

# 기업 고유환경기반 실제구현을 위한 소프트웨어 프로세스 개선모델 확장

(Expanding SPI Model for Practical Implementation based on  
Industry Characteristics)

김 강 태 †  
(Kangtae Kim)

**요 약** 소프트웨어 개발 성숙도를 향상시키기 위한 프로세스 프레임워크로서 IS 15504(SPICE) 및 CMMI 등은 이미 검증된 모델이다. 하지만 이들 프로세스 성숙도 모델들이 기업의 개발능력 및 효율을 높여주는 구체적인 실천방법을 제시하지는 않는다. 기업 내에서의 실제적인 적용을 위해서는 위의 모델들을 기반으로 제품이나 기술 도메인, 사업 및 조직의 규모 등 기업환경이 고려된 기업 고유의 모델의 개발 및 적용이 필요하다. 본 논문에서는 소프트웨어 개발 프로세스에 대한 수년 간의 지속적인 적용 및 개선 경험을 바탕으로 기존의 SPI(Software Process Improvement) 모델을 확장하여 기업의 비즈니스 목적과 환경에 부합하도록 조직차원의 전반적인 소프트웨어 개발 경쟁력을 강화한 사례를 소개한다. 기존의 SPI 모델이 프로세스 각 분야에 대한 정의와 성숙도를 판단하기 위한 능력 수준을 정의한 것에 비해 본 논문에서는 그 영역을 핵심기술 및 조직구조까지 확대하고, 능력 수준을 일반적인 프로세스 능력 수준이 아닌 조직에서의 모델의 확산도로 확장하여 소프트웨어 개발 경쟁력을 강화하기 위한 통합된 모델을 정의하여 적용하였다. 본 논문은 기업에서 확장된 프로세스 성숙도 모델을 이용하여 조직의 능력을 향상시키는 구체적인 사례로 유사 영역에서 참조 모델로 활용될 수 있을 것이며, 프로세스 성숙도 모델 연구에 대한 기업의 피드백으로의 기여가 있을 것이다.

**키워드** : 소프트웨어 개발 프로세스, 프로세스 성숙도 모델, 프로세스 개선, 프로세스 모델링

**Abstract** IS 15504 and CMMI, etc are already proven models as a framework to improve the maturity of enterprise software development. However, these process maturity models can't present the detail and practical methods with which people can enhance the development competence and efficiency of an enterprise. Each company or organization should develop its own model or tailor the above models to make them suitable to its unique environment such as product or technology domain, scale of business or organization and cultural environment, etc for the practical application. This study introduces experiences that organizational and technical capability was reinforced based on our own process capability improvement model to improve software development strength in Samsung Electronics. We modeled our own improvement model which is expanded from IS 15504 against our experience. Our SPI model expanded its capability to organizational and technical issues including newly introduced capability level for evaluating its implementation. We expect that our study would give contribution for presenting industry experience and reference model for reinforcing software development competence.

**Key words** : software process improvement, software development process, process maturity model, process modeling

## 1. 서 론

소프트웨어 개발 성숙도를 향상시키기 위한 프로세스

프레임워크로서 IS 15504(SPICE) 및 CMMI 등은 이미 검증된 모델이다. 하지만 이들 프로세스 성숙도 모델들이 기업의 개발 능력 및 효율을 높여주는 구체적인 실천 방법을 제시하지는 않는다. 특히 기업 내에서의 실제적 적용을 위해서는 각 기업의 제품 및 기술의 도메인, 사업 및 조직의 규모 등과 같은 비즈니스 환경을 고려

† 정 회 원 : 삼성전자 연구원  
victwin@korea.com  
논문접수 : 2005년 10월 4일  
심사완료 : 2005년 12월 28일

하여 기업 고유의 모델을 개발해야 한다. 다양한 소비자의 요구와 기술의 발전에 따라 전자 제품에서 소프트웨어가 차지하는 비중은 점점 커져가고 있다. 이러한 변화의 속도에 대응하기 위해서 소프트웨어 개발에 대한 기술과 경험이 부족한 전자업체는 전반적인 조직차원에서 소프트웨어 개발 경쟁력을 강화시키는 것이 중요한 당면 과제이다[1].

본 논문은 개발 프로세스를 수년 간 적용한 경험을 바탕으로 프로세스 평가, 개선에 대한 국제 및 시장 표준 모델인 IS-15504나 CMMI 등을 기준으로 기업이 처해 있는 환경과 비즈니스 목적에 부합하도록 조직 차원의 전반적인 소프트웨어 개발 경쟁력을 강화한 사례를 소개하며, 적용 사례를 반영하여 반복성과 일관성을 갖는 혁신활동의 톨로 활용할 수 있도록 개발된 역량평가 모델에 관하여 논한다. 역량평가 모델은 IS 15504 및 CMM-SW를 프레임워크로 하여 프로세스 자체만 아니라 기업의 소프트웨어 개발 경쟁력에 영향을 미치는 요소를 추출하여 추가하였으며, 이는 기존 SPI 모델에 대한 확장의 요소로 생각할 수 있다. 본 논문에서 제시되는 모델은 제품 개발 프로세스와 연계된 소프트웨어 개발 프로세스, 소프트웨어 기술 개발과 지속적인 기술 혁신, 소프트웨어 개발에 대응할 수 있는 유연한 조직 체제, 개발인력의 고급화, 개발 효율성을 높이는 개발 환경에 대응할 수 있도록 개발 되었다.

본 논문은 기업에서 확장된 프로세스 성숙도 모델을 이용하여 조직의 능력을 향상시키는 구체적인 사례로 유사 영역에서 참조모델로 활용할 수 있을 것이며, 프로세스 성숙도 연구에 대한 기업의 피드백으로써의 기여가 있을 것이다.

논문의 구성은 다음과 같다. 1장에서는 기업 내 소프트웨어 개발 역량강화의 필요성과 배경을 제시하고, 2장에서는 관련된 기존 연구를 통해 본 논문이 제시하는 모델의 참조모델 및 기술적 연계성을 보이며, 3장과 4 장에서는 개발 역량강화의 기업 내 실 적용사례를 제시하고, 이를 통해 기존 SPI 모델에서 확장된 역량강화 모델을 정의한다. 마지막으로 5장에서 본 논문의 결론으로 모델 적용의 결과와 장단점, 향후 연구과제를 제시한다.

## 2. 배경 및 관련연구

### 2.1 내장 소프트웨어 경향과 기업 요구사항

CE(Consumer Electronics)업계에서 소프트웨어의 비중은 날로 높아 가고 있다. 새로운 IT기술들이 속속 전자제품에 내장되고 있으며, 제품의 네트워크화, 퍼스널화, 모바일화가 고객은 이러한 기술을 바탕으로 기존의 제품에서 제공하지 못했던 수준 높은 서비스를 요구하고 있다. 이는 결국 전자제품 내에서 소프트웨어가 차지

하는 비중이 점점 커지고 있다는 것을 반증하며, 제품 내에 장착되는 소프트웨어의 복잡도와 크기가 증가한다는 것을 의미한다[2]. 이로 인한 시장 불량 역시 크게 증가하고 있는 실정이다. 결국 기업에서 제품을 개발, 양산 그리고 판매하는 각 단계에서 소프트웨어 개발을 위해 투입되는 자원이 그만큼 급격히 증가한다는 것을 의미한다. 따라서, 품질과 기능적인 면에서 차별화된 소프트웨어가 제품의 경쟁력으로 직접 연결되며, 소프트웨어 개발역량이 곧 회사의 경쟁력이 된다는 것이다[3].

이러한 측면에서 기업에서 소프트웨어 개발역량을 강화시키기 위한 방법은 매우 중요한 과제이며, 이를 위한 실천적인 방법론의 개발이 필요하다. IS 15504, CMMI 등은 이를 위한 기준으로써의 프레임워크를 제공해 주지만 실천적인 방법은 제공하지 못한다. 기업이 처한 내, 외부적인 환경을 종합적으로 고려한 모델을 개발해야 하는 이유이다[4,5].

### 2.2 프로세스 성숙도 모델

IS 15504는 소프트웨어 프로세스를 평가하는 국제 표준 모델로 SPICE(Software Process Improvement and Capability dEtermination)로도 알려져 있다. 소프트웨어 프로세스를 획득, 공급, 개발 등의 5개 영역으로 나누어 각 프로세스에 대한 계획, 관리, 관리 및 제어, 프로세스 개선을 수행하는 프레임워크를 제공한다[6].

CMM(Capability Maturity Model)은 1986년 미국 SEI 에서 개발된 프로세스 성숙도 프로세스 성숙도 프레임워크로 조직의 소프트웨어 개발 프로세스 성숙도를 측정하고, 프로세스 개선 활동을 위한 지침을 제공한다. CMMI는 CMM의 확장 프레임워크로 제품과 프로세스의 개선을 통합한 개념으로 시스템이나 제품을 개발, 유지 보수하는 데 필요한 프로세스를 개선하기 위한 톨로써 개발된, 기존의 여러 성숙도 모델을 하나의 모델로 통합한 프레임워크이다[7].

IS 15504와 CMMI는 프로세스 프레임워크와 성숙도 평가를 프레임워크를 제공한다. 이를 프로세스 수립의 기반으로 활용할 수 있으며, 프로세스 적용 수준을 평가하는 기준으로 활용할 수 있다. 하지만 이러한 표준은 기업에서 적용하기 위한 실행 기준을 제시하지는 않는다. 따라서 기업이 처한 내외부적인 환경과 문화적인 요소 등을 종합적으로 고려한 방법론 개발이 필요하다[4]. 기업에서 소프트웨어 전반에 대한 개선의 도구로 사용하기 위해서는 인력, 조직, 기술 등에 대한 구체적인 실행 지침이 필요하다[8].

### 2.3 모델 기반의 접근 방법

모델이란 시스템, 이론 혹은 현상 등에 대해서 이미 알려져 있거나 내재하고 있는 특성들을 설명하며, 이를 통해서 향후의 연구에 활용할 수 있도록 기술된 것이다.

실세계의 현상에 대한 추상화를 통해서 일반화되지 않을 사소한 현상들을 제거하고, 수 많은 전제사항, 제약사항들을 단순 화합으로써 현상에 대한 이해도를 높이고, 가시화하여 논리화할 수 있다. SE에 대한 모델링 활동의 사례로는 USC(University of Southern California)에서 개발된 MBASE(Model Based Software Engineering) 등이 있다[9]. 본 논문에서는 당사에서 사례로서 실행된 SPI 활동을 바탕으로 전 조직에 적용할 수 있는 전사적인 차원에서의 방법론을 개발에 모델링의 개념을 도입 하였다.

**3. 개발경쟁력 강화 활동: 적용 사례**

**3.1 사례 1: 표준 프로세스 수립 및 전개**

기업에는 구매, 마케팅, 개발, 제조 등 많은 프로세스가 존재하며, 각 프로세스는 유기적인 관계를 가지고 병렬적으로 운영된다. 본 논문에서 다루는 소프트웨어 프로세스는 당사의 7대 프로세스 중 개발 프로세스 내에 정의되는 프로세스로 한정한다. 당사의 표준 프로세스는 제품 개발 프로세스를 기준으로 IS 12207과 IS 15504를 참조하여 개발되었다. IS 15504의 총 48개 프로세스 중에서 기업 차원에서 개발환경에 반드시 필요한 필수 프로세스 요소로써 8개의 프로세스를 선택하였고, IS 15504의 레벨 3의 성숙도 수준으로 개발되었다. 전사 차원의 표준 프로세스는 각 개발 조직에 배포되어 해당 조직의 프로세스에 대한 기준으로 활용되고, 제품 개발 과정에 적용되었으며, 그 적용 결과는 다시 해당 조직 및 전사 표준 프로세스로의 피드백을 통해 프로세스 개선이 이루어졌다. 프로세스 개선이 기업의 생산성, 품질 향상에 효과적이라는 것은 기존의 여러 연구들을 통해서 이미 검증된 바이다[3,10].

프로세스가 각 개발조직에 전개되는 과정은 점증적 접근방법이 적용되어, 일부 시범과제를 중심으로 적용성 평가 후 전 개발 조직으로 확산되었다. 이를 추진하기 위한 톨로 IS 15504 심사를 활용하였다. 시범과제를 통한 조직의 프로세스 전개 방법은 기존의 연구에서 제시된 성공적인 접근방법이다[11].

IS 15504 심사는 2000년 이후 총 37 건이 수행되었으며, CMMI 심사는 1건이 수행되었다. 심사 대상 프로세스는 전사적 차원에서 엔지니어링, 과제관리, 형상관리, 품질보증 등의 8개 프로세스를 정의하여 권고하였다. 2004년까지 수행된 심사를 분석하면 위의 8대 프로세스는 모든 심사에서 대상 프로세스로 선정되었으며, 그 밖의 프로세스로 품질관리, 확인, 검증, 외주관리 프로세스들이 포함되었다. 또한, 당사 자체적인 성숙도 개념을 정의하여 표 1과 같이 개발 조직의 프로세스 전개 및 내재화 수준을 중심으로 도입(수준 2), 전개(수준 3),

표 1 프로세스 수준에 따른 레벨링 정책

수준	추진 정책	IS 15504 정의	추가 정의
0		Incomplete	N/A
1	전사적 차원의	Performed	
2	최소 요구 사항	Managed	Introduced
3		Established	Deployed
4	각 개발 조직의	Predictable	Systemized
5	선택 사항	Optimizing	Institutionalized

체제화(수준 4), 내재화(수준 5)의 5 단계로 운영하였다.

위와 같은 일련의 활동은 전사 차원의 SE(software engineering)조직과 각 개발조직 별 SE 조직이 구성 되어 수행되었고, 표준 프로세스는 각각의 조직의 SE 조직에 의해서 지속적으로 변화 관리되었다.

**3.2 사례 2: 조직 특화 프로세스 전개**

표준 프로세스의 전개는 단위 개발 조직 별 프로세스 전개와 독립하여 진행될 수 없다. 표준 프로세스는 개발 조직에서 그대로 사용할 수 있는 실행 차원의 프로세스가 아니며, 단어의 의미 그대로 전체적인 프레임워크만 정의하고 있기 때문이다[12]. 따라서 각 조직은 조직 별로 개발 프로세스에 대한 테일러링과 내재화 활동을 수행해야 한다. 이러한 각각의 개발 조직에 대한 테일러링 및 내재화 작업들이 진행되지 않는다면 그 표준 프로세스는 사용될 수 없는 사장된 프로세스이기 때문이다 [13].

일련의 프로세스 수립 및 전개 활동은 각 개발조직의 비즈니스 목표, 제품 개발 목표에 따라 그 형태가 달라지게 된다. 또한 각 조직의 프로세스 성숙도가 각각 다르기 때문에 조직의 프로세스 역량에 따라 독립적인 접근방법이 필요하다[14,15]. 각 개발조직이 고려해야 할 요소들로 프로세스 정의와 개발 조직의 조직적 위상 및 크기를 중요 요소로 관리하였다. 비즈니스 목표와 사업의 영역에 따라 각 개발 조직 별로 수립, 전개해야 하는 프로세스 구성요소가 다를 수 있기 때문에 전사 프로세스를 기반으로 대상 프로세스에 대한 조정이 필요하고 이에 대한 주요 요소는 무엇보다 조직적인 요소이다. 그 이유는 개발조직의 조직구성 형태에 따라서 프로세스의 형태가 달라질 수 있기 때문이다. 즉, 기업 표준 프로세스를 각 개발조직에서 실제 구현할 때 각 개발조직이 어떻게 구성되어 있고 각 프로세스에서 정의하고 있는 활동의 책임이 어떤 조직에 있는가에 따라서 프로세스가 조정된다는 것이다.

위의 과정을 통해 정의된 프로세스에 대한 적용관리를 위해 개발조직 장에 의해 주관되는 개발 혁신 협의체를 구성하여 주기적으로 개발 프로세스의 적용 현황, 문제점 및 개선방안을 도출하는 활동을 수행하였다. 표

표 2 주기적인 관리 항목

분류	항목	내용
프로세스	프로세스 적용 현황	과제별 진척 관리, 필수 산출물 작성 현황, 과제 이슈 공유
	프로세스 문제점 발굴	과제 진행 시 발굴된 문제점 공유
인프라 스트럭처	형상관리	과제별 산출물의 형상관리 현황 보고
	품질관리	과제 단계별 Defect 추이(발생, 해결) 보고
	Tool	상용 Tool 분석 및 확산, 자체 개발 Tool 검토
정보공유	벤치마킹	타 사업부 및 전사 우수사례

2는 개발혁신 협의체에서 다루는 이슈이다. 조직 장에 의한 주기적인 모니터링 활동은 프로세스 전개 및 SPI 활동에 가장 중요한 조직장의 개선의지를 전달하여 조직원에게 동기 부여를 하는 효과적인 방법이다[16].

3.3 사례 3: 소프트웨어 개발체계 재정비

CE 제품군별로 내장 소프트웨어의 중요성이 급격히 증가하는 분야가 있다. 위와 같은 상황에 처해있는 조직들은 갑작스러운 소프트웨어의 기술적 진보와 이에 따른 조직적인 변화에 적응하기 위한 중요한 시기를 보내고 있다. 이와 같은 조직에서의 소프트웨어 혁신은 비단 프로세스에만 국한된 것이 아니다. 물론 광의로 본 프로세스는 조직과 기술 등을 모두 포함하는 매우 넓은 의미이지만 기업에서는 프로세스와 조직, 그리고 기술을 각각 독립적인 차원에서 관리하는 것이 혁신의 결과를 보다 빨리 얻을 수 있는 길이다. 특히 조직적 정비와 기술혁신의 활동이 더욱 강조된다. 특히 위와 같은 개발 조직은 개발인력이나 조직 위상 측면에서의 개선 점이 많이 발생한다. 소프트웨어 개발 인력 부족과 조직 내 개발자의 비전 및 커리어패스가 미약하고 교육 체계가 매우 부실하며, 소프트웨어 개발 조직의 위상이 제품의 비중이 일치하지 않는 경우가 많다[15].

또한 소프트웨어 공학 업무(SPI: Software Process Improvement, SQA: Software Quality Assurance, Infrastructure 등)를 전문적으로 수행하는 SE 인력부족도 문제점 중의 하나이다. 따라서 이에 대한 별도의 조직적인 정비작업이 수행되었다. 개발 환경에 대한 정비 차원에서 전사 차원의 표준 툴을 선정하고, 이에 대한 기술지원을 수행하는 활동도 개발 체계 정비의 일환으로 진행되었다.

이와 같은 개발체계의 정비 활동은 조직의 현재 개발 역량을 파악을 위해 외주현황, 칩셋 사용 현황, 인프라 현황, 인력현황 등에 대한 종합적인 평가와 더불어 기술 분석을 수행하였다. 현재의 기술과 필요기술을 분석하여 기술 로드맵을 작성하였고 동시에 관련 업계의 기술 트렌드를 분석하여 확보해야 할 필요 기술을 정의하였다.

이와 같은 활동을 수행한 후 개발조직의 체계정비를 위한 전략으로 그림 1과 같이 역량강화 방안과 전사 차원의 협업과제가 도출되었고, 그림 2와 같이 인력/조직,

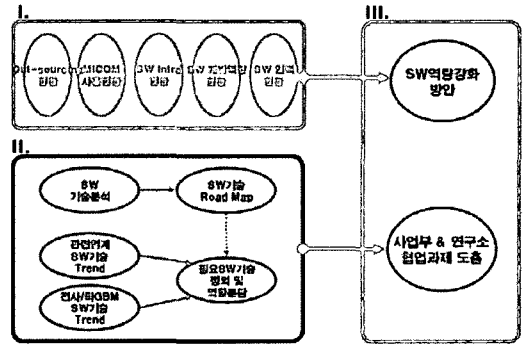


그림 1 조직의 개발체계 정비를 위한 전략 수립

기술력, 개발 역량의 3대 혁신항목과 연도별 추진 목표를 수립하고 추진하였다.

3.4 사례 4: 프로세스 내제화를 위한 관리 활동

프로세스는 기술적인 측면과 문화적인 측면이 결합되어 있다[19]. 국내의 기업 문화와 업무 추진 방향은 하부 조직으로부터의 접근보다는 상부 조직(혹은 중앙 조직)의 방침에 따라서 전 개발조직이 하나의 방향으로 접근하는 방식이 주를 이루고 있다. 조직의 프로세스를 내제화 가는 방법 역시 이러한 하향식 방법을 취하며, 중앙조직은 기업 표준 프로세스를 수립하고 전개해 나가는 방향 수립과 함께, 추진 사항을 모니터링한다. 각 개발 조직의 프로세스 적용 및 그 수준을 파악하고 개선 하여 지속적인 프로세스 내제화 활동을 수행한다. 이러한 내제화 활동은 각 개발 조직에 대한 주기적인 프로세스 심사활동을 통해서 이루어진다. 프로세스 내제화를 위한 주기적인 관리활동은 장기적인 관점에서 강력한 관리 활동 없이는 SPI의 효과가 지속되지 않는다는 기존의 연구를 통해서도 그 필요성을 확인할 수 있다[17].

그림 3과 같이 전사 차원에서 현재의 수준을 파악하고 전사의 프로세스 내제화 지수를 적용율, 준수율 및 충실도 등의 메트릭을 통해 산출하였다. 개발 조직 별로 프로세스 내제화 지수를 산출하고 이를 전사의 지수와 비교하여 각 개발조직 별 취약점을 도출하였고, 개발 조직에 대한 인적 현황을 파악하며, 소프트웨어 개발 툴 및 관리 툴 등의 개발환경에 대한 전반적 현황을 주기적으로 측정, 분석하였다. 이러한 활동은 각 개발 조직

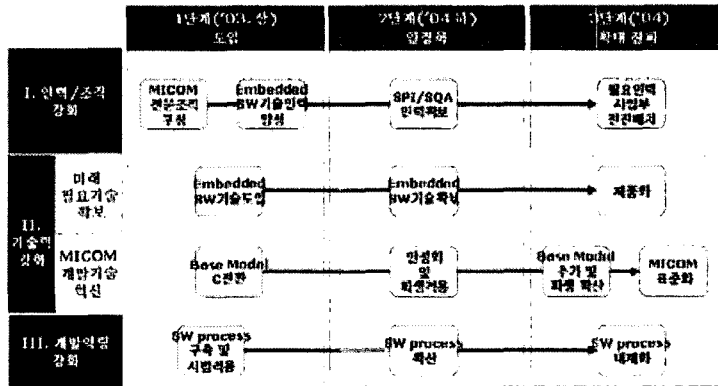


그림 2 조직의 개발체계 정비 활동

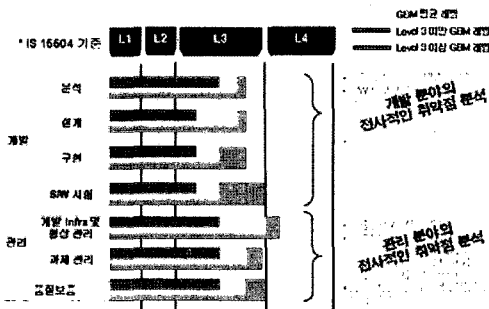


그림 3 전사 차원의 프로세스 적용 수준 파악

에 대한 차원의 현황 파악을 통한 프로세스 내제화 활동이다.

### 4. 개발 역량 모델: SPI 모델의 확장

#### 4.1 개발역량 요소 및 모델 개요

기업에서 소프트웨어 개발 프로세스를 수립하고 적용하는 목적은 이를 통해 제품의 품질과 생산성을 높이며, 궁극적으로는 기업의 이익을 높이기 위함이다[3]. 각 개발 조직은 제품을 개발하기 위한 총체적인 능력을 강화함으로써 상기의 목표 달성을 추구한다.

본 논문의 3장에서는 프로세스 강화를 위한 기업의 활동을 사례로써 제시하였다. 본 장에서는 3장의 사례를 바탕으로 소프트웨어 개발 역량의 요소를 정의하고 이

를 강화하기 위한 모델을 제시한다. 본 논문에서 제시되는 모델은 기존의 SPI 모델을 기반으로 하고 있으며, 사례를 통해 발굴된 개발 역량의 요소를 확장된 개념으로 제시한다. 개발 역량 요소는 표 3과 같다.

소프트웨어 개발 역량 모델은 개발조직에서 개발역량 강화를 위한 참조모델과 이에 대한 실행 지침으로 사용된다. 모델을 개발하는 이유는 기업에서 SPI 및 개발 역량 강화를 위해서는 무엇을 할 것인가 보다는 어떻게 할 것인가에 대한 지침이 더 필요하기 때문이다[18].

#### 4.2 모델 설계

모델 개발을 위해 모델의 목적과 구성요소를 정의하고 각 구성요소 별 세부 실행절차를 정의하여 각 개발 조직에서 소프트웨어 개발역량 향상에 적용될 수 있도록 개발 되었다. 이를 이용하여 개발조직의 프로세스 능력을 평가하고 개선안을 도출하며, 신규 확보해야 할 소프트웨어 기술을 분석하여 기술력의 향상을 추구하며, 이를 추진하는 개발조직의 구조와 개발인력의 역량을 강화한다.

개발역량 모델은 그림 4와 같이 크게 5개 부분으로 구성되어 있으며, 각각은 영역 모델(Domain model), 프로세스 평가 모델(Process Evaluation Model), 조직 및 인력 모델(Organization & People Model), 기반 환경 모델(infrastructure Model), 기술 모델(Technology Model)이다.

표 3 소프트웨어 개발 역량 요소

분류	항목	내용
기술	소프트웨어 기술	제품 개발에 필요한 소프트웨어 개발, 테스트 기술 등
	지속적인 기술 혁신	기반기술 및 미래대비 기술의 확보를 위한 기술기획 및 선행연구 등
조직	개발 조직체계	개발조직, 테스트조직, SE 조직 별 조직위상 및 구조의 유연성, 효율성
	인력	개발인력의 구성, 개인별 기술 보유, 기술강화를 위한 교육 프로그램 등
인프라 스트럭처	프로세스	효율적인 프로세스의 내제화, 프로세스의 시스템화 등
	도구	개발 및 관리 툴

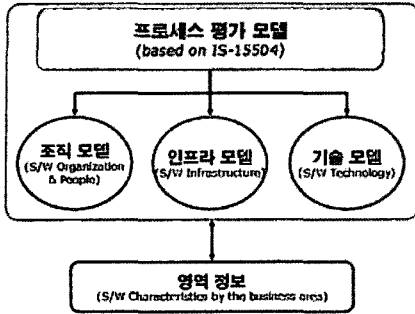


그림 4 소프트웨어 개발 역량 모델

영역 모델은 모델 적용 시 필요한 기본 정보를 제공하는 모델로 각 개발조직의 환경정보를 가지고 4개의 세부 모델을 조정하는 지침(테일러링 가이드)으로 활용된다. 각 모델들은 상세 분석을 위해 세부 항목으로 구분되며, 모델의 적용 결과 각 항목에 대한 현 상태 파악, 차이점 분석, 이슈 분석 등이 수행되고, 각 항목 및 모델 별 추진사항을 추출하여 역량강화를 위한 실천적인 방법을 제시한다.

각 세부 모델들은 3장에서 제시된 적용 사례를 바탕으로 설계되었으며, 관련사례의 상세 내용과 세부모델과의 관계는 표 4와 같다. 프로세스 평가 모델을 제외한 조직, 기술, 기반환경 모델 등이 기존의 SPI 모델에 대한 본 논문이 제시하는 확장 요소이다.

4.3 프로세스 평가모델

소프트웨어 개발역량은 그 조직이 가지고 있는 프로세스 능력, 인적 자원, 기술력에 의해 판단될 수 있다. 당사의 표준 프로세스 모델은 IS 15504를 기초로 한다. 표 1의 프로세스 수준에 의한 전사 적용 정책에 의해 최소한의 기준을 제시하는 기본 모델과 기본 모델에서 프로세스 영역을 확장하고 성숙도를 향상시키기 위한 확장 모델을 정의한다.

기본 모델은 IS 15504의 총 48개 프로세스 중에서 현

재의 개발 환경에 필수적인 14개 프로세스를 선정 하였으며, 14개 프로세스가 상호유기적으로 보완하여 프로세스 성숙도 레벨 2 이상의 수준을 확보할 수 있도록 설계 하였다. 개발 프로세스인 Eng. 4~9를 선정하고, 최근 급증하고 있는 소프트웨어 외주 개발에 대응하기 위해 습득 (ACQ.4) 프로세스를 선정하였다. 관리 프로세스 영역에서는 3 대 주요 프로세스 영역으로 과제 관리 (MAN.3), 형상관리(CFG.2.3), 품질보증 (QU A.1) 프로세스를 필수로 선정하였다. 이밖에 표준 프로세스의 관리 및 개선을 위한 PIM.1~3 프로세스를 포함 하였다. 기본 모델은 그림 5와 같이 IS 15504의 10 개의 프로세스 그룹 중 6개의 프로세스 그룹에 대응 하고, 48개 프로세스 중 14개의 프로세스에 대응하도록 설계되었다.

확장모델은 기본 모델을 포함하며 멀티 사이트 개발에서 필요한 제품 릴리즈 프로세스(RIN 1, 4), 품질과 생산성 향상을 위해 필수적인 재사용 프로세스(REU.1), 그리고 품질관련 프로세스를 보장하여 9개 프로세스 그룹 및 25개 프로세스로 확장한 모델이다. 확장 모델은 이미 표준 프로세스를 도입했거나 개발 분야의 경쟁력이 어느 정도 갖춰졌다고 판단 되는 개발 조직을 대상으로 적용하며, 그 판단은 각 개발 조직에 대한 사전 프

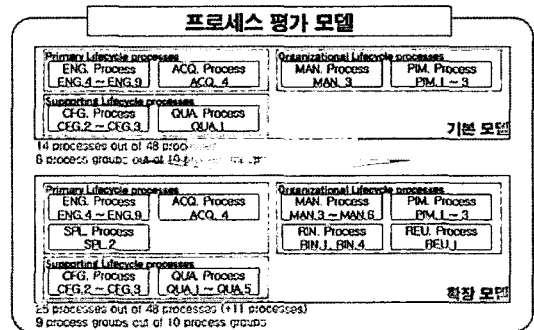


그림 5 프로세스 평가 모델

표 4 소프트웨어 개발 역량 요소

세부모델	관련 사례	참조 내용
프로세스 평가	사례 1	기업 표준 프로세스 정립
	사례 2	개발 조직 별 프로세스 역량 심사(IS-15504)
	사례 2	조직 프로세스 역량에 따른 차별적 SPI
	사례 4	프로세스 내제화를 위한 정기적인 관리활동
조직 및 인력	사례 3	SE 인력의 보강
	사례 3	소프트웨어 조직 재정비
	사례 3	소프트웨어 개발자 커리어 패스 개발
기술	사례 3	소프트웨어 교육체계 강화
	사례 3	경쟁업체 대비 낙후 기술 확보
	사례 3	지속적인 기술 혁신
기반 환경	사례 4	기업 표준 툴 선정 및 기술지원
	사례 4	개발 툴 및 관리 툴에 대한 주기적 점검
영역	사례 2	비즈니스 및 개발 목표에 따른 SPI

로세스 성숙도 평가를 통해서 결정한다.

4.4 확장 모델: 조직, 기술, 인프라스트럭처

프로세스 평가모델이 역량 강화 모델의 기본 뼈대의 역할을 한다면, 당사의 적용사례를 통해서 기존 SPI 모델을 개선하는 확장의 개념은 본 절에서 제시되는 확장 모델에 정의된다. 확장 모델은 그림 6과 같이 조직 모델, 기술 모델, 인프라 모델로 세분화되어 정의 된다. 조직 모델은 소프트웨어 개발 조직과 구성원에 대한 평가 및 분석을 위한 모델이다. IS 15504나 CMMI와 같은 프로세스 모델은 이미 개발 조직에 대한 프로세스 영역이 정의되어 있다. 하지만 3장의 사례 연구에서 볼 수 있듯이 개발 역량을 강화하기 위해서는 조직에 대한 이슈가 중요하기 때문에 별도의 모델로 분리하여 정의한다. 조직 모델은 1. R&D 인력대비 소프트웨어 개발 인력 2. 소프트웨어 개발 경력에 따른 인력구조, 3. 소프트웨어 기술에 따른 구조, 4. 소프트웨어 개발 조직 5. 소프트웨어 교육 체계의 5개 영역으로 나누어진다. 각각의 영역에 대해서 현재 상황과 목표치, 그리고 그들 간 차이점을 분석한다.

R&D 인력대비 개발 인력을 통해서 직접적으로 소프트웨어 개발 인력 수를 파악함과 동시에 각 개발 조직에서의 소프트웨어 중요도를 간접적으로 나타내는 지표로 활용할 수 있다. 경력 별 인력구조를 통해서 개발 인력의 경력별 분포를 파악할 수 있으며 이를 통해 인력 분포에 대한 현황을 파악할 수 있다. 예를 들면, 특정 개발 조직에서 아키텍처급 인력이 부족하다고 가정한다면, 이로 인해 전문기술을 확보하고 기술 경향을 정리할 수 있는 핵심 리더급 인력이 부족하다는 점을 분석해낼 수 있다. 본 모델에서의 핵심은 바로 위와 같이 기본적인 인력구조를 기반으로 경력이나 기술과 같은 핵심요소가 함께 고려된 보다 상세화된 인력구조에 대한 현황을 분석하고 이를 토대로 한 보다 상세한 접근 방법을 제시하는 것에 있다.

인력구조 분석에서 중요한 또 하나의 요소는 기술 분야 별 인력구조이다. 이는 개발조직에서 현재 혹은 향후 필요로 하는 기술에 대한 정의와 그 기술을 보유하고 있는 인력확보 측면에 대한 시각이다. 그렇기 때문에 이 항목은 기술모델에 의존적이며, 기술모델에서 제시되는 소프트웨어 기술 로드맵과 더불어 미래기술 확보를 위한 주요 요소이다. 이 두 항목을 통해서 역량 강화 모델의 주요 산출물 중의 하나인 핵심 소프트웨어 분야 및 관련인력과 기술확보 방안을 산출한다. 개발 조직 구조는 개발 조직의 조직적 위상과 관련 조직과의 인터페이스 체계를 파악한다. 또한 기술개발 및 관리를 지원하기 위한 전문적인 조직으로써 양산 과제에 투입되지 않는 연구조직, 인프라 조직 및 프로세스 조직 등 개발 간접 조직에 대한 시점을 제시하고 역량강화의 주요 대상으로 삼는다. 분석된 현재의 상황과 개발조직의 목표 간의 차이는 결국 인력의 충원, 교육에 의해서 좁혀질 수 있으며, 이와 같은 환경 조성을 위해 소프트웨어 기술 교육체계가 정의된다. 3개 요소인 개발자 구성, 경력 구조 및 기술 구조는 개발자 구성에 해당하며 각각의 모델은 기본적으로 개발자 구성에 대한 확장 정보의 역할을 수행한다. 이와 함께 조직구조와 교육구조를 통해 개발조직의 특성 파악하고 이를 통한 문제점 도출이 가능하다. 본 모델의 주요 산출물은 1. 각 항목에 대한 현재 상태 2. 각 항목에 대한 차이점 분석(목표 및 추이분석 포함) 3. 인력계획 및 조직변경 계획 4. 소프트웨어 조직 및 인력에 대한 주요 이슈 사항이다.

기술모델은 개발조직의 현재 기술수준을 파악하고 중장기적으로 확보해야 할 필요 기술을 정의하기 위한 모델로 조직의 기술역량 강화를 위한 모델이다. 특히 조직 모델과 상호작용을 통해서 기술과 인력 간의 유기적인 시너지 효과를 낼 수 있어야 한다. 기술 모델의 하위 항목으로 표 5와 같이 기술 분류체계, 기술 로드맵, 재사용 기술이 정의된다. 기술 분류체계는 조직에서 소프트웨어 기술에 대한 프레임워크로 기술 정의, 인력 확보, 기술 로드맵 작성을 위해 필요한 기술 들의 위상 구조 및 연관 관계를 정의하는 요소이다.

개발 조직에서 개발하는 소프트웨어에 대한 정의를

표 5 기술모델 요소

Category	Description
기술분류체계	제품군에 따른 소프트웨어 분류체계
	소프트웨어 기술(복잡도) 및 Architecture에 따른 소프트웨어 분류체계
기술 로드맵	확보 및 필요기술
	중장기 기술개발 전략
재사용 기술	조직의 재사용 레벨 및 레벨 별 추진 전략과 필요 기술

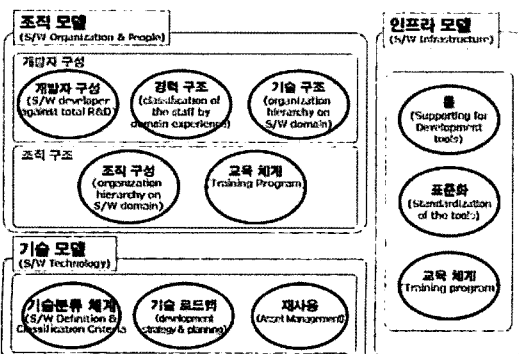


그림 6 확장 모델

시작으로 개발되는 제품군에 장착되는 소프트웨어를 아키텍처, 조직, 기술 등에 의해 분류하는 기준으로 활용된다. 기술분류체계를 바탕으로 두 번째 요소인 기술 로드맵은 기술 분류체계를 기준으로 조직이 현재 확보하고 있는 기술을 파악하고 경쟁사 대비 열등 기술과 미래 확보 기술을 분석하고 이를 토대로 중장기 기술개발 전략을 작성할 수 있도록 한다. 재사용 기술은 제품/기능의 용, 복합화에 따라 공통 개발 영역을 정의하고 제품간, 개발조직 간 공유할 수 있는 재사용 단위를 개발하고 이를 관리하기 위한 기술 요소이다. 조직의 재사용 레벨을 평가하고 각 레벨에 맞는 재사용 정책수립 및 기술확보를 가능케 한다. 이를 통해 얻을 수 있는 이점은 조직 간 중복개발 방지, 보유 강점기술을 상호 활용 등을 개발 효율을 높이는데 있다.

인프라 모델은 소프트웨어 개발에 필요한 툴의 표준화 및 활용현황을 평가하고 이에 대한 개선점을 도출하는 모델이다. 세부 항목으로 1. 툴 사용현황 2. 툴 표준화 현황 3. 기술지원 현황(능력)으로 구성된다. 도구 사용현황은 확보된 툴에 대한 수요/공급을 파악한다. 툴 표준화 현황은 개발조직에서 지정한 전사 표준 툴에 대한 확산도를 파악한다. 개발 툴의 효과적인 적용을 위한 기술 지원이 어떻게 이루어지고 있는 지를 인력과 확보 기술을 통해서 파악한다. 또한 설치 및 유지보수에 대한 기술적 지원을 수행하기 위한 환경 구축 및 운영현황을 파악한다.

5. 결론

5.1 모델 적용 절차 및 결과

본 논문에서 제시된 역량강화 모델은 지금까지 당사에서 수행해왔던 소프트웨어 조직의 개발역량 강화활동을 기초로 개발된 것으로 과제적용을 통해 개선되고 있

다. IS 15504를 이용한 SPI 활동의 한계를 극복하고 각 조직에서의 적용성을 높이기 위해 방법론적 측면에서 모델화했으며 이를 개발조직에 적용하였다.

역량강화 모델을 적용하기 위해 2가지의 입력 정보를 정의한다. 하나는 조직 및 유관분야(선진사 혹은 유사 조직)에 대한 정보를 정의하는 환경정보이고, 다른 하나는 해당 개발 조직에서 본 모델을 적용하는 목적 및 필요성이다. 이 두 가지 입력사항은 본 모델을 개발조직에 적용할 때 필요한 조정활동(모델 구성요소의 테일러링 및 결과에 대한 테일러링)을 위한 기초정보로 활용된다. 모델을 적용한 결과는 4장에 제시된 각 하위 모델의 구성요소에 대한 현재상태와 설정된 목표와의 겹 분석 그리고 이를 극복하기 위해 각각의 항목을 아이템화한 실천사항들이다. 모델의 입력정보 및 개발조직의 환경에 따라 핵심 추진요소를 별도 정의한다. 그림 7은 특정 조직을 대상으로 본 모델을 적용하기 전과 후의 현황을 비교한 것으로 핵심 기술 측면에서 C/Assembly 언어를 기반으로 하는 컴포넌트화 기술 및 ARM 기반의 RTOS 도입의 성과를, 조직 경험의 측면에서는 소프트웨어 개발 조직의 독립화, 개발 인력의 확충과 SPI 및 SQA 등 SE인력의 확충 등의 성과를, 프로세스 측면에서는 IS 15504 레벨 2 수준의 프로세스 성숙도와 개발 프로세스 상 소프트웨어 품질보증 활동을 제품개발 프로세스의 초기단계부터 적용하는 결과를 거두었다.

Tracy 등에 의해 수행된 기존 연구에서는 기업에서의 프로세스 개선활동에 대한 성공요소로 SPI 인력의 확보와 경영자의 후원을 꼽고 있다[1]. 본 역량강화 모델의 적용에서도 혁신업무를 수행하는 SE인력의 확보 및 그들의 역량이 가장 중요한 핵심요소였으며, 지속적인 장기적 투자에 대한 경영층의 관심과 후원이 가장 중요한 환경요소였다. 당사의 소프트웨어 개발 역량 강

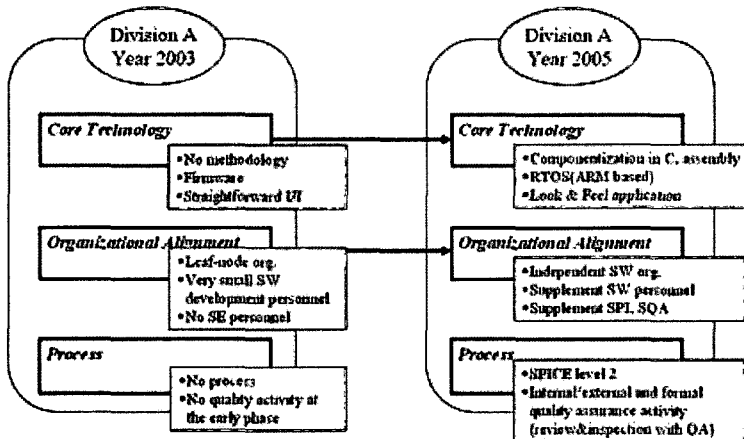


그림 7 역량강화 모델의 적용 결과



화의 수행은 5년간 SPI 인력이 10배 이상의 규모로 커졌으며, SQA 및 소프트웨어 인프라 인력도 그 이상으로 확장되었다는 것으로 대변될 수 있다.

**5.2 향후 연구과제**

본 논문에서 제시된 모델은 일반 현상을 추상화하여 4개의 세부 상세 모델을 정의하였으나 각 모델 간 상호연관관계는 제한적이다. 실 세계에서는 각 모델 간의 상호연관 관계가 매우 복잡할 수 있으나 본 모델은 가장 기본적인 연관관계 외에는 세부 모델 간 연관관계를 모델링하지는 않았다. 각 세부 모델에서 발생할 수 있는 모델간 불일치는 지속적으로 보완, 개선할 예정이다.

개발 프로세스를 포함하여 개발 역량을 강화시키는 것은 엔지니어적인 접근과 더불어 문화적인 접근도 필요하다. 특히 중앙집중적인 한국의 기업 문화에서 중앙조직에 의한 하향식 혁신은 단기적인 성과 창출에는 효과적이나 장기적인 전 조직으로의 확산을 위해서는 조직원들의 자발적인 참여를 통한 상향식 혁신도 필수적이다. 결론적으로 개발 역량 강화를 위한 시작은 하향식 접근을, 확산을 위해서는 상향식 접근을 조화할 필요가 있다. 기업의 개발역량 강화를 위한 기술적, 문화적 접근방법과 상향식, 하향식 접근의 융합에 대해서는 추가적인 연구가 필요하다.

또한, 지금까지 개발 프로세스를 적용하는 것을 비롯한 개발역량 강화 활동에 대한 계량화된 지표의 운영과 이를 통한 정확한 성과측정이 불가능하였다. 기존 SPI에 대한 연구를 통해서도 개선활동에 대한 효과 측정이 매우 중요한 요소로 파악되고 있다[8]. 향후 본 모델의 개선 포인트는 계량적인 관리를 위한 지표 및 지표 운영 프로세스가 될 것이다.

**참 고 문 헌**

[1] Hall T, Rainer A, Baddoo N, "Implementing software process improvement: An Empirical Study," Software process improvement and practice, John Willey & sons. 7:3-1510, 2002.  
 [2] VDC(Venture Development Corporation), "Global Markets and Application for Embedded Software," 2004.  
 [3] Humphery WS, Snyder TR, Wiillis RR, "Software process improvement at Hughes Aircraft," IEEE software 8(4):11-23, 1991.  
 [4] McDermid JA, Bennett KH, "Software Engineering Research: A Critical Appraisal," IEE proceedings-software 146(4): 179-186, 1999.  
 [5] Kangtae kim, Taesik kim, "Reinforcing the S/W development competence by the SCR(M Samsung's S/W Competence Reinforcement Model)," Proceeding of SERA 05, IEEE, 2005.  
 [6] ISO/IEC 15504-1:2004, "Information technology --

Process assessment -- Part 1: Concepts and vocabulary," 2004.  
 [7] Paulk MC, Weber CV, Curtis B, Chrissis MB, "The capability maturity model: Guidelines for improving the software process," Addison-Wesley Publishing, Carnegie Mellon University Software Engineering Institute, 1995.  
 [8] Paulk MC, "Analyzing the conceptual Relationship between ISO/IEC 15504 (Software Process Assessment) and the capability maturity model for software," Software Engineering Institute Carnegie Mellon University Pittsburgh PA, 1999.  
 [9] Boehm B. and Port D., "Escaping the Software Tar Pit: Model Clashes and How to Avoid Them," Software Engineering Notes, Association for Computing Machinery, pp. 36-48, January, 1999.  
 [10] Johnson A, "Software process improvement experience in the DP/MIS function," Proceedings of IEEE International Conference on Software Engineering, 1994.  
 [11] El Emam K, Briand L, "Lister costs and benefits of software process improvement," Technical report ISERN-97-12, Fraunhofer institute for experimental software engineering, 1997.  
 [12] Humphrey WS, "Why don't they practice what we preach?," Annals of software engineering 6:201-222, 1998.  
 [13] Kalito T, Kinnula A, "Deploying the defined software process," Software process improvement journal 5: 65-83, 2000.  
 [14] Paulk MC, Chrissis MB, "The november 1999 high maturity workshop," Software Engineering Institute Carnegie Mellon University, 2000.  
 [15] Hall T, Baddoo N. "The software process improvement paradox: Approaches to quality management," British computer society publication, BCS, 97-107, 2000.  
 [16] Butler KL, "Process lessons learned while reaching level 4," CrossTalk 10:1-6, 1997.  
 [17] Paulish DJ, Carleton AD, "Case studies of software process improvement measurement," IEEE computer 27(9) 50-57, 1994.  
 [18] Herbsleb JD, Goldenson DR, "A systematic survey of CMM experience and results," 18<sup>th</sup> international Conference on Software Engineering -ICSE, Berlin Germany 25-29 March, 323-330, 1996.  
 [19] Austen R, Tracy H. "An analysis of some 'core studies' of software process improvement," Software process improvement and practice 6:169-187, 2001.



김 강 태

2001년 2월 중앙대학교 공과대학 컴퓨터 공학과 공학박사(소프트웨어 공학 전공)  
2001년 3월~현재 삼성전자 책임연구원 (SPI, S/W reuse). 2004년 3월~현재 한국 프로세스 심사인 협회(KASPA) 운영 이사. 2005년 8월 SERA 2005 program committee. 관심분야는 Software process improvement, software economics, software product line engineering