

Executive Summary

Dagstuhl Seminar 06261: Foundations and Practice of Programming Multi-Agent Systems

Mehdi Dastani¹, John-Jules Ch. Meyer¹, Rafael H. Bordini²

¹ Department of Computer Science
University of Utrecht, Netherlands

² Department of Computer Science
University of Durham, UK

The “Foundations and Practice of Programming Multi-Agent Systems” Dagstuhl Seminar aimed at bringing together researchers interested in programming languages for multi-agent systems, agent-oriented software engineering, and various related aspects such as verification, and formal semantics. We were delighted with the result of this seminar, which gave participants a clear view of the most advanced techniques being currently investigated in research on those topics throughout the world, and also a clear understanding of all the most important open problems that need to be addressed by this research community. The seminar was particularly successful in elucidating the relationship between work being done by the “programming languages for multi-agent systems” (ProMAS) research community and the “agent-oriented software engineering” (AOSE) research community. Even though the initiative for this seminar arose from the ProMAS community, we were delighted to attract many prominent researchers from the AOSE community, which allowed us to achieve the positive result on the connection of ProMAS and AOSE research.

In order to achieve the most productive setting for technical presentations and discussion in the various aspects we wanted to cover, we structured the talks in thematic days or half-days. In the first and last days, we included only AOSE-related talks. The second day was dedicated to ProMAS specifically, the morning of the third day we dedicated to Verification of Multi-Agent Systems; the fourth day had talks on Semantics in the morning and AOSE in the afternoon. Figure 1 shows in the detail the seminar schedule. We will now describe briefly each of those talks.

After a brief opening by the organisers, the theme of the day was “agent-oriented software engineering”. Michael Winikoff discussed the gap between agent-oriented software engineering and agent-oriented programming languages, and discussed approaches and ongoing work to try to bridge such gap. This was followed by Leon Sterling who argued for a development methodology that would be accessible to the broader IT community (rather than researchers in multi-agent systems). Viviana Mascardi presented a combined

	Mon 26 June AOSE	Tue 27 June ProMAS	Wed 28 June Verification	Thu 29 June Semantics / AOSE (afternoon)	Fri 30 June AOSE
9:00-10:30	Opening Winikoff	Dastani Hindriks	Lomuscio Bordini	Shapiro Sardiña	Jonker Hübner
10:30-10:45 Coffee Break					
10:45-12:15	Sterling Mascardi	Boella & Baldoni Sierhuis	Broersen Baldoni	Riemsdijk Ricci	Braubach & Pokahr Nickles
12:15-14:00 Lunch					
14:00-15:30	Gomez-Sanz Ricci	Steunebrink Novak		Hirsch Zhang	
15:30-16:00 Coffee Break					
16:00-18:00	Discussion group: AOSE	Discussion group: ProMAS		Discussion group: Semantics and Verification	

Figure 1: Dagstuhl Seminar 06261 Schedule

approach of agent methodologies and web-services technology. Jorge Gomez-Sanz advocated the use of model-driven methodologies, and ongoing work to support transition from specification to implementation. Alessandro Ricci then gave the first part of his talk (continued on the Thursday), in which he argued for a shift of multi-agent systems research from a close relationship with the artificial intelligence communities towards traditional software engineering and distributed computing research communities.

On the day themed “programming multi-agent systems”, Mehdi Dastani introduced 2APL, a new variant of 3APL with constructs that are useful in practical programming of multi-agent systems. Koen Hindriks discussed GOAL, a language that has essential features of agent-programming and has a corresponding logic. Matteo Baldoni and Guido Boella then jointly presented Powerjava, where agent notions such as roles are incorporated in an object-oriented language. Maarten Sierhuis then presented the Brahms approach to modelling and simulating organisations and communities of practice. Bas Steunebrink discussed the use of emotions within the deliberation cycle of an interpreter for 3APL. Peter Novak discussed a modular approach to combining different languages within a BDI architecture.

The session on the following day was on verification. Alessio Lomuscio gave an overview of his work on symbolic model checking for the verification of multi-agent systems. Rafael Bordini presented an algorithm for property-based slicing of AgentSpeak systems which can reduce the state-space when model checking such systems. Jan Broersen then discussed the issue of how to choose an appropriate logical formalism given the task of verifying a particular agent program. Matteo Baldoni presented an approach to checking interoperability of services in open environments such as the Web.

The following day had two different sessions, the first on semantics and the second on agent-oriented software engineering. In the first session, Steven Shapiro addressed the issue of how agents can drop goals believed to be impossible to achieve without being explicitly told so. Sebastian Sardiña presented an approach integrating HTN planning within BDI agents. Birna van Riemsdijk discussed the semantics of approaches to modu-

larisation within agent-oriented programming languages. In the second topic of the day, Alessandro Ricci gave the final part of his talk, now concentrating on the “Agents and Artifacts” approach to combining agent and object-oriented programming. Benjamin Hirsch discussed the relationships between agents and services, then presented the JIAC IV system. Yingqian Zhang then gave a talk on “survivability of multi-agent systems”, where she addressed deployment issues to avoid system failures and security attacks.

In the last day, we went back to agent-oriented software engineering, with the initial talks concentrating on the issue of organisations. Catholijn Jonker talked about organisation design, in particular including humans as part of the system. Jomi Hübner presented the Moise+ approach to modelling organisations, involving structural, functional, and deontic dimensions. Lars Braubach and Alexandar Pokahr jointly presented their work on integrating agent techniques into mainstream software technology. Matthias Nickles proposed an approach to engineering open multi-agent systems based on *evolving and reflective expectation*.

There were three discussion sessions, on the following general topics:

- agent-oriented software engineering;
- programming multi-agent systems;
- semantics and verification.

It was in the discussion sessions that many interesting ideas emerged. We were fortunate to have Maarten Sierhuis generating a detailed report of all discussions using the Compendium Tool (<http://www.compendiuminstitute.org/tools/compendium.htm>). He also kindly agreed to make the Compendium diagrams available online, which provides a complete report of all discussion sessions, so we feel it is not necessary to include that material in this summary. The interested reader will find all the details about the discussions we had at the following URL: <http://www.dur.ac.uk/r.bordini/Dagstuhl06261/> (choose “Discussions” in the main menu).

From the discussion session, we just would like to add a list of agent programming languages currently being used in university courses taught by the participants. The growing number of universities teaching agent-oriented programming languages is a clear indication of the increased maturity of the area and an excellent promise for a major growth of the paradigm in the medium term.

Anyone who has attended a Dagstuhl seminar agrees that Dagstuhl seminars are the most productive of all types of academic events. We were certainly expecting an excellent seminar, but this seminar exceeded our expectations in all aspects. With so many issues still to be resolved in this exciting and promising area of research, we are hoping that a follow up of this seminar will be organised in a couple of years, and we are looking forward to it already!

Acknowledgements

Many thanks to Maarten Sierhuis for the Compendium diagrams he generated on-the-fly during the discussion sessions, which allowed us to have a detailed account of the discussions we had. Many thanks also to Catholijn Jonker who also took notes during the discussion sessions.