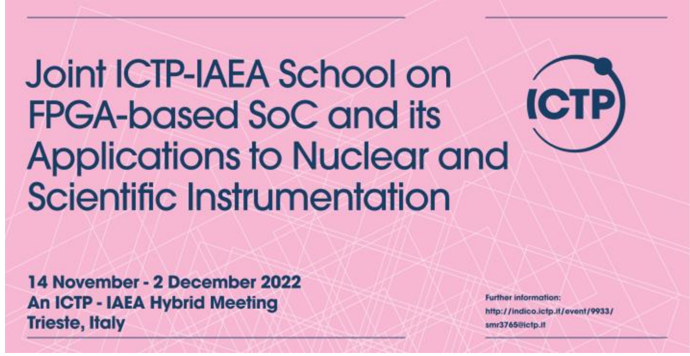
TELECOMMUNICATIONS,
COMPUTER
ENGINEERING,
AND PHOTONICS
INSTITUTE

Sant'Anna
School of Advanced Studies – Pisa



Joint ICTP-IAEA School on
FPGA-based SoC and its
Applications to Nuclear and
Scientific Instrumentation

ICTP

14 November - 2 December 2022
An ICTP - IAEA Hybrid Meeting
Trieste, Italy

Further information:
http://indico.ictp.it/event/9933/
smr3765@ictp.it

Programmable hardware acceleration in Communication Networks

L. Valcarenghi

Scuola Superiore Sant'Anna, Pisa, Italy

Joint ICTP-IAEA School on FPGA-based SoC and its
Applications to Nuclear and Scientific Instrumentation
Nov. 17, 2022, Trieste, Italy

Outline

- Edge cloud continuum
- SDN
- NFV
- Chances for in network elaboration
- Network programmability

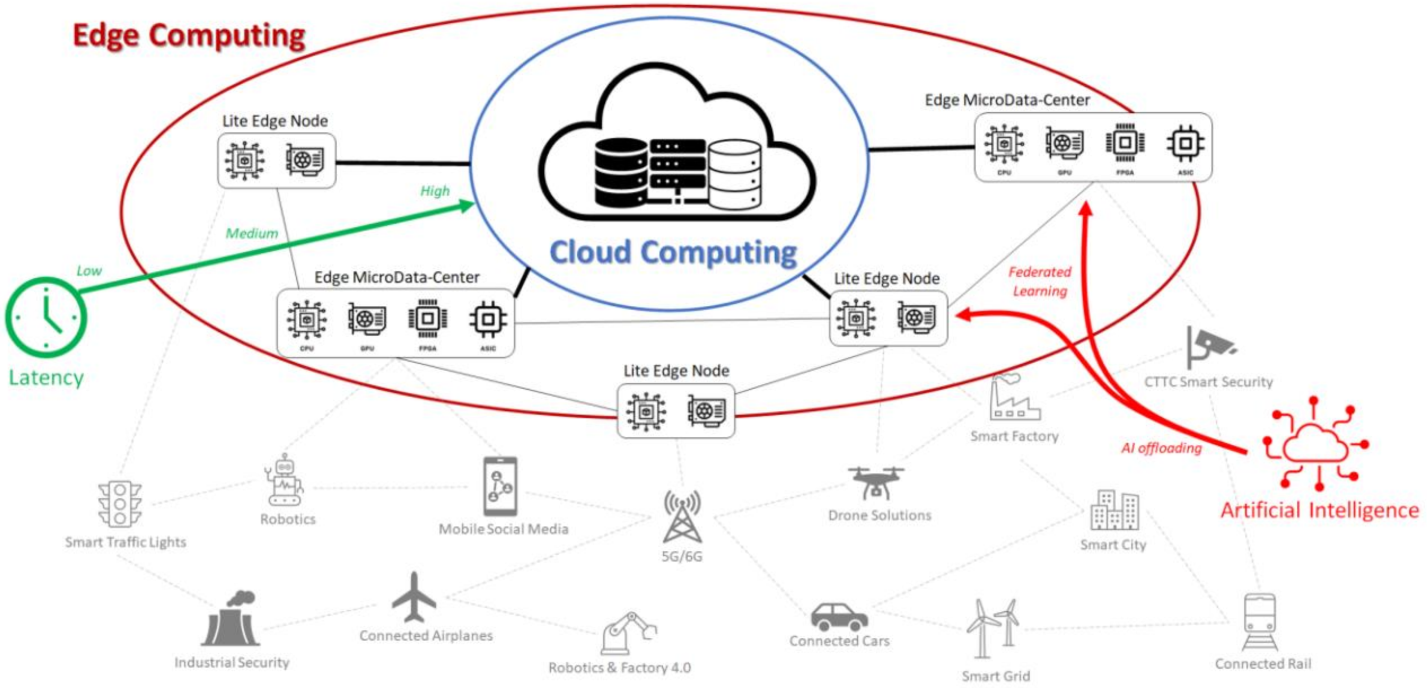
Edge Cloud Continuum and In-network computing

- Traditionally, little integration of computing tasks into networks:
  - The intelligence remains in end nodes
  - Network fabric is optimized for speed and security
- Hardware has evolved
  - virtualisation and data processing
  - new ways to design in-network services based on local processing of the data
- With the emergence of white boxes and smart Network Interface Cards (smart NICs), network nodes have increased their computation capabilities including also hardware acceleration.
- The availability of new hardware architectures (e.g. Tofino) and programming frameworks (e.g. P4) makes it possible to perform some in-network computing at line speed

Source: Marie-José Montpetit, Noel Crespi, "The Core-Edge Continuum in 6G Network" Chapter 10 in "Computing in the Network" Book Editor(s):Emmanuel Bertin, Noel Crespi, Thomas Magedanz First published: 05 November 2021 https://doi.org/10.1002/9781119765554.ch10

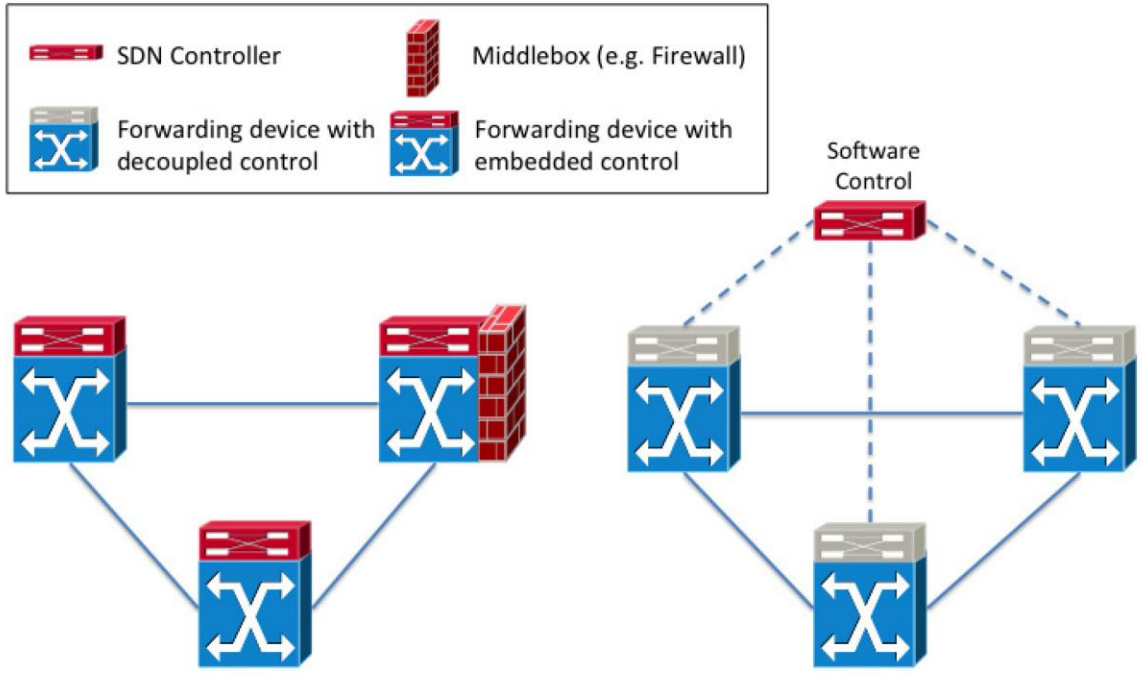
Computing Continuum, Digital Continuum, Transcontinuum

- 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.



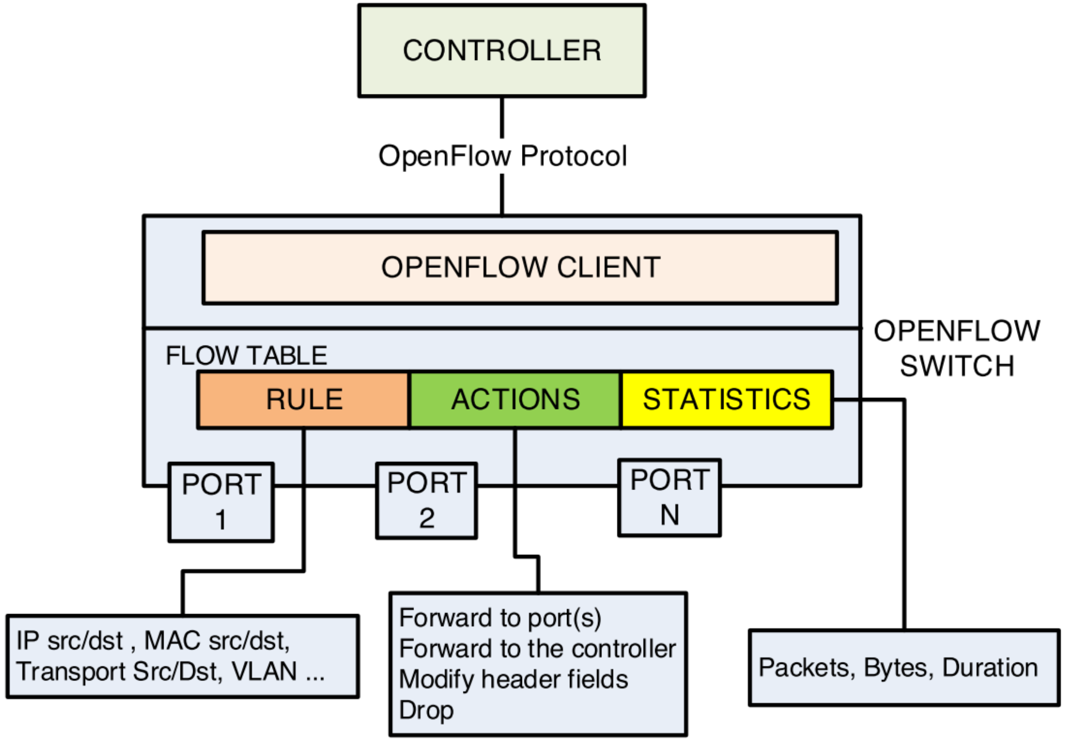
Source: Daniel Rosendo, Alexandru Costan, Patrick Valduriez, Gabriel Antoniu, Distributed intelligence on the Edge-to-Cloud Continuum: A systematic literature review, Journal of Parallel and Distributed Computing, Volume 166, 2022, Pages 71-94, ISSN 0743-7315, https://doi.org/10.1016/j.jpdc.2022.04.004.

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)

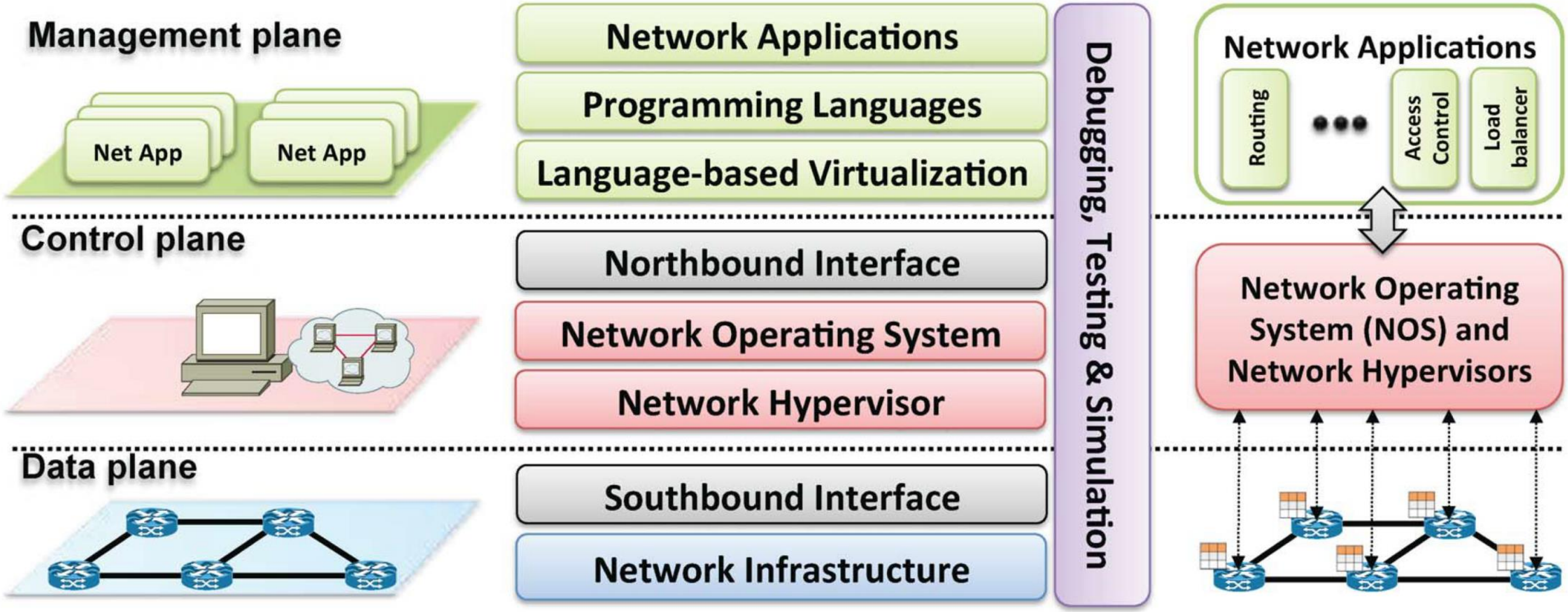
Software-Defined Network
(with decoupled control)



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.

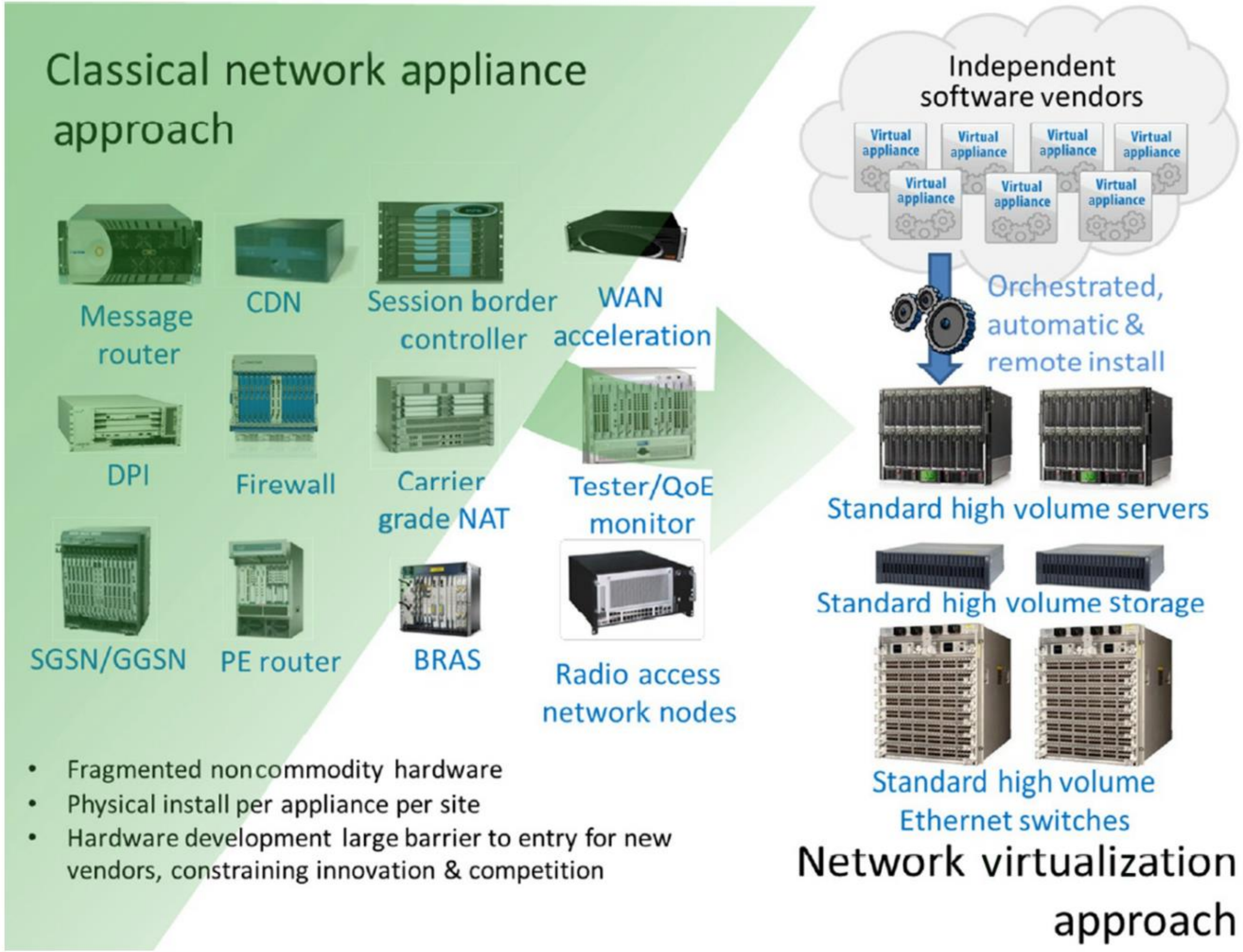
SDN in different representations: planes, layers, system design architecture



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.

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
Virtual appliance
Virtual appliance
Virtual appliance
Virtual appliance
Virtual appliance
Virtual appliance
Orchestrated, automatic & remote install
Standard high volume servers
Standard high volume storage
Standard high volume Ethernet switches
Network virtualization approach

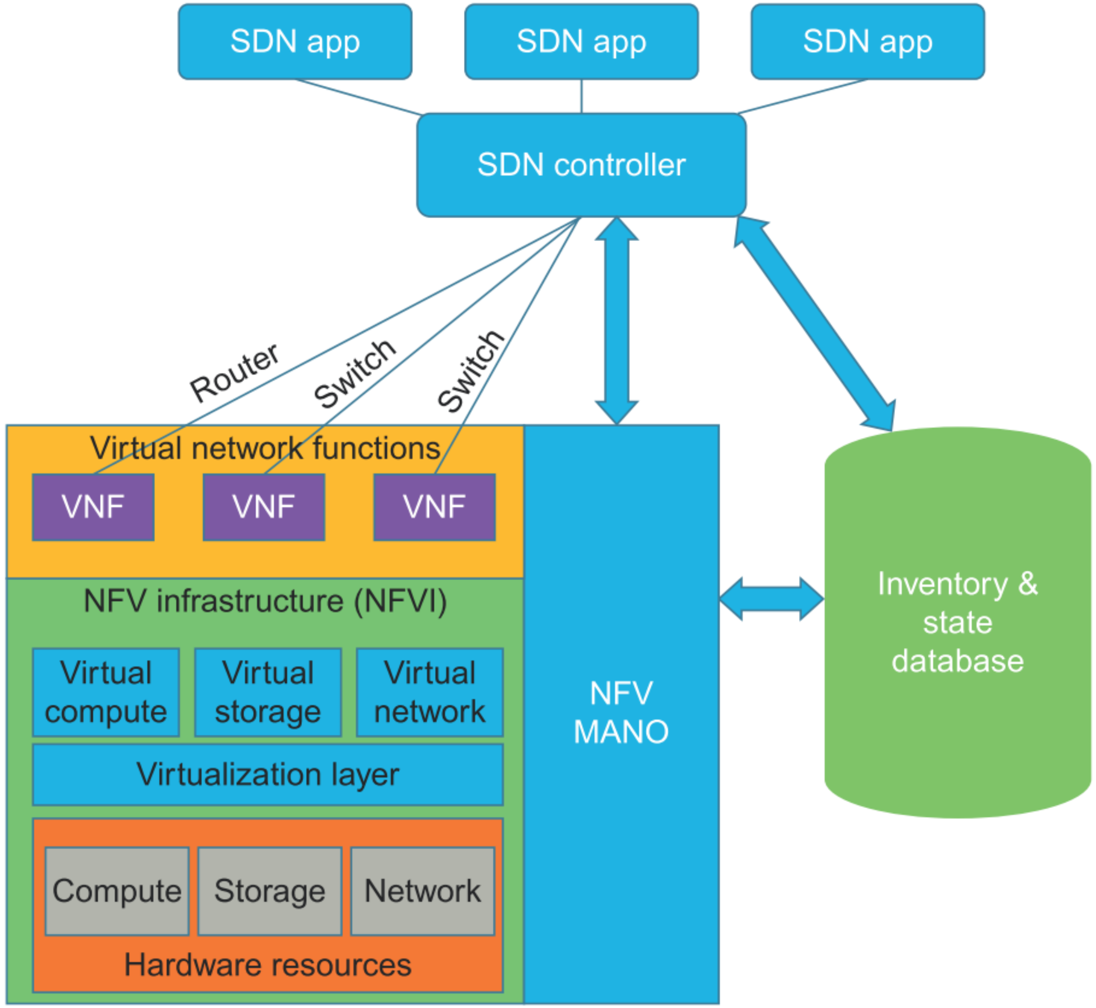
- Fragmented noncommodity hardware
- Physical install per appliance per site
- Hardware development large barrier to entry for new vendors, constraining innovation & competition

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

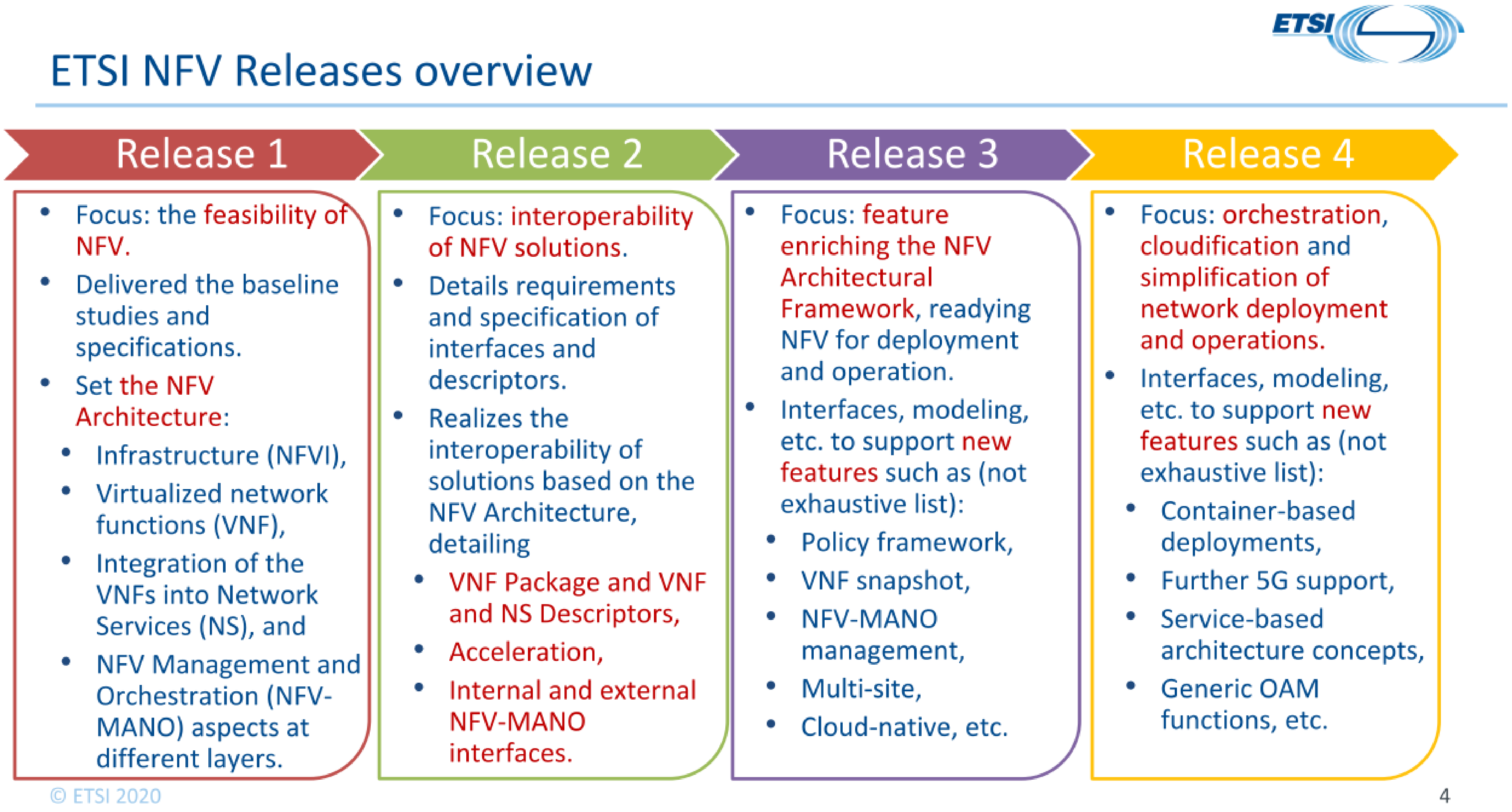
SDN and NFV


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

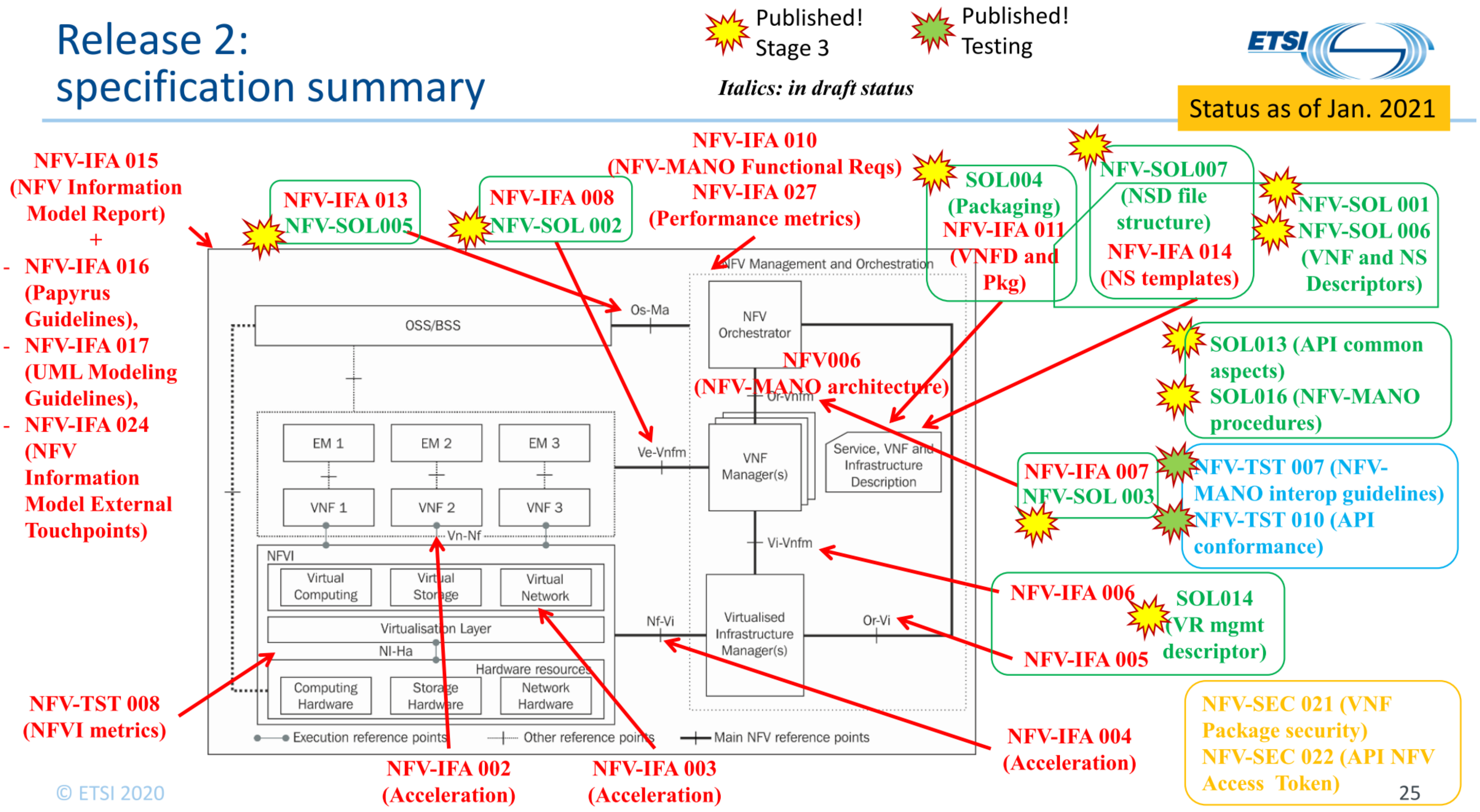
- Focus: feature enriching the NFV Architectural Framework, readying NFV for deployment and operation.
- Interfaces, modeling, etc. to support new features such as (not exhaustive list):
  - Policy framework,
  - VNF snapshot,
  - NFV-MANO management,
  - Multi-site,
  - Cloud-native, etc.

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**

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-IFA 004 (Acceleration)

NFV-TST 008 (NFVI metrics)

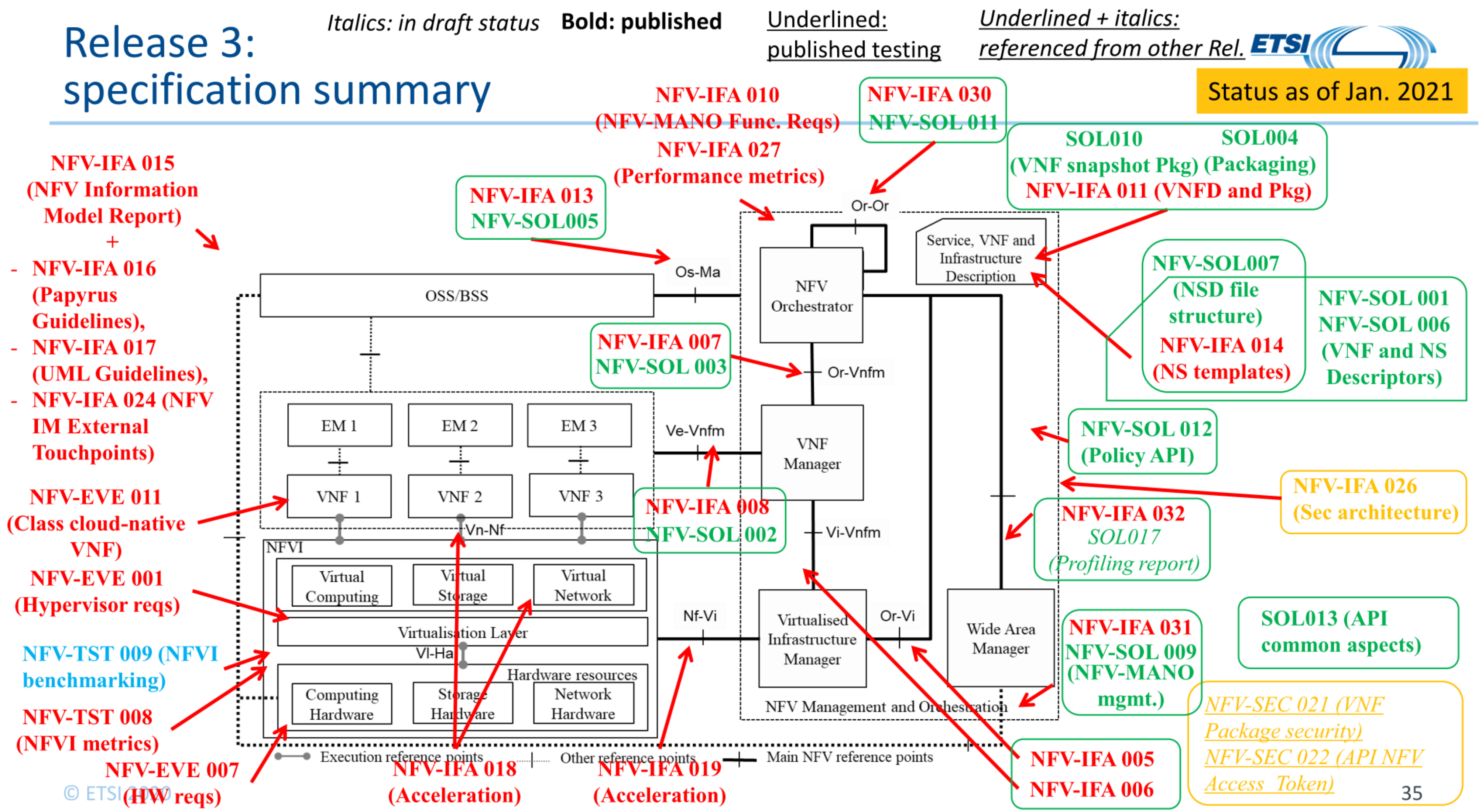
NFV-IFA 002 (Acceleration)

NFV-IFA 003 (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)

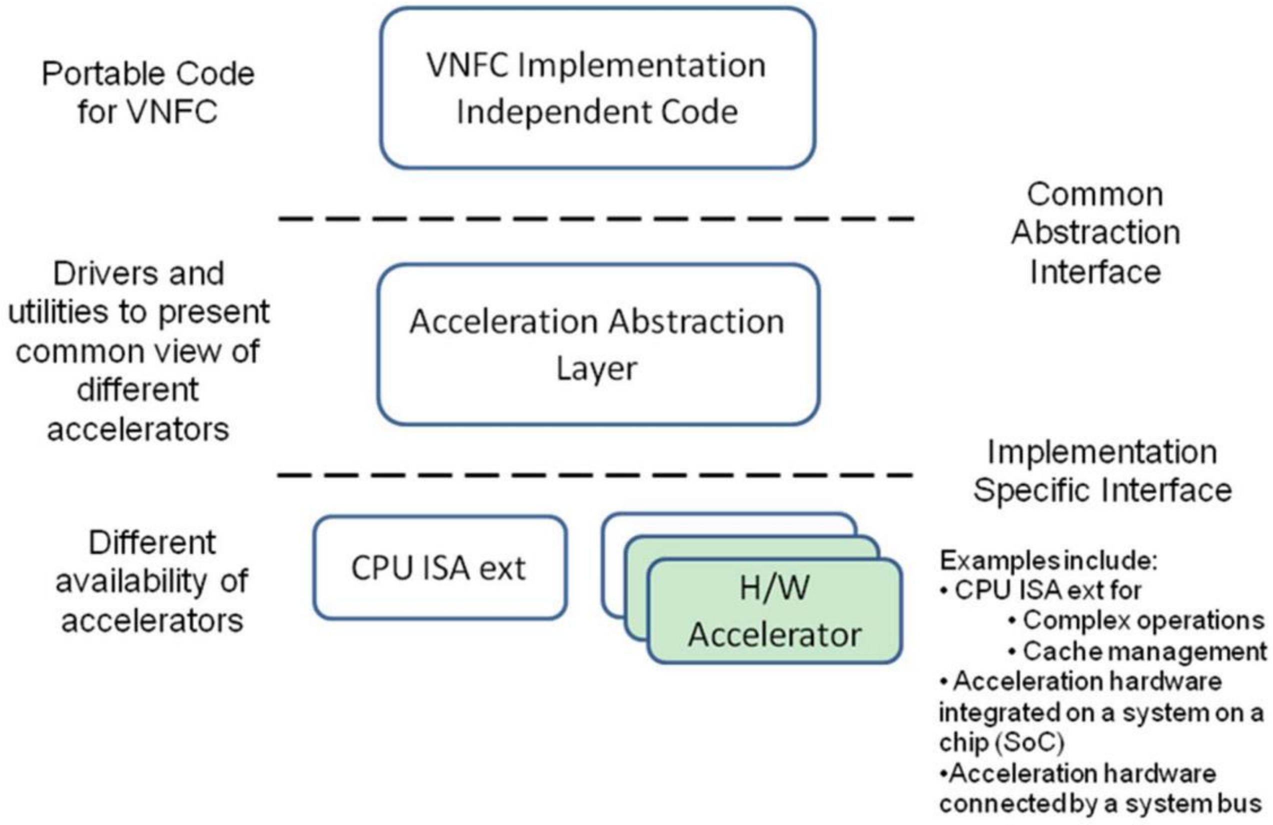
NFV-IFA 018 (Acceleration)

NFV-IFA 019 (Acceleration)

NFV-IFA 005
NFV-IFA 006

VNF Acceleration Use Cases

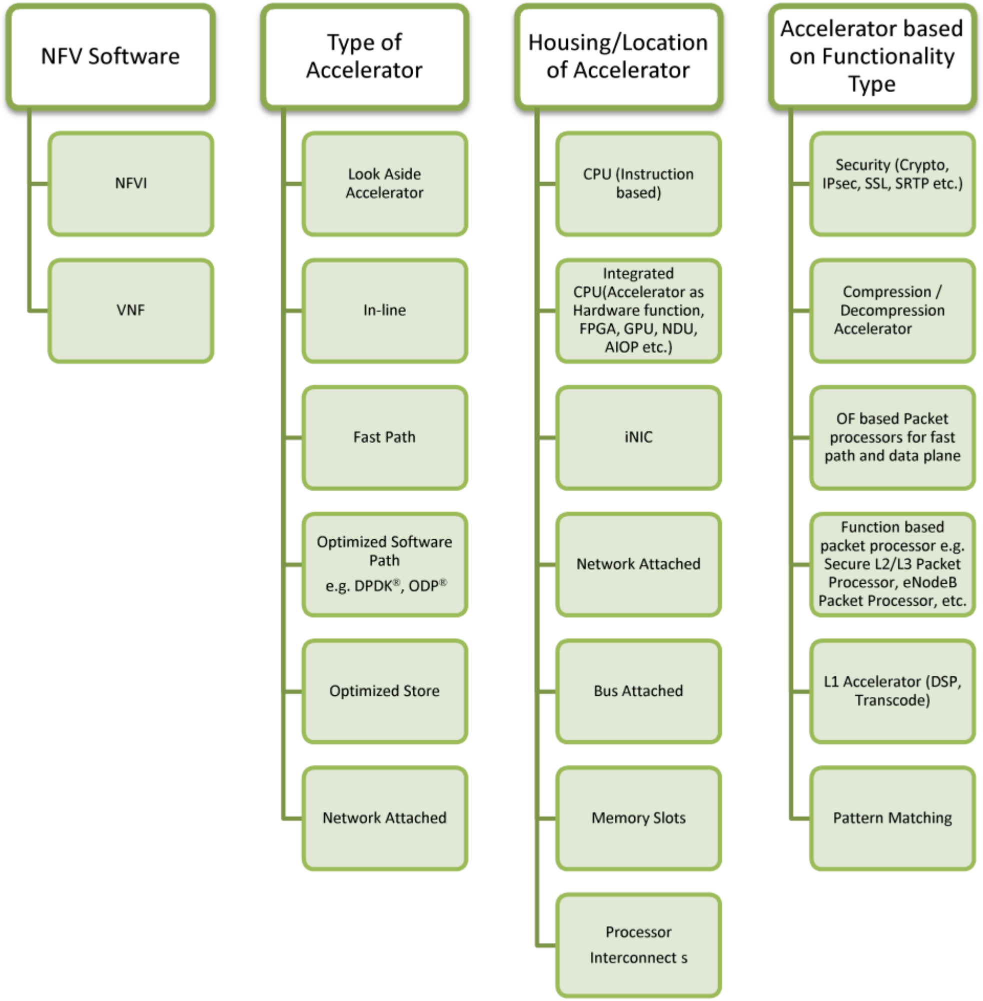
- Acceleration is not just about increasing performance.
- NFVI operators may seek different goals as far as acceleration is concerned:
  - Reaching the desirable performance metric at a reasonable price.
  - Best performance per processor core/cost/watt/square foot, whatever the absolute performance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

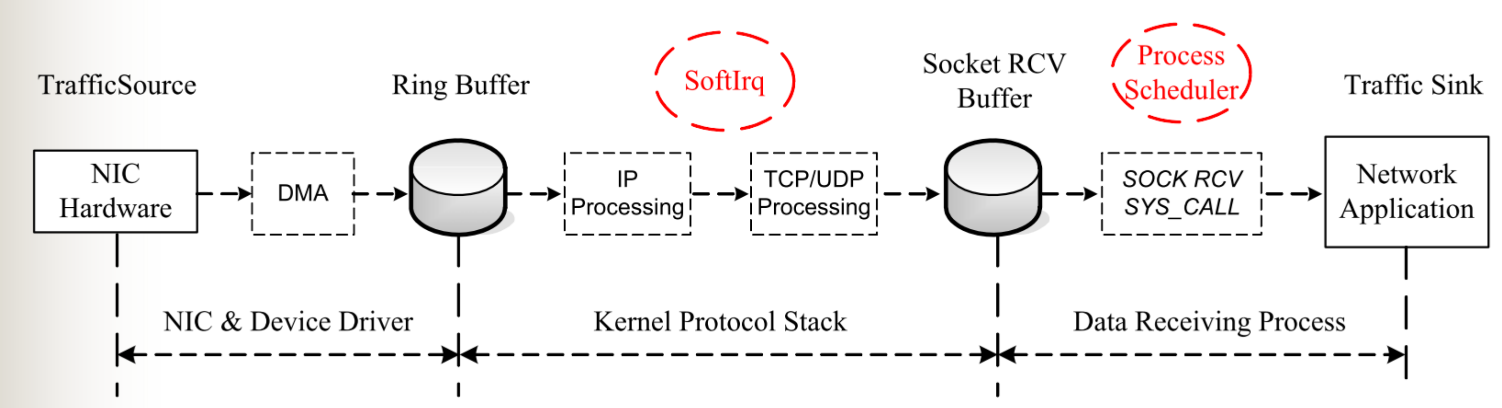
Classification of accelerators

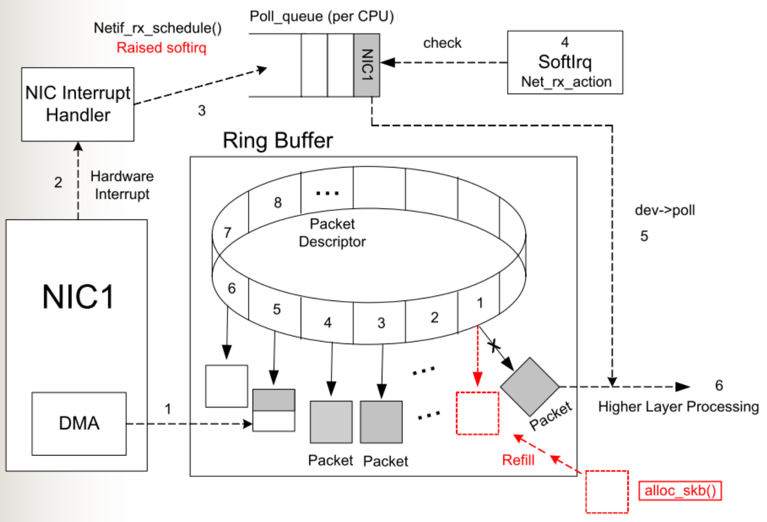


- NFV Software
  - NFVI
  - VNF
- Type of Accelerator
  - Look Aside Accelerator
  - In-line
  - Fast Path
  - Optimized Software Path e.g. DPDK®, ODP®
  - Optimized Store
  - Network Attached
- Housing/Location of Accelerator
  - CPU (Instruction based)
  - Integrated CPU(Accelerator as Hardware function, 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

Linux Packet Receiving process





NIC & Device Driver Processing Steps

1. Packet is transferred from NIC to Ring Buffer through DMA
2. NIC raises hardware interrupt
3. Hardware interrupt handler schedules packet receiving software interrupt (Softirq)
4. Softirq checks its corresponding CPU's NIC device poll-queue
5. Softirq polls the corresponding NIC's ring buffer
6. Packets are removed from its receiving ring buffer for higher layer processing; the corresponding slot in the ring buffer is reinitialized and refilled.

Source: Wu, Wenji, Crawford, Matt, Bowden, Mark, and /Fermilab. The performance analysis of linux networking - packet receiving. United States: N. p., 2006. Web.

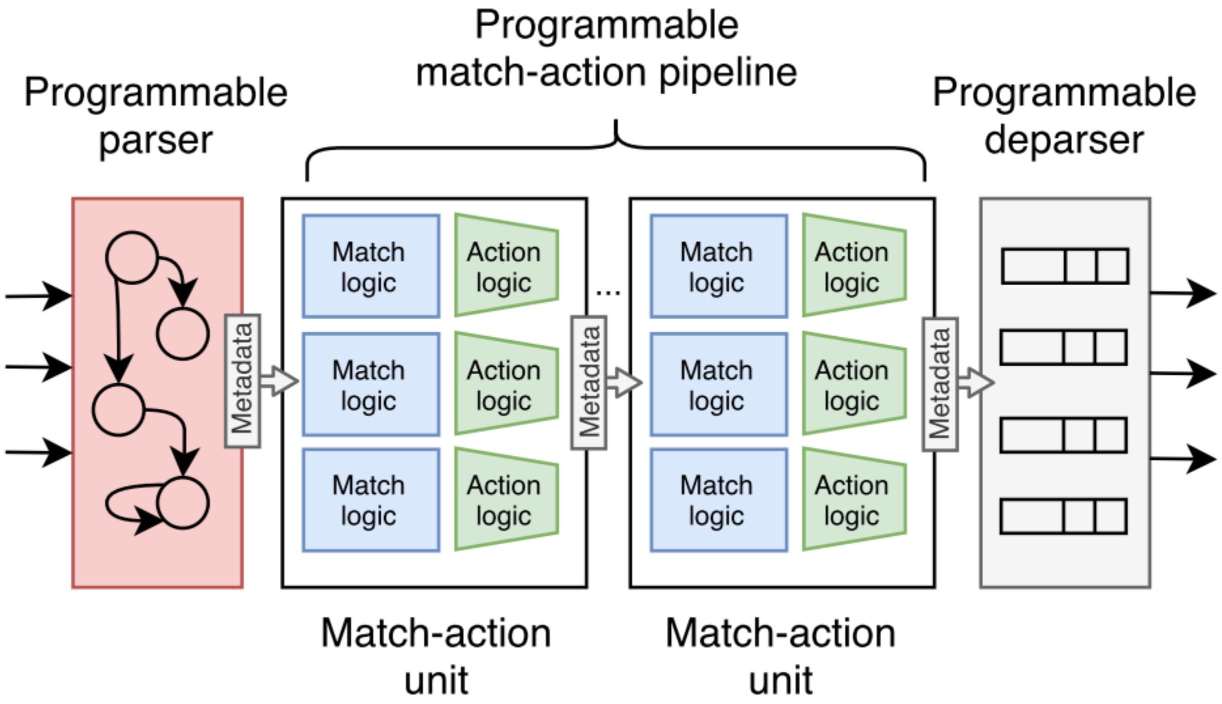
Network Data Plane Programmability

- Programmability
  - the ability of the software or the hardware to execute an externally defined processing algorithm.
- Network programmability
  - the ability to define the processing algorithm executed in a network and specifically in individual processing nodes, such as switches, routers, load balancers, etc.
- Data plane models have been proposed as abstractions of the hardware
- Data plane programming languages are tailored to those data plane models and provide ways to express algorithms for them in an abstract way
- The resulting code is then compiled for execution on a specific packet processing node supporting the respective data plane programming model.

Source: https://arxiv.org/abs/2101.10632

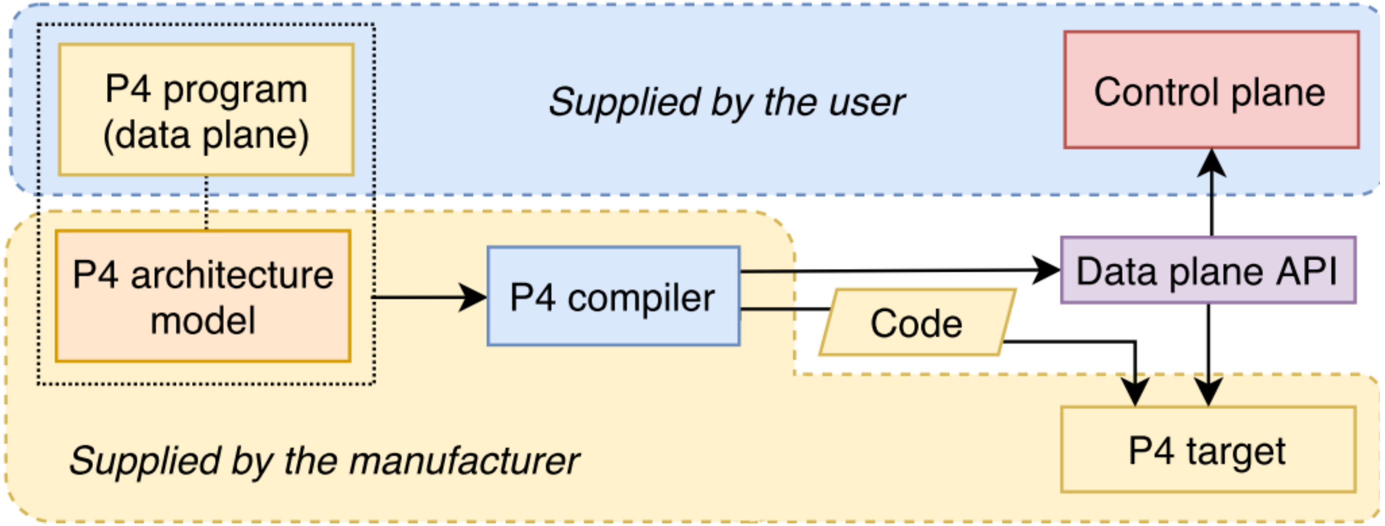
Protocol-Independent Switching Architecture (PISA)

- Based on the concept of a programmable match-action pipeline
- The programmable parser allows programmers to declare arbitrary headers together with a finite state machine that defines the order of the headers within packets.
- The programmable match-action pipeline consists of multiple match-action units. Each unit includes one or more match-action-tables (MATs) to match packets and perform match-specific actions with supplied action data.
  - Each MAT includes matching logic coupled with the memory (static random-access memory (SRAM) or ternary content-addressable memory (TCAM)) to store lookup keys and the corresponding action data.
- In the programmable deparser , programmers declare how packets are serialized.



Data Plane Programming Languages

- A packet, processed by a PISA pipeline, consists of packet payload and packet metadata. PISA only processes packet metadata that travels from the parser all the way to the deparser but not the packet payload that travels separately.
- P4 (Programming Protocol-Independent Packet Processors) is the most widely used domain-specific programming language for describing data plane algorithms for PISA.

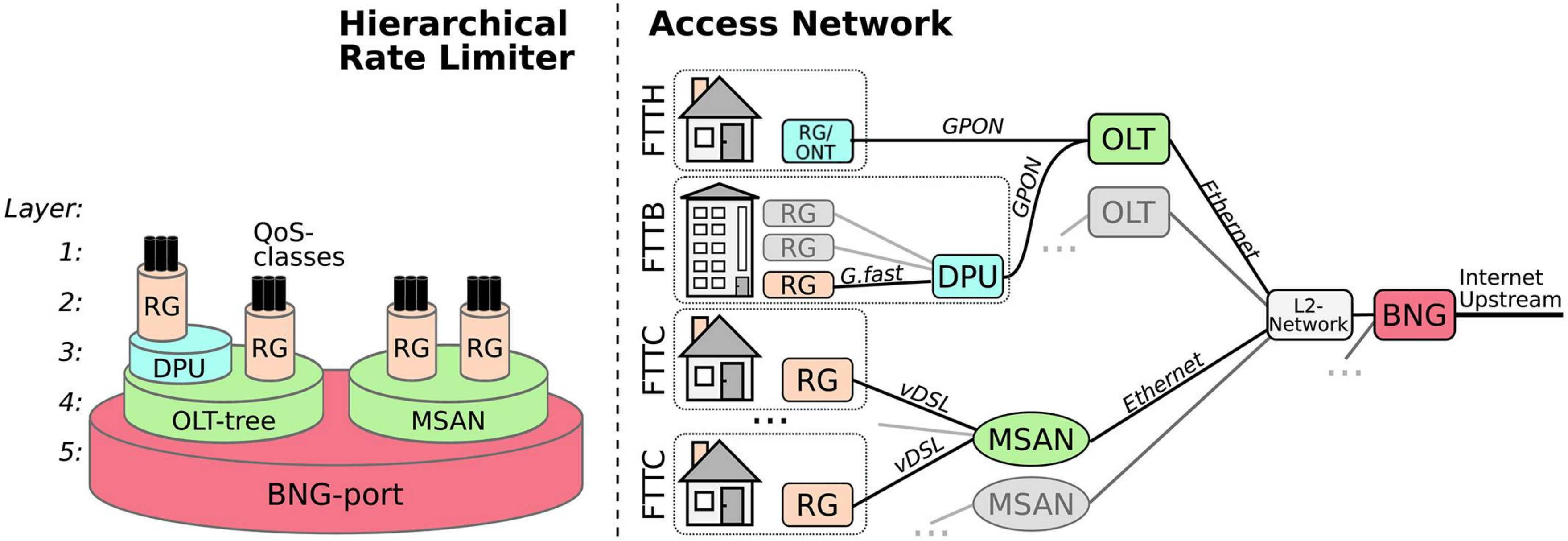


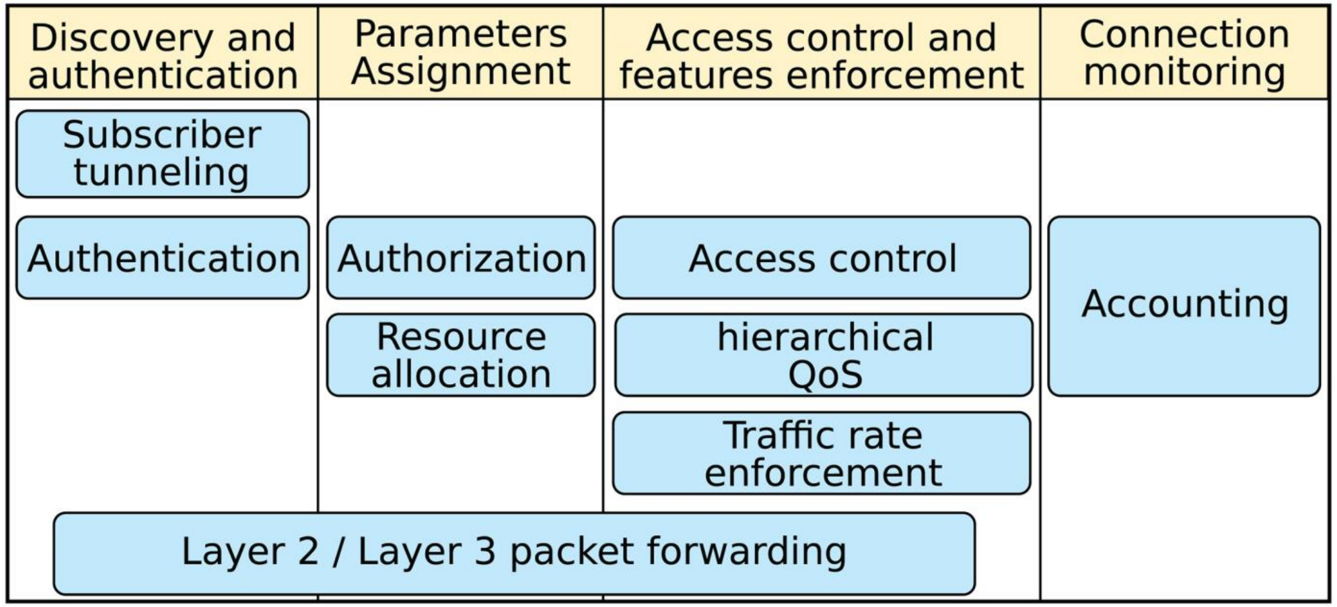
Example: Open Broadband Network Gateway (openBNG)

- Implementing NFV on commodity processors does not match the performance requirements of the high-throughput data plane components in large carrier access networks
- Programmable hardware architectures like field programmable gate arrays (FPGAs), network processors, and switch silicon supporting the flexibility of the P4 language offer a promising way to account for both performance requirements and the demand to quickly introduce innovations into networks

Source: Ralf Kundel, Leonhard Nobach, Jeremias Blendin, Wilfried Maas, Andreas Zimber, Hans-Joerg Kolbe, Georg Schyguda, Vladimir Gurevich, Rhaban Hark, Boris Koldehofe, Ralf Steinmetz, "OpenBNG: Central office network functions on programmable data plane hardware", DOI: 10.1002/nem.2134

OpenBGN functions



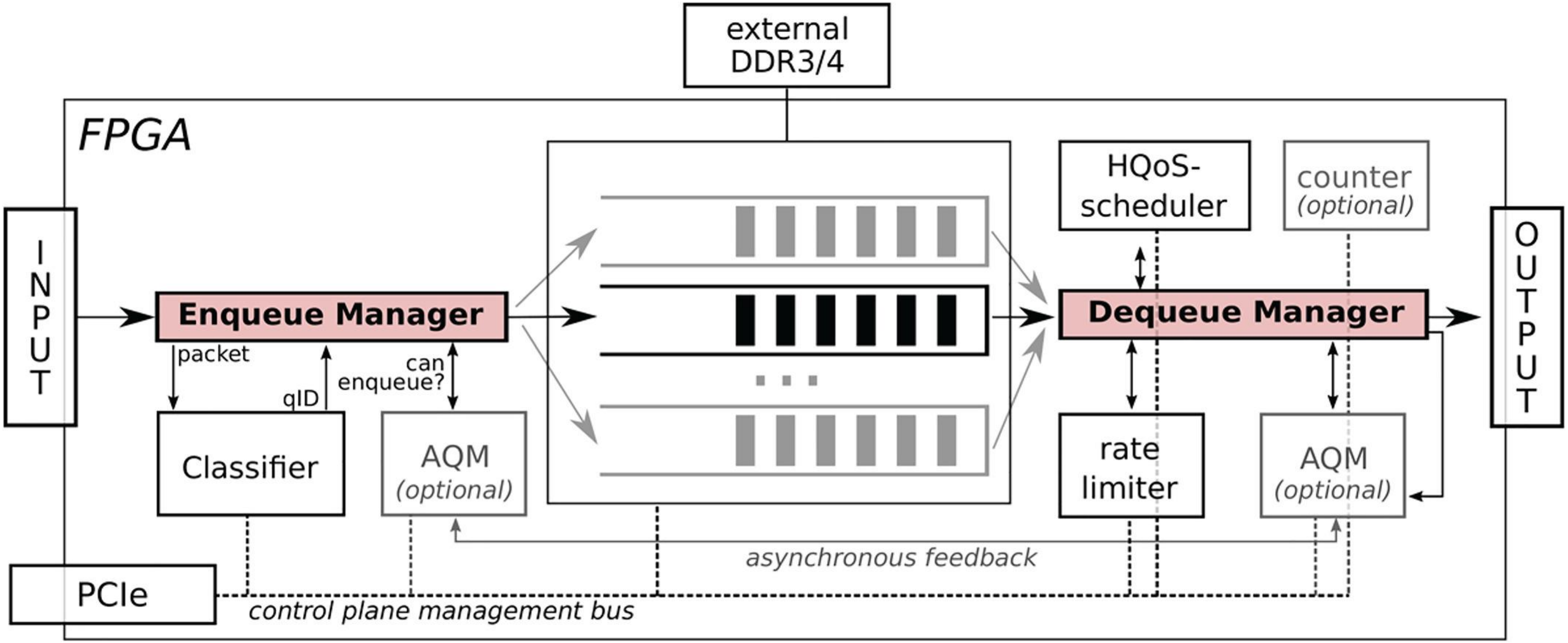


Discovery and authentication	Parameters Assignment	Access control and features enforcement	Connection monitoring
Subscriber tunneling			
Authentication	Authorization	Access control	Accounting
	Resource allocation	hierarchical QoS	
		Traffic rate enforcement	
Layer 2 / Layer 3 packet forwarding			

OpenBGN Hierachical QoS Implementation

- Design overview of an FPGA-based QoS and shaping processor.
  - After being processed by the P4-BNG pipeline, packets will enter the FPGA on the INPUT side.
  - The enqueue manager decides, supported by a classifier and an optional Active Queue Management (AQM) module, if and in which queue a packet should be inserted.
  - While being queued, packets are stored in external DRAM.
  - The dequeue manager decides, which packet should be sent next and if packets should be dropped from a queue by an AQM algorithm.
  - Parameter configuration can be done via PCIe

Architecture



Conclusions

- Edge-cloud continuum
- In network computing
- SDN
- NFVI
- Network data plane programmability
- Flexibility+performance

Seasonal School Artist 2022/2023

- https://www.santannapisa.it/it/seasonalschool/artist-5g6g-networks-transforming-digital-society

BRAINE

ECSEL
Joint Undertaking

justinecris.borromeo@santannpisa.it,
luca.valcarenghi@santannapisa.it

THANKS !!!