CRAAX Lab

Driving towards a reliable edge

Lessons from the H2020 mF2C project

Xavi Masip (@CRAAX at UPC)

DRCN 2019, Coimbra, March 20, 2019

Outline

- First…..why the edge?
- Fog vs Edge: mF2C view
- Reliability: A must
- Fog node as a concept
- Combining Fog and Cloud
- Making reliability a nightmare
  - A real ongoing effort: The EU H2020 mF2C project

The context


212 BILLION
Total number of available sensor enabled objects by 2020
212B is 28x the total population of the world
30 BILLION
Sensor enabled objects connected to networks by 2020

Internet of Things:
What If, We Deliver a 1% Improvement?

Industry	Segment	Type of Savings	Estimated Value Over 15 Years (Billion nominal USD)
Aviation	Commercial	1% fuel savings	$30B
Power	Gas-fired Generation	1% fuel savings	$66B
Healthcare	System-wide	1% Reduction in System Inefficiency	$63B
Rail	Freight	1% Reduction in System Inefficiency	$27B
Oil & Gas	Exploration & Development	1% Reduction in Capital Expenditures	$90B
Total			$276B

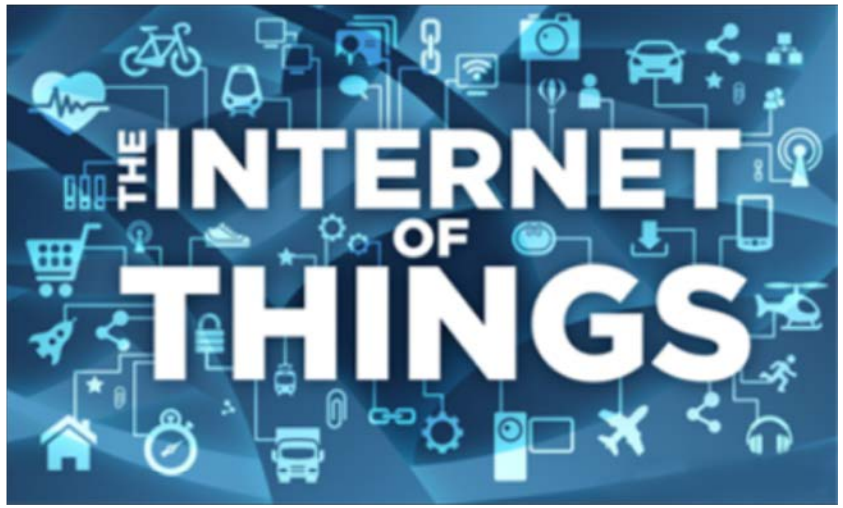
Source: "Industrial Internet: Pushing the Boundaries of Minds and Machines," GE, November 26, 2012

Old view

…not bad at all

CRAAX Lab

Challenges?




Not that good though
the light is at the edge


THE
FOG COMPUTING

Main actors

industrial internet CONSORTIUM

OpenFog

mF2C

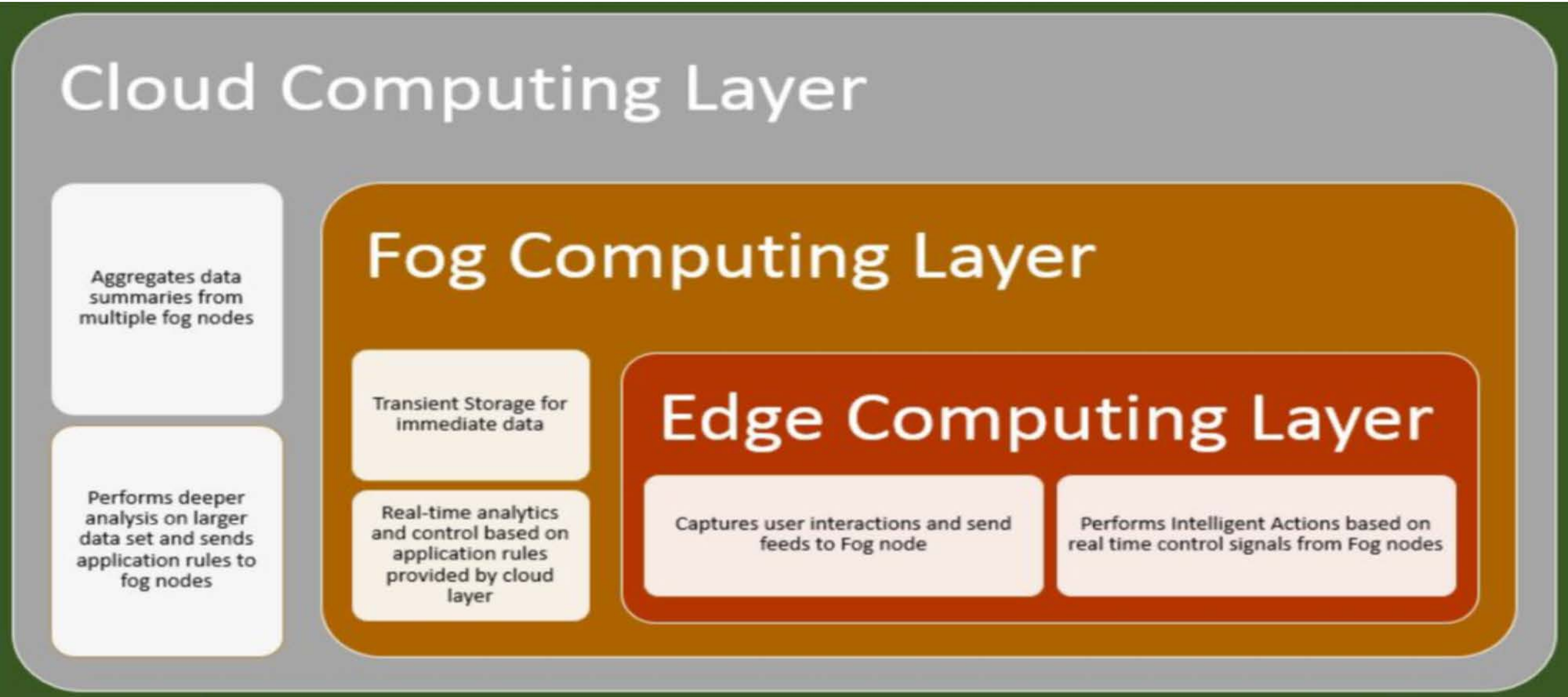
openEDGEcomputing

ETSI

Multi-access Edge Computing (MEC)

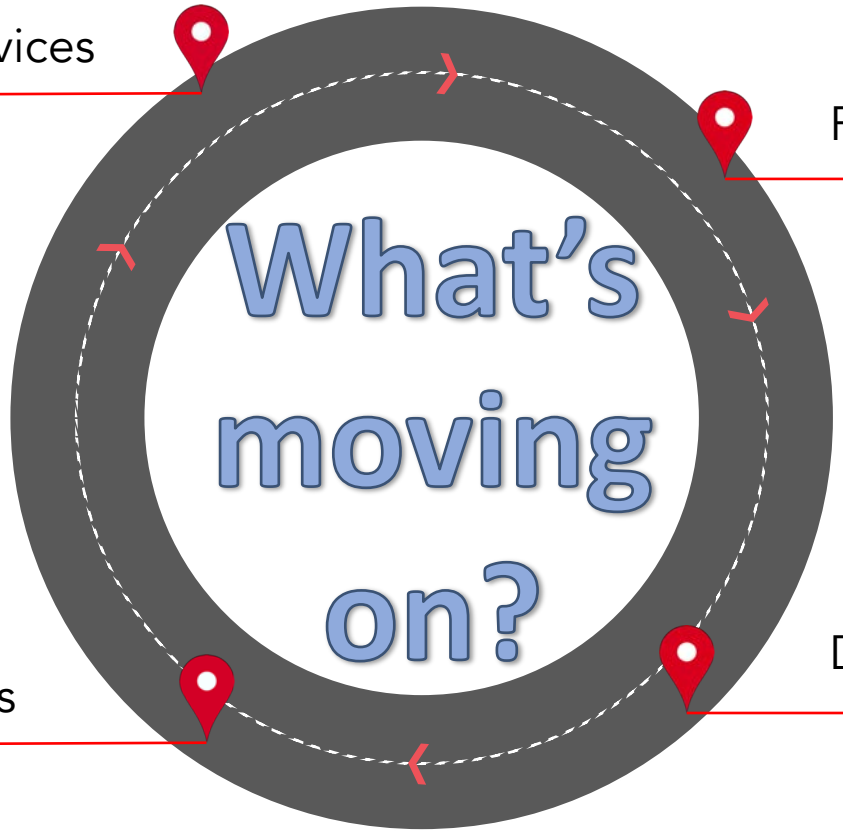
CRAAX Lab

Strategy?...moving to the edge

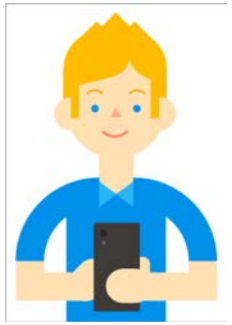


The edge, why?

Services



Processing



Users

Data

The edge, why?

Proximity

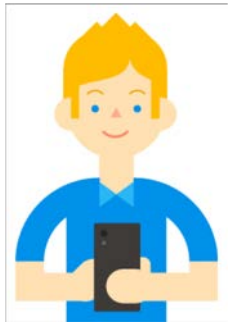
Services

Processing

Nice for

Users

Data



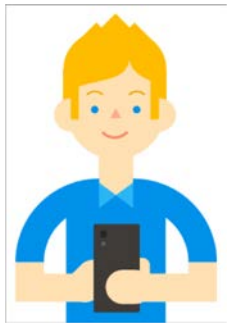
users

The edge, why?

Proximity

A must for telcos

The edge, why?



Service quer

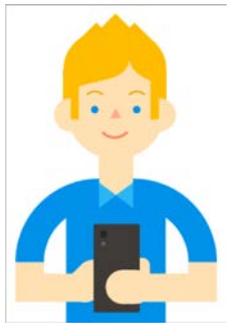
Outcome

Data

Data query

Cloud

The edge, why?



Service query

Outcome

Access Network

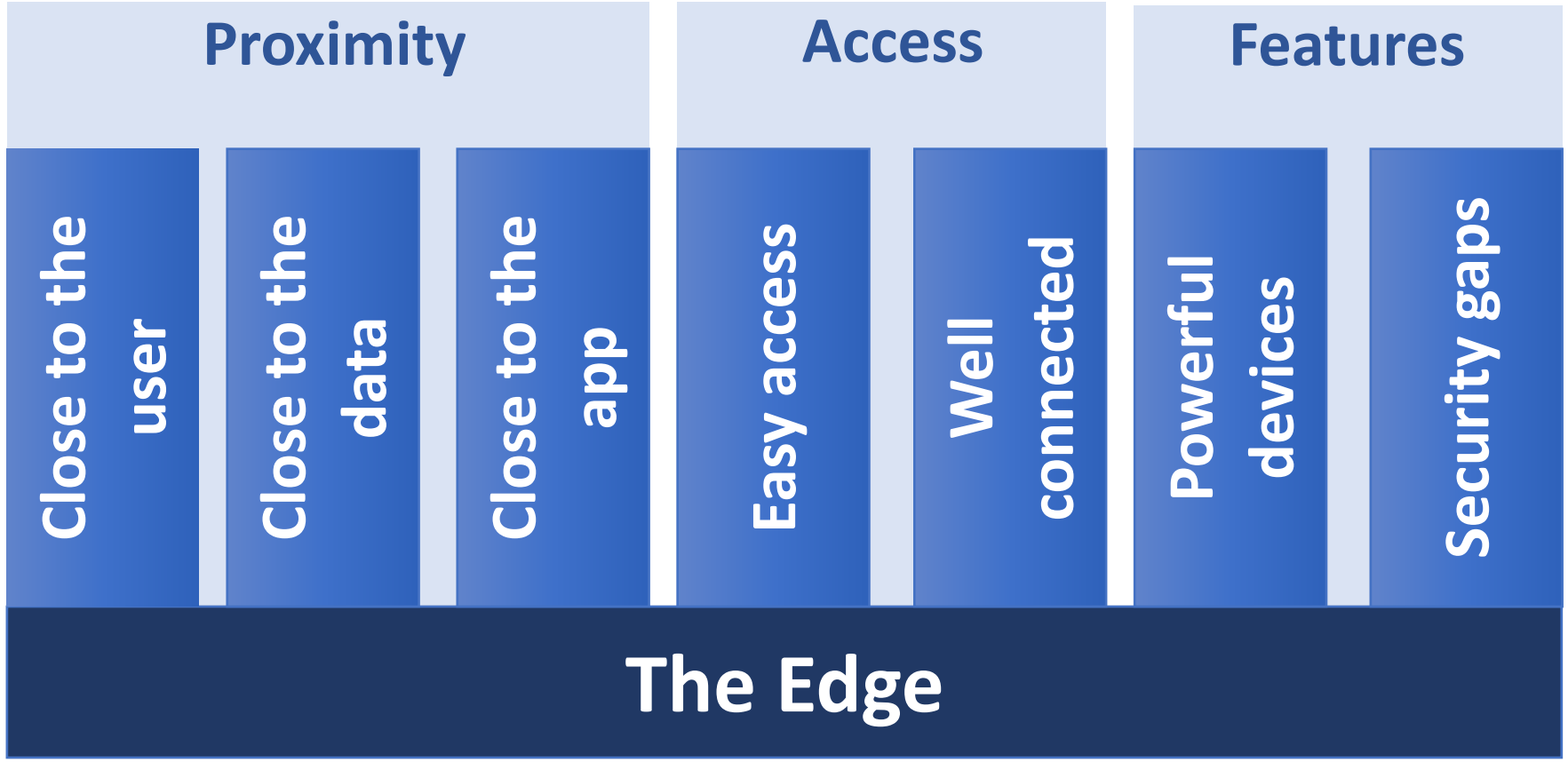
Data

Data query

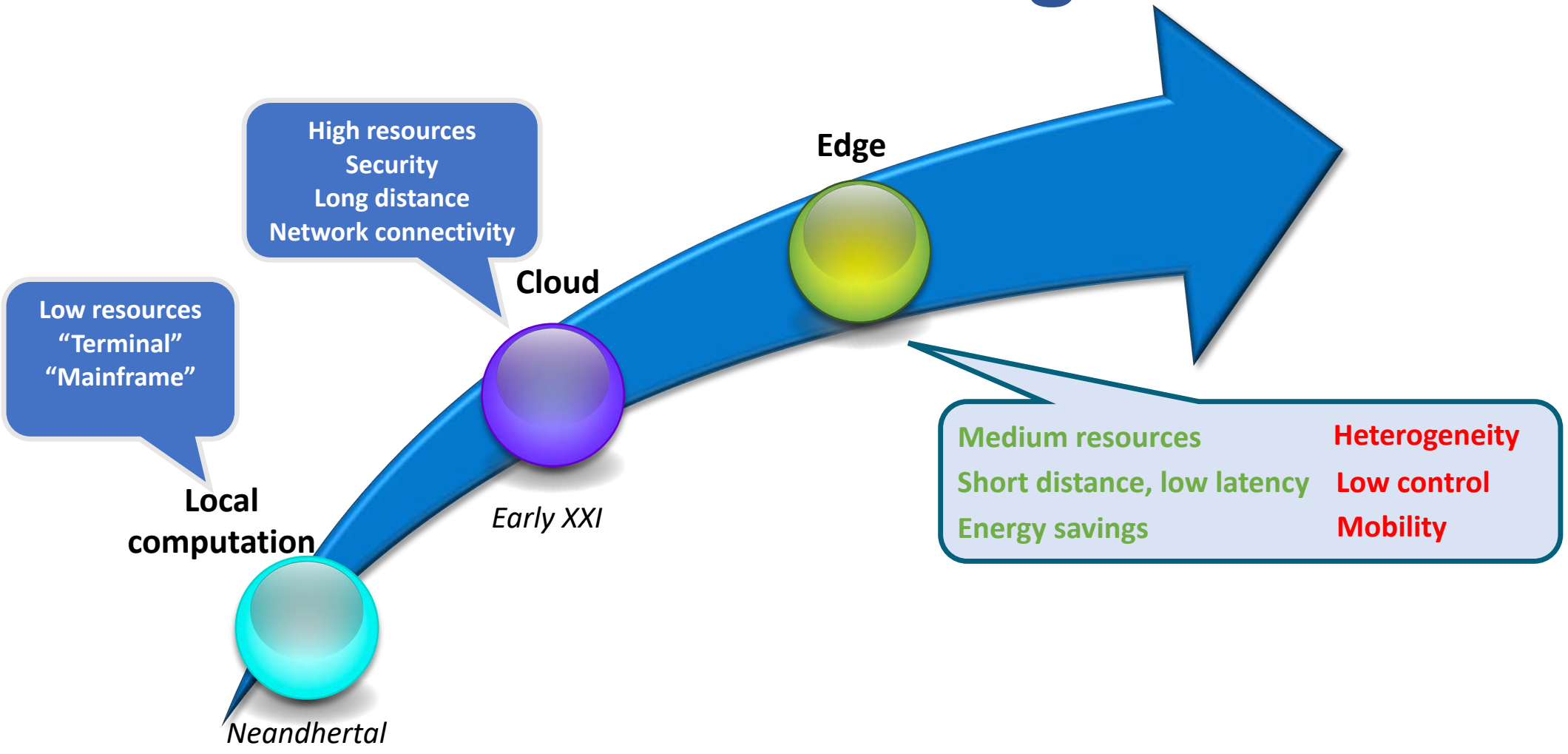
Fog

Cloud

The edge, why?

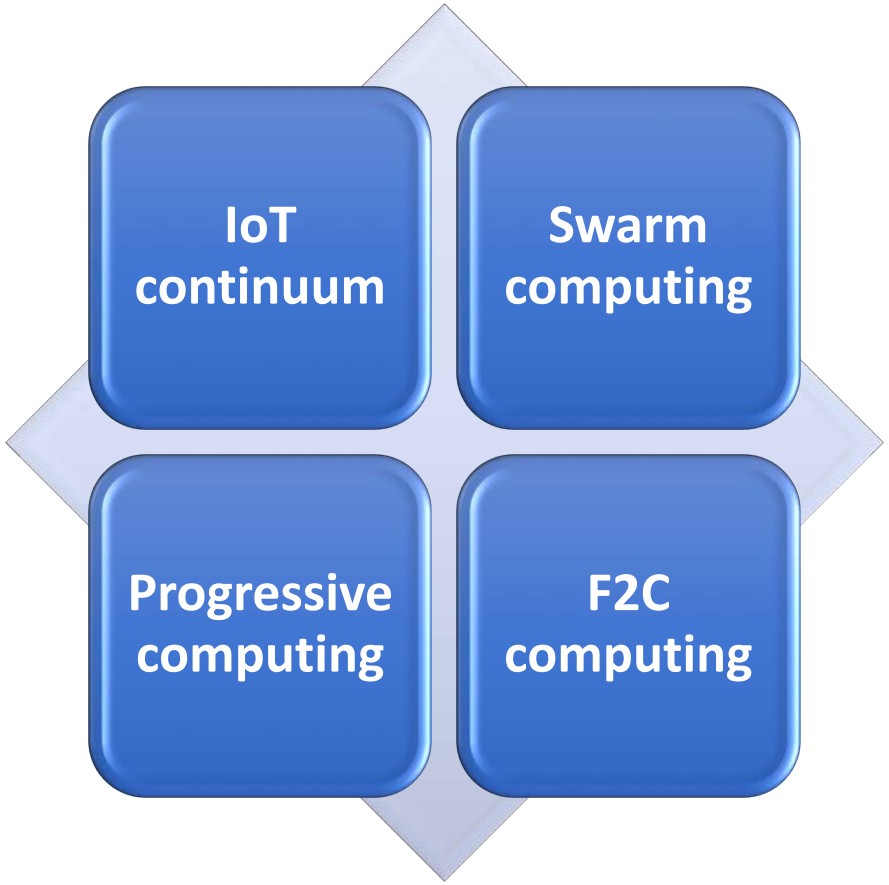

Proximity
Access
Features
Close to the user
Close to the data
Close to the app
Easy access
Well connected
Powerful devices
Security gaps
The Edge

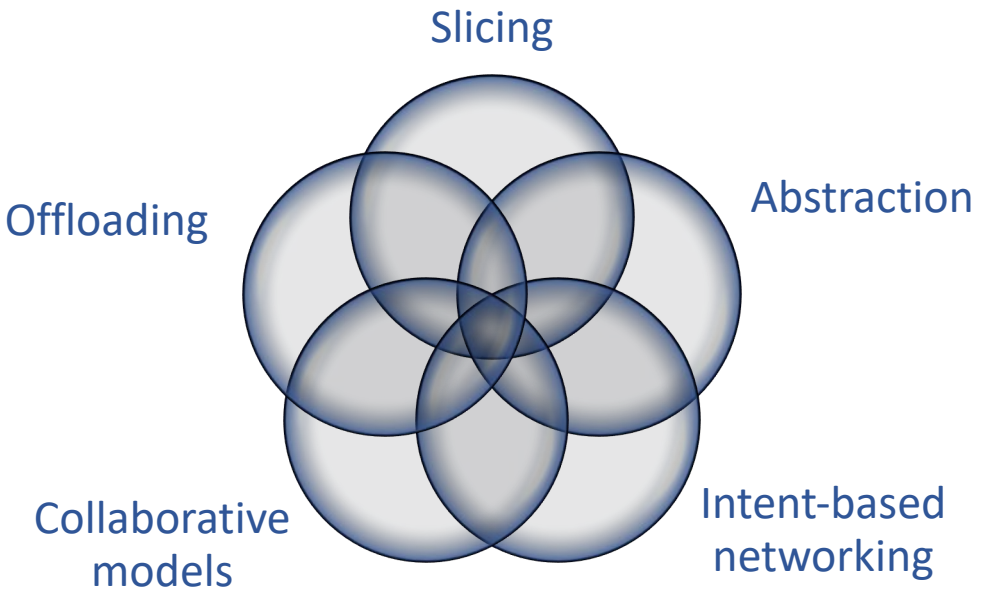
Towards the Edge



Wording

Wording


IoT continuum
Swarm computing
Progressive computing
F2C computing

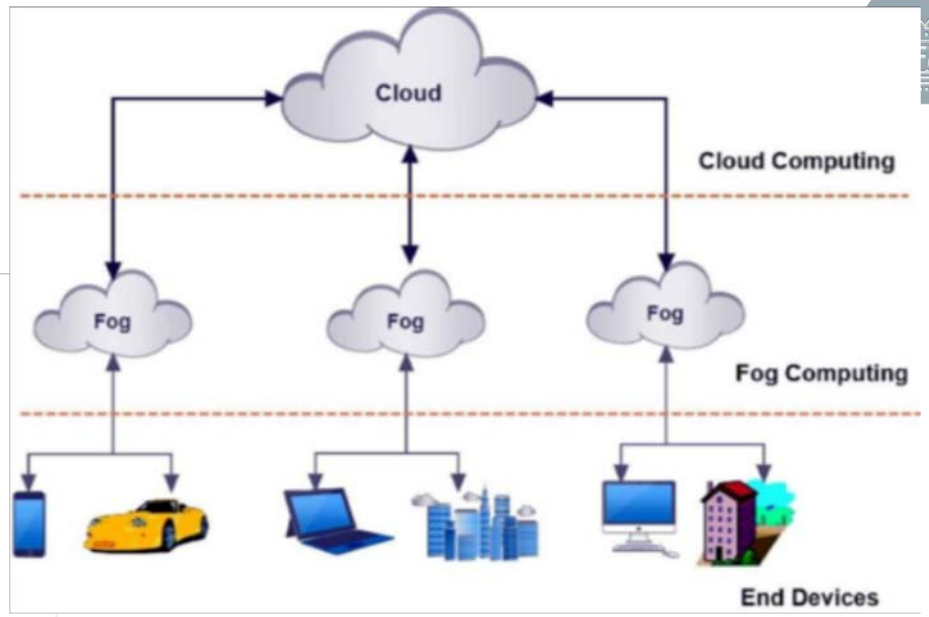

Slicing
Offloading
Abstraction
Collaborative models
Intent-based networking

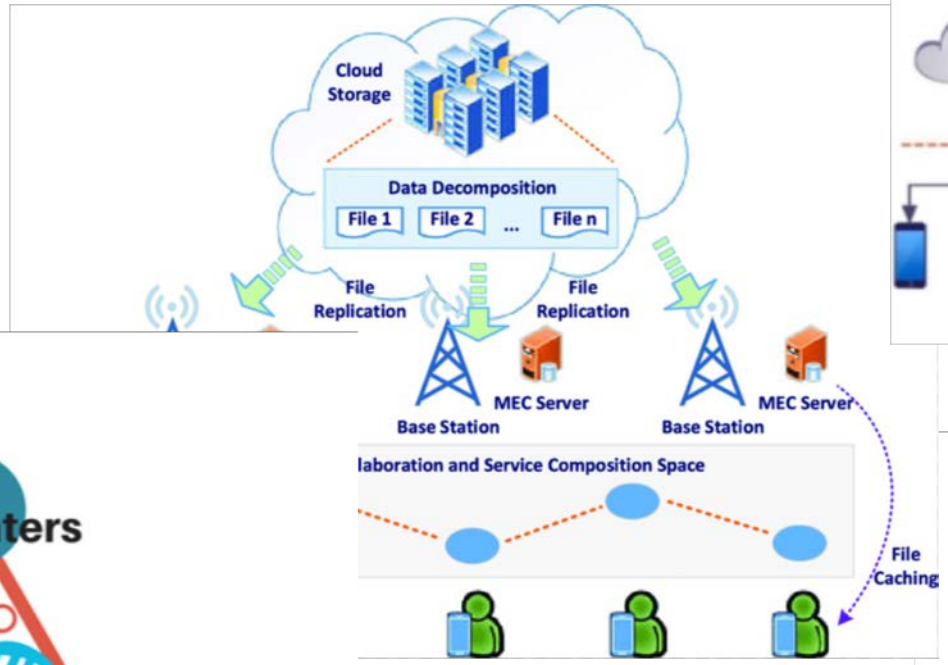
Wording

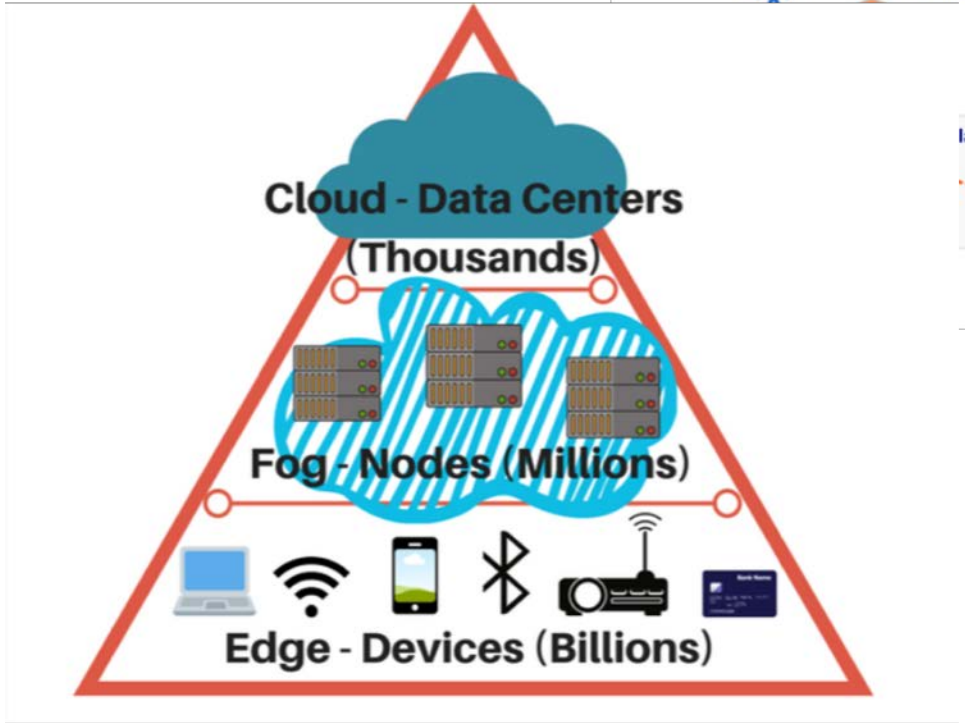
Fog vs Edge

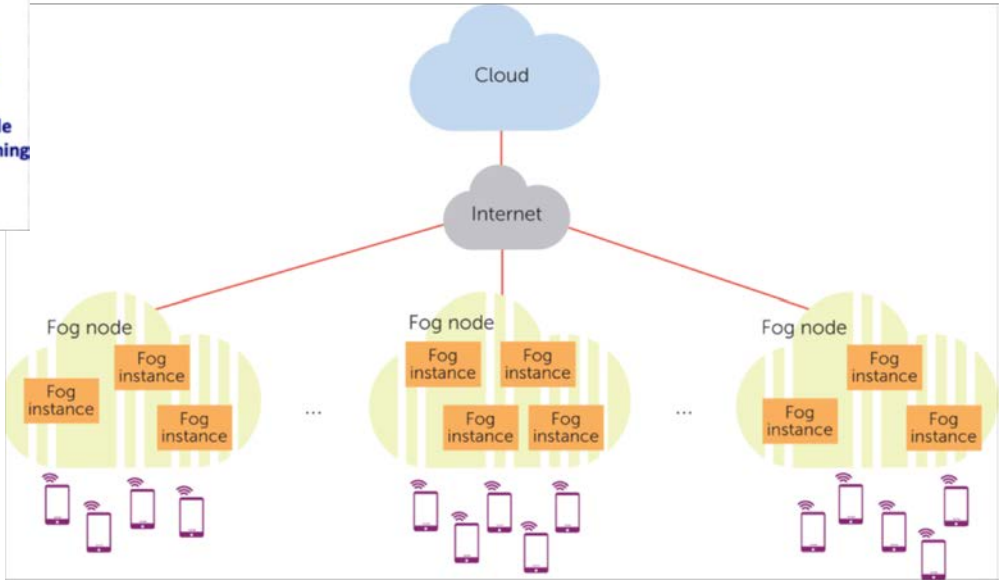
CRAAX Lab

What is fog?









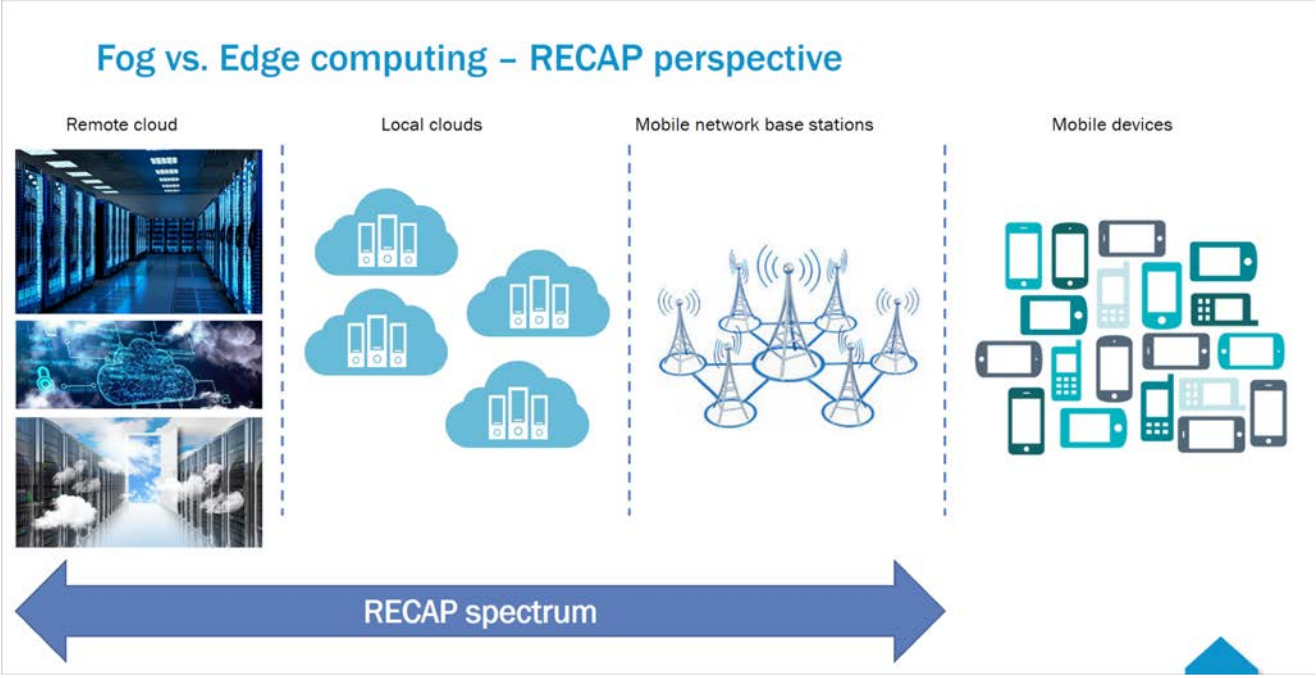
Fog computing

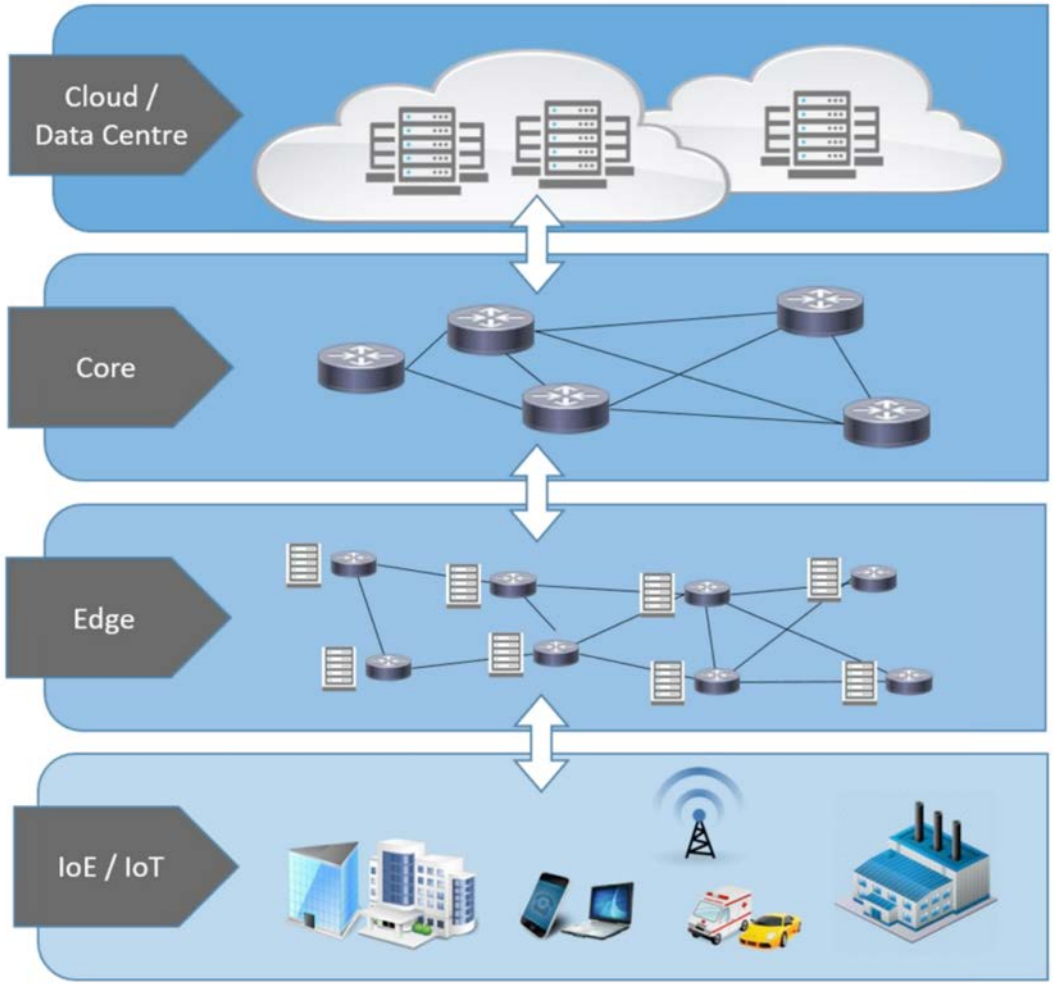
by the OpenFog Consortium

*”A horizontal, system-level architecture that **distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum**”*

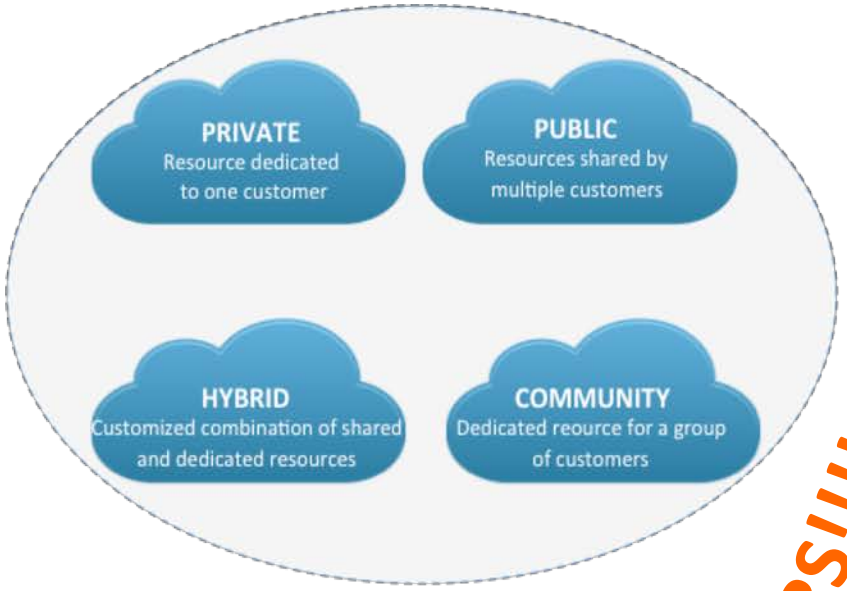
*“Fog computing also is often **erroneously called edge computing**, but there are key differences. **Fog works with the cloud**, whereas edge is defined by the exclusion of cloud. **Fog is hierarchical**, where edge tends to be limited to a small number of layers. In additional to computation, **fog also addresses networking, storage, control and acceleration**.”*

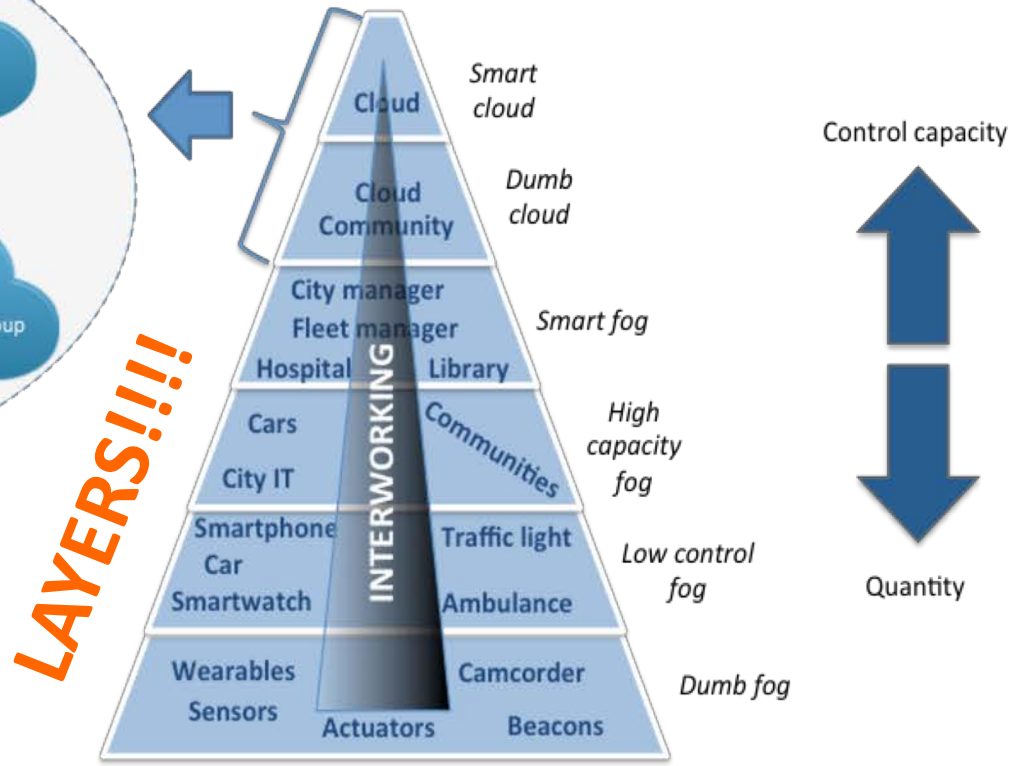
RECAP view





Recap at www.recap.project.eu





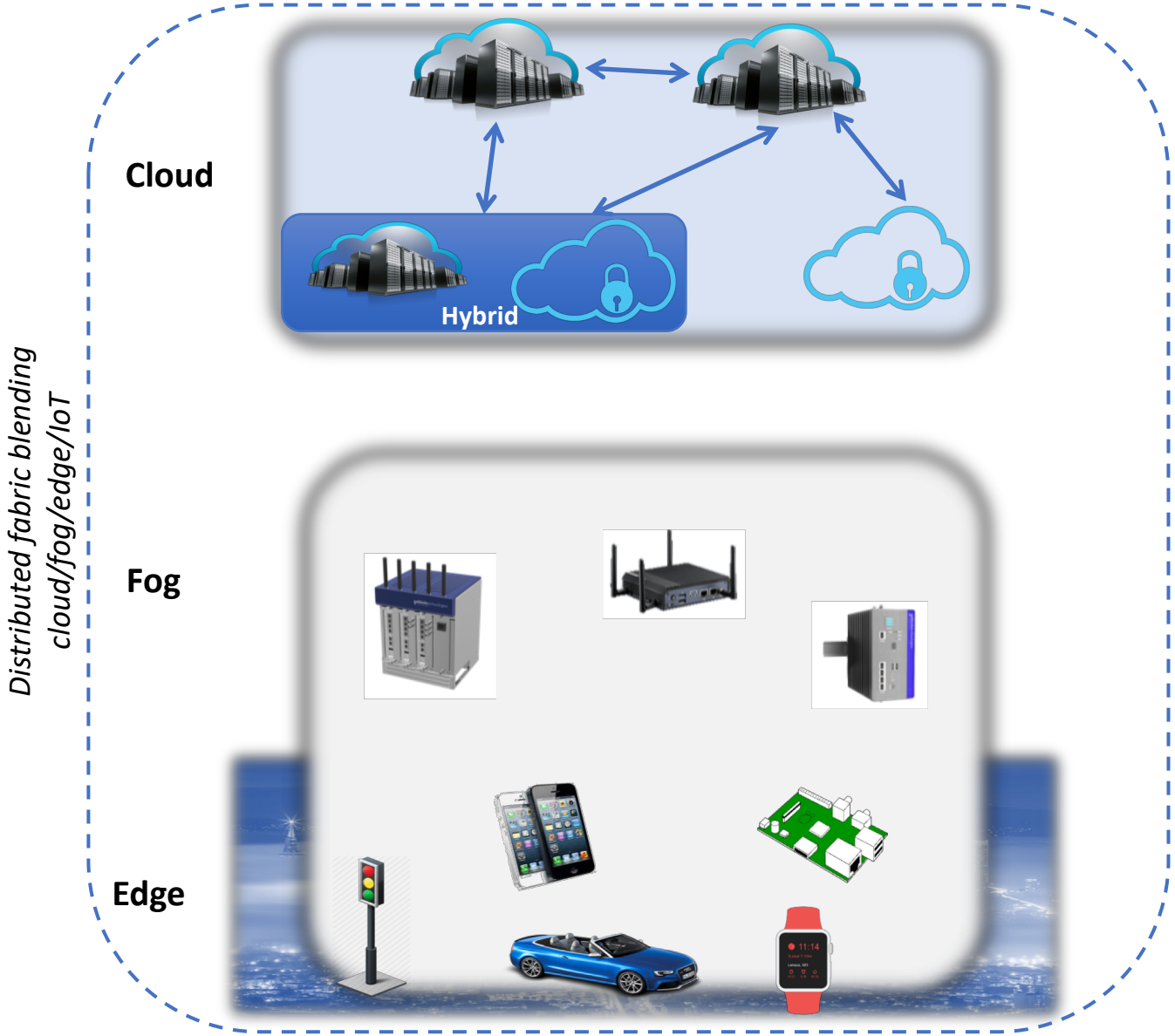
The F2C concept**

mF2C* view

* http://www.mf2c-project.eu

** X. Masip-Bruin, E. Marin-Tordera, G. Tashakor, A. Jukan, G.J. Ren, Foggy clouds and cloudy fogs: A real need for coordinated management of fog-to-cloud (F2C) computing systems, IEEE Wirel. Commun. Mag. 23 (5) (2016).

mF2C view

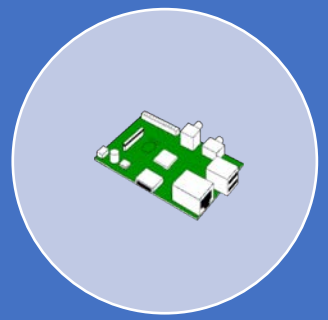

Distributed fabric blending
cloud/fog/edge/IoT
Cloud
Hybrid
Fog
Edge

		Resource continuity from edge to cloud			
		Fog			Cloud
		Edge devices	**Basic/aggregation nodes**	**Intermediate nodes**	**Cloud**
	Device	*Sensor, actuator, werables*	*Car, phone, computer*	*Smart building, cluster of devices*	*Datacenter*
Features	**Response time**	*Miliseconds*	*Subseconds, seconds*	*Seconds, minutes*	*Minutes, weeks, days*
	Application examples	*M2M communication haptics*	*Dependable services (e-health)*	*Visualizations simple analytics*	*Big data analytics statistics*
	How long IoT data is stored	*Transient*	*Minutes, hours*	*Days, weeks*	*Months, years*
	Geographic coverage	*Device*	*Connected devices*	*Area, cluster*	*Global*

So….in this context

High mobility

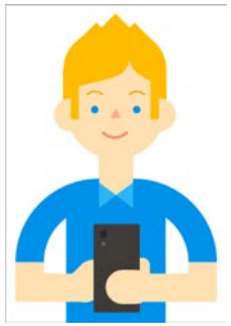
Low control



Constrained devices

Is reliability yet possible?

Reliability: how and where



Service query

Outcome

Data

Data query

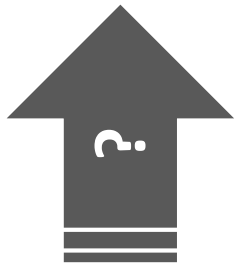
Radio Network

Two views: architectural-wise and devices-wise

Fog

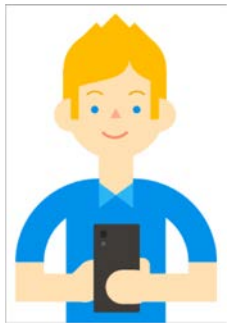
Network Core

Cloud



1. What is a fog node?

Reliability: how and where



Service query

Outcome

Data

Data query

Two views: devices-wise and architectural-wise

2. How may we manage all together?

Fog node concept

- OFC refers to a **fog node** as:

  *The physical and logical network element that implements fog computing services that allow it to interoperate with other fog nodes. It is somewhat analogous to a server in cloud computing. Fog nodes **may be physical, logical, or virtual fog nodes** and may be nested (e.g. a virtual fog node on a physical fog node).*

  And also to a **Fog Node Cluster** as:

  *Commonly referred to as **logical fog node**, this represents a group of nodes that are managed and orchestrated as a single logical entity in the fog*

- Even some industrial efforts are aligned to that trend. See, for example **Nebbiolo** Technologies proposing a fogNode to be a flexible hardware architecture defined as "***A modular computer for advanced edge computing and secure data storage with a variety of network interfaces for broad IoT connectivity***".

CRAAX Lab

HW based

nebbiolotechnologies
fog computing pioneers

PLATFORM CUSTOMERS NEWS RESOURCES COMPANY PARTNER LOGIN

The Nebbiolo fogNodes are purpose-built hardware based on an architecture that enables virtualized compute, network and storage elements. The modular design enables the manifestation of the architecture in many different form factors and with a wide range of capabilities.

Overview

fogOS is a rich software stack , enabling fast, secure, flexible communications, data management and application deployment at the fog layer. Distributed computing at the edge and device connectivity features enable a scalable solution.

Overview

fogSM implements the system manager functionality for the fogNodes and the connected devices. It offers end-to-end system management of distributed networking and computing systems, assets, software and applications. It can be deployed in the cloud or in On-prem servers.

Overview

CRAAX Lab

fogNode™ Series	
	• Fan cooled chassis • Per Slot: 4-8 core x86 i5/i7, • 128-512G Storage, • 8-16G memory, • LTE and WiFi • Secure Hardware, • Real Time capable with embedded Switch • 3 slots connected backplane for High Availability, Scale and Aux cards (e.g. GPU, Storage, Safety)
	• Fanless, 24V DC powered • 4-8 core x86 Corei5/i7, • 128-512G Storage, • 8-16G memory, • LTE and WiFi, • Secure Hardware, • Real Time capable with Embedded Switch
	• 4 core Atom, • 32-128G Storage, • 8G memory, • LTE and WiFi, • Secure Hardware, • Real Time capable with Embedded Switch
	• 2 core Atom, 32G Storage, 4-8G memory, • 3G and WiFi Gateway functions

Fog node concept

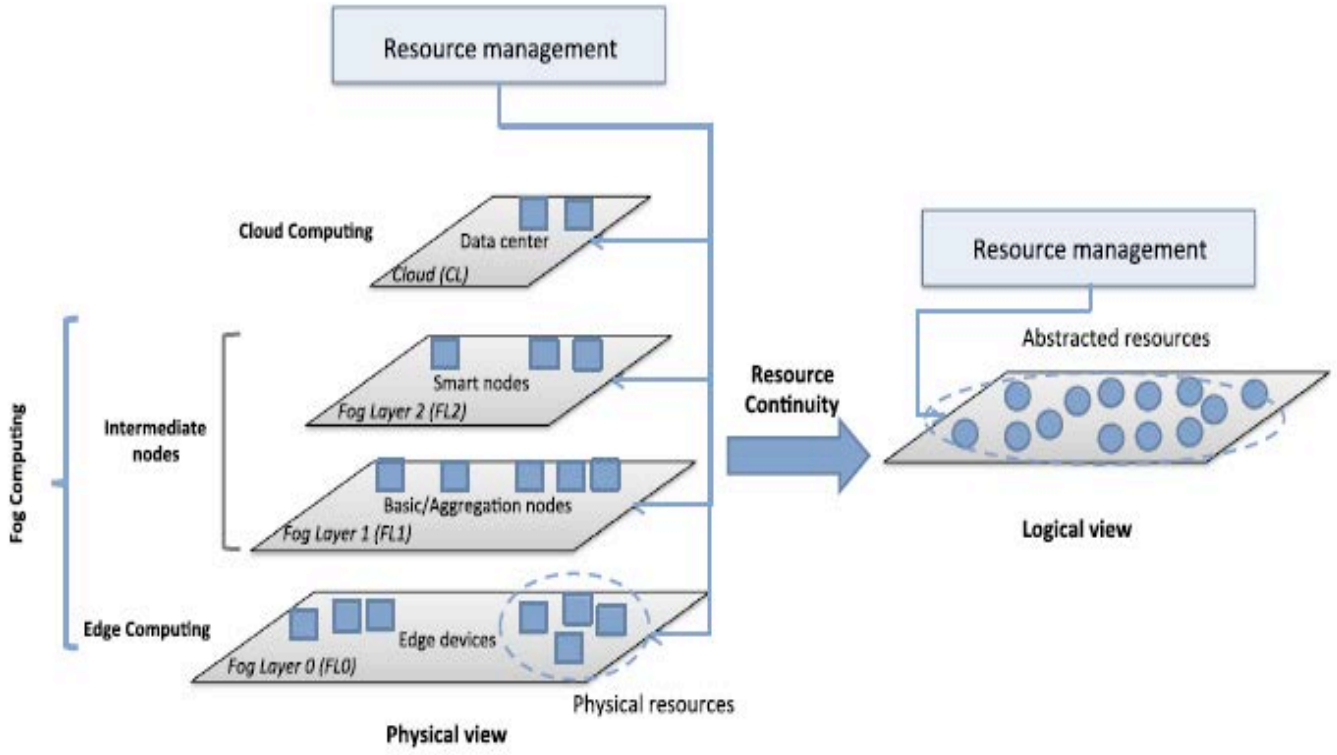
- Hardware based…….do you remember ATM FORE switches?
- Why not a different approach…say software based
  - Aligned to the OFC definition "… *The physical and **logical** network element ..:*"
- Key, key and undoubtedly key challenge: **ABSTRACTION**

Fog node: the logical concept*

Fog nodes *are distributed* ***fog computing entities*** *enabling the deployment of* ***fog services****, and formed by* ***at least one or more physical devices*** *with processing and sensing capabilities (e.g., computer, mobile phone, smart edge device, car, temperature sensors, etc.). All physical devices of a fog node are connected by different network technologies (wired and wireless) and* ***aggregated and abstracted*** *to be viewed as* ***one single logical entity****, that is the fog node, able to seamlessly execute distributed services, as it were on a single device.*

** Eva E. Marín-Tordera, X. Masip-Bruin, J. Garcia, A. Jukan, G.J. Ren, J. Zhu, "Do we all really know what a Fog Node is? Current trends towards an open definition", Computer Communications, Vol. 109, pp.117-130, September 2017*

The Abstraction Model



- Aligned to the slicing concept
- Aligned to the IoT continuum, etc.,

X.Masip-Bruin, E.Marín-Tordera, A.Jukan, G.J.Ren, "Managing Resources Continuity from the Edge to the Cloud: Architecture and Performance", Future Generation Computer Systems, Vol. 37, February 2018

But….

Abstraction means to say virtualization?

Virtualizing the “edge”????

Are you for real!!!!

Challenges virtualizing the edge

Characteristics inherent to fog computing impacting on virtualization:

- Mobility
- Resource scarcity
- Lack of control
- Heterogeneity
- Energy management
- System lock-in
- Security

What do we want to virtualize??

Virtualizing the edge

- Computer edge devices
  - VMs, Containers
- Sensor and actuator edge devices (IoT)
  - Edge virtualization extended to sensors and actuators.
  - Options: hypervisor responsible for abstraction, using semantic web technologies to get a virtual IoT resources description
- Network
  - Many different network technologies –e.g. 3G/4G/5G, LTE, Ethernet, WiFi, Bluetooth, LoRA, etc and etc….(*actually no idea on what is next*)
  - Option: manage clusters of edge devices through an SDN-like controller handling the programmability of the network

Challenges virtualizing the edge

Characteristics inherent to fog computing impacting on virtualization:

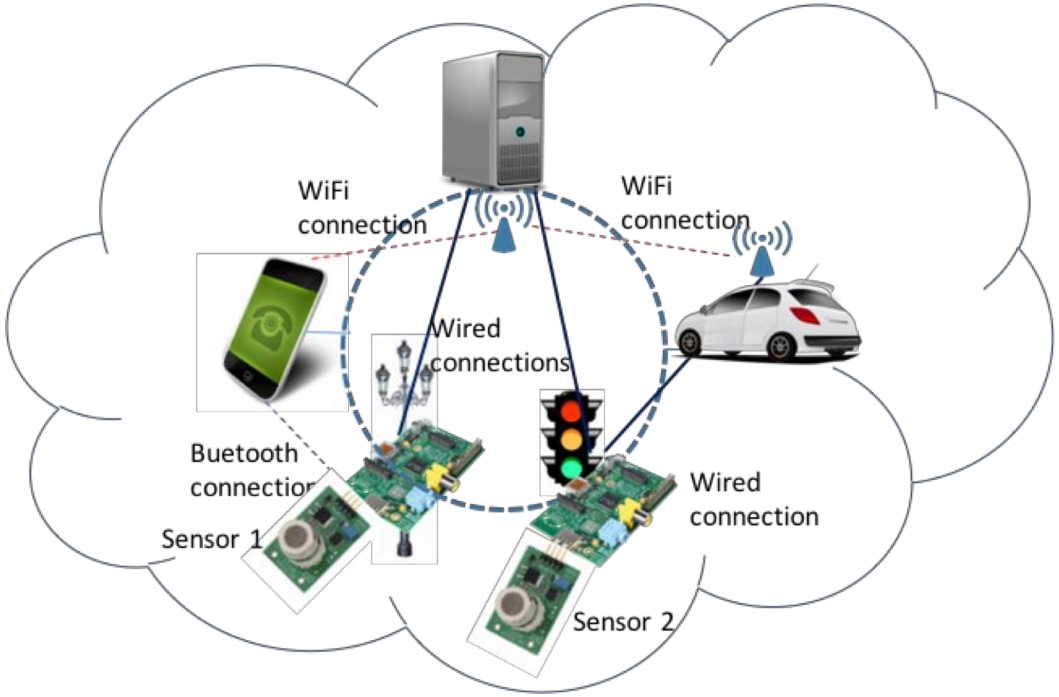
- Mobility
- Resource scarcity
- Lack of control
- Heterogeneity
- Energy management
- System lock-in
- Security

What do we want to virtualize??

Everything we can!!!!

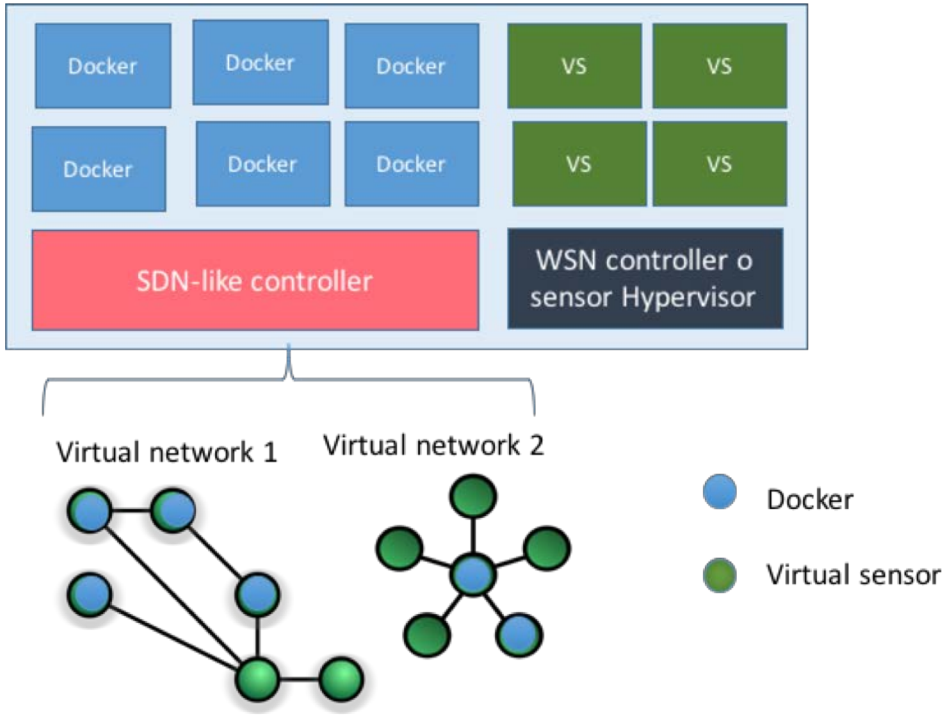
Illustrative Example

Physical devices and topology



Devices with computing, storage, memory, network and sensing (including sensors in the car) capacities.

Two possible virtualizations for two different services



The two different services have a different and isolated view of the resources.

Architectural view

New scenario

Coordinated combination of Edge/Fog and Cloud

Friends will be friends…

Source: http://www.forbes.com/sites/theopriestley/2015/10/08/salesforce-and-amazon-take-iot-fight-to-the-cloud/#35697a5347fb

http://www.indianweb2.com/2014/12/08/new-fog-computing-smart-cloud-computing/

The Context

Cloud computing

"Unlimited" capacity

… as well as
- Cost efficiency
- Elasticity
- Ubiquity

But too far!

Limited capacity

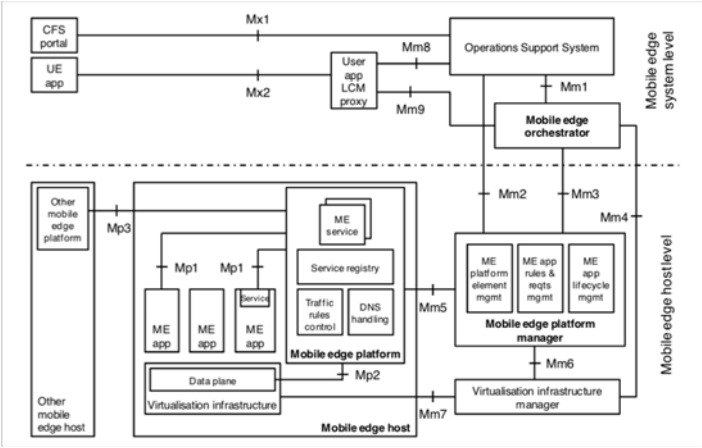
Edge Computing

… but, **advantages of locality**
- Reduces network traffic
- Decreases latency
- Saves energy

Architectural view: Efforts

MEC (Multi-access Edge Computing)

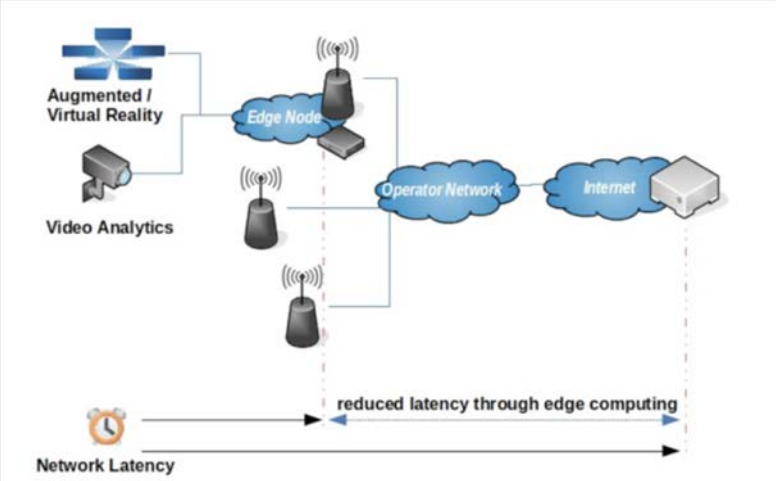
- Industry Specification Group (ISG) initiative within ETSI,
- MEC's main focus falls into the networking field at the edge,



*Mobile edge system reference architecture**

OEC (Open Edge Computing)

- Industrial initiative, focusing on edge computing issues, especially motivated to drive new business opportunities and technologies around the edge computing concept.



*OEC edge computing architecture***

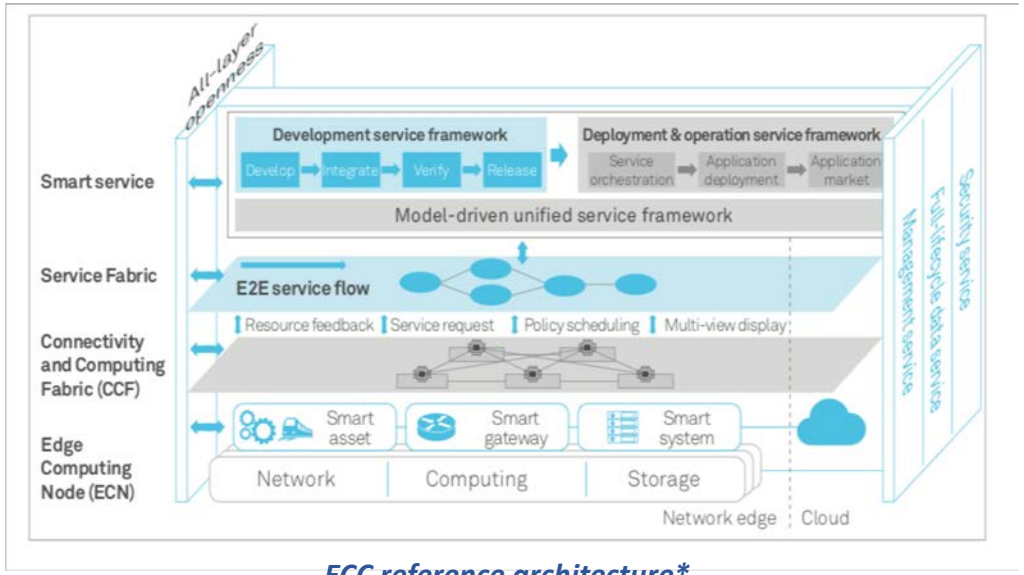
* MEC: Framework and Reference Architecture at http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf)

** Open Edge Computing at http://openedgecomputing.org/about.html

Architectural view: Efforts

ECC (Edge Computing Consortium)

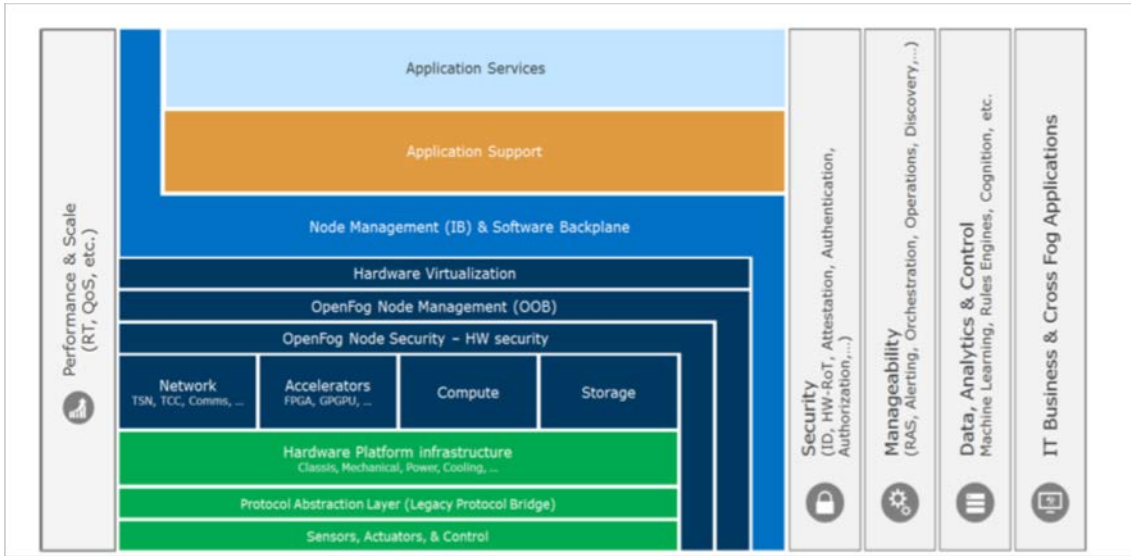
- Industrial consortium, white paper with the Edge Computing Reference Architecture 2.0, assessing the need for edge and fog computing to collaborate



*ECC reference architecture**

OFC (The OpenFog Consortium)

- Industrial and academic, extending the scope across multiple protocol layers, not only radio systems but spanning across the edge to the upper cloud



*OFC's architecture description***

* Edge Computing Reference Architecture at http://en.ecconsortium.org/Uploads/file/20180328/1522232376480704.pdf

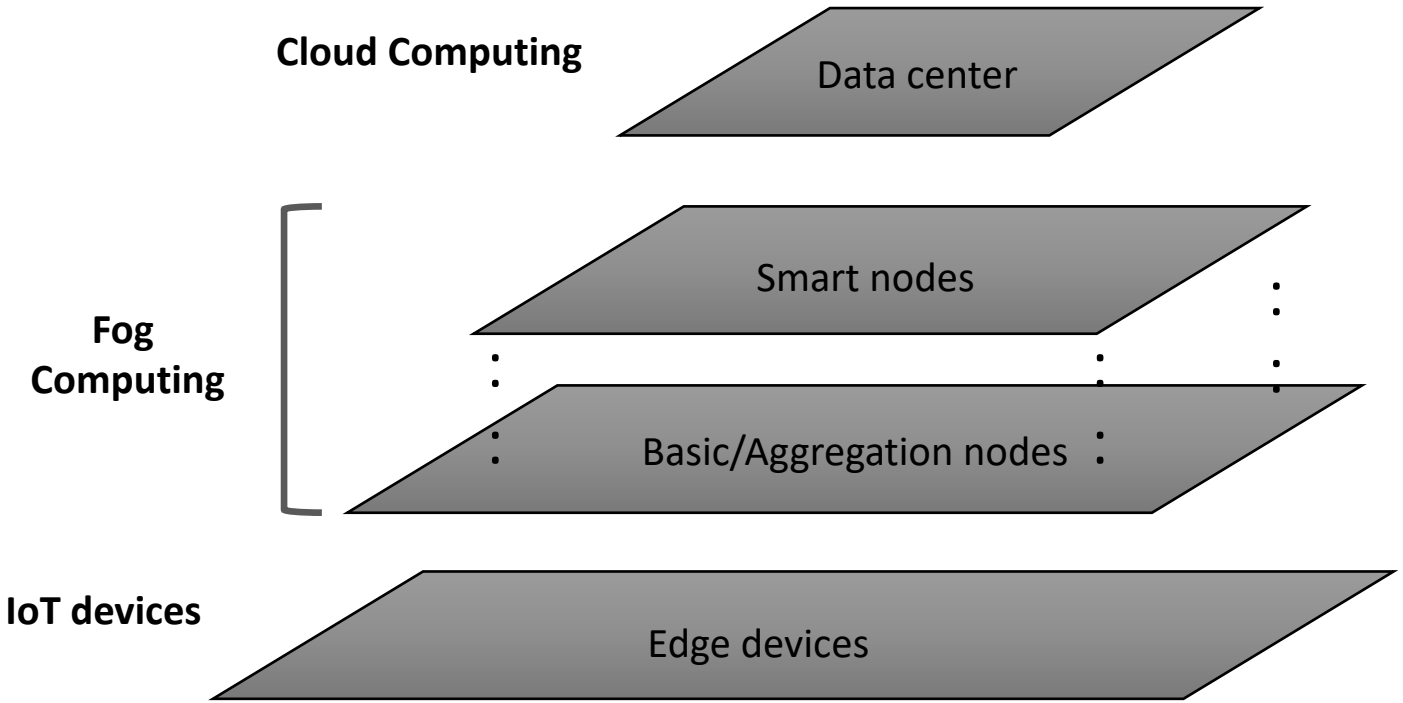
*** Open fog consortium working group: OpenFog Reference Architecture for Fog Computing White paper, Feb 2017*

Architectural view: mF2C

- EU H2020 research project
- Lasting 3 years till the end of 2019.
- Main aim is to design and develop the F2C concept.
- In practice, to design a layered, open, secure and hierarchical architecture to control the large set of distributed heterogeneous devices distributed from the edge up to the cloud.

http://www.mf2c-project.eu

Layered Architecture



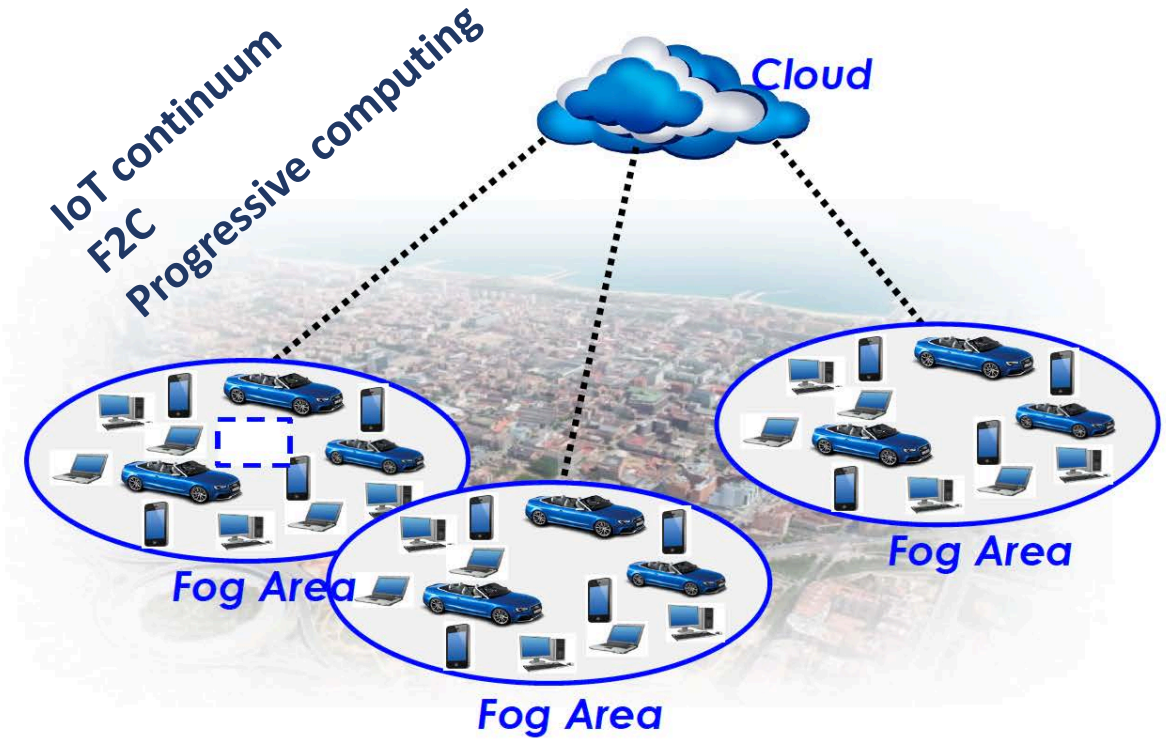
Advantages:

- IoT applications with Low-latency requirements are executed in fog.
- Applications with high performance computing (HPC) or Big Data requirements are executed in cloud.

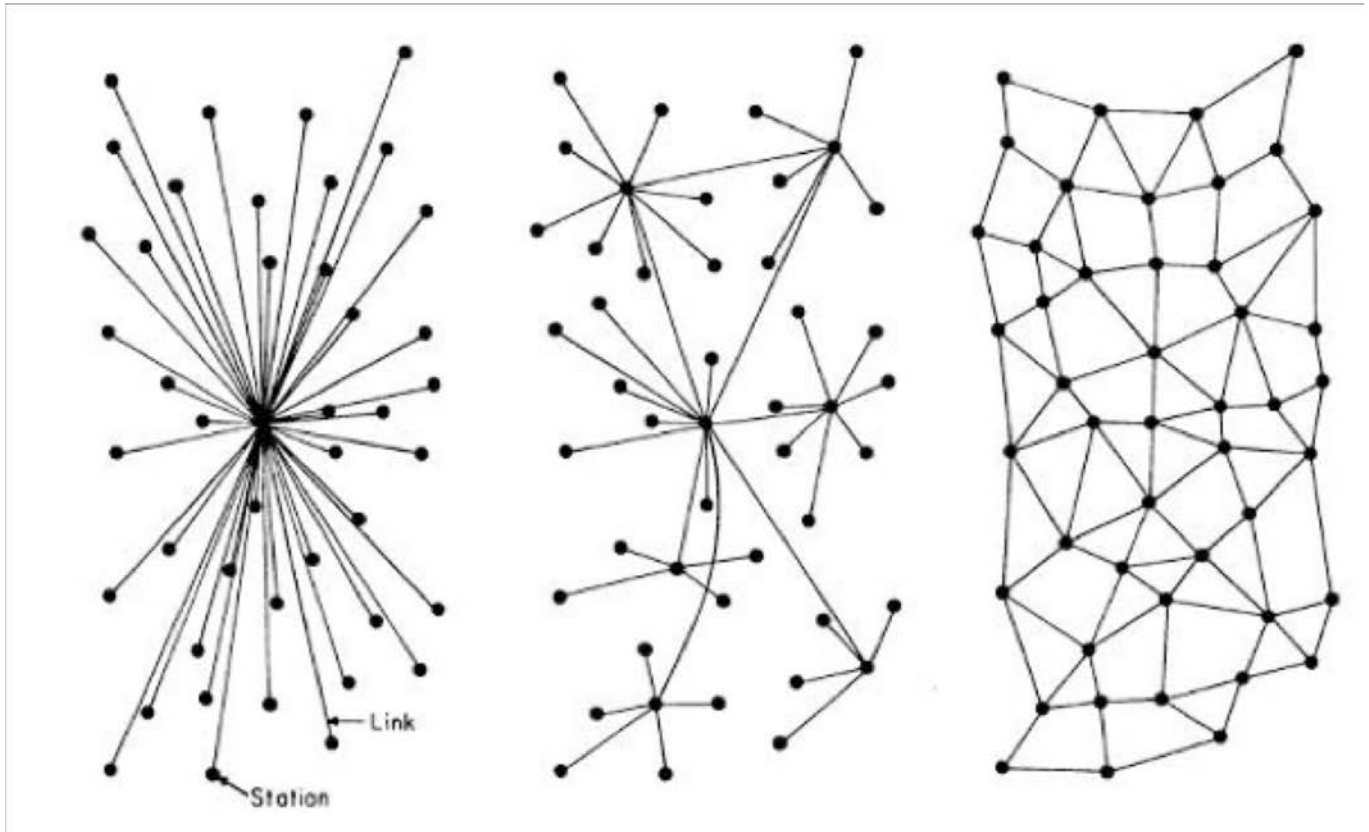
Drawbacks:

- Need of resource/service management between cloud and fog.
- Need of new programming models.
- New virtualization strategies.

Context


IoT continuum
F2C
Progressive computing
Cloud
Fog Area
Fog Area
Fog Area

Architectural view



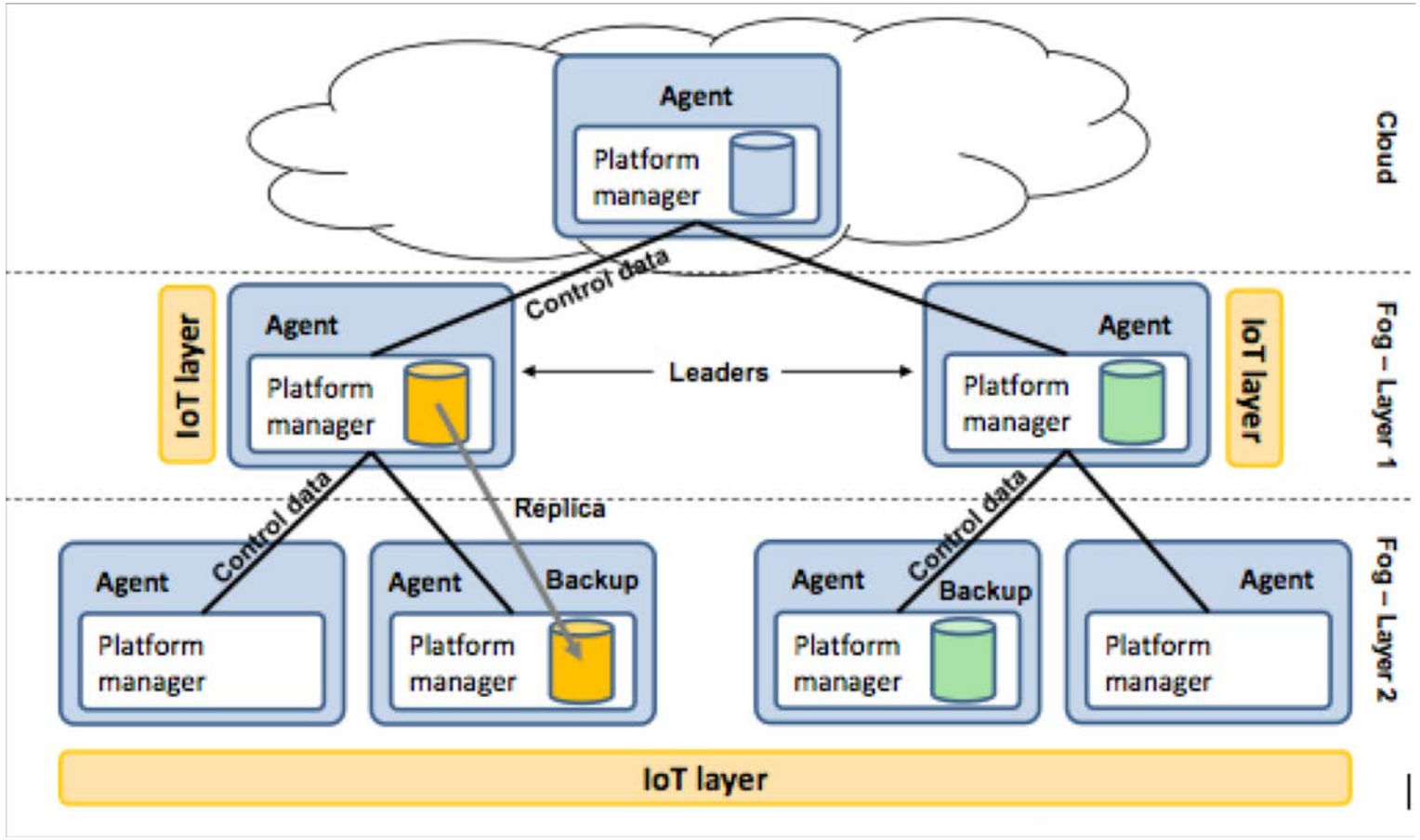
Centralized *Decentralized* *Distributed*

Architectural view

Characteristic	Centralized	Decentralized	Distributed
Reliability	L	M	H
Maintenance	H	M	L
Stability	L	M	H
Scalability	L	M	H
Settings	H	M	L
Evolution	H	M	L

Control architectures characteristics (L: low; M: moderate; H: high)

Hierarchical Architecture



Reliability at the edge

- At the edge 
- Many open problems
- Let’s put the focus on the **F2C architecture**

Reliability

- Dynamicity demands novel failure recovery mechanisms,
- Fog scenarios demands low service allocation time and protection cost
- Learning from the past ("old" examples):
  - data centers [*I. F. Akyildiz, A. Lee, P. Wang, M. Luo, W. Chou. Research challenges for traffic engineering in software defined networks. In: IEEE Network, 30(3), 52-58, 2016*]
  - wireless sensor networks [*M. Younis, I. F. Senturk, K. Akkaya, S. Lee, F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: A survey", Computer Networks, 58-15, 254-283, 2014*].
  - Backup paths [*A. Sgambelluri, A. Giorgetti, F. Cugini, F. Paolucci, P. Castoldi. OpenFlow-based segment protection in Ethernet networks. In: Journal of Optical Communications and Networking, 5(9), 1066-1075, 2013*]
  - service execution replication in fog [*Y. W. Kwon, E. Tilevich. Energy-efficient and fault-tolerant distributed mobile execution. In: IEEE 32nd International Conference on Distributed Computing Systems (ICDCS), (pp. 586-595), 2012*]
  - Cloud resilience through network virtualization [*I. B. B. Harter, D. A. Schupke, M. Hoffmann and G. Carle, "Network virtualization for disaster resilience of cloud services". In: IEEE Communications Magazine, vol. 52, no. 12, pp. 88-95, December 2014*]
  - shared-path shared-computing (SPSC) protection for cloud [*C. Natalino et al., "Dimensioning optical clouds with shared-path shared-computing (SPSC) protection". In: 2015 IEEE 16th International Conference on High Performance Switching and Routing (HPSR), Budapest, 2015, pp. 1-6*]
  - MEC failures [*D. Satria, D. Park, M. Jo, "Recovery for Overloaded Mobile Edge Computing", In: Future Generation Computer Systems, 2016*]
- **What to do in F2C scenarios??**

Reliability in F2C: key challenges

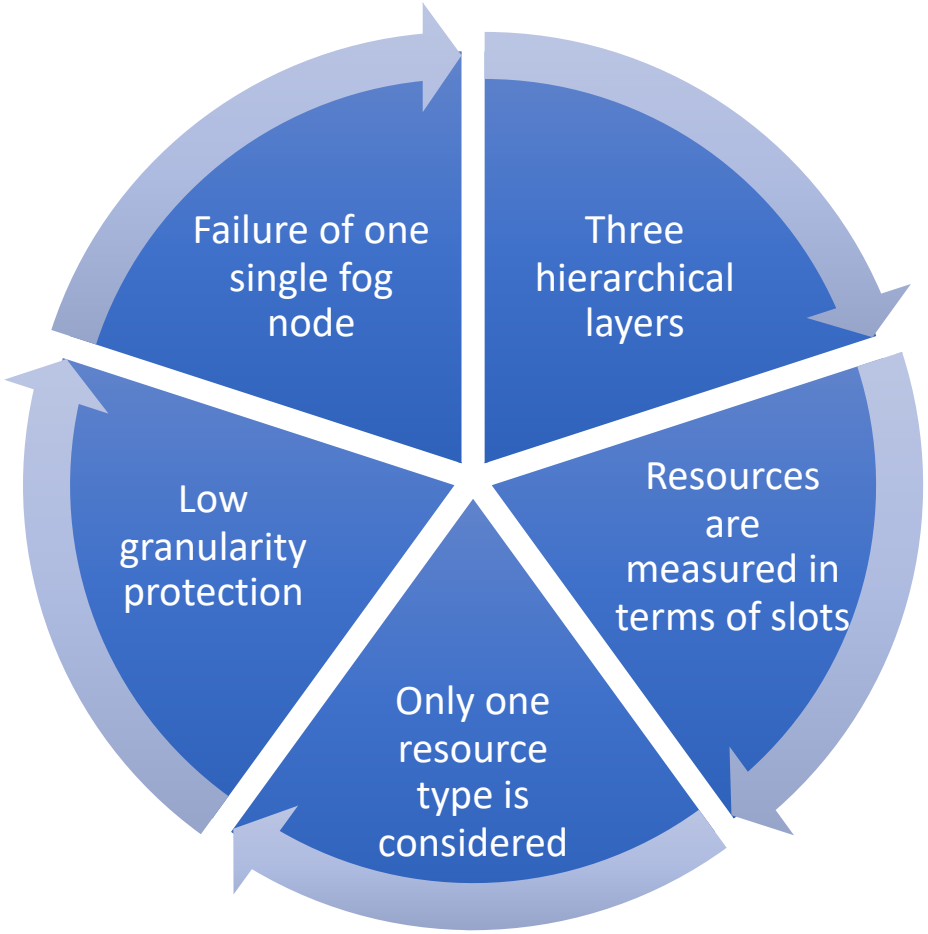
- Failure recovery
- Discovery
- Clustering
- Categorization
- SLA+QoS enforcement

1. F2C: Failure recovery

Need for failure recovery mechanisms	
Edge devices: High dynamic	Distributed scenarios

Protection strategies	
Proactive	Reactive

mF2C: Assumptions



Context & Proposal

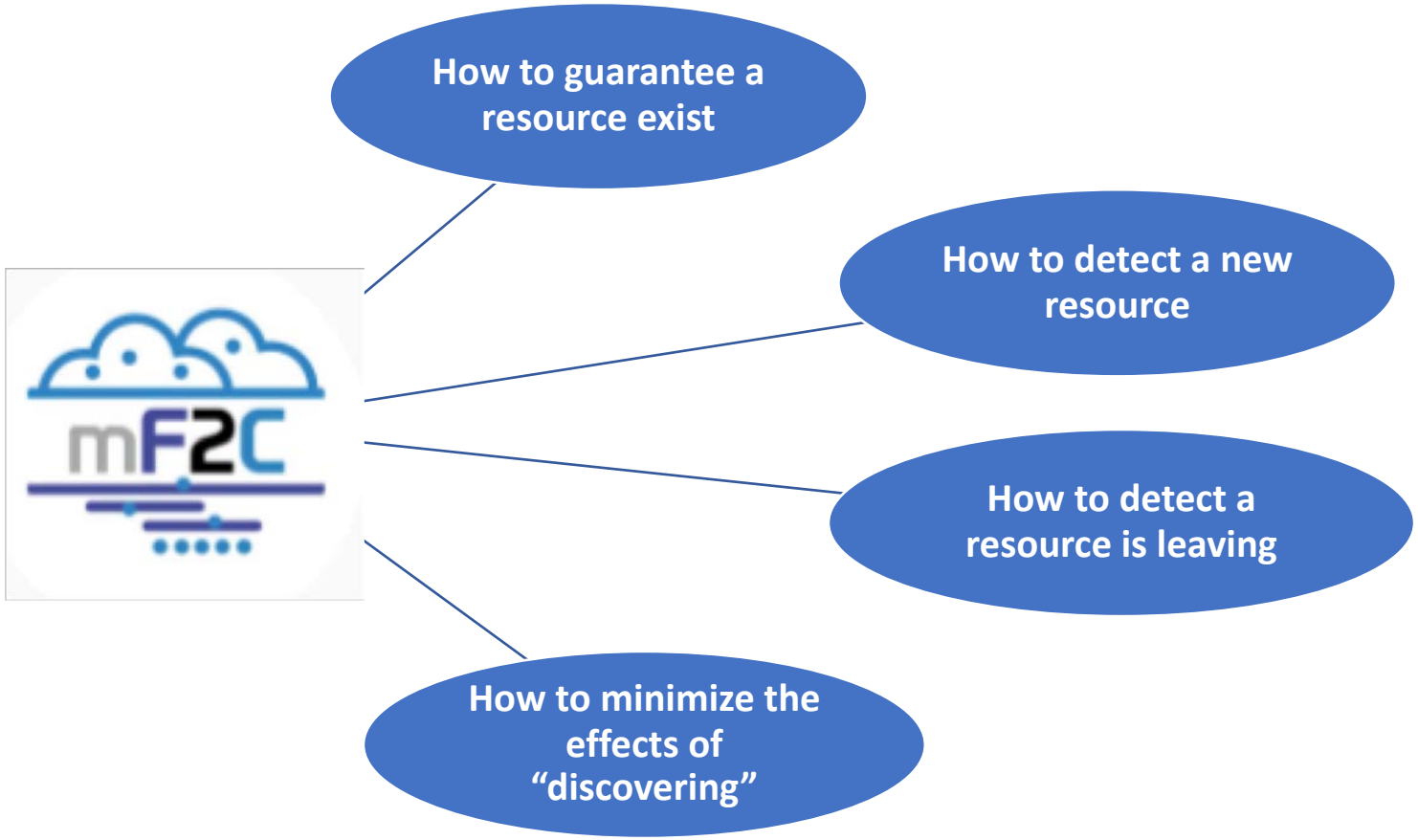
- Proactive recovery
  - 1 to 1 pre-allocation
  - No added allocation delay
- Reactive recovery
  - Resource reserved for protection
  - No pre-allocation
  - Diminish resource underutilization
- Modeling the failure recovery problem as a Multidimensional Knapsack Problem (MKP)*
- Two objectives
  - decrease the delay for transmission of each service
  - decrease the protection cost –by reducing the slots consumed for protection
  - diminish the recovery latency –by staying at the edge

*V.Barbosa, X.Masip-Bruin, E.Marín-Tordera, W.Ramírez, S-Sánchez-López, "Proactive vs Reactive Failure Recovery Assessment in Combined Fog-to-Cloud (F2C) Systems", IEEE 22nd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Sweden, June 2017

Outcome

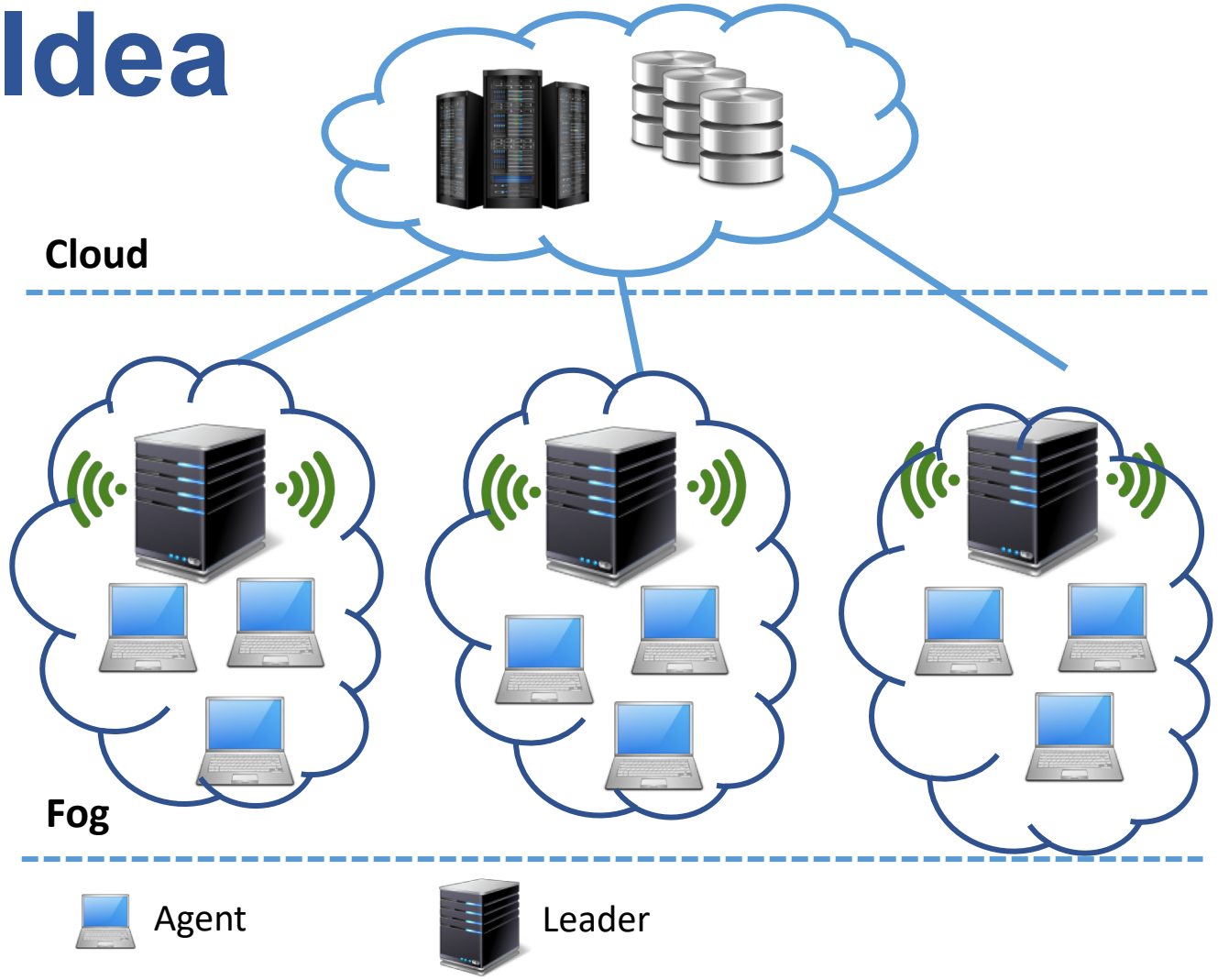
- Protection strategies show a considerable impact on the recovery performance
- F2C architecture may employ both proactive and reactive protection strategies
  - Reactive: several service requests, such as Smart Transportations
  - Proactive: sensitive services that may benefit from redundancy, such as e-Health

2. F2C: Discovery



Idea

- Assumption:
  - Consider the resources to be discovered to be part of a same controlled domain
- Proposal:
  - Leaders broadcast custom WiFi beacons to make devices in their vicinity aware of their presence, with no preassociation required



*Z. Rejiba, X. Masip-Bruin, A. Jurnet, E. Marin-Tordera, and G.-J. Ren, "F2C-Aware: Enabling Discovery in Wi-Fi-Powered Fog-to-Cloud (F2C) Systems," in 2018 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2018, pp. 113–116.

But (I)…

Legend:

Broadcast

Scan

Fog Area Leader

However,

- Mobility scenario
- Unnecessary scans in areas with no fog coverage

➔ Energy consumption penalties

- Disabling the scan

➔ Discovery opportunities may be missed

Optimize this scan process

Proposal

- **BDSS: Beacon-assisted Direction-aware Scan Scheme**
- Assistance information: **(1) Remaining distance** until **next leader** will be **reached**, in each of the 4 cardinal directions **(2) Channel** in use for broadcasting beacons
- **Ongoing work**

But (II)

However,

- **Mobility** scenario

- **Unnecessary** scans in areas with **no** fog **coverage**

➔ Energy consumption penalties

- Disabling the scan

➔ Discovery **opportunities** may be **missed**

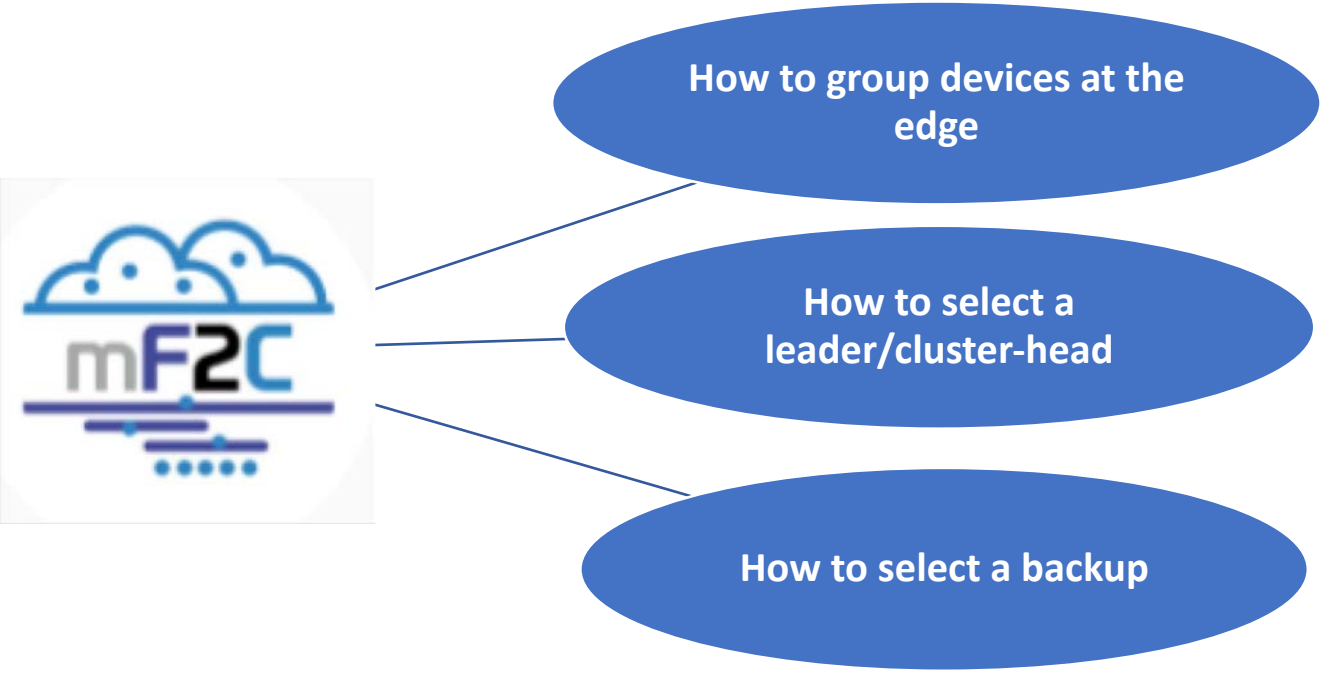
Optimize this scan process

ON going work

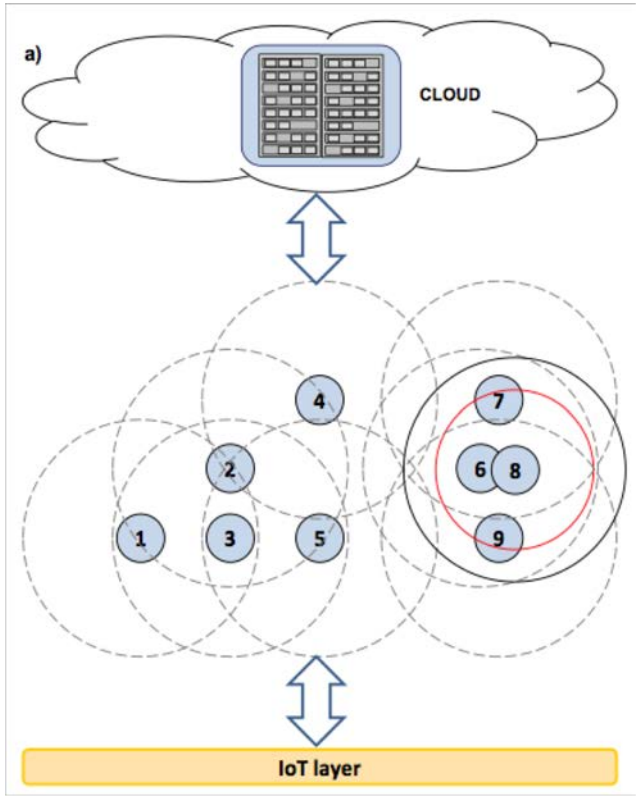
Main idea: If we know contexts associated with FN locations

➔ Scan only when those contexts are observed!

3. F2C: Clustering



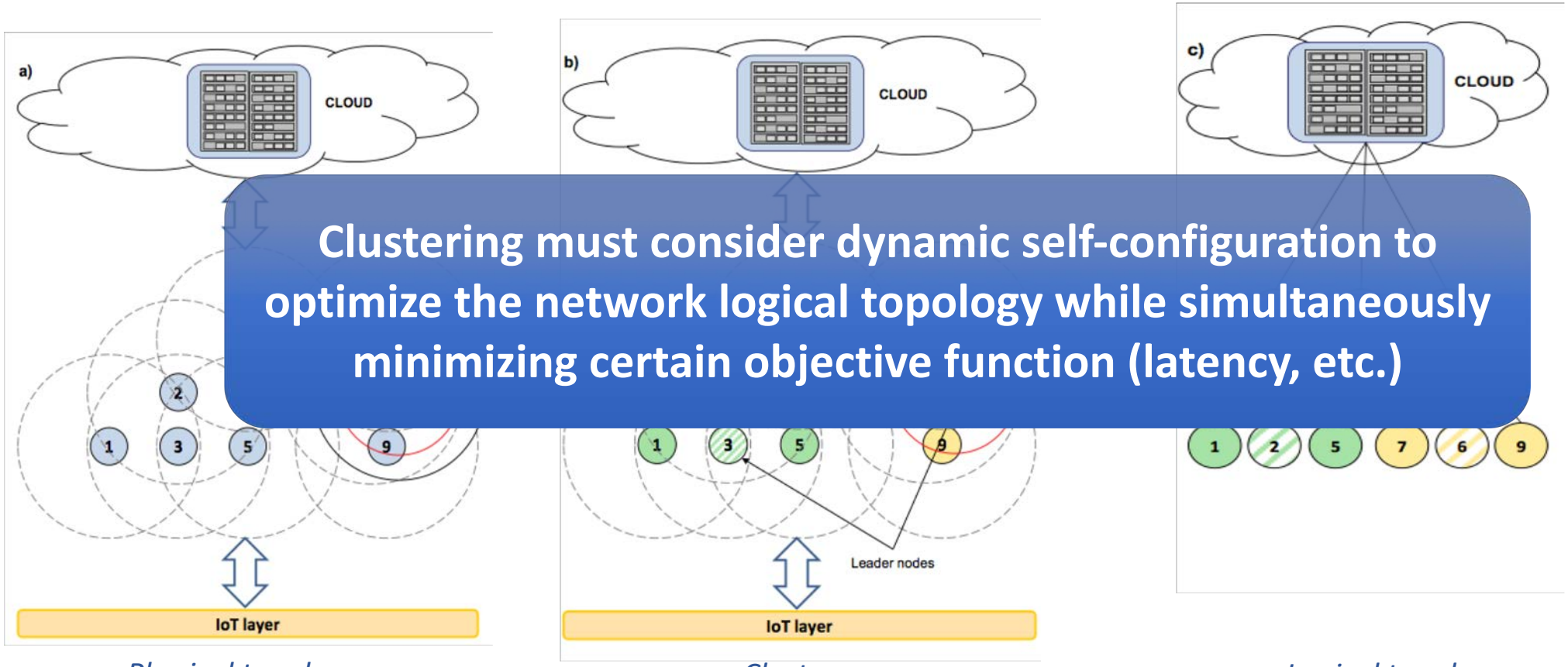
3. F2C: Clustering objectives



- ensuring access to management functions from fog to cloud
- guaranteeing the desired processing capacity at the edge (e.g., to reduce amount of data sent to cloud or to reduce latency between edge devices and control functions)
- reducing transmission power to both save energy consumption and try to avoid interference
- trying to minimize rapid changes in the edge devices providing management functions.

AND RESILIENCE

F2C: Clustering



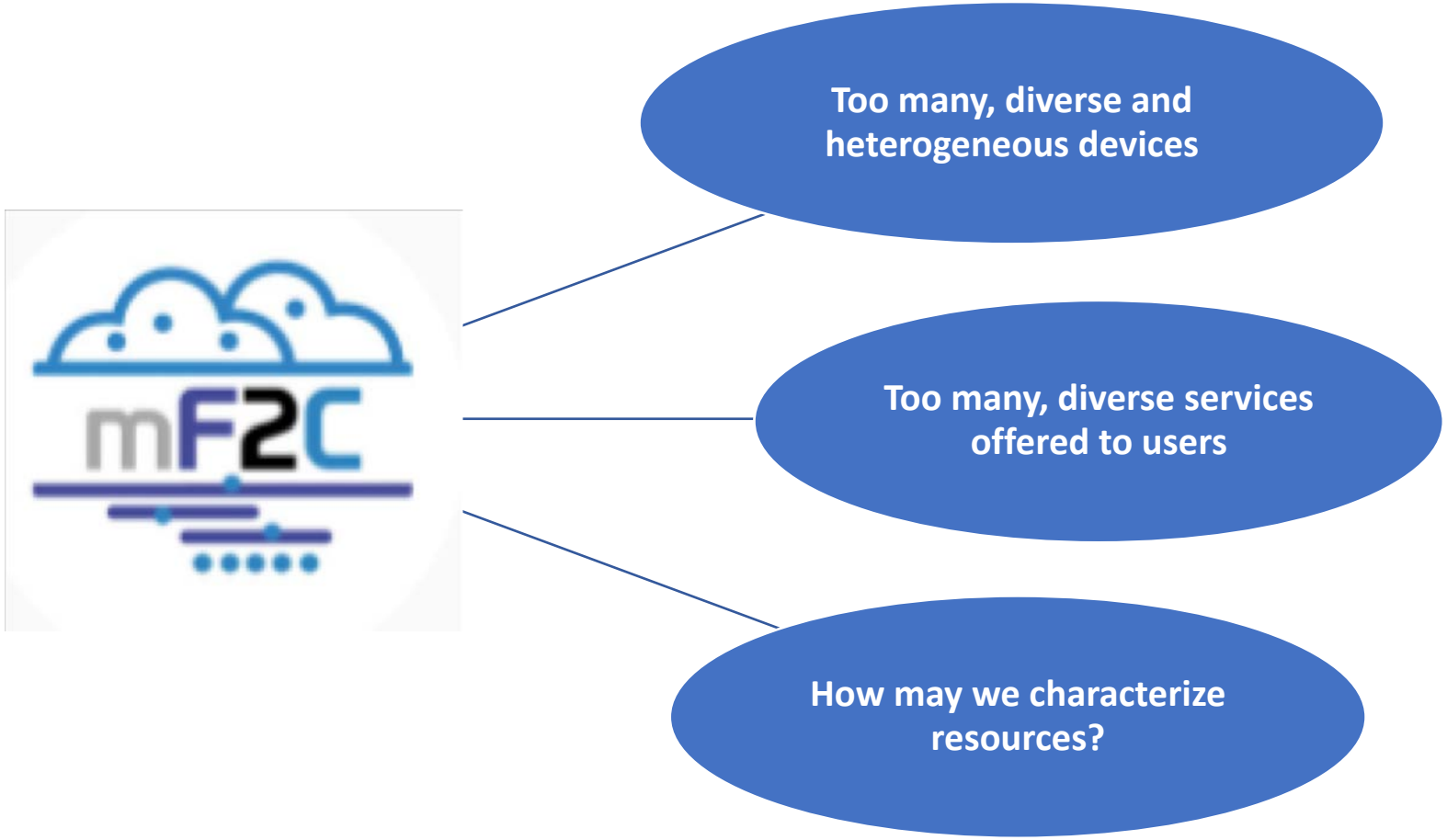
Physical topology *Clusters* *Logical topology*

Clustering must consider dynamic self-configuration to optimize the network logical topology while simultaneously minimizing certain objective function (latency, etc.)

Yet cooking

- MILP vs heuristic in two scenarios: *i*) high connectivity, referred to as HC; and *ii*) low connectivity, referred to as LC.
- The results obtained in our studies show that in scenarios having very low connectivity (i.e., having isolated nodes) and scenarios having high connectivity (which in turn are related to scenarios with very high density of devices) the solutions obtained by the algorithm are very close to the optimal ones.
- Interestingly, the number of clusters obtained is very similar whereas the objective function values are slightly increased with respect to that values obtained when the mathematical model is considered; below 6% for the fastest configurations of the algorithm and when backup is considered and below 1% when no backup is considered and time to obtain solutions is increased. In certain LC scenarios, costs about 11% and 10% are observed when backup is considered and is not considered, respectively.
- However, in any case, the solving time is dramatically reduced compared to that of the model.

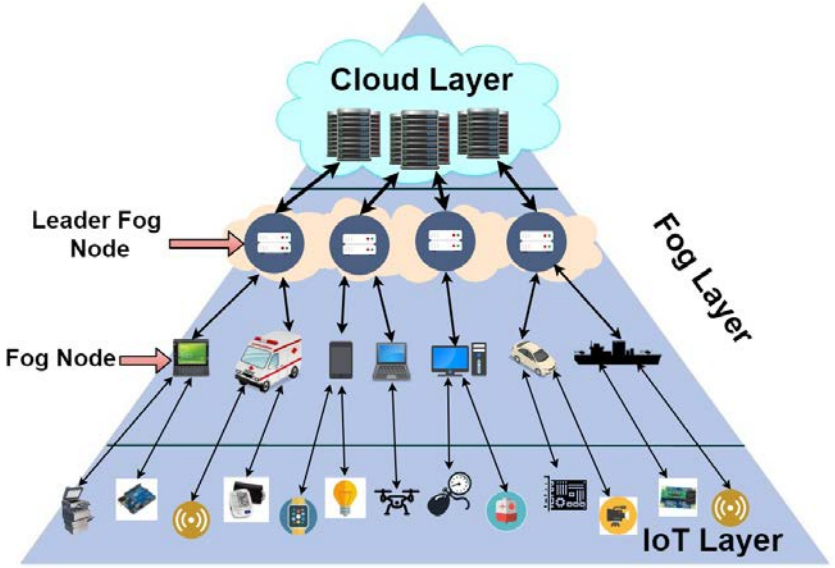
4. F2C: Categorization



4. F2C: Categorization

- Enormous diversity and heterogeneity of participating devices

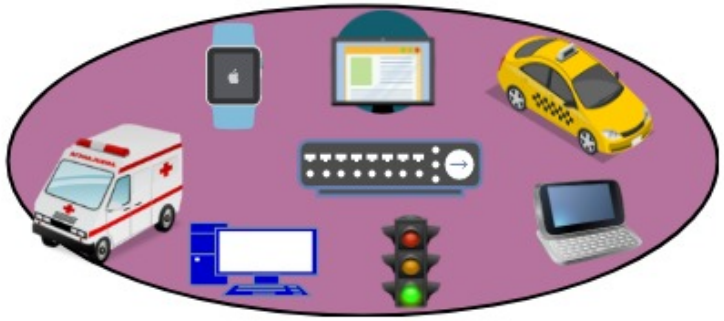
- Different types of services are offered to users



Hierarchical Architecture of the F2C paradigm

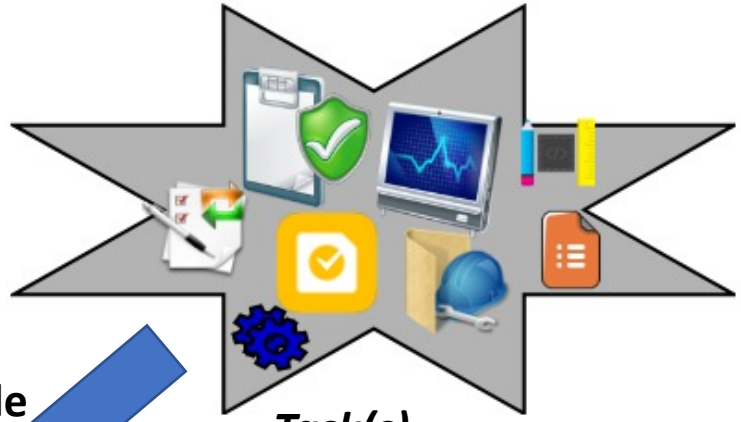
- Do not have the proper optimal resource allocation mechanism, in order to allocate the resource(s) to execute some task(s) and provide some service(s)

4. F2C: Categorization Scenario



Resource(s)

Allocate to execute

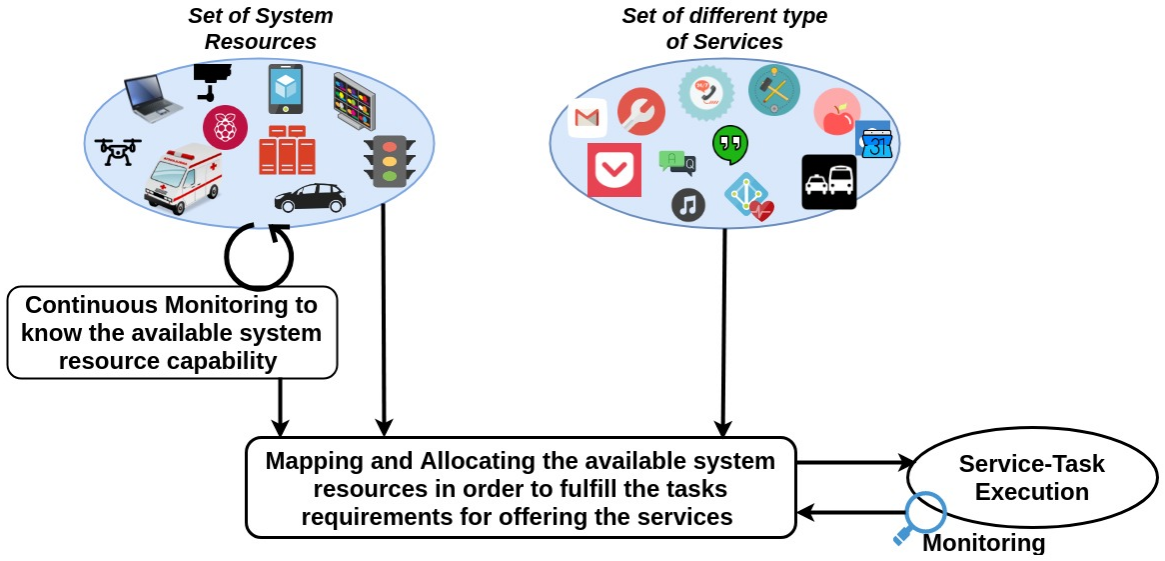


Task(s)

Provide

Service(s)

- To build an efficient system, it is necessary to properly utilize the system resource(s) for executing some task(s) and provide the service(s)

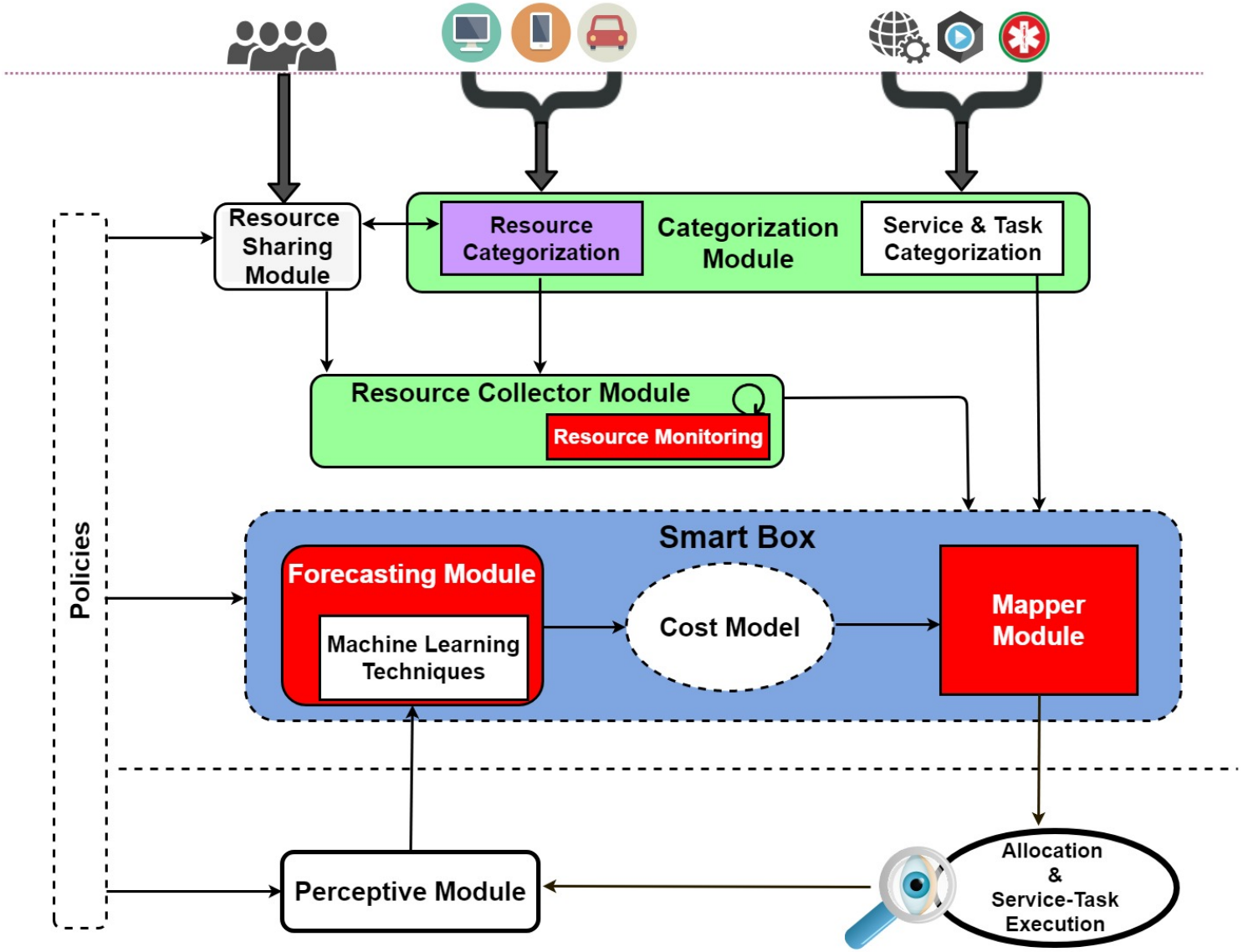


Outline structure of the Resource Management mechanism in the coordinated Fog-to-Cloud computing paradigm

- **Challenge for building the efficient resource management mechanism –**

  - ❖ Identification of the characteristics of participating devices
  - ❖ Knowledge about the service characteristics and task requirements
  - ❖ Need to know the available capacity of the system
  - ❖ Security-related issues, communication-related issues, SLA and Policy related issues *etc*…

Proposed Model: Resource Management Strategy for F2C



All F2C resources can be categorized on the basis of five (5) key aspects –

- ✓ **Device attributes –** *Hardware, Software, Network specification and Device-type info (i.e., Virtual or Physical)*
- ✓ **Cost information –** *Chargeable* or *Non-Chargeable*
- ✓ **History and Behavioural information –** *Reliability, Mobility, Participating role (i.e., Consumer, Contributor, or Both), information of the device location etc.*
- ✓ **Security and Privacy aspects -** *Device hardware security, Network security and Data privacy*
- ✓ **IoT and Attached components –** *Sensors, Actuators, RFID tags, and Attached resource components*

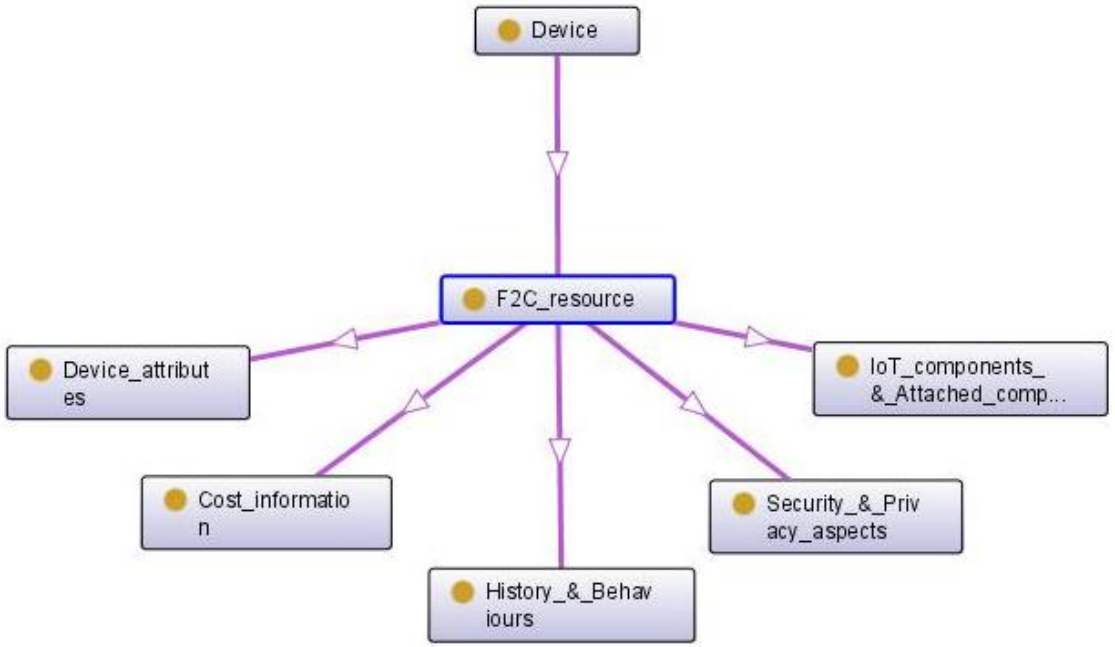


Fig 5. F2C Resource Characterization

Proper collection of this information not only helps to get the full knowledge about the F2C resource, but also adequate maintenance of this information for all F2C resources helps us identify the whole system capacity.

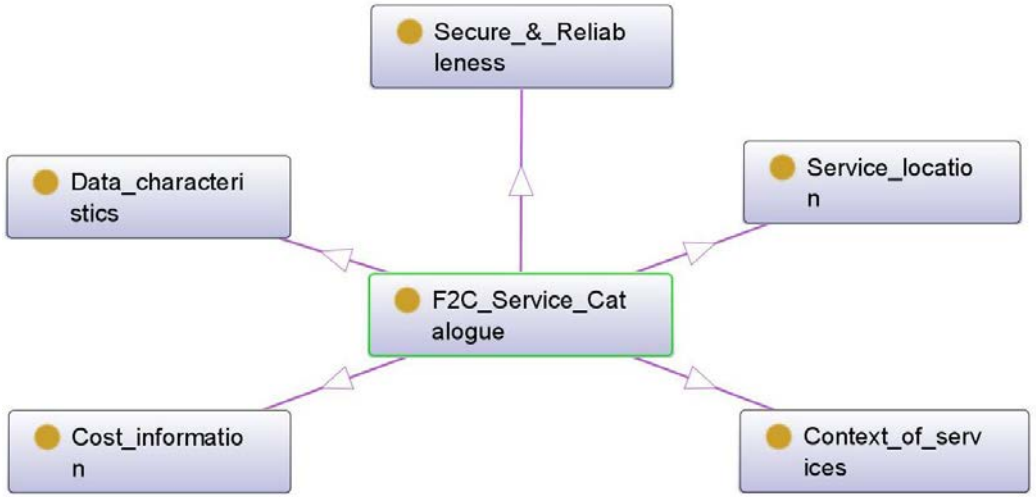
- **Definition of Service and Task:-**

'Service' is a composite made up of small blocks of functionalities, which can be offered by performing some certain 'Task'(s)

Relationship between Service and Task in the F2C

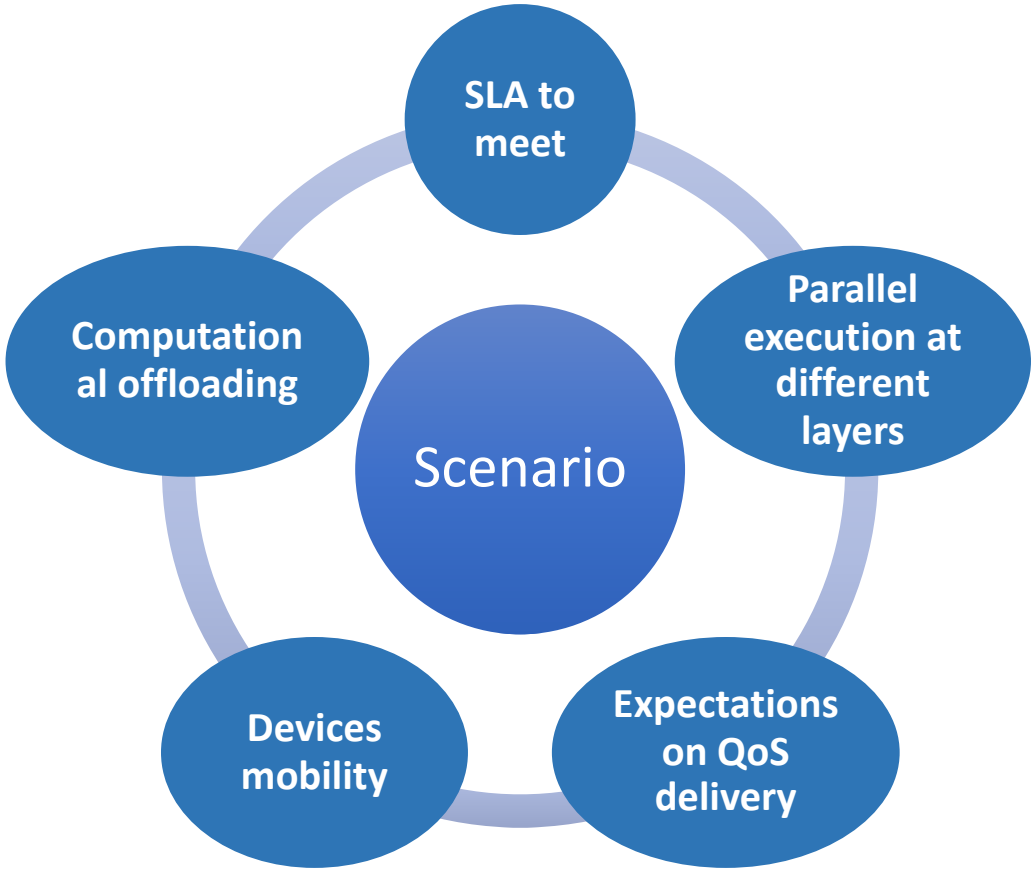
- **All the F2C Services can be classified based on *Five (5) key aspects –***

  - ✓ *Context of services (i.e.,* governmental, educational, transport, etc. related services*)*
  - ✓ *Service location (i.e.,* Cloud , Fog*)*
  - ✓ *Secure and Reliableness (i.e.,* security preferences)
  - ✓ *Data characteristics (i.e.,* amount of data processing*)*
  - ✓ *Cost information (i.e., Free service or chargeable)*

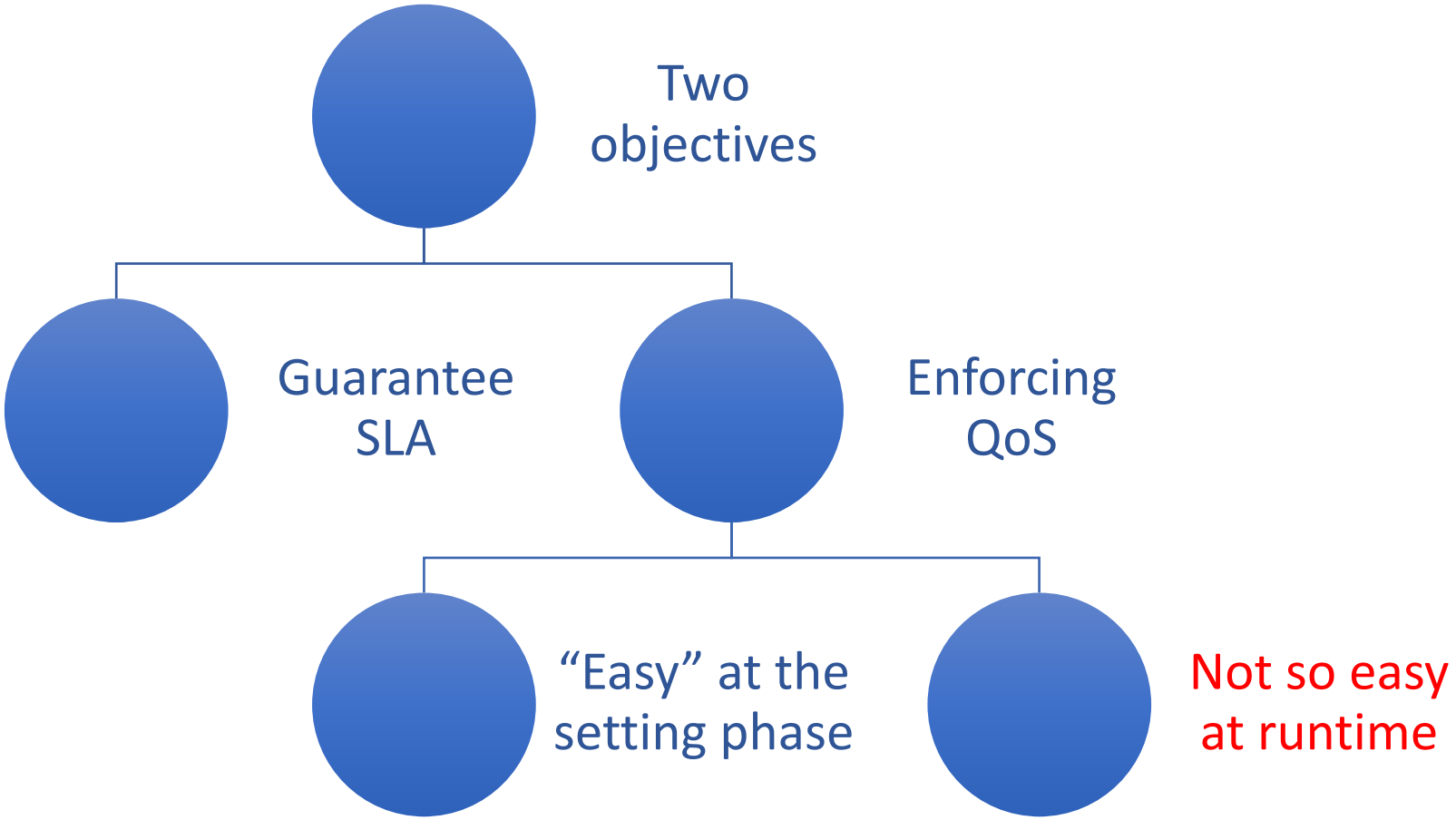


Service Characterization in the F2C

F2C: Migration (SLA + QoS enforcement)



F2C: Migration (SLA + QoS enforcement)



Glimpses towards the future

Multi-tenant fog

- Shall we share the edge infrastructure?
- May we infer from the public/private/hybrid cloud models?
- May telcos/vendors contribute?
- What about city owners?
- Data at the edge as a resource to share as well?

Challenges

- Engage users to contribute
- Engage infrastructure owners to contribute
- Develop new services
- Identify clients
- Define novel business models

Thank you all!!!

Questions time