Network Working Group
Internet-Draft
Intended status: Informational
Expires: May 7, 2020

J. Hong
Y-G. Hong
ETRI
X. de Foy
InterDigital Communications, LLC
M. Kovatsch
Huawei Technologies Duesseldorf GmbH
E. Schooler
Intel
D. Kutscher
University of Applied Sciences Emden/Leer

November 04, 2019

IoT Edge Computing Challenges and Functions
draft-hong-t2trg-iot-edge-computing-01

Abstract

This document describes new challenges such as strict latency, uplink cost, uninterrupted services, privacy and security, for IoT services originated from the IoT environmental changes. In order to address those new challenges, the integration of Edge computing and IoT has emerged as a promising solution. This document describes the concept of IoT integrated with Edge computing as well as the state-of-the-art of IoT Edge computing. It also proposes a general model for IoT Edge computing. The direction of Edge computing for IoT should be discussed in the IETF/IRTF.

Table of Contents

1. Introduction

Nowadays, most IoT services are based on Cloud computing since it can provide virtually unlimited storage and processing power. The integration of IoT with Cloud computing brings many advantages such as flexibility, efficiency, and ability to store and use data.

However, the IoT environment is changing in such a way that vast amounts of data are created at edge/local networks and about a half of data is stored, processed, analyzed and acted upon close to the data producer. Thus, emerging IoT services introduce new challenges that cannot be addressed by today's centralized Cloud computing models alone.

In this document, we describe new challenges for emerging IoT services such as strict latency, uplink cost, uninterrupted services, privacy and security due to the IoT environmental changes.

In order to address those new challenges for IoT services, the integration of Edge computing with IoT has been emerged as a promising solution. In this document, we describe the concept of IoT integrated with Edge computing as well as the state-of-the-art of IoT Edge computing and propose an architecture of IoT Edge computing. The purpose of this document is to bring up the issues of Edge computing for IoT services in IETF/IRTF.

2. Conventions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Background

3.1. Internet of Things (IoT)

Since the phrase 'Internet of Things (IoT)' was coined by Kevin Ashton in 1999 working on Radio-frequency identification (RFID) technology at the Auto-ID Center of the Massachusetts Institute of Technology (MIT) [Ashton], the concept of IoT has been that things connected to the Internet can send and receive information collected by sensors without human intervention, where things are various embedded systems such as home appliances, mobile equipment, wearable devices, etc. IoT has become one of the notable innovations playing an important role in our daily lives [Lin]. IoT is generally characterized by real world small things that are widely distributed but have limited storage and processing power, which involve concerns regarding reliability, performance, security, and privacy.

3.2. Cloud Computing

Cloud computing have been defined in [NIST]: "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". Cloud computing has been a predominant technology which has virtually unlimited capacity in terms of storage and processing power. The availability of virtually unlimited storage and processing capabilities at low cost enabled the realization of a new computing model, in which virtualized resources can be leased in an on-demand fashion, being provided as general utilities. Companies like Amazon, Google, Facebook, etc. widely adopted this paradigm for delivering services over the Internet, gaining both economical and technical benefits [Botta].

Now with IoT, we will reach the era of post-Clouds where unprecedented volume and variety of data will be generated by things at edge/local networks and many applications will be deployed on the edge netwoks to consume these IoT data. Some of the applications may need very short response times, some may contain personal data, and others may generate vast amounts of data. Today's Cloud based service models are not suitable for these applications.

It is predicted that by 2019, 45% of the data created in IoT will be stored, processed, analyzed and acted close to, or at the edge of the network and about 50 billion devices will connect to the Internet by 2020 [Evans]. So, moving all data from edge/local networks to the cloud data center may not be an efficient way anymore to process vast amounts of data.

In Cloud computing, users traditionally only consumed IoT data through Cloud services. Now, however, users are also producing IoT data with their mobile devices. This change requires more functionality at edge/local networks [Shi].

3.3. Edge computing

Edge computing is a new paradigm in which substantial computing and storage resources are placed at the Internet's edge in close proximity to mobile devices or sensors so that computing happens near data sources [Mahadev]. It works on both downstream data on behalf of cloud services and upstream data on behalf of IoT services. An edge device is any computing or networking resource residing between data sources and cloud-based datacenters. In Edge computing, the end device not only consumes data but also produces data. And at the network edge, devices not only request services and information from the cloud but also handle computing tasks including processing, storage, caching, and load balancing on data sent to and from the cloud [Shi].

The definition of Edge computing from ISO is 'Form of distributed computing in which significant processing and data storage takes place on nodes which are at the edge of the network' [ISO_TR]. And the similar concept of Fog computing from Open Fog Consortium is 'A horizontal, system-level architecture that distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum' [OpenFog]. Based on these definitions, we can summarize a general philosophy of Edge computing as "Distribute the required functions close to users and data".

4. Challenges for IoT and Impacts of Edge Computing

As the IoT is maturing, systems are converging, deployments are growing, and IoT technology is used with more and more demanding applications such as industrial, automotive, or healthcare. This leads to new challenges for the IoT. In particular, the amount of data created at the edge is expected to be vast. Industrial machines such as laser cutters already produce over 1 terabyte per hour, the same applies for autonomous cars [NVIDIA]. 90% of IoT data is expected to be stored, processed, analyzed, and acted upon close to the source [Kelly], as Cloud Computing models alone cannot address the new challenges [Chiang].

4.1. Strict Latency and Jitter

Many industrial control systems, such as manufacturing systems, smart grids, oil and gas systems, etc., often require stringent end-to-end latency between the sensor and control node. While some IoT

applications may require latency below a few tens of milliseconds [Weiner], industrial robots and motion control systems have use cases for cycle times in the order of microseconds [_60802]. An important aspect for real-time communications is not only the latency, but also guarantees for jitter. This means control packets need to arrive with as little variation as possible with a strict deadline. Given the best-effort characteristics of the Internet, this challenge is virtually impossible to address with a pure cloud model, when also taking the further challenges into account.

4.2. Uplink Cost

Many IoT deployments are not challenged by a constrained network bandwidth to the cloud. The fifth generation mobile networks (5G) and Wi-Fi 6 both theoretically top out at 10 gigabits per second (i.e., 4.5 terabyte per hour), which enables high-bandwidth uplinks. However, the resulting cost for high-bandwidth connectivity to upload all data to the cloud is unjustifiable and impractical for most IoT applications.

4.3. Uninterrupted Services

Many IoT devices such as sensors, data collectors, actuators, controllers, etc. have very limited hardware resources and cannot rely solely on their limited resources to meet all their computing and/or storage needs. They require reliable, uninterrupted services to augment their capabilities in order to fulfill their application tasks. This is hard and partly impossible to achieve with cloud services for systems such as vehicles, drones, or oil rigs that have intermittent network connectivity. Example of related challenges include support for IoT device and Edge computing node mobility, as well as software instance migration.

4.4. Privacy and Security

When IoT services are deployed at home, personal information can be learned from detected usage data. For example, one can extract information about employment, family status, age, and income by analyzing smart meter data [ENERGY]. Policy makers started to provide frameworks that limit the usage of personal data and put strict requirements on data controllers and processors. However, data stored indefinitely in the cloud also increases the risk of data leakage, for instance, through attacks on rich targets.

Industrial systems are often argued to not have privacy implications, as no personal data is gathered. Yet data from such systems is often highly classified, as one might be able to infer trade secrets such

as the setup of production lines. Hence, the owner of these systems are generally reluctant to upload related IoT to the cloud.

5. IoT Edge Computing Model

It is expected Edge computing will play an important role to deploy new IoT services integrated with Big data, AI services. Although there are lots of approach to Edge computing, this section focus on common function of Edge computing, therefore draw an IoT Edge computing model. In this section we discuss a general model that aims to be applicable to multiple Edge computing architectures, such as:

- A single IoT gateway, or a hierarchy of IoT gateways, typically connected to the cloud (e.g., to extend the traditionally cloud-based management of IoT devices and data to the edge). A common role of an IoT Gateway is to provide access to an heterogeneous set of IoT devices/sensors; handle IoT data; and deliver IoT data to its final destination in a cloud network. Whereas an IoT gateway needs interactoins with cloud like as conventional Cloud computing, Edge computing can operate independently.

- A set of distributed computing nodes, e.g. embedded in switches, routers, edge cloud servers or mobile devices. In the future, some IoT end devices may have enough computing capabilities to participate in such distributed systems. In this model, each Edge computing node can collaborate with each other to share its resources to others or ask other's resources.

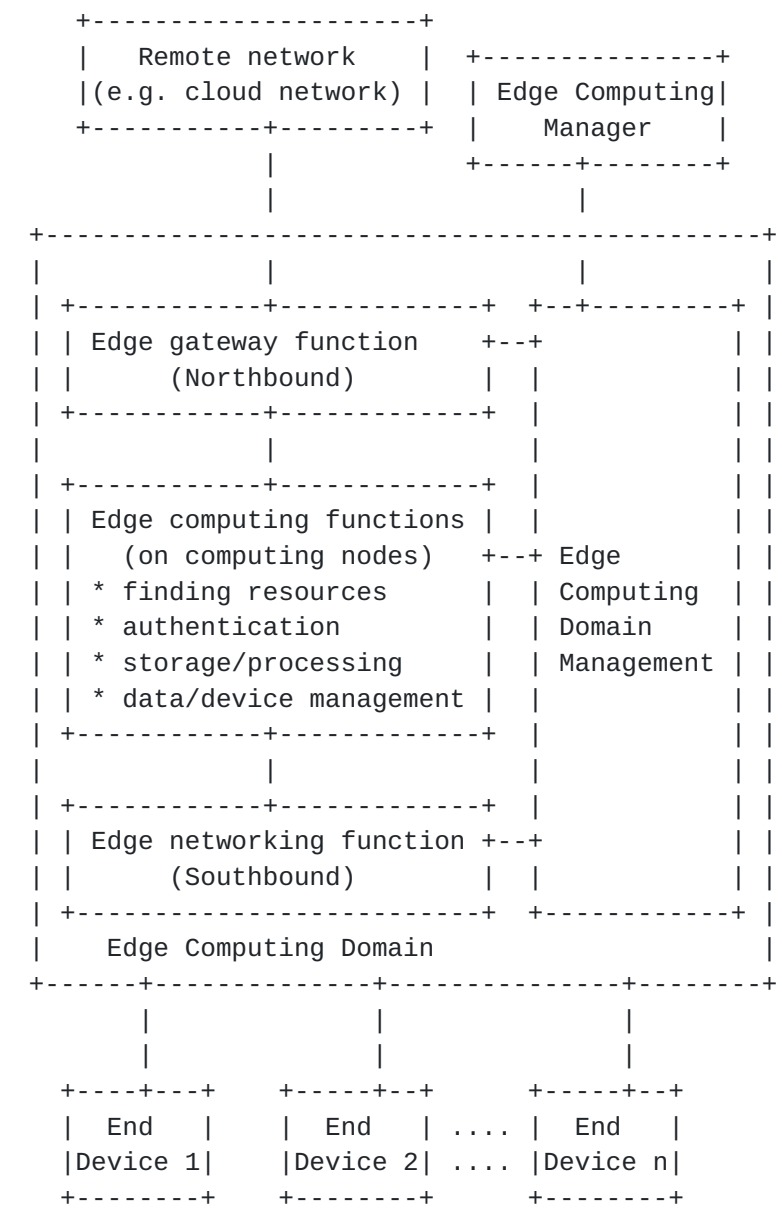

Figure 1: Model of IoT integrated with Edge computing

The Edge computing domains is interconnected with IoT end devices (southbound connectivity) and possibly with a remote/cloud network (northbound connectivity). Edge computing nodes provide multiple logical functions such as finding resources, authentication, storage/ processing and management.

5.1. Gateway Function and Remote Network

A northbound interface is provided by a gateway component to a remote network, e.g. a cloud, home or enterprise network. These components may not exist in standalone scenarios, where Edge computing is

provided locally without a connection to a remote network. The northbound interface is a data plane interface. Nevertheless, the remote network may also host an edge cloud manager function.

5.2. Edge Computing Domain Management and Manager Role

Edge computing domain management includes management of resources and functions in the Edge computing domain. Management of IoT end devices and IoT data may be included or may be part of the Edge computing logical functions (OPEN QUESTION). The management function can provides SaaS, PaaS, IaaS service APIs to an Edge computing manager. The edge management role may be taken by an entity in the cloud, but it may also be a local entity, or even be non-existent (during normal operation) in autonomic systems.

5.3. Edge Computing Logical Functions

Edge computing nodes host logical functions relative to:

- Finding resources, such as compute, storage or data resources;
- Authenticating platforms, end devices, functions, data;
- Providing compute and storage offloading;
- Management, e.g. of IoT end devices and data.

With regard to the high level challenges listed in Section 4, data storage and processing at the edge is a major aspect of IoT Edge computing. Data may therefore need to be classified (e.g. in terms of privacy, importance, validity, etc.). Data analysis such as performed in AI/ML tasks performed at the edge may benefit from specialized hardware support on computing nodes. IoT Edge computing will face detailed challenges in term of, for example, programmability, naming, data abstraction and service management. Furthermore, while Edge computing can support IoT services independently of Cloud computing, it is increasingly connected to Cloud computing in most IoT systems: thus, the relationship of IoT Edge Computing to Cloud Computing is another potential challenge [ISO_TR].

5.4. Edge Networking Function and IoT End Devices

IoT end devices can be sensors, actuators, or more generally IoT things. Not only the big volume of IoT data but also the massive number of IoT end devices are the cause of a massive scalability issue in future IoT environments. To address this challenge Edge

computing separates the scalability domain into edge/local networks and remote network.

Edge computing nodes communicate between themselves and with end devices over an underlying network. There is therefore a need for the Edge computing domain to directly or indirectly control those network functions.

6. State-of-the-Art of IoT Edge Computing

6.1. Common aspects of IoT Edge computing service platforms

This section provides an overview of today's IoT Edge Computing field, based on a limited review of standards, research, open-source and proprietary products in Appendix A.

Common aspects of IoT Edge computing service platforms are summarized here:

Computing devices: IoT gateways (Appendix A.2.1, Appendix A.1.1) represent a common class of IoT Edge computing products, where the gateway is providing a local service on customer premises, and is remotely managed through a cloud service. IoT communication protocols are typically used between IoT devices and the gateway, including CoAP, MQTT and many specialized IoT protocols, while the gateway communicates with the distant cloud using typically HTTP and WebSocket. Virtualization platforms enable the deployment of virtual Edge computing functions, including IoT gateway software, on servers in the mobile network infrastructure (at base station and concentration points), in edge datacenters (in central offices) or regional datacenters located near central offices. End devices as computing devices are envisioned in fog architecture and research projects, but are not commonly used as such today.

Service models: Physical or virtual IoT gateways can host application programs built using an SDK. Edge cloud system operators host their customers' applications VMs or containers on servers located in or near access networks. These application have access to edge service APIs. For example, mobile network services include radio network information, location, bandwidth management. In a cloud-like service model, service providers consume low-level edge platform APIs and offer high-level APIs to their own customers' applications. This cloud-like model can be offered as an edge cloud service, or as an hybrid cloud service covering edge and distant cloud.

Management: Life cycle management of services and applications on physical IoT gateways is often cloud-based. Edge cloud management platforms and products (Appendix A.1.2, Appendix A.2.2) adapt cloud management technologies (e.g. kubernetes) to the edge cloud, i.e. to smaller, distributed computing devices running outside a controlled data center. Services and application life-cycle is typically using a NFV-like management and orchestration model.

Communication services: The platform typically includes services to advertise or consume APIs, and enables communicating with local and remote endpoints. The service platform is typically extensible by edge applications, since they can advertise an API that other edge applications can consume. IoT communication services include protocols translation, analytics and transcoding. Communication between Edge computing devices is enabled in tiered deployments or distributed deployments.

Storage models: An edge cloud platform may enable pass-through without storage, local storage (e.g. on IoT gateways). Some edge cloud platforms use a distributed form of storage, e.g. an ICN network or a distributed storage platform. External storage, e.g. on databases in distant or local IT cloud, is typically used for filtered data deemed worthy of long term storage, or in some cases for all data, for example when required for regulatory reasons.

Computing models: Stateful computing is supported on platforms hosting native programs, VMs or containers. Stateless computing is supported on platforms providing a "serverless computing" service (a.k.a. function-as-a-service), or on systems based on named function networking.

Network traffic patterns: Network traffic is typically high volume uplink with throttling by Edge computing devices (or deferred to off-peak hours or using physical shipping); and downlink for control and software updates.

6.2. Use Cases of IoT Edge Computing

6.2.1. Smart Constructions

In traditional construction domain, there are many heavy equipment and machineries and dangerous elements. Even though human pay attention to risk elements, it is not easy to avoid them. If some accidents are happened in a construction site, it causes a loss of lives and property. Thus, there have been many trials in a construction area to protect lives and property.

Measurements of noise, vibration, and gas in a construction area are recorded on a remote server and reported to an inspector. Today, data produced bu such measurements is collected by a gateway in a construction area and transferred to a remote server. This incurs transmission cost, e.g. over a LTE connection, and storage cost, e.g. when using Amazon Web Services. When an inspector wants to investigate some accidents, he checks the information stored in a server.

If we deploy Edge computing in a construction area, the sensor data can be processed and analyzed in a gateway located within or near a construction area. And with the help of a statistical analysis or machine learning technologies, we can predict future accidents in advance and this prediction can be used as an alarm in a construction area and a notification to an inspector.

To determine the exact cause of some accident, not only sensor data but also audio and video data are transferred to a remote server or cloud networks. In this case, the data volume of audio and video is quite big and the cost of transmission can be a problem. If Edge computing can predict the time of accident, it can reduce the data volume of transmission; in general period, it can transmit the audio and video data with a low resolution/degree and in emergent period, it transmits the audio and video data with a high resolution/degree. By adjusting the resolution/degree of audio and video data, it can reduce transmission cost significantly.

6.2.2. Smart Grid

In future smart cities, Smart grids will be critical in ensuring availability and efficiency for energy saving and control in city-wide electricity management. Edge computing is expected to play a significant role in those systems to improve transmission efficiency of electricity, react and restore for power disturbances, reduce operation cost, reuse renewable energy effectively, save energy of electricity for future usage, and so on. In addition, Edge computing can help monitoring power generation and power demands, and making electrical energy storage decisions in the Smart grid system.

6.2.3. Smart Water System

The Water system is one of the most important aspects for building smart city. Effective use of water, and cost-effective and environment-friendly treatment of water are critical for water control and management. This can be facilitated by Edge computing in Smart water systems, to help monitor water consumption, transportation, prediction of future water use, and so on. For example, water harvesting and ground water monitoring will be

supported from Edge computing. Also, a Smart water system is able to analyze collected information related to water control and management, control the reduction of water losses and improve the city water system through Edge computing.

7. Security Considerations

T.B.D.

8. Acknowledgement

The authors would like to thank Joo-Sang Youn and Akbak Rahman for their valuable comments and suggestions on this document.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

9.2. Informative References

[_3GPP.23.501] 3GPP, ., "System Architecture for the 5G System", 3GPP TS 23.501 , 2019, <http://www.3gpp.org/ftp/Specs/html-info/23501.htm>.

[_5G-CORAL] Horizon 2020 Programme, ., "5G Convergent Virtualised Radio Access Network Living at the Edge (5G-CORAL) Project", Portal , 2019, <http://5g-coral.eu/>.

[_60802] IEC/IEEE, ., "Use Cases IEC/IEEE 60802 V1.3", IEC/IEEE 60802 , 2018, <http://www.ieee802.org/1/files/public/docs2018/60802-industrial-use-cases-0818-v13.pdf>.

[Ashton] Ashton, K., "That Internet of Things thing", RFID J. vol. 22, no. 7, pp. 97-114 , 2009.

[Botta] Botta, A., Donato, W., Persico, V., and A. Pescape, "Integration of Cloud computing and Internet of Things: A survey", Future Gener. Comput. Syst., vol. 56, pp. 684-700 , 2016.

[Chiang] Chiang, M. and T. Zhang, "Fog and IoT: An overview of research opportunities", IEEE Internet Things J., vol. 3, no. 6, pp. 854-864 , 2016.

[ENERGY] Beckel, C., Sadamori, L., Staake, T., and S. Santini, "Revealing Household Characteristics from Smart Meter Data", Energy, vol. 78, pp. 397-410 , 2014.

[ETSI_MEC_02] ETSI, ., "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements", ETSI GS 002 , 2016, <https://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.01.01_60/gs_MEC002v020101p.pdf>.

[ETSI_MEC_03] ETSI, ., "Mobile Edge Computing (MEC); Framework and Reference Architecture", ETSI GS 003 , 2019, <https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.01.01_60/gs_MEC003v020101p.pdf>.

[ETSI_MEC_WP_28] ETSI, ., "MEC in 5G networks", White Paper , 2018, <https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp28_mec_in_5G_FINAL.pdf>.

[Evans] Evans, D., "The Internet of Things: How the next evolution of the Internet is changing everything", CISCO White Paper , 2011.

[FLAME] Horizon 2020 Programme, ., "FLAME Project", Portal , 2019, <https://www.ict-flame.eu/>.

[IEEE-1934] IEEE, ., "FOG - Fog Computing and Networking Architecture Framework", Portal , 2019, <https://standards.ieee.org/standard/1934-2018.html>.

[ISO_TR] "Information Technology - Cloud Computing - Edge Computing Landscape", ISO/IEC TR 23188 , 2018.

[Kelly] Kelly, R., "Internet of Things Data to Top 1.6 Zettabytes by 2022", 2016, <https://campustechnology.com/articles/2015/04/15/internet-of-thingsdata-to-top-1-6-zettabytes-by-2020.aspx>.

[Lin] Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications", IEEE Internet of Things J., vol. 4, no. 5, pp. 1125-1142 , 2017.

[Linux_Foundation_Edge] Linux Foundation, ., "Linux Foundation Edge", Portal , 2019, <https://www.lfedge.org/>.

[Mahadev] Satyanarayanan, M., "The Emergence of Edge Computing", Computer, vol. 50, no. 1, pp. 30-39 , 2017.

[NIST] Mell, P. and T. Grance, "The NIST definition of Cloud computing", Natl. Inst. Stand. Technol, vol. 53, no. 6, p. 50 , 2009.

[NVIDIA] Grzywaczewski, A., "Training AI for Self-Driving Vehicles: the Challenge of Scale", NVIDIA Developer Blog , 2017, <https://devblogs.nvidia.com/training-self-driving-vehicles-challenge-scale/>.

[OpenEdgeComputing] "Open Edge Computing", Portal , 2019, <http://openedgecomputing.org/>.

[OpenFog] "OpenFog Reference Architecture for Fog Computing", OpenFog Consortium , 2017.

[POINT] Horizon 2020 Programme, ., "IP Over ICN - the better IP (POINT) Project", Portal , 2019, <https://www.point-h2020.eu/>.

[Shi] Shi, W., Cao, J., Zhang, Q., Li, Y., and L. Xu, "Edge computing: vision and challenges", IEEE Internet of Things J., vol. 3, no. 5, pp. 637-646 , 2016.

[Sifalakis] Sifalakis, M., Kohler, B., Scherb, C., and C. Tschudin, "An Information Centric Network for Computing the Distribution of Computations", Proceedings of the 1st International Conference on Information-centric networking (INC) , 2014.

[StarlingX] OpenStack Foundation, ., "StarlingX", Portal , 2019, <https://www.starlingx.io/>.

[Weiner] Weiner, M., Jorgovanovic, M., Sahai, A., and B. Nikolie, "Design of a low-latency, high-reliability wireless communication system for control applications", IEEE Int. Conf. Commun. (ICC), Sydney, NSW, Australia, pp. 3829-3835 , 2014.

Appendix A. Overview of the IoT Edge Computing

This list of initiatives, projects and products aim to provide an overview of the IoT Edge Computing.

Our goal is to be representative rather than exhaustive.

Please help us complete this overview by communicating with us about entries we have missed.

A.1. Open Source Projects

A.1.1. Gateway/CPE Platforms

EdgeX Foundry, Home Edge, Edge Virtualization Engine are Linux Foundation projects ([Linux_Foundation_Edge]) aiming to provide a platform for edge computing devices.

Such an open source platform can, for example, host proprietary programs currently run on IoT gateway products (Appendix A.2).

EdgeX Foundry develops an edge computing framework running on the IoT gateway.

Home Edge develops an edge computing framework especially dedicated to home computing devices, controlling home appliances, sensors, etc., and enabling AI applications, especially distributed and parallel machine learning.

The Edge Virtualization Engine (EVE) project develops a virtualization platform (for VMs and containers) designed to run outside of the datacenter, in an edge network; EVE is deployed on bare-metal hardware.

Computing devices: Hardware support for EdgeX and EVE is similar: they support x86 and ARM-based computing devices; A typical target can be a Linux Raspberry Pi with 1GB RAM, 64bit CPU, 32GB storage.

Service platform: EdgeX uses a micro-service architecture. Micro-services on the gateway are connected together, and to outside applications, through REST, or messaging technologies such as MQTT, AMQP and 0MQ. The gateway can communicate with external

backend applications or other gateways (north-south in tiered deployments or east-west in more distributed deployments). Gateway-device communication can use a wide range of IoT protocols. "Export services" enable on-gateway and off-gateway clients to register as recipient for data from devices. Core services are microservices that deal with persisting data from devices or alternatively "streaming" device data through, without persistence (core data service); managing information about the IoT devices, including their sensors, how to communicate with them, etc. (metadata service); and actual communication with IoT devices, on behalf of other on-gateway or off-gateway services (command service). A rule engine provides an API to register actions in response to conditions typically including an IoT device ID, sensor values to check, thresholds, etc. The scheduling micro service deals with organizing the removal of data persisted on the gateway. Alerts and notifications microservice can be used to dispatch alert/notifications from internal or external sources to interested consumers including backend servers, or human operators through email or SMS.

Edge cloud applications: Target applications for EdgeX include industrial IoT (e.g. IoT sensor data and actuator control mixed with augmented reality application for technicians). Home Edge focuses on smart home use cases, including using AI lifestyle and safety applications.

A.1.2. Edge Cloud Management Platforms

This set of open-source projects setup and manage clouds of individual edge computing devices.

StarlingX ([StarlingX]) extends OpenStack to provide virtualization platform management for edge clouds, which are distributed (in the range of 100 compute devices), secure and highly available.

Akraino Edge Stack, another project from the Linux Fundation Edge [Linux_Foundation_Edge], has a wider scope of developing a management platform adapted for the edge (e.g., covering 1000 plus locations), aiming for zero-touch provisioning, and zero-touch lifecycle management.

Computing devices: Compute devices are typically Linux-based application servers or more constrained devices.

Service platform: StarlingX adds new management services to OpenStack by leveraging building blocks such as Ceph for distributed storage, Kubernetes for orchestration. The new services are for management of configuration (enabling auto-

discovery and configuration), faults, hosts (enabling host failure detection and auto-recovery), services (providing high availability through service redundancy and multi-path communication) and software (enabling updates).

Edge cloud applications: An edge computing platform may support a wide range of use cases. E.g., autonomous vehicles, industrial automation and robotics, cloud RAN, metering and monitoring, mobile HD video, content delivery, healthcare imaging and diagnostics, caching and surveillance, augmented/virtual reality, small cell services for high density locations (stadiums), universal CPE applications, retail.

A.1.3. Related Projects

Open Edge Computing ([OpenEdgeComputing]) is an initiative from universities, manufacturers, infrastructure providers and operators, enabling efficiently offloading cloudlets (VMs) to the edge. Computing devices are typically powerful, well-connected servers located in mobile networks (e.g. collocated with base stations or aggregation sites). The service platform is built on top of OpenStack++, an extension of OpenStack to support cloudlets. This project is mentioned here as a related project because of its edge computing focus, and potential for some IoT use cases. Nevertheless, its primary use cases are typically non-IoT related, such as offloading processing-intensive applications from a mobile device to the edge.

A.2. Products

A.2.1. IoT Gateways

Multiple products are marketed as IoT gateways (Amazon Greengrass, Microsoft Azure IoT Edge, Google Cloud IoT Core, and gateway solutions from Bosh and Siemens). They are typically composed of a software frameworks that can run on a wide range of IoT gateway hardware devices to provide local support for cloud services, as well as some other local IoT gateway features such as relaying communication and caching content. Remote cloud is both used for management of the IoT gateways, and for hosting customer application components. Some IoT gateway products (Amazon Snowball) have a primary purpose of storing edge data on premises, to enable physically moving this data into the cloud without incurring digital data transfer cost.

Computing devices: Typical computing devices run Linux, Windows or a Real-Time OS over an ARM or x86 architecture. The level of service support on the computing device can range from low-level

packages giving maximum control to embedded developers, to high-level SDKs. Typical requirements can start at 1GHz and 128MB RAM, e.g. ranging from Raspberry Pi to a server-level appliance.

Service platform: IoT gateways can provide a range of service including: running stateless functions; routing messages between connected IoT devices (using a wide range of IoT protocols); caching data; enabling some form of synchronization between IoT devices; authenticating and encrypting device data. Association between IoT devices and gateway based can require a device certificate.

Edge cloud applications: Pre-processing of IoT data for later processing in the Cloud is a major driver. Use cases include industrial automation, farming, etc.

A.2.2. Edge Cloud Platforms

Services such as MobileEdgeX provide a platform for application developers to deploy software (e.g. as software containers) on edge networks.

Computing devices: Bare metal and virtual servers provided by mobile network operators are used as computing devices.

Service platform: The service platform provides end device location service, using GPS data obtained from platform software deployed in end devices, correlated with location information obtained from the mobile network. The service platform manages the deployment of application instances (containers) on servers close to end devices, using a declarative specification of optimal location from the application provider.

Edge cloud applications: Use cases include autonomous mobility, asset management, AI-based systems (e.g. quality inspection, assistance systems, safety and security cameras) and privacy-preserving video processing. There are also non-IoT use cases such as augmented reality and gaming.

A.3. Standards Initiatives

A.3.1. ETSI Multi-access Edge Computing

The ETSI MEC industry standardization group develops specifications that enable efficient and seamless integration of applications from vendors, service providers, and 3rd parties across multi-vendor MEC platforms ([ETSI_MEC_03]).

Basic principles followed include: leveraging NFV infrastructure; being compliant with 3GPP systems; focusing on orchestration, MEC services, applications and platforms.

Phase 1 (2015-2016) focused on basic platform services. Phase 2 (2017-2019) focuses on: supporting non-3GPP radio access technologies, especially WiFi; supporting a distributed, multi-operator and multi-vendor architecture; supporting non-VM based virtualization such as containers and PaaS.

Computing devices: Computing devices are typically application servers, attached to an eNodeB or at a higher level of aggregation point, and provide service to end users.

Service platform: The mobile edge platform offers an environment where the mobile edge applications can discover, advertise, consume and offer mobile edge services. The platform can provide certain native services such as radio network information, location, bandwidth management etc. The platform manager is responsible for managing the life cycle of applications including informing the mobile edge orchestrator of relevant application related events, managing the application rules and requirements including service authorizations, traffic rules, DNS configuration.

Edge cloud applications: Some of the use cases for MEC ([ETSI_MEC_02]) are IoT-related, including: security and safety (face recognition and monitoring), sensor data monitoring, active device location (e.g., crowd management), low latency vehicle-to-infrastructure and vehicle-to-vehicle (V2X, e.g., hazard warnings), video production and delivery, camera as a service.

A.3.2. Edge Computing Support in 3GPP

The 3GPP standards organization included edge computing support in 5G [_3GPP.23.501]. Integration of MEC and 5G systems has been studied in ETSI as well [ETSI_MEC_WP_28].

Computing devices: From 3GPP standpoint, a mobile device may access any computing device located in a local data network, i.e. traffic is steered towards the local data network where the computing device is located.

Service platform: An external party may influence steering, QoS and charging of traffic towards the computing device. Session and service continuity can ensure that edge service is maintained when a client device moves. The network supports multiple-anchor connections, which makes it possible to connect a client device to

both a local and a remote data network. The client device can be made aware of the availability of a local area data network, based on its location.

Edge cloud applications: Edge cloud applications in 3GPP can help support the major use cases envisioned for 5G, including massive IoT and V2X.

A.3.3. OpenFog Consortium

The OpenFog Consortium (now part of the Industrial Internet Consortium) aims to standardize industrial IoT, fog and edge computing. It produced a reference architecture for the Fog ([OpenFog]), which has been published as IEEE standard P1934 in 2018.

Computing devices: Fog nodes include computational, networking, storage and acceleration elements. This includes nodes collocated with sensors and actuators, roadside or mobile nodes involved in V2X connectivity. Fog nodes should be programmable and may support multi-tenancy. Fog computing devices must employ a hardware-based immutable root of trust, i.e. a trusted hardware component which receives control at power-on.

Service platform: The service platform is structured around "pillars" including: security end-to-end, scalability by adding internal components or adding more fog nodes,openness in term of discovery of/by other nodes and networks, autonomy from centralized clouds (for discovery, orchestration and management, security and operation) and hierarchical organization of fog nodes.

Edge cloud applications: Major use cases include smart cars and traffic control, visual security and surveillance, smart cities.

A.3.4. Related Standards

The IEEE Fog Computing and Networking Architecture Framework Working Group [IEEE-1934] published the OpenFog architecture as an IEEE document, and plan to do further work on taxonomy, architecture framework, and compliance guidelines.

A.4. Research Projects

A.4.1. Named Function Networking

Named Function Networking ([Sifalakis]) is a research project that aims to extend ICN concepts (especially named data networking) to have the network orchestrate computation. Interests are sent for a

combination of function and argument names, instead of using the content name in NDN.

Computing devices: NFN-capable switches are collocated with computing devices.

Service platform: NFN enables accessing static data and dynamic computation results in one data-oriented framework, thus benefiting from usual ICN features such as data authenticity and caching, as well as enabling the network to perform various optimizations, e.g. moving data, code or both closer to requesters. NFN also enables secure access to individual elements within Named Data Objects, e.g. for filtering or aggregation.

Edge cloud applications: Use cases include some form of MapReduce operations and service chaining. NDN, on which NFN is based, has been studied in the context of IoT, where it can provide local trust management and rendezvous service.

A.4.2. 5G-CORAL

The 5G-CORAL project ([5G-CORAL]) aims to enable convergence of access across multiple RATs using Fog computing, using for this purpose an Edge and Fog Computing System (EFS).

Computing devices: Computing devices used in 5G-CORAL include cloud and central data center servers, edge data center servers, and fixed or mobile "Fog Computing Devices", which can be computing devices located in vehicles or factories, e.g. IoT gateways, mobile phones, cyber-physical devices, etc.

Service platform: 5G-CORAL architecture is based on an integrated virtualized edge and fog computing system (EFS), that aims to be flexible, scalable and interoperable with other domains including transport (fronthaul, backhaul), core and clouds. An Orchestration and Control System (OCS) enables automatic discovery of heterogeneous, multiple-owner resources, and federate them into a unified hosting environment. OCS monitors resource usage to guarantee service levels. Finally, OCS also includes orchestration and life cycle functions, including live migration and scaling. Applications (user and third-party) both inside and outside the EFS subscribe to EFS services through APIs, with emphasis on IoT and cyber-physical functionalities.

Edge cloud applications: EFS-hosted services include analytics obtained from IoT gateways (e.g. LORA or eNodeB gateways), context information services from RATs, transport (fronthaul and backhaul) and core networks. EFS-hosted functions include network

performance acceleration functions, virtualized C-RAN functions for access nodes and possible end user devices.

A.4.3. FLAME

The FLAME project ([FLAME]) aims to improve performance of interactive media systems while keeping infrastructure costs low.

It builds over virtualization technologies such as XOS, OpenStack and ONOS/ODL to offer a programmable media service platform.

FLAME leverages IP-over-ICN technology developed through earlier projects including POINT ([POINT]).

Computing devices: The FLAME platform provides a service layer on top of an infrastructure platform, which can include cloud servers as well as computing devices collocated with WiFi access points.

Service platform: The FLAME platform can be seen as an edge + cloud computing platform with a use case focus on media dissemination, although the basic platform can be suitable for micro-services in general. The computing platform is comprised of: computing devices, an infrastructure platform (XOS, OpenStack, ONOS/ODL), NFV-MANO components (orchestrator, virtual infrastructure manager) and FLAME platform core services (PCE, network access point, surrogate manager).

Edge cloud applications: IoT use cases include public safety, such as supporting body-worn camera for police and social workers. As opposed to other multi-media applications that are also envisioned (pre-processing, user reporting, curation...), where a typical goal is to curate content early at the edge, to reduce expected high data volume, public safety use cases are typically about implementing triggers at the edge: everything needs to be kept anyway, to be available in case of an audit. Content is stored offline during off peak-hours delivery. For privacy and data volume concerns, triggers for, e.g., alerting police, cannot be performed in the cloud and should be performed as close to the data source as possible.

Authors' Addresses

Jungha Hong
ETRI
218 Gajeong-ro, Yuseung-Gu
Daejeon 34129
Korea

Email: jhong@etri.re.kr

Yong-Geun Hong
ETRI
218 Gajeong-ro, Yuseung-Gu
Daejeon 34129
Korea

Email: yghong@etri.re.kr

Xavier de Foy
InterDigital Communications, LLC
1000 Sherbrooke West
Montreal H3A 3G4
Canada

Email: xavier.defoy@interdigital.com

Matthias Kovatsch
Huawei Technologies Duesseldorf GmbH
Riesstr. 25 C // 3.OG
Munich 80992
Germany

Email: ietf@kovatsch.net

Eve Schooler
Intel
2200 Mission College Blvd.
Santa Clara, CA 95054-1537
USA

Email: eve.m.schooler@intel.com

Dirk Kutscher
University of Applied Sciences Emden/Leer
Constantiaplatz 4
Emden 26723
Germany

Email: ietf@dkutscher.net