MEC support for C-V2X System Architecture

Yancong Wang, Jian Wang
College of Computer Science and Technology
Jilin University
Changchun, China
e-mail: yancong18@mails.jlu.edu.cn,
wangjian591@jlu.edu.cn

Yuming Ge, Bingyan Yu, Cheng Li, Lu Li
Institute of Technology and Standards Research
China Academy of Information and Communications Technology
Beijing, China
e-mail: geyuming@caict.ac.cn, yubingyan@ritt.cn,
licheng@caict.ac.cn, lilu@ritt.cn

***Abstract*—This paper proposed an original four-layer architecture of MEC support for C-V2X system, summarizing the participants in C-V2X scenarios and the logical relationship between the participants. Three novel APIs are designed to meet the requirement of C-V2X end-to-end service under the MEC support for C-V2X system architecture. The general process of MEC support for C-V2X service under our architecture is also considered in this paper with two typical C-V2X user cases as examples: HD Map and Intelligent Intersection.**

Keywords-MEC; C-V2X; system architecture; end-to-end interface; service process

I. INTRODUCTION

With the rapid development of the mobile network and the Internet of things, massive intelligent terminal devices are widely used in various industries. Deploying different applications in the central cloud, users could experience convenient and rich services. However, network load and data transmission delay would be greatly increased due to the massive access of devices into the central cloud, which does not satisfy the services that require low-latency, high-bandwidth, and high-reliability. Mobile Edge Computing as a possible solution that migrates the computing platform from the mobile core network to the edge of the mobile access network, could reduce the end-to-end delay of mobile service delivery and enhance the user's experience, which is received extensive attention and studied by academia and industry [1-3].

For Mobile Edge Computing, academia has generally focused on technical research in three aspects of Computing, Caching, and Communication [4-7]. The key enablers of Mobile Edge Computing such as cloud technology, SDN/NFV, and smart devices are also hot topics in academia [8].

To meet the requirement of the industry, ETSI (European Telecommunication Standard Institute) extended the concept of Mobile Edge Computing to Multi-access Edge Computing (MEC), enlarging the application scenarios from mobile cellular networks to other access networks (All MEC below represents Multi-access Edge Computing). ETSI has deeply studied in the use cases, technical requirements, framework and reference architecture of MEC, and completed the standardization of platform requirements and reference architecture [9]. The standard defined the platform's orchestration and management of applications, including the life cycle management of applications. Based on the RESTful design principles, a standardized communication mechanism for the APIs of network and application interaction are designed, covering the basic functions such as service discovery, registration, invocation, and security [10]. ETSI MEC reference architecture "Fig. 1" includes the MEC functional entities and the reference points between them. The MEC system consists of the service domain and the management domain. Service domain includes MEC platform, MEC applications, and virtualization infrastructure which provides computing, storage, networking, and other resources. Management domain includes MEC system level management and MEC host level management. The reference points include the one related to the functions of the MEC platform (Mp), the one about management (Mm) and the one with external entities of the MEC system (Mx).

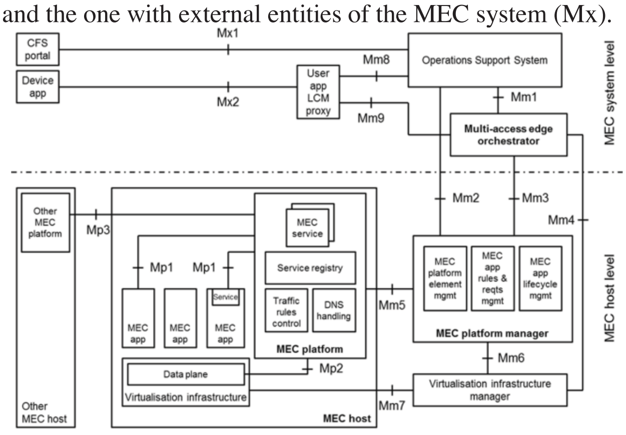

Figure 1. ETSI MEC reference architecture

MEC is considered within the framework of 5G networks by 3GPP (3rd Generation Partnership Project), which provides the closer deployment and operational environments to users for operators and the third-party service applications to reduce end-to-end latency and backhaul bandwidth and enable the efficient distribution of service application content [11]. In addition, 3GPP developed the northbound API framework which considered

978-1-7281-0535-2/19/$31.00 ©2019 IEEE

the corresponding APIs of ETSI MEC reference architecture [12].

C-V2X services such as autonomous driving and real-time HD map downloading are extremely sensitive to latency and reliability. It is not only necessary to realize "people-vehicle-road-cloud" collaborative interaction via PC5 interface or Uu interface, but also to deploy C-V2X services locally with MEC to reduce end-to-end communication delay and increase network bandwidth. MEC could also provide supports of auxiliary computing and data storage for C-V2X applications.

ETSI has collected and analyzed the relevant V2X use cases, but the standards of architecture and APIs for V2X-specific requirements are still being drafted [13]. 5GAA (5G Automotive Association) supposed that MEC could effectively provide low latency, flexible deployment, and access to local context information for four types of V2X use cases such as safety, Convenience, Advanced Driving Assistance, and Vulnerable Road User [14]. ETSI MEC reference architecture is recommended by 5GAA to be used as the solution for edge computing.

II. MEC Support for C-V2X System Architecture and End-to-End APIs

Based on MEC support for C-V2X scenarios, the original four-layer system architecture of MEC support for C-V2X is firstly proposed all over the world "Fig. 2", including central cloud layer, edge computing platform layer, roadside equipment layer and user equipment layer. Although the basic elements involved refer to ETSI MEC architecture, our MEC system architecture fully considers all participants involved in V2X scenarios, which is more detailed and targeted than the general MEC reference architecture published by ETSI in the special vertical sector. In addition, the general ETSI MEC reference architecture has great openness and automated lifecycle management capabilities, but the end-to-end service interfaces of the vertical sectors as C-V2X have not been developed by ETSI. It is unable to meet the end-to-end service requirements of MEC support for C-V2X in the multi-access, multi-vendor, and multi-operator scenarios based on the existing architecture of ETSI and other related work. Therefore, we firstly summarize and propose three types of APIs related to the implementation of C-V2X end-to-end applications.

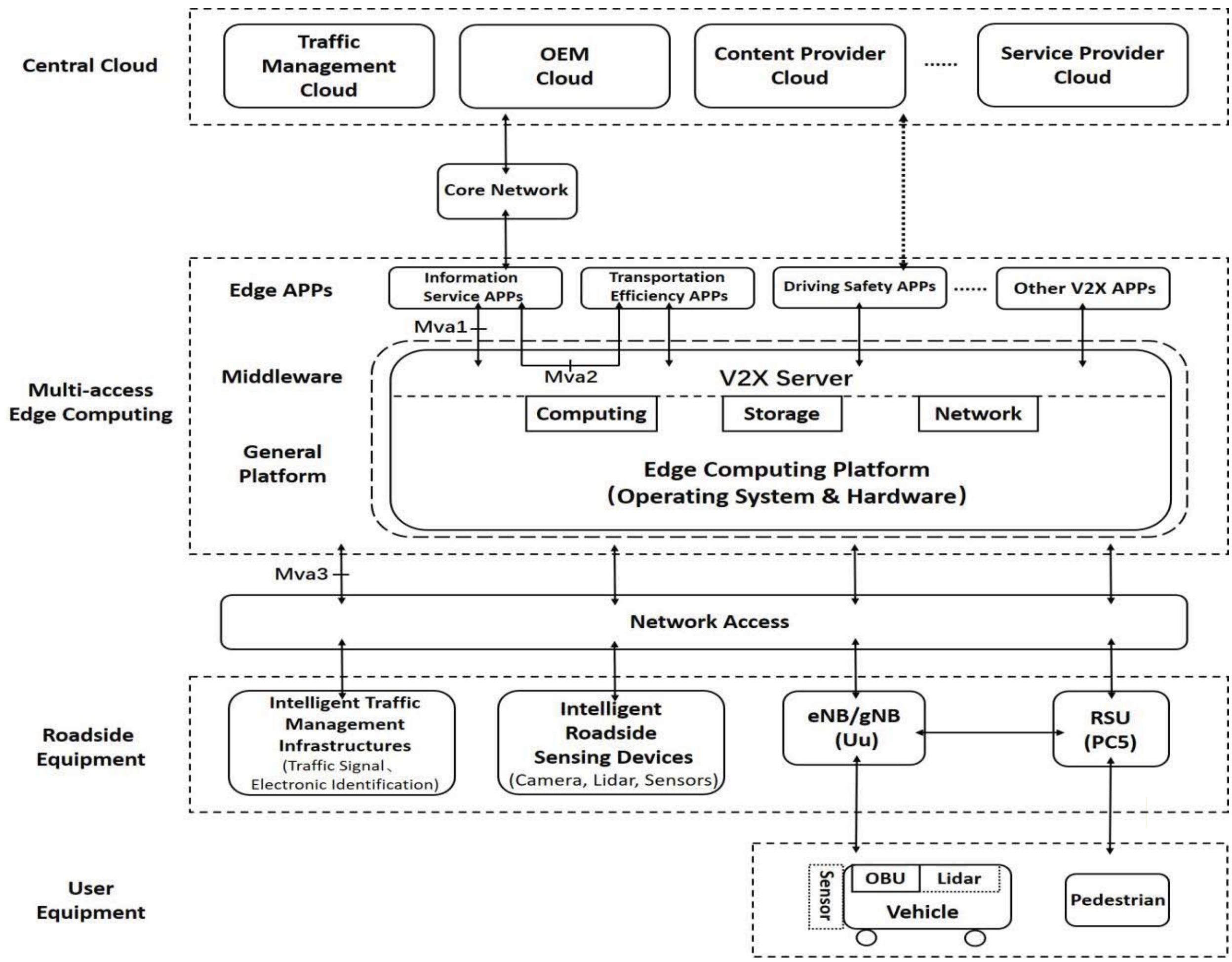

Figure 2. MEC support for C-V2X System Architecture

As the core of this architecture, the edge computing platform layer consists of a general platform, middleware, and edge applications. The general platform for edge computing contains the operating system and hardware resources (or virtualized resources). On the one hand, the general platform for edge computing could collect the real-time surrounding environment and vehicles information which is perceived from the intelligent roadside facilities such as traffic signal controllers, cameras and lidars, or collect the relevant information of vehicles and pedestrians through RSUs and various base stations. On the other hand, the general platform provides an operating environment and core capabilities like applications management, message transmission, computing, storage, and network for V2X Server and edge applications which are deployed on it. V2X Server is abstracted from the resource pool of the edge computing general platform as a specific server for C-V2X edge applications, which supports and manages the C-V2X edge applications service that is registered and published on V2X Server. C-V2X edge applications could either provide services to other C-V2X edge applications in need or access data information from other C-V2X edge applications through V2X Server. V2X Server could also create new services with the data information provided by the edge computing general platform and C-V2X edge application services, then provides the service or data support for C-V2X edge applications and user equipment. C-V2X edge applications are divided into four categories: information services, transportation efficiency, driving safety and others such as customized C-V2X application services for user needs). In some scenarios, C-V2X edge applications need to be able to interact with central cloud.

In addition, three types of APIs (Mva1, Mva2, Mva3 as shown in Fig. 2) related to the implementation of C-V2X end-to-end applications are summarized and proposed. The Mva1 interface between V2X Server and C-V2X edge applications has similar but not the same functions as Mp1 reference point in ETSI MEC reference architecture. V2X Server provides the life cycle management like service registration, discovery, access, update and undo for C-V2X edge application services through Mva1 interface. The Mva1 interface is also used for mutual access and data transfer between C-V2X edge application services and the services created by the V2X server. The Mva2 interface is defined between C-V2X edge applications through V2X Server, which allows C-V2X edge applications to provide their own services or request services from other applications registered and published in V2X Server and the services created by V2X Server. The data between these applications could also interact via Mva2 interface. The Mva3 interface is located between the edge computing platform and roadside equipment. On the one hand, the edge computing platform collects data information in a specific format gathered from various roadside equipment via Mva3 interface. On the other hand, UE requests the V2X Server in the edge computing platform for the C-V2X edge application services through RSU or base station via Mva3 interface and received the service information sent by the edge computing platform after successfully responding to the request.

III. MEC Support for C-V2X Service Process and Examples

According to the MEC support for C-V2X system architecture proposed in chapter 3, the following general process of MEC support for C-V2X service is extracted (Fig 3).

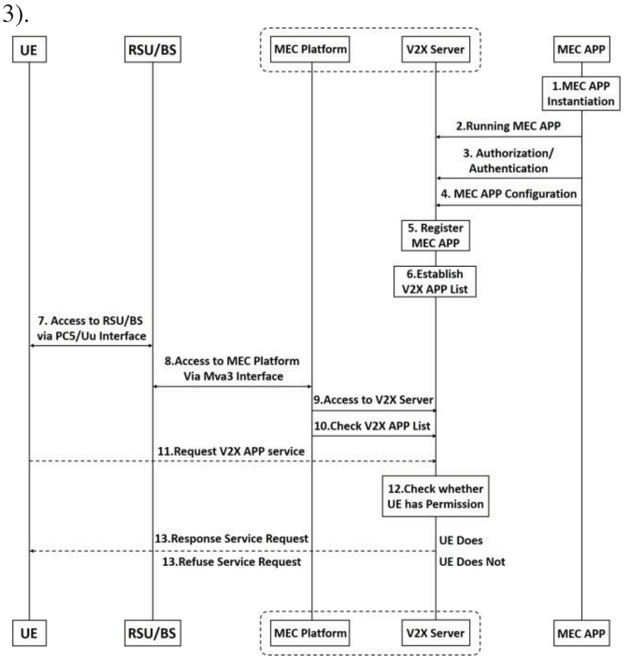

Figure 3. General process of MEC support for C-V2X service

After instantiated, authorized and authenticated the C-V2X edge applications, the edge computing platform configured them and allowed C-V2X edge application service providers to register their application services in V2X Server. Based on the instantiated information of C-V2X edge application services, V2X Server establishes a V2X application service list according to C-V2X application categories. When UE accesses to the edge computing platform through RSU or base station via Mva3 interface, UE could access V2X Server in the edge computing platform and apply to check the V2X application service list. According to the requirement, UE selects C-V2X edge applications to request service. V2X Server received the request from UE then checked whether UE has the permission to use the service. If UE does, V2X Server opens the access resources of the C-V2X edge application to UE and implements the corresponding operation response. Some algorithms of C-V2X edge applications rely on the real-time information of roads and vehicles, V2X Server could provide data collected from the edge computing platform for these C-V2X edge applications. When a C-V2X edge application needs to access other C-V2X edge applications, C-V2X edge application requests to check the V2X application service list from V2X Server, V2X Server treats the C-V2X edge application with access requirements as a special client which could request service from the corresponding C-V2X edge application via the Mva2 interface, then responds the service request.

Combined with two typical C-V2X use cases, HD Map and Intelligent Intersection, the specific service processes of these two examples are shown in the flow charts (Fig. 4 and Fig. 5), separately. The following flow charts have omitted the general service process in Fig. 3.

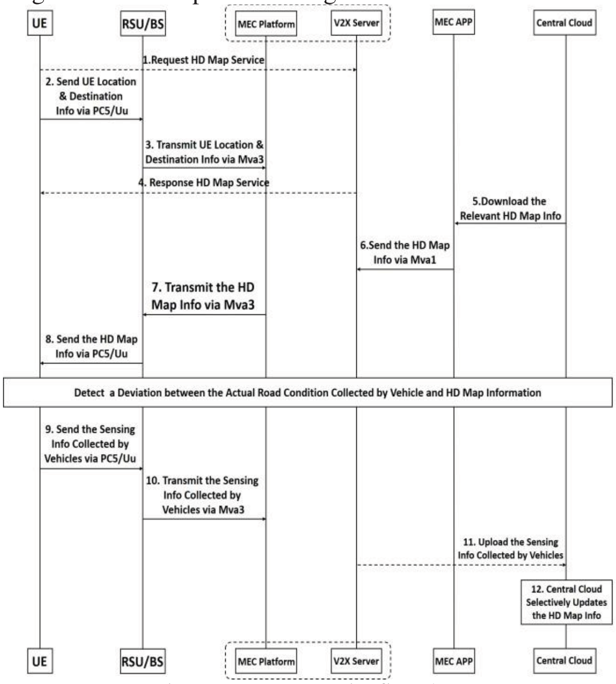

Figure 4. HD Map service flow chart

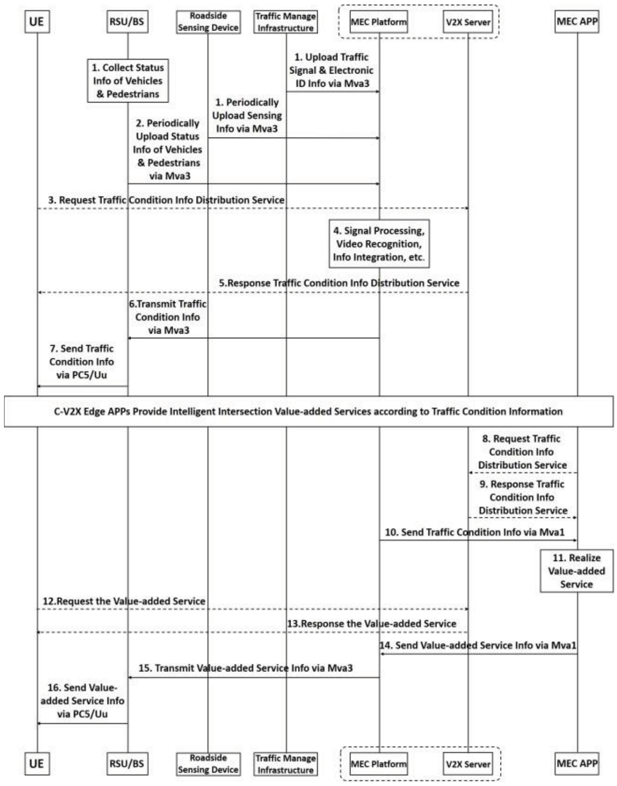

Figure 5. Intelligent intersection service flow chart

A. Example 1: HD Map

The vehicle sends its current location and the destination information to RSU or base station via PC5 interface or Uu interface, then the information is transmitted to the edge computing platform by RSU or base station. The edge computing platform extracts the HD map information of the vehicle's current and target geographic region from the central cloud or the local cache and sends it to the vehicle through RSU or base station via PC5 interface or Uu interface, respectively. If the vehicle is equipped with sensors such as lidar or camera, the vehicle could compare its collected sensor information with HD map information. When there is a deviation between the actual road condition collected by vehicle and HD map information, the vehicle could transmit its sensing information to the edge computing platform through RSU or base station via PC5 interface or Uu interface. The edge computing platform uploads the sensing information collected by the vehicle to the central cloud which could selectively update the HD map information. HD Map with the support of MEC could reduce the latency and the burden on the transmission bandwidth of the core network.

B. Example 2: Intelligent Intersection

The intelligent roadside sensing devices such as camera, lidar and different kinds of sensors deploy at the intersection and fully cover the intersection. The intelligent roadside sensing devices send the information gathered from the intersection to the edge computing platform. RSUs also send the status information of the relevant vehicles and pedestrians collected at the intersection to the edge computing platform through the Mva3 interface. The edge computing platform performs signal processing, video recognition, information integration and other operations on the collected information from intelligent roadside sensing devices and intelligent traffic manage infrastructures, which could analyze and predict the whole traffic situation including the location, speed and direction angle of all vehicles, pedestrians and other traffic elements in real time at the intersection. The processed data could be transferred to RSU or base station via Mva3 interface, and then to the vehicles via PC5 interface or Uu interface. Moreover, it could also be transmitted to the C-V2X edge applications via V2X Server or further processed by V2X Server to provide more value-added services for vehicles as signal phase and timing message sending, vulnerable road user collision warning, etc. The intelligent intersection with the support of MEC could enable the vehicles to obtain the information of all traffic participants and traffic management infrastructures within the scope of the whole intersection, which comprehensively improves the safety, comfort, and efficiency of vehicles passing through the intersection.

IV. Conclusion

Based on our novel architecture and three end-to-end service APIs of MEC support for C-V2X system, the possible service processes of two typical C-V2X applications, HD map and intelligent intersection, verify the feasibility of the proposed architecture and interfaces in this paper. Guided by our reference system architecture, it is helpful for the participants involved in MEC support for C-V2X service process to clarify their respective responsibilities and the collaboration with other participants, which fills the research gap of MEC support for the vertical sector that ETSI and other related work did not consider in detail. In addition, as the key research in the future work, the standardization of these three APIs will drive the MEC support for C-V2X service in the multi-access, multi-vendor, and multi-operator scenarios.

ACKNOWLEDGMENT

This paper was subsidized by the project of Research on Key Technologies and Social Legal Problems of Autonomous Electric Vehicles, project ID: 2018YFB0105202.

REFERENCES

[1] A. Ahmed and E. Ahmed, “A survey on mobile edge computing,” in Proc. 10th Int. Conf. Intell. Syst. Control (ISCO), Coimbatore, India, 2016, pp. 1–8.

[2] M. Patel et al., “Mobile-edge computing,” Eur. Telecommun. Standards Inst., Sophia Antipolis, France, White Paper, 2014.

[3] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, “Mobile edge computing—A key technology towards 5G,” Eur. Telecommun. Standards Inst., Sophia Antipolis, France, White Paper, 2015.

[4] K. Zhang, Y. Mao, S. Leng, A. Vinel, and Y. Zhang, “Delay constrained offloading for mobile edge computing in cloud-enabled vehicular networks,” in Proc. 8th Int. Workshop Resilient Netw. Design Modeling (RNDM), Halmstad, Sweden, Sep. 2016, pp. 288–294.

[5] P. Mach and Z. Becvar, “Cloud-aware power control for real-time application offloading in mobile edge computing,” Trans. Emerg. Telecommun. Technol., vol. 27, no. 5, pp. 648–661, Dec. 2015.

[6] J. Q. Fajardo et al., “Introducing mobile edge computing capabilities through distributed 5G cloud enabled small cells,” Mobile Netw. Appl., vol. 21, no. 4, pp. 564–574, 2016.

[7] D. Liu, B. Chen, C. Yang, and A. F. Molisch, “Caching at the wireless edge: Design aspects, challenges, and future directions,” IEEE Commun. Mag., vol. 54, no. 9, pp. 22–28, Sep. 2016.

[8] 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, Mar. 2017.

[9] Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG), “ETSI GS MEC 003 V2.1.1 (2019-01), Multi-access Edge Computing (MEC); Framework and Reference Architecture,” Eur. Telecommun. Standards Inst., Group Specification, Jan. 2019.

[10] Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG), “ETSI GS MEC 009 V2.1.1 (2019-01), Multi-access Edge Computing (MEC); General principles for MEC Service APIs,” Eur. Telecommun. Standards Inst., Group Specification, Jan. 2019.

[11] Technical Specification Group Core Network and Terminals, “3GPP TS 29.222 V15.3.0 (2019-03), Common API Framework for 3GPP Northbound APIs (Release 15),” 3rd Generation Partnership Project, Technical Specification, Mar. 2019.

[12] Technical Specification Group Services and System Aspects, “3GPP TS 23.501 V15.5.0 (2019-03), System Architecture for the 5G System Stage 2 (Release 15),” 3rd Generation Partnership Project, Technical Specification, Mar. 2019.

[13] Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG), “ETSI GS MEC 022 V2.1.1 (2018-09), Multi-access Edge Computing (MEC); Study on MEC Support for V2X Use Cases,” Eur. Telecommun. Standards Inst., Group Specification, Sep. 2018.

[14] D. Sabella et al., “Toward fully connected vehicles: Edge computing for advanced automotive communications,” 5G Automotive Association, White Paper, Dec. 2017.