Preface

In the last few years, the search for radically new approaches to software engineering has witnessed a great momentum. These efforts are well justified by the troubling state of present day computer science.

Software engineering practices based on design-time architectural composition (the only assessed way of doing software engineering so far), lead to brittle and fragile systems, unable to gracefully cope with reconfiguration and faults. While such practices can be acceptable when dealing with software systems to be deployed in closed and static scenarios, they are definitely unsuitable for most emerging computing scenarios.

More and more, software systems involve autonomous and distributed software components that have to execute and interact in open and dynamic environments. This is the case of information economies, pervasive and mobile computing systems, wide-area Internet applications, and P2P computing. In all these scenarios, the dynamism, openness, and decentralization of the application's operational environments call for new approaches to software design and development, capable of supporting spontaneous configuration and networking, and capable of tolerating partial failures and adaptive reorganization of the software system.

Hints for the feasibility of such innovative approaches can come from a variety of natural systems. The process of morphogenesis in organisms demonstrates that well-defined shapes and functional structures can develop through the interaction of cells under the control of a genetic program, even though the precise arrangements and numbers of the individual cells are variable. The process of ant foraging demonstrates how the application goal of finding and carrying home food in hostile environments can be achieved by simple interactions among a multitude of individuals of limited intelligence.

By getting inspiration from natural systems, scientists and engineers are starting to understand that, to construct self-organizing and adaptive systems, it may be more appropriate focusing on the engineering of proper interaction mechanisms for the components of the system, rather than on the engineering of their overall system architecture.

In line with the above consideration, this book focuses on a physically inspired interaction model, i.e., field-based coordination. Field-based coordination relies on virtual computational fields, mimicking gravitational and electromagnetic fields, as the basic mechanisms with which to coordinate activities in open and dynamic ensembles of application components. This enables components to spontaneously interact with each other by the mediation of fields and – as in physical systems – to self-organize in an adaptive way their activity patterns. All of this with the additional advantage that – unlike in real-world physical systems – one can shape fields according to any needed virtual physical law, to achieve a variety of coordination patterns in support of a variety of application goals.

This book summarizes in a readable and accessible way some four years of work in the area of advanced field-based coordination models. The specific model presented in this book together with the middleware technologies that have been developed to support it, define a general-purpose approach for the engineering of self-organizing adaptive applications in a number of scenarios. The title of the book evokes the fact that the model was originally conceived for multiagent systems in pervasive computing scenarios. However, we invite readers to consider it as reflecting the fact that field-based coordination may be suitable for all systems made up of autonomous interacting components (agents *de facto*), from sensor networks to P2P computing systems, that will soon pervade our everyday environments.

Additional material for this book, including code of the simulations and of the TOTA middleware, can be found at the Web site of the Agents and Pervasive Computing Group, *http://www.agentgroup.unimore.it.*

Acknowledgments

A number of persons have directly or indirectly contributed to this book.

We thank all the members of the Agents and Pervasive Computing Group at the University of Modena and Reggio Emilia, for their continuous support and frienship during these years.

We thank Alfred Hofmann and Ralf Gerstner from Springer, for having supported this work and for having tolerated our delays.

A final special thanks is extended to all our students (now engineers) who have actively contributed to our researches with notable implementation and experimental work.

Reggio Emilia,
May 2005,
Marco Mamei and Franco Zambonelli