Performance Prediction of Service-Oriented Architecture - A survey

Haitham A. Moniem,
College of Graduate Studies,
Sudan University of Science and Technology,
Khartoum, Sudan

Hany H Ammar,
Lane Department of Computer Science and
Electrical Engineering,
College of Engineering and Mineral Resources,
West Virginia University
Morgantown, USA

Abstract: Performance prediction and evaluation for SOA based applications assist software consumers to estimate their applications based on service specifications created by service developers. Incorporating traditional performance models such as Stochastic Petri Nets, Queuing Networks, and Simulation present drawbacks of SOA based applications due to special characteristics of SOA such as lose coupling, self-contained and interoperability. Although, researchers have suggested many methods in this area during last decade, none of them has obtained popular industrial use. Based on this, we have conducted a comprehensive survey on these methods to estimate their applicability. This survey classified these approaches according to their performance metrics analyzed, performance models used, and applicable project stage. Our survey helps SOA architects to select the appropriate approach based on target performance metric and researchers to identify the SOA state-of-art performance prediction.

Keywords: Service; Service-Oriented Architecture; Performance; Prediction; Evaluation

1. INTRODUCTION

Service-Oriented Architecture (SOA) is an architectural style as well as technology of delivering services to either users or other services through a network. SOA architecture created in order to satisfy business goals that include easy and flexible integration with other systems. SOA has many advantages such as reducing development costs, creative services to customers, and agile deployment [1].

There are many definitions for SOA, but they are all point to the same core idea that SOA is simply a collection of application services. The service defined as “a function or some processing logic or business processing that well-defined, self-contained, and does not depend on the context or state of other services” [2]. It also states that “Generally SOA can be classified into two terms: Services and Connectors.”

Open Management Group (OMG) defines SOA as: “an architectural style that supports service orientation”. It goes further to define service orientation, “service orientation is a way of thinking in terms of services and services-based development and the outcomes of services”. Moreover, SOA is communication between services and applications which sometimes involves data transfer. But the communication between applications does not happen as a point-to-point interaction; instead it happens through a platform-independent, general purpose middle-ware that handles all communications by the use of web services [2].

The main goal of this paper is to report a detail survey in performance prediction of SOA. Section 2 lays the important concepts of SOA. Section 3 explains by a diagram an example of SOA base application and how the system exchanges the messages. Section 4 presents the performance metrics of SOA based applications. We considered three important metrics which are response time, throughput, and resource utilization. Section 5 summarizes the previous work in table 1 mentioning the date of published paper, the name of authors, objectives, performance metrics, performance model, and applicability stage. Section 5 concludes the paper.

2. SERVICE-ORIENTED ARCHITECTURE CONCEPTS

This part briefly tries to describe some important concepts related to SOA.

2.1 Enterprise Service Bus (ESB)

An ESB is a standard infrastructure that combines messaging, web services, data transformation, and intelligent routing in a highly distributed and different environment [7] [9].

2.2 Business Process Execution Language (BPEL)

BPEL is a language for designing SOA based systems. It contains a lot of facilities such as web services composition, publishing available services, organizing service execution, and handling exceptions.

2.3 ACME

ACME is a generic language for describing software architecture. It presents constructs for describing systems as graphs of components interacting through connectors [11].

3. EXAMPLE

Figure 1 presents an example of SOA architecture. The example will explain in steps the requests and responses flow between service provider, service consumer, and the directory. Step 1 Service provider publishes its service description on a directory, step 2 Consumer performs queries to the directory to locate a service and find out to communicate with the provider, step 3 Service description is written in a special language called Web Service Description Language (WSDL), step 4 Messages are sent and received from the directory in a special language called Simple Object Access Protocol (SOAP), step 5 Consumer formulate its message to the provider using tag based language called Extensible Markup Language (XML). The message is generated in XML but it is
Based on specifications defined in WSDL, step 6 the response generated by the provider is also in tag based XML format.

**Figure 1** Example of SOA Architecture

Based on ISO 9126 performance metrics are response time, throughput, and resource utilization [12]. Therefore, accurate measuring of SOA application plays an important role to business success. If the application has an efficient performance, this will lead to high productivity, well hardware utilization, and customer satisfaction. Otherwise, SOA based application capability will have limited benefits, resource wasting, low productivity, and unsatisfied customer.

Applying performance to SOA applications one of the challenging non-functional quality attribute, this because of the physical geographic distribution of services, communication overhead, use of standard message format, and varying service workload [3]. Performance evaluation and analysis differs in each situation of SOA based application. However, previous works on service performance are not accurate and practical enough to effectively understand and diagnose the reasons behind performance degradation.

**4. SERVICE-ORIENTED ARCHITECTURE PERFORMANCE METRICS**

4.1 Service Response Time

Service Response Time is the measure of the time between the end of a request to a service and the beginning of the time service provider response. There are many considerations to measure service response time [4] as Figure 2 stated. The main reasons that cause low performance of SOA based applications are:

- The potential problems of XML which is the standard message format increases the time needed to process a request.
- The time needed to discover the services through the directory either in design time or run time.
- Rules that govern services contain a business process by business process’s need.
- Adaptation of service composition by adding new service or adapting existing services.
- Think time is an elapsed time between the end of a response time generated by a service and the beginning of an end user’s request [4].

**4.2 Throughput**

Throughput defined as the number of requests SOA application can process at a given period of time. There are two metrics for throughput; throughput of a service and throughput of a business process [4] as Figure 3 stated.

The value range of these two metrics service throughput and business process throughput must be greater than zero. The higher the values indicate a better SOA application performance.

**4.3 Resource Utilization**

To analyze the performance of SOA based applications in terms of resource utilization, there are three basics information needed: firstly, workload information, which consists of concurrent users, and request arrival rates. Secondly, software specification, which consists of execution path, components to be executed, and the protocol of contention used by the software [5]. Finally, environmental information, this information consists of system specification such as configuration and device service rates, and scheduling policies.
Figure. 2 Sub-metrics of SOA Response Times

Figure. 3 Sub-metrics of SOA Throughput

Figure. 4 Sub-metrics of SOA Resource Utilization
5. SOA PREDICTION AND EVALUATION APPROACHES

Several approaches have been created to evaluate and predict SOA based application performance. In the following we provide summaries of SOA performance prediction approaches in the scope of the survey. We have divided the approaches on seven columns such as author name and year of publication, main objective, prediction approach, analyzed metrics, performance model, validation method, and applicable project stage.

<table>
<thead>
<tr>
<th>Author Name/Year</th>
<th>Main Objective</th>
<th>Approach used</th>
<th>Metrics Analyzed</th>
<th>Performance Model</th>
<th>Method’s Validation</th>
<th>Applicable Project Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kounev, Samuel, et al. [6], 2010</td>
<td>Designing systems with build-in self-aware performance and resource management capabilities</td>
<td>Use dynamic architecture-level performance model at run-time for online performance and resource management</td>
<td>Response time and Resource utilization</td>
<td>Queuing Petri net Model</td>
<td>Compared with PCM model result</td>
<td>Runtime</td>
</tr>
<tr>
<td>Liu, et al. [7], 2007</td>
<td>Develop a performance model for predicting runtime performance based on COTS ESB (Enterprise Service Bus)</td>
<td>Measure primitive performance overheads of service routing activities in the ESB</td>
<td>Throughput and response time</td>
<td>Queuing Network Model</td>
<td>Compared with the results of Microsoft Web Stress Tool</td>
<td>Runtime</td>
</tr>
<tr>
<td>Tribastone, et al. [8], 2010</td>
<td>Present a method for performance predication of SOA at early stage of development.</td>
<td>Modeling the system using UML and two profiles, UML4SOA, and MARTE</td>
<td>Response time, Processor utilization</td>
<td>Layered Queuing Network Model</td>
<td>Compared with Mobile Payment case study performance result</td>
<td>Design time</td>
</tr>
<tr>
<td>Teixeira, et al. [9], 2009</td>
<td>Propose approach to estimate performance of SOA</td>
<td>The model uses Petri Net formalism to represent the process and estimate its performance using simulation.</td>
<td>Resource consumption, Service levels degradation</td>
<td>Stochastic Petri Nets Model</td>
<td>Compared with (Rud et al) Analytical Method and values from real applications</td>
<td>Design time</td>
</tr>
<tr>
<td>Punitha, et al. [11], 2008</td>
<td>Developing an architectural performance model for SOA</td>
<td>Building and measuring the performance model using ACME language</td>
<td>Response time, throughput, load capacity, heavily loaded components.</td>
<td>Queuing Network Model</td>
<td>Prototype SOA Application has been implemented and measured</td>
<td>Design time</td>
</tr>
<tr>
<td>Brüseke, et al. [12], 2014</td>
<td>Developing PBlaman (Performance Blame Analysis)</td>
<td>Comparing the observed response time of each component in a failed test case to expected response time from the contract</td>
<td>Response time</td>
<td>Palladio Component Model (PCM)</td>
<td>Applied on two case studies</td>
<td>Design time</td>
</tr>
<tr>
<td>Reddy, et al. [13], 2011</td>
<td>Modeling Web Service using UML</td>
<td>Simulate the model using Simulation of Multi-tiered Queuing Applications (SMTQA)</td>
<td>Response time and Server utilization</td>
<td>SMTQA Model</td>
<td>Applied on case study</td>
<td>Design time</td>
</tr>
</tbody>
</table>
6. CONCLUSION

We have surveyed the state-of-art in the research of performance prediction methods for service-oriented architecture based applications. The survey categorized the approaches according to the performance metrics analyzed, performance model, method validation, and approach applicable stage.

The field of performance evaluation and prediction for service-oriented architecture based application has been developed and matured over the last decade. Many tools and ideas have been implemented as good software engineering practice and should lead the creation of new approaches.

Our survey helps both architects and researchers. Architects can obtain a complete view of the performance evaluation and prediction approaches proposed to transfer them to industry, on the other hand researchers can align themselves with the proposed approaches and add more features in the future to enhance and enrich the area.

7. REFERENCES


