Preface

Alexander Petrenko
ISPRAS
Moscow, Russia
petrenko@ispras.ru

Bernd-Holger Schlingloff
Humboldt Universität zu Berlin
Berlin, Germany
hs@informatik.hu-berlin.de

Nikolay Pakulin
ISPRAS
Moscow, Russia
npak@ispras.ru

This volume contains the proceedings of the 10th Workshop on Model-Based Testing (MBT 2015), held in London on April 18th, 2015, as a satellite workshop of the European Joint Conferences on Theory and Practice of Software (ETAPS 2015). The first Workshop on Model-Based Testing in this series took place in 2004, in Barcelona.

A tenth anniversary is a good opportunity to look back and to strike a balance, analyzing tendencies of investigations in MBT for the last 10+ years. This preface is not aimed at a thorough analysis; a flavor of the topics under consideration can be tasted by reviewing the invited papers made at the previous MBT workshops. The list of speakers and titles of their talks was as follows:

2004	Keith Stobie	Microsoft	Model Based Testing in Practice at Microsoft
2004	Rober V. Binder	mVerify	Test automation
2006	Harry Robinson	Microsoft	Model-based Testing for the Masses
2006	Alan Hartman	IBM Haifa Research Labs	Ten Years of Model Based Testing – A sober evaluation
2007	Rob Hierons	Brunel University	Finding a Good Order for Applying Adaptive Test Cases
2007	Antti Huima	Conformiq	Model-Based Testing in the Standardization of Information and Communication Technologies: the ETSI Perspective
2008	Wolfgang Grieskamp	Microsoft	Using Model-Based Testing for Quality Assurance of Protocol Documentation
2008	Marie-Claude Gaudel	Université de Paris-Sud, Orsay	Coverage-Based Random Exploration of Large Models
2009	Patrice Godefroid	Microsoft	Whitebox Fuzzing for Security Testing
2009	Darko Marinov	University of Illinois at Urbana-Champaign	Model-Based Testing Using Test Abstractions
2010	Jan Tretmans	Radboud University Nijmegen	Model Based System Development
2012	Ina Schieferdecker	Fraunhofer FOKUS	Model-Based Security Testing
2013	Jan Peleska	University of Bremen, Verified Systems International GmbH	Industrial-Strength Model-Based Testing – State of the Art and Current Challenges

Petrenko, Schlingloff, Pakulin (Eds.): Tenth Workshop on
Model-Based Testing (MBT-2015)
EPTCS ??, 2015, pp. 1–6, doi:10.4204/EPTCS.??.0

2014	Alexandre Petrenko	Computer Research Institute of Montreal	How Does Non-determinism Occur in Test Models and What Do We Do with It?

We would like to give a short retrospective on MBT beginning with a talk of Alan Harman, made in 2006, because of its notable title: *Ten Years of Model Based Testing – A sober evaluation* [6]. The title of the talk claims that model-based testing has started at about 1996. In fact, MBT appeared already in the 1970s, and, apparently, even earlier. The main MBT applications in those years were in the field of testing hardware logic and telecommunication protocols. The main modeling paradigms in this field were Finite-State machine, Petri nets and other transition systems. The late 1990s is the period when the community understood that this approach can be effectively used not only for a rather narrow class of tasks close to telecommunication protocols, but also for practically all types of software, including operating systems, compilers, DBMS and others. This new approach to MBT immediately set the task to seek more adequate modeling paradigms for a new class of target systems. As the result the developers began to investigate requirement specification in the form of assertions or constraints and their special cases for program contracts.

A formal specification of requirements can be treated as a model of the behavior of a system under test. The testing task can be considered as the task of generating tests which properly cover the requirements model, with a subsequent "translation" of these "model tests" to the platform of "implementation" or "target tests".

Apparently, this new trend of MBT applications was reflected in articles such as T. J. Ostrand and M. J. Balcer's *The Category-Partition Method for Specifying and Generating Functional Tests* [7]. This method later became known as CPM. The essence of the CPM method is to design tests on the basis of a certain state machine, the states of which correspond to the domains of input space, i.e., partitions.

Those input space domains, in turn, correspond to the data domains where predicates making up the assertions of functional specification of the system under test operations take true or false values – that is, the input space is divided into partitions according to the terms described in specifications of functional requirements to operations under test.

Ostrand and Balcer offered a certain method of projecting tests. However, nothing was said about the tools supporting those methods. Probably, one of the first works aimed at designing a tool kit to support the method was the KVEST technology (I. Burdonov, A. Kossatchev, A.K.Petrenko, D. Galter *KVEST: Automated Generation of Test Suites from Formal Specifications* [3]) and the following UniTESK technology (I. Bourdonov, A. Kossatchev, V. Kuliamin, and A.K. Petrenko. *UniTesK Test Suite Architecture* [2]). Both technologies were based on the experience of the development testing methodology within the frames of Space Shuttle "Buran" designed in the late 1980s (A.K.Petrenko. *Test specification based on trace description* [8]).

This stage of MBT development was advanced with the emergence of test generation based on abstract state machines (W. Grieskamp, Y. Gurevich, W. Schulte, and M. Veanes. *Conformance Testing with Abstract State Machines* [4]), followed by SpecExplorer (A. Blass, Y. Gurevich, L. Nachmanson, and M. Veanes. *Play to test* [1]) in 2005.

Actually, the idea of organizing an MBT workshop appeared after one of the authors of this Preface met Yury Gurevich in Novosibirsk at Ershov Conference PSI-2001. Both future organizers of the MBT workshop aimed at designing technologies for testing of system software (including operating systems). The group of A.K. Petrenko has been developing the testing approach on the basis of behavioral models in the form of program contracts. The group headed by Yury Gurevich used ASMs as the main modeling technique. After some time it became clear that the techniques of explicit (executable) and implicit (in the form of constraints) specifications are complementary to each other. As a result, the contemporary tools UniTESK and Microsoft Spec Explorer use both model types.

At the end of 2002, the idea appeared to organize MBT workshop as a satellite event of ETAPS-2004, and devote it to testing on the basis of formal models – the idea that was gaining more and more supporters that time. We decided not to restrict it to only a certain narrow class of models or test generation techniques. Thus, from the very first workshops there were supporters of many different variations of MBT.

The list of the invited talks, presented above, shows, on the one hand, the rather a wide range of work that has been carried out in the field of MBT and, on the other hand, suggests some reflections.

In particular, it is interesting to note that in the beginning of the list, practitioners prevailed, while towards the end of the list theoreticians dominate. Why so? We could imagine a number of potential reasons:

a. Firstly, when this new theme was developing, practitioners were invited to show convincing use cases of MBT in large-scale industrial projects.

b. Secondly, it seems that in recent years practical MBT tools reached their limit in utilizing existing techniques: by now the tools have implemented most of what can be achieved at the present technological level. However, the development does not end here; we face a period of accumulating new knowledge that will make new fundamental advances in practical MBT development possible. Therefore, in recent years interest has shifted towards more theoretical research questions.

And one more observation: we can see an obvious predominance of giants such as Microsoft, IBM, Nokia among practitioners (Conformiq is mainly oriented towards Nokia's demands). This can probably be explained by the fact that development of an industrial-strength MBT tool and MBT applications requires significant resources, which are hardly available for small and medium-sized companies.

Observing the list of invited speakers (or, more generally, the list of all the speakers at recent MBT workshops), there is no prevailing trend in the modeling paradigms or testing techniques employed. On the basis of this observation we propose that different approaches to MBT are, on the one hand, complementary, and, on the other hand, the approaches need both methodological integration and unification on the level of testing system components and unified interfaces.

This allows us to implement shared use of model analyzers and program source codes, data generators, provers, systems for collecting and run-time monitoring, etc.

Let us briefly reflect on the perspectives of MBT:

- Though no revolutionary ideas in MBT development have been proposed in recent years, there are many problems for which the development of methods and underlying theories is required.
- There are challenging technical and engineering problems in the development of tools for MBT; still there are no MBT-specific solutions, as construction of MBT tools is based on a wide range of software engineering technologies (including both traditional means and more recent ones like provers and solvers). A current trend in developing MBT tools is its integration with a variety of approaches – static and static-dynamic analysis of programs, as well as software model checking technologies.
- There are some fundamental challenges in MBT deployment:
  - The first difficulty (not the main one) is related to teaching new methods of testing for testers. This requires not only ”training”, but education of the tester, who also has to be an expert in

requirements analysis, their modeling and "translation" of high-level requirements presented to the level of implementation under test interfaces.

  - The second problem is the issue of how to obtain "good" models. It is both technical (in what form models should be designed) and organizational (how to build a joint process of implementation design and model design). The experience shows that skilled software designers don't want to duplicate their work and waste time on designing a model, besides the implementation itself; and unskilled programmers can do well neither in coding nor in behavior modeling.

- MBT deployment requires substantial resources, however there are no fundamental difficulties if the organization decides to introduce MBT in practice. A good example to this is the experience of Microsoft lab in China (see for instance W. Grieskamp, Xiao Qu, Xiangjun Wei, N. Kicillof, M. B. Cohen. *Interaction Coverage meets Path Coverage by SMT Constraint Solving* [5]).

- MBT is actively used in the fields having the experience of systematic certification of software systems, particularly in avionics and automotive. Since this domain is becoming more and more important, a challenge is to align MBT with certification processes and regulations.

This year's MBT workshop features Ana Cavalli from Institut National des Telecommunications, Paris, France as invited speaker. In her speech, entitled "Evolution of testing techniques: from active testing to monitoring techniques", she presents the evolution of these testing techniques, their advantages and limitations, and illustrates the application of monitoring techniques to the security testing of real case studies.

The contributions selected by the Program Committee reflect both applications of MBT in industrial practice and further development of MBT theory and techniques.

Ana Rosario Espada, Maria Del Mar Gallardo, Alberto Salmerón and Pedro Merino present an approach to automated model construction and test generation for Android mobile applications. Marcus Gerhold and Mariëlle Stoelinga extend the well-known notion of input-output conformance to probabilistic state machines, opening the door to development of new classes of models and test construction techniques. Paul Jorgensen presents a novel variation of Petri-nets to facilitate visual modeling of interacting components in complex systems. Hartmut Lackner and Martin Schmidt discuss quality of test suites for product lines and develop an assessment approach based on mutation operators applied to software product lines. Natalia Kushik and Nina Yevtushenko present new result in the theory of FSM; they show that for some FSMs its homing sequence can be built in polynomial time.

The workshop is concluded by a remotely presented talk by Yury Gurevich on "Testing Philosophy".

We would like to thank the program committee members and all reviewers for their work in evaluating the submissions for the whole period of MBT workshops. We also thank the ETAPS organizers for their assistance in preparing the workshops and the editors of EPTCS for help in publishing these workshop proceedings.

Alexander K. Petrenko,
Holger Schlingloff,
Nikolay Pakulin
March 2015

References

[1] Andreas Blass, Yuri Gurevich, Lev Nachmanson & Margus Veanes (2005): *Play to test*. Technical Report MSR-TR-2005-04, Microsoft Research. Available at http://research.microsoft.com/apps/pubs/

default.aspx?id=70129. A short version appears in: FATES 2005, LNCS vol. 3997, pp. 32-46.

[2] Igor B. Bourdonov, Alexander Kossatchev, Victor V. Kuliamin & Alexander K. Petrenko (2002): *UniTesK Test Suite Architecture*. In: *FME 2002: Formal Methods - Getting IT Right, International Symposium of Formal Methods Europe, Copenhagen, Denmark, July 22-24, 2002, Proceedings*, pp. 77–88, doi:10.1007/3-540-45614-7_5.

[3] Igor B. Burdonov, Alexander Kossatchev, Alexandre Petrenko & Dmitri Galter (1999): *KVEST: Automated Generation of Test Suites from Formal Specifications*. In: *FM'99 - Formal Methods, World Congress on Formal Methods in the Development of Computing Systems, Toulouse, France, September 20-24, 1999, Proceedings, Volume I*, pp. 608–621, doi:10.1007/3-540-48119-2_34.

[4] Wolfgang Grieskamp, Yuri Gurevich, Wolfram Schulte & Margus Veanes (2001): *Conformance Testing with Abstract State Machines*. In R. Moreno-Diaz & A. Quesada-Arencibia, editors: *Formal Methods and Tools for Computer Science; Proceedings of Eurocast 2001*. Available at http://research.microsoft.com/apps/pubs/default.aspx?id=77832.

[5] Wolfgang Grieskamp, Xiao Qu, Xiangjun Wei, Nicolas Kicillof & Myra B. Cohen (2009): *Interaction Coverage Meets Path Coverage by SMT Constraint Solving*. In: *Testing of Software and Communication Systems, 21st IFIP WG 6.1 International Conference, TESTCOM 2009 and 9th International Workshop, FATES 2009, Eindhoven, The Netherlands, November 2-4, 2009. Proceedings*, pp. 97–112, doi:10.1007/978-3-642-05031-2_7.

[6] Alan Hartman (2006): *Ten Years of Model Based Testing – A sober evaluation*. Available at https://www.react.uni-saarland.de/mbt2006/talks/ModelBasedTestingSoberEvaluation.PDF. Invited talk at the Second Workshop on Model Based Testing March 25-26, 2006 Vienna , Austria.

[7] Thomas J. Ostrand & Marc J. Balcer (1988): *The Category-Partition Method for Specifying and Generating Functional Tests*. *Commun. ACM* 31(6), pp. 676–686, doi:10.1145/62959.62964.

[8] Alexander K. Petrenko (1993): *Test Specfication Based on Trace Description*. *Programming and Computer Software* 19(1).

Program committee

- Bernhard Aichernig (Graz University of Technology, Austria)
- Jonathan Bowen (Birmingham City University, Worshipful Company of Information Technologists, UK)
- Mirko Conrad (samoconsult GmbH, TU Mnchen, Humboldt University of Berlin, Germany)
- John Derrick (University of Sheffield, UK)
- Bernd Finkbeiner (Universitat des Saarlandes, Germany)
- Lars Frantzen (Radboud University Nijmegen , Netherlands)
- Ziyad Hanna (Jasper Design Automation, USA)
- Philipp Helle (EADS, Germany)
- Mika Katara (Tampere University of Technology, Finland)
- Alexander S. Kossatchev (ISP RAS, Russia)
- Victor Kulaimin (ISP RAS, Russia)
- Bruno Legeard (Smartesting, France)
- Bruno Marre (CEA LIST, France)
- Laurent Mounier (VERIMAG, France)
- Nickolay Pakulin (ISP RAS, Russia)
- Jan Peleska (University of Bremen, Germany)
- Alexander K. Petrenko (ISP RAS, Russia)
- Alexandre Petrenko (Computer Research Institute of Montreal, Canada)
- Fabien Peureux (University of Franche-Comte, France)
- Holger Schlingloff (Fraungofer FIRST, Germany)
- Julien Schmaltz (Open University of The Netherlands, Netherlands)
- Nikolai Tillmann (Microsoft Research, USA)
- Stephan Weissleder (Fraunhofer FOKUS, Germany)
- Nina Yevtushenko (Tomsk State University, Russia)