

구조 기반 BPMN 모델의 Feature 모델로 변환 기법

송치양*, 김철진²

¹경북대학교 소프트웨어학과, ²인하공업전문대학 컴퓨터시스템과

A mechanism for Converting BPMN model into Feature model based on syntax

Chee-Yang Song^{1*}, Chul-Jin Kim²

¹Department of Software, Kyungpook National University

²Department of Computer Systems & Engineering, Inha Technical College

요약 BPMN 모델로부터 휘처(Feature) 모델로 변환하는 기존 방법들이 도메인 분석가의 직관에 의존하여 자동화된 변환이 어려운바, 비즈니스 모델링 연계의 휘처 지향 개발의 활성화에 저해가 되고 있다. 본 고는 구조 기반의 BPMN 비즈니스 모델을 휘처 도메인 모델로 변환하는 방법을 제시한다. 상호 이질적인 BPMN(Business Process Modeling Notation)과 FM(Feature Model) 모델간의 변환을 위해서, 액티비티의 구조에 기반한 그룹핑 기법을 정의하고, 이들 모델의 공통 구조물인 요소(비즈니스 기능을 표현)와 구조(요소간 관계 및 프로세스)에 기반해서 모델간 변환 규칙과 방법을 정립한다. 온라인쇼핑몰 시스템을 대상으로 적용 사례를 보인다. 이로써, BPMN 모델로부터 휘처 모델로의 기계적인 혹은 자동화된 구조 변환을 도모할 수 있다.

Abstract The legacy methods for converting a business model to a feature model make it difficult to support an automatic transformation due to a dependence on a domain analyzers' intuitions, which hinders the feature oriented development for the integration of feature modeling in business modeling. This paper proposes a method for converting a BPMN business model into a feature model based on syntax. To allow the conversion between the heterogeneous models from BPMN to the FM(Feature Model), it defines the grouping mechanism based activities' syntax, and then makes translation rules and a method based on the element (represent business function) and structure (relationships and process among elements), which are common constructs of their models. This method was applied to an online shopping mall system as a case study. With this mechanism, it will help develop a mechanical or automated structure transformation from the BPMN model to the FM.

Keywords : BPMN model, Feature model, Model transformation, MetaModel

1. 서론

시스템 개발의 패러다임은 주로 객체/컴포넌트 지향의 모델링 방법을 사용하고 그 외 휘처(feature) 지향의 모델링 방법이 사용되고 있다. 객체 지향 개발은 특정 시스템에 초점을 두는 반면, 휘처 기반 개발은 범용 도메인에 모델링의 관점을 둔다. 따라서, 휘처 모델링 방법은

좀더 일반적이고 범용적인 모델을 생성하여 재사용성이 높은 이점을 갖고 있다. 그래서, 이들 개발방법들간의 연계 모델링을 위해, 휘처 모델을 UML(Unified Modeling Language) 모델로의 변환에 대한 연구가 있어왔다. 한편, 비즈니스 가치가 중요시됨에 따라 개발 패러다임도 고객 가치 중심적이고, 가변적 비즈니스 프로세스에 유연한 SOA(Service Oriented Architecture) 기반의 서비스

*Corresponding Author : Chee-Yang Song(Kyungpook Univ.)

Tel: +82-54-530-1453 email: cysong@knu.ac.kr

Received October 2, 2015

Revised December 7, 2015

Accepted January 5, 2016

Published January 31, 2016

스 지향 개발방법이 적용되고 있다. 이에, Fig. 1과 같이, BPMN(Business Process Modeling Notation), EPC(Event-driven Process Chain) 등을 사용해서 비즈니스 모델을 생성하고, 이를 휘쳐 모델(약칭 FM)로 변환하여 생성한 후, 이 모델은 UML 모델을 이용해서 특정 시스템에 적합토록 구체화되어 디자인하는 것이 이상적인 시스템의 개발 프로세스라 할 수 있다.

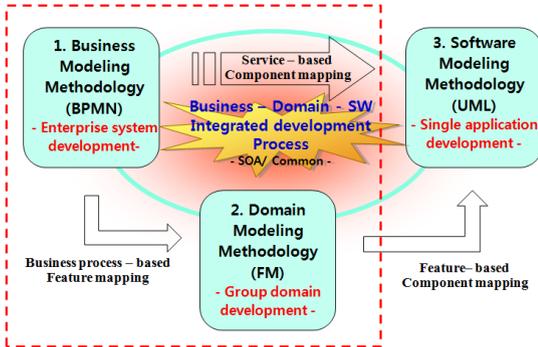


Fig. 1. Methodology based system development process

이를 좀더 세부적으로 나타낸 것이 Fig. 2이다. Fig. 2에서, 비즈니스 도메인을 대상으로 BPMN 모델로 디자인하고 이때 다수 도메인에 걸쳐 공통 테스크(task)를 식별하고, 이 테스크에 대해 휘쳐 모델로 공통 서비스와 가변 서비스로 모델링한다. UML 모델을 사용해서 특정 프로젝트별로 요구되는 공통 혹은 가변 서비스를 조합해

서 시스템을 구축한다.

기존 연구에서 비즈니스 모델링 방법(BPMN)에서 도메인 모델링 방법(Feature Model: FM, 약칭)으로의 모델간 변환을 위해 약간의 연구가 있어왔다. [1]에서, 비즈니스 프로세스(BPMN, Activity model)로부터 도메인 모델(FM)을 생성하기 위한 방법을 제시한다. 액티비티(activity)들을 유스케이스로 변경해서 이들을 그룹화하고, 이 그룹을 휘쳐로 매핑하여 휘쳐 모델을 생성한다. 이 방법은 그룹화시 주관적 판단에 의해 이루어지고, 그래서 분석가에 의존하며, 모델간 자동화 도구의 개발이 어렵다.

본 논문은 고객 관점의 비즈니스 프로세스 모델인 BPMN 모델(수평적 구조, 동적 모델)로부터, 도메인 관점의 휘쳐 모델(수직적 구조, 정적 모델)로의 변환 기법을 제시한다. 본 고의 목적은 두 모델간 변환이 분석가의 주관적 영향을 최소화해서 가능한 기계적인 변환을 도모하는 것이다. 수평적 모델을 계층적 모델로 생성하기 위해 BPMN 모델의 액티비티들을 그룹화(grouping)한다. 생성된 그룹을 대상으로 기능 요소별 그리고 관계 구조별에 의해 FM로 변환한다. 기능 요소별에 의한 방법은 기능을 표현하는 비즈니스 액티비티를 휘쳐로 변환한다. 구문적 구조에 의한 방법은 이 두 모델이 가진 구조를 가지고 변환 규칙을 정의한다. 이때, 일부 FM의 모델링 요소가 새로이 확장하여 표현한다. 이 변환 기법은 온라인쇼핑몰시스템을 대상으로 사례 적용한다. 이로서, 비

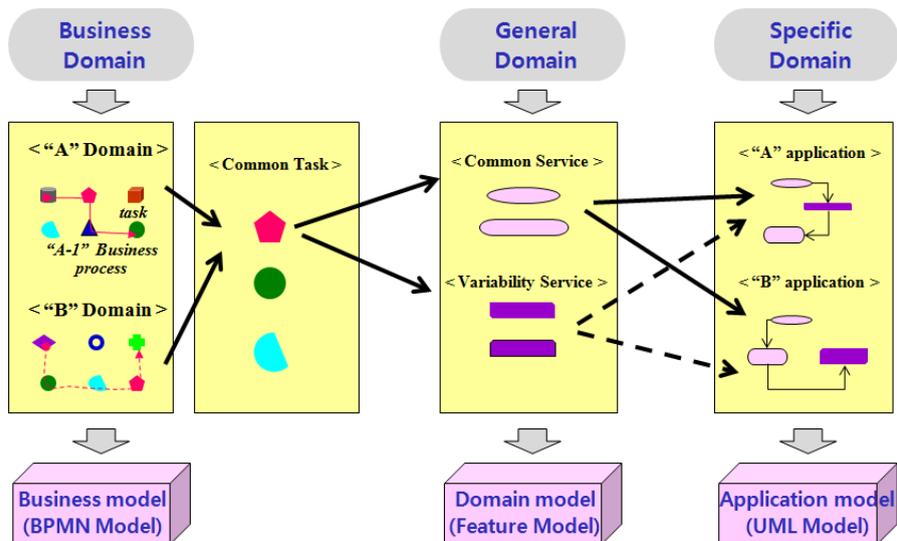


Fig. 2. Domain- and Model-based System design process

즈니스 모델로부터 휘처 모델로의 자동화된 구조 변환을 도모할 수 있으며, 비즈니스 모델링 연계의 휘처 지향 개발을 더욱 활성화할 수 있을 것이다.

본 논문의 구성은 2장에서 비즈니스 모델, 휘처 모델, 이들 모델간 변환의 연구현황을 살펴본다. 3장은 비즈니스 모델-휘처 모델간 변환규칙에 대해서, 4장에서는 온라인쇼핑몰 시스템을 대상의 적용사례를 다루고, 5장에서 본 연구의 평가를 다룬다.

2. 관련연구

2.1 BPMN 모델과 휘처 모델

BPMN[2]은 OMG(Object Management Group)의 표준으로서 비즈니스 프로세스를 디자인 할 수 있는 표준화된 시각적 표기법이다. UML의 활동 모델(activity model)보다 강력한 표현력을 갖는다. 이 모델의 구성 요소는 액티비티가 핵심이며, 관계적 요소로 AND-split, AND-join, OR-split, OR-join, XOR 등이 있다. 이러한 요소들을 가지고 정의한 BPMN V2.0의 메타모델인 [3] 그리고 확장해서 정의한 [4]가 있다. 또한, [5]에서 BPMN 모델의 구조패턴을 20개로 분류하고 패턴별 BPEL(Business Process Execution Language)과 XMI(XML Metadata Interchange) 로의 명세를 보여준다. 나아가, [4]에서 BPMN을 사용한 서비스 지향 비즈니스 모델링 방법을 제시한다.

휘처 모델은 휘처와 그들 간의 관계에 의해 SPL (Software Product Line)의 가능한 모든 제품들의 정보를 계층적 트리 구조의 형태로 표현하는 정보 모델이다 [6]. [7]에서 휘처는 “구현, 테스트되고 배포 및 유지 되어야하는 기능적, 비기능적 추상화를 말하는 것으로 요구사항이나 특징적인 기능”이라 정의한다. 휘처간 관계적 요소로 AND, OR, composed-of, generalization/specification/implemented-by, mandatory, alternative 등이 있다. [8]에서 휘처모델의 메타모델 및 그 요소를 정의했다. 휘처 기반의 모델링 방법으로 FODA(Feature-Oriented Domain Analysis)[9]이 있고, 휘처의 재사용에 기반한 FORM (Feature-Oriented Reused Method)이 있다. [10]에서 재사용에 기반한 프리덕트 라인공학 기술의 실용적 적용을 위한 방법론을 제시했다.

2.2 BPMN-Feature-UML 모델로의 변환

BPMN 모델을 휘처 모델로의 변환에 대해 [1]에서 제시했다. 이 기법은 비즈니스 프로세스로부터 도메인 모델을 생성하기 위해 고객 관점의 비즈니스 액티비티를 소프트웨어 프로덕트(product)간의 연결을 보여준다. 이 방법은 비즈니스 프로세스를 BPMN 혹은 활동 모델로 표현하고, 이 액티비티를 유스케이스로의 일대일로 변환하고, 유스케이스들을 폴딩기법에 의해 그룹화해서 휘처로 매핑하고 휘처들을 계층화해서 변환한다. 그러나, 이 기법은 분석가의 주관적 판단에 의해 그룹화함으로써 그 분석가의 의존하며, 모델 변환을 지원하는 자동화 도구의 개발이 어렵다.

휘처 모델의 UML 모델로의 변환에 대해서, [11]은 메타모델 차원에서 UML 메타모델이 휘처 모델링 요소를 표현할 수 있도록 확장했다. 즉, 휘처를 컴포넌트로의 확장 타입으로 매핑하여 표현하였다. 또한 [12]에서도 좀더 구체적으로 Product, ProductLine 그리고 휘처를 컴포넌트의 실체화 타입으로 상호 변환을 매핑하였다. 기존 UML 도구에 프리덕트라인 개발(Product line development)을 할 수 있도록 지원도구의 확장에 대해 [13]에서 다루었다. [14]는 서비스 지향 기능의 UML 모델과 비 기능의 휘처 모델로 디자인하고, 이들을 조합한 기능과 비기능이 통합된 UML 서비스 모델의 생성 방법을 제시했다.

3. BPMN 모델의 휘처 모델로 변환

비즈니스 모델의 휘처 모델로 변환은 비즈니스 액티비티/프로세스(BPMN)와 소프트웨어 프리덕트간의 통합을 의미한다. BPMN은 그래픽하게 표현하며, 플로우 차트 기반 프로세스 모델링 언어이며, 동적 비즈니스 모델인데 반해, 대상 도메인에 초점을 두는 휘처 모델은 휘처들과 그들 간의 관계로서 표현된 비정형적이고 계층적 트리 구조의 정적 도메인 정보 모델이다. 즉,이질적인 두 모델간의 변환 규칙이 필요하다.

BPMN 모델과 휘처 모델간의 변환을 위한 접근 모델은 Fig. 3과 같다. Fig. 3에서, 제시방법에 따른 절차는 어떤 시스템의 비즈니스 프로세스에 대한 BPMN 모델이 생성되면, 이 모델이 지닌 요소 및 구조에 기반해서 정의된 변환 규칙에 의해 휘처 모델을 생성하는 것이다.

(회색박스 부문 제시)

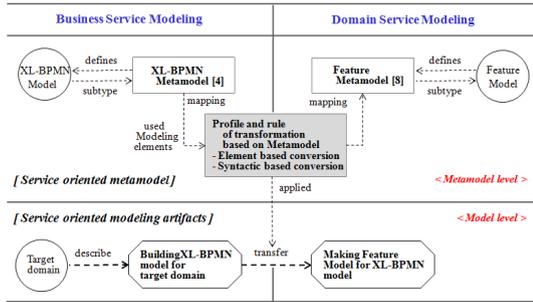


Fig. 3. Approach model for the translation between BPMN and FM

본 연구의 범주를 나타낸 것이 Fig. 4이다. Fig. 4에서 서비스 지향으로 개발 프로세스를 3 단계로 구성(CIM: Conceptual Independent Modeling, PIM: Platform Independent Modeling, PSM: Platform Dependent Modeling)하여 XL(eXtended Layered)-BPMN(이하 약칭 BPMN)에서 휘처 모델로, 이 모델은 logical component model로, 구현 모델로의 WSEL(Web Service Execution Language), WSDL(Web Service Description Language), physical component model에 이르는 모델간 변환 과정을 보여준다. 회색박스의 부문이 본 고에서 해결하고자하는 연구 범주의 영역이다.

3.1 BPMN과 휘처의 모델 정의

BPMN 모델과 휘처 모델은 목적과 구조가 상이하므

로 이들 모델들이 가진 구조물을 정의해야 한다. 각 모델은 기능을 표현하는 요소와 그 요소들간 관계하는 구조 패턴을 가지고 정의한다.

BPMN모델의 구조물을 다음과 같이 정의한다.

$$BPMN = \bigcup_{i=1}^n \{ BPMN_i \mid A_{activity}, R_{relationship} \}$$

$$\forall A_{activity} = \bigcup_{i=1}^n A_i = \bigcup_{i=1}^n \{ SA_i \}$$

$$\forall R_{relationship} = \bigcup_{J=1}^n R_j = \bigcup_{J=1}^n \{ RA_j \mid RO_j \}$$

SA_i는 service activity를 나타내며, RA_i는 relation AND를 의미하며, RO_i는 relation OR를 표현한다.

휘처 모델의 구조물에 대해 다음과 같이 정의한다.

$$FM = \bigcup_{i=1}^n \{ FM_i \mid F_{feature}, R_{relationship} \}$$

$$\forall F_{feature} = \bigcup_{i=1}^n F_i = \bigcup_{i=1}^n \{ F_i \}$$

$$\forall R_{relationship} = \bigcup_{J=1}^n R_j = \bigcup_{J=1}^n \{ RA_j \mid RO_j \mid RC_j \mid RR_j \mid RE_j \mid RRO_j \}$$

여기서, F_i는 휘처이며, RA_i는 relation AND, RO_i는 relation OR, RC_i는 relation Composed-of, RR_i는 relation Requires, RE_i는 relation Excludes를 각각 나타낸다.

결국, 두 모델이 갖는 공통적 구조물로 요소와 관계적 구조를 가진다. 따라서, 요소와 구조(관계와 프로세스)에

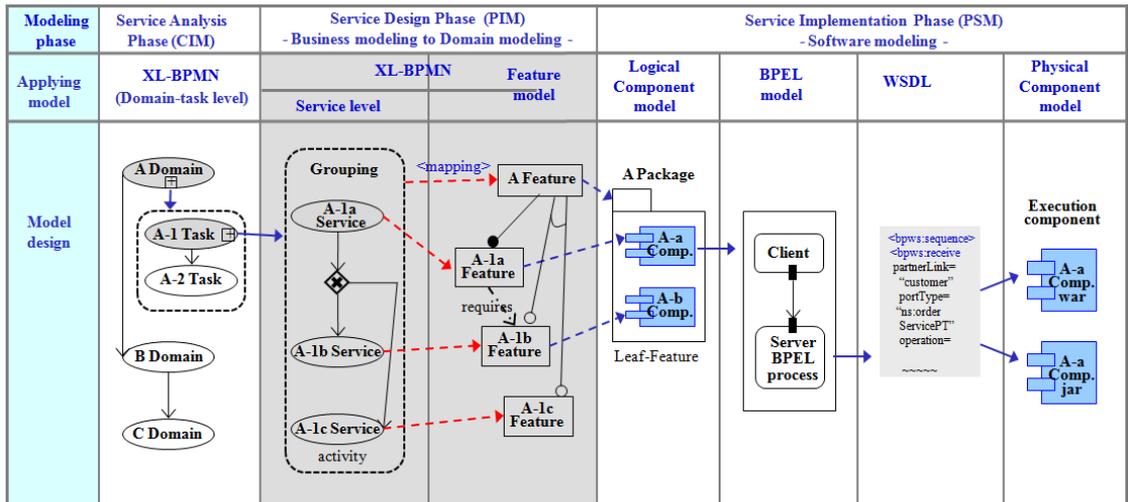


Fig. 4. Research scope in this paper

기반해서 두 모델간 변환 규칙을 정의한다. 이때, 각 모델의 구조물에 대해 모델링 요소와 관계를 정의한 XL-BPMN(약칭 BPMN)의 메타모델은 [4]에서 제시한 것을 그리고 휘처 모델의 메타모델은 [8]에서 정의한 것을 사용한다.

3.2 BPMN 모델의 휘처 모델로 변환 규칙

BPMN 비즈니스 모델을 휘처 도메인 모델로의 변환 과정은 먼저, BPMN 모델을 그룹핑 기법에 의해 계층화 해서 상위 휘처를 생성하고, 이 그룹화된 부문 모델에 대한 요소간 그리고 그들 간의 구조에 의해서 FM로 변환 하는 절차로 구성한다.

3.2.1 그룹핑 기반 BPMN 모델의 계층화

BPMN 모델은 수평적 프로세스 중심의 모델인데 반해, FM 모델은 수직적 특징(기능) 중심의 모델이다. 이에, 계층적 구조로의 상위 휘처를 생성하기 위해서, BPMN 모델상의 액티비티들간의 그룹화가 필요하다. 이때, 이 그룹을 상위 휘처로 생성한다.

규칙 3-1 (BPMN모델 그룹화, 그룹휘처생성)

- 그룹의 생성은 BPMN 모델상에서 구조 패 턴별에 기준해서 상위와 하위의 액티비티들 로 그룹화한다.
 - 구조의 패턴에는 조건(AND, XOR 등), 반복, 조합, 순차의 패턴이 있다.
 - 구조 패턴이 다단계로 nested된 경우, 다 계층의 그룹핑을 생성한다.
 - BPMN 모델에 pool, lane가 있으면, 이들 단위로 포함된 액티비티들로 그룹핑을 할 수 있다.
 - 그룹내 액티비티간에 기능적 종속 혹은 의 미적 유사성이 있다면 상기 패턴에 관계없 이 그룹을 구성 할 수 있으며, 그룹 휘처를 생성하지 않는다.
- 그룹 휘처의 생성은 그룹별로 그룹 휘처를 만 든다.
 - 그룹(상위) 휘처의 명칭은 구성되는 액티비 티들을 총칭하는 이름으로 정한다.
 - 그룹 휘처들에 대한 최상위 휘처를 생성한 다.
 - 그룹 휘처간의 관계는 BPMN 상에서의 그 룰간 관계(nested, sequence)에 준해서 변환 한다. nested 라면, 그룹 휘처들이 수직적 (다 계층) 휘처 모델 로, sequence라면 수평 적(1 계층) 휘처 모델을 생 성한다.

만일, BPMN 모델의 프로세스 구조가 순차로만 되어

있다면 상위의 그룹 휘처와 하위의 액티비티의 휘처로 구성된 2 계층을 가진 FM이 생성된다. 반면에 4장의 Fig. 10에서처럼 1개 이상의 조건 패턴을 가진 BPMN 모델의 경우는 3 계층 이상의 FM이 생성된다.

그룹핑을 위해 구조 패턴에 기반한 이유는 구조가 BPMN 모델의 근간이며, 가변성이 많은 부분을 갖고 있기 때문이다.

3.2.2 요소와 구조 기반 BPMN-FM 모델간 변환

BPMN 모델에 대해 규칙 3-1에 의거해서 생성된 각 그룹별을 대상으로 구성된 요소와 구조에 기반해서 FM 모델로 변환한다. Table 1은 BPMN 메타모델의 속성 분류에 따라서 FM로 변환시, 적용되는 변환의 유형을 보여 준다. 이때, BPMN의 각 모델링 요소들을 대상으로 변환 유형에 따라서 매핑되는 FM의 각 요소로 변환된다. 여기서, 구조 기반 변환은 BPMN 요소의 특성에 따라 다시 관계 기반과 프로세스 기반으로 분류하여 변환을 규칙화한다.

Table 1. Transformation's type to apply converting BPMN into FM based on BPMN modeling element

Large category	Small category	Modeling element	Transformation type
Flow objects	activity	Task (service level) collapsedSub-process expandedSub-process	Element based conversion [Table 2]
	gateway	AND-split (parallel), AND-join (synch), OR (inclusive), XOR (exclusive), Complex	syntactic based conversion (relations hip) [Table 3]
	Event	start, intermediate,end	Not conversion
Connection objects	sequence flow	Normal , Uncotrolled, Conditional, Default , Exception flow	syntactic based conversion (process)
	message flow	Message flow	
	association	Association	
swimlanes	pool	Pool	Not conversion
	lane	Lane	
artifacts documentation	data objects	Data object	Not conversion
	group	Group	
	annotation	Annotation	

3.2.2.1 요소 기반 BPMN의 휘쳐 모델로 변환

BPMN과 휘쳐 모델간의 요소 기반 변환은 비즈니스 혹은 도메인에서 “기능”의 특성을 나타내는 요소간의 변환을 말한다. 즉, BPMN의 "activity"(task, service level)와 FM의 "feature" 모델링 요소간의 변환을 의미한다. 이 요소들간의 변환 프로파일은 Table 2이다.

Table 2. Profile for converting BPMN into FM based on element

BPMN (Activity)	FM (Feature)	Comment
Mandatory Service	Mandatory Feature	Common function
Variable Service	Variable Feature	Optional or Alternative function
sub-Service (sub-Activity)	sub-Feature	Composed service Composed feature
A	F	Converting notation

요소 기반 BPMN과 FM간 변환을 일반화해서 정의한 것이 규칙 3-2이다.

규칙 3-2 (요소 기반 BPMN-FM 변환)

- BPMN의 “Service” Activity 요소를 휘쳐 모델의 “Feature” 요소로 변환한다.
- 한개 activity는 한개 휘쳐로 1:1로 변환한다. 다수의 activity는 다수의 휘쳐 리스트로 매핑한다.
- Service내 하부 sub-Service는 휘쳐내 sub-Feature로 매핑한다.
- 필수(공통)와 가변 Service는 각각 필수와 가변 휘쳐로 변환, 매핑한다.
- Service에 대응되는 휘쳐간 granularity 는 동일한 수준에서 일치토록 변환한다.

아울러, Table 1에서 BPMN 모델링 요소 중에 sub_process(collapsed 혹은 expanded)는 휘쳐 모델의 sub_feature로 변환한다. 이때, FM은 composition 관계를 제공하지 않기때문에, Fig. 5와 같이 새로운 모델링 요소를 추가해서 이를 표현한다. FM 모델의 요소를 확장할 경우, 이해성을 높이기 위해 가급적 BPMN의 표기법과 동일 혹은 유사한 표기법으로 정한다.

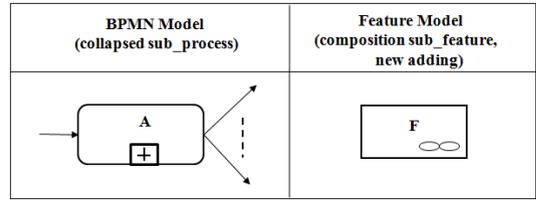


Fig. 5. Converting collapsed sub_process of BPMN into composition sub_feature of Feature model

3.2.2.2 구조 기반 BPMN의 휘쳐 모델로 변환

구조 기반의 BPMN과 FM 모델간 변환은 activity 들간의 구분적 구조인 관계와 프로세스에 의한 변환으로 정의한다.

가. 관계 구조 기반 BPMN-FM간 모델 변환

관계 구조에 의한 변환은 그래프형을 갖는 BPMN의 구조를 계층적 트리형의 휘쳐 모델 구조로 변환하는 것이다. 이 변환은 BPMN 모델에서 액티비티간의 순차 흐름을 나타내는 순차, 조건(AND, OR - 게이트웨이, Table 1), 조합에 대해서 FM 모델에 관계하는 요소들을 분석한 것이 Table 3이다.

[5]에서 BPMN의 구조 패턴으로 20개를 분류하여 정의했다. 이들을 구조 패턴의 유사성에 기반하여 집약하면 4개의 분기 패턴(AND-split, XOR-split), 동기화 패턴(AND-join, XOR-join), 반복 패턴 그리고 상호(inclusive, exclusive) 패턴으로 그룹화 할 수 있다.

Table 2의 분석을 토대로, BPMN 모델의 관계적 구조의 주요 4개 패턴들에 대해 어떻게 휘쳐 모델의 구조 패턴으로 변환을 해야 하는지에 대한 규칙을 정의한다.

규칙3-3 (분기, 동기화패턴의BPMN-FM변환)

- BPMN의 “AND-split, AND-join, XOR-split, XOR-join ” 패턴은 휘쳐 모델의 Mandatory와 composed-of의 구조로 변환한다. (Fig. 6, Fig. 7)
- 액티비티간에 상호 종속적 혹은 의미적 유사성이 있는 없는 경우는 그룹 휘쳐를 사용해서 변환하며, 반면 있는 경우는 그룹 휘쳐를 사용하지 않고 변환한다.

Table 3. Compare analysis between BPMN and Feature model with syntactic element

BPMN \ Feature Model	Mandatory	Optional	Alternative	OR	Composed-of	Generalized-by	Implemented-by	Requires	Excludes
Sequence	○								
AND-Split (Fork)	○				○				
AND-Join (Join)	○				○				
OR-Split (Decision, inclusive)		○	○	○				○	
OR-Join (Merge, inclusive)	○							○	
XOR-Split (Decision, Exclusive)		○	○	○					○
XOR-Join (Merge, Exclusive)	○								○
Complex					○				

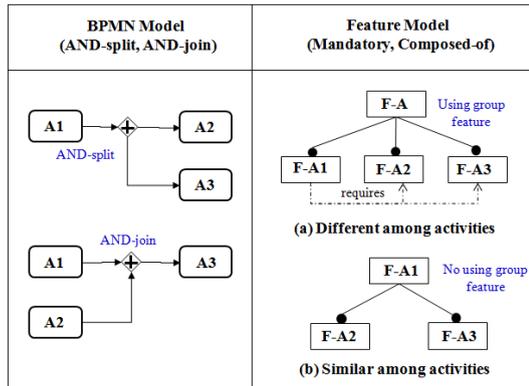


Fig. 6. Conversion of “AND-split, AND-join” pattern-based “BPMN to Feature model”

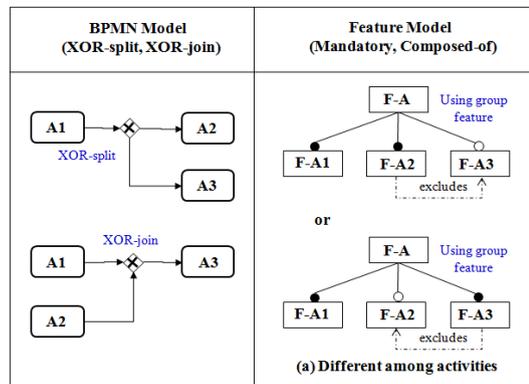


Fig. 7. Conversion of “XOR-split, XOR-join” pattern-based “BPMN to Feature model”

규칙 3-4 (반복 패턴의 BPMN-FM 변환)

- BPMN의 “반복” 패턴은 특정 activity들로 구성된 프로세스가 반복하는 것으로 FM에 새로이 반복 표기를 추가하여 표현한다. (Fig. 8)

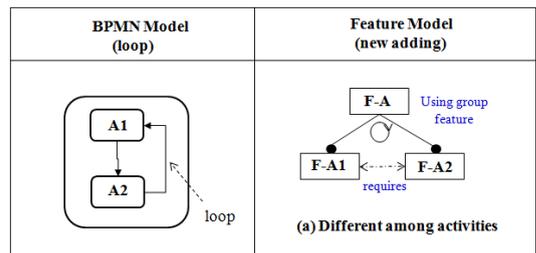


Fig. 8. Conversion of “loop” pattern-based “BPMN to Feature model”

나. 프로세스구조 기반 BPMN-FM간 모델변환 프로세스 구조에 의한 변환은 BPMN의 connection objects [Table 1]인 sequence flow, message flow, association의 프로세스에 대한 FM 구조로의 변환을 의미한다. 여기서, 프로세스란 업무흐름에 대한 작업들간의 순서를 나타낸다. 그러나, FM은 정적인 도메인 모델이기에 이러한 동적인 프로세스를 표현하지 못한다. 따라서, FM에 프로세스 구조를 표현하기 위해서 새로이 모델링 요소를 추가, 확장하여 변환한다.

규칙 3-5 (프로세스 패턴의 BPMN-FM 변환) (Fig. 9)

- BPMN의 “정상흐름”(normal flow) 패턴은 FM에 새로이 “정상흐름” 프로세스 요소 (notation)를 추가해서 변환한다.
- BPMN의 "디폴트 흐름"(default flow) 패턴 은 FM에 새로이 “디폴트 흐름” 프로세스 요소를 추가해서 변환한다.
- BPMN의 "메시지 흐름"(message flow) 패턴은 FM에 새로이 “메시지 흐름” 프로세스 요소를 추가해서 변환한다.
- BPMN의 "연관”(association) 패턴은 FM에 새로이 “연관” 프로세스 요소를 추가해서 변환한다.

Pattern name of process	BPMN Model	Feature Model (new adding)
Normal flow		
Default flow		
Message flow		
Association		

(a) Similar among activities

Fig. 9. Conversion of “default flow” pattern-based “BPMN to Feature model”

4. 적용 사례

제시된 BPMN 모델을 휘처 모델로의 변환 방법에 대해 그 실효성을 보이기 위해 온라인쇼핑몰시스템(OSMS: Online Shopping Mall System)을 대상으로 적용 사례를 보인다.

OSMS 시스템은 인터넷 온라인상에서 고객이 판매 제품을 살펴보고 구매 품목을 장바구니에 담고, 이를 주문하고, 대금을 지불하고, 제품 배송의 서비스를 제공하는 시스템이다. 이 시스템의 개발을 위한 BPMN 비즈니스 모델을 표현한 것이 Fig. 10이다[4]. 이 모델은 Domain의 task 크기의 수준을 대상으로 모델링한 것으로 1개 activity가 1개 service를 표현한다.

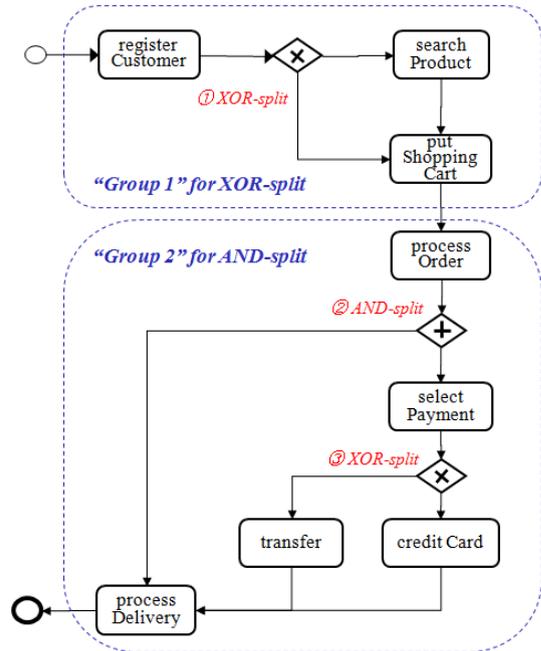


Fig. 10. BPMN model for OSMS system

이 BPMN 모델을 휘처 모델로 변환한 것이 Fig. 11이다. 먼저, 규칙 3-1에 의해, BPMN 모델의 구조 패턴에 의거해서 액티비티들을 그룹화하고 그룹 휘처를 생성한다. Fig. 10에서 구조 패턴으로 2개의 AND-split와 1개의 XOR-split가 있다. “① AND-split”는 “registerCustomer” 선행 액티비티와 후행 “searchProduct”와 “putShoppingCart” 액티비티에 관계하는 구조 패턴이다. 선행과 후행의 액티비티간에는 기능의 종속성 및 의미적 유사성이 낮으므로 변환 규칙 3-1에 의해 그룹 휘처를 생성한다. 이것이 Fig. 11의 "information" 휘처이다. ”② AND-split”의 경우도 동일하다.

그러나, “③ XOR-split”의 경우, 선행 액티비티인 selectPayment에 대해 후행 액티비티인 transfer와 creditCard가 지불의 종류이므로 상하위 액티비티간에 종속적인 관계를 가진다. 따라서, 그룹 휘처를 생성하지 않고 직접 선행 및 후행 액티비티를 대응하는 휘처들을 생성해서 BPMN과 동일한 구조로 휘처 모델을 만든다.

다음으로 이렇게 그룹화된 Group 1/ Group 2 그리고 그룹 휘처를 가지고 요소(규칙 3-2) 기반과 관계 기반의 변환 규칙(규칙 3-2 ~ 3-5)을 적용해서 FM로 변환한다.

먼저, 요소 기반 BPMN-FM 모델간 변환에 대해서,

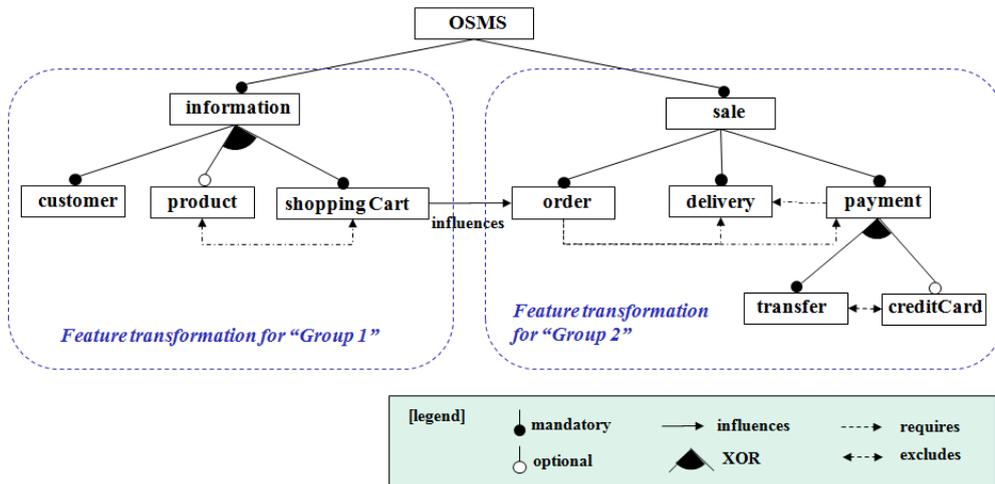


Fig. 11. The transformed Feature model for OSMS system

Table 2의 요소 기반 변환 프로파일에 의거해서, Fig. 10의 BPMN 모델에서 서비스 레벨(level)의 한 개 activity는 Fig. 11의 FM에서의 한 개 휘처로 변환해서 표현한다. 이때, 휘처는 특징적인 기능을 갖도록 이름이 부여되어야 하므로 액티비티의 이름을 일반화시킨 명사로서 휘처 명을 개명하여 부여하였다. Fig. 11의 “Feature transformation for Group 1”에서 이를 보여준다.

다음으로, 구조 기반 BPMN-FM 모델간 변환에 대해서, BPMN 모델에서 상위 “registerCustomer” 액티비티와 하위 “searchProduct”/ “putShoopingCart” 액티비티들 간에는 XOR(한개만 선택)의 구조적 관계를 가진다. 따라서, 규칙 3-3의 XOR-split 변환 규칙을 적용해서 Fig. 7에 따른다. 즉, 그룹 휘처인 “information”에 customer 필수 휘처를, product과 shoppingCart중 한 개를 필수 휘처로 해서 Group 1의 FM 모델을 구성한다. 또한, 둘 중 한 개 선택이므로 이들간 상호배제(excludes) 관계로서 FM을 생성한다.

Fig. 10에서 “③ XOR-split”의 경우, 그룹 휘처를 갖지 않으므로 Fig. 6의 (b)에 해당한다. Fig. 11의 Group 2에서처럼, “payment” 상위 휘처에 대해 “transfer”와 “creditCard”를 하부 휘처로 각각 매핑, 구성한다. 한편, BPMN 모델에서 그룹간에는 순차적인 관계를 가지므로 Fig. 9의 association 변환 규칙을 적용해서 “shoppingCart” 휘처와 “order” 휘처간에 연관관계인 influences로 하였다.

아울러, 휘처 모델의 계층적 구조를 생성하기 위해, OSMS 시스템의 최상위 휘처로서 Group 1의

“information”과 Group 2의 ”sale”에 대해 총칭하는 “OSMS” 휘처를 생성하였다.

Table 4는 요소 기반 모델간 변환에 대해서, BPMN의 activity가 휘처 모델의 휘처 리스트로 매핑되는 것을 좀더 상세히 보여준다.

Table 4. List mapping between BPMN activity and FM Feature

Activity	Feature	Group feature
registerCustomer	customer	information
searchProduct	product	
putShoopingCart	shoppingCart	
processOrder	order	sale
processDelivery	delivery	
selectPayment	payment	
transfer	transfer	
creditCard	creditCard	

5. 비교 평가

4장에서 OSMS 시스템을 대상으로 제시 기법을 적용했다. 도메인 분석가의 주관성에 무관하게, 제시된 변환 규칙을 적용해서 BPMN 모델을 기계적, 자동적으로 변경해서 휘처 모델이 생성됨을 보였다. 본 장은 기존 연구와의 비교평가 그리고 제시 방법의 한계를 다룬다.

5.1 기존 방법과의 비교평가

BPMN 모델의 FM로 변환에 대한 가장 유사한 기존 연구는 [1]이다. 이들의 접근 방법 특징을 가지고 비교한

Table 5. Comparison with approach characteristics and translation method

Evaluation items	BPM-to-FM [1]	Proposed method
Transformation process and used model	BPMN(or AC) → UCM → FM	BPMN → FM
Conversion characteristics	Identifying business process based on similarity	Element- and syntactics-based model' transformation
Generation technique for upper feature and target model	Folding mechanism with use case in UCM	Grouping mechanism with activity in BPMN
Intuitions of domain analyzers	Heavy dependency	low dependency
Accuracy of converted/generated FM	High	Middle
Supportability or implementation of conversion tool	Low possibility or Difficult	High possibility or Easy
appropriateness according to Scale of application system	Suitability to application of small scale	Suitability to application of above middle scale

것이 Table 5이다. 여기서 AC는 Activity Model, UCD는 Use Case Model를 나타낸다.

Table 5에서, [1]은 BPMN 모델을 use case model로 변환하고 use case들을 Folding(grouping)해서 상위 휘처를 생성해서 FM을 생성한다. 반면, 제시 방법은 BPMN 모델의 그룹화 및 요소/구조 기반의 변환의 특징을 갖는다. 또한, [1]은 도메인 분석가의 직관에 많이 의존하여 FM의 정확성은 높으나, 자동화 지원도구의 개발이 어렵고, 방대한 휘처 수가 발생하는 대규모 시스템의 변환에 적합하지 않다. 반면에, 제시 방법은 구조 기반의 변환 규칙에 의해서 자동화된 변환과 지원도구의 개발이 용이하여 대규모 시스템의 모델링에 적합하나, 변환된 FM의 정확도가 [1] 방법에 대비해서 낮다.

5.2 특징과 한계점

제시 방법이 갖는 특징 및 기대효과는 같습니다.

- 제시 방법은 아직까지 명확하게 정립되지 않은 BPMN 비즈니스 모델을 휘처 모델로의 직접적인 변환 방법을 제시한 것에 그 독창성을 가진다.
- 기존 방법은 BPMN 모델을 유스케이스 모델로 변환하고, 이 모델 기반의 휘처 모델을 생성하는 간접적인 변환이다.
- 제시 방법은 실질적인 BPMN 비즈니스 프로세스 모델과 휘처 도메인 모델간의 직접적인 변환이다. 이는 도메인 모델을 통한 재사용을 향상시킬 수 있으며, 또한 휘처 모델에서 UML 모델로의 개발을 활성화할 수 있다.
- BPMN 모델의 구조 기반 휘처 모델로 변환
- 메타모델 수준에서 모델 구조물의 요소와 관계/프로세스가 통합된 모델간 변환 기법을 규칙화하여

제공한다. 제시 방법과 구조물이 CASE Tool 로 제공된다면, 어플리케이션 모델간 자동화 변환 과 명확성/추적성의 검사가 가능해진다.

- 액티비티들간 구조 기반의 그룹핑 기법을 사용한 상위 계층적 휘처들을 식별한다.
- 휘처 모델의 확장을 통한 변환
- BPMN 프로세스 패턴(정상흐름 등)에 대해 휘처 모델의 모델링 요소 확장을 통해 변환을 제시한다. 즉, 휘처 모델에 비즈니스 프로세스의 개념을 표현할 수 있게 제공한다.

반면, 제시 방법의 한계점은 다음과 같이 들 수 있다.

- BPMN 모델의 구조에 기반한 자동 변환이기에 FM 모델의 정확성에 그 한계가 갖고 있다.
- 즉, 도메인 분석가에 변환된다면, OSMS에 대한 FM 구조는 OSMS 최상위 휘처에 대해 직계의 하부 형제 휘처들로서 customer, product, shoppingCart, order, payment(하부 휘처 tranfer, creditCard), delivery 휘처로 구성될 것이다. OSMS 시스템에 대해 상기의 FM 구성이 Fig. 11에 비해 좀더 정확성이 있다고 할 수 있다.
- 이에, 먼저 구조적 변환 후, 의미적인 변환을 수반해서 FM 모델을 생성하는 절충적인 변환이 이상적일 것이다. 여기서, 의미적 변환 기법은 BPMN의 액티비티를 명세하고 이것을 분석해서 그들 간의 유사성에 따라 그룹 휘처를 생성해서 계층적인 FM을 생성하는 것이다.
- BPMN 모델이 가진 모든 요소와 구조를 FM 모델로의 변환을 커버(cover)하지 못한다. 특히, 프로세스에 대한 부분의 정교한 변환 기법과 휘처 모델의 확장이 필요하다.

5. 결론

기존 BPMN 모델의 FM로 변환함에 있어서, 도메인 분석가의 직관에 의존하며, 대규모 시스템의 경우 모델 변환이 어렵고, 그리고 자동화 변환에 어려움을 안고 있다.

본고는 비즈니스 모델링과 휘처 모델링의 연계 작업을 유연하게 제공하기 위하여, 프로세스 중심의 동적이고 수평적 구조를 갖는 BPMN 비즈니스 모델을 특징 중심의 정적이고 수직적인 구조를 갖는 휘처 도메인 모델로 변환을 제시했다. 즉, BPMN 모델의 액티비티들 구조에 기반해서 그룹핑하고 그룹 휘처를 생성한다. 이어서, 그룹 대상의 요소 및 관계 패턴에 의해 하부 휘처를 생성한다. 생성된 그룹 휘처와 하부 휘처를 구조화해서 FM을 생성하였다. 이로서, BPMN-FM간 변환을 분석가의 직관에 의존하지 않으며, 기계적인 변환에 따른 자동화 도구의 개발이 가능해진다. 또한, 일원화된 비즈니스 모델링 연계의 휘처 지향 모델링을 수행할 수 있으며 휘처 지향 개발을 더욱 활성화시킬 수 있을 것이다.

이후, 의미적 분석에 의한 그룹화 기법이 추가 되어야 하며, 두 구조물간 모든 요소들에 대한 상세한 변환이 다루어져야 할 것이다.

References

- [1] J. S. Bae, S. W. Kang, "A Method to Generate a Feature Model from a Business Process Model for Business Applications," Computer and Information Technology, 7th IEEE International Conference, pp.879-884, Oct. 2007.
DOI: <http://dx.doi.org/10.1109/cit.2007.78>
- [2] OMG, Business Process Model and Notation(BPMN) V2.0, 2010. 6.
<http://www.omg.org/spec/BPMN/2.0/Beta2/PDF>,
- [3] ebpm1, BPMN 1.0 Metamodel, 2007.
<http://www.ebpm1.org/wsper/wsper/bpmn1.0.jpg>,
- [4] C. Y. Song, E. S. Cho, "An Service oriented XL-BPMN Metamodel and Business Modeling Process", KIPS, Vol.2, No.4, pp.227-238, Apr. 2013.
DOI: <http://dx.doi.org/10.3745/ksde.2013.2.4.227>
- [5] National Information Society Agency, "Development of a Guideline for Implementing BPMS based on BPM Standards," Dec. 2005.
- [6] D. Benavides, S. Segura, A. Ruiz-Cort'es, "Automated Analysis of Feature Models 20 Years Later: A Literature Review," Journal of Information Systems, March 2010.
<http://www.lsi.us.es/~dbc/en/?download=benavides10-is.pdf>,

- [7] Lee, K., Kang, K., Lee, J, "Concepts and Guidelines of Feature Modeling for Product Line Software Engineering", Software Reuse: Method, Techniques, and Tools, Springer Berlin/Heidelberg, Vol. 2319, pp. 62-77, 2002.
DOI: http://dx.doi.org/10.1007/3-540-46020-9_5
- [8] C. Y. Song, E. S. Cho, C. j. Kim, "A Formal Specification and Checking Technique of Feature model using Z language", KSCI, Vol.18, No.1, pp.123-136, Apr. 2013.
DOI: <http://dx.doi.org/10.9708/jksci.2013.18.1.123>
- [9] K. Kang, S. Cohen, J. Hess, W. Nowak, and S. Peterson, "Feature-Oriented Domain Analysis (FODA) Feasibility Study", TechnicalReportCMU/ SEI-90-TR-21, Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University, Nov. 1990.
- [10] National IT Industry Promotion Agency, "Practice guide for Product line engineering technology based on software reused", Sep. 2007.
- [11] V. Vrani'c, J. Snirc, "Integrating Feature Modeling into UML," 2006.
<http://www2.fiit.stuba.sk/~vranic/pub/FM-UML.pdf>.
- [12] T. Possompès, C. Dony, M. Huchard, H. Rey, C. Tibermacine, X. Vasques, "A UML profile for feature diagrams: Initiating a model driven engineering approach for software product lines," published in Journee Lignes de Produits, France, Dec 2010.
http://www.rider-project.com/sites/default/files/AUMLprofileforfeaturediagrams_0.pdf
- [13] Maria Papendieck, "Improving Usability of UML Modeling Tools for Feature-Based Product Line Development, Bachelor Thesis, Jan. 2011.
<http://www.iti.cs.uni-magdeburg.de/.../thesisPapendieck.pdf>.
- [14] H. Wada, J. Suzuki, K. Oba, "A Feature Modeling Support for Non-Functional Constraints in Service Oriented Architecture", IEEE International Conference(SCC 2007), pp.187-195, July 2007.
DOI: <http://dx.doi.org/10.1109/scc.2007.5>

송치양(Chee-Yang Song)

[정회원]



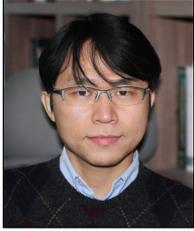
- 2003년 8월 : 고려대학교 컴퓨터학과(이학박사)
- 1990년 5월 ~ 2005년 9월 : KT 중앙연구소 책임연구원
- 2005년 10월 ~ 2008년 2월 : 상주대학교 소프트웨어공학과 조교수
- 2008년 3월 ~ 현재 : 경북대학교 소프트웨어학과 부교수

<관심분야>

CBD, MDA, Service-Oriented Modeling, BPMN 모델링, 휘처 지향 개발

김 철 진(Chul-Jin Kim)

[종신회원]



- 2004년 2월 : 숭실대학교 대학원 컴퓨터학과 (공학박사)
- 2004년 3월 ~ 2009년 2월 : 삼성전자 책임연구원
- 2009년 3월 ~ 현재 : 인하공전 컴퓨터시스템과 교수

<관심분야>

컴포넌트 기반 개발 방법론, 컴포넌트 커스터마이제이션, 모바일 서비스, 클라우드 컴퓨팅