A Platform for Computing at the Mobile Edge: Joint Solution with HPE, Saguna, and AWS

February 2018

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract

This whitepaper is written for communication service providers with network infrastructure, as well as for application developers and technology suppliers who are exploring applications that can benefit from edge computing. In this paper, we establish the value of a standards-based computing platform at the mobile network edge, describe use-cases that are well-suited for this platform, and present a reference architecture based on the solutions offered by AWS, Saguna, and HPE. A subset of use cases are reviewed in detail that illustrate how the reference architecture can be adapted as a platform to serve use case-specific requirements.

Introduction

Imagine a world where cars can alert drivers about dangerous road conditions to help them take action to avoid collision, and where devices can help fleets of cars drive autonomously and predict traffic patterns. Consider a new Industrial Revolution where Internet of Things (IoT) devices or sensors report data collected in real time from large and small machines, allowing for intelligent automation and orchestration in industries such as manufacturing, agriculture, healthcare, and logistics. Envision city and public services that provide intelligent parking, congestion management, pollution detection and mitigation, emergency response, and security. While this is happening, internet users access bandwidth of 10 times the current maximums and latencies at 1/100th of current averages, using a seamless combination of mobile, WiFi, and fixed access. Fifth-generation mobile network (5G) applications are enabling these scenarios by providing 10 times the current bandwidth maximum and 1.

This new generation of applications is fueling technological developments and creating new business opportunities for mobile operators. One such technological *and* business development, which is key to enabling many new generation of applications, is “edge computing.” Edge computing addresses the latency requirements of specialized 5G applications, helps manage the potentially exorbitant access cost and network load due to fast-growing data demand, and supports data localization where necessary. By providing a cloud-enabled platform for edge computing, mobile operators are well positioned to take a leading role in the 5G ecosystem, while opening up completely new business cases and revenue streams.

This whitepaper presents a solution that allows you to leverage the infrastructure of your existing mobile networks and establish a platform to enable new revenue-generating applications and 5G use cases.

The Business Case for Multi-Access Edge Computing

Multi-Access Edge Computing (MEC) is a cloud-based IT service environment at the edge infrastructure of networks that serves multiple channels of telecommunications access, for example, mobile-wide-area networks, Wi-Fi or LTE-based local-area-networks, and wireline.

In this section, we discuss the many benefits of a MEC platform that sits at the edge of the cellular mobile network.

MEC Addresses the Need for Localized Cloud Services

Agility, scalability, elasticity, and cost efficiencies of cloud computing have made it the platform of choice for application development and delivery. IoT applications need local cloud services that operate close to connected devices to improve the economics of telemetry data processing to minimize latency for time-critical applications and to ensure that sensitive information is protected locally.

MEC Leverages the Capabilities Inherent in Mobile Networks

Mobile networks have expanded to the point where they offer coverage in most countries around the world. These networks combine wireless access, broadband capacity, and security.

MEC Provides a Standards-Based Solution that Enables an Ecosystem of Edge Applications

MEC transforms mobile communication networks into distributed cloud computing platforms that operate at the mobile access network. Strategically located in proximity to end users and connected devices, MEC enables mobile operators to open their networks to new, differentiated services while providing application developers and content providers access to Edge Cloud benefits.

The ETSI MEC Industry Specification Group (ISG) has defined the first set of standardized APIs and services for MEC. The standard is supported by a wide range of industry participants, including leading mobile operators and industry vendors. Both HPE and Saguna are active members in the ETSI ISG.

In the following sections, we outline the key benefits provided by MEC.

Extremely Low Latency

Traditional internet-based cloud environments have physical limitations that prohibit you from hosting applications that require extremely low latency. Alternatively, MEC provides a low-latency cloud computing environment for edge applications by operating close to end users and connected IoT devices.

Broadband Delivery

Video content is typically delivered using TCP streams. When network latency is compounded by congestion, users experience annoying delays due to the drop in bitrate. The MEC environment provides low latency and minimal jitter, which creates a broadband highway for streaming at high bitrates.

Economical and Scalable

In massive IoT uses cases, many devices such as sensors or cameras send vast amounts of data upstream, which current backhaul networks[1] cannot support. MEC provides a cloud computing environment at the network edge where IoT data can be aggregated and processed locally, thus significantly reducing upstream data. MEC infrastructure can scale as you grow by expanding local capacity or by deploying additional edge clouds in new locations.

Privacy and Security

By deploying the MEC Edge Cloud locally, you can ensure that your private data stays on premises. However, unlike server-based on-premises installations, MEC is a fully automated edge cloud environment with centralized management.

Role of MEC in 5G

MEC enables ultra low-latency use cases specified as part of the 5G network goals. MEC also enables fast delivery of data and the connection of billions of devices, while allowing for cost economization related to transporting enormous volumes of data from user devices and IoT over the backhaul network.

It is important to note that MEC is currently deployed in 4G networks. By deploying this standard-based technology in existing networks, communication service providers can benefit from MEC today while creating an evolutionary path to their next-generation 5G network.

Mobile Edge Solution Overview

Saguna has developed a MEC virtualized radio access network (vRAN) solution that runs on Hewlett Packard Enterprise (HPE) edge infrastructure. This solution lets application developers create mobile edge applications using AWS services, while allowing mobile operators to effectively deploy MEC and operate edge applications within their mobile network.

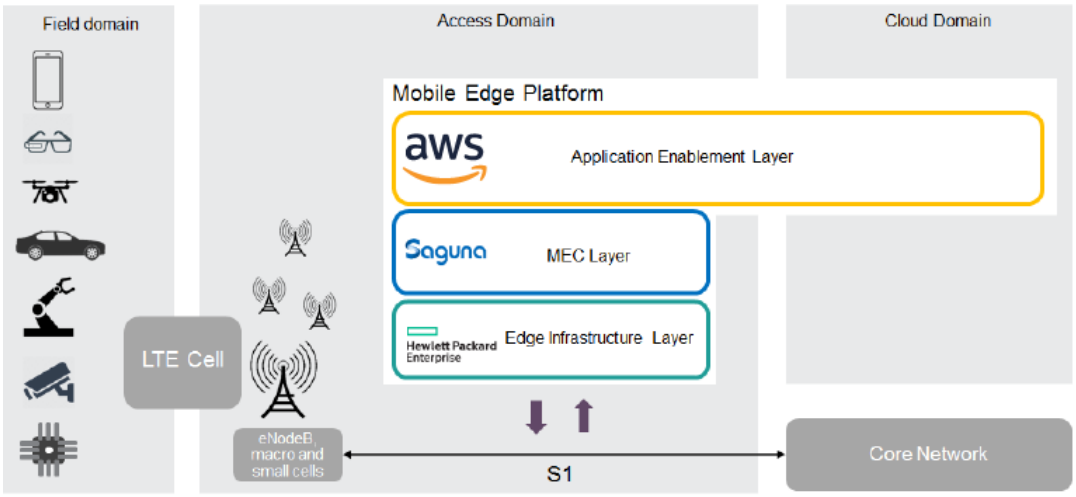

Figure 1: End-to-end MEC solution architecture

The proposed mobile edge solution consists of three main layers as illustrated in Figure 1:

- **Edge Infrastructure Layer** – Based on the powerful x86 compute platform, this layer provides compute, storage, and networking resources at edge locations. It supports a wide range of deployment options from RAN-based station sites to backhaul aggregation sites and regional branch offices.
- **MEC Layer** – This layer lets you place an application within a mobile access network and provides a number of services including mobile traffic breakout and steering, registration and certification services for applications deployed at the edge, and radio network information services. It also provides optional integration points with mobile core network services such as charging and lawful intercept.

- **Application Enablement Layer** – This layer provides tools and frameworks to build, deploy, and maintain edge-assisted applications. This layer allows you to place certain application modules locally at the edge (e.g., latency-critical or bandwidth-hungry components) while keeping other application functions in the cloud.

The flexible design inherent in the MEC solution architecture allows you to scale the edge component to fit the needs of concrete use cases. You can deploy the edge component at the deepest edge of mobile network (e.g., co-located with eNodeB equipment at a RAN site), which lets you to deploy low-latency and bandwidth-demanding application components in close proximity to end devices. You can also deploy an edge component at any traffic aggregation point between a base station and mobile core, which allows you to serve traffic from multiple base stations.

The proposed mobile edge platform provides a variety of tools to build, deploy, and manage edge-assisted applications such as:

- Development libraries and frameworks spanning edge-to-cloud, including function-as-a-service at the edge and cloud, AI frameworks for creating and training models in the cloud, seamless deployment and inference at the edge, and communication brokerage between edge application services and cloud. These development libraries and frameworks expose well-defined APIs and have been widely adopted in the developer community, shortening the learning curve and accelerating time-to-market for edge-assisted applications and use cases.
- Tools to automate deployment and life-cycle management of edge application components throughout massively distributed edge infrastructure.
- Infrastructure services such as virtual infrastructure services at the edge, traffic steering policies at the edge, DNS services, radio awareness services, integration of edge platform into overall network function virtualization (NFV) framework of mobile operator.
- Diverse compute resources, fitted to the particular needs of edge application such as CPU, GPU for acceleration of graphics-intensive or AI workloads, FPGA accelerators, cryptographic and data compression accelerators, etc.

This unique combination of functionalities lets you quickly develop edge applications, deploy and manage edge infrastructure and applications at scale, and lets you achieve a fast time-to-market with edge-enabled use cases.

Example Reference Architectures for Edge Applications

A mobile edge platform enables new application behaviors. By adding the ability to run certain components and application logic at the mobile network edge in close proximity to the user devices/clients, the mobile edge platform allows you to re-engineer the functional split between client and application servers and enables a new generation of application experiences.

The following list provides examples of possible mobile edge computing applications in industrial, automotive, public, and consumer domains:

- Industrial
  - Next-generation augmented reality (AR) wearables (e.g., smart glasses)
  - IoT for automation, predictive maintenance
  - Asset tracking
- Automotive
  - Driverless cars
  - Connected vehicle-to-vehicle or vehicle-to-infrastructure (V2X)
- Smart Cities
  - Surveillance cameras
  - Smart parking
  - Emergency response management
- Consumer-Enhanced Mobile Broadband
  - Next-generation Augmented Reality/Virtual Reality (AR/VR) and video analytics
  - Social media high-bandwidth media sharing

- Live event streaming
- Gaming

In the following sections, we provide examples of how the mobile edge solution can be implemented for smart city surveillance, AR/VR edge applications, and Connected V2X.

Smart City Surveillance

Cities can take advantage of IoT technologies to increase the safety, security, and overall quality of life for residents and keep operational costs down. For example, video recognition technology enables real-time situational analysis (also called "video as a sensor"), which allows you to detect a variety of objects from video feed (e.g., people, vehicles, personal items), recognize the overall situation (e.g., a traffic jam, fight, trespassing, and abandoned objects), and classify recognized objects (e.g., faces, license plates).

The mobile edge solution enables new abilities in building robust and cost-efficient smart city surveillance systems:

- **Efficient video processing at the edge** – Computer vision systems in general require high-quality video input (especially for extracting advanced attributes) and hardware acceleration of inference models. The mobile edge solution lets you host a computing environment at the network edge. This lets you offload backhaul networks and cloud connectivity from bandwidth-hungry, high-resolution video feeds and allows low-latency actions based on recognition results (e.g., opening gates for recognized vehicles or people, controlling traffic with adaptive traffic lights). The mobile edge platform provides industry-standard GPU resources to accelerate video recognition and any other artificial intelligence (AI) models deployed at the edge.
- **Flexible access network** – End-to-end smart city surveillance systems might leverage different means to generate video input such as existing fixed surveillance cameras, mobile wearable cameras (e.g., for law enforcement services or first responders), and drone-mounted mobile surveillance. The diversity of endpoints generating video input requires a high degree of flexibility from access network – leveraging fixed video networks and mobile cellular networks with native mobility support for wearable or unmanned aerial vehicle (UAV)-mounted

cameras. Additionally, automated drone-mounted systems require low-latency access to control the flight of the drone, which might require end-to-end latencies of millisecond scale. The mobile edge platform provides a means to use robust, low-latency cellular access with native mobility support for the latter cases and incorporates existing fixed video networks.

- **Flexible video recognition models** – Robust video recognition AI models usually require extensive training on sample sets of objects and events, as well as periodic tuning (or development of models for extracting some new attributes). These compute-intensive tasks use highly scalable, lower cost compute cloud resources. However, seamless deployment of the trained models to the edge for execution and managing the life cycle of the deployed models is a complex operational task. The mobile edge platform provides seamless development and operational experience, starting from creation, training, and tuning an AI model in the cloud, to deploying it at edge locations and managing the lifecycle of the deployed models.

The following diagram shows an example architecture of a smart city surveillance edge application:

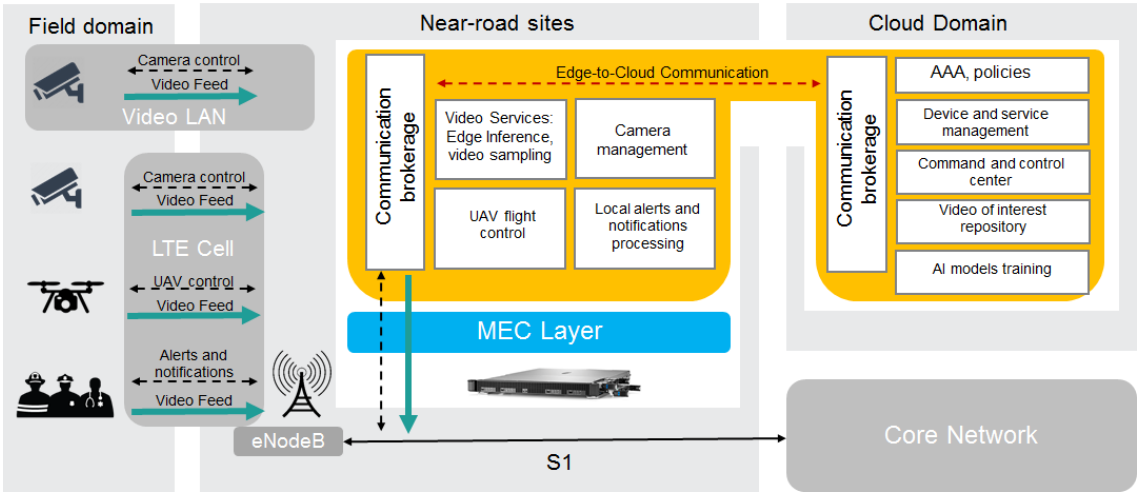

Figure 2: Edge-assisted smart city surveillance application

A smart city surveillance solution has three main domains:

- Field domain – Diverse ecosystem of video-producing devices, e.g., body-worn cameras from first responder units, drones, fixed video

surveillance systems, and wireless fixed cameras. Video feeds are ingested into the mobile edge platform via cellular connectivity and use existing video networks.

- Edge sites – Located in close proximity to the video-generating devices and host latency-sensitive services (e.g., UAV flight control, local alerts processing), bandwidth-hungry, compute-intensive applications (edge inference), and gateway functionalities for video infrastructure control (camera management). Video services extract target attributes from the video streams and share metadata with local alerting services and cloud services. Video services at the edge can also produce low-resolution video proxy or sampling videos for transferring only the videos of interest to the cloud.
- Cloud domain – Hosts centralized, non-latency-critical functions such as device and service management functions, AAA and policies, command and control center functions, as well as compute-intensive non-latency critical tasks of AI model training.

You can augment a MEC smart city surveillance application with machine learning (ML) and inference models via:

- Model training (for surveillance patterns of interest, e.g., facial recognition, person counts, dwell time analysis, heat maps, activity detection) using deep learning AMIs on the AWS Cloud
- Deployment of trained models to the MEC platform's application-container using AWS Greengrass and Amazon SageMaker
- Application of inference logic (e.g., alerts or alarms based on select pattern detection) using AWS Greengrass ML inference

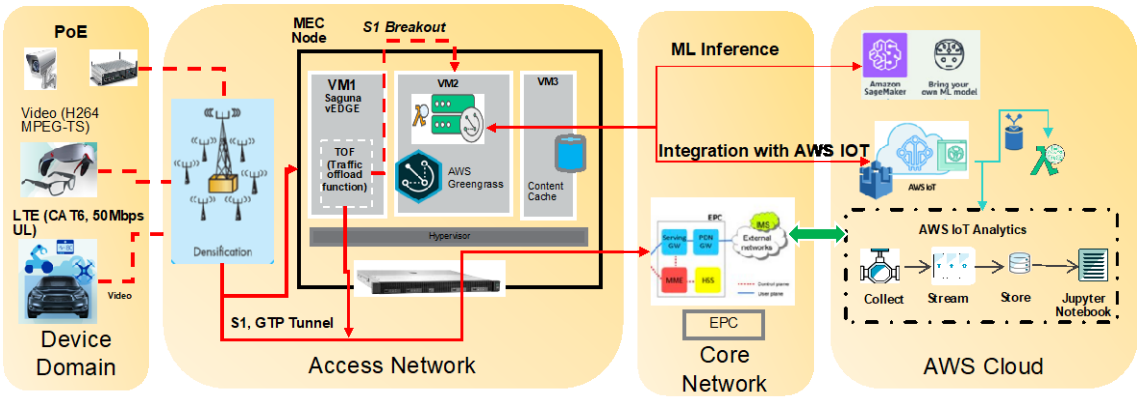

Figure 3: Detailed view of solution for smart city surveillance application

This design approach based on the mobile edge platform is a cost-efficient way of building and operating a smart city surveillance system with edge processing for bandwidth-hungry and latency-sensitive services.

AR/VR Edge Applications

AR/VR is one of the use cases that benefits most from a mobile edge platform. AR/VR edge applications can benefit from the mobile edge platform in the following ways:

- **Next generation AR wearables**

Current immersive AR experiences require heavy processing on the client side (e.g., calculating head and eye position and motion information from tracking sensors, rendering of high-quality 3-D graphics for the AR experience, and running video recognition models). The requirement to run heavy computations on AR devices (e.g., head-mounted displays, smart glasses, smartphones) has influenced the characteristics of these devices—cost, size, weight, battery life, and overall aesthetic appeal.

Figure 4: Next-generation AR devices

You can avoid bulkiness, cost, weight, ergonomic, and aesthetic limitations on the devices by offloading the heaviest computational tasks from the devices to a remote server or cloud. However, a truly immersive AR experience requires keeping coherence between AR content and the surrounding physical world with an end-to-end latency below 10 ms, which is unachievable by offloading to a traditional centralized cloud.

The mobile edge platform provides compute power at the network edge, which allows you to offload latency-critical functions from the AR device to the

network, and enables the next generation of lightweight, compact devices with longer battery life and native mobility.

- **Mission-critical operations**

AR experiences have been valuable in workforce enablement applications with remote collaboration applications, AR-assisted maintenance in the industrial space, etc.

In many cases, those AR experiences have become an important part of mission-critical operations, for example, AR-assisted maintenance of equipment in hazardous conditions (e.g., oil extraction sites, refineries, and mines) and in AR-assisted healthcare. Those use cases require high reliability from the AR application, even when global connectivity from the client to the server side is degraded or broken.

The mobile edge platform provides the capability to re-engineer an AR application in a way that the solution can operate offline, with critical components deployed both locally in close proximity to devices and globally in the cloud as a fallback option.

- **Localized data processing**

In many cases, AR devices combine data from different local sources (e.g., adding live sensor readings from a local piece of equipment to an AR maintenance application). In many cases, ingesting data into the cloud requires high bandwidth and is governed by data security or privacy frameworks. A true AR experience requires localized data processing and ingest.

The mobile edge platform allows you to ingest data from any local source into the AR application, as well as execute commands from the AR application to the local data sources (e.g., perform equipment maintenance tasks).

The following diagram shows an example architecture for an AR edge application.

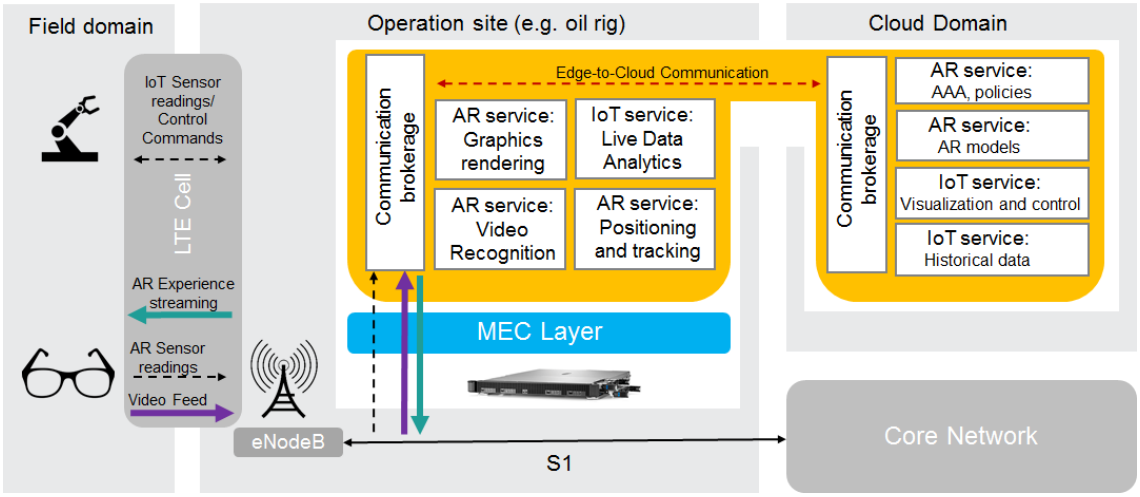

Figure 5: Edge-assisted AR application

The edge-assisted AR application has three main domains:

- Ultra-thin client (e.g., head-mounted display) – Generates sensor readings of head and eye position, location, and other relevant data such as live video feed from embedded cameras.
- Edge services – Part of an AR backend hosted in close proximity to the client on network side. These services execute latency-critical functions (computing positioning and tracking from AR sensor readings, AR graphics rendering), bandwidth-hungry functions (e.g., computer vision models for video recognition), and local data (processing of IoT sensor readings from localized equipment).
- Cloud services – Part of AR backend hosted in a traditional centralized cloud. These services execute functions centralized in nature (e.g., authentication and policies, command and control center, and AR model repository), resource-hungry, non-latency-critical functions (computer vision model training), and horizontal, cross-enterprise functions (e.g., data lakes, integration points with other enterprise systems, etc.).

This design approach allows clients to offload heavy computations, which makes client devices cost-efficient, lightweight, and battery-efficient. This design also allows local data to be ingested from external sources and controls actions to local systems, enables offline operation, saves costs of WAN connectivity, and secures compliance with potential data localization guidelines. By working as an integrated part of the mobile network, this use case natively supports global mobility, telco-grade reliability, and security.

Connected Vehicle (V2X)

Connectivity between vehicles, pedestrians, roadside infrastructure, and other elements in the environment is enabling a tectonic shift in transportation. The full promise of V2X solutions can only be realized with a new generation of mobile edge applications:

- **Transportation safety** – V2X promises the ability to coordinate actions between vehicles sharing the road. (This ability is sometimes called "Cooperative Cruise Control.") Information exchange between connected vehicles about intention to change speed or trajectory can significantly improve the safety and robustness of automated or autonomous driving through cooperative maneurvering. However, due to the very dynamic nature of car traffic, these decisions must be made in near real time (with end-to-end latencies on a millisecond time scale). The massively distributed nature of road infrastructure, near-real-time decision making, and the requirements for high-speed mobility make the mobile edge platform perfect for hosting the distributed logic of cooperative driving.
- **Transportation efficiency** – Cooperative driving promises not only increased safety on the road, but also a significant boost in transportation efficiency. With coordinated vehicle maneuvers, the overall capacity of road infrastructure can increase without significant investment in road reconstruction. The promise of higher transportation efficiency is further supported by vehicle-to-infrastructure solutions. Vehicles can communicate with roadside equipment for speed guidance, to coordinate traffic light changes, and to reserve parking lots. While some information requires only short-range communication (e.g., from a vehicle to a roadside unit), the coordinated actions of a distributed infrastructure (e.g., coordinating traffic light changes between multiple intersections) requires the mobile edge platform to host the logic.
- **Transportation experience** – With autonomous driving technologies, car infotainment systems are becoming more widespread. The mobile edge platform enables the unique possibility of massively distributed content caching with high localization and context-awareness, as well as the ability to enable location and context-based interactions with vehicle passengers (e.g., guidance about local

attractions for travelers, time- and location- limited promotions from local vendors, etc.).

The following diagram shows an example architecture of a V2X edge application.

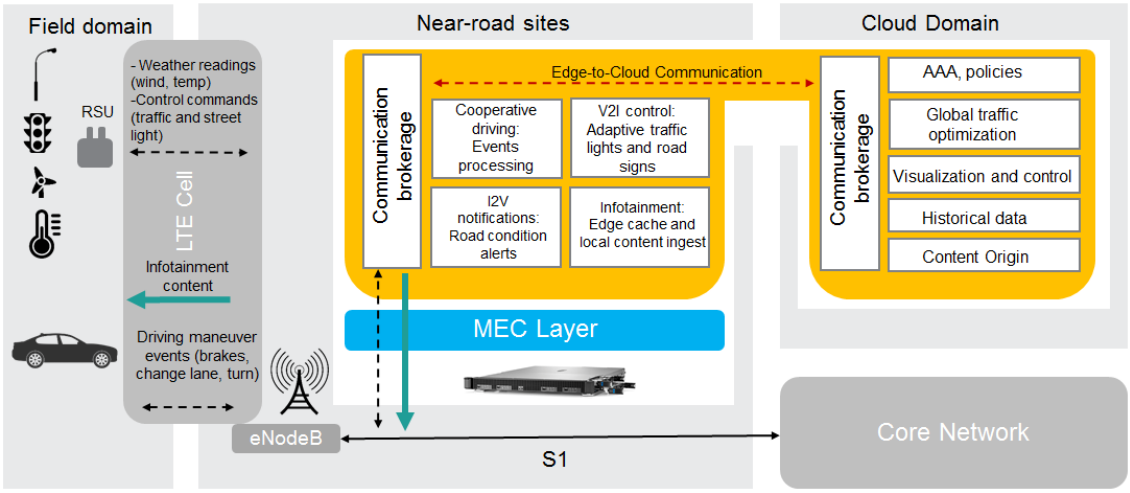

Figure 6: Edge-assisted connected vehible (V2X) application

The V2X solution has three main domains:

- Field domain – Vehicles that generate data about intended driving maneuvers (e.g., braking, lane changes, turns, acceleration) and receive notifications from surrounding vehicles. Road infrastructure that includes all sensors and actuators that are relevant to the driving experience (e.g., wind and temperature sensors, street lighting, connected traffic lights that are controlled via gateway devices such as Road Side Unit).
- Edge sites – Located in close proximity to the road (e.g., respective RAN eNodeB sites) and host latency-sensitive or highly localized V2X application services. Examples of those services include processing and relaying driving maneuver notifications for vehicle coordination, processing local sensor readings from road infrastructure, dynamic generation of control commands to road infrastructure (e.g., coordinated traffic lights across several intersections), and caching highly localized infotainment content.
- Cloud domain – Hosts centralized and non-latency critical functions, such as AAA and policy control, historical data collection and

processing, command and control center functions, and centralized infotainment content origin.

With this design approach, you can realize low-latency and a coordinated exchange of data and control commands between vehicles and surrounding infrastructure. This provides a highly specific context for every interaction.

Conclusion

Many technological and market developments are converging to create an opportunity for new applications that take advantage of modern mobile networks and the edge access infrastructure. This paper emphasizes the need for an application enablement ecosystem approach and presents a platform to serve multiple edge use cases.

Contributors

The following individuals and organizations contributed to this document:

- Shoma Chakravarty, WW Technical Leader, Telecom, Amazon Web Services
- Tim Mattison, Partner Solutions Architect, Amazon Web Services
- Alex Reznik, Enterprise Solution Architect and ETSI MEC Chair, HPE
- Rodion Naurzalin, Lead Architect, Edge Solutions, HPE
- Tally Netzer, Marketing Leader, Saguna
- Danny Frydman, CTO, Saguna

Appendix

This Appendix gives a more detailed overview of the functional components of the proposed mobile edge platform solution, as well as technical characteristics of each component.

Figure 7 illustrates a functional diagram of the mobile edge platform:

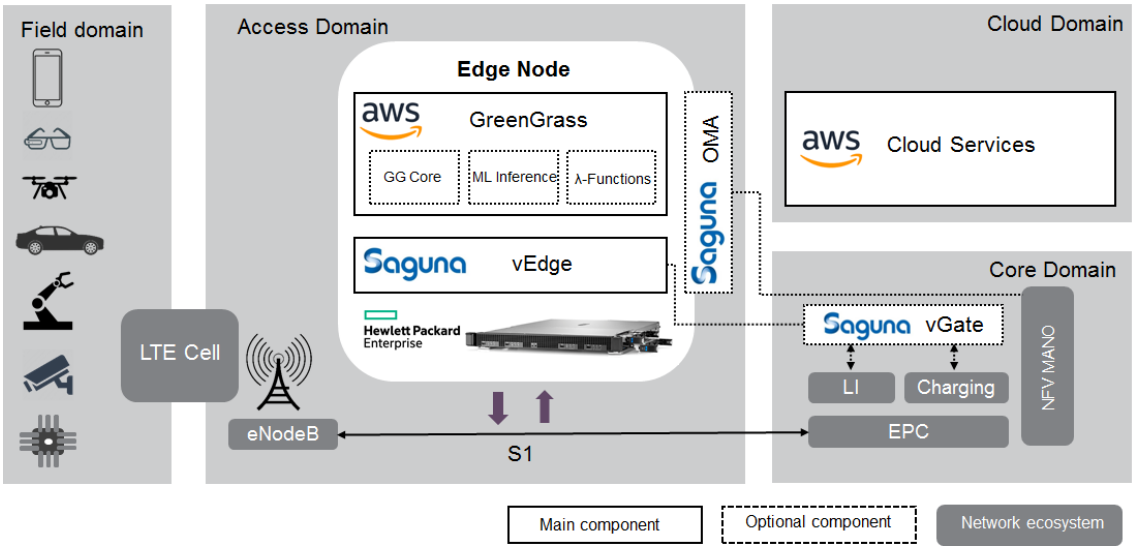

Figure 7: Mobile edge platform functional diagram

Infrastructure Layer

The physical infrastructure for a MEC node is based on an edge-optimized, converged HPE Edgeline EL4000 platform (Figure 8).

Figure 8: HPE Edgeline EL4000 chassis and four m710x cartridges

The end-to-end MEC solution gives you the ability to place workloads within any segment of your mobile access network, for example at a RAN site, backhaul aggregation hub, or CRAN hub. The HPE Edgeline EL4000 has been optimized for the MEC solution as follows:

Compute Density

The Edgeline EL4000 hosts up to four hot-swap SoC cartridges in 1U chassis, providing up to 64 Xeon-D cores with optimized price/core and watt/core characteristics. That design provides 2x – 3x higher compute density compared to a typical traditional data center platform, while keeping power consumption low. These characteristics allow an operator to place a MEC node based on Edgeline EL4000 at the deepest edge of access network down to a RAN site,

where space and power constraints make other general purpose compute platforms inefficient.

Workload-Specific Compute

The diversity of MEC use cases requires that the underlying infrastructure be able to provide different types of compute resources. The Edgeline EL4000 platform provides diverse compute and hardware acceleration capabilities, which allows you to co-locate workloads with different compute needs:

- x86 processors that serve general workloads. Typical workload examples include a Virtual Network Function, virtualized edge application enablement platform, and applications that provide fast control actions at the edge for low-latency use cases.
- Built-in GPU that accelerates graphics processing. Typical workload examples are video transcoding at the edge for MEC-assisted content distribution and 3-D graphics rendering at the edge for AR/VR streaming application.
- Plug-in dedicated GPU cards that accelerate deep learning algorithms. Enabled by strategic partnership with NVIDIA, the Edgeline platform can be used for deep learning hardware acceleration at the edge. Typical workload examples include video analytics and computer vision at the edge and ML inference at the edge for anomaly detection and predictive maintenance.
- Built-in acceleration of cryptographic operations with QuckAssist Technology (e.g., accelerating cryptographic or data compression workloads).
- Support of up to four PCI-E extension slots in a single chassis, which provides options for specialized plug-in units such as dedicated FPGA boards, neuromorphic chips, etc. Such specialized hardware acceleration is being evaluated for many network function workloads (such as RAN baseband processing) and applications (efficient deep learning inference).

Physical and Operational Characteristics

A MEC node should be ready to operate at physical sites and is traditionally used for hosting telco purpose-built appliances that are optimized for the physical site environment (e.g., radio base station equipment at RAN sites,

access routers at traffic hubs, etc.). The operational environment of the MEC node sites may be very different from the traditional data center, with limited physical space for equipment hosting, consumer-grade climate control, and limited physical accessibility. The Edgeline EL4000 is optimized to operate in such environments, with operational characteristics comparable to the telco purpose-built appliances:

Parameter	RAN Baseband Appliance	Typical Data Center Platform	Edgeline EL4000
Operating Temperature (ºC)	+0 …+50	+10 … +35	0 … +55
Non-Destructive Shock Tolerance (G)	30	2	30
Expected Mean Time Between Failures (MTBF) (years)	30-35	10-15	>35

On top of enhanced operational characteristics, the Edgeline EL4000 exposes open iLO interface for the management of highly distributed infrastructure of MEC nodes. The iLO interface is compliant with RedFish industry standard. It exposes infrastructure management functions via simple RESTful service.

Saguna OpenRAN Components Overview

The MEC platform layer is based on the Saguna OpenRAN solution and consists of the following functions:

- Saguna vEdge function, located within MEC node
- Saguna vGate function (optional), located at the core network site
- Saguna OMA function (optional), located within a MEC node or at the aggregation point of several MEC nodes

Saguna vEdge resides in the MEC node and enables services and applications to operate inside the mobile RAN by providing MEC services such as registration and certification, Traffic Offload Function (TOF), real-time Radio Network Information Services (RNIS), and optional DNS services.

The virtualized software node is deployed in the RAN on a server at a RAN site or aggregation point of mobile backhaul traffic. It may serve single or multiple

eNodeB base stations and small-cells. It can easily be extended to support WiFi and other communications standards in heterogeneous network (HetNet) deployments.

Saguna vEdge taps the S1 interface (GTP-U and S1-AP protocols) and steers the traffic to the appropriate local or remote endpoint based on configured policies. Saguna vEdge implements local LTE traffic steering in number of modes (inline steering, breakout, tap).

It has a communication link that connects it to the optional Saguna vGate node using Saguna's OPTP (Open RAN Transport Protocol). It exposes open REST APIs for managing the platform and providing platform services to the MEC-assisted applications.

Saguna vGate is an optional component that resides in the core network. It is responsible for preserving core functionality for RAN-generated traffic: lawful interception (LI), charging, and policy control. The Saguna vGate also enables mobility support for session generated by an MEC-assisted application.

Operating in a virtual machine, Saguna vGate is adjacent to the enhanced packet core (EPC). It has a communication link that connects it to the Saguna vEdge nodes using Saguna's OPTP (Open RAN Transport Protocol) and mobile network integrations for LI and charging functions.

Saguna OMA (Open Management and Automation) is an optional subsystem that resides in the MEC node or at the aggregation point of several MEC nodes. It provides a management layer for the MEC nodes and integrates into the cloud Network Function Virtualization (NFV) environment, which includes the NFV Orchestrator, the Virtual Infrastructure Manager (VIM), and Operations Support Systems (OSS).

Saguna OMA provides two management modules:

- Virtualized Network Function Manager (VNFM) - Provides Life-Cycle-Management and monitoring for MEC Platform (Saguna vEdge) and MEC-assisted applications. This is a standard layer of management required within NFV environments. It resides at the edge to manage the local MEC environment.

- Mobile Edge Platform Manager (MEPM) – Provides an additional layer of management required for operating and prioritizing MEC applications. It is responsible for managing the rules and requirements presented by each MEC application rules and resolving conflicts between different MEC-assisted applications.

The Saguna OMA node operates on a virtual machine and manages on-boarded MEC-assisted applications via its workflow engine using Saguna and third-party plugins. The Saguna OMA is managed via REST API.

Saguna OpenRAN Services

As a MEC platform layer, Saguna OpenRAN provides the following services:

Mobile Network Integration Services

- *Mobility* with Internal Handover support for mobility events between cells connected to the same MEC node and External Handover support between two or more MEC nodes and between cells connected to a MEC node and unconnected cells
- *Lawful Interception (LI)* for RAN-based generated data. It supports X1 (Admin), X2 (IRI), and X3 (CC) interfaces and is pre-integrated with Utimaco and Verint LI systems
- *Charging* support using CDR generation for application-based charging (based on 3GPP TDF-CDR) and charging triggering based on time, session, and data. Supported charging methods are File based (ASN.1) and GTP'
- *Management*, vEdge REST API for MEC services discovery and registration, MEPM, and VNFM let you efficiently operate a MEC solution and integrate it into your existing NFV environment

Edge Services

- *Registration* for MEC-assisted applications. The MEC Registration service provides dynamic registration and certification of MEC applications and registration to other MEC services provided by the MEC Platform, setting the MEC application type.
- *Traffic Offload Function* routes specific traffic flows to the relevant applications as configured by the user. The TOF also handles tunneling

protocols such as GPRS Tunneling Protocol (GTP) for Long Term Evolution (LTE) network, Standard A10/A11 interfaces for 3GPP2 CDMA Network and handles plain IP traffic for WiFi/DSL Network.

- *DNS* provides DNS caching service by storing recent DNS addresses locally to accelerate the mobile internet and DNS server functionality, preconfiguring specific DNS responses for specific domains. This lets the User Equipment (UE) connect to a local application for specific TCP sessions.
- *Radio Network Information Service*, provided per Cell and per Radio Access Bearer (RAB). The service is vendor-independent and can support eNodeBs from multiple RAN vendors simultaneously. It supports standard ETSI queries (e.g., cell info) and notification mechanism (e.g., RAB establishment events). Additional information based on Saguna proprietary model provides real-time feedback on cell congestion level and RAB available throughput using statistical analysis.
- *Instant Messaging* with Short Message Service (SMS) provided as a REST API request. It offers smart messaging capabilities, for example sending SMS to UEs on a specific area (e.g., sports stadium) or sending SMS to UE when entering or exiting a specific area (e.g. shop).

Mobile Edge Applications

- *Throughput guidance application* uses the internal RNIS algorithm to deliver throughput guidance for specific IP addresses on the server side or according to domain names of the servers. The application can be configured with the period of such Throughput Guidance update per target.
- *DDoS Mitigation application* monitors traffic originating from the connected device for specific DDoS attacks on different layers (IP layer for ICMP flooding, IP scanning, Ping of death; TCP/UDP layer for TCP sync attacks, UDP message flooding; Application layer). Devices that are detected as generating DDoS traffic are reported to the network management and traffic from these devices can be locally stopped or the device can be remotely disabled by the network core

Application Enablement Layer

The Application Enablement layer consists of AWS Greengrass hosted on the MEC node side.

AWS Greengrass is designed to support IoT solutions that connect different types of devices with the cloud and each other. It also runs local functions and parts of applications at the network edge. Devices that run Linux and support ARM or x86 architectures can host the AWS Greengrass Core. The AWS Greengrass Core enables the local execution of AWS Lambda code, messaging, data caching, and security.

Devices running the AWS Greengrass Core act as a hub that can communicate with other devices that have the AWS IoT Device SDK installed, such as micro-controller-based devices or large appliances. These AWS Greengrass Core devices and the AWS IoT Device SDK-enabled devices can be configured to communicate with one another in a Greengrass Group. If the AWS Greengrass Core device loses connection to the cloud, devices in the Greengrass Group can continue to communicate with each other over the local network. A Greengrass Group represents localized assembly of devices.For example, it may represent one floor of a building, one truck, or one home.

AWS Greengrass builds on AWS IoT and AWS Lambda, and it can also access other AWS services. It is built for offline operation and greatly simplifies the implementation of local processing. Code running in the field can collect, filter, and aggregate freshly collected data and then push it up to the cloud for long-term storage and further aggregation. Further, code running in the field can also take action very quickly, even in cases where connectivity to the cloud is temporarily unavailable.

AWS Greengrass has two constituent parts: the AWS Greengrass Core and the IoT Device SDK. Both of these components run on on-premises hardware, out in the field.

The **AWS Greengrass Core** is designed to run on devices that have at least 128 MB of memory and an x86 or ARM CPU running at 1 GHz or better, and can take advantage of additional resources if available. It runs Lambda functions locally, interacts with the AWS Cloud, manages security and authentication, and communicates with the other devices under its purview.

The **IoT Device SDK** is used to build the applications on devices connected to the AWS Greengrass Core device (generally via a LAN or other local connection). These applications capture data from sensors, subscribe to MQTT topics, and use AWS IoT device shadows to store and retrieve state information.

AWS Greengrass features include**:**

- **Local support for AWS Lambda** – AWS Greengrass includes support for AWS Lambda and AWS IoT device shadows. With AWS Greengrass, you can run AWS Lambda functions right on the device to execute code quickly.
- **Local support for AWS IoT device shadows** – AWS Greengrass also includes the functionality of AWS IoT device shadows. The device shadow caches the state of your device, like a virtual version or "shadow," and tracks the device's current versus desired state.
- **Local messaging and protocol adapters** – AWS Greengrass enables messaging between devices on a local network so they can communicate with each other even when there is no connection to AWS. With AWS Greengrass, devices can process messages and deliver them to other devices or to AWS IoT based on business rules that the user defines. Devices that communicate via the popular industrial protocol, OPC-UA, are supported by the AWS Greengrass protocol adapter framework and the out-of-the-box OPC-UA protocol module. Additionally, AWS Greengrass provides protocol adapter framework to implement support for custom, legacy, and proprietary protocols.
- **Local resource access** – AWS Lambda functions deployed on an AWS Greengrass Core can access local resources that are attached to the device. This allows you to use serial ports, USB peripherals such as add-on security devices, sensors and actuators, on-board GPUs, or the local file system to quickly access and process local data.
- **Local machine learning inference** – Allows you to locally run an MLmodel that's built and trained in the cloud. With hardware acceleration available in the MEC infrastructure layer, this feature provides a powerful mechanism for solving any machine learning task at the local edge, e.g., discovering patterns in data, building computer vision systems, and running anomaly detection and predictive maintenance algorithms.

AWS Greengrass has a growing list of features. Current features are shown in Figure 9.

Figure 9: AWS Greengrass features

AWS Greengrass on the MEC node acts as a pivot point. It integrates the MEC platform with the AWS IoT solution and other AWS services, providing a powerful application enablement environment for developing, deploying, and managing MEC-assisted applications at scale.

The figure below illustrates the current portfolio of AWS services that enable a seamless IoT pipeline—from endpoints connecting via Amazon FreeRTOS or the IoT SDK through MQTT or OPC-UA, to edge gateways that host AWS Greengrass and Lambda functions providing data-processing capabilities at the edge, up to cloud-hosted AWS IoT Core, AWS Device Management, AWS Device Defender, and AWS IoT Analytics services, as well as enterprise applications.

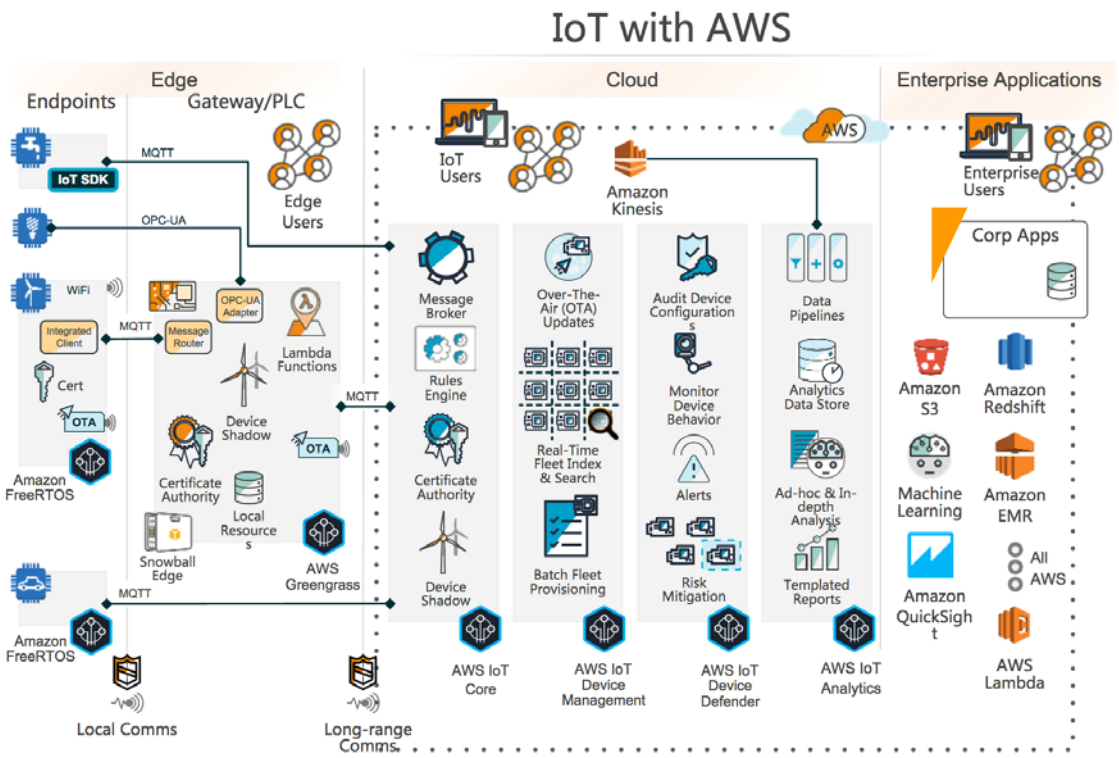

Figure 10: AWS services that enable a seamless IoT pipeline

[1] In a telecommunications network, the **backhaul** portion of the network comprises the intermediate links between the core network, or backbone network, and the small subnetworks at the "edge."