

# User Interaction 이 많은 시스템 환경에서의 프로세스 모델링을 지원하기 위한 지식베이스 시스템

김수연, 서의호, 황현석  
포항공과대학교 산업공학과

## A knowledge-based system to support process modeling in a system environment with high user interaction

Su-Yeon Kim, Eui-Ho Suh, Hyun-Seok Hwang

### Abstract

정보 시스템 개발은 크게 계획, 분석, 설계, 구축의 네 단계로 이루어진다. 이중 사용자 요구사항을 파악하는 분석 단계는 시스템 개발 수명주기에 있어 가장 큰 비중을 갖는다. 또한 수명주기의 초기 단계에서 발견되지 못한 결점은 개발이 진행될수록 수정하는데 많은 비용과 노력을 필요로 하게 되어 분석 결과물의 품질은 전체 시스템 품질에 큰 영향을 미치게 된다. 분석 단계의 주요 작업은 데이터 모델링과 프로세스 모델링이다. 이중 데이터 모델링을 위한 지식베이스 시스템 개발에 대한 노력은 기존 연구에서 수행되어 왔으나 프로세스 모델링을 위한 지식베이스 시스템에 대한 연구는 부족하다. 특히 최근 User Interaction 이 많은 시스템이 점점 증가하고 있는 추세에 적합한 프로세스 모델링 방법과 지식베이스에 대한 연구가 필요하다.

이 연구에서는 사용자 상호작용이 많은 시스템 환경에서의 프로세스 모델링을 위한 절차를 제안하고, 제안된 절차를 효과적으로 지원하고 결과물의 품질을 보증하기 위한 지식베이스 시스템을 구축한다. 모델은 다음의 주요 작업들로 구성된다: 이벤트 분석, 프로세스 분석, 이벤트/프로세스 상호작용 분석. 이벤트 분석은 업무에 영향을 주는 이벤트와 그로 인해 수행되어야 하는 업무 절차(Response)를 파악한다. 프로세스 분석은 이벤트 분석과는 독립적으로 수행되며 상위 수준의 업무부터 최하위 수준의 프로세스까지 도출한다. 이벤트/프로세스 상호작용 분석은 이벤트와 프로세스의 분석 결과를 상호 검증하기 위하여 실시된다. 제안된 프로세스 모델링 방법을 지원하기 위한 지식베이스 시스템을 웹 환경에서 구현하였다.

### 개요

정보시스템 개발의 수명주기는 일반적으로 계획, 분석, 설계, 구축 단계로 구성된다. 계획 단계는 기업의 목적, 전략, 주요성공요인 등과 관련되어 있으며, 새로운 사업 기회나 경쟁 우위를 창출

하기 위하여 정보기술을 어떻게 사용할 것인지에 대한 계획을 수립한다. 기업 내의 상위 수준의 업무기능, 데이터, 정보요구가 도출된다. 분석단계는 업무 영역에 대하여 어떤 프로세스가 필요한지, 이들 프로세스들이 어떻게 상호 연관

되는지, 어떤 데이터가 필요한지, 데이터들간의 관계는 어떠한지를 분석한다. 분석 단계에서는 데이터 모델과 프로세스 모델이 생성된다. 실제 단계는 분석 단계에서 파악된 프로세스들이 어떻게 프로시저로 구현되는지와 프로시저가 어떻게 작동하는지와 연관되어 있다. 데이터 모델은 목표 시스템 환경에 맞는 데이터 구조로 변환된다. 이 단계에서는 사용자의 직접적인 참여가 필요하다. 구축 단계에서는 코드 생성기 또는 개발 틀을 이용하여 프로시저를 구현한다.

이들 단계 중 분석 단계는 사용자의 모든 업무 요구사항을 도출하며 시스템 개발에 대한 업무적인 명세서가 완성되므로 가장 큰 비중을 갖는다고 할 수 있다. 또한 이 단계에서 결점을 파악하지 못하면 이후 시스템 개발 단계에서 많은 비용과 노력을 필요로 하게 되므로 분석 단계 결과물의 품질은 최종 시스템의 품질에 결정적인 영향을 미치게 된다.

분석 단계는 초기 업무 분석, 데이터 모델링, 프로세스 모델링, 상호작용 모델링, 현시스템 분석, 분산 분석, 업무모델 확정 등의 작업으로 구성된다 [Kim, 1994]. 이들 작업 중에서 데이터 모델링과 프로세스 모델링이 가장 핵심적인 작업이 된다. 데이터 모델링과 프로세스 모델링은 분석가의 경험과 지식에 상당히 영향을 받는 기법이므로 모델링 전문가의 풍부한 경험과 노하우를 토대로 한 지식베이스 시스템은 작업하는데 많은 도움을 줄 것이다. 데이터 모델링에 관한 지식베이스 시스템은 많이 연구되어 왔으나 프로세스 모델링에 대한 지식베이스 시스템을 구축하려는 시도는 아직 별로 없었다 [Storey, 1993; Curtis, 1992; Harmon, 1993; Lee, 1999]. 특히 최근 User Interaction 이 많은 시스템이 점점 증가하고 있는 추세에 적합한 프로세스 모델링 방법과 지식베이스에 대한 연구가 필요하다.

이 연구에서는 사용자 상호작용이 많은 시스템 환경에 적합한 프로세스 모델링을 위한 절차를

제안하고, 제안된 절차를 효과적으로 지원하고 결과물의 품질을 보증하기 위한 지식베이스 시스템을 구축한다. 프로세스 모델링은 다음의 주요 작업들로 구성된다: 이벤트 분석, 프로세스 분석, 이벤트/프로세스 상호작용 분석. 이벤트 분석은 업무에 영향을 주는 이벤트와 그로 인해 수행되어야 하는 업무 절차(Response)를 파악한다. 프로세스 분석은 이벤트 분석과는 독립적으로 수행되며 상위 수준의 업무부터 최하위 수준의 프로세스까지 도출한다. 이벤트/프로세스 상호작용 분석은 프로세스 분석 결과를 검증하기 위한 것으로 이벤트 대응을 지원하기 위한 프로세스가 모두 도출되었는지와 프로세스 분석에서 도출된 모든 프로세스가 이벤트 대응 목록에 반영되어 있는지 확인한다. 제안된 프로세스 모델링 방법을 지원하기 위한 지식베이스 시스템을 구현하였다. 데이터베이스는 MS SQL-Server 를 사용하였고 유저 인터페이스는 웹 환경에서 구현하였다.

## 프로세스 모델링 프레임워크

성공적인 시스템 개발은 분석의 품질에 매우 의존적이다. 따라서, 실제 업무 요구를 정확하고 완전하며 신속하고 중복되지 않게 모델링할 수 있도록 기존 기법들을 지속적으로 개선할 필요가 있다. 그러한 개선은 프로세스 분석 기법에서 특히 필요하다. 기존 방법론이 데이터와 프로세스 분석을 동시에 수행하라고 되어 있지만, 프로세스 분석보다는 데이터 분석에 대해 훨씬 더 많은 기법이 연구되어 왔다.

최근의 시스템은 점점 사용자 상호작용이 증가하고 있다. 예를 들어, 웹을 이용한 시스템은 쇼핑몰, 인터넷뱅킹 등 업무 트랜잭션이 발생하는 전자상거래 뿐만 아니라 온라인 상담, 토론방, 동호인 커뮤니티 등 다양한 형태로 사용자의 참여를 필요로 하게 된다. 웹 기반의 시스템이 아닌 경우에도 사용자 요청이 빈번하게 발생하는

업무가 증가하고 있다. 금융기관의 경우 대부분의 업무는 사용자 요청에 의해 발생하게 된다. 이러한 시스템 환경에서 프로세스 모델링은 사용자 상호작용을 충분히 고려하여 실시되어야 한다. 즉 사용자에게 의한 외부 이벤트는 충분히 분석되어야 한다. 이벤트 분석에서는 먼저 외부 자극(external stimuli)을 파악하고 사용자가 해야 할 일에 초점을 맞춘다.

이벤트 분석을 함으로써 얻을 수 있는 효과는 다음과 같다. 먼저, 사용자 참여를 개선할 수 있다. 둘째, 기존 시스템 process 와 조직 구조로 인한 편견을 막아준다. 셋째, 프로세스 모델의 정확성과 완전성을 검증할 수 있도록 도와 준다. 넷째, 여러 event response 에서 공유될 수 있는 대상 process 를 찾아낼 수 있으므로, 프로세스의 재사용성을 높일 수 있다.

이 연구에서는 사용자 상호작용이 많은 시스템 환경을 고려하여 이벤트 중심의 프로세스 모델링 방법을 제안한다. 전체적인 작업 체계는 그림 1 과 같다.

사실에 기초한다. 이벤트 분석은 모델링과 무관한 새로운 관점에서 이벤트 및 그에 관련된 대응을 도출하게 되므로 모델링의 보완적인 개념 및 기법이라 할 수 있다. 즉, 데이터와 프로세스 분석을 통해 작성된 모델의 완전성과 정확성을 일상업무의 처리 관점에서 검증하는 기법이다. 프로세스 모델의 완전성과 정확성을 검증하기 위하여 이벤트와 관련된 업무프로세스를 정의하고 이벤트 대응 프로세스간의 데이터 의존성을 분석하며, 프로세스의 선행조건을 분석한다. 이벤트 분석은 다음 세 가지 작업으로 구성된다: 이벤트 도출, 이벤트에 대한 계획된 대응 분석, 이벤트 목록 작성.

#### Event 도출

event는 시스템의 통제 외부에서 발생하는 사건(happening)으로 시스템은 그에 대한 계획된 response를 갖는다. 여기서 시스템이란 용어는 광의로 사용되는데, 업무시스템이나 업무영역을

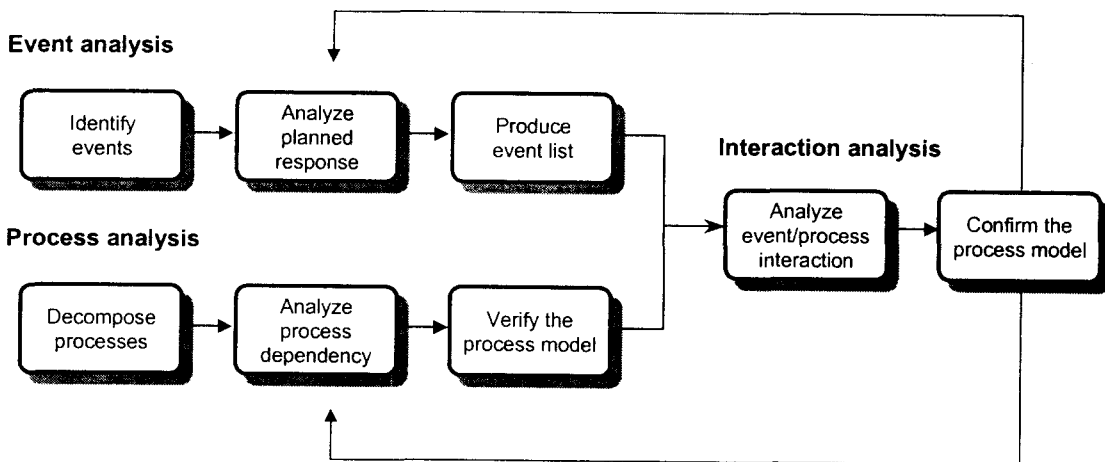


그림 1. 프로세스 모델링의 전체 작업구성도

#### 이벤트 분석

이벤트 분석은 모든 업무활동이 예상되는 상황에 따라 정해진 대응방법이 존재한다는

말한다. ‘고객이 보험을 신규 계약한다’, ‘계약조건을 변경한다’, ‘고객이 보험료를 입금한다’, ‘고객이 손해 청구를 신청한다’ 등은 보험회사의 전형적인 event이다. 이벤트 분석에서는 업무에

영향을 주는 모든 이벤트를 도출한다. 이벤트는 외부 이벤트와 시간 이벤트로 나누어질 수 있다. 외부 이벤트(External event)는 사용자나 외부 시스템이 유발시키는 이벤트이다. 시간 이벤트(Temporal event)는 어떤 업무가 특정 시간이 되면 실행되어야 하는 경우이다. 이벤트 도출 시에는 현업과의 면담, 또는 양식을 이용한 설문을 실시한다.

*Event 에 대한 대응(Planned response) 분석*

도출된 이벤트에 대하여 업무에서 수행해야 할 대응(Response)을 분석한다. 필요한 response는 한번에 하나의 event에 대해 기술한다. 여기서는 업무적인 이벤트만을 대상으로 하며, 시스템 내 부적인 이벤트는 대상에서 제외시킨다. 즉, '고객이 손해 청구를 신청한다'에 대한 response를 분석하는 것이지, '고객 청구는 기록된다' 또는 '결제 금액이 계산된다'에 대한 것을 분석하는 것이 아니다.

*이벤트 목록 작성*

이벤트 분석 결과는 예비(Preliminary) 이벤트 목록, 확장(Augmented) 이벤트 목록, 정련(Refined) 이벤트 목록, 구현(Implementable) 이벤트 목록으로 명세화될 수 있다. 이를 업무활동과 관련하여 단계별로 나타내면 표 1 과 같다.

표 1. 단계별 이벤트 목록

Activity	Event list
Function	Preliminary event list
Process	Augmented event list
Elementary process	Refined event list
Implementable process	Implementable event list

업무 지식이 있는 사용자들이 같이 작업하면 하루나 이틀 만에 비교적 이해하기 쉬운 event 목록을 만들어낼 수 있다. 목록을 관리하게 되면 무엇이 분석되었고 무엇이 남아 있으며 미결 현안들은 무엇인지를 명확하게 이해할 수 있게 된다. 표 2 는 이벤트/대응 목록의 샘플이다.

표 2. 이벤트/대응 목록 샘플

Event	Response	Description
자출납은 보유하고 있는 통화시재를 산상 시재와 맞추어 본다.	- 이미 마감되었는지를 확인한다. - 통화시재를 등록한다. - 등록완료된 텔러의 출납내역집계에 등록내역을 기록한다.	- 텔러마감확인 - 통화시재등록 - 텔러출납집계작성
자출납이 텔러마감을 한다.	- 부점내 미달환이 있는지 확인한다 - 통화시재가 등록되었는지를 확인한다. - 대체거래의 차대가 일치하는지를 확인한다. - 텔러마감 SET 을 한다.	- 미달환대사 - 텔러별대체일치여부확인 - 텔러마감

시스템적 요소를 배제하게 되면 어떻게(how) 해야 하는지보다는 무엇을(what) 해야 하는지에 초점을 맞추게 된다. 이것은 기존 업무 process 와 조직 구조에 치우치는 것을 막아주므로 업무 영역분석에서는 중요하다.

작업규칙

이벤트를 분석할 때는 다음과 같은 규칙이 적용된다.

- 모든 이벤트는 적어도 하나이상의 기본프로세스와 관련된다.

- 모든 기본프로세스는 적어도 하나 이상의 이벤트 대응에 포함된다.
- 이벤트 대응에 포함되는 기본프로세스는 반드시 프로세스계층도에 포함된다.
- 모든 외부오브젝트는 적어도 하나 이상의 기본프로세스와 관련된다.

### 프로세스 분석

프로세스 분석 작업에서는 프로세스 분해 기법과 의존성 분석 기법을 이용하여 업무영역의 프로세스를 모델링한다. 분석 결과는 프로세스계층도(Process hierarchy diagram)와 프로세스의존도(process dependency diagram)로 표현된다 [Martin, 1990]. 프로세스계층도는 기업의 모든 프로세스를 한 눈에 파악할 수 있도록 해 주며, 프로세스의존도는 동일 레벨의 프로세스에 대한 선후 관계를 나타낸다. 프로세스 분해는 의존성 분석을 통하여 검증될 수 있다. 분해가 완료되면 최하위 프로세스인 기본프로세스(Elementary process)의 상세 정보와 입출력 정보를 정의한다. 프로세스 분석 작업의 수행목적은 업무의 전체적인 구조를 정의하고 기본프로세스를 도출하는 것이다.

### 프로세스 분해

기업의 상위 프로세스부터 최하위 프로세스까지 분석한다. 주로 현업과의 면담을 실시하거나 업무 매뉴얼 조사, 또는 기업의 주요자원이나 제품의 업무주기를 파악하여 프로세스를 도출한다. 프로세스계층도는 업무기능, 프로세스, 기본프로세스로 구성되며 이는 레벨에 따라 다음과 같이 구분할 수도 있다. 업무활동(Activity)은 기능, 프로세스, 기본프로세스의 총칭이다. 페어런트(Parent)는 상위수준의 업무활동을 의미하며, 차일드(Child)는 상위 업무활동에 종속된 하위 업무활동, 시블링(Sibling)은 하나의 상위 업무활동을 구성하는 동일 레벨의 하위 업무활동이다

(그림2 참조).

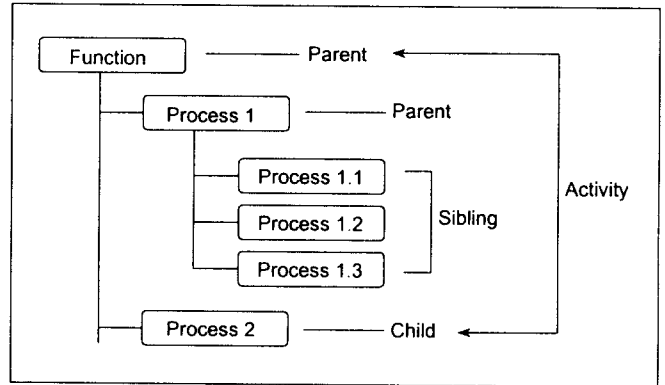


그림 2. 프로세스계층도

프로세스를 분해하는 기본원칙은 응집도(Cohesion)는 높게 결합도(Coupling)는 낮게 분해하는 것이다. 응집도는 시블링간의 관련강도를 의미하며 결합도는 다른 계층의 업무활동과의 관련정도를 의미한다 [Pressman, 1992]. 측정기준은 업무활동간 공유하는 데이터이다. 응집도가 높고 결합도가 낮을수록 분석의 복잡도와 모호성이 감소하게 되어 분석의 집중력이 향상된다. 또한, 프로젝트 관리와 모델 조정이 용이하다. 프로세스계층도 작성시에는 계층의 균형을 유지하여야 한다. 즉, 각 프로세스는 최소한 두 개 이상의 서브프로세스로 분해되어야 하고 7 개를 넘어가서는 안된다. 프로세스가 하나의 프로세스로 분해된다면 동일 레벨의 프로세스가 아닌지 또는 누락된 프로세스가 없는지 확인해 보아야 한다. 또한, 동일 레벨의 프로세스가 7 개를 넘는 경우에는 중간에 한 계층이 빠진 것은 아닌지, 또는 다른 parent의 서브프로세스가 아닌지 확인해 보아야 한다. 모든 프로세스는 최하위 프로세스까지 도출되었는지 확인해 보아야 한다. 더 낮은 레벨로 분해되는 프로세스가 없는지, 프로세스를 너무 많이 분해한 것은 아닌지를 검증해 보아야 한다.

### 작업규칙

프로세스를 분해할 때 적용되는 규칙을 요약하

면 다음과 같다.

- 프로세스는 반드시 2개 이상의 서브프로세스로 분해되어야 한다.
- 한 페어런트의 서브프로세스의 합은 7개를 초과하면 안된다.
- 차일드 프로세스는 페어런트를 완벽하게 설명할 수 있어야 한다.
- 동일한 프로세스가 업무계층 상에 중복으로 존재하면 안된다.
- 분해는 정확하게 기본프로세스까지 진행되어야 한다.

로 인해 사후 프로세스(post-process)의 실행이 가능한 경우 의존성이 성립하게 된다. 의존성을 정의할 때에는 의존성의 이름과 선택성(optionality) 및 cardinality 등의 특성을 기술한다. 의존성의 유형으로는 순차적 의존성 (Sequential dependency), 병렬 의존성 (parallel dependency), 상호배타적 의존성 (mutually exclusive dependency), 재귀적 의존성 (recursive dependency) 등이 있다 [POSDATA, 1994].

그림 3은 프로세스의존도의 예를 보여준다.

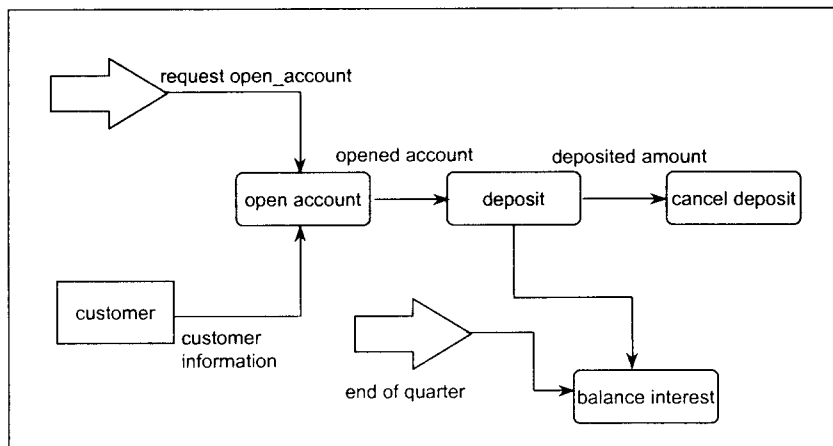


그림 3. 프로세스의존도

### 프로세스의존성 분석

동일 레벨의 프로세스(sibling process)에 대한 프로세스의존도를 작성한다. 프로세스의존도는 프로세스 분해 결과를 검증할 수 있도록 도와 준다. 동일 레벨에 있는 프로세스들은 반드시 의존성을 가져야 한다. 의존성을 갖지 않는 프로세스는 다른 페어런트 밑에 가야 할 프로세스가 아닌지 체크해 보아야 한다. 또는 누락된 프로세스로 인해 의존성이 없는 것은 아닌지 확인하여 누락된 프로세스를 찾아낸다.

프로세스의존도를 구성하는 요소로는 프로세스, 외부오브젝트, 이벤트, 의존성(dependency)이 있다. 의존성은 두 프로세스간에 존재하는 종속관계로서, 선행 프로세스(pre-process)의 실행결과

### 작업규칙

프로세스의존성 분석 시에는 다음과 같은 규칙이 적용되어야 한다.

- 모든 프로세스는 적어도 하나의 의존성을 가져야 한다.
- 동일 레벨에 있는 서브프로세스들은 의존성에 의해 모두 연결되어야 한다.

### 프로세스 모델 검증

프로세스 모델을 확정하기 위하여 모델의 완전성, 정확성, 일관성을 검증하여야 한다.

- 완전성(Completeness): 모든 프로세스가 빠짐없이 도출되었는가
- 정확성(Correctness): 각각의 프로세스는

계층도에서 적절한 위치에 놓여 있는가  
 페어런트가 잘못 매치된 프로세스가 없는지  
 확인한다. 특히 여러 업무에서 공통으로 사  
 용되는 재사용 프로세스의 위치를 정할 때는  
 주의하여야 한다.

- 일관성(Consistency): 상위 프로세스는 하위  
 프로세스들의 합과 정확히 일치하는가  
 프로세스 P의 하위 프로세스가 A, B, C 라면  
 $P = A+B+C$  이어야 한다.  $P > A+B+C$  인 경우  
 는 하위 레벨에서 빠진 프로세스를 찾고, 반  
 대인 경우에는 페어런트가 잘못 매치된 하위  
 프로세스가 없는지 판단한다.

### 상호작용 분석

#### 이벤트/프로세스 상호작용 분석

이 작업에서는 이벤트와 프로세스가 빠짐없이  
 도출되었는지와 정확하게 연결되어 있는지 검증  
 한다. 본 연구에서는 이벤트와 프로세스간의 상  
 호작용을 매트릭스 기법을 이용하여 분석하였다.  
 반드시 매트릭스를 이용할 필요는 없지만 데이  
 터 사전의 교차참조 특성(예를 들어, 이벤트별  
 프로세스 목록 또는 프로세스별 이벤트 목록)을  
 이용하는 것보다는 매트릭스를 이용하는 것이  
 사용자가 검증하기에 편리하다.

이벤트/프로세스 상호작용은 다음과 같은 점에  
 착안하여 검증한다.

- 관련된 프로세스가 하나도 없는 이벤트가  
 존재하지는 않는가

- 이벤트에 대하여 프로세스가 너무 적게(또는  
 많이) 연관되어 있지는 않은가
- 관련된 이벤트가 하나도 없는 프로세스가  
 존재하지는 않는가
- 프로세스에 대하여 이벤트가 너무 많이  
 연관되어 있지는 않은가

#### 프로세스 모델 확정

사용자에 의해 리뷰되고 확정된 모델은 이후 시  
 스템 개발 단계에서 견고한 토대가 된다. 이 작  
 업에서는 사용자와의 워크쓰루를 통하여 프로세  
 스 모델 작업의 모든 산출물 - 이벤트 목록, 이  
 벤트 정의서, 프로세스계층도, 프로세스의존도,  
 기본프로세스 정의서 등 - 을 검토한다. 워크쓰  
 루를 통하여 발견된 문제점을 해결한 후에는 프  
 로세스 모델을 최종적으로 확정한다.

### 시스템 구현

앞에서 제안한 프로세스 모델링 방법을 지원하  
 는 시스템(KBSPRM, Knowledge-Base System for  
 Process Modeling)을 구현하였다. 시스템 설계 및  
 데이터 입력은 모델링 전문가와의 면담을 통하  
 여 진행되었다. KBSPRM은 일반적인 지식베이  
 스 시스템의 아키텍처를 따랐으며 데이터베이스  
 및 지식베이스는 MS SQL-server를 사용하였고,  
 웹 기반의 user interface 하에서 운영된다.  
 KBSPRM의 전체적인 아키텍처는 그림 4와 같  
 다.

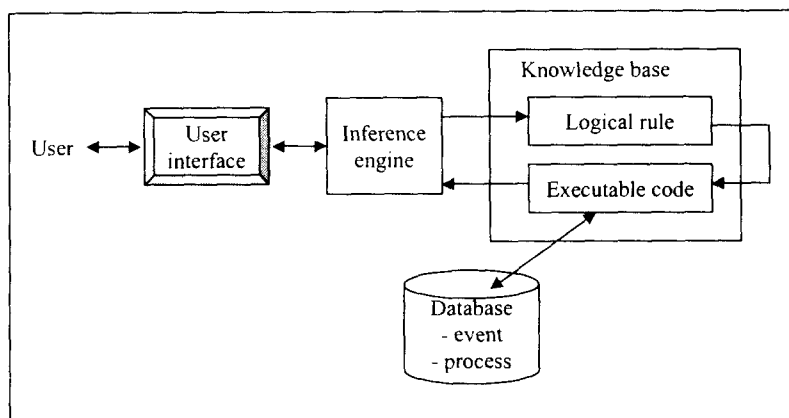


그림 4. KBSPRM의 전체 아키텍처

KBSPRM 은 모델링 과정에서 발생하는 모든 데이터를 저장하는 데이터베이스와 모델링 시 지켜져야 하는 규칙들을 저장하는 지식베이스, 시스템에서 특정 상황이 발생하였을 때 해당 규칙을 찾아 실행시키는 추론엔진, 그리고 사용자가 입력한 데이터를 이용하여 다이어그램을 보여 주거나 모델에 대한 검증 결과 보고서를 출력해주는 사용자 인터페이스로 구성된다. 지식베이스는 모델링 규칙을 의사코드 형태로 표현하는 logical rule base 와 실행 가능한 프로그램을 저장하는 executable rule base 로 구성된다. 추론엔진이 지식베이스에서 해당 logical rule 을 찾으면 그 rule 이 실행 코드를 찾아 관련 프로그램을 실행하게 된다.

### 시스템의 주요 기능

시스템의 기본적인 메뉴는 계획, 분석, 설계, 구축으로 구성되며, 분석 단계에서는 데이터 모델링, 프로세스 모델링, 상호작용 모델링의 서브메뉴가 제공된다. 프로세스 모델링은 이벤트 분석, 프로세스 분석, 이벤트/프로세스 상호작용 분석 작업으로 구성된다. 이 연구에서는 프로세스 모델링 파트만 시스템으로 구현하였다.

- 이벤트 분석: 이벤트 정의와 이벤트 대응 목록(관련 프로세스 연결), 일관성 체크 등의 기능이 제공된다. 사용자가 이벤트를 정의하면 시스템은 이벤트를 목록 형태로 보여 준다.
- 프로세스 분석: 프로세스 정의, 프로세스계층도, 프로세스의존도, 프로세스상세 프로파일 작성(관련 이벤트 연결), 일관성 체크 등으로 구성된다. 사용자가 프로세스를 레벨에 따라 정의하면 시스템은 프로세스계층도를 출력하여 준다. 그림5는 KBSPRM의 프로세스계층도 화면이다.

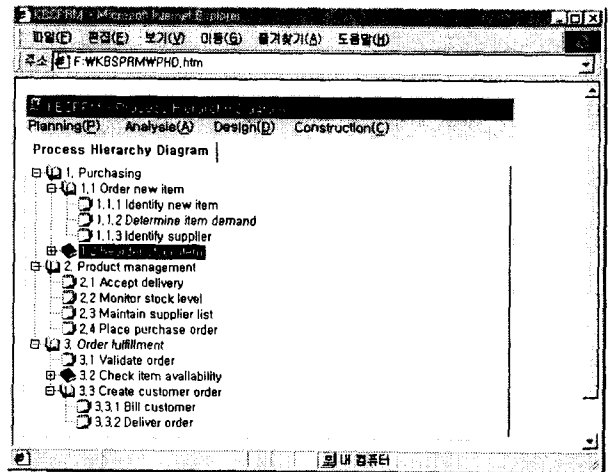


그림 5. 프로세스계층도 (KBSPRM)

- 이벤트/프로세스 상호작용 분석: 상호작용 분석을 위하여 이벤트/프로세스 매트릭스를 제공한다. 매트릭스의 행은 이벤트를, 열은 프로세스를 나타낸다. 특정 이벤트와 프로세스가 만나는 셀에 있는 'X' 값은 그 프로세스가 이벤트 대응에서 사용된다는 것을 의미한다. 특정 이벤트에 대해 연결된 프로세스가 있으면 매트릭스의 해당 셀에 'X'가 populate된다. 마찬가지로 특정 프로세스에 대해 연결된 이벤트에 대해서도 해당 셀에 'X'를 populate한다. 이미 등록된 셀값에 대하여 이벤트와 프로세스가 올바르게 호출되었는지 검증한다. 그림6은 KBSPRM의 이벤트/프로세스 매트릭스 화면이다.

Event	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
E1										
E2										
E3										
E4										
E5										
E6										
E7										
E8										
E9										
E10										

그림 6. 이벤트/프로세스 상호작용 매트릭스 (KBSPRM)



KBSPRM 은 이벤트 분석, 프로세스 분석, 상호 작용 분석 등 각각의 모델링 작업에 대한 검증 규칙을 제공한다. 모든 규칙은 지식베이스 내에 저장되며 시스템 사용자가 특정 액션을 취하면 KBSPRM 이 해당 규칙을 기동하여 실행시키게 된다. 그림 7 은 이벤트/프로세스 상호작용 분석에서 특정 이벤트에 대한 관련 프로세스의 적정 수를 검증하는 규칙을 구조적으로 표현하고 있다.

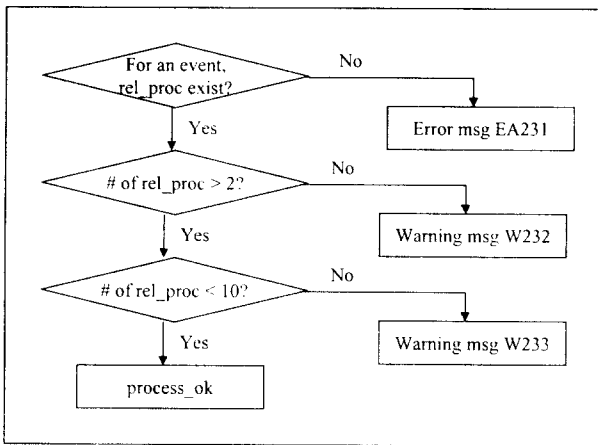


그림 7. 이벤트/프로세스 상호작용 검증규칙 (KBSPRM)

위의 그림은 이벤트/프로세스 상호작용 분석을 위하여 이벤트에 대한 관련 프로세스들을 검증하는 규칙을 나타내고 있다. 첫번째 의사결정에서는 특정 이벤트에 관련된 프로세스가 하나도 없는 경우 해당 이벤트에 대하여 에러 메시지 E231 을 출력하게 된다. 두번째 의사결정에서는 관련 프로세스가 두 개 미만인 경우에 대하여 경고 메시지 W232 를 출력한다. 세번째 의사결정에서는 관련 프로세스가 10 개 이상인 경우에 대하여 경고 메시지 W233 를 출력한다. 프로세스 분석시 하나의 프로세스가 7 개 이내의 서브 프로세스로 분해되어야 하는 것과는 달리 하나의 이벤트에 관련된 프로세스는 많을 수 있으므로 여기서는 상한 기준을 10 개로 지정하였다. 세 가지 의사결정을 전부 'Yes'로 통과한 이벤트는 정상적으로 종료 (process\_ok) 된다.

프로세스에 대해서도 유사한 규칙이 적용된다. 특정 프로세스에 대하여 관련 이벤트가 하나도 없는 경우에는 에러 메시지 E234 를 출력한다. 다만 이벤트에 관련된 프로세스가 3 개 이상이어야 하는 것과는 달리 프로세스에 관련되는 이벤트가 하나인 것은 별 문제가 되지 않는다. 한 이벤트만을 지원하기 위한 프로세스는 존재할 수 있기 때문이다. 프로세스에 관련된 이벤트가 5 개 이상인 경우에는 경고 메시지 W235 를 출력한다.

다음은 매트릭스 검증 규칙의 일부로서 그림 7 에 대한 규칙을 나타내고 있다.

```

IF matrix_check is triggered
do while event is empty (or for all event)
loop (IF for an event, related_process IS NULL
THEN store event name to list_of_temp_E231
/* E231 = 'Error: related process(es) not exist' */
ELSEIF for an event, related-process < 2
THEN store event name to list_of_temp_W232
/* W232 = 'Warning: number of related
process(es) less than 2' */
ELSEIF for an event, related-process >= 10
THEN store event name to list_of_temp_W233
/* W233 = 'Warning: number of related
process(es) greater than 10' */);

/* generate consistency report */
IF temp_E231 is not null
THEN display error message E231,
list_of_temp_E231
ELSEIF temp_W232 is not null
THEN display warning message W232,
list_of_temp_E232
ELSEIF temp_W233 is not null
THEN display warning message W233,
list_of_temp_E233
ELSEIF display 'process_ok';
  
```

이는 다시 실행 가능한 소스코드로 전환되어 지식베이스에 저장된다.

시스템의 검증 기능을 선택하면 추론엔진은 지

식베이스에서 적절한 규칙을 추출하여 프로그램을 실행시킨다. 그 결과 일관성 검증 보고서가 출력되며 이는 분석가와 사용자가 모델을 다시 한번 검증할 수 있는 기회를 제공함으로써 분석 결과의 품질을 향상시킬 수 있다.

## 결론

시스템 개발 생명주기에서 모델링 작업은 분석가의 경험과 노하우에 의해 품질이 영향을 받는 지식 집약적인 작업이다. 따라서 모델링 절차와 규칙에 대한 전문가의 지식과 경험을 지식베이스로 구축하여 활용하는 것은 모델링 작업을 보다 용이하게 하고 결과 시스템의 품질을 향상시키게 된다.

이 연구에서는 업무 분석 단계의 주요 작업 중 프로세스 모델링에 대한 지식베이스를 구축하는 과정을 살펴 보았다. 이를 위하여 먼저 최근에 많이 채택되고 있는 사용자 상호작용이 많은 시스템 환경에서의 적합한 프로세스 모델링 방법을 제시하였다. 대부분의 업무가 외부로부터의 이벤트를 받아 들임으로써 기능이 수행되는 점에 착안하여 이벤트 중심의 프로세스 모델링 방법을 제안하였다.

제안되는 방법은 먼저 이벤트 및 관련된 대응 분석을 하고 프로세스 분석을 수행하며, 마지막으로 두 작업을 상호검증하기 위하여 이벤트/프로세스 상호작용 분석을 수행한다. 이벤트 분석에서는 이벤트 대응 목록을 작성하며 프로세스 분석에서는 프로세스계층도와 프로세스의존도를 작성한다. 상호작용 분석에서는 사용자에게 직관적인 인터페이스를 제공하기 위하여 매트릭스 기법을 이용하였다. 제안된 방법을 웹 상에서 시스템으로 구현하였다. 시스템은 사용자 데이터베이스, 지식베이스, 추론엔진, 사용자 인터페이스로 구성되며, 지식베이스는 다시 논리적 규칙과 실행코드로 구성된다. 특정 상황이 발생하면 추론엔진에 의해 관련 규칙이 기동되고 실행

코드가 호출된다.

향후 연구방향은 분석 단계 전체를 지원하는 지식베이스 시스템을 개발하는 것이다. 이 연구에서 수행한 프로세스 모델링을 위한 지식베이스 시스템과의 일관성을 유지하는 데이터 모델링 지원 지식베이스 시스템을 개발하고, 두 시스템을 통합한다. 통합된 시스템은 프로세스 모델과 데이터 모델을 상호 검증하는 기법을 제공해야 한다.

## 참고문헌

- [Curtis, 1992] Curtis, B., Kellner, M. I., Over, J., Process modeling, Communications of the ACM, 35(9), 75-90, 1992.
- [Harmone, 1993] Harmon, P., Hall, C., CASE technology, Intelligent Software Systems Development: An IS Manager's Guide, New York: Wiley, 1993.
- [Kim, 1994] Jung-Eui Kim, Un-Bae Hwang, Jung-Ryul Park, Su-Yeon Kim, Information Engineering: Practical Advice from the Experts, POSDATA, 1994.
- [Lee, 1999] Ted Lee, Bohn-Oh Kim, A Knowledge-based consulting system for process modeling in systems analysis, Expert Systems with Applications, 16, 245-251, 1999.
- [Martin, 1990] James Martin, Information Engineering, Book II: Planning and Analysis, Prentical Hall, 1990.
- [POSDATA, 1994] POSDATA, POS-IEM V.1.1, Vol. 3, POSDATA, 1994.
- [Pressman, 1992] Roger S. pressman, Software Engineering: a practitioner's approach, McGraw-Hill Inc., 1992.
- [Storey, 1993] Storey, V. C., Goldstein, R. C., Knowledge-based approaches to database deisgn, Management Information Systems Quarterly, 17(1), 25-46, 1993.