

1

Introduction

Agent-Oriented Programming (AOP) is a relatively new software paradigm that brings concepts from the theories of artificial intelligence into the mainstream realm of distributed systems. AOP essentially models an application as a collection of components called agents that are characterized by, among other things, autonomy, proactivity and an ability to communicate. Being autonomous they can independently carry out complex, and often long-term, tasks. Being proactive they can take the initiative to perform a given task even without an explicit stimulus from a user. Being communicative they can interact with other entities to assist with achieving their own and others' goals. The architectural model of an agent-oriented application is intrinsically peer to peer, as any agent is able to initiate communication with any other agent or be the subject of an incoming communication at any time.

Agent technology has been the subject of extensive discussion and investigation within the scientific community for several years, but it is perhaps only recently that it has seen any significant degree of exploitation in commercial applications. Multi-agent systems are being used in an increasingly wide variety of applications, ranging from comparatively small systems for personal assistance to open, complex, mission-critical systems for industrial applications. Examples of industrial domains where multi-agent systems have been fruitfully employed include process control, system diagnostics, manufacturing, transportation logistics and network management.

When adopting an agent-oriented approach to solving a problem, there are a number of domain-independent issues that must always be solved, such as how to allow agents to communicate. Rather than expecting developers to develop this core infrastructure themselves, it is convenient to build multi-agent systems on top of an agent-oriented middleware that provides the domain-independent infrastructure, allowing the developers to focus on the production of the key business logic.

This book describes JADE (Java Agent DEvelopment framework), probably the most widespread agent-oriented middleware in use today. JADE is a completely distributed middleware system with a flexible infrastructure allowing easy extension with add-on modules. The framework facilitates the development of complete agent-based applications by means of a run-time environment implementing the life-cycle support features required by agents, the core logic of agents themselves, and a rich suite of graphical tools. As JADE is written completely in Java, it benefits from the huge set of language features and third-party libraries on offer, and thus offers a rich set of programming abstractions allowing developers to construct JADE multi-agent systems with relatively minimal expertise in agent theory. JADE was initially developed by the Research & Development department of Telecom Italia s.p.a., but is now a community project and distributed as open source under the LGPL licence.

The JADE website is <http://jade.tilab.com>. In addition, a book companion website from Wiley is available at http://wiley.com/go/bellifemine_jade.

The book's intended audience is primarily application developers with goals to both provide a comprehensive explanation of the features provided by JADE and also to serve as a handbook for programmers. Many of the features discussed are supported with exemplary code, either ad hoc or positioned with the context of a pervasive 'book trading' example. All code snippets and application examples provided in the book relate to version 3.4.1 of JADE released on November 2006; although we expect this material will remain consistent with future versions and is, in the most part, backward compatible with previous versions of JADE.

The book is structured as follows: Chapter 2 presents an overview of agent technology discussing the most relevant agent-related concepts, architectures and tools. Moreover, it provides a summary of the FIPA specifications that represent the most important standardization activity conducted in the field of agent technology. Chapter 3 presents the core architecture of JADE, its components and supported features. It also shows how to start the platform, launch agents and use the graphical administration tools. Chapters 4, 5 and 6 focus on the main Application Programming Interfaces (APIs) that JADE provides to access its features. Chapter 4 describes the basic features, Chapter 5 the advanced features and Chapter 6 is dedicated to agent mobility. Chapter 7 presents the internal architecture of the JADE kernel and explains how to modify and extend its behaviour. Chapter 8 addresses the development and deployment of JADE-based applications in the mobile and wireless environment. Chapters 9, 10 and 11 present additional configurations and tools that can be used to solve issues that must typically be taken into account when deploying real-world applications. Chapter 12 then discusses the JADE Semantic Framework, an important and recently released module that exploits the formal semantics of messages exchanged by agents. Finally, Chapter 13 gives an overview of other relevant tools that can be used with JADE.