An Ontology for Quality-Aware Service Discovery

Steffen Bleul and Thomas Weise

Kassel University, Distributed Systems
{*bleul, weise*}*@vs.uni-kassel.de*

Abstract. The fast emergence and acceptance of service oriented architectures leads to fast development of extensional technologies like service delivery, discovery and composition. As main effort is being spent on automatic discovery and composition, current solutions do not reflect real world scenarios sufficiently. Services are offered by different vendors with different quality levels and prices. Large service oriented architectures with dynamic service compositions are not able to adapt without manual inspection of service quality and negotiation of service contracts. We propose an ontology for modelling Quality of Services (QoS) and Service-Level-Agreements (SLA). A semantic approach should bridge the gap of different terminology, languages and metrics making Service-Level offers and requests agent understandable and automatic quality-aware discovery possible.

1 Introduction

Service oriented architectures (SOA) have become important means for implementing enterprise processes in the industry and on the web. Service oriented architectures are potentially easy to maintain, to extend and to improve. Services are building blocks in implementing business processes in companies and to integrate heterogeneous resources and external systems [BHMS05]. Therefore a SOA is a powerful software design paradigm.
Multiple services are necessary to realize a process which is usually represented by a workflow [RS04]. Companies implement processes like ordering, billing and accounting by composing several services where each service realizes a process task and services communicate with messages. Composition of services is supported by a process execution language like BPEL4WS [IBM03], and a message choreography language like WS-CDL [Wor04]. These languages allow automatic execution of service compositions because they allow the specification of workflows of services and message choreography between services.
Service oriented architectures are easy to evolve by replacing each service with a better substitute. Substitutes may offer less response time or traffic. Especially outsourced services offer flexibility in choosing software vendors or providers of specialised services, e.g. providing an improved search function for the internal infrastructure. Therefore it is beneficial to use service discovery mechanisms to adapt service compositions to changing quality and pricing.

Although automatic service discovery is already possible, e.g. with IOPE matching algorithms, like the one presented in [JT04]. However it currently does not take into account quality and pricing aspects of services. Using outsourced services also demands special actions like dealing out service level obligations, contract signing, accounting and quality monitoring. Based on current SLA (Service Level Agreement) description languages like WSLA [ea02] and WSML [SDM02] we believe a semantic approach can automate quality aware discovery of services. We propose an ontology for machine understandable service level requests and offers for quality-aware service discovery. As a result it enables dynamic binding of external services in an internal workflow under quality and pricing aspects. Therefore the ontology contributes to service brokering and dynamic service bindings in large-scale service oriented architectures and compositions.
The remainder of this paper is structured as follows. The next section is an introduction to quality-aware service discovery and presents a use case. Section 3 is an overview of the proposed ontology and section 4 gives an example on matchmaking in quality-aware service discovery. Section 5 is about related work before the paper closes with a conclusion.

2 Quality-Aware Service Discovery

Quality-aware discovery of services comprises finding a service with the requested quality dimensions and checking if the service can offer the requested guarantees. Even if service descriptions offer specifications of service dimensions and guarantees, quality-aware service discovery remains cumbersome. The task is even harder if different parties use different terms and metrics for the same quality dimension. Furthermore, providers can offer different service levels at different prices and enhance certain quality levels by offering supplementary service packages. A service package is a union of obligations and its price. The user can combine service packages and supplementary packages to accomplish his requested service level.
A quality level consists of several obligations. Each obligation offers a guarantee on a quality dimension, e.g. a guarantee on response time with at most 200 milliseconds. Service provider and service user both negotiate a certain quality level by agreeing on obligations on quality dimensions. The outcome is a SLA with several obligated SLO (Service Level Objectives). Service Level Agreements are necessary for continuous service delivery and quality. The providing party must enforce these objectives or face negotiated violation penalties like contract cancellation or paying violation fees.
In Figure 1 we present an example for a service level offer on the left and service level request on the right side. The service level request consists of three guarantees on the quality dimensions response time, transactions and contract time. A service provider offers the necessary service for travel booking with guarantees on the requested dimensions but two kinds of service levels. The Gold-Guarantees are better than the Silver-Guarantees but also at a higher price. A supplementary package offers enlarging the contract time for one month at a price of five

Service Offer Travel Booking Service				Service Request Travel Booking Service	
Quality-Dimensions	Gold-Guarantees	Silver-Guarantees	Supplementary-Package	Quality-Dimensions	Requested-Guarantees
Responsetime	< 200ms	< 400ms		Responsetime	< 0,4 s
Transactions	10000	5000		Transactions	> 4000
Contract time	12 month	12 month	+1 month	Contract time	1 year
Price	50 €	20 €	5 €		

Fig. 1. Example scenario

Euros. As you can see, the offered guarantees on response time are measured in milliseconds but the requested is measured in seconds. So they have semantically the same dimensions but different metrics.

Existing service descriptions allow searching for syntactically matching services. Discovery of services is done manually by browsing search results for semantically matching services which is cumbersome in large registries. There is not only the need for syntactic description languages but also for semantic interface descriptions. Currently this is realized mostly by the WSML [SDM02] and OWL-S [ea04] ontologies. OWL-S is based on OWL (Ontology Web Language) the W3C proprietary language for the Semantic Web [MH04]. Both languages allow the semantic description of Web Service interfaces.

3 Service Level Ontology

As already mentioned, current service brokering approaches lack the ability to model service level packages and pricing. In the real world service brokering is much more flexible. Service providers offer service level packages, e.g. gold, silver or copper packages, with different pricings and service level objectives. A service level package can be optimized by buying supplementary packages, e.g. buying additional transactions or a better guarantee on response time.

Otherwise quality-aware service discovery is limited to matching obligations on quality dimensions against each other. This decreases the probability of successful service discovery. Our approach proposes a SL-Ontology (Service Level) for modelling service level packages, supplementary packages and pricing for service offers. It includes modelling of QoS, metrics and units. The following paragraphs give an overview of our SL-Ontology followed by an example on matchmaking in the next section.

The structure of our SL-Ontology can be illustrated as an ontology pyramid, which is presented in Figure 2. On top we distinguish between three individuals of the SL-Ontology. The *Service-Level-Offer* concept represents the service level provider and his offered service level packages. The *Service-Level-Request* concept represents a requested obligations for the wanted service level and finally

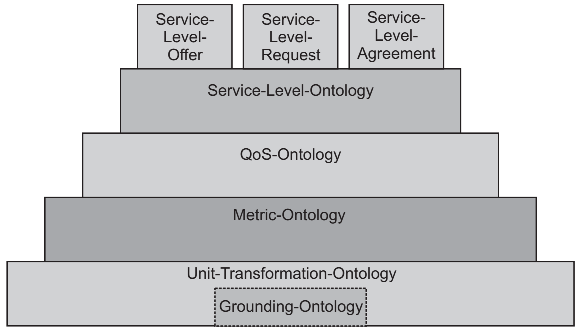

Fig. 2. Service-Level-Ontology Pyramid

the *Service-Level-Agreement* concept specifies the service level both parties agree on. While current service discovery does not distinguish between offers and requests, we believe that differentiation between them is necessary when handling multiple requests and offers in one knowledge base.

The *SL-Ontology* is build on top of a *QoS-Ontology*. The *QoS-Ontology* is used for modelling service quality dimensions. Since different providers can use different identifiers for the same quality dimension, the ontology allows binding of individuals to external taxonomies. Therefore we can distinguish between dimensions, which relate to semantically the same concept. Each quality dimension value is defined by a metric. The *Metric-Ontology* specifies the unit and data type the provider uses to measure his quality dimension. We have to know the data-type of the quality dimensions. Otherwise if we use a data type that is too limited, we will have inaccurateness by interpreting the quality dimension value. At the bottom of the ontology pyramid the *Unit-Transformation-Ontology* is the base ontology for semantic specification of service levels. Each metric has its unit. The most obvious problem in matching service dimensions is the use of different units for the same metric. Response time for example can be measured in milliseconds or seconds. Therefore the *Unit-Transformation-Ontology* allows specifying the functional relation between different units. The *Grounding-Ontology* is used for grounding semantic concepts on logical entities like rule descriptions for guarantees or service execution for unit transformation.

4 Quality-Aware Matchmaking

Let us look at matchmaking between a Service-Level-Offer and a Service-Level-Request. Matchmaking as part of a discovery process will show how our ontology enhances quality-aware service discovery. We start to illustrate how to specify individuals with our SL-Ontology and realize matchmaking between them. We use the example of section 2, Figure 1.

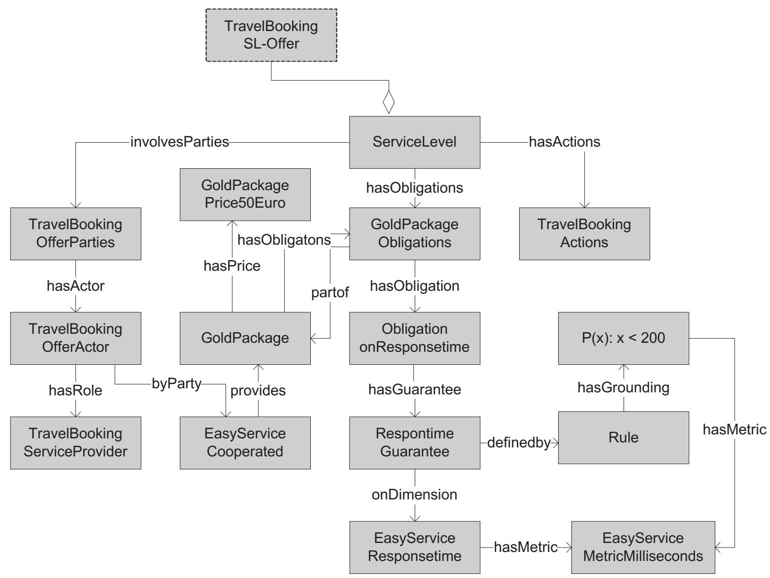

Fig. 3. Service-Level-Ontology Example

In Figure 3 we illustrate an individual of the SL-Ontology. The individual is a SL-Offer of the Travel Booking service. The service provider is the Easy Service Company which is specified by the *EasyServiceCooperated*-Individual. The service level obligations are part of the gold package. This is specified by the *partof*-Property between the *GoldPackage*-Individual and the *GoldPackageObligations*-Individual. The other packages are modelled analogously to this example. Our semantic matchmaking example of quality dimensions consists of three parts:

1. We have to decide if any offered package deals with guarantees on the same quality dimensions as the requested quality dimensions. Otherwise the package can not achieve at least one requested quality guarantee.
2. Check if two semantic equivalent quality dimensions use different metrics.
3. Check if the offered guarantee satisfies the requested guarantee.

As a matter of simplicity we limit the example of matching guarantees on response time and leave out the effects of supplementary packages. Actions for contract signing and contract cancellation are matters of service choreography or orchestration and therefore out of the scope of this paper. Obligations consist of a guarantee on one quality dimension. The guarantee consists of a *hasEffect*-Property which is connected to a quality-rule; in our example the rule guarantees a response time lower than 200. Each quality dimension is connected with a metric with the *hasMetric*-Property.
The example is continued in Figure 4. We are looking for semantic equivalent quality dimensions. In Figure 4 we present on the left side the specified quality

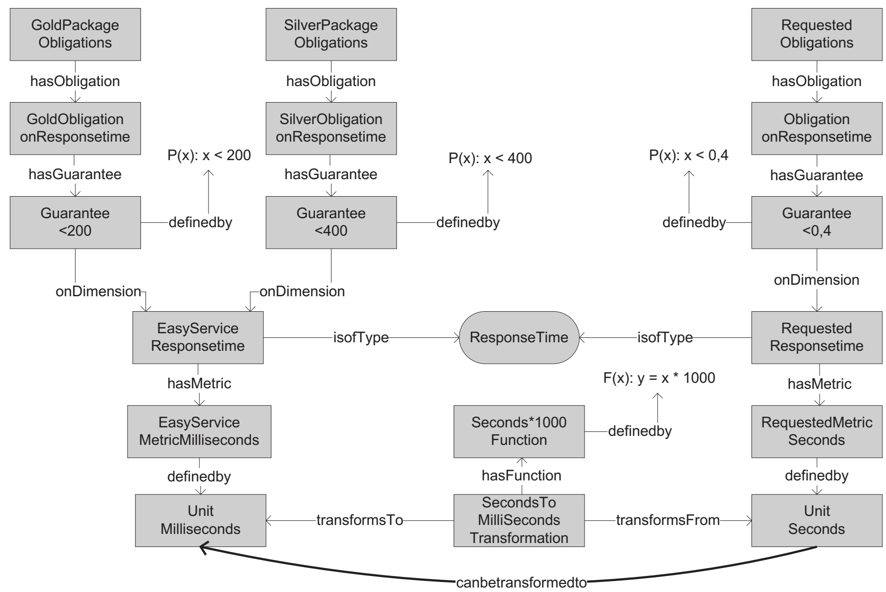

Fig. 4. Matchmaking of quality dimensions.

dimension *ResponseTime* of the gold and silver package and on the right side the requested quality dimension *RequestedResponsetime*. We have two quality dimension individuals of the same class but with different identifiers. To check if both parties mean the same quality dimension we use the *isofType*-Property. It is a binding-property to map to an external taxonomy which specifies the semantic relation of the quality dimensions. We see, that both specified dimensions are semantically related to the external specified individual *Responsetime* and therefore semantically equivalent.

Afterwards we check if both used metrics are semantically equivalent. Easy Service Company uses the individual *EasyServiceMetricMilliseconds* to measure its response time, which measure unit is milliseconds. Whereas the requested guarantee on response time is expressed in seconds, which is specified by the *definedby*-Property between the *RequestedMetricSeconds*- and the *UnitSeconds*-Individual.

At this point we use the Unit-Transformation-Ontology to specify the functional relation between units. Without this ontology we could only match quality dimension guarantees with the same measurement unit against each other, but we also want to match guarantees whose measurement units can be transformed into each other. If we have two *Unit*-Individuals, e.g. *UnitMilliseconds* and *UnitSeconds* which have a transformation between them, we specify a transitive *canbetranformedto*-property.

We can conclude that there is a stepwise transformation from seconds to milliseconds by multiplying seconds with 1000. Functions are expressed by function groundings, which can be realized by a rule or an execution of a service. The function grounding is used to adapt the guarantee rules to equal units, e.g. transforming the rule $P(x) : x < 0,4$ to $P(x) : x < 400$. Afterwards we can match the guarantees. As both packages contribute the necessary quality level, the requestor takes the packages with the lowest price which is the silver package.

5 Related Work

Some work dealt with specifying quality of service and service level agreements. WSML [SDM02] and WSLA [ea02] are specification languages for SLA. Both languages allow specification of quality dimensions, metrics and guarantees, but both approaches lack the usage of semantics and therefore cannot bridge the gap of using different terms and semantic related metrics.
In OWL-S [ea04] QoS is specified as service parameters but lacks the specification of metrics and guarantees. It also lacks the specification of functional relations between metrics and is therefore not applicable for quality-aware service discovery. WSMO [RKL04] has indirect support for QoS by using non-functional properties for quality dimensions and functional properties for relation between quality dimensions. Their approach lacks semantic definition of stepwise transformation between metrics and therefore you have to specify a transformation for each unit into another unit.
DAML-QoS [ZCL05] is an ontology for QoS-Specification. Like our approach DAML-QoS differentiates between QoS-Offers and QoS-Requests but does not support the specification of service-packages and pricing. It uses object-oriented identifiers to bridge different terminologies and metrics in quality dimensions. These identifiers are predefined metrics. Our approach has an underlying Unit-Transformation-Ontology to define functional relations between metrics.

6 Conclusion

This paper introduces an ontology for quality aware service discovery. We have presented the necessary elements of quality aware service discovery and importance of integrating quality aspects in service integration. A description language needs flexibility for service level packages and service providing parties. It must also handle different terms in specifying QoS-Dimensions. Therefore we propose our ontology for semantic modelling of Service-Levels.
This is followed by an example for matchmaking of Service-Level-Offers and Service-Level-Requests 4. Our ontology is able to express the necessary generality for the specification of Service-Levels, Service-Level-Offers and Service-Level-Requests. Additionally we are able to bridge the usage of different terms by using binding-properties to include external ontologies.
Furthermore we are able to specify transformations between metrics. A transitive property is used to specify semantically if there exists a stepwise transformation

between two different metrics.
Currently we are testing a prototype matchmaker for different examples. We are confident to realize an automatic service brokering system and extend our approach not only on simple services but for whole service compositions.

References

[BHMS05] R. Berbner, O. Heckmann, Andreas M., and Ralf S. Eine dienstgüte unterstützende webservice-architectur für flexible geschäftsprozesse. *Wirtschaftsinformatik*, 47(4):268–277, 2005.

[ea02] A. Dan et al. Web service level agreement (wsla) language specification. In *In documentation for Web Services Toolkit, version 3.2.1*. International Business Machines Corporation (IBM), August 2002.

[ea04] D. Martin et al. *OWL-S, OWL-based Web Service Ontology*, 2004. URL: http://www.daml.org/services/owl-s/1.1/.

[IBM03] IBM. *BPEL4WS, Business Process Execution Language for Web Services Version 1.1*, 2003. URL: http://www-128.ibm.com/developerworks/library/specification/ws-bpel/.

[JT04] M. C. Jaeger and S. Tang. Ranked matching for service descriptions using daml-s. In Janis Grundspenkis and Marite Kirikova, editors, *Proceedings of CAiSE'04 Workshops*, pages 217–228, Riga, Latvia, June 2004. Riga Technical University.

[MH04] E. Miller and J. Hendler. *Web Ontology Language (OWL)*. World Wide Web Consortium (W3C), 2004. URL: http://www.w3.org/2004/OWL/.

[RKL04] D. Roman, U. Keller, and H. Lausen. Wsmo - web service modeling ontology. In *DERI Working Draft 14*. Digital Enterprise Research Institute (DERI), February 2004.

[RS04] J. Rao and X. Su. A survey of automated web service composition methods. In *Semantic Web Services ancd Web Process Composition, 2004, Springer-Verlag, pp 43-54*. First International Workshop, SWSWPC 6. July 2004, San Diego, July 2004.

[SDM02] A. Sahai, A. Durante, and V. Machiraju. *Towards Automated SLA Management for Web Services*. Hewlett-Packard Laboratories Palo Alto, July 2002. URL: http://www.hpl.hp.com/techreports/2001/HPL-2001-310R1.pdf.

[Wor04] World Wide Web Consortium. *WS-CDL, Web Services Choreography Description Language*, December 2004. URL: http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/.

[ZCL05] C. Zhou, L.-T. Chia, and B.-S. Lee. Semantics in service discovery and qos measurement. *Wirtschaftsinformatik*, IT Pro(March/April):29–35, 2005.

```
@inproceedings{BW2005QASD,
  author       = {Steffen Bleul and Thomas Weise},
  title        = {An Ontology for Quality-Aware Service Discovery},
  booktitle    = {Engineering Service Compositions: First International Workshop, WESC05},
  series       = {IBM Research Report},
  editor       = {C. Zirpins and G. Ortiz and W. Lamerdorf and and W. Emmerich},
  volume       = {RC23821},
  publisher    = {Yorktown Heights: IBM Research Devision},
  year         = {2005},
  month        =  Dec,
  affiliation  = {University of Kassel},
  location     = {Vrije Universiteit Amsterdam, The Netherlands},
  note         = {Workshop Home\\
                  The work is online available at
                  http://www.it-weise.de/documents/index.html\#BW2005QASD.\\
                  The publication can be downloaded at
                  http://www.it-weise.de/documents/files/BW2005QASD.pdf.\\
                  The presentation can be downloaded at
                  http://www.it-weise.de/documents/files/BW2005QASD\_slides.pdf.\\
                  Contact Thomas Weise at tweise@gmx.de or http://www.it-weise.de/.},
  abstract     = {The fast emergence and acceptance of service oriented architectures
                  leads to fast development of extensional technologies like service
                  delivery, discovery and composition. As main effort is being spent on
                  automatic discovery and composition, current solutions do not reflect
                  real world scenarios sufficiently. Services are offered by different
                  vendors with different quality levels and prices. Large service oriented
                  architectures with dynamic service compositions are not able to adapt
                  without manual inspection of service quality and negotiation of service
                  contracts. We propose an ontology for modelling Quality of Services
                  (QoS) and Service- Level-Agreements (SLA). A semantic approach should
                  bridge the gap of different terminology, languages and metrics making
                  Service-Level offers and requests agent understandable and automatic
                  quality-aware discovery possible.},
  contents     = {* Introduction\\
                  * Quality-Aware Service Discovery\\
                  * Service Level Ontology\\
                  * Quality-Aware Matchmaking\\
                  * Related Work\\
                  * Conclusion},
  keywords     = {Web Service Discovery, Quality-Aware, QoS},
  language     = {en},
  url          = {http://www.it-weise.de/documents/index.html\#BW2005QASD}
}
```