

Unified Modeling Language를 활용한 다관점 업무 시스템 모형화

김 종 우[†] · 김 진 삼^{††} · 조 진 희^{†††} · 전 진 옥^{††††}

요 약

최근에 객체지향 프로그래밍 언어의 광범위한 활용과 더불어 객체지향 개발방법론, 객체지향 모형화 언어가 활발히 보급되고 있다. 객체지향 개발방법론을 채택하여 정보 시스템을 개발할 경우, 사전 작업인 업무 분석 및 재설계 작업을 객체지향 모형화 언어를 활용하여 진행하면 업무 분석 및 재설계 산출물을 정보시스템 개발자들이 쉽게 이해하고 정보시스템 개발에 참조할 수 있다. 본 논문에서는 객체지향 모형화 언어의 표준으로 대두되고 있는 UML(Unified Modeling Language)에 기반한 업무 프로세스 모형화 기법(UML-B, UML extension for Business Modeling)을 제시한다. UML-B에서는 UML의 표기법을 사용하여 업무 시스템의 조직 구조, 행위자, 사용사례, 업무 프로세스, 개체 관점을 표현하도록 한다. 또한 UML이 제공하는 확장 메커니즘을 사용하여 업무에 대한 모형화 작업의 용이성을 증진시키고 경영혁신(BPR, Business Process Reengineering) 기법을 객체지향 모형화와 함께 사용할 수 있도록 지원한다.

Multiple Perspective Business System Modeling Using Unified Modeling Language

Jong-Woo Kim[†] · Jin-Sam Kim^{††} · Jin-Hee Cho^{†††} · Jin-Ok Jeon^{††††}

ABSTRACT

Recently, due to the popularity of object-oriented programming languages, object-oriented modeling and development methodologies become widely applied to information system development. When object-oriented methodology is adopted, using object-oriented modeling languages for business analysis and redesign has the advantages such that business modeling results can be easily understood and referred by information system developers. In this paper, UML-B, Unified Modeling Language extension for Business modeling is proposed, which uses UML notation for modeling organization structure, actors, use cases, business processes, and entities in business systems. It also utilizes extension mechanisms of UML to facilitate business modeling activities, and supports business process reengineering with object-oriented modeling.

1. 서 론

객체지향 개발 방법론은 정보시스템의 복잡성과 분산화의 증가, 객체지향 언어의 보급과 함께 활발하게

활용되고 있다. 객체지향 개발 방법론은 80년대 말부터 활발히 논의가 되기 시작하여 Rumbaugh 등의 OMT (Object Modeling Technique)[14], Booch의 OOD (Object-Oriented Design)[4], Coad와 Yourdon의 OOA(Object-Oriented Analysis)[5], Jacobson의 OOSE(Object-Oriented Software Engineering)[10], Hewlett Packard사의 Fusion[6] 등이 제시되었다. 90년대 중반부터 객체지향

† 정 회 원 : 충남대학교 자연과학대학 통계학과 교수
 †† 정 회 원 : 한국전자통신연구원 선임연구원
 ††† 정 회 원 : 한국전자통신연구원 연구원
 †††† 중신회원 : 한국전자통신연구원 실시간컴퓨팅연구부 부장
 논문접수 : 1999년 2월 12일, 심사완료 : 1999년 8월 20일

개발 방법론의 표준화 논의가 활발해지면서 UML(Unified Modeling Language)[18], OPEN(Object-oriented Process, Environment, and Notation)[8] 등이 제시되었으나, Rational사의 UML이 향후 객체지향 모형화의 표준 표기법으로 정착될 것으로 보여진다[1, 3].

업무 시스템의 일부로 활용되어야 하는 정보 시스템의 개발에 있어서, 업무 시스템에 대한 분석 및 재설계 작업이 정보시스템의 분석 및 설계 작업에 선행해서 수행되어야 한다. 기존의 업무 시스템의 분석과 설계를 위해서는 정보공학[12], BPR(Business Process Reengineering)[7, 9] 등의 기법들이 주로 활용되어왔다. 객체지향 방법론을 근거로 정보시스템을 개발하는 경우, 업무시스템의 분석 및 재설계에 있어서도 객체지향 모형화를 근거로 표현하고 분석한다면, 업무시스템의 분석 및 재설계 산출물이 정보시스템의 개발자들에게 쉽게 이해되고 정보시스템 개발에 참조될 수 있을 것이다. 이러한 필요성 때문에 객체지향 업무 분석을 위한 방법론들로 Jacobson의 사용사례 분석(Use Case Analysis)[11], Convergent Approach[15], SOCCA[16], OOBIS(Object-Oriented approach to Business process Redesign and IS requirement analysis)[2] 등이 제시되고 있다. 하지만 아직까지 객체지향 모형화 언어의 표준으로 자리잡고 있는 UML에 근간한 객체지향 업무 모형화 기법은 제시되고 있지 못하다. 본 논문에서는 UML기반 객체지향 업무 모형화 기법으로 UML-B(UML Extension for Business Modeling)를 제시한다. UML-B에서는 UML 표기법의 표준 확장 메카니즘을 준수하여 업무시스템을 표현하는데 필요한 모형화 구조물을 정의하고 있다. UML-B에서는 업무 시스템의 다양한 모형화 관점을 지원하고, 이들 모형화 관점간의 관련성을 보여주기 위한 2차원 행렬 기법을 포함하고 있다.

본 논문의 구성은 다음과 같다. 제2장에서는 UML-B의 개요와 모형화 구조물을 예제를 중심으로 설명한다. 제3장에서는 모형화 연관물간의 상호연관도를 검토하기 위한 행렬 기법을 소개한다. 4장에서는 UML-B를 활용한 업무 모형의 작성 및 재설계 절차를 소개한다. 5장에서는 결론을 제시한다.

2. UML-B(Unified Modeling Language Extension for Business Modeling)

2.1 개요

UML-B는 UML을 활용하면서 체계적인 업무 프로

세스의 모형화와 분석을 수행하기 위하여 개발되었다. UML-B가 가지는 특징은 다음과 같다.

2.1.1 UML 표기법 표준의 준수

UML-B는 UML 표준 표기법을 준수하고, UML에서 제시하는 확장 메카니즘을 충분히 활용하여 업무 시스템을 모형화하도록 하였다. UML이 제공하는 확장 메카니즘은 3가지로 (1) 태그-값 집합(tag-value set), (2) 스테레오타입, (3) 제약식(constraint) 등이다. UML에 기반한 업무 모형화는 업무 시스템에 대한 분석 작업 이후의 정보시스템 개발 작업 시에 업무 분석 산출물들이 시스템 개발자들에게 손쉽게 이해되고 활용될 수 있도록 한다.

2.1.2 BPR(Business Process Reengineering)의 프로세스 관점의 지원

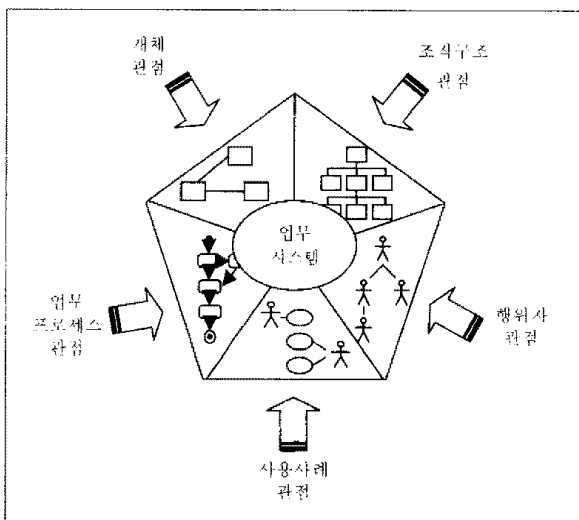
BPR은 업무 시스템을 기능(function) 단위로 보는 것이 아니라, 고객의 요구로부터 시작되고 여러 행위자를 거쳐 수행되는 프로세스 관점에서 바라보는 시각을 제공한다. UML에서도 BPR에 활용을 위해서 UML 버전 1.0부터 활동도(activity diagram)를 제공하고 있다. UML이 제공하고 있는 책임구분선(swimlanes)이 있는 활동도는 업무 프로세스의 절차적인 측면의 표현과 단위 업무들의 책임 주체를 명확히 함으로써, 프로세스 개선을 위한 다이어그램으로 유용하게 사용될 수 있다. 하지만, 프로세스 소요시간, 재작업을 등과 같은 업무 프로세스 분석을 위한 정량적 또는 정성적 척도(measure)를 표현하고 분석하도록 지원하지는 못한다. UML-B에서는 활동도를 중심으로 UML의 확장 메카니즘인 태그값과 스테레오타입을 이용하여, 업무 프로세스의 모형화를 지원하고, 업무 프로세스 관련 척도들도 함께 표현하고 분석할 수 있도록 하였다.

2.1.3 업무 시스템에 대한 다양한 관점의 제공

업무 시스템의 복잡성으로 인해서, 업무 시스템의 모형화는 일반적으로 다관점 모형화 접근법을 따른다. UML-B에서도 업무 시스템의 다양한 관점을 표현할 수 있는 5 관점, 조직구조 관점, 행위자 관점, 사용사례 관점, 업무 프로세스 관점, 개체 관점을 지원한다. 이와 함께 UML-B에서는 이들 5관점간의 상호관계의 이해를 위한 행렬 기법을 포함한다. 정보공학에서는 2차원 행렬을 이용하여 자료, 기능, 위치 등 시스템의 다

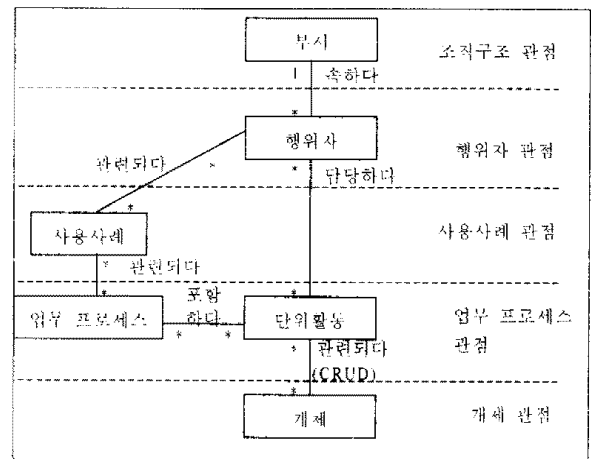
관점들간의 상호관련성은 분석하였다[12]. 유사하게 Zachman의 'Framework for Information Systems Architecture'에서는 업무 관점(business view)의 기능과 자료간의 관계를 CRUD-행렬(Create, Read, Update, and Delete Matrix)을 사용하여 표현하고 있다[17]. UML-B에서는 이러한 2차원 행렬 기법을 사용하여 표현된 업무 시스템 관점간의 연관성을 체계적으로 이해할 수 있도록 지원한다.

UML-B는 (그림 1)과 같이 업무시스템을 (1) 조직 구조 관점, (2) 행위자 관점, (3) 사용사례 관점, (4) 업무 프로세스 관점, (5) 개체 관점의 5개 관점에서 표현하고 분석한다. 조직 구조 관점은 조직의 공식적 구조를 나타낸다. 행위자 관점은 업무 시스템 내부의 행위



(그림 1) 업무 시스템의 5 관점

자와 고객을 포함하며, 행위자간에서 계층적 형태로 표현한다. 사용사례 관점에서는 업무 시스템의 기능(본 논문에서는 사용사례라 부르기로 함)을 표현한다. 업무 프로세스 관점은 개선의 대상이 되는 업무 프로세스를 업무 프로세스에 포함된 단위 업무, 절차, 담당자를 중심으로 표현한다. 개체 관점은 업무 시스템에서 생성, 참조, 갱신, 삭제되는 문서를 비롯한 수동적인(passive) 객체(본 논문에서는 개체(entity)라고 부르기로 함)들에 대한 내용들을 표현한다. UML-B의 업무 시스템에 대한 보다 상세한 메타 모형은 다음 (그림 2)와 같다.



(그림 2) 업무 시스템의 메타 모형

UML-B는 UML의 사용사례도, 클래스도, 활동도를 중심으로 업무 시스템을 표현한다. UML-B는 UML의 확장 매카니즘을 적용하여 개발되었는데 확장 내용에 대한 요약은 다음 <표 1>, <표 2>와 같다.

<표 1> UML-B의 스테레오타입 확장 내용

메타모형 클래스	스테레오타입 명	설 명
Class	부서(department)	조직의 부서 객체
Class	직원(worker)	업무시스템 내부에 존재하는 직원
Class	개체(entity)	수동적인 클래스로 업무 프로세스에서 행위자에 의해서 다루어지는 객체
Class	문서(form)	개체의 일종으로 업무 프로세스 중 생성, 참조, 갱신, 삭제되는 정보들의 집합
Class	장비(equipment)	개체의 일종으로 자재, 제품을 처리하고 가공하기 위한 설비
Class	재료(material)	개체의 일종으로 업무시스템 내에서의 생산, 가공을 위한 원자재와 제품
Association	상위부서(dominate)	부서와 부서간의 상하관계
State	단위업무(business activity)	업무 프로세스를 구성하는 업무 활동
Transition	업무선후관계(precedence)	업무 활동간의 선후관계

* 첫번째 열의 "메타모형 클래스"는 UML 메타모형의 클래스 임[18].

<표 2> UML-B의 태그값 확장 내용

메타모형 클래스	태그 명	설 명
State(단위업무 스테레오타입의 경우)	처리시간(processing time)	단위업무를 처리하는데 걸리는 평균 시간
State(단위업무 스테레오타입의 경우)	입력물(input)	단위업무에서 처리하는 입력물이 되는 개체
State(단위업무 스테레오타입의 경우)	출력물(output)	단위업무에서 처리결과로 생성되거나 갱신되는 개체
Transition(업무선후관계 스테레오타입의 경우)	대기시간(waiting time)	단위업무 처리 후 다음 단위업무 처리까지의 평균 대기 시간
Transition(업무선후관계 스테레오타입의 경우)	분기확률(branch rate)	Branch 유형의 PseudoState 메타클래스를 입력 노드(source vertex)로 가지는 경우에 사용되며, 분기에 대한 발생 비율
Transition(업무선후관계 스테레오타입의 경우)	발생빈도(frequency)	Initial 유형의 PseudoState 메타클래스를 입력 노드(source vertex)로 가지는 경우에 사용되며, 업무 프로세스의 발생 빈도를 나타냄

2.2 업무 시스템의 표현

2.2.1 행위자 관점

설명을 위해서 전형적인 대학 도서관의 업무 시스템을 예제로 사용하기로 한다. 해당 대학 도서관은 도서의 대출 업무, 수서 업무(도서의 구매), 편목 업무(입수한 도서의 목록 작업) 등을 수행한다. 대출 업무와 관련해서는 도서의 대출, 반납, 분실 시 처리, 반납 연기 시의 처리, 파손 시의 처리 등이 있다.

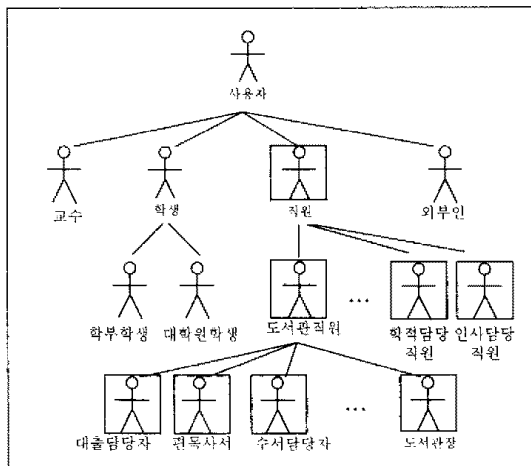
행위자는 해당 업무 시스템의 서비스를 받는 고객과 서비스를 제공하기 위한 시스템 내의 행위자들로 구분된다. 엄밀한 의미에서, 서비스를 제공하기 위한 시스템 내부의 행위자는 행위자라기보다는 객체로 구분하는 것이 더 정확하다. 하지만, 업무 시스템의 분석을 위해서는 외부의 행위자뿐만 아니라, 내부에서 서비스를 제공하는 업무담당자의 표현과 분석 또한 중요하므로, 행위자로 다루기로 한다. 서비스를 제공하는 내부

행위자는 행위자 표시에 박스를 둘러서 외부의 행위자와 구분하도록 한다. UML-B에서는 행위자 관점이 행위자들의 계층도를 통해서 표현되며, 행위자 계층도는 행위자들간의 일반화(generalization) 관계를 표현한다. 예제 도서관 시스템의 행위자 계층도는 (그림 3)과 같다. 행위자 계층도와 함께 행위자의 역할에 대한 간략한 설명과 실제 행위자들을 나열하는 행위자 기술서가 함께 작성된다. 행위자 기술서는 행위자명, 행위자 설명, 실제 행위자 명단의 세 개의 열(column)들로 구성된 테이블 형태를 갖는다.

2.2.2 조직 구조 관점

조직 구조 관점은 해당 조직의 공식적인 구조를 표현한다. 조직의 공식적 구조는 전통적인 계층구조, 팀 구조, 매트릭스구조 등 다양한 형태를 가질 수 있다. 조직구조에 대한 이해는 업무 개선 시에 권한의 이양, 업무 이관, 업무의 분산화 등의 의사결정을 위해서 필요하며, 이 경우 조직 구조에 대한 변경이나 자원의 재할당, 업무 분장의 조정 등이 수반된다. 조직 구조는 UML의 클래스도로 표현되는데, "부서" 스테레오타입의 갖는 객체 클래스 및 인스턴스들과 "상위부서(dominate)" 스테레오타입의 연관성(association)만을 나타낸 형태의 클래스도이다.

예제 시스템의 경우, 도서관 이외의 부서 조직에 대하여는 관심이 없으므로 상세하게 표현하지 않고, (그림 4)와 같이 단과대학, 학과, 행정부서로 축약하여 클래스 수준으로 표현하였다. 단과대학, 학과, 행정부서는 "부서" 스테레오타입을 갖는 클래스이며, 총장, 정보본부, 도서관, 정보관리실, 정보조사실은 객체 인스턴

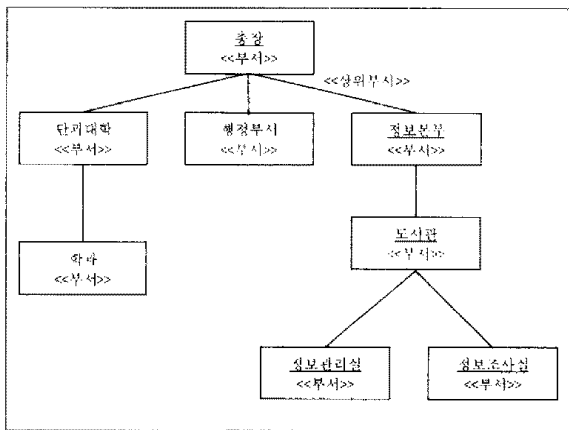


(그림 3) 행위자 계층도

스로 표현되었다.

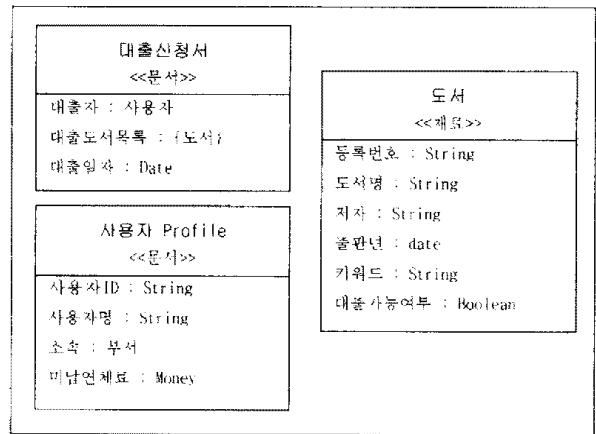
2.2.3 개체 관점

개체는 업무 시스템에서 다루어지는 수동적인 객체들로, 업무 시스템의 결과물, 입력물, 중간 생성물들을 모두 포함한다. 업무 시스템 모형화를 위한 개체의 유용한 3개 스트레오타입으로 문서(form), 장비(equipment), 재료(material)를 정의한다. 문서(form)는 특히 업무 시스템 내에서 정보를 전달하는 주요한 매개체가 된다. 여기서 문서는 정형적 또는 비정형적인 서류 양식뿐만 아니라 컴퓨터 입력 화면, 컴퓨터 출력 보고서를 포함한다. 개체 관점은 개체의 구성항목을 중심으로 클래스도 형태로 표현된다. (그림 5)는 예제 도서관 시스템의 개체 클래스의 예를 보여주고 있다.



(그림 4) 조직 관점을 표현하는 클래스도

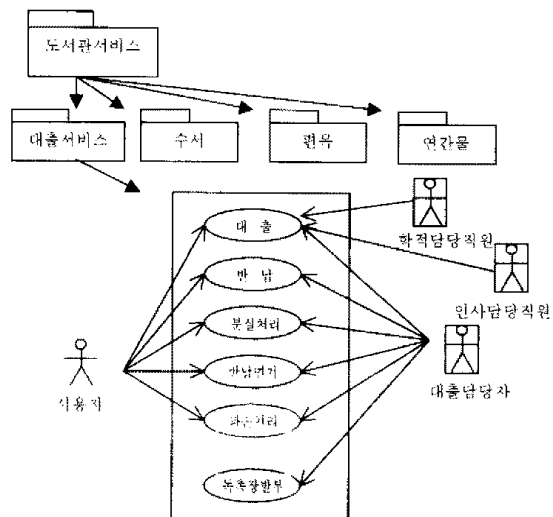
각례들 중에서 BPR 관점에서의 업무 분석을 수행한 주요 사용사례들을 추출한다. BPR에서 일반적으로 사용사례의 추출은 80:20 규칙에 근거한다. 즉, 업무 시스템의 기능 중에서 20 %에 해당하는 기능이 전체 비용의 80 %를 차지한다는 규칙이다. 그러므로, 주요한 기능만을 중점적으로 분석하고 개선함으로써 개선 활동의 집중력을 높이고 좋은 개선 효과를 얻고자 하는 것이다. 사용사례의 선정에는 고객만족도, 핵심 기능 여부, 가치 창조 기능 여부 등 여러 가지를 고려할 수 있다[7, 9].



(그림 5) 개체 관점을 표현하는 클래스도

2.2.4 사용사례 관점

사용사례 관점은 업무 시스템의 기능성을 나열한다. 업무 시스템이 제공하는 기능 또는 서비스들이 하나의 사용사례로 표현된다. 업무시스템을 이해하기 위해서는 업무를 단순히 나열하는 것만으로는 부족하고 이를 전체적으로 이해할 수 있도록 묶어주는 것이 필요하다. UML-B에서는 사용사례들의 그룹핑을 UML에서 제공하는 패키지 개념을 사용하여 수행하도록 한다. 즉, 업무 시스템의 모든 서비스나 기능들을 하나의 패키지로 정의하고, 이들 패키지를 하향식으로 분화하여 최종적으로는 단위 서비스들이 나열되도록 한다.

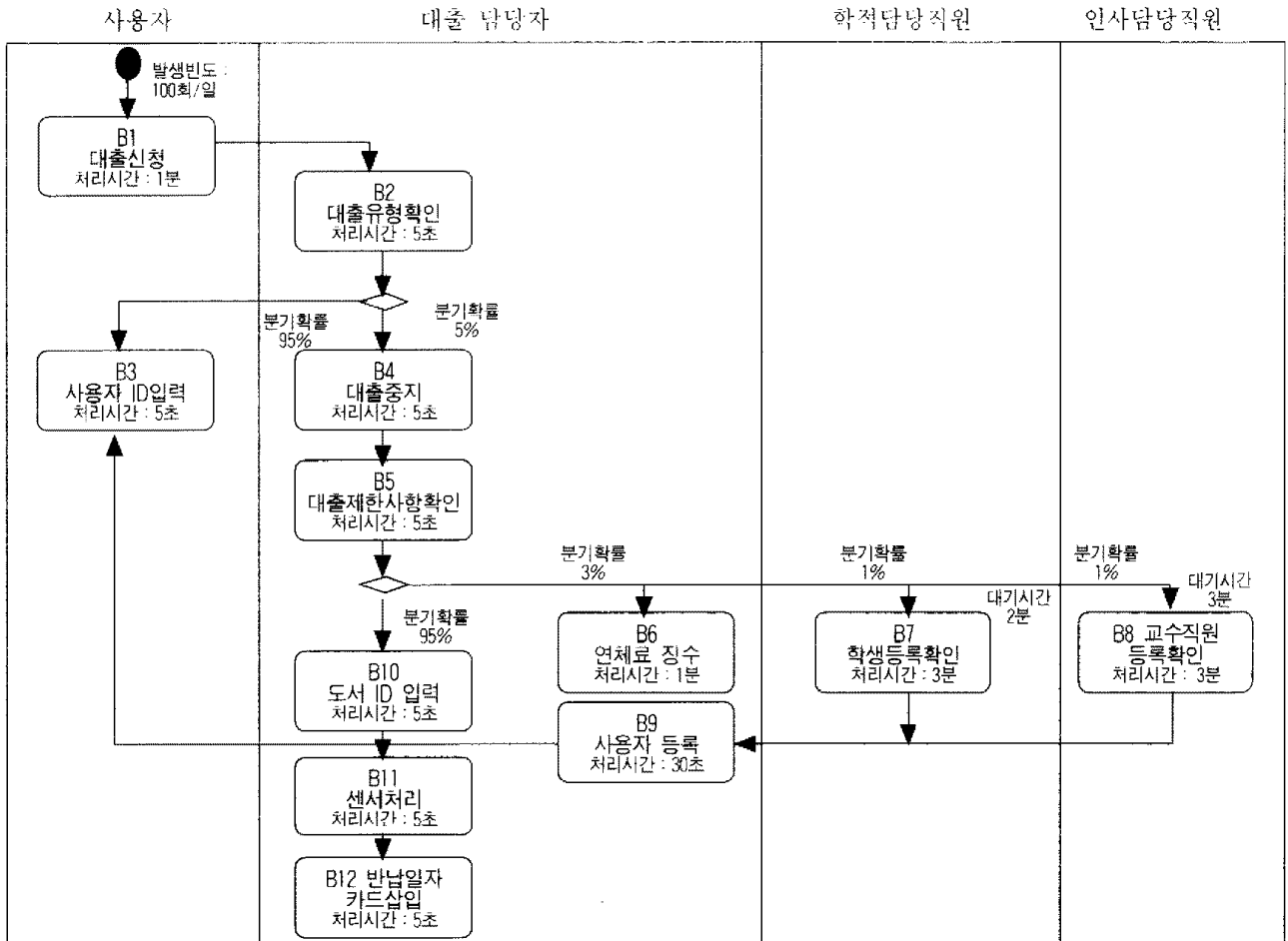


(그림 6) 사용사례 관점을 표현하는 사용사례도

2.2.5 업무 프로세스 관점

업무 프로세스 관점은 업무 프로세스에 대한 가시화를 제공하는 관점이다. 사용사례 관점에 표현된 사용

선정된 사용사례에 대하여는 프로세스(흐름) 분석 작업을 수행하게 된다. 프로세스 분석 작업은 책임구분선(swimlanes)이 포함된 활동도를 중심으로 수행하게



(그림 7) 대출 사용사례의 활동도

된다. 프로세스의 개선을 위한 척도들을 도출하기 위해서 활동도에 단위업무의 처리시간, 입력물, 출력물, 간선의 대기시간, 분기 노드의 분기확률, 초기 노드의 발생빈도 등의 태그값을 추가하여 표현한다. (그림 7)은 대출 사용사례에 대한 활동도이며, 각 단위업무의 입력물과 출력물 태그값은 지면 상의 복잡성으로 인해서 생략하였다.

3. 상호연관도 행렬

2장에서 제시된 UML-B의 모형화 구조물들은 업무 시스템에 대한 5개의 다른 관점을 제공한다. 하지만 이러한 5개의 관점은 서로 독립적인 관점은 아니고 상호 관련성을 갖는 관점들이다. UML-B에서는 이러한 5개의 관점간의 관련성을 2차원 행렬을 통해서 표현하고 관리한다. 작성되거나 자동적으로 생성되는 행렬들은 다음 <표 3>과 같다.

3.1 AO 행렬

AO 행렬은 행위자와 조직구조간의 관계를 보여주는 행렬이다. 이는 조직 관점을 표현하는 클래스도와 행위자 계층도를 바탕으로 업무 분석가에 의해서 작성된다. <표 4>(a)와 같이 AO행렬의 열에는 부서기, 행에는 행위자가 나열된다. 행렬의 원소 값은 다음과 같이 정의된다.

$$AO_{ij} = \begin{cases} 1 & \text{행위자 } i \text{가 부서 } j \text{에 속한 경우} \\ 0 & \text{otherwise} \end{cases}$$

AO 행렬은 행위자 계층도와 조직 관점의 클래스도의 말단 노드에 해당하는 행위자와 부서만으로 작성이 된다. 행위자 계층도의 모든 행위자와 상위 부서를 포함하는 확장된 AO행렬(AO')은 다음과 같이 AO행렬로부터 생성된다.

〈표 3〉 상호연관도 행렬

행렬명	관련 관점	주요 내용	구분
AO 행렬	행위자-조직구조	행위자가 어떤 부서에 속했는지를 표현	분석가 작성
PU 행렬	업무프로세스-사용사례	활동도가 어떤 사용사례에 대한 업무 프로세스를 표현하는지 나타냄	분석가 작성
AP 행렬	행위자-업무프로세스	어떤 행위자와 어떤 업무프로세스에 참여하는지 표현	활동도에서 자동생성
AU 행렬	행위자-사용사례	행위자와 사용사례간의 관계	사용사례도에서 자동생성
AB 행렬	행위자-단위업무	행위자가 업무프로세스 내의 어떤 단위업무를 담당하는지 표현	활동도에서 자동생성
EB 행렬	개체-단위업무	개체가 업무프로세스 내의 어떤 단위업무에서 생성, 참조, 갱신, 삭제되는지 표현	활동도에서 자동생성
OC 행렬	조직구조-사용사례	어떤 부서가 어떤 사용사례와 관련되는지 표현	AO, AU 행렬로부터 자동생성
OP 행렬	조직구조-업무프로세스	어떤 부서가 어떤 업무 프로세스에 관련되어있는지 표현	AO, AP 행렬로부터 자동 생성

$$AO'_{ij} = \begin{cases} 1 & \text{(i) 행위자 } i, \text{부서 } j \text{가 말단 노드인 경우,} \\ & \text{행위자 } i \text{가 부서 } j \text{에 속한 경우} \\ & \text{(ii) 행위자 } i \text{가 말단 노드가 아니고,} \\ & \text{부서 } j \text{가 말단노드인 경우,} \\ & \text{부서 } j \text{에 속한 행위자가 존재하는 경우} \\ & \text{(iii) 행위자 } i \text{가 말단 노드이고} \\ & \text{부서 } j \text{가 말단노드가 아닌 경우,} \\ & \text{행위자 } i \text{가 부서 } j \text{의 하위 부서에 속한 경우} \\ & \text{(iv) 행위자 } i, \text{부서 } j \text{가 말단 노드가 아닌 경우,} \\ & \text{행위자 } i \text{의 하위 행위자 중 부서 } j \text{의} \\ & \text{하위 부서에 속한 행위자가 존재하는 경우} \\ 0 & \text{otherwise} \end{cases}$$

3.2 PU 행렬

PU 행렬은 업무 프로세스와 사용사례간의 관계를 보여준다. 일반적으로 하나의 업무 프로세스를 나타내는 활동도는 하나의 사용사례에 대하여 작성된다. 하지만 프로세스 분석이 필요없는 사용사례에 대해서는 활동도를 작성하지 않는다. 또한, 사용사례들을 연결하는 업무 프로세스의 분석이 필요한 경우, 여러 사용사례를 포함하는 업무 프로세스에 대한 활동도를 작성할 수 있다. 예를 들어, 도서관 예제에서 대출과 반납 사용사례를 포함하는 대출/반납 업무 프로세스에 대한 활동도를 작성할 수도 있다. PU행렬은 다음과 같이 정의된다.

$$PU_{ij} = \begin{cases} 1 & \text{업무프로세스 } i \text{가 사용사례 } j \text{에} \\ & \text{관련된 업무 프로세스인 경우,} \\ 0 & \text{otherwise} \end{cases}$$

3.3 AP 행렬

AP 행렬은 행위자와 업무 프로세스의 관계를 보여주는 행렬이다. AP 행렬은 활동도들로부터 자동적으로 생성되며, 다음과 같이 정의된다.

$$AP_{ij} = \begin{cases} 1 & \text{행위자 } i \text{가 업무 프로세스 } j \text{에} \\ & \text{참여하는 경우,} \\ 0 & \text{otherwise} \end{cases}$$

3.4 AU 행렬

AU 행렬은 행위자와 사용사례간의 관계를 보여주는 행렬이다. AU 행렬은 사용사례 관점의 사용사례도로부터 자동적으로 생성이 되며, 다음과 같이 정의된다.

$$AU_{ij} = \begin{cases} 1 & \text{행위자 } i \text{가 업무 사용사례 } j \text{에} \\ & \text{관련된 경우,} \\ 0 & \text{otherwise} \end{cases}$$

3.5 AB 행렬

AB 행렬은 업무 프로세스 내에서 행위자와 단위업무간의 관련성을 보여주는 행렬이다. AB 행렬은 업무 프로세스에 대한 활동도별로 작성되며 다음과 같이 정의 된다.

$$AB_{ij} = \begin{cases} 1 & \text{행위자 } i \text{가 단위업무 } j \text{의 주체인 경우} \\ 0 & \text{otherwise} \end{cases}$$

3.6 EB 행렬

EB 행렬은 개체와 업무 프로세스 내의 단위업무간의 관련성을 보여주는 행렬이다. 개체는 업무 프로세스 내의 단위 업무에서 생성, 참조, 갱신, 삭제되는 데이터를 정보공학의 CRUD(Create, Read, Update, Delete) 행렬 형태로 표현한다. EB 행렬은 현행 업무 프로세스와 개선 업무 프로세스간의 관련 입출력물의 변화를 한눈에 볼 수 있도록 한다. EB 행렬은 업무 프로세스마다 별도로 작성되며 다음과 같이 정의된다.

$$EB_{ij} = \begin{cases} C & \text{개체 } i \text{가 단위업무 } j \text{에서 생성되는 경우,} \\ R & \text{개체 } i \text{가 단위업무 } j \text{에서 참조되는 경우,} \\ U & \text{개체 } i \text{가 단위업무 } j \text{에서 갱신되는 경우,} \\ D & \text{개체 } i \text{가 단위업무 } j \text{에서 삭제되는 경우,} \\ 0 & \text{otherwise} \end{cases}$$

3.7 OU 행렬

OU 행렬은 조직구조와 사용사례간의 관계를 표현하는 행렬이다. OU 행렬은 AO 행렬과 AU 행렬로부터 유도된다. OU 행렬은 다음과 같이 정의된다.

$$OU_{ij} = \begin{cases} 1 & \sum_k AO_{ki} AU_{kj} > 0 \text{인 경우} \\ 0 & \text{otherwise} \end{cases}$$

3.8 OP 행렬

OP 행렬은 부서와 업무 프로세스와의 관련성을 표현하는데 활용된다. 업무재설계와 함께 조직 구조를 개편하는 경우, 업무 프로세스와 조직 구조간의 관계의 변화를 OP 행렬을 통해서 가시적으로 볼 수 있다. OP 행렬은 AO 행렬과 AP 행렬로부터 유도되며, 다음과 같이 정의된다.

$$OP_{ij} = \begin{cases} 1 & \sum_k AO_{ki} AP_{kj} > 0 \text{인 경우} \\ 0 & \text{otherwise} \end{cases}$$

<표 4> 상호연관도 행렬의 예

(a) 예제 시스템의 AO 행렬

행위자 \ 부서	정보관리실	정보조사실	학과	행정부서
대출담당자	1	0	0	0	
편목사서	1	0	0	0	
수서담당자	0	1	0	0	
교수	0	0	1	0	
학부학생	0	0	1	0	
대학원학생	0	0	1	0	
.....					

(b) 예제 시스템의 AO' 행렬

행위자 \ 부서	정보관리실	정보조사실	학과	행정부서	도서관	정보본부	단과대학
대출담당자	1	0	0	0	1	1	0	
편목사서	1	0	0	0	1	1	0	
수서담당자	0	1	0	0	1	1	0	
도서관직원	1	1	0	0	1	1	0	
교수	0	0	1	0	0	0	1	
학부학생	0	0	1	0	0	0	1	
대학원학생	0	0	1	0	0	0	1	
학생	0	0	1	0	0	0	1	
.....								

(c) 예제 시스템 중 대출 사용사례 활동도에 대한 AB 행렬

행위자 \ 단위업무	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12
사용자	1	0	1	0	0	0	0	0	0	0	0	0
대출담당자	0	1	0	1	1	1	0	0	1	1	1	1
학적과	0	0	0	0	0	0	1	0	0	0	0	0
인사과	0	0	0	0	0	0	0	1	0	0	0	0

앞에서 정의된 8개의 행렬은 현행 업무 시스템과 개선 업무 시스템의 변화를 가시적으로 볼 수 있도록 하며, 또한 업무 시스템의 다른 관점간의 일관성을 검토할 수 있도록 도와준다. 예를 들어, AP 행렬과 AU 행렬은 행위자가 사용사례 관점과 업무 프로세스 관점에서 일관되게 기술되었는지를 검토할 수 있도록 한다. 즉, 임의의 행위자가 사용사례도에는 존재하나, 관련된 활동도에는 존재하지 않는지, 또는 그 반대의 경우는 없는지를 검토하는데 사용될 수 있다. 검토하는 알고리즘은 다음과 같다.

[STEP 1] APU 행렬을 AP, PU 행렬을 이용하여 다음과 같이 생성한다.

$$APU_{ij} = \begin{cases} 1 & \sum_k AP_{ik} PU_{kj} > 0 \text{인 경우} \\ 0 & \text{otherwise} \end{cases}$$

[STEP 2] APU 행렬과 AU 행렬이 동일한지 검토한다. 만일, $APU_{ij} = 1$ 이고 $AU_{ij} = 0$ 이면, 행위자 i 가 사용사례 j 와 관련된 업무 프로세스에는 기술되어 있으나, 사용사례도에는 누락된 경우이고, 반대로, $APU_{ij} = 0$ 이고 $AU_{ij} = 1$ 이면, 행위자 i 가 사용

사례 /)는 사용사례도에서 권위를 갖는다. 사용사례 /)와 관련된 업무 프로세스의 활동도에서는 사용되지 않은 경우이다.

4. UML-B를 이용한 업무 프로세스 모형화 절차

2장에서 제시한 UML-B를 활용하여 업무 프로세스를 모형화하고 분석하는 절차는 다음 (그림 8)과 같다.

4.1 행위자 규명

업무 시스템의 고객, 협력업체 등의 외부 행위자와 업무 시스템 내부의 행위자를 나열한다. 행위자의 업무 시스템에 대한 요구사항, 불만사항 등을 조사하기 위해서 면담 등의 행위자 분석 작업을 수행한다.

4.2 조직 구조 분석

해당 업무 조직의 공식적인 조직 구조를 검토한다. 조직 구조에 대한 문제점, 개선 방향도 함께 검토한다. 조직 구조는 클래스도 형태로 표현한다.

4.3 개체 모형화

업무 시스템에서 다루어지는 문서, 재료, 장비 등을 기초로 하여 개체들을 규명하고 모형화한다. 개체들과 관련된 행위자, 업무, 자료 항목을 중심으로 객체들을 검토하고, 클래스도 형태로 개체와 개체간의 관계를 표현한다.

4.4 사용사례 모형화

업무 시스템 내의 사용사례들을 나열하고, 사용사례도 형태로 사용사례들과 행위자간의 관계를 정의한다. 사용사례간의 "use"나 "extend" 관계를 정의한다. 각 사용사례에 대한 서술적 설명과 사건흐름(Flow of Event)을 사용사례기술서에 작성한다.

4.5 업무분석 대상 프로세스 선정

조직에는 많은 업무 프로세스가 존재한다. 이러한 프로세스들을 동시에 리엔지니어링 할 수는 없으므로 소수의 중요한 프로세스를 선택하여 리엔지니어링을 수행한다. 동시에 너무 많은 프로세스를 대상으로 하면 초점과 경영자층의 관심이 흩어진다. 따라서 다음과 같은 평가기준으로 업무분석 대상 프로세스를 선정한다.

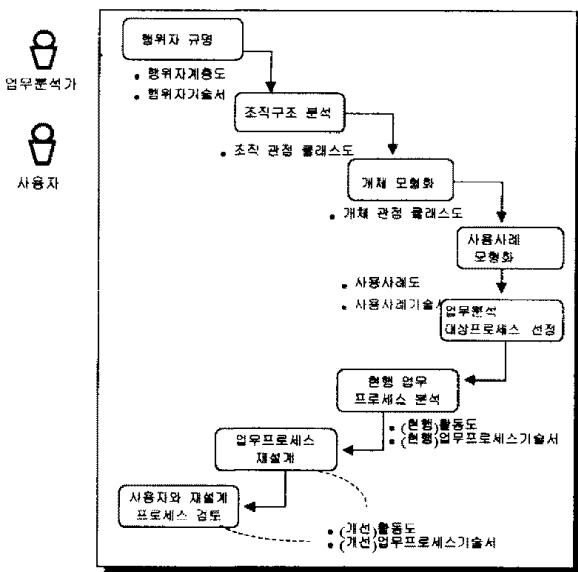
- ① 전략적으로 중요한 프로세스를 선택한다.
- ② 고객에게 영향을 미치는 프로세스를 선택한다.
- ③ 종업원들의 불만이 많은 프로세스를 선택한다.
- ④ 개선하기 쉬운 프로세스를 선택한다.
- ⑤ 가용자원의 확보가 용이한 프로세스를 선택한다.

4.6 현행 업무 프로세스 분석

선정된 프로세스들의 현재 작업방식을 분석하고 주요 프로세스 평가 척도를 측정한다. 경영혁신의 기초가 되는 현상태는 왜곡없이 정확한 이해가 이루어져야 한다. 특히 작업 방식의 파악을 기초로 프로세스 각 단위업무별로 입력물, 출력물, 문제점들의 파악이 필요하다. 예외사항의 해결방식, 고객의 피드백에 대한 처리과정도 분석되어야 한다. 활동도와 업무 프로세스기술서를 중심으로 업무 프로세스에 대한 이해 및 분석 작업을 수행한다. 활동도는 책임구분선을 포함하는 형태를 사용하여 활동의 주체를 표현한다. 업무 프로세스기술서는 업무 프로세스에 대한 개요적 설명과 함께 프로세스 내에 포함된 단위업무에 대한 상세한 정보(설명, 입력물, 출력물, 처리시간, 담당자, 문제점 및 개선방안)를 포함한다.

4.7 업무 프로세스 재설계

업무 프로세스의 재설계가 필요한 경우, 업무 프로세스 개선안을 작성한다. 개선모형은 활동도 형태로 작성되고 상세한 개선 내용은 (개선)업무프로세스기술서에 정리한다. 개선 모형을 작성하기 위한 절차는 다



(그림 8) 업무 프로세스 모형화 절차

음과 같다.

1. 해당 업무 프로세스 개선을 위해서 활용 가능한 정보기술을 검토한다.
2. 고객 입장에서 새로운 프로세스의 개발을 시도한다.
3. 기존 업무 프로세스는 병렬작업과 업무의 통폐합을 이용하여 프로세스를 재설계한다.
4. 개선 업무 프로세스에 대한 활동도를 작성한다.
5. 개선 업무 프로세스의 개선 효과를 프로세스 평가 척도를 기준으로 예측한다.
6. 개선 업무 프로세스의 타당성을 검토한다.

3.8 사용자와 재설계 프로세스 검토

재설계된 업무 프로세스를 사용자와 검토하고 사용자의 의견을 수렴하여 보완한다. 필요하다면 실제 구축에 앞서서 개선 프로세스를 시범적으로 적용한다.

- ① 사용자와 개선 업무 프로세스를 검토한다.
- ② 사용자의 의견을 반영하여 개선 업무 프로세스를 보완한다.
- ③ 필요한 경우 업무 프로세스의 실제 구축에 앞서 초기 실험을 수행한다.

4. 결 론

정보시스템 개발의 사전 작업으로 수행되는 업무 개선 작업을 위해서는 업무 모형화가 필요하다. 이러한 업무 모형화 산출물은 시스템의 사용자, 업무 분석 및 설계자뿐만 아니라, 정보시스템 개발자까지 이해하고 참조할 수 있도록 작성되는 것이 필요하다. 본 논문에서는 객체지향 모형화의 표준으로 자리잡고 있는 UML에 기반하면서, 5개의 관점(행위자, 조직 구조, 개체, 사용사례, 업무 프로세스)을 포괄적으로 수용하는 업무 시스템 모형화를 위한 UML-B(UML for Business Extension)를 제시하였다. UML-B는 UML의 표준 표기법을 준수하여 정보시스템 개발자가 업무 시스템 모형을 이해하기 위해서 추가적인 노력이 필요하지 않도록 하였다. UML-B에서는 UML이 제공하는 확장 메카니즘을 활용하여 업무시스템의 모형화를 효율적으로 수행하고 업무 개선 작업을 수행할 수 있도록 하였다. 현재 UML 1.1에서는 업무시스템을 위한 UML 확장을 제시하고 있으나, 스테레오타입을 추가하는 수준이며, 그조차도 아이디어 수준으로 실제적인 활용을 위해서는 추가적인 연구가 필요한 상태이다. 하지만

UML-B에서는 실제적인 활용이 가능한 수준의 상세화된 업무 모형화 기법을 제공하고 있다. 또한 모형화된 관점들간의 관련성과 일관성을 가시화할 수 있는 행렬 기법을 포함하고 있다. 추후 연구과제로는 (1) UML-B의 실제적인 적용을 통한 보완 작업과 (2) UML-B의 유용성에 대한 정성적, 정량적 평가, (3) UML-B 모형의 정형화(formalization) 작업 (4) 지원도구의 개발 등이 필요하다.

참 고 문 헌

- [1] 강문설, 김태희, "객체지향 소프트웨어 개발 방법론의 표준화: UML(Unified Modeling Language)," 정보처리학회지, 제5권, 제5호, 1998.
- [2] 박성주 외, '클라이언트-서버 정보 시스템을 위한 객체 지향 개발 방법론에 관한 연구', 정보통신부, 1997.
- [3] 조은숙, 김수동, 류성열, "UML을 기반으로 한 실무 중심의 객체지향 방법론," 한국정보처리학회 논문지, 제6권 제3호, 1999.
- [4] Booch, Grady, 'Object-oriented Design with Application,' Benjamin-Cummings, 1991.
- [5] Coad, P. and Yourdon, E., 'Object-Oriented Analysis,' Yourdon Press, Englewood, Cliffs, NJ, 1991.
- [6] Coleman, Derek, et al. 'Object-Oriented Development: The Fusion Method,' Prentice Hall, 1994.
- [7] Davenport, T.H., 'Process Innovation, Reengineering Work through Information Technology,' Harvard Business School Press, 1993.
- [8] Graham, Ian, Henderson-Sellers, Brian, and Younessi, Houman, 'The Open Process Specification,' ACM Press, Harlow, England, 1997.
- [9] Hammer, Michael, "Reengineering Work: Don't Automate, Obliterate!," Harvard Business Review, July, pp.104-112, 1990.
- [10] Jacobson Ivar, 'Object-Oriented Software Engineering: A Use Case Driven Approach,' Addison-Wesley, 1992.
- [11] Jacobson Ivar, 'The Object Advantage: Business Process Reengineering with Object Technology,' ACM Press, 1995.
- [12] Martin, James, 'Information Engineering Book

II: Planning and Analysis,' Prentice Hall, 1991.

[13] Mertins, K., Jochem, R., Jkel, F.W., "A Tool for Object Oriented Modeling and Analysis of Business Processes," Computer in Industry, Vol.33, pp.345-356, 1997.

[14] Rumbaugh James, et al., 'Object-Oriented Modeling and Design,' Prentice Hall, 1991.

[15] Taylor, David A. 'Business Engineering with Object Technology,' Wiley, 1995.

[16] Toussaint, Pieter J., Bakker, Albert R., Groenewegen, Luuk P.J., "Constructing an Enterprise Viewpoint: Evaluation of Four Business Modeling Techniques," Computer Methods and Programs in Biomedicine, Vol.55, pp.11-30, 1998.

[17] Zachman, J.A., "A Framework for Information Systems Architecture," IBM Systems Journal, Vol.26, No.3, 1987.

[18] Rational, 'Unified Modeling Language Version 1.1,' Rational, 1997.



김종우

e-mail : jwkim@stat.chungnam.ac.kr
 1989년 서울대학교 수학과(학사)
 1991년 한국과학기술원 경영학과(공학석사)
 1995년 한국과학기술원 산업경영학과(공학박사)

1995년~1996년 한국과학기술원 연수연구원(Post Doc.)
 1996년~현재 충남대학교 통계학과 조교수
 관심분야 : 객체지향개발방법론, 전자상거래, 의사결정 지원시스템



김진삼

e-mail : jinsam@etri.re.kr
 1984년 중앙대학교 전자계산학과(학사)
 1986년 중앙대학교 대학원 전자계산학과(석사)
 1987년~현재 한국전자통신연구원 선임연구원

관심분야 : 소프트웨어 공학, 프로그래밍 언어



조진희

e-mail : chojh@etri.re.kr
 1992년 경북대학교 컴퓨터공학과(학사)
 1996년 경북대학교 컴퓨터공학과(석사)
 1995년~현재 한국전자통신연구원 연구원

관심분야 : 소프트웨어 공학, 객체지향 모델링



전진욱

e-mail : jojeon@etri.re.kr
 1984년 한국외국어대학교 교육학부 독일어(학사)
 1987년 미국 조지아 주립대학원 정보시스템(석사)
 1995년 한국외국어대학교 경영정보학과(박사)

1986년~1987년 미국 INTEC 연구소(위촉연구원)
 1987년~현재 한국전자통신연구원 실시간컴퓨팅연구부장(책임연구원)

관심분야 : 소프트웨어 공학, 실시간 시스템