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

Mobile Systems M course (8 ECTS)
II Term – Academic Year 2022/2023

07 (opt) – 5G and Mobile Edge Computing

Paolo Bellavista
paolo.bellavista@unibo.it

5G converged world

Voice (VoIP)

Audio/Video Conference

Chat and messagging

Video on Demand (VoD)

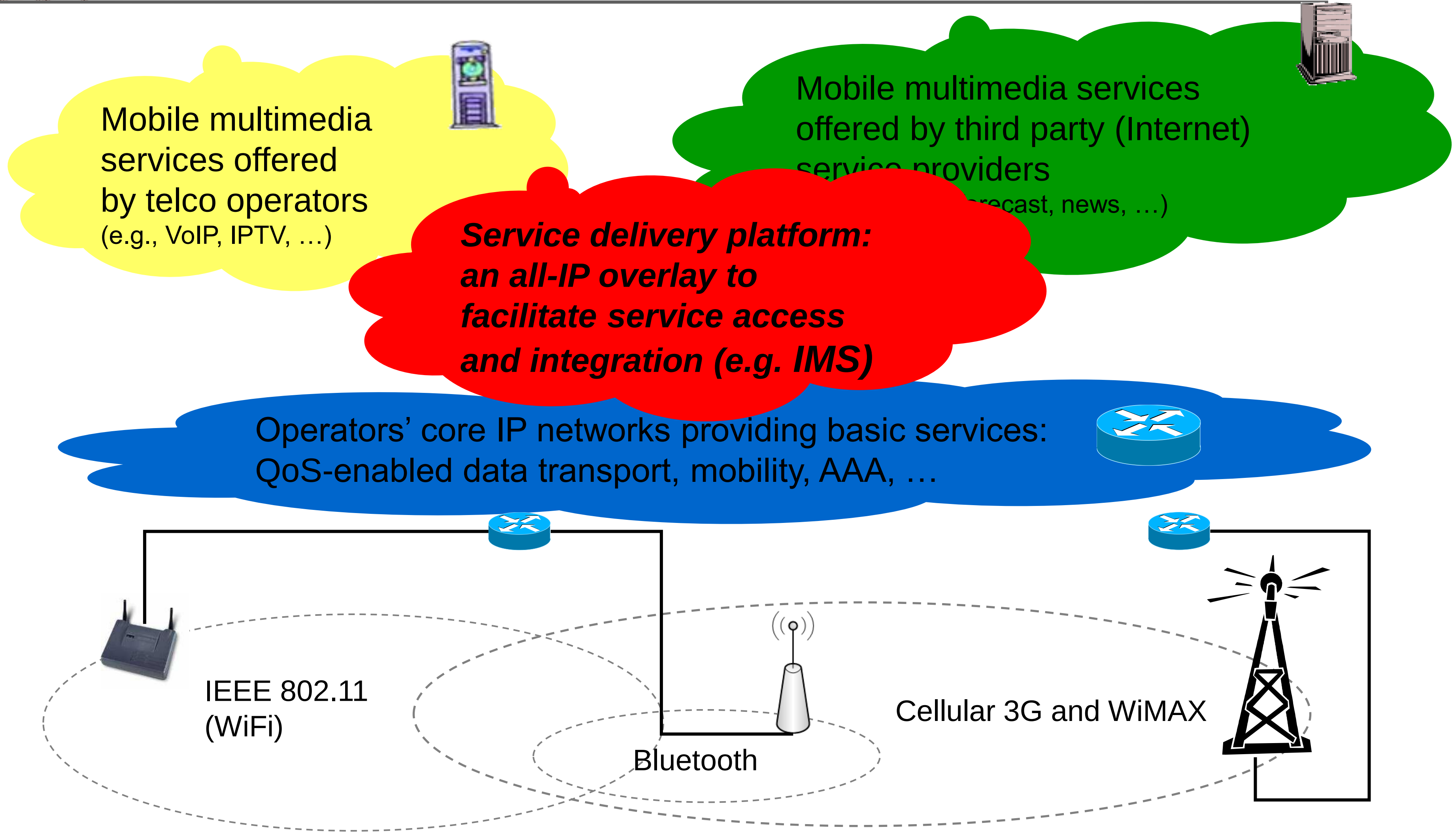
And many more…

- **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

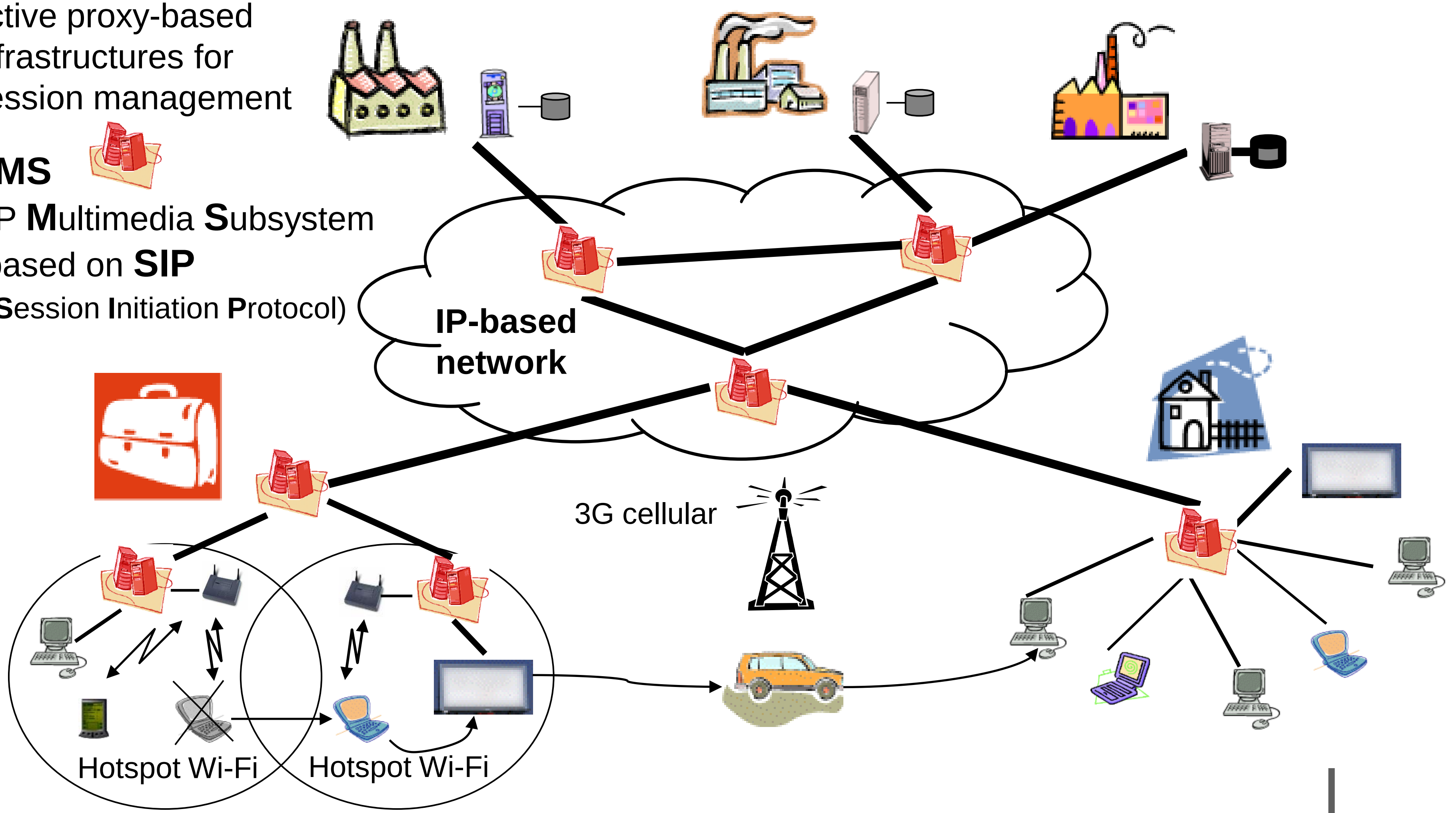
- Basic services – Call / session layer
  - Signalling entities
  - Databases for
  - Interworking with 2G/3G/4G…
- Value added services layer
  - Application servers
  - Media resources

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

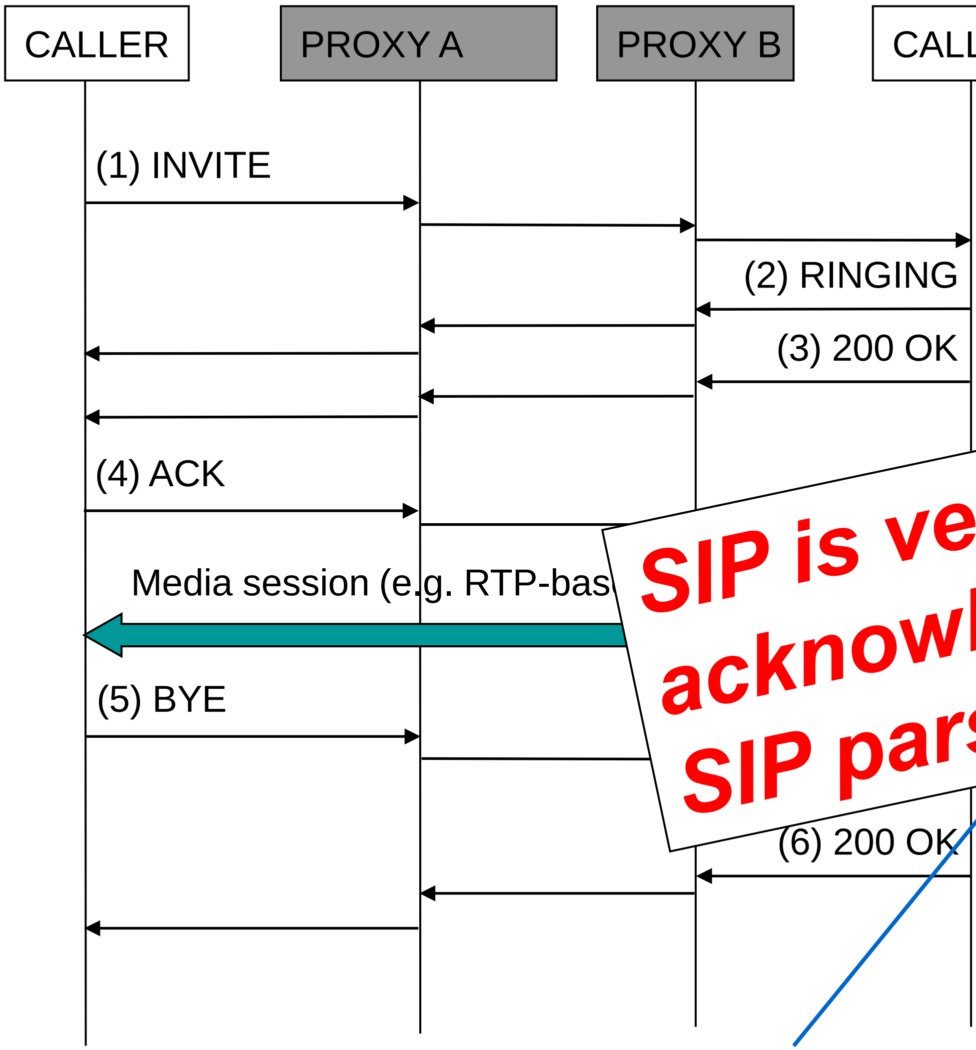
- SIP defines a ***signaling framework*** and related ***protocols and messages*** to setup ***any kind of session*** (work at the Open Systems Interconnection – OSI – ***session layer***)
  - SIP is very ***open*** and ***general purpose*** ☺
  - SIP includes several core facilities for ***mobility 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



Start line:
- ***request line*** (in requests)
- ***status line*** (in responses)

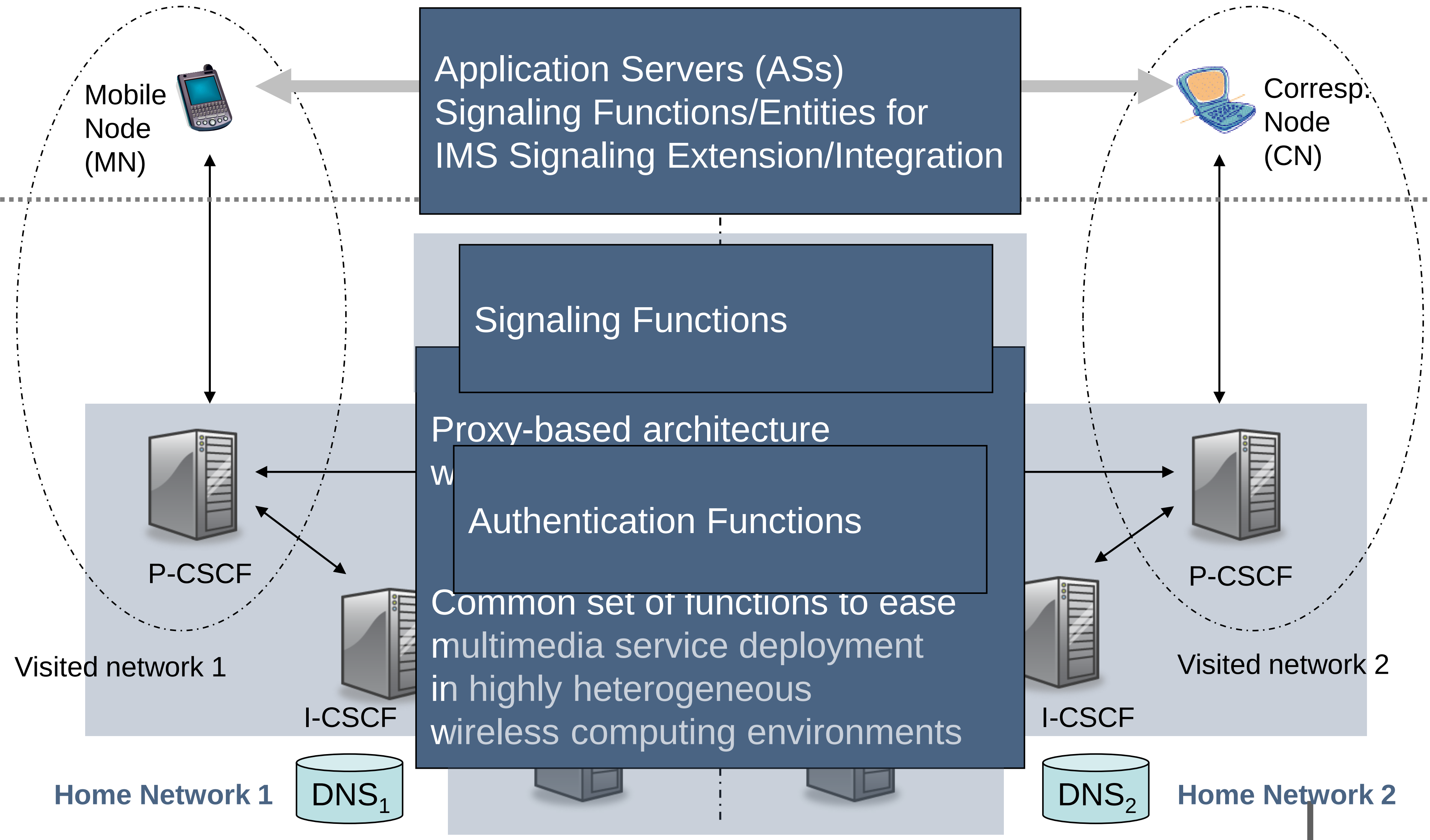
```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;
branch=z9hG4bK776asdhds
Max-Forwards: 70
To:
<sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
… SDP description …
```

SIP is very verbose and acknowledged; in addition, SIP parsing is CPU-intensive

Message body (optional): for example, an SDP description to negotiate audio/video codecs/formats

Header: with a number of ***header fields***

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*** ☺

IMS functional entities: AS

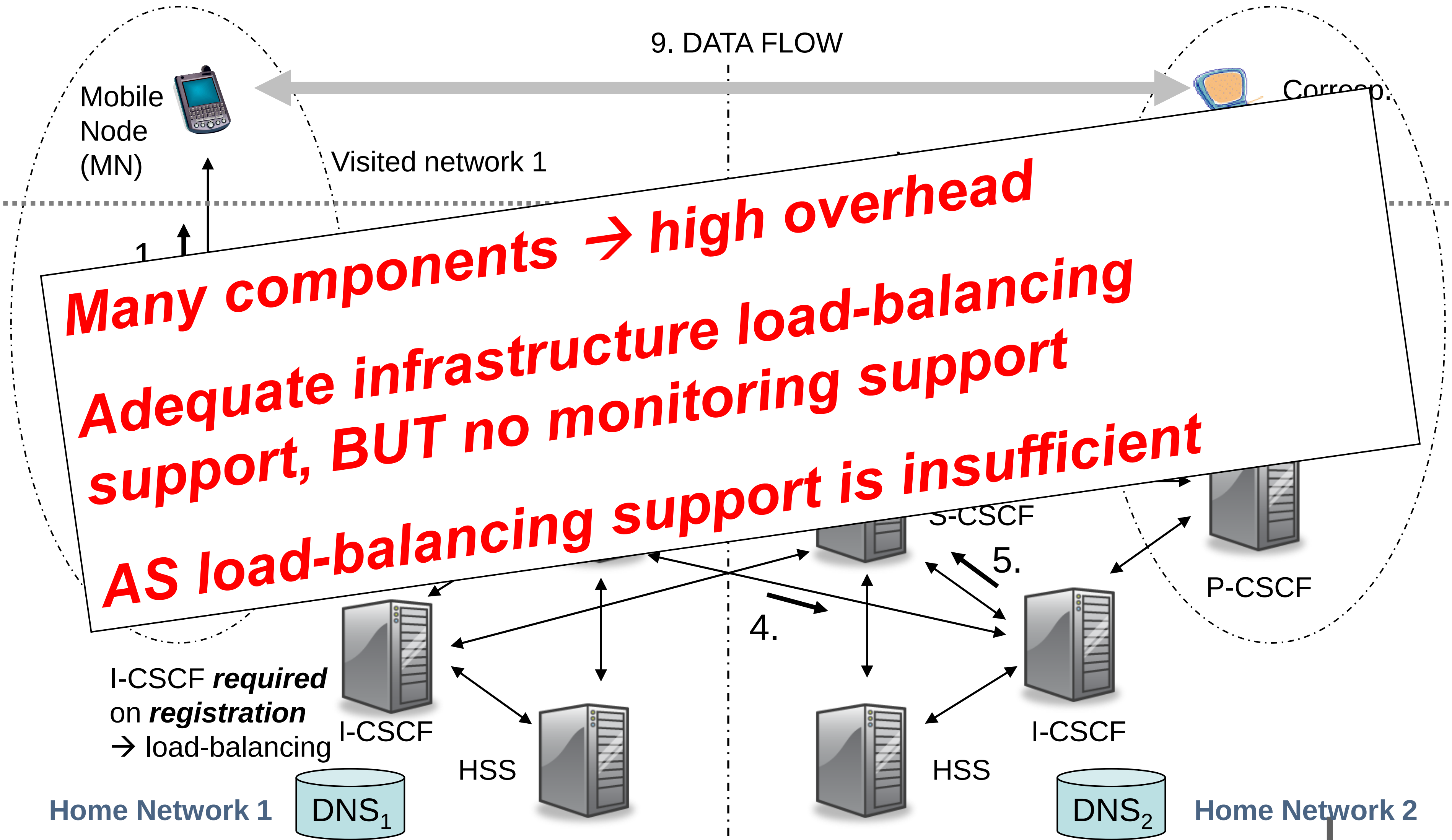
Application Server (***AS***):

- ***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), …

IMS in action for a VoIP call

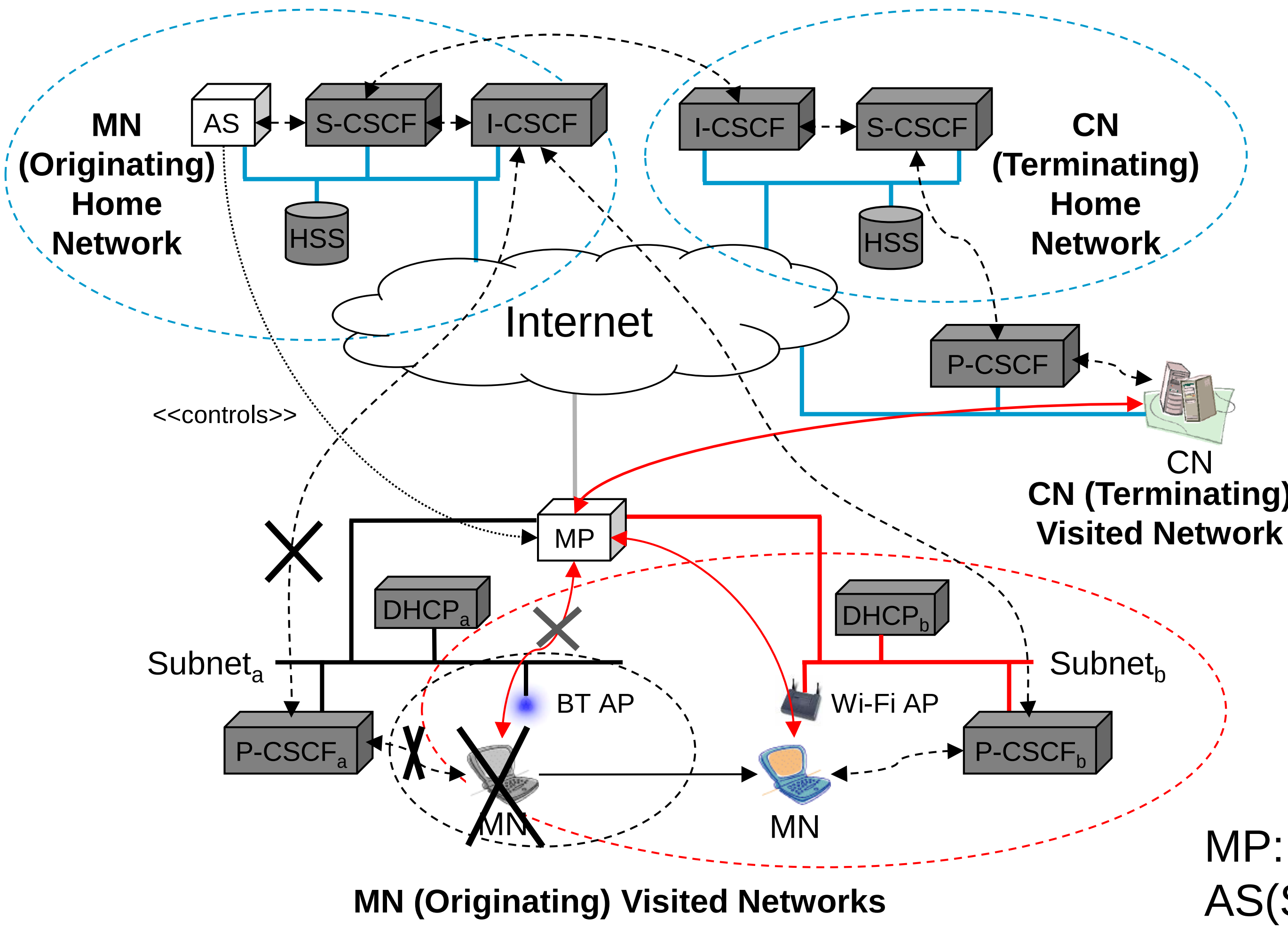


Many components → high overhead

Adequate infrastructure load-balancing support, BUT no monitoring support

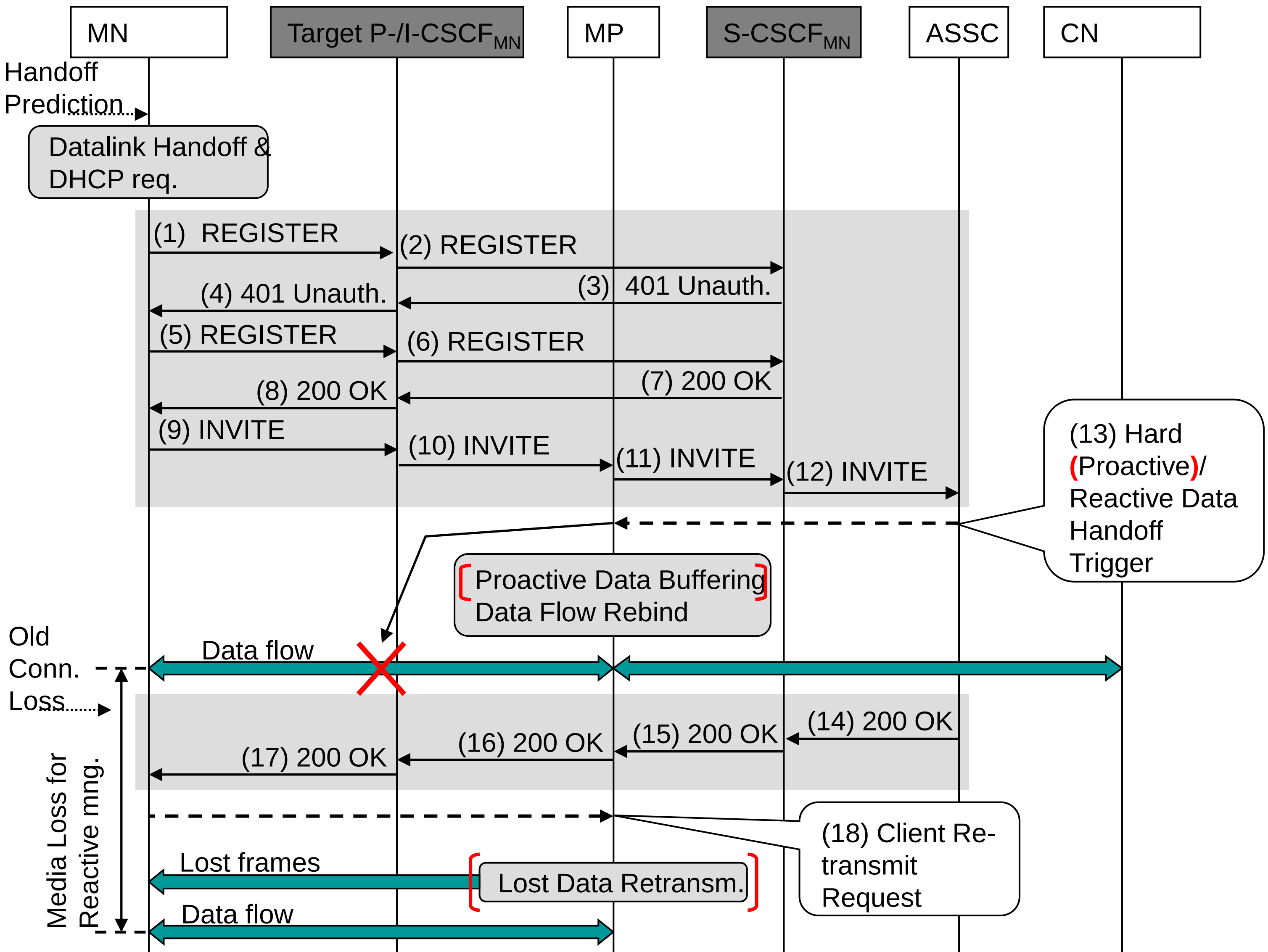
AS load-balancing support is insufficient

IMS Handoff Management

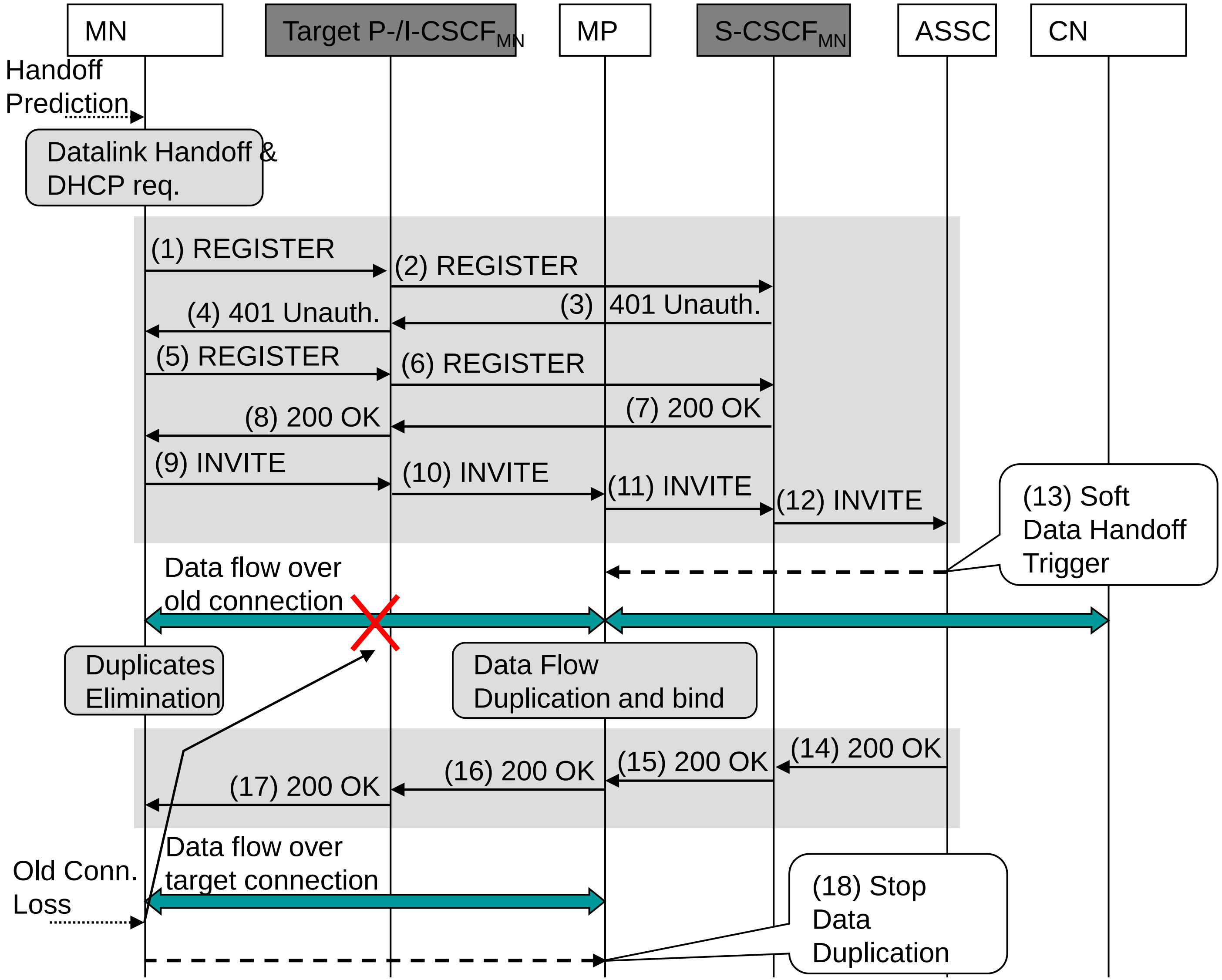


MP: Media Proxy
AS(SC): AS for Service Continuity

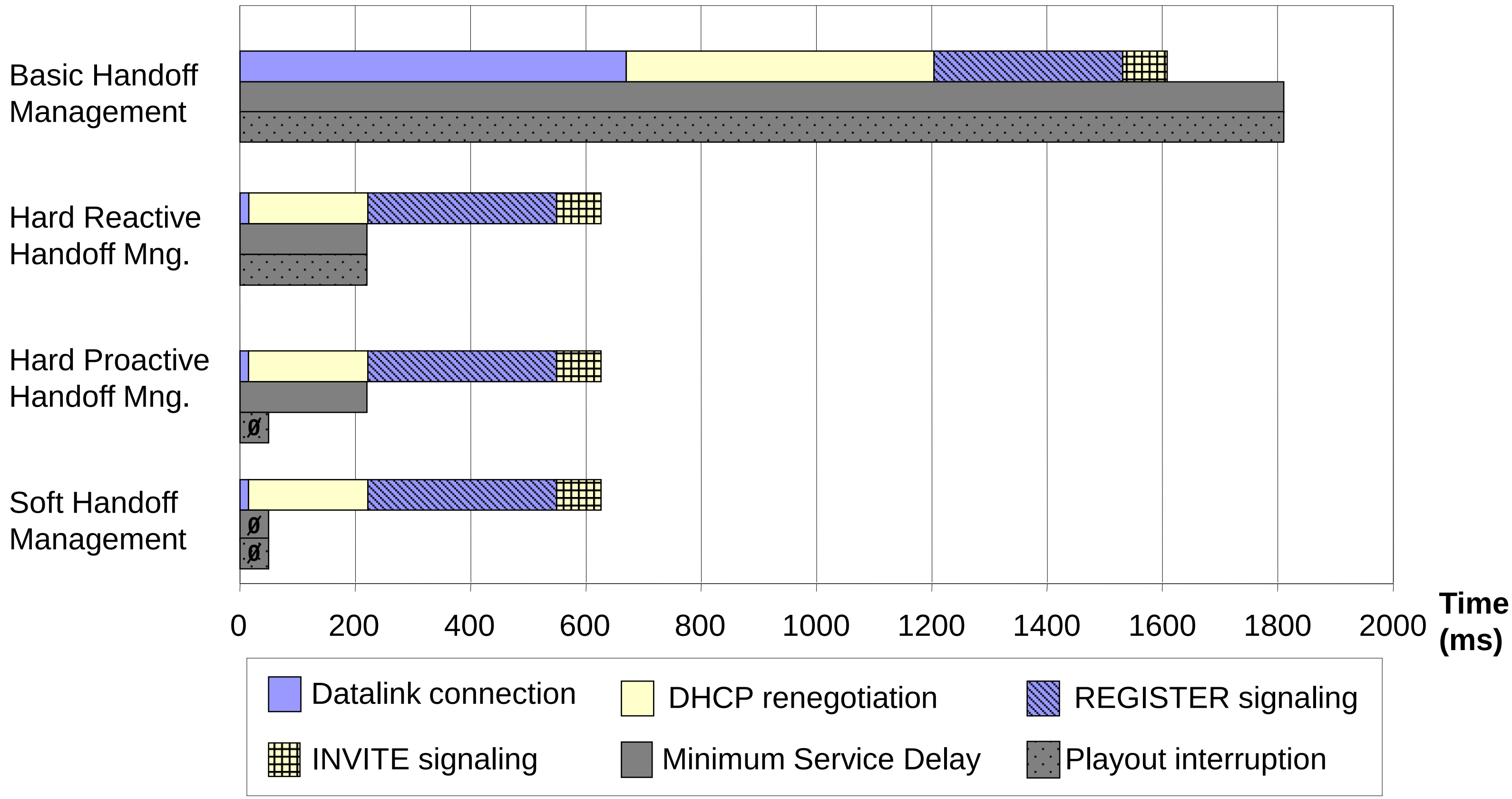
Optimized hard proactive/reactive IMS Handoff Management



Optimized soft IMS Handoff Management

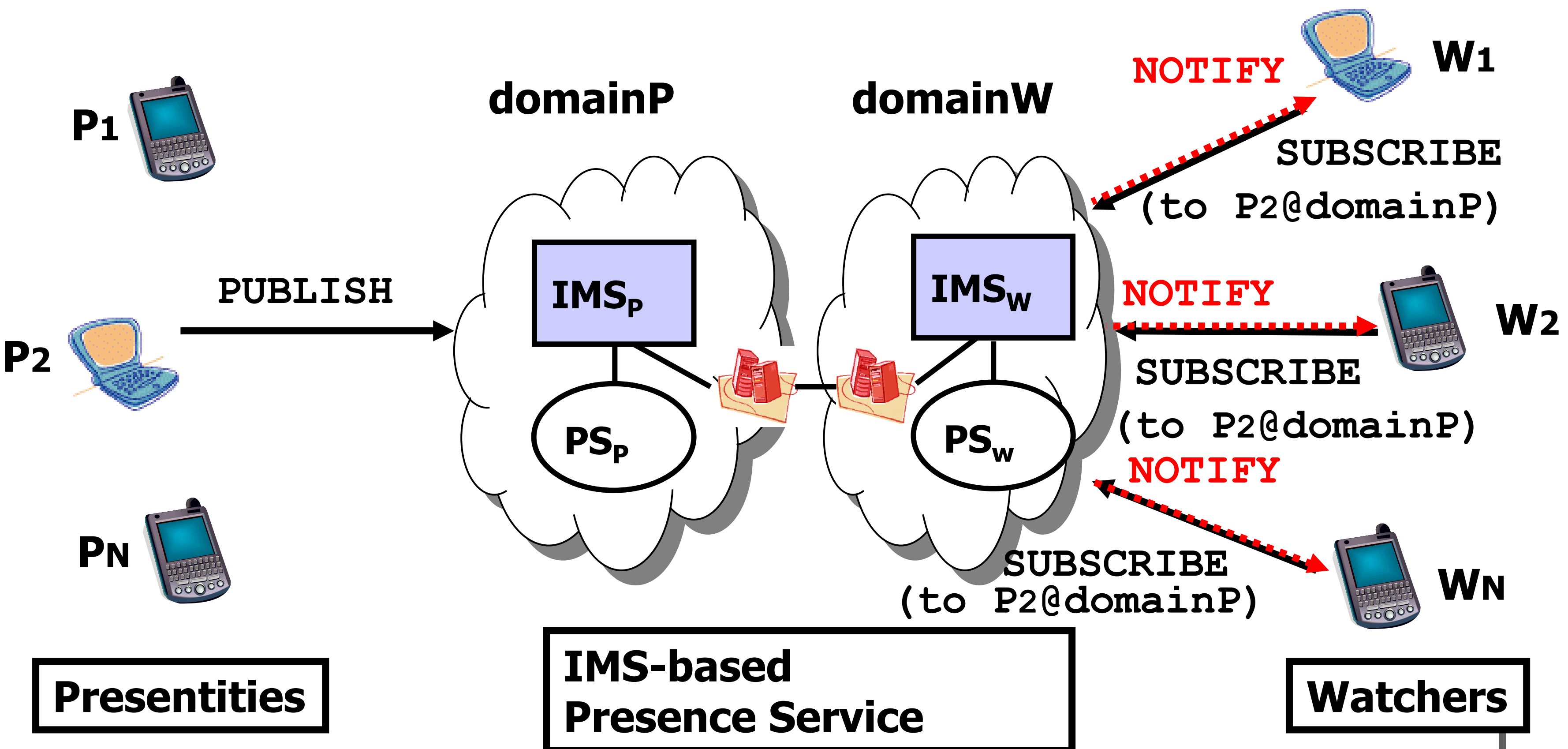


Advanced IMS Handoff Management: Some experimental results



Another example: IMS-based presence service

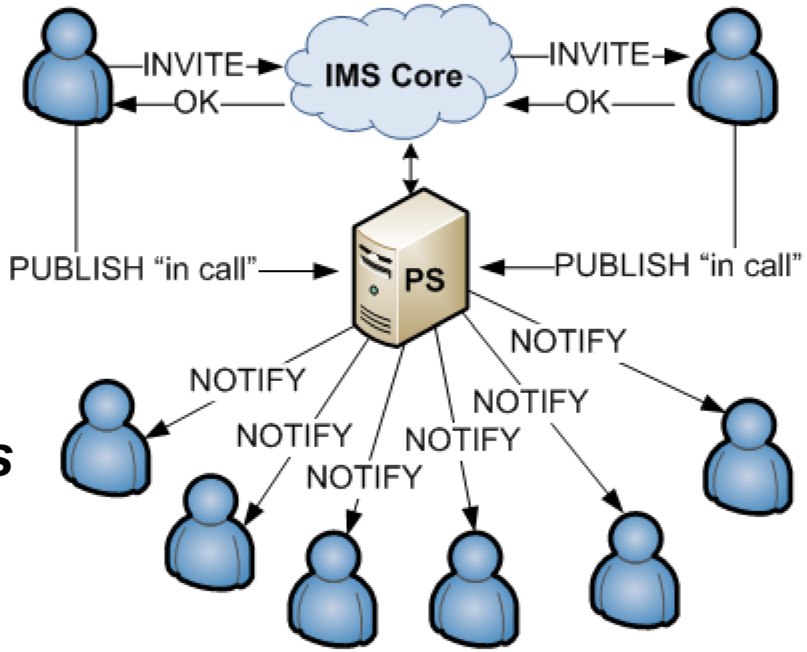
Presence service (PS) permits users and hw/sw components, called ***presentities (P_i),*** to convey their ability and willingness to communicate with subscribed ***watchers (W_j)***



Scalability issues at a glance

High mobility & context changes

New services VoIP+PS (call-status notification)



- 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***)

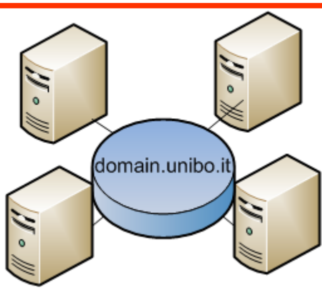
→ Need for a better understanding of IMS ***scalability shortcomings*** and ***load-balancing support*** both at ***infrastructure*** and ***service*** levels

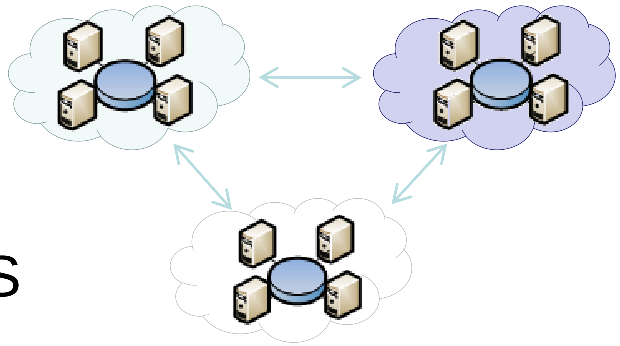
IMS scalability: (partial) solutions

- **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)***



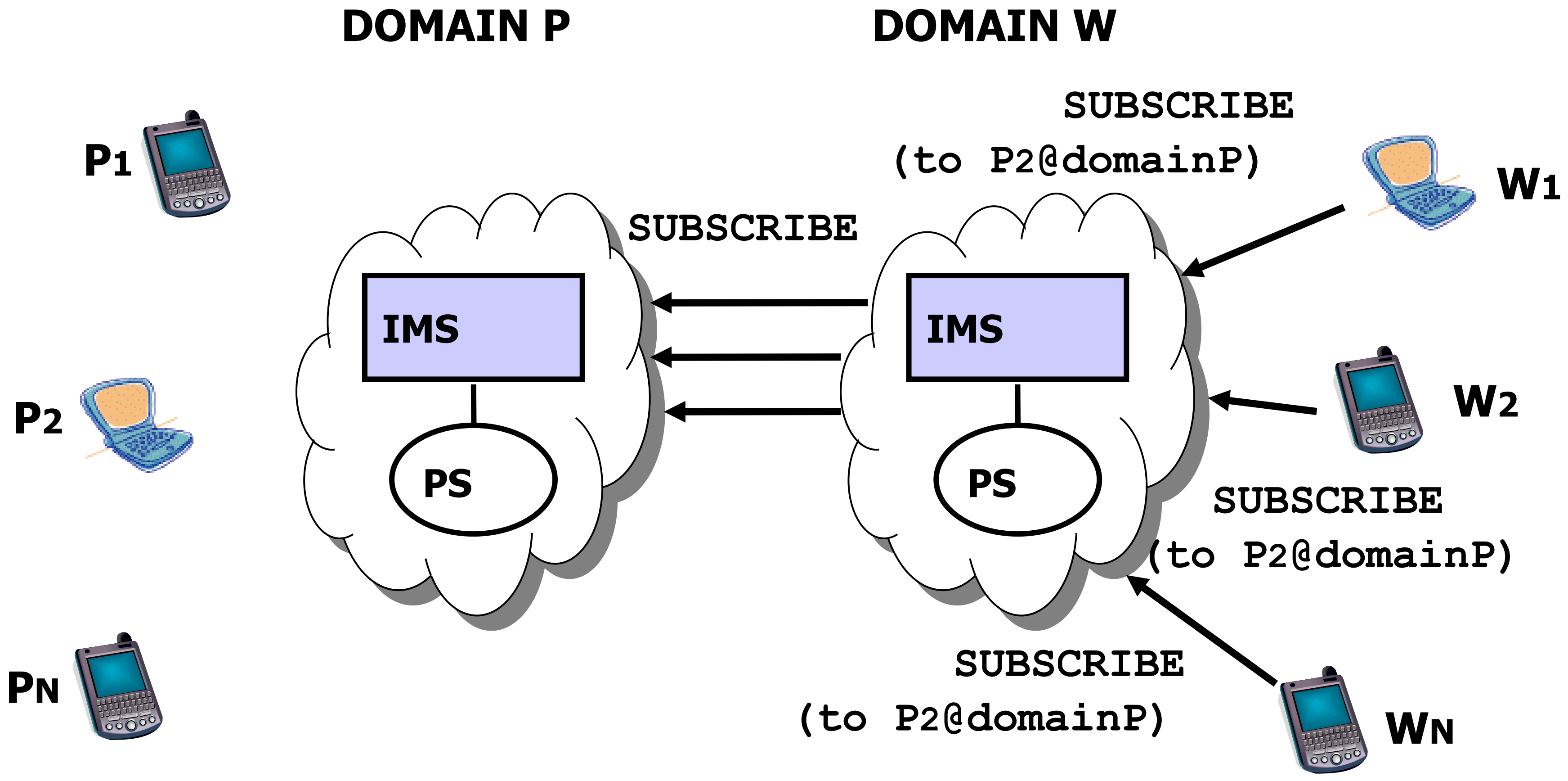


IMS PS scalability use case

P: Presentity **PS**: Presence Server

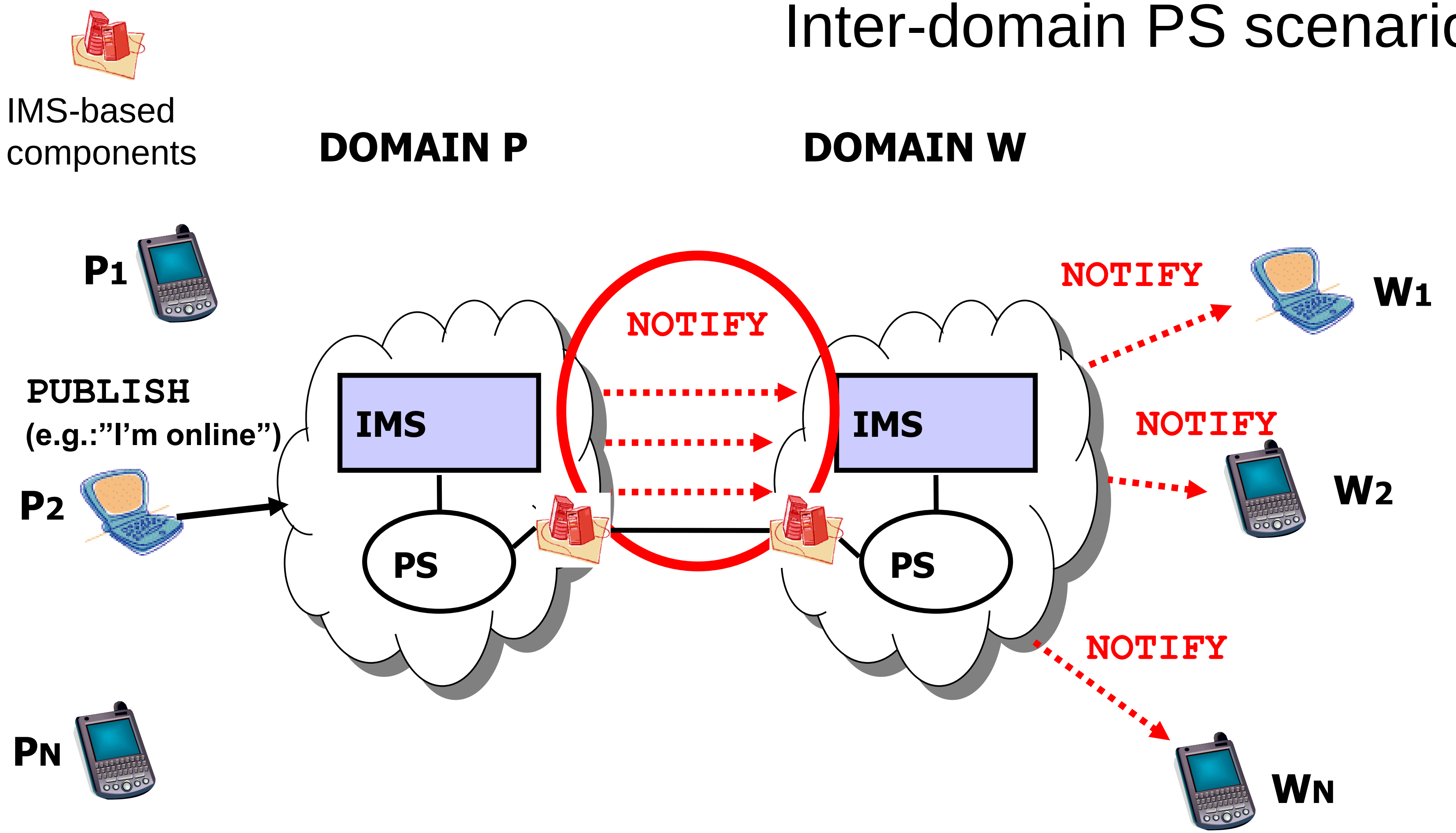
W: Watcher

Inter-domain PS scenario



IMS PS scalability use case

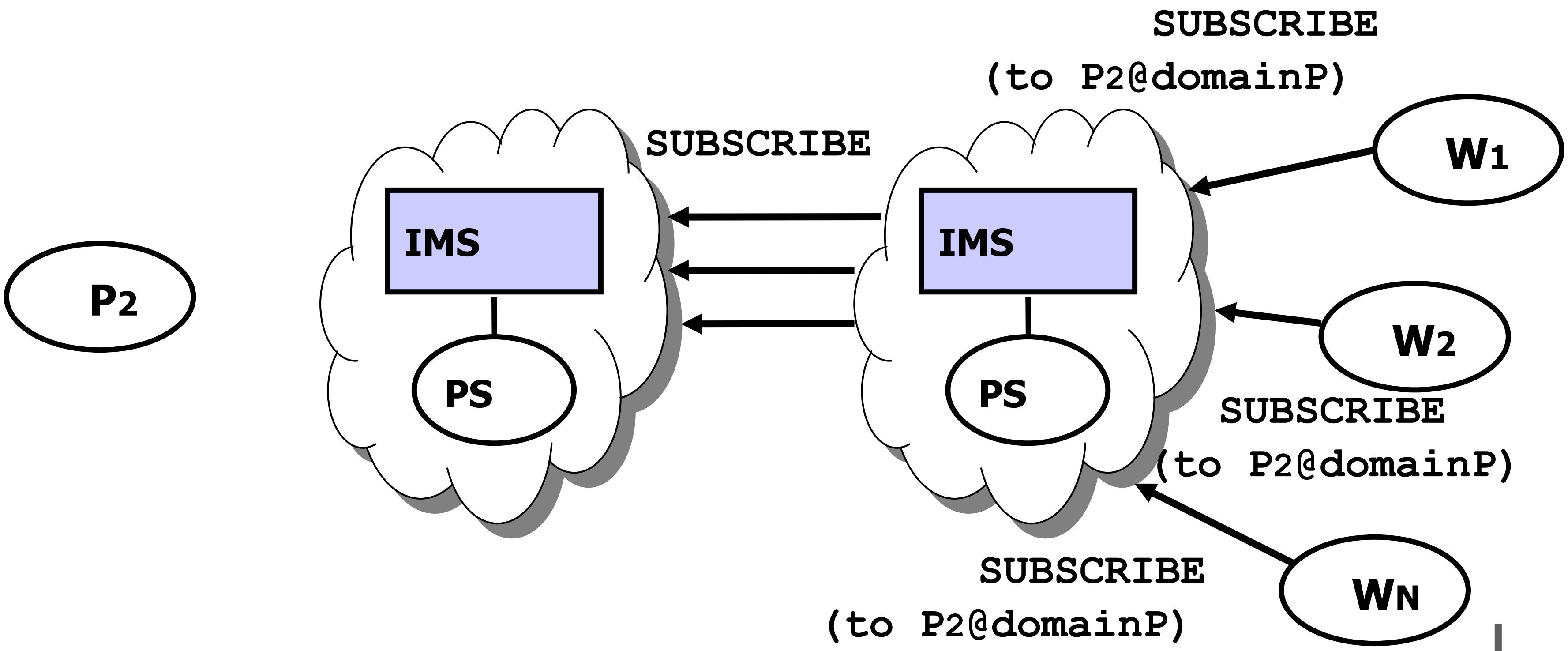
Inter-domain PS scenario



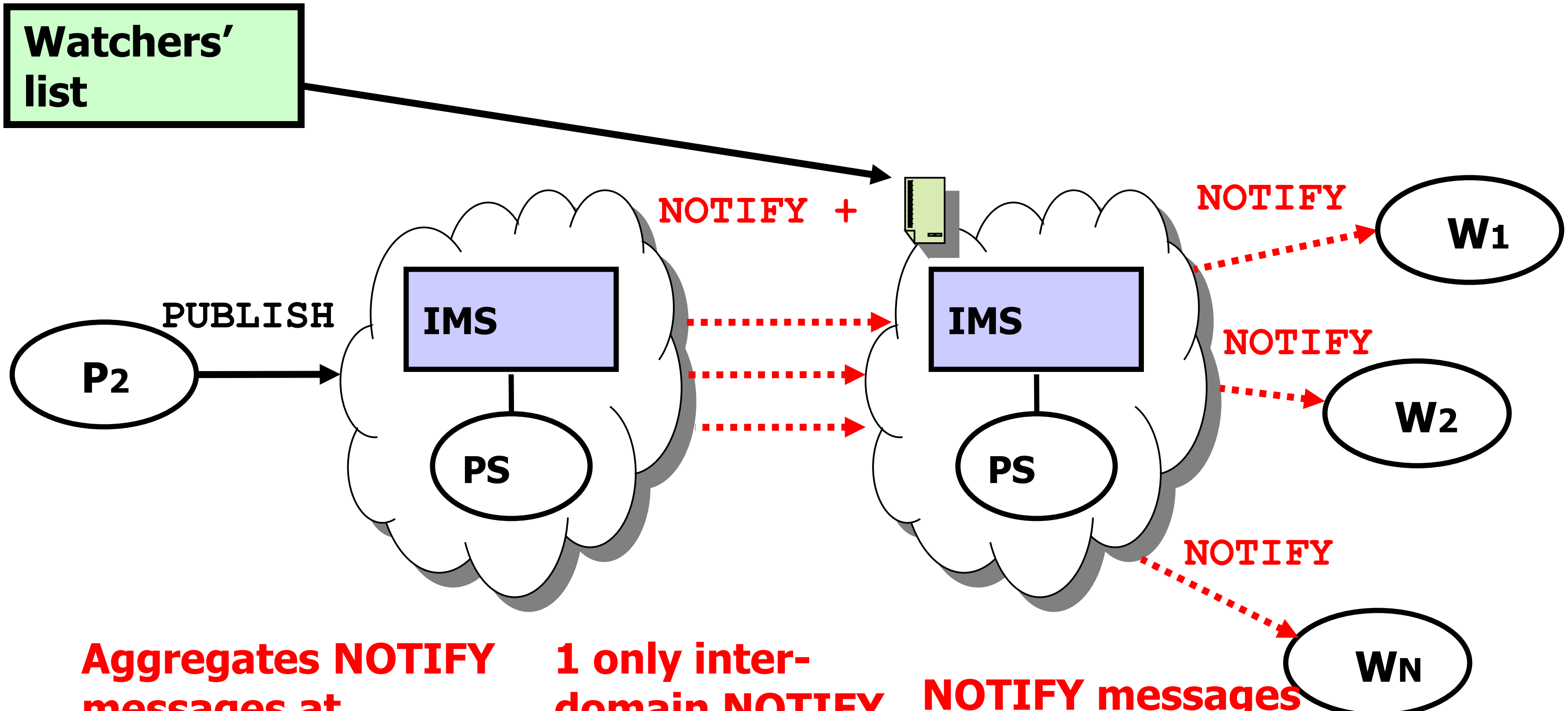
PS is very prone to scalability issues!!

Possible inter-domain optimizations: Common NOTIFY

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



Common NOTIFY



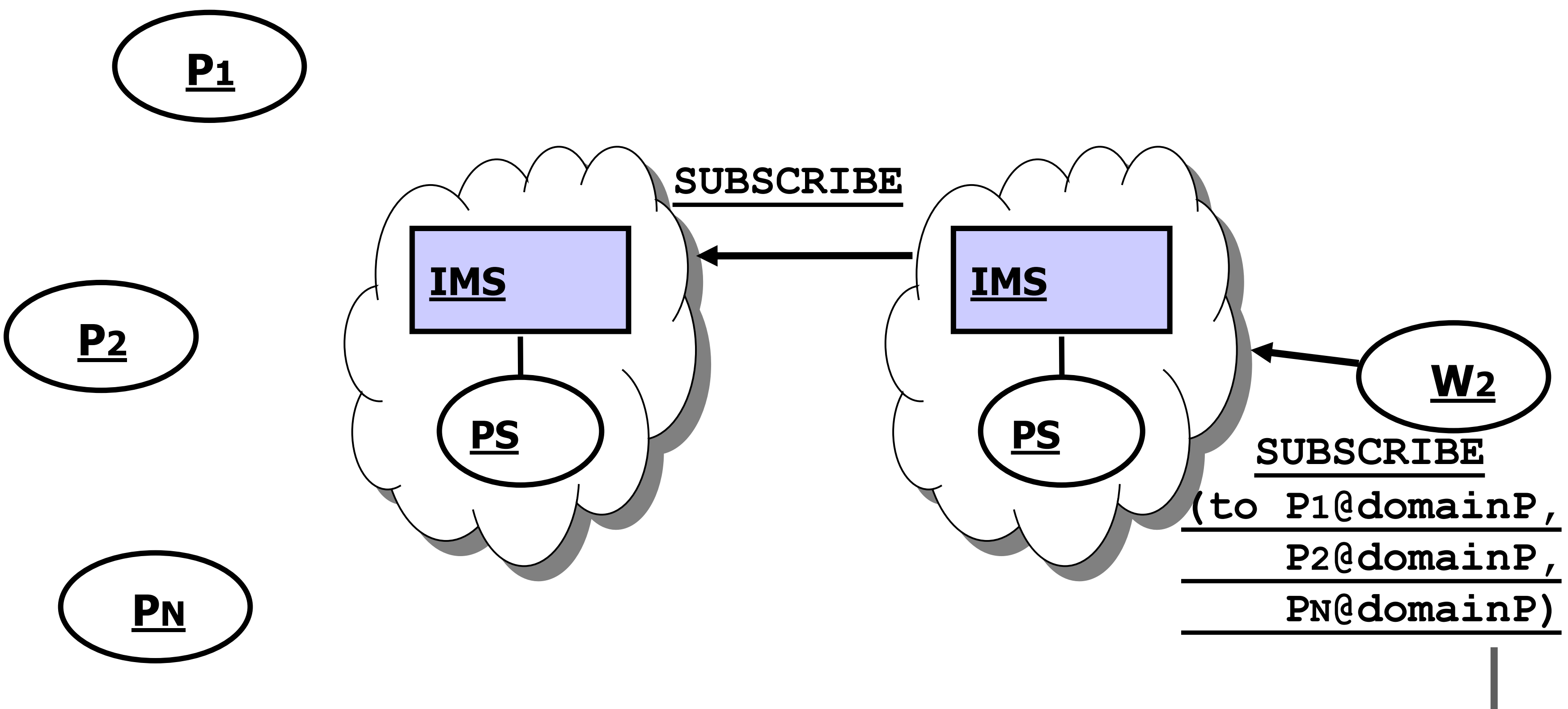
Aggregates NOTIFY messages at *Presentity's* domain

1 only inter-domain NOTIFY message

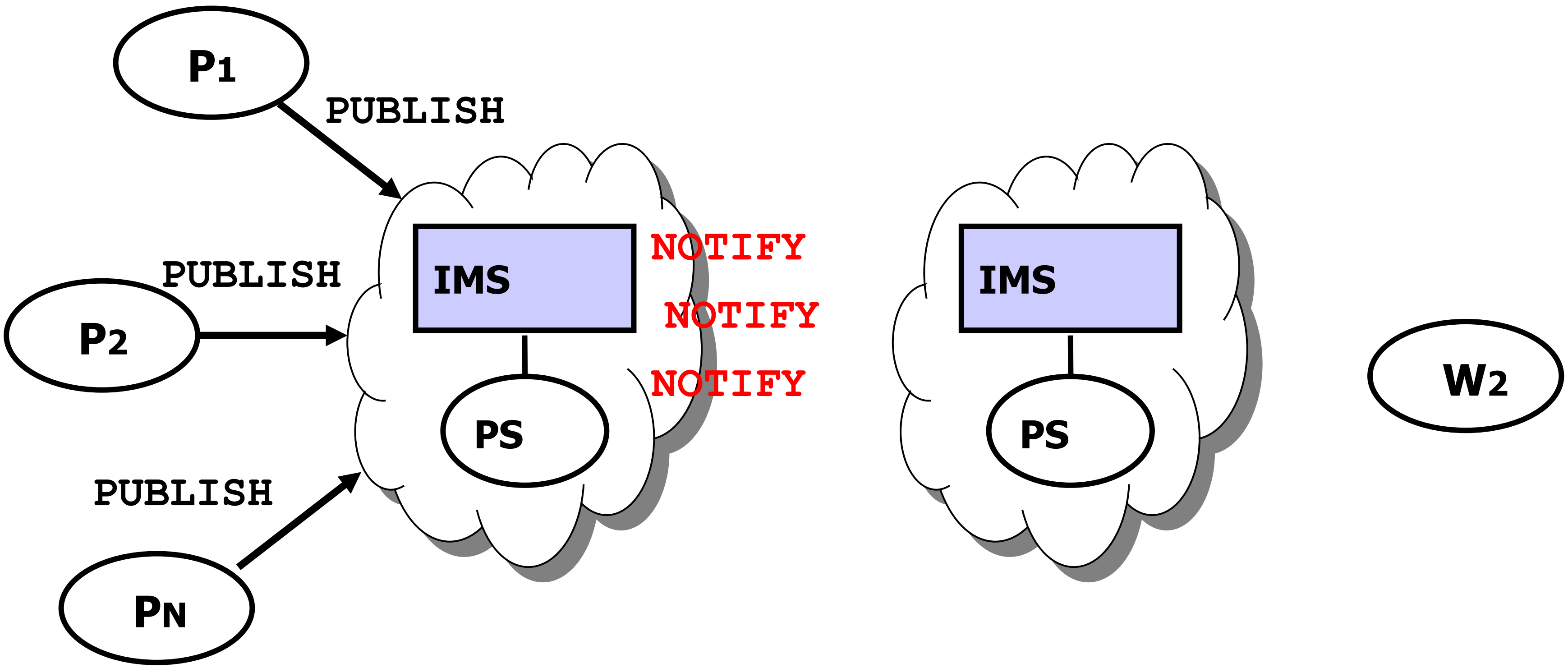
NOTIFY messages creation at Watchers's domain

Batched NOTIFY

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

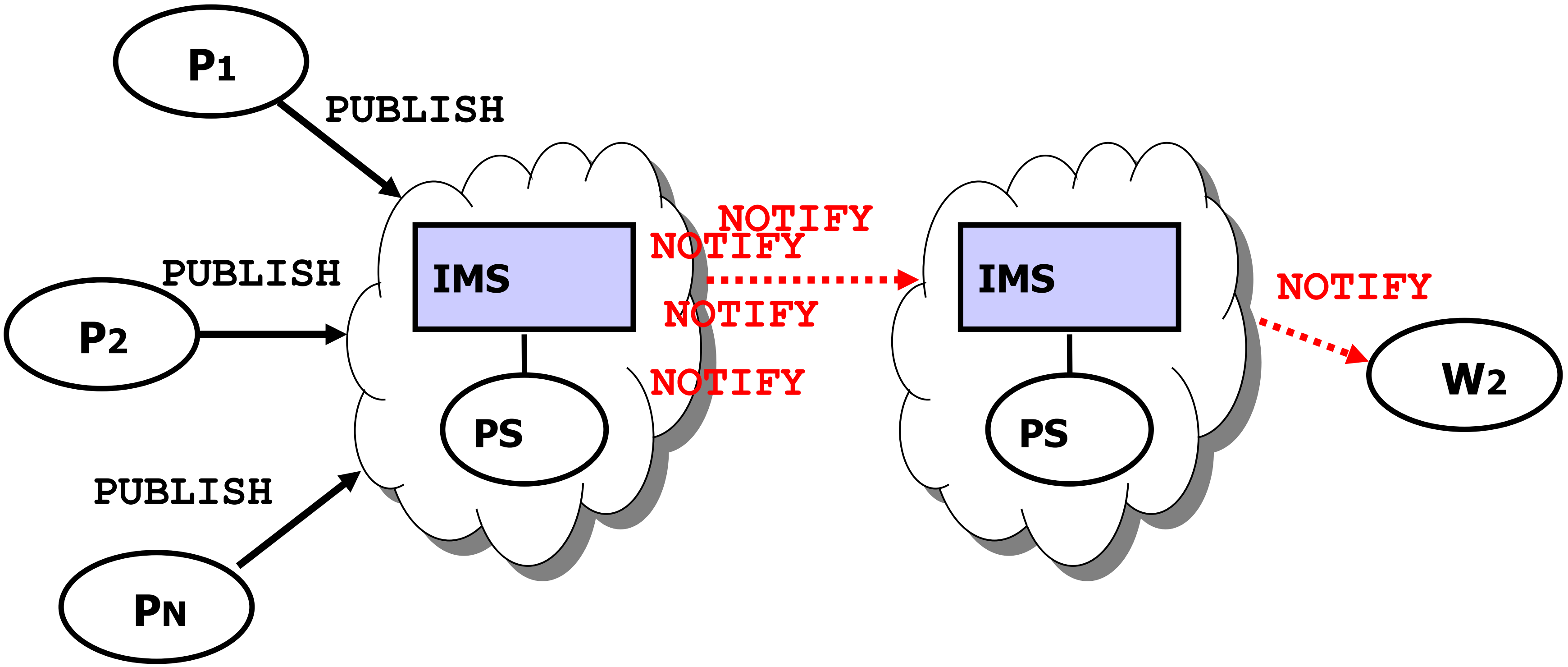


Batched NOTIFY



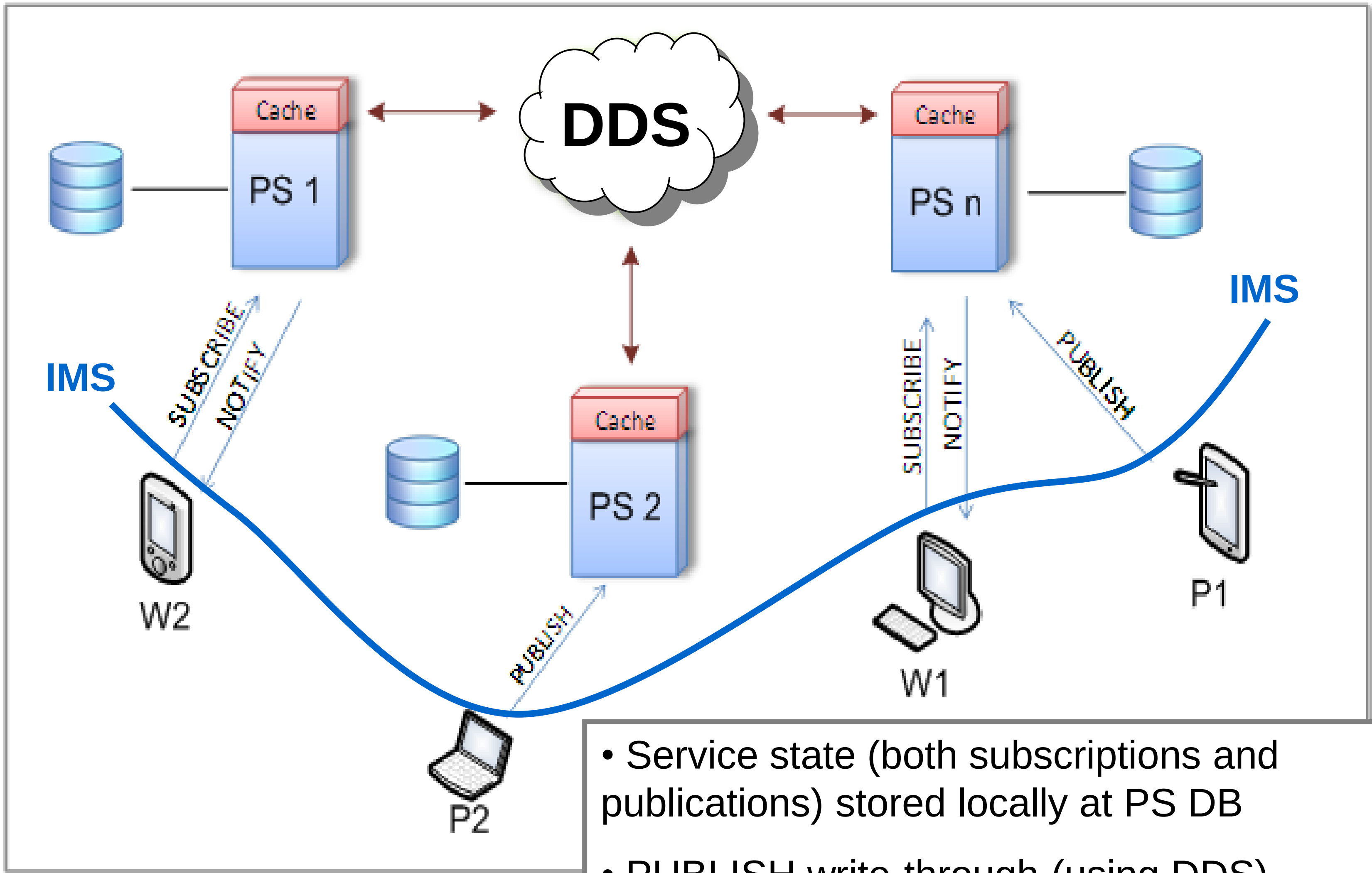
Time-based (periodic) NOTIFY message batching

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, …

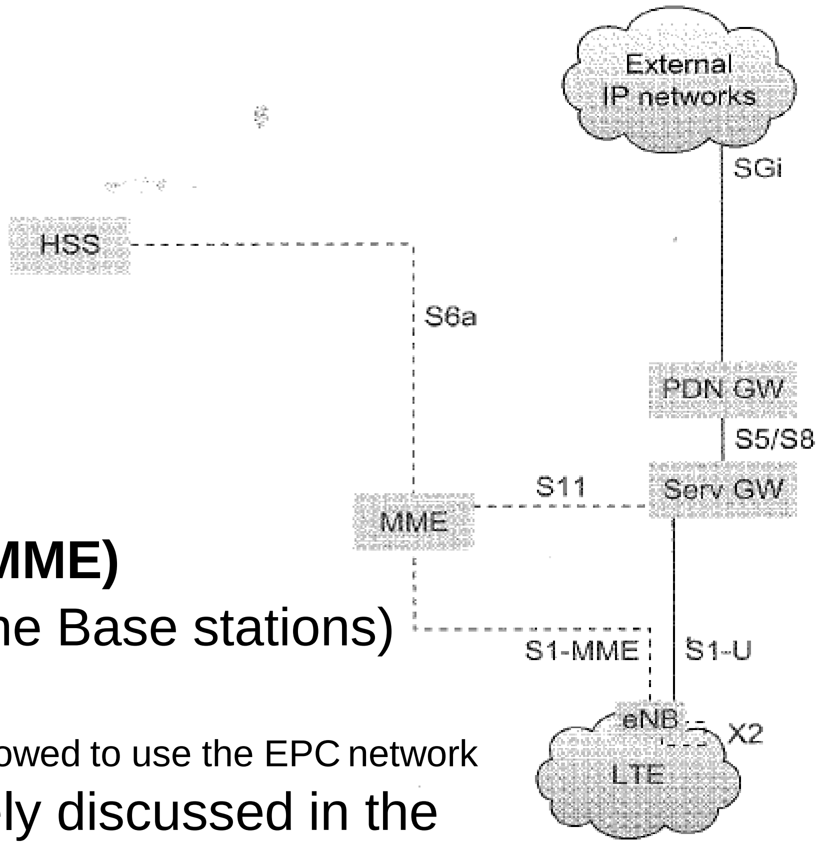
EPC Architecture

- 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**

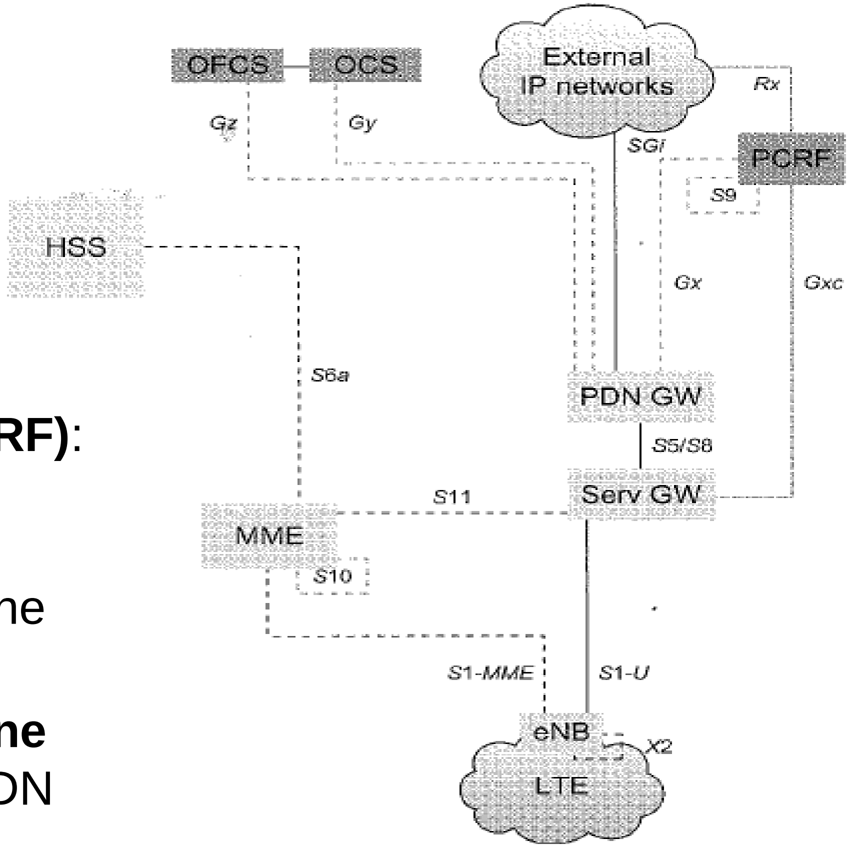
  On the signaling/control path
  - Controls the MME
  - Marks “packets” for QoS differentiation purpose

  On the data path
  - Buffers data as appropriate

EPC – A more advanced architecture

EPC for LTE (Reference 1)

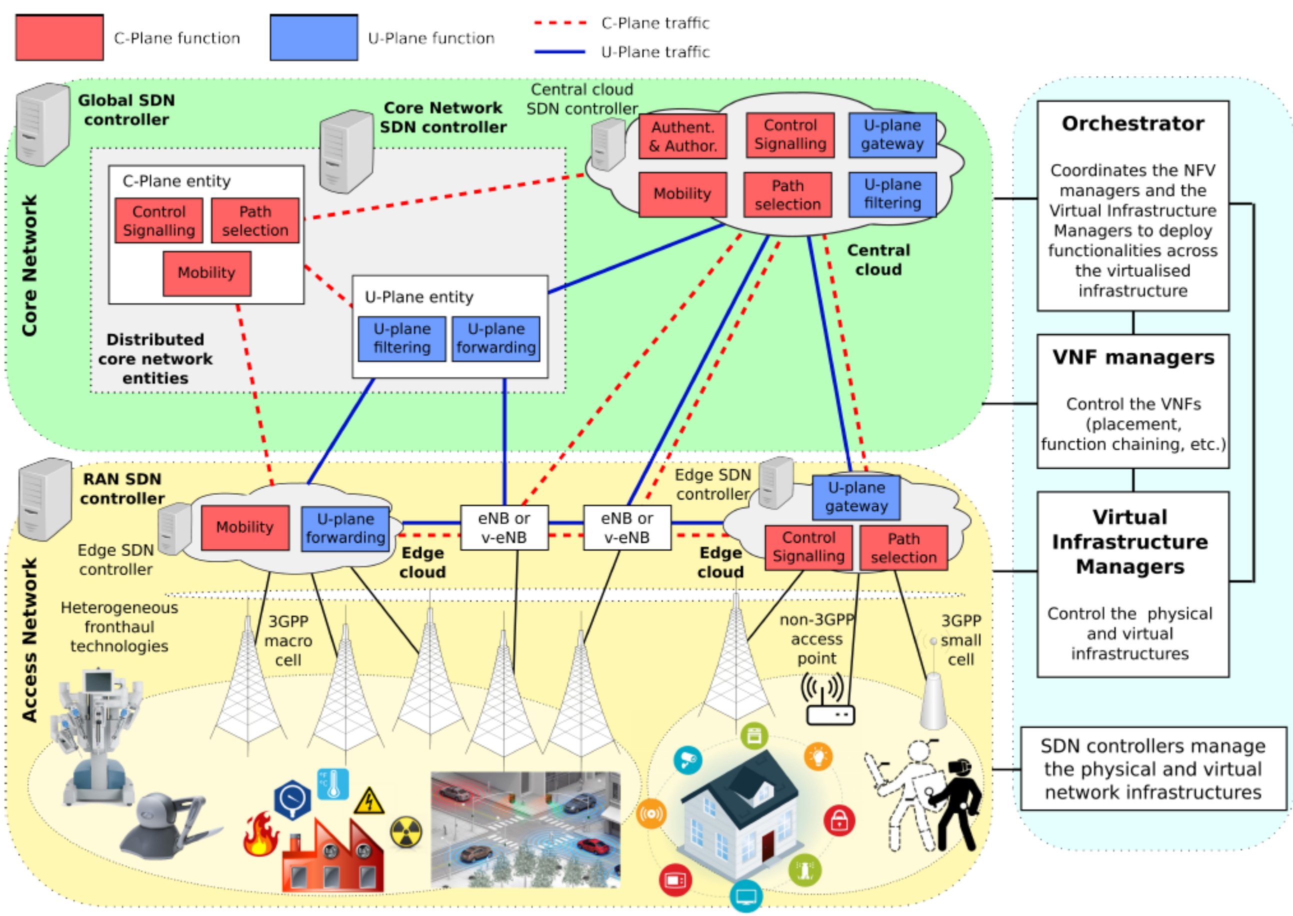
- Dotted lines: signaling/control path
- Solid: data path



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***)

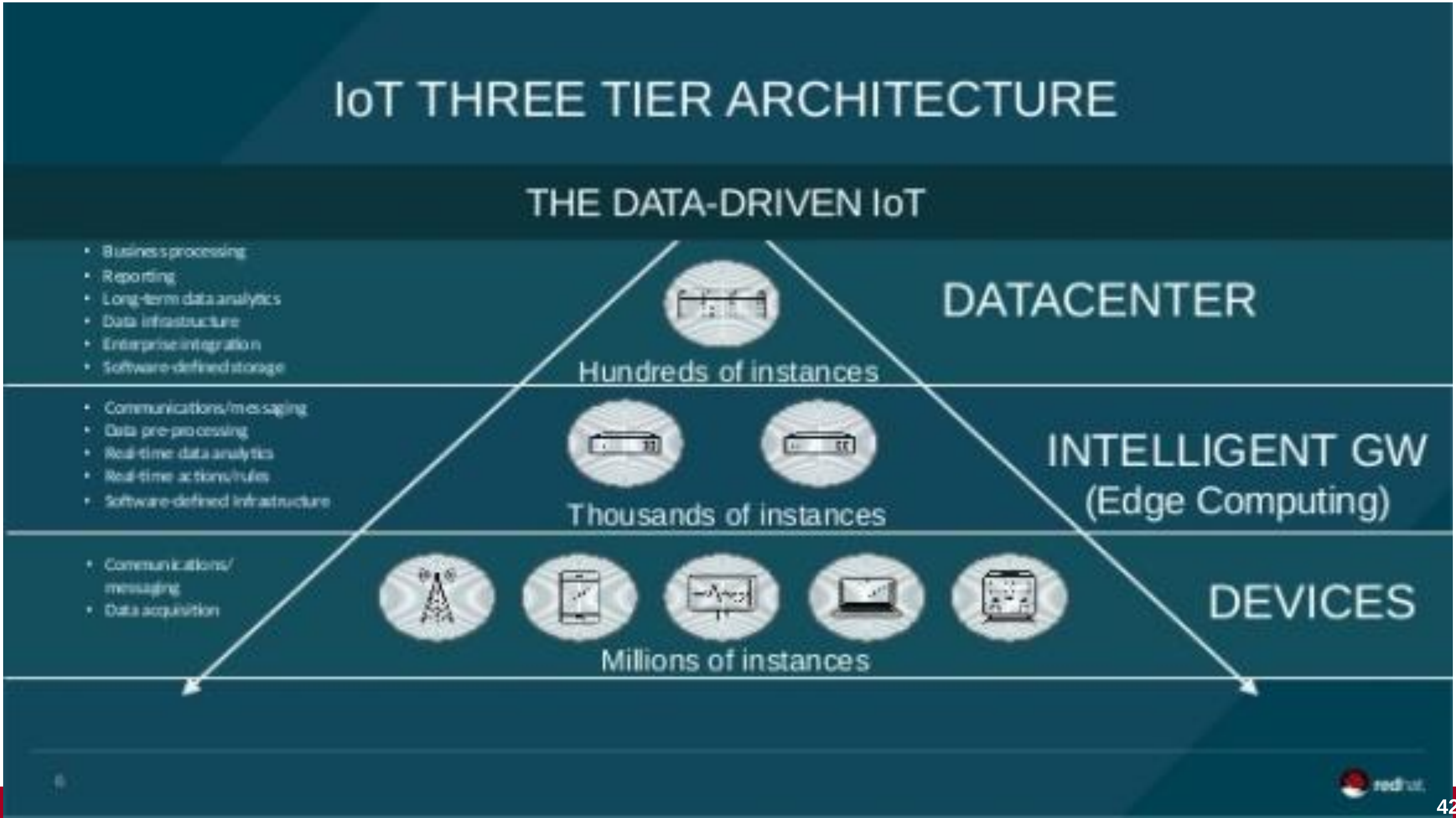
Deep transformation of how we organize, manage, and access ***virtualized distributed resources***

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***

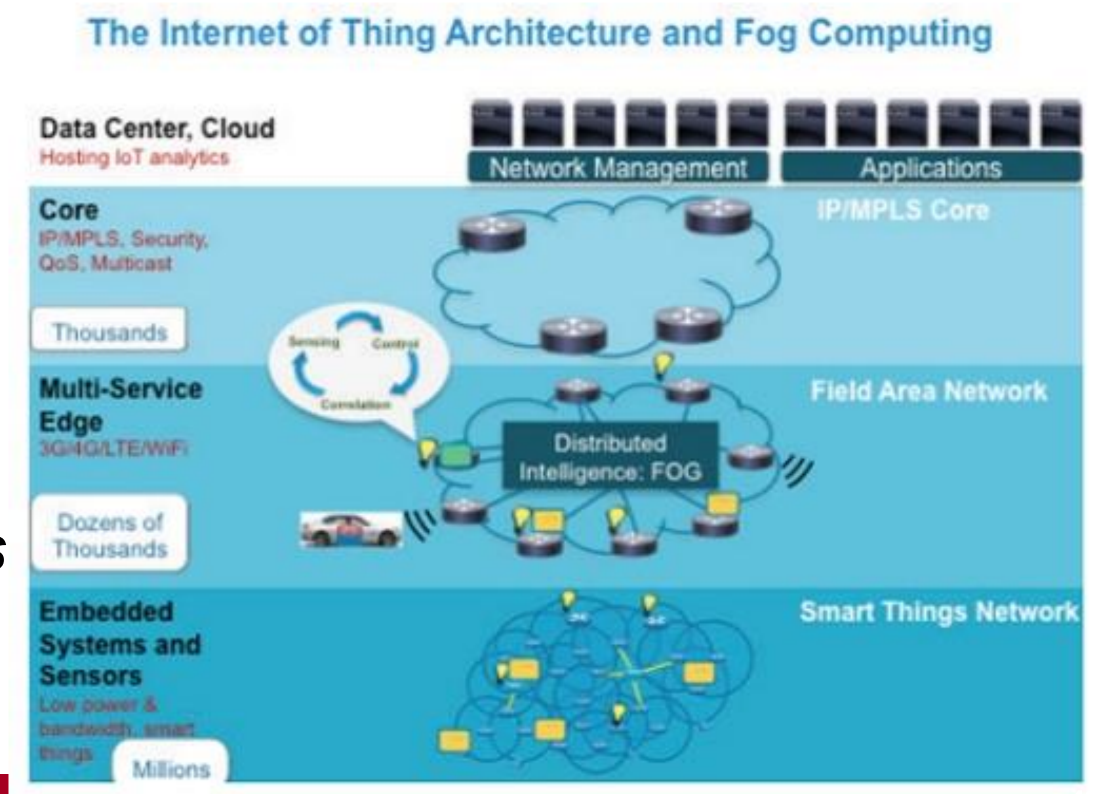
Edge Computing for Industrial IoT: Quality Requirements



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

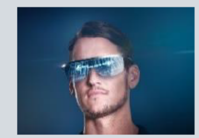
Edge and 5G
for Constrained Latency

Credits to Bosch

Selected Performance Requirements

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

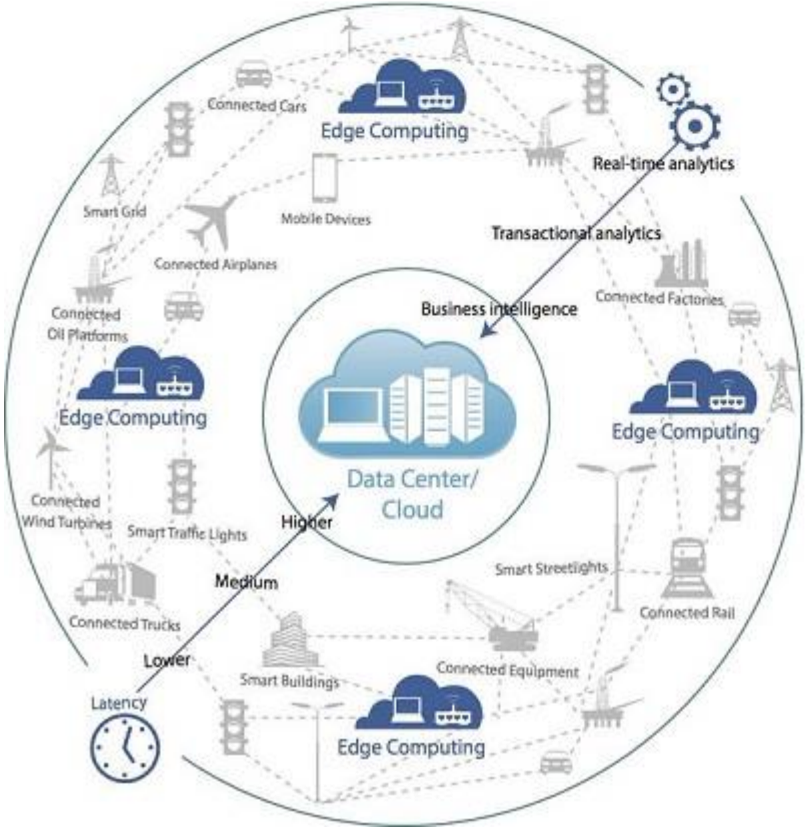
uRLLC[2] → most challenging (Motion Control, Safety Traffic)

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, …

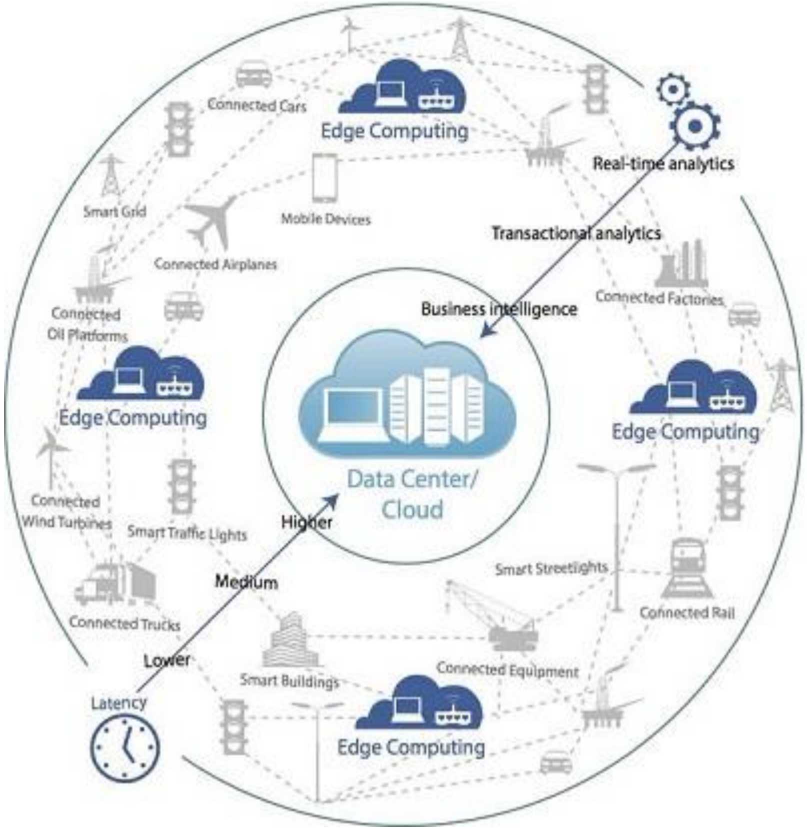
Edge Computing: Definition

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, …



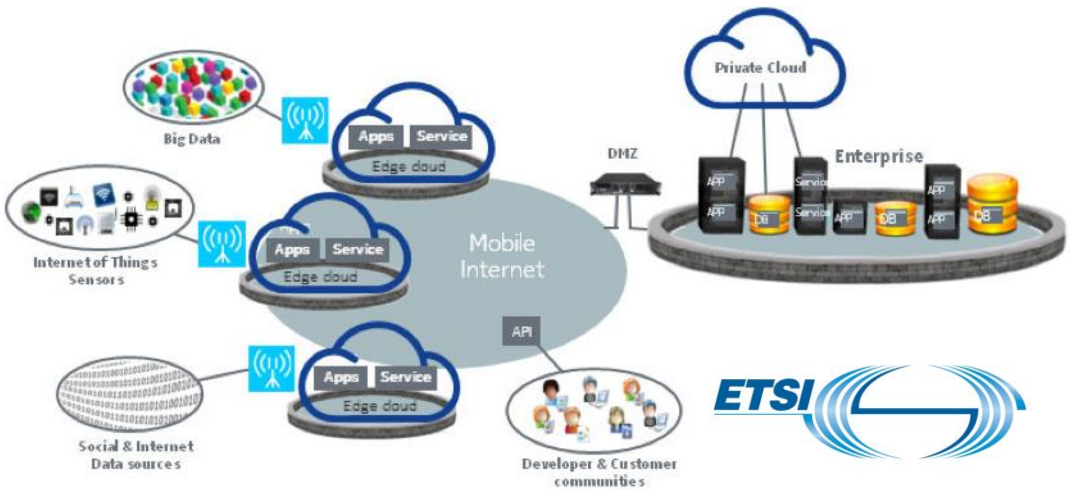
Fog Computing

- **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

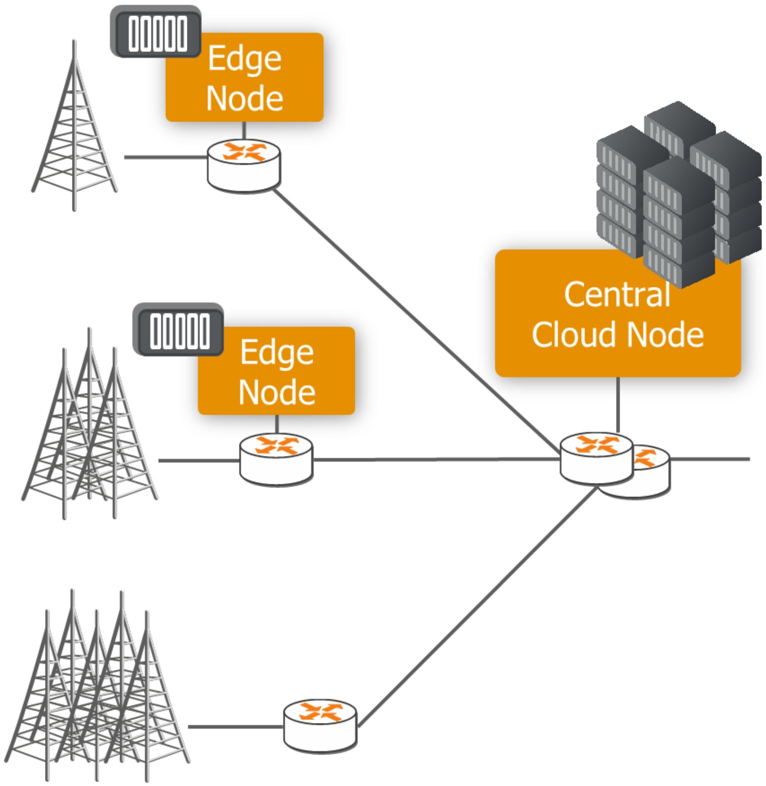


- **Cons**: limited number of edges and low re-configuration rate, due to high costs of configuration and maintenance

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

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

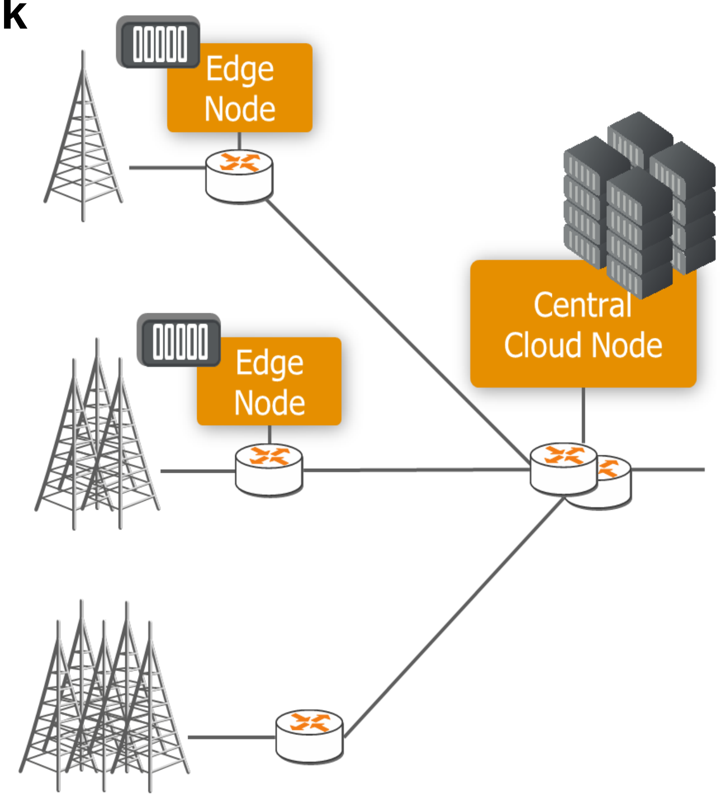
- **On-Premises:** the edge can be completely isolated from the rest of the network
- **Proximity:** capturing key information for analytics and big data
- **Lower Latency:** considerable latency reduction is possible
- **Location awareness:** for location-based services and for local targeted services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



Local vs Global: the MEC Use Cases

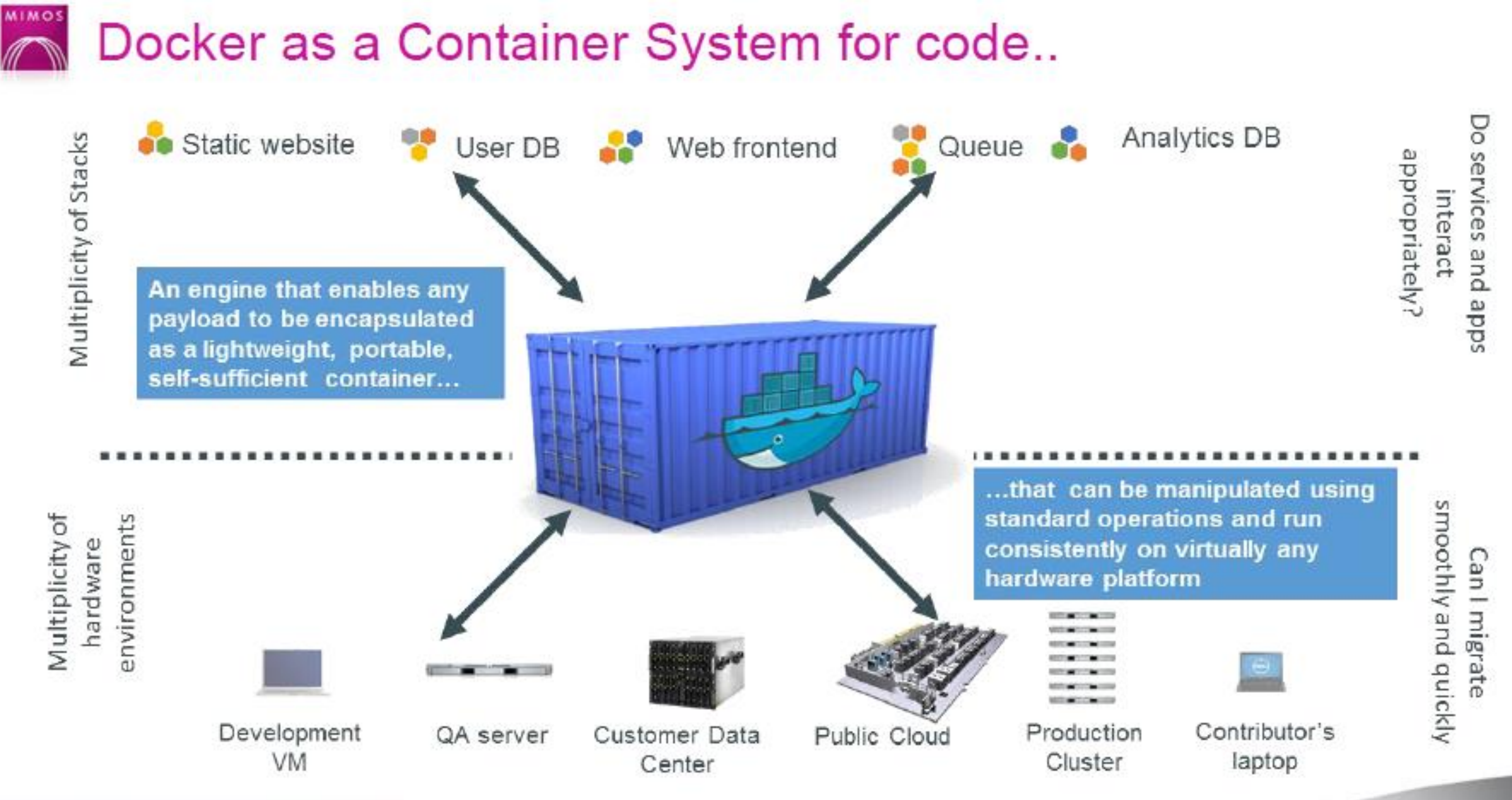
Depending on the integration with the core network three types of use cases are defined

- **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



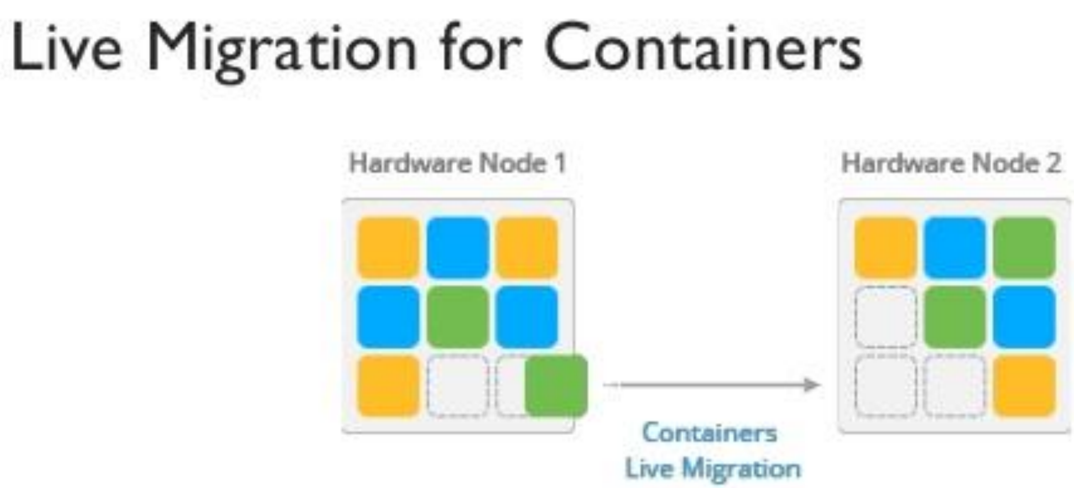
We get back to this in few slides...

Edge Computing & Docker



Edge computing empowered by ***containerization***

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

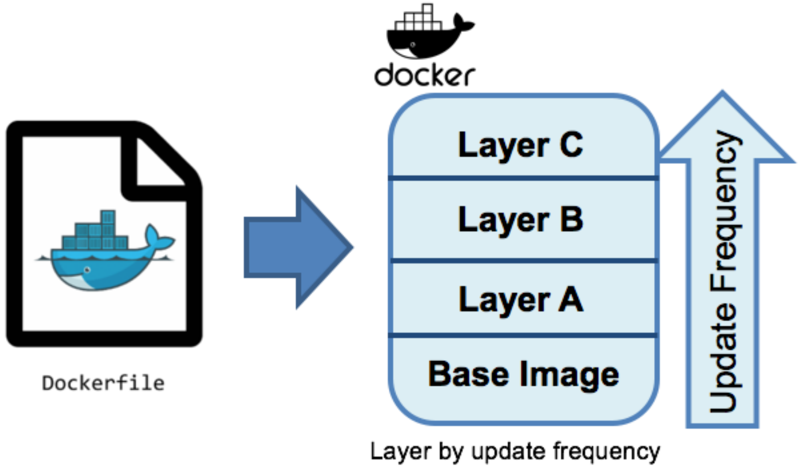


CRIU – Checkpoint/Restore In Userspace

OpenVZ
Virtuozzo Containers

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, …***

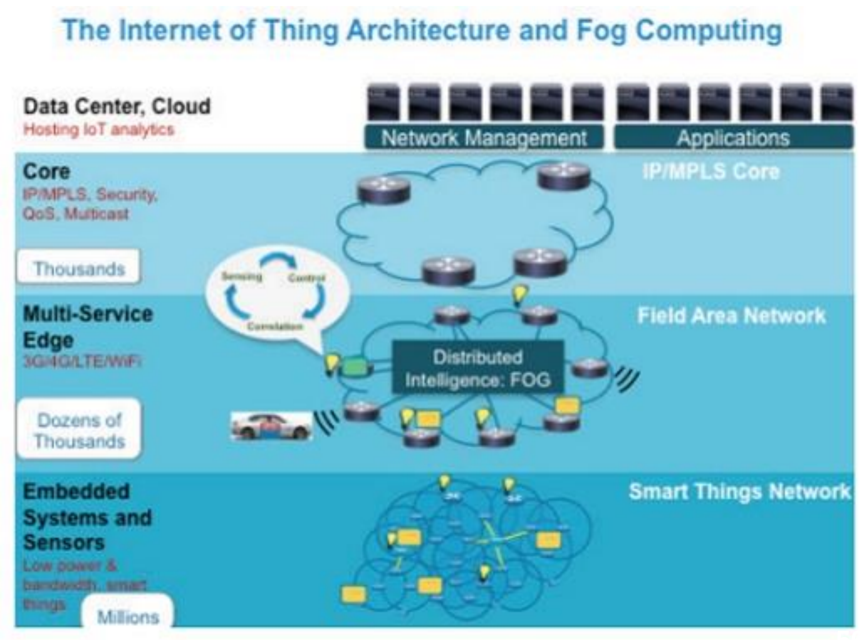
And not only for the above use cases!!!

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
- …

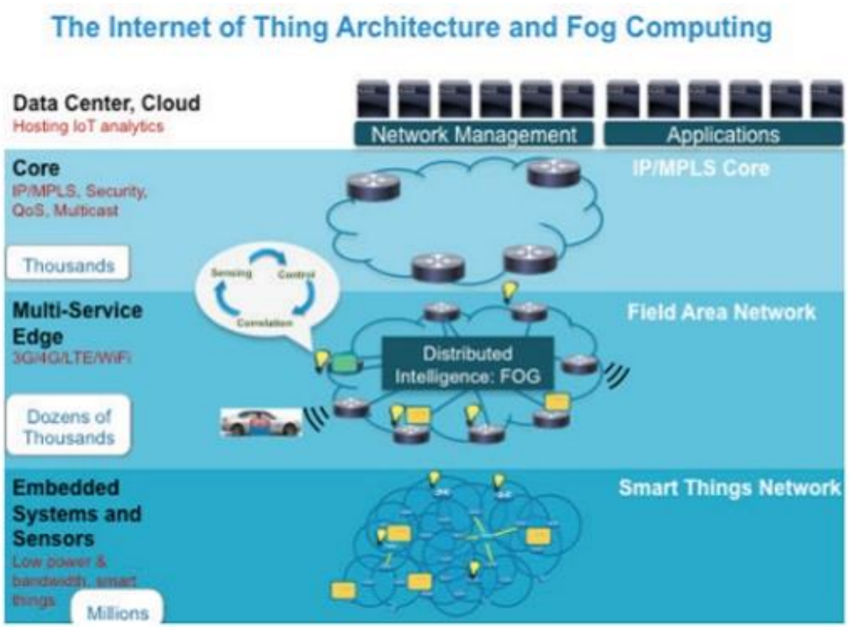


Architecture Modeling

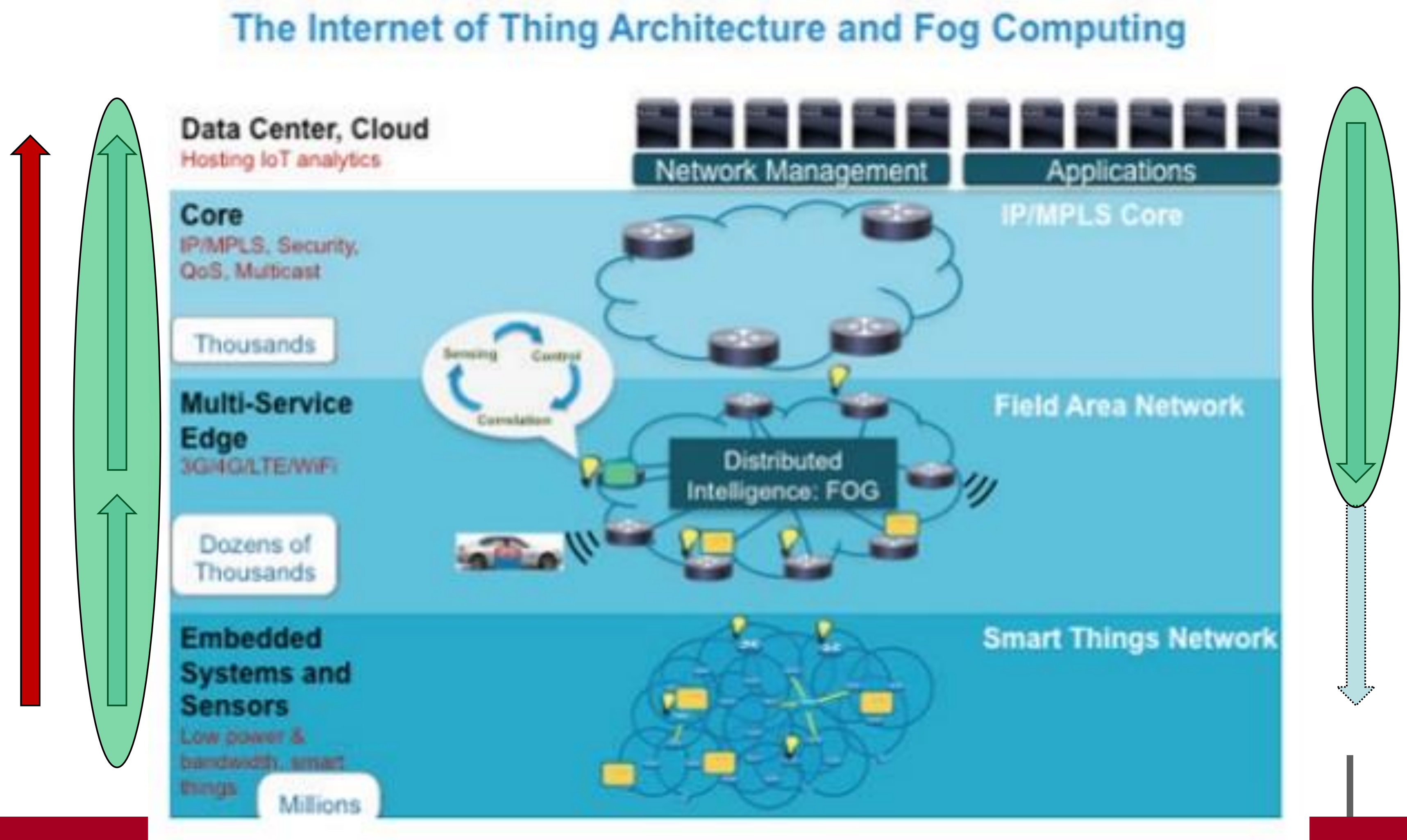
Dynamic distribution of storage/processing (network resource allocation?) functions in all the three layers of a node-edge-cloud IoT deployment environment

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

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



Architecture Modeling
Need for new models
Offloading and Onloading



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