# Dynamic Service Composition for Evaluating Services in Service-Oriented Architectures: A Review Analysis

Sarvesh[1], Naveen Bilandi[2]

*[1]M.Tech (CSE) Student, DAV University*
*Jalandhar*
*[2]Assistant Professor in Dept. of CSE, DAV University*
*Jalandhar*

**Abstract:In the recent trends Organizations face various business challenges in rapidly changing environments, often in forms of partnership changes or Merger & Acquisitions. Coping with these challenges within manageable time and budget and to deliver quality products is a high-priority goal for survival. Service Oriented Architecture (SOA) is a promising approach for handling these issues. Capable of providing mechanisms for integrating legacy systems at low cost or handing rapid business changes effectively, SOA mitigates risks introduced by fluctuations in the business world. However, there are currently few metrics suite available to evaluate the SOA solution. This paper proposes a new set of a metric suite for quantitative analysis of SOA solution. This metric suite helps in continuous monitoring and evaluation of Quality of Service (QOS) parameter for SOA.**

**Keywords: Service Oriented Architecture (SOA), web services, Quality of service, Object Oriented Design (OOD), Architectural design**

## 1. INTRODUCTION

Service Oriented Architecture (SOA) is becoming an increasingly popular architectural style that focuses on providing the right tools and methods for building distributed application. It is a promising approach for delivering a service system by rapidly composing individual services.SOA is directly built upon the principle of service-orientation, which talks about services with simple interface that can be accessed independently by users without worrying about the actual platform implementation of the service.SOA is a way of designing, developing, deploying, and managing systems[2]. Although it is called "architecture," it does not really comply with the normal definition of the word; it is more of an architectural style for enterprise systems.

SOA is a way to implement decentralized computing and e-business applications progressing from Object Oriented Design and distributed computing to enable the development of agile infrastructures of collaborating business "services" dispersed within and across organizational boundaries. This proposed model utilizes services that are sovereign, platform-independent computational components that can be described, published, discovered and accessed over the Internet using standard protocols for developing tools/applications/solutions. In this context, services become the next level of abstraction in the process of creating systems that would enable automation of businesses [3]. This shift is changing the way the computer software is designed, developed, delivered, tested, analyzed, integrated and utilized. Various forms of SOAs have emerged in the market place including Web services, Grid services, Semantic Web Services, and e -Services. Even though they share some SOA principles, they differ in many other aspects, which is an undesirable situation in the context of service-oriented architectures. All these forms of SOA have developed different conceptual models, resulting in different methodologies for modeling and designing service-oriented architectures [7]. These are challenges that must be addressed if we are to realize the promise of SOA as it was intended. SOA model requires a well-defined set of information and service interfaces between service suppliers and service consumers tied together via multiple well architected information sources and uses an enterprise service bus to coordinate traffic between suppliers and consumers. The goal is to make new services available to the bus with a minimum of effort.

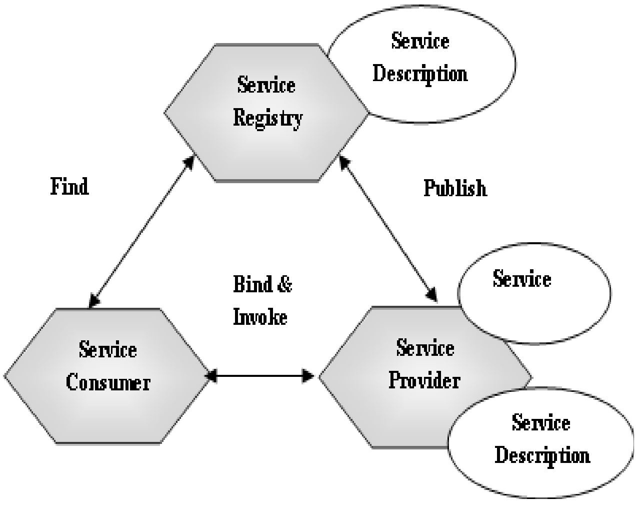

Figure1: Service Oriented Architecture [7]

Figure1 shows the interaction pattern in a service-oriented architecture, where a service consumer performs dynamic service locating by querying the service registry for a service that matches its criteria [7]. If the service exists, the registry provides the consumer with the interface contract and the endpoint address for the service. The following diagram illustrates the entities in a service-oriented architecture:

The actors in service-oriented architecture are:

*Service consumer:* A service consumer searches for a required service in a service registry. When consumer finds the required service, the service description is used to dynamically bind the provider and consumer. Then service consumer invokes the service and interacts directly with the service.

*Service provider:* The service provider defines service descriptions and publishes the descriptions of services so that services can be discovered.

*Service registry:* A service registry contains service description and references to service providers and also provide the mechanism for service discovery and publishing.

To build high quality SOA-based systems, the overall system quality needs to be explicitly managed from early stages of development. Explicit quality management applied early in development discovers signs of problems that need to be addressed to prevent waste of effort which would otherwise be magnified in later stages. The use of metrics enables us to quantify system quality and manage it explicitly, thereby providing clear signs of design anomalies to the developers. Therefore, assessment metrics should be identified and applied as early in development as possible to maximize its benefits.

## 2. RELATED WORK

Existing metrics are not directly applicable to the service oriented designs since the service oriented development paradigm is different object oriented and component-based development. Therefore some researcher proposed many metrics but only few metrics provided approach for improving the quality of service of SOA.

LathaSrinivasan et.al[1] introduced the key concepts, relationships and benefits of web services and SOA and indicated how they can be combined to develop highly scalable application systems that can span management and ownership domains, regardless of the hardware and software platforms deployed in each. They also show how the careful development of key standards is increasing the appeal of service-oriented environments, and how they are being used as the basis for development of interoperable next-generation grids.

Young kon lee et.al[2] described the quality of service metrics for service level measurement and focuses especially to the two quality factors: business value and service level measurement affecting real world business activities and a service performance.

Liam O' Brien et.al[3] presented how the different quality attributes of a system can positively or negatively affected by the use of choosing the architecture for SOA. They describe the factor related to each attribute, as well as the possible trade-offs and existing effort to achieve that quality. They also describe the open issues in service level agreements that are used to contract the level of service quality between service provider and user.

Si Won Chai et.al[4] has proposed a quality model for evaluating services in SOA regarding customer perspective. They first analyses attributes of quality model then define metrics for each quality attribute, to show the applicability and usefulness of quality model.

MamounHirzalla et.al[5] presented a technique such as Soma technique, which provide the guidance for analysis, design, implementation, testing and deployment of services, information and polices need to implement the SOA solution. But the Service Oriented Modelling Architecture (SOMA) technique does not provide any metrics for underlying complexity and flexibility of SOA solution. So they proposed a SOA metric framework which includes service level and SOA-Wide metrics to measure design and qualities of SOA solution.

SaadAlahmari et.al[6] proposed a new set of metrics for measuring the SOA internal attributes like complexity, cohesion & coupling by using the syntax code. The metrics proposed have been devised to address the key aspects of a service the business functionality and data manipulation. These aspects are considered individually together with metrics derived from service interface. These metrics will assist in the development of optimal service.

## 3. CLASSIFICATION FOR QUALITY ATTRIBUTES FOR THE DESIGNING A SOA

One of the significant challenges for making service-oriented architectures (SOA) effective for enterprise systems is quality of service (QoS) management because of the dynamic, flexible, and compositional nature of SOA. Services can be characterized as functional and non-functional attributes, while non-functional attributes are the element of QoS like performance, reliability, availability, and security etc. Each SOA service has a quality of service (QoS) associated with it.QoS differentiates functionally equivalent services. With the use of QoS the consumer can express their needs and select the best service with respect to their needs. On the other hand, providers can differentiate their services, can better respond to user needs, and can better utilize their services [5]. The dynamic composition feature in SOSE makes QoS a major challenge. So we need to implements techniques for continuous monitoring and evaluation of QoS parameters, QoS adaption in case of QoS violation and service failures, and end-to-end QoS management for composite service etc.

Table 1: Qos Attributes and Proposed Metrics

| **Quality Attributes** | **Description** | **Proposed Metrics** |
|---|---|---|
| Timeliness | The ability to respond in a minimum specified time period. | • Service Response Time |
| Availability | This quality attribute measures the degree to which service is accessible and operational when service consumer request to use | • Service Support for Transaction |
| Performance | This quality attribute measures the capability of service to provide appropriate response and processing times and throughput rates | • Throughput<br>• Number of Services |
| Accessibility | Represents the probability of which service platform is accessible while | |
| Reliability | It deals with the capability of services to provide transparent, reliable and fault tolerant services to its user | • Service Failure Ratio |
| Composability | This quality attribute measures the capability of a service composition to operate successfully by composing atomic services | • Number of Version Per Service |
| Discoverability | A service interface should use metadata to define the service capabilities and constraints. | • Service Access Method<br>• Service Realization Pattern |

Each of the elements in the definition caters for a quality service. It is important that the service suppliersinclude the ability for the user to append additional information regarding the required QoS and compliance to these principles.

## 4. Metrics suite for evaluating SOA

The data collected from SOA solution is use for the quantitative analysis. For this quantitative analysis of data we identify the different metrics [3]. On the bases of which different quality attributes of SOA solution will be measure. So that different issues affecting the SOA solution can be resolved earlier in the software life cycle. These metrics are discussed below

Table 2: Metrics and Evaluating Formula's

| **Metric Name** | **Description** | **Evaluating Formula** |
|---|---|---|
| **Weighted Service Interface Count (WSIC)** | The weighted number of exposed interfaces or operation per service on the Web Service Definition Language (WSDL) documents. | Default Weight = 1 |
| **Stateless Services (SS)** | The fraction of services which are stateless (SLS) as opposed to state full (SFS) as defined in the service resource framework. | **SS = SLS/(SLS + SFS)** |
| **Service Realization Pattern (SRP)** | The fraction of services that are realized through indirect exposure (IE) in respect to the total number of services that are realized using both IE and direct exposure (DE). | **SRM = IE / ( IE + DE)** |
| **Service Access Method (SAM)** | The fraction of services accessed using virtualization layer, referred to as Virtualized Access Services (VAS), in respect to the total number of services that are VAS or accessed point to point (PPS | **SAM = VAS / ( VAS + PPS** |
| **Dynamic vs. Static Service Selection (DSSS)** | The number of services that are selected dynamically (DS) Over the total number that are selected dynamically or statically (SS). | **DSSE = DS / (DS + SS)** |
| **Number of Versions Per Service (NOVS)** | The total number of versions over the total number of services within the SOA solution. | **NOVS = VERSIONS / SERVICES** |
| **Service Response Time (SRT)** | This metric is an elapsed time between the end of a request to a service and the beginning of the service's response [6]. It can be computed as | **SRT =** Time when Service Consumer finishes sending request to the service – Time when Service Consumer starts receiving response from the service |
| **Service Failure Ratio (SFR)** | *SFR* metric is to measure the ratio of how many services failed during a specific time interval. This metric can be calculated with the following formula. | ***SFR = Time Period / Number of Failures*** |
| **Throughput** | Throughput refers to the maximum amount of services provider can process in a given time period. The maximum number of responses which can be processed in a unit time | **Throughput = Number of requested processed by service in measured time/ Measured time** |

## 5. CONCLUSIONS AND FUTURE WORKS

The primary contribution of this paper is the proposed metrics that should be tracked and measured for every SOA engagement so that better insight can be gleaned into the complexity, agility and flexibility of the SOA solution. For some metrics there are currently some limitations in practically applying them since the current techniques for discovery, adaption and composition are not mature enough. Future work will include empirical assessment of these metrics and identifications of ones order to build a demonstrably useful set of SOA metrics for predicting QOS attributes across a broad spectrum of SOA solutions. Additional work is also required to provide clear methods for calculating the values proposed aggregate SOA indices and interpret their implementation in terms of SOA quality of service attributes.

## REFERENCES:

[1] LathaSrinivasan and Jem Treadwell,"An Overview of Service oriented Architecture, Web Services and Grid Computing ". HP Software Global Business Unit November 3, 2005.

[2] Youngkon Lee," QoS Metrics for Service Level Measurement for SOA Environment". IEEE 2006.

[3] Liam O' Brien, Paulo Merson and Len Bass, "Quality Attributes for Service-Oriented Architectures". International Workshop on System Development in SOA Environments (SDSOA2007), IEEE 2007.

[4] SI Won Choi, Jin Sun Her, and Soo Dong Kim," Modeling QOS Attributes and metrics for evaluating Services in SOA Considering Consumer's Perspective as the First Class Requirement". IEEE Asia-Pacific Services Computing Conference, 2007

[5] Mamounhirzalla, Jane Cleland and Ali Arsanjani, "A Metrics Suite for Evaluating and complexity in Service Oriented Architectures". ICSOC 2009, LNCS 5472, Pages 41-52, Springer-Verlag berlin Heidelberg, 2009.

[6] SaadAlahmari, Ed Zaluskaand David C De Roure, "A Metrics Framework for Evaluating SOA Service Granularity". IEEE International Conference on Services Computing, 2011.

[7] ShaziaUsmani,NaheedAzeem, Asia Samreen " Dynamic Service Composition in SOA and QoS Related Issues" International Journal of Computer Technology and Applications, Vol 2 (5), 1315-1321

[8] B. S. Farroha, D. L. Farroha "POLICY-BASED QOS REQUIREMENTS IN A SOA Enterprise Framework-An Investigative Analysis", IEEE 2007

**Sarvesh** completed his B.Tech. in Computer Science & Technology from Shaheed Bhagat Singh State Technical Campus, Ferozepur, Punjab (India) in 2013 and pursuing M.Tech. in Computer Science & Technology from DAV University Jalandhar, Punjab (India). He has a vast knowledge in the field of DBMS, Software Engineering, Network Security, Operating System and Formal Language and automata theory.

**Naveen Bilandi** received his Bachelor degree in Computer Science from the University of Rajasthan and Master degrees in Computer Science from NIT Jalandhar, Punjab (India) respectively in 2008 and 2012.From 2012 to 2013 he worked as assistant professor in DAVIET, Jalandhar, Punjab (India) and from 2013 till now working as a coordinator of CSE department at DAV University Jalandhar, Punjab (India). He has a vast experience in the field of Wireless Data Network, Computer Networking, DBMS, Network Security and Cryptography. His major fields are Mobile ad-hoc networks and current research interests include routing protocol and QoS of multimedia applications for MANET.

.