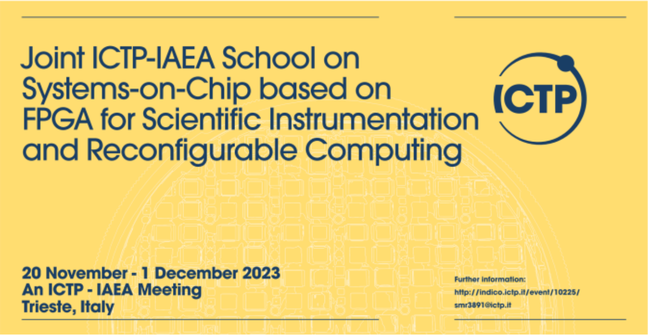
TELECOMMUNICATIONS,
COMPUTER
ENGINEERING,
AND PHOTONICS
INSTITUTE

Sant'Anna
School of Advanced Studies – Pisa



Joint ICTP-IAEA School on
Systems-on-Chip based on
FPGA for Scientific Instrumentation
and Reconfigurable Computing

20 November - 1 December 2023
An ICTP - IAEA Meeting
Trieste, Italy

Further information:
http://indico.ictp.it/event/10225/
smr3891@ictp.it

The Role of Programmable Hardware Acceleration in Virtualized Communications Networks

L. Valcarenghi

Scuola Superiore Sant'Anna, Pisa, Italy

Joint ICTP-IAEA School on Systems-on-Chip Based on FPGA for Scientific Instrumentation and Reconfigurable Computing

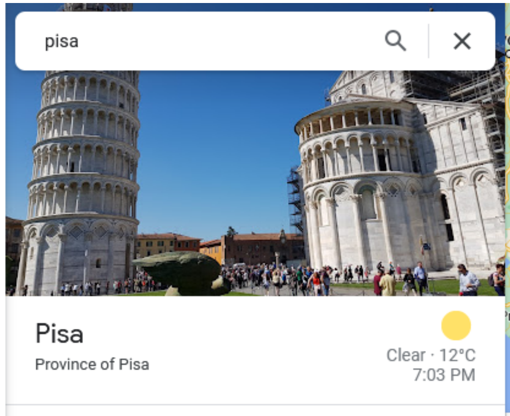
Nov. 23, 2023, Trieste, Italy

Outline

- Who we are
- Introduction to Network Function Virtualisation
- Where and how acceleration can help
- A focus on 5G protocol stack
- Results
- Conclusions

Sant’Anna School of Advanced Studies

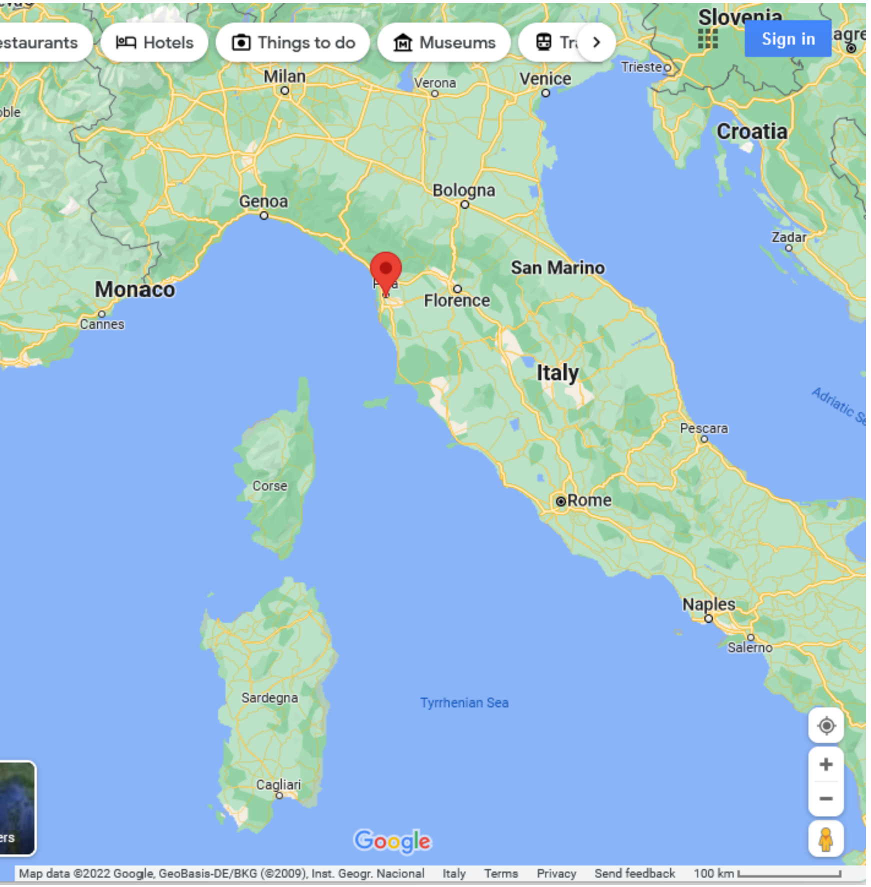
- Scuola Superiore Sant’Anna … SSSA … in “short”
- https://www.santannapisa.it/en
- https://www.youtube.com/user/ScuolaSantanna

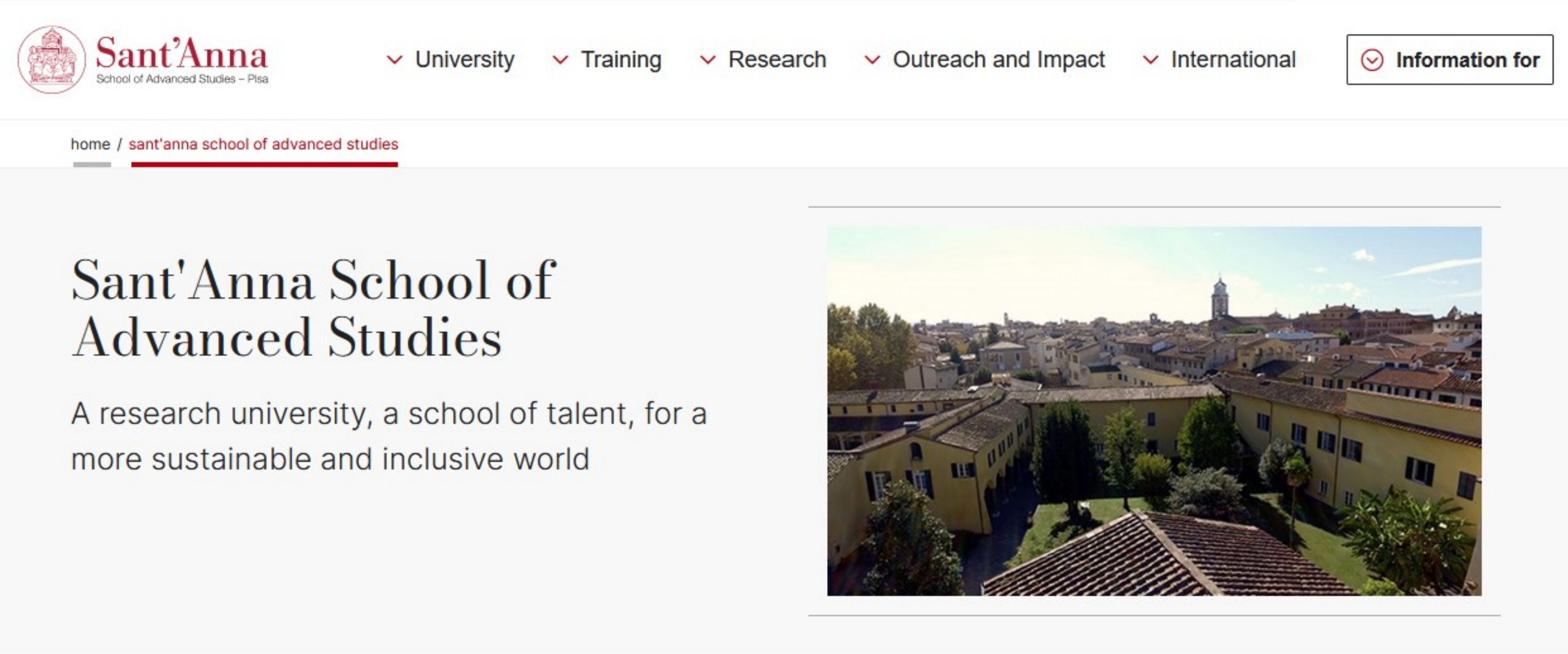


pisa

Pisa
Province of Pisa

Clear · 12°C
7:03 PM





Sant’Anna
School of Advanced Studies – Pisa

University · Training · Research · Outreach and Impact · International · Information for

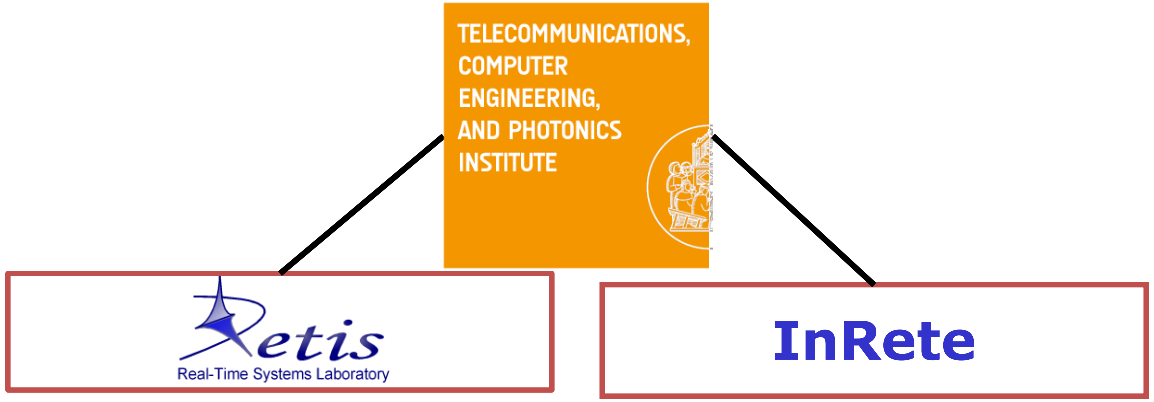
home / sant'anna school of advanced studies

Sant'Anna School of Advanced Studies

A research university, a school of talent, for a more sustainable and inclusive world

The TeCIP Institute

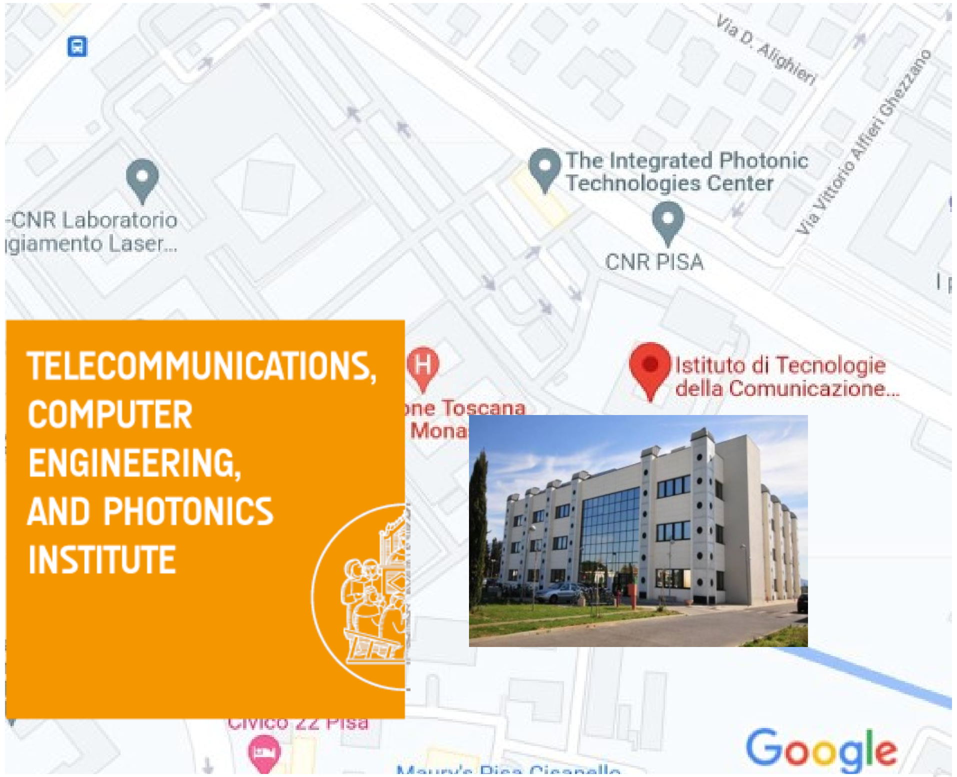
- TeCIP Institute= Telecommunications, Computer Engineering, and Photonics Institute



TELECOMMUNICATIONS, COMPUTER ENGINEERING, AND PHOTONICS INSTITUTE
Retis
Real-Time Systems Laboratory
InRete

- Scheduling algorithms
- Adaptive Resource Management
- Design of cyber-physical systems
- Open source real-time operating systems
- Methodologies and tools for the design of embedded systems
- Automatic code generation
- Real-time communication protocols
- Algorithms for energy saving management

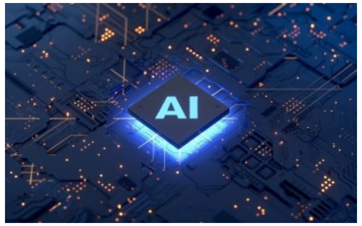
- Networks and services
- Optical communication systems
- Optical communication theory and techniques
- Digital and microwave photonics
- Optical sensors and integrated photonic subsystems
- High-capacity optical systems
- Advanced technologies for integrated photonics.

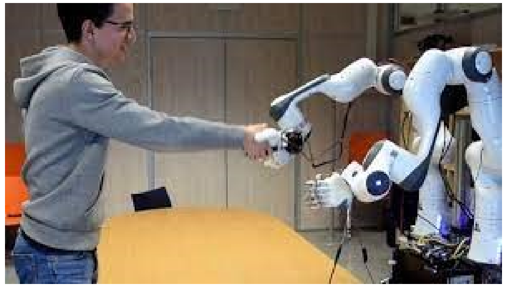


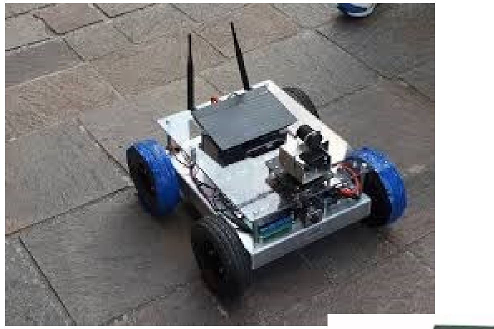
Via D. Alighieri
Via Vittorio Alfieri Ghezzano
-CNR Laboratorio
igiamento Laser...
The Integrated Photonic Technologies Center
CNR PISA
TELECOMMUNICATIONS, COMPUTER ENGINEERING, AND PHOTONICS INSTITUTE
ne Toscana
Mona
Istituto di Tecnologie della Comunicazione...
Civico 22 Pisa
Maury's Pisa Cisanello
Google

Via Giuseppe Moruzzi, 1, 56127 Pisa PI

Research


AI





Sant'A
Retis
APACHE

Photonic transceiver
DSP
RF front end

Virtualisation basic principles

- Virtualization technology emerged with the emergence of computer technology and has always played an essential role in computer technology development
- Examples
  - Introduction of the concept of virtualization in the 1950s
  - Commercialization of virtualization on mainframes by IBM in the 1960s
  - Virtual memory of the operating system
  - Java virtual machine
  - Server virtualization technology based on the x86 architecture

Huawei Technologies Co., Ltd.. (2023). Virtualization Technology. In: Cloud Computing Technology. Springer, Singapore. https://doi.org/10.1007/978-981-19-3026-3_3

Virtualisation Definition

- Virtualizing a system or component, such as a processor, memory, or an I/O device, at a given abstraction level maps its interface and visible resources onto the interface and resources of an underlying, possibly different, real system.
- Consequently, the real system appears as a different virtual system or even as multiple virtual systems.

Source: J. E. Smith and Ravi Nair, "The architecture of virtual machines," in Computer, vol. 38, no. 5, pp. 32-38, May 2005, doi: 10.1109/MC.2005.173.

Abstraction and virtualisation

- Computer systems exist and continue to evolve because they are designed as hierarchies with well-defined interfaces that separate levels of abstraction
- The operating system abstracts hard-disk addressing details (for example, that it is comprised of sectors and tracks)
- Application programmers can then create, write, and read files without knowing the hard disk's construction and physical organization

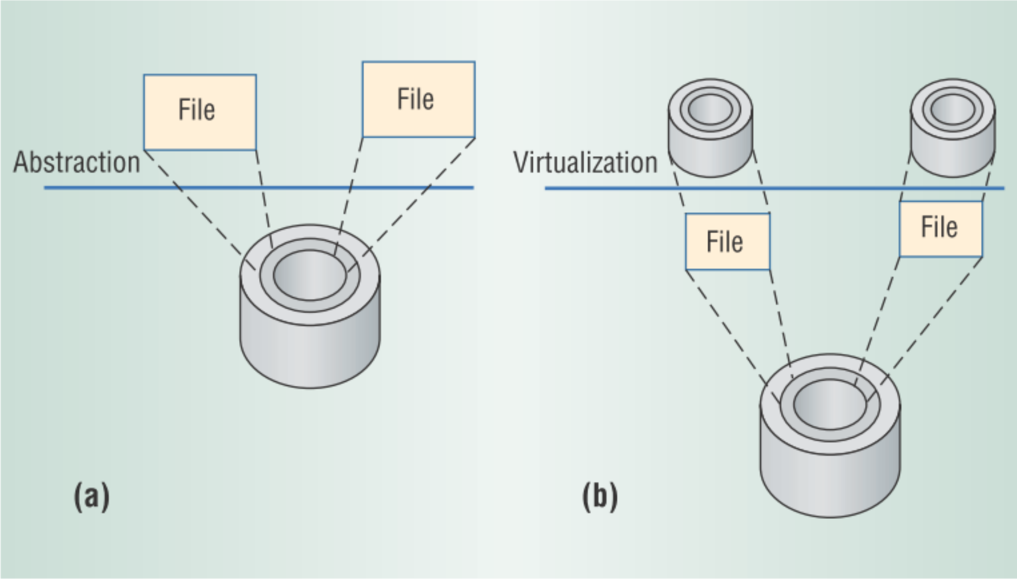


Figure 1. Abstraction and virtualization applied to disk storage. (a) Abstraction provides a simplified interface to underlying resources. (b) Virtualization provides a different interface or different resources at the same abstraction level.

Computer Instruction

- Computer's instruction set architecture (ISA) clearly exemplifies the advantages of well-defined interfaces
- For example, Intel and AMD designers develop microprocessors that implement the Intel IA-32 (x86) instruction set, while Microsoft developers write software that is compiled to the same instruction set.
- Because both groups satisfy the ISA specification, the software can be expected to execute correctly on any PC built with an IA-32 microprocessor.

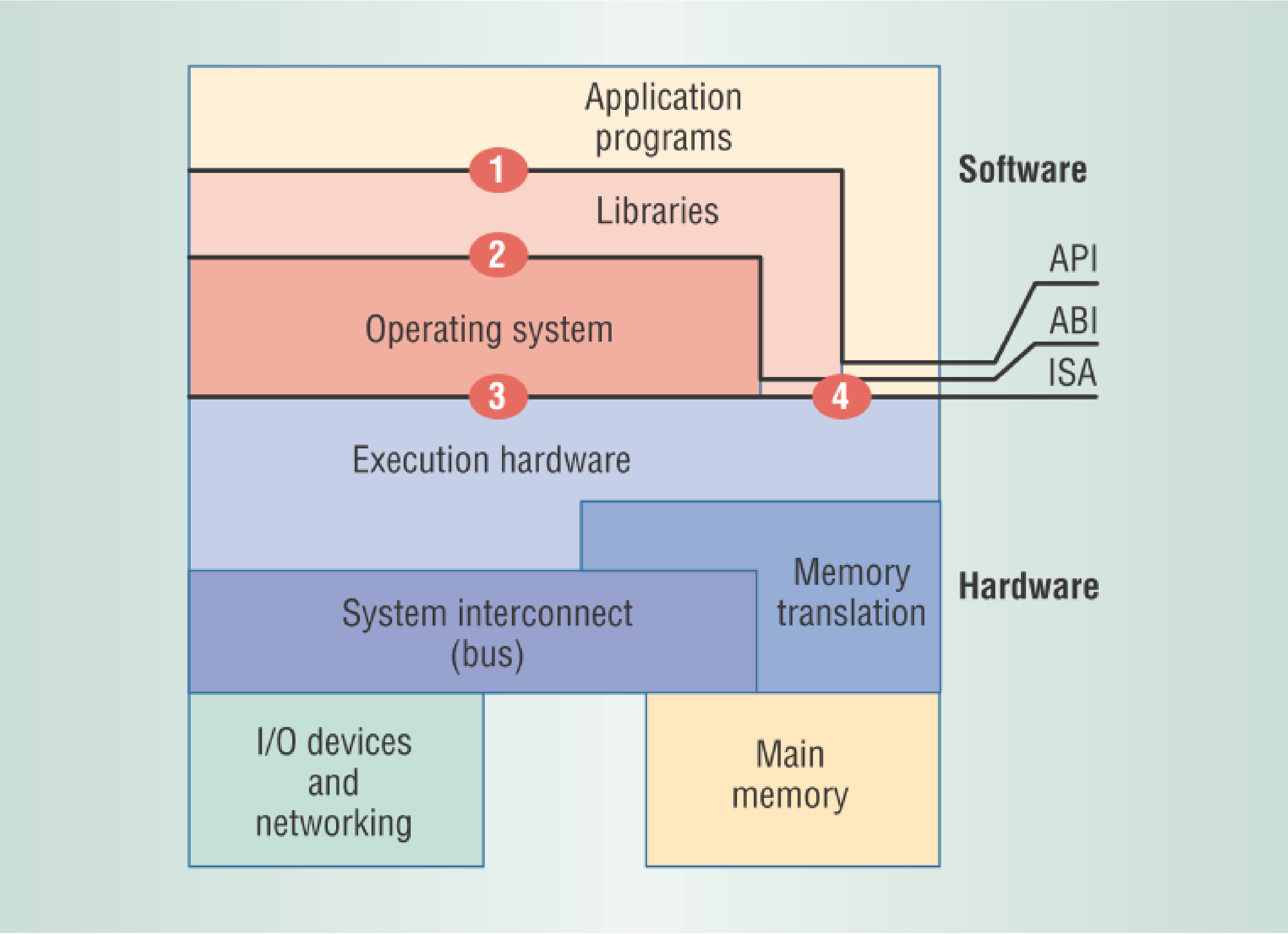


Figure 2. Computer system architecture. Key implementation layers communicate vertically via the instruction set architecture (ISA), application binary interface (ABI), and application programming interface (API).

Advantages of virtualisation

- Virtualization provides a way of getting around interface constraints.
- Virtualizing a system or component—such as a processor, memory, or an I/O device—at a given abstraction level maps its interface and visible resources onto the interface and resources of an underlying, possibly different, real system.
- Consequently, the real system appears as a different virtual system or even as multiple virtual systems.
- Unlike abstraction, virtualization does not necessarily aim to simplify or hide details.
- Virtualizing software uses the file abstraction as an intermediate step to provide a mapping between the virtual and real disks.

Other advantages of virtualisation technologies

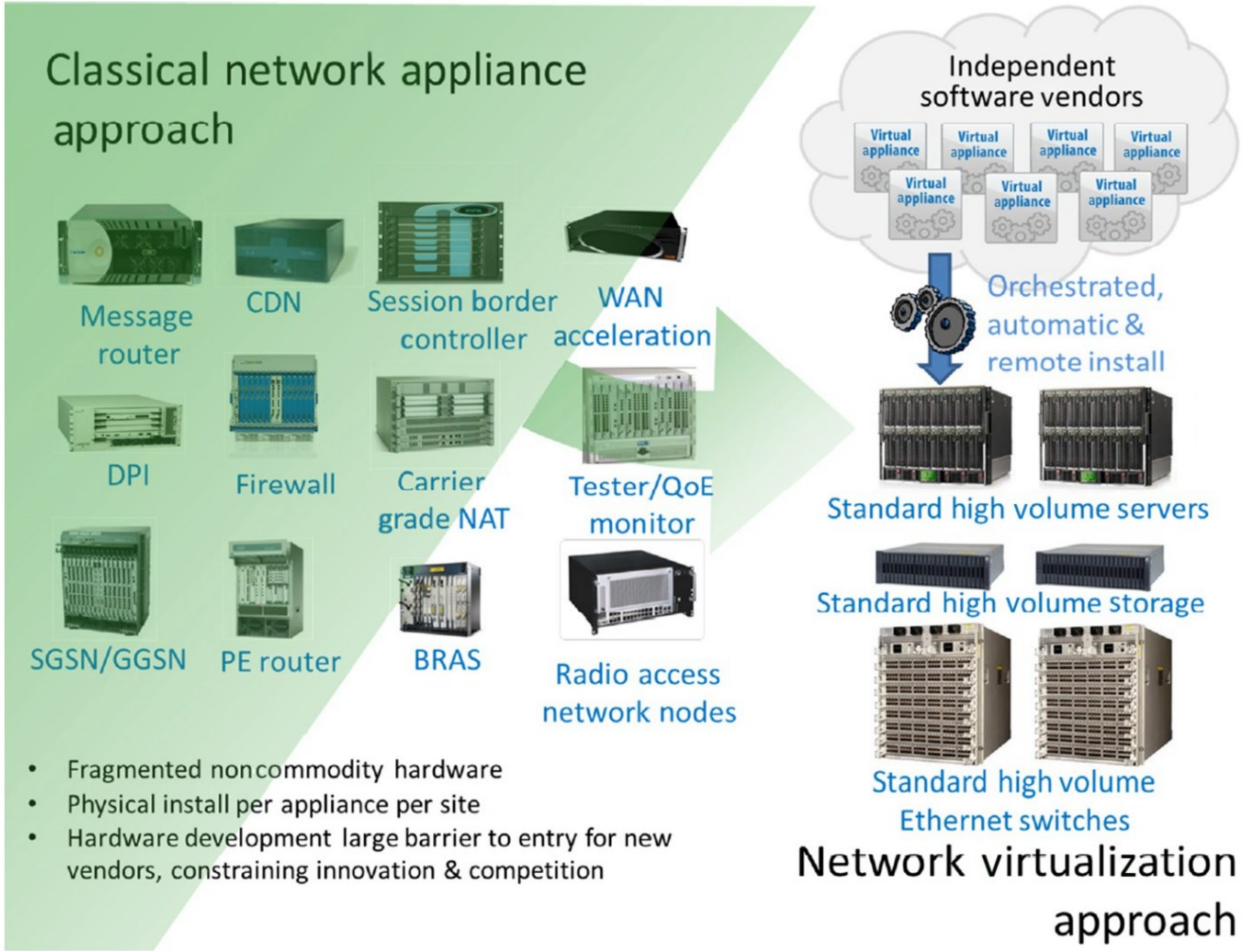
- Reduce the number of terminal equipment
  - reduce the number of managed physical resources such as servers, workstations, and other equipment
- Higher security
  - Virtualization technology can achieve isolation and division that simpler sharing mechanisms cannot achieve
  - These features can achieve controllabl and secure access to data and services
- Higher availability
  - Reduce the management of resources and processes, thereby reducing the complexity of the network management system ’ s hardware architecture.

Common type of virtualisation technologies: infrastructure virtualisation

- Network virtualization, and storage virtualization can be classified as infrastructure virtualization
- Network virtualisation
  - refers to the virtualization technology that integrates network hardware and software resources to provide users with virtual network connections.
- Storage virtualization
  - refers to providing an abstract logical view for physical storage devices.

Network Function Virtualisation (VNF)

- ETSI has created an approach to migrate a Physical Network Function (PNF) to a Virtual Network Function (VNF)
- The core concept is that those physical functions run on commodity hardware with virtualized resources driven by software
- It is the replacement of network appliance hardware with virtual machines/containers


Classical network appliance approach
Message router
CDN
Session border controller
WAN acceleration
DPI
Firewall
Carrier grade NAT
Tester/QoE monitor
SGSN/GGSN
PE router
BRAS
Radio access network nodes
Independent software vendors
Virtual appliance
Orchestrated, automatic & remote install
Standard high volume servers
Standard high volume storage
Standard high volume Ethernet switches
Network virtualization approach

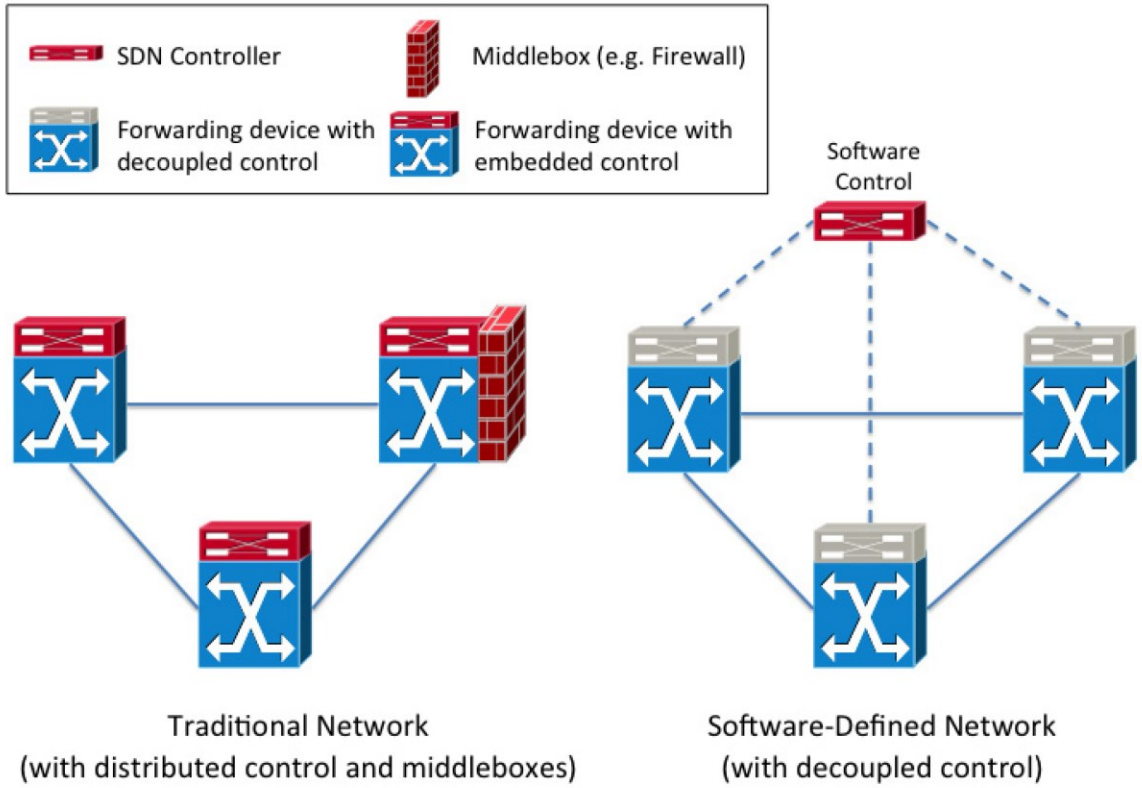
- Fragmented noncommodity hardware
- Physical install per appliance per site
- Hardware development large barrier to entry for new vendors, constraining innovation & competition

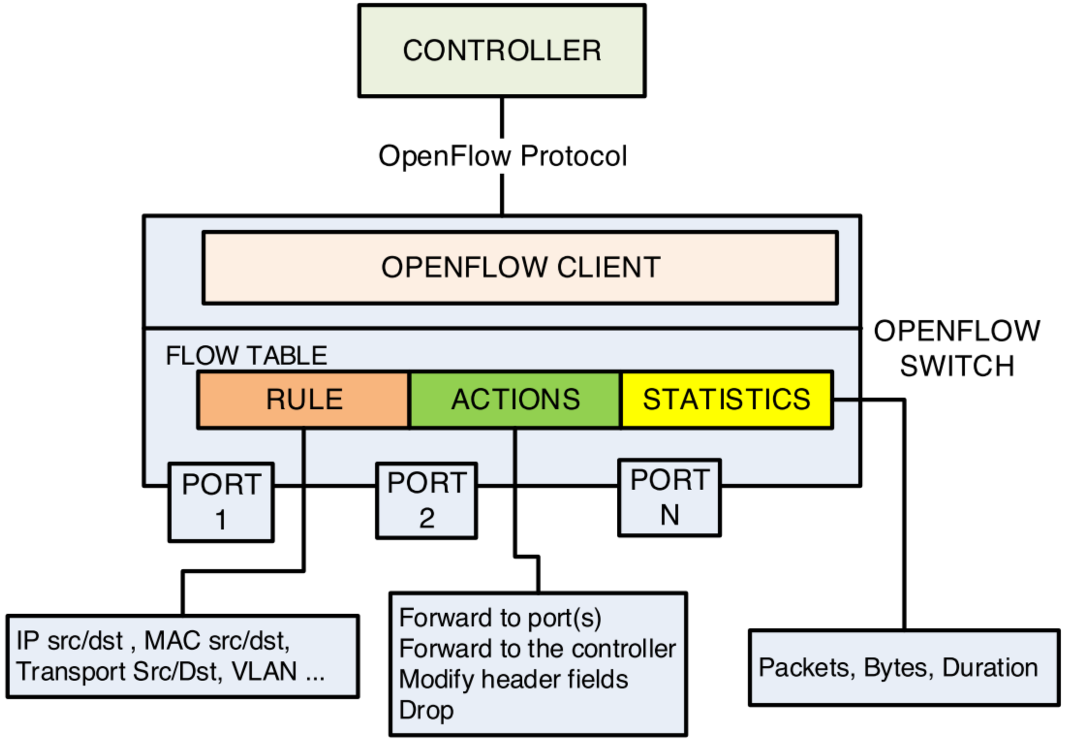
Paul Göransson, Chuck Black, Timothy Culver, Chapter 10 - Network Functions Virtualization, Editor(s): Paul Göransson, Chuck Black, Timothy Culver, Software Defined Networks (Second Edition), Morgan Kaufmann, 2017, Pages 241-252, ISBN 9780128045558, https://doi.org/10.1016/B978-0-12-804555-8.00010-7

What can we virtualize ?

- ETSI NFV use cases includes a wide range of functions
  - AR: Enterprise Access Router/Enterprise CPE
  - PE: Provider Edge Router
  - FW: Enterprise Firewall
  - NG-FW: Enterprise NG-FW
  - WOC: Enterprise WAN Optimization Controller
  - DPI: Deep Packet Inspection (Appliance or a function)
  - IPS: Intrusion Prevention System and other Security appliances
  - Network Performance Monitoring

Software Defined Networks

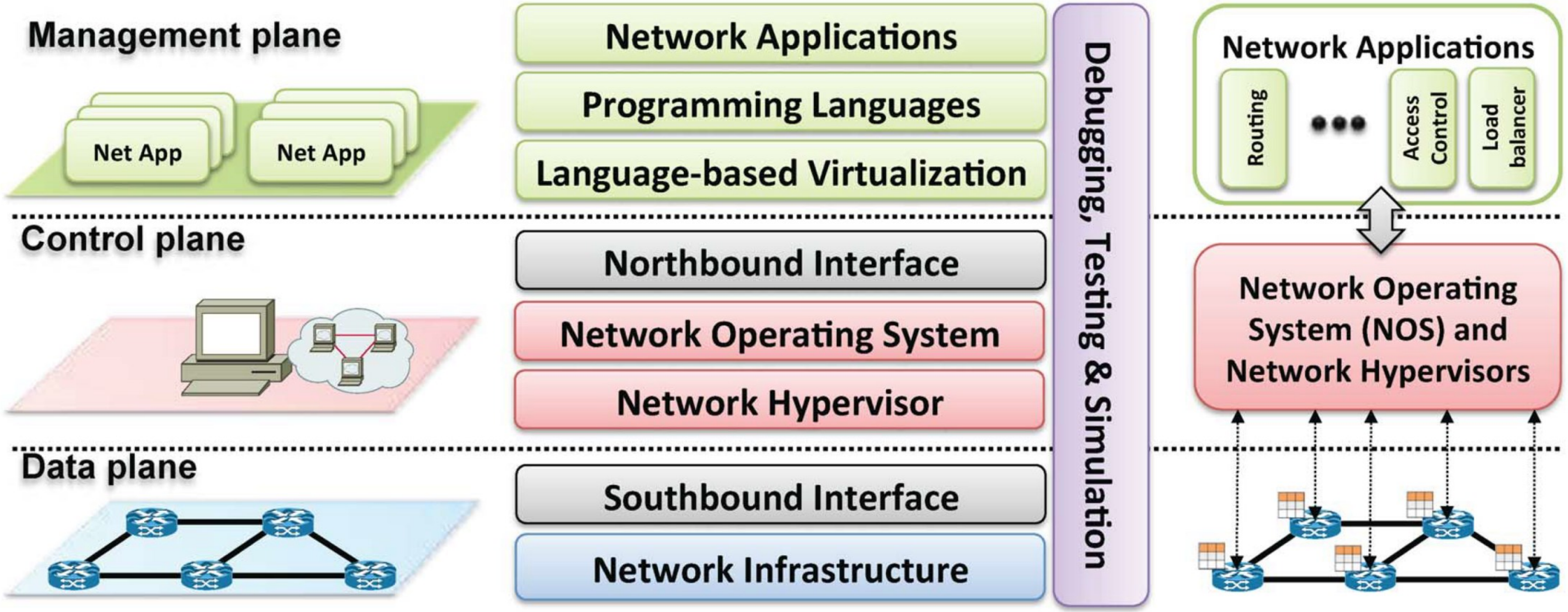

SDN Controller
Middlebox (e.g. Firewall)
Forwarding device with decoupled control
Forwarding device with embedded control
Software Control
Traditional Network
(with distributed control and middleboxes)
Software-Defined Network
(with decoupled control)



Source: B. A. A. Nunes, M. Mendonca, X. -N. Nguyen, K. Obraczka and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," in IEEE Communications Surveys & Tutorials, vol. 16, no. 3, pp. 1617-1634, Third Quarter 2014, doi: 10.1109/SURV.2014.012214.00180.

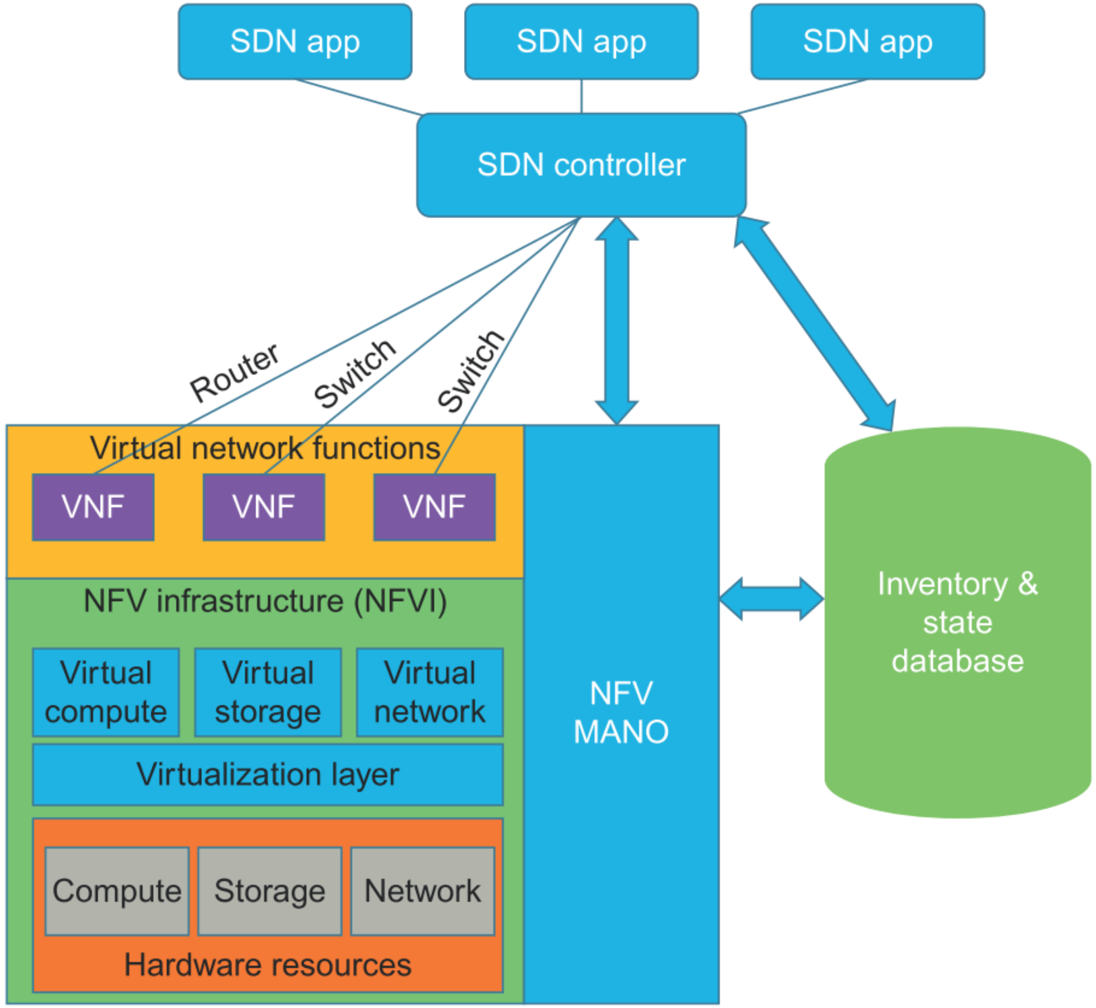
Source: D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.

SDN in different representations: planes, layers, system design architecture



Source: D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.

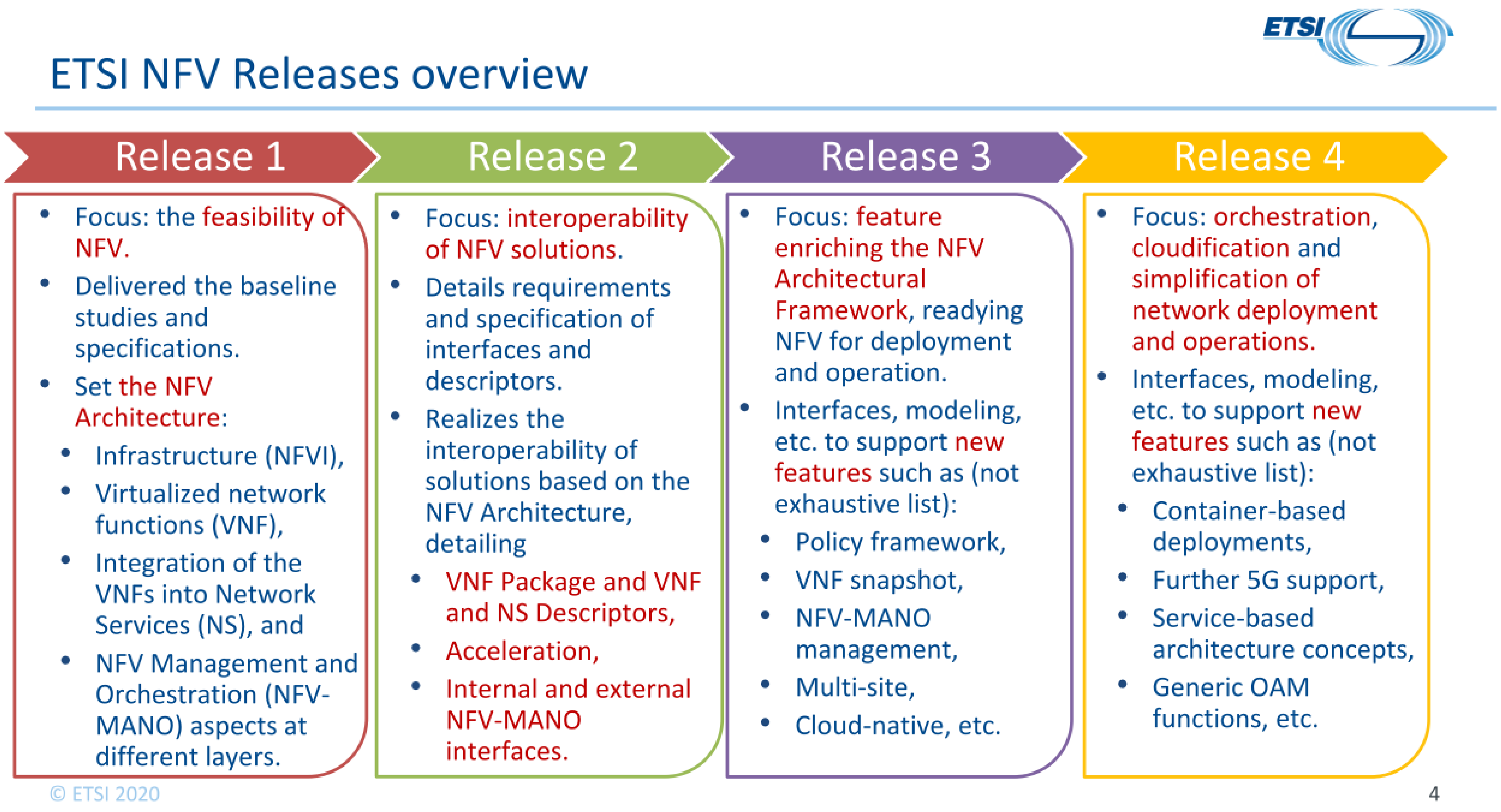
SDN and NFV


SDN app
SDN app
SDN app
SDN controller
Router
Switch
Switch
Virtual network functions
VNF
VNF
VNF
NFV infrastructure (NFVI)
Virtual compute
Virtual storage
Virtual network
Virtualization layer
Compute
Storage
Network
Hardware resources
NFV MANO
Inventory & state database

Trade off

Flexibility ⟷ **Performance**

ETSI NFV Releases



ETSI NFV Releases overview

Release 1

- Focus: the feasibility of NFV.
- Delivered the baseline studies and specifications.
- Set the NFV Architecture:
  - Infrastructure (NFVI),
  - Virtualized network functions (VNF),
  - Integration of the VNFs into Network Services (NS), and
  - NFV Management and Orchestration (NFV-MANO) aspects at different layers.

Release 2

- Focus: interoperability of NFV solutions.
- Details requirements and specification of interfaces and descriptors.
- Realizes the interoperability of solutions based on the NFV Architecture, detailing
  - VNF Package and VNF and NS Descriptors,
  - Acceleration,
  - Internal and external NFV-MANO interfaces.

Release 3

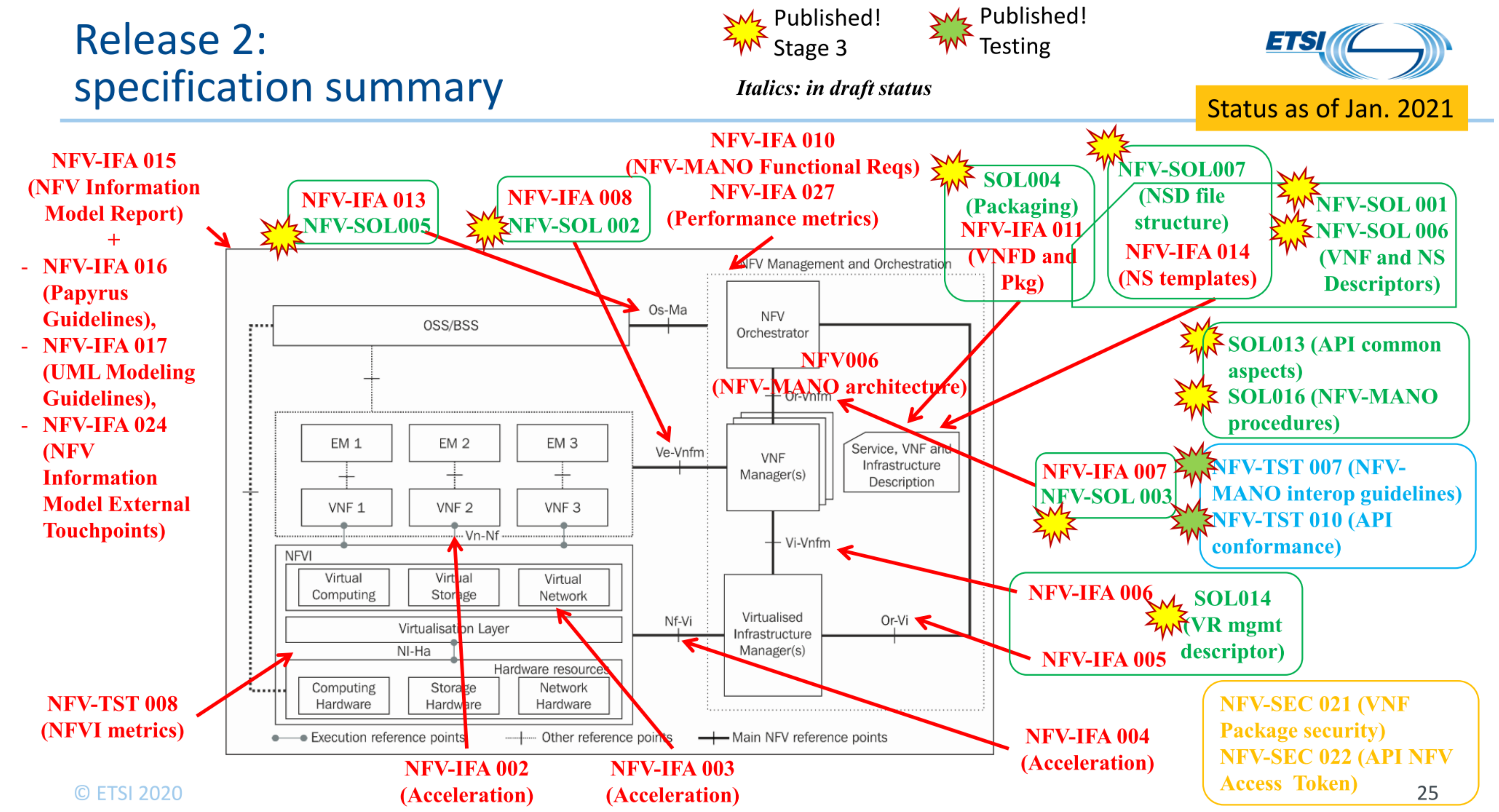
- Focus: feature enriching the NFV Architectural Framework, readying NFV for deployment and operation.
- Interfaces, modeling, etc. to support new features such as (not exhaustive list):
  - Policy framework,
  - VNF snapshot,
  - NFV-MANO management,
  - Multi-site,
  - Cloud-native, etc.

Release 4

- Focus: orchestration, cloudification and simplification of network deployment and operations.
- Interfaces, modeling, etc. to support new features such as (not exhaustive list):
  - Container-based deployments,
  - Further 5G support,
  - Service-based architecture concepts,
  - Generic OAM functions, etc.

Source:
https://docbox.etsi.org/ISG/NFV/Open/Other/ReleaseDocumentation/NFVTSC(21)000004r1_Summary_of_NFV_Releases_by_Jan_2021.pdf
https://images.app.goo.gl/KYtt5ikYpUgTk6h5A

ETSI NFV Release 2



Release 2: specification summary

Published! Stage 3

Published! Testing

Italics: in draft status

Status as of Jan. 2021

NFV-IFA 015 (NFV Information Model Report)
+
- NFV-IFA 016 (Papyrus Guidelines),
- NFV-IFA 017 (UML Modeling Guidelines),
- NFV-IFA 024 (NFV Information Model External Touchpoints)

NFV-IFA 013
NFV-SOL005

NFV-IFA 008
NFV-SOL 002

NFV-IFA 010 (NFV-MANO Functional Reqs)
NFV-IFA 027 (Performance metrics)

SOL004 (Packaging)
NFV-IFA 011 (VNFD and Pkg)

NFV-SOL007 (NSD file structure)
NFV-IFA 014 (NS templates)

NFV-SOL 001
NFV-SOL 006 (VNF and NS Descriptors)

NFV006 (NFV-MANO architecture)

SOL013 (API common aspects)
SOL016 (NFV-MANO procedures)

NFV-IFA 007
NFV-SOL 003

NFV-TST 007 (NFV-MANO interop guidelines)
NFV-TST 010 (API conformance)

NFV-IFA 006

SOL014 (VR mgmt descriptor)

NFV-IFA 005

NFV-SEC 021 (VNF Package security)
NFV-SEC 022 (API NFV Access Token)

NFV-TST 008 (NFVI metrics)

NFV-IFA 002 (Acceleration)

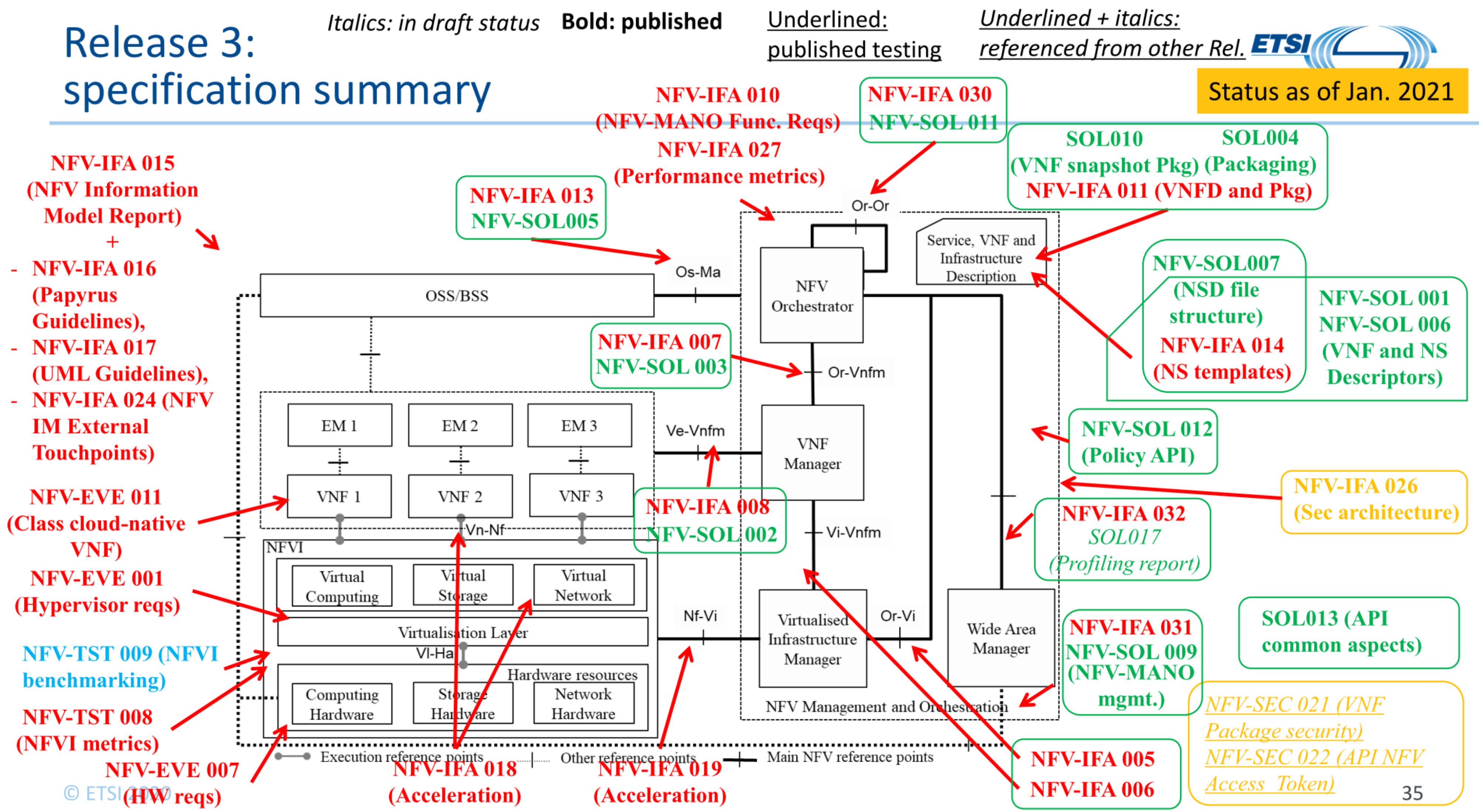
NFV-IFA 003 (Acceleration)

NFV-IFA 004 (Acceleration)

© ETSI 2020

25

ETSI NFV Release 3



Release 3: specification summary

Italics: in draft status **Bold: published** Underlined: published testing *Underlined + italics: referenced from other Rel.*

Status as of Jan. 2021

NFV-IFA 015 (NFV Information Model Report)
+
- **NFV-IFA 016 (Papyrus Guidelines),**
- **NFV-IFA 017 (UML Guidelines),**
- **NFV-IFA 024 (NFV IM External Touchpoints)**

NFV-EVE 011 (Class cloud-native VNF)

NFV-EVE 001 (Hypervisor reqs)

NFV-TST 009 (NFVI benchmarking)

NFV-TST 008 (NFVI metrics)

NFV-EVE 007 (HW reqs)

NFV-IFA 013
NFV-SOL005

NFV-IFA 010 (NFV-MANO Func. Reqs)
NFV-IFA 027 (Performance metrics)

NFV-IFA 030
NFV-SOL 011

SOL010 (VNF snapshot Pkg) **SOL004 (Packaging)**
NFV-IFA 011 (VNFD and Pkg)

NFV-SOL007 (NSD file structure)
NFV-IFA 014 (NS templates)

NFV-SOL 001
NFV-SOL 006
(VNF and NS Descriptors)

NFV-IFA 007
NFV-SOL 003

NFV-IFA 008
NFV-SOL 002

NFV-SOL 012 (Policy API)

NFV-IFA 026 (Sec architecture)

NFV-IFA 032
SOL017 (Profiling report)

NFV-IFA 031
NFV-SOL 009 (NFV-MANO mgmt.)

SOL013 (API common aspects)

NFV-SEC 021 (VNF Package security)
NFV-SEC 022 (API NFV Access Token)

NFV-IFA 018 (Acceleration)

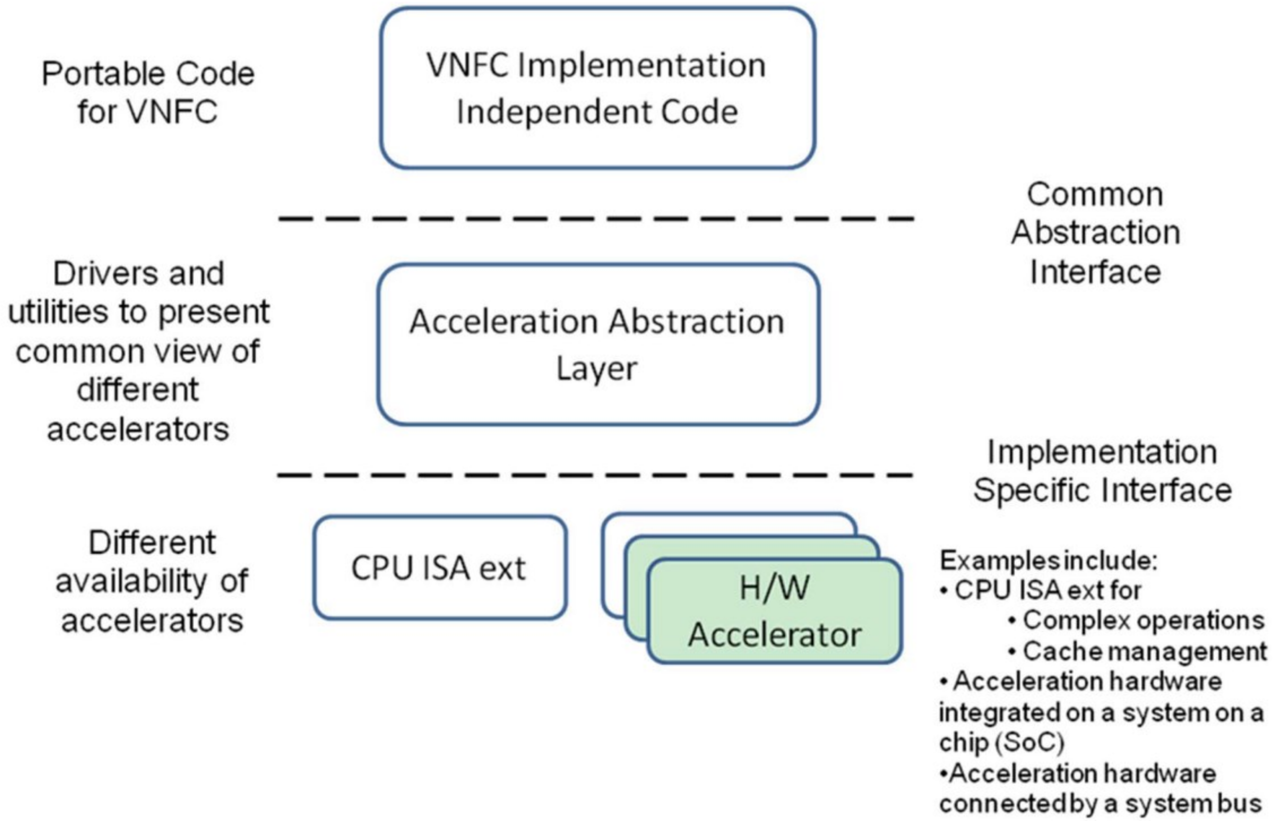
NFV-IFA 019 (Acceleration)

NFV-IFA 005
NFV-IFA 006

35

VNF Acceleration Use Cases

- Acceleration is not just about increasing performance.
- NFVI operators may seek different goals as far as acceleration is concerned:
  - Reaching the desirable performance metric at a reasonable price.
  - Best performance per processor core/cost/watt/square foot, whatever the absolute performance metric is.
  - Reaching the maximum theoretical performance level.

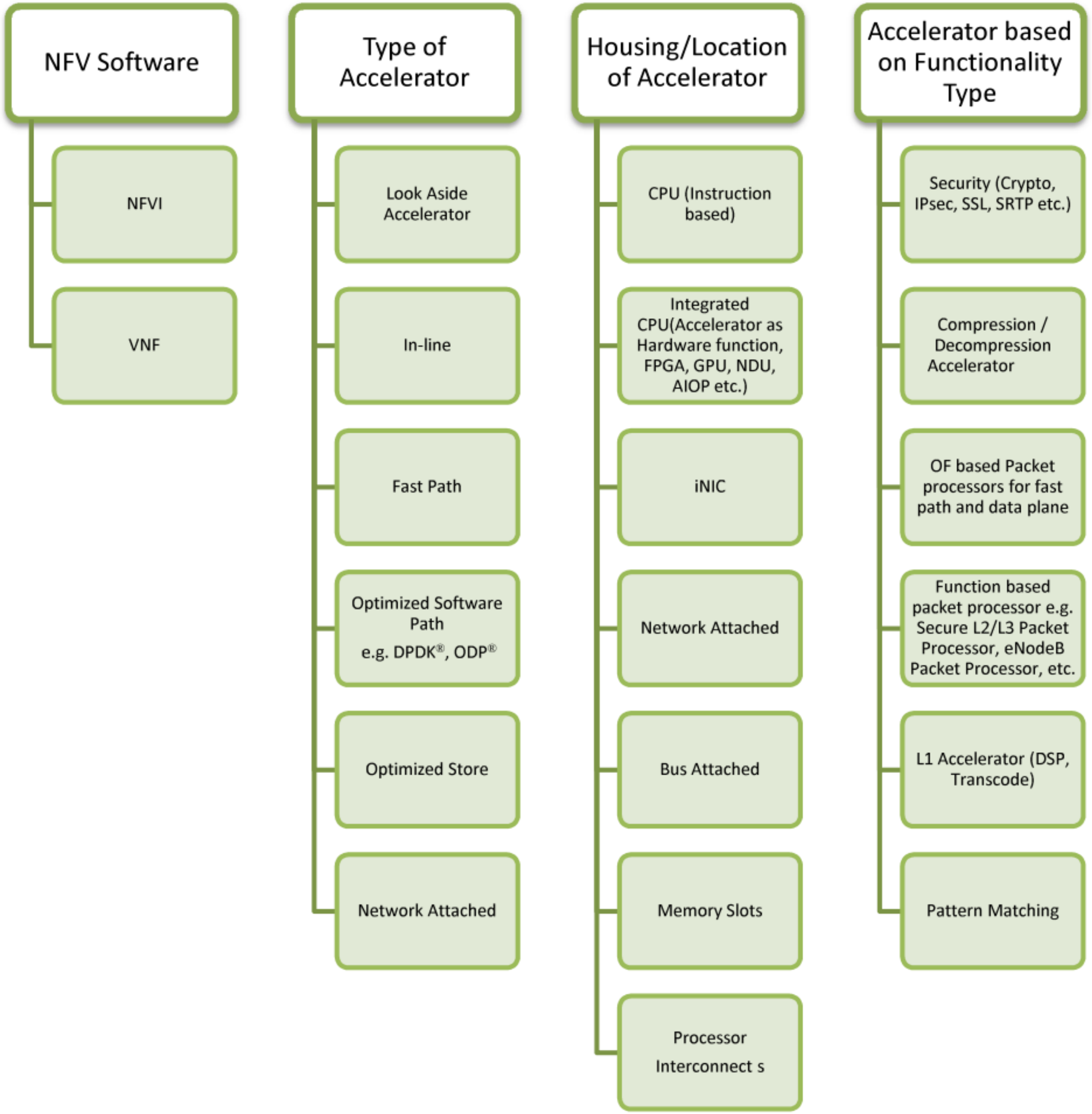

Portable Code for VNFC
VNFC Implementation Independent Code
Common Abstraction Interface
Drivers and utilities to present common view of different accelerators
Acceleration Abstraction Layer
Implementation Specific Interface
Different availability of accelerators
CPU ISA ext
H/W Accelerator

Examples include:
- CPU ISA ext for
  - Complex operations
  - Cache management
- Acceleration hardware integrated on a system on a chip (SoC)
- Acceleration hardware connected by a system bus

VNFC=Virtual Network Function Component

Source: ETSI GS NFV-IFA 001 V1.1.1 (2015-12), Network Functions Virtualisation (NFV); Acceleration Technologies; Report on Acceleration Technologies & Use Cases

Classification of accelerators



- NFV Software
  - NFVI
  - VNF
- Type of Accelerator
  - Look Aside Accelerator
  - In-line
  - Fast Path
  - Optimized Software Path e.g. DPDK®, ODP®
  - Optimized Store
  - Network Attached
- Housing/Location of Accelerator
  - CPU (Instruction based)
  - Integrated CPU(Accelerator as Hardware function, FPGA, GPU, NDU, AIOP etc.)
  - iNIC
  - Network Attached
  - Bus Attached
  - Memory Slots
  - Processor Interconnect s
- Accelerator based on Functionality Type
  - Security (Crypto, IPsec, SSL, SRTP etc.)
  - Compression / Decompression Accelerator
  - OF based Packet processors for fast path and data plane
  - Function based packet processor e.g. Secure L2/L3 Packet Processor, eNodeB Packet Processor, etc.
  - L1 Accelerator (DSP, Transcode)
  - Pattern Matching

Source: ETSI GS NFV-IFA 001 V1.1.1 (2015-12), Network Functions Virtualisation (NFV); Acceleration Technologies; Report on Acceleration Technologies & Use Cases

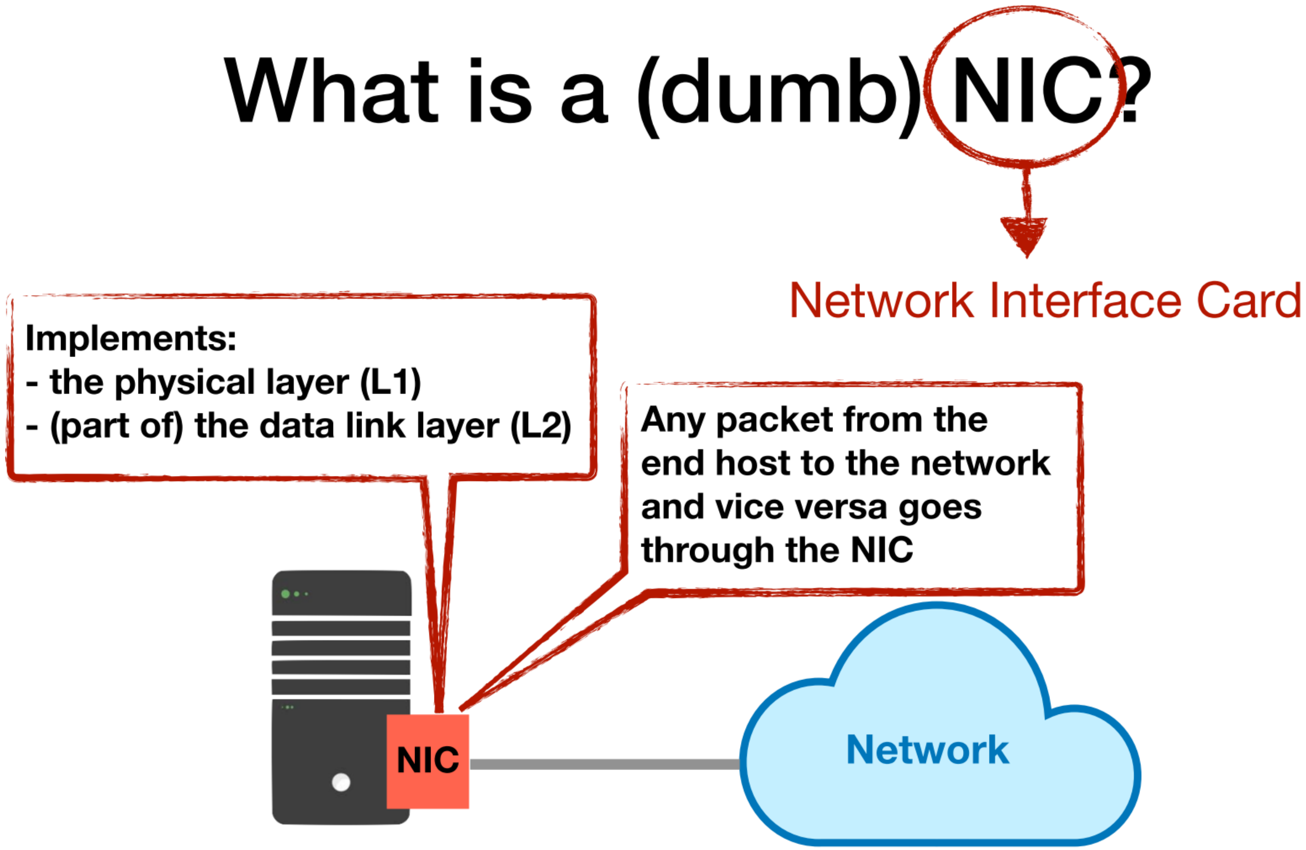
Data transfer impact for VNF offloaded into programmable hardware

- Each packet processing in a network function consists of some packet processing tasks, such as as packet parsing, classification, searching, forwarding, modification, queueing, and traffic management.
- Packet processing tasks issue memory requests to read or write the memory to understand the information of the packet, update the content of the packet, and search the databases.
- Therefore, the more memory accesses are completed in a certain period of time, the more packets are processed in the network
- For the past two decades, the communication channel between the NIC and CPU has largely remained the same
  - – issuing memory requests across a slow PCIe peripheral interconnect.
- Today, with application service times and network fabric delays measuring hundreds of nanoseconds, the NIC--CPU interface can account for most of the overhead when programming modern warehouse-scale computers.

Sources: T. Korikawa and E. Oki, "Memory Network Architecture for Packet Processing in Functions Virtualization," 2021 IEEE 7th International Conference on Network Softwarization (NetSoft), Tokyo, Japan, 2021, pp. 88-96, doi: 10.1109/NetSoft51509.2021.9492715.

Stephen Ibanez, Muhammad Shahbaz, and Nick McKeown. 2019. The Case for a Network Fast Path to the CPU. In Proceedings of the 18th ACM Workshop on Hot Topics in Networks (HotNets '19). Association for Computing Machinery, New York, NY, USA, 52–59. https://doi.org/10.1145/3365609.3365851

smartNIC as a possible solution



What is a (dumb) NIC?

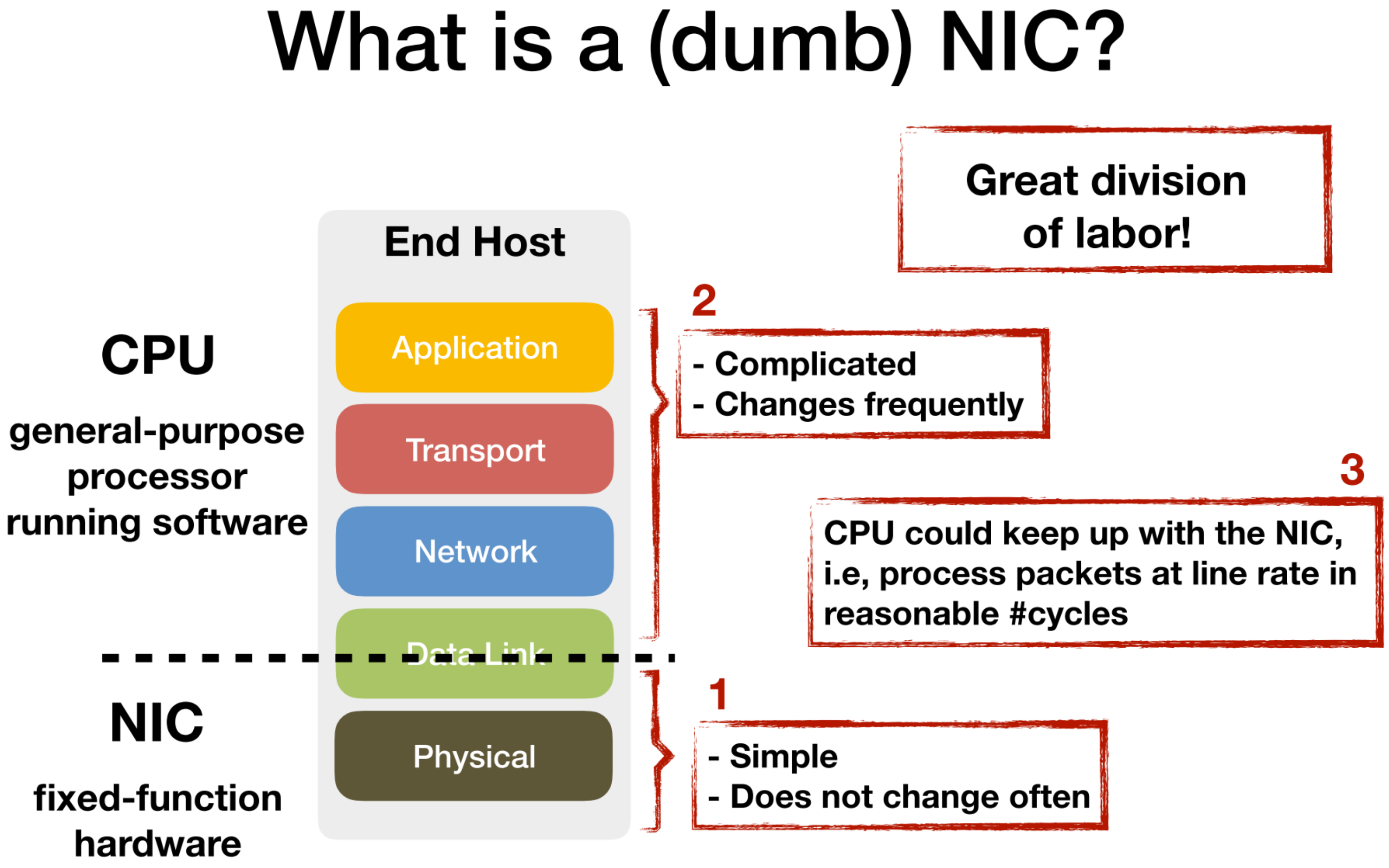
Network Interface Card

Implements:
- the physical layer (L1)
- (part of) the data link layer (L2)

Any packet from the end host to the network and vice versa goes through the NIC

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (2)



What is a (dumb) NIC?

Great division of labor!

End Host

Application

Transport

Network

Data Link

Physical

CPU

general-purpose processor running software

NIC

fixed-function hardware

2

- Complicated
- Changes frequently

3

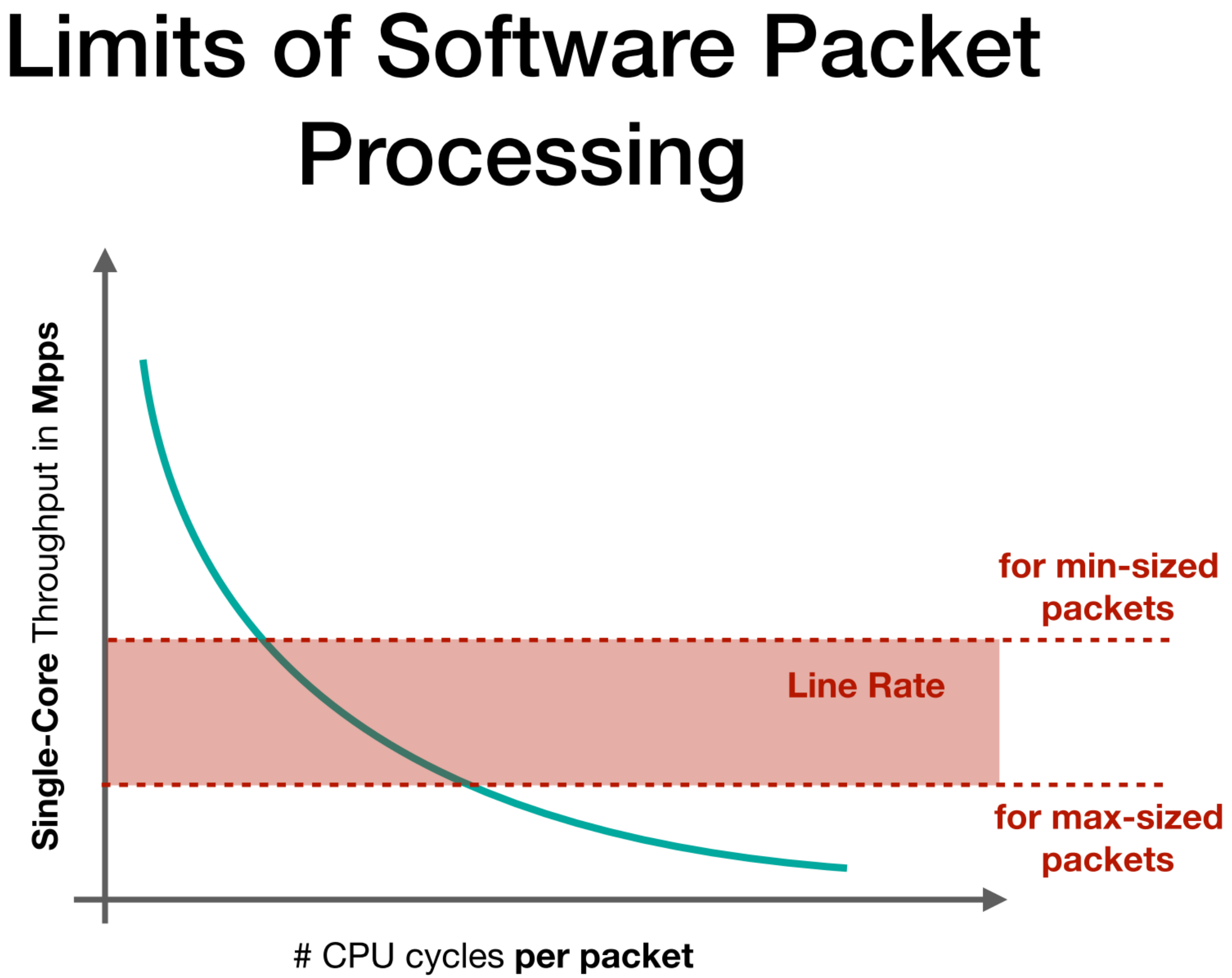
CPU could keep up with the NIC, i.e, process packets at line rate in reasonable #cycles

1

- Simple
- Does not change often

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (3)

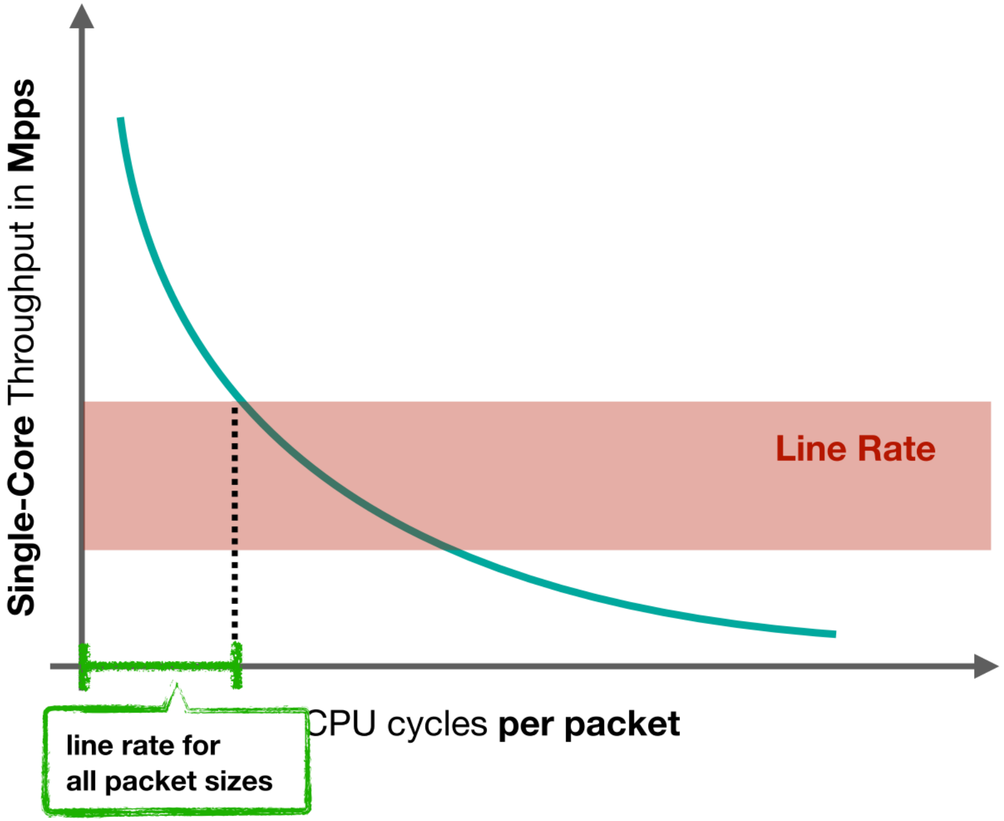


Limits of Software Packet Processing

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

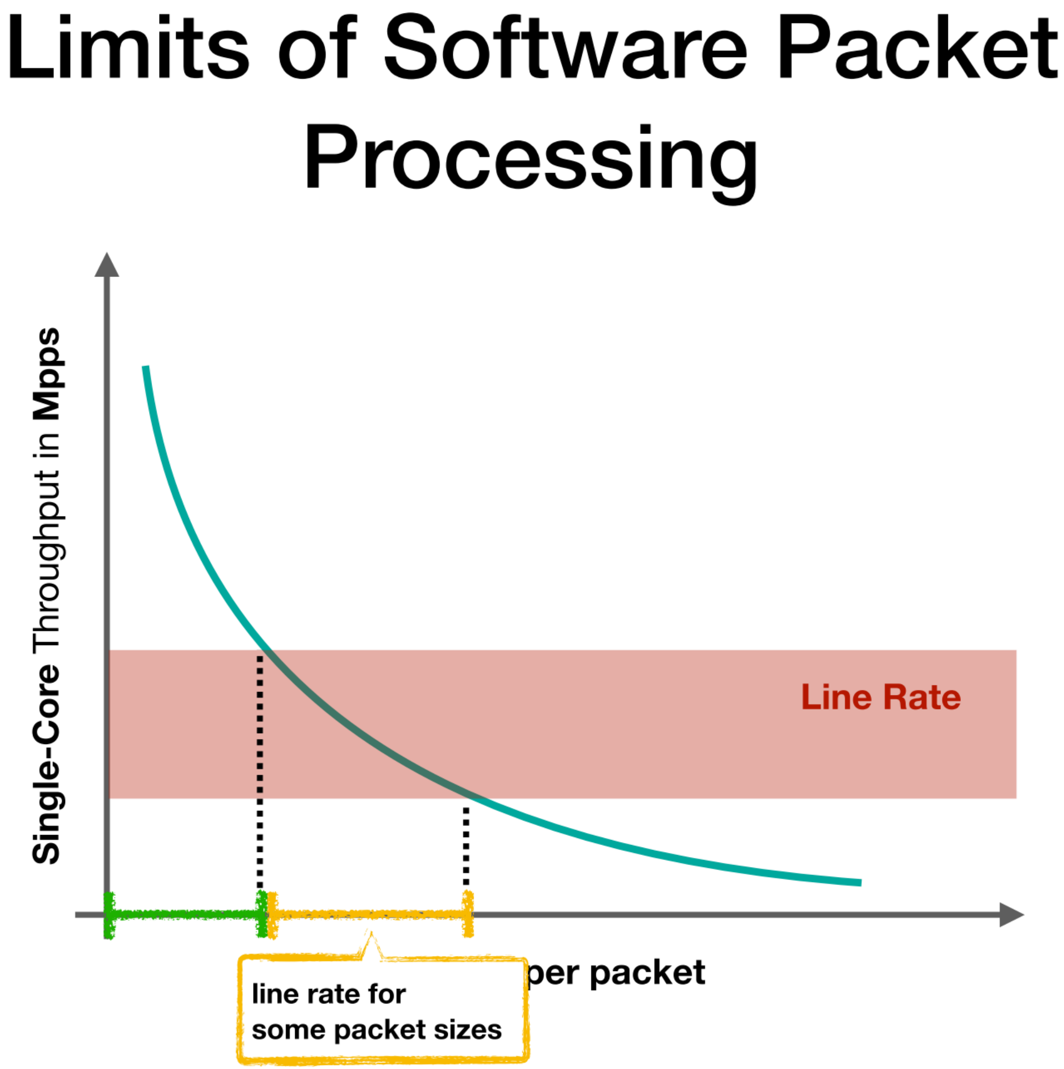
smartNIC as a possible solution (4)

Limits of Software Packet Processing



Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (5)



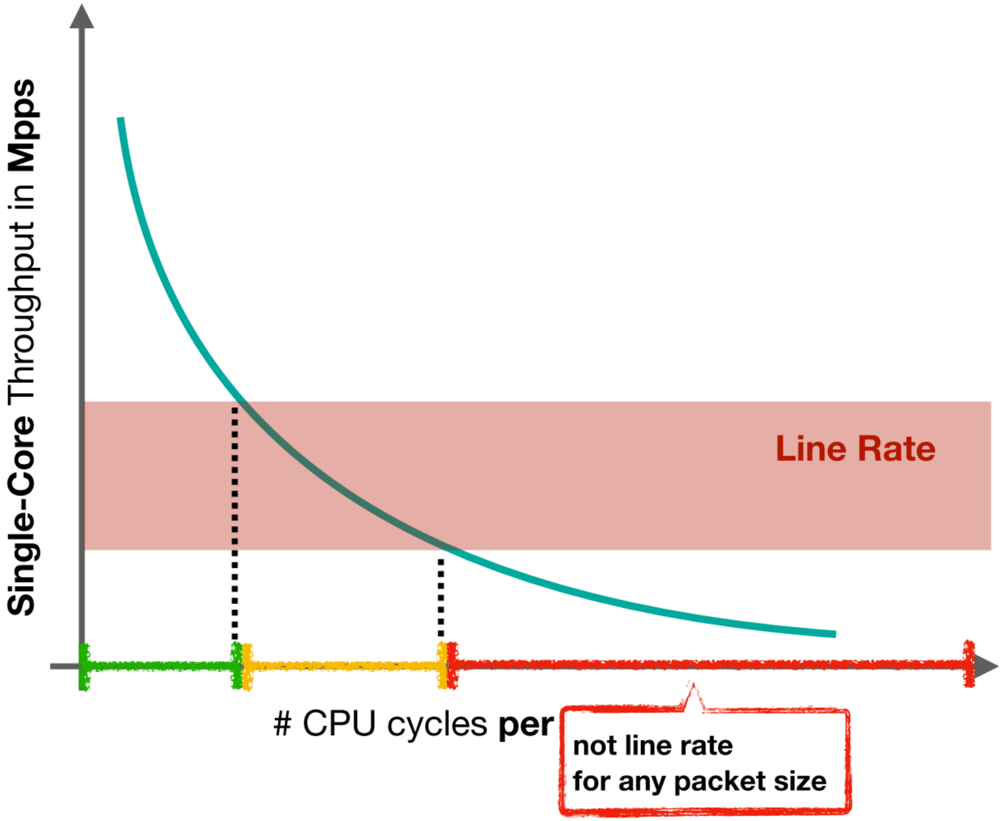
Limits of Software Packet Processing

line rate for some packet sizes

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (6)

Limits of Software Packet Processing

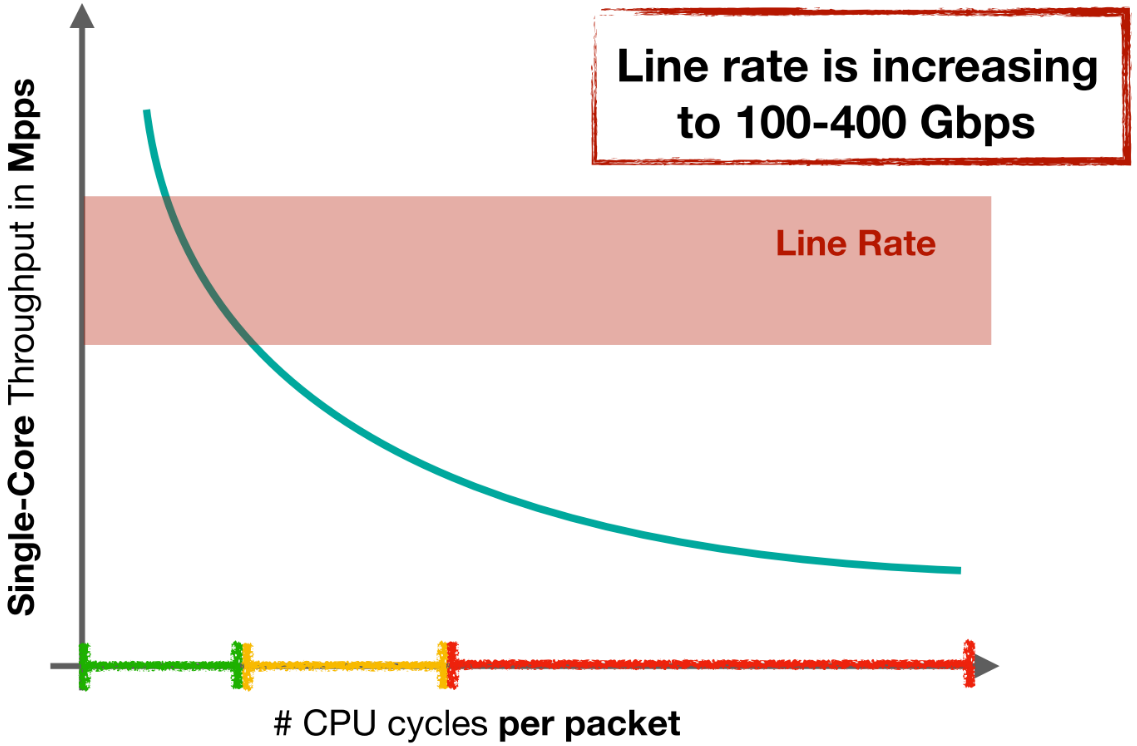


not line rate
for any packet size

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (7)

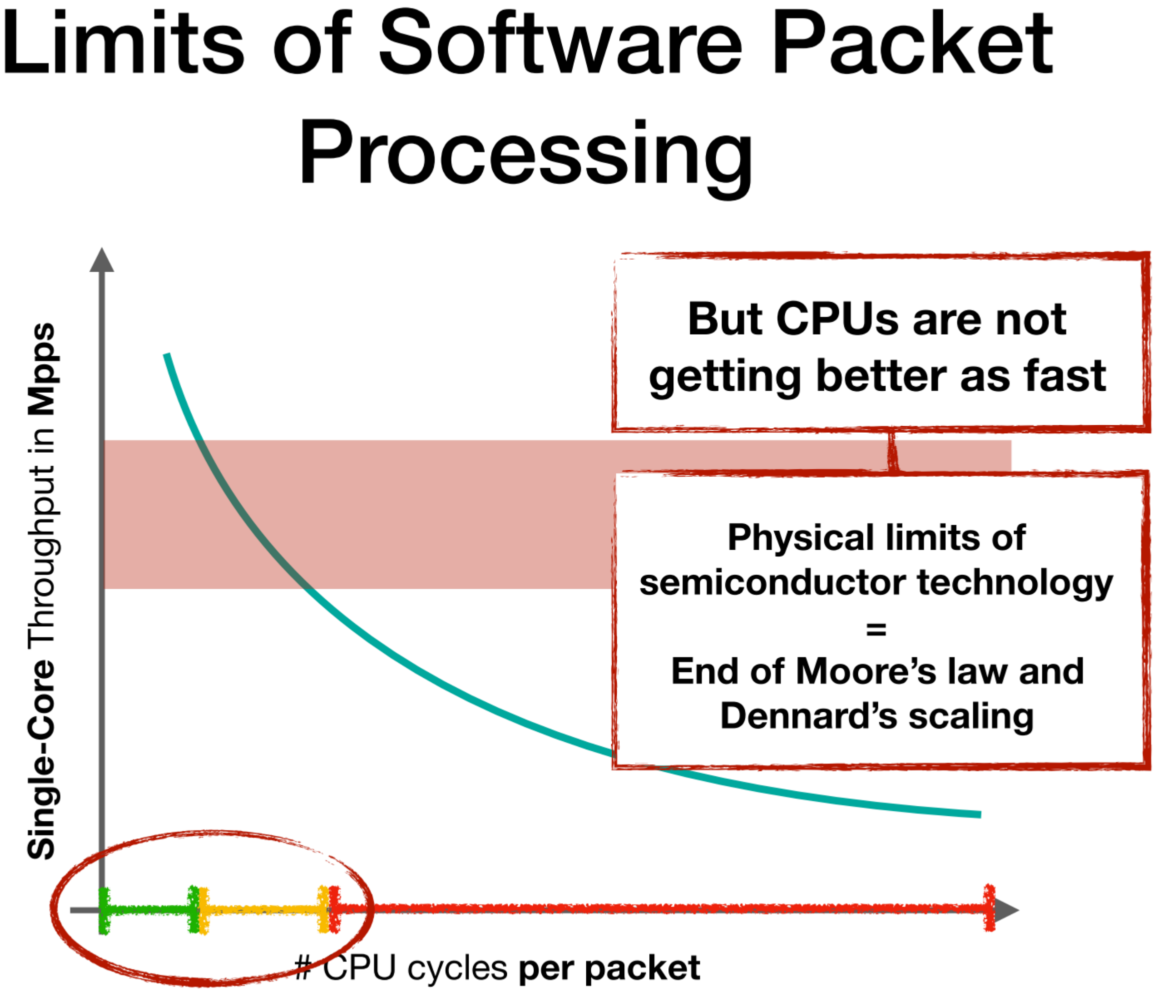
Limits of Software Packet Processing



Line rate is increasing to 100-400 Gbps

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (8)



Limits of Software Packet Processing

But CPUs are not getting better as fast

Physical limits of semiconductor technology
=
End of Moore's law and Dennard's scaling

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (9)

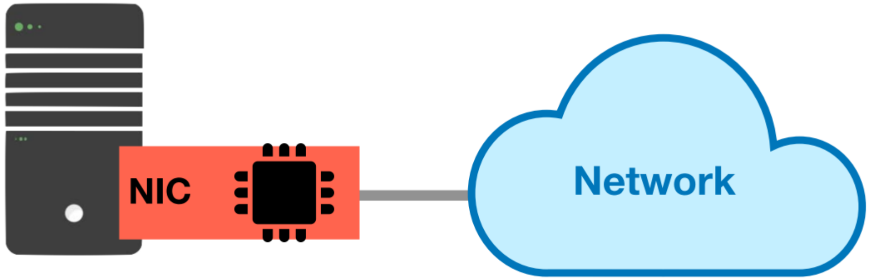
So, what *is* a smart NIC?

You can think of it as

a dumb NIC

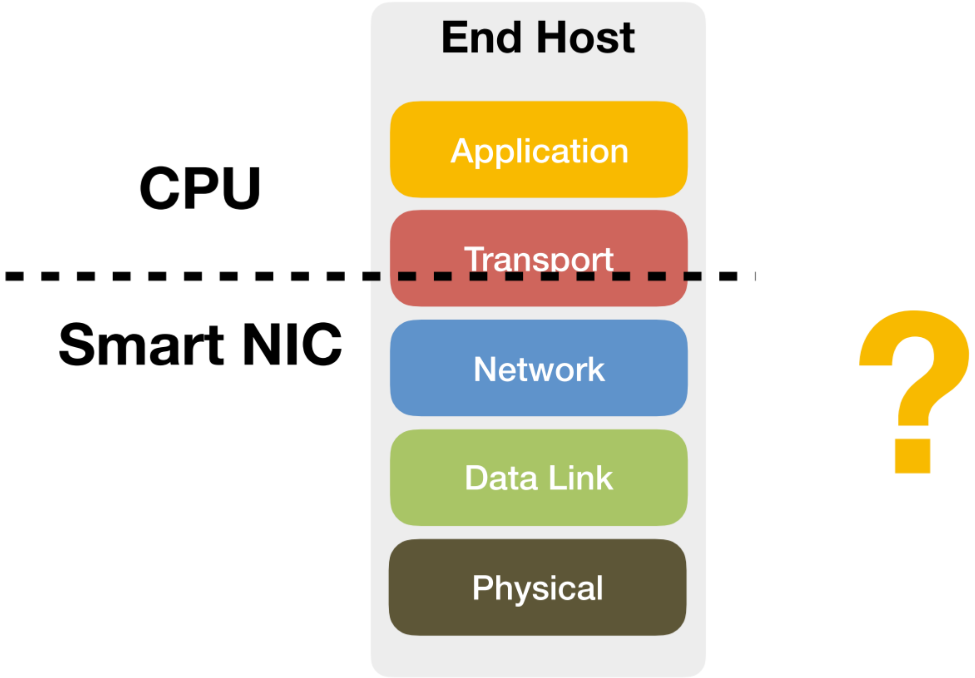
+

a programmable domain-specific hardware



smartNIC as a possible solution (10)

So, what *is* a smart NIC?



End Host

CPU

Smart NIC

Application

Transport

Network

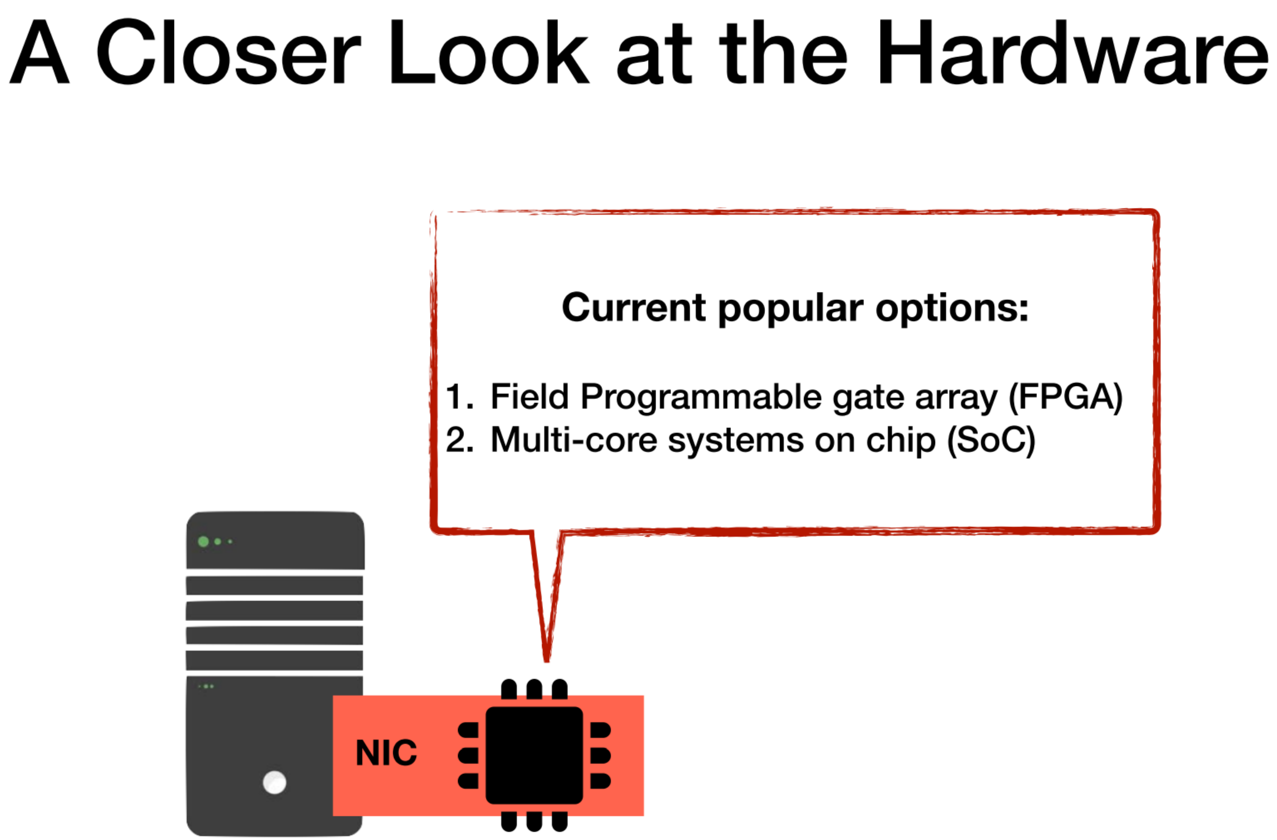
Data Link

Physical

?

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

smartNIC as a possible solution (11)



A Closer Look at the Hardware

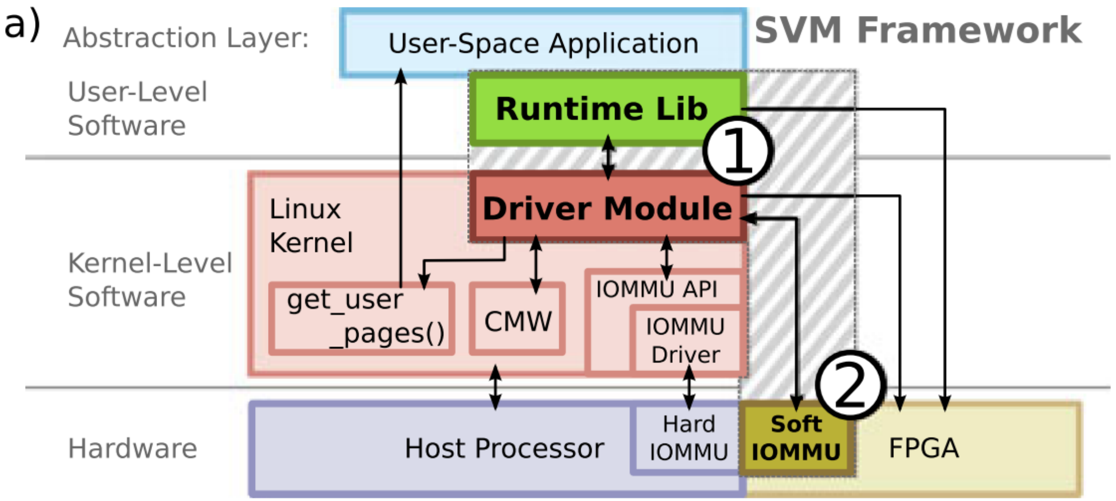
Current popular options:

1. Field Programmable gate array (FPGA)
2. Multi-core systems on chip (SoC)

Source: http://www.cs.cornell.edu/~mt822/docs/smartNICs_lecture.pdf

Complementary solutions: Shared Virtual Memory for FPGA Accelerators

- A key enabler for the ever-increasing adoption of FPGA accelerators is the availability of frameworks allowing for the seamless coupling to general-purpose host processors.
- Embedded FPGA+CPU systems still heavily rely on copy-based host-to-accelerator communication, which complicates application development.
- A hardware/software framework has been proposed for enabling transparent, shared virtual memory for FPGA accelerators in embedded SoCs.



Source: P. Vogel, A. Marongiu and L. Benini, "Exploring Shared Virtual Memory for FPGA Accelerators with a Configurable IOMMU," in IEEE Transactions on Computers, vol. 68, no. 4, pp. 510-525, 1 April 2019, doi: 10.1109/TC.2018.2879080.

Complementary solutions: Direct Path to CPU

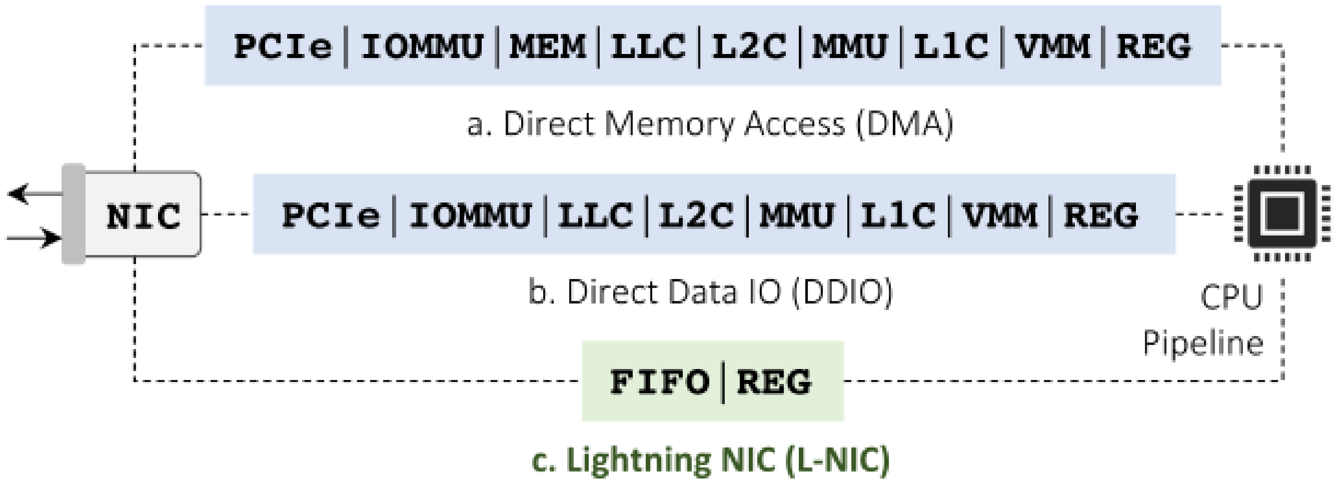
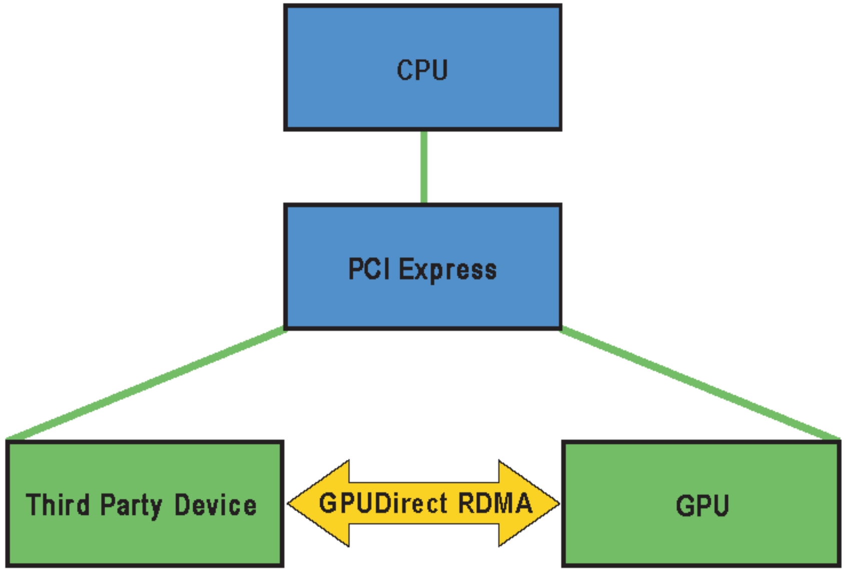

PCIe | IOMMU | MEM | LLC | L2C | MMU | L1C | VMM | REG
a. Direct Memory Access (DMA)
NIC
PCIe | IOMMU | LLC | L2C | MMU | L1C | VMM | REG
b. Direct Data IO (DDIO)
CPU
Pipeline
FIFO | REG
c. Lightning NIC (L-NIC)

Figure 2: L-NIC places packet data directly into a CPU register, eliminating intermediate overheads such as interconnect delay (PCIe), address translations (IOMMU, MMU, and VMM), memory (MEM) and cache accesses (LLC, L2C, and L1C).

Source: Stephen Ibanez, Muhammad Shahbaz, and Nick McKeown. 2019. The Case for a Network Fast Path to the CPU. In Proceedings of the 18th ACM Workshop on Hot Topics in Networks (HotNets '19). Association for Computing Machinery, New York, NY, USA, 52–59. https://doi.org/10.1145/3365609.3365851

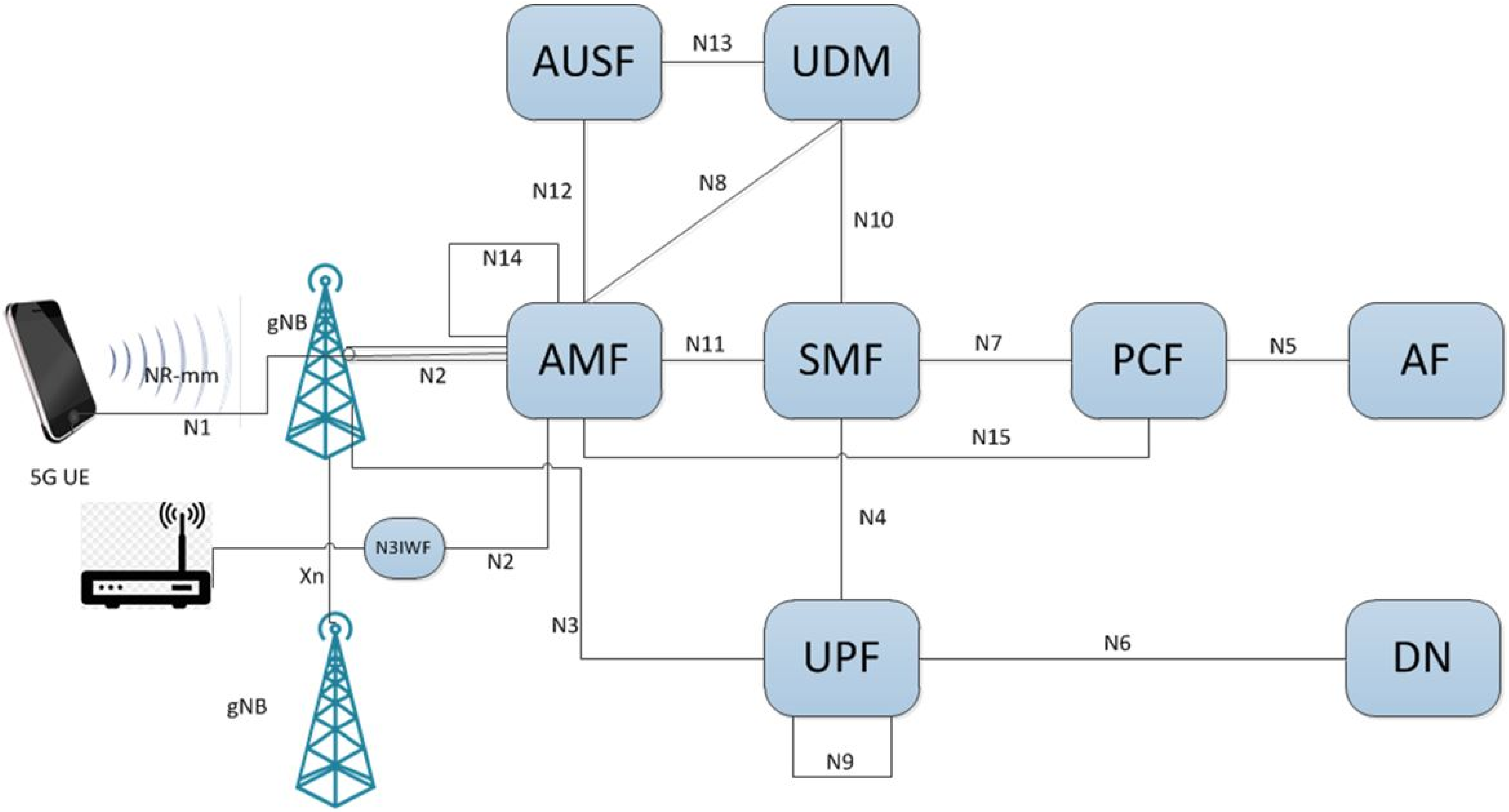
Complimentary solutions: GPUDirect RDMA (NVIDIA)

- GPUDirect Remote Direct Memory Access (RDMA) is a technology introduced in Kepler-class GPUs and CUDA 5.0 that enables a direct path for data exchange between the GPU and a third-party peer device using standard features of PCI Express.

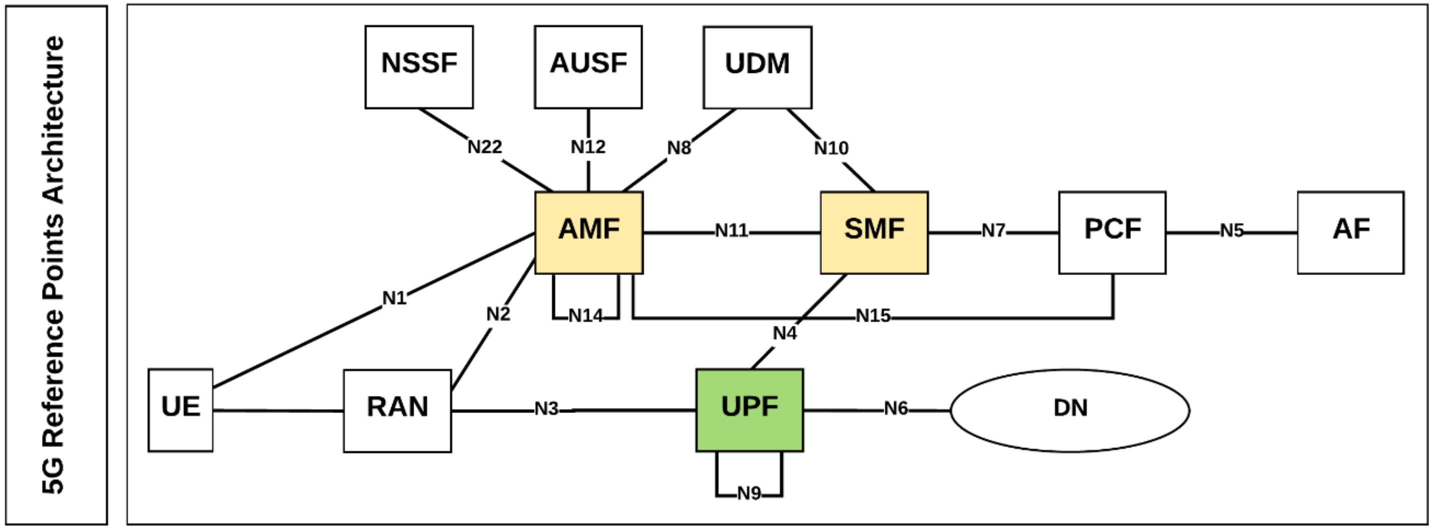


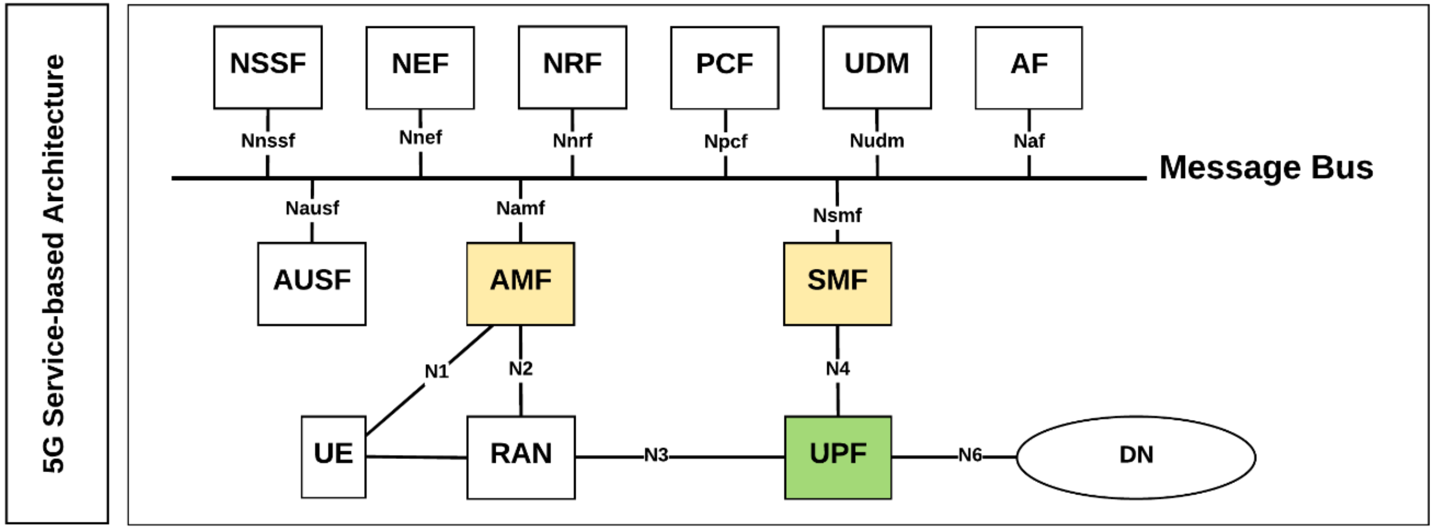
Source: https://docs.nvidia.com/cuda/gpudirect-rdma/index.html

5G Architecture



5G Architecture Functional Elements





Source: ETSI TS 123 501 V15.2.0 (2018-06)5G;System Architecture for the 5G System (3GPP TS 23.501 version 15.2.0 Release 15)

AMF: Access & Mobility Management Function

SMF: Session Management Function

UPF: User Plane Function

NEF: Network Exposure Function

NRF: NF Repository Function

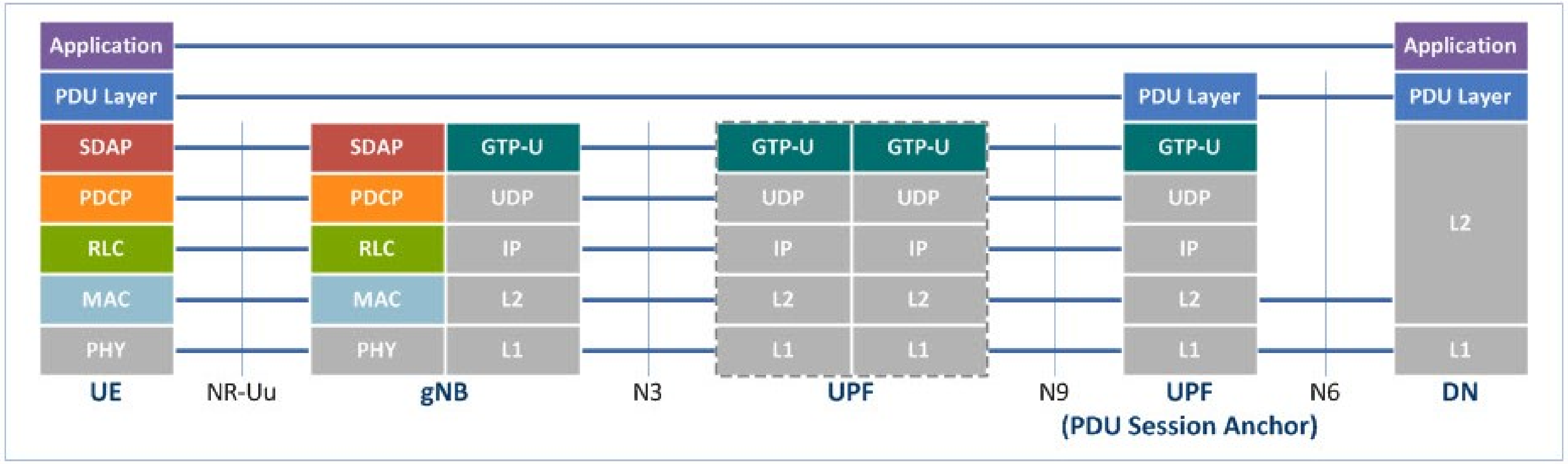
NSSF: Network Slice Selection Function

UDM: Unified Data Function

AUSF: Authentication Server Function

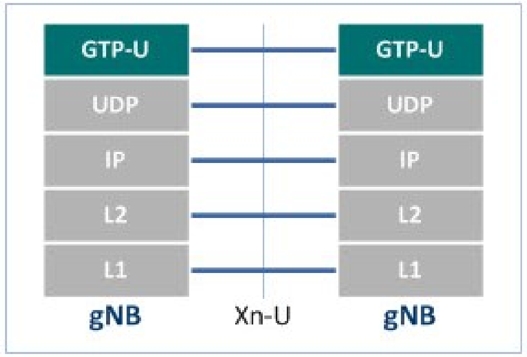
PCF: Policy Control Function

5G Protocol Stack: User Plane

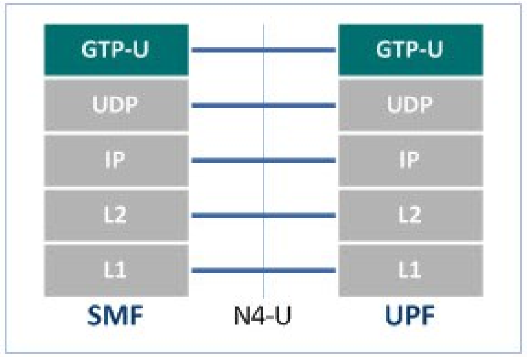


UE	NR-Uu	gNB		N3	UPF		N9	UPF (PDU Session Anchor)	N6	DN
Application										Application
PDU Layer								PDU Layer		PDU Layer
SDAP		SDAP	GTP-U		GTP-U	GTP-U		GTP-U		L2
PDCP		PDCP	UDP		UDP	UDP		UDP		
RLC		RLC	IP		IP	IP		IP		
MAC		MAC	L2		L2	L2		L2		
PHY		PHY	L1		L1	L1		L1		L1

PDU Layer: IP, Ethernet, etc.



gNB	Xn-U	gNB
GTP-U		GTP-U
UDP		UDP
IP		IP
L2		L2
L1		L1



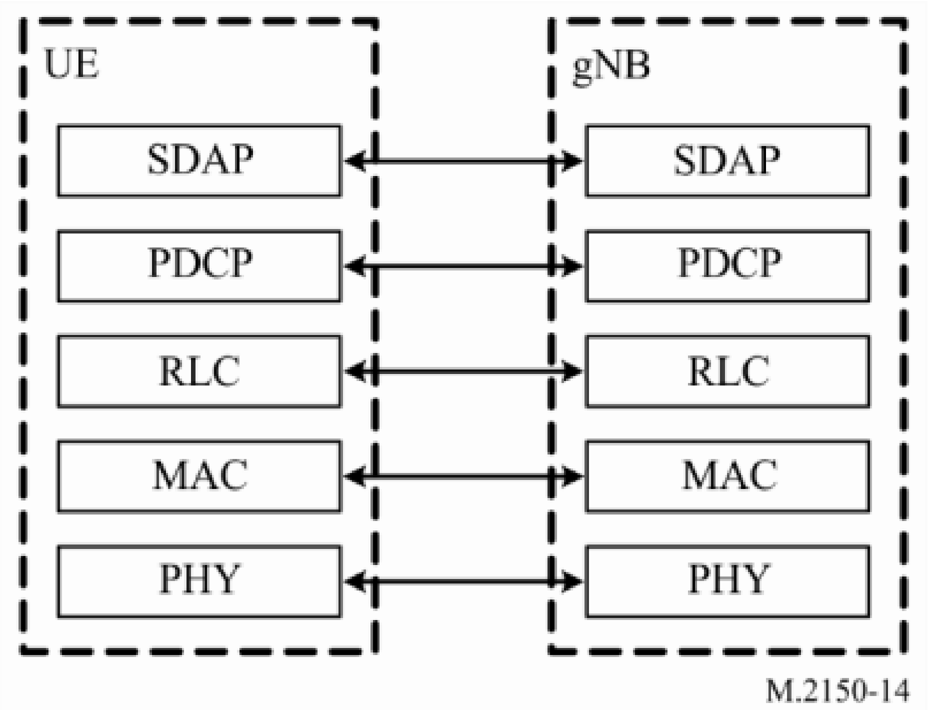
SMF	N4-U	UPF
GTP-U		GTP-U
UDP		UDP
IP		IP
L2		L2
L1		L1

DN	: Data Network	RLC	: Radio Link Control
gNB	: Next generation NodeB	SDAP	: Service Data Adaptation Protocol
GTP-U	: GPRS Tunneling Protocol User plane	SMF	: Session Management Function
MAC	: Medium Access Control	UE	: User Equipment
PDCP	: Packet Data Convergence Protocol	UPF	: User Plane Function
PDU	: Protocol Data Unit	Xn-U	: Xn User plane

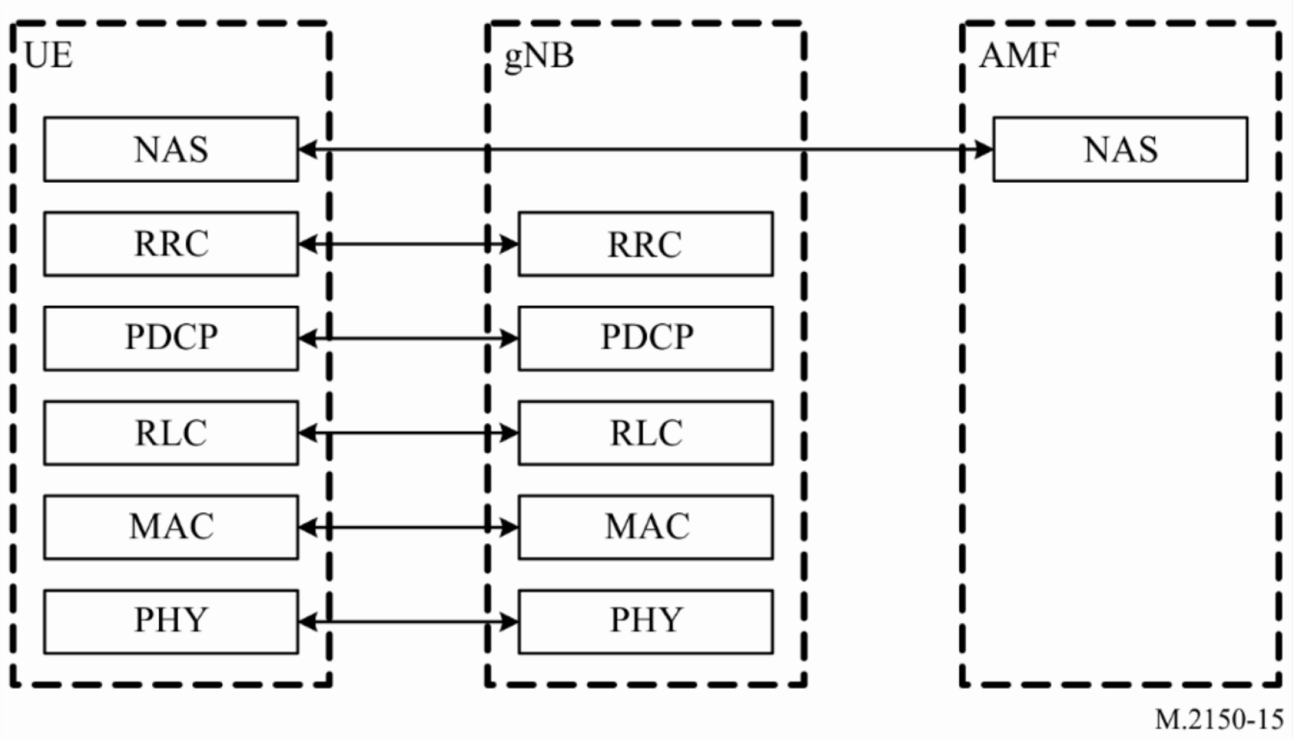
Source: https://www.netmanias.com/en/post/oneshot/14103/5g/5g-protocol-stack-user-plane-control-plane

5G Radio Protocol Architecture

User Plane protocol stack



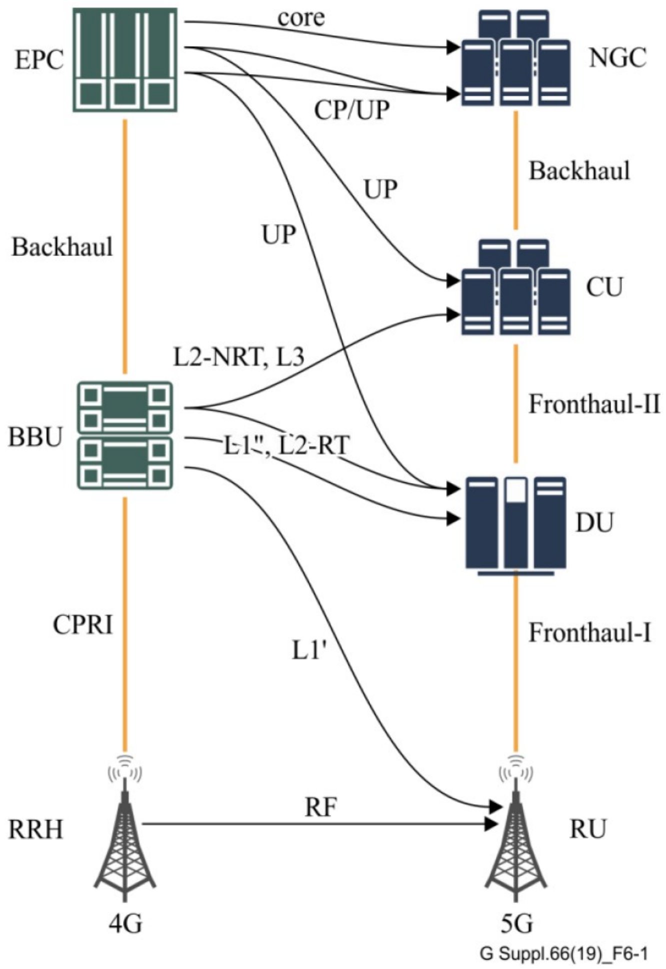
Control Plane Protocol Stack

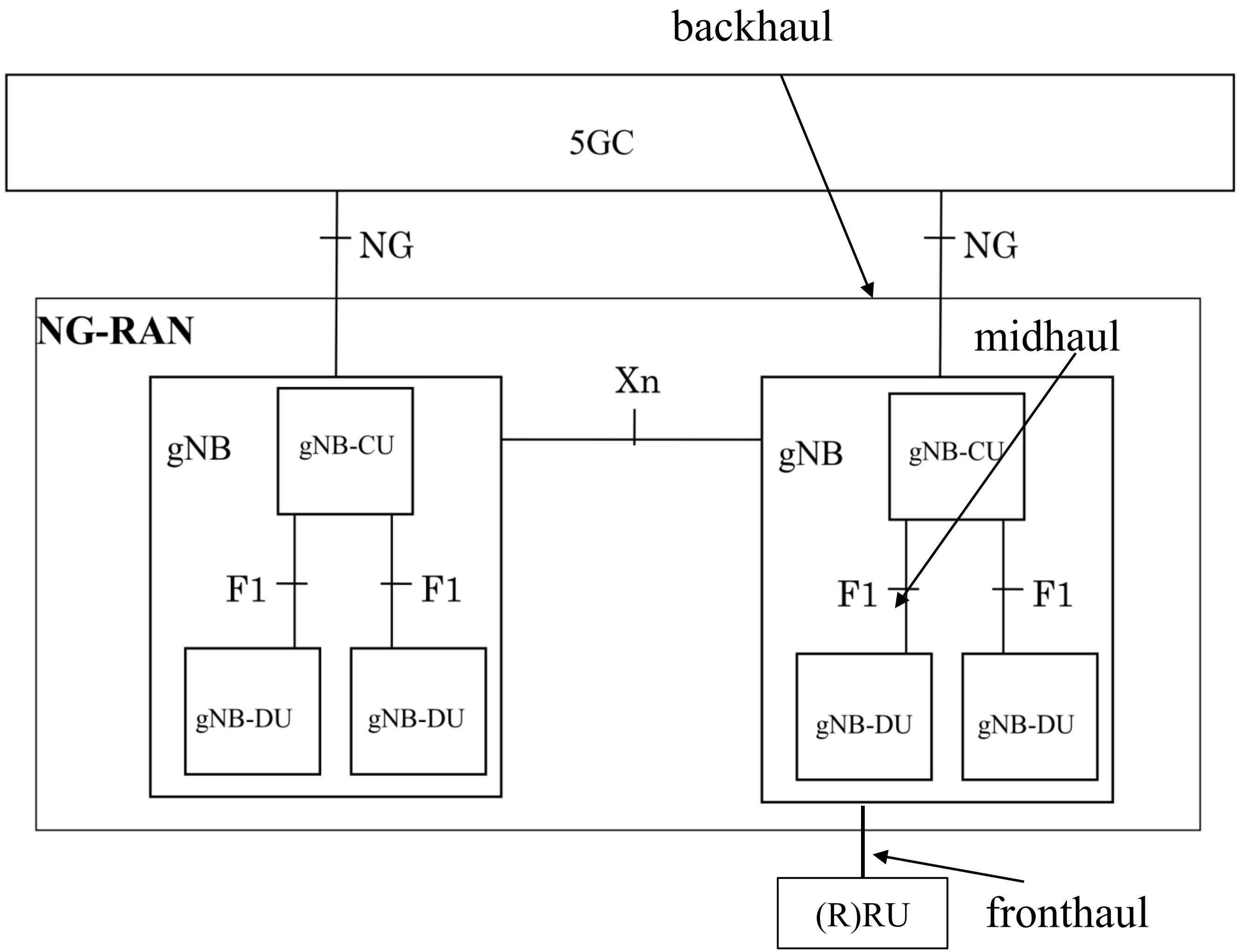


TELECOMMUNICATIONS, COMPUTER ENGINEERING, AND PHOTONICS INSTITUTE

Sant'Anna
School of Advanced Studies – Pisa

NG-RAN and Disaggreagated gNB

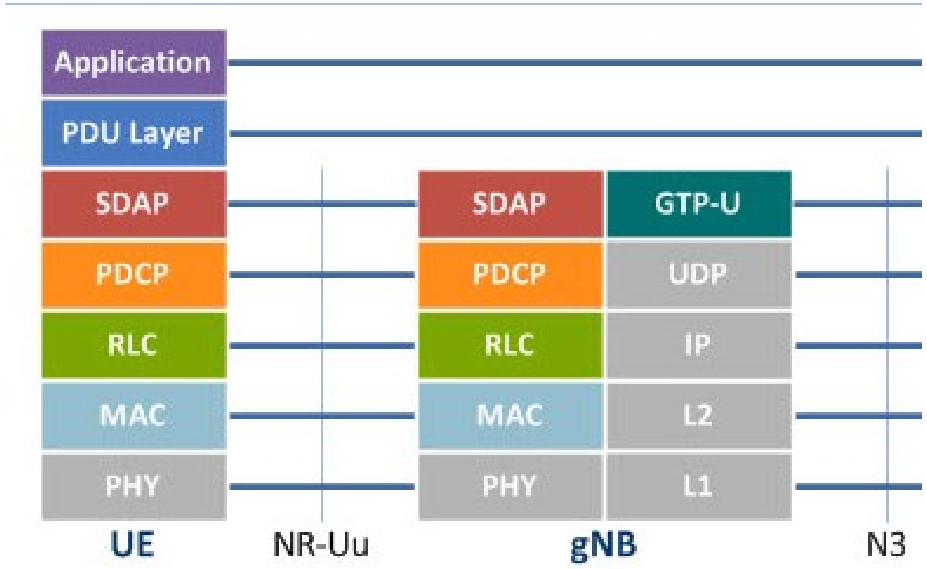


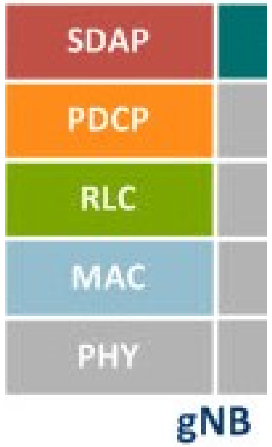


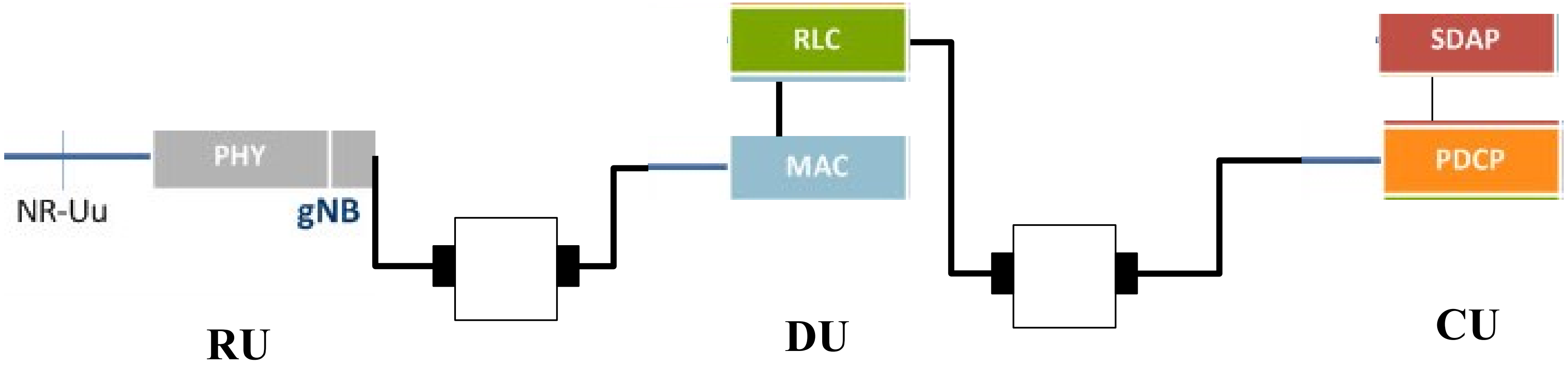
Source: ITU-T, Technical Report, "GSTR-TN5G Transport network support of IMT-2020/5G, 19 October 2018

ITU-T Series G Supplement 66 (09/2020) (G.Sup66), 5G wireless fronthaul requirements in a passive optical network context

5G Protocol Stack: User Plane

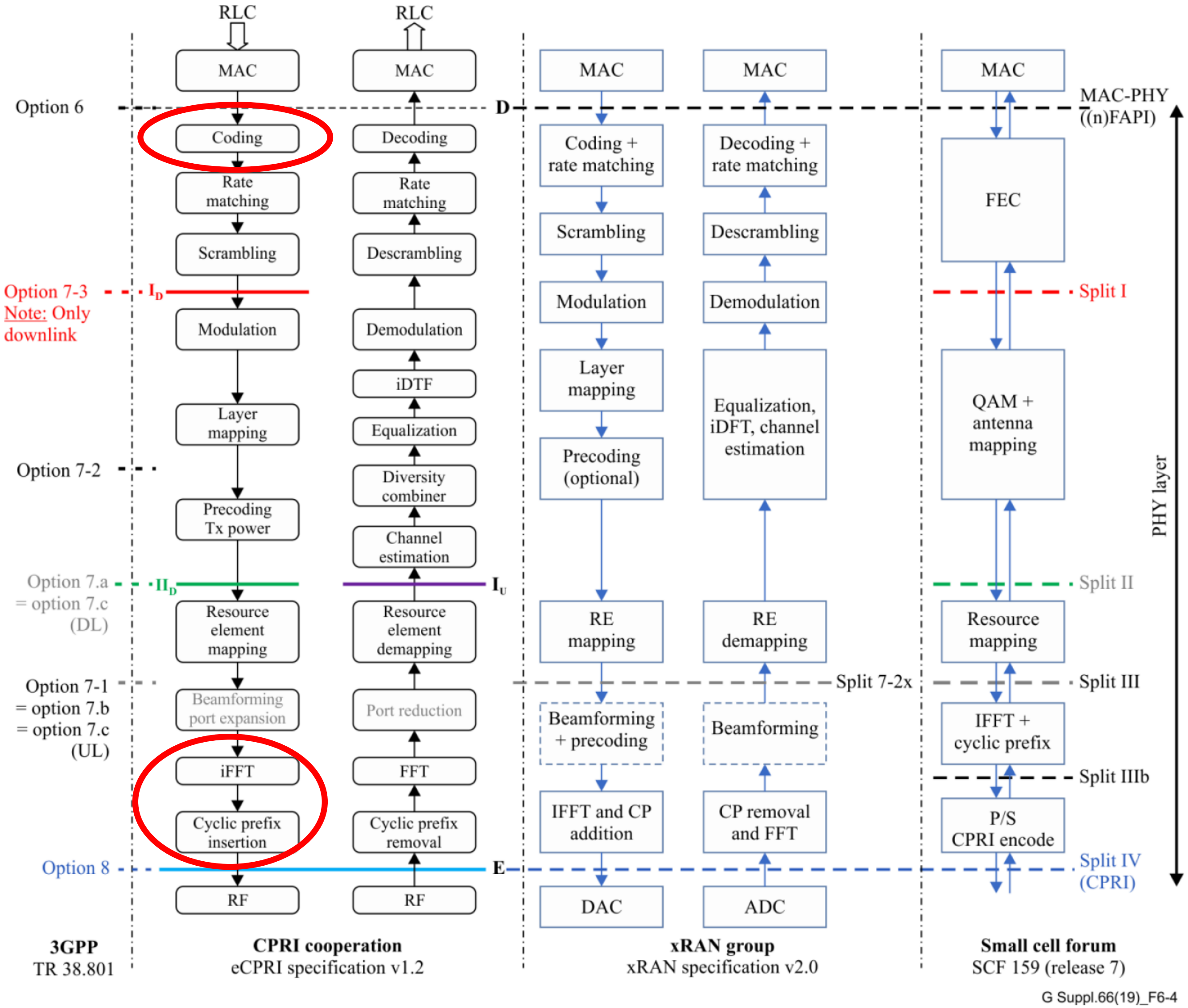






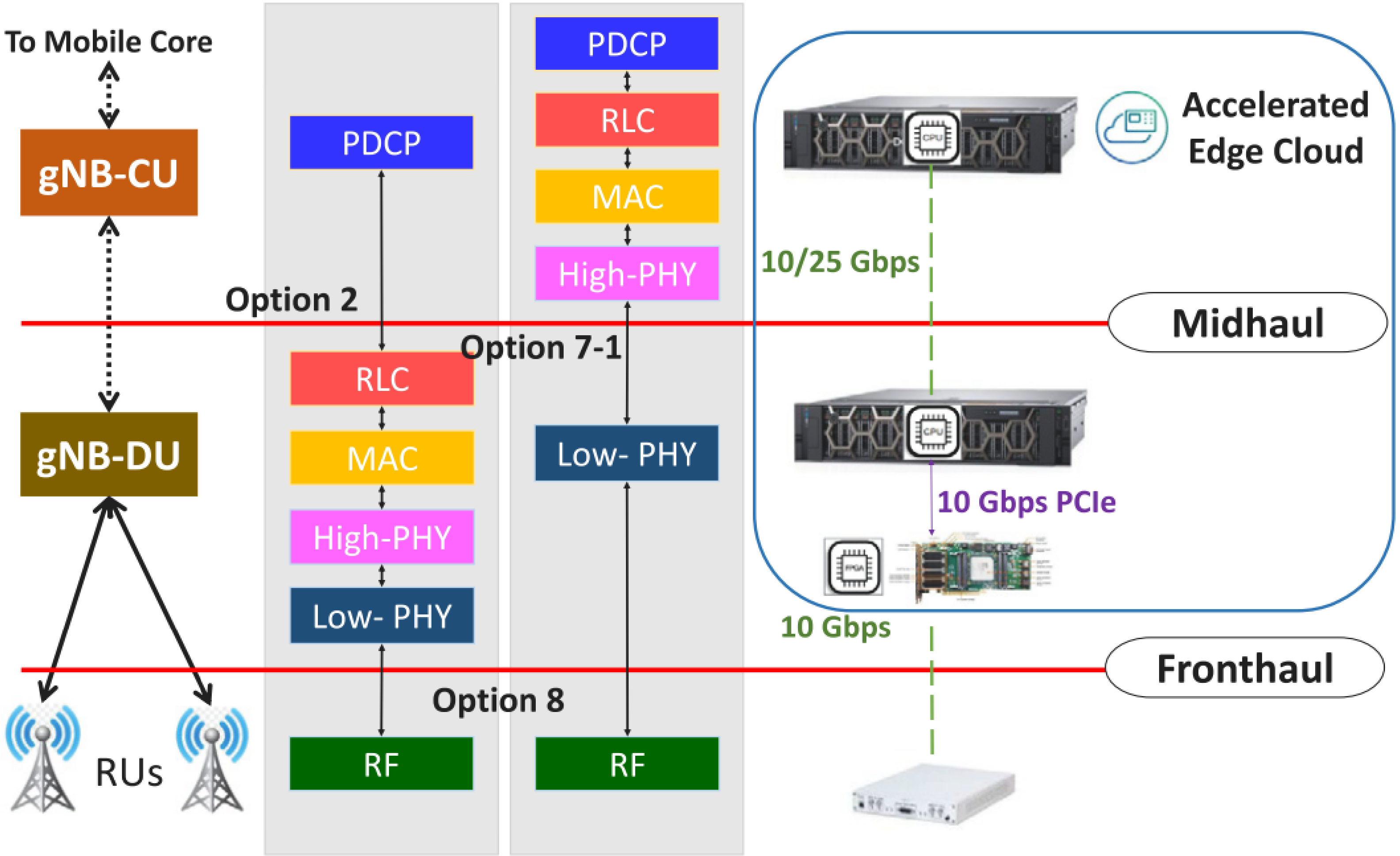
Source: https://www.netmanias.com/en/post/oneshot/14103/5g/5g-protocol-stack-user-plane-control-plane

Detailed Comparison of Option 7 Subsplit



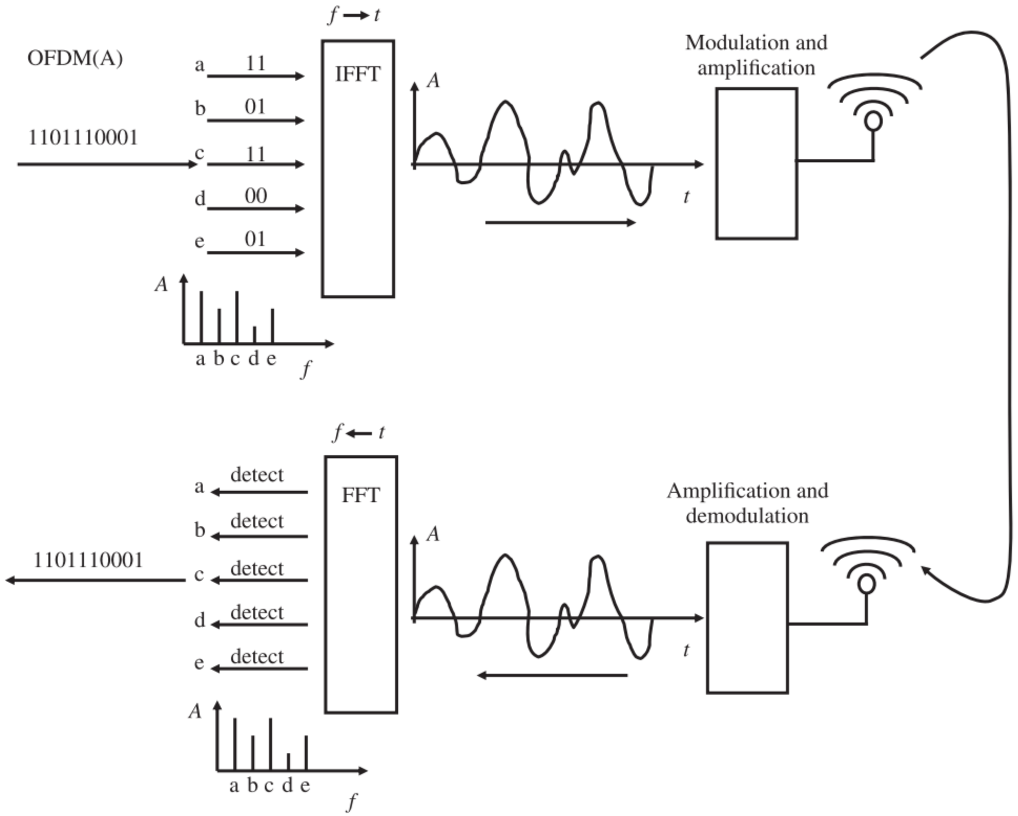
Source: ITU-T Series G Supplement 66 (09/2020) (G.Sup66), 5G wireless fronthaul requirements in a passive optical network context

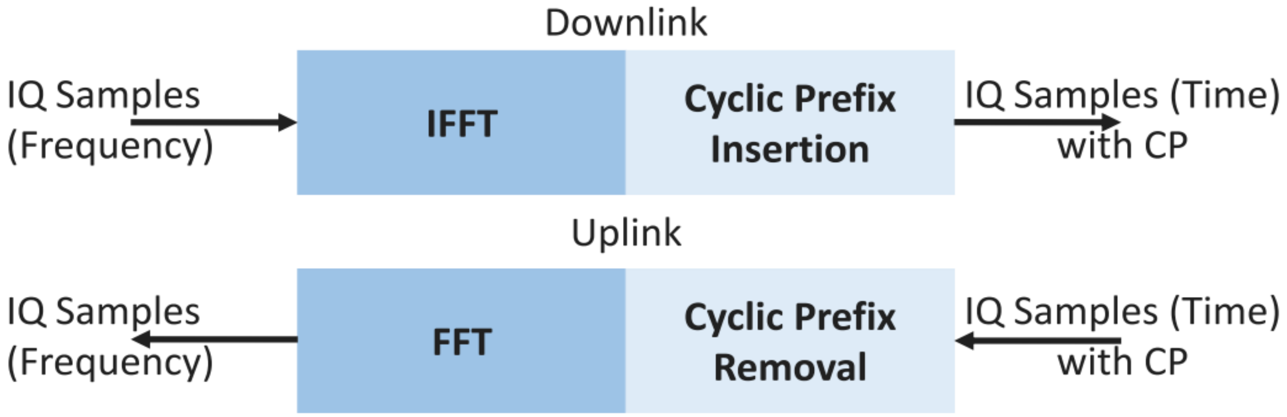
Programmable Hardware Acceleration of 5G Functions



Source: Justine Cris Borromeo, Koteswararao Kondepu, Nicola Andriolli, Luca Valcarenghi, “FPGA-accelerated SmartNIC for supporting 5G virtualized Radio Access Network,” Computer Networks, Volume 210, 2022, 108931, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2022.108931.

FFT/IFFT and OFDM





Downlink

IQ Samples (Frequency) → IFFT → Cyclic Prefix Insertion → IQ Samples (Time) with CP

Uplink

IQ Samples (Frequency) ← FFT ← Cyclic Prefix Removal ← IQ Samples (Time) with CP

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}; k = 0, 1, \ldots, N-1$$

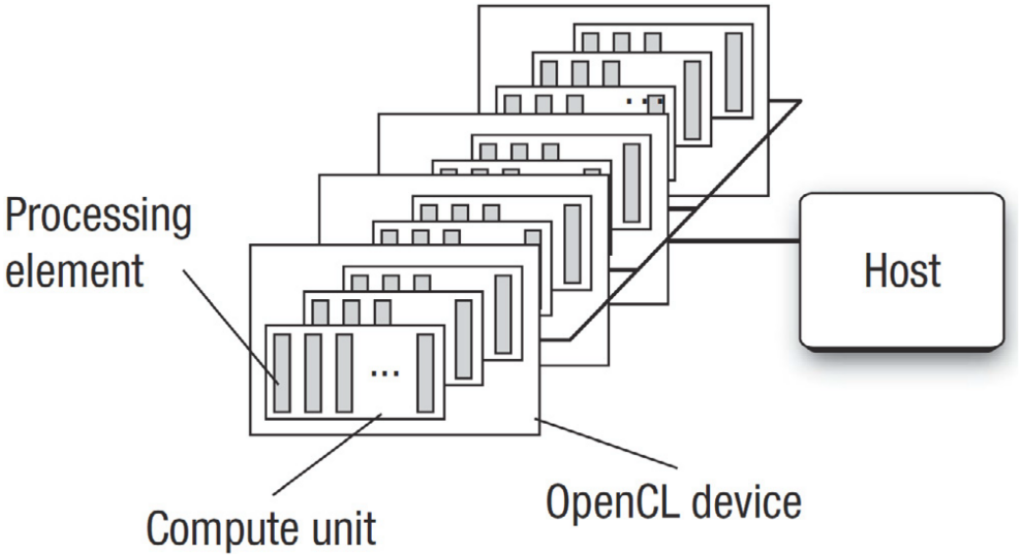
Samples in the time domain

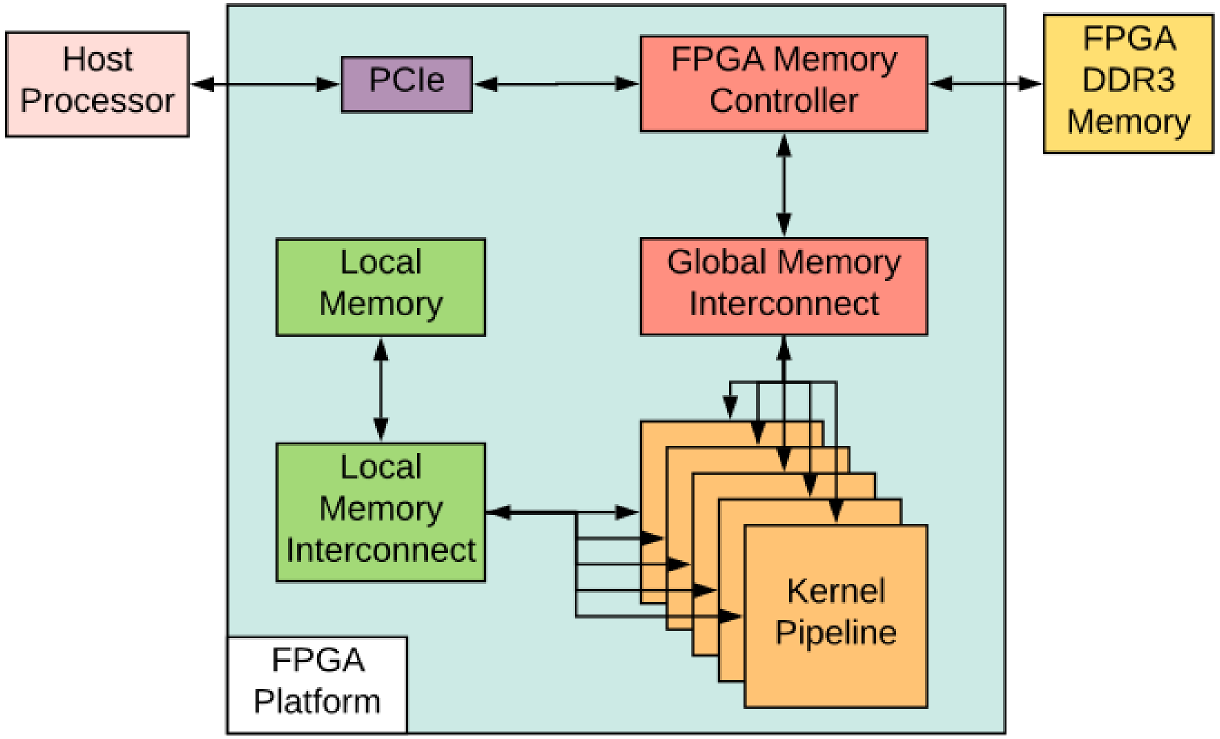
Samples in the frequency domain

$$W_N^{nk} = e^{-j2nk\pi/N}$$

Number of subcarriers = FFT points

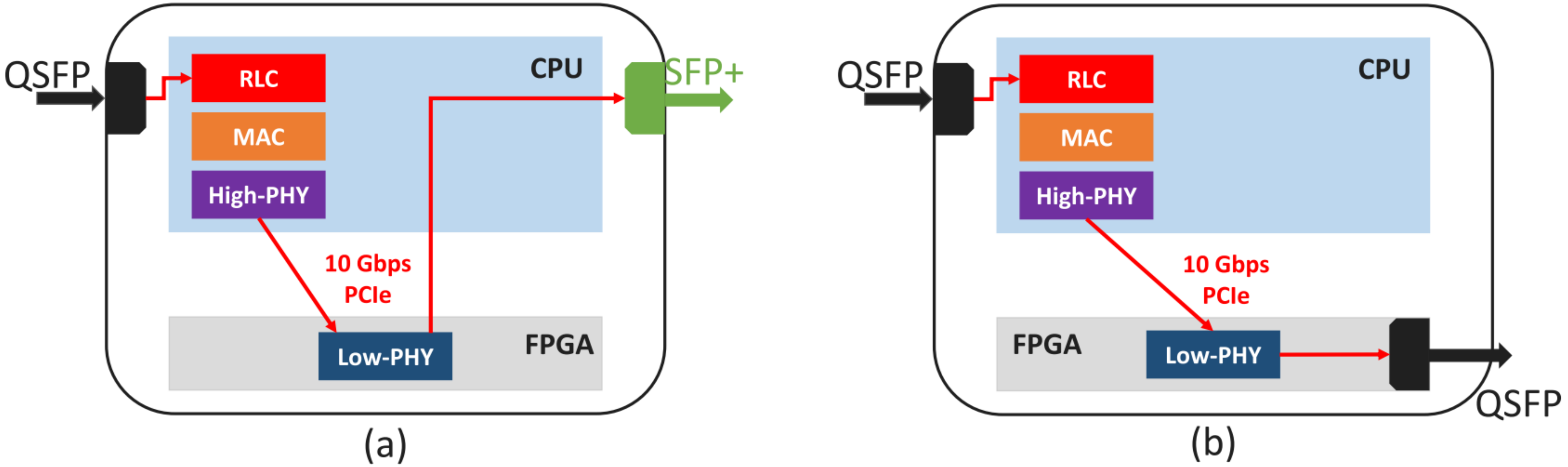
Open CL for Programmable Hardware and Software Integration



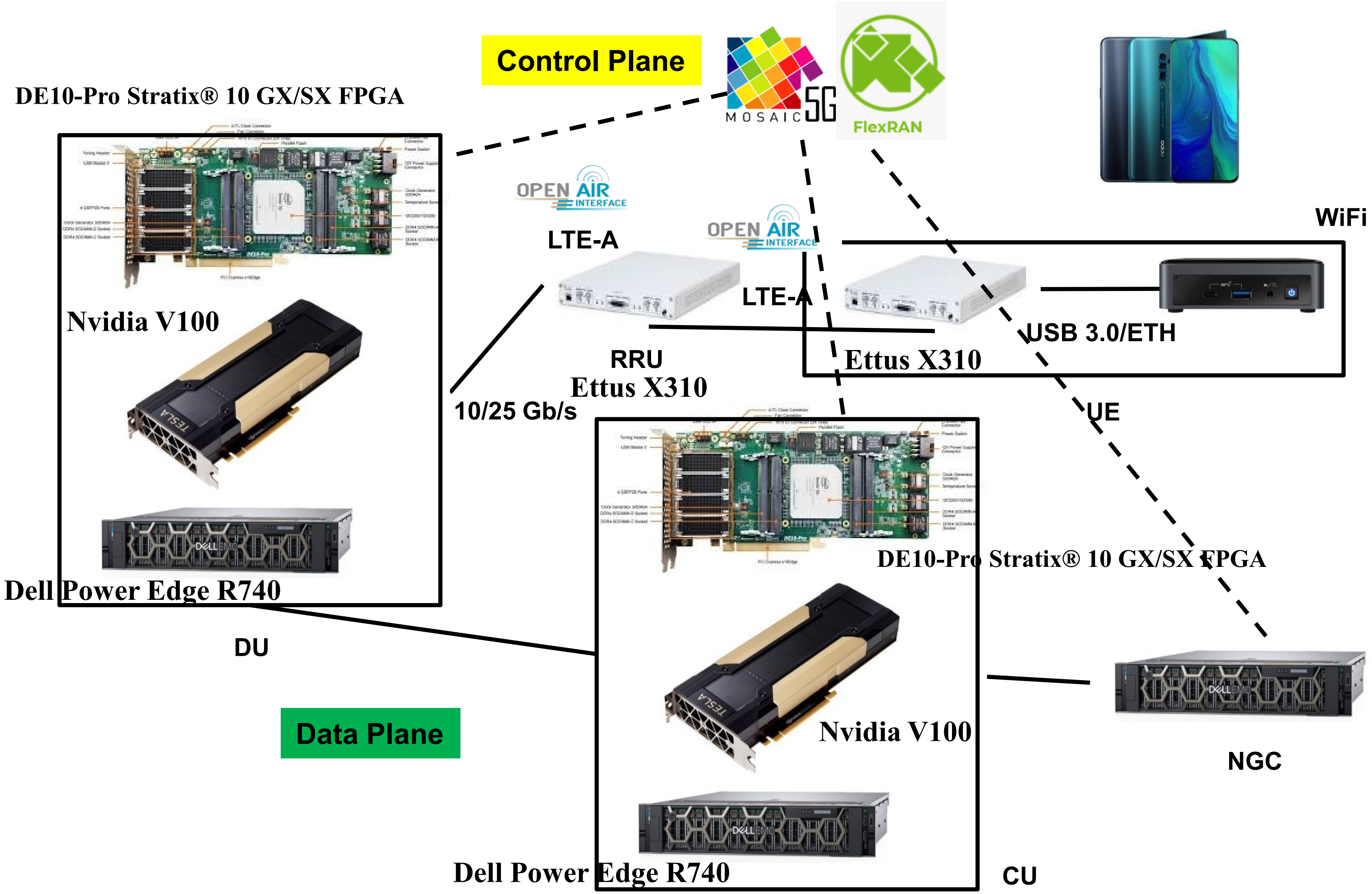


Source: F. Civerchia, M. Pelcat, L. Maggiani, K. Kondepu, P. Castoldi and L. Valcarenghi, "Is OpenCL Driven Reconfigurable Hardware Suitable for Virtualising 5G Infrastructure?," in IEEE Transactions on Network and Service Management, vol. 17, no. 2, pp. 849-863, June 2020, doi: 10.1109/TNSM.2020.2964392.

SmartNIC-based Implementation



SSSA 5G Testbed Setup



5G Testbed Setup Picture

Performance Evaluation for IFFT Implementation

- DU Low-PHY functions are offloaded onto a DE10-pro development board with Stratix 10 FPGA, two 8 GB DDR4 memory modules and PCIe v3.0 with 16 slots at 32 GB/s bandwidth
- The CPU-based implementation is executed in an Intel Core i7-7700K@4.2 GHz and based on Intel Advanced Vector Extension 2 (AVX2)
- The GPU-based implementation *clfft* and *cufft* is based on an NVIDIA Tesla T4 GPU featuring 320 NVIDIA Turing tensor cores,16 GB GDDR6 memory modules, and PCIe v3.0 with 16 slots

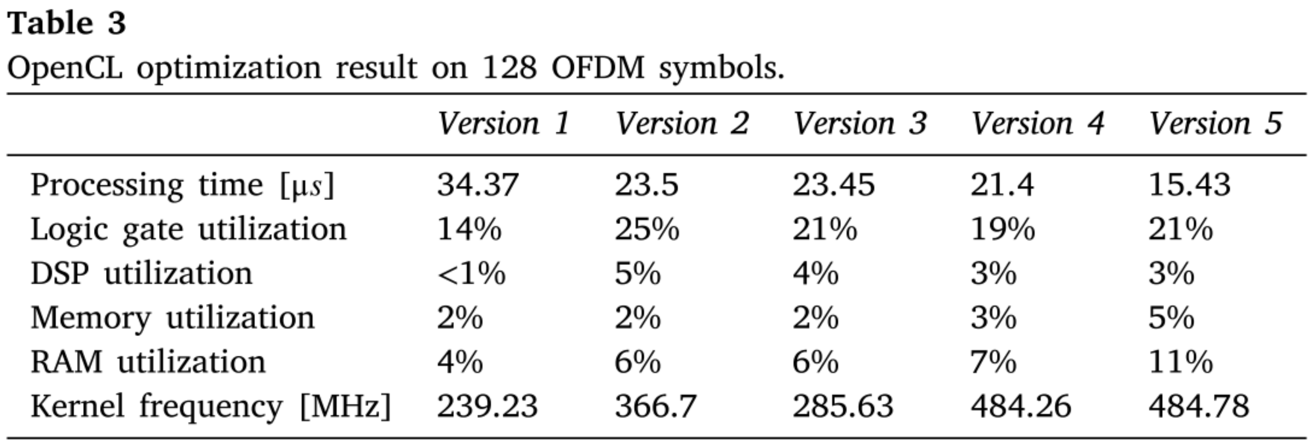


Table 3
OpenCL optimization result on 128 OFDM symbols.

	Version 1	*Version 2*	*Version 3*	*Version 4*	*Version 5*
Processing time [μs]	34.37	23.5	23.45	21.4	15.43
Logic gate utilization	14%	25%	21%	19%	21%
DSP utilization	<1%	5%	4%	3%	3%
Memory utilization	2%	2%	2%	3%	5%
RAM utilization	4%	6%	6%	7%	11%
Kernel frequency [MHz]	239.23	366.7	285.63	484.26	484.78

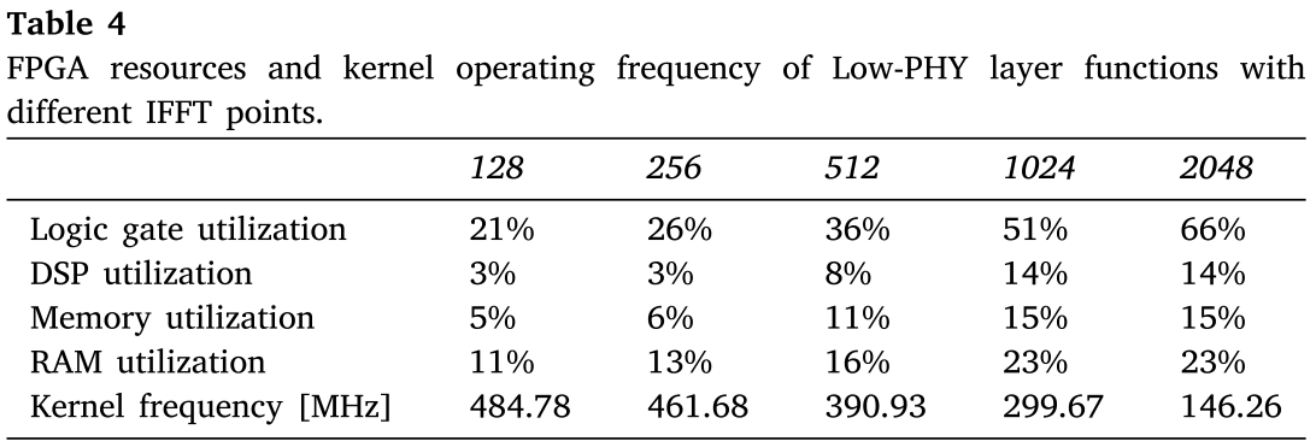
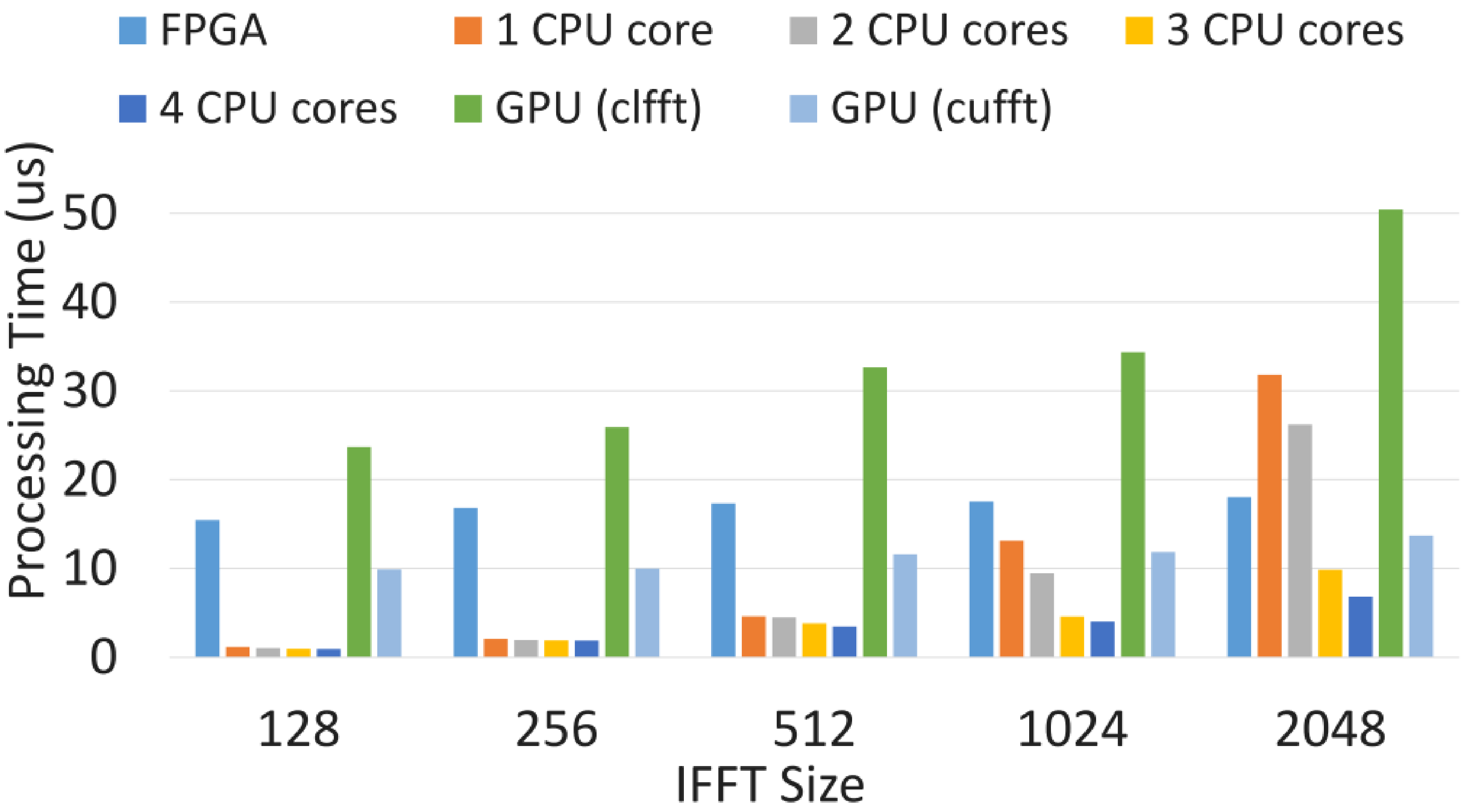


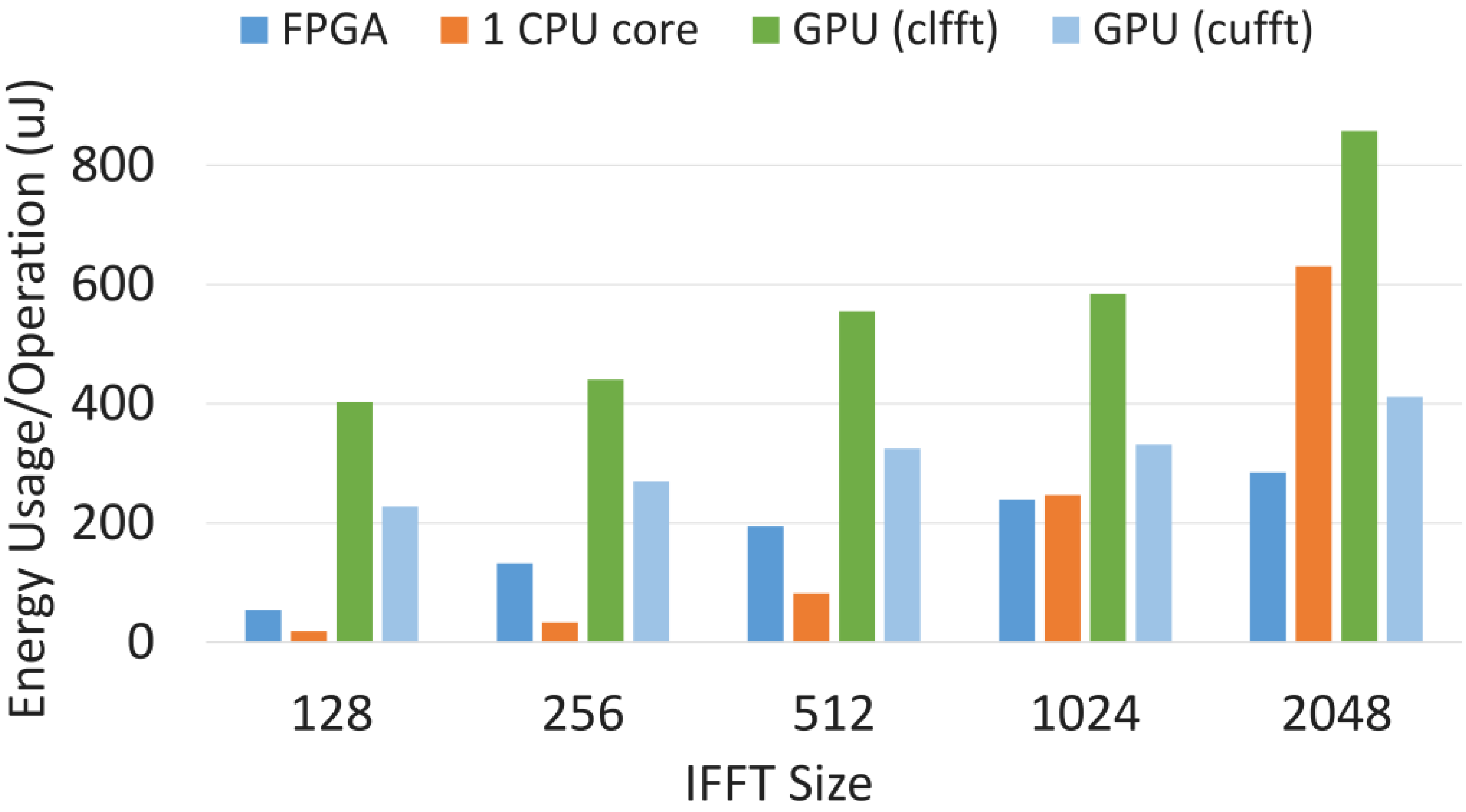
Table 4
FPGA resources and kernel operating frequency of Low-PHY layer functions with different IFFT points.

	128	*256*	*512*	*1024*	*2048*
Logic gate utilization	21%	26%	36%	51%	66%
DSP utilization	3%	3%	8%	14%	14%
Memory utilization	5%	6%	11%	15%	15%
RAM utilization	11%	13%	16%	23%	23%
Kernel frequency [MHz]	484.78	461.68	390.93	299.67	146.26

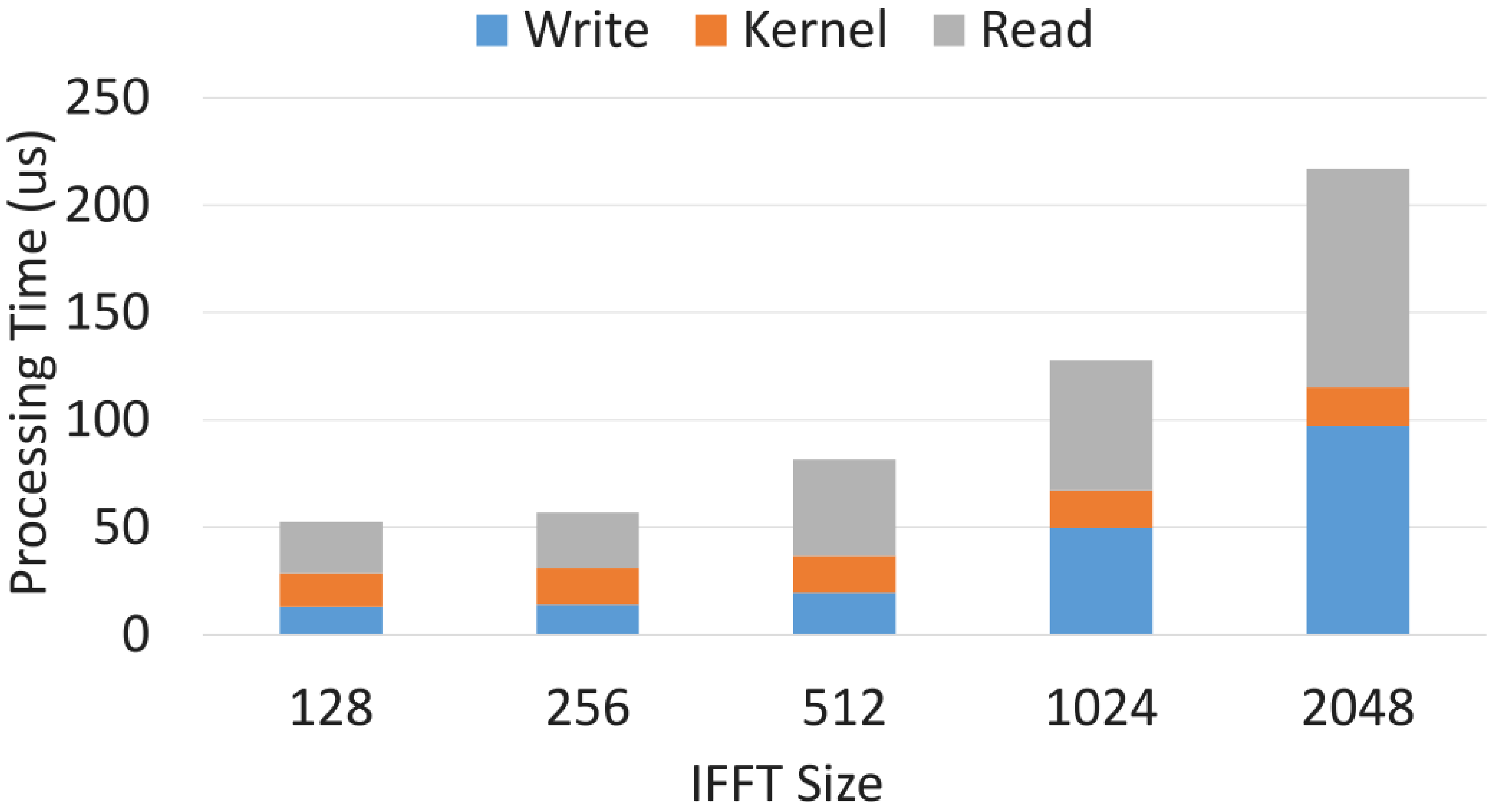
FPGA, CPU, and GPU Comparison: Processing Time



FPGA, CPU, and GPU Comparison: Energy Consumption

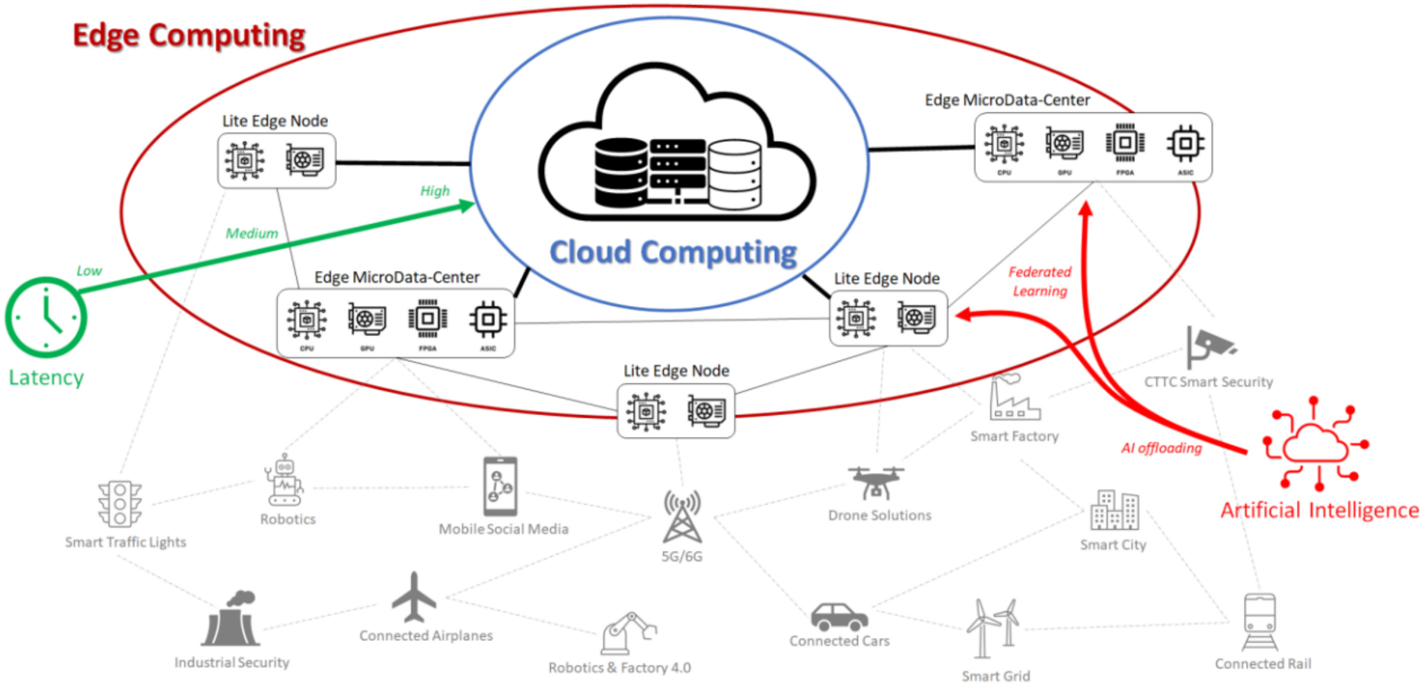


FPGA-based Implementation: Processing Time Contributions



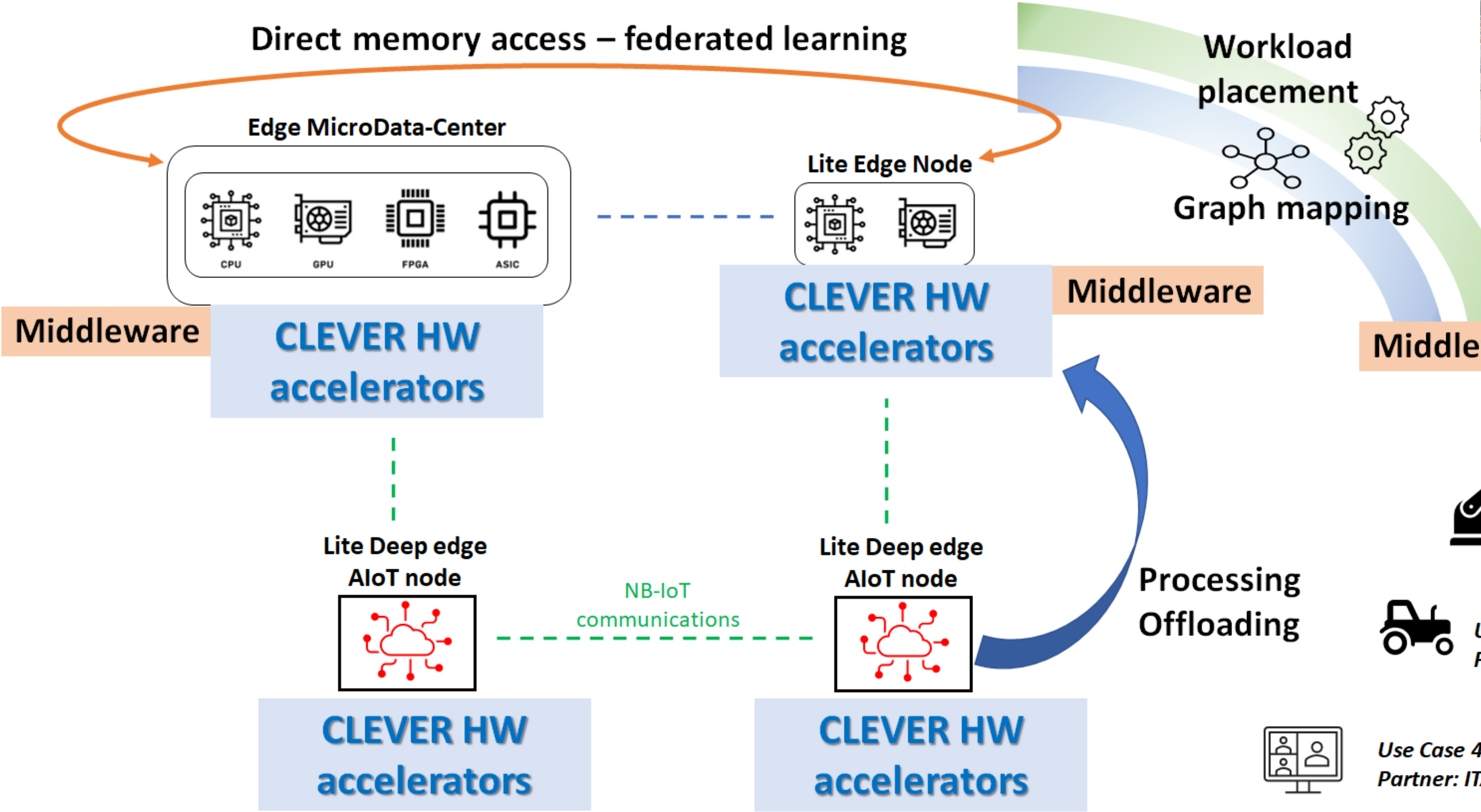
Computing Continuum, Digital Continuum, Transcontinuum

- It seamlessly combines resources and services at the center of the network (*e.g.*, in Cloud datacenters), at its Edge, and *in-transit*, along the data path.
- Typically, data is first generated and pre-processed (*e.g.*, filtering, basic inference) on Edge devices, while Fog nodes further process partially aggregated data.



Source: Daniel Rosendo, Alexandru Costan, Patrick Valduriez, Gabriel Antoniu, Distributed intelligence on the Edge-to-Cloud Continuum: A systematic literature review, Journal of Parallel and Distributed Computing, Volume 166, 2022, Pages 71-94, ISSN 0743-7315, https://doi.org/10.1016/j.jpdc.2022.04.004.

CLEVER Methodology



Use Case 1: Digital Twin for In-Factory Optimization (Lead Partner: BMW)

Use Case 2: Smart Agriculture for High Yield Eco-Farms (Lead Partner: AGRICOLUS)

Use Case 4: Augmented Reality for Shopping Sites (Lead Partner: ITALTEL)

Conclusions

- Overview of network function virtualisation
- Potential application of FPGA acceleration in 5G
  - LDPC
  - iFFT/FFT
- Short processing time and energy consumptions are achievable
- Bottleneck: data transfer
- Need for SmartNIC and direct data access solutions

Open positions

- https://www.santannapisa.it/en/institute/tecip
- https://www.santannapisa.it/en/training/international-phd-course-emerging-digital-technologies

CLEVER

Collaborative edge-cLoud continuum and Embedded AI for a Visionary industry of the future

https://www.cleverproject.eu/

EUROPEAN PARTNERSHIP

KEY DIGITAL TECHNOLOGIES JOINT UNDERTAKING

luca.valcarenghi@santannapisa.it

https://www.youtube.com/@Clever-Project

CleveSocial

CLEVERproject_EU

https://www.facebook.com/profile.php?id=61552443714715

THANKS !!!