

수직적 추상의 도입에 의한 BPMN 추상기능의 확장

강 성 원[†] · 이 단 형^{††} · 안 유 환^{†††}

요 약

OMG가 개발한 비즈니스 프로세스 모델링 언어인 BPMN은 하위 프로세스를 하나의 노드로 나타낼 수 있게 하여, 사용자가 프로세스의 세부사항으로부터 추상화된 뷰(view)를 볼 수 있도록 한다. 이러한 능력은 수평적 추상으로 볼 수 있는 유용한 추상의 방향이다. 그러나 다양한 비즈니스 이해당사자(stakeholder)들이 비즈니스 프로세스를 자신의 관심(interest)의 관점에서 볼 수 없기를 희망하는 경우에는 수직적 추상이라는 다른 종류의 추상능력이 필요하다. 예를 들어, 이해당사자들이 특정 수행자집단의 관점에서 프로세스를 보기를 원할 수도 있고, 특정한 목표들의 관점에서 프로세스를 보고자 할 수도 있다. 이 논문은 슈퍼에지(super edge)의 개념을 도입하여 BPMN의 수평적 추상능력을 확장하고, 나아가 '관점속성(aspect attribute)'과 '관심명세(interest specification)'의 개념과 기호를 도입하여 BPMN이 수직적 추상능력을 갖도록 확장한다.

키워드 : 비즈니스 프로세스, 프로세스 모델링, BPMN

Extending the Abstraction Capability of BPMN by Introducing Vertical Abstraction

Sungwon Kang[†] · Danhyung Lee^{††} · Yu-Whoan Ahn^{†††}

ABSTRACT

BPMN is a standard business process description notation developed by OMG. It allows the user to have an abstract view of a process that hides its details with the Collapsed Sub-Process notation. While it is a useful direction of abstraction that can be called *the horizontal abstraction*, a different kind of abstraction, *the vertical abstraction*, is necessary when different stakeholders of business would like to have different views of the business process from their own viewpoints of interest. For example, stakeholders may want to see a process from the viewpoint of a particular group of actors or from the viewpoint of a certain set of goals. This paper first extends horizontal abstraction capability of BPMN by introducing the notion of super edge and, moreover, adds the vertical abstraction capability to it by introducing the notions of 'aspect attribute' and 'interest specification' and notations for them.

Keywords : Business Process, Process Modeling, BPMN

1. 서 론

비즈니스 프로세스(Business Process: BP)는 제품을 생산하거나 또는 서비스를 제공하는데 요구되는 일련의 활동을 의미한다. 비즈니스 프로세스 관리(Business Process Management: BPM)는 비즈니스 프로세스를 효과적이고 효율적으로 수행할 수 있도록 지원하는 활동으로, 역사적으로는 1980년대의 Total Quality Management, 1990년대의 Business Process Reengineering (BPR) 의 패러다임을 이어받아 2000년대에 기업의 관심을 끌게 된 프로세스 운동(Process Movement) 패

러다임의 제3의 물결이다[1,2].

BPM이라는 용어의 의미에 대하여는 다양한 해석이 존재하여, 조직의 핵심프로세스에 기초하여 조직을 구성, 관리 및 측정하는 하향식 방법 혹은 특정한 프로젝트 혹은 프로세스를 분석, 재설계, 개선 및 관리하는 체계적인 접근방법 혹은 프로세스 워크플로우와 프로세스 소프트웨어응용의 실행을 IT가 더 쉽게 관리하고 측정하도록 만들어주는 새로운 소프트웨어 기술 혹은 특정한 워크플로우 프로세스의 생산성 증가를 겨냥한 비용절감 노력 등으로 이해되고 있다[3]. 비즈니스 프로세스 관리 시스템 (BPMS)는 BPM 관리를 지원하는 기술적 플랫폼으로 BPM 엔진, 비즈니스 프로세스 모니터링을 위한 컴포넌트, 설계 도구, 시뮬레이션 컴포넌트 등으로 구성된다[2]. BPM 엔진은 Business Process Execution Language (BPEL) [4] 와 같은 정형적인 프로세스 기술언어로 명세 된 프로세스의 워크플로우를 진행시키며 비즈니스

[†] 정 회 원 : KAIST 정보과학기술대학 부교수

^{††} 정 회 원 : KAIST 정보과학기술대학 교수

^{†††} 종신회원 : (주)헨디피엠지 CEO/사장

논문접수 : 2008년 3월 12일

수 정 일 : 1차 2008년 9월 16일, 2차 2008년 12월 29일

심사완료 : 2009년 2월 16일

프로세스를 위한 응용들을 상호 연결해 주는 역할을 수행한다. BPEL은 컴퓨터처리를 위하여는 효율적인 텍스트기반 언어이지만, 사람이 비즈니스 프로세스를 설계, 관리, 모니터링 하기에는 적합하지 않은 언어이다. 이러한 인간의 활동을 용이하도록 고안된 언어가 Object Management Group (OMG)에서 표준으로 개발한 Business Process Modeling Notation (BPMN) [5]으로, BPMN은 비즈니스 프로세스들을 시각적 형상으로 나타내는 Business Process Diagram (BPD)을 제공하고, 또한 이를 BPEL과 같은 실행언어로 자동적으로 번역하는데 사용될 수 있는 표준 맵핑도 정의하고 있다.

기업의 비즈니스 프로세스가 복잡해 짐에 따라 이를 모델링 한 프로세스 다이어그램, 즉 비즈니스 프로세스 다이어그램도 복잡하여지므로, 이를 분석하고 관리하기 위하여는 복잡한 프로세스 다이어그램을 추상화(abstraction)하여, 들여다보고자 하는 관점에 맞는 비즈니스 프로세스의 모습만을 볼 수 있도록 하는 능력이 중요하다. 이러한 능력을 비즈니스 프로세스 모델링 도구(tool)에서 제공할 수도 있으나, 비즈니스 프로세스가 여러 사람들간에 공통적으로 인식되고 도구간 상호 호환이 가능하기 위하여는 표준언어인 BPMN에 포함되는 것이 바람직하다.

본 논문에서는 BPMN으로 모델링 된 프로세스 다이어그램을 사용자의 다양한 관심의 관점(Aspect)에서 추상화하여 볼 수 있도록 기존의 BPMN을 확장한다. 추상화는 수평적 추상(horizontal abstraction)과 수직적 추상(vertical abstraction)으로 나눌 수 있다.¹⁾ 수평적 추상은 규모의 관점에서의 추상을 말하며 그 예로 지도를 생각할 수 있다. 지도는 축척도에 따라 사물을 더 자세히 혹은 덜 자세히 들여다 볼 수 있는 추상을 제공한다. 다른 한편으로는 사용자의 관심이 특정한 측면 혹은 측면들에 있을 수 있는데, 수직적 추상은 그러한 관점들로 여과된 (즉 그러한 관점에서 유효한) 결과만을 추출한 추상화된 모델을 만들어 낸다. 예를 들어 전국의 고속도로 휴게소를 보여주는 지도의 경우 축척에 따른 실제 지형, 지물의 사실적인 정도가 중요한 것이 아니라, 모든 고속도로의 표시와 고속도로 상의 휴게소간의 상대적 위치 등이 중요한 요소가 된다. 마찬가지로 비즈니스 프로세스의 경우에도 수직적 추상은 비즈니스 프로세스의 현황을 특정한 관심의 측면에서 들여다 보고, 따라서 관련된 판단을 더 효율적으로 내릴 수 있게 하여 주는 추상의 방향이다.

BPMN은 내부 구조를 가질 수 있는 활동(activity)를 통하여 부분적으로 수평적 추상을 지원한다. 본 논문은 이러한 노드(node)를 ‘슈퍼노드(super node)’라고 부르기로 하고, 또한 BPMN의 수평적 추상능력의 확장을 위하여 ‘슈퍼에지(super edge)’의 개념을 새로이 도입한다. 다른 한편, BPMN

은 수직적 추상의 방법은 제공하지 않는다. 본 연구에서는 ‘관점 속성(Aspect Attribute)’과 ‘관심명세(Interest Specification)’의 두 가지 메커니즘으로 BPMN을 확장하여, 다양한 관점의 수직적 추상연산(abstraction operation)을 프로세스 다이어그램에 적용할 수 있게 한다.

본 논문의 제2절에서는 관련연구에 대한 논의한다. 제3절에서는 BPMN을 간략히 소개하고, BPMN이 제공하는 제한된 추상화 능력을 설명한다. 제4절에서는 수평적 추상과 수직적 추상을 위하여 BPMN을 확장하는 방법을 제시한다. 제5절에서는 4절에서 정의한 추상연산들을 적용한 예를 보여준다. 마지막으로 제6절은 결론으로 본 연구의 기여를 설명한다.

2. 관련 연구

Harel은 논문 [7]에서 기존의 상태기계(state machine)의 표기법을 확장하여 상태기계를 계층적으로 표현할 수 있게 하는 한편, 상태기계들을 동시성과 상호통신의 기능으로 확장하여 이를 상태차트(statechart)라고 불렀다 그리고 하나의 상태가 펼쳐서(expanded) 보면 상태기계로 보이며 상태기계가 하나의 상태로 압축되는 동작을 각각 줌인(zoom in), 줌아웃(zoom out)이라고 부르고, 이러한 동작이 가능한 상태를 슈퍼상태(superstate)라고 불렀다.²⁾ 이러한 확장된 상태기계에 대한 주된 개념들은 UML 2.0 상태기계 다이어그램(State Machine Diagram)에서 수용되었다[8]. 그러나 이러한 추상은 우리가 수평적 추상이라고 부르는 추상으로 상태차트에 대하여 우리가 수직적 추상이라고 부르는 추상을 시도하지는 않았다. BPMN[5]이나 UML 2.0 활동 다이어그램[8]과 같은 모델링 메커니즘은 페트리넷(Petri Net)을 기반으로 하여 활동의 다양한 참여자들 간의 동기화(synchronization)를 명시적으로 표현할 수 있기 때문에, 비즈니스 프로세스나 웹서비스에서와 같이 협업이 필수적인 도메인의 모델링에 적합하다[9]. 반면 상태기계는 개체의 행위를 기술하는 데에는 적절하지만, 여러 참여자가 협력하는 상황을 직접적으로 기술하는 데에는 적절하지 않다.

최근 서비스를 컴포넌트 단위로 보는 분산응용 개발방법론인 서비스 지향 아키텍처(SOA: Service Oriented Architecture)에 대한 많은 연구와 적용이 진행되고 있다[2,10]. SOA에서는 자동화 하고자 하는 비즈니스 프로세스를 모델링 하기 위하여 BPMN과 같은 언어를 사용한다. 그리고 BPMN 모델을 먼저 실행 가능한 언어인 BPEL [4]로 맵핑을 하고 그 결과를 BPEL 실행엔진으로 실행하면 비즈니스 프로세스가 자동으로 수행되게 된다. 그러나 BPMN의 적용에 대하여 활발한 연구가 진행되고 또 OMG안에서 BPMN을 확장하고자 하는 노력이 진행되고 있는 반면, 추상능력의 확장으로 별다른 진전이 없었다. 그러나, BPMN이 비즈니스 프로세스의 모델링과 자동화된 실행을 위한 역할뿐 아니라, 업무를 수행하는 관리자들로 하여금 다양한 분석을 할 수 있도록 하

1) 수평적 추이라는 용어에서 수평의 의미는 계층이라는 용어와는 다른 의미를 갖는다. 수평적 시이나 수평적 시장 소프트웨어와 같은 용어에서 수평은 시장의 여러 업종을 가로지른다는 의미로 사용되고 있는데 이러한 의미에 더 가깝다. 마찬가지로 수직적 시이나 수직적 시장 소프트웨어와 같은 용어에서 수직의 의미가 단일업종에 속한다는 의미와 더 유사한 개념이다. 이러한 분류에 따르면 모델체킹(Model Checking)[6]은 수직적 추상을 주로 이용하여 시스템의 모델을 만들어 이를 검증하는 활동으로 볼 수 있다. 이 논문에서 수직적 추상이라고 말하는 추상활동도 이와 밀접한 관련이 있기 때문에 이 용어를 사용하고 있다.

2) 슈퍼상태라는 용어가 이와 같이 널리 수용된 용어이므로, 본 논문에서는 이에 착안하여 슈퍼노드와 슈퍼에지의 용어를 제안하였다.

기 위하여는 본 연구에서 제안하는 바와 같은 추상능력이 필요하다.

수평적 추상의 측면에서는 BPMN은 이미 부분적으로 추상화 메커니즘을 갖고 있다. 즉 BPMN에는 두 가지의 활동이 있는데, 하나는 내부구조를 갖지 않는 원자적 활동(atomic activity)으로 BPMN에서는 이를 태스크(task)로 부른다. 다른 하나의 활동은 내부의 구조를 갖는 복합적인(complex) 활동으로서 내부구조를 보여주는 다이어그램을 펼쳐진 하위 프로세스(expanded subprocess)라고 부르고, 내부구조를 보여주지 않는 활동을 압축된 하위 프로세스(collapsed subprocess)라고 부른다 [5]. 본 논문에서는 이 후자의 활동을 나타내는 노드(node)를 ‘슈퍼노드(super node)’라고 부르기로 한다. 슈퍼노드는 순환적으로 (recursively) 다시 그 안에 활동들로 구성된 프로세스를 포함함으로써 복잡한 프로세스에 대한 계층적인 뷰를 가질 수 있도록 하여, 계층별 뷰를 쉽게 이해하고 분석할 수 있게 한다. 그러나 현재의 BPMN에는 슈퍼노드의 개념이 없고, 본 논문의 4절과 5절에서 주로 논의하게 될 수직적 추상의 개념도 갖고 있지 않다. 본 논문에서 제안하는 슈퍼노드는 UML 2.0 [8]의 상호작용 개괄 다이어그램(Interaction Overview Diagram)에서 다이어그램의 일부를 압축하여 보여주는 프레임(Frame)에 비견할 수 있는 추상이다. UML 2.0의 활동 다이어그램에서도 복잡적 활동을 나타내는 노드를 통하여 본 논문의 슈퍼노드에 상응하는 추상을 허용하나, 추가적으로 상호작용 개괄 다이어그램을 통하여 필요에 따라 다이어그램을 보다 더 간결하게 나타낼 수 있도록 하였다.

현재 세계적으로 50개가 넘는 BPMN 모델링 도구 벤더가 존재하고, 그 중에 대표적인 선두주자 벤더의 도구로 Pegasystems SmarBPM Suite, BEA AquaLogic BPM Designer, IBM WebSphere Business Modeler이 있고 국내에는 HandySoft의 BizFlow가 있다 [11-15]. 이러한 모델링 도구도 하위 프로세스의 개념을 통하여 줌인, 줌아웃의 기능을 수행하고 프로세스모델 이외의 조직모델(organization model)나 정보모델 (information model)와 같이 사전 정의된 모델 혹은 뷰를 보여줄 수 있지만, 예들 들어 고객만족도관점과 같이 사용자가 정의한 뷰에 맞게 다이어그램을 수직적으로 추상하는 기능은 갖고 있지 않다.

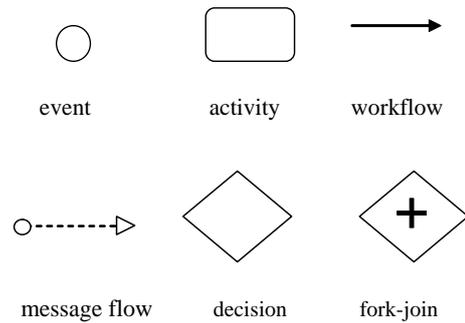
3. BPMN

이 절에서는 먼저 BPMN [5]의 다이어그램을 구성하는 기본적인 요소(element)들을 소개하고, 이들을 이용하여 프로세스 다이어그램을 기술하는 예를 보여준다. 또한 BPMN이 제공하는 수평적 추상을 설명한다.

3.1 BPMN의 기본 요소들

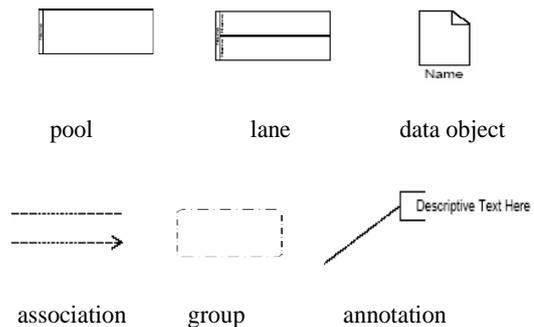
BPMN[5]은 프로세스 다이어그램 작성에 사용될 수 있는 많은 다양한 기호들을 정의하고 있으나, 프로세스 흐름을 나타내기 위한 가장 기본적인 기호의 유형은 다음의 6

가지이다:



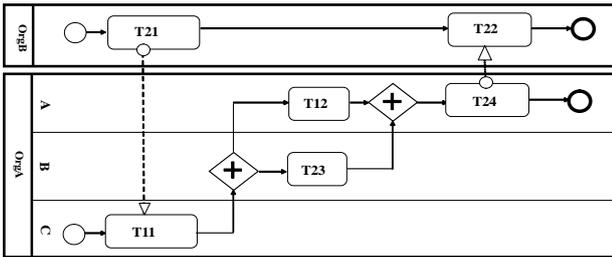
이러한 기호들을 사용하여 BPMN의 사용자는 프로세스를 나타내는 그래프(graph)를 그리게 되고, BPMN 기호로 그려진 프로세스는 노드(node)들과 이 들을 연결하는 에지(edge)들로 구성되게 된다. 위의 6 가지 기본적인 기호도 흐름을 연결하는 노드(decision, fork-join), 활동을 나타내는 노드(즉, 활동노드), 이벤트를 나타내는 노드(즉, 이벤트 노드)와 이 들 노드를 연결하는 에지(workflow, message flow)로 나뉘어 진다.

BPMN은 이 외에도 아래와 같이, 활동의 수행자(participant)를 나타내기 위한 구획면(swimlanes)의 기호로 풀(pool)과 레인(lane), 자료객체(data object) 기호, 자료객체와 프로세스 흐름을 연결하는 연관관계(association), 문서화나 분석을 목적으로 일 군의 객체(object)들을 묶는 그룹(group)기호, 그리고 프로세스 다이어그램의 독자를 위한 코멘트를 기술하기 위한 주석(annotation)의 기호를 기본적인 기호로 채택하고 있다.



3.2 BPMN을 사용한 프로세스 다이어그램

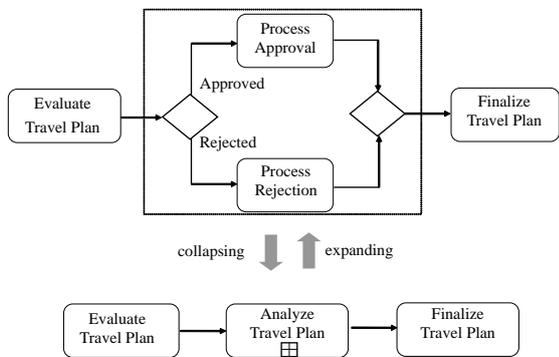
구획면으로 나타내는 수행자는 풀과 레인의 두 가지가 있다. 레인이 개별적인 수행주체를 나타내는데 반하여, 풀은 일 군의 레인들을 나타내는 개념으로, 조직 혹은 부서와 같은 수행주체이다. 업무의 흐름을 나타내는 흐름선(sequence flow)은 풀의 경계를 넘을 수 없고, 풀 간의 상호작용은 메시지(message)를 교환하여 이루어진다. 반면 동일한 풀의 레인들 안에 있는 노드(flow object)들 간에는 메시지가 교환될 수 없다. (그림 1)은 이러한 개념들을 사용한 프로세스 다이어그램의 예이다.



(그림 1) 수행자를 명시적으로 보여주는 프로세스 다이어그램

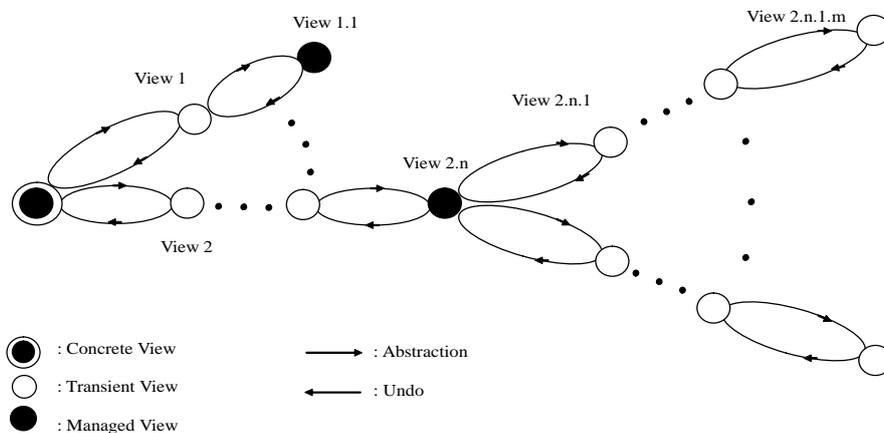
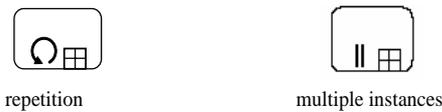
3.3 BPMN이 지원하는 추상

(그림 2)는 BPMN이 지원하는 슈퍼노드를 통한 수평적 추상의 예를 보여준다. 프로세스 다이어그램의 일부가 슈퍼노드로 추상화되고 슈퍼노드를 상세화하여 세부 다이어그램을 얻는 예를 보여준다.



(그림 2) 슈퍼노드로의 압축(collapsing)과 슈퍼노드로부터의 펼침(expanding)

BPMN은 반복(repetition) 혹은 복수 인스턴스(multiple instances)의 내부구조를 갖는 활동을 위한 특별한 기호를 다음과 같이 정의하고 있다:



(그림 3) 프로세스 다이어그램을 위한 추상연산과 각종 뷰 사이의 관계

4. BPMN의 추상능력의 확장

이 절에서는 수평적 추상을 위하여 슈퍼에지의 개념으로 BPMN을 확장하고, 수직적 추상을 위하여 관점 속성과 관심명세의 개념을 도입하여 BPMN을 확장한다.

4.1 상세 뷰와 추상뷰

뷰는 수평적 추상의 결과이거나 혹은 수직적 추상 즉 특정 관점으로부터 형성된 추상의 결과를 지칭하며, 상세 뷰(concrete view)와 추상뷰(abstract view)의 두 가지 종류로 나누어진다. 상세 뷰는 프로세스 다이어그램을 완전히 펼친 프로세스 다이어그램 즉 모든 활동들이 태스크인 프로세스 다이어그램을 말한다. 추상뷰는 상세 뷰의 추상이다. 추상뷰는 상세 뷰 혹은 추상뷰에 추상연산을 적용하여 얻어진다. 어떤 뷰에 도달하기 위하여, 반복적으로 추상연산들을 적용하여, 중간뷰(intermediate view)들을 얻는다. 중간뷰들은 원상복귀연산자(undo operation)을 적용하여 추상화 이전상태의 다이어그램이 회복될 수 있다. 이러한 관점에서 볼 때에, 중간뷰는 크게 임시뷰(transient view)와 관리대상뷰(managed view)로 나누어진다.

추상연산(혹은 은닉(hiding)연산)과 펼침(expansion)연산은 프로세스 다이어그램을 위한 도구의 뷰어(viewer)의 연산으로서 수행될 수 있다. (그림 3)에서 이중으로 된 원은 완전히 펼쳐진 (혹은 상세화된) 프로세스 다이어그램을 나타낸다. 이로부터 추상연산을 적용하여 추상적인 뷰 (예를 들어 View2)를 도출할 수 있다. 추상연산을 반복하여 적용함으로써, 추상의 정도가 높은 뷰(예를 들어 View2.n)를 도출할 수 있다. 어떤 뷰는 사용자가 높은 관심을 갖는 뷰로 저장한 후 다른 뷰들을 도출하기 위한 출발점으로 사용될 수 있다. 그러한 뷰를 관리대상뷰라고 부른다. 관리대상뷰가 아닌 뷰들은 임시뷰(transient view)라고 부른다. (그림 3)의 View2.n 는 관리대상뷰로서 View2.n.1, ..., View2.n.1.m과 같은 다른 뷰들을 도출하기 위한 중간시작점으로 사용되고 있음을 보여주고 있다. 관리대상뷰는 해당 관심명세와 함께 저장되어야 한다.

이와 같은 추상연산을 적용하기 위하여 활동은 다음의 속성들을 포함하여야 한다:

- (1) <<actor>> 속성
- (2) <<aspect: x>> 속성

BPMN에서 활동은 원자적인 활동이거나 다른 활동들로 구성된 복합적인 활동일 수 있으므로, 이러한 속성들은 슈퍼노드를 포함한 하나의 노드의 속성이 될 수도 있고 또는 활동 다이어그램 전체들의 속성이 될 수 있으며, 슈퍼에지가 하나의 복합적인 활동을 나타내므로 슈퍼에지의 속성이 될 수도 있다. 노드, 슈퍼노드, 활동 다이어그램, 슈퍼에지의 이 네 가지 경우가 다 활동을 나타내므로, 이들을 총칭하여 활동으로 부르기로 한다.

<<actor>> 속성은 수행자(actor)를 나타내기 위한 속성이며, 주어진 활동을 수행하는 개체로서, 조직, 조직의 부서, 조직의 직원 혹은 시스템이 그 예가 된다. BPMN에서는 풀과 레인으로 수행자를 간략하게 표시하는 기호를 제시하고 있다. 그러나 프로세스를 수행하는 조직이 복잡한 구조를 가질 경우, 이러한 기호는 확장성(scalability)에 제약을 갖게 되므로 활동의 수행자를 표시할 수 있는 독립된 일반적인 방법이 필요하며, 활동의 수행자를 활동의 <<actor>> 속성의 값으로 정의함으로써 이 문제를 해결할 수 있다. “<<aspect: x>>”에서 x는 사용자가 정의한 관점의 이름으로, 프로세스의 이해당사자(stakeholder)들의 관심의 초점 혹은 관점이다. 관점의 표현을 위하여 트리(tree)구조를 사용할 것을 본 연구는 제안한다. 실제 관점은 복잡한 트리 구조를 가질 수도 있고, 하나의 뿌리노드(root node)만 있는 단순한 구조의 트리일 수도 있다. <<actor>> 속성을 사용하지 않고 <<aspect: actor>>의 형식을 사용하여 <<actor>> 속성을 하나의 관점으로 볼 수도 있으나, <<actor>>속성은 매우 중요한 속성이므로 특히 BPMN의 구획면 기호와 연결되는 속성이므로 본 연구에서는 따로 표현할 것을 권장한다.

4.2 슈퍼노드와 슈퍼에지에 의한 수평적 추상

제 3.3절에서 BPMN이 반복과 복수 인스턴스의 내부구조를 가지고 있는 두 개의 활동을 위하여 특별한 기호를 갖고 있으며, 이러한 슈퍼노드가 수평적 추상의 메커니즘을 제공하고 있음을 보았다. 결정(decision)이나 분기-결합(fork-join)의 구조는 프로세스 다이어그램에 자주 사용되는 구조로서 본 연구에서는 이들을 위한 BPMN과 일관된 다음 2개의 새로운 기호를 추가할 것을 제안한다:



decision fork-join

(그림 4)의 위의 프로세스 다이어그램에서 점선 안에 있는 지역이 추상을 위하여 선택되었다고 하면, 추상의 결과를 (그림 4)의 아래와 같이 하나의 에지로 나타내질 수 있는데 이러한 에지를 슈퍼에지라고 부른다.

정의 1. (슈퍼에지) 슈퍼에지는 BPMN의 확장된 기호의 하나로, 확장되지 않은 BPMN의 에지와 같이 노드와 노드를 연결한다는 역할을 수행하나, 하나의 입구 에지(entry edge)와 하나의 출구 에지(exit edge)를 갖는 활동 다이어그램 혹은 활동 다이어그램의 일부를 감추는 기능을 갖는다.

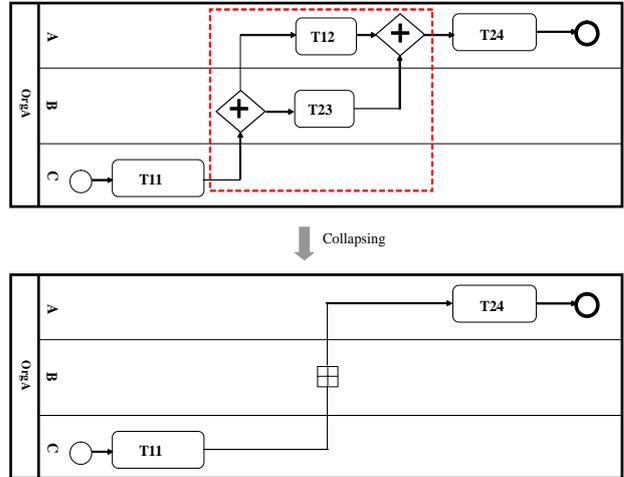
슈퍼에지를 사용하여 활동 다이어그램을 추상화 하는 연산을 이 논문에서는 ‘지역추상연산(region abstraction operation)’이라고 부르기로 한다.

정의 2. (수평적 추상연산) PDs가 프로세스 다이어그램들의 집합이면, 수평적 추상연산 $a^H: PDs \rightarrow PDs$ 은

$$a^H(PD,R,X) = PD[X/R]$$

으로 정의되며 여기서 PD는 프로세스 다이어그램이고, R은 하나의 입구 에지와 하나의 출구 에지를 갖는 PD의 하나의 지역이고, X는 새로운 슈퍼노드 혹은 슈퍼에지이고 PD[X/R]은 PD에서R을 X로 대치한 결과이다.

이 연산이 수행될 때, 추상된 세부사항은 슈퍼에지의 밑에 감추어진다. 슈퍼에지가 중첩되어(nested) 있지 않을 경우 슈퍼에지를 펼치면 해당 슈퍼에지의 자리에 단일한 입구 에지와 단일한 출구 에지가 있는 활동 다이어그램 혹은 그러한 활동 다이어그램의 부분이 들어서게 된다.



(그림 4) 추상을 위한 지역의 선택과 지역추상의 결과

슈퍼노드와 슈퍼에지는 지역을 구조적으로 추상하여 준다는 공통점을 갖고 있으며, 슈퍼노드로 가능한 모든 추상이 슈퍼에지로 가능하다. 이 둘 사이의 차이점은 첫째로 슈퍼노드는 하나의 활동으로서 이름을 갖는데 비하여, 슈퍼에지는 이름을 갖지 않는다는 것이다. 따라서 어느 지역(region)을 추상화하여 이름을 부여하고자 하면 슈퍼노드를 사용하여야 한다. 두번째 차이점은 레인으로 수행자가 표현된 프로세스 다이어그램의 경우, 슈퍼노드는 활동의 수행자가 레인의 소유자(owner) 즉 수행자와 일치하여야 하나, 슈퍼에지는 레인의 소유자와의 일치성이 불필요하다. 따라서 슈퍼에지로

가능한 모든 추상이 슈퍼노드로 가능한 것은 아니다.³⁾

4.3 관점 속성을 사용한 수직적 추상

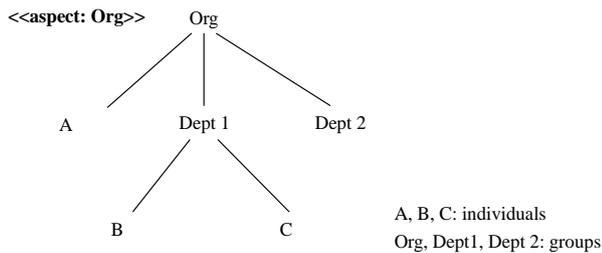
이 절에서는 <<aspect: x>>로 표현하는 활동의 속성을 사용하여, 사용자가 관심을 가질 수 있는 관점을 선언하고 또 선언된 관점 관점에서의 각 활동의 관점값을 명시적으로 기술할 수 있도록 하는 방법을 제시한다. 먼저 관점은 다음과 같이 정의될 수 있다.

정의 3. (관점) “<<aspect: x>> 속성”은 활동의 속성으로 그와 연결된 자료구조를 갖는다. 연결된 자료구조는

- (1) 열거형 타입(enumeration type)의 노드들
 - (2) 열거형 타입의 노드들로 구성된 트리 혹은
 - (3) 그러한 트리들의 집합(forest)
- 이 된다.

아래 예제 1은 관점이 <<actor>>인 예이다. 4.1절에서 설명한 바와 같이, 활동들의 수행주체들 간의 관계도 하나의 관점으로 볼 수 있다. 또한 조직을 구성하는 부서 혹은 직원들을 수행주체로 볼 때 이들의 관계를 하나의 관점으로 볼 수 있다. 예제 1은 이러한 관점을 선언하고 그 속성의 자료구조를 정의한 예이다.

예제 1. (<<actor>> 관점) 아래는 <<actor>> 관점인 Org 관점을 선언하고 그 자료구조를 정의한 예제이다. A, B, C 와 Org, Dept1, Dept2는 각각 individuals와 groups의 열거형 타입의 노드라는 선언과 정의 2에 의하여, Org 관점의 자료구조가 정의된다. 아래의 Org 노드를 뿌리로 갖는 트리는 이러한 자료 구조의 유형에 맞는 현재 값의 예 이다.



특정 관점을 나타내는 <<aspect: x>> 속성은 그것을 위한 자료구조를 프로세스 다이어그램과 독립적으로 먼저 정의함으로써 사용할 수 있게 된다. 프로세스 다이어그램에서는 각 활동 노드의 <<aspect: x>> 속성에 특정한 값을 할당함으로써 그 속성의 사용이 가능하게 된다.

예제 1에서는 관점 트리의 뿌리(root)의 이름을 관점 이

3) 슈퍼에는 UML 2.0의 상호작용 개괄 다이어그램에서 프레임유사하게, 다이어그램의 일부를 압축하여 보여준다.UML 활동 다이어그램에서도 복잡한 활동을 나타내는 노드를 통하여 본 논문의 슈퍼노드에 상응하는 추상을 허용하나, 추가적으로 상호작용 개괄 다이어그램을 통하여 필요에 따라 다이어그램을 보다 더 간결하게 나타낼 수 있도록 하였다.

름으로 그대로 사용하였다. 그러나 다른 이름을 관점의 이름으로 사용할 수도 있다.

품질(quality), 목표(goal) 그리고 가치(value) 와 같은 추상적인 개념들도 관점이 될 수 있다. 목표와 그 하위목표(subgoals) (그리고 다시 하위목표의 하위목표 등) 은 트리 구조를 가진 것으로 볼 수 있으며, 따라서 예제 2와 같은 목표 관점을 만들어 볼 수 있다.

예제 2. (<<goal>> 관점) 어떤 회사가 고객만족을 회사의 하나의 목표로 설정하였다고 하자. 그 회사는 그 목표를 달성하기 위하여, ‘제품의 품질(Quality of product)’, ‘고객요구사항’, 고객요구사항충족(Customer requirements compliance) 그리고 ‘정시배달(On-time delivery)’ 의 세 개의 하위목표를 달성해야 한다고 판단하였다. 이러한 결정은 다음과 같이 ‘고객만족(Customer Satisfaction)’이라는 관점으로 나타낼 수 있다:

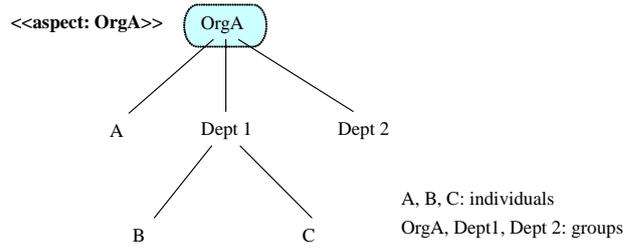
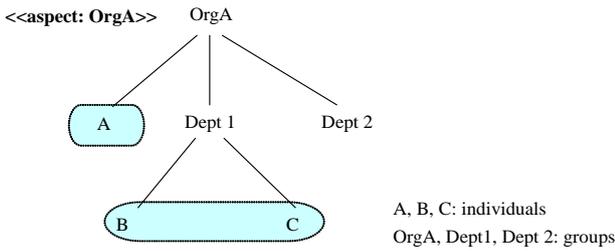


이렇게 속성을 통하여 각 활동들을 들여다 볼 때에, 그 활동의 관련된 관점 안에서의 (위치)값을 알고자 한다면, 이를 프로세스 다이어그램의 추상화에 활용할 수 있도록 사용자의 관심을 기술하여야 한다. ■

정의 4. (관심명세) 어떤 관점이 하나의 자료구조로 정의된 경우, 그 관점에 대한 사용자나 이해당사자의 관심을 명세할 수 있다. 이러한 특정 관점에 대한 관심을 기술한 결과를 “관심명세(interest specification)”라고 부른다.

앞서 설명한 바와 같이 수행자도 하나의 관점이고 BPMN에서는 구획면을 사용하여 초보적으로 <<actor>> 관점을 표현할 수 있게 하고 있다. 그러나 관심명세를 통하여 보다 정교하고 체계적으로 수행자들의 집합과 그들의 관계를 정의할 수 있고, 이를 바탕으로 유연한 추상과 상세화가 가능하게 된다.

예제 3. (관심명세) 관점 속성과 프로세스 다이어그램의 뷰들을 연결하기 위하여, 관점 데이터에 “관심(interest)”을 명세할 수 있다. 예를 들어, 다음의 관심명세는 사용자가 개인 A, B, C에만 관심이 있음을 나타낸다. 그리고, 이 특정한 관심의 관점에서 추상적인 뷰를 도출하고자 할 경우, 다른 세부사항은 감춘 채로 현재의 관심사항만을 보여주는 뷰를 얻을 수 있어야 한다.



(그림 6)는 위의 OrgA 관점을 (그림 1)에 적용한 결과이다.

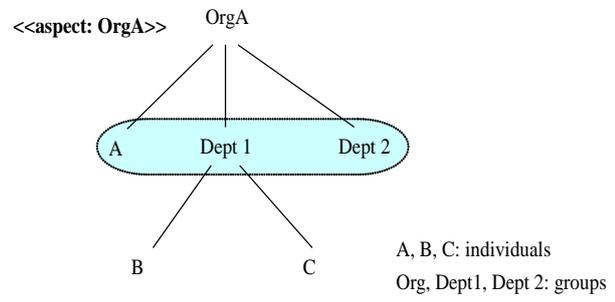
제2절의 (그림 1)의 프로세스 다이어그램은 OrgA 관점을 예제 3의 관심명세의 측면에서 본 결과이다.

정의 5. (수직적 추상연산) PDs가 프로세스 다이어그램들의 집합이면, 수직적 추상연산 α^V : PDs \rightarrow PDs 은

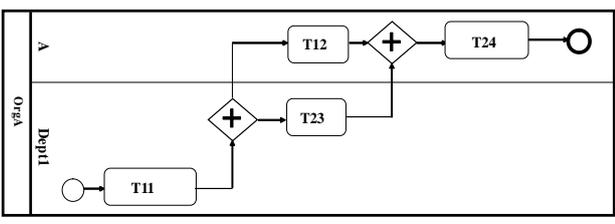
$$\alpha^V(PD, A, IS_A') = PD[IS_A(PD) \rightarrow IS_A']$$

으로 정의되며 여기서 PD는 프로세스 다이어그램이고, A는 관점, IS_A' 는 A관점의 새로운 관심명세, $IS_A(PD)$ 는 PD의 기존의 A관점의 관심명세이고, $PD[IS_A(PD) \rightarrow IS_A']$ 는 PD를 기존의 관심명세로부터 새로운 관심명세에 맞추어 변경한 결과이다.

예제 4. (관심명세) 아래와 같이 관점과 사용자 관심이 명세 되었다고 하자.

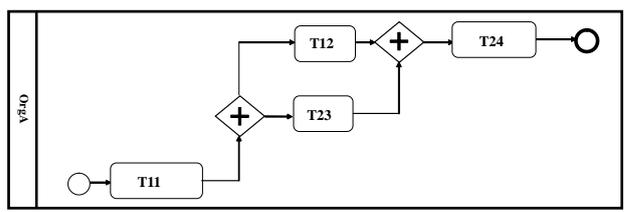


원하는 뷰는 개인으로서의 A만, 그룹으로서의 Dept1과 Dept2만을 포함할 것을 요구하고 있다. 이러한 관심명세를 정의할 경우, (그림 1)의 프로세스 다이어그램은 (그림 5)과 같이 추상되게 된다.



(그림 5) 예제 4의 관심명세를 (그림 1)의 OrgA에 적용한 추상

예제 5. (관심명세) 다음과 같이 OrgA 관점에 대한 관심명세가 정의되었다고 하자:



(그림 6) 예제 5의 관심명세를 (그림 1)의 OrgA에 적용한 추상결과

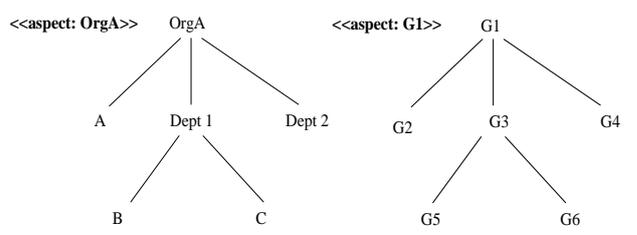
위의 예제들에서 볼 수 있는 바와 같이 수행자를 속성으로 보는 접근방법에 비교하여, 구현된 접근방법은 관심명세를 갖지 않으므로, 정교하고 복잡한 추상을 할 수 없다. 구현된 접근방법으로는 프로세스의 노드들에 대한 수행자들을 표시함으로써 수행자 관점의 제한된 추상만이 가능하다.

4.4 수직적 추상 연산의 순서 독립성

이 절에서는 수직적 추상연산의 적용이 적절한 조건이 성립될 경우 적용 순서에 관계없이 동일한 결과를 가져다 준다는 것을 보인다. 이를 위하여 먼저 일련의 개념들을 엄밀히 정의한다.

정의 6. (관점의 독립성) 두 개의 관점 사이에 어떠한 제약도 없을 경우 이 두 관점은 서로 독립적이라고 말한다.

예제 6. (관점의 독립성) 조직관점인 OrgA 관점과 목표관점인 G1이 선언되고 그 자료구조가 (그림 7)과 같이 정의되어 있다고 가정하자. (그림 7)의 관점 OrgA와 관점 G1 사이에 어떠한 제약도 없으므로 두 관점은 서로 독립적이다.4)



(그림 7) 두 개의 관점 OrgA와 G1의 자료구조 정의

4) 그러나 예를 들어 만일 Dept1이 G3를 요구한다면 OrgA와 G1는 서로로부터 비독립적인 관점들이 된다.

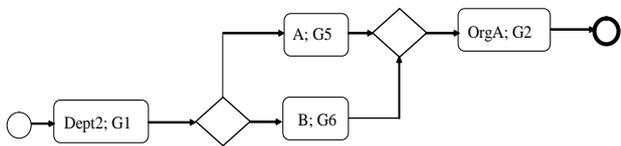
프로세스에 관심속성을 명세할 경우 주로 활동이나 태스크에 대하여만 관심의 대상이 되는 속성을 명시적으로 명세할 수 있다. 또한 구획면을 사용하여 다이어그램이 표현될 수도 있다. 수직적 추상연산의 순서독립성을 증명하기 위하여 먼저, 프로세스의 노드들의 속성을 명시적으로 일정하게 표현한 프로세스 다이어그램으로의 변환이 먼저 필요하다.

정의 7. (관심관점에서 완전히 명세한 프로세스 다이어그램)

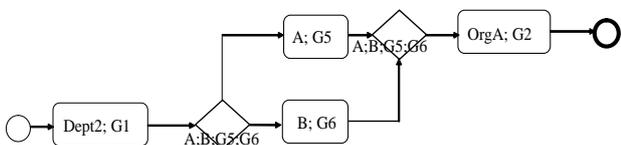
3.1절에서 제시된 BPMN 의 기본적인 기호에 대하여 따라 주어진 프로세스 다이어그램을 *관심관점에서 완전히 명세한 프로세스 다이어그램*은 주어진 프로세스 다이어그램의 노드들의 속성을 다음의 조건에 맞게 기술한 프로세스 다이어그램이다:

- 1) 구획면 대신 <<actor>>관점의 속성으로 참여자를 표시한다.
- 2) 프로세스의 흐름에 관련된 decision과 fork-join의 노드를 포함한 모든 노드들에 대하여 속성들을 명시적으로 명세한다.

예제 7. 아래 다이어그램은 (그림 7)의 두 개의 관점에 대하여 속성값이 할당된 프로세스 다이어그램이다. 참여자 기호가 사용되지 않고, <<actor>> 관점은 하나의 관점으로 표현되고 있다. 노드 안에 이 속성값을 표시하여 속성을 쉽게 알아볼 수 있게 하였다. 여러 개의 속성을 갖는 노드의 경우 속성값들 사이에 “;” 를 넣었다.



이 다이어그램에 decision에 대한 속성집합을 (그림 8)과 같이 기술함으로써 관심관점에서 완전히 명세한 프로세스 다이어그램을 얻게 된다.



(그림 8) 관심관점에서 완전히 명세된 프로세스의 예

정의 5의 수직적 추상연산을 프로세스 다이어그램에 적용하기 위하여는 한 스텝의 수직적 추상연산의 적용에 해당되는 다이어그램의 변형(transformation)이 이루어지고 이 변형을 엄밀하게 수행하기 위하여 변형규칙들(transformation rules)이 필요하다.

수직적 추상연산을 위한 변형규칙

P 규칙. 관심명세보다 추상수준이 낮은 속성은 삭제한다.

N 규칙. 노드의 속성집합이 공집합인 경우 그 노드와 그 노드의 입구예지와 출구예지를 하나의 슈퍼예지로 대체한다.

R 규칙. 지역 안의 모든 노드의 속성집합이 공집합이며 하나의 입구와 하나의 출구를 갖는 지역으로 가능한 최대지역을 슈퍼예지로 대체한다.

수직적 추상 연산의 적용은 먼저 P규칙을 적용하고, 이어 그 결과로서 속성집합이 공집합인 노드들에 대하여 그 개수 만큼 N규칙을 적용하고, 이어 그 결과로서 얻어진 다이어그램에서 R규칙의 적용조건을 만족시키는 지역들에 대하여 모두 R규칙을 적용하는 순서로 이루어진다.

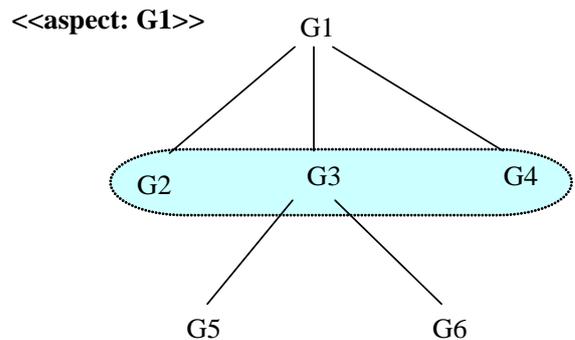
정리 1. α^v_1, α^v_2 가 두 개의 수직적 연산으로 서로 독립적인 관점의 수직적 추상연산인 경우 $\alpha^v_1 \circ \alpha^v_2 = \alpha^v_2 \circ \alpha^v_1$, 즉 두 연산의 적용결과는 적용순서에 관계없이 동일하다.

(증명) 관심관점에서 완전히 명세된 프로세스 다이어그램에 적용되는 수직적 연산을 위한 변형규칙은 P규칙의 적용에 이어 속성집합이 공집합인 노드들에 적용되는 일련의 N규칙과 이러한 일련의 N규칙의 적용에 의하여 하나의 입구예지와 출구예지를 갖는 지역에 대하여 적용되는 R규칙의 적용으로 완결된다. 이제 두 번의 수직적 연산의 적용을 고려하면 두 번의 P규칙의 적용의 결과로서 노드의 속성집합은 각 노드 별로 동일하다. 나아가, 노드의 속성집합이 공집합이 되는 노드들도 동일하므로 두 차례의 일련의 N규칙이 적용에 의하여 슈퍼예지로 대체되는 노드들의 집합도 동일하다. 그러므로 앞의 P규칙과 N규칙의 적용 후에 속성집합이 공집합이 아닌 임의의 두 노드 간의 최대 지역이 모두 속성집합이 공집합인 경우도 두 번의 수직적 연산의 적용 순서에 관계없이 동일하므로 두 번의 수직적 연산의 적용은 순서에 관계없이 동일한 프로세스 다이어그램을 주게 된다. ■

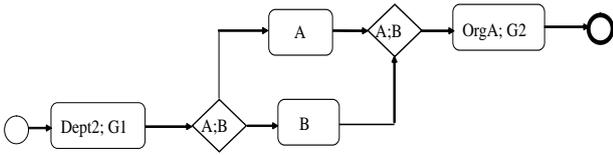
예제 9. { G1- α^v_1 , OrgA- α^v_1 } 연산의 적용.

(그림 8)의 프로세스 다이어그램에 다음과 같이 <G1- α^v_1 , OrgA- α^v_1 >의 순서로 먼저 추상연산을 적용한다:

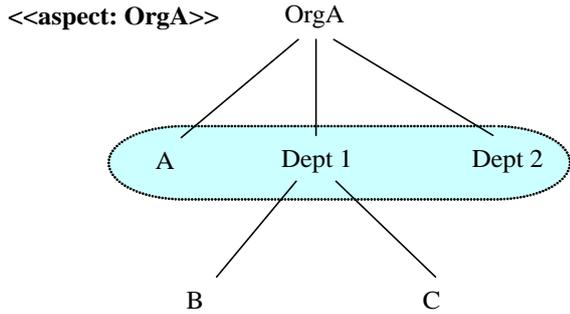
1-1) G1- α^v_1 은 다음의 관심명세를 (그림 8)에 적용하는 연산이다:



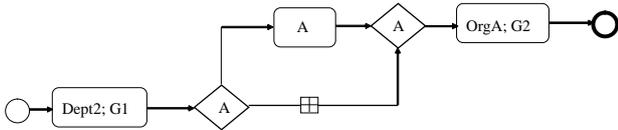
P 규칙이 해당되며 그 적용결과는 다음과 같다:



1-2) OrgA- α^v_1 은 다음의 관심명세를 위의G1- α^v_1 의 결과에 적용하는 연산이다:

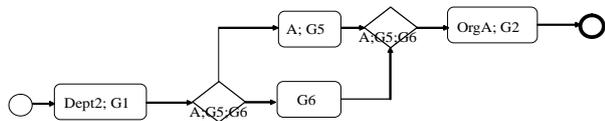


먼저 P 규칙이 해당되고, 그 결과 N 규칙이 적용 가능하다. N 규칙을 적용한 결과는 다음과 같다:



이제 (그림 8)의 프로세스 다이어그램에 앞에서 적용한 두 개의 추상연산들을 순서를 바꾸어 <OrgA- α^v_1 , G1- α^v_1 >의 순서로 적용한다. 각 연산의 적용 시 사용하는 관심명세는 위에서 각 연산을 적용할 때의 관심명세와 동일하다.

2-1) OrgA- α^v_1 을 적용하면 P규칙이 적용 가능하여 다음과 같은 뷰가 얻어진다:



2-2) G1- α^v_1 을 적용하면 P 규칙이 적용가능하고, 그 결과에 N 규칙이 적용 가능하다. 최종적인 결과는 1-2의 결과와 같다. 따라서 <G1- α^v_1 , OrgA- α^v_1 >와 <OrgA- α^v_1 , G1- α^v_1 >의 적용결과가 동일하다. ■

정리 1은 두 개의 서로 다른 관점의 수직적 추상연산의 적용이 순서독립적임을 보였다. 정리 1에 의하여 일반적으로 서로 다른 n개의 관점의 수직적 추상의 경우 추상연산의 적용순서에 관계없이 같은 결과를 얻게 됨을 알 수 있다.

정리 2. $\alpha^v_1, \dots, \alpha^v_n$ 가 서로 독립적인 관점의 n개의 수직적

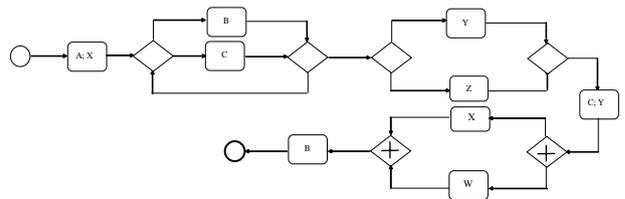
연산인 경우 이 들 n개의 연산의 적용결과는 적용순서에 관계없이 동일하다.

5. 추상연산의 적용 예

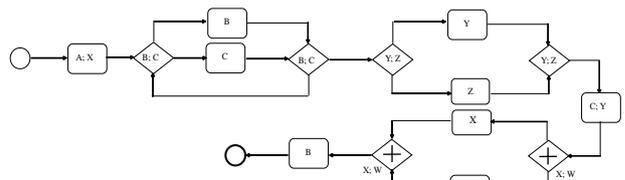
이 절에서는 앞에서 제시한 BPMN의 추상연산들의 추가적인 적용 예제를 제공한다. 5.1절에는 수직적 추상연산이 적용순서에 관계없이 일관된 결과를 주는 추가적인 예를 보이고, 5.2절에서는 추가적인 수직적 추상의 예, 5.3절에서는 수평적 추상의 예를 보여준다.

5.1 일련의 수직적 추상연산을 서로 다른 순서로 적용한 추가적인 예

아래는 두 개의 관점 P1과 P2의 속성들을 갖고 있는 프로세스이다. P1의 속성의 집합을 {A, B, C}라고 하고 P2의 속성의 집합을 {W, X, Y, Z}이라고 가정한다.

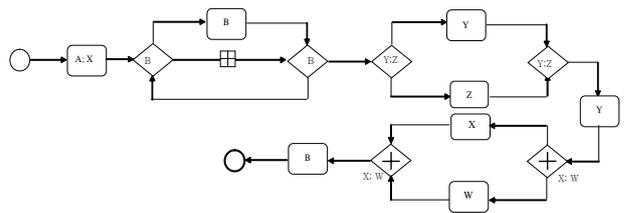


P1 관점의 관심명세가 $I1 = \{A, B\}$, P2 관점의 관심명세가 $I2 = \{X\}$ 인 경우 위의 프로세스를 이들 관심관점에서 완전히 명세한 프로세스는 아래와 같다5):



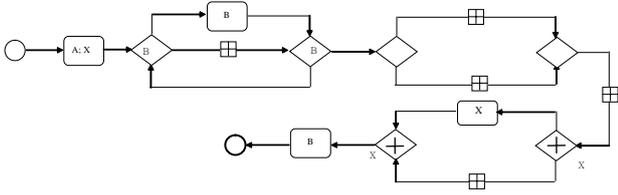
(그림 9) I1 과 I2의 관심관점에서 완전히 명세된 프로세스

P1 관점의 관심명세와 P2 관점의 관심명세가 예제 7과 같은 경우, I1의 적용을 α^v_1 , I2의 적용을 α^v_2 라고 하자. 먼저 α^v_1 을 적용하면 P규칙이 적용 가능하여 C 속성이 보이지 않게 되고, 그 결과에 N규칙이 적용 가능하다. 그 결과는 다음과 같다:

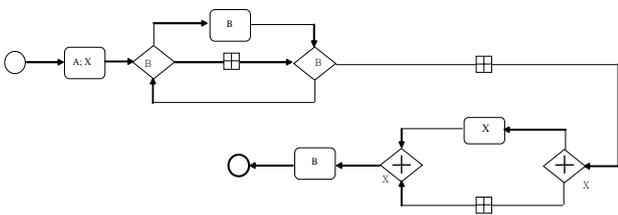


5) Fork-join의 경우 노드 옆에 표다

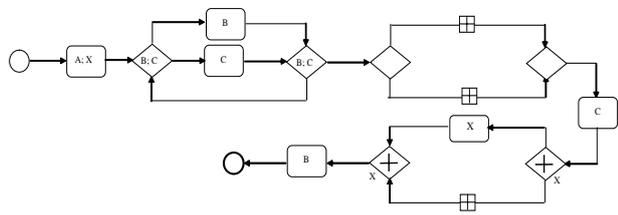
이제 α^V_2 를 적용하면 P규칙이 적용되고, 그 결과 5개의 노드에 대하여 N규칙이 적용 가능하게 된다. 이를 적용한 결과는 다음과 같다:



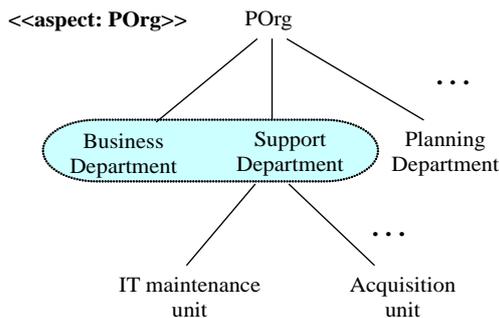
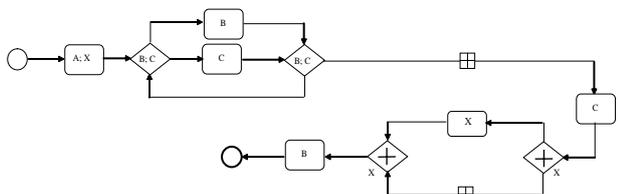
이 뷰에는 R규칙이 적용가능하며, 그 결과는 다음과 같다:



이제 (그림 9)의 프로세스 다이어그램에 앞에서 적용한 두 개의 추상연산들을 순서를 바꾸어 적용한다. 먼저 α^V_2 를 적용하면 P규칙이 적용 가능하다. P규칙을 적용한 결과에 3군데에서 N규칙의 적용이 가능하며, 그 결과는 다음과 같다:



이 뷰에는 R규칙이 적용가능하며 그 결과는 다음과 같다:



이제 α^V_1 을 적용하면 P규칙이 적용되며, 그 결과에 N규칙이 한번 적용 가능하다. 최종적인 결과는 앞서 수직적 연산의 순서를 바꾸어 적용한 결과와 동일하다.

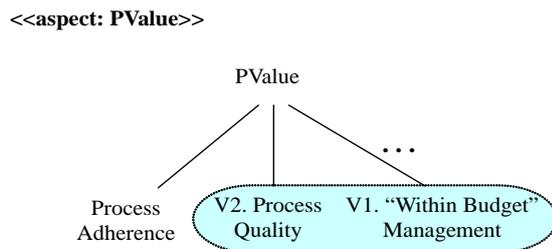
5.2 수직적 추상의 추가적인 예

이 절의 예는 P 회사의 유지보수 프로세스에 기반하고 있다. (그림 11)는 이 프로세스를 보여준다. P회사의 조직은 (그림 10)의 POrg라는 관점에서 보이는 것처럼 모델링 된다. 또한 (그림 10)의 PValue라는 관점에서 보이는 것처럼, P회사가 추구하는 기업의 가치로 업무 프로세스 준수(process adherence), 품질 높은 업무 프로세스(process quality), 예산 범위에 맞는 관리(“Within Budget” management)등이 있다. P회사의 유지보수 업무 프로세스는 (그림 11)에 정의되어 있다. (그림 11)의 프로세스의 세 참여자는 (그림 10)의 POrg 관점에 나타나 있다.

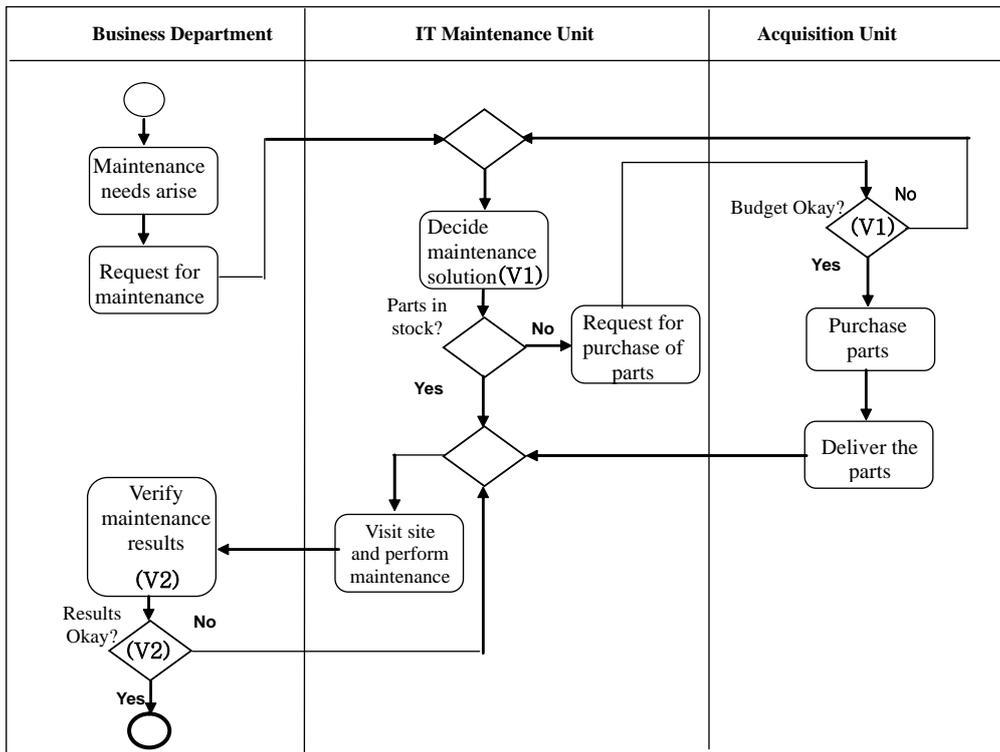
먼저 (그림 11)의 프로세스를 POrg관점에서 (그림 10)에 표시된 관심명세처럼 Business Department와 Support Department 로 보면, (그림 12)과 같이 IT Maintenance Unit과 Acquisition Unit의 두 개의 참여자가 Support Department 라는 하나의 참여자로 합쳐지고, 그 나머지는 전과 동일한 다이어그램으로 변환되게 된다. PValue 관점에서 (그림 11)은 ‘예산범위에 맞는 관리(“Within Budget” management value)’ 가치에 관련된 다이어그램 요소를 (V1)로 표시하고, ‘높은 품질을 추구하는 업무 프로세스(Process quality value)’ 가치에 관련된 요소를 (V2)로 나타내고, ‘프로세스 준수(Process Adherence)’의 가치는 프로세스 전체에 대한 준수를 요구하는 상위수준의 가치이므로 어느 특정 다이어그램 요소에 나타나지 않았다. (그림 10)에서 PValue 관점의 관심명세가 {Process Adherence, “Within Budget” management value}이므로 (V1)과 (V2)의 두 가지 중 어느 가치에도 해당되지 않는 다이어그램의 요소는 슈퍼에지로서 추상한다. (그림 12)은 이와 같이 POrg 와 PValue 관점의 관심명세들에 따라 (그림 12)을 추상한 결과를 보여주고 있다.

5.3 수평적 추상의 추가적인 예

이 절의 예는 [5]에 나오는 이메일투표의 예에 기반한다. (그림 13)는 이메일투표 프로세스를 보여준다. 이 프로세스는 참여자가 이메일투표 관리자 하나이고, 투표를 하는 사람들은 외부 참여자로 프로세스 다이어그램에는 명시적으로

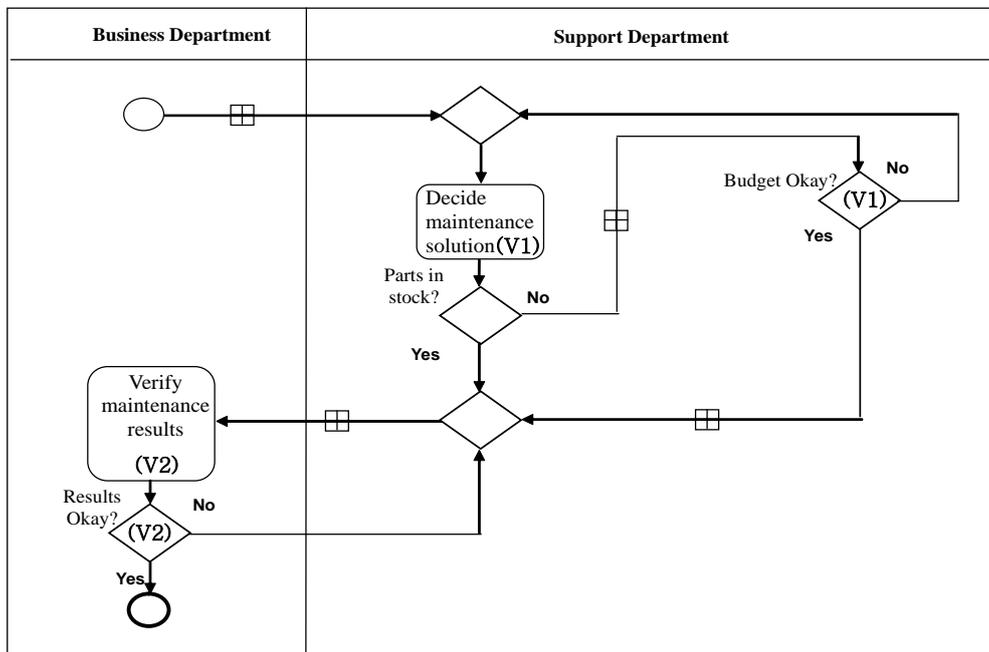


(그림 10) P회사의 조직도와 새로운 뷰를 위하여 제시된 새로운 관심명세



(V1): "Within Budget" management value
 (V2): Process quality value

(그림 11) P회사의 유지보수 프로세스

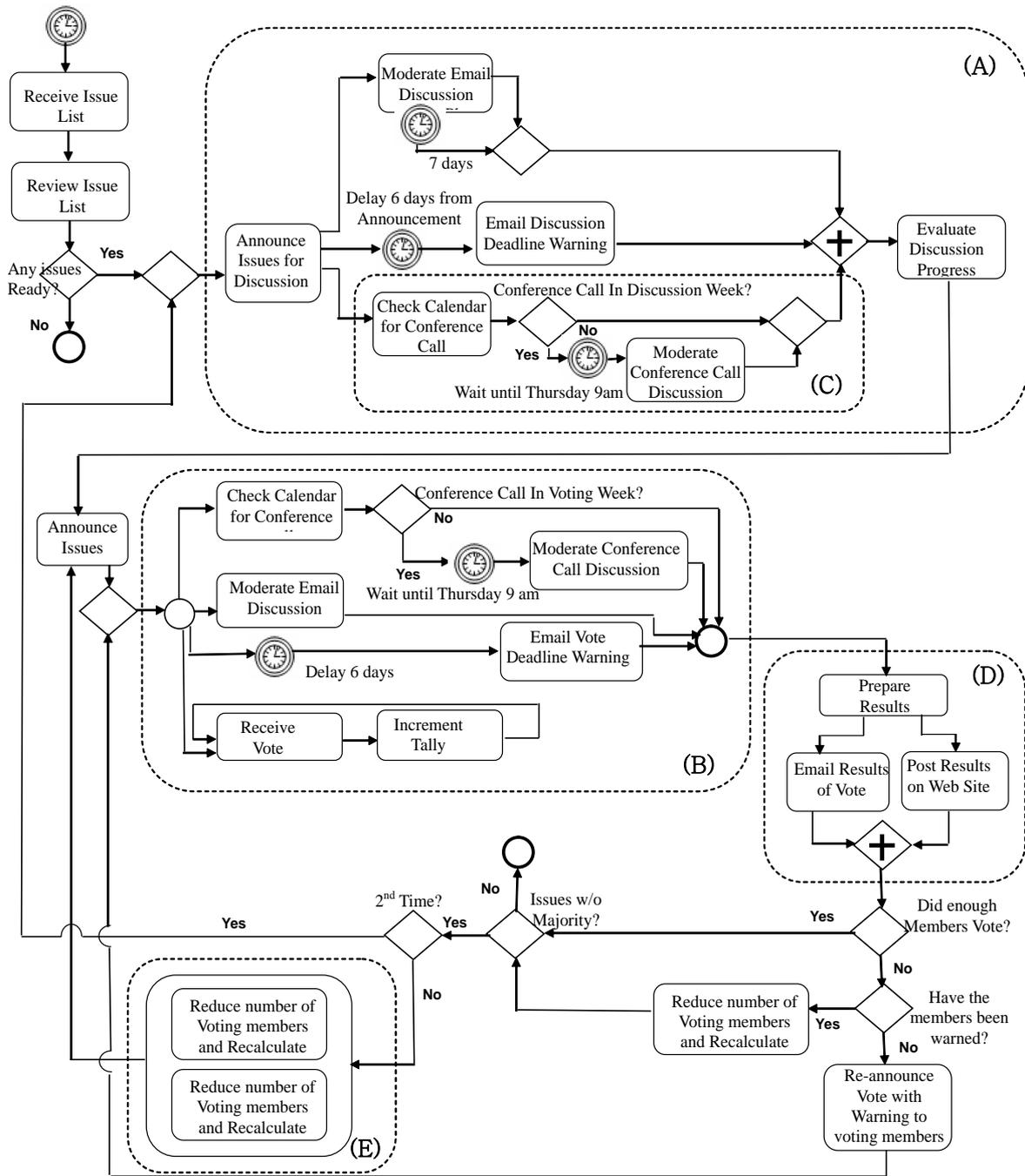


(그림 12) (그림 11)의 프로세스로부터 (그림 10)의 관심명세들을 적용하여 도출한 프로세스

나타나지 않는다. (그림 13)의 프로세스에서 수평적 추상이 되는 다섯 개의 지역을 (A)~(E)로 나타내었다.

(그림 13)에서 (A)와 (B)는 프로세스의 세부내용이 관심을 가져야 할 대상이므로 구체적인 명칭을 가진 슈퍼노드로 추상화 할 대상이고, (C), (D), (E)는 추상화 대상이지만 특

별히 세부내용에 대한 명칭을 줄 대상이 아니어서 슈퍼노드로 추상화 할 대상이라고 가정한다. 이에 따라 추상을 적용한 결과가 (그림 14)에 있다. 지역 (C)은 지역 (A)안에 들어 있으므로, (A)가 추상화 되기 전에 먼저 추상화 되어야 하고 따라서 그림 14에는 보이지 않는다. 그러나 (그림 14)에



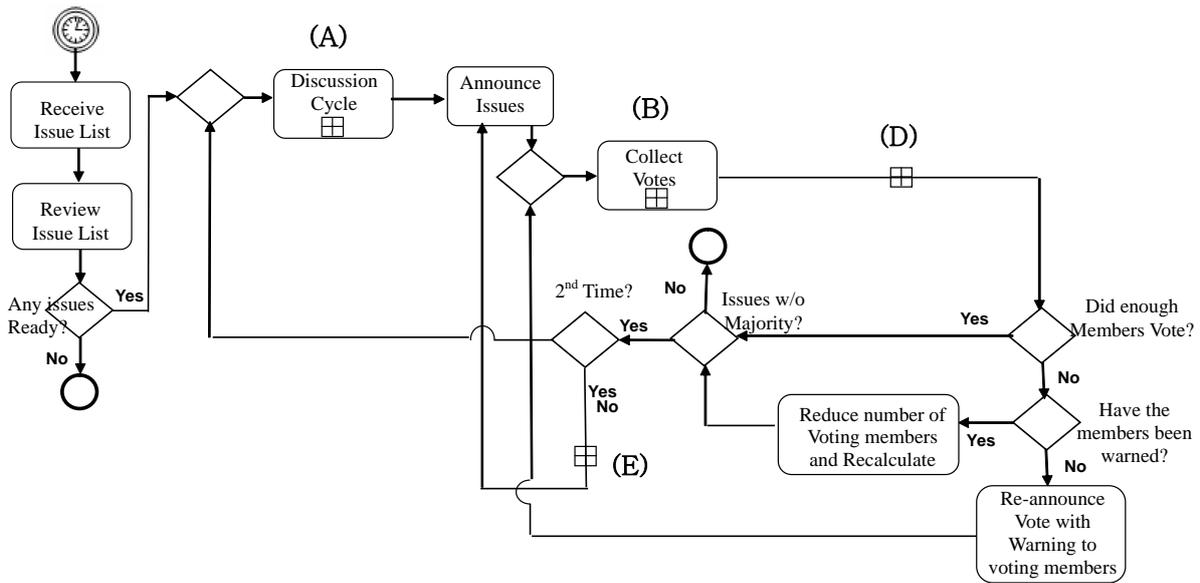
(그림 13) 이메일 투표 프로세스

서 Discussion Cycle 활동을 줄여주면 (그림 13)의 (A)와 같지만 (C)지역만 슈퍼페이지로 변경된 세부 다이어그램이 보이게 되고 여기서 다시 (C)를 줄여줌으로써 (그림 13)의 (C)에서 보이는 것과 같은 세부 하위프로세스가 보이게 된다.

6. 결론

모델링 언어의 수평적 추상기능은 대상의 크기가 크던 작던 관계없이 모델링이 가능하게 해 주고, 수직적 추상기능은 이미 구축된 모델을 특정한 관점에 맞추어 그 단면을 들

여다 보게 함으로써 모델링 언어에 분석의 능력을 가져다 준다. 현재의 BPMN 언어는 복잡한 활동을 나타내는 노드 (즉, 슈퍼노드)를 통하여 수평적인 추상을 제공하고 있으나, 다양한 이해당사자들이 자신들의 다양한 관점에 따라 가질 수 있는 추상화된 프로세스 뷰를 제공하지 않고 있다. 본 논문에서는 슈퍼페이지의 개념을 도입하여 BPMN이 제공할 수 있는 수평적인 추상의 범위를 확장하였다. 본 논문에서 제한한 슈퍼페이지는 UML 2.0의 상호작용 개괄 다이어그램이 이미 갖고 있는, 다이어그램의 일부를 압축하여 보여주는 프레임 기능에 비견할 수 있는 추상 방법이다. 또한 본 논



(그림 14) (그림 13)의 프로세스에 수평적 추상을 적용한 결과의 예

문에서는 이해당사자의 관심을 나타내는 관점 자료구조와 특정이해당사자의 현재의 관심을 기술하는 관심명세의 개념을 도입하여 수직적인 추상의 체계를 BPMN에 도입하였다. 이러한 확장된 BPMN의 유연한 추상 능력은 BPM 기반으로 업무를 수행하는 기업에서 BPMN이 비즈니스 프로세스의 모델링과 자동화된 실행을 위한 역할뿐 아니라, 다양한 분석을 할 수 있도록 지원하여 BPMN을 더욱 실용적인 비즈니스 프로세스 모델링 언어로 만들어 주고 BPMN이 비즈니스 프로세스의 자동화에 더욱 크게 기여할 수 있도록 만들어 줄 것이다.

향후 연구 방향으로 비즈니스 프로세스 모델링에 적절한 언어로서 BPMN과 UML2.0의 활동다이어그램이 갖는 유사성[16]을 고려하여 본 연구를 UML 2.0의 활동다이어그램에 적용하고 확장하는 연구가 필요하다. 또한 추상화 연산들에 대한 대수(algebra)의 개발과 추상화연산이론에 근거한 도구의 개발이 요구된다. 추상화 연산 대수의 개발은 사용자가 추상연산의 성질을 이해하여 원하는 뷰를 추상연산의 합성으로 표현하도록 하여주고, 동시에 도구가 이를 지원함으로써 일관성 있고 사용자의도에 맞는 뷰를 보여줄 수 있게 하여 준다. 관계형 데이터베이스를 위한 쿼리(Query) 언어의 역할과 같이 도구가 대수적인 성질을 이용하여 최적화된 연산을 통하여 사용자가 원하는 뷰를 효율적으로 제시할 수 있게 된다.

참 고 문 헌

[1] Smith, H., and Fingar, P., BPM. The Third Wave, Tampa: Meghan-Kiffer Press, 2003.
 [2] Krafzig, D., Banke, K., and Slama, D., Enterprise SOA: Service-Oriented Architecture Best Practices, Prentice-Hall, 2005.
 [3] Wolf, C., and Harmon, P., "The State of Business Process

Management," BPTrends Report, www.bptrends.com, 2008.
 [4] OASIS Standards Body, Web Services Business Process Execution Language Version 2.0, http://www.oasis-open.org/apps/org/workgroup/wsbpel/, May, 2006.
 [5] Object Management Group (OMG), Business Process Modeling Notation (BPMN) Version 1.1, January, 2008.
 [6] Clarke Jr., E. M., Grumberg, O., and Peled, D. A., Model Checking, MIT Press, 1999.
 [7] Harel, D., "Statecharts: A visual formalism for complex systems," Science of Computer Programming Vol.8, pp.231-274, North-Holland, 1987.
 [8] Object Management Group, OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2, November 2, 2007.
 [9] Dumas, M., and ter Hofstede, Arthur H.M., "UML Activity Diagrams as a Workflow Specification Language," Proc. 4th International Conference on the Unified Modeling Language (UML), Toronto, Canada, 2001.
 [10] Erl, T., Service-Oriented Architecture: Concepts, Technology, and Design, Pearson Education, Aug., 2005.
 [11] Hill, J. B., Cantara, M., Deitert, E., and Kerremans, M., "Magic Quadrant for Business Process Management Suites, 2007," Gartner Core Research Note G00152906, 14, December, 2007.
 [12] Pegasystems, Inc., "Pegasystems SmartBPM Suite Version: 4.2," BPTrends: The 2007 EA, Process Modeling & Simulation Tools report-2.1, 2007.
 [13] BEA Systems, Inc., "BEA AquaLogic BPM Suite Version: 5.7," BPTrends: The 2007 EA, Process Modeling & Simulation Tools report-2.1, 2007.
 [14] IBM "WebSphere Business Modeler, Ver.6.0.1," BPTrends:

The 2007 EA, Process Modeling & Simulation Tools report-2.1, 2007.

- [15] HandySoft Global Corporation, "BizFlow Version 1.0" BPTrends: The 2007 EA, Process Modeling & Simulation Tools report-2.1, 2007.
- [16] Russell, N., Aalst, Wil M. P., ter Hofstede, A., and Wohed, P. "On the Suitability of UML 2.0 Activity Diagrams for Business Process Modelling," The 3rd Asia-Pacific Conference on Conceptual Modelling (APCCM 2006), Vol.53 CRPIT, pp.95-104, Hobart, Australia, 2006.



강성원

e-mail : kangsw@icu.ac.kr
 1982년 서울대학교 사회과학대학(정치학사)
 1989년 Univ. of Iowa 전산학(전산학석사)
 1992년 Univ. of Iowa 전산학(전산학박사)
 1993년~2001년 한국통신 연구개발본부 선임 연구원

1995년~1996년 미국 국립표준기술연구소(NIST) 객원연구원
 2001년~2009년 2월 한국정보통신대학교 부교수
 2003년~현재 미국 Carnegie-Mellon University 소프트웨어공학 석사과정 겸임교수
 2009년 3월~현재 KAIST 정보과학기술대학 부교수
 관심분야: 소프트웨어 아키텍처, 소프트웨어 분석, 소프트웨어 시험, 형식기법



이 단 형

e-mail : danlee@icu.ac.kr
 1971년 서울대학교 공과대학(원자력공학 학사)
 1983년 미국 Arthur D. Little(경영과학 석사)
 1990년 미국 Virginia Commonwealth Univ. (정보시스템 박사)

1972년~1996년 KIST/SERI 연구원
 1997년~1998년 ETRI 연구원
 1999년~2000년 LGCNS 부사장
 2000년~2003년 한국소프트웨어진흥원 원장
 2003년~2009년 2월 한국정보통신대학교 교수
 2004년~현재 재 미국 Carnegie-Mellon University 소프트웨어공학 석사과정 겸임교수
 2009년 3월~현재 재 KAIST 정보과학기술대학 교수
 관심분야: 소프트웨어 요구공학, 소프트웨어 프로세스, 소프트웨어 프로덕트라인



안유환

e-mail : ywahn@handypmg.co.kr
 1984년 서울대학교 산업공학과(공학사)
 1986년 한국과학기술원 경영과학(공학석사)
 2000년 한국과학기술원 테크노경영대학원 경영공학(공학박사)
 1986년~2000년 한국전자통신연구원 책임 연구원

2000년~2007년 (주)핸디소프트 글로벌 CTO / 사장
 2007년~현재 재 BPM Korea Forum 회장
 2008년~현재 재 (주)핸디피엠지 CEO/사장
 관심분야: 소프트웨어 프로세스 개선, 비즈니스 프로세스 관리, 소프트웨어 품질관리