

# 시맨틱 웹과 웹 서비스 기반의 비즈니스운영체계 아키텍처

## A Business Operating System Architecture based on Semantic Web and Web Service

최미영, 방찬석, 권정민, 최인준  
포항공과대학교 산업공학과

### Abstract

True process collaboration can be accomplished through seamless integration of business processes and enterprise knowledge. Therefore, it is natural that the concept of Business Operating System (BOS), proposed by Delphi Group in 1994, is currently considered as a next evolutionary step for Business Process Management System (BPMS). Literature reports very little work, however, especially, on a comprehensive architecture of the system. This paper proposes an architecture of BOS with the following definition: 'BOS is an intelligent activity supporting system that provides a comprehensive and personalized work environment to each knowledge-intensive worker.' To propose an architecture of BOS, the paper first identifies and classifies functional requirements for Business Operating System. Then, it proposes a data model and an architecture of the system to satisfy the functional requirements. The proposed architecture focuses on two essential technical requirements. First, the system should provide an effective means to integrate data and processes and to standardize distributed component systems. Secondly, the system should also be intelligent enough to assist workers to perform their knowledge-intensive work. The paper shows how these requirements can be achieved by using Semantic Web and Web Service.

### 1. 서론

오늘날 기업의 시스템 구축에 있어, 대부분의 기업이 직면한 가장 큰 문제는 시스템통합이다. 실제로 시스템통합과 관련된 기술적인 연구, 개발 노력들이 많이 있어왔다. 대표적으로 ERP(Enterprise Resource Planning), EAI(Enterprise, Application Integration), 워크플로우(workflow), 비즈니스 프로세스 관리(Business Process Management; BPM) 등이 시스템 통합에 활용되고 있다. 최근에는 이와 같은 시스템통합과 관련된 기술들이 BPM으로 흡수되는 추세이다. 비즈니스 프로세스(Business Process)란 기업이 타 기업과 개인 고객을 포함한 모든 파트너에게

가치(서비스, 제품 등)를 제공하기 위해 순차적이거나 동시적으로 발생할 수 있는 모든 업무들의 집합을 의미한다[9]. BPM은 기업 내 외부에서 발생하는 전체 비즈니스 프로세스를 제대로 구축하고 관리하기 위한 관련 기술의 집합체이다. 기업들은 BPM이라는 기업 중심의 시스템통합을 바탕으로 점진적인 기업정보화 및 업무자동화를 이루고 있다.

그러나 기업 중심의 시스템통합이 상당부분 이루어지고 있는 현 시점에서는 효율적인 업무수행지원을 위한 통합적인 정보시스템과 체계적인 지식관리, 그리고 효율적인 업무수행을 지원하는 체계로서의 비즈니스 프로세스의 지적 자산화가 기업정보화의 최대 관심사항이다. Koulopoulos[7]는 기업 중심의 시스템통합을 넘어서 각 기업별로 분산되어있는 프로세스들을 업무수행자 관점에서 완전히 통합시킬 수 있는 새로운 데스크탑 환경인 비즈니스운영체계(Business Operating System; BOS)로의 진화를 제안하였다. 그가 제안한 BOS는 각기 다른 어플리케이션에서 수행되던 분산된 프로세스들을 통합시켜, 각 프로세스에 속해 있는 업무를 업무수행자가 하나의 공통된 인터페이스를 통해 수행할 수 있도록 한다. 그가 언급하는 BOS는 그룹웨어나 워크플로우를 내장한 거대한 지식창고로서 포괄적인 업무환경을 제공해준다[7]. 그러나 Koulopoulos의 제안에서는 개념적인 부분만이 막연하게 제시되고 있을 뿐, 구체적인 아키텍처나 기능이 제시되지 않았다.

따라서 본 연구에서는 조직 내외의 여러 시스템에 분산되어 있는 업무 및 지식을 통합하고 사용자의 모든 행정업무처리를 대신하거나 도와줌으로써, 맞춤화되고 포괄적인 업무수행환경을 단일 인터페이스로 제공해주는 업무수행지원시스템인 비즈니스운영체계(BOS)의 아키텍처를 제시하고자 한다. 그 중에서도 특히 지식 집약적인 업무수행을 지원해줄 수 있는 기능을 제시하는 데에 초점을 맞추었다. BOS를 구현하기 위해서는 크게 두 가지 관점에서 어려움이 있다. 첫째, 분산되어 있는 각기 다른 시스템들로부터 업무를 받아서 수행하기 위해서는 데이터 및 프로세스를 통합하고 분산객체기술을 표준화해야 한다. 본 연구에서는 온톨로지 언어로 프로세스와 데이터에 대한 의미적 연관성을 지음으로써 통합이 가능함을 제시하였다. 또한 HTTP 표준 프로토콜을 사용하는 웹 서비스를 통하여 프로그래밍 언어나 플랫폼에 독립적인 분산처리시스템간의 상호작용이 가능하도록 하였다. 둘째, 업무수행자에게 맞춤화된 업무수행환경을 제공하기 위해서는 BOS가 지능적인 시스템

이 되어야 한다. 본 연구에서는 기존의 웹을 시맨틱 웹으로 대체하여 사용함으로써 BOS가 지능적으로 업무수행자에게 적합한 업무를 협상해오고, 업무와 관련된 지식을 찾아오며, 업무수행자 정보를 시스템 스스로 외부에 홍보할 수 있도록 하였다.

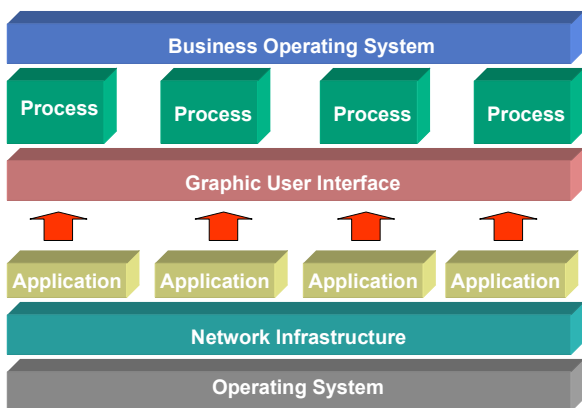
본 논문의 구성은 다음과 같다. 2 장에서는 본 연구와 관련된 기존연구를 살펴보고 3 장에서는 BOS의 요구사항을 분석하고 아키텍처를 제시한다. 4 장에서는 프로토타입 시스템에 대한 설명과 그 수행 과정 및 결과를 살펴보고 5 장에서는 결론과 추후 연구방향을 제시한다.

## 2. 관련연구

이 장에서는 BOS와 EIP에 대해 알아보고, 본 연구에서 적용하고 있는 분산환경기술인 시맨틱 웹과 웹 서비스를 살펴본다.

### 2.1. BOS와 EIP

BOS는 비즈니스 프로세스 업무를 지원하기 위한 차세대 운영체계를 지칭하는 말로 [그림 1]과 같은 구조를 가지고 있다. BOS가 구체화되기 시작한 것은 1998년 메릴린치에서 기업정보포털(Enterprise Information Portal; EIP)이라는 용어가 제안되면서부터였다[8]. EIP는 Corporate Portal(CP), Enterprise Portal(EP) 등의 명칭으로도 사용되며, 정보의 급증에 따라 필요한 정보를 검색하는데 어려움이 증가하면서 나타난 개념이다[8]. 그러나 EIP는 기업 중심의 지식관리이기 때문에 업무수행자를 위한 업무지원 기능이 미약하다.



[그림 1] BOS의 구성요소

### 2.2. 분산환경기술

시맨틱 웹은 기계가 정보의 의미를 이해하고 이를 바탕으로 논리적 추론이 가능하도록 함으로써 사람의 개입 없이 기계들 사이에 커뮤니케이션을 가능하게 한다 [5]. 온톨로지란 용어 사이의 관계를 정의하고 있는 일종의 사전과 같은 것으로 시맨틱 웹에서 가장 중심에 있는 개념이다. 대표적인 RDF/RDF-S 기반의 온톨로지 언어로는 DAML(DARPA Agent Markup Language)+OIL(Ontology Inference Layer), OWL(Ontology Web Language) 등이 있다.

이렇게 정의된 문서는 추론 도구를 활용하여 의미적인 해석이 가능하다. [6, 10].

웹 서비스는 분산 컴퓨팅을 위한 최신 기술로 표준 인터넷 기술들을 이용하여 동적으로 상호작용하는 소프트웨어 컴포넌트라고 정의할 수 있다.

본 연구에서는 OWL을 이용하여 웹으로 공개되는 데이터 정보들을 정의하였다. 프로토타입에서는 OWL 추론도구인 Pellet API[10]를 사용하여 OWL로 정의한 파일을 처리하였고, 웹 서비스 기술을 이용하여 분산된 두 시스템 간에 메시지 교환을 하는 에이전트를 구현하였다.

## 3. BOS의 아키텍처 및 데이터 모델

이 장에서는 BOS의 요구사항을 분석하고, 이를 기초로 시스템의 아키텍처를 제시한다. 3.1 장에서 BOS가 갖춰야 할 요구 사항 분석 시 고려해야 할 항목에 대해서 설명한다. 3.2 장에서 3.1 장의 항목을 고려해 분석하여 설계된 시스템의 데이터 모델을 설명하고, 3.3 장에서는 요구사항을 충족시키는 시스템의 아키텍처와 세부 구성요소를 분석한다.

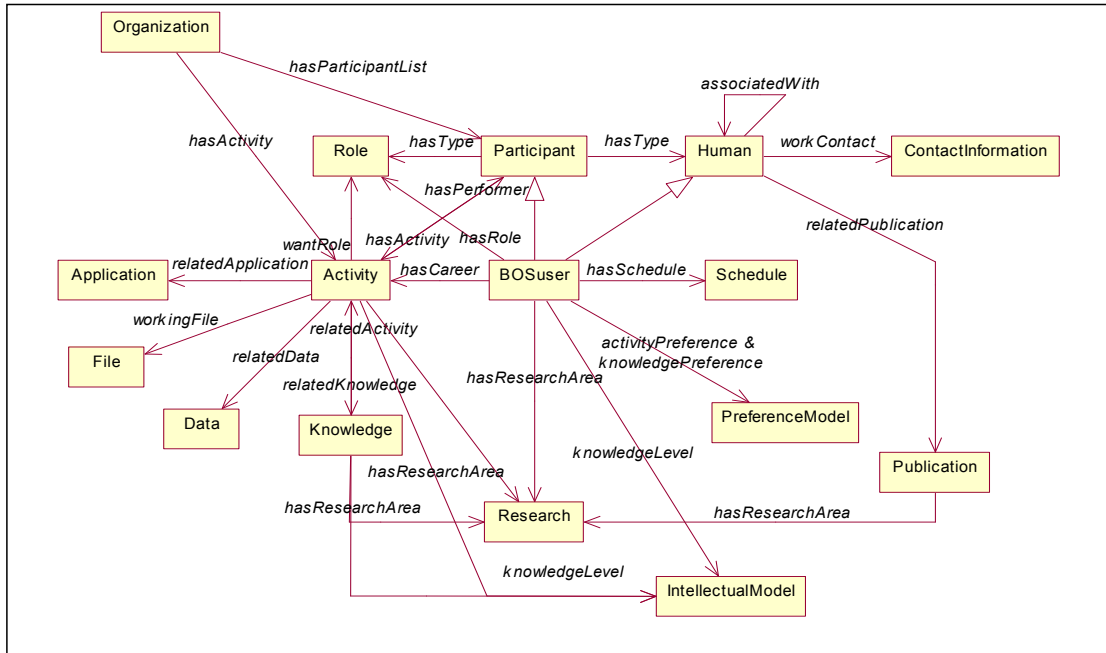
### 3.1. 요구사항 분석

비즈니스 업무지원과 관련된 기존 연구들[1-4]을 바탕으로 BOS에 적합한 기능적 요구사항들을 추출하여 업무관리, 지식관리, 업무수행자 정보관리, 스케줄 관리의 네 가지 항목으로 정리하였다. 각 항목의 고려 내용은 다음과 같다.

- ✓ 업무관리에 대한 요구사항  
 업무관리에 대한 요구사항들은 크게 세 가지 범주로 구분된다. 첫째, 업무목록에 대한 전반적인 관리에 대한 요구사항(Work & work list management), 둘째, 업무수행 중에 발생하는 요구사항(Supporting execution of activity), 셋째, 외부의 조직에서 할당 받을 업무의 협상과정에 대한 요구사항(Work item negotiation to select a proper work for an actor)이다.
- ✓ 지식관리에 대한 요구사항  
 지식관리에 대한 요구사항으로 업무와 관련된 지식관리(Activity related knowledge management)와 업무와 관련되지 않은 지식관리(Knowledge management for future work & improving one's capability)에 대한 요구 사항을 모두 고려한다.
- ✓ 업무수행자 정보관리에 대한 요구사항  
 업무수행자 개인정보를 관리하는 요구사항으로, 자신의 프로파일들을 공유하고 자신의 이력을 관리하기 위한 기능들을 고려한다.
- ✓ 스케줄 관리에 대한 요구사항  
 스케줄 관리에 대한 요구사항은 업무시간을 포함한 업무수행자의 전체적인 시간을 관리해주기 위한 요구사항이다.

에 대해 어느 정도의 지식수준을 가지고 있는가에 대한 척도이다.

### 3.2. BOS의 데이터모델



[그림 2] 전체 데이터 모델에 대한 Class diagram

#### 3.2.1. BOS의 전체 데이터모델

[그림 2]에서는 BOS의 요구사항을 바탕으로 시스템에 필요한 데이터모델을 도식화하였다. BOS를 사용하는 업무수행자인 BOSuser 클래스를 중심으로 데이터모델을 정의하였다.

BOSuser 클래스는 Human 클래스와 업무를 수행하는 주체인 Participant 클래스의 속성을 모두 갖고 있기 때문에 그 둘의 하위클래스로 정의하였다. IPM-PDL에 따르면, Participant는 업무를 수행하는 사람, 역할, 시스템, 조직 단위 등을 말한다[6]. BOSuser는 Participant로부터 상속받은 hasActivity라는 속성뿐만 아니라 hasCareer라는 속성도 갖는다. 이 두 속성은 모두 Activity 타입을 갖는다. 전자는 기존에 했던 활동들 중 경력으로 취급되어 외부에 홍보할 수 있는 성질의 업무이고, 후자는 현재 자신이 행하고 있는 개인적인 업무를 포함한 활동들을 정의한다. 업무수행 후에 업무정보를 경력으로 바꾸어주기 위해서는 hasActivity에 있던 업무정보를 그대로 경력정보인 hasCareer로 저장하면 된다. BOSuser는 또한 hasRole, hasResearchArea, hasSchedule, knowledgeLevel, activityPreference, knowledgePreference 등의 속성을 가지고 있다.

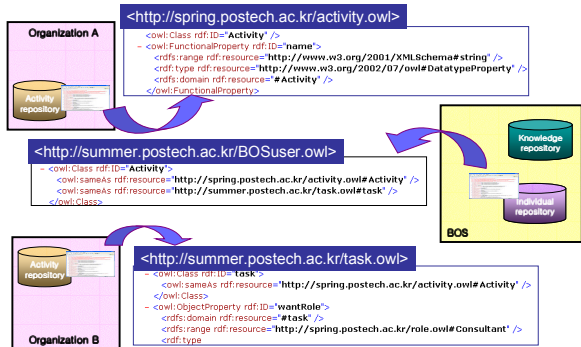
Activity 클래스의 경우에는 업무의 특징을 나타내는 wantRole, hasResearchArea, knowledgeLevel이라는 속성과 그 업무를 수행하는데 필요한 존재를 나타내는 hasPerformer, relatedApplication, relatedKnowledge, workingFile, relatedData 등의 속성을 가지고 있다. knowledgeLevel은 업무수행자와 관련이 있는 척도로, 이 업무

기존 프로세스 정의에서와는 달리, 본 연구에서는 Knowledge 클래스도 hasResearchArea, relatedActivity 등의 속성을 갖는 업무와 독립적인 존재로 정의하였다. 이런 속성을 바탕으로 좀 더 정확한 지식검색이 가능해지고, 상황에 따라 관련업무와 유연한 결합을 할 수 있다. 또한 하나의 지식이 여러 종류의 업무에 사용될 경우에 대한 중복성 문제도 해결된다.

#### 3.2.2. 데이터 표현 및 저장

본 연구에서는 비공개 정보를 데이터베이스로 관리하고, 공개 정보를 시맨틱 웹 언어 중 현재 가장 표현력이 좋은 OWL로 표현하였다. OWL은 이미 정의되어 있는 데이터모델을 재사용할 수 있을 뿐만 아니라, 정의한 클래스들간의 의미적 연관성을 지을 수 있다는 장점이다. 예를 들어 XML의 경우에는 같은 의미를 나타내는 개체라 하더라도 다른 이름으로 정의를 해 놓으면 데이터간 호환이 되지 않는다. 물론, 전 세계의 모든 기업들이 하나로 통일된 용어만 사용한다면, 이런 문제는 발생하지 않을 것이다. 그러나 나라마다 사용하는 언어만 해도 2000개가 넘는 현 상황에서 모든 용어를 하나로 통일한다는 것은 불가능하다. 그러나, OWL을 사용하게 되면, owl:sameAs라는 태그를 사용함으로써 간단하게 다른 태그로 정의해놓은 두 클래스가 같다는 것을 표현해줄 수 있다. 예를 들어 [그림 3]과 같이 업무를 나타내는 태그가 “Activity”와 “task”의 두 가지로 사용되는 경우 그림에서 나타내듯이 의미적 연관성을 지어주면, 사람뿐만 아니라 컴퓨터가 이해할 수 있는 데이터 표현이 된다.

본 연구에서는 여러 곳에 분산되어 각기 다르게 정의해 놓은 프로세스들을 BOS 라는 운영체제 위에서 OWL 과 같은 온톨로지 언어를 사용함으로써 마치 하나의 프로세스를 다루듯 통합시키고 있다.



[그림 3] 온톨로지 언어를 이용한 의미의 통합

### 3.3. 시스템 구성요소 및 아키텍처

BOS 의 기능적 요구사항을 분류한 것을 바탕으로 각각의 요구사항 범주에 적합한 시스템의 기능적 구성요소를 [표 1]에 제시하였다.

구성요소들을 토대로 BOS 전반의 아키텍처를 제시하면 [그림 4]과 같다. 본 아키텍처에서는 UDDI 와 같은 레지스트리 부분은 배제하고, 비즈니스 프로세스가 정의되어 있는 조직의 시스템(이하 조직시스템)과 조직시스템으로부터 업무를 받아서 수행할 수 있도록 해주는 BOS 의 관계만을 나타내었다. 각 구성요소는

Functional requirement	Functional component
Work and worklist management	Worklist handler
Supporting execution of an activity	Activity executer
Work item negotiation to select a proper work for an actor	Activity negotiator
Activity related knowledge management	Knowledge manager Communication tool Knowledge navigator
Knowledge management for future work and improving one's capability	E-learning tool Knowledge navigator
User profile	User profile manager

management and advertisement	
Schedule management and arrangement	Schedule manager

[표 1] 기능적 요구사항에 따른 시스템의 구성요소 및 표현을 위한 요구사항

작은 직사각형으로, 데이터베이스는 원주로, 각 구성요소들이 주고 받는 정보는 화살표로 표현하였다. 그리고 이러한 요소들을 둘러 싸고 있는 큰 직사각형은 하나의 시스템을 나타낸다.

그림의 아랫부분에 있는 큰 직사각형이 BOS 이고, 윗부분에 작은 두 개의 직사각형은 조직의 시스템이다.

#### 3.3.1. 업무협상도구 (Activity Negotiator)

BOS 의 업무협상도구(Activity negotiator)와 조직시스템의 업무협상도구는 웹 서비스로 구성된 에이전트로서, 네트워크 상의 메시지 교환을 통해 업무협상을 이룬다. 협상이 이루어지기 전에 웹을 검색하여 업무수행자의 연구분야와 역할에 맞는 업무들의 정보(Work information)를 받아온다. 받아온 업무들 중에 업무수행자에게 최적의 조건을 갖춘 업무를 선정하여 준다. 업무, 연구분야 및 역할에 대한 선호, 스케줄, 업무부하 등을 고려한 최적화 알고리즘이 이 도구에 내장되어 있다.

#### 3.3.2. 업무관리도구 (Worklist handler)

업무수행자가 임시 업무목록에 공지되어 있던 업무를 거절할 경우, 업무정보를 삭제한다. 수락할 경우에는 임시 목록에 있던 업무가 현재 수행중인 업무목록으로 이동하고 업무정보(Activity)가 개인저장소에 저장된다. 업무수행자가 자체적으로 업무를 추가, 수정, 삭제할 수 있는 기능도 제공한다. 업무관리도구에서는 업무목록뿐만 아니라 개인저장소에서 부터 받은 업무정보를 업무수행자에게 보여준다. 또한, 업무의 기한이나 업무의 진행상황, 업무의 중요도 등을 바탕으로 업무의 우선순위를 결정해줌으로써 업무수행자가 주어진 업무를 간과하지 않도록 해준다.

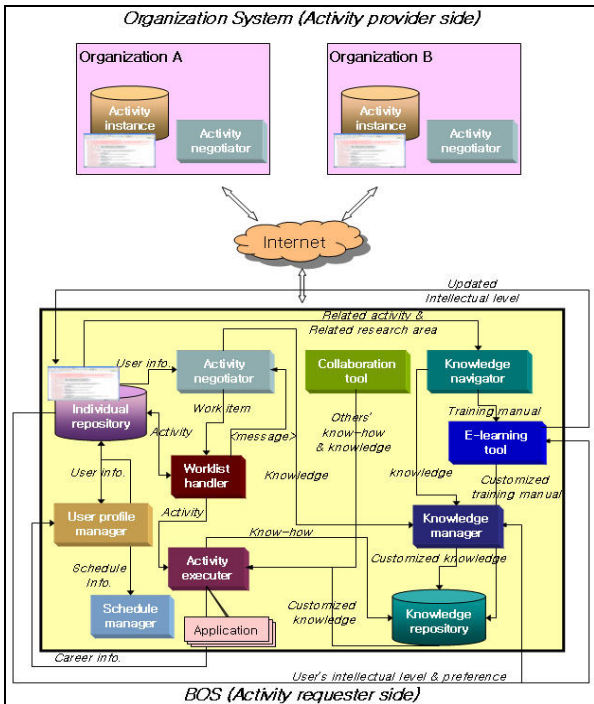
#### 3.3.3. 업무수행도구 (Activity performing tool)

업무관리도구로부터 수행해야 할 업무를 받으면, 업무수행자가 수월하게 업무를 수행할 수 있도록 어플리케이션(Application), 업무 로직, 업무와 관련된 규칙, 업무와 관련된 데이터, 맞춤형된 업무관련지식(Customized knowledge) 등을 업무와 함께 제공해준다. 업무수행 중에는 의사소통도구(Collaboration tool)와 연계해서 다른 사람의 노하우나 지식(Others' know-how & knowledge)을 받을 수 있다. 업무수행 중에 업무수행자는 자신의 노하우(Know-how)를 지식저장소(Knowledge repository)

에 저장할 수 있다.

### 3.3.4. 지식관리도구 (Knowledge manager)

지식관리도구에서는 업무협상도구, 지식검색도구 (Knowledge navigator)로부터 받은 업무관련 지식들



「그림 41 BOS의 전체 아키텍처

을 맞춤화(Customized knowledge)하여 지식저장소에 저장하는 역할을 수행한다. 이때 지식을 맞춤화하기 위하여 개인저장소로부터 업무수행자의 지적 모델과 지식에 대한 선호도(User's intellectual level & preference)를 받는다. 업무수행자의 지적 모델과 지식에 대한 선호도 정보는 업무수행 경험이나 지식 습득 정도에 따라 지속적인 업데이트가 됨으로써 지식의 개인 맞춤화를 위한 지표로 사용된다.

### 3.3.5. 지식검색도구 (Knowledge navigator)

지식검색도구는 네트워크를 통해 BOS를 사용하는 업무수행자에게 필요한 모든 지식을 검색해온다. 이 도구에서는 개인저장소로부터 업무와 연구분야정보(Related activity & Related research area)를 받은 후, 이에 맞는 업무관련지식을 모두 검색해서 지식관리도구로 보낸다. 또한, E-learning 웹 사이트들을 검색하여 업무수행자의 연구분야 및 관심분야에 맞는 교육 프로그램들을 전부 찾아와서 E-learning 도구로 보낸다.

### 3.3.6. 의사소통도구 (Communication tool)

업무를 수행함에 있어서 자신의 노하우나 지식, 검색된 지식만으로는 부족한 경우가 발생한다. BOS에서는 함께 업무를 수행하거나 비슷한 업무를 수행하는 사람들이 모인 가상작업공간에 대한 의사소통도구를 제공함으로써 다른 업무수행자의 노하우나 지식을 습득할 수 있도록 하였다. 반대로 나의 지식

이나 노하우 역시 의사소통도구를 통하여 다른 사람들에게 전달될 수 있다. 의사소통도구는 전자우편, 전자게시판, 전자결재, 데이터 공유, 전자회의 지원 등의 통신 기능을 갖춘 그룹웨어의 형태로 제공된다.

### 3.3.7. E-Learning 도구 (E-learning tool)

지식 집약적인 업무수행자일수록 능력의 지속적인 개발이 중요하다. BOS의 E-learning 도구에서는 지식검색도구로부터 받은 업무수행자의 관심분야와 연구분야에 맞는 교육프로그램(Training manual)들 중에 사용자의 지적 모델이나 지식에 대한 선호도에 가장 잘 맞는 맞춤형 교육프로그램(Customized training manual)을 지식저장소에 저장함으로써, 수행자가 원할 때 그 교육을 받을 수 있다. 또한, 교육수행 결과(Updated intellectual level)를 사용자의 경력에 반영하여 추후 업무 협상에 대한 지표로 이용할 수 있도록 한다.

### 3.3.8. 업무수행자 정보관리도구 (User profile manager)

이 도구에서는 업무수행자 자신에 대한 정보들을 전반적으로 관리해준다. 업무수행자정보(User information)를 개인저장소로부터 불러와 업무수행자에게 보여주는 기능과 새로 기입되거나 수정된 정보를 다시 개인저장소로 써주는 기능을 갖고 있다. 업무수행도구로부터 받은 경력정보를 개인저장소에 업데이트하는 기능도 갖는다.

### 3.3.9. 스케줄 관리도구(Schedule manager)

기존의 스케줄 관리 소프트웨어는 개인의 일정을 등록하고, 시간이 되면 공지를 해주는 정도의 기능만을 가지고 있어 실제로 사용하는 사람이 많지 않다. 그러나 여기서 제시하고 있는 스케줄 관리 도구는 업무관리, 지식관리, 프로필관리 등과 통합된 스케줄 관리를 제공한다. 업무나 교육 등에 사용되는 시간을 체크하여 업무부하를 계산하는 기능도 가지고 있다. 또한, 시스템 사용자의 현재 업무나 일정에 따라 개인의 스케줄 패턴을 분석하여 하루하루의 생활에 대한 가이드라인을 제시해준다.

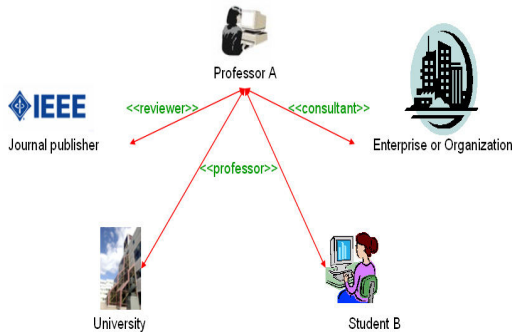
## 4. 프로토타입 개발

이 장에서는 지금까지 설명한 BOS의 아키텍처를 프로토타입으로 구현해 봄으로써 그 유효성을 증명한다. 프로토타입의 시나리오를 설명하고, 실행화면을 통한 구현 결과를 보여준다.

### 4.1. 시나리오

BOS를 사용하는 가상의 상황을 설정해보자. A 교수는 대학에서는 교수로, 기업에서는 컨설턴트로, 출판사에서는 논문심사원(Reviewer)으로서의 역할을 가지고 있다. 물론, 그는 대학이라는 조직에 속해있으나, 기업과 출판사 등에는 소속되지 않는 업무수행자이다. 기존에 그는 기업과 관련된 프로젝트를 직접 찾아서 수행하고, 스케줄을 고려하지 않은 논문심사

를 부탁 받았다. 그의 업무는 대부분 지식 집약형이어서 지식이나 노하우 관리가 중요함에도 불구하고 기존에는 지식을 단지 윈도우 폴더 별로 정리하는 방법밖에 없었다. 따라서, 그는 업무수행을 원할 때 그에게 맞는 업무를 찾아주고, 업무를 효율적으로 수행할 수 있게 업무관련지식관리, 시간관리 및 홍보를 대신해줄 수 있는 시스템으로써 BOS 를 사용하려고 한다. A 교수의 BOS 와 다른 시스템들간의 관계를 도식화하면 [그림 5]과 같다



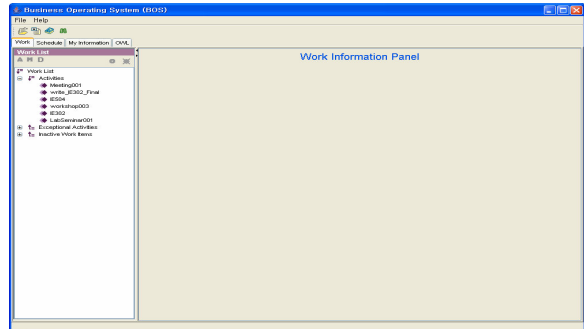
[그림 5] A 교수의 BOS 와 다른 시스템들과의 관계

4.2. 구현결과

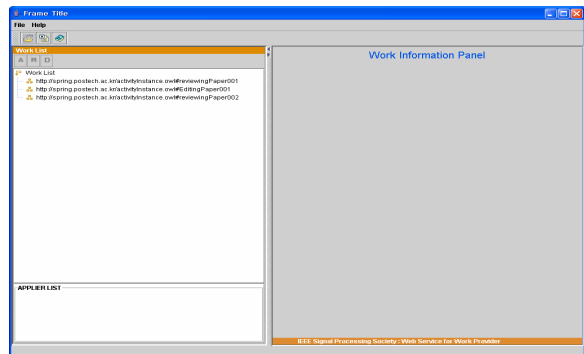
BOS 와 업무를 제공하는 조직의 시스템(이하 출판사시스템) 프로토타입의 개발환경은 다음과 같다. BOS 와 출판사시스템 둘 다 구현하였고, 그 각각의 시스템은 서버와 클라이언트 모듈을 동시에 가지고 있어 상호 교류가 일어날 수 있도록 하였다. 시스템의 개발환경은 다음과 같다.

- 개발언어 : JDK 1.4.2
- 정의언어 : OWL
- 개발환경 : 서버 & 클라이언트 모듈
  - ✓ 서버 : Web service server, Apache Axis, Servlet 2.3, JSP 1.2, SOAP 1.1
  - ✓ 클라이언트 : Application
- 개발도구 : Protégé, Jbuilder X
- OWL Parsing Tool : OWL API, Jena API[11], Pellet API[10]

[그림 6]과 [그림 7]는 BOS 와 논문심사업무를 제공하는 출판사시스템 각각의 메인 페이지를 보여준다.



[그림 6] BOS 의 메인 화면

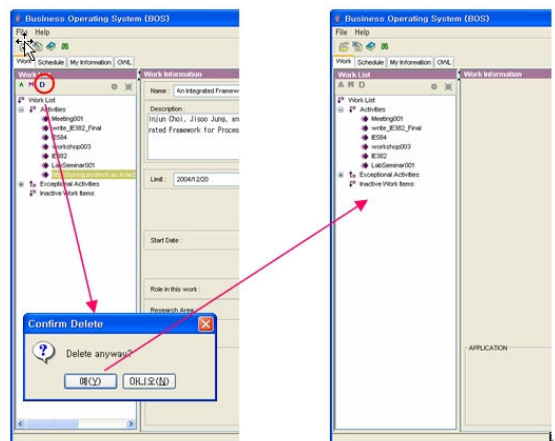


[그림 7] 출판사 시스템의 메인 화면

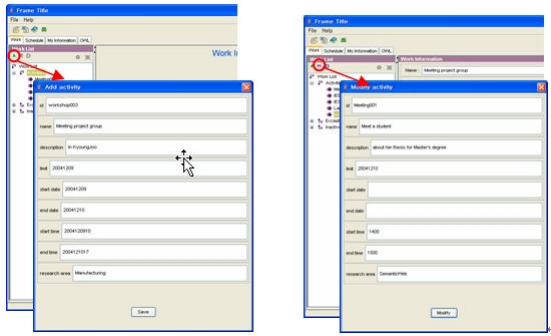
BOS 는 [그림 8], [그림 9]와 같이 업무를 삭제, 추가 수정 할 수 있는 기능을 가지고 있다.

또한 업무협상기능을 가지고 있어서 [그림 10]와 같이 업무요청버튼을 클릭하면, 외부의 시스템에서 시맨틱 웹으로 공개하고 있는 업무들을 BOS 가 모두 검색한다.

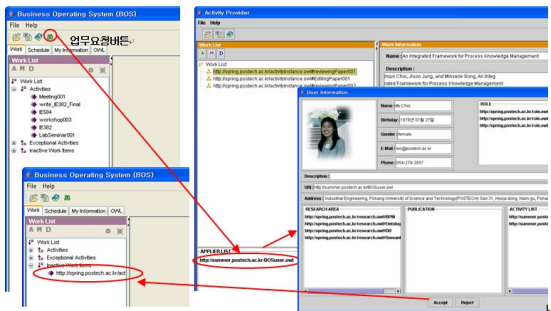
[그림 11]에서처럼 공지된 업무를 업무수행자가 수락하면 해당업무가 Activities 하위로 이동한다. 또한 업무목록의 각각의 업무에 대한 상세 정보도 보여준다.



[그림 8] 업무 삭제

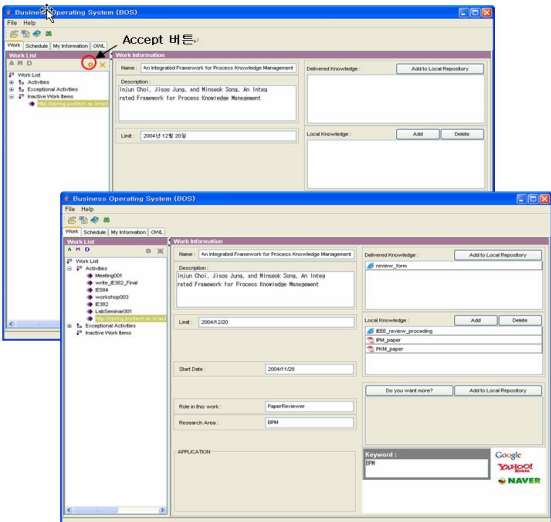


[그림 9] 업무 추가 및 수정



[그림 10] 업무협상화면(왼쪽: BOS, 오른쪽: 출판사 시스템)

그 밖에도 업무와 관련된 지식을 분류해서 제공하며, 지식관리기능으로 인해 제공된 지식을 클릭하면 지식의 정보화면이 나타나 세부 내용도 볼 수 있다. 검색엔진을 통한 지식검색이 가능하며 업무수행자의 정보관리기능을 구현하여 개인정보를 등록, 수정할 수 있다.



[그림 11] 협상이 끝나고 공지된 업무에 대한 수락 화면과 업무정보화면

### 5. 결론 및 추후 연구 과제

본 연구에서는 조직 내외의 여러 시스템에 분산되어 있는 업무 및 지식을 통합하고 사용자의 모든 행정

업무처리를 대신하거나 도와줌으로써, 맞춤화되고 포괄적인 업무수행환경을 단일 인터페이스로 제공해주는 업무수행지원시스템인 비즈니스운영체계(BOS)의 아키텍처를 제시하였다. 특히 Koulopoulos 가 제시한 비즈니스운영체계가 지식공유에만 중점을 두고 있는 반면에, 본 연구에서 제시한 비즈니스운영체계는 지식공유뿐만 아니라 업무수행지원을 위한 포괄적인 기능을 제안하고 있다. 본 연구가 지식집약적인 업무수행자의 효율적인 업무수행에 초점을 맞춘 BOS 이기 때문에, 추후연구에서는 본 연구의 BOS 기능을 일반적인 사용자의 행정업무를 지원하도록 확장할 수 있을 것이다. 또한, 인텔리전트한 BOS를 위한 각종 AI 기법들의 적용 등을 생각해 볼 수 있다.

### 참고문헌

[1] Choi, I., Song, M., Park, C., and Park, N., "An XML-based Process Definition Language for Integrated Process Management" in Computers in Industry, Vol. 50, Issue 1, pp.85-102, Jan. 2003.  
 [2] Injun Choi, Jisoo Jung, and Minseok Song "An Integrated Framework for Process Knowledge Management " in International Journal of Innovation and Learning, Vol. 1 No. 4, pp.399-408, 2004.  
 [3] 윤호성, “ 웹서비스 기반의 효율적인 프로세스 실행을 위한 프로세스 정의 언어 및 실행 아키텍처”, 석사논문, 포항공대 산업공학과, 2004  
 [4] 이영권, “ 웹서비스 기반의 유연한 비즈니스 프로세스 수행 아키텍처”, 석사논문, 포항공대 산업공학과, 2004  
 [5] Berners-Lee, T., “ The Semantic Web ”, Scientific America, vol. 501, 2001  
 [6] W3C, “ OWL Test Results (Semi-Official Semi-Static View) ”, http://www.w3.org/2003/08/owl-systems/test-results-out, 2004  
 [7] Koulopoulos, T.M., “ Smart Companies, Smart Tools”, John Wiley & sons, 1997  
 [8] Shilakes, C. C. and Tylaman, J. “ Enterprise information portals”, Merrill Lynch, Inc, New York, November 16, 1998  
 [9] BPMI Homepage, http://www.bpmi.org, 2004  
 [10] Maryland Information and Network Dynamics Lab, “ Pellet OWL Reasoner”, http://www.mindswap.org/2003/pellet/, 2004  
 [11] Jena Framework Homepage, http://jena.sourceforge.net/javadoc/index.html, 2004