Semantically Empowered Service-oriented Architectures

Dieter Fensel et al., DERI

In [Fensel et al., 2002] two major challenges for applied computer science had been defined:

- dealing with heterogeneity and fast change rates in data, protocol, and process formats of components that need to be integrated in an automated fashion;
- mechanization of component discovery, adaptation, composition, and monitoring.

Both aspects were identified as being mission critical for any software engineering approach that is based on reusing existing *components* rather than trying to re-invent the wheel from scratch. That is, both aspects are mission critical for modern software engineering which is faced with the fact that integration in heterogeneous environments and integration with large investments in legacy systems are essential practical requirements. Currently, computer science is in the next period of abstraction. A generation ago, we learnt to abstract from hardware and currently we learn to abstract from software in terms of *service-oriented architectures (SOA)*. It is the service that counts for a customer and not the specific software or hardware component that is used to implement this service. Service-oriented architectures will quickly become the leading software paradigm. However, SOAs will not scale without significant mechanization of service discovery, service adaptation, negotiation, service composition, service invocation, and service monitoring; as well as data, protocol, and process mediation. Therefore, these recent efforts around service-oriented architectures provide even increased urgency for the above mentioned fundamental requirements for any scalable and economic software solution. In consequence, machine processable semantics need to be added to bring SOAs to their full potential. Only then, important subtasks can be delegated to the computer and users can focus on the service they need. This is the mission and the vision of *W<Triple>* that add semantics to service descriptions injecting semantic description into the core of modern software architectures leading to *Semantically Empowered Service-oriented Architectures (SESA)*, see Figure 1 for an overall view of it.

At the lowest level, we define the **technical kernel** of our approach. This kernel consists ofthree main elements: Syntactical instruments to specify service-oriented architectures, semantical instruments to specify service-oriented architectures, and their proper alignment to provide a workable technological platform that fulfils the requirements as discussed earlier.

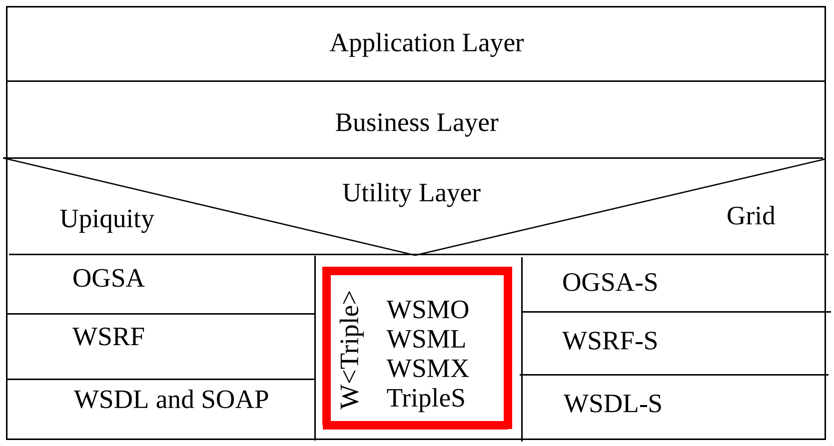

Fig. 1 Semantically empowered service-oriented architectures (SESA).

Syntactical instruments are WSDL, WSRF, and OGSA. The *Web Service Description Language (WSDL)* provides a means to access services via the definition of endpoints and operations. The *Web Service Resource Framework (WSRF)* adds the standardized description of stateful resources (i.e., services), i.e., it defines conventions for managing state so that applications discover, inspect, and interact with stateful resources in standard and interoperable ways. Finally, the *Open Grid Services Architecture (OGSA)* represents an evolution towards a service-oriented architecture based on Web services concepts and technologies. **Semantical instruments are WSMO, WSML, WSMX and TripleS**. The *Web Service Modeling Ontology (WSMO)*[1] provides a conceptual model for adding semantics to service-oriented architectures. Its main elements are goal definitions of user, service definitions of providers, and ontologies and mediators as declarative and procedural means to facility interoperability at the level of data, protocols, and processes. The *Web Service Modeling Language (WSML)* is a family of languages providing formal semantics for WSMO models. Its four major dialects form a lattice based on rule languages and descriptions logics as well as on their minimal and maximal intersection. The *Web Service Execution Environment (WSMX)* is an execution environment for the dynamic discovery, selection, mediation, invocation and inter-operation of the Semantic Web Services providing a sample implementation for WSMO. Finally, *triple-space computing (TripleS)* (cf. [Fensel, 2004], [Bussler, 2005]) is a new communication paradigm for services based on the semantic web paradigm. Communication is achieved via persistent publication and explicit definition of semantics via meta data definitions. **Combining syntax and semantics leads to WSDL-S, WSRF-S, and OGSA-S**. A straightforward and promising proposal has been made by *WSDL-S*. A simple extension of WSDL provides a grounding mechanism for semantic specifications of otherwise syntactical entities. Currently, efforts are underway to extend WSDL-S to a WSMO complaint grounding mechanism and similar efforts are requested for WSRF and OGSA.

The next layer build on top is the **utility layer**. It customizes service-oriented architecture towards the computational setting required by a class of applications. Two endpoints in this spectrum are ubiquitus computing on the one hand and grid-based computing on the other hand. The former is characterized by low-end devices and fast change rates of service providers whereas the latter is characterized by computational complexity in more stable settings. On top of these technology centred layers, the **business layer** introduces the application-oriented point of view. By defining generic types of business scenarios, a bridge towards application scenarios is introduced in this architecture. Finally, the **application layer** is the level where domain, task, and application specific definitions are introduced to provide actual added value to the users of the software artifact.

As a future evolvement of software-oriented architectures, we may even talk in terms of problem-oriented architectures (or more positively expressed in terms of *problem-solving* oriented architectures) because SOAs are biased towards the service provider and not towards the customer who has a problem that needs to be solved.

[Bussler, 2005] C. Bussler: A Minimal Triple Space Computing Architecture. In *Proceedings of the 2nd WSMO Implementation Workshop*, Innsbruck, Austria, June 2005.

[Fensel, 2004] D. Fensel: Triple-space computing: Semantic Web Services based on persistent publication of information, In *Proceedings of the IFIP International Conference. on Intelligence in Communication Systems November 2004*, Bangkok, Thailand, Lecture Notes in Computer Science (LNCS), no ????, Springer-Verlag, Berlin, 2004.

[Fensel et al., 2002] D. Fensel, C. Bussler, Y. Ding, and B. Omelayenko: The Web Service Modeling Framework WSMF, *Electronic Commerce Research and Applications*, 1(2), 2002.

1. http://www.wsmo.org/