



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0140154
(43) 공개일자 2014년12월09일

(51) 국제특허분류(Int. Cl.)

G06F 17/00 (2006.01)

(21) 출원번호 10-2013-0060204

(22) 출원일자 2013년05월28일

심사청구일자 2013년05월28일

(71) 출원인

서강대학교산학협력단

서울특별시 마포구 백범로 35 (신수동, 서강대학교)

(72) 발명자

박수용

서울특별시 마포구 상수동 369 두산위브아파트 101동 1001호

최수진

서울 양천구 목동동로 130, 1423동 1306호 (신정동, 목동14단지아파트)

김대규

경기 안산시 단원구 선부광장남로 113, 1202동 1507호 (선부동, 주공12단지아파트)

(74) 대리인

특허법인충현

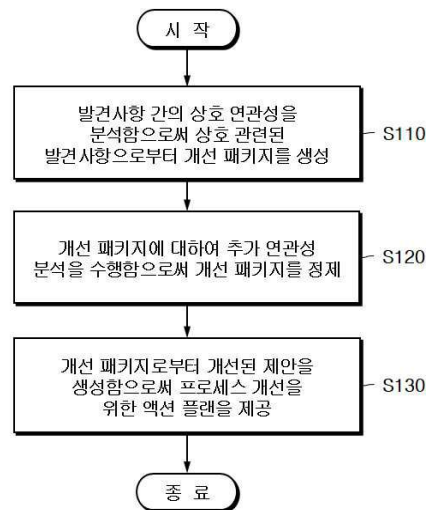
전체 청구항 수 : 총 15 항

(54) 발명의 명칭 소프트웨어 프로세스 개선을 추천하는 장치 및 방법

(57) 요약

소프트웨어 프로세스 개선을 추천하는 장치 및 방법이 개시된다. 소프트웨어 프로세스 개선을 추천하는 방법은, 서로 다른 프로세스에서 식별된 발견사항(findings) 간의 상호 연관성(correlation)을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지(improvement packages)를 생성하고, 생성된 개선 패키지에 대하여 추가 연관성 분석을 수행함으로써 개선 패키지를 정제하며, 정제된 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜(action plan)을 제공한다.

대 표 도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호 2012033348

부처명 미래창조과학부

연구관리전문기관 서강대학교 산학협력단

연구사업명 차세대정보컴퓨팅기술개발사업

연구과제명 동적 자가 적응을 위한 집단 지성 기반 컴포넌트 설계 및 검증 기술

기 여 율 1/1

주관기관 서강대학교 산학협력단

연구기간 2012.07.01 ~ 2017.06.30

특허청구의 범위

청구항 1

적어도 하나의 처리기(processor)를 이용하여 소프트웨어 프로세스 개선을 추천하는 방법에 있어서,
서로 다른 프로세스에서 식별된 발견사항(findings) 간의 상호 연관성(correlation)을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지(improvement packages)를 생성하는 단계;
생성된 상기 개선 패키지에 대하여 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써 개선 패키지를 정제하는 단계; 및
정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜(action plan)을 제공하는 단계;를 포함하는 방법.

청구항 2

제 1 항에 있어서,
개선 패키지를 생성하는 단계는,
프로세스 심사를 통해 식별된 발견사항을 입력받는 단계;
입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하는 단계; 및
산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로서 그룹핑(grouping)하는 단계;를 포함하는 방법.

청구항 3

제 2 항에 있어서,
상기 개선 패키지는,
이미 식별된 상호 연관된 발견사항을 그룹핑하되, 상기 개선 패키지 내에 포함되는 발견사항의 수와 프로세스 영역의 수가 많고, 개선 패키지 간의 중복이 최소화되도록 그룹핑하는 것을 특징으로 하는 방법.

청구항 4

제 1 항에 있어서,
상기 조직 관점에서 추가 연관성 분석을 수행하는 경우, 상기 개선 패키지를 정제하는 단계는,
상기 소프트웨어 프로세스 개선을 수행하는 조직 내에 보유한 자원과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 자원과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정하는 것을 특징으로 하는 방법.

청구항 5

제 1 항에 있어서,
상기 소프트웨어 라이프사이클 관점에서 추가 연관성 분석을 수행하는 경우, 상기 개선 패키지를 정제하는 단계는,
소프트웨어 라이프사이클상에서 시간적인 연관성을 고려하여 라이프사이클과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 라이프사이클과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정하는 것을 특징으로 하는 방법.

청구항 6

제 1 항에 있어서,

상기 심사 모델 관점에서 추가 연관성 분석을 수행하는 경우, 상기 개선 패키지를 정제하는 단계는,

상기 소프트웨어 프로세스 개선을 수행하는 조직에서 사용하는 프로세스 심사모델의 제약사항을 고려하여 상기 심사 모델과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 심사 모델과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정하는 것을 특징으로 하는 방법.

청구항 7

제 1 항에 있어서,

상기 비즈니스 가치 관점에서 추가 연관성 분석을 수행하는 경우, 상기 개선 패키지를 정제하는 단계는,

비즈니스 우선순위에 따라 상기 개선 패키지의 우선순위를 결정하도록 상기 비즈니스 우선순위와 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 비즈니스 우선순위와 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정하는 것을 특징으로 하는 방법.

청구항 8

제 1 항에 있어서,

상기 프로세스 개선을 위한 액션을 플랜을 제공하는 단계는,

미리 저장된 개선 제안 명세 템플릿(template)에 따라 상기 개선 패키지 별로 요약 정보를 입력받음으로써 요약 개선 제안이 식별되는 단계; 및

식별된 상기 요약 개선 제안에 대한 개선 패키지 내의 발견사항들로부터 단위 개선 항목을 식별함으로써 상세 개선 제안이 생성되는 단계;를 포함하는 방법.

청구항 9

제 1 항 내지 제 8 항 중에 어느 한 항의 방법을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

청구항 10

소프트웨어 프로세스 개선을 추천하는 장치에 있어서,

프로세스 심사를 통해 서로 다른 프로세스에서 식별된 발견사항을 입력받는 입력부; 및

적어도 하나의 처리기를 구비하여 입력된 상기 발견사항 간의 상호 연관성을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지를 생성하고, 생성된 상기 개선 패키지를 정제하며, 정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜을 제공하는 처리부;를 포함하는 장치.

청구항 11

제 10 항에 있어서,

참조 모델에서 제시된 프로세스 상호 연관성과 소프트웨어 프로세스 개선 과정에서 실험적으로 수집된 상호 연관성 데이터에 기반하여 생성된 프로세스 상호 연관성 모델을 저장하는 모델 저장부;를 더 포함하고,

상기 처리부는, 저장된 상기 프로세스 상호 연관성 모델을 참조하여 입력된 상기 발견사항 간의 입출력 여부를 판별함으로써 상기 발견사항 간의 상호 연관성을 분석하는 것을 특징으로 하는 장치.

청구항 12

제 11 항에 있어서,

상기 프로세스 상호 연관성 모델은,

프로세스 참조 모델에서 발견사항이 사상(mapping)되는 활동(practice) 간의 연관 강도(correlation weight)를

나타내는 제 1 연관 강도(process model relation, PMR)를 포함하는 제 1 그래프; 및
실험적으로 수집된 프로젝트 사례 중 활동 간의 관계의 발생 빈도에 따라 부여되는 가중치를 나타내는 제 2 연관 강도(improvement package relation, IPR)를 포함하는 제 2 그래프;에 기초하여 생성되며,
상기 제 1 연관 강도 및 상기 제 2 연관 강도를 합산하여 상기 프로세스 상호 연관성 모델의 전체 연관 강도(relation weight, RW)를 산출하는 것을 특징으로 하는 장치.

청구항 13

제 11 항에 있어서,
상기 처리부는,
상기 프로세스 상호 연관성 모델을 참조하되,
발견사항 간 연관 강도가 최대가 되고, 하나의 개선 패키지에 포함되는 발견사항의 수가 최대가 되며, 개선 패키지 간 중복이 최소가 되도록 상기 상호 연관된 발견사항을 그룹핑하는 것을 특징으로 하는 장치.

청구항 14

제 10 항에 있어서,
상기 처리부는,
입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하고,
산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로써 그룹핑함으로써 개선 패키지를 생성하는 것을 특징으로 하는 장치.

청구항 15

제 10 항에 있어서,
상기 처리부는,
생성된 상기 개선 패키지에 대하여 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정하는 것을 특징으로 하는 장치.

명세서

기술 분야

[0001] 본 발명은 소프트웨어 프로세스 개선을 추천하는 기술에 관한 것으로, 특히 프로세스 심사 결과를 입력받아 발견사항으로부터 상호 연관성을 분석하여 소프트웨어 프로세스 개선을 위해 개선된 제안을 제공할 수 있는 장치, 방법 및 그 방법을 기록한 기록매체에 관한 것이다.

배경 기술

[0002] 많은 소프트웨어 조직이 비즈니스 목표를 달성하기 위해 소프트웨어 프로세스 개선 활동을 추진하고 있다. 소프트웨어 프로세스 개선활동은 일반적으로 현수준 심사, 개선계획 수립, 그리고 개선계획 이행 및 모니터링의 사이클로 이루어진다. 현수준 심사는 프로세스 참조 모델(process reference model)을 기반으로 소프트웨어 조직의 소프트웨어 프로세스 이행 수준과 참조 모델간의 차이를 이해하는 활동이다. 가장 대표적인 소프트웨어 프로세스 참조 모델로는 미국 SEI(Software Engineering Institute)의 CMMI(Capability Maturity Model Integration)와 ISO/IEC 15504(가칭 SPICE), 그리고 한국 표준인 소프트웨어 프로세스 품질인증 기준(가칭 K-model)이 있다. 프로세스 개선 계획은 현수준 심사 결과를 기반으로 수립되며, 개선 계획의 이행 및 모니터링을 통해 프로세스의 수립 및 내재화가 이루어진다. 그리고 심사, 계획 수립, 이행의 프로세스 개선 사이클이 다시 반복되면서 지속적인 프로세스 개선 활동이 수행된다.

[0003] 지속적인 소프트웨어 프로세스 개선을 위해 많은 기업은 프로세스 심사 및 그에 따른 프로세스 개선 계획 수립

을 주기적으로 수행하고 있으며, 이러한 프로세스 재심사 비율은 지속적으로 증가하고 있다. 이와 관련하여 성공적인 소프트웨어 프로세스 개선 활동을 위해서는 프로세스 개선 계획의 품질이 중요함은 아무리 강조해도 지나치지 않다.

[0004] 한편, 프로세스 참조 모델 및 심사 방법에 대해서는 표준 제정 및 연구 활동이 활발하며 이행 사례가 누적되고 있는 반면, 프로세스 심사 결과로부터 프로세스 개선 계획을 수립하는 활동에 대해서는 체계적인 방법에 대한 연구가 아직 부족하고 개선 현장에서는 프로세스 전문가에게 의존해 상당 부분 임의적으로 이행되는 것이 현실이다.

[0005] 이하에서 인용되는 선행기술문헌에는 CMMI 참조 모델을 위한 프로세스 심사 방법과 프로세스 개선 사이클 가이드라인이 제시되어 있으나, 여전히 소프트웨어 프로세스 심사를 통해 도출된 강점을 활용한 프로세스 개선 제안을 체계적으로 정의할 수 있는 기술적 수단을 제공하지 못하고 있다.

선행기술문헌

비특허문헌

[0006] (비특허문헌 0001) SCAMPI Upgrade Team. Standard CMMI appraisal method for process improvement (SCAMPI) A, version 1.3: Method definition document. Technical Report CMU/SEI-2011-HB-001, Software Engineering Inst., Carnegie Mellon University, 2011.

(비특허문헌 0002) Bob McFeeley. IDEAL: A user's guide for software process improvement. Technical Report CMU/SEI-96-HB-001, Software Engineering Inst., Carnegie Mellon University, 1996.

발명의 내용

해결하려는 과제

[0007] 본 발명의 실시예들이 해결하고자 하는 기술적 과제는, 소프트웨어 프로세스 심사를 통해 도출된 강점을 활용하여 프로세스 개선 제안을 정의하는 구조화된 기술적 수단이 존재하지 못함으로써, 프로세스 개선 계획 수립을 개인의 역량에 의존함으로써 인해 결과물의 품질을 예측하기 어렵다는 문제점을 해결하고, 조직의 강점이 존재함에도 불구하고 이를 활용하지 못하고 약점만을 강조한 프로세스 개선 계획이 수립될 수 있다는 한계를 극복하고자 한다.

과제의 해결 수단

[0008] 상기 기술적 과제를 해결하기 위하여, 본 발명의 일 실시예에 따른 적어도 하나의 처리기(processor)를 이용하여 소프트웨어 프로세스 개선을 추천하는 방법은, 서로 다른 프로세스에서 식별된 발견사항(findings) 간의 상호 연관성(correlation)을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지(improvement packages)를 생성하는 단계; 생성된 상기 개선 패키지에 대하여 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써 개선 패키지를 정제하는 단계; 및 정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜(action plan)을 제공하는 단계;를 포함한다.

[0009] 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서, 개선 패키지를 생성하는 단계는, 프로세스 심사를 통해 식별된 발견사항을 입력받는 단계; 입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하는 단계; 및 산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로서 그룹핑(grouping)하는 단계;를 포함한다. 또한, 상기 개선 패키지는, 이미 식별된 상호 연관된 발견사항을 그룹핑하되, 상기 개선 패키지 내에 포함되는 발견사항의 수와 프로세스 영역의 수가 많고, 개선 패키지 간의 중복이 최소화되도록 그룹핑한다.

[0010] 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서, 상기 프로세스 개선을 위한 액션을 플랜을 제공하는 단계는, 미리 저장된 개선 제안 명세 템플릿(template)에 따라 상기 개선 패키지 별로 요약 정보를 입력받음으로써 요약 개선 제안이 식별되는 단계; 및 식별된 상기 요약 개선 제안에 대한 개선 패키지 내의 발견사항들로부터 단위 개선 항목을 식별함으로써 상세 개선 제안이 생성되는 단계;를 포함한다.

- [0011] 한편, 이하에서는 상기 기재된 소프트웨어 프로세스 개선을 추천하는 방법을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체를 제공한다.
- [0012] 상기 기술적 과제를 해결하기 위하여, 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치는, 프로세스 심사를 통해 서로 다른 프로세스에서 식별된 발견사항을 입력받는 입력부; 및 적어도 하나의 처리기를 구비하여 입력된 상기 발견사항 간의 상호 연관성을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지를 생성하고, 생성된 상기 개선 패키지를 정제하며, 정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜을 제공하는 처리부;를 포함한다.
- [0013] 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치는, 참조 모델에서 제시된 프로세스 상호 연관성과 소프트웨어 프로세스 개선 과정에서 실험적으로 수집된 상호 연관성 데이터에 기반하여 생성된 프로세스 상호 연관성 모델을 저장하는 모델 저장부;를 더 포함하고, 상기 처리부는, 저장된 상기 프로세스 상호 연관성 모델을 참조하여 입력된 상기 발견사항 간의 입출력 여부를 판별함으로써 상기 발견사항 간의 상호 연관성을 분석한다.
- [0014] 또한, 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치에서, 상기 프로세스 상호 연관성 모델은, 프로세스 참조 모델에서 발견사항이 사상(mapping)되는 활동(practice) 간의 연관 강도(correlation weight)를 나타내는 제 1 연관 강도(process model relation, PMR)를 포함하는 제 1 그래프; 및 실험적으로 수집된 프로젝트 사례 중 활동 간의 관계의 발생 빈도에 따라 부여되는 가중치를 나타내는 제 2 연관 강도(improvement package relation, IPR)를 포함하는 제 2 그래프;에 기초하여 생성되며, 상기 제 1 연관 강도 및 상기 제 2 연관 강도를 합산하여 상기 프로세스 상호 연관성 모델의 전체 연관 강도(relation weight, RW)를 산출한다.
- [0015] 또한, 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치에서, 상기 처리부는, 상기 프로세스 상호 연관성 모델을 참조하되, 발견사항 간 연관 강도가 최대가 되고, 하나의 개선 패키지에 포함되는 발견사항의 수가 최대가 되며, 개선 패키지 간 중복이 최소가 되도록 상기 상호 연관된 발견사항을 그룹핑한다.
- [0016] 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치에서, 상기 처리부는, 입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하고, 산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로서 그룹핑함으로써 개선 패키지를 생성한다.
- [0017] 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치에서, 상기 처리부는, 생성된 상기 개선 패키지에 대하여 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정한다.

발명의 효과

- [0018] 본 발명의 실시예들은, 정량적인 기술 수단을 이용하여 체계화되고 구조화된 개선 계획의 수립이 가능하고, 경험이 부족한 사용자도 소프트웨어 프로세스 개선을 위한 고려 사항을 누락없이 검토할 수 있으며, 상호 연관성 분석을 통해 각 활동이 어떻게 다른 활동에 영향을 미치는지 이해할 수 있어 구체적인 개선의 필요성을 제시할 수 있을 뿐만 아니라, 대상 조직의 특성에 적합한 개선 제안의 제공이 가능하다.

도면의 간단한 설명

- [0019] 도 1은 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법을 도시한 흐름도이다.
- 도 2는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 발견사항 상호 연관성 매트릭스를 예시한 도면이다.
- 도 3은 도 2의 상호 연관성 매트릭스를 이용하여 발견사항 간의 상호 연관성을 도출하는 과정을 설명하기 위한 도면이다.
- 도 4a 및 도 4b는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 개선 패키지를 정제하기 위해 활용될 수 있는 발견사항-조직 관점 매트릭스를 예시한 도면이다.
- 도 5는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 개선된 제안을 생성하기 위해 활용되는 상세 개선 제안을 예시한 도면이다.

도 6은 단위 개선 항목의 식별 과정을 설명하기 위한 예시도이다.

도 7은 도 6의 그룹핑 과정을 통해 생성된 상세 개선 제안을 예시한 도면이다.

도 8은 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치를 도시한 블록도이다.

도 9는 본 발명의 일 실시예에 따른 도 8의 소프트웨어 프로세스 개선을 추천하는 장치에서 상호 연관성 모델을 생성 과정을 설명하기 위한 도면이다.

도 10a 및 도 10b는 각각 참조 모델 및 케이스 스터디에 기반한 프로세스 상호 연관성 그래프를 생성하는 과정을 설명하기 위한 도면이다.

도 11은 도 10a 및 도 10b의 상호 연관성 그래프를 이용하여 최종적으로 생성된 하이브리드 프로세스 상호 연관성 모델을 예시한 도면이다.

도 12a 및 도 12b는 도 11의 하이브리드 프로세스 상호 연관성 모델을 활용하여 개선 패키지를 제안하는 과정을 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0020] 본 발명의 실시예들을 설명하기에 앞서, 소프트웨어 프로세스 개선에 대한 관련 연구들과 문제점들을 소개한 후, 이들 문제점을 해결하기 위해 본 발명의 실시예들이 채택하고 있는 기술적 수단을 순차적으로 제시하도록 한다.
- [0021] 소프트웨어 프로세스 개선 계획은 소프트웨어 프로세스 심사 결과를 근간으로 하여 수립된다. 프로세스 심사의 목적은 소프트웨어 조직의 강점과 약점을 식별하고, 이로부터 개선 계획 수립을 위한 개선 제안(improvement recommendations)을 제공하는데 있다. 심사 결과로부터 도출된 개선 제안은 개선 계획의 '무엇을 개선할 것인가(what)'에 해당하는 핵심적인 요소이다. 따라서 심사팀으로부터 적절한 개선 제안이 제공되지 않는다면, 개선 계획의 품질에 부정적인 영향을 끼치게 된다.
- [0022] 프로세스 심사는 일반적으로 프로세스 참조 모델과 함께 제공되는 심사 방법론에 따라 수행되며 대표적인 심사 방법론으로는 CMMI 참조 모델을 위한 SCAMPI(Standard CMMI Appraisal Method for Process Improvement)가 있다.
- [0023] 이러한 방법론은 조직 소프트웨어 프로세스의 강점과 약점을 발견하는데 집중되어 있으며 개선 제안은 프로세스 심사 결과의 필수 산출물로 여기지 않고 있다. 따라서 식별된 강점과 약점으로부터 개선 제안을 정의하는 방법에 대해서도 구체적으로 다루고 있지 않다. IDEAL과 같은 프로세스 개선 사이클 가이드라인은 현수준 심사, 개선 계획 수립, 그리고 이행 및 모니터링과 관련한 주요 역할과 행동을 다루고 있으나, 개선 제안 도출과 관련하여서는 심사팀과 대상 조직 내 이해관계자 간의 아이디어 도출 회의 수행 정도로 간단히 언급하고 있으며, 구체적인 방안은 제시하고 있지 않다. 이는 프로세스 개선 계획 수립이 프로세스 전문가에 의해서만 수행될 수 있음을 의미하며, 그 수행 과정이나 작성된 계획의 품질을 예측하기 어려움을 의미한다.
- [0024] 또한, 현재의 프로세스 심사 방법론 및 프로세스 개선 사이클 가이드는 강점보다는 약점의 식별에 중점을 두고 있으며, 강점이 식별된다 하더라도 이후 개선 프로세스에서 강점이 어떻게 활용되는지에 대한 구체적인 언급이 부족하다. 이는 프로세스 개선 계획 수립 및 이행시 강점이 기여할 수 있는 부분을 제약하는 결과를 초래하게 된다.
- [0025] 나아가, 소프트웨어 프로세스 참조 모델은 상호 연관된 수개에서 수십여개의 방대한 프로세스로 구성되어 있다(예를 들어, CMMI 22개, SPICE 48개, K-model 17개). 강점과 약점은 각 프로세스 별로 식별되나, 개선 계획의 수립시에는 프로세스 간 상호 연관성을 고려하여 시너지를 최대화하는 방향으로 개선 계획이 수립될 필요가 있다. 그러나, 구체적인 개선 제안 개발 절차의 부재는 프로세스간 상호 연관성을 고려한 개선 제안의 수립을 어렵게 하며, 이는 프로세스 개선 현장에서 심사 이후 조치 사항의 식별이 개별 약점으로부터 단편적으로 이루어질 위험으로 이어지게 된다.
- [0026] 즉, 소프트웨어 프로세스 심사를 통해 도출된 강점과 약점으로부터 프로세스 개선 제안을 정의하는 체계적인 절차가 마련되지 못한 경우, 프로세스 개선 계획 수립을 개인의 역량에 의존하게 되고, 그에 따라 결과물의 품질을 예측하기 어렵고, 조직의 강점이 존재함에도 불구하고 이를 활용하지 못하고 약점만을 강조한 프로세스 개선 계획이 수립될 수 있으며, 프로세스간 상호 연관성과 시너지를 고려한 프로세스 개선 계획 수립이 어렵다.

- [0027] 이에 따라, 이하에서 기술되는 본 발명의 실시예들은 상기된 문제점들을 해소하기 위해 소프트웨어 프로세스 개선을 위한 개선 제안 모델을 제안하고자 한다. 본 발명의 실시예들은 프로세스 심사 결과인 강점과 약점을 입력받아 개선 제안을 결과로서 출력하는 구체적인 절차와 기술적 수단을 제시한다. 이는 강점에 대한 고려, 프로세스 측면의 상호 연관성에 기반한 발견사항 분석, 그리고 개선 대상 조직의 특성(비즈니스 목표, 개발 라이프사이클, 인적/물적 자원, 선정한 프로세스 심사모델 등)을 감안한 발견사항의 분석, 분석 결과를 기반으로 한 통합적인 개선 제안 명세를 통해 이루어진다. 또한, 본 발명의 실시예들은 프로세스 개선 제안 모델의 실제 현장에서의 적용 용이성을 향상시키기 위해 자동화 도구를 제공한다.
- [0028] 한편, 본 발명의 실시예들에서 사용하는 용어는 다음과 같이 정의된다.
- [0029] - 소프트웨어 프로세스 개선(software process improvement) : '시스템의 품질은 시스템을 획득, 개발, 유지보수하는 프로세스의 품질에 좌우된다'는 전제하에 소프트웨어 프로세스 개선을 통해 소프트웨어 품질을 향상시키고자 하는 활동
- [0030] - 발견사항(finding) : 프로세스 심사를 통해 발견한 강점 및 약점
- [0031] - 강점(strength) : 참조 모델의 요구사항을 뛰어 넘어서 구현된 모범이 되는 사례
- [0032] - 약점(weakness) : 참조 모델의 요구사항을 충족하지 못하거나 비효과적으로 구현된 사례
- [0033] - 개선 패키지(improvement package) : 상호 연관된 발견 사항들의 집합으로 개선 제안 개발의 단위가 됨
- [0034] - 개선 제안(recommendation) : 약점을 해결하거나 강점을 확산하기 위한 프로세스 개선 제안
- [0035] 이하에서는 첨부된 도면을 참조하여 본 발명의 실시예들을 구체적으로 설명한다.
- [0036] 도 1은 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법을 도시한 흐름도이다. 도 1의 방법은 소프트웨어 프로세스 개선을 위한 추천 시스템에 의해 수행될 수 있으며, 이러한 추천 시스템은 적어도 하나의 물리적인 하드웨어인 처리기(processor)를 구비하되, 필요에 따라 연산에 필요한 자원으로서 메모리(memory)와 저장소(storage)가 활용될 수도 있다.
- [0037] S110 단계에서, 소프트웨어 프로세스 개선을 위한 추천 시스템은 서로 다른 프로세스에서 식별된 발견사항(findings) 간의 상호 연관성(correlation)을 분석함으로써 상호 관련된 발견사항으로부터 개선 패키지(improvement packages)를 생성한다. 이러한 상호 연관성 분석 과정에서는 프로세스 심사를 통해 발견된 강점과 약점을 모두 입력값으로 사용한다.
- [0038] S120 단계에서, 추천 시스템은 S110 단계를 통해 생성된 상기 개선 패키지에 대하여 조직 관점(organizational view), 소프트웨어 라이프사이클 관점(software life cycle view), 심사 모델 관점(assessment model view) 및 비즈니스 가치 관점(business value view) 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써 개선 패키지를 정제한다.
- [0039] S130 단계에서, 추천 시스템은 S120 단계를 통해 정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜(action plan)을 제공한다. 이 때, 개선된 제안은 2 단계에 걸쳐, 상위 레벨의 요약 개선 제안(abstract recommendation)이 식별되고, 그런 다음 구체적인 상세 개선 제안(concrete recommendation)으로 상세화된다.
- [0040] 이하에서는 각각의 과정을 보다 구체적으로 설명하도록 한다.
- [0041] **(1) 상호 연관성 분석(correlation analysis)**
- [0042] 프로세스 심사를 통해 식별된 발견사항은 상호 연관성 분석을 통해 관련된 발견사항들은 하나의 개선제안 패키지로 묶이게 된다. 이러한 상호 연관성 분석은 두가지 액티비티(activity)를 가질 수 있는데, 발견사항 상호 연관성 분석(analyzing findings correlations) 액티비티와 개선 패키지 식별(identifying improvement packages) 액티비티가 그것이다.
- [0043] 첫째, 발견사항 상호 연관성 분석 액티비티에서는 발견사항-발견사항 간의 대칭 매트릭스를 통해 상호 연관된

발견사항들이 식별된다. 예를 들어, 개선 패키지를 생성하는 과정은, 프로세스 심사를 통해 식별된 발견사항을 입력받고, 입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하며, 산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로써 그룹핑(grouping)함으로써 수행될 수 있다.

[0044] 도 2는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 발견사항 상호 연관성 매트릭스를 예시한 도면으로서, 하나의 발견사항에 대해 다른 모든 발견사항이 상호 연관성 측면에서 검토된다.

[0045] 발견사항 간의 상호 연관성은 참조 모델이 제시하는 프로세스 간의 상호 연관성에 기인하여 식별되게 된다. 이는 프로세스 심사에서 개별 발견사항은 참조 모델의 특정 프로세스에 대해 식별되기 때문이다.

[0046] 참조 모델과 발견사항 간의 관계는 도 3과 같이 표현될 수 있으며, 도 3은 도 2의 상호 연관성 매트릭스를 이용하여 발견사항 간의 상호 연관성을 도출하는 과정을 설명하기 위한 도면이다.

[0047] 둘째, 개선 패키지 식별 액티비티에서는 이전 액티비티에서 식별된 상호 연관된 발견사항들을 묶은 개선 패키지를 식별한다. 이때, 개선 패키지 내에 포함되는 발견사항의 수와 프로세스 영역의 수가 많고, 개선 패키지 간 중복은 최소화하는 방향으로 그룹핑함으로써 개선 패키지를 식별하는 것이 바람직하다. 개선 패키지 간 유사성이 높은 것은 검토를 통해 하나로 통합된다. 이를 통해 수십여개의 발견사항은 통상 수개의 개선 패키지로 그룹핑된다.

[0048] (2) 개선 패키지 정제(refining improvement packages)

[0049] 앞서 상호 연관성 분석 과정을 통해 식별된 개선 패키지는 추가적인 상호 연관성 분석을 통해서 정제된다. 본 발명의 실시예들이 제안하고 있는 관점은 총 4가지로서 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점이 그것이다.

[0050] 첫째, 조직 관점은 조직 내 보유한 자원에 대한 연관성 측면에서 개선 제안 패키지를 분석한다. 즉, 동일한 자원을 사용하는 서로 다른 발견사항을 연관된 것으로 식별한다. 만약, 조직 관점에서 추가 연관성 분석을 수행하는 경우, 개선 패키지를 정제하는 과정은, 소프트웨어 프로세스 개선을 수행하는 조직 내에 보유한 자원과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 자원과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정할 수 있다.

[0051] 둘째, 소프트웨어 라이프사이클 관점은 소프트웨어 라이프사이클상에서 시간적인 연관성 측면에서 개선 제안 패키지를 분석한다. 만약, 소프트웨어 라이프사이클 관점에서 추가 연관성 분석을 수행하는 경우, 개선 패키지를 정제하는 과정은, 소프트웨어 라이프사이클상에서 시간적인 연관성을 고려하여 라이프사이클과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 라이프사이클과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정할 수 있다.

[0052] 셋째, 심사 모델 관점은 조직에서 사용하는 프로세스 심사 모델의 제약사항 관점에서 개선 제안 패키지를 검토한다. 만약, 심사 모델 관점에서 추가 연관성 분석을 수행하는 경우, 개선 패키지를 정제하는 과정은, 소프트웨어 프로세스 개선을 수행하는 조직에서 사용하는 프로세스 심사모델의 제약사항을 고려하여 상기 심사 모델과 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 심사 모델과 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정할 수 있다.

[0053] 넷째, 비즈니스 가치 관점에서는 비즈니스 우선 순위에 따라 개선 제안의 우선 순위를 결정한다. 만약, 비즈니스 가치 관점에서 추가 연관성 분석을 수행하는 경우, 개선 패키지를 정제하는 과정은, 비즈니스 우선순위에 따라 상기 개선 패키지의 우선순위를 결정하도록 상기 비즈니스 우선순위와 상기 발견사항을 각각 행과 열로 설정한 매트릭스를 이용하여 상기 비즈니스 우선순위와 상기 발견사항 간의 상호 연관성을 산출함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정할 수 있다.

[0054] 각 정제 단계를 거치면서 개선 제안 패키지 내의 발견사항이 제거되거나 추가되기도 하고, 개선 제안 패키지가 분리되거나 유사한 개선 제안 패키지로 통합될 수도 있다. 또한, 4가지 관점에서의 새로운 발견사항이 추가 기술될 수도 있다.

[0055] 도 4a 및 도 4b는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 개선 패키지를 정제하기 위해 활용될 수 있는 발견사항-조직 관점 매트릭스를 예시한 도면이다. 도 4a 및 도 4b를 참조하면,

프로세스 관점의 상호 연관성은 없으나, 조직 관점의 상호 연관성이 식별되어 일부 발견사항이 개선 패키지로 추가되는 것을 확인할 수 있다.

(3) 개선 제안 생성(building recommendations)

이제, 앞서 개선 패키지 정제 과정을 통해 정제된 개선 패키지를 기반으로 개선 제안이 생성된다. 이러한 개선 제안을 생성하는 과정은 두가지 액티비티를 포함할 수 있는데, 요약 개선 제안 정의(defining abstract recommendation) 액티비티와 상세 개선 제안 생성(building concrete recommendation) 액티비티가 그것이다.

첫째, 요약 개선 제안 정의 액티비티에서는 다음의 표 1의 요약 개선 제안 템플릿(template)에 따라 각 개선 패키지의 개선 의도와 방법을 표현한다. 즉, 미리 저장된 개선 제안 명세 템플릿에 따라 개선 패키지 별로 요약 정보를 사용자로부터 입력받음으로써 요약 개선 제안이 식별된다.

표 1

Type	Findings	Description Template
W-S	Strengths, Weaknesses	IMPROVE <weakness>* BY [[USING <strength>*], [IMPROVING <weakness>*]] WITH <resource>* IN <life cycle phase>* TO ACHIEVE <business objective>*
W-W	Weaknesses only	IMPROVE <weakness>* BY IMPROVING <weakness>* WITH <resource>* IN <life cycle phase>* TO ACHIEVE <business objective>*
S-S	Strengths only	ENHANCE <strength>* BY USING <strength>* WITH <resource>* IN <life cycle phase>* TO ACHIEVE <business objective>*
S-W	Strengths, Weaknesses	ENHANCE <strength>* BY [[USING <strength>*], [IMPROVING <weakness>*]] WITH <resource>* IN <life cycle phase>* TO ACHIEVE <business objective>*

또한, 요약 개선 제안은 도 5의 예시와 같이 무엇을(IMPROVE 절), 어떤 방법으로(BY 절), 어떤 리소스를 활용하여(WITH 절), 언제(IN 절), 왜(TO ACHIEVE 절)를 요약하여 명시한다. 도 5는 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 방법에서 개선된 제안을 생성하기 위해 활용되는 상세 개선 제안을 예시하고 있다.

둘째, 상세 개선 제안 생성 액티비티에서는 개선 패키지의 요약 개선 제안이 세부 개선 제안으로 상세화되는데, 앞서 식별된 요약 개선 제안에 대한 개선 패키지 내의 발견사항들로부터 단위 개선 항목(recommendation seed)을 식별함으로써 상세 개선 제안이 생성되게 된다. 단위 개선 항목은 강점은 조직 내에 확산하고, 약점은 개선하거나 대안을 제시하는 방향으로 식별한다. 상호 관련된 단위 개선 항목은 도 6과 같이 그룹핑되고 이를 기반으로 한 추가적인 개성 항목이 식별되면서 상세 개선 제안이 점차 완성되게 된다. 도 6은 단위 개선 항목의 식별 과정을 설명하기 위한 예시도로서, 이러한 과정을 통해 서로 다른 프로세스로부터 식별된 발견사항들이 상호 시너지를 창출하는 방향으로 개선 제안이 개발된다. 도 7은 도 6의 그룹핑 과정을 통해 완성된 상세 개선 제안을 예시하고 있다.

도 8은 본 발명의 일 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치(800)를 도시한 블록도로서, 앞서 제시된 본 발명의 실시예들을 자동화된 하드웨어를 이용한 장치 구성을 중심으로 재구성하였다. 따라서, 설명의 중복을 피하기 위해 여기서는 물리적인 장치를 중심으로 그 기능을 약술하도록 한다.

소프트웨어 프로세스 개선을 추천하는 장치(800)는 크게 입력부(10)와 처리부(20)를 포함하며, 모델 저장부(30)를 더 포함하는 것이 바람직하다.

입력부(10)는, 프로세스 심사를 통해 서로 다른 프로세스에서 식별된 발견사항을 입력받는다.

처리부(20)는, 적어도 하나의 처리기를 구비하여 입력된 상기 발견사항 간의 상호 연관성을 분석함으로써 상호

관련된 발견사항으로부터 개선 패키지를 생성하고, 생성된 상기 개선 패키지를 정제하며, 정제된 상기 개선 패키지로부터 개선된 제안을 생성함으로써 프로세스 개선을 위한 액션 플랜을 제공한다. 이러한 처리부(20)는 논리적으로 분리된 기능 단위의 처리 모듈을 포함할 수 있다.

[0066] 처리부(20)는, 입력된 상기 발견사항을 각각 행과 열로 설정한 대칭 매트릭스를 이용하여 상기 발견사항 간의 입출력 여부를 판별하여 상호 연관성 가중치를 산출하고, 산출된 상기 상호 연관성 가중치가 큰 순서대로 소정 개수의 발견사항을 개선 패키지로써 그룹핑함으로써 개선 패키지를 생성할 수 있다. 또한, 처리부(20)는, 생성된 상기 개선 패키지에 대하여 조직 관점, 소프트웨어 라이프사이클 관점, 심사 모델 관점 및 비즈니스 가치 관점 중 적어도 하나의 관점에서 추가 연관성 분석을 수행함으로써, 상기 개선 패키지 또는 상기 개선 패키지 내의 발견사항을 수정할 수 있다.

[0067] 한편, 본 발명의 실시예들이 채택하고 있는 상호 연관성 모델은 도 9에 제시된 구조에 따라 참조 모델에서 제시한 프로세스 상호 연관성과 실제 사례 연구를 적용한 개선 프로젝트에서 실험적으로 분석된 상호 연관성 데이터를 기반으로 한 하이브리드 모델로서 제안한다.

[0068] 도 9에 도시된 바와 같이, 하이브리드 상호 연관성 그래프(930)는 참조 모델에 기반하여 생성된 상호 연관성 그래프(910)과 케이스 스터디에 기반한 상호 연관성 그래프(920)의 결합에 의해 도출 가능하며, 각각의 그래프(910, 920)는 그래프의 연관 강도로서 PMR(process model relation)과 IPR(improvement package relation)의 두가지 값을 생성함으로써 이로부터 하이브리드 상호 연관성 그래프의 연관 강도를 결정하게 된다.

[0069] 이를 위해 도 8의 소프트웨어 프로세스 개선을 추천하는 장치(800)는 모델 저장부(30)를 더 포함할 수 있다. 모델 저장부(30)는, 참조 모델에서 제시된 프로세스 상호 연관성과 소프트웨어 프로세스 개선 과정에서 실험적으로 수집된 상호 연관성 데이터에 기반하여 생성된 프로세스 상호 연관성 모델을 저장하는 구성으로서, 처리부(20)는 모델 저장부(30)에 저장된 상기 프로세스 상호 연관성 모델을 참조하여 입력된 상기 발견사항 간의 입출력 여부를 판별함으로써 상기 발견사항 간의 상호 연관성을 분석할 수 있다.

[0070] 도 10a는 참조 모델에 기반한 프로세스 상호 연관성 그래프를 생성하는 과정을 설명하기 위한 도면으로서, 프로세스 참조 모델로부터 활동(practice) 간의 관계를 그래프로 도시한 것이다. 각 노드(node)는 참조 모델상의 개별 활동이고, 연결선(edge)은 두 활동 간에 관계가 있음을 나타내며, 두 활동 간의 산출물(work product)의 흐름을 통해 추출 가능하다. PMR은 프로세스 참조 모델 측면에서 활동 간의 연관 강도(correlation weight)를 표현하는 것으로서, 서로 다른 프로세스에 있는 활동 간의 관계에 가중치를 부여한다.

[0071] 도 10b는 케이스 스터디에 기반한 프로세스 상호 연관성 그래프를 생성하는 과정을 설명하기 위한 도면으로서, 본 발명의 실시예에 따른 소프트웨어 프로세스 개선을 추천하는 장치를 통해 도출된 개선 패키지를 기준으로 활동 간의 관계를 그래프로 도식화한 것이다. IPR은 전체 프로젝트 사례 중 해당 관계의 발생 빈도에 따라 부여되는 가중치로서, 두 활동이 여러 개의 프로젝트 사례에서 반복적으로 동일한 개선 패키지로 묶인 경우 그 강도가 보다 높게 설정된다.

[0072] 도 11은 도 10a 및 도 10b의 상호 연관성 그래프를 이용하여 최종적으로 생성된 하이브리드 프로세스 상호 연관성 모델을 예시한 도면이다. 연관 강도(relation weight, RW)는 도 10a 및 도 10b 각각의 연관 강도인 IPR과 PMR을 합산하여 산출하게 된다.

[0073] 도 12a 및 도 12b는 도 11의 하이브리드 프로세스 상호 연관성 모델을 활용하여 개선 패키지를 제안하는 과정을 설명하기 위한 도면으로서, 하이브리드 프로세스 상호 연관성 모델을 심사 발견사항에 적용하여 개선 패키지를 제안하는 것을 보여준다.

[0074] 우선, 도 12a와 같이 대상 조직의 발견사항에 대해 하이브리드 프로세스 상호 연관성 모델을 기반으로 발견사항 간 관계를 그래프로 도식화한다. 이는 각 발견사항을 연결된 활동에 사상(mapping)시킴으로써 가능하다. 도 12a에서 발견사항 F'는 도 10a 및 도 10b의 활동 F에 사상된다. 하이브리드 프로세스 상호 연관성 모델의 두 활동 간의 연관 강도에 의해 두 발견사항 간의 연관 강도가 결정된다.

[0075] 다음으로, 도 12b와 같이 연관 강도를 기준으로 개선 패키지를 제안한다. 이때, 발견사항 간 연관 강도는 높고, 한 개선 패키지에 포함되는 발견사항의 수는 많으며, 개선 패키지 간 중복은 적은 조합을 생성하여 제시하는 것이 바람직하다.

[0076] 요약하건대, 프로세스 상호 연관성 모델은, 프로세스 참조 모델에서 발견사항이 사상(mapping)되는 활동(practice) 간의 연관 강도(correlation weight)를 나타내는 제 1 연관 강도(process model relation, PMR)를

포함하는 제 1 그래프 및 실험적으로 수집된 프로젝트 사례 중 활동 간의 관계의 발생 빈도에 따라 부여되는 가중치를 나타내는 제 2 연관 강도(improvement package relation, IPR)를 포함하는 제 2 그래프에 기초하여 생성된다. 또한, 상기 제 1 연관 강도 및 상기 제 2 연관 강도를 합산하여 상기 프로세스 상호 연관성 모델의 전체 연관 강도(relation weight, RW)를 산출하게 된다. 이때, 상기 프로세스 상호 연관성 모델을 참조하되, 발견 사항 간 연관 강도가 최대가 되고, 하나의 개선 패키지에 포함되는 발견사항의 수가 최대가 되며, 개선 패키지 간 중복이 최소가 되도록 상기 상호 연관된 발견사항을 그룹핑하는 것이 바람직하다.

[0077] 상기된 본 발명의 실시예들에 따르면, 기존의 소프트웨어 프로세스 개선 방법론들이 개인의 역량에 의존하여 수행되었던 것에 반해, 정량적인 기술 수단을 이용하여 체계화되고 구조화된 개선 계획의 수립이 가능하고, 경험 이 부족한 사용자도 소프트웨어 프로세스 개선을 위한 고려 사항을 누락없이 검토할 수 있으며, 상호 연관성 분석을 통해 각 활동이 어떻게 다른 활동에 영향을 미치는지 이해할 수 있어 구체적인 개선의 필요성을 제시할 수 있을 뿐만 아니라, 대상 조직의 특성에 적합한 개선 제안의 제공이 가능하다.

[0078] 한편, 본 발명의 실시예들은 컴퓨터로 읽을 수 있는 기록 매체에 컴퓨터가 읽을 수 있는 코드로 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다.

[0079] 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어 인터넷을 통한 전송)의 형태로 구현하는 것을 포함한다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다. 그리고 본 발명을 구현하기 위한 기능적인(functional) 프로그램, 코드 및 코드 세그먼트들은 본 발명이 속하는 기술 분야의 프로그래머들에 의하여 용이하게 추론될 수 있다.

[0080] 이상에서 본 발명에 대하여 그 다양한 실시예들을 중심으로 살펴보았다. 본 발명에 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명이 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다. 그러므로 개시된 실시예들은 한정적인 관점이 아니라 설명적인 관점에서 고려되어야 한다. 본 발명의 범위는 전술한 설명이 아니라 특허청구범위에 나타나 있으며, 그와 동등한 범위 내에 있는 모든 차이점은 본 발명에 포함된 것으로 해석되어야 할 것이다.

부호의 설명

[0081] 800 : 소프트웨어 프로세스 개선을 추천하는 장치

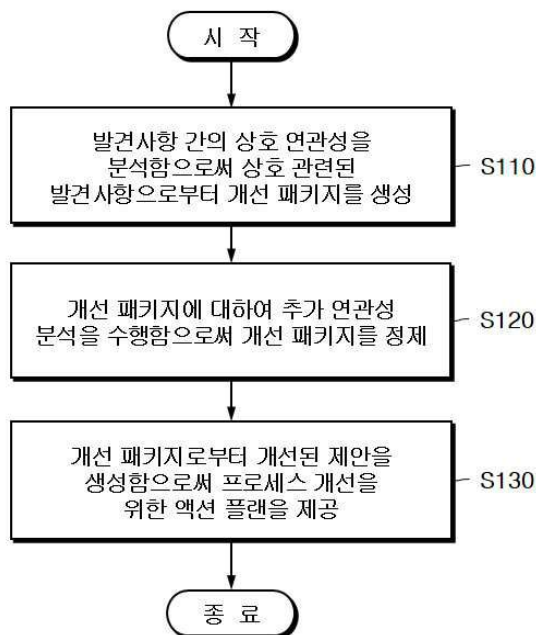
10 : 입력부

20 : 처리부

30 : 모델 저장부

도면

도면1

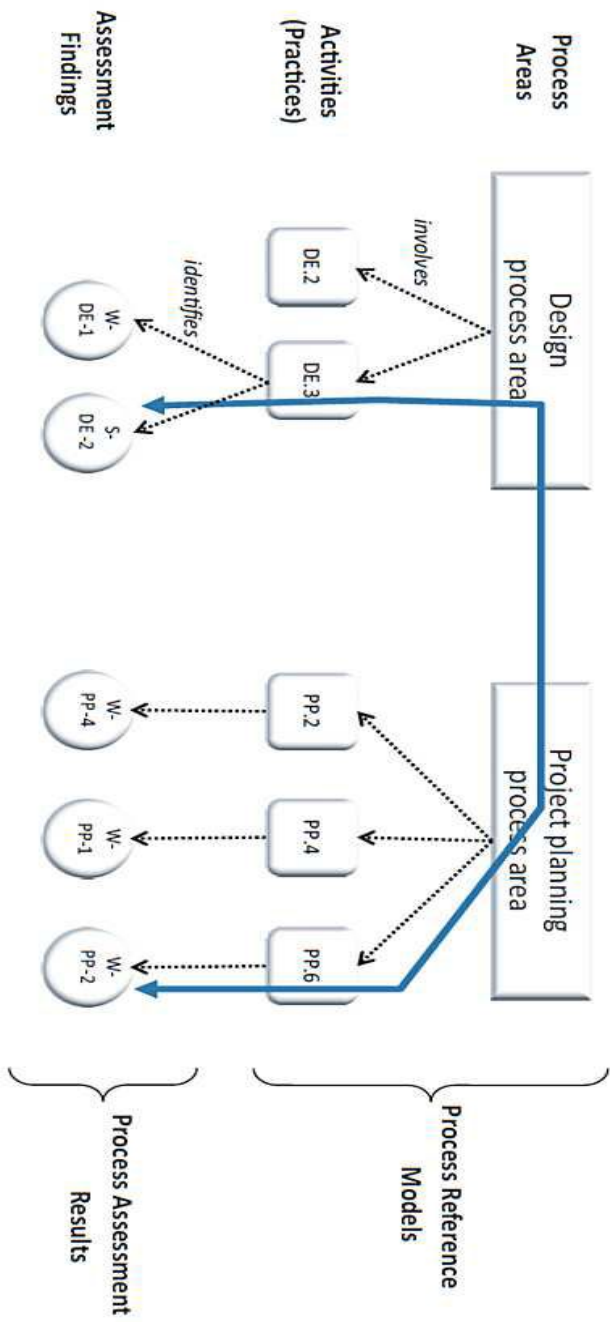


도면2

Correlation kinds i: input o: output i/o:input-output 1: self			Findings													
			Design		Implementation			Testing		Project Planning			M&A	Quality Assurance		
			S-DE-2	W-DE-1	S-IM-1	W-IM-1	W-IM-2	W-TE-1	W-TE-2	W-PP-1	W-PP-2	W-PP-4	S-MA-1	W-QA-2	W-QA-4	
Findings	Design	S-DE-2	1	i/o		i		o	i/o		o		i	i/o		
		W-DE-1	i/o	1		i		i/o				i/o				
	Implementa- tion	S-IM-1			1		o	o						i/o		
		W-IM-1	o	o		1		o	o					i/o		
		W-IM-2			i		1							i/o		
	Testing	W-TE-1	i	i/o	i	i		1						i/o		
		W-TE-2	i/o			i			1							
	Project Planning	W-PP-1								1			i			
		W-PP-2	i								1				i/o	
		W-PP-4										1				
	M&A	S-MA-1	o	i/o						o			1			
	Quality Assurance	W-QA-2	i/o		i/o	i/o	i/o	i/o						1	i	
		W-QA-4									i/o			o	1	
'i' weight			5	3	3	5	1	2	1	-	1	-	3	5	2	
'o' weight			5	4	1	1	2	5	2	1	2	-	1	6	1	
total weight			8	5	4	6	3	6	3	2	3	1	4	7	3	

M&A : Measurement & Analysis

도면3



도면4a

IP.S-DE-2
version 0.1

		Resource Types			Comments on resources
		Human Resources (HR)	Supporting tools (TL)	Other Resources	
Finding	S-DE-2				
	W-DE-1				
	S-IM-1		X		Daily build tool [TL], SVN (configuration repository) [TL]
	W-IM-1				
	W-TE-1				
	W-TE-2				
	W-PP-2				
	S-MA-1				
	W-QA-2	X	X		Although QA team exists, human resources are not sufficient [HR], Static analysis tool [TL], Performance testing tool [TL]
	W-IM-2		X		Static analysis tool [TL]
	W-PP-1				
	W-PP-4				
	W-QA-4		X		Project management system [TL]

added

도면4b

*IP.S-DE-2
version 0.2*

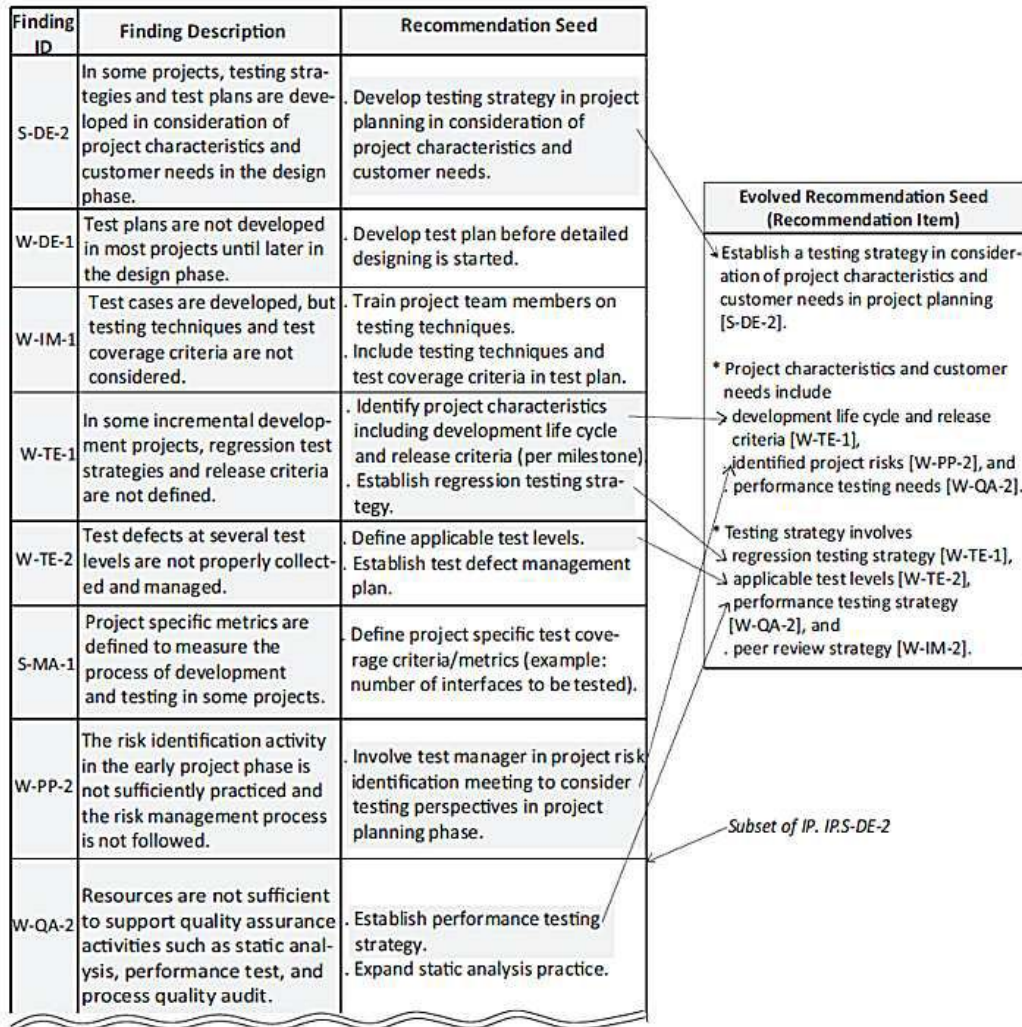
		Roles				
		PM	Tester	Developer	Configuration manager	QA (Quality Assurance)
Finding	S-DE-2		X	X		
	W-DE-1		X			
	S-IM-1			X	X	
	W-IM-1		X	X		
	W-TE-1		X			
	W-TE-2		X			
	W-PP-2	X				
	S-MA-1	X	X	X		
	W-QA-2					X
	W-IM-2			X		X
	W-PP-1	X				
	W-PP-4	X				
	W-QA-4	X				X

added

도면5

Recommendation ID	RECO01
Business Value	Very High
Base IP	IP.S-DE-2
Description	IMPROVE testing process BY USING testing strategy and test planning practice, and IMPROVING test planning process and quality assurance process WITH QA team and other automated tools IN project planning phase TO ACHIEVE system quality and time-to-market

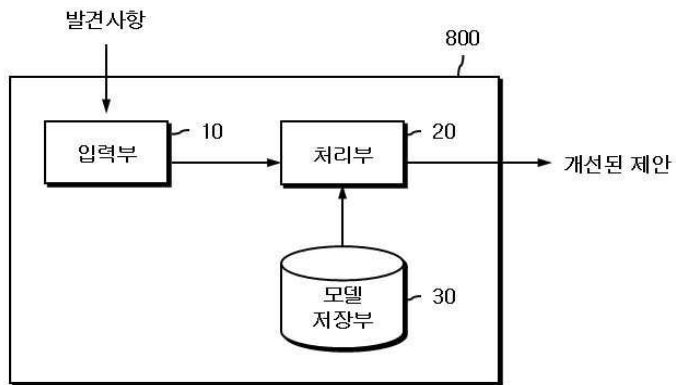
도면6



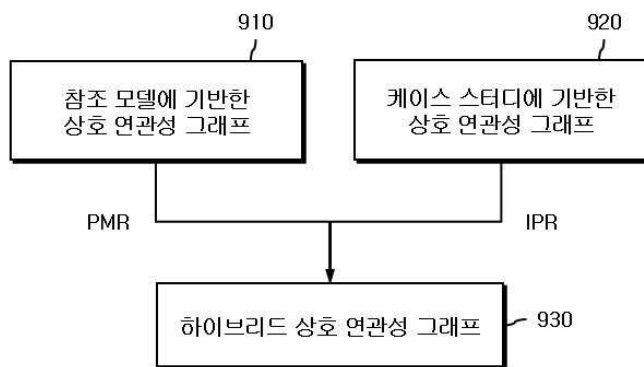
도면7

Concrete Recommendation	
Perspectives	Recommendation Items
<i>Recommendation Item in the previous figure</i> Software Life Cycle View	Project Planning phase 1) Establish testing strategy in consideration of project characteristics and customer needs such as project risks, development life cycle, release criteria, and performance testing needs [S-DE-2, W-PP-2, W-TE-1, W-QA-2]. 2) Establish testing strategy in project planning [S-DE-2]. * Test strategy may include regression test strategy [W-TE-1], applicable test levels [W-TE-2], performance testing strategy [W-QA-2], and peer review strategy [W-IM-2]. 3) Plan training on testing (e.g., test design) for project team members. 4) Conduct formal QA review for the established testing strategy [W-QA-2, W-ORG-2]. Design phase 1) Establish test plan before detailed designing is started [S-DE-2, W-DE-1, V-BIS-2]. * Test plan includes defect management plan [W-TE-2], project specific test coverage criteria and metrics (e.g., number of interfaces to be tested) [S-MA-1], and peer review plan (example criteria: focus peer review activity for defect-prone or changed codes) [W-IM-2]. 2) Conduct formal QA review activity for the established test plan [W-QA-2, W-ORG-2]. Testing Phase 1) Review progress of testing using defined test coverage criteria and metrics [W-IM-1, W-TE-1]. Deployment phase 1) Before product release, conduct formal review to ensure the test coverage is satisfied [W-IM-1, W-TE-1, W-ORG-2, V-SLC-1]. Project Closure phase 1) Collect the lessons learned and good practices of the project and store them for reference in other projects [V-SLC-1].
Organizational View	Tools and Systems 1) Integrate static analysis tool with auto-build system for frequent analysis with less effort [S-IM-1, W-IM-2, W-QA-2, V-ORG-1]. 2) Introduce new system or improve the existing PMS system for defect management and QA audit result management [V-ORG-1]. Human resources 1) Assign QA personnel responsible for supporting testing activities [V-ORG-2, V-ORG-3]. Role 1) Designate test manager during project planning phase and have test manager participate in project planning such as risk identification and development life cycle selection [W-PP-2]. 2) Have QA provide guidance on establishing testing strategy and test plan of each project and conduct formal milestone review on testing activity against established testing coverage criteria and metrics [V-ORG-2, V-ORG-3].
Assessment Model View	None
Business Value View	1) Establish risk identification checklist using historical risk data and encourage to use the checklist for project planning [V-BIS-3]. 2) Provide guidelines and training for regression testing [V-BIS-2].
Others	1) Improve training program on testing by referencing current good practices of projects [W-IM-1, W-TE-2, W-IM-2, V-BIS-3]. * training topics include 'test planning' for project managers and test managers and 'testing techniques' for test managers and developers.
Suggested Team Composition	
Include tester, project manager, and QA in the process action team	

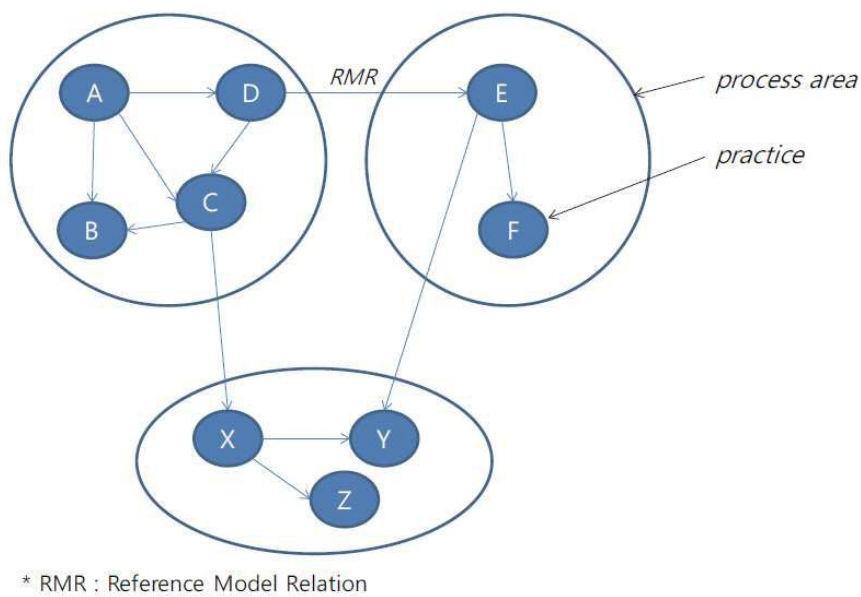
도면8



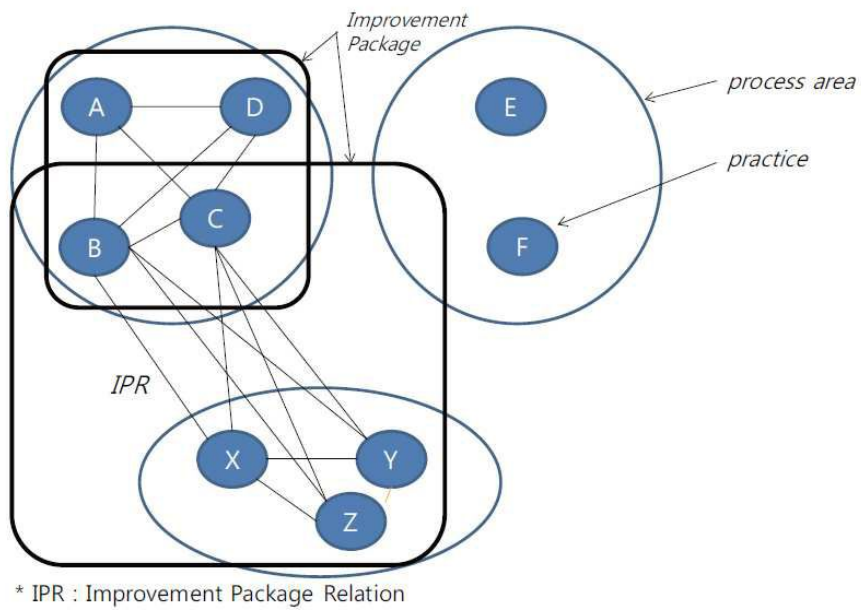
도면9



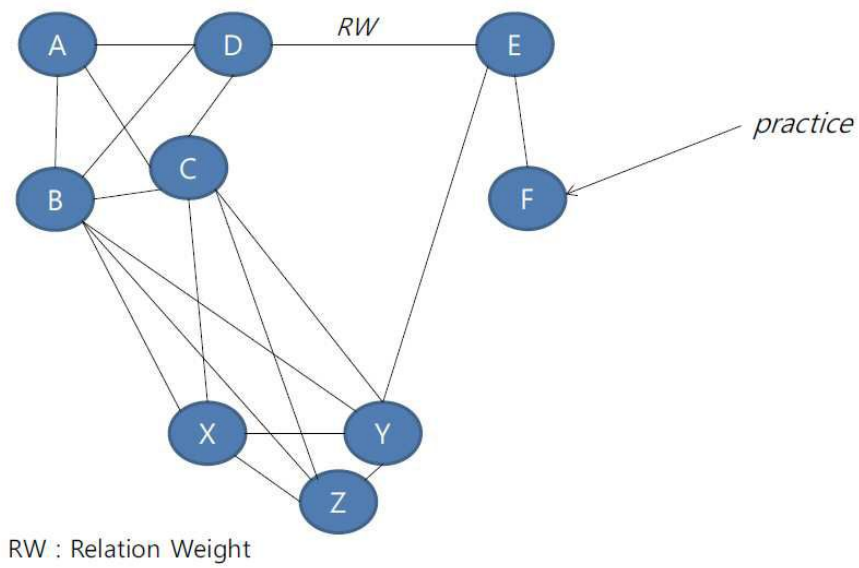
도면10a



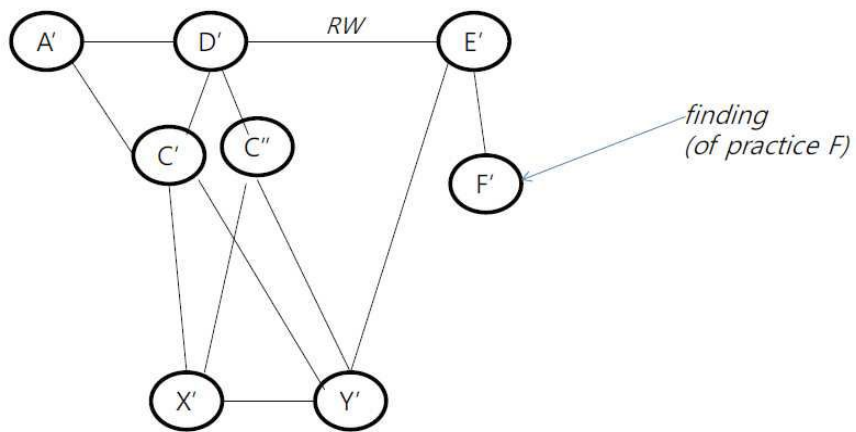
도면10b



도면11

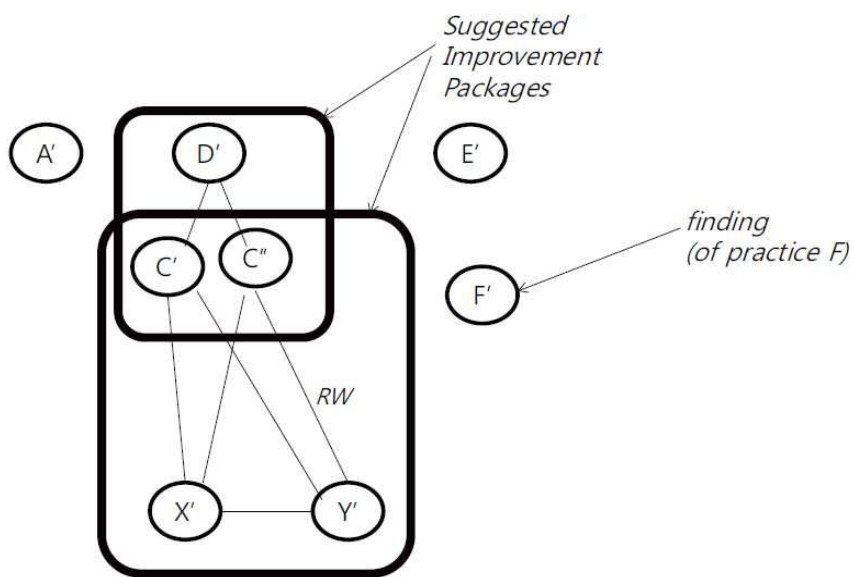


도면12a



RW : Relation Weight

도면12b



RW : Relation Weight