Reliable Capacity Provisioning for Distributed Cloud/Edge/Fog Computing Applications

Per-Olov Östberg[‡], James Byrne[✠], Paolo Casari[♯], Philip Eardley[*], Antonio Fernandez Anta[♯], Johan Forsman[◇], John Kennedy[⊕], Thang Le Duc[‡], Manuel Noya Mariño[▲], Radhika Loomba[⊕], Miguel Ángel López Peña[□], José López Veiga[▲], Theo Lynn[✠], Vincenzo Mancuso[♯], Sergej Svorobej[✠], Anders Torneus[◇], Stefan Wesner[§], Peter Willis[*], Jörg Domaschka[§]

[‡]Department of Computing Science, Umeå University, Sweden [✠]IC4, Dublin City University Business School, Dublin, Ireland
[♯]IMDEA Networks Institute, Madrid, Spain [*]BT R&I, Ipswich, UK [◇]Tieto AB, Umeå, Sweden [▲]Linknovate Science SL
[⊕]Intel Labs Europe, Leixlip, Ireland [□]Sistemas Avanzados de Tecnología S.A., Madrid, Spain [§]OMI, Ulm University, Germany

***Abstract*—The REliable CApacity Provisioning and enhanced remediation for distributed cloud applications (RECAP) project aims to advance cloud and edge computing technology, to develop mechanisms for reliable capacity provisioning, and to make application placement, infrastructure management, and capacity provisioning autonomous, predictable and optimized. This paper presents the RECAP vision for an integrated edge-cloud architecture, discusses the scientific foundation of the project, and outlines plans for toolsets for continuous data collection, application performance modeling, application and component auto-scaling and remediation, and deployment optimization. The paper also presents four use cases from complementing fields that will be used to showcase the advancements of RECAP.**

***Index Terms*—Cloud computing; capacity provisioning; application modeling; workload propagation; data collection; analytics; machine learning; simulation; optimization**

I. Introduction

For reasons of scale, robustness, cost and energy efficiency, large-scale computing systems today are typically built as distributed systems, whereby components and services are spread and accessed remotely through clients and devices. Particularly, in latency-sensitive systems, or in systems where high availability is important, it is common that some components are also placed closer to end-users, for example in radio base stations and other systems on the edge of access networks. This style of computing is often referred to as *edge* or *fog* computing, whereby the former typically refers to the location where services are instantiated, and the latter implies distribution of the computation, communication, and storage resources and services on or close to devices and systems in the control of end-users [1].

While recent years have seen significant advances in system instrumentation and data center energy efficiency and automation, computational resources and network capacity are often provisioned using best-effort models and coarse-grained quality-of-service (QoS) mechanisms. In a future networked society permeated by connected objects, such approaches will not be sustainable given the increased loads on networks and data centers. A similar manifestation of the limits of todays large-scale computing architectures is offered by the limited adoption of cloud infrastructures to deploy systems with low latency demands, such as telecommunications services.

In this paper we present the roadmap for a novel architecture for cloud/edge/fog computing capacity provisioning and remediation, based on targeted advances in cloud infrastructure optimization, simulation, and automation. More specifically, we argue that the next generation of distributed cloud architectures should be based on the modeling of complex applications and infrastructures using fine-grained and accurate application deployment and behavior models. This effort will make it possible to understand application component-level quality-of-service (QoS) models and workload models, which capture application and component load and capacity requirements, their variations over time, and their impact on the cloud architecture. On the one hand, for a fixed architecture and deployment configuration (possibly involving network-, context- and geographically distributed computing resources), such a scheme would allow the cloud service orchestrator to understand and predict the behavior of the application. On the other hand, modeling would make it possible to perform advanced "what-if" reasoning by considering different deployment strategies and through simulating the impact that the same application requirements and workload would impose on the new cloud configuration. In turn, this would allow network and service administrators to optimize data center infrastructure with scheduling systems, decentralized monitoring and load balancing, and systems management/control tools.

The deployment of QoS and workload models for applications and components can also double as a prediction tool for the proactive remediation of complex distributed systems and networks. By this we imply that, once the models have been learned, the evolution of the workload and application performance can be predicted over time so as to infer when the current service deployment configuration in the cloud will become insufficient to support the required QoS. This will be carried out through the concerted activation of: (i) the prediction of the evolution of workload and application performance; (ii) the simulation of different deployments, possibly involving different nodes and locations in the cloud/edge/fog; (iii) the optimization of the deployment given the output of simulations; and (iv) the relocation of services and application components where their execution guarantees the achievement of the required QoS, as driven by the optimization process. This will make it possible to automate the detection and correction of failures at the network and infrastructure levels while maintaining QoS.

To illustrate the RECAP vision, the concepts above are elaborated upon and four relevant use cases are introduced,

978–1–5386–3873–6/17/$31.00

which have been selected to be characterized by different types of constraints and to take advantage of the vision for enhanced capacity provisioning and service remediation in distributed cloud architectures. The research directions described in this paper will be explored within the umbrella of the RECAP project [2], whose overarching result will be the next generation of agile and optimized cloud computing systems. These outcomes will pave the way for a radically novel concept in the provision of cloud services, whereby services are elastically instantiated and provisioned close to the end-users that require them via self-configurable cloud computing systems.

The remainder of this paper is organized as follows: Section II elaborates on the challenges entailed by the inclusion of self-orchestration and remediation in cloud systems. The vision and the context offered by the RECAP project is described in Section III; relevant use cases in Section IV. A summary of related work and projects in this area is provided in Section V, before concluding in Section VI.

II. Challenges for a Self-orchestrated, Self-remediated Cloud System

The RECAP consortium envisages that the next generation of cloud systems will have the ability to automatically provision capacity and distribute workload dynamically in a cloud-to-edge-to-fog continuum. In the presence of events that may decrease QoS below acceptable levels, these cloud systems will be able to preemptively take reconfiguration and remediation actions in a fully automatic fashion. Such actions will be based on a continuous measurements of the performance of application components running in the cloud, and on the learning of models for the evolution and propagation of workload across distributed cloud computing servers.

To realize this vision, the future of resource provisioning is here argued to lie in software-defined infrastructures (SDIs). SDIs are realized as cyber-physical systems that dynamically and seamlessly distribute software components among a mix of resources. These include large, energy-efficient and scalable data centers (usually placed at remote locations and interconnected with high capacity networks), and low-latency edge data centers (smaller, placed closer to end users and interconnected at the edge of the access networks). To address these goals, we formulate an approach based on advances in three primary research areas: resource management; data science and data analytics; and intelligent automation.

Resource Management—Resource management in SDIs revolves around the control of cyber-physical systems where physical components are fully abstracted and controlled via software. Resource management addresses the questions of how much capacity and what type of resources to allocate to applications, and when and where to deploy resources in and between data centers. The managed entities of SDI environments are provided by low-level virtualization technologies, e.g., Virtual Machines (VMs), virtual networks, and containers operating in data centers. Such techniques deliver high levels of flexibility in the management of resource capacity. However, while virtualization is today a well-established technology, efficient resource management remains a significant challenge due to complexity and scale of resource provisioning in distributed cloud systems.

Resource management in SDI environments requires an understanding of a wide variety of system components (e.g., user behaviors, workload characteristics and variations, interactions among application components, and performance bottlenecks) and can be enacted through a range of subsystems (including capacity auto scaling, scheduling, differentiated QoS control, as well as higher level systems for the coordination of these). As a research topic, resource management covers a broad spectrum of approaches, ranging from the performance and manageability of virtualized entities to the autonomous management of the software-defined infrastructure. It is key to the development of dependable ICT infrastructures and of fundamental interest to both academia and industry.

Data Science and Analytics—*Data Science is the use of computer science, statistics, machine learning, visualization and human-computer interactions to collect, clean, integrate, analyze, visualize, and interact with data*[1] in order to turn such data into structured knowledge products. Big data and data science analytics are redefining how research is being done in a wide variety of fields. In order to support resource management, but also more importantly to refine and create new understanding of complex systems, efficient ways to (automatically) structure and analyze data are needed. The RECAP consortium employs analytics for the modeling and understanding of the complex interactions of distributed applications and infrastructure resource management over varying-quality networks, as well as for end-user behavior and workload characterization. In this context, analytics needs to be more distributed and real-time oriented, compared to today's solutions. The main challenges here include a need for pre-analytics and higher-order analytical techniques, where parts of the analysis needs to be performed in a distributed manner.

A wide range of use cases illustrate the utility of distributing some analysis near the user, with further (more computationally or data-demanding) analysis performed remotely on higher-capacity resources in data centers. Such use cases include interactive applications with high computational complexity and strict response time requirements, e.g., image analysis in augmented reality applications, remote operation and control systems, or autonomous vehicle systems.

Data analytics can provide input to decision making for up to tens or even hundreds of thousands of servers and their applications, where each server includes tens to hundreds of monitoring and data sources. RECAP focusses on identifying correlations, trends, customer preferences, and resource behaviors from structured, semi-structured, and unstructured data partially analyzed in real time. The results will be relevant to stakeholders in cloud resource management and data-driven applications in virtually all areas.

Intelligent automation—Cloud computing has been adopted at large by traditional ICT applications and businesses. However, this technology is comparatively less trusted for high-data security and high-risk services, where cloud adoption

[1]https://cise.ufl.edu/class/cis6930sp14ids/2.\%20Data\%20Science\%20overview.pdf

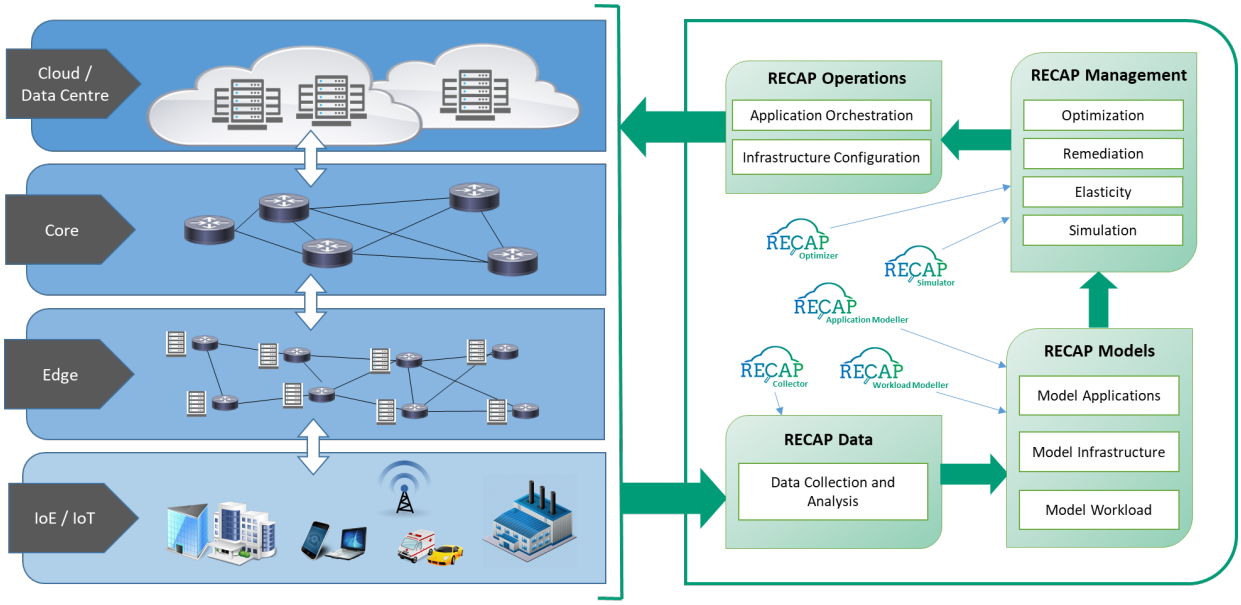

Fig. 1. A comprehensive view of the RECAP approach. Data collection and analysis at different locations where cloud services can be instantiated makes it possible to model application requirements, component-level workload and workload propagation. As a result, orchestration and remediation actions can be taken automatically, and can affect cloud capacity provisioning in remote data center-to-fog continuum.

remains low [3]. The main reason behind this is the cloud service provisioning model: cloud environments mainly provision capacity using shared resource aggregation schemes that are rarely able to guarantee prescribed QoS levels. Still, the commercial interest for dependable capacity provisioning with guaranteed QoS ICT for services in the cloud is very high: in fact, the distribution and data center hosting of ICT services and systems enables new services and applications (for example, instrumentation, live visualization, and analysis of data). At present, the uncertainties related to guaranteed QoS provisioning, automated management, and remediation in cloud systems tend to hamper the adoption of cloud technologies for ICT services.

The objective of the RECAP project's consortium is to address the challenges above, and fill the related technological gaps. The evolution of the networked society and the emerging challenges of the Internet of Everything (IoE) are expected to place new requirements on infrastructures. Notwithstanding this, new opportunities are expected for the development of advanced and intelligent automation systems. In particular, data centers are currently experiencing an unprecedented instrumentation growth, where much more fine-grained data is now available from monitoring systems. Many more types of actuators are also available for the control of resources and resource management systems. We aim to develop and contribute to the fundamental automation systems used in large-scale ICT resource provisioning to date. This will be achieved thanks to recent developments in data center technology, and will build on tools such as Software-Defined Networks (SDNs), disaggregated resource models, and large-scale resource management systems, in combination with established and new theoretical contributions from such fields as machine learning, control theory, optimization, and simulation.

III. The RECAP vision

In order to fulfill the vision conveyed in the previous sections, the RECAP consortium will define and implement a novel architecture to realize the ideas relating to resource management, data science and analytics, and intelligent automation. Fig. 1 illustrates different infrastructure categories where computing infrastructure can be located, including the access network (even down to the very devices connected to the network, in fog computing scenarios), the edge network, and the core network. "Local" data centers feature lower latency and capacity and are located close to the user, whereas larger data centers provide higher capacity at the cost of higher latency, and are placed at "remote" locations. Different components tap into this infrastructure and actuate on it to realize the architecture of interest in the RECAP project. The main components of this architecture are described as follows.

The RECAP Collector—This component gathers, synthesizes, and analyzes the relevant metrics to be monitored across the infrastructure, and in particular at the edge computing layer. It carries out data acquisition in the edge, and pre-processes application and infrastructure monitoring data; moreover, it applies data characterization and analysis algorithms to learn the relationship among workload patterns, the status of the infrastructure, and other key performance indicators (KPIs). It also provides options to visualize, annotate, archive, and manage collected data.

The RECAP Application Modeler—This component is tasked with discovering and defining the internal structure of cloud applications and their QoS requirements. This enables more intelligent decision making related to application placement, such as where to deploy and how to scale components. The RECAP Application Modeler will utilize techniques by which the internal structure of different applications can be inferred even when those applications are located at the

extreme edges of the network. This will be made possible in an autonomous fashion through the data gathered and analyzed by the RECAP Collector. Formal QoS requirements definitions at application- and component- level, as well as application characterization with respect to vertical and horizontal scaling tolerances will make it possible to build models that capture the intra- and inter-dependencies of application components, and to relate those to QoS delivery and infrastructure load.

The RECAP Workload Modeler—This component implements models for decomposition, classification, and prediction of workloads; as well as models for load propagation in applications. The RECAP Workload Modeler will be initially built along the lines of the existing approaches regarding workload classification and predictions. Notwithstanding this, current state-of-the-art approaches are not well suited to software-defined data centers or to edge computing scenarios, where application components may be even distributed across (even widely) different locations. The RECAP Workload Modeler will model and estimate how the load propagates across application components over different dimensions (CPU, memory, network, etc.), with a specific focus on the edge network infrastructure. Understanding load propagation patterns and correlating those to the load on the underlying computing infrastructure will enable the identification of bottlenecks which in turn will improve on planning decisions and ensure that desirable levels for application KPIs are met. The models built for this purpose will finally be leveraged to provide an artificial workload generation tool, which will allow for the validation and training of the workload decomposition and application load propagation models. This can then be delivered to the community as a supplement to address the lack of detailed traces measured from real systems.

The RECAP Optimizer—The planning and continuous optimization of the placement of virtual components on physical resources greatly influences the effectiveness and energy efficiency of data center resources. Infrastructure and application placement optimization actions are taken by the RECAP Optimizer in order to make more efficient use of resources and to maintain applications KPIs. The RECAP Optimizer takes advantage of the application and workload modeling performed by the previously defined RECAP components in order to make fully autonomous application placement (including vertical and horizontal scaling as well as trade-offs regarding migration costs) and infrastructure management decisions (taking into account energy efficiency, utilization rates, load balancing, saturation, etc.) throughout the network from the main data centers to the extreme edge. Ultimately, the RECAP Optimizer will achieve a mapping of application performance and load to actual resource capacity needs. This will make it possible to use infrastructure in an efficient and sustainable way through the accurate assignment of the right amount of a specific resource to a specific application.

The RECAP Simulator—Simulation technologies will be extensively employed in RECAP, to both simulate the interactions of distributed cloud application behaviors, and to emulate data center and connectivity networks systems. The complexity and size of the systems addressed are in themselves prohibitive for full-scale deployment, and will be studied at several levels in simulation. The RECAP Simulation Framework will assist the RECAP Optimizer in the evaluation of different deployment and infrastructure management alternatives in terms of cost, energy, resource allocation and utilization, before actuating on real application deployments. This will require accurate output through the monitoring of the status of data center infrastructure in order to provide an effective decision support instrument. This joint operation is of particular importance for understanding and managing trade-offs in edge and fog computing scenarios. Offline, bulkier simulation output data will be processed through the same analytics techniques employed in the RECAP Data Collector, in order to create a knowledge-base that will increase the speed of decision making and remediation in production environments.

IV. Example of use cases for the RECAP system

The RECAP concept architecture (cf. Section III) will be applied to four industry-focused use cases, representative of different types of applications with diverse requirements in terms of QoS and of interaction among local and remote cloud computing facilities. We briefly describe these use cases below.

Automatic profiling—Commercial telecommunications and service application providers are interested in the development of innovative solutions leveraging the possibilities enabled by 4G-and-beyond mobile technologies in conjunction with fog/edge/cloud computing. This includes (for example) eHealth, eCommerce and traffic safety services. For each of these scenarios, this use case will demonstrate how the profiling and simulation of infrastructure, network function, and service function characteristics can be automated to ensure the desired QoS is met for the different networks managed by the providers. Key to this will be the simulation of how a common physical infrastructure spanning over vertical regions with different characteristics can provide end-to-end communication and content services for different categories of horizontal content services (network slices) with different QoS requirements.

Automatically enriched data analytics—To demonstrate the use of RECAP technology in big data scenarios, this use case focuses on a big data analytics service that helps customers to identify emerging technologies, markets, and the key players behind them. This type of “competitive intelligence” is mainly based on a discovery engine deployed in the cloud, as opposed to a classic search engine. The engine is based on ElasticSearch (which provides scalability and performance) and Myriad open data (which provides visualization tools). These components of the search engine are deployed in the cloud, providing a big data search engine solution. Using RECAP technology this use case will demonstrate how complex applications and virtual data centers can be modeled and automatically improved in cloud environments, both reducing costs and improving performance.

Data processing in the fog—It is expected that the uptake of the Internet of Everything (IoE) and of smart cities will lead to a deluge of different kinds of data (e.g., pollution, traffic, weather, energy, etc.), generated by a great quantity of distributed, connected devices such as cameras, connected

vehicles, home appliances, and smart phones. The heart of the smart cities' intelligence lies in the data processing. In order to extract this value, the data needs to be transported through data networks to be stored and processed, and then moved back to the city data infrastructures where they can be forwarded to interested entities and authorities. This use case will illustrate how RECAP technology enables the storage and processing of data produced in local contexts as close as possible to those contexts in fog and edge computing scenarios.

Service remediation—Communication Service Providers (CSPs) use static provisioning rules or policies for placement of Virtual Network Functions (VNFs) on the Network Function Virtualization (NFV) Infrastructure. These rules are currently conservative, and under-utilize infrastructure to meet Service-Level Agreements (SLAs), because to meet the Key Performance Indicators (KPIs) of the SLAs the VNF vendors specify the CSP must provide dedicated resources to meet the worst case traffic type and the CSPs dimension to the peak traffic load. Compute resources cannot be oversubscribed or shared as this leads to poor network performance [4]. In addition, static rules also do not address untested combinations of VNFs, necessitating expensive and time-consuming testing of a full matrix of VNFs, thus preventing the CSPs and end-users from taking full advantage of the potential flexibility of NFV. Automated and intelligent mechanisms are required to ensure that VNFs are optimally placed, NFV infrastructure utilization is maximised, and new innovative services can be introduced quickly but safely and reliably without requiring extensive testing. In this use case the RECAP consortium will demonstrate how an intelligent system can model, predict, and prepare automated remediation responses to failures in advance and respond in near real time to repair faults that could not be predicted.

V. Related work

Modeling performance implies accounting for performance objectives from the early stages (the design phase) of the software development process. with the open question being how to define useful performance models for service-based systems. As such, defining useful performance models for service based systems becomes an essential task. Layered Queuing Networks (LQN) represent the theoretical base for many approaches in Performance Modeling, Analysis and Optimization [5], [6]. Mytilinis et al. [7] do not build on LQN but focus on improving I/O performance prediction of Big Data applications, based on the outcome of the EU projects ASAP [8] and CELAR []. Performance models can also be categorized with different granularities as per Grozev et al. [9]. Another approach is Q-Clouds [10], a QoS-aware control framework that tunes resource allocations to mitigate performance interference effects.

Multiple commercial tools exist with the aim of automating the delivery process of new and updated applications (commonly referred to as DevOps) such as Chef, Ansible, Puppet, and Salt. Each provides a domain-specific language allowing the user to describe the successive steps necessary to deploy a component to a single node in an existing infrastructure. Multiple commercial and open-source cloud orchestration frameworks exist such as Apache Brooklyn, Apache Stratos, Cloudify, and Slipstream. Model-driven engineering is an approach used for describing and deploying applications. Multiple modeling languages exist such as CAMP, TOSCA, CloudML, and CAMEL. While CAMP targets the Platform-as-a-Service (PaaS) layer,the others mainly target Infrastructure-as-a-Service (IaaS). Cloud providers also offer tools such as Amazons AWS CloudFormation, Heat for Openstack, Azures template deployments, or Googles deployment descriptors for App Engine. RoboConf [11] is a cloud orchestrator for deploying applications in a hybrid cloud environment.

Existing work in supporting decision making for the deployment of applications across cloud providers come from either the area of application migration to the cloud, or from the optimization of the application topology distribution. With respect to the former area, approaches such as Kingfisher [12] and CloudAdoption [13] aim to assist application designers in migrating their applications to the cloud. With respect to the latter, related work builds on distributed application topologies in order to optimize across different dimensions, typically including operational expenses. For example, in [14] an approach is proposed that matches and dynamically adapts the allocation of infrastructure resources to an application topology in order to ensure SLAs. CloudMig [15] builds on an initial topology of the application that is adapted through model transformation, in order to optimize the distribution of the application across different cloud offerings. A similar approach is proposed by the MODAClouds [16].

In relation to the RECAP Simulator, simulation research and development efforts are increasing towards carrying out experiments in distributed systems. These development efforts range from very specific component simulation to the simulation of very large systems such as cloud computing systems. For networking simulation, NS-3 [17], DaSSF [18] and OMNET++ [19] are commonly used. Several simulation tools support cloud computing simulation and the design of these simulators can be divided into two classes: layered architectures representing virtualized data center components (CloudSim [20], EMUSIM [21], DCSim [22], SPECI [23], Palladio [24], CactoSim [25]) and network-based components layout (GreenCloud [18], SimGrid [26]). As the RECAP Simulator is to assist the RECAP Optimizer in experimentation and evaluation, examples of related projects are the CACTOS project [27] and the CloudLightning project [28], where the simulation environment uses a footprint of the data center allowing the users to study different optimization strategies and VM placements.

In order to optimize the utilization of the applications' components, it is not only necessary to place the hosting virtual machines (or containers) in a meaningful way onto the available physical hosts, but also to select the best-suited virtual machine flavor for each specific component. Most of the literature considers the former, leading to a waste of resources and/or inappropriate application performance [29]. There are works which also consider the workload pattern while taking reconfiguration actions (such as scaling or (re)placement), of special importance in the edge computing deployment

scenario, as per by Zhang et al. [30]. Mistral [31] is a holistic controller framework that optimizes trade-offs among power consumption, application performance, and adaptation costs. Goudarzi et al. [32] focus on minimizing the total operational cost of the system including power and migration costs, and apply penalties for violating response time constraints. AGILE [33] dynamically and proactively adjusts the number of VM assigned to a cloud application in a way that minimizes the costs of infrastructure provisioning and penalties imposed due to service-level objectives (SLO) violations. Other examples include pMapper [34] and Sandpiper [35].

VI. Conclusions

This paper presents the vision of the RECAP project for the next generation of intelligent, self-managed and self-remediated cloud computing systems. The plan of the project is to i) endow distributed cloud infrastructures with the capability to collect data concerning application and component workload; ii) model such workloads and their propagation throughout applications; iii) simulate the effect of different decisions on application and component placement in terms of application performance and capacity provisioning, and finally iv) optimize the allocation of infrastructure resources throughout multiple data centers, through concertation between local data centers in the edge/fog and green data centers at remote geographical locations. The RECAP project runs until 2019 and research outputs will be presented through four use cases related to: automatic profiling; automatically enriched data analytics; data processing in the fog; and service remediation.

Acknowledgment

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 732667 (RECAP).

References

[1] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE IoT Journal*, vol. 3, no. 6, pp. 854–864, Dec. 2016.
[2] "RECAP web site," 2017. [Online]. Available: http://recap-project.eu/
[3] J. Mooney, J. Ross, and J. Phipps, "Embrace the inevitable: Six imperatives to prepare your company for cloud computing," *CISR Research Briefing*, vol. 12, no. 10, 2012.
[4] Intel Corporation, "End-to-end Optimized Network Function Virtualization Deployment," Whitepaper, 2015. [Online]. Available: http://www.intel.co.uk/content/dam/www/public/us/en/documents/white-papers/end-to-end-optimized-nfv-paper.pdf
[5] W. Zhang *et al.*, "PaaS-oriented performance modeling for cloud computing," in *Proc. IEEE COMPSAC*, Izmir, Turkey, Jul. 2012.
[6] Y. Shoaib and O. Das, "Web application performance modeling using layered queueing networks," *Elsevier Electronic Notes in Theoretical Computer Science*, vol. 275, pp. 123–142, Sep. 2011.
[7] I. Mytilinis *et al.*, "I/O performance modeling for big data applications over cloud infrastructures," in *Proc. IEEE IC2E*, Tempe, AZ, Mar. 2015.
[8] ASAP Consortium, "A Scalable Analytics Platform (ASAP) - EU FP7 Research Project," 2016. [Online]. Available: http://www.asap-fp7.eu/
[9] N. Grozev and R. Buyya, "Performance modelling and simulation of three-tier applications in cloud and multi-cloud environments," *The Computer Journal*, vol. 58, no. 1, pp. 1–22, Jan. 2015.
[10] R. Nathuji, A. Kansal, and A. Ghaffarkhah, "Q-clouds: managing performance interference effects for QoS-aware clouds," in *Proc. EuroSys*, Paris, France, Apr. 2010.
[11] L. M. Pham *et al.*, "Roboconf: A hybrid cloud orchestrator to deploy complex applications," in *Proc. IEEE CLOUD*, New York, NY, 2015.
[12] U. Sharma *et al.*, "Kingfisher: Cost-aware elasticity in the cloud," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011.
[13] A. Khajeh-Hosseini *et al.*, "The cloud adoption toolkit: supporting cloud adoption decisions in the enterprise," *Software: Practice and Experience*, vol. 42, no. 4, pp. 447–465, Apr. 2012.
[14] A. Antonescu, P. Robinson, and T. Braun, "Dynamic topology orchestration for distributed cloud-based applications," in *Proc. IEEE NCCA*, London, UK, Oct. 2012.
[15] S. Frey and W. Hasselbring, "The cloudmig approach: Model-based migration of software systems to cloud-optimized applications," *International Journal on Advances in Software*, vol. 4, pp. 342–353, 2011.
[16] D. Ardagna *et al.*, "MODACLOUDS : A model-driven approach for the design and execution of applications on multiple clouds," in *Proc. ICSE*, Zurich, Switzerland, Jun. 2012.
[17] K. Fall and K. Varadhan, "The network simulator (ns-2)." [Online]. Available: http://www.isi.edu/nsnam/ns/
[18] J. Liu *et al.*, "Performance prediction of a parallel simulator," in *Proc. PADS*, Atlanta, Georgia, May 1999.
[19] *The OMNeT++ Discrete Event Simulation System*, Prague, Czech Republic, Jun. 2001.
[20] T. Goyal, A. Singh, and A. Agrawa, "Cloudsim: Simulator for cloud computing infrastructure and modeling," pp. 3566–3572, 2012.
[21] R. N. Calheiros, R. Ranjan, and R. Buyya, "Virtual machine provisioning based on analytical performance and QoS in cloud computing environments," in *Proc. ICPP*, Taipei, Taiwan, Sep. 2011, pp. 295–304.
[22] M. Tighe, "DCSim: A data centre simulation tool for evaluating dynamic virtualized resource management," *Proc. IEEE CNSM*, Oct. 2012.
[23] I. L. Sriram and D. Cliff, "SPECI-2: An open-source framework for predictive simulation of cloud-scale data-centres," in *Proc. SIMULTECH*, Noordwijkerhout, The Netherlands, Jul. 2011.
[24] S. Becker, H. Koziolek, and R. Reussner, "The Palladio component model for model-driven performance prediction," *Journal of Systems and Software*, vol. 82, no. 1, pp. 3–22, Jan. 2009.
[25] S. Svorobej *et al.*, "Towards automated data-driven model creation for cloud computing simulation," in *Proceedings of the 8th International Conference on Simulation Tools and Techniques*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2015, pp. 248–255.
[26] H. Casanova, A. Legrand, and M. Quinson, "SimGrid: A generic framework for large-scale distributed experiments," *Proc. UKSIM*, Apr. 2008.
[27] P.-O. Östberg *et al.*, "The CACTOS vision of context-aware cloud topology optimization and simulation," in *Proc. IEEE CloudCom*, Dec. 2014.
[28] "CloudLightning H2020 Project," 2016. [Online]. Available: http://cloudlightning.eu/
[29] K. Qazi, Y. Li, and A. Sohn, "Workload prediction of virtual machines for harnessing data center resources," in *Proc. IEEE CLOUD*, Alaska, USA, Jun. 2014.
[30] L. Zhang *et al.*, "Service workload patterns for QoS-driven cloud resource management," *Journal of Cloud Computing*, vol. 4, no. 1, pp. 1–21, 2015.
[31] G. Jung *et al.*, "Mistral: Dynamically managing power, performance, and adaptation cost in cloud infrastructures," in *Proc. IEEE ICDCS*, Genoa, Italy, Jun. 2010.
[32] H. Goudarzi, M. Ghasemazar, and M. Pedram, "SLA-based optimization of power and migration cost in cloud computing," in *Proc. IEEE/ACM CCGrid*, Ottawa, Canada, may 2012.
[33] H. Nguyen *et al.*, "AGILE: Elastic distributed resource scaling for infrastructure-as-a-service," in *Proc. USENIX ICAC*, San Jose, CA, Jun. 2013.
[34] A. Verma, P. Ahuja, and A. Neogi, "pMapper: Power and migration cost aware application placement in virtualized systems," in *Proc. ACM/IFIP/USENIX Middleware*, Leuven, Belgium, Dec. 2008.
[35] T. Wood *et al.*, "Sandpiper: Black-box and gray-box resource management for virtual machines," *Computer Networks*, vol. 53, no. 17, pp. 2923–2938, Dec. 2009.