10351 Executive Summary
Modelling, Controlling and Reasoning About State
— Dagstuhl Seminar —

Amal Ahmed[1], Nick Benton[2], Lars Birkedal[3] and Martin Hofmann[4]

[1] Indiana University - Bloomington, US
amal@cs.indiana.edu
[2] Microsoft Research UK - Cambridge, GB
nick@microsoft.com
[3] IT University of Copenhagen, DK
birkedal@itu.dk
[4] LMU München, DE
mhofmann@informatik.uni-muenchen.de

Abstract. From 29 August 2010 to 3 September 2010, the Dagstuhl Seminar 10351 "Modelling, Controlling and Reasoning About State " was held in Schloss Dagstuhl – Leibniz Center for Informatics. 44 researchers, with interests and expertise in many different aspects of modelling and reasoning about mutable state, met to present their current work and discuss ongoing projects and open problems. This executive summary provides a general overview of the goals of the seminar and of the topics discussed.

Keywords. Mutable State, Program Logics, Semantics, Type Systems, Verification

1 Introduction

The combination of dynamically allocated, mutable data structures and higher-order features is present in almost all programming languages, from C to ML, Java and C♯. The search for good models and reasoning principles for, and language features that tame the complexity of, this combination goes back many decades. Recent years have seen a number of significant advances in our semantic understanding of state and encapsulation, including the development of separation logic, substructural type systems, models using parametric and step-indexed logical relations, and new bisimulation-based reasoning methods.

At the same time, concern about reliability, correctness and security of software has led to increased interest in tool and language support for specification and verification of realistic languages (for example JML and Spec♯), certified and certifying compilation, proof-carrying code, safe systems programming languages (such as Cyclone and CCured), and practical type systems capturing

Dagstuhl Seminar Proceedings 10351
Modelling, Controlling and Reasoning About State
http://drops.dagstuhl.de/opus/volltexte/2010/2810

and controlling subtle aspects of state, such as ownership, effects, information flow and protocol conformance. Formalizing the meaning and the soundness of these new languages, analyses and type systems is a major motivation for the development of the theory described above.

This is an exciting and important research area. Mathematically sound reasoning principles for state, combined with recent advances in program analysis and machine-assisted proof, have the potential to lead to improved programming languages, compilers, verification technology and even to new approaches to software deployment and operating system design.

This seminar built on the success of Dagstuhl Seminar 08061 'Types, Logics and Semantics for State', held in February 2008, though with slightly less emphasis on bringing together researchers from very different communities, and slightly more on in-depth technical discussion and collaboration on key technical issues such as the correct modelling of independence, recursive store, step-indexing and purity.

Among the research challenges addressed at the workshop were:

- What are the semantic foundations of existing logics and type systems for ownership, confinement, effects and permissions, and how may such foundations be used not only to understand and improve these systems, but also to relate them formally to one another?
- How can we reason about controlled use of state at multiple levels of abstraction, for example in relating high-level, language-enforced restrictions to low-level guarantees on the behaviour of compiled code?
- What is the right mix of approaches to the control of state and other effects, both in low-level languages and in modern high-level languages with higher-order features? How to balance language design and type systems, automated verification tools and machine assisted proof?
- What is the relationship between the recently appeared step-indexing method for establishing soundness of type systems and fully denotational approaches? In particular, how can denotational methods for mixed-variance equations for predicates and relations be transferred to step-indexed models of types, and how can the respective soundness properties, which are in general not logically equivalent, be compared?
- How can we quantify and use the additional information gained by modular analyses that do not require knowledge of the whole program? Is observational equivalence really the ultimate equivalence?
- How should we deal with the mixed-variance equations for predicates and relations that appear in the denotational modelling of storable procedures ("higher-order store"). In recent developments we have seen such equations that were solvable in an ad-hoc way but escaped the established solution theory.
- How can the algebraic approach developed for global state via Lawvere theories be extended to local state ?

2 Participation and Programme

The seminar brought together 44 researchers from Europe, Japan, Singapore and the United States with interests and expertise in all aspects of modelling and reasoning about programs with mutable state. There were 32 talks over the course of the week, including invited overview talks on particular topics shorter contributed talks on recent work, open problems, and issues that arose during the week's discussions. The overview talks were on the use of ultrametric spaces in semantics, the state of the art in logical relations, verifying liveness properties of higher-order programs, and automated verification based on separation logic.

It was clear that modelling and reasoning about state is still a vibrant and fast-advancing research area. Even since the previous seminar on these topics, very significant progress has been made in a number of areas, including semantic foundations, reasoning techniques and the pragmatics of mechanizing both of these in proof assistants and automated verification tools. Many talks concerned, or involved, formalizations using the Coq proof assistant. Parametricity and logical relations continue to be central ideas, and some very impressive recent results were presented. One of these was the construction of very powerful Kripke logical relations models for ML-like languages, dealing with nearly all the 'difficult' features: higher-order functions, polymorphism, general references, recursive types and control operators. Another was the use of logical relations to prove a compositional full functional correctness result for a compiler for such a language. Ideas from separation logic were used in many talks, with a new emphasis on how to make such reasoning more abstract and compositional. The technical device of 'step-indexing' was used in many talks, and is now starting to be better understood and investigated more closely in its own right. A particularly exciting recent development is the use (or rediscovery) of the uses of ultrametric spaces in semantics, in a sense generalizing both step-indexing and the use of sequences of projections in domain theory. The use of algebraic techniques for understanding local state was also an theme of increasing importance.

One of the most interesting developments was renewed attention to game semantics. Whilst game semantics has been an active and extremely successful area of research since the early 1990s, it has not quite broken through into the mainstream of 'applied' semantics. However, several recent developments have brought it to the attention of researchers who have hitherto used other techniques. One is the development of quite novel software model-checking techniques, built from the ground up on game semantic ideas. Another is the combination of nominal structure (which featured in several talks) with games; this has allowed fully abstract models to be built for both ML-like languages and the π-calculus. During the seminar, however, it also became clear that researchers using logical relations have started to incorporate elements of game semantics into their models. There is an exciting possibility of something of a rapprochement between the different styles of semantics in the near future. Bisimulation-based reasoning techniques were also much in evidence, and although the long-hoped-for unification with logical relations has still not quite happened, we did learn how bisimulation can be used to prove parametricity properties and see a comparison between bisimulations and other methods for reasoning about the nu-calculus. Recent progress in the verification of concurrent algorithms and in the static analysis of resource usage were both the topic of several talks and discussions.

This was an intense and productive week. With a relatively large number of particpants, most of whom wanted to speak, the schedule was relatively tight; though we did this time manage to incorporate the traditional hiking excursion on Wednesday afternoon. Discussions continued late into the night throughout the week, and were still unbelievably technical even at 2am! The proceedings contain five papers on techniques, all of which were inspired or influenced by discussions at the seminar.

The organizers and participants thank the staff and management of Schloss Dagstuhl for their assistance and support in the arrangement of a very successful meeting.