

# 중형업체를 위한 웹기반의 프로세스 관리 시스템 제안

이만호\*(연세대 대학원 기계공학과), 이수홍\*\*(연세대 기계공학과)

## A Development of the Web-based Process Management system for a Medium-sized Company

M. H. Lee(Mech. Eng. Dept., Yonsei Univ.), S. H. Lee(Mech. Eng. Dept., Yonsei Univ.)

### ABSTRACT

In this study, we investigate several problems of previous workflow system, and suggest an enhanced process management system that involves a substantial portion of workflow system. To overcome the differences in systems, we selected a web as platform to develop a process management system, and we investigate merit and capability of web-based process management system. We improved WFMS's ability to define various process through many behavior rules and multiple layered process structure. We suggest X-WPM system for the process management system with XML technology.

**Key Words** : ASP, XML, Web-based(웹기반), Workflow(워크플로우), Process(프로세스), X-WPM system

### 1. 서론

오늘날 비즈니스 환경은 전역적인 경쟁으로 인해 비즈니스 수행 비용의 감축, 새로운 서비스와 제품의 빠른 개발을 수행하지 않으면 안 된다. 이러한 환경 하에서 기업은 그들이 비즈니스를 수행하는 방식을 새로이 고려하고 최적화 하려는 노력을 기울이게 되었다. 기업들은 정보시스템을 변화시키고 비즈니스의 프로세스들을 개선시켜 나감으로써 이러한 노력들을 실현해가고 있다. 흔히 BPR이라고 일컬어지는 이러한 기업들의 노력에 대해 워크플로우 관리 시스템이 훌륭한 솔루션으로 주목받고 있으며 이는 워크플로우 관리시스템이 정보의 단순한 공유를 넘어서 프로세스의 진행을 따라가면서 정보의 흐름과 관련된 부서 및 관련자들 간의 업무 연계를 돕고, 정보의 유연한 전달을 위한 기능들을 제공하기 때문이다.

흔히 프로세스는 물질 프로세스(material process), 정보 프로세스(information process), 비즈니스 프로세스(business process)로 구분이 되며 워크플로우는 이해, 평가, 재설계라는 관점에서 비즈니스 프로세스의

태스크를 개념적 수준에서 기술하기도 하지만 정보 시스템의 기능과 사람의 기술적 요구사항을 만족하는 수준에서 정보 프로세스를 제어해야 한다.

### 1.1 관련 기술 동향

CORBA(Common Object Request Broker Architecture)는 가장 먼저 탄생한 분산 객체의 표준으로 기업에서 널리 쓰이고 있다. 최근에는 BOCA(Business Object Component Architecture)라는 CORBA기반의 비즈니스 객체의 표준도 OMG에서 정의되었다. CORBA를 사용하는 경우에 분산 컴퓨팅을 위한 응용 프로그램 개발이 쉬워질 뿐만 아니라, 하드웨어와 운용체계, 프로그래밍 언어와 무관하게 분산 객체들간에 커뮤니케이션이 이루어진다. CORBA는 어떠한 언어로 구성되어 있건 간에 클라이언트와 서버는 서로 커뮤니케이션을 할 수 있다. CORBA로 접속할 수 있는 연결수단을 제공하는 워크플로우 시스템은 CORBA로 구현된 시스템에 접속해 워크플로우 시스템의 연동범위를 확대할 수 있다.

EJB(Enterprise JavaBean)는 썬(SUN)에서 최근에

발표된 자바기반의 분산 객체의 표준으로 트랜잭션 연계/접속관리 등의 비즈니스 객체로서 갖추어야 할 요소들이 충실히 정의돼 있는 컴포넌트의 표준이다. EJB 커넥터를 제공하는 워크플로우 시스템은 이들 컴포넌트로 워크플로우의 기능을 확장할 수 있다. 최근에는 애플리케이션 서버의 등장과 더불어 EJB의 활용도가 증대되고 있어, 워크플로우 시스템에서의 이의 지원여부는 앞으로 중요한 요소가 될 것으로 기대된다.

SWAP(Simple Workflow Access Protocol)은 여러 업체에서 만든 워크플로우 시스템들이 정보를 주고 받기 위한 프로토콜의 표준이다. SWAP은 HTTP 기반의 프로토콜이며, 그 내용은 XML로 구성된다. HP 등이 중심으로 현재 표준을 제정 중이며, XML 기반의 프로토콜을 정의해 워크플로우 시스템들의 상호 연동을 가능하게 하는 것을 목적으로 하고 있다. 직접적인 API 인터페이스를 기반으로 하는 WfMC나 JFlow의 표준과는 달리, 프로토콜만으로 정보를 주고받기 때문에 상대적으로 업체들이 이를 구현하기 쉬운 장점으로 향후 워크플로 상호연동구조의 핵으로 떠오를 것이 예상된다.

## 1.2 웹기반 워크플로우 시스템 현황

현재까지 소개되고 있는 대다수의 웹기반 워크플로우 시스템들을 살펴보면, 그 성능이나 데이터 처리의 안정성과 수행능력, 다양한 뷰 등에서 기존에 구현되어왔던 클라이언트/서버(C/S) 기반의 워크플로우 시스템에 미치지 못한다고 할 수 있다. 그러나 앞에서 이야기한 바와 같이 웹 기반의 시스템이 가질 수 있는 이점에 주목하고, 많은 개발사들이 개발 계획을 발표하고 있으며, 관련 기술이 개발되고, 표준들이 제정되는 등 향후에는 워크플로우 발전 방향이 웹을 기반으로 전개될 것으로 본다. 따라서 앞으로는 점차적으로 웹기반의 워크플로우 시스템이 클라이언트/서버 기반의 워크플로우 시스템을 대체해 나갈 것으로 전망된다.

웹기반 워크플로우를 보다 세분화하여 구분하면, 다음과 같이 웹기반 워크플로우와 웹가능 워크플로우로 나눌 수 있다. 이와 같은 구분에 의하면 대부분의 웹기반의 워크플로우는 'Web-enabled'에서 점차 'Web-based'의 형태로 변화하고 있다.<sup>[2]</sup>

- 웹가능 워크플로우(Web-enabled Workflow)
  - 웹을 지원하는 워크플로우로 기존의 C/S기반의 워크플로우에서 웹을 지원하도록 게이트웨이와 인터페이스를 지원
    - C/S기능은 존재하며, 부분적으로 웹을 지원
    - 프로세스의 시작/단위 작업의 처리/작업 목록의 일람 등의 워크플로우 기능 중 일부를 웹을 이용

하여 액세스할 수 있도록 지원

- 웹기반 워크플로우(Web-based Workflow)
  - 최근 들어 IT업계 전반에서 핫이슈로 등장한 웹 애플리케이션과 유사한 개념으로 웹기술이 유일한 기반구조가 되는 워크플로우 시스템을 지칭함
    - 웹기반의 프로토콜(HTTP, S-HTTP, SSL)을 기반으로 구현
    - 워크플로우의 모든 기능을 웹상에서 지원 (Work item Handling, Graphical Process Definition, Administration 등)

## 1.3 기존 워크플로우시스템의 문제점

지금까지 워크플로우만을 정의하고 관리하는 시스템, PDM 시스템 등에 모듈로 구성된 시스템 등 많은 제품들이 워크플로우의 기능을 구현하였다. 프로세스를 정의, 관리, 지원하는 도구로서 요구되는 많은 기능들이 개발되었고, 기술적인 진보도 이루어져 왔다. 그렇지만 아직까지도 많은 워크플로우 시스템들에서 부족하다고 지적되는 몇 가지 사항을 짚어보면 다음과 같다.

첫째, 시스템간의 이질성 극복과 상호 운영성 확보 및 상호 연동의 문제이다. 워크플로우 시스템은 분산 환경에서 사용되기 때문에 다양한 플랫폼에서 참여하는 사용자들을 모두 지원할 수 있어야 하고, 기타 클라이언트 애플리케이션이나 워크플로우 내에 Invoke된 애플리케이션과 효율적으로 연계될 수 있어야 한다. 그리고 다른 워크플로우 엔진과의 상호 연동이나, 수행 결과 내역의 감시 및 통계처리를 위한 인터페이스 등도 필요하다. 이러한 문제를 해결하기 위해서 많은 노력이 있어왔으며, 이를 위한 표준을 제정하려는 움직임도 있어왔다. WfMC(Workflow Management Coalition)는 표준 워크플로우 구조를 정의하고, 워크플로우 서비스를 5개의 기능적 인터페이스로 구분한 후, 각 인터페이스를 구성하는 API들을 정의하였다.<sup>[1]</sup> OMG(Object Management Group)는 워크플로우 시스템이 객체로 구성되어 있다고 가정하고, 워크플로우 내부에 존재하는 객체의 종류와 이들의 연관관계, 객체들이 가져야 할 인터페이스의 표준을 정의하였다.

둘째, 프로세스 표현력의 문제이다. 즉, Build-time에 비즈니스 프로세스를 분석하고 이를 모델링 및 정의하는 과정에서 실제 현업에 존재하는 다양하고 예외적인 프로세스들을 제대로 표현하고 관리할 수 있는가 하는 것이다.

셋째, 오류(Failure)에 대한 대응과 이체 따른 시스템의 신뢰성 문제이다. 오류 발생에 따른 대처 기능이 부족하여 높은 신뢰도가 요구되는 업무영역에서 워크플로우 시스템이 적용될 수 없는 경우가 발생한다.<sup>[3],[4]</sup>

이와 같은 문제점들이 그 동안의 워크플로우 시스템의 기능 개선과 기술적 진보에도 불구하고 자주 제기되는 일반적인 문제점들이다. 그리고 이런 문제점들로 인해, 워크플로우 시스템들이 보다 광범위한 영역에 적용되지 못하고 있다.

## 2. XML & Web-based Process Management(X-WPM) system 제안

### 2.1 X-WPM 시스템의 목표

- 광범위한 범위의 워크플로우 중에서 제일 중요한 프로세스 관리 부분의 문제점을 해결할 수 있는 시스템을 개발한다.
- Thin Client 개념을 도입함으로써 장소적, 시스템적 제한 사항을 해결한다.
- XML을 도입함으로써 웹상에서의 복잡한 데이터 전송의 표준안을 도입할 수 있는 기반 마련 및 다른 애플리케이션과의 데이터 통합성을 확인한다.
- 계층적 프로세스 표현으로 프로세스 정의 및 관리의 편리성을 도모한다.

#### 2.1.1 Thin Client

기존의 Thick Client 개념상에서는 업무처리에 장소적, 시스템적 제한이 생기게 되어 프로세스의 정의와 혁신에 한계를 들어낸다.

본 연구에서 구현한 프로세스관리 시스템은 웹을 기반으로 하므로, 이러한 환경은 앞에서 제기된 이질성과 상호 운영성의 문제를 극복하는데 용이하다. 아래의 Fig.2 은 이번 연구에서 구현한 시스템의 개략적 Architecture를 보여주고 있다.

Thin Client 는 웹브라우저, 노트북, 핸드헬드 PC, 휴대폰 등을 활용한 모바일 컴퓨팅에 이르기까지 다양한 업무 환경에서의 업무 처리를 가능하게 하는 것을 골자로 하고 있다.

여기서는 웹기반(Web-based)을 기초로 하여, 장소적, 시스템적 한계사항을 넘어설 수 있는 프로세스 관리 시스템을 구현한다.

#### 2.1.2 XML을 통한 데이터 표준안 마련

XML의 유용함은 인터넷을 통해서 XML 문서가 전달된 후에 그것을 가지고 어떤 일을 할 수 있다는 것을 뜻한다. 인터넷이라는 환경에서 실패한 SGML과 달리 XML은 인터넷에 적용하기 위해서 만들었으며, 가장 큰 설계 목표이기도 하다. (물론 인터넷 이상으로 XML의 적용 범위를 충분히 확대할 수 있다.)

XML이라는 것은 문서를 처리하는 프로그램 작성을 쉽게 할 수 있도록 하기 위해서 읽기 쉽고, 합리적이며 분명한 것을 특징으로 한다. HTML에서는 시작 태그(start tag)와 마침 태그(end tag) 생략이 가

능한 경우를 쉽게 찾아볼 수 있지만, XML에서는 이를 허용하지 않는다. 읽기 쉽고 분명하다는 것은 눈으로 볼 수 있는 내용과 논리상의 정의와 최대한 일치한다는 것을 이야기한다. 또한 사용자 입장에서 만들기 쉬운 문서를 추구하는 것은 HTML, CSS 등 다른 W3C 스펙에서도 찾아볼 수 있는 내용이다. 일반 텍스트 편집기로 그 내용을 보고 고칠 수 있는 것은 물론 WYSIWYG 소프트웨어를 이용해서 유효한 문서를 만들 수 있도록 한다.

태그, 특히 마침 태그를 생략하는 것은 직접 코딩할 때 편리하다고 할 수 있을 지 몰라도 사람의 눈으로 볼 때 명확하게 파악하기 어렵게 만들며, 소프트웨어가 구조를 해석하는데도 좋은 조건이 아니다. 따라서 xml 마크업의 간명함은 최소한의 범위 내에서 보장되어야 한다.<sup>[6]</sup>

복잡한 데이터를 정의하고 전송할 수 있는 특성을 지니는 언어로서 프로세스 관리 엔진도 다른 애플리케이션들과 자료를 주고받기 위해서 XML을 지원해야 한다.

X-WPM 시스템의 프로세스 디자이너 부분과 관리자 부분의 데이터 형식들이 XML 로 정의됨에 따라 복잡한 데이터의 웹상의 전송 및 표현이 간결해질 수 있다. 이러한 XML의 특징은 데이터 통합 및 다른 XML 기반의 애플리케이션과의 밀접한 연동 또는 호환을 가능케 하고 있다. WfMC 측에서도 이미 표준화 작업을 시작하였고, 그 일환으로 Wf-XML Beta 1.0을 2000년 상반기에 발표하였다.

#### 2.1.3 다양한 프로세스의 표현

현업에서 일어날 수 있는 다양하고 변칙적인 프로세스들의 무결성을 보장하면서도 제대로 표현하기 위해서는 기본적으로 다음과 같은 것을 표현할 수 있어야 한다.<sup>[1][5]</sup>

- Sequential Routing : 업무간 선행 관계에서의 기본적인 단위로서 한 업무가 끝나고 나서 다른 업무가 시작되는 것을 나타낸다.
- Parallel Routing : 동시에 여러 군데로 업무가 전달되는 것을 나타낸다.
- Conditional Routing : 업무 규정이 복잡하여 조건에 따라 업무의 진행 순서가 변경되는 것을 나타낸다.
- AND-Split : 한 업무가 끝난 후에 동시에 하나 이상의 모든 후행 업무가 시작되는 관계이다.
- AND-Join : 한 업무가 시작되기 전에 모든 선행 업무가 끝나야 하는 관계이다.
- OR-Split : 한 업무가 끝난 후에 선택적으로 후행 업무가 시작되는 관계이다.
- OR-Join : 한 업무가 시작되기 전에 선택적으로 선행 업무가 끝나야 하는 경우이다.

또한 본 연구에서는 보다 복잡한 형태의 프로세스를 표현하기 위하여 Fig.1와 같이 계층적 형태의 프로세스를 표현할 수 있도록 하였다. 이러한 프로세스의 계층적 구성은 프로세스 관리에 있어서 캡슐화(encapsulation)를 가능하게 하여, 복잡한 프로세스를 계층적으로 잘 나타낼 수 있을 뿐만 아니라, 상위 프로세스만으로도 쉽게 전체 프로세스를 파악할 수 있다는 이점이 있다.

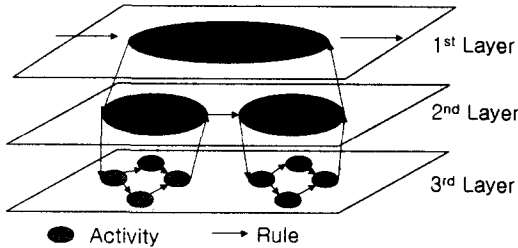


Fig. 1 Layered Structure of Process

## 2.2 X-WPM 시스템 구조

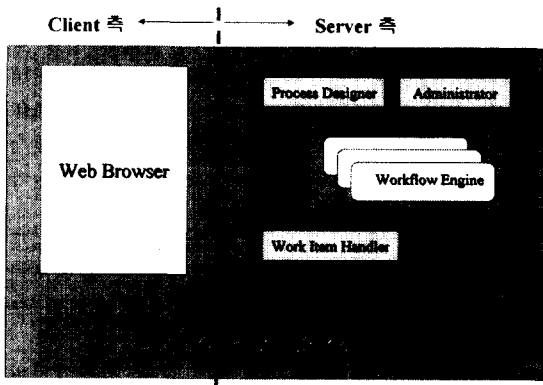


Fig. 2 A X-WPM system architecture

Fig.2 의 시스템 Architecture에서 볼 수 있듯이 기본적으로 웹브라우저(Web Browser)를 사용함으로써 편안한 GUI, 장소에 구애받지 않는 업무처리를 할 수 있다.

본 시스템에서 지원되는 기능 모듈로는 프로세스 디자이너(Process Designer)와 관리자도구(Administrator)가 있다. 프로세스 디자이너에서는 기업내의 업무 프로세스를 디자인하고 정의하고, 관리자도구에서는 기업내의 업무 프로세스, 양식과 권한 등을 관리하게 된다.

## 3. 결론

본 연구에서는 기존의 워크플로우 시스템들이 가지고 있던 여러 가지 문제점들을 짚어보고, 이 중에서 가장 중요하다고 할 수 있는 프로세스 부분에서의 문제점을 개선한 X-WPM시스템을 제안하였다. 프로세스를 모델링하고 정의하는 과정에서 보다 다양한 프로세스 표현하고 관리하기 위한 방법을 연구하였다. 또한 프로세스관리 시스템의 개발 환경으로 웹을 선택하여 최근에 각광받고 있는 웹기반 프로세스관리 시스템이 가질 수 있는 이점과 그 가능성을 알아볼 수 있었다.

- VE(Virtual Enterprise) 구현을 위한 인프라로서 분산 기업 환경에서 프로세스 처리의 기반, 즉 비즈니스를 위한 운영체제와 같은 역할을 하기 위해서 Web-enabled가 아닌 Web-based로 개발함이 타당하다.

- 복잡한 데이터의 정의 및 표준화로 인한 애플리케이션들과의 통합성 및 호환성 보장을 위하여 XML 기술사용이 필수적이다.

- 프로세스의 계층적 구성을 통한 캡슐화는 프로세스를 정의하고 관리하는데 이점이 있다.

앞으로 남은 과제는 표준화 작업의 완성도를 높이는 작업과, 아직 구현 중에 있는 X-WPM시스템의 완성도를 높이고, 이를 실제 프로세스에 적용해 보는 과정을 통해서 시스템의 기능을 개선하고 신뢰도를 높이는 일이다.

## 참고문헌

1. David Hollingworth, Workflow Mgt. Coalition : The Workflow Reference Model, WfMC 1994
2. John A. Miller, Amit P. Sheth, Krys J. Kochut and Devanand Palaniswami. "The Future of Web-based Workflows", Technical Report, Large Scale Distributed Information Systems Lab(LSDIS). 1997.
3. Alberto Silva, Miguel Mira da Silva, Jose Delgado. "Conceptual Frameworks for Web Information Systems Development". Technical Report, Technical University of Lisbon. 1997.
4. John A. Miller, Amit P. Sheth, Krys J. Kochut and Devanand Palaniswami. "The Future of Web-based Workflows", Technical Report, Large Scale Distributed Information Systems Lab(LSDIS). 1997.
5. Bill Curtis, Marc I. Kellner and Jim Over, "Process Modeling.", COMMUNICATIONS OF THE ACM, Vol33. No.9, pp. 132-173, September 1992
6. ELLIOTTE RUSTY HAROLD, XML BIBLE, 정보문화사, 2000