제22회 한국정보처리학회 추계학술발표대회 논문집 제11권 제2호 (2004. 11)

B2B 전자거래를 위한 워크플로우 시스템과 어플리케이션 연동 방법

채정숙° 황재각 이용준
전자통신연구원 U-Logistics 연구팀
{chaejs°,jghwang,yil }@etri.re.kr

Interoperation Method Of workflow system and Application based on B to B

Jeongsook Chae° Jaegak Hwang,Yongjun Lee
e-Logistics team, Postal Technology Research Center, ETRI

요 약

최근에는 B2B 시스템 통합 및 전략적 제휴가 상거래 형태를 변화시키고 있고, 이에 따라 관련 기업들의 효율성이 증가됨으로써 기업간 시스템들의 연동이 중요한 문제로 부각되고 있다. 이에 본 논문에서는 워크플로우 시스템과 어플리케이션간의 연결을 통한 프로세스 자동화는 Wf-XML HTTP 바인딩을 사용함으로써 가능하며, 워크플로우 시스템과 어플리케이션의 연동을 위해 Wf-XML 문서를 HTTP 프로토콜을 사용하여 전송하는 방법을 논하며, 실제 구현을 통해 연동이 가능함을 보인다.

1. 서 론

오늘날 기업내 업무 형태가 다양해지고 기업간 전자상거래가 일반화되면서 워크플로우에 대한 관심이 높아지고 있다. 워크플로우 시스템은 문서, 정보 등 한 사용자가 다른 사용자로 일련의 업무 처리 절차의 규칙에 의해 전달되는 일련의 작업흐름을 정의하며 정의된 비즈니스 프로세스의 자동화를 실현함으로써 완성된다. 이러한 워크플로우 관리 시스템을 통해 기업 내 프로세스의 업무 성능 향상을 얻었으나, 워크플로우 시스템도 업무조직이 복잡해지고 분산화되어 감에 따라 업무의 흐름 및 정보에 대한 공유 및 관리에 대한 많은 문제점들이 노출되어 있다. 현대의 기업 정보 시스템은 기업 업무 처리 애플리케이션의 모듈화, 통합성, 재 사용성, 상호연동이 필요하게 되었으며, 기업 업무 처리 형태의 다양성으로 기업내부는 물론 외부에까지 연계되면서 애플리케이션과의 상호 연동의 중요성이 증대되었다. 현재 모든 기업 내· 외부의 업무는 조직의 고유 업무 프로세스를 가지고 있다. 업무 프로세스란 조직 내의 개인 혹은 부서들간의 외부 조직에서 이루어지는 의사결정에 필요한 일련의 과정을 말하는 것으로 이러한 업무 프로세스를 효율적으로 관리하고, 외부 환경에 신속히 대응하기 위해 업무 프로세스의 관리 및 유지가 필요하다. 워크플로우 관리 시스템은 조직의 신속하고 정확한 의사결정을 지원하기 위하여 효율적인 업무 프로세스를 정의하고 이를 관리하며 업무 분석의 최적화를 이룰 수 있도록 지원하는데 목적이 있다. 워크플로우 시스템은 기업 내의 업무 프로세스를 진행하는 데 있어 시스템/애플리케이션 시점 또는 어떤 이벤트에 따라서 어디에서 어디로 업무가 진행되어야 하는 지를 정의하고 운용할 수 있는 기능을 담당한다. 본 논문에서는 이러한 시스템과 애플리케이션 상호간의 데이터 교환을 위한 데이터 연동 방법을 정의한다.

본 논문의 구성은 다음과 같다. 2 장에서는 워크플로우 시스템과 상호운영에 관계된 reference 모델에 대한 관련연구에 대해 살펴보고, 3 장에서는 애플리케이션과 연동을 위한 연관 데이터 연동의 설계 및 구현 방법에 대해 논한다. 4 장에서는 결론 및 향후 연구방향을 제시한다.

2. 관련 연구

2.1 WfMC 의 상호운용성

워크플로우 시스템은 WfMC(Workflow Management Coalition)과 OMG(Object Management Group)의 표준에 근거한 많은 시스템들이 여러 벤더들에 의해 개발되고 있다. 워크플로우 시스템은 다양한 비즈니스 프로세스 자동화에 공통적으로 요구되는 기능들을 미리 갖추어 놓음으로써 비즈니스 프로세스의 자동화를 컴퓨터상에서 효율적으로 구현하도록 도와준다. 최근에는 웹을 통한 서비스가 급격하게 증가하게 됨에 따라 기업내의 워크플로우 관리 시스템뿐만 아니라 외부 기업간의 거래가 이루어지기 위한 이 기종간의 워크플로우관리 시스템의 상호작동이 필요하게 되었다. 이에 워크플로우 관련 단체의 모임인 WfMC(Workflow Management Coalition)에서는 서로 다른 벤더들의 서로 다른 워크플로우

관리 시스템들간의 통신을 가능하게 하기 위하여 워크플로우 시스템에 대한 표준 및 워크플로우 시스템의 인터페이스와 상호 운영성에 대한 표준을 제정하였다[2]. 그림 1 은 WfMC 에서 제안한 워크플로우 reference model 을 보여준다.

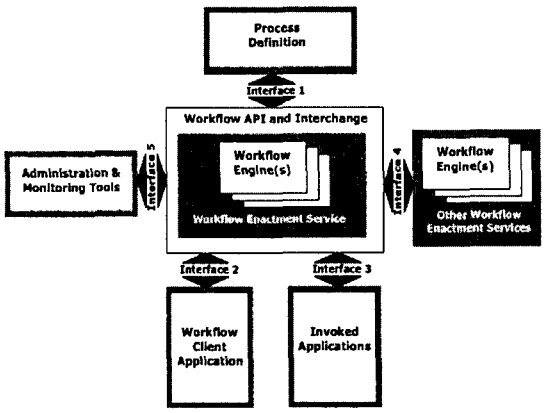

그림 1 WfMC 워크플로우 참조 모델[1

워크플로우 참조 모델은 주요 워크플로우 시스템의 주요 구성요소와 인터페이스를 보여주며, WfMC는 워크플로우 엔진과 나머지 구성 사이에 대하여 표준을 제정하고 있으며, 이들 중 인터페이스 4에서는 다른 워크플로우와의 상호운용성에 필요한 요소들을 정의하고 있다. 또한 WfMC reference model은 워크플로우 시스템들간에 프로세스 관련 정보를 주고 받고, 프로세스를 연계하기 위한 표준과 API들의 집합에 대한 표준을 정의하고 있으며, 서로 다른 워크플로우 엔진들이 상호 연동하기 위해 필요한 기능들의 규약을 설명한다. 현재 WfMC에서 제시한 워크플로우 상호운용성을 다루고 있는 워크플로우 표준-인터페이스 4 (Workflow Standard-Interface 4)[2]에 대한 실질적인 구현 방안으로 MIME을 이용한 방법과 Wf-XML[4]을 이용한 방법이 있다. 이러한 상호운영을 위해서 어플리케이션과의 연동이 이루어져야 하는데, 이때 프로세스를 처리하기 위해서 연관 데이터 처리가 필요하게 된다. 본 논문에서는 이러한 어플리케이션과의 연동을 위해 수행되어야 하는 연관데이터 처리 방법에 대해 논한다.

3. 워크플로우 시스템간의 상호운용

3.1 Interoperability method

본 논문에서 제안한 기업간 상호 운용이 가능한 워크플로우 시스템은 WfMC에서 제안한 표준 인터페이스 4에 의해 설계되고 두 회사의 워크플로우 시스템을 직접 연결은 Wf-XML HTTP 바인딩을 사용하였다. Workflow 엔진과 admin client와 runtime client 모듈인 웹 서버 그리고 business process를 모델링 할 수 있는 modeler로 구성된다. 워크플로우 엔진의 gtmig는 BPMT 모델러와 admin tool과 통신하기 위한 엔진으로 CORBA/IIOP를 사용하며, wlhdr은 runtime client tool과 통신하기 위하여 사용된다. tmif는 리모트 요청, 응답 정보를 Wf-XML로 변환하기 위한 모듈인 Wf-XML generate와 Wf-XML을 이용하여 상호운용성을 위한 데이터를 Wf-XML에서 내부 데이터로 변환하기 위한 Wf-XML translator와 통신한다. Remote engine manager는 상호운용성을 위한 모듈로서 데이터를 받기 위한 "HTTP Server Component"와 전송하기 위한 "HTTP Client Component"를 포함한다. "HTTP Client Component"는 Multi-threaded pool구조를 가짐으로써 여러 프로세스에 대한 요청을 동시에 수행할 수 있다. Inter-Workflow 시스템은 분산환경으로 CORBA 기반으로 설계되었으며, Runtime Client 와 엔진간에 통신을 위하여 사용하는 CORBA/IIOP 통신을 사용하고 Admin Tool 과 엔진, BPMT와 엔진간의 통신은 원격외부 엔진과 통신을 해야 하므로 HTTP를 사용하여 통신한다. 그림 2는 워크플로우 시스템의 상호운용 시스템 구성도를 나타낸다.

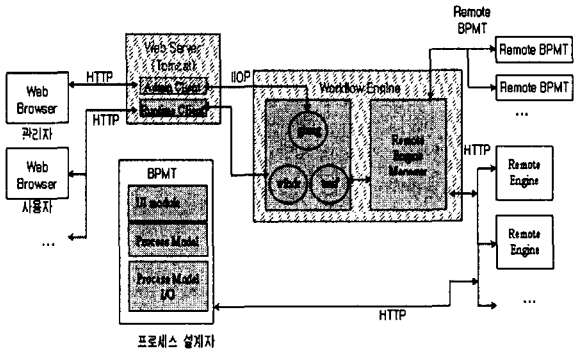

그림 2 워크플로우 시스템간 상호운용 모델

기업간 상호 운영 시스템의 주요 기능은 다음과 같다.

- Chained process 상호운용성 모델 지원
- Nested sub-process 상호운용성 모델 지원
- 외부 엔진에 수행을 의뢰한 프로세스의 상태 모니터링 기능
- 외부 엔진에서 수행되는 프로세스 상태의 모니터링

상호운용 프로세스는 상호운용성을 위한 연결된 chained model of interoperability와 Nested sub-process model of interoperability을 지원한다. chained model of interoperability 모델에서는 워크플로우 엔진 A에서 수행중인 프로세스가 워크플로우 엔진 B 상의 프로세스를 생성 및 수행하도록 한다. 워크플로우 엔진 A 상의 프로세스는 바로 종료하거나 또는 다음 단위업무를 수행할 수 있다. 단지 워크플로우 엔진 B에 새로 생성된 프로세스와 더 이상 상관이 없다고 가정한다.

Nested sub-process model of interoperability 모델에서는 워크플로우 엔진 A 상에서 수행중인 프로세스가 워크플로우 엔진 B 상의 프로세스를 생성 및 수행을 유발하고 이 프로세스가 종료할 때까지 기다린다.
시스템은 다음과 같은 예외사항을 갖는다. 첫 번째로 외부

워크플로우 엔진은 같은 버전의 WfMC에서 제안한 인터페이스 4 표준에 맞게 설계되어 구현된 워크플로우 엔진이어야 하며, 통신 메커니즘은 asynchronous mechanism, synchronous mechanism을 지원한다. 또한 통신 프로토콜은 HTTP를 사용한다. 상호운용성을 위한 데이터 교환은 한 기업의 엔진과 원격 엔진간의 상호운용성을 위한 정보교환과 BPMT와 원격 엔진간에 프로세스 정의정보 요청 및 응답을 나타내는 부분으로 나타낼 수 있다.

3.2 Data Interoperation Processing Method of Interoperability

워크플로우 시스템간의 상호 연동을 위해서 어플리케이션의 등록과 함께 연관 데이터 연동이 필요하게 된다. workflow relevant data는 참조 데이터라고 하며 워크플로우 엔진에 접근하기 위한 애플리케이션 타입으로 프로세스 진행방향 결정이나 워크플로우 응용 프로그램에 의해 생성/갱신된 데이터를 기반으로 하는 워크플로우 엔진은 내부의 control operation 들에 의해 데이터 처리를 담당한다. 워크플로우 엔진으로부터 연관 데이터를 받아 업무수행 결과를 연관 데이터로 반환 받아 Worklist에 제시된 workitem(activity) 처리 및 결과를 통보한다. 워크플로우 애플리케이션 데이터는 오직 워크플로우 애플리케이션에 의해서만 직접 연산 수행되며, 워크플로우 엔진은 다른 애플리케이션들의 워크플로우 프로세스 안에 포함된 다른 활동 정에서 invoke된 다른 애플리케이션들 사이에서의 데이터와 같은 것들에서 전달되기 위한 책임을 가지고 있다. 본 연구에서의 연관데이터 처리 방법은 다음과 같다.

기본적으로 프로세스 전이는 단위업무, 프로세스 모두 상태전이에 따라 진행된다 즉, 정의된 프로세스 모델에 따라 state machine의 구조인 생성 -> 실행 -> 통제의 구조를 따른다. 그림3은 active-> run-> complete 등의 예를 들어 설계한 데이터 연동 모델이다. 여기서 active는 Process Instance의 관점에서 프로세스를 구성하는 처리되어야 할 단위 업무를 의미한다. 단위 업무가 complete 되게 되면 Input/Output 파라미터로 매핑 된 후Output 파라미터를 프로세스에 전송하여 연관 데이터에 적용하게 되며 이런 과정에서 데이터 전송이 일어나며 이러한 과정에서 데이터는 엔진 DB 에 저장된다. 단위업무 Input/Output 파라미터가 모델에서 정의된 데로 단위업무 시작 또는 종료 시에 연관 데이터로 매핑 된다

3.3 애플리케이션 연동

여기에서 말하는 application 은 java 응용어플리케이션으로 등록되는 프로그램을 의미하며 현재는 자바프로그램으로 한정하여 고려한다. JSP application 중에서 JSP 폼으로 등록되는 응용프로그램을 말하며 이 역시 현재 런타임클라이언트가 돌고 있는 tomcat 서버에 JSP 로 등록하여 호출이 가능하다. 단, application이 jsp로 되어 있지 않으면 고려하지 않는다.

가. Java application 작성 및 등록

워크플로우 시스템과 어플리케이션의 연동을 위해 우선적으로 수행되어야 하는 것이 어플리케이션 작성 및 등록 과정이다. 어플리케이션의 작성 및 등록을 하기 위해서는 다음과 같은 절차를 따른다.

- dreamflow.engine.worklist.DfWorkerServant을 Extends 한 클래스를 작성한다.
- dreamflow.tool.runtime.DfRTClient을 Extends 한 클래스를 작성한다.
- dreamflow.tool.runtime.DfWorkClient을 Extends 한 클래스를 작성한다.
- dreamflow.tool.runtime.DfRuntimeManager을 Extends 한 클래스를 작성한다.
- 작성한 클래스를 jar로 묶는다.
- Application 을 호출할 수 있는 URL로 이동하여 저장한다.
- 모델 정의에 Application 정보를 추가한다.

그림 3은 java로 작성된 어플리케이션의 등록 과정을 보여준다.

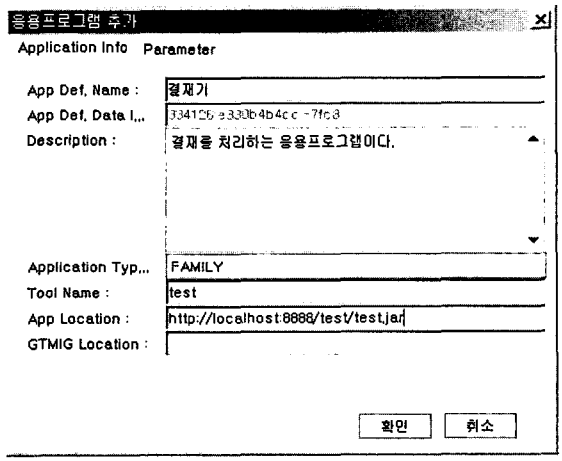

그림 3 어플리케이션 등록

나. JSP application 작성 및 등록

워크플로우 시스템과 어플리케이션의 연동을 위해 JSP application의 작성 및 등록을 위해서는 다음과 같은 절차를 따른다.

- JSP 파일을 작성한다.
- JSP 파일은 아래 샘플 예제와 같은 스펙을 따른다.
- 작성한 JSP 파일을 Tomcat 서버의 JSP 디렉토리에 넣는다.
- 모델 정의에 Application 정보를 추가한다.
- Application Location 에 JSP 파일을 호출할 수 있는 URL 을 적는다.

그림 4는 JSP로 작성된 어플리케이션의 등록 화면이다.

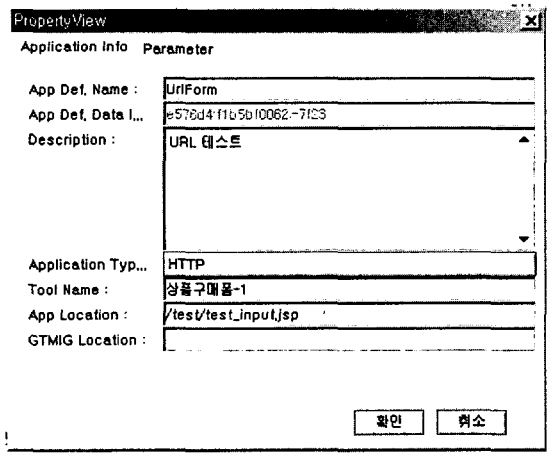

그림 4 어플리케이션 등록

다음은 자바 어플리케이션의 작성 알고리즘을 보여준다.

Interoperability algorithms

```
/*
 *      <code>DfApplForTest</code>      클래스는
Dreamflow Engine Test용 Application 클래스이다.
*/
public      class      DfAppForTest      extends
DfSimpleFamilyAppl {
    어플리케이션의 주요 실행부분
    @param context      워크플로우 엔진으로부터 주
    어지는 relevantData들이다.
    수행시에 참조할 수 있다.
    @return      수행결과물을 반환한다. context와 같
    은 구조를 취한다.
    @throws DfApplException 수행과정에서 오류가
    발생했을 경우 처리한다.
        }
        public DfDataTable execute(DfDataTable
context) throws DfApplException {
        // 여기에서 수행하고자 하는 로직을 작성한다.
                return context;
        }
}
```

4. 결론 및 향후 연구

본 논문에서는 WfMC(Workflow Management Coalition)에서 제안한 워크플로우 표준기반 기술 인터페이스 4 및 reference model에 기반하여 워크플로우 시스템과 어플리케이션간의 상호운영성을 지원하기 위한 상호 운용 방법에 대해 논하였다. 워크플로우 시스템과 어플리케이션간의 상호운영성을 제공하기 위한 Wf-XML은 XML을 이용하여 워크플로우간 통신 메시지를 구성하며, 표준화된 DTD(Data Type Definition)을 정의한 고정된 언어를 사용하여 메시지를 전송한다. 본 논문에서 제시한 웹 기반 워크플로우 시스템은 admin client와 runtime client 모듈인 웹 서버 그리고 business process를 모델링 할 수 있는 모델러로 구성되며 admin Tool 과 엔진, BPMT와 엔진간의 통신은 원격외부 엔진과 통신을 해야 하므로 HTTP를 사용하여 통신한다. 그러나 기존의 여러 legacy system 과의 수주 및 발주와 같은 일괄처리 업무를 위해서는 비동기적인 상호작용이 필요하다. 그러나 TCP/IP를 기반으로 한 HTTP 프로토콜은 기본적으로 동기적 상호 작용만 가능하므로 이를 응용 프로그램 레벨에서 비동기적으로 처리할 수 있어야 한다. 따라서 워크플로우 시스템에서는 동기적 및 비동기적으로 상호작용을 처리할 수 있는 모듈이 향후 연구사항으로 남아있다.

[참고문헌]

[1] Workflow Management Coalition The Workflow Reference Model http://www.wfmc.org/standards/docs/tc003v11.pdf

[2] Workflow Management Coalition, "Workflow Standard – Interoperability Abstract Specification," Workflow Management Coalition, 20, October 1996.

[4] Workflow Management Coalition, "Workflow Standard – Interoperability Wf-XML Binding," Workflow Management Coalition, 20, July 1998.

[5] Workflow Management Coalition, "Workflow Standard – Interoperability Abstraction Specification," WfMC-TC-1012, Version 1.0, October 1996.

[7] Zhou Shijie, Qin Zhiguang, Liu Jinde. Study on interoperability of workflow management system. [Journal Paper] *Journal of University of Electronic Science & Technology of China, vol.31, no.2, April 2002, pp.145-50. Publisher: Editorial Department of J. of UEST of China, China.*

[8] Distributed Applications and Interoperable Systems II. IFIP TC6 WG6.1 Second International Working Conference on Distributed Applications and Interoperable Systems (DAIS'99). [Conference Proceedings] *Kluwer Academic Publishers. 1999, pp.xiv+458. Norwell, MA, USA.*

[9] Paul S, Park E, Hutches D, Chaar J. RainMaker: workflow execution using distributed, interoperable components. [Conference Paper] *Research and Advanced Technology for Digital Libraries. Second European Conference, ECDL'98. Proceedings. Springer-Verlag. 1998, pp.801-18. Berlin, Germany*