AECC

AUTOMOTIVE EDGE
COMPUTING CONSORTIUM

Distributed Computing in an AECC System

White Paper

Version 1.0.0 • 2021-08-18

Contents

Figures

Tables

1 Preface

The Automotive Edge Computing Consortium (AECC) is a not-for-profit association of cross-industry global leaders working to explore the rapidly evolving and significant data and communications needs involved in instrumenting billions of vehicles worldwide.

The AECC has prepared this white paper to communicate the AECC's assumptions about the off-vehicle computing environment that supports connected vehicles. It is supplemental information to the AECC's "General Principle and Vision" White Paper.

Notice: the information presented in this white paper reflects the current state of discussions, but it is not final and is subject to change.

For information about the AECC and AECC publications, see https://aecc.org/resources/publications/.

2 Executive summary

The computing requirements of managing and supporting connected vehicles require a significant change to the way in which stakeholders implement the off-vehicle computing ecosystem.

This white paper explores the requirements that motivate distributed and decentralized implementations of the mobility services provided to connected vehicles.

There are similarities with other mobility service applications. For example, the clients of standard consumer mobility services and of connected vehicle services share a characteristic that the client's location can change frequently. Users change locations all the time, within a city, from town to town, across different states and countries throughout the world. Mobility services users moving from place to place in multiple locations are similar to vehicles moving across our streets and highways.

There are two important differences in connectivity and distributed computing for connected vehicles. Data transfer for standard consumer mobility services flows mostly one way, as downloads from the provider to the consumer. For connected vehicles, the primary direction of data transfer is uploads from the vehicle to the provider. For connected vehicles, the amount of uploaded data is massive, concentrated (in time and location) and often time sensitive. The AECC estimates that the connected vehicles will transfer up to 10 exabytes per month in 2025 (see the AECC WP).

To provide desired service levels, the mobility service may need to use different mobility service instances such as ones that are close to the vehicle.

With these points in mind, the whitepaper provides an overview of distributed computing and how it can be applied in the automotive industry. The document illustrates service and architecture level requirements as well as a functional architecture for distributed computing in the context of automotive edge computing. It also provides an initial solution profiling to analyze existing technologies from other standards organizations and open source communities to identify candidate solutions to satisfy AECC requirements and capture potential gaps to be addressed by connected vehicle stakeholders.

3 The AECC ecosystem

An AECC ecosystem is envisioned as a collection of AECC systems (see Figure 1: AECC system overview). Each AECC system is composed of an AECC service, computing infrastructure services, mobility services and the access networks, and gateway networks that provide access to vehicles and to clients of the AECC and mobility services.

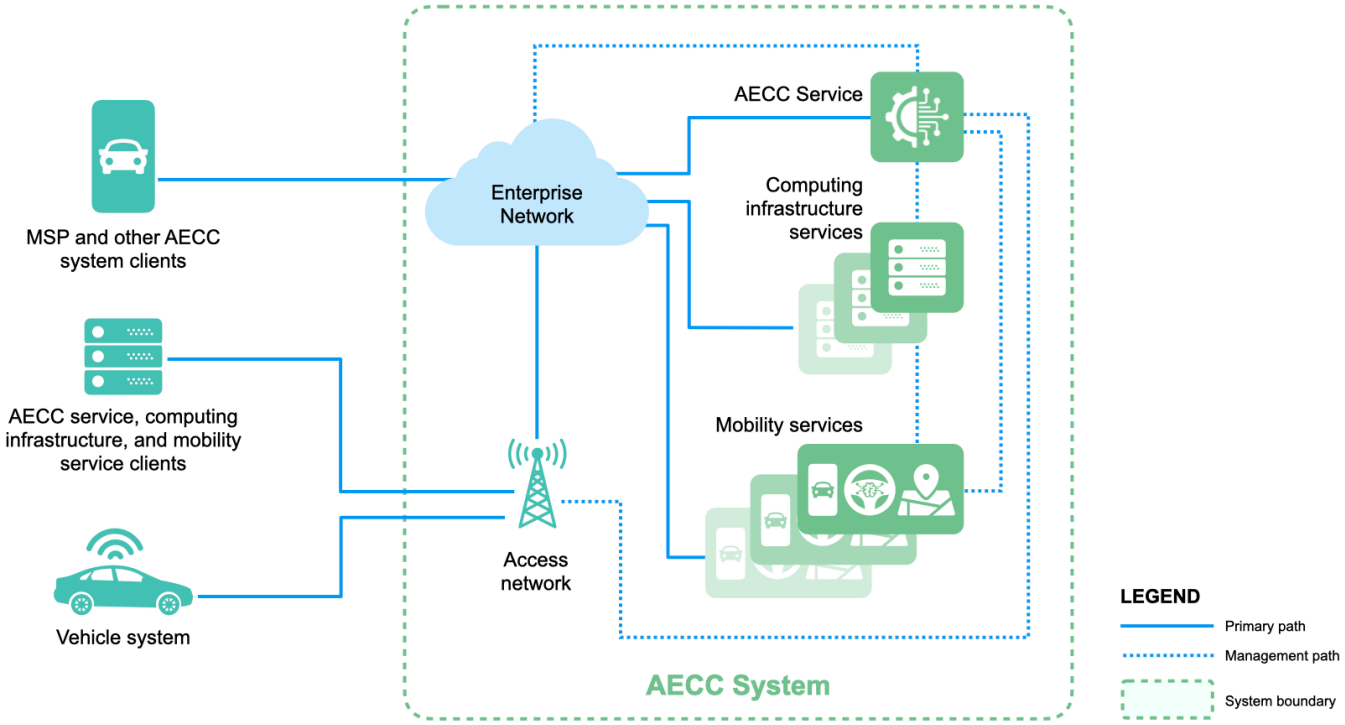


Figure 1: AECC system overview

The AECC service is responsible for overall management of the AECC system. Computing infrastructure services provide access and management of the supporting computing platforms, which may include any number and combination of computer systems implemented in any type of data center, the cloud, as edge computing or in the vehicle systems. Mobility services are applications that provide services to vehicles. There may be multiple types, including intelligent driving and high-definition mapping services. The enterprise network is the IP-based network that connects between the AECC service, the computing infrastructure services and the mobility services. Access networks are typically cellular but may use other technologies including WLAN.

Vehicles connect to provide or receive information from various mobility services (see Figure 2: Connected vehicles). As shown, a mobility service may be both decentralized and distributed between center and edge locations. Edge components of a mobility service play important roles. One is to provide faster responses to requests from vehicles. A second is offload the cloud and transport networks. Another is to comply with legal requirements; for instance, some types of data might not be allowed to cross country borders.

As shown in Figure 2, the services at the mobility service (edge) act as a decentralized anchor point for the massive amount of data collectively uploaded by vehicle systems. A connected mobility service may use different edge platforms depending on service needs. The upload challenges addressed by this decentralized off-vehicle configuration are twofold:

1. The service at the edge allows application owners or mobility service providers to define reduction contexts. For example, when multiple images are uploaded by different vehicles for the same road segments, the service at the edge can preprocess these images and propagate a reduced, curated subset of the uploads. Examples of such curated reductions include:
   - Improved localization, correcting GPS error using anchor-image localization
   - Coalescing multiple images and or detections, aggregating and de-duplicating
   - Verifying, aging, and propagating uploaded detection to related edge contexts for further reduction
2. The service at the edge allows applications to set global quotas for a given regional area. Such quotas might govern the number or size of uploads in a specified timeframe. We expect access network bandwidth to increase significantly with the progression to 5G. Smart management of uploads can "flatten the curve" of usage of the access network and avoid spikes of duplicate data from a busy street and peak hours. This enables better planning and better best-effort pricing of data plans, and better designs for average and peak loads. Mechanisms at the edge can be used to throttle access to the edge. For example:
   - Keeping a contextual data handle for image and sensory data stored in the vehicle system
   - Cherry-picking based on merit, relevance, and other attributes
   - Fetching data from vehicle systems to the mobility edge using a contextual handle-based ledger

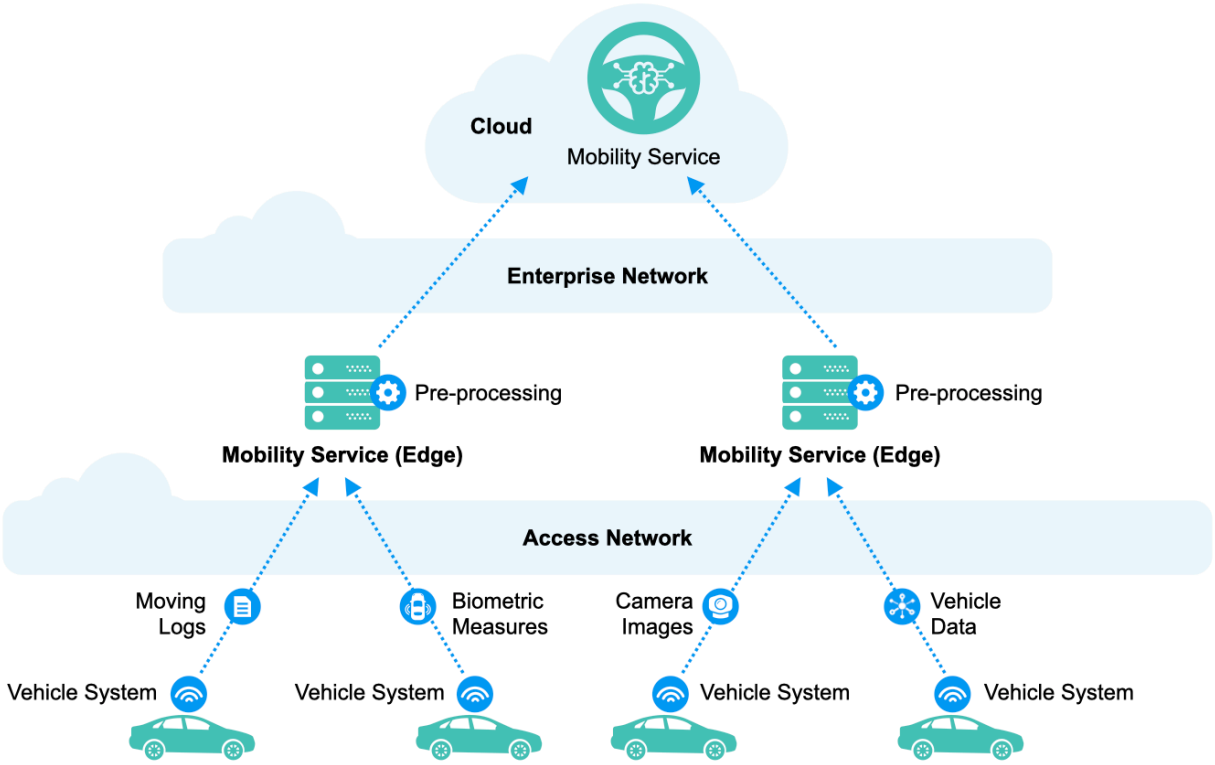


Figure 2: Connected vehicles

4 System requirements

4.1 Service requirements

4.1.1 Service requirements overview

In the near future, vehicles will exchange large amounts of data with off-vehicle services. In aggregate, the amount of data is so large that it is impossible to manage the data to one, or even a few central places. This would be like sending all postal mail to one central clearing house. As with postal services, processing the data must be distributed closer to the sending and receiving parties. This minimizes the need to transport large amounts of data over large distances. Thus, the distributed processing requirement leads to geographically distributed mobility services.

Other factors that drive the geographic distribution of data processing are the regional legal and privacy requirements on who or what can access data and from where. These requirements may not permit the centralization of data gathered from vehicles operating across regional boundaries.

Because of the need to manage where computing happens, most AECC systems cannot be realized by a single, location-agnostic cloud system.

An AECC system supports two types of services. The first are mobility services, such as intelligent driving and high-definition mapping. These provide services to connected vehicles and are deployed to the computing platforms of an AECC system by mobility service providers. The second are the AECC services that present a common management interface for the constituent computing infrastructure, enabling the orchestration of resources and services necessary to support the deployment of distributed mobility services at scale.

In the edge environment, the focus of service delivery changes from centrally hosted services to services distributed across multiple computing resources. This allows the computing resources of the service to be much closer to the service's clients. Such distributed service implementations must provide expected levels of service. The following clauses discuss important service characteristics.

Availability

The ability to provide a service on demand and without interruption.

Business Value

The importance, worth or usefulness of the utility provided by the service.

Complexity of data transmission

The data is distributed geographically. Transmission of large amounts of data is complex and may increase the running cost for edge service operators and mobility service providers. Solutions must distribute data economically.

Configurability

The ability to upgrade services while maintaining required service levels at an affordable cost. The ability to configure the behavior of the system.

Service implementations must be flexible enough to allow a service provider to either actively or passively choose the appropriate locations to store data based on current or future legal and compliance requirements.

Interoperability

The ability of services to exchange and make use of data and other information.

Discoverable

For providing services effectively, the available service instances and infrastructures need to be discoverable and accessible by other entities; these include vehicle systems, roadside units and back-end service instances that are addressable and accessible within the AECC system.

Interoperability for mobility services

Interoperability implies the ability of independent systems to work together as a unified system. Mobility services require transportation resources in a region to jointly participate in service planning and delivery. For mobility services used while driving, interoperability also implies the need to interact with devices on the roadside. We expect systems to exchange data seamlessly when required.

Performance

The degree to which various metrics, such as throughput, latency, utilization, power consumption and others, are satisfied.

According to the estimate in the AECC's white paper on high-definition mapping, by 2022, every five million economy model cars with rich vehicle-side clients would generate 123.8 terabytes per day, and thin vehicle-side clients would generate 10.8 petabytes per day in real-time objects reporting mode. This is a huge amount of data to process.

Regulatory Compliance

Many countries have published laws to regulate the use of data based on the location where data is generated and on where that data is stored (e.g. GDPR). Additional regulations govern the transmission and use of personal identifiable information, payment card information and other attributes found in the data, (e.g. PCI DSS).

Sensitive data must be identified by the system and treated in the right way according to scenarios and service requirements.

Resilience

The ability to recover quickly from failures or difficulties.

Robustness

The ability to tolerate conditions that might affect the functionality of a service.

For mobility service delivery, the basic requirement is to keep the service stable and uninterrupted. Further, with higher service levels, the back-end computing system must be able to process huge amounts of data with latency in an acceptable range.

Scalability

The ability to accommodate increased and decreased workloads while maintaining required levels of service.

To comply with service availability requirements triggered by pre-defined conditions, the system must be able to optimize resource utilization, including bandwidth, storage, and computing resources.

For service instance deployment, the system must be able to list the hosts that meet user requirements and provide specific deployment details and necessary management interfaces. This allows users to always be aware of the system's running status for managing the resources of the computing platform and the lifecycle of the instance.

Security

A secure computing environment provides safety from downtime, interference, or malicious intrusion, including protection against unauthorized disclosure of personal identifiable information (PII).

Creating a secure computing environment requires identification of trusted entities and setting up security boundaries to ensure that multiple parties of communication are from trusted services or systems; this is critical for providing stable and reliable services. Authorization mechanisms must be applied to the third-party services to ensure providers are trusted parties.

To ensure the safety of each service, the management of service instances and isolation of underlying computing resources are necessary prerequisites. It is essential to have the ability to verify the behaviors of services and to shut down service instances when necessary.

Standards

Implementation of a distributed mobility service uses multiple component services from different vendors. Without industry coordination, similar mobility services from different vendors risk to having different service interfaces and functions. To eliminate complexity and inconsistency, it is desirable to abstract these service interfaces to hide unnecessary implementation differences. Such abstract interfaces can provide common interfaces that enable use of the service without needing to support multiple different interfaces. The use of standards supports and enables deployment of mobility services at scale.

Versioning

The ability to support multiple versions of a service at the same time.

4.2 Architectural requirements

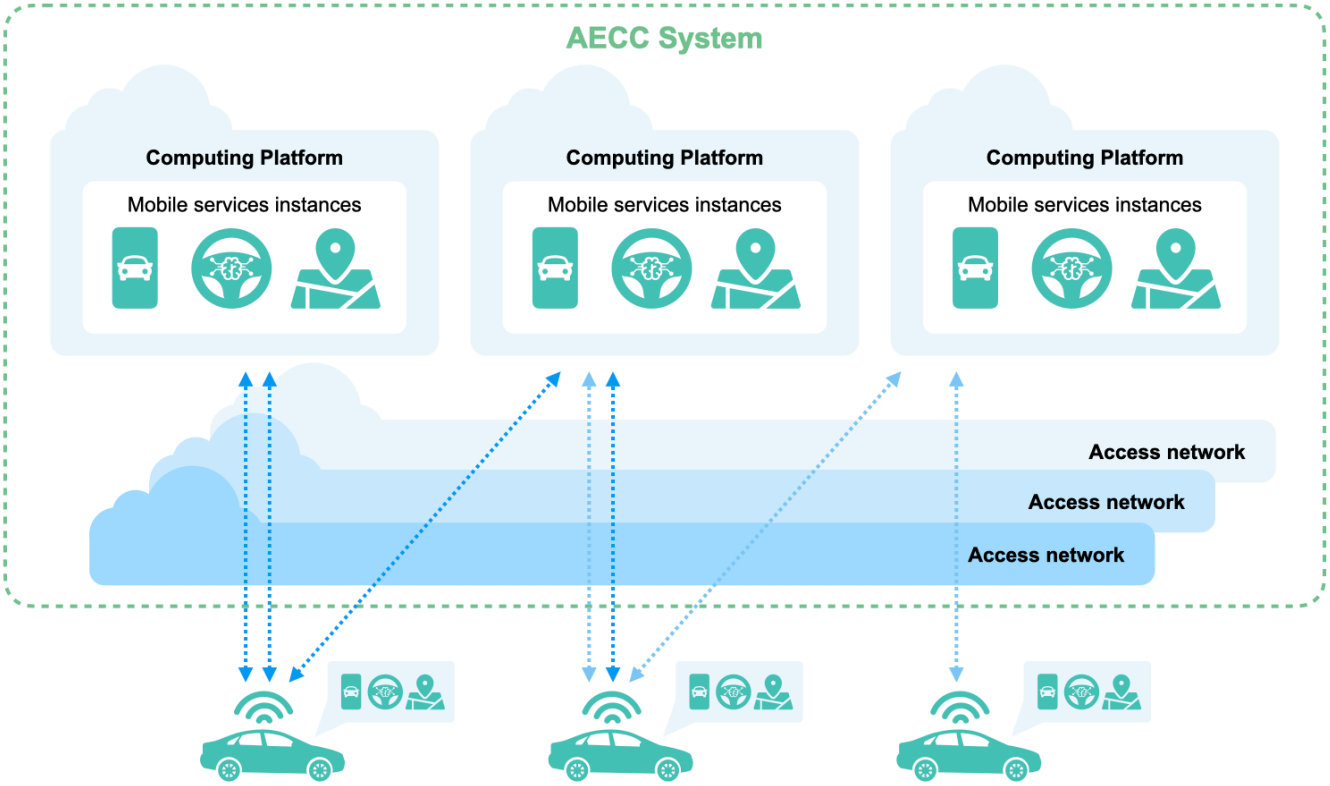


Figure 3: High-level illustration of the AECC system

In an AECC system, mobility service instances are deployed onto computing platforms. Computing platforms may take different forms and may have different proximities to connected vehicles. A primary goal of a distributed deployment is to fulfill the service requirements of the hosted mobility services.

4.2.1 Access networks

Access networks provide a connection from end users, such as vehicles, to an enterprise network. Access networks typically use some form of wireless technology, such as cellular or WLAN.

An AECC system acknowledges that a vehicle may connect to a mobility service instance through available access networks, including cellular (4G and 5G) as well as WLAN. Typically, these networks are operated by different providers. For whichever access network is selected, the vehicle must be able to connect to an appropriate mobility service instance. This may be one that is in proximity to the vehicle or farther away, such as in a regional cloud. A routing scheme is needed to forward data packets between vehicles and mobility service instances. While the logic of the routing scheme may differ between access network providers, the data packets should have enough information to enable correct forwarding.

The mobility services hosted by distributed computing platforms shall be accessible through available access networks, including 4G/5G and WLAN.

4.2.2 Computing platform

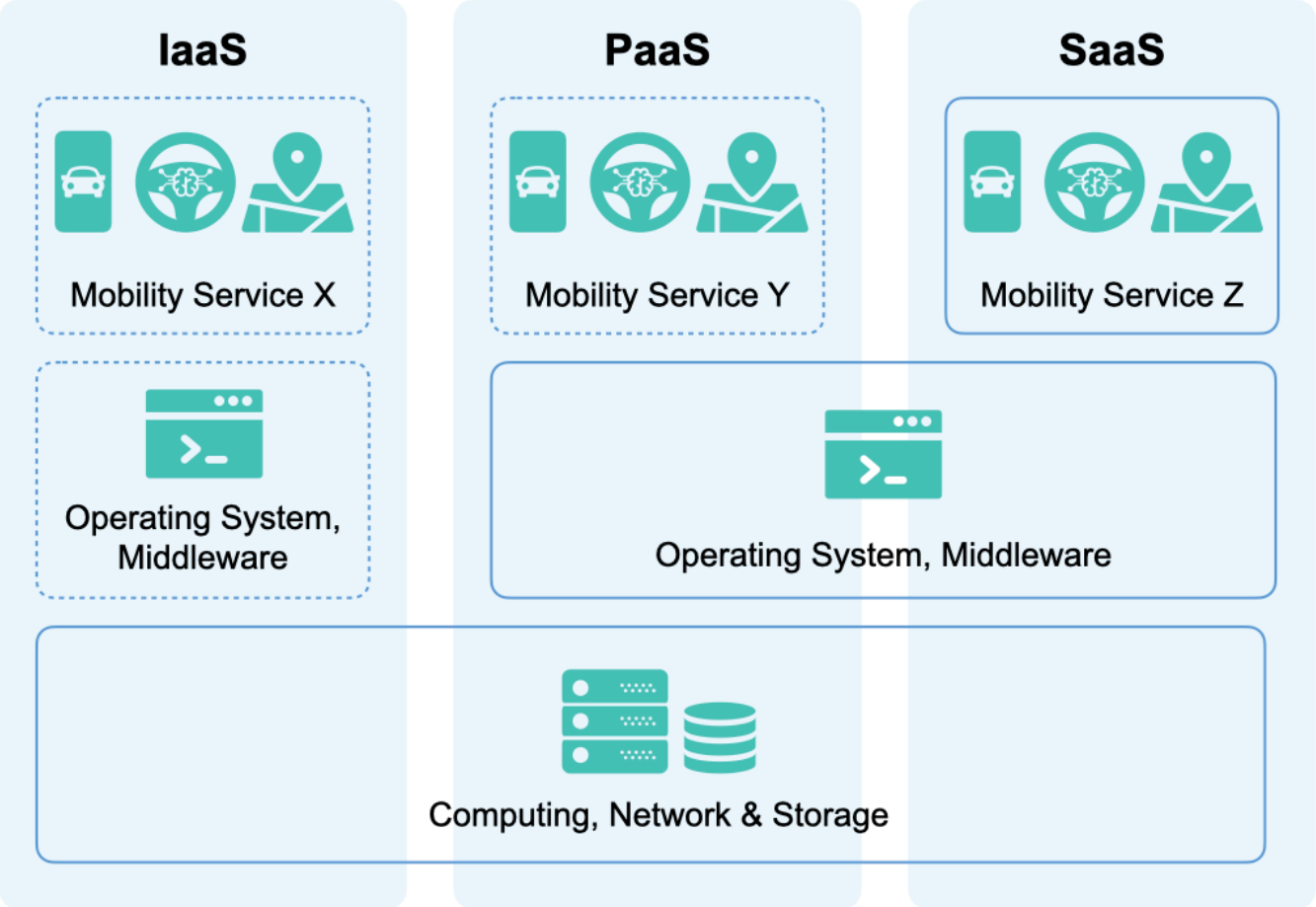


Figure 4: Different types of computing platforms

Current computing platforms offer their services in the form of IaaS, PaaS, or SaaS. Mobility service providers may choose to combine the usage of these types of computing platforms to deploy their services. Aside from the different stacks that are offered, these platforms may also vary with respect to their configuration, such as the availability of a GPU or the network bandwidth. To enable mobility service instances to be appropriately deployed, each computing platform must be able to expose its capabilities and availability to the mobility service providers.

A computing platform exposes computing infrastructure services that enable control and use of the computing platform within an AECC system. The AECC service uses these services to deploy mobility services onto the computing platforms of the AECC system.

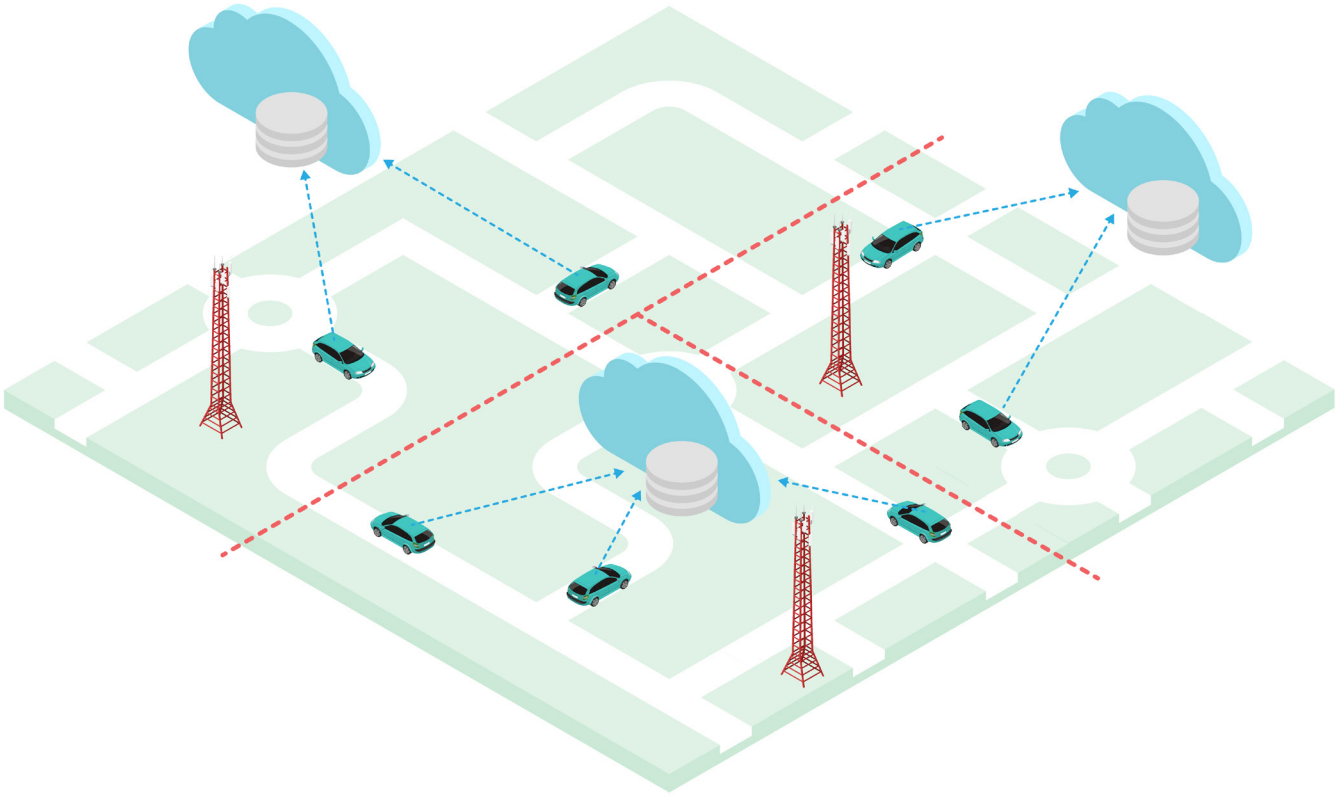


Figure 5: Illustration of the geo-distributed nature of edge computing

Edge computing distributes a service to computing platforms that satisfy service requirements. These may stem from legal requirements on where data is stored or processed. These requirements may be related to negotiated service levels associated with factors such as performance, reliability, or service. For any of these reasons, the geographic location of a service may be an element of interest.

Geographic location is not the only factor. The ability of a service to provide an appropriate level of service must also be considered. For example, the geographically closest service instance to a vehicle may not be the one that is shortest with respect to network transport distance (i.e., latency), as shown in Figure 5; thus, appropriate mobility service instance selection is required.

The AECC system shall enable mobility services instances to be deployed on multiple computing platforms.

The AECC system shall expose the capability, availability, and the geographical region of computing infrastructures to enable deployment of mobility service instances.

4.2.3 **Mobility Services**

Mobility services provide functionality to mobile end users. In the case of the AECC, the primary focus is on vehicles. However, this involvement might be indirect. For instance, roadside units counting cars at an intersection provide information to traffic management applications. Other examples include high-definition mapping, intelligent driving assistance, mobile billing, entertainment and more.

A mobility service is composed of mobility service instances. Each mobility service instance may be deployed and distributed, and each instance may not be aware of how other instances in the same mobility service are

deployed. Each mobility service instance may be distributed in the form of a virtual machine package, a container, or a software package.

The computing environment in the AECC system may be heterogeneous, which means it includes different types of computing platforms. One challenge is to enable a deployment and orchestration scheme that works across different computing platform providers. Each mobility service should provide enough information to enable deployment on the appropriate computing platform.

A vehicle may access several mobility services instances at any given time. At times there is a need to identify clients accessing multiple mobility services, such as for data aggregation. An access credential and client identifier that are acceptable across different mobility services while still enabling security will be of great leverage in enabling this scenario.

The AECC system shall have features to enable deployment, orchestration, and termination of mobility services on different stacks of computing platforms.

The AECC system shall expose available mobility service offerings.

4.2.4 **Distributed data routing**

The AECC system shall enable appropriate selection of mobility service instances for a vehicle service that wishes to consume a mobility service based on service requirements that may include geo-fencing, load distribution, etc.

Mobility services must be prepared to handle changes in the point of presence used by a vehicle system. Such changes might require migration of connections between a vehicle system and an instance of a mobility service to another instance of that mobility service. In such cases, the AECC system should facilitate this migration with minimal loss of service.

The AECC system shall support the ability to share the geographic region supported by mobility service instances and the geographic locations of supporting computing infrastructures.

5 Functional architecture

5.1 AECC system

An AECC system consists of several key elements, as shown in Figure 1: AECC system overview. These are enterprise networks, access networks, computing infrastructure services, mobility services and an AECC service, as described below.

5.1.1 Enterprise networks

The physical and virtual connections between the computing platforms and the users of an AECC system are made through networks using various technologies and often using internet protocol (IP).

5.1.2 AECC service

The AECC service performs administrative operations to orchestrate available infrastructure resources. To create a platform to support the mobility service scenarios, computing infrastructure providers with appropriate resources will expose information about their resources to an AECC service. The AECC service provides an implementation-agnostic method for using the capabilities of the infrastructure to manage the lifecycle of applications and other services. Such information may include computing availability, storage capacity, network functions and so on. In addition, the information may include the concept of “location.” The AECC service may expose this information in the form of service offerings via a service exposure interface.

Using information provided via the AECC service and from other sources, mobility service providers can determine where to place mobility service instances to meet customer requirements: for example, clients running within connected vehicles, and other mobility services and external customers from outside the AECC system.

When applications are instantiated on a computing platform, information can be exposed to the AECC service (see Figure 6). Such information may include the set of applications currently available, the state of the applications and so on. The AECC service should provide information about instantiated applications, including their supported interfaces.

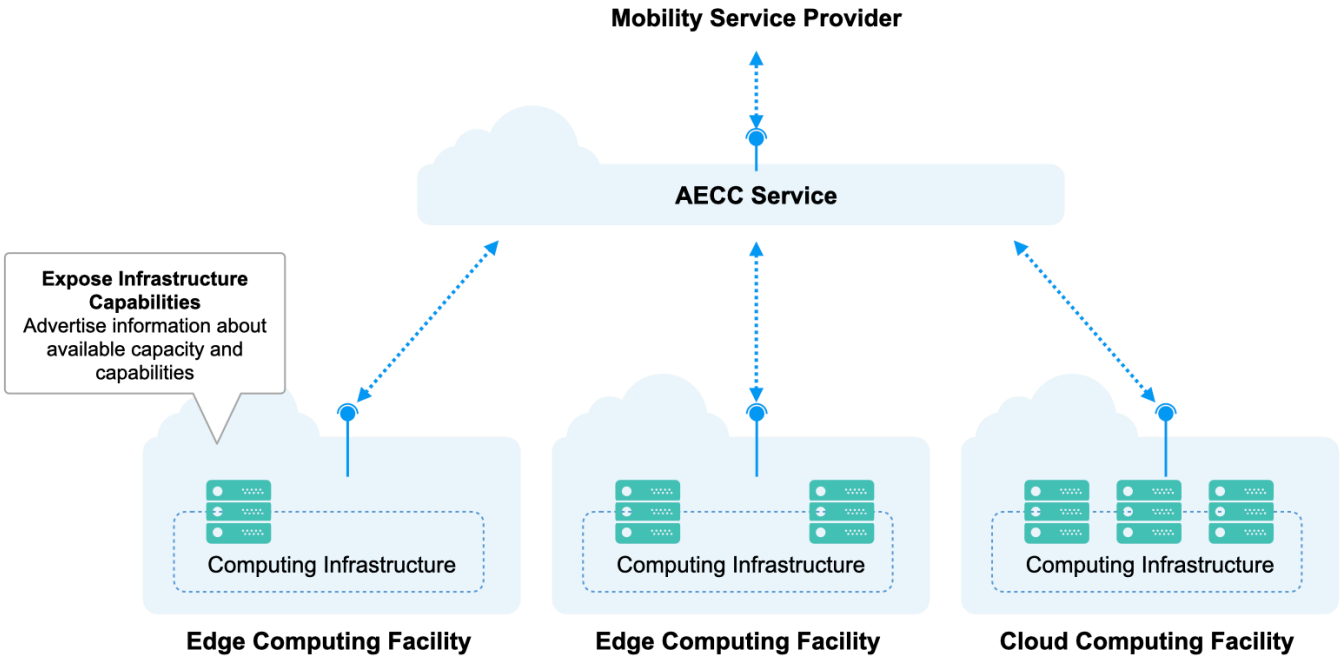


Figure 6: Infrastructure capabilities

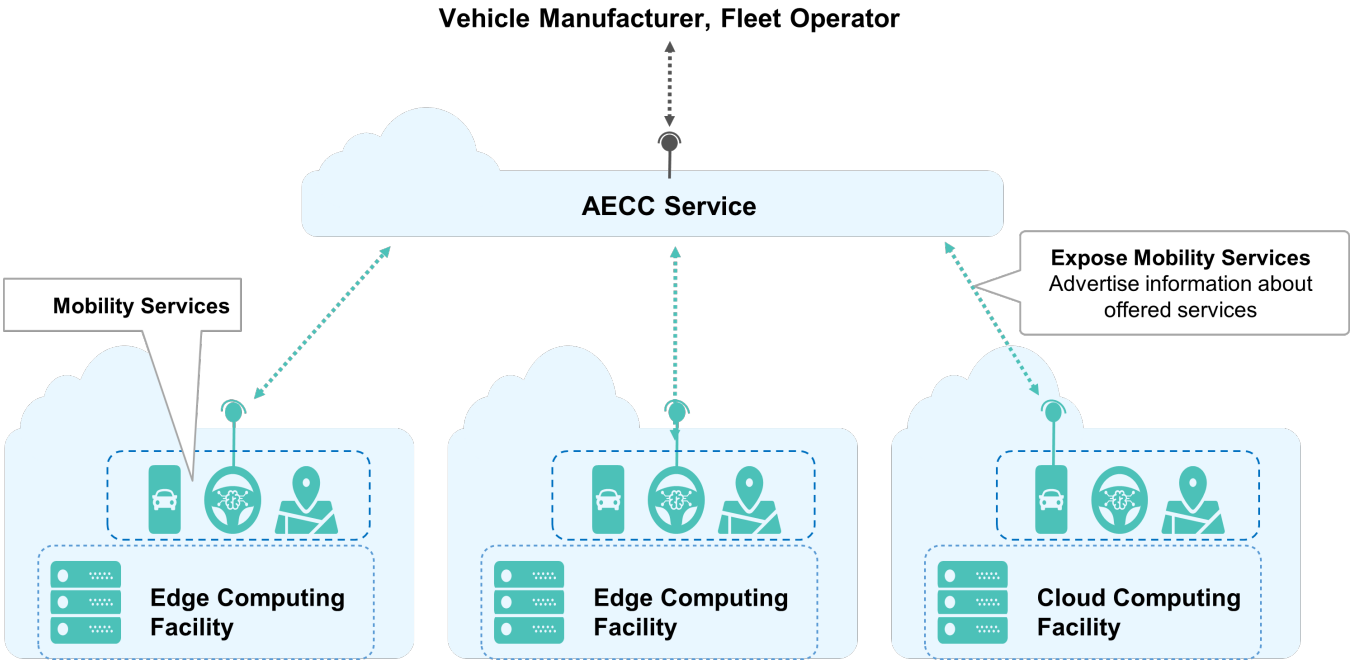


Figure 7: Mobility services

Mobility services are realized using software applications (see Figure 7)) and may be composed of multiple component applications such as microservices. A mobility service may be designed and implemented such that, when instantiated, it runs on multiple computing platforms across multiple computing facilities. When applications are instantiated, information about their component application instances will be exposed to the AECC service (see Figure 8). Information about the mobility service instances may be provided to consumers via the AECC

service. Such information may include the set of mobility services that are available, the state of each mobility service, etc. The AECC service is not intended to provide application-specific interfaces.

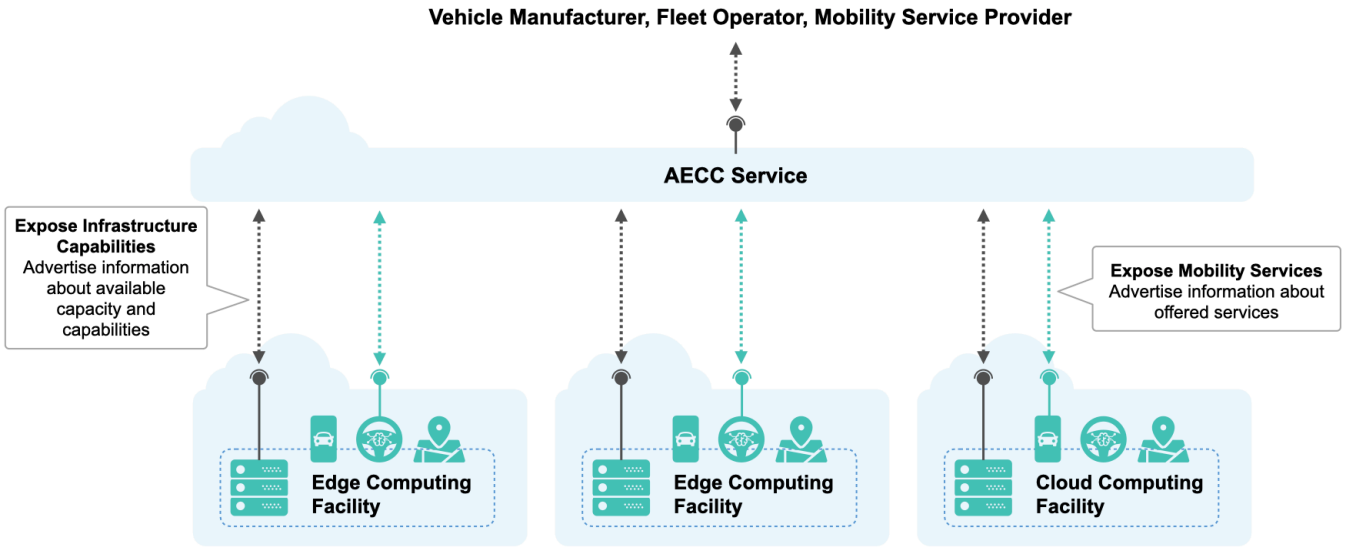


Figure 8: Combined view of all stakeholder systems

The AECC service provides a view of the available resources (computing, network, storage) and the set of mobility services (intelligent driving, high-definition mapping, etc.) that are leveraging those resources.

6 Existing distributed computing solutions

6.1 3GPP

3GPP (the 3rd Generation Partnership Project) is directly addressing edge computing, especially in Technical Specification (TS) 23.501 (Clause 5.13)[1]. The SA2 WG specified the architecture for 5G systems, in which a set of new functional enablers is given for the integration of edge computing in 5G networks.

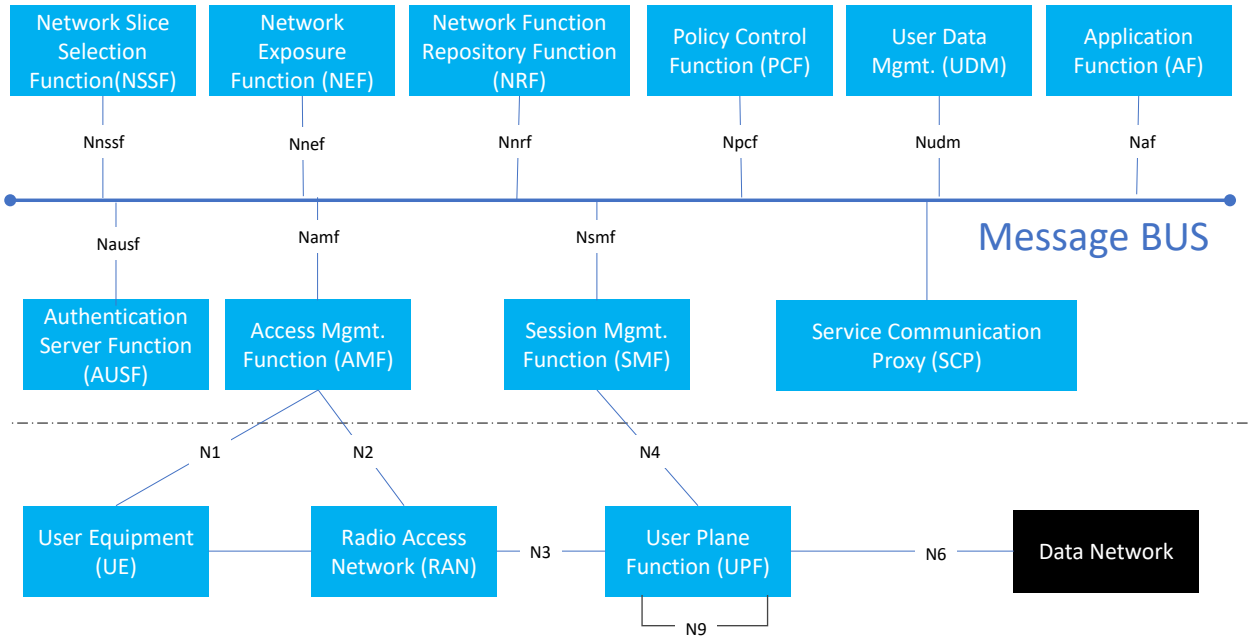


Figure 9: 5G system architecture

Edge computing enables operators and third-party services to be hosted close to the UE's access point of attachment, to achieve an efficient service delivery through reduced end-to-end latency and load on the transport network.

Note: edge computing typically applies to non-roaming and LBO roaming scenarios.

Edge computing can be supported by one or a combination of the following enablers:

- User plane (re)selection: the 5G core network (re)selects UPF to route the user traffic to the local data network.
- Local routing and traffic steering: the 5G core network selects the traffic to be routed to the applications in the local data network.

[1] © 2019 - 3GPP™ deliverables and material are the property of ARIB, ATIS, CCSA, ETSI,TSDSI, TTA and TTC who jointly own the copyright in them. They may be subject to further modifications and are therefore provided to you "as is" for information purposes only. Further use is strictly prohibited.

  - This includes the use of a single PDU Session with multiple PDU session anchor(s) (UL CL/IP v6 multi-homing).
- Session and service continuity to enable UE and application mobility.
- An application function may influence UPF (re)selection and traffic routing via the PCF or NEF.
- Network capability exposure: 5G core network and application function to provide information to each other via the NEF or directly.
- QoS and charging: PCF provides rules for QoS control and charging for the traffic routed to the local DN.
- Support of local area data network: 5G core network provides support to connect to the LADN in a certain area.

The 3GPP SA6 WG also specified an application layer architecture for enabling edge applications over 3GPP networks in Technical Specification (TS) 23.558[2].

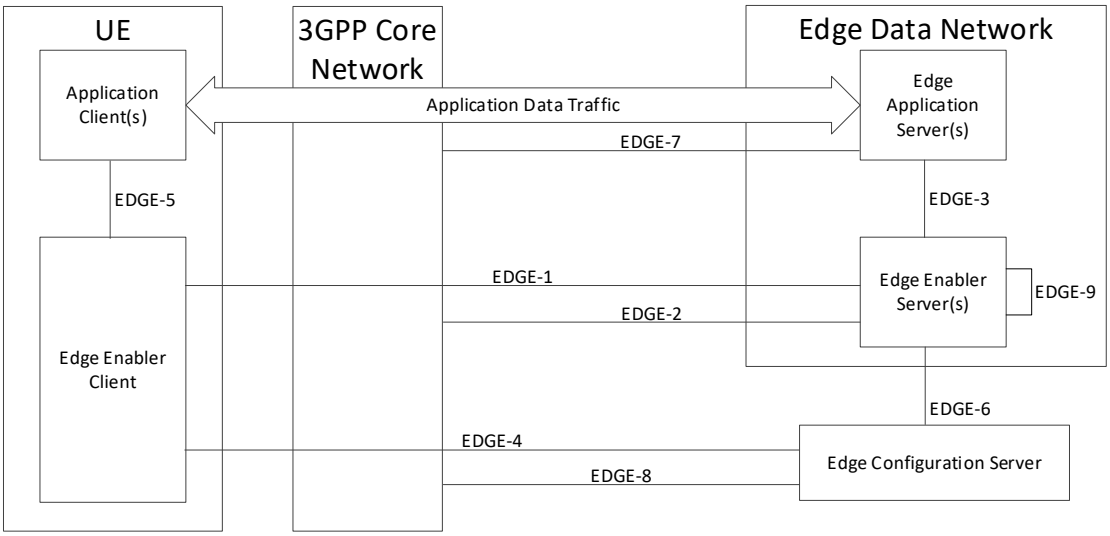


Figure 10: Architecture for enabling edge applications

The edge data network is a local data network. Edge application server(s) and the edge enabler server are contained within the EDN. The edge configuration server provides configurations related to the EES, including details of the edge data network hosting the EES. The UE contains application client(s) and the edge enabler client. The edge application server(s), the edge enabler server and the edge configuration server may interact with the 3GPP core network.

This application layer architecture with functional entities and reference points (as shown in the figure above) is targeted to fulfill the following technical requirements:

a) Edge configuration data
b) Registration
c) Edge application server discovery
d) Capability exposure to edge application servers

[2] © 2020 - 3GPP™ deliverables and material are the property of ARIB, ATIS, CCSA, ETSI,TSDSI, TTA and TTC who jointly own the copyright in them. They may be subject to further modifications and are therefore provided to you "as is" for information purposes only. Further use is strictly prohibited.

e) Security
f) Subscription service
g) Traffic management
h) Lifecycle management
i) Edge application key performance indicators
j) Service continuity

6.2 5GAA

In 2020 the 5G Automotive Association (5GAA) established a work item, "MEC technology to support automotive services (MEC4AUTO)." Use cases "See Through," "In-vehicle Entertainment," "Intersection Movement Assist," "Vulnerable Road User (VRU)," and "Platooning" were selected as guidance for the work, and their relevance for edge computing was analyzed.

The work item defined a reference architecture [5GAA MEC] (see Figure 11) focused on deployment and interoperability aspects, especially considering the presence of multiple MNOs and car OEMs. The challenge addressed was ensuring interoperability and maintaining a controlled end-to-end QoS, where vehicles served by one MNO need to communicate with MEC applications hosted by another MNO and/or need to access interfaces from its MEC platform, e.g., for edge service consumption purposes.

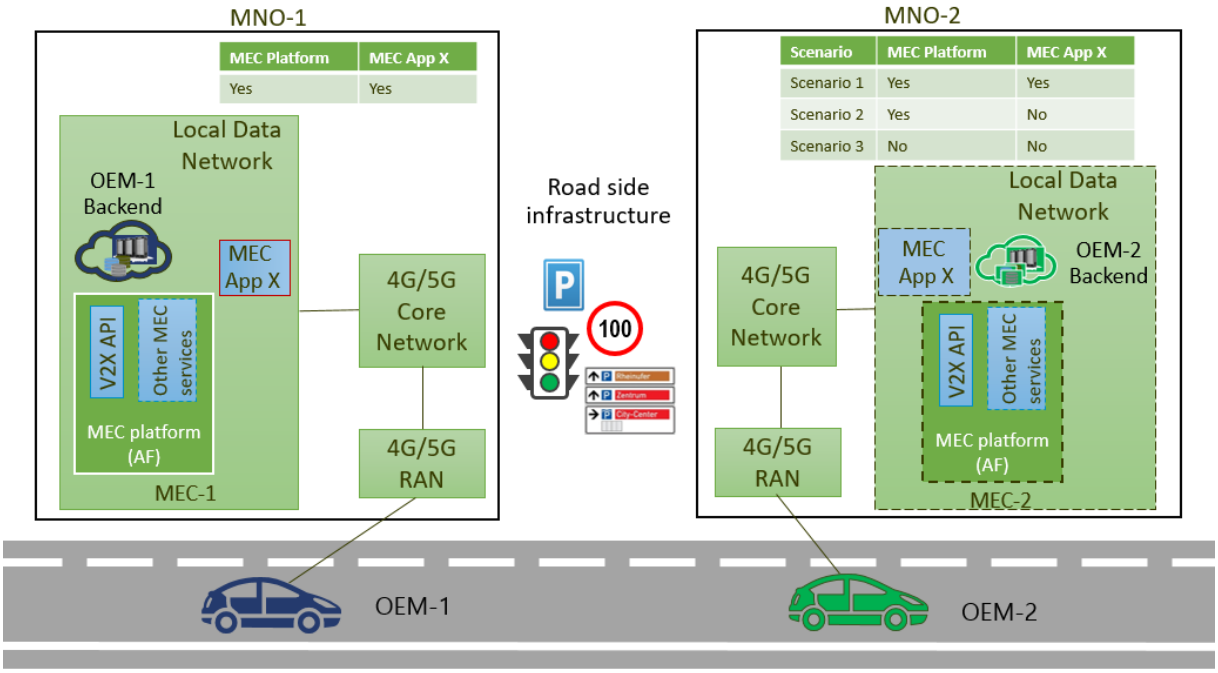


Figure 11: 5GAA MEC reference architecture and multi-MNO scenarios[3]

The 5GAA V2X application layer reference architecture [5GAA V2X] is applied to formalize the description of the selected use cases and define involved stakeholders and the information they exchange.

[3] The source for Figure 11 is Figure 6.1-1 of section 6.1 in the MEC4AUTO reference architecture [5GAA MEC].

The 5GAA MEC reference architecture describes a few options on the usage of wide area networks (WANs) and data centers:

- Moving MEC platforms/applications of different MNOs to a shared data center and assuring controlled connectivity toward the MNOs. This option also includes moving the user plane of the core network to the shared data center.
- Use of WAN links with controlled QoS between MNO data centers to handle cases where multiple MNOs are involved, and vehicles and/or road infrastructure served by different MNOs need to exchange information with controlled end-to-end QoS.

The study also describes how a global management and orchestration system can assure that required MEC applications are instantiated where vehicles that need them are present, regardless of the MNO, and assure that the data traffic is routed accordingly.

In addition, it describes MEC security functions, including "identify," "protect," "detect," "respond," "recover," and "privacy," as well as defining the security boundaries in case more than one MNO is involved. Finally, the study identified the cases beyond the security boundaries, and those that require dedicated cybersecurity solutions.

6.3 ETSI

6.3.1 NFV

NFV (Network Function Virtualization) is defined by ETSI ISG NFV[4]. It is a principle of separating network functions from the hardware they run on. NFV envisages the implementation of NFs as software-only entities that run over the NFV Infrastructure (NFVI).

Figure 12 illustrates the high-level NFV framework.

---

[4] © ETSI 2014. All rights reserved.

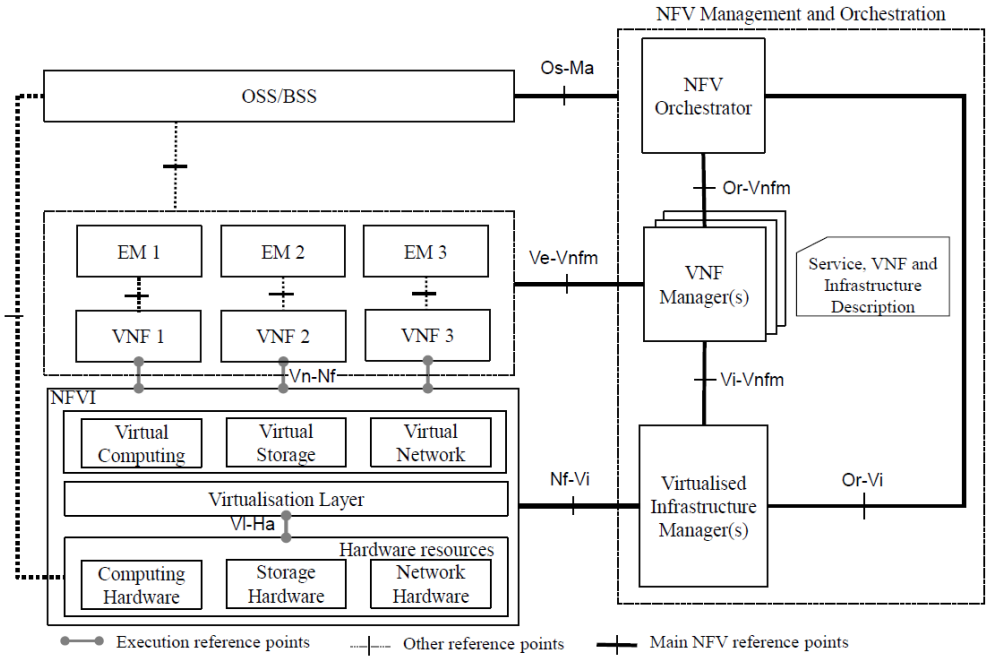


Figure 12: NFV reference architectural framework

The NFV architectural framework, ETSI GS NFV 002, identifies functional blocks and the main reference points between such blocks:

Virtualized Network Function - A VNF is a virtualization of a network function in a legacy non-virtualized network.

Element Management (EM) - The Element Management performs the typical management functionality for one or several VNFs.

NFV Infrastructure - The NFV Infrastructure is the totality of all hardware and software components which build up the environment in which VNFs are deployed, managed, and executed. The NFV Infrastructure can span across several locations, such as places where NFVI-PoPs are operated. The network providing connectivity between these locations is regarded to be part of the NFV Infrastructure.

Virtualized Infrastructure Manager(s) - From NFV's point of view, virtualized infrastructure management comprises the functionalities that are used to control and manage the interaction of a VNF with computing, storage, and network resources under its authority, as well as their virtualization.

NFV Orchestrator - The NFV Orchestrator is in charge of the orchestration and management of NFV infrastructure and software resources and realizing network services on NFVI.

VNF Manager(s) A VNF Manager is responsible for VNF lifecycle management (e.g. instantiation, update, query, scaling, termination). Multiple VNF Managers may be deployed; a VNF Manager may be deployed for each VNF, or a VNF Manager may serve multiple VNFs.

Service, VNF and Infrastructure Description - This dataset provides information regarding the VNF deployment template, VNF Forwarding Graph, service-related information, and NFV infrastructure information models. These templates/descriptors are used internally within NFV Management and Orchestration. The NFV Management and Orchestration functional blocks handle information contained in the templates/descriptors and may expose (subsets of) such information to applicable functional blocks, as needed.

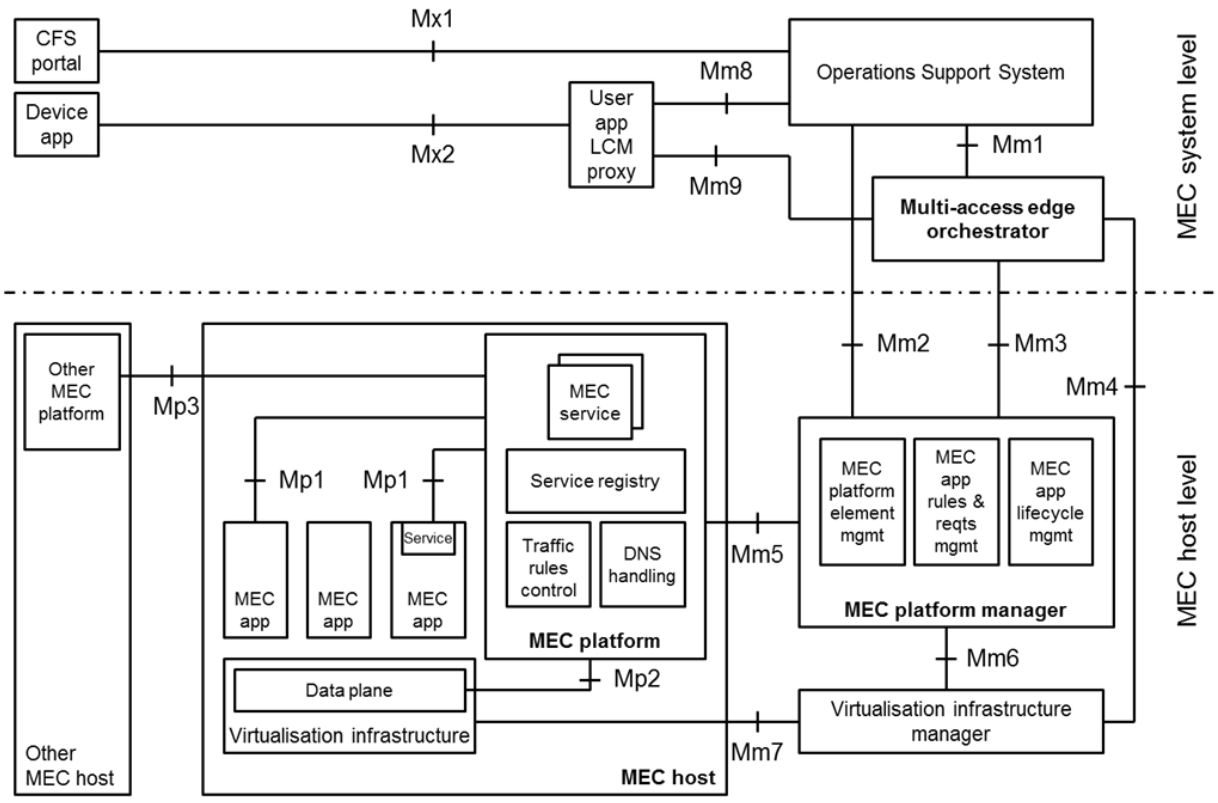
Operations Support Systems and Business Support Systems - (OSS/BSS) in *Figure 12* refers to the OSS/BSS of an Operator.

6.3.2 **MEC**

The MEC (multi-access edge computing) system reference architecture is defined in the ETSI (European Telecommunications Standards Institute) Industry Specification Group (ISG) MEC Group Specification (GS) MEC 003[5]. The reference architecture shows the functional elements that comprise the multi-access edge system and the reference points between them.

Figure 13 depicts the generic multi-access edge system reference architecture. There are three groups of reference points defined between the system entities:

- Reference points for connections to the MEC platform (Mp)
- Reference points for connections to external elements (Mx)
- Reference points for connections to manager elements (Mm)



[5] © ETSI 2020. All rights reserved.

Figure 13: Multi-access edge system reference architecture

The multi-access edge system consists of the MEC hosts and the MEC management necessary to run MEC applications within an operator network or a subset of an operator network.

The **MEC host** is an entity that contains a MEC platform and a virtualization infrastructure that provides computing, storage, and network resources, for the purpose of running MEC applications.

The **MEC platform** is the collection of essential functionalities required to run MEC applications on a virtualization infrastructure and enable them to provide and consume MEC services. The MEC platform can also provide services.

MEC applications are instantiated on the virtualization infrastructure of the MEC host based on configuration or requests validated by the MEC management.

The MEC management comprises the MEC system level management and the MEC host level management.

The MEC system level management includes the **multi-access edge orchestrator** as its core component, which has an overview of the complete MEC system.

The MEC host-level management comprises the **MEC platform manager** and the **virtualization infrastructure manager** and handles the management of the MEC-specific functionality of a particular MEC host and the applications running on it.

A MEC service is a service provided and consumed either by the MEC platform or a MEC application. When provided by an application, it can be registered in the list of services to the MEC platform over the Mp1 reference point. A MEC application can subscribe to a service for which it is authorized over the Mp1 reference point. A certain number of MEC services are necessary to fulfil the requirements defined in ETSI GS MEC 002 and are described below:

- Radio network information service, when available, provides authorized applications with radio network related information.
- Location service, when available, provides authorized applications with location-related information.
- Traffic management services, including:
  - Bandwidth management (BWM) service
  - Multi-access traffic steering (MTS) service

Several key concepts have been provided based on ETSI MEC architecture to fulfill ETSI MEC technical requirements:

- MEC host selection
- DNS support
- Application traffic filtering and routing
- Support of application and UE mobility
- Data plane
- API gateway support

Multi-access Edge Computing (MEC) and Network Functions Virtualization (NFV) are complementary concepts. There is also an architecture variant for MEC in NFV that allows to instantiate MEC applications and NFV virtualized network functions on the same virtualization infrastructure, and to reuse ETSI NFV MANO components to fulfill a part of the MEC management and orchestration tasks. Please refer to ETSI GS MEC 003 for the multi-

access edge system reference architecture for the deployment in a Network Functions Virtualization (NFV) environment.

6.4 IETF distributed overlay for connected vehicles

The LISP model [RFC6830bis] enables overlay networking with logical addressing, for private networks, application-specific networks, balanced and roaming networks, and functional network virtualization.

The Locator/ID Separation Protocol (LISP) splits current IP addresses in two different namespaces: end-point identifiers (EIDs) that identify end-hosts, and routing locators (RLOCs) that identify network attachment points.

LISP uses a map-and-encapsulate approach that relies on:

1. A mapping system (distributed database) that stores and disseminates EID-RLOC mappings
2. LISP tunnel routers (xTRs) that encapsulate and decapsulate data packets based on the content of those mappings

These namespaces effectively separate control from data and allow routers to create overlay networks. LISP-capable routers exchange encapsulated packets according to EID-to-RLOC mappings stored in a local mapping cache.

The draft LISP-NEXAGON RFC uses logical addresses as edge anchor indexes for the roads and curbs, and for ephemeral vehicular publishers and subscribers. The model assumes that logic is pervasive in both vehicles and the edge network, and that the routed indexes can serve simultaneously as:

- EID-indexed content delivery network for vision and sensory data uploaded by vehicles and roadside infrastructure
- EID-indexed multicast listener discovery protocol-subscribed feeds or channels of enumerated attributes per each road tile
- Ledger of what content is stored per which EID road tile by which vehicle per each timestamp

The draft LISP-NEXAGON RFC document specifies a geospatial indexing system using a hexagonal grid of cells with hierarchical subdivisions called H3. H3 cells come in 15 different resolutions designated H3.r1 through H3.r15. Each finer resolution has cells with one seventh of the area of the coarser resolution. H3.r15 represents a hexagon cell of one square meter. (For reference, the area covered by an H3.r9 cell is equivalent to about 20 lane miles of road, and an H3.r1 cell would cover the earth.) Hexagons cannot be perfectly subdivided into seven hexagons, so the finer cells are only approximately contained within a parent cell. Each H3 cell is identified by a 64-bit Hexagon ID (HID).

The draft LISP-NEXAGON RFC document uses LISP to publish, subscribe and ledger the real-time state and status of public spaces and public roads.

These standards are combined to create an in-network state that reflects the condition of each H3.r15 hexagonal tile (~1sqm) in every road. The LISP network maps and encapsulates traffic between mobility client endpoint identifiers (EID), and tile objects (HID=>EID). Tile objects are aggregated by H3ServiceEIDs.

The H3-LISP mobility network bridges timing and location gaps between the production and consumption of information by mobility clients.

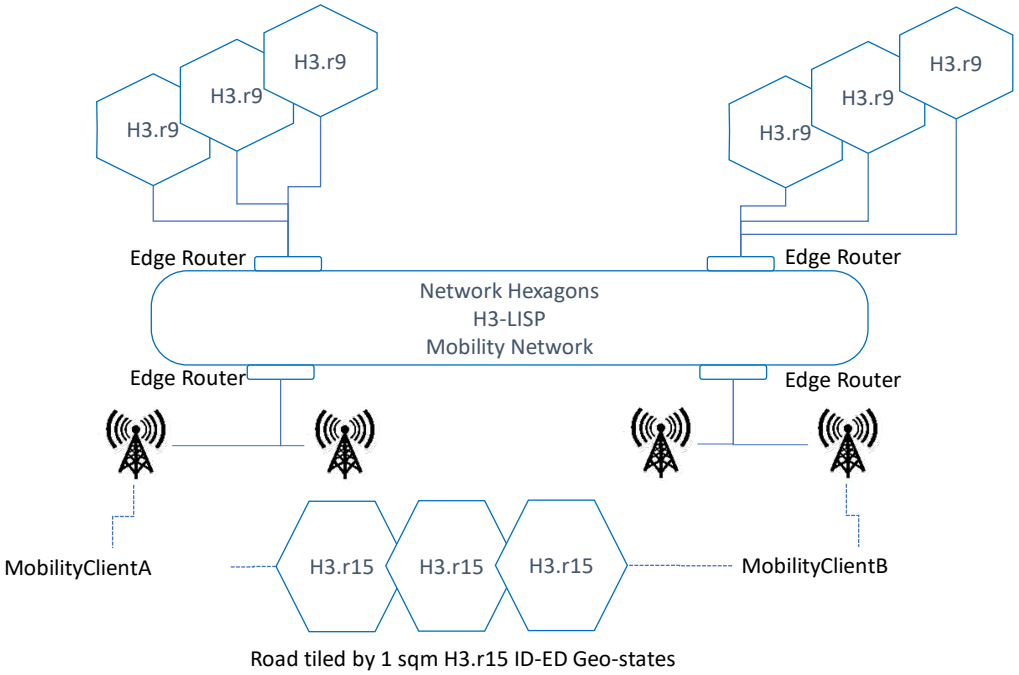


Figure 14: Shared addressable state grid

6.5 Linux Foundation Edge

Linux Foundation (LF) Edge is an open-source community that consists of several LF projects on edge computing. It leverages both specifications and running code, aiming at facilitating integration and interoperability of these projects across various vertical industries. Participating companies represent industries that include manufacturing, energy, transportation, retail, home automation, automotive, and health care. LF Edge tries to create a common framework for hardware and software and identifies best practices essential to supporting current and next generations of edge devices.

One important project related to the AECC is the Akraino™ (A LF Edge project) Edge Stack. It aims to create an open-source software stack that supports high-availability cloud services optimized for edge computing systems and applications. To support end-to-end edge solutions from the Akraino community, Akraino uses a blueprint concept to address specific edge use cases. Some of the blueprints are Network Cloud & Radio Edge, Connected Vehicle, 5G & MEC, AI/ML and AR/VR Applications at the Edge. The Connected Vehicle Blueprint (CVB) demonstrates a connected vehicle application run on edge computing. The overall architecture of this blueprint is illustrated in Figure 15 [see Akraino wiki], which consists of the following key components:

- Commodity hardware, ARM/X86 physical server
- Virtualization layer
- TARS™ microservice platform
- Connected vehicle application layers

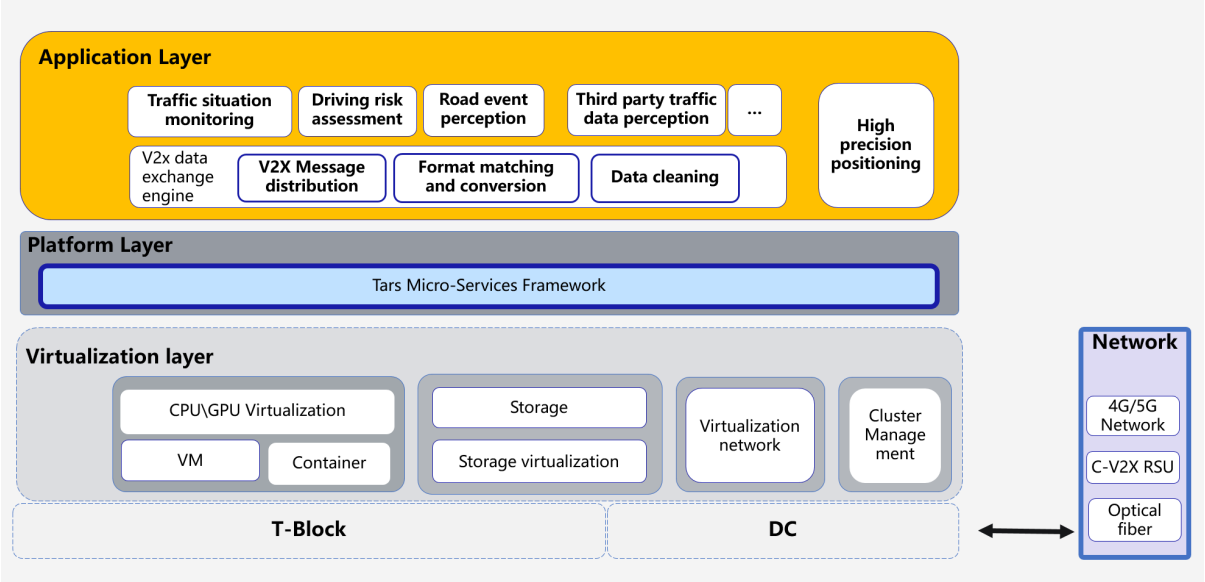


Figure 15: Architecture of Connected Vehicle Blueprint in Akraino Edge Stack[6]

The combination of commodity hardware and IaaS software provides flexible deployments, such as Bare Metal and Virtual Machine as well as Container.

TARS is a microservice framework that can manage/monitor/deploy the connected vehicle applications in the edge and data center. TARS can be flexibly deployed in Bare Metal and Virtual Machine as well as Container.

Connected vehicle applications can provide accurate location information, smarter navigation, and safe driving improvements; these help to reduce traffic violations and enable cooperative vehicle infrastructure systems.

6.6 Intel® Smart Edge Open

The Intel® Smart Edge Open (formerly known as OpenNESS: Open Network Edge Services Software) is an open software toolkit (hereafter called the toolkit) that enables highly optimized and performant edge platforms to on-board and manage applications and network functions with cloud like agility across any type of network. The toolkit follows an “edge native” paradigm, with a base platform and a variety of building blocks that provide for edge application lifecycle management, high performance networking, and numerous optimizations for hardware platforms and accelerators frequently used in edge platforms.

The toolkit can be used to deploy AECC systems either by starting with pre-validated Experience Kits that integrate networking, computing, and controller functionality, or by using individual building blocks in an existing AECC system deployment.

---

[6] This graphic and related text in this section are from the Akraino community’s Connected Vehicle Blueprint Release 4 Documents: Architecture Doc, *available at:* https://wiki.akraino.org/display/AK/CVB+Release+4+Architecture+Doc (“Akraino Document”). © 2020 Tao Wang and contributors. The Akraino Document is made available under the Creative Commons Attribution 4.0 license, *available at:* https://creativecommons.org/licenses/by/4.0/legalcode. Tao Wang and contributors to the Akraino Document disclaim all warranties relating to and all liability arising from the Akraino Document in accordance with Section 5 of the license.

The toolkit system architecture is shown in

Figure 16: System architecture. It depicts a typical edge node, on which edge applications execute, and a controller, which is used to deploy and manage edge nodes. An edge cluster consists of one or more edge nodes. Multiple edge nodes may be associated with one controller.

The edge nodes and controller contain a variety of functions to support the basic functionality of the system, to deploy accelerators, and to manage the lifecycles of edge applications.

The toolkit is based on Kubernetes, and supports containers and Kubernetes pods natively. Virtual machines for applications are supported, but this is done via KubeVirt. In

Figure 16: System architecture, system building blocks integrate closely with Kubernetes and provide the fundamental framework for edge computing infrastructure services and applications. The system building blocks are shown in two pods in the edge node – the system pod and container networking pod as well as two building blocks in the controller.

The container networking pod implements the container network interface (CNI) plug-ins that allow the building blocks in the cluster to communicate with each other.

The platform building blocks are deployed in one or more platform pods. These building blocks are used to support computing and networking accelerators.

In the controller, the toolkit building blocks work with the edge node building blocks, discovering information from the edge node, and making orchestration decisions based on that information. The information received includes capabilities and attributes, as well as telemetry information.

Edge applications and services run in application service pods. They are developed by third parties and are on-boarded and instantiated from the control plane.

The on-premises and network functions block represents the access network, which may be either a cellular network or a non-3GPP network. The mobile network and the edge platform cooperate to steer traffic from the network to edge applications.

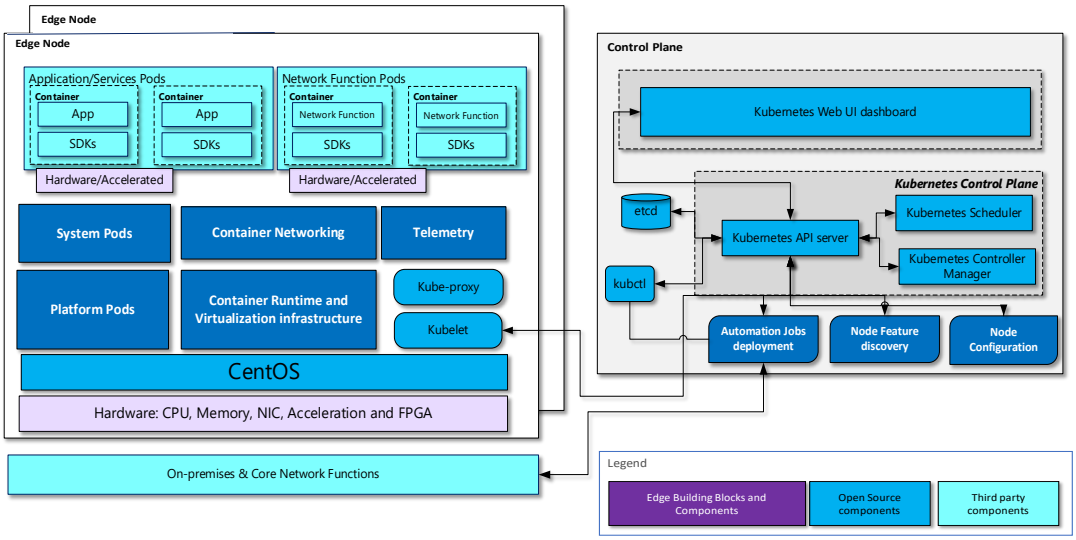


Figure 16: System architecture

6.7 Profiling of existing distributed computing solutions

From Section 6.1 through Section 6.6, we introduced organizations that contribute to the creation of a distributed computing architecture framework for automotive use cases. Each organization works on different layers of the architecture. The IETF and 3GPP provide underlying infrastructure specifications. The ETSI NFV and MEC provides an abstraction of the overall architecture and specifies application programming interfaces (APIs). LF Edge provides deployment blueprints that may leverage technologies, such as the way it is implemented in Intel® Smart Edge Open. Meanwhile, consortia such as the 5GAA and the AECC are focusing on how to leverage existing standards and technology in a business scenario.

Although each organization works on different layers and produces different kinds of outputs, in essence they produce or follow a particular architecture. In Table 1, we analyze whether organizations listed in the previous sections have fulfilled the requirements as mentioned in Section 4.2. We focus only on the architecture and specifications that each organization produces/follows.

Table 1: Available technology and known gaps

	Requirement	Existing Standards/Technology	Gaps
	Access Network		
1	The mobility services hosted by distributed computing platform shall be accessible through available access networks, including 4G/5G and WLAN.	3GPP supports edge data offloading for 4G as well as 5G.	
	Computing Platform		
2	The AECC system shall enable mobility services instances to be deployed on multiple computing platforms.	- Intel® Smart Edge Open supports multi-node computing orchestration. - ETSI NFV specifies interfaces for management and orchestration of VNFs. - ETSI MEC specifies an interface to communicate between edge nodes.	
3	The AECC system shall expose the capability, availability, and the geographical region of computing infrastructures to enable deployment of mobility service instances.	- Intel® Smart Edge Open has edge nodes and controllers that contain a variety of functions to support the basic functionality of the system. This architecture is also leveraged by LF Edge to enable deployment. - ETSI NFV provides orchestration functionality via NFV-MANO. - Within ETSI MEC, the MEC platform manager and the virtualization infrastructure manager handle the management of the MEC-specific functionality.	

	Requirement	Existing Standards/Technology	Gaps
	Mobility Service Instance Orchestration		
4	The AECC system shall have features to enable deployment, orchestration, and termination of mobility services on different stacks of computing platforms.	- LF Edge and Intel® Smart Edge Open implement their systems using containers and microservices, which works for different stacks of computing platforms. - ETSI NFV and MEC architectures have flexibility for application deployment on different kinds of edge platforms.	
5	The AECC system shall expose available mobility service offerings.	- Intel® Smart Edge Open uses containers, where typically orchestration can be configured.	There is no current standard or well-known method to understand mobility service offerings for geographically distributed cloud computing.

	Requirement	Existing Standards/Technology	Gaps
	Distributed Data Routing		
6	The AECC system shall enable appropriate selection of mobility service instances for a vehicle service that wishes to consume a mobility service based on service requirements that may include geo-fencing, load distribution, etc.	- ETSI MEC has defined a host selection function. - LISP specified by the IETF provides a routed overlay network across computing infrastructure providers. - Edge application server discovery in 3GPP SA6 and local routing and traffic steering in 3GPP SA2 are used to enable appropriate selections of mobility services.	
7	In cases where multiple access networks (e.g., network providers) supporting the AECC system are available, the mobility services must be prepared to handle changes in the point of presence, including transfer session and service information, when used by a vehicle system. Such changes might require migration of connections between a vehicle system and one instance of a mobility service to another instance of that mobility service. In such cases, the AECC system should facilitate this migration with minimal loss of service.	- LISP enables encapsulation of state/computation while the client roams between locations/carriers, and addressable mobility services, which migrate between servers and edge locations. Addressable services facilitate low-latency (resolution-less, connectionless) vehicle-to-edge exchanges. - Session and service continuity enable UE and application mobility in 3GPP SA2 and service continuity in 3GPP SA6. This includes point-to-point (P2P) sessions between clients and addressable services, as well as P2MP sessions needed to facilitate bulk urgent updates over any network.	There are no known formal recommendations on interfaces or methods to allow state transfers. State transfers are specified in the context of data management durability.
8	The AECC system shall support the ability to share the geographic regions supported by mobility service instances and the geographic locations of supporting computing infrastructures.	- A draft IETF request for comment [LISP] that uses LISP in conjunction with geospatial indexing H3 to create an abstraction of logical addresses as interfaces for physical road objects hosted as data anchors at the edge. Road objects can support geo-services such as transient road conditions. - Addressable object interfaces allow for interoperable stability while enterprise-specific methods for ingestion, curation and propagation continue to evolve and differentiate.	This method has not been formally introduced for use in distributed computing.

7 Next steps

This whitepaper provides an overview of distributed computing and how it can be applied in the automotive industry. The document illustrates service and architecture level requirements as well as a functional architecture for distributed computing in the context of automotive edge computing. It also provides an initial solution profiling to analyze existing technologies from other standards organizations and open source communities in order to identify candidate solutions to satisfy AECC requirements, and capture potential gaps.

The work is still underway to further detail AECC requirements and solution profiling that can enable this AECC distributed computing functional architecture.

The AECC will also continue to fine tune the distributed computing functional architecture and enhance the technical solutions to fix any gaps and fulfil AECC architectural and functional requirements in conjunction with the other organizations' existing solutions.

8 Bibliography

- [3GPP TS 23.501] V16.3.0, System architecture for the 5G System, https://www.3gpp.org/ftp/Specs/archive/23_series/23.501/23501-g30.zip
- [3GPP TS 23.558] V1.2.0, Architecture for enabling Edge Applications, https://www.3gpp.org/ftp/Specs/archive/23_series/23.558/23558-120.zip
- [5GAA MEC] "MEC for Automotive in Multi-Operator Scenarios," 2021, https://5gaa.org/news/mec-for-automotive-in-multi-operator-scenarios/
- [5GAA V2X] "V2X Application Layer Reference Architecture," 2020, https://5gaa.org/news/v2x-application-layer-reference-architecture/
- [AECC HD Map] Operational Behavior of a High Definition Map Application, Version 1.0.0, May 26, 2020, https://aecc.org/resources/publications/
- [AECC WP] Automotive Edge Computing Consortium, "General Principle and Vision White Paper Version 3.0," p. 12, 2020 https://aecc.org/resources/publications/
- [Akraino wiki] Connected Vehicle Blueprint Release 4 Documents: Architecture Doc, https://wiki.akraino.org/display/AK/CVB+Release+4+Architecture+Doc
- [GDPR] General Data Protection Regulation (EU) 2016/679, https://gdpr-info.eu/
- [ETSI GS MEC] 003 V2.2.1, Multi-access Edge Computing (MEC); Framework and Reference Architecture, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.02.01_60/gs_MEC003v020201p.pdf
- [ETSI GS MEC] 002 V2.1.1, Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/02.01.01_60/gs_MEC002v020101p.pdf
- [ETSI GS NFV] 002 V1.2.1 (2014) Network Functions Virtualisation (NFV); Architectural Framework, https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf
- [IPWAVE] IPv6 Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases, July 29, 2020, draft-ietf-ipwave-vehicular-networking-19
- [Kubernetes] The Kubernetes API, https://kubernetes.io/docs/concepts/overview/kubernetes-api/
- [LISP-Nexagon] Barkai, S., Fernandez-Ruiz, B., Tamir, R., Rodriguez-Natal, A., Maino, F., Cabellos-Aparicio, A, Paillissé Vilanova, J., and D. Farinacci, https://datatracker.ietf.org/doc/html/draft-ietf-lisp-nexagon
- [PCI DSS] PCI Data Security Standards, v3.2.1, May 2018, https://www.pcisecuritystandards.org
- [PII] Guidance on the Protection of Personal Identifiable Information, https://www.dol.gov/general/ppii
- [LISP] The Locator/ID Separation Protocol (LISP), draft-ietf-lisp-rfc6830bis-36, 10 March 2021, https://datatracker.ietf.org/doc/draft-ietf-lisp-rfc6830bis/ [datatracker.ietf.org]

- [RFC6830bis] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)," March 5, 2020, https://tools.ietf.org/html/draft-ietf-lisp-rfc6830bis-32

9 Abbreviations

Abbreviation	Definition
DN	Data Network
EDN	Edge Data Network
EES	Edge Enabler Server
FaaS	Function as a Service
IaaS	Infrastructure as a Service
LADN	Local Area Data Network
LBO	Local Break Out (roaming)
MaaS	Mobility as a Service
MSP	Mobility Service Provider
NEF	Network Exposure Function
PaaS	Platform as a Service
PCF	Policy Control Function
PII	Personal Identifiable Information
QoS	Quality of Service
SA	Service and System Aspects
SaaS	Software as a Service
UE	User Equipment
UL CL	Uplink Classifier
UPF	User Plane Function
WG	Working Group
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network

10 Terms and definitions

The following common industry or dictionary terms should be used when applicable within the AECC community.

AECC service
A service that provides configuration and control of components of an AECC system.

AECC system
A logical system composed of one or more AECC services, mobility services, computing platforms and the networks that connect them to Vehicle Systems and other clients.

Note: requirements of AECC use cases placed on an AECC system may be met by one or more of its services.

Computing facility
A physical facility that houses computing infrastructure.

Computing infrastructure
The resources and services on which systems and services are built.

Note: this may include but is not limited to power, cooling, computing, network, and storage.

Computing infrastructure service
A service that exposes functionality provided by a computing infrastructure.

Computing platform
The environment in which data is processed.

Note: this refers to the hardware or the operating system (OS), even a web browser and associated application programming interfaces, or other underlying software, provided the application code is executed with it.

Connected vehicle
A network-attached vehicle that shares data with other network-attached devices and servers.

Data center
A dedicated facility used to house computer systems and associated components, such as telecommunications and storage systems.

Distributed computing
Computing that divides an application into many tasks that can be served by many computers.

Driver
An individual or an autonomous service that operates the vehicle system.

Function as a service
A category of cloud computing services that provides a platform allowing customers to develop, run, and manage application functionalities without the complexity of building and maintaining the infrastructure typically associated with developing and launching an app.[7]

Mobility as a service
The integration of various forms of services into a mobility service that is accessible on demand.

Mobility service
A service provided to the passengers or driver of a vehicle (e.g., telematics, traffic, map, car/ride sharing, insurance).

Note: this may be a service embedded in or external to a system.

Mobility service provider
A platform-independent provider that provides customers with one or more mobility services.

Note: the abbreviated term is MSP.

Personal identifiable information
The representation of information that permits the identity of an individual to whom the information applies to be reasonably inferred by either direct or indirect means.

Note: see reference PII.

Wireless fidelity or Wi-Fi
The radio wireless local area networking of devices based on the IEEE 802.11 standards.

---

[7] https://en.wikipedia.org/wiki/Function_as_a_service

11 Contributors

The following AECC members have contributed to this document.

Company	Member
Dell Technologies	George Ericson Wenlei Wu
Ericsson	Bastian Cellarius Mikael Klein Morgan Lindquist
Intel	Neal Oliver Leifeng Ruan
Nexar, Ltd	Sharon Barkai
NTT Corporation	Lidwina Andarini
Toyota Motor Corporation	Lei Zhong