

Volume 4
Number 1
2015

Advanced and Applied Convergence Letters

AACL 04

Advanced and Applied Convergence

**1st International Joint Conference, IJCC 2015
Ho Chi Minh City, Vietnam, February 2015
Revised Selected Papers**

 **IIBC**

 **IPACT**

Mobile-Assisted Pronunciation Training based on Mispronunciation Association Rule / 100

Jinhyuk Jo, Jaesung Lee, Dae-Won Kim

A Robust Cloud Service Processing for Healthcare Big Data System / 104

Seung-Hyun Lee, Min-Sun Lee, Dong-Ryeol Shin

PHR (Personal Healthcare Record) Platform in Smart Healthcare Service / 106

Seung-Hyun Lee, Dong-Ryeol Shin

Application of Bond Valence Method to Estimate the Valence Charge Distribution in the Metal-to-Oxygen Bonding Spheres in Perovskites / 108

Hoang Nam Nhat, Dinh Van Chau, Dinh Van Thuong, Nguyen Thi Hang

SoundSource Localization for User Content Creation using Eigen Value Ratio and Correlation of Channels from Mixed Stereo Signal / 109

Chai-Jong Song, Chang-Mo Yang, Sung-Ju Park, Taek-Jin Han, Hochong Park

Web-based Multi-view 3D Real-time Display System / 113

Jung-Hwan Ko

Improve Energy Efficiency SEP for Wireless Sensor Networks / 114

Young-Il Song, WooSuk LEE, SeaYoung Park, Yong Min Kim, Noriyuki Iwane, Kyedong Jung, Jong-Yong Lee

Cooperation System based on Social for Business Process in Cloud / 115

Seok-Jae Moon, Chi-Gon Hwang, Jong-Yong Lee, Kye-Dong Jung

A Study on DBaaS System for Healthcare Information Data Integration in Cloud Environment / 117

Kwang-Cheol Lee, Sungbin Park, Yean-Woo Jung, Yan-Sheng Zhang, Jong-Yong Lee, Kye-Dong Jung

Improving Test Maturity Level for Test Organization Based on TPI next / 119

Kidu Kim and R. Young Chul Kim

Energy Aware based Multi-path Routing Protocol / 122

Seongsoo Cho, Seong Ro Lee, Jong-Yong Lee

Correlation Function Design for Unambiguous CBOC(6,1,1/11) Signal Tracking / 124

Keunhong Chae, Seong Ro lee, Seokho Yoon

Automatic Translator to Bi-directly Interchange XMI with XML on Model Transformation for Heterogeneous Smartphone UIs / 129

Hyun Seung Son, R. Young Chul Kim

Wideband Meanderline Bandstop Filter for X-band Application / 132

Bhanu Shrestha, Nam-Young Kim

Automatic Translator to Bi-directly Interchange XMI with XML on Model Transformation for Heterogeneous Smartphone UIs

Hyun Seung Son*, and R. Young Chul Kim*[†]

* SE Lab, Dept. of CIC(Computer and Information Communication), Hongik University,
Sejong Campus, 339-701, Korea
e-mail : {son, bob[†]}@selab.hongik.ac.kr

Abstract

When it performs model transformation from a model to another model in MDD paradigm, the transformation engine translates the XMI file from the XML file of model. But, for adopting the MDD into heterogeneous Smartphone mechanism, developing User Interface (UI) of different Smartphone generally absolutely uses the type of XML file. Therefore, bi-directly interchanging XMI with XML is required for heterogeneous Smartphone UI model transformation. In this paper, we propose the bi-direction translator to possibly interchange the XML to XMI or the XMI to XML for using UI model files in the smartphone platform. Our proposed translator is implemented by program language as C++. It can automatically translate the XMI from XML file of UI Model, and the reverse direction using two steps.

Keywords: Model Transformation, Smartphone, Metamodel, User Interface, Reusability, Multi-Platform.

1. Introduction

Model transformation is a core technology of Model Driven Development (MDD) [1], which is able to transform the Platform Specify Model (PSM) from the Platform In-dependent Model (PIM). The model transformation consists of three parts: 1) meta-model of model, transformation language, and transformation engine. Transforming model perform three steps: 1) to design metamodel, 2) to write rule of transformation, and 3) to execute model transformation. In the first step of metamodel design, architect draws the metamodel to understand the structure of models of input/output. In the second step to write rule of transformation, architect makes the rule to analyze the similarities & differences between two models, and writes code using transformation language. In the last step to execute model transformation, the engine inputted with data of model, metamodel, and transformation language creates the transformed model as command of transformation language. The all file type in model transformation using data of model and metamodel is XML-based Metadata Interchange (XMI) [2].

In order to generate the full code with the result of our research [3-8], three elements are required for model transformation such as the UI to configure the mobile screen, the code that performs functions, and the project to configure the development environment. So, we consider the transformation about UI Model in the model transformation of smartphone platform. But the UI Model saves the file's name as eXtensible Markup Language (XML). It has a problem not to immediately apply in the model transformation using XMI. In this paper, we focus on bi-directly interchanging the file type (XML to XMI, or XMI to XML) of UI Model to apply the model transformation. And we implement the XMI translator as C++. The proposed translator consists of two steps: 1) to move from value of XML to attribute value of XML because XMI is not possible to use value of XML, 2) to insert the namespace of metamodel to the root node of XML because XMI require the namespace of metamodel. Our translator can help to perform the model transformation.

This paper is organized as follows. Chapter 2 mentions the method of bi-directly interchanging XMI with XML. Chapter 3 describes a case study, and Last chapter mentions the conclusion and future work.

2. An Interchanging Method for XMI with XML

The UI of each platform (Android, iPhone, MS Phone) is expressed as the XML file. It looks the same for the file type, but is not in fact. Nevertheless, the UI file of platforms has the common information such as name, position, handler name, etc. The common UI model is defined by abstracting this information, which is used to transfer the UI models of each platform. But, model transformation requires the XMI file type. Therefore, the XMI translator is needed to perform model transformation.

To illustrate these ideas, it shows a model transformation scenario of smartphone UI in which a model (UI model) are connected by a model transformation that generate a model (UI Model) like figure 1. But XML is the file type of input model, and XMI is the model transformation of needed model. It does no match to perform the model transformation. Therefore, we propose the bi-direction translator to possibly interchange the XML to XMI or XMI to XML for UI model files in Smartphone platform. The proposed translator consists of two steps: 1) to move from value of XML to attribute value of XML because XMI is not possible to use value of XML, 2) to insert the namespace of metamodel to the root node of XML because XMI requires the namespace of metamodel.

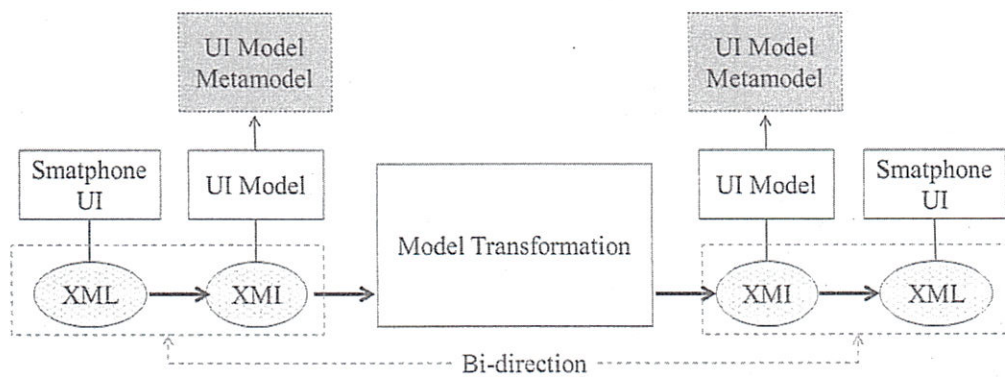


Figure 1. The model transformation scenario of smartphone UI

3. A Case Study

It needs the two steps to bi-directly interchange XMI with XML. In the First step, it moves from value of XML to attribute value of XML. In the Second step inserts the namespace of metamodel to the root node of XML. The method of first step finds the all element of XML, and inserts the attribute in the found element named "XML_VALUE" which is brought the value of element. The XML_VALUE is not exist to metamodel of smartphone UI. Therefore it is necessary in metamodel to insert the attribute named "XML_VALUE" of string type. Figure 2 shows the result on performing XMI translator.

<pre> ... <object class="NSColor" key="IBUIBackgroundColor"> <int key="NSColorSpace">3</int> <bytes key="NSWhite">MC43NOA</bytes> <object class="NSColorSpace" key="NSCustomColorSpace"> <int key="NSID">2</int> </object> </object> </object> ... </pre>	<pre> ... <object class="NSColor" key="IBUIBackgroundColor"> <int key="NSColorSpace" XML_VALUE="3" /> <bytes key="NSWhite" XML_VALUE="MC43NOA" /> <object class="NSColorSpace" key="NSCustomColorSpace"> <int key="NSID" XML_VALUE="2" /> </object> </object> </object> ... </pre>
(a) Before transformation	(b) After transformation

Figure 2. The result of translating the attribute from value in iPhone UI file

In the method of second step, linking the metamodel with model, it inserts the attributes (such as the namespace of metamodel, XMI version, and XMI namespace) into the root node in figure 3. The considerable point in this step is that the name of namespace inserted in XML should be same with the name in metamodel.

<pre><archive type="CocoaTouch.XIB" version="7.10"> ... </archive></pre>	<pre><iphonem:Archive type=" CocoaTouch.XIB" version="7.10" xmi:version="2.0" xmlns:xmi=http://www.omg.org/XMI_ xmlns:iphonem=http://iphonem/1.0 > ... </iphonem:Archive></pre>
(a) Before transformation	(b) After transformation

Figure 3. The result of inserting the namespace.

4. Conclusions

The model transformation is the best solution for heterogeneous smartphone platform at the same time. But three elements are required for model transformation in smartphone platforms such as the UI to configure the mobile screen, the code that performs functions, and the project to configure the development environment. In this paper, to generate the full code with the result of our research, we propose the bi-direction translator to possibly interchange the XML to XMI or XMI to XML for using UI model files in each smartphone platform. The proposed translator is implemented by program language as C++. It is able to automatically translate the XMI from XML file of UI Model, and the reverse direction using two steps. Further research should be conducted, which is not dealt in this study on model and UI transformation for heterogeneous platform. Our translator can help to perform the model transformation in further research.

Acknowledgments. This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. 2012M3C4A7033348) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2013R1A1A2011601).

References

- [1] Bran Selic, "The pragmatics of model-driven development", *Software, IEEE*, Vol. 20, Issue 5, pp. 19-25, 2003.
- [2] OMG, MOF 2.0/XMI Mapping, version 2.1.1, formal/2007-12-01
- [3] Woo Yeol Kim, Hyun Seung Son, Jae Seung Kim, Robert YoungChul Kim, "Development of Windows Mobile Applications using Model Transformation techniques", *Journal of KIISE : Computing Practices and Letters*, Vol. 16, No. 11, pp. 1091-1095, 2010.
- [4] Woo Yeol Kim, Hyun Seung Son, Robert Young Chul Kim, "A Study on UML Model convergence Using Model Transformation Technique for Heterogeneous SmartPhone Application", *Software Engineering, Business Continuity, and Education*, CCIS 257, pp. 292-297, Dec. 2011.
- [5] Woo Yeol Kim, Hyun Seung Son, R. Young Chul Kim, "Rule Extraction Method for Model Transformations in Heterogeneous Smartphone Applications", *Information Journal*, Vol.16, No.1(B), pp. 615-626, Jan. 2013.
- [6] Hyun Seung Son, Woo Yeol Kim, R. Young Chul Kim, "Concretization of the Structural and Behavioral Models based on model Transformation Paradigm for Heterogeneous Mobile Software", *International Journal of Software Engineering and Its Applications*, Vol. 7, No. 4, pp. 389-399, Jul. 2013.
- [7] Hyun Seung Son, Jae Seung Kim, R. Young Chul Kim, "SMTL Oriented Model Transformation Mechanism for Heterogeneous Smart Mobile Models", *International Journal of Software Engineering and Its Applications*, Vol.7, No.3, pp. 323-331, May. 2013.
- [8] Hyun Seung Son, Woo Yeol Kim, R. Young Chul kim, "Metamodel Design for Model Transformation of Heterogeneous Smartphone UI", in *Proceedings of the 16th Koera Conference on Software Engineering (KCSE 2014)*, Vol. 16, No. 1, pp. 173-180, Feb. 2014.
- [9] Hyun Seung Son, Woo Yeol Kim, R. Young Chul kim, "An Automatic UI Code generation for Android Mobile Platform," in *Proceedings of Advanced and Applied Convergence & Advanced Culture Technology*, AACL 03, pp. 151-154, 2014.