

Cloudifying the mobile network stack: benefits and challenges

Pablo Serrano

<http://www.it.uc3m.es/pablo/>



CONTEXT AND OVERVIEW

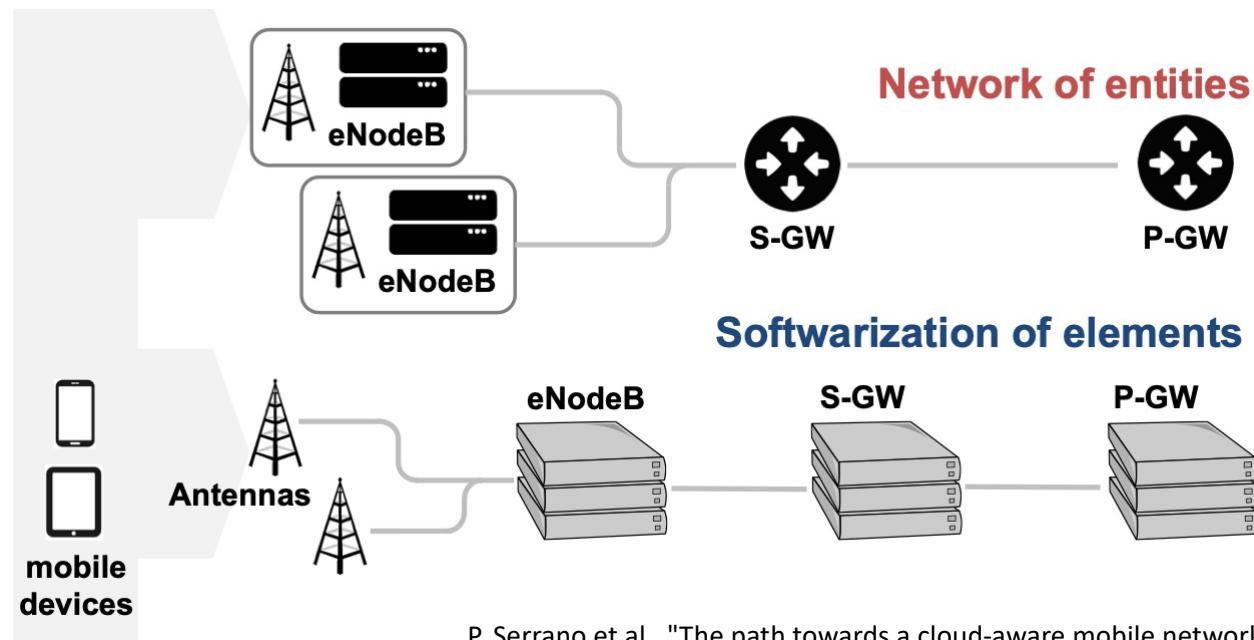
Softwarize all the things

- “Software is eating the world”, Marc Andreessen, The Wall Street Journal on August 20, 2011.
- Software Defined Networking
 - OpenFlow, 2008
- Virtualization
 - OpenStack, 2010
 - VMware, 2000s



Softwarizing the mobile stack

- Physical Network Functions (PNFs) tightly coupled with the hardware substrate running them



P. Serrano et al., "The path towards a cloud-aware mobile network protocol stack",
Transactions on Emerging Telecommunications Technologies, Vol. 25, Issue 5, May 2018

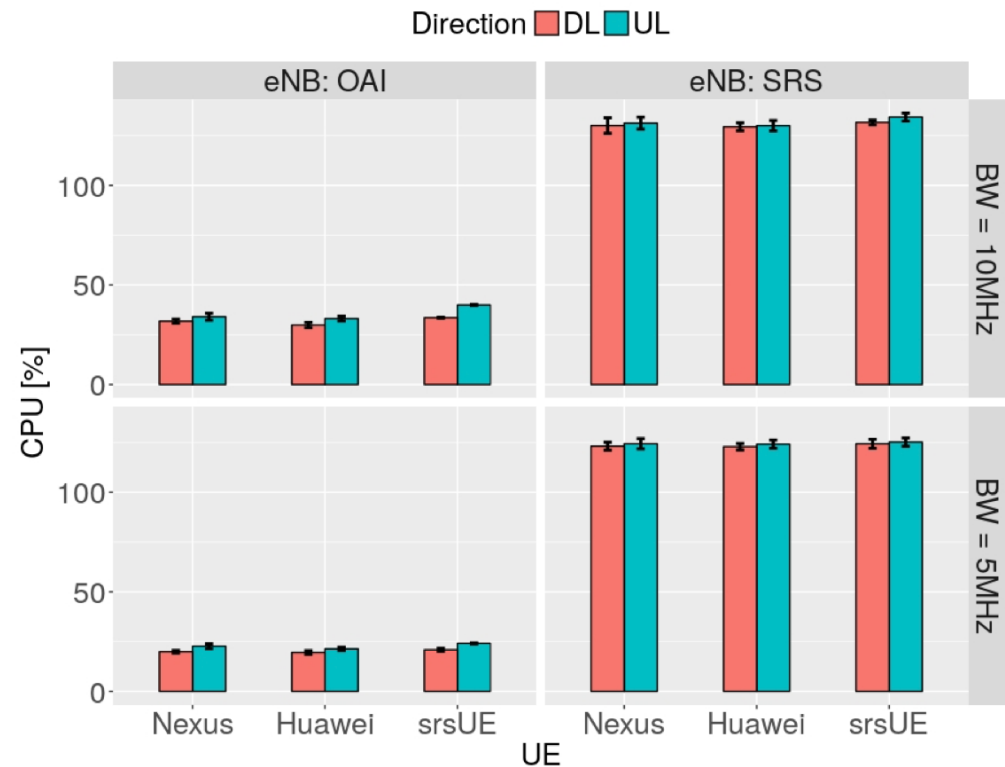
Two SW projects

- I. Gomez-Miguel et al., “SrsLTE: An Open-Source Platform for LTE Evolution and Experimentation,” in ACM WiNTECH 2016
- F. Gringoli et al., “Performance Assessment of Open Software Platforms for 5G Prototyping”, IEEE Wir. Comm. Magazine, 2018



Resource Consumption

- Software
 - Ubuntu 16.04
 - OAI – version 0.6.1;
 - SRS – version 2.0-17.09 of the srsENB application.
- HW
 - USRP-B210
 - Intel Core i7-7700K CPU
 - 4 Cores at 4.2GHz,
 - 16GB of DDR4 memory



Customization and Extensibility

- Task: dynamically fix the MCS assignments that the eNB enforces on the UEs
- OAI
 - Less straightforward
 - MCS index hardcoded
- srsLTE
 - Fairly intuitive
 - Modular framework

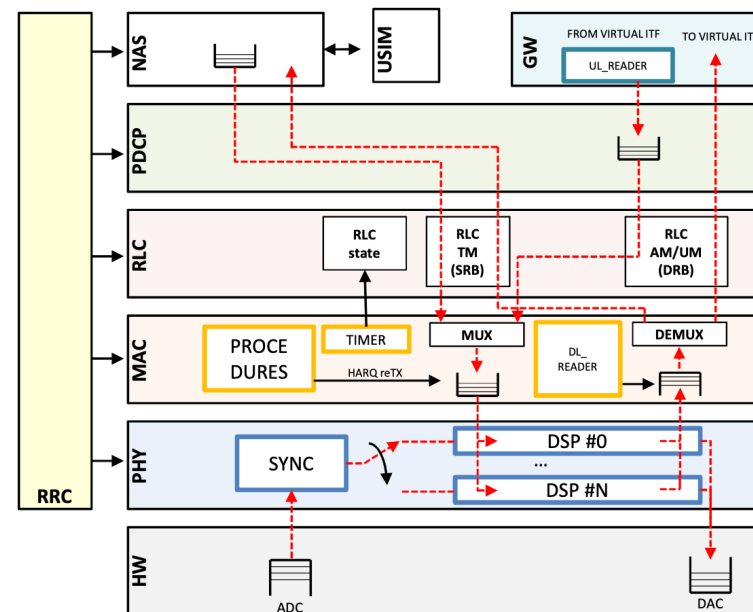
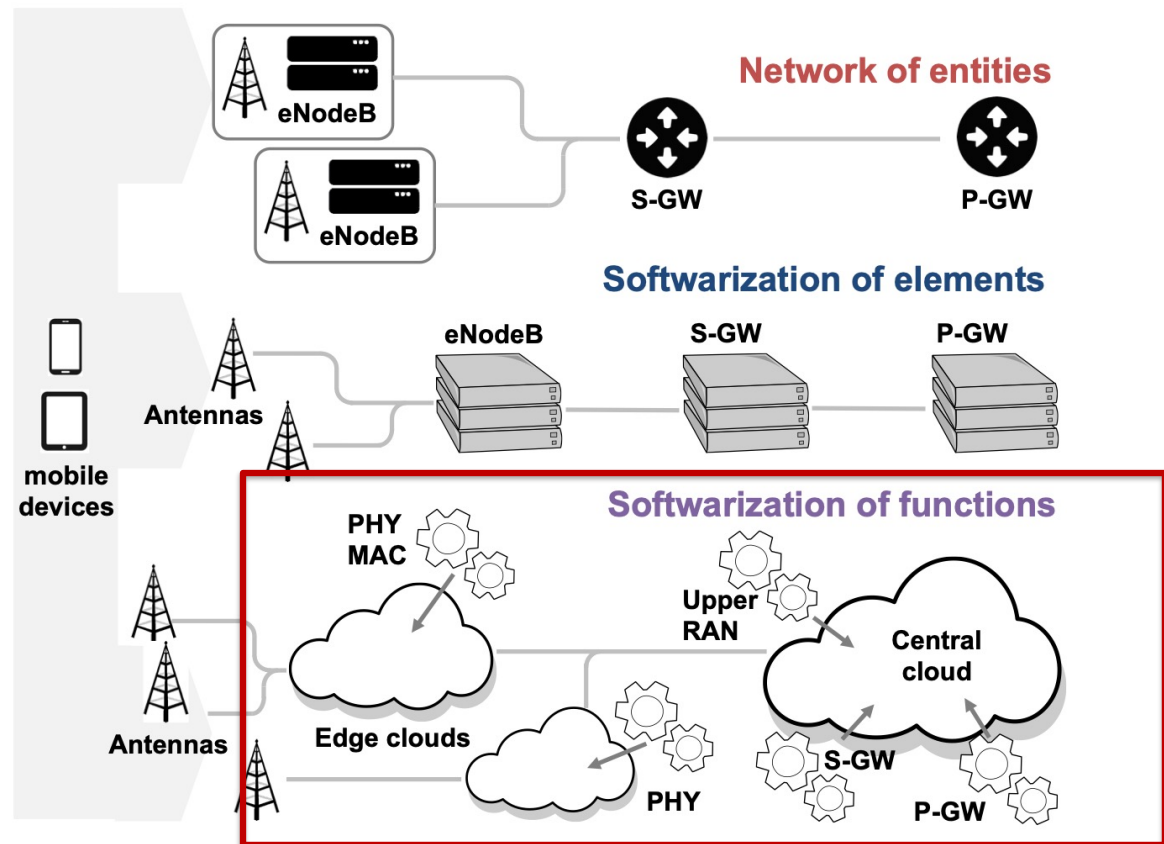


Figure 2: Threading architecture in srsUE. Boxes with coloured borders are threads.

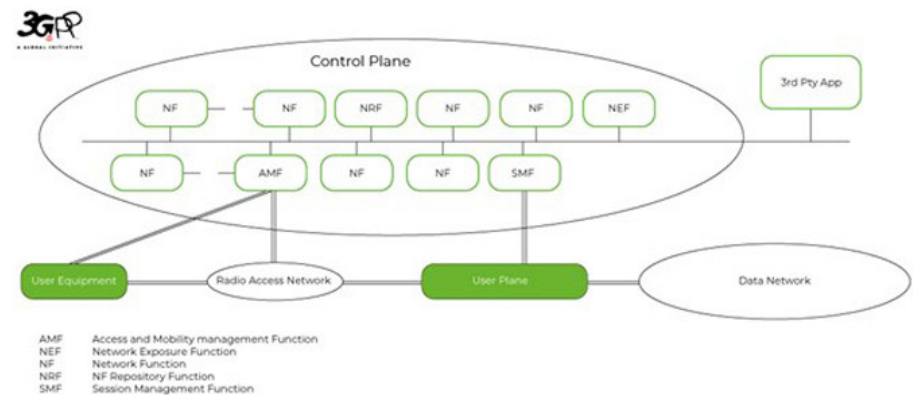
Modularizing the mobile stack

- Modularization:
defining and
instantiating
re-usable and
highly focused
Virtual Network
Functions (VNF)



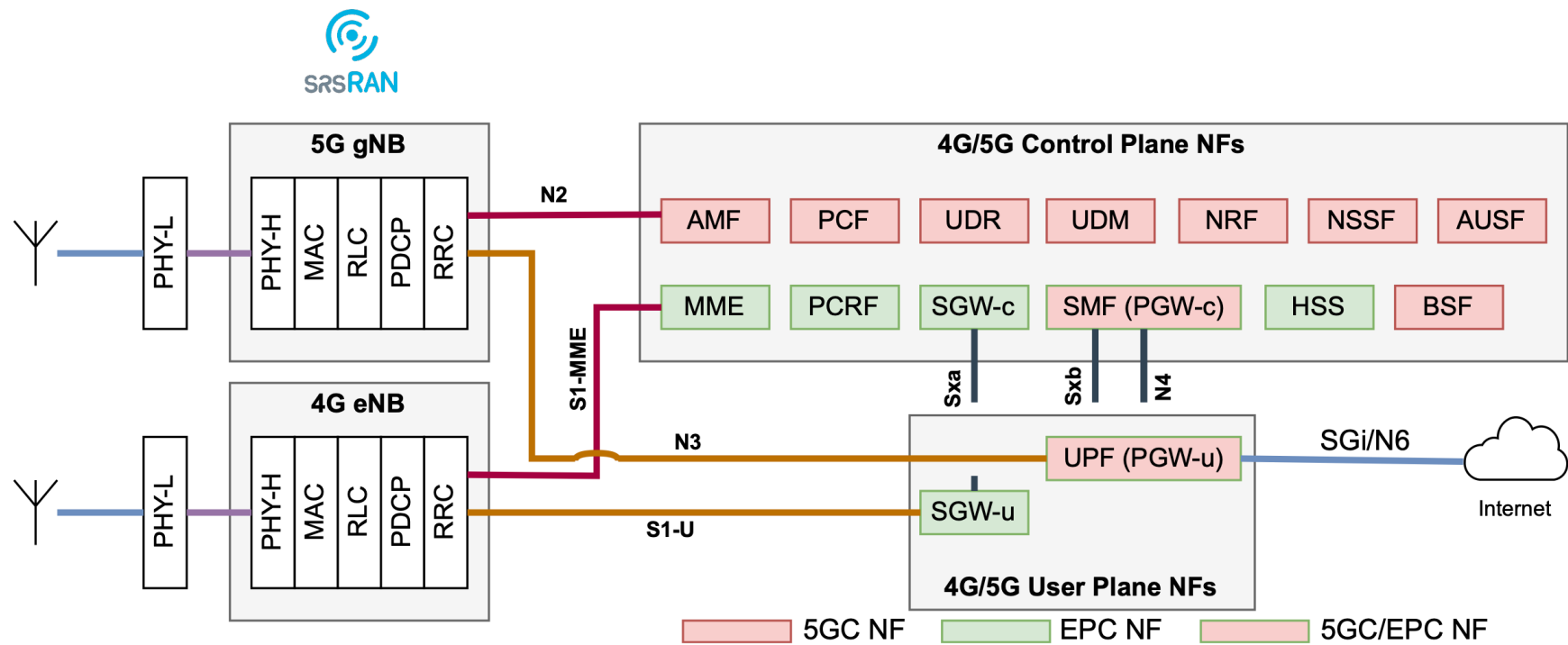
It's already happening

- Cloud-Native Network Functions (CNF)
 - Making its way into the current technology.
 - Core Network VNFs only



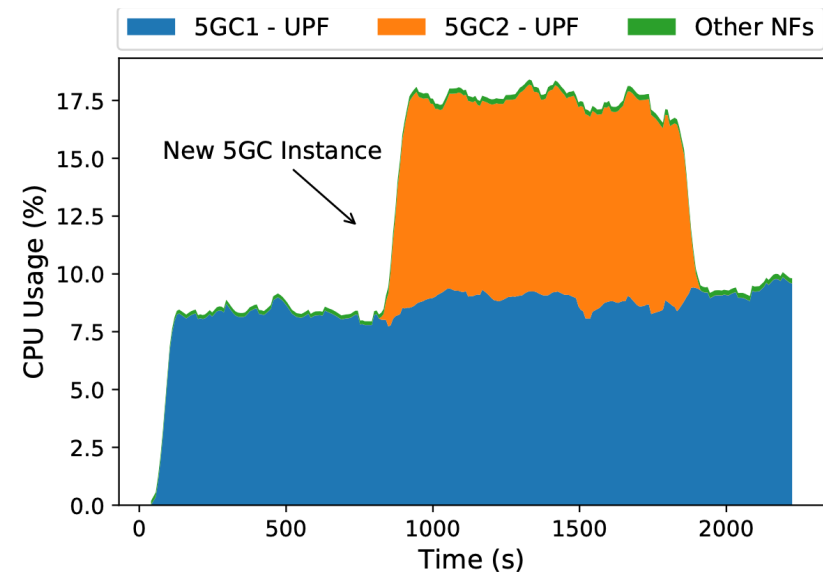
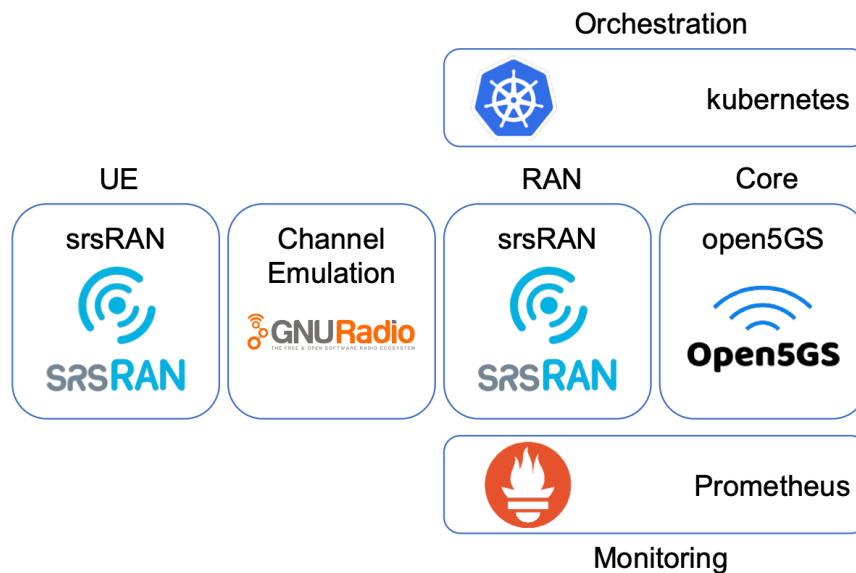
- 3GPP Release 15
 - Service Based Architecture (SBA)

Already happening (core network)



Already happening (core network)



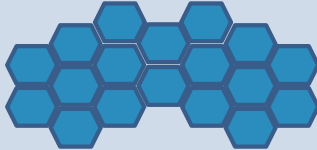

- N. Apostolakis et al. “Design and Validation of an Open Source Cloud Native Mobile Network”, IEEE Comm. Magazine, 2022



Summary

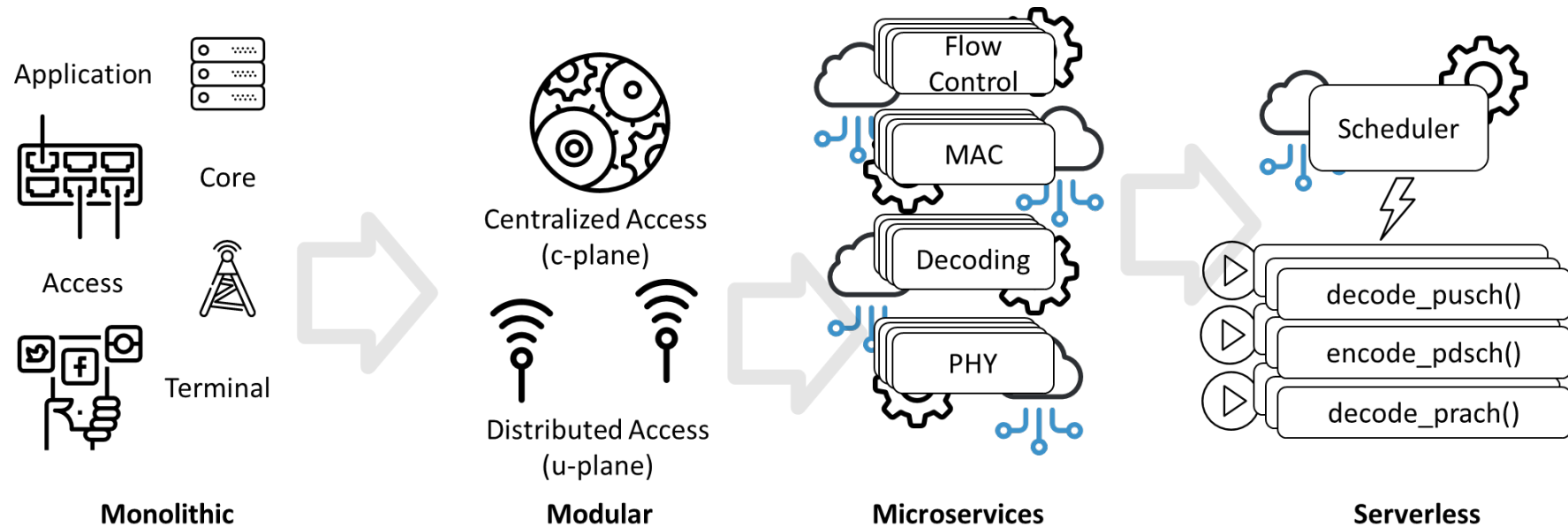
- Mobile Networking is adopting two key technologies from Computer Science:
 - Softwarization & Modularization
 - From telco engineers to software engineers
- But it's lagging
 - Little attention to the RAN (this talk)
 - Adapt to micro-services

Mobile SW in Context

	Single Server PNF	Multi-tier VNF	Microservices H. Modular VNFs	Serverless
Architecture				
Re-Configuration Re-Orchestration Frequency	Years	Months	Many times per day	Continuous
Orchestration Complexity	Low	Moderate	High	Very High

Vision

- The softwarization shall involve all domains, including the most challenging: the RAN



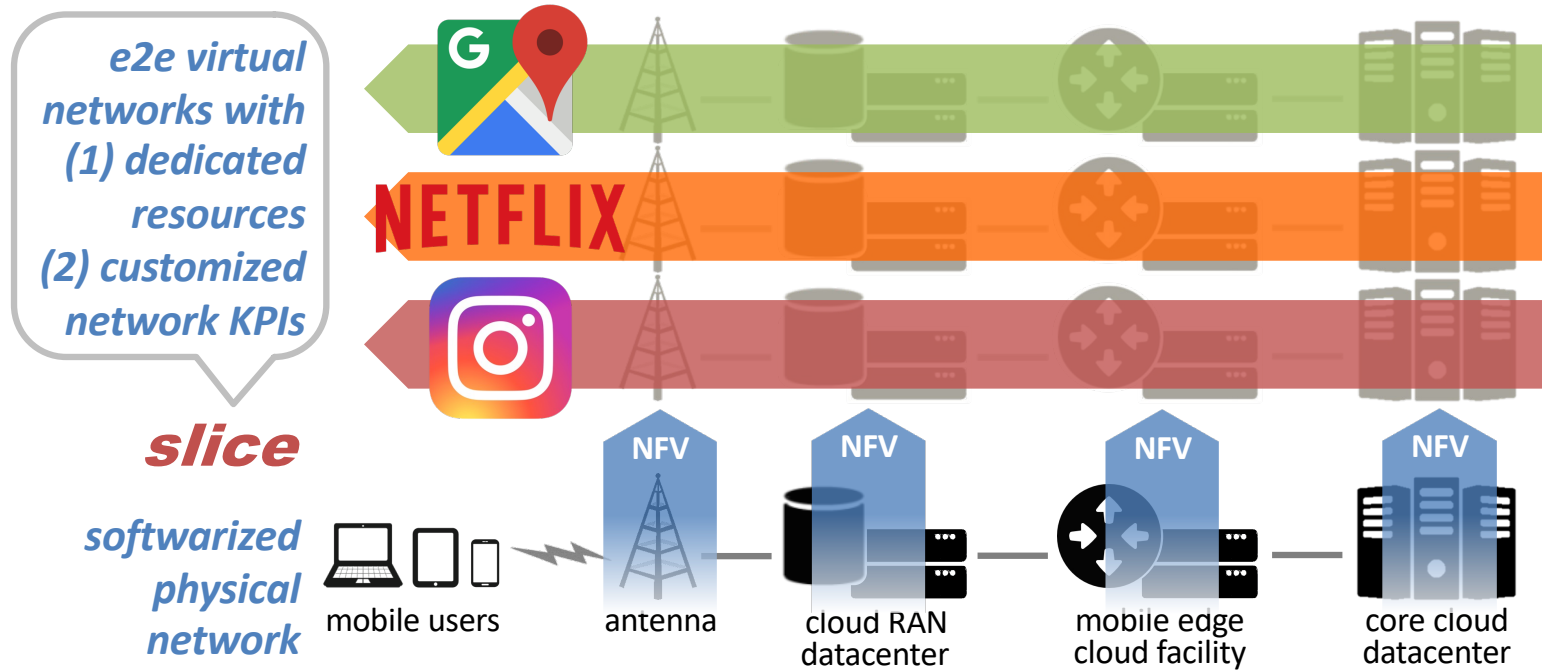
Benefits

- General-purpose hardware (from €€€ to €)
- More agility
 - Development times *“From 90 days to 90 minutes” (2017)*
- Operate *à la* cloud (cloudify)
 1. Resource on demand: efficiency
 2. Resource elasticity: resiliency

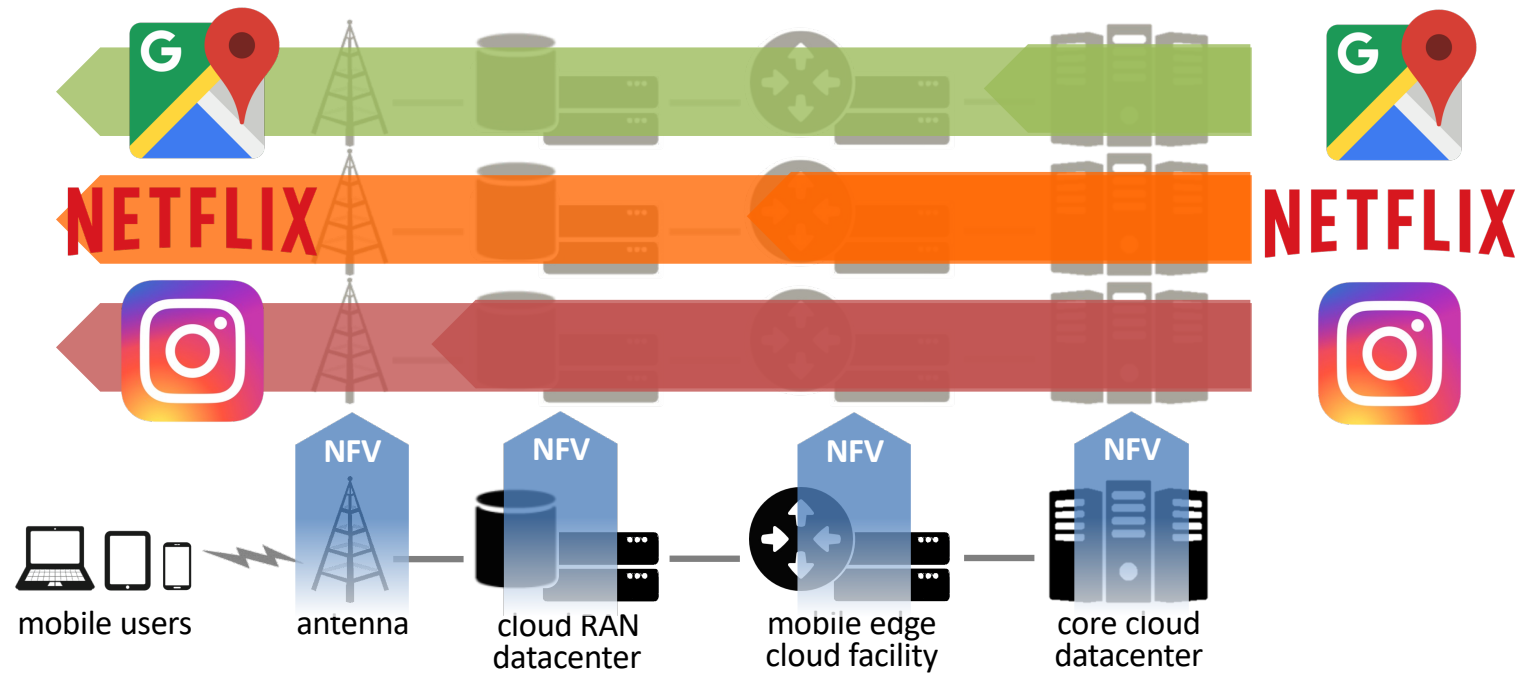
} Rest of the talk

1. RESOURCE ON DEMAND: EFFICIENCY

Network Slicing



Slicing depth / Aggregation level



Trade-off

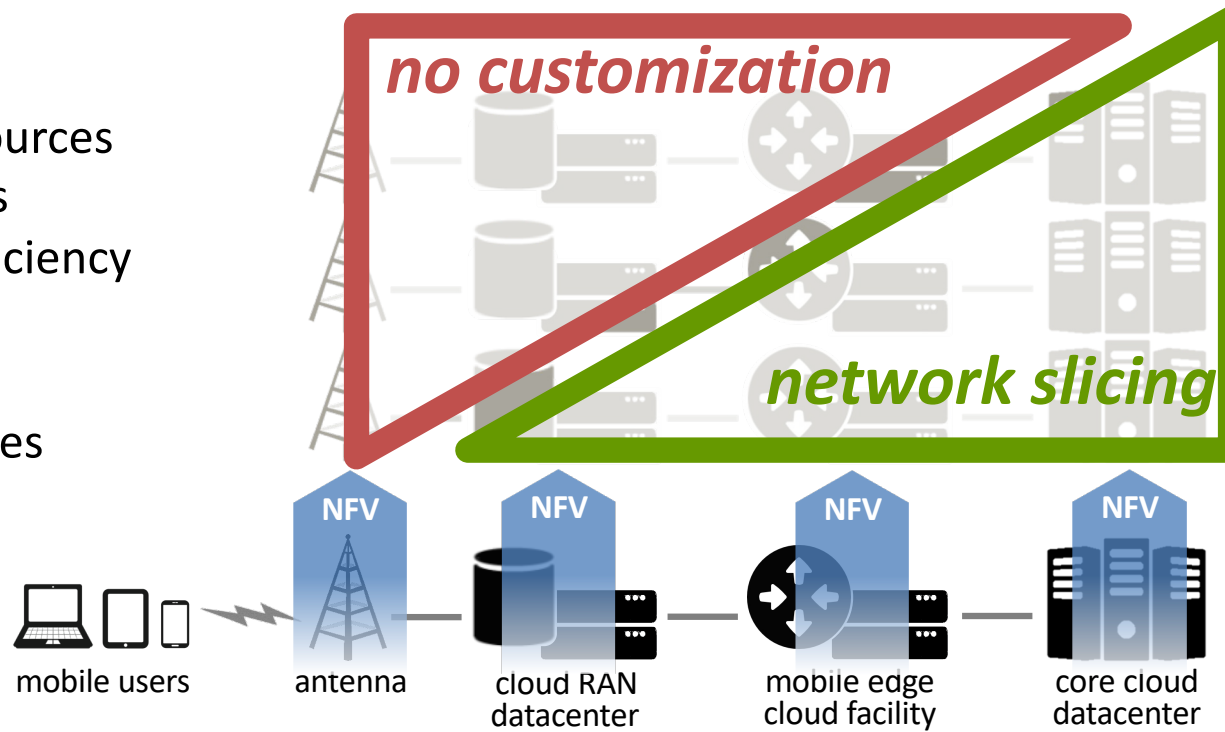
No customization

No booking of resources
No QoS guarantees
High multiplex. efficiency

Network slicing

Booking of resources
QoS guarantees
Poor efficiency

Objective:
How to
quantify this

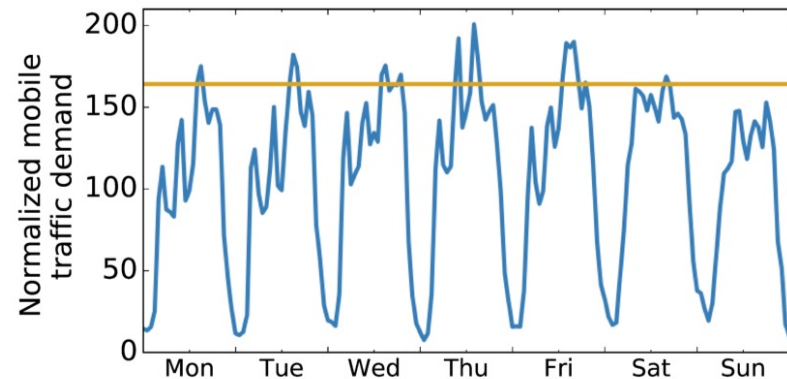
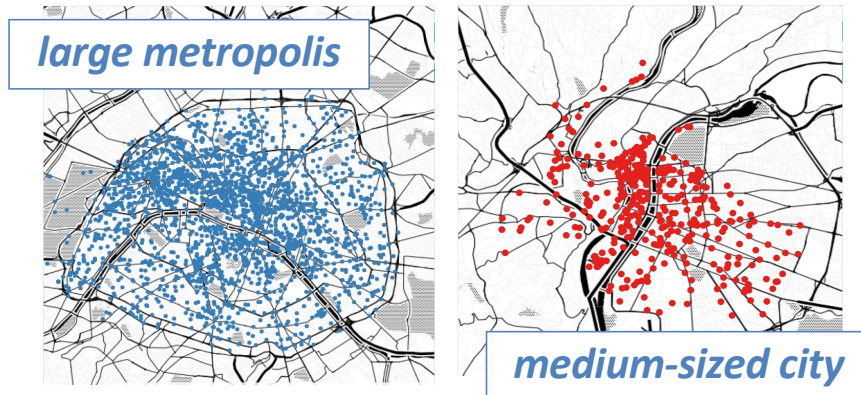


NETFLIX



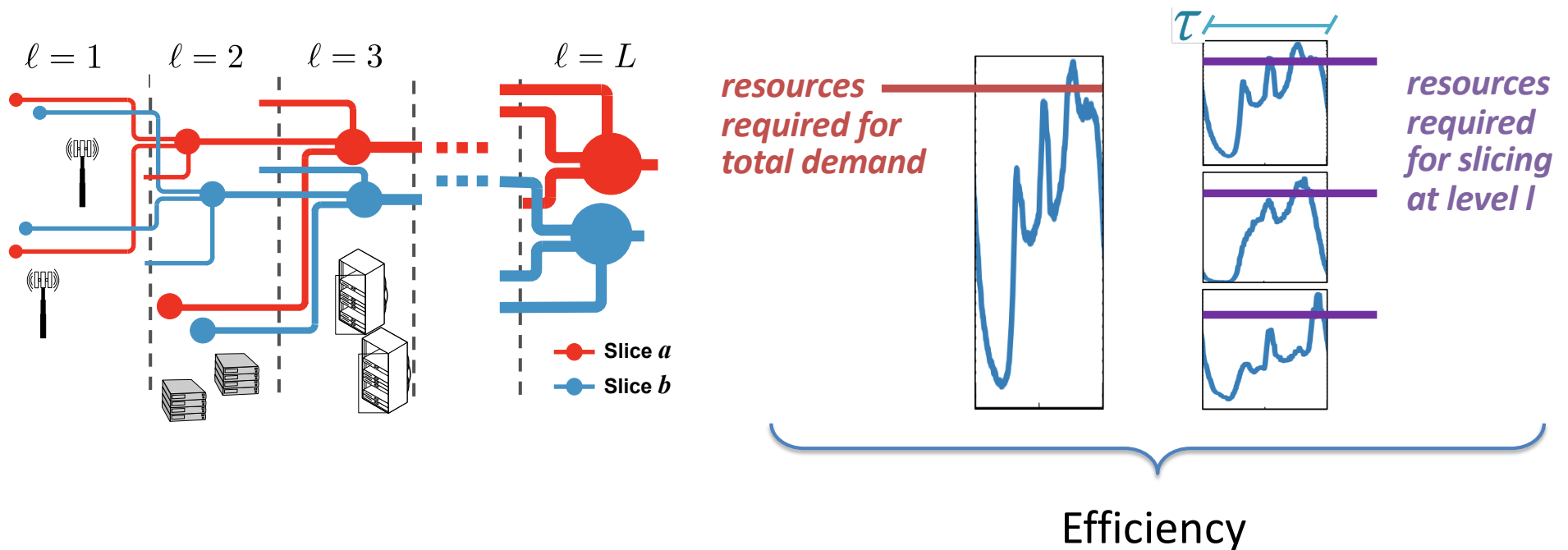
Data

- Two urban areas in a European country
 - large metropolis + medium-sized city
 - 3 months data from a mobile network operator
- Service demands measured at the antenna sector



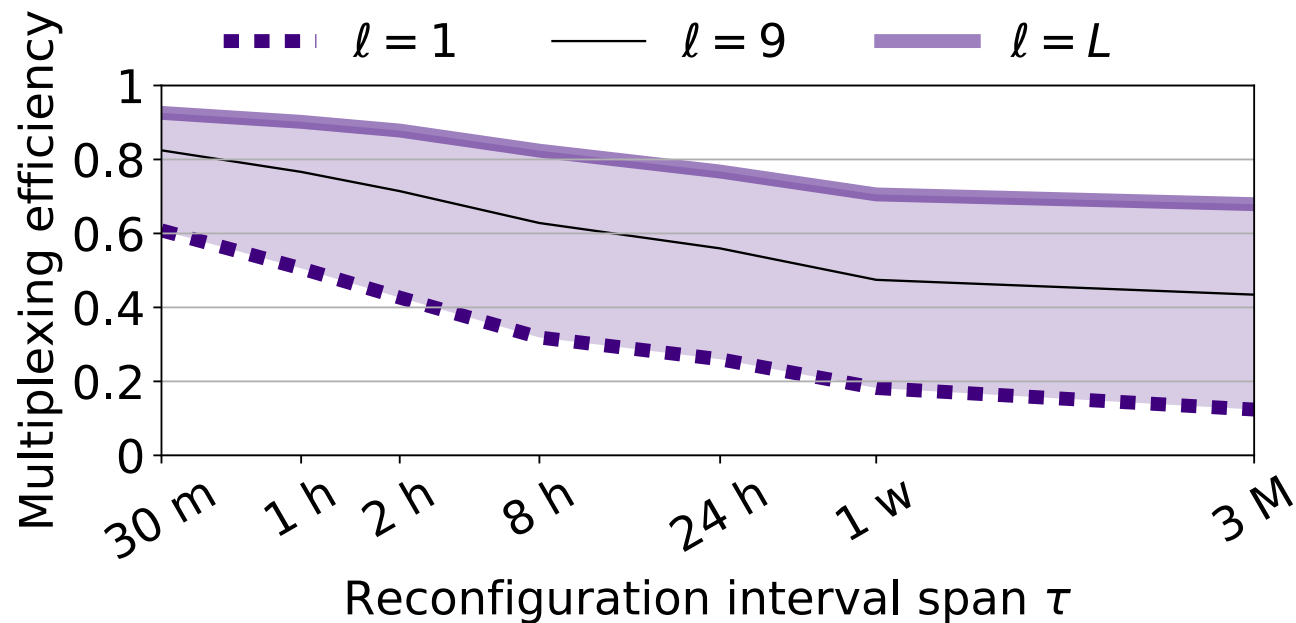
Depth (level) and Update freq.

- Impact of depth and reconfiguration time



Software needs to be *agile*

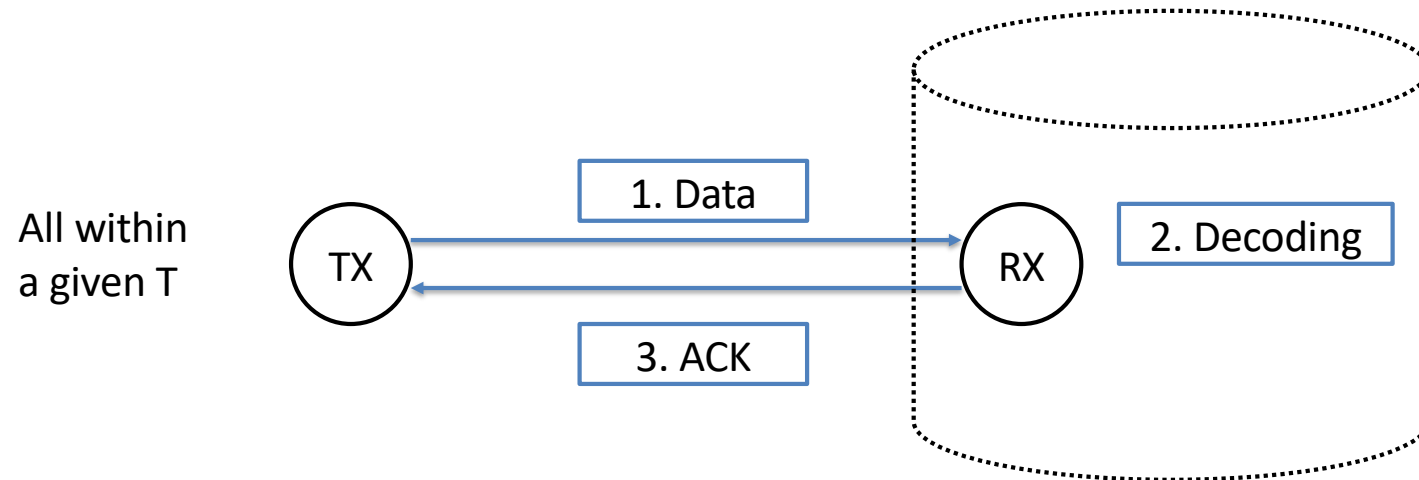
- Impact of aggregation level and reconfiguration time



2. RESOURCE ELASTICITY: RESILIENCY

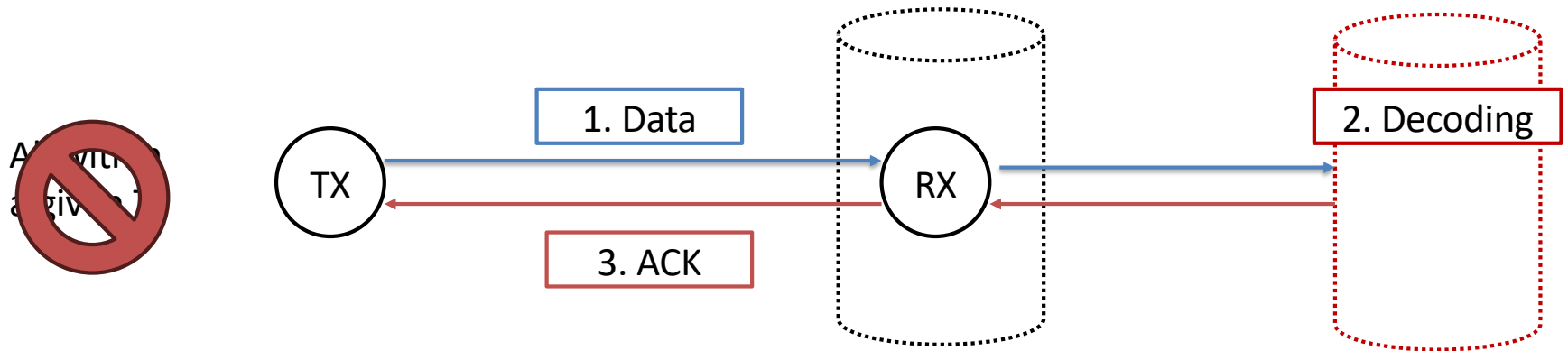
Resource elasticity

- Communication stack: tight interactions

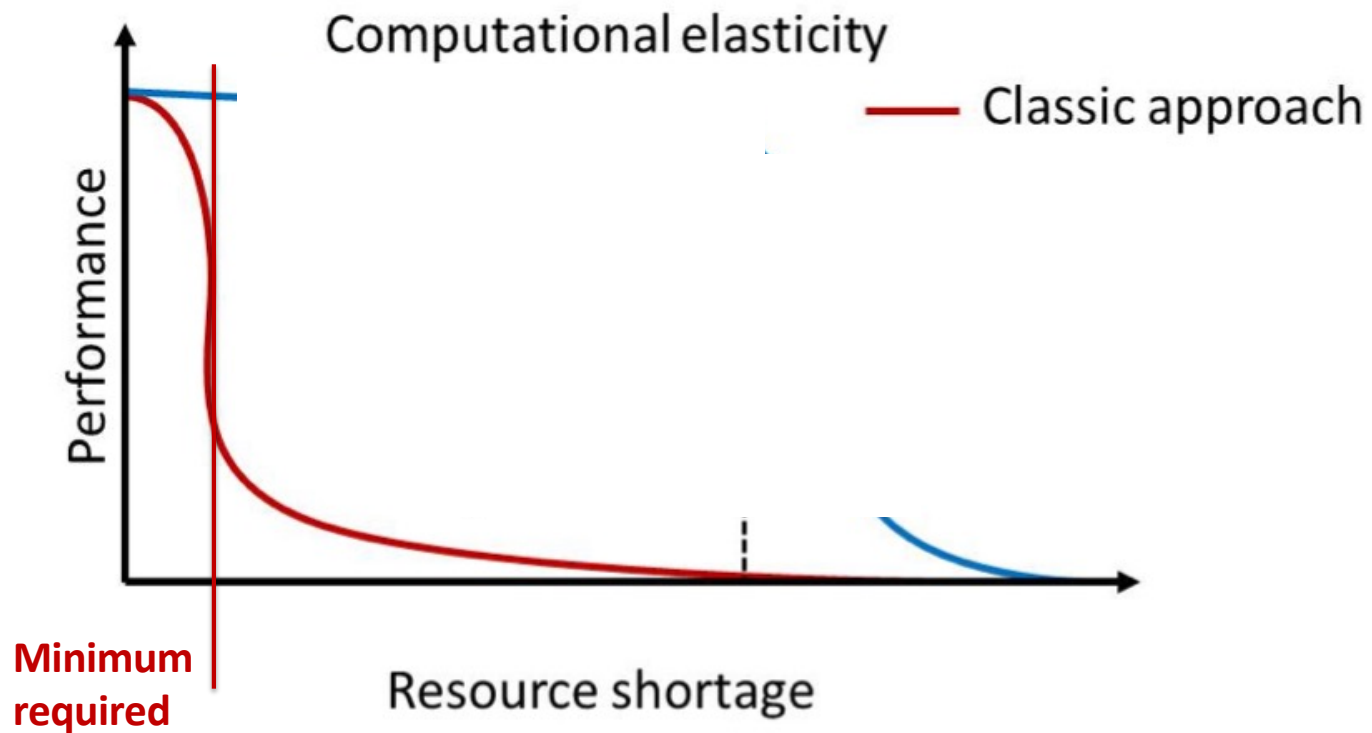


Resource elasticity

- What if we (careless) *cloudify* the decoding?

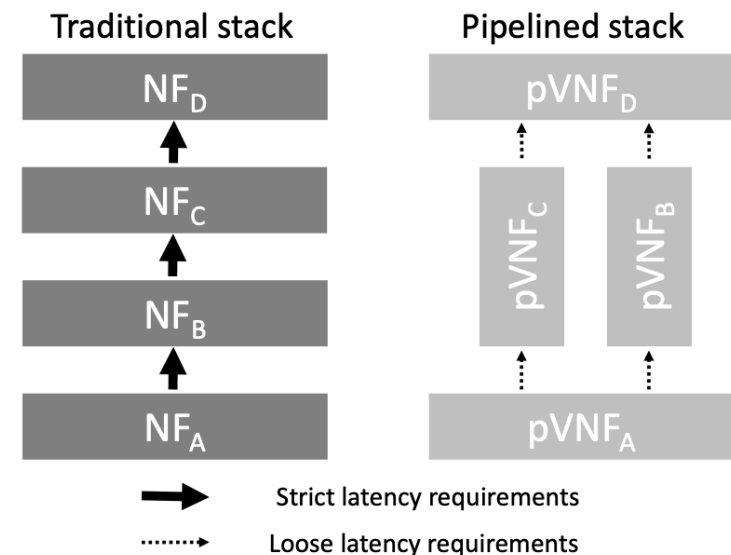


Inelastic application

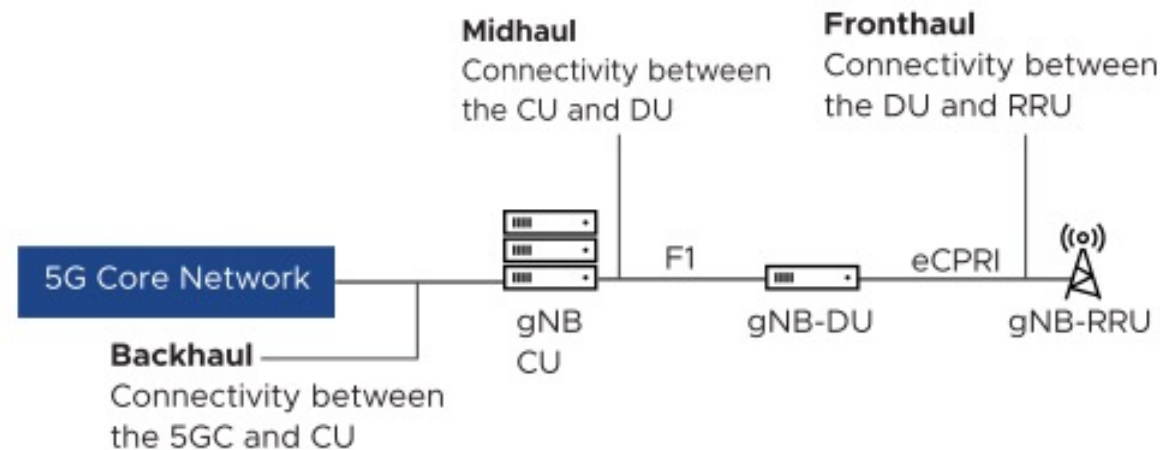


Challenge

- Need to re-design VNFs
- Current RAN functions
 - High load on the CPU
 - Stringent timing requirements
- We need new functions
 - Lessen requirements
 - Resource-aware execution



vRAN Architecture



- Centralized Unit (CU): non-real-time processing
- Distributed Unit (DU): real-time processing and coordinates MAC, RLC and PHY
- Remote Radio Unit (RU): amp. & sampling

A DU has to perform many tasks

1. Receive Uplink (UL) subframe (SF) n (OFDM symbols, after FFT)



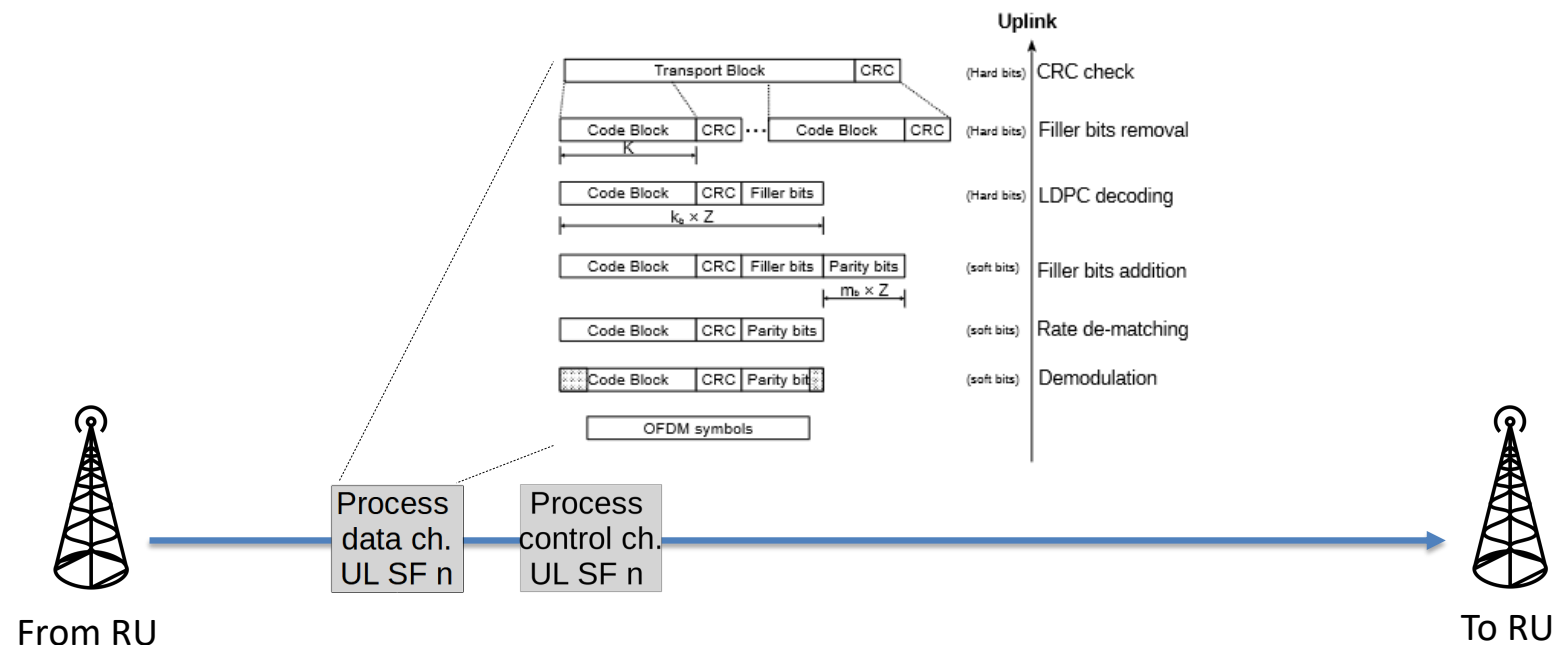
From RU



To RU

A DU has to perform many tasks

1. Receive Uplink (UL) subframe (SF) n (OFDM symbols, after FFT)
2. Process UL data channels in UL SF n
3. Process UL control channels in UL SF n



A DU has to perform many tasks

1. Receive Uplink (UL) subframe (SF) n (OFDM symbols, after FFT)
2. Process UL data channels in UL SF n
3. Process UL control channels in UL SF n
4. Prepare Downlink (DL) SF $n + M$ ($M=4$)
 - Prepare basic synchronization signals
 - Compute radio scheduling grants



From RU

Process
data ch.
UL SF n

Process
control ch.
UL SF n

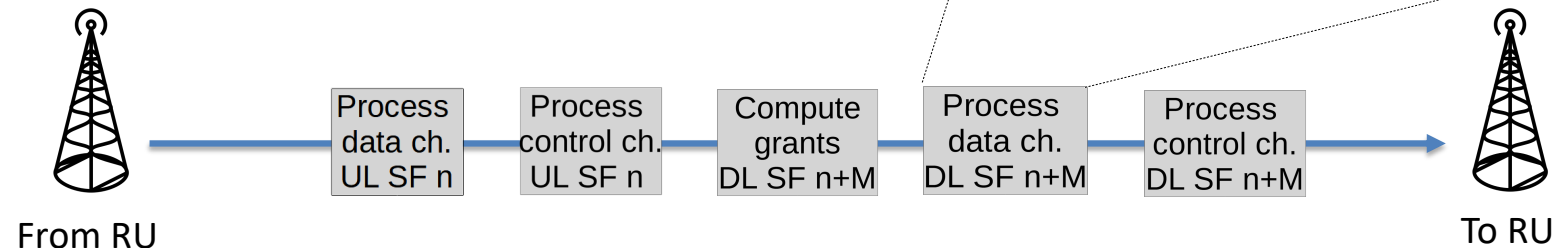
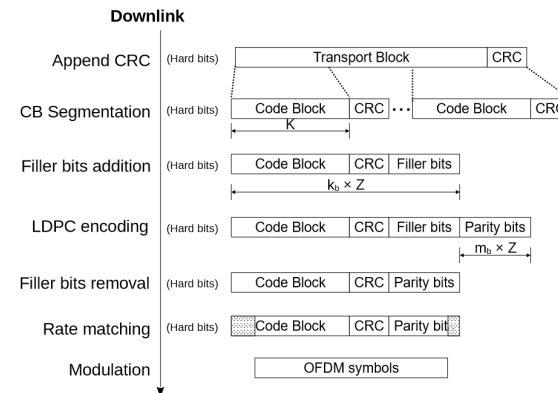
Compute
grants
DL SF $n+M$

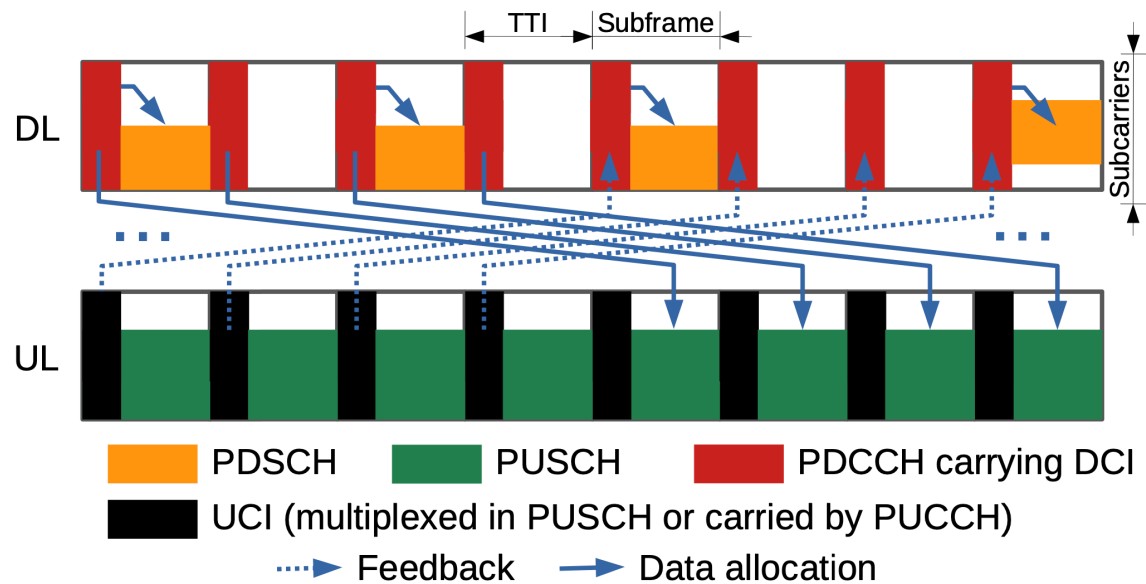


To RU

A DU has to perform many tasks

1. Receive Uplink (UL) subframe (SF) n (OFDM symbols, after FFT)
2. Process UL data channels in UL SF n
3. Process UL control channels in UL SF n
4. Prepare Downlink (DL) SF $n + M$ ($M=4$)
5. Process DL data channels in DL SF $n + M$
6. Process DL control channels in DL SF $n + M$
7. Send DL SF $n+M$ to RU (to perform IFFT)

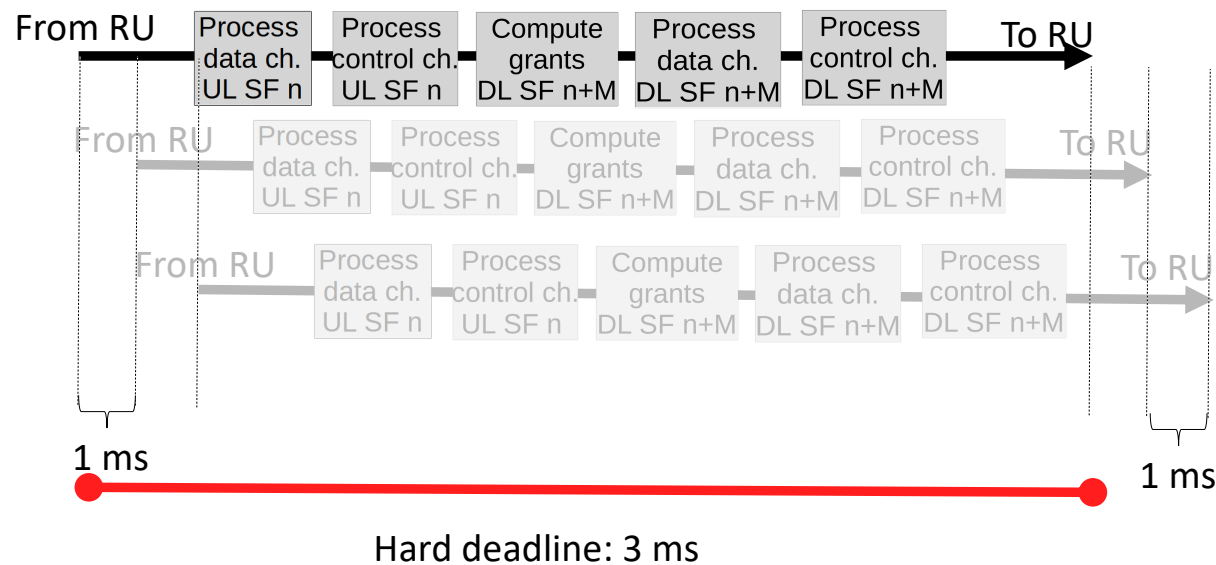




- DL and UL grants -> Downlink Control Information (DCI)
- HARQ feedback -> UL Control Information (UCI)

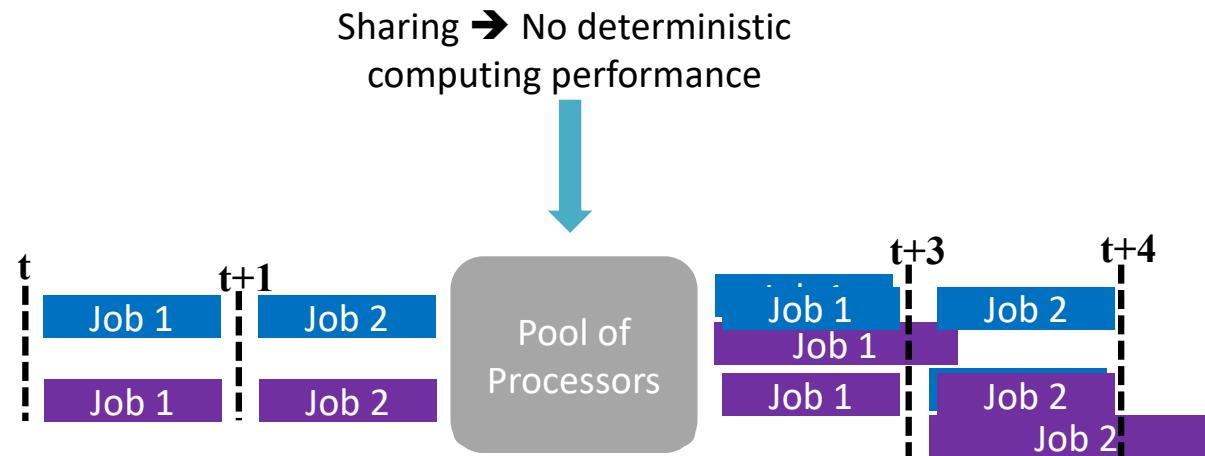
Timing is critical

- Tight deadline to process each DU job
 - Otherwise sync is lost



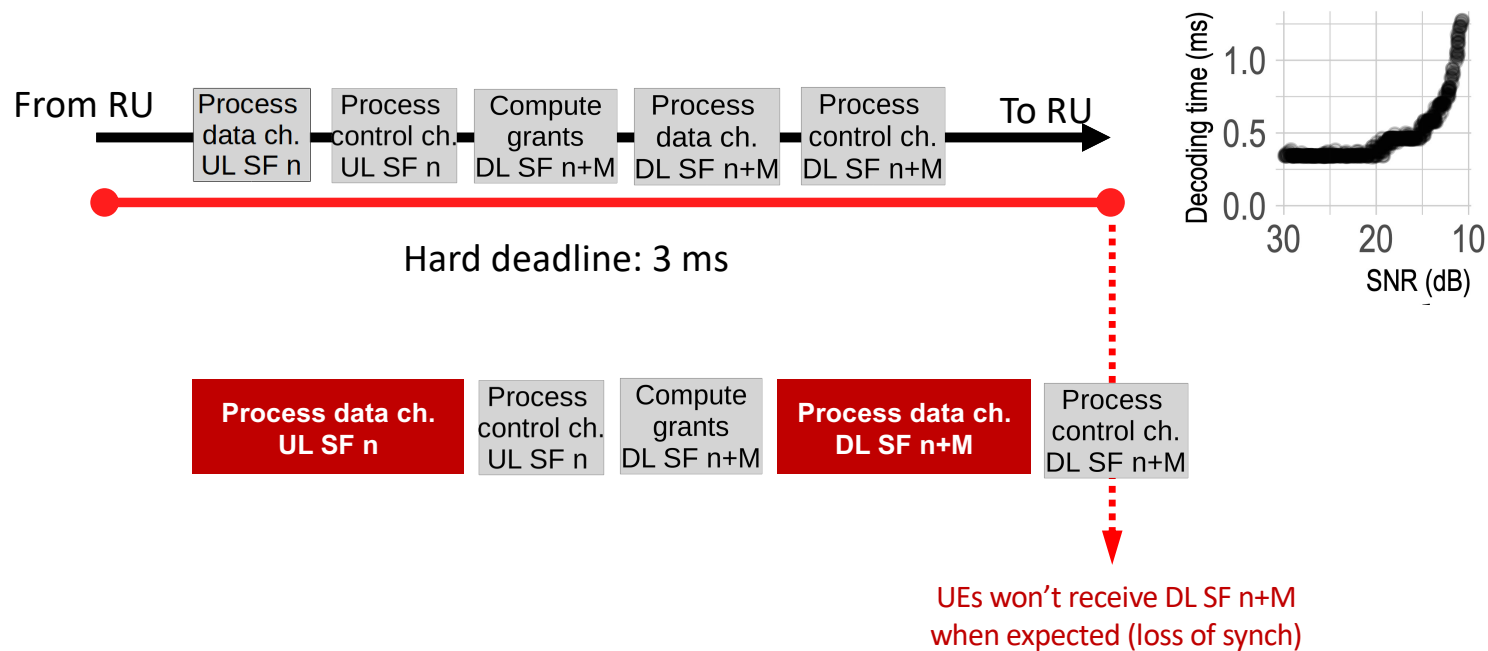
Dedlines and Shared resources

- Virtualizing a base station (eNB/gNB) is hard
 - Distributed Unit (DU) pipeline has tight computing deadlines
 - Violating deadlines loses UE-DU synchronization (network collapse)



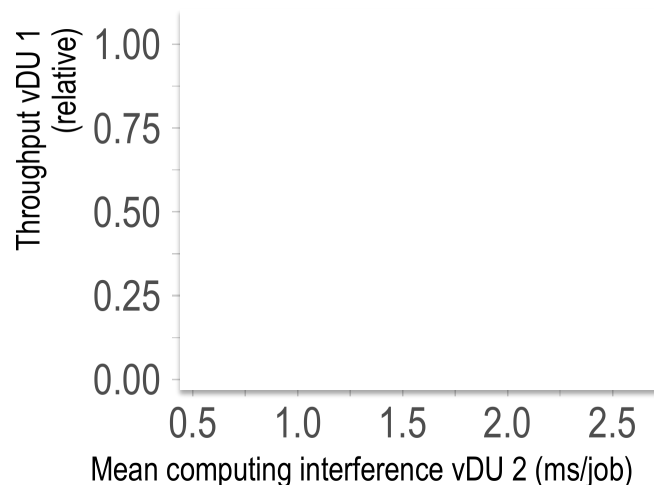
Challenge x2

- Variable capacity **and** variable demand



Toy experiment

- 5x CPUs @ 1.9 GHz, 2x vDUs sharing platform
 - vDU 1 (y-axis): Max. load uplink and downlink
 - vDU 2 (x-axis): Increasing load (noisy neigh.)

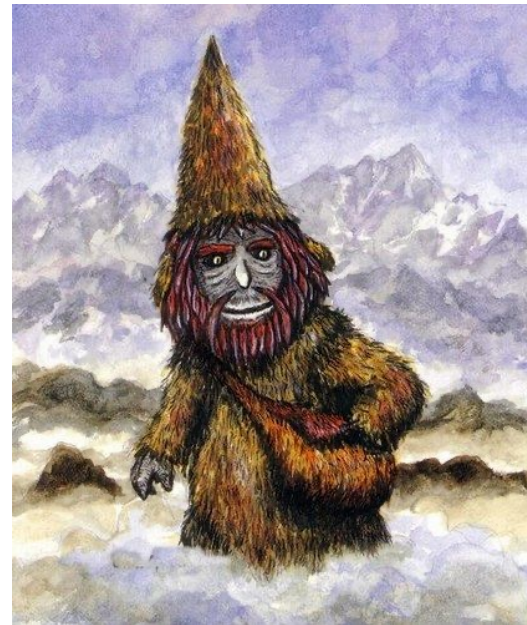


- vDU 1's throughput collapses
- Reason: Processing deadlines are violated

SOLUTION: NUBERU

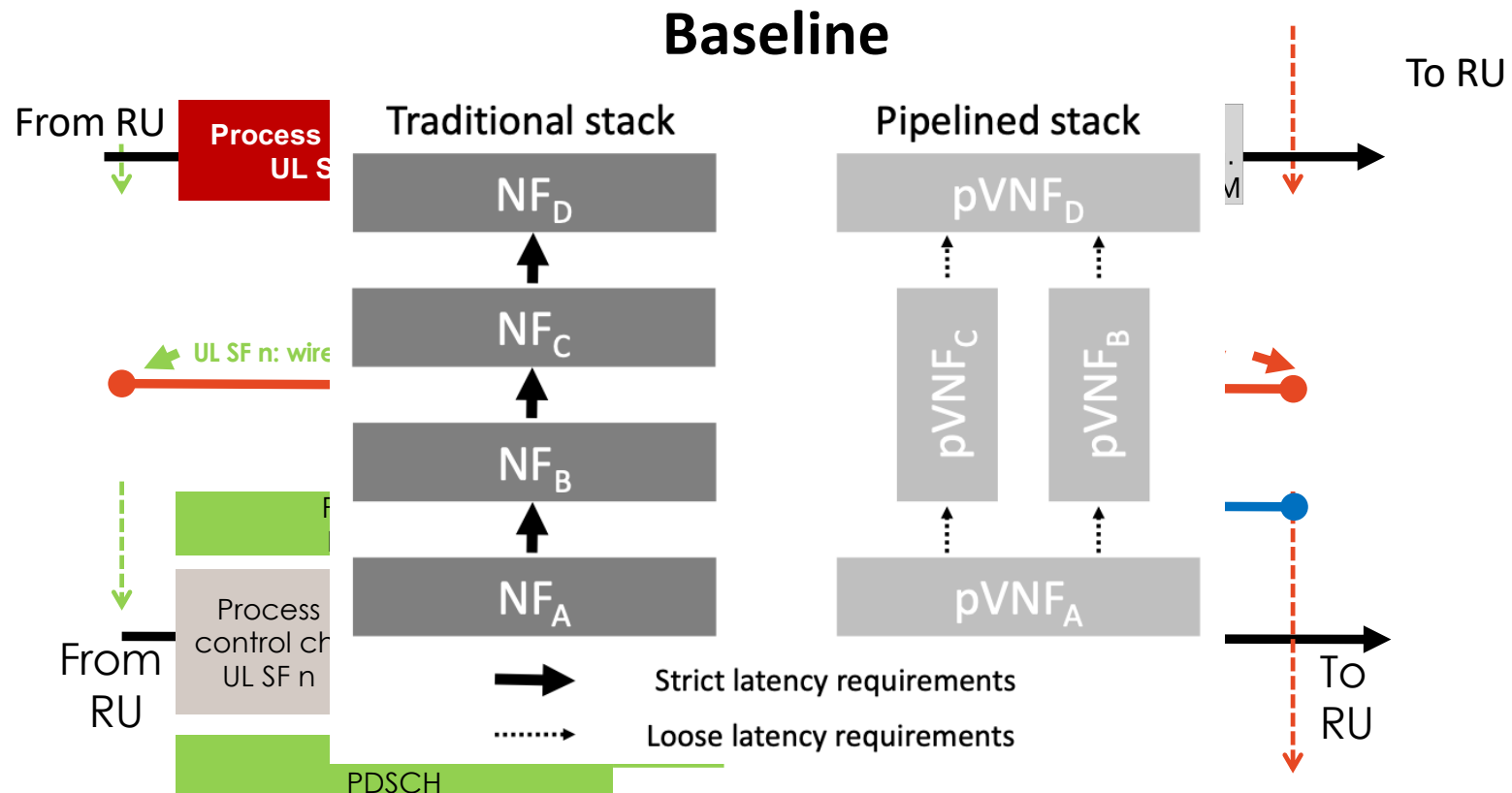
Nuberu

- "The Clouder": the divinity of clouds (and storms)



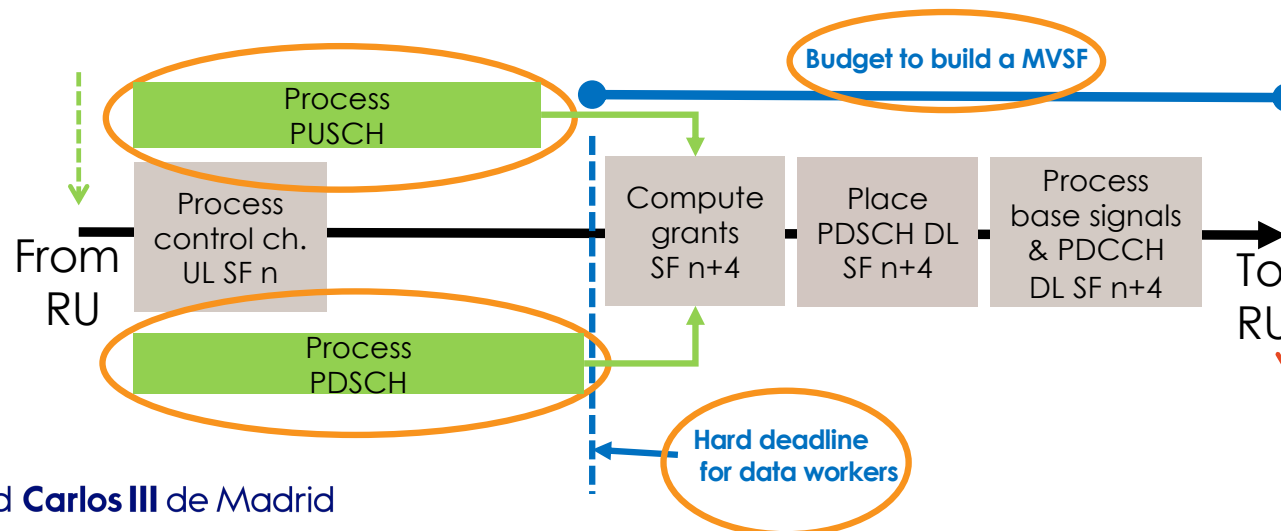
“Their appearance changes from region to region but they are usually elderly, winged, dark and **terribly ugly.**”

A resilient pipelined stack

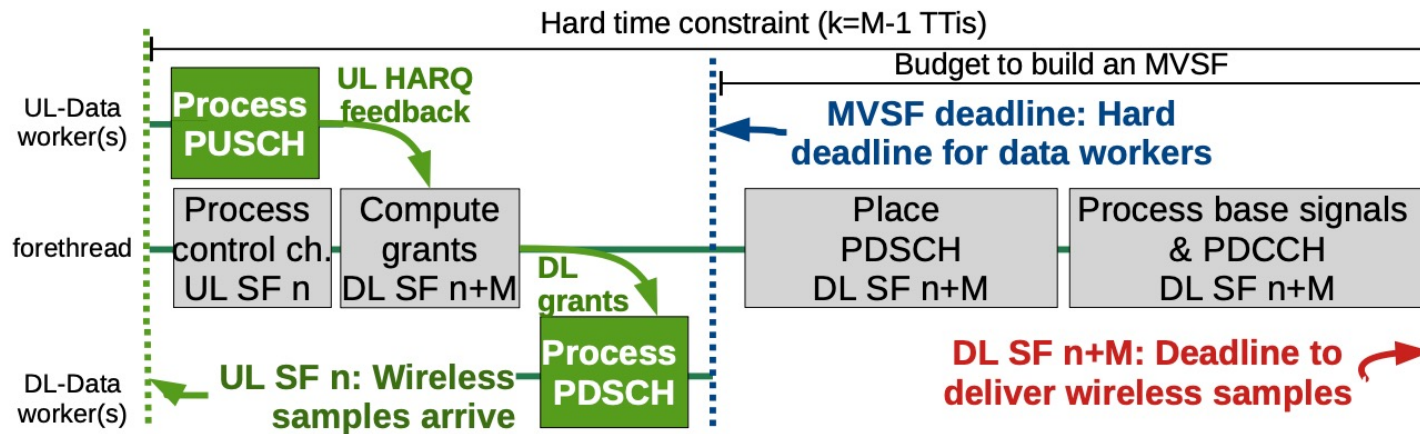


A resilient pipelined stack

- Decouple heavy tasks (PUSCH, PDSCH), which alleviates head-of-line blocking)
- Hard deadline for data processing workers
 - This guarantees sufficient residual time to build a **minimum viable SF** (MVSF), which preserves sync



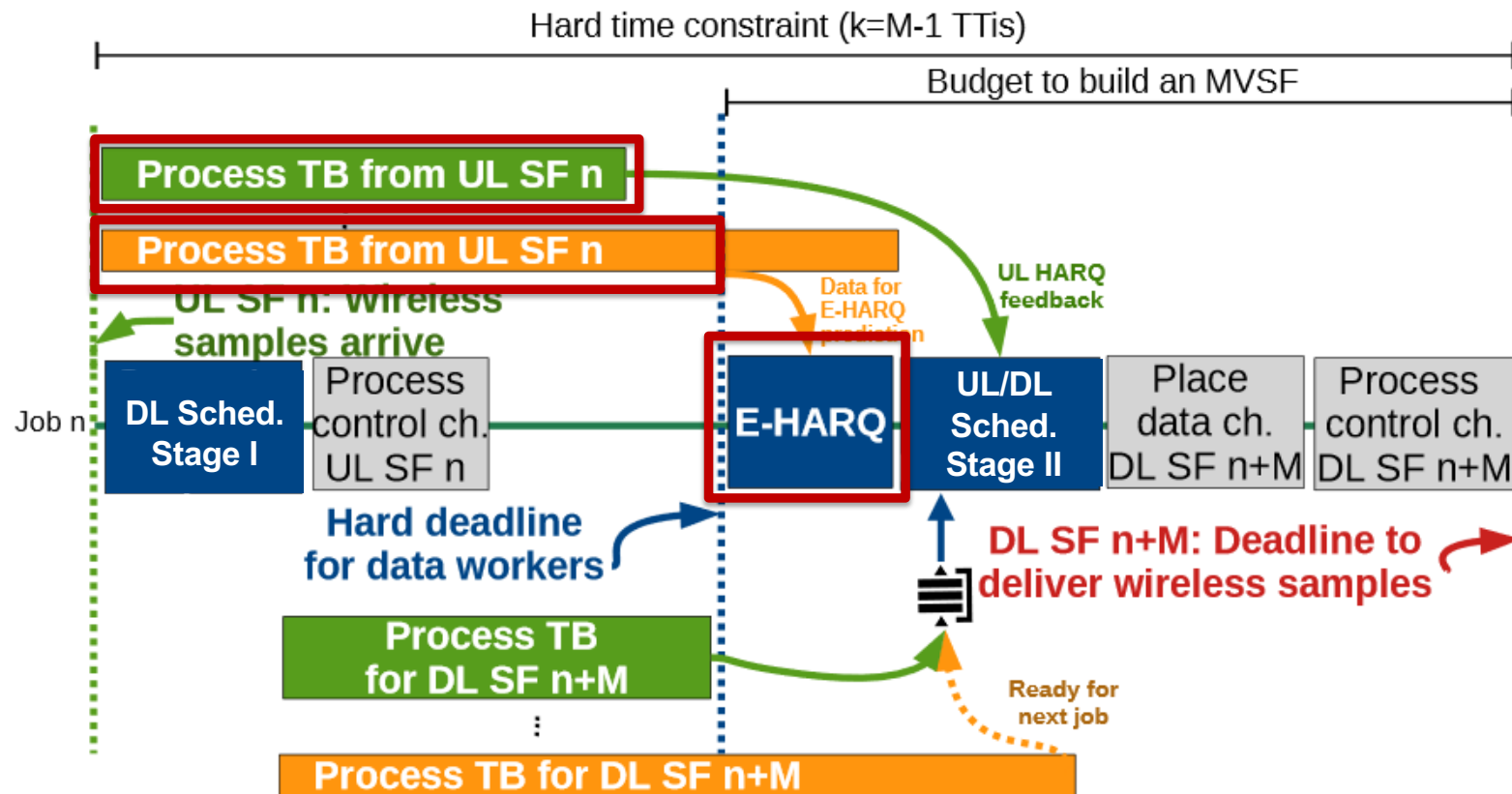
Approach: Three families of workers



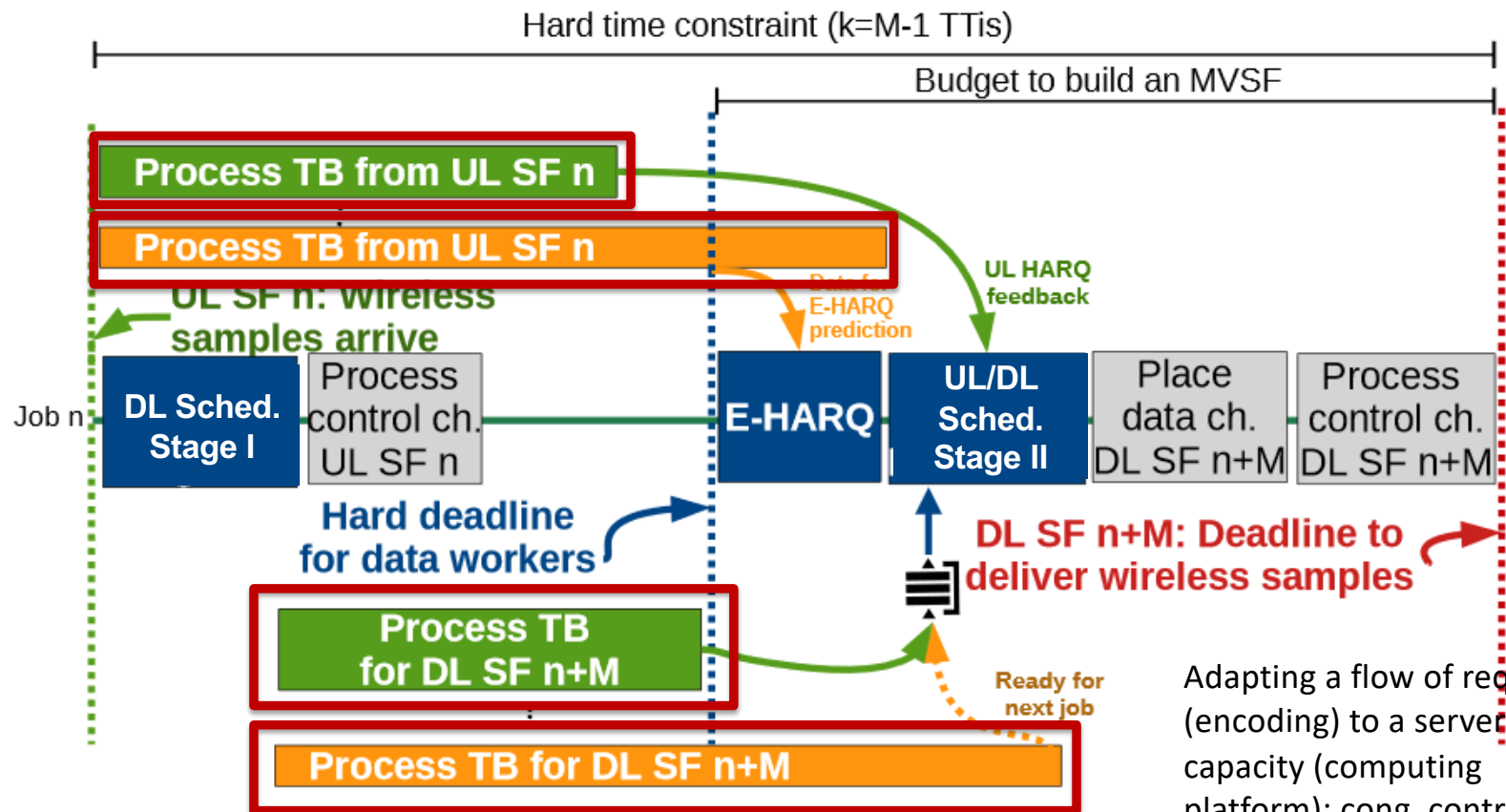
- DU forethread
 - (i) building the MVSF;
 - (ii) coordinating the remaining workers
- DL-Data DU workers: process PDSCH tasks
- UL-Data DU workers: process PUSCH tasks



UL: Early Hybrid-ARQ (E-HARQ)



Adapt DL/UL grants to capacity

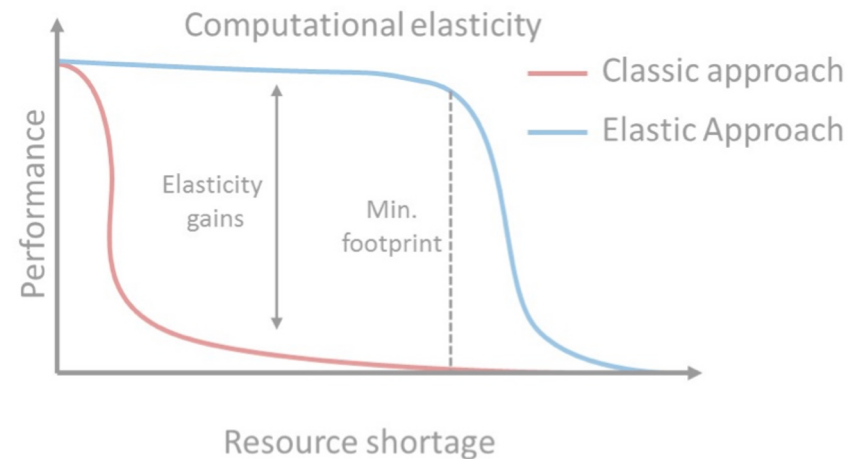
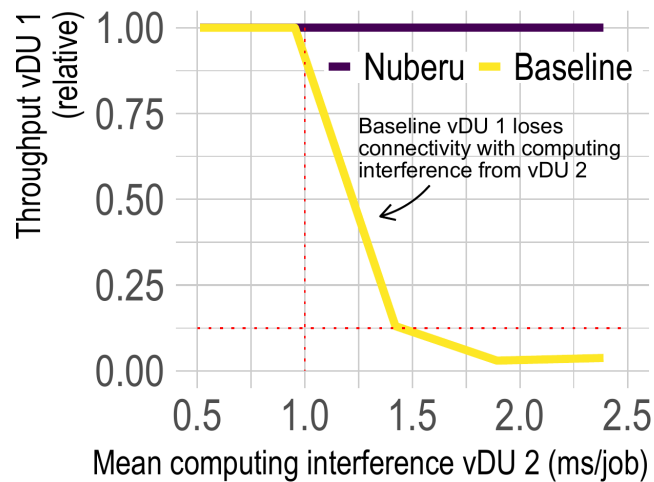


Two Congestion Control Algorithms

- Predict & schedule DL/UL grants that can be processed in time
- Simple AIMD approach with a CW that limits the size of grants
 - DL: if the buffer of encoded grants $> \lambda$ times the BS's bandwidth -> reduce CW
 - UL: Failed E-HARQ predictions -> reduce CW

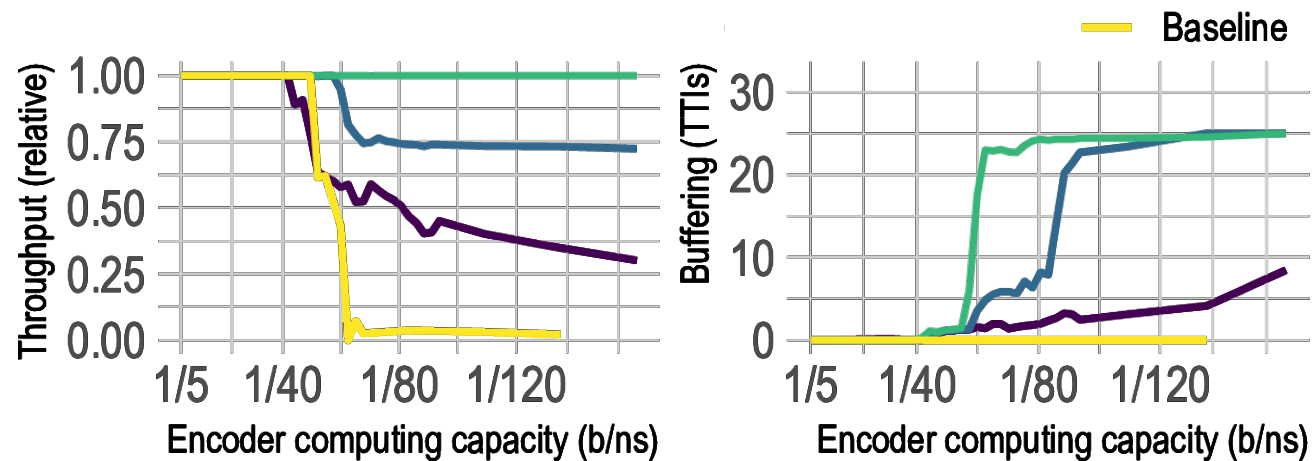
Results: Validation

- Same toy experiment as before



Throughput-delay trade-off

- Artificially slow down the CPU processor



Wrap up

- Cloud computing is already embracing microservices and serverless, while mobile networking is lagging
- There are gains, if the software is agile
- Need to re-design the VNFs (e.g., Nuberu)
- Other challenges
 - Prepare the underlying infrastructure
 - Novel orchestration approaches

Acknowledgements

- All my great co-authors!
- European Union's Horizon 2020 research and innovation programme under grant agreement no. 101015956 (Hexa-X).
- Spanish Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU through the UNICO 5G I+D SORUS projects.



Cloudifying the mobile network stack: benefits and challenges

Pablo Serrano

<http://www.it.uc3m.es/pablo/>

