

Adrian Stoica
Jeong Jin Kang
Sabah Mohammed
Yvette E. Gelogo
Ronnie D. Caytiles (Eds.)

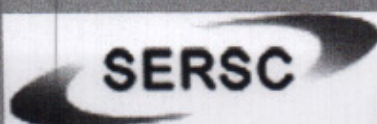
Advanced Science and Technology Letters

ASTL 02

Information Science and Technology

Proceedings
International Conference, IST 2012
Shanghai China, April 2012

Part 1



**Science and Engineering
Research Support soCietv**

Quantitative Analysis of Meteorological Disasters and Its Impact on National Economy	267
Chen Yunfeng, Li Lianshui, Cui Mengmeng, and Wang Jin	
An Adaptive Load Balancing Management for Distributed Virtual Environment Systems	271
Yuanxing Yao and Tae-Hyung Kim	
Secure high-precision face recognition based on immune computation.....	274
Tao Gong	
A Study on System Development Process based on Embedded System	277
Shin-Hyeong Choi and Hyoung-Keun Park	
Robust IPSec Key Recovery Solution for IKEv2 under Mobile and Wireless Environment.....	280
Yunjung Lee, Jungwon Cho, Keun-Wang Lee	
A Coverage Area Maximizing Algorithm for Deadline Situations	283
Heung Seok Jeon	
Concretization of UML Models based on Model Transformation for Windows Phone Application	288
Hyun Seung Son, Woo Yeol Kim, Robert Youngchul Kim	
Analysis of Temperature Track Monitoring Processes	292
Youn-Boo Kook, Jung-Sik Kim, Thom J. Hodgson	
A Hybrid Verification System using Speech and Video.....	296
Taewoong Choi, Soonhyob Kim, and Soonhyob Kim	
Provisioning Software Execution Platform on Cloud for Social Media Cloud Applications	299
In-Yong Jung and Chang-Sung Jeong	
Shear Skew Warp Volume Sketch on Cloud Collaboration	302
Kiyoungh Choi, Sungup Jo, Ki-Hyun Kim, and Chang-Sung Jeong	

Concretization of UML Models based on Model Transformation for Windows Phone Application

Hyun Seung Son, Woo Yeol Kim, Jae Seung Kim, and Robert YoungChul Kim

Dept. of CIC(Computer and Information Communication), Hongik University,
Sejong Campus, 339-701, Korea
{son, john, bob}@selab.hongik.ac.kr

Abstract. In this paper, we propose model transformation based on a message sequence diagram with a class diagram, which makes it possible to give a structural & behavioral expression to generate the detailed codes. This approach can perform more specific design than the previous model transformation based on only static structure [2,3,4]. We show a case study to illustrate our model transformation for the Windows Phone application.

Keywords: Model Transformation, Windows Phone, UML(Unified Modeling Language), MDD(Model Driven Development)

1 Introduction

The e-MDD (Embedded Model Driven Development) is a method for developing these heterogeneous platforms, adapted with the existing MDD [1]. The existing model transformation based on e-MDD expresses the model with only the class diagram, which can give effective expression to the static structure of software [2,4]. And it can provide various characteristics in terms of heterogeneous platforms [5,6]. However, the previous method with the class diagram is to express the static structure of software without the related detail information, so to result only in skeleton codes. This paper extends the behavior model to solve some problems of the existing structure-based model transformation. That is, it uses the message sequence diagram to express the dynamic behavior of software. We apply both class diagram and message sequence diagram in the process of model transformation to develop heterogeneous software. As a result, it finds out that our proposed model transformation expresses more characteristics than the previous model transformation [2,3,4].

This paper is organized as follows. Chapter 2 explains the method of model transformation for Windows Phone. Chapter 3 addresses the case study. Chapter 4 mentions the conclusion.

2 Model Transformation based on UML Models

The model transformation method based on UML models for windows phones is involved with a static structure of class diagram, and a dynamic interactive structure of the message sequence diagram. Our proposed model transformation focuses on these two diagrams on TIM (target independent model), which should transform into two converted models on TSM (target specific model). It generates each code with the converted diagrams on TSM for each particular platform. In this paper, the proposed method also uses the class diagram for changing the static structure of a system, and expresses & transforms the parts of the behavioral sequences in the message sequence diagram. On model transformation, it transforms both the class diagram and message sequence diagram simultaneously. To perform model transformation, it is to define the stereotype used on each diagram. And it shows the model transformation rule of the class diagram and message sequence diagram.

3 Case Study

Fig. 1 shows model transformation based on the class diagram. Three classes in fig.1 (a) are necessary to load a picture, and then move it on the screen. View is a class that shows the screen; Timer is a class that calls in a method periodically; Character is a class that includes a picture. What to transform should be expressed in a stereotype.

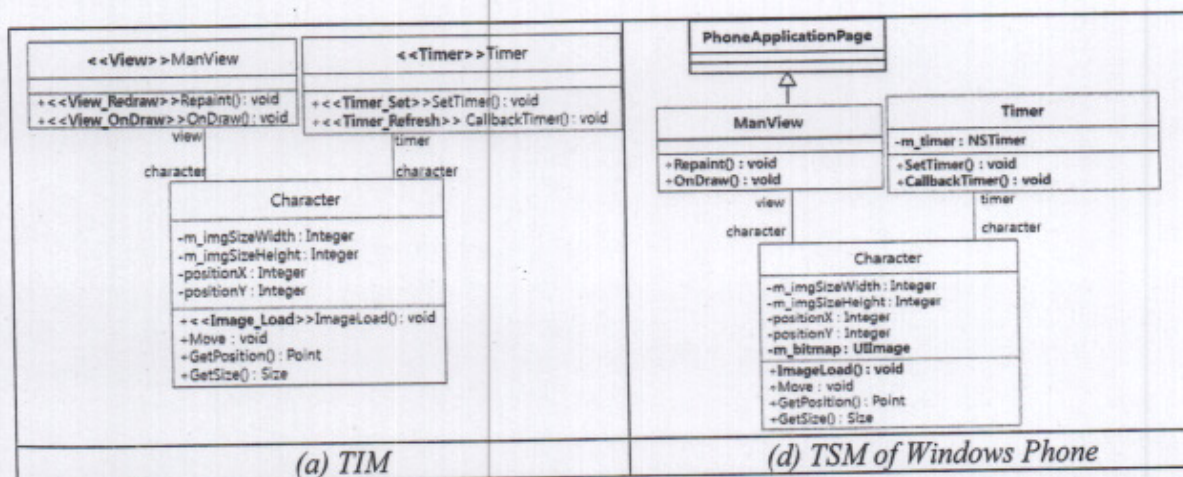


Fig. 1. Model Transformation of Class Diagram

Fig. 2 shows model transformation used message sequence diagram. This diagram indicates when an object should be created, and expresses the order in which a method is called.

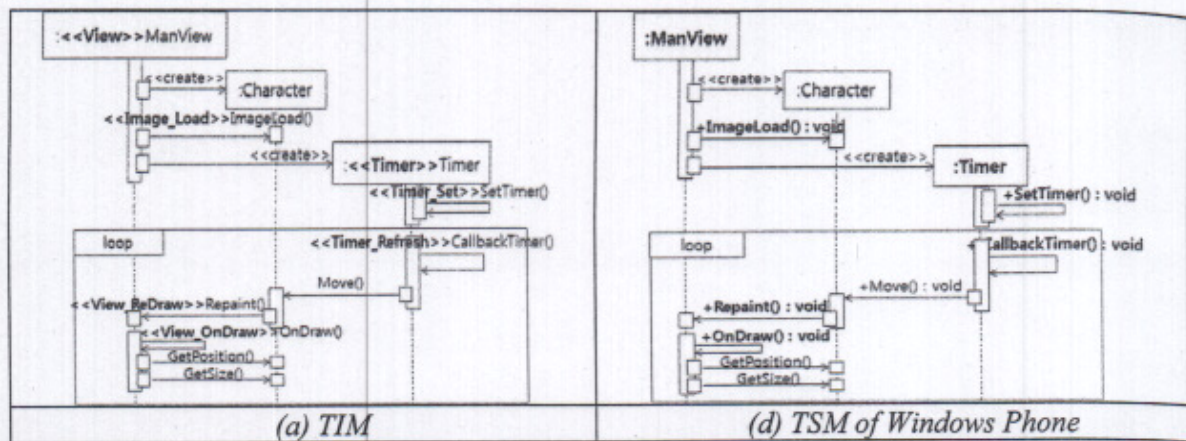


Fig. 2. Model Transformation of Message Sequence Diagram

The model transformation is performed depending on Windows Phone platform based on both class diagram and message sequence diagram. The class diagram can reflect the characteristics of class, method and attribute; the message sequence diagram can reflect the order of methods called. This result shows that when it is necessary to have the sequence of execution in order to perform a particular method, it is more effective to apply the message sequence diagram.

4 Conclusion

In this paper, we show concretization of the class diagram and message sequence diagram based on model transformation. And it shows the result for the Windows Phone platform. The result indicates that the class diagram can reflect the characteristics of class, method and attribute, while the message sequence diagram can reflect only the order of methods. This result also shows that when the order of execution is performing a particular method, it is more effective to apply the message sequence diagram. In addition, because it is important on the message sequence diagram to express the order of methods, the creation of the objects, and the time of performance, it can express the more specific characteristics of the platforms.

Acknowledgments. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0004203) and the Ministry of Education, Science Technology (MEST) and National Research Foundation of Korea (NRF) through the Human Resource Training Project for Regional Innovation.

References

1. Selic, B. The pragmatics of model-driven development. *Software, IEEE*, Vol. 20, Issue 5, pp. 19–25 (2003)
2. Woo Yeol Kim, Hyun Seung Son, R. YoungChul Kim: A Study of UML Model convergence Using Model Transformation Technique for Heterogeneous SmartPhone Application. *Software Engineering, Business Continuity, and Education, CCIS 257*, pp. 292–297 (2011)

3. Woo Yeol Kim, Hyun Seung Son, R. YoungChul Kim: Design of Code Template for Automatic Code Generation of Heterogeneous Smartphone Application. Advanced Communication and Networking, CCIS 199, pp. 292-297 (2011)
4. Woo Yeol Kim, Hyun Seung Son, Jae Seung Kim, R. YoungChul Kim: Adapting Model Transformation Approach for Android Smartphone Application. Advanced Communication and Networking, CCIS 199, pp. 421--429 (2011)
5. K. Czarnecki, S. Helsen: Feature-Based Survey of Model Transformation Approaches. IBM Systems Journal, Vol. 45 No. 3, pp. 621--64 (2006)
6. Sendall, S., Kozaczynski, W. Model transformation: The heart and soul of model-driven software development. IEEE Software, Vol. 20, No. 5, pp. 42--45 (2003)