



**HAL**  
open science

## Performance Oriented Decision Making to Guide Web Service Lifecycle

Tehreem Masood, Chantal Cherifi, Néjib Moalla, Muhammad Fahad

► **To cite this version:**

Tehreem Masood, Chantal Cherifi, Néjib Moalla, Muhammad Fahad. Performance Oriented Decision Making to Guide Web Service Lifecycle. Enterprise Interoperability VII , Mar 2016, Guimarães, Portugal. pp 113-122 10.1007/978-3-319-30957-6\_9 . hal-01470788

**HAL Id: hal-01470788**

**<https://hal.science/hal-01470788>**

Submitted on 17 Feb 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Performance Oriented Decision Making to Guide Web Service Lifecycle

---

Tehreem Masood<sup>1</sup>, Chantal Bonner Cherifi<sup>1</sup>, Néjib Moalla<sup>1</sup>, Muhammad Fahad<sup>1</sup>,

<sup>1</sup> Decision and Information for Production Systems (DISP), Université Lumière Lyon 2, France

**Abstract.** Service composition networks are becoming increasingly important and more critical for the enterprise collaborations. The sustainability of service based systems is driven with the support of performance metrics analysis for decision making. Some of the existing works only focused performance at messaging level. Ontology based performance monitoring techniques did not provide the recommendation model for better quality of service. In this paper we propose a semantic service recommendation framework (SSRF) based on Service Networks Monitoring (SNM) concepts. This framework includes creation of Service Network Monitoring Ontology (SNMOnt), extracting traces of technical indicators for services and storing decision data in the existing database. The decision support covers the unavailability of service and recommendation of services based on technical indicators. Our framework is meant to support the creation of shared domain knowledge ensuring an optimal decision making system for web service lifecycle.

**Keywords:** Service Network Monitoring, Decision Support, Ontology, Performance, Technical Indicators.

## 1 Introduction

Service-Oriented Architecture (SOA) is an emerging cross-disciplinary paradigm for the distributed and service oriented computing [1]. One of the challenges in SOA is to provide an efficient way to discover compatible Web services and compose them into business process. It is important to create dynamically scalable business processes and guarantee their performance, availability, reliability and throughput to be used by internal and external partners. Selecting the most significant web service according to a consumer requirement is a challenging task. Technical indicators needs to be considered when there are several functionally similar web services.

Semantic web uses the notion of ontology's for the creation and elicitation of domain knowledge [2]. Ontology's represent formal specifications about the component of systems and their relationships in a machine understandable and processable manner [3]. They have played an important role in both semantic web applications and knowledge engineering systems [4]. Several tasks such as information storage, processing, retrieval, decision making etc. are done on the basis of ontologies by such systems.

The problem of providing seamless decision making capabilities based on the new project requirements of users is very important. There are different levels of performance in the service network monitoring like binding level, service level, server level and business process level. Business process level is used for the

performance analysis at the composition of service [5]. Binding level covers the performance at the protocols level [6]. Service is composed of different operations. Performance can be monitored at both the service and different operations level. Server level covers the performance of resources available. Current research support specific activities, but do not fulfil the requirements of all the activities in the Service-based System's lifecycle. Different technologies are available that can be effectively utilized for better performance of services. Now-a-days, WSO2 application server is widely being used for the deployment of web services. It has sophisticated infrastructure to capture statistics at operation, service and server level [7].

Our objective in this research is to provide decision support combining service/process technical indicators in order to generate validation arguments for the expansion of the service environment. Therefore in this paper, we propose SSRF based on the SNM. We provide decision support that covers the unavailability of services and recommendation of services based on technical indicators. The proposed model integrates the following building blocks. The first block is the identification and classification of performance metrics from Information Technology Infrastructure Library (ITIL) [8]. Second block is to structure the performance based technical indicators for atomic service, business process, server and binding. For this purpose we design a sophisticated SNM Ont as a base infrastructure. It aggregates the main concepts and relationships between them. QoS requirements, service domain concepts, technical indicators and performance levels are the major concepts. Third block is the decision support based on some inference rules like availability, reliability, throughput, memory available and delay.

The remaining of the paper is organized as follows: Section 2 includes related work of different frameworks. Section 3 discusses our proposed approach. We conclude the paper in the last section.

## **2 Related works**

There are different categories of techniques available that can be effectively utilized in order to provide better performance of service-oriented networks. Therefore related work section has two dimensions. First, we study techniques related to the performance issues of web services. Second, we include ontology based QoS monitoring techniques. Third, we provide analysis of techniques of both sections.

### **2.1. Performance based Web services Techniques**

In this category we will discuss some techniques that have been proposed to ensure performance. While analyzing these techniques, it has been observed that all these techniques only focused the binding level performance.

Tari et al. proposed a benchmark of different SOAP bindings in wireless environments. Three sets of experiments were carried out: loopback mode, wireless

network mode and mobile device mode. The experimental results show that HTTP binding inherits very high protocol overhead (30%–50% higher than UDP binding) from TCP due to the slow connection establishments and tear-down processes and the packet acknowledgement mechanism. UDP binding has the lower overhead because it does not require establishing connections before transmitting datagram's and does not address reliability. This results in a reduction in the response time and an increase in the total throughput. Its configuration and results can serve as a standard benchmark for other researchers who are also interested in the performance of SOAP bindings in wireless networks [9].

Tari et al. proposed a similarity-based SOAP multicast protocol (SMP) which reduces the network load by reducing the total generated traffic size. It is based on the syntactic similarity of SOAP messages. In particular, the SMP reuses common templates and payload values among the SOAP messages and only sends one copy of the common part to multiple clients. SMP makes use of the commonly available WSDL description of a SOAP Web service when determining the similarity of response messages. For messages that are highly similar, instead of generating messages with duplicated similar parts for different clients, the duplicated parts are reused for multiple clients and are sent only once from the source [10].

Tekli et al. conducted an extensive review of XML processing in the context of SOAP-based web services. The review analysed different techniques to improve performance in XML serialization, parsing and deserialization. Delay distributions model the distribution of waiting times that represent the time needed to complete a given task, such as, the time necessary to execute an operation, to parse an XML input message, or to build a SOAP output message. Distributions of waiting times are specified using the results of the service testing on the total operation execution time [11].

## **2.2. Ontology based QoS Monitoring Techniques**

We have narrow down our related work with respect to ontology because of its various advantages. Several tasks such as information storage, processing, retrieval, decision making etc. are done on the basis of ontologies by such systems. In this section we list ontology based research that focuses on quality of services.

Lin et al. proposed an ontology based QoS-Aware support for semantic web services. They have used ontology to describe Quality of service metrics. They have composed their ontology into upper and lower level property. They have not considered the input/output operations. They have not included real time values for the performance analysis to reach towards optimal web service [12].

Moraes et al. designed an ontology named MonONTO for proposing recommendation for the advanced internet applications users. They have considered information concerning the application type, traffic generated and user profile along with network performance metrics. Their expert system monitors the performance of advanced internet applications. Their ontology serves as a support to a decision reference tool by providing high-level information to the user about the agreement of the network facing the service level demands. They have used a

fixed list of network parameters. Therefore, it does not deal with the heterogeneity and extensibility issues. Implementations of web services have not been done by them. Additionally, it does not deal with quality of service mapping [13].

Benaboud et al. proposed Semantic Web Service Discovery Based on Agents and Ontologies considering the fuzzy constraints. Their framework is modeled by adding semantics of QoS attributes with web service profiles. It describes the design and implementation of a web service matchmaking agent. Agent uses an OWL-S based ontology and an OWL reasoner to compare ontology based service descriptions. They have used fuzzy constraints increases the efficiency of the web service discovery approach by providing the customers the web services which are not actually satisfying the input QoS constraints, but are close to the QoS constraints specification [14].

### 2.3. Analysis

The experimental results of the first technique of section 2.1 shows that HTTP binding acquires very high protocol overhead in terms of response time from TCP due to the slow connection establishments. UDP binding has the lower overhead in terms of response time because it does not require establishing connections before transmitting datagram's. Results for the second technique of section 2.1 shows that SMP has medium response time, throughput and network traffic. The third technique of section 2.1 is a survey of XML processing in the context of SOAP-based web services. All these techniques are related to the performance of web service at binding level which we can utilize in proposed framework. The focus of our research work is at the infrastructure level covering service, server, binding and business process level.

Section 2.2 covers the techniques that are related to ontology based QoS monitoring. First technique of this section has not considered the input/output operations. Second technique of this section is related to the ontology of performance at the network traffic level. The third ontology based technique is specific to the monitoring of agent based systems.

To be effective, QoS evaluation approaches should rely on models representing the systems in an accurate/realistic way, and whose parameters can be adjusted at run-time according to measured data. Therefore we propose a semantic service recommendation framework that infers decision support at the infrastructure level. Infrastructure level covers the service, server, binding and business process level.

## 3. Proposed Approach

This section presents SSRF to recommend the services based on technical indicators requested by the consumer. Figure 1.1 illustrates SSRF. First of all, user send request to consume existing service. Second step has two parts. First part is to get traces or events from application server. First we deploy services and BPEL in the WSO2 server. Server stores values of technical indicators in some database. Second part includes the design and instantiation of SNMOnt by storing the values

of technical indicators. It aggregates the main concepts and relationships between them. Third step is the decision support. Decision support constitutes some inference rules. Inference rules includes availability, reliability, throughput, memory available and delay. Based on these rules, proposed SSRF generates some recommendations like reuse existing service, service is not available and physical resource is not available.

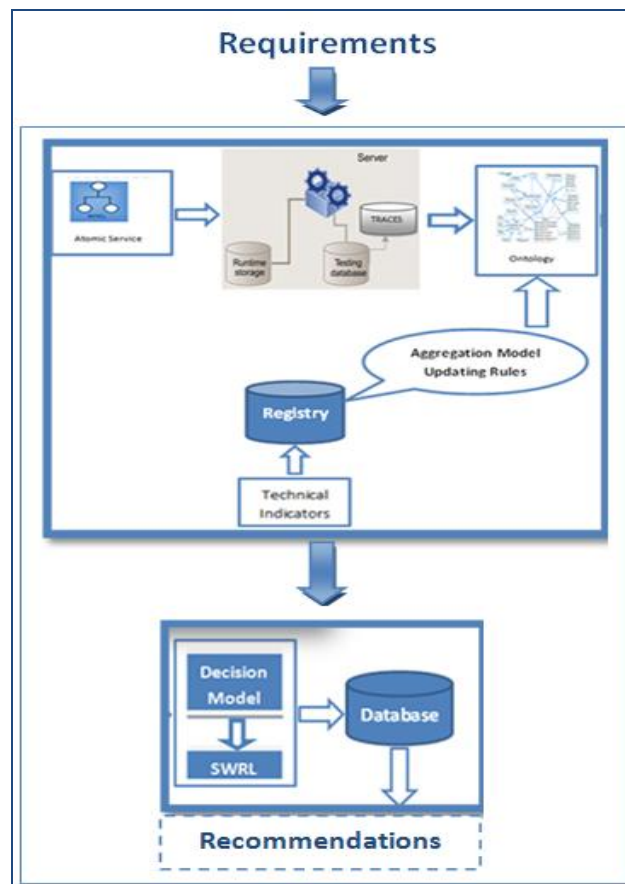


Fig.1.1. Semantic Service Recommendation Framework (SSRF)

### 3.1. Service Network Monitoring Ontology (SNMOnt)

We define an ontology named as SNMOnt in order to store the traces of events gathered from the execution of service/business process in WSO2 server. We use some concepts from the literature as well as define some new concepts. Fig 1.2 presents SNMOnt.

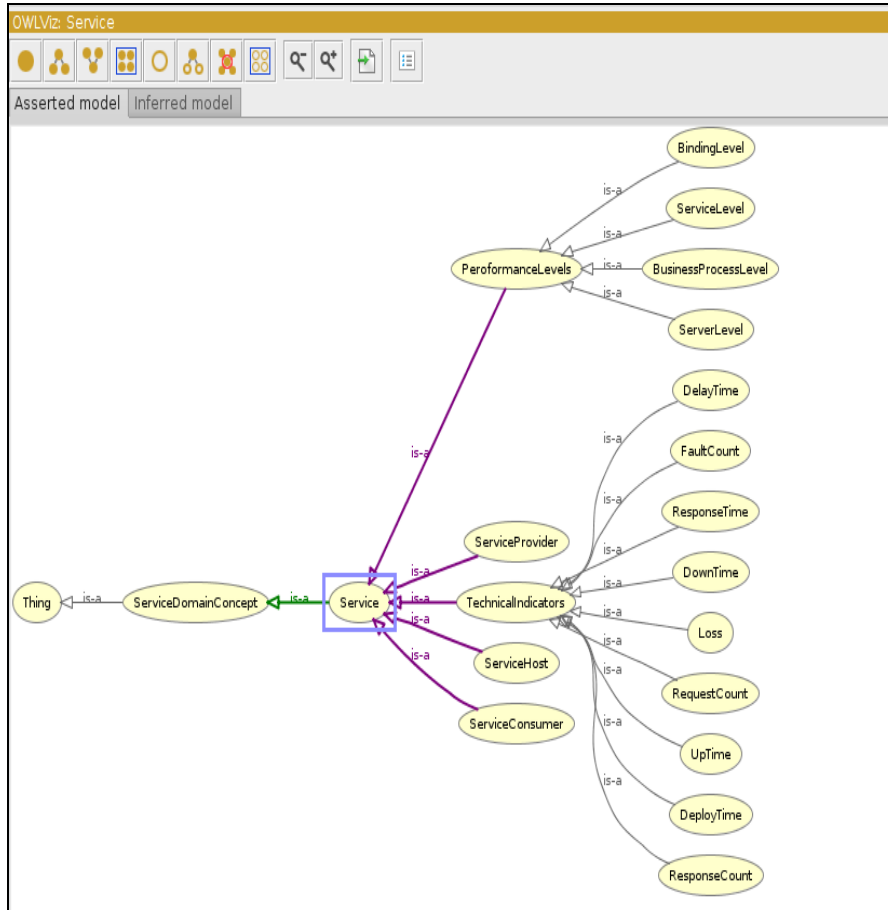


Fig.1.2. Service Network Monitoring Ontology (SNMONT)

Service provider, service consumer and service host are the concepts that have already been used in the literature. Technical indicators like response time and delay are also used [15]. But the technical indicators that we show in our ontology are specific to WSO2 server. Further we classify different performance levels. Performance Level concept uses the level where service network can be monitored. It has various sub concepts such as binding level, service level, business process level, and server level.

Each of the service and its operations are monitored. Response Time captures the response time of a service/process. It has three sub concepts to record Maximum, Minimum and Average response time. Request Count shows the number of invocation of a service. Response Count shows the number of replies for an invocation of a service. Fault Count shows the number of invocations the service

has not replied. Deploy Time shows when the service is deployed at the server. Up Time shows the time period the service is available since its deployment. Down Time shows the time period of un-availability of a service since its deployment. Delay shows the average response time of a service. Loss shows that the service is un-available.

Data properties that are used to link concepts are shown in Fig 1.3. Consumed\_By relationship captures the relationship of each service consumed by its consumers. Has\_Indicators relationship describes that performance of each service is monitored via various technical indicators. Hosted\_By relationship captures the relation of each service hosted by its Host. Measured\_At relationship describes that each service is monitored and performance is measured at the Performance\_Level. Provided\_By captures the relation of each service provided by its provider.

Concepts	Relationships
Service Consumer	Consumed By
Service	Consumes
Service	Has Indicators
Service Host	Hosted By
Performance Levels	Measured At
Service	Measures
Service Provider	Provided By

Fig.1.3. Object Properties of SNMOnt

We design our SNMOnt in Protégé 5 and uses jess reasoner to verify its consistency to avoid all types of ontological error [16]. Protégè structure permits the addition of plug-ins to interact with other tools. Its graphical capabilities can draw concept trees and ontology flow diagrams that help in visualization and generates OWL code automatically.

### 3.2. Decision Support

We use SNMOnt concepts values and generate inference rules. SNMOnt concepts are total response count, total request count, service deployed time, service up time, memory allocated, memory usage, no of successful invocations, total no of invocations, total response count and average response time. Inference rules are for the availability, delay, memory available, reliability and throughput. Following is the description of these rules. Availability is calculated as the total response count of a service/business process divided by total request count of a service/business process. If the requested service is available then we can reuse this service. Delay is measured as the difference of service deployed time and service up time. Memory available is measured as the difference of memory usage and memory



allocated. Reliability is calculated as number of successful service operation invocation divided by total number of service operation invocations. Throughput is calculated as the total response count divided by average response time. Figure 1.4 shows these inference rules that compose the knowledge organization originated from the relationships between SNMOnt concepts.

Technical Indicators	SWRL Rules
<b>Calculate_Availability</b>	swrl:divide(Total_Response_Count,Total_Request_Count,?x) -> Availability(?x)
<b>Calculate_Delay</b>	sqwrl:difference(Service_Deployed_Time,Service_Up_Time,?x) ->Delay(?x)
<b>Calculate_Memory_Available</b>	sqwrl:difference(Memory_Allocated,Memory_Usage,?x) ->Available_Memory(?x)
<b>Calculate_Reliability</b>	swrl:divide(no_of_successful_invocations,total_no_of_invocations, ?x)-> Reliability(?x)
<b>Calculate_Throuput</b>	swrl:divide(Total_Response_Count,Average_Response_Time,?x) ->Throughput(?x)

Fig.1.4 Inference Rules using SWRL

We use aggregate QoS functions (like divide, difference) that can be easily defined and managed. Based on these rules we recommend that “the selected service or business process is available or not”, “reuse existing service”, “delay of a deployed service or business process”, “memory is available or not to support the new deployment”, “reliability of a service or business process in terms of service operation invocations” and “throughput of a service or business process”. We store decision data in existing database to deploy decision. We define these rules using SWRL (Semantic Web Rule Language) [17] in Protégé. Future recommendations will include “service composition is not possible” and “security compliance problem in the deployment”.

#### 4. CONCLUSION

In this paper, we focus on how to accomplish and improve enterprise business and IT together. We propose a framework named as SSRF based on the SNM. This framework includes the extraction of traces of technical indicators for services or business process, design of ontology named as SNMOnt, instantiating ontology by values of technical indicators, providing decision support based on inference rules and storing decision data in the existing database. Proposed framework generates recommendations like service availability, reuse existing service, delay of a deployed service, memory is available or not to support the new deployment and reliability in terms of service operation. The use of the framework aims at increasing particularly the alignment of the requirements and of the objectives of the enterprise business and IT in decision making. We will enhance our work to

cover the recommendation of service composition and security compliance problem. We will implement our work by using real time case study.

## REFERENCES

- [1] Gottschalk, K., Graham, S., Kreger, H., Snell, J.: Introduction to web services architecture. IBM Systems Journal, 2002, pp 170–177.
- [2] Antoniou, G., and Harmelen, F.V. A Semantic Web Primer. MIT Press Cambridge, 2004, ISBN 0-262-01210-3.
- [3] Fahad, M, and Qadir, M. A., “A Framework for Ontology Evaluation.” ICCS Supply, 2008, vol. 354, pp 149–158.
- [4] Gomez-Perez, A., Lopez, M.F, and Garcia, O.C. Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web. Springer ISBN: 2001, 1-85253-55j-3.
- [5] Documents Associated with Business Process Model and Notation (BPMN) Version 2.0, Release date, January 2011, <http://www.omg.org/spec/BPMN/2.0/PDF>
- [6] Simple Object Access Protocol (SOAP) 1.2, Part 0, Primer: (2007) – World Wide Web Consortium, <http://www.w3.org/TR/soap12-part0/>
- [7] <http://wso2.com/products/application-server/>
- [8] Donna K, A Guide to Service Desk Concepts: Service Desk and the IT Infrastructure Library, 4th edition, Course Technology Press Boston, MA, United States ©2014, ISBN- 10: 1285063457 | ISBN-13: 9781285063454
- [9] Tari, Z., Phan. A. K. A., Jayasinghe. M., Abhaya. V. G, “Benchmarking Soap Binding. On the Performance of Web Services,” pp 35-58, Springer 2011.
- [10] Tari, Z., Phan. A. K. A., Jayasinghe. M., Abhaya. V. G, “The Use of Similarity & Multicast Protocols to Improve performance,” On the Performance of Web Services, Springer pp 59-104, 2011
- [11] Tekli, J., Damiani, E., Chbeir, R., Gianini, G. SOAP processing performance and enhancement. *IEEE Trans. Serv. Comput.* 2011, 5, 387–403.
- [12] Lin, L., Kai, S., and Sen, S. “Ontology-based QoS-aware support for semantic web services.” Technical Report at Beijing University of Posts and Telecommunications, 2008.
- [13] Moraes, P., Sampaio, L., Monteiro, J., Portnoi M. Mononto: A domain ontology for network monitoring and recommendation for advanced internet applications users. In: Network Operations and Management Symposium Workshops, IEEE 2008.
- [14] Benaboud, R., Maamri, R., and Sahnoun, Z., Semantic Web Service Discovery Based on Agents and Ontologies, International Journal of Innovation, Management and Technology, 2012, Vol. 3, No. 4, pp 467-472
- [15] Chhun, S., Cherifi, C., Moalla, N., Ouzrout, Y. A multi-criteria service selection algorithm for business process requirements. 2015, CoRR abs/1505.03998
- [16] Protégè. Stanford Medical Informatics. Available at: <http://protege.stanford.edu/>, 2007.
- [17] W3C. SWRL: Semantic Web Rules Language. <http://www.w3.org/Submission/SWRL/>, 2004.