

워크플로우 패턴 기반 ebBP(ebXML Business Process Specification Schema)의 코레오그래피 변환 분석

김남준*, 김자희**
서울산업대학교 IT정책전문대학원

Analysis, Transformation of ebBP Choreography based on Workflow Patterns

Kim, Nam Jun*, Kim, Ja-Hee**
Seoul National University of Technology
E-mail : juni74@gmail.com, jahee@snut.ac.kr

요 약

국제 전자거래 표준 언어인 ebXML은 비즈니스 프로세스의 모델링과 구현이 독립되어 있다. 특히 ebXML의 비즈니스 프로세스 구현 언어는 UML (Unified Modeling Language)기반의 BPSS (Business Process Specification Schema)에서 BPMN (Business Process Modeling Notation)기반의 ebBP (ebXML Business Process Specification Schema)로 변경되었다. ebXML은 단일화된 e-비즈니스 시장의 형성을 목표로 하고 있다. 따라서 ebBP가 비즈니스 프로세스 모델링으로 표현되는 다양한 업무 흐름을 지원 할 수 있는지 ebBP의 코레오그래피(Choreography) 표현 능력을 평가하는 것은 중요한 문제이다. 본 논문에서는 ebBP의 표현력을 워크플로우 패턴 입장에서 기존의 BPSS와 비교한다. 이 분석을 토대로 비즈니스 프로세스 모델에서 ebBP로 변환하는 방법을 제시한다.

1. 서론

ebXML (electronic business XML)은 UN/CEFACT (United Nations/ Centre for Trade Facilitation and Electronic Business)와 OASIS (Organization for the Advancement of Structured Information Standards)에서 단일화된 e-비즈니스 시장의 형성을 목표로 제안한 국제 전자거래 표준이다. ebXML은 비즈니스 협력 모델을 ebXML 표준 방법론인 UMM

(UN/CEFACT Modeling Methodology)에 따라 분석하도록 권고하고 있으며, UMM은 접근방법에 있어서 ISO 표준인 Open-EDI 참조모델에 그 그간을 두고 있다[4][10]. ebBP는 ebXML의 비즈니스 협력을 지원하기 위해 UMM 메타 모델에 근거하여 비즈니스 프로세스 명세를 위한 표준 언어를 정의하였다. 비즈니스 프로세스 명세를 위한 표준 언어인 BPSS는 초기에 UN/CEFACT TMG(Techniques

and Methodologies Group) 주관으로 v1.1까지 만들어 졌으나 OASIS에서는 BPSS v1.0 기반으로 ebBP를 개발하여 현재 v2.0.4를 발표 하였다[9]. 이 두 단체는 2006년3월 BPSS의 표준화를 위한 합의를 통해 ebBP를 OASIS 표준으로 확정하고 ISO표준으로 제정하기로 하였다. 2007년5월 ebBP v2.0.4는 OASIS 표준으로 확정된 상태이다[6]. BPSS의 표현력 분석에 대해 Kim과 Huemer는 Aalst등이 2003년 상용 워크플로우 관리 시스템을 분석하여 제시한 20개의 워크플로우 패턴을 기반으로 13개의 패턴이 BPSS에서 표현됨을 연구한 바 있다[1][7]. 그러나 ebXML의 비즈니스 프로세스 명세를 위한 표준언어가 BPSS에서 ebBP로 변경되고, 워크플로우 패턴도 23개가 추가되어 43개의 패턴으로 확장되면서 이와 관련되어 BPSS와 ebBP의 표현력을 비교 분석한 연구는 아직 없는 상태이다. 그러므로 본 논문에서는 BPSS와 ebBP에서 비즈니스 트랜잭션 또는 하위 협력들의 정렬과 전이를 기술하는 코레오그래피(Choreography)의 표현력 분석을 위해 Aalst등이 2003년 제시한 20개의 워크플로우 패턴과 2006년 추가로 제시한 23개의 워크플로우 패턴을 기반으로 BPSS와 ebBP의 코레오그래피 표현력을 비교 분석한다. 그리고 이 분석을 토대로 ebBP로 변환하는 방법을 제시한다. 2장에서는 관련연구로 UMM과 ebBP, 워크플로우 패턴에 대해 살펴보고 3장에서는 ebBP에서 추가로 지원 가능한 워크플로우 패턴을 설명하면서 ebBP의 코레오그래피 표현을 기존의 BPSS와 비교하여 설명한다. 4장에서는 BPSS와 ebBP에서 지원 가능한 워크플로우 패턴에 대해서 비교 정리한다.

2. 관련연구

2.1. UMM 과 ebBP

ebXML의 비즈니스 프로세스 모델링 방법론으로 권장하고 있는 UMM 모델링 기법은 OMG(Open Management Group)의 UML에 기반하고 있다[2][10]. UMM은 기술에 중립적이고 구현에 독

립적인 방식으로 정보교환을 포함한 비즈니스 프로세스(비즈니스 협력)를 모델링 하는 절차를 제공한다[2]. UMM의 기본범위는 비즈니스 트랜잭션을 기술하는데 필요한 비즈니스 결정 및 약속(Commitment)과 관련되는 비즈니스 트랜잭션의 관점을 제공하는 것으로 비즈니스 운영 관점(BOV: Business Operational View)을 중심으로 모델링 하는 방법론을 제시한다. 비즈니스 운영 관점(BOV)은 BDV, BRV, BTV, BSV의 4가지 기본 관점들로 구성되어 있다. BDV(Business Domain View)는 비즈니스 도메인을 BA(Business Area), PA(Process Area), BP(Business Process)로 분할하여 현재의 비즈니스 프로세스를 파악하고 재사용 가능한 기존의 프로세스를 발견할 수 있는 방법을 제공한다. BRV(Business Requirements View)는 비즈니스 프로세스 협력에 있는 비즈니스 프로세스들간의 관계를 확인하여 요구사항을 파악하는 관점을 제공한다. BTV(Business Transaction View)는 비즈니스 정보 개체(Entity)의 의미(Semantics)와 역할 간의 교환흐름(Choreography)을 파악하는 관점을 제공한다. BSV(Business Service View)는 컴포넌트 서비스와 에이전트, 그리고 비즈니스 협력을 실행하고 허가하는데 필요한 상호작용으로서 이들간의 메시지(정보) 교환을 정의하기 위한 관점을 제공한다.

ebXML의 비즈니스 프로세스 구현 언어인 ebBP 기술 명세는 비즈니스 트랜잭션으로 구성되는 비즈니스 협력의 실행을 지원하기 위해 UMM 메타 모델에 근거하여 비즈니스 프로세스 명세를 위한 표준 언어를 정의하였다. ebBP 정의의 목표는 비즈니스 프로세스 모델링과 e-비즈니스 소프트웨어 컴포넌트들 사이의 차이를 메우기 위한 것이다[9]. 비즈니스 협력은 비즈니스 파트너 또는 협력 당사자 사이에서 비즈니스 트랜잭션을 수행하는 활동으로 비즈니스 트랜잭션의 코레오그래피를 통해 협력하는 일련의 역할들로 이루어져 있다. 비즈니스 파트너들은 각각의 역할을 통해서 비즈니스 협력에 참여한다. 비즈니스 트랜잭션은 두 비즈니스

파트너 사이의 상거래 협정과 관련되는 가장 기본적인 작업 단위를 나타내며, 요청 역할과 응답 역할 그리고 문서 흐름으로 이루어져 있다. 비즈니스 협력은 비즈니스 파트너의 역할 수에 따라 양자간 또는 다자간 협력으로 특화 될 수 있다. ebBP는 협력의 수와 상관없이 두 역할 이상의 다자간 협력을 지원하는 BusinessCollaboration을 기본 요소로 사용하며, ebBP의 이전 버전인 BPSS와의 호환성을 위해 양자간 협력 요소인 BinaryCollaboration 과 다자간 협력 요소인 MultipartyCollaboration을 유지하고 있다. ebBP에서 코레오그래피는 비즈니스 협력내의 비즈니스 트랜잭션 또는 하위 협력들의 정렬과 전이를 기술한다. 코레오그래피는 시작상태, 완료상태, 활동, 분기, 결합, 결정, 활동 사이의 전이 와 전이의 감시 같은 활동 다이어그램의 개념을 사용하여 ebBP 스키마(Schema)에 지정될 수 있다.

2.2 워크플로우패턴(Workflow Patterns)

패턴은 일반적인 특정 상황에서 반복적으로 발생하는 구체적인 형태의 추상화를 의미한다[3]. 공통적으로 반복되는 요소들을 설명하여 패턴을 찾고, 이를 분석 및 설계에 연계하려는 노력은 소프트웨어 개발과 관련하여 1990년대 중반부터 많은 관심을 받았다. Gamma등은 객체지향 시스템에서 반복적으로 발생하는 최소한의 상호작용을 기술하는 23가지 디자인 패턴을 최초로 제시하였다[5]. Aalst등은 2003년 상용 워크플로우 관리 시스템을 분석하여 다양한 업무 패턴을 여섯 종류로 분류하여 총 20개의 워크플로우 패턴을 제시하였다 [12][13]. 2006년에는 Russell등과 함께 워크플로우 패턴을 여덟 종류로 세분화 하여 23개의 패턴을 추가하였다[8]. 추가된 워크플로우 패턴은 기존에 제시 하였던 기본제어, 고급분기 및 동기화, 구조적, 다중 인스턴스, 상태기반, 취소 패턴의 여섯 분류 중에서 구조적 패턴을 반복 패턴과 종료 패턴으로 세분화 하였고, 새롭게 트리거 패턴을 추

가 하였다. 기본제어 흐름 패턴은 일반적인 순차 전이와 분기, 병합 등을 나타내는 패턴으로 WfMC(Workflow Management Coalition)에서 제공되는 기본적인 제어흐름 개념과 유사하다[11]. 고급 분기 및 동기화 패턴은 분기 및 병합의 표현에 있어서 조건적인 제약이 따르는 패턴이다. 다중 인스턴스 패턴은 생성되는 인스턴스의 발생시기와 동기화 여부에 따라 분류되며, 상태기반 패턴은 하나의 활동이 실행되는데 있어 다른 활동의 상태에 의존적인 패턴이다. 취소 및 강제완료 패턴은 취소 할 수 있는 활동의 개수와 프로세스의 취소 여부에 따라 분류된다. 반복 패턴은 반복 구조에 따라 분류되며, 종료 패턴은 명시적인 표현 여부에 따라 분류된다. 새롭게 추가된 트리거 패턴은 프로세스의 다른 활동 또는 외부 신호에 따라서 특정 작업의 시작을 요청하는 패턴으로 트리거 생명주기(life cycle)에 따라서 분류된다. 각각의 분류 별로 추가된 패턴들의 가장 큰 특징은 다중 인스턴스 처리를 위한 부분이 강화된 것과 트리거 패턴이 추가된 것이다.

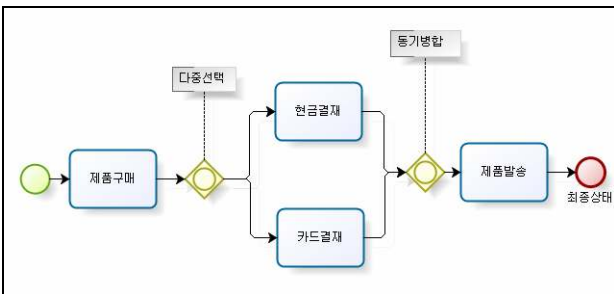
3. 워크플로우 패턴 기반 코레오그래피 (Choreography) 변환 분석

이 장에서는 ebXML의 구현 언어가 BPSS에서 ebBP로 변경되면서 달라진 사항을 설명하고 기존 20개의 패턴에서 43개의 패턴으로 확장된 워크플로우 패턴을 기반으로 BPSS와 ebBP의 코레오그래피를 비교 분석 한다. 이 분석을 토대로 BPSS와 ebBP가 표현 가능한 워크플로우 패턴에 대해 다룬다.

BPSS에서 코레오그래피는 BinaryCollaboration의 요소로 정의 되었으나 ebBP에서 코레오그래피에 관련되는 요소들이 collaborationGroup으로 묶여서 정의 되었다. 그리고 BPSS에서 속성(Attribute)으로 정의되었던 항목들이 ebBP에서는 요소(Element)로 변경되어 스키마(Schema) 구조를 좀더 명확하게 하였다. 예를 들면 <그림2>의 '06'라인에 사용된

것과 같이 TimeToPerform 은 BPSS에서는 속성 (Attribute) 이었으나 ebBP에서는 요소(Element)로 정의되어 duration 속성을 갖는다. 그리고 전이 및 분기, 병합 요소인 Transition, Fork, Decision, Join 요소들은 <그림2>의 '32'라인과 '40'라인, '50'라인에서 사용된 것과 같이 FromLink 와 ToLink 요소가 추가되어 표현의 명확성과 가독성을 높였다. Aalst 등이 기존에 제시 하였던 20개의 패턴 중에서 ebBP에서 추가로 지원 가능한 패턴은 동기병합 패턴이 있다. 그리고 2006년 추가로 정의된 23개의 워크플로우 패턴들 중에서는 명시적 종료 패턴이 BPSS와 ebBP에서 모두 지원 가능하다.

ebBP에서 추가로 지원되는 동기병합 패턴은 실제로 전이된 하나 또는 다수의 전이를 모두 받은 후에 동기화 시켜 하나의 출력이 이루어지는 OR-결합이다. 이 동기병합 패턴은 패턴의 특성에 따라 다중선택 패턴과 조합되어 구성되는 패턴으로 다중선택 패턴은 조건에 따라 하나 또는 다수의 전이(Transition)가 이루어 지는 OR-분기 이다. <그림 1>은 BPMN(Business Process Modeling Notation)으로 표현된 다중선택패턴과 동기병합 패턴을 나타낸다. 예를 들면 구매자가 제품주문 이후 선택할 수 있는 지불 방법이 현금결제, 신용카드 결제, 현금과 신용카드 부분결제의 3종류가 있을 경우 조건에 따라서 현금결제 활동을 수행하거나 신용카드 결제 활동을 수행 할 수 있다. 또는 현금 및 신용카드 부분결제를 선택하여 두 활동 모두를 수행 할 수 있다. 판매자는 지정된 시간 내에 구매자의 지불활동이 완료된 경우 제품발송 작업을 수행한다.



<그림1> 다중선택 및 동기병합 패턴

ebBP에서 동기병합 패턴은 <그림2>의 '41'라인에 사용된 것과 같이 Fork에 존속기간(duration)을 지정하여 표현 할 수 있다. Fork는 존속기간(duration)을 지정 할 수 있는 TimeToPerform 요소가 있고, 이 TimeToPerform의 duration 속성에 지정되는 존속기간(duration="PT1H" : 1시간) 이 끝나면 비즈니스 협력의 상태는 대응되는 Join으로 자동적으로 이동된다. Fork는 대응되는 Join없이 정의될 수 있지만, 이 경우 TimeToPerform 요소는 사용되면 안 된다. <그림2>는 다중선택 및 동기병합 패턴에 대한 예제 이다.

```

01 <!-- more -->
02 <BusinessCollaboration nameID="ID00BC01"
03 name="Example">
04   <Role name="Buyer" nameID="IDRole01"/>
05   <Role name="Seller" nameID="IDRole02"/>
06   <TimeToPerform duration="P1D"/>
07   <Start>
08     <ToLink toBusinessStateRef="A"/>
09   </Start>
10   <BusinessTransactionActivity name="OrderProducts"
11 nameID="A" businessTransactionRef="ID00CT01">
12     <TimeToPerform duration="PT3H"/>
13     <Performs currentRoleRef="IDRole01"
14     performsRoleRef="IDInitiator01"/>
15     <Performs currentRoleRef="IDRole02"
16     performsRoleRef="IDresponder01"/>
17   </BusinessTransactionActivity>
18   <BusinessTransactionActivity name="Cash" nameID="B"
19   businessTransactionRef="ID00CT01">
20     <TimeToPerform duration="PT3H"/>
21     <Performs currentRoleRef="IDRole01"/>
22     <Performs currentRoleRef="IDRole02"/>
23   </BusinessTransactionActivity>
24   <BusinessTransactionActivity name="CreditCard"
25   nameID="C" businessTransactionRef="ID00CT01">
26     <TimeToPerform duration="PT3H"/>
27     <Performs currentRoleRef="IDRole01"/>
28     <Performs currentRoleRef="IDRole02"/>
29   </BusinessTransactionActivity>
30   <BusinessTransactionActivity name="Shipment"
31   nameID="D" businessTransactionRef="ID00CT01">
32     <TimeToPerform duration="PT3H"/>
33     <Performs currentRoleRef="IDRole01"/>
34     <Performs currentRoleRef="IDRole02"/>
35   </BusinessTransactionActivity>
36   <Success name="Success" nameID="IDFinishSuccess01"/>
37   <Failure name="Failure" nameID="IDFinishFailure01"/>
38   <Transition nameID="IDT01">
39     <FromLink fromBusinessStateRef="A"/>
40     <ToLink toBusinessStateRef="ID00FORK01"/>
41   </Transition>

```

```

36 <Transition nameID="IDT02">
37 <FromLink fromBusinessStateRef=" D"/>
38 <ToLink toBusinessStateRef="IDFinishSuccess01"/>
39 </Transition>
40 <Fork name="MultiChoice" nameID="ID00FORK01"
type="OR">
41 <TimeToPerform duration="PT1H"/>
42 <FromLink fromBusinessStateRef="A"/>
43 <ToLink toBusinessStateRef="B">
44 <ConditionExpression expressionLanguage="XPath1"
expression="//PaymentMethod='Cash' or
//PaymentMethod='CashCreditCard'"/>
45 </ToLink>
46 <ToLink toBusinessStateRef="C">
47 <ConditionExpression expressionLanguage="XPath1"
expression="//PaymentMethod='CreditCard' or
//PaymentMethod='CashCreditCard'"/>
48 </ToLink>
49 </Fork>
50 <Join name="SynchronizingMerge" nameID="ID00JOIN01"
waitForAll="true">
51 <FromLink fromBusinessStateRef="B"
conditionGuard="Success"/>
52 <FromLink fromBusinessStateRef="C"
conditionGuard="Success"/>
53 <ToLink toBusinessStateRef="D"/>
54 </Join>
55 </BusinessCollaboration>
56 <!-- more -->

```

<그림2> 다중선택 및 동기병합

BPSS와 ebBP에서 모두 지원 가능한 **명시적 종료** (Explicit Termination) 패턴은 워크플로우 패턴이 세부적으로 분류되면서 추가되었으나, BPSS에서도 명시적인 종료 패턴은 사용 된다. 비즈니스 프로세스의 모델링에서 명시적 종료패턴은 일반적으로 <그림1>과 같이 최종상태 형태로 표현되며, 전이가 최종상태에 도달되는 즉시 프로세스에 남아 있는 모든 잔여 인스턴스가 취소되고 해당 비즈니스 협력은 종료 된다. BPSS와 ebBP에서 명시적 종료 패턴의 구현은 <그림2>의 ‘30’ 라인 Success와 ‘31’ 라인의 Failure 요소에 의해 표현 된다.

4. 결론

ebXML은 단일화된 e-비즈니스 시장의 형성을 목표로 제안한 국제 전자거래 표준이다. ebBP는 ebXML의 비즈니스 협력을 지원하기 위해 비즈니스 프로세스 명세를 위한 표준 언어를 정의하였다.

그러므로 본 논문은 ebBP의 표현력 분석을 위해 BPSS의 코레오그래피 변환에 대한 사전연구를 기반으로 ebBP의 코레오그래피 변환에 대한 분석을 하였다. 본 논문은 Aalst 등이 제시한 43개의 워크플로우 패턴에 대해서 BPSS와 ebBP에서 표현 될 수 있는 워크플로우 패턴을 다음 <표1>과 같이 정리 한다. 음영 처리된 부분은 2006년 추가된 패턴이며, "+"는 지원 가능 "-"는 지원하지 않음을 의미 한다.

분류	패턴명	BPSS	ebBP
Basic Control Flow Patterns	Sequence	+	+
	Parallel Split	+	+
	Synchronization	+	+
	Exclusive Choice	+	+
	Simple Merge	+	+
Advanced Branching and Synchronization Patterns	Multi-Choice	+	+
	Structured Synchronizing Merge	-	+
	Multi-Merge	-	-
	Structured Discriminator	+	+
	Blocking Discriminator	-	-
	Cancelling Discriminator	-	-
	Structured Partial Join	-	-
	Blocking Partial Join	-	-
	Cancelling Partial Join	-	-
	Generalised AND-Join	-	-
	Local Synchronizing Merge	-	-
	General Synchronizing Merge	-	-
	Thread Merge	-	-
Thread Split	-	-	
Multiple Instance Patterns	MI without Synchronization	+	+
	MI with a Priori Design-Time Knowledge	-	-
	MI with a Priori Run-Time Knowledge	-	-
	MI without a Priori Run-Time Knowledge	-	-
	Static Partial Join for MI	-	-
	Cancelling Partial Join for MI	-	-

	Dynamic Partial Join for MI	-	-
State-based Patterns	Deferred Choice	+	+
	Interleaved Parallel Routing	-	-
	Milestone	+	+
	Critical Section	-	-
	Interleaved Routing	-	-
Cancellation and Force Completion Patterns	Cancel Task	-	-
	Cancel Case	+	+
	Cancel Region	-	-
	Cancel Multiple Instance Activity	-	-
	Complete Multiple Instance Activity	-	-
Iteration Patterns	Arbitrary Cycles	+	+
	Structured Loop	-	-
	Recursion	-	-
Termination Patterns	Implicit Termination	+	+
	Explicit Termination	+	+
Trigger Patterns	Transient Trigger	-	-
	Persistent Trigger	-	-

<표1> BPSS와 ebBP가 지원하는 워크플로우 패턴

<표1>과 같이 BPSS와 ebBP의 표현력은 거의 동일하다. 그러나 ebBP에서는 표현의 명확성과 가독성을 높이기 위해 BPSS에서 속성으로 정의되었던 항목들을 요소로 변경하여 스키마 구조를 좀더 명확하게 하였다. BPSS와 ebBP의 표현력 분석을 위해 적용한 워크플로우 패턴들 중에서 Aalst등이 추가로 정의한 패턴들의 대부분은 다중 인스턴스 처리를 강화하기 위해 정의된 패턴들이다. 그러나 BPSS와 ebBP 모두 isConcurrent 속성을 통해 비동기 다중 인스턴스 패턴만 지원한다. 따라서 ebBP의 표현력 증대를 위해 다중 인스턴스 처리에 대해 미흡하였던 개념들을 ebBP에 적용하기 위한 추가적인 연구가 필요하다.

[참고문헌]

- [1] 김자희, UML2.0을 지원하기 위한 ebXML BPSS의 개선, 한국전자거래학회지 제12권 제2호 pp.17~30, 2007
- [2] 한국전자거래진흥원, 2006년 하반기

- UN/CEFACT 표준화 작업동향 보고서, 2006
- [3] D. Riehle, H Zullighoven, "Understanding and using patterns in software development", Theory and Practice of Object Systems archive Volume 2 Issue 1, pp.3-13, 1996
- [4] e비즈니스 정보센터, <http://www.kebic.or.kr/>
- [5] E. Gamma, R. Helm, R. Johnson, J. Vlissides, GOF의 디자인 패턴(개정판), 2007
- [6] ISO Standards Development ISO TC, <http://www.iso.org/tc154>
- [7] Kim J.H, C. Huemer, "Analysis, Transformation and Improvements of ebXML Choreographies based on Workflow Patterns", in Proceedings of the International Conference on Cooperative Information Systems (CoopIS04). Pp.66-88, 2004
- [8] N. Russell, A.H.M. ter Hofstede, W.M.P. van der Aalst, and N. Mulyar "Workflow Control-Flow Patterns: A Revised View", BPM Center Report BPM-06-22, BPMcenter.org, 2006
- [9] OASIS, ebXML Business Process Specification Schema Technical Specification v2.0.4, 2006
- [10] UN/CEFACT, UN/CEFACT Modeling Methodology (UMM) User Guide, CEFACT/TMG/N093, 2003
- [11] WfMC, Workflow Management Coalition Terminology & Glossary. WfMC-TC-1011, 1999
- [12] W.M.P van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros, "Workflow Patterns", Distributed and Parallel Databases, Vol. 14, pp.5-51, 2003
- [13] Workflow Patterns, <http://www.workflowpatterns.com/>