Fog Computing: Helping the Internet of Things Realize its Potential

Amir Vahid Dastjerdi and Rajkumar Buyya, University of Melbourne

The Internet of Things (IoT) could enable innovations that enhance the quality of life, but it generates unprecedented amounts of data that are difficult for traditional systems, the cloud, and even edge computing to handle. Fog computing is designed to overcome these limitations.

The Internet of Things (IoT) promises to make many items—including consumer electronic devices, home appliances, medical devices, cameras, and all types of sensors—part of the Internet environment.[1] This opens the door to innovations that facilitate new interactions among things and humans, and enables the realization of smart cities, infrastructures, and services that enhance the quality of life.

By 2025, researchers estimate that the IoT could have an economic impact—including, for example, revenue generated and operational savings—of $11 trillion per year, which would represent about 11 percent of the world economy;[2] and that users will deploy 1 trillion IoT devices.

COPING WITH INTERNET OF THINGS DATA

IoT environments generate unprecedented amounts of data that can be useful in many ways, particularly if analyzed for insights. However, the data volume can overwhelm today's storage systems and analytics applications.

Cloud computing could help by offering on-demand and scalable storage, as well as processing services that can scale to IoT requirements. However, for health-monitoring, emergency-response, and other latency-sensitive applications, the delay caused by transferring data to the cloud and back to the application is unacceptable. In addition, it isn't efficient to send so much data to the cloud for storage and processing, as it would saturate network bandwidth and not be scalable.

Recent analysis of a healthcare-related IoT application with 30 million users showed data flows up to 25,000 tuples per second.[3] And real-time data flows in smart cities with

many more data sources could easily reach millions of tuples per second.

To address these issues, edge computing[4] was proposed to use computing resources near IoT sensors for local storage and preliminary data processing. This would decrease network congestion, as well as accelerate analysis and the resulting decision making. However, edge devices can't handle multiple IoT applications competing for their limited resources, which results in resource contention and increases processing latency.

Fog computing—which seamlessly integrates edge devices and cloud resources—helps overcome these limitations. It avoids resource contention at the edge by leveraging cloud resources and coordinating the use of geographically distributed edge devices.

FOG COMPUTING CHARACTERISTICS

Fog computing is a distributed paradigm that provides cloud-like services to the network edge. It leverages cloud and edge resources along with its own infrastructure, as Figure 1 shows. In essence, the technology deals with IoT data locally by utilizing clients or edge devices near users to carry out a substantial amount of storage, communication, control, configuration, and management. The approach benefits from edge devices' close proximity to sensors, while leveraging the on-demand scalability of cloud resources.

Fog computing involves the components of data-processing or analytics applications running in distributed cloud and edge devices. It also facilitates the management and programming of computing, networking, and storage services between datacenters and end devices. In addition, it supports user mobility, resource and interface heterogeneity, and distributed data analytics to address the
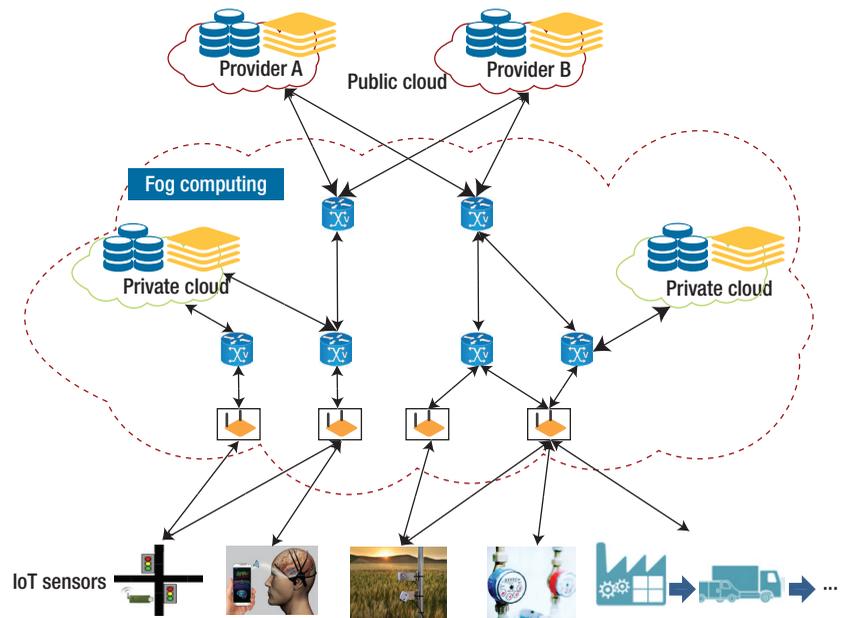
Figure 1. Distributed data processing in a fog-computing environment. Based on the desired functionality of a system, users can deploy Internet of Things sensors in different environments including roads, medical centers, and farms. Once the system collects information from the sensors, fog devices—including nearby gateways and private clouds—dynamically conduct data analytics.

requirements of widely distributed applications that need low latency.

FOG-COMPUTING COMPONENTS

Figure 2 presents a fog-computing reference architecture. Fog systems generally use the sense-process-actuate and stream-processing programming models. Sensors stream data to IoT networks, applications running on fog devices subscribe to and process the information, and the obtained insights are translated into actions sent to actuators.

Fog systems dynamically discover and use APIs to build complex functionalities. Components at the resource-management layer use information from the resource-monitoring service to track the state

of available cloud, fog, and network resources and identify the best candidates to process incoming tasks. With multitenant applications, the resource-management components prioritize the tasks of the various participating users or programs.

Edge and cloud resources communicate using machine-to-machine (M2M) standards such as MQTT (formerly MQ Telemetry Transport) and the Constrained Application Protocol (CoAP). Software-defined networking (SDN) helps with the efficient management of heterogeneous fog networks.

FOG-COMPUTING SOFTWARE SYSTEMS

There are four prominent software systems for building fog computing environments and applications.
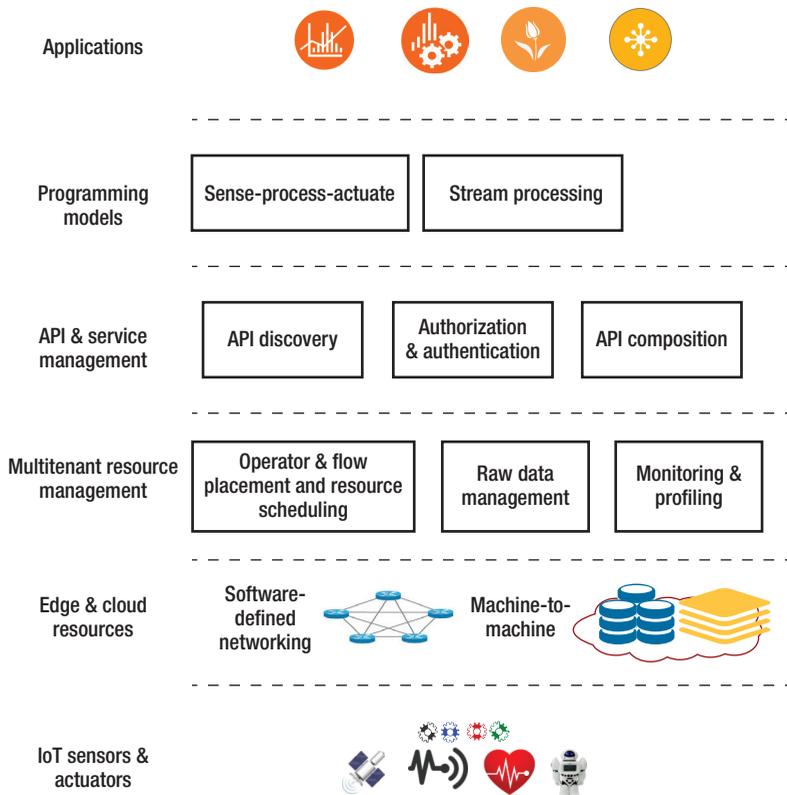
Figure 2. Fog-computing architecture. In the bottom layer are end devices—including sensors and actuators—along with applications that enhance their functionality. These elements use the next layer, the network, for communicating with edge devices, such as gateways, and then with cloud services. The resource-management layer runs the entire infrastructure and enables quality-of-service enforcement. Finally, applications leverage fog-computing programming models to deliver intelligent services to users.

Cisco IOx provides device management and enables M2M services in fog environments.[5] Using device abstractions provided by Cisco IOx APIs, applications running on fog devices can communicate with other IoT devices via M2M protocols.

Cisco Data in Motion (DMo) enables data management and analysis at the network edge and is built into products that Cisco Systems and its partners provide.

LocalGrid's fog-computing platform is software installed on network devices in smart grids. It provides reliable M2M communication between devices and data-processing services without going through the cloud.

Cisco ParStream's fog-computing platform enables real-time IoT analytics.

FOG-COMPUTING APPLICATIONS

Various applications could benefit from fog computing.

Healthcare and activity tracking

Fog computing could be useful in healthcare, in which real-time processing and event response are critical. One proposed system utilizes fog computing to detect, predict, and prevent falls by stroke patients.[6] The fall-detection learning algorithms are dynamically deployed across edge devices and cloud resources. Experiments concluded that this system had a lower response time and consumed less energy than cloud-only approaches.

A proposed fog computing–based smart-healthcare system enables low latency, mobility support, and location and privacy awareness.[7]

Smart utility services

Fog computing can be used with smart utility services,[8] whose focus is improving energy generation, delivery, and billing. In such environments, edge devices can report more fine-grained energy-consumption details (for example, hourly and daily, rather than monthly, readings) to users' mobile devices than traditional smart utility services. These edge devices can also calculate the cost of power consumption throughout the day and suggest which energy source is most economical at any given time or when home appliances should be turned on to minimize utility use.

Augmented reality, cognitive systems, and gaming

Fog computing plays a major role in augmented-reality applications, which are latency sensitive. For example, the EEG Tractor Beam augmented multiplayer, online brain–computer-interaction game performs continuous real-time brain-state classification on fog devices and then tunes classification models on cloud servers, based on electroencephalogram readings that sensors collect.[9]

A wearable cognitive-assistance system that uses Google Glass devices helps people with reduced mental acuity perform various tasks, including telling them the names of people they meet but don't remember.[10] In this application, devices communicate with the cloud for delay-tolerant jobs such as error reporting and logging. For time-sensitive tasks, the system streams video from the Glass camera to the fog devices for processing. The system demonstrates how using nearby fog devices greatly decreases end-to-end latency.

MODELING AND SIMULATION

To enable real-time analytics in fog computing, we must investigate various resource-management and scheduling

techniques including the placement, migration, and consolidation of stream-processing operators, application modules, and tasks. This significantly impacts processing latency and decision-making times.

However, constructing a real IoT environment as a testbed for evaluating such techniques is costly and doesn't provide a controllable environment for conducting repeatable experiments. To overcome this limitation, we developed an open source simulator called iFogSim.[11] iFogSim enables the modeling and simulation of fog-computing environments for the evaluation of resource-management and scheduling policies across edge and cloud resources under multiple scenarios, based on their impact on latency, energy consumption, network congestion, and operational costs. It measures performance metrics and simulates edge devices, cloud datacenters, sensors, network links, data streams, and stream-processing applications.

CHALLENGES

Realizing fog computing's full potential presents several challenges including balancing load distribution between edge and cloud resources, API and service management and sharing, and SDN communications. There are several other important examples.

Enabling real-time analytics

In fog environments, resource management systems should be able to dynamically determine which analytics tasks are being pushed to which cloud- or edge-based resource to minimize latency and maximize throughput. These systems also must consider other criteria such as various countries' data privacy laws involving, for example, medical and financial information.

Programming models and architectures

Most stream- and data-processing frameworks, including Apache Storm and S4, don't provide enough scalability and flexibility for fog and IoT environments because their architecture is based on static configurations. Fog environments require the ability to add and remove resources dynamically because processing nodes are generally mobile devices that frequently join and leave networks.

Security, reliability, and fault tolerance

Enforcing security in fog environments—which have multiple service providers and users, as well as distributed resources—is a key challenge. Designing and implementing authentication and authorization techniques that can work with multiple fog nodes that have different computing capacities is difficult. Public-key infrastructures and trusted execution environments are potential solutions.[12]

Users of fog deployments also must plan for the failure of individual sensors, networks, service platforms, and applications. To help with this, they could apply standards, such as the Stream Control Transmission Protocol, that deal with packet and event reliability in wireless sensor networks.[13]

Power consumption

Fog environments consist of many nodes. Thus, the computation is distributed and can be less energy efficient than in centralized cloud systems. Using efficient communications protocols such as CoAP, effective filtering and sampling techniques, and joint computing and network resource optimization can minimize energy consumption in fog environments.

Fog computing enables the seamless integration of edge and cloud resources. It supports the decentralized and intelligent processing of unprecedented data volumes generated by IoT sensors deployed for smooth integration of physical and cyber environments.

This could generate many benefits to society by, for example, enabling smart healthcare applications. The further development of fog computing could thus help the IoT reach its vast potential. ◼

REFERENCES

1. R. Buyya and A. Dastjerdi, *Internet of Things: Principles and Paradigms*, Morgan Kaufmann, 2016.
2. J. Manyika et al., *Unlocking the Potential of the Internet of Things*, McKinsey & Company, 2015.
3. R. Cortés et al., "Stream Processing of Healthcare Sensor Data: Studying User Traces to Identify Challenges from a Big Data Perspective," *Procedia Computer Science*, vol. 52, 2015, pp. 1004–1009.
4. W. Shi and S. Dustdar, "The Promise of Edge Computing," *Computer*, vol. 49, no. 5, 2016, pp. 78–81.
5. F. Bonomi et al., "Fog Computing: A Platform for Internet of Things and Analytics," *Big Data and Internet of Things: A Roadmap for Smart Environments*, N. Bessis and C. Dobre, eds., Springer, 2014, pp. 169–186.
6. Y. Cao et al., "FAST: A Fog Computing Assisted Distributed Analytics System to Monitor Fall for Stroke Mitigation," *Proc. 10th IEEE Int'l Conf. Networking, Architecture and Storage* (NAS 15), 2015, pp. 2–11.
7. V. Stantchev et al., "Smart Items, Fog and Cloud Computing as Enablers of Servitization in Healthcare," *J. Sensors & Transducers*, vol. 185, no. 2, 2015, pp. 121–128.
8. J. Li et al., "EHOPES: Data-Centered Fog Platform for Smart Living," *Proc. 2015 Int'l Telecommunication Networks and Applications Conf.* (ITNAC 15), 2015, pp. 308–313.
9. J. Zao et al., "Augmented Brain Computer Interaction Based on Fog Computing and Linked Data," *Proc. 10th IEEE Int'l Conf. Intelligent Environments* (IE 14), 2014, pp. 374–377.
10. K. Ha et al., "Towards Wearable Cognitive Assistance," *Proc. 12th Int'l Conf. Mobile Systems, Applications, and Services* (MobiSys 14), 2014, pp. 68–81.
11. H. Gupta et al., *iFogSim: A Toolkit for Modeling and Simulation of Resource Management Techniques*

It's already at your fingertips

If you do computational work, you are going to love *Computing in Science & Engineering (CiSE)*. Because *CiSE* appears in the IEEE Xplore and AIP library packages, representing more than 50 scientific and engineering societies, your institution is bound to have it.

in *Internet of Things, Edge and Fog Computing Environments*, tech. report CLOUDS-TR-2016-2, Cloud Computing and Distributed Systems Laboratory, Univ. of Melbourne, 2016; http://cloudbus.org/tech_reports.html.

12. I. Stojmenovic and S. Wen, "The Fog Computing Paradigm: Scenarios and Security Issues," *Proc. 2014 Federated Conf. Comp. Sci. and Info. Sys.* (FedCSIS 14), 2014, pp. 1–8.

13. H. Madsen et al., "Reliability in the Utility Computing Era: Towards Reliable Fog Computing," *Proc. 20th Int'l Conf. Systems, Signals, and Image Processing* (IWSSIP 13), 2013, pp. 43–46.

AMIR VAHID DASTJERDI is a Research Fellow with the University of Melbourne's Cloud Computing and Distributed Systems (CLOUDS) Laboratory. Contact him at amir.vahid@unimelb.edu.au.

RAJKUMAR BUYYA is director of the CLOUDS Laboratory and CEO of Manjrasoft Pty Ltd., a university spin-off company that commercializes cloud-computing innovations. Contact him at rbuyya@unimelb.edu.au.

Selected CS articles and columns are also available for free at **http://ComputingNow.computer.org**.