Chapter XLV
Caste-Centric Development of Agent-Oriented Information Systems

Lijun Shan
National University of Defense Technology, China

Rui Shen
National Laboratory for Parallel and Distributed Processing, China

Ji Wang
National Laboratory for Parallel and Distributed Processing, China

Hong Zhu
Oxford Brookes University, UK

ABSTRACT

Based on the meta-model of information systems presented in Zhu (2006), this chapter presents a caste-centric agent-oriented methodology for evolutionary and collaborative development of information systems. It consists of a process model called growth model, and a set of agent-oriented languages and software tools that support various development activities in the process. At the requirements analysis phase, a modelling language and environment called CAMLE supports the analysis and design of information systems. The semi-formal models in CAMLE can be automatically transformed into formal specifications in SLABS, which is a formal specification language designed for formal engineering of multi-agent systems. At implementation, agent-oriented information systems are implemented directly in an agent-oriented programming language called SLABSp. The features of agent-oriented information systems in general and our methodology in particular are illustrated by an example throughout the chapter.

INTRODUCTION

In Zhu (2006), we presented a vision of future information systems through an agent-oriented meta-model. The promising features of the meta-model were illustrated in the context of software development on the Internet/Web platforms and the utilisation of mobile computing devices. In this chapter, we address the problem of how to develop such agent-oriented information systems (AOISs). Based on the meta-model introduced in Zhu (2006), we propose a methodology for developing an AOIS which consists of a process model that guides the development activities, along with a set of languages and software tools that support various development activities in the process.

The chapter is organised as follows. We begin by describing an information system used as the running example in the chapter. We then propose an evolutionary development process model for AOIS and outline the caste-centric agent-oriented modelling language and environment, CAMLE. The next section reviews the formal specification language SLABS, which stands for a Specification Language for agent-based systems. The focus then turns to implementation issues, and the SLABSp experimental programming language is briefly described. We conclude the chapter with a discussion of related work and further work.

DESCRIPTION OF THE RUNNING EXAMPLE

We will use a simple, but non-trivial, information system to illustrate our methodology as a running example throughout the chapter. The example was proposed and used as a case study by FIPA's AUML Technique Committee (2004) to study agent-oriented modelling methods and notations. It was inspired by the procedure of the United Nations Security Council to pass a resolution. The description of the system follows.

The United Nation Security Council (UNSC) consists of a number of members, some permanent and others elected from UN members. Members become the chair of the Security Council in turn monthly. To pass a UNSC resolution, the following procedure is followed:

1. At least one member of UNSC submits a proposal to the current chair.
2. The chair distributes the proposal to all members of UNSC and sets a date for a vote on the proposal.
3. At a given date that the chair sets, a vote from the members is made.
4. Each member of the Security Council can vote either FOR or AGAINST or ABSTAIN.
5. The proposal becomes a UNSC resolution, if the majority of the members voted FOR and no permanent member voted AGAINST.
6. The members vote one at a time.
7. The chair calls members in a given order to vote, and the chair is always the last one to vote.
8. The vote is open (in other words, when one votes, all the other members know the vote).
9. The proposing member(s) can withdraw the proposal before the vote starts, and in that case no vote on the proposal will take place.
10. All members vote on the same day, one after another, so that the chair does not change within the vote call; but it is possible for the chair to change from one member to another between the time a proposal is submitted until it goes into vote. In this case the earlier chair must forward the proposal to the new one.
11. A vote is always finished in one day and no chair change happens on that day. The date of the vote is set by the chair.

14 more pages are available in the full version of this document, which may
be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/caste-centric-development-agent-oriented/21160

Related Content

Certification Attainment - A Gizmo to Evaluate Provider's Trust: Trust Evaluation is Grounded on
Provider's Attainment Status Concerning Recommended Certifications
Archana B. Saxenaand Meenu Dave (2020). *International Journal of Natural Computing Research (pp. 1-12).*
www.irma-international.org/article/certification-attainment---a-gizmo-to-evaluate-providers-trust/241938

Evolutionary Algorithms for Multisensor Data Fusion
Jianhua Yang, Evor L. Hines, John E. Sloper, D. D. Iliescuand Mark S. Leeson (2011). *Intelligent Systems
for Machine Olfaction: Tools and Methodologies (pp. 62-77).*
www.irma-international.org/chapter/evolutionary-algorithms-multisensor-data-fusion/52448

Addressing the Challenges of Detecting Epistasis in Genome-Wide Association Studies of
Common Human Diseases Using Biological Expert Knowledge
Kristine A. Pattinand Jason H. Moore (2011). *Handbook of Research on Computational and Systems
Biology: Interdisciplinary Applications (pp. 128-147).*
www.irma-international.org/chapter/addressing-challenges-detecting-epistasis-genome/52314

Green Computing and Its Impact
Shailendra Singhand Sunita Gond (2017). *Nature-Inspired Computing: Concepts, Methodologies, Tools,
and Applications (pp. 1628-1642).*
www.irma-international.org/chapter/green-computing-and-its-impact/161086

An Improved Artificial Bee Colony Algorithm for the Object Recognition Problem in Complex
Digital Images Using Template Matching
Chidambaram Chidambaramand Heitor Silvério Lopes (2010). *International Journal of Natural Computing
Research (pp. 54-70).*
www.irma-international.org/article/improved-artificial-bee-colony-algorithm/45886