

Preface

In spite of the increasing complexity of software, clients expect improvements in its quality. In order to enable high-quality software in this context, the tools and approaches for testing have to be reconsidered. This Brief summarizes technical as well as organizational aspects of testing. For the latter, an empirical study of best practices for testing has been conducted. As a result, a set of recommendations has been formulated, which help to organize the testing process in a company while taking into account parameters such as its size. It turns out that companies test very differently. While some (typically larger) companies have installed well established testing procedures and use state of the art tools e.g. in order to automate testing as far as possible, other (often smaller) companies test on an ad-hoc basis without predefined processes and with little tool support. In particular, the latter can benefit from the best practices formulated in this book. This part is particularly interesting for practitioners.

From a technical perspective, this Brief presents a new way for generating so-called glass-box test cases automatically. The approach is based on a symbolic execution of Java bytecode by a symbolic Java Virtual Machine (JVM). The latter is an extension of the usual JVM by components known from abstract machines for logic programming languages such as the Warren Abstract Machine. In particular, it contains a trail, choice points, and logic variables and it uses a system of constraint solvers. The symbolic JVM systematically follows all relevant computation paths and generates a set of test cases ensuring a predefined coverage of the control and data flow of the tested software. This part of the Brief is particularly interesting for researchers working on testing.

A third part shows how the mentioned test-case generator can be profitably used in an E-assessment system which automatically corrects Java classes uploaded by students. The test cases are generated from an example solution and serve as a measure for evaluating the correctness of the uploaded solutions to programming exercises. This approach has been successfully applied in a practical programming course. Corresponding experimental results are given.

This Brief is an extract of the PhD thesis of the author. It gives a concise overview of the mentioned aspects with many references to the relevant literature. It is interesting for practitioners as well as researchers. For studying the most interesting aspects in depth, a look into the mentioned literature is recommended.

Münster, December 2011

Herbert Kuchen



<http://www.springer.com/978-3-642-27463-3>

Improving Software Testing
Technical and Organizational Developments
Majchrzak, T.A.
2012, XVIII, 160 p. 17 illus., Softcover
ISBN: 978-3-642-27463-3