Problem Statement of IoT integrated with Edge Computing
draft-hong-t2trg-iot-edge-computing-00

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 been 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 an architecture of IoT Edge computing. The direction of Edge computing for IoT should be discussed in the IETF/IRTF.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 9, 2020.
Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the
document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal
Provisions Relating to IETF Documents
(https://trustee.ietf.org/license-info) in effect on the date of
publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction .......................................................... 3
2. Conventions and Terminology ........................................ 3
3. Background ............................................................ 4
   3.1. Internet of Things (IoT) .................................... 4
   3.2. Cloud computing .............................................. 4
   3.3. Edge computing ............................................... 5
4. New challenges of IoT .................................................. 5
   4.1. Strict Latency and Jitter .................................... 5
   4.2. Uplink Cost .................................................... 6
   4.3. Uninterrupted Services ........................................ 6
   4.4. Privacy and Security ......................................... 6
5. IoT integrated with Edge Computing ................................. 7
   5.1. IoT Data in Edge Computing ................................ 7
       5.1.1. Data Storage ............................................ 8
       5.1.2. Data Processing ......................................... 8
       5.1.3. Data Analyzing .......................................... 8
   5.2. IoT Device Management in Edge Computing .................... 9
6. Architecture of IoT integrated with Edge Computing ............. 9
7. State-of-the-art of IoT Edge Computing ............................ 11
   7.1. Common aspects of IoT edge computing service platforms . 11
   7.2. Use Cases of IoT Edge Computing ............................. 12
8. Security Considerations .............................................. 14
9. Acknowledgements .................................................... 14
10. References .......................................................... 14
    10.1. Normative References ..................................... 14
    10.2. Informative References .................................... 14
Appendix A. Overview of the IoT Edge Computing ...................... 17
   A.1. Open Source Projects ........................................ 17
       A.1.1. Gateway/CPE Platforms .................................. 17
       A.1.2. Edge Cloud Management Platforms ....................... 18
       A.1.3. Related Projects ....................................... 19
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 networks 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. New challenges of IoT

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.

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 integrated with Edge Computing

As described in section 4, there are new challenges for supporting emerging IoT services and Edge computing is one of the candidates to satisfy these challenges. The motivation for IoT Edge computing was discussed at an Edge computing discussion in IETF/IRTF meetings as follows: [IETF_Edge]

- Delay-sensitive
- High-volume
- Trust-sensitive
- (Intermittently) disconnected
- Energy-challenged
- Costly to transmit

As we described at previous sections, the above motivation for IoT Edge computing could directly be benefits of Edge computing in the IoT environment. The above motivation for IoT Edge computing is mainly related to IoT data and other motivation for IoT Edge computing can exist as other aspects of networking and communication.

In spite of its benefits, Edge computing in IoT services has challenges such as programmability, naming, data abstraction, service management, privacy and security and optimization metrics.

Edge computing can support IoT services independently of Cloud computing. However, Edge computing is increasingly connected to Cloud computing in most IoT systems for processing and storing data. Thus, the relationship of Edge Computing to Cloud Computing is also another challenge of Edge Computing in IoT [ISO_TR].

5.1. IoT Data in Edge Computing

As an aspect of IoT, Edge computing can provide many capabilities for IoT services because IoT systems are based on sensors and actuator devices in edge area and IoT data generated from sensors and actuator devices are gathered through a gateway [ISO_TR]. Besides on IoT data, other functions such as computing, control and network functions are also very remarkable to support IoT services. In this document, we will first concentrate on IoT data’s aspect since the benefit of Edge computing with IoT data is very big in use cases.
5.1.1. Data Storage

As tremendous IoT sensors, IoT actuators, and IoT devices are connected to the Internet, IoT data volume from these things are expected to increase explosively. And it is expected that much of this high volume of IoT data is produced and/or consumed within edge/local networks, not to traverse through cloud networks. Until now, most IoT data generated by IoT things is transferred and accumulated in a remote server and storage of IoT data in a remote server is expensive in transmission and storage. To mitigate the cost of transmission and storage, it is required to divide IoT data into two types of data; one is stored in edge/local networks and the other is stored in cloud networks. The effect of Edge computing is revealed with the handling IoT data in edge/local networks.

5.1.2. Data Processing

Until now, most network equipment such as routers, gateways, and switches just forward data delivered from other network devices without reading or modifying the content. In end-to-end communication, data is acknowledged and proceed at a final corresponding node. This is a typical usage of cloud computing and a client-server communication. But, in the IoT environment, some IoT data will be transferred to a cloud network and some will be delivered to an edge node. The main reason of this separation is to provide real-time processing and security enhancement in IoT. Although there are many new technologies to reduce the delay and transmission time, it is not easy to guarantee real-time processing. The typical use case of this requirement is industrial Internet and smart factory. Even though there are also several solutions to provide security in IoT, the more basic rule is not to expose the privacy data to public networks. If we separate IoT data into private and non-private data, and keep private data within an edge/local network not to expose them in a public network, the security and privacy in IoT can be addressed by the separation.

5.1.3. Data Analyzing

If it is possible to separate IoT data in edge/local networks and cloud networks, Edge computing can do more functions with IoT data in edge/local networks. Because Edge computing has the capabilities to handle IoT data in edge/local networks, it is also possible to analyze IoT data to provide enhanced IoT services such as intelligence. To analyze IoT data in an edge/local network, it is required to have comparatively processing performance and this requirement is not obstacle to deploy Edge computing due to the development of H/W and S/W.
5.2. IoT Device Management in Edge Computing

If we consider new challenges of IoT services, not only the big volume of IoT data but also the massive number of IoT things can be a critical problem. Even though, we acknowledge this future problem, the Internet architecture originally has the capability of scalability and it will mitigate scalability issue in the IoT environment. But, we cannot estimate the number of IoT things in the future and we cannot guarantee the Internet architecture still sustain the scalability issue in the IoT environment. Edge computing will separate the scalability domain into edge/local networks and outside network (e.g., cloud networks) and this separation of scalability domain can provide more efficient way to tackle the massive number of IoT things.

Because Edge computing can handle IoT data in an edge area and store the IoT data in an edge node, and proceed IoT data if it is needed, it can also separate the management domain into two parts. Edge Computing can concentrate on management of IoT things in an edge area and cooperate with the management of other outside networks.

6. Architecture of IoT integrated with Edge Computing

When we consider the implementation and deployment of Edge computing, it can be mainly referred to an IoT Gateway. The role of an IoT Gateway is to provide multiple accesses to the heterogeneous IoT devices/sensors, handling IoT data and delivering the IoT data to the final destinations such as cloud networks. Similar to an IoT Gateway, an Edge computing architecture as an edge computing node provides downside connectivity to IoT sensors and devices (southbound connectivity) and upside connectivity to cloud networks (northbound connectivity). Also, the architecture provides the function of data storage. Beside these functions, the Edge computing architecture should provide the computing functions, such as data processing, data analyzing, and additional function of intelligence.
Figure 1: Architecture of IoT integrated with Edge computing
It is expected that the Edge computing architecture will play an important role to deploy new IoT services with integration to big data and AI services.

7. State-of-the-art of IoT Edge Computing

7.1. 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.

7.2. Use Cases of IoT Edge Computing

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 by 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.

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.

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.

Smart Buildings: [TBA]

Smart Cities: [TBA]
Connected Vehicles: [TBA]

8. Security Considerations

[TBA]

9. Acknowledgements

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

10. References

10.1. Normative References


10.2. Informative References


[ETSI_MEC_WP_28]
ETSI, "MEC in 5G networks", White Paper, June 2018,

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

[StarlingX]
OpenStack Foundation, "StarlingX", Portal, 2019,

[Sifalakis]
Sifalakis, M., Kohler, B., Scherb, C., and C. Tschudin,
"An Information Centric Network for Computing the Distribution of Computations",

[FLAME]
Horizon 2020 Programme, "FLAME Project", Portal, 2019,

[POINT]
Horizon 2020 Programme, "IP Over ICN - the better IP (POINT) Project", Portal, 2019,

[_5G-CORAL]

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

[IEEE-1934]
IEEE, "FOG - Fog Computing and Networking Architecture Framework", Portal, 2019,

[NVIDIA]
Grzywaczewski, A., "Training AI for Self-Driving Vehicles: the Challenge of Scale",
NVIDIA Developer Blog, October 2017,
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 OMQ. 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 Foundation 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 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


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