In: Proceedings of the 3rd International Conference on Semantic Technologies (I-Semantics 07), Graz, Austria, 2007

User-Driven Semantic Wiki-based Business Service Description

Heiko Paoli, Andreas Schmidt, Peter C. Lockemann
(FZI - Research Center for Information Technologies, Karlsruhe, GERMANY
{Heiko.Paoli, Andreas.Schmidt, Peter.Lockemann}@fzi.de)

Abstract: A key factor for success of companies operating in a globalized market environment is a modern SOA-based infrastructure. An essential component of a SOA infrastructure is the central service registry. Current standards for organizing service registries and their implementations are driven by the technical aspects of the infrastructure. When using such technically organized service registries, business users often fail to find the needed information. With the concepts of Web 2.0 in mind, we present a new approach to the organization and implementation of the business registries that are driven by the needs of business users. The paper discusses the problems of the current technically driven approaches, presents an architecture for a business user-driven service registry and introduces an implementation of the architecture using UDDI and Semantic MediaWiki.

Key Words: Service Oriented Architecture, Service Registry, UDDI, Web 2.0, Semantic MediaWiki, Ontology Engineering, Work Integration, Collaboration
Category: H.1.m, H.3.3, H.3.5, H.5.3, I.2.4, K.4.3, K.6.4

1 Introduction

A key factor for the success of companies operating in a global market environment is a flexible communication and information infrastructure that can be quickly and easily adapted to changing needs. Lately, service orientation has evolved as one of the more promising concepts for providing this flexibility. Information infrastructures that follow the paradigm of Service-Oriented Architecture (SOA) allow information processes to be defined conveniently and with minimal effort as a succession of calls on available services [Cearley et al. 2005, He 2003].

Judging from the many trade journals, service orientation does not yet live up to these expectations. We claim as our thesis that the failure is due to service descriptions that are of little help to the business users. Current descriptions have been written by service developers and just cover technical aspects such as service interface, formal parameters, or supported protocols. But this is not the world of the business users who initiate and control the business processes and react to numerous events in them. They need to know which services are available for which business purpose, which services have to be replaced when a business process has to be changed or whether new services are needed in order to adapt to new requirements [Huhns and Singh 2005].

As part of the solution we propose differentiating between different stakeholders. The design of information processes should be the responsibility of personnel that understands both, information systems and the business processes (we refer to them as *business analysts*). They need to know what the services have to offer to those executing the business (the *business users*). How these services have been technically implemented should be of little concern to them. The implementation of the services, and their connection to information processes, is the domain of *service developers*.

Service registries should address all stakeholders. Current service descriptions, though, concentrate on the service developers. To include the business aspects would be the task of the business analysts. The objective of this paper is to discuss how the analysts can be supported effectively to carry out this task. Any solution should keep in mind that in an environment subject to frequent change, service description cannot be a one-time affair but rather a continuous and collaborative effort among business analysts and service developers [Stojanovic and Dahanayake 2005].

Web 2.0 seems to be an appropriate interaction paradigm in which all stakeholders can be given an active part in service description. This paper presents a new collaborative and lightweight approach to describing services, and shows how business users can take an active part in it, so that a service registry would be able to cover their needs as well.

2 Problem Analysis

As discussed before, service discovery has technical and business ("semantic") facets. The technical part of a service description has always been formulated in a way to make algorithmic processing possible. For the purpose of computer-assisted service discovery the same should hold for the semantic part. Consequently, the business analyst must build a formal model of his or her conceptualization of the business domain, and relate the services to this model.

Technical descriptions specify how services can, and must be used within a computational environment. Consequently, technical descriptions should only concern the service developers. Likewise, semantic descriptions should be solely of interest to the business analysts and users. Moreover, being an abstraction the same service implementation may be applicable in different business situations and, hence, may have more than one semantic description. Consequently, both for technical and application reasons the technical and business aspects of the service description should be kept separate, something that has been known in software engineering as *separation of concerns*[1].

[1] Progr. for Separation of Concerns, `http://www.dmi.unict.it/~tramonta/PSC07/`

Modern business is not a static affair. Continuous change to the business descriptions in the registry is, therefore, a constant challenge. In today's interlinked world the flexibility of SOA should be complemented by a flexible approach where the organization of the business registry should be turned into a collaborative and continuous task along the lines of, say, the Web 2.0 concept.

To summarize, a business-oriented service registry should meet three requirements:

R1 Capture the semantics of business aspects to make services more accessible to business analysts

R2 Keep technical and business aspects of the service description separate for optimal support of the different user groups

R3 Support the collaborative and dynamic evolution of the service description to accommodate changing needs

3 A Comprehensive Approach to Business Service Description

3.1 Basic architecture and workflow

UDDI is practically the only standard for publishing technical descriptions, well suited for this purpose, but lacking the capability to describe business aspects appropriately. To meet requirement R2 Uddi is used a basis. To fulfill requirement R3, a Semantic MediaWiki (SMW) is taken as a collaborative front end of the business registry. Finally, to satisfy requirement R1 we employ ontologies to capture the network of related terms. In particular, our aim is a lightweight ontology that can be easily handled by business experts without extensive training in ontology engineering.

Figure 1 shows the system architecture. It consists of four main components: a UDDI-based technical registry, a SMW-based business registry, an ontology server and an ontology engineering component. The figure also indicates the basic workflow within the architecture. A software developer can use any UDDI-compatible client to publish a technical description of a new service. The developer may add some keywords based on the ontology which are used as an initial categorization for the service. The content of the UDDI Registry is dynamically embedded into the content of the SMW. From now on business users can search or navigate along the content of the SMW and provide additional business information. A SMW is chosen to make the content machine-understandable and to add implicit facts with the help of an ontology server. The ontology engineering component allows the business users to adapt the used business ontology to their needs in a lightweight and collaborative way.

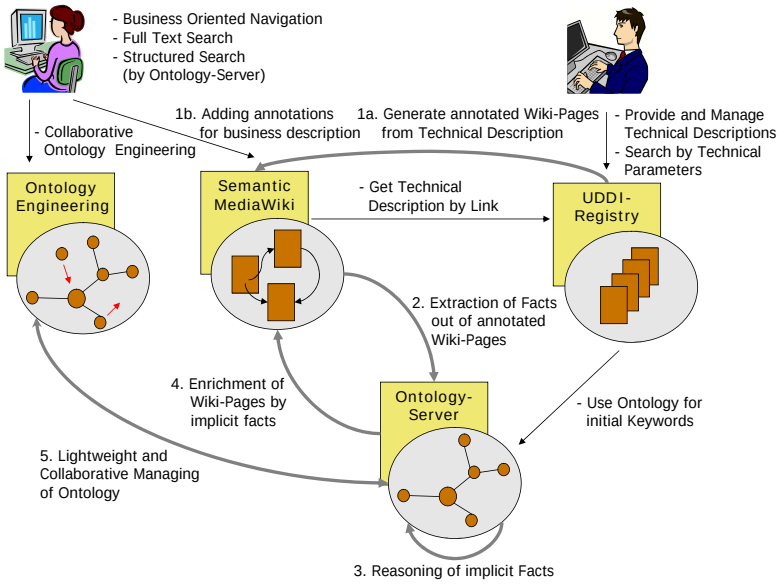

Figure 1: Combining UDDI with a Semantic MediaWiki

3.2 Ontology and Lightweight Ontology Engineering

The left-hand side of Figure 2 gives an example of the organization of our ontology. The top level part provides the domain-independent concepts such as the terms *Concept*, *Business Object* and *Service*. These are refined to a network of concepts of the business domain. Figure 2 shows just three examples of such concepts: the terms *water*, *water level* and *water gage information*. In a collaborative environment the presentation of ontologies is particularly important for effective and efficient use by the business analysts. The business registry is presented in the form of Wiki pages, with relations between concepts mapped to semantic links (right-hand side of Figure 2). In the example, *water* is a top level (business) concept while *water gage* is a business object concept, and *water gage information* is of type *service* and stands for a published service in the SOA infrastructure which will return a *water gage*. We use OWL-Lite as the ontology description language. Currently we use KAON2 as a reasoner, but any other compatible reasoner is also possible [Motik and Sattler 2006].

In the dynamic business environment the ontology itself is also bound to change frequently (see Section 2). Rather than entrusting a central authority with modifying the ontology we rely on the combined and distributed compe-

tency of all business analysts. Accordingly, we let the ontology evolve in collaboration of the business experts whenever one sees the need. Since we cannot expect the analysts to be experts in building ontologies, the engineering of the ontology should be made as simple as possible. We ease the task in two ways. For one the ontology is visualized as a graph, and all modifications can be easily done by dragging and dropping the nodes of the visual presentation rather than in some formal language. Second, the range of possible modifications is restricted (hence the name "lightweight engineering"), e.g., concepts can only be connected via *broader-narrower* and *related* relations. By using Wiki pages all modifications become immediately visible to other analysts.

3.3 Annotation of Wiki pages and service discovery

A service such as *water gage information* is initially entered into the system by its developer, who publishes it to the UDDI registry and is encouraged to augment it by intuitive keywords found in the ontology. After the publication a Wiki page is generated for the service, and automatically annotated with the keywords as well as semantic links that are obtained from the relations of the general UDDI data model. Subsequently, the business analyst may carry out the annotation of Wiki pages by means of such SMW features as semantic links, semantic attributes, and inline queries (to embed dynamic content). Many annotations can be obtained from the ontology by navigating through it and extracting further facts, or by using the reasoner to derive implicit facts. For example, on the left-hand side of Figure 2 the solid arrows represent relations that are explicitly available from the ontology (*hasType*, *belongsTo*, *provides*), while the dashed arrows represent relations that are implicitly available because of reasoning through the ontology server. Not only does our approach satisfy requirements R2 and R3, but it clearly does so with great benefit to both, business analyst and service developer. A business analyst can concentrate on the business description and organize and annotate the Wiki pages freely. For example he or she may express the business context of a service, e.g., business use cases, business value etc. The business description is limited neither by the data model of UDDI nor the facilities of WSDL. On the other hand the UDDI registry remains compatible to current SOA implementations.

We do not foresee automatic service discovery. This explains the emphasis we give to the presentation via Wiki pages. Take again the right-hand side of Figure 2. Note that much of the page contents for all terms is automatically generated. In particular, business object pages list all relevant services. Consequently, our approach satisfies requirement R1 as well. The proposed organization of the business registry and the use of a domain ontology well known to the business analysts and users provides a familiar and easy-to-use environment for them.

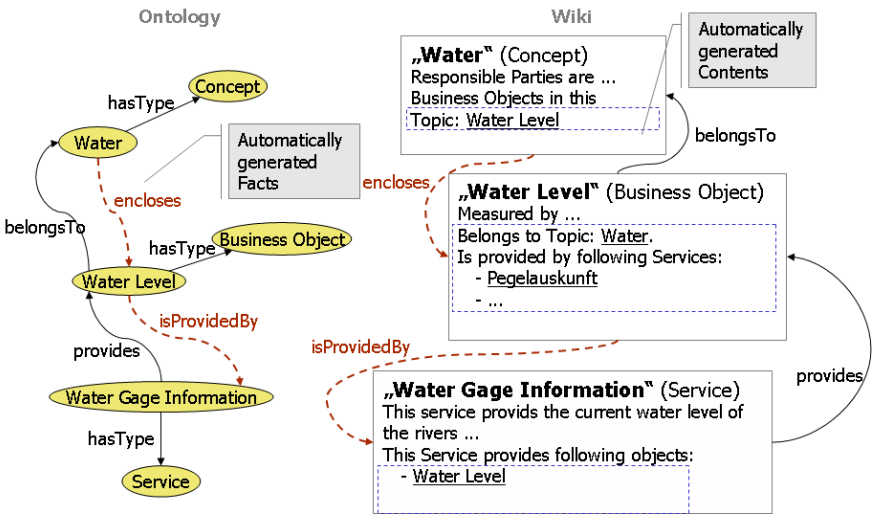

Figure 2: Organization and presentation of the business registry

4 Implementation

The implementation of our service registry consists of a central relational database, which holds the UDDI entries, the SMW pages and the ontology. On top of the relational database we have a J2EE application server and an HTTP server with PHP support. The J2EE application server represents the technical UDDI-compatible registry, which is realized through a standard UDDI framework. This way our implementation is fully compatible to the standard and explicitly allows publishers to use their own UDDI browser if they wish to. The HTTP Server with PHP support represents the business-oriented registry realized through an extended SMW component – the extension is necessary to support the automatic generation of content from the UDDI registry. For ontology engineering we use the existing tool SOBOLEO, a Web-based implementation of a Simple Knowledge Organisation System [Zacharias and Braun 2007].

5 Related Work

Automated discovery of Web services in a UDDI environment using WSDL descriptions is the subject in [Sivashanmugam et al. 2003]. The authors propose the use of an ontology to annotate WSDL message parts in order to add the necessary semantics. The semantics is entirely embedded in WSDL and thus cannot be separated from the technical description. While this may appear acceptable for automatic discovery, it burdens the business analyst, not the least because

the semantics is expressed in notations unnatural to the business user. The approach thus fails to meet requirement R2. A bit earlier, [Paolucci et al. 2002] took a similar approach. The authors present in greater detail an algorithm for matching service requests to advertised services based on semantic descriptions. From the point of view of R2 their approach seems somewhat more advanced, since the DAML-S semantics is kept on a semantic layer. A DAML-S/UDDI translator is used to connect the semantic layer to the UDDI registry. But it seems doubtful that business users would feel comfortable with the semantic description or would consider the approach transparent enough to evaluate the outcome of their search.

Consequently, separation of technical and business concerns has not been a pressing issue in the past. [Bergmans et al. 2001] introduce a general model for examining whether and when it makes sense to compose systems from multiple concerns. The authors define a category of composability problems inherent in given composition models and provide criteria for using the separation of concerns paradigm. We conclude that our approach does not fall into the category of composition anomalies so that requirement R2 is indeed justified.

Wiki as a standard for collaborative authoring has been proposed in the past. The authors of [Krötzsch et al. 2006] extend the Wiki concept so that the content of a Wiki becomes machine-processable, and provides an embedded query language. Citing a number of reasons, they seem to confirm that a SMW is ideal as a front end for business analysts. More specifically, for the collaborative task of continued engineering of lightweight ontologies the SOBOLEO system employs a tagging mechanisms: interesting information is shared within a community and tagged by the latter to categorize it [Zacharias and Braun 2007]. Concepts of a lightweight ontology can then be derived from the used tags. The ontology is constructed and changed in a collaborative and Web 2.0-like way.

6 Experiences and Conclusions

The work presented in this paper has its origin in a project that was financed by the Ministry of Environment of Baden-Wuerttemberg. The environmental administration of Baden-Wuerttemberg has a long experience with environmental information systems in service oriented architectures. At the moment a redesign to a modern SOA-based infrastructure is planned by the State Institute for Environment, Measurements and Nature Conservation on behalf of the Ministry of Environment. The main objective is to provide all relevant parts of the system as services by a registry, and it should be possible to add a wide though unknown range of the services in the future. The system should be capable of handling hundreds of business users and service developers. To avoid duplicate work and to make all published services transparent to all business users a business oriented service registry seemed essential. The initial ontology we have used is based

on an already existing and widely used taxonomy developed for the environmental information system of Baden-Wuerttemberg. The technical infrastructure as described above was developed in close communication with more than 10 representatives of business analysts and 5 representatives of developers, and was rolled out for a first testing period in April of 2007. First feedback by users sounds encouraging.

The thesis underlying our work is that service orientation will become widespread only if services can be discovered and employed with ease not just by service developers but also by business analysts. We have translated the needs to three requirements, the separation of technical and semantic descriptions, natural use of the semantic descriptions by business people, and a collaborative approach to dealing with the business dynamics. First experiences seem to support our thesis for the narrow scope of environmental information systems. What is definitely needed is more systematic and wider ranging empirical studies before we can be sure that our approach is an important step in overcoming the still existing doubts on the effectiveness of service-oriented architectures.

References

[Bergmans et al. 2001] Bergmans, L., Tekinerdogan, B., Glandrup, M., Aksit, M.: "Composing Software from Multiple Concerns: Composability and Composition Anomalies"; in Proc. of Int. Conf. on Softw. Engineering 2001, Toronto, Can., 2001

[Cearley et al. 2005] Cearley, D., Fenn, J., Plummer, D.: "Gartner's Positions on the Five Hottest IT Topics and Trends in 2005"; Gartner Web Site, 2005. `http://www.gartner.com/DisplayDocument?doc_cd=125868` as seen on 2006-09-20

[He 2003] He, H.: "What Is Service-Oriented Architecture"; O'Reilly, 2003. `http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html` as seen on 2006-09-20

[Huhns and Singh 2005] Huhns, M., Singh, M.: "Service-Oriented Computing: Key Concepts and Principles"; in: IEEE Internet Comp., vol. 09, no. 1, pp. 75-81, 2005.

[Krötzsch et al. 2006] Krötzsch, M., Vrandecic, D., Völkel, M.: "Semantic MediaWiki"; In Proc. of ISWC, Lecture Notes in Comp. Sc., vol. 4273, pp. 935-942, Springer, 2006.

[Motik and Sattler 2006] Motik, B., Sattler, U.:"A Comparison of Reasoning Techniques for Querying Large Description Logic ABoxes"; In Proc. of the 13th Int. Conf. on Logic for Progr. AI and Reas. (LPAR 2006), Phnom Penh, Cambodia, 2006

[Nickull et al. 2006] Nickull, D., McCabe, F., MacKenzi, M.: "SOA Reference Model TC"; OASIS Web Site, `http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm`, as seen on 2006-09-20.

[Paolucci et al. 2002] Paolucci, M., Kawmura, T., Payne, T., Sycara, K.: "Semantic Matching of Web Services Capabilities"; In First Int. Semantic Web Conf., 2002.

[Sivashanmugam et al. 2003] Sivashanmugam, K., Verma, K., Sheth, A., Miller, J.: "Adding semantics to web services standards"; In Proc. of the 1st ICWS, 2003.

[Shen 2004] Shen, Z.: "UDDI v3.0 (Universal Description, Discovery and Integration)"; OASIS Web Site, 2004. `http://www.oasis-open.org/committees/uddi-spec/doc/spec/v3/uddi-v3.0.2-20041019.pdf`

[Stojanovic and Dahanayake 2005] Stojanovic, Z., Dahanayake, A.: "Service-Oriented Software System Engineering, Challenges and Practices"; Idea Group Publish., 2005.

[Zacharias and Braun 2007] Zacharias, V., Braun, S.: "SOBOLEO - Social Bookmarking and Lighweight Engineering of Ontologies"; in Proc. of WWW '07, Workshop on Soc. and Collabor. Constr. of Struct. Knowl. (CKC), Banff, Canada, 2007.