Mobile Systems M

Alma Mater Studiorum – University of Bologna
CdS Laurea Magistrale (MSc) in
Computer Science Engineering

Mobile Systems M course (8 ECTS)
II Term – Academic Year 2019/2020

07 (opt) – 5G and Mobile Edge Computing

Paolo Bellavista
paolo.bellavista@unibo.it

http://lia.disi.unibo.it/Courses/sm2122-info/

5G converged world

Voice (VoIP)

Audio/Video Conference

Chat and messagging

Video on Demand (VoD)

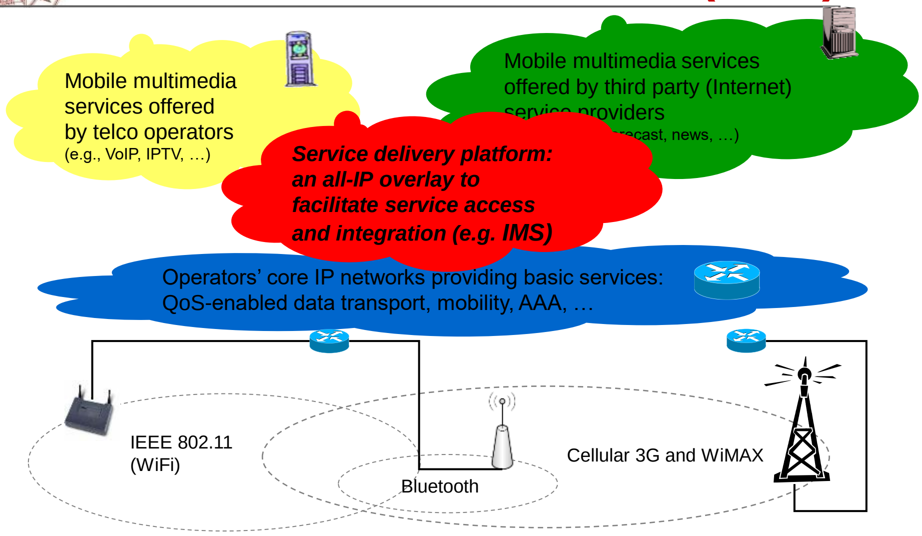
And many more…

- Push To Talk (PTT)
- PTT over Cellular (PoC)
- IPTV
- Video sharing
- …

Ever-increasing demand and diffusion of mobile multimedia services during the last two decades, driven by:

– New powerful ***devices*** and ***wireless technologies/infrastructures***
– New (mobile) ***services services services***

Service delivery *(from 3G on…)* in Next Generation Networks (NGN)



Highly differentiated ***(wireless) access networks***

Service delivery in 3G (and more…)

- Legacy
  - Circuit switched part (GSM)
  - Packet switched (GPRS)
- NGN portion
- Interworking between ***Legacy*** and ***NGN portion***
- Our focus now is on the NGN portion (refer to initial parts of the course for the Legacy portion)

Key components

- Transport (Radio access network known as UTRAN – UMTS Terrestrial Radio Access Network)
  - Below IP: radio technology, such as WCDMA (for 3G)
  - IP + TCP/UDP

- Services (Basic + value added services)
  - **IP Multimedia Subsystem (IMS) standard**

    → ***Overlaid on top of the IP transport***

IMS Basics & Layering

- Basic services – Call / session layer
  - Signalling entities
  - Databases for
  - Interworking with 2G/3G/4G…

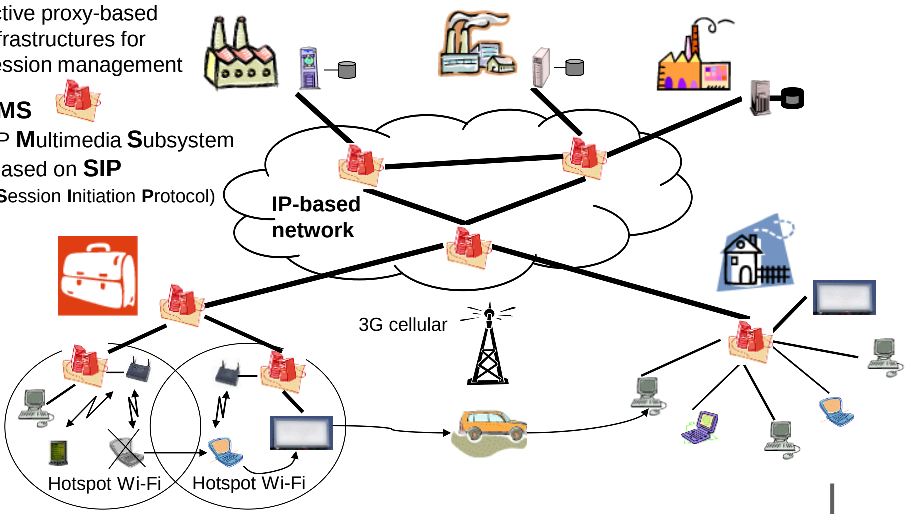
- Value added services layer
  - Application servers
  - Media resources

Services (value-added services) also called application / services
Services (Basic services) also called call/session control
Transport (Below IP + IP + transport layer) also called bearer

Overall:
a proxy-based approach

New protocols and active proxy-based infrastructures for session management

IMS
IP **M**ultimedia **S**ubsystem based on **SIP**
(**S**ession **I**nitiation **P**rotocol)



Some background:
SIP – Session Initiation Protocol

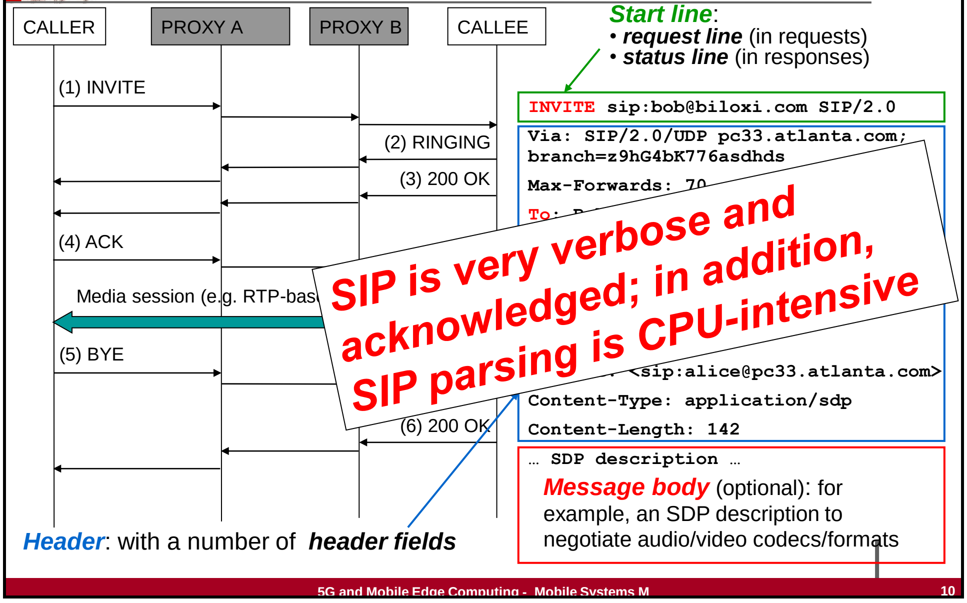
- SIP defines a ***signaling framework*** and related ***protocols and messages*** to setup ***any kind of session*** (work at the Open Systems Interconnection – OSI – ***session layer***)
  - SIP is very ***open*** and ***general purpose*** ☺
  - SIP includes several core facilities for ***mobility management***, ***session initiation***, ***termination***, and ***transfer***, …
  - SIP ***does not*** include some basic services ☹ (e.g., AAA, resource booking, …)
- SIP ***is not*** a ***data/media transmission protocol***

  Other specific protocols for that: Real-time Transport Protocol (RTP), RTP Control Protocol (RTCP), Real Time Streaming (RTSP),…
- SIP usage ***examples***
  - Setting up and tearing down VoIP voice calls
  - Instance messaging and presence service: SIP for Instant Messaging and Presence Leveraging Extensions – ***SIMPLE***
  - Session transfer and call re-direction

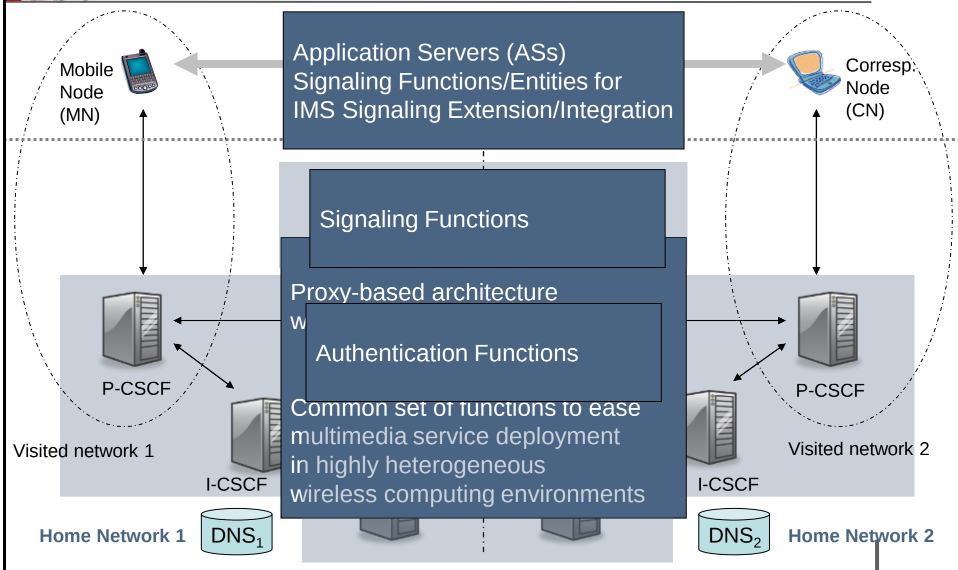
SIP in a nutshell

- SIP core signaling
  - HTTP-like text-based protocol and email-like SIP identifiers (**addresses**)
  - Client/server protocol (request/response protocol)
  - Standardized session control messages
    - INVITE, REGISTER, OK, ACK, BYE, …
- SIP proxy-based framework and ***main entities***
  - **User agents:** end points, can act as both user agent client and as user agent server
    - **User Agent Client:** create new SIP requests
    - **User Agent Server:** generate responses to SIP requests
  - **Dialog:** peer to peer relationship between two user agents, **established by specific methods**
  - **Proxy servers:** application level routers
  - **Redirect servers:** redirect clients to alternate servers
  - **Registrars:** keep tracks of users

SIP VoIP call initiation example: INVITE dialog



IMS – IP Multimedia Subsystem



IMS functional entities: DNS and HSS

Domain Name System (***DNS***):

- Standard Internet naming service
- Employed by IMS to ***resolve the IP addresses of CSCFs*** and ***ASs***
  - → can be used for ***load balancing*** ☺
    (but… only with limited DNS-query frequency)

Home Subscriber Server (***HSS***):

- ***SIP requests forwarding*** in the appropriate direction (terminals or IMS network)
- Use of Diameter for user AAA
- Storage of all user-related subscription data, such as authentication data and profiles for clients (by using standard Data Base Management System – DBMS)
- A network may contain one or several
  - Subscriber Location Function (SLF) to map users to specific HSS

IMS functional entities: Proxy-CSCF

Proxy-Call Session Control Function (***P-CSCF***):

- First contact point in the IMS network in ***either visited domain*** or ***home domain***
- Outbound / In-bound SIP proxy
  (all requests from/to IMS terminals go through it)

Main P-CSCF functions

- ***SIP requests forwarding*** in the appropriate direction (terminals or IMS network)
- Several ***other functions***:
  - Security
  - Generation of charging information
  - Compression and decompression of messages

IMS functional entities: Interrogating-CSCF

Interrogating-Call Session Control Function (***I-CSCF***):

- SIP proxy at the edge of the administrative ***home domain***
  - There may be several in the same network for scalability reasons
  - Listed in the domain name server (DNS-based scalability)
- SIP redirect stateless server

Main I-CSCF functions

- ***Interaction with HSS*** to determine the S-CSCF associated with the client (***Diameter*** protocol)
- ***Redirection*** and ***routing of incoming SIP requests*** to S-CSCF

  → can be used to ***dynamically select less-loaded S-CSCFs (e.g. through DNS)*** ☺

IMS functional entities: Serving-CSCF

Serving-Call Session Control Function (***S-CSCF***):

- Always located ***in home domain***
- SIP proxy + SIP registrar with possibility of performing session control

Main S-CSCF functions

- ***Binding*** between ***IP address*** (terminal location) and ***user SIP address***
- Interaction with application servers for ***value added service purpose***
- Translation services (Telephone number / Sip URIs)
- Message routing (by using so-called ***IMS filtering criteria***)

  → can be used to ***statically divide incoming load according to user identity/profile*** ☺

15

IMS functional entities: AS

Application Server (***AS***):

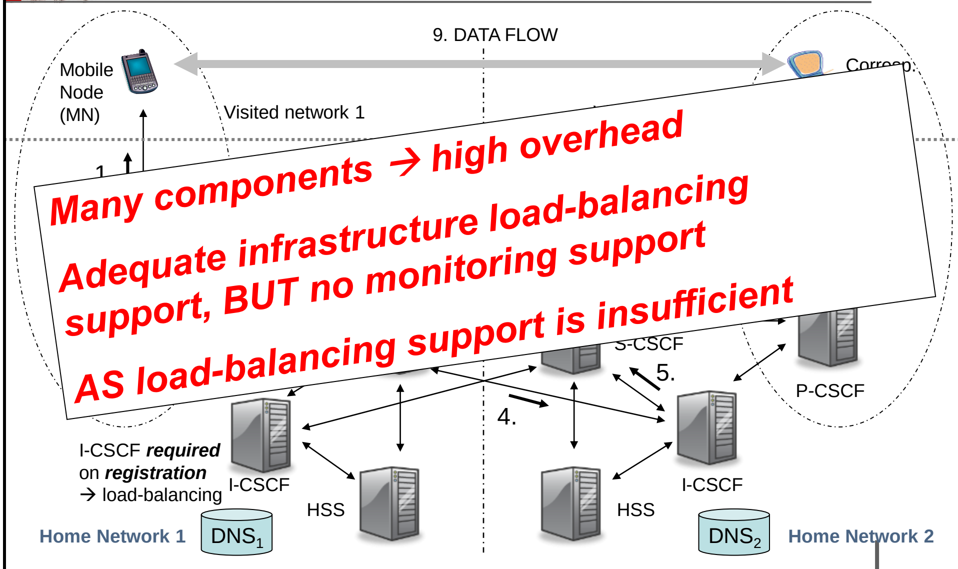
- ***Host services*** and ***execute services***
- Communicates using SIP: ***very costly!!*** ☹
  - Each ***interposed AS*** generates 2 msgs (processed+ACK)
  - Complex coordination for ***stateful*** and ***distributed ASs***

Several AS types with different functions

- ***SIP AS***: ***signaling specific*** architecture (services can work only in SIP environment)
- Other types: Open Service Architecture – Service Capability Server (OSA/SCS), IP Multimedia Service Switching Function (IM-SSF), …

16

IMS in action for a VoIP call

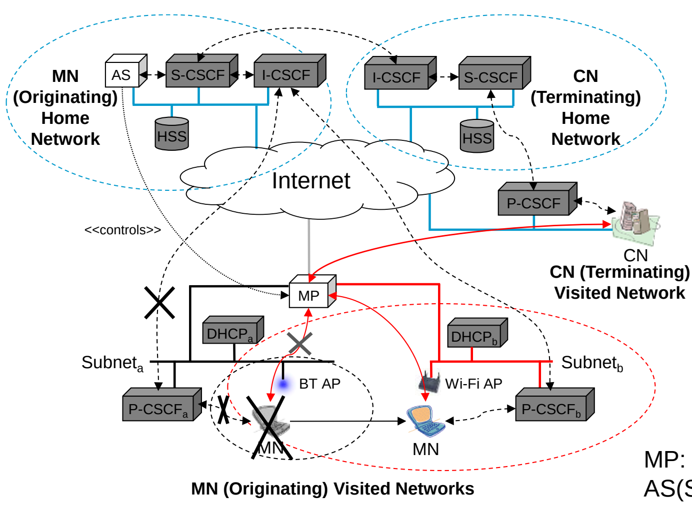


Many components → high overhead

Adequate infrastructure load-balancing support, BUT no monitoring support

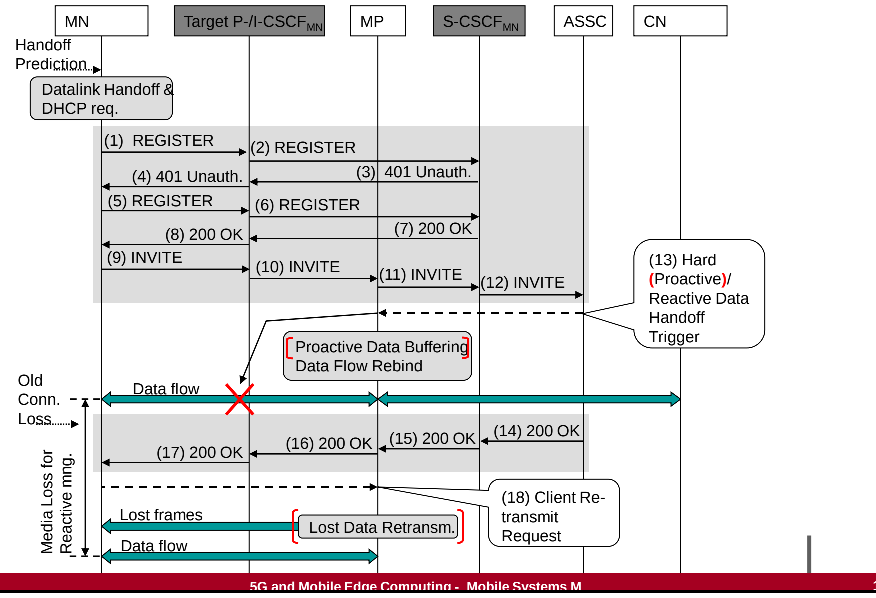
AS load-balancing support is insufficient

IMS Handoff Management



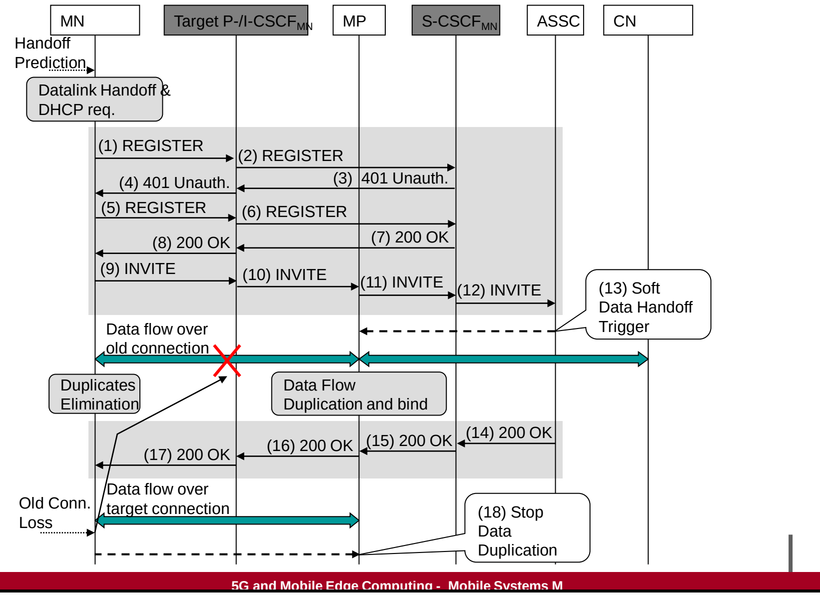
MP: Media Proxy
AS(SC): AS for Service Continuity

Optimized hard proactive/reactive IMS Handoff Management



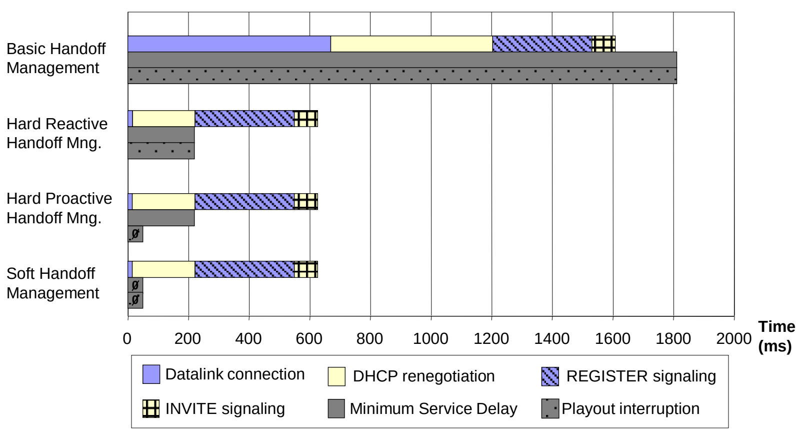
19

Optimized soft IMS Handoff Management



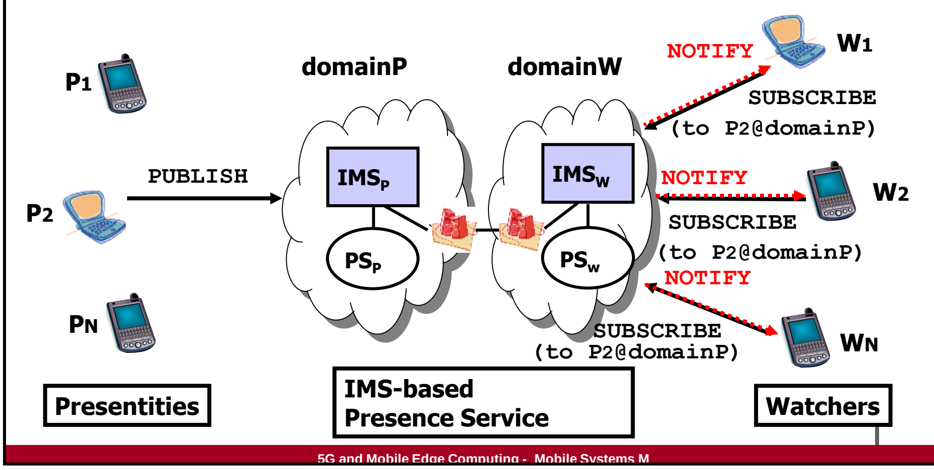
20

Advanced IMS Handoff Management: Some experimental results



Another example: IMS-based presence service

Presence service (PS) permits users and hw/sw components, called ***presentities (P_i)***, to convey their ability and willingness to communicate with subscribed ***watchers (W_j)***

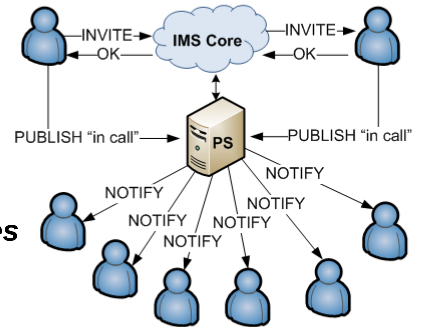


Scalability issues at a glance

High mobility & context changes

- Higher signaling traffic ***(message dimension + frequency)***
- Richer services, such as VoIP+PS (message ***multiplying effect***)
- ***Many traversed signaling entities*** (proxies-based architecture…)
- Plus, specific SIP protocol issues (***message verbosity*** and ***ACKs***)

New services VoIP+PS (call-status notification)



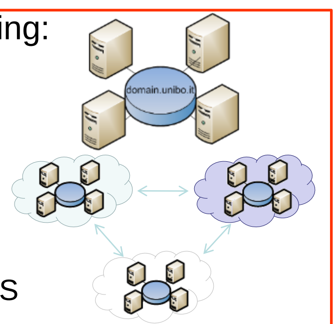
→ Need for a better understanding of IMS ***scalability shortcomings*** and ***load-balancing support*** both at ***infrastructure*** and ***service*** levels

IMS scalability: (partial) solutions

- **Single host** (local) optimizations w/out (or with minimal) coordination:
  - Selective ***message dropping***
  - SIP message ***compression*** and ***incremental parsing*** techniques
  - Stateful vs Stateless SIP proxies

Widely diffused and standardized

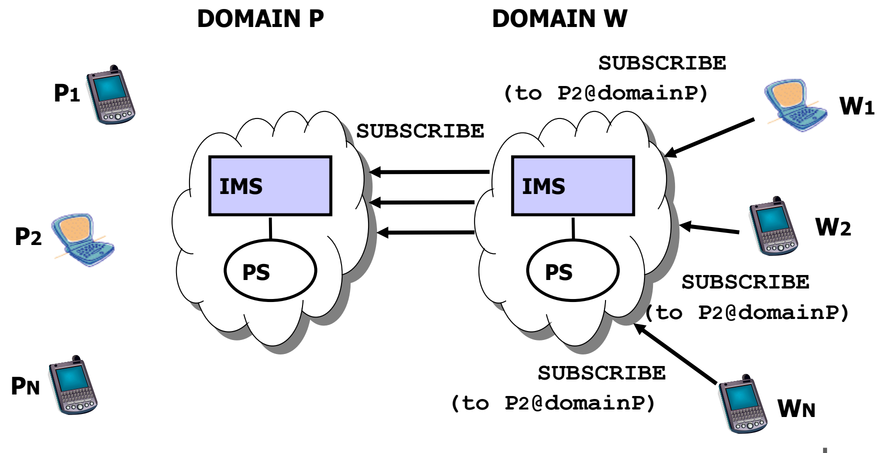
- **Intra-domain** (distributed) load-balancing:
  - ***Infrastructure-level monitoring*** and ***dynamic load-balancing*** operations
  - ***Service-level*** AS coordination protocols ***(also ad-hoc and NON-IMS-compliant optimized protocols!!)***
- **Inter-domain** protocol optimizations:
  - Limit traffic among different domains
  - ***Service-level*** message processing at IMS domain borders ***(BUT, IMS compliant)***



IMS PS scalability use case

P: Presentity **PS**: Presence Server
W: Watcher

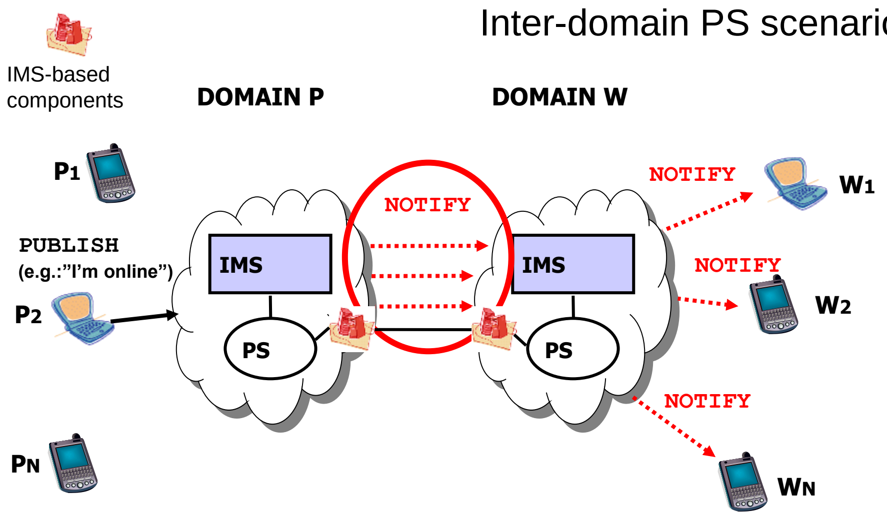
Inter-domain PS scenario



25

IMS PS scalability use case

Inter-domain PS scenario

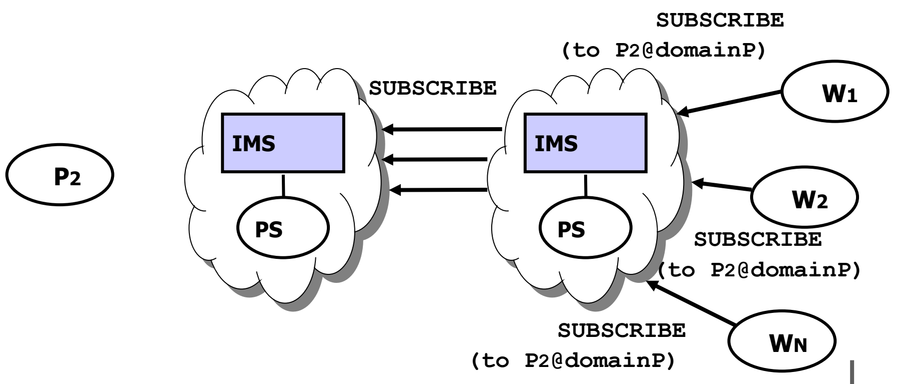


PS is very prone to scalability issues!!

26

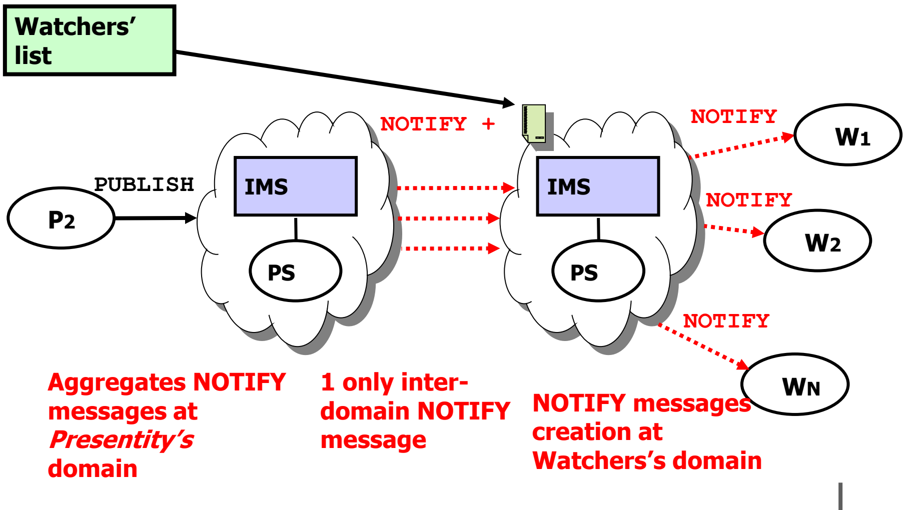
Possible inter-domain optimizations: Common NOTIFY

"***Several watchers*** subscribed to ***one presentity***"



27

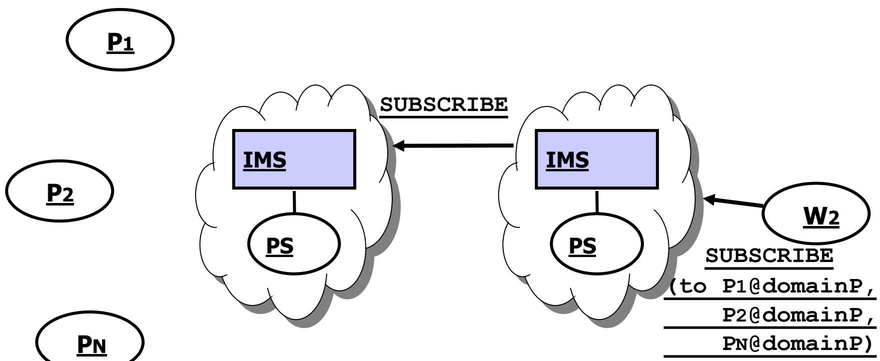
Common NOTIFY



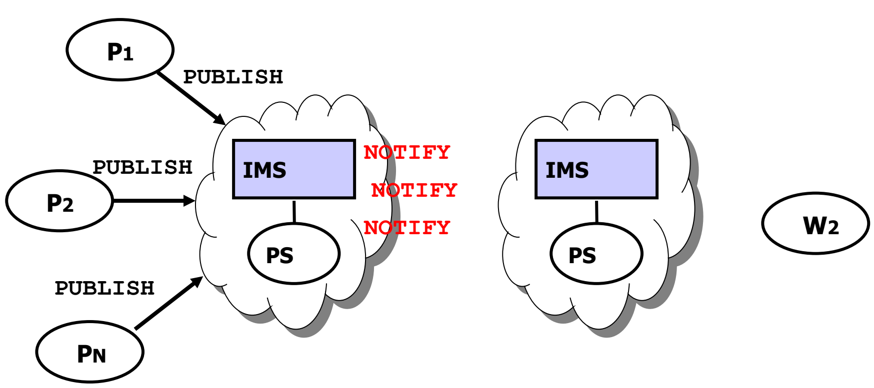
28

Batched NOTIFY

“One ***single watcher*** subscribed for ***multiple presentities***”

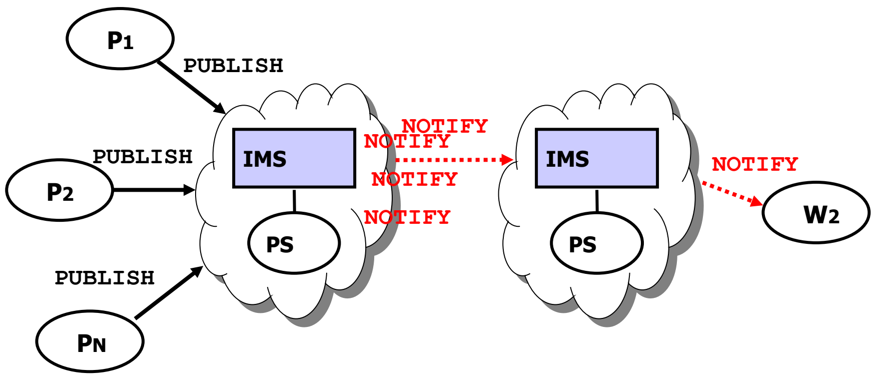


Batched NOTIFY



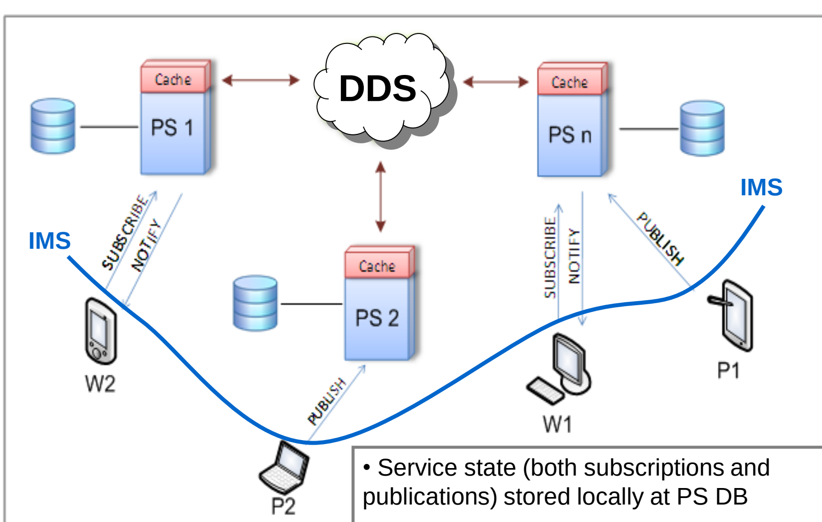
Time-based (periodic)
NOTIFY message batching

Batched NOTIFY



only 1 inter-domain NOTIFY message

Intra-domain optimizations for the IMS PS



- Service state (both subscriptions and publications) stored locally at PS DB
- PUBLISH write-through (using DDS)

Session Control and IMS Wrap-up

- Session control management is key to integrate added-value services in telco NGNs
- For IMS, strong need for scalable solutions
  - Both at the ***infrastructure*** and ***service*** level
- Interoperability and standard compliancy
  - ***Full IMS standard compliance*** for inter-domain optimization techniques
  - ***Ad-hoc solutions*** and ***integration with other emerging standards*** at intra-domain level

4G and beyond…

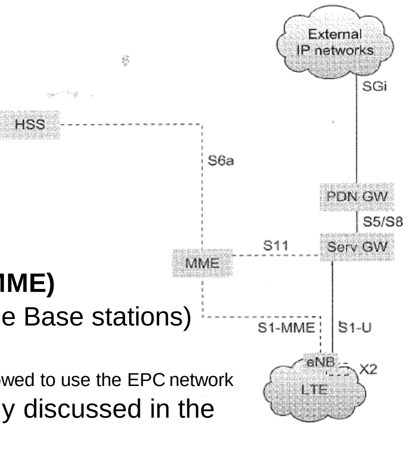
- 4G Transport
  - LTE:
    - Radio access network known also known as Evolved - UTRAN
      - Base stations called eNodeB
      - OFDM technology
  - IP
  - UDP/TCP/ SCTP (a more reliable alternative to TCP)
- Above 4G Transport, Evolved Packet Core (EPC) can accommodate other radio access networks such as:
  - Legacy 3GPP radio access: GPRS (2.5G), UTRAN (3G), HSPA (3.5G)
  - Non 3GPP radio access: WiFi, WiMax, CDMA2000, …

EPC Architecture

- Key principles
  - Flat architecture
- As few entities/nodes as possible
  - Clean separation between control / signalling path and data path Note:
    - signalling has a very broad meaning and does not mean multimedia session signalling in this context
      - Means control of data path

EPC Basic Architecture

- Basic EPC architecture for LTE (Reference 1)
  - Dotted lines: Signaling/control path
  - Solid lines: Data path



Signaling / control path

HSS
- Subscriber data base

Mobility Management Entity (MME)
- Controls the ENodeB (eNB, the Base stations)
- Interacts with the HSS
  - Find out if for instance the user is allowed to use the EPC network
- Mobility (using Mobile IP widely discussed in the first part of the course)
- Security

EPC Basic Architecture

About employed protocols:

- **SCTP (Stream Control Transport Protocol) used by MME for reliability reasons**
  - SCTP is a more reliable alternative to TCP
    Multi homing
    Four way handshaking
- **Diameter over SCTP is used for interactions with the HSS**

EPC Basic Architecture

Data path:

- **Packet Data Network (PDN) Gateway: gateway towards external networks / nodes such as:**
  - Internet
  - Application servers
  - IMS
  - Other service delivery platforms
- **Serving Gateway (Serv GW): belongs to both signaling/control path and data path**

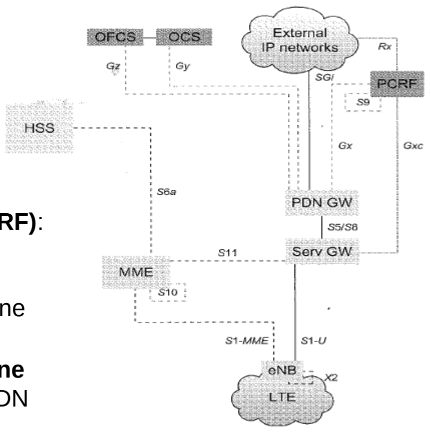
  On the signaling/control path
  - Controls the MME
  - Marks "packets" for QoS differentiation purpose

  On the data path
  - Buffers data as appropriate

EPC – A more advanced architecture

EPC for LTE (Reference 1)

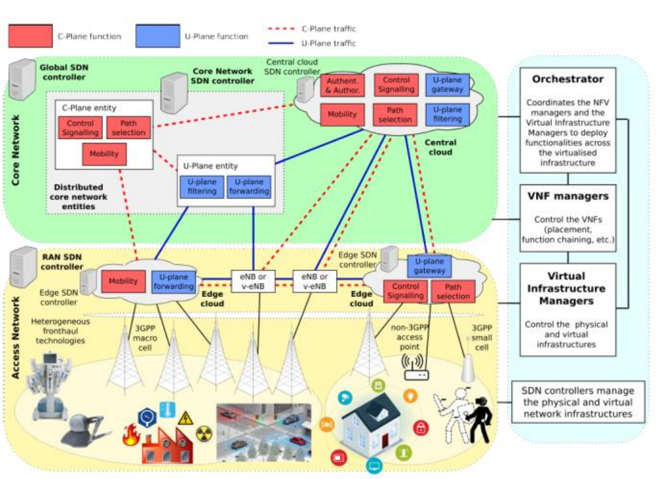
- Dotted lines: signaling/control path
- Solid: data path



New entities:

- **Policy and Charging Rule Function (PCRF)**: defines through policies the treatment a specific IP flow shall receive (e.g., QoS preferences and/or charging, such as on-line credit card verification)
- **Online charging system (OCS) and offline charging system (OFCS)**: interact with PDN gateways for charging purpose based on parameters such as time, volume, events

And, eventually, 5G Core Network



Picture from Elsevier ComNet, "Softwarization and virtualization in 5G mobile networks: Benefits, trends and challenges", 2018.

We do not see all the details of the (not so many) evolutions of 4G elements, BUT main novelties about the 5G core network, that are:

- Virtualization of all core elements and functions for i) (radio) access, ii) signaling/control plane, and iii) data plane + added-value services
- Computation at the edge
- High flexibility

More details in the next part...

Edge Computing (and IoT…): Motivations

Number of connected devices worldwide continues to grow (triple by the end of 2019, ***from 15 to 50 billions***)

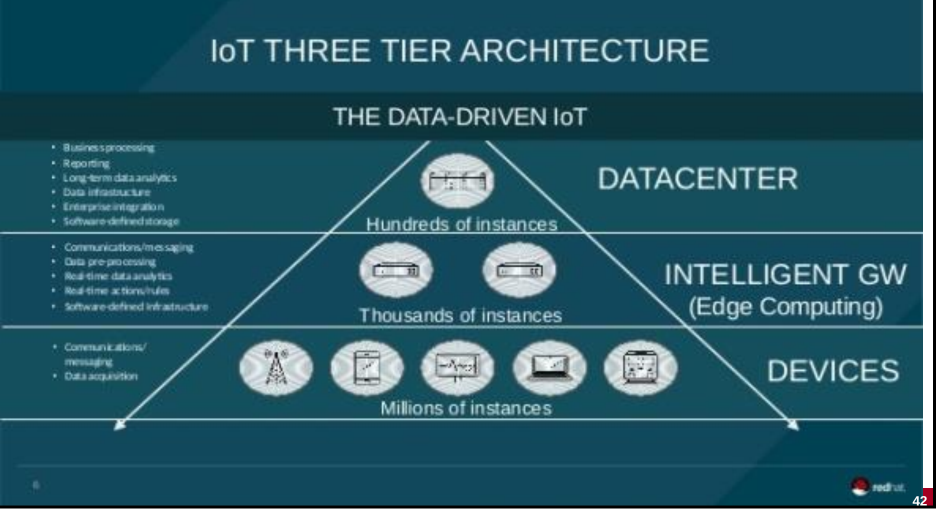
Deep transformation of how we organize, manage, and access ***virtualized distributed resources***

Is it reasonable that we continue to identify them with the ***global location-transparent cloud***?

In particular, in many ***industrial IoT application scenarios***:

- strict ***latency*** requirements
- strict ***reliability*** requirements
  - For instance, ***prompt actuation of control loops***
  - Also associated with ***overall stability and overall emerging behavior***

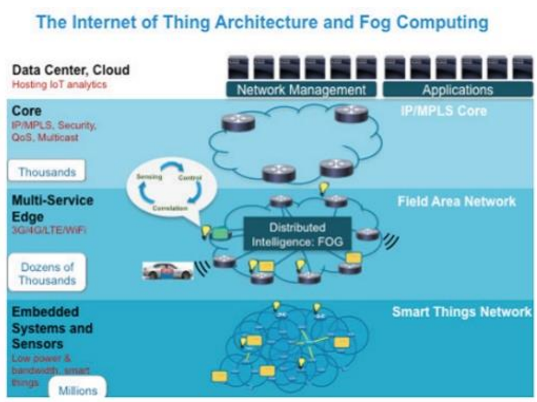
Edge Computing for Industrial IoT: Quality Requirements



Edge Computing for Industrial IoT: Quality Requirements

Towards the vision of ***efficient edge computing support*** for ***"industrial-grade" IoT applications***

- ***Latency constraints***
- ***Reliability***
- ***Privacy of industrial data***
- ***Decentralized control***
- ***Safe operational areas***
- ***Scalability***



Edge and 5G for Constrained Latency

Industry 4.0

- Increase the **flexibility, versatility, productivity, resource efficiency & usability** of industrial production
- **Connectivity as a key enabler** for cyber-physical production systems

Future Industrial Connectivity Infrastructures

5G

- Strong focus on **machine-type communication** and the IoT[1]
- **URLLC[2] + mMTC[3]** enable completely new applications, also in industry
- 5G is **more than wireless**

Enabler for new applications & use cases and for lifting I4.0 to the next level

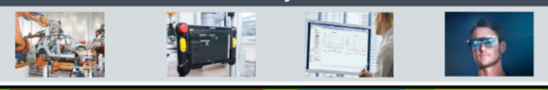
Images: BOSCH

Edge and 5G for Constrained Latency

Selected Performance Requirements

Credits to Bosch

Industry 4.0



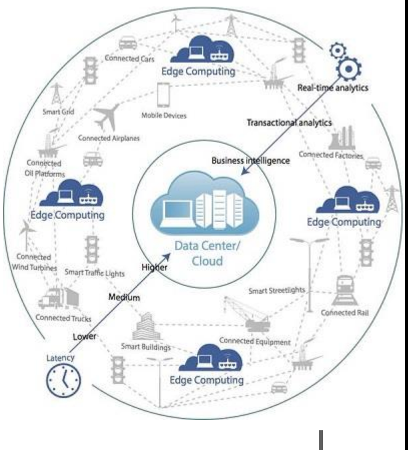
	Motion Control	Safety Traffic	Condition Monitoring	Augmented Reality
Latency / Cycle Time	250 µs – 1 ms	~10 ms	100 ms	10 ms
Reliability (PER[1])	1e-8	1e-8	1e-5	1e-5
Data Rate	kbit/s – Mbit/s	< 1 Mbit/s	kbit/s	Mbit/s - Gbit/s
Typical Data Block Size	20-50 byte	64 byte	1-50 byte	> 200 byte
Battery Lifetime	n/a	1 day	10 years	1 day

uRLLC[2] → most challenging (Motion Control, Safety Traffic) · Massive MTC[3] (Condition Monitoring) · Extreme Broadband + Low Latency (Augmented Reality)

Edge Computing: Definition (to be discussed…)

Edge computing = ***optimization of “cloud computing systems”*** by performing data processing (only?) at ***the edge of the network***, near data sources. ***Possibility of intermittent connectivity***

Edge computing can include technologies such as ***wireless sensor networks, mobile data acquisition,*** mobile signature analysis, ***cooperative distributed peer-to-peer ad hoc networking and processing***, distributed data storage and retrieval, ***autonomic self-healing networks***, remote cloud services, …

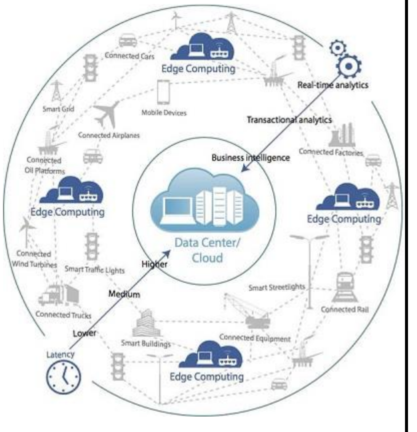


Edge Computing: Definition

Edge computing = ***optimization of "cloud computing systems"*** by performing data processing (only?) at ***the edge of the network***, near datasources. ***Possibility of intermittent connectivity***

IMHO, crucial to have ***virtualization techniques at edge nodes***

Synonyms (???) = fog computing, mobile edge computing, multi-access edge computing, cloudlets, …



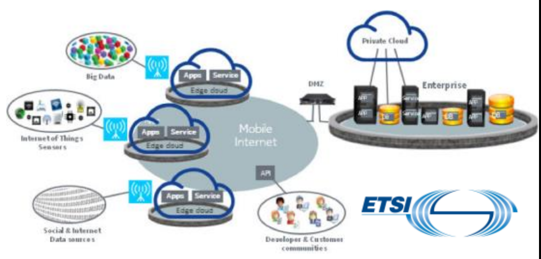
Fog Computing

- **Fog Computing** paradigm is proposed to overcome the limitations of Cloud Computing
- Fog supports the **IoT** concept

- **Cons**: typically fog is used for resource-poor devices and sensing scenario and **Smart Gateways (SGs)** are unable to host heavy computations

Multi-access Edge Computing

- The **MEC** architecture is an ETSI standard to overcome the challenges of limited-resources mobile devices
- MEC offers high bandwidth, low latency and support to the mobility of nodes

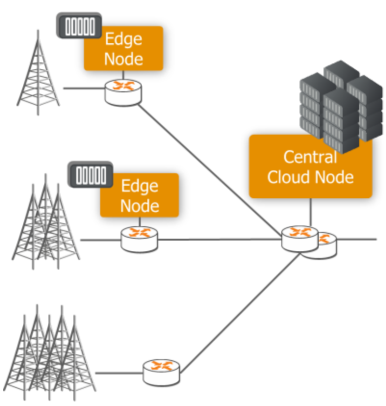


- **Cons**: limited number of edges and low re-configuration rate, due to high costs of configuration and maintenance

Notable example: ETSI Multi-access Edge Computing (MEC)

MEC is bringing computing close to the devices (in the base stations or aggregation points)

- **On-Premises:** the edge can be completely isolated from the rest of the network
- **Proximity:** capturing key information for analytics and big data
- **Lower Latency:** considerable latency reduction is possible
- **Location awareness:** for location-based services and for local targeted services
- **Network Information Context:** real time network data can be used by applications to differentiate experience

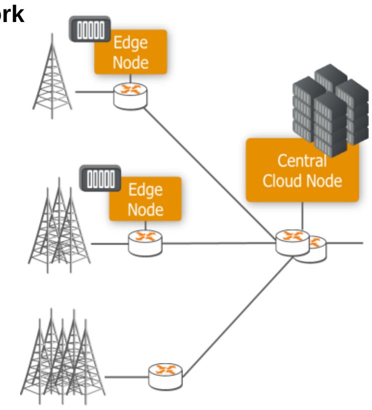


Local vs Global: the MEC Use Cases

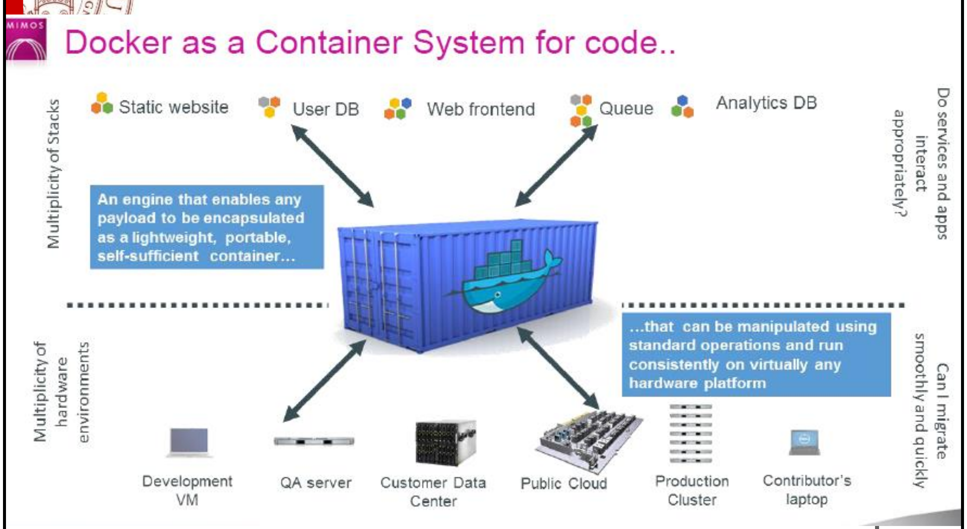
Depending on the integration with the core network three types of use cases are defined

- **Private Network Communication (factory and enterprise communication)**
  - ❑ Providing support for on-premises low-delay private communication
  - ❑ Providing secure interconnection with external entities
- **Localized Communication (traffic information and advertisements)**
  - ❑ Providing support for localized services (executed for a specific area)
  - ❑ Specific ultra-flat service architectures
- **Distributed Functionality (content caching, data aggregation)**
  - ❑ Providing extra-functionality in specific network areas

We get back to this in few slides...

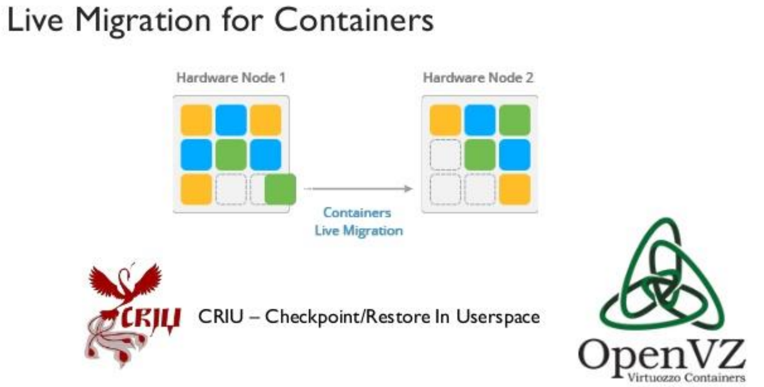


Edge Computing & Docker



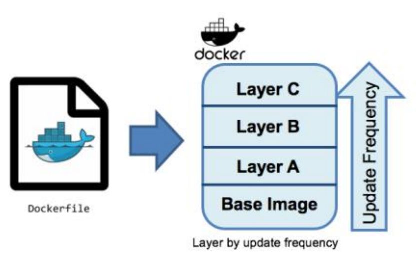
Edge computing empowered by *containerization*

Container live migration and state maintenance:
which tradeoff between state consistency and overhead?



Edge computing empowered by containerization

- ***Layering of session/application state***
- Big data analytics on ***probability of state modification*** in the different layers
- Dynamic tradeoff selected for each state layer separately
  - Migration, local/distributed checkpointing



- Service components?
- Data/state?

Plus ever-increasing frequencies in CI/CD DevOps processes…

I'll go back to this… and for additional details, please see our papers (refs section)

Edge/Fog Computing and 5G: A first wrap-up

5G plus edge/foc cloud computing (***cloud continuum***) can contribute to improve:

- ***Efficiency***
- ***Latency minimization***
- ***Cost reduction***
- ***QoE in terms of interaction and collaboration***
- ***With customized/personalized properties about security, privacy, data protection/ownership, …***

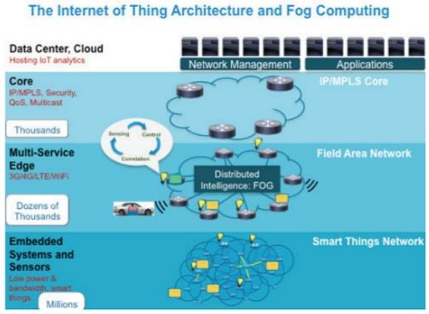
And not only for the above use cases!!!

Edge Computing for IoT Apps: Recent/Ongoing Directions

- ***Architecture modeling***
- ***Quality support even in virtualized envs***

But also:

- Data aggregation
- Control triggering and operations
- Mgmt policies and their enforcement
- …

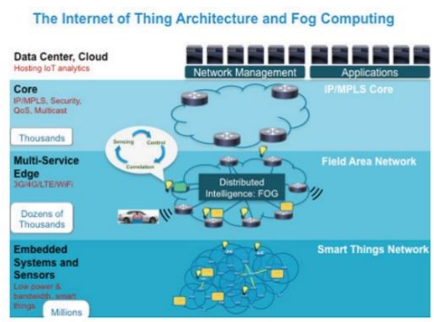


Architecture Modeling

Dynamic distribution of storage/processing (network resource allocation?) functions in all the three layers of a node-edge-cloud IoT deployment environment

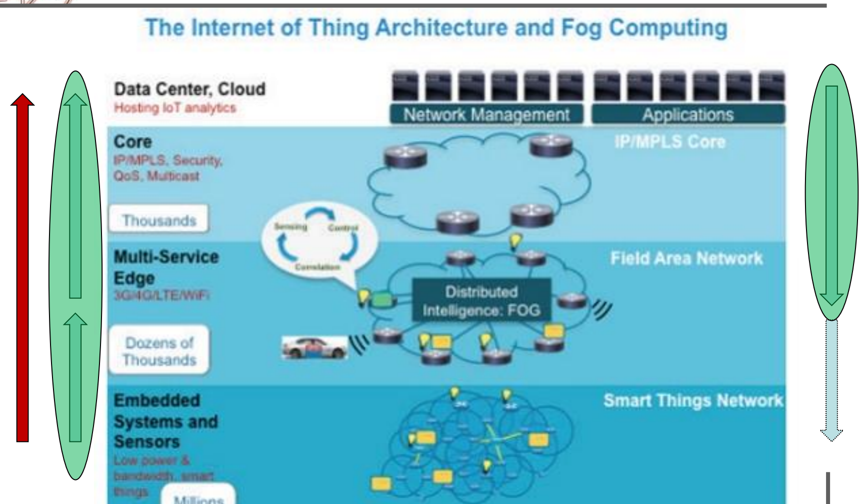
Different and richer concept of ***mobile offloading***

- mobile app avatars/clones in living in edge/core cloud
- not only offloading…



57

Architecture Modeling
Need for new models
Offloading and Onloading



58

Architecture Modeling
Need for new models

Need for new models for richer mobile offloading:
- From sensors/actuators to the cloud (traditional)
- **From sensors/actuators to the edge**
- **From the edge to the cloud**

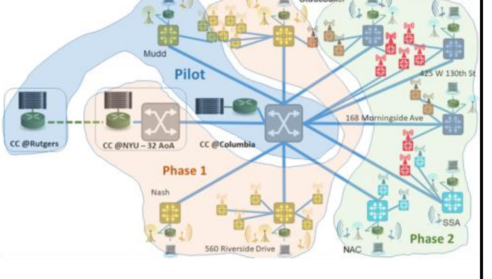
But also:
- **From the cloud to the edge**
- From the edge to sensors/actuators

Growing overall status visibility vs.
growing decentralization and autonomy

Offloading heavy computations: the real COSMOS example

COSMOS Deployment: NYC Coverage Areas

- Pilot – planned for end of 2018
- Phase 1 in 2019, Phase 2 by 2020

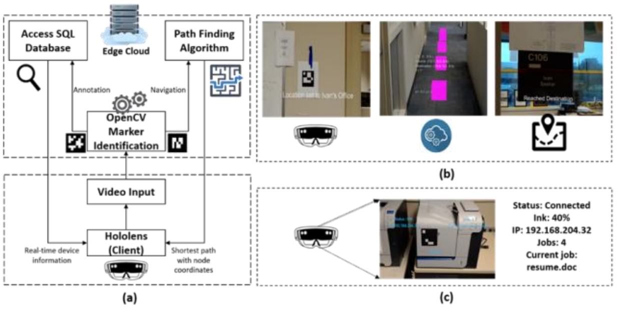


Mudd Broadway Amsterdam

- Phase 1 Columbia/CCNY – ~15-20 nodes
- Phase 2 – ~40 nodes

Offloading heavy computations: the real COSMOS example

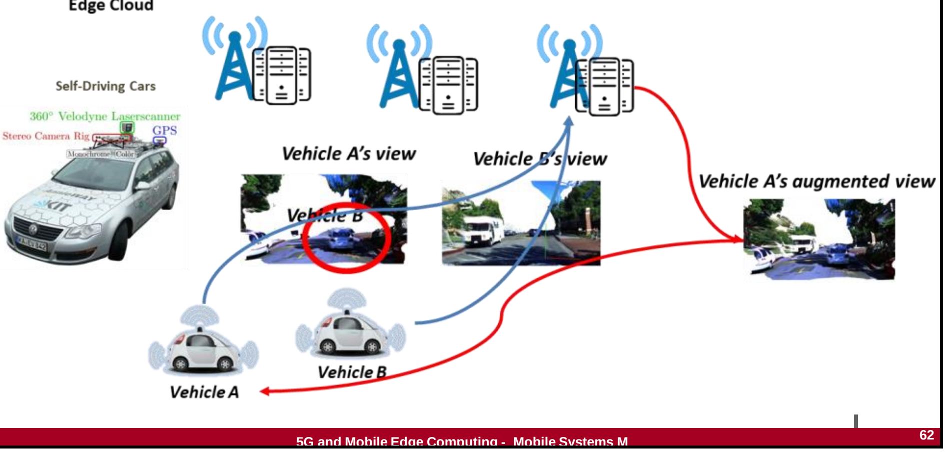
COSMOS Experiments: AR Applications



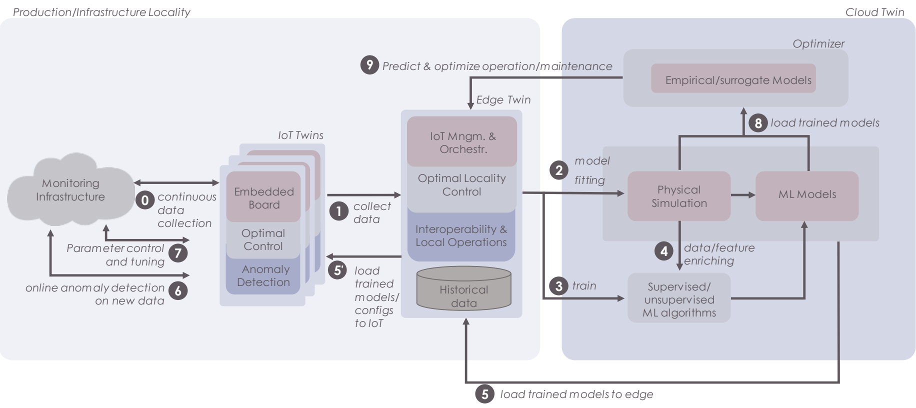
(a) AR application flow; (b) Smart meeting application using indoor navigation; (c) Annotation based assistance

Offloading heavy computations: the real COSMOS example

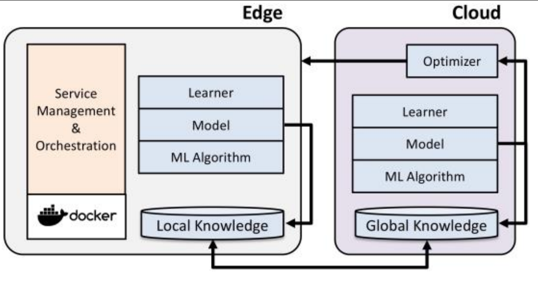
COSMOS Experiments: Cloud Assisted Autonomous Vehicle



On-loading to the edge in Fog/Edge/Core-cloud Continuum: the IoTwins EU H2020 Project

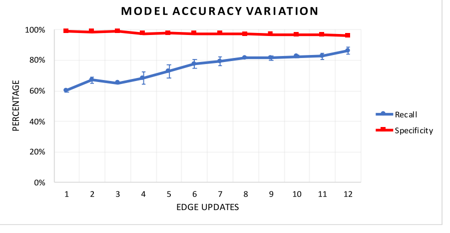


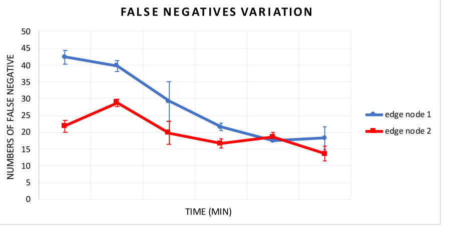
Onloading updated ML models: A continuous process



- A set of ML algorithms **run at the edge** for online analysis
- Learning module able to train model (**Digital Twins**)
- An **Optimizer** module that sends feedback to reinforce distributed models

Onloading updated ML models: A continuous process





- **By sending reinforced models from the cloud towards the edge:**
  - the total model accuracy is more or less the same
  - more accuracy to predict negative instances

Quality Support even in Virtualized Envs

But definitely, here we are not starting from scratch…

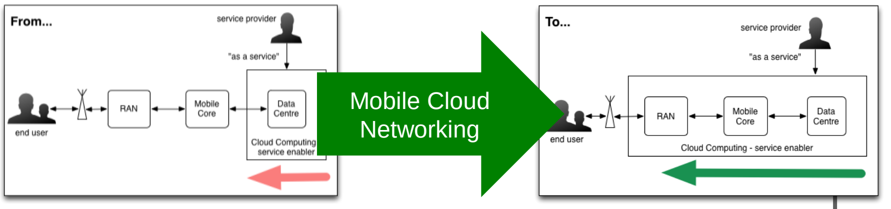
Notable experience of mobile cloud networking for telco services with quality requirements

- ***Carrier-grade industrial usage of elastic*** distributed cloud resources for telco support infrastructures
- ***Quality constraints of typical telco providers***
  - ***Latency***
  - ***Scalability***
  - ***Reliability***

EU Mobile Cloud Networking Project: Network Functions as a Service

FP7 Integrated Project (2013-2016) targeted to bringing cloud computing features to mobile operator core networks (e.g., EPCaaS):

- Virtualization of components
- Software defined networking
- Elasticity
- Infrastructure sharing
- Redefining roaming



First lesson learnt: sufficient quality levels?

In the last years, growing industrial interest in Mobile Cloud Networking (MCN) as the opportunity to exploit the ***cloud computing paradigm through Network Function Virtualization (NFV)***

- primarily with the goal to reduce CAPEX/OPEX for future mobile networks deployment and operation

Risk/skepticism:

a virtualized infrastructure could not reach ***the levels of service reliability, availability, and quality usual for mobile telcos***

EU MCN project – http://www.mobile-cloud-networking.eu

First lesson learnt: sufficient quality levels?

EU MCN project – http://www.mobile-cloud-networking.eu

Large experimental campaigns and results from ***wide-scale industrial testbeds*** have demonstrated that it is possible via the adoption of advanced techniques for:

- ***lazy coordination*** of distributed cloud resources
- ***standardized*** virtualization of network functions
- ***proactive mobility-aware resource management***, including load balancing, handovers, …
- ***interoperable orchestration*** of ***infrastructure+service*** components

Motivations: Why NFV is needed?

Source: www.cse.wustl.edu

① **Virtualization**: use network resource without worrying about where it is physically located, how much it is, how it is organized, etc

② **Orchestration & Automation**: configuration through complied global policies versus the current manual translation and per device download

③ **Programmability** & **Openness**: modular design allows evolvability and customization to own choices

④ **Dynamic Scaling**

⑤ **Visibility**: Monitor resources, connectivity

⑥ **Performance**: Optimize network device utilization

⑦ **Multi-tenancy**: Should be able to serve new business models

⑧ **Service Integration**: seamlessly integrating interdependent services

ETSI Network Functions Virtualization (NFV)

The objective of NFV is to translate the classic network appliances to software modules

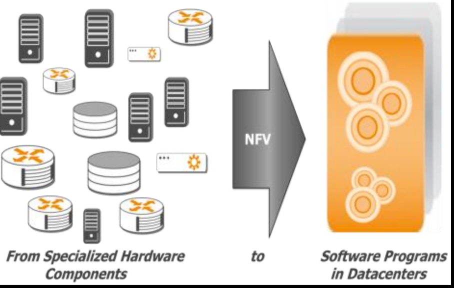
- Running on high volume servers with high volume storage
- Interconnected by generic high volume switches
- Automatically orchestrated and remotely installed

NFV is a novel paradigm that presumes that the network functions

- Are implemented only as software (programs)
- Can run on top of common servers

NFV has to fix the following main issues:

- ***Performance***
- ***Co-existence, portability, and interoperability***
- ***Automation***
- ***Scalability***



71

NFV and SDN as the support technologies for 5G

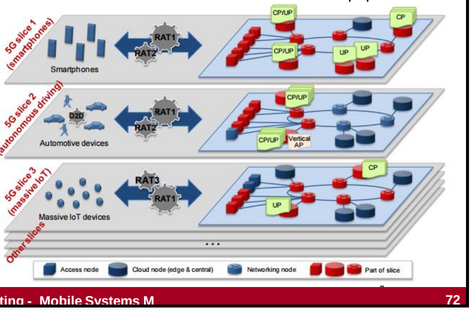
5G will be based on ***network slices*** *on top of same infrastructure*

NFV and SDN as the main enablers for:

- ***business agility*** – with its capabilities for on-demand, fast deployments
- ***network adaptability and flexibility*** – requires redesign of network functions (to cloud native), support for functions variance, flexible function allocation, etc.
- ***composition*** – putting multiple services together in a slice – end-to-end managemen
- ***slicing*** – separation at network level
- ***programmability*** – software-only network functions and their interaction with physical systems

➔ ***Orchestration*** is the cornerstone for all of these features

Source: NGMN Alliance - 5G whitepaper

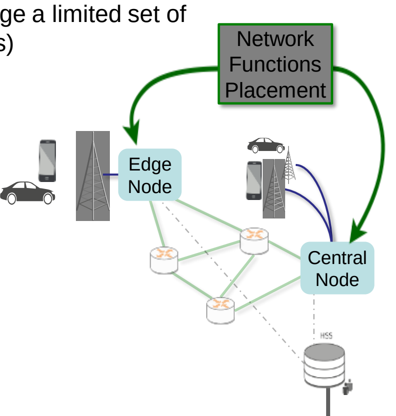


72

Network Function Placement Problem

Through ***edge cloud computing***:

- Network functions can be deployed in both ***edge nodes and central node***
- Edge controller has to be very simple to manage a limited set of devices (energy efficiency, compute limitations)



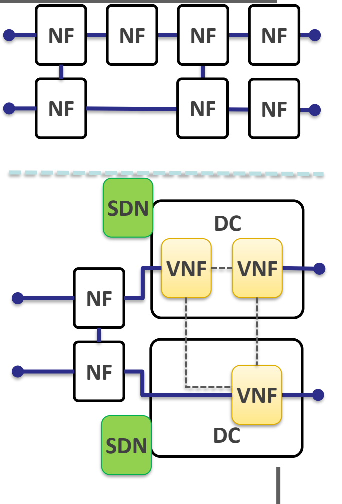
- ***Dynamic decisions*** about where to execute functionalities, depending on
  - ❑ state of subscribers
  - ❑ network congestion
  - ❑ single device/group) mobility pattern
- ***Autonomic functioning of edge nodes*** when no backhaul is available / backhaul communication is interrupted
- Policy-based functioning of edge networking for making decisions when edge routing is used

NFV and SDN

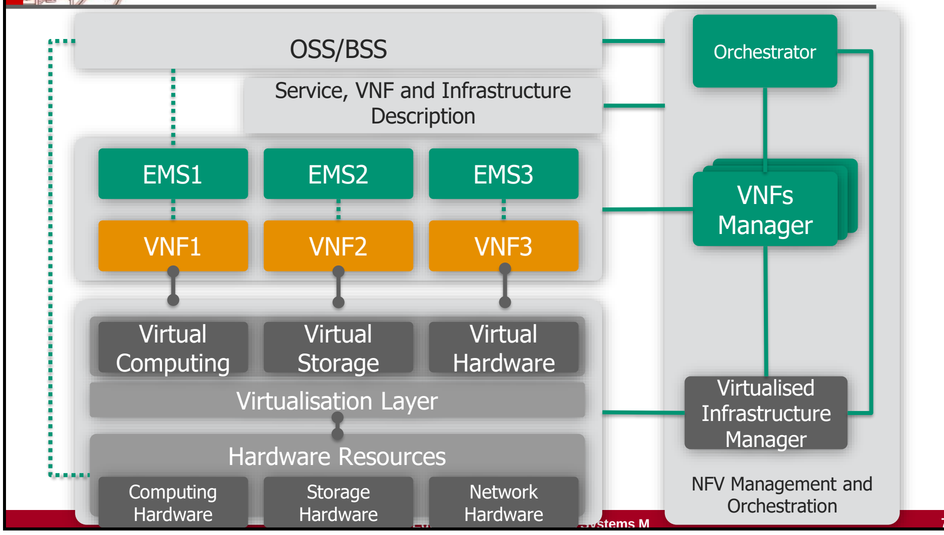
- NFV requires network functions to be implemented *as* ***software on top of common hardware***
- SDN brings ***remote programmability*** of the network
- NFV/SDN platform acts as an end-to-end middleware between:
  - ❑ A distributed heterogeneous infrastructure for compute and storage
  - ❑ Interconnected through a controlled network
  - ❑ Generic network functions implemented in software running in isolated containers/virtual machines
    - VPNs, NATs, DNSs, IMSs, EPCs, Application Servers, etc.

The main value added differentiator between different solutions is the *quality of the software*

- how well it can solve the specific service needs

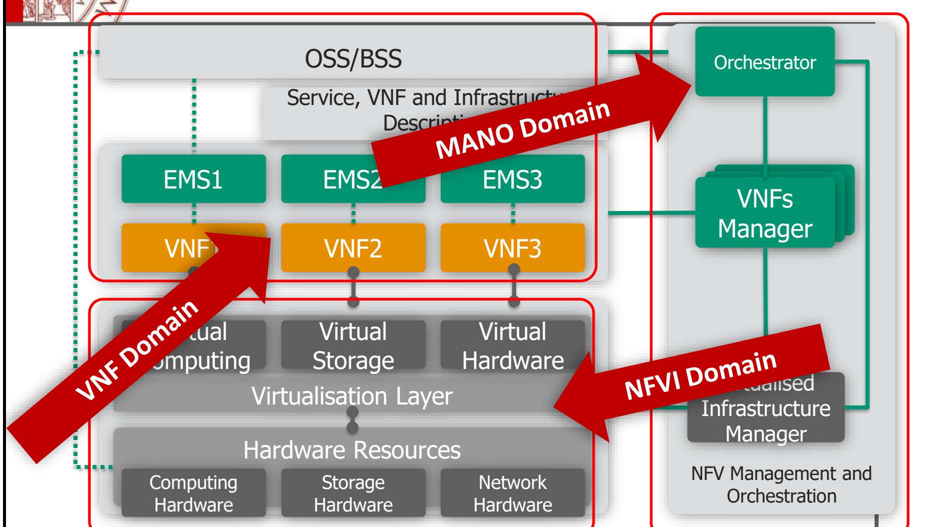


NFV Architecture Blue print is ready since Nov. 2012…



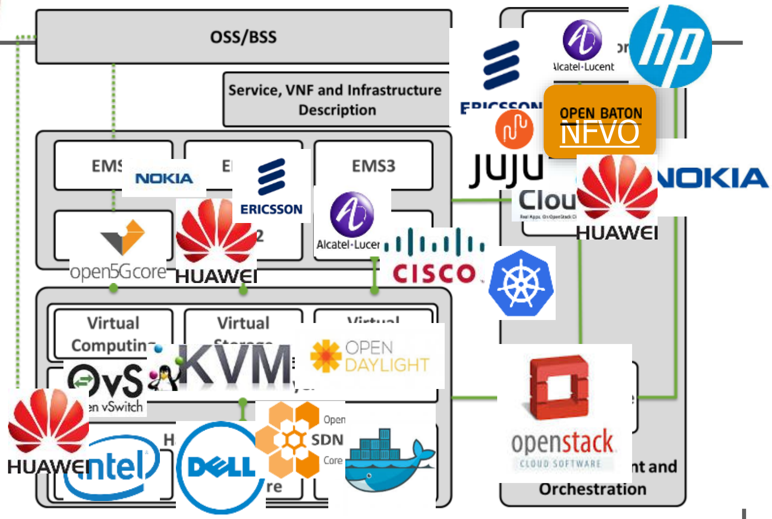
75

NFV Architecture Blue print is ready since Nov. 2012…



76

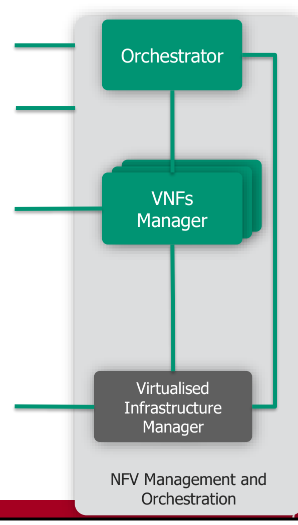
The NFV Ecosystem



77

Virtualized Infrastructure Manager (VIM)

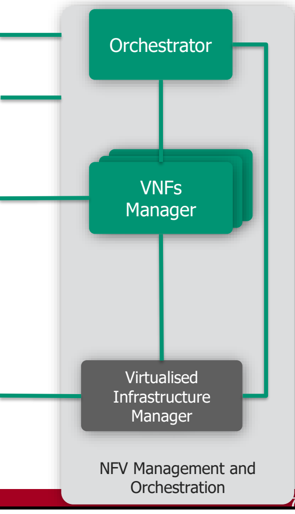
- Responsible for the ***lifecycle management*** of compute, storage and network resources provided by the NFVI
- It is basically a ***Cloud Management System*** which exposes an API for standard CRUD operations on those resources
- ***OpenStack*** is the ***de facto standard implementation*** of this functional block



78

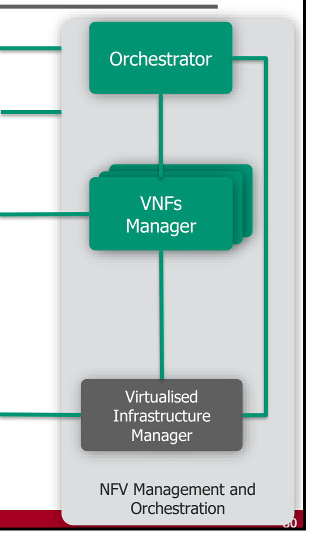
VNF Manager (VNFM)

- Responsible for the lifecycle management of Virtual Network Function instances
  - ❑One per NF
  - ❑One per multiple VNF instances even of different type
- It has to support the:
  - ❑VNF instantiation
  - ❑VNF configuration
  - ❑VNF update
  - ❑VNF scaling in / out
  - ❑VNF instance termination



Network Function Virtualization Orchestration (NFVO)

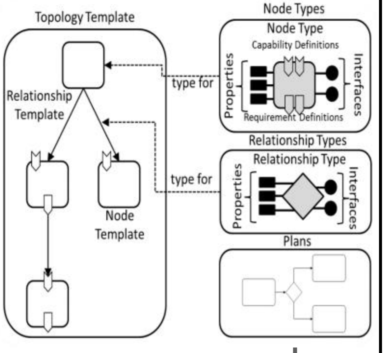
- Responsible for the ***lifecycle management of Network Services***:
  - ❑In a single domain
  - ❑Over multiple datacenters
- Applies policies for resource utilization
- Requests the instantiation of VNFs via the VNF Managers



OASIS TOSCA

Topology and Orchestration Specification for Cloud Applications (TOSCA)

- The ***OASIS TOSCA Technical Committee*** works to enhance the portability of cloud applications and services
- TOSCA will enable the ***interoperable description of application and infrastructure cloud services***, the relationships between parts of the service, and the operational behavior of these services (e.g., deploy, patch, shutdown) - independent of the supplier creating the service, and any particular cloud provider or hosting technology
- TOSCA will also make it possible for higher-level operational behavior to be associated with cloud infrastructure management



81

A comprehensive MANO orchestrator was (still) missing…

Two approaches in regard to orchestration were taken:

1) Orchestrating from the infrastructure perspective

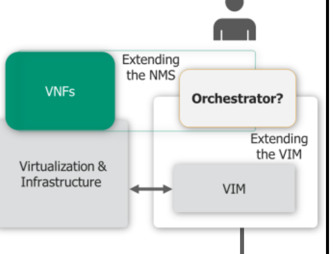
Extending VIM towards service orchestration. Missing:

- Adaptation to complex network services requirements, e.g. fault management, scaling, network function placement, virtual network configuration, information flow paths, security, reliability

2) Orchestrating from the network service perspective

Extending the Network Management System to handle orchestration. Missing:

- Capitalize through native components on cloud opportunities: scaling, dynamic resource allocation
- Define the appropriate network service KPIs, end-to-end fault management, end-to-end reliability insurance, etc.



82

What is OpenBaton?

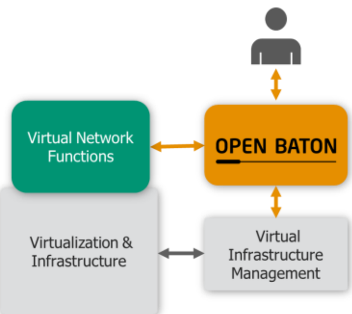
OpenBaton is Open Source implementation of the ETSI MANO specification

OpenBaton aims to foster, within the NFV framework, the integration between:

- ***Virtual Network Function providers***
- ***Cloud Infrastructure providers***

Functionality:

- Installation, deployment, and config. network services
- Runs on top of multi-site OpenStack
- Provides independent infrastructure slices
- Support for generic or specific VNF management



Designed for answering R&D requirements

- Easy to configure and to deploy
- Providing a centralized view of the testbed

github: https://github.com/openbaton

83

What OpenBaton stands for

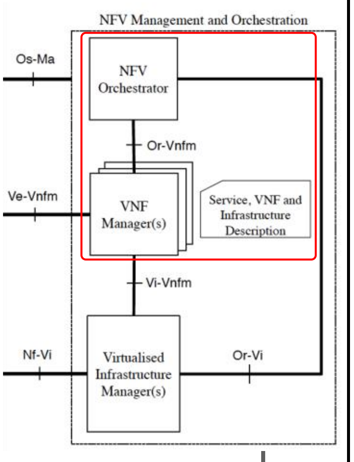
- ***No vendor lock-in***: OpenBaton does not contain any vendor specific features. It follows open specifications and it is open to the community
- Built from scratch following the ***ETSI MANO specification***
  - ❑The NFVO uses the ETSI NFV data model internally for the definition of the Network Service and Virtual Network Descriptors
- ***Allows interoperability***
  - ❑Being interoperable is one of the challenges brought by the fragmented ecosystem in the management and orchestration area. It requires a lot of work to make two different vendors solution working together → need of a single vendor-independent platform

84

OpenBaton & more

OpenBaton is based on the ETSI NFV MANO v1.1.1 (2014-12) specification. It provides:

- A ***NFV Orchestrator*** managing the lifecycle of Network Service Descriptors (NSD) and interfacing with one or more VNF Manager(s) (VNFM)
- A ***generic VNF Manager***, which can be easily extended for supporting different type of VNFs
- A ***set of libraries*** which could be used for building your own VNFMs (vnfm-sdk)
- A ***dashboard*** for easily managing all the VNFs



Nowadays, also other open solutions such as:

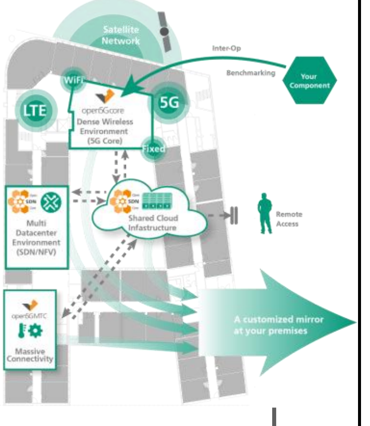
Open Source MANO (OSM)

→ **https://osm.etsi.org/**

The Fraunhofer FOKUS 5G Playground

5G Playground: A comprehensive testbed environment for prototyping 5G-ready VNFs using OpenBaton orchestration

- Open5GCore providing the next wireless system beyond LTE/EPC with more efficient communication for the subscribers and improved automation/reliability (applying SDN and NFV principles)
- Open5GMTC enabling connectivity management and end-to-end service establishment for a huge number of connected devices
- OpenSDNCore enabling SDN experimentation for data path, backhaul networks or customized network environments



- All those are software components and can be customized, deployed and configured on demand via OpenBaton enabling automatic just-in-time test environment creation, experimentation and demonstration

To conclude:
Open Research Directions (1)

- ***Fog-enabled federated management*** - efficiently deploying and managing federations of dense inter-connected and decentralized cloud infrastructures, by dynamically moving (partial) MCN functions to the edge of the network by taking local decisions and optimizations
- ***Edge computing for extremely high availability*** - How to exploit mobile edge computing towards disaster resilient and emergency robust MCN solutions? How should it be efficiently combined with DC networking virtualization?
- ***Scalability and quality for data-intensive applications*** - Effective and efficient solutions for scale, quality, and privacy/security, in particular in data-intensive applications deployed over federated environments, such as in the case of MCN for smart cities or wide-scale IoT with dominant M2M communications

87

To conclude:
Open Research Directions (2)

- Efficient MEC solutions for industrial IoT ***Machine Learning*** – Local execution of ***partly learned models*** (cloud-based learning), ***federated learning***, online local refinement of partly learned models, cloud notification and update for offline model refinement only when needed, …
- ***State, state, state…*** - efficient state migration, replication, eventual consistency, proactive state management, etc
- Etc etc…

88

To conclude: Open Innovation Challenges for **Industrial Exploitation**

About immediate industrial applicability of solutions in the field, in several sub-areas with specific performance/functional constraints we are far from ready-to-deploy frameworks:

- ***high-availability by design***, in particular in the case of federated infrastructures
- ***cost-efficient scalability***
- ***QoS differentiation*** with reasonable guarantees under dynamically changing (in both time and space) load profiles
- Prototyping and demonstrating ***wide-scale pilots*** that show the advantages of edge computing techniques in "hard" application scenarios, such as ***federated mobile public safety networks***, with specific challenges in terms of reliability and privacy

Conclusions?

Still a lot of ***research & innovation work to complete*** to make edge computing solutions applicable in ***different application domains*** (e.g., machine learning for predictive diagnostics, online process quality optimization in manufacturing, …) and ***economically sustainable to leverage new business models*** (e.g., need for portable orchestration solutions for federated environments, especially container-based)

Opportunities for both academia & industries

Some Primary References (1)

- ETSI's Mobile Edge Computing initiative https://portal.etsi.org/Portals/0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_Paper_V1%2018-09-14.pdf
- IEEE Future Networks Initiative, https://futurenetworks.ieee.org/
- Open Fog Consortium, https://www.openfogconsortium.org/
- Open Edge Computing (OEC) open source project, http://openedgecomputing.org/
- Cisco IOx, https://developer.cisco.com/site/iox/
- M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," IEEE journal on Pervasive Computing, vol.8, no.4, pp.14-23, Oct.-Dec. 2009

Some Primary References (2)

- F. Bonomi, R. Milito, J. Zhu and S. Addepalli, "Fog Computing and Its Role in the Internet of Things,"Proceedings of the first edition of the MCC workshop on Mobile cloud computing (MCC 2012), ACM, August 2012
- S. Singh, Yen-Chang Chiu, Yi-Hsing Tsai, Jen-Shun Yang, "Mobile Edge Fog Computing in 5G Era: Architecture and Implementation", Computer Symposium (ICS), 2016
- M. Tao, K. Ota, M. Dong, "Foud: Integrating Fog and Cloud for 5G-Enabled V2G Networks," IEEE Network, Vol. 31, No. 2, March/April 2017
- D. Keren, G. Sagy, A. Abboud, D. Ben-David, A. Schuster, I. Sharfman, A. Deligiannakis, «Geometric Monitoring of Heterogeneous Streams», IEEE Transactions on Knowledge and Data Engineering, Vol. 26, No. 8, 2014
- D. Keren, I. Sharfman, A. Schuster, A. Livne, «Shape Sensitive Geometric Monitoring», IEEE Transactions on Knowledge and Data Engineering, Vol. 24, No. 8, 2012

Some Secondary ☺ References (3)

- P. Bellavista, J. Berrocal, A. Corradi, S.K. Das, L. Foschini, A. Zanni, «A survey on fog computing for the Internet of Things», Elsever PMC,accepted, 2019
- P. Bellavista, C. Giannelli, T. Lagkas, P.G. Sarigiannidis, «Quality Management of Surveillance Multimedia Streams Via Federated SDN Controllers in Fiwi-Iot Integrated Deployment Environments», IEEE Access, 2018
- P. Bellavista, A. Corradi, L. Foschini, S. Monti, «Improved Adaptation and Survivability via Dynamic Service Composition of Ubiquitous Computing Middleware», IEEE Access, 2018
- P. Bellavista, S. Chessa, L. Foschini, L. Gioia, M. Girolami, «Human-Enabled Edge Computing: Exploiting the Crowd as a Dynamic Extension of Mobile Edge Computing», IEEE Communications Magazine, 2018
- L. Tartarini, M. Gerla, P. Bellavista, et al., «Software-defined handover decision engine for heterogeneous cloud radio access networks», Computer Comm., 2018

Some Secondary ☺ References (4)

- R. Venanzi, B. Kantarci, L. Foschini, P. Bellavista, «MQTT-Driven Sustainable Node Discovery for Internet of Things-Fog Environments», IEEE ICC, 2018
- L. Scalzotto, K.E. Benson, G. Bouloukakis, P. Bellavista, V. Issarny, S. Mehrotra, N. Venkatasubramanian, «An Implementation Experience with SDN-enabled IoT Data Exchange Middleware», ACM Middleware (Posters), 2018
- A. Zanni, S. Forsström, U. Jennehag, P. Bellavista, «Elastic Provisioning of Internet of Things Services Using Fog Computing: An Experience Report», IEEE Mobile Cloud, 2018
- R. Venanzi, B. Kantarci, L. Foschini, P. Bellavista, «MQTT-Driven Node Discovery for Integrated IoT-Fog Settings Revisited: The Impact of Advertiser Dynamicity», IEEE SOSE, 2018
- L. Acquaviva, P. Bellavista, F. Bosi, A. Corradi, L. Foschini, S. Monti, A. Sabbioni, «NoMISHAP: A Novel Middleware Support for High Availability in Multicloud PaaS», IEEE Cloud Computing, 2017

Some Secondary ☺ References (5)

- J. De Benedetto, P. Bellavista, L. Foschini, «Proximity discovery and data dissemination for mobile crowd sensing using LTE direct», Computer Networks, 2017
- P. Bellavista, A. Corradi, A. Zanni, «Integrating mobile internet of things and cloud computing towards scalability: lessons learned from existing fog computing architectures and solutions», IJCC, 2017
- H. Kim, H. Oh, P. Bellavista, J. Ben-Othman, «Constructing event-driven partial barriers with resilience in wireless mobile sensor networks», J. Network and Computer Applications, 2017
- P. Bellavista, C. Giannelli, S. Lanzone, G. Riberto, C. Stefanelli, M. Tortonesi, «A Middleware Solution for Wireless IoT Applications in Sparse Smart Cities», Sensors, 2017
- A. Zanni, S.-Y. Yu, P. Bellavista, R. Langar, S. Secci, «Automated selection of offloadable tasks for mobile computation offloading in edge computing», CNSM, 2017

Some Secondary ☺ References (6)

- P. Bellavista, L. Foschini, D. Scotece, «Converging Mobile Edge Computing, Fog Computing, and IoT Quality Requirements», FiCloud, 2017
- A. Zanni, S.-Y. Yu, S. Secci, R. Langar, P. Bellavista, D.F. Macedo, «Automated offloading of android applications for computation/energy optimizations», IEEE INFOCOM Workshops, 2017
- I. Aljawarneh, P. Bellavista, A. Corradi, R. Montanari, L. Foschini, A. Zanotti, «Efficient spark-based framework for big geospatial data query processing and analysis», IEEE ISCC, 2017
- P. Bellavista, A. Zanni, M. Solimando, «A migration-enhanced edge computing support for mobile devices in hostile environments», IWCMC, 2017
- G.A. Carella, M. Pauls, T. Magedanz, M. Cilloni, P. Bellavista, L. Foschini, «Prototyping nfv-based multi-access edge computing in 5G ready networks with open baton», NetSoft, 2017

Some Secondary ☺ References (7)

- A. Edmonds, G. Carella, F. Zarrar Yousaf, C. Goncalves, T.M. Bohnert, T. Metsch, P. Bellavista, L. Foschini, «An OCCI-compliant framework for fine-grained resource-aware management in Mobile Cloud Networking», IEEE ISCC, 2016
- P. Bellavista, A. Zanni, «Scalability of Kura-extended Gateways via MQTT-CoAP Integration and Hierarchical Optimizations», Mobilware, Springer, 2016
- P. Bellavista, invited keynote speech, «Mobile Cloud Networking: Lessons Learnt, Open Research Directions, and Industrial Innovation Opportunities», IEEE MobileCloud, 2016
- E. Cau, M. Corici, P. Bellavista, L. Foschini, G. Carella, A. Edmonds, T.M. Bohnert, «Efficient Exploitation of Mobile Edge Computing for Virtualized 5G in EPC Architectures». IEEE MobileCloud, 2016
- P. Bellavista, A. Zanni, «Towards better scalability for IoT-cloud interactions via combined exploitation of MQTT and CoAP», IEEE RTSI, 2016
- … and several ☺ others under review…