

SI 프로젝트 수행을 위한 내부관리 관점의 위험관리프레임워크

이주희*, 궁한경**, 백상기**, 양대철**, 장송봉** 최성운***
*명지대학교 컴퓨터공학과
**삼성 SDS
***명지대학교 컴퓨터공학과 교수
e-mail : lee6096@hanmail.net

Risk Management Framework based on internal viewpoints for SI Project

Ju-hee Lee*,
Han-kyong Kung**, Sang-gi Back**, Dae-chul Yang**, Song-bong Jang**, Sung-woon Choi***
*Dept. of Computer Engineering, Myong-ji University
**SDS, Samsung
*** Dept. of Computer Engineering, Myong-ji University professor

요 약

많은 기업의 정보화 프로젝트가 계획된 예산과 기한 내에 완수하지 못하고, 폐지되거나 당초 예상된 성과를 달성하지 못하고 있다. 정보화 프로젝트의 성공 확률을 높이기 위해서는 프로젝트 실패의 원인이 될 수 있는 잠재된 위험을 식별하고, 식별된 위험요인을 체계적이며, 지속적으로 관리할 필요가 있다. 본 논문에서는 정보화 사업 중 시스템통합 사업에서 발생할 수 있는 위험요인을 체계적으로 분류하고 관리 할 수 있는 기반을 제안하고자 한다. 위험관리 프레임워크는 이해당사자들의 관점을 중심으로 위험요인을 체계적으로 분류하고, 위험요인을 점검할 수 있는 체크리스트를 제공함으로써 각 이해당사자들의 요구에 맞게 SI 프로젝트의 위험에 대응하여 정보화 프로젝트의 성공 확률을 높이는 데 이바지 할 수 있다.

1. 서론

많은 기업의 정보화 프로젝트가 계획된 예산과 기한 내에 완수하지 못하고, 폐지되거나 당초 예상된 성과를 달성하지 못하고 있다. Standish 컨설팅 그룹의 2000 년도 조사에 따르면, 기업 소프트웨어 개발 프로젝트의 23%가 실패하고, 단지 28%만이 원래 계획된 예산과 기한 내에 약속된 기능을 제공하였다.[1] 국내의 경우에도 60%이상의 정보시스템 프로젝트가 1 년 이내에 폐기되는 등 실패율이 높은 것으로 알려져 있다.[2] 따라서 정보시스템 프로젝트의 실패를 줄일 수 있는 적절한 대안에 대한 요구가 절실하다.

정보화 프로젝트에 내제된 위험요인을 초기에 인식하고 지속 적인 관리활동을 수행한다면 프로젝트의 실패를 줄이고 성공 가능성을 높일 수 있다. 반면, 프로젝트를 관리 함에 있어 체계적이 위험 관리가 이루어 지지 않는다면 당초 계획된 예산과 일정을 초과하여 실패에 이르거나, 성공하더라도 내제된 위험요소들에 의해 추가적인 문제들이 발생할 수 있다.

[3]의 연구에서는 프로젝트에서 발생할 수 있는 위험요소를 인식하고 관리활동을 수행했을 경우 프로젝트 성과를 향상시킬 수 있다는 가설을 세우고, 프로

젝트 관리자들을 통해 설문 조사한 결과, 이러한 가설이 유효함을 확인할 수 있었다.

본 논문에서는 시스템통합 사업에서 위험요인을 체계화한 위험관리 모델 제시하여, 기업이 이를 통해 효율적인 위험평가와 최적의 대응방안을 모색하여 성공적인 시스템통합 사업의 기반을 마련할 수 있도록 하고자 한다.

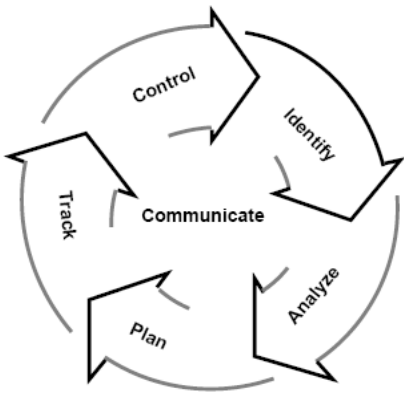
2. 관련연구

The Software Engineering Institute (SEI)에서는 위험을 손실을 초래할 수 있는 가능성으로 정의한다. 개발 프로젝트에서 손실은 최종 제품의 질 감소, 비용의 증가, 종료시기의 지연, 시장 점유율의 감소, 또는 실패의 형식이 될 수 있다. 위험관리는 시스템적이고 지속적인 프로세스로서 다음의 반복적인 프로세스를 따른다.

- 식별 - 위험관리의 대상인 위험요소를 식별하는 것으로 문제요인이 발생하기 이전에 이를 표면화 시키는 작업, 위험평가 기법 제기
- 분석 - 식별된 위험요소를 의사결정을 위한 정보로 전환하는 것으로 가장 핵심적인 요소의 파악을 통

하여 프로젝트 관리자가 올바른 의사결정을 할 수 있는 정보를 제공한다

- 계획수립 - 앞서 정의된 위험관련 정보를 통해 의사결정을 내리고 실행하는 것으로써 보통 위험요소의 나열, 위험관리 실행의 우선순위 및 통합관리 계획의 수립 등을 수행한다.
- 추적 - 위험요소의 상태를 감시하고 이의 대안을 개선하는 것으로 위험감소 및 제거를 위해 추적 매트릭스-위험요소의 상태를 판단하기 위한 요소 별 측정적도를 활용한다.
- 통제 - 계획과 실행간의 편차를 수정 해가는 작업으로, 위험제거 활동의 조정, 편차조정, 위험발생에의 응답 및 위험관리 프로세스의 개선을 포함한다



(그림 1) SEI 위험 관리 패러다임

[5] Foo 와 Muruganathan 은 위험 분석의 정량적 측정을 위해 질의 기반 접근을 제안했다. Software Risk Assessment Model(SRAM)로 불리는 이 모델은 프로젝트의 특성과 환경 요인에 의존한다. 이 모델에서는 9 개의 주요 위험 요소들을 고려한다: 소프트웨어 복잡성, 프로젝트 관계자, 목표로 삼은 신뢰성, 제품 요구 사항, 측정 방법론, 모니터링 방법론, 적용된 개발 프로세스, 개발 소프트웨어의 활용성, 도구. 그 이후에, 치명적인 위험 요소들 각각에 대해 세 개의 선택을 제공함으로써, 위험 평가자들을 위한 체크 리스트의 틀을 잡았다. 위험 평가자들의 답변들은 평가를 거쳐, 위험 레벨에 따라 오름차순으로 정렬된다. 그러나 이 연구는 개발 단계의 위험만을 다루고 있으며, 여전히 주관적인 체크리스트에만 의존하고 있어, 다양한 환경에 적응하기 어렵다.

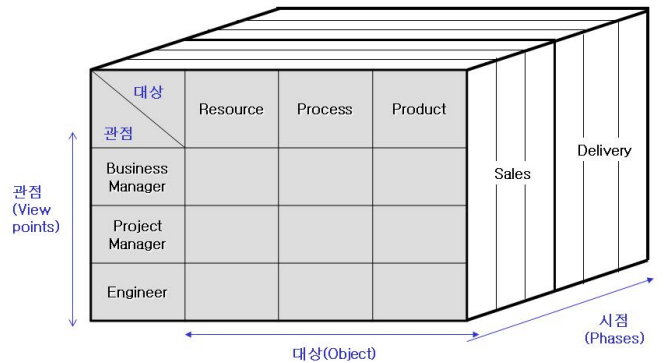
[6] Kwan 와 Leung 은 SEI 의 위험 분류 방법론(Risk taxonomy)을 최근 정보화 프로젝트의 상황에 맞게 개선한 위험 식별 프레임워크를 제안했다. 이 소프트웨어 공학 측면에서 시스템 공학 측면으로 확장하여 SEI 분류 방법론을 개선하였다. 시스템 공학 측면에서 식별될 수 있는 위험 요소들을 추가적으로 식별하였으며, SEI 위험 분류에서 소프트웨어를 제작하기 위한 방법론, 절차, 도구들에 초점을 맞춘 부분을 IT 지원, IT 서비스 관리를 포함하는 영역으로 확장하여 다양한 프로젝트를 지원할 수 있도록 하였다. 그러나 위험 분류법에 의한 분류는 분류법에 명시된 많은 위

험요소들을 충분히 이해하고 있어야 하기에, 위험 분류가 직관적이지 못하며, 또한 프로젝트 관리자의 관점에서만 위험을 다루고 있어, 다른 이해당사자들과는 위험 식별에 대한 견해의 차이가 있을 수 있다.

[7] 정보시스템감리 점검 Framework 는 일반적으로 감리를 시행하는 논리적인 흐름에 따라 구성된다. 점검 Framework 의 구성은 사업유형/감리시점, 감리영역, 감리관점/점검기준의 세 가지 축으로 구성된다. 그러나 위험을 식별하는 데만 역점을 두고 위험관리 체계가 없기 때문에, 기업의 자체적인 위험관리 방안으로 활용되기에는 미흡하여 참조 모델로만 활용될 수는 있으나 구체적인 적용을 위해 응용할 수 없다.

3. 위험 관리 프레임워크

기존의 위험 관리 방안은 위험을 경험에 의한 주관적인 방법으로 위험을 식별하고 분류하고, 프로젝트의 환경에 의존적인 면을 보였다. 또한, 이해당사자들간의 서로 다른 관점을 고려하지 않은 위험 관리 체계는 관점 및 견해의 차이로 인한 또 다른 위험 요소를 가져올 수 있다. 본 논문에서 제안하는 위험 관리 프레임워크는 이를 해결하기 위해 위험을 시스템 통합 프로젝트의 이해당사자들의 시각에서 정리하고, 위험요인의 발생 시점과 위험관리 대상에 기반한 분류체계를 통해, 시스템통합 프로젝트의 성과에 영향을 미치는 위험관리 요인들을 관점에 따라 체계적으로 분류할 수 있는 모델을 제공한다. 이러한 위험관리 요인들을 효율적으로 식별하기 위해 시간적인 흐름에 따라 구분되는 ‘시점’, 위험 관리 활동의 당사자에 따라 구분되는 ‘관점’, 위험 관리의 대상에 따라 구분되는 ‘대상’의 세 개의 축으로 정의하며, 각 축간의 관계에 의해 삼차원의 육면체로 표현할 수 있다.



(그림 2) 위험 관리 프레임워크

3.1 시점(Phases)

시스템 통합 사업의 경우, 소프트웨어 개발 단계 이전의 사업활동이 이루어진다. 따라서 시스템 통합 사업에서의 위험관리는 소프트웨어 개발단계 뿐만 아니라 비즈니스 단계를 포함해야 한다. 본 논문에서 위험관리의 범위는 영업착수부터 프로젝트의 실행 종료까지의 기간을 다룬다.

시점은 사업착수부터 실행, 종료까지 사업 전기간

위험관리 활동을 의미하며, 영업(Sales)과 실행(Delivery)으로 나뉜다.

영업(Sales) 시점은 사전영업, 제안, 견적, 계약 및 대금수금 활동으로 시스템통합 프로젝트가 착수되기 이전 비즈니스 단계를 나타낸다.

실행(Delivery) 시점은 계약에 정의된 고객의 서비스 요구에 대하여 범위, 일정, 원가, 품질 목표를 달성하는 활동으로 개발 프로젝트가 착수되기 시작하여 종료되기까지의 단계를 나타낸다.

3.2 대상(Object)

대상은 위험관리 활동과 위험 통제 대상의 의미를 의미하며, ISO/IEC 9126 과 1459 을 근거하여 자원(Resource), 절차(Process), 제품(Product)으로 분류한다.

자원(Resource)은 사업 수행을 위한 인적, 물적, 비용적인 요소들이다.

절차(Process)는 자원을 이용하여 제품을 생산하기 위한 방법으로 프로젝트 수행을 위한 관리 및 지원절차와 개발방법론을 포함한다.

제품(Product)은 절차를 통해 생산되는 각종 산출물 및 시스템과 사업성과를 의미한다.



(그림 3) 대상간의 관계

3.3 관점(Viewpoints)

관점은 프로젝트의 이해당사자들의 다양한 위험에 대한 시각을 의미하며, 경영자(Business Manager), 프로젝트 관리자(Project Manager), 엔지니어(Engineer)를 포함한다. 이들은 프로젝트 수행 시 각자 역할과 책임, 지식과 경험이 다르며, 이로 인해 프로젝트의 위험을 바라보는 시각과 인식이 달라진다. 즉, 같은 시점의 위험 대상에 관해서도 서로의 입장이 다르며, 이를 고려하지 않은 채 위험을 정의한다면, 프로젝트를 더 큰 위험에 빠뜨릴 수 있다. 따라서, 위험을 바라보는 다양한 시각을 정리하여 역할과 책임을 정의한다.

경영자(Business Manager)는 사업의 진척상황과 위험요인을 모니터링하여 중요사안을 결정하거나, 전사수준에서 지원하는 자들이다. 경영자(Business Manager)는 의사결정자, 영업대표, 지원조직 등의 역할을 담당한다.

프로젝트 관리자(Project Manager)는 프로젝트의 이해관계자들의 요구와 기대를 충족시키기 위하여 계획 수립, 자원조달, 진행통제를 행하는 자들이다. 프로젝트 관리자(Project Manager)는 프로젝트의 관리자, 품질보증담당자, 형상관리자, 테스트관리자 등의 역할을 담당한다.

엔지니어(Engineer)는 프로젝트가 목표한 서비스 구현을 담당하는 자들이다. 엔지니어(Engineer)는 개발자, 기술지원 등의 역할을 담당한다.

3.4 각 축간의 관계

다음의 표는 시점-관점의 관계를 통해 세부활동들을 정의한다.

시점		관점		
		경영자	관리자	엔지니어
영업	착수	사업참여여부 및 참여방식 결정	의사결정지원	의사결정지원
	제안	제안전략 수립 제안가 확정, 비용 산정	제안서 발표, 공수산정	제안서 작성, 공수산정
	계약	협상팀 구성, 계약서 작성, 계약검토 및 체결	고객협상 실시	고객협상 실시
실행	착수	예산확정, 자원투입	프로젝트 관리, 영역별 계획수립, SPM 정의	업무범위 정의, 개발전략 수립, 수행환경 구축
	진행	성공지표 모니터링	수행공정 및 산출물 평가/정리, 고객승인	SW 설계/개발/테스트
	종료	결과평가 및 개선과제 도출	최종고객 검수, 산출물/교훈 등록	최종고객 검수준비, 산출물 정리

<표 1> 시점-관점 관계

다음의 표는 시점-대상의 관계를 통해 시점 별 담당자 및 절차와 제품을 정의한다.

시점		대상		
		자원	절차	제품
영업	착수	영업대표, 전사지원조직, 사전영업 비용, 사전영업인력	사업참여 결정	RFI
	제안	제안인력, 견적인력	제안/견적	견적서, 제안서
	계약	영업대표, 법무팀	협상/계약	계약서
실행	착수	관리자, 예산, 기술/인프라 (HW, SW, 솔루션 등)	프로젝트 착수	프로젝트 계획서, 품질계획서
	진행	프로젝트팀	요구정의	관리

	개발인프라	분석,설계, 개발,구현	산출물, 공정산출물
종료	프로젝트팀, 운영인프라	프로젝트 종료	시스템

<표 2> 시점-대상 관계

다음의 표는 관점-대상 관계를 통해 관점 별 담당자, 절차, 제품을 정의한다.

관점	대상		
	자원	절차	제품
경영자	영업대표, 지원조직, 의사결정자	영업/지원/통제 절차, IFSS	영업산출물 (제안서, 견적서, 계약서), 사업손익
관리자	예산,시간, PMO (PM,사업관리, 품질관리자)	프로젝트 관리방법론, SFSS	관리산출물 (계획/결과서)
엔지니어	인력(자체,외주), 기술/인프라	개발방법, SPM	공정산출물, 정보시스템

<표 3> 관점-대상 관계

3.5 위험식별 체크리스트

본 논문에서 제안하는 위험관리 프레임워크는 [4] Foo 와 Muruganathan 의 연구와 마찬가지로 위험 분류 별로 체크리스트를 제공하고 있다. 그러나 주요 위험 요인을 단순히 하나의 축으로 분류하고 주관적인 체크리스트를 제공하고 있는 반면, 본 논문에서 제안하는 위험관리 프레임워크는 세 개의 축을 제공함으로써 위험 요인이 보다 가시적으로 표현되며, 다양한 환경에 적용이 가능하다.

기본 점검표는 시점-관점-대상이 만나는 하나의 위험관리 영역에 대한 점검내용을 정의한다.

시점 - 단계				/*영업-착수*/
관점				/*프로젝트관리자*/
대상	핵심목표		/*중요자원은 파악*/	
	자원	중점검토사항	효과성	/*중요자원은 적정한가?*/
안정성			/*중요자원은 확보가능?*/	
절차	핵심목표		/*사업참여의사결정*/	
	중점검토사항	준수성	/*의사결정절차 준수?*/	
제품	핵심목표		/*사업적 가치달성*/	
	중점검토사항	기능성	/*사업가치 실현가능?*/	

	신뢰성	/*사업은 안정적?*/
	효율성	/*사업가치는 얼마?*/

<표 4> 기본 점검표 예시

상세 점검표는 기본 점검표의 하나의 점검 내역에 대해 세부 체크리스트를 구성한다. 상세 점검표의 체크리스트를 통해 위험관리 요소를 점검, 평가한다.

4. 결론

시스템통합 프로젝트의 실패를 줄이고 성과를 높이기 위하여 본 논문에서는 효율적이고 체계적으로 프로젝트 위험요인을 식별하고 관리를 할 수 있는 기반을 제공하였다, 기존의 위험 평가와 관리에 관한 연구는 위험을 주관적인 경험에 의해 분류하거나, 프로젝트 환경에 의존적인 면을 보이고 있어, 기업의 전반적인 위험 관리에는 부족한 면을 보이고 있다. 이를 해결하기 위하여, 시점, 관점, 대상의 세 개의 축으로 위험요인을 분류하여 다양한 관점에서 위험에 대처하고 위험요소를 관리할 수 있는 모델을 제시하여, 기업의 총체적인 위험관리를 가능하게 하였다. 따라서, 본 논문에서 제안된 위험 관리 프레임워크를 시스템통합 프로젝트의 위험관리 활동에 활용함으로써, 위험을 효율적으로 식별하고, 평가하며, 위험관리 내용을 효과적이고 체계적으로 축적하여, 향후 프로젝트의 위험대응 시 기존 위험사례를 분석하여 최적의 위험 해결방안을 제시 가능하게 하는 이점을 제공한다.

참고문헌

- [1] The Standish Group International, 2000, Extreme Chaos 2001
- [2] 쌍용정보통신, 1999, 투자효과분석 및 정보시스템 평가 방법론 개발
- [3] 서창교 외 1 명, 2003, 프로젝트 위험관리가 소프트웨어 개발 프로젝트 성과에 미치는 영향, 한국경영정보학회
- [4] Ronald P. Higuera, 1996, Software Risk Management, CMU/SEI
- [5] Say-Wei Foo and Arumugam Muruganatham, 2000, Software Risk Assessment Mode, IEEE International Conference on Management of Innovation and Technology.
- [6] Tak Wah KWAN and Hareton K.N. LEUNG, 2005, An Enhanced Risk Taxonomy for Information Technology, 9th LASTED International Conference on Software Engineering and Applications.
- [7] 한국전산원, 2007, 정보시스템감리점검 해설서 v2.0