

Lecture Notes in Computer Science

1222

Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Advisory Board: W. Brauer D. Gries J. Stoer

Jan Vitek Christian Tschudin (Eds.)

Mobile Object Systems

Towards the Programmable Internet

Second International Workshop, MOS'96
Linz, Austria, July 8-9, 1996
Selected Presentations and Invited Papers



Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany

Juris Hartmanis, Cornell University, NY, USA

Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Jan Vitek

University of Geneva, Computer Science Department, Object Systems Group

CH-1211 Geneva 4, Switzerland

E-mail: jvitek@cui.unige.ch

Christian Tschudin

University of Zurich, Computer Science Department

CH-8057 Zurich, Switzerland

E-mail: tschudin@ifi.unizh.ch

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Mobile object systems : towards the programmable Internet ; second international workshop ; selected presentations and invited papers / MOS '96, Linz, Austria, July 8 - 9, 1996. Jan Vitek ; Christian Tschudin (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Santa Clara ; Singapore ; Tokyo : Springer, 1997

(Lecture notes in computer science ; Vol. 1222)

ISBN 3-540-62852-5 kart.

CR Subject Classification (1991): C.2, D.1.3, D.1.5, D.2.4, H.4.3, I.2

ISSN 0302-9743

ISBN 3-540-62852-5 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1997

Printed in Germany

Typesetting: Camera-ready by author

SPIN 10549577 06/3142 - 5 4 3 2 1 0 Printed on acid-free paper

Preface

This book is a collection of papers on mobile agents and mobile object systems. The papers touch on many issues, both conceptual and implementation oriented, related to this new programming paradigm. Before we dwell on the details it is always fair to ask: If mobile objects are the solutions, what is the problem?

Computer science history, albeit relatively short, often repeats itself. We are now facing a situation similar to that of the 1960s. At the time we had a new hardware/software architecture (multiprocessing/multiprogramming/virtual memory) and we were trying to offer a utility to customers (time sharing systems). It took some time for the programming-language and operating-system specialists to come up with the proper programming paradigms. Since then we have seen many novel architectures (parallel computers, neural networks) but none have had the impact of what we are facing now. The Internet is the new architecture. Based on the Internet an immense utility is evolving for information, communication, and media services. So if the Internet *is* the machine, what is the proper programming paradigm to make it easier to develop the novel applications that this utility must offer?

We need a quantum jump in our conceptual framework. The interest of practitioners is so great that any new proposal for a tool or an environment, e.g., Telescript or Java, gets immediate attention. Eventually this attention is diluted by the specific details, and the discussion becomes influenced by commercial interests. The problem remains, what is the proper programming paradigm to program Internet applications? Needless to say, I do not believe that we have the answer yet. Not in this book and not in general. There are attempts, proposals, ideas, and tools but nothing that everybody would agree on.

It will not be so easy to find the right paradigm and model. Both in parallel computers and in neural networks we have had limited success with new programming paradigms and new tools. The Internet combines all the problems we have faced so far. Its parallelism is mind-boggling. It has the connectivity of neural networks. It is asymmetric with respect to connectivity bandwidths. It is dynamic in all its aspects. Last but not least, it is practically uncontrollable. For such an environment we need a paradigm which is highly robust and can deal with uncontrolled and even chaotic situations.

It is my opinion that mobile agents will offer such an environment, provided we can agree on what mobile agents are and what facilities they offer. It will take both discussion and experimentation. This book should be considered as a forum for discussion of mobile agents. In a few years we will be in a situation where defining terms like mobile agents is no longer needed, just as now nobody needs to define what is meant by "procedure" or "process". When this point is reached, then we can say that we have a new programming paradigm.

Table of Contents

Introduction	IX
-------------------------------	----

Part I: Foundations

Mobile Computation (<i>Luca Cardelli</i>)	3
Objectworld (<i>Denis Tsichritzis</i>)	7
Commentary on “Objectworld” (<i>Denis Tsichritzis</i>)	23
Mobile Agents: Are They a Good Idea? (<i>David Chess, Colin Harrison and Aaron Kershenbaum</i>)	25
Update (<i>David Chess, Colin Harrison and Aaron Kershenbaum</i>)	46
A Note on Distributed Computing (<i>Jim Waldo, Geoff Wyant, Ann Wollrath and Sam Kendall</i>)	49
Afterword (<i>Jim Waldo, Ann Wollrath and Sam Kendall</i>)	65
Instruction-Based Communications (<i>Christian Tschudin</i>)	67

Part II: Concepts

Analyzing Mobile Code Languages (<i>Gianpaolo Cugola, Carlo Ghezzi, Gian Pietro Picco and Giovanni Vigna</i>)	93
Sumatra: A Language for Resource-Aware Mobile Programs (<i>Anurag Acharya, M. Ranganathan and Joel Saltz</i>)	111
Migratory Applications (<i>Krishna Bharat and Luca Cardelli</i>)	131
The Messenger Environment $M\emptyset$ – A Condensed Description (<i>Christian Tschudin</i>)	149
Mobility and Persistence (<i>Miguel Mira da Silva</i>)	157
Security and Communication in Mobile Object Systems (<i>Jan Vitek, Manuel Serrano and Dimitri Thanos</i>)	177
Safe and Secure Execution Mechanisms for Mobile Objects (<i>Kazuhiko Kato</i>)	201
Jada – Coordination and Communication for Java Agents (<i>Paolo Ciancarini and Davide Rossi</i>)	213

Part III: Implementation

Performance-Oriented Implementation Strategies for a Mobile Agent Language (<i>Frederick Knabe</i>)	229
--	-----

Dynamic Linking for Mobile Programs (<i>Anurag Acharya and Joel Saltz</i>) . . .	245
Adaptive Compression of Syntax Trees and Iterative Dynamic Code Optimization: Two Basic Technologies for Mobile Object Systems (<i>Michael Franz</i>)	263
A Type-Based Implementation of a Language with Distributed Scope (<i>Dominic Duggan</i>)	277
Interaction of Java and Telescript Agents (<i>Peter Dömel</i>)	295
 Appendix	
Glossary	315
List of Authors	317

Introduction

This book presents a collection of papers dealing with different aspects of *mobile computations*. Mobile computations are computations that are not bound to single locations, but may move at will to best use the computer network's resources. In this view, the network becomes a single, vast, *programmable* environment. Among computer scientists, many feel that this approach will have a profound effect on the way we design and implement distributed applications, and they agree that we are witnessing a paradigm change. However, this new and exciting paradigm requires advances, both theoretical and applied, in fields such as programming languages (where we need a sound semantic foundation and efficient implementations), operating systems and software safety and security. Some of the first steps towards a programmable Internet are documented here.

This book follows a particular approach to mobile computation. It emphasizes the synergy between mobility and object-oriented programming, hence the title *Mobile Object Systems*. Mobile object systems, in our view, are self-contained and autonomous groups of objects. They carry out a computation for an end-user in some initial computational environment and may dynamically change their environment for one that is on a remote computer. There is a close relationship between mobile object systems and research on so-called *mobile software agents*. In both cases, the focus is software mobility, but we prefer the terms *mobile computation* and *mobile object systems* as they are more accurate descriptions of the technology, while *mobile agents* has a fuzzy meaning that overlaps with artificial intelligence research. The papers in this book are more concerned with problems of software mobility per se. Nevertheless, the terms are quite close and are often used interchangeably even within the present work.

The starting point of this book was a number of discussions and presentations given at the second International Workshop on Mobile Object Systems¹ (MOS'96) held in 1996 in conjunction with the European Object Oriented Programming Conference (ECOOP'96) in Linz. The core of the book is made up of reworked versions of the submitted papers. However, we wanted to broaden the scope of this volume and survey a large portion of the research in this rapidly expanding field. We invited a number of researchers to contribute reprints of important papers or to write entirely new pieces. This book is the result.

Overview

The book is organized in three parts: (I) Foundations, (II) Concepts, and (III) Implementation, followed by an appendix. We detail the content of each part next.

Part I of the book contains chapters giving background and motivation for the research on mobile computations. The chapter by Cardelli is a brief introduction to the issues and challenges of mobile computation. It is followed by a reprint of a paper by Tsichritzis, written twelve years ago, which describes many of the features we are looking for in mobile object systems using the animal world as a metaphor for mo-

¹ The workshop home page is <http://cuiwww.unige.ch/~ecoopws>.

mobile computations. The chapter by Waldo et al. was written in 1994 and convincingly argues against transparency in distributed computing. To some extent it motivates the work on mobile computations, as mobile computations naturally suggest making mobility and location visible to the programmer. The chapter by Chess et al. investigates advantages and disadvantages of mobile computations by looking at their application. The last chapter, by Tschudin, contains a philosophical and philological discussion of messages and instructions.

Part II contains descriptions of systems and concepts for mobile computations. The chapter by Cugola et al. analyzes languages that support some form of code mobility, trying to compare them and get at some of the basic principles of those languages. The chapter by Acharya et al. presents Sumatra, a Java based environment for mobile applications. The chapter by Bharat and Cardelli presents Visual Obliq, an implementation of mobile computation based on Obliq. The chapter by Tschudin presents a messenger environment. The chapter by Mira da Silva discusses the relationship between persistence and mobility. The chapter by Vitek et al. considers security for communication between object systems. The chapter by Kato looks further into security issues. The chapter by Ciancarini and Rossi presents an architecture for coordination and communication on wide area networks that can be used between mobile computations or plain Java programs.

Part III contains papers detailing implementation considerations and techniques. The chapter by Knabe looks at the trade-offs between different representations of agents and efficiency. In the second chapter, Franz presents a particular representation called Slim Binaries which is particularly well suited to mobile code, as well as a more general dynamic code optimization technique. The chapter by Dugan describes the implementation of mobility of polymorphic data in a strongly typed programming language. Finally, the paper by Dömel discusses the implementation of a system that allows Java programs to interact with Telescript agents.

Acknowledgments

We would like to thank the program committee of the MOS'96 workshop, Joachim Baumann, Luca Cardelli, Paolo Ciancarini, and Doug Lea, for their help and excellent reviewing.

February 1997

J. Vitek and C. Tschudin
Geneva and Zurich