Mobile Edge Computing: Requirements for Powerful Mobile Near Real-Time Applications

Harald Frank
IQDoQ
Document Management Company of MATERNA
Bad Vilbel, Germany
harald.frank@iqdoq.de

Woldemar Fuhrmann
Department of Computer Science
University of Applied Sciences Darmstadt
Darmstadt, Germany
woldemar.fuhrmann@h-da.de

Bogdan Ghita
Centre for Security, Communications and Network Research
University of Plymouth
Plymouth, United Kingdom
bogdan.ghita@plymouth.ac.uk

***Abstract*— The progressive distribution of the components in Mobile Edge Computing at different locations, need to be controlled from a centralized logical unit. This paper proposes a new controlling unit between the mobile transport network and the cloud infrastructure, called Mobile Edge Computing Manager, which reuse the concept of the Black Rider, which is context- and policy-based itself. This centralized processing unit bundled informations from the mobile transport network as well as from the cloud-infrastructure. The paper provides a first architectural overview.**

Keywords—component; Mobile Edge Computing, Mobile Cloud Computing, Black Rider, policy

I. Introduction

The Vision of Mark Weiser in 1991 [1] become slowly reality. Ubiquitous and pervasive computing is ready to be the basis of the internet of things in the near future.

Mobile Edge Computing (MEC) enables powerful application on mobile devices in near real-time quality. This was reached by the extension of the original Mobile Cloud Computing (MCC) concept via the introduction of an additional infrastructure component, the so called "Cloudlet". The cloudlet is an additional small data center between the mobile device and the centralized data center. This small data center is usually located in the near of the cell-tower of the radio network. The proposal of the cloudlet is reducing the network latencies between the application on the mobile device and its counterpart in the centralized data center, which is usually far away. The cloudlet consist of an amount of virtual machines, which provide computing resources close to the mobile device.

This is particularly necessary for the envisaged new 5th generation of mobile networks. The planned high data transfer rates, low internal latency and high movement speeds of mobile devices require an appropriate infrastructure on the cloud side, to be able to use it optimally.

MCC and MEC have basically the same objectives, such as the following:

- Extension of the operation time of the mobile terminal with limited battery capacity [2].
- Outsourcing of compute-intensive applications from the mobile device to the cloud, in order to enable applications that are not possible on the mobile device, e.g. real-time applications with high workload.

MEC has the following additional objectives:

- Increase of the quality of experience (QoE) by reducing the response time of the application for the user.
- Reduction of traffic in the packet-core network.

Both MCC and MEC consist of a mobile and a wired infrastructure. The advantages of MEC are only effective, if the application in the cloudlet stays in the close neighborhood of the mobile device over the entire runtime of the applications on the mobile device. Therefore we need to classify the relationship between applications.

On the one hand we have fixed applications, where the server-side application stays at a fixed location in the cloud, and on the other hand we have mobile applications, where the server-side application can follow the mobile device within the cloudlet infrastructure:

- Case 1: Wireless mobile user application – fixed cloud/infrastructure application
- Case 2: Wireless mobile user application - mobile cloud/infrastructure application

There are a variety of proposals and detailed solutions on parts of the issues of MEC. However, an approach is missing that enables the effective control of the combined mobility of applications on the device and on the cloudlet side. In order to minimise the length of the transport path in the fixed network infrastructure the applications have to follow the mobile devices.

Requirements have to be formulated for the control of the mobility of applications taking into account the specific roles of mobile devices, the network providers, the cloudlet providers, and the cloud providers. Furthermore the provision of infrastructure elements from different vendors makes open interfaces necessary. In this context a working group was formed by the ETSI on Mobile Edge Computing [3]. The standardization efforts of ETSI have the goal to avoid proprietary solutions wherever possible.

In [4] the functional entity 'Black Rider' (BR) had been introduced which is able to gather static and dynamic user specific data in a mobile network. These information will be used to improve activities in the mobile network, for example to reduce energy consumption. The BR will be used as an example for the required control unit.

II. Related Work

In earlier days there were mainframes, which were usually placed at a central location. Also virtualization (VSE, VMS, etc.) has already been used, but primarily for separating multiple user environments and executing them in parallel. These mainframe systems had front-end computers, which were partly distributed to connect remote workstations. However, these front-end computers supported only communication functions and did not provide any processing and storage capacity. In these systems the applications on the fixed infrastructure side was static at a fixed location with no need for mobility control.

Nowadays computers are distributed in the cloud and the applications are virtualized. It is not transparent to the user where his applications are stored and processed. With MEC additional mini computing centers (cloudlets) are placed close to the radio interface. This allows the distribution of client- or server-side applications in the immediate vicinity of the mobile user with the objectives to minimize the length of the transmission path and the transmission latencies. Cloudlets support powerful applications with near real-time requirements and help to relieve the battery capacity constraints of the mobile terminals [5]. A typical MEC architecture is shown in fig. 1.

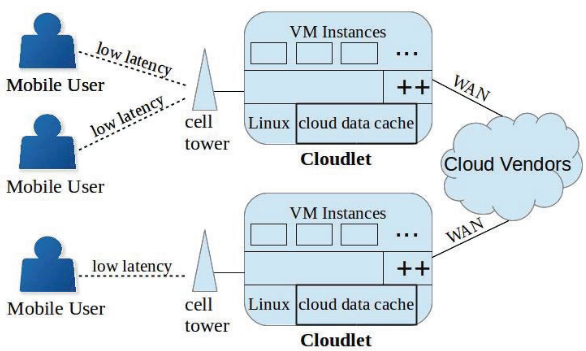

Fig. 1. Typical MEC architecture

Many important building blocks for MEC architectures are already available, for example in the OpenStack extension OpenStack++. Also mechanisms are available for effective moving one VM from one cloudlet to another cloudlet in another location in the network [6].

However, open interfaces between the components of the MEC architecture and the underlying transport networks are not yet defined. Furthermore, the MEC architecture has to support roaming within one provider infrastructure and between different provider infrastructures. Fig. 2 illustrates this situation showing different providers with different cloudlet solutions.

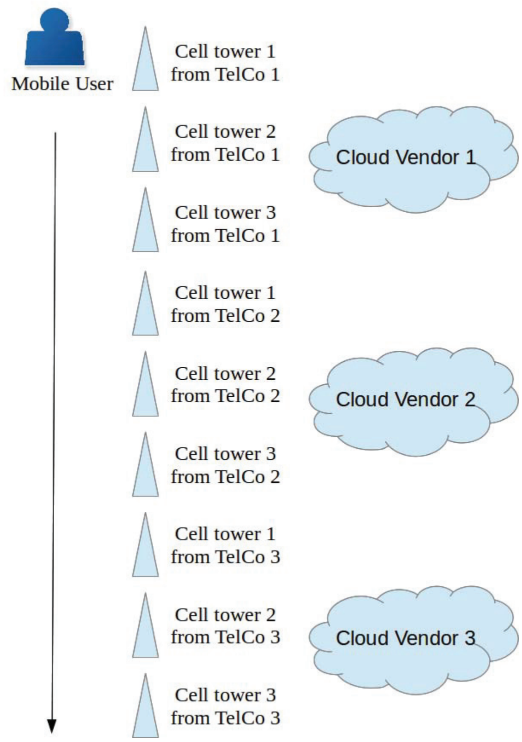

Fig. 2. Different Provider with different Cloudlet Solutions

The BR is a functional entity in a mobile network which is able to gather static and dynamic user specific data, for example related to the mobility management context and bearer management context of the mobile user. These data are stored in a BR Database (BR DB) which complements the centralized database for user data convergence UDR (User Data Repository). The BR is involved in decision making processes of the transport network control, such as handover and traffic offloading, and for that purpose, it can make use of external components, such as mobility analyzers estimating the mobility behavior of a mobile user. Decision made by the BR based on the gathered information and defined policies are propagated by commands to other network elements [7].

III. Use case

Critical use cases in MEC are typically scenarios of real-time applications in (fast) moving vehicles, like moving cars, trains or motor cycles. These application run too slow on mobile devices or need too much energy. An example for these is the using of computer graphics or virtual reality applications. In order to display of 3D-pictures of computer tomography a lot of computing power and memory is necessary. Figure 3 show the use case diagram.

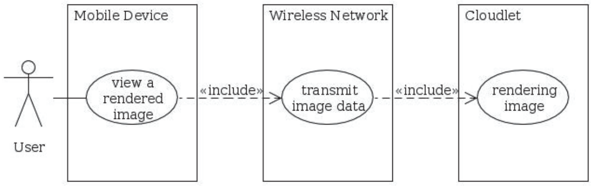

Fig. 3. Display 3D-pictures of computer tomography on a mobile device

In this case a person ride a fast moving train and want to show some 3D-animations of computer tomography data. The person want to zoom in and zoom out because of the limited size of the mobile. In order to render the picture a lot of computing power is necessary. Usually a workstation will be used in a office environment to process the data.

If the render process is located at a central cloud no real-time feeling can be established because of the high latency between the mobile device and the central cloud. To get a near real-time feeling in this scenario, the processing of the data has to be in the near of the mobile device to prevent high latency. This can be reached by the using of cloudlets.

But without the moving of applications between different cloudlets the latency increase by time again. Methods for controlling the positioning of the data and processing-power in order to follow the mobile device is necessary to keep the latency constant on a low level.

IV. Requirements

In order to keep the latency constant on a low level with long running applications, the applications on the fixed infrastructure side need to be mobile and have to follow the movement of the mobile user. This require a better control of the serverside applications on the consideration of the mobile device. In order to optimize the control of the mobility of the applications on the fixed infrastructure, a first requirement is to collect necessary informations from different sources. The following informations would be reasonable, because they affects the long term latency of the application:

- Information about the location of the mobile device
- Information about the location of the serverside application (the location of the cloudlet)
- Locations of available cloudlets in the network with their properties.
- Energy consumption of the mobile device
- Capacity status of the battery of the mobile device
- The workload of the cloudlet in the environment of the mobile device
- Radio-Bandwidth of the mobile device
- Latency requirements of the application
- Processing power requirements of the application
- Transmission path length between the mobile device and the processing unit (cloudlet)

The above information can be divided into two groups – static and dynamic information. Static information are typically constant over the runtime of the usage of the application. For example the location of cell towers are mostly fix. Dynamic information changes over the runtime of the application. Examples of this kind of information are the status of the capacity of the battery of the mobile device.

The second requirements for the task is to control the resources reasonable depending of the collected information. Those a unit is necessary. which use this information to optimize the behavior of the applications on the fix infrastructure. Therefore the second requirement for the task to control the mobility of the fixed infrastructure is a centralized component, which collect the information and provide them for optimize the location of the application in the different available cloudlets.

Furthermore, it should be noted that open interfaces are required and roaming within the infrastructure of one provider and between infrastructures of different providers.

V. Architecture Concept

Based on the requirements above, an additional central component is necessary, which collect the information and provide the results to other network components. An analog concept exists in the form of the Black Rider, which also collect information and provide these to other network components. However the component Black Rider is justified to collect information about the mobile device, not for the complete infrastructure inclusive cloudlets.

An architecture concept containing the old entity Black Rider an the new entity MEC Manager for the mobility management of the applications in the MEC environment is shown in Fig. 4.

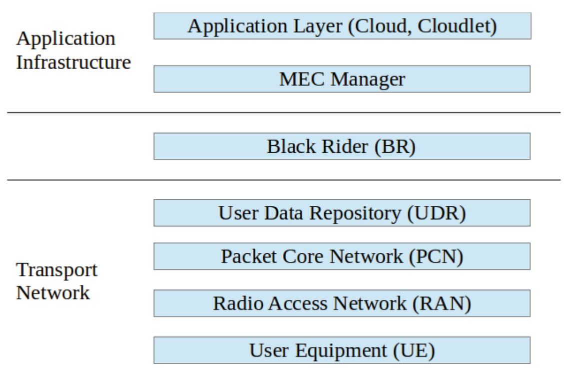

Fig. 4. Reference-Architecture of the MEC Manager

On top there is the application layer including the Cloud and Cloudlets infrastructure. The MEC Manager gets information on the mobility management context and bearer management context of the mobile users from the BR, for example by subscribing appropriate user events, and can make its own decisions on application mobility management within the application infrastructure. The UDR stores all user data, except user content and user context data, which are required for transport network control. The transport network consists of user equipment, radio base stations of the radio access network and the packet core network. Basically, the BR monitors the mobility behavior of the mobile user in the transport network and makes optimized decisions on the traffic distribution in the transport network. Based on these decisions in the transport network, the MEC Manager uses appropriate mobility management and bearer management information from the BR and his own collected information from the cloudlet infrastructure to optimize the behavior of the Application Layer, like movement of VMs between cloudlets in order to improve the QoE for the mobile user.

Thereby the MEC Manager can execute context- and policy-bases decisions to control the components of the cloud infrastructure. By using the MEC Manager important information about the network infrastructure and their conditions are available at a central point in the network. As a result new opportunities for the optimization for other network components are available. In the final architecture parts of the Black Rider architecture can be reused. So a common use of the database of the Black Rider is conceivable.

VI. Conclusion

Starting from simple use-cases, is shown first that an efficient management of application mobility is required. Afterwards requirements for the control of the cloud infrastructure was derived. Next a simple control architecture is developed based on existing blocks as cloudlets and Black Rider. Through the use of the concept "Black Rider" it is possible in principle to establish a central management instance called MEC Manager that enables the distribution and management of mobile applications in the cloud infrastructure.

An evaluation of the new architectural concepts with concrete examples is pending. Likewise, the development of decision strategies for optimization. Furthermore, interfaces between different providers should be defined so that VMs can be transported in cloudlets over provider boundaries.

References

[1] M. Weiser, “The computer for the 21st century,” Scientific American, pp. 94-104, September 1991.

[2] R. Want, “The power of smartphones,” IEEE Pervasive Computing vol. 13, issue 3, pp. 76-79, 2014.

[3] ETSI Consortium 2016, “Mobile Edge Computing,” http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing.

[4] S. Frei, W. Fuhrmann and B. Ghita, “Generic real-time traffic distribution framework: Black Rider,” 22nd International Conference on Computer Communication and Networks (ICCCN 2013), pp. 1-8, 2013.

[5] M. Satyanarayanan, P. Bahl, R. Cáceres and N. Davies, “The case for vm-based cloudlets in Mobile Computing,” IEEE Pervasive Computing, 2011.

[6] K. Ha, Y. Abe, Z. Chen, W. Hu, B. Amos, P. Pillai and M. Satyanarayanan, “Adaptive vm handoff across cloudlets,” Technical Report CMU-CS-15-113, CMU School of Computer Science, 2015.

[7] S. Frei, W. Fuhrmann and B. Ghita, “Framework for generic context- and policy-based traffic distribution in heterogeneous wireless networks: Black Rider,” Workshops of 27th International Conference on Advanced Information Networking and Applications (WAINA), pp. 534-541, 2013.