

비즈니스 프로세스 모델과 컴포넌트 기반 개발의 매핑에 관한 연구

서윤숙*, 김재수**, 김영철
홍익대학교 컴퓨터정보통신
한국과학기술정보연구원 표준화기술지원실**
e-mail: {jyun*, bob}@selab.hongik.ac.kr
kjaesoo**@kisti.re.kr

A Study on Mapping Business Process Model onto Component Based Development

Yun-Suk Seo*, Jae-Soo Kim**, R. Young-Chul Kim
Dept. of CIC, Hongik University*
Technical Support for Standardization Dep. Korea Institute
of Science and Technology Information**

요 약

본 논문은 BPM(Business Process Model)에 룰(rule/policy)을 적용하여 3-Tier 구조의 BPM을 제안하고자 한다. 이는 비즈니스 프로세스를 효율적으로 관리하고, 빠르게 변화하는 업무를 비즈니스 프로세스 모델링하고 그에 필요한 응용 프로그램의 개발을 위해 BPM과 CBD(Component Based Development)간의 접목에 초점을 둔다. 이를 통해서 새로운 비즈니스 프로세스를 추가/변경 시, 필요한 응용업무 프로그램 개발 기간 단축을 할 수 있으리라 본다. 은행 비즈니스업무에 대해 비즈니스 프로세스 모델링을 적용 사례로 보여준다.

1. 서론

현재 빠르게 변화하는 비즈니스 업무에 대처할 수 있는 비즈니스 프로세스가 필요하며, 소프트웨어 공학적 측면에서도 효율적인 소프트웨어개발과 재사용을 통해 신속한 프로그램 개발을 위해 컴포넌트 기반의 개발 방법이 대두되고 있다[1,2]. 그러나 기존 비즈니스 업무 분석을 위한 BPM(Business Process Modeling)과 소프트웨어 개발을 위한 CBD(Component Based Development)에는 서로 괴리가 존재하여, 각각의 비즈니스 프로세스 모델과 소프트웨어개발 분야가 따로따로 발전되고 있다.

우리는 빠르게 변화하는 업무에 대하여 비즈니스 프로세스를 모델링하고 그에 필요한 응용 프로그램 개발을 위한 CBD를 접목하고자 한다. 이는 더 효율적인 비즈니스 프로세스 관리와 그에 필요한 응용 프로그램 개발을 위해 BPM에 규칙(rule/policy)을

적용하고 3-Tier 구조의 BPM으로 새로운 소프트웨어를 개발 때 CBD와 매핑을 통해 재사용성이 높이고자 합니다.

본 논문 2장에서는 관련연구에 대해 기술하고, 3장에서는 BPM과 CBD접목 그리고 3-계층 구조에 대해 설명하고, 4장은 적용사례를 언급한다. 마지막 장에서는 결론 및 향후 과제에 대해 기술한다.

2. 관련 연구

전반적인 업무 프로세스를 체계적으로 관리하기 위한 BPM은 모든 프로세스와 프로세스 관리 시스템을 위한 표준이 정의되고 다른 기술과의 통합성을 보장하며 프로세스 모델링 환경과 프로세스 실행 환경을 동시에 가능하게 하고 있다[1]. 다시 말하면 BPM의 핵심 역량은 프로세스를 발견, 설계, 적용, 실행, 상호작용, 운영, 최적화 및 분석하는 종합적인

역량이며, 프로세스의 전 생명주기를 완벽하게 제어할 수 있다. 그리고 소프트웨어 공학에서의 CBD는 시스템을 개발 할 때 컴포넌트 기반의 개발을 통해 컴포넌트의 재사용에 초점을 두고 있다[2].

즉, BPM을 효율적으로 활용하기위해 CBD방법론으로 개발된 컴포넌트들과의 매핑을 시도한다[4]. 여러 개의 공통된 컴포넌트들을 연결(묶은)하는 워크플로우 즉, 하나의 업무 단위 프로세스들로, 이 프로세스들을 엮어서 비즈니스 프로세스들로 매핑된다. 역시 새로운 비즈니스 프로세스를 수정/추가 시 같은 매카니즘으로 수행된다.

이는 컴포넌트와 비즈니스 프로세스들로 다른 새로운 시스템을 개발 할 때 재사용/재배치를 통해 변화에 대한 대처능력이 뛰어난 비즈니스 프로세스를 만들 수 있을 것이다[5].

또한, 제임스 오델[6,7]은 객체 상에 룰(rule)을 표현하는 방법을 제안하였다. 우리는 BPM을 효율적으로 제어하기 위해 비즈니스 프로세스 모델에 오델의 룰(rule/policy)을 적용한다[5]. 시나리오 기반의 업무 프로세스를 제어하는 룰을 적용하여, 전반적인 비즈니스 프로세스를 효율적으로 관리와 더불어 확장된 BPM의 더 효과적인 활용을 위해서 3-계층 구조를 제시하고 하위 레벨의 컴포넌트 레퍼지토리의 컴포넌트와 매핑을 시도한다.

우리는 룰 기반의 BPM을 통해 비즈니스 프로세스 모델링을 할 때 비즈니스 프로세스 내에 정책과 조건 등의 제약 사항을 통해 효율적으로 비즈니스 프로세스 모델링이 가능하고, 모델링이 더 쉬워질 수 있다. 만약 모델링을 수정하게 된다 할지라도 복잡해지거나, 다시 모델링 하는 번거로움을 없앨 수 있을 것이고 룰을 적용함으로써 섬세한 비즈니스 프로세스를 제어 할 수 있는 장점을 얻을 수 있으리라 본다.

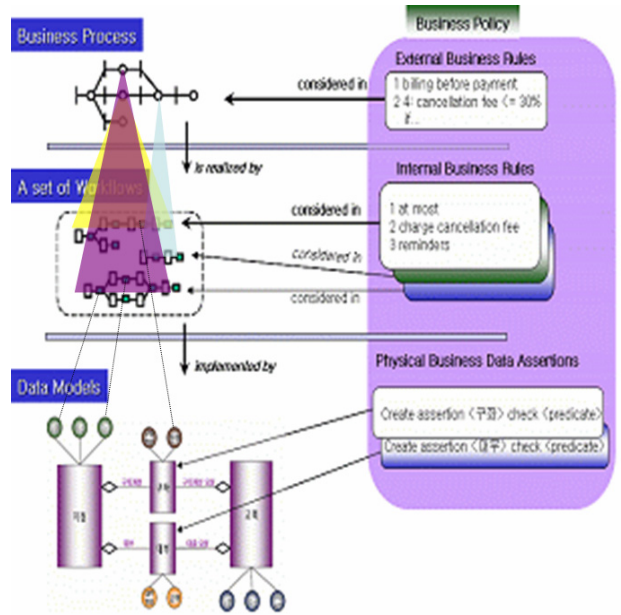
3. BPM과 CBD의 접목

우리는 BPM과 CBD의 접목을 위해 3-계층 구조를 제시하고 하위 레벨의 컴포넌트 레퍼지토리의 컴포넌트와 매핑의 시도를 제안하려 한다.

3.1 BPM상의 3-계층 구조

3-계층 구조의 윗부분인 비즈니스 프로세스 레이어는 비즈니스 프로세스 액티비티들의 집합으로 이

루어지고 이러한 액티비티들 수행에 있어 제한을 둔다[5]. 비즈니스 룰의 형식인 조직의 비즈니스 정책을 살펴보면, 외부 비즈니스 룰은 조직이 수행할 총체적인 룰의 구조를 정의하는 것이고 내부 비즈니스 룰은 조직이 실제적으로 업무를 수행할 때의 룰을 정의한다. 마지막 물리적인 비즈니스 데이터 Assertion에서는 실질적인 데이터 모델이 이루어지는 것을 정의한다. 우리는 내부 비즈니스 룰을 프로세스 명세에 정의한다. 정책이 고객 정보와 함께 데이터베이스로 관리되는 것과 같이 내부 비즈니스 룰은



<그림 1> 3-계층 구조의 기본적 개념

프로세스 명세에 상술된다. 3-계층의 중간 부분인 워크플로우 레이어는 특정한 조직에서 현재의 비즈니스 사례들이 어떻게 다루어지는지 나타내며 일련의 워크플로우로 구성된다.

3-계층 구조의 하위 계층인 데이터 모델 레이어는 각각의 워크플로우를 실질적인 데이터 모델로 구현한 것이다.

다시 말하자면, 상위레벨의 비즈니스 프로세스계층에서는 여러 개의 비즈니스 프로세스가 존재한다. 상위레벨에서 중복된 비즈니스 프로세스를 찾아 새로운 비즈니스 업무를 만들 때 비즈니스 프로세스를 재사용하여 쉽게 만들 수 있다.

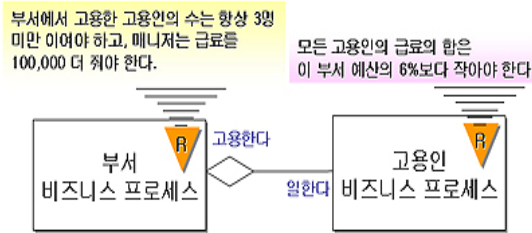
결국 상위 레벨의 단위프로세스는 중간 레벨의 워크플로우로 구성(매핑)되고, 워크플로우내의 컴포넌트들은 하위 레벨의 컴포넌트 레퍼지토리에 구성(참조/매핑)되어 진다.

그리고 <그림 1>에서처럼, 모든 레벨 상에서도 룰

을 적용하고자 한다[8].

<그림 2>은 실제 프로세스 내에 룰을 표현하는 방법을 보여준다.

부서 비즈니스 프로세스와 고용 비즈니스 프로세스에 <그림 2>와 같은 룰을 표현하기 위해 각 프로세스 내에 룰을 정의하여 기술한다. 룰이 존재하는 프로세스는 <그림 2>와 같이 "R"로 표시하고 우리가 개발한 도구인 BPMSA에 적용하였다.



<그림 2> 프로세스 내 룰 표현

중간레벨의 워크플로우 계층에서는 여러 개의 워크플로우들이 존재한다. 이러한 워크플로우들은 많은 컴포넌트들로 구성되어있다. 이러한 컴포넌트들을 CBD의 Domain 분석 방법론으로 중요한 컴포넌트를 추출하여 재사용한다면 새로운 응용 소프트웨어 개발을 용이하게 한다.

하위레벨의 데이터모델링 계층은 중간레벨의 워크플로우들을 구현하는 단계이다. 컴포넌트들의 레파지토리에 구축되어 보관 또는 관리되는 하부 계층이다. 이 계층은 관련 컴포넌트들을 관리하는 데이터베이스, 즉 컴포넌트 레파지토리아다

우리가 개발한 3-tier 구조의 BPM 메카니즘 도구인 BPMSA를 간략히 소개한다.

첫 번째, 이 도구는 <그림 3>에서 보는 것과 같이 비즈니스 프로세스 내 룰을 적용하였다.

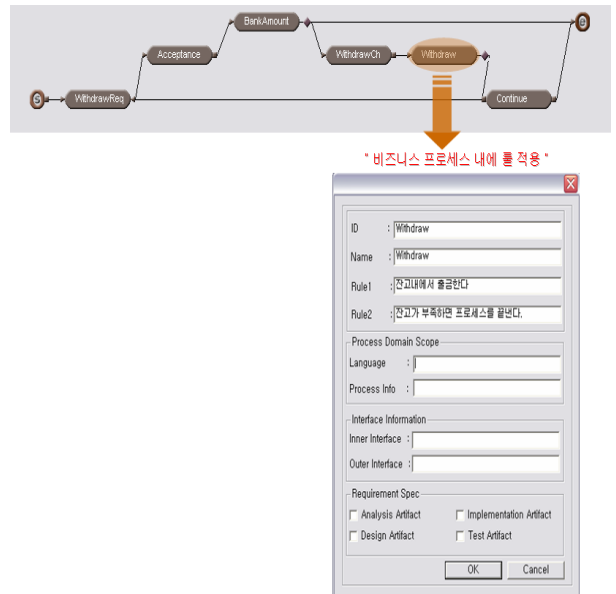
각각의 프로세스 내에 있는 룰을 보기 위해서는 각각의 프로세스를 두 번 클릭하면 프로세스 스펙을 볼 수 있다. 위의 그림은 "Withdraw"프로세스 내에 정의한 룰을 보여준다.

"Withdraw"프로세스의 룰을 살펴보면, 첫 번째 룰은 자신의 잔고를 확인하고, 두 번째 룰은 잔고 내에서만 출금이 가능하며, 만약에 잔고가 부족하다면 "Withdraw"프로세스를 종료하는 제약사항을 확인할 수 있다.

BPMSA도구에서와 같이 우리가 구현한 BPM도구에서 룰의 정의를 프로세스 속성 내에 적용할 수 있도록 표현하였다. 이처럼 각각의 프로세스에 선(先) 처리해야할 룰을 기술해준다.

두 번째, 이 도구에서 모든 레벨에서, 즉 비즈니스 프로세스내의 공통/비공통 프로세스, 워크플로우내에서의 공통/비공통 컴포넌트들의 중요성 및 빈도수를 자동으로 구해 준다. 이것을 토대로 새로운 비즈니스 프로세스와 그 프로세스의 워크플로우를 생성시에 재사용을 가능하게 해주는 가이드를 보여 줄 것이다.

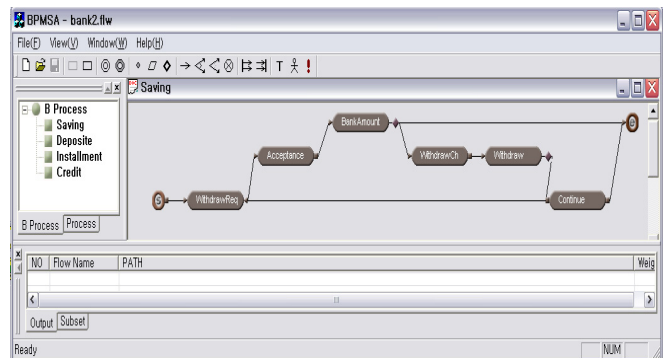
그리고 또한 프로세스/워크플로우내의 패턴들을 그룹핑과 모든 경로를 찾아줘 중요한 경로를 결정해 줄 수 있다. 이것은 새로운 비즈니스 프로세스에 대한 리팩토링 기법을 적용하려는 의도가 있다.



<그림 3> BPMSA 도구를 이용하여 구현한 비즈니스 프로세스 내 룰 적용

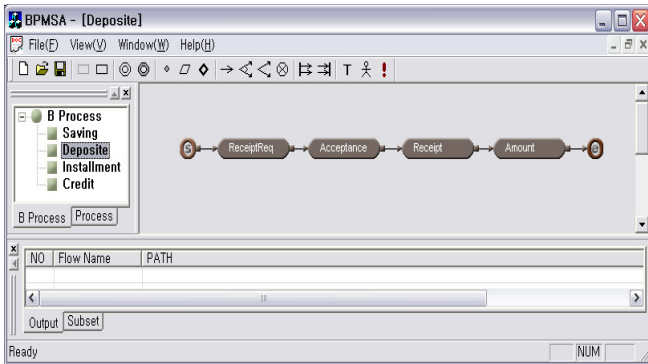
4. 적용사례

우리는 BPM에 관련된 도구를 만들어 이 도구를 이용하여 비즈니스 프로세스를 구현을 모델링 하였다. <그림 4>은 입금 및 출금 비즈니스 프로세스의 워크플로우를 BPMSA의 도구로 모델링 한 예이다.



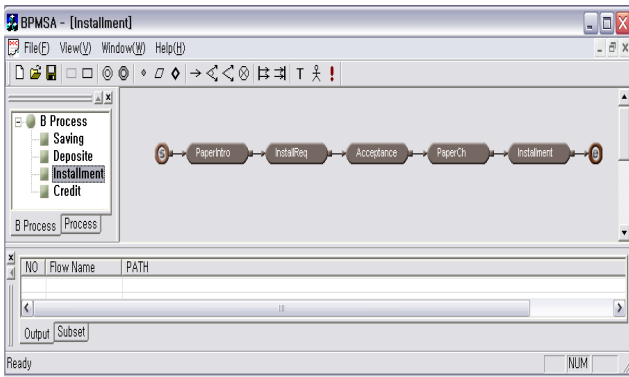
<그림 4>BPMSA도구로 구현한 은행업무의 입금및출금

<그림 5>는 적금 비즈니스 프로세스의 워크플로우를 BPMSA의 도구로 모델링 한 예이다.



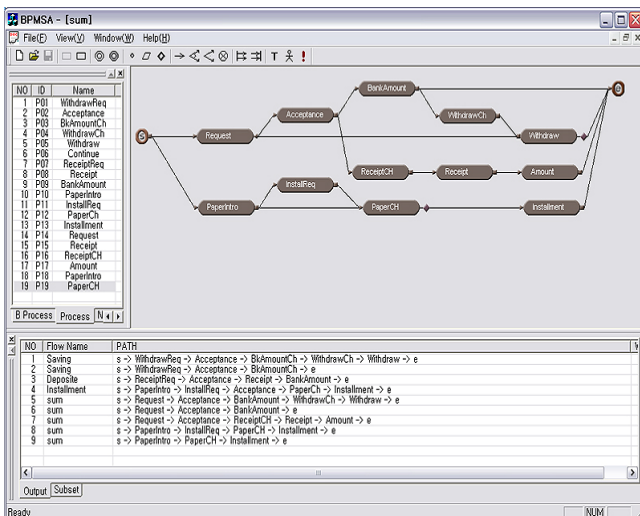
<그림 5> BPMSA도구로 구현한 은행업무의 적금

<그림 6>는 은행업무의 통장 개설 비즈니스 프로세스의 워크플로우를 BPMSA의 도구로 모델링 한 예이다.



<그림 6> BPMSA도구로 구현한 은행업무의 통장 개설

<그림 7>은 통합 은행 업무 비즈니스 프로세스의 워크플로우를 BPMSA의 도구로 모델링 한 예이다.



<그림 7> BPMSA도 구로 구현한 통합 은행 업무 프로세스

5. 결론

BPM은 업무의 단위가 프로세스 단위로 이루어지기 때문에 변화에 잘 적응할 수 있는 장점을 갖고 있다. 이 프로세스는 컴포넌트와 같은 역할을 하기 때문에 새로운 업무 개발 시, 재사용하여 개발 시간과 비용을 절약할 수 있는 이점을 갖고 있다. 또한, 비즈니스 프로세스가 동적으로 변화하는 업무에 잘 적용할 수 있도록 모델링 할 수 있다.

이러한 비즈니스 프로세스를 효율적으로 관리 및 통제를 하기위해 룰을 적용하였다.

앞으로 우리가 제안한 3-tier 구조의 각 계층을 제어할 수 있는 BPSQL(Business Process modeling SQL)을 제안하여, 계층 간의 연관된 자료들을 효율적으로 관리할 것이며, BPM과 CBD간의 매핑도구를 개발 중이다. 또한 하위 레벨의 컴포넌트 레파지토리의 관계 테이블화에 대해 연구가 진행되고 있다.

6. 참고문헌

- [1] BMP 2004 FORUM, "BMP 2004 FORUM", 2004.
- [2] Howard Smith. "Business Process Management". 시그마인사이트컴. 2004
- [3] 삼성 SDS, "Innovator CBD 방법론" 2002.
- [4] 김영철, 전병국, 김윤정. "워크플로우 메커니즘을 통한 소프트웨어 컴포넌트 식별에 관한 연구", 한국정보처리학회 추계학술발표대회 논문집 제10권 제2호, 2003.
- [5] 서윤숙, "확장된 BPM과 컴포넌트 기반 방법론 매핑에 관한 연구", 2005.2
- [6] James Odell. "Modeling Control in Rule-based Systems". JOOP. 1993.
- [7] James Odell. "Using Business Rules with Diagrams". JOOP. 1993.
- [8] Bichler, P. Preuner, G. Schrefl, M. "Workflow Transparency". Conference on Advanced Information Systems Engineering, 1997.