Track 12: SATA Workshop

A Study on Preventative Measures for Ninja Hacking ...........................................552
Byungho Park, R. Young Chul Kim and Youn B. Park

An Efficient I/O Interface Control Block Design Methodology for
Application-Specific MPSoC Platforms .................................................................555
Myoung-Seo Kim and Jean-Luc Gaudiot

Applying MapReduce Programming Model for Handling Scientific Problems ...........559
Yun Hee Kang and Young B. Park

Design and Development of Multimedia Cultural Contents for Virtual Reality -
Gogurye Ancient Tomb Mural ..............................................................................561
YuKyong Lee, Yong-Hwan Lee, Sung-Ho Hong,
Youngseop Kim and Chang-June Kim

Design of System Framework for Searching Medical Information .........................563
Sung-Ho Hong, Youngseop Kim, Yong-Hwan Lee and Je-Ho Park

Dynamic Configuration of SSD File Management .................................................565
Hyuk-Kyu Lim and Je-Ho Park

Effective Fat Quantification Using Improved Least-Square Fit at High-Field MRI ..........568
Sung-Jong Eun and Taeg-Keun Whangbo

Generation of Metadata Using Content-Based Image Retrieval System ..................570
Sun-A Lee, Min-Uk Kim and Kyoungro Yoon

Goal-Oriented Requirements Analysis Using Goal Accumulation RDF Graph ...........572
Jun H. Lee and Young B. Park

Image Recognition System That Uses Visual Word ..............................................575
Min-Uk Kim and Kyoungro Yoon

Implementation of a Continuous Playing System Based on RESTful Web Services ....577
Chompoo Suppatoomsin, Apiradee Ampawasiri and Cheong-Ghil Kim

Metadata Schema for Augmented Reality .........................................................579
Takaaki Ishikawa and Je-Ho Park

Model Transformation Rule for Generating Database Schema ..............................581
Chae Yun Seo, R. Young Chul Kim and Young B. Park

Modeling and Simulation for Embedded Software System ....................................584
So Young Moon, Bo Kyung Park, R. Young Chul Kim
and Young B. Park
Model Transformation Rule for generating Database Schema

Chae Yun Seo
Dept. of Computer & Information Communication, Hongik University, Sejong Campus, 339-701 Korea
chyun@selab.hongik.ac.kr

Young B. PARK
Dept. of Computer Science, Dankook University, 330-714 Korea
ybpark@dankook.ac.kr

R. Young Chul Kim
Dept. of Computer & Information Communication, Hongik University, Sejong Campus, 339-701 Korea
bob@selab.hongik.ac.kr

Abstract—The previous researches had developed business process frameworks for easily integrating the separating data processing and managing, and decision-support systems built by different times in different places, but never mentioned how to develop this complex business process structures, that is, six-layer architecture. This paper suggests how automatically to develop a whole database schema of business process framework. To do this, we apply with Model-To-Text transformation based on metamodel to automatically build the schema based business process model. This procedure is follows: 1) defining each metamodel of the entire structure and of database schema, and 2) also defining model transformation rules for it. With of model transformation rules of this procedure, we can automatically transform through meta-modeling of an integrated information system to the schema based model information table specification defined of the entire layer.

Keywords—Model Driven Architecture (MDA); UML; Metamodel; Model Transformation; Business Process Framework (BPF); Query Language; BPSQL; Model-To-Text Transformation Language

I. INTRODUCTION

An enterprise needs to provide with a business-integrated system framework to quickly change and adapt new business. It is to use the framework possible to easily integrate the system by different teams at different time in different places [1]. Most enterprises have a computer system to efficiently operate systematic information, and also to appropriately preserve/manage it, which consists of closed layer architecture [2]. The closed architecture is based on the layer mechanism, and directly access right under the layer. Seo and Kim[3] suggested and defined five layer structure based on the closed architecture. Seo[1] also suggested to reuse the existing software component for reducing development time and cost with mapping CBD(component based development) and BPM(Business process modeling)[2]. On the previous proposed business process framework based on closed architecture, we defined BPSQL(Business Process Structured Query Language)[6], and showed to retrieve and access information between each layer with the simple associated query statements. In this moment, we need to store each data, but manually develop the structure of each layer on BPF until now. With this database schema, we can create a whole business process framework. Therefore, we suggest how to generate the schema-based business process model with the complex business process framework based on a closed architecture. To do this, we study automatically to generate database schema with BPF modeling information, which 1) defines metamodel of the whole structure for Business process framework, 2) models our own target business based on the metamodel, 3) and also needs M2T (model-to-text) transformation rules for automatic BPF schema creation. In this time, it is limited that this research represents Business process framework with just XMI data without UI and visualization.

This paper describes as follows: chapter 2 mentions related work, chapter 3 describes how to design BPF with a metamodel chapter 4 describes transformation rule with ACCELEO and shows M2T transformation rules for creating automatic BPF schema, and conclusion.

II. THEORY

A. M2T(Model To Text Transformation Language)

MOF Model to Text Transformation Language(Mo2Text or MOFM2T) is an Object Management Group(OMG) specification for a model transformation language. Specifically, it can be used to express transformations that transform a model into text (M2T), for example a platform-specific model into source code or documentation. MOFM2T is one part of OMG's Model-driven architecture (MDA) and reuses many concepts of MOF, OMG's metamodeling architecture. Whereas
MOFM2T is used for expressing M2T transformations, OMG's QVT is used for expressing M2M transformations [5].

I) Acceleo

Acceleo is an open-source code generator from the Eclipse Foundation that allows people to use a model-driven approach to building applications. It is an implementation of the "MOFM2T" standard, from the Object Management Group (OMG), for performing model-to-text transformation [4].

III. BUSINESS PROCESS FRAMEWORK METAMODEL

The A BPF is comprised of five layers: business rules, business processes, services, components, and data modeling. A repository like DB tabulation is present at each layer [1]. A BPF is a closed architecture where each layer is directly connected to the next layer.

The layered structure can quickly produce new services by reusing existing components when business demands exist. New businesses can be configured with this service. The repository at each layer is tabulated, and a layer generates data query using BPSQL [2]. Finally, the required data are extracted.

A BPF metamodel defines the essential elements, grammar, and structure of UML metamodel which creates the business process framework model. For model transformation, Source Metamodel is designed based on BPF Metamodel. Figure 1 is the Business Process Framework Metamodel. It is a detailed description of the BPF metamodel as follows.

![Fig. 1. Business Process Framework Metamodel](image)

Transformation model, the appropriate Source Model produces the XML data. Table1 is the XML data of the BPF metamodeling about Rule Layer, Process Layer, Service Layer, and Component Layer.

![Fig. 2. Transformation Rule with Acceleo](image)

XMI elements "<layer xsi:type="bpmetamodel:__Layer">" in each layer are separated. The DownReference is associated with the sub-layer, the UpReference is connected with the upper layer. <link> is connected within the equivalent layer, the source and target is distinguished as input and output respectively. <node> is the name of each layer, which describes the role of <node>.

IV. TRANSFORMATION RULE WITH ACCELEO

We transformed the XML data (which is obtained from the result value of Modeling) into SQL schema with a conversion tool, and use transformation rule with the Acceleo. Figure 2 shows Transformation rule with Acceleo. We store SQL transformation with each model by the elements in XMI data.

This approach automatically generates SQL schema structure with the modeled BPF structure is generated. All layer information is automatically stored in schema structure.
V. CONCLUSION

Our previous approaches do not focus on how to automatically develop the whole Business process framework. In this paper, nobody we suggest ModelToText transformation based on metamodel to automatically build the schema based business process model. This procedure is follows: 1) defining each meta-model of the entire structure and of database schema, and 2) also defining model transformation rules for it. With of model transformation rules of this procedure, we can automatically transform through meta-modeling of an integrated information system to the schema based model information table specification defined of the entire layer. Therefore, we can easily and automatically build the whole DB schematic with Model transformation technique.

ACKNOWLEDGMENT

This work was supported by the IT R&D Program of MKE/KEIT [10035708, "The Development of CPS(Cyber-Physical Systems) Core Technologies for High Confidential Autonomic Control Software"] and Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2013R1A1A2011601)

REFERENCES


ICITCS 5th will be held in Beijing, China from October 27th-29th, 2014. This will also include joint conferences from ICISST, ICMWT and ICIEEM. Beijing is a city of great majestic history and has endless activities for anyone to enjoy. We are excited to invite you and your fellow scholars to come join us for a new experience.

ICISA2015
http://icisa2015.org/

Come join us for the 6th ICISA from January 5th-7th, 2015. ICISA2015 will be held in Las Vegas, USA the City of Lights. Las Vegas has everything from amazing views, performances, food and everything for the family. ICISA will have scholars from all over the world and will be the perfect time to come and participate in the conference.

ICISA 2014
Technically Co-Sponsored by IEEE Computer Society

Technically Co-Sponsored by

IEEE

computer society