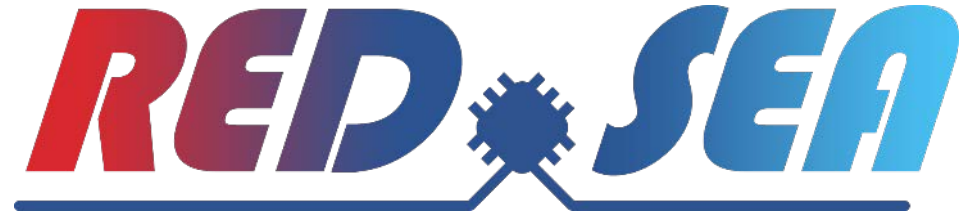
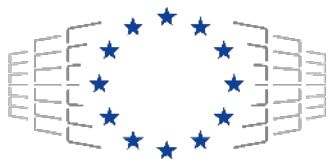


redsea-project.eu | @redsea_EU | @redsea-project

RED-SEA: Hardware and Software Perspectives for new Generations of Exascale interconnects

Andrea Biagioni, INFN

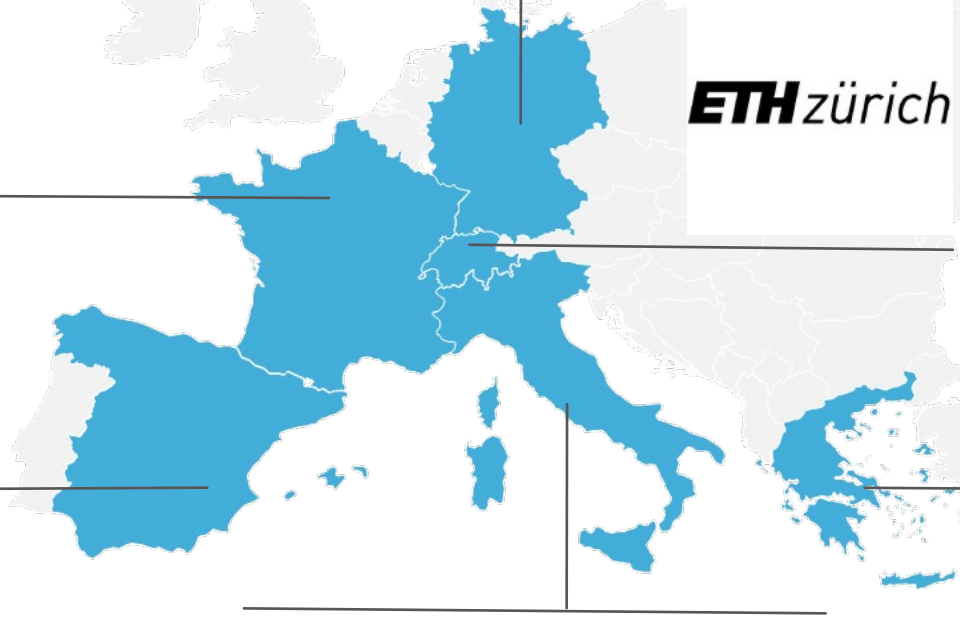
EuroHPC JU Projects Shaping Europe's HPC Landscape



EuroHPC Joint Undertaking

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 955776. The JU receives support from the European Union's Horizon 2020 research and innovation programme and France, Greece, Germany, Spain, Italy, Switzerland.

The RED-SEA consortium



1. Network and interconnects
2. Performance Evaluation
3. Simulation frameworks
4. Hardware design and tools
5. HPC System and integration

Project start: 01/04/2021
Project duration: 36 months
Project budget: 8 M€

We are one of the “SEA” projects

3 complementary projects addressing Exascale challenges in a Modular Supercomputing Architecture (MSA) context

- In line with several HW/SW Exascale projects funded under previous European programmes
- Funded by the EuroHPC 2019-1 call focused on SW and applications
  - The EuroHPC Joint Undertaking targets Exascale computers in Europe in 2023-24
  - Should contain as many European components as possible
- Coordinated with other on-going European projects, particularly the European Processor Initiative

DEEP-SEA: DEEP Software for Exascale Architectures

- Better manage and program compute and memory heterogeneity
- Targets easier programming for Modular Supercomputers
- Continuation of the DEEP projects series

IO-SEA: Input/Output Software for Exascale Architectures

- Improve I/O and data management in large scale systems
- Builds upon results of SAGE1-2 projects and MAESTRO

RED-SEA: Network Solution for Exascale Architectures

- Develop European network solution
- Focus on BXI (Bull eXascale Interconnect)

RED-SEA objectives

Enable

Enable the design of a new generation of high performance network interconnect

- Exploiting existing European technology both in the academic and industrial field (BXI, ExaNeSt, ExaNet)
- Able to power the future EU Exascale systems

Explore

Explore new innovative solutions

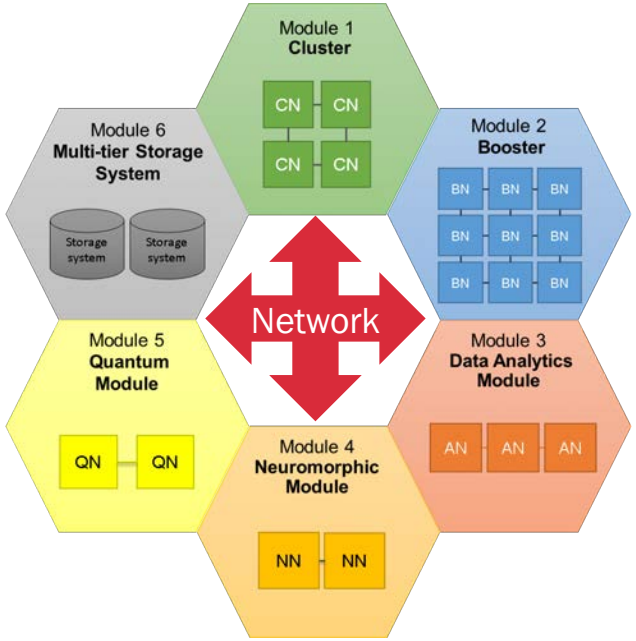
- End-to-end network services – from programming models to reliability, security, low latency, and new processors
- Management of data traffic (congestion generated by collective communication), QoS delivery mechanism

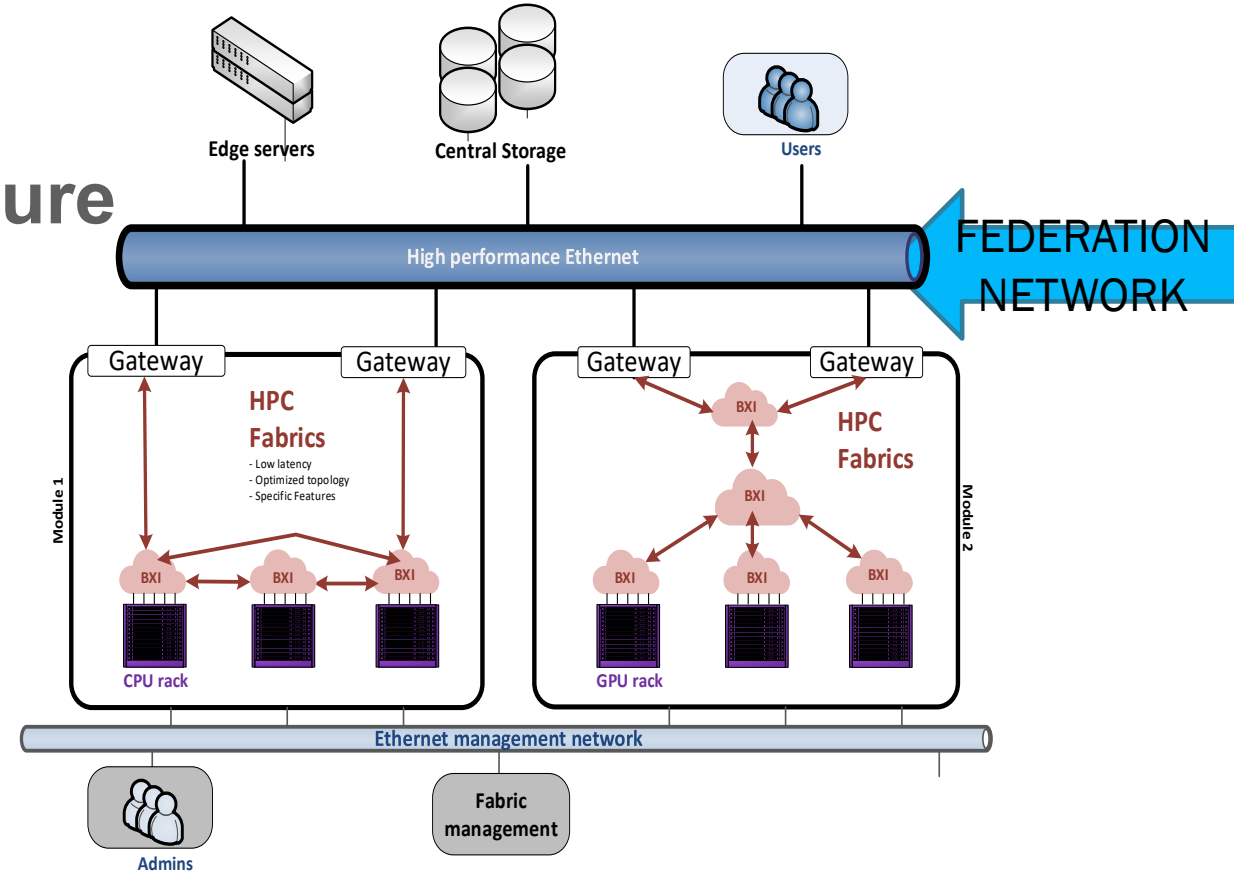
Develop

Develop the ecosystem and create a broader community of users and developers combining Research and Industrial teams

- Leveraging open standard and compatible API to develop innovative re-useable libraries and Fabrics management solutions

RED-SEA: MSA network architecture



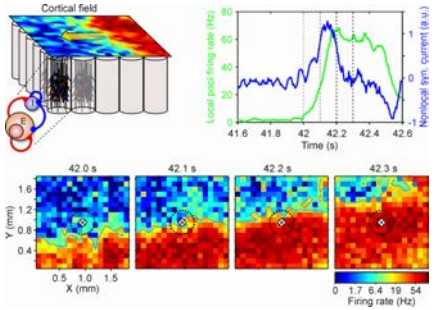


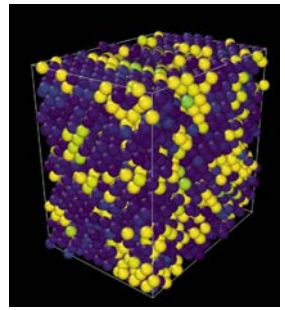
- HPC (High Performance Computing) ; HPDA (High-Performance Data Analytics); AI (Artificial Intelligence)
- Supercomputer: aggregation of resources that are organized to facilitate the mapping of applicative workflows
- HPC is part of the continuum of computing

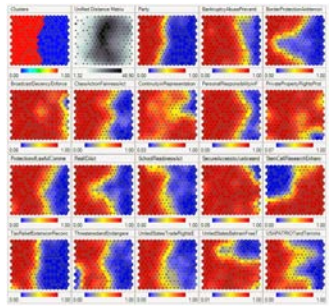
- High performance Ethernet as federation network featuring state-of-the-art low latency RDMA communication semantics;
- BXI as the HPC fabric consisting of two discrete components, a BXI NIC plus a BXI switch, and the BXI fabric manager.

RED-SEA: methodology for Co-Design Activity

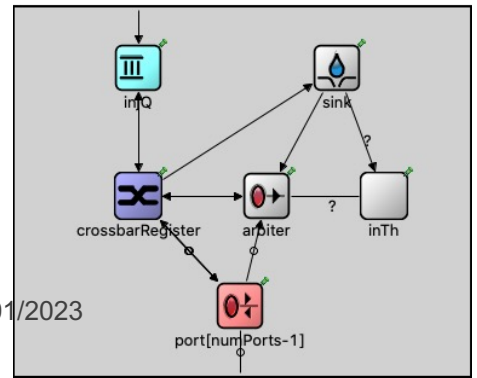
- **Application portfolio**
  - NEST: simulator for spiking neural network models
  - LAMMPS: molecular dynamic engine with focus on material modelling
  - SOM: artificial neural networks used in the context of unsupervised ML







- **Benchmark portfolio**
  - GSAS: Global Shared Address Space environment provides a shared memory abstraction model to distributed applications
  - DAW: stress the NI capabilities at scale and the QoS capabilities of the interconnect
  - LinkTest: scalable benchmark for point-to-point communications
  - PCVS: validation engine designed to evaluate the offloading capabilities of high-speed network
- **Collection and Analysis of MPI Network Traces generated by applications**
  - VEF traces + DIBONA (12 nodes, 768 ARM cores, BXI interconnect)
  - Requirements for the applications and co-design recommendations
- **Simulator as reference to support the design and implementation of novel IPs proposed in the project**
  - Network traces feed the project simulators
  - Extrapolation of the behaviour at large scales (up to 10k nodes)



Hardware Testbeds

TESTBED	Features	Outcome	Availability Date	Remote Access
DIBONA	4 blades; 768 Arm v8 cores (12 nodes) OS: RHEL 8.4 Memory: 256GB per Node: 16x16GB DDR4@2666MT/s	Analysis of BXI 1.3 • net. Traces of apps • benchmarks	16 November 2021	YES
DEEPcluster	2CN + BXI switch	T1.2: partec	Q4 2021	YES
ExaNeSt	64 arm cores; 16 QFDB; 4 mezzanines	Prototype of FORTH RDMA + cong. mgmt	Q4 2021	NO
INFN-dev	Alveo board (u50; u200; U280) PCIe gen3/gen4 I/O 100gbps (APElink; BXI-link) ExaNet protocol compliant	• Prototype of APEnetX • Debug & development INFN WP3 and WP4 IPs	Q3 2021 APEnet v6 (0.1): Q4 2022	NO
TGCC KNL	828 nodes (276 blades) Intel(R) Xeon Phi(TM) CPU 7250 96 Go of memory (6x16) + 16 Go mcdram OS: RHEL-7.9; interconnect: BXI v1.2	VEF traces / BXI traces	now (only to CEA partner and subject to quota availability)	YES (up to 14/11/22)
INTI-BXI	nodes (AMD rome); 2*64 cores/node Mem: 240Go /node 4 BXI NICs /node	WP4 – T4.5 multirail	Q1 2022	No Only to CEA

Table with all simulators

Simulator (partner)	Features	Tasks involved
COSSIM (EXAPSYS)	**Current** • Processing in ARM, RISC-V (work-in-progress/eProcessor), Intel (deprecated) • Network topologies, routing algos, switches, etc are those supported by OMNET++ • Main change of OMNET++ has to do with INET packages that have been adapted so as to support full IP, Linux-compatible packets (e.g. including payload) **RED-SEA:** • NIC Architectural model with several implementation details needed • Interconnection scheme of CPU with NIC	**T1.4 :** MPI packets generated in COSSIM can be integrated in SAURON (VEF Traces) Identify if COSSIM can be connected to SAURON instead of plain OMNET++ From **WP2** get NIC design compatible with GEM5
SAURON (UCLM)	**Current:** • Network topologies: Fat-trees, Dragonflies, Slim-flies, KNS, etc. • Routing algorithms: deterministic (D-mod-K, DESTRO), Oblivious (VLB), and adaptive (PAR, UGAL, Fully, ARNs, etc.) • Switch buffer organizations (input-queued, virtual output queues, etc.) • Congestion management and QoS models • Compatible with **VEF Traces Framework** **RED-SEA:** • BXI3 Architecture (NIC and switch) • Protocols designed in WP3 and WP4	**T1.4 :** - Migration to OMNET++ 6.0 - Exploring connection with COSSIM All the tasks in **WP3:** modeling new network management proposals **T4.1:** modeling e2e protocols

Table with all simulator

Simulator (partner)	Features	Tasks involved
DQN_SIM (INFN)	**Current** • Simulation models developed from scratch using the OMNeT++ 5.4 framework • N-dim Torus Topology • Modelled after the APEnet RDMA network architecture: data-link layer (buffers, virtual channels), network layer (VCT switching, deterministic routing (DOR), Oblivious (random) and Adaptive Routing (*ch, DQN-Routing), transport layer (packet definition, network interface). • Interface between OMNeT++ and the Ray distributed execution framework to exploit its services in order to get routing actions from the Deep Q-Network reinforcement learning agent. **RED-SEA:** • Port the models to the SAURON framework in order to assess DQN scalability and performance under realistic traffic conditions • Study the application of the DQN adaptive routing algo to other topologies and/or network architectures.	

HW testbed: Dibona

Compute Blades:
X4 Up & Running
x1 Under test

Login Server:
mb3-host

Compute Rack

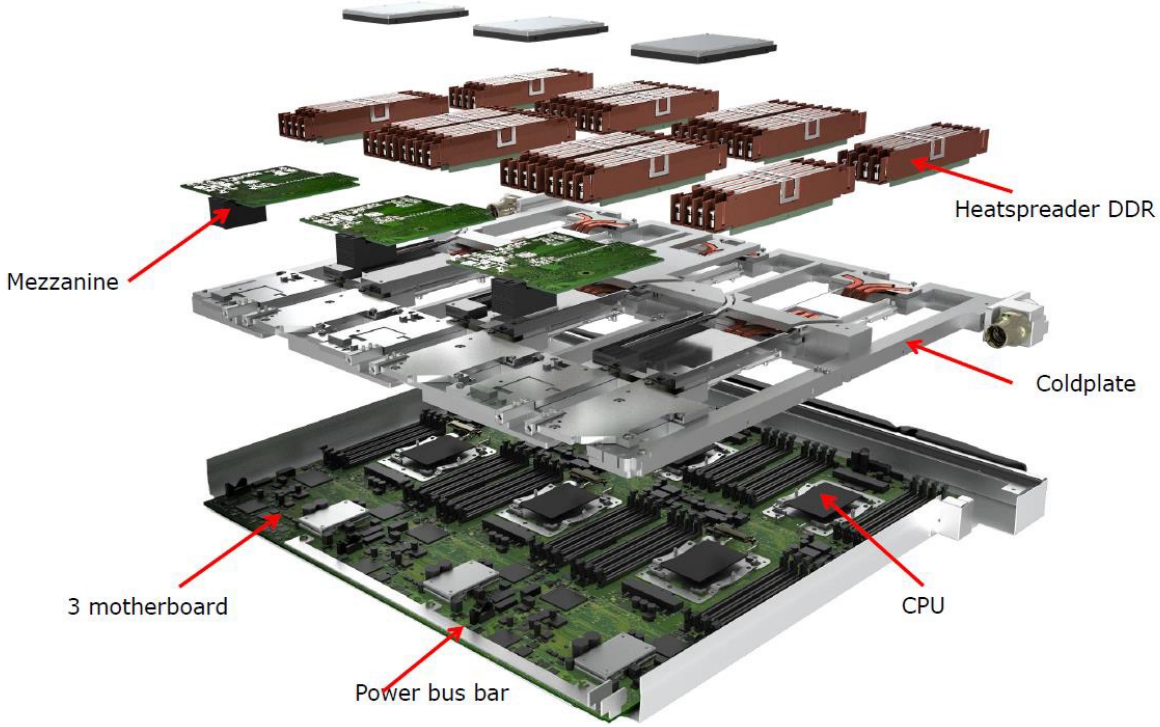
Management Node

1 BXI Switch:
Connects the 12 up & running nodes

Switch Rack

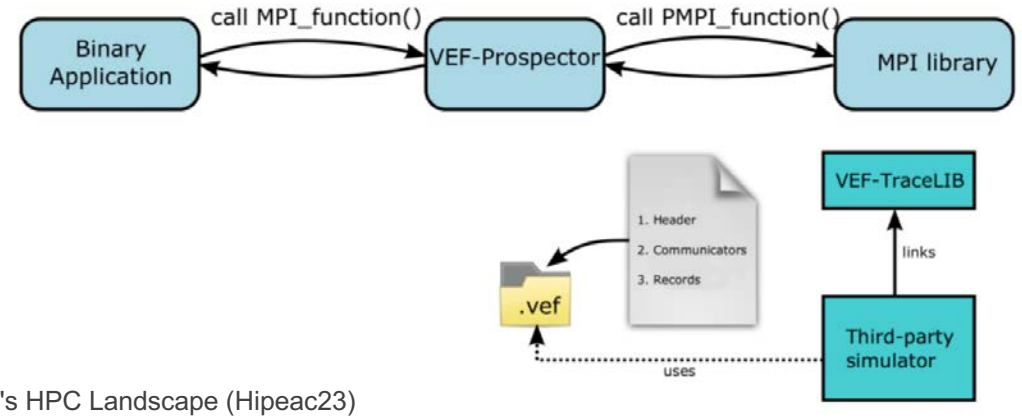
HW testbed Dibona (ATOS)

	Dibona Blade (x4)
Design	1U blade comprising 3 compute nodes side-by-side
Processors	3 Bi-socket Cavium® ThunderX2™ Armv8 processors with 32 cores @2GHz
Architecture	3 motherboard compatible with Cavium reference platform
Memory	3 x16 DDR4 memory slots (max 1024 GB with 64 GB DIMMs)
I/O Slots	BXI1.3 Port mezzanine board
Power Supply	In cabinet
Cooling	Cooling by direct contact with DLC cold plate or through heat spreaders for DIMMS
OS	Red Hat Entreprise Linux 8.4 && Smart Management Center (SMC) & Smart Software Suit (SLURM, OpenMPI)



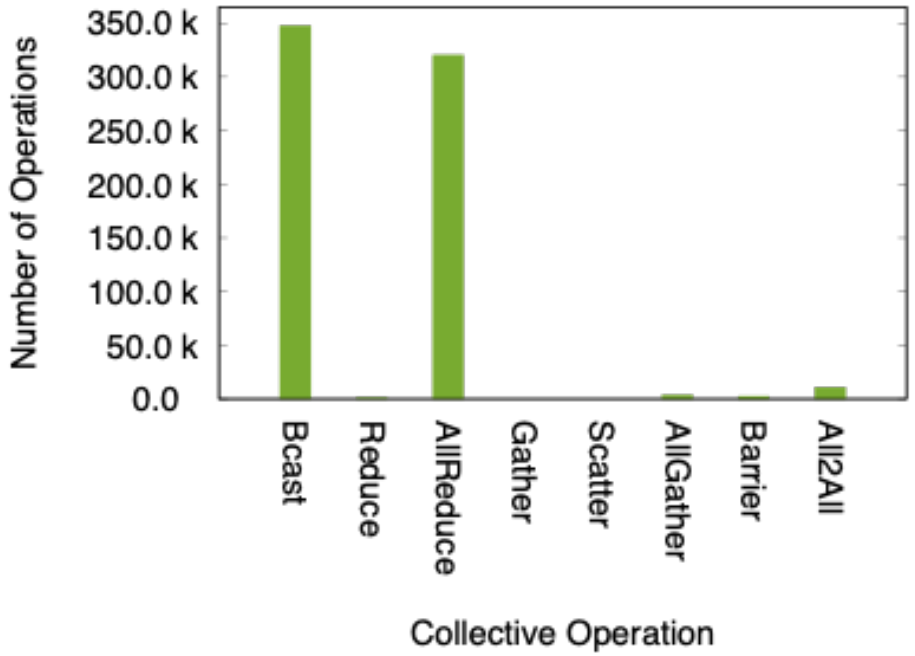
VEF-traces (UCLM)

- VEF-prospector: captures the application MPI calls, and gathers them in trace files using a special format
- TraceLib: integrating this library each simulator can reproduce the application behaviour
  - reading the VEF trace and generating the corresponding messages that can be inserted in the simulated NICs
  - running several applications simultaneously
  - Agnostic way: node architecture and timestamps of the system are not stored in the trace
- Offline (static) analysis
  - Tracetor: to test the traces obtained, analyzing the number of communication operations
  - Offline-vef-analysis.sh: to provide number of plots, text files, and PDF reports with specific information about the message generation, destination generation distribution, workloads size, number and types of collective operations

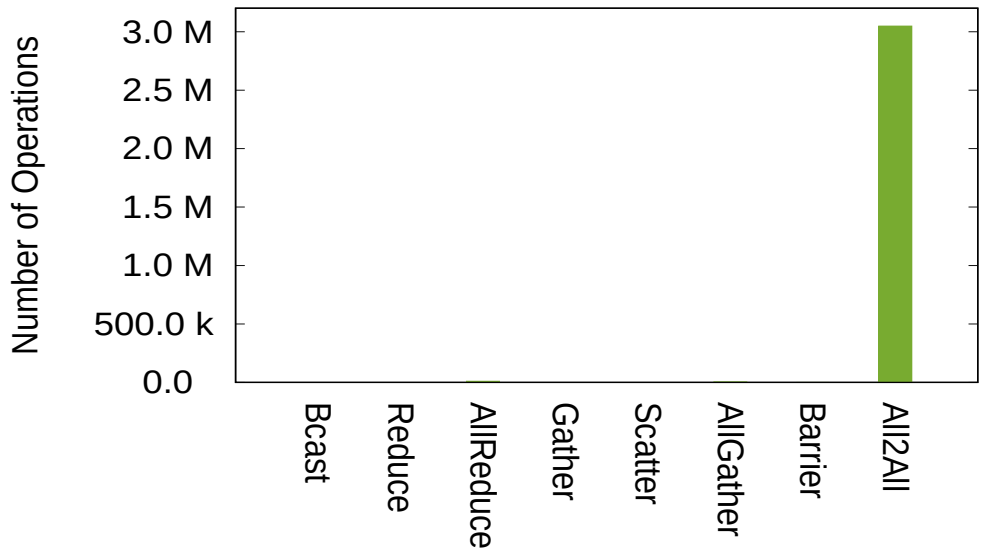


Static analysis

- Congestion characterization
  - Static analysis: different behaviour, LAMMPS wider varity of collectives, NEST dominated by All2All



LAMMPS

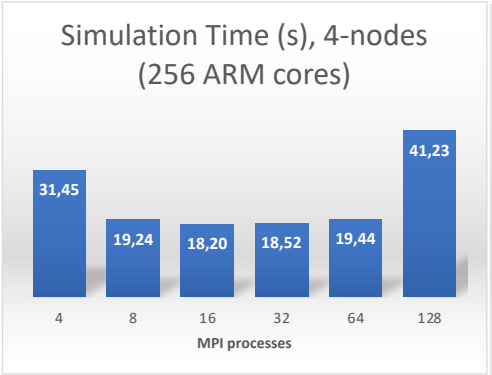


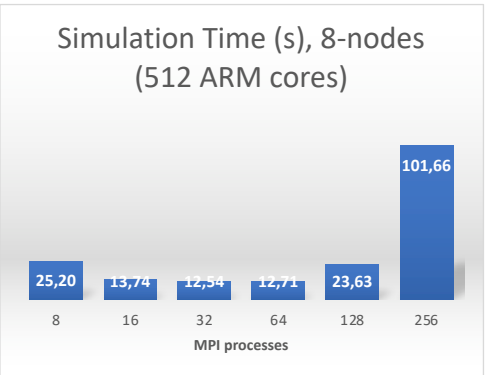
NEST

NEST on DIBONA (INFN)

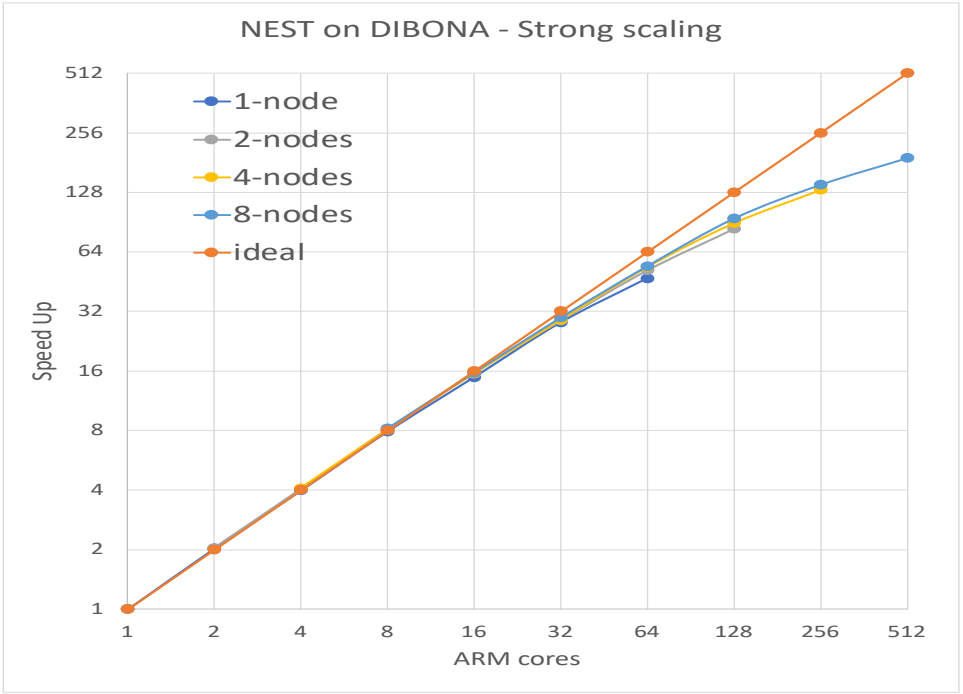
- the dynamics undergoing Slow Waves Activity of the cortex of one brain hemisphere of a mouse in a deep sleep state
- The parallelization scheme assigns neurons to different virtual processes (core) in a round-robin fashion
  - MPI processes may spread across the nodes of the system
  - OpenMP threads depends on the CPU architecture of the node
  - Total Core = MPI process X OpenMP threads
- Virtual processes belonging to different MPI processes keep their status in sync exchanging data with each other by calling MPI_Alltoall at the end of every step of integration

MPI process	OpenMP Threads	Number of messages	total bytes	average size [B]	Range of Interest	Messages in range [%]	Average Size [B]	Average Size Ratio
8	64	6,82E+04	8,02E+08	11760	512B-16kB	95,9	4854	
16	32	2,90E+05	1,24E+09	4259	256B-8kB	97,1	2558	0,53
32	16	1,19E+06	2,05E+09	1716	128B-4kB	97,8	1300	0,51
64	8	4,84E+06	3,97E+09	821	128B-2kB	96,6	718	0,55
128	4	1,95E+07	7,99E+09	410	64B-1kB	97,8	383	0,53
256	2	7,79E+07	2,06E+10	264	64B-512B	97,2	244	0,64

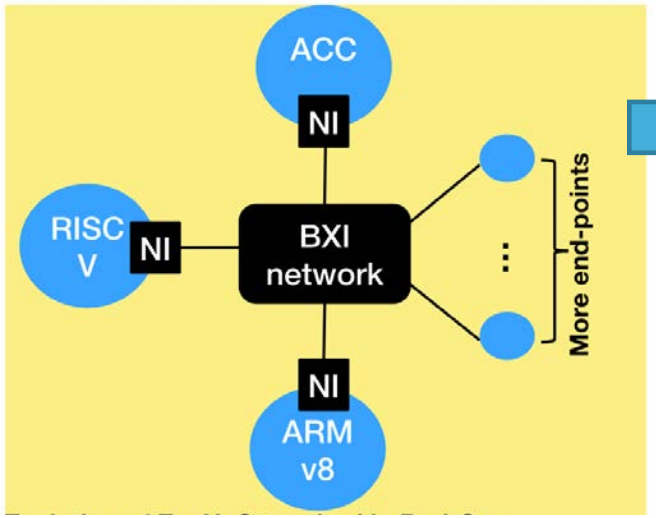




- Deviation from ideal scaling is already significant exploiting 256 ARM cores
- The NEST simulation could be distributed on a maximum of **64 MPI processes.**



BXI ECOSYSTEM: INFN APEnetX

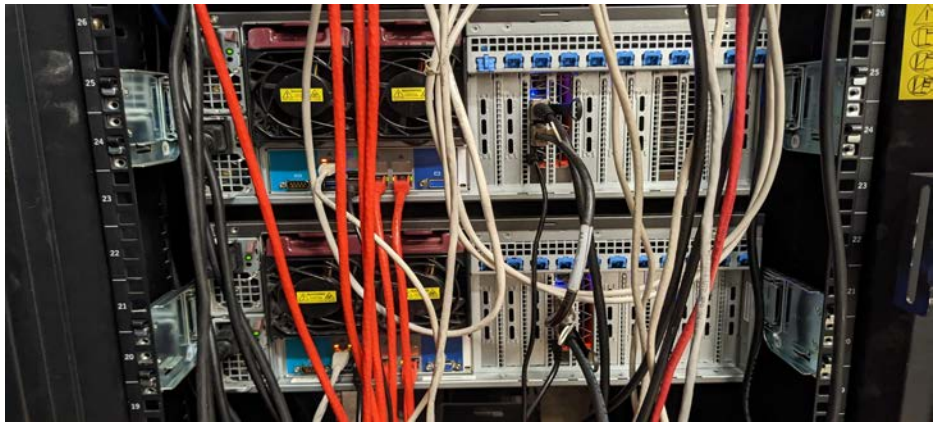


END-POINT: INFN APEnetX

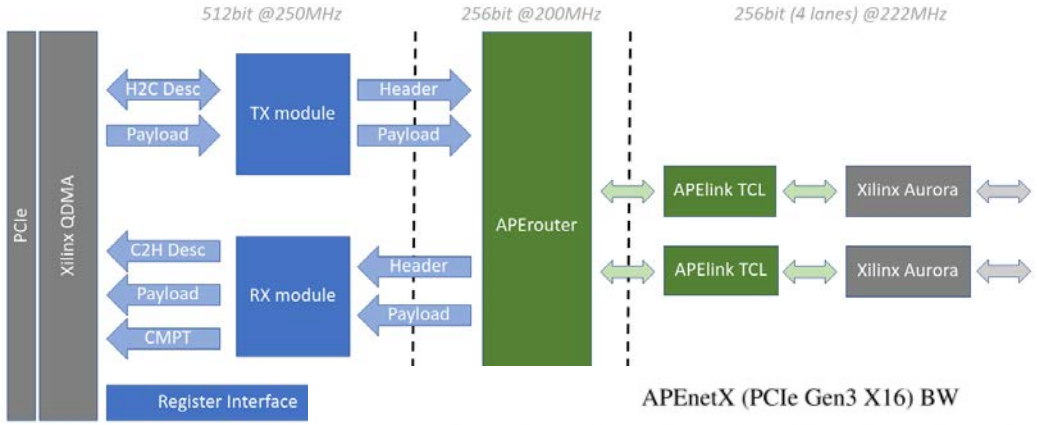
- to tightly **integrate** the network interfaces (NIs) **to RISC-V and ARMv8 cores and to FPGA-based accelerators and GPUs**
  - To prepare a number of EPI-related IPs
  - To create a highly heterogeneous programmable platform connected with state-of-the-art interconnect technologies.
- INFN duties
  - Network Interface Card (APEnetX)
    - PCIe gen4 (GPU+CPU) + BXI link (Xilinx Alveo FPGA)
  - Co-Design through applications (NEST)
- TARGET (Q4-2023)
  - Developing network IPs to optimize spiking neural network communication

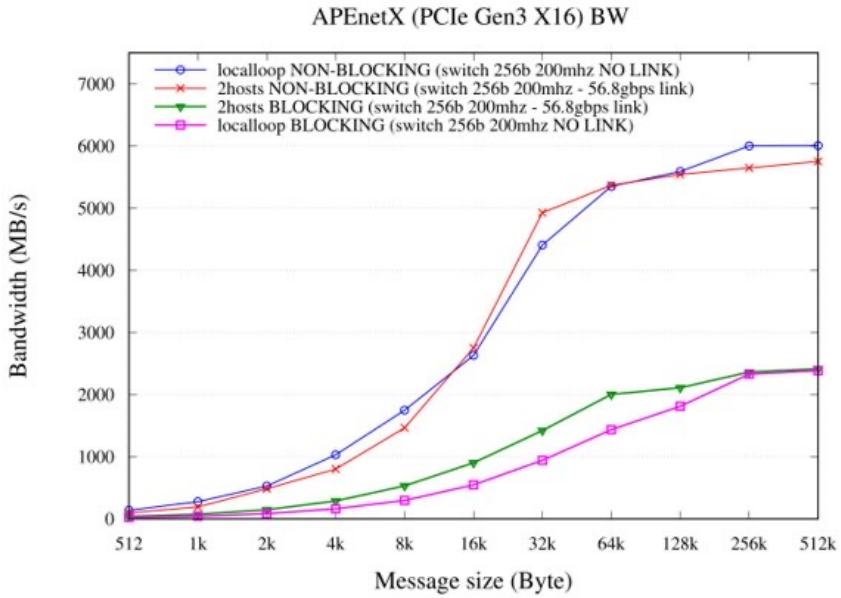
HW TESTBED: INFN

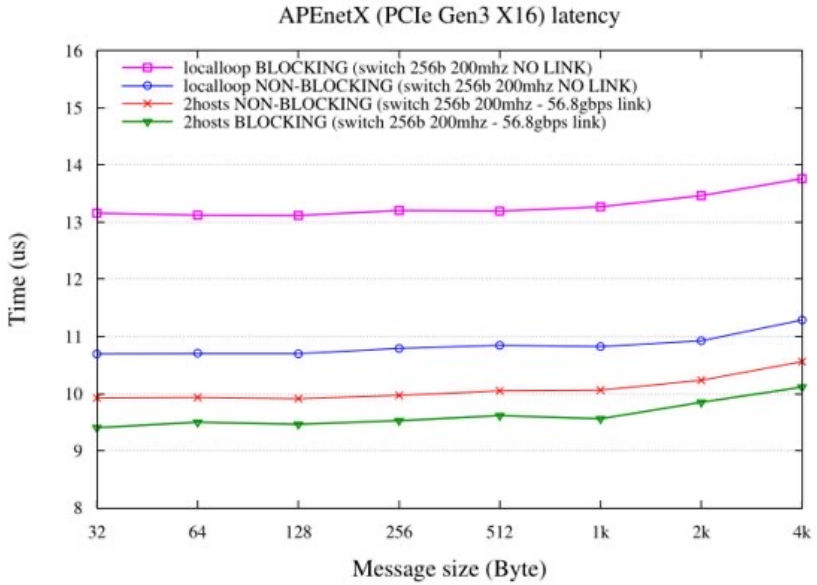
- 2x Supermicro SuperWorkstation 7049GP-TRT server
  - 2 x 8-cores 4200-series 14nm Intel Xeon Scalable Silver Processors (Cascade Lake) running @ 2.10GHz.
    - PCI gen3 support (not PCIe gen4)
  - Memory: 192GB DDR4 @3.2GHz
  - APEnetX prototype: Xilinx Alveo U200 built on the Xilinx 16nm UltraScale architecture, which natively supports the QDMA IP.
  - OS: GNU/Linux Centos 8 with Linux 4.18.0 kernel.



APEnetX architecture (INFN)





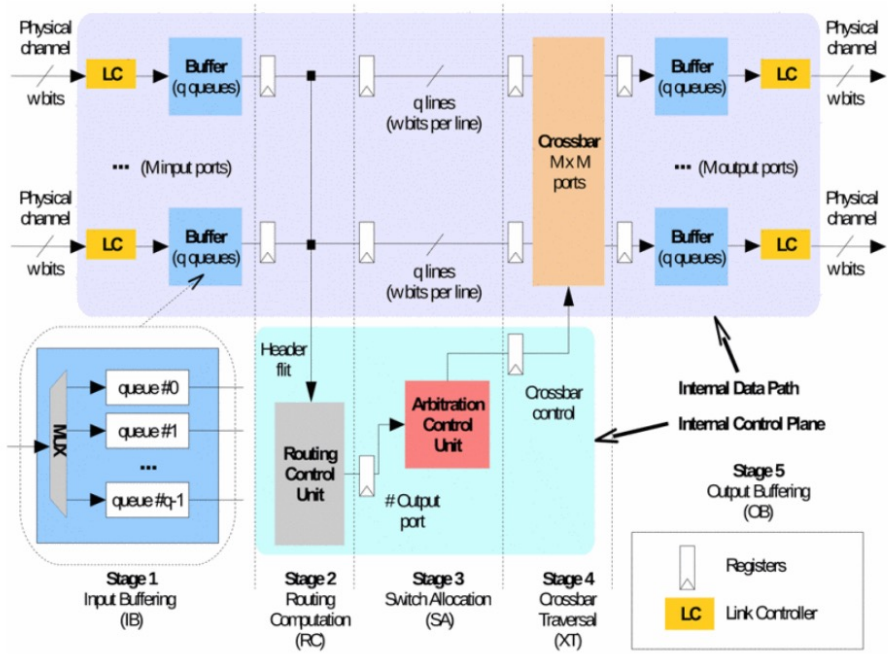


Software stack (QDMA-based)

- Xilinx open-source driver, with custom software added
- To modify as little as possible the existing Xilinx driver we used the IOCTL syscall as the entry-point for our custom addition (instead of already taken read/write):
  - Register (pin and lock) RX buffer -> IOMMU is used as address translator
  - Starting the send phase -> custom TX descriptor with bypass mode enable
  - Take RX completion -> custom completion
  - Notify the user application (polling)
  - Deregister everything when done

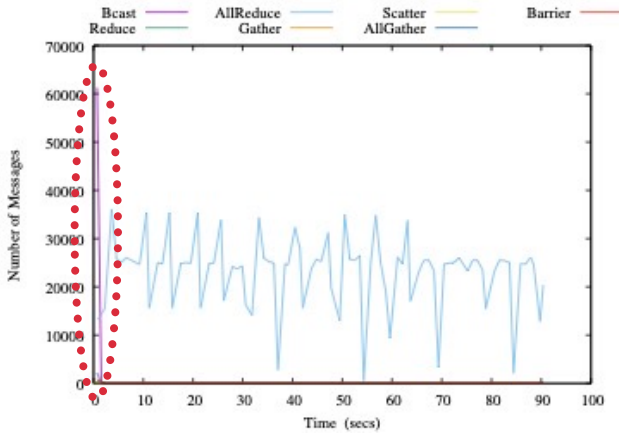
Sauron (UCLM & UPV)

- Modeling network workloads of the targeted systems:
  - **Hardware and software support for collective operations** is being modeled in the VEF TraceLib library and tested the Sauron simulator.
- Modeling the BXI's network architecture in Sauron
  - **Switch architecture** having buffers both at input and output ports (CIOQ switches), multiple queues per buffer, Stop&Go link-level flow control (e.g., PFC).
  - **BXI3 switch model** has been completed and it is being validated compared to the System-C model
- Modeling proposals in SAURON:
  - **Congestion characterization**



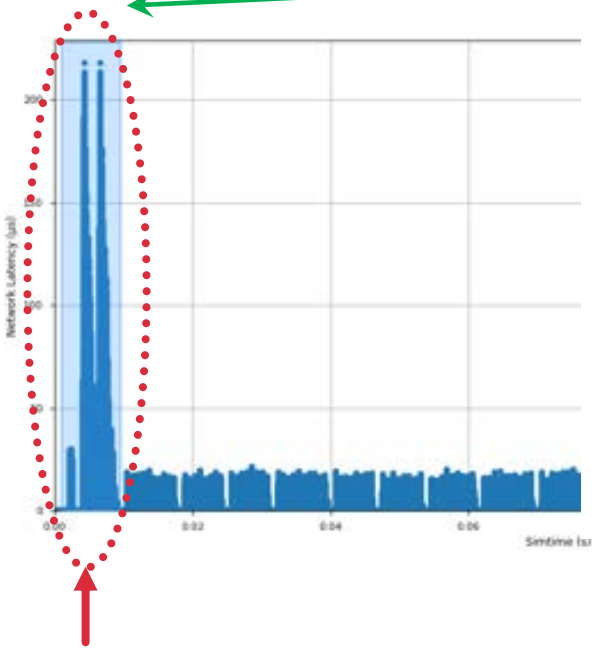
Simple Module	Description	Used in
Link Controller (LC)	It is placed both at HCAs and switches and implements flow-control policies and the packet forwarding performed by switches and HCAs.	HCA, Switch port
Buffer	It stores arriving packets in queues, the space for a whole packet in that buffer being reserved a priori by the flow-control policy.	HCA, Switch port
Router	It implements routing algorithms that determine the output port that packets must request at each switch to reach their final destination.	Switch
Crossbar registers	It stores packets which have requested an output port while they are waiting for a grant. It is connected with the buffers through the crossbar.	Switch
Arbiter	It decides and establishes which input buffers, containing packets ready for transmission, will be allowed to send those packets to their requested output ports.	Switch
Source	It is in charge of creating packets	HCA
Sink	It processes the incoming data packets received.	HCA
Scheduler	It maps packets to queues.	HCA

Dynamic Analysis

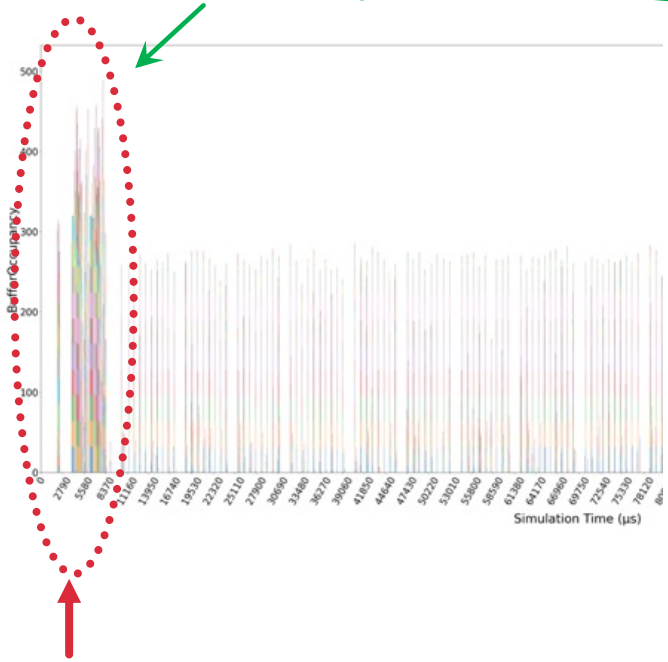


- Congestion characterization finished
  - Dynamic analysis: example LAMMPS: highly determined by the the collective primitives (type and frequency)

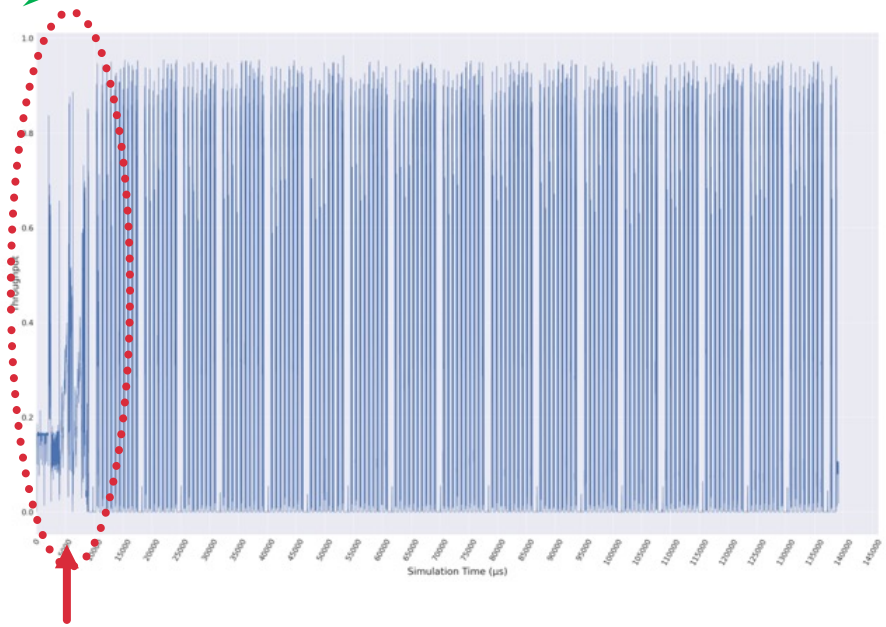
Broadcast phase



High network latency

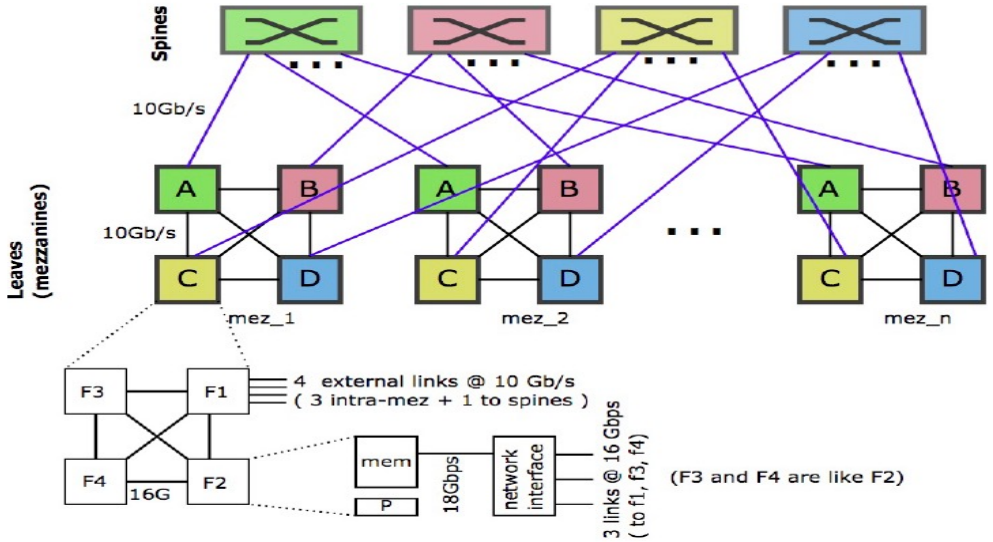


Long queues at the switches

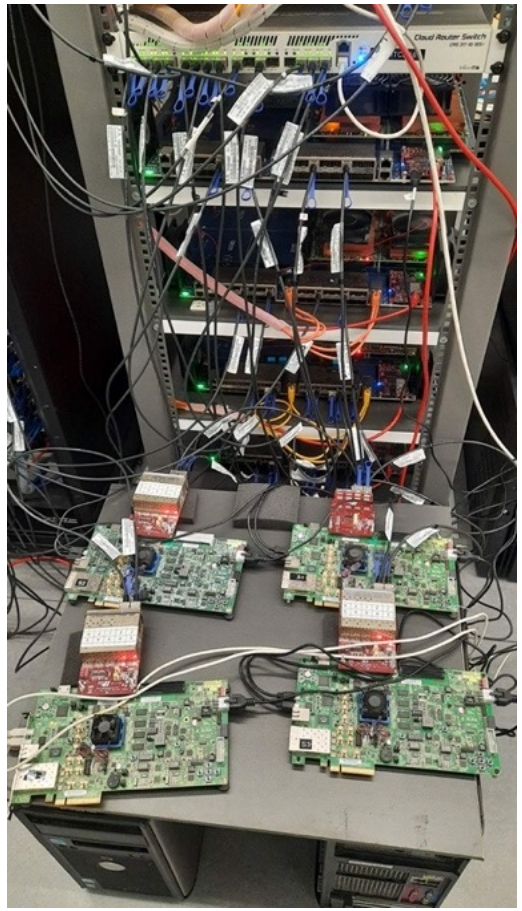


Low network throughput

HW testbed: upgrade of the ExaNeSt platform (FORTH)



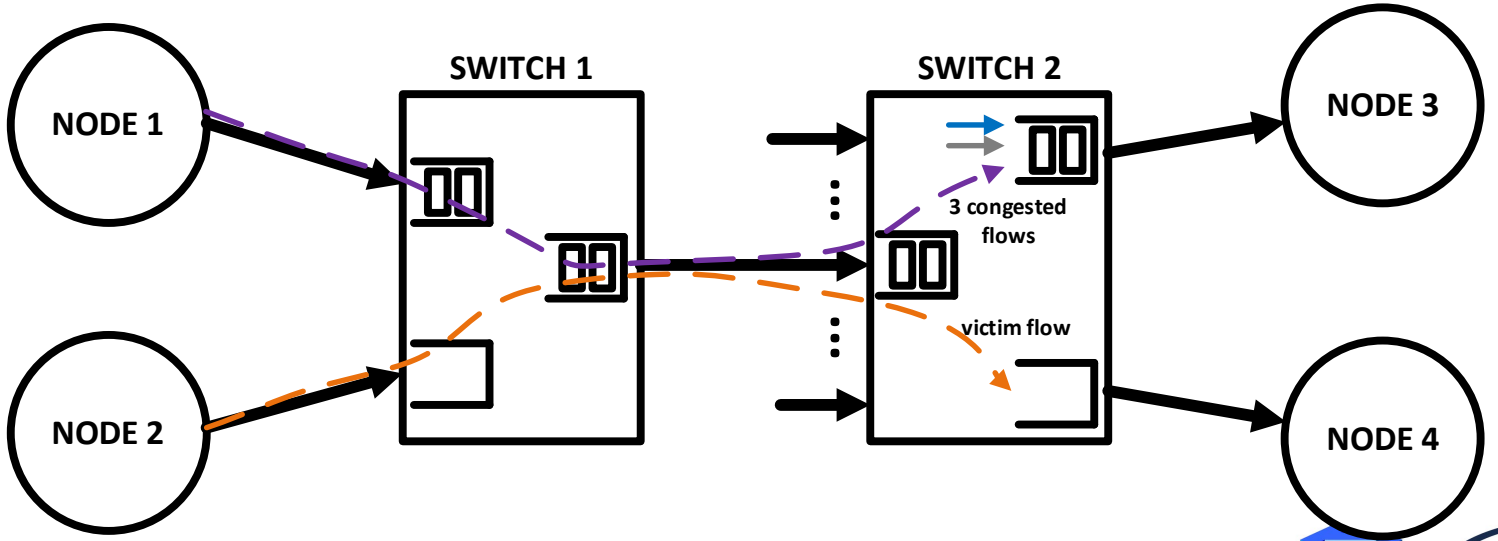
- New topology using the QFDBs:
  - Spine-leaf offering path diversity
  - Deadlock-free routing w/o VCs



- Global clock mechanism
- Clock injector
- Traffic generator
- Flow measurer
- Flow Rate Packet (FRP)
- Multiple Priority Crossbar
- Transceiver FIFOs with Configurable Depth
- caRVnet Interface Drivers

Congestion Management based on injection throttling

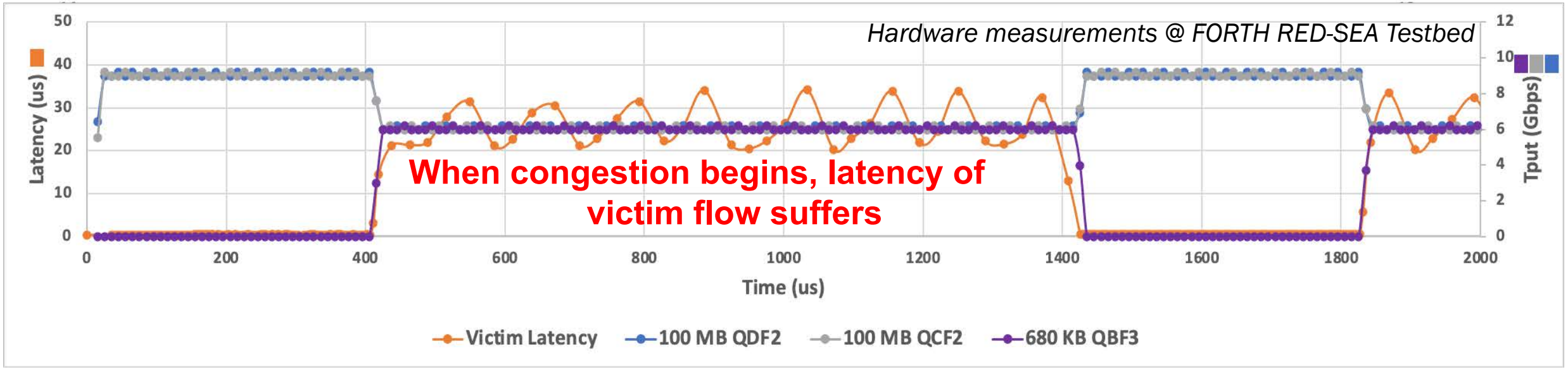
- Evaluating different versions of FORTH congestion management mechanism using datacenter workloads (DAW workloads) running on ExaNest hardware
  - The scheme was evaluated using simulation and, within the RED-SEA project, it is adapted for BXI interconnect, implemented and evaluated in real hardware
- FORTH results show that the mechanism is able to protect the latency of victim flows against the congestion of other flows



Without Hardware Congestion control

Network congestion

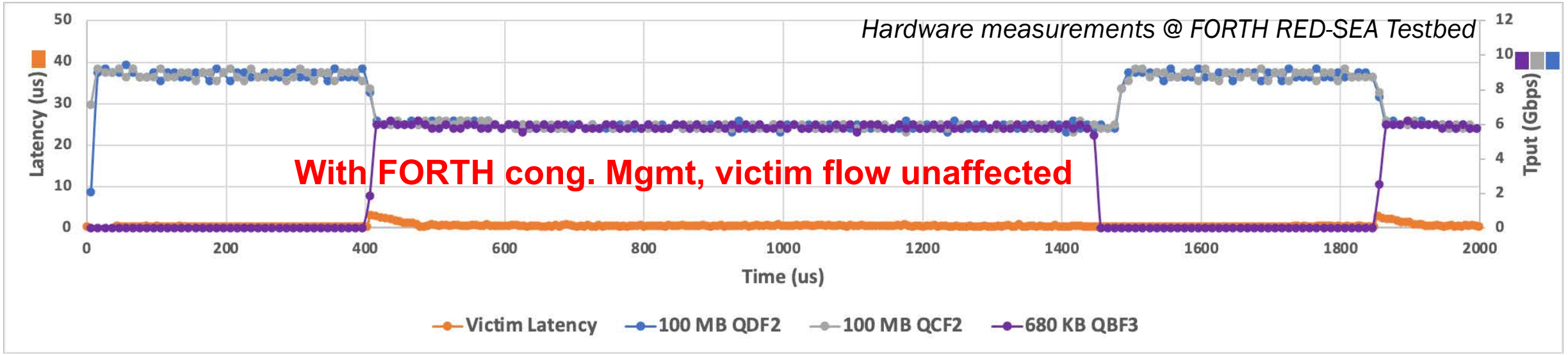
- An old unresolved problem
  - Not even TCP present in RDMA
- HPC systems traditionally overlooked the problem
  - One self throttled app per time



Using Hardware rate throttling at source

Network congestion

- An old unresolved problem
  - Not even TCP present in RDMA
- HPC systems traditionally overlooked the problem
  - One self throttled app per time

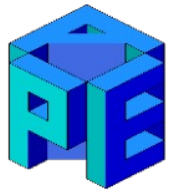


D. Giannopoulos, N. Chrysos, E. Mageiropoulos, G. Vardas, L. Tzanakis, M. Katevenis, “Accurate Congestion Control”, IEEE/ACM NOCS. Torino, Italy, 2018.

RED-SEA at glance

- ☺ Network requirement and architecture defined, execution of applications and benchmarks on testbeds verified
- ☺ Internet Protocol over BXI2 kernel module optimised (up to x4 times), 3 Architectures specifications defined (Ethernet Gateway & MAC/PCS, transport layer), First RTL version of Ethernet MAC & PCS completed
- ☺ Congestion characterization finished; first results of congestion management obtained (latency gains)
- ☺ Preliminary results of BXI2 Link Layer design for FPGA met expectations, New RDMA HW Engine made & small transfers optimised,

☞ **Expecting** to have major impact/contribution to the future EU Interconnect by the end of the project



https://apegate.roma1.infn.it/

@APELab_INFN

Thank you!!!