

ON DESCRIPTION OF CONTRACTS AND AGREEMENTS IN THE CONTEXT OF SOA

Laima Paliulionienė

Vilnius University Institute of Mathematics and Informatics
laima.paliulioniene@mii.vu.lt

Abstract. The issue of making contracts between services in Service-Oriented Architecture (SOA) arises naturally, because in real life services are subjects for negotiation and contracting. The notion of contract in SOA is rather misleading, because usually means just technical specification – input and output parameters, pre- and postconditions. However, there are also other aspects of contract. Service Level Agreements (SLAs) often are used together with technical contract. SLAs describes the level of service expected by a customer from a supplier, laying out the metrics by which that service is measured, the remedies or penalties in case the agreed-upon levels are not achieved. The paper aim to clear up the notions of contract and agreement in SOA, discusses the properties of languages that are used to specify contacts and agreements, paying special attention to their legal aspects and comparing contract features in WS-Agreement ir WSLA.

Keywords: service-oriented architecture, SOA, contract, service-level agreement, SLA

Introduction

Contemporary information technologies, including SOA-based systems, grid and cloud computing, has substantially moved into the relationships between and inside business entities in the last years. E-services are substituting and supplementing ordinary services more and more. The issue of making contracts for these relationships arises naturally, because in real life services are subjects for negotiation and contracting. Instead of drawing up the contracts manually, new technologies seek to provide means to automatically match service providers and requesters.

The term “contract” has originated from law and means an agreement with legal obligation. However, there are various definitions of the contract in service-oriented architecture (SOA). Most often, the definition of the service contract achieves mostly technical rather than legal spirit. In case of Web Services, that are the most common implementation of SOA services, the contract usually means service description documents – WSDL definition, XSD schema, and policy. Beside such contracts, there are other types of agreements in SOA. The most popular is a Service Level Agreement (SLA) that outlines the level of the service using appropriate metrics, and the remedies or penalties in case the levels are not achieved. SLAs deal with quality of service (QoS).

Just as legal contracts are documents written in a language that both parties can understand, in SOA a contract language is used that must be understandable for both provider and consumer. Usually the languages for contracts and other agreements in SOA are XML-based.

The paper aims to clear up the notions of contract and agreement in the context of SOA and discuss the properties of languages that are used to specify contracts and agreements, paying special attention to their legal aspects.

The remainder of this work is organized as follows: in section 1, related work is discussed; section 2 presents a short overview of the notions of contract and agreement in law; section 3 considers different definitions of a contract in the context of SOA and languages that are used to write service contracts; section 4 deals with Service Level Agreements and other related agreements; in section 5, features of two leading SLA languages are presented and compared; section 6 deals with similarities and differences of contracts/agreements in SOA and cloud computing; section 7 outlines problems with legislation for SOA contracts; the last section contains conclusions of the work.

1. Related work

SOA is the natural evolution of previous developments models – object-oriented models of 1980s and component-based models of 1990s. Distributed software components remain the core of service-oriented architectures, but remotely invoking methods of objects were changed into passing messages between services. In “traditional” component-based development, contracts are understood as specification attached to interfaces. A four-level approach for contract awareness has been proposed in (Beugnard et al., 1999): basic or syntactic contracts, behavioral contracts, synchronization contracts, quality-of-service (QoS) contracts. *Basic contracts* specify static behaviour and determine the signature or the interface (the operations a component can perform, the input and output parameters, and possible exceptions raised during operation). *Behavioral contracts* describe invariants, pre- and post-conditions, as in the “design-by-contract” approach. *Synchronization contracts* specify synchronizations of method calls, describes sequence, parallelism and other dependencies. *Quality-of-service contracts* specify quantitative contractual issues, like maximum response delay, precision of a result, etc. Actually, these types of contract can also be defined in SOA – the first three types can be described by WSDL and appropriate WS-* specifications, while the latter in an SLA language. These contract types are ordered by increase of dynamic changeability and negotiability. Syntactic contracts are non-negotiable, while QoS contracts are usually dynamically negotiable.

A number of works deals with classification of contract types. (Tosic, Pagurek, 2005) classified Web Services contract types into three categories: functionality, quality, and infrastructure contracts. The authors examined Web Service languages to check what types of contracts can be specified by them and presented their results in the form of a decision table for WSDL, BPEL4WS, WS-CDL, WS-Policy, WSLA, WSOL, OWL-S.

(Okika, Ravn, 2008) has grouped SOA contract specification languages into three broad families: dealing with *Web services* (WSDL, WSOL, WSBPEL, WS-CDL, WS-Security, WSLA, WS-Policy, and WS-Trust), belonging to the *Semantic Web services* (OWL-S, WSMO, WSML), and concerned with *Electronic business* (ebXML framework). They also classified these languages by the aspects of web services they cover, namely aspects of interface, functionality, protocol, security and quality (here the aspects of quality are policy, trust, availability, performance).

The survey (Duan, 2012) presents a table of 27 overviewed publications on service contracts and indicates contributions and limitations of each. Unfortunately, the contributions and limitations are presented very briefly (mainly up to ten words for each publication), so it is difficult to form a view of the work being cited. For each related work, the author checks “yes” or “no” to note whether the paper deals with contract content dimensions like QoD (quality of data), QoS (quality of service), legal issue, context, and business term. Additionally, for each work, “yes” or “no” are checked for several contract management dimensions – description of contract, monitor/control, selection, matchmaking, and composition. As stated in the paper, 13 works touch legal issues, but they are mostly implicitly related, because law infrastructure and judicial system which would be dedicated to SOA and cloud computing are still not constructed. The author makes a conclusion, that if it is not guided, the result of the service contract among stakeholders does not represent the best choice from the economical view for the whole society. However, it is not clear what the author means, and why and how e-contracts should serve for the whole society. The author identified some directions for future research in very general form. One of them says that it is highly demanded that IT specialists, law specialists and government staff can work together to create relative judicial system.

The analysis of works mentioned above and a number of other papers has driven to the conclusion that legal aspects of contract law are not studied enough when referring to e-service contracts. One of reasons for this probably is rather misleading notion of contract in SOA that will be discussed in the next section.

2. Contract and agreement notions in law

As the terms “contract” and “agreement” has come to SOA from law, let us clear up shortly, what these terms mean and what is the difference between them.

The terms “contract” and “agreement” are sometimes used interchangeably in law. However, if we want to make a difference between them, then a contract is a stricter term. In the civil systems of law, we have the law of contract (not the law of agreement) as part of a general law of obligations. A *contract* is an agreement with legal obligation. For example, business agreements create legal obligation for the parties. On the contrary, there is no legal obligation for moral, religious or social agreements.

There are the following *elements of a contract* (Farlex, 2013): a) an offer; b) an acceptance of the offer; c) a promise to perform; d) a consideration (which can be a promise or payment in some form); e) a time or event when performance must be made; f) terms and conditions for performance; g) performance. Most of these elements can be found in Service Level Agreements (SLAs) used in SOA, as we will see in the next sections.

Bilateral contracts are the most common type of contracts. In these contracts, two parties are bound by their exchange of promises. Usually one promises to do or restrain from doing something, while the other promises to pay for this. *Unilateral contracts* are less common, but are a valid type of contracts. They contain promises that are made by only one party. The offeror (i.e., an entity who makes a proposal) promises to do a certain thing if the offeree performs a requested act. The performance means an acceptance of the offer, and the contract becomes executed.

3. Contracts in service-oriented architecture

Sharing service contracts is a cornerstone principle of SOA. This principle says that for services to interact, they need share only a formal contract that describes each service and defines the terms of information exchange (Erl, 2005). Service contracts support other principles of SOA, especially the principles of service abstraction, loose coupling, composability, and discoverability.

When talking about service contracts, usually web service contracts are meant. *Web services* are the most widely adopted implementation of a service-oriented architecture (though SOA implementation without web services is also possible). Web services are based on a set of protocols and standards by which the services can be published, discovered and used.

SOAP and WSDL-based web services are the most popular web services, though in the last years REST (or RESTful) web services are gaining popularity because of their simplicity (Erl et al., 2012). The Web Application Description Language (WADL) is the REST equivalent of Web Services Description Language (WSDL), but it is not officially standardized and not widely adopted. Though WSDL can also be used for RESTful Web Services, RESTful services usually go without formal description of their interface, or it is described merely in plain text or some programming language. So, a SOAP-based design is more preferable when a formal contract must be established to describe the interface that the web service offers, and when complex non-functional requirements must be addressed.

Let us compare the definitions of a contract in SOA of various authors – researchers, standardization organizations, and IT corporations.

1. According to (Erl, 2005), a service contract is comprised of service descriptions and *possibly* additional documents. There are three service description documents – WSDL definition, XSD schema, policy. They can be classified as service metadata and are unilateral and non-negotiable. Additional documents can include Service Level Agreements (SLAs). In his later book, Erl with co-authors (Erl et al., 2012) presents even a narrower definition: “a service is a software program that makes its functionality available via a published technical interface, called a service contract”.

2. According to the Technical Standard for Service-Oriented Architecture Ontology, that has been developed by the Open Group (Open Group, 2010), a service contract encompass both the interaction aspects and the legal agreement aspects. The legal agreement aspects include concepts such as Service-Level Agreements (SLAs). It is possible to split the interaction and legal aspects into two different service contracts.

3. According to the World Wide Web Consortium (W3C, 2004), the WSDL-based service description represents a contract governing the mechanics of interacting with a particular service, while the semantics (that expresses the effect of invoking the service) represents a contract governing the meaning and purpose of that interaction. The contract, that defines the semantic, may be explicit or implicit, machine-readable or human oriented, and it may be a legal agreement or an informal (non-legal) agreement. According to the W3C, if semantically rich languages are used to describe the mechanics of the interaction, then the essential information may migrate from the informal semantics to the service description, and more of the work can be automated.

4. According to OASIS (OASIS, 2006) contracts can cover a wide range of aspects of services: quality of service agreements, interface and choreography agreements and commercial agreements. Since a contract is the result of agreement by the parties, there is an agreement action associated with the contract. A contract may be arrived at by a mechanism that is not part of an SOA (an out of band process), or a contract may be arrived at during the course of a service interaction (an in-band process).

6. According to Oracle Corporation (Reynolds, Wright, 2010), the contract of a web service is mainly made up of technical components, namely WSDL Definition (it defines the operations of the service, input and output parameters of the operations, and the protocols (bindings) the service supports), XML Schema Definition (XSD; it defines the XML elements and types which constitute the input and output parameters), and WS-Policy Definition (an optional component that describes the service's constraints on security, quality of service, etc.). Additionally, the service contract may be supported by non-technical documents, for example, service-level agreements.

5. In Microsoft Windows Communication Foundation (WCF), all communication with a service occurs through the endpoints of the service (Microsoft, 2013). WCF (previously known as "Indigo") is a framework for building service-oriented applications in the .NET framework. A contract is written in WSDL and outlines what functionality the endpoint exposes to the client. It specifies what operations can be performed by the service, the form of the messages, the type of input parameters, and what type of processing or response message the client can expect. In most cases, the contract is defined by creating an interface in a programming language and using the appropriate class properties to generate WSDL file from the interface.

As we can see, there are two types of definitions of the contract:

1. In narrow sense definitions, WSDL is mentioned as a main contract language. As WSDL is an XML-based language for describing the functionality and interface of a web service, then the notion of contract emphasizes mostly technical aspects of interaction between services. Moreover, such contracts can be generated automatically from code describing an interface in some programming language. This is called the *code-first* approach, in contrast to the *contract-first* one. Though code-first approach is not considered to be good practice (as it leads to less robustly designed services), this approach is highly popular among service designers, because removes necessity for learning WSDL.

2. Broad sense definitions state that contracts encompass both technical aspects of services and non-technical documents, for example, service-level agreements. However, even broad sense definitions (perhaps excluding OASIS' definition) consider technical aspects as the major part of the contract.

Contracts in SOA, if they are understood as interfaces or other technical things, are unilateral. The offerer is the service provider, and through the contract it informs about its operations, input and output parameters, pre- and post-conditions. The offeree is the service requester. If the service requester accepts the terms of the contract offered by the offerer (i.e. provides appropriate input parameters and preconditions), the service provider performs the requested action (i.e., the appropriate operation with promised output parameters and post-conditions).

It should be noted that mostly technical understanding of contracts has its origin in Bertrand's Meyers's "design by contract" that was first introduced in 1986, in connection with the creating of the object-oriented programming language Eiffel. According to design-by-contract approach, software designers should define formal interface specifications, including preconditions, postconditions and invariants. These specifications are referred to as "contracts", in accordance with a metaphor of the conditions and obligations of contracts between humans or companies.

4. Service Level Agreement (SLA)

As we have seen above, a service-level agreement (SLA) may be considered as part of a service contract or as a stand-alone document. A service-level agreement describes the level of service expected by a customer from a supplier, laying out the metrics by which that service is measured, the remedies or penalties in case the agreed-upon levels are not achieved. SLAs describe quality of service (QoS). Examples of quality-of-service parameters include mean time between failures, mean time to repair, various data rates, throughput, etc.

Service level agreements are not the invention of SOA. SLAs have been applied since 1980s – actually before the Internet became widely used – by fixed line telecommunication companies as part of the contracts with their corporate clients (Blokdijs, 2008). SLAs are currently used widely between service providers and their customers, especially since outsourcing has become popular. Most service providers have standard SLAs, frequently with several levels of service at different prices, and these SLAs can be a subject for negotiation to be reviewed and modified.

SLAs are used not only for regulating service providing/consuming relationships, but they also allow one to select the service that best supports quality requirements. This is essential, when possibly several services can fulfil the functional requirements, but some of them can fail to meet important quality requirements.

There are also other contract types associated with SLA, namely Operational Level Agreements (OLAs) and Underpinning Contracts (UCs). The OLA is an internal agreement between the service provider and its internal teams in order to support SLA for a service consumer. It defines how various IT groups within a company should work, including the process and timeframe for delivery of their services, in order to underpin SLAs for their customers. The UC is an external agreement between the service provider and a third party (e.g., communication provider) to deliver agreed level of quality services or goods at specified time. The purpose of both OLA and UC is to ensure SLA for the service consumer. In the case of Underpinning Contracts, the term "contract" is legitimate, because it is a legally binding contract, and violation of agreed terms may result in legal action. So, UCs include both essential components of legally binding contract – an agreement and legal obligation.

One more type of agreements are business level agreements (BLAs) (Bratanis et al., 2010). The need for them arises because SLAs still concern technical characteristics of a service and their terms can be unclear for business specialists. For example, rather than promising 99% availability for a service, it would be more understandable to say that the

number of undelivered orders may not exceed two per month. BLAs are agreements of higher business goals and are created by business analysts. Their transformation to SLAs is one more challenge in the context of SOA.

5. SLA languages

WSLA and WS-Agreement are the most widely accepted approaches to define machine-readable SLA specifications for web services. Below, these approaches are described briefly, and a comparison is made to see some features of WS-Agreement and WSLA that are related to contract law.

5.1. WS-Agreement Specification

WS-Agreement Specification (Andrieux et al., 2007) is developed by the participants of the GRAAP WG (Grid Resource Allocation and Agreement Protocol Working Group) of the Compute Area of the OGF (Open Grid Forum). WS-Agreement authors are professionals from such information technology companies as IBM, HP, NEC, Univa Corporation, Platform Computing, and several universities. There are several tools implementing the WS-Agreement specification, and the most complete implementation is the WSAG4J framework implemented in Java.

The objective of the WS-Agreement specification is to define a language and a protocol for advertising the capabilities of service providers, creating agreements, and monitoring agreement compliance at runtime. To create an agreement, a pre-defined agreement template and agreement creation constraints are used. The agreement template contains customizable aspects that can be instantiated according to rules specified in the agreement creation constraints. The WS-Agreement specification is designed so that it would be composable with various negotiation models. A special WS-Agreement Negotiation Specification was also created later by the OGF (Waeldrich et al., 2011). WS-Agreement also standardizes the terminology, defining concepts “agreement”, “agreement offer”, “agreement acceptance”, “agreement terms”, etc. The concepts are rather similar to typical legal definitions.

According to WS-Agreement specification, an agreement defines a dynamically-established and dynamically-managed relationship between parties. It may specify both functional properties for identification or creation of the service and non-functional properties of the service such as performance or availability. Entities can dynamically establish and manage agreements via Web service interfaces. The specification defines Service Level Objectives (SLOs) as assertions representing the quality of service aspect of the agreement, such as response time, throughput, required number of CPU, etc.

In WS-Agreement, an agreement or an agreement creation offer has a part named “agreement context” that contains involved parties and various metadata such as the expiration time of the agreement or offer, and optionally, the name of the template from which the agreement is created. Two types of terms are defined in the specification – *service terms* that are refined as service description (functionality), service reference and service property terms, and optional *guarantee terms* that specify the service levels that the parties

are agreeing to and that can be monitored by management systems. The guarantees (both promises and penalties) are associated with the other terms in the agreement.

An expression of assurance in guarantee terms also includes *qualifying conditions on external factors* such as time of the day as well as the conditions that a service consumer must meet. An assurance also includes specification of *business values* associated with an SLO and meaning the importance of this assurance to the service consumer and/or to the service provider.

Violation of a guarantee term will incur a certain penalty, and meeting an objective can generate a reward (in money or other unit).

5.2. WSLA Framework

Web Service Level Agreement (WSLA) framework (Ludwig et al., 2003; Keller, Ludwig, 2003) was proposed by the IBM Corporation in 2001, several years before the OGF proposed WS-Agreement. It is also XML-based. The WSLA specification language was used in some experimental systems within IBM. However, it has not been widely adopted, but rather outrivaled by WS-Agreement. IBM representatives also participated in creating WS-Agreement, and core concepts of WSLA were brought into WS-Agreement. Nevertheless, WSLA is still available.

Table 1 presents a comparison of some features of WS-Agreement and WSLA that are related to contract law. As we can see, while the two specifications have rather similar facilities to express the agreements, WS-Agreement has the advantage over WSLA in some positions (e.g., in describing penalties and agreement lifecycle). The single noticeable WSLA advantage is in the possibility to describe a larger number of parties.

6. Contracts in cloud computing

Cloud computing and SOA have important overlapping concerns and common considerations (Raines, 2009). Both cloud computing and SOA share concepts of service orientation. Both of them require a robust network to connect providers and consumers, and therefore, both have the same weakness when the network goes down.

Both cloud computing and SOA require contractual relationships and trust between service providers and service consumers. However, their contracts often are of different nature. Managing contracts in SOA relies heavily on automated tools, the contracts are offered and accepted by software entities. On the contrary, contracts in cloud computing are typically plain-text documents, sometimes published online, they are usually made between legal persons (people or companies).

Like in case of SOA, cloud computing providers usually have a standard form of contract. It is typically written to favour the provider's company, so a consumer needs contract negotiation skills to establish the terms. While there are multiple variations of cloud computing – platform as a service (PaaS), storage as a service (STaaS), Software as a service (SaaS), component as a service (CaaS), infrastructure as a service (IaaS) – the contract issues are similar for all of them and encompass service level agreements, data processing and storage, infrastructure/security, and vendor relationship (Trappler, 2010).

Table 1. Comparison of WS-Agreement and WSLA

| | WS-Agreement | WSLA |
|--|---|--|
| Description of the parties | Only two parties may enter into agreement – the Agreement Initiator and Agreement Responder, and one of the two is denoted as the Service Provider. | Two or more parties may enter into agreement – provider, customer, and third (supporting) parties. |
| Agreement terms with measurable properties | Included. | Included. |
| Logic expressions of obligations | Included. | Included. |
| Action guarantee and action invocations | Not included. | Expresses a commitment to perform a particular activity if a given precondition is met. |
| Penalties and rewards | Included. | Not included directly, but in the case of a guarantee violation, an action is invoked. |
| Representation of an agreement template | Included. | Beyond the scope of this specification document. |
| Agreement states | Agreements and Terms have a runtime state that can be monitored. There the following agreement states in the specification: Pending, Observed, Rejected, Complete, Terminated, PendingAndTerminating, ObservedAndTerminating. | Not specified. |
| Negotiation | Defining a protocol for negotiating agreements is outside the scope of the specification. Separate WS-Agreement Negotiation Specification is created in 2011. | When the specification was proposed, a workgroup under OASIS had been working on defining a generic negotiation protocol for negotiating any electronic document. In 2002, OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee presented the specification (OASIS, 2002). |

Cloud computing and SOA share many features, though they are complementary activities that can be implemented independently or concurrently. SOA-based software can be created without using cloud computing. On the other hand, using cloud computing to outsource IT functions does not inherently mean that SOA standards are applied. However, cloud computing's platform and storage services can provide valuable support for SOA efforts. In this case, their contracts also need to be coordinated.

7. SOA contracts and legislation

Law infrastructure and judicial system which are dedicated to SOA and cloud computing are still undergoing construction process (Duan, 2012). However, enterprise SOA systems should work with compliance to other regulations, for example, accounting standards, which require the accuracy of financial information, or regulations for the secure handling of sensitive information. Measures to ensure this compliance need also to be provided in service contract and agreements.

Business-to-business e-contracting is related to SOA contracting to some extent, and aims at automating the contracting process between companies. Like contracts in SOA, e-contracts are to be prepared and signed automatically. The goal of e-contracting is to exclude paperwork and use only electronic communication. However, such e-contracts are often prepared for people and are not machine-readable. The appropriate legislation (especially the legislation for an electronic signature) has been enacted in most countries to underpin the creating of enforceable electronic contracts.

Conclusions

The analysis of related literature has showed that the term “contract” in the context of SOA is rather misleading. Though there are different definitions of a contract in SOA, this concept has mostly technical aspect. It is inherited from design-by-contract programming approach. It has been trying to go away from this approach in SOA, and some elements of legal contract are tending to be added.

More elements of a legal contract can be found in service level agreements (SLA). SLAs are used not only for regulating service providing/consuming relationships, but they also allow one to select the service that best supports quality requirements.

The most popular languages for the defining of machine-readable SLAs are WS-Agreement and WSLA. A comparison of their features has shown that these two specifications have rather similar facilities to express agreements, however WS-Agreement has the advantage over WSLA in some positions (e.g., in describing penalties and agreement lifecycle). The WSLA advantage is the possibility to describe a larger number of parties.

Since the subject of this paper is related with notions of contracts and agreements and their legal aspects in SOA, many contract and agreement issues are beyond the scope of this paper, though they are also important for better understanding of contracts and agreement in SOA. We can mention, for example, policies, that describe requirements that a service must follow and that may be a basis for drawing up a contract. Another issue is versioning of contracts. The automating of negotiation is also a broad area for research. Business level agreements (BLAs) transformation to SLAs is one more challenge in the context of SOA. Another important problem is related to composing of contracts in service compositions. All these things are important for elaborating contracts and agreements in SOA, including their legal aspects.

Dedicated legislation is not created so far for SOA-based systems, therefore it is difficult to speak about legal obligation. However, enterprise SOA systems should work with

compliance to other regulations, for example, accounting standards or regulations for the secure handling of sensitive information. Measures to ensure this compliance need also to be provided in service contract and agreements.

Acknowledgments

This work has been supported by the project „Theoretical and engineering aspects of e-service technology development and application in high-performance computing platforms“ (No. VP1-3.1-ŠMM-08-K-01-010) funded by the European Social Fund.

References

- Andrieux, Alain; Czajkowski, Karl; Dan, Asit; Keahey, Kate; Ludwig, Heiko; Nakata, Toshiyuki; Pruyne, Jim; Rofrano, John; Tuecke, Steve; Xu, Ming (2007). Web Service Agreement Specification (WS-Agreement). Open Grid Forum. [Accessed 20 May 2013].
Available from: <http://www.ogf.org/documents/GFD.107.pdf>.
- Beugnard, Antoine; Jezequel, Jean-Marc; Plouzeau, Noël; Watkins, Damien (1999). “Making components contract aware,” *Computer*, 32(7): 38–45.
- Blokdijs, Gerard (2008). *Service Level Agreement SLA: 100 Success Secrets: Covering SLA, Service Level Agreements, Service Level Management Topics and Much More*. Emereo Publishing. 176 p. ISBN 0980471613.
- Bratanis, Konstantinos; Dranidis, Dimitris; Simons, Anthony J. H. (2010). “Towards run-time monitoring of web services conformance to business-level agreements,” in: TAIC PART’10 Proceedings of the 5th international academic and industrial conference on Testing - practice and research techniques. Springer-Verlag, 203–206.
- Duan, Yucong (2012). “A Survey on Service Contract,” in: 2012 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2012). IEEE CS Press, 805–810.
- Erl, Thomas (2005). *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*. Prentice Hall. 792 p. ISBN 0-13-185858-0.
- Erl, Thomas; Carlyle, Benjamin; Pautasso, Cesare; Balasubramanian, Raj (2012). *SOA with REST: Principles, Patterns & Constraints for Building Enterprise Solutions with REST*. Upper Saddle River, New Jersey: The Prentice Hall Service Technology Series from Thomas Erl. 577 p. ISBN 0137012519 (978-0-13-701251-0).
- Farlex (2013). “Contract,” in: *The Free Dictionary by Farlex*. [Accessed 20 May 2013].
Available from: <http://legal-dictionary.thefreedictionary.com/contract>.
- Keller, Alexander; Ludwig, Heiko (2003). “The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services,” *Journal of Network and Systems Management*, 11(1): 57–81.
- Ludwig, Heiko; Keller, Alexander; Dan, Asit; King, Richard P; Franck, Richard (2003). *Web Service Level Agreement (WSLA) Language Specification*. IBM Corporation. [Accessed 20 May 2013].
Available from: <http://www.research.ibm.com/people/a/akeller/Data/WSLASpecV1-20030128.pdf>.
- Microsoft (2013). “Endpoints: Addresses, Bindings, and Contracts,” in: *Microsoft Developer Network (MSDN)*. [Accessed 20 May 2013].
Available from: <http://msdn.microsoft.com/en-us/library/ms733107.aspx>.
- OASIS (2002). *Collaboration-Protocol Profile and Agreement Specification*. OASIS. [Accessed 20 May 2013].

- Available from: <https://www.oasis-open.org/committees/download.php/204/ebcpp-2.0.pdf>.
- OASIS (2006). Reference Model for Service Oriented Architecture 1.0. OASIS. [Accessed 20 May 2013].
- Available from: <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html>.
- Okika, Joseph C.; Ravn, Anders P. (2008). "Classification of SOA Contract Specification Languages," in: Proceedings of the IEEE International Conference on Web Services ICWS 2008. IEEE Computer Society Press, 433–440.
- Open Group (2010). "SOA Ontology Technical Standard: Service, ServiceContract, and ServiceInterface," in: The Open Group SOA Source Book. [Accessed 21 May 2013].
- Available from: <http://www.opengroup.org/soa/source-book/ontology/service.htm>.
- Raines, Geoffrey (2009). Cloud Computing and SOA.
- Available from: http://www.mitre.org/work/tech_papers/tech_papers_09/09_0743/09_0743.pdf.
- Reynolds, Antony; Wright, Matt (2010). Oracle SOA Suite 11g R1 Developer's Guide (Google eBook). Packt Publishing Ltd. 720 p. ISBN 1849680191.
- Tosic, V; Pagurek, B (2005). "On comprehensive contractual descriptions of Web services," in: 2005 IEEE International Conference on eTechnology eCommerce and eService. IEEE, 444–449.
- Trappler, Thomas (2010). "If It's in the Cloud, Get It on Paper: Cloud Computing Contract Issues," in: EDUCAUSE Quarterly. [Accessed 20 May 2013].
- Available from: <http://www.educause.edu/ero/article/if-its-cloud-get-it-paper-cloud-computing-contract-issues>
- W3C (2004). Web Service Architecture. W3C Working Group Note 11 February 2004. [Accessed 20 May 2013].
- Available from: <http://www.w3.org/TR/ws-arch/>.
- Waeldrich, Oliver; Battré, Dominic; Brazier, Francis; Clark, Cassidy; Oey, Michel; Papaspyrou, Alexander; Wieder, Philipp; Ziegler, Wolfgang (2011). "WS-Agreement Negotiation Version 1.0," in: Open Grid Forum. [Accessed 20 May 2013].
- Available from: http://www.gridforum.org/Public_Comment_Docs/Documents/2011-03/WS-Agreement-Negotiation+v1.0.pdf.

L. Paliulionienė is an assistant researcher in the Institute of Mathematics and Informatics of Vilnius University. She graduated from Vilnius University, Faculty of Mathematics, in 1989. Her research interests include representing of legal knowledge, inference in legal knowledge bases, contracts and agreements in the context of SOA. She has authored or co-authored over 20 publications.

KONTRAKTAI IR SUSITARIMAI SAITYNO PASLAUGŲ ARCHITEKTŪROJE

Laima Paliulionienė

Santrauka

Kontraktų sąvokos klausimas natūraliai iškyla kalbant apie saityno paslaugas, nes realiaame gyvenime paslaugos yra derybų ir kontraktų objektas. Tačiau kontrakto sąvoka saityno paslaugų architektūroje (angl. *Service-Oriented Architecture (SOA)*) yra šiek tiek klaidinanti, nes realiai dažniausiai reiškia tik techninę specifikaciją – kokius parametrus reikia pateikti ir kokios prieš sąlygos turi galioti, kad galima būtų pasinaudoti paslauga. Tačiau yra ir kiti, ne vien techniniai kontrakto aspektai. Kartu su kontraktu dažnai naudojamas susitarimas dėl paslaugos lygmens (angl. *Service Level Agreement (SLA)*), kuriame nurodomos matuojamos savybės (reakcijos greitis, paslaugos pasiekiamumo tikimybė ir kt.) ir jų sutartos reikšmės. Straipsnyje nagrinėjamos kontrakto ir susitarimo sąvokos SOA kontekste, atkreipiant dėmesį į jų teisinius aspektus. Aptarta, kokios kalbos naudojamos tokiems kontraktams ir susitarimams užrašyti. Palygintos susitarimams dėl paslaugos lygmens skirtos kalbos WS-Agreement ir WSLA, išskiriant jų elementus, susijusius su sutarčių teise.

Pagrindiniai žodžiai: saityno paslaugų architektūra, SOA, kontraktas, susitarimas dėl paslaugos lygmens, SLA