

CMMI 기반 IT 프로젝트를 위한 위험관리 프로세스 개선에 관한 연구

장종기*, 이송희**, 최진영*

*고려대학교 컴퓨터정보통신대학원

**고려대학교 컴퓨터정보통신연구소

cosmos384@korea.ac.kr, shlee@formal.korea.ac.kr, choi@formal.korea.ac.kr

A Study on Risk Management Process Improvement for IT Project Based on CMMI

Jong-Ki Jang*, Song-Hee Lee**, Jin-Young Choi*

*Graduate School of Computer Information & Communication, Korea University

**Institute of Computer Information and Communications, Korea University

요 약

IT 프로젝트는 많은 불확실성을 내포하고 있다. 이러한 불확실성은 프로젝트의 성공적인 수행에 좋지 않은 영향을 미치게 된다. 이처럼 악영향을 미치는 근본적인 이유는 프로젝트가 가지고 있는 잠재적 위험(Risk)이 중요한 원인이라 할 수 있다. CMMI-DEV(Capability Maturity Model Integration for Development, version 1.2)모델을 기반으로 IT 프로젝트를 수행하는 많은 조직에서 소프트웨어 개발 시 실제 CMMI-DEV 모델을 적용하여 IT 프로젝트의 위험관리를 수행하고 있다. 그러나 대다수 기업에서 위험을 관리하기 위한 프로세스의 범위(영역)를 선정하는 것에는 많은 노력을 기울이지만 단위 프로세스별로 어떠한 방법(How)과 도구 및 관리기법을 사용하여 위험을 정량적으로 관리하고 완화시킬지에 대한 구체화된 노력은 미흡한 것이 사실이다. 본 논문에서는 CMMI-DEV 모델의 RSKM(Risk Management) 프로세스와 PMBOK(Project Management Body of Knowledge, 4th Edition)의 위험관리 지식영역(Knowledge Area)을 프로세스의 지속적인 개선을 위하여 PDCA(Plan-Do-Check-Action) Cycle의 각 단계별 목적에 맞게 통합(Integration)시켜 IT 프로젝트를 위한 개선된 위험관리 프로세스를 제안하였다. 제안 프로세스의 개선효과를 검증하고 분석하기 위하여 측정지표를 제시하였으며, 개선 프로세스의 적용 전후 결과를 정량적으로 비교 및 분석함으로써 개선효과를 도출하였다.

Keywords : CMMI, CMMI-DEV, RSKM, PMBOK, PDCA

1. 서론

일반적으로 IT 프로젝트는 불확실성을 포함하는 미래에 대한 예측의 관리이다. 그러므로 불확실성에 기반한 위험이 존재하기 마련이다. 기업의 IT 프로젝트 조직에서 위험을 어떻게 잘 관리하여 이것을 제거 또는 완화하는가에 따라 프로젝트의 성패에 많은 영향을 줄 수 있다.

미래에 대한 관점에서의 위험은 발생하고 나서 대응하는 것보다 사전에 철저한 준비를 통해 관리하는 것이 효과적이다. 위험관리는 IT 프로젝트 라이프사이클 전체에 대하여 수행하게 되는데 특히 프로젝트 시작 시점에 위험요인을 얼마나 정확히 식별하고 잘 관리 하는가에 따라 전체 프로젝트 공정에 위험요인이 미치는 영향을 최소화 할 수 있다[1]. 이에 본 논문에서는 IT 프로젝트 수행 시 다수 기업이 참조하고 있는 CMMI-DEV 모델의 RSKM 프로세스를 개선한 새로운 프로세스를 제안하고자 한다. 프로세스를 개선함에 있어 프로세스들 간의 연관성을 이해하고 상호작용의 기본이 되는 PDCA Cycle을 프로세스 통합의 기준으로 정하였다[2]. 따라서 프로세스를 개선함에 있어 PDCA Cycle을 기준으로 하여 CMMI-DEV 모델의 RSKM 프로세스의 단위 프로세스별로 위험관리를 보다

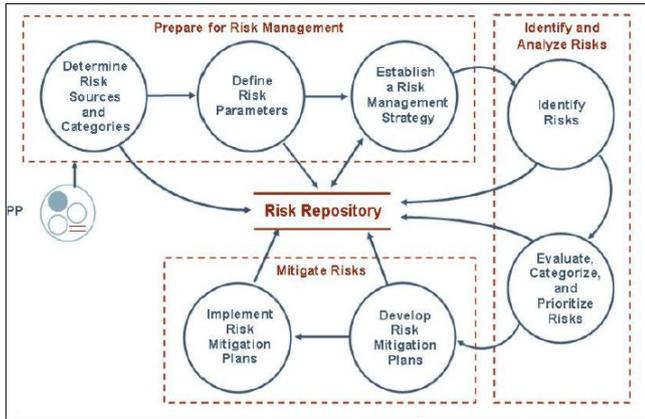
더 정량적으로 표현하기 위해 관련 지침서인 PMBOK에서 제시한 도구 및 관리기법을 적용하였다. 본 논문의 구성은 다음과 같다. 1 장의 서론에 이어 2 장에서는 본 논문에서 개선하고자 하는 CMMI-DEV 모델의 RSKM 프로세스 및 정량적 기법의 적용을 위한 PMBOK의 위험관리 지식영역의 구성에 대하여 연구하였고, 3 장에서는 프로세스간의 상호작용을 통한 지속적 개선을 위하여 PDCA Cycle을 기반으로 각 프로세스를 통합하여 개선된 위험관리 프로세스를 제시하였고, 4 장에서는 제안 프로세스에 대한 개선효과의 검증을 위해 측정지표를 제시하였으며, 개선 프로세스 적용 전후의 데이터 비교를 통해 결과를 분석하였다. 마지막으로 5 장에서는 본 연구의 결론에 대해서 기술하였다.

2. 관련연구

CMMI-DEV 모델의 RSKM 프로세스에 대한 구성과 프로세스 사이의 상호관계를 살펴보고 위험을 식별하고 분석하는데 기존 프로세스보다 더 정량적인 방법으로 산정하기 위해 PMBOK의 위험관리 지식영역에서 제시한 도구 및 관리기법에 대하여 연구하였다.

2.1 CMMI-DEV RSKM 프로세스

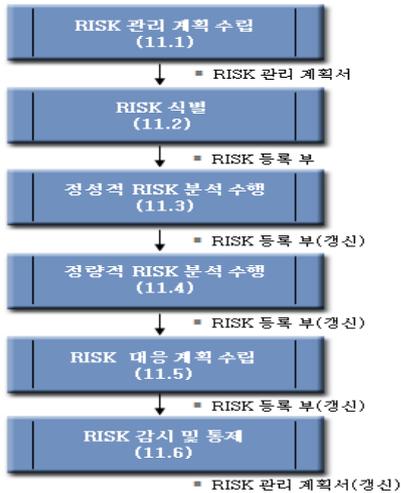
CMMI는 1993년 미 국방성의 후원으로 카네기멜론 대학의 소프트웨어 공학 연구소(SEI, Software Engineering Institute)에서 소프트웨어 프로젝트의 성공 가능성을 높이고 위험을 최소화하기 위하여 개발되었다[3]. CMM 모델 간의 상호 중복성 또는 구조적 차이 등 문제가 발생하여 CMM의 여러 모델을 하나로 통합한 CMMI 모델이 2001년 발표되었다. ISO/IEC 15504 표준과의 호환성을 갖춘 모델로 통합된 평가방법을 제공한다[4]. CMMI-DEV의 RSKM 프로세스는 3개의 SG(Specific Goal)와 7개의 SP(Specific Practice)로 구성되어 있으며 그림 1과 같이 표현할 수 있다[5].



(그림 1) CMMI-DEV RSKM 프로세스

2.2 PMBOK 위험관리 지식영역

PMBOK은 프로젝트 관리를 위하여 광범위한 범위의 프로젝트 관리 지식체계를 요약하고 체계적으로 정리하기 위해 미국의 PMI(Project Management Institute)에 의해 개발되었다[6]. 또한, 프로젝트 관리 지침서로 사용하되 필요에 따라 수정해서 사용할 수 있는 Good practice를 제공하고 있다. 전체적인 구성은 9개의 지식영역과 42개의 프로세스(Process)로 구성되어 있으며 그 중 위험관리 지식영역은 그림 2와 같이 표현할 수 있다[7].



(그림 2) PMBOK 위험관리 지식영역

3. 제안 프로세스

제안 프로세스는 IT 프로젝트를 수행함에 있어 프로젝트 내부에 잠재적으로 내포되어 있는 위험을 정확히 식별하고 효율적인 위험관리 전략을 수립할 수 있도록 구성하였다. 특히 프로젝트 초기 단계부터 전체 공정에 대하여 위험을 감시하고 통제할 수 있도록 프로세스의 지속적 개선을 위해 PDCA Cycle의 목적에 충실하였다.

3.1 제안 프로세스 구현 기준

제안 프로세스 구현의 단계별 적용 기준은 PDCA Cycle이며, 이것은 1930년경 Walter Shewhart에 의해 착안되었고, 그 후 W. Edwards Demming에 의해 채택되고 발전되었다. PDCA Cycle의 각 주기별 기능에 대한 정의는 표 1과 같다[8][10].

<표 1> PDCA Cycle의 주기별 기능 정의

PDCA Cycle	기능
계획(Plan)	프로세스 향상을 위한 이슈나 문제를 찾고 정의한다.
실행(Do)	정의된 이슈나 문제를 기반으로 데이터를 수집하고 데이터의 변화를 잘 표현한다.
점검(Check)	문제의 해결책에 대한 성과를 잘 이해하기 위해 데이터를 점검한다.
조치(Action)	결과를 기록하고 변화를 감시하며 변화의 지속이 유지되는지 결정한다.

본 연구에서 선택한 PDCA Cycle은 프로세스를 개발하는데 비교적 쉬운 구조를 이루고 있으며, 프로세스 상호작용의 기본이 된다. 그림 3과 같이 순환을 반복함으로써 지속적인 개선효과를 가져오는 장점이 있다[2][9].



(그림 3) PDCA Cycle

3.2 제안 프로세스 구현

제안 프로세스는 지속적인 프로세스 개선을 목적으로 PDCA Cycle을 기반으로 하고 있으며, CMMI-DEV 모델의 RSKM 프로세스 및 PMBOK의 위험관리 지식영역의 개별 프로세스를 프로세스별 도구 및 관리기법과 함께 표 1에 정의된 PDCA의 각 주기별 기능에 맞게 통합 하였다. 표 2는 CMMI-DEV RSKM 프로세스를 PDCA Cycle에 각각 병합한 결과이며, 표 3은 도구 및 관리기법을 포함한 PMBOK의 위험관리 지식영역의 개별 프로세스를 PDCA Cycle에 병합한 결과이다.

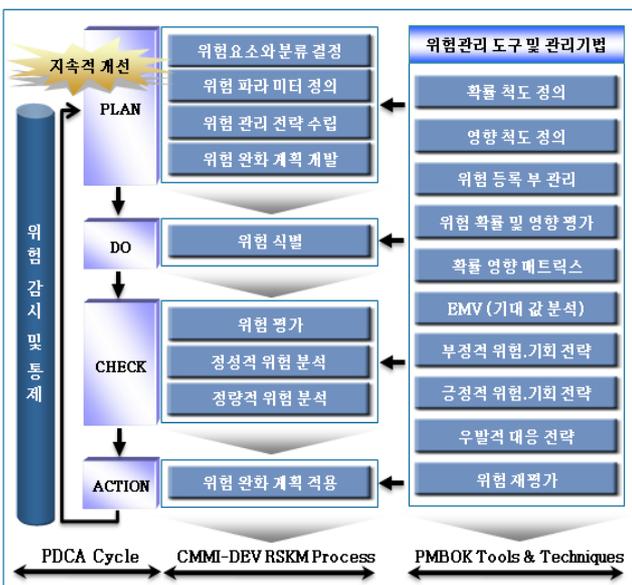
<표 2> CMMI-DEV RSKM 프로세스와 PDCA Cycle의 병합 결과

CMMI-DEV RSKM Process		PDCA Cycle
SG 1 위험관리 준비	SP 1.1 위험 요소와 분류를 결정	Plan
	SP 1.2 위험 파라미터 정의	Plan
	SP 1.3 위험 관리 전략 수립	Plan
SG 2 위험 식별과 분석	SP 2.1 위험 식별	Do
	SP 2.2 위험평가, 분류, 우선순위 결정	Check
SG 3 위험 완화	SP 3.1 위험 완화 계획 개발	Plan
	SP 3.2 위험 완화 계획 적용	Action

<표 3> PMBOK 위험관리 지식 영역과 PDCA Cycle의 병합 결과

PMBOK Risk Management Knowledge Area		PDCA Cycle
11 위험관리 지식영역	11.1 위험 관리 계획 수립	Plan
	11.2 위험 식별	Do
	11.3 정성적 위험 분석 수행	Check
	11.4 정량적 위험 분석 수행	Check
	11.5 위험 대응 계획 수립	Plan
	11.6 위험 감시 및 통제	Plan-Do-Check-Action

표 2, 표 3에서 병합된 각각의 결과에 대하여 PDCA Cycle의 Plan-Do-Check-Action의 각 주기에 맞게 프로세스를 통합하여 구성하였다. 특히, PMBOK의 위험관리 지식영역의 각 프로세스에서 사용하는 도구 및 관리기법을 통합하여 구성함으로써 IT 프로젝트의 위험관리를 위한 프로세스를 보다 더 정량적으로 바라볼 수 있도록 하였다. 본 연구에서는 위험관리를 위한 통합된 개선 프로세스를 그림 4와 같이 구성하였다. 여기서 PDCA Cycle의 순환적 구조는 화살표를 통하여 표현하였다.



(그림 4) 개선된 위험관리 프로세스

4. 개선효과 검증 및 분석

4.1 측정지표 선정

제안된 위험관리 프로세스의 개선 효과를 측정하기 위해서는 측정 대상이 되는 지표가 필요하다. 본 연구에서는 개선 프로세스의 검증에 위해 소프트웨어 프로세스 성숙도 평가를 위한 국제 표준인 ISO/IEC 15504에 근거하여 본 연구에 적합한 일부 카테고리를 참조하였다[1][11]. 선정된 측정지표별 의미를 정리하면 다음과 같다. 첫째, 위험관리 투입율은 위험관리를 위한 투입공수의 절감여부를 측정한다. 둘째, 위험 발생율은 위험완화 활동의 실패로 인해 현실화된 위험을 측정한다. 셋째, 결함 제거율은 프로젝트 진행 중 발견된 전체 결함 수 대비 실제 제거된 결함 수를 측정한다. 세 가지 측정지표 및 각각의 의미는 표 4와 같다.

<표 4> 측정지표 및 공식

측정지표(%)	측정의미	측정공식
위험관리 투입율	투입공수 절감 여부	(실제 위험관리 투입 MD / 계획된 전체 투입 MD) * 100
위험 발생율	위험 발생 감소 여부	(실제 현실화된 위험 수 / 식별된 전체 위험 수) * 100
결함 제거율	결함 제거 효과 여부	(실제 제거된 결함 수 / 발견된 전체 결함 수) * 100

4.2 검증 및 분석

제안 프로세스 검증을 위해 IT 프로젝트를 전문으로 수행하고 CMMI 레벨 3 인증을 취득한 K사의 IT 프로젝트중 3개(표기형태 : A, B, C)를 선정하여 자료를 수집하였고, 비교 대상으로 선정된 1개의 파일럿 프로젝트는 이전 3개의 프로젝트를 수행했던 동일 조직에 의해 개선된 위험관리 프로세스를 적용하여 진행되었다. 개선된 위험관리 프로세스를 적용한 결과 값과 비교가 쉽게 이루어지도록 개선 프로세스 적용 전 3개의 프로젝트에 대해서는 각 측정지표의 평균값을 계산하였다. 그 결과는 표 5와 같다.

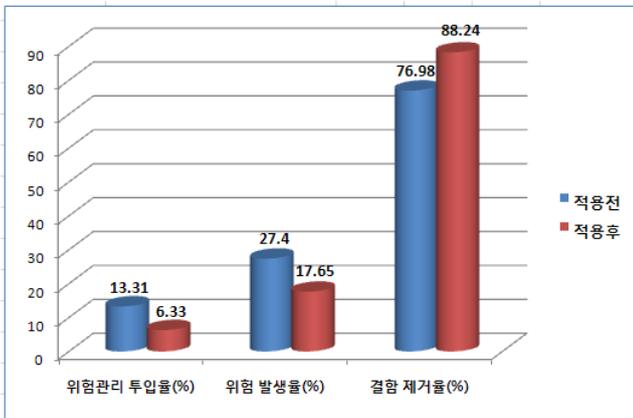
<표 5> 개선 프로세스 적용 전후 결과

측정지표(%)	측정공식	적용전 프로젝트			적용후 프로젝트
		A	B	C	파일럿
위험관리 투입율	실제 위험관리 투입 MD	22	16	24	5
	계획된 전체 투입 MD	153	122	193	79
	위험관리 투입율(%)	14.38	13.11	12.44	6.33
	평균(%)	13.31			6.33
위험 발생율	실제 현실화된 위험 수	7	5	9	3
	식별된 전체 위험 수	23	22	31	17
	위험 발생율(%)	30.43	22.73	29.03	17.65
	평균(%)	27.4			17.65
결함 제거율	실제 제거된 결함 수	19	16	35	15
	발견된 전체 결함 수	25	23	41	17
	결함 제거율(%)	76	69.57	85.37	88.24
	평균(%)	76.98			88.24

측정지표별 개선 프로세스 적용 전후의 차이를 보여주는 표 6과 그림 5의 결과를 분석해 보면 첫째, 프로젝트 전체 투입공수 대비 위험관리를 위해 투입된 공수비율에서 6.98% 감소하였으므로 개선 프로세스 적용 시 위험관리를 위해 필요로 하는 투입공수가 적게 필요함을 알 수 있다. 둘째, 식별된 전체 위험 수 대비 프로젝트 진행시 현실화된 위험 수의 비율은 9.75% 감소하여 위험 발생률이 감소하였다. 셋째, 발견된 전체 결함 수 대비 제거된 결함수의 비율에서는 11.26% 증가하여 결함 제거의 효과성이 좋아졌다.

<표 6> 측정지표별 개선 프로세스 적용 전후 차이

측정지표(%)	적용전(프로젝트 전체 평균)	적용후(파일럿)	차이	결과
위험관리 투입률	13.31	6.33	-6.98	감소
위험 발생률	27.4	17.65	-9.75	감소
결함 제거율	76.98	88.24	11.26	증가



(그림 5) 개선 프로세스 적용 전후 비교

5. 결론

IT 프로젝트는 불확실성을 가지고 수행되며, 이러한 불확실성으로 인해 발생하는 위험은 프로세스에 의해 적절히 관리되어야 한다. 위험을 적절하게 관리하기 위해서는 개별 프로세스 관리의 효율성을 높이고 위험완화의 목적을 달성하기 위해 위험관리를 위한 각 단계별 프로세스의 명확한 목적을 가지고 관리되어야 할 것이다. 본 논문에서는 IT 프로젝트를 수행하는 조직에서 CMMI-DEV 모델의 RSKM 프로세스를 적용하여 위험관리 수행 시 위험의 정량적 분석을 위해 PMBOK에서 제시하는 위험관리 지식영역의 도구 및 관리기법 적용하였고, 프로세스의 지속적인 개선을 유도하기 위해 PDCA Cycle의 각 주기에 통합하였다. 이렇게 통합된 제안 프로세스를 이용하여 프로젝트를 수행한 결과 “위험관리를 위한 투입 공수율”, “위험 발생률”, “결함 제거율”을 개선할 수 있었다. 본 논문은 결과적으로 프로세스 중심적인 CMMI-DEV 모델의 RSKM 프로세스와 프로젝트 관리중심이라 할 수 있는 PMBOK의 위험관리 지식영역에 포함된 도구 및 관리기

법을 상호 보완하고, PDCA Cycle의 개별 주기에 통합함으로써 CMMI를 기반으로 위험관리를 수행하는 조직에서 IT 프로젝트 수행 시 위험관리를 위한 프로세스를 개선하였다는데 의미가 있다.

참고문헌

- [1] 지식경제부 기술표준원 “ISO/IEC 15504(SPICE) 기반의 중소기업을 위한 소프트웨어 표준 프로세스”, 2007.
- [2] 염상봉, “품질 및 정보화 경영시스템 통합모델에 관한 연구”, 서울산업대학교 산업대학원, 2008.
- [3] 박영미, “중.소규모 조직을 위한 소프트웨어 위험관리 프로세스 프레임워크 개발”, 상명대학교 대학원, 2008.
- [4] 이진국, “CMM 적용의 문제점 분석을 통한 효율적인 CMMI의 적용방안”, 서강대학교 정보통신대학원, 2004.
- [5] Ray C. Williams, “The CMMI RSKM Process Area as a Risk Management Standard”, Carnegie Mellon University, Software Engineering Institute, July 2006.
- [6] 이돈희, “CMM 및 PMBOK에 기반한 프로젝트 관리 방법론에 관한 연구”, 연세대학교 공학대학원, 2004.
- [7] PMI “A Guide to the Project Management Body of Knowledge, 4th Edition”, PMI Inc, 2009.
- [8] James A. Rozum, “Software Measurement Concepts for Acquisition Program Managers”, Software Acquisition Metrics Working Group, June 1992.
- [9] PDCA Security - Information Security Consulting , Auditing and Training “<http://www.pdca-security.com/>”.
- [10] Producing the most excellent MBA knowledge for the student who need to learn MBA Course and help, “<http://www.onlinembawiki.com/pdca-cycle-pdca-model/>”
- [11] Sun-Myung Hwang and Hee-Gyun Yeom, “Metrics Design for Software Process Assessment Based on ISO/IEC 15504”, Department of Computer Engineering Daejeon University, 2006.