

COCOMOII의 cost driver를 이용한 PCD결과의 위험분석

박지협⁰, 이은서, 장윤정, 이경환
 중앙대학교 소프트웨어공학 연구실

{mux98⁰, eslee, yjjang, kwlee}@object.cau.ac.kr

Risk analysis using PCD result and cost driver of COCOMOII

Ji-Hyub Park⁰ Eun-Seo Lee Yun-Jung Jang Kyung-Whan Lee
 Dept. of Computer Science and Engineering, Chung-Ang University

요약

본 논문에서는 신뢰성 있는 소프트웨어 개발을 위한 위험분석 방법을 모색한다. 기존의 프로세스 능력 결정(PCD: Process Capability Determination)결과의 차이를 이용한 전체 위험결정 방법의 취약점을 보완하기 위해 MBASE(COCOMOII)에서 제시하는 위험분석 방법을 이용한다. 이 과정에서 필수적으로 수행되어야 할 과제는 레벨의 MP(Management Practice)단위 속성과 cost driver를 매핑함으로써 개별 위험항목(risk item)을 얻는 것이다. SPICE 레벨4 단계를 예로 cost driver를 추출하고 rating결과를 입력하여 위험요소의 우선순위를 결정하며, 결정된 위험 우선순위에 따라 위험을 제거할 경우 전체 위험이 급격히 감소되어 RE(Risk Exposure) 그래프의 하강 곡선을 따름을 증명하였다. 이러한 산정결과를 통해 레벨에 영향을 미치는 위험 속성 및 위험 제거시 전체위험의 감소율을 예상할 수 있다.

1. 서론

기업에서 개발하는 소프트웨어의 신뢰성은 해당 프로젝트의 프로세스에 많은 영향을 받는다. 이러한 신뢰성 있는 소프트웨어 생산을 위해 수행되는 프로세스 심사 지침으로 SPICE가 이용되는데, SPICE (Software Process Improvement and Capability dEtermination)는 SPA (Software Process Assessment)에 대한 국제적인 표준 작성을 위한 ISO와 IEC의 노력의 산물이다.

프로젝트 수행에 있어서 신뢰성 있는 프로젝트를 위해서 개선 이전에 선행되어야 할 단계가 바로 위험 분석으로, 정량적 위험분석은 개선을 위한 토대가 된다. SPICE 심사에서의 위험분석은 프로세스 능력 결정(PCD: Process Capability Determination) 단계에서 목표 레벨과 현재 능력 레벨을 비교하여 프로세스 능력을 결정하고, 이들의 차이분석 결과를 이용하여 위험 분석이 수행 가능하다[SPICE/ISO/IEC TR2 15504].

또한 비용산출을 위한 COCOMOII는 최근 비용 뿐 아니라 위험 평가(risk assesment) 부분까지 확장되어 위험 확률 모델을 제시하고 있다. 신뢰성 있는 예상 비용산출을 위해 가장 빈번히 요구되는 요소 22가지를 cost driver로 정의하여 경험적 데이터를 기반으로 예상 위험 산출을 위한 산출방법을 정의하고 있다[COCOMOII].

본 논문에서는 SPICE에서 프로세스 능력결정 단계인 PCD 결과를 이용한 위험분석 부분에 COCOMOII의 위험 확률산출 방법을 도입하기 위해서 SPICE 레벨 4의 각 단계와 cost driver를 매핑하고 레벨의 잠재적 위험 확률을 산출한다.

2. 기반연구

위험분석 방법을 실행하기 위해 COCOMOII에서 정의된 위험분석 방법과 SPICE 심사 단계에서 프로세스 능력결정 결과를 이용한 전체 위험 분석 방법을 알아본다.

2.1 COCOMOII의 위험분석 방법

COCOMOII에서는 품질을 결정하는 비용, 스케줄, defect 등에

관련된 5개의 scale factors와 17개의 effort multipliers를 합한 22개의 cost driver를 정의하였는데 이는 다음 표와 같다.

< 표 1 > 22개의 cost driver

요구되는 소프트웨어 신뢰성	W/W와 S/W의 복잡성
베어메이스 크기가 미치는 영향	멀티미디어를 통한 커뮤니케이션
프로젝트 복잡도(제어, 장치외관..)	스케줄대로 이행되는 정도
제작성을 위한 컴퍼넌트 구축	이전의 프로젝트들의 유사성도
인플리케이션 적용 경험	CMM을 통한 프로세스 성숙도
플랫폼 사용 경험 및 이해	분석가의 능력(분석 및 설계)
산/인어와 불 경험 수준	프로그램의 능력 정도
현장에서 관리보장 불가능의 단계	프로젝트 1년 주기의 직면 어려움
바어시이클에 맞는 분석 계획성	디자인 견고성 및 위험제거
부하된 실행시간 제약의 측정항	스태플러들의 이해 일치
불기약 장치 제약 정도	소프트웨어 용량성

또한 위험분석을 수행하기 위해, 위의 cost driver를 이용하여 6개의 위험 카테고리별로 분류하여 각 위험항목(risk item)을 통해 driver 간의 위험 연관성을 정의하고 있다.

< 표 2 > 프로젝트 위험 카테고리

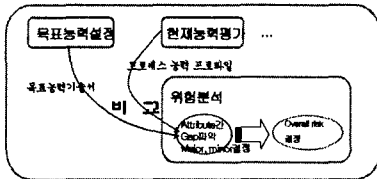
전체 프로젝트 위험 카테고리

Schedule risk	Product risk	Platform risk	Personnel risk	Process risk	Reuse risk
SCDD	RELY	TIME	ACAP	TOOL	REUSE
Ssd_spl	Rdy_acp	Ssd_line	Asp_rsk	Ssd_tool	Rust_rly
Ssd_rly	Rdy_pcp	Time_scp	Cpk_acp	Yol_acp	Rust_ssp
Ssd_line	Rdy_pmt	Time_jcap	Cpk_acp_pcp	Yol_pcp	Rust_sls
Ssd_pcd	Ssd_rly	Cpk_line_pcd	Prst_acp	Yol_tool	Rust_scp
Ssd_tool	Rdy_line_pcd	Time_spr_pcd	Rdy_jcap	Time_tool	Rust_line
Ssd_ssp	Rdy_star_pcd	Time_gvst_pcd	Rdy_acp_pcp	Yol_pmt	Rust_rst
Ssd_ssp	Rdy_star_pcp	Rst_line	Rst_acp	Rst_tool	Rust_scc
Ssd_pcp	Prst_rly	Time_tool	Ssd_acp	Ssd_tool	Prst_line
Ssd_vst	Rst_rly	Prst_line	Ssd_scp	Rst_tool	Rst_line
Ssd_rst	Rst_rly	Time_line	Time_scp	SITE	
Ssd_data	DATA	Rst_line	Tool_scp	Ssd_tool	
Ssd_smat	Rdy_line_pcd	STDR	Pch_acp	Ssd_tool	
Ssd_scp	SIZE	Ssd_acp	Prst_acp	Rst_line	
Ssd_sls	Ssd_scp	Ssd_pcp	Rst_acp	Ssd_line	
Ssd_ssc	Cpk	Rst_line	AEXP	Prst_line	
Ssd_rst	Cpk_acp	Cpk_star_pcd	Trst_star_pcd	Trst_line	
Ssd_rst	Cpk_acp_pcp	Time_star_pcd	Rst_scp	Prst	
Ssd_tool	Cpk_jcap	Prst_star	Ssd_scp	Prst_line	

본 논문에서는 위의 cost driver 및 위험 카테고리를 이용하여 카테고리 단위 및 전체 위험확률을 도출한다.

2.2 기존의 PCD결과를 이용한 위험분석

SPICE의 심사단계에서 프로세스 능력 결정(PCD)은 특별히 명시된 요구사항을 만족하기 위해 프로세스를 전개하는데 따른 강점 및 개선점을 파악하고 프로세스들을 평가, 분석하는 것을 말한다. 주 목적은 의사결정을 위해 정보를 획득하는 것으로 프로세스의 프로파일은 기본 데이터가 된다.



< 그림 1 > 프로세스 능력 결정 결과의 위험 분석 과정

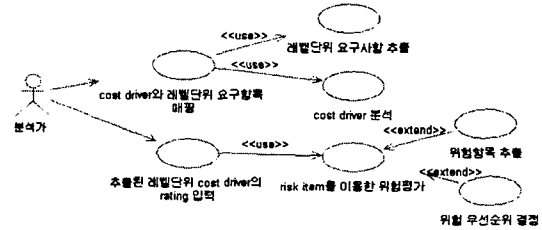
PCD 결과를 통한 위험분석은 <그림 1>과 같이 수행된다. 즉, 목표능력과 현재능력간의 차이를 분석하여 주 요소와 보조 요소(major, minor)를 결정하고, 이를 통해 전반적 위험(overall risk)을 도출한다. 목표능력설정 부분에서는 목표설정 및 계획, 착수 범위를 정의하며 목표능력기술서를 작성하고, 현재능력평가에서는 조직의 현 프로세스 능력을 실제로 반영하여 능력 프로파일을 작성한다. 이를 기반으로 두 프로파일을 비교, 대조하여 각각의 레벨단위 차이를 도출하고 정의된 능력차이 표를 통해 위험분석을 수행한다.

2.3 기존 위험분석 방법의 개선점

일반적으로 수행되는 기존의 위험분석 방법으로는 프로세스 항목 차이를 통해 전반적 위험(overall risk)을 도출하는 정도이다. 뿐만 아니라 이와 같이 결정되는 전체 위험은 위험 정도가 각 레벨 단위로 (no identifiable, medium risk, high risk)로만 분류된다. 즉, 각 레벨의 보다 상세한 PA(Process Attribute)단위 내에서 어떠한 요소가 위험을 일으키는 지 구체적인 위험 항목 및 연관관계를 추출할 수 없으며, 각 위험 요소별 위험 확률과 그에 따른 위험요소의 우선순위를 알 수 없다. 이렇듯 위험 요소 항목 및 가장 큰 영향을 미치는 위험항목을 알 수 없으므로, 위험해결을 통한 프로세스 성능 향상(SPI: Software Process Improvement)을 위한 위험 해결 지표를 제공하지 못한다. 다만, 원인이 되는 항목을 추출하기 위한 방법으로, 위험 분석을 수행하여 추출된 능력프로파일의 차이를 통해 해당 PA별 MP(Management Practice)를 추적해 나가는 방법이 있다. 그러나 이 방법은 각각의 차이마다 역추적해 나가야 한다는 번거로움이 있으며 위험 항목의 개별적 확률을 나타내지 못하는 단점이 있다. 이러한 취약점을 보완하기 위한 방법으로 COCOMOII의 cost driver를 적용해 볼 수 있다.

3. COCOMOII의 위험항목 분류를 적용한 위험분석

<그림 1>과 같은 위험 분석은 전반적인 위험 정도만을 감지하는데 비해, COCOMOII의 cost driver를 기반으로 분류된 위험항목을 이용하면 MP단위로 추출된 드라이버에 ranking을 입력하여 개별 확률 및 레벨 단위 전체 위험 확률이 도출 가능하다. 이와 같이 COCOMOII를 이용한 위험평가방법은 다음과 같이 수행될 수 있다.



< 그림 2 > COCOMOII를 적용한 레벨단위 위험평가 방법

이에 대한 방법은 SPICE 레벨 4를 기반으로 다음과 같이 수행하였다.

3.1 SPICE 레벨단위의 cost driver 위험항목 매핑

SPICE 레벨 단위의 위험분석에 COCOMOII를 적용하기 위해서는 먼저 레벨 항목별로 cost driver를 매핑하는 단계가 필요하다. COCOMOII의 비용산정 cost driver는 22개이며, 위험 산정을 위한 수식을 포함하고 있다. 이를 바탕으로 위험 분석에 대한 지표를 제시하고 있는데, 각각의 정의된 cost driver는 연관된 개별 위험항목을 포함한다. 우선 레벨 4의 PA단위 MP들의 요소들을 분석하여 매핑한 cost driver는 다음과 같다. 각 MP들의 요구되는 요소(required attribute)는 MP별로 만족되어야 할 사항들이며, 개별 위험항목(individual risk item)은 각각의 추출된 MP의 해당 항목들과의 연관성을 기준으로 cost driver를 추출했다.

< 표 3 > SPICE 레벨 단위 위험요소 분류표

레벨	프로세스 측정의 목적	위험항목	비용산정 요소
4	4.1 프로세스 측정의 목적	4.1.1 제품/프로세스의 목표 및 식별	프로세스 목표와 업무목표 일치성 데이터의 실제 적용성 및 신뢰성 프로젝트 측정이 계획서의 일정 포함 정신성 및 품질 매트릭을 위한 제품 매트릭
		4.1.2 제품/프로세스 측정 수집	데이터의 신뢰성 측정치의 실제 상황 반영 여부 보고관객의 생명주기모형 적용성
		4.1.3 프로세스 수행 추이분석	측정 보고서의 원형 전달성 데이터 품질보증활동 및 기법 결함 기록/추적 시스템
	4.1.4 프로세스 통제 측정	수행 능력과 수행 목표의 차이일 때 관련 수정행동이 일어나는 문제영역 식별 계획 및 일정 조정 기준 및 방법 프로세스 추적 도구 측정 보고서의 원형 전달성 데이터 품질보증활동 및 기법	
4	4.2 프로세스 측정의 목적	4.2.1 위험한 측정기법 식별	프로세스 통제기법 요구사항 만족 충분한 분석기법 사용 분석기법이 사례상황에 맞는가

4	4.2 프로세스 수행의 확장	프로세스 수행 감시를 위한 종래의 측정치 수집
		시정조치 필요한 점을 알기 위한 감시
	4.2.2 측정치 수집	메트릭을 실제 사용하고 있는가
		프로세스 및 제품 메트릭과 실제상태의 일치성
4.2.3 프로세스 수행 통제	재사용을 위한 프로세스 심사 결과 저장	
	계획된 통제계획과 프로젝트 수행이 일치하지 않을 때에 대한 조치 - 시정조치와 관련된 문제영 역 식별 - 계획과 일정 조정	
	측정치 주어 데이터를 사용하여 시정조치 결과 감시	
		품질 일계치 평가

위의 표에서 개별 위험 요소 부분은 각 PA단위 MP가 만족해야 할 요구항목들이다. 이러한 요구항목 또는 cost driver는 이 이외에도 필요에 의해 추가, 수정 가능한 항목한 항목으로, 위 표에서 < >내의 드라이버는 MP별로 요구된 요구항목들과 매핑되지 않음으로 인해 추가되어야 할 새로운 cost driver이다. 이는 다음과 같다.

• MERI(Metric)

: 프로젝트 목표에 부합한 자체적 기준 및 메트릭

• CHEF(Change-Effect)

: 하나의 변경이 다른 영역에 미치는 영향의 정도

그러나 위에서 추가되는 두개의 cost driver는 데이터를 기반으로 정량적으로 정의되지 못했으므로 다음절에서 수행되는 위험확률평가에서는 제외시키기로 한다.

3.2 전체 프로젝트 위험확률 실행 및 결과

레벨 단위 항목별로 매핑된 cost driver를 통해 위험분석을 수행한다. 우선 추출된 cost driver에 해당 항목에 대한 rating 결과를 입력한다. 각 cost driver 별로 연관된 위험요소는 정의된 산출식에 따라 정의되며, 각각 COCOMOII 내에 정의된 위험 카테고리 분류표에 의해 분류된다.

SPICE 레벨 4 전체의 cost driver(PA 4.1, 4.2)를 추출해서 rating을 입력했을 때의 각 위험률은 다음과 같다.

< 표 4 > cost driver 단위 rating 입력값

VH	L	N	L	H	L	N	L	L	H
----	---	---	---	---	---	---	---	---	---

Risk Assessment Summary (0-100)

Total Project Risk	4.4
Schedule Risk	5.0
Product Risk	10.1
Platform Risk	0.0
Personnel Risk	3.0
Process Risk	0.0
Reuse Risk	0.0

--- This project is a low risk. Try to mitigate the top individual risk items. ---

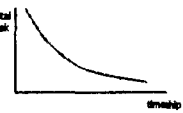
< 그림 3 > 위험평가 실행 결과 (SLOC=15000로 가정)

위의 <그림 3>는 USC(http://sunset.usc.edu 사이트 참조)에서 제공되는 툴에 <표 4>의 값을 입력하고 실행했을 때의 결과이다. 위험을 제거하지 않았을 때의 초기 등급은 4.4이며 가장 큰 영향을 미치는 위험은 product risk(10.1)이다. 그리고 개별 위험요소는 rely_acap_risk가 3.00으로 가장 높다. 따라서 이 항목의 해당 cost driver를 추적하면 RELY, ACAP인데, <표 5>는 이러한 과정을 설명하고 있다.

< 표 5 > 위험분석 수행 결과

product(10.1)	rely_acap_risk(3.00)	4.4
product(3.0)	rely_acap_risk(1.31)	1.6
product(1.4)	rely_pmat_risk(1.16)	0.3
모든 리스크 카테고리(0.0)	모든 개별 리스크 아이템 (0.00)	0.0

이 드라이버들의 위험을 차례로 제거했을 경우, 아래의 <그림 4>와 같은 결과가 나타난다.



< 그림 4 >

< 그림 5 >

위의 <그림 4> 그래프에서 마지막 4번째로 위험을 제거했을 경우 전체 위험은 0.0이다. 그러나 현실적으로 어떠한 프로젝트 수행에 있어서 잠재 위험을 배제할 수 없으므로 잠재 위험이 항상 존재한다고 가정하면 위험 곡선은 <그림 5>와 같이 나타난다.

이와 같은 결과를 통해 전체 위험에 영향을 미치는 위험요소의 우선순위를 결정하고 그에 연관된 속성을 추출할 수 있다.

< 표 6 > 레벨에서 ACAP와 연관된 속성

ACAP	4.1.1	프로세스 목표와 업무목표 일치성
		프로젝트 측정이 계획서의 일정 포함
	4.1.3	결함 기록 / 추적 시스템
	4.1.4	수행 능력과 수행 목표의 차이시 관련 수정행동이 일어나는 문제영역 식별
	4.2.1	충분한 분석기법 사용
	4.2.3	계획된 통제계획과 프로젝트 수행이 일치하지 않을 때 시정 조치와 관련된 문제영역 식별

이러한 <표 6>은 <표 5>의 결과에서 두 번째 우선순위의 driver ACAP에 연관된 속성을 <표 3>을 통해 추출한 결과이다.

4. 결론 및 향후 과제

각 레벨 단위로 cost driver를 결정하여 전체 프로젝트 위험확률을 도출했을 경우, PCD의 위험분석 단계에서 레벨에 영향을 미치는 위험요소와 각각의 개별 위험요소확률 및 해당 레벨의 전체 프로젝트 위험 확률을 정량적인 방법을 통해 예측할 수 있다.

이러한 구체적인 위험 항목 및 확률 정보는 이후의 위험 개선 단계의 토대가 된다. 그러나 이러한 단계에서 가장 중요한 레벨의 MP와 cost driver 기반의 위험요소를 점검시키는데 있어서 매핑되지 않는 항목들에 대한 처리가 요구되는데, 이 과정에서 적합한 위험항목을 위한 새로운 cost driver 정의 단계가 필요하다. 또한 위에서 논의한 레벨 단위에서의 위험확률 뿐 아니라 예상 위험의 비용 산출 또한 논의해 볼 과제이다.

참고문헌

- [1] ISO/IEC JTC1/SC7 15504 Information Technology-Software Process Assessment 1998.01
- [2] Barry W. Boehm, etc Software cost estimation with COCOMOII Prentice Hall PTR 2000.
- [3] R.Madachy,Heuristic Risk Assessment Using Cost Factors IEEE, 1996.5
- [4] B.Boehm Software Risk Management IEEE 1989
- [5] USC COCOMOII Post-Architecture Model with Heuristic Risk Assessment Internet implementation Tool http://sunset.usc.edu/research/COCOMOII/expert_cocomo/expert_cocomo.html