Ten years of COKA

(Computing On Kepler Architectures)

Enrico Calore, Sebastiano Fabio Schifano

INFN and University of Ferrara, Italy

26/11/2024

Outline

HPC at INFN Ferrara

Long-lasting experience in computational physics, using custom and commercial HPC systems. Several highly parallel HPC systems were designed, implemented and operated to solve specific problems:

- APE (Array Processor Experiment) series of supercomputers, to solve Lattice-QCD simulations;
- Janus to solve spin-system simulations.

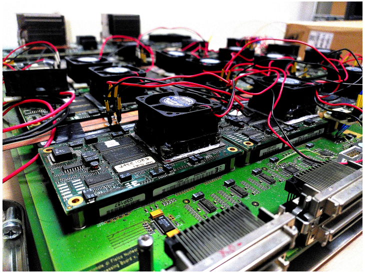


Figure: One board of APEnext

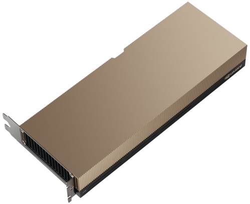
Figure: One board of Janus II

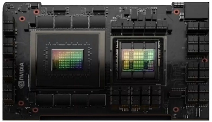
HPC at INFN Ferrara

In the last 15 years, due to **higher costs in producing custom ASIC** processors, we focused on:

- technology tracking;
- benchmarking;
- exploitation to speed-up scientific simulations;

of **off-the-shelf hardware**, spanning from Arm many-core CPUs, to GPUs and FPGAs as compute accelerators.





The COKA Cluster

The COKA Cluster Project started in 2014, thanks to a grant of the University of Ferrara for an on-premises HPC system; co-funded also by INFN Ferrara.

Cluster Architecture

GPU dense HPC cluster; front-end hosting the storage and 5 compute nodes:

```
                Network
                   |
            _______|______
            |    COKA    |                 _____
            | Login Node |________________/     \
            ==============                | HDD |
     ______________|______________        \_____/
    /       /      |      \       \
   /       /       |       \       \
  |        |       |        |       |
Node01   Node02  Node03   Node04  Node05
```

Diskless compute nodes booting CentOS7 using PXE/TFTP from the front-end.

Cluster Architecture

Compute nodes mount / filesystem from the master node, where a different copy for each node is saved:

`/srv/node01/ /srv/node02/ /srv/node03/ /srv/node04/ /srv/node05/`

Common folders for all nodes:

/bin, /home, /homeraid, /lib, /lib64, /opt, /root, /sbin, /scratch and /usr

Private folders for each node:

/, /boot, /dev, /etc, /mnt, /proc, /run, /sys and /var

Upgrading packages in the master node may lead to inconsistencies for all the packages that write/modifies files in one of theses folders, such as /etc.

Need to manually check and repair:

```
diff --brief -r /etc/ /srv/node01/etc/ | less

diff -r /etc/ /srv/node01/etc/ | less
```

Cluster Architecture

Each node is equipped with:

No.	Device	Model	Architecture
2×	CPU	Intel Xeon E5 2630	(Haswell)
8×	2xGPU	NVIDIA Tesla K80	(Kepler)
2×	IB NIC	Mellanox ConnectX-3	(56Gb/s FDR)

It reaches a double-precision computing performance of $\approx$ 100 TFLOPs, making it the most powerful cluster installed in the premises of an Italian University!

...at the time.

Cluster Architecture

Each node is equipped with:

No.	Device	Model	Architecture
2×	CPU	Intel Xeon E5 2630	(Haswell)
8×	2xGPU	NVIDIA Tesla K80	(Kepler)
2×	IB NIC	Mellanox ConnectX-3	(56Gb/s FDR)

It reaches a double-precision computing performance of $\approx$ 100 TFLOPs, making it the most powerful cluster installed in the premises of an Italian University!

...at the time.

Cluster Architecture

Figure: The inside of a COKA node.

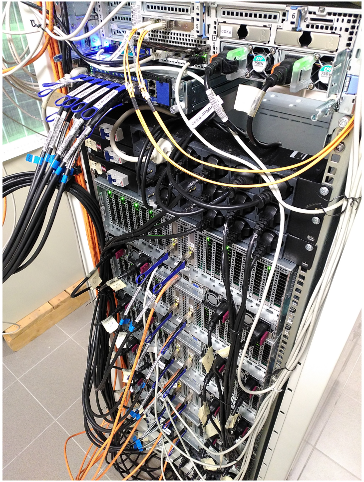


Figure: Rear view of COKA Cluster.

COKA Node Architecture

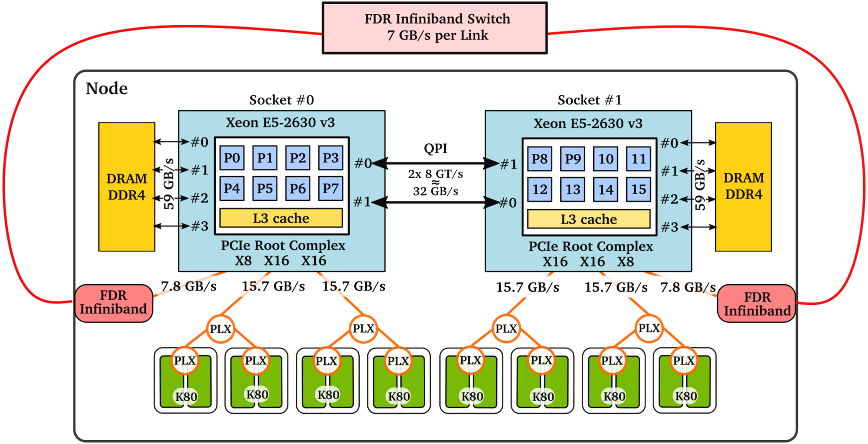


Figure: Schematic representation of one COKA node, highlighting its NUMA nature and the bandwidth of each communication channel.

COKA Node Architecture

The 16 cores available on each node are divided across two different sockets, connected respectively to different memory sub-systems.

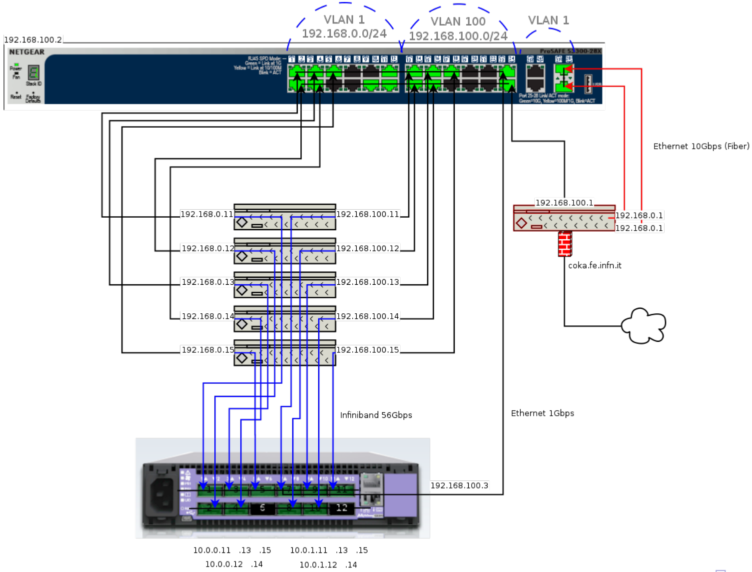
Although cache coherency is granted, from the performance pint of view you should take care of where your processes and threads are being executed!

COKA Node Architecture

```
#!/bin/bash
#SBATCH --job-name=gpu-test
#SBATCH --error=gpu-test-%j.err
#SBATCH --output=gpu-test-%j.out
#SBATCH --ntasks=2
#SBATCH --ntasks-per-node=2
#SBATCH --cpus-per-task=8
#SBATCH --partition=veryshortrun

module load cuda
module load openmpi

# We want one thread per core and thus 8 threads
# (as many threads as --cpus-per-task):
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

srun --cpu_bind=v,sockets --mem_bind=v,local ./my_program
```

Job script requesting resources for 2 MPI processes and 8 cores per process. Each process is bound to a socket and forced to allocate memory buffers on local memory.

COKA Networks Architecture



COKA IB Network Peculiarity

All NUMA nodes can communicate with each others with similar bandwidths and latencies while being on the same, or different, compute nodes.
Faster intra-node communications have even been observed, between NUMA nodes, using the Infiniband network, instead of the QPI inter-socket link:

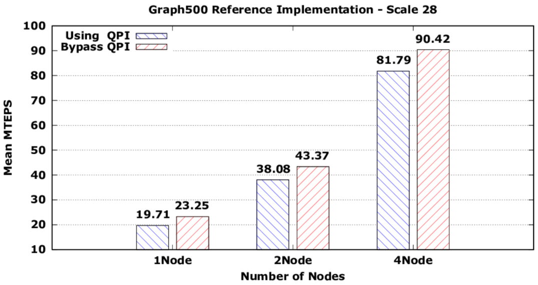


Figure: Performance of the Graph 500 benchmark, using QPI and bypassing it using the Infiniband network.

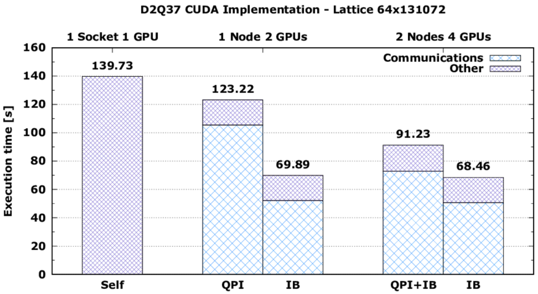
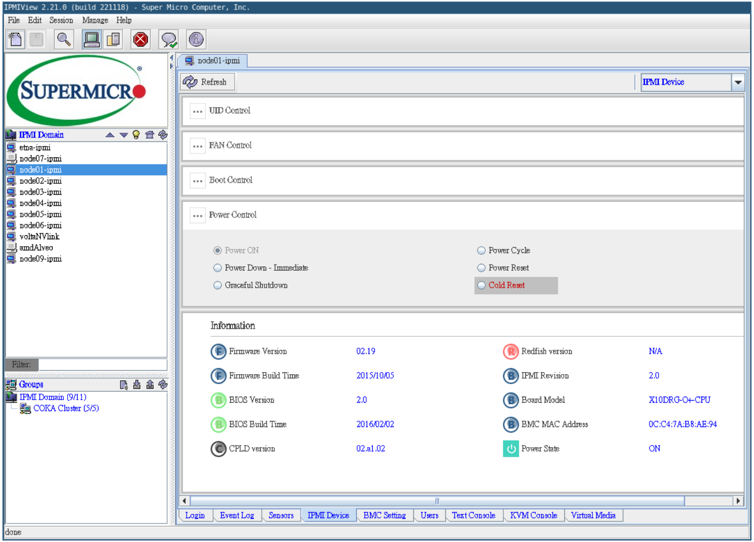


Figure: Performance of a Lattice-Boltzmann simulation on the GPUs, using QPI or IB for intra-node communications.

COKA IPMI Network



COKA Cluster Usage

Since the commissioning of COKA, the SLURM scheduler accounted for:

179 users, more than **2.1M of Jobs** completed; **1.99M CPU Core/h** and **0.72M GPU/h**.

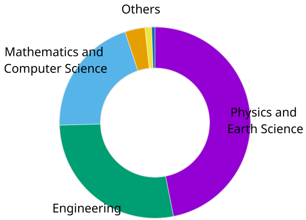


Figure: Distribution of users.

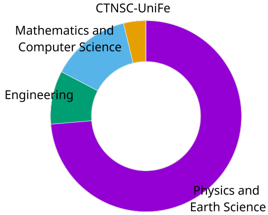


Figure: CPU Core/h usage.

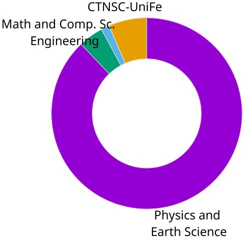


Figure: GPU/h usage.

Applications

COKA has been extensively used for lattice-based simulations and more recently also for AI workloads:

- Lattice-Boltzmann Models (LBM) Simulations.
- Lattice Quantum Chromodinamics (LQCD) Simulations.
- Experimental Data Processing and Analysis.

Additional heterogeneous compute nodes were also attached to the COKA infrastructure to benchmark novel architectures, such as Arm CPUs; AMD CPUs, GPUs, and FPGAs.

Lattice-Boltzmann Models (LBM) Simulations.

LBM is a class of CFD methods. Instead of solving the Navier–Stokes equations directly, a fluid density is simulated on a lattice, with streaming and collision (relaxation) processes.

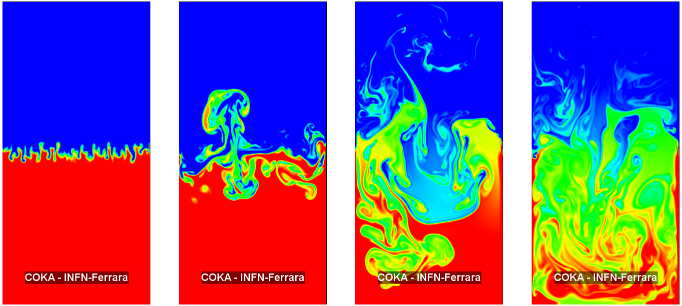


Figure: Simulation of the Rayleigh-Taylor (RT) Instability using the D2Q37 LBM model. A cold-dense fluid over a less dense and warmer fluid triggers an instability that mixes the two fluid-regions (till equilibrium is reached).

D2Q37 implementation included in **SPEChpc 2021** Benchmark Suites

Lattice-Boltzmann Models (LBM) Simulations.

LBM is a class of CFD methods. Instead of solving the Navier–Stokes equations directly, a fluid density is simulated on a lattice, with streaming and collision (relaxation) processes.

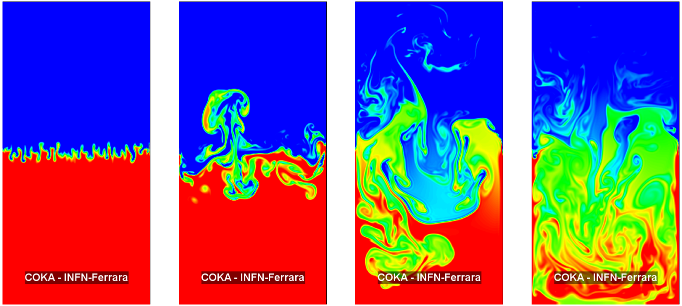


Figure: Simulation of the Rayleigh-Taylor (RT) Instability using the D2Q37 LBM model. A cold-dense fluid over a less dense and warmer fluid triggers an instability that mixes the two fluid-regions (till equilibrium is reached).

D2Q37 implementation included in **SPEChpc 2021** Benchmark Suites

Artificial Intelligence Workloads.

In the last years AI related workloads increased significantly, in particular in the context of experimental data analysis and medical imaging.

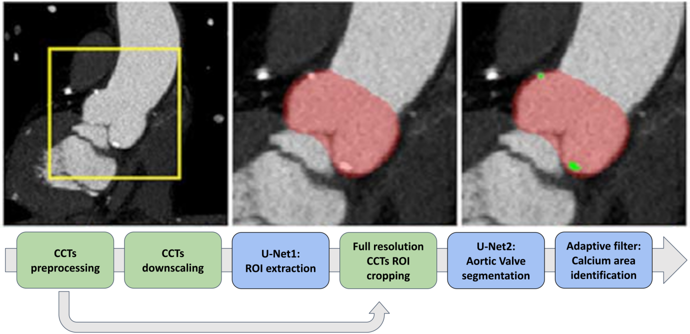


Figure: Medical images segmentation using Deep Neural Networks (DNN) acceleration. Schematic pipeline for the identification of calcification regions inside the aortic valve from CCTs scans using a double U-Net adaptive filter approach. Boxes in green identify the CCTs processing steps, while blue ones the U-Nets and filter phases.

Technology tracking of hardware accelerators.

Additional heterogeneous nodes have been added to the original COKA Cluster in order to benchmark novel accelerators, such as FPGAs.

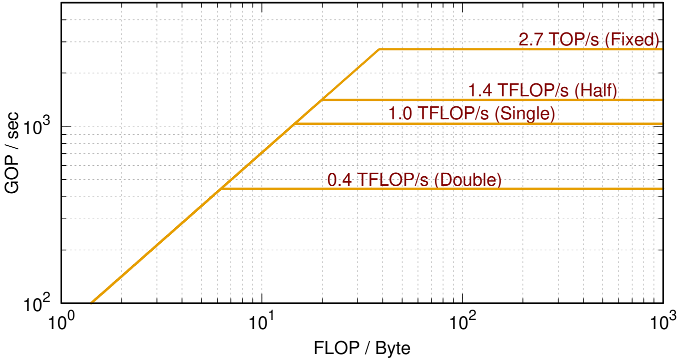


Figure: Roofline model of an Alveo U250 accelerator for different numerical precisions of the FMA operator.

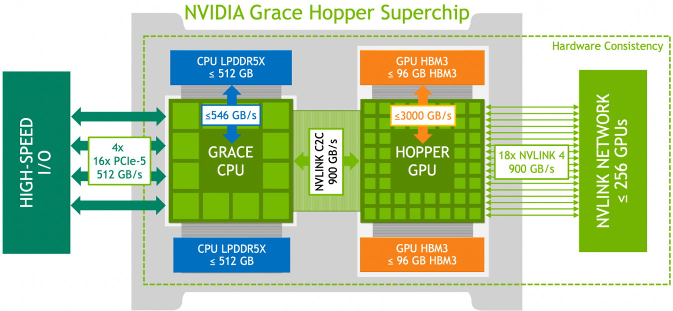
What next?

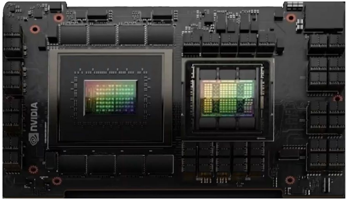
The COKA Cluster is being completely renovated:

- a new front-end;
- new compute nodes with NVIDIA Grace Hopper Superchip (i.e., Arm CPU + Hopper GPU)

Figure: First prototype node acquired and tested.

What next?


NVIDIA Grace Hopper Superchip
Hardware Consistency
HIGH-SPEED I/O
4x 16x PCIe-5 512 GB/s
CPU LPDDR5X ≤ 512 GB
≤546 GB/s
GRACE CPU
NVLINK C2C 900 GB/s
GPU HBM3 ≤ 96 GB HBM3
≤3000 GB/s
HOPPER GPU
18x NVLINK 4 900 GB/s
NVLINK NETWORK ≤ 256 GPUs
CPU LPDDR5X ≤ 512 GB
GPU HBM3 ≤ 96 GB HBM3



What next?

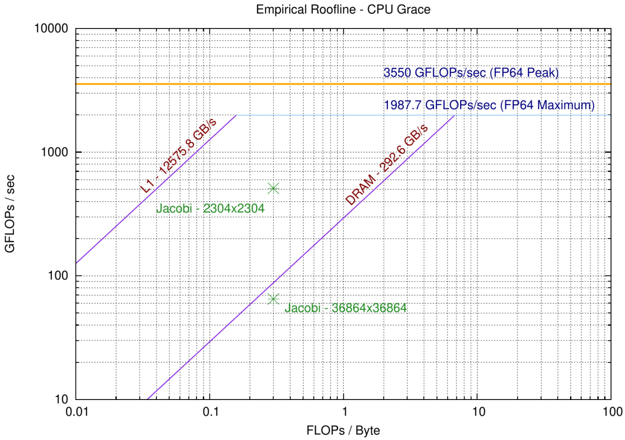


Figure: Theoretical, empirical-synthetic and -miniApp, performance of Grace CPU in the framework of the Roofline model. Credits: Giorgia Gammone, BSc Thesis.

What next?

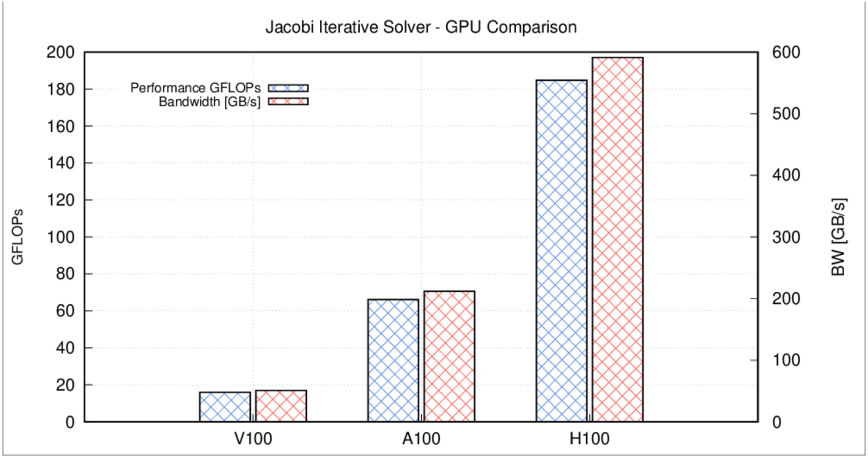


Figure: Comparison of measured bandwidth and compute performance of the Hopper GPU, with previous architectures. Credits: Giorgia Gammone, BSc Thesis.

What next?

What to change?

- operating system $\rightarrow$ Ubuntu;
- compute node installation $\rightarrow$ local;
- services $\rightarrow$ containerized / virtualized;

What to additionally support?

- containers launch through SLURM;
- use of Jupyter notebooks through SLURM;
- cross-compiling on front-end;

What next?

What to change?

- operating system $\rightarrow$ Ubuntu;
- compute node installation $\rightarrow$ local;
- services $\rightarrow$ containerized / virtualized;

What to additionally support?

- containers launch through SLURM;
- use of Jupyter notebooks through SLURM;
- cross-compiling on front-end;

References

S.F. Schifano, et al., High throughput edit distance computation on FPGA-based accelerators using HLS, in Future Generation Computer Systems, vol. 164, 2025, doi: *10.1016/j.future.2024.107591*.

G. Minghini et al., *An HPC Pipeline for Calcium Quantification of Aortic Root From Contrast-Enhanced CCT Scans*, in IEEE Access, vol. 11, pp. 101309–101319, 2023, doi: *10.1109/ACCESS.2023.3315734*.

E. Calore et al., *FER: A Benchmark for the Roofline Analysis of FPGA Based HPC Accelerators*, in IEEE Access, vol. 10, pp. 94220–94234 (2022), doi: *10.1109/ACCESS.2022.3203566*.

E. Calore et al., *ThunderX2 Performance and Energy-Efficiency for HPC Workloads*, in Computation, vol. 8(1):20, doi: *10.3390/computation8010020*.

E. Calore et al., *Optimization of lattice Boltzmann simulations on heterogeneous computers*, International Journal of High Performance Computing Applications, 33(1), pp. 124–139, 2019, doi: *10.1177/1094342017703771*

Bonati, et al., *Portable multi-node LQCD Monte Carlo simulations using OpenACC*, International Journal of Modern Physics C, 29(1), 2018, doi: *10.1142/S0129183118500109*.

E. Calore, et al., *Massively parallel lattice-Boltzmann codes on large GPU clusters*, Parallel Computing, vol. 58, pp. 1–24, 2016, doi: *10.1016/j.parco.2016.08.005*

E. Calore et al., *Performance and portability of accelerated lattice Boltzmann applications with OpenACC*, Concurrency and Computation: Practice and Experience, 28(12), pp. 3485–3502, 2016, doi: *10.1002/cpe.3862*

Thanks for Your Attention

FERRARIAE UNIVERSITAS · EX LABORE FRUCTUS · 13 91

ICSC
Centro Nazionale di Ricerca in HPC,
Big Data and Quantum Computing