

EDGE COMPUTING AND CLOUDLETS: CHALLENGES AND OPPORTUNITIES

Richa

Research Scholar, Department of Computer Science, Punjabi University, Patiala, Punjab, India

Received: January 27, 2019

Accepted: March 04, 2019

ABSTRACT: Many cloud-based applications employ a data centre as a central server to process data that is generated by edge devices, such as smart phones, tablets and wearable. This model places ever increasing demands on communication and computational infrastructure with inevitable adverse effect on Quality- of-Service and Experience. The concept of Edge Computing is predicated on moving some of this computational load towards the edge of the network to harness computational capabilities that are currently untapped in edge nodes, such as base stations, routers and switches. This position paper considers the challenges and opportunities that arise out of this new direction in the computing landscape. The purpose of this article is to explain the importance of using Cloudlets in our everyday life, by presenting the main challenges and possibilities of further research. We present a short introduction about Cloudlets; the cloudlets architecture and the way they operate. Moreover, we discuss about what their features are, how they work with thin and tick clients, and how they operate as offload elements. In the end, we state the challenges and open research question related to the topic.

Key Words

INTRODUCTION

The article presented challenges and opportunities in cloud computing [1]. Cloud research since it has rapidly progressed paving the way for many competitors in a crowded marketplace. Although the vision of offering computing as a utility was achieved [2], the research space is still far from saturation and offers interesting opportunities.

Many cloud applications are user-driven, which has resulted in opportunities for large-scale data analytics. However, using the cloud as a centralized server simply increases the frequency of communication between user devices, such as smart phones, tablets, wearables and gadgets, we refer to as *edge devices*, and geographically distant *cloud* data centres. This is limiting for applications that require real-time response. Hence, there has been a need for looking 'beyond the clouds' towards the edge of the network as shown in Figure 1, we refer to as *edge computing* [3], [4], but is also known as *fog computing* [5], [6] or *cloudlet computing* [7], [8]. The aim is to explore possibilities of performing computations on nodes through which network traffic is directed, such as routers, switches and base stations, we refer to as *edge nodes*. The objective of this paper is to define the motivation, challenges and opportunities in edge computing, which is summarised in Figure 2.

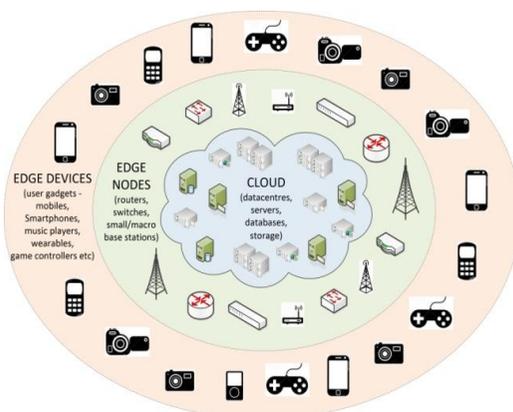


Fig. 1. Edge devices and edge nodes in relation to the cloud

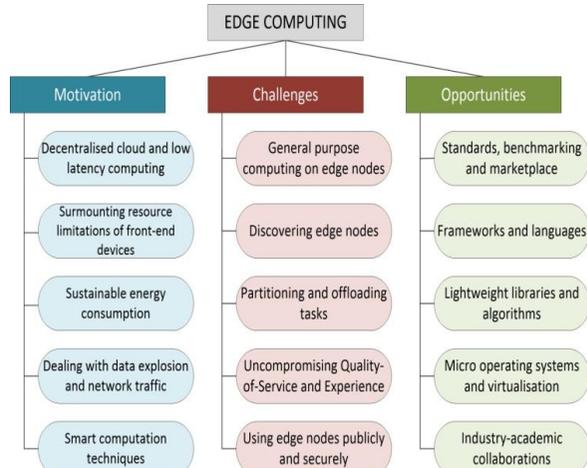


Fig. 2. Motivation, challenges and opportunities in edge computing

The wide availability of Internet access through mobile devices, such as phones, tablets, media players, allows users to access the cloud services while they are on the go. Currently there are more than 90 million mobile Internet users.

BACKGROUND

In this section we explain the main motivation for using cloudlets, and present a description of their architecture and the way they work.

Why Cloudlet?

The number of customers that use a mobile phone to connect to the Internet is constantly rising. It has long been recognized that mobile hardware is necessarily resource-poor relative to static client and server hardware. Mobile devices have more drawbacks compared to static clients and servers. Battery life, memory, weight and heat dissipation limit the computational resources of the device [2].

One of the solutions for increasing the problems about mobile device limitations can be cloud computing.

However, WAN delays in the critical path of user interaction can hurt usability by degrading the crispness of system response. Even trivial user-application interaction incur these delays in cloud computing.

That is the reason why cloudlet needs to be implemented. It is a small cloud located nearby mobile users which will be connected through LAN network with clouds located far away [3].

One of the cloudlet benefits is definitely a possibility that mobile users can rapidly instantiate custom virtual machines (VMs) on the cloudlet running the required software in a thin client fashion.

What is Cloudlet?

Cloudlet is a small cloud located close to the mobile users [4]. It is installed on discoverable, localized, stateless servers running one or more virtual machines (VMs) on which mobile devices can offload expensive computations.

A cloudlet is a set of trusted, resource-rich computers that is well-connected to the Internet and is available for use by nearby mobile devices. Cloudlets do not have to be a fixed infrastructure close to the wireless access point, but instead can be formed in a dynamic way with any device in the LAN network with available resources.

The basic cloud infrastructure is located far from the mobile user. For example, Amazon's EC2 infrastructure is located only in 6 cities worldwide. So when the customer is trying to reach one of these sites from his mobile device,

its "end to end" path would include many network segments. High WAN latencies make this approach of direct using the clouds insufficient for real-time applications of mobile users.

MOTIVATION

We have identified the following five needs that motivate computing on edge nodes.

1) Decentralized Cloud and Low Latency Computing

Centralized cloud computing may not always be the best strategy for applications that are geographically distributed. Computing needs to be performed closer to the source of the data to improve the service that is to be delivered. This benefit can

be generalised for any web-based application [9]. Location-aware applications such as Foursquare² and Google Now³ are becoming popular among mobile users. Computing on edge nodes closer to application users could be exploited as a platform for application providers to improve their service. Numerous data streams are generated by edge devices and making real-time decisions is not possible when analytics is performed on a distant cloud. Real-time applications, such as a visual guiding service using a wearable camera have a preferred response time between 25ms to 50ms [10]. The use of current cloud infrastructures poses a serious latency challenge between an edge device and the cloud. The round trip time from Canberra to Berkeley is approximately 175ms [7], which are far from requirements of latency-sensitive applications. Video streaming is the largest mobile traffic generator⁴ and satisfying users remains challenging if the bulk of this traffic needs to be channeled to the same source. Similarly, multimedia applications, such as on-demand gaming on current cloud infrastructure, pose similar latency issues for the gamer [11]. Here, to complement the computations performed on the cloud, edge nodes that are located closer (for example, routers or base stations one-hop away from an edge device) to users can be leveraged to reduce network latency.

2) Surmounting Resource Limitations of Front-end Devices

User devices, such as smart phones, have relatively restricted hardware resources when compared to a server in a data centre. These front-end devices capture sensory input in the form of text, audio, video, touch or motion and are processed by a service provided by the cloud [12]. The front-end devices cannot perform complex analytics due to middleware and hardware limitations [13]; this may sometimes be possible at the expense of draining the battery. Hence,

often data needs to be sent to the cloud to meet the computational demands of processing data and meaningful information is then relayed back to the front-end. However, not all data from a front-end device will need to be used by the service to construct analytical workloads on the cloud. Potentially, data can be filtered or even analysed at edge nodes, which may have spare computational resources to accommodate data management tasks.

3) Sustainable Energy Consumption

There is a significant body of research that has investigated the energy consumption of cloud data centres [14], [15]. Data centres in the next decade are likely to consume three times as much energy consumed today [5] and there is a greater need for adopting energy efficient strategies that can minimise the energy usage [6]. With more and more applications moving on to the cloud it may become untenable to meet the increasing energy demands. The challenge of increasing energy could be alleviated in a small proportion by incorporating sensible power management strategies in that a number of analytical tasks are performed on edge nodes, such as base stations or routers, closer to the data source instead of overloading data centres with trivial tasks that could perhaps be performed on edge nodes without significant energy implications.

4) Dealing with Data Explosion and Network Traffic

The number of edge devices is growing at an enormous rate and one-third of the world's population is expected to have a smart phone by 2018 [7]. Consequently, the volume of data that will be generated will also increase [9]; it is anticipated that 43 trillion gigabytes of data will be generated in 2020 [10]. This places the need for expanding data centers to support monitoring and analytical workloads, but this again raises concerns with regard to sustainable energy consumption of data centers. There are attempts to mitigate the energy challenge by performing analytics on the edge device [16], [17], [18]. However, this is restrictive due to resource limitations in the edge devices and collective analytics (of multiple edge devices) cannot be realistically done on an edge device. Another concern with increasing data generation is the volume of network traffic to a central server or cloud thereby reducing the response time of edge devices. This presents the potential of using nodes that are a hop away in the network for complementing computations of the device or of the data center to cope with the growth of data as well as distributing traffic in a network.

5) Smart Computation Techniques

Data generated at a user end needs to be often transported to a cloud server for performing any meaningful analytics which has obvious latency and energy implications. However, there is potential to harness resources at the far end of the network if computation can be hierarchically distributed [19]. For example, a typical application pipeline may initially filter data generated on the device, after which analytical workloads are executed on the edge nodes through which data is transmitted, before finally arriving at the cloud server where more complex tasks are performed. Alternatively, it may be possible for data centres to offload computations requiring limited resources on to edge nodes, or for the edge nodes to make use of volunteer devices to enhance computational capabilities [20]. Edge nodes can facilitate computations nearer to the source of data (or where data is generated) and can incorporate strategies for remotely enhancing capabilities of front-end devices.

CHALLENGES

Edge computing is still in its infancy and a framework to facilitate this is not yet available. Such frameworks will need to satisfy requirements, such as application development to process requests in real-time on edge nodes. Current cloud computing frameworks, such as the Amazon Web Service [11], Microsoft Azure [12] and Google App Engine [13], can support data-intensive applications, but implementing real-time data processing at the edge of the network is still an open research area [21], [22]. Additionally, the requirement of deploying application workloads on edge nodes will need to be well understood. Deployment strategies - where to place a workload, connection policies - when to use edge nodes and heterogeneity - how to deal with different types of nodes need to be taken into account for deploying applications on the edge. For achieving such a framework, we envisage that the following five research challenges at the hardware, middleware and software layer will need to be addressed.

Challenge 1 - General Purpose Computing on Edge Nodes

In theory, edge computing can be facilitated on several nodes that are located between the edge device and the cloud, including access points, base stations, gateways, traffic aggregation points, routers, switches, etc. Base stations, for example, incorporate Digital Signal Processors (DSPs) that are customised to the workloads they handle. In practice, base stations may not be suitable for handling analytical workloads simply because DSPs are not designed for general purpose

computing. Moreover, it is not readily known if these nodes can perform computations in addition to their existing workloads. The OCTEON Fusion@Family¹⁴ by CAVIUM, a small cell "Base Station-on-a-chip" family, scales from 6 to 14 cores to support users ranging from 32 to 300+. Such base stations could perhaps be used during off-peak hours to exploit the computational capabilities of multiple computing cores available. A number of commercial vendors have taken a first step to realise edge computing using software solutions. For example, Nokia's software solution¹⁵ for mobile edge computing (MEC) aims to enable base station sites for edge computing. Similarly, Cisco's IOx¹⁶ offers an execution environment for its integrated service routers. These solutions are specific to hardware and hence may not be suitable in a heterogeneous environment. One challenge in the software space will be to develop solutions that are portable across different environments.

There is research in upgrading the resources of edge nodes to support general purpose computing. For example, a wireless home router can be upgraded to support additional workloads [23]. Intel's Smart Cell Platform¹⁷ uses virtualisation for supporting additional workloads. Replacing specialised DSPs with comparable general purpose CPUs gives an alternative solution but this requires a huge investment.

Challenge 2 - Discovering Edge Nodes

Discovering resources and services in a distributed computing environment is an area that is well explored. This is facilitated in both tightly and loosely coupled environments through a variety of techniques that are incorporated into monitoring tools [24], [25], [26], [27] and service brokerages [28], [29], [30], [31]. Techniques such as benchmarking underpin decision-making for mapping tasks onto the most suitable resources for improving performance.

However, exploiting the edge of the network requires discovery mechanisms to find appropriate nodes that can be leveraged in a decentralised cloud set up. These mechanisms cannot be simply manual due to the sheer volume of devices that will be available at this layer. Moreover, they will need to cater for heterogeneous devices from multiple generations as well as modern workloads, for example large scale machine learning tasks, which were previously not considered. Benchmarking methods will need to be significantly rapid in making known the availability and capability of resources. These mechanisms must allow for

seamless integration (and removal) of nodes in the computational workflow at different hierarchical levels without increasing latencies or compromising the user experience. Reliably and proactively dealing with faults on the node and autonomically recovering from them will be desirable. Existing methods used in the cloud will not be practical in this context for the discovery of edge nodes.

Challenge 4 - Partitioning and Offloading Tasks

Evolving distributed computing environments have resulted in the development of numerous techniques to facilitate partitioning of tasks that can be executed at multiple geographic locations [32], [33]. For example, workflows are partitioned for execution in different locations [34], [35]. Task partitioning is usually expressed explicitly in a language or management tool. However, making use of edge nodes for offloading computations poses the challenge of not simply partitioning computational tasks efficiently, but doing this in an automated manner without necessarily requiring to explicitly define the capabilities or location of edge nodes. The user of a language that can leverage edge nodes may anticipate flexibility to define a computation pipeline - hierarchically in sequence (first at the data centre then at the edge nodes or first at the edge node and then at the data centre) or potentially over multiple edge nodes simultaneously. Inherently, there arises the need for developing schedulers that deploy partitioned tasks onto edge nodes.

Challenge 5 - Uncompromising Quality-of-Service (QoS) and Experience (QoE)

Quality delivered by the edge nodes can be captured by QoS and quality delivered to the user by QoE. One principle that will need to be adopted in edge computing is to not overload nodes with computationally intensive workloads [36], [37]. The challenge here is to ensure that the nodes achieve high throughput and are reliable when delivering for their intended workloads if they accommodate additional workloads from a data center or from edge devices. Regardless of whether an edge node is exploited, the user of an edge device or a data centre expects a minimum level of service. *Challenge 3 - Using Edge Nodes Publicly and Securely*

Hardware resources that are owned by data centres, supercomputing centres and private organisations using virtualisation can be transformed to offer computing as a utility. The associated risks for a provider and users are articulated¹⁹, thereby offering computing on a pay-as-you-go basis. This has resulted in a

competitive marketplace with numerous options and choices to satisfy computing consumers by meeting Service Level Agreements (SLAs) [38]. However, if alternative devices, such as switches, routers and base stations, need to be used as publicly accessible edge nodes a number of challenges will need to be addressed. The risk associated by public and private organizations.

OPPORTUNITIES

Despite challenges that arise when realising edge computing, there are numerous opportunities for academic research. We identify five such opportunities.

Opportunity 1 - Standards, Benchmarks and Marketplace

Edge computing can be realised in practice and be made accessible publicly if responsibilities, relationships and risks of all parties involved are articulated. There are numerous efforts to define a variety of cloud standards, such as by National Institutes of Standards and Technology (NIST) [2021], IEEE Standards Association [22], International Standards Organisation (ISO) [23], Cloud Standards Customer Council (CSCC) [24] and the International Telecommunication Union (ITU) [25]. However, such standards will now need to be reconsidered in light of additional stakeholders, such as public and private organisations that own edge nodes, to define the social, legal and ethical aspects of using edge nodes. This is certainly not an easy task and requires commitment and investment from public and private organisations and academic institutions.

Standards can be implemented only if the performance of edge nodes can be reliably benchmarked against well known metrics. Benchmarking initiatives for the cloud include those by the Standard Performance Evaluation Corporation (SPEC) [26] and by numerous academic researchers [40], [41], [42], [43]. In a noisy environment, such as the cloud, benchmarking poses significant challenges. The current state-of-the-art is not yet mature and significant research is required to deliver comprehensive benchmarking suites that can gather metrics accurately. Benchmarking edge nodes will therefore be more challenging, but opens new avenues of research.

Using edge nodes is an attractive prospective when responsibilities, relationships and risks are defined. Similar to a cloud marketplace, an edge computing marketplace that offers a variety of edge nodes on a pay-as-you-go basis is feasible. Research in defining SLAs for edge nodes and

pricing models will be required to create such a marketplace.

Opportunity 2 - Frameworks and Languages

There are many options to execute applications in the cloud paradigm. In addition to popular programming languages, there is a wide variety of services to deploy applications on the cloud. When resources outside the cloud are employed, for example, running a bioinformatics workload on the public cloud where input data is obtained from a private database, a workflow is usually employed. Software frameworks and toolkits for programming large workflows in a distributed environment is a well defined research avenue [44], [45].

However, with the addition of edge nodes that may support general purpose computing, there will be the need for developing frameworks and toolkits. The use cases for edge analytics are likely to differ from existing workflows, which are mostly explored for scientific domains, such as bioinformatics [46] or astronomy [47]. Given that edge analytics will find its use-cases in user-driven applications, the existing frameworks may not be appropriate to express an edge analytics workflow. The programming model that aims to exploit edge nodes will need to support task and data level parallelism and at the same time execute workloads on multiple hierarchical levels of hardware. The language that supports the programming model will need to take into account the heterogeneity of hardware and the capacity of resources in the workflow. If edge nodes are more vendor specific, then the frameworks supporting the workflow

will need to account for it. This is more complex than existing models that make the cloud accessible.

Opportunity 3 - Lightweight Libraries and Algorithms

Unlike large servers edge nodes will not support heavy-weight software due to hardware constraints. For instance, a small cell base station with Intel's T3K Concurrent Dual-Mode system-on-chip (SoC) [27] has a 4-core ARM-based CPU and limited memory, which will not be sufficient for executing complex data processing tools such as Apache Spark [28] that requires at least 8 cores CPU and 8 gigabyte memory for good performance. Edge analytics require lightweight algorithms that can do reasonable machine learning or data processing tasks [48], [49]. Apache Quarks [29], for example, is a lightweight library that can be employed on small footprint edge devices such as smart phones to enable real-time data analytics. However, Quarks supports basic data processing, such as filtering and windowed aggregates, which

are not sufficient for advanced analytical tasks such as context-aware recommendations. Machine learning libraries that consume less memory and disk usage would benefit data analytical tools for edge nodes. TensorFlow³⁰ is another example framework that supports deep learning algorithms and supports heterogeneous distributed systems, but its potential for edge analytics is still to be explored.

Opportunity 4 - Micro Operating Systems and Virtualisation

Research in micro operating systems or microkernels can provide inroads to tackling challenges related to deployment of applications on heterogeneous edge nodes. Given that these nodes do not have substantial resources like in a server, the general purpose computing environment that is facilitated on the edge will need to exhaust fewer resources. The benefits of quick deployment, reduced boot up times and resource isolation are desirable [50]. There is preliminary research suggesting that mobile containers that multiplex device hardware across multiple virtual devices can provide similar performance to native hardware [51]. Container technologies, such as Docker³¹ are maturing and enable quick deployment of applications on heterogeneous platform. More research is required to adopt containers as a suitable mechanism for deploying applications on edge nodes.

Opportunity 5 - Industry-Academic Collaborations

Edge computing offers a unique opportunity for academia to re-focus its research activities broadly in applied distributed computing, specifically within cloud and mobile computing. It is not easy for academic research to focus on scale without making assumptions that may not always correspond to reality. This is simply because a large number of academic institutions and researchers cannot gain access to infrastructure owned by the industry or government to validate and refine their research [52]. Premier academic institutions that have reliable industry and government relations have however produced more meaningful and impactful research [53].

Research in the edge computing space can be driven by an open consortium of industry partners, such as mobile operators and developers, gadget developers and cloud providers, as well as interested academic partners to the mutual benefit of both.

CONCLUSION

Edge computing is still in its infancy and has potential to pave the way for more efficient distributed computing. This paper has highlighted the significant potential of computing at the edge of the network and presented five research challenges and five rewarding opportunities in realising edge computing.

REFERENCES

1. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski,
2. G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
3. R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599 – 616, 2009.
4. P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, Iammitchi, M. Barcellos, P. Felber, and E. Riviere, "Edge-centric Computing: Vision and Challenges," *SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 37–42, Sep. 2015.
5. M. Satyanarayanan, P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha,
6. Hu, and B. Amos, "Edge Analytics in the Internet of Things," *IEEE Pervasive Computing*, vol. 14, no. 2, pp. 24–31, Apr 2015.
7. T. H. Luan, L. Gao, Z. Li, Y. Xiang, and L. Sun, "Fog Computing: Focusing on Mobile Users at the Edge," *CoRR*, vol. abs/1502.01815, 2015. [Online]. Available: <http://arxiv.org/abs/1502.01815>
8. F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," in *Proceedings of the Workshop on Mobile Cloud Computing*, 2012, pp. 13–16.
9. M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.
10. K. Gai, M. Qiu, H. Zhao, L. Tao, and Z. Zong, "Dynamic Energy-aware Cloudlet-based Mobile Cloud Computing Model for Green Computing," *Journal of Network and Computer Applications*, vol. 59, no. C, pp. 46– 54, 2016.
11. J. Zhu, D. S. Chan, M. S. Prabhu, P. Natarajan, H. Hu, and F. Bonomi, "Improving Web Sites Performance Using Edge Servers in Fog Computing Architecture," in *Proceedings of the International Symposium on Service Oriented System Engineering*, 2013, pp. 320–323.
12. S. Agarwal, M. Philipose, and P. Bahl, "Vision: The Case for Cellular Small Cells for Cloudlets," in *Proceedings of the International Workshop*

- on Mobile Cloud Computing & Services, 2014, pp. 1–5.
13. S. Choy, B. Wong, G. Simon, and C. Rosenberg, "The Brewing Storm in Cloud Gaming: A Measurement Study on Cloud to End-user Latency," in Proceedings of the Workshop on Network and Systems Support for Games, 2012.
 14. D. Damopoulos, G. Kambourakis, and G. Portokalidis, "The Best of Both Worlds: A Framework for the Synergistic Operation of Host and Cloud Anomaly-based IDS for Smartphones," in Proceedings of the European Workshop on System Security, 2014, pp. 6:1–6:6.
 15. A. Houmansadr, S. A. Zonouz, and R. Berthier, "A Cloud-based Intrusion Detection and Response System for Mobile Phones," in Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, 2011, pp. 31–32.
 16. A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing," *Future Generation Computer Systems*, vol. 28, no. 5, pp. 755–768, May 2012.
 17. A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The Cost of a Cloud: Research Problems in Data Center Networks," *SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 68–73, Dec. 2008.
 18. M. Sun, M. Zheng, J. C. S. Lui, and X. Jiang, "Design and implementation of an android host-based intrusion prevention system," in Proceedings of the Annual Computer Security Applications Conference, 2014, pp. 226–235.
 19. A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer, and Y. Weiss, "Andromaly: A Behavioral Malware Detection Framework for Android Devices," *Journal of Intelligent Information Systems*, vol. 38, no. 1, pp. 161–190, Feb. 2012.
 20. G. Dini, F. Martinelli, A. Saracino, and D. Sgandurra, "MADAM: A Multi-level Anomaly Detector for Android Malware," in Proceedings of the International Conference on Mathematical Methods, Models and Architectures for Computer Network Security: Computer Network Security, 2012, pp. 240–253.
 21. P. Hari, K. Ko, E. Koukoumidis, U. Kremer, M. Martonosi, D. Ottoni, L.-S. Peh, and P. Zhang, "SARANA: Language, Compiler and Run-time System Support for Spatially Aware and Resource-aware Mobile Computing," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1881, pp. 3699–3708, 2008.
 22. M. Shiraz and A. Gani, "A Lightweight Active Service Migration Framework for Computational Offloading in Mobile Cloud Computing," *The Journal of Supercomputing*, vol. 68, no. 2, pp. 978–995, 2014.
 23. W. Li, Y. Zhao, S. Lu, and D. Chen, "Mechanisms and challenges on Mobility-augmented Service Provisioning for Mobile Cloud Computing," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 89–97, 2015.
 24. H. Hromic, D. Le Phuoc, M. Serrano, A. Antonic, I. P. Zarko, C. Hayes, and S. Decker, "Real Time Analysis of Sensor Data for the Internet of Things by Means of Clustering and Event Processing," in Proceedings of the IEEE International Conference on Communications, 2015, pp. 685–691.
 25. C. Meurisch, A. Seeliger, B. Schmidt, I. Schweizer, F. Kaup, and
 26. M. Muhlhauser, "Upgrading Wireless Home Routers for Enabling Large-scale Deployment of Cloudlets," in *Mobile Computing, Applications, and Services*, 2015, pp. 12–29.
 27. J. Povedano-Molina, J. M. Lopez-Vega, J. M. Lopez-Soler, A. Corradi, and L. Foschini, "DARGOS: A Highly Adaptable and Scalable Monitoring Architecture for Multi-tenant Clouds," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 2041–2056, 2013.
 28. S. A. D. Chaves, R. B. Uriarte, and C. B. Westphall, "Toward an Architecture for Monitoring Private Clouds," *IEEE Communications Magazine*, vol. 49, no. 12, pp. 130–137, 2011.
 29. J. Montes, A. Sánchez, B. Memishi, M. S. Pérez, and G. Antoniu, "GMonE: A Complete Approach to Cloud Monitoring," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 2026–2040, 2013.
 30. J. S. Ward and A. Barker, "Varanus: In Situ Monitoring for Large Scale Cloud Systems," in Proceedings of the IEEE International Conference on Cloud Computing Technology and Science, vol. 2, 2013, pp. 341–344.
 31. A. Li, X. Zong, S. Kandula, X. Yang, and M. Zhang, "CloudProphet: Towards Application Performance Prediction in Cloud," *SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 426–427, 2011.
 32. N. Grozev and R. Buyya, "Inter-Cloud Architectures and Application Brokering: Taxonomy and Survey," *Software: Practice and Experience*, vol. 44, no. 3, pp. 369–390, 2014.
 33. S. K. Garg, S. Versteeg, and R. Buyya, "A Framework for Ranking of Cloud Computing Services," *Future Generation Computer Systems*, vol. 29, no. 4, pp. 1012–1023, 2013.
 34. D. Yin and T. Kosar, "A Data-aware Workflow Scheduling Algorithm for Heterogeneous Distributed Systems," in *International Conference on High Performance Computing and Simulation*, 2011, pp. 114–120.
 35. T. Ghafariana and B. Javadi, "Cloud-aware Data Intensive Workflow Scheduling on Volunteer Computing Systems," *Future Generation Computer Systems*, vol. 51, pp. 87–97, 2015.

36. W. Chen and E. Deelman, "Integration of Workflow Partitioning and Resource Provisioning," in Proceedings of the IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 764–768.
37. W. Tang, J. Jenkins, F. Meyer, R. Ross, R. Kettimuthu, L. Winkler,
38. Yang, T. Lehman, and N. Desai, "Data-Aware Resource Scheduling for Multicloud Workflows: A Fine-Grained Simulation Approach," in Proceedings of the IEEE International Conference on Cloud Computing Technology and Science, 2014, pp. 887–892.
39. M. T. Beck and M. Maier, "Mobile Edge Computing: Challenges for Future Virtual Network Embedding Algorithms," in Proceedings of the International Conference on Advanced Engineering Computing and Applications in Sciences, 2014, pp. 65–70.
40. P. Simoens, L. Van Herzeele, F. Vandeputte, and L. Vermoesen, "Challenges for Orchestration and Instance Selection of Composite Services in Distributed Edge Clouds," in Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management, 2015, pp. 1196–1201.
41. S. A. Baset, "Cloud SLAs: Present and Future," ACM SIGOPS Operating Systems Review, vol. 46, no. 2, pp. 57–66, 2012.
42. T. Bui, "Analysis of Docker Security," CoRR, vol. abs/1501.02967, 2015. [Online]. Available: <http://arxiv.org/abs/1501.02967>
43. B. Varghese, L. Subba, L. Thai, and A. Barker, "DocLite: A Docker- Based Lightweight Cloud Benchmarking Tool," in Proceedings of the IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2016.
44. A. Iosup, S. Ostermann, N. Yigitbasi, R. Prodan, T. Fahringer, and
45. D. Epema, "Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing," IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 6, pp. 931–945, 2011.
46. B. Varghese, O. Akgun, I. Miguel, L. Thai, and A. Barker, "Cloud Benchmarking for Performance," in Proceedings of the IEEE International Conference on Cloud Computing Technology and Science, 2014, pp. 535–540.
47. B. F. Cooper, A. Silberstein, E. Tam, R. Ramakrishnan, and R. Sears, "Benchmarking Cloud Serving Systems with YCSB," in Proceedings of the ACM Symposium on Cloud Computing, 2010, pp. 143–154.
48. E. Deelman, K. Vahi, G. Juve, M. Rynge, S. Callaghan, P. J. Maechling,
49. Mayani, W. Chen, R. Ferreira da Silva, M. Livny, and K. Wenger, "Pegasus: A Workflow Management System for Science Automation," Future Generation Computer Systems, vol. 46, pp. 17–35, 2015.
50. J. Goecks, A. Nekrutenko, and J. Taylor, "Galaxy: A Comprehensive Approach for Supporting Accessible, Reproducible, and Transparent Computational Research in the Life Sciences," Genome Biology, vol. 11, no. 8, pp. 1–13, 2010.
51. J. Ruiz, J. Garrido, J. Santander-Vela, S. Sanchez-Exposito, and L. Verdes-Montenegro, "AstroTaverna-Building Workflows with Virtual Observatory Services," Astronomy and Computing, vol. 78, pp. 3 – 11, 2014.
52. S. Kartakis and J. A. McCann, "Real-time Edge Analytics for Cyber Physical Systems Using Compression Rates," in Proceedings of the International Conference on Autonomic Computing, 2014, pp. 153–159.
53. I. Santos, M. Tilly, B. Chandramouli, and J. Goldstein, "DiAl: Distributed Streaming Analytics Anywhere, Anytime," Proceedings of the VLDB Endowment, vol. 6, no. 12, pp. 1386–1389, 2013.
54. L. Xu, Z. Wang, and W. Chen, "The Study and Evaluation of ARM- based Mobile Virtualization," International Journal of Distributed Sensor Networks, Jan. 2015.
55. J. Andrus, C. Dall, A. V. Hof, O. Laadan, and J. Nieh, "Cells: A Virtual Mobile Smartphone Architecture," in Proceedings of the ACM Symposium on Operating Systems Principles, 2011, pp. 173–187.
56. M. Schwarzkopf, D. Murray, and S. Hand, "The Seven Deadly Sins of Cloud Computing Research," in Proceedings of the USENIX Conference on Hot Topics in Cloud Computing, 2012.
57. A. Barker, B. Varghese, J. S. Ward, and I. Sommerville, "Academic Cloud Computing Research: Five Pitfalls and Five Opportunities," in Proceedings of the USENIX Conference on Hot Topics in Cloud Computing, 2013.