Control and Cybernetics

vol. **36** (2007) No. 2

Book review:

ORGANIZATIONAL PRINCIPLES FOR MULTI-AGENT ARCHITECTURES

by

Chris van Aart

Characterisation of the subject domain of the book

Scientists consider multi-agent systems as a domain within artificial intelligence. For software developers, a multi-agent system it is a paradigm for distributed software design. Even though engineering of such systems has been evolving for roughly ten years, we cannot consider it as a mature information technology. We already have at our disposal many platforms for multi-agent system development, and we can still observe many research projects devoted for this paradigm of distributed information systems - although multi-agent approach is far from such a wide acceptance, as is characteristic for the object oriented or web services paradigms.

The first international conference on multi-agent systems (ICMAS-95) took place in June 1995 in San Francisco. FIPA (Foundation for Intelligent Physical Agents) is the first organization, formed in 1996, to produce software standard specifications for heterogeneous and interacting agents, and for agent-based systems. The IEEE officially accepted FIPA on 8 June 2005. Today in the FIPA repository we can find several hundred documents. Status of most of them is still experimental. The multi-agent paradigm encounters high interest from the side of the IT community – there are already about one thousand books related to multi-agent systems. A remarkable part of these publications are conference proceedings, lecture notes in computer science and PhD dissertations.

There is no commonly accepted and precise definition of a multi-agent system. Even the description given in Wikipedia contains a clause “This article or section may contain original research or unverified claims.” We can define an agent as a computer system that is capable to execute autonomous actions in its environment in order to achieve predefined goal. Thus, a multi-agent system is composed of several or many agents, collectively capable of reaching goals – as a result, frequently agents are interdependent. FIPA defines an agent as “a computational process that implements the autonomous, communicating functionality of an application.”

Multi-agent systems can play a role in large-scale problem solving systems, which we find in very different domains e.g.: web search services, distributed intrusion detection systems, distributed warehouse management, and so on. Due to the globalisation of economy, the markets that are more and more open, and the ubiquity of Internet, the large-scale problem solving systems will be more and more desirable. Hence, the need for multi-agent systems design will grow in a near future. This is the reason why Chris van Aart's book can be recommended for software developers.

His book presents pragmatic results of a scientific work on multi-agent system conceptualisation. The leading idea of this research was that an organization of a multi-agent system could be developed in the same way as an enterprise organization. For scientists it is nothing extraordinary to look for rules and solutions in technical science and to apply them in human sciences, or vice versa. However, engineers are rather sceptical about profiting from any idea having come from human sciences. This book demonstrates that for the multi-agent domain it could be fruitful to create a design technique on the basis of human organization schemas.

Chris van Aart demonstrates how to build an agent organization framework pragmatically, using some knowledge from enterprise organizations. This framework is based on three research lines: organization modelling, interoperability and agent models. Interesting for complex system design is the organization model, which distinguishes coordination tasks from processing tasks. The author characterizes several organization structures, which can suit different needs. The interoperability model distinguishes four abstraction levels: *technical* (transport), *syntactic*, *semantic* and *coordination*. The first three levels correspond to traditional interoperability structures in agent communication systems, when the new *coordination level* allows regulating communication patterns and the flow of information. The author characterizes several strategies for coordination. Selection of a strategy depends on an application of the multi-agent system. The third research line leads to the definition of the agent design principles based on the five capability models: *communication model*, *competence model*, *self model*, *planner model* and *environment model*. These models are design patterns that help conceptualise different agents, which will cooperate in a system. The capabilities have to be made concrete during agent design. Simple agents can have only *communication* and *competence models*. Designer has to decide about the complexity of an agent, first of all so as to eliminate the internal functional overhead.

The author defines a framework on a symbol level for every research line, which makes it useful for any concrete project. A project can adapt a subset of elements defined in the framework, just to suit the complexity and needs. The book illustrates every described method by a simple case study, which can be seen as constituting an evidence for the presented concepts.

Place against the literature of the subject domain

Chris van Aart is not the first one who demonstrates the links between human organizations and computational systems. Already in the 1980s, the relations between human organizations and computational systems were described in technical journals. Some research works on applying human organization rules to multi-agent systems have been presented in technical papers in 2000 and 2001. The work of Chris van Aart enhances the previously described ideas and imbeds them in symbolic frameworks, creating practical patterns for developers.

We can classify books in multi-agent domain into three categories: application oriented, describing implementation tools and techniques, and describing conceptual approaches to design. Chris van Aart book belongs to the third category.

Examples of recent books that belong to the application category are:

- *Agent Technology For E-Commerce* by Maria Fasli (2007);
- *Distributed Autonomous Robotic Systems 7* by Maria Gini and Richard Voyles (2006);
- *Innovations in Design & Decision Support Systems in Architecture and Urban Planning* by Jos P. Van Leeuwen and Harry J.P. Timmermans (2006);
- *IP Network-Based Multi-Agent Systems for Industrial Automation: Information Management, Condition Monitoring and Control of Power Systems* by David P., Buse and Qing-Hua, Wu (2006);
- *Multiagent Engineering: Theory and Applications in Enterprises (International Handbooks on Information Systems)* by Stefan Kirn, Otthein Herzog, Peter Lockemann, and Otto Spaniol (2006).

An example of recent books that belong to the implementation category is:

- *Multi-Agent Programming: Languages, Platforms and Applications (Multiagent Systems, Artificial Societies, and Simulated Organizations)* by Rafael H. Bordini, Mehdi Dastani, Jürgen Dix, and Amal El Fallah Seghrouchni (2005).

Examples of recent books that belong to the conceptual category are:

- *Field-Based Coordination for Pervasive Multiagent Systems* by Marco Mamei and Franco Zambonelli (2005);
- *Ontologies for Agents: Theory and Experiences* by Valentina Tamma, Stephen Cranefield, Timothy W. Finin, and Steven Willmott (2005).

The multi-agent domain still evolves. The new concepts complement one another. A concept achieves maturity when it converts into a standard or is incorporated into a designing tool. Before it happens, engineers can apply the concepts in innovative projects. The concepts presented by Chris van Aart should be taken into consideration in such projects.

The primary addressees

The primary addresses of this book are: software architects and developers of distributed systems.

Indication of the particularly valuable aspects of the book

The particular value of this book is the systematic and pragmatic approach for development of a multi-agent system. It teaches how to think about complex multi-agent systems, and how to build functional decomposition and internal architecture of such systems.

Editorial assessment

The book is really well written and well organised. Someone who is not an IT professional can read it and learn a lot of. The book contains some UML diagrams with so fine description, that even a person who does not know UML can understand them. The bibliography provided demonstrates a strong scientific background of this book; moreover a reader can find a lot of useful Internet links. The author, when introducing any subject, gives valuable references pointing to related publications and engineering tools available on the net.

Jacek Wytrębowicz

Chris van Aart, *Organizational Principles for Multi-Agent Architectures.* Series: WSSAT – Whitestein Series in Software Agent Technologies. Birkhäuser Verlag, Basel - Boston - Berlin; 2005, 216 pages. ISBN 3-7643-7213-2. Price (softcover): 34 EUR.