Preface

This book, situated at the intersection of behavior-based artificial intelligence and concurrent and distributed computing, defines programming paradigms to support the design and the concurrent and distributed implementation of *Multi-Agent Systems* (MAS) that simulate collective robotics applications. We analyze the research that has tried to fill the gap between agent theory and applications, observing that the proposed methodologies, languages, and tools are mostly concentrated on intra-agent aspects. In contrast to those approaches, we propose that the modeling of MAS should be a bottom-up and interaction-oriented process, grouping existing autonomous agents and describing how they interact, thus managing the *coordination* of these agents. To that aim, we distinguish *objective* coordination, which handles inter-agent dependencies (the organization of the environment and the agent interactions), and *subjective* coordination, which handles intra-agent dependencies often involving mentalistic categories. We then promote a methodology that focuses the modeling of MAS on objective coordination, and we propose the use of *coordination models* and corresponding *languages* (from the fields of concurrent and distributed computing, programming languages, and software engineering) in order to respectively support the design phase of a MAS and allow its implementation on a concurrent and distributed architecture.

After reviewing coordination models and languages, we examine the prerequisites that a coordination model and language should include in order to support our target MAS. On this basis, a general coordination model named ECM and a corresponding object-oriented coordination language named STL++ are presented. ECM/STL++ uses an encapsulation mechanism as its primary abstraction, offering structured separate name spaces which can be hierarchically organized. Agents communicate anonymously within and/or across name spaces through connections, which are established by the matching of the communication interfaces of the participating agents. Three basic communication paradigms are supported, namely point-to-point stream, group, and blackboard communication. Furthermore, an event mechanism is introduced for supporting dynamicity by reacting to state changes of the communication interfaces.

The use of ECM/STL++ is illustrated by the simulation of a particular collective robotics application and of the automation of a trading system.

Acknowledgements

This book compiles four years of research work that I carried out as a member of the Parallelism and Artificial Intelligence group of the Computer Science Department at the University of Fribourg in Switzerland, working on a PhD thesis funded by the Swiss National Science Foundation. I benefited enormously from the knowledge, the expertise, and the support of many people.

My special thanks go to: Béat Hirsbrunner who gave me the opportunity to work in a highly qualified research group and who has guided my research activity during the past years; Rolf Ingold, Andrea Omicini, and George Papadopoulos, for accepting the invitation to act as jury members; Fabrice Chantemargue and Oliver Krone for the numerous discussions, ideas, criticisms, and also for their constant help and enthusiasm; Antony Robert, Valerio Scarani, Michele Schumacher, and Robert Van Kommer for reviewing previous versions of this document; Christian Wettstein for implementing the first version of STL++; Ivan Doitchinov for implementing AGENT&CO; all colleagues in the PAI group for creating a stimulating research environment; and last but not least, my family and my friends for their constant support and encouragement.

January 2001 *Michael Schumacher*

Foreword

This book addresses the engineering of Multi-Agent Systems (MAS) based on suitably-defined coordination models and languages. Its contribution is twofold: on the one hand, the theoretical part defines a new coordination model (ECM) for designing MAS and proposes two corresponding coordination languages (STL++ and AGENT&CO) for implementing designed systems in distributed environments; on the other hand, the experimental part presents a software prototype by starting off the existing PT-PVM (a distributed threads environment developed by Oliver Krone, a former PhD student of our PAI research group) and validates the theoretical part with two case studies. Definitely one of the main result is that the ECM model and its explicit distinction of objective and subjective coordination define an original and comprehensive framework for modeling, designing, and implementing interactions in complex MAS.

This work has been carried out within the framework of the Parallelism and Artificial Intelligence research group (PAI) at the University of Fribourg, Switzerland (`http://www-iiuf.unifr.ch/pai`). The PAI group originated from a synergy between the fields of Distributed Computing and Artificial Intelligence. Since 1986, it has been involved in massively distributed programming methodologies, and since 1995 in decentralized control strategies as far as they are suited for providing adaptive capabilities in network computing. Research topics of the group follow an engineering trend, which can be characterized by:

- The abandonment of centralized control and stringent hierarchical data structures in favor of decentralized control strategies based on interactions through influences. These strategies, which require autonomous components, lead to solutions, which are more flexible, more tolerant to perturbations, and which support the emergence of new properties;
- A bio-inspired approach, which draws its models from neuro-biology and the study of animal societies and participates in the concept of embodiment of intelligence now surfacing in AI;

Between 1995 and 1999, PAI's research fitted inside the AXE project (AXE is a compressed acronym for "Autonomy and Coordination Supports Evolution") whose aim was to devise a self-organization and a coordination theory for massively distributed systems. AXE's main acquisitions concern (i) the development of coordination platforms for distributed applications; (ii) the definition of models for autonomous, adaptive, and evolutive agents; (iii) the design of coordination and evolution models, namely collective intelligence strategies. Since 1998, human computer interaction aspects are also part of PAI's research, in the measure that universal networking and ubiquitous computing formulate special requirements in this field. With AXE's termination at the beginning of 2000, this domain sets the new focus adopted by PAI inside the new WELCOME framework, building on the know-how acquired on autonomous systems and coordination.

It was a pleasure for me to work with Michael Schumacher. His multidisciplinary research results have been well received in the MAS and coordination communities. The modeling, design, and implementation of MAS, the ECM coordination model, and the STL++ and AGENT&CO coordination languages, which are the central parts of his PhD thesis described in this book, are now used in our projects presently under development in WELCOME.

Finally, I warmly acknowledge the financial support of the University of Fribourg, the Swiss National Science Foundation, and the European Programme of R&D for having granted PAI's research projects for the past fourteen years.

December 2000

Béat Hirsbrunner
Leader of the PAI research group