SP 2.2	개선사항 배치 관리

표 15. SW 유지보수 활동 내 SW Safety 정의 Specific Practice(SP)
Table 15. The SP for SW Safety definition during SW maintenance activities

구분	SP No.	Practice 내용
(SG 1) 유지보수 활동 내 SW Safety 계획	SP 1.1	수정 SW의 Safety 계획 수립
	SP 1.2	수정 SW Safety 수행 환경 설정
(SG 2) 유지보수 간 SW Safety 활동	SP 2.1	수정 SW Safety 확립
	SP 2.2	수정 SW와 미수정 SW 간 연동 식별
(SG 3) 유지보수 간 SW Safety 활동 평가	SP 3.1	SW Safety 활동 평가 및 분석

3.4 Safety 성숙도 모델과 CMMI/TMMi 비교

Safety 성숙도 모델에 대한 PA와 CMMI와 TMMi의 PA를 비교를 해 볼 때 CMMI와 TMMi를 결합한 총 PA 수는 38개이지만 SW Safety 성숙도 모델의 총 PA 수는 39개로 비슷하게 정의되었다. PA 상세 내용을 분석하면 기존 CMMI/TMMi 대비 삭제 PA는 1개 항목이며 CMMI/TMMi 대비 추가 항목은 2개 항목이다.

삭제된 PA 항목은 CMMI에 성숙도 레벨 3에 정의된 Validation으로 제품 및 제품 요소가 의도한 환경에서 원하는 기능이 수행되는지 증명하기 위한 것이다[10]. 이러한 Validation은 TMMi의 성숙도레벨 3 PA에 기 반영이 된 사항이므로 중

복된 영역이다.

추가된 PA 항목은 Safety 성숙도 레벨 3 엔지니어링 범주의 “SW 유지보수 활동 내 SW Safety 정의”와 Safety 성숙도 레벨 5 지원 범주의 “SW 재사용에 대한 Safety 영향성 개선”이다. 이 2개 항목은 유지보수와 재사용성 측면에서 SW 생명주기의 연장과 정형화된 SW 개발 환경이 반영된 PA로써 이는 기존 SW에 대한 Safety 고려와 함께 새로운 제품 개발 시 재사용 SW에 대한 Safety 고려가 반영된 항목이다.

표 16. SW Safety 성숙도와 CMMI/TMMi 비교
Table 16. The Compare between SW Safety Maturity and CMMI/TMMi

구분	CMMI/TMMi	SW Safety 성숙도
총 PA	38	39
CMMI/TMMi 대비 삭제 건수	-	-1
CMMI/TMMi 대비 추가 건수	-	+2

4. 결론

현재 소프트웨어는 4차산업을 위한 핵심 요소이지만 소프트웨어가 모든 사업군으로 확장됨에 따라 Safety에 대한 요구가 대두되고 있다.

특히 제조, 항공, 자동차 등의 Safety가 중요한 사업군의 확장으로 인해 소프트웨어 Safety가 그 어떤 때보다도 중요한 이슈가 되었다. 이러한 이슈 해결을 위해서 소프트웨어 Safety의 체계적인 활동이 필요하고, 관련된 활동은 산업 공통인 IEC 61508, 자동차 분야의 ISO 26262, 항공분야의 DO-178 등으로 기 정의되어 있으나 이는 현

개발 소프트웨어에 국한되어 다소 한정적이다. 따라서 프로젝트뿐만 아니라 조직적인 활동까지 정의된, 소프트웨어 Safety 성숙도 모델의 정의는 필수다. 본 연구는 소프트웨어 Safety 성숙도 모델을 프로세스 품질과 제품 품질 측면으로 5가지 레벨의 성숙도와 PA(Process Area), Goal 및 Practice를 연구했다. 고안된 Safety 성숙도 모델은 SW Safety에 대해서 정량적인 평가를 수행함으로써 현재 SW Safety에 대한 조직적 차원의 지속적 개선에 기여할 것으로 판단된다.

참 고 문 헌

- [1] 김준래, “SW 오류, 대형 사고 원인 될 수 있다”, 더사이언스타임즈, <https://www.sciencetimes.co.kr/news/sw-오류-대형-사고-원인-될-수-있다>
- [2] 송지환, “쉴 산업 안전성 확보를 위한 SW의 역할과 과제”, 2019 SPRI Spring 컨퍼런스, SW+(플러스) : 지난 5년간의 SW정책 재적과 미래, p7; 대한한국, 서울, April, 2019, <https://spri.kr/posts/view/22609?code=conference>
- [2] 과학기술정보통신부, 정보통신산업진흥원, “중소기업의 SW안전기술 역량 강화를 위한 실무 가이드”, 정보통신산업진흥원, <https://www.sw-safety.co.kr/%EA%B0%80%EC%9D%B4%EB%93%9C%EB%8B%A4%EC%9A%B4%EB%A1%9C%EB%93%9C>
- [4] 송지환, 진희승, 권영환, 민상윤, 김형호, 김성원, 이지연, 박재진, “SW 안전 분야 선정 기준 및 SW 안전 공통기준 도출 연구”, 소프트웨어정책연구소, <https://spri.kr/posts/view/22635?code=research>
- [5] 진희승, 송지환, 권영환, 이낙선, 양희석, 장은미, 김문희, “소프트웨어안전 확보를 위한 개발프로세스 적용 확산 방안 연구”, 소프트웨어정책연구소, <https://spri.kr/posts/view/22636?code=research>
- [6] 진희승, 송지환, 권영환, 정영철, 주민규, 임인중, 최정윤, “소프트웨어 안전(Safety) 산업 동향 조사”, 소프트웨어정책연구소, <https://spri.kr/posts/view/22637?code=research>
- [7] 김지선, “[SW진흥법 좌담회]18년만에 개정된 법, 어떤 내용이 들어있나”, 전자신문, <https://m.etnews.com/20191115000154>
- [8] 홍영희, “소프트웨어 품질개선을 위한 CMMI와 적용에 관한 연구”(석사학위논문, 충주대학교, 2006), pp. 3-4, http://www.riss.kr/search/detail/DetailView.do?p_mat_type=be54d9b8bc7cdb09&control_no=b130522bfbba0c7dffe0bdc3ef48d419
- [9] 최영길, “CMMI 2.0 주요 변경 내용과 대응 전략”, 에스이피엠씨, pp. 2, http://www.sepmc.co.kr/common/download.asp?downfile=CMMI_2.0_주요_개정내용과_대응전략_홈페이지등록.pdf&path=board
- [10] 송봉규, “CMMI 적용을 통한 정보시스템 개발감리 개선 연구”(석사학위논문, 건국대학교, 2012), pp. 8, http://www.riss.kr/search/detail/DetailView.do?p_mat_type=be54d9b8bc7cdb09&control_no=74779932997a0dc8ffe0bdc3ef48d419
- [11] 임형도, “정량적 평가를 도입한 정보시스템 안정성 진단 평가 모델 연구”(석사학위논문, 호서대학교, 2016), pp. 17-18, http://www.riss.kr/search/detail/DetailView.do?p_mat_type=be54d9b8bc7cdb09&control_no=9c91a24d2195f410ffe0bdc3ef48d419
- [12] 윤진우, “테스트 프로세스 진단을 통한 국내 소프트웨어 테스트 성숙도 현황 연구”(석사학위논문, 강원대학교, 2014), pp. 21-22, http://www.riss.kr/search/detail/DetailView.do?p_mat_type=be54d9b8bc7cdb09&control_no=115f30a7883c4b1dffe0bdc3ef48d419
- [13] 이재웅, “TMMi 심사 결과 분석을 통한 취약점 개선방안에 대한 연구”(석사학위논문, 강원대학교, 2016), pp. 19, http://www.riss.kr/search/detail/DetailView.do?p_mat_type=be54d9b8bc7cdb09&control_no=f970c84eb6ea998bffe0bdc3ef48d419&control_no=f970c84eb6ea998bffe0bdc3ef48d419

- [14] Erik van Veenendaal, “Test Maturity Model integration (TMMi®) Guidelines for Test Process Improvement Release 1.2”, TMMi Foundation, pp. 9, Nov 2018, <https://tmmi.org/tm6/wp-content/uploads/2018/11/TMMi-Framework-R1-2.pdf>

저 자 소 개



이승목(Seung-Mok Lee)

2009 경북대학교 전기전자컴퓨터학부 졸업
2009 ~ 현재 : 한화시스템 근무
<주관심분야> 항공전자, 임베디드, 소프트웨어 품질



김영곤(Young-Gon Kim)

2009 포항공대 정보통신학 석사 졸업
2016 ~ 현재 : 한화시스템 근무
<주관심분야> DO-178/DO-330 인증, 디스플레이, 컴퓨터 그래픽스



안경수(Kyung-Soo An)

2009 경북대학교 정보통신공학 석사 졸업
2002 ~ 현재 : 한화시스템 근무
현재 : 한화시스템 항공시스템팀장
<주관심분야> 임무컴퓨터, 대화면 시험기