



# Introduction to Mobile Edge Computing

**Xiaoming Fu**

**Acknowledgement:**

Eugen Borcoci (University POLITEHNICA Bucharest)



# Mobile Edge Computing: Motivation

---

## Facts:

- Internet and Telecom convergence → Integrated networks: **Future Internet**
- Novel services, applications and communication paradigms
  - Internet of Things (IoT) and Smart cities, M2M and Vehicular communications, Information-centric networking, social networks and big data,
  - Internet of Everything (IoE), etc.
- **Novel, emergent technologies** are changing networks & services architectures:
  - **Supporting technologies**
    - **Cloud Computing**
    - **Fog/Edge Computing /Mobile Edge Computing /Cloudlets**  
defined independently, but they can cooperate
    - **Software Defined Networks (SDN)**
    - **Network Function Virtualization (NFV)**
    - **Advances in wireless technologies: 4G-LTE, LTE-A, WiFi, 5G**

# Mobile Edge Computing

---



## Trends:

- **Cloud computing (CC)** is more and more used, including private/local and mixed cloud development
- However, traditional **CC centralization** (processing, storage,..) may lead to some **limitations**
- **Novel services and applications like IoT, mobility-related, .. would be better served by decentralized systems**
- **Edge networking devices and even user terminals – more powerful** in terms of processing, storage, communication capabilities
- **Result:** recent attempts to push CC capabilities to the network edge:
  - Fog/Edge Computing
  - Mobile Edge Computing
  - Cloudlets, ...
- **To discuss:**
  - What are their fundamentals? their relationship?
  - Competition? Cooperation? Complementary?

- 1. Introduction**
- 2. Fog/Edge Computing**
- 3. Fog Computing Architectures and IoT**
- 4. Mobile Edge Computing**
- 5. Cloudlets**
- 6. Fog computing - MEC - Cloudlet**
- 7. Open topics – cooperation and research**
- 8. Conclusions**

1. **➔ Introduction**
2. **Fog/Edge Computing**
3. **Fog Computing Architectures and IoT**
4. **Mobile Edge Computing**
5. **Cloudlets**
6. **Fog computing - MEC - Cloudlet**
7. **Open topics – cooperation and research**
8. **Conclusions**



# 1. Introduction

---

## Fog Computing, Mobile Edge Computing, Cloudlets, Micro datacenters, ...

- **Fog Computing (FC)** - (CISCO ~ 2011) extends the CC to the edge of networks, in particular wireless networks for the Internet of Things (IoT)
  - FC nodes (FCNs) are typically located away from the main cloud data centers, i.e., at the network edge
- **Mobile Edge Computing (MEC)** – ETSI - an industry spec. ~2014
  - MEC pushes the CC capabilities close to the **Radio Access Networks** in 4G, 5G
  - ETSI is developing a system architecture and std. for a number of APIs
- **Cloudlet** – developed by *Carnegie Mellon University* ~2013
  - A cloudlet is middle tier of a 3-tier hierarchy: 'mobile device – cloudlet – cloud'
  - Cloudlet ~ "data center in a box" whose goal is to "bring the cloud closer to the users"
- **Micro data center** – developed by *Microsoft Research*- ~2015
  - Is an extension of today's hyperscale cloud data centers (as Microsoft Azure)
  - to meet new application demands like lower latency and new demands related to devices (e.g. lower battery consumption)
- **The above approaches include partially overlapping concepts and are also complementary**

1. **Introduction**
2. **→ Fog/Edge Computing**
3. **Fog Computing Architectures and IoT**
4. **Mobile Edge Computing**
5. **Cloudlets**
6. **Cooperation or competition?**
7. **Open research topics**
8. **Conclusions**

## 2. Fog/Edge Computing

---

### Fog Computing (FC) definitions

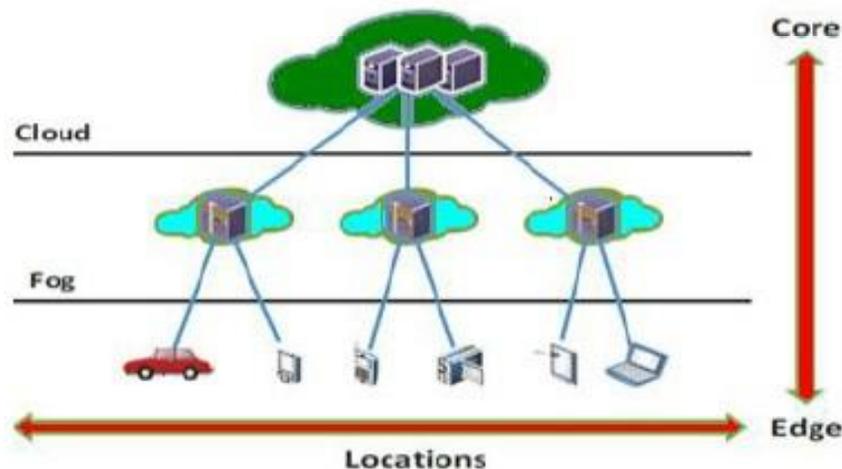
- **Initial (FC) – term coined by Cisco** to make the data transfer easier in wireless and distributed environments
  - Rationale : „**fog**” **means that „cloud” is closer to the ground**
    - **FC = Cloud Computing (CC)** carried out closer to the end users' networks
  - FC = virtualized platform, located between cloud data centers (hosted within the Internet) and end user devices
  - **FC offers strong support for Internet of Things**
  - **FC is not intended to replace CC; they are complementary**
- *Source: F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the Internet of Things, in ACM SIGCOMM 2012 Workshop on Mobile Cloud Computing, Helsinki, Finland, pp.13-16*
- **Fog computing/networking**
  - **decentralized** computing infrastructure
  - computing resources and appl. services are distributed in the most logical, efficient places, **at any point along the continuum** from the data source to the cloud
  - Higher efficiency: lower amount of data to be transported to the cloud for data processing, analysis and storage
    - Reasons: efficiency, security and compliance



## 2. Fog/Edge Computing

### Fog Computing (FC) definitions (cont'd)

- FC performs/offers significant amount of
  - *storage at or near the end-user* (avoid primarily to store in large-scale data centers)
  - *communication at or near the end-user* (avoid routing through the backbone network)
  - *management, including network* measurement, control and configuration, is performed at or near the end-user



Deployment of IoT applications in a 2-tiered way (Cloud-things) does not meet the requirements related to low latency, mobility of the “things” and location awareness

Solution : a multi-tiered architecture (at least 3 tiers) - Fog computing

Source[1]: I. Stojmenovic, S.Wen, ” The Fog Computing Paradigm: Scenarios and Security Issues”, Proc. of the 2014 Federated Conf. on Computer Science and Information Systems pp. 1–8



## 2. Fog/Edge Computing

---

### Fog Computing (FC) definitions (cont'd)

- **OpenFog Consortium (2015) definition**  
<http://www.openfogconsortium.org/resources/#definition-of-fog-computing>
- “**Fog computing is a system-level horizontal architecture** that distributes resources and services of computing, storage, control and networking **anywhere along the continuum from Cloud to Things**”
  - **Horizontal architecture:** Support multiple industry verticals and apps. domains, delivering intelligence and services to users and business
  - **Cloud-to-Thing continuum of services:** services and apps. can be distributed closer to Things, and **anywhere** along the continuum between Cloud and Things
  - **FC concept is at system-level:** spanning between the Things and the Cloud
    - over the network edges
    - across multiple protocol layers
    - not dependent on specific radio systems, protocol layer
    - it is not just at one part of an E2E system

## 2. Fog/Edge Computing

---

- **Fog Computing (FC) definitions (cont'd)**
- FC focuses **processing efforts at the LAN end** of the chain
  - Data are gathered, processed, and stored within the network, by way of an IoT GW or **FC node (FCN)**
    - Information is transmitted to this GW from various sources in the network
    - it is processed in FCN: then pertinent data + additional commands, are transmitted back, towards the necessary devices
- FC(+) :
  - enable a single, powerful processing device to
    - process data received from multiple end points
    - and send information exactly where it is needed
  - offers lower latency than centralized CC processing
- FC is scalable [2] as it gives to a centralized processing body a more big- picture view of the network as it has multiple data points feeding it information
- *Source[2]: R. Pieson, How does fog computing differ from edge computing? <http://readwrite.com/2016/08/05/fog-computing-different-edge-computing-pl1/>*

## 2. Fog/Edge Computing

---

- **Fog Computing (FC) definitions (cont'd)**
  - **More comprehensive FC definition:**
    - **FC** - scenario where
      - a huge **number of heterogeneous** (wireless and sometimes autonomous) ubiquitous and **decentralised devices**,
      - **communicate** and potentially **cooperate** among them and with the network to perform storage and processing tasks without the intervention of third-parties
    - tasks performed: **basic network functions** or **new services** and **applications** that run in a sandboxed environment.
    - *Source [3]: L.M. Vaquero, L.Rodero-Merino, "Finding your Way in the Fog: Towards a Comprehensive Definition of Fog Computing", ACM SIGCOMM Computer Comm. Review, Vol. 44, No 5, October 2014*



## 2. Fog/Edge Computing

---

- **Fog Computing (FC) definitions (cont'd)**
- **Related term to FC – having a larger scope**
  - **Edge Computing (EC)** - pushes the computing applications, data, and services *away from centralized nodes to the network edge*, enabling analytics and knowledge generation to occur close to the data sources
  - **Edge Computing (EC)** deals with resources that *might not be continuously connected to a network* : laptops, smartphones, tablets and sensors
  - EC covers a *wide range of technologies and services*:
    - wireless sensor networks
    - mobile data acquisition and mobile signature analysis
    - cooperative distributed P2P adhoc networking and processing also classifiable as Local Cloud/Fog computing and Grid/Mesh Computing
    - mobile edge computing
    - cloudlets
    - distributed data storage and retrieval
    - autonomic self-healing networks
    - remote cloud services
    - augmented reality, etc.



## 2. Fog/Edge Computing

- **Fog/Edge (FC) computing- summary of characteristics** [4]:
  - **Fog computing nodes (FCN)** are typically located away from the main cloud data centres, at the network edge
  - FC enables *low and predictable latency*
  - **FCNs**
    - are *wide-spread* and geographically available *in large numbers*
    - provide applications with *awareness of device geo location* and *device context*
    - *can support mobility of devices*
      - i.e. if a device moves far away from the in- service FCN, the fog node can redirect the app. on the mobile device to associate with a new app. instance on a FCN that is currently closer to the device
    - *offer special services that may only be required in the IoT context* (e.g. translation between IP to non-IP transport)
    - are typically accessed over *wireless* network
  - Fog app. code runs on FCNs as part of a distributed cloud application

Source [4] : Guenter I. Klas “Fog Computing and Mobile Edge Cloud Gain Momentum”, Open Fog Consortium, ETSI MEC and Cloudlets , Version 1.1, Nov 22, 2015

## 2. Fog/Edge Computing

### Comparison : Cloud Computing versus Fog Computing

- **FC provides**
  - **light-weight cloud-like facility** close of mobile users
  - users with a **direct short-fat** connection versus **long-thin** mobile cloud connection
  - customized and engaged **location-aware** services
- **FC is still new and there is still lack of a standardized definition**
- **Comparison between Fog/Edge (FC) and Conventional Cloud Computing [5]:**

	<b>Fog Computing</b>	<b>Cloud Computing</b>
Target User	Mobile users	General Internet users.
Service Type	Limited localized information services related to specific deployment locations	Global information collected from world-wide
Hardware	Limited storage, compute power and wireless interface	Ample and scalable storage space and compute power
Distance to Users	In the physical proximity and communicate through single-hop wireless connection	Faraway from users and communicate through IP networks
Working Environment	Outdoor (streets, parklands, <i>etc.</i> ) or indoor (restaurants, shopping malls, <i>etc.</i> )	Warehouse-size building with air conditioning systems
Deployment	Centralized or distributed in regional areas by local business (local telecommunication vendor, shopping mall retailer, <i>etc.</i> )	Centralized and maintained by Amazon, Google, <i>etc.</i>

Source [5] T H. Luan et.; al. , "Fog Computing: Focusing on Mobile Users at the Edge" arXiv:1502.01815v3 [cs.NI] 30 Mar 2016

## 2. Fog/Edge Computing

- **Comparison: Cloud Computing versus Fog Computing**
  - Comparisons on different parameters [6]:

Parameters	Cloud Computing	Fog Computing
Server nodes location	Within the Internet	At the edge of the local network
Client and server distance	Multiple hops	Single/multiple hop
Latency	High	Low
Delay Jitter	High	Low
Security	Non-locally controllable	Locally controllable
Location awareness	No	Yes
Vulnerability	Higher probability	Lower probability
Geographical distribution	Centralized	Dense and Distributed
Number of server nodes	Few	Very large
Real time interactions	Not fully supported	Supported
Usual last mile connectivity	Leased line / wireless	Mainly wireless
Mobility	Limited support	Supported

See also [6] K.P.Saharan, A.Kumar "Fog in Comparison to Cloud: A Survey", *Int'l. Journal of Computer Applications*, Volume 122, No.3, pp. 10-12, July 2015



## 2. Fog/Edge Computing

- **Comparison: Cloud Computing versus Fog Computing [3]**

	Cloud	Fog
<i>Latency</i>	High (eventual consistency)	Low (locality)
<i>Access</i>	Fixed and wireless	Mainly wireless
<i>Explicit mobility</i>	NA	Lispmob <sup>8</sup>
<i>Control</i>	Centralised/hierarchical (full control)	distributed/hierarchical (partial control)
<i>Service access</i>	through core	at the edge/ on handheld device
<i>Availability</i>	99.99%	Highly volatile/ highly redundant
<i># of users/devices</i>	Tens/Hundreds of millions	Tens of billions
<i>Price per server device</i>	\$1500-3000	\$50-200
<i>Main content generator</i>	Humans	Devices/sensors
<i>Content generation</i>	Central location	Anywhere
<i>Content consumption</i>	End devices	Anywhere
<i>Software virtual infrastructure</i>	Central corporate servers	User devices

Source [3] :L.M. Vaquero, L.Rodero-Merino, "Finding your Way in the Fog: Towards a Comprehensive Definition of Fog Computing", ACM SIGCOMM Computer Comm. Review, Vol. 44, No 5, October 2014



## 2. Fog/Edge Computing

---

- **Fog/Edge (FC) computing applications areas**
- Fog is considered to be an appropriate platform for a number of critical Internet of Things (IoT) services and applications:
  - Connected Vehicle
  - Smart Grid
  - Smart Cities
  - Wireless Sensors and Actuators Networks (WSANs)
  - ....
- **Note: no yet exists a globally accepted unique definition of Fog Computing versus Edge Computing**



## 2. Fog/Edge Computing

---

### Fog/Edge (FC) computing enabled applications

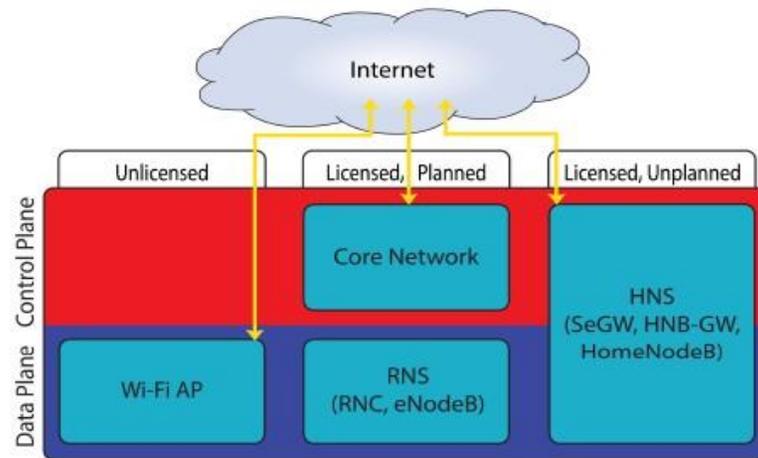
- **Data plane:**
  - **Pooling of clients idle computing/storage/bandwidth resources and local content**
  - **Content caching at the edge** and bandwidth management at home
  - Client-driven distributed beam-forming
  - **Client-to-client direct** communications (e.g., FlashLinQ, LTE/WiFi Direct, Air Drop)
  - **Cloudlets** (mobility-enhanced small-scale cloud data center located at the edge of the Internet) and micro data-centers
- **Control plane:**
  - Over the Top (OTT) content management
  - **Fog-RAN: Fog driven radio access network**
  - **Client-based HetNets control**
  - Client-controlled Cloud storage
  - **Session management and signaling load at the edge**
  - Crowd-sensing inference of network states
  - Edge analytics and real-time stream-mining
- **On top of Control Plane + Data Plane: apps such as 5G Mobile, IoT, Cyber-Physical, Data analytics**

Source[7]: M.Chiang, "Fog Networking: An Overview on Research Opportunities", December 2015, <https://arxiv.org/ftp/arxiv/papers/1601/1601.00835.pdf>

## 2. Fog/Edge Computing

### Fog/Edge (FC) use cases examples [7]

- **1. OTT network provisioning and content management**
  - Network services can be innovated faster: FC can directly leverage the “things” and phones, thus removing the need of introducing new boxes-in-the-network
  - SDKs sitting behind apps on client devices, allow tasks such as URL wrapping, content tagging, location tracking, behaviour monitoring
  
- **2. Client-based HetNets control (in 3GPP standards)**
  - In a HetNet (e.g., LTE, femto, WiFi) a client could observe its local conditions and **decide on which network to join** (in contrast to traditional network operator control)
  - Through randomization and hysteresis, such local actions may converge globally to a desirable configuration.



Source[7]: M.Chiang, "Fog Networking: An Overview on Research Opportunities", December 2015, <https://arxiv.org/ftp/arxiv/papers/1601/1601.00835.pdf>

## 2. Fog/Edge Computing

---

### Fog/Edge (FC) use cases examples [7]

- **3. Crowd-sensing LTE states – for resource management**
  - A collection of client devices can combine
    - **passive measurement** (e.g., Reference Signal Received Quality-RSRQ)
    - with **active probing** (e.g., packet train)
    - appl. throughput correlation and historical data mining,
  - in order to **infer in real-time the states of an eNB** (e.g., the number of Resource Blocks used)
- **4. Client-controlled Cloud storage**
  - Combined storage (in the Cloud) with Fog control (from client side control) can offer better data privacy
    - E.g., by spreading the bytes (of a given file), in a client shim layer, across multiple Cloud storage providers -7 better data privacy (even if encryption key is leaked by any given Cloud provider)

## 2. Fog/Edge Computing

---

### Fog/Edge (FC) use cases examples *(cont'd)*

- **5. Sharing bandwidth/resources with neighbors**
  - A terminal device can ask the neighbors to **share their LTE/WiFi (idle) bandwidth** by downloading other parts of the same file and transmitting, via WiFi Direct, client to client
  - Some neighbours become helpers of a given device
- **6. Bandwidth management at home gateway**
  - In a home set-top box/gateway, the limited broadband capacity can be allocated among competing users and application sessions,
  - according to each session's priority and individual preferences

## 2. Fog/Edge Computing

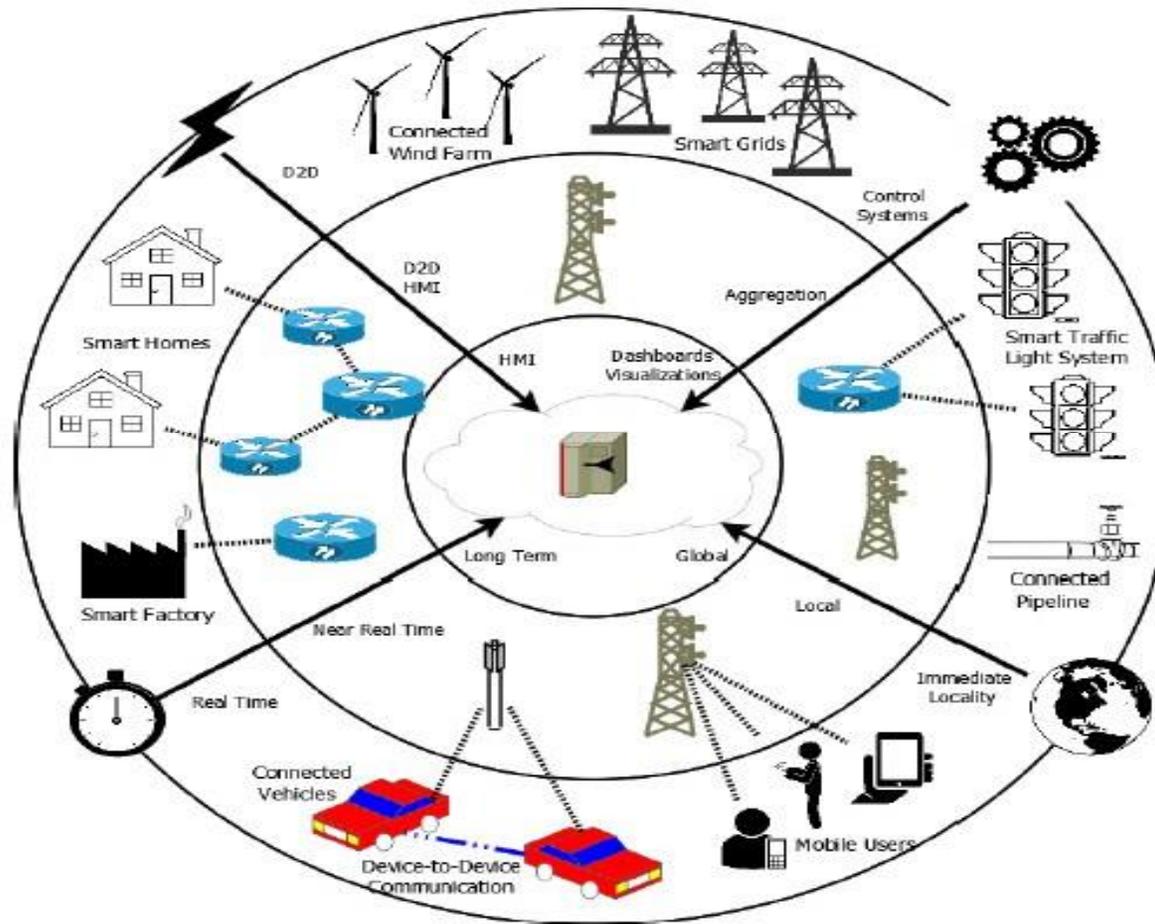
---

- **Fog/Edge (FC) use cases examples** (cont'd)
- **7. A Smart Traffic Light System (STLS) in** urban scenario
  - The STLS is a component in a Smart Connected Vehicle (SCV) and Advanced Transportation Systems (ATS)
  - **STLS goals/functions:**
    - local, real-time (rt) : accidents prevention
    - global, near - rt: efficient traffic management
    - global, non - rt : collection of relevant data to evaluate and improve the system
  - **Key STLS requirements examples:**
    - Local subsystem latency, Middleware orchestration platform
    - Distributed Networking infrastructure; Interplay with the Cloud
    - Consistency of a highly distributed system; Multi-tenancy; Multiplicity of providers
  - The STL
    - is deployed at each intersection.
    - has sensors measuring the vehicles' distance and speed and detects the presence of pedestrians and cyclists crossing the street.
    - can also issue “slow down” warnings to vehicles at risk to crossing in red, and even modifies its own cycle to prevent collisions.

*Source [9]: F.Bonomi, R.Milito, P.Natarajan and J.Zhu, “Fog Computing: A Platform for Internet of Things and Analytics”, in N. Bessis and C. Dobre (eds.), “Big Data and Internet of Things”: 169 A Roadmap for Smart Environments, Studies in Computational Intelligence 546, Springer Int'l Publishing 2014*

## 2. Fog/Edge Computing

- Fog/Edge (FC) use cases examples (cont'd)



Source [8]: A.V. Dastjerdi, et.al., "Fog Computing: Principles, Architectures, and Applications", 2016, Book Chapter in Internet of Things: Principles and Paradigms, <http://arxiv.org/abs/1601.02752>



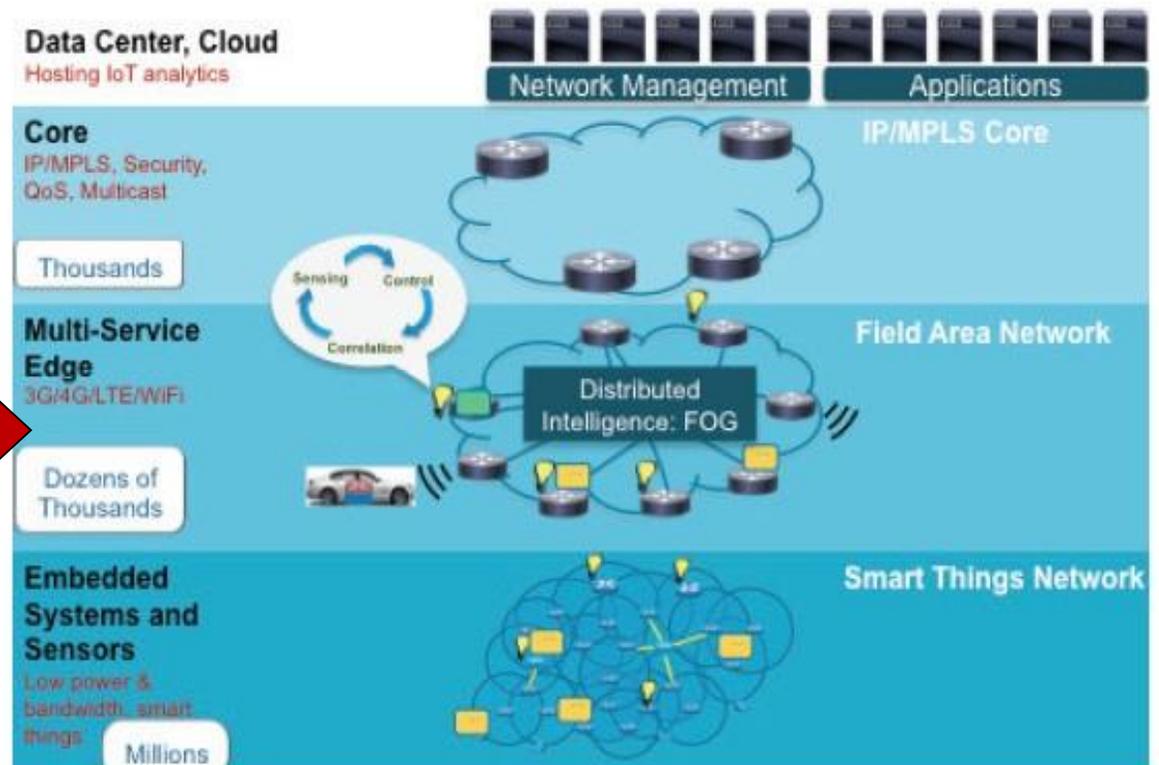


1. Introduction
2. Fog/Edge Computing
3. **➔ Fog Computing Architectures and IoT**
4. Mobile Edge Computing
5. Cloudlets
6. Fog computing - MEC - Cloudlet
7. Open topics – cooperation and research
8. Conclusions

## 3. Fog Computing Architectures and IoT

- From CISCO: view on idealized **information and computing architecture** supporting the future IoT applications; Fog distributed infrastructure for IoT/IoE

- Edge location, location awareness, low latency
- Geographical distribution
- Large-scale sensor networks to monitor the environment, and the Smart Grid
- Very large number of nodes
- Support for mobility
- Real-time interactions
- Predominance of wireless access
- Heterogeneity
- Interoperability and federation
- Support for on-line analytic and interplay with the Cloud

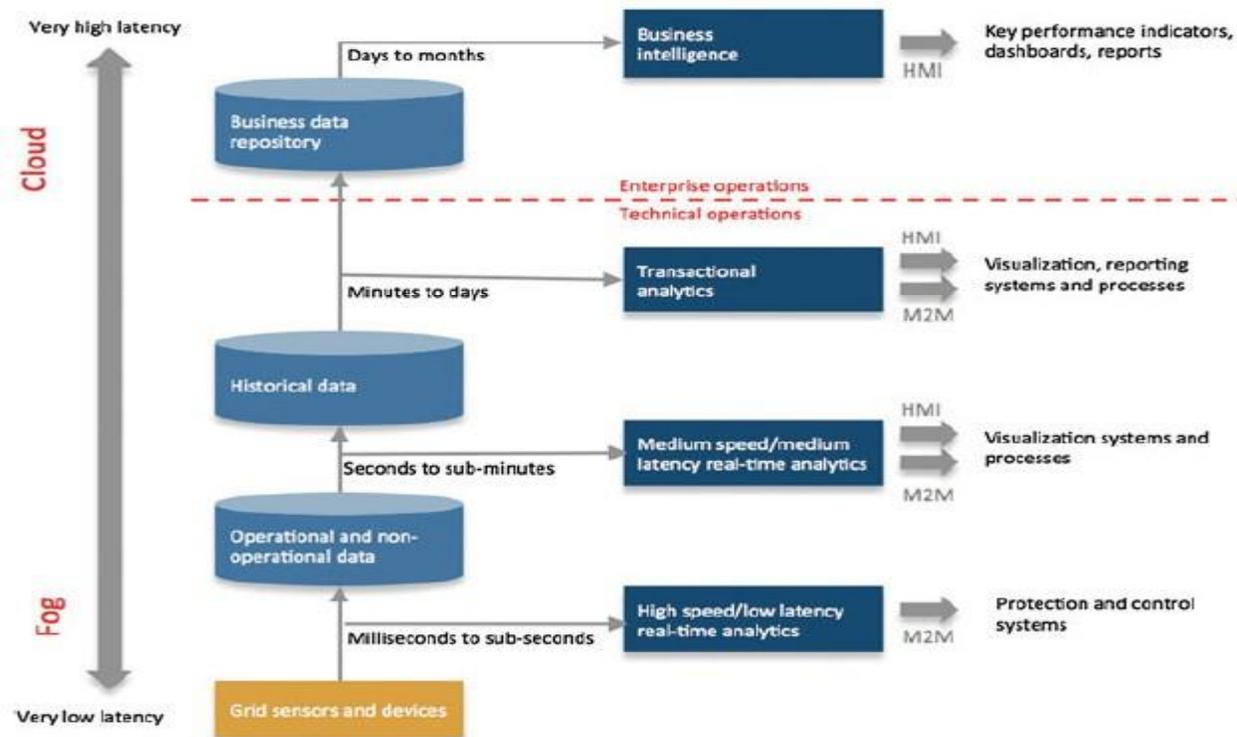


Source [1]: F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, ser. MCC'12*. ACM, 2012, pp. 13–16.

## 3. Fog Computing Architectures and IoT

- **Fog and Big Data**

- Big Data : characterized by Volume, Velocity, Variety and
  - *geo-distribution* – in case of Fog applications
  - data are processed in several layers

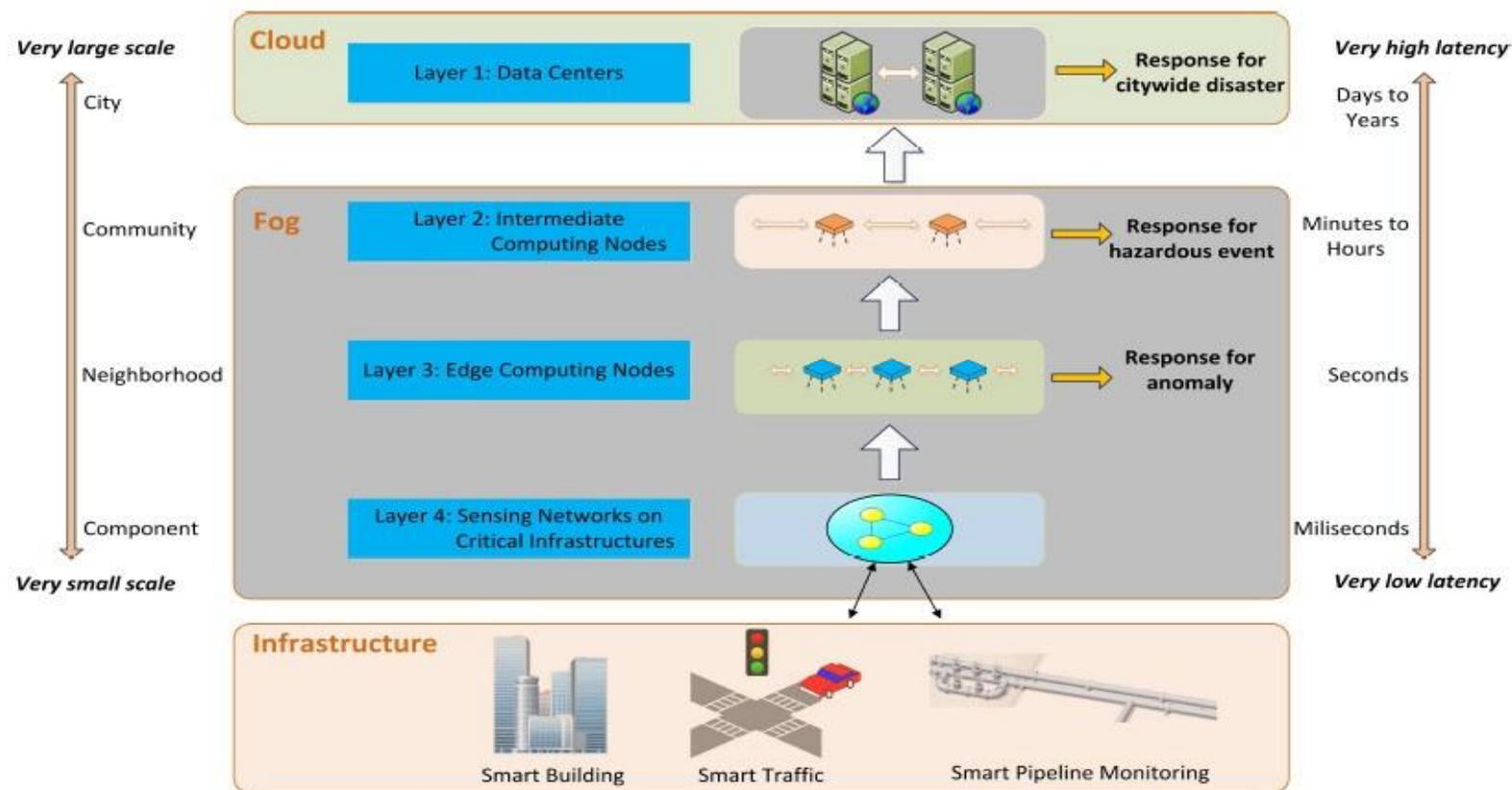


Source [9]: F.Bonomi, R.Milito, P.Natarajan and J.Zhu, "Fog Computing: A Platform for Internet of Things and Analytics", in N. Bessis and C. Dobre (eds.), "Big Data and Internet of Things": 169A Roadmap for Smart Environments, Studies in Computational Intelligence 546, Springer Int'l Publishing, 2014

## 3. Fog Computing Architectures and IoT

### FC in future smart cities

- Example: Hierarchical distributed FC layered architecture for smart cities



Source [10]: B. Tang, et.al., „A hierarchical distributed fog computing architecture for big data analysis in smart cities”, ASE BD&SI 2015, October 07-09, 2015, Kaohsiung, Taiwan, ACM, <https://www.researchgate.net/publication/281287012>, ISBN 978-1-4503-3735-9

## 3. Fog Computing Architectures and IoT

---

### FC in future smart cities (cont'd)

- **Example : Hierarchical distributed FC architecture**
- It realizes a quick response at neighborhood-wide, community-wide, and city-wide levels, providing high computing performance and intelligence in future smart cities
  - Integrating massive number of infrastructure components and services [10]
  - Geo-distributed system, having to process big data generated by massive number of sensors
  - The intelligence is distributed at the edge of a 4-layer FC network.
  - The FCNs at each layer perform latency-sensitive applications and provide quick control loop to ensure the safety of critical infrastructure components
- **Layer 4 (bottom)**
  - the sensing network (numerous non-invasive, highly reliable, and low cost, sensory nodes)
  - they are distributed at various public infrastructures to monitor condition changes over time
  - massive sensing data streams are generated, geospatially distributed, which should be processed as a coherent whole



## 3. Fog Computing Architectures and IoT

---

### FC in future smart cities

- **Hierarchical distributed FC architecture (cont'd)**
- **Layer 3: Edge computing nodes**
  - Many low-power and high-performance computing nodes or edge devices
  - Each edge device controls a local group of sensors that usually cover a neighborhood or a small community, performing data analysis in a timely manner
  - An edge device output has two parts:
    - reports of the results of data processing sent to the next upper layer intermediate computing node
    - simple and quick feedback control to a local infrastructure to respond to isolated and small threats to the monitored infrastructure components

## 3. Fog Computing Architectures and IoT

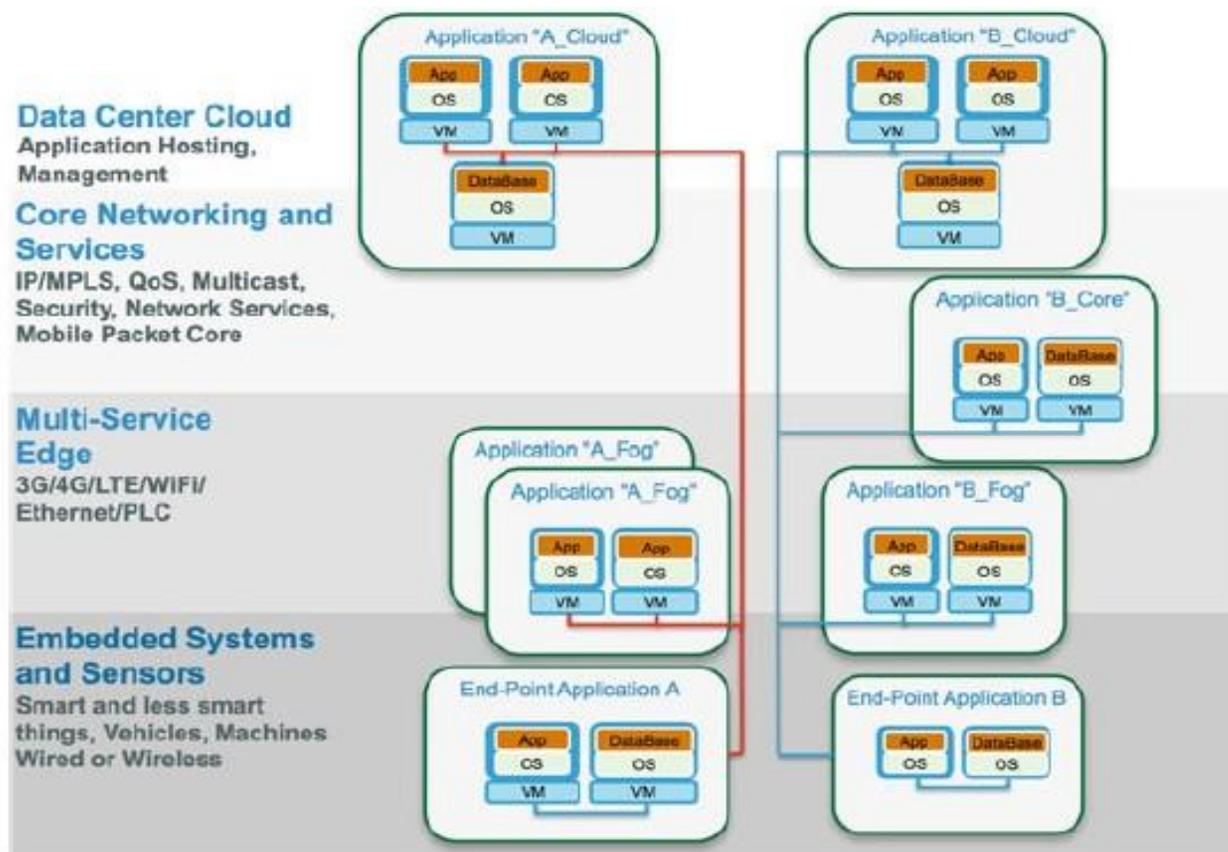
---

### FC in future smart cities

- Hierarchical distributed FC architecture (cont'd)
- Layer 2 : intermediate computing nodes
  - Each node controls an L3 group of edge devices and associates spatial and temporal data to identify potential hazardous events
  - It makes quick response to control the infrastructure when hazardous events are detected
  - The quick feedback control provided at L2 and L3 acts as localized “reflex” decisions to avoid potential damage
  - The data analysis results at L2 & L3 are reported to the top L1, for large- scaled and long-term behavior analysis and condition monitoring
- Layer 1( top) : Cloud Computing data center
  - L1 provides city-wide monitoring and centralized control
  - Complex, long-term, and city-wide behavior analyses can be performed
    - E.g., large-scale event detection, long-term pattern recognition, relationship modeling, to support dynamic decision making
  - L1: city-wide response and resource management in the case of a natural disaster or a large-scale service interruption

## 3. Fog Computing Architectures and IoT

- Distributed IoT/IoE applications on the fog infrastructure



Source [ 9]: F.Bonomi, R.Milito, P.Natarajan and J.Zhu, "Fog Computing: A Platform for Internet of Things and Analytics", in N. Bessis and C. Dobre (eds.), "Big Data and Internet of Things": 169  
 A Roadmap for Smart Environments, Studies in Computational Intelligence 546, Springer Int'l Publishing 2014



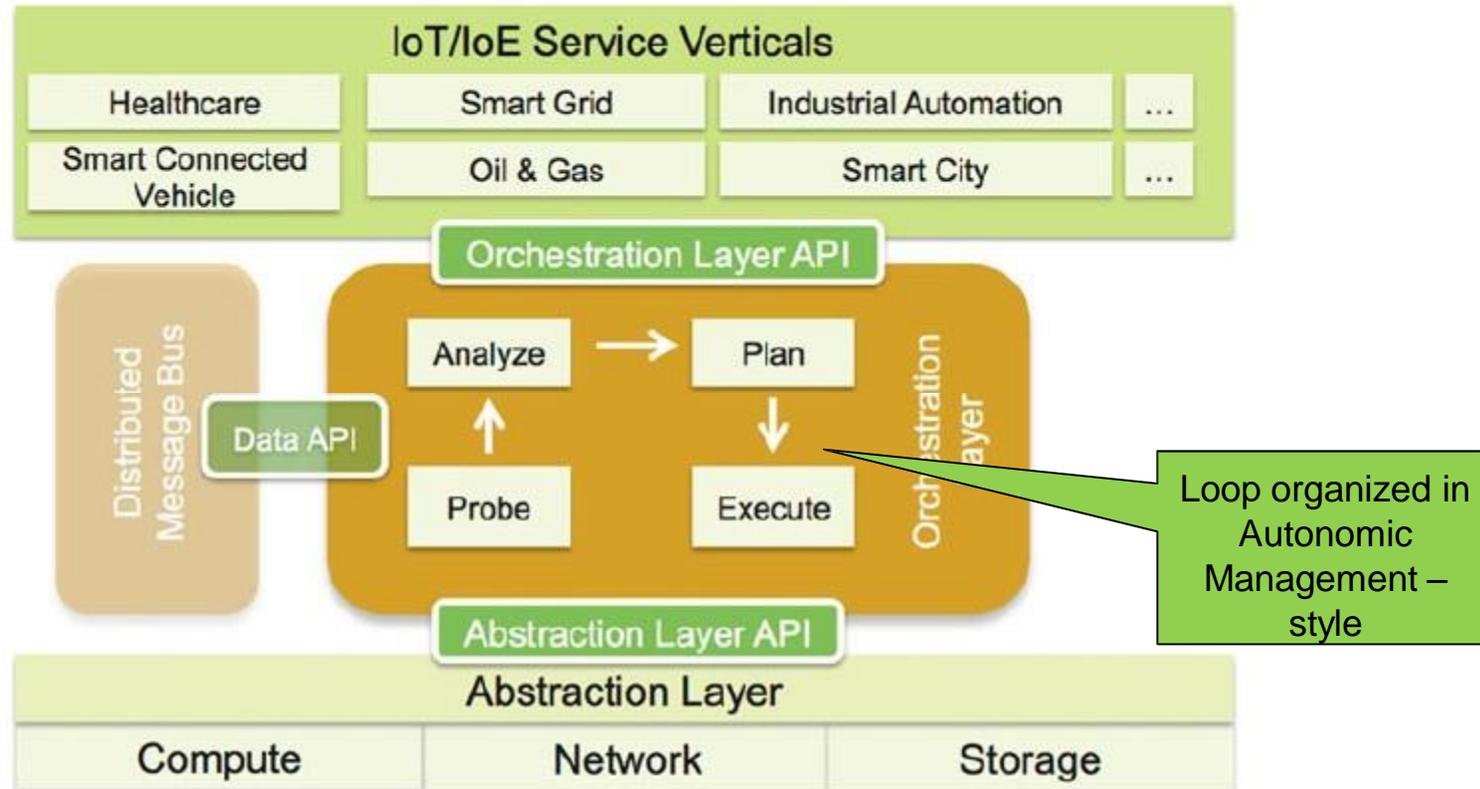
## 3. Fog Computing Architectures and IoT

---

- **Technology components needed for scalable virtualization of the resource classes:**
  - **Computing**, requiring the selection of hypervisors, to virtualize both the computing and I/O resources
  - **Storage** - needs a Virtual File System and a Virtual Block and/or Object Store
  - **Networking** - needs a Network Virtualization Infrastructure (e.g., SDN+ NFV)
- Fog leverages (similar to CC) a **policy-based orchestration** and **provisioning mechanism** on top of the resource virtualization layer for scalable and automatic resource management
- Fog architecture **should expose APIs** for application development and deployment

### 3. Fog Computing Architectures and IoT

#### Example of HW/SW Components in Fog architecture



See also Source [9]: F.Bonomi, et.al., "Fog Computing: A Platform for Internet of Things and Analytics", in N. Bessis and C. Dobre (eds.), "Big Data and Internet of Things": 169  
 A Roadmap for Smart Environments, Studies in Computational Intelligence 546, Springer Int'l Publishing 2014



## 3. Fog Computing Architectures and IoT

---

### HW/SW Components in Fog architecture (cont'd)

- ***Heterogeneous Physical Resources***
- **FCNs are heterogeneous**
  - Large range: high end servers, edge routers, access points, set-top boxes, .. to end devices such as vehicles, sensors, mobile phones etc.
  - Various HW/SW resources: processing, storage, capability to support new functionalities, OSes, software applications, etc.
- **The Fog net infrastructure is heterogeneous :**
  - High-speed links connecting enterprise data centers and the core ....to multiple wireless access technologies (ex: 3G/4G, LTE, WiFi, etc.)
- **Need an abstraction layer on top of these**

## 3. Fog Computing Architectures and IoT

---

### HW/SW Components in Fog architecture (cont'd)

- **Fog Abstraction Layer**
  - hides the platform heterogeneity and exposes a uniform and programmable interface for seamless resource Management and Control ( M&C)
  - provides generic APIs
    - for monitoring, provisioning and controlling PHY resources
    - to monitor and manage various hypervisors, Oses, service containers, and service instances on a PHY machine
    - to specify security, privacy and isolation policies for Oses or containers belonging to different tenants on the same physical machine.
  - includes support for virtualization,
    - e.g., the ability to run multiple Oses or service containers on a PHY machine and support multi-tenancy
- **Specific multi-tenancy features:**
  - data and resource isolation
  - expose a single, consistent model across PHY machine to provide these isolation services
  - exposes both the physical and the logical (per-tenant) network to administrators, and the resource usage per-tenant

## 3. Fog Computing Architectures and IoT

---

### HW/SW Components in Fog architecture (cont'd)

- **Fog Service Orchestration Layer**
  - provides dynamic, **policy-based** life-cycle management of Fog services
  - the orchestration functionality is as distributed as the underlying Fog infrastructure and services
- **Managing services** – is done with technology and components as :
  - a **SW agent, Foglet** to bear the orchestration functionality and perf. requirements that could be embedded in various edge devices.
  - a **distributed, persistent storage to store policies** and resource meta-data (capability, performance, etc) that support high transaction rate update and retrieval
  - a **scalable messaging bus** to carry control messages for service orchestration and resource management.
  - a **distributed policy engine** with a single global view and local enforcement



## 3. Fog Computing Architectures and IoT

---

### HW/SW Components in Fog architecture (cont'd)

- **Foglet Software Agent (FSA)**
  - The **distributed Fog orchestration framework** – includes several FSAs, one running on every node in the Fog platform.
  - The FSA uses **abstraction layer APIs** to monitor the health and state associated with the PHY machine and services deployed on it
  - This information is both **locally analyzed** and **also pushed** to the distributed storage for global processing
  - Foglet also performs **life-cycle mgmt.** activities (standing up/down guest OSES, service containers, and provisioning and tearing down service instances, etc.)
  - Thus, Foglet's interactions on a **Fog node span over a range of entities** starting from the **PHY machine, hypervisor, guest OSES, service containers, and service instances**
  - Each of these entities implements the necessary functions for programmatic M&C
  - Foglet invokes these functions via the abstraction layer APIs.

## 3. Fog Computing Architectures and IoT

---

### HW/SW Components in Fog architecture (cont'd)

- **Distributed Database (DDB)**
  - increases **Fog's scalability** and fault-tolerance
  - provides **fast storage and retrieval** of data
  - stores both application data and meta-data to aid in Fog service orchestration.
- **Sample meta-data examples:**
  - Fog **node's HW/SW capabilities** to enable service instantiation on a platform with matching capabilities
  - **Health and other state info of Fog nodes** and running service instances for load balancing, and generating performance reports
  - **Business policies** that should be enforced throughout a service's life cycle

### Policy-Based Service Orchestration

- The orchestration framework provides **policy-based service routing**, i.e., routes an incoming service request to the appropriate service instance that confirms to the relevant business policies
- The policy manager is responsible of this
- The policy framework is extensible



## 3. Fog Computing Architectures and IoT

---

### Policy-Based Service Orchestration Framework (cont'd)

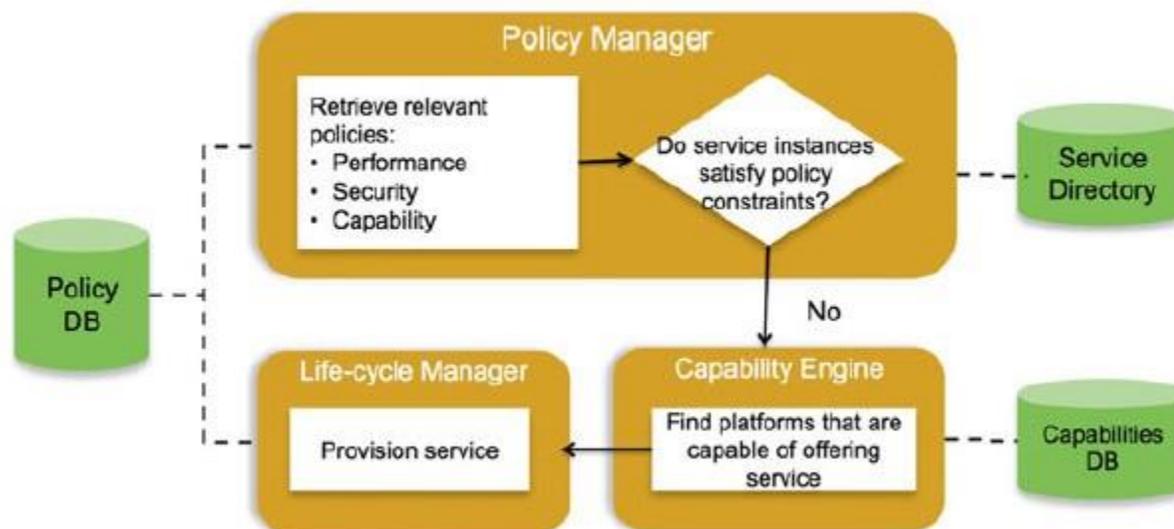
- **Administrators**
  - interact with the orchestration framework (intuitive dashboard-style user interface -UI)
  - enter business policies, manage, and monitor the Fog platform through this UI
  - The UI offers policy templates that admins can refine based on needs
- **Examples of policies :**
  - specify **thresholds** for load balancing such as maximum number of users, connections, CPU load etc.
  - specify **QoS requirements** (network, storage, compute) with a service
  - **configure device, service instance** in a specific setting
  - **associate power management capabilities** with a tenant/Fog platform
  - specify **security, isolation and privacy** during multi-tenancy
  - specify **how and what services must be chained before delivery**
    - E.g., firewall before video service



## 3. Fog Computing Architectures and IoT

### Policy-Based Service Orchestration Framework (cont'd)

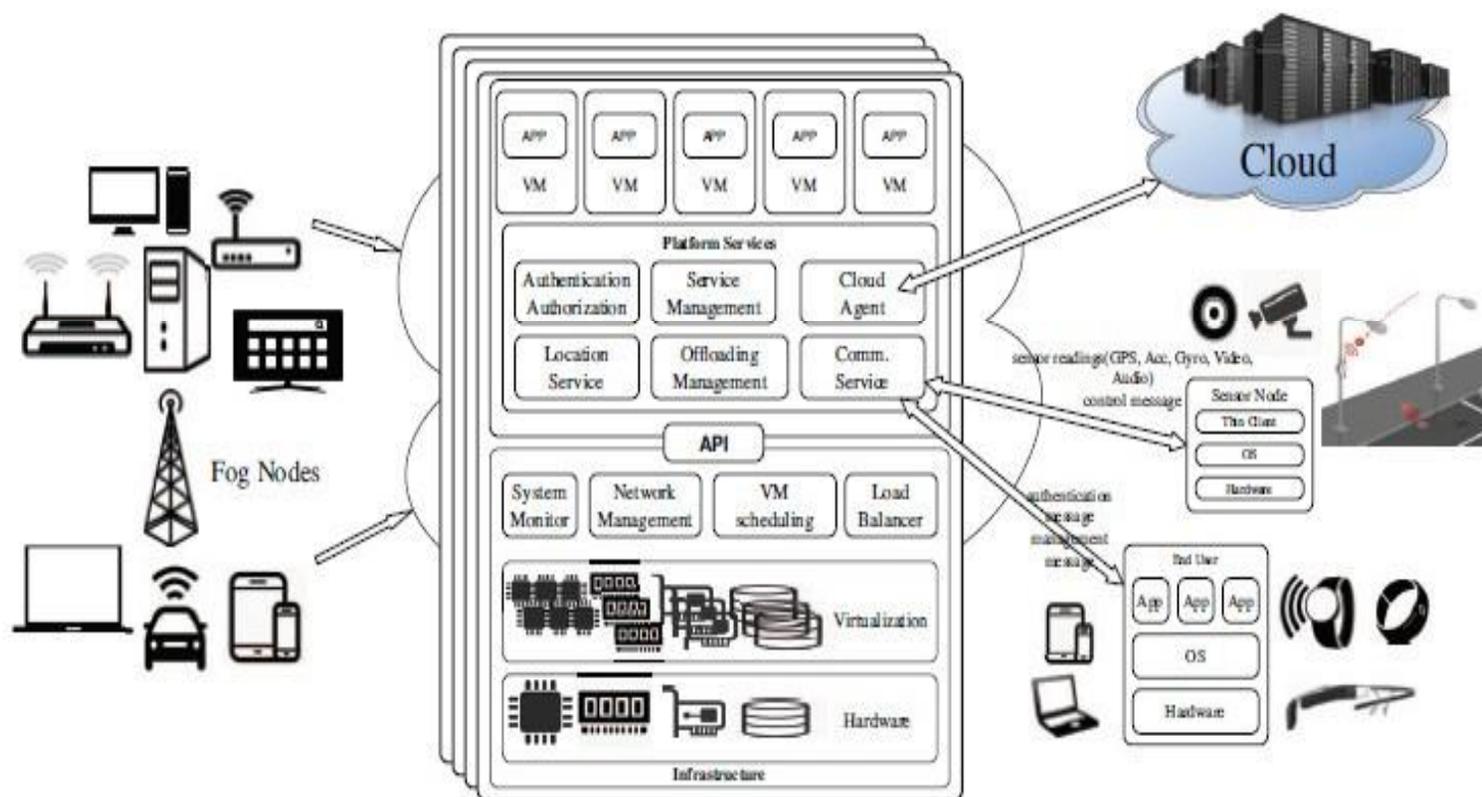
- Business policies are pushed to a distributed policy database
- The **policy manager**
  - is triggered by an incoming service request
  - gathers (from the policy repository) the **relevant policies** i.e., those pertaining to the service, subscriber, tenant etc.
  - retrieves (from the services directory) **meta-data** about **active service instances**
  - tries to **find an active service** instance that satisfies the policy constraints, and forwards the service request to that instance
  - If no such instance is available, then a new instance must be created.



The orchestration functionality is distributed across the Fog deployment such that the **logic is embedded in every Foglet**

## 3. Fog Computing Architectures and IoT

### Another view of Fog Computing architecture

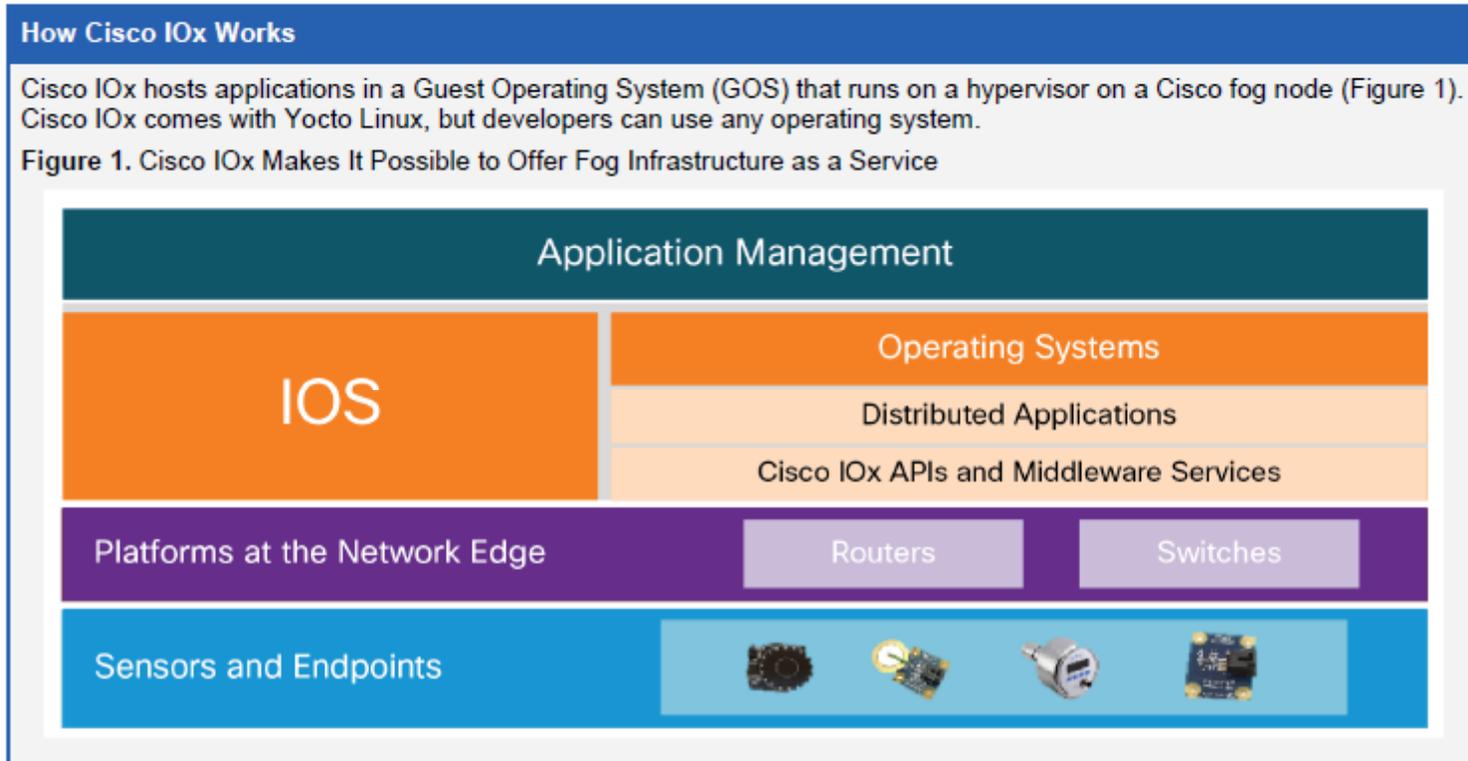


Source [11]: Shanhe Yi, et.al., "Fog Computing: Platform and Applications", 2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies,

<https://www.computer.org/csdl/proceedings/hotweb/2015/9688/00/9688a073.pdf>

# 3. Fog Computing Architectures and IoT

## Fog Computing Infrastructure as a Service- architecture example



Source [12]: White Paper, “Cisco Fog Computing Solutions: Unleash the Power of the Internet of Things”, [https://www.cisco.com/c/dam/en\\_us/solutions/trends/iot/docs/computing-solutions.pdf](https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-solutions.pdf)

1. Introduction
2. Fog/Edge Computing
3. Fog Computing Architectures and IoT
4. **➔ Mobile Edge Computing**
5. Cloudlets
6. Fog computing - MEC - Cloudlet
7. Open topics – cooperation and research
8. Conclusions

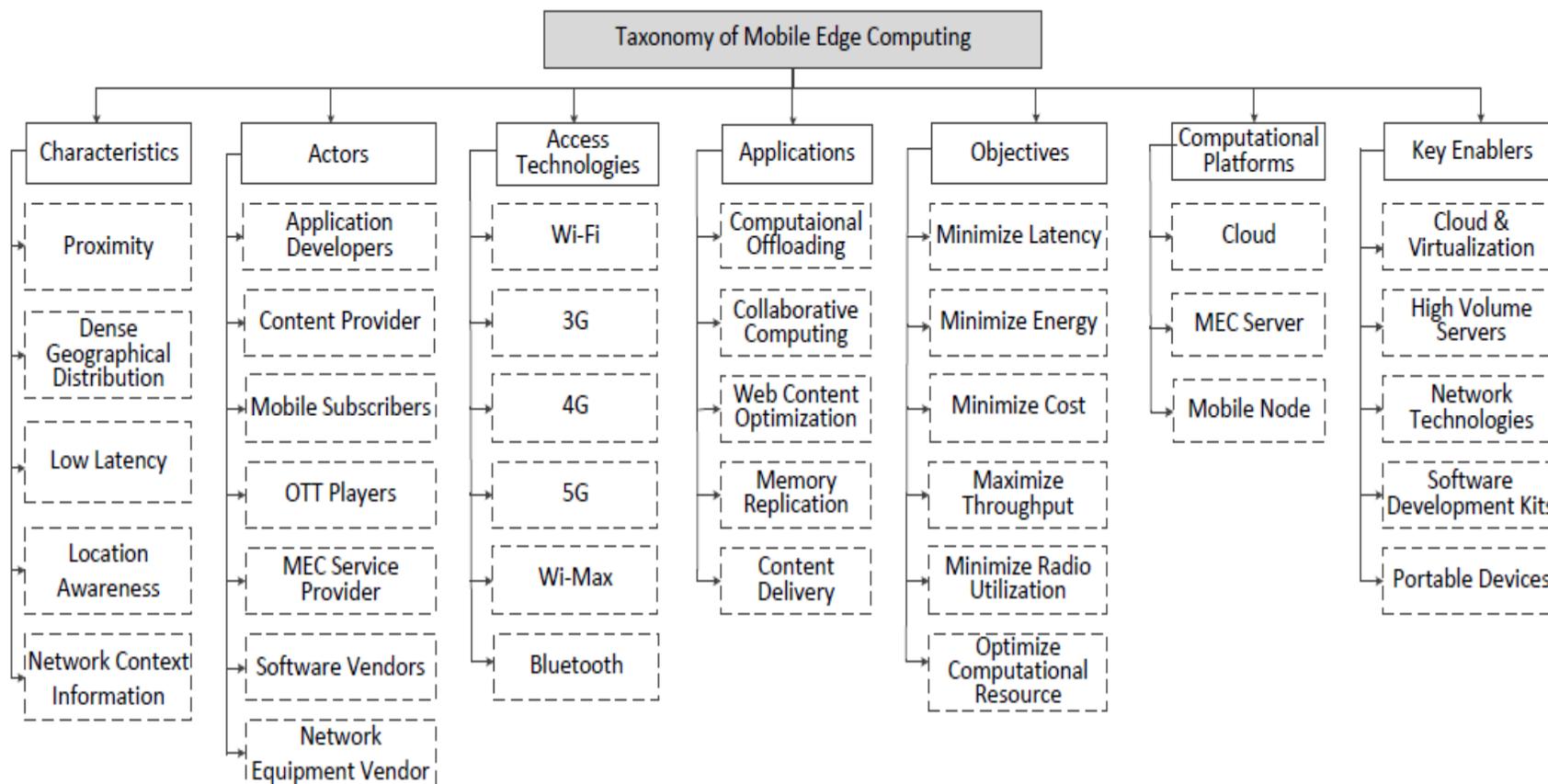
# 4. Mobile Edge Computing

---



- **Why MEC?**
  - **MEC provides IT and cloud-computing capabilities within the RAN in close proximity to mobile subscribers**
  - MEC accelerates content, services and applications responsiveness from the edge
  - **Main standardization actors:** ETSI, 3GPP, ITU-T
  - RAN edge offers a service environment with **ultra-low latency** and **high-bandwidth** as well as **direct access to real-time RAN information**
    - (subscriber location, cell load, channels load, etc.) useful for applications and services to offer **context-related services**
  - **Operators can open the radio network edge to third-party partners**
  - **Proximity, context, agility and speed** can create value and opportunities for mobile operators, service and content providers, **Over the Top (OTT)** players and **Independent Software Vendors (ISVs)**

- MEC Taxonomy

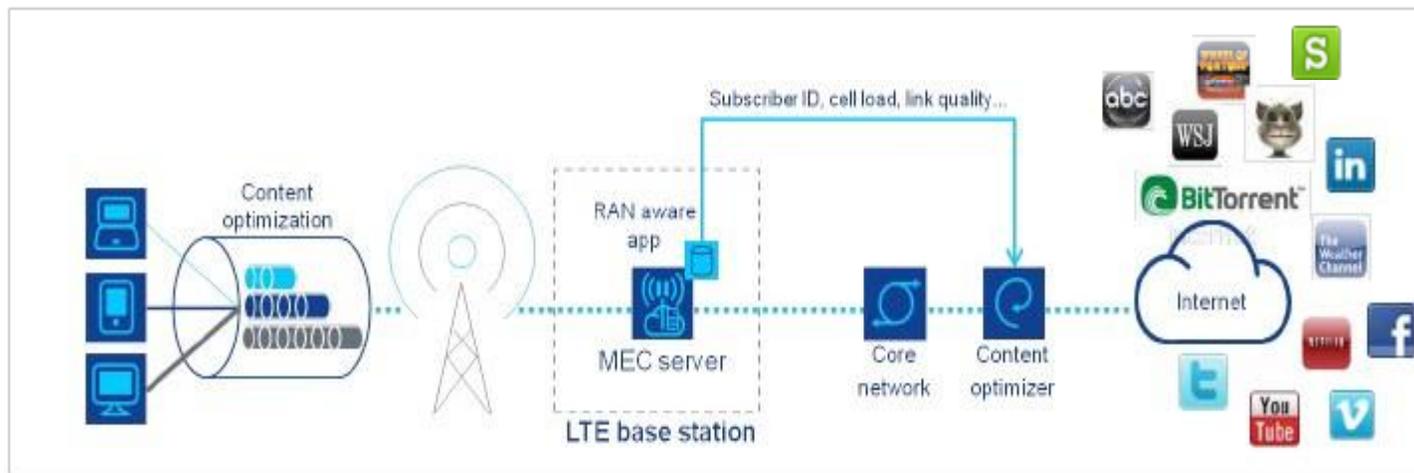


Source [13]: A. Ahmed, E. Ahmed, "A Survey on Mobile Edge Computing"  
 IEEE, Int'l Conf. on Intelligent System and Control ISCO 2016  
<https://www.researchgate.net/publication/285765997>

- **MEC Use Cases examples (content- oriented)**

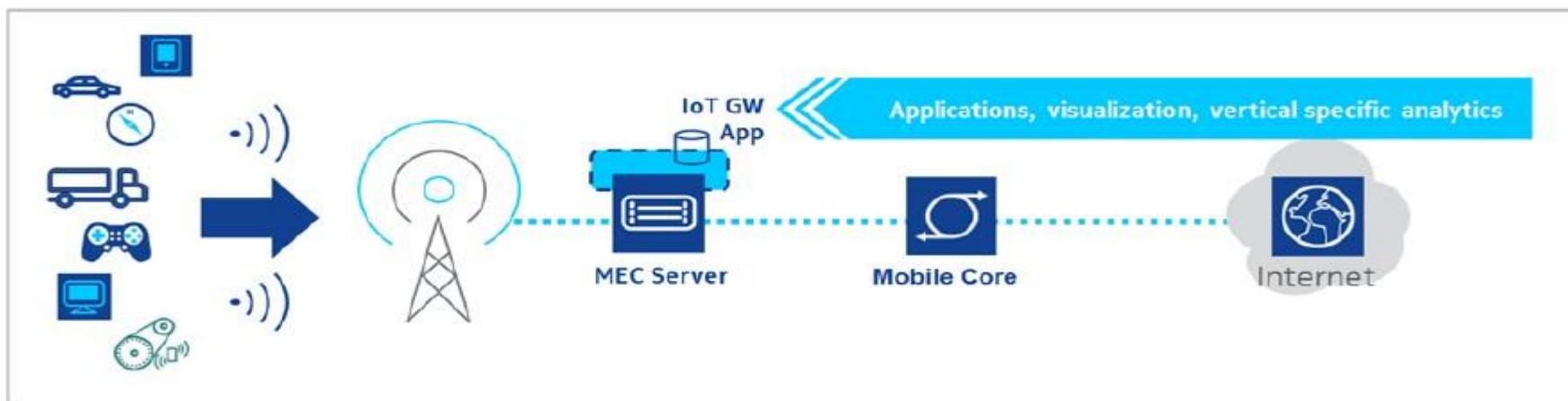
- **RAN-aware Content Optimization**

- The application exposes accurate cell and subscriber radio interface information (cell load, link quality) to the **content optimizer, enabling dynamic content optimization**, improving QoE, network efficiency and enabling new service and revenue opportunities.
- **Dynamic content optimization** enhances video delivery through reduced stalling, reduced time-to-start and 'best' video quality.



Source[13]: [https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge\\_Computing\\_-\\_Introductory\\_Technical\\_White\\_Paper\\_V1%2018-09-14.pdf](https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf)  
 Mobile-Edge Computing – Introductory Technical White Paper

- **MEC Use Cases examples**
- Internet of Things (IoT)
  - IoT generates additional messaging on telecoms networks, and requires gateways to aggregate the messages and ensure security and low latency
  - Required: real time capability; grouping of sensors and devices is needed for efficient service
  - IoT devices are often low (processor, memory capacity)-7 need to aggregate various IoT messages connected through the mobile network close to the devices
  - This also provides an analytics processing capability and a low latency response time.



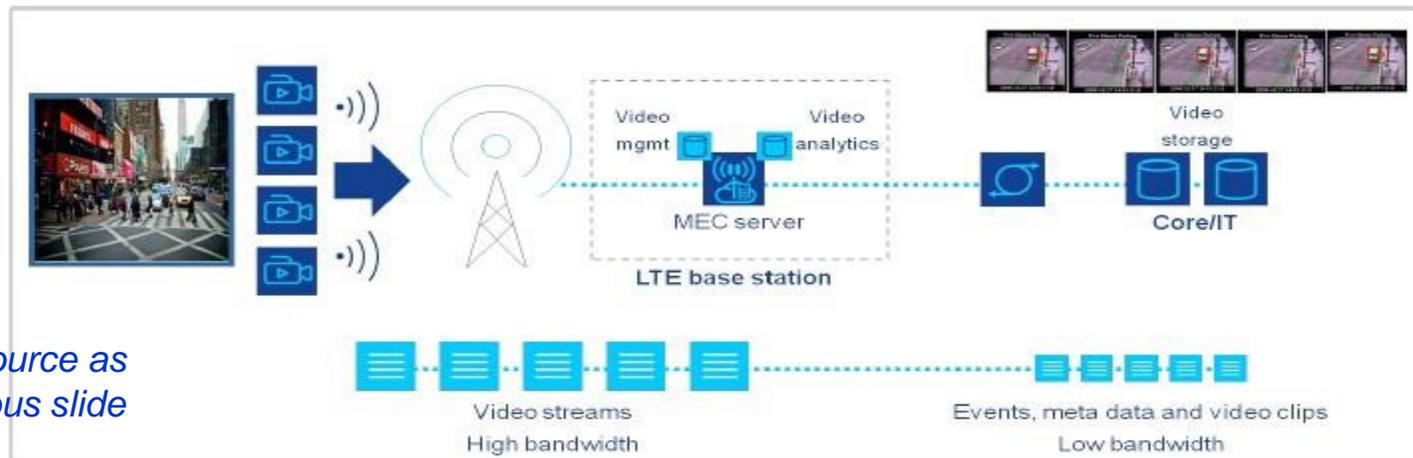
Source [14]: Yun Chao Hu et.al., "Mobile Edge Computing A key technology towards 5G" ETSI White Paper No. 11 September 2015, ISBN No. 979-10-92620-08-5



- **MEC Use Cases examples (content- oriented) (cont'd)**

- **Video Analytics**

- distributed **video analytics solution**: efficient and scalable mobile solution for LTE
    - The **video mgmt.** application **transcodes** and **stores captured video** streams from cameras, received on the LTE uplink
    - The **video analytics** application **processes the video data to detect and notify specific configurable events** e.g. object movement, lost child, abandoned luggage, etc.
    - The application **sends low bandwidth video metadata to the central operations and management server** for database searches. Applications : safety, public security to smart cities

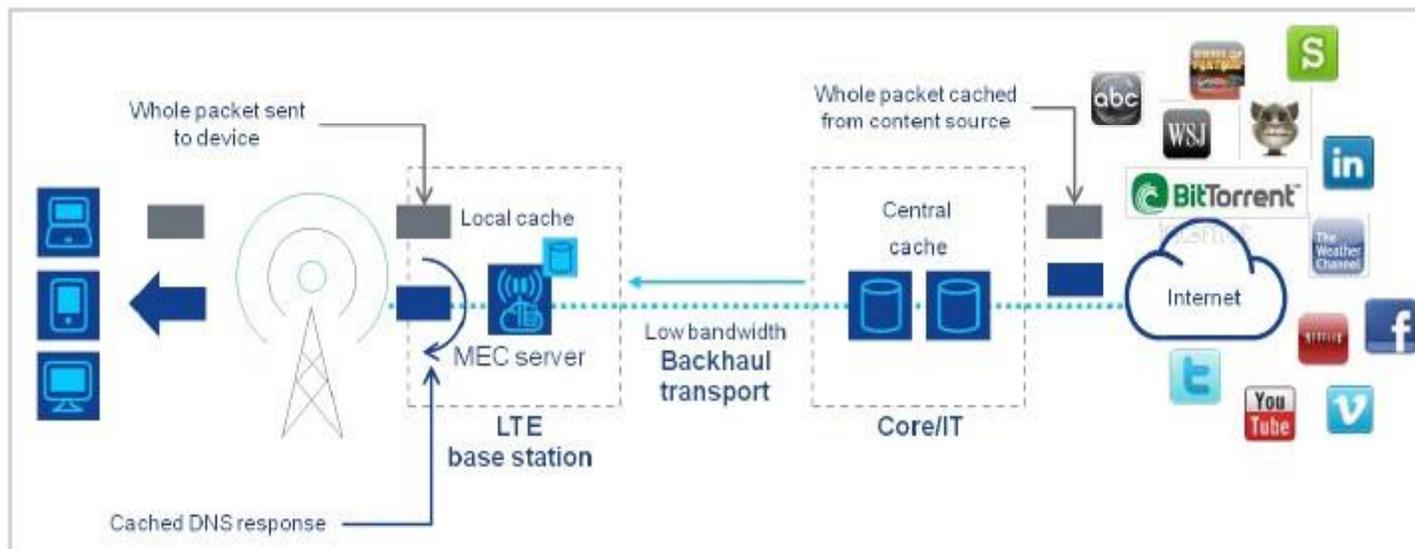


*Same source as in previous slide*

- **MEC Use Cases examples ( content- oriented) (cont'd)**

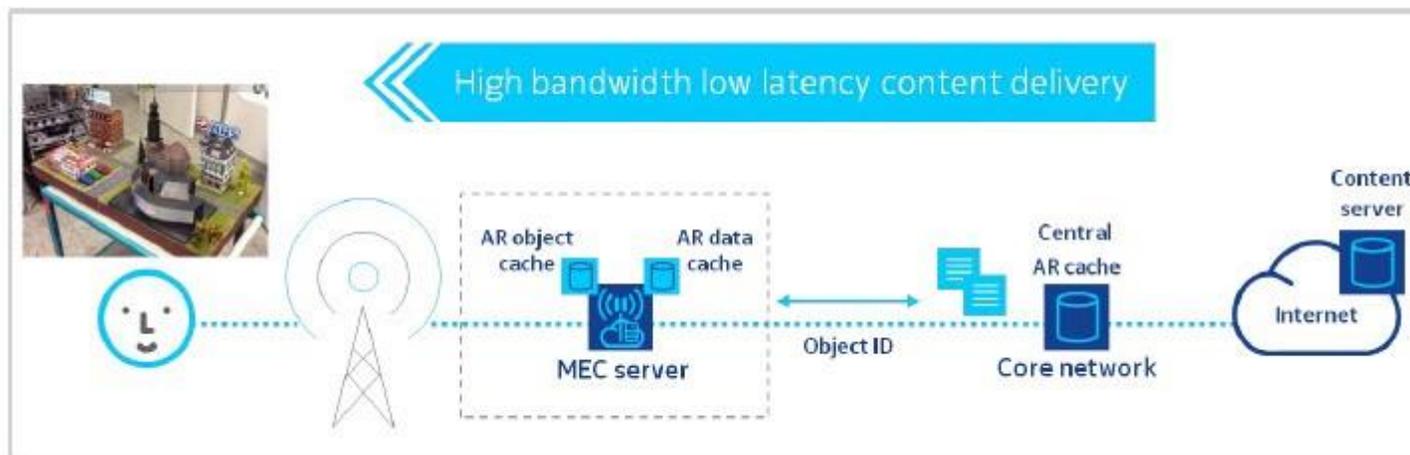
- **Distributed Content and DNS Caching**

- A distributed caching technology can provide backhaul and transport savings and improved QoE.
- Content caching could reduce backhaul capacity requirements by ~35%
- Local DNS caching can reduce web page download time by ~20%



*Same source as previous slide*

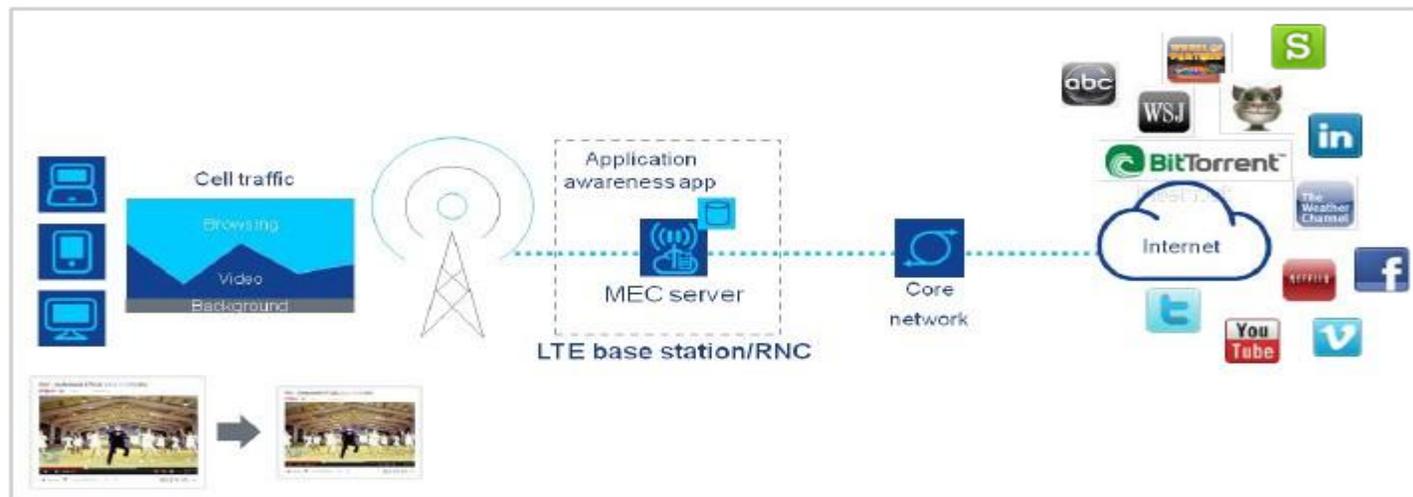
- MEC Use Cases examples (content- oriented)
- Augmented Reality (AR) content delivery
  - An AR application on a smart-phone or tablet - overlays augmented reality content onto objects viewed on the device camera
  - Applications on the MEC server can provide local object tracking and local AR content caching;
    - RTT is minimized and throughput is maximized for optimum QoE



ing event

Source [24]: [https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge\\_Computing\\_-\\_Introductory\\_Technical\\_White\\_Paper\\_V1%2018-09-14.pdf](https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf)  
 Mobile-Edge Computing – Introductory Technical White Paper

- **MEC Use Cases examples**
- **Application-aware cell performance optimization**
  - Applied for each device in real time can improve network efficiency and customer experience
  - It can reduce video stalling and increase browsing throughput
  - Reduce latency
  - Provide independent metrics on application performance (video stalls, browsing throughput, and latency) for enhanced network management and reporting

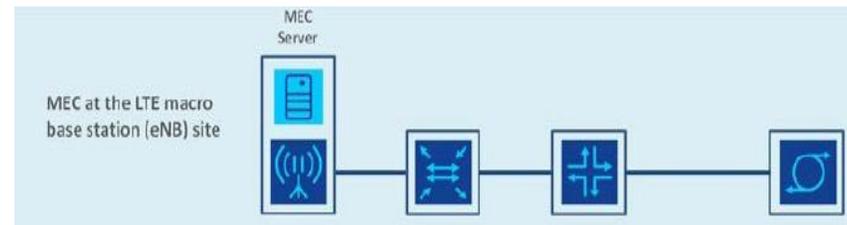


Source[24]: [https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge\\_Computing\\_-\\_Introductory\\_Technical\\_White\\_Paper\\_V1%2018-09-14.pdf](https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf)  
 Mobile-Edge Computing – Introductory Technical White Paper

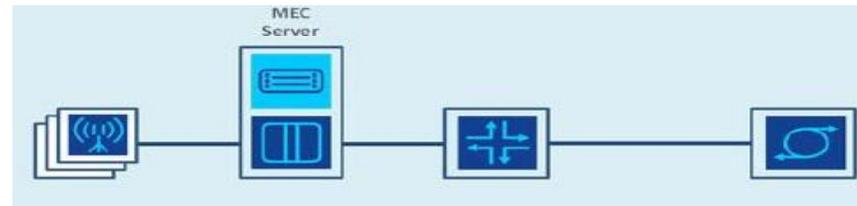
- **Possible Deployment Scenarios (ETSI)**

- The MEC server can be deployed in several variants
- Note: the multi-technology (LTE/3G) cell aggregation site can be indoor or outdoor

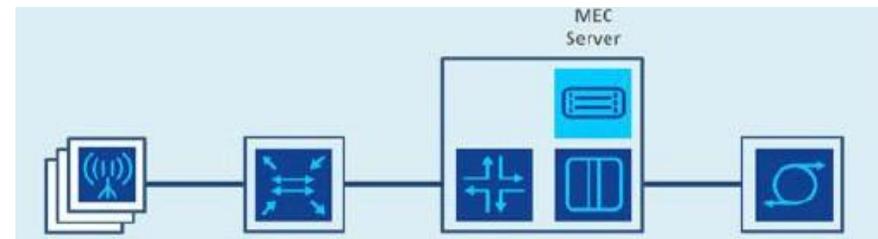
MEC at the LTE macro base station (eNB) site



MEC at the multi-technology (3G/LTE) cell aggregation site



MEC at the 3G Radio Network Controller (RNC) site

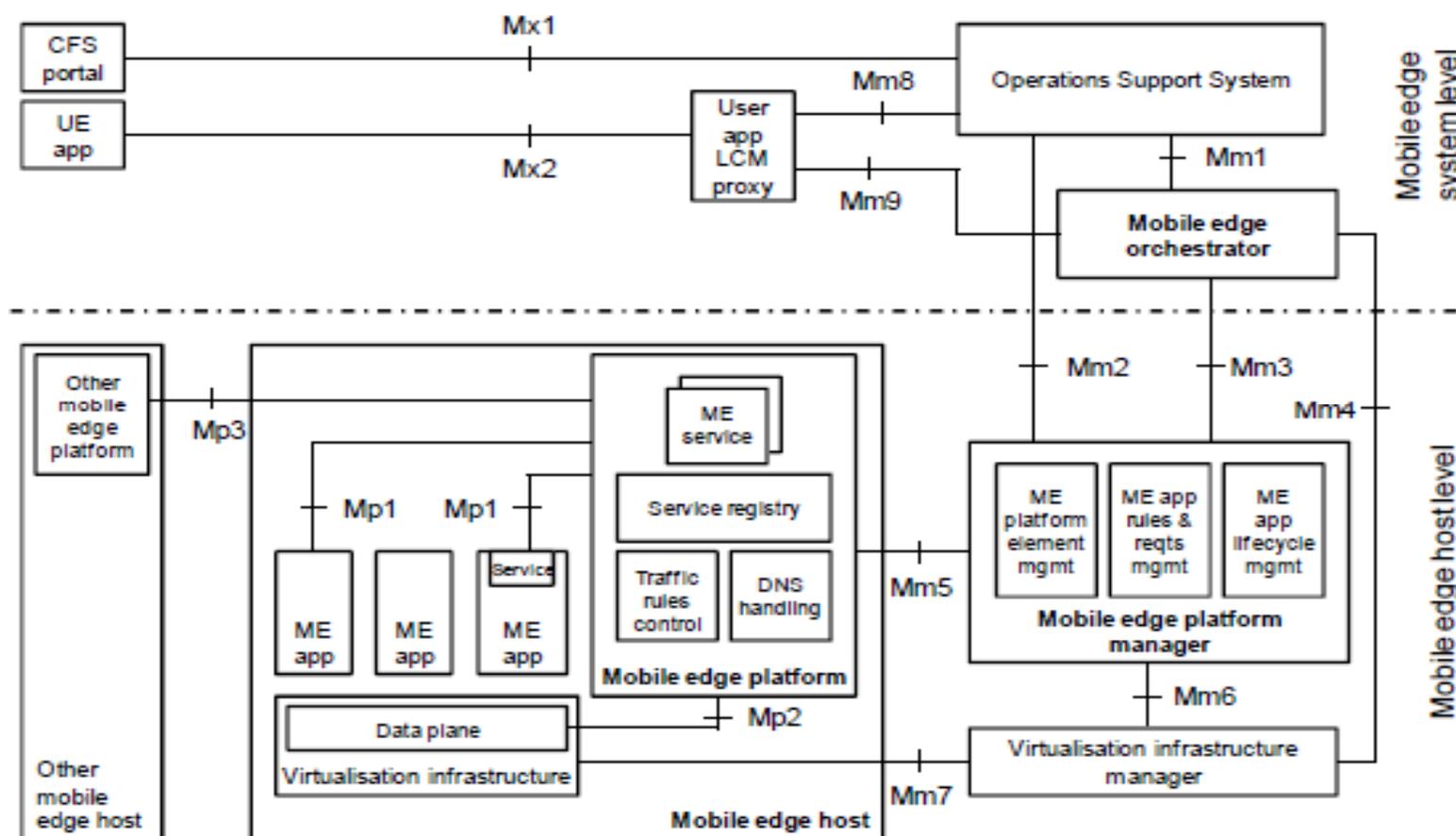


Source[24]: [https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge\\_Computing\\_-\\_Introductory\\_Technical\\_White\\_Paper\\_V1%2018-09-14.pdf](https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf)  
 Mobile-Edge Computing – Introductory Technical White Paper

- **MEC Architecture**

- MEC provides a highly distributed computing environment that can be used to deploy applications and services as well as to store and process content in close proximity to mobile users.
- Applications can benefit from real-time radio and network information and can offer a personalized and contextualized experience to the mobile subscriber.
- The mobile-broadband experience is more responsive and opens up new monetization opportunities. This creates an ecosystem where new services are developed in and around the BS
- **Key element : (MEC) IT application server** which is integrated in RAN (as above)
  - **The MEC server provides computing resources, storage capacity, connectivity, and access to user traffic and radio and network information**

- MEC Reference Architecture -ETSI



Source [21]: ETSI GS MEC 003 V1.1.1 (2016-03), "Mobile Edge Computing (MEC); Framework and Reference Architecture"

- **MEC Architecture**
- **ETSI MEC Terminology - ETSI GS MEC 001 V1.1.1 (2016-03)**
  - **ME application:** appl. **instantiable on a ME host** within the ME system and can potentially provide or consume ME services
  - **ME host:** entity **containing a ME platform** and a **virtualization** infrastructure to provide compute, storage and network resources to ME apps.
  - **ME platform:** set of functionalities
    - required to **run ME apps.** on a **specific ME host virtualization infrastructure**
    - and to **enable them to provide and consume ME** services, and that can provide itself a number of ME services
  - **ME host level management:** components handling the management of the ME specific functionality of a particular ME platform, ME host and the ME applications running on it

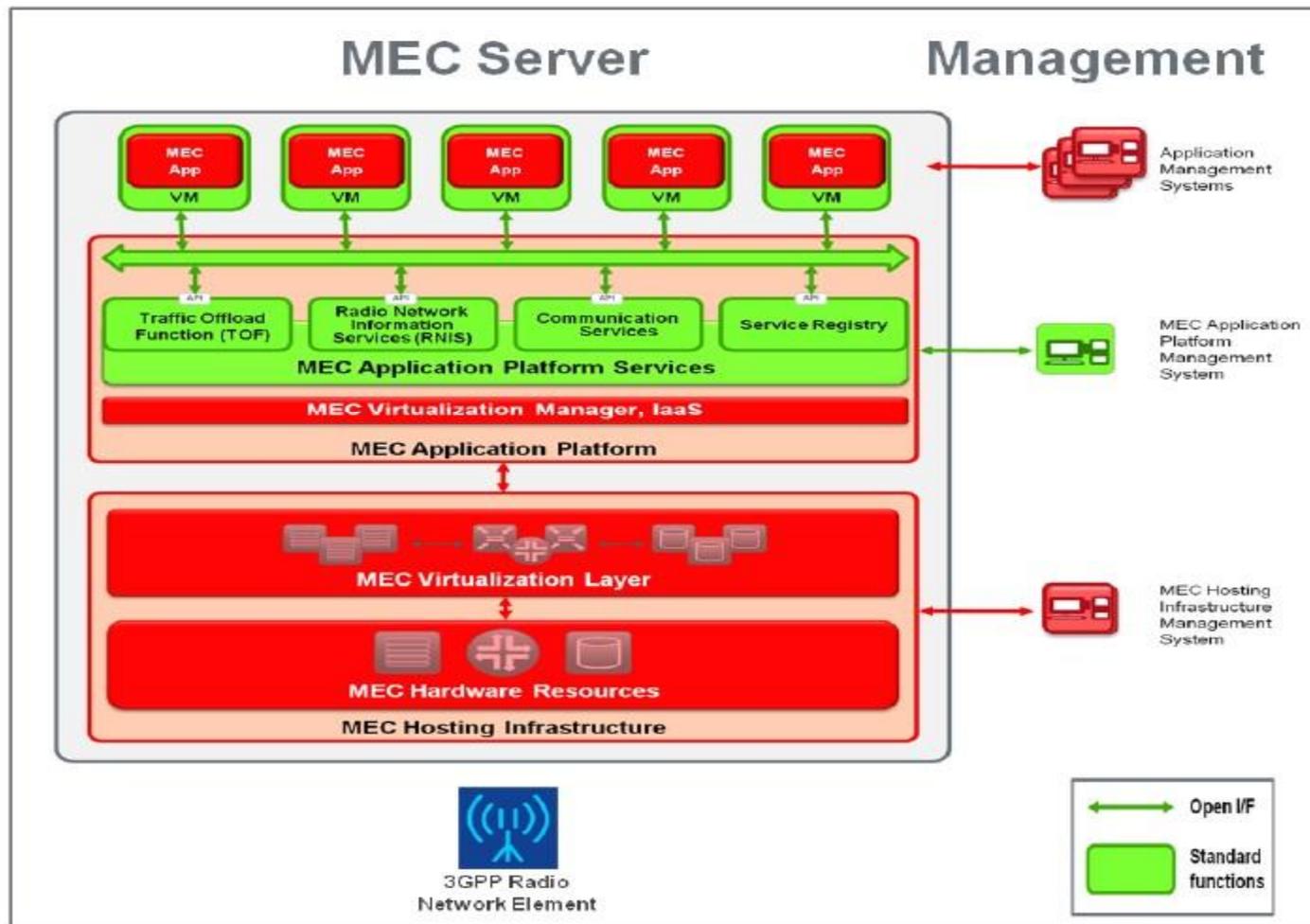
*Source [ 21]:) ETSI GS MEC 003 V1.1.1 (2016-03), "Mobile Edge Computing (MEC); Framework and Reference Architecture"*

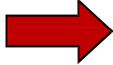


- **MEC Architecture**
- ETSI MEC Terminology ETSI GS MEC 001 V1.1.1 (2016-03)
  - **ME management:** mobile edge **system level management** and **mobile edge host level management**
  - **ME service:** service provided via the ME platform either
    - by the mobile edge platform itself
    - or by a ME application
  - **ME system:** **collection of ME hosts and ME management** necessary to run ME apps. within an operator network or a subset of an operator network
  - **ME system level management:** management components which have the **overview of the complete ME system**

*Source [ 21]:) ETSI GS MEC 003 V1.1.1 (2016-03), "Mobile Edge Computing (MEC); Framework and Reference Architecture"*

- MEC Platform Overview ( *source: ETSI*)- *NFV inspired arch*



1. Introduction
2. Fog/Edge Computing
3. Fog Computing Architectures and IoT
4. Mobile Edge Computing
5.  Cloudlets
6. Fog computing - MEC - Cloudlet
7. Open topics – cooperation and research
8. Conclusions

## What is a Cloudlet?

- **Carnegie Mellon University (CMU)** has developed **Cloudlets** [26, 27]
- A cloudlet
  - represents the **middle tier of a 3-tier hierarchy**: “mobile device – cloudlet – cloud”
  - can be viewed as a "data center in a box“, with no hard state, whose goal is to "bring the cloud closer”
  - CMU have also implemented various mechanisms as open source code which is e.g. available at [ 27]
- Related proposal- Microsoft Research [28]:
  - concept of micro datacentre as an extension of today’s hyperscale cloud datacentres (as Microsoft Azure)
  - to meet new application demands like lower latency and new demands related to devices (e.g. lower battery consumption).

Source [26] :Cloudlets: all about cloudlet-enabled mobile computing. <http://elijah.cs.cmu.edu/>

Source [27] M.Satyanarayanan, et.al., "Cloudlets: at the Leading Edge of Mobile-Cloud Convergence", 2014 6th International Conference on Mobile Computing, Applications and Services (MobiCASE)[http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=7026272](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7026272)

Source [28]: Victor Bahl, Microsoft, interview about micro datacentres, Sept 2015.  
<http://www.networkworld.com/article/2979570/cloud-computing/microsoft-researcher-why-micro-datacenters-really-matter-to-mobiles-future.html>

- **Clouplet – short overview**

- Clouplet: architectural element realizing convergence between CC and mobile computing, middle tier of the hierarchy [Cloud-clouplet-device]

- Clouplet ~ “data center in a box”

- **Main characteristics**

- **Technology:**

- based on standard cloud technology
- encapsulates offload code from mobile devices in virtual machines (VMs)
- may have specific role and functionality
- Similar infrastructure to clouds based on Openstack

- **Soft state only**

- no hard state, but may contain cached states from the cloud
- may buffer data originating from a mobile device and going to the cloud
- after installation it is entirely self-managing

- **Location**

- “Logical proximity” of the mobile devices, i.e., capable to have low E2E latency and high bandwidth (e.g., one-hop Wi-Fi)

- **Resources and connectivity**

- sufficient CPU, RAM, etc. to offload resource-intensive computations from several mobile devices
- good connectivity (bandwidth) to the cloud
- not limited by electric power supply

- 
- **Clouplet – achievements**
    - **Carnegie Mellon University** has created an open source platform
    - *Open- Stack++* (<http://elijah.cs.cmu.edu>)
      - derivative of the widely used OpenStack platform for cloud computing (<http://openstack.org>).
      - The “++” refers to the unique extensions necessary for use of OpenStack in clouplet environments.
    - Some key components of OpenStack++ such as clouplet discovery and just-in-time provisioning have already been developed and are available as open source.

1. Introduction
2. Fog/Edge Computing
3. Fog Computing Architectures and IoT
4. Mobile Edge Computing
5. Cloudlets
6. **➡ Fog computing - MEC - Cloudlet**
7. Open topics – cooperation and research
8. Conclusions

- **Initial promotion of these technologies**
  - **Fog Computing (FC)**- Cisco
  - **Mobile Edge Computing (MEC)**- ETSI ISG MEC founders: Nokia, Huawei, IBM, Intel, NTT DoCoMo, Vodafone
  - **Cloulet** - Carnegie Mellon University; later supported by Intel, Huawei, Vodafone,
- **Support from Organizations**
  - **FC - Open Fog Consortium** website: <http://www.openfogconsortium.org>
    - drive industry/academic leadership in FC architecture, testbed development, interoperability and composability deliverables that seamlessly leverage CC and edge architectures to enable E2E IoT scenarios
  - **ETSI MEC**
    - <http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing>
    - aims to unite the telco and IT-cloud worlds, providing IT and cloud-computing capabilities within the RAN;
    - ISG MEC will specify the elements required to enable apps. to be hosted in a multi-vendor MEC environment
  - **Cloulet** : [OpenEdgeComputing.org](http://openedgecomputing.org): <http://openedgecomputing.org/about-oec.html>
    - OEC offers open source code for Cloulets (extension to OpenStack)
    - Targets any industry that benefits from low-latency edge CC, which products of IoT, Tactile Internet, 5G, web content delivery, or on-line gaming, etc.
    - while limited itself to focus on a few key enablers, the spectrum of use is open



- **General note:**
  - many common characteristics
  - synergy is possible
- **General characteristics ( FC, MEC, Cloudlets)**
  - Location, access
    - Geo-distributed and usually located between end device and main data center
    - Cloudlets and even some FC may run on terminal devices
    - Usually located in BSs, APs, aggregation points, routers, switches, GWs
    - Generally – wireless access, but not excluding the fixed one
    - Enabling low latency and jitter
    - Ruggedized for outdoor usage- possible
  - Multi-tenancy of apps at the edge and use of virtualized IaaS platform
  - **Typically they are extensions of the cloud** (MEC might be independent)
  - **IoT – is a driving factor** ( services requiring distributed computing and storage)
    - Essentially for Fog computing, less for MEC and Cloudlets)
  - End device mobility support
  - **Context awareness** of the applications
    - Yes for FC and MEC, it can be added for cloudlets

- **General characteristics ( FC, MEC, Cloudlets) (cont'd)**

- **Federation of services across domains of different edge node ownership and providers**

- FC- **yes**, however need interoperability features ( Open Fog Consortium)
- MEC ( specified in stds.)
- Cloudlet – **yes** , need API included in OpenStack

- **On-line data analytics** and interaction with cloud

- FC – **yes**
- MEC, Cloudlets- **N.A**

- Support for appl. developed on **N-tier hierarchy- yes**, N = 3-FC, (2,3)-MEC, 3- Cloudlet

- **Near-real-time interaction** amongst same apps on different edge nodes

- FC- **yes**, inter-fog node communication is supporting a fully distributed application
- MEC, Cloudlets – **Partial** (so far only to support device mobility: the device disassociates from edge node 1 and associates with a new edge node 2 )

Notes:

- **Specify the need of efficient communication between edge nodes**

- FC- **yes**, MEC, Cloudlet - **No**

*Data analytics :examining raw data , to draw conclusions about that information*

*DA focuses on inference, the conclusion are derived based solely on what is already known*

*Data mining - to sort through huge data sets using sophisticated algorithms to identify undiscovered patterns and establish hidden relationships*

- **General characteristics ( FC, MEC, Cloudlets) (cont'd)**
  - **Providing APIs for provisioning and monitoring virtual resources** for compute, storage, network
    - **FC** – **Fog Abstraction Layer** - provides such APIs. Foglet SW agents use such APIs and constitute a distributed fog orchestration framework
    - **MEC**- this is done via a **Mobile Orchestrator** (borrowing from ETSI network function virtualization (NFV) MANO and service orchestration for NFV
    - **Cloudlets** - such **APIs are exposed via OpenStack** and extensions to OpenStack
  - **Life-cycle management** of distributed cloud apps
    - **FC**- Fog Service Orchestration Layer
    - Mobile Orchestrator and OSS/BSS of the telecoms network operator
    - **Cloudlets**- partially specified
  - **Support for different use cases** from multiple vertical industries
    - **FC**- yes, e.g., smart cities with smart traffic lights, energy (wind farms)
    - **MEC**- yes, e.g., security industry, content delivery industry
    - **Cloudlet**- yes , e.g. health sector, security sector, consumer services discretionary with cognitive assistance

- **Open Edge Computing (OEC) – novel general approach**
  - **Edge Computing:** **small data centers** at the **network edge** offering computing and storage resources next to the user
- **Carnegie Mellon University**- early work on Cloudlets at the edge
- **2015: overlapping interests in MEC -7 a few parties joined research efforts under the open source banner of Open Edge Computing (OEC)**
  - Currently, OEC ecosystem includes **CMU**, **Intel**, **Huawei**, and **Vodafone**
- **Main goals :**
  - To promote Cloudlets as enabling technology
  - **To drive**
    - The necessary technology for various use cases (low latency and computation at the edge) (e.g. extensions to OpenStack, KVM, QEMU).
    - prototyping of applications that leverage edge CC (ECC) pushing the boundaries and demonstrating benefits.
    - the eco system development for OEC and use current IT solutions
  - Engaging with
    - target service industries/sectors through demonstrators and joint projects
    - developer communities, seeking feedback and driving ECC acceptance
  - Synchronising work with other efforts incl. ETSI ISG MEC and OPNFV.

Source [31] <http://openedgecomputing.org/about-oec.html>

- **Open Edge Computing (OEC) : OEC status – (cont'd)**
  - **Basic Edge Server** technology – market-available from several telco vendors (~2013); ~ All vendors work on edge solutions.
  - Many telecom operators : - perform trials with edge technology
  - Some operators have already launched edge services (e.g. edge caching)
  - EC initiatives in several industries
    - ETSI Industry Specification Group **MEC** - develops req., arch. and specs
    - IT industry initiatives: **OpenStack Foundation, OpenFog**

### OEC Initiative: Mission

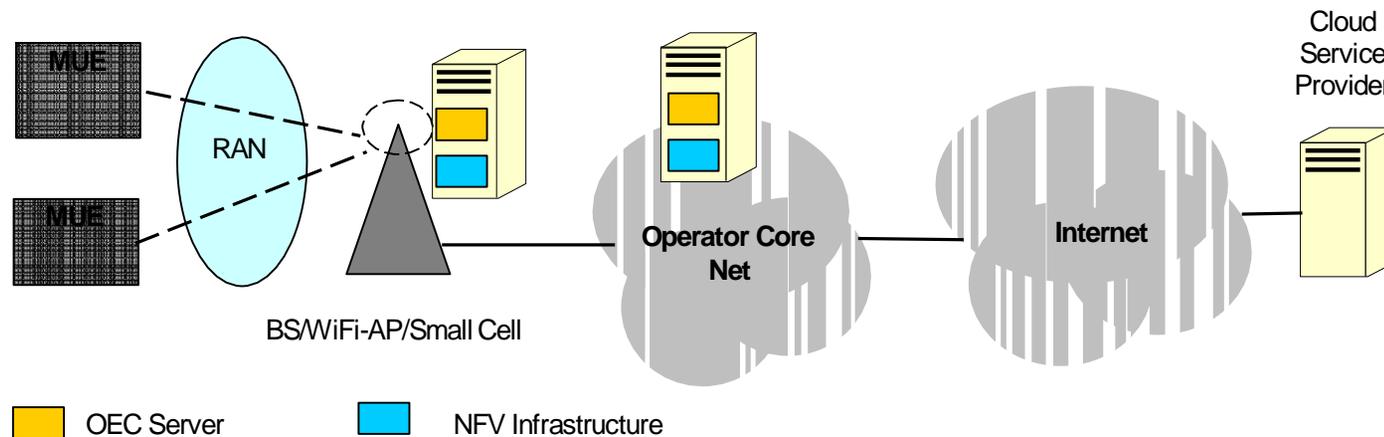


**Main target: provide an OE API and reference platform for OEC and align it with the IT industry and the telecoms industry**

*Source [32] : R.Schuster, P. Ramchandran, OPEN EDGE COMPUTING – FROM VISION TO REALITY-2016, OPNFV Design Summit, Berlin, Germany*

- **Open Edge Computing (OEC) - general approach**

- The OEC servers can be located close/associated to Base Stations, Access Points, Small Cells, ...or even in the Operator Core Network
- **Edge Computing** will utilize the **Network Function Virtualization (NFV)** infrastructure wherever possible
- This will reduce deployment cost of EC significantly

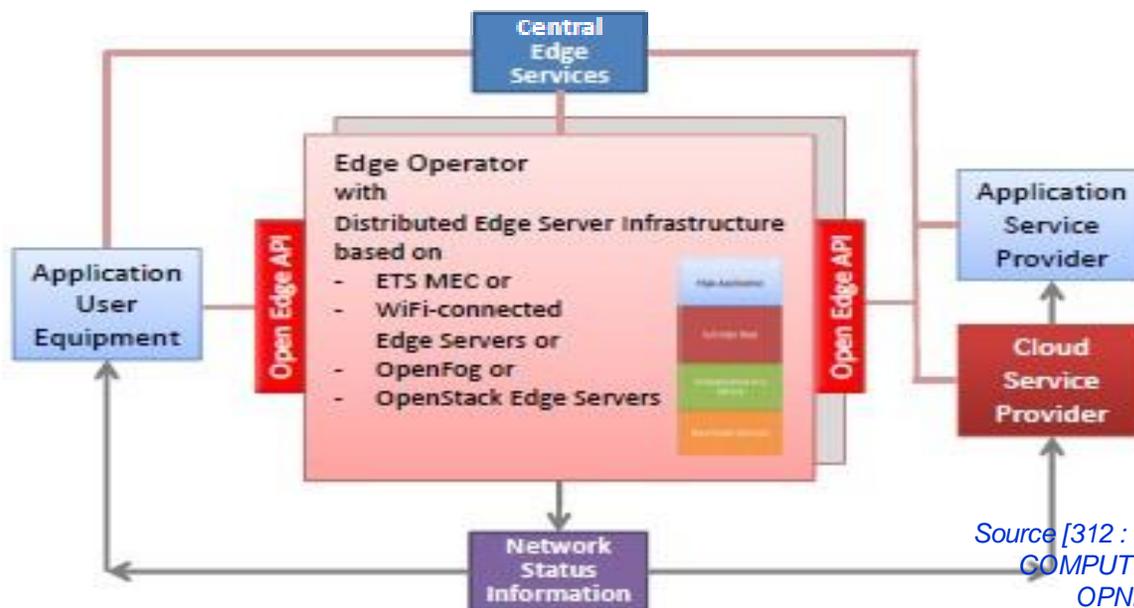


*Adapted from Source [32] : R.Schuster, P. Ramchandran, OPEN EDGE COMPUTING – FROM VISION TO REALITY -2016, OPNFV Design Summit, Berlin, Germany*

- **OEC (cont'd): Open Edge Services main Requirements**

- to have a globally agreed structure and to be available everywhere
- to be independent of
  - the communication bearer and network provider
  - the underlying technology and its provider
- open to all application categories + technologies
- to support all relevant business scenarios

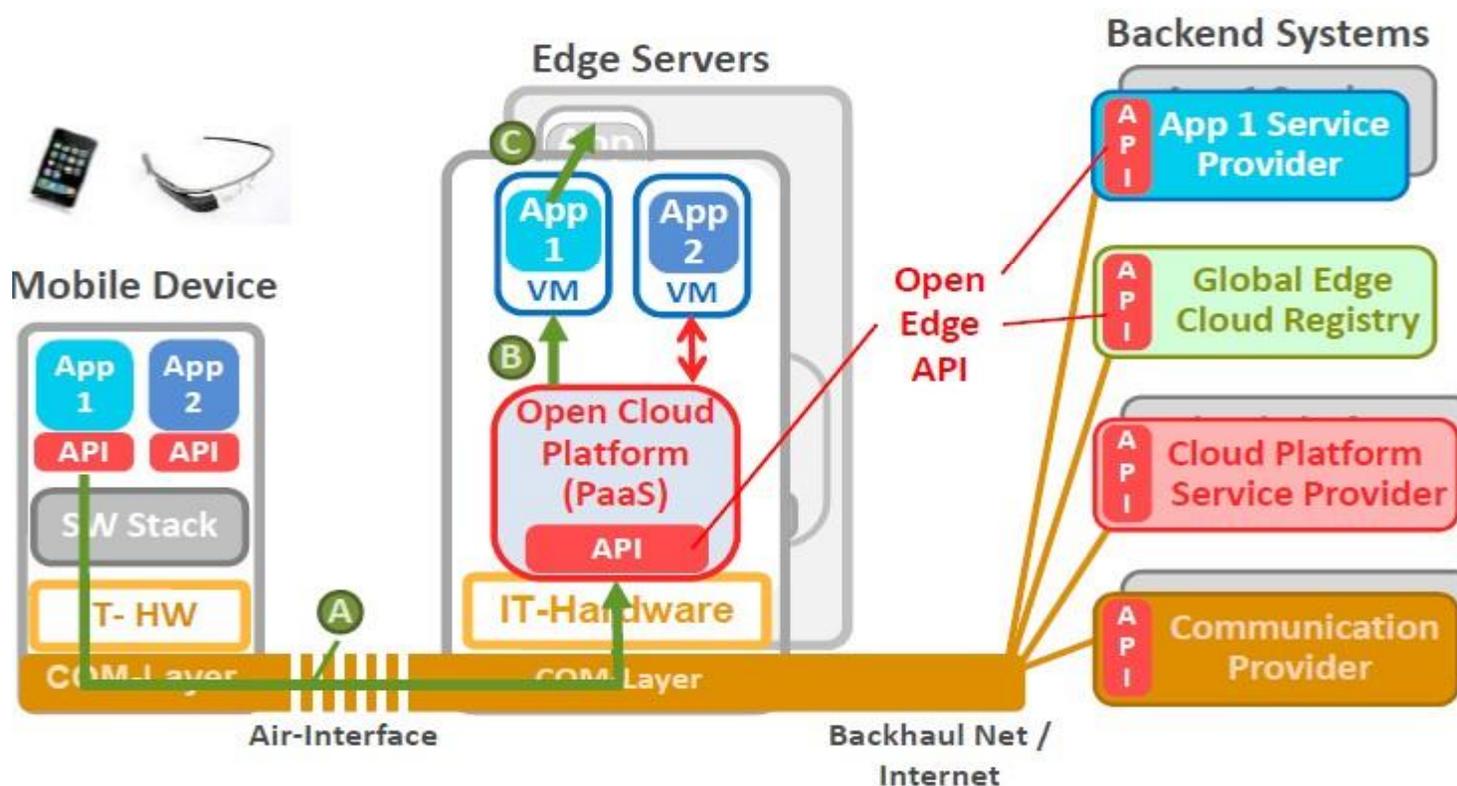
- **Edge Operator –novel business entity**, placed between the users and **App. Service Provider** or **Cloud Provider**



OEC:  
OpenEdge API

Source [312]: R.Schuster, P. Ramchandran, OPEN EDGE COMPUTING - FROM VISION TO REALITY -2016  
OPNFV Design Summit, Berlin, Germany

- **OEC (cont'd): Key Reference Platform Functions and API**
  - A -Edge Server (Cloudlet) Discovery – discover the best and closest edge server
  - B- VM Provisioning – fast provisioning of VM and app on selected edge server
  - C-VM Handoff – handoff of application to next edge server in case of movement



Source [32]: R.Schuster, P. Ramchandran, OPEN EDGE COMPUTING - FROM VISION TO REALITY -2016, OPNFV Design Summit, Berlin



1. Introduction
2. Fog/Edge Computing
3. Fog Computing Architectures and IoT
4. Mobile Edge Computing
5. Cloudlets
6. Fog computing - MEC – Cloudlet
7.  Open topics – cooperation and research
8. Conclusions

- **Business level cooperation needs**
  - Stronger **cooperation** between
    - **Fog and Edge Cloud Computing** interested players and traditional CC big players
    - **ETSI ISG MEC** and **traditional CC big players** (Microsoft, Amazon, ..) and also with OEC
  - **Bridging the gap** between “the novel concept” FC, MEC, Cloudlet) and the industrial corporate clients/actors
    - e.g. to sell the concept to industry leaders in CC, and convince such leaders to promote and introduce the new technology to their clients
  - **Conventional Cloud – to migrate towards Intelligent Cloud** (including IoT, IoE)
- **Specific steps already started-** examples
  - Expanding the coverage of the public cloud data centers into more regions
  - Adding cloud GWs close to the IoT data sources and inject data into the main cloud
  - Adding PaaS or SaaS features (e.g., machine learning software - Microsoft Azure ML)
  - Adding real-time analytics engines to the clouds
  - Adding tools for corporate clients to deploy IoT applications across the Cloud.
  - Partnering with key players in the IoT device platform market

- **Technical/research open topics**

- **Interworking** between **edge clouds**

- (in FC, ETSI MEC server, Cloudlet) and **main clouds** (e.g., Microsoft, HP, IBM, Google etc.).
- belonging to different domains (operated by different providers)

- **Design techniques for cloud-native applications** (e.g. , for IoT) to be deployable in a distributed cloud environment (e.g., IoT device, a fog platform or Cloudlet of provider X, and a main cloud).

- Example use case: smart traffic light

- **Extension of**

- IP routers to become fog nodes (e.g. Cisco IOX),
- LTE base stations to become MEC nodes (e.g. NOKIA)
- hyperscale cloud data center to reach out to the edge as well (e.g. Microsoft Azure)

- **Design of distributed cloud applications** for deployment across the chain :

- a main data center + multiple FC nodes + end points (the “things” in the Internet)

- **Develop methodology to deploy edge/fog appl. in a 3+ tier way** , similar to the smooth way, as for today’s cloud offerings (e.g. Amazon EC2, Microsoft Azure etc.)

- Fog/edge architectures have to further develop APIs for application development and deployment

---

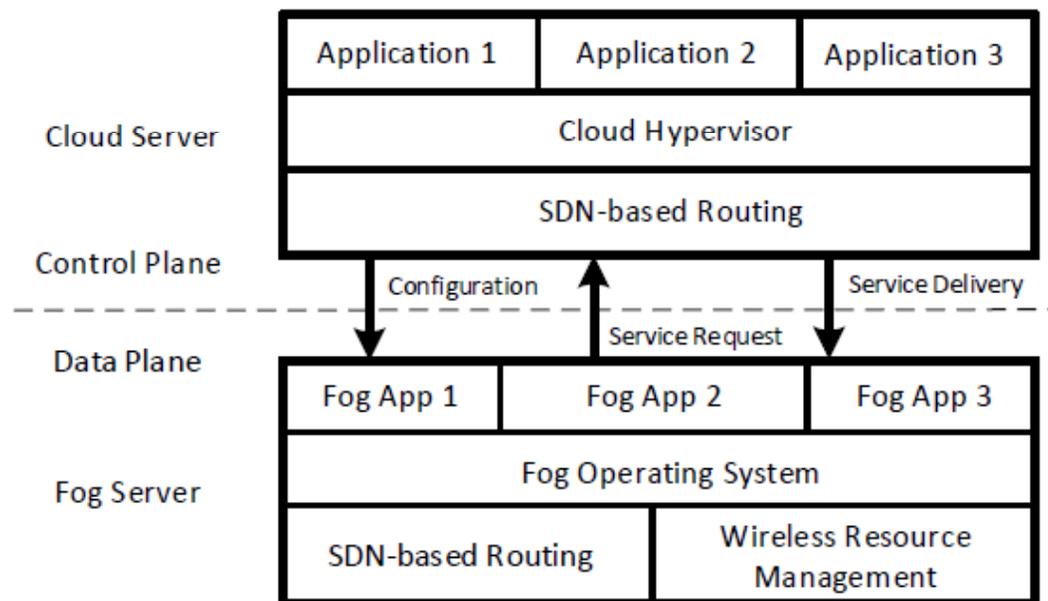
- **Technical/research open topics (cont'd)**

- Determining the needs/requirements and constraints of vertical industries for **security and privacy** in the context of distributed cloud, edge cloud and fog.
- Studying possible **operational /business models**:
  - Who would operate E2E appl. ?
  - Where/which part of the application is hosted
    - in a big public or hybrid cloud,
    - in edge cloud computing nodes and fog nodes
    - on the devices?
  - What are the **service guarantees**, **who** provides them ( SLAs?)
    - how do service guarantees fit together
    - **which party is responsible** for the E2E quality of the distributed cloud application?

- 
- **Technical/research open topics (cont'd)**
  - **Fog Computing specific issues**
  
  - **Communications between Mobile User Equipment and Fog Servers**
    - **Cross-layer Design:**
      - The **fog server manages an autonomous, network** by providing both service applications and wireless communications to mobile users.
      - It **can manage all layers and enable the cross-layer design to** provide the best service quality to users (e.g., caching a number of videos)
  
    - **Predictable User Feature and Demand:**
      - A **Fog server needs to adapt to RAN interfaces** to fully explore the localized user features and service demand.
      - It may run specific **prediction algorithms** (e.g., for shopping malls, bus, V2X, etc.) to evaluate estimate the demand.

- 
- **Technical/research open topics (cont'd)**
  - **Fog Computing specific issues (cont'd)**
  
  - **Fog Servers – Central Cloud Communications**
    - The cloud is the central controller - it manages and coordinates the distributed Fog servers
  
    - The cloud is the central information store.
    - Different Fog servers select the information contents from the cloud and then deliver the replicas from its cache to the mobile users
  
    - A cloud server manages the applications and contents for the entire system
  
    - At a particular Fog server, selective localized applications should be provisioned and synchronized with the cloud

- **Technical/research open topics**
- **Fog Computing specific issues (cont'd)**
  - **Fog - Cloud Communications (cont'd)**
  - If SDN approach is adopted then the cloud
    - manages the network with a global (SDN principle) view
    - the cloud establish the routing path of data to update the geo-distributed Fog servers (+ control plane and data plane separation).



Source [ 5]: T H. Luan et.; al. , "Fog Computing: Focusing on Mobile Users at the Edge", arXiv:1502.01815v3 [cs.NI] 30 Mar 2016

- **Technical/research open topics (cont'd)**
- **Fog Computing specific issues (cont'd)**
- **Communications between Fog Servers**
  - A Fog Server (FS) manages a pool of resources locally -7 need of collaborative service provision and content delivery among peered FSs
  - The data routing among FSs can be managed
    - either by a **centralized manner** using the SDN-based approach,
    - or, by a **fully distributed manner** ~ traditional routing mechanism (e.g., OSPF).
- **Data transmission challenges:**
  - **Service policy:**
    - the FSs at **different locations** may belong to **different business actors**
    - they may conform to **different policies** defined by owners -7 data routing among FSs needs to address the heterogeneous service policies.
  - **Topology:**
    - If FSs co-located in the same region are connected to the Internet through the same ISP with the high-rate low-cost connections,
      - **then direct communications via Internet can alleviate the traffic between cloud and FSs**
  - **Connection:**
    - Optimization problem is open - the data routing among FSs (wired connections over Internet or wireless connections through opportunistic connections)



- 
- **Technical/research open topics (cont'd)**
  - **Fog Computing specific issues (cont'd)**
  - **Fog Computing Deployment**
    - Additional computing and storage resources at the edge
    - The **FS needs to adapt its services** -7 extra management and maintenance cost.
    - The FC operator Should address challenges
      - **Application:** customize the applications embedded in each of the FS based on the local demand.
      - **Scaling:** anticipate the demand of each of the FSs and deploy adequate fog resources so as to sufficiently provision.
      - **Placement:** A group of FS can collaboratively provide service applications to mobile users nearby.

---

## Technical/research open topics

- **Fog Computing specific issues**
- **Supporting Technologies: 5G, SDN, NFV**
  - 5G: The Fog layers can be adapted by using the existing accessing networks, e.g., WiFi, or emerging 5G wireless technologies with a virtualized architecture
  - **Network Function Virtualization (NFV)** - enable VNFs inside network nodes, e.g., switches and routers
  - FC enable virtualized location-based applications at the edge device and providing desirable services to localized mobile users.
  - With a global network view, the cloud can manage the entire network using a SDN approach

---

- **Technical/research open topics (cont'd)**

- **Mobile Edge computing – specific issues**

- **Standardized open environment** is needed to **be further developed** to allow seamless and proficiently integration of traditional applications across the MEC platform
- **simulation platform** to experiment for various MEC scenario can cut costs of development
- implementing a **mobility management** (vertical and horizontal) allowing users to seamlessly access edge applications
- **Heterogeneous access** management : 3G, 4G, 5G, WiFi, WiMAX, BlueTooth, ..
- **Combining MEC with Fog in CRAN/H-CRAN architectures**
- Pricing models
- **Scalability** assurance ( application migration, load balancing, ..)
- **Security problems** for applications running in MEC

- **Cloudlet specific issues:**

- **optimal cloudlet selection and seamless cloudlet handoff ?**
- OpenStack++
  - effort to become a universally deployable cloudlet platform
  - above and below which many proprietary HW/SW and service innovations can emerge

- **Technical/research open topics (cont'd)**
- **Identified by OEC Initiative**
- **ETSI ISG MEC to continue work on**
  - Requirements for MEC , Edge arch. focusing on mobile BS
  - Standard API's for telecom related edge applications
- **Current Gaps:**
  - **Customer Facing Service**
  - **mobile edge system level management**
    - **Mobile edge orchestration, OSS Operations Support System**
    - **User application life cycle management proxy**
  - **mobile edge host**
    - Deployment model [one instance per /user or host or edge] – topology, cost etc.
    - Latency and location requirements
    - connectivity or mobility requirements (e.g. app. state relocation, app. instance relocation);
  - **mobile edge platform**
    - **offering an environment** where the mobile edge applications can discover, advertise, consume and offer mobile edge services
    - receiving traffic rules from the mobile edge platform manager, apps, or services, and instructing the data plane accordingly
    - hosting mobile edge services
    - providing access to persistent storage and time of day information

- 
- **Technical/research open topics**
  - **Identified by OEC Initiative (cont'd)**
    - **Mobile edge platform management**
      - **managing the life cycle** of applications including informing the mobile edge orchestrator of relevant application related events;
      - **providing element management functions** (site, cluster, fog) to the mobile edge platform;
      - **managing the application rules** and requirements including service authorisations, traffic rules, DNS configuration and resolving conflicts.



1. Introduction
2. Fog/Edge Computing
3. Fog Computing Architectures and IoT
4. Mobile Edge Computing
5. Cloudlets
6. Fog computing - MEC - Cloudlet
7. Open topics – cooperation and research
8.  **Conclusions**

# Conclusions



- There are some differences, however FC, MEC, Cloudlets share a similar vision
- Main domain of applications and fields:
  - Internet of Things, Internet of Everything, Smart cities
  - Tactile Internet, and existence of appropriate wireless connectivity ( low data rate minimal power consumption at a device)
  - 5G networks and related services
  - Big data wit near real-time response
- The current paradigm of **bigger and more consolidated, more hyperscale cloud computing data centers will fall short of future industry needs**
- It is required an extension of cloud computing to the edge of networks
- Many players have recognized this, as reflected in the existence of concepts like
  - **Fog Computing, Mobile Edge Computing and Cloudlets (Open Edge Computing)**
- For the time being - the trend is less supported less by big public cloud providers like HP, IBM, Google, Microsoft, or Amazon.
- However, cooperation Cloud- Middleware – devices seems to be the best tradeoffs to take benefit of powerful CC and also solve the scalability, real-time and bandwidth problems of a lot of applications

- 
- **FC, MEC, Cloudlet** approaches can well serve and play a role in specific domain of Future Internet and especially in IoT, IoE, ...:
  - **Connected vehicles**, V2V, V2X. Automotive safety services (like ice on motorway real-time warning, platooning, coordinated lane change manoeuvres etc.)
  - Services in **infotainment** (e.g. in automotive)
  - **Safety and emergency** systems
  - **Mission critical** systems
  - **Smart city** system components like smart traffic lights (beyond what's available today)
  - **Big data and analytics** for sectors like industrial: real-time analytics at the edge, long-term analytics in the main cloud, for purposes like predictive maintenance and others.
  - **Smart grid**, closed loop system (a degree of central processing occurs at the edge with special requirements on stability and rapid response).
  - Efficient operation of **wind farms**: semi-autonomous controllers at each wind turbine are orchestrated on the level of the wind farm by software at the edge
  - **Robotics for various sectors** including assisted living, remote diagnostics etc.
  - Any sector that makes use of **wireless sensor systems**, (e.g. oil & gas or building industry)





# List of Acronyms

---

•	ACE	Ancestral Communication Entity
•	AVC	Audio Video Conference
•	BBU	Baseband Processing Unit
•	BS	Base Station
•	BSC	Base Station Controller (2G/3G)
•	BSS	Business Support System
•	CC	Cloud Computing
•	CCN	Content Centric Networking
•	CDN	Content Delivery Network
•	COTS	Commercial-off-the-Shelf
•	CoMP	Coordinated multi-point
•	CP	Content Provider
•	CRAN	Cloud RAN
•	DASH	Dynamic adaptive streaming over HTTP
•	DRM	Digital Rights Management
•	EC	Edge Computing
•	EPC	Evolved Packet Core
•	ETSI	European Telecommunications Standards Institute
•	FC	Fog Computing
•	FCN	Fog Computing Node
•	HCRAN	Heterogeneous CRAN
•	HPN	High Power Node
•	HSPA	High Speed Packet Access
•	HTTP	Hyper Text Transport Protocol
•	IaaS	Infrastructure as a Service



# List of Acronyms (Cont.)

---

•	IMS	IP Multimedia System Industry
•	ISG	Specification Group.
•	IT	Information Technology Kernel-
•	KVM	based Virtual Machine Low
•	LPN	Power Node
•	LTE	Long Term Evolution
•	MEC	Mobile Edge Computing
•	MBMS	Multicast Broadcast Media Service
•	M&O	Management and Orchestration
•	MME	Mobility Management Entity
•	NF	Network Function
•	NFV	Network Functions Virtualization
•	NFVI	Network Functions Virtualization Infrastructure
•	NO	Network Operator
•	NP	Network Provider
•	NS	Network Service
•	OEC	Open Edge Computing
•	OSS	Operations Support System
•	OSPF	Open Shortest Path First
•	PaaS	Platform as a Service
•	PoC	Proof of Concept Radio
•	RAN	Access Network Remote
•	RRH	Radio Head
•	RNC	Radio Network Controller
•	QEMU	Quick Emulator
•	SaaS	Software as a Service
•	SDN	Software Defined Network
•	SLA	Service Level Agreement
•	S/P-GW	Serving and Packet Data Networks Gateway
•	SP	Service Provider
•	TCP	Transmission Control Protocol
•	UDP	User Datagram Protocol
•	VM	Virtual Machine
•	VNF	Virtual Network Function