Edge Computing

1st Aditya Mehta
University of Oulu
Oulu, Finland
amehta20@student.oulu.fi

2nd Andrei Baraian
University of Oulu
Oulu, Finland
abaraian19@student.oulu.fi

3rd Baptiste Hutteau
University of Oulu
Oulu, Finland
bhutteau19@student.oulu.fi

***Abstract*—With the advancement in network availability, continuous adoption of internet and cloud computing technologies, the number of internet users have increased phenomenally. With this proliferation of the number of devices connected to the internet, there is an unprecedented need for data processing and storage. Traditionally, the data has been sent from the source to the data centres for processing through the internet. However, with billions of devices on the internet, data generation speed has been comparatively higher than data transmission speed. Cloud computing often involves latency due to back and forth data transmission, therefore it is proving to be inadequate for time-sensitive applications. Edge computing paradigm suggests that rather than transmitting data through the constrained networks, the computation can be moved near the source of data, which can solve issues of latency, bandwidth, privacy and security as well. The objective of this paper is to provide an overview of edge computing, technological implementations, advantages, use cases, challenges and further research problems.**

***Index Terms*—Edge computing, Post-Cloud Era, Decentralized Computing**

I. Introduction

With the evolution of internet and web technologies, the number of devices connected to the internet is increasing rapidly, which is collectively generating huge amounts of data. For instance, Cisco study [1] suggests that 50 billion devices are estimated to connect to the internet by 2020. Traditionally, the majority of devices behaved only as data consumers, however recently they have become data producers as well. This has increased data processing, storage requirements and the load on the available networks.

The cloud computing technology has been a success, but despite its various advantages such as scalability, cost-saving and resource pooling, it alone is not able to handle all needs. The limited network bandwidth has been a bottleneck for cloud computing, though recent development in the network area, like 5G, is aiming at increasing the speed of data transmission. To keep up with the needs of billions of devices, complementary technology needs to be designed.

Devices can have varied needs such as short response time, low bandwidth usage, private data and security [2]. Also, some devices produce huge amounts of data continuously. For instance, an autonomous car generates large amounts of data through sensors and cameras. Relying on the cloud to get a response to decide in real-time is not a feasible idea because transmitting the data to the cloud for analysis would involve time latency.

Traditional computing paradigms cannot serve all of these requirements because existing approaches involve moving data back and forth, which adds latency in the response time. Moreover, cloud servers communicate only with devices using Internet Protocol (IP), not various other protocols used by IoT devices. To resolve challenges with existing cloud computing technologies and to enable Internet-of-Things (IoT), certainly, there is a need of new computing paradigm, which can be considered as a push from cloud computing and a pull from IoT [3].

The OpenFog Consortium[1] proposed fog computing paradigm as infrastructure, in which some data needs are handled by devices at the network edge and some by a cloud-based data centre [1]. Edge computing can be defined as a paradigm which involves data computation away from centralised infrastructure and closer to the logical edge of networks, i.e. towards individual data sources. The edge computing proposes data processing to happen partially at the network edge rather than completely in the cloud for downstream data on behalf of cloud services and upstream data on behalf of IoT services, to enable computing locally to data source [2, 3].

The "edge" can be considered as any computing and/or network resource on the path between data sources and cloud data centres. Though fog and edge computing can be referred to as similar technologies, however, the main difference is that edge computing is focused more on the side of the things, while fog computing focuses more on the infrastructure side [3]. In simple terms, resources near the cloud are fog nodes while that near the end-devices are edge nodes.

However, it is essential to highlight that edge computing or fog computing is not a substitute for cloud computing. Rather, it extends the cloud closer to the edge of the network, where huge amounts of data are generated from

[1] http://openfogconsortium.org/

devices. The most time-sensitive data is analyzed on the edge node or nearest fog node. Data that can wait a few minutes is passed to nearest fog node for analysis. Data that is less time-sensitive can be sent to the cloud for historical analysis, big data analytics, and long-term storage. [1]

As the goal of this paper is to explore edge computing paradigm, technical implementations, benefits, challenges. Having discussed the background context in previous section, the rest of the paper has been organized as follows. The state-of-the-art, benefits, technological implementations have been discussed in section II. Few case studies from healthcare and transportation domain have been described in III. Section IV discusses various challenges, existing issues and future research areas. Section V concludes by providing a summary of this novel computing paradigm.

II. State of the Art

To better understand the concept of edge computing and the building blocks of it, we are going to present the advantages followed by main computing paradigms that implement this idea. The basic idea behind edge computing is bringing cloud computational resources and services in the proximity of the user [4]. The three paradigms are fog computing, cloudlet and mobile edge computing. To further demonstrate their use in real-life applications, we will present cutting-edge frameworks in which such paradigms can be found.

A. Advantages

The computation near the data source (on edge) will reduce data transmission, which will offload traffic from the core network to help reducing latency. This can be considered as extending the capabilities of the cloud closer to the devices. This can help to analyze the time-sensitive data at the network edge itself and to send selective essential data to the cloud for processing and storage [1]. As data processing happens near the source, there is less transfer of sensitive information between devices and the cloud, hence it can enhance security. Also, this will help better data management for the ever-increasing number of devices on the internet.

Using these technologies, response time in face recognition application was reduced from 900 to 169 ms by moving computation from cloud to the edge [5]. Using cloudlets to offload computing tasks for wearable cognitive assistance resulted in improvement of response time between 80 and 200ms. The energy consumption could also be reduced by 30% to 40% by cloudlet offloading [6].

B. Technological Implementation

1) Multi-access Edge Computing:

Multi-access Edge Computing's goal is mainly to provide additional computing power to low computing power devices, such as mobile phones or IoT devices. To achieve that goal, MEC proposes to locate servers in the base stations, or the radio network controller. This way, the servers will provide additional computing power to these devices, with low latency, because they are located near the devices. The architecture of MEC is apps and host-based, providing a Platform-as-a-Service like architecture. Apps/host distribution is handled by a Mobile Edge Orchestrator. The apps have a high context-awareness, which means that they can factor-in some data that base stations/radio network controllers provide, such as network-related information, and the load and capacity of the network [7].

2) Fog computing:

Fog computing is more generic than MEC. It gives the ability to compute on more devices than classic computing. As such, it uses the computing capacity of routers, switches, and generally, network-based devices. Another difference with MEC is that it enables the Edge nodes to communicate with each other, whereas MEC acts like a standalone server. That permits the nodes to be stateful and to contain information that can be requested by other nodes. As such, when a client requests some information to a node, it can be located on another node, and the information will have to be requested to another node: the connection can be not direct, and there can be multiple “hops”[8]. Fog computing needs a supervisor to work, which will provide “fog abstraction”.

3) Cloudlet:

Cloudlet is the closest of the three implementations to classic cloud architectures. Its goal is to mimic cloud architectures, and improving it using Edge Computing principles: instead of having one centralized cloud data centre, which can be very far to the client, it makes the choice of having multiple, smaller data centres. Its architecture is Virtual Machine based, which means it is closer to an Infrastructure-as-a-Service architecture. Also, as it's IP-based, the devices directly know to which instance to connect to. That implicates that the connection between a client and a client does not need more than one hop. That also means that any device can connect to cloudlets, not only specific devices, like in MEC[8].

C. Frameworks

Although this may seem like a thing from the future, steps are taken today in that direction. A good example is NVIDIA Clara [9], which is a healthcare application framework for AI-powered imaging and genomics, proving that healthcare is becoming more and more digitized. The framework can be depicted in Fig. 1. This technology allows for easier integration in clinical workflows and brings compute capabilities to edge medical devices.

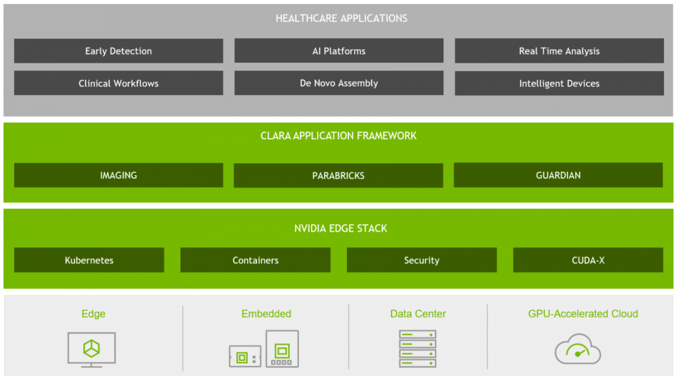

Fig. 1. NVIDIA Clara application stack [9]

III. Case Study

In this section, two major case studies are presented to highlight the need for edge computing and the huge benefits that it can be derived by using this paradigm.

A. Healthcare

As we are writing this report, we are experiencing one of the worst flu outbreaks that happened in modern times, namely the COVID-19 virus. This flu managed to subdue even the most prepared and advanced healthcare systems in the world, proving that there is still a long way to go in modernising the health sector. A promising solution for such a scenario is the connected health, implemented by a collaborative edge. A visual representation of such a system can be depicted in Fig.

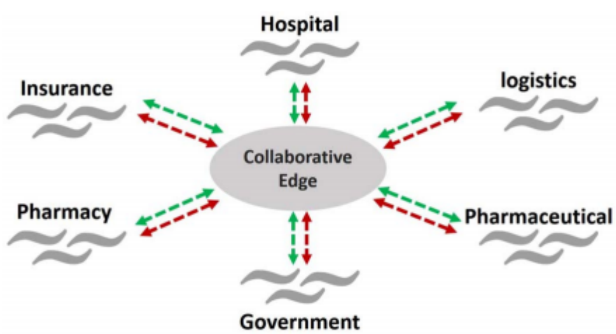

Fig. 2. Collaborative edge in healthcare context [3]

In the case of COVID-19, one of the biggest problems encountered was the lack of tracing and prediction. Tracing the infected people, the ones that crossed borders, the ones that were in contact with infected people, the recovered cases and so on, would have offered valuable information that could help in stopping the spread of the virus and better predict where the hotspots of infections would be. For such a use case, a central processing unit would be simply out of the discussion, since besides the vast amount of information that would be overwhelming, another critical aspect is the security of personal data and national-level data. Because of this, a collaborative edge is the desired solution.

In this case, the hospital acts as a major information collector about the status of the patients. But what happens when patients leave the hospital? Or in the critical time before arriving at the hospital? If we think of people as a swarm of mobile devices, then it is easy to leverage the idea of distributed architecture, which leads to edge computing. Mobile devices could exchange information between them and perform the needed computation locally *e.g. Checking the surrounding devices to see if another person in proximity was infected or travelled in a high-risk area*. Having this information available, it can also be used in identifying potential hotspots of virus infections or even better, predicting it.

Similarly, pharmacies can also be involved, by accessing the collaborative edge services of nearby hospitals and local authorities, they could get insight into what medicines should they order and by applying the same procedure, pharmaceutical companies can get a prediction of what medicines are in demand and where the production should be accelerated. This also affects the logistics, so transport companies could also benefit from an exchange of information with the pharmaceutical companies. So instead of having a large, central database that stores all the information and process all the information individually, a far better option is to have the computation at the edge of this whole network, in a distributed fashion as much as possible.

B. Edge computing devices for transportation in smart cities

Another big trend in the modern world is the fast digitalisation of the whole transportation system and infrastructure. The automotive industry is striving in ensuring a high level of safety as well as autonomy by not only focusing solely on the vehicles, but also leveraging the connection with the static infrastructure. If we think about the vehicular network, we can find static access points, i.e. roadside units (RSUs) that can communicate with the mobile agents (cars, trucks, etc.) and then further on with datacenters. In this case, the edge computing is realized between the cars and RSUs to ensure low latency and fast response for critical use cases and then cloud computing paradigm is used for communicating less critical data to a datacenter. Such an architecture, can be depicted in Fig. 3. Although the architecture may imply that we have solved the problem of connectivity for transportation systems in smart cities, there are in fact some problems that need to be addressed.

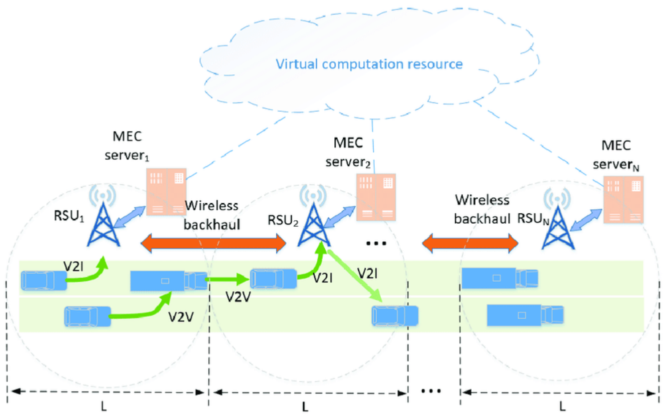

Fig. 3. Edge computing for vehicular applications [10]

In [10], the authors try to solve the problem of efficiently deploying edge computing devices in an urban scenario, depending on the imposed constraints. One of these constraints is the continuous wireless connectivity between the car and the RSUs in densely populated areas. Another one is the latency and computational constraint of such applications, that would require detailed network planning. The authors formulated the optimization problem as a mixed integer linear programming formulation to minimize the deployment cost of the edge devices by considering at the same time the network coverage and the computational requirements. To prove their heuristic, they ran their experiments on the city of Dublin, obtaining the map of buildings from OpenStreetMap [2]. In their evaluation, they have used three metrics:

- message loss: number of sent messages that have not reached the recipient
- exceeded CPU capacity: percentage of messages that the RSUs could not process because of CPU unavailability
- deployment cost: number of RSUs placed in a target area.

By analyzing their results, it is clear that edge computing is a great new paradigm in modern applications, but which also brings interesting research problems that need to be addressed.

IV. Challenges and Further Improvements

Although Edge Computing seems to be a promising paradigms, it still has many challenges that need to be addressed such as programmability, naming, data abstraction, service management, provisioning and resource management, privacy and security. One of the major challenge and open research area is the Quality of Services (QoS) i.e. connectivity, reliability, network bandwidth storage capacity. The programmability remains a major challenge due to the heterogeneous nature of devices involved. Another challenge is naming. No efficient naming mechanism has been built or standardized yet [3]. Named data networking and MobilityFirst are new naming conventions being developed, but they are not yet tested enough. Service management, specifically differentiation, extensibility, isolation, and reliability of devices and their applications are also hot topics for which the research community is expected to bring novel ideas.

Another existing issue is when a user moves from one edge node to another, how should we make various edges collaborate in a synchronized manner. Potential solutions such as selective synchronization to cache the data to all edges likely to be used by the user [3]. To optimize this collaboration with minimum data exchange is one of the future research problems to be solved. Distribution and management of MEC resources, offloading decision, resources allocation, mobility management includes multiple directions of current and future research work for the realization of this computing paradigm. [7]

[2]http://www.openstreetmap.org

V. Conclusion

We have explored different edge computing implementations, analysed them, and their use cases. Fog Computing, Cloudlet, and MEC are different implementations of this principle. They bring computation power at the edge of the network, closer to the client devices, and permit offload computation. However, these technologies are still new, and more advances have yet to be made, to refine it and prepare it for real-world application. As a summary, Edge Computing provides computation, storage, and networking services between end devices and traditional cloud servers. This may cause a paradigm shift in the computing world from centralization (cloud computing) to decentralization (edge computing). However, as already stated, edge computing is not a replacement for cloud computing but rather a necessary complement to it. The implementation of edge computing for application use case depends on the time-sensitivity and permissible latency for response times.

References

[1] Fog Computing et al. Fog computing and the internet of things: Extend the cloud to where the things are. In *Technical Report*. Cisco Systems, 2016.

[2] Weisong Shi and Schahram Dustdar. The promise of edge computing. *Computer*, 49(5):78–81, 2016.

[3] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5):637–646, 2016.

[4] E. Ahmed, A. Ahmed, I. Yaqoob, J. Shuja, A. Gani, M. Imran, and M. Shoaib. Bringing computation closer toward the user network: Is edge computing the solution? *IEEE Communications Magazine*, 55(11):138–144, 2017.

[5] Shanhe Yi, Zijiang Hao, Zhengrui Qin, and Qun Li. Fog computing: Platform and applications. In *2015*

Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb), pages 73–78. IEEE, 2015.

[6] Kiryong Ha, Zhuo Chen, Wenlu Hu, Wolfgang Richter, Padmanabhan Pillai, and Mahadev Satyanarayanan. Towards wearable cognitive assistance. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, pages 68–81, 2014.

[7] Pavel Mach and Zdenek Becvar. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials*, 19(3):1628–1656, 2017.

[8] Koustabh Dolui and Soumya Kanti Datta. Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing. In *2017 Global Internet of Things Summit (GIoTS)*, pages 1–6. IEEE, 2017.

[9] NVIDIA. An application framework optimized for healthcare and life sciences developers. https://developer.nvidia.com/clara, 2020.

[10] Longjiang Li, Hongmei Zhou, Shawn Xiong, Jianjun Yang, and Yuming Mao. Compound model of task arrivals and load-aware offloading for vehicular mobile edge computing networks. *IEEE Access*, PP:1–1, 02 2019.