Chapter VIII

Designing Agent-Based Process Systems—Extending the OPEN Process Framework

J. Debenham
University of Technology, Sydney, Australia

B. Henderson-Sellers
University of Technology, Sydney, Australia

ABSTRACT

Originally a development methodology targeted at object technology, the OPEN Process Framework (OPF) is found to be a successful basis for extensions that support agent-oriented software development. Here we describe the process components necessary to agent-oriented support and illustrate the extensions by means of two small case studies that illustrate the extensions by means of two small case studies that illustrate both task-driven processes and goal-driven processes. The additional process components for Tasks and Techniques are all generated from the OPF’s metamodel, which gives the OPE its flexibility and tailorability to a wide variety of situations—here agent-orientation.
INTRODUCTION AND BACKGROUND

Modern software engineering encompasses a wide range of areas of interest. Three of particular interest are object orientation, component-based development, and intelligent multiagent systems. While all three have a different genesis, they all have elements in common, notably, modularity and information hiding with a clear distinction between interface (or specification) and implementation. Intelligent agents have emerged from artificial intelligence research (Wooldridge, 1997), whereas intelligent agent software engineering and methodologies have a firmer relationship to object technology (e.g., Wooldridge et al., 2000). The majority of agent-oriented methodologies have an object-oriented (OO) heredity despite observations that with previously published OO methodologies there is inadequate support for agents; while others adapt knowledge engineering techniques (Wooldridge & Ciancarini, 2001).

Of course, OO methodologies that are highly prescriptive and overly specified are hard to extend when a new variant or a new paradigm appears. What is required is a more flexible approach to building methodologies or processes. One such approach will be utilized here: OPEN (Object-oriented Process, Environment and Notation; Graham et al., 1997). Using OPEN, process components are selected from a repository, and the actual methodology (or process) is constructed using identified construction and tailoring guidelines. We identify here what support must be added to OPEN to fully support agents. Following this identification, we give an example application in the context of the development of business process management systems in which the processes may be decomposed into conditional sequences of goals. Goal orientation, as embodied in this class of applications, is readily accommodated in the existing and new tasks and techniques of the OPEN Process Framework (OPF; Firesmith & Henderson-Sellers, 2002) leading to a specification of the individual agents in the system.

DESIGNING AGENT-BASED SYSTEMS

A multiagent system is a society of autonomous components, each of which may be constructed independently. They may also be mobile (across hardware platforms and networks)—this aspect is out of the scope of this paper and is not discussed further. Agents can be regarded as powerful versions of objects, constructed using intelligent machinery that supports their autonomous nature and their capability to “take the initiative.” The high-level
Trends in Improving Performances in Distributed Database Management Systems
[www.igi-global.com/chapter/trends-improving-performances-distributed-database/37045?camid=4v1a](www.igi-global.com/chapter/trends-improving-performances-distributed-database/37045?camid=4v1a)

Service Innovation Analytics: Leveraging Existing Unstructured Data to Assess Service Innovation Capability
[www.igi-global.com/article/service-innovation-analytics/112039?camid=4v1a](www.igi-global.com/article/service-innovation-analytics/112039?camid=4v1a)