

# 소프트웨어 프로세스 개선을 위한 프로세스 성과 측정 모델에 관한 연구

이정윤\*

\*고려대학교 컴퓨터과학기술대학원 소프트웨어공학과  
jyle@kyobodirect.com

## A Study on Process Performance Measurement Model in Software Process Improvement

Jeong Yun Lee\*

\*Dept. of , Computer Science Technology, Korea University

### 요 약

소프트웨어 위기의 근본적인 문제점이 소프트웨어 프로세스의 관리 부재라는 것을 인식하게 됨에 따라 이를 해결하기 위해 조직의 현재 소프트웨어 프로세스 상태를 파악하고, 해당 프로세스가 원하는 목표에 도달할 수 있도록 성과측정모델을 제시함으로써 지속적인 개선 활동을 수행하는데 있어 체계적인 프로세스 관리를 하고자 한다.

본 논문에서는 CMM 을 기반으로 한 소프트웨어 프로세스 개선활동이 제공하는 효과를 달성하기 위한 측정모델을 제시하며 이에 소프트웨어 프로세스 개선 활동이 실제 업무 프로세스에 있어 얼마나 효율적인 결과 및 해당 조직의 성숙도에 미치는 영향을 심층적으로 연구를 함으로써 새로운 프로세스 개선방안을 도출하고자 한다.

### 1. 서론

1970 년 중반부터 하드웨어 성능에 대한 소프트웨어의 개발이 미비하여 “소프트웨어 위기(Software Crisis)”라는 말이 여러 분야의 연구 개발 프로젝트에서 제시되었다. 실제 1994 년 Standish Group 의 조사에 따르면 조사 대상 8,380 개 프로젝트에서 품질, 납기, 비용을 충족한 프로젝트 비율은 16.2%에 불과하며, 재작업 없이 완료한 프로젝트는 6%에 불과하다.

이러한 위기를 극복하기 위해 소프트웨어 업계에서는 70 년대의 구조적 개발 기법에서부터 시작하여 80 년대, 90 년대에 걸쳐 C 와 Ada 등의

구조적 언어의 도입, CASE 활용, 객체지향, 프로토타이핑 개발 방법 등을 적용하였지만, 근본적인 해결책은 되지 못하고 부분적인 성과만 거두었다.

새로운 Tool 이나 기술 도입의 한계를 인식한 소프트웨어 업계는 소프트웨어 위기의 근본적인 문제점이 소프트웨어 프로세스의 관리 부재라는 것을 인식하게 되었으며, 이를 해결하기 위해 조직의 현재 소프트웨어 프로세스 상태를 파악하고, 해당 프로세스가 원하는 목표에 도달할 수 있도록 지속적인 개선 활동을 수행하는 프로세스적인 접근 방법을 모색하게 되었다.

프로세스적인 접근 방법은 Philip Crosby가 1978년 “Quality is free”에서 정의한 “Five Evolutionary stages in adopting quality practices”를 Watts S. Humphrey가 소프트웨어 프로세스에 적용할 수 있도록 한 “Process Maturity Framework”에 의해 개념이 정립되었으며, 미 국방성의 요구로 만들어진 Capability Maturity Model for S/W(이하 CMM) version 1.0, 1.1[25-26]에 의해 체계를 갖추게 되었다. [1]

이후 해외 선진 소프트웨어 업체에서는 이러한 CMM 적용을 통하여 가시적인 성과를 거둔 것으로 나타나고 있으며, 점차 전세계적으로 확산되고 있는 추세에 있다.

이러한 시대적 흐름에 발맞추어 국내에서도 일부 대기업의 SI 업체와 연구소를 중심으로 CMM 적용을 위한 준비 및 연구가 이루어지고 있다. 그러나 국내 대표적인 SI 업체인 A사의 경우 1996년부터 CMM을 도입/적용하였지만, 외국에서 개발한 모델의 한계로 인해 국내 환경과는 맞지 않은 부분이 존재하며, CMM 적용 후의 개선 효과 파악이 미흡하여, 프로세스 개선 활동에 있어 조직원들의 지속적인 참여 유도에 실패함에 따라 모델에서 제시한 개선 효과를 거두지 못하고 있는 실정이다.

## 2. 관련 연구

### 2.1 정의 및 분류

본 연구의 목적인 소프트웨어 프로세스 능력 성숙도 향상을 위한 개선 방안을 도출하기에 앞서 현 적용 모델인 CMM과 SPICE에 대한 연구를 선행할 것이다. 프로세스에 대한 개념과 프로세스 개선 모델 중 현재 가장 널리 사용하는 모델에 대한 기존 이론을 정리, 이를 바탕으로 현재 국내 금융기관의 프로세스 능력 성숙도를 진단 할 예정이다. [4]

본 연구는 금융기관을 중심으로 프로세스 능력 성숙도에 대한 현재의 상태를 진단하고 현재의 상태로부터 개선의 상태로 진화하는 과정의 요소를 분석, 평가하여 현재의 상태와 개선의 상태의 상관관계를 규명하고 우리 실정에서의 개선 요소를 분석, 보다 효율적인 개선 방안을 제시코자 한다.

### 2.2 CMM / IDEAL Model

국내 SI 업체가 수행한 소프트웨어 프로세스 개선

사례에 대해 살펴 보고, 활동결과에 대한 개선점과 연구 모형과의 관계를 설정함으로써, 연구 결과의 향후 활용 방향을 제시하고자 한다. [7]

### 2.3 프로세스 개선계획 이행

CMM 2단계 프로세스 표준화 부분과 전사차원의 프로세스 정비는 조직이 보유하고 있는 프로세스 개체에 따라 부분적으로 내용을 수정함으로써 완료하였으며, 다음에서 내부 프로세스 능력 심사 모델 개발 부분의 이행 내용을 설명하고, 성과 측정을 위한 메트릭 프로그램 구축부분은 본 연구 결과를 바탕으로 이행할 계획이다. [2]

[표1] A사 CMM 공식 심사 후 개선 계획

개선 항목	개선 방향
CMM 2단계 프로세스 표준화	<ul style="list-style-type: none"> <li>공식 심사 시 발견된 문제를 보완하고 2 단계 프로세스 및 하위 프로세스를 추가 개발, 표준화하여 모든 조직에서 쉽게 활용하도록 함</li> </ul>
전사 차원의 프로세스 정비	<ul style="list-style-type: none"> <li>프로세스 개선 활동의 조정 역할을 수행할 수 있는 조직을 구성하여 지속적으로 전사 차원의 표준 소프트웨어 프로세스 자산(표준 프로세스, 프로젝트 수행 테일러링 가이드, 메트릭 DB, 표준 문서/양식)을 정비하도록 함</li> </ul>
내부 프로세스 능력 심사 모델 개발	<ul style="list-style-type: none"> <li>동기 부여를 위해 CMM 단계뿐만 아니라 각 프로세스별로 인정할 수 있는 내부 심사 제도 개발/적용</li> </ul>
성과 측정을 위한 메트릭 프로그램 구축	<ul style="list-style-type: none"> <li>프로세스 개선 프로그램의 제도화</li> <li>CMM 적용시의 주요 메트릭을 선정, 관리할 수 있도록 하고, 이를 통해 프로세스 개선 활동을 조직의 성과와 연계하여 관리하도록 함</li> </ul>

### 3. Software Process 개선 활동 방안 연구

본 논문에서 제시하고자 하는 소프트웨어 프로세스 개선활동 내용에 따라 소프트웨어 프로세스 개선 제도 활동 및 산출물에 관련된 세부 내용 및 절차에 대해 알아보겠다. [5]

[표2] A사 소프트웨어 프로세스 개선 제도 활동 및 산출물

단계	활동	산출물
1단계: 기본 조직 수립	<ul style="list-style-type: none"> <li>● 사업부, 팀, 프로젝트 관리자의 참여를 유도</li> <li>● SPI(S/W Process Improvement)활동에 참여할 프로젝트 결정</li> <li>● SPI 활동을 수행할 조직 및 그들의 책임과 역할을 정의</li> </ul>	<ul style="list-style-type: none"> <li>● 팀/프로젝트 프로파일</li> <li>● 설문 분석 결과</li> <li>● SPI 리더 및 요원</li> </ul>
2단계: SPI 목표 설정	<ul style="list-style-type: none"> <li>● SPI 활동 결과에 대한 목표를 설정</li> <li>● 목표에 대해 팀/프로젝트 인원들이 합의</li> </ul>	<ul style="list-style-type: none"> <li>● 팀 SPI 계획서</li> </ul>
3단계: SPI 접근 방식 개발	<ul style="list-style-type: none"> <li>● 팀에서 SPI 활동을 수행하기 위한 사전 계획 및 일정을 개발</li> </ul>	<ul style="list-style-type: none"> <li>● 팀 SPI 계획서</li> </ul>
4단계: 관리자 승인 획득	<ul style="list-style-type: none"> <li>● 팀에서 SPI 활동을 이행하기 위한 계획 발표</li> <li>● SPI 접근 방식, 목표 등에 대한 관리자의 승인</li> </ul>	<ul style="list-style-type: none"> <li>● 팀 SPI 계획서</li> </ul>
5단계: 프로세스 심사 수행 및 강약점 파악	<ul style="list-style-type: none"> <li>● SPI 계획에 따라 조직의 강점과 약점을 파악하기 위해 프로세스 심사를 수행</li> <li>● 프로젝트 프로세스의 강점(최상의 활동)과 약점(개선할 영역)을 결정</li> </ul>	<ul style="list-style-type: none"> <li>● 프로세스 심사 계획서</li> <li>● 프로세스 심사 결과 보고서</li> </ul>
6단계: 시정 활동 결정 및 개선 계획 수립	<ul style="list-style-type: none"> <li>● 프로세스 심사 결과를 바탕으로 프로젝트 SPI 계획서를 문서화</li> <li>● 개정된 팀 SPI 계획서와 프로젝트 SPI 계획서에 대한 팀/프로젝트 관리자의 승인</li> </ul>	<ul style="list-style-type: none"> <li>● 개정된 팀 SPI 계획서</li> <li>● 프로젝트 SPI 계획서</li> </ul>

[표2] A사 소프트웨어 프로세스 개선 제도 활동 및 산출물

단계	활동	산출물
7단계: 개선안 이행	<ul style="list-style-type: none"> <li>● 팀/프로젝트 SPI 계획서를 이행</li> </ul>	<ul style="list-style-type: none"> <li>● 팀/프로젝트 SPI 활동 결과물</li> <li>● 팀/프로젝트 SPI 활동 목표 달성도</li> </ul>
8단계: 진척, 문제점, 이슈 보고	<ul style="list-style-type: none"> <li>● SPI 활동의 진척도를 판단하기 위해 주기적으로 활동 상태를 보고</li> <li>● SPI 활동에서의 문제점이나 이슈를 보고</li> </ul>	<ul style="list-style-type: none"> <li>● SPI 활동 보고서</li> <li>● 개정된 팀/프로젝트 SPI 계획서</li> </ul>
9단계: SPI 활동 검토	<ul style="list-style-type: none"> <li>● 주기적인 검토 활동을 실시하여 S/W 프로세스 개선 목표를 검증</li> <li>● 지속적인 프로세스 개선을 위해 팀/프로젝트 SPI 계획서를 Update</li> </ul>	<ul style="list-style-type: none"> <li>● SPI 상태 보고서</li> <li>● 개정된 팀/프로젝트 SPI 계획서</li> </ul>
10단계: 개선 결과를 다른 프로젝 트에 전파	<ul style="list-style-type: none"> <li>● 팀/프로젝트의 SPI 활동이 다른 프로젝트로 확장될 수 있는지를 결정/전파</li> </ul>	<ul style="list-style-type: none"> <li>● SPI Best Practice</li> </ul>

구축된 소프트웨어 프로세스 개선 제도에서는 팀/프로젝트 SPI 계획서 작성시 조직의 목표에 따른 메트릭 설정 부분을 자율에 맡기고 있어 실제 프로세스 개선 제도화의 잇점을 살리지 못하고 있는 실정이다. 그러므로 본 연구에서는 소프트웨어 프로세스 개선 활동 결과와 조직의 성과 측정 유형과의 관계를 분석/검증하여, 조직에서 소프트웨어 프로세스 개선 계획서를 작성하거나 내부 프로세스 심사 결과보고서의 개선안 부분에 대한 가이드를 제시함으로써 이러한 문제점을 해결하고자 한다.

3.1 소프트웨어 프로세스 개선활동 결과 측정 메트릭

일반적으로 성과 측정 프로그램은 조직에게 제품, 서비스, 프로세스에 대해 계량화된 정보를 제공하며, 성공적인 성과 측정 프로그램은 많은 수의 메트릭을 측정하는 것이 아니라 필요한 부분을 측정하는 것이다. 그러므로 본 연구에서는 CMM 2, 3 단계의

핵심 프로세스 영역에 대하여 GQM 방식[7]을 사용하여 소프트웨어 프로세스 개선 활동 결과 측정 메트릭을 설정하여 결과를 측정하였다.

### 3.2 성과 측정 유형

본 연구에서 조직의 성과 측정 유형변수는 1995년에 미국의 DOE(DEPARTMENT OF ENERGY)의 TRADE(TRAINING RESOURCES AND DATA EXCHANGE)와 PBM SIG(PERFORMANCE-BASED MANAGEMENT SPECIAL INTEREST GROUP)에서 성과 관리 기법의 이용을 장려하기 위해 발간한 “How to Measure Performance: A Handbook of Techniques and Tools”에서 정의한 6가지 성과 측정 유형을 활용하고자 한다. 성과 측정유형은 생산성(Productivity), 효과성(Effectiveness), 효율성(Efficiency), 품질(Quality), 적시성(Timeliness), 안전성(Safety)으로 나뉘어져 성과 측정 영역에 포함하였다. [6]

### 3.3 연구모형 설정

본 연구에서는 소프트웨어 프로세스 개선 활동 결과와 조직의 성과와의 관계를 분석하기 위해 CMM2, 3단계 핵심 프로세스 영역에서 추출한 소프트웨어 프로세스 개선 활동 결과 측정 메트릭의 특성을 DOE(Department Of Energy)의 TRADE(Training Resources And Data Exchange)와 PBM SIG(Performance-Based Management Special Interest Group)에서 제시하는 4개의 성과 측정 유형과 2개의 관리 주체(조직 및 프로세스) 변수를 사용하여 분류하고자 한다. 요구사항 충분성, 요구사항 테스트율, Estimates의 정확도, 태스크 수행율, 산출물 검토 비율, 조직 프로세스 활용도 등을 조직 및 프로젝트 기준으로 생산성, 품질, 적시성, 안정성을 고려하여 최소한의 기본 메트릭만을 사용할 수 있도록 하였다. [6]

## 4. 결론 및 향후 과제

소프트웨어 프로세스 개선 활동의 결과를 조직의 성과 측정 유형과 연계하여 조직원들에게 정기적으로 개선 정도를 보여줌으로써 지속적인 조직원의 참여를

유도하기 위해 CMM2, 3단계의 핵심 프로세스 영역에 대한 실증적 분석을 수행하고, 소프트웨어 프로세스 개선 활동을 성공적으로 수행하기 위하여 개선활동의 결과가 조직의 성과와 연결되어 조직원들의 지속적인 참여를 유도함으로써, 소프트웨어 프로세스 개선활동의 성과를 파악하는데 활용되었다. 조직의 성과 및 관리 주체와의 관계를 분석하기 위해 설문조사를 수행함으로써 소프트웨어 프로세스 개선 활동의 성과를 파악하는데 활용되었으며, 향후 프로세스 개선활동을 하는데 있어 중요한 역할이 되어질 것으로 보인다.

### [참고 문헌]

- [1] CMU/SEI, “The Evidence for CMM-Based Software Process Improvement”, Software Engineering Institute, May1998
- [2] 이민재, 조창현, 정학중, “소프트웨어 프로세스 심사 모델에 관한 연구”, SQMS98제2회 소프트웨어 품질관리 심포지엄 논문집, 1998
- [3] 김도균, 박남직, 정학중, “CMM과 프로세스 개선 사례 연구”, 정보공학회지,
- [4] CMU/SEI, Introduction to the Capability Maturity Model, Software Engineering Institute, 1996
- [5] LG-CNS “사내 소프트웨어 프로세스 개선 핸드북”, 1999
- [6] TRADE(Training Resources and Data Exchange), “How to Measure Performance: A Handbook of Techniques and Tools”, 1995
- [7] Tom Kimbrough, Linda Levine, “The IDEAL Model Transition Framework”, SEI Software Engineering Symposium,1997