Abstract

Code coverage analysis holds a very important role in software testing procedure. By the test case runs it provides developers by means to quantify of how well their source code is being exercised. By detecting errors/bugs in the code it estimates the efficiency of the test. We must implement a systematic way and support the theoretical bases for testing the programs with the purpose of performing effective software testing and coverage analysis. In our research we use the crossword application where we automatically make test cases and systematically discover the impact of context, as captured by criterion functions which we described in our source code. Our studying demonstrates that by increasing the event combinations tested and by organizing the comparative positions of events defined by the new criteria, we can become aware of a large number of faults that were undetectable by earlier techniques. In this paper we are implementing the event based test case generation by the use of object oriented and aspect oriented event based test case generation. Before this we have used the GUI based test case generation as an existing work. The experimental result shows that our proposed work test case generation process providing better code coverage range when compared with the existing work.
Generating Object-Oriented and Aspect-Oriented Sequence based Test Cases with Optimum Code Coverage


R. Ferguson, B. Korel 1996 &quot;The chaining approach for software test data generation&quot; ACM Trans on Software Engineering and Methodology, pp. 63-86.


Zhao, J. 2003 &quot;Data-Flow-Based Unit Testing of Aspect-Oriented&quot; Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC&lt;apos;&lt;br&gt;&lt;apos;); In proceedings of IEEE.

Rajan, H. , Sullivan, K. 2005 &quot;Generalizing AOP for Aspect-Oriented Testing&quot; in conference &apos;ACM 1-58113-000-0/00/0004.

Zhao, J. , Rinard, M. &quot;System Dependence Graph Construction for Aspect-Oriented Programs&quot; , Cambridge, USA.

Hailpern, B. , Santhanam, P. 2002 &quot;Software debugging, testing, and verification&quot; IBM SYSTEMS JOURNAL, VOL 41, NO 1.


Xu, G. , Yang, Z. , Huang H. , Chen, O. , Chen, L. , Xu, F. : JAOUT 2004 &quot;Automated Generation of Aspect-Oriented Unit Test&quot; Proceedings of the 11th Asia-Pacific Software Engineering Conference (APSEC&lt;apos;&lt;br&gt;&lt;apos;); Also in proceedings of IEEE.

Hughes, D. , Greenwood, P. &quot;Aspect Testing Framework&quot; Computing Department, Lancaster University, UK.

Augusto, O. , Lemos, L. , Maldonado, J. C. , Masiero, P. C. &quot;Data Flow Integration Testing Criteria for Aspect-Oriented Programs&quot; Universidade de S´ao Paulo, Av. do Trabalhador S´ao-Carlense, 400, S´ao Carlos, SP.

Alexander, R. T. , Bieman, J. M. 2004 &quot;Towards the Systematic Testing of Aspect-Oriented Programs&quot; Colorado State University, Department of Computer Science, Safety Systems Research Center, University of Bristol, Bristol, UK, Published by Elsevier Science B. V.

Index Terms
Keywords

Code Coverage  Aspect oriented testing  Object oriented testing  GUI testing
automated testing
test case generation
Testing Process