

워크플로우 시스템에서의 비결정적 단위업무 처리

정우진[○] 안형진 박민재 김광훈 백수기
경기대학교 전자계산학과

{woojin[○],hjahn, mean222, kwang, skpaik }@kyonggi.ac.kr

Treatment For Non-deterministic Activities Of Workflow System

Woo-Jin Jung[○], Hyung-Jin Ahn, Min-Jae Park,
Kwang-Hoon Kim, Su-Ki Paik
Dept of Computer Science, Kyonggi University

요 약 내 용

기업이나 조직의 업무활동은 수많은 프로세스들로 이루어진다. 프로세스란 개인, 부서, 관련 외부 조직들 사이에 이루어지는 의사결정에 필요한 일련의 과정들을 의미하는 것으로 최종적인 의사결정에 이르기까지의 과정을 의미한다. 기업들이 비즈니스를 처리하는 모든 과정과 활동 속에서 수많은 업무 프로세스가 동시 다발적으로 생성되고 처리되고 있다. 정보통신 기술의 발달과 함께 등장한 전자업무 환경은 업무의 처리를 간편하게 만드는 동시에 더욱 복잡한 업무 형태를 생성해내었고 이에 따라 효율적인 프로세스 관리가 더욱 필요하게 되었다.[4]

프로세스 관리의 중요성은 정보통신 기술의 발전과 맞물려 워크플로우에 대한 개념을 만들었다. 기업 및 조직은 워크플로우 관리 시스템을 도입함으로써 비즈니스 프로세스(Business Process)로 정의된 일련의 업무 처리 과정을 자동화하여 보다 효과적으로 관리 감독할 수 있게 되었다. 워크플로우 시스템이 적용된 각 비즈니스 프로세스는 여러 개의 단위업무들로 구성되어 있고 각 단위업무는 독립적으로 수행 가능하거나 앞 단계의 단위업무 처리의 결과에 따라 수행가능하다. 또한 프로세스 인스턴스들간의 상호 운용도 가능하다.

이렇게 복잡한 처리 구조를 가진 워크플로우 시스템에서 각 프로세스의 수행에 대한 결정성은 중요하다. 프로세스 인스턴스가 수행되는 동안 어떤 순간에 참조되는 데이터의 값에 따라 처리 결과가 매번 달라지고, 또한 그 변화의 영향이 수행중인 인스턴스 내부적인 요인이 아닌 외부 간섭에 의한 작용이라면 이것은 기업이나 조직의 입장에서는 원활한 업무의 진행을 방해하는 요소가 될 수 있다.

현재 대부분의 워크플로우 시스템에서 업무에 대한 트랜잭션의 처리는 EJB같은 컨테이너를 통하거나 DBMS에서 제공되는 데이터단위에서의 처리를 이용한다. 이것은 단지 업무를 처리할 때 활용되는 애플리케이션이 사용하는 데이터들간의 병행성을 제어하는 정도이다. 이러한 구조의 워크플로우 시스템은 프로세스 전체를 바라보는 입장에서는 각 프로세스들간 혹은 단위업무간의 트랜잭션 파악이 쉽지 않게 되고, 서비스 구현의 입장에서는 데이터 접근성에 대한 결정적 요소들을 포함한 효율성을 고려해야 하기 때문에 설계에서부터 많은 노력이 필요하다.

비즈니스 프로세스를 처리하는 워크플로우 모델에서는 구현단위에서의 데이터 트랜잭션 보장이 아닌 서비스단위에서의 트랜잭션 보장이 필요하다. 데이터 자체에 대한 비결정적 문제는 데이터필드에 대한 기록과 참조에 집중하지만 서비스 트랜잭션은 업무 처리 자체에 관점을 둔다. 공통의 데이터 레파지토리에 접근하는 단위업무들 간 혹은 프로세스 모델들간에는 처리되는 순서에 따른 결과의 비결정적 문제가 발생한다. 공용적인 레파지토리에 접근하는 업무들 중 어떤 업무가 먼저 처리되는가에 따라 다른 업무들이 영향을 받고 결국 전체 프로세스의 흐름이 변화될 수 있기 때문에 워크플로우 모델에서의 서비스 트랜잭션에 대한 결정성 문제는 중요하다. 이것을 해결하기 위해서는 워크플로우 엔진의 논리적 처

리를 이용한다. 각 단위업무들간의 관계와 레파지토리의 관계, 혹은 프로세스 모델들간의 관계 정보를 알고 있는 워크플로우 엔진에게 현재 수행되는 트랜지션에 대한 정보 메시지를 전달함으로써 다른 업무들이나 프로세스들의 진행을 지연시키는 방법을 사용한다. 현재 처리되는 서비스의 트랜지션 정보는 다른 업무들을 수행시킬 수 있는지의 여부를 결정하는 메시지가 될 수 있다.

본 논문에서 제시한 방법에서 더 나아가 구체적인 트랜잭션 메시지에 대한 스키마 정의와 이것을 처리할 수 있는 워크플로우 엔진의 서비스 인터페이스가 정립된다면 더욱 기능적인 비즈니스 프로세스 모델이 가능할 것이다.

참고문헌

- [1] WFMC, Workflow Management Coalition Workflow Standard, 'Process Definition Interface-XML Process Definition Language', Oct-3-2005
- [2] Gary J. Nutt, "Centralized and Distributed Operation Systems", Prentice Hall,1992
- [3] A. J. Bernstein, "Program Analysis for Parallel Processing", IEEE Transactions on Electronic Computers, Oct-1996, 757-762
- [4] 김광훈, "워크플로우 기술 I", TTA 저널 제 85호, 표준기술 동향, p107