SERVICE SIMULATION IN INDUSTRY 4.0: A COMPARISON OF SIMULATORS

Tsoutsa P. M.Sc.[1,2], Professor Fitsilis P. PhD.[1], Assistant Professor Ragos O. PhD.[2]
Faculty of Business and Economics, University of Applied Sciences of Thessaly, Greece[1]
Department of Mathematics, University of Patras, Greece[2]

tsoutsa@teilar.gr, pfitsilis@gmail.com, ragos@math.upatras.gr

Abstract: *Web services are application components that can be linked together for creating new functionality. As such, they support industry environments by binding together inhabitant services with distributed on demand services for implementing the business strategies. However, web services are operating in dynamic environments and an important issue during the composition process is change management either at the local or at the web, in general. Therefore, service changes together with other nondeterministic behaviours must be efficiently propagated to ensure the validity and consistency of the enterprise plans. Simulations are effective tools for enhancing factory competitiveness. Simulating the operation of a developed composite service for verifying the developed product could be a solution to this problem. The results of the simulation will allow to predict and react to such behaviours before applying the developed service in real production. As such, this paper describes and compares the existing service simulators that could be used in smart industry environments in terms of their functionality. Based on this review, the identification of gaps in performing simulation of service composition in industrial environments is presented, and challenges to be met on the field are outlined.*

Keywords: SIMULATION, WEB SERVICES, EMBEDED SYSTEMS, INDUSTRY 4.0, COMPOSITION OF SERVICES

1. Introduction

Automation of the industry includes automation in binding together local, cloud and web services to provide support for its inhabitants. Services technologies offer applicable solutions for service composition that can be exploited for automated planning and composition. In many cases the composition process produces a system of systems that might be quite complex. The better comprehension of the functionality of these complex systems along with the prediction of unwanted results during their execution is challenging.

However, there are practical issues that industry environments will face when automation of services is involved in testing, monitoring and verifying their performance in such systems.

The utilization of simulations tools that give the ability to evaluate the hypothesis prior to the application of the developed service could be a solution to the above problem. Simulation studies are obtaining popularity and they are used in many scientific fields. An environment that mimics as closely as possible the real setting, such as a service simulator can play an important role in evaluating the quality of a service, before its actual execution [1]. A domain that have the ability to use a service simulator can discover and evaluate which combinations of individual services cause what positive and negative results and avoid costly and error prone process.

Composite service simulation of the execution enable developers to understand performance issues that might occur and ensure that a service will meet the expected performance. Based on the simulation output, it is possible to verify the composition process and perform more experiments.

Choosing a simulator in order to perform functionality testing in services architectures is not a trivial task [2]. Simulators offer different facilities to assists users and applications to test the new developed services and obtain results for performing evaluations.

This paper contributes with a survey on existing simulators and makes a comparison based on specific properties with reference to industry 4.0 environments.

The survey offers an overview and comparison of six simulators for service systems, desrcribing for each: (i) the main objective, (ii) the latest release of the simulator (iii) the service modeling approach it supports, (iv) the use of evaluation method for services, (v) the communication platforms or languages are used, (vi) the semantic web technologies support, (vii) potential behavioral observation, and (viii) whether the simulator has been validated.

The rest of this paper is organized as follows: Section 2 describes the selected service simulators, in Section 3 we compare them according to specific characteristics defined with reference to industry 4.0 environments, while the last section presents the conclusion of the reearch on service simulation.

2. Background

An important aspect of simulation is the ability to experiment with alternative arrangements (Shannon, 1975; Sol, 1982). The idea is to develop a model of the existing situation and, based on a diagnosis, develop one or more alternative models and find improved arrangements using 'what-if' analyses. Simulation can thus be used to test and analyse various scenarios and decide which scenario is most promising. According to [3], the disadvantages of using a simulator are: (i) potentially long time of model preparation, (ii) every simulation model is unique – its solutions cannot be used for analyzing other decisional issues, (iii) it allows for preparing alternative decisional solutions in subsequent experiments, but these are not optimal solutions for all conditions, (iv) simulation models generate answers to the questions related to specific and changeable conditions.

A service's simulator main goal is to facilitate the planning and development of atomic and composite services. They present the results of the execution and predict the performance of the tested system under various parameters of the system and workloads[4].

A wide range of service simulators implementation environments were reviewed. This review is intended to examine only service simulators platforms are available today. Hence, in the following, we present six simulators that were selected as more compatible for simulating services in academia and industry environments and their development progress have not stopped. Initially we shortly describe each simulator while a pivot table with a comparison is presented at the end of the section.

2.1 Planning and Execution for Experiments in Service Oriented Systems (PEESOS)

PEESOS [4] is a tool that aims to perform capacity-planning tests for end-to-end SOA applications. The tool provides a collaborative workload generator based on a Client-Server model to establish a realistic load test environment for capacity planning test. It makes functional testing and can capture the results from the simulations to predict the performance of the target system, under different resource configurations. It is also capable to predict non-functional metric such as QoS for clients in different environments. It includes facilities that assist developers to test new service based applications.

The latest version of the simulator is PEESOS-Cloud [5], it enables cloud services to be evaluated as well as to improve the ability of the workload so that it conforms to their described characteristics. In this version, experiments in a cloud environment were conducted that present how PEESOS-Cloud validate its capabilities.

2.2 Devices Profile for Web Services Simulator (DPWSim)

DPWSim is a simulation toolkit enabling the simulation of DPWS devices and protocols [6]. It enables developers to prototype, develop, and test IoT applications and services without the presence of the physical devices. It mimics all the software and protocol features and provide a way to simulate and manage DPWS devices. It is especially useful when developers want to focus on designing the business logic of IoT applications rather than the physical performance of devices.

It simulates the environment where DPWS devices exist by creating virtual devices that can be discovered on the network and can communicate with other devices or clients via DPWS protocols. It also provides management of new and existing simulations created with which, it offers a high flexibility for users. It also provides a graphical interface designed in Java Swing that users can interact with the virtual devices.

2.3 Services-Aware Simulation Framework (SASF)

The SASF [7] is a suite of tools supporting the modeling of a service-oriented system and the execution of this model in a virtual environment. Its intended use is to predict the behavior and the performance of a software system, under different resource configurations and workloads. Some features of the SASF simulation framework are:

(i) it generates simulated services from WSDL documents and a performance profile, thus, it reduces the development effort required

(ii) it provides recording and visualizing behavioral metrics of interest, supporting special-purpose metrics and visualizations

(iii) it supports automatically generating executable service simulations, based on existing data about the services of interest, including their WSDL specifications and their implementations

(iv) has the ability to receive interactive input from users and external programs, thus enabling the simulation of systems evolving at runtime and the integration of real and simulated services

(v) has extensibility both in terms of the implementations of the simulated components and the data collected during simulation.

The intended use of the simulator is both in the pre-deployment and the post-deployment stage. Simulations for new implemented services are created to support the process of better understanding, and systematically analyzing them in the pre-deployment stage. On the other hand, after deployment, the simulation can be improved using the SASF tools by capturing more accurate performance profiles. With simulation, alternative configurations can be tested to improve specific aspects of services. In addition, all this reasoning about the service performance happens offline, without interfering with it.

2.4 PlanetSim

PlanetSim [8] is an extensible and scalable service simulator that supports simulation of peer-to-peer systems and services. The simulation is developed in three layers: (i) the network layer, (ii) the node and (iii) the application layer. The network layer manages the node inter-communication, the node layer deals with the implementation of peer-to-peer overlay protocols while the application layer is focused on services running at existing peer-to-peer overlay protocols. All entities within the simulator are defined through separate interfaces. The simulator dictates the overall life cycle of the framework by calling the appropriate methods and obtaining information to dispatch messages through the Network. It provides only basic statistics as part of the simulator in order for the tool to be efficient for large-scale scenarios as well.

The simulator is a Java framework while the main entities are separately defined and permit extension. It provides algorithm-based implementation of all the peer-to-peer protocol into the Node entity, answering correspondingly to any incoming message and behavior-based implementation, which encapsulates any individual part of the protocol into a Behavior entity, processing the incoming messages as expected by the protocol.

2.5 Service Composition and Execution Tool (SCET)

SCET [9] enable users to compose and simulate services. It is capable of monitoring a service that is being executed and captures bottlenecks and performance problems in the service components, suggesting reordering. It visually shows the number of service invocations at the host and evaluates the execution and performance of services under various hypothetical conditions while presenting statistical performance estimates. These results approximate the actual invocation and allows decisions to be made based on the behavior of the simulated services without actual execution.

The definition of the statically composed services is made using the Web Service Flow Language (WSFL). It integrates the JSIM simulator [10], therefore it converts the WSFL specification to a model that the JSIM simulator can interpret.

2.6 CloudSim

CloudSim [11] is a framework for modeling and simulation of cloud computing infrastructures and services. It has become a popular open source cloud simulator in the research and academia. It is an extensible simulation framework that enables modelling, simulation, and experimentation of Cloud computing infrastructures and application services. By using CloudSim, developers can focus on specific system design issues that they investigate, without getting concerned about the low level details related to infrastructures and services.

The framework has the following features: (i) support for modelling and simulation of large scale Cloud computing infrastructure, (ii) it is a self-contained platform for modelling data centers, service brokers, scheduling, and allocations policies, (iii) provides availability of virtualization engine, which aids in creation and management of multiple, independent, and co-hosted virtualized services and (iv) offers flexibility to switch between space-shared and time-shared allocation of processing cores to virtualized services.

It is customizable tool, thus allows extension and definition of policies in the components and this helps developers to handle the complexities arising from simulated environments. It is written in Java and it is using existing libraries such as GridSim and SimJava to handle low-level requirements of the system. An extended version of CloudSim which is CloudReports[12] have a graphical user interface and offer a GUI for CloudSim simulations.

3. Comparison of Service Simulators

The subject of research is to compare the selected simulators according to the specific properties are chosen as relevant to the intended environment of use, i.e. the industry 4.0 domain.

Simulator / Characteristic	PEESOS	DPW SIM	SASF	PLANET SIM	SCET	CLOUD SIM
Objective	Focus on capacity planning	Supports simulation for the development of IoT services	Virtual model of an application, focus on capacity planning	Supports simulation of peer-to-peer systems and services	Compose and simulate services	Simulation of Cloud infrastructures and application services
Latest release/ Open Source	2016 No	2014 No	2013 No	2009 No	2003 No	2016 Yes
Service Modeling	SOA entries	DPWS	WSDL services profiles	Platform defined syntax	WSFL language	Platform defined syntax
Evaluation	Validate workflows	No	Statistical metrics	Basic statistical metrics	Statistical metrics /Time Analysis	Yes
Communication	Platform defined	SOAP	SOAP	Platform defined	SOAP	Platform defined
Platforms/ languages	Web tools	Java	Java	Java	Java, Perl	Java
Semantic Web technologies support	No	No	Yes	No	Yes	No
Behavioral observation	No	No	Yes	Yes	Yes	No
Validated	In presented case study.	IoT application scenario.	In presented case study.	Used in scientific projects.	Real Web service scenario.	Used in cloud environments.

Table 1: *Comparative matrix of simulator properties.*

3.1 Criteria for comparing the tools

We first describe the properties defined to compare the deferent simulators:

Objective :The objective of each simulator is described and if its use is specific to services or other systems can also be simulated.

Latest release :The year of the latest release. As it is already mentioned only in use simulators are included.

Open Source :Whether a simulator is open source or not.

Service Modeling :The language/model in which services are defined.

Evaluation :The evaluation method that provides the simulator to evaluate participant services and how it does this.

Communication :The protocol/language is used for communicating simulated services.

Platforms :The platforms or languages the simulator is build on.

Semantic web technologies support:

Whether semantic web technologies are used during interaction or semantically meaningful information derive.

Behavioral observation:

Whether the composite behavior of the services being simulated is taken into account or only the atomic responses to a specific call.

Validated : Under which circumstances the simulator is validated.

3.2 Discussion

The basic characteristics of each simulator are presented in Table 1. Based on our review, we discuss our observations for each characteristic.

We notice that PEESOS and CLOUD SIM simulator have the most recent versions while only the Planet simulator is open source.

The performance of an atomic service has the potential to affect the performance of a composite service. Performance evaluation in a simulator of services can help implementers understand the behavior of the atomic services and the composed services under specific application and workload. We notice that the majority of simulators offer performance evaluation with some of them offer more sophisticated by using graphical interfaces.

Regarding the communication protocols are used to communicate services we see that PLANETsim and CLOUDsim use platform defined syntax language while all others follow standard protocols.

Furthermore, our intention is to study which simulator platforms can be used in the Web, hence we tried to figure out which of them support semantic web technologies. From the potential of a platform however could not conclude the simulators ability of modeling semantic web services. However, SASF and SCET have the ability to simulate services by semantically describe them.

The simulators that provide connections, which link the behavior of the individual components to the resulting system effects, are SASF, PLANETsim and SCET.

PEESOS and SASF simulators are using a case study to prove the validation of the system while DPWsim, PLANETsim, SCET and CLOUDsim are used in real word scenarios to be validated.

4. Conclusion

Industry is an increasingly complex environment. Systems and models that are used are continously becoming more complex and dificult to analyse. In this context, simulation technology could be a powerful and useful tool for developers.

In this article, we have presented six available simulators that are or could be used in simulating service oriented systems in industry. Ultimately, the goal of the study is achieved since the table in section 3 analyzes nine properties of the described service simulators and figures out that there exist simulators partially ready to be used in industry environments. This finding is really hopeful since service simulators have potentials in industry 4.0.

Our research agenta includes experimenting with the aformentioned simulators especially those that support semantic web technologies.

References

[1] Chandrasekaran, S., Miller, J. A., Silver, G. S., Arpinar, B., & Sheth, A. P. (2003). Performance analysis and simulation of composite web services. Electronic Markets, 13(2), 120-132.

[2] Nunes, L. H., Nakamura, L. H. V., Kuehne, B. T., De Oliveira, E. M., Libardi, R. M. D. O., Adami, L. J., ... & Reiff-Marganiec, S. (2014, June). Peesos: A web tool for planning and execution of experiments in service oriented systems. In Web Services (ICWS), 2014 IEEE International Conference on (pp. 606-613). IEEE.

[3] Kikolski, M. (2017). Study of production scenarios with the use of simulation models. Procedia Engineering, 182, 321-328.

[4] Nunes, L. H., Nakamura, L. H. V., Kuehne, B. T., De Oliveira, E. M., Libardi, R. M. D. O., Adami, L. J., ... & Reiff-Marganiec, S. (2014, June). Peesos: A web tool for planning and execution of experiments in service oriented systems. In Web Services (ICWS), 2014 IEEE International Conference on (pp. 606-613). IEEE.

[5] Ferreira, C. H., Nunes, L. H., Pereira, L. A., Nakamura, L. H., Estrella, J. C., & Reiff-Marganiec, S. (2016, June). PEESOS-Cloud: a workload-aware architecture for performance evaluation in service-oriented systems. In Services (SERVICES), 2016 IEEE World Congress on (pp. 118-125). IEEE.

[6] Han, S. N., Lee, G. M., Crespi, N., Van Luong, N., Heo, K., Brut, M., & Gatellier, P. (2014, March). DPWSim: A simulation toolkit for IoT applications using devices profile for web services. In Internet of Things (WF-IoT), 2014 IEEE World Forum on (pp. 544-547). IEEE.

[7] Smit, M., & Stroulia, E. (2013). Simulating service-oriented systems: A survey and the services-aware simulation framework. IEEE Transactions on Services Computing, 6(4), 443-456.

[8] Pujol-Ahullo, J., & Garcia-Lopez, P. (2009, September). Planetsim: An extensible simulation tool for peer-to-peer networks and services. In Peer-to-Peer Computing, 2009. P2P'09. IEEE Ninth International Conference on (pp. 85-86). IEEE.

[9] Chandrasekaran, S., Miller, J. A., Silver, G. S., Arpinar, B., & Sheth, A. P. (2003). Performance analysis and simulation of composite web services. Electronic Markets, 13(2), 120-132.

[10] Miller, J. A., Seila, A. F., & Xiang, X. (2000). The JSIM web-based simulation environment. Future Generation Computer Systems, 17(2), 119-133.

[11] Calheiros, R. N., Ranjan, R., De Rose, C. A., & Buyya, R. (2009). Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. arXiv preprint arXiv:0903.2525.

[12] Sá, T. T., Calheiros, R. N., & Gomes, D. G. (2014). CloudReports: An extensible simulation tool for energy-aware cloud computing environments. In cloud computing (pp. 127-142). Springer, Cham.