Chapter 7
Interoperability Support for E–Business Applications through Standards, Services, and Multi–Agent Systems

Rainer Unland
Institute for Computer Science and Business Information Systems (ICB), University of Duisburg-Essen, Germany

ABSTRACT
Electronic business has become more and more popular in the digital world and, especially, on the Internet. It not only promises productivity gains but may also help to substantially simplify business processes and value chains (from the human beings’ point of view). However, in order to exploit its full potential business-critical information and transactions need to be expressed in such a way that they can be easily and automatically exchanged, understood, and processed by information systems especially also across company boundaries. On the one hand this means that the use of common standards, protocols, and conventions on representations are fundamental to the success of electronic business. But we argue that this is just the first necessary step or the syntactical foundation for successful electronic business. A second step, which we consider to be the semantic step on top on the syntactical one, is to allow information systems to automatically, dynamically, and flexibly adapt to the fast changing requirements of electronic businesses. From our point of view, one promising candidate to achieve this is service-oriented computing. It stands for a standards-based, however, technology-independent distributed computing paradigm and architectural style. Based on a comprehensive set of independent or at most loosely-coupled and network-available software services service-oriented computing provides a platform for an efficient and effective discovery, assembly, and integrated execution of these services. Thus, this chapter will mainly concentrate on this second part, especially also, because the first part (electronic business standards) was already extensively discussed in other parts of this book. But we will
not only concentrate on the discussion of the service-oriented computing paradigm. Instead, it will be argued that the current state-of-the-art of it is still pretty preliminary and that this discipline can profit quite a lot from the integration of multi-agent systems technology.

Intelligent agents can be regarded as autonomous, problem-solving computational entities with social abilities that are capable of effective pro-active behavior in open and dynamic environments. If the term entity is replaced by service the substantial overlap in interests between both communities can easily be imagined. Nevertheless, right now the main research focus of each community seems to be different. The service-oriented computing community concentrates mainly on developing service engineering methodologies. Active topics in the multi-agent systems community are collaboration, self-organization, adaptability, flexibility, proactiveness, and interoperability. The overlap between those two communities and the fact that they concentrate on different research topics can definitely be seen as a huge chance since it means that each community may be able to benefit from the research efforts of the other.

**INTRODUCTION AND MOTIVATION**

Since the introduction of computers one dream has been to be able to just let the computer know what kind of problem one wants to be solved and leave it to the computer to develop, run, and manage the corresponding software. One of the first serious approaches in that direction was Z (Spivey, 1988). Z provides a set of mathematical (formal) conventions which roughly allows the user to model a computer system (hardware as well as software!) by describing its state and the necessary operations that can change its state (cf. Jacky, 1997). While Z may have been a first approach to pave the way for a declarative description of computing systems it never got popular, mainly due to its theoretical foundation (which could never be hidden at its interface) and due to the (too) many restrictions that came with it. Many other less ambitious approaches for software development on more abstract, thus, less algorithm-oriented levels were proposed but none of them got really close to the above mentioned goal of an easy to understand purely declarative application description language.

**Electronic Business**

The introduction of the Internet and the WWW once again changed the way of thinking in computer science. Suddenly an abundance of data, information, and services were available for everybody. It also took not long till the Internet was discovered as a great business opportunity for industry. IBM coined the term electronic business (e-Business for short) in 1999. It comprises all forms of automated and computer-aided data exchange carried out within companies and between business partners over electronic networks like the Web, the Internet, intranets, extranets, or some combination of these. It usually requires companies to link their internal and external data processing systems efficiently and flexibly in order to work more closely with suppliers and partners, and to better satisfy the needs and expectations of their customers. E-Business involves business processes spanning the entire value chain, as, e.g., electronic purchasing and supply chain management, processing orders electronically, handling customer service, and cooperating with business partners. E-Commerce, as a specialization of e-Business, restricts itself to the buying and selling of products or services over the Internet and other open computer networks.