Jeong Jin Kang
Edward J. Rothwell
Yang Hao (Eds.)

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

IIBC

IPACT

Implementation of Stochastic Based State Diagram for EmbeddedSoftware System / 160
So Young Moon, Byungkook Jeon, Jae Hyub Lee, R. Young Chul Kim

Location Estimation based Personalizationi using Support Vector Machine and Received Signal Strength of Mobile Phone within a Building / 163
Sungjin Cho, Sungkuk Cho, Byungkook Jeon

Metamodeling for automatically Extracting Test Case based on Use Case Diagram / 166
Bokyung Park, Byungkook Jeon, Jae Hyub Lee, R. Young Chul Kim

A Linearization of The Time-Varying Nonlinear System with Disturbance Using The Robust Feedback Linearization Method / 169
Jong-Yong Lee, ChangHee Lee, Do-Hyeoung Cho, Kye-dong Jung, Michat Strzelecki

Wavelet Transform based Image Registration using MCDT Method for Multi-Image / 170
Choel Lee, Jungsuk Lee, Kyedong Jung, Jong-Yong Lee, Junaid Ahsenali Chaudh

Development of Multi Sensor Array Electrode and Programmable Multi-channel Electrical Stimulator / 171
Suhong Kim, Sunguk Yun, Gunchel Park, Wooyoung Jang, Susung Kim, Jaehyung Kim, Gyerok Jeon

The Method for Providing Customized Advertisements related to Tailored Interactive Contents / 173
Hyun-Jin Lee, SeungJoon Kwon, KeeSeong Cho

An Equivalent Circuit Model for GaN-based MIM Capacitor / 177
Sang-Heung Lee, Seong-Il Kim, Ho-Kyun Ahn, Byoung-Gue Min, and Jong-Won Lim

A Study on Individual and Environmental Factors affecting Mobile Payment Service Acceptance / 181
Jae-Beom Lee, Sin-Bok Lee, Chanuk Park

Coexistence Study between Wi-Fi and ESL system / 184
Hye-Ju Yun, Seung-Nam Kim, Il-Kyoo Lee

Architecture for Fast data processing in big data platform / 188
Wooyung Lee, Youngdae Lee, Jeong-Jin Kang

Design of Dynamic Power Management Mechanism based on Device Characteristics / 189
Myungsub Lee

A Study on Inter-Noise According to the Materials / 193
Kyoung-Su Yeo, Myoung-Jin Bae

Sweet Spot Characteristics Analysis of Desktop Speakers at Various Positions and Angles / 196
Seoun-Young Park, Myung-Jin Bae

Metamodeling for automatically Extracting Test Case based on Use Case Diagram

Bokyung Park*, Byungkook Jeon**, Jae Hyub Lee***, R. Young Chul Kim*

* SE Lab., *Dept. of Computer and Information Communication, Hongik University, Sejong Campus, 339-701, Korea*
e-mail : {bkpark, bob}@selab.hongik.ac.kr*
***Dept. of IT Eng., Gangneung-Wonju Nat'l University, Korea*
jeonbk@gwnu.ac.kr
***Dept. Of Computer Division, Koreatech, Korea
jae@koreatech.ac.kr***

Abstract

The previous research suggested test case generation from Use Case [1], which makes use case diagram and specification, identifies use case scenario based on event flows, generates scenario matrices, and then extract test case. But it is not considered of automatic mechanism for test case generation. To automatically generate test case, we suggest to adapt metamodel approach, which easily transform each model, and possibly develop automatic tool. In this paper, we suggest to design metamodel and define each model needed to extract test case based on use case diagram.

Keywords: *Use Case, Test Case, Meta-model, Cause-Effect Diagram*

1. Introduction

Currently as it gradually increases the complexity and scale of software, it is necessary for testing software to find error or defects [2]. Testing is sure to find error, and check whether requirements are reflected or not. It needs to make an effort at each development stage due on increasing cost to fix software developed with wrongly analyzing requirements. The rational software [1] also suggested to generate test cases from use cases, but not considered of an automatic tool. Kim & Son [3] suggested automatic test case generation based on state diagram with metamodeling and model transformation. In these tools, it is difficult to change if it does fix input model. To solve a problem, this paper mentions to adapt use case approach to extract test case. Use Case works at early stage of development lifecycle, and is defined requirements of software.

This paper is organized as follows. Chapter 2 explains related studies. Chapter 3 mentions to design metamodel for automatic test case generation. Chapter 4 mentions the conclusion and future work.

2. Generating Test Cases from Use Cases

The rational software [1] used use case approach to generate test case, which represent use case diagram based on UML. Use Case Specification is included Use Case Name, Flow of Events (Basic Flow, Alternate Flow), Preconditions, and Postconditions. Flow of events is important to extract test case from use case. That is, it makes use case scenario based on flow of event, then extracts test case based on this scenarios. Kim [2] also suggested 'Metamodel oriented Automatic Test Case Generation Based on Transforming UML 2.4.1 Message-Sequence Diagram via Cause-Effect Diagram' as follows:

J.J. Kang et al. (Eds.): ISAAC 2014 & ICACT 2014, AACL 03, pp. 166~168, 2014
© The Institute of Internet, Broadcasting and Communication 2014

Step 1: For each use case, generate a full set of use case scenario and scenario matrix.
Step 2: For each scenario, identify at least one test case and the conditions that will make it execute.
Step 3: For each test case, identify the data values with which to test.

3. Design of Metamodel for Automatically Extracting Test Cases Based on Use Case

In this paper, we adapt metamodel mechanism to generate test cases from use case diagram, which is similar with the previous use case approach [1] and also applied other diagrams in UML [2, 3]. We mention how to design metamodel for generating test cases.

3.1 Generation of Test Case

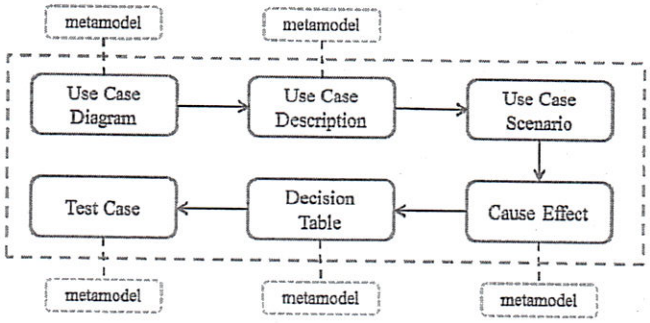

Figure 1. Test Case Generation based on Use Cases adopted with Metamodel approach.

Figure 1 shows the procedure of test case generation based on use case approach. Our suggested test case generation consists of six steps.

3.2 Metamodel Design

Metamodel of use case diagram is considered of attributes (such as Actor, Use Case, Include, and Extends relationship) in Use case. An actor has associated with one more use case, which has a relationship between Actor and Use Case, and also has Generalization, Inclusion, and Extension relationship between Use cases. We design metamodel of use case diagram included attributes of any relationships.

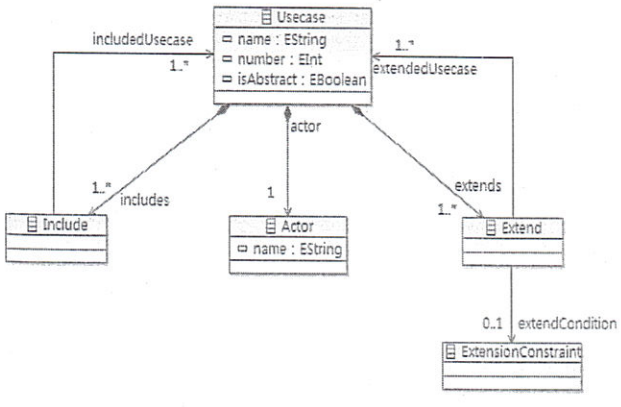

Figure 2. Metamodel of Use Case Diagram.

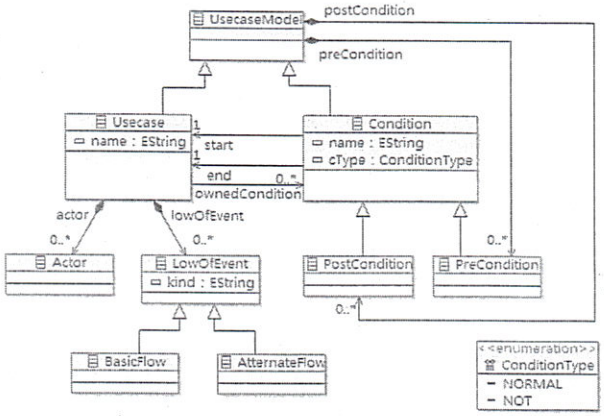

Figure 3. Metamodel of Use Case Description

Figure 2 shows the metamodel of Use case diagram. In Use Case Description, Use Case Model consists of Use case and Condition. UseCase consists of Actor and FlowOfEvent which separates BasicFlow and

AlternateFlow. Precondition is linked at the start point. Postcondition is linked at the end point. Metamodel of Use Case is difficult to use and complex, but exists the similar research [4].

Figure 3 shows metamodel of use case description. Cause-Effect Model is included all elements at the most top node. This model has Cause, Effect, and Connector. Cause is linked with the start attribute and effect also linked with the end attribute at a Connector as a line linked with Cause and Effect.

4. Conclusion

In this paper, we adapt Cause-Effect Diagram to generate test case based on use case diagram and scenarios, and also design each metamodel within each step. The reason is satisfied with 100% functional requirements with minimal test cases Cause-Effect Diagram. It is necessary to use metamodel and model transformation for automatic test case generation, but not mention them in this paper.

Acknowledgement

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] The Rational Software 2001, Generating Test Cases From Use Cases, http://www.therationaledge.com
[2] S.J. Woo, *Metamodel oriented Automatic Test Case Generation Based on Transforming UML 2.4.1 Message-Sequence Diagram via Cause-Effect Diagram.* M.D. Dissertation, Hongik University, 2012.
[3] J.S. Kim, *A Study on Automatic Test Case Generation Based on UML State Diagram with Meta-modeling and Model Transformation.* M.D. Dissertation, Hongik University, 2013.
[4] S.S. Stephane, "A Meta-Model for Textual Use Case Description," *Journal of Object Technology*, vol. 8, no. 7, 2009.

ISSN 2288-6060

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.