

CMMI기반 프로세스 지원도구를 위한 SIR-CM 분석 및 설계

박은주* · 김행곤**

1. 서 론

오늘 날 많은 IT기업들은 소프트웨어 및 시스템 개발과 관련해 세계 산업에서 사실상 표준이 된 CMMI 인증을 받기위해 많은 노력을 꺾하고 있다. 국내에서는 2007년 2월을 기준으로 5개 ·이 레벨 5에 도달했고, 레벨4가 8개, 레벨3이 25개, 레벨2가 12개로 총 50개 조직이 CMMI에 따른 공식 레벨을 확보하고 있다[1]. 이렇게 CMMI레벨을 획득함으로써 제품의 품질 향상은 물론이거니와 국내외 상품판매에 높은 신뢰를 얻을 수 있다. 그림 1은 여러 소프트웨어 프로세스 개선 모델을 통한 소프트웨어 생산성과 품질향상을 표현한 것이다.

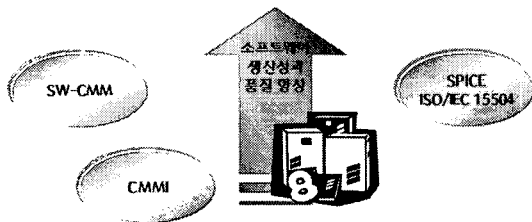


그림 1. 소프트웨어 프로세스 개선 모델

※ 교신저자(Corresponding Author): 김행곤, 주소: 경북 경산시 하양읍 금락1리 330번지(712-702), 전화: 053)850-2743, FAX: 053)850-2740, E-mail : hangkon@cu.ac.kr

* 대구가톨릭대학교 컴퓨터정보통신공학과 박사과정 (E-mail : ejpark@cu.ac.kr)

** 대구가톨릭대학교 컴퓨터정보통신공학과 교수
 ※ 본 논문은 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 육성, 지원사업의 연구결과로 수행되었음

그러나 인증을 받는 기업들 중 대부분은 대기업 혹은 SI 업계에서 알아주는 기업들이고, 소규모 중소기업은 찾아보기 힘들다. 그 이유는 소프트웨어 프로세스 개선 모델인 CMMI에서는 ‘무엇을 하라(What to do)’에 대한 내용만 있을 뿐, ‘어떻게 하라(How to do)’에 대한 내용은 없다[2]. 이러한 상황에서 소규모 중소기업들은 막대한 자원을 들여 CMMI 인증을 받을 만큼의 여유가 없다. 기존의 IBM Rational社와 Borland社에 많은 솔루션들이 있지만 어느 프로세스 영역에 특정지어 사용함으로써 각 영역별 산출되는 산출물을 CMMI 모델에 따라 활용하기란 쉽지 않다.

본 논문에서는 이러한 단점을 개선하고 궁극적으로는 소규모 조직의 소프트웨어 프로세스 개선을 돕기 위한 방안으로 CMMI기반 프로세스 지원 도구들의 산출물을 통합 관리하는 SIR-CM 시스템을 제안한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서 논문의 관련연구로 CMMI와 측정저장소 그리고 저장되는 산출물에 대한 관리 및 서비스를 위해 형상관리와 서비스 지향 아키텍처에 대해서 설명한다. 3장에서는 본 논문이 제안하는 SIR-CM을 분석하고 설계하며 메타데이터를 정의한다. 4장에서는 결론 및 연구 방향을 기술한다.

2. 관련연구

2.1 CMMI(Capability Maturity Model Integration)

CMMI는 1991년 미 국방부의 요청으로 미국 카네기멜론 대학 소프트웨어 공학연구소에서 개발한 소프트웨어 프로세스 성숙도 평가 모델로 조직의 소프트웨어 프로세스 개선 활동을 효율적으로 지원하는 모델이다. 2006년 8월 SEI에서 발표한 1.2 버전에는 총 22개의 프로세스 영역을 가지며 그림 2에서 나타내는 것처럼 CMMI 모델은 (a)단계적(Staged) 표현과 (b)연속적(Continuous) 표현 두 가지 방법으로 표현된다[3]. CMMI 지침서에서는 각 프로세스 영역의 특정 목표와 일반 목표를 기술하고, 이에 따르는 특정 수행과 일반 수행을 기술한다. 그 밖에 지침서에서는 목표를 달성하기 위한 수행을 GQM(Goal-Question-Metric) 기법을 사용하여 메트릭을 측정한다. 각

프로세스별 메트릭을 측정하고 CMMI 레벨 인증을 받기 위한 평가에서 메트릭 값을 활용한다[2].

2.2 측정저장소

프로세스 관리 활동에 있어서 측정 활동을 통해 현재의 프로세스 품질과 향후 프로세스 개선에 관한 정보를 효과적으로 얻을 수 있다[4]. CMMI 성숙도 레벨 2에 포함되는 프로세스 영역 중 Measurement and Analysis에서 측정 목표의 수립 및 측정 항목, 측정 절차를 정의하여 측정저장소에서 측정 데이터들을 저장, 관리한다. CMMI 성숙도 레벨 3의 프로세스 영역 중 Organization Process Definition에서 조직 차원의 측정저장소를 확립하여 프로세스의 자산으로써 유지 및 사용하는 활동을 지원토록 한다[2].

2.3 형상관리

형상관리는 하드웨어 혹은 소프트웨어 개발에 있어서 발생하는 형상을 관리하는 것이다. 개발 과정에 생기는 형상들이 어떻게 변경되는지를 기록하고 관리하는 프로세스로 그 정보를 저장소에 저장 관리한다. 기본적으로 형상관리에서는 형상들이 바뀔 때 마다 버전을 매겨 이 정보를 저장소에 기록하고 관리한다. 이는 변경관리 혹은 버전관리라고 하는 기능으로 이 정보를 바탕으로 개발자들은 전체적인 프로젝트 흐름에 반영하여 개발한다. 게다가 하나의 큰 프로젝트에 대해서 여러 명이 수행할 경우 다수의 병렬 작업을 지원하는 작업공간관리 기능이 있다. 이러한 형상관리를 도입한 소프트웨어 형상관리 도구들이 다양하게 존재한다. 오픈소스 프로젝트에 많이 쓰이는 CVS를 비롯하여 이를 좀 더 개선시킨 서브버전이 있고 상용화된 툴로는 ClearCase, PVCS, VSS 등 많은 제품들이 있다[5,6].

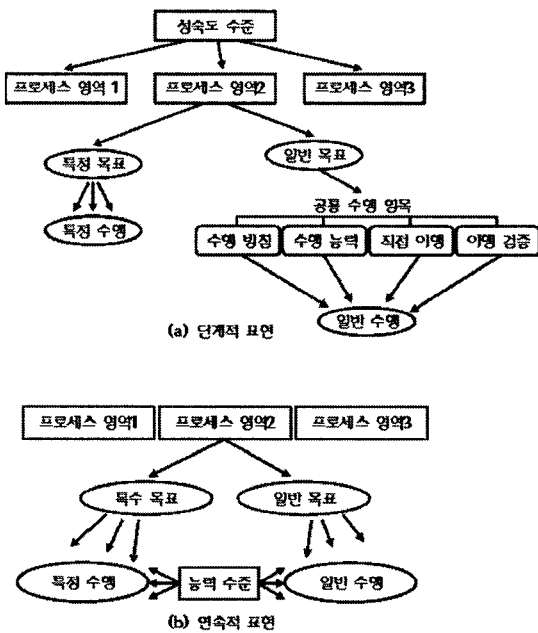


그림 2. CMMI 모델 구성요소

2.4 SOA

SOA는 서비스 지향 활동을 위한 기술적 기반이며, 원격으로 객체나 서비스에 접근하는 능력, 서비스 검색, 동적 바인딩 그리고 낮은 결합력을 가지는 특징을 가진다.

그림 3과 같이 아키텍처 내부의 역할들은 서비스 요청자(Service Requestor), 저장소(Discovery Agencies), 서비스 제공자(Service Provider)로 나누어져 있다. 또 SOA에서는 Publish, Find, Bind 등의 세 가지 오퍼레이션을 정의하고 있다. 서비스 제공자는 서비스의 제공을 위해 Publish를 통해 저장소에 서비스 제공의 위치와 정보들을 게시한다. 서비스 요청자는 Find를 통해 저장소에서 원하는 서비스를 검색을 하고, 검색된 서비스는 Bind를 통해 서비스 제공자에게 서비스를 제공받는다.

SOA의 기본적인 구성요소에는 SOAP, XML, UDDI, WSDL이 있다. SOAP은 정보를 실어 전달하는데 쓰이는 플랫폼 중립적인 표준 메시지 프로토콜이며 메시지의 내용은 XML로서 표현된다. WSDL는 서비스 사용자가 서비스 제공자가 공개한 인터페이스의 호출 방법을 설명하는 문서로서 XML로 작성되어 있다. 이 문서는 UDDI에 등록되며, 문서 안에는 서비스 요청자와 서비스 제공자 양자간에 전달되는 파라미터의 이름들, 등록된 웹 서비스가 실제 존재하는 URL(Uniform Resource Locator) 등의 정보가 기술되어 있다.

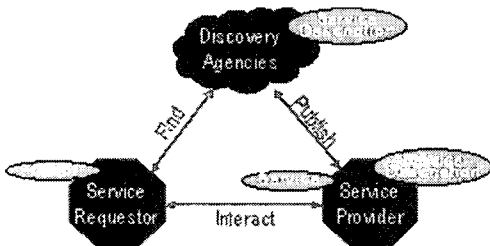


그림 3. 서비스 지향 아키텍처

UDDI는 서비스 제공자가 웹 서비스를 등록하고 서비스 요청자는 등록된 웹 서비스를 찾기 위한 프레임워크이다. 그러나 꼭 UDDI를 거쳐야만 되는 것은 아니고 서비스 제공자의 URL을 이미 알고 있다면 서비스 요청자가 바로 서비스 제공자를 호출할 수도 있다.

3. SIR-CM

3.1 개요

본 논문에서 제안하는 SIR-CM(SPIC Integration Repository - Configuration Management) 시스템은 소프트웨어 프로세스 개선과 평가를 위한 지원 도구들의 이질적인 산출물들을 통합하여 저장, 관리할 수 있는 저장소이다. SIR-CM을 중심으로 각 도구로부터의 산출물들이 저장되고, 관리된다. 또한 SIR-CM시스템에 작업 공간 관리와 프로세스관리를 포함하는 형상관리 기법을 적용하여, 과거 프로세스 정보들을 형상 관리의 측면에서 체계적으로 관리할 수 있도록 한다.

SIR-CM의 전체적인 시스템 구조는 그림 4와 같다. 소프트웨어 프로세스 개선을 지원하는 도구들의 모든 산출물을 입력 데이터로 저장하고 이를

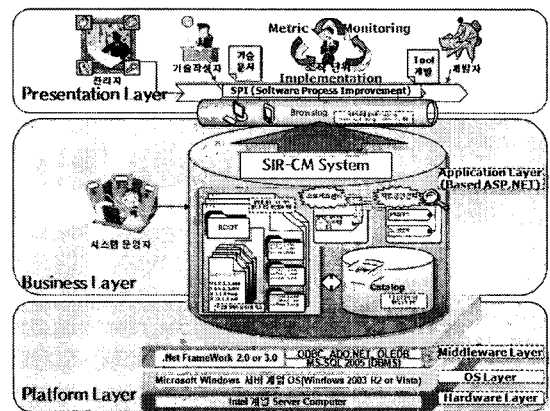


그림 4. SIR-CM 시스템 구조

각 도구에 서비스한다. 소프트웨어 프로세스 개선 지원 도구의 산출물로 CMMI의 지침서와 각 프로세스 별 측정데이터와 기타 방법론 및 도구를 포함한다. SIR-CM에 접근할 수 있는 유저로는 조직 단위, 프로젝트 관리자, 개발자, 시스템 운영자 등이 되겠다. SIR-CM에 저장되는 데이터는 형상 관리를 통해 과거정보와 새로운 정보를 저장 하고 있으며 사용자는 언제든지 브라우저로 다양한 형태의 서비스를 제공받는다. 이러한 SIR-CM은 기본적인 x86계열의 서버 플랫폼과 Microsoft사의 Windows Server를 바탕으로 .Net Framework 기반의 언어로 개발한다. 이를 바탕으로 SIR-CM 시스템을 계층적으로 표현하면 그림 5와 같다.

3.2 SIR-CM 분석

SIR-CM은 하나의 조직이나 프로젝트 단위의 프로세스를 SPIC에서 만들어진 프로세스 개선활동을 돕는 도구들과 연계될 수 있도록 지원하는 도구이다. 소프트웨어 프로세스 개선활동에 있어서 개발자 및 해당 프로세스 영역의 사람에게 필요한 지침서나 자산을 단순히 저장하는 것에 그치지 않고, 사용자가 요구하는 데이터를 재가공 하여 그래프나 차트 등의 도식적인 형태로 보고하거나 사용자가 필요로 하는 연관된 프로세스를 나타내 준다. 이 때 프로세스 영역별로 나오는 산출물

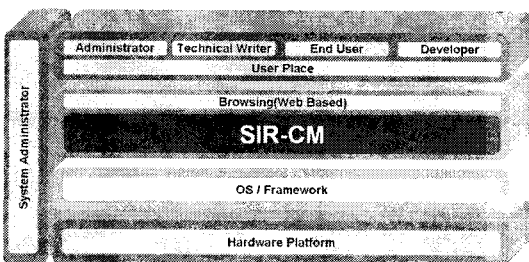


그림 5. SIR-CM 계층도

들은 표준화하여 저장하고, 또한 관련된 지침서들도 저장 한다. CMMI 지원도구들 간 데이터 연계를 바탕으로 해당 도구에서 사용하는 데이터 타입으로 사용할 수 있도록 자동화 하는 기능을 가진다. 기존에 개발된 도구들의 데이터를 수집, 재가공(XML형태 구조화 된 문서)하여 산출되는 매트릭 값을 저장하고 현재 프로세스에서 필요한 목표치를 확인하여 평가하고 개선한다. 산출되는 데이터는 버전관리 및 작업 공간 관리를 통해 저장소의 데이터 무결성을 보장한다.

다음 그림 6에서 SIR-CM의 처리절차에 대한 개념을 설명하고 있다. SPIC에서 만들어진 도구들의 표준화 되지 않은 형태의 여러 데이터를 XML이라는 문서표준규격을 통해 서버로 송신하게 되는데 이때 API나 미들웨어와 같은 문서변환 모듈을 통해 표준화 한다. 표준화된 산출물들은 SIR-CM에서 저장, 관리되며 다른 도구나 사용자의 요청이 있을 경우 필요한 매트릭이나 지침서등의 산출물을 해당도구에 맞도록 다시 변환하여 도구로 송신하는 절차를 가진다.

본 논문에서는 다양한 SPIC 지원도구를 세 가지로 분류하여 표기한다. 측정값을 저장하는 측정 도구, 지침서와 같은 문서형태로 나오는 산출물을 위한 도구, 프로세스 수행 시 자동화된 처리를 수행하는 도구이다. SIR-CM은 다음과 같은 기능을 포함한다.

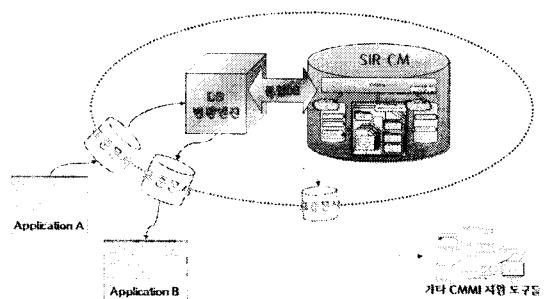


그림 6. SIR-CM의 데이터 흐름

- 표준화된 문서규격을 지원 (XML, html)
- 기존 개발된 도구와 표준화 된 방법으로 통신 (SOAP, WSDL)
- 기존에 만들어진 도구는 많은 수정 작업 없이 재사용하여 SIR-CM과 연동
- 프로세스의 영역별 산출되는 측정값은 측정을 위한 도구, 자동화를 지원하는 도구, 혹은 지침서 등을 저장
- 각종 산출물에 대한 메타정보 관리
- 프로세스 영역별 혹은 수준별로 분류를 통해 지침서, 측정값을 확인
- 프로젝트 관리자 혹은 개발자는 접근영역에 따라 다양한 Reporting 기능과 Searching기능을 제공
- 각 도구들에서 메시지 형태의 데이터가 송수신 되었을 때 그 데이터의 신뢰성을 보장
- 수행하는 조직단위나 프로젝트 단위로 구성
- 측정된 메트릭 값의 이전 결과도 저장 되어 필요할 경우 확인 가능
- 측정된 메트릭 값을 그래프, 뷰 등을 통해 Reporting 기능 제공

그림 7은 위의 요구사항 명세를 유스케이스 다이어그램으로 나타낸 것이고, 표 1은 이 유스케이스 다이어그램에 대한 구체적인 설명을 나타낸다.

3.3 SIR-CM 메타데이터

CMMI 모델에 따라 SIR-CM에 저장하는 산출

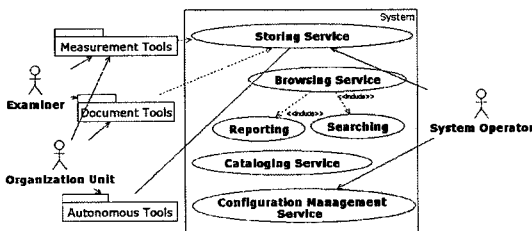


그림 7. 유스케이스 다이어그램

표 1. SIR-CM의 유스케이스

Actor & Usecase	Description
User	관리자, 기술 작성자 및 개발자들의 프로젝트에 관련된 모든 man-power
Storing Service	사용자로부터 들어오는 모든 데이터를 저장하는 서비스 제공
Browsing Service	사용자가 원하는 데이터에 대한 searching과 reporting등의 서비스 제공
Cataloging Service	사용자가 CMMI의 해당하는 PA를 확인하거나 메트릭 측정 요소들의 분류에 대한 레벨 별 분류
Configuration Management Service	저장된 데이터들의 형상관리 서비스 담당
Measurement Tools	CMMI를 통해 프로세스 개선활동 중 측정기법(ex: GQM, PSM)을 통한 메트릭 측정 도구들로 본 논문에서는 개념적인 도구로 도입
Document Tools	CMMI를 통해 프로세스개선을 수행하면서 나오는 문서 산출물들을 관리해주는 도구들로 본 논문에서는 개념적인 도구로 도입
Autonomous Tools	CMMI 프로세스 개선업무에 불필요한 사람의 공수를 제거하기 위한 도구들로 본 논문에서는 개념적인 도구로 도입

물은 크게 지침서, 22개의 프로세스 영역에 따른 측정값, 지원 도구와 도구내부 데이터가 있다. 표 2는 이 데이터에 대한 메타데이터를 표현한다.

SIR-CM의 Meta-data는 기본적으로 CMMI 지침서를 프로세스 영역별로 가진다. CMMI에서는 연속적 혹은 단계적으로 표현할 수 있지만 SIR-CM에서는 지원도구들의 산출물이 속하는 프로세스 영역에 대해서만 명시하므로 구분하지 않는다. CMMI를 적용 시킬 수 있는 Organization

표 2. SIR-CM 메타데이터

메타데이터 카테고리	메타데이터 이름
ORGANIZATION	ORGANIZATION_ID
	ORGANIZATION_NAME
	CMMI_LEVEL
PROJECT	PROJECT_ID
	PROJECT_NAME
	CMMI_LEVEL
	RELATE_PROCESS_AREA
PROCESS	PROCESS_ID
	PROCESS_NAME
	RELATE_PROCESS_AREA
GUIDELINE	GUIDELINE_ID
	LEVEL
	PROCESS_AREA
	SPECIFIC_GOAL
	SPECIFIC_PRACTICE
	GENERAL_GOAL
	GENERAL_PRACTICE
TOOL	TOOL_ID
	TOOL_NAME
	TOOL_URL
ARTIFACT	ARTIFACT_ID
	ARTIFACT_NAME
	PROCESS_AREA
	METRIC

Unit(OU)은 조직, 프로젝트, 프로세스 3가지로 구성이 이루어진다. 한 OU가 개선활동을 할 때 각 도구들 사이의 산출물의 트랜잭션은 SIR-CM에 의해 활용된다. 또한 OU와 산출물, 그리고 추가되는 도구들을 모아 발전된 지침서를 제시한다. CMMI는 한 번 인증 받았다고 해서 영원히 지속되는 것이 아니기 때문에 SIR-CM을 사용함으로써

추후에 이전데이터를 통해 비교를 할 경우나 개선점을 파악하기 위한 자산으로써 활용가치가 높다고 할 수 있다.

3.4 SIR-CM 설계

SIR-CM은 크게 네 개의 컴포넌트의 단위로 구성된다. 각 컴포넌트의 단위는 Configuration Management와 Storing Service를 제외한 나머지 컴포넌트는 인터페이스를 통해 직접적인 연결을 최대한 배제하여 설계한다. 이는 추후 유지보수와 재사용성을 향상시키기 위함이다. 이를 UML의 패키지 다이어그램으로 나타낸 것은 그림 8과 같다. 전체 프로그램이 하나의 패키지를 구성하고 있으며 각각의 기능을 컴포넌트단위로 설계하였다. Browsing Service와 Cataloging Service, Storing Service는 모두 직접적으로 연결되지 않고 인터페이스로만 연결되며 실제 데이터 이동도 메시지 단위로 이루어진다.

그림 9는 패키지 다이어그램에 대해 상세한 클래스 다이어그램을 UML로 표현하였다. Searching Service와 Reporting Service 두 가지의 클래스로 Browsing Service가 나뉜 것을 제외하고는

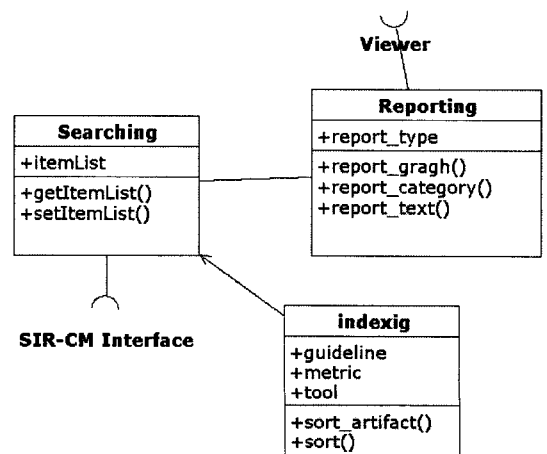


그림 8. 클래스 다이어그램

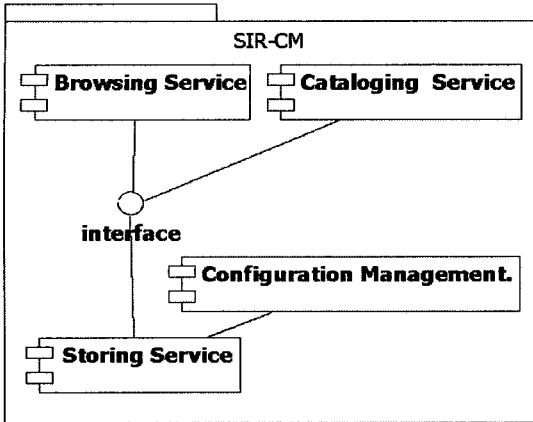


그림 9. 패키지 다이어그램

패키지 다이어그램의 컴포넌트와 동일하다고 할 수 있다. 외부에서 읽어지는 데이터는 모두 뷰(view)라는 인터페이스를 통해 사용자에게 표현된다. 내부에서 데이터가 처리되는 것도 Storage와 Configuration Management를 제외하고는 인터페이스를 통한 연결로 상호 직접적인 메서드의 조작 등은 지원하지 않도록 설계하였다. SIR-CM 시스템은 앞서 언급되었던 기존 SPIC의 개발도구들과 메시지를 통하여 송신하고 그에 대한 응답을 주고받는 형태로 구현된다. 그림 10은 위의 분석 및 설계의 결과를 토대로 작성한 SIR-CM 브라우저이다.

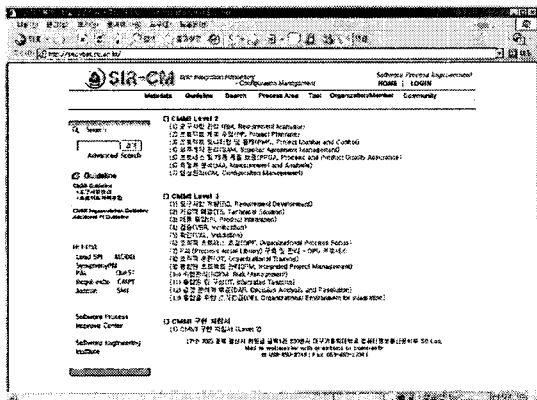


그림 10. SIR-CM 브라우저

4. 결론 및 향후연구

소프트웨어의 품질향상은 소프트웨어 프로세스의 개선활동을 통해 얻을 수 있다. 세계 산업에서 사실상 표준이 되어가고 있는 소프트웨어 프로세스 개선 모델인 CMMI는 국내에서도 많은 IT 기업들이 관심을 보이고 인증을 받으려 하고 있다. 하지만 국내 소규모 중소기업들은 대기업과 달리 쉽게 소프트웨어 프로세스 개선활동에 접근하지 못하고 있다.

따라서 본 논문은 소규모 중소기업들의 소프트웨어 프로세스 개선의 어려움을 CMMI기반 프로세스 지원 도구들의 이질적인 산출물들을 통합하여 저장, 관리할 수 있는 저장소인 SIR-CM 시스템을 제안했다. CMMI기반 프로세스 지원 도구들 사이의 연계성과 효율적인 관리를 위해 저장소에 CM과 SOA를 도입하였고, 이에 SIR-CM은 각 도구들의 산출물을 표준화된 데이터로 통합하여 관리한다. 이는 처음 소프트웨어 프로세스 개선을 위해 CMMI의 인증을 받으려는 기업에게 충분히 필요한 시스템이라 판단되고 더 나은 소프트웨어 품질 향상을 가져다 줄 것이다.

향후 본 논문에서의 분석 설계를 바탕으로 시스템을 개발하고, 이를 실제 소규모 중소기업에 도입하여 피드백을 통한 기능 개선에 관한 연구가 필요하다.

참고 문헌

[1] 이민재, "SW국제기준 CMMI 현주소," http://www.dt.co.kr/dt_txt_see.htm?article_no=2007031402011060600001, 2007.

[2] 이민재, CMMI의 이해 : 프로세스 개선을 위한 접근 방법, Pearson Education Korea, 2006.

[3] Carnegie Mellon SEI site, <http://www.sei.cmu.edu/cmmi/cmmi.html>.

- [4] 김유리, 한혁수, “CMMI 기반의 측정 저장소 구현”, 한국정보처리학회 추계학술발표논문집, 제 11권 제2호, pp. 303-306, 2004.
- [5] Anne Mette Janassen Hass, “Configuration Management Principles and Practice,” pp. 3-27, Addison Wesley, 2002.
- [6] Assembling configuration Management Environments (for Software Development) site, http://www.cmcrossroads.com/bradapp/acme/scm-defs.html#SEI_Software_CMM.



김 행 곤

- 1985년 중앙대학교 전자계산학과(공학사)
 - 1987년 중앙대학교 대학원(공학석사)
 - 1991년 중앙대학교 대학원(공학박사)
 - 1978년~1979년 미 NASA 연구원
 - 1987년~1989년 미 Bell Lab. 연구원
 - 1979년~1983년 공군방공포병학교 교관
 - 2000년~2002년 미 CMU 교환교수
 - 1990년~현재 대구가톨릭대학교 교수
 - 관심분야 : CBSE, Web 서버 구현, 소프트웨어 아키텍처, 소프트웨어 제공학 및 역공학 등
-
-



박 은 주

- 1999년 대구가톨릭대학교 전자계산학과(이학사)
 - 2001년 대구가톨릭대학교 전자계산학과(이학석사)
 - 2005년~현재 대구가톨릭대학교 컴퓨터정보통신공학과 박사과정
 - 관심분야 : 컴포넌트 기반 소프트웨어(CBD), 소프트웨어 프로세스, 형상관리 등
-
-