

워크플로우관리시스템 개관

An Overview of Workflow Management Systems

배혜림, 김영호, 강석호

서울대학교 산업공학과

초록

워크플로우관리시스템(Workflow Management System, WFMS)은 기업의 비즈니스 프로세스를 컴퓨터화하고 이를 자동으로 실행해주는 소프트웨어이다. BPR 이후 계속된 기업의 프로세스에 대한 관심은 WFMS의 중요성을 부각시켜주었으며, 최근에는 ERP, CRM, EAI 등과 같은 기업 정보시스템의 핵심모듈로서 기존 시스템 통합의 필요성을 널리 인정받고 있다. 본 논문에서는 먼저 WFMS의 소개를 통해 WFMS의 이해를 돕고, 이를 토대로 그 필요성과 중요성을 설명하고자 한다. 또한, WFMS가 가지는 장점을 현재 국내외 워크플로우 시스템의 현황을 통해 살펴보고, 저자들의 다년간 연구, 개발 경험을 토대로 한 WFMS의 연구분야를 소개한다. 이를 통해, 앞으로 WFMS의 향후 전망과 정보시스템의 미래를 조망하는 것이 본 논문의 목표이다.

1. 서론

워크플로우(Workflow)는 컴퓨터에 의해 자동으로 실행 및 관리되는 업무 프로세스를 말하며, 워크플로우관리시스템은 이 워크플로우를 정의하고 그에 따라 업무 프로세스를 정의, 실행, 관리해주는 소프트웨어 시스템이다[10].

최근의 정보시스템 추세는 네트워크의 발달로 인한 기업간 전자적 거래를 통한 e-Business(Electronic Business)이며 이러한 추세는 ERP (Enterprise Resource Planning), SCM (Supply Chain Management), CRM (Customer Relationship Management)과 같은 대단위 소프트웨어 패키지의 경쟁적 도입을 초래하였다. e-Business 환경에서 이와 관련하여 새로운 키워드로 등장한 것이 통합이다. 이로 인해 ebXML, Biztalk과 같은 XML (eXtensible Markup Language) 기반의 프레임워크가 분산환경의 e-Business를 지원하는 표준으로 자리 잡아가고 있으며 EAI(Enterprise Application Integration)가 기업내 여러 정보시스템간의 인터페이스를 통합하는 해법이 될 수 있는 것으로 알려져 있다. 이러한 통합에 있어서 중요한 요소로 부각되고 있는 것이 프

로세스이며, 각 정보시스템과 표준 프레임워크가 프로세스 관리를 핵심모듈로 채택하고 있음은 이를 증명해주고 있다.

WFMS는 프로세스에 대한 자동화를 추구하는 시스템으로 최근 10여년 동안 활발하게 연구되어 왔으며, 이러한 연구는 소프트웨어를 직접 개발하는 업체뿐 아니라, 미국, 유럽, 일본 등의 학계에서도 많은 연구 결과들이 보고되고 있다. 본 연구에서는 WFMS를 소개하여 시스템의 이해를 돕고 그 필요성과 중요성을 설명하며, 저자들의 연구, 개발 경험을 토대로 WFMS의 연구분야를 소개하고 향후 전망을 제시한다.

2. 워크플로우관리시스템 기본 구조

WFMS는 그 본질상 여러 하부 모듈로 구성되며 프로세스를 수행하기 위해서는 많은 구성원들이 참여하고 여러 종류의 자원들을 이용한다. 따라서, WFMS를 이루는 여러 컴포넌트는 분산되어있는 이기종 시스템들인 경우가 대부분이다. 이러한 WFMS를 이해하기 위해서는 WfMC(Workflow Management Coalition)라고 하는 표준화 단체에서 제안한 워크플로우 참조모델(Workflow Reference Model, WRM)을 이해하여야 한다. 대부분의 상용시스템들은 WRM을 기반으로 만들어지며 WRM에서 제안하는 표준 인터페이스를 준수하고 있다.

WRM은 WfMC에서 제안한 워크플로우 기본 참조 모델로 다섯 개의 컴포넌트와, 이들 다섯 개 컴포넌트와 엔진 사이의 인터페이스를 명세하고 있다. 먼저 이들 다섯 개의 컴포넌트의 핵심모듈은 엔진에 해당하는 중앙 서버이다. 이 모듈은 정의된 프로세스를 해석하여 시작시키며 올바르게 수행이 될 수 있도록 제어 관리한다. 다음으로 프로세스 정의도구(Process Definition Tools)는 프로세스를 설계할 수 있는 그래픽 사용자 인터페이스를 제공하고 이를 컴퓨터가 해석 가능한 형태로 만들어 저장한다. 워크플로우 클라이언트(Workflow Client Applications)는 프로세스를

이루고 있는 단위업무를 사용자가 수행할 수 있는 사용자 환경을 제공해준다. 호출 응용프로그램(Invoked Applications)는 단위업무의 수행을 위하여 엔진에 의하여 호출, 실행되는 프로그램 모듈이다. 마지막으로 관리 및 모니터링 도구(Administration & Monitoring Tools)는 프로세스를 관리, 감독할 수 있는 사용자 환경을 제공한다.

이들 컴포넌트들간의 관계는 그림1에 표현되어 있으며 이들 사이에 정의된 인터페이스의 내용은 다음과 같다[10].

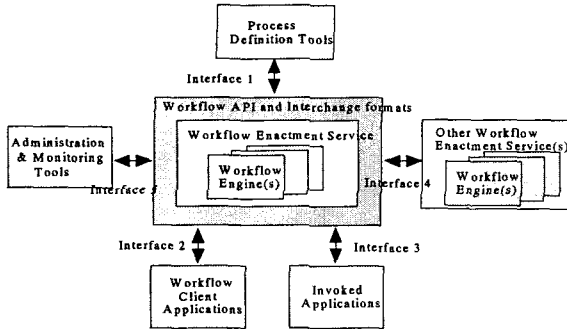


그림 1. 워크플로우 참조 모델 [10]

- 1) 인터페이스1 (IF1): 프로세스 정의의 교환에 대한, 엔진과 프로세스 정의도구간의 인터페이스 표준
- 2) 인터페이스2&3 (IF2&3): 엔진과 워크플로우 클라이언트 및 피호출 응용프로그램간의 인터페이스에 대한 표준
- 3) 인터페이스4 (IF4): 서로다른 워크플로우 엔진간의 상호운영성을 위한 인터페이스 표준
- 4) 인터페이스5 (IF5): 관리 및 감독 도구와 엔진간의 인터페이스 표준

3. 워크플로우 시스템의 필요성 및 연구분야

3.1 기업 프로세스 관련 시스템 현황

기업의 프로세스를 자동화해주는 세계적인 WFMS로는 Staffware사의 Staffware[15], Action technology의 ActionWorkflow, 그리고 후지쯔의 i-Flow 등이 있으며, 웹기반의 사용자 환경, 이동통신장치와의 연계, 폼 기반의 문서작업 통합 등의 기능을 제공하면서 시장을 넓혔으며, 국내에서는 핸디소프트 등에서 폼문서 지원, 웹기반 사용자 인터페이스, 멀티 서버 기능을 가진 시스템을 소개하여 국내 Workflow시장을 개척하고 있다.

한편, 학계의 연구로는 미국의 LSDIS (Large Scale Distributed Information Systems) 연구실의 METEOR 프로젝트[13],

IBM 연구소의 Exotica 프로젝트[14] 등을 들 수 있으며 국내에서는 ETRI와 서울대 등에서 활발한 연구를 수행하고 있다. 각 연구단체의 구체적인 연구분야 및 연구내용은 표기된 참고문헌을 참고하기 바란다.

3.2 워크플로우관리시스템의 필요성

워크플로우 관리 시스템은 기업이 e-Business를 위하여 경쟁적으로 도입하고 있는 각 정보시스템의 통합에 있어 중심축이 될 수 있으며 프로세스를 기반으로 기업활동을 응집시킬 수 있는 좋은 방법론이 된다. 워크플로우관리 시스템의 구체적인 필요성과 중요성은 다음과 같이 정리할 수 있다.

- 정보 통합/공유의 기반: WFMS는 프로세스 진행에 포함되는 여러 응용프로그램을 연동하도록 함으로써 정보를 통합해서 관리하고 사용자가 정보를 공유하여 효과적으로 작업할 수 있도록 한다.
 - 단위 정보시스템 확장의 기반: ERP, SCM, CRM 등과 같은 대단위 정보 시스템이 함께 공존하는 환경에서 요구되는 사항은 이들 시스템간의 연동이다. XML을 이용한 웹기반 WFMS는 이들 시스템이 확장되고 상호 연계되도록 한다.
 - 대규모 프로세스 구현의 기반: 엔터프라이즈 환경에서는 여러 기업에 걸친 정보 교환과 프로세스 통합을 지원하는 시스템에 대한 필요성이 점차 드높아지고 있으며 WFMS의 표준안 중 하나인 워크플로우간 상호 운영성을 통해 이러한 기업간 대규모 프로세스의 구현이 가능하다.
 - 지식 관리의 기반: 기업의 핵심 프로세스를 지식화 시켜주는 도구로서 필요한 정보 지식을 정의하고 정보의 공유 및 축적 체제를 지원할 수 있는 WFMS는 지식관리를 위한 핵심 엔진이자 기간 기술로 응용된다.
 - 프로세스 개선 기술의 기반: BPR (Business Process Re-engineering)과 같은 프로세스 혁신의 도구로 사용이 될 뿐 아니라 프로세스 진행의 결과물인 여러 데이터의 가공을 통해 프로세스 개선의 기반을 마련해준다.
- #### 3.3 워크플로우 관련 연구분야
- 프로세스 모델의 정확성에 관한 연구: 프로세스 모델은 업무프로세스의 추상화에 관한 것으로 대부분 이를 위한 공식화를 제공하며 프로세스 자체에 대한 분석도 포함된다. [1], [2], [3] 등의 연구가 이에 해당한다.
 - 확장 트랜잭션 모델 (Extended Transaction Model)을 이용한 프로세스 실행: 트랜잭션 워크플로우(Transactional Workflow)에 관한 연구로 ACID

(Atomicity, Consistency, Isolation, Durability) 속성을 완화시킨 트랜잭션 모델을 통해 워크플로우의 실행과 제어를 연구한다[5].

- 분산 워크플로우 실행 및 에이전트 응용: 네트워크 환경에서 워크플로우의 엔진을 분산시켜서 실행함으로써 서버의 과부하를 방지한다. 특히, 이동성을 갖추고 있으면서 지능적인 에이전트를 통해 프로세스를 진행시키는 프레임워크에 대한 연구가 [11] 등에서 수행되었다.
- 워크플로우간 상호 운영성: 기업환경이 변모함에 따라 서로 다른 기업, 혹은 부서들이 채용하고 있는 워크플로우 시스템간에 상호 작용해야 하는 필요성이 증대되고 있다. 이에 따라 이들간의 상호운영성을 추구하고자 하는 시도가 이루어지고 있으며 DCOM (Distributed Component Object Model), CORBA (Common Object Request Broker Architecture), EJB (Enterprise Java Beans) 등의 분산객체기술을 이용하는 방법과 최근에는 플랫폼에 독립적인 XML을 이용하고자 하는 연구들이 이루어지고 있다.
- 프로세스의 동적 변경: 이미 실행되고 있는 프로세스의 변경요구를 처리하고자 하는 노력에 관한 연구로 현재까지 많은 연구들이 이루어져왔다. 대표적인 연구로는 동적 변경을 처리할 수 있는 연산을 정의하는 방법[16], 실행시 encapsulation을 이용하는 방법[12] 등이 있다.
- 예외 처리 및 복구: 워크플로우 실행에서 발생하는 예외상황에 대처하며 이들에 대한 복구를 다루는 연구분야이다. 대부분의 연구는 ECA (Event Condition Action) 규칙을 기반으로 접근하는 경우가 많으며[8], [9]와 같이 워크플로우 예외처리를 위한 알고리즘을 개발하는 경우도 있다.

4. 새로운 Workflow 연구분야

앞의 3절에서는 기존 워크플로우 연구분야를 살펴보았다. 본 절에서는 저자들의 연구 경험을 바탕으로 워크플로우의 기능을 새롭게 하고 정보시스템의 발전 및 기업환경의 변화 추세를 반영할 수 있는 연구분야를 설명한다.

4.1 효과적인 Monitoring

워크플로우 관리 시스템은 많은 사용자들이 관여하여 프로세스를 진행하는 시스템이며, 각 사용자들이 프로세스에 대하여 가지는 관심이 서로 다르다. 따라서, 시스템이 사용자에게 적합한 자료를 사용자가 원하는 형태로 표현하는 것이 필요하게 된다. 이와 관련한 대표적인 적용사례는 워크플로우 모니터링이며 사용자, 혹은 시스템으로부터 이벤트가 발생했을 때,

원하는 데이터를, 적절한 연산을 통해 가공한 후, 사용자가 원하는 형태로 표현해주는 것이 필요하다.

4.2 프로세스와 문서 처리의 연계

프로세스를 진행하는 데는 많은 응용데이터들이 필요하며 가장 대표적인 것은 문서이다. 많은 WFMS들은 문서관리시스템 기능을 부분적으로 내장하거나 문서 전달 기능을 포함하고 있다. 그러나, 이러한 문서의 처리는 프로세스 관점에서 이루어질 때, 그 효과가 배가될 수 있으며, 이를 통해 문서의 저장, 검색의 효율화를 추구하여야 한다. 특히, XML 문서는 워크플로우 프로세스에서 사용하는 품문서로 사용될 수 있고 문서작업을 특정 단위업무와 연결시킬 수 있도록 문서를 구조화할 수 있다.

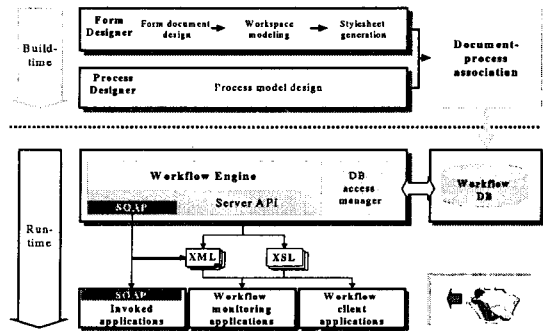


그림 2. 프로세스와 연계된 문서처리구조

그림2는 문서를 프로세스와 연계시키기 위한 시스템 구조로 프로세스를 정의할 때 문서를 단위업무와 연계시킨 뒤 이를 실행시에 자동으로 전달, 저장해준다.

4.3 Data 가공을 통한 규칙 생성

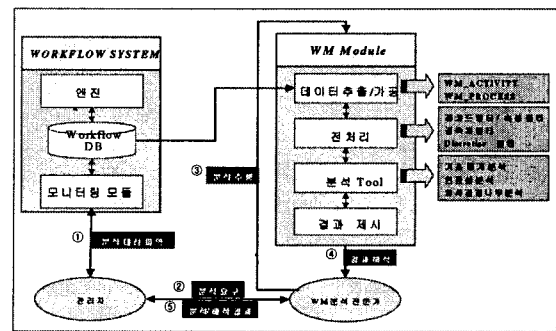


그림 3. 프로세스데이터 가공을 통한 규칙생성

WFMS가 효과적으로 적용되는 분야는 대개 프로세스의 발생빈도가 빈번하며 정형화된 형태로 프로세스가 진행되는 경우가 많다. 따라서 많은 데이터들이 쌓이게 되는데 이들을 가공하여 규칙화하면 프로세스 개선에 중요한 정보를 줄 수 있다. 그림3은 워크플로우 데이터베이스로부터 규칙을 생성하는 시스템의 구조를 보여준다.

4.4 프로세스 개선 및 진화

기존의 프로세스에 관한 연구의 초점은 프로세스의 동적 변경에 관한 것으로 앞서도 기술한 바와 같이 규칙을 이용한 접근법이 대부분이었다. 프로세스와 관련하여 새로운 접근이 필요한 분야는 프로세스의 개선에 관한 것이며 이러한 프로세스 개선 모델이 자기진화(Self Evolution) 형태를 띠도록 하는 것이다. 자기진화 모델은 전자회로구성과 같은 하드웨어 분야에서 많이 응용되어 왔으나[7] 워크플로우 프로세스에도 응용될 가능성이 있으며, 실제로 워크플로우에 진화개념을 도입하고자 하는 시도들이 최근에 이루어지고 있다[4].

4.5 XML 기술을 이용한 상호 운영성

서로 다른 워크플로우 시스템간의 상호 운영성에 관한 연구는 관련 표준화 단체인 WfMC를 중심으로 활발하게 진행되어 왔다. 특히, Wf-XML binding[17]이라는 표준을 통해 XML 기반으로 상호운영성에 관한 표준적인 API (Application Programming Interface)를 정의하였다. 최근에는 SOAP (Simple Object Access Protocol)과 같은 프로토콜 표준이 등장했고 ebXML, BizTalk과 같은 XML 기반 전자상거래 프레임워크에서 프로세스에 대해 강조를 하고 있는 상황이어서 XML을 기반으로 한 WFMS간의 상호운영성이 절실히 필요한 시점이라고 할 수 있다.

5. 결론

본 연구에서는 WfMC의 워크플로우 참조 모델을 기준으로 WFMS의 기본 구조를 간단히 설명하고 그 중요성과 필요성에 대하여 살펴 보았다. 특히, 인터넷이 발달하면서 파생된 e-Business 환경에서 WFMS가 프로세스를 축으로 각종 대단위 소프트웨어를 엔터프라이즈 환경에서 통합하여 기업간의 정보를 주고 받는 매개체로서의 역할을 넓혀나갈 것이라는 점을 강조하였다. 또한, 다년간의 연구 경험을 토대로 앞으로 WFMS의 기능을 풍부하게 할 수 있는 새로운 연구분야를 제안하고 이에 대해 간단히 설명하였다.

워크플로우 관리 시스템은 프로세스를 기본으로 각 기업간 정보교환 및 인터페이스를 원활하게 할 것이며, 데이터베이스의 등장이 데이터를 응용으로부터 분리했던 것처럼, 프로세스를 응용으로부터 분리하여 비즈니스의 OS로 자리잡게 될 것이다.

참고 문헌

1. W. M. P. van der Aalst, "Formalization and verification of event-driven process chains," *Information and Software Technology*, vol. 41, no. 10, pp. 639-650, 1999.

2. I. B. Arpinar, U. Halici, S. Arpinar and A. Dogac, "Formalization of Workflows and Correctness Issues in the Presence of Concurrency," *Distributed and Parallel Database*, pp.199-248, 1999.

3. J. Bae, S. Jung, Y. Seo, Y. Kim and S. Kang, "Integration of Workflow Management and Simulation," *Computers and Industrial Engineering* 37, pp. 203-206, 1999.

4. F. Casati, S. Ceri, B. Pernici, G. Pozzi, "Workflow evolution," *Data & Knowledge Engineering*, vol. 24, no. 3, pp. 211-238, 1998.

5. A. Dogac, E. Gokkoca, S. Arpinar, P. Koksak, I. Cingil, B. Arpinar, N. Tatbul, P. Karagoz, U. Halici, and M. Altinel, "Design and Implementation of a Distributed Workflow Management System: METUFlow," *Proc. NATO Advanced Study Institute (ASI) Workshop on Workflow Management Systems and Interoperability*, pp. 61-66, Aug. 1997.

6. Extensible Markup Language(XML) 1.0, W3C Recommendation 10-February 1998, <http://www.w3c.org/TR/REC-xml.html>

7. H. de Garis, M. Korin, F. Gers, E. Nawa and M. Hough, "Building an artificial brain using an FPGA based CAM-Brain Machine," *Applied Mathematics and Computation* 111, pp. 163-192, 2000.

8. A. Goh, Y.-K. Koh, and D. S. Domazet, "ECA Rule-based Support for Workflows," *Artificial Intelligence in Eng.*, vol. 15, no. 1, pp. 37-46, 2001.

9. C. Hagen and G. Alonso, "Exception Handling in Workflow Management Systems," *IEEE Trans. Software Eng.*, vol. 26, no. 10, pp 943-958, Oct. 2000.

10. D. Hollingsworth "Workflow Management Coalition Specification: The Workflow Reference Model", WfMC, 29 November, 1994

11. A. Inamoto, "Agent Oriented System Approach for Workflow Automation," *International Journal of Production Economics* 60-61, pp. 327-335, 1999

12. Y. Kim, S.H. Kang, D.S. Kim, J.S. Bae, and K.J. Joo, "WW-flow: A Web-based Workflow Management System Supporting Run-time Encapsulation," *IEEE Internet Computing*, vol. 4, no. 3, pp. 55-64, May/June 2000.

13. J. Miller, D. Palaniswami, A. Sheth, K. Kochut, and H. Singh, "WebWork: METEOR's Web-based Workflow Management System," *Journal of Intelligent Information Systems*, vol. 10, no. 2, pp. 185-215, 1998

14. C. Mohan, G. Alonso, R. Guenthoer, R., Kamath, "Exotica: A Research Perspective on Workflow Management Systems," *Data Engineering Bulletin*, vol. 18, no. 1, pp19-26, March 1995.

15. Staffware Pic., Staffware 2000 Functionality, White Paper, <http://www.staffware.com>, 1994.

16. G. Vossen and M. Weske, "The WASA Approach to Workflow Management for Scientific Applications," *Advanced in Workflow Management Systems and Interoperability*, Istanbul, Turkey, pp.145-165, 1997.

17. WfMC, Interoperability Wf-XML Binding, WfMC standards, WfMC-TC-1023, 11-Jan-00, <http://www.wfmc.org>