

□ 특집 □

워크플로우 모형화 및 관리시스템

김 소 연[†] 이 강 수^{††}

◆ 목 차 ◆

- | | |
|----------------------------|-------------------------------|
| 1. 서론 | 4 패트리넷 기반 워크플로우 모형화 및 관리방법의 예 |
| 2. 용어 정의 및 관점 | 5 결론 |
| 3. 일반 워크플로우 모형화 및 관리 프레임워크 | |

1. 서론

비즈니스 프로세스 수행 시간의 90%는 기다리거나 전달하는데 소모된다고 한다[1]. 따라서, 수행 시간을 단축하여 비즈니스 프로세스의 생산성과 품질을 높이고자하는 시도가 경영, 시스템, 컴퓨터, 산업공학 및 정보통신 분야 등에서 활발히 진행되고 있다. 이들 각 분야에서의 연구 관점, 해결 방법 및 용어들은 다소 다르지만, “비즈니스 프로세스의 재설계와 자동화를 통한 생산성과 품질 향상”이라는 공통의 최종 연구 목표를 갖는다.

이러한 목표는 최근의 사회적 환경 변화에 의해 연구 동기가 부여되었고, 컴퓨터 기술 및 정보통신 하부구조의 발전에 의해 실현 가능성이 높아지게 되었다. 여기서, “비즈니스 프로세스”란 서비스, 소프트웨어 개발, 제조 및 금융 산업을 포함한 모든 산업에서의 생산과정을 의미하며, 학교의 교육과정도 포함된다.

이러한 배경에서, 본 글의 목적은 최근 활발히

사용되고 있으며 혼동적인 분야인 워크플로우, BPR, 소프트웨어 프로세스 공학, 정보시스템 공학, 엔터프라이즈 모형화, 동시성 공학, 컴퓨터 통합생산 등의 개념을 정립하고, 특히 “워크플로우 모형화 및 관리 시스템”(WFM)의 일반적인 사항의 조사, 분석 결과를 제시한다. 또한, 제시된 결과의 적용예로서, 패트리넷을 기반으로한 WFM 방법을 제시한다. 이는 워크플로우 분야에 패트리넷의 응용 가능성을 보인 것이다.

본 글의 2장에서는 WFM분야의 개념적 혼동을 줄이기 위해, 이 분야와 관련된 연구 분야들을 정의하고 WFM에 대한 다양한 관점들을 논한다. 3장에서는 모형화의 일반적인 개념과 기존의 WFM들의 공통적인 프레임워크인 일반(generic) WFM에 관해 논한다. 4장에서는 일반WFM의 한 사례라 할 수 있는, 패트리넷을 WFM의 기본 모형으로서 활용하는 방법(즉, “패트리넷 기반 WFM”)을 제시하고 5장에서 결론을 맺는다.

2. 용어 정의 및 관점

WFM분야는 연구자의 배경과 관점(예; 소프트웨어공학, 데이터베이스, 정보통신, 시스템 통합

[^] 본 연구는 1996년도 한남대학교 지원 연구 결과의 일부임.

[†] 정희원 : 한남대학교 전자계산공학과 박사과정

^{††} 정희원 : 한남대학교 전자계산공학과 교수

및 경영정보학 등)에 따라 그 정의와 해석이 다르다. 따라서, 이 분야는 가장 용어의 혼란이 심한 분야중의 하나로서, 관련 용어와 관점을 명확히 해야한다.

2.1 워크플로우 모형화 및 관리

그림 1처럼 어떤 조직내의 “프로세스”는 조직의 특성과 관점에 따라 워크플로우(일반적인 경우에서), 비즈니스 프로세스(주로 기업에서) 및 소프트웨어 프로세스(소프트웨어 개발조직에서)란 용어를 사용한다. 따라서, 세 가지 용어는 실세계의 프로세스에 대한 이미지에 해당하며 거의 같은 개념이다.

(1) 프로세스

제품의 개발 및 유지보수에 관련된 모든 실세계 요소로서, 생산 프로세스, 메타 프로세스 및 지원 프로세스로 구분되며, 활동 또는 타스크, 역할(즉, 인간 및 기계의 역할) 및 참조(즉, 문서, 자

료, 자원에 대한 참조)로 구성된다. 프로세스의 예는 인공물(artifact), 생산 지원도구, 활동(activity) 및 에이전트 등이다. 프로세스를 구성하고 있는 “활동”은 프로세스내의 관리되지 않은 스텝을 의미하며, 처리요소(즉, 사람 또는 컴퓨터 등)에 의해 실행된다. 활동은 재충구조이며, 병행성, 비동기성 및 조건적 수행성을 갖는다. 한편, “타스크”는 관리된 스텝을 의미한다.

(2) 워크플로우

“다수의 인원이 관련된 프로세스에서 사용된 순차적 및 동시적 순서 또는 스텝”으로 정의되며, 활동, 활동간의 제어구조(통신, 협동, 조화 및 동기) 및 활동간의 자료흐름 구조로 구성된다[2~6]. 따라서, 워크플로우는 분산된 조직내의 자료흐름 및 제어흐름을 통합한 개념으로서, computer support cooperative work[7], form processing, cooperative system 및 OA[8] 개념으로부터 유래되었으며, 다음과 같은 특성을 갖는다[2~6].

- 활동의 불확실성, 분산성, 추상성, 변화성, 반복성

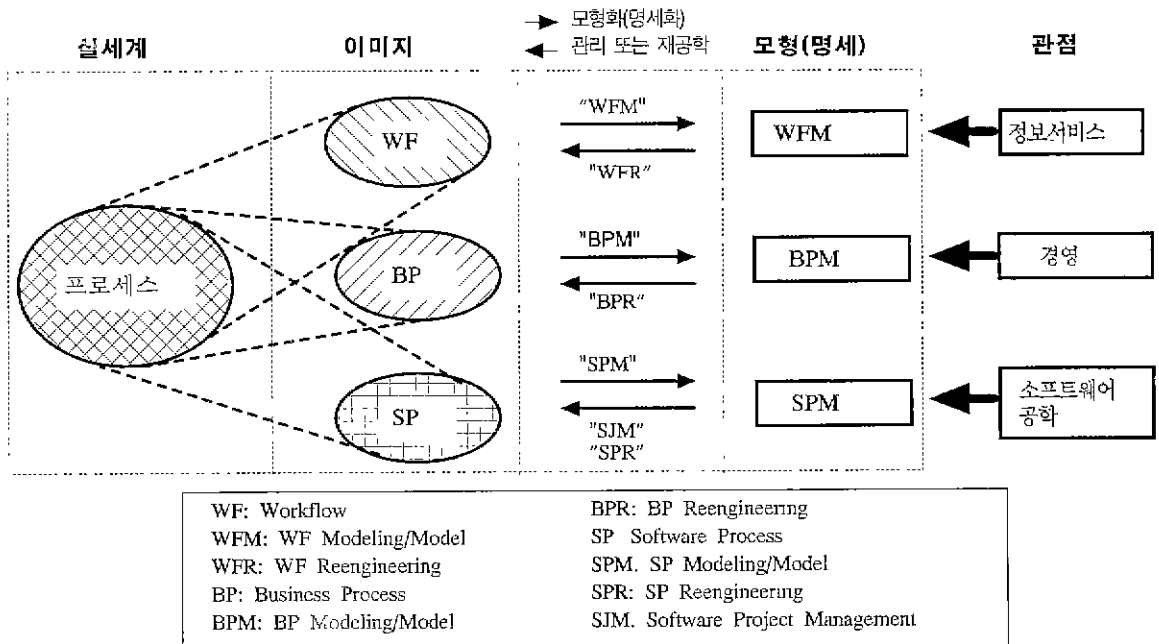


그림 1. WFM과 관련 분야에서 관점에따른 용어의 차이

- 활동간의 협동성, 병행성과 비동기성
- 사건 지향적인 상태변화
- 멀티미디어 객체의 분산처리
- 오랜 지속시간(long-duration, long-lived)동안 지속됨: 수행시간이 수일 부터 수개월
- 상태의 저장량이 많으며, 상태의 역추적 기능이 있어야함

(3) 워크플로우 모형화(WFM) 또는 명세화

어떤 조직의 현재 또는 새로운 워크플로우를 이해, 분석, 관리, 자동화 및 재설계하기위해 “명세화”하는 과정을 의미하며, 명세된 결과를 “워크플로우 모형”이라 한다. 이때 사용된 명세 방법 또는 언어(예; 활동 다이어그램 등)를 “명세모형”이라 한다. 명세모형은 3장에서 소개할 것이다. 모형화시 프로세스(즉, 워크플로우)의 시작 및 종료, 활동들의 정의, 활동의 흐름, 활동간의 관계 및 통신 등 워크플로우의 특성을 사실 그대로 명세 한다.

(4) 워크플로우 관리(WFM)

관리의 목표에 따라서 다음 사항들을 수행하는 것이다[2~6, 9~17].

- ① 분석시: 현재 또는 재설계된 워크플로우의 특성을 분석한다. 워크플로우의 특성들은 정적 또는 동적 특성, 주관적 또는 객관적 특성, 정성적 또는 정량적 특성으로 분류할 수 있으며, 예를 들어 성능, 효율, 서비스의 질, 보안성, 업무의 생산성 등을 의미한다.
- ② 관리시: 분석의 결과로부터, 워크플로우의 현재 상태를 파악하고 성능, 효율, 생산성 및 보안성들의 저하 문제들을 관리적인 방법으로 개선한다. 여기서, 관리적인 방법이란 현재의 워크플로우하에서 조직내의 활동과 자원을 감시, 통합, 재분배, 조정, 회복 및 스케줄링하는 것을 의미한다.
- ③ 자동화시: 워크플로우 자동화는 워크플로우 연구의 최종 목표이며, 모형화된 워크플로우대로 조직의 프로세스를 자동화한다. 즉, 워크플로우 모

형을 프로그래밍(즉, 제어 프로그램의 역할)하여 이를 수행하므로써, 모형대로 프로세스를 자동으로 실행하는 것이다. 이를 위해서는, 컴퓨터 및 정보통신 기술(예; 분산객체 관리, 분산 OS, 분산 DB, 템포랄 DB, 미들웨어, 그룹웨어, 초고속 정보통신망 등), 멀티미디어 기술, 보안 기술(예; 전자서명, 전자화폐, EDI), 제어 기술 등이 종합적으로 필요하다. 이들의 공통점은 동시성 제어, 고장처리, 인터페이스, 보안 및 스케줄링 문제이다.

④ 설계 및 재설계시: 새로운 조직에 대한 워크플로우를 설계하거나, 현재 워크플로우의 효율이 낮을 때 혁신적으로 워크플로우를 재설계한다. 여기서, 재설계는 BPR과 같은 개념이다. 새로운 워크플로우는 일반적인 소프트웨어 개발 라이프 싸이클(즉, 워크플로우 계획, 분석, 설계, 구현, 시험 및 유지보수 단계)처럼 개발 및 운영된다.

2.2 워크플로우와 그 유사분야와의 차이점

본 절에서는 워크플로우를 포함한 “프로세스” 개념에 기반을 둔 분야들을 고찰함으로써, 그 차이점을 명확히 하고자 한다.

(1) 비즈니스 프로세스 리엔지니어링(Business Process Reengineering; BPR)

BPR은 “비용, 품질, 서비스 및 처리속도와 같은 현재의 성능 측정치를 극적으로 향상시키기 위해, 비즈니스 프로세스를 근본적으로 재 사고하고 혁신적으로 재 설계하는 것”으로 정의된다[1,22]. 따라서, BPR은 현재의 프로세스나 조직의 제약조건을 고려하지 않고, 서비스 제공 방법 또는 제품 생산 방법을 완전히 재 사고하는 것이다. 워크플로우 분야는 BPR의 기술적 및 절차적 측면을 강조한 것이다.

- ① 비즈니스 프로세스(BP): 주로 BPR을위해 경영분야에서 사용되는 용어로서, 워크플로우와 대응되며 가장 광범위하게 사용되는 용어이다.
- ② 비즈니스 프로세스 모형화 및 관리(BPM): 워크플로우 모형화 및 관리에 대응하는 용어로서,

모형화의 종류는 다음과 같다. (a) BP의 기능 모형화: 활동과 그들간의 자료흐름 명세, (b) 행위 모형화: 활동 수행시간 및 순서 명세, (c) 조직 모형화: 활동의 장소 및 주체, 통신, 저장 매체 명세, (d) 정보 모형화: 프로세스내의 자료 엔티티, 구조 및 상호관계 명세.

(2) 소프트웨어 프로세스 공학(Software Process Engineering; SPE)

SPE는 좁은 의미의 “소프트웨어 공학”이라 할 수 있으며, 다음과 같은 개념들에 공학적인 원리를 부가한 것이다.

① 소프트웨어 프로세스(SP): 소프트웨어 제품을 생산 및 유지보수하기위해, 부분적으로 순서화된 프로세스 스텝의 집합으로서, 관련 인공물, 인력 및 컴퓨터 자원, 조직적인 구조 및 제약조건들을 포함한다[18~21]. 소프트웨어 개발 회사의 경영자 관점에서 보면, SP는 소프트웨어 개발을 위한 “비즈니스 프로세스” 또는 “워크플로우”와 같다. 소프트웨어 제품의 비즈니스(즉, 마케팅, 생산, 판매 등)는 다른 제품의 비즈니스 특성과는 다르므로, 특별히 “비즈니스” 대신 “소프트웨어”란 수식어를 붙인 것이다. 사실, “소프트웨어 개발 프로세스”란 용어가 정확한 표현이다.

소프트웨어 제품은 제조되는 것이 아니라, 개발(엔지니어링)되는 것이므로, 개발 프로세스 자체가 곧 제품이 된다. 또한, 제품의 품질과 생산성은 개발 프로세스의 품질과 효율성에 비례한다. 예를 들어, 대학을 소프트웨어 개발 기업으로 비유한다면, 학위를 받은 졸업생은 소프트웨어 제품에 비유되고, 커리큘럼 및 학습 과정은 소프트웨어 프로세스에 비유된다. 따라서, 졸업생의 품질은 학위 자체가 아니라 4년간의 학습 과정에 의해 결정된다.

② 소프트웨어 프로세스 모형화 및 관리(SPM): WFM의 특수한 경우 해당하며, 특히, “소프트웨어 개발”을 위한 WFM을 SPM이라 할 수 있으며 표 1은 WFM과 SPM의 차이를 보인다. 이 분야의 연구는 소프트웨어공학 차원에서 활발하며[18~21]

다음과 같은 문제점들이 있다. (a) 용어와 개념의 혼돈, (b) 모형화(명세) 언어와 파라다임이 다양(50개 이상), (c) 타입과 인스턴스의 구분 애매, (d) 개념의 변화가 심하며, 표준화가 부족.

<표1> 워크플로우와 소프트웨어 프로세스 관리의 차이[19]

항목		워크플로우 관리	소프트웨어 프로세스 관리
프로세스 특성	프로세스 특성	관리적 절차	개발 프로세스
	용용성	낮음	높음
	스텝의 유형	사전에 완전 정의	창조적인 스텝 포함
	결과의 의미	분석 또는 평가 문서	응용의 수행
사용자의 역할 인식	행동	프로세스를 승인	실제 응용
	책임	책임지는 핵심업무 수행 및 허부 활동 위임	관리자는 조정 및 업무 위임
	표준	표준화된 규칙	결과의 표준화
프로세스 문서	표기법	제어흐름	자료흐름
	필요성	법적, 제도적 필요	활동의 기록(관리용)
	기록 저장	모든 기록 보존	일부 기록만 보존
보안성	권한	통제된 권한 부여	없음
	접근방법	접근제어 실시	특허물과 변경 기록 통제
	시스템 고장	프로세스 상태 보존 및 재시동 가능	특허물 보존 및 고장 내인성
다중성	트랜잭션의 병행성	낮음	높음
	프로세스 수	많음(수백개 이상)	적음(수십개 이하)
	인스턴스 수	적음 (각 트랜잭션당 1개)	많음(수백개)
	fan-in/fan-out	적음	많음
프로세스 실행	순서	미리 정의됨	자료의 특성에 따름
	후속기능	적음, 기능 선택권 적음	많음
	반복	통제되고 공식 정의됨	필요하며 반복 결과를 기록

③ 소프트웨어 프로젝트 모형화 및 관리(Software Project Modeling/Management; SJM): SPM의 응용분야라 할 수 있으며 거의 같은 의미로 사용되는 개념이다. 여기서, “소프트웨어 (개발) 프로젝트”란 시작 시간, 특정목적, 제약조건, 책임, 예산, 스케줄, 종료 시간으로 특성 지워지는 잠정적인 활동의 집합이며, “프로젝트 관리”는 주어진 자원

하에서 소프트웨어 개발 프로젝트를 성공적으로 관리하기 위한, 계획, 조직, 인적구성, 방향 및 제어를 제공하는, 절차, 실시, 기술 및 노하우 시스템으로 정의된다. 또한, “프로젝트 모형화”는 소프트웨어 개발 프로젝트의 양상을 설명하기 위한 인공물 또는 활동의 표현을 의미한다[23]

(3) 기타 유사 분야들[19]

① 정보 시스템 공학(Information Systems Engineering; ISE): ISE는 “정보 시스템을 개발하기 위한 체계적인 공학적 접근방법들의 응용”으로 정의된다. 여기서, “정보 시스템”은 어떤 문제의 정보처리 요구사항을 충족시키기 위해 개발, 훈련 및 조립된 컴퓨터, 소프트웨어 및 인간과 조직 컴포넌트들로 구성된 시스템이다.

② 엔터프라이즈 모형화(Enterprise Modeling; EM): EM은 “모형을 구축하므로써 어떤 복잡한 사회 조직을 이해하는 과정”으로 정의된다. 따라서, 모형 자체가 아니라, 프로세스의 모형화 과정을 강조한다.

③ 조직 설계(Organizational Design; OD): OD는 “프로세스 모형과 컴퓨터를 사용한 조직 성능에 대한 연구”로 정의된다. OD과정은 우선 가상적인 비즈니스 문제를 해결하는 가상적인 조직의 프로세스 모형을 개발하고, 컴퓨터 시뮬레이션을 이용해 이 조직의 성능을 예측하는 것이다. 정확히 예측할 수 있는 모형을 개발하기 위해, “조직이론”이 이용된다.

④ 동시성 공학(Concurrent Engineering; CE): CE는 “제품과 그 관련 프로세스(제조 및 지원 포함)를 통합적이고 동시적으로 설계하기 위한 체계적인 접근방법”으로 정의된다. 제품 개발의 착수 및 개념화 단계로부터 폐기 단계에 이르는 제품 라이프 싸이클의 요소(예; 품질, 비용, 스케줄 및 사용자 요구사항)를 사용자가 모두 고려할 수 있게 한다. CE의 주 아이디어는 정보기술과 조직 재구성을 통해 활동의 병렬성을 극대화하므로써, 제품 출하 시간을 단축하고, 품질과 생산성을 높이며

생산 원가를 절감한다는 것이다.

⑤ 컴퓨터 통합 생산(Computer Integrated Manufacturing; CIM): CIM은 “제품의 모든 생산 공정들이 통합되고 컴퓨터에 의해 제어되는, 반자동 또는 전자동 공장 개념”으로 정의되며, 기존의 유연 생산셀(FMS)과 공장 자동화(FA)를 통합한 개념이다. CIM을 구축하기 위해서는 컴퓨터 기술, 정보통신 기술, 생산 및 자재 관리 기술, 제어계측 기술, 경영 기술이 종합적으로 필요하다.

⑥ 그룹웨어(Groupware)[7]: 다양한 하드웨어 및 소프트웨어 환경에서, 다수의 분산된 사람들이 공통적인 업무를 효율적으로 수행할 수 있게 하는 소프트웨어를 의미한다. 전자결재, 전자우편, 원격 화상회의, 주문형 비디오와 같은 서비스를 제공하며, 이를 위해서는 초고속 종합 정보통신망, 대용량 데이터베이스, 분산처리 운영체제 등이 필요하다.

2.3 워크플로우 모델링 및 관리의 관점

(1) 데이터 베이스(DB) 관점

워크플로우는 분산 온라인 트랜잭션 프로세싱(OLTP)을 포함한 DB 분야에서 가장 널리 사용되는 개념으로서, DM 개념을 확장하여 BP나 SP를 자동화할 수 있도록 하지는 것이다. 즉, 프로세스(또는 워크플로우)내의 모든 자원은 데이터로, 활동은 트랜잭션으로, 활동간의 제어순서는 DB프로그램으로 각각 간주하므로써, DB시스템을 프로세스 자동화에 이용한다는 것이다.

일반적으로, 프로세스(또는 워크플로우)는 오랜 지속시간을 갖으며, 활동(또는 트랜잭션)들간의 병행성, 동기성, 협력성, 분산성, 사건 지향성을 가진다. 따라서, DB분야에서는 트랜잭션 개념을 확장하였으며, 1990년 Long duration(long-lived) transaction[16]과 Flexible transaction[15] 개념으로부터 Transactional workflow[6], Multi DB transaction model, Extended transaction model, New transaction model, Activity management, Third generation transaction

<표 2> 프로세스 기술을 사용하는 분야들의 비교

분야	항목	적용 대상 분야	프로세스 기술의 사용 목적	프로세스 요소
WFM	정보 서비스 (예; 금융, 보험 등)		이해, 가시화, 생산의 부분적 자동화	비즈니스 규칙, 활동, 인공물, 역할 (인공물의 흐름 경로를 강조)
BPR	모든 분야		현 프로세스의 이해, 미래 프로세스의 분석 및 의사소통	활동, 역할
SE	소프트웨어 개발		이해, 가시화, 생산의 부분적 자동화	활동, 인공물, 역할, 도구 (개발 프로세스와 메타프로세스를 각각 모형화)
ISE	정보 서비스 (예; 금융, 보험 등)		특정 분야의 지식에 대한 이해, 분석, 의사소통 (프로세스의 감시, 측정, 실행을 강조하지 않음)	활동, 역할, 도구, 인공물, 비즈니스 규칙, 인적 행위
EM	모든 분야		이해, 프로세스에 대한 논의	목적, 비즈니스 규칙, 활동, 인공물, 역할
OD	모든 분야		이해, 시뮬레이션 (공식적이고 상세한 모형화 필요)	활동, 인공물, 역할, 행위자, 도구 (행위자와 그 능력을 상세히 모형화, 시뮬레이션 강조)
CE	모든 분야		이해, 의사소통, 분석	활동, 역할, 인공물
CIM	제조 (설계보다는 제조 프로세스를 강조)		시뮬레이션, 계획, 측정, 자동화	활동, 인공물, 도구 (인적요소를 강조하지 않음)

monitor, Interactive transaction, Changed transaction 등의 개념과 그 관리 시스템들이 등장하였다[2~6, 9~17]. 이들은 워크플로우 모형 및 관리분야에 대한 DB의 솔루션들이라 할 수 있다.

(2) 소프트웨어 공학 관점

소프트웨어공학 분야에서는 SP에 관한 연구가 활발하다[18~21]. 이는 소프트웨어의 개발 프로세스를 자동화하고 관리(즉, 프로젝트 관리)함으로써, 생산성과 품질을 향상하기 위한 것이다. 소프트웨어의 개발환경은 점차 분산되고 협력적이므로, 그룹웨어가 필요해지고 있다. 즉, 개발 환경은 프로세스나 워크플로우의 특성을 갖는다. 따라서, 최근에는 소프트웨어 프로세스 분야에 워크플로우나 BP개념을 이용하고 있으며[15], BP나 워크플로우의 특수한 경우가 SP이다.

(3) 시스템 통합(SI) 및 BPR관점

SI나 BPR분야에서도 프로세스의 자동화를 위해, 확장형 DB나 워크플로우 개념을 다루고 있다. 일반적인 비즈니스를 대상으로 한다는 점 이외에는 DB나 소프트웨어 공학분야에서의 관점과 유사하다.

3. 일반 워크플로우 모형화 및 관리 프레임워크

3.1 모형과 모형화의 기본 개념

“모형”이란 연구중인 객체 또는 시스템의 중요한 현상 또는 요구사항을 위협이나 비용의 부담 없이 가급적 자동 및 수학적으로 분석 할 수 있도록 명세한 것이다[24]. 모형화할 대상의 특성, 분석 수준과 목표에 따라 다양한 모형이 존재한다. 자연과학 분야에서, 모형이란 수학적인(또는 수식화된) 모형을 의미한다. 그러나, 병행성이나 불확실성이 높은 시스템은 수학적으로 모형화하

기가 어려우므로, 그래프 모형이나 확률 모형 등을 이용한다. “모형화” 또는 “명세화”란 모형을 제작 또는 기술하는 절차를 의미하며, “명세모형”이란 모형화에서 사용된 표기법, 이론, 알고리즘 및 방법론을 의미한다.

3.2 일반 워크플로우 모형화 및 관리 시스템

일반적으로, 어떤 문제가 주어졌을 때 이를 해결하는 방법들은 많은 공통점을 가지고 있다. WFM문제의 경우에도 많은 연구 결과들[2~16]이 제시되어 있지만, 이들은 대부분 그림 2와 같은 기본적인 프레임워크에 약간의 변형을 가한 것들이다. 즉, 사용한 모형과 명세형태 등이 다를 뿐이다.

분석용 명세모형으로 사용하면 될 것이다. 특히, 워크플로우 모형화를 위해서는 워크플로우의 특성을 모형화 할 수 있고 정형적, 도형적 및 수행 가능적(즉, 오포레이션날)인 모델이 추천된다. 다음은 “소프트웨어 프로세스 명세모형 또는 분석 지원도구”들을 파라다임별로 분류하여 보이며, 이들에 대한 상세한 설명은 생략한다[18~22].

- 비 수행가능: ETVX(IBM), IDEFO(SADT), SADT, object-oriented notation
- 상태기반:
 - 상태기계: Statemate, FSM, SREM, PAISley, SDL, Esterel, Statechart, RTRL, ESM, ECSP
 - 패트리넷: FUNSOFT net(MELMAC), SLANG(SP-ADE), Process Weaver, TPN(timed Petri-nets), SPN

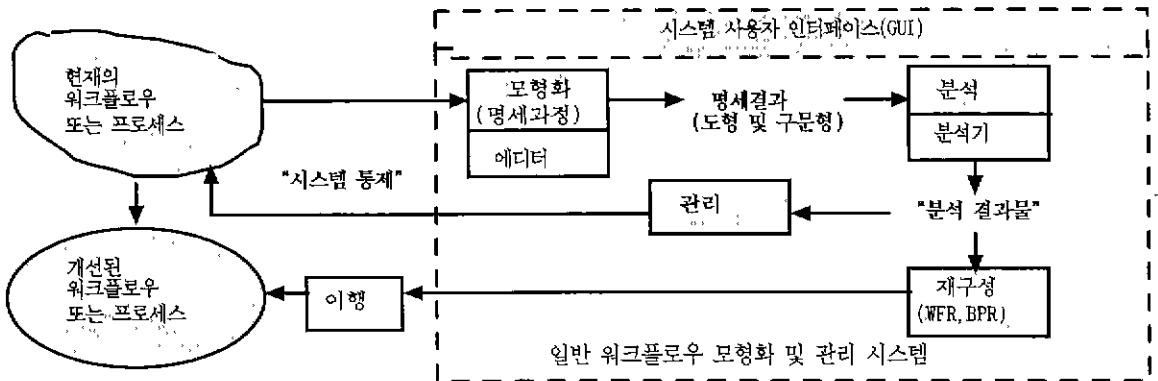


그림 2. 일반 워크플로우 모형화 및 관리 프레임워크

(1) 워크플로우 명세모형의 종류

비즈니스 프로세스나 워크플로우 모형화 분야는 아직까지 활동 다이어그램, 사건 다이어그램, 구조도, 자료흐름도, 상태 전이도 등 비정형적이며 도형적인 명세모형들이 이용되고 있으며, 이 분야 적용할 수 있는 “정형적 명세”(formal specification) 모형의 연구가 필요하다. 소프트웨어 프로세스 분야에서는 소프트웨어 공학에서 연구된 많은 정형적 명세모형들을 이용하고 있으며 [18~22], 이들을 수정하여 워크플로우 모형화 및

(stochastic Petri-nets), CPN(Colored Petri-nets)

- Formal Grammars: HFSP
- 규칙기반: MARVEL, GRAPPLE, Interact/Intermediate, Prolog, Merlin, OIKOS, EPOS, ALF
- 트리저 기반: Adele
- 명령적(Imperative): APPL/A

(2) 워크플로우 모형화(명세과정)

워크플로우의 모형화 또는 명세는 현재의 워크플로우의 특성을 분석하기 위해 관심 있는 속성

을 추상화하는 과정으로서 다음과 같은 관점들에서 모형화 방법들을 분류할 수 있다.

① 형식성(formalism): “형식” 모형화와 “비형식” 모형화로 분류할 수 있으며, 특히 형식 모형화를 관계적으로 “모형화”이라 한다. DFD, Structured Chart, Pseudo Code, Object Diagram 등은 비형식 명세모형에 해당하며, 앞 절에서 제시된 각종 모형들을 이용하면 형식 모형화에 해당한다. WFM 작성자는 이들 형식 명세모형들 중 하나 또는 몇 개를 선택하여 WFM을 작성한다. 형식 모형화를 이용함으로써, 워크플로우의 해석 및 의사소통상의 애매성을 해소할 수 있고 수학적 분석이 가능해진다.

② 표기법: “구문” 모형화와 “도형” 모형화로 분류할 수 있다. Prolog 및 Ada와 같은 프로그램언어나, PSL, RSL, SDL, IDL, Z-notation, VDM 등과 같은 스크립트형 표현 방법은 구문 명세모형에 해당된다. 구문 모형화는 워크플로우와 같이 병행성이 있는 경우, 모형화 결과(즉, 명세결과)의 가시성을 높일 수 있는 도형 모형화가 필요하다. Pseudo code를 제외한 대부분의 비형식 명세모형들은 도형 모형화에 해당하며, 패트리넷, SDL, State Transition Diagram 등은 형식적이면서 도형적 모형화가 가능하므로 추천할 만 하다.

③ 수행성: “수행가능” (즉, 오퍼레이션널) 모형화와 “비수행가능” 모형화로 분류할 수 있다. 비수행가능 모형화는 모형화 결과가 워크플로우의 정적인 이미지를 표현할 뿐이며 정적인 특성밖에는 분석할 수 없다. 오토마타(상태 기계)에 기반을 두지 않은 대부분의 명세모형들은 이 부류에 속한다. 패트리넷을 포함한 오토마타에 기반을 둔 모형들은 수행가능하며, 분석기 내에 수행엔진이 필요하다. 예를 들어, 수행엔진은 C언어(모형에 비유)로된 C프로그램(모형화 결과, 즉 명세에 비유)의 컴파일러에 비유된다. 수행가능 모형화를 이용함으로써, 워크플로우의 정적인 특성 뿐 아니라 동적인 특성들도 분석할 수 있으며, 이전의 워크플로우 상태로 복귀 할 수도 있다.

④ 대상: 워크플로우내의 모형화 대상 요소에 따라, “조직구조”의 모형화, “활동순서”의 모형화 및 “자원”의 모형화로 분류할 수 있다. 조직구조의 모형화에서는 조직의 부서, 에이전트 및 역할들을 그래프 형태(예; 구조도, ER 다이어그램 등)로 표현한다. 활동순서의 모형화에서는 워크플로우를 구성하는 각 활동과 이들간의 순서관계를 활동 다이어그램(또는, 이와 유사한 도형) 형태로 표현하며, 자원의 모형화에서는 워크플로우내의 각종 자원을 표현한다. 각 자원은 그와 관계하는 활동과 연관시켜 표현한다.

시스템(조직)의 워크플로우에는 원칙적으로 이들 요소를 모두 모형화해야 하지만, 필요에 따라 요소들중 한가지만을 모형화할 수도 있다. 따라서, WFM에서는 선택적으로 모형화할 수 있는 기능이 필요하다. 예를 들어, 워크플로우내의 자금의 흐름이나 특정 서류의 흐름만을 분리해서 모형화하면 좋을 것이다.

(3) 워크플로우 분석

워크플로우의 분석은 다음과 같이 육하원칙 하에서 실시한다.

① 분석 목표와 목적의 설정: 어떤 시스템(또는 조직)의 워크플로우의 속성은 다양하므로, 분석의 목표를 명확히 한다. “워크플로우에서 효율성을 극대화해야 한다”, “워크플로우의 보안성이 유지되어야 한다.” 등은 분석 목표의 예이다. 분석 목적은 분석 목표를 구체화한 것이다.

② 분석 대상의 설정: 어떤 시스템(또는 조직)의 워크플로우는 복합적인 객체(예; 비용, 인원, 시간, 장비 등)들의 흐름이므로, 분석의 목표를 달성하기 위한 분석 대상을 식별한다. 분석 대상의 예를 들면, 임계적인(즉, 가장 시간, 인원, 장비가 많이 투입되는) 워크플로우 경로, 워크플로우상의 병목 지점, 워크플로우의 보안성 및 신뢰성, 비상사태에 대한 내인성 등이 있다.

③ 분석 방법 개발: 분석 방법은 WFM에서 사용된 명세모형에 따라 달라진다. 일반적으로 모형화

및 분석 지원 도구는 그림 2처럼 구성된다.

- 에디터 모듈: 구문 또는 도형적 모형화(명세)를 지원
- 분석기 모듈: 모형화 결과로부터 워크플로우의 일관성, 도달성 시나리오, 구조적 특성, 동적 특성(데드록 등), 정량적 특성(성능, 신뢰도 등) 등을 분석
- 관리기 모듈: 분석 결과물을 워크플로우 관리자에게 GUI형태로 제공하므로써, 워크플로우를 관리 및 통제용 자료를 제공
- 재구성 모듈: 워크플로우의 재설계 또는 BPR이 필요할 때 재설계 과정을 지원
- 워크플로우 데이터베이스: 그림 2에서는 생략되었지만, 워크플로우상의 자원 객체들에 대한 정보와 명세 및 분석 결과를 저장 및 검색

④ 분석자 및 분석 장소: 워크플로우의 분석은 시스템(또는 조직)내의 분석 팀에 의해 내부적으로 이루어지거나, 제삼자 기관(Third Party)에 의해 이루어진다. 특히, 시스템의 워크플로우의 품질 또는 보안성을 “인증” 받고 적용되는 워크플로우의 운영을 “인가” 받으려면, 국가적으로 인가된 평가 및 인증기관에 의뢰하여 평가 및 분석을 실시한다. 보안에 관련된 워크플로우의 경우(예를 들면, 보안기관에서의 보안자료의 처리에 대한 워크플로우), DAC(discretionary access control), MAC(mandatory access control)과 같은 접근제어 정책과 기밀 정보의 흐름에 대한 통제정책이 워크플로우내에 포함되어 있는가를 평가 및 인증해야한다. 그러나, 이러한 “안전한(secure) 워크플로우”에 대한 연구는 아직 이루어지고 있지는 않다

⑤ 분석 시기: WFM은 시스템의 성능 평가와 마찬가지로, 어떤 신규 시스템(조직)의 워크플로우 구성시(잘 되었는가를 분석), 운영중인 워크플로우의 평가시(잘 운영되고 있는가를 분석), BPR을 위한 워크플로우의 개선 및 BPR을 위한 작업시(무엇이 문제인가를 분석)에 이루어진다.

4) 워크플로우 관리

측정할 수 있는 것만 통제 및 관리할 수 있다. 따라서, 관리 및 통제를 위해서는 분석과정에서 얻은 결과물들을 활용하여 현재의 워크플로우에 대한 관리를 실시한다. 즉, 시스템내의 가용 자원인 인원, 장비, 자금 등을 적절하게 재분배하고 스케줄링을 수정한다. WFM의 주목적은 워크플로우 관리의 자동화이므로, 자원의 재분배 및 스케줄링도 자동화할 수 있으면 좋다.

(5) 워크플로우 재설계

현재 시스템의 워크플로우 자체가 비효율적이라면, 통제와 관리활동만으로는 일정 수준 이상의 효율을 기대할 수 없다. 따라서, 워크플로우 자체의 재구성 또는 BPR이 필요하다. 앞에서 얻은 모형화(명세화)와 분석의 결과를 이용하면, 워크플로우의 현재 상태를 쉽게 이해하고 재구성이 용이해진다.

4. 패트리넷 기반 워크플로우 모형화 및 관리 방법의 예

3장에서 제시한 “일반 워크플로우 모형화 및 관리 프레임워크”의 응용 방법을 보이기 위해서, 본 장에서는 패트리넷을 기본적인 모형으로한 “패트리넷 기반 워크플로우 모형 및 관리 시스템”을 제시한다. 패트리넷은 프로젝트 관리를 위한 “소프트웨어 프로세스 모형”분야에서는 활발하게 사용되고 있으며, 그림 3은 제시된 시스템의 프레임워크를 보인다.

4.1 워크플로우 모형화

(1) 패트리넷

1962년 독일의 패트리에의해 제시된 패트리넷 [24]은 통신 프로토콜, 병행성 제어, 소프트웨어 프로세스 모형화, 분산 시스템, 공정 관리, 병렬 처리 시스템 등과 같은 병행성, 비결정성(nondeterminism), 비동기적 특성을 갖는 시스템들의 모형

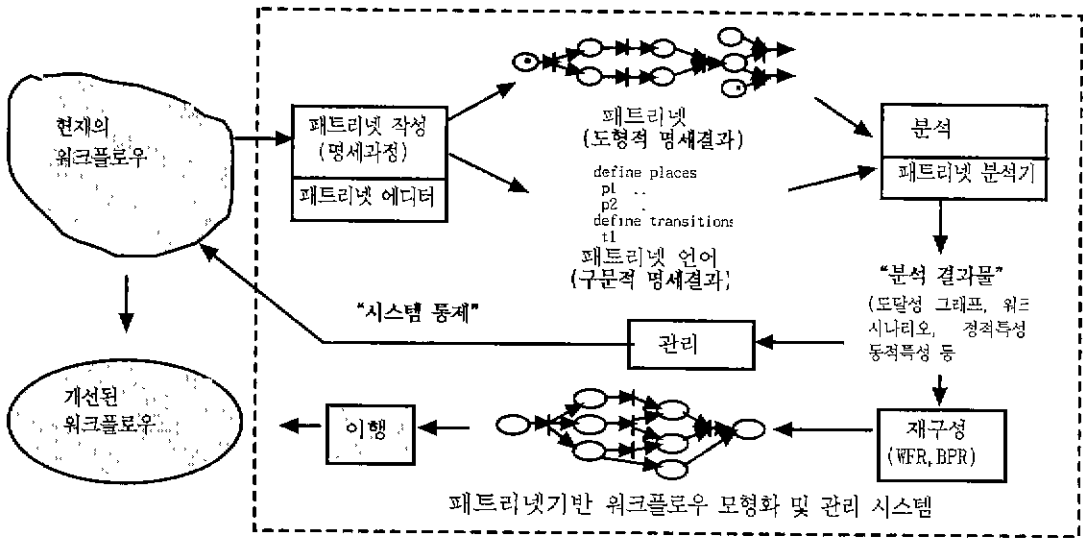


그림 3. 패트리넷 기반 워크플로우 모형화 및 관리 프레임워크

화(또는 명세화)와 분석 모형으로서 널리 알려져 있다. 패트리넷은 그래프의 한 형태이므로 시각성이 높으며, 하나의 오토마타(튜링머신과 유한 상태 기계의 중간에 위치)이므로 형식적인 명세와 분석이 가능하고 수행이 가능한 오퍼레이셔널 명세모형이다. 또한, 패트리넷은 국내외적으로 이론적인 연구결과가 많으며 많은 분석 지원도구들이 상용화되어 있다[24,25] 따라서, 패트리넷은 워크플로우 모형화를 위한 명세모형으로서 이용되기에 충분한 특성을 지니고 있다.

일반적으로, 패트리넷을 이용하여 시스템을 모형화 및 분석하는 방법은 다음과 같다.

- ① 모형화 및 분석 목표 정의: 모형화 및 분석할 시스템의 특성과 모형화 및 분석의 목표를 설정한다.
- ② 패트리넷의 조정 및 그 이론적 특성 분석: 응용할 시스템의 특성과 모형화 및 분석 목표에 따라, 표준적인 패트리넷을 확장한다[25]. 성능평가를 위해 시간과 확률 개념을 추가한 Timed Petri Net(TPN) 및 Stochastic Petri Net(SPN), 퍼지 이론을 추가한 Fuzzy Petri Net 등은 조정된(또는, 확장

된) 패트리넷의 예들이다.

- ③ 패트리넷 작성: 응용할 시스템을 패트리넷으로 그리는 작업으로서, 완성된 패트리넷은 응용할 시스템을 반영(즉, 모형화)한다. 완성된 패트리넷은 시스템의 형식, 도형적 및 수행가능 명세에 해당된다. 또한, 패트리넷은 도형적이지만 스크립트(패트리넷 언어)형태로도 나타낼 수 있으므로, 구문적 명세가 될 수도 있다.
- ④ 패트리넷 분석: 패트리넷의 정적인 구조를 이용하여 시스템의 정적인 특성을 분석할 수 있다. 또한, 패트리넷 수행엔진을 이용하여 패트리넷을 수행하므로써, 도달성 그래프를 얻고 동적인 특성 등을 분석할 수 있다. 패트리넷은 분석할 시스템을 모형화 했으므로, 패트리넷의 특성은 곧 분석할 시스템의 특성을 나타낸다.
- ⑤ 분석 결과의 이용: 분석된 결과들을 이용하여 시스템에 대한 통제와 관리를 실시한다.
- ⑥ 시스템 재구성: 분석된 현재의 패트리넷(즉, 현재의 시스템의 워크플로우)의 문제점 및 비효율성을 개선하여 새로운 패트리넷을 재구성 및 재설계한다.

(2) 패트리넷의 조정[25]

워크플로우를 구성하는 각 활동들은 입력 및 출력 자원을 포함하고 활동의 수행 지연시간이 불확실하게 분포된다. 따라서, “표준형” 패트리넷 [24]의 트랜지션(워크플로우내의 각 활동들을 모형화)에 객체 개념을 레이블링하고, 점화 지연시간을 베타분포를 갖는다고 가정하면, 트랜지션은 “활동트랜지션”이 되고, 이렇게 조정된 패트리넷은 “객체지향형 베타분포 시간형 패트리넷”이 된다.

여기서, 각 트랜지션에 레이블링된 “객체”는 활동을 위한 “자원”과 자원의 처리 “메소드”로 구성된다. 자원은 자금, 시간, 필요 장비, 인력 등으로 구성되며 각 자원에 대한 속성 값이 부여된다. 메소드는 각 자원의 처리 절차 또는 알고리즘을 의미한다. 점화 지연시간(즉, 활동의 지연시간)을 베타분포로 가정한 것은, 프로젝트 관리 모형에서 가장 널리 사용되는 PERT/CPM모형으로부터 도입한 것이며, 베타분포는 인간에 의해 수행되는 활동들의 지연시간 분포를 잘 표현하고 있다[24].

(3) 패트리넷 모형화 절차

패트리넷을 작성하는 일은 자료흐름도 또는 플로우차트를 작성하는 일처럼 작성자의 직관과 시행착오가 필요하며 “패트리넷 에디터”를 이용하면 좀더 빠르게 작성할 수 있다. 패트리넷의 작성과정은 곧 워크플로우에 관한 형식, 도형적 및 수행 가능 명세과정이기도하며, 다음 과정을 통해 작성된다.

- ① 시스템(또는 조직)내의 활동 객체 조사: 각 활동 객체는 활동트랜지션으로 모형화
- ② 활동 객체간의 부분적 인과 관계 조사: 몇 개의 활동트랜지션(두꺼운 선으로 표현)을 포함한 부분 패트리넷들의 구성. 이때 활동트랜지션만으로는 부분 관계를 표현할 수 없으므로, 즉시(immediate)트랜지션(얇은 선으로 표현하며, 부분 관계만을 표현하기 위한 것임), 플레이스(원으로 표현) 및 아크(화살표로 표현)를 이용한다.
- ③ 부분적 인과 관계로부터 전체 활동순서 조사: 부분 패트리넷으로부터 전체 패트리넷을 단계적

으로 구성해 간다.

이렇게 완성된 워크플로우의 도형적 명세인 패트리넷은 필요에 따라 스크립트 형태(패트리넷 언어)로 표현할 수 있으며, 이 경우 워크플로우의 구분적 명세가 된다.

4-2 워크플로우 분석 및 관리

(1) 정적 특성 분석

패트리넷의 구조적 특성을 분석하므로써, 워크플로우의 정적 특성인, 복잡성, 구조적 생존성, 보존성, 일관성, 불변성 등을 분석한다.

(2) 동적 특성 분석

패트리넷에 초기 마킹(워크플로우의 초기 상태에 해당)을 부여한 후 “패트리넷 수행엔진”에 의해 수행하여 “도달성 그래프”를 얻는다. 도달성 그래프에는 패트리넷의 동적 특성들이 포함되어 있으므로, 이를 분석함으로써, 워크플로우의 동적이며 정성적인 특성인, 도달성(즉, 특정 활동이 실제로 수행되는가), 데드록 여부(워크플로우상에 데드록이 있는가), 자원 보존성(시스템내의 자원이 무한대로 필요되지 않는가), 자료흐름, 제어흐름, 워크 시나리오, 공평성(상호 보완적인 활동들이 공평하게 수행되는가) 등을 분석한다. 또한, 조정된 패트리넷이 TPN, SPN 및 FPN이라면, 워크플로우의 동적이면 정량적인 특성(즉, 성능값, 신뢰도값, 안전도값 등)을 분석할 수 있다. 패트리넷의 수행 과정은 워크플로우의 시뮬레이션 과정을 의미하며, “역수행”하므로써 이전의 상태로 복귀할 수도 있다. 이는 워크플로우 모형이 가져야 할 특성중의 하나이다.

(3) 워크플로우 관리

패트리넷은 분석하려는 워크플로우를 나타내므로, 패트리넷의 “수행”결과는 실제 시스템의 워크플로우의 수행 결과를 나타낸다. 즉, 패트리넷의 도달성 그래프는 워크플로우의 수행시에 가능한 모든 사건과 전이 및 그 결과로 나타나는 상태

변화를 나타낸다. 따라서, 워크플로우 관리자는 워크플로우의 모든 상태를 파악할 수 있으므로, 관리 및 통제를 실시할 수 있다.

이 개념은 하드웨어 장비의 제어분야에서 많이 쓰이는 program logic controller(PLC) 방식과 유사하다. 즉, 패트리넷은 PLC에서 내장된 제어프로그램에 해당되고 패트리넷 수행기 및 관리부분은 PLC에서 제어프로그램의 수행기에 해당된다. 그러나, 패트리넷 방식은 병행성과 비결정성을 표현하므로, 기존의 순차적인 PLC와는 달리 병행 및 비결정적 제어가 가능하다.

4.3 워크플로우 재설계 및 BPR

기존의 WFM방법들에서는 현재 시스템의 워크플로우를 모형화(명세)하고 분석 및 관리하는 방법을 주로 다루고 있으며, 분석된 결과를 활용하여 현재 시스템의 워크플로우를 재설계하는 방법이 부족하다. 본 연구에서 제시하는 방법은 현재의 워크플로우에 대한 패트리넷 명세와 분석된 결과를 그대로 활용하여 워크플로우 재설계 또는 BPR을 실시한다는 것이다.

현재의 워크플로우가 패트리넷으로 모형화되어 있으므로, 패트리넷을 재구성하고 재구성된 패트리넷에 대한 분석을 실시한다. 패트리넷에 대한 재구성은 패트리넷 에디터를 이용하므로써 쉽게 수행될 수 있고 재구성된 패트리넷도 분석기를 이용하므로써 고속으로 재분석될 수 있다. 재구성 과정은 워크플로우의 재구성 및 개선 효과가 목표치에 이를 때까지 단계적으로 반복한다. 새로운 패트리넷은 곧 개선될 시스템의 워크플로우를 나타내므로, 이 방법은 “패트리넷을 이용한 BPR 또는 WFR방법”에 해당된다.

5. 결론

본 글에서는 혼동적으로 사용되어온 워크플로우와 그 유사 분야들의 개념을 정립하고, 기존의 방법들을 일반화하여 일반 워크플로우 모형화 및

관리의 프레임워크를 제시했다. 또한, 이 프레임워크의 한 인스턴스를 보이기 위해 패트리넷을 기반으로 하는 패트리넷 기반의 워크플로우 모형화 및 관리 방법을 제시했다.

결론적으로 워크플로우, 비즈니스 프로세스 및 소프트웨어 프로세스 등의 분야는 비록 용어는 다르지만, 프로세스(공정) 개념을 서로 다른 배경과 관점에서 다룬 것에 지나지 않는다. 또한, 워크플로우의 특성들은 패트리넷 모형의 장점과 거의 일치하므로, 이 분야에 패트리넷을 이용할 수 있다. 이미 소프트웨어 프로세스 모형화 분야에는 패트리넷이 활발히 이용되고 있다.

이 분야의 향후 연구과제는 본 글의 내용에 대한 계속 연구로서 패트리넷을 워크플로우와 BPR 분야의 모형화와 분석에 이용하는 연구, 분산 멀티미디어 객체를 처리하는 워크플로우 모형화 및 안전성이 보장된 워크플로우(Secure Workflow) 모형화 및 관리 방법과 그 지원도구의 연구, 개발이다.

참고 문헌

- [1] M. Hammer, "Reengineering Work: Don't Automate, Obliterate," *Harvard Business Review*, pp.104-111, July-Aug. 1990.
- [2] "Special Issue on Workflow Systems", *Bulletin of the Technical Committee on Data Engineering*, Vol. 18-1, March 1995
- [3] "Special Issue on Workflow and Extended Transaction Model", *Bulletin of the Technical Committee on Data Engineering*, Vol. 16-2, June 1993.
- [4] "Database and Workflow Management: What is it About?," *Proc. 23rd VLDB*, 1995
- [5] "Transactional Workflow Management in Distributed Object Computing Environments", *Proc. 7th Data Engineering*, 1995.
- [6] A. Sheth, "Workflow Automation: Applications, Technology and Research", *Proc ACM SIGMOD*, 1995.
- [7] D. Bowen, "Work Group Research: Past

Strategies and Future Opportunities", *IEEE tran. on Engineering Management*, Vol 41-1, Feb. 1995, pp.30~38.

[8] S.K. Chang, W.Chan, "Transformation and Verification of Office Procedure", *IEEE tran. on S.E.*, Vol11-8, Aug. 1985, pp.724~734.

[9] M. P. Singh, "Synchronizing Distributed Constrained Events from Transactional Workflow Specifications", *Proc. IEEE Computer Society Technical Committee on Data Engineering(TCDE)*, Feb. 1996, pp.616~623.

[10] G. Alonso, et. al., "Advanced Transaction Models in Workflow Contexts", *Proc. TCDE*, Feb. 1996. pp.574~581.

[11] S. Sarin, "Workflow and Data Management in InConcert", *Proc. TCDE*, Feb. 1996, pp.497~499.

[12] D. Wodtke, et. al., "The Mentor Project: Steps Towards Enterprise-Wide Workflow Management", *Proc. TCDE*, Feb. 1996, pp.556~565

[13] W. Du, et. al., *Enterprise Workflow Architecture*, *Proc 7'th ICDE*, May 1995, pp.63~64.

[14] W. Jin, L. Ness, "Concurrency Control and Recovery of Multidatabase WorkFlow in Telecommunication Applications", *Proc SIGMOD*, May 1993, pp.456~459.

[15] G. Kaiser, "A Flexible Transaction Model for Software Engineering", *Proc 6'th ICDE*, Feb. 1990, pp.560~567.

[16] H. Korth, G. Speegle, "Long-Duration Transactions in Software", *Proc 6'th ICDE*, Feb. 1990, pp.568~574.

[17] R. Mora, K. Cartron, "ActionWorkflow in Use: Clark Country Department of Business License", *Proc TCDE*, Feb. 1996, pp.288~294.

[18] R. Warboys (ed.), "Software Process Technology", *Proc. 3'rd EWSPT, Lecture Notes in Computer Science*, Vol. 772, 1994.

[19] W. Schafer (ed), "Software Process Technology", *Proc. 4'th EWSPT, Lecture Notes in Computer Science*, Vol. 913, 1995.

[20] A. Finelstein, et al., *Software Process Modeling and Technology*, Research Studies Press, 1994.

[21] A. Fugetta, A. Wolf (ed.), *Software Process*, John Wiley & Son, 1996.

[22] K. Spurr, et al., (ed.), *Software Assistance for Business Re-engineering*, John Wiley & Son, 1995.

[23] Gang-Soo Lee, T. Murata, "A Beta-distributed Stochastic Petri Net Model for Software Project Time/Cost Management", *Journal of Systems and Software*, Vol. 26, 1994, pp.149~165.

[24] J. L. Peterson, *Petri Net Theory and Modeling of Systems*, Prentice Halls, Englewood Cliffs, New Jersey, 1981.

[25] 이강수, "페트리넷의 분류법", *한국정보과학회 논문집*, 21권 8호, Aug. 1994, pp. 1379~1389.



이 강 수

1981년 홍익대학교 컴퓨터공학과 졸업(학사)
 1983년 서울대학교 대학원 전산학과 졸업(이학석사)
 1989년 서울대학교 대학원 전산학과 졸업(이학박사)

1985년~1987년 국립대전산업대학 전자계산학과 전임 강사
 1992년~1993년 미국 일리노이대학교 객원교수
 1995년 한국전자통신연구소 초빙연구원
 1987년~현재 한남대학교 전자계산공학과 부교수
 관심분야 : 소프트웨어 공학, 실시간 시스템 모형화 방법론, 페트리넷 응용, 보안 프로토콜 모형화, 멀티미디어 동기화, 보안 평가, 워크플로우 모형화 등



김 소 연

1990년 한남대학교 전자계산공학과 졸업(학사)
 1992년 한남대학교 대학원 전자계산공학과 졸업(공학사)
 1994년~현재 한남대학교 대학원 전자계산공학과 박사과정

관심분야 : 소프트웨어 공학, 실시간 시스템 모형화 방법론, 페트리넷 응용, 워크플로우 모형화 등