

CBD 기반으로 설계 및 구현한 연구관리 시스템

오상균*, 김은기*

*한밭대학교 정보통신공학과
e-mail:unearth@nate.com,

CBD Based Design and Implementation of Research Management Systems

Sang-Gyun Oh*, Eun-Ki Kim*

*Dept. of Information Communication Engineering, Han-Bat University

요 약

본 연구에서는 CBD 기반의 RUP 모델을 이용하여 재 사용이 가능한 컴포넌트들을 설계하였으며, 이를 이용하여 연구관리 시스템을 구현하였다. 이를 위하여 연구관리 시스템의 주요 사용자 인터페이스인 등록관리 및 변경관리를 위한 컴포넌트를 구성하였으며, 구성된 컴포넌트들은 관련 SW개발에 효율적으로 재 사용될 수 있을 것으로 생각된다.

1. 서론

본 논문에서는 SW 개발의 생산성을 높일 수 있는 CBD 기반의 개발 방법론을 비교 분석하고, 이를 이용한 연구관리 시스템의 설계 및 구현에 관하여 기술하였다. 알려진 여러 컴포넌트 기반 개발 방법론들 중에 널리 사용되는 네 가지 CBD 방법론을 대상으로 비교/분석을 실시하여 각 방법론을 평가함으로써 각 방법론의 장단점 및 특성을 파악하고자 하였다.

현재 많은 연구기관들은 연구관리 시스템을 자체적으로 또는 외부 용역을 통해 개발하고 있으나 통합 정보시스템과의 연계기능을 정확히 파악하지 않고 개발함으로써 시스템 운용에 많은 어려움을 겪고 있다. 특히, 연구업무의 관리에 소요되는 공통적인 기능들의 재사용이 불가능하므로 많은 개발비용을 초래하고 있으며, 연구업무의 변화에 유연하게 대처할 수 없으므로 시스템의 유지보수에 많은 어려움이 따르고 있다.

기존 방식의 경우, 구조적 분석, 설계 방법을 이용하여 연구관리 시스템을 개발함으로써 시스템의 복잡도 증가에 따라 유지보수의 유연성을 보장할 수 없으며, 또한 시스템의 개발 적시성(time-to market)과 신뢰성 등에서 많은 문제점을 가지고 있다.

시스템 개발 분야에서 중요한 이슈로는 소프트웨어의 개발 생산성 및 유지보수성의 제고 문제를 들 수 있다. 소프트웨어의 개발 생산성을 높이기 위해서는 소프트웨어를 구성하는 여러 기능 모듈들의 재사용이 가능해야 한다. 기능 모듈의 재사용 단위로써 컴포넌트(component) 개념이 도입되면서 소프트웨어의 개발 생산성은 비약적으로 향상되고 있다. 이질적인 운용환경과 기능 요소들 간의 상호운동을 허용하며, 기능들의 동적인 변화에 신속히 대처할

수 있는 컴포넌트 기반의 소프트웨어 개발(CBD: Component Based Development) 방법이 새로운 비즈니스 요구를 해결할 수 있는 최상의 전략으로 간주되고 있다. CBD는 잘 정의된 인터페이스를 기반으로 소프트웨어 부품들의 조립을 통해 응용 개발의 생산성과 유지보수성을 보장할 수 있다. 뿐만 아니라, 웹과 인터넷 활용기술의 지원을 통한 컴포넌트의 보급과 대중화에 따라 비즈니스 측면의 경제성은 CBD의 이용을 더욱 가속화 시키고 있다. 따라서 기반 시스템에서 미들웨어(middleware), 그리고 어플리케이션(application)에 이르기 까지 CBD의 적용이 확산 되고 있다.

따라서 연구 분야의 특성에 따라 연구관리 프로세스의 모델을 정립하고, 각 분야별 기능을 컴포넌트화하여 연구조직 및 연구원들이 필요한 기능을 선택으로 사용할 수 있도록 연구관리 시스템을 구축한다면 짧은 시간에 적은 비용의 투자만으로도 연구관리 업무를 효율적으로 수행할 수 있을 것이다.

1.1 연구 목적 및 내용

본 논문에서는 연구관리 시스템의 특성 분석을 통하여 CBD 프로세스를 체계적으로 지원할 수 있는 컴포넌트 구축방법에 대한 이론적 근거를 제시하며, 이를 바탕으로 CBD를 기반으로 하는 RUP(Rational Unified Process)모델을 이용하는 연구관리 시스템을 제안 하고자 한다. 본 논문의 목적은 다음과 같이 3가지로 분류할 수 있다.

첫째, 연구관리 시스템의 특성 분석

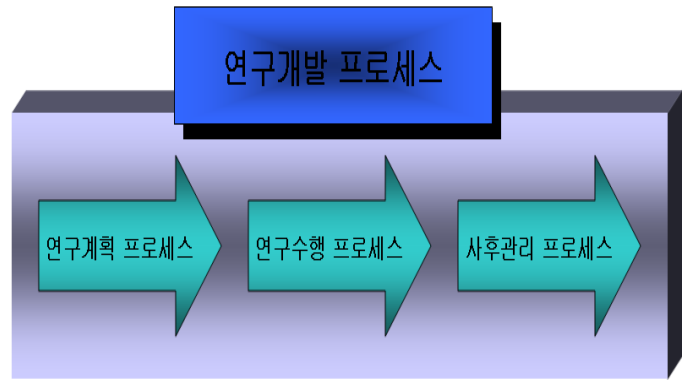
둘째, 재사용이 가능한 컴포넌트 단위의 체계화

셋째, CBD 기반의 RUP 모델을 이용한 연구관리 시스템의 설계 및 구현이다.

1.2 연구관리 시스템의 특성

연구관리 시스템이란 연구기관의 연구 활동에 있어서 연구기획부터 진행과정 및 연구개발 종료 후 성과물 관리 까지 모든 일련의 과정들을 체계적으로 관리하여, 연구의 효율성을 높이고 연구 성과물의 가치를 극대화 하도록 지원하는 시스템으로 크게 연구계획관리, 연구수행관리, 사후관리로 나눌 수 있다.

적용분야로는 국가 연구기관 및 공공기관의 연구과제 관리, 대학 및 교육기관의 내부연구과제 관리 및 외부연구과제 관리, 과제관리가 주 사업인 기관의 연구과제 관리, 기업 연구소의 내부연구과제 관리 및 외부 위탁연구과제 관리에 적용 할 수 있다.



(그림 1). 연구관리 프로세스 구성도

2 재사용이 가능한 컴포넌트 단위의 체계화

업무프로세스가 급변하고, 다변화되고 있는 상황에서 새로운 정보기술과 더불어 업무 프로세스의 혁신을 가져올 체계적인 소프트웨어 개발 방법이 요구됨에 따라 객체지향 방법은 현실 세계의 객체를 중심으로 자연스러운 모델링으로써 요구를 충족시켜 주고 있다. 기존의 구조적 분석/설계 방법을 적용해서 만들어진 기능들은 시스템이 변하지 않는다는 전제조건하에 만들어져서 업무프로세스가 바뀌게 되면 유지보수에 많은 비용을 투자하는 문제점을 가지고 있다. 또한 정보공학 방법은 데이터 모델링에 비중을 두었기 때문에 업무 프로세스 관점이 취약하고, 실제 프로젝트를 수행할 경우 많은 관리자나 개발자들이 향상된 개발 프로세스를 적용하는데 어려움이 있다. 구조적 분석/설계 방법이 기능 중심적이고, 정보공학 방법이 데이터 중심이라면, 객체지향 방법은 데이터 요소와 기능적 요소를 하나의 관점으로 표현한 객체중심이다. 객체지향방법은 다양한 표현 기법을 통해 쉽게 객체를 추출하고, 추상화 수준을 높여 정확하고 단순하게 실세계를 표현하여 복잡성을 줄이고, 재사용이 가능하게 하고 있다. 객체지향 방법은 구조적 방법과 정보공학 방법의 장점을 수용함으로써 상호 보완적인 방법을 추구하고 있다.

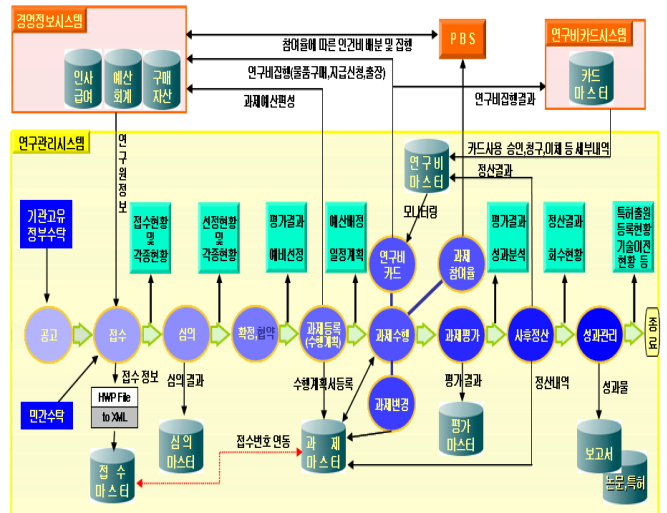
3. RUP 기반의 연구관리 시스템 구현

본 장에서는 컴포넌트 기반 소프트웨어 개발 방법인 RUP를 이용하여 업무영역 분석 및 컴포넌트 설계기법과 그리고 이를 통한 구체적인 적용 사례로서 연구관리 시스템의 주요업무인 과제등록관리, 과제변경관리 기능을 분석하고 설계하여 컴포넌트가 도출되는 과정을 살펴보고, 이를 기반으로 시스템 아키텍처 및 데이터베이스 모델링을 설계 하고자 한다.

3.1 연구관리 프로세스

연구관리 프로세스는 그림 1과 같이 실행 단계별로 연구계획 프로세스, 연구수행 프로세스 및 사후관리 프로세스 3개의 단계로 구분하여 분석 하였다.

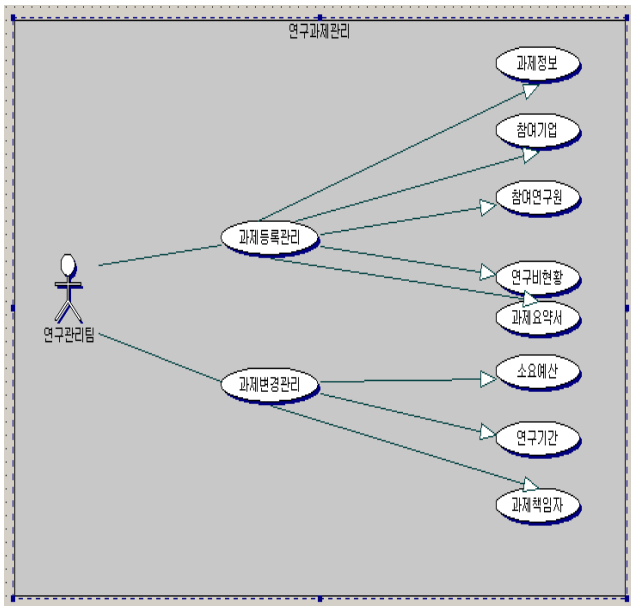
연구관리 시스템 전체 구성도는 (그림 2)와 같다.



(그림 2). 연구관리 시스템 전체 구성도

3.2 사용사례 다이어그램

그림 3은 업무분석과 요구사항을 통하여 시스템과 상호 작용하는 액터와 시스템이 수행해야하는 처리기능인 사용 사례를 추출하여, 액터와 사용사례 간의 관계를 나타낸 것이다. 1개의 액터와 8개의 사용사례를 추출하여 표현 하였다. 행정부서인 연구 관리팀에서 과제등록 및 과제변경을 등록 관리한다. 과제관리는 크게 과제를 등록하는 과제등록관리 및 변경을 관리하는 과제변경관리로 나눌 수 있다.



(그림 3). 과제관리 사용사례 다이어그램

3.3 순서 다이어그램

순서 다이어그램에서는 액터에 의해 시스템을 이용하는 과정을 순서대로 표현한 것으로, 가로방향의 객체와 세로방향의 진행시간에 따른 시스템의 동작을 표현하고 있다. 순서 다이어그램은 크게 과제등록관리 및 과제변경관리로 나누어 구성 하였다.

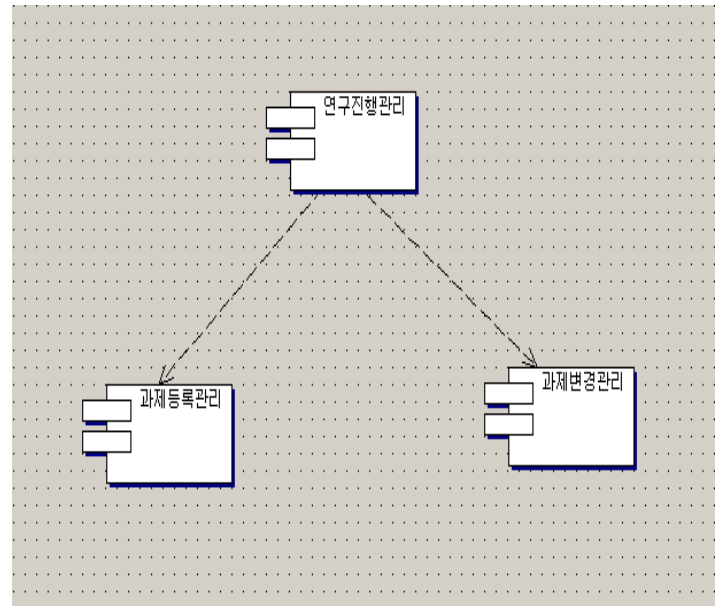
첫 번째로 과제등록관리 순서 다이어그램은 연구 관리자인 액터가 과제등록메인화면에서 사업신청서 내용을 호출하여 기본정보를 확인 후 참여기업, 참여연구원, 연구비현황, 및 과제요약서 등을 등록한다.

두 번째로 과제변경관리 순서 다이어그램은 연구 관리자인 액터가 변경관리화면에서 해당 과제에 대한 소요예산, 연구기간 및 과제책임자를 추가 및 수정, 삭제 한다.

3.4 객체 모델링 및 컴포넌트 추출

업무분석을 통하여 액터와 사용사례를 추출하였다. 또한 클래스 다이어그램을 통하여, 클래스를 추출하였으며, 순서 다이어그램을 이용하여 액터에 의해 시스템을 이용하는 과정을 추출하였다. 클래스 중에서 Entity 클래스의 Attribute는 데이터 모델을 전환하는데 중요한 역할을 한다. Entity 클래스의 Attribute는 데이터 모델에서 데이터 컬럼의 역할을 하여 데이터 모델을 산출 한다.

마지막 컴포넌트 도출 단계는 지금까지 추출했던 사용사례 다이어그램 및 클래스다이어그램 그리고 순서다이어그램을 이용하여 컴포넌트를 구성한다. 컴포넌트를 작성하는 최고의 목표는 자주 사용되는 클래스 및 메소드를 부품화 하여, 프로그램 작성 시 컴포넌트를 쉽게 사용함으로써 생산성을 높이고자 하는데 목적이 있다. 그림 4은 과제관리의 컴포넌트를 다이어그램으로 나열한 것이다. 과제관리는 과제등록관리 컴포넌트 및 과제변경관리 컴포넌트로 구성하였다.



(그림 4). 과제관리 컴포넌트 구성

4. 결과분석

지금까지 컴포넌트 기반의 RUP 모델을 적용하여 연구관리 시스템의 주요 기능을 컴포넌트로 실험하였다. 기존의 구조적 분석/설계 방법을 적용해서 만들어진 기능들은 프로세스 중심으로 구현되어 있기 때문에 요구사항에 유연하게 대처 할 수 없으므로 유지보수에 많은 비용이 발생하였고, 또한 컴포넌트 단위로 구현되어있지 않아 재사용성의 결여와 개발에 많은 시간과 비용이 발생 하였다. 이런 문제점을 극복하기 위한 객체지향 방법은 문제로부터 객체의 추출과 객체의 속성과 객체에 적용할 연산의 캡슐화를 통하여 복잡한 실세계의 표현을 단순화시키고, 주요 모듈을 부품화 함으로써 확장성과 재사용성이 용이하고, 컴포넌트 단위로 구현되기 때문에 시스템 변화에 유연하게 대처 할 수 있어 유지보수가 용이하다.

5. 결론

논문에서는 연구관리 시스템의 주요 기능인 과제관리를 과제 등록관리 및 과제 변경관리로 분류하여 컴포넌트 기반의 RUP 모델을 사용하여 설계 및 구현 하였다.

실험에서는 컴포넌트들의 구현에 표준 웹 플랫폼인 J2EE 환경에서 RUP 모델을 적용하였다. DBMS로는 Oracle 9i를 사용하였으며, 요구분석 및 설계 도구로는 Together 2005를 사용하였다. 또한 미들웨어 WAS로는 Resin2.1.17을 사용하였다. 실험 데이터로는 연구관리 시스템으로부터 추출한 데이터(과제 마스터정보, 참여연구원 정보, 연구기관 정보, 과제변경 정보)를 사용하였으며, 기능별로 도출된 컴포넌트를 RUP 모델에 적용하였다.

시스템의 기능들을 컴포넌트 단위로 구현하여 그 결과를 분석 하였다. 과제 등록관리 에서는 컴포넌트의 호출을

통해 과제 기본정보를 등록 및 조회 할 수 있었으며, 과제 변경관리에서는 과제 진행에 따른 변경내용을 등록 및 조회 할 수 있었다.

컴포넌트 기반의 RUP 모델을 이용하여 연구관리 시스템을 구현한 결과 다음과 같은 결론을 얻을 수 있었다.

첫째, 요구사항을 사용사례 및 액터별로 추출하여 사용자가 이해하기 쉬운 형태의 다이어그램으로 표현함으로써 사용자와의 긴밀한 의사교환(communication)이 가능하다.

둘째, 소프트웨어 개발 프로세스의 품질향상이 가능하다. 이는 중복된 프로세스들을 하나의 프로세스로 컴포넌트화 함으로써 데이터의 신뢰성 및 컴포넌트의 성능을 최적화 할 수 있다.

셋째, 소프트웨어 개발 기간의 단축과 유지보수의 효율성이 향상 되었다. 컴포넌트를 이용함으로써 시스템 개발 기간이 단축되었으며, 오류 발생으로 인한 복잡한 유지보수가 필요할 경우 컴포넌트를 수정함으로써 전체적인 시스템에 영향을 미치지 않는다.

CBD 기반의 RUP 모델은 사용사례 크기에 매우 종속적이다. 따라서 무엇보다도 요구사항에 적합한 사용사례의 도출이 매우 중요하다. 적은 수의 사용사례를 도출할 경우는 요구사항을 충족시키지 못하는 시스템이 될 수 있으며, 많은 수의 사용사례를 도출할 경우는 보다 많은 컴포넌트가 생성되므로 소프트웨어의 개발과 유지보수에 어려움이 따른다. 따라서 요구사항에 적합한 사용사례 수의 도출 문제와 사용사례들 간에 공통적으로 사용하는 기능들을 별도의 컴포넌트로 식별하여 명세화 할 수 있는 방법에 대한 연구가 필요하다.

참고문헌

- [1] 이경환, 최신 소프트웨어 공학, 청문각, 1998.
- [2] 한국소프트웨어산업협회, "해외 S/W산업 시장/정책 현황 동향분석", 2000
- [3] 최영근, 허계범, 객체지향 소프트웨어공학, 한국실리콘, 1998.
- [4] 박병형, 블록놀이와 CBD, 태영, 2002.
- [5] Alan, W.Brown and Balbir Barn, "Enterprise-Scale CBD Building Complex Computer Systems From Components", Sterling Software, 1999.
- [6] D. Chappell, "On COM ActiveX vs. Java Beans", Object Magazine, January 1998.
- [7] Object Management Group, CORBA Components - Volume 1, Joint Revised Submission, OMG TC Document, August 1999.
- [8] Enterprise JavaBeans, at URL: <http://www.javasoft.com>, <http://www.java.sun.com>
- [9] Butlergroup "Component-Based Development", Butlergroup Management Guides, Sept., 1998.
- [10] Ivar Jacobson, Grady Booch, James Rumbaugh, The Unified Software Development Process,

Addison-Wesley, 1999