

분산환경하에서의 Web 기반 작업관리기능 연구

김태규*, 이강원*, 조광문**, 김태운*

*고려대학교 컴퓨터학과

** 천안대학교 정보통신학부

e-mail : delon@unitel.co.kr

A Study on Web-based Workforce function in distributed Environments

Tae-Kyu Kim*, Kang-Won Lee*, Kwang-Moon Cho**, Tai-Yun Kim*

*Dept. of Computer Science & Engineering, Korea University

**Div. of Information & Communication Cheonan University

요 약

작업관리기능은 서비스 처리의 적절한 흐름유지를 위하여 활용되고 있다. 작업관리는 기업의 업무에 참여하는 구성원들 사이에서 이루어지는 문서와 정보 또는 작업의 절차를 일종의 정의된 규칙에 의하여 자동화하는 것을 의미하며, 정의된 규칙을 수행하기 위하여 사전에 정의된 절차대로 타스크(task)를 생성, 관리하는 기능이다.

본 논문에서는 기업의 조직과 분산된 환경하에서의 작업관리기능을 CORBA 를 기반으로 활용이 편리하도록 Web 으로 설계하였다. 타스크 템플릿을 이용하여 정형화된 규칙(rule)에 기반된 작업관리를 유지하였으며 예외처리, 다중 작업처리, 이기종간의 상호운용성은 CORBA 기반을 통하여 해결하였다.

연구된 기능은 이기종간 분산된 환경하에서 작업흐름의 반영이 용이하며, 단순한 상태기반과 사건기반의 흐름에 의한 작업관리기능에 비하여 신속한 서비스 제공이 가능함을 볼 수 있다.

1. 서론

기업의 업무절차가 복잡화되어 가면서 작업관리는 다른 작업관리 기능과 상호 연결성(interoperability)뿐만 아니라 특정 업무 수행도 지원하여야 한다. 전통적인 작업관리는 작업간의 의존성(dependency)에 따라 단일화된 절차를 수행하였으나, 최근에는 작업의 다양성으로 작업들의 영역은 구분이 없어지고 동시작업(parallel processing)이 요구되었다. 그러나 동시작업은 작업간의 절차적인 조건이 필요하게 되었다[1][2][3]. 작업관리기능은 일반적으로 사건(event)과 상태(state) 기준에서 일반적으로 사건기반흐름(event-based flow)을 기준으로 한다. 사건기반흐름은 상태기반흐름(state-based flow)에 비하여 첫째, 작업들간의 관계가 명확하

며, 둘째, 동시작업에 대하여 명확하게 처리할 수 있다. 또한 작업간에 종속적인 관계를 유지할 수 있다. 셋째, 발생하는 가능한 경우의 수에 대하여 변경은 필수적으로 특징지을 수 있다. 반면 상태기반흐름은 변경이 용이하기 때문에 대부분 petri-net 등으로 나타낸다[4]. 본 논문에서는 두가지 흐름을 수용하여 에이전트 기반의 트랜잭션 모델을 이용하였다. 무결성/신뢰성이 요구되는 업무 흐름의 지원, 트랜잭션의 처리가 필요한 업무 프로세스의 지원가능, 이기종/분산/자동화 응용프로그램들간의 연계, 각각의 단위 업무(activity)들의 트랜잭션 조절, 오류 핸들링 복구(recovery)가 용이하기 때문이다. 그러나 일반적인 트랜잭션 모델은 다양한 통신환경을 적용하기에는 다소 미흡하다. 기업은 지역적으로 분산되어 있으며 절차적

이기보다는 동시작업이 빈번하게 발생한다. 에이전트는 모든 작업이 서비스 기반으로 처리될 때, 각자가 제시하는 서비스를 책임지고 처리하며, 각 에이전트는 정보제공자 및 정보 요청자의 역할을 동시에 수행한다. 이것은 에이전트간의 대화 및 협의의 조정을 통하여 가능하며, 고장이나 예외상황에 적절히 대응하여 신뢰성을 높일 수 있고, 내부의 작은 변화에 유연하게 동적으로 대처할 수 있어야 한다. 따라서 본 논문에서 제시한 모델은 시간적, 공간적 요소를 극복하기 위하여 객체지향적인 모델을 도입하여 작업의 실행과 다양한 환경하에서 지원이 가능하도록 CORBA를 적용한 다양한 환경하에서 지원이 가능한 다중 트랜잭션 작업관리 모델을 제시하였다. 2 장에서는 시스템의 요구사항, 3 장에서는 모델을 제시하였고, 4 장에서는 시험 결과를 제시하였다. 끝으로 5 장에서 결론을 도출하였다.

2. 시스템 요구사항

하나의 업무는 반복적인 작은 단위(task)로 나누어진다. 수작업을 통해서 또는 전산화를 통하여 다양한 작업들이 관리된다. 작업관리기능은 다양한 작업들을 적절한 자원을 이용하여 처리될 수 있어야 한다. 그러나 많은 초기의 연구에는 문제점이 있었다[5]. 본 논문에서는 다음과 같은 사항을 고려하여 기능을 설계하였다. 첫째, 작업단위는 여러 조직과 사람, 시스템의 연결로 처리된다. 조직은 항상 변화가 되므로 기능은 조직의 변화에 유연성을 제공하여야 한다. 둘째, 작업흐름이 중간에서 많은 인터페이스를 가지므로 타 부분과의 인터페이스가 중요하다. 셋째, 작업 흐름기능은 여러 가지 에이전트(agent)를 가진다. 각각의 에이전트는 서로 연결성과 독립성을 보장하며 구성된다. 넷째, 작업흐름은 서로 다른 작업단위와 연결성 및 동시에 진행되어야 한다.

3. 시스템 설계

3.1 조직과 Task 간 모델링

조직은 시간적, 지리적으로 서로 떨어져 있는 상태에서 운용된다. 또한 조직의 잦은 변경에 의한 작업 처리의 할당은 잘못된 작업할당 및 작업손실의 위험을 가져올 수 있다. 따라서 기업조직은 작업관리기능에서 유연성을 제공할 수 있도록 프로세스와 독립된 형태로 유지되어야 한다. 프로세스 모델은 조직모델 상의 에이전트에 의하여 연결함으로써 조직과 작업단위간의 연결을 할당된 권한에 의하여 처리되도록 설계하였다.

3.2 사건기반 모델링

신규로 발생된 작업은 상태기반(event-based)의 모델작업후 이벤트를 발생하여 작업처리를 수행한다. 작업은 의존적 관계외에 동시에 작업이 수행되어야 한다.

3.3 룰기반 모델

Event-Condition-Action 규칙에 의거한 룰(rule)은 한 개인 또는 시스템이 일정시간에 수행하는 일의 단

위다. 일반적인 업무의 처리에 정형화된 룰이 존재한다면 기능의 정확성 및 처리가 향상될 수 있을 것이다. 템플릿을 이용하여 정형화 후 저장되며 해당 템플릿에 해당되지 않는 경우는 예외로 처리한다. 작업흐름을 통하여 분배된 작업은 할당을 통하여 운용자에게 통보되며 이때 통보되는 작업은 지역적, 시간적 사항을 고려하여 그림 1 에서와 같이 CORBA를 활용한 이벤트 서비스를 통하여 처리한다.

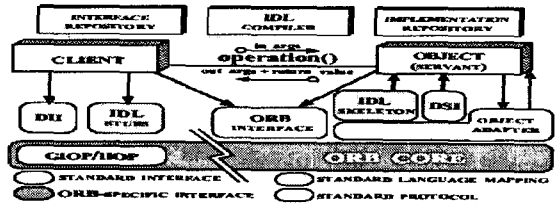


그림 1. CORBA 구조도

3.4 작업관리기능 구조

본 논문에서 제안한 모델은 룰에 의한 템플릿을 이용하여 작업자에게 단위작업을 접수 및 분배를 하며 우선순위에 의하여 작업을 수행하도록 한다. 그림 2 에서와 같이 설계한 모델은 8개의 에이전트로 구성되어 있다. 각각의 에이전트는 작업단위의 접수 및 분배부터 작업 할당까지의 역할을 수행한다.

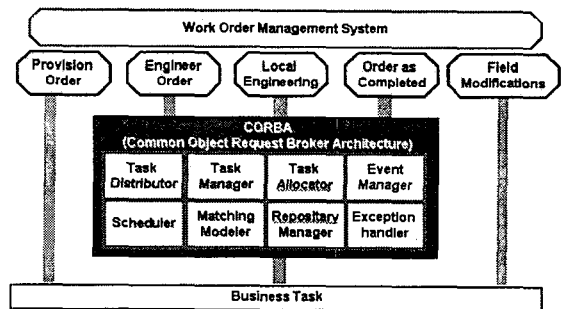


그림 2. 제안된 Model Architecture

1. 작업분배자(task distributor) : 단위작업의 접수와 분배의 역할을 담당한다. 작업분배자는 단위작업을 접수한 후 저장소(repository)에 자료를 저장한다. 이때 작업의 우선순위를 비교하여 저장한다. 우선순위가 높은 것은 priority 를 높게 주어 작업할당자가 작업할당시 먼저 처리하도록 한다.

2. 작업통제자(task manager) : 작업의 할당과 분배사이에서 전체적인 작업의 흐름을 조절한다. 작업통제자는 작업관리기능의 전체적인 흐름을 관리한다.

3. 작업할당자(task allocator) : 작업할당자는 저장소로부터 작업사항을 읽어서 작업을 처리하도록 한다. 템플릿과의 비교를 위하여 모델일치자를 실행시켜 일치하는 템플릿 유형을 찾아낸다. 템플릿이 정상적인 경우 이벤트 관리자에게 작업을 넘겨준다.

4. 이벤트관리자(event manager) : 이벤트관리자는 작업이 할당되면 관련 작업자에게 즉시 통보하여 작업이 이루어지도록 한다. 이벤트 관리자는 지역적으

로 분산되어 있는 작업자들을 위하여 조직모델에 의하여 구성된 부서에 작업처리를 통보한다.

5. 개인별 일정관리자(scheduler) : 개인별 일정관리자는 작업할당자의 지시에 의하여 할당 조직의 개인별 일정 현황을 확인하여 작업의 우선순위와 개인별 일정을 비교하여 작업을 개인별로 할당한다. 이때 운용자정보와 작업유형, 위치 등을 고려하여 작업을 할당한다.

6. 모델일치자(matching modeler) : 템플릿의 유형을 비교하여 작업의 요청에 해당하는 작업할당을 결정한다.

7. 저장관리자(repository manager) : 각 에이전트의 정보를 저장하고 활용할 수 있도록 관리한다.

8. 예외처리자(exception handler) : 작업할당자의 요청에 따라 불일치 자료를 처리한다. 예외처리자는 예외처리 기능 외에 히스토리 기능(history management), 효율적인 인적자원 활용(effective management of human resource)이 필요하다.

4. 시험결과

3장에서 설계된 정보모델링 구조로 설계된 작업관리기능은 작업관리의 통합성과 효율성 및 분석된 기능적 요소의 설계 목표를 반영하였다. 분산환경하에서의 객체서비스를 제공하기 위하여 CORBA 를 기반으로 객체를 포함하였고 주요 환경은 Sun W/S 환경하에서 Java 을 이용하여 기능을 설계 및 시험하였다. 또한 분산환경을 지원하는 CORBA 는 Visibroker for Java 4.0 제품을 활용하여 객체를 구현하였다.

4.1 기능 구현

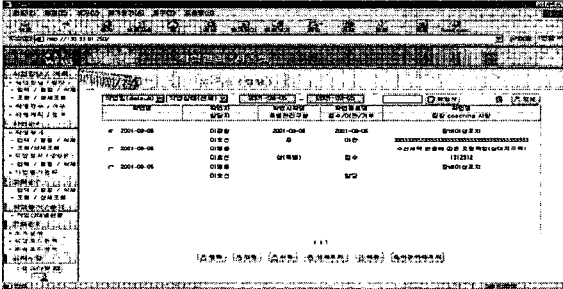


그림 3. 작업처리 기능

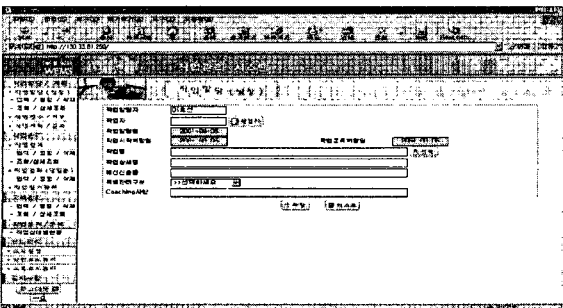


그림 4. 작업할당 기능

4.2 구현결과

업무의 활용과 효율성을 고려한 기능의 평가는 다음과 같다.

4.2.1 분산객체환경의 지원

분산객체환경을 실현하는 표준기술인 CORBA 를 채택하여 사용자에게 지역적으로 분리되어 있는 여러 장소의 장비와 이기종 시스템의 자원과 정보를 공유하여 통합된 시스템에서 처리되는 듯한 투명성을 제공한다.

4.2.2 작업수행 시간의 단축

여러 지역의 업무수행자들이 서로 협력적이고 의존적인 다중 태스크로 구성되며, 실시간처리 업무도 있지만 대부분 복잡하고 장기간에 걸쳐 수행되는 작업으로 이루어진다. 그리고 잘못된 작업정보, 작업의 지연, 중복작업에 의하여 작업수행시간은 지연된다. 구현된 기능은 프로세스 룰 과 데이터의 동기화로 동시작업의 처리를 가능하게 한다.

4.2.3 동적 트랜잭션의 권한/트랜잭션 무결성 지원
트랜잭션의 처리에 있어 작업의 승인(approve)이나 작업의 거부(reject) 등으로 프로세스의 설계를 변경하도록 설계되어 동일 프로세스상의 작업에서 사용자들간의 의사결정의 충돌을 방지한다.

4.2.4 작업 진행사항의 모니터링/관리

기능이 제공하는 프로세스 모니터링 사용자 인터페이스를 활용하여 작업의 진행상황과 지연상황을 직접 모니터링 함으로써 작업 프로세스의 추적 및 진척관리를 효율적으로 관리할 수 있도록 지원하였다.

5. 결론

본 논문에서 설계된 작업관리기능은 첫째, 통신망 운용에서의 다양한 서비스의 반영이 용이하도록 한다. 둘째, 작업단위의 업무는 정형화되어 있고 또한 비정형화된 작업단위는 예외처리로 변경 등이 가능하다. 셋째, 작업단위의 처리부서의 작업흐름이 전체적으로 관리되어야 한다는 사항을 고려하였다.

설계된 기능은 비즈니스의 프로세스 통합측면에서 분산객체환경에서 템플릿을 이용한 프로세스의 모델링, 프로세스의 제어 및 통제, 태스크의 수행, 모니터링 등 고유의 기능을 갖고 네트워크를 통하여 공유 자원과 서비스를 이용한 객체들의 집합으로 정의하여 모델링하였다.

제한한 작업관리기능의 모델은 작업요청에서부터 예외처리까지 일련의 흐름을 연결하여 신속한 서비스 제공 및 운용업무의 효율성을 제공하였다. 그러나 룰에 기반한 모델이기 때문에 다양한 서비스의 추가에 따라 예외처리의 문제는 계속 발생할 수 있다. 이러한 기능은 향후 복잡한 작업의 할당을 knowledge-based의 인공지능을 활용한 기술, heuristic Control 지원[6], 더 효율적인 자원 할당 및 계획 등을 필요로 할 것이다.

기업 통합의 핵심인 IT 통합 기반의 역할로서 작업관리기능의 향후 과제는 작업관리를 기반으로 기존(legacy) 시스템과의 연계 및 확장을 무엇(what)과 어떤 방법(how)의 관점에서 유연한 개방형 구조의 작

업관리의 모델링 및 전환 전략의 연구를 필요로 할 것이다.

6. 참고문헌

- [1] Q.Long and V.Lesser, "A Heuristic Real-Time Parallel Scheduler Based on Task Structures", University of Massachusettes Technical Report 95-92, 1995.
- [2] P.Muth, J.Weissenfels, M.Gillmann and G.Weikum, "Integrating Light-Weight Workflow Management Systems within Exciting Business Environments", Proceeding of International Conference on Data Engineering(ICDE), Sydney, Australia, 1999.
- [3] O.Bukhres, D.Hoang and C.Davenport, "Image Work-flow Management System Architecture for Internet Integration", proceeding of the 1st Int'l conference on web Information Systems engineering, Voll, pp 173-177, June 2000.
- [4] W.M.P van der Aalst, "A class of Petri net for modeling and analyzing business processes", Computing Science Reports 95/26, Eindhoven University of Technology, Eindhoven, 1995.
- [5] A. Sheth, M. Rusinkiewicz, "On Transactional Work-flows", *Bulletin of the Technical Committee on Data Engineering*, 16(2), IEEE Computer Society, June 1993.
- [6] Q.Long and V.Lesser, "A Heuristic Real-Time Parallel Scheduler Based on Task Structures", University of Massachusettes Technical Report 95-92, 1995.