대한산업공학회/한국경영과학회 2002 춘계공동학술대회
한국과학기술원(KAIST) 2002년 5월 3일-4일

EJB 기반의 Workflow Engine으로의 변환

김 원섭*, 김 선호**
㈜ 포스데이타*, 명지대학교**

Abstract

최근 Workflow System의 컴퍼넌트화는 필수적인 것으로 인식되고 있다. 컴퍼넌트화의 장점으로는 유지, 보수의 용이성과 다른 어플리케이션과의 원활한 통합을 들 수 있고, 컴퍼넌트 기반의 Workflow System은 e-business infra를 형성하는데 필수적이다. 따라서 본 연구에서는 기업 내부의 프로세스를 관리해 주는 Workflow System을 컴퍼넌트 형태로 설계하여 가장 진보된 형태의 Workflow System의 기반을 마련한다. 본 논문에서는 RUP방법론에 따라 use case분석, class diagram분석을 통해서 DISFlow(Daewoo Information Workflow)를 EJB Architecture에 맞게 변환하였다.

1. 서론

WfMS(Workflow Management System)는 기업 간, 기업 내의 업무 프로세스를 정의(Definition), 실행(Execution), 평가(Assessment), 개선(Improvement)하기 위한 시스템으로 프로세스와 관련된 정보 및 자원의 흐름을 통합 관리한다.

WfMS는 몇 단계의 발전과정을 거쳐서 발전하는데, 초기에 부분적인 업무 프로세스를 자동화하는데 사용되며 주로 프로세스가 정형화된 은행이나 보험분야에 적용되어 문서 중심의 승인 프로세스에 사용되었다. 1990년대 중반 그룹웨어에 추가되는 Component 형태로 발전하였으며, 주로 정형화된 업무 프로세스에 적용되었다. 다음 단계로 비즈니스 프로세스 관리를 대규모 기업 어플리케이션 개발에서 분리하여 기존의 어플리케이션과 원활하게 통합되는 형태로 발전하였다. 이로 인해 보다 유연하게 기업의 어플리케이션을 개발하고 관리할 수 있게 되었다.

어플리케이션 Component가 점점 더 모듈화 됨에 따라 J2EE(Java(TM) 2 Platform, Enterprise Edition) 같은 표준 Framework을 사용하여 어플리케이션 Component를 개발하는 방식이 등장하였다. 이 방법은 표준 Framework을 준수하면서 그에 맞는 Component 들을 개발하므로 비즈니스 method 구현에 집중할 수 있고, 하부 Infrastructure 관한 개발자의 부담을 줄일 수 있다.

위에서 제시한 J2EE 기반의 컴퍼넌트 개발을 위해서는 컴퍼넌트 기반의 소프트웨어 개발 방법론이 필요하다. 이 방법론은 소프트웨어 위기와 객체지향 개발방법론의 한계성의 인식에서 시작되었다. 컴퍼넌트 기반 소프트웨어 개발 방법론은 소프트웨어 모듈의 재사용성과 독립성을 보장하여 소프트웨어의 복잡성과 생산성 문제를 해결하는 새로운 패러다임이다. [6 , 7]

최근 Workflow 시스템의 컴퍼넌트화는 필수적인 사항으로 인식되고 있다. 컴퍼넌트화의 장점으로는 유지, 보수의 용이성과 다른 어플리케이션과 원활하게 통합 등이 있다. 또한 전자상거래 환경에 적합한 e-business Infrastructure를 형성하는데 중요한 역할을 한다. 따라서, 본 연구에서는 기업 내부 프로세스를 관리해주는 Workflow 시스템을 Component 형태로 개발하여 가장 진보된 형태의 Workflow 시스템을 구현하고자 한다. [3, 8]

본 논문에서는 객체지향 방법론을 사용하여 개발 중인 Workflow 관리 시스템을 컴퍼넌트화 하고 분산 객체를 지원하는 EJB Framework을 적용하여 위에서 제시하고 있는 획일적인 Architecture 기반의 Workflow 시스템이 야기하는 문제들 키고자 한다. 또한 Workflow 시스템은 독립적인 기능을 수행하기 보다는 다른 어플리케이션에 Embedded 되어 역할을 수행하는 경우가 많다.

즉, ERP 시스템이나 PDM 시스템의 Procsss 관리기능을 수행할 경우 컴퍼넌트화하여 다른 시스템들과의 원활한 통합을 이룰 수 있는 Architecture를 지원하고자 한다. [3, 7]

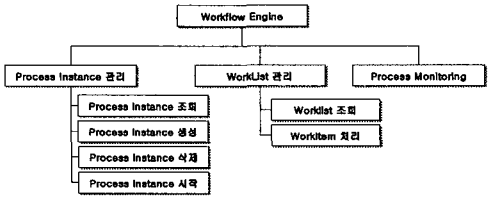
워크플로우 시스템 개발에서 전통적인 아키텍쳐는 객체지향 어플리케이션 프로세스를 구현하는데 한계를 가지고 있다. 이러한 한계점에 관한 연구로는 몇 가지 사례는 다음과 같다. 우선, Saarland 대학의 Mentor 워크플로우 시스템은 현재 단순하고 무거운 워크플로우 시스템 아키텍쳐의 한계를 인식하고 lightweight core를 중심으로 생성된 새로운 워크플로우 시스템이다. 두 번째로 OPERA 프로젝트에서는 단순한 아키텍쳐로는 새로운 종류의 어플리케이션을 적용하는데 부족하므로 CORBA를 기반으로 하는 각각의 컴퍼넌트에 의해 Inter-process communication을 지원하는 프로세스 관리 커널을 제안하였다. 하고 있다. 위의 문제점들을 컴퍼넌트화로 일반화 시키고 공통 기능들을 구현한 컴퍼넌트들을 공유함으로써 이러한 문제점을 해결하고자 했다. [1, 2, 4, 5]

2. 본론

2.1 재설계 대상 워크플로우 엔진 (Daewoo Information System Worklfow)

재설계 대상 DISFlow는 명지대학교와 대우정보시스템이 공동 개발한 워크플로우 관리시스템으로 중앙집중식 구조를 가지고 있다. 워크플로우 엔진은 정의된 프로세스를 통해 프로세스 인스턴스를 생성하고 프로세스의 흐름에 따라 프로세스를 관리하는 기능을 수행한다. 또한, 프로세스의 흐름에 따라 작업자에게 작업목록을 전달하여 작업을 수행하고 업무 수행에 대한 결과를 모니터링하여 평가할 수 있으며 프로세스의 흐름을 제어할 수 있는 기능을 수행한다.

프로세스 디자이너에서 정의된 데이터들(프로세스 정의, 비즈니스 룰 정의, relevant data type 정의, 어플리케이션 정의, participant 정보)을 Workflow Engine으로 전송하게 되면 그 데이터를 기준 데이터로 사용하여 Workflow Engine에 대한 기능을 사용하게 된다



프로세스 인스턴스 관리는 프로세스 디자이너에서 정의된 프로세스를 이용하여 작업자가 워크플로우를 사용하기 위한 프로세스 인스턴스를 생성하여 시작할 수 있는 기능을 수행한다. 워크플로우 엔진은 시작된 프로세스의 흐름에 따라서 작업자에게 작업을 할당하고 작업자가 처리한 작업과 비즈니스 룰에 따라서 프로세스의 흐름을 제어한다. 워크리스트 관리는 워크프로우 엔진에서 작업자에게 할당해준 작업들을 관리한다. 프로세스 모니터링 관리는 워크플로우에서 진행되고 있는 프로세스 인스턴스에 대한 상태를 모니터링하고 관리한다.

2.2 컴퍼넌트 기반 구조로 변환

Enterprise Beans로의 변환은 3가지 타입의 빈으로 변환 되며, 각각의 빈들은 HomeInterface와 Remote Interface를 통해서 비즈니스 메소드가 구현된 Remote Interface를 Invocation할 수 있다. Enterprise Bean의 타입은 3가지 타입이 있다. Stateless Session Bean, Stateful Session Bean과 EJB Specification 2.0에서 추가된 Message-Driven Bean이 있다. WorkflowEngineEJB 클래스는 Enterprise Bean으로 실제 비즈니스 메소드가 구현되어 있는 클래스로 외부와 RMI/IIOP 프로토콜을 사용해서 통신하는 부분은 WorkflowEngineEJBHome과 WorkflowEngineEJB 이다. Home 인터페이스는 Enterprise Bean을 생성, 소멸하는 역할을 수행하고, WorkflowEngine이라는 Remote 인터페이스는 외부에서 WorkflowEngineEJB 클래스에 접근할 수 있도록 인터페이스 역할을 수행한다. Message-Driven Bean은 JMS통해 어떤 메시지가 도착했을 경우 onMessage()를 수행하도록 설계된 Bean이다

- 변환 내용은 다음과 같다

① JSP에 Presentation 부분과 Function layer부분을 분리한다.

② WorflowEngine은 Stateless SessionBean으로 변환한다.

③ ProcessDefinition, ProcessInstance, Activity, ActivityInstance, Participant, WorkItem, Transition, TransitionCondtion은 Entity Bean으로 변환한다.

④ WorkListHandler는 Message-Driven Bean으로 변환한다.

본 논문에서 제시한 컴퍼넌트기 기반의 Workflow Engine은 J2EE 프레임웍 기반으로 비즈니스 로직을 구현하여 확장성과 소프트웨어 유지관리에 용이한 구조를 가진다. 기업의 프로세스를 효율적으로 관리하는 Workflow Engine은 기업의 어플리케이션들과 원활한 통합이 필수적인데 컴퍼넌트 기반의 미들웨어 프레임웍을 사용하여 이를 실현할 수 있으며, 이 프레임웍을 사용하여 비즈니스 로직만을 개발하는 Workflow 시스템을 light-weight Workflow라고 하며 Workflow의 유연성을 크게 높인다.

3. 결론

본 논문에서 제시한 컴퍼넌트기 기반의 Workflow Engine은 J2EE 프레임웍 기반으로 비즈니스 로직을 구현하여 확장성과 소프트웨어 유지관리에 용이한 구조를 가진다. 기업의 프로세스를 효율적으로 관리하는 Workflow Engine은 기업의 어플리케이션들과 원활한 통합이 필수적이다.

컴퍼넌트 기반의 미들웨어 프레임웍을 사용하여 이를 실현할 수 있으며, 이 프레임웍을 사용하여 비즈니스 로직만을 개발하는 Workflow 시스템을 light-weight Workflow라고 하며 Workflow의 유연성을 크게 높인다. 또한 다양한 legacy 어플리케이션을 수용하여 Process 중심으로 기업내부 어플리케이션을 통합하고, 인터넷 환경을 지원하는 개방형 구조를 가진다

추후 연구과제로는 현재까지는 기업내부 혹은 조직 내부의 프로세스를 관리하는데 Workflow 시스템이었지만 조직간, 기업간 프로세스에 대한 관심이 커지고 있다. 단계적으로 기업 내부 어플리케이션의 통합(EAI : Enterprise Application Integration) 에서 기업간 어플리케이션의 통합(B2Bi : Business-to-Business Integration)으로 통합의 범위를 확장해 가고 있다. 이러한 환경에 적합한 Workflow 시스템에 관한 연구가 필요하다.

참고 문헌

[1] Christoph Bussler, "Enterprise-Wide Workflow Management," IEEE Concurrency, Volume : 7, Issue : 3, pp 32-43, 1999

[2] Dragon Manolescu, "A Extensible Workflow Architecture with Object and Pattern," TOOLSEE 2001, http://micro-workflow.com/Writing.phtml,Sofia,Bulg

[3] Sun Microsystem, Enterprise JavaBeans Specification, version2.0 Final Release, Sun Microsystems, 2001

[4] Qiming Chen, Meichun Hsu, "Inter-enterprise collaborative business process management," Proceedings of 17th International Conference on Data Engineering, 2001

[5] Jeanine Weissenfels, Michael Gillman, Olivier Roth, German Shegalov, Wolfgang Wonner,"The Mentor-lite Prototype: A Light-Weight Workflow Management System," Proceedings of 16th International Conference on Data Engineering, pp685-686, 2000

[6] Sun Microsystem, Sun Open Network Enterprise Architecture, http://www.sun.com/software/sunone/,2001

[7] 김세근, 서창근, 김민식, EJB Bible: enterprise javabeans bible, 정보문화사, 2001

[8] 김준, 컴퍼넌트 기반의 e-business 개발방법론과 모델링, http://e-bizgroup.com/working/paper.htm,2001