Narciso Martí-Oliet, Kazuhiro Ogata (Eds.)

Rewriting Logic and Its Applications

15th International Workshop, WRLA 2024

Part of the European Joint Conferences on Theory & Practice of Software (ETAPS 2024)

Luxembourg City, Luxembourg, April 6–7, 2024.

Informal Proceedings

Preface

This volume contains the preliminary proceedings of the 15th International Workshop on Rewriting Logic and Its Applications (WRLA 2024).

Rewriting logic is a natural model of computation and an expressive semantic framework for concurrency, parallelism, communication, and interaction. It can be used to specify and verify a wide range of systems with their desired properties, and define domain specific languages in various application fields. It also has good properties as a metalogical framework for representing logics. Over the years, several languages based on rewriting logic have been designed and implemented. The aim of the workshop is to bring together researchers with a common interest in rewriting logic and its applications, and to give them the opportunity to present their recent work, discuss future research directions, and exchange ideas. The previous meetings were held in Asilomar (USA) 1996, Pont-à-Mousson (France) 1998, Kanazawa (Japan) 2000, Pisa (Italy) 2002, Barcelona (Spain) 2004, Vienna (Austria) 2006, Budapest (Hungary) 2008, Paphos (Cyprus) 2010, Tallinn (Estonia) 2012, Grenoble (France) 2014, Eindhoven (Netherlands) 2016, Thessaloniki (Greece) 2018, online 2020 (during Covid-19), and Munich (Germany) 2022.

WRLA 2024 was held on April 6–7, 2024, in Luxembourg City, Luxembourg, as a satellite event of the European Joint Conferences on Theory & Practice of Software (ETAPS 2024). We received 16 submissions; two were withdrawn due to out of the scope, and one was withdrawn due to out of the submission categories; each of the remaining (13 submissions) was reviewed by at least three program committee members. After an extensive discussion, the program committee decided to accept 13 papers for presentation at the workshop. The program of WRLA 2024 consisted of 13 paper presentations and five invited tutorials. The preliminary proceedings includes the five invited tutorial abstracts as well as the 13 papers (eight regular papers, three tool papers, and two education papers). A selection of the papers accepted for presentation will appear in the proceedings that will be published in the Springer LNCS series, following the tradition of previous meetings in this series.

We sincerely thank all the authors of papers submitted to the workshop, and the invited speakers for kindly accepting to contribute to WRLA 2024. We are grateful to the members of the program committee and the additonal reviewers for their careful work in the review process. We also thank the members of the WRLA steering committee for their valuable suggestions. Finally, we express our gratitude to all members of the local organization of ETAPS 2024, whose work has made the workshop possible.

April, 2024

Narciso Martí-Oliet
Kazuhiro Ogata

Table of Contents

Regular Papers

Tool Papers

Regular Papers

Education Papers

Program Committee

Erika Abraham	RWTH Aachen, Germany
Kyungmin Bae	POSTECH, Korea
Canh Minh Do	JAIST, Japan
Francisco Durán	University of Málaga, Spain
Santiago Escobar	Universitat Politècnica de València, Spain
Maribel Fernandez	King's College London, United Kingdom
Nao Hirokawa	JAIST, Japan
Alexander Knapp	University Augsburg, Germany
Temur Kutsia	Johannes Kepler University Linz, Austria
Alberto Lluch-Lafuente	Technical University of Denmark, Denmark
Dorel Lucanu	Alexandru Ioan Cuza University, Romania
Salvador Lucas	Universitat Politècnica de València, Spain
Narciso Martí-Oliet	Universidad Complutense de Madrid, Spain
José Meseguer	University of Illinois at Urbana-Champaign, USA
Aart Middeldorp	University of Innsbruck, Austria
Masaki Nakamura	Toyama Prefectural University, Japan
Kazuhiro Ogata	JAIST, Japan (Chair)
Peter Ölveczky	University of Oslo, Norway
Adrián Riesco	Universidad Complutense de Madrid, Spain
Christophe Ringeissen	INRIA, France
Camilo Rocha	Pontificia Universidad Javeriana, Colombia
Traian-Florin Serbanuta	University of Bucharest, Romania
Carolyn Talcott	SRI International, USA

Additional Reviewers

Duong Dinh Tran	JAIST, Japan
Rubén Rubio	Universidad Complutense de Madrid, Spain

Publicity Chair

Duong Dinh Tran	JAIST, Japan

Invited Tutorials

1. New Advances in Maude 3.4
   by Francisco Durán, University of Málaga, Spain.
   Maude is a High-Performance Logical Framework providing specification, programming, and verification of systems written in Rewriting Logic. This tutorial will report on the new features available in Maude 3.4.

2. NuITP: A New Theorem Prover for Maude Specifications
   by Francisco Durán, University of Málaga, Spain.
   NuITP is an inductive theorem prover for Maude equational specifications that combines powerful state-of-the-art techniques such as narrowing, equality predicates, constructor variant unification, order-sorted congruence closure, ordered rewriting, strategy-based rewriting, and several others in order to reason about Maude equational programs. This tutorial will introduce the main features of the tool through various examples.

3. The CafeInMaude Tool Set
   by Adrián Riesco, Universidad Complutense de Madrid, Spain.
   CafeOBJ is a specification language that has been used for verifying a wide variety of systems. In recent years, CafeInMaude, a CafeOBJ interpreter implemented in Maude, has served as the underlying platform for developing various formal tools to enhance the verification experience. Specifically, we have implemented: (i) the CafeInMaude Proof Assistant (CiMPA), an inductive theorem prover for CafeOBJ specifications; (ii) the CafeInMaude Proof Generator (CiMPG), which generates CiMPA scripts from proof scores; and (iii) the CafeInMaude Proof Generator & Fixer-Upper (CiMPG+F), which generates CiMPA scripts from scratch. We summarize the main features of these tools, the benchmarks used to evaluate them, and their future challenges.

4. Design and Validation of Cloud Storage Systems Using Maude
   by Peter Ölveczky, University of Oslo, Norway.
   Today's large cloud-based applications (e.g., Gmail, Facebook, etc.) store and manipulate enormous amounts of data, that, furthermore, must be available all the time. In addition, such applications must be both correct and have high performance.
   To deal with large amounts of data while offering high availability and throughput and low latency, cloud computing systems rely on distributed, partitioned, and replicated data stores. Such cloud storage systems are complex software artifacts that are very hard to design, analyze, and implement. I argue that Maude, together with a statistical model checker such as PVeStA, should be a suitable tool to model and formally analyze both the correctness

and the performance of complex cloud storage designs early in the development process. This is not only useful to arrive at correct designs, but also to very early compare the expected performance of different design choices.
This talk summarizes work done in the context of UIUC's Center for Assured Cloud Computing to apply Maude to a wide range of state-of-the-art cloud transaction systems, such as Apache Cassandra, Google's Megastore, UC Berkeley's RAMP transactions, and variations of these. I discuss how the model-based performance estimates relate to real implementations, and also how a correct design can be automatically transformed into a correct-by-construction distributed implementation that can execute real workloads (e.g., YCSB workloads).
Finally, I briefly summarize the experiences of the use of a different formal method for similar purposes by engineers at Amazon Web Services.

5. Formal Model Engineering of Distributed Cyber-Physical Systems in AADL Using Maude
by Kyungmin Bae, Pohang University of Science and Technology, South Korea.
Formal model engineering, equipping industrial modeling tools with automatic formal analysis capabilities, is a promising way of integrating formal methods into the model development process. However, supporting formal model engineering for cyber-physical systems (CPSs) introduces several challenges. These include the high expressiveness of industrial modeling languages, the difficulty of exhaustive model-checking analysis due to asynchronous behaviors, and the intricate mix of advanced control programs with continuous behaviors typical in many CPSs.
This talk outlines our approach to supporting efficient formal model engineering for synchronous CPS designs using the industrial CPS modeling standard AADL. We have identified a suitable sublanguage of AADL, called Synchronous AADL, that can naturally define the synchronous designs of CPSs, including those with continuous behaviors. We have defined the formal semantics of Synchronous AADL in rewriting logic and developed the HybridSynchAADL tool, an extension of the OSATE tool environment for AADL with automatic formal analysis capabilities via Maude and SMT solving.
Our approach effectively addresses the challenges by leveraging the expressive power of Maude for control programs, integrating Maude and SMT solving for analyzing continuous behaviors, and focusing on synchronous designs to enhance analysis efficiency. Additionally, we show how synchronizers, such as PALS, TTA, and MSYNC, can verify the correctness of distributed CPS implementations based on their synchronous designs. We summarize our experiences on applications such as industrial avionics control systems, airplane turning algorithms, and collaborating drones.