

테스트 성숙도 모델(TMM)과 능력 성숙도 모델 통합(CMMI)의 매핑 전략

김기두[○] 신석규 김영철^{○○}
한국정보통신기술협회, 홍익대학교 컴퓨터정보통신^{○○}
{kdkim[○], skshin}@tta.or.kr, bob@hongik.ac.kr^{○○}

A Strategy for mapping Test Maturity Model(TMM) onto Capability Maturity Model Integration(CMMI)

Kidu KIM[○] Seok-kyoo Shin R. Youngchul Kim^{○○}
Telecommunication Technology Association, Dept. of CIC., Hongik University^{○○}

요 약

빠른 산업 발전 속에서 조직과 시스템이 복잡하고 대형화되면서 소프트웨어 품질을 향상시키고 그들의 신뢰성을 측정하기 위한 성숙도 모델들이 개발되었다. 그 중에서도 CMM와 이를 발전시킨 CMMI는 많은 조직에서 적용하기 위해 노력을 기울이고 있다. 하지만 두 모델들은 테스트 속성과 테스트 프로세스에 대한 언급이 부족하다. 본 논문에서는 모든 영역을 고려한 새로운 모델을 만들기 보다는 테스트 중심으로 개발된 TMM을 사용하여 CMMI과의 매핑을 통해 소프트웨어 조직의 생산품질과 신뢰성을 향상 시키는 방법을 제안한다.

논문에서는 CMMI 단점과 이를 보완하는 방법을 제안하고 이를 위해 TMM과 CMMI의 매핑전략에 언급한다. 이를 통해 기대되는 효과를 제시한다.

1. 서 론

소프트웨어를 개발 하는 조직에서 사용하는 방법들이 점점 대형화되고 복잡해지고 있다. 그만큼 대형화되고 거대화된 소프트웨어의 품질과 신뢰성을 평가하는 것은 힘든 일이다. 지금까지 CMMI, SPICE를 비롯한 여러 가지 모델들이 조직의 신뢰성을 측정하기 위한 모델로 개발되었다.

여러 모델들 중 카네기 멜론의 CMM(Capability Maturity Model)은 소프트웨어 공학 분야에서 많은 관심을 받은 모델로 CMMI(Capability Maturity Model Integration)로 진보되었다. CMMI는 SW-CMM, SE-CMM 등 여러 모델들이 통합된 모델이다. 하지만 테스트 영역에 대한 언급이 약한 기존의 CMM의 단점을 CMMI 역시 개선시키지 못했다. 차후에 CMM 혹은 CMMI를 적용하려고 하거나 현재 CMMI를 사용하고 있는 조직에서 단점을 개선하기 위해 새로운 모델을 개발하기 보다는 TMM을 함께 적용한다면 신뢰성의 수준을 높일 수 있음을 예상한다.

본 논문에서는 CMMI에서 부족한 테스트 영역을 향상시키기 위해 테스트 모델을 새롭게 개발하기 보다는 기존에 개발된 TMM과 CMMI와의 매핑을 통해 조직의 생산 품질과 신뢰성을 높이는 방법을 제안한다.

2. 관련 연구

2.1 CMM(Capability Maturity Model)

CMM은 카네기 멜론의 소프트웨어 공학 협회 (Software Engineering Institute :SEI)에 의해 소프트웨어 프로세스를 포괄적으로 평가하고와 향상시키기 위한 목적으로 개발되었다. CMM의 구조는 그림 1과 같은 구조로 되어있으며, 레벨의 구분을 통해 프로세스 성숙도 성장을 위한 지원이 이루어진다.

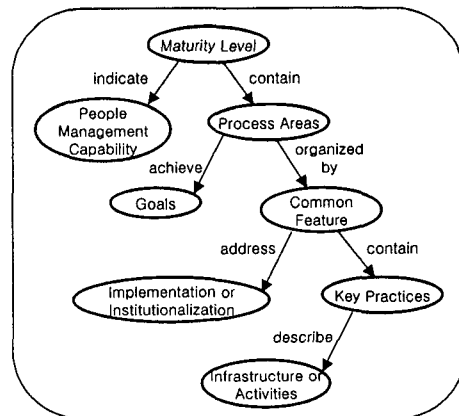


그림 1 CMM의 구조[1]

프로세스 성숙도 레벨은 CMM의 5개의 레벨중의 하나로 식별된다. 각각의 레벨은 프로세스 목적의 집합, 소프트웨어 프로세스의 중요 컴포넌트를 안정화하는 각각의 요소로 구성된다. CMM에서의 레벨1을 제외한 모든 레벨은 핵심 프로세스 영역(key process area)이 일반적인 특징으로 조직되는 내부적 구조를 갖고 있다. 핵심 프로세스 영역은 주어진 레벨에서 성숙도를 이루기 위해 조직이 중점으로 두어야 하는 것을 나타낸다. 그 일반적인 특징은 핵심 프로세스 영역으로 요구되는 성취를 완전히 지원하는 중요한 사례를 명시한다.[7],[8]

2.2 CMMI

CMMI는 카네기 멜론의 SEI에서 미 국방성(DoD)의 지원을 받아 개발한 모델로 기존의 CMM이 확장된 모델이다. CMMI는 1993년에 개발된 SW-CMM(CMM for Software)와 1994년에 개발된 SE-CMM(CMM for System)이 통합된 형태이다. CMMI내에는 SW-CMM과 SE-CMM외에도 Systems Engineering Capability Assessment Model(SECAM)과 IPD-CMM은 Integrated Product and Process Development(IPPD) 환경에 초점이 맞추어진 프로세스 성숙도 모델이 포함되어 있다. 그리고 CMMI version 1.1 모델의 공표와 함께 2003년도 말을 기준으로 기존의 SW-CMM 모델의 공식적인 폐기를 선언하였다. CMMI의 초창기 의도는 시스템공학(CMMI-SE)과 소프트웨어공학(CMMI-SW)에 따라 두 가지로 분리된 모델을 생각하였으나, 여러 토론을 거쳐서 EIA/IS 731.1로부터 Continuous representation을 유지하고, SW-CMM에서 익숙한 Staged representation으로 두 가지의 representation을 유지하면서 프로세스 모델은 하나로 통합하는 것으로 최종적으로 결정되었다. 따라서, 어떤 representation이 좋은지가 중요한 것이 아니라, 적용하려는 조직의 요구에 어느 representation이 가장 적합한지를 결정하여 적용하는 것이 중요하다 할 수 있다 [2]

2.3 TMM(Testing Maturity Model)

TMM은 1996년 일리노이 공대의 Burnstein 교수 팀에 의해 개발된 테스트 성숙도 측정을 위한 모델이다. TMM에서는 CMM의 테스트 활동에 대한 프로세스 개선을 보조하기 위해 개발되었으며, 기존의 심사 모델 중에서도 가장 일관성 있고 완전한 성숙도 모델 구조를 갖고 있으며, 심사 모델 및 절차, 심사 도구 및 질문서, 팀 교육 등에 관한 기준을 제시하고 있다. TMM 또한 CMM과 같이 모두 5개의 레벨로 정의하고 있으며 초기, 정의, 통

합, 관리와 측정, 최적화/결함 예방과 품질 제어의 레벨로 정의되고 있다.[3][4]

3. 본론

3.1 매핑 전략

TMM과 CMMI를 매핑하기에 앞서 두 모델과 CMM이 갖고 있는 관계를 알아야 한다. CMM은 앞서 관련연구에서 언급했듯이 소프트웨어 개발 성숙도를 알아보기 위해 개발된 모델이다. 하지만 CMM의 테스트 부분의 언급이 부족하여 CMM을 테스트 초점으로 개발한 것이 TMM이다. CMMI는 CMM의 후속모델로 시스템 공학과 소프트웨어 공학의 기능적 통합에 중점을 두고 있으며, 통합된 제품(product)을 개발하기 위한 기반을 제공하는 모델이다.[5] CMMI도 CMM이 안고 있는 테스트 영역에 대한 부족한 부분들을 갖고 개발되었다. TMM과 CMMI는 그림 2에서 보여주듯이 CMM을 기반으로 개발된 모델이다. 따라서 모델 간에 깊은 관계성을 갖게 된다.

하지만 TMM과 CMMI는 서로 개발 초점이 다르고 상호 고려하지 않고 개발된 모델이기 때문에 두 모델간의 공통성과 관련성을 찾아 매핑해야 한다.

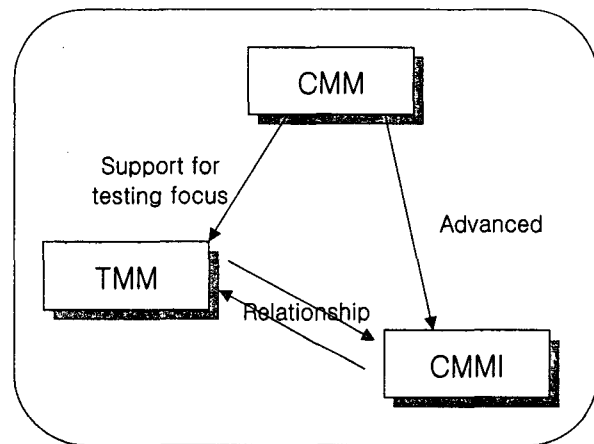


그림 2 CMM, TMM, CMMI의 관계

TMM과 CMMI의 매핑을 위해서 우리는 두 모델의 구조와 특성에 대해서 비교해 보았다. 두 모델 모두 CMM을 기반으로 개발된 모델들이기 때문에 구조와 각 레벨에 따른 정의는 유사하였다. 특히 CMMI를 표현하는 방법 중 staged representation이 TMM의 구조가 부분적으로 일치함을 알 수 있었다.

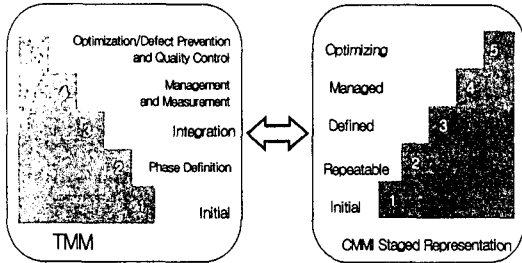


그림 3 TMM과 CMMI의 레벨

TMM이 갖고 있는 5개의 레벨과 CMMI의 staged representation이 갖고 있는 5개의 레벨을 비교하면 그림 4와 같다. 두개의 모델 모두 1~5까지 다섯 단계로 레벨을 정의하고 있으며 레벨 정의 또한 공통된 요소를 갖고 있다.

두 모델의 비교는 TMM 레벨을 기반으로 CMMI를 분석하였고, 실제 TMM과 CMMI는 매핑하는데 있어 정확히 1:1 매핑이 되지는 않았다.[6]

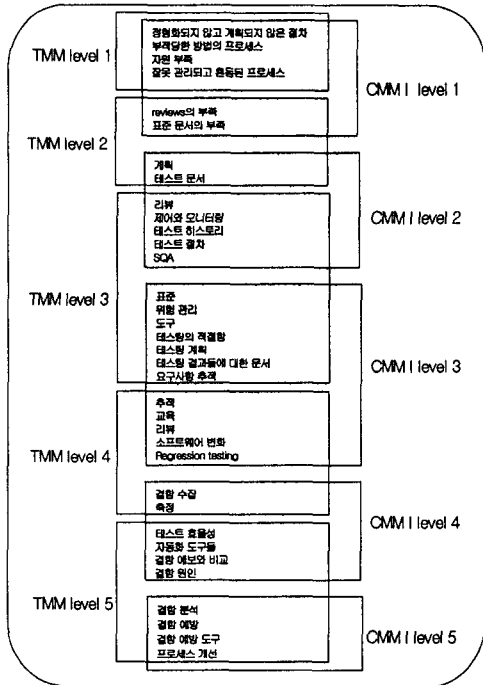


그림 4 TMM과 CMMI의 매핑 맵

4 결론

소프트웨어의 품질을 측정하고 향상 시키기 위해 많은 개발 조직들이 CMM 혹은 CMMI를 사용하거나 적용하려

고 한다. 하지만 CMM은 테스트 영역에 대한 언급이 부족한 단점을 갖고 있다. CMMI역시 CMM을 기반으로 개발되었기 때문에 테스트에 대한 부족함이 채워지지 않았다. 테스트 영역을 보강하기 위해서는 새로운 테스트 모델을 개발하는 방법도 있다. 하지만 점점 복잡하고 대형화되는 현실에서 새로운 테스트 모델을 개발하는데 사용되는 비용과 시간을 예상하기 어렵다. 이를 해결하기 위한 방법으로 우리는 기존의 TMM을 CMMI와 매핑시켜 사용하는 방법을 제안하였다.

우리는 TMM을 기반으로 CMMI의 구조와 staged representation이 갖고 있는 레벨을 비교하여, TMM과 CMMI가 갖고 있는 공통된 요소를 통해 두 모델이 매핑될 수 있음을 보였다.

두 모델의 매핑을 통해 우리는 CMMI를 사용하려하거나 적용하고 있는 조직에서 CMMI의 약점인 테스트 영역을 보강함으로써 조직의 품질 향상과 함께 개발 비용 절감의 효과도 얻을 수 있으리라 예상한다.

향후 연구과제로 TMM을 이용하여 Embedded TMM으로 확장하고 맞춤형 자동화 도구 생성을 진행하고 있다.

참고문헌

- [1]Paulk M.C., Curtis, B., Chrissis, M.B., Weber, C.V., "Capability maturity model" Software, IEEE Volume 10, Issue 4, July 1993 Page(s):18 - 27
- [2]P. Curtis, D.M. Phillips, and J. Weszka, "CMMI SM-The Evolution Continues Systems Engineering", Vol.5, No.1, 2002.
- [3]Ilene Burnstein, Taratip Suwannasart, and C.R. Carlson, "Developing a Testing Maturity Model: Part I", Crosstalk (August 1996)
- [4]Ilene Burnstein, Taratip Suwannasart, and C.R. Carlson, "Developing a Testing Maturity Model: Part II, Crosstalk" (September 1996)
- [5]http://www.sol-link.com/resources/3_report_cmml.html
- [6]Taratip Suwannasart, Towards Development of A Testing Maturity Model, IIT(1996)
- [7]김기두, "테스트 프로세스 성숙도 향상을 통한 테스트 성숙도 모델(TMM) 개선에 관한 연구", 홍익대 (2004년)
- [8]김기두, 김영철, "테스트 프로세스 향상(TPI)을 통한 테스트 성숙도 모델(TMM) 개선에 관한 연구", 한국 소프트웨어공학 학술대회 논문집, 제1권 제 1호, 2005년 2월