

웹기반 비즈니스 프로세스 명세를 위한 BPN 모형

최상수, 이강수
한남대학교 컴퓨터공학과

A BPN model for Web-based Business Process Modeling

Sang-Su Choi, Gang-Soo Lee
Department of Computer Engineering, Hannam University
E-mail : gcss09@se.hannam.ac.kr, gslee@eve.hannam.ac.kr

요약

최근 대부분의 정보시스템은 웹기반 정보시스템으로 이주하고 있으며 이의 개발과 유지보수시에 “웹 위기” 현상이 발생하고 있다. 이를 해결하기 위한 웹엔지니어링 기술 중 웹기반 비즈니스 프로세스 명세 기술이 필요하다. 따라서 본 논문에서는 웹기반 비즈니스 프로세스 명세를 위한 BPN(Business Process Net) 모형을 제시한다. BPN 모형은 베타분포형 확률 패트리넷이며 수행가능형 Activity Diagram이라 할 수 있다. BPN을 모형화할 때 Use Case 분석을 이용하며, 비즈니스 프로세스의 수행 시간 및 비용적 불확실성은 베타분포를 이용하고 있다. BPN 모형은 XML 기반 비즈니스 프로세스 명세언어를 위한 공통 명세모형으로 이용될 수 있다.

1. 서론

정보통신 인프라의 발달로 인해 90년대 중반 이후부터는 대부분의 정보시스템이 ebXML 프레임워크 기반의 웹기반 정보시스템 형태로 이주되고 있고 웹의 응용도 복잡해지고 있다. 또한, 웹기반 정보시스템의 개발시에 개발비용과 기간이 초과되고 유지보수 비용이 증가하는 웹 위기(web crisis) 현상이 발생함에 따라 이를 해결하기 위한 웹공학 기술이 개발되고 있다[1]. 이는 60년대 말의 소프트웨어위기 발생과 소프트웨어공학 기술의 등장 상황이 30년 만에 재현되고 있는 것이다.

웹기반 정보시스템은 e-비즈니스 시스템의 전형적인 구조이며 비즈니스 프로세스(BP: Business Process)에 따라 B2B, B2C 등으로 구분할 수 있다. 특히 e-비즈니스 시스템의 성능과 수익성을 이를 운영하는 조직의 성패를 좌우함으로 체계적이고 비용-효과적인 웹공학 기술이 연구 및 개발되고 있다[2~4].

또한, 웹기반 정보시스템으로의 재설계를 위해서는 혁신적인 BP의 창조(BPR: BP Reengineering) 기술[5]과 BP의 모형화(BPM: BP Modeling) 및 지원도구를 활용할 수 있다[6]. BPM은 BP의 기능(what), 행위(when, how), 조직(where, by whom) 및 정보 모형으로 구성되며[5] 이를 통해 BP의 활동, 자원, 통제, 자료흐름, 조직구조를 표현할 수 있어야 한다. 또한, BPM은 시스템 리엔지니어와 협업 담당자간의 의사소통 수단이므로 가독성, 수행 및 분석 가능성, 정형성 및 표현성을 가져야 한다. DFD, 플로우차트, ERD, UML 및 패트리넷 등과 같은 기존의 소프트웨어 명세모형[7]은 BPM으로 널리 사용되고 있다.

특히, ebXML 또는 유사 프레임워크 기반의 웹기반 정보시스템을 위하여 협력 시스템간의 BP의 상호운용성과 호환성을 제공할

목적으로 BPM의 상위수준에서 사용하는 XML 기반의 BP 명세언어(마크업언어)[8,9]들을 개발하였지만, 이들은 스크립트 형태이므로 가독성 및 정형성이 부족하고 명세결과를 분석(수행)할 수 있는 특성인 수행가능성이 부족하다. 또한 기존 명세언어들은 호환성이 부족하므로 이들과 독립적인 공통의 정형적 명세모형이 필요하고, 웹기반 정보시스템으로의 이주를 위한 BPR에 대한 지침과 명세된 BP의 성능에 대한 정량평가를 위한 지침이 부족하다.

BPM 수준에서, UML 기반의 BP 명세모형[10]은 UML의 장점을 가지지만 수많은 도형(부분 모형화)을 사용하므로 부분 모형간의 일관성 검증이 어렵고 수행가능하지 않으며 정량평가를 위한 지침이 부족하고, 패트리넷 기반의 BP 명세모형[11~15]은 정형성이 있고 수행가능하지만 정량적 평가지침과 BPR을 위한 지침이 부족하다.

이러한 배경에서, 본 논문에서는 웹공학 기술중의 일부인 웹기반 BP의 모형화 및 분석을 위한 BPN(Business Process Net) 모형을 제시하고, 웹기반 BP를 BPN으로 명세하는 방법과 BP의 분석 방법을 제시한다. BPN 모형은 기존의 패트리넷 및 Activity Diagram을 통합한 형태로, BP에 대한 정량평가 가능을 갖는 정형적이고 수행가능하며 도형적인 BP 명세 모형이다. 본 연구의 결과는 웹기반 정보시스템의 명세, 분석 및 설계단계에서 활용할 수 있다.

본 논문의 2장에서는 기존의 BP 모형들을 조사 및 비교하며, 3장에서는 새로운 BP 모형인 BPN을 제시하고 BPN으로 모형화하는 방법을 보인다. 4장에서는 ‘웹기반 물품구매업무’에 대한 사례 연구를, 끝으로 5장에서 결론을 맺는다.

2. 관련연구

2.1 UML 기반의 명세모형

(*) 본 연구는 한국과학재단 목적기초연구(과제번호: 2001-1-51300-002-2) 지원으로 수행된 결과의 일부임.

Eriksson-Penker Business Extension(EPBE)[10]은 BP 모형화에 적합하도록 기본 UML에 프로세스(17가지), 지원 및 규칙(7가지), 목표(9가지) 및 기타(2가지)를 확장한 BP 모형이다. EPBE를 통해 4가지의 비즈니스 뷰(비전, 프로세스, 구조 및 행위)에 따라 BP, 지원, 목표, 비즈니스 규칙 및 관계를 모형화한다. 특히, 프로세스 다이어그램과 어셈블리라인 다이어그램은 EPBE의 특징이다. EPBE는 BP의 모든 요소를 모형화할 수 있지만 다음과 같은 단점을 갖는다. 첫째, 너무 다양한 모형화 요소가 존재하며(표준 UML에 35개의 새로운 아이콘 추가) 둘째, 많은 모형화 요소들간의 일관성 검증을 위한 지침이 부족하고 세째, 모형의 정적 및 동적 분석방법과 정량평가방법을 제시하지 못하고 있다. 따라서, EPBE는 UML에 숙달된 전문가만이 사용할 수 있으며 BP의 명세만을 위한 모형이라 평가된다.

2.2 패트리넷 기반의 명세모형

유한상태기계와 튜링머신의 중간에 위치하는 패트리넷은 BPR 분야에서 널리 사용하는 EPC(Event Driven Process Chain)와 함께, BP의 특성(병행성, 비동기성 및 비결정성 등)을 도형적 및 정형적으로 모형화하고 분석하기에 적합한 모형이다. 이에 따라, 패트리넷의 개념과 도형은 다양한 모형(UML, Workflow 모형 등)에서 이용되고 있다[11].

표준 패트리넷의 모형화 능력을 높이기 위해 시간, 객체, 컬러, 확률 및 퍼지 패트리넷 등이 제시되어 있으며, 이들을 'High-level' 패트리넷으로 통칭한다. 또한, LOOPN++, Design/CPN, STATEMATE, LEU, RBN, GSPN, PAILS, MPN 등과 같은 분석도구가 개발되어있다[12]. BPM은 BP의 활동, 지원, 통제, 자료흐름, 조직구조 등을 모형화할 수 있어야 하므로 다음과 같은 High-level 패트리넷들이 사용된다.

MPN(Multilevel Petri Net)[12]은 ERMIS라는 프로토타입 수준의 도구까지 개발되어있는 BP의 모형화 및 시뮬레이션 모형이다. MPN은 계층적으로 BP의 각 요소들을 모형화하여 BPR을 위한 BPM으로써 사용하고 있다.

BPP-net(Business Process Petri nets)[13]은 객체지향 구조를 통해 모형화할 때의 적응성과 융통성을 높이고 있다. 즉, 모형을 응용에 따라 분할하고 프로세스 중심의 접근방법을 택하였으며 시스템의 Workflow 부분을 별도의 객체로 모형화하고 있다.

HLPN(High-level Petri Net)[11]은 BP 또는 Workflow의 지원(인간, 기계, 문서 등)과 활동의 수행시간(상수형, 구간형, 확률형)을 컬러토콘으로 모형화한다. 컬러토콘은 수행시간을 모형화한 타임스텝프를 포함한다. 복잡한 모형은 '시스템'(서브넷에 해당)을 통해 계층적으로 모형화한다.

UML과 패트리넷의 통합 모형(UML-PNs)[14]은 아이디어 수준의 모형으로 UML의 친근성, 효과성 및 구현용이성과 패트리넷의 정형성, 분석 및 시뮬레이션 기능을 결합한 모형이다. UML을 통해 BP의 요구사항을 명세하고 설계 및 구현하며, 패트리넷을 통해서는 UML로 명세된 BP를 정성분석하고 시뮬레이션을 수행하여 BP를 개선해간다. 그러나, UML-PNs는 구체적인 모형화 및 분석방법이 제시되지 않았으며 아이디어수준의 모형이다.

끝으로, BP-nets(Business Procedure nets)[15]은 free-choice

패트리넷(표준 패트리넷에 제한을 가한것)을 이용하여 BP를 모형화하고 8가지 재설계 규칙을 제시하였다.

패트리넷 기반의 명세모형들은 수행시간 및 비용에 대한 정량평가 방법이 제시되지 않았으며 BP 내의 제어흐름만을 모형화하고 지원의 흐름은 모형화하지 않고 있다. BP의 특징인 수행시간과 비용의 '불확실성'을 모형화하기 어려우며, BP의 시나리오라 할 수 있는 도달성 그래프의 생성방법과 패트리넷으로 모형화할 때의 구체적인 방법이 부족하다. 마지막으로, 기존의 패트리넷 기반 BPM들은 BP의 모형화 및 분석용으로만 사용되며 모형화되고 분석된 BP에 대한 패트리넷을 웹기반 정보시스템 상에서 구현하기 위한 방법이 제시되어 있지 못하다.

3. BPM 모형

BPN(Business Process Net) 모형은 BP 모형이 가져야 할 요구 사항들을 고려하여 정의하였다.

3.1 BPN의 구조

BPN은 패트리넷의 구조상의 복잡성을 단순화하고 Activity Diagram을 확장한 레이블된 방향성 그래프이며, 그 정의와 각 구성요소는 다음과 같이 정의한다.

$$\text{BPN} = (\mathbf{P}, \mathbf{T}, \mathbf{A}, \mathbf{TK}, \mathbf{Mo})$$

- $\mathbf{P} = (p_{so}, p_1, \dots, p_n, p_{si})$ 은 비즈니스 '활동'을 정의하는 플레이스의 집합이며, 끝이 둥근 사각형으로 표기한다. p_{so} 와 p_{si} 는 각각 소스와 싱크 플레이스이며 비즈니스 활동의 시작과 종료를 모형화한다. p_{so} 와 p_{si} 를 제외한 플레이스(P')에는 활동명, 입력 및 출력자료타입, 활동시간, 활동비용이 매핑된다. 즉,

$$P' = (\text{Activity_Id} \times \text{In_Data_type_List} \times \text{Out_Data_type_List} \times \text{Activity_Time} \times \text{Activity_Cost})$$

여기서, Activity_Id는 스크립트 타입이며 In_Data_type_List 와 Out_Data_type_List는 0개 이상의 자료타입(XML-schema 또는 DTD) 리스트이다. Activity_Time과 Activity_Cost는 $\langle a, b, m \rangle$, $a, b, m \in \mathbb{R}$ 인 3개의 실수 벡터타입이다.

- $\mathbf{T} = \{t_{fork}, t_{join}\}$ 은 비즈니스 활동간의 포크 및 조인관계를 정의하는 트랜지션의 집합이다.
 - t_{fork} (포크 트랜지션)은 후속 비즈니스 활동의 동시시작 (concurrent initialization)을 모형화하며 막대로 표기한다.
 - t_{join} (조인 트랜지션)은 사전 비즈니스 활동의 비동기적 결합 (asynchronous join)을 의미하며 막대로 표기한다.
- $\mathbf{A} \subseteq (P \times T) \cup (T \times P) \cup (P \times P)$ 는 플레이스와 트랜지션, 트랜지션과 플레이스 및 플레이스와 플레이스를 연결하는 아크(arc)의 집합이며, 비즈니스 활동의 순서를 의미하며 화살표 또는 직선의 끝을 작은 원으로 표기한다. 직선의 끝을 작은 원으로 표시한 아크를 '금지아크' (inhibitor arc)라 하며 부정(NOT)의 의미를 나타낸다. 특히, 플레이스간을 연결하는 레이블된 아크는 '트랜지션아크(transitional arc)'라 명명하며 트랜지션의 개념이 함축되어있다. 한 플레이스로 입력 및 출력되는 2개 이상의 트랜지션아크는 각각 OR 및 선택 관계를 나타낸다.
- \mathbf{TK} 는 토큰이며 플레이스의 활성화를 표시한다.

- Mo는 초기 마킹을 의미한다.

'BPN 모듈' BPNm은 BPN에서 Mo, p_{so} 및 p_{si} 를 제외한 부분집합이며 BP 내에서 하나의 행위자(agent)에 의해 처리되는 부분을 모형화한다.

$$\text{BPNm} \subseteq [\text{BPN} - (\text{Mo}, p_{so}, p_{si})]$$

BPN은 활동간의 AND, OR, NOT 논리를 표시할 수 있으며 구조상 Activity Diagram과 유사하지만 BP의 활동간의 순서관계뿐 아니라 활동의 수행시간 및 비용, 활동에서의 입력 및 출력 자료구조와 자료흐름을 모형화할 수 있다. 표준 패트리넷과는 달리 포크 및 조인 트랜지션을 제외하고는 플레이스간에 트랜지션을 표시하지 않고 '트랜지션 아크'로 표시한다. 즉 '플레이스 중심'의 패트리넷이기 때문에, 트랜지션과 아크의 수가 반정도 줄어들어 가독성이 향상되는 효과를 얻을 수 있다.

정의된 BPN에 대한 XML-DTD는 <표 1>과 같으며, DTD로 표현된 BPN은 BPNML(BPN Markup Language)이라 할 수 있다.

표 1. BPN에 대한 XML-DTD (ver 0.8)

```
<!DOCTYPE bpn [ 
  <ELEMENT bpn (place*, transition*, arc*)>
  <ELEMENT place (activityId, inDataList, dataTypeList, activityTime,
    activityCost, initialMarking, graphics)>
  <ATTLIST place id ID #REQUIRED>
    agent IDREF #REQUIRED>
  <ELEMENT activityId (#PCDATA)>
  <ELEMENT inDataList (inData*)>
  <ELEMENT inData (#PCDATA)>
  <ELEMENT outDataList (outData*)>
  <ELEMENT outData (#PCDATA)>
  <ELEMENT activityTime (at, bt, mt)>
  <ATTLIST activityTime at IDREF #REQUIRED>
    bt IDREF #REQUIRED>
    mt IDREF #REQUIRED>
  <ELEMENT activityCost (ac, bc, mc)>
  <ATTLIST activityCost ac IDREF #REQUIRED>
    bc IDREF #REQUIRED>
    mc IDREF #REQUIRED>
  <ELEMENT graphics (x, y)>
  <ATTLIST graphics x IDREF #REQUIRED>
    y IDREF #REQUIRED>
  <ELEMENT transition (transitionType, graphics)>
  <ATTLIST transition id ID #REQUIRED>
  <ELEMENT transitionType>
  <ATTLIST transitionType value IDREF #REQUIRED>
  <ELEMENT arc (arcGraphics)>
  <ATTLIST arc id ID #REQUIRED>
  <ELEMENT arcGraphics (sourcePosition, targetPosition)>
  <ELEMENT sourcePosition>
  <ATTLIST sourcePosition x IDREF #REQUIRED>
    y IDREF #REQUIRED>
  <ELEMENT targetPosition>
  <ATTLIST targetPosition x IDREF #REQUIRED>
    y IDREF #REQUIRED>
]>
```

3.2 점화규칙

(1) 일반규칙

BPN 내의 활동 플레이스에 토큰이 존재(marking)하면 플레이스는 점화가능(enabling)하다. 주어진 시간동안 점화(firing)가 계속되어 점화가 완료되면 점화된 플레이스의 토큰 수는 1개 감소하고 선택된 출력 플레이스에는 토큰수가 1개 증가한다. 점화는 플레이스에 부여된 활동의 수행 및 입·출력자료타입의 '인스턴스화'(즉, 실제 자료 생성)를 나타낸다. 한편, 포크 및 조인 트랜지션은 입력 플레이스가 모두 점화가 완료되면 점화되고 모든 출력 플

레이스를 마킹한다.

(2) 베타분포형 점화 지연규칙

BPN의 점화 지연시간은 베타분포를 따르며, 특히 구간 $[a, b]$ 상에서 최선의 값(a), 최적의 값(m) 및 최악의 값(b)을 통해 다음 추정공식으로 평균과 분산을 구한다.

$$\text{평균: } \mu = (a + 4m + b) / 6 \quad \text{분산: } \nu = [(b - a) / 6]^2$$

즉, BPN에서 마킹된 플레이스는 <그림 1>과 같이 플레이스에 부여된 3개의 활동시간(a , b , m)의 평균(μ)시간동안 점화상태를 유지한 후 출력 플레이스를 마킹한다. 여기서 a , b , m 은 각각 베타분포의 최선, 최악 및 최적값을 의미한다. 점화기간 동안에는 마킹이 계속 유지된다(즉, 플레이스에 토큰이 존재한다). 여기서, 점화기간이란 플레이스로 모형화된 단위 활동의 수행시간을 모형화한 것이다. <그림 2>는 BPN의 점화규칙들을 보인다.

(3) 점화규칙의 특징

베타분포는 1950년대부터 PERT/CPM과 같은 프로젝트관리 모형에서 사용되어 프로젝트 추진시간의 불확실성을 효과적으로 모형화하는 것으로 평가받고 있으며, BPN 모형에서도 BP의 시간적 및 비용적 불확실성을 모형화하기 위해 적용하였다. 기존의 Activity Diagram이나 XML 기반 BP 명세표준들에서는 이와 같은 개념이 포함되어있지 못하다.

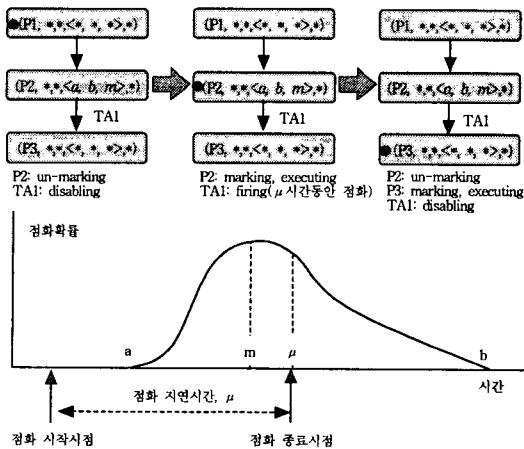


그림 1. BPN의 점화 지연규칙

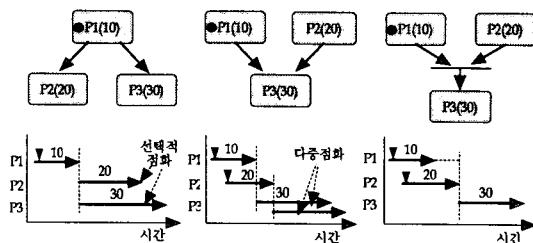


그림 2. BPN의 점화규칙

(괄호 안의 숫자는 활동수행시간 a, b, m의 평균 μ임)

BPN에서는 일반 패트리넷에서 포크와 조인 트랜지션을 제외한 트랜지션을 '트랜지션아크'로 합축하고, 트랜지션과 플레이스를 하나의 플레이스로 통합했다. 따라서, 플레이스에는 마킹(토큰이 존재)과 점화가능 및 점화(플레이스에 부여된 활동의 수행) 개념이 모두 포함되어있다.

3.3 BPN 모형화 방법

어떤 BP를 BPN으로 모형화할 때, UCD를 기반으로 하며 활동 중심으로 모형화한다. 여기서, 'UCD 기반'이란 UML의 UCD(Use-Case Diagram)을 이용하여 사전 모형화를 실시한 후 이를 토대로 BPN을 모형화하는 방법이며, '활동중심'이란 BP의 다양한 뷰(활동, 제어흐름, 자료구조, 자료흐름, 조직구조 등)들 중에서 활동을 중심으로 모형화하는 방법을 의미한다.

(1) UCD기반의 모형화 단계

UML의 UCD 모형화 방법[10]을 기반으로, 비즈니스 활동을 조사하여 '비즈니스 활동 시나리오'를 작성하고 비즈니스 활동 및 주체와 비즈니스 활동에 대한 외부/내부 관점을 명시하기 위한 행위자(actor 또는 agent)와 활동(또는, Use Case)을 파악하여 UCD를 작성한다. '행위자'는 비즈니스 활동을 수행하는 개인, 부서 또는 단위 정보시스템을 의미하며 BPN에서는 모듈로 모형화한다. 또한, '활동'은 행위자에 의해 수행되는 업무와 업무의 처리시간, 비용 및 입·출력자료를 의미하며 BPN에서는 플레이스로 모형화한다.

(2) 활동중심의 BPN 모형화 단계

BPN 모형을 작성하며 각 활동에 대한 예상 수행시간 또는 수행비용의 3가지 베타분포 값 $\langle a, m, b \rangle$ 을 해당 플레이스에 부여하고 평균 μ 과 분산 ν 를 계산한 후 입·출력자료타입을 모형화 한다. 모형화는 단계적 정제화(stepwise refinement)에 의하여 진행한다.

- ① UCD로부터 단위 활동간의 부분관계 파악: 단위활동간의 부분관계는 순차, 병행, 포크, 조인, OR 및 선택이며 복합(composit) 관계는 포크-조인, 조인-포크, 조인-선택, OR-포크 그리고 OR-선택 등이다.
- ② 각 부분관계별로 BPN 패턴을 대체: 활동간의 관계들은 <그림 3>과 <그림 4>에서 보인 BPN 패턴으로 모형화한다. 여기서, OR, 선택, 조인-선택, OR-포크 그리고 OR-선택은 PERT와 같은 전통적인 활동 네트워크에서는 정의되지 않는다.
- ③ 행위자별 모듈 구성: 하나의 행위자에 의해 수행되는 활동들은 하나의 BPN 모듈(BPNM)로 구성한다. BPNM간의 인터페이스 또는 coupling 관계는 BP의 복잡도를 측정할 때 사용되며 BP를 위한 조직구조에 해당한다. 어떤 BPNM_i의 in_degree (out_degree)는 BPNM_i 내부(외부)로 입력(출력)되는 아크수를 나타낸다.
- ④ 각 플레이스에 a, m, b 값을 추정하여 부여하고 평균 μ 과 분산 ν 계산: 각 활동의 시간 및 비용은 BP를 운영하는 조직에서 직접 측정하거나 기존의 성능 또는 비용자료를 통해 구할 수 있다. 특히, 비용 값 산정시 원가이론이나 소프트웨어비용

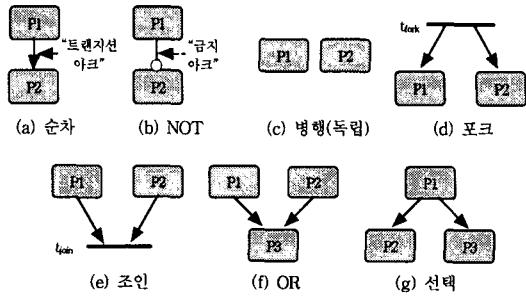


그림 3. 비즈니스 활동의 부분관계에 대한 BPN 모형화 방법

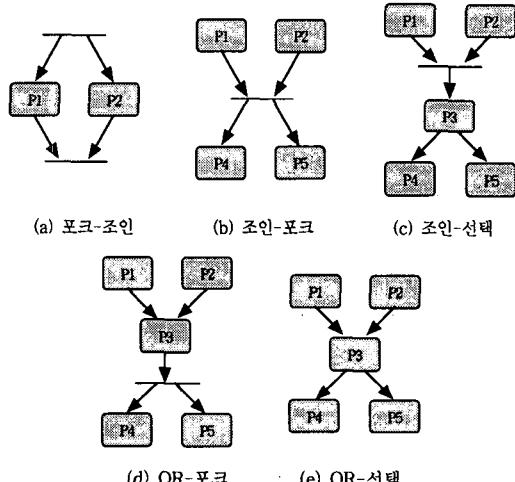


그림 4. 비즈니스 활동의 복합관계에 대한 BPN 패턴

산정기술 및 표준[16] 등을 이용한다.

- ⑤ 입·출력 자료타입의 모형화: 활동으로 입력 및 출력되는 자료타입을 모형화한다. 자료타입은 XML-Schema나 DTD 형태로 사전에 정의한다.

(3) BPN의 특징

BPN은 명세모형으로써 다음과 같은 특징을 가진다. 첫째, '활동'은 플레이스로, '자원'은 플레이스 내에 레이블링된 자원 클래스로, '제어흐름'과 '자료흐름'은 BPN 구조로, '조직구조'는 BPN 모듈간의 관계로 모형화함으로써 BPM의 필수 기능을 가진다. 둘째, 플레이스와 트랜지션의 개념을 플레이스에 통합한 활동중심의 BP 모형이다. 셋째, BP의 수행시간과 비용의 불확실성과 정량평가를 위해 점화지연시간이 베타분포를 이용하여 플레이스와 토큰을 통해 BP상의 자료 자원을 모형화하는 'High-level' 패트리넷이다. 넷째, 베타분포형 활동 수행시간과 비용을 모형화한 Activity Diagram이라 할 수 있다.

4. 사례연구

본 장에서는 '웹기반 물품구매 BP'에 대해 BPN을 이용한 모형화 사례를 보인다.

- 3.3절의 내용에 따라 물품구매 BP를 BPN으로 모형화한다. 사례 BP의 경우 5개의 행위자와 14개의 활동이 식별되었고 <그림 6>

표 2. BPN 모형의 플레이스의 의미와 활동 수행시간 및 비용

플레이스	Activity_Id	In_Data_type_List	Out_Data_type_List	Activity_Time				Activity_Cost				비용 /시간
				a	m	b	평균	a	m	b	평균	
P ₀	시작											
P1	제안내용도출(OL+FL)	구매물품목록	RFP	1.5	1.7	2	1.717	10	13	15	12.833	7.476
P2	정식입찰(OL)	RFP	정식응찰서	7	9	12	9.167	50	70	90	70	7.636
P3	주문(OL+FL)	정식응찰서	주문서	0.5	0.7	1	0.717	3	6	8	5.833	8.14
P4	제무DB에자료입력(OL)	주문서	전산자료	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P5	제무DB갱신(OL)	전산자료	거래내역,업체정보,계좌정보	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P6	거래정보조회(OL)	거래내역,업체정보	예비구매요청서	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P7	신용정보조회(OL)	업체정보,계좌정보	신용정보	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P8	지불승인2(OL)	거래내역	지불승인,거래내역	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P9	지불승인3(OL)	신용정보	지불승인,계좌정보,업체정보	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P10	물품납품(FL)	주문서	거래물품,송장	144	156	168	156	650	710	750	706.667	4.53
P11	물품검수(FL)	거래물품	물품확인서	0.1	0.3	0.5	0.3	1	2	3	2	6.667
P12	지불승인1(OL)	물품확인서	지불승인	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P13	대금지불(OL)	지불승인1,2,3,거래내역,계좌정보,업체정보	은행입금표	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P14	후속업무(OL)	은행입금표	제무자료	0.05	0.07	0.1	0.072	0.3	0.6	0.8	0.583	8.14
P _s	종료											

과 <그림 7>은 각각 사례 BP에 대한 UCD와 BPN 모형을 보인다. BPN 모형 내의 각 플레이스의 의미와 가정된 활동기간 및 비용에 대한 분석 결과는 <표 2>와 같으며, BPML을 이용한 XML 명세는 <표 3>과 같다.

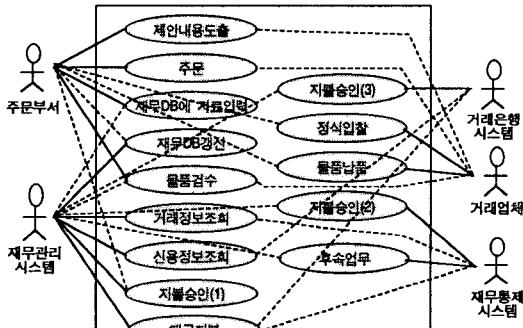


그림 6. 사례 BP에 대한 리엔지니어링된 UCD 모형

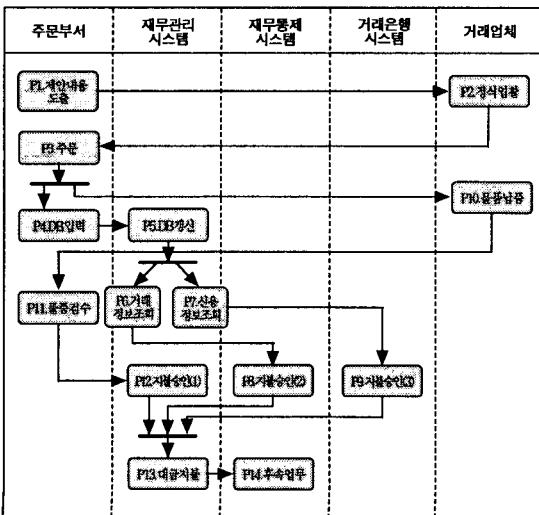
그림 7. 사례 BP의 리엔지니어링된 BPN 모형
(시작 및 종료 플레이스는 생략함)

표 3. 사례 BPN에 대한 XML 명세

```

<?xml version="1.0" encoding="UTF-8"?>
<bpmn
    xmlns="http://www.w3.org/1999/XMLSchema-instance"
    xmlns-noNamespaceSchemaLocation="http://www.w3.org/1999/XMLSchema-instance"
    id="BPMN1"
    type="BPMN">
<place id="P0" name="주문부서">
<activity value="제안내용도출"/>
<inDataTypeList>
<inData>구매물품목록</inData>
</inDataTypeList>
<outDataTypeList>
<outData>RFP</outData>
</outDataTypeList>
<activityTime>a=1.5 m=1.7 b=2</activityTime>
<activityCost>a=10 m=13 b=15</activityCost>
<initialMarking><value>1</value></initialMarking>
<initialMarking><value>1</value></initialMarking>
<graphics>xPosition="20" yPosition="10" />
</place>
<place id="P1" name="제무관리시스템">
<activity value="제무DB에자료입력"/>
<transition id="T1">
<transitionType>fork</transitionType>
<graphics>xPosition="20" yPosition="20" />
</transition>
<transition id="T2" name="T2 ~ T3 경과" target="P2">
<transitionType>join</transitionType>
<graphics>xPosition="20" yPosition="30" />
</transition>
<arc id="A1" source="P1" target="P2">
<arcGraphics>
<sourcePosition x="20" y="10" />
<targetPosition x="20" y="20" />
</arcGraphics>
</arc>
<place id="P2" name="거래업체">
<activity value="정식입찰"/>
<transition id="T3" target="P3">
<transitionType>join</transitionType>
<graphics>xPosition="20" yPosition="40" />
</transition>
<place id="P3" name="거래업체">
<activity value="지불승인(3)"/>
<transition id="T4" target="P4">
<transitionType>fork</transitionType>
<graphics>xPosition="20" yPosition="50" />
</transition>
<place id="P4" name="재무통제시스템">
<activity value="제무DB갱신"/>
<transition id="T5" target="P5">
<transitionType>join</transitionType>
<graphics>xPosition="20" yPosition="60" />
</transition>
<place id="P5" name="주문부서">
<activity value="제안내용도출"/>
<transition id="T6" target="P0">
<transitionType>join</transitionType>
<graphics>xPosition="20" yPosition="70" />
</transition>
</place>
</bpmn>
  
```

5. 결론

본 논문에서 제시하는 BPN 모형은 웹기반 정보시스템으로의 이주를 위한 웹기반 BP의 명세를 위한 모형으로써 제시하였으며, 제시한 모형의 사용방법을 보이기 위하여 웹기반 물품구매 BP를 예로 하여 전체 적용과정을 보였다.

BPN은 패트리넷과 기존의 BP 명세표준 및 Activity Diagram의 장단점을 보완한 새로운 BP 명세모형이며 제시한 모형화 방법은 웹기반 정보시스템의 분석 및 설계단계에서 적용될 수 있다. BPN을 이용한 BP의 모형화를 통하여 웹기반 정보시스템의 개발

및 운용상의 수익성을 극대화할 수 있을 것이다.

본 논문에서 제안하는 BPN 모형은 분석을 위한 알고리즘의 부족 및 오프라인 BP를 온라인상으로 리엔지니어링하기 위한 방법론이 부족하다. 따라서, 정량적 분석을 위한 알고리즘 개발, 오프라인 BP를 온라인 BP로 리엔지니어링하기 위한 방법론 제시를 향후 연구과제로 남긴다.

참고문헌

- [1] A. Ginige, S. Murugesan, "Web Engineering: An Introduction," *IEEE Multimedia*, Vol.8, No.1, pp.14-18, Jan-Mar. 2001.
- [2] P. Killela, *Web Performance Tuning*, O'reiley, 1998.
- [3] N. V. Flor, *Web Business Engineering*, Addison Wesley, 2000.
- [4] D. Menasce, *Scaling for E-business: Technologies, Models, Performance, and capacity Planning*, Prentice-Hall, 2000.
- [5] W. Curtis et al., "Process Modeling," *Communications of ACM*, Vol.35, No.9, pp. 75-90, 1992.
- [6] D. Brustard, P. Kawalek, M. Norris, (ed.), *Systems Modeling for Business Process Improvement*, Artech House, 2000.
- [7] A. Divis, *Software Requirements Objects, Functions and States*, Prentice-Hall, 1993.
- [8] "Business Process Modeling Language (BPML) Working Draft v0.4," BPMI.org, <http://www.bpmi.org/bpmi-downloads/WD-BPML-20010308.pdf>, Mar. 2001.
- [9] D. A. Chappell, V. Chopra, et al. *Professional ebXML Foundations*, Wrox, 2001.
- [10] H. E. Eriksson and M. Penker, *Business Modeling with UML: Business Patterns at Work*, Wiley&Sons, 2000.
- [11] van der Aalst and K. Hee, "Business Process Redesign: A Petri-net-based approach," *Journal of Computers in industry*, Vol. 29, No. 1, pp. 15-26. 1996.
- [12] A. Tsalgatidou, et al, "Multilevel Petri nets for Modeling and Simulating Organizational Dynamic Behavior," *Simulation & Gaming*, Vol. 27, No.4, pp.484-506, Dec. 1996.
- [13] D. Moldt and R.Valk, " Object oriented Petri nets in Business Process Modeling," *Lecture Notes in Computer Science*, Vol. 1806, pp.254-273, 2000.
- [14] H. Gou, B. Huang, S. Ren, "A UML and Petri net integrated modeling method for business processes in virtual enterprises, " *Proc. 2000 AAAI Symp. - Bring Knowledge to Business Process*, pp. 142-144, March 2000.
- [15] van der Aalst , A Class of Petri nets for Modeling and Analyzing Business Process, *Computing Science Report*, No. 95/26, Eindhoven University of Technology, 1995.
- [16] B. Boehm, COCOMO home page, <http://sunset.usc.edu/research/COCOMOII/index.html>.