

SW 프로세스개선과 조직 변화관리

김승권*, 조성현**, 윤종수***

SW Process improvement and Organization Change Management

Seung-Gweon Kim*, Sung-Hyun Jo**, Joong-Soo Yoon***

요 약

본 논문에서는 조직의 변화인식 수준과 소프트웨어 프로세스 개선사이의 관계를 대립되는 가치 틀을 이용하여 분석하였다. 조직의 변화인식 수준 측정을 위해, 변화관리의 성과를 예측하는데 사용되는 DICE 모델을 적용하였다. 조직 변화의 핵심요소인 4가지 요소인 Duration(기간), Integrity(성실성), Commitment(조직몰입), Effort(노력)에 대한 점수화가 이루어지고, 이를 바탕으로 조직의 변화의지를 win, worry 혹은 woe 구역(zone)으로 구분한다. 본 논문에서는 DICE 점수를 소프트웨어 프로세스 개선의 성과를 예측하기 위해 독립변수로 활용하였다. 분석 결과에 따르면, DICE 점수가 높은 조직일수록 소프트웨어 프로세스 개선의 성과가 좋은 것으로 나타나고 있다.

▶ Keywords : 소프트웨어 프로세스 개선, DICE 프레임워크, 소프트웨어 공학수준, 붓스트랩, 일원분산분석

Abstract

We explored the relationship between the level of change awareness and deployment of software process improvement (SPI) approaches using a competing values framework. To measure awareness level of organization's change, we used DICE framework provides means for predicting the outcome of change management initiatives. The four factors for organization's change: duration, integrity, commitment, and effort are evaluated and a score is calculated. The DICE® score is used to classify projects into win, worry, or woe zones. In this paper, we apply the DICE® score as an independent variable to predict the outcome of a software process improvement. Our results

• 제1저자 : 김승권 • 제2저자 : 조성현 • 교신저자 : 윤종수

• 투고일 : 2012. 11. 21, 심사일 : 2013. 1. 15, 게재확정일 : 2013. 2. 5.

*,** 정보통신산업진흥원 소프트웨어 공학센터(Software Engineering Center of National Information Promotion Agency)

*** 강남대학교 경영대학(School of Business Administration, Kangnam University)

indicated that the Organization have a higher chance of success have the better outcome in software process improvement.

▶ Keywords : Software Process Improvement, DICE® framework, Software Engineering Level, Bootstrap, Oneway ANOVA

I. 서 론

소프트웨어를 개발하는 작업은 복잡한 업무로 알려져 있다. 새로운 개발방법론이나 기술이 발전되고 있음에도 불구하고 여전히 많은 소프트웨어 개발 프로젝트들이 납기나 예산을 준수하지 못하고 실패하고 있다. 새로운 기술 위주의 개발이 성공을 거두지 못하게 됨에 따라 소프트웨어 개발의 조직적인 측면과 프로세스 측면이 강조되면서 품질경영과 소프트웨어 프로세스 개선이 소프트웨어 산업계로 퍼져나가게 되었다.

많은 SW 기업들이 SW 개발 조직의 개발 능력과 SW 제품 품질을 개선하기 위한 방안으로 SW 프로세스 개선 (Software Process Improvement, SPI)를 추진하고 있다 [13][14][2][3]. 하지만 SW 프로세스 개선의 성과에 대한 정량적인 증거들이 여전히 부족한 것으로 제시되고 있다 [4][5]. SPI는 효과적인 SW개발을 위한 방법론, 실천 (practice)과 절차를 지원하고, 지속적인 기업 활동으로 통합할 수 있는 기반구조와 문화를 갖추는 것을 목표로 삼고 있다. 하지만 이러한 목표는 달성하기 어려운 것으로 제시되고 있다 [6]. 기본적으로 SPI는 경영진의 추진의지(commitment), 충분한 지원(resource) 및 조직수준의 기술(skill)을 요구할 뿐만 아니라 실패의 위험이 높은 광범위하고 복잡한 조직 변화를 수반하기 때문이다.

Ravichandran and Rai[16]는 SW 프로세스 개선을 추진하는 기업들이 기술적인 문제보다는 조직적인 문제를 해결하는데 어려움을 겪는 것으로 제시하고 있다. 또한 여러 연구들이 SW 프로세스 개선이 조직의 사회적인 측면을 효과적으로 처리하지 못하고 있음을 제시하고 있다[17][15][19].

미 카네기멜론 대학의 SEI(Software Engineering Institute)의 보고서에 따르면, SW 프로세스 개선 활동이 초기 SW 프로세스 심사이후에 요구되는 변화를 관리하는데 어려움을 겪고 있는 것으로 나타나고 있다. 실제로 1987년에서 2004사이에 총 2,561개의 심사가 수행되었지만, 이 중에서 약 25%에 해당하는 630개의 조직만이 재심사를 받은 것

으로 나타나고 있다[22]. 나머지 75%의 조직의 소프트웨어 프로세스 개선 활동은 중간에 중지하거나, 지속적으로 추진되지 못했다는 것을 알 수 있다.

SW 프로세스 개선의 어려움으로 인해서 많은 기업들이 소프트웨어 프로세스 개선을 추진하는데 반해, 아직도 많은 SW 개발 조직들이 낮은 성숙도 수준에 머물게 되고, 많은 SW 개발 프로젝트들은 실패하고 있는 것으로 보인다[21]. 실제로 2009년 Standish Chaos Report(8)에 의하면 고객이 요구하는 특징과 기능을 갖춘 시스템을 주어진 납기와 비용을 준수하면서 성공적으로 SW개발 프로젝트를 수행한 성공률(successful)은 32%, 요구사항 충족 부족이거나 납기 또는 비용 초과인 도전(challenged) 상태는 44%, 중간에 중단되거나 사용되지 못한 실패(failed) 상태는 24%로 분석되고 있다[8].

일반적으로 SW 프로세스 개선을 바라보는 시각들이 기술이나 공학적인 관점이나 이슈에 집중하면서 상대적으로 조직 변화에 대한 관점이 소홀하게 다루어져 왔다. 소프트웨어 프로세스 개선은 모든 조직 활동에서 소프트웨어 개발자를 포함한 소프트웨어 인력들의 사고방식과 행동양식을 바꾸는 것이 바탕이 된다[12]. 또한 이러한 변화가 단순한 일회성의 변화가 아니라 지속적으로 유지되면서 발전해 나가고, 조직원들에게 체화되어 모든 조직원이 항상 주어진 절차와 규정을 준수하고, 실천할 수 있는 내재화(institutionalization)과정은 성공적인 소프트웨어 프로세스 개선의 핵심적인 요소이다.

성공적인 조직의 변화를 위해 문화, 리더쉽, 동기유발과 같은 소프트 측면이 강조되어 왔다. 이런 소프트적인 요인들이 성공적인 변화관리를 위한 중요 요인이기는 하지만, 이런 소프트 측면만으로는 조직의 성공적인 변화를 보장하기는 어렵다. 더욱이 이런 소프트 요인들은 정량화나 측정이 어려운 것이 현실이다. 하지만 시간, 자원(resource), 비즈니스 목표와 같은 변화관리의 하드 요인들은 측정하기 쉬운 뿐만 아니라 조직차원에서도 훨씬 빠르고, 직접적으로 영향을 미칠 수 있는 요소로 인식되고 있다.

DICE(Duration, Integrity, Commitment, Effort) 프레임워크는 조직변화 프로젝트의 성공적인 구현이나 성공적

인 개선결과를 예측하는데 사용될 수 있는 유용한 도구로 동기유발이나 비전과 같은 소프트적인 요소보다는 하드요소에 중점을 두고 있다. 이 모델은 보스턴 컨설팅 그룹의 Sirkin et al[20]이 개발된 모델로 성공적인 조직 변화를 위한 공통분모를 식별하기 위해 225개 프로젝트의 분석을 통해 개발되었다. DICE 모델이 제시하는 4가지 핵심적인 조직변화 요인은 기간(Duration), 성실성(Integrity), 조직몰입(Commitment), 노력(Effort)라고 제시하고 있다.

SW 프로세스 개선은 조직변화의 일환으로 많은 비용과 인력이 투입 될 뿐만 아니라 오랜 시간이 소요되는 어렵고 힘든 작업이라는 점을 고려하면, 성공적인 SW 프로세스 개선에 영향을 미치는 요인을 식별하는 것은 매우 중요하다. 따라서 본 연구는 DICE 모델을 활용하여, 해당 조직의 변화의지를 측정하고, 조직변화의 결과물인 SW공학수준점수와의 관계를 살펴보고자 한다.

II. 관련 연구

1. SW 프로세스 개선과 조직변화

소프트웨어를 개발하는 일은 아주 복잡한 업무로 알려져 있다. 새로운 방법론이나 기술의 등장과 같은 기술적인 진보에도 불구하고 여전히 SW 프로젝트의 납기 지연, 비용초과 및 프로젝트 실패가 빈번하게 발생하고 있다. 이렇듯 기술 지향적인 접근방식의 제한된 성공으로 인해서 조직 혹은 프로세스 지향의 소프트웨어 개발에 대한 관심이 집중되었다. 그 결과, 품질관리와 소프트웨어 프로세스 영역에서 혁신적인 사례들이 소개되면서, 소프트웨어 산업계에 소프트웨어 프로세스 개선이 보급되기 시작했다.

소프트웨어 프로세스 개선은 조직의 변화를 의미한다. 실제로 많은 연구자들이 소프트웨어 프로세스를 개선하면서 직면하는 문제로 불충분한 조직변화관리를 제시되고 있다 [17][19]. 조직 내에서 발생하는 모든 변화는 조직원들이 새로운 업무방식에 적응하려고 노력하고, 현재의 기술에 새로운 기술을 통합하려고 노력함에 따라 필연적으로 갈등과 혼란을 야기하게 된다. 게다가 새로운 기술들은 좋은 환경에서 도입되는 경우는 거의 없다. 왜냐하면, 프로젝트는 불가피하게 다양한 압력과 조직원들이 익숙한 업무패턴을 바꾸어야 하는 스트레스에서 진행되기 때문이다.

실제로 많은 기업들이 소프트웨어 프로세스 개선이 기업들에게 좋다는 것을 인정하고 적극적으로 도입하고 있지만 이런

많은 시도들이 실패하는 것으로 나타나고 있다[17]. 실패요인들을 살펴보면 조직적인 상황을 무시했거나 조직 문화 자체를 하나의 기계적인 투입-산출의 개념으로 인식했다는 점이다. 이런 관점들은 소프트웨어 프로세스 개선이 다양한 요인들로 구성된 조직의 변화를 기반으로 하고 있다는 점을 무시하고 있는 것이다[9].

따라서 소프트웨어 프로세스의 개선을 추진하는 소프트웨어 엔지니어들은 항상 변화의 범위나 복잡성을 과소평가한다. 이런 과소평가의 주요인으로는 소프트웨어 관리자들이 조직 차원의 변화에 대한 경험이 없고, 기존의 소프트웨어 프로세스 관련 연구들이 조직의 변화관리를 성공요인으로 분석하지 않았기 때문이다.

소프트웨어 프로세스 모델을 활용해서 프로세스 개선할 때 너무 관료적인 요소가 많다는 점이 지적되고 있다. 프로세스 개선 모델 중에서 가장 대표적인 모델인 CMMI도 너무 모호하고, 경직되어 있다는 비평을 듣고 있다. 이러한 모호성과 경직성 때문에 CMMI를 도입하려는 기업들은 자신들의 필요에 맞게 모델을 해석하기 위한 많은 노력을 투입하게 된다 [10]. 즉, CMMI를 프로세스 개선을 위한 요리책으로 가정하면, 요리법이 구체적으로 완성되지 않았음을 알게 된다. 즉, 설당 1수준, 밀가루 3컵과 같이 구체적인 조리법 대신에 CMMI는 프로세스개선을 위한 조직이나 기업에 맞게 자신들만의 요리법을 정의하도록 요구하고 있다.

이와 더불어 소프트웨어 프로세스 구현에 대한 이슈 중에서 간과되고 있는 부분이 변화에 필요한 적절한 기간이다. CMMI의 경우 프로세스 수준을 올리는데 평균 2년정도의 시간이 소요되는 것으로 분석되고 있다[6]. 또 다른 성공적인 소프트웨어 프로세스 구현 이슈는 소프트웨어 프로세스 개선의 성과관리로써 경영층이나 관련 이해당사자들이 소프트웨어 프로세스 개선에 얼마나 관심을 가지고 있는지를 보여주는 지표이다. 또한 소프트웨어 프로세스 개선에 개발자들의 가담시키고, 그들에 달성해야 할 명확한 목표를 제시하고, 개발자들의 자발적인 참여를 이끌어내는 것은 중요한 성공요인이다. 즉, 변화에 대한 개발자들의 저항을 극복하는 것이 아니라 제안된 변화를 적극적으로 받아들이게 해야 한다[11].

조직의 프로세스를 정의하는 일은 어려운 작업으로 경영진의 적극적인 유도나 지속적인 지원이 없다면 불가능한 일이다 따라서 경영진의 추진의지(commitment)와 프로세스의 정의와 구현을 위해 필요한 추가적인 노력에 대한 지원이 필수적인 요소이다. 또한 프로세스 개선의 구현 모델인 IDEAL 모델을 설명하면서 가장 중요한 요소는 추진의지와 실질적인 지원(sponsorship)이라고 지적하고 있다[18].

IDEAL모델은 SEI가 CMMI를 기반으로 프로세스 개선을 추진할 때 활용하도록 권고하는 모델로 소프트웨어 프로세스 개선 프로그램에서 수행해야 하는 활동들에 대한 로드맵을 제공한다. 이 모델은 조직 내에서 CMMI 프로세스 개선활동을 시작하고, 계획해서 구현하는 로드맵을 제시하는 조직개선 모델로써, 준비(Initiating), 진단(Diagnosing), 구축(Establishing), 실행(Action), 학습(Learning)단계로 구성된 프로세스 개선의 반복적인 라이프사이클 모델이다.

준비단계는 향후 개선을 위한 기반구조를 수립하는 단계로 역할과 책임이 정의되고, 초기 자원들이 할당된다. 또한 비즈니스 목적에 부합하는 개선 목표들이 정의되고, 경영진 조정 위원회와 소프트웨어 프로세스 그룹들이 식별된다. 진단 단계는 조직의 현 상태에 대한 베이스라인을 설정하기 위한 평가 활동들이 수행된다. 이전단계의 평가결과와 권고사항이 첫 번째 구현계획에 반영된다.

구현단계는 조직이 다루어야 할 이슈들에 대한 우선순위를 결정하고, 이를 수행하기 위한 전략이 설정된다. 수행계획서 초안이 완성되고, 측정 가능한 목표와 목표달성을 관리하기 위한 측정치(metrics)가 개발된다. 실행단계에서는 진단단계에서 식별된 개선 영역에 대한 해결책들이 제시되고, 시험사업이 진행되고, 조직전반에 걸쳐 배포된다. 학습단계에서는 보다 효과적인 IDEAL모델의 차후 적용을 위해 이전단계에서 수집된 정보, 교훈(lessons learned), 수행성과 측정치에 대한 평가가 이루어진다. 이런 과정을 통해 향후 SPI 프로그램에 사용될 수 있는 전략, 방법론 및 기반구조의 조정이 이루어지게 된다. IDEAL 모델은 조직 내에 프로세스 개선의 기반구조가 갖추어지고, 목표가 달성될 때까지 반복해서 수행된다.

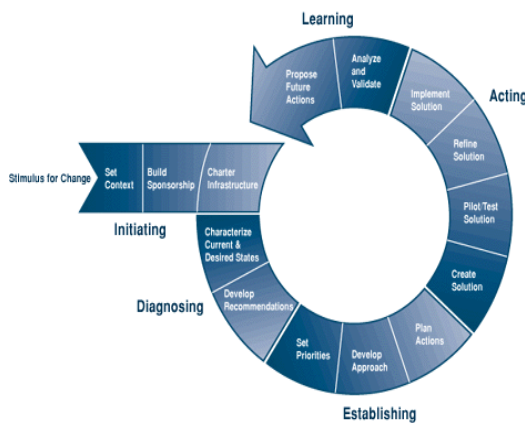


그림 1. IDEAL 모델
Fig.1. IDEAL Model

2. SW공학수준과 CMMI모델

본 논문에서는 국내 SW기업들의 SW공학기술 적용 현황을 파악하고 기업들이 수행하는 프로젝트의 성과 수준을 파악하기 위한 방안으로 SW공학수준을 정의하였다. SW공학수준은 프로세스(process), 인력(people), 기술(technology)과 같은 3가지 요소로 구성되었다(그림 2 참조).

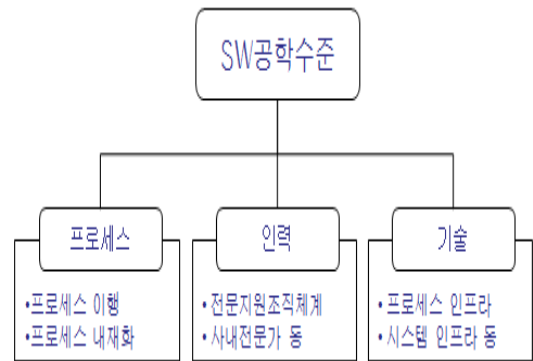


그림 2. SW공학수준의 구조
Fig. 2. The Structure of SW Engineering Level

프로세스(Process) 영역의 지표는 성공적인 SW개발 활동에 있어서 반드시 수행되어야 하거나 수행이 필요한 활동들이 SW조직에서 제대로 수행하고 있는지를 종합적으로 확인하기 위한 항목이다. 인력(People) 영역의 지표는 원활한 SW개발을 위하여 필요한 인력 보유, 지원 조직의 체제 및 보유, 인력 육성을 위한 교육 수준 등 인력 및 조직 측면에서 필요한 사항이 제대로 갖추어 졌는지를 확인하는 항목들이다. 그리고 기술(Technology) 영역 지표는 SW기업의 직원들이 SW 개발을 제대로 수행하기위하여 필요한 인프라인 프로세스 체계, 자동화 시스템 및 툴 보유 등의 구성과 활용이 어떠한지, 프로세스 자산 및 조직 정보 관리 및 활용이 어떠한지 등 특정 기술에 대한 수준이 아닌 SW개발을 위해 기본적으로 갖추어야 하는 시스템 인프라, 개발 표준 및 기법 등을 확인하는 항목들이다

SW공학수준을 측정하기 위한 SW공학수준 지표는 기본적으로 미국 SEI (Software Engineering Institute)에서 개발한 CMMI (Capability Maturity Model Integration)을 기반으로 개발되었다[1]. 특히, 프로세스 지표들은 CMMI 모델의 능력수준(Capability Level) 2와 3의 18개 주요 프로세스 영역(Key Process Area; KPA)을 바탕으로 개발되었다.

CMMI의 레벨 2에 해당하는 프로세스는 프로젝트 계획수립, 요구사항관리, 형상관리, 프로젝트 모니터링 및 통제, 프로세스 및 프로세스 및 제품품질 보증, 공공업체관리 및 측정 및 분석 프로세스 영역을 포함한 7개의 영역이 존재하고, CMMI 레벨 3영역에는 통합프로젝트, 요구사항개발, 의사결정분석 및 해결, 조직프로세스 정의 등을 포함한 11개 프로세스 영역으로 구성된다(표 3 참조).

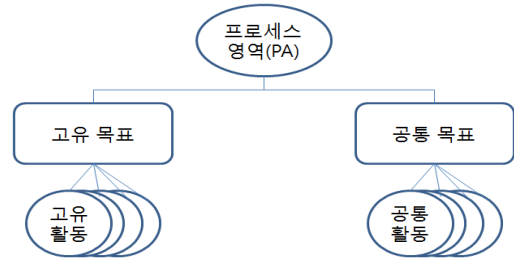


그림 3. CMMI의 프로세스 영역의 구조
Fig.3. The process area structure of CMMI

표 4 CMMI 프로세스 영역 및 성숙도 수준
Table. 1 Process area and maturity level of CMMI

수준	프로젝트관리 영역	공학 영역	지원 영역	프로세스 관리 영역
level 2	• 프로젝트 계획 수립	• 요구사항 관리	• 형상관리	
	• 프로젝트 모니터링 및 통제		• 프로세스 및 제품품질 보증	
	• 공공업체 관리		• 측정 및 분석	
level 3	• 통합 프로젝트 관리	• 요구사항 개발	• 의사결정 분석 및 해결	• 조직 프로세스 정의
	• 위험관리	• 기술적 해결		• 조직 프로세스 중점관리
		• 제품 통합		• 조직 교육관리
		• 검증		
level 4				• 조직 프로세스 초점
				• 정량적 프로세스 성과관리
level 5			• 원인 분석 및 해결	• 조직혁신 및 확산

CMMI의 개별 프로세스들은 달성해야 할 고유목표(Specific Goal)와 공통 목표(Generic Goal)를 가지고 있으며, 이런 목표를 달성하기 위해 수행되어야 하는 활동으로 구성되어 있다(그림 3 참조). 이를 기반으로 SW공학수준 점수를 산출하기 위해 각 프로세스의 활동들의 수행 여부를 2점 척도(수행/미수행)로 측정하고, 수행하면 1점 아니면 0점을 부여하여 SW공학 수준점수를 산출하였다. 인력과 기술 영역의 SW공학 수준점수도 동일한 척도를 기준으로 개별 프로세스를 수행하는 인력이나 기술이 존재하면 1점을 아니면 0점을 부여하였다.

이렇게 산출된 SW공학수준점수를 기반으로 기업의 특성을 파악하기 위해 Advanced, Average, Absent의 3가지 등급으로 분류하였다. 이는 SW공학수준과 CMMI의 Level과 벤치마킹을 위해 CMMI의 Level 체계와 유사한 구조로 구성하였다. 하지만 등급을 부여하는 방식에서 다소 차이가 있다. 예를 들어 CMMI level2는 CMMI의 level 2영역인 7개의 영역만을 대상으로 심사가 이루어지지만, SW공학수준 등급은 전체 18개 영역을 대상으로 점수가 산출된다는 차이가 있다. SW공학수준별 SW기업의 전반적인 역량 특성에 대한 설명은 표 5에 나타난다.

표 2. SW공학 수준별 기업의 특징
Table. 2 Organizational features of SW engineering level

등급	점수분포	등급별 SW기업 특성
Advanced	80점 이상	프로젝트 관리, 개발, 지원, 프로세스 관리 활동이 균형을 이루며 실제 업무 환경에 잘 적용되어 그 효과가 충분히 발생되고 있는 조직
Average	60점 이상 ~ 80점 미만	프로젝트 관리, 개발, 지원, 프로세스 관리 활동의 일부만이 실제 업무 환경에 적용되어 일부의 효과가 발생되고 있으며, 해당 활동들의 내재화가 일부 수행되고 있는 조직
Absent	60점 미만	프로젝트 관리, 개발, 지원, 프로세스 관리 활동의 대부분이 실제 업무 환경에 적용되지 못하고 있으며, 해당 활동들의 내재화 또한 수행되고 있지 않은 조직

3. DICE(Duration, Integrity, Commitment, Effort) 모델

DICE모델은 기업이 얼마나 성공적으로 변화를 주도하고 실행할 수 있는지를 산출하는데 사용될 수 있는 분석기법으로, 보스턴 컨설팅 그룹의 Sirkin et al[20]이 성공적인 조직변화 요인을 찾기 위해 1992년부터 1994년까지 225개의

기업을 분석한 BCG의 연구프로젝트를 근거로 하고 있다. 이 모델은 1,100개 이상의 기업에서 사용되어 왔으며, 그 결과는 이러 요인들은 조직변화 프로젝트의 결과물을 예측하는데 상관관계가 있다고 제시되고 있다. 하지만 실제로 조직을 변화시키기 위한 2/3의 프로젝트들이 실패하는 것으로 제시되고 있다[21]. 이는 조직의 변화의 어려움을 보여주는 일면이라고 할 수 있다.

최근 변화관리 전문가들은 문화, 리더십, 및 동기부여와 같은 소프트 이슈들에 초점을 맞춰왔다. 이런 이슈들이 성공 요인임에 틀림없지만 이런 요인들만 관리하는 것만으로는 조직 변화 프로그램을 구현하는데 부족한 측면이 있다. 이런 소프트 이슈들은 많은 변화 프로그램의 성과에 직접적인 영향 요인이 아니다. 예를 들어 비전을 제시하는 리더십이 변화 프로젝트에 중요한 요인이지만 항상 그런 것은 아니다.

조직원과의 의사소통도 마찬가지다. 게다가 태도나 리더십을 바꾸는 일은 쉽지 않은 일이다. 왜냐하면 이런 요인들이 조직과 조직 구성원들 속에 녹아있기 때문이다. 또한 문화나 동기부여 측면에서 변화가 일어난다고 하더라도 설문이나 인터뷰와 같은 간접적인 방법을 통해 측정될 수밖에 없고, 안정적인 데이터를 얻기가 힘든 것이 사실이다.

DICE 모델이 초점을 맞추고 있는 부분은 이런 앞서 살펴본 조직의 소프트 이슈 보다는 확실한 요인(hard factor)이다. 이런 확실한 정보는 세 가지의 독특한 특성을 지니고 있다. 첫째는 기업들이 직접적이거나 혹은 간접적인 방법을 통해 이들을 측정할 수 있다. 둘째는 이런 요인의 중요성을 조직 내/외부와 쉽게 의사소통할 수 있다는 것이다. 마지막으로 비즈니스가 빠르게 이런 요인들에 영향을 미친다는 것이다. 즉, 조직의 변화에 직접적으로 영향을 미치는 '확실한 요인'은 조직변화를 완수해야하는 시간(time), 투입되는 인력과 향후 기대되는 재무적인 성과들을 들 수 있다.

DICE 모델이 제시하는 조직 변화의 성공요소로는 변화의 수행 및 개선활동에 대한 공식적인 검토주기와 같은 기간(Duration), 프로세스 개선을 추진하는 인력의 역량 및 투입되는 공수와 같은 성실성(Integrity), 경영진의 변화인식 수준 및 지원의지와 프로세스 개선에 대한 조직원의 인식수준과 같은 조직몰입(Commitment), 변화를 위해 현장의 종업원들이 나타내는 변화를 극복해야 하려는 노력을 의미하는 추가 투입공수와 같은 노력(Effort)이다 (표 3 참조).

표 6. DICE 구성요소
Table 5. Components of DICE

구성요소	설명
Duration (기간)	D 단계별 검토 주기 - 짧으면 짧을수록 좋음
Integrity (성실성)	I 프로젝트를 수행하는 직원들의 능력
Commitment (조직몰입)	C 상위경영층(C1)과 현장 관리자(C2)의 프로젝트에 대한 충실도
Effort (노력)	E 새로운 프로세스를 수용하기 위해 기존직원들이 수행해야할 추가적인 업무

DICE모델의 계산공식은 간단한 질문을 통해, 각각의 요인에 1점(매우 좋음)부터 4점(매우 안 좋음)까지 점수를 부여하고 아래의 공식을 활용하여 변화인식에 대한 수준을 판단할 수 있도록 점수를 산정한다.

$$DICE \text{ 점수} = D + (2 * I) + (2 * C1) + C2$$

DICE 점수는 최소 7점에서 최대 28점까지 나올 수 있으며, 점수에 따라 7점과 14점 사이는 성공지역(win zone), 14점과 17점 사이는 우려지역(worry zone), 17점 이상은 재난지역(woe zone)으로 분류된다. 3가지 분류에 대한 설명은 다음과 같다.

표 7. DICE영역 분류체계
Table 4. Categories of DICE zone

구분	설명
성공지역 (win zone)	경영진의 개선에 대한 지원과 개선주체의 역량이 높아 개선의 성공확률이 높음
우려지역 (worry zone)	경영진의 개선의지 및 개선주체의 역량이 중간 정도 이며 개선의 성공을 확신할 수 없음
재난지역 (woe zone)	경영진의 개선의지 및 개선주체의 역량이 부족하며, 조직원의 거부감이 높아 개선성공 확률이 매우 낮음

아래의 그림은 DICE 점수와 225개의 실제 조직변화 프로젝트의 결과를 보여주고 있다. 매우 성공적인 프로젝트의 DICE 점수는 7에서 14점 사이에 분포하고 있고, 이 점수 구간 성공지역이다. 14점과 17점사이의 프로젝트는 성공지역보다 예측하기 어렵고 우려지역에 해당한다. DICE 점수가 17점 이상인 재난지역에 분포하는 프로젝트들은 일반적으로 프로젝트 성과가 좋지 않음을 알 수 있다. 그림위에 표시된 숫자는 225개 프로젝트 중에서 해당 점수에 해당하는 프로젝트의 개수를 의미한다.

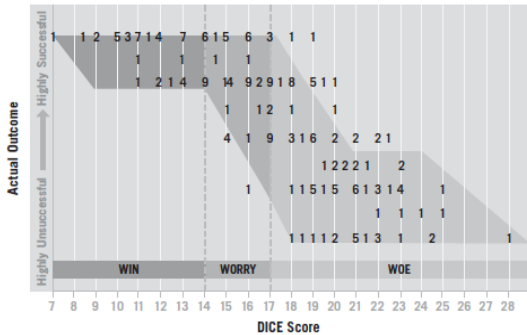


그림 4. 255개 변화 프로젝트의 DICE 점수(20)
Fig. 4. DICE® scores of 225 change initiatives

SW공학분야에 DICE 모델을 활용분야를 살펴보면, 2000년 중반 등장한 새로운 개발 방법론인 Agile 방법론의 도입의 타당성 입증에 의해 활용되었다. 특히, DICE 모델을 활용하여 소프트웨어 개발자뿐만 아니라 경영진을 설득하기 위한 논리로 활용되고 있다. Bartlomiej & Geoffrey(35)는 전통적인 개발 방법론인 폭포수 모델을 기반으로 소프트웨어를 개발하면서 남지연으로 어려움을 겪고 있는 소프트웨어 개발 조직에 Agile 방법론 도입의 타당성을 제시하기 위해 DICE 모델을 활용하였다. Madeyski and Biela(34)도 8명의 개발자로 이루어진 중/소기업이 eXtreme Programming을 도입의 타당성을 제시하는 프로젝트에 DICE 프레임워크를 활용하고 있다.

III. 데이터 수집 및 측정지표

1. 데이터 수집

본 연구에 사용된 데이터는 정보통신산업진흥원의 SW공학센터에 수행되는 SW공학수준조사를 통해 수집되었다. 조사기간은 2011년 9월에 11월에 걸쳐 약 3개월에 걸쳐 수집된 250개 프로젝트 중에서 144개 프로젝트 데이터가 활용되었다.

설문 조사 및 분석대상은 SW개발 프로젝트이고, 설문 응답자는 해당 프로젝트를 수행하는 프로젝트 관리자(Project Manager)나 프로젝트의 리더(Project Leader)들로 구성되었다. SW 개발 프로젝트의 유형은 IT 서비스(50.5%), 패키지 SW (22%) 및 임베디드 SW(27.5%) 프로젝트로 구성되었다.

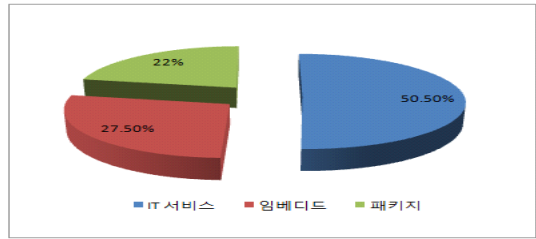


그림 5 SW 개발 프로젝트 유형
Fig. 5. Categories of SW development projects

2. SW공학수준 측정지표

2.1 프로세스 수준 지표

프로세스수준 지표는 SW개발 수행 시 조직 차원에서 필요한 활동들의 이행 수준과 원활한 이행을 위하여 필요한 내재화 수준의 확인을 목적으로 SW개발 수행 시 조직 차원에서 필요한 활동들의 이행 수준을 확인하기 위한 프로세스 이행수준 지표와 필요한 활동들이 조직 내 내재화하기 위하여 필요한 사항들의 수준을 확인하기 위한 프로세스내재화수준 지표로 구성되었다. 즉, SW 개발시 조직차원에서 필요한 활동들을 실제로 수행하고 있는지와 이러한 프로세스의 이행활동이 일회성이 아니라 지속적으로 수행될 수 있도록 공식적인 규정이나 해당활동들을 직접으로 수행하고, 관리할 수 있는 조직이나 인력이 존재하고, 이들 활동을 위해 적절한 예산이 지원되는지를 확인하는 지표들로 구성되었다. 프로세스 수준 측정지표는 18개 프로세스 영역에 대해 총 145개의 문항으로 구성되었다.

표 8 프로젝트 계획 프로세스의 측정내용
Table. 5 The measure of project planing process

프로세스	측정 내용
프로젝트 계획	프로젝트 범위 견적 이행여부
	작업산출물 및 작업 속성 견적 이행여부
	프로젝트 개발방법론 정의 여부
	노력 및 비용 견적 이행여부
	예산 및 일정수립 이행여부
	프로젝트 위험요소 식별 이행여부
	데이터 관리 계획 수립 이행여부
	프로젝트자원계획수립 이행여부
	필요지식 및 기술 확보 계획수립 이행여부
	관련 이해관계자 참여계획 수립 이행여부
	프로젝트 계획 수립 이행여부
	프로젝트에 영향을 주는 타 계획 검토 이행여부
	작업 및 자원 수준조정 이행여부
	계획에 대한 공동합의 확보 여부

2.2 인력 수준 지표

원활한 SW개발을 위하여 필요한 인력 보유, 지원 조직의 보유, 인력 육성을 위한 교육 수준 등 인력 측면에서 필요한 사항이 제대로 갖추어 졌는지를 확인하기 위한 지표로 국내 SW기업의 조직 및 인력육성에 대한 현황을 이해할 수 있으며, SW개발을 적절히 수행하기 위하여 필요한 조직의 역할, 인력, 교육 등을 측정하였다. 이를 위해 전문지원조직체계, 사내전문가 보유여부, 프로젝트 수행 역할체계, 조직원 역량 강화, 조직원 인력운영 등 5개 영역에 71개 세부 측정항목으로 구성되었다.

예들들어 전문 지원 조직 체계의 측정항목을 살펴보면, 총 9개의 측정항목으로 구성되었으며, 내용은 아래와 같다(표 6 참조).

표 9 전문지원조직체계의 측정항목
Table. 6 The measure of professional support system

구분	측정 내용
전문 지원 조직 체계	개발관련 지원 전문조직 보유여부
	테스팅 지원 전문조직 보유여부
	프로젝트 관리 전문조직 보유여부
	건설 지원 전문조직 보유여부
	업체(외주업체 포함)관리 조직 보유여부
	품질보증 전문조직 보유여부
	데이터 측정 및 분석 지원조직 보유여부
	프로세스 개선 전문 조직
교육훈련 전문조직 보유여부	

2.3 기술 수준 지표

기술수준 지표는 SW개발 기업의 조직원들이 SW개발을 제대로 수행하기 위하여 필요한 인프라(프로세스 체계, 자동화 시스템 및 툴 보유)의 구성과 활용 여부, 프로세스 자산 및 조직 정보 관리 활용 정도 등으로 특정 기술에 대한 수준이 아닌 SW개발에 있어서 기본적으로 갖추어지거나 적용되어야 하는 기술(인프라, 체계, 기법, 시스템, 툴 등) 수준을 확인하기 위한 지표이다. 국내 SW기업의 기본 인프라(프로세스, 자동화 시스템, 툴, 정량적 데이터 체계 등), 개발에 필요한 기본 기법 및 표준 등으로 구성되었다.

기술수준 측정항목 중에서 시스템의 인프라의 측정항목을 구체적으로 살펴보면, '조직 표준 프로세스 및 산출물의 통합 시스템 보유 여부'를 포함한 7개의 하위 측정지표로 구성하였다. 주로 조직의 표준 프로세스 자산을 통합적으로 관리할 수 있는 시스템이나 프로젝트관리와 관련 시스템이나 도구들이 구비되었는지를 측정하는 항목으로 구성되었다

표 10 시스템 인프라의 측정내용
Table. 7 The measure of system infrastructure

구분	측정내용
시스템 인프라	조직 표준 프로세스 및 산출물의 통합 시스템 보유 여부
	프로젝트 관리 시스템 보유여부
	공수 수집관리 시스템 보유여부
	형상관리 툴 및 시스템 보유여부
	모델링 전용툴 보유여부
	요구사항 관리툴 보유여부
	테스팅 관련 시스템 보유여부

2.4 DICE 지표

SW 프로세스 개선을 위한 조직의 변화의지는 DICE 모델을 기반으로 SW 프로세스 개선활동에 대한 공식적인 검토 활동 주기인 기간(Duration), SW프로세스 개선을 위해 투입되는 인력이나 팀의 역량을 의미하는 성실성(Integrity), SW 프로세스 개선에 대한 의지나 인식수준을 의미하는 조직 몰입(Commitment), SW 프로세스 개선을 위해 조직원들 자발적으로 추가적인 업무를 수용하는 정도를 의미하는 노력(Effort)을 측정하였다.

표 11 DICE 측정항목
Table 8. The measure of DICE

구성요소		설명
Duration (기간)	D	SW 프로세스 개선활동에 대한 공식적인 검토활동 주기
Integrity (성실성)	I	SW 프로세스 개선활동을 주도하고 있는 인력 또는 팀의 역량과 투입되는 공수의 정도
Commitment (조직몰입)	C	C1: 경영진의 프로세스 개선의지 C2: 프로세스 개선에 대한 현장조직(개발자/관리자)의 인식 수준
Effort (노력)	E	SW 프로세스 개선을 위해 현재의 업무에 어느 정도의 추가 투입공수가 발생할 것인지에 대한 예측

IV. 데이터 분석방법

본 연구에서는 DICE 수준에 따른 프로젝트의 SW공학점수를 분석하기 위해 일원분석을 실시하였다. 또한 수집된 자료의 무작위 추출된 자료가 아니기 때문에 데이터의 모집단 추정을 위해 붓스트랩 분석방법을 활용하였다.

1. 일원분산분석

DICE 수준에 따른 SW공학수준점수의 차이를 검증하기 위해서, 두 개 이상 다수의 집단 간 평균의 차이를 검증하는 통계분석 방법으로, 독립변수가 하나일 경우에 사용하는 일원 분산분석을 실시하였다. 즉, 3개의 DICE 수준에 따라 SW공학수준점수의 평균이 다른지를 검증하는 것이다.

분산분석은 독립변인(처치효과)과 종속변인 사이의 일원 분산의 통계적 모형은 다음과 같다.

$$Y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$$

Y_{ij} 는 i 개의 표본 집단과 j 개의 관찰치를 의미하고, α_i 는 i 개의 표본 집단의 효과를, ε_{ij} 는 i 개의 표본 집단과 j 개의 관찰치의 오차를 의미한다[23].

2. 붓스트랩 분석

Efron [24][25][26][27]의 붓스트랩(bootstrap)은 컴퓨터 기반의 비모수적 방법으로 샘플 추정치의 정확성을 평가하기 위한 컴퓨터 기반 통계기법이다. 이 기법은 매우 단순한 방법을 활용하여 거의 모든 통계적 표본의 분포를 추정할 수 있게 해주는 기법이다[27].

Efron & Tibshirani[26]는 정확한 신뢰구간 추정을 위해서는 최소 1,000개의 붓스트랩 표본이 적절하다고 제안하였다. 수집된 SW공학점수 데이터가 전체 국내 SW개발 기업을 대상으로 무작위 추출된 것이 아니기 때문에 분석결과의 안정성(stability)을 검증하는 작업이 필요하다. 이를 위해 본 연구에서는 수집된 데이터에서 1000번 이상 복원 추출하는 스트랩 표본 재추출 방법을 활용하였다.

원 수집자료의 분석을 통해 산출된 통계 값들이 경험적 신뢰구간의 중앙에 위치하면 안정적인 것으로 분류된다. 즉, 경험적 신뢰구간의 상/하위 경계에 존재하지 않아야 한다는 것을 의미한다. 이러한 안정성 편향(stability bias)을 평가하는 식은 다음과 같다.

$$Bias = \frac{\sum_{b=1}^B t_b^*}{B} - \hat{\theta}$$

t_b^* 는 b 번째 샘플의 통계 값이고, b 는 1, ..., B를, $\hat{\theta}$ 는 원

수집자료에서 분석된 통계 값을 의미한다. 편향의 정도는 B 번의 반복을 통해 나온 분포의 표준오차(standard error)를 기반으로 평가되며, 이때 표준오차는 다음과 같이 계산된다.

$$SE = \sqrt{\frac{\sum_{b=1}^B (t_b^* - \bar{t}^*)^2}{B-1}} \quad \text{단, } \bar{t}^* = \frac{\sum_{b=1}^B t_b^*}{B}$$

만약 편향(bias)이 표준오차에 비해 크다면, 안정성에 문제가 있는 것이다. 안정성을 판단하는 기준(criteria)은 편향의 절대값이 표준오차의 1/4보다 적다면 편향은 무시될 수 있다[27].

붓스트랩 방법론은 실증적 SW공학연구에서 이전부터 활용되어왔다. El-Emam과 Garro[29]는 포획/재포획(capture-recapture)방법을 활용하여 ISO/IEC 15504 심사 횟수를 추정하였다. Jung과 Hunter[30]는 개별 ISO/IEC 15504 프로세스의 능력 수준(capability level)의 신뢰구간을 계산하는데 붓스트랩을 활용하였으며, Jung과 Goldenson[32]은 SW-CMM의 내적 일관성(Internal consistency)에 대한 안정성을 평가하기 위해 붓스트랩을 활용하였다.

V. 실증적 분석

1. 표본의 특성

본 연구는 국내에서 수행된 SW 개발 프로젝트를 대상으로 DICE 수준에 따라 SW 프로세스 개선의 효과를 정량적으로 보여주는 SW공학수준점수가 달라지는 것을 살펴보는 것이다. DICE 수준별 분포를 살펴보면, 총 255개 프로젝트 중 Win 수준이 75개 Worry 수준이 55개, 마지막으로 Woe 수준이 125개로 나타나고 있다.

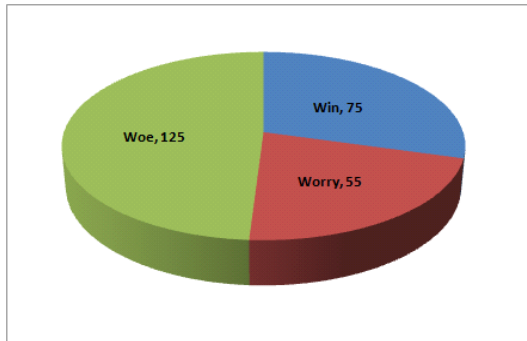


그림 6 DICE 수준별 분포
Fig.6. The Distribution of DICE Level

DICE 영역별 SW 공학수준점수의 평균을 살펴보면, Win 영역의 SW공학수준점수가 72.5점, Worry 영역점수가 62.6 점, Woe 영역이 54.8로 나타나고 있다. 이는 변화의지가 강한 프로젝트일수록 SW 공학수준점수가 높은 것으로 나타나고 있다.

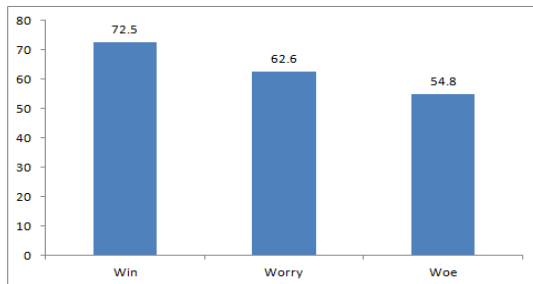


그림 7 DICE영역별 SW공학수준점수
Fig. 7 SW engineering level score by DICE Level

2. 실증적 분석결과

데이터 조사를 통해 수집된 255개 SW 개발 프로젝트의 데이터를 대상으로 수행된 일원분산분석의 결과를 살펴보면, F값의 유의확률 값이 0.015 로 유의수준 0.05 보다 작은 것을 알 수 있다. 이는 유의 수준 0.05% 수준에서 SW공학수준별 품질비용이 같지 않다는 것을 의미한다.

SW공학수준별로 SW품질비용이 다르다는 것을 보여주는 일원분산분석 결과는 표 9와 같다. 그러나 분석에 사용된 자료가 임의 추출방식(Random sampling)으로 수집된 자료가 아니라 편이 추출되었다는 한계를 가지고 있다. 이러한 자료의 한계를 보완하고 분석결과에의 신뢰성을 확보하기 위해 붓스트랩 분석방법을 활용하여 모델의 안정성 검토와 일원분산분석을 실시하였다.

표 12. 분산분석결과
Table. 9 The results of ANOVA

	제공합	df	평균제곱	F	유의 확률
집단간	15332.505	2	7666.252	30.587	.000
집단내		252	250.708		
합계		254			

* 유의수준 (0.00)

모델의 안정성을 검증하기 위해 SW공학점수 항목에 대해 1000번의 재표본 추출을 통해 1000개의 SW공학수준점수 세트를 추출하고, 이를 기반으로 붓스트랩 샘플의 SW공학수준점수의 평균(boot.means)과 표준오차(Standard Error: SE)값을 구하였다. 붓 스트랩 샘플의 SW공학수준 평균과 표본(수집된 데이터) 평균과 편차와 표준편차에 대한 분석결과 는 아래의 표와 같다.

표 13. 붓스트랩 편차와 표준오차
Table.10. Bias and standard error of bootstrap

표본 평균 (original)	편차 (bias)	표준편차 (std. error)
62.37	-0.02	1.10

SW공학수준 점수에 대한 붓스트랩 표본의 평균값 (boot.means)는 62.39로 분석되었으며, 이는 최초 수집된 자료의 SW공학수준 점수의 평균은 62.37과 거의 유사한 수준으로 나타났다. 즉, 원래 표본의 평균과 복원추출을 통해 만들어진 표본의 평균간의 차이가 거의 없다는 것을 의미하며, 두 표본이 거의 동등하다는 것을 보여주고 있다. 표본평균과 붓스트랩 평균의 편차는 -0.02이며, 이 값은 표준오차 (1.10)의 1/4인 0.275보다 작기 때문에 무시될 수 있는 작은 값이라는 것을 알 수 있다(그림 7참조).

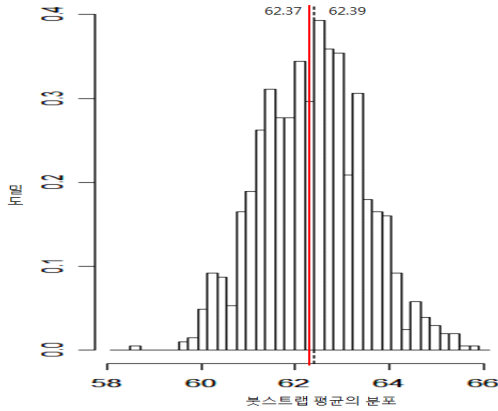


그림 8. 부스트랩 평균값의 분포
Fig. 8. The distribution of bootstrap means

수집된 자료를 SW공학수준별로 분류하고, 개별 SW공학 수준별로 1000번의 복원 추출한 샘플을 기반으로 DICE 수준별로 SW공학수준 값이 다른가를 검증하기 위한 F 검증량을 구하기 위한 새로운 Fstar 값들의 분포를 구하였다(그림 8 참조). 1000번의 부스트랩을 통해 산출된 F값의 분포를 기준으로 유의수준 0.5%수준의 F값(2, 255)은 4.47로 분석되고 있다. 이는 원래 일원분산 분석의 검증통계량 F값(2,255) 4.68과 다소 차이를 보이고 있다. 수집된 데이터의 F 값은 30.587은 부스트랩을 통해 분석된 F값 검증통계량인 4.47이나 일반적인 분산분석의 F값 검증통계량인 4.68보다 크다는 것을 알 수 있다. 일반적인 F검증통계량과 부스트랩 F 검증통계량이 다르지만 수집된 데이터의 F분석값 30.587는 DICE 수준별로 SW공학수준점수가 같다는 귀무가설을 기각하고 있다. 즉, DICE 수준에 따라 SW공학수준점수의 차이가 존재한다는 것을 알 수 있다.

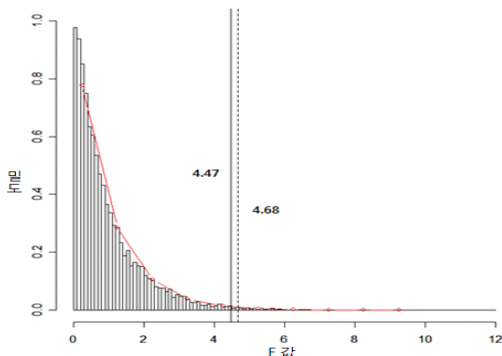


그림 9. 부스트랩 F값의 분포
Fig. 9. The F values distribution of bootstrap

분산분석에서 귀무가설을 기각하는 경우에, 모평균들이 모두 같지 않다는 결론을 얻을 수 있을 뿐이며 구체적으로 어떤 집단간에 유의한 차이가 존재하는지 것인지는 알 수 없다. 따라서 이를 알아보기 위한 추가적인 검정이 필요하게 되는데, 이 때 두 집단씩(예: Win 영역 평균과 Worry 영역 평균)과 묶어서 평균의 차이가 있는지 혹은 수준별 수준의 선형 결합 대비(contrast)차이가 있는지 검증해야 하는데 이를 사후 검증이라고 한다.

분산분석의 사후검증방식은, Duncan, Turkey, Sheffee 방식이 있으며, 이 중에서 Sheffee방식이 가장 엄격한 것으로 알려져 있다. 본 연구에서는 3가지 방식 모두를 적용하였다. 이 중 Turkey, Scheffe의 분석결과를 제시하였다. DICE 영역별 쌍대비교의 분석결과는 세 가지 쌍대 비교 값들의 차이가 5% 수준에 유의한 차이가 있음을 보여주고 있다(표 11 참조).

표 14. 사후검증 분석결과
Table 11. Results of post hoc analysis

변화인식수준(I)	공학수준(J)	평균차(I-J)	유의확률	
Turkey	win	worry	10.94	.001
		woe	30.70	.000
	worry	win	-10.93	.001
		woe	19.76	.000
Scheffe	woe	win	-30.70	.000
		worry	-19.76	.000
	win	worry	10.93	.002
		woe	30.70	.000
Scheffe	worry	win	-10.93	.002
		woe	19.76	.000
Scheffe	woe	win	-30.70	.000
		worry	-19.76	.000

V. 결론

소프트웨어 프로세스 개선은 소프트웨어 기업의 공학적인 측면뿐만 아니라 관리적인 관점에 영향을 미치는 복잡한 조직 변화의 과정이다. 따라서 소프트웨어 프로세스 개선을 추진하면서 발생하는 변화 이슈를 다루는 분야에 대한 관심이 높아지고 있다. 본 논문은 SW 프로세스 개선사업의 성공요인을 DICE 프레임워크를 통해 조직의 변화의지와 SW 프로세스 개선의 성과를 살펴보았다. 이를 위해 국내 SW개발 프로젝트를 대상으로 DICE수준과 SW공학수준점수 자료를 수집하였다.

수집된 자료를 바탕으로 국내 SW개발 프로젝트 DICE 수준을 살펴보고, DICE수준에 따라 SW공학수준점수의 차이가 존재하는지를 분석하였다. DICE 수준에 따른 SW공학수준의 점수를 분석해 결과, DICE영역에 따라 SW공학수준점수의 차이가 존재하는 것으로 나타났다. 이와 더불어 DICE 수준이 높은 조직일수록 SW공학수준점수가 높은 것으로 분석되었다.

이는 소프트웨어 프로세스 개선을 성공적으로 추진하기 위해서는 변화관리의 하드(hard) 요소인 기간(Duration), 성실성(Integrity), 조직몰입(Commitment) 과 노력(Effort) 이 중요한 요인임을 보여주고 있다. 이는 변화관리의 하드 요인들에 비해 측정과 관리가 용이하고, 변화의 상태를 가시적으로 확인할 수 있기 때문에 판단된다.

이와 같은 결과는 현재 SW공학 경쟁력이 취약한 중소기업들을 대상으로 지원되고 있는 지원 사업들에 대한 변화의 필요성을 보여주고 있다. 즉, 기술 중심의 기업지원정책과 더불어 조직의 변화 관점이라는 장기적인 관점이 필요하고, 단기적인 성과 위주보다는 장기적인 지원이 필요하다는 것을 보여주고 있다. 특히 인력이나 자금 측면에서 대기업들에 비해 상대적으로 열세인 중소기업들의 현실을 감안하면 더욱 체계적인 접근방식이 필요함을 알 수 있다. 즉, 이를 위해 지원 대상 업체를 선정하는 단계 혹은 지원사업의 시작단계에서 대상 기업들에 대한 사전조사를 통해 해당 기업들이 스스로 인식하는 문제와 실제 조직이 직면하고 있는 문제를 분석하고, 해당 문제에 대처할 수 있는 지원체계를 고려해 보아야 할 것으로 판단된다. 또한 이는 기존의 소프트웨어 프로세스 개선을 단지 기술적인 측면에서 통제(control)라는 측면에서 관리에서 벗어나 조직차원의 변화관리도 동시에 고려해야 할 것으로 판단된다.

일반적인 실증적인 연구들과 마찬가지로 본 연구도 몇 가지 한계점을 가지고 있기 때문에 이를 극복하기 위한 추가적인 연구가 반드시 필요하다. 첫째, 데이터 수집단계에서 철저한 검증을 거쳤음에도 불구하고, 수집 및 분석에 활용된 데이터가 자동화된 도구나 시스템을 통해 직접 측정된 데이터가 아니라 응답자들이 스스로 작성하여 제출한 자료이기 때문에 자료에 다소간의 편차가 있을 수 있다는 점이다. 둘째, 본 연구의 SW공학수준은 CMMI 모델의 KPA를 기반으로 CMMI Level3까지의 데이터를 기반으로 분석이 수행되었다. 이로 인해서 모집단에 따라 표본이 구분되는 범위 제한(range restriction)의 문제점이 발생한다. 따라서 향후 CMMI Level 4, Level 5를 포함한 표본에 대한 추가 연구와 본 연구의 결과를 뒷받침할 수 있도록 SW공학수준과 비용, 생산

성, 품질 및 고객 만족도와 같은 성과변수들을 포함한 추가연구가 진행되어야 한다.

참고문헌

- [1] CMMI, "CMMI Product Team CMMI for Development, Version 1.3," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2010.
- [2] Gibson, D.L., Goldenson, D.G and Kost, K, "Performance Results of CMMI-Based Process Improvement," Technical Report CMU/SEI-2006-TR-004, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2006.
- [3] Kuilboer, J. P., and Ashraf, N., "Software process and product improvement: an empirical assessment," Information & Software Technology, Vol. 42, No. 1, 2000, pp.27~34
- [4] Cattaneo, F., Fuggetta, A., and Sciuto, D., "Pursuing coherence in software process assessment and improvement," Software Process Improvement and Practice, Vol. 6, No. 1, 2001, pp. 3-22.
- [5] Staples, M., and Niazi, M., "Systematic review of organizational motivations for adopting CMM-based SPI," Information and Software Technology, Vol. 50, No. 7, 2008, pp. 605-620.
- [6] Iversen, J. H., Mathiassen, L., Nielsen, P. A., "Managing Risk in Software Process Improvement: An Action Research Approach," MIS Quarterly, Vol.28, No.3, 2004, pp.395-433.
- [7] Ngwenyama, O., and Nielsen, P., "Competing values in software process improvement: an assumption analysis of CMM from an organizational culture perspective," IEEE Transaction on software Engineering Vol. 50, No. 1, 2003, pp. 100-112.
- [8] Standish Group, "Chaos: A Recipe for Success," tech. report, Standish Group Int'l, 2009.
- [9] Rainer, A. & Hall, T., "An analysis of some core studies of software process improvement",

- Software Process Improvement and Practice, Vol. 6, No. 4, 2001, pp. 169-187.
- [10] Kulpa, M. K., & Johnson, K. A., "Interpreting the CMMI: A Process Improvement Approach", Boca Raton, Florida: Auerbach Publication, 2003.
- [11] Piderit, S. K., "Rethinking resistance and recognizing ambivalence: a multidimensional view of attitudes toward an organizational change," *Academy of Management Review*, Vol. 25, No. 4, 2000, pp. 783-795.
- [12] Nasir, M. H., Ahmad, R. and Hassan, N. H., "Resistance Factors in the Implementation of Software Process Improvement Project in Malaysia", *Journal of Computer Science*, Vol. 4, No. 3, 2008, pp. 211-219.
- [13] Ashrafi N., "The impact of software process improvement on quality in theory and practice," *Information & Management*, Vol. 40, No. 7, 2003, pp. 677 - 690.
- [14] Baddoo, N. and Hall, T., "De-motivators for software process improvement: an analysis of practitioners' views," *The Journal of Systems and Software*, Vol. 66, No. 1, 2003, pp. 23 - 33.
- [15] Hansen, B., Rose, J. and Tjørnehøj, G., "Prescription, description, reflection: the shape of the software process improvement field," *International Journal of Information Management*, Vol. 24, No. 6, 2004, pp. 457 - 472.
- [16] Ravichandran, T. and Rai, A., "Total Quality management in information systems development: key constructs and relationships," *Journal of Management Information Systems*, Vol. 24, No. 3, Winter 1999 - 2000, pp. 81 - 415.
- [17] Aaen, I., Arendt, J. Mathiassen, L., and Ngwenyama, O., "A conceptual MAP of software process improvement," *Scandinavian Journal of Information Systems*, Vol. 13, 2001, pp.79 - 99.
- [18] Niazi, M., Wilson, D. and Zowghi, D., "A framework for guiding the design of effective implementation strategies for software process improvement," *Journal of Systems and Software*, Vol.78, No. 2, 2005, pp. 204-222.
- [19] Nielsen, P.A. and Nørbjerg J., "Software process maturity and organizational politics," in: *Proceedings of the IFIP TC8/WG8.2 Working Conference on Realignment Research and Practice in Information Systems Development: The Social and Organizational Perspective*, 2001, pp. 221 - 240.
- [20] Sirkin, H. L., Keenan, P. and Jackson, A., "The Hard Side of Change Management," *Harvard Business Review*, Vol. 83, No. 10, 2005, pp 108 - 118
- [21] Abrahamsson, P. "Commitment in software process improvement - in search of the process System Sciences," *Proceedings of the 35th Annual Hawaii International Conference on 2002*.
- [22] SEMA, *Process Maturity Profile: Software CMM 2004, Mid-year Update*, Software Eng. Inst., Caregie Mellon Univ., 2004.
- [23] Kerr, M. K., Martin, M. and Churchill, G. A., "Analysis of Variance for Gene Expression Microarray Data," *Journal of Computational Biology*, Vol. 7, 2000, pp. 819-837.
- [24] Efron, B., "Bootstrap Methods: Another Look at the Jackknife," *The Annals of Statistics*, Vol.7, 1979, pp.1-26.
- [25] Efron, B., "The Jackknife, the Bootstrap and Other Resampling Plans," *Society for Industrial and Mathematics CBMS-NSF Monographs*, Vol. 38, 1982.
- [26] Efron, B. and Tibshirani, R., "Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Measures of Statistical Accuracy," *Statistical Science*, Vol.1, No.1, pp.54-75, 1986.
- [27] Efron, B. and Tibshirani, R., "An Introduction to the Bootstrap," Chapman and Hall, 1994.
- [28] Varian, H. R., "Bootstrap Tutorial", *Mathematica Journal*, Vol.9, No.3, 2004.
- [29] El-Emam, K. and Garro, I., "Estimating the

Extent of Standards Use: The Case of ISO/IEC 15504," Journal of Systems and Software, Vol.53, No.2, 2000, pp.137-143.

[30] Ho-Won, Jung and Hunter, R., "An Evaluation of the SPICE Rating Scale with Regard to the Internal Consistency of Capability Measurement," Proceedings of the Second International SPICE Conference, 2002, pp.105-115.

[31] Herbsleb, J. and Goldenson, D., "A Systematic Survey of CMM Experience and Results," 18th International Conference on Software Engineering(ICSE-18), Berlin, Germany, 1996, 25th-26th March.

[32] Ho-Won, Jung, & Goldenson, D., "The Internal Consistency of Key Process Areas in the Capability Maturity Model® (SW-CMM®) for Software," Carnegie Mellon University/Software Engineering Institute, CMU/SEI-2002-TR-037, 1994.

[33] Ziolkowski, B. and Drake, G., "Rolling the DICE for Agile Software Projects," Proceedings of XP, 2006, pp.114-122.

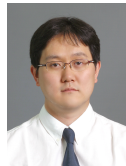
[34] Madeyski, L. and Biela W., "Capable Leader and Skilled and Motivated Team Practices to Introduce eXtreme Programming," CEE-SET, 2007, pp. 96-102.

[35] Bartłomiej, Z. and Geoffrey, D., "Rolling the DICE for Agile Software Projects," XP'06 Proceedings of the 7th international conference on Extreme Programming and Agile Processes in Software Engineering, 2006, pp. 114-122

저 자 소개



김 승 권
 2009: 고려대학교
 경영학과 경영학박사(MIS)
 현 재: 정보통신산업진흥원
 소프트웨어공학센터 수석
 관심분야: SW 프로세스, SW품질,
 정보 시스템 성과평가
 Email : sgkim@nipa.kr



조 성 현
 2002: 한국외국어대학교
 경영정보대학원 MIS 전공(석사)
 2010: 한국외국어대학교
 대학원 박사(MIS)수료
 현 재: 정보통신산업진흥원
 소프트웨어공학센터/수석
 Email : shjo@nipa.kr



윤 종 수
 2000 : 고려대학교
 경영학과 경영학박사(MIS).
 현재 : 강남대학교 경영대학 부교수.
 관심분야 : e-Business/EC 기획,
 정보시스템 활용 및 평가,
 모바일 인터넷 서비스,
 CRM/BSC 응용
 E-mail : jongsoo@kangnam.ac.kr