Special Track on Coordination Models, Languages and Applications

Andrea Omicini George A. Papadopoulos
Track Chairmen

1 INTRODUCTION

Over the last decade we have witnessed the emergence of models, formalisms and mechanisms for describing concurrent and distributed computations based on the concept of *coordination*. The purpose of a coordination model is to enable the integration of a number of possibly heterogeneous components (processes, objects, agents) in such a way that the resulting ensemble forms a single application that can execute on as a whole and take advantage of parallel and distributed systems. The coordination paradigm is closely related to other contemporary software engineering approaches such as component-based systems and middleware platforms. Furthermore, the concept of coordination exists in many other Computer Science areas such as Cooperative Information Systems, Artificial Intelligence and Internet Technologies.

The Special Track on Coordination Models, Languages and Applications takes deliberately a broad view of what is coordination. In addition to the traditional areas covering data-driven (such as Linda) and control-driven (such as Manifold) models and languages, the track invited contributions from several areas where the concept of coordination is relevant, such as software architectures, middleware platforms, groupware and workflow management, multiagent systems, etc.

In response to the call-for-papers, 52 high quality submissions from 15 different countries were submitted to this special track. 21 of these submissions were forwarded to more appropriate tracks of SAC'2000, and the other 31 were fed into the reviewing process. There were altogether more than 70 reviewers and 150 reviews were submitted by them, an average of almost 5 reviews for each paper. Based on the reviewers' reports, the general ACM SAC guidelines for acceptance and rejection of submissions, and the unavoidable time and space constraints associated with any conference, it was possible to select only 17 of these submissions as regular papers and 2 more as short papers. In the process, a number of good and interesting papers had to be rejected.

2 THE CONTRIBUTED PAPERS

Two of the papers are concerned with software architectures and architecture description languages. In particular, [1] presents a formal semantics for the software architecture language Splice whereas [2] presents a formal model (using Z specifications) for describing ADLs.

A number of papers are concerned with mobile computing and agent issues. [3] extends the coordination language Manifold so that mobile components can be handled. [4] introduces the language KLAIM for coordinating interaction of mobile agents, [5] presents a mobile agents coordination platform based on XML, and [6] illustrates the use of the language GroupLog for coordinating ensembles of autonomous agents. Two papers are concerned with coordination in workflow management systems: [7] proposes a cooperation environment between workflows executing in the same organisation or in different ones, whereas [8] argues for the need for formal, logic-based techniques for representing workflows.

The Linda coordination model has inspired a number of the papers in this track, which are concerned with extending the basic model with new features. [9] extends tuple handling with scoping rules, [10] develops more "laws" for safer use of the tuple space, [11] introduces some optimisation techniques for the *in* primitive. Also, [12] develops a calculus for modelling event notification in data-driven languages.

Coordination in scientific computing has also been of concern to some of the papers in this track. In particular, [13] addresses issues of coordination for languages supporting both task- and data-parallelism, [14] presents the Network of Tasks - a directed graph coordination model for node programs written in a variety of languages, and [15] illustrates the use of Activity Graphs as an intermediate coordination formalism for expressing the functionality of skeletons.

Finally, a number of other issues are addressed by the rest of the papers. [16] exploits Manifold to solve constraint satisfaction problems, [17] develops an object-oriented component-based system, [18] uses the configuration paradigm for modelling distributed multimedia environments, and [19] explores issues of coordination particular to transaction systems, especially with regard to data mining.

BIOGRAPHIES

Andrea Omicini holds a Laurea in Electronic Engineering (1991) and a Ph.D. in Computer System Engineering (1995), both from the Alma Mater Studiorum, the University of Bologna (Italy). He has participated and he is still actively involved in a number of national and international research programs (CNR Project "Computer Systems and Parallel Computations", MOSAICO, AgentLink, etc.). He is currently a Researcher in the DEIS Department and a Professor of Computer Science in the Faculty of Engineering at the University of Bologna (Italy). His research interests include multi-paradigm programming languages, multiagent systems, coordination models and languages, and intelligent system engineering.

George A. Papadopoulos holds a B.Sc. in Computer Science and Mathematics (1982) and an M.Sc. in Computer Science with Applications (1983), both from the University of Aston in Birmingham, UK, and a Ph.D. in Computer Science (1989) from the University of East Anglia, Norwich, UK. He has participated and he is still actively involved in a number of national and international research programs (Alvey's Flagship, ESPRIT II's EDS and PCA, MED-CAMPUS, INCO-DC, etc.). He is currently an Associate Professor in the Department of Computer Science at the University of Cyprus in Nicosia, Cyprus. His research interests include parallel programming, concurrent object-oriented programming techniques, design and implementation of declarative (concurrent constraint and functional) programming languages, coordination models and languages, and multimedia systems. Professor Papadopoulos is a recipient of an ERCIM Fellowship Award for 1994-95 supported financially by EU's Human Capital and Mobility programme.

REFERENCES

Following, in this volume:

[1] Roel Bloo, Jozef Hooman, and Edwin de Jong. Semantical issues in the architecture of distributed embedded systems.
[2] Kurt Lichtner, Paulo Alencar, and Don Cowan. An extensible model of architecture description.
[3] Farhad Arbab, Marcello Bonsangue, and Frank de Boer. A coordination language for mobile components.
[4] Lorenzo Bettini, Michele Loreti, and Rosario Pugliese. Structured nets in KLAIM.
[5] Giacomo Cabri, Letizia Leonardi, and Franco Zambonelli. XML dataspaces for mobile agent coordination.
[6] Fernanda Barbosa and Jose C. Cunha. A coordination language for collective agents based systems: GroupLog.
[7] Fabio Casati and Angela Discenza. Supporting workflow cooperation within and across organizations.
[8] Jacques Wainer. Logic representation of processes in work activity coordination.
[9] Iain Merrick and Alan Wood. Coordination with scopes.
[10] Naftaly H. Minsky, Yaron M. Minsky, and Victoria Ungureanu. Making tuple spaces safer for heterogeneous distributed systems.
[11] Antony Rowstron. Optimising the Linda in primitive: Understanding tuple space run-times.
[12] Nadia Busi and Gianluigi Zavattaro. Event notification in data-driven coordination languages: Comparing the ordered and unordered interpretations.
[13] Salvatore Orlando, Paolo Palmerini, and Raffaele Perego. Coordinating HPF programs to mix task and data parallelism.
[14] David B. Skillicorn and Susanna Pelagatti. Building programs in the network of tasks model.
[15] Murray Cole and Andrea Zavanella. Activity Graphs: A model-independent intermediate layer for skeletal co-ordination.
[16] Eric Monfroy. A coordination-based chaotic iteration algorithm for constraint propagation.
[17] Sander Tichelaar, Juan Carlos Cruz, and Serge Demeyer. Design guidelines for coordination components.
[18] Ahmed Saleh and George R.R. Justo. A configuration-oriented framework for distributed multimedia applications. Short paper.
[19] Srinath Srinivasa and Myra Spiliopoulou. Analyzing transaction data for building coordination models. Short paper.