

An Service oriented XL-BPMN Metamodel and Business Modeling Process

Chee-Yang Song^{*} · Eun-Sook Cho^{**}

ABSTRACT

The business based existing BPMN model is a lack of service oriented modeling techniques. Therefore, it requires a layered technique of service oriented business modeling so that can meet the design for a complex application system, developing a system based on SOA. In order to enhance reusability and modularity of BPMN business model, this paper proposes a metamodel and business modeling process based on this metamodel that can hierarchically build a BPMN model. Towards this end, the XL-BPMN metamodel hierarchically established based on MDA and MVS styles are first defined. Then a BPMN service modeling process is constructed based on modeling elements of this metamodel according to the modeling phases. Finally, the result of a case study in which the proposed method is applied to an online shopping mall system is discussed. With the use of well-defined metamodel and modeling process, it is hoped that it can be shown that a service dominated and layered BPMN business model can be established, and that the modularity and reusability of the constructed BPMN business model can be maximized.

Keywords : Business Process, BPMN, Metamodel, MDA, MVC, Service Oriented Modeling, Layered Modeling

서비스 지향 XL-BPMN 메타모델과 비즈니스 모델링 프로세스

송치양^{*} · 조은숙^{**}

요 약

업무에 초점을 둔 BPMN 모델은 서비스 지향 비즈니스 모델링에 한계점을 안고 있다. 이에, SOA 기반의 시스템 개발을 위해 서비스 수준에서 복잡한 응용 시스템을 디자인할 수 있는 서비스 지향의 비즈니스 모델링 기법이 요구된다. 본 논문은 BPMN 비즈니스 모델의 모듈성과 재사용성을 향상시키기 위해, 서비스 지향으로 BPMN 모델을 계층적으로 구축할 수 있는 메타모델과 이에 기반한 비즈니스 모델링 프로세스를 제시한다. 이를 위해, BPMN 모델의 모델링 요소들을 MDA와 MVC 패턴에 의해 계층화하고 서비스 지향의 모델링을 위한 확장된 XL-BPMN 메타모델을 정의한다. 이 메타모델의 모델링 요소를 이용해서, 모델링 단계에 의한 BPMN 서비스 모델링 프로세스를 정립한다. 이 기법은 온라인쇼핑몰시스템의 비즈니스 모델링에 사례로서 적용되고 평가되어진다. 잘 정의된 메타모델과 모델링 프로세스를 통해, 서비스 중심의 BPMN 비즈니스 모델을 생성할 수 있고, BPMN 모델의 모듈성을 증진시킬 수 있으며, 또한 독립적으로 재사용 가능한 모델의 수를 극대화할 수 있다.

키워드 : 비즈니스 프로세스, BPMN, 메타모델, MDA, MVC, 서비스 지향 모델링, 계층적 모델링

1. 서 론

최근 응용 시스템의 개발은 서비스 지향의 SOA(Service-Oriented Architecture)[1-5]에 기반하는 추세이다. 통상, 어플리케이션의 개발은 비즈니스 모델링, 서비스 모델링, 그리고 컴포넌트 모델링을 거쳐 실제 구현의 개발 프로세스를

수행하게 된다. 따라서, 상위 비즈니스 서비스의 프로세스를 어떻게 구성하느냐에 따라 사용되는 하부의 컴포넌트 수가 달라 질 수 있다. 즉, 최적의 서비스 프로세스를 설계해야 최소의 컴포넌트로 시스템을 구축할 수 있다. 그래서, 복잡한 시스템의 비즈니스 프로세스를 모델링할 수 있고, 최적화된 비즈니스 모델을 설계할 수 있는 체계적이고 구체적인 지침을 제공하는 서비스 중심의 비즈니스 모델링 프로세스가 필요하다.

대상 도메인의 비즈니스 프로세스 디자인을 위한 대표적인 모델링 도구인 BPMN(Business Process Modeling Notation)[6-7] 모델은 OMG(Object Management Group)의 표준으로서 비즈니스 프로세스를 디자인 할 수 있는 표준화

※ 이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2012002161).

* 정 회 원: 경북대학교 컴퓨터정보학부 부교수

** 정 회 원: 서울대학교 컴퓨터소프트웨어학과 부교수

논문접수: 2012년 11월 19일

수정일: 1차 2013년 1월 4일

심사완료: 2013년 1월 4일

* Corresponding Author: Eun-Sook Cho(escho@seoil.ac.kr)

된 그래픽 표기법이다. BPMN은 SOA 표현을 위한 최적의 표현기법이라 평가받고 있다. 기존, BPMN을 이용한 비즈니스 모델링 방법들[8-9]은 그 사용법과 적용 사례는 풍부하나, 업무에 초점을 둔 비즈니스 프로세스 모델링의 한계점을 갖는다. 이에, SOA 기반의 복잡한 서비스 시스템을 디자인할 수 있도록 서비스 차원에서 BPMN의 계층적 모델링 접근에 대해서는 구체적인 가이드를 제시하고 있지 않다. 단순히, activity의 process가 sub-process를 갖는 개념하에, 상위 BPMN 모델이 하부 BPMN 모델을 복합(composition) 관계로서 표현한다. 따라서, 개발 단계의 추상화 수준에 따라 BPMN 모델을 모델링할 수 있는 계층적 모델링을 제공하지 않아, 복잡한 업무처리를 계층적 비즈니스 구조 모델로 표현하기가 어렵고, 생성된 모델의 재사용 가능한 비즈니스 모델의 수가 적게 된다. 그 이유는 MDA(Model-Driven Architecture)[10-13]와 MVC(Model View Controller)[14-15]에 기반하여 상위의 추상적 모델링에서 하위의 구현적 플랫폼에 독립적인 모델링에 이르기까지 범주를 포함하는 계층화된 모델링 프로세스를 제공하지 못하기 때문이다. 이후, 이를 해결하기 위해, 서비스 지향의 비즈니스 모델링에 관련 연구들로서, BPDS-BB[5](가칭)와 MINERVA[16]에서 서비스 지향 BPMN 비즈니스 모델링을 제공하나, MDA와 MVC 기반의 모델링 방법을 지원하지 않는다. ARIS[17]에서 서비스 패러다임에 따라 MDA와 MVC 기반 비즈니스 모델링을 제공하나, BPMN을 대상으로 하지 않으며, 메타모델의 모델링 요소들에 기반의 모델링 프로세스를 갖고 있지 않다.

MDA/MVC 기반 BPMN 서비스 모델링 접근의 필요성은 "관심의 분리(separation of concerns)"의 설계원리에 입각하여 MDA와 MVC 패턴에 의해 계층적으로 분리된 BPMN 모델들을 생성함으로써, 재사용 가능한 독립된 모듈성이 강한 비즈니스 모델의 수를 극대화하기 위함이다. 특수화된 다양한 비즈니스 영역별 그리고 개발의 추상화 수준별로 공통된 BPMN 모델을 계층화하여 구축하고, 이들을 하부를 구성하는 실행 컴포넌트까지 연계하여 CASE Tool로 제공된다면, 분석모델, 설계모델 및 실행 모듈의 재사용을 향상시켜 개발의 노력을 최소화할 수 있다. 결국, BPMN 모델을 이용한 비즈니스 모델링 프로세스는 SOA 지향 MDA와 MVC의 계층적 패러다임을 지원하는 UML 기반의 객체 및 컴포넌트 지향 개발 방법들[18-19]처럼, 성숙되어 있지 않다. 그래서, BPMN 모델의 구조물을 구성하는 범용적 및 공통적 모델링 요소들로 MDA에 의해 정의된 계층적 메타모델이 필요하고 이 메타모델의 모델링 요소들을 사용한 서비스 기반 BPMN 비즈니스 모델링 프로세스가 요구된다.

본 논문은 개발 대상 도메인의 상위 시스템 레벨에서 하위 단위 서비스 수준에 이르는 BPMN 모델의 계층적이고, 확장적인 XL(eXtended/Layered)-BPMN 메타모델과 서비스 기반 BPMN 비즈니스 모델링 프로세스를 제시한다. 먼저, XL-BPMN 메타모델은 MDA 지향의 CIM(Conceptual

Independent Model), PIM(Platform Independent Model) 및 PSM(Platform Specific Model) 패턴의 적용, 그리고 비즈니스 개발의 추상화 정도(모델링 대상의 크기(granularity))에 따라 사용되는 모델링 요소들을 가미하여, 정형화되고 계층적으로 정의한다. 또한, 이 메타모델에 MVC 패턴을 적용해서, 각 모델링 단계별 비즈니스 서비스들(즉, Model service, View service, Controller service)을 분리하여 이차원(수준별 및 계층적)적으로 모델링할 수 있도록 정의한다. 즉, 기존 BPMN 메타모델의 구조물에 서비스 수준의 모델링을 가능하게 하고, MDA 및 MVC 방식을 적용할 수 있도록, 추가적 모델링 요소들을 가미하여 확장된 XL-BPMN 메타모델을 정의한다. 아울러, 비즈니스 모델링 작업에 적용하기 위해, XL-BPMN 메타모델에 기반해서 CIM/PIM 모델링 단계(비즈니스 서비스 모델링, 시스템 서비스 모델링)로 서비스 지향 BPMN 비즈니스 프로세스를 정립한다. 이로서, 서비스 중심의 모듈성이 강한 BPMN 비즈니스 모델을 생성할 수 있고, 정형화된 BPMN 메타모델을 이용한 패턴별 비즈니스 모델의 생성은 재사용성과 독립적 재사용 가능한 모델의 수를 향상시켜준다.

본 논문의 구성은 2장 관련연구에서 서비스 지향의 BPMN 모델링 방법들을 분석한다. 3장에서 확장된 BPMN 모델과 서비스 지향 모델링 프로세스를 기술한다. 4장은 OSS(Online Shopping System)을 가지고 사례적용을 다루며, 5장은 기존 방법 대비 제시방법을 비교평가한다.

2. 관련 연구

2.1 BPMN 비즈니스 모델링

비즈니스 프로세스를 모델링하기 위한 대표적인 모델들로서 BPMN, EPC(Event-driven Process Chain), Petri-Net, ICN 등이 있다. OMG의 표준으로 널리 사용되는 BPMN[6-7]은 비즈니스 프로세스를 디자인 할 수 있는 표준화된 시각적 표기법이다. BPMN으로 모델링된 서비스의 비즈니스 모델은 통상, BPEL(Business Process Execution Language)로 패턴별(순차, 분기 등) 변환, 명세되고, 이 BPEL은 BPM(Business Process Management) 실행엔진을 통해 비즈니스 서비스의 프로세스가 수행된다[9]. BPMN은 UML의 활동 모델(Activity diagram)과 유사하나, 보다 강력한 표현력과 확장성을 가지고 있다. BPMN의 주요한 모델링 요소로서 흐름객체(Flow Object), 연결객체(Connecting Object), 스윘레인(Swimlane), 부가객체(Artifact)로 구성되어 있다. 이를 지원하는 도구로서 50여개가 존재하며, 대표적 도구로 국외의 Pegasystems SmarBPM Suite, BEA AquaLogic BPM Designer, IBM WebSphere Business Modeler 그리고 국내의 HandySoft의 BizFlow가 있다.

BPMN 모델을 정의한 BPMN 메타모델에 대해 살펴보자. 먼저, BPMN V1.0의 메타모델인 [20] 그리고 BPMN 모델링 지원도구 구축을 지원하기 위해 실행수준에서 제시한 메타

Table 1. Study on verification and extension for BPMN model

Method	Features	Proposed content
Validation checking Of business process	<ul style="list-style-type: none"> ○ Validation for four anomalies based on BPMN ○ Using of the π-calculus which the representation of workflow pattern has easy 	<ul style="list-style-type: none"> ○ Utilizing the legacy studies for converting BPMN model into π-calculus ○ Anomaly validation for single and mixed stream in BPMN and its proving with π-calculus
Design anomalies in BPM	<ul style="list-style-type: none"> ○ Syntactic and semantics identification for BPMN model 	<ul style="list-style-type: none"> ○ Identification of syntactic anomalies in BPMN (Using mistaken trigger, and etc) ○ Identification of structural anomalies in BPMN (Deadlock, endless loop, synchronizing) ○ Identification of semantics anomalies in BPMN
Extending the abstraction capability of BPMN	<ul style="list-style-type: none"> ○ Providing the abstraction capability of BPMN model on various view among stakeholders ○ Extending the horizontal abstraction scope of BPMN using super edge concept 	<ul style="list-style-type: none"> ○ Defining the abstract and concrete view of BPMN model ○ Horizontal abstraction technique by super-node and super-edge ○ Defining the order independency of vertical abstraction operation
BPMN extension for security requirements	<ul style="list-style-type: none"> ○ Building the security business process model for business ○ Modeling method by embedding the security requirements In early phase 	<ul style="list-style-type: none"> ○ Extended BPMN metamodel for considering the security requirements ○ Defining the concepts and notation of security modeling elements for Non repudiation, AttackHamDetection, Integrity, Privacy, Access control, Security role, Security

모델인 [21]가 있다. 최근, OMG BPMN metamodel[7] (V2.0 Beta version)은 구성 요소들을 metaclass들에 의해 체계적인 패키지 속으로 조직화하여 정의했다. 핵심 요소별(가령, event, artifact, flow 등)로 구문(syntax)과 시멘틱(semantics)에 의해 클래스 모델(class model)로 정의하고 있다. 그러나, 상위 패키지 모델과 하위 요소별 클래스 모델 간의 연계 구조 및 의미가 복잡하고, 메타모델에 기반한 프로세스를 제시하고 있지 않아 사용이 용이치 않다. 그 외 연구로서, BPMN 모델 자체의 검증[22-23]과 BPMN 표현력 확장[24-25]있는데, 그 내역은 Table 1과 같다.

한편, BPMN 모델의 실행을 위한 BPEL로의 WSDL 변환 및 명세에 대한 기법[26-27] 등이 있다. 이에, 서비스 지향 비즈니스 프로세스를 모델링하기 위한 BPMN 표준, 개발 프로세스, 타 언어로의 변환, 지원도구 등에 대한 연구는 성숙되어 있다.

2.2 서비스 지향 비즈니스 모델링

인터넷의 초고속화와 IT 기술의 성장에 힘입어 산업간, 이질적 미디어간 혹은 이기종 환경간의 융합화가 가속화됨에 따라, 서비스는 점차 복잡화, 대형화 및 다양화되어 가고 있다. 이러한 이기종 환경에서 구현 기술로부터 독립적인 SOA[1-5] 혹은 SOC(Service Oriented Computing)[15]를 통한 서비스 접근이 활성화되고 있다. SOA는 서비스 지향을 지원하는 IT 스타일 아키텍처로서, 전통적 프로그램 중심의 설계와 개발 방식에서 벗어나, 비즈니스 프로세스 관점에서 재사용 가능한 단위로 서비스를 설계, 개발함으로써 특정 프로세스나 서비스 변경, 외부 시스템과의 비즈니스 통합시, 효율적이고 신속한 대응이 가능하다. 이러한 SOA에 따른 어플리케이션 시스템의 서비스 구축은 도메인 업무에 초점을 둔 상위의 비즈니스 모델링 부문과 시스템 기능에 기반하는 하부의 소프트웨어 모델링 부문에 의해 수행된다. 즉, 비즈니스 모델링을 통해 업무 비즈니스의 프로세스를 분석하여 서비스를 도출하고, 소프트웨어 모델링을 통해 이 서

비스에 필요한 컴포넌트들을 식별하고 기존 컴포넌트를 재사용하거나, CORBA(Common Object Request Broker Architecture), DCOM(Distributed Common Model), COM+, EJB(Enterprise Java Bean), J2EE(Java 2 Enterprise Edition), .NET 등의 분산 컴포넌트 플랫폼 환경을 사용해서 새로이 컴포넌트를 구축하여 서비스를 구축한다. 소프트웨어 모델링 부문은 전산학 학문영역에서, UML(Unified Modeling Language)을 사용해서 MDA(Model Driven Architecture)[10-13]와 MVC[14-15] 기반으로 계층적으로 컴포넌트를 생성하는 방법과 프로세스가 [18-19]와 같이 잘 체계화되어 있다. 소프트웨어 모델링 부문을 지원하기 위한 상용 개발 방법론으로 UP(Unified Process), SOUP(Service Oriented Unified process) 및 마르미 III 등이 있다. 반면, 비즈니스 모델링 부문은 산업공학 혹은 경영정보학 학문영역에서, 비즈니스 영역을 추출하고, 비즈니스 서비스를 식별하여, BPMN 등을 사용해서 비즈니스 프로세스를 모델링 작업을 수행한다. BPMN 기반이 아닌 일반적인 비즈니스 모델링을 위한 개발 방법론으로 국외 UMM(UN/CEFACT Modeling Methodology)[28-29]과 IDEF(ICAM Definition, Integration Definition for Function Modeling)[30-31] 등, 그리고 국내 S사의 업무 프로세스 설계(FP 모델: Function Process)와 정보시스템 모델 설계의 작업 활동으로 구성된 BPR/ISP 방법론이 있다.

한편, BPMN을 사용한 서비스 지향 비즈니스 모델링에 관해서 살펴보자. BPDS-BB[5]은 BPMN을 사용해서 비즈니스 모델링을 수행하고 이의 구현을 위한 기술 플랫폼으로 SOA 연계의 접근모형을 제시한다. 그러나, 서비스 지향의 구체적인 BPMN 모델링 방법을 제시하고 있지 않다. MINERVA[16]는 SOC 패러다임과 MDD(Model Driven Development) 기반의 BPMN 비즈니스 프로세스를 위한 MINERVA 프레임워크를 제시한다. 이는 플랫폼은 3차원 구조로서, 첫째 개념적 차원은 비즈니스 프로세스 모델링/서비스 지향 모델링/서비스 지향 실행의 상위수준 구조, 둘째

방법론적 차원은 비즈니스 모델링/요구분석/설계/구현 단계에 따른 비즈니스 프로세스/서비스 식별, 명세/서비스 구현, 셋째 이 프로세스를 지원하는 톨 차원으로 제시했다. 그러나, MDA/MVC 기반의 접근을 제공하지 않는다. ARIS[17]는 EPC과 BPMN 모델을 사용한 SOA/MDA(CIM, PIM)/MVC 기반의 ARIS 모델링 방법을 제시한다. 특히, MDA 기반의 정적/동적 및 내부/외부 관점에 의한 서비스 식별기법을 위한 SOA 메타모델을 다룬다. 그러나, 비즈니스 모델링 및 서비스 식별 방법이 EPC 메타모델의 모델링요소를 대상으로 한다. 즉, BPMN 모델 대상의 비즈니스 서비스 모델링 방법과는 상이하다.

따라서, 이들의 비즈니스 모델링 개발방법에서는 MDA와 MVC에 기반한 개발의 추상화 수준에 따른 서비스 중심의 계층적인 개발 프로세스가 미약하다. 이로서, 생성된 비즈니스 모델 및 시스템 서비스 모델의 재사용이 활성화되지 못함으로서 다른 전체적인 시스템 개발의 생산성의 저하를 야기한다. 또한, 비즈니스 모델링과 소프트웨어 모델링 작업간에 조직적으로 상호 연계, 통합된 개발 프로세스를 지원하고 있지 않다.

결과적으로, 기존 방법들의 문제점은 생성된 BPMN 비즈니스 모델의 재사용성을 향상시키기 위해, BPMN 모델을 구축함에 있어 개발의 추상화 수준에 따른 계층적이고 모델 기반의 모델링 접근에 관한 연구가 미약하다. 즉, BPMN 메타모델의 모델링 요소들과 그들 간의 구문적 관계에 기반해서, BPMN 모델을 추상화 정도에 따른 MDA의 CIM 및 PIM 방식에 의한 독립적 BPMN 모델들의 구축 그리고 어플리케이션 아키텍처의 MVC 패턴에 의한 분리 독립된 BPMN 모델을 디자인을 위한 방법이 결여되어 있다. 따라서, 모델링 단계의 수준별/영역별로 독립적인 재사용 가능한 많은 비즈니스 모델들을 생성할 수 없으므로, 작성된 BPMN 모델의 재사용에 저해가 되고 있다. 아울러, 계층적 접근의 부재는 복잡한 비즈니스 모델의 모델링 작업을 어렵게 만든다. 이것은 정형화되고 계층화된 BPMN 모델의 메타모델 및 모델링 프로세스를 지원하지 않기 때문이다.

3. 서비스 기반 BPMN 메타모델 및 프로세스

본 논문은 복잡한 비즈니스 서비스를 효과적으로 모델링하기 위해서 추상적/계층적 접근에 의한 비즈니스 서비스의 모델링 방법을 제공하는 것이다. 목표는 다양한 비즈니스 응용분야에 적용이 가능한 범용적이고 일원화된 서비스 지향의 계층적 BPMN 모델링 기법에 초점을 둔다. 따라서, MDA/MVC에 준거하여 BPMN 모델의 모델링 요소들이 개발 단계에 따라 상이하게 적용되는 계층적 접근을 제시한다. 제시 내용은 다음과 같다.

- BPMN 모델의 계층성을 가진 확장된 메타모델 정의(CIM, PIM, PSM)

- 메타모델 사용의 MDA 기반 계층적 서비스 모델링 프로세스 정립(CIM, PIM)
- MVC 기반의 서비스 분리 모델링

연구 접근의 범주 모델은 Fig. 1과 같이, BPMN 모델의 구조물을 계층화하여 메타모델로 정의하고 이 메타모델에 기반하여 BPMN 서비스 모델링 프로세스를 정립한다. Fig. 1에서 회색 굵은 박스 부분이 본 논문에서 제시하는 것이다.

본 접근을 위한 개념(concept)들[19]로 메타모델(metamodel), 계층(hierarchy), MDA, 그리고 MVC를 이용한다. 메타모델은 BPMN 모델이 가진 추상적 구문(abstract syntax)을 정의하기 위해 사용한다. 계층은 생성된 비즈니스 모델의 재사용성을 높이기 위해 메타모델과 모델링 프로세스의 정의에 적용된다.

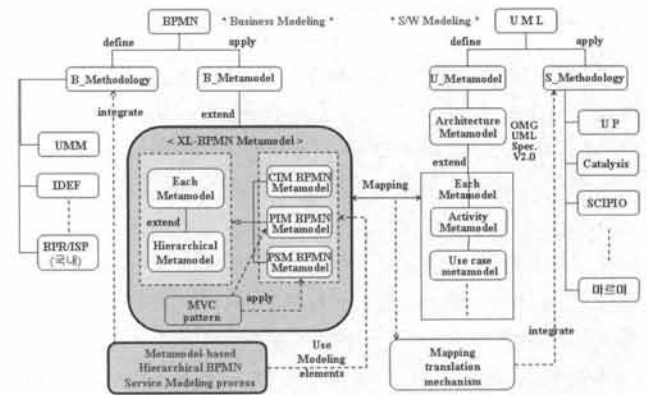


Fig. 1. Scope model of layered BPMN modeling approach

이것은 “관심의 분리” 원리에 기반하여 디자인 모델들을 분리하기 위한 계층적 구조로 조직화 해준다. 도메인의 크기에 의한 계층화 수준은 시스템(system), 도메인 태스크(domain task), 서비스(service) 층으로 분류한다. 반면, 추상화에 의한 계층화 타입은 개발의 생명주기에 따라 개념적 수준의 CIM modeling phase, 명세적 수준의 PIM modeling phase로 구성한다. 계층화된 비즈니스 모델을 구축하기 위한 MDA는 구현 플랫폼을 고려하지 않는 모델의 분리, 생성을 지원하는 제공하는 PIM 기술을 의미한다. MVC는 클라이언트측의 사용자 인터페이스에 대해, 서버측의 비즈니스 로직 모델을 거쳐 데이터 모형에 관련시키기 위한 설계 방식이다. 즉, M은 entity 클래스, V는 GUI(Graphic User Interface) 클래스, C는 controller 클래스를 나타낸다. 이러한 개념에 입각하여, 메타모델과 프로세스를 구축하기 위한 원리[19]는 다음과 같다.

- 계층적 BPMN 메타모델 정의
 - MDA와 MVC 패턴을 적용하여 계층적 메타모델을 정의한다.
 - 계층적 메타모델의 구축은 필수적/실용적/범용적으로 사용되는 핵심 모델링 요소들과 그들 간의 관계 요소

들을 가지고 CIM/PIM/PSM 모델을 구축토록 계층화하여 정의한다.

- 계층적 모델링을 제공하기 위해, BPMN 모델이 가지고 있는 모델링 요소들은 개발의 추상화 정도 혹은 어플리케이션의 크기에 따라 명확하게 계층화하여 정의한다.
- 계층적 메타모델내 모델링 요소는 개발 단계가 구체화될수록 추가, 확장된다.
- 계층적 BPMN 서비스 비즈니스 프로세스 구축
 - 계층적 메타모델을 사용해서 2개 모델링 단계에 의해 모델링 절차를 정립한다. 즉, 비즈니스 서비스(Business service) 그리고 시스템 서비스(System service)의 모델링 단계로 구성한다.
 - 도메인 및 구현 플랫폼에 종속 여부에 따라, MDA 기반의 CIM과 PIM 패턴에 의해 모델을 구축한다.
 - MDA와 MVC에 의해 생성된 산출물(비즈니스 모델)들은 단위 독립적 모델로서 재사용 될 수 있도록 한다.

3.1 BPMN 모델상의 서비스

서비스 지향의 BPMN 모델을 디자인하기 위해, 서비스에 대한 정의 및 분류가 필요하다. 본 논문에서 서비스(혹은 비즈니스 서비스)는 한 개의 재사용 가능한 독립적 업무 단위 혹은 기능을 의미하며, 업무 기능이 시작해서 종료되면 서비스 작업이 완료되어야 한다(원자성, 단일성 보유). BPMN 모델에서 서비스 구성은 아래와 같다.

- 한 개 어플리케이션 시스템 = 도메인 태스크의 집합체
- 한 개 도메인 태스크 = 비즈니스 서비스의 집합체
- 한 개 비즈니스 서비스 = 한 개 서비스

이때, 한 개 서비스별로 MDA 및 MVC 패턴에 의해 다수의 BPMN 모델들이 작성된다. 이러한 서비스 구성과 유형을 고려해서 서비스 메타모델을 정의한 것이 Fig. 2이다. Fig. 2에서 "coarse-grained service"는 상위 서비스를 의미하며, "fine-grained service"는 상위 서비스에 복합(composite)의 관계를 갖는 하위서비스를 나타낸다.

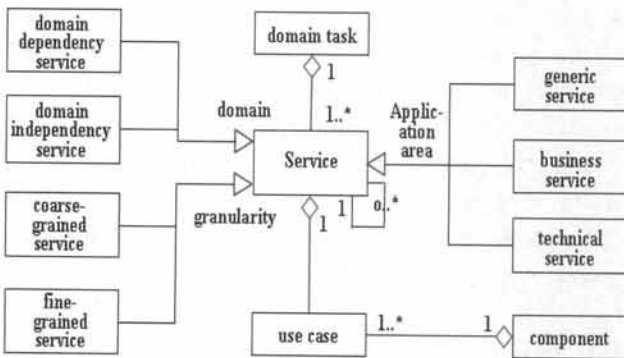


Fig. 2. Service metamodel

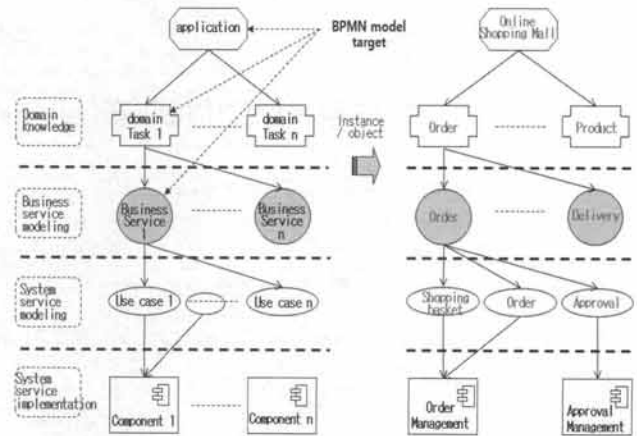


Fig. 3. Service construction of online shopping mall system

모델링 단계에 따른 Fig. 2의 메타모델에 기반한 "온라인 쇼핑몰시스템" 대상의 서비스 구축 예가 Fig. 3이다.

3.2 MDA/MVC 기반 계층적 BPMN 메타모델

BPMN 모델을 정의하기 위한 메타모델은 두 가지로 정립한다. 먼저, BPMN 모델을 정의한 OMG 표준에 기반하여 BPMN 메타모델(Fig. 4)을 정의한다. 다음으로, 이 BPMN 메타모델에 서비스 기반의 확장된 XL-BPMN 메타모델(Fig. 6)을 정의한다.

1) BPMN 메타모델

BPMN 모델은 흐름 객체, 연결 객체, 스웜레인 및 부가 객체 등의 핵심 모델링 요소를 가지고, 구축 대상 도메인의 업무에 대해 비즈니스 프로세스(혹은 control flow)를 표현하는 플로우차트 타입(flowchart-type) 다이어그램이다. OMG BPMN V1.0 표준[6-7]과 [9]에서 제시한 BPMN 모델의 구조물을 구성하는 모델링 요소들에 근거하여 BPMN 모델을 정의한 기본적인 BPMN 메타모델이 Fig. 4이다.

2) XL-BPMN 메타모델

BPMN 모델에 MDA 기반 모델링 단계의 계층적 접근과 MVC 기반 어플리케이션 아키텍처의 분리적 접근을 제공하기 위해, Fig. 4의 BPMN 메타모델에 모델링 요소들이 추가되어야 하고, 이들을 계층화하여 확장된 메타모델로 정의되어야 한다. Table 2는 확장되고 계층화된 XL-BPMN 메타모델을 정립하기 위한 모델링 대상, 모델링 방법 및 수준을 정립한 것이다.

Table 2에서, MDA 방식은 CIM과 PIM 패턴에 의해 3계층 BPMN 메타모델로 정의한다. 즉, CIM BPMN 메타모델 그리고 PIM BPMN 메타모델로 구성한다. 각 메타모델의 구성은 개발 프로세스상에서 도메인 및 플랫폼의 종속여부에 따라 BPMN의 기존 및 확장된 모델링 요소들을 가지고 메타모델을 계층화하여 정의한다. MVC 패턴은 PIM BPMN 메타모델에 모델링요소로 적용된다. 가령, PIM BPMN 메타모델의 M/V/C 모델링요소를 사용해서 PIM 수준에서 실제 디자

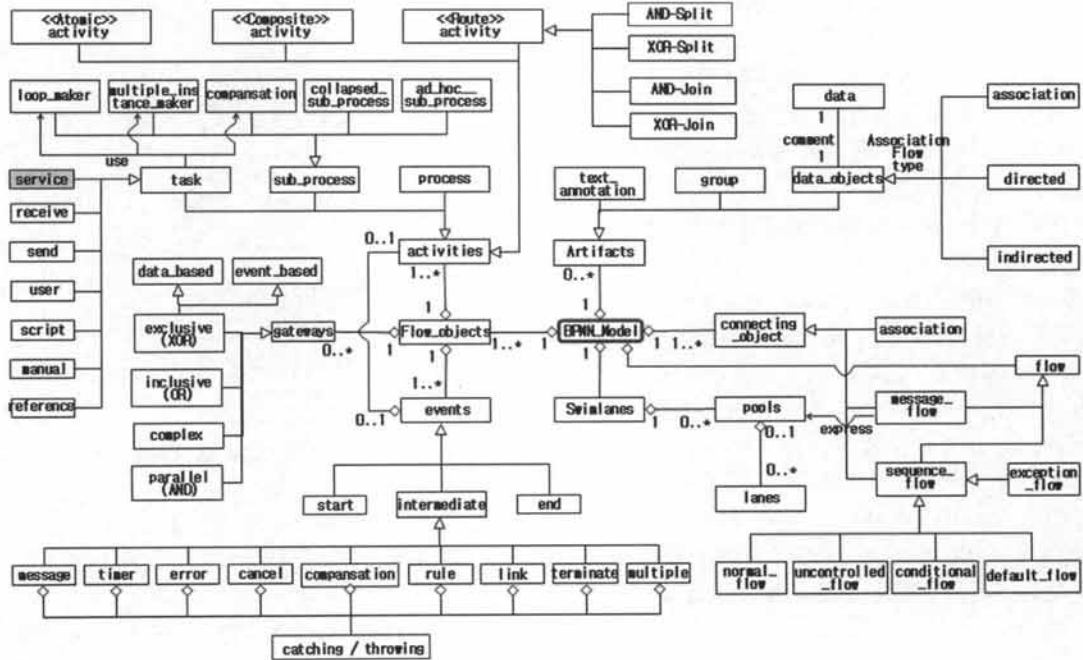


Fig. 4. BPMN metamodel

Table 2. Criteria and method to build XL-BPMN Metamodel

Appl. Arch. Patt. MDA Type	Client (View)	Business (Controller)	Data (Model)
CIM	<ul style="list-style-type: none"> Basic domain business modeling Designing the layered BPMN models according to different levels of granularity of modeling target - system, domain task, service The construction of metamodel with essential modeling elements of BPMN model 		
PIM	<ul style="list-style-type: none"> Specific domain business modeling which makes CIM BPMN model with service-sized target domain Building the metamodel with concrete modeling elements of BPMN model - process, sub-process, task, and etc 	<ul style="list-style-type: none"> Identification of Client service for one service target 	<ul style="list-style-type: none"> Identification of Data service for one service target
PSM	<ul style="list-style-type: none"> Making the model by using the GUI modeling elements with platform dependent (BPEL and Java target) 	<ul style="list-style-type: none"> Making the model by using the Business modeling elements with platform dependent (BPEL and Java target) 	<ul style="list-style-type: none"> Making the model by using the DB modeling elements with platform dependent (BPEL and Java target)

인시 생성되는 모델은 Client service BPMN 모델, Business logic BPMN 모델 및 Data service BPMN이 작성된다.

여기서, "service" 수준에서 MVC 패턴 적용의 방법은 Fig. 5와 같다.

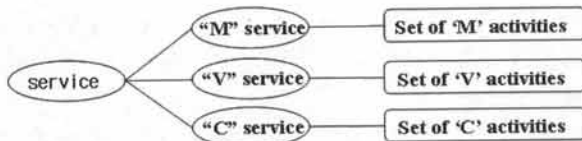


Fig. 5. Applying of MVC pattern at the "service" level

BPMN 모델을 계층적 모델링을 제공하기 위한 Layered MetaModel(L-MM)[19]은 다음과 같이 정의한다.

[정의 3-1] (Layered Metamodel). A L-MM은 L-MM = {n, AL, MECl_a, Rel, Mul, ST, Lab}로 구성한

다: (i) n은 metamodel name (CIM/PIM/PSM BPMN L-MM); (ii) AL은 계층적 BPMN Model의 추상화 수준; (iii) MDD와 MVC가 체화된 BPMN Model 구조물 modeling element classes인 유한집합 MECl_a; (iv) MECl_a들 간의 relation Rel ⊆ (MECl_a × MECl_a); (v) MECl_a들 간의 대응수 Mul ⊆ (MECl_a × MECl_a) (*, 1..*, ...); (vi) MECl_a(i, j, k, ...)에 대한 스테레오타입 클래스 ST, 여기서 스테레오타입은 플랫폼 타입, 크기 수준 등; (vii) MECl_a들 간의 관계 라벨 ST(읍선).

따라서, XL-BPMN 메타모델은 [정의 3-1]에 의해 정의된 L-MM의 인스턴스(instance)로 다음과 같이 정의된다.

[정의 3-2] (XL-BPMN Metamodel). XL-BPMN Metamodel은 다음의 요소들로 구성한다. (1) XL-BPMN Metamodel은 메타모델 이름; (2) CIM, PIM, PSM은 추상화 수준; (3) process, activities, task 등은 BPMN 모델링 요소인 클래스들; (4) association, inheritance, aggregation의 3개 관계들; (5) 1, 0..1, 0..*, 1..*의 4개 대응수; (6) client service, business_logic service, data service 등은 스테레오타입된 클래스들; (7) express, comment의 관계라벨들.

한편, XL-BPMN 메타모델에 MDA/MVC 패턴 그리고 크기에 의해 새로이 확장되는 모델링요소들에 대한 프로파일을 정의한 것이 Table 3이다. Table 3은 기존의 BPMN 모델이 가진 모델링 요소에 "서비스 유형"과 모델링 대상의 "크기"(Fig. 2 참조), "MVC 패턴", 그리고 액티비티를 수행하는 "행위자"들에 관련된 모델링 요소들을 추가한 것을 나타낸다.

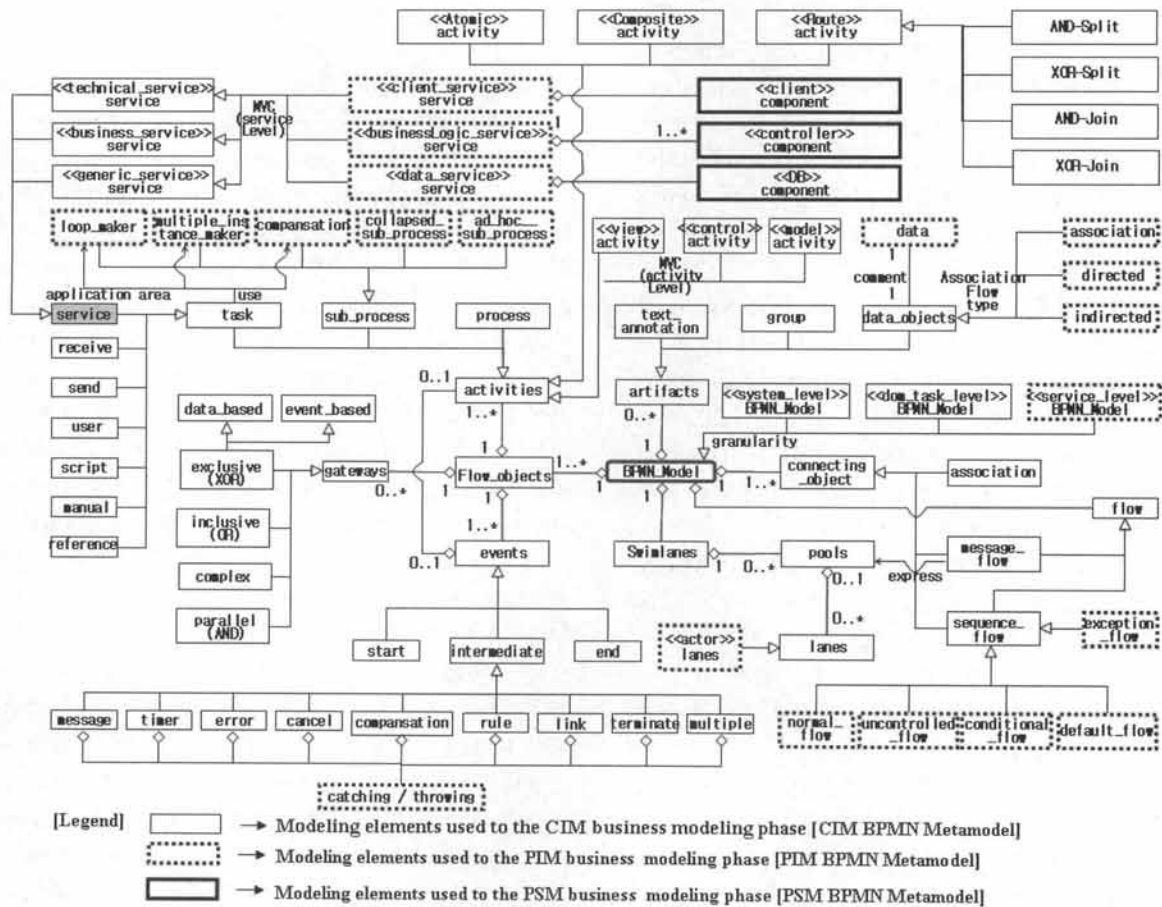


Fig. 6. XL-BPMN metamodel

Table 3. Profile of extended XL-BPMN modeling element

Section		Description	Representation
Type	Modeling element		
Service	generic service	Common and generic service in specific domain task level	Stereotype
	business service	Business service in specific domain task level	Stereotype
	technical service	Technical or software service in specific domain task level	Stereotype
MVC	client service	The Service for handling user interface in specific domain task	Stereotype
	business logic service	The Service for handling business logic in specific domain task	Stereotype
	data service	The Service for handling data in specific domain task level	Stereotype
Size	system	Generating the BPMN model with system-sized granularity	Stereotype
	domain task	Building the BPMN model with domain task-sized granularity	Stereotype
	service	Making the BPMN model with service-sized granularity	Stereotype
Behavior	actor	Separated representation of each actor by using swimlane	Stereotype

이러한 확장된 모델링 요소들을 Fig. 4의 BPMN 메타모델에 반영하여 계층적으로 정의한 확장된 XL-BPMN 메타모델이 Fig. 6이다. Fig. 6은 모델링 단계에 고려한 CIM/PIM/PSM BPMN 메타모델을 보여준다. 각 메타모델들은 개발의 추상화 수준 및 MDA/MVC에 부합하는 확장된 모델링 요소들로 정의한다. MVC 패턴에 의한 모델링 요소는 "service"에 대해 각각 "MVC"로 분리하여 스테레오타입으

로 표현한다. 가령, 'V'에 의해 GUI service와 GUI activity로 표현한다.

3.3 메타모델 기반 BPMN 서비스 모델링 프로세스

본 논문의 접근은 MDA와 MVC가 결합된 모델링을 위한 아키텍처에 초점을 둔다. 그 이유는 관점(view)에 따라 생성된 산출물에 대해 service 수준에서, model 수준에서, 그리고 component 수준에서 다양한 크기 대상으로 재사용성을 향상시키기 위함이다. 그래서, MVC 패턴은 개발의 추상화 정도를 고려한 MDA 패러다임을 지향하며, 도메인 의존성과 구현 플랫폼 환경의 적용여부에 의존해서, 어플리케이션 모델들을 분리하여 구축할 수 있는 2계층(CIM/PIM) 서비스 모델링 단계들로 정의한다. 본 고에서 PSM 수준을 위한 모델링 단계의 프로세스는 포함하지 않았다. 왜냐하면, SOA 기반 M/V/C/벨 컴포넌트 모델, BPEL 모델 및 WSDL 명세가 결합된 메타모델과 프로세스의 정립 그리고 이질적 모델 간 변환 알고리즘 등의 작업이 요구되는 또 다른 이슈가 되기에 후속연구에서 제시할 것이다. 이에, 2단계 BPMN 모델링 아키텍처 및 프로세스는 다음과 같이 정의한다.

[정의 3-3] (BPMN-모델링 아키텍처). BPMN 모델링 아키텍처 B-MA = {MDA U MVC}는 다음과 같이 구성한

다. (i) model-driven architecture *MDA* 접근, 여기서 $MDA = \{For\ BPMN\ models \mid \{CIM-BPMN-Model(Bus-Ser-mp)\} \cup \{PIM-BPMN-Model(Sys-Ser-mp)\}\}$, Bus-Ser-mp는 비즈니스 서비스 모델링 단계이며, Sys-Ser-mp는 시스템 서비스 모델링 단계를 각각 의미한다; (ii) 어플리케이션 아키텍처 *MVC* 패턴, 여기서 $MVC = \{For\ BPMN\ models \mid \{View-Service\} \cup \{Controller-Service\} \cup \{Model-Service\}\}$ 이다.

[정의 3-4] (서비스 생성: SG). 단위 cell service를 B-MA에서 작성되는 모델 단위의 산출물이라 하자. 각 서비스 cell은 서비스 집합의 하나이다. $SG = \{For\ BPMN\ models \mid \{(CIM) \wedge \{PIM\}\} \cup \{(M) \wedge \{V\} \wedge \{C\}\}$, 여기서 $\{(CIM) \cap \{PIM\}\} = \emptyset$ and $\{(M) \cap \{V\} \cap \{C\}\} = \emptyset$ (공집합). 즉, 한 개의 cell 서비스는 모든 type들에 대한 MVC와 MDA가 존재한다면 작성 가능하다.

[정의 3-5] (BPMN 모델링 프로세스: B-MP). BPMN 모델링 프로세스는 $B-MP = (n, MP, -, I)$ 로 구성한다. (i) 비즈니스 모델링 프로세스 이름 n ; (ii) 모델링 단계들의 유한집합 MP ; (iii) 모델링 단계간의 프로세스 순서 관계 $- \subseteq (MP \times MP)$; (iv) 나선형 프로세스 모델에 기반한 반복 흐름 I (반복 #1, 반복 #2, ...).

[정의 3-6] (모델링 단계: MP). 모델링 단계는 $MP = (n, ACT, \rightarrow, WP)$ 로 구성한다. (1) 모델링 단계의 이름 n (비즈니스 서비스 MP , 시스템 서비스 MP); (2) 액티비티들의 유한집합 ACT , $x, y, z \in MP$, 액티비티는 이전 액티비티를 통한 입력물과 XL-BPMN 메타모델에 정의한 모델링요소들을 사용해서 모델을 작성한다; (3) 전이 관계 $\rightarrow \subseteq (ACT \times ACT)$, 관계 유형으로 previous/after 그리고 fork/join를 갖는다; (4) 액티비티의 출력 혹은 산출물 WP .

상기 [정의 3-3]에서 [정의 3-6]에 의거하여, Fig. 7은

MDA와 MVC에 의한 비즈니스 서비스의 모델링 아키텍처 및 모델링 프로세스를 보여준다. 모델링 단계에 걸쳐 MDA에 의해 생성되는 산출물들은 CIM BPMN business model과 PIM BPMN service model이다. MVC pattern은 PIM BPMN 모델에 대해 application의 service architecture를 Client/Controller/Data로 분리, 구축된 BPMN Model을 Client service/View service/Controller service로 분리하여 디자인한다. 이러한 MDA와 MVC를 적용해서 계층적 BPMN 모델링 프로세스가 정립된다. 계층적 비즈니스 모델링 작업은 MDA 기반 개발 단계의 생명주기에 따라 상위의 추상적 모델링에서 하위의 구체적 모델링으로 디자인해나가는 프로세스를 정립한다. Fig. 7에서 순번 1에서 7번에 이르는 활동은 Fig. 6에서 정의한 CIM/PIM XL-BPMN 메타모델의 모델링 요소들을 사용해서 수행한다. 이것이 메타모델에 기반한 모델링 프로세스가 수행됨을 의미한다.

이 프로세스는 단계에 걸쳐, 나선형의 반복적 생명주기로 구성하여, 지속적으로 모델이 개선되고 확장되는 특성을 갖는다. 각 단계는 활동들로 조직화되고, 각 활동은 세부적 프로세스 지침을 정의하며, 활동을 통한 산출물이 생성되고, 산출물간의 연계성을 가지도록 정립한다. 여기서, PIM 단계에서 BPMN 모델을 M, V 및 C 패턴에 의해 분리된 형태로 생성한다. 이 모델링 프로세스를 통한 BPMN 모델의 디자인 과정은 비즈니스 서비스 모델링 단계에서 Fig. 6에서 정의한 CIM 적용의 모델링 요소들(즉, Fig. 4의 BPMN 기본 모델링 요소)을 가지고 "system" 대상 크기의 BPMN 모델을 작성하고, 또한 이 모델을 구성하는 각 "domain task" (즉, 서비스 단위) 및 "service" 대상 크기의 CIM BPMN 모델들을 생성한다. 이후, 시스템 서비스 모델링 단계에서는 XL-BPMN 메타모델에서 정의한 PIM 모델링 요소들을 사용해서, 이전 단계의 서비스별 각 BPMN 모델에 대해 MVC 패턴에 의해 각각 분리된 PSM BPMN 모델을 생성한다.

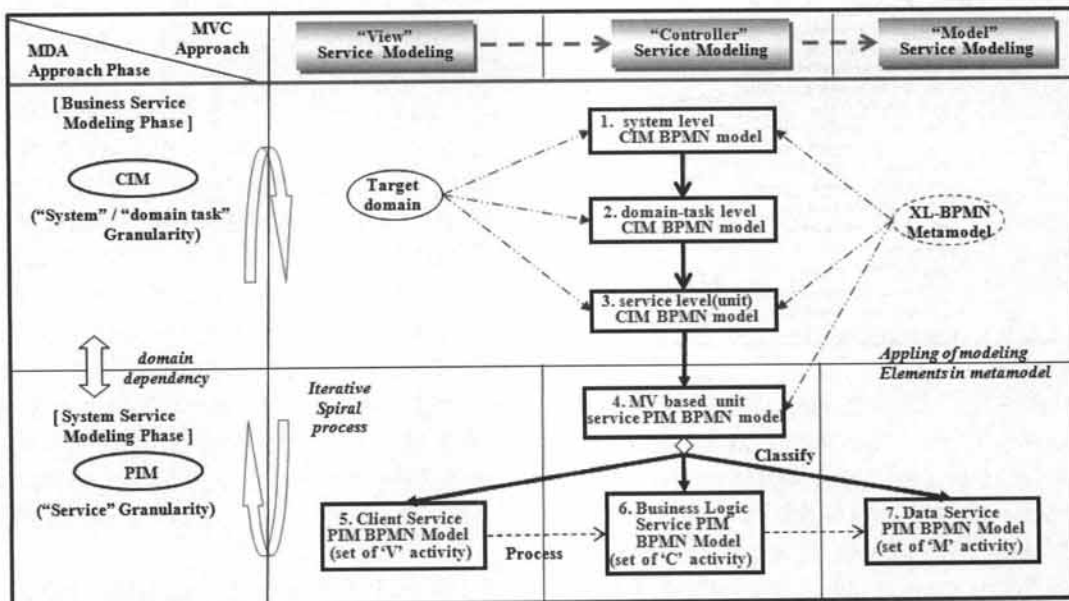


Fig. 7. Layered BPMN modeling architecture and process based on MDA/MVC

4. 적용사례: OSS

제시한 계층적 BPMN 모델링 프로세스의 실효성을 보이기 위하여, 온라인쇼핑몰시스템(OSS)을 대상으로 서비스 지향의 BPMN 비즈니스 모델들을 기술한다.

4.1 OSS의 Business service modeling phase

비즈니스 서비스 모델링 단계는 전체 업무에 대한 도메인의 기존의 개략적 비즈니스 제어흐름을 디자인한다. 이를 위해, 최상위 "system" 수준에서 단위 "service" 수준에 이르는 비즈니스 업무를 모델링한다. 본 단계는 MDA에 의한 CIM 모델을 작성하나, MVC 패턴은 적용되지 않는다. Fig. 6의 CIM BPMN Metamodel의 모델링요소들을 사용하고, Fig. 7의 개발 프로세스에 따라, 최상위 수준에서 상위 "system" 기반 CIM BPMN 모델을 나타낸 것이 Fig. 8이다. Fig. 8에서 XL-BPMN 메타모델의 모델링 요소로 "sub_process" activity, "exclusive(XOR)" 등을 사용했다. 즉, 작은 사각형 내 "+"로 표기된 액티비티는 계층적으로 하부 BPMN 모델을 가짐을 의미한다. 즉, 이들 간에는 복합 혹은 sub-process의 관계를 가진다.

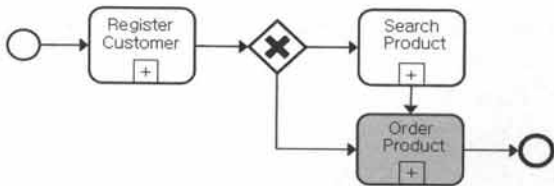


Fig. 8. CIM BPMN model at the "system" level

다음으로 중위 수준의 "domain-task" 기반 CIM BPMN 모델을 표현한 것이 Fig. 9이다. 이것은 Fig. 8의 "Order Product" 도메인 태스크를 대상으로 서비스 분류인 "technical/business/generic" 서비스의 모델링요소를 추가, 적용하여 작성했다.

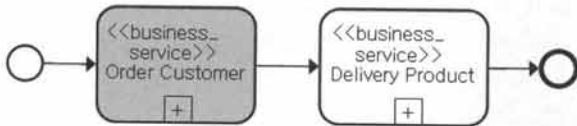


Fig. 9. CIM BPMN model at the "domain-task" level

또한, 하위 수준의 단위 셀(cell) "service" 기반으로 그림 9의 "Order Customer" 비즈니스 서비스를 계층적으로 복합(composite)한 CIM BPMN 모델이 Fig. 10이다. 이 단위 서비스가 서비스 지향의 BPMN 모델을 의미하고, 이를 통해 독립적 재사용의 단위 모델이 되는 것이다.

4.2 OSS의 System service modeling phase

본 모델링 단계는 business service modeling phase에서 식별된 단위의 셀 서비스별로 서비스 플랫폼에 독립적인

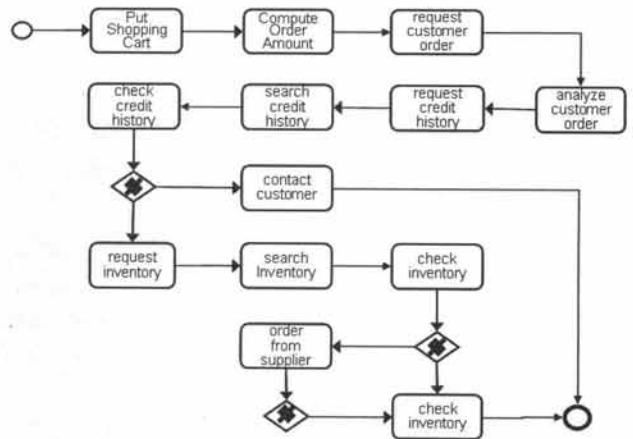


Fig. 10. CIM BPMN model at the unit "service" level

PIM 패턴 그리고 MVC 패턴을 적용하여 비즈니스 측면이 아닌, 시스템 측면에서 상세한 서비스 모델링을 수행한다. 따라서, PIM XL-BPMN Metamodel의 확장된 모델링 요소들(즉, "Client_service", "business_service" 및 "data_service")을 사용해서 BPMN 모델을 디자인한다. Fig. 7의 모델링 프로세스에 따라 각 서비스별로 MVC 기반 PIM BPMN 모델을 작성한다. 이에, Client service PIM BPMN 모델, Business logic service PIM BPMN 모델, 그리고 Data service PIM BPMN 모델들로 생성된다. 여기서, Business logic과 Data의 service PIM BPMN 모델은 EA(Enterprise Architecture) 기반의 application architecture와 DB architecture에 초석이 될 수 있다. Fig. 11은 MVC 기반 통합 PIM BPMN 모델을 나타내고, Fig. 12는 Client service PIM BPMN 모델을 각각 보여준다.

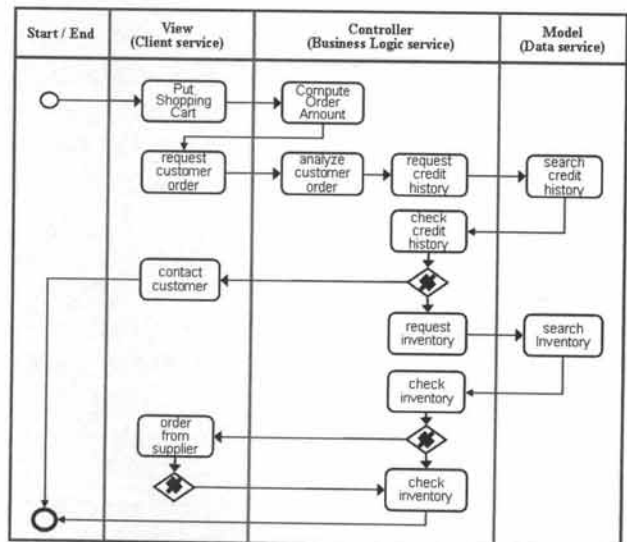


Fig. 11. MVC based Integrated service PIM BPMN model

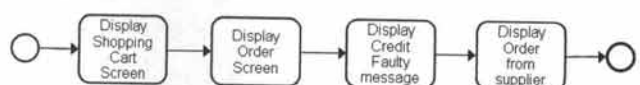


Fig. 12. PIM BPMN model focused on client service view

5. 평 가

5.1 기존 방법과의 비교평가

제시 방법과 기존 방법간의 비교를 위해 BPMN 메타모델, BPMN 모델링 프로세스 그리고 개발 노력 비용의 비교 매트릭스들을 가지고 분석한다.

1) BPMN 메타모델

BPMN 메타모델의 구성 측면에서, 제시 XL-BPMN 대비해서 기존 BPMN 표준 V1.0[6, 20-21]과 BPMN Beta V2.0[7]을 대상으로 비교한 것이 Table 4이다.

Table 4. Comparison in terms of formation of BPMN metamodel

Evaluation factor	BPMN standard V1.0 metamodel	BPMN Beta V2.0 metamodel	Proposed XL-BPMN Metamodel
Cover level of entire modeling phase	Low	High	High
Defining the layered modeling elements by MDA	Middle	Middle	High
Specifying the separated modeling elements by MVC	Low	Low	Middle
Additional scalability of modeling element	Low	High	High
Providing a service modeling element	Middle	Middle	High

Table 4에서, BPMN 표준 V1.0은 BPD(Business Process Diagram)의 핵심요소와 확장 요소들로 그 의미와 표기법을 정의하였으나, 모델 구조물의 메타모델은 제공하지 않는다. 즉, [20]은 BPMN 메타모델을 제시했고, [21]는 BPMN 모델을 실행 수준에서 모델링요소들을 스테레오타입과 객체화하여 메타모델을 제시했다. 최근, BPMN Beta V2.0은 BPD를 BPMN이라 개칭하여, 핵심 및 확장 요소 그리고 요소들간의 인프라 및 협력관계에 대한 구문을 메타모델로 정의하고, 요소들의 시맨틱을 제공한다. 그러나, BPMN 표준 V1.0은 메타모델을 정의하고 있지 않아, 각 평가 항목에서 낮은 수준을 갖는다. 메타모델을 제공하는 BPMN Beta V2.0은 MDA 및 MVC 기반의 계층화된 모델링 요소들로 구성된 메타모델을 제공하지 않는다. 반면, 제시 XL-BPMN 메타모델은 서비스 중심 그리고 MDA/MVC의 계층화된 모델링 요소들을 제공한다.

2) BPMN 모델링 프로세스

BPMN 모델링 프로세스 측면에서, BPDS-BB[5], MINERVA[16] 그리고 ARIS[17]를 대상으로 비교, 분석한 것이 Table 5이다. Table 5에서, 본 논문과 ARIS 방법은 서비스 지향 패러다임에 따라 MDA/MVC 기반의 BPMN 모델링 방법을 다룬 것에 유사성을 갖는다. 그러나, ARIS는 계층화되고 BPMN 메타모델 기반의 모델링요소들을 사용한 프로세스를 제공하지 않는다. 이외, BPDS-BB와 MINERVA

방법들은 MDA 및 MVC 패턴에 의한 프로세스 제공이 부족하다. 한편, 제시하는 메타모델을 사용하면, BPMN 모델의 전체적 모델링 요소들간의 구문적 구조의 이해성을 향상시킬 수 있으며, 메타모델에 의한 구축된 CIM과 PIM 모델들간의 추적성 혹은 일치성을 검사가 가능해진다.

Table 5. Comparison of the details of BPMN modeling process

Evaluation factor	BPDS-BB	MINERVA	ARIS	Proposed method
Target business model	BPMN	BPMN	EPC	BPMN
Designing a model based CIM, PIM pattern	No	No	Yes	Yes
Designing a layered model based MVC	No	No	Yes	Yes
Designing a model using a layered modeling elements by development phase	No	PS	No	Yes
Supporting case study	Yes	Yes	PS	No
Service oriented model design	PS	Yes	Yes	Yes

[legend : Yes(support), No(not support), PS(Partial support)]

3) 개발 노력비용 평가

노력비용 측면에서, 시스템의 규모가 클수록 기존 방법에 비해 제시 방법이 많은 재사용 비즈니스 모델을 사용할 수 있기 때문에 좀 더 노력 비용을 절감할 수 있을 것이다. 반면, 시스템의 규모가 작으면, 제시 방법은 재사용 모델을 생성하기 위한 많은 모델링 활동을 수행하기에 기존 방법 대비 더 많은 노력 비용이 요구된다.

유지보수 및 재사용 기반 개발 측면에서, 개발비용 평가는 Table 6과 같다. Table 6에서, 제시 방법은 추상화 수준별(MDA/MVC) 및 모델별 정형화된 메타모델의 모델링 요소들을 사용하기에 기존 방법들보다 우수하다고 할 수 있다. 가령, 새로운 시스템의 구축에 대해, Fig. 7에서, 단위 서비스 수준에서 "6. Data Service PIM BPMN Model"가 상이할 경우, 1번에서 5번에 이르는 즉, 기 작성된 "CIM BPMN Model", "Business Logic PIM BPMN Model"들을 재사용할 수 있기 때문에 그 만큼 개발 노력 비용을 절감할 수 있다.

Table 6. Evaluation of effort cost in case of maintenance and reuse based development

Evaluation factor	BPDS-BB	MINERVA	ARIS	Proposed method	
Modification cost	High	High	Low	Low	
Maintenance cost	High	High	Low	Low	
Reuse-based development cost	Extended cost	High	High	Low	Low
	Reuse cost	High	High	Low	Low

5.2 재사용성 평가

기존 방법들과 대비해서, 제시 방법을 적용해서 작성된 산출물 모델들의 재사용성 측면에서 정량적 평가를 보인다.

재사용성은 MDA/MVC 기반으로 작성된 BPMN 모델들이 새로운 도메인의 BPMN 모델 구축시 재사용되는 것에 대한 척도이다. 제시한 계층적 메타모델과 모델링 프로세스에 의해 다양한 추상화 수준 및 크기에서 재사용 될 수 있음을 4장의 적용사례를 통해서 알 수 있다. 가령, 새로이 구축되는 BPMN 모델이 PIM 수준 이하에서 상이할 경우, 이전 단계의 산출물인 “system/domain-task/service” 수준에서 작성된 CIM BPMN 모델들을 그대로 재사용할 수 있다. 따라서, 제시한 MDA/MVC 기반으로 작성된 BPMN 모델들의 재사용성을 좀 더 향상시킬 수 있다. 재사용 가능 모델 수를 산정하는 방식은 다음과 같다.

- 산정 방법
 - 재사용 가능 구축 모델 수 = CIM/PIM 모델 수(개발단계 수) x MVC 모델 수
 - 모델링 과정에서 생성되는 재사용 모델 수들을 누적하여 얻을 수 있다.
- 산정 수식

$$U_{(R)}^{Num} = \sum_{i=CIM}^{PIM} (MDA\ model) \left(\sum_{j=M}^C (MVC\ model) \right)$$

상기의 산정 수식에 의거, 각 모델링 방법들과의 재사용 모델 수를 비교한 것을 Table 7에서 보여준다. BPDS-BB와 MINERVA에 대한 MDA 모델 수가 “1”인 이유는 그들 방법이 MDA와 MVC에 의한 모델을 작성하지 않지만, 구축 대상 도메인의 비즈니스 모델로서 1개의 모델을 생성하기 때문이다. 반면, MDA 모델 수 산정에 대해서, ARIS와 제시 방법은 CIM 모델 및 PIM 모델을 작성하기에 모델의 수가 “2”이 된다.

Table 7. Analysis of number of reusable model

Method/ MDD technique	BPDS-BB	MINERVA	ARIS	Proposed method
Number by MDA	1	1	2	2
Number by MVC	0	0	3	3
Total number of reuse-available models (weight)	1 (20%)	1 (20%)	5 (100%)	5 (100%)

6. 결 론

현존 BPMN 비즈니스 모델링 방법들은 서비스 지향의 개발 그리고 MDA와 MVC에 기반한 계층적인 개발 프로세스가 성숙되지 않았다. 본 논문은 MDA와 MVC pattern에 의거한 서비스 지향으로 BPMN 모델을 계층적으로 모델링할 수 있는 확장된 XL-BPMN 메타모델을 제시했다. 또한, 정형화된 메타모델의 모델링 요소들을 사용한 2개 서비스 모델링 단계로 구축된 BPMN 모델링 프로세스를 정립했다. 이를 통해, 서비스 관점에서 초기 비즈니스 모델링에서 플랫폼 독립적인 서비스 디자인에 이르는 비즈니스 모델링 방

법을 제시했다. 이로서, BPMN 기반의 비즈니스 프로세스 모델링에 대한 서비스 지향의 계층적 모델의 구축을 제공할 수 있다. 따라서, BPMN 비즈니스 모델의 모듈성 강화에 따른 고 품질화와 재사용성을 제고시킬 수 있다. 아울러, 비즈니스 모델을 MVC 패턴에 의해 분리하여 모델링함으로써 어느 한쪽의 변경시, 그 수정의 영향을 최소화시켜준다. 아울러, BPMN 모델의 서비스 표현력 확장 및 계층적 접근에 의한 비즈니스 서비스의 복잡성 해소할 수 있다. 향후 연구로서, XL-BPMN 모델의 PSM 수준의 모델링 프로세스, BPEL, WSDL이 가미된 SOA 기반 비즈니스-소프트웨어 통합 모델링 프레임워크 및 프로세스, 이들 이질적 모델간 변환 알고리즘의 개발을 후속할 것이다.

참 고 문 헌

- [1] Pete Rivett, Applying MDA Standards to bring the A to SOA, Adaptive company[Internet], http://www.omg.org/news/meetings/workshops/SOA_MDA_WS_Workshop_CD/05-3_Rivett_Revised.pdf, 2006. 3.
- [2] J. Touzi, F. Benaben, H. Pingaud, “Prototype to Support Morphism between BPMN Collaborative Process Model and Collaborative SOA Architecture Model,” in Enterprise Interoperability III Springer London, pp.145-157, 2008.
- [3] Olaf Zimmermann, Jana Koehler, Frank Leymann, Architectural Decision Models as Micro-Methodology for Service-Oriented Analysis and Design [Internet], <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.3974&rep=rep1&type=pdf>, 2006.
- [4] S. H. Lee, D. S. Kang, C. Y. Song, D. K. Baik, “A Method of Test Case Generation using BPMN-based Model Reduction for Service System,” The Kips Transactions:part D, Vol.16D, No.4, pp.595-612, 2009.
- [5] M. Juric, K. Pant, “Business Process Driven SOA using BPMN and BPEL,” PACKT Publishing, 2008. 8.
- [6] OMG, Business Process Model and Notation, V1.1[Internet], http://www.omg.org/bpmn/Documents/BPMN_1-1_Specification.pdf, 2008. 1.
- [7] OMG, Business Process Model and Notation(BPMN) V2.0[Internet], <http://www.omg.org/spec/BPMN/2.0/Beta2/PDF>, 2010. 6.
- [8] Kerstrn pftzner, Gero Decker, Oiver Kopp, Fank Leymann, Web Service Choreography Configurations for BPMN [Internet], <http://www.epc-editor.org/pub/Public/GeroDecker/wesoa2007-bpmn2bpel4chor.pdf>, 2007.
- [9] National Information Society Agency(NIA), Development of a Guideline forImplementing BPMS based on BPM Standards[Internet], NCA IV-RER-05011, http://www.nia.or.kr/bbs/board_view.asp?BoardID=201111281321074458&id=2396&Order=010200&search_target=&keyword=&Flag=01000, 2005. 12.
- [10] Joaquin Miller and Jishnu Mukerji., MDA guide version 1.0.1[Internet], <http://www.omg.org/cgi-bin/doc?omg/03-06-01>, Object Management Group, Technical Report, 2003.

- [11] G. M. Kapitsaki, D. A. Kateros, G. N. Prezerakos, and I. S. Venieris, "Model-driven development of composite context-aware web applications," *Information and Software Technology*, Vol.51, No.8, pp.1244-1260, 2009.
- [12] H. G. Min and S. D. Kim, "Process and Techniques to Generate Components in MDA/CB-PIM for Automation," in *Proceedings of ICCSA'06 of the 2006 international conference on Computational Science and Its Applications*, Vol.Part IV, 2006, pp.736-745.
- [13] A. Joukhadar, "EliXir: a framework for Building e-business applications," in *proceedings of the ICTTA 2008*, 2008, pp.1-5.
- [14] H. S. Cheo, *Object oriented CBD Development Bible*, Hanbit Media, 2003.
- [15] Mike Ormerod, *Defining the OPENEDGE® reference architecture presentation: Model-view-controller pattern* [Internet], <http://communities.progress.com/pcom/docs/DOC-14628>, 2009.
- [16] Andrea Delgado, Francisco Ruiz, Ignacio Guzmán, and Mario Piattini, *A Model-driven and Service-oriented framework for the business process improvement*[Internet], <http://www.si-journal.org/index.php/JSI/article/viewFile/55/41>, 2010.
- [17] Sebastian Stein, *Modelling Method Extension for Service-Oriented Business Process Management*[Internet], Doctor's dissertation, <http://sebstein.hpfc.de/publications/stein2009phd.pdf>, 2009.
- [18] C. Y. Song, E. S. Cho, "A PIM/PSM Component Modeling Technique based on 2+1 View Integrated Metamodel," *The Kips Transactions:part D*, Vol.16D, No.3, pp.381-394, 2009. 6.
- [19] C. Y. Song, E. S. Cho, and C. J. Kim, "AN INTEGRATED GUI-BUSINESS COMPONENT MODELING METHOD FOR THE MDD-AND MVC-BASED HIERARCHICAL DESIGNS", *IJSEKE*, Vol.21, No.3, pp.447-490, June, 2011.
- [20] ebpm1, *BPMN 1.0 Metamodel*[Internet], <http://www.ebpm1.org/wesper/wesper/bpmn1.0.jpg>, 2007.
- [21] OMG, *BPMN Execution Level Metamodel*[Internet], http://www.bpmn.org/Documents/NWG-2002-01-07R6_BP MN_Metamodel.pdf, 2002.
- [22] K. W. Kim, J. W. Lee, J. H. Son, "Design Anomalies in the Business Process Modeling," *Journal of KIISE : Computing Practices and Letters*, Vol.14, No.9, pp.850-863, 2008. 12.
- [23] J. Y. Sul, J. O. Bak, H. S. Kim, Y. J. Sin, J. K. Jang, C. H. Bak, T. I. Kim, J. H. Son, "Validation Checking Mechanisms of Business Processes", Vol.14, No.9, *Proceedings of the Korea Institute of Information Scientists 2006 autumn conference*, Vol.33, No.2, 2006. 10.
- [24] S. W. Kang, D. H. Lee, Y. W. Ahn, "Extending the Abstraction Capability of BPMN by Introducing Vertical Abstraction" *The KIPS Transactions*, Vol.16, No.2, 2009. 4.
- [25] A. Rodriguez, E. Fernandez-medina, and M. Piattini, "A BPMN Extension for the Modeling of Security Requirements in Business Processes", *IEICE TRANS. INF. & SYST.*, Vol.E90-D, No.4, pp.745-752, 2007. 4.
- [26] Stephen A. White, *Using BPMN to Model a BPEL Process*[Internet], IBM BPTrends, <http://w.bptrends.com/publicationfiles/03-05%20WP%20Mapping%20BPMN%20to%20BPEL-%20White.pdf>, 2005. 3.
- [27] Chun. Ouyang, Wil Aalst, Marlons, Arthur, *Translating BPMN to BPEL*[Internet], <http://eprints.qut.edu.au/3615/>, 2006. 7.
- [28] UN/CEFACT, *UN/CEFACT Modeling Methodology (UMM) User Guide CEFACT/TMG/093* [Internet], http://www.unece.org/cefact/umm/UMM_userguide_220606.pdf, 2003. 9.
- [29] B. Hofreiter1, C. Huemer1, P. Liegl2, R. Schuster2, and M. Zapletal3, *UN/CEFACT' Modeling Methodology (UMM): A UML Profile for B2B e-Commerce*[Internet], http://www.ec.tuwien.ac.at/~zapletal/pub/A_UML_Profile_For_B2B_eCommerce.pdf, 2007.
- [30] Victor. E. Sanvido, *AN INTEGRATED BUILDING PROCESS MODEL*[Internet], http://www.engr.psu.edu/aecic/publications/TechReports/TR_001_Sanvido_1990_IBPM.pdf, 1990.
- [31] Ang Cheng-Leong; Khoo Li Pheng; Gay Robert Keng Leng, *IDEF*: a comprehensive modelling methodology for the development of manufacturing enterprise systems*[Internet], *int. j. prod. res.*, Vol.37, No.17, <http://www.plus.ie/bpr/papers/BPR%20Modelling.pdf>, 1999.

송 치 양



e-mail : cysong@knu.ac.kr
 1985년 한남대학교 전자계산학과(학사)
 1987년 중앙대학교 전자계산학과(공학석사)
 2003년 고려대학교 컴퓨터학과(이학박사)
 1990년~2005년 KT 중앙연구소 책임연구원
 2005년~2008년 상주대학교 소프트웨어 공학과 조교수

2008년~현 재 경북대학교 컴퓨터정보학부 부교수
 관심분야: 서비스 지향 모델링, 비즈니스-소프트웨어 통합 디자인, CBSE

조 은 속



e-mail : escho@seoil.ac.kr
 1993년 동의대학교 전산통계학과(학사)
 1996년 숭실대학교 컴퓨터학과(공학석사)
 2000년 숭실대학교 컴퓨터학과(공학박사)
 2000년~2005년 동덕여자대학교 정보학부 강의전임교수

2005년~현 재 서일대학 컴퓨터소프트웨어과 부교수
 관심분야: CBSE, Embedded Software, Web Service, SOA, Service-Oriented Computing