

Jeong Jin Kang  
Edward J. Rothwell  
Yonghaol Eos

**Advanced and Applied Convergence Letters**

**AACL 03**

# **Advanced and Applied Convergence & Advanced Culture Technology**

**2nd International Symposium, ISAAC 2014  
in conjunction with ICACT 2014  
Jeju, Korea, November 2014  
Revised Selected Papers**





- The Influence of PTC & NTC using Watching 3D Display / 124  
Jung Ho Kim, Deok-Young Yun, Kwang Chul Son, Seung Hyun Lee, Soon Chul Kwon
- High Quality Depth Map Generation by Gradual Pixel Bundling Method at TOF Image Sensor / 126  
Soon Chul Kwon, Ho Byung Chae, Sung Jun Lee, Kwang Chul Son, Seung Hyun Lee
- Proposed Home DCP Model using open Source Codec / 127  
Sunil Pardeshi, Soon Chul Kwon, Seung Hyun Lee, Alaric Hamacher
- Implementation of Context-awareness Reasoning based Intelligent SoC / 128  
BonJae Koo, DongHyuk Lim, SungHyun Yang
- Multiple License Plates Recognition System Implementation using HD CCTVs / 132  
Hyunwook Kim, Jonghoon Kim
- A Rule-making Method Using State Diagram for Rule-Based Context-Aware System in Smart Home Environment / 138  
Jun-Hwan Jang, Hyungyu Seo, SungHyun Yang
- Implementation of Boolean Control Network Based Intelligent System in Smart Home Environment / 142  
M. Humayun Kabir, M. Robiul Hoque, SungHyun Yang
- Enhancing Music Quality over IP Network / 145  
Qing-fu Han, Jun-yong Lee, Hyoung-gook Kim
- Energy Efficiency Hierarchy Multi-hop Routing Protocol for Wireless Sensor Network / 148  
Young-Il Song, Jongsup Lee, KyeDong Jung, Jong-Yong Lee
- The Routing Technology of Wireless Sensor Networks Using the Stochastic Cluster Head Selection Method / 149  
Young Min Kim, Choel Lee, Kyedong Jung, Jong-Yong Lee, Abdul Hanan Abdullah
- Improving Wireless Sensor Network Lifetime based on LEACH-C Protocol / 150  
WooSuk Lee, Bhanu Shrestha, KyeDong Jung, Jong-Yong Lee, Lei Liu
- An Automatic UI Code generation for Android Mobile Platform / 151  
Hyun Seung Son, Woo Yeol Kim, R. Young Chul Kim
- A Lightweight Finger Tracking Algorithm for Human Computer Interaction / 155  
Byungkook Jeon, Sungkuk Cho
- A Survey on Development Tools for Heterogeneous Smart Learning Contents / 158  
Woo Yeol Kim



# An Automatic UI Code generation for Android Mobile Platform

Hyun Seung Son\*, Woo Yeol Kim\*\*, 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*

*\*\*Department of Computer Education, Daegu National University of Education,  
Daegu, 705-715, Korea*

*e-mail : john@dnue.ac.kr*

## Abstract

*On heterogeneous mobile platforms (Android, iPhone, MS phone) it should develop each suitable software of a same service for each platform. Therefore, it may be very big issue how to reduce lifecycle time of software development. To solve this issue, one of solutions will be the automatic code generation for even simple or huge mobile software. Specially, it can reduce the unnecessary code developing time of mobile software development, and also improve the good quality of the software through reusing the verified codes. In this paper, we propose our method automatically to generate User Interface (UI) code in a special Android platform. The proposed method consists of 1) the transformation step of transforming class diagram from the abstracted UI model, and 2) the generation step of generating UI code from the class diagram. Our approach is possible to rapidly develop the smartphone application by reducing the amount of programming code which a developer should develop.*

**Keywords:** Automatic code generation, User Interface (UI), Reusability, Android, Model Driven Development (MDD)

## 1. Introduction

The previous mobile phone applications had small size and function due to limited resources such as the lack of memory, hardware speed, and etc. But the current smartphone applications gradually increase the amount of program, and the smartphone market also becomes popular in the world, which is similar hardware performance of smartphone with Personal Computer (PC) [1]. In this environment, rapid software development is still required.

The effective method for rapid development is likely to use the automatic tools with reusability in software. In particular, the automatic code generation gives to shorten development time because it automatically generates code and reduces unnecessary coding time of developers. Also, it can improve the good quality of the software through re-using the verified codes. To achieve them, the model transformation will be a core technology in Model Driven Development (MDD) [2] that automatically transfers platform specific model (PSM) from platform independent model (PIM). The model transformation consists of model-to-model, and model-to-text.

We have performed our research to adapt this model transformation into smartphone development [3-7] such as transferring a model from a model with ATLAS Transformation Language (ATL) [8] and transferring a code from a model with Acceleo [9] & code template [10]. On the result of our research, we confirmed the possibility that the model transformation automatically converts a code from the model. Recently, we have been studying another method to generate a code from sequence diagrams [11] and a



model transformation language [12].

In order to generate the full code with the result of our research, three elements of model transformation is required 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, we are limited to propose the automatic generation method of UI code in Android platform to improve and extend a part of our research. The proposed method is consisted of two steps: 1) transfer class diagram from the abstracted UI model and 2) generate UI code from the class diagram. If the method is applied, it is possible to rapidly develop the smartphone application by reducing the amount of code that the programmer must write.

This paper is organized as follows. Chapter 2 explains related studies, including the basic concept of Model-to-Text transformation. Chapter 3 mentions the method of UI code generation for Android platform. Chapter 4 describes a case study to show code generation process. Last chapter mentions the conclusion and future work.

## 2. Related Works

Model-to-Text transformation is a method to automatically form a code from the model. This method algorithm is classified with two type mechanisms: visitor-based and template-based [13]. The visitor based mechanism is composed of purifying the internal expression of model tree and writing text in text stream. The object-oriented framework such as Jamba provides the assembly of classes that represent UML model [14]. Template-based method is more similar with the code than the visitor mechanism. This is easily used in the repeated development of template. Template such as Aceleo [9] can represent text work due to the approach presented in section as well as code pieces within text that are incorrect in syntax or meaning.

## 3. A Method of UI Code Generation

The UML class diagram just generates skeleton code due to including just structural information, and needs additional written code. Therefore to generate more detailed code, class diagram with additional information is required. We proposed the UI model [15] abstracting UI component of Android platform to generate the full code of UI. Because UI model consists of the structural information of UI component, the additional information of event handler is required to generate the code for the behavioral information. To generate the full code of UI, we make profiling the information of the event handlers. The code generator uses this profiles to generate Java code. Therefore, a whole UI model transformation process like figure 1 consists of two steps: 1) to transfer class diagram from the abstracted UI model, and 2) to generate UI code from the class diagram.

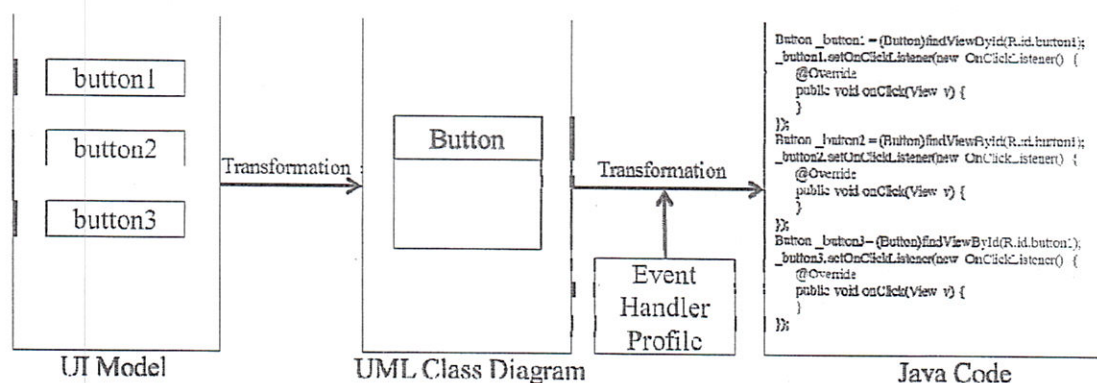


Figure 1. A method of code generation for User Interface (UI) in Android platform



#### 4. Case Study

To experiment the proposed method, we make an application of Android platform performing some process to load the file. Figure 2(a) shows to design an UI Model of this application. Figure 2(b) shows to transfer class diagram from the UI Model. The result of this transformation is created an Edit class and four Button classes. Each UI components is separated by stereo type such that a Button represents `<<ui_button>>`, an Edit `<<ui_text>>`, and a page `<<ui_page>>`.

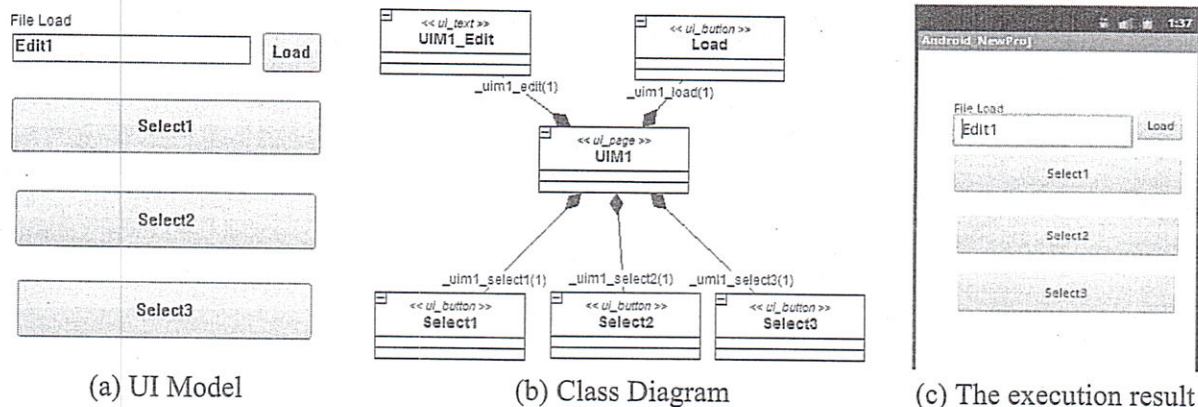


Figure 2. The result of transformation to class diagram from the UI Model

On mapping the resulted class diagram of the stereo type with event handler profile, the code can be generated like figure 3. UI Model information is transformed into a layout's xml file, which generates to form the configuration in Android screen. Then we execute the generated UI code from the class diagram. Finally we got the result as shown figure 2(c).

```
EditText _edit = (EditText) findViewById(R.id.editText1);
_edit.setOnFocusChangeListener(new OnFocusChangeListener() {
    @Override
    public void onFocusChange(View v, boolean hasFocus) { });
Button _load = (Button) findViewById(R.id.btn_load);
_load.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) { });
Button _select1 = (Button) findViewById(R.id.btn_select1);
_select1.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) { });
Button _select2 = (Button) findViewById(R.id.btn_select2);
_select2.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) { });
Button _select3 = (Button) findViewById(R.id.btn_select3);
_select3.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) { });
```

Figure 3. The result of code generation for Android Platform



## 5. Conclusions

The automatic code generation gives to shorten development time because it reduces unnecessary coding time of his/her developer, which can improve the quality of the software due to re-using the verified codes.

The existing code generation just generates the skeleton code of UML class diagram due to including just structural information, but needs additional written code. To solve this problem in this paper, we proposed the method of automatic UI code generation in Android platform. The proposed method consists of two steps: 1) to transfer class diagram from abstracted UI model and 2) to generate UI code from class diagram. This method increase code generate rate with class diagram, UI model, and event handler profile. Further research should be conducted, which is not dealt in this study on model and UI transformation for heterogeneous platform.

## Acknowledgments

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 (NRF-2013R1A1A2011601).

## References

- [1] Smartphone & chip market opportunities. Forward Concepts, 2008.  
[http://www.3g.co.uk/PR/Feb2009/Extensive\\_Smartphone\\_and\\_Chip\\_Market\\_Study\\_3G.htm](http://www.3g.co.uk/PR/Feb2009/Extensive_Smartphone_and_Chip_Market_Study_3G.htm)
- [2] B. Selic, "The pragmatics of model-driven development," *Software, IEEE*, Vol. 20, Issue 5, pp. 19-25, 2003.
- [3] W.Y. Kim, H.S. Son, J.S. Kim, R.Y.C. 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] W.Y. Kim, H.S. Son, J.B. Yoo, Y.B. Park, R.Y.C. Kim, "A Study on Target Model Generation for Smartphone Applications using Model Transformation Technique," *International Conference on Internet (ICONI) 2010*, Vol. 2, pp. 557-558, 2010.
- [5] W.Y. Kim, H.S. Son, J.S. Kim, R.Y.C. Kim, "Adapting Model Transformation Approach for Android Smartphone Application," *Advanced Communication and Networking*, Springer CCIS 199, pp. 421-429, 2011.
- [6] W.Y. Kim, H.S. Son, R.Y.C. 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, 2011.
- [7] W.Y. Kim, H.S. Son, R.Y.C. Kim, "Rule Extraction Method for Model Transformations in Heterogeneous Smartphone Applications," *Information Journal*, Vol.16, No.1(B), pp. 615-626, 2013.
- [8] Wikipedia, ATL, [http://en.wikipedia.org/wiki/ATLAS\\_Transformation\\_Language](http://en.wikipedia.org/wiki/ATLAS_Transformation_Language)
- [9] Obeo, Acceleo User Guide, <http://www.acceleo.org/>
- [10] W.Y. Kim, H.S. Son, R.Y.C. Kim, "Design of Code Template for Automatic Code Generation of Heterogeneous Smartphone Application," *Advanced Communication and Networking*, Springer CCIS 199, pp. 292-297, 2011.
- [11] H.S. Son, W.Y. Kim, R.Y.C. 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, 2013.
- [12] H.S. Son, J.S. Kim, R.Y.C. 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, 2013.
- [13] K. Czarnecki, S. Helsen, "Feature-Based Survey of Model Transformation Approaches," *IBM Systems Journal*, Vol. 45, No. 3, pp. 621-64, 2006.
- [14] Jamda, The Java Model Driven Architecture 0.2, <http://sourceforge.net/projects/jamda/>
- [15] H.S. Son, W.Y. Kim, R.Y.C. Kim, "A Design of Metamodel for Model Transformation of Heterogeneous Smartphone UI," *Proceedings of the 16th Korea Conference on Software Engineering (KCSE 2014)*, Vol. 16, No. 1, pp. 173-180, 2014.



## **Advanced and Applied Convergence Letters**

The AACL series is committed to the publication of proceedings of Advanced and Applied Convergence. Its objective is to publish original researches in various areas of Smart Convergence. This will provide good chances for academia and industry professionals as well as practitioners to share their ideas, problems and solutions relating to the multifaceted aspects.

Research papers were strictly peer-reviewed by program committees to make sure that the papers accepted were high quality and relevant to the current and future issues and trends in Advanced and Applied Smart Convergence.

The scope of AACL includes the entire area of advanced and applied convergence from the current and future trends. The language of publication is English.

