

10주 2강

프로세스 품질 특성 평가 모델



이번 주차에는...

소프트웨어 품질

- 프로세스 품질 특성 평가 모델
- 대표적인 프로세스 능력 평가 모델
- 품질 관리

35. 프로세스 품질

- 제품 품질의 비유

- 좋은 수돗물을 마시려면 무엇보다 수질 자체가 좋아야 한다.

- 프로세스 품질의 비유

- 가정까지 오는 수도관이 오래되어 녹슬거나 파손되었으면 깨끗한 물을 마실 수 없다.

- 프로세스 품질

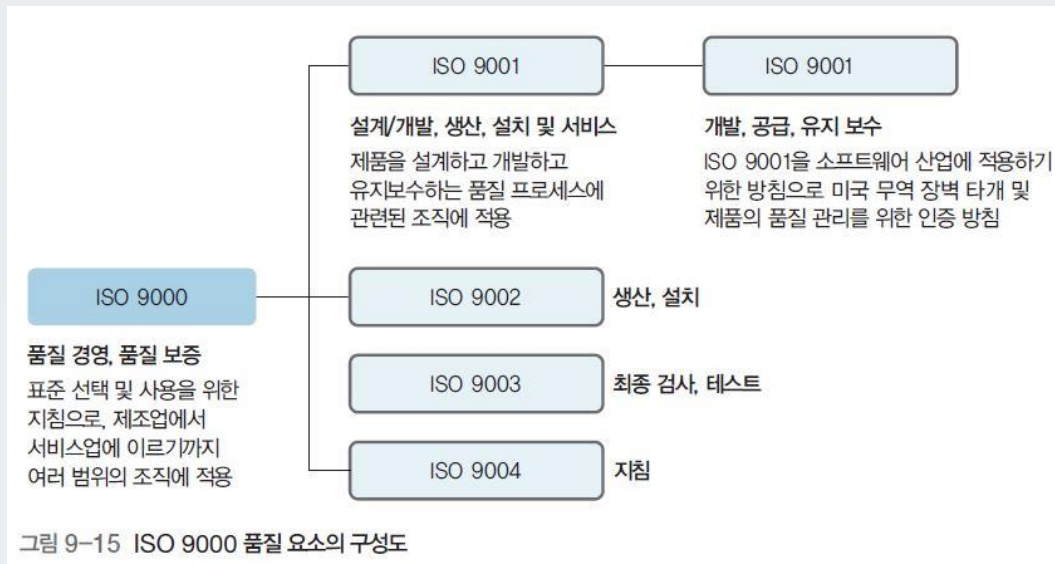
- 소프트웨어 제품의 최종 품질에 영향을 줄 수 있는 소프트웨어 개발과정에 대한 품질

- 품질 시스템 보증을 위한 ISO 9000 모델
- 소프트웨어 생명주기 프로세스 표준을 위한 ISO 12207 모델
- 소프트웨어 프로세스 능력 평가를 위한 CMMI와 SPICE(ISO 15504) 모델

36. ISO/IEC 9000 모델의 품질 요소(1)

■ ISO 9000

- 국제 표준화 기구ISO가 정한 품질 관리와 품질 보증을 위한 모델
- 해당 제품이나 서비스 설계에서부터 생산 시설, 시험 검사 등 전반에 걸쳐 규격준수 여부를 확인해 인증
- 목적: 제품의 품질을 객관적으로 인증 받아 사용자에게 신뢰감을 주는 것



37. ISO/IEC 9000 모델의 품질 요소(2)

■ ISO 9000 모델의 특징

- SDLC의 과정에 대한 품질 보증 모델이다.
- 소프트웨어 개발을 목표로 구체화되지 않았다.
- 소프트웨어에 적용할 수 있는 일반 원리를 설정한다.
- 품질 프로세스의 다양한 측면을 기술한다.
- 기업이 정의해야 하는 조직의 표준과 절차를 나열한다.
- 조직의 품질 매뉴얼로 문서화한다.
- 사용되어야 하는 품질 프로세스를 정의하지 않는다.
- 공급자와 구매자 간의 관리 책임을 명시한다.

38. 프로세스 표준을 위한 ISO 12207 모델(1)

■ ISO 12207

- 소프트웨어 개발 생명주기 프로세스인 소프트웨어 생성부터 폐기까지의 프로세스에 해당
- 구성: 기본 생명주기, 지원 생명주기, 조직 생명주기
- 기본 생명주기 프로세스: 소프트웨어의 획득, 공급, 개발, 운영, 유지보수 프로세스를 체계적으로 관리하기 위한 life cycle process standard를 제공
- 실무자들에게 공통 사항에 대한 프레임워크를 제공

39. 프로세스 표준을 위한 ISO 12207 모델(2)

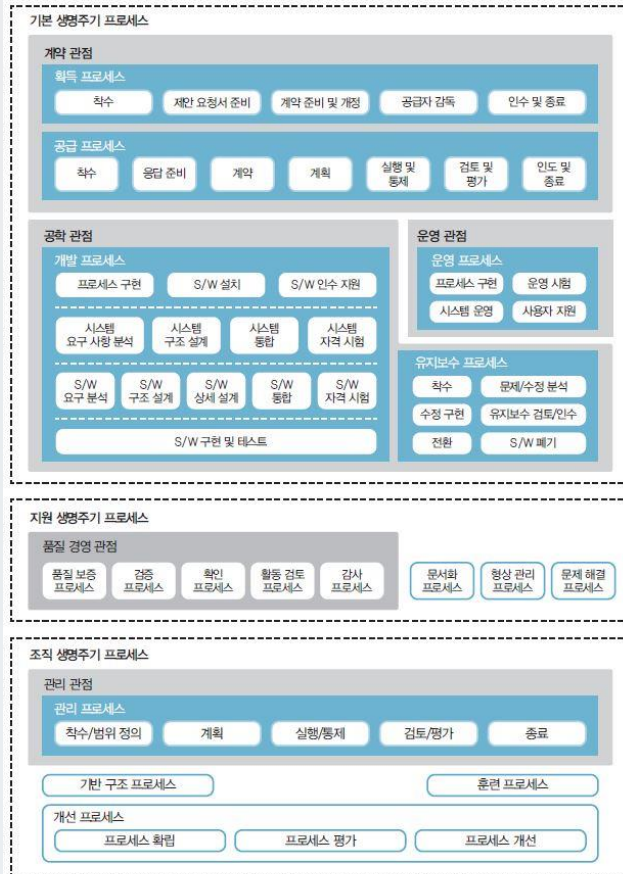


그림 9-16 ISO 12207 모델의 품질 요소 구성도

40. 표준 프로세스의 필요성(1)



■ 표준 프로세스

- 소프트웨어 개발에서 레시피, 하나의 매뉴얼, 내비게이션과 같은 역할
- 직원들이 우왕좌왕 해매는 시간을 줄여주고 생산성을 높인다.
- 기준과 목표, 방향을 제시해주기 때문에 업무 처리 프로세스가 명확하고 계획적이며 결과를 충분히 예측할 수 있다.

41. 표준 프로세스의 필요성(2)

■ CMMI Capability Maturity Model Integration

- 조직의 프로세스 개선을 위해 개발
- 기업에 표준 프로세스를 만들 수 있는 지침을 제시하고, 그 기준이 된다.

■ 어떤 조직이 표준 프로세스를 사용하고 지속적으로 개선한다면

- 조직의 업무 수행 능력 및 품질 향상
- 현재 조직의 표준 프로세스 수준을 잘 파악 → 향후 개선 방향 판단 가능
- 프로젝트의 정량적 목표 및 계획 수립 가능
- 최종 목표 달성 예측 가능

→ 소프트웨어 공학의 목표인 개발의 생산성 향상과 품질 향상을 꾀할 수 있다.

42. CMMI 모델(1)

■ C: 능력 Capability

- 능력: 개발 목표(주어진 기간, 정해진 비용, 고품질 등)를 달성할 수 있는 힘

■ M: 성숙도 Maturity

- 성숙(사전적 의미): ‘생물의 발육이 완전히 이루어짐’, ‘몸과 마음이 자라서 어른스럽게 됨’, ‘경험이나 습관을 쌓아 익숙해짐’ 등으로, 다 자라서(완성되어) 책임감이 있는 느낌을 준다.
- 성숙도가 높은 조직: 책임감이 있는 조직으로서 개발 과정에서 객관적이고 정량적인 근거에 따라 프로세스가 측정되고 지속적인 개선이 이루어지는 조직

■ M: 모델 Model

- 프로세스를 감사(audit)하는 의미로 사용, 기준으로 하고 있는지, 그렇지 않은지를 검사
- CMMI는 그 기준을 제시하고 있는데 그것이 ‘수행 지침(best practice)’이라는 모델

43. CMMI 모델(2)

■ I: 통합^{Integration}

- 여러 가지 프로세스의 기준을 하나로 통합했다는 의미
- 소프트웨어 개발 생명주기의 각 단계를 통합한 모델이라는 의미

■ CMMI

- 조직의 프로세스에 대한 가이드이자 기준
- '능력'과 '성숙도'로 조직의 프로세스를 측정하고 평가하는 모델의 통합 버전인 프로세스 개선 성숙도 모델

44. CMMI 모델(3)

■ 성숙도가 높은 조직과 낮은 조직의 특징

표 9-13 성숙도가 높은 조직과 낮은 조직의 특징

구분	특징
성숙도 높음	<ul style="list-style-type: none">• 소프트웨어 개발/관리 프로세스가 조직 차원에서 이루어진다.• 구성원들이 소프트웨어 프로세스를 잘 알고 있다.• 프로세스를 따라 수행함으로써 역할과 책임이 명확하다.• 제품 품질을 중요하게 여기고, 사용자의 만족도를 측정한다.• 조직 차원의 표준 프로세스가 일관성 있게 준수되고 있다.
성숙도 낮음	<ul style="list-style-type: none">• 조직 내에 정해진 프로세스가 없어 필요할 때마다 임시방편으로 만들어 사용하고 있다.• 문제가 발생했을 때 근본적인 해결 방안을 찾기보다는 임시방편으로 해결하려고 한다.• 객관적인 비용 산정과 근거에 의한 일정이 산출되지 않아 개발 기간과 비용이 초과하는 경우가 많다.• 제품 품질에 대해 객관적으로 평가하지 못한다.• 제품의 기능, 품질보다 납기일을 최우선으로 생각한다.

45. CMMI의 구성(1)

■ CMMI 5단계

표 9-14 CMMI 5단계(소프트웨어 프로세스 성숙도)

단계	프로세스	내용
❶ 초기 ^{initial} 단계	프로세스 없음	예측/통제 불가능
❷ 관리 ^{managed} 단계	규칙화된 프로세스	기본적인 프로젝트 관리 체계 수립
❸ 정의 ^{defined} 단계	표준화된 프로세스	조직 차원의 표준 프로세스를 통한 프로젝트 지원
❹ 정량적 관리 ^{quantitative managed} 단계	예측 가능한 프로세스	정량적으로 프로세스가 측정/통제됨
❺ 최적화 ^{optimizing} 단계	지속적 개선 프로세스	프로세스 개선 활동

46. CMMI의 구성(2)

표 9-15 CMMI의 4가지 범주로 구분된 22개의 프로세스 영역

범주	프로세스 영역
프로젝트 관리	<p>프로젝트 계획, 감시, 제어와 관련된 프로젝트 관리 행위들을 다루는 프로세스 영역들로 구성됨.</p> <p>① 프로젝트 계획PP: project planning-L2</p> <p>② 프로젝트 감시 및 통제PMC: Project Monitoring and Control-L2</p> <p>③ 협력 업체 관리SAM: Supplier Agreement Management-L2</p> <p>④ 통합된 프로젝트 관리IPM: Integrated Project Management + IPPD-L3</p> <p>* IPPD: Integrated Product and Process Development</p> <p>⑤ 위험 관리RSKM: Risk Management-L3</p> <p>⑥ 정량적 프로젝트 관리QPM: Quantitative Project Management-L4</p>
공학	<p>여러 공학 분야에 걸쳐서 공유되는 개발과 유지보수와 관련된 활동들을 다루는 프로세스 영역들로 구성됨.</p> <p>⑦ 요구 사항 관리REQM: Requirements Management-L2</p> <p>⑧ 요구 사항 개발RD: Requirements Development-L3</p> <p>⑨ 기술적 솔루션TS: Technical Solution-L3</p> <p>⑩ 제품 통합PI: Product Integration-L3</p> <p>⑪ 확인VER: Verification-L3</p> <p>⑫ 검증VAL: Validation-L3</p>
프로세스 관리	<p>프로세스의 정의, 계획, 배치, 구현, 감시, 제어, 평가, 측정, 개선과 관련된 여러 프로젝트에 걸쳐진 활동들을 포함하는 프로세스 영역들로 구성됨.</p> <p>⑬ 조직 차원의 프로세스 개선OPF: Organizational Process Focus-L3</p> <p>⑭ 조직 차원의 프로세스 정의OPD: Organizational Process Definition + IPPD-L3</p> <p>⑮ 조직 차원의 교육 훈련OT: Organizational Training-L3</p> <p>⑯ 조직 차원의 프로세스 성과 관리OPP: Organizational Process Performance-L4</p> <p>⑰ 조직 차원의 혁신 활동 전개OID: Organizational Innovation and Deployment-L5</p>
지원	<p>제품 개발과 유지보수를 지원하는 활동들을 다루는 내용으로, 프로젝트를 목적으로 한 프로세스 영역과 조직에 적용하는 것을 목적으로 하는 프로세스 영역들로 구성됨.</p> <p>⑱ 형상 관리CM: Configuration Management-L2</p> <p>⑲ 프로세스/제품 품질 보증PPQA: Process and Product Quality Assurance-L2</p> <p>⑳ 측정 및 분석MA: Measurement and Analysis-L2</p> <p>㉑ 의사결정 분석 및 해결DAR: Decision Analysis and Resolution-L3</p> <p>㉒ 근본 원인 분석 및 해결CAR: Causal Analysis and Resolution-L5</p>

47. CMMI의 구성(3)

■ CMMI 프로세스 영역의 구조도

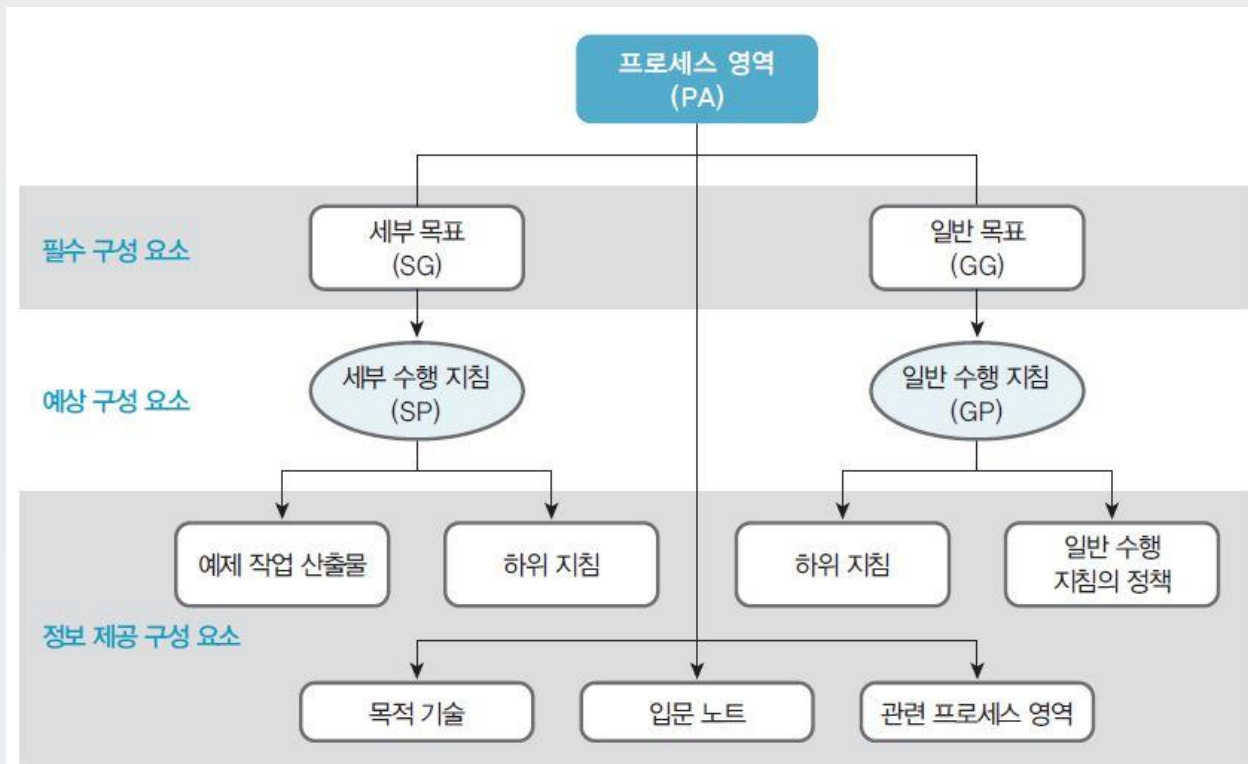


그림 9-18 CMMI 프로세스 영역(PA)의 구조도

48. CMMI의 구성(4)

■ 일반 목표와 수행 지침

- 목표: 모든 프로세스 영역에 공통으로 적용되는 목표
- 일반 목표 달성: 해당 프로세스의 활동들이 조직에 내재화되어 자연스럽게 수행될 수 있음
- 수행 지침: 일반적인 목표를 만족시키기 위해 수행해야 하는 활동이 무엇인지를 설명

■ 세부 목표와 수행 지침

- 목표: 특정 프로세스 영역에만 적용되는 좁은 의미의 구체적인 목표
- 수행 지침: 세부적인 목표를 만족시키기 위해 수행해야 하는 활동이 무엇인지를 설명

49. CMMI의 구성(5)

■ 필수 구성 요소

- 조직이 프로세스 영역을 만족시키기 위해 무엇을 성취해야 하는지를 기술하는 구성 요소
- 세부 목표와 일반 목표가 이에 해당

■ 예상 구성 요소

- 조직이 필수 구성 요소를 성취하기 위해 전형적으로 무엇을 구현해야 하는지를 기술
- 이 구성 요소들은 누가 평가를 수행하고 개선을 구현하는지를 가이드
- 일반 수행 지침과 세부 수행 지침이 이에 해당

■ 정보 제공 구성 요소

- 필수 구성 요소와 예상 구성 요소에 접근할 수 있도록 돕는 세부 내용을 제공
- 예제 작업 산출물, 하위 지침, 일반 수행 지침의 정책, 입문 노트, 관련 프로세스 영역 등이 이에 해당

50. CMMI의 평가 방법

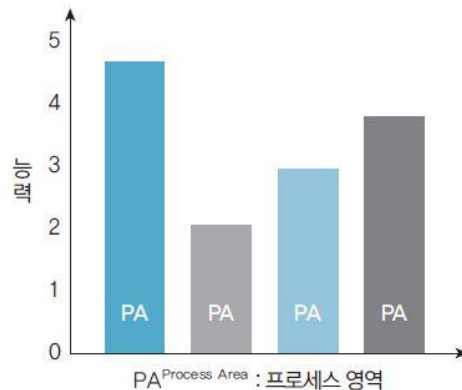
평가 방법

단계적 표현^{staged representation} 방법의 성숙 단계^{maturity level}

연속적 표현^{continuous representation} 방법의 능력 단계^{capability level}



(a) 단계적 표현 방법의 성숙 단계



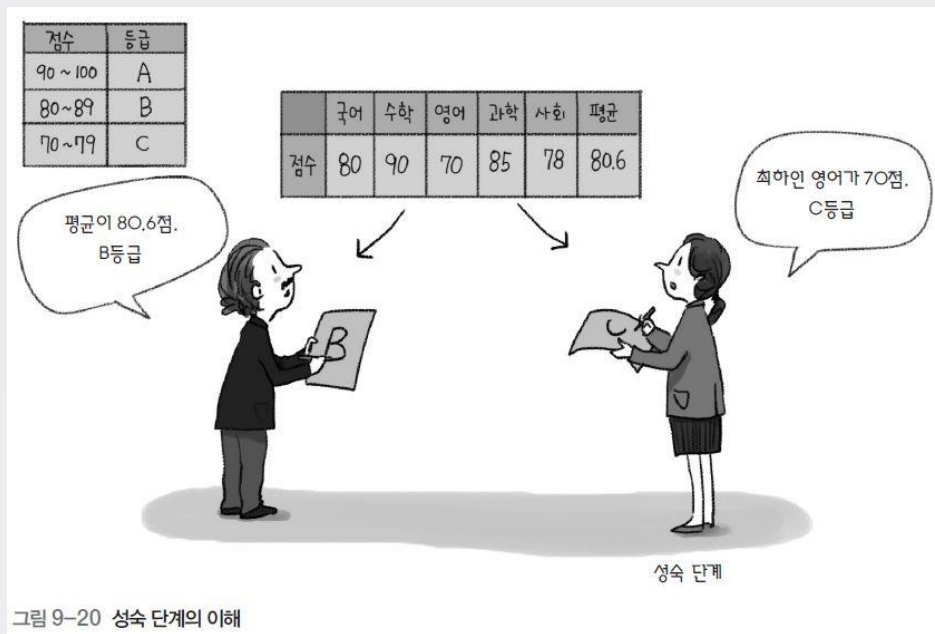
(b) 연속적 표현 방법의 능력 단계

그림 9-19 CMMI의 평가 방법 : 성숙 단계와 능력 단계

51. 단계적 표현 방법의 성숙 단계(1)

■ 성숙 단계

- 조직에서 해당 업무를 얼마나 체계적으로 수행하고 있는지를 나타내며, 지표로는 1에서 5까지 5단계로 구분하여 사용



52. 단계적 표현 방법의 성숙 단계(2)

■ 성숙도 단계별 프로세스 영역

표 9-16 성숙도 단계별 프로세스 영역

단계	범주	프로세스 영역
❶ 초기 단계	프로세스 없음	
❷ 관리 단계	프로젝트별로 프로세스 존재	<ul style="list-style-type: none"> • 요구 사항 관리 • 프로젝트 계획 수립 • 프로젝트 감시 및 통제 • 협력 업체 관리 • 측정 및 분석 • 프로세스/제품 품질 보증 • 형상 관리
❸ 정의 단계	조직 차원의 프로세스 존재(프로세스 표준화)	<ul style="list-style-type: none"> • 요구 사항 개발 • 기술적 솔루션 • 제품 통합 • 검증, 확인 • 조직 차원의 프로세스 개선 • 조직 차원의 프로세스 정의 • 조직 차원의 교육 훈련 • 통합된 프로젝트 관리 • 위험 관리 • 의사결정 분석 및 해결
❹ 정량적 관리 단계	측정 가능한 정량적 프로세스 존재	<ul style="list-style-type: none"> • 조직 차원의 프로세스 성과 관리 • 정량적 프로젝트 관리
❺ 최적화 단계	프로세스를 지속적으로 개선	<ul style="list-style-type: none"> • 조직 차원의 혁신 활동 전개 • 근본 원인 분석 및 해결

53. 단계적 표현 방법의 성숙 단계(3)

■ CMMI 모델의 단계적 표현



54. 단계적 표현 방법의 성숙 단계(4)

- 초기^{initial} 단계 : 프로세스 없음



그림 9-22 초기 단계의 개념 비유 예

55. 단계적 표현 방법의 성숙 단계(5)

- 초기initial 단계 : 프로세스 없음

- 프로세스가 정확히 정의 안됨 → 개발자 임의로 프로세스 정의 및 수시로 바꿈

통제가 불가능하고 비공식적이며 수행 결과도 예측하기 어렵다.

프로젝트 리더의 능력에 따른 프로젝트 성공, 실패

개인 역량이 뛰어난 한두 명에 의해 프로젝트의 결과가 결정

- 개발 조직: 팀 협업 안됨, 적시에 교육 훈련 제공 안됨, 체계적 관리 안됨, 개발과 유지보수를 위한 조직 환경의 뒷받침이 불안정

- 개발 과정: 무질서, 계획과 일정 구별 못함, 진척 상황 측정 안됨

제품개발 과정 추적 및 통제 안됨 → 임시방편으로 처리.

계획된 절차 없이 개발 → 시간에 쫓김 → 분석/설계 없이 바로 코딩

56. 단계적 표현 방법의 성숙 단계(6)

- 관리^{managed} 단계: 프로젝트 별로 프로세스 존재



그림 9-23 관리 단계의 개념 비유 예

57. 단계적 표현 방법의 성숙 단계(7)

■ 관리^{managed} 단계: 프로젝트 별로 프로세스 존재

① 기본 프로세스 확립 → 프로젝트 예산, 일정, 기능에 대해 추적 및 예측이 가능

{ 문제가 발생 시: 규칙화되고 훈련된 프로세스에 따라 움직임
결과물에 대해 사용자의 요구를 만족시키는지 확인 가능
프로세스 원칙이 잘 지켜짐 → 개인 경험보다는 과거의 축적된 데이터를 참조함

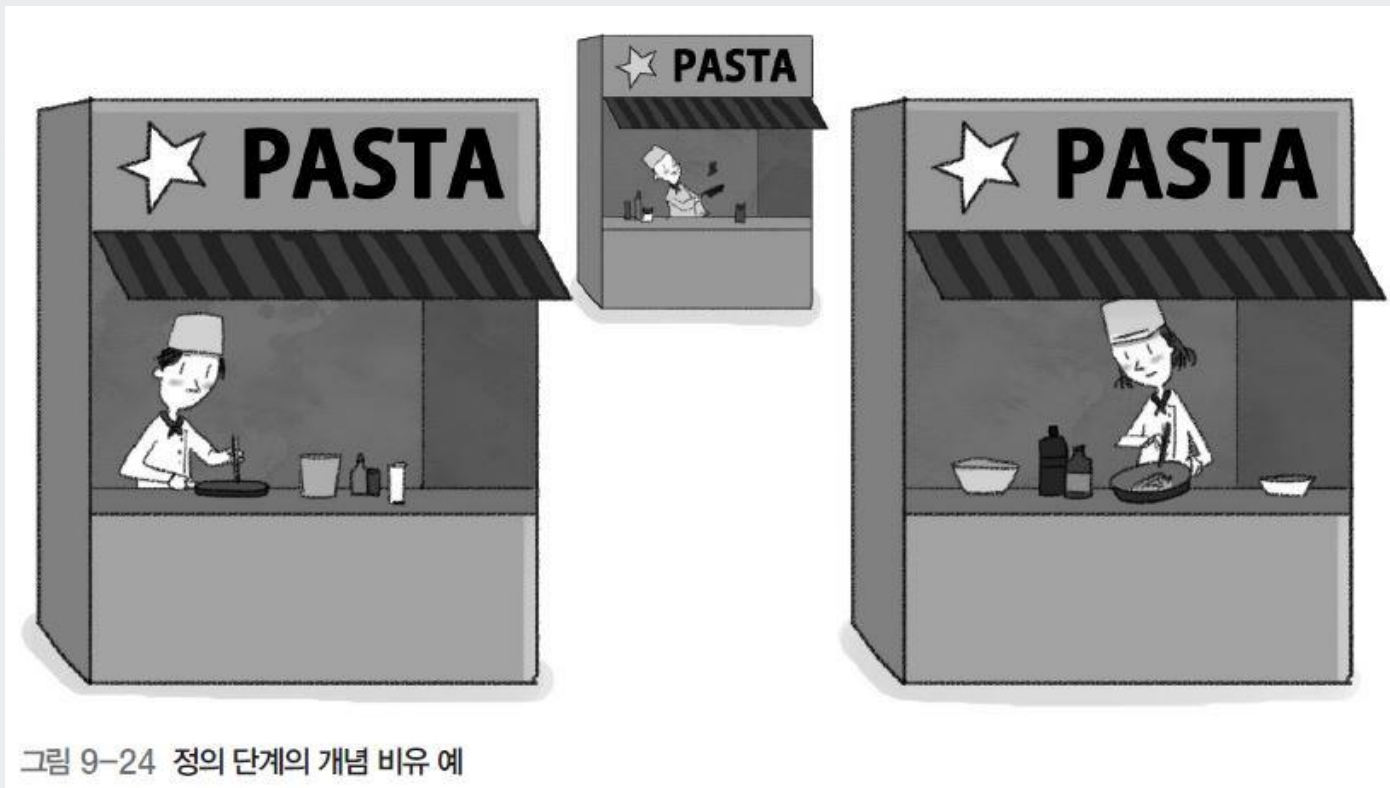
② 개발 조직: 프로젝트 관리를 위한 정책을 세우고 절차들을 실행

③ 개발 과정: 기본 프로젝트 관리 통제 체계가 확립 됨

{ 요구 사항이 명확하게 문서로 작성
프로젝트 계획과 진행 상황 추적 가능

58. 단계적 표현 방법의 성숙 단계(8)

- 정의^{defined} 단계 : 조직 차원의 표준 프로세스 존재



59. 단계적 표현 방법의 성숙 단계(9)

■ 정의^{defined} 단계 : 조직 차원의 표준 프로세스 존재

① 표준화된 프로세스 존재

{ 개별 프로젝트: 제공된 가이드를 참조로 표준 프로세스를 수정하여 활용
조직 전체: 표준 프로세스를 따름

② 조직 내: 소프트웨어 프로세스를 전담하기 위한 SEPG 존재

SEPG: 관리자를 위한 기술 교육 실시

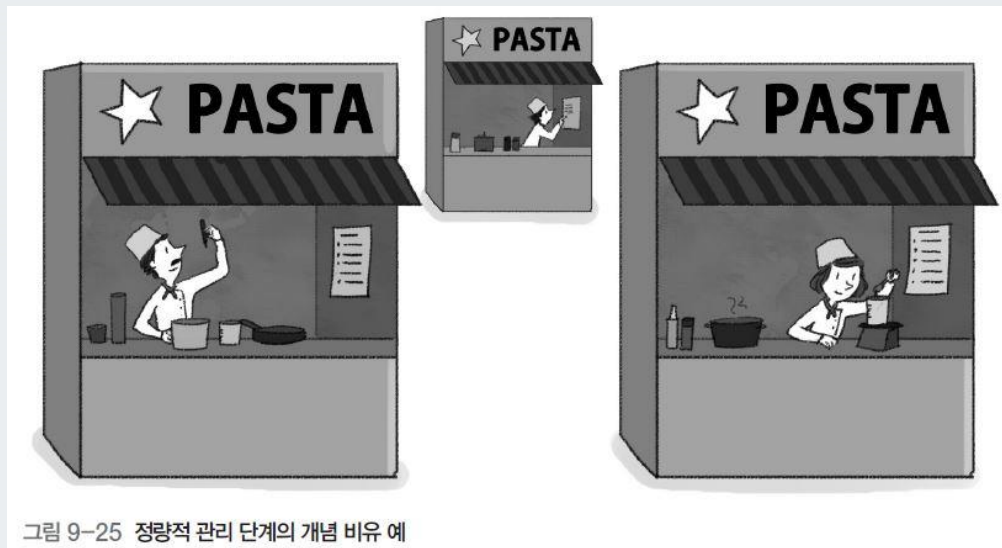
③ 개발 과정: 표준 프로세스를 사용하여 개발 공정, 비용, 일정 등을 통제 가능

{ 소프트웨어 품질 또한 추적 가능하여 관리 활동이 전반적으로 안정적
잘 지켜지는 표준화와 일관성으로 프로세스 안정화

60. 단계적 표현 방법의 성숙 단계(10)

- 정량적 관리(quantitatively managed) 단계: 측정 가능한 정량적 프로세스

수치화, 계량화 { SW 프로세스
제품의 품질 } 철저한 관리 → 성과 예측 가능



61. 단계적 표현 방법의 성숙 단계(11)

■ 최적화optimizing 단계 : 프로세스를 지속적으로 개선

- 지금까지 조직 차원에서 사용해오던 표준 프로세스를 개선하여 최적화
- 표준 프로세스 검토 후 보완 및 최신 기술 반영 → 지속적 프로세스 개선 → 개선된 프로세스를 전 조직이 사용

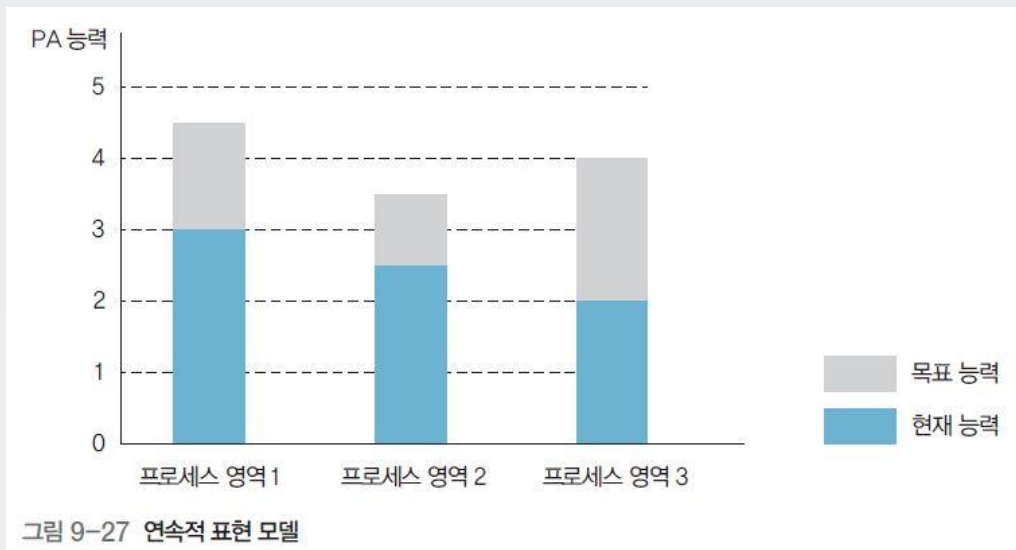


그림 9-26 최적화 단계의 개념 비유 예

62. 연속적 표현 방법의 능력 단계(1)

■ 능력 단계

- 프로세스 영역 능력 수준을 측정하는 연속적 표현 모델
- 해당 조직의 각 프로세스 영역에 대한 능력이 얼마나 되는지를 나타냄
- 프로세스 영역별 능력 수준을 확인 → 잘되는 영역, 떨어지는 영역 구별 파악 가능



63. 연속적 표현 방법의 능력 단계(2)

- 프로세스 영역별 능력 수준 점검의 이점
 - 떨어지는 프로세스 영역: 집중 관리 가능
 - 중요한 프로세스 영역: 집중적으로 자원 투입 → 능력 수준 향상

64. SPICE(ISO 15504) 모델(1)

■ SPICE 모델

- 소프트웨어 프로세스 평가를 위한 프레임워크를 제공
- 정보 시스템 분야에 특화된 품질 표준이자 인증 규격의 역할

표 9-17 SPICE 모델의 프로세스

범주	내용
고객-공급 customer-supplier 프로세스	인수, 공급, 요구 도출, 운영 : 소프트웨어를 개발하여 고객에게 전달하는 것을 지원, 소프트웨어를 정확하게 운용하고 사용하도록 하기 위한 프로세스로 구성(프로세스 10개)
공학 engineering 프로세스	개발, 소프트웨어 유지보수 : 시스템과 소프트웨어 제품의 명세화, 구현, 유지보수하는 프로세스로 구성(프로세스 9개)
지원 support 프로세스	문서화, 형상, 품질 보증, 검증, 확인, 리뷰, 감사, 품질 문제 해결 : 소프트웨어 생명주기에서 다른 프로세스에 의해 이용되는 프로세스로 구성(프로세스 8개)
관리 management 프로세스	관리, 프로젝트 관리, 품질/위험 관리 : 소프트웨어 생명주기에서 프로젝트 관리자에 의해 사용되는 프로세스로 구성(프로세스 4개)
조직 organization 프로세스	조직 배치, 개선 활동 프로세스, 인력 관리, 기반 관리, 측정 도구, 재사용 : 조직의 업무 목적을 수립하고, 조직이 업무 목표를 달성하는 데 도움을 주는 프로세스로 구성(프로세스 9개)

65. SPICE(ISO 15504) 모델(2)

■ SPICE 모델의 프로세스 수행 능력 단계

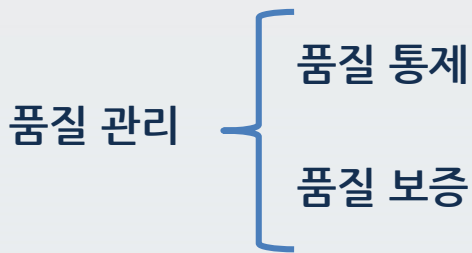
표 9-18 SPICE 모델의 프로세스 수행 능력 단계

단계	내용
1 불완전 ^{incomplete} 단계 : 미구현 또는 미달성	<ul style="list-style-type: none">• 프로세스가 구현되지 않음.• 프로세스가 목적을 달성하지 못함.
2 수행 ^{performed} 단계 : 프로세스 수행 및 목적 달성	<ul style="list-style-type: none">• 프로세스를 수행하고 목적을 달성함.• 프로세스가 정의된 산출물을 생산함.
3 관리 ^{managed} 단계 : 프로세스 수행 계획 및 관리	<ul style="list-style-type: none">• 정의된 자원의 한도 내에서 그 프로세스가 작업 산출물을 인도함.
4 확립 ^{established} 단계 : 정의된 표준 프로세스 사용	<ul style="list-style-type: none">• 소프트웨어 공학 원칙을 기반으로 정의된 프로세스를 수행함.
5 예측 ^{predictable} 단계 : 프로세스의 정량적 이해 및 통제	<ul style="list-style-type: none">• 프로세스가 목적 달성을 위해 통제됨.• 프로세스가 양적 측정을 통해 일관되게 수행됨.
6 최적화 ^{optimizing} 단계 : 프로세스를 지속적으로 개선	<ul style="list-style-type: none">• 프로세스 수행을 최적화함.• 지속적 개선을 통해 업무 목적을 만족시킴.

66. 품질 관리

■ 품질 관리quality management

- 개발의 각 단계에서 일어나는 모든 활동과 활동 중에 생성되는 여러 산출물을 통제하고 보증하여 품질을 관리하기 위한 활동



■ 품질 통제quality control

- 품질 절차와 표준을 개발자들이 준수하도록 프로세스를 정의하고 규정을 만드는 것
- 목적: 품질 좋은 소프트웨어를 만들기 위함

67. 품질 보증(1)

■ IEEE 정의

- 개발된 소프트웨어가 사용자의 요구를 만족시킨다는 것을 보장하는 데 필요한 계획적이고 체계적인 활동
- 개발된 소프트웨어가 기술적인 요구 사항과 일치하는가를 적절하게 확인하는 데 필요한 체계적이고도 계획적인 유형의 활동

■ 품질 보증quality assurance 정의

소프트웨어의 결함을 줄여 품질 좋은 소프트웨어를 만들기 위해,
사용자가 요구하는 품질 수준을 파악하고
이를 어떻게 달성할 수 있는지를 정의하는 개발 단계 전역에 걸친 체계적인 작업

68. 품질 보증(2)

■ 품질 보증 활동

- 개발 단계 전역에 걸쳐 품질에 영향을 미치는 문제점을 조기에 발견하여 제거하는 것
- 개발된 소프트웨어의 품질이 목표한 수준에 있다는 것을 보증
- 소프트웨어 개발 단계 전역에 걸쳐 적용되는 보호 활동umbrella activity

■ 개발 과정 전·후의 품질 보증 작업

표 9-19 개발 과정의 전·후 작업

단계	작업
개발 시작 전 표준 지침서 개발과 프로그래밍	<ul style="list-style-type: none">• 표준 지침서(개발과 프로그래밍 표준 지침서, 관리 표준 지침서)를 개발한다.• 그 지침서를 어떻게 준수할 것인지에 대한 습득 훈련을 한다.
개발 중	<ul style="list-style-type: none">• 요구 분석 명세서나 설계 사양서 등을 검토한다.• 테스트를 수행한다.• 각종 산출물의 문서를 검토한다.
개발 완료 후	<ul style="list-style-type: none">• 유지보수 및 변경 관리 계획을 검토한다.

69. 품질 보증(3)

■ 품질 보증의 기대 효과

- 개발 체계 및 품질 환경 적립
 - 명확한 요구 사항 정립 및 검토와 개발 기법의 표준화로 프로그램의 이해도 증진
 - 문서 작성의 표준화 유도과 품질 정보의 체계적 관리로 유사 문제점 재발 방지
- 개발 시스템 품질 향상
 - 개발 초기 단계에서 문제점 발견 및 보완
 - 객관적 품질 평가로 사용성 증대

■ 품질 보증의 문제점

- 품질 보증에 대한 인식 부족
- 품질 요원을 중요시하지 않는 시각에 따른 경험 많은 품질 보증 요원의 부족
- 품질 보증에 대한 제도의 표준화와 절차 확립의 부족

70. 품질 보증(4)

■ IEEE의 품질 보증 계획서

표 9-20 IEEE의 품질 보증 계획서

1. 목적
2. 참고 문헌
3. 관리
 - 1) 조직
 2. 업무 분담
 - 3) 책임 한계
4. 문서화
 - 1) 목적
 - 2) 문서화 요구 사항
 - ① 소프트웨어 요구 분석 명세서
 - ② 소프트웨어 설계서
 - ③ 소프트웨어 검증 및 확인 계획
 - ④ 사용자 지침서
 - 3) 기타
5. 표준 제도, 규범 및 규약
 - 1) 목적
 - 2) 내용
6. 검토 및 검사
 - 1) 목적
 - 2) 필요 사항
 - ① 소프트웨어 요구 검토
 - ② 기초 설계 검토
 - ③ 상세 설계 검토
 - ④ 소프트웨어 검증 및 확인
 - ⑤ 기능 검사
 - ⑥ 물리적 검사
 - ⑦ 절차 검사
 - ⑧ 관리 목적용 검토
7. 형상 관리
8. 문제 보고 및 시정 조치
9. 도구, 기법, 방법론
10. 코드 통제
11. 미디어 통제
12. 납품 업체 통제
13. 기록 보완 및 유지



다음 시간

기본 자바 소개 3



송실사이버대학교

송실사이버대학교의 강의콘텐츠는
저작권법에 의하여 보호를 받는바, 무단
전재, 배포, 전송, 대여 등을 금합니다.

* 사용처: 나눔글꼴