A Web Service Market Model based on Dependencies

Robert Tolksdorf
Freie Universität Berlin,
Institut für Informatik
Netzbasierte
Informationssysteme
Takustrasse 9,
D-14195 Berlin, Germany
research@robert-tolksdorf.de

Christian Bizer
Freie Universität Berlin,
Institut für Produktion,
Wirtschaftsinformatik und
Operations Research
Garystrasse 21,
D-14195 Berlin, Germany
bizer@wiwiss.fu-berlin.de

Ralf Heese
Humboldt University Berlin,
Dpt. of Computer Science
Databases and Information
Systems
Unter den Linden 6,
D-10099 Berlin, Germany
rheese@informatik.hu-berlin.de

ABSTRACT

The construction of composite Web Services from service fragments requires semantic descriptions of service offers and service requests. We propose the use of dependencies as a modeling concept to describe service requests and service offers and outline a Web Service Market, that constructs composite services, offered by a temporary network of economically independent service providers, by resolving dependencies based on coordination theory.

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed Systems - distributed applications

General Terms

Design, Economics, Experimentation, Security

Keywords

Coordination, Semantic Web, Web Services

1. INTRODUCTION

The work described in the following has two goals in the development and deployment of Semantic Web Services (SWS):

1. Explore the use of the notion of dependencies as a modeling concept to capture semantic characteristics. It will be used in service representation, comparison, composition and coordination.

2. Study the technical and economic presumptions of an open web service market, where buyers can request complex, composite services, offered by a temporary network of economically independent service providers.

Semantic Web Services consider the semantic aspects of service description, discovery, composition. We combine a specific approach to model SWS by using the notion of dependencies with an economic perspective on their deployment. The technical aspects offer a flexible approach for a new way to describe and discover Web Services. The economic aspects focus on issues that will ultimately determine whether service composition over Web Service Markets will play a role in real world business processes.

Our *Web Service Market* will contain a coordination environment which matches business process requests and business process fragment offers in order to construct an executable workflow which matches the requirements of all parties. The process fragments offered in the market are either atomic services or composite services which aggregate services from different service providers.

2. USING DEPENDENCIES FOR SERVICE DESCRIPTIONS

For the description of service offers and service requests the notion of dependencies is our core modeling concept. Dependencies exist amongst services and service fragments. An example common dependency is temporal: a certain service fragment must be completed before the next starts or the overall service is completed.

Also, there exist technical and economic dependencies between the service request and service offers. The satisfaction of a request can depend on the fulfillment of technical or functional characteristics of the offer (e.g. a certain service interface or technical quality parameters), but also on non-functional and economic characteristics (e.g. price, price/time ratio, trust in the service providers etc.). Given similar technical aspects of offers, the non-functional characteristics become decisive for the selection of an offerer.

We consider the dependencies of service requests and service fragment offers as *semantic characteristics*. We assume that a Semantic Web Service and a request for it can be described completely by a set of dependencies. Following the ideas from coordination theory [4] the market architecture will contain mechanisms to resolve dependencies. Based on this ability, it can perform semantic service comparison and selection as well as service composition in order to construct executable service flows. Some first research on using dependencies as a modeling concept for Web Services has been undertaken in [8] and [2].

Our work will include the development of a basic model of dependencies and a collection of common dependencies from various models of coordination [7] and ontologies thereof. This will lead to an RDF-based representation for the semantic description of Web Services. Based on dependency resolution, a market architecture for SWS comparison, composition and coordination will be built.

3. MARKET ARCHITECTURE

This section gives a brief overview on how we integrate dependencies into the concept of process models and on how service requests will be answered by our market architecture. At some points an example will illustrate the use of dependencies.

WWW2003, May 20–24, 2003, Budapest, Hungary.
ACM xxx.

3.1 Hierarchies of Process Models

In our context a process model consists of a set of composite or atomic Web Services together with their dependencies. Applying to Web Services the concepts of abstraction/specialization for processes [9] and the two dimensions parts and types when analyzing processes [3] hierarchies of process models can be built. We specialize a service by adding new or changing dependencies. Therefore the most generic process is “do something realized by someone” and the most specialized one is “do a concrete task described by a set of dependencies and realized by a given provider” — we refer to them as abstract and concrete services, respectively.

Let us consider the placement of an advertisement supplement in a journal as an example. Contained services may include a designer service, a printing service, a delivery service and a accounting office. Provider and requester define their set of technical dependencies as there are medium, format, file format and publishing date of an ad as well as service flow. So defining the maximum size of an ad results in a specialization of this composite service.

Also, non-functional dependencies are defined. Examples are pricing, duration of service provision or the specification of a preferred service provider.

3.2 Processing of Service Requests

A service request in the form of an abstract process model with its set of dependencies is handled by a coordination service in two steps. First the coordination service searches for the process model in the hierarchy of the service market that best fits and satisfies all dependencies. Secondly abstract services contained in the found process model are specialized step by step until they are all concrete, always checking that no given dependency is violated. So with dependencies the requester can influence the specialization of a process model in some extent.

The coordination service is a core component that does the matchmaking between the process hierarchy and the request. This component may be provided by a marketplace operator or be user-defined.

Turning back to our example the requester may not need some services, e.g. because a self-made supplement should be published. In this case the coordination service has to find process models where the printing service is absent or can be skipped.

3.3 Management of Dependency

When managing dependencies we need to consider a conceptual representation and a proper management system. The resolution of technical dependencies is relative to some execution and communication model which is given by the Web Services standards.

The representation of dependencies in a service market needs to satisfy constraints like openness, usability and interoperability and to use a common vocabulary. Therefore a dependency ontology will be necessary based on Semantic Web technology, e.g. the Web Ontology Language.

In a real world service market with hundreds of thousands of service requests and offers, the representation of dependencies leads to a large amount of metadata that needs proper management to efficiently perform the construction of an executable service flow. An efficient and in particularly scalable storage and querying system will be required for this non-standard DBMS application.

3.4 Trust as an enabling factor

Beside of proper service descriptions, a main success factor for an automated service market will be the creation of trust amongst the parties involved. The level of trust depends on the proper identification of transaction partners, data and transaction security, service level and warranty agreements and the way privacy issues are handled. To enable trust, a portfolio of transaction independent trust-enabling services has to be integrated in the market architecture. Trust requirements are subjective and depend on the transaction value. In order to mediate between offered and required trust information and credentials, an ontology for different trust and security mechanisms will be developed and integrated in the market architecture. Basic security mechanisms for Web Services are already being standardized in the OASIS Web Service Security initiative. The special focus of our work is the role reputation mechanisms [6, 1, 5] can play in the matchmaking process on the service market.

4. CONCLUSION

In our work we capture the semantic characteristics of Web Services offers and requests using a set of dependencies. We described a marketplace to construct executable service flows by resolving dependencies. The modeling with dependencies makes it possible to match process parts on different abstraction/specialization levels. Secondly we stress the role of trust-enabling services in the market architecture and develop a framework for integrating different trust mechanisms.

5. REFERENCES

[1] Chris Bizer. Design von Reputationsmechanismen für online Communities. In *4. Paderborner Frühjahrtagung, Modelle im E-Business*, 2001. www.alb.fhg.de/download/4pbft/maerkte/bizer.pdf.

[2] Mark Klein and Abraham Bernstein. Searching for services on the semantic web using process ontologies. In Isabel Cruz, S. Decker, J. Euzenat, and D. McGuinness, editors, *The Emerging Semantic Web - Selected papers from the first Semantic Web Working Symposium*, pages 159–172, Amsterdam, 2002. IOS press.

[3] Thomas W. Malone et al. Tools for Inventing Organizations: Towards a Handbook of Organizational Processes. *Management Science*, 45(3):425–443, 1999.

[4] T.W. Malone and K. Crowston. The Interdisciplinary Study of Coordination. *ACM Computing Surveys*, 26(1):87–119, 1994.

[5] E. Michael Maximilien and Munindar P. Singh. Conceptual Model of Service Reputation. *Special Section on Semantic Web and Data Management, SIGMOD Record*, 4, December 2002.

[6] Paul Resnick, Ko Kuwabara, Richard Zeckhauser, and Eric Friedman. Reputation Systems. *Communications of the ACM*, 43(12):45–48, December 2000.

[7] Robert Tolksdorf. Models of Coordination. In Andrea Omicini, Robert Tolksdorf, and Franco Zambonelli, editors, *Engineering Societies in the Agent World First International Workshop, ESAW 2000, Berlin, Germany, August 21, 2000*, number LNAI 1972, pages 78–92. Springer Verlag, 2000.

[8] Robert Tolksdorf. A Dependency Markup Language for Web Services. In *Proceedings of net.Objectdays*, LNCS. Springer, 2002. to appear.

[9] George M. Wyner and Jitae Lee. Applying Specialization to Process Models. In *Conference on Organizational Computing Systems*, pages 290–301, 1995.