Edge and Dew Computing for Streaming IoT

Marjan Gusev
University Ss Cyril and Methodius
Skopje, Macedonia
Email: marjan.gushev@finki.ukim.mk

***Abstract*—The main question in the case of streaming data coming from end-user IoT devices with big quantities is where to process data. The available choices are whether the processing of the required information should take place on a local device or to offload data to a nearby or remote server for further processing. Basic IoT schemes include only local processing, while more sophisticated schemes include offloading to nearby servers on the edge of the network, or to remote distant cloud servers. In this paper, we analyze the implementation details and organizational approaches related to dew computing, where the processing is brought even closer to the user than the edge computing concept. The relevant features will be compared to classical edge approaches, such as cloudlets, fog computing, mobile edge computing or similar computer architecture approaches.**

***Index Terms*—Mobile Cloud Computing, Cloudlet, Edge computing, Fog computing, computation offload**

I. Introduction

Computing devices are embedded in almost all end-user devices used in everyday activities. With the growth of the Internet and the advances of modern technology, users can control and use these devices over the Internet. This is the basis of the Internet of Things (IoT) as an organized interconnection of all these devices [1], as independent computing devices that function in a shared environment over the Internet.

In this research, we address issues that arise with IoT devices that generate data with high volumes and velocity, characterizing them in the Big Data concept as they need computing units that provide fast processing and massive storage capacities. In addition, another problem arises when the user tries to use them as independent battery-operated mobile devices with a wireless connection. It requires special designs, so the IoT devices will be relieved of all tasks that consume a lot of energy.

An example of such a device is a wearable eHealth or ECG sensor. It is a small device that can be patched on a user's chest. To make it more comfortable, it needs a very small weight and small size, such that will not cause any obstacles for user daily activities and movements. Therefore, the designers of such a device face the constraint of using a very small battery that should be recharged on a couple of days, for example, a week. The device needs to process a lot of data generated as a 2-byte integer samples on a regular sampling frequency higher than 250 Hz, which will generate a data stream with a rate of 30 KB per minute, and storage demands of 1.8 MB per hour or 54 MB per day. Essential data processing and diagnosis may require up to 500 executable commands per sample, so the processing needs a processing power of at least 125.000 operations per second, excluding the operations required by the operating system. Although it may not look so demanding for a modern computer, still it will spend a lot of energy on a smaller embedded sensor, and will not fit in the constraint for a small battery.

Offloading is a promising alternative, but still, the users are concerned when to offload and where to offload. This paper analyzes several different approaches and architectural designs. A comprehensive comparison is provided to discuss all relevant issues and help a solution provider how to organize the computing, storage in order to minimize energy consumption of the end-user IoT device without degrading the performances of the application.

The main concepts of dew and edge computing are compared to distinguish between different designs and approaches. In this paper, we will explain what is the difference between edge and dew computing, and answer where and when to offload. We will present differences in design and implementation, addressing the application domains.

The paper organization is as follows. Section II gives a state-of-the-art and related work on edge and dew computing architectural concepts. Our view and distinction between different architectural approaches are explained in Section III followed by a discussion in Section IV. Finally, relevant conclusions and future work directions are given in Section V.

II. Related Work

Streaming IoT solutions belong to a wider class of ubiquitous and pervasive computing solutions for IoT devices [2]. A streaming IoT device is considered to be a device that generates at least 100 samples per seconds [3], [4].

The first idea to offload data and computations initiates a cloud server connection to a mobile device. The mobile device is considered to be the end-user IoT device and the cloud server is the computing unit that will process streaming data.

Dinh et al. [5] discuss the advantages of dynamic provisioning, scalability, multitenancy, and ease of integration for related mobile cloud computing applications. Issues that need to be addressed in the mobile cloud computing include low bandwidth, availability, heterogeneity, static and dynamic environments in computation offloading, security, privacy and other quality of service and related open issues.

In addition, the presented architecture does not address wearable mobile IoT devices with limited power supply capabilities and small computing capacities. This is why the edge computing is introduced as an architecture solution [6].

A. Edge Computing

The focus of architectures and computer implementations has shifted towards gaining real-time responses along with support for context-awareness and mobility in the IoT [7], enabled by edge computing.

The edge computing technology promises to deliver highly responsive cloud services for mobile computing, scalability and privacy-policy enforcement for the IoT, and the ability to mask transient cloud outages. Satyanarayanan [8] elaborates that the idea of caching is used in edge computing for caching the cloud services.

Edge computing pushes the cloud services closer to the user and also pulls the IoT micro-services from IoT devices [9]. It changes the vision of data consumer to data producer of an IoT device.

Two approaches dominate the use of edge computing architectural organizations. They differ by the implementation provider [10], [4], so if the mobile operator is providing an infrastructure, then it is a basis of fog computing and if an Internet provider uses LAN networking for the edge devices, then it is a cloudlet solution. Some authors find these terms to be synonyms to edge computing [11].

Satyanarayanan [12] specifies *a cloudlet* as an infrastructure based on a virtual machine located in the proximity of the end-user device accessed in a LAN environment. Verbelen et al. [13] describe that the cloudlets do not have to be fixed infrastructure close to the wireless access point, but can be formed in a dynamic way with any device in the LAN network with available resources.

Cloudlet challenges have been analyzed [14] for their architectural and implementation issues. The corresponding definition clearly identifies another architectural layer between the cloud server and the end-user device.

Bonomi et al. [15] define essential *fog computing* concepts setting servers at the base stations to reduce the latencies and distribute the processing in a number of IoT applications.

Chiang and Zhang [16] analyze the latency requirements and bandwidth constraints in the context of IoT resource-constrained devices, along with other non-functional issues including security and protection.

Some authors use the fog computing concept simply to present the realization of computer communication infrastructure with routers, switches, access points, and gateways, and others as computing nodes at the edge of the mobile network. Therefore, some authors think that the fog concept is equal to the edge computing concept in the context of computing, similar to the concept of servers used in mobile edge computing. Other authors observe the fog computing only as a communication infrastructure.

Another related concept is the *mobile edge computing* aiming at reducing network stress by shifting computational efforts from the Internet to the mobile operator's edge. Although according to the previous understanding of fog computing as an edge computing communication infrastructure, the complete idea of mobile edge computing is an application of the fog computing concept.

An early definition of mobile edge computing can be found in several papers, although some of these definitions in our context are a specification of dew computing. For example, Kim et al. [17] introduce the concept of Mobile Edge Computing Devices as an interface between distributed sensors and the end server in order to reduce processing and bandwidth requirements to the end servers, and provide enhanced scalability, flexibility, reliability, and cost-efficiency.

ETSI [18] tries to standardize it as a key technology towards 5G, to provide an IT service environment and cloud-computing capabilities at the edge of the mobile network and in close proximity to mobile subscribers, aiming at reducing the latency and improving the user experience. In addition, ETSI [18] designs it as a natural development in the evolution of mobile base stations and the convergence of IT and telecommunications networking by using virtualized environments.

Mobile edge computing is able to provide IoT services, which are not technically or economically feasible otherwise. In addition, bringing mobility support functions to the mobile edge platform may have a dramatic impact on the existing architecture.

Mobile edge computing architecture has been analyzed by Beck et al. [19] and a taxonomy is specified according to the following criteria: offloading, local connectivity, content scaling, augmentation, edge content delivery, and aggregation.

Particularly, according to their definition, cloudlets, can also use offloading to a mobile edge computing server, which in our case is a definition of a dew computing layer. The difference in specifying it as a non-cloudlet layer lies in our definition that a cloudlet server is a representation of an edge computing layer, and it is on the same level to the mobile edge server.

Wang et al. [20] conclude that mobile edge networks provide cloud computing and caching capabilities at the edge of mobile operator networks.

B. Dew Computing

Dew computing has been defined by several research papers [21], [22], [23] as an architecture that brings computing closer to the user. Wang et al. [24] discuss the transition of Internet computing paradigms towards dew computing.

Dew computing concepts are complementary to the edge computing concept. We define dew computing concept for streaming IoT devices as end-user devices that do not have Internet access via LAN network in order to transfer data or offload computations to an edge or cloud server.

The basic definition of dew computing concept [22] specifies two essential features:

- *independence*, by enabling an environment where the IoT device can perform locally and interact with the end-user without the need of a permanent Internet connection, and
- *collaboration*, by enabling an environment where the IoT device can collaborate with other devices via an Internet connection.

Ray [25] discusses that the independence and collaboration features in the Wang definition [22] need an addition of the microservice concept [21] although it is indirectly assumed that a microservice is the key feature to allow independence feature. The independence concept and new formulation of the collaboration feature have been analyzed by Ristov et al. [23], where the information-centric feature is added to the existing two essential features, although, they can be treated as an indirect.

A specification of a dew computing architecture was given by Wang [26] along with an elaboration of functional requirements. It is an extension of a cloud-based client-server architectural concept adapted to a new environment.

A dew server is defined by Wang [26] as a tiny light-weight server that provides microservices [21]. Ray [25] enhances it with a more detailed specification. He discusses three types of novel-services: infrastructure-as-a-dew, software-as-a-dew service, and software-as-a-dew product. In addition, he defines that the dew computing model is composed of six essential characteristics: Rule-based Data Collection, Synchronization, Scalability, Re-origination, Transparency, and Any Time Any How Accessibility.

In this paper we address dew computing application in IoT, especially targeting the streaming IoT devices. An overview of dew computing solution for streaming IoT is presented in [3]. The implementation details address the way the devices connect to each other in various environments. The communication can be established directly from the IoT devices to the cloud server, or to the edge devices via various personal area network or LAN technologies.

III. Architectural Concepts

We start with a basic client-server architecture, where a client (a computing device located in a lower architectural layer) is connected to a server (located in the upper architectural layer) by a communication link. The edge computing concept introduces an intermediate server, called edge server between the cloud server and the client, which in our case is a streaming IoT device.

According to the provider of the communication infrastructure, there are two approaches of the edge computing architecture, the first based on a cloudlet edge server and the second on edge servers mobile operator's network.

A. Cloudlet Edge Computing Architecture

A simple design of adding an edge server between the end-user IoT device (client) and the cloud server is based on a cloudlet. A cloudlet is a smaller server on the edge of Internet network provided by an Internet provider. The cloudlet server collaborates with the end-user IoT device by a LAN technology. Its main function is to provide services to the client, since the end-user streaming IoT device can neither perform complex computations nor store big amounts of data. In addition, it may be a mobile device which is wirelessly connected to the edge server and is battery-operated, so its function is constrained by the capacity of an installed battery.

The cloudlet edge computing concept is presented in Fig. 1. The top layer consists of a cloud server, the lower layer of IoT devices. The intermediate layer specifies the cloudlet edge

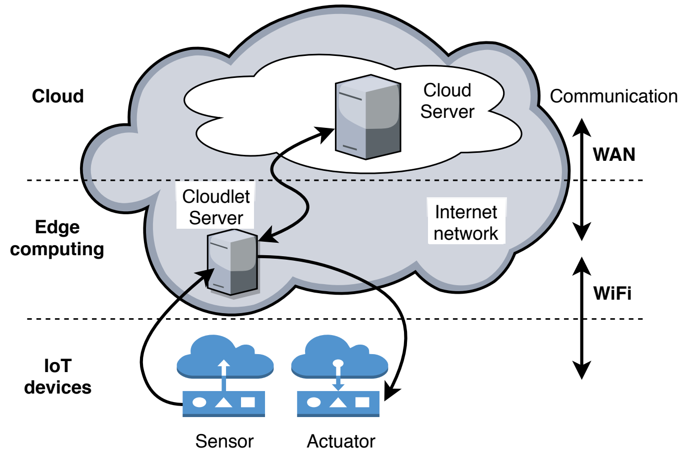

Fig. 1. Cloudlet edge computing architectural approach

server. The communication between the cloudlet and cloud server is based on WAN and between the cloudlet and IoT devices on WiFi LAN networking.

The IoT device offloads data and computations to the nearby cloudlet. It relieves the energy supply demands since complex computations and data storage are transferred to the cloudlet. Since the communication is local via a wireless network, the expected delays are relatively small and are much lower than standard WAN delays used in the case of a remote cloud server.

B. Mobile Edge Computing Architecture

The second approach of building an edge computing solution is based on using an edge server on the mobile operator's network. When analyzed from an architectural design view, this is the same three-layer architecture, where an edge server is added to the client-server architecture. The difference to the cloudlet approach is that the communication infrastructure is provided by the mobile operator. The edge server is located on the edge of the Internet perimeter of the mobile operator, while the IoT devices are using the mobile operator's network.

The presented design differs from the previous since the edge server is not anymore a cloudlet, it is another server owned by the mobile operator and located at the base station on the edge of Internet network. The whole communication between mobile operator's cloud server and the edge server at the base station is via WAN provided by the mobile operator. The end-user IoT device communicates with the edge server via radio waves of the mobile operator's network, such as 3G/4G or 5G.

Fig. 2 presents a mobile edge computing architectural approach. The IoT device can offload data and computations to a more powerful edge server. Since it is located at the base station of the mobile operator's network, the expected transmission delay is very small and the IoT device performance is relatively high.

The problems associated with streaming IoT and enabling their mobility by wireless connection and small size battery-operated function cannot be solved by classical edge com-

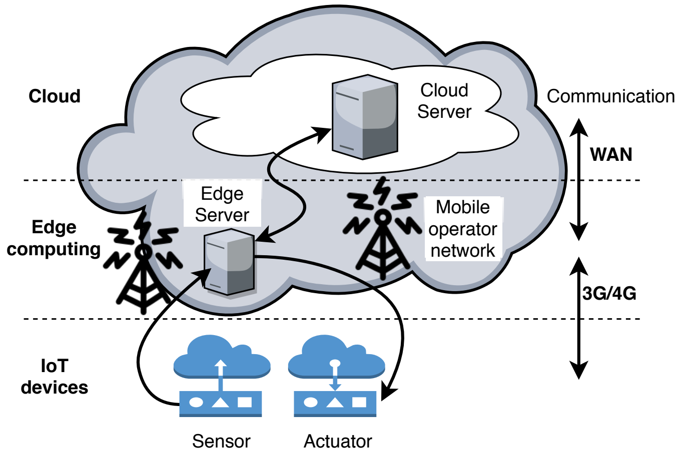

Fig. 2. Mobile edge computing architectural approach

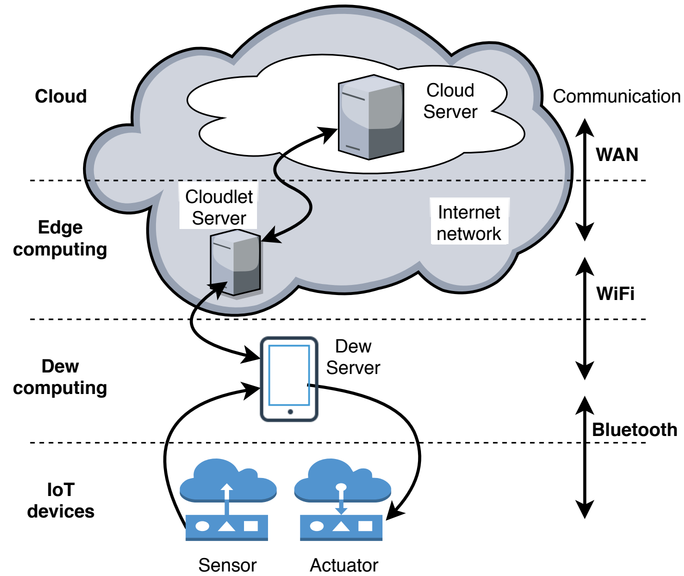

Fig. 3. Cloudlet dew computing architecture

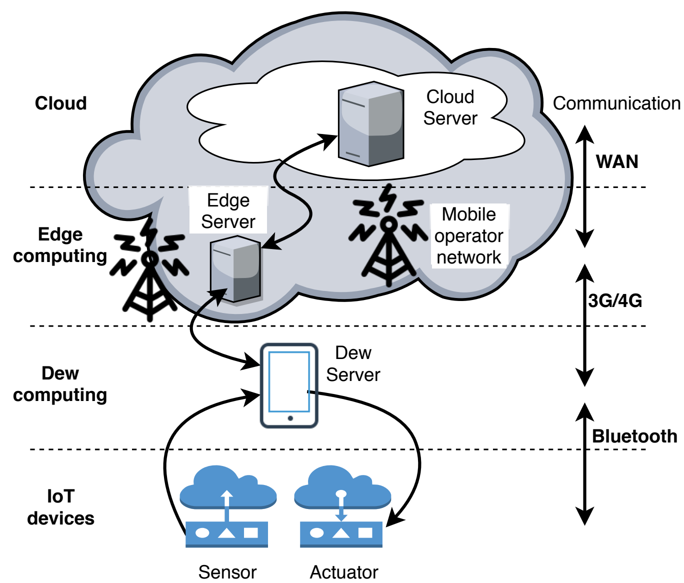

Fig. 4. Dew mobile edge computing architecture

puting solutions. It cannot be solved either using cloudlets nor direct mobile edge computing architecture. The solution is provided by adding a new layer between the edge server and the IoT device. Similar to the previous categorization, two approaches are possible, the first by using a LAN connected cloudlet or an edge server on a mobile operator's network.

C. Dew Computing Cloudlet Architecture

Adding another dew computing layer in the cloudlet architecture is presented in Fig. 3. A streaming IoT device with limited power supply and mobility enabled with personal area networking wireless connectivity can communicate only with a dew server. A dew server is out of Internet perimeter outside the edge of a network and therefore it cannot be treated as an edge device. It communicates to the cloudlet edge server via LAN. The cloudlet server can also communicate with other cloudlet or cloud servers to exchange results and any relevant information.

The IoT device streams data to a nearby dew server via Bluetooth or other personal area network communication. It neither performs any computation nor stores any data. It is a simple realization of a sensor that senses a signal and transfers data to the nearby dew server. The dew server takes the role of essential signal processing and data storing. In addition, it is able to transmit data and complex computations to a cloudlet server, or exchange information about the outside world.

In the context of dew computing, the dew server performs its functions independently and can collaborate with other devices.

D. Dew Mobile Edge Computing Architecture

The dew mobile edge computing architectural design differs from the cloudlet dew computing architecture in the communication of the dew server with the edge server. Instead of a cloudlet on the LAN provided by the Internet provider, the design uses an edge server on the edge of the mobile operator's network. A mobile operator's radio communication is used for communication between the edge server and dew server, instead of LAN used as a communication between the cloudlet edge server and dew server.

Fig. 4 presents the dew mobile edge computing architecture. The IoT device is a light mobile device wirelessly connected with limited battery-operated power supply. It can communicate with a dew server via Bluetooth or any other personal area network communication link. The dew server uses 3G/4G or 5G radio communication link established by the mobile operator. The edge server found on the edge of the mobile operator's Internet network can communicate with the main cloud server to exchange information.

Some readers may argue that this is another edge computing

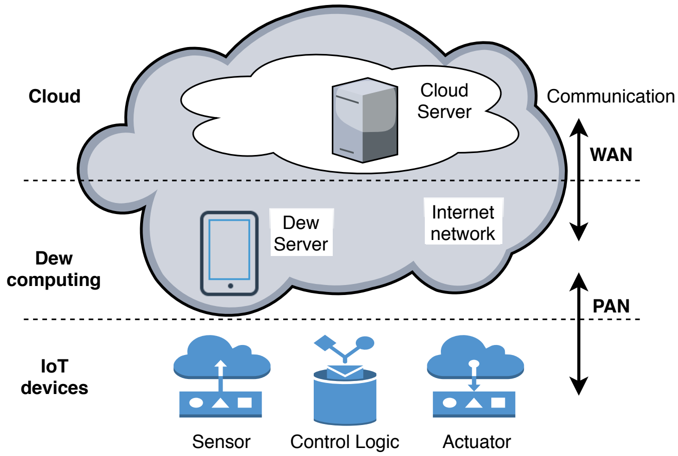

Fig. 5. A reduced dew computing architecture

implementation. Let's specify the main differences. The dew server is out of the Internet network, it uses 3G/4G or 5G to communicate to the edge server and therefore is out of the Internet edge, although, indirectly it is connected to the Internet. In addition, the IoT device uses a personal area network to connect to the dew server, instead of the mobile operator's radio network.

The analyzed issues make a clear distinction between the two identified edge computing approaches and these two dew computing approaches. Using another dew computing layer between the edge server and the IoT device results in enabling an environment for light mobile streaming IoT devices that spend only a small portion of energy for a low power personal area network communication link.

E. Dew Computing Cloud Architecture

To be consistent we will provide another dew computing architecture which does not belong to the previous edge computing architectures (cloudlet and mobile edge computing). It is based on direct communication between the dew server and the cloud, as presented in Fig. 5. A streaming IoT device is a light mobile IoT device wirelessly connected to the dew server, which is capable to establish a WAN connection to the cloud server.

There is a great similarity between the direct dew cloud architectural approach and the cloudlet edge computing architecture. The difference is in the communication between the IoT device and the dew or edge server. A dew server is capable to accept personal area network communication, while the edge device only WiFi LAN network. Personal area communications are required by the streaming IoT device to save the energy consumption. Since it is a small mobile device and it is intended to spend only a small portion of energy, it cannot support WiFi connection, but only low energy Bluetooth or similar local radio connection.

It also explains the similarity to the mobile edge computing architecture, since it is using 3G/4G or 5G mobile operator's communication link instead of low energy Bluetooth or similar local radio connection.

IV. Discussion

Different analyzed architectural approaches are compared to implement an effective solution for streaming IoT devices that demand mobility, low power local wireless communication and battery-operated devices that spend only small energy for its performance.

A. Edge vs Dew computing approaches

According to Wang [26], a dew server is added on a path between the client and cloud server. In addition, it can work independently and collaborate with others.

However, the idea of adding a server between the client and the cloud server fits more to the edge computing concept. By definition, edge computing brings the computing to the edge of the network, so a new edge server is located next to the client, that is the streaming IoT device in our case.

Comparing these two approaches, dew computing is an extension of the edge computing concept, not the client-server concept, as elaborated in the previous Section.

The evolution of edge computing concepts and their implementation in IoT has been discussed by Gusev and Dustdar [4]. Two approaches are defined in the case processing is realized on the edge of the network, provided either on the mobile operator's network or on the LAN provided by an Internet provider. The dew concept is defined by bringing the processing even closer to the IoT device than the edge computing concept. The end-user IoT device is not on the edge of the network but will communicate to an edge device that will provide a connection to the Internet and all relevant functionalities.

Zhou et al. [27] define post-cloud computing paradigms to include fog computing, mobile edge computing, and dew computing. According to their definition, fog computing is a horizontal architecture for a virtualized platform that provides computation, storage, and services between end devices and cloud servers, which slightly differs to our understanding that fog computing refers only to a communication infrastructure environment. Further on they define mobile edge computing as an architecture offered at the edge of a mobile network. Although these two items differ in their definition, we can conclude that they have specified fog computing as a specialized virtualized environment and mobile edge computing as an environment provided on the edge of the mobile operator's network. In our definition, both are considered as part of edge computing, and fog is a synonym to edge, or its special case when virtualized environments are used.

As a special form of a post-cloud computing paradigm [27], dew computing is specified as a software organization model where local computers provide rich functionality independent of cloud services. So this fits into our definition that they are not on the edge of the network, and these devices can communicate to edge devices and access edge or cloud servers to exchange information or even offload. data and computing.

Another interesting feature is the communication access to end and intermediate devices. Some authors [7] define fog computing as an infrastructure where all communication mechanisms are supported, including PAN (Bluetooth), LAN (WiFi) and mobile operator networks (3G/4G). Although most of them set an equivalency between the fog computing and edge computing concepts, the main difference to the mobile edge computing as they only use mobile networks (3G/4G), while cloudlet implementations use only LAN (WiFi) connection. We do not agree to this classification and define the dew computing layer, which communicates to the IoT layer by PAN (Bluetooth), and to the above layer via LAN (WiFi) in the cloudlet implementation or via mobile operator network (3G/4G) in the mobile edge computing concept.

According to the classification presented by Dolui and Datta [7], the context awareness in fog computing is medium, and in mobile-edge computing high, while the cloudlet context awareness is low. We do not agree to this classification and add that a dew-computing cloudlet solution may also have high context awareness.

Zhou et al. [27] discuss that edge computing is just another computing paradigm advocated by the academic community [28] with a wider description and broader meaning than fog computing. Actually, edge computing is based on fog computing and refers to and is included in the definition and categories of mobile edge computing, as a general term that covers both fog computing and mobile edge computing. However, there is no broad consensus on the concept of edge computing.

B. *Dew computing challenges for IoT streaming devices*

According to several other definitions of dew computing, the dew devices does not have permanent LAN connection and Internet access, but can have occasionally. However, in our specification of dew computing solution for streaming IoT devices, these devices can collaborate with edge devices via personal area network and, therefore, access the wider world via LAN network and Internet connectivity.

Analyzing the basic definition of dew computing devices with features to perform independently and collaborate with other devices, the edge computing concept provides only the second feature, since it must have an Internet or other connection to the server where the processing of offloaded computation is performed and where the massively collected data is stored. However, in our case, a dew device is a streaming IoT device without direct connection to the server and can perform regularly without communicating a remote edge or cloud server. Connecting to an edge device via personal area network technologies will enable indirect Internet availability.

Note that although a streaming IoT device can process data locally, still it may not perform all required functions that need an exchange of information with the wider world. For example, if the streaming IoT device is not connected to an edge device, it may lose its data and fail to deliver results to the outside world.

Difference between the classical implementation of a mobile-edge computing solution and its dew computing implementation is in the way the IoT devices are used. The conventional approach means that the smartphone or any other mobile device is at the same time an end-user device and IoT device. However, the dew computing implementation introduces two layers, instead of one, the first is the IoT devices layer, and the second is the dew computing layer with smartphones or other mobile devices.

Analyzing the edge and dew computing architecture, we can conclude that they change the centralized approaches to distributed decentralized environment. In addition, Nastic et al. [29] define a serverless real-time data analytics platform where the (micro) services provided by edge devices, edge servers or, in our case, dew servers, can be transferred (offloaded) to another device found in the nearby proximity and ready to accept its function.

The main challenges in dew computing architectures that support streaming IoT devices are:

- autonomous functioning, by communication to a local dew server
- lower latency and high bandwidth communication, and
- minimal energy consumption, by minimizing the number of operations, data storage and data transfer.

V. Conclusion

To bring the processing closer to the end-user streaming IoT devices we can use the concept of edge computing. The edge in this context has LAN connectivity, and Internet access provided either by a mobile operator or by an Internet provider.

We have analyzed a typical case when end-user streaming IoT devices are constrained by mobility, wireless connectivity, and limited battery-operated power supply. These features prevent using the edge concept directly, so another architectural concept needs to be designed to cope with these issues. The presented solution belongs to the dew computing concept.

Dew devices are the end-user streaming IoT devices that can perform required activities independently and collaborate with neighboring edge devices to access a LAN network and Internet connection. We have presented architectural design details on these dew concept implementations in both cases where the edge is provided by a mobile operator or Internet provider.

The main difference between the edge and dew computing concepts are in the definition of the location of the end-user device. If it is on the edge of the Internet network, then it is treated as an edge computing architecture, while if it is outside of the Internet perimeter edge, then it is a dew computing architecture. Both architectural styles aim at bringing the computing closer to the user, and we can say that dew computing brings it even closer to the user. This fits in the definition of going back to the roots that are to the end-user devices.

In the case the end-user device demands mobility, low energy local radio wireless connection performing as small processing tasks as it is possible to save the battery-operated power supply, then the answer is in the use of a dew computing architecture. The dew server design can be also constrained by low energy consumption to save the energy and may use LAN or mobile operator radio communication links, determining if the solution will use cloudlet or mobile edge networking.

REFERENCES

[1] O. Vermesan, P. Friess, P. Guillemin, S. Gusmeroli, H. Sundmaeker, A. Bassi, I. S. Jubert, M. Mazura, M. Harrison, M. Eisenhauer *et al.*, "Internet of things strategic research roadmap," *Internet of Things-Global Technological and Societal Trends*, vol. 1, no. 2011, pp. 9–52, 2011.

[2] M. Satyanarayanan, "Pervasive computing: Vision and challenges," *IEEE Personal communications*, vol. 8, no. 4, pp. 10–17, 2001.

[3] M. Gusev, "A dew computing solution for IoT streaming devices," in *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2017 40th International Convention on*. IEEE, 2017, pp. 387–392.

[4] M. Gusev and S. Dustdar, "Going back to the roots: The evolution of edge computing, an IoT perspective," *IEEE Internet Computing*, vol. 22, no. 2, pp. 5–15, 2018.

[5] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[6] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future generation computer systems*, vol. 29, no. 1, pp. 84–106, 2013.

[7] K. Dolui and S. K. Datta, "Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing," in *Global Internet of Things Summit (GIoTS), 2017*. IEEE, 2017, pp. 1–6.

[8] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.

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

[10] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," in *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*. IEEE, 2014, pp. 1–8.

[11] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[12] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE pervasive Computing*, vol. 8, no. 4, 2009.

[13] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *Proceedings of the third ACM workshop on Mobile cloud computing and services*. ACM, 2012, pp. 29–36.

[14] A. Bahtovski and M. Gusev, "Cloudlet challenges," *Proceedia Engineering*, vol. 69, pp. 704–711, 2014.

[15] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM, 2012, pp. 13–16.

[16] M. Chiang and T. Zhang, "Fog and iot: An overview of research opportunities," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 854–864, 2016.

[17] S. J. Kim, G. Deng, S. K. Gupta, and M. Murphy-Hoye, "Enhancing cargo container security during transportation: A mesh networking based approach," in *Technologies for Homeland Security, 2008 IEEE Conference on*. IEEE, 2008, pp. 90–95.

[18] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computinga key technology towards 5g," *ETSI white paper*, vol. 11, no. 11, pp. 1–16, 2015.

[19] M. T. Beck, M. Werner, S. Feld, and S. Schimper, "Mobile edge computing: A taxonomy," in *Proc. of the Sixth International Conference on Advances in Future Internet*. Citeseer, 2014, pp. 48–55.

[20] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.

[21] K. Skala, D. Davidovic, E. Afgan, I. Sovic, and Z. Sojat, "Scalable distributed computing hierarchy: Cloud, fog and dew computing," *Open Journal of Cloud Computing (OJCC)*, vol. 2, no. 1, pp. 16–24, 2015.

[22] Y. Wang, "Definition and categorization of dew computing," *Open Journal of Cloud Computing (OJCC)*, vol. 3, no. 1, pp. 1–7, 2016.

[23] S. Ristov, K. Cvetkov, and M. Gusev, "Implementation of a horizontal scalable balancer for dew computing services," *Scalable Computing: Practice and Experience*, vol. 17, no. 2, pp. 79–90, 2016.

[24] Y. Wang, K. Skala, A. Rindos, M. Gusev, S. Yang, and Y. PAN, "Dew computing and transition of Internet computing paradigms," *ZTE COMMUNICATIONS*, vol. 15, no. 4, 2017.

[25] P. P. Ray, "An introduction to dew computing: Definition, concept and implications," *IEEE Access*, vol. 6, pp. 723–737, 2018.

[26] Y. Wang, "Cloud-dew architecture," *International Journal of Cloud Computing*, vol. 4, no. 3, pp. 199–210, 2015.

[27] Y. Zhou, D. Zhang, and N. Xiong, "Post-cloud computing paradigms: a survey and comparison," *Tsinghua Science and Technology*, vol. 22, no. 6, pp. 714–732, 2017.

[28] Open Edge Computing, "Open edge computing initiative," 2017. [Online]. Available: http://openedgecomputing.org/index.html

[29] S. Nastic, T. Rausch, O. Scekic, S. Dustdar, M. Gusev, B. Koteska, M. Kostoska, B. Jakimovski, S. Ristov, and R. Prodan, "A serverless real-time data analytics platform for edge computing," *IEEE Internet Computing*, vol. 21, no. 4, pp. 64–71, 2017.