

프로세스 이행 수준과 개발 기술 품질 관리 모델

Quality Management Model for
Process Performance Level and Development Technology

박정현 (J.-H. Park, jh-park@etri.re.kr)	품질보증연구실 책임연구원
박영식 (Y.-S. Park, ysp@etri.re.kr)	품질보증연구실 책임연구원/실장
정효택 (H.-T. Jung, htjung@etri.re.kr)	품질혁신실 책임연구원/실장
김상훈 (S.-H. Kim, ksh@etri.re.kr)	음성지능연구그룹 책임연구원/PL

This paper describes a project - based quality management model that identifies development technologies and codes while at the same time verifying the ability to implement processes that are essential in R & D projects. In order to verify the process implemented in the R & D project implementation, there are review, checking points, and evaluating methods the process performance levels for five processes such as defining requirements which is the beginning stage of system development, testing process which is the completion stage of system development, and project management and peer review process for project management and support in the proposed project-based quality management model. For development technology and code validation, the model included the documented test cases for each requirement by the developer in the requirements definition stage, debugging and testing in the design and implementation stages, static analysis and open source licence verification procedure, and system environment. After applying the model in SW development R&D project for evaluating the process performance, and verifying the development technology and the code, the developers responded that the improvement in the development technique and the code, and upgrade of process performance level for project are more than 10%.

* DOI: 10.22648/ETRI.2017.J.320612

* 본 연구는 '경영품질성과 향상을 위한 프로세스 개선 및 혁신전략 연구 과제(16OL1220)' 와 '언어 장벽 없는 국가 구현을 위한 자동 통번역 산업 경쟁력 강화 사업(17ZS1210)'수행 결과임/본 연구 결과는 ETRI 공식 견해와 다를 수 있음.



본 저작물은 공공누리 제4유형
출처표시+상업적이용금지+변경금지 조건에 따라 이용할 수 있습니다.

- I. 서론
- II. 프로세스 이행 및 개발
기술 검증
- III. 프로젝트 기반 R&D
품질 관리 모델
- IV. R&D 품질 관리 모델
영향
- V. 결론

I. 서론

기초, 원천, 표준화, 정책, GCS(Global Challenge Software), 그리고 SW Bank 등록 대상 과제 등을 포함하여 ICT 분야에서 연간 500~600개의 다양한 R&D 과제를 수행하는 ETRI에서 SW를 개발하는 R&D 프로젝트 수행 비중은 매우 크다. 임베디드, 빅데이터, 콘텐츠, 클라우드, 모바일, 음성 인식, 융합, 보안 등 다양한 SW 개발 R&D 과제를 수행하는 만큼 프로젝트의 특성과 규모가 매우 넓고 다양하며, 프로젝트 TRL 1~9 수준, 프로젝트의 개발 언어, 개발 환경 또한 분야마다 상이하다. 그러나 SW 개발 R&D 프로젝트에서 개발 기술이나 시스템에서 정해진 '요구사항의 달성 수준'으로 평가될 수 있는 품질의 중요성은 모든 과제의 공통점이다.

프로젝트마다 목표가 다른 품질 수준은 SW 개발 주기에선 예산과 기간에 영향을 주는 핵심 요소이다. 또한, 품질은 요구사항 정의, 설계, 구현, 시험 등 SW 개발 주기의 각 단계에서 발생하는 결함 발생 수준에 영향을 받는다. 개발 과정의 각 단계에서 발생하는 결함을 감소시키기 위해서는 각 단계마다 리뷰 및 테스트와 같은 결함 식별 활동과 함께 각 단계에서 기존에 발생된 결함에 대하여 분석하고 사전에 차단하는 것이 필요하다. 과거 TDX(Time Division Exchange)나 CDMA(Code Division Multiple Access)를 개발할 때는 프로젝트 성공과 R&D 품질 관리를 위해 연구 사업에 국한된 시스템 개발 체계와 방법론을 개발하여 적용했으나, 2008년부터는 R&D 프로젝트 관리 및 수행을 위한 표준 프로세스를 수립하여 수행 과제 특성에 맞게 조정하여 적용하고 있다. 이와 동시에 SW 개발 R&D 프로젝트에서 프로세스와 개발 기술 및 코드 검증을 동시에 확인할 수 있는 프로젝트 기반 품질관리 인증모델을 개발하여 운영해 오고 있다. 이에 본 연구에서는 SW 개발 R&D 프로젝트에서 프로세스 이행 수준과 개발 기술을 동시에 검증하는 프로젝트 기반 품질 관리 모델을 소개한다. 먼

저 프로젝트 기반 품질관리 모델 배경을 기술하고, II 장에서는 프로젝트 기반 품질 관리 모델이 포함하고 있는 핵심 프로세스 구성 내용과 검토 사항 및 평가 방법, 그리고 프로젝트 수행에서 개발하는 기술과 코드 검증 절차를 기술한다. III장에서 프로젝트 기반 품질 관리 모델 특징과 절차 및 시스템 구축 환경을 세부적으로 기술한다. IV장에서는 SW 개발 R&D 프로젝트에서 프로젝트 기반 품질 관리 모델의 운영에 따른 프로젝트 성공과 개발 기술 및 결과물 품질 향상 영향에 대한 설문 조사 결과를 프로젝트 관리 측면, 개발 기술과 코드 등 결과물에서의 결함 제거 향상 측면, 기술이전과 사업화 측면에서 기술하고, V장에서 본 연구를 요약한다.

II. 프로세스 이행 및 개발 기술 검증

1. 프로세스 이행 검증

프로젝트와 기술 개발 성공을 위해서는 이에 맞는 프로세스 수립과 이행, 적절한 인력 구성과 필요한 개발 환경 구축 및 툴 활용이 필수적이다. 그중 프로젝트 수행에 적합한 프로세스 수립과 이행은 프로젝트와 기술 개발 성공의 중요한 영향을 준다. 이에 프로젝트와 기술 개발 성공을 위한 프로젝트 기반 품질 관리 모델에 포함된 프로세스 검증 내용을 기술한다. 모델에는 프로젝트 수행을 위해 시스템 개발 시작 단계인 요구사항 정의 프로세스와 시스템 개발 완료 단계인 시스템시험 프로세스, 그리고 과제관리 프로세스와 동료검토 프로세스 등 5개 핵심 프로세스를 포함하고 있으며 그 외 프로세스는 프로젝트 규모와 특성에 맞게 선택 및 조정하여 프로세스 이행을 할 수 있도록 한다.

가. 요구사항 정의

요구사항 정의에서는 사업수행계획서와 개발 기술의 사용 및 운영 환경 그리고 사용 시나리오와 기타 고객에 대한 의견 수렴을 통해 개발 기술 범위와 내용을 구체적

으로 정의하고 분석하여 시스템요구사항을 정의하는 프로세스[1]-[4]이다. 또한, 요구사항정의 프로세스에서는 도출 및 정의한 요구사항을 검증할 수 있는 문서적 테스트 케이스와 요구사항추적표 등을 준비하며, 이에 대한 동료검토 수행 여부를 확인한다.

1) Check Point

요구사항정의 프로세스에서 Check Point는 요구사항이 일관성있고 구체적으로 명시되었는지? 기능과 비기능, 인터페이스, 제약사항 등이 세분화되어 명시되었는지? 요구사항 간 상호 상충하는 부분이 없는지? 요구사항 검증을 위한 문서적 테스트 케이스는 준비되었는지? 기타 요구사항의 용어 통일과 표준 양식 활용 여부 등에 대한 확인을 위해 <표 1>과 같은 요구사항 정의 프로세스 이행에 대한 점검 항목이 준비된다.

<표 1> 요구사항 정의 프로세스 점검 사항

점검 항목	점검 내용	점수
단순성/명확성	- 요구사항이 구체적으로 명시되었는지?	
세분화	- 기능, 비기능, 인터페이스, 제약사항 등이 세분화되어 명시되었는지?	
상충성	- 요구사항 간 상충 충돌이 없는지?	
일관성	- 용어 및 형식의 일관성 여부?	
문서적 테스트 케이스 준비	- 요구사항 검증을 위한 문서적 테스트 케이스는 준비되었는지?	
동료검토	- 요구사항정의서와 테스트케이스에 대한 동료검토 수행 여부?	

2) Scoring Point

또 요구사항정의 프로세스의 이행 수준을 평가하기 위해 요구사항 정의와 프로세스 수행이 일관성과 추적성이 있고 객관적으로 진행되었다는 내용이 확인되면, 각각의 세부 점검 항목을 근거로 수식 ①, ②, ③, ④와 같이 점수화 및 등급화하여 수행과제에 대한 요구사항 정의 프로세스의 이행 수준을 평가한다.

$$\textcircled{1} \text{ PS[Process Scoring]} = \text{PCP}_1[\text{Process Check Point}] + \text{PCP}_2 + \text{PCP}_k = \sum_{i=1}^k \text{PCP}_i$$

$$\textcircled{2} \text{ PG[Process Grade]} = \text{PS}/K = \sum_{i=1}^k \text{PCP}_i / K$$

$$\textcircled{3} \text{ PPG[Process Performance Grade]} = \text{PG}_1 [\text{Process Grade}] + \text{PG}_2 + \text{PG}_k = \sum_{i=1}^k \text{PG}_i$$

$$\textcircled{4} \text{ PPPG[Project Process Performance Grade]} = \text{PPG}/K = \sum_{i=1}^k \text{PG}_i / K$$

여기서, $90 \leq \text{PCP}$, PG , $\text{PPPG} \leq 100$ 이면 S, $80 \leq \text{PCP}$, PG , $\text{PPPG} \leq 89$ 이면 A, $70 \leq \text{PCP}$, PG , $\text{PPPG} \leq 79$ 이면 B, $50 \leq \text{PCP}$, PG , $\text{PPPG} \leq 69$ 이면 C, 그리고 PCP , PG , $\text{PPPG} \leq 50$ 이면 D이다.

나. 시험

시험은 설계 및 구현, 그리고 통합 시험 이후 시스템 기능 및 비기능 요구사항에 대한 검증을 진행하는 프로세스이다. 요구사항정의서, 요구사항추적표, 시험절차서 및 시험결과서, 그리고 동료검토서를 확인하고 각각 요구사항별 테스트케이스에 따른 시험 절차와 시험 결과를 확인한다.

1) Check Point

시험 프로세스의 Check Point는 요구사항정의서, 요구사항추적표, 시험절차서 및 시험결과서, 그리고 동료검토서를 확인하고 각각 요구사항별 테스트케이스에 따

<표 2> 시험프로세스 점검 사항

점검 항목	점검 내용	점수
테스트케이스	- 요구사항별 테스트 케이스가 구체적으로 명시되었는지?	
시험환경/툴	- 시스템 기능과 비기능 검증을 위한 시험 환경과 툴이 있는지?	
시험절차	- 요구사항별 시험절차가 구체적이고 명확한지?	
요구사항과 테스트 케이스 추적성	- 요구사항과 테스트 케이스간 추적이 가능한지?	
요구사항 검증	- 요구사항별 성공/실패에 대한 시험이 진행되었는지?	
동료검토	- 시험절차서/결과서/요구사항추적표/테스트케이스에 대한 동료검토 수행 여부?	

른 시험 절차와 시험 결과를 확인한 후 <표 2>와 같은 사항을 점검한다.

2) Scoring Point

시험 프로세스의 세부 점검 항목을 근거로 점수화 및 등급화하고 수행과제에 대한 시험 프로세스 이행 수준을 평가한다(프로세스 이행 수준 평가 방식은 요구사항 정의 프로세스와 동일).

다. 동료검토

동료검토는 요구사항정의서, 시험절차서 등 시스템개발 문서와 프로젝트관리계획서, 위험관리계획서 등 사업관리 문서 등에 대한 동료검토 범위와 방법을 정의하고 각 개발 단계에서 동료검토를 수행하여 동료검토 결과를 실질적으로 반영하고 있는지를 확인하는 프로세스이다.

1) Check Point

동료검토 프로세스의 Check Point는 동료검토 범위와 방법이 정의되었는지? 동료검토 결과가 실질적으로 반영되고 있는지? 등의 내용을 포함한다.

2) Scoring Point

객관적이고 실질적인 동료검토 이행 여부, 동료검토 결과의 반영 여부, 그리고 동료검토프로세스 점검항목에 근거하여 점수화 및 등급화하여 동료검토프로세스 이행 수준을 평가한다.

라. 과제관리

과제관리는 위험관리, 품질관리, 그리고 의사소통관리 등 과제관리를 위한 종합적인 과제관리 계획과 WBS, 이슈관리계획 등 과제관리에 필요한 세부적인 내용과 방법을 준비하는 프로세스이다. 또한 과제관리프로세스에서는 과제관리를 위해 준비한 각 프로세스 이

행 계획과 구체적으로 기술된 프로세스 조정 근거가 포함된다. 따라서 과제관리프로세스에서는 성공적인 과제관리를 위해 과제관리 계획이 구체적이고 정확하게 준비되었는지 내용을 확인하고 검증한다.

1) Check Point

과제관리 프로세스의 Check Point는 과제관리에 포함되어야 할 의사소통계획, 위험관리계획, 그리고 품질관리계획 등 종합적인 과제관리계획이 준비되었는지? WBS 등 과제관리를 위한 구체적인 내용과 방법이 준비되었는지? 등이 점검항목으로 포함된다.

2) Scoring Point

과제관리계획 준비에서의 조직 자산 활용 여부, 과제관리계획에 대한 동료검토 수행 여부, 그리고 과제관리계획프로세스 점검항목에 근거하여 점수화 및 등급화하여 과제관리계획프로세스 이행 수준을 평가한다.

마. 형상관리

형상관리는 과제수행 및 시스템개발 등에서 발생하는 모든 결과물에 대한 형상 식별 체계와 방법, 그리고 베이스라인 설정 기준과 형상 환경과 상태 관리 방법 등을 구체적으로 준비하고 제시하는 프로세스이다.

1) Check Point

형상관리 프로세스 Check Point는 과제 수행에서 생성하는 과제 관리 및 시스템개발 결과물에 대한 형상 식별 체계 및 방법이 준비되었는지? 과제 결과물 관리를 위한 형상 관리 방법과 환경, 그리고 베이스라인 기준과 방법이 기술되었는지 등의 내용을 포함한다.

2) Scoring Point

일관성과 추적성 있는 형상관리 식별 체계와 방법 준비 여부, 형상관리 환경과 틀 준비 여부 등 형상관리프로세스 점검항목에 근거하여 점수화 및 등급화하여 형

상관리프로세스 이행 수준을 평가한다.

바. 기타 프로세스

그 외 성공적인 과제수행과 결과물의 완성도 향상과 자산화를 위해 시스템개발 관련 프로세스와 과제관리 관련 프로세스 그리고 지원 관련 프로세스를 과제수행에 맞게 적절히 선택하여 조정 및 적용할 수 있도록 표준프로세스가 준비되어있고, 각각 프로세스는 점검항목과 점수화 및 등급화 방법을 통해 프로세스 이행 수준을 점검할 수 있도록 한다.

2.개발 기술 검증

프로젝트 기반 품질 관리 모델에서는 프로젝트에서 수행하는 프로세스 이행 수준에 대한 평가는 물론 개발 기술과 코드에 대한 검증 과정도 포함한다. 개발 기술과 코드 검증에서는 요구사항정의와 요구사항 별 문서화된 테스트 케이스 준비 단계, 코딩과 디버깅 및 테스트 단계, 소스 코드 적정 분석 및 오픈 소스 라이선스 검증 단계, 그리고 품질 요원에 의한 개발 기술과 코드의 2차적 검증 단계를 포함하여 기본적으로 5단계 과정으로 이루어진다.

가. 테스트 케이스 기반 요구사항 정의 및 검증

개발자는 사용자 요구사항을 도출 및 정의하고, 사용자 요구사항을 개발할 시스템 기능으로 표현하는 시스템요구사항[1], [5], [6]을 기술한다. 준비한 요구사항에 대해 동료 검토를 통해 보완하고 정의된 요구사항을 검증할 테스트 케이스와 검증 방법(Test/Inspection/Demonstration/Analysis/Simulation/Modeling)을 정의한다. 아래는 활동과 결과물 예이다.

〈표 3〉 테스트 케이스 기반 요구사항 검증 보고서

순번	시스템 요구사항 ID	테스트 케이스 ID	검증 방법	동료 검토	QA 검토

- 요구사항검증보고서 〈표 3〉.
- 사용자/시스템/테스트 케이스 요구사항추적표.

나. 코딩 & 디버깅 & 테스트

개발자는 개발할 요구사항 및 기능에 대해 동작 시나리오와 데이터 흐름을 정의 한 후 각각의 클래스와 메소드 정의를 통해 코딩을 수행한다. 또한, 코딩 과정에서는 기능 동작에 따른 취약점과 테스트 코드를 고려하여 코딩과 디버깅을 반복적으로 수행하고, 요구사항정의 단계에서 준비한 테스트 케이스를 이용해 기능 검증과 동시에 결함을 제거한다. 이후 기능 구현 과정에서 이용했던 테스트 케이스를 보완하여 통합 및 시스템 시험에 활용하도록 한다. 아래는 활동과 결과물 예이다.

- 각 요구사항과 기능에 대한 클래스 및 메소드 정의.
- 각 클래스 및 메소드 별 코딩 및 디버깅.
- 각 클래스 및 메소드에 대한 실패, 성공 테스트 데이터 준비 및 시험.
- 가독성과 의존성을 고려한 코드 리팩토링.
- 반복적 작업을 통해 코드 품질 및 완성도 향상.
- 업데이트 테스트 케이스/결함보고서.
- 사용자&시스템요구사항정의서/코드/요구사항별 테스트 케이스/요구사항 추적표.

다. 정적분석

개발자는 개발한 소스코드의 정적분석[2], [5], [7], [8]을 통해 결함과 취약성 제거를 하면서 코드보완작업을 병행한다. 이렇게 보완 개발된 최종 소스 코드에 대해 정적분석을 추가로 진행하여 메모리 릭(Leak)이나 실행 오류(Run time error)와 같은 치명적인 코드 결함이나 보안 취약점 등과 같은 코드 결함이 추가적으로 발생하지 않으면 변경한 코드에 대해 회귀 시험을 진행하고 테스트 케이스를 업데이트 한다. 그리고 최종적인 소스코드에 대한1차적(사전) 정적분석결과서를 준비한다.

〈표 4〉 정적분석결과서(Code Compliance Report)

프로젝트 이름		개발자 이름		날짜		QA 검토
프로그램 언어		라인수		총 결함수		
코딩표준 규칙		보안취약 규칙		실행오류 규칙		
결함	중요 마이너	결함	중요 마이너	결함	중요 마이너	

아래는 활동과 결과물 예이다.

- 소스 코드 정적분석 실행 및 치명적 결함 제거.
- 코드 수정(클래스, 메소드, 변수 등)에 따른 회귀 시험.
- 업데이트 테스트 케이스.
- 반복적 작업을 통해 코드 품질 및 완성도 향상.
- 정적분석결과서 〈표 4〉.

라. OSS 라이선스 검증

개발자는 개발 코드에 대해 정적분석을 완료한 후 OSS(Open source software)[1], [4], [6] 라이선스 검증을 진행한다. 이는 개발한 소스 코드에서 OSS 사용 여부와 OSS 사용에 따른 제약 사항을 사전에 파악하고 대처하기 위함이다. 또한, 개발자는 개발 소스 코드에 대한 OSS 라이선스 검증에 앞서 개발 소스에 대해 파일별, 혹은 기능별, 모듈별, 클래스별, 혹은 디렉토리별 구분

〈표 5〉 OSS 라이선스 검증 보고서(OSA)

프로젝트 이름		개발자 이름					QA 검토
검증 대상 파일 수		분석 파일 수					
분석 바이트수		공개 SW 사용 파일 수					
준법성 주의 파일 수		준법성 위반 파일 수					
Bill of materials	준법성	컴포넌트	버전	라이선스	결함	해당파일수	
공개 SW 라이선스 종합요약							
검증 및 분석 의견							

하여 OSS 사용 여부와 사용 버전 그리고 사용 내용 등을 내용 등을 사전에 구분 처리하여 OSS 라이선스 검증을 용이하게 한다. OSS 라이선스 검증을 진행한 후 OSS 사용에 따른 제약 사항이 없는 경우 개발자는 개발 소스에 대한 OSS 라이선스 검증보고서를 준비한다. 아래는 활동과 결과물 예이다.

- 개발 소스에 대한 OSS 사용 구분 명시.
- OSS 사용에 따른 제약 사항 제거.
- OSS 라이선스 검증 보고서 〈표 5〉.

마. Quality Assurance 검증

프로젝트 수행 및 기술 개발 과정에서 개발자는 개발 기술과 소스 코드에 대해 1차적 검증 과정을 거친 후 개발 기술에 해당하는 문서와 소스 코드를 QA(Quality assurance)로 보낸다. QA는 개발자가 보낸 소스 코드와 요구사항 정의서, 시험 절차서 및 시험 결과서 등 개발 기술 관련 문서를 검토하고, 요구사항 별 테스트 케이스와 시험 절차서 그리고 시험 결과서를 바탕으로 개발 기술을 검증한다. 또 QA는 개발 기술에 대한 문서적 검증과 시연을 통해 개발 기술 검증을 마친 후 개발 소스 코드에 대해 정적분석과 OSS 라이선스 검증을 추가적으로 진행한다. QA에 의한 요구사항검증과 정적분석, 그리고 OSS 라이선스 검증을 완료하면 QA는 개발 기술에 대한 검증보고서를 준비하여 조직의 최고 책임자 승인을 득하여 개발 기술 검증과 코드에 대해 인증을 부여한다. 아래는 활동과 결과물 예이다.

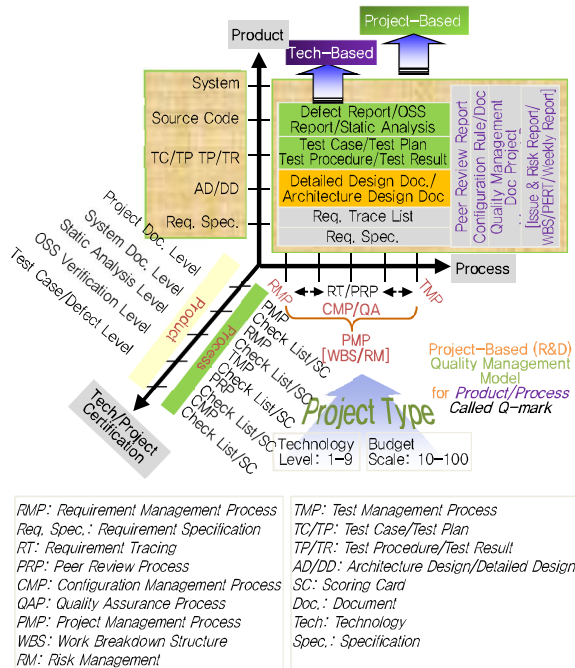
- 요구사항 정의서, 시험 절차서, 시험 결과서.
- 테스트 케이스, 동료 검토서, 결함 보고서.
- 요구사항 추적표 검증.
- 정적분석 보고서(by QA).
- OSS 라이선스 검증 보고서(by QA).
- 개발 기술 시연 & 확인.
- 요구사항 검증보고서(by QA).

III. 프로젝트 기반 R&D 품질 관리 모델

프로젝트 기반 품질관리 모델은 SW 개발 R&D 프로젝트에서 개발 기술 검증을 위해 요구사항정의 단계에서 개발자에 의한 문서적 테스트 케이스 준비와 검토, 설계 및 구현 단계에서의 준비된 테스트 케이스 기반 시험과 디버깅, 그리고 코드 검증과 릴리즈를 위한 정적분석과 OSS 라이선스 검증 과정 외에 프로젝트 및 개발 기술 완료 시점에 개발 기술과 코드에 대해 추가적인 검토 및 검증 과정이 품질담당자에 의해 진행된다. 또 프로젝트 기반 품질관리 모델에서는 프로젝트에서 수행한 프로세스 이행 수준 평가 과정을 포함한다. 다음은 SW 개발 R&D 프로젝트에서 프로젝트 성공률 향상과 개발 기술 및 코드 등 결과물에 대한 품질 및 완성도 향상에 기여할 수 있는 프로젝트 기반 품질관리 모델에 대한 내용과 절차 및 시스템 환경을 기술한다.

1. 프로젝트 기반 품질 관리 모델

(그림 1)은 프로젝트 기반 품질관리[1], [3], [4], [8] 모



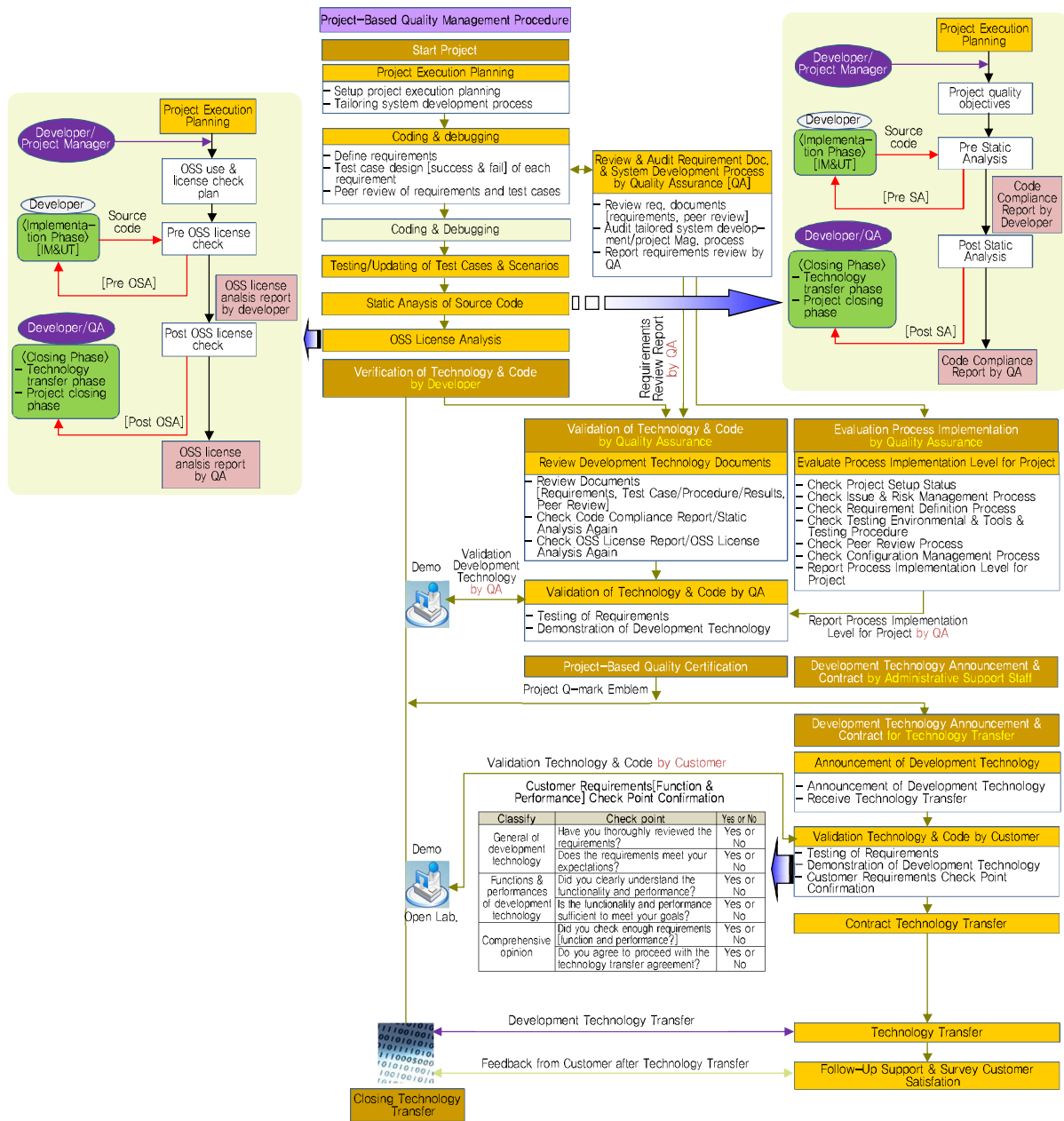
(그림 1) 프로젝트 기반 R&D 품질관리 모델

델로 X축은 과제 수행에 적용하는 프로세스 범위를 표현하고, Y축은 과제 수행에서 만들어 지는 개발 결과물의 범위와 종류를 나타낸다. 또한, X, Y 축의 평면에는 과제에서 개발하는 기술을 검증하는데 필요한 결과물 검증 내용을 나타내고, 과제 관리를 위해 수행한 문서들을 나타낸다. Z 축은 개발 기술 검증을 위한 지표와 내용, 그리고 프로세스 검증을 위한 점검 항목과 내용을 나타낸다.

프로젝트 기반 품질관리 모델에서 프로세스 이행 수준 평가는 시스템개발 시작 단계인 요구사항 정의 프로세스와 요구사항을 검증하는 시스템시험프로세스를 기본으로 하고, 과제관리를 위한 과제관리계획프로세스 그리고 동료검토프로세스 등을 포함한다. 각 프로세스에는 프로세스 점검사항이 포함되며 이를 근거해서 과제에서 각 프로세스 이행 수준을 평가하여 프로세스를 평가한다. 프로젝트 기반 품질 관리 모델은 개발 기술에 대한 검증도 동시에 수행하는데, 개발 기술 검증은 문서적 테스트 케이스 기반 요구사항정의와 동료 검토, 설계 및 구현 단계에서의 디버깅과 시험, 그리고 개발 소스에 대한 정적분석과 오픈소스 라이선스 검증을 근거로 진행한다. 소스에 대한 정적 분석 단계에서는 개발 소스의 결정적 취약점과 실행 오류, 그리고 코딩 표준 준수율 여부를 개발 기술에 대한 소스 코드를 검증한다. 이 과정은 기술 개발이 진행되는 단계에서는 개발자에 의해 1차적으로 진행되고, 기술 개발이 완료되어 개발 기술을 기업에 이전하거나 개발 기술 및 코드를 릴리즈 하는 단계에서 QA 의해 다시 한번 검증 및 확인된다.

2. 프로젝트 기반 프로세스 이행 수준과 개발 기술 및 코드 검증 절차

(그림 2)는 프로젝트 기반 품질 관리 모델에 포함된 프로세스에 대한 이행 수준 평가와 개발 기술 및 코드를 검증하는 절차이다. 이 절차는 프로젝트에서 개발한 기술을 기업에 이전하는 프로젝트 경우나 TRL 5 이상의



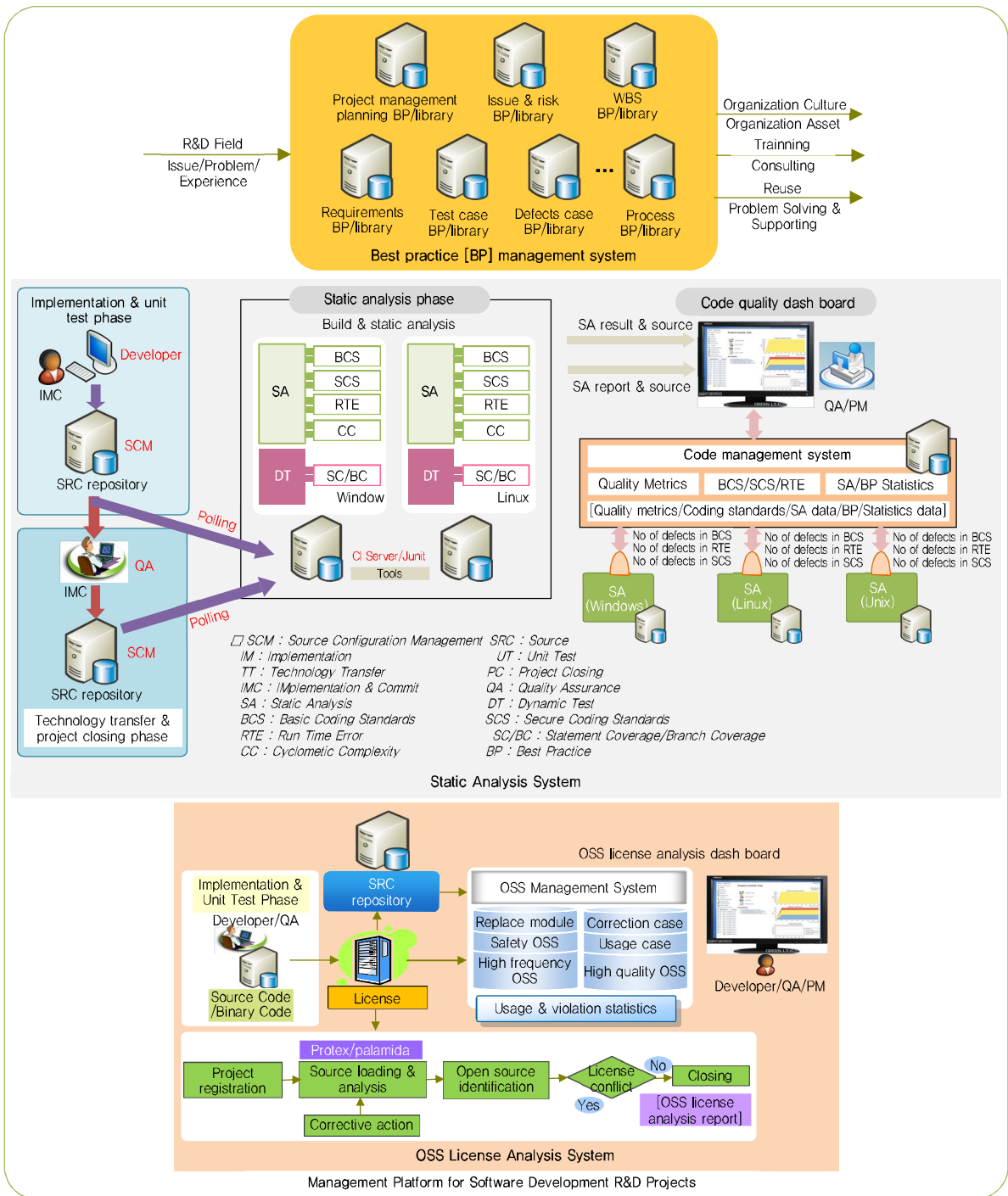
(그림 2) 프로젝트 기반 품질관리 모델의 프로세스 및 개발 기술 검증 절차

SW 개발 R&D 프로젝트에서는 필수적으로 진행되는 내용이다. 또 이 과정에는 요구사항 정의부터 구현 및 시험 단계에서 개발자에 의해 수행하는 개발 기술과 코드를 검증하는 부분, 품질담당자에 의해 개발된 기술과 코드를 검증하고 프로세스 이행 수준을 평가하는 부분, 그리고 개발 기술의 기업 이전에 앞서 오픈 랩을 통해

사용자에 의해 개발 기술을 확인 및 검증 받는 데모 수행 부분을 포함한다.

3. 프로젝트 기반 개발 기술 및 결과물 공유 환경

(그림 3)은 SW 개발 R&D 프로젝트 수행 과정에서 생성한 개발 기술과 코드, 문서, 그리고 프로젝트 관리에



(그림 3) 프로젝트 기반 개발 및 공유 플랫폼 환경

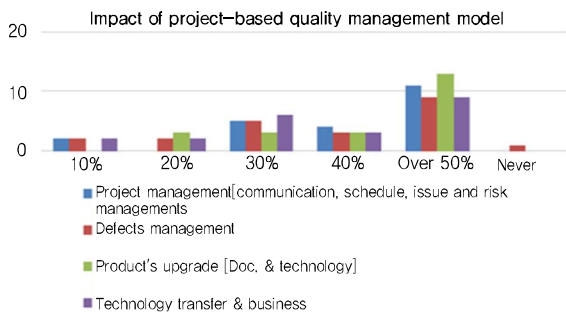
서 발생한 프로세스 이행 결과물 등 프로젝트 수행에서 만들어진 결과물 관리 및 공유 플랫폼 환경이다. 따라서 이 모델에는 개발 기술과 코드 검증 및 관리를 위해 정

적분석과 OSS 라이선스 검증 환경이 포함되며, 프로젝트 수행 과정에서 생성한 개발 기술, 코드, 문서, 그리고 프로세스 이행 과정에서 발생한 각종 Best Practice의

자산화 및 관리 환경을 포함한다.

IV. 프로젝트 기반 R&D 품질 관리 모델 영향

(그림 4)는 프로젝트 기반 품질 관리 모델이 SW 개발 R&D 프로젝트에서 의사 소통과 위험 관리 및 프로젝트 성공 향상 측면, 결함 제거 측면, 결과물 품질 향상과 기술 이전 및 사업화 측면에서 종합적으로 따른 긍정적 영향을 보여주고 있음을 보인다. 프로젝트 기반 품질 관리 모델은 SW 개발 R&D 프로젝트에서 프로젝트 관리 및 수행 측면에서 참여원간의 의사소통 및 이슈 그리고 위험관리 등 프로젝트 성공에 50% 이상 직·간접적 좋은 영향을 주었다고 50%가 응답하였고, 프로젝트 성공에 40% 및 30% 이상의 좋은 영향을 주었다고 응답한 사람도 18%와 23%였다. 그 외 잘모르겠다와 도움 되지 않았다는 응답자도 10% 정도를 차지했으나 전반적으로 프로젝트 기반 품질 관리 모델이 프로젝트 수행에 좋은 영향을 주었다고 답하고 있다. 결함 제거 측면에서도 50% 이상 직·간접적 좋은 영향을 주었다고 41%가 응답하였고, 40% 이상 좋은 영향을 주었다고 응답한 사람도 14%, 그리고 30% 이상의 좋은 영향을 주었다고 응답한 사람이 23%였다. 또 개발 기술과 코드를 포함해 연구 결과물 완성도 향상 측면에서도 프로젝트 기반 품질 관리 모델은 60% 이상 좋은 영향을 주었다고 50% 이상 응답하였고, 대부분의 응답자들은 연구 결과물 완성도 향상에 최소 30% 이상 좋은 영향을 주었다고 답하였다.



(그림 4) 프로젝트 기반 품질관리 모델 영향

프로젝트 기반 품질 관리 모델은 개발 기술의 기업 이전과 사업화에도 50% 이상 좋은 영향을 주었다고 41%가 응답하였고, 40~30% 이상 좋은 영향을 주었다고 응답한 사람도 41%였다.

V. 결론

성공적인 프로젝트 수행과 프로젝트 결과물 및 개발 기술에 대한 완성도 향상은 프로젝트 매니저는 물론이고 프로젝트 참여원들의 한결 같은 기대이다. 프로젝트 수행 일정 지연이나 개발 결과물에 대한 품질 및 완성도 저하는 프로젝트 실패의 주요 요인이며 과제를 수행하면서 쉽게 개선 혹은 해결되지 않는 문제이기도 하다. 국내외적으로 프로젝트 성공률은 여전히 50% 이상을 넘지 못하고 있다. 미국이나 유럽에서는 프로젝트 성공률 향상을 위해 PMP나 PRINCE2와 같은 프로젝트 관리 기법을 개발하여 프로젝트 수행에 적용하도록 교육 및 홍보하고 있다. 국내에서도 조직이나 기업 등에서 성공적인 프로젝트 수행을 위한 프로젝트 관리 기법을 도입 및 활용하고 있다. 프로젝트와 기술 개발 성공을 위해서는 이에 맞는 프로세스 수립과 적절한 인력 구성 그리고 필요한 개발 환경 구축 및 툴 활용이 필수적이다. 그 중에서 프로젝트 수행에 적합한 프로세스 수립과 이행은 프로젝트와 기술 개발 성공의 중요한 영향을 준다. 이에 본 연구에서는 SW 개발 R&D 프로젝트에서 개발 결과물의 품질과 완성도 향상은 물론 프로젝트 성공률을 높이는데 긍정적 영향을 주고 있는 프로젝트 기반 품질 관리 모델을 소개했다. 프로젝트 기반 품질 관리 모델은 프로젝트에서 수행한 프로세스의 이행 수준 평가와 프로젝트에서 개발한 기술과 코드 검증을 동시에 확인하며 이를 통해 프로젝트 성공률 향상은 물론 개발 기술의 완성도 향상을 가져올 수 있는 프로세스 이행 수준 평가 및 개발 기술과 결과물 검증에 기초한 모델이다. 제시한 프로젝트 기반 품질관리 모델은 요구사항정의와

시험프로세스 등 5개 핵심 프로세스에 대한 프로세스 이행 수준 점검 사항과 평가 방법을 포함한다. 그리고 개발 기술과 코드 검증을 위해 요구사항정의 단계에서 개발자에 의해 1차적으로 수행하는 요구사항정의와 요구사항 별 문서적 테스트 케이스 준비 및 검토, 그리고 설계 및 구현 단계에서 테스트 케이스 업데이트와 정적 분석 및 오픈소스 라이선스 검증 방법과 내용이 포함되고, 프로젝트 완료 단계에서 개발 기술을 기업에 이전하기에 앞서 품질담당자에 의해 추가적으로 수행하는 개발 기술과 코드에 대한 정적분석 및 오픈소스 라이선스 검증 방법 및 절차를 포함한다. 프로젝트 기반 품질 관리 모델은 SW 개발 R&D 프로젝트 수행에서 프로젝트 성공률 향상은 물론 프로세스 이행 능력 향상과 개발 기술 및 소스 코드를 포함한 결과물에서의 결함 제거 및 완성도 향상에 최소 10% 이상 좋은 영향을 주고 있다고 개발자들은 응답하였다. 앞으로 프로젝트 기반 품질 관리 모델은 R&D 프로젝트 수행에서 조직의 프로세스 이행 능력과 성숙도 수준을 향상시키고 동시에 개발 기술과 코드를 검증하는 프로젝트 인증 모델로의 적용 확대는 물론 SW 개발 R&D 프로젝트 성공률 향상에 지속적으로 기여할 수 있을 것으로 기대한다.

약어 정리

BP Best Practice

OSA	Open Source Analysis
OSS	Open Source Software
PM	Project Manager
PMP	Project Management Professional
PRINCE2	PRojects IN Controlled Environments Version 2
QA	Quality Assurance
R&D	Research & Development
SA	Static Analysis
TRL	Technology Readiness Level

참고문헌

- [1] Quality Innovation Section, "ETRI R&D Standard Process V4.0," ETRI, 2016. 8.
- [2] NIPA, "Static Analysis Using Open Source," GCS School, 2015. 5.
- [3] J.-H. Park et al., "Q-Mark Model Assessment Using CBAM (Cost Benefit Analysis Method)," in *Int. Conf. Manag. Eng. Technol.*, Portland, OR, USA, Aug. 2-6, 2015, pp. 1805-1808.
- [4] Quality Innovation Section, "Q-Mark 6.0," ETRI, 2017. 4.
- [5] S. McConnell, *Code Complete*, Albuquerque, NM, USA: Microsoft Press, 1993. 5.
- [6] NIPA, "Understanding and Management of Open Source," GCS School, 2016. 5.
- [7] J.-H. Park et al., "Static Analysis for Code Quality on R&D Projects of SW Development," *ETRI Mag.*, 2017. 2.
- [8] J.-H. Park et al., "Q-Mark Model for Quality of R&D Projects," *FGCN2014, SERSC Proc.*, Suwon, Rep. of Korea, Nov. 21-22, 2014, pp. 1743-1756.