

웹 서비스를 이용한 워크플로우 레지스트리 구현

김경수* · 송영기** · 신현철*

요 약

워크플로우 시스템의 필요성이 증가함에 따라 기존에 주된 응용분야인 정부나 교육기관과 같은 공공부문의 일반 관리들뿐만 아니라 많은 분야의 기관들이 워크플로우 시스템을 사용할 것이다. 그러나 기존의 워크플로우 시스템들은 오직 시스템 내부에서만 프로세스 정의의 공유가 이루어지고 있기 때문에 서로 다른 플랫폼을 사용하는 협력업체와의 프로세스 정의 공유가 어렵다는 단점이 있다.

본 논문에서는 이러한 단점을 해결하기 위해 웹 서비스를 이용하였다. 웹 서비스는 네트워크가 구축되어 있으면 어느 곳에서든 이용할 수 있으며 서로 다른 플랫폼간의 연동이 가능하다는 장점이 있으므로 이를 이용하여 워크플로우 레지스트리를 개발하였다. 이로 인해 지리적으로 분산되어 있는 기업 내의 여러 부서나 서로 다른 플랫폼을 사용하는 협력 기업들 간에도 워크플로우 프로세스 정의를 공유할 수 있게 될 것이다.

Implementation the Workflow Process Definition Registry Using Web Service

Kyung-Soo Kim* · Young-Ki Song** · Hyun-Cheul Shin*

ABSTRACT

According to increasing necessities of workflow systems, it will be useful to most Enterprises. However existing workflow systems are shared only inside of systems that use a same process definition. It is difficult to share a process definition with another enterprise using different platforms.

In this paper, we used Web Service for sharing process definition. Web Service have advantage that can linked between different platforms. Therefore we developed workflow process definition by using Web Service. Using Web Service, we can be share workflow process definition between enterprises using different platform.

Key words : Web Service, Workflow Process Definition, Registry

* 백석문화대학 컴퓨터정보학부

** 백석문화대학 보건재활학부

1. 서 론

분산 환경에서 동작하는 워크플로우 시스템은 다양한 플랫폼에서 참여하는 사용자를 모두 지원해야 하며, 다른 종류의 워크플로우 시스템과의 상호 운용성(Interoperating)이 있어야 한다. 이를 위해서는 단위 업무들 간의 관계와 프로세스 생성 및 종료에 대한 기준, 그리고 단위 업무에 대한 정보로 구성된 워크플로우 프로세스 정의에 대한 공유가 필요하다.

그러나 기존의 워크플로우 시스템들은 시스템 내부에서만 프로세스 정의의 공유가 이루어지고 있으며 서로 다른 플랫폼을 사용하는 협력업체와는 프로세스 정의에 대한 공유가 어렵다는 단점이 있다. 이러한 단점을 해결하기 위한 방법 중 하나가 웹 서비스를 이용하는 것이다.

웹 서비스는 최근 기업 내 애플리케이션 통합에 대한 요구가 커지고 기업간 협업의 필요성이 증대됨에 따라 주목받고 있는 해결책 중 하나이다. 기업의 IT 투자가 지속적으로 이루어지면서 각 업무 솔루션 위주로 개별적으로 구축된 기업 내 시스템들 사이의 통합 이슈가 부각하였다. 그러나 기존의 애플리케이션 모델로는 이기종 플랫폼 간의 통합이 매우 어려웠으며, 이러한 문제를 해결하기 위해 웹 서비스가 등장한 것이다[1].

본 논문에서는 기업 내의 시스템뿐만 아니라 사용하는 플랫폼이 서로 다른 협력 기업 간에도 워크플로우 프로세스 정의를 공유하기 위해 웹 서비스를 이용하여 워크플로우 레지스트리를 구현하였다. 제 2장에서는 관련연구로서 워크플로우와 웹 서비스에 관한 개념과 웹 서비스 애플리케이션 개발에 필요한 Axis에 대해 설명하고, 제 3장은 시스템 아키텍처 및 구현된 시스템에 대해 다루고, 마지막으로 제 4장에서는 본 논문의 결론에 대하여 기술한다.

2. 관련 연구

2.1 워크플로우

워크플로우는 업무처리 절차를 수행하기 위해서 일어나는 일련의 업무들의 흐름을 뜻한다. 따라서 워크플로우 기술이란 어느 기업 또는 조직체 내부나 조직체 간에 발생하는 일련의 업무의 흐름 즉 프로세스를 정의하고, 정해진 시간 안에 업무가 관련 정보와 함께 자동으로 수행되도록 제어하는 분산 컴퓨터 기반의 시스템을 일컫는 말이다. 워크플로우 기술 및 시스템을 통해 복잡하고 다양한 업무의 흐름을 효과적으로 제어함으로써 해당 기업은 비용의 절감, 생산성의 향상, 빠르고 정확한 서비스 등을 제공할 수 있기 때문에 결과적으로 워크플로우 기술의 도입 및 적용은 기업의 경쟁력 자체를 증가시키는 중요한 요소가 된다[2]. 워크플로우가 지금까지 적용된 분야로는 정부나 교육기관과 같은 공공부문의 일반 관리와 서비스 업무, 금융·보험업과 서비스업의 고객업무, 제조업의 일반 관리 업무들이 있다[3].

2.2 워크플로우 레지스트리

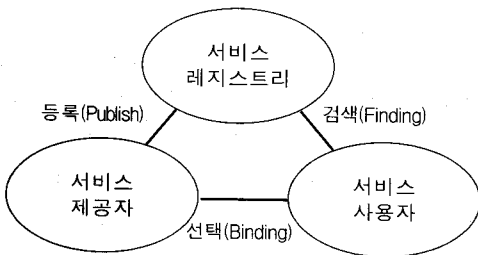
현재 워크플로우 시스템이 가지고 있는 단점 중 하나는 시스템의 이질성 및 상호운용성을 지원하는 측면에서의 제한과 급격히 증가하는 프로세스에 따른 시스템 오버헤드, 장애에 대한 대응이 부족하여 정확성과 신뢰성이 떨어진다는 점이다[4]. 또한, 기존의 워크플로우 시스템을 사용하고 있는 기관 중 새로운 프로세스를 정의할 필요가 있는 곳이나 워크플로우 시스템을 처음 도입하는 기업이나 병원, 등의 개인 업체나, 공공기관들은 새로운 워크플로우 프로세스에 대한 정의를 위해 시간과 비용을 낭비하고 있다.

이런 문제점을 해결하기 위해 개발된 워크플로우 등록소는 워크플로우 시스템을 사용하는 기업이나

단체들 간에 워크플로우 프로세스 정의를 웹상에서 공유함으로써 사용자들은 각자 필요로 하는 프로세스 정의를 검색하여 그대로 사용하거나, 약간의 수정작업을 거쳐 사용할 수 있도록 하였다. 따라서 새로운 워크플로우 프로세스 정의를 생성하기 위한 시간 및 비용의 손실을 막을 수 있게 된다. 또한, 다른 워크플로우 시스템을 사용하는 기관 간에도 워크플로우 프로세스 정의의 공유가 이루어질 수 있으므로 시스템의 이질성 및 상호 운용성 등의 문제를 해결할 수 있다. 또한, 역할 기반의 접근제어를 이용하여 사용자들에게 각각의 직업 및 직책에 따라 적절한 역할과 그에 따른 권한을 부여하였기 때문에 사용자들은 자신의 역할에 부여된 권한의 범위를 벗어난 기능은 수행할 수 없게 될 것이다. 따라서 관리자 이외에는 워크플로우 레지스트리 내의 데이터 변경 및 삭제는 없을 것이다.

2.3 웹 서비스

웹 서비스는 XML 메시징을 통하여 인터넷을 통하여 접근 가능한 네트워크 명령들의 집합을 기술한 인터페이스로서, 단일 또는 서로 다른 플랫폼과의 상관 관계없이 상호운영이 가능하도록 해주는 표준화된 기술이다. 웹 서비스는 (그림 1)에서 나타난 것과 같이 크게 세 가지로 구성된다.



(그림 1) 웹 서비스 구조

- (1) 서비스 제공자(Service Provider) : 웹 서비스를 개발하는 역할을 함. 공개적으로 접근할 수 있는 중앙 레지스트리에 웹 서비스의 세부사항

을 공개(Publish)한다.

- (2) 서비스 사용자(Service Consumer : Client) : 웹 서비스를 사용하는 애플리케이션을 의미함. 레지스트리를 검색(Find)하여 서비스의 세부사항을 알아내고, 그 후 원하는 서비스에 binding하여 실제로 해당서비스의 기능을 이용한다.
- (3) 서비스 레지스트리(Service Registry) : 서비스 제공자가 자사의 웹 서비스 상세내역을 올려두고 Client가 발견할 수 있도록 하는 중앙 저장소를 의미한다.

(그림 1)에서 보이는 세 가지 오퍼레이션[검색, 선택, 등록(공개)]에 관한 표준안은 WSDL(Web Service Description Language), SOAP(Simple Object Access Protocol), UDDI(Universal Description, Discovery, and Integration)로 구성된다.

웹 서비스는 XML을 근거로 표현하고 그 위에 웹 서비스를 기술하는 WSDL, 웹 서비스를 나열하고 찾기 위한 UDDI, 그리고 웹 서비스를 호출하고 메시지를 교환하기 위한 SOAP 등을 사용하고 있다[5].

SOAP은 소프트웨어 서비스들 간 네트워크 통신을 위한 표준으로 분산 환경에서 정보를 교환하기 위해 제안된 XML 기반의 메시징 프로토콜이다. 또한, SOAP은 XML 문서를 SMTP, HTTP, FTP 등 다양한 표준 인터넷 기술로 전송하는 표준 포장 구조(standard packaging structure)를 제공한다. XML을 사용하기 때문에 애플리케이션이 동작하는 운영체제, 언어 그리고 객체 모델에 관계없이 애플리케이션 사이의 통신을 할 수 있게 된다.

WSDL은 표준화된 방식으로 웹 서비스의 인터페이스를 기술하는 XML 기술이다. WSDL은 웹 서비스가 호출된 입출력 변수, 함수구조, 호출의 성질, 서비스 프로토콜 바인딩을 위부에 표시하는 방식을 표준화한다. 또한, 서로 다른 클라이언트가 웹 서비스를 통해 상호작용 하는 방식을 파악하게 해준다. 또한, WSDL 문서에는 웹 서비스 구현의 세부사항 정보도 포함되어 있다. 따라서 소비자는 WSDL 문서를 통해 웹 서비스의 여러 오퍼레이션과 URL 정보를 얻을 수 있다.

UDDI는 웹 서비스를 온라인 디렉토리에 등록, 공개하기 위해 개발된 규약이다. UDDI는 웹 서비스의 전 세계 레지스트리를 제공하며 비즈니스, 웹 서비스, 규격 메타데이터, 웹 서비스 접근 위치를 나타내는 구조를 제공한다. 웹 서비스에서는 UDDI 명세를 통해 웹 서비스 정의의 공개(publish)와 발견(discovery)에 대한 사항을 정의하고 있다.

HTTP나 SMTP같은 표준 기반 인터넷 프로토콜을 통해 접근할 수 있다는 점을 고려하면 J2EE, CORBA, CGI 스크립을 웹 서비스 기술로 분류할 수도 있지만 그렇지 않다. 이러한 기술들과 웹 서비스라고 하는 새로운 기술들과의 가장 큰 차이는 표준화에 있다. SOAP, UDDI, WSDL등의 새로운 기술들은 표준화된 XML을 기반으로 하여 전 세계적으로 주요 기술 회사의 지원을 받는다.

2.3.1 IBM Web Services Toolkit

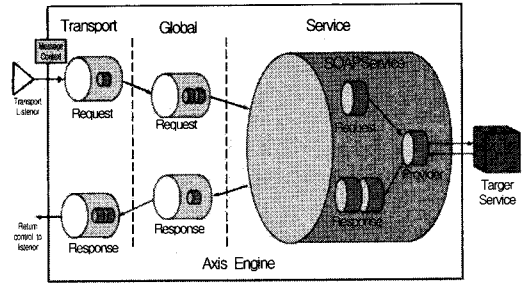
IBM은 AlphaWorks 프로젝트로 WSTK(Web Service Toolkit) 개발을 시작하여 현재 버전 3.2.1까지 발표한 상태이다. WSTK V3.2.1에서는 웹 서비스 관련 최신 기술들을 지원하는 모듈을 제공한다.

2.3.2 SUN Web Services Developer's Pack

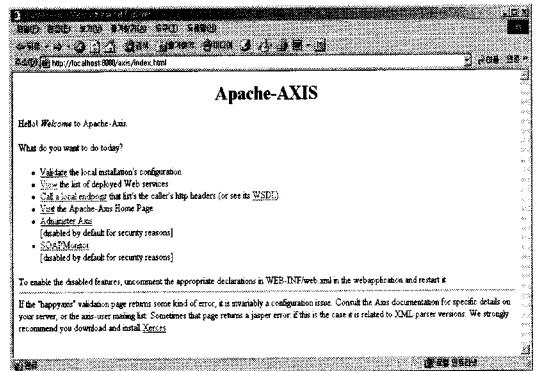
SUN에서는 JWSDP(Java Web Services Developer Pack)을 발표하였다. JWSDP에는 JAXP(Java API for XML Processing), JAXR(XML Registries), JAXM(XML Messaging), JAX-RPC(XML-based RPC), SAAJ(SOAP with Attachments API for Java)와 같은 모듈을 제공하고 있다.

2.3.3 Apache AXIS

웹 서비스 애플리케이션에 필요한 API를 서버에 설치하기 위해 사용되는 AXIS는 아파치의 전 프로젝트인 아파치 SOAP 2.2를 모태로 하여, 보다 나은 유연성 제공을 목표로 개발되었으며(그림 2)는 그 구조를 나타내고 있다[6].



(그림 2) AXIS 구조



(그림 3) AXIS 실행화면

AXIS는 이벤트 방식의 SAX XML 파서를 이용하며 DOM 방식의 아파치 SOAP보다 훨씬 빠르다. AXIS의 시스템 구조는 개발자들에게 사용자 정의 헤더를 처리하기 위해 엔진을 확장하거나 시스템 관리 기능을 추가하는 등 어떠한 형태의 확장도 가능하도록 해준다. AXIS는 지속적으로 제공되는 인터페이스를 정의함으로써 개발자들이 AXIS의 업그레이드에 영향을 받지 않도록 한다. AXIS는 개발자들이 개발한 응용 프로그램을 처리하기 위해 일반적으로 사용될 패턴을 구현하는 재사용 가능한 핸들러를 정의할 수 있고 이들을 분산시킬 수 있다. (그림 3)은 AXIS의 초기 실행 화면이다.

2.4 워크플로우 시스템의 한계

현재 다양한 회사에서 만들어낸 워크플로우 상

용시스템이 많이 존재하고 있으며, 이 중 사용자들이 주로 사용하고 있는 워크플로우 시스템은 20여 개 정도가 있다. 그러나 이러한 시스템들은 다음과 같은 한계성을 지니고 있다[7].

첫째, 시스템의 이질성 및 상호 운용성 지원 측면에서 많은 제한이 된다는 점이다. 분산 환경에서 동작하는 워크플로우 시스템은 다양한 플랫폼에서 참여하는 사용자를 모두 지원해야 하며, 다른 정보 시스템과의 통합을 지원해야 하고, 또한 다른 종류의 워크플로우 시스템과의 상호 운용성이 있어야 한다. 둘째, 시스템의 정적 특성 때문에 동적으로 변화하는 비즈니스 환경에 적용이 어렵다는 점이다. 대부분의 시스템이 한 번 정의된 프로세스를 따라서 업무를 완료하도록 하고 있는데, 실제 비즈니스 환경은 동적 변화가 자주 발생하므로 대부분의 시스템들이 이에 대한 대처가 부족하다. 셋째, 워크플로우 분석이나 관리 감독 기능이 부족하다. 정의된 프로세스에 대한 분석 기능이 부족하고, 프로세스 진행 모니터링 기능이 미비하다. 넷째, 오류(Failure)에 대한 대응이 부족하여 정확성과 신뢰성이 떨어진다는 점이다. 오류가 발생한 경우에 대한 대응은 필수적인 부분임에도 불구하고 대부분의 시스템들이 이 기능에 대한 대비가 부족하다[8].

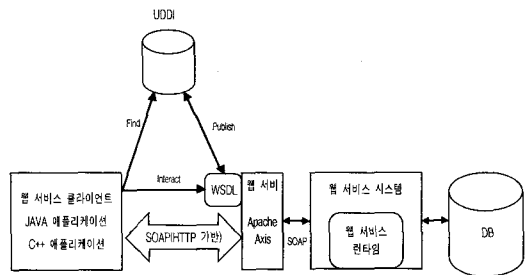
본 논문에서는 시스템의 이질성 및 상호 운용성에 대한 문제를 해결하기 위해 웹 서비스를 이용하여 기업 내의 시스템뿐만 아니라 협력 기업 간에도 워크플로우 프로세스 정의를 공유할 수 있는 워크플로우 레지스트리를 개발하였다.

3. 웹 서비스를 이용한 워크플로우 레지스트리 구현

3.1 워크플로우 레지스트리 아키텍처

(그림 4)는 본 논문에서 개발한 워크플로우 레지스트리의 아키텍처이다. 웹 서비스 시스템은 J2EE

에서 말하는 컨테이너 개념과 유사한 것으로 웹 서비스를 실행하기 위한 런타임 환경을 제공하는 것으로서 SOAP 요청을 받아들이고 이 요청을 처리할 수 있는 적절한 자바 컴포넌트로 보내 그에 따른 기능을 처리한다. 웹 서버는 SOAP 클라이언트의 요청을 서비스 제공자에게 전달하는 중요한 게이트웨이이며 WSDL 문서가 위치하는 곳이다. 웹 서비스 클라이언트는 웹 서비스의 소비자로서 플랫폼 독립적이기 때문에 어떤 언어로 작성된 클라이언트도 웹 서비스를 호출할 수 있다. 또한, UDDI에서 원하는 서비스의 정보를 참조하는데, 보통 UDDI에서 제시된 URL을 통해 WSDL 문서를 받아오며 이는 해당 웹 서비스의 웹 서버를 거쳐게 된다.

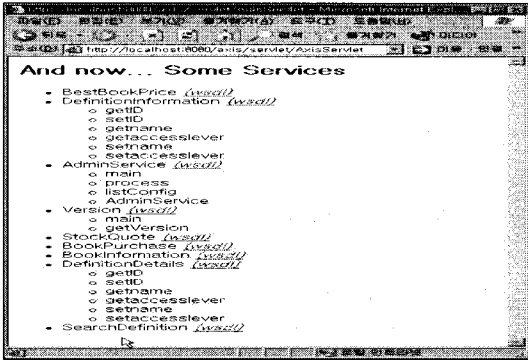


(그림 4) 워크플로우 레지스트리 아키텍처

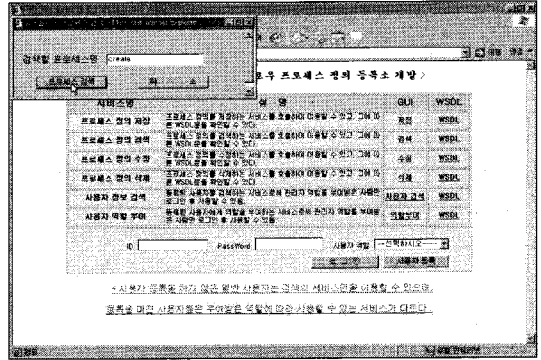
3.2 워크플로우 레지스트리 구현

본 논문에서 구현한 시스템은 DBMS로 오라클 8i를 사용하였고, 웹 서버로 Tomcat 4.1.3을 사용하였으며 웹 서비스 애플리케이션 개발을 위해 Axis v1.1을 사용하였다. AXIS를 이용하여 웹 서비스를 배치하는 방법은 JAVA 파일을 JWS 파일로 변경하여 서블릿 컨테이너에 설치하는 경우와 WSDD 파일을 이용하는 경우 두 가지가 있는데, 본 논문에서는 WSDD 파일을 이용하는 방법을 사용하여 서비스를 배치하였다.

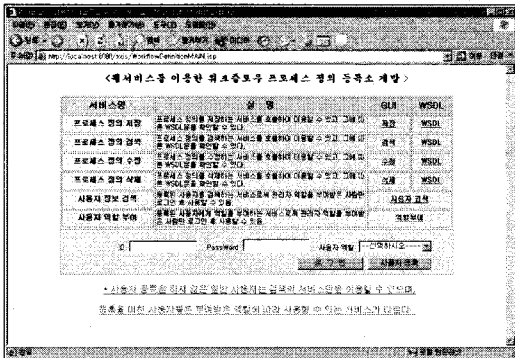
(그림 5)는 본 논문에서 개발한 웹 서비스가 AXIS 서버에 정상적으로 등록되었다는 것을 보여주고 있다.



(그림 5) 웹 서비스의 배치



(그림 7) 검색 서비스 호출

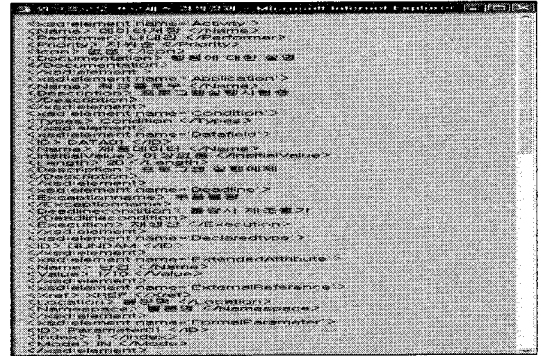


(그림 6) 워크플로우 정의 레지스트리 메인화면

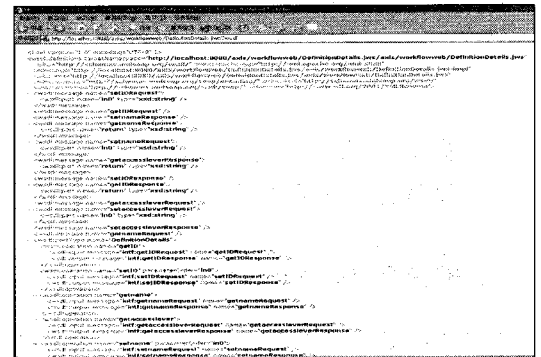
(그림 6)은 본 시스템의 메인화면으로서 개발된 서비스들의 이름과 그 서비스가 수행하는 역할에 대한 설명, GUI, 그리고 해당 서비스에 대한 WSDL을 검색할 수 있도록 하였다. 프로세스 정의를 저장하거나 검색하는 서비스는 누구라도 이용할 수 있지만, 수정과 삭제의 서비스는 프로세스 정의를 저장한 본인만이 수행할 수 있도록 하기 위해 사용자에게 역할을 부여하고 자신의 역할의 범위 내에서 다른 서비스를 이용할 수 있도록 하였다. 등록된 사용자에 대한 정보를 검색하거나 사용자들에게 역할을 부여하는 기능은 관리자의 역할을 부여받은 사람만이 이용할 수 있다.

(그림 7)은 이러한 시스템의 기능 중 검색 서비스를 이용하기 위해 GUI를 호출한 것이다.

(그림 8)은 검색에 대한 결과가 XML형식으로 출력되는 것을 보여주는 것이며, (그림 9)는 이러한 검색 서비스에 대한 WSDL문을 보여주고 있다.



(그림 8) 검색결과



(그림 9) 검색에 관한 WSDL

4. 결 론

워크플로우 시스템의 필요성이 증가함에 따라 기존에 주된 응용분야인 정부나 교육기관과 같은 공공부문의 일반 관리와 서비스 업무, 금융·보험업과 같은 정보통신 서비스업의 고객업무나 제조업의 일반 관리 업무들뿐만 아니라 많은 분야의 기관들이 워크플로우 시스템을 사용할 것이다. 그러나 현재 여러 기관들이 사용하고 있는 워크플로우 시스템은 몇 가지 한계를 나타내고 있는데, 그 중 하나가 시스템 내부에서만 프로세스 정의의 공유가 가능하며 서로 다른 플랫폼을 사용하는 협력업체와는 프로세스 정의의 공유가 어렵다는 것이다.

본 논문에서는 서로 다른 플랫폼 간의 연동이 가능해지고, 네트워크가 구축되어 있는 곳이면 어느 곳에서나 사용자가 원하는 서비스를 이용할 수 있다는 웹 서비스의 장점을 이용하여 워크플로우 레지스트리를 개발하였다. 따라서 서로 다른 플랫폼을 사용하고 있는 협력업체 간에도 워크플로우 프로세스 정의에 관한 정보를 효율적으로 공유할 수 있을 것이다.

각각의 회사들은 자신이 가지고 있는 워크플로우 프로세스 정의에 대한 정보를 WSDL 형식으로 UDDI에 등록시킨다. 그리고 이를 필요로 하는 회사들은 다른 회사에서 프로세스 정의를 등록할 때 기술한 상세 내역을 통해 프로세스 정의에 대한 내용을 검색한 후, 자신들이 원하는 프로세스 정의에 대한 정보를 획득할 수 있다. 이로 인하여 다른 정보 시스템과의 통합 및 서로 다른 워크플로우 시스템과의 상호 운용성이 향상될 것이다.

참 고 문 헌

- [1] 김동수, 유천수, “웹 서비스 이용 현황 조사 및 도입 활성화 방안”, 한국전자거래학회, 제8권, 제2호, pp. 1-22, 2003.
- [2] 김광훈, “워크플로우 기술(표준기술 동향)”, TTA 저널, 2003.

- [3] 서창교, 김정삼, 이형석, “B2B 워크플로우의 메시징 시스템 설계”, 한국경영정보학회, 제11권, 제1호, pp. 117-137, 2001.
- [4] 오동근, 김광훈, “EJB 기반의 워크플로우 정의 데이터베이스 에이전트 설계 및 구현”, 한국인터넷정보학회, 제2권, 제5호, pp. 41-47, 2001.
- [5] 문창모, “웹 서비스 적용 방안”, 한국정보처리학회, 제9권, 제4호, pp. 53-57, 2002.
- [6] 이수정, “웹 서비스를 이용한 바이오 서열 정보 데이터베이스 및 통합 검색 시스템 개발”, 이화여대 석사학위논문, 2003.
- [7] 장재준, 이도현, “워크플로우 정의를 위한 사용자 인터페이스 설계”, 한국정보과학회, 1999.
- [8] 김동수, 배준수, 서영호, 허원창, 김영호, 강석호, “효율적 워크플로우 관리를 위한 시스템 특성 및 설계”, 한국경영과학회, 춘계공동학술대회 논문집, pp. 1-7, 1998.



김 경 수

2001년 순천향대학교 전산학과
(공학박사)

2005년~2006년 Virginia

Common-wealth

University(객원교수)

1998년~현재 백석문화대학 컴퓨터정보학부 조교수



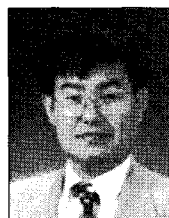
송 영 기

1977년 서울대학교 계산통계학과
(이학사)

1981년 한국과학기술원 전산학과
(공학석사)

2004년 대전대학교 컴퓨터공학과
(공학박사)

1999년~현재 백석문화대학 보건재활학부 부교수



신 현 철

2002년 원광대학교 컴퓨터공학과
(공학박사)

2004년 (주)아이비루션 자문위원

1994년~현재 백석문화대학

컴퓨터정보학부 부교수