

Fragment-Driven 워크플로우 모델링 방법론[☆]

A Fragment-Driven Workflow Modeling Methodology

문 기 동* 김 형 목** 김 광 훈*** 백 수 기****
Ki-Dong Moon Hyung-Mok Kim Kwang-Hoon Kim Su-Ki Paik

요 약

최근에 업무효율성 향상에 대한 요구가 정보관리기술 분야의 핵심이슈로 대두되면서 많은 기업들이 워크플로우 관리 시스템의 도입을 통한 업무프로세스 자동화에 주력해 오고 있다. 특히, 기업내부 업무프로세스 즉 워크플로우 프로세스에 대한 자동화에 이어 기업간의 업무프로세스 상호연동을 필요로 하는 비즈니스 모델이 전자상거래, 전자무역 등의 급속한 확산과 더불어 활성화되면서 새로운 형태의 워크플로우 프로세스인 기업간 글로벌 워크플로우 모델을 위한 워크플로우 관리 시스템의 확장을 필요로 하고 있다.

본 논문에서는 기업간 글로벌 워크플로우 모델을 효과적으로 지원할 수 있는 새로운 형태의 기업간 워크플로우 모델링 방법론으로서 Fragment-driven 워크플로우 모델링 방법론을 제안한다. Fragment-Driven 워크플로우 모델링 방법론은 실제 조직에서 액티비티를 수행하게 될 각 참여자가 자신의 조직에 연관된 액티비티들만을 정의하게 한 후에 그들의 통합을 통해 하나의 기업간 글로벌 워크플로우 프로세스의 모델링을 가능하게 하는 Bottom-Up방식이다. 이 방식은 각 조직의 참여자들이 자신들의 정보만을 이용하여 모델링하는 방법이기 때문에 각 조직의 독립성을 유지하면서도 모델링 작업을 단순화 시킬 수 있는 장점을 기대할 수 있다. 본 논문에서는 이러한 Fragment-driven 워크플로우 모델링 방법론의 이론적 배경과 내용을 기술하며, 이 방법론을 기반으로 하는 협업적 스몰레인 워크플로우 모델링 시스템을 구현한다.

Abstract

Many organizations have been recognizing the necessity of workflow automation technologies according to the rapid expansion of business process oriented applications, such as enterprise resource planning, customer relationship management, electronic approval management, and so on. Thus, they have started adopting workflow management systems as an essential technological solution for their workflow processes. However, we need some technological extensions and improvements on them in order to accommodate a new type of workflow processes, which is called cross-organizational global workflow processes that require a certain level of collaborations between the organizations engaged in the global workflow processes.

Fragment-driven workflow modeling methodology is a Bottom-Up methodology composing a global workflow by defining each organization's own activities, which is called a fragment, through a realtime cooperative system. The approach is able to not only simplify the modeling work but also keep each organization's independence in modeling a global workflow. In this paper, we describe the fragment-driven workflow modeling methodology and realize the methodology through the implementation of a cooperative swimlane workflow modeling system.

↳ Keyword : Fragment-Driven Workflow Modeling Methodology, Cooperative Global Workflow Model

* 정 회 원 : 경기대학교 대학원 전자계산학과 박사 수료
kdmooon@kyonggi.ac.kr(제 1저자)

** 종신회원 : 경기대학교 대학원 전자계산학과 박사 수료
webmaster@bazi.pe.kr(공동저자)

*** 종신회원 : 경기대학교 정보과학부 교수
kwang@kyonggi.ac.kr(공동저자)

**** 정 회 원 : 경기대학교 정보과학부 교수
skpaik@kyonggi.ac.kr(공동저자)

☆ 이 논문은 2002년도 한국학술진흥재단의 지원에 의하여 연구되었음. (KRF-2002-0031-D00247)

[2004/07/23 투고 - 2004/08/09 심사 - 2004/10/11 심사 완료]

1. 서론

1.1 연구배경

워크플로우 응용사례의 급속한 확대단계(Maturity Phase)에 앞서 그의 선행연구인 워크플로우 모델링 기법 및 운용환경에 대한 연구개발이 시급히 요구된다. 특히, 국내외적으로 전자상거래를 비롯한 고객관리기술(CRM), 공급망 및 가치사슬망 관리기술(SCM), 데이터 및 응용 프로그램 통합기술(EAI), 기업자원계획기술(ERP) 등과 같은 최첨단 정보기술의 핵심 기반 기술로서 워크플로우 기술이 매우 중요하게 인식되고 있고, 워크플로우 응용기술의 적용사례 측면에서도 그 수가 급속하게 증가하고 있기 때문에 워크플로우 기술을 적용하는데 있어서 반드시 필요한 워크플로우 모델링 기술 및 워크플로우 기반 소프트웨어 개발방법론에 대한 연구가 매우 필요한 시점이다. 그리고, 차세대 전자상거래 기술로 세계적으로 인정받고 있는 기업간 워크플로우기술(Cross-Organizational Workflow)의 핵심인 기업간 워크플로우 모델과 그의 효율적인 운용환경을 제공하는 것이 국내외적으로 주목을 받고 있는 연구개발 분야이다. 이에 본 논문에서는 워크플로우기술의 대표적인 응용분야인 전자상거래분야의 급속한 확산과 더불어 최근의 연구개발이슈인 인터넷기반 기업간 전자상거래(B2Bi)를 효과적이고 효율적으로 구현하고자 Bottom-Up방식의 Fragment-Driven 워크플로우 모델링 방법을 제안한다.

1.2 연구내용

본 논문에서는 Top-Down 접근방식이라고 할 수 있는 기존의 글로벌 워크플로우 모델링 방식이 아닌 Bottom-Up방식의 Fragment-Driven 글로벌 워크플로우 모델링 방법을 제안하고 이를 바탕으로 하는 새로운 형태의 글로벌 워크플로우 모델링 시스템을 설계 및 구현하는 것을 그 핵심 내용으

로 한다. 따라서, 본 논문의 연구 내용은 다음과 같은 핵심내용으로 구성된다.

- ▶ 협동 스웸레인 워크플로우 모델 고찰
- ▶ Bottom-Up방식의 Fragment-Driven 워크플로우 모델링 방법론 제안
- ▶ 협동 스웸레인 워크플로우 모델링 시스템 설계 및 구현

2. 이론적 배경 및 관련 연구

2.1 워크플로우 모델링 이론

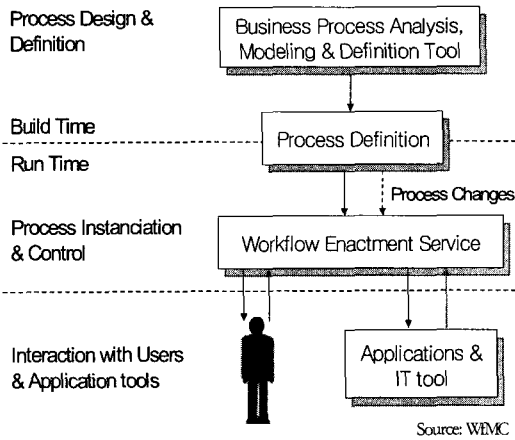
일반적으로 워크플로우 관리 시스템은 모델링(Modeling)과 업무 규칙의 두 가지 구성 요소로 이루어져 있다. 모델링 구성요소는 모델러(modeler)가 실질적으로 워크플로우 프로시저(Workflow Procedure)들을 실행 하는데 필요한 모든 정보와 함께 워크플로우 프로시저들을 관리, 정의, 분석하는 일을 가능하게 하며, 워크플로우 프로시저는 최종적으로 관리 시스템의 업무규칙에 의해 실행된다.

워크플로우 모델은 일반적으로 조직의 모습을 할당업무(task), 업무자(Actors), 역할(Role), 단위 업무(Activity)와 자료저장소(Repository)의 모습으로 표현된다. 워크플로우 모델은 컴퓨터로 표현될 때 업무 프로시저의 생성, 변화 및 시물레이션 작업 등을 할 수 있다.

모델링 도구는 수학적인 의미를 내포하고 있어서 실제로 모델링 된 워크플로우 결과를 실행해보지 않더라도 모델링 된 상태에서 바로 분석이 가능해야 한다. 워크플로우 모델링 도구는 ICN(Information Control Net)으로 표현될 수 있다.

2.2 워크플로우 관리 시스템의 빌드타임 구성요소

워크플로우 프로세스는 일련의 단위업무로 구성되며 프로세스를 정의하기 위해서는 단위업무 정의가 선행되어야 한다. 단위업무 정의에서는 단위



〈그림 1〉 워크플로우 관리시스템의 구성요소

업무 구성요소인 참여자(사람, 에이전트 등의 객체), 자원, 내용(단위업무 수행을 위한 어플리케이션 혹은 단위 업무의 수행사항의 지시내용)을 정의한다.

단위 업무가 모두 정의되면 이들을 연결하기 위한 프로세스의 경로, 즉 단위업무의 수행 경로를 정의하고 경로 설정 시 사용 될 참조 데이터들을 정의한다. 프로세스에 사용될 데이터들이 정의 되면 각각의 단위 업무에서 어떠한 데이터를 참조하여 수행할 것인가 하는 연관 관계를 정의한다. 단위 업무와 이들의 수행경로, 그리고 단위업무 수행 시 사용될 데이터들의 정의를 마치면, 하나의 워크플로우 프로세스가 완성된다[4-7].

2.3 WfMC 레퍼런스 모델

WfMC에서는 그림 2와 같이 참조 모델을 정의하였다. 참조 모델은 워크플로우 엔진과 내부 동작을 지원하는 5가지 API를 각각 정의하고 있다.

그림에서 보듯이 5개의 인터페이스는 각각 워크플로우 프로세스 정의에 관한 인터페이스 1, 워크플로우의 수행을 위한 사용자 또는 응용 프로그램의 호출을 위한 인터페이스 2 & 3, 서로 다른 워크플로우 관리 시스템간의 상호 운용을 위한 인터페이스 4, 그리고 워크플로우 관리 시스템 상에

서 프로세스의 관리 및 진행 상태를 모니터링 하기 위한 인터페이스 5 등이다.

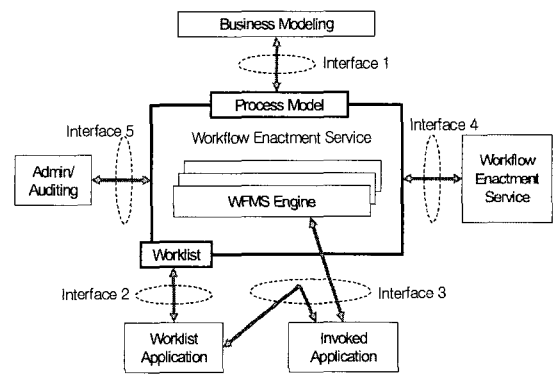
본 논문에서 제안하는 모델링 방법은 인터페이스 1을 통하여 자료를 주고받는데, WFMS Engine_1과 WFMC Engine_2에 모델링 자료를 전달해 주면 두개의 워크플로우 엔진들끼리 인터페이스4를 이용하여 상호 연동할 수 있다.

3. 모델링 시스템

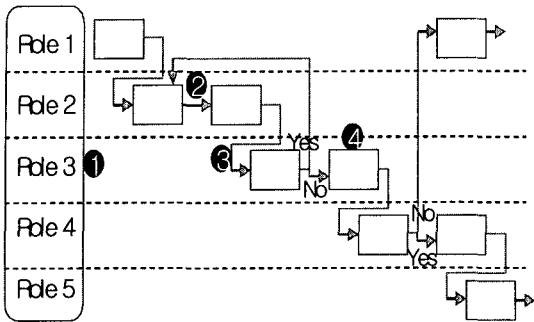
3.1 스웜레인 워크플로우 모델

스웜레인 다이어그램(Swimlane Diagram)은 조직 내에서 발생하는 일련의 업무흐름 표현 방법으로서 Fragment Driven Workflow 모델링에 적합한데 누가, 무엇을 그리고 언제에 초점이 맞춰져 있기 때문에 이해나 숙련에 관계없이 단순하게 표기할 수 있다. 또한 시작부터 끝까지, 그리고 As-is Process의 이해와 To-be Process를 묘사하고 설계 하는데 완전한 비즈니스 프로세스를 보여준다.

무엇보다도 정보의 흐름을 한눈에 볼 수 있기 때문에 본 논문에서는 스웜레인 워크플로우를 확장하여 Fragment-Driven 글로벌 워크플로우 모델링 방법에 적용할 것이다. 그림 3은 스웜레인 다이어그램의 예를 든 것인데 Label된 스웜레인을 갖고 있는 프로세스에서 각각의 액터(Actor)를 나타내었다.



〈그림 2〉 WfMC 레퍼런스 모델



〈그림 3〉 액터, 태스크, 플로우, 핸드오프로 연결한 스윘레인 다이어그램

그림 3을 설명하면 다음과 같다.

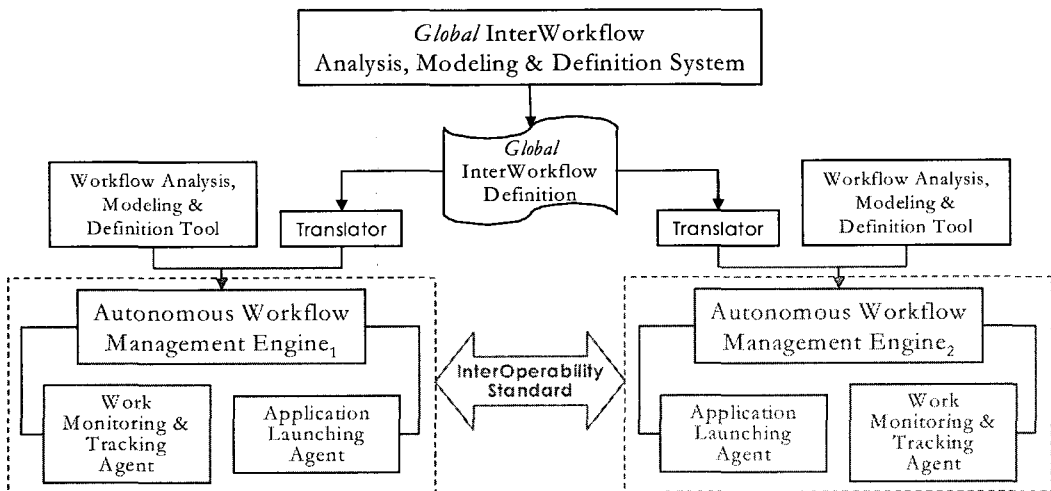
- ▶ **롤(Role):** 각 액터들을 나타낸 것인데 역할을 의미하기도 하며 ①번과 같이 표현할 수 있다. 각 액터들은 자기 자신에게 주어진 역할만 수행한다.
- ▶ **Flow:** 한 액터가 하나의 스텝(Step)에서 다음 스텝으로 일을 전달 하는 것을 나타내는데 ②번 화살표로 표현한다.
- ▶ **Handoff:** Flow의 특별한 종류인데 한 액터의 스윘레인에서 다른 액터의 스윘레인으로 일을 전달하는 것을 나타내며 ③번 화살표로

표현한다.

- ▶ **스텝(Step):** 액터에 의해서 수행되는 일이나 일들의 집합을 나타낸다. 스텝들은 수행되는 액터의 레인(Lane)에서 ④번처럼 Label된 박스로 보여진다. 또한 스텝은 여러 액터와도 관계될 수 있다.

3.2 인터워크플로우 모델(Top-Down방식)

인터워크플로우 매커니즘에서 비즈니스 프로세스 정의는 조직이 서로 분리된 상태에서 프로세스 정의도구인 모델러가 비즈니스 프로세스를 관리하는 형태이다. 각 시스템간의 프로세스 정의에 대한 해석은 트랜스레이터(Translator)를 두어 관리한다. 인터워크플로우의 정의 데이터는 트랜스레이터에 의해 호출된 각 시스템의 프로세스 정의 데이터를 변환시키는 역할을 한다. 그러나 한 사람이 모든 프로세스를 상세하게 정의하도록 규정하고 있는 액티비티 기반의 Top-down 방식이기 때문에 초대형 워크플로우에서 한 사람이 모든 프로세스를 정의한다는 것은 사실상 불가능 하다[8]. 이러한 문제점을 해결하기 위하여 조직의 독립성을 유지 하면서, 보다 효율적으로 기업간에 위



〈그림 4〉 인터워크플로우 관리 시스템

크플로우 모델을 지원할 수 있는 방법이 새로운 Bottom-Up방식의 Fragment Driven 글로벌 워크플로우 모델링 방법이다.

3.3 Fragment-Driven 워크플로우 모델(Bottom-Up방식)

프로세스 정의에 참여하는 조직이 자신의 액티비티를 먼저 정의하고 나서 전체 프로세스를 완성시켜 나가는 것을 Bottom-up 방식이라 한다. 이러한 방식에서는 각 룰의 그룹이 직접 프로세스 정의에 참여할 수 있다.

프로세스 정의에 참여하는 그룹은 전체 프로세스를 고려하지 않고 자신의 룰에 해당하는 액티비티들만 고려하기 때문에 프로세스 정의가 용이하고 안정적이다.

또한 Fragment-Driven 글로벌 워크플로우 모델링 방식은 조직의 독립성을 유지하면서도 효율적으로 기업간에 워크플로우 모델을 지원할 수 있다.

스윙레인 모델을 Fragment-Driven 방식의 모델링에 사용하는 이유도 각 액티비티로부터 전체 프로세스를 모델링 하는데 있어서 현재 정의하는

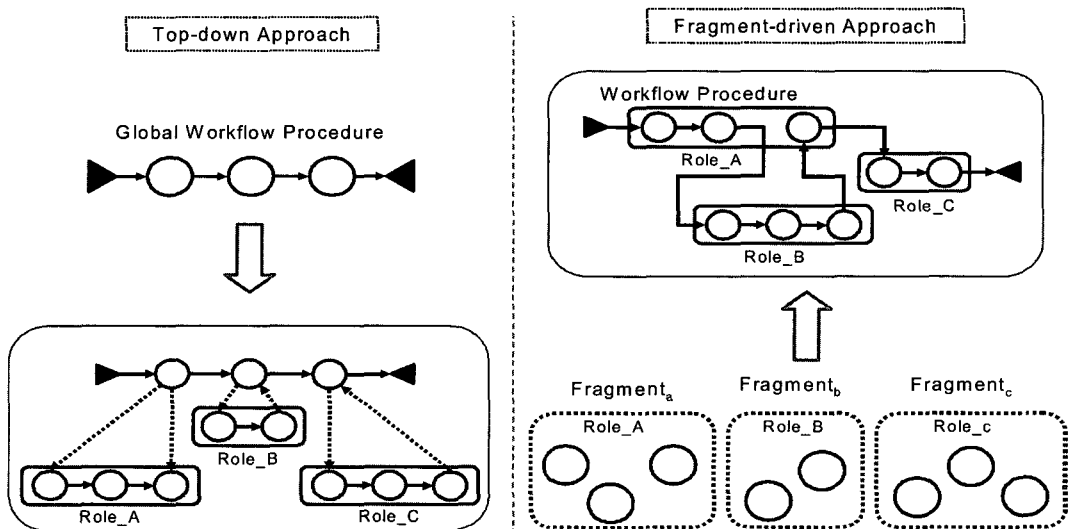
액티비티가 어느 룰에 속하는지, 무엇을 표현 하는지를 알아야 하는데 스윙레인 모델에서는 앞에서 논했던 것처럼 사용자의 이해나 숙련에 관계 없이 정보의 흐름을 한눈에 보여 주기 때문이다.

이와 같은 글로벌 프로세스의 정의를 각 룰에 기반한 Bottom-Up방식으로 모델링 하기 위하여 본 논문에서는 Fragment-Driven 글로벌 워크플로우 모델링 기법을 제안 한다.

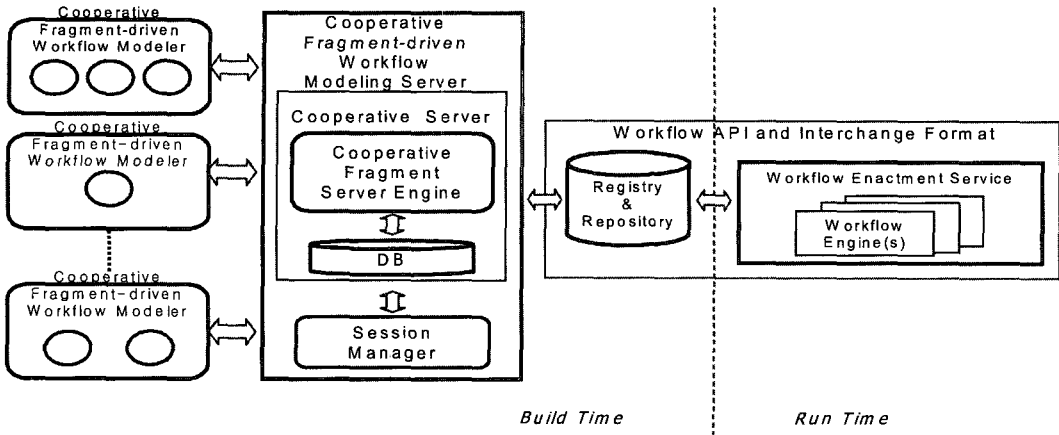
3.4 모델링 구성요소

협동 스윙레인 워크플로우 모델에서 각 룰이 액티비티로부터 프로세스의 작은 부분들을 정의해 나갈 때 레지스트리 시스템은 모델러에서 정의된 비즈니스 프로세스를 좀 더 효율적으로 관리하고 분산 처리 형태를 유지하며 각 조직의 독립성을 보장해 준다. 아울러 런타임 환경에서 엔진과의 연동을 유연하게 한다[1,2]. 이러한 환경에서 협동 스윙레인 워크플로우 모델에는 다음과 같은 정책을 정의한다.

- ▶ 각 룰은 한 개의 레인을 가진다.



〈그림 5〉 Top-Down방식과 Fragment-Driven 방식의 비교



▷ ○ : Fragment-Driven Global Workflow Process Definition Tool

〈그림 6〉 Fragment-Driven 워크플로우 모델링 시스템 구조

- ▶ 하나의 레인에는 프로세스에서 각각의 롤이 수행하게 될 액티비티를 정의할 수 있다.
- ▶ 각 레인에는 해당 롤이 액티비티를 정의하는 것을 기본으로 하며 필요한 경우, 프로세스 전체 관리자가 해당 레인이 액티비티를 정의할 수 있도록 지원한다.
- ▶ 하나의 레인에서 각 액티비티 자체는 기본적으로 모든 액티비티에 보여지지만(public) 액티비티의 속성들은 해당 롤만이 볼 수 있다(private).
- ▶ 각 액티비티의 흐름은 서로 관련되어 있는 롤의 동의 하에 연결할 수 있다. 필요한 경우에는 전체 비즈니스 프로세스의 관리자가 정의할 수 있다.

정의하고, 이렇게 정의된 액티비티를 관리자격을 갖춘 디자이너가 전체 프로세스로 연결하게 되는 방법을 사용 하는데 이것을 Fragment-driven 글로벌 워크플로우 모델링 방법이라 한다.

서로 다른 조직이 모델링 작업에 참여할 때 Fragment-Driven 글로벌 워크플로우 모델의 전체적인 시나리오를 그림 7과 같이 표현할 수 있다.

그림에서 아래의 점선으로 표현된 박스 부분은 워크플로우 엔진 부분으로 본 논문에서 제시하는 연구 범위에 속하지는 않지만, 본 시스템을 통해서 구축된 글로벌 워크플로우 모델은 각 연관 기업의 워크플로우 엔진으로 나뉘어져 상호 운용의 기본 데이터로 활용되게 된다.

협동 스윙레인 워크플로우 모델링 시스템 구조는 다음과 같은 4가지 요소로 나누어진다.

4. Fragment-Driven 워크플로우 모델링 방법론

4.1 Fragment-Driven 글로벌 워크플로우 구조

역할 기반의 워크플로우 모델인 협동 스윙레인 워크플로우 모델은 실제 조직에서 액티비티를 수행하게 될 각 참여자가 자신의 액티비티를 먼저

- ▶ 협동 워크플로우 모델링 시스템(Cooperative Workflow Analysis, Modeling, and Definition System)

☞ Private View - 로컬 사용자에게 의한 워크플로우 모델링 작업 환경

☞ Public View - 글로벌 워크플로우 모델링 작업 환경

- ▶ 협동 워크플로우 모델링 이벤트 순서화 서버(Cooperative Workflow Modeling Event Serialization Server)
- ▶ EJB 기반의 데이터베이스 컴포넌트
- ▶ EJB 기반의 통신 인프라 (Middleware)

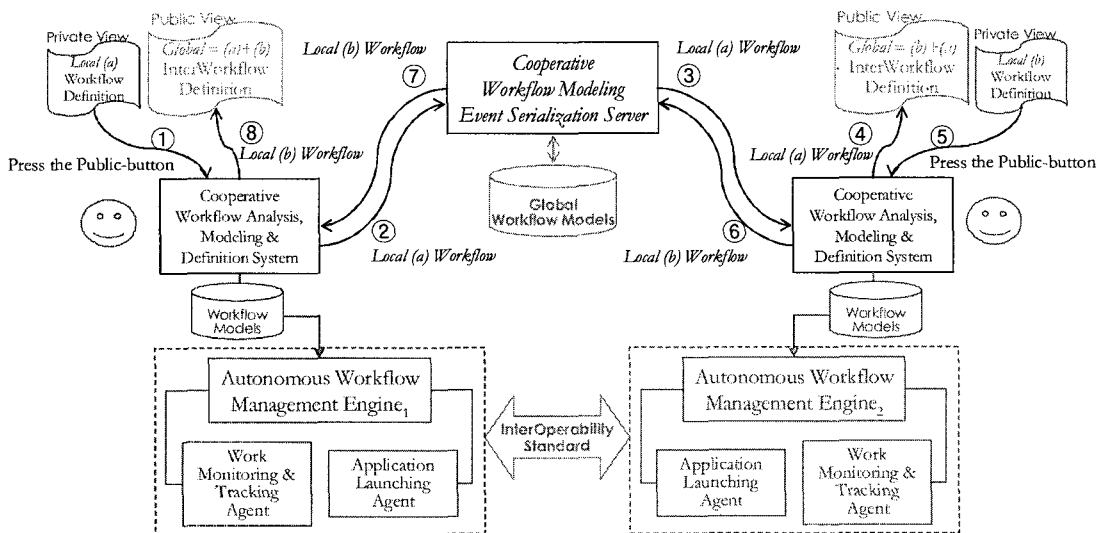
그림 7에서 Fragment-Driven 글로벌 워크플로우 모델링 방식에 의한 글로벌 워크플로우 정의과정을 간단히 개념적 시나리오를 통해 나타내었다.

결국 글로벌 워크플로우 모델링 작업환경(Public View)에서는 자신이 정의한 프로세스와 다른 조직에서 정의한 프로세스들이 공존하게 되어 런타임시 워크플로우 엔진들과의 연동이 가능하도록 하는 것이다. 이때 서로 다른 조직에서 프로세스를 독립적으로 정의 하지만 워크플로우 협업엔진들 간에 연동을 유연하게 하기 위하여 상호교환이 가능한 XPDL로 저장한다. 하지만 이러한 이론을 실제로 구현 하려면 어려움이 크기 때문에 다음과 같은 방법으로 설계와 구현이 가능하다.

- ▶ 글로벌한 환경에서 공통으로 사용할 수 있는 글로벌 워크플로우 모델링 작업환경을

하나의 큰 조직으로 만든 다음, 그 내부에 각각의 조직에서 전용으로 사용할 수 있는 로컬 사용자에 의한 워크플로우 모델링 작업환경(Private View)을 둔다.

- ▶ 외부에서 프로세스 정의에 참여하는 각 조직들이 웹 환경이나 네트워크 환경으로 각자 로컬 사용자에 의한 워크플로우 모델링 작업환경에서 작업을 한다.
- ▶ 각각의 로컬 사용자에 의한 워크플로우 모델링 작업환경에서 정의가 완료된 프로세스들은 $X_{veo(A)}$ 역할 구분이 $X_{veo(A)}$ 되어 있는 바깥쪽 글로벌 워크플로우 모델링 작업환경으로 이동되어 보관된다.
- ▶ 글로벌 $X_{veo(p)}$ 워크플로우 모델링 작업환경은 프로세스 정의에 참여하는 각 조직들의 역할을 미리 구분해 놓는다. 그러므로 각 조직에서 정의한 프로세스들을 역할에 따라 정렬할 수 있다.
- ▶ 런타임시 다른 워크플로우 엔진들간에 연동을 쉽게 하기 위하여 XPDL로 저장한다.
- ▶ 필요한 워크플로우 엔진에서 연결하여 사용할 수 있다.



〈그림 7〉 Fragment-Driven 방식의 조작 시나리오

4.2 Fragment-Driven 워크플로우 모델 구성요소

앞에서 주어진 Fragment-Driven 워크플로우 모델의 구성요소들을 PM-Graph으로 정리해 보면 다음과 같다.

튜플 $G=(V, \iota, o, N, \Psi, C, \Omega, \varepsilon, E, \Phi, \Delta,)$ 를 프로세스 모델 그래프라고 한다.

- (1) V 는 데이터 요소의 유한 집합이다. 각 데이터 요소 $v \in V$ 는 $DOM(v)$ 에 할당된 유한집합이다.
- (2) $\iota : N \cup \{G\} \rightarrow \wp(V)$ 는 입력 컨테이너 맵이라 한다.
- (3) $o : N \cup \{G\} \rightarrow \wp(V)$ 는 출력 컨테이너 맵이라 한다.
- (4) N 은 액티비티의 유한집합이다.
- (5) $\Psi: N \rightarrow \varepsilon$ 은 액티비티 구현 맵이라 하며 $\Psi(A): DOM(v) \rightarrow DOM(v)$
- (6) C 는 조건들의 유한 집합이고 각 조건 p 는 맵이다.

$$P \cdot DOM(v) \rightarrow \{0, 1\}$$

- (7) $\Omega: N \rightarrow Q$ 스테프 할당 맵이다.
 $\Omega(A): N \rightarrow \wp(A): A$ 는 에이전트들의 유한집합이다.
- (8) $\varepsilon: N \rightarrow C$ 각 액티비티에 할당된 종료 조건이다.
- (9) $E \subseteq N \times N \times C$ 는 다음 조건을 갖는 제어 연결자들의 집합이다.

(가) E 는 unified인데 그것의 의미는

$$\forall e, e' \in E: \pi_{1,2}(e) = \pi_{1,2}(e') \Rightarrow \pi_3(e) = \pi_3(e')$$

(나) (N,E) is acyclic.

- (10) $\Phi : N \rightarrow \bigcup_{A \in N} \phi_A$ 는 조인 조건을 할당한다.
 $\Phi(A) ? \phi_A := \{ \bigvee_{1 \leq j \leq k} \bigwedge_{1 \leq i \leq l} p' \mid p' ? \{p, p \mid p ? C^{\leftarrow} (A)\} \}$
 (가) $\forall A \in N : \Phi(A) \in \phi_A,$
 (나) $\forall A \in N. : \Phi(A) \bigcup_{A \in N, B \in N \cup C} \equiv 1$
- (11) $\Delta : N \times (N \cup C) \rightarrow \wp$

$(o(A) \times \iota(B))$ 는 아래와 같이 데이터 연결 맵이다.

$$(가) \Delta(A,B) \in \wp(o(A) \times \iota(B))$$

(나) $\Delta(A,B) \neq 0 \Rightarrow B$ 는 A 로부터 도달할 수 있다.

$$(다) \forall B \in N: (x,z),(y,z) \in \bigcup_{A \in N} \Delta(A, B) \Rightarrow x=y$$

$$(12) \overrightarrow{\Delta}: N \rightarrow \bigcup_{A \in N} (\wp(\iota(P) \times \iota(A)) \cup \wp(o(A) \times o(P)))$$

는 아래같이 프로세스 데이터 연결 맵이다.

$$(가) \forall A \in N: \overrightarrow{\Delta}(A) \in \wp(\iota(P) \times \iota(A)) \cup \wp(o(A) \times o(P))$$

$$(나) \forall B \in N: (x,z),(y,z) \in \wp(\iota(P) \times \iota(B)) \cup \bigcup_{A \in N} \Delta(A, B) \Rightarrow x=y$$

$$(다) (x,z),(y,z) \in \bigcup_{B \in N} \wp(o(B) \times o(P)) \Rightarrow x=y$$

프로세스 모델 그래프 $G=(V, \iota, o, N, \Psi, C, \Omega, E, \Phi, \Delta, \overrightarrow{\Delta})$ 는 다음 두 개의 그래프를 포함한다.

▶ 제어흐름 그래프 $G_{control}=(N, E, V, \iota, o, C, \varepsilon, \Phi)$

▶ 데이터 흐름 그래프 $G_{data}=(N,E,V,\Delta,\overrightarrow{\Delta})$, 여기서 Edge의 집합 E 는 데이터 연결자들을 표현한 것이다.

$$E \subseteq N \times N \times \wp(V \times V) \text{ with } (A,B,\Delta(A,B)) \in E :\Leftrightarrow \Delta(A,B) \neq 0$$

4.3 Fragment Driven ICN 표현방법

기본 ICN(Information Control Net)은 4개의 구성요소인 프로시저, 액티비티, 선후관계와 자료 저장소로 구성되며 A 를 일련의 액티비티들의 집합이라 하고, R 을 자료 저장소의 집합이라 할 때 수식적인 표현과 정의는 다음과 같다.

▶ I 는 초기에 입력되는 자료 저장소들의 유한 집합이며, ICN의 실행 전에 외부의 프로

세스에 의해서 로드(Load)된다고 가정한다.

- ▶ O 는 마지막으로 출력되는 자료 저장소들의 유한 집합이며, ICN의 실행 후 외부의 프로세스에 의해 이용되는 정보들을 포함한다고 가정한다.

▶ $\delta = \delta_i \cup \delta_o$.

여기서 $\delta_i : A \rightarrow \wp(A)$ 는 하나의 액티비티를 실행하는 액티비티들의 집합에 연결하는 관계를 나타내며 $\delta_o : A \rightarrow \wp(A)$ 는 하나의 액티비티를 후행하는 액티비티들의 집합에 연결하는 관계를 나타낸다.

▶ $\gamma = \gamma_i \cup \gamma_o$.

여기서 $\gamma_i : A \rightarrow \wp(R)$ 은 하나의 액티비티를 실행하는 액티비티 집합들을 출력자료 저장소들의 집합과 연결하는 것 중 하나이며, $\gamma_o : A \rightarrow \wp(R)$ 은 하나의 액티비티를 후행하는 액티비티 집합들을 입력자료 저장소들의 집합과 연결하는 관계를 나타내는 것 중 하나이다[3].

(표 1) ICN모델의 정형적 정의

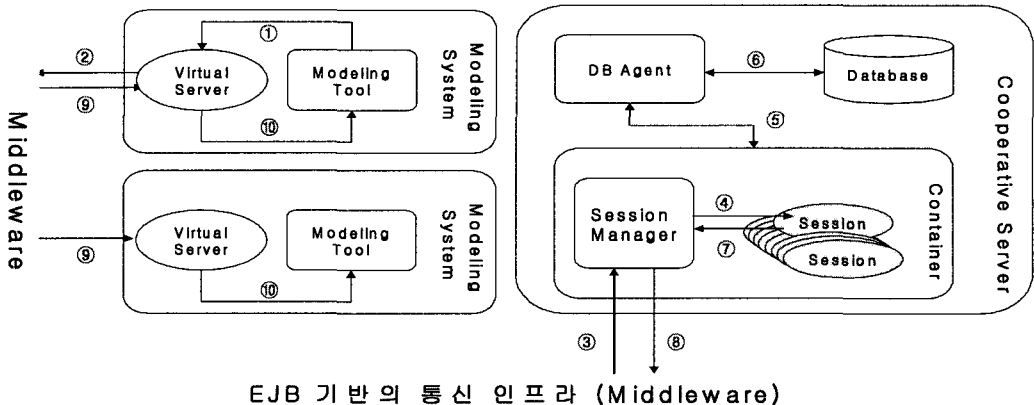
$\Gamma = (\delta, \gamma, I, O)$ δ : precedence constraint among activities γ : repository input/output requirement of activity I : initial input repository O : final output repository

5. 협동 스웜레인 시스템

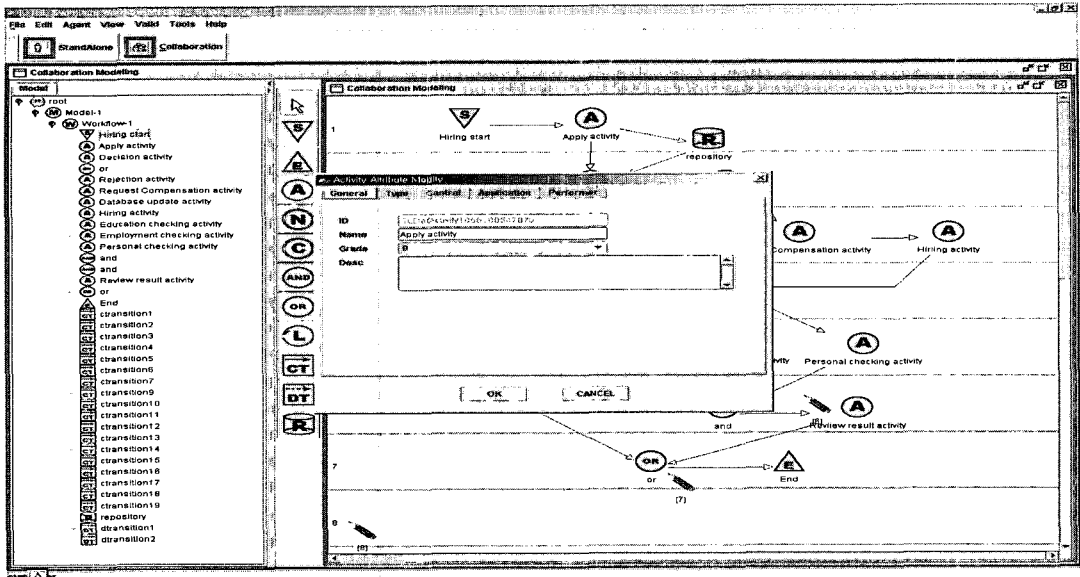
5.1 시스템 컴포넌트 구성

그림 8은 **Fragment-Driven** 글로벌 워크플로우 모델링 시스템 전체의 아키텍처이다. 클라이언트 모델링 시스템은 사용자와 인터랙션을 할 수 있는 사용자 인터페이스를 가지고서 사용자의 액션에 따라 이벤트를 다른 사용자에게 전달하게 되는데 이런 이벤트를 전달 하기 위하여 실제 협업 서버와 직접적으로 통신하는 것이 아니라 각 조직의 물리적인 변동사항에 독립적으로 프로세스를 정의할 수 있는 가상 협업 서버를 거친다. 그러면 가상 협업 서버와 모델러로 구성된 **Fragment-Driven** 글로벌 워크플로우 모델링 클라이언트는 **EJB** 기반의 통신 인프라(Middleware)를 통해 **Fragment-Driven** 글로벌 워크플로우 모델링 협업 서버와 이벤트를 주고 받으며 통신을 한다. 실제 협업 서버는 각 사용자로부터 수신된 이벤트들을 처리한 후 이벤트를 송신한 사용자가 속해 있는 그룹에게 브로드 캐스팅 함으로서 협업을 가능케 한다. 또한 작업이 완료된 모델들은 레지스트리 시스템을 통하여 외부로 publish된다.

그림 8에서 레지스트리 시스템은 완전히 정의된 워크플로우 모델이 등록되는 곳으로서 워크플



(그림 8) 협동 스웜레인 모델링 시스템 아키텍처



(그림 9) 스윙레인 모델링 클라이언트의 모델링 화면

로우 관리 시스템이 워크플로우 모델을 참조한다.

5.2 구 현

그림 9는 각 레인에 해당하는 사용자 7명이 동시에 Fragment-Driven 글로벌 워크플로우 모델링 클라이언트를 통해서 프로세스를 정의하고 있는데, 그 중 한 사용자가 액티비티 정보를 설정하는 스윙레인 모델링 클라이언트의 모습이다.

그림에서 좌측은 워크플로우 모델 정의를 계층적인 트리형태로 보여주고 있는데, 트리 패널이라 하며 각각 정의된 액티비티 정보를 나열한다. 그림의 오른쪽 화면은 드로우 패널이라 하며 Fragment-Driven 글로벌 워크플로우 모델링 작업 화면을 보여주고 있다. 가운데 세로로 놓여진 툴바는 액티비티를 비롯한 디자인에 사용되는 요소들로서 이것들 중에 하나를 선택하여 우측에 보이는 드로우 패널의 레인 중에서 자신의 레인에 해당하는 영역에 디자인할 수 있다.

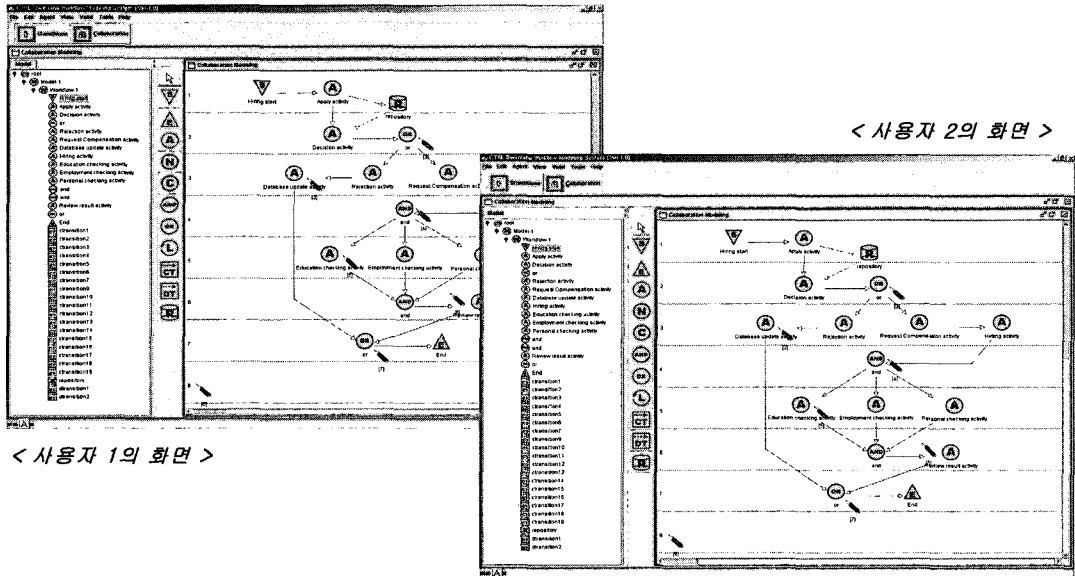
그림 10은 7명의 사용자 중에서 2개의 스윙레인 모델링 클라이언트 화면을 보여주고 있다. 그

림에 나타난 바와 같이 모든 스윙레인 모델링 클라이언트의 화면이 똑 같다. 따라서 사용자는 작업에 참여중인 그룹 멤버들이 실제 멀리 떨어져 있어도 근처에서 작업하는 것처럼 느끼게 해준다. 그리고 정지된 화면이어서 잘 나타나지는 않지만 특히 그림에서 나타난 연필은 각 유저의 마우스와 같은 것으로서 사용자의 마우스 움직임과 동기화 되어 있다. 이것은 작업에 참여중인 그룹 멤버들에게 Awareness를 제공한다.

6. 결 론

Bottom-Up 방식의 Fragment-Driven 글로벌 워크플로우 모델링 방법에 의한 협업 스윙레인 워크플로우 시스템은 서로 다른 워크플로우 시스템 환경에서 조직 정의시 효과적으로 활용될 수 있다.

기존의 워크플로우 모델링 도구를 활용하여 글로벌 워크플로우를 정의해야 한다면 각 기업의 정보(조직정보 등)에 대한 보호를 보장할 수 없으며, 각각의 워크플로우 모델링 도구를 사용할 경우는 각 기업의 로컬 워크플로우들을 통합하여 하나의



(그림 10) 협동 스웸레인 워크플로우 모델링 클라이언트간 협업화면

글로벌 워크플로우를 생성하는데 어려움이 있었다. 본 논문에서 제안하는 Fragment-Driven 워크플로우 모델링 방법은 기존의 모든 모델링 방법에서 가지고 있는 문제점들을 해결하면서도 멤버조직들 간의 독립성을 보장할 수 있다. 그리고 협동을 통한 효율적인 글로벌 워크플로우 모델링 방법을 제공할 수 있다는 점에서 매우 우수하며 워크플로우 기반의 전자상거래를 구축 하는데 있어서 중요한 역할을 할 것으로 기대한다.

앞으로 이러한 톨을 기반으로 하는 시스템에서 접속 제어에 대한 RBAC(Role Based Access Control)과 관련된 연구가 가능하며, 또한 지금 급속하게 진행 되고 있는 P2P환경과 물류시스템 환경으로도 연구를 확장해 볼 수 있다.

참고 문헌

[1] 김광훈, "워크플로우 기술 III", TTA 저널 88호, pp.105-120, 2003
 [2] 한승혁, 오동근, 이정훈, 원재강, 김형목, 김광훈, "협동 스웸레인 워크플로우 모델", 한

국정보과학회 춘계학술발표논문집A, 30권1호, 2003. 4

[3] 김형목, "Fragment-Driven Global Workflow modeling methodology and system", 경기대학교 박사학위논문, pp.21-68, 2003
 [4] Clarence A. Ellis, Gary J. Nutt, "ICNs Revisited and Revised", Technical Report
 [5] Clarence A. Ellis, Gary J. Nutt, "Modeling and Enactment of Workflow Systems", in proceedings of the 1993, jun
 [6] Dimitrios Georgakopoulos and Mark Horricks, "An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure", Kluwer Academic Publishers, Boston. Manufactured in The Netherlands, Distributed and Parallel Database, 3, 119-153, 1995
 [7] Gary J. Nutt, "Using Workflow in Contemporary Applications", Technical Report, CU-CS-663-93, Aug 1993
 [8] Haruo Hayami, Masashi Katsumata, Ken-

ichi Okada, "Interworkflow : A Challenge
for Business-to-Business Electronic Com

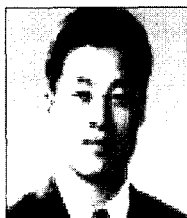
merce", Workflow Handbook 2001, 2002
[9] <http://www.wfmc.org/standard>

◎ 저 자 소개 ◎



문 기 동 (Ki-Dong Moon)

1992년 국립서울산업대학교 전자계산학과 졸업(학사)
1995년 동국대학교 대학원 컴퓨터시스템학과 졸업(석사)
2005년 경기대학교 대학원 전자계산학과 졸업(박사)
관심분야 : 워크플로우, 네트워크보안
E-Mail : kdmooon@kyonggi.ac.kr



김 형 목 (Hyung-Mok Kim)

1984년 원광대학교 토목공학과 졸업(학사)
1986년 건국대학교 대학원 토목공학과 졸업(석사)
1996년 경기대학교 전자계산학과 졸업(학사)
1998년 아주대학교 대학원 컴퓨터교육과 졸업(석사)
2004년 경기대학교 대학원 전자계산학과 졸업(박사)
관심분야 : U-네트워크, 워크플로우, 무선인터넷
E-Mail : webmaster@bazi.pe.kr



김 광 훈

1984년 경기대학교 전자계산학과 졸업(학사)
1986년 중앙대학교 대학원 전자계산학과 졸업(석사)
1994년 미국 콜로라도대학교 전자계산학과 졸업(석사)
1998년 미국 콜로라도대학교 전자계산학과 졸업(박사)
1998년~현재 경기대학교 정보과학부 교수
관심분야 : 협동시스템, 그룹웨어, 워크플로우
E-Mail : kwang@kyonggi.ac.kr



백 수 기

1972년 연세대학교 토목공학과 졸업(학사)
1979년 동국대학교 대학원 전산통계학과 졸업(석사)
1992년 동국대학교 대학원 전산통계학과 졸업(박사)
1980년~현재 경기대학교 정보과학부 교수
관심분야 : 컴퓨터 네트워크, 워크플로우, 네트워크 알고리즘
E-Mail : skpaik@kyonggi.ac.kr