A Hybrid Approach for User Activity Experiments on Internet of Services Platforms

Josef Spillner
Technische Universität Dresden
Faculty of Computer Science, Chair for Computer Networks
01062 Dresden, Germany
E-mail: josef.spillner@tu-dresden.de

Abstract – The Internet of Services (IoS) idea promotes collaborative service exchange and trading platforms. Activity in the IoS is determined by metrics such as the number of users, number of services with associated descriptions and contract templates, negotiated contracts, performed invocations and generated measurements. In order to develop scalable and high-quality IoS platforms, experiments with user activity simulations are needed to achieve realistic metrics. We present a hybrid approach to keep the amount of simulation to a minimum based on evolving real IoS system replication.

I. Background and Related Work

The Internet of Services (IoS) vision as outlined by current research agendas describes a roadmap towards an interconnected world based on advanced service-oriented architecture concepts. Heterogeneous services on a wide spectrum between electronic web services and real-world services become tangible, distributable and tradeable entities. Services can be offered, composed and bundled into value-added offerings by service providers. Service consumers can use services ad-hoc or with contract protection for guaranteed functional and non-functional properties.

The backbone of the IoS is supposed to be built on distributed service platforms which support the brokering and execution of services. NEXOF-RA [1] and TEXO [2] are examples for IoS prototypes with varying states of development, concreteness and completeness. Such platforms handle service offering, matchmaking, usage and feedback tasks. New service description languages like the Unified Service Description Language USDL [3] and new service packaging and delivery approaches are required to convey the advantages of these platforms to users.

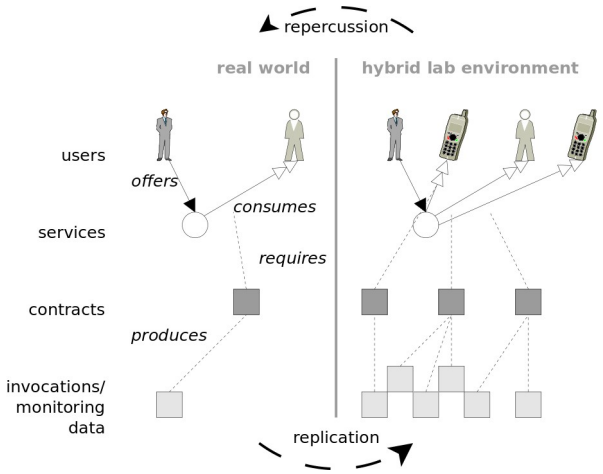
The suitability, scalability and general applicability of these innovations often needs to be evaluated in experimental facilities to come close to realistic conditions. We introduce a hybrid approach to perform simulations based on replicated IoS platforms which retains many aspects of real IoS systems while at the same time comprehends advantages of isolated laboratory experiments.

We are not aware of any concrete existing simulation approach for provider or consumer activity in the IoS. However, certain user activities like service auctions can already be simulated with specific toolkits [4]. We owe ideas and design considerations to hybrid approaches from other areas, such as physical structure testing [5] and network testbed federation [6].

II. Hybrid Approach for IoS Experiments

The diagram in Fig. 1 gives a conceptual overview on the proposed hybrid simulation environment. An existing single-server IoS platform is replicated into a controlled lab environment isolated from the origin platform instance. In the replicated instance, simulation techniques are used to generate user activities which trigger system activities and thus further populate the service usage databases with artificial data about contracts and monitoring results.

Figure 1: Hybrid simulation approach for IoS platforms



The cloning process can be either coarse-grained, by copying and reinstantiating a virtual machine image of the IoS platform, or fine-grained by individually copying files and dumping and restoring the contents of the databases, e.g., SQL and XML structures. We favour a fine-grained approach due to the possibility to anonymise the data and perform further selective modifications for subsequent processing and publication of the data. This allows researchers to create open-access datasets and others to use those in replay scenarios for verification purposes. Modern virtual machine and filesystem snapshot techniques make it feasible to retrieve internally consistent system replica at any point in time. The fine-grained variant even allows for repeated synchronisation of the replica with the origin system in long-running experiments.

The lab instance of the IoS platform is then used to retrieve, assess and generate key IoS metrics such as registered users, offered services, negotiated service level agreements (SLAs) and monitored invocations. We restrict our work to the two latter metrics due to the non-triviality of generating services, although we estimate that it would be possible to overcome this limitation partially by permutating at least numeric data such as non-functional properties in service descriptions. A further restriction results from the likely differences in how the origin and replica instances are connected to their respective clients.

For the generation of new metrics, the proposed approach includes agents which simulate human behaviour in how frequent which services and SLAs are needed. The agents' tasks encompass the selection of services, followed by negotiating SLAs and using the resulting contract for a number of contract-bound service invocations. The invocation will lead to a monitored service execution. Additional monitoring data will be generated independently from invocations, such as service endpoint availability metrics through regular HTTP monitoring probes.

A key benefit of the hybrid approach is that in the future, with an expected increase in availability and use of

IoS platforms, the simulation part can subsequently be gradually reduced. Hence, the results will by and by reflect real conditions. A second benefit results from a repercussion of simulation results on the original instance because the agents access the same interfaces like human users.

III. Applied Hybrid Simulation: Experiments on the SPACE Platform

The SPACE platform is a prototype for offering and using contract-bound heterogeneous web services. It can host SOAP and RESTful web services along with registered descriptions for real-world services [7]. SPACE has been designed to be modular and extensible. As such, it is a suitable base platform for experiments. The platform can be combined with user-friendly portals such as social networks [8] to gain a large and active user base. We operate an instance of such a social service network named Crowdserving with (at the time of initiating the experiment) 89 users, 4 communities, 7 virtual service-offering organisations and 87 offered services.

In order to perform a scalability experiment on SPACE with as much as real IoS usage data from Crowdserving, we planned and passed five experimentation phases.

- Phase 1: Building up the network, gaining users and services
- Phase 2: Preparing a concrete experiment with users, among them students from Vellore Institute of Technology in India (VIT), Kiev Polytechnic Institute in Ukraine (KPI) and TU Dresden in Germany
- Phase 3: Running two activity sprints by guiding users to offer, contract and consume services
- Phase 4: Replicating the platform and unleashing the IoS user activity simulator
- Phase 5: Postprocessing and evaluating the results

For phase 4, we developed a multi-threaded user activity simulator which can be further parallelised by being run on multiple hosts. Each thread obeys to a certain activity frequency which determines the rate with which new SLAs are established and services are invoked. The simulator uses the SPACE platform service interfaces to query existing SLA templates, register SLAs and run authenticated and monitored web service invocations through a SOAP proxy. Its architecture consists of a main thread which contains all simulator agent threads, as shown in Fig. 2.

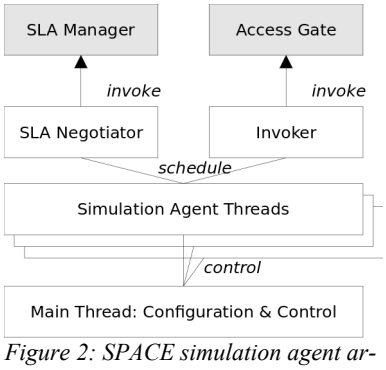


Figure 2: SPACE simulation agent architecture

Each activity is to a large extent encapsulated in an external module which can be implemented using different technologies. For example, while the simulator is written in Python, the SLA negotiation module is a Ruby script which uses the existing Ruby API to the SLA manager interface.

The following concrete results were achieved during the first sprint: We instructed 44 users from VIT to establish an SLA with a scenario service and invoke the service. During the sprint, 13 SLAs with varying conditions were fully negotiated, mostly based on an SLA template with fixed (non-negotiable) objectives. Only 5 service invocations were performed with SLAs, and in addition several anonymous invocations without SLAs. We concluded that the contract-bound service frontend delivery needs to become more intuitive. This impression was affirmed by the 6 students who sent us partially extensive feedback sheets.

The observed data from the public instance was used to parameterise the agents. Hence, the activity frequencies and distribution were projected onto the lab instance as logical continuation of the user study in a shorter timeframe. This gave us insight into predicting the system performance over a span of one month in just one hour. This time is a combination of a fixed replication time and a time dependent on activities during the observation period, excluding all non-active intervals.

IV. Discussion

We have presented an approach for hybrid simulation experiments based on replicated platforms in the Internet of Services. The approach has already been applied to an experiment on contract-bound service execution. We have moderately broached limitations of this approach especially concerning network-related results, but have omitted a thorough examination of further limitations.

The SPACE service platform and the simulation agent are made publicly available by us at the SPACE website [7]. We anticipate an adoption by researchers in order to improve service descriptions, packages, platform services and algorithms. In particular, we would like to see additional agent-simulated activities such as user registration and feedback on service quality. In the future, we also want to offer an open-access repository with replayable experiments.

V. Acknowledgment

The project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference ``01MQ07012''. The authors take the responsibility for the contents.

VI. References

[1] NEXOF-RA: http://www.nexof-ra.eu/
[2] TEXO: `http://theseus-programm.de/`
[3] Unified Service Description Language: `http://internet-of-services.com`
[4] *S. König, S. Hudert, T. Eymann*: Socio-Economic Mechanisms to Coordinate the Internet of Services: The Simulation Environment SimIS. Journal of Artificial Societies and Social Simulation 13 (2) 6, 2010.
[5] *M. V. Sivaselvan*: A Unified View of Hybrid Simulation Algorithms. NEES Hybrid Simulation Workshop, Berkeley, California, USA, April 2006.
[6] *M. Lacage, M. Ferrari, M. Hansen, T. Turletti*: NEPI: Using Independent Simulators, Emulators, and Testbeds for Easy Experimentation. 4th Workshop on Real Overlays and Distributed Systems (ROADS), Big Sky, Montana, USA, October 2009.
[7] SPACE Service Platform: `http://serviceplatform.org`
[8] *J. Spillner, A. Caceres, B. Buder, R. Kursawe, L. S. Globa, A. Schill*: Extending Social Networks with Service Delivery Capabilities for User-Centric Service Trading. Xth International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), Lviv-Slavske, Ukraine, February 2010.