

프로덕트 라인에서 공통성과 가변성 분석을 통한 요구사항 관리방법

박 동 수^{*} · 김 동 규^{**} · 정 기 원^{***}

요 약

프로덕트 라인 공학(Product Line Engineering)에서 핵심자산은 제품계열에 속하는 패밀리 멤버들이 어플리케이션을 만드는 데 기초가 되는 모든 자산을 포함한다. 핵심자산 가운데 하나인 요구사항은 전통적인 시스템 개발에서와 마찬가지로 모든 핵심자산의 기초가 되는 부분이며, 다른 핵심자산의 공통성과 가변성(Commonality and Variability : C&V)을 결정하는 중요한 요소가 된다. 하지만 요구사항은 도메인 전문가 혹은 개발자가 경험에 기반하여 정형적인 프로세스 없이 분석하고 있으며, 동일한 도메인 요구사항임에도 불구하고 요구사항이 재사용되지 않고 있다. 따라서 도메인 요구사항은 객관적인 방법을 통해 공통성과 가변성을 도출하고 이를 재사용할 수 있는 프로세스가 정립되어야 할 필요가 있다. 이에 본 논문에서는 도메인 요구사항을 객관적인 근거에 따라 분석할 수 있는 4가지 활동 즉 도메인 범위 결정, 도메인 요구사항 추출 및 일반화, 도메인 요구사항 분석 및 모델링, 도메인 요구사항 변경 관리 활동 및 그에 따른 세부적인 활동을 제시하였다. 이를 요구사항 관리에 적용할 경우 분석한 도메인 요구사항과 관련된 아키텍처 혹은 컴포넌트의 재사용성 증대를 통하여 제품 개발기간과 비용을 줄이며 생산성을 향상시키는 등의 장점을 가져올 수 있을 것으로 기대한다. 또한 프로덕트 라인 공학에서 핵심자산의 기초가 되는 요구사항을 체계적으로 분석하고 관리함으로써, 요구사항을 기반으로 산출되는 다양한 산출물의 품질을 높일 수 있을 것으로 기대한다.

키워드 : 요구사항, 프로덕트 라인 공학

A Method to Manage Requirements Analyzing the Commonality and Variability in Product Line

Dongsu Park^{*} · Dongkyu Kim^{**} · Kiwon Chong^{***}

ABSTRACT

The core assets include all properties which consist of an application in Product Line Engineering. The requirement, one of the core assets, is a basis of other core assets and commonality and variability of other core assets are classified by the requirement. accordingly, commonality and variability of the domain requirement should be managed objectively and it is necessary to make a process to reuse the domain requirements. However the requirement is analyzed by domain experts or developers without proper process. In this paper, we proposed the 4 activities: (1)the domain scoping, (2)the extraction and generalization of the domain requirement, (3)the domain requirement analyzing and modeling, (4)the change management, and sub activities. For all reasons given previously, it is possible to reduce the development time and cost by reusing the architectures and components related to the domain requirement. In addition, it is possible to increase the quality of the artifacts produced based on the requirements by managing them systematically.

Key Words : Requirement, Product Line Engineering(PLE)

1. 서 론

프로덕트 라인 공학은 빠르게 변화하는 시장에서 성공적으로 소프트웨어를 개발하기 위한 하나의 방법론이다. 프로덕트 라인 공학의 목적은 유사한 시스템들의 공통성을 이해

하고 제어하여 핵심자산을 만들며[1], 변경되는 부분을 구별하여 개발함으로써 체계적으로 소프트웨어 프로덕트 패밀리를 개발할 수 있도록 지원하는 것이다. 프로덕트 라인 공학은 제품 계열 내의 핵심자산을 구축하는 도메인 공학과 핵심자산으로부터 개별 제품을 생산하는 어플리케이션 공학으로 구분할 수 있다[2]. 도메인 공학 프로세스에서 산출되는 핵심자산은 요구사항, 아키텍처, 컴포넌트뿐만 아니라 테스트 계획, 개발문서, 구현코드 등이 있다[2]. 이 가운데 요구사항은 전통적인 시스템 개발에서와 마찬가지로 모든 개발

* 본 연구는 숭실대학교 교내연구비 지원으로 이루어졌음

† 정 회 원 : 숭실대학교 정보과학대학원

** 정 회 원 : 숭실대학교 대학원 컴퓨터학과 박사과정

*** 종신회원 : 숭실대학교 컴퓨터학부 교수

논문접수 : 2006년 3월 20일, 심사완료 : 2006년 10월 16일

에 기초가 되는 사항이며 다른 핵심자산의 공통성과 가변성을 결정하는 기준이 된다[3].

공통성과 가변성 분석에 관한 기존 연구에서는 객관적인 분석이 아닌 개발자의 직관이나 도메인 전문가의 경험에 의존하고 있다[14]. 그러나 요구사항의 공통성과 가변성은 개발과정에서 산출되는 다른 핵심자산들의 공통성과 가변성에 영향을 미치기 때문에 객관적 기준에 의해 판단되어야 한다. 또한 요구공학 측면에서 살펴보면, 요구공학 프로세스 자체에 C&V를 고려한 요구공학 프로세스를 제시하지 못하고 있는 실정이다. 따라서 본 논문에서는 소프트웨어 공학 분야에서 예측과 추정을 위한 가장 신뢰받는 자료 중에 하나인 과거의 유사한 경험을 모형화하는 방법을 통해 프로덕트 라인 공학의 핵심자산 가운데 하나인 도메인 요구사항을 효율적이며, 객관적으로 관리하는 방법을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서는 도메인 공학, 도메인 분석기법 등에 대해 알아보며, 3장에서는 도메인 요구사항 관리 활동들을 제시한다. 4장에서는 사례연구로 도메인 요구사항 관리 활동을 바탕으로 인사관리 도메인 중의 하나인 조직관리 도메인에 적용한다. 5장에서는 결론 및 평가에 관하여 논의한다.

2. 도메인 공학

도메인에 대한 정의는 해당 영역에 대한 지식에 중점을 두는 것과 해당 영역의 어플리케이션에 중점을 두는 것으로 구분할 수 있다. 전자는 도메인을 지식분야, 일련의 개념 또는 현장 종사자들이 이해하고 있는 용어에 의해 특성화되는 행위로 정의하며[10], 후자는 어플리케이션 관점에서 도메인의 공통된 특성을 공유하는 현재와 미래의 어플리케이션의 집합이라고 정의한다[11]. 본 논문에서는 도메인 영역을 정의하는 단계에서는 전자의 개념을 사용하였으며, 나머지 단계에서는 후자의 개념을 사용하여 전개했다. 즉, 도메인 영역 정의단계를 제외한 나머지 단계에서는 도메인이란 핵심적인 공통성을 지닌 것으로 인식하는 것과 자체적으로 변화할 수 있는 가변성을 식별하고 예측, 계획할 수 있어야 한다는 것을 의미한다. 따라서 도메인 요구사항은 시스템들의 공통적인 골격에 대한 요구사항들과 함께 시스템 마다 다소 상의하게 나타나는 부분들을 의미한다.

2.1 도메인 분석기법

해당 도메인 내의 공통성과 가변성을 분석하기 위해 도메인 분석 기법이 사용되고 있다. 도메인 분석의 대표적인 방법으로는 휘쳐 지향 도메인 분석(FODA: Feature Oriented Domain Analysis) 방법을 꼽을 수 있다. 이는 휘쳐 모델을 이용하여 시스템이 제공해야 할 휘쳐들을 식별함으로써 도메인을 분석하는 방법이다[4]. 휘쳐 모델은 도메인 영역 분석의 핵심인 공통성과 가변성을 분석하는데 매우 유용하지만, 하나의 휘쳐를 추출하기 위해서 필요한 요구사항 분석의 세부적인 활동은 언급하지 않고 있다. 다만 휘쳐 모델링

과 동시에 도메인 요구사항 분석활동이 있어야 하며 이 활동의 결과로 유스케이스 모델, 분석객체 모델, 품질요소 등이 분석되어야 한다고 언급하고 있다[5].

또한 휘쳐 모델링 방법에 재사용 중심 소프트웨어 공학 업무(RSEB: Reuse-Driven Software Engineering Business)의 프로세스를 통합한 FeatureRSEB 방법이 있다[6]. 이 방법은 휘쳐 모델을 만드는 활동과 병행하여 유스케이스 모델을 만드는 작업을 한다. 또 다른 방법으로 UML 확장기법을 이용하여 가변성을 모델링 한 연구가 있다[7]. 여기서는 UML 모델에 스테레오타입으로 가변점 즉 《variationPoint》을 기술하고 각 가변값들을 상속관계로 표현한다. 본 논문에서는 이러한 가변점을 PR (Primitive Requirement) 명세서에 기재하여 PR 분석모델에서 가변점들을 스테레오타입으로 도식화 하였다.

이처럼 도메인을 분석하는 분야에서는 휘쳐를 이용한 분석이 주를 이루었다. 그러나 앞에서 언급한 바와 같이 휘쳐를 결정하는 객관적인 근거 즉 요구사항 분석에 대한 세부적인 활동들은 제시하지 않고 있다. 휘쳐를 결정하기 위한 작업과 동시에 요구사항 분석이 이루어져야 한다고 언급할 뿐이다.

2.2 도메인 요구공학과 요구공학

요구사항은 모든 소프트웨어 계획 수립 및 소프트웨어 개발 활동의 기초를 제공한다. 소프트웨어 개발에서는 요구공학이라는 용어를 사용하면서 요구사항에 관련된 모든 활동들 즉, 요구사항 추출, 분석, 기술 및 관리의 공학적 접근 방법을 체계적으로 정리한 것이다. 요구공학은 요구사항 추출, 요구사항 분석, 요구사항 명세, 요구사항 검증의 4 가지 활동으로 구성이 되는데, 프로덕트 라인 공학에서 말하는 요구사항의 공통성과 가변성은 각 활동들 마다 고려하지 않고 있다. 요구공학의 4 가지 활동들이 프로덕트 라인에서 수행되려면 각 활동마다 C&V를 고려해야 한다. 다음의 <표 1>은 요구공학 기본 활동들을 언급하고 각 활동들을 도메인 요구공학에서 초점을 두어야 할 점을 나타낸 것이다 [13].

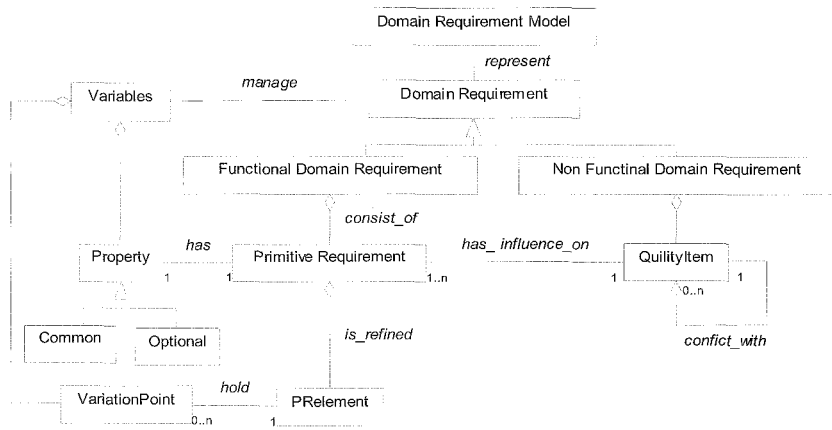
2.3 도메인 요구사항 메타모델

본 논문에서는 도메인 요구사항 관리에서 각 모델 요소들 사이의 추적가능성을 보장하기 위해 메타모델을 제시한다. (그림 1)은 도메인 요구사항 메타모델을 나타낸 것이다.

요구사항은 두 가지로 분류할 수 있다[12]. 하나는 기능적인 요구사항으로 시스템이 수행하는 기능을 나타내고, 다른 하나는 비기능적 요구사항으로 가능성을 달성하는데 부가되는 제약조건 및 품질특성을 나타낸다. 도메인 요구사항도 기능적 요구사항과 비기능적 요구사항으로 구분된다. 기능적 요구사항은 PR을 가지는데, 여기서 PR은 해당 도메인 내의 범위를 분석하여 의미상의 최소단위의 요구사항으로 정의되며, 하나의 PR은 여러 개의 PR요소들로 구성된다. 도메인 요구사항은 가변성을 가지는데, 이는 상세수준에 따라

<표 1> 요구공학과 도메인 요구공학의 주요 활동

| | 요구공학 | 도메인 요구공학 |
|---|--|--|
| 요구사항 추출 (Eliciting Requirements) | 요구사항 추출은 고객과 시스템 사용자, 그 외 시스템 개발에 관심을 가진 자들과 의사소통함으로써 요구사항으로 발견하는 과정 | 도메인 분석 기법을 이용하여 예상되는 가변성을 명확히 추출하고 그 범위를 확정하는데 초점 |
| 요구사항 분석 (Analyzing Requirements) | 초기 추출된 요구사항들에 대한 갈등, 중복, 누락, 불일치성 등을 분석, 정제 | C&V를 식별한다. 가변성이 일어날 수 있는 부분을 지적 |
| 요구사항 명세 기술 (Developing Requirements Specification) | 다른 시스템 관심자들 사이에 요구사항들을 효율적으로 전달하기 위해 각각의 요구사항들이 서로 모호하지 않고 간결하고 이해력 있게 작성하는 과정 | 가변될 수 있는 부분은 특정 프로덕트에 따라 확정되고, 인스턴스 되며 채워질 수 있도록 상징적인 대체자를 명세서에 포함 |
| 요구사항 검증 (Validating Requirements) | 요구사항의 완전성, 정확성, 일관성, 명백성을 보증하는 과정 | 프로덕트 라인 개발 시의 요구사항 뿐만 아니라 특정 시스템 개발 시의 요구사항에 대해서도 검증이 실행되어야 함 |



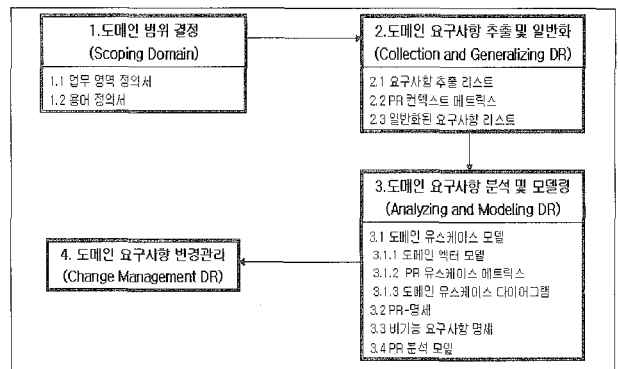
(그림 1) 도메인 요구사항 메타모델

속성과 가변성 지점으로 구분된다. 속성은 해당 도메인의 시스템에 나타나는 빈도수에 따라서 공통성과 선택성으로 나타나며, 하위레벨로 내려가서는 가변지점을 식별해 낼 수 있다. 비기능적 요구사항은 여러 개의 품질특성으로 이루어진다. 일반적으로 품질요소는 하나 이상의 기능적 요구사항과 관련된다[12]. 또한 동시에 다른 품질요소와 같이 존재함으로써 상호 영향을 받는다. 그러므로 비기능적인 요구사항을 구성하는 품질특성은 기능적 요구사항과 다른 품질특성과의 연관관계를 가지고 있다.

3. 도메인 요구사항 관리

도메인 요구사항을 관리하기 위한 활동은 도메인 범위 결정, 도메인 요구사항 추출 및 일반화, 도메인 요구사항 분석 및 모델링, 도메인 요구사항 변경관리로 이루어지며 각 활동마다 세부 프로세스를 가진다.

(그림 2)와 같이 도메인 요구사항 관리는 4개의 주요활동들과 각 활동별 세부 활동이 존재하며, 세부 활동 프로세스를 통해 산출물이 나온다. 도메인 요구사항 관리에서는 일반적인 요구사항 관리와 달리 객관적인 방법으로 공통성과 가변성을 분석해야한다. 이를 위해서는 도메인 범위를 결정

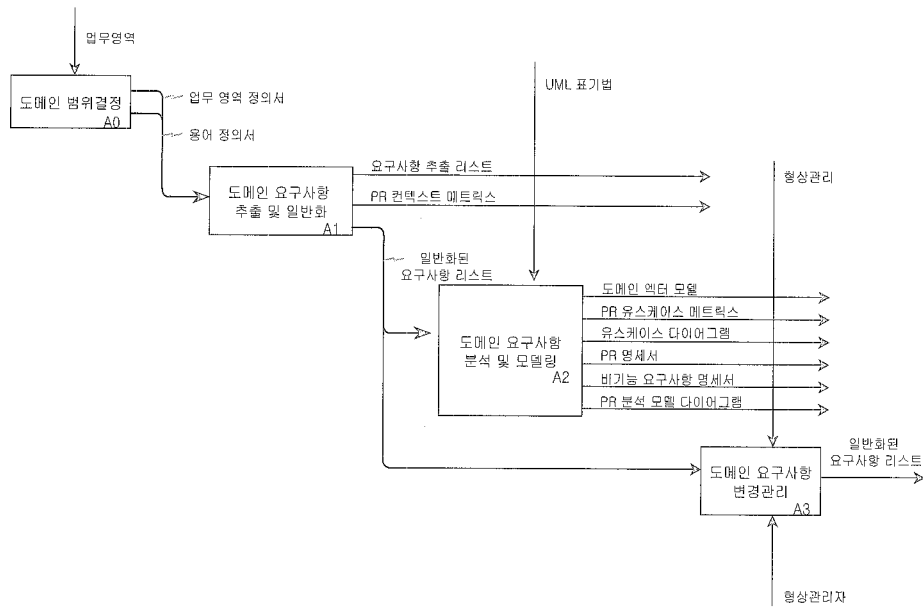


(그림 2) 도메인 요구사항 관리 프로세스

하고, 요구사항을 추출하여 PR 컨텍스트 매트릭스를 통해 대체할 수 있는 요구사항을 일반화시킴으로써 객관적으로 공통성과 가변성을 분석 할 수 있다. (그림 3)은 IDEF0 (Integrated Definition) 모델을 이용하여 도메인 요구사항 관리 프로세스를 나타낸 것이다.

3.1 도메인 범위 결정

도메인은 시스템들의 집합 또는 한 시스템의 유지보수성, 이해도, 사용성, 재사용성 등의 특징을 이용한 기능 영역으



(그림 3) IDEF0 모델을 이용한 도메인 요구사항 관리 프로세스

로 정의된다[8]. 본 논문에서는 각각의 시스템의 업무영역을 정의하는 것과 업무영역에서 산출되는 용어가 해당 시스템과 일치한다는 것으로 도메인 범위를 정하였다.

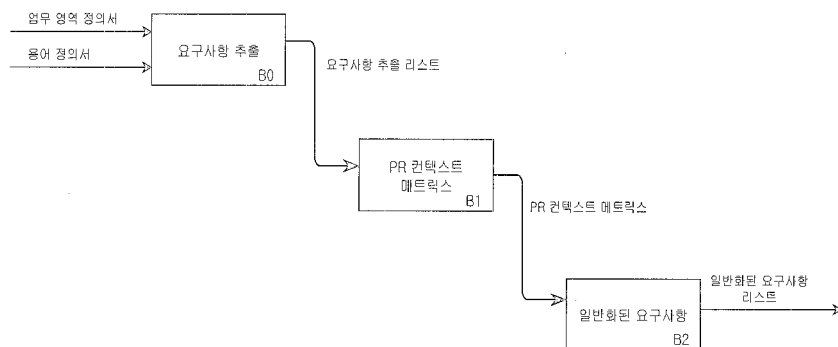
업무영역을 정의하기 위해서는 도메인 전문가 또는 해당 관리부서의 담당자를 통해 정의하는 것이 바람직하다. 그렇지 못할 경우에는 관련된 시스템을 분석하거나 레거시 시스템을 분석하는 것이 좋은 방법이 되며, 해당 도메인 관련서적이나 자료를 찾아서 분석하는 방법도 효과적이다. 업무영역에서 산출되는 용어가 해당 도메인과 일치하는 것을 알 수 있는 방법은 용어정의서를 작성하는 것이다. 용어정의서에는 도메인에서 사용하는 기본 개념에 대하여 정의하고 용어를 정리하며 용어간의 관련성을 기술한다. 여기서 산출되는 업무범위정의서와 용어정의서가 도메인 요구사항 추출 및 일반화 단계의 입력물이 된다.

3.2 도메인 요구사항 추출 및 일반화

도메인 요구사항을 추출하기 위해서는 해당 도메인 범위에 속하는 기존 시스템 혹은 차후 개발될 시스템을 분석하는 것이 바람직하다. 만약 기존 시스템이 존재하지 않을 경

우에는 유사한 도메인에 속하는 시스템을 분석하는 것이 유용하다[3]. 도메인 요구사항을 분석하는 것은 어떤 특정 고객만을 위한 것이 아니라 해당 도메인 범위에 속하는 시스템을 원하는 임의의 고객의 요구까지도 만족할 수 있도록 분석되어야 하기 때문이다. 그러므로 도메인 요구사항은 도메인 전문가와 개발자의 직관이 아닌 객관적인 근거를 통해 공통성과 가변성을 추출할 수 있어야 한다.

따라서 본 논문에서는 도메인 요구사항 추출과 일반화 단계를 제공함으로써 요구사항을 좀 더 객관적으로 공통성과 가변성으로 분석할 수 있도록 하며, 이 단계에서 산출되는 일반화된 요구사항 리스트를 통해 요구사항 추적성을 보장하게 된다. 일반화된 요구사항 리스트는 요구사항 추출리스트와 PR 컨텍스트 매트릭스를 바탕으로 산출되고, 일반화된 요구사항 리스트를 통해 다음 단계인 도메인 요구사항 분석 및 모델링단계에서 중요한 입력으로 사용되기 때문이다. 요구사항 변경이 일어날 경우 일반화된 요구사항 리스트에 기재되어 있는 PR ID를 통해 각 산출물에 작성된 PR을 손쉽게 추적할 수 있다. (그림 4)는 도메인 요구사항 추출 및 일반화 프로세스를 나타낸다. 이것은 요구사항 추출,



(그림 4) 도메인 요구사항 추출 및 일반화 프로세스

PR 컨텍스트 매트릭스, 일반화된 요구사항의 3단계로 이루어져 있으며, 각각 요구사항 추출 리스트, PR 컨텍스트 매트릭스, 일반화된 요구사항 리스트가 산출된다.

3.2.1 요구사항 추출 리스트

도메인 요구사항을 추출할 때 기존 시스템 혹은 차후에 개발할 시스템을 분석하고, 만약 기존 시스템이나 레거시 시스템이 없을 경우에는 비슷한 도메인을 가진 시스템들을 분석하여 리스트를 작성해야 한다. 도메인 요구사항은 어느 특정 시스템의 요구사항이 아니라 해당 도메인과 관련된 시스템들의 공통성과 가변성을 분석해야 하기 때문이다. 요구사항 추출 리스트를 기반으로 하여 PR 컨텍스트 매트릭스를 작성하는데, 이는 객관적으로 요구사항을 분석할 수 있는 근거를 가지게 된다. <표 2>는 요구사항 추출 리스트의 예를 나타낸 것이다.

추출된 요구사항 리스트를 사용하여 PR 컨텍스트 매트릭스를 작성한다. 이는 각 도메인의 요구사항의 공통성과 가변성을 객관적으로 분석하는데 중요한 산출물이 된다. PR 컨텍스트 매트릭스는 요구사항 추출 리스트를 바탕으로 (그림 5)와 같이 매트릭스 형태로 병합한다. 요구사항 추출 리스트에 추출된 요구사항들이 행으로 작성되고, 기존의 시스템들과 레거시 시스템들은 열로 작성된다. 행과 열이 만나는 곳에 해당 시스템이 PR을 가지고 있으면 'O'를 표시하

<표 2> 요구사항 추출 리스트 예

| 요구사항 ID | 요구사항 | 요구사항 설명 |
|---------|---------|-------------------------|
| PR 1 | Login | 사용자는 인증을 통해 시스템에 접속함 |
| PR 2 | Logout | 사용자 로그아웃 |
| PR 3 | 조직신설 | 새로운 조직을 신설함 |
| PR 4 | 조직발령 | 조직명 변경, 조직 폐지 등의 기능을 가짐 |
| PR 5 | 과거조직 관리 | 폐지된 조직을 관리한다. |
| | | |

고, 해당 시스템이 PR을 가지고 있지 않을 경우에는 'X'를 표시하여 PR의 속성과 비율을 결정한다. 이러한 PR 컨텍스트 매트릭스를 통해 통계적으로 PR이 어떠한 속성을 가지고 있으며, 그 속성의 비율을 결정한다. 다음은 PR 속성의 비율을 구하는 식이다.

$$PR속성비율 = \frac{\text{행}(PR)\text{과열}(시스템)\text{이일치하는수}}{\text{해당도메인시스템수}} * 100$$

위의 식을 바탕으로 (그림 5)와 같이 PR 컨텍스트 매트릭스를 작성한다.

<표 3>은 4가지 PR 속성에 대한 설명을 나타낸 것이다.

<표 3> PR 속성에 대한 설명

| PR 속성 | 설명 |
|----------------------------|--|
| C(Common) | 서로 대체할 수 있는 요구사항이 없으며, 행과 열이 모두 일치하는 PR일 경우 |
| P(Optional) | 서로 대체할 수 있는 요구사항이 없으며, 행과 열이 일치하지 않은 PR이 있는 경우 |
| CV(Common with Variable) | 서로 대체할 수 있는 요구사항이 있으며, 그 요구사항이 시스템에 모두 존재할 경우 |
| PV(Optional with Variable) | 서로 대체할 수 있는 요구사항이 있으며, 그 요구사항 중 시스템에 존재하지 않은 요구사항이 있을 경우 |

위의 PR 속성에 대한 설명으로 볼 때 PR1과 PR2는 모두 'O' 이므로 공통성을 지닌 PR이 된다. PR3은 PR3i1과 PR3i2를 일반화시킨 PR이다. PR3i1은 행과 열이 일치하지 않는 경우가 두 곳이 있으며, PR3i2는 한곳이 있다. 이때 PR3i1과 PR3i2는 서로 대체할 수 있는 PR이라고 할 수 있다. PR3의 속성은 가변성을 지닌 선택적인 PR로 작성되는데, 이는 PR3i1과 PR3i2의 행, 열을 합집합으로 보고 두 개의 요구사항 중 하나라도 행과 열이 맞으면 일반화 된 PR3

| Req \ System | Property/ Ratio | System n1 | System n2 | System n3 | System n4 |
|--------------|-----------------|-----------|-----------|-----------|-----------|
| PR 1 | C(100%) | O | O | O | O |
| PR 2 | C(100%) | O | O | O | O |
| PR 3 | PV(77%) | O | O | X | O |
| PR3i1 | | O | O | X | X |
| PR3i2 | | O | O | X | O |
| PR 4 | CV(100%) | O | O | O | O |
| PR4i1 | | O | O | O | X |
| PR4i2 | | O | X | O | O |
| PR 5 | P(77%) | O | O | X | O |
| PR 6 | C(100%) | O | O | O | O |
| | | | | | |

C : Common PR P : Optional PR CV : Common PR with Variable PV : Optional PR with Variable

(그림 5) 컨텍스트 매트릭스

에서는 행과 열이 일치한다고 보지만, PR3i1과 PR3i2 중 모두 행과 열이 일치하지 않는다면 일반화 된 PR3에서는 행과 열이 일치하지 않는다고 보기 때문에 PR3은 가변성을 지닌 선택적인 PR로 볼 수 있다. PR4 또한 일반화된 요구사항이지만 PR3과 다른 점은 공통적인 속성을 지닌 가변성 PR이다. PR4i1 과 PR4i2의 행과 열이 일치하는 것을 합집합으로 볼 수 있으므로, 두 개의 요구사항 중 모두 행과 열이 일치하는 PR이기 때문이다. PR5는 가변성 PR이라고 작성할 수 있는데, 일반화 할 수 있는 대체할 수 있는 PR이 없고, 행과 열이 일치하지 않는 부분이 한 부분이라도 있으면 가변성 PR로 작성하기 때문에 PR5는 'P'로 작성할 수 있다. 이처럼 PR 컨텍스트 매트릭스를 통해 서로 대체할 수 있는 PR은 일반화 할 수 있으며, 그렇지 않을 경우에는 행과 열이 일치하는 지점을 확인함으로써 공통성을 지닌 PR, 가변성을 지닌 PR로 작성 될 수 있다. 이러한 PR 컨텍스트 매트릭스는 도메인 전문가 또는 개발자의 주관이 아닌 객관적으로 공통적인 속성을 지닌 PR, 가변적 속성을 지닌 PR을 분석할 수 있는 근거가 된다.

3.2.2 일반화된 요구사항 리스트

PR 컨텍스트 매트릭스를 통해 대체할 수 있는 PR들은

일반화 시키고, 객관적으로 요구사항의 공통성과 가변성을 분석할 수 있는 근거를 제시하였다. 이를 바탕으로 일반화된 요구사항 리스트를 제시한다. (그림 6)은 PR 컨텍스트 매트릭스를 통해 일반화 된 요구사항을 나타낸 것이며 색깔이 칠해진 부분이 일반화가 되어야 할 PR들이다.

PR 컨텍스트 매트릭스를 통해 일반화된 요구사항 리스트를 통해 레거시 시스템의 요구사항을 추출하여 요구사항들을 일반화시킴으로서 요구사항 변경 발생 시 요구사항을 추적할 수 있는 근거가 될 수 있다.

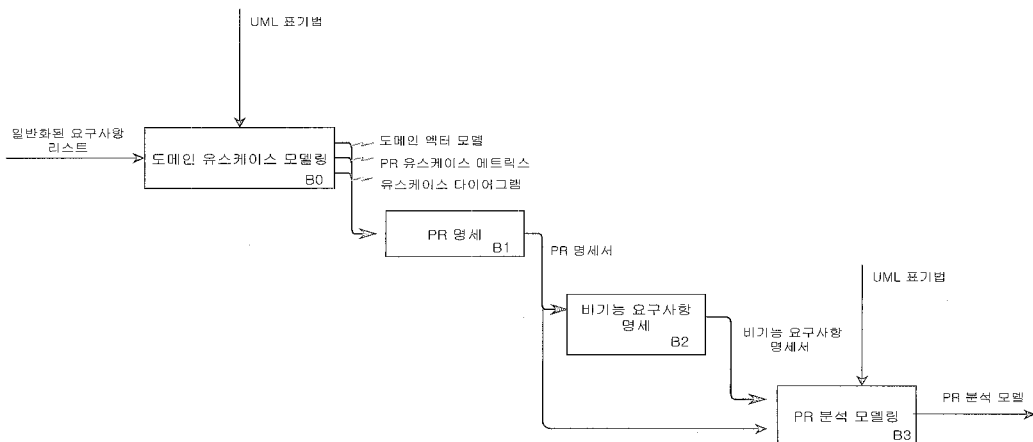
3.3 도메인 요구사항 분석 및 모델링

분석 단계에서는 분석자와 개발자, 분석자와 사용자의 원활한 의사소통을 위해 그림으로 생각을 표현하는 도구들이 활용되고 있다. 대표적인 표기법으로 UML을 꼽을 수 있다. UML은 소프트웨어 개발 과정의 산출물들을 명시, 개발, 문서화하기 위한 모델링 언어로서 OMG의 표준이다. 도메인 요구사항 추출 및 일반화 단계에서 산출되는 일반화된 요구사항 리스트를 입력받아 도메인 유스케이스 모델, PR 명세서 기술, PR 분석모델 도식의 단계로 도메인 요구사항 분석 및 모델링이 이루어진다. 이때 도메인 유스케이스 모델과 PR 분석모델 작업을 할 경우 UML 표기법을 이용하여 도식

| System Req \ Property/ Ratio | | System n1 | System n2 | System n3 | System n4 | |
|------------------------------|-----------|-----------|-----------|-----------|-----------|------------------|
| PR 1 | C(100%) | O | O | O | O | C |
| PR 2 | C(100%) | O | O | O | O | |
| PR 3 | PV (77%) | O | O | X | O | substitutive PRs |
| PR3i1 | | O | O | X | X | |
| PR3i2 | | O | O | X | O | |
| PR 4 | CV (100%) | O | O | O | O | CV |
| PR4i1 | | O | O | O | X | |
| PR4i2 | | O | X | O | O | substitutive PRs |
| PR 5 | P(77%) | O | O | X | O | |
| PR 6 | C(100%) | O | O | O | O | P |
| | | | | | | |

C : Common PR P : Optional PR CV : Common PR with Variable PV : Optional PR with Variable

(그림 6) 일반화된 PR 컨텍스트 매트릭스



(그림 7) 도메인 요구사항 분석 및 모델링 프로세스

할 것이며, 가변지점과 유스케이스 속성을 표기하기 위해 UML 확장 메커니즘인 스테레오타입을 사용할 것이다.

3.3.1 도메인 유스케이스 모델

도메인 유스케이스 모델은 요구사항을 분석하고 명세하기 위하여 유스케이스 모델링 방법을 사용하며, 해당 도메인의 시스템들의 기능과 환경에 대한 모델링을 한 것이다. 본 논문에서 도메인 유스케이스 모델이라 명하는 이유는 도메인의 공통성과 가변성을 표현한 유스케이스 모델이기 때문이다. 도메인 유스케이스 모델은 도메인 액터 모델, PR 유스케이스 모델, 유스케이스 다이어그램으로 구성된다. 그러나 도메인 속성을 표현하기 위해 기존의 UML 요소엔 한계가 있으므로 UML의 스테레오타입 확장 메커니즘을 사용하여 도식화한다. <표 4>는 도메인 유스케이스 모델 도식을 위해 필요한 스테레오타입을 나타낸 것이다.

<표 4> 도메인 유스케이스 모델에 대한 UML 스테레오타입

| 모델 | UML 요소 | 스테레오타입 |
|------------------|---------|---------------------|
| common Actor | Actor | <<common>>(default) |
| optional Actor | Actor | <<optional>> |
| common Usecase | Usecase | <<common>>(default) |
| optional Usecase | Usecase | <<optional>> |

3.3.1.1 도메인 액터 모델

액터는 시스템의 외부에 존재하면서 시스템과 직접 상호 작용하는 사람 혹은 외부시스템이 될 수 있으며, 액터를 고려할 때 중요한 것은 역할이다. 액터 모델에서는 액터 다이어그램을 도식하며, 각각의 액터에 대한 정의와 책임을 기술하는 액터 기술서를 작성한다. 액터들에 대한 관계는 일반화 관계, 집합 관계를 가질 수 있으며, 외부 시스템은 독립적으로 존재도 가능하다.

3.3.1.2 PR 유스케이스 매트릭스

PR 유스케이스 매트릭스에서는 일반화된 요구사항 리스트에서 추출된 PR들을 행으로 나열하고 열에는 요구사항 추출 리스트를 통해 얻어진 유스케이스를 기술한다. 이를 통해 각각의 유스케이스가 어떠한 PR로 이루어져있는가를

알 수 있으며, 이를 근거로 유스케이스를 재구성하도록 해 주고 유스케이스 속성을 결정할 수 있다. PR 유스케이스 매트릭스는 PR 컨텍스트 매트릭스와 마찬가지로 행은 PR 리스트로 구성되고, 열은 유스케이스로 구성되며 행과 열이 일치하면 'O', 일치하지 않으면 'X' 로 표시한다.

작성된 PR 유스케이스 매트릭스를 기반으로 다음과 같이 유스케이스를 재구성한다.

1. 유스케이스에 포함되는 PR이 공통성을 지닌 PR 경우이고 해당 유스케이스에 걸쳐 있는 경우 ((그림 8)에서 ②)는 공통적으로 겹쳐져 있는 경우로 본다. 그래서 유스케이스2와 PR2와 PR4는 <<include>> 관계로 연관시킨다.
2. 유스케이스가 선택성 PR을 포함하고 있는 경우이고 해당 유스케이스에 걸쳐 있는 경우 PR과 해당 유스케이스는 <<extend>> 관계로 연관시킨다.

위의 과정을 거쳐 재구성된 유스케이스의 속성은 다음과 같다.

- 공통성 유스케이스 : 유스케이스가 해당 도메인 내에 반드시 있어야 하는 PR 즉 공통적인 PR(C 또는 CV)을 가지고 있는 경우 공통성 유스케이스로 분류한다.
- 선택성 유스케이스 : 유스케이스가 해당 도메인 내에 반드시 존재하지 않아도 가능한 유스케이스를 나타낸다. 이는 선택적인 PR(P 또는 PV)로 구성된 유스케이스인 경우이다.

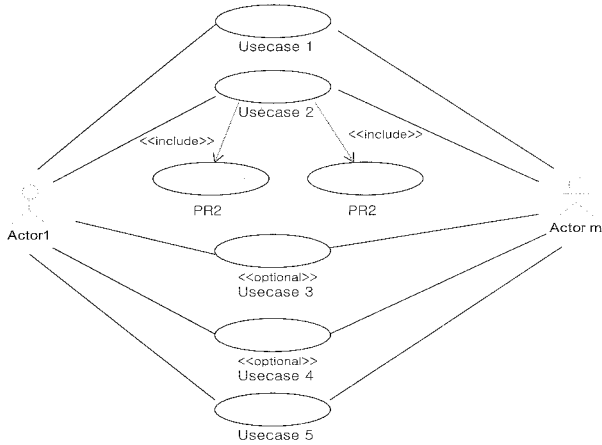
3.3.1.3 유스케이스 다이어그램

PR 유스케이스 매트릭스를 통해 유스케이스 다이어그램을 작성할 수 있다. 이는 액터와 유스케이스의 속성을 도식을 통해 확인 할 수 있으며, PR 유스케이스 매트릭스를 통해 재정의 된 유스케이스를 확인 할 수 있다.

(그림 9)에서 보는 바와 같이 유스케이스1, 유스케이스2, 유스케이스5는 공통적인 속성을 지닌 유스케이스이지만, 가변성이 포함된 PR과 일치물 이루는 유스케이스3, 유스케이스4는 선택적인 유스케이스가 된다. 또한 유스케이스2에서는 PR2와 PR4에 일치물 하고 두 개의 PR 모두 공통적인 속성을 지닌 PR이므로 <<include>>로 연관 지을 수 있다.

| Req | Property/ratio | Usecase 1 | Usecase 2 | Usecase 3 | Usecase 4 | Usecase 5 |
|-------|----------------|-----------|-----------|-----------|-----------|-----------|
| PR1 | C | O | | | | |
| PR2 | C | ① | O | | | |
| PR3 | PV | | | O | | |
| PR4 | CV | | O | ③ | | |
| PR5 | P | | ② | | O | ⑤ |
| PR6 | C | | | | ④ | O |
| | | | | | | |

(그림 8) PR 유스케이스 매트릭스



(그림 9) 유스케이스 다이어그램

3.3.2 PR 명세

도메인 유스케이스 모델이 작성되면 PR 명세서를 기술하는데, 이는 일반화된 요구사항 리스트에 있는 PR과 그 PR의 요소들과 해당 PR의 속성과 비율을 기술한다. PR은 두 가지 측면으로 구성요소들을 나열한다. 동적인 측면은 시간 흐름에 의해 기술되는 요소이며, 이것은 세부활동 중 유스케이스 명세서의 바탕이 된다. 그러나 모든 시스템이 워크플로우 지향적이지 않기 때문에 정적인 측면의 기술도 필요하다. 따라서 이를 고려하여 두 가지 측면의 PR 요소들의 기술에서 가변성 지점을 식별할 수 있다. 먼저 동적인 측면에서는 computation과 control에 대한 가변성을 식별하며, 정적인 측면에서는 data와 system interface에 대한 가변성

을 식별한다. <표 5>는 가변지점 종류와 그에 대한 설명을 나타낸 것이다.

위와 같이 가변성을 분류함으로써 도메인 요구사항을 관리할 때 나타날 수 있는 가변적인 속성에 대하여 계획적으로 대처할 수 있으며, 이후 단계에서 개발될 아키텍처, 컴포넌트의 가변성의 근거가 되는 값을 제공할 수 있다. 위의 가변지점의 종류를 바탕으로 PR 명세서를 기술할 수 있는데, PR 명세서 기술의 예는 다음과 같다.

3.3.3 비기능 요구사항 명세

PR 명세서의 작성이 완료되면 각각의 PR과 연관된 비기능적 요구사항을 명세해야 한다. 따라서 본 논문에서는 이러한 비기능적 요구사항을 분석하기 위해서 FURPS+ 모델을 이용하여 PR과의 관계를 명세 한다. FURPS+ 모델을 사용하면 해당 시스템의 중요한 품질적인 요소들을 고려하지 않을 때 일어나는 위험요소를 줄일 수 있다. 아래는 FURPS+ 모델의 각각의 요소를 나열한 것이다.

- ① F(Functional : 기능성) : 본 논문에서는 PR을 가리킨다.
- ② U(Usability : 유용성) : 도움말, 문서
- ③ R(Reliability : 신뢰성) : 오류의 빈도, 회복성, 예측성
- ④ P(Performance : 성능) : 반응 시간, 일정 시간 내 처리할 수 있는 작업량, 정확성, 유용성
- ⑤ S(Supportability : 지원성) : 적합성, 유지보수성, 상호연관성

위와 같은 비기능적 요구사항들을 통틀어 해당 시스템의

<표 5> 가변지점 종류

| 가변지점 종류 | | 설명 |
|--------------------------------|------------------|--|
| 동적인 측면 (Behavior PRelement) | computation | 비즈니스 규칙이나 법규 또는 외부 서비스에 의해 처리되는 경우 |
| | control | 선택적인 상황에서 판단되는 기준 또는 일정한 흐름의 패턴이 가변되는 경우 |
| 정적인 측면 (Static PRelement) | data | 데이터의 구조에서 가변되는 경우 |
| | system interface | 액터와 시스템과의 상호작용에서 나타나는 가변성이 되는 경우 |

<표 6> PR 명세서

| PR ID | 설명 | |
|--------------------------------|--|---------------|
| Ratio | 00% | 가변지점 |
| 동적인 요소 (Behavior PRelement) | PR1a. [control]은 제어흐름이 가변 될수 있는 지점이다. | [control] |
| | PR1b.[c]는 비즈니스 규칙 또는 법규에 의해 가변될 수 있는 지점이다. | [computation] |
| | PR1c.[ec]는 외부 서비스에 의해서 처리될 수 있는 경우이다. | |
| 정적인 요소 (Static PRelement) | [d]PR이 다루는 데이터의 가변성 지점을 나타낸다. | [data] |
| | [I]액터와 시스템과의 상호작용에서 나타나는 가변성 지점을 나타낸다. | [Interface] |

<표 7> 비기능 요구사항 명세서

| 품질요소 | PR | 품질 시나리오 | 영향받는 품질요소 | 우선순위 | 난이도 및 위험도 |
|--------|----------|---------|-----------|------|-----------|
| U(유용성) | PR1. REQ | | R(신뢰성) | H | L |
| | PR3. REQ | | S(지원성) | M | L |
| P(성능성) | PR2. REQ | | R | H | M |
| | PR4. REQ | | R | H | H |
| | PR5. REQ | | R | H | L |
| S(지원성) | PR6. REQ | | | L | M |

품질요소 혹은 품질 요구사항이라고 한다. 이러한 품질요소들은 아키텍처 분석 시 시스템 아키텍처에 많은 영향을 미친다. 예를 들어 높은 성능, 높은 신뢰성에 대한 요구사항은 하드웨어와 소프트웨어를 선택하는데 영향을 끼친다. 품질요소는 하나 이상의 기능적인 요구사항과 관련이 되어있다. 따라서 비기능 요구사항 명세서에서는 각각의 품질요소는 그와 연결되어 있는 기능적 요구사항을 기술하고 각각의 품질요소가 함께 존재하는 다른 품질요소에 의해 어떠한 영향을 받는지를 기술한다. 또한 우선순위 와 난이도 및 위험도는 High, Medium, Low 로 나눌 수 있으며, 본 비기능 요구사항 명세서에서의 표기는 High는 H, Medium은 M, Low는 L로 표기한다.

3.3.4 PR 분석 모델

PR 명세서에서 식별된 PR 요소들의 가변성 지점들은 PR 분석모델로 도식화 된다. 이 모델에서는 컨트롤 클래스, 경계 클래스, 엔티티 클래스를 이용한다. PR 분석모델에서 3 가지 분석 클래스들은 유스케이스에 참여하는 객체들을 식별하고 이들이 내포하는 가변 지점을 나타내기 위한 도구로서 사용된다. <표 8>은 PR 분석모델에서 사용하게 될 UML 확장 메커니즘인 스테레오타입을 정리한 것이다.

<표 8> PR 분석모델에 대한 스테레오타입

| 모델 개념 | 분석 클래스 | 스테레오타입 |
|-------------------------|----------------|-------------|
| computation VP | control class | <<v.p:C>> |
| external computation VP | control class | <<v.p:EC>> |
| control VP | control class | <<v.p:Ctl>> |
| data VP | entity class | <<v.p:D>> |
| system interface VP | boundary class | <<v.p:I>> |

3.4 도메인 요구사항 변경관리

소프트웨어 개발 중 추가되는 요구사항이 도출되며, 기존 요구사항이 변경되면서 현존하는 요구사항을 만든다. 이러한 추가와 변경을 효율적으로 관리하는 것은 매우 중요하다. 도메인 요구사항 변경관리 또한 해당 시스템의 요구사항 변경으로 인해 발생하는 모든 관련 활동들을 다루는 아주 광범위한 활동이다. 도메인 요구사항 변경은 해당 도메인의 시스템을 개발 시 필요로 하는 요구사항으로 형상화될 때 발생할 수 있다. 즉 도메인 요구사항 추출 및 분석단계에서 산출되는 일반화된 요구사항 리스트가 개발하려는 시스템의 요구사항이 될 수 있다. 일반화된 요구사항 리스트를 살펴보면 PR ID가 존재하는데, PR ID를 통해 요구사항을 추적할 수 있다. 요구사항 추적 가능성은 요구사항 변경의 영향과 결과를 분석하기 위한 기초자료라는 점에서 요구사항 관리의 핵심이 되며, 도메인 요구사항 관리에서는 PR 컨텍스트 매트릭스를 통해 일반화된 요구사항 리스트를 산출하기 때문에 요구사항 추적가능성을 향상시켜준다. 요구사항 변경이 발생할 때 일반화 된 요구사항 리스트가 변경이 일어날 수 있는데, 이때 해당 산출물의 무결성과 추적성을 보장하기 위해 형상관리를 수행한다. 형상관리는 다음

과 같은 프로세스를 따른다[9].

- ① 계획 : 형상관리 아이템을 결정하며, 그것을 저장할 장소와 변경통제를 위한 절차를 결정한다.
- ② 실행 : 적절한 도구를 사용하여 아이템을 배포하며, 산출물 버전을 관리하고, 저장소를 지정하여 형상관리 아이템을 저장한다.
- ③ 형상관리 아이템의 상태를 모니터링 하며 감사를 실시한다.

형상관리 프로세스는 형상관리자를 통해서 이루어지며, 산출물 또는 소스코드, 요구사항 명세서, 테스트 계획서 등이 형상관리 아이템이 될 수 있다. 도메인 요구사항 변경관리는 도메인 요구사항 추출 및 일반화 단계에서 산출되는 일반화된 요구사항 리스트를 통해 요구사항 추적이 가능하며, 형상관리를 통한 산출물의 무결성과 추적성을 보장함으로써 도메인 요구사항 변경관리를 향상 시킬 수 있다.

4. 사례연구

본 논문에서 제안한 도메인 요구사항 관리 활동과 세부 활동을 인사시스템에서의 조직관리 도메인에 적용해 보았다. 조직 관리를 시스템화함으로써 사용자에게 해당 조직의 정보, 조직이력, 조직원의 구성정보, 조직도를 제공할 수 있으며, 조직 관리를 하는데 있어서 생산성 향상과 업무 효율성을 극대화할 수 있다. 해당 도메인을 제공하는 대표적인 시스템으로는 LD사, LH사, LL사 의 시스템을 꼽을 수 있다. 조직관리 시스템을 구축하려는 모든 업체는 공통적인 요구사항이 있을 수 있고, 각 업체만이 가지고 있는 요구사항이 있을 수 있다. 따라서 공통성과 가변성을 고려하지 않은 프로세스에 입각하여 요구사항을 관리 할 경우 업체에 따라 별도로 요구사항을 관리하게 되어 다른 업체의 조직관리 시스템을 구축할 때 비용과 시간의 낭비를 가져올 수 있다. 따라서 본 논문에서는 조직 관리의 요구사항을 분석하여 공통 요구사항을 식별하고 가변적 요구사항을 일반화시켜 재사용할 수 있는 핵심자산으로 관리할 수 있는 방법을 제안하였다.

4.1 적용사례

먼저 조직 관리의 도메인 범위를 결정해야 한다. 도메인 범위를 결정하기 위해 해야 할 작업은 업무 영역을 정의하는 것과 용어를 정리하는 것이다. 조직 관리의 업무 영역 정의는 아래와 같다. 조직은 경영목적을 달성하기 위해 계통적으로 편성되는 업무처리 기구이다. 조직 관리의 업무영역은 ①조직신설, ②조직발령, ③조직폐지로 나눌 수 있다. 조직관리 업무의 전사적인 주관은 주관부서가 담당을 하며, 조직책임자가 해당 조직의 전사적 업무를 담당하며 권한을 위임할 수도 있다. (그림 10)은 위의 조직관리 업무영역을 바탕으로 조직 관리에서 사용되는 용어를 정리한 것이다.

조직 관리의 업무 영역과 용어를 정의가 되면 조직관리

| 도메인 | 용어 | 용어 상세 설명 |
|-------|--------|-----------------------------------|
| 조직관리 | 조직코드 | 해당 조직을 식별할 수 있는 코드를 말한다. |
| 조직관리 | 조직발령구분 | 조직신설, 조직명변경, 조직폐지 등을 구분으로 둘 수 있다. |
| 조직관리 | 조직생성일 | 조직 신설시 입력되는 일자를 말한다. |
| 조직관리 | 조직발령일 | 조직을 발령을 일자를 말한다. |
| 조직관리 | 조직폐지일 | 조직이 폐지될때 입력되는 일자를 말한다. |
| 조직관리 | 조직레벨 | 조직의 상위조직, 하위조직을 결정하는 것을 말한다. |
| 조직관리 | 상위조직 | 해당 조직을 조회시 그 조직의 상위조직을 말한다. |
| 조직관리 | 사업장 | 해당 조직을 관리하고 있는 사업장을 말한다. |
| 조직관리 | 사업부 | 여러 개의 조직을 관리하는 부서를 말한다. |
| 조직관리 | 조직책임자 | 해당 조직의 책임자를 말한다. |
| 조직관리 | 소재지 | 해당 조직의 주소를 말한다. |
| | | |

(그림 10) 조직관리 용어 정의

<< LD Req.>>

| Req ID | 요구사항 | 요구사항 Description |
|--------|-----------|-------------------------|
| PR1 | Login | 사용자는 인증을 통해 시스템에 접속한다. |
| PR2 | Logout | Logout |
| PR3 | 조직신설 | 새로운 조직을 생성한다. |
| PR4 | 조직발령 | 조직명 변경, 조직폐지 기능 |
| PR5 | 과거조직관리 | 폐지된 조직을 관리한다. |
| PR6 | 조직원 정보 보기 | 해당 조직의 구성원의 정보를 볼 수 있다. |
| PR7 | 조직정보 보기 | 사용자 권한에 따라 조직의 정보를 본다 |
| PR8 | 조직이력관리 | 조직발령에 따라 이력을 관리한다. |
| PR10 | 조직책임자 관리 | 발령에서 수정 할 수 있다. |
| | | |

<< LL Req.>>

| Req ID | 요구사항 | 요구사항 Description |
|--------|-----------|-------------------------|
| PR1 | Login | 사용자는 인증을 통해 시스템에 접속한다. |
| PR2 | Logout | Logout |
| PR3 | 조직신설 | 새로운 조직을 생성한다. |
| PR4 | 조직발령 | 조직명 변경, 조직폐지 기능 |
| PR5 | 과거조직관리 | 폐지된 조직을 관리한다. |
| PR6 | 조직원 정보 보기 | 해당 조직의 구성원의 정보를 볼 수 있다. |
| PR7 | 조직정보 보기 | 사용자의 권한에 따라 조직의 정보를 본다. |
| PR8 | 조직이력 관리 | 조직발령에 따라 이력을 관리한다. |
| PR10 | 조직책임자 관리 | 발령에서 수정 할 수 있다. |
| | | |

<< LH Req.>>

| Req ID | 요구사항 | 요구사항 Description |
|--------|-----------|-------------------------|
| PR1 | Login | 사용자는 인증을 통해 시스템에 접속한다. |
| PR2 | Logout | Logout |
| PR3 | 조직신설 | 관리자가 직접 입력하여 생성한다. |
| PR4 | 조직발령 | 조직명 변경, 조직 폐지 기능 |
| PR5 | 조직원 정보 보기 | 해당 조직의 구성원의 정보를 볼 수 있다. |
| PR6 | 조직정보 보기 | 모든 사용자가 조직을 검색 할 수 있다. |
| PR7 | 조직이력 관리 | 조직발령에 따라 이력을 관리한다. |
| PR8 | 조직책임자 관리 | 조직에서 직접 수정한다. |
| | | |

(그림 11) 조직 관리 요구사항 추출 리스트

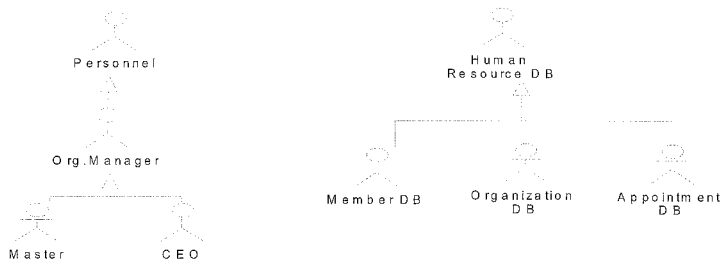
| PR ID | PR | Property/ratio | LD | LL | LH |
|-------|--------------------------------------|----------------|----|----|----|
| PR1 | Login | C100% | ○ | ○ | ○ |
| PR2 | Logout | C100% | ○ | ○ | ○ |
| PR3 | 새로운 조직을 생성한다. | C100% | ○ | ○ | ○ |
| PR4 | 조직코드생성 | CV100% | ○ | ○ | ○ |
| | PR311 : 조직 신설시 자동생성 | | ○ | ○ | X |
| | PR312 : 관리자가 직접 입력(조직코드중복체크) | | X | X | ○ |
| PR5 | 조직발령 | C100% | ○ | ○ | ○ |
| PR6 | 과거조직 관리 | P66% | ○ | ○ | X |
| PR7 | 해당 조직의 구성원 리스트를 보여준다. | C100% | ○ | ○ | ○ |
| PR8 | 조직정보 보여 준다 | CV100% | ○ | ○ | ○ |
| | PR711 : 사용자 권한에 관계 없이 모든 조직을 볼 수 있다. | | X | X | ○ |
| | PR712 : 사용자 권한에 따라 조직을 보여 준다. | | ○ | ○ | X |
| PR9 | 조직 이력을 관리한다. | C100% | ○ | ○ | ○ |
| PR10 | 조직책임자를 관리한다. | CV100% | ○ | ○ | ○ |
| | PR911 : 조직관리에서 변경할 수 있다. | | X | X | ○ |
| | PR912 : 조직책임자는 발령에서만 변경할 수 있다. | | ○ | ○ | X |

(그림 12) 조직관리 PR 컨텍스트 매트릭스

| PR ID | Property / ratio | 요구사항 |
|-------|------------------|-----------------------|
| PR1 | C100% | Login |
| PR2 | C100% | Logout |
| PR3 | C100% | 조직신설 |
| PR4 | CV100% | 조직코드 생성 |
| PR5 | C100% | 조직발령 |
| PR6 | P66% | 과거조직 관리 |
| PR7 | C100% | 해당 조직의 구성원 리스트를 보여준다. |
| PR8 | CV100% | 조직의 정보를 보여준다. |
| PR9 | CV100% | 조직이력을 관리한다. |
| PR10 | CV100% | 조직책임자를 관리한다. |

(그림 13) 일반화된 조직 관리 요구사항 리스트

<< 액터 모델 >>



<< 액터 명세서 >>

| 액터 | 설명 | Responsibility |
|--------------|-------------------|---|
| Master | 조직관리의 모든 업무를 담당한다 | 조직발령(조직 명칭 변경, 조직폐지) 조직 신설, 조직구성원 조회 |
| CEO | 모든 업무를 관할한다. | 모든 업무를 모니터링 하고 지시한다. |
| Personnel | 권한이 없는 일반사용자 이다. | 조직의 정보 및 해당 조직의 구성원리스트를 조회 할 수 있다 |
| Org. Manager | 특정 조직의 업무를 담당한다. | 해당 조직 구성원 검색 해당 조직 구성원 관리 |

(그림 14) 조직관리 도메인 액터 모델

도메인 요구사항 추출 및 일반화 작업을 하며, 이 작업은 요구사항 추출 리스트 작성, PR 컨텍스트 매트릭스 작성, 일반화된 요구사항 리스트 작성으로 구성된다. 요구사항 추출 리스트는 조직 관리를 시스템으로 사용하고 있는 업체 즉 LD사, LH사, LL사의 시스템을 분석하여 요구사항 추출 리스트를 작성했고, PR 컨텍스트 매트릭스를 작성하여 각 요구사항의 속성과 그 속성의 비율을 결정하였으며, 가변성이 있는 요구사항들은 일반화 작업을 하여 작성하였다.

(그림 11)을 바탕으로 작성된 조직관리 PR 컨텍스트 매트릭스를 볼 때 Login, Logout, 조직신설, 조직발령, 조직이력, 조직원 구성원 리스트는 모든 시스템에 존재하는 것으로 추출되었기 때문에 공통적인 속성으로 판단되었고, 조직코드생성, 조직책임자 관리는 가변적인 속성을 가지고 있으나 모든 시스템에 공통적인 요구사항 이므로 가변성을 지닌 공통적인 속성으로 판단되었다. 또한 과거조직 관리는 하나의 시스템에 존재하지 않은 요구사항이므로 가변적인 속성을 판단했다. (그림 13)은 조직 관리 PR 컨텍스트 매트릭스를 바탕으로 일반화된 요구사항의 리스트를 나타낸 것이다.

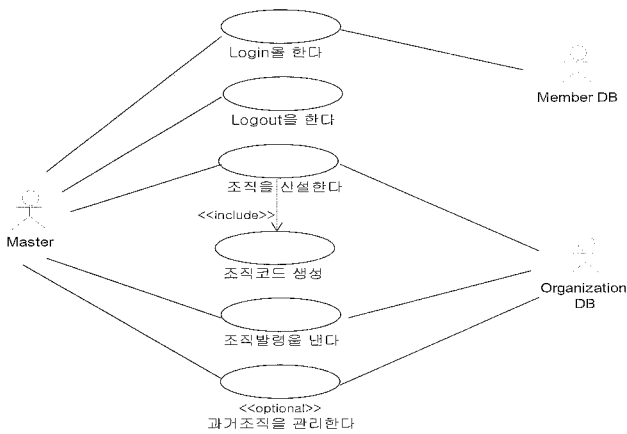
도메인 요구사항 추출 및 일반화 작업이 완료되면 도메

인 요구사항 분석 및 모델링 작업을 한다. 도메인 요구사항 분석 및 모델링은 도메인 유스케이스 모델, PR 명세서 기술, PR 분석모델 도식으로 구성된다. 또한 도메인 유스케이스 모델은 도메인 액터 모델, PR 유스케이스 매트릭스, 유스케이스 다이어그램으로 구성된다. (그림 14)는 조직 관리의 도메인 액터 모델을 나타낸 것으로서, 조직을 관리하는데 있어서 Master, CEO는 모든 권한을 가지고 있으며 업무의 주체가 되며, 조직책임자는 해당 조직의 업무를 보는 권한을 가지는 것을 보여준다. 외부 시스템으로는 사용자의 정보를 저장해 놓은 사용자 데이터베이스, 모든 조직의 데이터 및 정보가 저장되어 있는 조직 데이터베이스, 사원의 발령 데이터 및 정보가 저장되어 있는 발령 데이터베이스로 나타낼 수가 있다. 3개의 외부 시스템은 하나의 인사관리 데이터베이스로 일반화 할 수 있다.

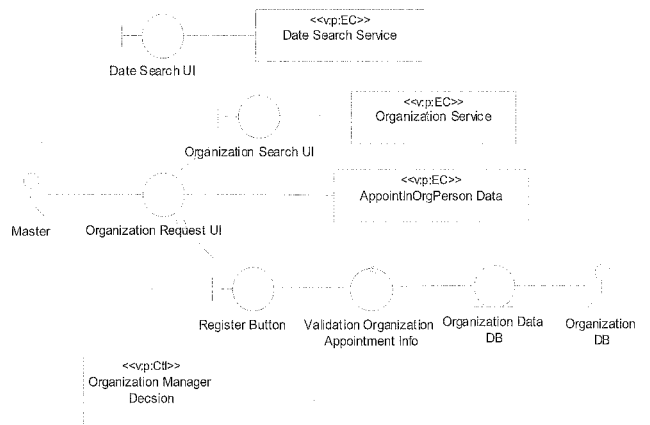
PR 유스케이스 매트릭스를 보면 '조직을 신설한다' 라는 유스케이스는 PR4, PR3에 공통적으로 일치한다. 그래서 '조직을 신설한다' 라는 유스케이스는 PR3와 PR4 와의 관계는 <<include>>로 형성되며, '과거조직을 관리 한다'라는 유스케이스는 가변적인 속성을 지닌 PR과 일치하므로 선택적인

| PR ID | PR | Usecase | Property/ Ratio | Login을 한다. | Logout을 한다. | 조직을 신설 한다. | 조직을 발령을 낸다 | 과거조직을 관리한다. | |
|-------|---------|---------|-----------------|------------|-------------|------------|------------|-------------|-------|
| PR1 | Login | | C100% | 0 | | | | | |
| PR2 | Logout | | C100% | | 0 | | | | |
| PR3 | 조직신설 | | C100% | | | 0 | | | |
| PR4 | 조직코드 생성 | | CV100% | | | 0 | | | |
| PR5 | 조직발령 | | C100% | | | | 0 | | |
| PR6 | 과거조직관리 | | P66% | | | | | 0 | |
| ... | | | | | | | | | |

(그림 15) 조직 관리 PR 유스케이스 매트릭스



(그림 16) 조직관리 유스케이스 다이어그램



(그림 18) 조직발령 PR 분석모델

유스케이스로 된다. (그림 16)은 PR 유스케이스 매트릭스 바탕으로 유스케이스 다이어그램을 도식한 것이다.

유스케이스 다이어그램이 작성되면 각 PR에 해당하는 명세서를 기술해야 한다. 본 논문에서는 조직발령 PR을 예를 들어 PR 명세서를 기술하였다.

PR 명세서에서는 조직발령에 대한 요구사항의 동적인 PR 요소를 기술하면서 조직명 검색, 조직발령일을 검색할 날짜검색은 외부 서비스를 이용하여 처리할 수 있음을 [ec](external computation)로 명시하였고, 조직책임자를 결정할 때 인사발령에서 결정하는 시스템이 있고 조직발령에

서 직접 결정하는 시스템이 있으므로 가변성이 있는 [c] (computation)로 명시하였다. 정적인 PR요소에서는 조직발령에 있어서 조직발령 기본 자료를 입력하는 항목과 조직발령 사용자 인터페이스, 조직발령 데이터를 다루는 조직 데이터 객체를 식별하였고 발생할 수 있는 가변성으로 기술하였다.

(그림 18)에 나타난 조직발령 PR 분석모델을 통해 조직발령 시 구성되는 사용자 인터페이스를 확인할 수 있으며 데이터의 흐름, 외부시스템과 접근 데이터베이스들을 알 수 있다. 이상으로 본 논문에서는 조직관리 도메인을 정하고

| PR No.PR4 | 설명 : 조직발령 | |
|-----------|--|---|
| 속성 및 비율 | C 100% | 가변지점 |
| 동적인 PR 요소 | PR4a. [ec] 발령날 조직을 조직명, 조직코드로 검색한다. PR4b. [조직발령등록] 자료를 입력한다. PR4c. [ec]조직발령일을 검색한다. PR4d. 해당 조직의 구성원을 직급별로 보여준다. PR4e. [c] 조직책임자를 결정한다. PR4f. [ec] 상위조직을 검색하여 입력한다. | a[ec] 조직명 검색 service c[ec] 날짜검색 service e[c] 조직책임자 결정 규칙 적용 f[ec] 조직명 검색 service |
| | b1. [조직발령등록 기본자료]가 입력되지 않을 경우 예외 메시지를 보낸다. b2. 발령구분, 조직레벨, 사업장, 사업부, 소재지는 코드에서 불러온다. b3. 조직생성일은 수정 불가능하게 한다. d1. 발령에 해당 조직에 속해 있는 인원을 직급별로 보여 준다. | d[ec] 인사발령의 데이터를 접근하여 가져온다 |
| 정적인 PR 요소 | 조직발령기본자료 | [조직명, 조직코드, 조직발령구분, 조직발령일, 사업장, 사업부, 조직레벨, 소재지] |
| | 조직 데이터 Object 생성 | 조직기본자료 |
| | 조직발령 Interface | |

(그림 17) 조직관리 PR 명세서

도메인에 속한 요구사항들의 속성을 객관적인 지표를 만들어 분석하고 이를 모델링하는 작업을 제시하였다. 조직관리 도메인을 구축하는 시스템은 이러한 작업을 통해 공통성과 가변성을 분석함으로써 하나의 시스템에 국한된 요구사항 분석이 아니라 조직관리 도메인에 속하는 시스템에 적용할 수 있는 C&V의 확률적인 비율을 얻을 수 있을 것이다.

4.2 적용결론 및 평가

본 논문에서 제시한 프로세스를 조직관리 시스템에 적용하여 사례연구를 수행하였다. 공통성과 가변성을 고려하지 않은 요구사항 관리 프로세스를 적용할 경우는 해당 시스템에만 적용되는 요구사항 관리에 국한된다. 그러므로 유사하거나 혹은 동일한 도메인의 시스템을 구축할 경우에도 예전에 분석된 요구사항을 고려할 수 있으나, 같은 요구사항 프로세스를 반복하게 된다. 따라서 동일한 프로세스를 반복적으로 적용해야 하므로 요구사항 재사용성이 감소하며, 그에 따라 시간과 비용이 비효율적으로 사용되어 생산성이 감소되는 결과를 가져올 수 있다. 하지만 3장에서 제시한 공통성과 가변성을 고려한 요구사항 프로세스를 적용할 경우에는 유사하거나 혹은 동일한 도메인의 시스템을 구축할 때 기존의 분석된 공통성과 가변성을 바탕으로 한 요구사항이 분석될 수 있으며, 새로운 요구사항이 공통적인 속성을 가질 때는 공통성, 선택적인 속성을 가질 때는 가변성으로 포함시킴으로써 기존 요구사항을 재사용할 수 있다. 그 결과 해당 도메인의 시스템 개발의 시간과 비용을 줄일 수 있으며 생산성을 향상되는 결과를 가져올 수 있다.

결론적으로 기존의 요구사항 관리 프로세스를 적용하여 하나의 업체의 요구사항을 관리하는 것에 비해 공통성과 가변성을 고려한 도메인 요구사항 관리는 해당 도메인의 요구사항 재사용을 극대화 할 수 있다.

하지만 도메인 요구사항을 관리하는 프로세스 각 단계마다 나오는 산출물들의 체계적이고 지속적인 관리가 필요하다. 다시 말해 각 프로세스 별 산출물들의 공통적인 관리를 위한 모델이 필요하고, 각 산출물을 정형적으로 표현하고 산출물들 사이의 관계를 정형적으로 다루기 위한 방법을 바탕으로 한 저장소가 있어야 한다. 본 논문에서는 이러한 저장소에 관한 문제를 다루지 않았지만, 향후 산출물을 체계적이고 지속적인 관리를 위해 저장소에 관한 연구가 이루어져야 할 것이다.

5. 결론 및 평가

본 논문에서는 프로덕트 라인에서 핵심자산 중의 하나인 도메인 요구사항을 수집하고 이를 매트릭스 형태의 객관적인 근거로 작성한 후, 이를 바탕으로 C&V를 확률적인 비율로 판단하여, 체계적으로 관리 할 수 있는 프로세스를 제시하였다. 이때 소프트웨어 공학 분야에서 가장 신뢰할 수 있는 자료 즉 경험치를 바탕으로 시스템 개발 시 각 단계에 적절히 사용되고 또 다른 시스템에 개발에 반복적으로 사용

함으로써 프로덕트 군(family)을 만들 수 있도록 하였다. 이는 요구사항들도 프로덕트 라인 안에서 재사용될 수 있는 핵심자산이고, 해당 도메인의 일정한 범위 내에서 요구사항들을 분석, 관리한다면 재사용할 수 있고, 다른 핵심자산의 C&V를 분석할 때 객관적인 근거 자료가 될 수 있다.

제안한 방법은 요구사항 관리를 도메인 범위결정, 도메인 요구사항 추출 및 일반화, 도메인 요구사항 분석 및 모델링, 도메인 요구사항 변경관리로 세분화하였다. 도메인 범위 결정에서는 도메인 전문가 혹은 해당 관리 부서를 통해 업무영역을 정하고, 그 영역에 포함된 용어를 정의한다. 업무영역정의서와 용어정의서를 바탕으로 도메인의 요구사항을 추출하고, 추출된 요구사항을 일반화시킴으로써 산재되어 있는 요구사항의 수와 복잡성을 줄일 수 있다. 도메인 요구사항 추출 및 일반화 단계에서 산출된 일반화된 요구사항 리스트를 통해 요구사항 분석 및 모델링 작업을 하게 되는데, 이때 UML을 사용하여 도식하며 UML 확장개념인 스테레오타입을 이용하여 유스케이스와 요구사항의 공통성과 가변성 속성을 표현한다. 마지막으로 요구사항의 변경이 일어날 경우 형상관리를 통해 요구사항 변경관리를 함으로써 산출물 간의 무결성과 추적성을 보장하게 된다.

또한 각 프로세스는 도메인 요구사항의 공통성을 찾는 데 중점을 두었고, 공통성을 기반으로 하여 해당 도메인 요구사항의 선택적인 사항을 붙이는 가변성에 대하여 표현하였다. 특히 PR 컨텍스트 매트릭스를 통해 도메인 요구사항을 명세 최소 단위인 PR로 분할하여 일반화함으로써 그 수와 복잡성을 줄였다. 따라서 요구사항 관리자는 PR 컨텍스트 매트릭스를 통해서 해당 도메인 요구사항을 쉽게 파악할 수 있는 이점이 있다. 특히 PR 컨텍스트 매트릭스를 통해 먼저 요구사항의 속성과 속성의 비율을 논리적으로 표현하였고, 이를 바탕으로 하여 일반화된 요구사항 리스트를 작성하여, 요구사항 변경이 있을 시 추적성을 용이하게 했으며, 도메인 요구사항 분석 및 모델링에서는 UML 확장 메커니즘인 스테레오타입을 이용하여 유스케이스 속성 및 가변지점을 분석하였다.

향후에는 도메인 요구사항 관리 각 프로세스에서 나오는 산출물을 체계적이고, 지속적인 관리를 위해 산출물 저장소에 대해 연구가 이루어져야 하며[15], 이를 바탕으로 프로덕트 라인의 또 다른 핵심자산인 도메인 아키텍처를 개발하는 연구가 필요하다. 왜냐하면, 도메인 아키텍처는 도메인 요구사항과 관계가 잘 정립되어 있어야 하며, 또한 공통성과 가변성을 반영하여 명시할 수 있어야 하기 때문이다.

참고 문헌

[1] Muthig, D., Atkinson C., "Model-Driven Product Line Architecture," G. Chastek, editor, Software Product Lines: In Proceedings of the econd Software Product Line Conference(SPLC2), San Diego, U.S.A., Aug. 2002, Heidelberg, Germany: Springer Lecture Notes in Computer

Science Vol.2379, pp.110-129, 2002.

- [2] P.Clements and L. Northrop, "A Framework for Software Product Line Practice," SEI Report 2002.
- [3] 문미경, 엄근혁, "소프트웨어 프로덕트 라인에서 핵심자산으로서 요구사항을 관리하는 방법", 한국정보과학회 논문지, 2004. 6.
- [4] Clements,P.and Northrop,L.,Software Product Lines : Practices and Patterns, Addison-Wesley, Aug., 2001.
- [5] 이계준,강교철 "프로덕트 라인 소프트웨어 개발 프로세스", 정보과학학회지 학회지, 2002. 3.
- [6] Griss, M. L., Favaro, J., and d'Alessandro, M., "Integrating Feature Modeling with the RSEB," In Proceedings of 5th InternationalConference on Software Reuse, Victoria Canada, June, IEEE, pp.76-85, 1998.
- [7] ClauB, M., "Generic Modeling using UML extensions for variability," OOPSLA 2001 Workshop on Domain Sepecific Visual Language, 2001.
- [8] 김행곤, 김수연, "프로덕트 라인 기반의 웹 학습 시스템 개발", 한국정보처리학회 논문지, 2005. 8.
- [9] Pankaj J, Addison-Wesley CMM in Practice: Processes for Executing Software Projects at Infosys, 1999.
- [10] Bass,L., Clements, P., Donohoe, P., McGregor, J., Northrop,L., "Fourth Product Line Practice Workshop Report," Software Engineering Institute, USA, Nov., 1999.
- [11] Berard, E., Essays in Object-Oriented Software Engineering, Prentice Hall, 1992.
- [12] Kotonya, G., Sommerville, I., "Requirements Engineering with viewpoints," Software Engineering Journal, Volume: 11 Issue: 1, Jan., pp.5-18, 1996.
- [13] Clements,P.,and Northrop,L.,"A Framework for Software Product Line Practice-Version 4.1[online]," Carnegie Mellon, Software Engineering Institute, 2003.
- [14] Frakes, W., Prieto-Diaz, R., and Fox, C., "DARE-COTS: A Domain Analysis Support Tool," Proceedings of XVII International Conference of the Chilean Computer Science Society, pp.73-77, 1997.
- [15] 장호동, "소프트웨어 프로덕트 라인에서 재사용을 지원하기 위한 MOF기반 메타데이터 저장소 개발", 부산대학교 대학원, 2004.



박 동 수

e-mail : mysunsua@paran.com
 2006년 숭실대학교 정보과학대학원
 소프트웨어공학과 (공학석사)
 현 재 LG CNS 근무
 관심분야: 소프트웨어공학, 요구사항관리,
 PLE



김 동 규

e-mail : topaz1979@gmail.com
 2004년 숭실대학교 컴퓨터학부 (공학사)
 2006년 숭실대학교 대학원 컴퓨터학과
 (공학석사)
 2006년~현재 숭실대학교 대학원
 컴퓨터학과 박사과정
 관심분야: 소프트웨어공학, 요구사항관리, PLE, 소프트웨어
 테스트



정 기 원

e-mail : chong@ssu.ac.kr
 1967년 서울대학교 전기공학과(공학사)
 1981년 미국 알라바마주립대(현츠빌)
 전산학과 석사
 1983년 미국 텍사스주립대(알링턴)
 전산학과 박사
 1971년~1975년 한국과학기술연구소 연구원
 1975년~1990년 국방과학연구소 책임연구원
 1990년~현재 숭실대학교 컴퓨터학부 교수
 관심분야: 소프트웨어공학, 소프트웨어프로세스, 정보시스템
 감리, 전자거래(CALS/EC), 유비쿼터스 컴퓨팅