

Computing Environment

NIC5 cluster hardware overview

Orian Louant

ULiège - INFO0939 – Fall 2023

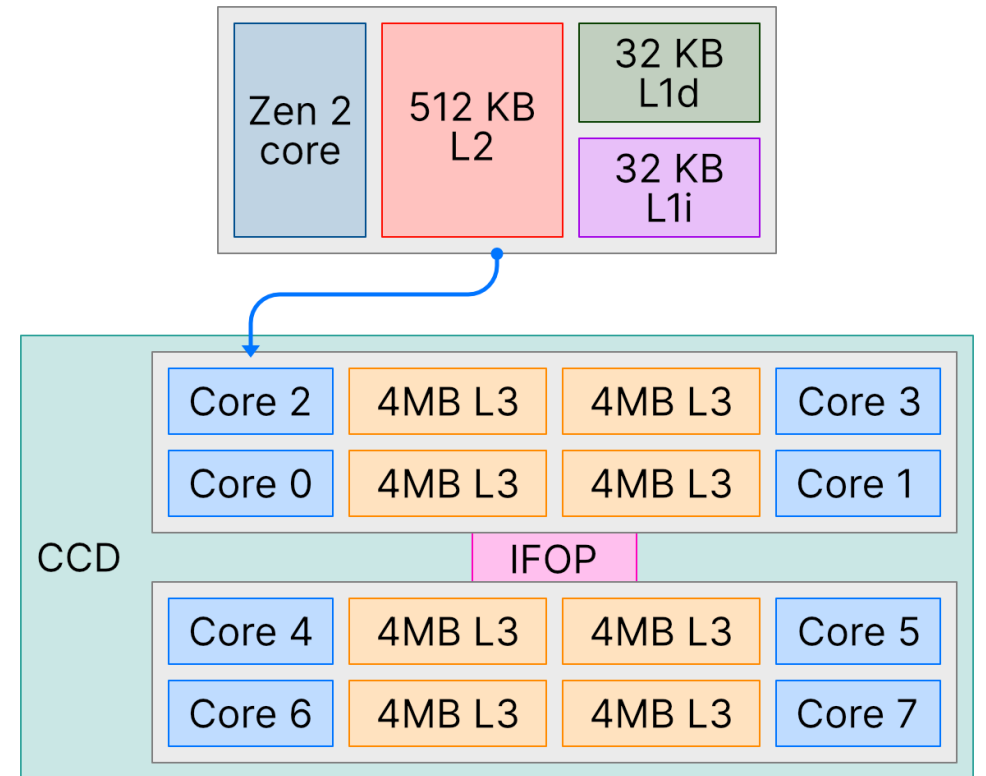
NIC5: Core and Core Complex Die

All NIC5 CPUs are **32 cores AMD EPYC 7542** using a multi-chip-module (MCM): multiple dies combined on one package:

- one more or more Core Complex Dies (CCDs) which contains the cores
- an I/O die which is a grid that connects the CCDs among each other and to external components

A CCD contains eight cores, grouped into **two Core Complexes** composed of **4 cores** sharing 16 MB of L3 cache (4 x 4 MB slices). The cores are equipped with

- 32 KB of L1 data cache (L1d)
- 32 KB of L1 instruction cache (L1i)
- 512 KB of L2 data cache



NIC5: The CPUs

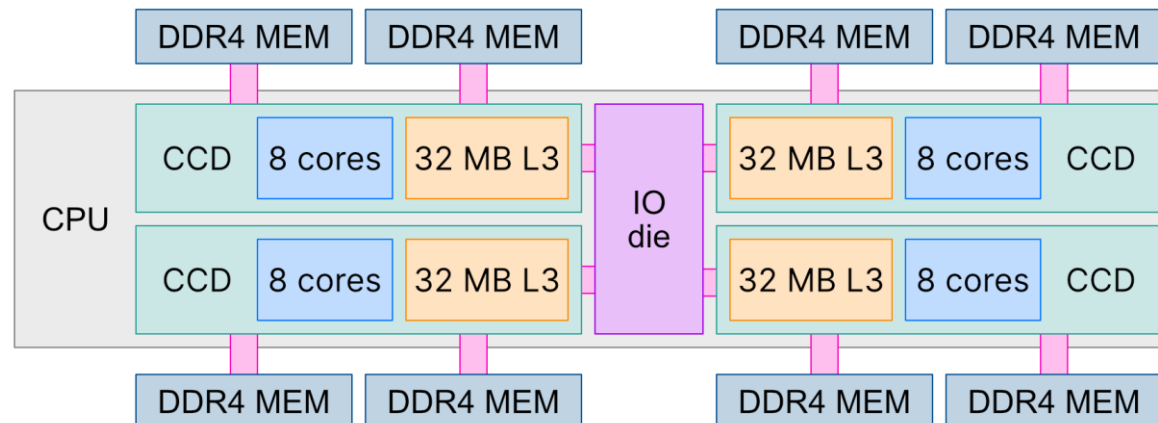
- The AMD EPYC 7542 processor has 4 CCDs forming a processor with 32 core (4 x 8 cores CCD)
- An Infinity Fabric on Package (IFOP) interface link the CCD to the I/O die
- Each CCD has two memory controllers: the complete CPU has eight memory controller (channels)

Memory bandwidth:

$$3200 \text{ MT/s} \times 8 \text{ bytes/transfer} \times 8 \text{ channels} = 204.8 \text{ GB/s}$$

Floating-point performance:

$$2 \text{ units} \times 4 \text{ Flops (AVX)} \times 2 \text{ Flops (FMA)} \times 2.9 \text{ GHz} \times 32 \text{ cores} = 1.485 \text{ TFlops}$$



NIC5: Compute node

The compute nodes of NIC5 feature two sockets:

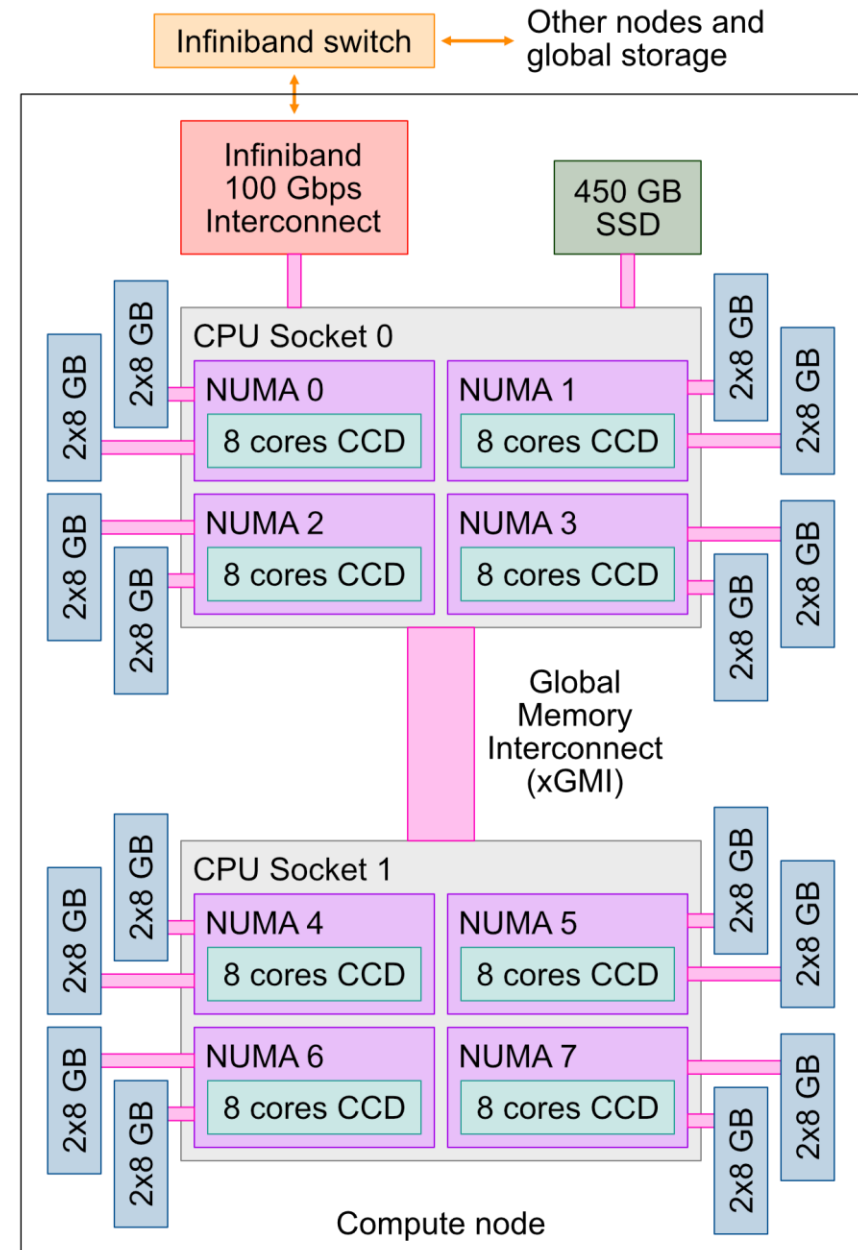
- each socket equipped with an AMD EPYC 7542
- a total of 64 cores per compute node (2 x 32 cores)

Sockets are interconnected by a specialized link known as Global Memory Interconnect (xGMI)

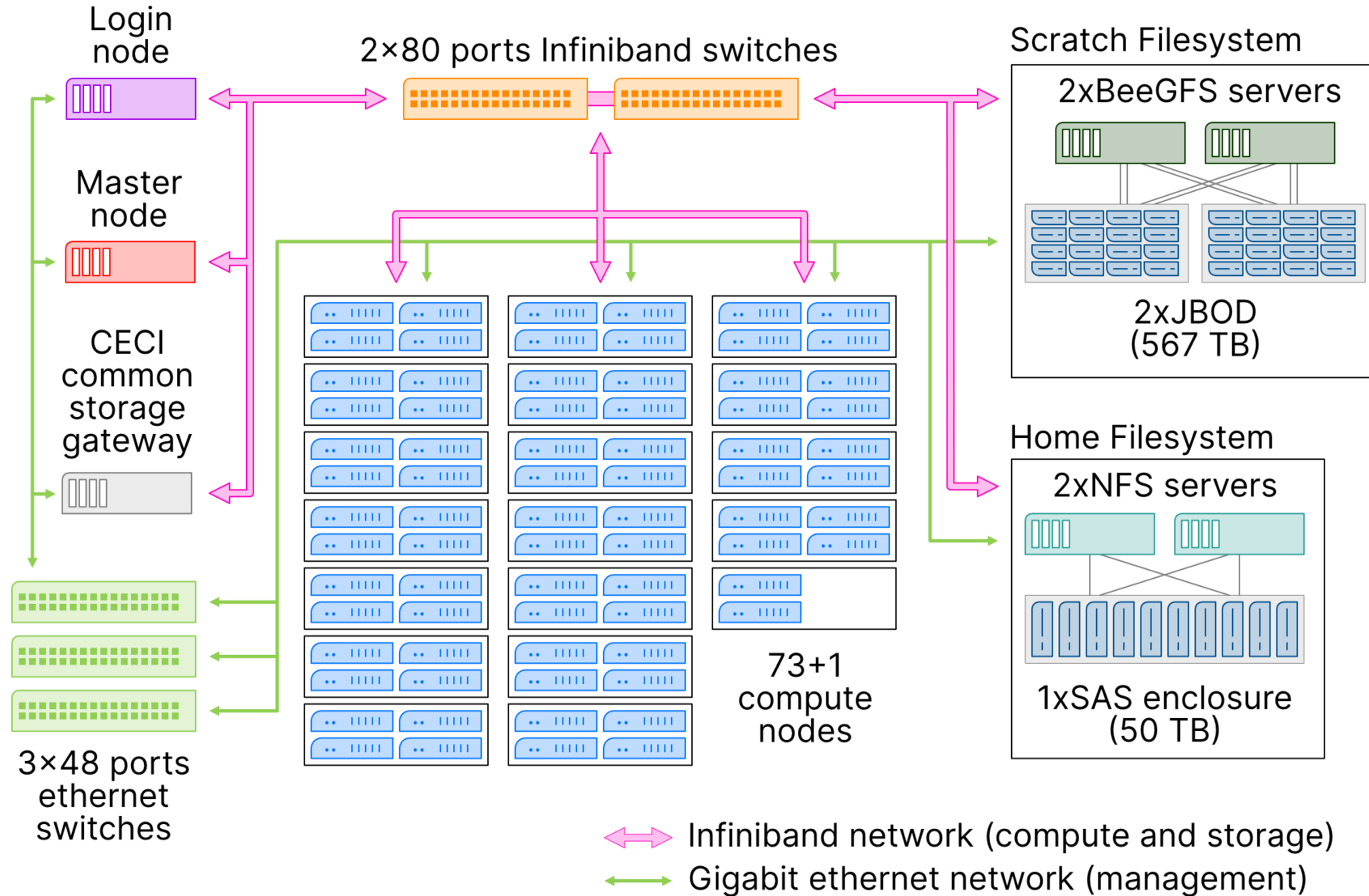
As each CCD is directly linked to two memory controllers, the CPU sockets are partitioned into four Non-Uniform Access Nodes

The compute nodes are also equipped with

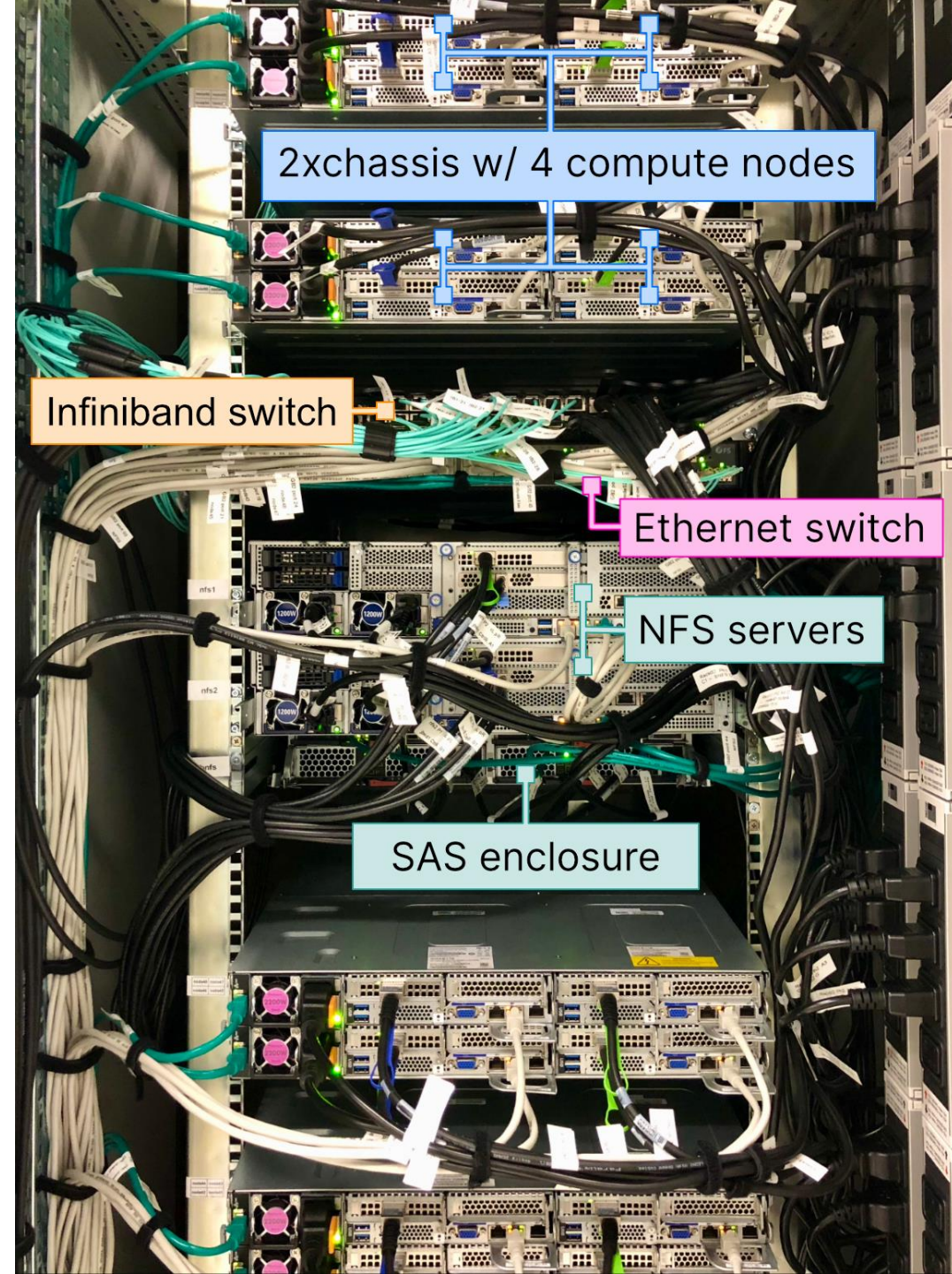
- a local SSD storage capacity of 450 GiB
- an InfiniBand HDR100 interconnect, providing a bandwidth of 100 Gbps (12.5 GB/s)



NIC5: Overview of the entire cluster



Behind the scene



2x chassis w/ 4 compute nodes

Infiniband switch

Ethernet switch

NFS servers

SAS enclosure