Chapter 4
Agent-Driven Semantic Interoperability for Cross-Organisational Business Processes

Ingo Zinnikus
German Research Center for Artificial Intelligence (DFKI) GmbH, Germany

Christian Hahn
German Research Center for Artificial Intelligence (DFKI) GmbH, Germany

Klaus Fischer
German Research Center for Artificial Intelligence (DFKI) GmbH, Germany

ABSTRACT

In cross-organisational business interactions, integrating different partners raises interoperability problems especially on the technical level. The internal processes and interfaces of the participating partners are often pre-existing and have to be taken as given. This imposes restrictions on the possible solutions for the problems which occur when partner processes are integrated. The aim of this chapter is the presentation of a three-tier framework for managing and implementing interoperable and cross-organizational business processes. Thereby the authors want to fill the gap currently existing between processes defined on a strategic level and executed models. We describe a solution which supports rapid prototyping by combining a model-driven framework for cross-organisational business processes with an agent-based approach for flexible process execution. We show how the W3C recommendation for Semantic Web service descriptions can be combined with the model-driven approach for rapid service integration.

INTRODUCTION

Today’s enterprises operate in a dynamic environment which is characterized by global outsourcing, shrinking product life-cycles, and unstable demand. To prosper in this environment, enterprises face a growing need to share information and to collaborate at all levels of the value chain. As organizations are gradually transforming into “networked organiza-
tions”, interoperability becomes the main challenge to realize the vision of seamless business interaction across organizational boundaries.

In cross-organisational business interactions, the most desirable solution for integrating different partners would suggest to integrate their processes and data on a rather low level. However, the internal processes and interfaces of the participating partners are often pre-existing and have to be taken as given. Furthermore, in cross-organisational scenarios partners are typically very sensitive about their product data and the algorithms that process it. In many cases, private processes are only partially visible and hidden behind public interface descriptions (Schulz & Orlowska, 2004). This imposes restrictions on the possible solutions for the problems which occur when partner processes are integrated.

Service-oriented architectures (SOA) (Erl, 2005) are today’s favourite answer to solve interoperability issues. It enables partners to offer the functionality of their systems via a public service interface (e.g. as WSDL description) and hide the sensitive parts behind it. A very important second advantage of SOA is the possibility of a loose coupling of partners. New partners can enter the system with little effort whereas obsolete partners are able to leave it easily. Especially in the case where e.g. additional smaller non-Original Equipment Manufacturers (OEM) are integrated in a sales process, the system needs to become robust against temporary unavailable partners.

Despite the advantages of a SOA, several difficulties arise especially in the case where the systems of the partners have evolved independently for several years:

- The philosophies of the systems differ, e.g. one partner service uses a strict sequential run through the product space whereas another service allows e.g. randomly browsing through the products and product features.
- The granularity of operations of the various partner services differs.
- Non-functional aspects such as exception handling, session management, transactional demarcation, which differ from partner to partner, supersede the core functionality of the services.
- Structural differences in the payload data of the exchanged messages stemming from data models used by the different partners’ sites are present.
- Semantical and pragmatic misunderstandings of the exchanged messages may arise due to different tagging of business data, different conventions, protocols etc.

These are typical interoperability problems occurring in cross-organisational scenarios which require SOA-based solutions. The European FP6 project ATHENA (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications) provided a comprehensive set of methodologies and tools to address interoperability problems of enterprise applications in order to realize seamless business interaction across organizational boundaries.

A core idea in the ATHENA project was to bring together different approaches and to combine them into a new framework: a modelling approach for designing collaborative business processes (CBP), a model-driven development framework for SOAs and an agent-based approach for flexible execution (see Figure 1). For modelling SOAs a platform independent metamodel for SOAs (PIM4SOA, Benguria et al., 2006) was developed in ATHENA. It turned out that these approaches fit nicely together, as e.g. the PIM4SOA metamodel and the agents’ metamodel bear a striking resemblance to each other.

The agent paradigm can be considered as promising approach to executable SOA in a nice manner. Several agent-based programming platforms exist to implement agent systems that offer different advantages when executing SOA. For the purpose of improving the interoperability between those platforms, we investigated a plat-