Deviceless Edge Computing: Extending Serverless Computing to the Edge of the Network

Alex Glikson
IBM Research
glikson@il.ibm.com

Stefan Nastic
TU Wien
snastic@dsg.tuwien.ac.at

Schahram Dustdar
TU Wien
dustdar@dsg.tuwien.ac.at

ABSTRACT

The serverless paradigm has been rapidly adopted by developers of cloud-native applications, mainly because it relieves them from the burden of provisioning, scaling and operating the underlying infrastructure. In this paper, we propose a novel computing paradigm – *Deviceless Edge Computing* that extends the serverless paradigm to the edge of the network, enabling IoT and Edge devices to be seamlessly integrated as application execution infrastructure. We also discuss open challenges to realize Deviceless Edge Computing, based on our experience in prototyping a deviceless platform.

CCS Concepts

•**Applied computing** → **Event-driven architectures;**
•**Software and its engineering** → *Runtime environments;*

Keywords

Serverless computing; edge computing; function as a service

1. FROM SERVERLESS TO DEVICELESS

Serverless computing is an emerging paradigm, typically referring to a software architecture where application is decomposed into 'triggers' (events) and 'actions' (functions), and there is a platform that provides seamless hosting and execution environment, making it easy to develop, manage, scale and operate them. There are multiple commercial and open source implementations of serverless platforms such as Amazon Lambda[1] and Apache OpenWhisk[2]. Furthermore, the unique nature of serverless platforms introduces a vast potential for systems research [2].

While originally designed for cloud, the benefits of serverless paradigm are also vital in Edge/Fog computing environments, which today often suffer from high complexity, labor-intensive lifecycle management and ultimately high cost. Some of the existing serverless techniques, such as *sandboxed execution of tenant-provided code* and *programmable mapping between event sources and actions*, can be applied at the Edge too. However, due to inherently different nature of Edge infrastructure, many architecture and design assumptions behind serverless computing need to be reexamined in order to realize *Deviceless Edge Computing*.

[1]http://aws.amazon.com/lambda
[2]http://openwhisk.org/

SYSTOR 2017, May 22-24, 2017, Haifa, Israel

ACM ISBN 978-1-4503-5035-8/17/05.
http://dx.doi.org/10.1145/3078468.3078497

2. DEVICELESS EDGE COMPUTING

We conducted a series of experiments addressing the key requirements to realize the Deviceless Edge Computing. The resulting reference architecture and prototype can be found in our report[3]. Furthermore, based on our initial research, we identified a number of challenges, which include:

Resource pooling and elasticity. Serverless platforms are designed to utilize small footprint and short execution duration of functions, combined with statistical multiplexing of a large number of heterogeneous workloads over time [1]. Unique characteristics of edge deployments, such as geographic dispertion, heterogeneous and often unreliable connectivity, affinity of data sources (e.g., sensors) suggest the need for different methods to resource pooling and optimization (e.g., peer-to-peer rather than centralized).

Security. Unlike serverless platforms which often operate behind a firewall, the edge environment is exposed to direct attacks, requiring different protection and isolation mechanisms for the individual hosts, tenants and applications (e.g., decentralized, hardware-assisted, etc).

Provisioning and management at scale. Due to heterogeneity and geographical distribution of Edge environments, traditional centralized management and provisioning approaches need to be augmented with properties critical at the Edge, such as higher tolerance to network disruption, or location-aware management of software images. Thus, novel techniques, which will provide a uniform view and interaction with both Cloud and Edge are needed [3].

3. REFERENCES

[1] D. Breitgand, A. Glikson, et al. Sla-aware resource over-commit in an iaas cloud. In *CNSM'12*.
[2] S. Hendrickson et al. Serverless computation with openlambda. In *HotCloud'16*, pages 33–39, 2016.
[3] S. Nastic, H.-L. Truong, and S. Dustdar. A middleware infrastructure for utility-based provisioning of iot cloud systems. In *SEC*, pages 28–40. IEEE, 2016.

[3]https://medium.com/openwhisk/serverless-edge-to-cloud-computing-the-open-source-way-28ea33f60bf6