

A View-based Approach to Quality of Service Modelling in Service-oriented Enterprise Systems

Audrone Lupeikiene, Jolanta Miliauskaite and Albertas Caplinskas

Institute of Mathematics and Informatics, Akademijos 4,
LT-08663, Vilnius, Lithuania
{audrone.lupeikiene, jolanta.miliauskaite, albertas.caplinskas}@vu.mii.lt

Abstract. Service-oriented enterprise architecture and emerging advanced technologies bring up new problems for enterprise systems engineering or, more exactly, for service-oriented enterprise systems (SoESs) engineering. One of the most important issues is the modelling of quality of services (QoS). The QoS is a complex and multi-sided concept. In various contexts and for different aims it may be defined quite differently. In the context of SoES, QoS has also some specifics that in several aspects differ from the systems implementing service-oriented architecture (SOA) in other contexts, for example, in business-to-consumer or utility-service ones. The paper suggests that it is possible to identify several understandings of QoS reflecting the viewpoints of different SoES stakeholders in a similar way as it is carried out for software products. It also proposes a conceptual view-based framework for QoS modelling in web-based SoES and for balancing different viewpoints.

Keywords: Service-oriented enterprise systems, quality of service, view-based modelling, view reconciliation.

1 Introduction

Service-oriented Architecture (SOA), cloud computing and other advanced technologies bring up new problems for enterprise engineering. The symbiosis of Enterprise Architecture and SOA results in the so-called Service-oriented Enterprise Architecture (SoEA).

A number of different stakeholders participate in the development, maintenance and service consumption activities during the SoES development lifecycle. They have different, often conflicting understandings of SoES service quality and the way how it should be described, specified, and evaluated. Thus, despite a large number of QoS models and ontologies, the question “What does QoS mean?” still has no final answer, at least, in the context of SoES. The open issue remains also the quality evaluation of the overall SoES. According to [1], “*Discussions in an industrial context brought up controversial opinions: both positive and negative impacts on the overall product quality have been attributed to SOA, which is a clear indicator that SOA is in need of specific quality modelling approaches*”.

This paper aims to cope with the conflicts in QoS evaluation given by different stakeholders of SoES. It proposes a conceptual view-based framework to describe and relate to each other the different viewpoints and perspectives of QoS in web-based SoES. Generalising the hypothesis presented in [2], we assume that a better overall quality of service can be achieved by the interplay of different viewpoints on quality and different perspectives or layers, in terms of the Open Systems Interconnection Reference Model [3], when the QoS is considered as a whole, than in cases when different viewpoints and perspectives are examined separately. Notwithstanding the fact that our research is still in progress, we hope that this paper will contribute to deeper understanding of the QoS phenomenon in SoES and its modelling.

The rest of the paper is organised as follows. Section 2 discusses the related works. Section 3 describes the research methodology. Section 4 presents our understanding of SOA and SoES. Section 5 discusses the proposed framework. Finally, Section 6 concludes the paper.

2 Related Works

A lot of QoS models and ontologies have been proposed. There are also many good and exhaustive surveys on the QoS models ([4] among others), QoS specification and evaluation. However, there are only a few works [5], [6], [7], [8] surveying QoS models devoted to SoES. Besides, these surveys focus on some specific aspects of the problem and, to the best of our knowledge, no one investigates this problem systematically.

It is not easy to say which of the proposed QoS models is best suited or most representative one. The closest to our research are: a) Quality Model (SQuaRE-WSQM) [9], b) S-Cube Reference Model (SCRM) [10], and c) OASIS Quality Model for Web Services (WSQM 2.0) [11], [12].

Most of QoS models are more or less exhaustive taxonomies of the quality attributes. SCRM is among the most exhaustive ones. It defines a hierarchical structure of 89 quality attributes on 4 levels together with the definitions and descriptions of these attributes. This classification follows ISO 9126 standard. SQuaRE-WSQM also follows the similar scheme established by ISO/IEC SQuaRE model. It was even developed to be compatible with universal software quality models. SQuaRE WSQM distinguishes 3 different viewpoints corresponding to internal web service quality, external web service quality, and web service quality in use. It takes into account that web services should be viewed as the union of two inseparable parts – a technical interface and business functionality this interface provides an access to. The QoS is decomposed into 6 layers: hardware quality, software quality, service quality, transportation quality, composition quality, and quality in use. This model follows IEEE1061 standard, which requires each quality model to enable the top-down decomposition of quality requirements and bottom-up measurement of quality indicators. It also defines the dependencies among QoS layers in web service lifecycle.

SCRM, partly SQuaRE-WSQM and a number of other similar QoS models can be characterised as taxonomy-based models. Such models structure quality along the characteristics (e.g., security, interoperability, reliability, usability, efficiency, maintainability and portability). As pointed out in [13], *“this classification approach often lacks clear semantics regarding the relationship between super and sub-attributes”*. To eliminate this shortcoming, the so-called activity-based quality models, in which the quality is described along the activities performed on or with SOA system, were introduced in [14]. Although the activity-based approach was proposed to model software maintainability, it was also successfully applied to model SOA systems quality [1].

Like the other taxonomy-based QoS models, WSQM 2.0 provides a hierarchical taxonomy of quality attributes, referred to as quality factors. The model categorises quality factors into two groups: the business quality group and the system quality group [12]. This is in line with the widely accepted understanding of QoS, which currently includes also knowledge about cost, payments and other business-related characteristics. The first group of quality factors is related to business perspective, the second one – to system perspective. The system quality group further is divided into the variant and invariant quality parts. The first one includes quality factors, whose values can be dynamically changed at run-time when a service is used. The second one refers to quality factors, whose values are determined as soon as the service development is completed. It includes interoperability quality, business processing quality, manageability quality, and security quality. The business quality group includes price, penalty and incentive, business performance, service recognition, service reputation and service provider reputation. Thus, WSQM 2.0 is based on the sophisticated and well-developed taxonomy of quality factors. Nevertheless, it cannot be classified as a pure taxonomy-based model – it has some activity-based flavour because the quality factors are interrelated with the so-called quality activities and quality associates. Quality associates are specific roles or tasks within an organisation, which interact with the artefacts or with each other through the activities. Quality activity refers to various action models performed by quality associates to ensure the stability of web services quality, for example, contracting among quality associates [11].

Some activity-based flavour can be found even in the SQuaRE WSQM because its perspectives are related to the development, management, and employment contracting activities. These activities, in turn, are related to roles (service developer, service provider, service broker, etc.) defined in terms of responsibilities. The model allows one to define additional domain-specific quality characteristics and to make it context-dependent. However, the proposed activities, roles and view dependencies are only mentioned, but not described in detail. The model is too much technology as well as product oriented and encompasses only three viewpoints. Nevertheless, our research to some extent is inspired by the philosophy beyond this model.

There are also ontology-based QoS modelling approaches, for example [15]. However, they are mostly used in QoS modelling for service discovery and are not directly related to our research. Besides, there are no obstacles to use appropriate QoS

ontology to model viewpoints and perspectives in the conceptual view-based framework proposed in our paper.

3 Research Methodology

The research was conducted in two steps. First of all, using the methodology proposed by Webster and Watson [16], a concept-centric literature survey on the QoS in SOA with special emphasis on SoES was carried out. In the second step, the results of the survey were generalised and juxtaposed against the results of empirical analysis of observations made by the authors working in a research project for the development of an innovative Advanced Planning and Optimisation system for make-to-order production environments.

4 SOA and SOEA

The most important innovation of service orientation is the manner in which the separation of concerns is done. SOA is an architectural style that implements this approach. SoEA is a substyle of SOA. However, various authors define SOA and SoEA differently. Therefore, before describing the proposed conceptual view-based framework for QoS modelling in SOES, it is necessary to explain what we mean by SOA and SoES in this paper.

We adopt the following SOA definition proposed by Bieberstein et al. [17]:

“A service-oriented architecture is a framework for integrating business processes and supporting IT infrastructure as secure, standardized components – services – that can be reused and combined to address changing business priorities”.

In this sense, SoEA introduces two new high-level abstractions, namely, enterprise business services (EBS) and enterprise business processes (EBP).

Enterprise business services are the abstractions of existing application capabilities, which are aligned with the enterprise business functions. Note that EBSs are not web services. According to [18], an EBS is the mechanism by which the needs and capabilities are brought together. It may be viewed “...as the notional or existing business functionality that would address a well-defined need. Service is therefore the implementation of such business functionality that it is accessible through a well-defined interface” [18]. In other words, an EBS is a unit of business logic that implements one well-defined action, for example, creates an order.

EBPs are the abstractions of the overall business functioning. They are defined as orchestrations of EBSs. As a result, SoEA sees a software system as a set of interacting EBSs coordinated by an enterprise business process.

Like SOA, SoEA is “an architectural style where systems consist of service users and service providers” [19]. Service providers are those functional units of the system that offer EBSs. They are analogous to servers in client-server architecture. In other words, they are software units “hosting” one or more EBS. It is assumed that each

EBS provider resides in the separate computer network node, which is accessible through the name or locator other than an absolute network address. Services or, more exactly, service consumers in SoEA are those functional units of the system that invoke EBSs provided by service providers. They are analogous to clients in client-server architecture. In other words, they are software units that form and send requests to service providers. Service consumers can dynamically discover service providers. Service providers and service consumers are role names. It means that in SoEA some functional units, depending on the context, may act in both roles. Each service consumer also resides in a separate computer network node.

SoEA provides a special directory service (also referred to as service discovery agency) that allows service consumers to register and discover any EBS. Besides, any EBS is dynamically bound. It means that a service consumer does not need the EBS implementation available at build time because the service is located and bound at runtime [19].

The SOA has several substyles. One of the most popular substyles is *web-based SOA*. Web-based SOA (wbSOA) is SOA that is implemented using web service technology. The same is true for SoEA. We will consider only web-based SoEA (wbSoEA) further in this paper. Although there are a number of different viewpoints how wbSOA (and wbSoEA) should be implemented using web services, Microsoft Windows Communication Foundation [20], IBM Websphere [21], SAP Enterprise SOA [22] among others, it is possible to discuss these issues at the most general conceptual level.

According to SOA Reference Architecture specification, a web service is

“...a mechanism to enable access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description” [18].

It means that web services are used to wrap the distributed components (or some legacy software) and to implement service providers' interfaces. In other words, a web service is the mechanism that converts components (or legacy software) into EBSs by encapsulating EBS providers and by creating unified platform agnostic interfaces, which allow accessing EBSs via the Internet. The W3C open specification [23] defines a platform independent XML-based machine-readable interface description language – Web Services Description Language (WSDL) – which allows describing the functionality offered by a web service that wraps the component or other software. Web services platform also provides a number of other open specifications centred around interface descriptions based on WSDL, web services messaging framework, and service description registration and discovery.

SoEA is *“a style of design that guides all aspects of creating and using business services throughout their lifecycle (from conception to retirement)”* [24]. SoEA provides guidelines for creating and using service-oriented applications. SoES is business-driven, that is, it must support enterprise's business strategy and objectives. It means that business processes in SoES must be designed keeping in mind this goal. On the other hand, business processes should be translated into abstracted and normalised EBSs drawing on global data types. Normalisation means that each EBS should be designed with the intent to avoid functional overlaps and to reduce the

redundancies in EBSs, i.e., to avoid similar or duplicate bodies of service logic. Global data types are enterprise-wide defined data types based on the international standards [25].

EBSs have also some specifics. First of all, all EBSs, including those maintained by the external providers, should be designed, developed and deployed in compliance with enterprise-wide standards. It means that the structures of both services and messages must be unified [26]. All the EBS interfaces must be clearly defined and stable, and make use of global data types [27]. The enterprise business service must meet the functional objectives within the context of the business unit and the enterprise [28]. Internal directory service must be provided for registration and discovery of EBSs. This specific enables one to balance viewpoints of different EBS stakeholders, i.e., to apply the approach proposed in our paper.

5 A View-based Approach to QoS Modelling in Service-oriented Enterprise Systems

Despite the fact, that there are a number of QoS definitions, this term still remains not well defined and frequently is misused [29]. The term *QoS* was introduced in telecommunications, where it was focused on the service performance measures from the network perspective. Later it was extended, including even hardly related to quality characteristics such as service requestor's satisfaction or service cost¹.

At present, the term *QoS* refers to several different things. As stated in [4], "*This set of quality attributes does not characterize only the service but any entity used in the path between the service and its client. Such an entity may exist in any of the three possible service levels. Thus, different QoS attributes may be used to define the QoS in the application, service, and infrastructure levels.*" There the term *client* refers to the service requestor.

In the context of SoES, the quality of an EBS is typically addressed by the term *QoS for web services* (WS QoS) that, unfortunately, causes some confusion. One of the most popular WS QoS definitions is presented in [30], where this term is defined as "*a set of nonfunctional attributes of the entities used in the path from the WS to the client that bear on the WS's ability to satisfy stated or implied needs in an end-to-end fashion*". The definition speaks about EBS end-to-end quality but calls it *web service quality*. However, in many contexts it is important to distinguish between these qualities, i.e., to differentiate between the extent to which an EBS satisfies stated or implied service requestor's needs and the manner in which this EBS is delivered. Using the terms proposed by Ch. Grönroos [31], it is important to distinguish between EBS *technical quality* and EBS *functional quality*. As pointed out in [32], a variety of factors, including resource constraints, market conditions, and incompetence of available developers, may result in a discrepancy between the real needs of end user and the actual Service Level Agreement. Consequently, the gap often appears

¹Note that in this paper service requestor and service consumer are different entities. The first one is a person or an organisation while the second one is a piece of software.

between service requestor's needs and service owner's treatment of those needs. Besides, from service requestor point of view, often it is important not only to satisfy his or her needs but also to take into account some additional factors. Thus, it is confusing to use the term *WS QoS* to address the end-to-end quality experienced by an EBS requestor.

Besides, the definition presented in [30] and similar ones emphasise only one aspect of the concept, namely, non-functional attributes. It is also unclear whose viewpoint (service requestor's, service owner's, etc.) it represents. The Web Service Quality Model (WSQM) [11], [12] proposed by OASIS attempts to take into account the many-sided nature of the concept. However, this model still does not encompass all aspects of EBS quality and is not purported to balance them. Thus, there is still a need to develop a framework that encompasses and balances all viewpoints and perspectives on EBS quality at a higher abstraction level.

We advocate that the consideration of QoS in the context of SoES should be performed applying view reconciliation methodology, which in the field of Computer Science was originated by SADT methodology [33] and J. Leite's PhD thesis [34], and was further developed mainly in software requirements engineering. Our claim is motivated by the fact that QoS requirements to some extent are akin to software quality requirements.

In the view reconciliation methodology, basic terms are *viewpoint*, *perspective*, and *view*. In our approach, the term *viewpoint* refers to an attitude used to define EBS QoS. Following [35], we define two kinds of viewpoints: viewpoints associated with a particular role and viewpoints reflecting a particular standpoint. Our hypothesis is that, despite many differences between the tangible products (e.g., software products) and intangible services, definitions of QoS may be grounded by the same attitudes to quality as definitions of the quality for products [36]. In accordance with this hypothesis, we define the following viewpoints on the EBS quality:

- **Metaphysical viewpoint.** EBS quality is a degree of excellence where excellence is defined as an abstract ideal, which shows the direction where services are heading to but will never get there.
- **Cost-based viewpoint.** EBS quality is a degree of excellence at an acceptable price.
- **Value-based viewpoint.** EBS quality is an adequacy of service to requestor's values and preferences. It differs depending on service requestor for whom it is defined.
- **Pragmatic viewpoint.** EBS quality is the balance of features and characteristics of service that bear on its ability to satisfy stated or implied needs of service requestor. It depends on a particular context in which the service is consumed or, in other words, the judgment about the quality of a service depends on the aims and goals for which this service is intended to be consumed.
- **Provider's viewpoint.** EBS quality is the compliance with the stated requirements, which are mostly formulated in business and technical terms.
- **Designer's viewpoint.** EBS quality is something that is defined by the values of quantifiable and measurable internal characteristics of a service. This viewpoint

assumes that the greater the number of desired attributes possessed by a service, the higher the quality of service.

In the proposed approach, the term *perspective* is defined as some aspect of the EBS quality on which any viewpoint, in principle, may be focused. It does not mean that all perspectives necessary cut across all viewpoints. Some viewpoints, for example, metaphysical ones, may ignore some perspectives.

A perspective defines a subset of QoS characteristics that are observed from the viewpoint. It is supposed that each pair <viewpoint, perspective> defines a different subset of QoS characteristics because any viewpoint focuses only on the quality aspect which is its concern. However, the intersection of such subsets does not necessarily have to be an empty set.

On the basis of a concept-centric literature survey on the QoS and our experience, we consider the following EBS quality perspectives:

- **Presentation perspective.** This perspective describes EBS quality from the point of view of the presentation of information produced by an EBS for service requestors. To some extent it overlaps with the data quality mentioned in [4]. The description includes: the relevance, granularity, and level of detail of presented information; its accuracy, consistency, completeness, and timeliness; appropriateness of its visualisation, perspicuity, and transparency for a service requestor; etc. One of the problems arising in the context of SoEA in describing QoS from this perspective is the separation of concerns between EBS itself and software, which is implemented by a service consumer. The latter describes rather the quality of the whole SoEA system than the quality of a service.
- **Transportation perspective.** This perspective describes EBS quality from the network point of view. The description includes the response time, maximal throughput, service availability, network reliability, etc. In the context of SoEA, the problem of the separation of system and service concerns arises again.
- **Infrastructure perspective.** This perspective describes EBS quality from the implementation platform point of view. The description includes the performance, reliability, security and other related issues. In SoEA the implementation platform usually is shared among many or even all services. As a result, the problem of the separation of system and service concerns arises again.
- **Web service perspective.** Web service is the software that exposes EBS. Thus, the quality of implementation of the software affects the quality of the whole EBS. However, it is even more important to evaluate the software, first of all, as a service but not as a software product. It means that the description of this perspective along with the reliability, security and other product-related issues must describe messaging, responsiveness, courtesy (politeness, respect for service requestor, friendliness, etc.).
- **Application perspective.** As a rule, the functionality of an EBS service is implemented by some application, i.e., by some software component or some legacy software. In various implementation platforms the components are named differently, for example, in Microsoft Windows Communication Foundation [20] the components are addressed to using a term *service classes*. The non-functional properties of the application affect the quality of the whole EBS. The application

perspective describes the nonfunctional properties of a component as a software product or EBS technical quality.

- **Data perspective.** According to [37] and many other sources, in SoES there is a special kind of services – SoES data service. In such services, web services encapsulate data and the supported behaviour, for example, the operations that manipulate the data. The term *data* is used here to address data stored in the enterprise’s data bases, as well as XML documents and various contents: “...a single data service will usually only expose or manipulate a core set of data, rather than all data for the entire enterprise” [37]. It is obvious that the quality of encapsulated data essentially affects the QoS of the whole SoEA service as well as QoS of components processing these data. Thus, the data perspective describes EBS quality from the point of view of the quality of encapsulated data.
- **Domain perspective.** The domain-dependent EBS quality attributes capture those properties which are specific to a particular business domain, for example, for online banking services or for online streaming multimedia services. In addition, for practical reasons, even the attributes that are defined in all business domains can be treated differently in different domains [30]. The specific nature of a particular domain may affect weights assigned to the values of such EBS quality attributes. For example, in online streaming multimedia services the attribute *bits-per-second* is more important than the *security*. In online banking services, vice versa, the *security* is more important than the *bits-per-second* [38]. *Inter alia*, despite the fact that media applications, including video-oriented ones, also emerge in SoEA systems, up to date they are rather marginal there (an exhaustive discussion on the QoS of video-oriented services can be found in [39]). In SoEA system, the domain perspective describes EBS quality from the point of view of a particular (internal) enterprise domain, for example, manufacturing or human resource management.
- **Socio-economic perspective.** This perspective describes EBS quality from the point of view of business, economic and social issues. The price of service, payment mode (e.g., kinds of accepted bank cards), legal constraints, and other similar issues should be described. *Business effect of the service* is one of the most important socio-economic characteristics for the EBS.

There is one more perspective that describes EBS quality from a service composition point of view [9]. However, in this paper we ignore it because of the limited space we do not consider the modelling of EBS composition quality.

In the proposed approach, the term *view* refers to the result of integration of perspectives taken from the same viewpoint. In this paper, we discuss neither the integration of perspectives nor viewpoint construction techniques, nor the view balancing techniques because the research is still in progress. The preliminary viewpoint construction techniques were developed by generalising, composing, extending, and modifying the perspective integration techniques proposed in the traditional requirements engineering [34], [35], business process modelling [40], enterprise modelling [41], architecture engineering [42], and development of distributed applications [43]. To balance the views, we model them as weighted QoS goals [44], [45] and try to apply interactive balancing procedure inspired by [40].

However, all these techniques and procedures should be investigated experimentally and may be modified.

6 Conclusion

Despite the increasing importance of service-oriented systems and numerous publications on QoS, this concept still remains rather vague defined. It is due to complexity, multi-dimensionality, and multi-sided and context-dependent nature of this concept. The QoS may be considered from a service owner's, requestor's, designer's, network's and other perspectives. Significant part of scientific publications is devoted to the quality perceived by a service requestor. However, it is not a simple task to precisely define the perceived quality, because this quality is associated with subjective estimates depending on requestor's expectations, past experience and preferences that in turn can be influenced even by the present fashion trends and the word of mouth. On the other hand, this concept is very important because the lack of common understanding of QoS is a serious obstacle to direct efforts of all stakeholders of service-oriented system under development towards a particular common cause. Of course, in those service-oriented systems, which work in unpredictable open-world settings without the clear boundary between a system and its environment, such common understanding cannot be achieved even in principle, because such systems are rather dynamically composed at run-time than systematically designed and developed.

The service-oriented enterprise systems can be considered at least partly based on the closed-world assumption and, consequently, can systematically be designed and developed. Besides, such systems less suffer from the distributed service delivery accountability. As a rule, the enterprise itself owns the most part of EBS and networks, and has long time contracts with the external service owners. Nevertheless, the current QoS modelling approaches – taxonomy-based, activity-based, and ontology-based ones – cannot serve for the modelling of the often contradictory QoS requirements stated by different stakeholders and for the balancing of these requirements.

The paper proposes a conceptual view-based framework for modelling and balancing different views on quality. It assumes that construction of viewpoints can be carried out by applying the combination of slightly modified and adapted for SOES context view construction techniques, which were developed in the traditional requirements engineering, business process modelling, enterprise modelling, architecture engineering, and development of distributed applications. It also suggests that in a service-oriented enterprise system it is possible to balance views, which reflect different viewpoints, by modelling these views as weighted QoS goals, and to balance these goals by applying an interactive procedure.

Acknowledgments

The research 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

1. Goeb, A., Lochmann, K.: A Software Quality Model for SOA. In: Proceedings of the 8th International Workshop on Software Quality (WoSQ'11), September 4, 2011, Szeged, Hungary. ACM, New York, NY, USA, pp. 18-25 (2011)
2. Siller, M., Woods, J.: Improving Quality of Experience for Multimedia Services by QoS Arbitration on a QoE Framework. In: Proceedings of the 13th IEEE Packet Video Workshop (IEEE PV 2003), Nantes, France, April 28-29 2003. pp.1-7 (2003)
3. ISO/IEC 7498-1 (E): International Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model. International Standard. International Standard, ISO/IEC (1996-06-15) (1994)
4. Benbernou, S., Brandic, I., Cappiello, C., Carro, M., Comuzzi, M., Kertesz, A., Kritikos, K., Parkin, M., Pernici, B., Pleban, P.I.: Modeling and Negotiating Service Quality. In: Service Research Challenges and Solutions for the Future Internet: S-Cube - Towards Engineering, Managing and Adapting Service-Based Systems. LNCS 6500. Springer, pp. 157-209 (2010)
5. Wang, G., Chen, A., Wang, Ch., Fung, C., Uczekaj, S.: Integrated Quality of Service (QoS) Management in Service-Oriented Enterprise Architectures. In: Proceedings of Eighth IEEE International Enterprise Distributed Object Computing Conference (EDOC 2004), pp.21-32 (2004)
6. Farroha, B., Farroha, D.L.: Policy-based QoS Requirements in a SOA Enterprise Framework-an Investigative Analysis. In: Proceedings of IEEE Military Communications Conference (MILCOM 2007), pp.1-7 (2007)
<http://dx.doi.org/10.1109/MILCOM.2007.4454852>
7. Wu, Q., Li, P.: Study and Implement of Dynamic Routing Based on QoS in Enterprise Service Bus. Journal of Computational Information Systems 6(7), pp. 2093-2098 (2010)
8. Usmani, Sh., Azeem, N., Samreen, A.: Dynamic Service Composition in SOA and QoS Related Issues. International Journal of Computer Technology and Applications 2(5), pp. 1315-1321 (2011)
9. Abramowicz, W., Hofman, R., Suryn, W., Zyskowski, D.: SQuaRE based Web Services Quality Model. In Ao, S., Castillo, O., Douglas, C., Feng, D.D., Lee, A-J., eds.: Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS 2008), 19-21 March 2008, Hong Kong. LNECS 2168 I. Newswood Limited, Hong Kong, pp. 827-835 (2008)
10. Papazoglou, M., Pohl, K., Parkin, M., Metzger, A., eds.: Service Research Challenges, LNC 6500. Springer, Berlin, Heidelberg, New York (2010)
11. Kim, E., Lee, Y., (eds.): Quality Model for Web Services (WSQM -2.0). September 2005, OASIS (2005).
<http://www.clip.dia.fi.upm.es/Projects/S-CUBE/papers/oasis05:WSQM-2.0.pdf>

12. Kim, E., Lee, Y., Kim, Y., Park, H., Kim, J., Moon, B., Yun, J., Kang, G.: Web Services Quality Factors, Version 1.0, Candidate OASIS Standard. 31 October 2012, OASIS (2012). <http://docs.oasis-open.org/wsrm/WS-Quality-Factors/v1.0/cos01/WS-Quality-Factors-v1.0-cos01.pdf>
13. Döweling, S., Schmidt, B., Göb, A.: A Model for the Design of Interactive Systems based on Activity Theory. In: Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work (CSCW'12), February 11–15, 2012, Seattle, Washington, USA. ACM, New York, NY, USA, pp. 530–548 (2012) <http://dx.doi.org/10.1145/2145204.2145287>
14. Deissenboeck, F., Wagner, S., Pizka, M., Teuchert, S., Girard, J.-F.: An Activity-Based Quality Model for Maintainability. In: Proceeding of the IEEE International Conference on Software Maintenance, (ICSM '07), Paris, 2–5 Oct. 2007. IEEE Computer Society Press, Washington, pp. 184–193 (2007)
15. Bianchini, D., De Antonellis, V., Melchiori, M.: QoS in ontology-based service classification and discovery. In: Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04), pp. 145–150 (2004)
16. Webster, J., Watson, R.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly* 26(2), pp. xiii–xxiii (2002)
17. Bieberstein, N., Bose, S., Fiammante, M., Jones, K., Shah, R.: Service-Oriented Architecture (SOA) Compass – Business Value, Planning, and Enterprise Roadmap. IBM Press (2005)
18. SOA-RAF: Reference Architecture Foundation for Service Oriented Architecture, Version 01. OASIS Committee Specification 01 (2012)
19. Bianco, P., Kotermanski, R., Merson, P.: Evaluating a Service-Oriented Architecture. Technical Report, CMU/SEI-2007-TR-015, ESC-TR-2007-015, Software Engineering Institute, Carnegie Mellon (2007)
20. Chapell, D.: Introducing Windows Communication Foundation. Microsoft Corporation (2007)
21. Sadtler, C., Cuomo, G., Ganci, J., Haberkorn, M., Jones, C., Kovari, P., Griffith, K., Marhas, D., Will, R.: WebSphere Product Family Overview and Architecture. Redbooks, IBM (2005)
22. Hack, S., Lindemann, M.: Enterprise SOA Roadmap. Galileo Press (2008)
23. Chinnici, R., Moreau, J., Ryman, A., Weerawarana, S.: Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. W3C Recommendation (2007)
24. Newcomer, E., Lomow, G.: Understanding SOA with Web services. Addison-Wesley (2005)
25. Sambeth, M.: Enterprise SOA. Mastering Future Business. Presentation slides, SAP AG (2006)
26. Fischer, C.: Enterprise SOA: General Principles, ES Bundles, ATP Check. Presentation slides, April 2007, SAP AG (2007)
27. SAP: Enterprise SOA Development Handbook 1.1., SAP AG (2008)
28. Bieberstein, N., Bose, S., Walker, L., Lynch, A.: Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals. *IBM Systems Journal* 44(4), pp. 691–708 (2005) <http://dx.doi.org/10.1147/sj.444.0691>
29. Zapater, M., Bressan, G.: A Proposed Approach for Quality of Experience Assurance for IPTV. In: Online Proceedings of the First International Conference on the Digital Society (ICDS '07), 2–6 January 2007, Gosier, Guadeloupe. IEEE Computer Society, Washington, DC, USA, p. 25 (2007)
30. Kritikos, K., Plexousakis, D.: Requirements for QoS-based Web Service Description and Discovery. *International Journal of Web Services Practices* 3(2), pp. 72–82 (2008)

- 31.Grönroos, Ch.: Strategic Management and Marketing in the Service Sector. Research report 8, Swedish School of Economics and Business Administration, Helsingfors (1982)
- 32.Parasuraman, A., Zeithaml, V., Berry, L.: A Conceptual Model of Service Quality and Its Implications for Future Research. *Journal of Marketing* 49, Fall 1985, pp. 41-50 (1985)
- 33.Ross, D.: Structured Analysis for Requirements Definition. *IEEE Transactions on Software Engineering* 1(3), pp. 363-386 (1977)
- 34.Leite, J.C.S.P.: Viewpoint resolution in requirements elicitation. Ph.D. Thesis, Dept. Computer Sci., Univ. California, Irvine (1988)
- 35.Sommerville, I., Sawyer, P.: Viewpoints: principles, problems and a practical approach to requirements engineering. *Annals of Software Engineering* 3, pp.101–130 (1997) <http://dx.doi.org/10.1023/A:1018946223345>
- 36.Garvin, D.: What does “product quality” really mean? *Sloan Management Review* 26(1), pp. 25-43 (1984)
- 37.Bean, J.: Data Services: Resolving the Integration Challenge with SOA, White Paper. In: Dataversity, Webshire Conferences, April 1 (2011). <http://www.dataversity.net/data-services-challenge-resolving-the-integration-challenge-with-soa/>
- 38.Pathak, J., Koul, N., Caragea, D., Honavar, V.G.: Discovering Web Services over the Semantic Web. In: Proceedings of the 7th ACM International Workshop on Web Information and Data Management, Bremen, Germany, pp.1-10 (2005)
- 39.Cisco: Enterprise QoS Solution Reference Network Design Guide, version 4.0. Cisco Systems, Inc. (2008)
- 40.Letsholo, K., Chioasca, E.-V., Zhao, L.: An Integration Framework for Multi-Perspective Business Process Modeling. In: Proceedings of the 2012 IEEE Ninth International Conference on Services Computing. IEEE Computer Society, Washington, DC, USA, pp. 33-40 (2012) <http://dx.doi.org/10.1109/SCC.2012.29>
- 41.Frank, U.: Multi-perspective Enterprise Modeling (MEMO) Conceptual Framework and Modeling Languages. In Sprague Jr., R.H., ed.: Proceedings of the Hawaii International Conference on System Sciences (HICSS-2002). Los Alamitos, CA. IEEE Computer Society, Washington, DC, USA, pp. 1258–1267 (2002)
- 42.Verjus, H., Cimpan, S., Razavizadeh, D.: BeeEye: A Framework for Constructing Architectural Views. In Babar, M.A., Gorton, I., eds.: Software Architecture. 4th European Conference, ECSA 2010, Copenhagen, Denmark, August 23-26, 2010 LNCS 6285. Springer-Verlag, Berlin, Heidelberg, pp. 376-383 (2010)
- 43.Thanitsukkarn, T.: Multiperspective Development Environment for Configurable Distributed Applications. PhD Thesis, Imperial College of Science, Technology and Medicine, University of London, Department of Computing, London (1999)
- 44.Yu, E.: Modelling Strategic Relationships for Process Reengineering. PhD Thesis, University of Toronto, Department of Computer Science, Toronto (1995)
- 45.Yu, E.S.K.: Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. In: Proceedings of the 3rd IEEE International Symposium on Requirements Engineering (RE'97) Jan. 6-8, 1997, Washington D.C., USA. IEEE, Washington D.C., USA pp. 226-235 (1997)
- 46.Horkoff, J., Yu, E.S.K.: Evaluating Goal Achievement in Enterprise Modeling – An Interactive Procedure and Experiences. In Persson, A., Stirna, J., eds.: The Practice of Enterprise Modeling. Proceedings of the 2nd IFIP WG8.1 Working Conference PoEM 2009. Stockholm, Sweden. Nov. 2009, Lecture Notes in Business Information Processing 39, Springer, Berlin, Heidelberg, pp. 145-160 (2009)