Service Oriented Computing and Applications (2018) 12:1–10
https://doi.org/10.1007/s11761-018-0228-2

SPECIAL ISSUE PAPER

Notes on ensembles of IoT, network functions and clouds for service-oriented computing and applications

Hong-Linh Truong[1] · Nanjangud C. Narendra[2] · Kwei-Jay Lin[3]

Received: 9 February 2018 / Accepted: 20 February 2018 / Published online: 3 March 2018

Abstract
Many advances have been introduced recently for service-oriented computing and applications (SOCA). The Internet of Things (IoT) has been pervasive in various application domains. Fog/Edge computing models have shown techniques that move computational and analytics capabilities from centralized data centers where most enterprise business services have been located to the edge where most customer's Things and their data and actions reside. Network functions between the edge and the cloud can be dynamically provisioned and managed through service APIs. Microservice architectures are increasingly used to simplify engineering, deployment and management of distributed services in not only cloud-based powerful machines but also in light-weighted devices. Therefore, a key question for the research in SOCA is how do we leverage existing techniques and develop new ones for coping with and supporting the changes of data and computation resources as well as customer interactions arising in the era of IoT and Fog/Edge computing. In this editorial paper, we attempt to address this question by focusing on the concept of ensembles for IoT, network functions and clouds.

Keywords IoT · Cloud computing · Network functions · Composition · Service computing

1 Introduction

One of the key aspects of service-oriented computing is the capability to compose various software components to offer on-demand services and applications. This key aspect is even more important in the current computing landscape where new IoT, network functions [34] and cloud resources are emerging on daily basis, enabling us to build new services and applications for solving various problems. With today's tools and frameworks, it is quite easy to expose capabilities of IoT, networks and cloud resources through well-defined service interfaces for on-demand needs with pay-per-use models, and the concept of resources is no longer limited to computing power with CPU and memory. We have seen different types of resources, such as data, analytics, firewall, intrusion detection, messaging, gateways, virtual machines, to name just a few, to be explicitly specified and composed into services and applications and to be controlled on-demand [28,34,67].

We have seen many advances in composition and provisioning services and IoT, network function virtualization (NFV) and cloud computing [21,30,38,47,50,62]. However, our interest in this paper is not about the discussion of silo works in individual IoT, network or cloud systems. We want to examine IoT, network functions and clouds as a whole, with specific application contexts. This leads to the key question of our work: how can we blend resources from IoT, cloud (micro) data centers and edge/core networks to build virtual continuum-based resource slices across various layers and systems? It is an important academic question [71], as well as crucial in industry[1], but we lack theoretical foundations. In our view, this needs strong support from capabilities of tools and frameworks for services engineering, provisioning, operation and analytics. This is, however, not easy to achieve due to the complexity and diversity of IoT [24], distributed clouds [16,54], and Fog/Edge computing [9,14]. While in

✉ Hong-Linh Truong
hong-linh.truong@tuwien.ac.at

Nanjangud C. Narendra
nanjangud.narendra@ericsson.com

Kwei-Jay Lin
klin@uci.edu

[1] Faculty of Informatics, TU Wien, Vienna, Austria

[2] Ericsson Research, Bangalore, India

[3] EECS, University of California, Irvine, Irvine, USA

[1] http://wings-ict-solutions.eu/cloud-continuum/, https://siliconangle.com/blog/2016/08/22/an-edge-to-cloud-continuum-inside-ge-digitals-predix-industrial-internet-of-things-platform-thecube/.

these fields the use of service computing is extensive—almost all resources are provided by web services in IoT, network functions, and clouds—our software components need to deal with a lot of data generated by Things [64]. We know that distributed analytics are carried out in different places [25,58], software interactions are not just based on REST [59], and the service response is almost near real time.

Realizing the impact of IoT and Fog/Edge computing, several calls have been made for services models and corresponding techniques for them [11,24,75]. While many works have started to produce and deploy services in the edge, there exist also many open research and challenging issues [18,29]. Conceptual, generic integration models between IoT and Cloud have been discussed and surveyed intensively [10,20]. Our goal in this editorial paper is not to discuss many aspects of the integration between IoT, network functions and clouds. Instead, we focus on building *ensembles* of IoT, network functions and clouds for SOCA. We consider this research direction important because—with the availability of various services for IoT, network functions, and clouds—we can compose, provision and operate such ensembles of resources across the edge and data centers for various application domains. However, not only we need to understand the potentials, we also have to analyze the challenges that one will have to face and what we need to do.

In this paper, we will focus on the key question of ensemble models, and the state-of-the art in terms of service composition, runtime aspects and analytics. We provide motivations for the concept of ensembles before focusing on state-of-the art. Since the work on ensembles will involve complex, cross-layer and cross-system activities, we will concentrate on certain important aspects related to SOCA, such as service models, composition, testing, and interoperability. We believe that, in this way, our editorial will provide a broader view onto existing problems and present guidelines for future research.

The rest of this paper is organized as follows: Sect. 2 discusses emerging ensembles for IoT, network functions and cloud services, and key issues in composition and management of ensembles. We analyze state-of-the-art in Sect. 3 before presenting key suggestions in Sect. 4. We conclude the paper in Sect. 5.

2 Emerging ensembles of IoT, network functions and clouds

2.1 Ensemble model

The first question we need to clarify is what does it mean by "ensembles" of IoT, network function and cloud resources? We (re)emphasize that our notion of resources can be used to indicate, e.g., data, network, analytics, and messaging. Thus resources can be mapped to virtual machines, containers, network firewall services, complex processing services, data streams of Things, to name just a few. Using this notion, in our work, ensembles consist of resources from IoT, network functions, and cloud services that establish a set of resources utilized for the application in a specific context (with clearly time, performance, cost, etc.). Resources in an embemble work together, clearly establishing the so-called resource slice [71]. At runtime, ensembles might be "intelligent" (or not) by being able to reconfigure and adapt themselves. In our work, we do not really emphasize the intelligence aspect of ensembles like in work in the artificial intelligence domain, but are concerned more on the question of resource management and analytics of ensembles.

Figure 1 presents the very high-level view of ensembles of IoT, network function and cloud resources. Such ensembles can be built by using service techniques because we have observed that not only current cloud resources provide excellent service models but also:

- NFV/5G is a key emerging area and more providers deploy services at the edge. In fact, many features of NFV are built based on elastic cloud models and cloud technologies
- IoT infrastructure-as-a-service and pay-per-use IoT communication are on the rise.
- IoT data-as-a-service and edge analytics as a service follow the existing cloud service models, provided by public and private IoT providers

Given these trends, eventually providers of IoT and network functions will present programmable APIs for us to utilize their resources, enabling the combination of such resources with well-utilized cloud resources for application-specific contexts.

The second question is whether we really need such ensembles, as the integration of IoT, network functions and clouds can be continuously done with other means. To answer this we need to distinguish between generic integration models, such as discussed in [10,20], with specific, concrete models for integration in our vision. Let us consider various application scenarios that demonstrate the need to combine various resources (data, network functions, cloud data services, etc.) for the application explicitly:

- Geosport/sport analytics: consider a sport analytics application we might analyze data of athletes in a short time, e.g., 2 hours of games. In such analytics, we need IoT systems to bring data to the cloud services but we also need network functions to guarantee the network traffics and to prevent data from being sent to wrong destinations. Concrete scenarios can be found in the H2020 U-Test

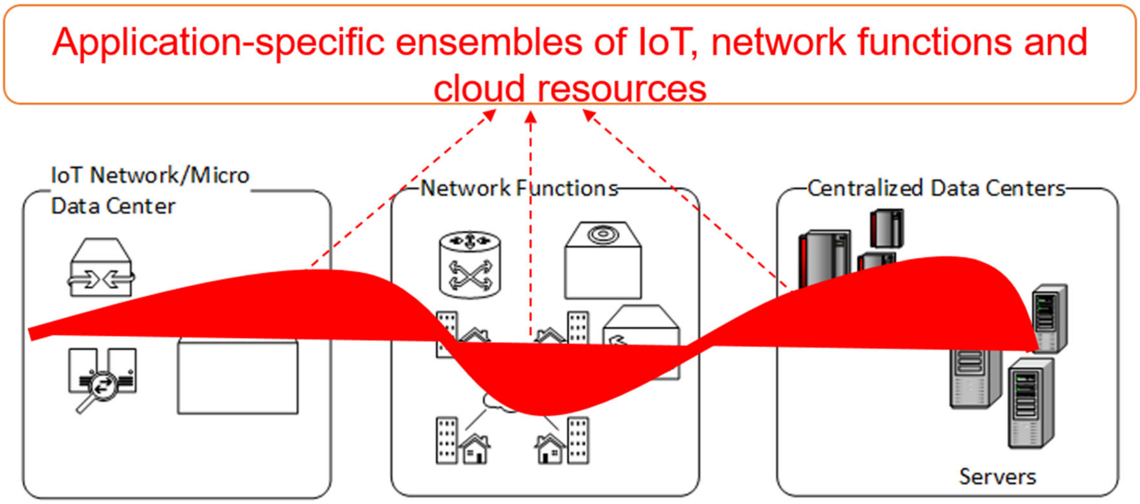
Application-specific ensembles of IoT, network functions and cloud resources
IoT Network/Micro Data Center
Network Functions
Centralized Data Centers
Servers

Fig. 1 High-level abstract view of ensembles

project[2] but as well as in various real use cases which could be extracted from those described in [12].

- Video analytics in smart cities: consider the case that we have many public and private cameras in the city that are exposed through Web services. For a particular event, e.g., football events, various clients want to obtain camera video as well as cloud and network, such as police and hospital for a few hours of data analytics. Huge amount of resources from IoT, network functions and clouds are needed in different contexts.
- Emergency responses in seaports: consider the case of an accident in a seaport, e.g., in the Inter-IoT project[3]. We need to gather IoT data from terminals, vessels, cranes, containers, etc., in order to make the right responses. Such IoT data will need to be analyzed by services, which might be in the cloud and in the edge system within the seaport. The resources and their configurations are very specific for this response which might last a few hours only.

Here, it is important to see that typically many applications have utilized resources from IoT, network functions, and clouds but not all of them need to explicitly acquire and manage all resources. However, in the aforementioned scenarios, applications need ensembles with specific constraints. All of these applications lead to the question of how to establish ensembles of resources that we need to manage, control and optimize at IoT, edge and cloud sides, not just data flows from IoT to clouds [70].

2.2 Key research problems

It has been argued that the service technique is the right approach for modeling and implementation of IoT-based systems [42]: for example, the sensing and actuation capabilities of devices are exposed through microservices [51]. Then, higher level services are composed from the basic microservices for various data processing and decision-making functions that are closer to the applications. This naturally fits well to the extensive use of cloud resources as services, leading to a natural way of integrating IoT and cloud for SOCA.

However, the network between the cloud and IoT has also been provided with service capabilities. This leads to the question is how to build services through the composition of IoT, network functions and cloud services. What would be the direction that we should follow? In our view, we should take a fresh look at how IoT, network functions and clouds can be exposed as services that eventually bring on-demand resources from different layers from different systems to be available as programmable objects for application-specific needs. This means that we need to develop techniques for composing such resources and provisioning them for applications, without worrying about the low-level particular details of specific layers or systems.

3 State-of-the-art

3.1 IoT, network functions and clouds as a service

It is needless to discuss about cloud as a service as it is very matured in today's computing landscape. In cloud computing, the developer and end-user can easily access

[2] http://www.u-test.eu/use-cases/#tab-1429727705-1-5.

[3] http://www.inter-iot-project.eu/.

cloud services and utilize them. However, in the core network and/or the network between the edge and the cloud, network functions[4] are not really exposed to the developer for composition and/or customization for applications. Typically, network functions are controlled and managed by providers as underlying infrastructures to enable more dynamic service provisioning for rapidly meeting service-level agreements. Certain network functions associated with cloud infrastructures can be programmed, such as virtual network connections and firewalls. However, they are not the network functions in the edge or between the edge and the cloud; they are in cloud data centers.

For ensembles, we do not expect that all Fog/Edge network systems will be opened for application developers. However, we expect that the Fog/Edge system providers will expose certain network and computing capabilities under the services when the providers claim their infrastructure as-a-service. In this view, similar to the current cloud computing offerings where the developer can easily program private network connection or firewall settings, network functions can be requested, provisioned and controlled by the applications. For instance, one can deploy a firewall (a completely software service) to the edge to prevent sending of sensor data to cloud instances outside Europe.

To this end, network functions as a service [57] technologies could enable us to compose and orchestrate network services as part of ensembles. In many situations, the use of containers to establish overlay networks, such as with Weave[5], could also be a solution for deploying network functions for ensembles. Other approaches on utilizing microservices and service function chaining to establish the link from IoT to the cloud like [45] are interesting. However, they are still quite low-level technologies that need to be exposed to high-level services for composing ensembles.

On the IoT side, IoT is also provided as a service [55]. Mostly, this means that the IoT network can be controlled and managed through the service model in the sense of dedicated IoT services. However, advanced providers can also offer IoT features similar to existing public clouds: IoT services can be rented and paid per use for different customers[6]. Another aspect to leverage IoT as a service is to offer APIs for exploiting IoT. The issue of API management for IoT has been raised [74], but from the perspective of IoT providers to make sure that they can hide the complexity of the access to their resources.

All of the above-mentioned models enable us to compose, acquire and operate IoT, network functions and cloud resources based on our special needs. Thus they enable the concept of ensembles of IoT, network functions and cloud that we advocate in this editorial note.

3.2 Modeling IoT, network functions and cloud elements

To be able to provision IoT and network functions together with cloud services, one needs to be able to capture their information and capabilities. We have seen several information models for network functions, such as [73], NetJSON (http://netjson.org/rfc.html) and YANG (http://www.netconfcentral.org/yang_docs). However, these information models have not been linked to IoT resources.

For modeling the relationship between things, sensors, and cloud services, many works have been introduced [7,27,37,53,76]. Such efforts produce information models that can be part of ensembles and thus can be leveraged for abstracting capabilities of IoT services and network services. However, they do not address the modeling of ensembles in our view.

One approach to modeling IoT, network functions and clouds is to use Model-driven Engineering (MDE). Although IoT is an emerging topic for MDE [46], we have seen several papers introducing different techniques for modeling IoT and Cyber-physical Systems (CPS) [15,66]. In cloud engineering, there already exist many MDE tools for cloud services [13,22]. However, these MDE tools are focused on silo systems (either IoT or Cloud). They are not developed for ensembles and they do not include network functions as modeling resources.

3.3 Programming and composition models for ensembles

In terms of services selection and composition, various works about service selections can be used but their focus is not on IoT services, especially for data-aware service selections and mashups [31]. Most of the work have no connection to the network services, reflecting network functions and back-end clouds. Searching and finding IoT services is an active research area but it focuses mainly on finding devices and services (semantic service selection) [33]. In [40] a declarative language is presented for creating ensembles of autonomic components. This work too does not focus on IoT and networks. In [75] orchestration challenges for IoT have been discussed. Key issues of optimization have not been outlined. Recently, there has been work discussing about the intelligent distribution between the edge and the cloud [32]. However, they do not introduce concepts to manage and programming resources for specific IoT services across the edge and the cloud. In terms of composition algorithms, network-aware service composition [30] presents an integrated QoS-aware composition method that integrates application services and

[4] https://www.sdxcentral.com/sdn-nfv-use-cases/network-virtualization/network-function-as-a-service/.

[5] https://www.weave.works/docs/net/latest/overview/.

[6] http://www.zdnet.com/article/microsoft-launches-iot-as-a-service-offering-for-enterprises/.

network services together. The work in [62] presents algorithms for IoT service composition that consider quality of service (QoS) and network latency. Such works however have never been considered in the context of IoT services, network services and clouds together. In [39] user-centric IoT application composition is shown. This, however, does not include networks.

3.4 Resource management and adaptation

In cloud computing, sophisticated features to make cloud services autonomous (or, self-*) are based on MAPE-K [36]. Examples of cloud service adaptation are described in [48,80]. As surveyed in [52] there are a huge number of IoT middleware. But most of them do not manage network services between IoT and clouds, thus they do not adapt network functions at runtime. There are a few works to coordinate the resources between IoT and Cloud [17] but not the networks in between. Works for virtualizing the network functions, such as OPNFV Arno[7], allow us to operate and maintain complex network systems. Although arising for different purposes, the combination of NFV and SDN presents powerful solutions for network control and reconfiguration of complex distributed systems. However, for ensembles, the adaptation of such complex networks must be carried out in sync with the IoT and cloud counterparts. To date, we do not have frameworks to do the adaptation across IoT, network functions and clouds.

In earlier work [6,43,44] we proposed the notion of an autonomic middleware for IoT-based systems. The key aspect of this middleware is the use of MAPE-K concepts to facilitate context-aware adaptation of IoT service compositions. This also necessitates the automatic generation of these compositions, especially in two cases: (1) possible large latency of computation required to generate the compositions, and (2) the composition generation depends upon specific formalisms to specify precondition/effects and the planning algorithms. We also present the concept of a multi-layered context model—application layer, environment layer, device layer—so as to facilitate contextual adaptation. We envision the application of this idea for automated generation and adaptation of ensembles.

3.5 Monitoring and policy management

Given the resources across IoT, network functions and clouds, a key question is how do we monitor and control the execution policies across subsystems. In terms of monitoring, there is no lack of monitoring tools to capture various information from different layers, e.g., industrial tools such as Prometheus[8], Telegraf[9], Logstash[10], Fluentd[11] and academic works [1,3]. Tools such as [63] would provide valuable information. However, the main problem is that these tools do not give an end-to-end view on the ensembles. They provide various metrics but leave the metrics correlation and cross system analysis to the developer.

Another aspect is from the provider viewpoint: how would the provider enforce resources provisioned for application-specific ensembles? Generally, execution policy enforcement should be carried out across IoT, network functions and clouds. The policy execution at the IoT side is different from that for clouds and networks, such as, based on software-defined machines profiles and policies [56] and blockchain-based permission control [61]. Another tricky problem is that if a solution provider develops an ensemble and deploys the ensemble as a service for his/her customers, how does this solution provider work with underlying IoT, network functions and cloud providers w.r.t. execution policy? This problem still remains unclear.

3.6 Testing

There are many testing techniques for cloud services. DevOps and testing tools, such as automatic testing with Jenkins[12], Gating[13], and Tsung[14], and academic works have demonstrated powerful applications to test cloud services. Essentially, such testing techniques can also be extended to network services and IoT, as long as they offer well-defined interfaces. However, ensembles are established at runtime, hence it is not sure how such tools can be used.

Testing IoT is not a very well-researched topic. The work in [60] analyzes various challenges in testing IoT. Some authors also try to apply traditional testing methods for IoT, like fault injection [5]. All of these works are valuable but they are far from the requirement for testing ensembles. The key point is that, from the ensembles perspective, we see IoT and network functions as services. Thus, in order to test them we must have a mechanism, similar to cloud testing, but for IoT and network services, to dynamically test performance and failure. Furthermore, the test has to be within the context of ensembles, imposed by specific constraints with respect to performance and costs, for example.

Another important aspect of testing for ensembles is to test their uncertainty. For complex IoT, network functions and

[7] OPNFV Arno—https://www.opnfv.org.

[8] https://prometheus.io/.

[9] https://github.com/influxdata/telegraf.

[10] https://www.elastic.co/products/logstash.

[11] https://www.fluentd.org/.

[12] https://jenkins.io/.

[13] https://gatling.io.

[14] http://tsung.erlang-projects.org/.

clouds, it is easy to see the lack of knowledge about resources, as well as the lack of knowledge about the change of underlying systems. This lack of knowledge brings uncertainty that requires novel techniques to monitor and test [2,4,49,77,78]. While work like in H2020 U-test have presented taxonomy and uncertainty profiles and tools [72], the impact of uncertainty is actually not well studied for IoT, network functions and clouds.

Overall, the work of testing in cloud is quite well developed. Similarly, testing performance of network layers has been done intensively. The current work on testing IoT is very primitive and we do not see any testing tools for ensembles. One possibility is to extend certain types of IoT Cloud testing, for example, to run a lot of IoT data to test its impact on cloud services. But this testing is only for the two ends and it is not clear how we could do it for the network functions.

3.7 Interoperability

In the cloud, interoperability has been addressed extensively [79]. Interoperability ranges from standardization [65] to tools for specific solutions. In IoT, a published special issue [23] has introduced various interoperability problems and tools. Furthermore, there are many other solutions for interoperability, such as middleware-based [8], MDE-based [26], and protocol translation-based [19].

If we consider IoT, cloud or network functions individually, we see that each of them has to deal with various interoperability problems, such as mismatched interfaces, incompatible protocols, semantic and syntax data problems. The key interoperability problem for ensembles is that the ensembles are established across various layers and systems. Therefore, we need to deal with a lot of individual interoperability solutions and combine them together for ensembles. Here, the composition and provisioning techniques of SOCA could be important for selecting the right interoperable solutions based on meta-data and runtime data.

4 Recommendations

From our high-level analysis of the state-of-the art, we make the following five recommendations:

Rec #1—resource modeling and interoperability For us it is quite clear about the service models for IoT, network functions and clouds, especially microservices and REST-based implementations. We have seen that mostly service interfaces are provided through REST. However, especially at the IoT side, various IoT services offer different ways to access data and controls, e.g., using messaging protocols like MQTT and AMQP. The nature of interaction models for data acquisition and control is not in one-to-one interaction like typical enterprise services but is in one-to-many or many-to-one. Therefore, we recommend the service interaction models to be developed with one-to-many or many-to-one protocols to suit IoT, while still maintaining the key principle of services, such as composability and discovery.

We recommend IoT resources be modeled with many new types of meta-data about data quality, execution policies, etc., to enable the composition of interoperable services. On the one hand, we need to continue the current approach on implementing interoperability solutions by introducing middleware, protocols translation, etc. On the other hand, the dynamic approach for software composition and provisioning would also be another way.

Rec #2—programming We recommend to follow the current two approaches: the top-down approach with model-driven engineering (MDE) and the bottom-up approach with programming frameworks and languages. In MDE, we have observed various tools and MDE profiles for IoT and Cloud. However, frameworks for supporting ensembles of IoT, network functions and clouds are missing. There are also work in business processes [41]. One of our previous works [69] for example allows us to model IoT Cloud systems. However, it has not included network functions yet and it is still focused on modeling without code generation. Here our suggestion is to combine various MDE approaches to develop a unified framework for ensembles. The approach from programming frameworks and languages introduce various techniques. In particular, the complex service compositions mainly are described by workflows or data processing pipelines in which clouds are used for complex processing and IoT are reflected through data inputs. However, we have not seen the incorporation of network functions into such languages. Thus we suggest to incorporate network function APIs into existing languages to allow us to build ensembles.

Rec #3—service management and adaptation Adaptation needs to be context-aware [6] and driven by stored knowledge about the system and earlier adaptation implementations. However, current adaptation techniques and implementations are only for specific systems. We recommend to exploit end-to-end context-aware adaptation [43,44] based on issues triggered from the IoT side as the ensembles are mainly for addressing problems from the IoT side with also resources in the edge and cloud. Considering a huge number of uncertainty across layers and systems as well as the vast knowledge that one needs to master to work with ensembles, we recommend to develop uncertainty measure for ensembles. This can be started from existing work on uncertainty for IoT/CPS [72]. Furthermore, uncertainty must be included into ensemble composition and adaptation. Works like uncertainty adaptation in IoT Cloud [49] could be a starting point.

Rec #4—end-to-end monitoring Given a large number of tools for infrastructure monitoring and application monitoring as well as their related services for managing monitoring data, we recommend a focus on correlation metrics and end-to-end monitoring, as they are missing at the moment. One approach is to establish end-to-end monitoring by leveraging existing tools, such Prometheus and Fluentd, by combining with end-to-end techniques, such as from [35]. However, new metrics characterizing ensembles must be defined, for example, how to calculate the availability of an ensemble based on the availability of IoT, network functions and cloud.

Rec #5—testing We suggest to focus on end-to-end testing through integration and system testing. We should see ensembles as a uniform unit to test flows from IoT to clouds via network functions. In this way, we can combine different testing techniques for IoT, network functions and cloud but we need to (i) define combination/composition of tests, (ii) testing utilities for different systems, (iii) test emerging properties like uncertainty, elasticity, actuation. One approach we suggest is to combine IoT testing with elastic testing of cloud services [68].

5 Conclusions

In this paper, we motivated the building of application-specific ensembles of IoT, network functions and clouds, given the current offering and technology trends in IoT and Fog/Edge computing. We presented the concept of ensembles of IoT, network functions and cloud resources. Based on our preliminary works and discussions, we believe that such ensembles are important because many applications need such ensembles for their specific context with respect to performance, time and costs that they can manage and control. We have discussed the current state-of-the art, various important aspects as well as presented recommendations for the composition and analytics of ensembles.

Our current work is to deal with theoretical concepts of ensembles, such as the formal way to describe and represent ensembles. We are working intensively into the tools to support this concept, especially, monitoring, uncertainty analytics and programming and execution management for ensembles. Certain samples of services and ensemble structures are being updated at https://github.com/rdsea/IoTCloudSamples.

Acknowledgements Open access funding provided by TU Wien (TUW). This paper is based on various discussions and our previous works. In particular, our paper [71] discusses the resource slices of IoT, network functions and cloud. Subsequently, we have developed further techniques for resource integration and execution policy. Various talks and discussions have helped us strengthen our views on ensembles. Application scenarios we have discussed based on our intensive collaboration with industries and other academic partners in many projects like H2020 U-Test and Inter-IoT projects. Hong-Linh Truong also would like to thank Profs Yusheng Ji and Shigeki Yamada from NII, Japan, for the discussion of ensembles of IoT and network functions. The work in this paper is partially supported by the H2020 U-Test project and H2020 Inter-IoT project via the subproject INTER-HINC.

References

1. Aceto G, Botta A, De Donato W, Pescapè A (2013) Survey cloud monitoring: a survey. Comput Netw 57(9):2093–2115. https://doi.org/10.1016/j.comnet.2013.04.001
2. Akkaya I, Liu Y, Lee EA (2016) Uncertainty analysis of middleware services for streaming smart grid applications. IEEE Trans Serv Comput 9(2):174–185. https://doi.org/10.1109/TSC.2015.2456888
3. Alhamazani K, Ranjan R, Mitra K, Rabhi F, Jayaraman PP, Khan SU, Guabtni A, Bhatnagar V (2015) An overview of the commercial cloud monitoring tools: research dimensions, design issues, and state-of-the-art. Computing 97(4):357–377. https://doi.org/10.1007/s00607-014-0398-5
4. Ali S, Yue T (2015) U-test: evolving, modelling and testing realistic uncertain behaviours of cyber-physical systems. In: 8th IEEE international conference on software testing, verification and validation, ICST 2015, Graz, Austria, April 13–17, 2015, pp 1–2. IEEE. https://doi.org/10.1109/ICST.2015.7102637
5. Alipour MA (2017) Fault injection in the internet of things applications. In: Proceedings of the 1st ACM SIGSOFT international workshop on testing embedded and cyber-physical systems, TECPS 2017, pp 9–11. ACM, New York, NY, USA. https://doi.org/10.1145/3107091.3107095
6. Bellur U, Narendra NC, Mohalik SK (2017) AUSOM: autonomic service-oriented middleware for IoT-based systems. In: 2017 IEEE world congress on services (SERVICES), pp 102–105. https://doi.org/10.1109/SERVICES.2017.25
7. Benazzouz Y, Munilla C, Gunalp O, Gallissot M, Gurgen L (2014) Sharing user IoT devices in the cloud. In: 2014 IEEE world forum on internet of things (WF-IoT), pp 373–374. https://doi.org/10.1109/WF-IoT.2014.6803193
8. Blackstock M, Lea R (2014) IoT interoperability: a hub-based approach. In: 2014 international conference on the internet of things (IOT), pp 79–84. https://doi.org/10.1109/IOT.2014.7030119
9. Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. In: Proceedings of the first edition of the MCC workshop on mobile cloud computing, pp 13–16. ACM
10. Botta A, de Donato W, Persico V, Pescapé A (2016) Integration of cloud computing and internet of things. Future Gener Comput Syst 56(C):684–700. https://doi.org/10.1016/j.future.2015.09.021
11. Bouguettaya A, Singh M, Huhns M, Sheng QZ, Dong H, Yu Q, Neiat AG, Mistry S, Benatallah B, Medjahed B, Ouzzani M, Casati F, Liu X, Wang H, Georgakopoulos D, Chen L, Nepal S, Malik Z, Erradi A, Wang Y, Blake B, Dustdar S, Leymann F, Papazoglou M (2017) A service computing manifesto: the next 10 years. Commun ACM 60(4):64–72. https://doi.org/10.1145/2983528
12. Brefeld U, Zimmermann A (2017) Guest editorial: special issue on sports analytics. Data Min Knowl Discov 31(6):1577–1579. https://doi.org/10.1007/s10618-017-0530-1
13. Casale G, Ardagna D, Artac M, Barbier F, Nitto ED, Henry A, Iuhasz G, Joubert C, Merseguer J, Munteanu VI, Pérez JF, Petcu D, Rossi M, Sheridan C, Spais I, Vladušič D (2015) Dice: quality-driven development of data-intensive cloud applications. In: Proceedings of the seventh international workshop on modeling in software engineering, MiSE'15, pp 78–83. IEEE Press,

Piscataway, NJ, USA. http://dl.acm.org/citation.cfm?id=2820489.2820507

14. Chiang M, Zhang T (2016) Fog and IoT: an overview of research opportunities. IEEE Internet Things J 3(6):854–864. https://doi.org/10.1109/JIOT.2016.2584538
15. Ciccozzi F, Spalazzese R (2017) MDE4IoT: supporting the internet of things with model-driven engineering. Springer, Cham, pp 67–76
16. Coady Y, Hohlfeld O, Kempf J, McGeer R, Schmid S (2015) Distributed cloud computing: applications, status quo, and challenges. SIGCOMM Comput Commun Rev 45(2):38–43. https://doi.org/10.1145/2766330.2766337
17. Copil G, Moldovan D, Truong HL, Dustdar S (2014) On controlling cloud services elasticity in heterogeneous clouds. In: Proceedings of the 2014 IEEE/ACM 7th international conference on utility and cloud computing, UCC '14, pp 573–578. IEEE Computer Society, Washington, DC, USA. https://doi.org/10.1109/UCC.2014.88
18. Creeger M (2010) Moving to the edge: CTO roundtable overview. Queue 8(7):40:40–40:42. https://doi.org/10.1145/1831327.1838135
19. Derhamy H, Eliasson J, Delsing J (2017) Iot interoperability: on-demand and low latency transparent multiprotocol translator. IEEE Internet Things J 4(5):1754–1763. https://doi.org/10.1109/JIOT.2017.2697718
20. Díaz M, Martín C, Rubio B (2016) State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. J Netw Comput Appl 67(C):99–117. https://doi.org/10.1016/j.jnca.2016.01.010
21. Echeverría S, Root J, Bradshaw B, Lewis GA (2014) On-demand VM provisioning for cloudlet-based cyber-foraging in resource-constrained environments. In: Julien C, Lane ND, Mishra S (Eds) 6th international conference on mobile computing, applications and services, MobiCASE 2014, Austin, TX, USA, November 6–7, 2014, pp 116–124. IEEE. https://doi.org/10.4108/icst.mobicase.2014.257768
22. Ferry N, Song H, Rossini A, Chauvel F, Solberg A (2014) CloudMF: Applying MDE to tame the complexity of managing multi-cloud applications. In: Proceedings of the 2014 IEEE/ACM 7th international conference on utility and cloud computing, UCC'14, pp 269–277. IEEE Computer Society, Washington, DC, USA. https://doi.org/10.1109/UCC.2014.36
23. Fortino G, Ganzha M, Palau C, Paprzycki M (2016) Interoperability in the internet of things. https://www.computer.org/web/computingnow/archive/interoperability-in-the-internet-of-things-december-2016-introduction
24. Georgakopoulos D, Jayaraman PP (2016) Internet of things: from internet scale sensing to smart services. Computing 98(10):1041–1058. https://doi.org/10.1007/s00607-016-0510-0
25. Govindarajan N, Simmhan Y, Jamadagni N, Misra P (2014) Event processing across edge and the cloud for internet of things applications. In: Proceedings of the 20th international conference on management of data, COMAD'14, pp 101–104. Computer Society of India, Mumbai, India. http://dl.acm.org/citation.cfm?id=2726970.2726985
26. Grace P, Barbosa J, Pickering B, Surridge M (2014) Taming the interoperability challenges of complex IoT systems. In: Proceedings of the 1st ACM workshop on middleware for context-aware applications in the IoT, M4IOT'14, pp 1–6. ACM, New York, NY, USA. https://doi.org/10.1145/2676743.2676744
27. Haller S, Serbanati A, Bauer M, Carrez F (2013) A domain model for the internet of things. In: Green computing and communications (GreenCom), 2013 IEEE and internet of things (iThings/CPSCom), IEEE international conference on and IEEE cyber, physical and social computing, pp 411–417. https://doi.org/10.1109/GreenCom-iThings-CPSCom.2013.87
28. Hirmer P, Mitschang B (2017) TOSCA4Mashups: enhanced method for on-demand data mashup provisioning. Comput Sci Res Dev 32(3):291–300. https://doi.org/10.1007/s00450-016-0330-7
29. Hu W, Gao Y, Ha K, Wang J, Amos B, Chen Z, Pillai P, Satyanarayanan M (2016) Quantifying the impact of edge computing on mobile applications. In: Proceedings of the 7th ACM SIGOPS Asia-Pacific workshop on systems, APSys'16, pp 5:1–5:8. ACM, New York, NY, USA. https://doi.org/10.1145/2967360.2967369
30. Huang J, Liu G, Duan Q, Yan Y (2014) Qos-aware service composition for converged network-cloud service provisioning. In: 2014 IEEE international conference on services computing (SCC), pp 67–74. IEEE
31. Im J, Kim S, Kim D (2013) IoT mashup as a service: cloud-based mashup service for the internet of things. In: Proceedings of the 2013 IEEE international conference on services computing, SCC'13, pp 462–469. IEEE Computer Society, Washington, DC, USA. https://doi.org/10.1109/SCC.2013.68
32. Jalali F, Smith OJ, Lynar T, Suits F (2017) Cognitive IoT gateways: automatic task sharing and switching between cloud and edge/fog computing. In: Proceedings of the SIGCOMM posters and demos, SIGCOMM posters and demos, pp 121–123. ACM, New York, NY, USA. https://doi.org/10.1145/3123878.3132008
33. Jenson S, Want R, Schilit BN, Kravets RH (2015) Building an on-ramp for the internet of things. In: Proceedings of the 2015 workshop on IoT challenges in mobile and industrial systems, IoT-Sys'15, pp 3–6. ACM, New York, NY, USA. https://doi.org/10.1145/2753476.2753483
34. Kablan M, Caldwell B, Han R, Jamjoom H, Keller E (2015) Stateless network functions. In: Proceedings of the 2015 ACM SIGCOMM workshop on hot topics in middleboxes and network function virtualization, HotMiddlebox'15, pp 49–54. ACM, New York, NY, USA. https://doi.org/10.1145/2785989.2785993
35. Kaldor J, Mace J, Bejda M, Gao E, Kuropatwa W, O'Neill J, Ong KW, Schaller B, Shan P, Viscomi B, Venkataraman V, Veeraraghavan K, Song YJ (2017) Canopy: an end-to-end performance tracing and analysis system. In: Proceedings of the 26th symposium on operating systems principles, SOSP'17, pp 34–50. ACM, New York, NY, USA. https://doi.org/10.1145/3132747.3132749
36. Kephart JO, Chess DM (2003) The vision of autonomic computing. Computer 36(1):41–50
37. Kim J, Lee JW (2014) OpenIoT: an open service framework for the internet of things. In: 2014 IEEE world forum on internet of things (WF-IoT), pp 89–93
38. Klein A, Ishikawa F, Honiden S (2012) Towards network-aware service composition in the cloud. In: Proceedings of the 21st international conference on world wide web, WWW'12, pp 959–968. ACM, New York, NY, USA. https://doi.org/10.1145/2187836.2187965
39. Ko IY, Ko HG, Molina AJ, Kwon JH (2016) SoIoT: toward a user-centric IoT-based service framework. ACM Trans Internet Technol 16(2):8:1–8:21. https://doi.org/10.1145/2835492
40. Krijt F, Jiracek Z, Bures T, Hnetynka P, Gerostathopoulos I (2017) Intelligent ensembles: a declarative group description language and java framework. In: Proceedings of the 12th international symposium on software engineering for adaptive and self-managing systems, SEAMS'17, pp 116–122. IEEE Press, Piscataway, NJ, USA. https://doi.org/10.1109/SEAMS.2017.17
41. Meyer S, Ruppen A, Magerkurth C (2013) Internet of things-aware process modeling: integrating iot devices as business process resources. In: Salinesi C, Norrie MC, Pastor Ó (eds) Advanced information systems engineering. Springer, Berlin, pp 84–98
42. Mohalik SK, Jayaraman MB, Ramamurthy B, Vulgarakis A (2015) SOA-PE : a service-oriented architecture for planning and execution in cyber-physical systems. In: Proceedings international conference on smart sensors and systems (IC-SSS-2015). IEEE

43. Mohalik SK, Narendra NC, Badrinath R, Jayaraman MB, Padala C (2016) Dynamic semantic interoperability of control in IoT-based systems: need for adaptive middleware. In: 2016 IEEE 3rd world forum on internet of things (WF-IoT), pp 199–203. https://doi.org/10.1109/WF-IoT.2016.7845399
44. Mohalik SK, Narendra NC, Badrinath R, Le DH (2017) Adaptive service-oriented architectures for cyber physical systems. In: 2017 IEEE symposium on service-oriented system engineering (SOSE), pp 57–62. https://doi.org/10.1109/SOSE.2017.10
45. Morabito R, Beijar N (2017) A framework based on SDN and containers for dynamic service chains on IoT gateways. In: Proceedings of the workshop on hot topics in container networking and networked systems, HotConNet'17, pp 42–47. ACM, New York, NY, USA. https://doi.org/10.1145/3094405.3094413
46. Morin B, Harrand N, Fleurey F (2017) Model-based software engineering to tame the IoT jungle. IEEE Softw 34(1):30–36. https://doi.org/10.1109/MS.2017.11
47. Muoz F, Muoz R, Rodrguez J, Lopez V, de Dios OG, Fernndez-Palacios JP (2013) End-to-end service provisioning across MPLS and IP/WDM domains. In: 2013 international conference on smart communications in network technologies (SaCoNeT), vol 02, pp 1–5. https://doi.org/10.1109/SaCoNeT.2013.6654561
48. Nallur V, Bahsoon R (2013) A decentralized self-adaptation mechanism for service-based applications in the cloud. IEEE Trans Softw Eng 39(5):591–612. https://doi.org/10.1109/TSE.2012.53
49. Nastic S, Copil G, Truong HL, Dustdar S (2015) Governing elastic IoT cloud systems under uncertainty. In: 7th IEEE international conference on cloud computing technology and science, CloudCom 2015, Vancouver, BC, Canada, November 30–December 3, 2015, pp 131–138. IEEE. https://doi.org/10.1109/CloudCom.2015.77
50. Nastic S, Sehic S, Le DH, Truong HL, Dustdar S (2014) Provisioning software-defined IoT cloud systems. In: Proceedings of the 2014 international conference on future internet of things and cloud, FICLOUD'14, pp 288–295. IEEE Computer Society, Washington, DC, USA. https://doi.org/10.1109/FiCloud.2014.52
51. Newman S (2015) Building microservices, 1st edn. O'Reilly Media, Inc., Sebastopol
52. Ngu AH, Gutierrez M, Metsis V, Nepal S, Sheng QZ (2017) IoT middleware: a survey on issues and enabling technologies. IEEE Internet Things J 4(1):1–20. https://doi.org/10.1109/JIOT.2016.2615180
53. Oteafy S, Hassanein H (2012) Towards a global IoT: Resource re-utilization in WSNS. In: 2012 international conference on computing, networking and communications (ICNC), pp 617–622. https://doi.org/10.1109/ICCNC.2012.6167496
54. Pahl C, Lee B (2015) Containers and clusters for edge cloud architectures—a technology review. In: 2015 3rd international conference on future internet of things and cloud, pp 379–386. https://doi.org/10.1109/FiCloud.2015.35
55. Pencheva E, Atanasov I (2016) Engineering of web services for internet of things applications. Inf Syst Front 18(2):277–292. https://doi.org/10.1007/s10796-014-9532-3
56. Phung PH, Truong HL, Yasoju DT (2017) P4SINC—an execution policy framework for IoT services in the edge. In: 2017 IEEE international congress on internet of things (ICIOT), pp 137–142. https://doi.org/10.1109/IEEE.ICIOT.2017.23
57. Qadir J, Ahmed N, Yousaf FZ, Taqweem A (2016) Network as a service: the new vista of opportunities. arXiv preprint arXiv:1606.03060
58. Ravindra P, Khochare A, Reddy S, Sharma S, Varshney P, Simmhan Y (2017) ECHO : an adaptive orchestration platform for hybrid dataflows across cloud and edge. In: Maximilien EM, Vallecillo A, Wang J, Oriol M (Eds) Service-oriented computing—15th international conference, ICSOC 2017, Malaga, Spain, November 13–16, 2017, Proceedings, Lecture notes in computer science, vol 10601, pp 395–410. Springer. https://doi.org/10.1007/978-3-319-69035-3_28
59. Reinfurt L, Breitenbücher U, Falkenthal M, Leymann F, Riegg A (2016) Internet of things patterns. In: Proceedings of the 21st European conference on pattern languages of programs, EuroPlop'16, pp 5:1–5:21. ACM, New York, NY, USA. https://doi.org/10.1145/3011784.3011789
60. Rosenkranz P, Wählisch M, Baccelli E, Ortmann L (2015) A distributed test system architecture for open-source IoT software. In: Proceedings of the 2015 workshop on IoT challenges in mobile and industrial systems, IoT-Sys'15, pp 43–48. ACM, New York, NY, USA. https://doi.org/10.1145/2753476.2753481
61. Samaniego M, Deters R (2016) Using blockchain to push software-defined IoT components onto edge hosts. In: Proceedings of the international conference on big data and advanced wireless technologies, BDAW'16, pp 58:1–58:9. ACM, New York, NY, USA. https://doi.org/10.1145/3010089.3016027
62. Shehu UG, Safdar GA, Epiphaniou G (2015) Network aware composition for internet of things services. Trans Netw Commun 3(1):45
63. Sterle J, Sedlar U, Rugelj M, Kos A, Volk M (2016) Application-driven OAM framework for heterogeneous IoT environments. Int J Distrib Sens Netw 2016:3:3–3:3. https://doi.org/10.1155/2016/5649291
64. Taherkordi A, Eliassen F, Horn G (2017) From IoT big data to IoT big services. In: Proceedings of the symposium on applied computing, SAC'17, pp 485–491. ACM, New York, NY, USA. https://doi.org/10.1145/3019612.3019700
65. Teckelmann R, Reich C, Sulistio A (2011) Mapping of cloud standards to the taxonomy of interoperability in IaaS. In: Proceedings of the 2011 IEEE third international conference on cloud computing technology and science, CLOUDCOM'11, pp 522–526. IEEE Computer Society, Washington, DC, USA. https://doi.org/10.1109/CloudCom.2011.78
66. Thramboulidis K, Christoulakis F (2016) UML4IoT—a UML-based approach to exploit iot in cyber-physical manufacturing systems. Comput Ind 82(C):259–272. https://doi.org/10.1016/j.compind.2016.05.010
67. Thuluva AS, Bröring A, Medagoda GP, Don H, Anicic D, Seeger J (2017) Recipes for iot applications. In: Proceedings of the seventh international conference on the internet of things, IoT'17, pp 10:1–10:8. ACM, New York, NY, USA. https://doi.org/10.1145/3131542.3131553
68. Truong HL, Berardinelli L (2017) Testing uncertainty of cyber-physical systems in IoT cloud infrastructures: combining model-driven engineering and elastic execution. In: Proceedings of the 1st ACM SIGSOFT international workshop on testing embedded and cyber-physical systems, TECPS 2017, pp 5–8. ACM, New York, NY, USA. https://doi.org/10.1145/3107091.3107093
69. Truong HL, Berardinelli L, Pavkovic I, Copil G (2017) Modeling and provisioning IoT cloud systems for testing uncertainties. In: Proceedings of the 14th EAI international conference on mobile and ubiquitous systems: computing, networking and services, MobiQuitous 2017. ACM. http://dsg.tuwien.ac.at/staff/truong/publications/2017/truong-mobiquitous2017.pdf
70. Truong HL, Copil G, Dustdar S, Le D, Moldovan D, Nastic S (2016) On engineering analytics for elastic IoT cloud platforms. In: Sheng QZ, Stroulia E, Tata S, Bhiri S (eds) Service-oriented computing—14th international conference, ICSOC 2016, Banff, AB, Canada, October 10–13, 2016, Proceedings, Lecture notes in computer science, vol 9936, pp 267–281. Springer. https://doi.org/10.1007/978-3-319-46295-0_17
71. Truong HL, Narendra NC (2016) SINC—an information-centric approach for end-to-end IoT cloud resource provisioning. In: International conference on cloud computing research and innovations, ICCCRI 2016, Singapore, Singapore, May 4–5, 2016, pp 17–24. IEEE Computer Society. https://doi.org/10.1109/ICCCRI.2016.12

72. U-Test H2020 Deliverable: Report on Uncertainty Modelling Framework V.2. https://www.simula.no/file/u-testd22pdf/download. Accessed 22 June 2017
73. van der Ham J, Stéger J, Laki S, Kryftis Y, Maglaris V, de Laat C (2015) The NOVI information models. Future Gener Comput Syst 42:64–73
74. Vukovic M (2015) Internet programmable IoT: on the role of apis in IoT: the internet of things (ubiquity symposium). Ubiquity 2015(November):3:1–3:10. https://doi.org/10.1145/2822873
75. Wen Z, Yang R, Garraghan P, Lin T, Xu J, Rovatsos M (2017) Fog orchestration for internet of things services. IEEE Internet Comput 21(2):16–24. https://doi.org/10.1109/MIC.2017.36
76. Zhang H, Meng C (2014) A multi-dimensional ontology-based IoT resource model. In: 2014 5th IEEE international conference on software engineering and service science (ICSESS), pp. 124–127. IEEE
77. Zhang M, Selic B, Ali S, Yue T, Okariz O, Norgren R (2016) Understanding uncertainty in cyber-physical systems: a conceptual model. In: Modelling foundations and applications—12th European conference, ECMFA 2016, held as part of STAF 2016, Vienna, Austria, July 6–7, 2016, Proceedings, pp 247–264
78. Zhang M, Selic B, Ali S, Yue T, Pradhan D, Nastic S, Truong HL, Schneider M, Bureck M, Hein C (2015) Understanding uncertainty in cyber-physical systems, https://www.simula.no/file/d12pdf/download. Technical report, U-Test Project
79. Zhang Z, Wu C, Cheung DW (2013) A survey on cloud interoperability: taxonomies, standards, and practice. SIGMETRICS Perform Eval Rev 40(4):13–22. https://doi.org/10.1145/2479942.2479945
80. Zoghi P, Shtern M, Litoiu M, Ghanbari H (2016) Designing adaptive applications deployed on cloud environments. ACM Trans Auton Adapt Syst 10(4):25:1–25:26. https://doi.org/10.1145/2822896