Big Data and Fog Computing [1]

Yogesh Simmhan
Indian Institute of Science, Bangalore, India
simmhan@iisc.ac.in

December 29, 2017

1 Synonyms

Cloudlets

2 Definitions

Fog Computing A model of distributed computing comprising of virtualized, heterogeneous, commodity computing and storage resources for hosting applications, analytics, content and services, that are accessible with low latency from the edge of wide area networks where clients are present, while also having back-end connectivity to cloud computing resources.

3 Background

Ever since computers could connect over a network, computing paradigms have under-gone cyclical phases on *where* within the network the computation is performed. While the original mainframes till the 1970's were large, centralized time-sharing machines accessed by multiple users through remote terminals, the Personal Computers (PCs) of the 1980's heralded local processing for individuals [35]. The growth of Local-Area Networks (LAN), the Internet, and the World Wide Web (WWW) brought about client-server models in the 1990's where many clients could access content hosted on individual servers, though most of the processing was still done on the PC [48]. Web services and eCommerce of the 2000's lead to the growth of *cloud computing*, where computation once again skewed to centralized data centers, but with server-farms rather than single servers hosting services that were consumed by PCs [27]. A complementary phenomenon in that decade was Peer-to-Peer (P2P) systems where PCs distributed across the Internet would work collaboratively for content sharing, to tackle bandwidth limitations of the Internet [32]. Both cloud computing and P2P signaled the arrival of the *Big Data* age, where the ability to collect large enterprise, scientific and web datasets and media content put an emphasis on being able to share and process them at large scales.

The decade of 2010 is seeing a similar cyclical shift, but at a faster pace due to several technology advances. Starting with a more centralized cloud computing model hosting thousands of Virtual Machines (VMs), we have seen the roll-out of pervasive broadband Internet and cellular network communication combined with the rapid growth of smart phones as general-purpose computing platforms backed by cloud computing. *Internet of Things (IoT)* is yet another paradigm, enabled by the convergence of these other technologies [21]. Sensors and constrained devices connected to the Internet are being deployed to support vertical IoT domains such as personal fitness using wearables, smart utilities using metering infrastructure, and even self-driving cars. Both smart phones and IoT mark the advent of *Edge Computing* (or mobile cloud computing). Here, ubiquitous devices numbering in the billions are present at the edge of the Wide-Area Network (WAN) that is the Internet, and host applications that
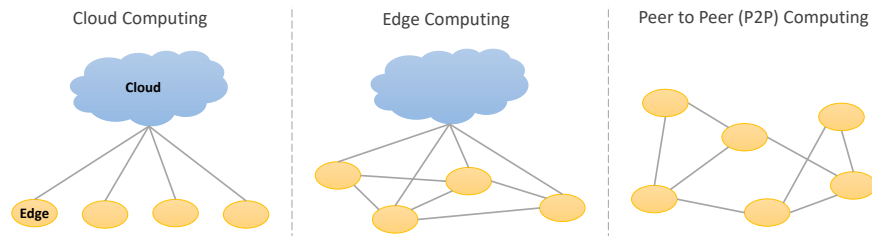
Figure 1: Connectivity between resources in Cloud, Edge and P2P models.

either operate locally or serve as light-weight clients that publish data to or consume content from cloud services [19, 18].

The limitations of an edge-only or a cloud-only model was recognized by Satyanarayanan, et al, who introduced the concept of *Cloudlets* [41]. These are resource-rich servers, relative to edge devices, that could host VMs, while also being closer in the network topology to the edge devices, relative to cloud data centers. They are designed to overcome the constrained resources available on edge platforms while reducing the network latency expended in being tethered to the cloud for interactive applications. *Fog computing* generalizes (and popularizes) the notion of cloudlets.

The term "Fog Computing" was coined by Cisco, and first appears publicly in a talk by Flavio Bonomi, Vice President and Fellow at Cisco Systems, as part of the *Network-Optimized Computing at the Edge Of the Network Workshop*, co-located with International Symposium on Computer Architecture (ISCA), in 2011 [9]. This was further described as extending the concept of cloud computing to the edge of the network to support low-latency and geo-distributed applications for mobile and IoT domains [10]. Since then, fog computing has been evolving as a concept and covers a wide class of resources that sit between the edge devices and cloud data centers on the network topology, have capacities that fall between edge devices and commodity clusters on clouds, and may be managed *ad hoc* as a smart gateway or professionally as a computing infrastructure [51, 50, 53].

4 Motivations

Fog computing is relevant in the context of *wide-area distributed systems*, with many clients at the edge of the Internet [17]. Such clients may be mobile or consumer devices (*e.g.,* smart phone, smart watch, Virtual Reality (VR) headset) used interactively by humans, or devices that are part of IoT (*e.g.,* smart meter, traffic cameras and signaling, driverless cars) for machine-to-machine (M2M) interactions. The client may serve both as *consumers* of data or actuators that receive control signals (*e.g.,* fitness notification on smart watch, signal change operation on traffic light), as well as *producers* of data (*e.g.,* heart-rate observations from smart watch, video streams from traffic cameras) [16].

2

There are two contemporary models of computing for applications and analytics that use data from, or generate content and controls for, such clients at the edge of the Internet [46], as illustrated in Fig. 1. In a *cloud-centric model*, also called cloud computing, data from these clients is sent to a central data-center where the processing is done and the responses, if any, are sent back to the same or different client(s) [47, 13]. In an *edge-centric model*, or edge-computing, part (or even all) of the processing is done at the data source with the rest done on the cloud [7, 3].

These have their relative advantages. Cloud computing outsources computing infrastructure management to providers, who offer elastic access to seemingly infinite compute resources on-demand which can be rented by the minute. They are also cheaper due to economies of scale at centralized locations [13]. Edge computing leverages the compute capacity of existing captive devices and reduces network transfers, both of which lower the costs. There may also be enhanced trust and context available closer to the edge [19].

While fog computing is still maturing, there are many reason why its rise is inevitable due to the gaps in these two common computing approaches. The *network latency* from the edge client to the cloud data-center is high and variable, averaging between $20 - 250\ ms$ depending on the location of the client and data center [24]. The *network bandwidth* between the edge and the cloud, similarly, averages at about $800 - 1200\ KB/s$. Both these mean that latency sensitive or bandwidth intensive applications will offer poor performance using a cloud-centric model due to the round-trip time between edge and cloud [41, 42]. Another factor is the connectivity of devices to the Internet. Mobile devices may be out of network coverage periodically, and cause cloud-centric applications to degrade or loose functionality [44].

Edge computing, while avoiding issues of network time, suffer from operating on constrained devices that have limited battery, compute and memory capacities [6]. This reduces application performance and limits sustained processing that can drain the battery. These devices may also be less robust and their network connectivity less available [4].

Fog computing serves as a computing layer that sits between the edge devices and the cloud in the network topology. They have more compute capacity than the edge but much less so than cloud data centers. They typically have high uptime and always-on Internet connectivity. Applications that make use of the fog can avoid the network performance limitation of cloud computing while being less resource constrained than edge computing. As a result, they offer a useful balance of the current paradigms.

5   Characteristics

Fog computing is an evolving model of computing. Hence, its perceived characteristics are broad, based on diverse view-points of the research community and industry [51]. It is generally agreed that a defining characteristic of fog computing is its smaller network distance from the edge, but multiple **network**
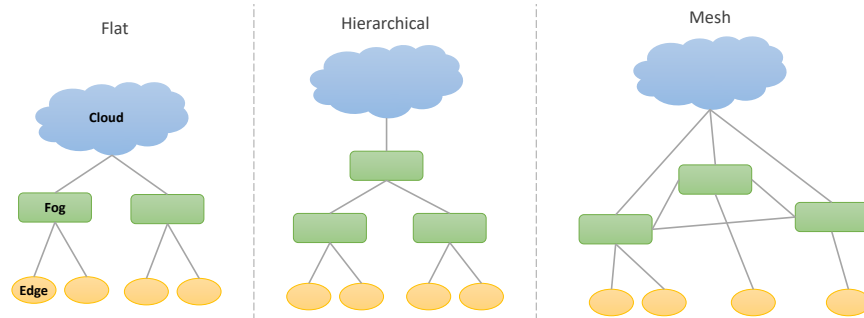
Figure 2: Different interaction models in fog computing.

topology architectures exist, as shown in Fig. 2 [1, 23, 26]. Some models consider there to be a *flat* fog layer that sits, say, at 1-hop from the edge devices and allows connectivity to the cloud. Each fog device may be responsible for a collection of edge devices that connect to it. This zone of responsibility may be based on logical grouping or spatial regions. Others consider a *hierarchical* approach where fog devices form layers, where each layer is further away from the edge device and responsible for fog/edge devices that fall within its sub-tree, with the cloud forming the root. Alternatively, there are *mesh* designs where fog resources can communicate with each other as well as the cloud, with each edge device assigned to a single fog. A variation of this does not distinguish between edge and fog devices, and any of these may communicate with each other, or the cloud. This approaches an edge-centric or P2P model.

The **resource capacity** in the fog can vary. At one end, Raspberry Pi devices with 1 GHz processors, 1 GB RAM and 100 $Mbps$ Ethernet may serve as a gateway fog resource for lower-end edge devices. On the other hand, fog resources could be provisioned as "micro" or "nano" data-centers with clusters of servers or even accelerators present [19]. Individual fog devices can also have heterogeneous capacities [15].

Fog resources tend to be more **reliable and available** than edge resources, though lacking the robust fail-over mechanisms in the cloud that is possible due to a large resource pool. This makes them well-suited to serve as a layer for persisting data and services for the short and medium term. Fog resources themselves may be deployed in a stationary environment (*e.g.,* coffee shop, airport, cell tower) or in a mobile platform (*e.g.,* train, cab) [15]. This can affect the network connectivity of the fog with the cloud, in case it uses cellular networks for Internet access, and even its energy footprint [28].

Fog **deployment models** are still emerging. These resources may be deployed within a public or a private network, depending on its end-use. Smart city deployments may make them available to utility services within the city network [52] while retail shops and transit services can make them available to their customers. A wider deployment for public use on-demand, say, by cellular providers, cities or even cloud providers, will make it comparable to cloud

computing in terms of accessibility. These have implications on the operational costs as well [50].

The fog resources may be made available **as-a-service**, similar to cloud resources. These may be virtualized or non-virtualized infrastructure [8], with containers offering a useful alternative to hypervisor-based VMs that may be too heavy-weight for lower-end fog resources [3]. However, there is a still a lack of a common platform, and programming models are just emerging [25, 38]. Most applications that use the fog tend to be custom designed, and there has only been some theoretical work on scheduling applications on edge, fog and cloud [11, 20].

6   Role of Big Data

One of the key rationales for deploying and using fog computing is Big Data generated at the edge of the network. This is accelerated by IoT deployments. Traditional web clients which just consume services and content from the WWW saw the growth of *Content Distribution Networks (CDN)* to serve these with low latency. The data from IoT sensors is instead generating data at the clients that is being pushed to the cloud [13]. In this context, fog computing has been described as acting like an inverse CDN [42].

A large swathe of IoT data comes as *observational streams*, or time-series data, from widely distributed sensors [34, 45]. These data streams vary in their rates – once every 15 *mins* for smart utility meters, every second for heart rate monitoring by a fitness watch to 50 Hz by Phasor Measurement Units (PMU) in smart power grids, and the number of sensors can range in the millions for city-scale deployments. These are **high velocity** data streams that are latency sensitive and need online analytics and decision-making to provide, say, health alerts or manage the power grid behavior [47]. Here, fog computing can help move the decision-making close to the edge to reduce latency.

Another class of **high volume** data that is emerging is from video streams from traffic and surveillance cameras, for public safety, intelligent traffic management and even driverless cars [2]. Here, the bandwidth consumed in moving the data from the edge to the cloud can be enormous as high-definition cameras become cheap but network capacity growth does not keep pace [42]. The applications that need to be supported can span real-time video analytics to just recording footage for future use. Fog computing can reduce the bandwidth consumed in the core Internet and limit data movement to the local network. In addition, it can offer higher compute resources and accelerators to deploy complex analytics as well.

In addition, **telemetry data** from monitoring the health of the IoT fabric itself may form a large corpus [52]. Related to this is provenance that describes the source and processing done on the distributed devices that may be essential to determine the quality and veracity of the data. Fog can help with the collection and curation of such ancillary data streams as well.

Data archival is another key requirement within such applications [13, 50].

Besides online analytics, the raw or pre-processed observational data may need to be persisted for medium or long-term to train models for Machine Learning (ML) algorithms, for auditing to justify automated decisions, or to analyze on-demand based on external factors. Depending on the duration of persistence, data may be buffered in the fog either transiently or for movement to the cloud during off-peak hours to shape bandwidth usage. Data can also be filtered or aggregated to send only the necessary subset to the cloud.

Lastly, **metadata** describing the entities in the eco-system will be essential for information integration from diverse domains [3]. These can be static or slow changing data, or even complex knowledge or semantic graphs that are constructed. They may need to be combined with real-time data to support decision making [55]. The fog layer can play a role in replicating and maintaining this across distributed resources closer to the edge.

There has been rapid progress on Big Data Platforms on clouds and clusters, with frameworks like Spark, Storm, Flink, HBase, Pregel and TensorFlow helping store and process large data volumes, velocities, and semi-structured data. Clouds also offer these platforms as a service. However, there is a lack of programming models, platforms and middleware to support various processing patterns necessary over Big Data at the edge of the network that can effectively leverage edge, fog and cloud resources [36].

7 Examples of Applications

The applications driving the deployment and need for fog computing are diverse. But some requirements are recurrent: low latency processing, high volume data, high computing or storage needs, privacy and security, and robustness. These span virtual and augmented reality (VR/AR) applications and gaming [53], Industrial IoT [15] and field support for the military [30]. Some of the emerging and high impact applications are highlighted below.

7.1 Smart Cities

Smart cities are a key driver for fog computing, and these are already being deployed. The Barcelona city's "street-side cabinets" offer fog resources as part of the city infrastructure [52]. Here, audio and environment sensors, video cameras, and power utility monitors are packaged along-side compute resources and network back-haul capacity as part of fog cabinets placed along streets. These help aggregate data from sensors, perform basic analytics and also offer WiFi hot-spots for public use. As an example, audio analytics at the fog helps identify loud noises that then triggers a surveillance camera to capture a segment of video for further analysis. Similar efforts are underway at other cities as well [5]. These go toward supporting diverse smart city applications for power and water utilities, intelligent transport, public safety, etc., both by the city and by app-developers.

One of the major drivers for such city-scale fog infrastructure is likely to be *video surveillance* that is starting to become pervasive in urban spaces [42]. Such city or even crowd-sourced video feeds form *meta-sensors* when combined with recent advances in deep neural networks (DNN). *E.g.,* feature extraction and classification can count traffic and crowds, detect pollution levels, identify anomalies, etc. As such, they can replace myriad other environment sensors when supported by real-time analytics. Such DNN and ML algorithms are computationally cost to train and even infer, and can make use of fog resources with accelerators [29].

7.2 Healthcare

Wearables are playing a big role in not just personal fitness but also as assistive technologies in healthcare. Projects have investigated the use of such on-person monitoring devices to detect when stroke patients have fallen and need external help [14]. Others use eye-glass cameras and head-up displays (HUDs) to offer verbal and cognitive cues to Alzheimer's patients suffering from memory loss, based on visual analytics [41]. Predictive analytics over brain signals monitored from EEG headsets have been used for real-time mental state monitoring [39]. These are then used to mitigate external conditions and stimuli that can affect patients' mental state.

All these applications require low latency and reliable analytics to be performed over observations that are being collected by wearables. Given that such devices need to be light-weight, the computation is often out-sourced to a smart phone or a server that acts as a fog resource, and to which the observational data is passed for computation. There are also decisions to be made with respect to what to compute on the wearable and what to communicate to the fog, so as to balance the energy usage of the device.

7.3 Mobility

Driverless cars and drones are emerging application domains where fog computing plays a key role [15]. Both these platforms have many on-board sensors and real-time analytics for autonomous mobility. Vehicular networks allow connected cars to communicate with each other (V2V) to cooperatively share information to make decisions on traffic and road conditions [26]. These can be extended to share compute capacity to perform these analytics. This allows a collection of parked or slow-moving vehicles to form a fog of resources among proximate ones. These may complement occasional road-side units that offer connectivity with the cloud. Here, the entities forming the fog can change dynamically, and requires distributed resource coordination.

Drones are finding use in asset monitoring, such as inspecting gas pipelines and power transmission towers at remote locations [31]. Here, a mobile base-station has a digital control tether with a collection of drones that follow a pre-set path to collect observational data. The drones have limited compute capacity to increase their endurance, and need to prioritize its use for autonomous

navigation. So they typically serve as "data mules", collecting data from sensors along the route but doing limited on-board processing [33]. However, when situations of interest arise, they can make use of their local capacity, resources available with near-by drones, or with fog resources in the base-station, while the trip is ongoing. This can help decide if a additional fly-past is necessary.

8   Research Directions

Fog computing is still exploratory in nature. While it has gained recent attention from researchers, many more topics need to be examined further.

8.1   Fog Architectures

Many of the proposed fog architectural designs have not seen large scale deployments, and are just plausible proposals. While city-scale deployments with 100's of fog devices are coming online, the experiences from their operations will inform future design [52]. Network management is likely to be a key technical challenge as traffic management within the metropolitan area network (MAN) gains importance [50]. Unlike static fog resources, mobile or *ad hoc* resources such as vehicular fog will pose challenges of resource discovery, access and coordination [26, 23]. Resource churn will need to be handled through intelligent scheduling [19]. Resources will also require robust adaptation mechanisms based on the situational context [37]. Open standards will be required to ensure interoperability. To this end, there are initial efforts on defining reference models for fog computing [12].

8.2   Data Management

Tracking and managing content across edge, fog and cloud will be a key challenge. Part of this is the result of devices in IoT acting as data sources and compute platforms, which necessitates coordination across the cyber and physical worlds [3]. The generation of event streams that are transient and need to be processed in a timely manner poses additional challenges to the velocity dimension of big data [34]. Data discovery, replication, placement and persistence will need careful examination in the context of wide area networks and transient computing resources. Sensing will need to be complemented with "sense-making" so that data is interpreted correctly by integrating multiple sources [37].

8.3   Programming Models and Platforms

Despite the growth of edge and fog resources, there is a lack of a common programming abstraction or runtime environments for defining and executing distributed applications on these resources [49]. There has been some preliminary work in defining a hierarchical pattern for composing applications that generate data from the edge and need to incrementally aggregate and process

them at the fog and cloud layers [25]. They use spatial partitioning to assign edge devices to fogs. The ECHO platform offers a dataflow model to compose applications that are then scheduled on distributed runtime engines that are present on edge, fog and cloud resources [38]. It supports diverse execution engines such as NiFi, Storm and TensorFlow, but the user couples the tasks to a specific engine. A declarative application specification and big data platform is necessary to ease the composition of applications in such complex environments.

Related to this are application deployment and resource scheduling. VMs are used in the cloud to configure the required environment, but they may prove to be too resource intensive for fog. Some have examined the use of just a subset of the VM's footprint on the fog, and migrating this image across resources to track the mobility of user(s) who access its services [8]. Resource scheduling on edge, fog and cloud have also been explored, though often validated just through simulations due to the lack of access to large scale fog setups [22, 54]. Spatial awareness and energy awareness are distinctive features that have been included in such schedulers [11, 20]. Formal modeling of the fog has been undertaken as well [40]. Quality of Experience (QoE) as a user-centric alternative metric to Quality of Service (QoS) is also being examined [2].

Such research will need to be revisited as the architectural and application models for fog computing become clearer, with mobility, availability and energy usage of resources offering unique challenges [43].

8.4   Security, Privacy, Trust

Unlike cloud computing where there is a degree of trust in the service provider, fog computing may contain resources from diverse and *ad hoc* providers. Further, fog devices may not be physically secured like a data center, and may be accessible by third-parties [15]. Containerization does not offer the same degree of sand-boxing between multiple tenant applications that virtualization does. Hence, data and applications in the fog operate within a mix of trusted and untrusted zones [19]. This requires constant supervision of the device, fabric, applications and data by multiple stakeholders to ensure that security and privacy are not compromised. Techniques like anomaly detection, intrusion detection, moving target defense, etc. will need to be employed. Credential and identity management will be required. Provenance and auditing mechanisms will prove essential as well. These will need to be considered as first-class features when designing the fog deployment or the application.

References

[1] Mohammad Aazam and Eui-Nam Huh. Fog computing and smart gateway based communication for cloud of things. In *International Conference on Future Internet of Things and Cloud (FiCloud)*, 2014.

[2] Mohammad Aazam, Marc St-Hilaire, Chung-Horng Lung, and Ioannis Lambadaris. Mefore: Qoe based resource estimation at fog to enhance qos in iot. In *International Conference on Telecommunications (ICT)*, 2016.

[3] Narendra Anand, Anuraag Chintalapally, Colin Puri, and Teresa Tung. Practical edge analytics: Architectural approach and use cases. In *IEEE International Conference on Edge Computing (EDGE)*, 2017.

[4] Atakan Aral and Ivona Brandic. Quality of service channelling for latency sensitive edge applications. In *IEEE International Conference on Edge Computing (EDGE)*, 2017.

[5] B. Amrutur, et al. An Open Smart City IoT Test Bed: Street Light Poles as Smart City Spines. In *ACM/IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI)*, 2017.

[6] Marco V Barbera, Sokol Kosta, Alessandro Mei, and Julinda Stefa. To offload or not to offload? the bandwidth and energy costs of mobile cloud computing. In *IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[7] Michael Till Beck, Martin Werner, Sebastian Feld, and Thomas Schimper. Mobile edge computing: A taxonomy. In *International Conference on Advances in Future Internet*, 2014.

[8] L. F. Bittencourt, M. M. Lopes, I. Petri, and O. F. Rana. Towards virtual machine migration in fog computing. In *IEEE International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 2015.

[9] Flavio Bonomi. Cloud and fog computing: Trade-offs and applications. In *Workshop on Network-Optimized Computing at the Edge Of the Network (EON)*, 2011.

[10] Flavio Bonomi, Rodolfo Milito, Preethi Natarajan, and Jiang Zhu. *Big Data and Internet of Things: A Roadmap for Smart Environments*, chapter Fog Computing: A Platform for Internet of Things and Analytics, pages 169–186. Springer, 2014.

[11] Antonio Brogi and Stefano Forti. Qos-aware deployment of iot applications through the fog. *IEEE Internet of Things Journal*, 4(5), 2017.

[12] Charles Byers and Robert Swanson. Openfog reference architecture for fog computing. Technical report, Open Fog Consortium, 2017.

[13] Hongming Cai, Boyi Xu, Lihong Jiang, and Athanasios V Vasilakos. Iot-based big data storage systems in cloud computing: Perspectives and challenges. *IEEE Internet of Things Journal*, 4(1), 2017.

[14] Yu Cao, Songqing Chen, Peng Hou, and Donald Brown. Fast: A fog computing assisted distributed analytics system to monitor fall for stroke mitigation. In *IEEE International Conference on Networking, Architecture and Storage (NAS)*, 2015.

[15] Mung Chiang and Tao Zhang. Fog and iot: An overview of research opportunities. *IEEE Internet of Things Journal*, 3(6), 2016.

[16] Amir Vahid Dastjerdi and Rajkumar Buyya. Fog computing: Helping the internet of things realize its potential. *IEEE Computer*, 49(8), 2016.

[17] Benoit Donnet and Timur Friedman. Internet topology discovery: a survey. *IEEE Communications Surveys & Tutorials*, 9(4):56–69, 2007.

[18] Niroshinie Fernando, Seng W Loke, and Wenny Rahayu. Mobile cloud computing: A survey. *Future Generation Computer Systems (FGCS)*, 29(1):84–106, 2013.

[19] Pedro Garcia Lopez, Alberto Montresor, Dick Epema, Anwitaman Datta, Teruo Higashino, Adriana Iamnitchi, Marinho Barcellos, Pascal Felber, and Etienne Riviere. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review*, 45(5):37–42, 2015.

[20] Rajrup Ghosh and Yogesh Simmhan. Distributed scheduling of event analytics across edge and cloud. *ACM Transactions on Cyber Physical Systems (TCPS)*, 2017.

[21] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems (FGCS)*, 29(7):1645–1660, 2013.

[22] Harshit Gupta, Amir Vahid Dastjerdi, Soumya K Ghosh, and Rajkumar Buyya. ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments. *Software: Practice and Experience (SPE)*, 47(9):12751296, 2017.

[23] Jianhua He, Jian Wei, Kai Chen, Zuoyin Tang, Yi Zhou, and Yan Zhang. Multi-tier fog computing with large-scale iot data analytics for smart cities. *IEEE Internet of Things Journal*, 2017.

[24] Keqiang He, Alexis Fisher, Liang Wang, Aaron Gember, Aditya Akella, and Thomas Ristenpart. Next stop, the cloud: Understanding modern web service deployment in ec2 and azure. In *Conference on Internet Measurement*, 2013.

[25] Kirak Hong, David Lillethun, Umakishore Ramachandran, Beate Ottenwälder, and Boris Koldehofe. Mobile fog: A programming model for large-scale applications on the internet of things. In *ACM SIGCOMM Workshop on Mobile Cloud Computing*, 2013.

[26] Xueshi Hou, Yong Li, Min Chen, Di Wu, Depeng Jin, and Sheng Chen. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology*, 65(6):3860–3873, 2016.

[27] Kai Hwang, Jack Dongarra, and Geoffrey C Fox. *Distributed and cloud computing: from parallel processing to the internet of things*. Morgan Kaufmann, 2013.

[28] Fatemeh Jalali, Safieh Khodadustan, Chrispin Gray, Kerry Hinton, and Frank Suits. Greening iot with fog: A survey. In *IEEE International Conference on Edge Computing (EDGE)*, 2017.

[29] Aakash Khochare, Pushkara Ravindra, Siva Prakash Reddy, and Yogesh Simmhan. Distributed video analytics across edge and cloud using echo. In *International Conference on Service-Oriented Computing (ICSOC) Demo*, 2017.

[30] Grace Lewis, Sebastián Echeverría, Soumya Simanta, Ben Bradshaw, and James Root. Tactical cloudlets: Moving cloud computing to the edge. In *IEEE Military Communications Conference (MILCOM)*, 2014.

[31] Seng W. Loke. The internet of flying-things: Opportunities and challenges with airborne fog computing and mobile cloud in the clouds. Technical Report 1507.04492v1, arXiv, 2015.

[32] Dejan S Milojicic, Vana Kalogeraki, Rajan Lukose, Kiran Nagaraja, Jim Pruyne, Bruno Richard, Sami Rollins, and Zhichen Xu. Peer-to-peer computing. Technical Report HPL-2002-57, HP Labs, 2002.

[33] Prasant Misra, Yogesh Simmhan, and Jay Warrior. Towards a practical architecture for internet of things: An india-centric view. *IEEE Internet of Things (IoT) Newsletter*, January 2015.

[34] Mohammed Islam Naas, Philippe Raipin Parvedy, Jalil Boukhobza, and Laurent Lemarchand. ifogstor: an iot data placement strategy for fog infrastructure. In *IEEE International Conference on Fog and Edge Computing (ICFEC)*, 2017.

[35] David L Nelson and C Gordon Bell. The evolution of workstations. *IEEE Circuits and Devices magazine*, 2(4):12–16, 1986.

[36] Subhav Pradhan, Abhishek Dubey, Shweta Khare, Saideep Nannapaneni, Aniruddha Gokhale, Sankaran Mahadevan, Douglas C. Schmidt, and Martin Lehofer. Chariot: Goal-driven orchestration middleware for resilient iot systems. *ACM Transactions on Cyber-Physical Systems (TCPS)*, 2017.

[37] J. S. Preden, K. Tammeme, A. Jantsch, M. Leier, A. Riid, and E. Calis. The benefits of self-awareness and attention in fog and mist computing. *IEEE Computer*, 48(7):37–45, 2015.

[38] Pushkara Ravindra, Aakash Khochare, Siva Prakash Reddy, Sarthak Sharma, Prateeksha Varshney, and Yogesh Simmhan. Echo: An adaptive orchestration platform for hybrid dataflows across cloud and edge. In *International Conference on Service-Oriented Computing (ICSOC)*, 2017.

[39] Koosha Sadeghi, Ayan Banerjee, Javad Sohankar, and Sandeep K.S. Gupta. Optimization of brain mobile interface applications using iot. In *IEEE International Conference on High Performance Computing (HiPC)*, 2016.

[40] S. Sarkar and S. Misra. Theoretical modelling of fog computing: a green computing paradigm to support iot applications. *IET Networks*, 5(2), 2016.

[41] Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, and Nigel Davies. The case for vm-based cloudlets in mobile computing. *IEEE Pervasive Computing*, 8(4):1536–1268, 2009.

[42] Mahadev Satyanarayanan, Pieter Simoens, Yu Xiao, Padmanabhan Pillai, Zhuo Chen, Kiryong Ha, Wenlu Hu, and Brandon Amos. Edge analytics in the internet of things. *IEEE Pervasive Computing*, 14(2):1536–1268, 2015.

[43] Cong Shi, Vasileios Lakafosis, Mostafa H Ammar, and Ellen W Zegura. Serendipity: enabling remote computing among intermittently connected mobile devices. In *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2012.

[44] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5):2327–4662, 2016.

[45] A. Shukla, S. Chaturvedi, and Y. Simmhan. Riotbench: An iot benchmark for distributed stream processing systems. *Concurrency and Computation: Practice and Experience*, 2017.

[46] Yogesh Simmhan. Iot analytics across edge and cloud platforms. *IEEE IoT Newsletter*, 2017.

[47] Yogesh Simmhan, Saima Aman, Alok Kumbhare, Rongyang Liu, Sam Stevens, Qunzhi Zhou, and Viktor Prasanna. Cloud-based software platform for big data analytics in smart grids. *Computing in Science & Engineering (CiSE)*, 15(4):38–47, 2013.

[48] Alok Sinha. Client-server computing. *Communications of the ACM*, 35(7):77–98, 1992.

[49] Ion Stoica, Dawn Song, Raluca Ada Popa, David A Patterson, Michael W Mahoney, Randy H Katz, Anthony D Joseph, Michael Jordan, Joseph M Hellerstein, Joseph Gonzalez, et al. A berkeley view of systems challenges for ai. Technical report, University of California, Berkeley, 2017.

[50] Luis M Vaquero and Luis Rodero-Merino. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review*, 44(5):27–32, 2014.

[51] Prateeksha Varshney and Yogesh Simmhan. Demystifying fog computing: Characterizing architectures, applications and abstractions. In *IEEE International Conference on Fog and Edge Computing (ICFEC)*, 2017.

[52] Marcelo Yannuzzi, Frank van Lingen, Anuj Jain, Oriol Lluch Parellada, Manel Mendoza Flores, David Carrera, Juan Luis Perez, Diego Montero, Pablo Chacin, Angelo Corsaro, and Albert Olive. A new era for cities with fog computing. *IEEE Internet Computing*, 21(2):54–67, 2017.

[53] Shanhe Yi, Cheng Li, and Qun Li. A survey of fog computing: concepts, applications and issues. In *Workshop on Mobile Big Data*, 2015.

[54] Xuezhi Zeng, Saurabh Kumar Garg, Peter Strazdins, Prem Prakash Jayaraman, Dimitrios Georgakopoulos, and Rajiv Ranjan. Iotsim: A simulator for analysing iot applications. *Journal of Systems Architecture*, 72:93–107, January 2017.

[55] Qunzhi Zhou, Yogesh Simmhan, and Viktor K. Prasanna. Knowledge-infused and consistent complex event processing over real-time and persistent streams. *Future Generation Computer Systems (FGCS)*, 76:391–406, November 2017.