Towards the Definition of a Framework Supporting High Level Reliability of Services

Firmino Silva[1,2] and Claudia-Melania Chituc[1]

[1] University of Porto, Faculty of Engineering, Informatics Engineering Department, Portugal
[2] Polythechnic Institute of Porto / ISCAP
fsilva@iscap.ipp.pt, cmchituc@fe.up.pt

Abstract. In today's networked economy, an increasing need exists for companies to interact dynamically focusing on optimizing their skills and better serve their joint customers. Service oriented computing provides the means to achieve such objectives. This paper presents an approach towards the definition of a framework supporting a choreography of services built according to customer's requirements. The proposed framework is built on a set of specific metrics that translates the high level reliability of a service, which are calculated at various levels of the choreography, focusing on four main dimensions: technical capacity and performance, product or service purchased, customer satisfaction perspective, and provider's and business partners' choreography. This approach is then illustrated and discussed with a case example from the automotive sector.

1 Introduction

Service Oriented Computing (SOC) provides support to cross-organizational business processes (COBP) build on different applications and crossing heterogeneous computing platforms, based on cooperating services where application components are assembled with little effort into a network of loosely coupled services [1]. These service applications are suitable to run in dynamic business environments and are able to address constantly evolving customers' requirements [2]. Thus, it is important that monitoring approaches (e.g., conceptual frameworks, metrics, tools) are defined so that business partners and managers can have details on process and service assessment and monitoring in order to identify where, how, why and when improvements can be made. Assessment and monitoring of COBP and services are fundamental to understand the real added-value each business partner brought into the choreography and business process. Results obtained from the assessment and monitoring are useful to adapt, correct or adjust the business processes and services and their choreography.

This paper presents partial results of an ongoing research project aiming at developing a conceptual framework and metrics to support high level reliability of services in a business environment. These metrics can be calculated at different levels of a service choreography targeting four dimensions: technology (e.g., technology performance, capacity), process, product or service to be acquired, used or offered (e.g., delivery cost, level of quality), prospect of customer satisfaction (e.g., trust, operability) and the provider (and partners) expectation (e.g., profit, rate of return).

A. Ghose et al. (Eds.): ICSOC 2012, LNCS 7759, pp. 143–154, 2013.

The rest of this paper is organized as follows. The main concepts are presented in the next section. Metrics for high level reliability of services are described in Section 3. The proposed framework and a basic scenario from the automotive industry are introduced in Section 4. Related work is presented in Section 5. The article concludes with a section addressing the need for future research work.

2 Main Concepts

Services are fundamental elements for developing rapid, low-cost, interoperable, evolvable, and massively distributed applications [3][4]. A main goal of SOC is to gather a collection of software services, make them available/accessible via Internet over standardized (XML-based) languages and protocols, which can be implemented via a self-describing interface based on open standards [4]. Their functionalities can be automatically discovered and integrated into applications or composed to form more complex services, and they can perform different functions, e.g., ranging from answering to simple requests, to executing sophisticated business processes requiring peer-to-peer relationships among multiple layers of service consumers and providers. According to [5], SOC based on Web services is currently one of the main drivers for the software industry.

SOC relies on the Service-Oriented Architecture (SOA) to build the service model, where services are autonomous, platform-independent entities that can be self-described, published, discovered, and loosely coupled [3]. They promote a distributed computing infrastructure for both intra and cross-enterprise application integration and collaboration. Service providers supply their service descriptions with related technical and business support, e.g., allowing business partners to discover, select, build and compose services. Service descriptions are used to advertise the service functionalities, interface, behavior, and quality.

As consumers move towards adopting SOA, the quality and reliability of the services become important aspects. The service requirements vary significantly from customer to customer. To balance the customer expectations, a negotiation process must occur and the service must be leveled with the commitment of an agreement: a Service Level Agreement (SLA). The SLA [6] specifies the service level objectives both as expressions of requirements of the service consumer and assurances by the service provider on the availability of resources and quality of service (QoS). In order to guarantee the compliance with the agreed parameters, SLAs typically define the consequences associated with failures or violations.

Web-Service Level agreement (WSLA) has come to constitute an increasingly important instrument of monitoring Web-services environment where enterprises rely on services that may be subscribed dynamically and on demand. The WSLA framework [7] consists of a flexible and extensible language based on XML Schema and a run-time architecture comprising several SLA monitoring services, which is able to handle four different parameter types of metrics: resource metrics, composite metrics, SLA parameters and business metrics.

3 Metrics for High Level Reliability Services

The proposed framework relies on a set of metrics that are defined from: business rules, services, customer requirements, and results of a learning process. These elements are designed to ensure that the service desired by the customer is monitored and assessed and its characteristics are equal to those who represent the highest guarantee to cover all the relevant evaluation aspects of the choreography (agreed in the SLA). These assessments, allowing qualifying and quantifying the performance of services, provide a ranking of the best behaviors enriching the knowledge of the framework. A service evaluation matrix is proposed that stores the assessment results of the services iterations and keeps a ranking (Fig. 1).

High level reliability services refer in this article to the degree to which a choreography is expected to meet the requirements set by the client, allowing the identification of services capable to provide a higher performance considering the client's requirements. This is achieved by calculating the metrics covering the relevant dimensions of analysis for the choreography. The services that are part of the choreography are then selected considering their previous performance, e.g., results evaluations made in previous instantiations. In some cases, the assessment may reveal a value lower than the expected one. As services are ranked based on performance evaluations, choreographies are also evaluated and compared with respect to the performance that was expected to be achieved, and the level of performance actually achieved. This approach is detailed in Section 4.

3.1 Service Evaluation Matrix

According to the business rules for defining the choreography, the functional scope of each service is defined to add each service in the same pool so that they can "compete" in terms of performance within the same type. For each pool of services, a matrix is defined to store the ranking of services. The matrix stores the assessment results for all iterations resulting from their use in choreographies. The weights assigned to the evaluation criteria reflect the clients' requirements and importance assigned to each item. The values of each service evaluation matrix are recalculated taking into account the values characterizing the services in the customer's SLA. However, the scoring algorithm for measuring the rating for each service is always the same though is based on the new values resulting from the distribution of the new weights assigned (e.g.: considering the customer's requirements). Based on these rankings, a choreography of services is selected.

Figure 1 illustrates a simple example. Considering three services: Services A, B and C that perform the same functional context, a weight (w) is assigned by the customer for each service level. Four metrics are represented here by a to d. Each metric has a range of values (e.g.: a[0-3]; b[1-5]; c[2-4]; d[1-6]). The weights allocated to each metric, according to the profile of the service requested by a customer, are listed in each column of the table, e.g.: a[5%]; b[40%]; c[35%]; d[20%]. The calculation of service activity (cAct) is based on the sum of the average (v) of the assessment result of a service, multiplied by the product of the weight (w) of each metric:

			weights									
			5%	40%	35%	20%						
			metrics					scoring				
# in best chor	# iterations	services	a	b	c	d	cAct = ∑ v(S).w(M) S∈{A,B,C}; M∈{a,b,c}	+scoring 1	+scoring 2	+scoring 3	ranking
		A									
		B									
		C									

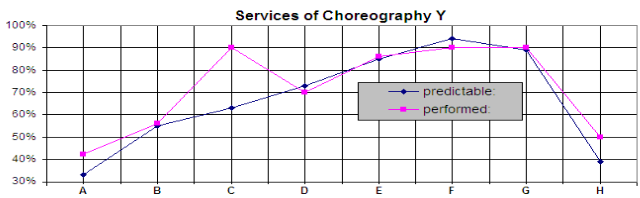
Fig. 1. Evaluation matrix

A scoring algorithm is then used to calculate a rating for each service, based on the information which reflects the behavior of services and the scoring rules. Different rules can be defined for the scoring algorithm, as follows:

- Scoring rule 1: Number of times the service is used in choreographies
- Scoring rule 2: Number of times the service is used in choreographies and its level of performance is above a predefined level
- Scoring rule 3: Ratio between the number of times the service is used by the best choreographies and the total of uses

3.2 Classification of Choreographies

After selecting the service considered the best (e.g., according to the ranking of services made for each functional service scope), the elements of the monitoring system trigger mechanisms to measure the metrics of the activity of each service. The values obtained for each metric are added to tables of iterations of each service and thereafter allow measurements of the degree of excellence of the choreography as it will only be considered an *ideal* choreography if the values of the metrics are above the average value of each one. For example, the framework can point out a suitable (or excellent) choreography if 90% of the obtained values from all the measured metrics from services that integrate the choreography are above their average values. Assuming the example illustrated in Fig. 2, we can state that the average of all the known measured metrics from service "*a*" is 33% which is less than the performance obtained value (42%). The predictable performance expected for the choreography *Y* was lower (66%) than the executed one (72%) which means that when the choreography was built and run, its performance was higher than it was expected. But this does not mean that this choreography achieves an excellent performance as its performance degree



	Avg. Performance	Performance degree
predictable:	66%	86%
performed:	72%	

Fig. 2. Predictable vs. executed performance (example)

was 86%, below the conventional value of 90% which guarantees the ideal performed choreographies. If the choreography performance degree exceeds 90%, the participating services (*A* to *H*) would be increased (Scoring rule 2). According to market conditions, the choreography performance degree parameter may be adjusted to configure the appropriate answer to customer needs.

3.3 Metrics

Assessment and monitoring of each service are fundamental to create the perception of real value added to the choreography from each part of the process. Results obtained from the assessment and monitoring are useful to adapt and eventually adjust business processes. Thus, the definition of metrics of a "high level reliability of service" provides the basis for relationships of type win-win. Aspects related to the four following dimensions need to be monitored and assessed. For example, following the approach presented in [20]:

- Technology–related aspects e.g. QoS-related elements, such as: Service availability; Service response; Operation Latency; Time between failures.
- Process and product or service-related aspects, e.g.: Product/service availability; Quantity; Cost of delivery; Delivery time; Service delivery; Form of delivery; Process cycle time;
- Customer-related aspects, e.g. QoE-related elements such as Customer satisfaction, preferences and expectations; Recognizable brand; Product quality; Product variety; Level of trust; Usability; Learnability; Understandability; Operability;
- Supplier (side) of (choreography) customer service and partners-related aspects, QoBiz-related elements such as Quality of Business; QoI - Quality of Information; Cost of choreography; Revenue; Rate of return; Accuracy; Cost of goods; Completeness; Relevancy;

Metrics Definition Approach

The metrics tree definition follows the approaches described in [11][12], Business Activity Monitoring and Key Performance Indicators (KPI) that help an organization define and measure progress toward organizational goals through quantifiable measurements, agreed upon. As illustrated in Fig. 3, KPIs depend on a numerous set of Process Performance Metrics (PPM) [13] and Quality of Service (QoS) metrics [14].

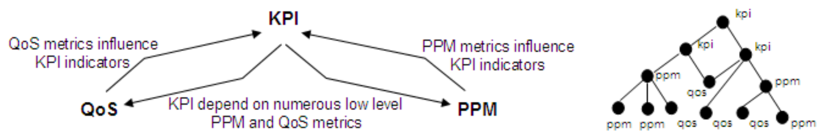


Fig. 3. KPIs, PPMs and QoS metrics(Source: adapted after[11]) and a tree dependency example

PPMs are specified based on process events (e.g. events published by a Business Process Execution Language [15] engine), whereas QoS metrics are measuring technical characteristics of the underlying service infrastructure (such as availability). Due to the correlation between metrics of different layers [16][17], the measurement results of one layer might impact the results of other layer. An example of this dependency is the "Customer Satisfaction" KPI defined on BPM layer, which is influenced by PPM metrics such as "Proposal Delivery Time", which is in turn affected by the technical QoS metrics such as the "Availability" of the external services used by the provider for placing the proposal. It is essential to design a tree of dependencies between metrics of different levels to monitor the KPIs within the cross-layer setting based.

4 Proposed Framework

The research work pursued allowed the elaboration of a conceptual framework to support the high reliability of a service.

4.1 Main Elements

The main elements of the proposed framework are presented in Fig. 4 and described in Table 1.

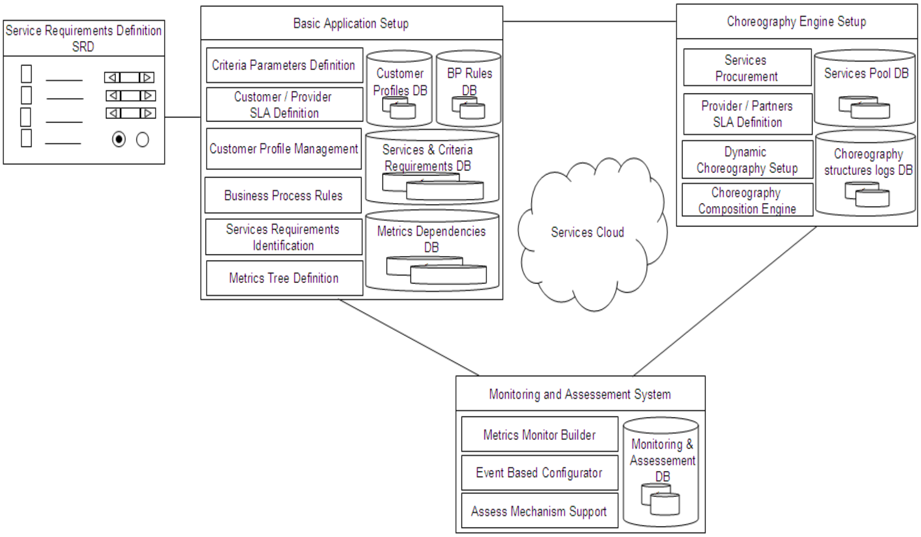


Fig. 4. Framework to support metrics of "high reliability of service"

Table 1. Description of framework elements

	1. Application Support SETUP (Application Basic setup) module	2. Choreography Engine Setup Module	3. Monitoring and Assessment Module
Main objectives	• Define the information basic structure, e.g., from customer selected criteria, through the segmentation of customer profile, services aspects identification and metrics definition.	• Identify and select from the cloud the best scored services and mount the choreography service based.	• Define a dynamic event based monitoring and assessment engine to assess the metrics tree upon the service acquired by customer.
Inputs	• Data from customer	• SLA parameters and guidelines • Services aspects and functions to be accomplished by the procurement application	• Generic metrics tree • Definition of services' choreography
Functional description	• Criteria Parameters Definition: Service Requirements Definition parsing and transformation into structured criteria parameters. • Provider builds a SLA considering input parameters • Definition of customer profile: if the customer is already known, Customer Profile Management will fulfill the Service Requirements Definition with a purpose based on the customer profile and decisions taken in previous interactions. According to the customer preferences and the level of service chosen, it will help framing and adjusting the system metrics tree to be launched over the choreography. • Business Process Rules: Through the knowledge of business rules and their requirements the required services and their level(s) will be identified. The functional scope of each set of services will receive instructions from the business rule to be organized in pools so that they can "compete" based on their functional behavior. • With the input received from the customer and Business Process Rules, Services Requirements are defined and required generic services and their characteristics are listed. • The generic metrics tree is constructed considering the client's input and metrics dependencies. All criteria and parameters (and their ranges) should be previously defined, e.g., to be able to list a set of metrics for each criteria. A database should provide a relationship between the parameters defined by the customer and the metrics necessary to monitor the service so that it is supplied with the closest approach of excellence of quality.	• Services Procurement: - the services with the best performance indicators are selected from the monitoring and assessment database based on the information received from the Services Requirements Identification and customer profile. The scoring algorithm to classify services performances in previous interactions support the service choreography engine to dynamically mount the services better ranked of the database. - in case a particular needed service is not stored in the database or the service last interaction is outdated or its ranking classification is below the required service level, services in the cloud will be procured (e.g., using a benchmarking approach) in order to fulfill the request • Dynamic Choreography Setup: according to rules of the Business Process (from the Basic Application setup) and Services Procurement Application, the choreography of services will be dynamically defined. • Choreography Composition Engine supports the choreography definition and instantiation. The monitoring and assessment system is triggered to activate the measurement of the metrics tree previously defined.	• Metrics Monitor Builder: based on the definition of the generic metrics tree and services' choreography, the metrics monitor builder defines the monitoring system. • Event Based Configurator: a set of events supports metrics' monitoring that is triggered by the metrics tree and services selected. • Assess Mechanism Support: tool based on the metrics monitor that collects the data by event triggered from each measurement and sends it to the database.
Outputs	• Identification of services and their characteristics • SLA parameters and guidelines • Generic metrics	• Definition of a SLA between the Provider and each Partner • Definition of the choreography of services	• Records data measured from each metric linked to services on database

4.2 Methodology

The proposed approach comprises four main steps:

1. **Pre-Selection:** During the pre-selection step the setup for the whole framework is prepared. Structures of meta-data based on the inter-dependencies between the criteria chosen, the services required to support the business process rules and its requirements, and the ideal set of metrics to optimize the assessment mechanism are "initialized" here. Based on the input data from customer and the definitions stored in the databases, the framework shall support dynamic interaction of components. The setup information built on this step will be used in the following step.
1. **Selection:** During the Selection phase, the instances values from the services cloud or databases are identified and selected in order to fulfill the structures layouts received from the previous step. Aligning the choreography setup with the identification of the services that will be invoked on execution step, and the tree metrics system setup in the previous step will take place.
2. **Execution:** After all the setup tasks are concluded and all inputs are gathered, the Choreography Composition Engine will launch the choreography of the selected services and, at the same time, the Assess Mechanism Support will be triggered to collect values from the metric's tree. The choreography is based on a dynamic environment, e.g., in each interaction with a customer, a different choreography is set. SLAs will be formalized.
3. **Post-Execution:** This step is related to a learning process. It relies on the data collected which is fundamental to enrich the framework for future interactions. A score algorithm will rank the services performance and assigns a value to the choreography behavior, so that it can be used in future, supporting the selection of the best services and suitable choreography at given customer criteria.

4.3 A Simple Scenario Description

A simple scenario from the automotive sector is illustrated in Fig. 5. This sector is currently facing severe difficulties, e.g., due to the global economic crisis. More than ever, it is important to establish and maintain a lasting relationship with the customer, ensuring them that the company is offering the best service available in the market, allowing improved levels of competitiveness. In this scenario, a customer requests a Car Maintenance Operation (CMO) on a Web portal and has to select a set of operational business requirements which may depend on vehicle characteristics, type of mechanical maintenance and customer preferences. It represents a set of customer criteria of how and when the CMO will be performed. The customer has to select also available date and time (e.g., from an agenda, which can be approximate or exact); operation duration (which can be approximate or exact); and type and origin of parts (from the brand, other manufacturers or white brand); hypothetical substitution car and financial loan. A parts list is then identified and their availability is checked. In case of stock rupture, market is queried to provide the necessary services. If the car maintenance takes longer than a certain period of time, the proposal to the client includes an option for vehicle replacement. If the CMO cost exceeds a certain value, the proposal includes a financial loan.

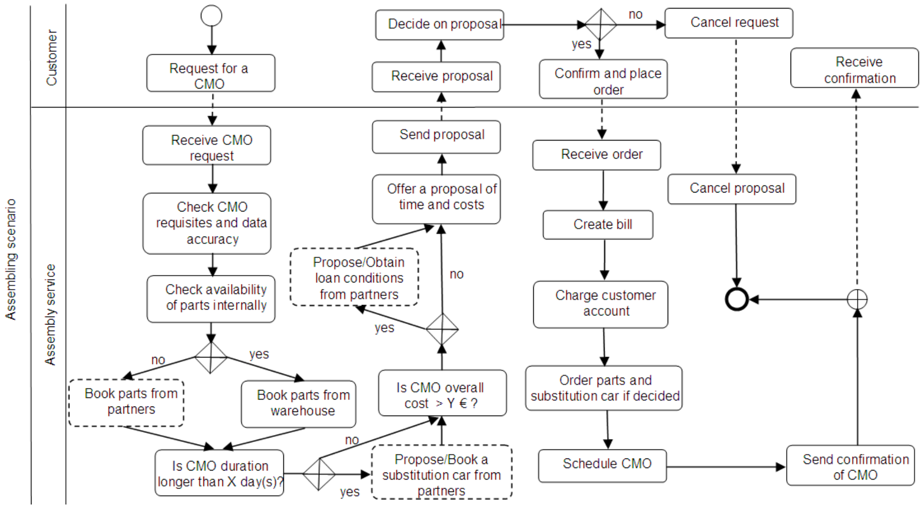


Fig. 5. A simple scenario: Car Maintenance Operation (CMO)

In these cases the market is also queried to provide the necessary services. This set of criteria provides data to the Service Requirements Definition and is needed to identify the customer profile, which in turn will help to decide what type of metrics and services should be proposed. The basic application gathers the customer data and, with support of the Business Process Rules DB, will identify in abstract the needed services to be composed to accomplish the intended level of the customer request. The Metrics Tree definition application, based on the specific business process rules and the service requirements definition, invokes an optimization engine to build the generic metrics tree according to the Metrics dependencies DB.

Based on the knowledge acquired from the learning process, the proposed metrics tree presents the list of interdependent metrics which are characterized to be the most adequate metrics that will measure the whole features of the service ensuring that the choreography is built with the best ranked services. Then, on the Choreography Engine Setup element the generic aspects of services are atomically identified, whether as a result of benchmarking procurement on the cloud of services, or they result as they are best ranked on the pool of services database and are valid for reuse. A monitoring and assessment engine is invoked to monitor and assess the choreography built based on the metrics tree. The results from the metrics measures mechanism will be relevant to assess SLA parameters and will feed the pool of services with a score that ranks the services in the pool. The choreography will also be scored. The results of several executions will always be subject to continuous improvement until it is obtained a stage of full guarantee of high level results.

In order to provide a competitive CMO in the market, several factors need to be considered. Firstly, being a business process built on a choreography of services, involving external business partners, to achieve the objectives of the business proposal one of the characteristics of the choreography refers to the dynamics of its design [8][9][10] so that it can be adjusted to the best service conditions to be proposed to the customer. At a given time, a set of suppliers can provide the best response to a customer request which may be different in other conditions, in particular, according to the characteristics of the vehicle which is involved in the service scheduling, customer preferences and/or market conditions (such as: prices, availability). The choreography should be rebuilt according to customer inputs and composed based on the procurement of services in the cloud or based on the knowledge DB from the framework, following a predefined set of criteria that will support building the choreography with the high level reliability services. The SLA generated should be closely monitored.

5 Related Work

Several approaches for service monitoring and business process monitoring exist, differing mostly in monitoring goals. An approach towards monitoring WS-BPEL processes focusing on runtime validation is presented in [18], focusing on the identification of services that deliver unexpected results (e.g., considering their functional expectation), and not on monitoring process performance metrics. A monitoring approach for WS-BPEL processes that supports run-time checking of assumptions under which a business partner services are expected to participate in a business process and the conditions to be fulfilled is described in [19].

In [20] are analyzed quality characteristics of metrics according to three perspectives: the service itself (without the customer or the business point of view), which is related to QoS - attributes like availability and performance of the service; the QoE that involves metrics which help to measure the customer interactivity (which could reflect subjective results under different occasions or customers), usability and trust; the Quality of Business (QoBiz) which is related to metrics that measure the business activity – e.g., revenue, profit. In [6] are described different quality attributes that are important to Service Based Applications like QoS, QoE and Quality of Information (QoI). Research on automated SLA negotiation between services has produced architectural solutions for negotiation on the fly as in [21].

An integrated framework for run-time monitoring and analysis of the performance of WS-BPEL processes is advanced in [22]. The authors present a dependency analysis, a machine learning based analysis of process performance metrics, and QoS metrics, in order to discover the main factors influencing the process performance (e.g., KPI) which is different from the approach described in this article, where the metrics tree results add values to the knowledge database which feeds the matrices of ranking between services.

A monitoring, predicting and adaptation approach for preventing KPI violations of business process instances is presented in [23]. A decision tree learning to construct classification models (which are then used to predict the KPI value of an instance while it is still running) is also discussed. The monitoring and assessment approach of this article is not focusing on adaptation, but on potential penalties or benefits related to service choreography.

6 Conclusion and Future Work

COBP monitoring and assessment is a challenging task. In this paper, the work pursued towards the development of a conceptual framework and metrics to support high level reliability of services in a business environment that was presented. The proposed framework comprises three elements: Application Support Setup; Choreography Engine Setup; and Monitoring and Assessment module. It is built on a set of specific metrics that translates the high reliability of a service. The metrics supporting service and business process monitoring and assessment are calculated at different levels of the choreography, focusing on four main dimensions: technical capacity and performance, product or service purchased, customer and provider's satisfaction prospect and business partners' choreography. A case example from the automotive sector was also discussed.

Future work will focus on defining new metrics to support COBP monitoring and assessment. Recurring to service composability, a prototype will be implemented to validate this approach. The proposed framework will be formally modeled to be adapted to computer-reading. Future work will also focus on preventing and real-time correction of unexpected behavior of services at run-time.

References

1. Papazoglou, M.P., Traverso, P., Dustdar, S., Leymann, F., Krämer, B.J.: Service-oriented computing: A research roadmap. In: Service Oriented Computing (SOC). Dagstuhl Seminar Proceedings, vol. 05462. IBFI, Schloss Dagstuhl, Germany (2006)
2. Di Nitto, E., Karastoyanova, D., Metzger, A., Parkin, M., Pistore, M., Pohl, K., Silvestri, F., Van den Heuvel, W.-J.: S-Cube: Addressing Multidisciplinary Research Challenges for the Internet of Services. In: Tselentis, G., et al. (eds.) Towards the Future Internet. IOS Press (2009)
3. Papazoglou, M.: Service-Oriented Computing: Concepts, Characteristics and Directions. In: 4th International Conference on Web Information Systems Engineering (WISE 2003) (2003)
4. Papazoglou, M., Traverso, P., Dustdar, S., Leymann, F.: Service-Oriented Computing: State of the Art and Research Challenges. IEEE Computer Society (2007), 0018-9162/07
5. Bichler, M., Lin, K.J.: Service-oriented computing, IT System Perspectives. IEEE Computer 39(3) (2006)
6. S-Cube consortium deliverable, CD-JRA-1.3.3 "Initial Concepts for Specifying End-to-End Quality Characteristics and Negotiating SLAs" (2010), `http://www.s-cube-network.eu/` (September 2011)
7. Keller, A., Ludwig, H.: WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services. Journal of Network and Systems Management (2003)
8. Mukhija, A., Rosenblum, D.S., Foster, H., Uchitel, S.: Runtime Support for Dynamic and Adaptive Service Composition. In: Wirsing, M., Hölzl, M. (eds.) SENSORIA Project. LNCS, vol. 6582, pp. 585–603. Springer, Heidelberg (2011)
9. Schmidt, R.: Web Services Based Architectures to Support Dynamic Inter-organizational Business Processes. In: Jeckle, M., Zhang, L.-J. (eds.) ICWS-Europe 2003. LNCS, vol. 2853, pp. 123–136. Springer, Heidelberg (2003)

10. Di Nitto, E., Ghezzi, C., Metzger, A., Papazoglou, M., Pohl, K.: A journey to highly dynamic, self-adaptive service-based applications. Automated Software Engineering Journal 15(3), 313–341 (2008) ISSN: 0928-8910
11. Wetzstein, B., Leitner, P., Rosenberg, F., Brandic, I., Dustdar, S., Leymann, F.: Monitoring and Analyzing Influential Factors of Business Process Performance. In: Published in EDOC 2009 Proc. of the 2009 IEEE (EDOC 2009) (2009)
12. S-Cube consortium deliverable, PO-JRA-1.3.1 "Survey of Quality Related Aspects Relevant for Service-based" (2008), http://www.s-cube-network.eu/ (September 2011)
13. Wetzstein, B., Strauch, S., Leymann, F.: Measuring Performance Metrics of WS-BPEL Service Compositions. In: The Fifth International Conference on Networking and Services (ICNS 2009), Valencia, Spain, April 20-25. IEEE Computer Society (2009)
14. Menasce, D.A.: QoS issues in Web services. IEEE Internet Computing (2002)
15. Karastoyanova, D., Khalaf, R., Schroth, R., Paluszek, M., Leymann, F.: BPEL Event Model. Technical Report 2006/10. University of Stuttgart (2006)
16. S-Cube consortium deliverable, PO-JRA-1.2.3 "Baseline of Adaptation and Monitoring Principles, Techniques, and Methodologies across Functional SBA Layers" (2009), http://www.s-cube-network.eu/results/deliverables/ (September 2011)
17. Kazhamiakin, R., Pistore, M., Zengin, A.: Cross-Layer Adaptation and Monitoring of Service-Based Applications. In: Dan, A., Gittler, F., Toumani, F. (eds.) ICSOC/ServiceWave 2009. LNCS, vol. 6275, pp. 325–334. Springer, Heidelberg (2010)
18. Baresi, L., Guinea, S.: Towards Dynamic Monitoring of WS-BPEL Processes. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 269–282. Springer, Heidelberg (2005)
19. Barbon, F., Traverso, P., Pistore, M., Trainotti, M.: Run-time monitoring of instances and classes of Web servisse compositions. In: Proc. IEEE Int. Conf. on Web Services (ICWS 2006), pp. 63–71 (2006)
20. van Moorsel, A.: Metrics for the Internet Age: Quality of Experience and Quality of Business. In: Proceedings of the 5th Int'l Workshop on Performability Modeling of Computer and Communication Systems, Erlangen, September 15-16. Arbeitsberichte des Instituts für Informatik, Universität Erlangen-Nürnberg (2001)
21. Chhetri, M., Lin, J., Goh, S., Zhang, J., Kowalczyk, R., Yan, J.: A coordinated architecture for the agentbased service level agreement negotiation of Web service composition. In: Australian Software Engineering Conference (ASWEC 2006), pp. 90–99 (2006)
22. Wetzstein, B., Leitner, P., Rosenberg, F., Brandic, I., Dustdar, S., Leyman, F.: Monitoring and Analyzing Influencial Factors of Business process performance. In: Proc. 2009 IEEE (EDOC), pp. 141–150 (2009)
23. Wetzstein, B., Zengin, A., Kazhamiakin, R., Marconi, A., Pistore, M., Karastoyanova, D., Leymann, F.: Preventing KPI violations in business processes based on decision tree learning and proactive runtime adaptation. Journal of Systems Integration 3(1), 3–18 (2012)