

| | |
|-------------------------|--|
| 1. Record Nr. | UNINA9910143618703321 |
| Titolo | Abstract State Machines - Theory and Applications : International Workshop, ASM 2000 Monte Verita, Switzerland, March 19-24, 2000 Proceedings / / edited by Yuri Gurevich, Philipp W. Kutter, Martin Odersky, Lothar Thiele |
| Pubbl/distr/stampa | Berlin, Heidelberg : , : Springer Berlin Heidelberg : , : Imprint : Springer, , 2000 |
| ISBN | 3-540-44518-8 |
| Edizione | [1st ed. 2000.] |
| Descrizione fisica | 1 online resource (X, 386 p.) |
| Collana | Lecture Notes in Computer Science, , 0302-9743 ; ; 1912 |
| Disciplina | 511.3 |
| Soggetti | Computer logic Software engineering Programming languages (Electronic computers) Logic, Symbolic and mathematical Database management Logics and Meanings of Programs Software Engineering/Programming and Operating Systems Programming Languages, Compilers, Interpreters Software Engineering Mathematical Logic and Formal Languages Database Management |
| Lingua di pubblicazione | Inglese |
| Formato | Materiale a stampa |
| Livello bibliografico | Monografia |
| Note generali | Bibliographic Level Mode of Issuance: Monograph |
| Nota di bibliografia | Includes bibliographical references and index. |
| Nota di contenuto | Abstract State Machines at the Cusp of the Millenium -- Mathematical Foundations -- Abstract State Machines and Pure Mathematics -- Abstract State Machines and Computationally Complete Query Languages -- On Verification of Refinements of Timed Distributed Algorithms -- Abstract State Machine Languages -- Objects + Views = Components? -- XASM- An Extensible, Component-Based Abstract State Machines Language -- Generic Facilities in Object-Oriented ASMs -- Distribution and Concurrency -- Towards an ASM Thesis for Unconventional Algorithms -- Partially Ordered Runs: A Case Study -- Investigating Java Concurrency Using Abstract State Machines -- |

Compilers and Semantics -- Verifying Compilers and ASMs or ASMs for Uniform Description of Multistep Transformations -- An ASM Dynamic Semantics for Standard ML -- Modeling the Dynamics of UML State Machines -- On the Formal Semantics of SDL-2000: A Compilation Approach Based on an Abstract SDL Machine -- Description and Simulation of Microprocessor Instruction Sets Using ASMs -- Automatic Verification and Model Checking -- Symbolic Analysis of Transition Systems? -- Encoding Abstract State Machines in PVS -- Model Checking Abstract State Machines and Beyond -- Towards a Methodology for Model Checking ASM: Lessons Learned from the FLASH Case Study -- Industrial Applications -- Report on a Practical Application of ASMs in Software Design -- Using Abstract State Machines at Microsoft: A Case Study.

Sommario/riassunto

The ASM 2000 workshop was held in the conference center of the Swiss Federal Institute of Technology (ETH) at Monte Verit a, Canton Ticino, March 19-24, 2000. The ASM formalism was proposed together with the thesis that it is suitable to model arbitrary computer systems on arbitrary abstraction levels. ASMs have been successfully used to analyze and specify various hardware and software systems including numerous computer languages. The aim of the workshop was to bring together domain-experts, using ASMs as a practical specification method, and theorists working with ASMs and related methods. In addition the workshop served as a forum on theoretical and practical topics that relate to ASMs in a broad sense. Three tutorials including hands-on experience with tools were organized by U. G"asser and G. del Castillo (on the topic "Specifying Concurrent Systems with ASMs"), H. Russ " and N. Shankar (on the topic "A Tutorial Introduction to PVS"), M. Anlaug, P.W. Kutter, and A. Pierantonio (on the topic "Developing Domain Specific Languages"). In response to the organization committee's call for papers, 30 papers were submitted, each of which was independently reviewed by four members of the program committee. This volume presents a selection of 12 of the refereed papers and two reports on industrial ASM application at Siemens AG and Microsoft Research, together with contributions based on the invited talks given by A.
