Fog Use Case Scenarios

Use Case: Autonomous Driving
Vertical: Smart Transportation

An OpenFog Consortium Architectural Use Case

1 Snapshot: Fog-Enabled Autonomous Driving

Autonomous Driving can’t be done without fog. Fog establishes trustworthiness of communications between low-level sensors while enabling high-bandwidth real-time processing. Fog interoperability enables hierarchical communication from Vehicle to Vehicle, Vehicle to Infrastructure and Vehicle to Everything. The programmability of the OpenFog Reference Architecture creates a 3rd party supplier ecosystem.

The Autonomy pillar is crucial to Autonomous Driving. Cars can’t realistically maintain constant communications with the cloud or with ground-based resources. But with fog, they can continue to function safely and reliably, even when out of range.

VALUE
What are the business advantages of
building this use case with fog?

Autonomous Driving involves hundreds and hundreds of simultaneous data processes and connections. The business advantage of deploying fog architectures for autonomous cars is that they enable significantly greater scalability than any other architecture. Fog interoperability enables on-board equipment to communicate at a variety of levels while providing standard interfaces that will provide a foundation for the fog ecosystem.

CLOUD & EDGE

How does this use case augment or supersede cloud and edge architectures?

Autonomous cars are a key part of the entire Smart Cities infrastructure of the future. Fog's autonomy and constant lateral communication at multiple layers extends and augments cloud and edge architectures. Fog provides bandwidth-intensive computing at the deepest layers – the dashboard display, Lidar, radio, and main car computation. These are all fog nodes that cooperate with, say, cloud nodes performing analytics, or edge-based roadside cabinets. Additionally, fog security provides a more impenetrable attack surface.

2 Table of Contents

3 Introduction

Note: The preamble section of this document (pages 4 through 11) is common across all OpenFog use cases. It provides descriptions and reference points for fog architectural attributes and properties. The Autonomous Driving use case begins on page 12.

The OpenFog Consortium is defining applications and architectures for fog computing. The Consortium defines fog computing as: **A horizontal, system-level architecture that distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum**.

The first step in this architectural process is understanding the spectrum of vertical markets and applications that we expect fog computing technologies may serve. This document focuses on a representative use case that we believe spans many aspects of fog computing and therefore serves to define the functions we hope fog architecture, fog implementations, and fog deployments will provide.

It is important to understand how this use case fits into the overall process the Consortium uses to define interoperable and certifiable architectures. As shown in Figure 1, the use case described in detail in this document is a starting point for the suite of OpenFog technical documentation. When taken together, OpenFog use cases cover the basic fog functions of approximately 80% of the comprehensive set of IoT network applications we have identified for fog.

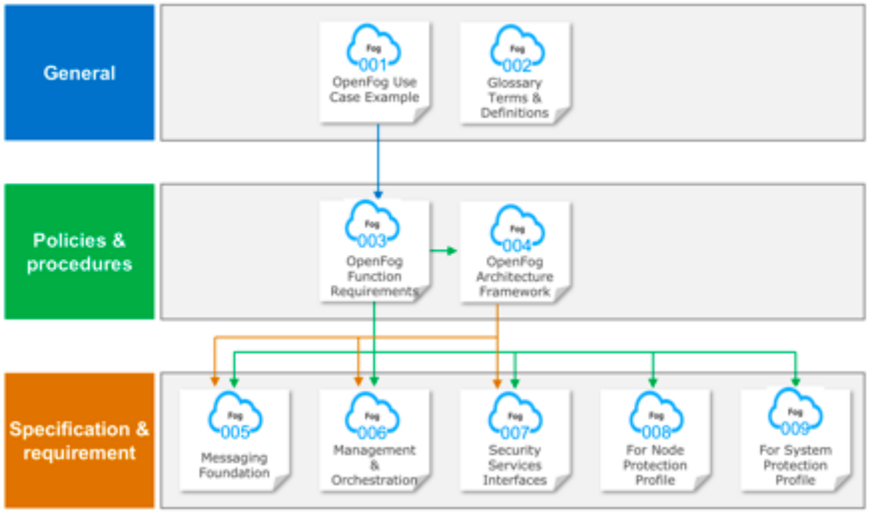
General
Fog
001
OpenFog Use Case Example
Fog
002
Glossary Terms & Definitions
Policies & procedures
Fog
003
OpenFog Function Requirements
Fog
004
OpenFog Architecture Framework
Specification & requirement
Fog
005
Messaging Foundation
Fog
006
Management & Orchestration
Fog
007
Security Services Interfaces
Fog
008
For Node Protection Profile
Fog
009
For System Protection Profile

Figure 1. Hierarchy of OpenFog Consortium specification documentation

The composite of all use cases outlines a problem statement for OpenFog, describing the essential functions for all fog elements and networks. The Consortium extracts requirements from these use cases, and distills and correlates them to produce a detailed *Fog Requirements Document*. These requirements serve three important purposes:

1. To drive the OpenFog Reference Architecture;
2. To guide the development of OpenFog testbeds for testing and validation purposes; and
3. To provide guidance to implementers of fog nodes and networks.

The *Architecture Framework Document* is a compendium document that describes the key functional components of OpenFog as well as the interfaces between these components.

The Consortium also publishes additional documents, which describe details in areas such as security, management and orchestration, and messaging. Implementers may use the compendium as a guide for the conceptual planning and architecture design for their fog-based systems, and as implementation best practices for OpenFog elements and networks that will interoperate and can be certified as OpenFog compliant.

OpenFog Consortium workgroups reviewed and discussed hundreds of potential fog use cases spanning more than a dozen vertical markets related to IoT. The Consortium carefully selected a set of use cases that we believe spans a representative set of potential fog applications.

These use cases will highlight one or more representative attributes of fog such as latency, network bandwidth, reliability, security, programmability, scalability. The derived requirements from the use cases we include will cover an illustrative sample.

As mentioned, OpenFog technical requirements comprise a platform that covers approximately 80% of common fog functions. The remaining 20% of requirements needed to support specific use cases which are application dependent and won't be defined by the Consortium.

Readers should pay detailed attention to the subset of use cases that most closely match their areas of interest. We encourage you to browse additional use cases, as they may highlight less obvious aspects of fog that could prove valuable, and give insight into the rationale of the OpenFog requirements.

Readers are also encouraged to collect additional use cases and submit them to OpenFog for requirements extraction and potential inclusion in future use case documents.

4 Fog Computing Overview

Fog computing provides the missing link in the cloud-to-thing continuum. It is a critical architecture for today's connected world as it enables low latency, reliable operation, and removes the requirement for persistent cloud connectivity to address emerging use cases in Internet of Things (IoT), 5G, Artificial Intelligence (AI), Virtual Reality and Tactile Internet applications.

Fog architectures selectively move compute, storage, communication, control, and decision making closer to the network edge where data is being generated and used. This solves the limitations in current infrastructure to enable mission-critical, data-dense use cases.

Fog computing is an extension of the traditional cloud-based computing model where implementations of the architecture reside in multiple layers of a network's hierarchy. These extensions to the fog architecture may retain all the benefits of cloud computing, such as containerization, virtualization, orchestration, manageability, and efficiency.

The fog computing model provides the ability to move computation and storage from the cloud closer the edge, based on the needs of the data and the service requirements. These functions can potentially reside right next to the IoT sensors and actuators. The computational, networking, storage and acceleration elements of this new model are known as fog nodes. These nodes may also reside in the cloud, as they comprise a fluid system of connectivity and don't have to be fixed to the physical edge.

5 The OpenFog Reference Architecture

The OpenFog Consortium was founded on the principle that an open and interoperable fog computing architecture is necessary in today's increasingly connected world. Through an independently-run open membership ecosystem of industry, end users and universities, we can apply a broad coalition of knowledge to these technical and market challenges. We believe that proprietary or single vendor fog solutions are of limited value, as they can limit supplier diversity and ecosystems, resulting in a detrimental impact on market adoption, system efficiency, quality and innovation.

The OpenFog Reference Architecture (RA) is a medium- to high-level view of system architectures for fog nodes and networks. It is the result of a broad collaborative effort of the OpenFog ecosystem of industry, technology and university/research leaders. It was created to help business leaders, software developers, silicon architects and system designers create and maintain the hardware, software and system elements necessary for fog computing, as well as design, architect and develop solutions that enable fog-cloud, fog-thing and fog-fog interfaces.

6 Benefits of Fog

Fog computing targets cross-cutting concerns such as the control of performance, latency and efficiency, which are also key to the success of fog networks. Cloud and fog computing are on path to a mutually beneficial, inter-dependent continuum.

Certain functions are naturally more advantageous to carry out in fog nodes, while others are better suited to cloud. The traditional backend cloud will continue to remain an important part of computing systems as fog computing emerges. The segmentation of what tasks and single purpose functions go to fog and what goes to the backend cloud, are application and implementation/use case specific.

This segmentation can be planned and static, but can also change dynamically if the network state changes in areas such as processor loads, link bandwidths, storage capacities, fault events, security threats, energy availability, cost targets, and so on.

The OpenFog RA enables fog-cloud and fog-fog interfaces. OpenFog architectures offer several unique advantages over other approaches, which we term SCALE:

- **S**ecurity: Additional security to ensure safe, trusted transactions
- **C**ognition: Awareness of client-centric objectives to enable autonomy
- **A**gility: Rapid innovation and affordable scaling under a common infrastructure
- **L**atency: Real-time processing and cyber-physical system control
- **E**fficiency: Dynamic pooling of local unused resources from participating end-user devices

To illustrate this concept, let’s look at a quick use case example: Consider an oil pipeline with pressure and flow sensors and control valves. One could transport all those sensor readings to the cloud (perhaps using expensive satellite links) to analyze the readings in cloud servers to detect abnormal conditions, and send commands back to adjust the positon of the valves.

There are several problems with this scenario: The bandwidth to transport the sensor and actuator data to and from the cloud could cost many thousands of dollars per month; those connections could be susceptible to hackers; it may take several hundred milliseconds to react to an abnormal sensor reading (during which time a major leak could spill significant oil); and if the connection to the cloud is down, or the cloud is overloaded, control is delayed or, in the worst case, completely lost.

Now, consider placing a hierarchy of local fog nodes near the pipeline. They can connect to sensors and actuators with inexpensive local networking facilities. These fog nodes immediately establish a community which provides the ability to collaborate. They can be highly secure, lessening the hacker threat. Fog nodes can also be given the authority to react to abnormal conditions in milliseconds, quickly closing valves to greatly reduce the severity of spills.

Local control in the fog nodes produces a more robust control system. Moving most of the decision-making functions of this control system to the fog – and only contacting the cloud occasionally to report status or receive commands – creates a superior control system.

Fog computing includes a set of high-level attributes of fog computing that we call the pillars; these include some of the fog advantages described in the pipeline control scenario. There are 8 pillars in total: security, scalability, openness, autonomy, reliability, agility, hierarchical organization and programmability. We will discuss all of these pillars in detail later in this document.

The OpenFog RA defines the required infrastructure to enable building Fog as a Service (FaaS) to address certain classes of business challenges. FaaS includes Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), and many service constructs specific to fog. The infrastructure and architecture building blocks below illustrate how FaaS may be enabled; this will be expanded upon in the reference architecture document.

The OpenFog RA describes a generic fog platform that is designed to be applicable to any vertical market or application. This architecture is applicable across many different markets including, but not limited to, transportation, agriculture, smart cities, smart buildings, healthcare, hospitality, financial services, and more, providing business value for IoT, 5G and AI applications that require real-time decision making, low latency, improved security, privacy protection and are network-constrained.

7 Use Case Scenario: Autonomous Driving

Use Case: Autonomous Driving

Vertical: Smart Transportation

Executive Summary

Autonomous Driving (AD) vehicles are moving faster than anyone could possibly imagine—rapid adoption, that is. A decade ago, AD was still more science fiction than fact. A novelty, perhaps, but not a mainstream problem solver for big transportation problems.

But the Internet of Things has changed that. Vehicles equipped with sensors already self-park and even self-brake. At the 2017 North American International Auto Show in Detroit, Waymo (Google's autonomous vehicle company) introduced a car equipped with radar, as well as sensors.

It's not non-traditional car manufacturers such as Waymo and Tesla that are behind AD. Mainstream car manufacturers around the world, like GM, Audi and Volvo, are testing autonomous vehicles with passengers. In London, autonomous vehicles are starting to be used for deliveries. From states in the U.S. with major highway traffic congestion problems to countries like Singapore with major urban traffic congestion problems, AD is being given the green light as a viable transportation alternative.

There are still significant roadblocks to wide-scale acceptance of AD vehicles. AD is a mission-critical application. Precise operations during every millisecond of drive time can have life-and-death consequences. But just consider the amount of information that has to be collected, shared, analyzed, and then acted upon in order for every vehicle on the road to execute the right action at precisely the right moment.

AD vehicles will create a staggering amount of data. By 2020, each AD vehicle may generate more than 4,000 GB per day:

- Video cameras ~20-40 MB/sec
- Radar ~10-100 KB/sec
- Sonar ~10-100KB/sec
- GPS ~50KB/sec
- Lidar ~10-70 MB/sec (laser-based LIght Detection and RAnging systems)

This makes AD a Big Data challenge with zero tolerance for processing delays. Even with 5G, satisfying the real-time connectivity requirements for this amount of data would be cost-prohibitive. Even if there was a way to solve the bandwidth cost problem, funneling all of that information from vehicles and data collection points along the road would simply take too long. The best analysis of a road hazard provided a split second too late is useless.

There must be a way to distribute compute power, storage and analytics across the smart transportation network—where data can move from east to west (between systems on the same hierarchical level) and north to south, (from systems on a lower hierarchy to systems on a higher hierarchy, including the cloud).

Vehicles also need sufficient on-board data and analytics to make safe decisions when there is no connection to the cloud. Essential data must be collected and forwarded to the cloud (or clouds) for long-term analytics.

AD systems must also work cooperatively in a larger system of sensors and systems. This larger system is almost staggering complex and dynamic. Vehicles must communicate with other vehicles in close proximity, as well as vehicles that are ahead of them that can share information about road conditions. Information must be shared with roadway sensors, which can be located in Road Side Units (RSUs) that connect to traffic monitoring and road surveillance services and provide real-time inputs on road status. RSUs are also connected to local transportation services for up-to-date information on road and lane closures. The same approach can be used for detecting road hazards and accidents.

Fog computing enables the critical functions for AD vehicles to collaborate, cooperate and utilize the underlying infrastructure to coordinate their operations within smart highways and roads and smart cities. It provides the "undertone" for resources and systems to ensure cooperation and collaboration, so that random events, interactions and participants can be "seen" and managed by AD applications in order to achieve predictable behavior. These infrastructure-enabled capabilities provide the foundational elements for more advanced AD architectures that depend on having this kind dynamic interoperability.

Challenges

- Confidence in autonomous vehicles is dependent on meeting safety and security expectations by the public and (evolving) regulations by a variety of agencies.
- Data collection and sharing must be constant across a constantly changing mix of autonomous and non-autonomous vehicles, infrastructure components, roadways, multiple cloud systems, pedestrians, and other stationery and moving pieces.
- Autonomous vehicles must be able to operate safely even when they are not connected to the cloud or can't get sufficient information from the cloud (perhaps based on interference or bandwidth congestion).
- There is a tremendous concern that autonomous vehicles are especially vulnerable to hacking in order to cause accidents, steal cars and intercept personal information.

- Fog distributes compute, storage, and control workload to aggregate, process, and act on conditions in real-time.
- The fog architecture is also hierarchical, supporting east-west and north-south data collection and sharing.
- Fog-based deployments provide the undertone of interoperability, eliminating the cost, complexity, translation latency, bandwidth

Solution	consumption, and manageability nightmare associated with creating network overlays for communications. • Fog-based deployments break down silos between applications, systems and networks, facilitating AD adoption, leveraging smart transportation infrastructure investments, and accelerating development of new applications.
Technology	• All fog nodes have the same behavior, manageability and control features. • Regardless of location, fog nodes automatically interoperate, share data and objects, and have the interfaces required for connectivity and inter-node communication. • Fog nodes can be mobile (in-vehicle), adjacent to or onboard roadway infrastructure (surveillance cameras, digital signs, etc.) for flexibility in planning with the assurance of interopability. • Regional fog nodes can aggregate and filter data higher up the hierarchy. • Pre-processing and analytics happen within the fog network, and filtered, normalized data is sent to the cloud for the deeper analytics required for planning and coordination.

Introduction

The industry has moved beyond theoretical discussions of whether or not Autonomous Driving (AD) vehicles (sometimes called smart cars) will be the way of the future. We’ve arrived at the practical work of seeing how far this transportation transformation can take us. How many transportation-related issues can be solved with AD? The more you think about this once-futuristic idea, the more ideas come to mind (Figure 2).

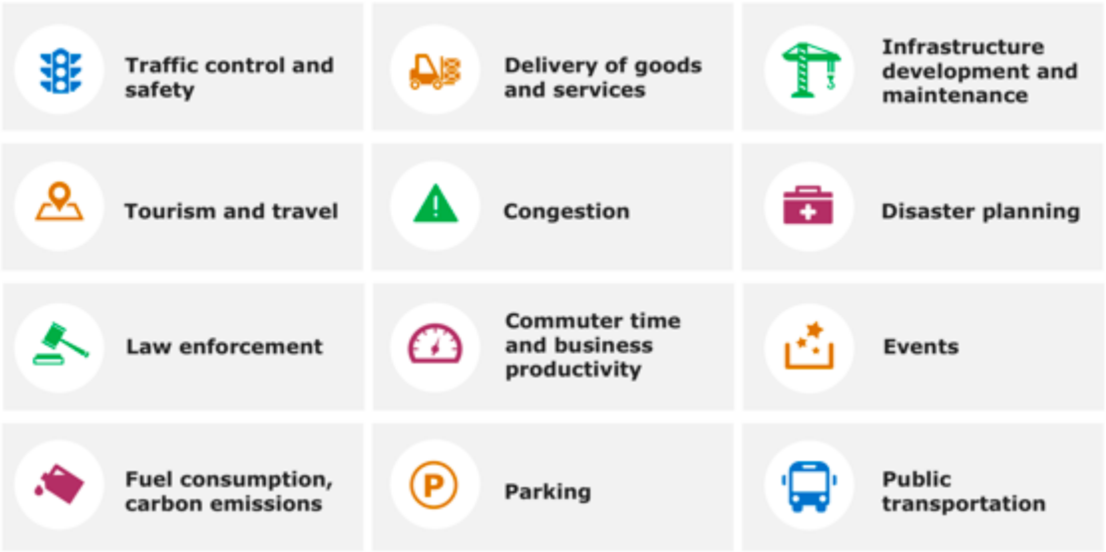
Traffic control and safety
Delivery of goods and services
Infrastructure development and maintenance
Tourism and travel
Congestion
Disaster planning
Law enforcement
Commuter time and business productivity
Events
Fuel consumption, carbon emissions
Parking
Public transportation

Figure 2. AD has traveled from a distant future vision to a cornerstone of modern transportation planning. This is just a partial list of the areas where AD is joining the conversation as a promising problem solver.

Business Case

The business case for AD has become stronger as we leap over technology hurdles. It's apparent from the actions of vehicle manufacturers and municipalities that investing in AD is an imperative. Any article about AD that talks about economic impact will cite things such as:

- Less road congestion
- Fewer accidents
- Faster emergency response
- More efficient transportation of goods
- Less dependence on labor for freight transportation (and eliminating concerns over fatigue and other reasons for impaired driving)
- Reduced emissions
- Potentially fewer roads and reduced infrastructure costs (due to more efficient use of existing roadways)
- Greater productivity for business commuters and businesses

- Facilitating new economic partnerships between cities and across regions because of easier transportation
- Broader access to low-cost transportation, potentially reaching and changing the lives of seniors, the disabled, inner city residents, rural communities and others with limited access to transportation.

Fog computing contributes to these economics by making AD viable. It can also contribute to ROI by leveraging existing assets on the smart highway and in smart cities, such as cameras, signs, intelligent parking meters, smart lighting, and even smart buildings. Lastly, it will eliminate the specter of developing layers and layers of overlay networks to accomplish what fog networking provides.

Interoperability

The realization of AD will depend on the cooperation, collaboration and partnership of a complex ecosystem of automobile manufacturers, transportation industry supply chain vendors, government agencies, law enforcement and first responders, and many others. In fact, as the applications for AD proliferate, it's impossible to predict how many entities, operating independently today, will need to be connected tomorrow. Furthermore, these entities will require common communication and data models, interfaces and exchanges in order to be interoperable.

There's another aspect of interoperability that's equally important is connectivity. These systems will require for all of the elements in a smart transportation system maintain awareness of each other in a timely, predictable manner. In a fast-moving environment, where connections are formed, lost, reformed and adjusted for limited availability, AD will still require the management of predictable services. Fog computing addresses situations for interoperability when there are no predictable connections with infrastructure capabilities that focus on ensuring continuous service delivery and delegation of authority.

Furthermore, we will have a transitional period, while fully autonomic vehicles will have to share the road with human driven and semi-autonomic vehicles. Fog computing will also provide the interoperability architecture elements to help with the retrofit of existing and mid-term vehicles (Figure 3) to begin outlining the collaboration and cooperation opportunities as this industry evolves.

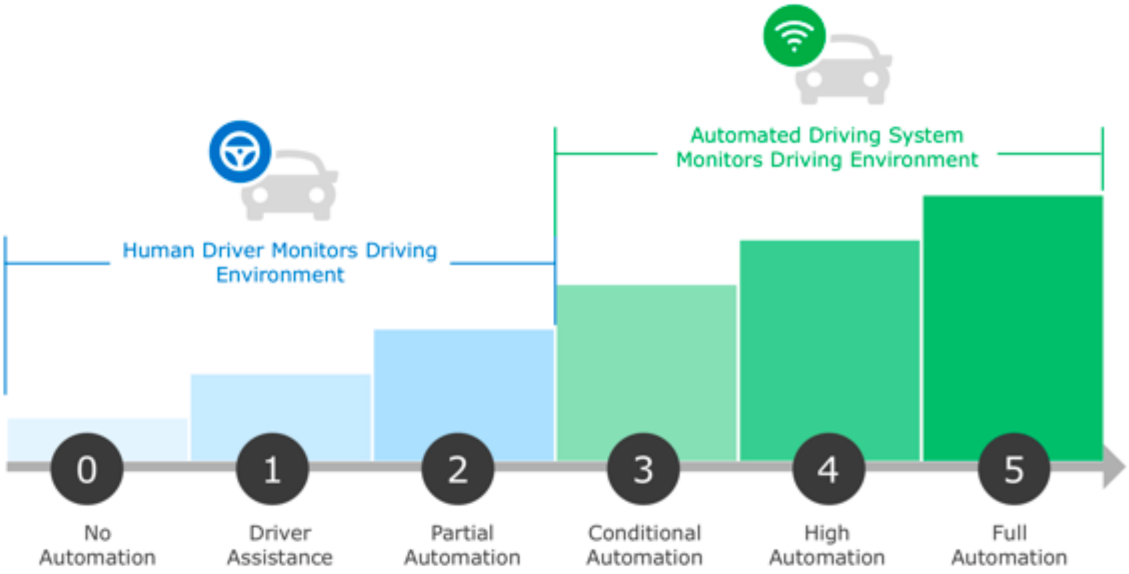

Figure 3: Levels of autonomy in vehicles will range from L0-L5, from no automation to full automation. (Source: SAE)

Safety

Public confidence in AD depends on the continuous safety of vehicles. Safety has two forms: the first is the safe operation of the vehicle on the road and among other vehicles, the second is the safe mechanical operation of the vehicle, in order to ensure proper and predictable behavior on the road.

In order to ensure safety, AD vehicles must ensure that the applications and services, like collision avoidance, intelligent routing and others are always available. Fog computing focuses on ensuring that the proper resources, systems and cooperative environments are always ready and available in order to be accessible when an outage happens.

This infrastructure must be in place, because a delayed response—due to bandwidth congestion or intermittent access—can result in catastrophic errors in autonomic actions.

Autonomy requires peripheral or ambient awareness, that is awareness of and the ability to share information with other “things.” It requires a predictable degree of self-awareness and self-management; furthermore, it requires that AD vehicles will take the appropriate action. Fog provides the infrastructure connectivity, standards of interoperability to ensure that all these “things” can interface, exchange information and collaborate to ensure safety for AD.

Safety also requires that the AD vehicle does not suffer mechanical failures. This requires on-board monitoring of all critical systems. There are two types of failures that you need to address: Predictable and imminent or sudden failure.

- To handle a predictable failure, the AD must be able to continuously evaluate the health of its own systems, especially critical ones like brakes. A braking system must operate as expected during both normal operation or in emergency situations. If, for any reason, the brakes are degrading, the AD must get early warning before there’s a failure.

- In the case of a sudden failure, like a flat tire, the vehicle must be able to recognize and respond to the problem in a split second, following a defined sequence of actions (e.g., decelerate, send alerts to cars in close proximity, call roadside assistance, alert the highway authorities, move the car out of a traffic—which also requires being aware of safe areas to stop, etc.). Mechanical operations safety can also encompass preventive maintenance.

Security

Customer confidence in ADs rests largely on alleviating the fear that AD vehicles can be hacked. Malicious intent might be interference with normal vehicle operation or theft of a vehicle or its cargo. Without this assurance of secure operation and consumer confidence, AD development may stall, adoption will be cautious, and regulations heavier.

Hackers have already exploited the federally-mandated on-board automobile diagnostics ports, interfaces and over the air (OTA) management and reporting systems. Perhaps the most widely publicized security threats are around hackers causing brake failures, taking control of engine operation, shutting down vehicles on the highway, and other life-threatening scenarios.

Future security scenarios will include the exchange of information and coordination of activities between vehicles, RSUs and other objects on and around roads. Fog computing will ensure not only the security of the data throughout its lifecycle, but also the transmission, receipt, acknowledgement, provenance and chain of custody of the data. With these functions covered, an AD will have the confidence to act upon an event with the assurance that the information is valid and the outcome is predictable.

Privacy protection is also a big concern. Proprietary manufacturer applications collect data regarding vehicle performance and wear. But these applications will probably extend to driver and passenger behavior (for fully autonomous vehicles). These applications require rigorous security and user-defined policies in order to protect against unauthorized interception of this new source of personal information.

When it comes to AD systems, which operate in life-or-death situations, security cannot be an afterthought. The goal of the fog architecture is to ensure that data is always available in a secure fashion, so the systems can have access to the information that is trusted, in order to make decisions with a high degree of confidence. Fog computing and networks are defined to provide security components, trust and trustworthiness functions as core elements of the architecture.

Use Case Overview

The architectural challenges described above makes the traditional edge-to-cloud-edge model insufficient for meeting the requirements of AD. A workable AD architecture will distribute workloads and provide orchestration for computing, communications, storage, control and services across:

- AD vehicles
- Network infrastructures
- RSUs, including actuators
- Private and public clouds

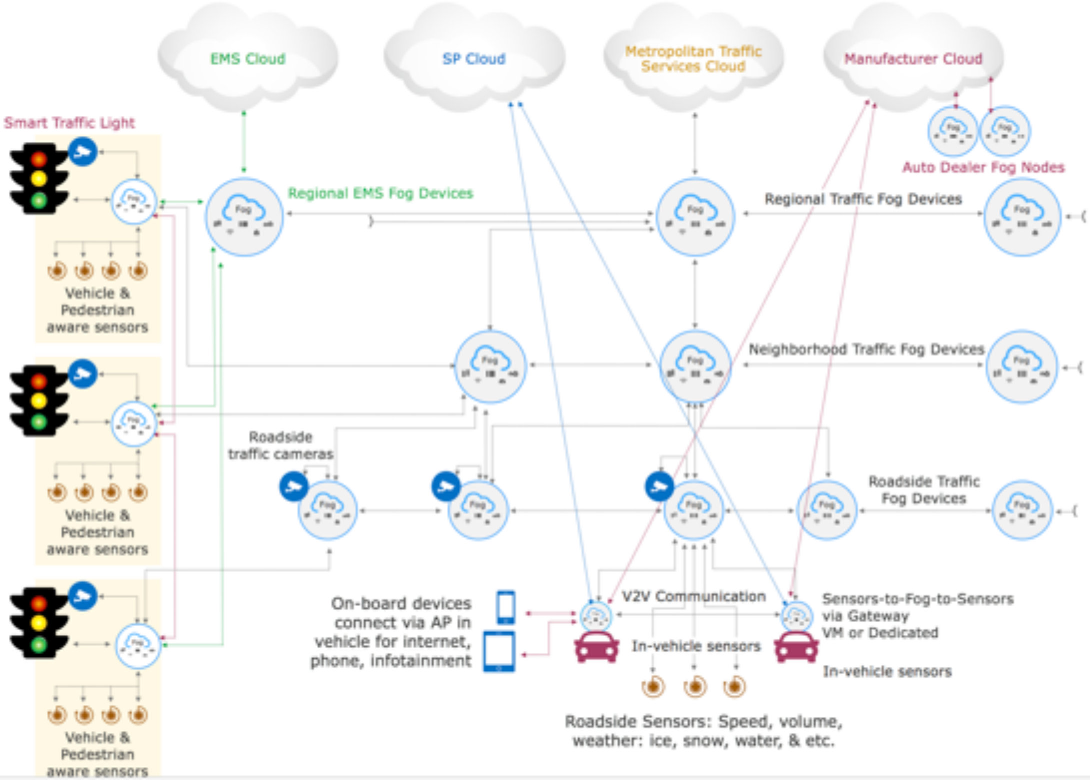
EMS Cloud
SP Cloud
Metropolitan Traffic Services Cloud
Manufacturer Cloud
Smart Traffic Light
Auto Dealer Fog Nodes
Regional EMS Fog Devices
Regional Traffic Fog Devices
Vehicle & Pedestrian aware sensors
Neighborhood Traffic Fog Devices
Vehicle & Pedestrian aware sensors
Roadside traffic cameras
Roadside Traffic Fog Devices
On-board devices connect via AP in vehicle for internet, phone, infotainment
V2V Communication
Sensors-to-Fog-to-Sensors via Gateway VM or Dedicated
In-vehicle sensors
In-vehicle sensors
Vehicle & Pedestrian aware sensors
Roadside Sensors: Speed, volume, weather: ice, snow, water, & etc.
Fog

Figure 4: The AD use case demonstrates the hierarchical and distributed advantages of a fog architecture. All of these fog nodes have the same behavior, manageability and control features. Regardless of location, they automatically interoperate, share data (because it has the same format and metadata), share objects, and have the proper interfaces for connectivity and inter-node communication. This fog-enabled hierarchy is how the fog network ensures quality and throughput to support vehicle-to-vehicle communication.

The OpenFog Reference Architecture for the AD use case, as shown in Figure 4, fills in the gaps along the cloud-to-thing continuum, providing:

- Access to mobile fog nodes supporting vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-everything (V2X) interactions.
- Access to and interoperability standards for multiple fog networks owned and operated by different authorities providing similar (and different) functionality.

- Fog node multi-tenancy with the ability to consolidate multiple workloads with strict isolation of distinct tenants, such as a brake control and fuel flow systems. However, some level of data sharing may be required, which means the ability to access multiple fog networks.
- A rich set of interactions, interfaces and messaging standards among multiple fog domains, as well as multiple cloud domains, including Element Management Systems (EMS), service providers (SPs), metropolitan traffic services, and system manufacturer clouds.
- The ability for both private and public fog and cloud networks to be reachable by a single end point device.

Whether you're designing for an AD component or a smart city transportation system, fog computing provides the infrastructure enablement and interoperability foundation. The following examples describe how fog computing will impact architectural considerations in key AD applications or environments.

Smart Navigation

Smart navigation has many more dimensions than using GPS for route mapping, such as combining destinations with vehicle or driver history to suggest more energy-efficient routes or connection with smart street lighting via Wireless Sensor Networks (WSNs). New concepts even include mobile pedestrian navigation systems (PNS), which provide route information based on landmarks instead of street maps.

With fog computing and smart transportation highways, vehicles can also "see" through things and mange "non-line of sight" road conditions through communication with other vehicles. The navigation system is constantly updated with this data.

In-vehicle fog nodes also communicate with fog nodes located on RSUs. This gives the AD immediate access to road or lane closures, accidents, road debris, and weather-related driving conditions. Smart navigation can also provide alternate routes based on the latest information on weather, traffic, construction, and road conditions provided by the cloud.

High Definition (HD) maps are critical for smart navigation. This requires a massive amount of data collection and aggregation by the vehicle and transmission to the map service for map refresh, as well as progressive download.

Cooperative Driving

Cooperative driving (also known as mobile platooning services) requires a convoy leader to constantly update the other vehicles with direction, speed, lane, road conditions, congestion, and other data. It requires hive-like communication, in which this information is shared with all the vehicles in the convoy.

In-vehicle fog nodes in a convoy can establish a LAN to aggregate data traffic from multiple vehicles before transmission over a cellular network.

In addition, a group of vehicles can act as federated resources for compute and path planning, ensuring RAS even if there is no connectivity to the access network. This is tightly coupled with a vehicle's sensing capabilities and the ability to adapt the automated driving based on variable road conditions and situations sensed locally. V2X adds to the sensing capabilities of the vehicle through safety messages.

Fog computing includes the concept of "delegation of authority." In a situation where the convoy leader is unable to connect to the cloud, it can locate other fog nodes in the vicinity. Because the computational workload is distributed across the fog infrastructure (in case of the convoy, this means all the participating vehicles), the convoy leader can make its own control decisions, even it's based on a more limited data set than the cloud might provide.

With fog computing, you can also create a peloton. Some of the advantages include fuel efficiency. It also enables a vehicle to leave the convoy and the convoy will self-heal. A vehicle can also join the convoy en route, and the convoy will can apply security mechanisms like discovery, authentication, and a reputation score to allow the new vehicle to join the convoy and place it in the correct position.

Smart Cities and AD

The following example shows how the AD use case overlaps smart cities. The urban revolution cannot be better characterized than through the convergence of AD and smart cities.

One of the key resources that our cities possess are roads, but anyone who ever tried to navigate them, especially around rush hour, knows that roads are a limited resource. The next evolution is how smart cities will use the data available from AD vehicles, RSU(s), smart buildings and other infrastructure to use roads more efficiently and safely. There are significant changes in our future:

- AD vehicles will replace drivers, which will change how urban roads, streets, parking, and parking structures will be used.
- Over time, most of the traffic infrastructure we know today, like signage, will be outdated and unnecessary.

Fog computing is at the center of this evolution. The OpenFog RA defines an open data format, interfaces and messaging standards in order to ensure interoperability. These core functions will be necessary as the combination of smart cities and self-driving, intelligent vehicles will essentially change the morphology of our cities, municipalities and urban governance models.

Advantages of the Fog Computing Approach

The fog computing approach provides the interoperability, messaging, and interface standards to create a cooperative community of nodes, while greatly reducing latency, network bandwidth and availability constraints. The architecture for fog computing is based on eight pillars (Figure 5), as identified by the OpenFog Consortium. In AD and related roadside infrastructure elements, the following pillars are addressed:

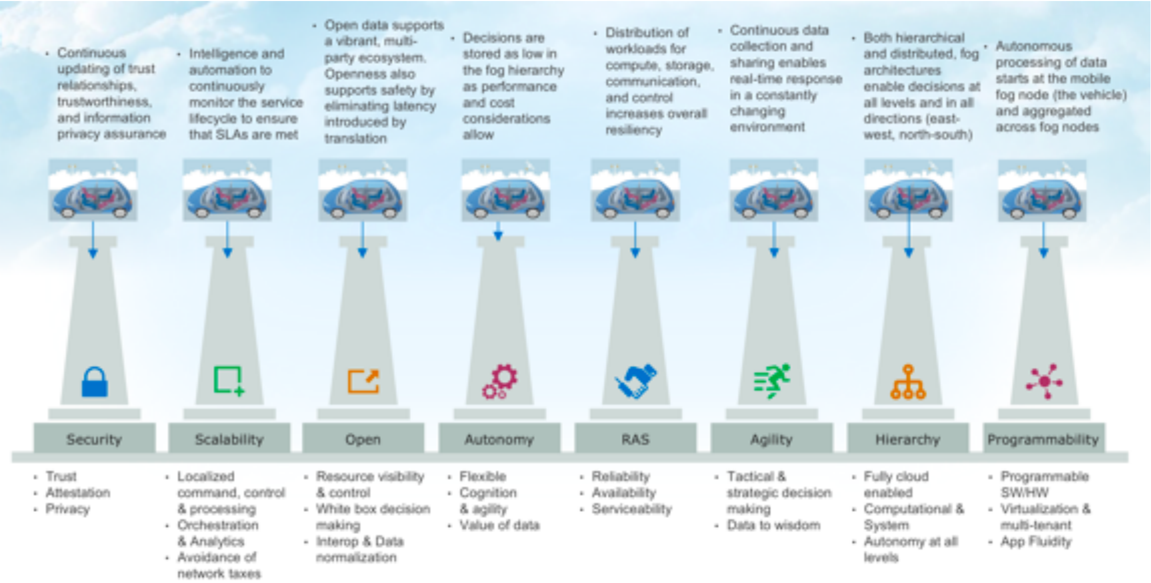
Continuous updating of trust relationships, trustworthiness, and information privacy assurance
Intelligence and automation to continuously monitor the service lifecycle to ensure that SLAs are met
Open data supports a vibrant, multi-party ecosystem. Openness also supports safety by eliminating latency introduced by translation
Decisions are stored as low in the fog hierarchy as performance and cost considerations allow
Distribution of workloads for compute, storage, communication, and control increases overall resiliency
Continuous data collection and sharing enables real-time response in a constantly changing environment
Both hierarchical and distributed, fog architectures enable decisions at all levels and in all directions (east-west, north-south)
Autonomous processing of data starts at the mobile fog node (the vehicle) and aggregated across fog nodes
Security
Scalability
Open
Autonomy
RAS
Agility
Hierarchy
Programmability
Trust
Attestation
Privacy
Localized command, control & processing
Orchestration & Analytics
Avoidance of network taxes
Resource visibility & control
White box decision making
Interop & Data normalization
Flexible
Cognition & agility
Value of data
Reliability
Availability
Serviceability
Tactical & strategic decision making
Data to wisdom
Fully cloud enabled
Computational & System
Autonomy at all levels
Programmable SW/HW
Virtualization & multi-tenant
App Fluidity

Figure 5. OpenFog Reference Architecture pillars mapped to the Autonomous Cars Use Case.

Security: As a mission-critical application, AD has strict security and privacy requirements for computing, communications, and other parts of fog platform architecture. The OpenFog RA specifies the process by which fog computing manages the physical interfaces, wireless protocols, and packets and data being transferred. This ensures that the right packet gets to the right location. This is one of the strengths of fog computing for AD, given the instantaneous trust relationships that must be continuously enabled and disabled across an unknowable number and type of elements.

Trustworthiness (including authentication, authorization, and a root of trust) adds understanding of the behavior of devices, fog nodes, networks and data in a way that is predictable and conforms to expected rules, policies and conditions.

Privacy assurance means that the data generated about the vehicle, its owners, its passengers and location meet information privacy requirements dictated by organizations or regulatory agencies.

Scalability: While some applications may not require service quality guarantees (e.g., self parking), more sophisticated capabilities (e.g., crash avoidance) require increased intelligence and automation. The service lifecycle must be continuously monitored and measured to guarantee that the terms of Service Level Agreements (SLAs) are measured and met.

Fog Management and Orchestration (Figure 6) ensures that the SLA is maintained. When an anomaly is detected, the orchestrator takes remediation action with minimal or no impact to AD applications and services. SLAs set the terms of behavior, the business requirements for reliability, availability, serviceability and security/safety (RASS), and measurements and metrics. Quality of Service (QoS) describes the infrastructure's ability to perform in a measurable way to meet the terms of an SLA.

For Management and Orchestration, transactions are defined as a sequence of events and related work. This sequence is comprised of sub-elements or subcontractors. The subcontractors are contextually managed and measured against individual metrics (such as in-band and out-of-band telemetry).

A transaction is treated as a unit for the purposes of satisfying a request between the consumer and provider. The transaction is managed via an orchestrator, which understands the soft bounds of both the consumer/provider and subcontractor service. SLAs direct the behavior of the subcontractors to achieve the consumer/provider contractual agreements within the predicted boundaries set in the SLA.

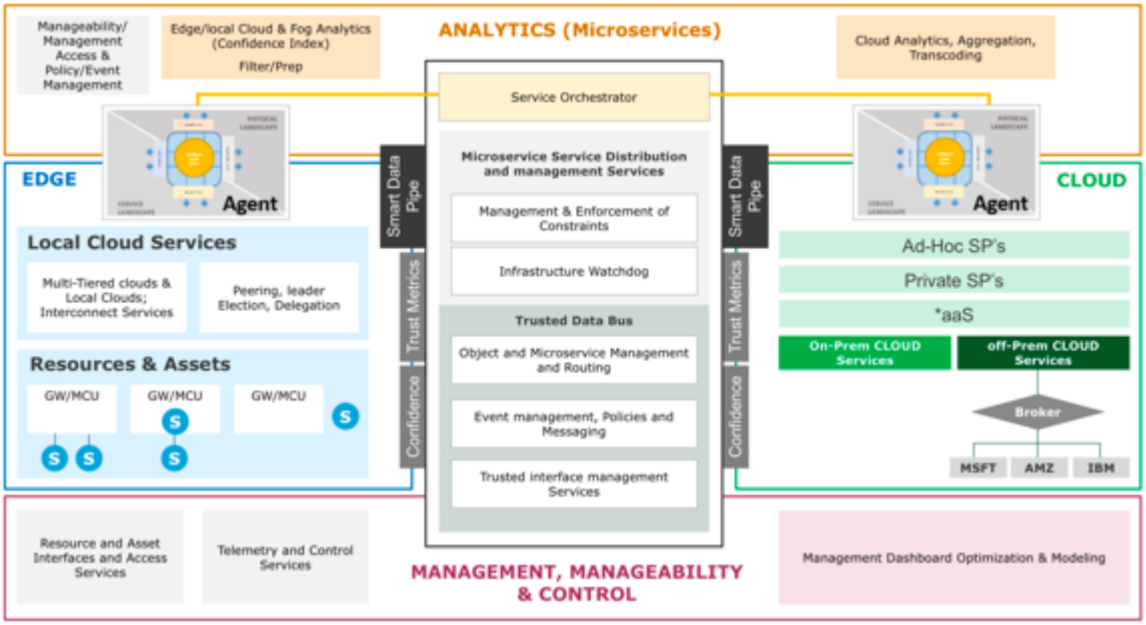
ANALYTICS (Microservices)
Manageability/ Management Access & Policy/Event Management
Edge/local Cloud & Fog Analytics (Confidence Index) Filter/Prep
Cloud Analytics, Aggregation, Transcoding
Service Orchestrator
EDGE
Agent
Smart Data Pipe
Trust Metrics
Confidence
Microservice Service Distribution and management Services
Management & Enforcement of Constraints
Infrastructure Watchdog
Trusted Data Bus
Object and Microservice Management and Routing
Event management, Policies and Messaging
Trusted interface management Services
Local Cloud Services
Multi-Tiered clouds & Local Clouds; Interconnect Services
Peering, leader Election, Delegation
Resources & Assets
GW/MCU
GW/MCU
GW/MCU
S
CLOUD
Agent
Ad-Hoc SP's
Private SP's
*aaS
On-Prem CLOUD Services
off-Prem CLOUD Services
Broker
MSFT
AMZ
IBM
Resource and Asset Interfaces and Access Services
Telemetry and Control Services
MANAGEMENT, MANAGEABILITY & CONTROL
Management Dashboard Optimization & Modeling

Figure 6: Transaction Management and Orchestration in fog.

Open: Vehicle manufacturers, fleet owners, the driving public, insurance companies, and government regulators all want an open, vibrant, multi-party ecosystem producing these capabilities to improve their cost, functionality, safety and rate of innovation well beyond what would be possible in a single vendor proprietary solution. One of the primary drivers for interoperability is open data, including: the header, metadata, payload formatting, data transport/access, interfaces, common location, etc. Open also supports agility, autonomy and safety by eliminating the need for translation, which can cause unacceptable latency.

Autonomy: Each fog element is autonomous with a certain level of processing, storage, and decision making. Autonomy begins with "self-awareness" and the ability for a fog node to express its own capability, ability and current participation with other nodes and services. In general, decisions will be made and data will be stored as low in the fog hierarchy as performance and cost tradeoffs allow. Orchestration and service management functions will have the ability to get to real-time and current data about all the participants in events and services without having to talk to a central authority.

Reliability, Availability, and Serviceability/Safety (RASS): Each fog node includes standards for the meta data and telemetry necessary to provide input for systems that describe RASS features. This is important because distribution of compute, storage, communication, and control can increase the potential for a problem due to a dropped connection or during a hand off between networks. Fog computing enhances the traditional RASS features in the data center. Because fog nodes are automatically aware of each other and act as a community of nodes, they also have the ability to check on each other, request feedback, and get a health status on local network connections and alternative interfaces (BT or BLE, etc.) when WiFi or 5G are not available.

Agility: Each fog node is agile, which means it has the ability to evaluate a request and provide services if it is has that capability. Because of this agility, systems can be repurposed on request based on their capabilities. Fog networking supports dynamic location update schemes (based on the inter-fog node network and community service capabilities). As a result, ADs can react in real time to location-based traffic patterns or needs. Lastly, data interoperability is also considered between different fog nodes through transcoding and/or normalization. These are the processes by which 1) erroneous data are filtered and dropped, and 2) policies are applied to highlight relevant data. This helps ensure that the proper data is available for analysis according to location or priority constraints.

Hierarchy: The fog architecture accommodates both hierarchical and distributed implementations. As shown in Figure 4, the system includes several types of sensors and actuators (“things”). Things include roadside sensors (infrastructure) and on-vehicle sensors. These sensors provide data so that the various systems can carry out their given functions. Smart transportation systems also manage the actuators that control parts of the infrastructure, such as traffic signals, gates, and digital signs. The vehicles may connect to the cloud via a hierarchy of fog nodes, or work through distributed, mesh connections across nodes.

Programmability: Programmability is at the heart of an AD ecosystem and provides the ability to customize the fog platform in order to have it ready for the specific purpose it is about to fulfill at that time. This means that a street light may be repurposed for a specific period of time to be a member of a distributed video sensor system and collect visual information during an accident. Additionally, fog nodes may be members of an autonomous processing system, where the data starts at the lowest level of the fog hierarchy (the mobile fog node), and additional systems at various level of the fog hierarchy are included for the purpose of analytics close to the data source.

For example, the mobile fog node in an AD rental car could be programmed differently by the rental car company to customize the user experience depending upon if the occupant. The rental car company could download a present program for teenage renters. Then if the same car is rented by a commercial enterprise (like a ridesharing company), the mobile fog node could be temporarily reprogrammed with proprietary software provided by the commercial enterprise.

Another example of programmability is context-dependent updates of roadside fog nodes in response to holiday weekend traffic, bad weather, emergency evacuations, or peak traffic times.

Code may also be generated by stakeholders, including manufacturers, domain experts (like map or entertainment providers), fleet owners, insurance companies, and third-party integrators. The OpenFog RA is defining the interoperability standards. It is important for programmability for all code to be formatted and interact seamlessly on a high performance, secure, reliable network.

Architectural Considerations

In this use case, multiple distributed fog nodes (or fog mobile nodes) provide a compute infrastructure. The distributed nodes communicate with each other over a high-speed network or data bus.

In-vehicle fog nodes include sensors, cameras, and communications which collect a massive amount of data for V2X interactions. Their main role is to continuously sense the status of the vehicle and environment and the vehicle's field view to adapt the driving behavior accordingly (e.g., path dynamic planning, lowering speed, changing route) by performing complex analytics on the available data in order to identify the best course of action. These adaptations can be made in communication with cloud-based analytics, or based on local analytics present in-vehicle. In addition to internal sensors, the in-vehicle platform also extends its sensing capabilities through messages from other vehicles or from RSUs.

There are multiple sensors on the vehicle generating raw data that needs to be converted to objects around the vehicle. Raw data itself cannot provide the insight need for decision systems. Data fusion takes disparate sensor inputs, puts the vehicle in situ and generates a vehicle-centered view of dynamic and static objects by analyzing the data. In addition, there are applications placing both the vehicle and objects on a map to localize the vehicle.

Several different networking technologies, including Dedicated Short Range Communications (DSRC), cellular (e.g. 3G, LTE, 5G, etc.), can be used to provide secure inter-vehicle connectivity as well as the network and communications infrastructure, the roadside infrastructure, and the cloud(s).

RSUs with sensor capabilities can provide useful regional information to vehicles:

- Serve as data aggregation points to analyze, simplify, and extract information from a vehicle’s raw data.
- Detect metrics such as vehicle density and ramp length.
- Connect to roadside fixed sensors and cameras that monitor traffic at intersections; local analytics in the RSUs could analyze this data to determine route and traffic guidance to communicate to nearby fog nodes and the cloud. An example of a local service could be a rapid, low-latency HD map refresh to the vehicles in an RSU's proximity in order to enable real-time traffic diversion from an accident.
- Act as a relay between vehicles for cooperative driving.

Fog provides a hierarchy of data collection, processing, storage, and analytics (Figure 6). In order to ensure better security, reduced communication costs, reduced latency and improved response time, fog eliminates the need to send unfiltered information to the cloud.

Much of the pre-processing and analytics can now happen within the fog network. For example, fog platforms further up the north-south hierarchy can handle significant regional computational requirements for tasks such as coordinating regional traffic patterns, optimizing smart highway efficiency, and looking for safety or security problems in lower level fog nodes.

The appropriate data, validated, cleansed, normalized, filtered and analyzed is still sent to the cloud for long-term analysis and planning, but now the governance is given to the stakeholders in order to maintain and control policies and cost targets.

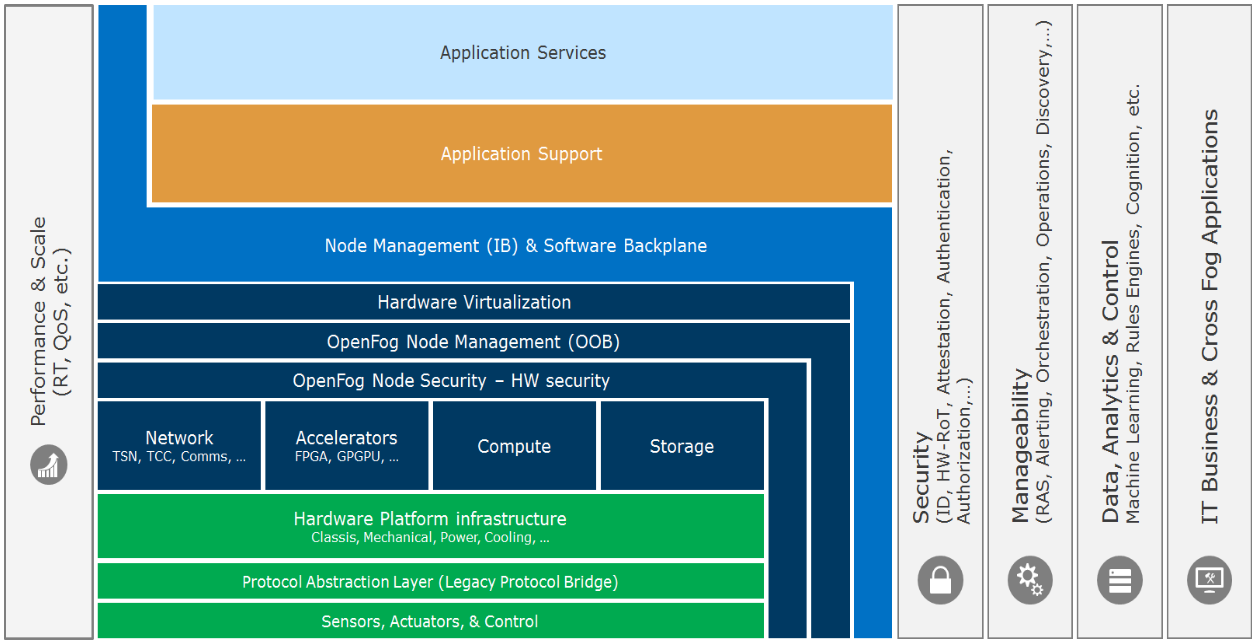
Performance & Scale
(RT, QoS, etc.)
Application Services
Application Support
Node Management (IB) & Software Backplane
Hardware Virtualization
OpenFog Node Management (OOB)
OpenFog Node Security – HW security
Network
TSN, TCC, Comms, ...
Accelerators
FPGA, GPGPU, ...
Compute
Storage
Hardware Platform infrastructure
Classis, Mechanical, Power, Cooling, ...
Protocol Abstraction Layer (Legacy Protocol Bridge)
Sensors, Actuators, & Control
Security
(ID, HW-RoT, Attestation, Authentication, Authorization,...)
Manageability
(RAS, Alerting, Orchestration, Operations, Discovery,...)
Data, Analytics & Control
Machine Learning, Rules Engines, Cognition, etc.
IT Business & Cross Fog Applications

Figure 7: The OpenFog Communications architecture offers guidelines for the communications functions required by AD vehicles.

Functionality

Sensors provide data so that the various systems (lights, cars, etc.) can carry out their given functions. Some sensors may include actuators that control parts of the infrastructure, such as traffic signals, gates, and digital signs.

Gateways and legacy protocol bridges are sensor surrogates for transmitting data to the system. These can be implemented in hardware or software within a fog node. Gateways can also provide security on behalf of sensors.

Neighborhood and regional transportation system fog nodes provide higher-level network connectivity. Typically, each fog layer in the hierarchy will provide additional processing, storage and network capabilities in service of vertical application at their level of the hierarchy. They provide further data analytics using additional processing capabilities. Additional storage is used to maintain data from a neighborhood fog node. The data is summarized and forwarded north to a higher level in the fog hierarchy.

The traffic control cloud and fog nodes receive input from other sources such as a Smart Traffic Light systems and a regional EMS system. These systems may be configured to autonomously modify the operation of traffic control devices and to create traffic routing suggestions on the fly based on traffic information input from other nodes.

Smart Traffic Light Systems are managed by the Traffic Control cloud or fog nodes, which provide schedules for its operation through the Traffic Control system. The schedule is dynamically modified based on current traffic conditions and predictions that are orchestrated by historical data from various levels in the hierarchy (based on the configuration and the rules and policies of the fog owner).

Interoperability

AD vehicle manufacturers are not in the position to develop, define and instrument the all the world's roads, roadside infrastructures and cities. This is why it's important for the OpenFog Consortium to define the standards for network, interface, messaging and data interoperability. The deployment of the infrastructure supporting AD and traffic management should be driven by an open approach that creates a shared infrastructure for use by many collaborating entities. It is also expected that the deployment and operation of this infrastructure will be subject to regulation by the local authorities.

The stakeholders in the architecture may include vehicle manufacturers, fleet owners, telecommunication companies and telecommunication equipment manufacturers, service providers, cloud service providers, and cities and government.

Topology

In this topology example, the AD vehicle is a mobile fog node that communicates with other fog nodes (V2I). However, it must be capable of performing all required in-vehicle operations autonomously if it cannot connect to other fog nodes or the cloud.

There are multiple in-vehicle systems, including infotainment, ADAS, etc. Each of these systems has different functions and they may reside on different fog nodes. In certain cases, these fog nodes may cooperate between each other in order to complete a high priority task.

The vehicle also has access to cellular and Internet and can provide these services to occupants or other systems within or around the vehicle. A key aspect to consider for these deployments is ensuring appropriate isolation. Isolation of functions addresses safety concerns by dictating that infotainment applications cannot impair ADAS' functions. Without fog computing, the only way to ensure is isolation is through physically separate systems. Fog can provide functional separation: each fog node provides a single purpose function that then can be used together when multiple fog nodes cooperate.

The in-vehicle fog node may also communicate directly with the RSUs. The infrastructure provides road condition and traffic data that the mobile fog node uses to make routing and driving decisions, as well as deal with road conditions that it has not yet encountered from its localized sensors (e.g., water, snow, ice, lane closures, etc.).

Fog nodes in-vehicle and fog nodes co-located with RSUs allow multi-radio access to the vehicles to ensure continuous connectivity and to support services access, for example:

- Office services through cellular connectivity and potentially 5G
- Video content distribution for AR/VR through cellular connectivity to the vehicle, with Wi-Fi or WiGig connectivity in-vehicle

The infrastructure will provide road condition information to other traffic system fog nodes as well as to vehicles. It may also provide both local and area information from other fog nodes based on data relevance. In-vehicle fog nodes may connect to other cloud systems, such as one or more service providers, government agency in order to have this data available for the community of cooperating fog nodes.

These nodes should be capable of some level of autonomous operation, based on the rules and policies determined by their owners. They will provide data to smart cars regardless of the connectivity to the upper layers of the fog network and peer east-west nodes. A full set of services may not be available in the case of a network failure. However, these services should be discoverable by the AD vehicle.

Data at Rest/Data In Motion/Data in Use

As vehicles move from L3 automation to fully autonomous L5 (see Figure 3), the data and orchestration volume and complexity challenge grows by an order of magnitude. The majority of the data that originates in the AD vehicle or infrastructure may not be sent to a cloud, at least not in its original form. In order to support the computational needs of these complex systems, each vehicle needs distributed, on-board compute architecture with significant connectivity between multiple on-board compute nodes supporting analytics, storage, and other resources.

By distributing computation (and other resources), a fog infrastructure can distill huge amounts of data generated by things closer to the data source. Instead of requiring a dedicated network to transmit raw data (trying to match bandwidth to data), the fog infrastructure manages the communication based on the available network.

Multi-tier fog architectures are based on upstream N-to-1 data creation and processing. This means that data that originates at the fog nodes—either in AD vehicles or in the infrastructure surrounding and sensing the vehicles—will flow upstream from vehicles to the cloud, as needed. Some of the pre-processing may happen in the vehicle and that data set may be passed to the cloud for post-processing.

Fog nodes can combine the data in an N-to-1 multi-tier manner. Fog nodes can combine multiple incoming streams of data through aggregation, compression, sub-sampling, transcoding, and others forms of processing and analyses. This is useful (and even necessary) in order to:

- Extract information at a lower level in the fog hierarchy to reduce latency
- Conserve limited (and expensive) resources, e.g., bandwidth, memory, and storage
- Combine data sources in different ways (as it moves across multiple tiers) to meet the needs of different stakeholders
- Maintain data provenance, trace and track data in motion, and ensure proper chain of custody
- Set and execute policies (for data transformation, data extraction, and local analytics) closer to the data source to minimize the raw data transmitted to the cloud.

Processing Algorithms and Analytics

Because AD is such a compute-intensive use case, processing algorithms and analytics are defined in the OpenFog RA. The architectural implications to support the AD data lifecycle include: data creation, data processing, security, routing, management, storage, consumption, and the migration and/or expiration of data.

The data created across the lifecycle includes rich meta data that captures the relationship between data streams. This data may be produced by different sensors, telemetry systems, and fog nodes, but they must find each other in cyber space. During creation, a data flow must be characterized not only by a unique identifier, but also by meta data such as the name, owner, access control policies, and other notable attributes. This meta data will enable data to be registered and also discovered, searched and/or queried.

Data must also be time stamped (e.g., to replay historical events). If possible and appropriate, it should be tagged to identify the geolocation from which it originated.

Data Access and Retention, Caching and Storage

As an ever-growing dataset, caching and storage are important architectural design considerations. There are three basic storage requirements:

- Local caching and storage for real-time action
- The preservation of data in the cloud for reuse and analysis
- Data that may be consumed and discarded immediately

Let’s look at local caching and storage for real-time action: At each aggregation point in the fog architecture, storage or caching must be available along with data migration and expiration algorithms. The retention policy for different AD data will be a function of criticality, historical access patterns, potential re-usability, as well as if retention is mandated by law for some duration (e.g., AD vehicle's black box).

A default policy would be to keep everything until local storage is no longer available. At that point, data can be transformed or compressed in some manner to take up less space, expired and/or simultaneously migrated to somewhere with more resources for longer-term storage.

Communications Considerations

The fog architecture will enable V2X communications and services. Each vehicle will contain one or more fog nodes that will communicate with the roadside infrastructure, the network, other vehicles, the cloud, pedestrians, and bikers as shown in Figure 8.

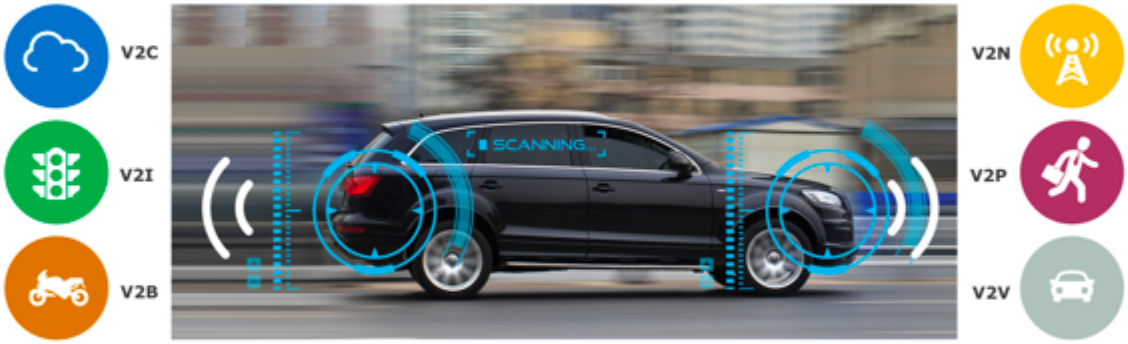

Figure 8: Communicating with other vehicles, infrastructure, cloud, pedestrians, and bikers, AD vehicles will be an integral part of the fog-based smart city ecosystems. AD vehicles will both enable fundamentally novel smart city services (e.g., automated parking and traffic congestion control, emergency services delivered with the help of participating AD vehicles), as well as relying on them.

Vehicle-to-Infrastructure (V2I): Each vehicle will receive data from fog nodes in the road or network infrastructure for traffic alerts and emergency control, as well as for a range of services such as services for infotainment.

Vehicle-to-Vehicle (V2V): Vehicles will communicate with each other through a direct link or through a multi-hop link using other vehicles/fog nodes. V2V communication allows sharing road information between vehicles and other useful scenarios as cooperative driving.

Vehicle-to-Network (V2N): V2N enables vehicles to get to critical services and utilize alternative network interfaces even when the best network is unreachable. For example, 5G networks may be used for high speed communication, but other interfaces and protocols may be used to complete a job or engage a service when 5G or other high speed network is not available.

Vehicle-to-Pedestrian (V2P): Vehicles will communicate with pedestrian devices for roads safety (e.g., warning before entering an intersection).

Vehicle-to-Biker (V2B): Mobile fog nodes on vehicles will communicate with mobile fog nodes on bikes about proximity to each other, road and weather conditions, structures, and pedestrians (in rural or city traffic).

Vehicle-to-Cloud (V2C): In a hierarchical manner, fog nodes aggregate and forward information to regional fog nodes, which may do additional filtering and forwarding to the cloud for analytics and storage. There is also constant cloud-to-vehicle communication, providing traffic alerts and a range of services such as over-the-air updates.

OpenFog Testbeds

The hierarchy of OpenFog testbeds will be structured as follows:

1. Many small, research-oriented locations that OpenFog Members are able to access will focus on proving the high-level OpenFog

architectural requirements and satisfying the minimum interoperability requirements via their Proof-Of-Technology(POT) Testbeds. The outcome of these Proof-Of-Technology testbeds could be open source code or a research publication available to OpenFog members.

2. Medium-sized, Interoperability Operation Model (IOM) testbeds will focus on overall solutions and end-to-end applications, with at least three OpenFog Sponsors participating to promote usage of diverse OpenFog Ready Solutions. They will demonstrate adherence to the OpenFog Reference Architecture and component-level interoperability and compatibility.
3. Large, regional testbeds will test pre-productization devices for application to the co-located OpenFog Certification Lab. After the OpenFog Certification Lab validates a product, members will be able to release it as an OpenFog Certified product. We expect many verticals, use cases, and individual applications will have specific requirements for interoperability and preferences for certain types of testbeds, and the Consortium intends to adapt to their needs.

8 Adherence to the OpenFog Reference Architecture

The OpenFog Consortium intends to partner with standards development organizations and provide detailed requirements to facilitate a deeper level of interoperability. This will take time, as establishing new standards is a lengthy process. Prior to finalization of these detailed standards, the Consortium is laying the groundwork for component level interoperability and certification. Testbeds will prove the validity of the OpenFog Reference Architecture (RA) through adherence to the architectural principles.

9 Next Steps

The OpenFog Reference Architecture (RA) is the first step in creating industry standards for fog computing. It represents an industry commitment toward cooperative, open and interoperable fog systems to accelerate advanced deployments in smart cities, smart energy, smart transportation, smart healthcare, smart manufacturing and more. Its eight pillars imply requirements to every part of the fog supply chain: component manufacturers, system vendors, software providers, application developers.

Looking forward, the OpenFog Consortium will publish additional details and guidance on this architecture, specify APIs for key interfaces, and work with standards organizations such as IEEE on recommended standards. The OpenFog technical community is working on a suite of follow-on specifications, testbeds which prove the architecture, lists of requirements, and new use cases to enable component-level interoperability. Eventually, this work will lead to certification of interoperable elements and systems, based on compliance to the OpenFog RA.

For more information, please contact info@openfogconsortium.org.

10 About the OpenFog Consortium

The OpenFog Consortium was founded to accelerate the adoption of fog computing and address bandwidth, latency and communications challenges associated with IoT, 5G and AI applications. Committed to creating open technologies, its mission is to create and validate a framework for secure and efficient information processing between clouds, endpoints, and services. OpenFog was founded in November 2015 and today represents the leading researchers and innovators in fog computing.

For more information, visit http://www.openfogconsortium.org/; Twitter @openfog; and LinkedIn /company/openfog-consortium.

11 Authors and Contributors List

Authors	Contributors
Hassnaa Moustafa, Intel	Robert Swanson, Intel
Maria Gorlatova, Princeton University	Raghu Kondapalli, Intel
Chuck Byers, Cisco Systems	Ryan Gentry, Intel
Eve Schooler, Intel	Evan Birkhead, OpenFog Consortium
Katalin Bartfai-Walcott, Intel	Judith Kelley, OpenFog Consortium
Joydeep Acharya, Hitachi	
Arsalan Mosenia, Princeton University	
Brett Murphy, RTI	
Clemens Vasters, Microsoft	
Srikanth Kambhatia, Intel	

Note: All publicly available use cases are reviewed and approved by the OpenFog Technical Committee.

12 Copyright / Disclaimer

This reference document is designed to provide a foundation for extracting requirements when developing fog-based architectures. It is a compendium document to the OpenFog Reference Architecture. https://www.openfogconsortium.org/ra/