

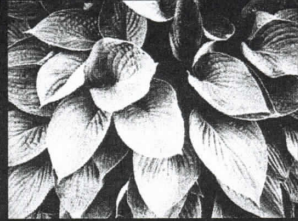


Printed in Japan

ISSN 1343-4500 (print)
ISSN 1344-8994 (electronic)

iNFORMATION

An International Interdisciplinary Journal



Volume 18 Number 6(A), June 2015

Published by International Information Institute
www.information-iii.org

in the GPS Signal Attenuation		
	<i>Seongjin Cho, Sungkuk Cho and Byungkook Jeon</i>	2425
Metamodeling for Automatic Test Case Generation		
	<i>Bokyoung Park, Byungkook Jeon, R. Young Chul Kim and Jae Hyub Lee</i>	2431
Advanced Efficiency of DC-LED Lighting Systems to Replace the Conventional AC-LED Lighting Systems		
	<i>Yeon Seok Lee, Sang-Ui Choi and Sherif A. S. Mohamed</i>	2437
Development of Reader for Non-metallic Permeability Enhancement on 134KHz Bandwidth		
	<i>Min Soo Kang, Yong Gyu Jung and Jae Ho Lee</i>	2443
Relative Location Measurement in Container-Grown Seedlings Based on Wireless Sensor Networks	<i>Sang-Hyun Lee and Kyung-II Moon</i>	2449
Neural Networks based Adaptive Tracking for Nonlinear Systems		
	<i>Raheel Quraishi, Nisar Ahmed and Muhammad Shafiq</i>	2457
Analysis of Competitiveness in Secondary Battery Using Bibliometric Analysis		
	<i>Young-il Kwon, Dae-hyun Jeong, Hyun-kyoo Choi and Jae Yun Lee</i>	2477
Calculation of Synthetic Electric Field and Ion Flow Density along UHVDC Transmission Lines Based on the Non-Grid Method		
	<i>Qinghui Zhang and Cui Yong</i>	2489
Design and Evaluation of Self-organizing Scheduling Mechanism Adaptive To Traffic Load for Wireless Networks		
	<i>Yoshiaki Taniguchi and Kyohei Inoue</i>	2503
Virus detection backdoor in Microsoft Security Essentials		
	<i>Silnov Dmitry Sergeevich and Tarakanov Oleg Vladimirovich</i>	2513
Cloud Storage Integration as a Learning Object Repository for Massive Open Online Course	<i>Ridi Ferdiana</i>	2521
Efficient Robust and Secure E-DRM with Encrypted Content Search		
	<i>Maged Hamada Ibrahim</i>	2531
Performance Comparison of Two Open-source CloudStack Client Libraries for High-performance Computing	<i>Guohua Li, Hyeyoung Cho,</i> <i>Seung Chan Shin, Jaekeun Yeom, Jaegyoon Hahm and Sang Boem Lim</i>	2547
Performance Analysis and Improvement of Meteorological Impact Assessment System		
	<i>Tongji Jin, Ying-Hua Bian, Rongxu Xu and Sang Boem Lim</i>	2561
The Evaluation of Vehicle Interior Acoustics Through Statistical Neural		

Metamodeling for Automatic Test Case Generation

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

**SE Lab., Department of Computer and Information Communication, Hongik University,
Sejong Campus, 339-701, Korea*

*E-mail: {park, bob}@selab.hongik.ac.kr**

*** Dept. of IT Eng., Gangneung-Wonju Nat'l University, Korea
jeonbk@gwun.ac.kr***

****Dept. of Computer Science and Engineering, Koreatech, Korea
jae@koreatech.ac.kr****

Abstract

The previous research suggested test case generation from Use Case [1], which made use case diagram and specification, identified use case scenario based on event flows, generated scenario matrices, and then pretend to extract test case. But it did not consider automatic mechanism for test case generation. We adapt metamodel approach to generate automatic test case. This approach easily transforms each model, and possibly develop automatic transformation. In this paper, we suggest to design metamodels for each model (that is, Use Case, Cause-Effect, Decision Table, and Test Case) needed to extract test case based on use case diagram.

Key Words: Use Case Approach, Meta-model, Cause-Effect Diagram, Decision Table, Test Case

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 the conclusion and future work.

2. Related Works

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:

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. Metamodel for Extracting Automatic 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 Test Case Generation

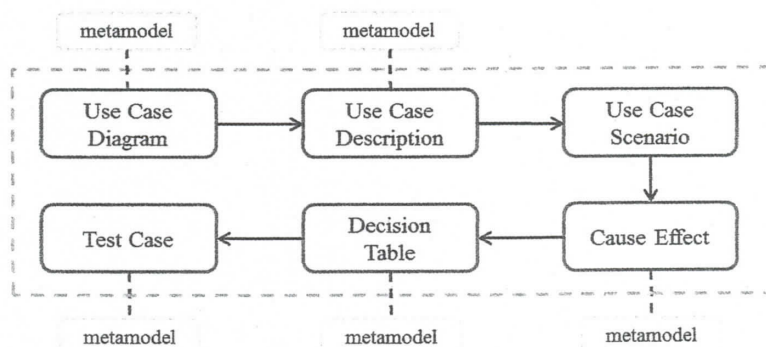


Fig. 1. Test Case Generation based on Use Cases adopted with Metamodel approach [4].

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

Step 1(Use Case Diagram): Identify actor and use cases and draw use case diagram based on relationship with use cases such as extends and includes relationships.

Step 2(Use Case Description): Describe the specification of use cases, and identify use case name, flow of events, preconditions, and postconditions as attributes of each Use Case based on the use case format[1].

Step 3(Use Case Scenario): To generate test cases from use case, it is important to consider 'flow of events' which consist of the basin main flow and alternative flow. A use case scenario is a kind of an instance of use case or a complete path [1]. Therefore it identifies 'flow of events' of each scenario, and then make the combination of them.

Step 4(Cause-Effect Diagram): Analyze the relationship between cause/effect/condition and the combination of events. That is, use Cause-Effect Diagram which is satisfied with 100% functional requirements with minimal test cases [2,3]. This diagram represents logical relationship through True/False of condition and Boolean expression between cause and effect.

Step 5(Decision Table): This table represents the logical relationship expression between cause and effect. Each column of decision table means a test case. A test case can be combined with input values (such as true, false, and constraints) of input.

Step 6(Test Case): Be generated with decision table via Cause-Effect Diagram. Test case consists of Input, Precondition, Test Condition, and Expected Result. Test Condition represents logical expressions with the combination of input's conditions.

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

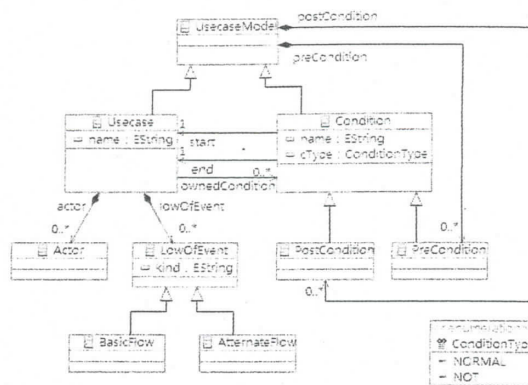
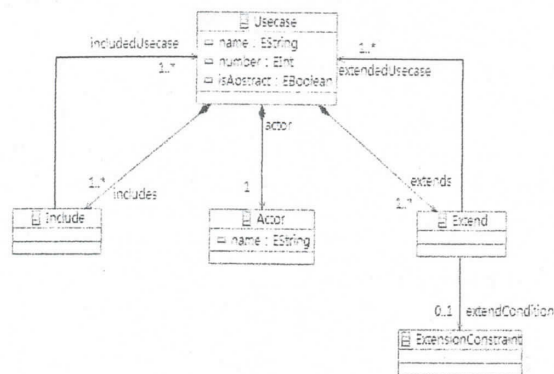


Fig. 2. Metamodel of Use Case Diagram [4] Fig. 3. Metamodel of Use Case Description [4]

Figure 2 shows the metamodel of Use case diagram. In Use Case Description, Use Case Model consists of Use case and Condition. Use Case 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 [5]. 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.

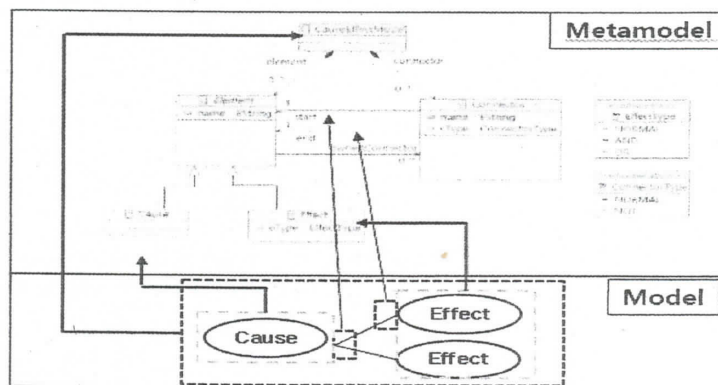


Fig. 4. Metamodel of Cause-Effect Diagram [2]

Figure 4 shows Metamodel of Cause-Effect Diagram. There are not defined the standard metamodel of Cause-Effect Diagram [3]. Therefore, we design metamodel with the name of this diagram.

The decision table represents as a table with the values of Cause-Effects Diagram. The condition of Effect is the result with the logical expression of Cause-Effects Diagram. The Cause consists of several inputs, and the Effect also consists of several outputs. The Value of

test case shows T(True)/F(False).

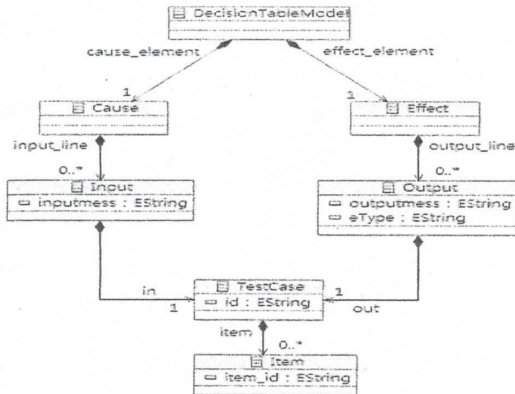


Fig. 5. Metamodel of Decision Table[2,3]

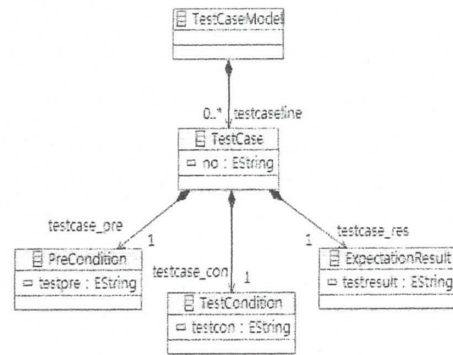


Fig. 6. Metamodel of Test Case[2,3]

Figure 5 shows Metamodel of Decision Table [2]. There are not defined the standard metamodel of Decision table [3]. Therefore, we design metamodel of the name of this table. The metamodel of Test Case ‘TestCaseModel’ may have several test cases. PreCondition means the condition of input, and TestCondition represents the logical expression of the combination of input conditions. ExpectationResult represents the value of expected results. Figure 6 shows Metamodel of Test Case. A Test Case includes TestCaseModel included with Precondition, TestCondition, and ExpectationResult.

4. Conclusion

In the previous automatic mechanism, it was used an algorithm approach. 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. With Cause-Effect Mechanism, it is satisfied with 100% functional requirements with minimal test cases. It is necessary to use metamodel and model transformation for automatic test case generation, but not mention them in this paper.

5. 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), and also funded for Education & Research

Promotion of Koreatech.

References

- [1] The Rational Software, Generating Test Cases from Use Cases, <http://www.therationaledge.com>, 2001.
- [2] S. J. Woo, Metamodel oriented Automatic Test Case Generation Based on Transforming UML 2.3.1 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] Bo Kyung Park, Metamodeling for automatically Extracting Test Case based on Use Case Diagram, Advanced and Applied Convergence & Advanced Culture Technology, AACL 03, pp. 166-168, 2014.
- [5] Hyun Seung Son, Test Case Generation from Cause-Effect Graph based on Model Transformation, ICISA 2014, pp. 591-593, 2014.

***Corresponding author: Jae Hyub Lee, Ph.D., Professor
Department of Computer Science and Engineering,
Koreatech, Korea
E-mail: jae@koreatech.ac.kr***

Mathematical and Natural Sciences

- On New Subclasses of Meromorphic Starlike Univalent Functions With
Positive Coefficients *Mostafa Albehhah and Maslina Darus* 2201
- Enhanced Method for Avoiding Worst Case Complexity in Quicksort
Algorithm *Mohammed Awad and Malek Imam* 2209
- Hartree-Fock and time-dependent Hartree-Fock calculations with the Gogny
interaction using a Lagrange mesh *Yukio Hashimoto* 2219

Management and Social Sciences

- A Study on Relation between Smartphone Addiction Factors and Gender:
Focused on Adolescent *Chloe Jihee Shin, Clara Sumin Yoon and Hun Choi* 2233
- The Effect of Ethical Leadership on Organizational Performances
Yun Ji Moon 2241
- A Study on Model Transformation Framework for Heterogeneous Smart
Learning Contents *Woo Yeol Kim* 2247
- The Development of WorldSkills Competition Mobile Robotics Course for
improving the STEM convergence capability of technical high school student
Bae Sang Yong, Ryue Dae Hyun and Shin Seung Jung 2253
- A Review and Classification on the Supply Chain Risk Management
Tang Liu and Hiroaki Matsukawa 2259
- Analysis of Student Behavior in the Beer Game
H. W. Jin 2275
- Organisations' Perceptions towards the Effectiveness of Integrated
Management System in Malaysia *Kadir Arifin and Muhammad Rizal Razman* 2283
- A New Trend in the Digital Economy: A New Era of Digital Convergence
Business *Gyanendra Prasad Joshi, Changsu Kim and Sung Won Kim* 2293
- Concept, Method and Business Intelligence Model
Andre M. R. Wajong 2305
- Listenability Measuring Method based on Learner Proficiency from Multiple
Perspective *Katsunori Kotani and Takehiko Yoshimi* 2315
- Cultivation Systems for human resources of Entrepreneur Implications for
Management and Marketing of the Education System *Jun Oheki* 2329

Agriculture and Engineering

- Estimating on the Optimal Productivity Indicators based on Simulation
Software *Sang-Kook Park and Jong-Bae Kim* 2351
- Attack Model on Smart Home Network under Internet on Things
Surroundin *Jung Tae Kim* 2357
- The Classification of Heart Sound during Systole and Diastole by Neural
Network *Sang-hee, Eum and Jae-hyun, Nam* 2363
- Analysis of Tunneling Current for Channel Dimension of Sub-10 nm
Asymmetric Double Gate MOSFET *Hakkee Jung and Ohshin Kwon* 2369
- Analysis of Drain Induced Barrier Lowering for Channel Length and
Thickness of Asymmetric Double Gate MOSFET *Hakkee Jung* 2375
- Vision based Crack Identification and Analysis on the Surface of Concrete
Slab Structures *Kwang Baek Kim and Doo Heon Song* 2381
- Intelligent 8-phases Traffic Signal Control Model based on Fuzzy Logic
for High Level Traffic Congestion Situation *Kwang Baek Kim and Doo Heon Song* 2387
- Fair Allocation of Sensor Measurements Using Shapley Value
Sang-Seon Byun 2393
- Development of an Around-View Monitoring System Using USB Rear
View Cameras and Image Processing *Gyu-Hyun Kim and Jong-Wook Jang* 2401
- Development of PE/PET Film Cutting System using the Displacement
Sensor and the 3-axis DC Servo Motor Control System
Gwan-Hyung Kim, Oh-Hyun Kwon and Am-Suk Oh 2407
- Design and Implementation of a Finger Tracking Algorithm for HCI
(Human-Computer Interaction) *Byungkook Jeon and Sungkuk Cho* 2413
- Stochastic Design Tool based on Extended State Diagram for Reactive
Software System *So Young Moon, ByungKook Jeon, R. Young Chul Kim and Jae Hyub Lee* 2419
- Location Estimation for Personalization using Received Signal Strength
in the GPS Signal Attenuation *Seongjin Cho, Sungkuk Cho and Byungkook Jeon* 2425
- Metamodeling for Automatic Test Case Generation
Bokyung Park, ByungKook Jeon, R. Young Chul Kim and Jae Hyub Lee 2431

(Continued)