CERN Openlab

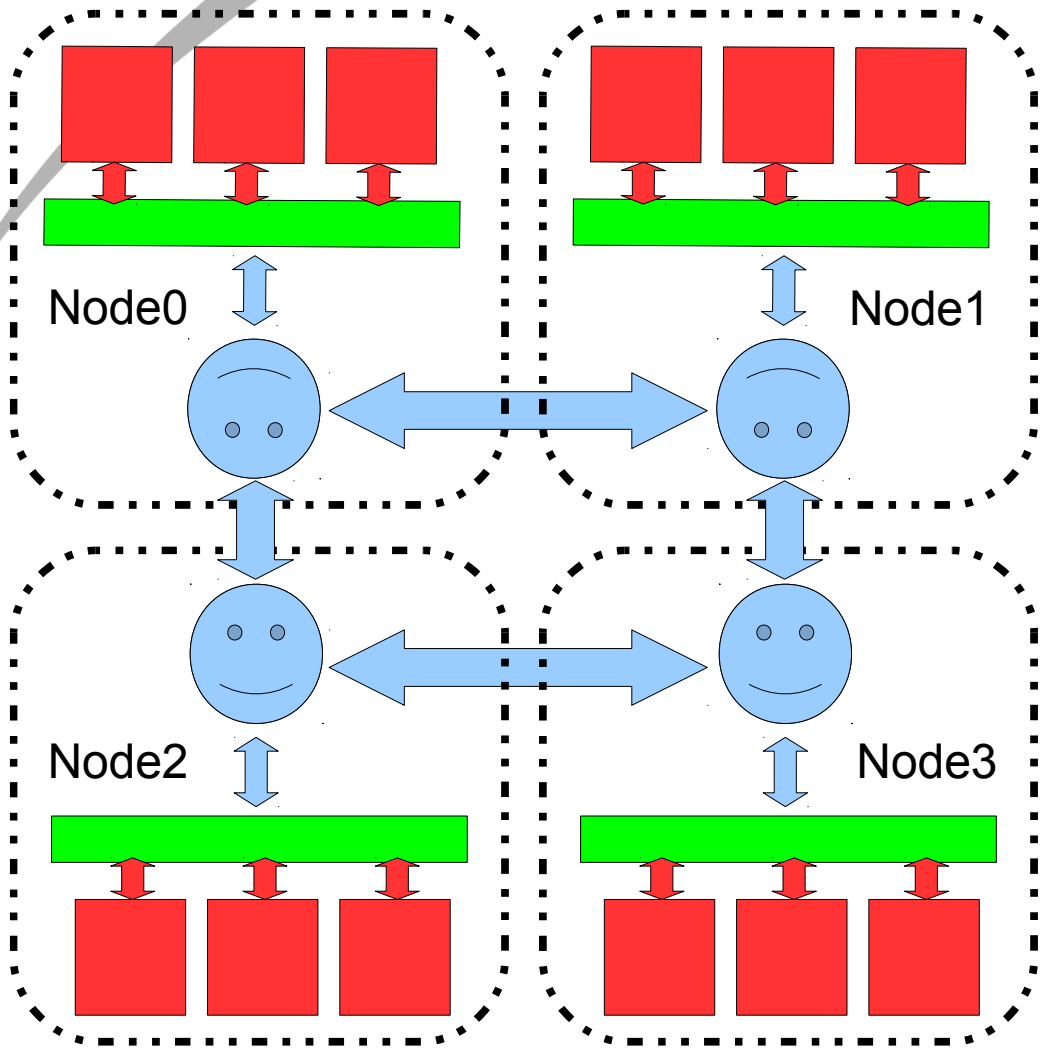
NUMA experience on large many-core servers from AMD/Intel

Julien Leduc
CERN openlab
julien.leduc at cern.ch

Current servers architectures

- Enterprise servers divide in 3 categories:
  - SMP (Symmetric Multi Processing)
    - Common architecture multiple processors connected symmetrically on the memory system
  - MPP (Massive Parallel Processing)
    - Non sharing architecture dividing the system into several nodes that can access local resources often connected with proprietary interconnect
  - NUMA (Non Uniform Memory Access)
    - The full system divides into multiple nodes which can access both local and remote memory.

NUMA architecture

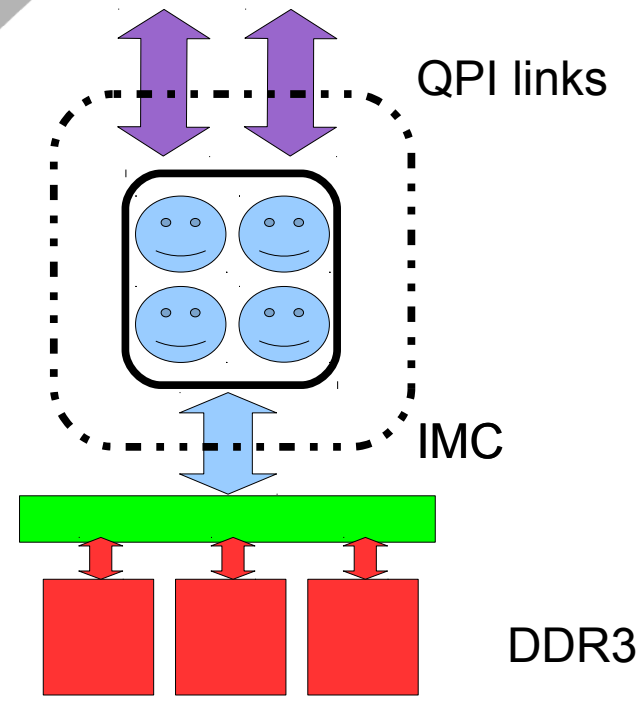


NUMA architecture divides in multiple nodes with access to local and remote memory, at a cost for remote memory.

Why NUMA now?

- Increasing performance now means more and more cores
- Both CPUs and memory don't see a boost in frequency anymore
- NUMA nicely solves these issues, offering a scalable design for future many-cores systems

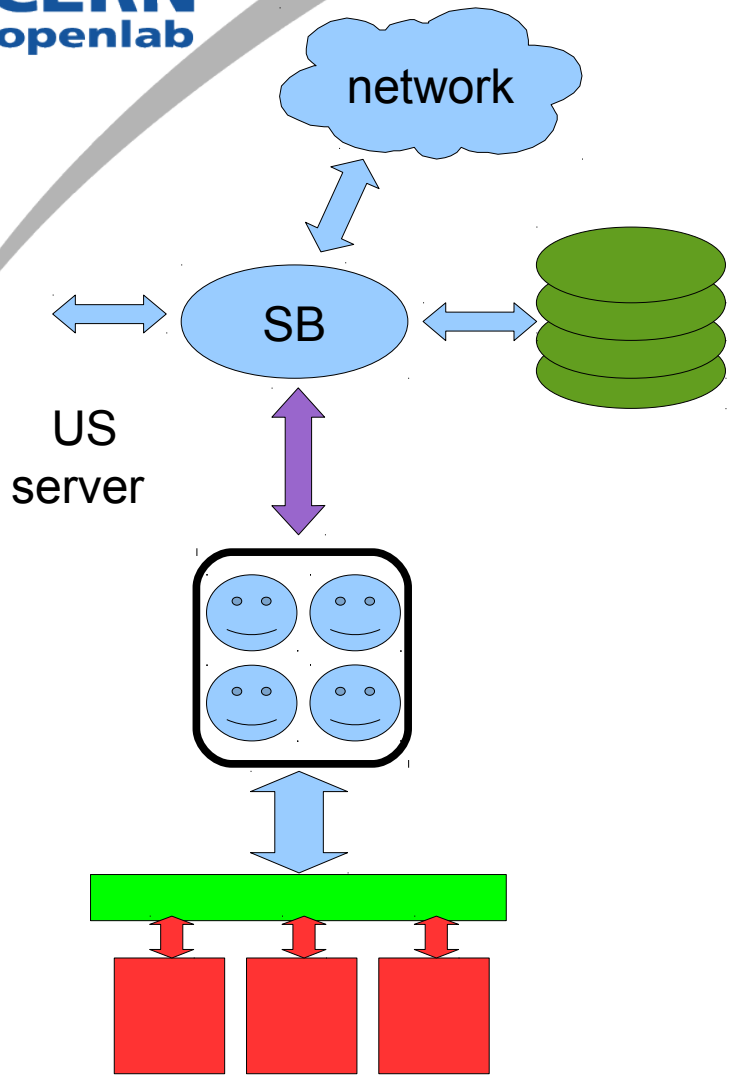
Nehalem microarchitecture

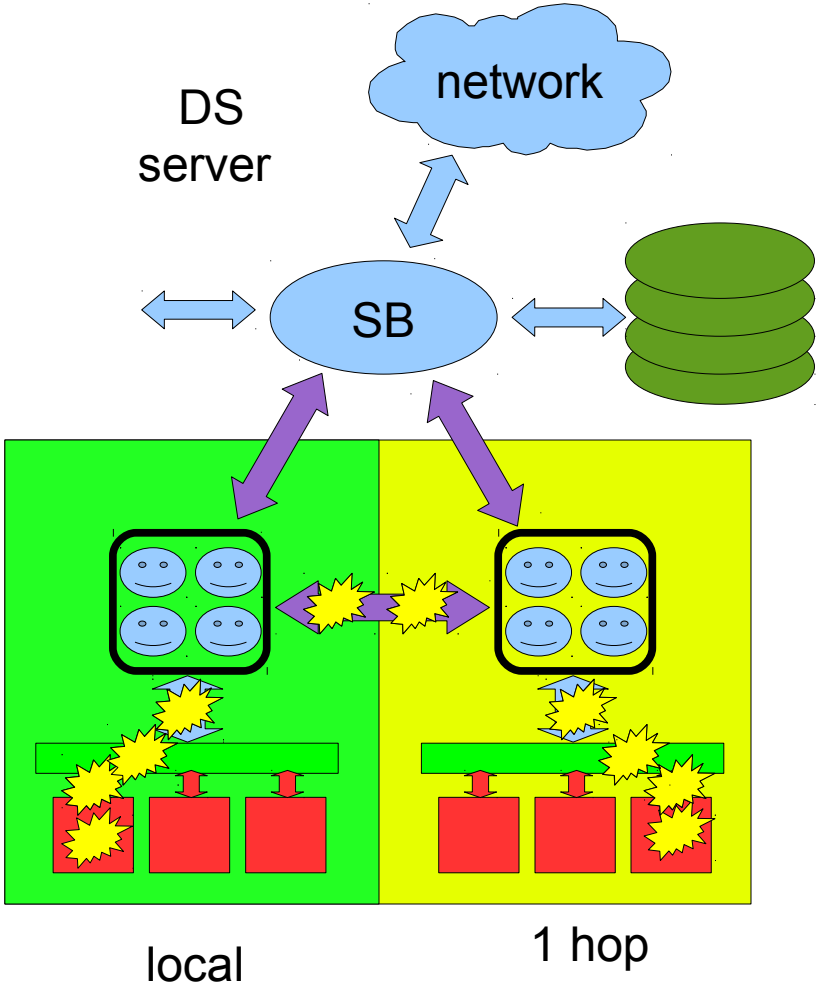


Dual Socket capable NHM

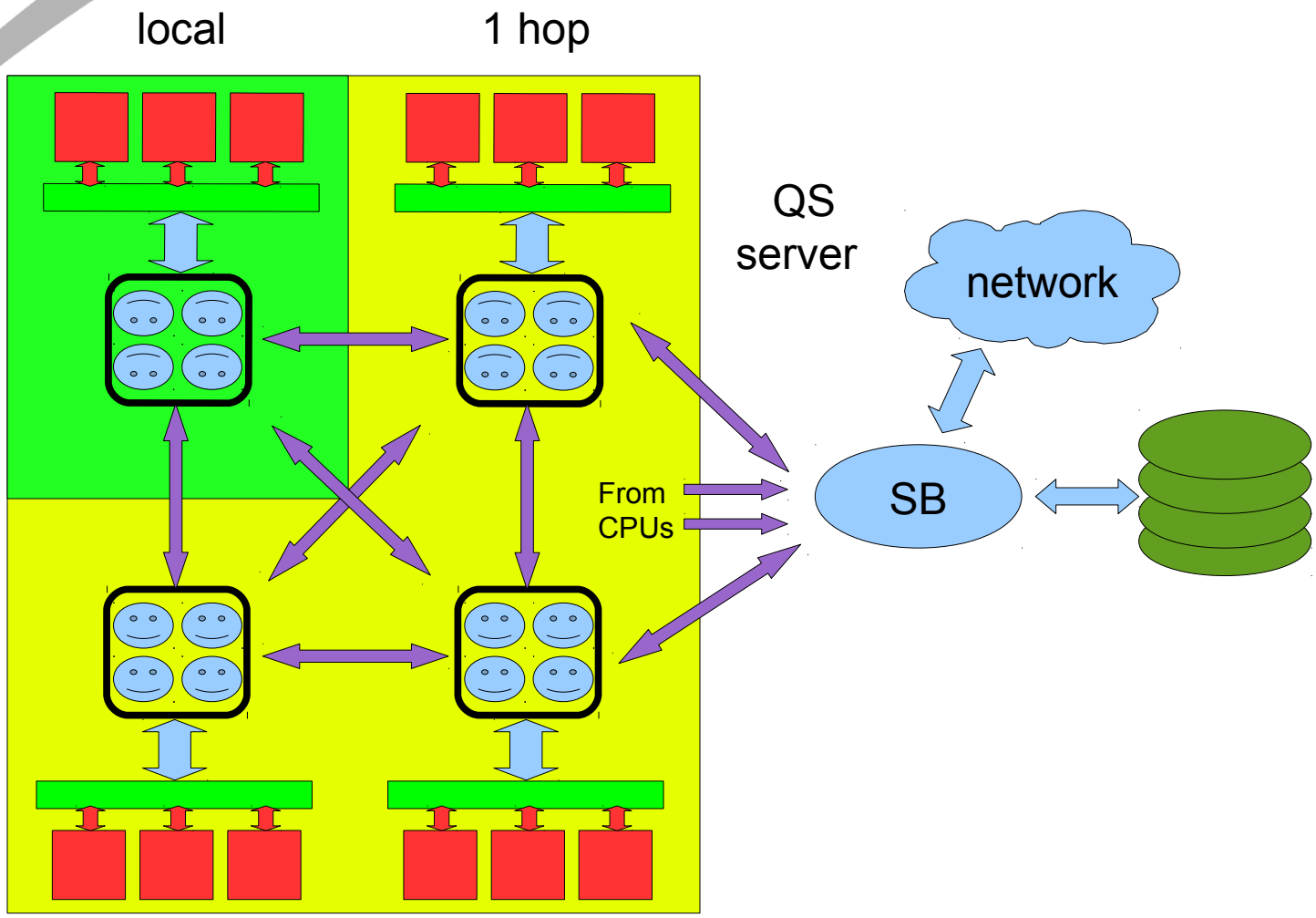
- Nehalem microarchitecture is equipped with an Integrated Memory Controller, and some QuickPath Interconnect links (1 for workstations, 2 for DS servers, 4 for MS servers)
- Microarchitecture allows scalable design for servers

Nehalem microarchitecture
Single and Dual Sockets designs



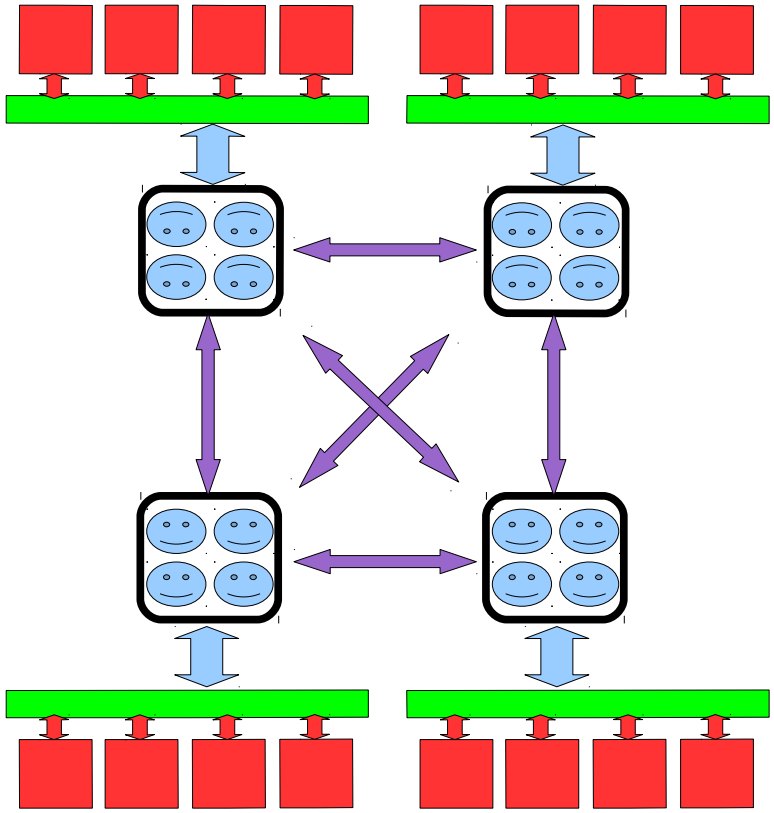


Nehalem microarchitecture Quad Sockets design



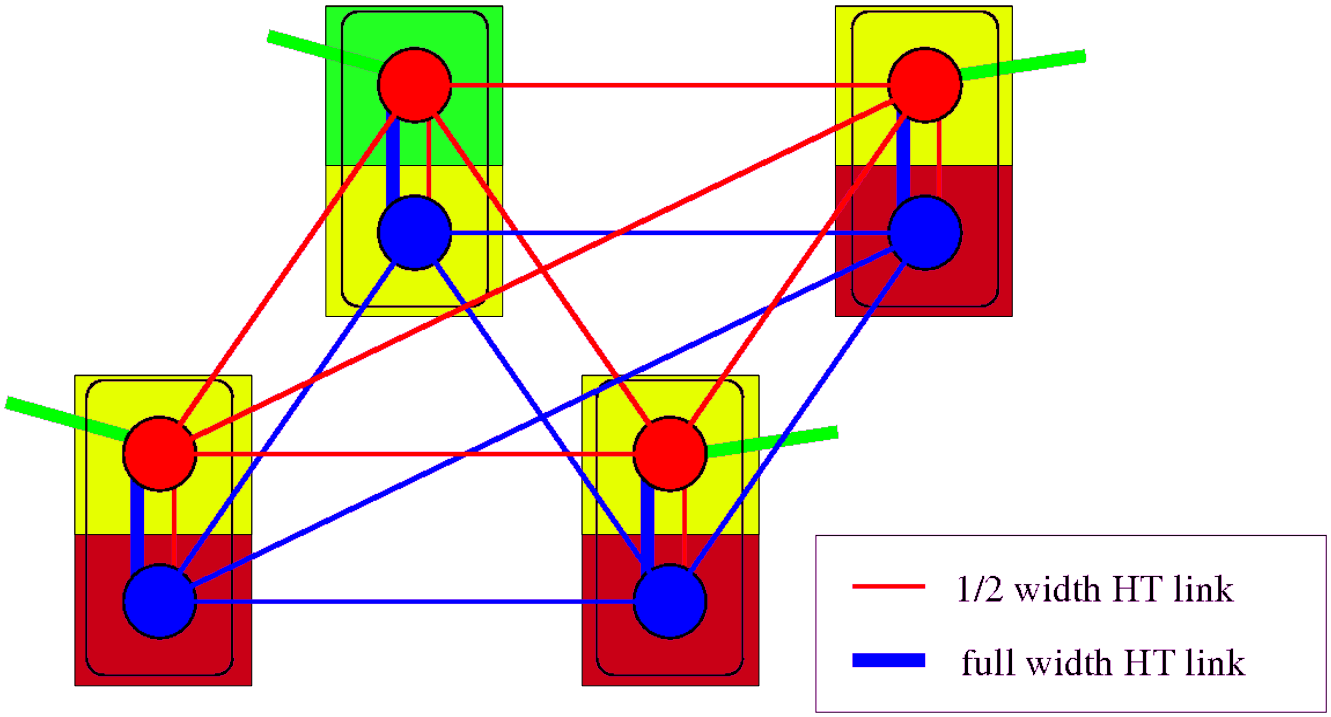
Magny-Cours Quad Socket design

- Each 12-core Magny-Cours is connected to its 3 other neighbors



Magny-cours Quad Socket design

- But each Magny-Cours is composed of a pair of 6-core Istanbul CPUs



NUMA considerations

- NUMA factor:
  - (remote latency)/(local latency)
  - for Westmere 140ns/90ns ~ 1.5
- Linux decomposes a Numa server into nodes:
  - A node is a set of CPUs and its associated memory given by ACPI tables
  - DP system QP system

available: 2 nodes (0-1)
node 0 size: 12279 MB
node 1 size: 12288 MB
node distances:

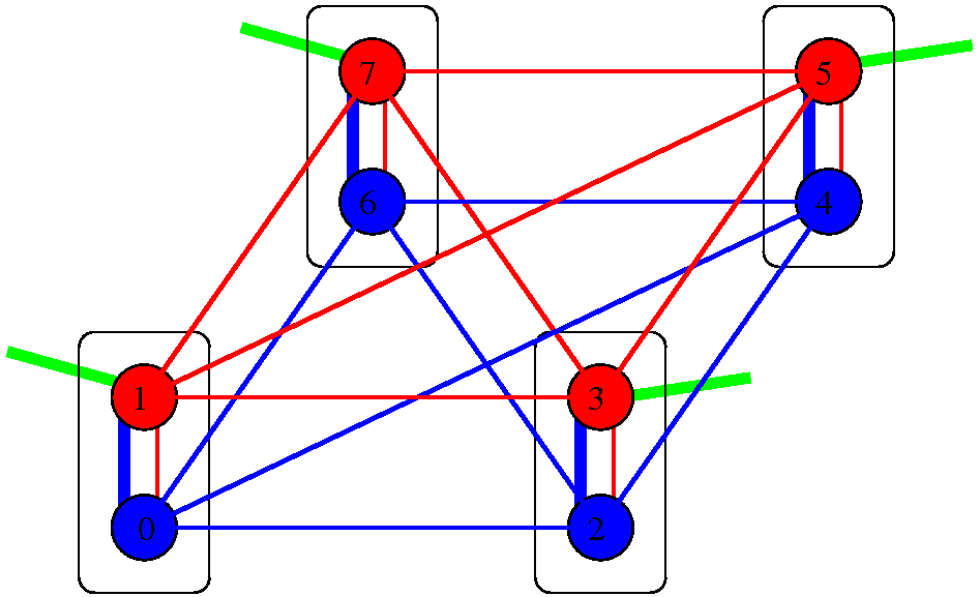
node	0	1
0:	10	20
1:	20	10

available: 4 nodes (0-3)
node 0 size: 32209 MB
node 1 size: 32320 MB
node 2 size: 32320 MB
node 3 size: 32320 MB
node distances:

node	0	1	2	3
0:	10	21	21	21
1:	21	10	21	21
2:	21	21	10	21
3:	21	21	21	10

NUMA considerations

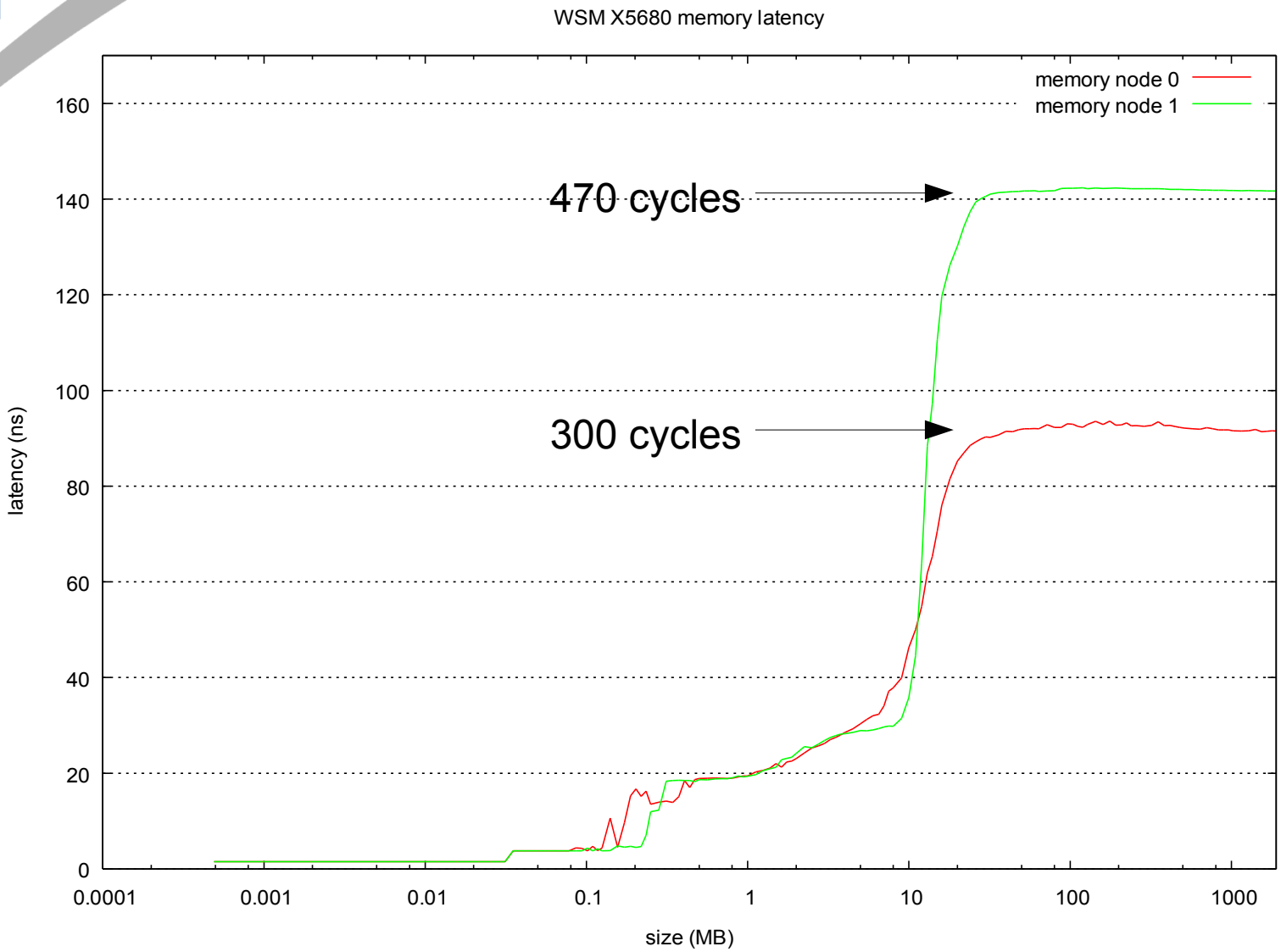
- QP Magny-Cours



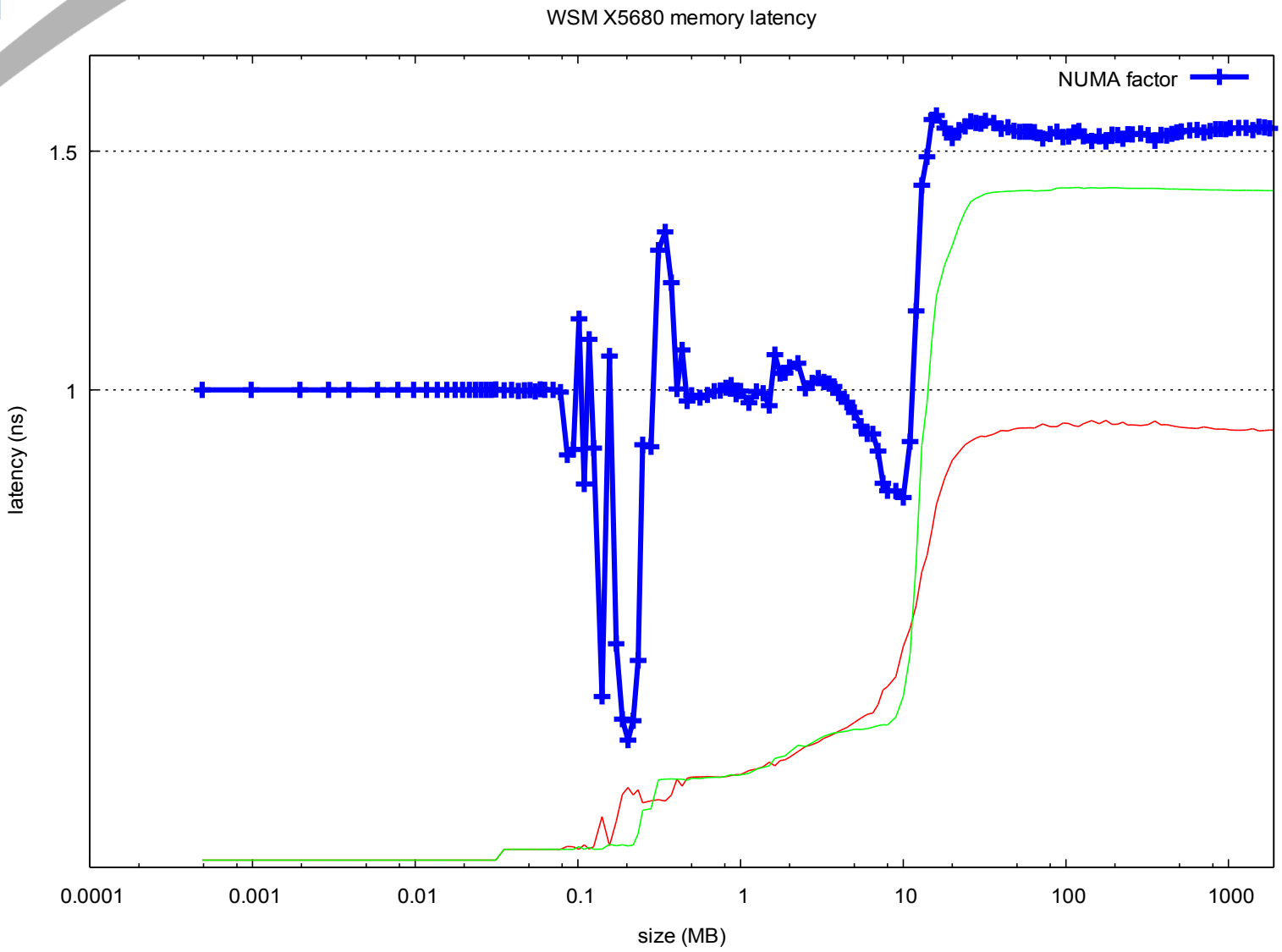
available: 8 nodes (0-7)
node 0 size: 16143 MB
node 0 free: 15261 MB
node 1 size: 8080 MB
node 1 free: 7854 MB
node 2 size: 16160 MB
node 2 free: 16124 MB
node 3 size: 8080 MB
node 3 free: 6463 MB
node 4 size: 16160 MB
node 4 free: 16111 MB
node 5 size: 8080 MB
node 5 free: 8045 MB
node 6 size: 16160 MB
node 6 free: 16092 MB
node 7 size: 7448 MB
node 7 free: 3833 MB
node distances:

node	0	1	2	3	4	5	6	7
0:	10	16	16	22	16	22	16	22
1:	16	10	22	16	22	16	22	16
2:	16	22	10	16	16	22	16	22
3:	22	16	16	10	22	16	22	16
4:	16	22	16	22	10	16	16	22
5:	22	16	22	16	16	10	22	16
6:	16	22	16	22	16	22	10	16
7:	22	16	22	16	22	16	16	10

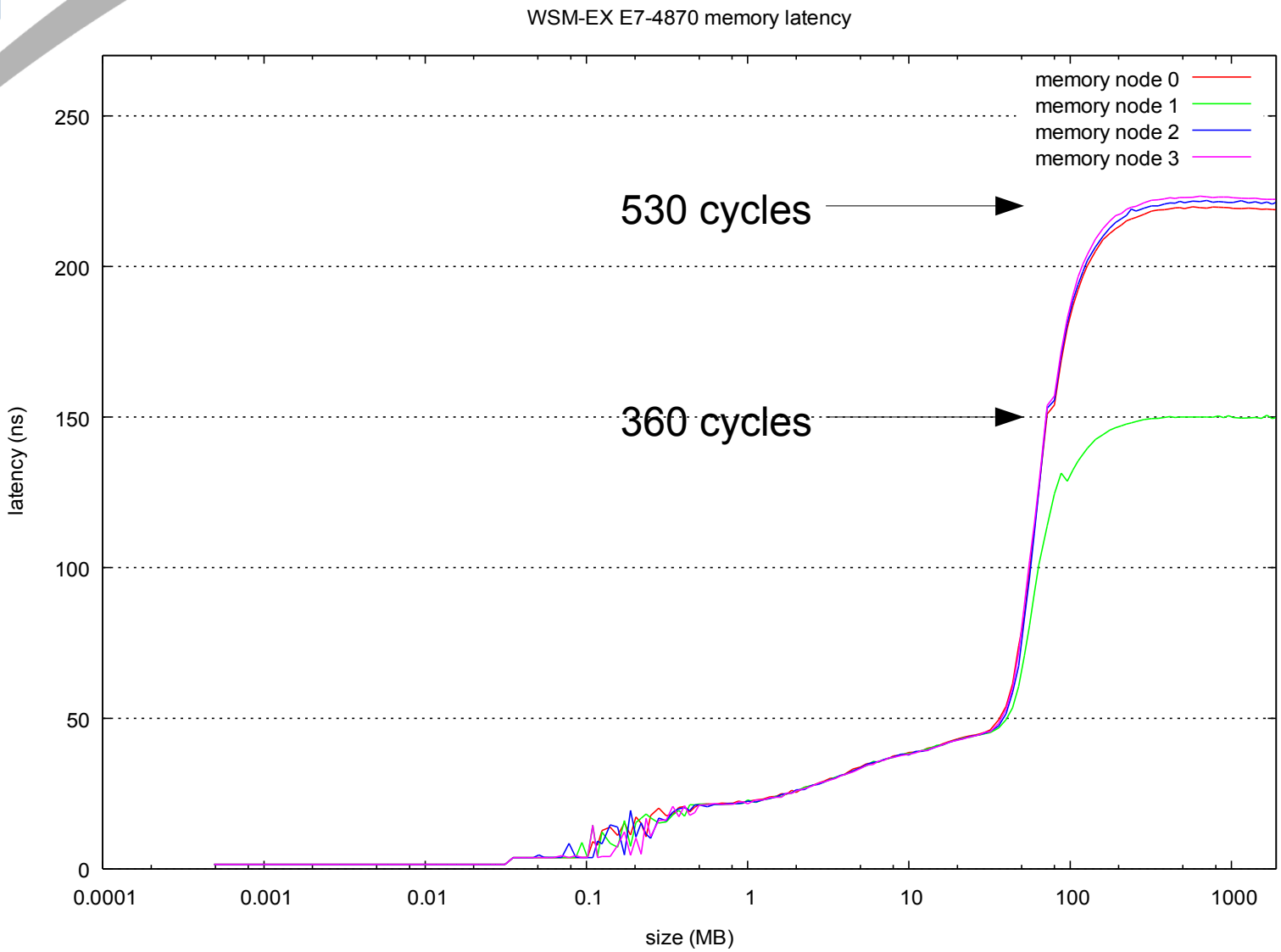
Memory latency measurements



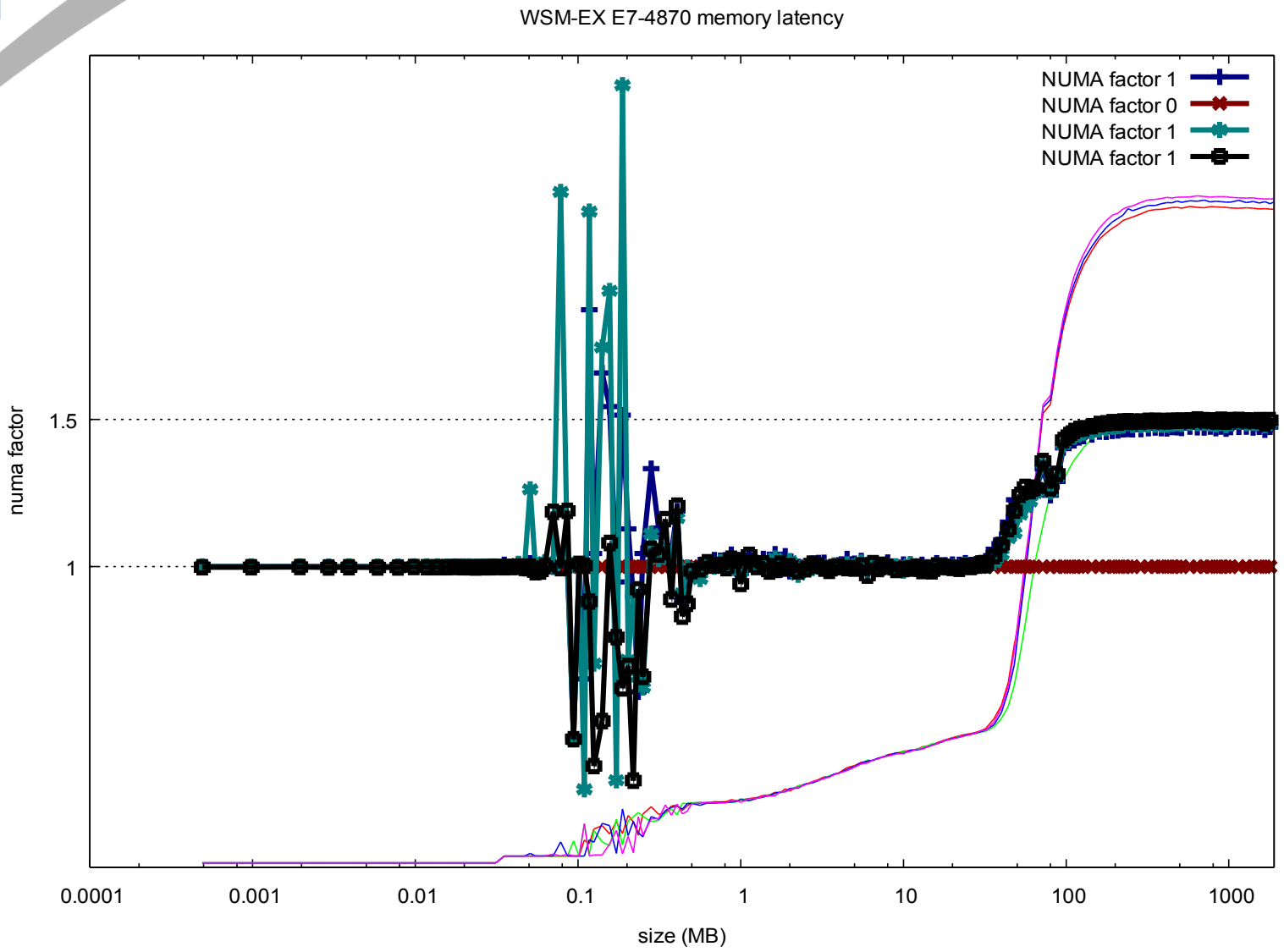
Memory latency measurements



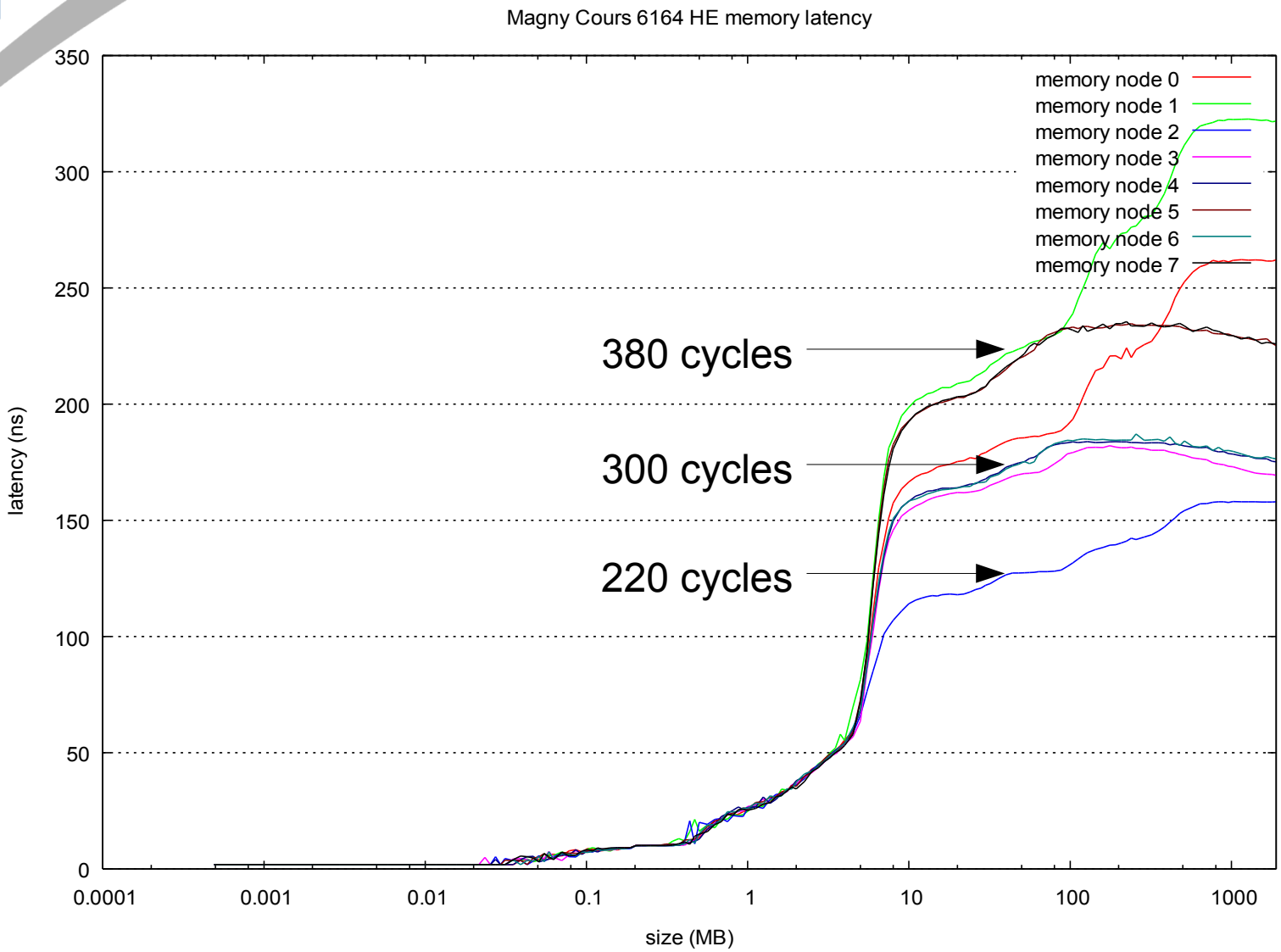
Memory latency measurements



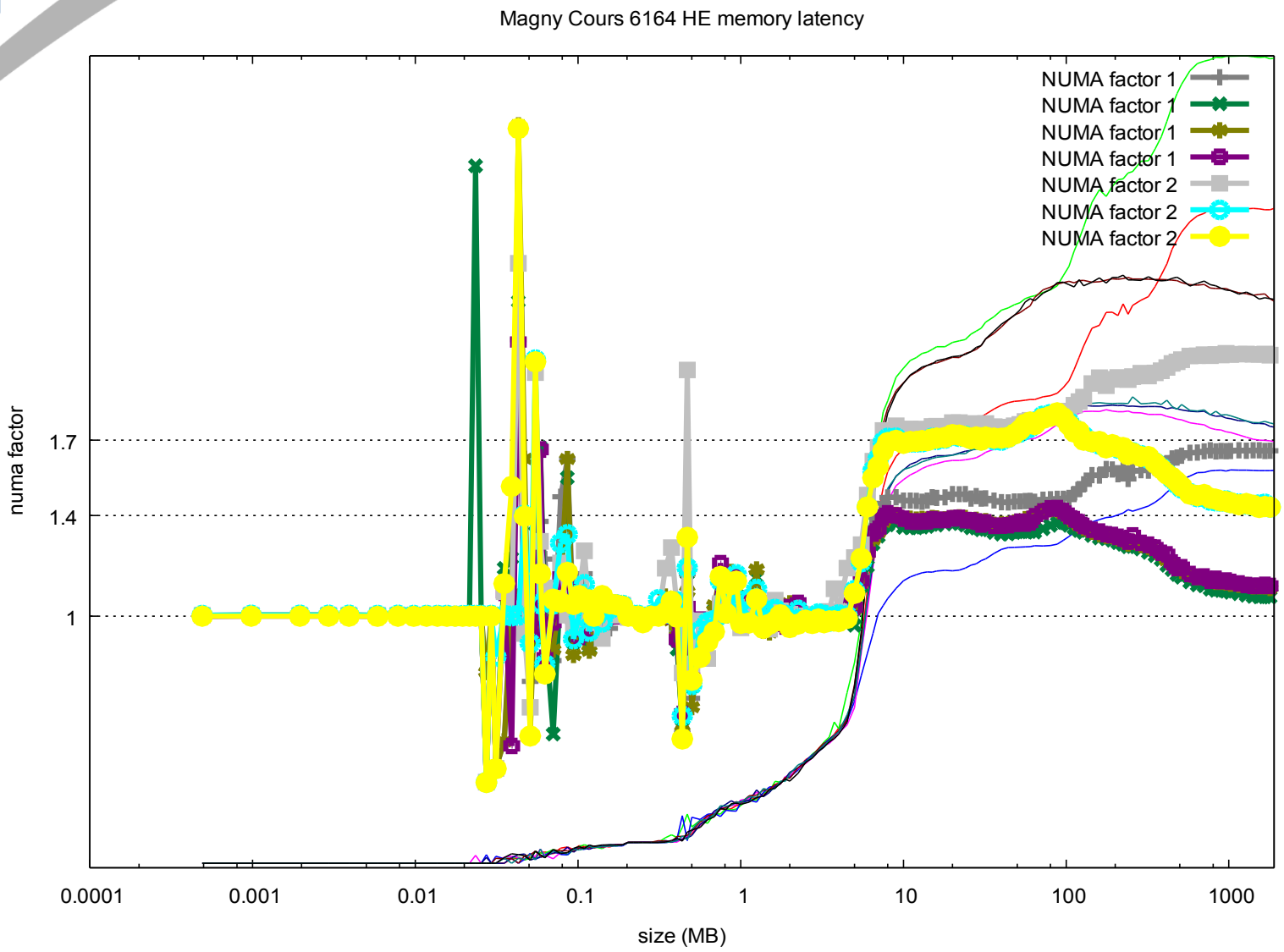
Memory latency measurements



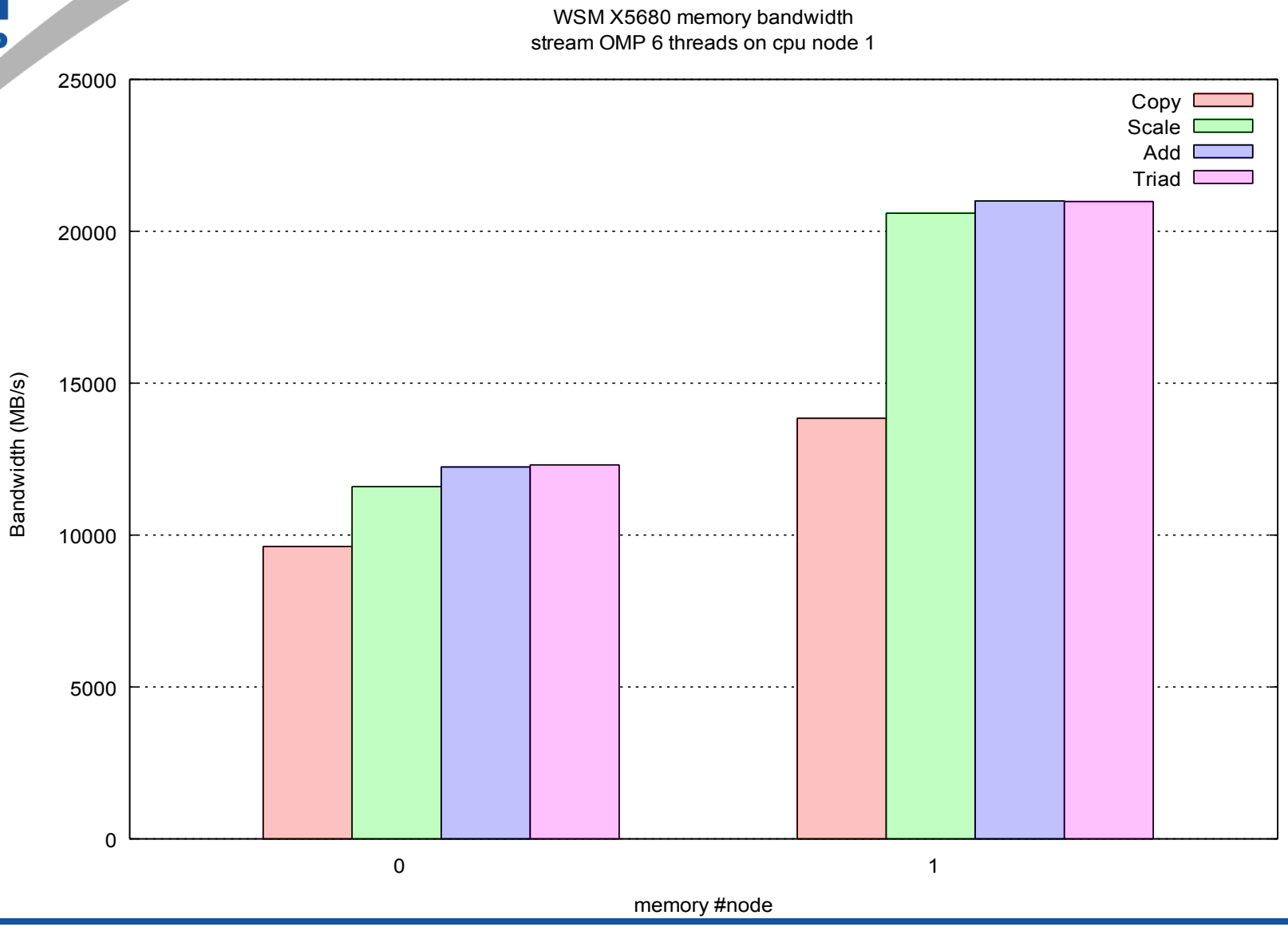
Memory latency measurements



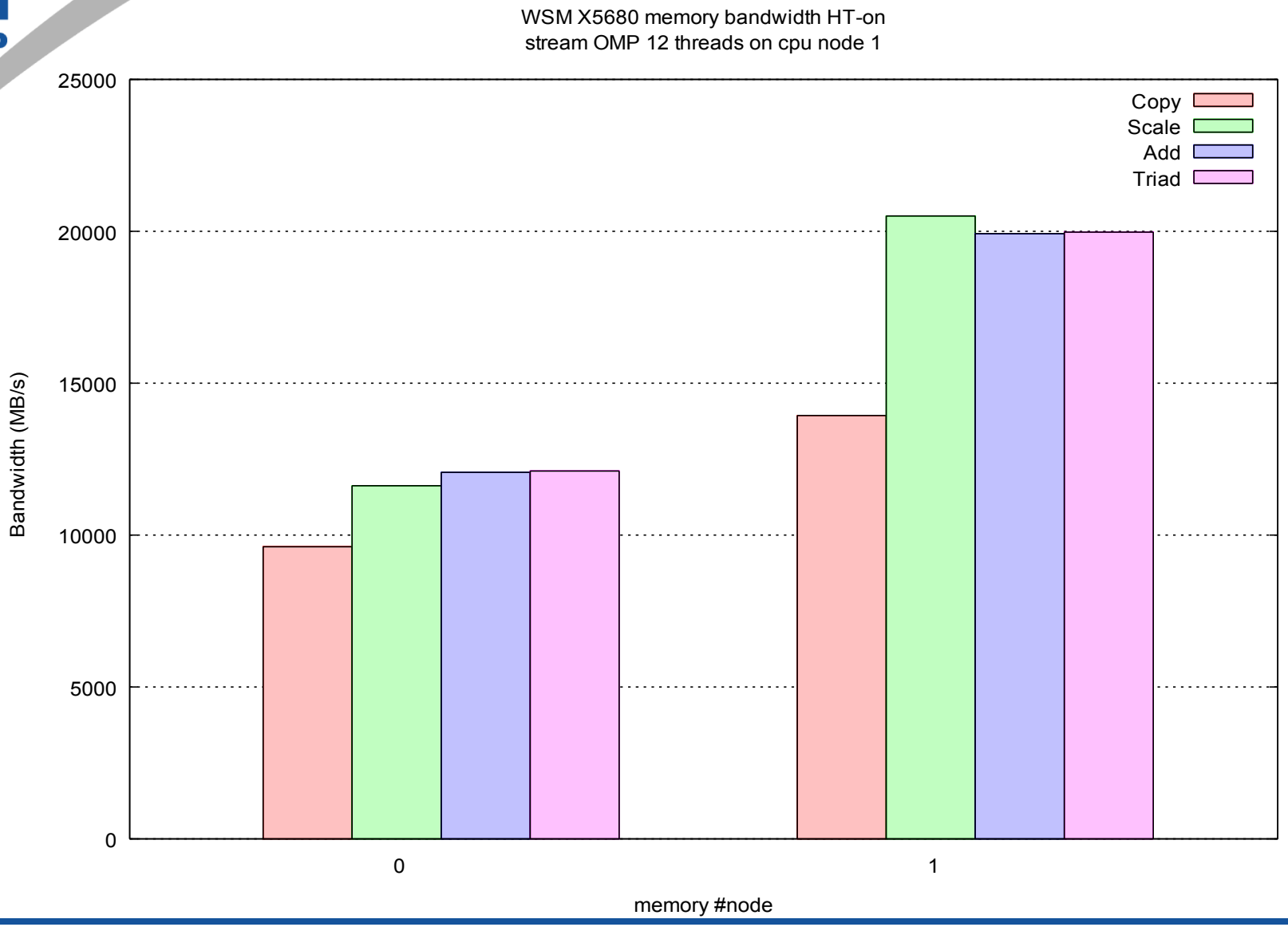
Memory latency measurements



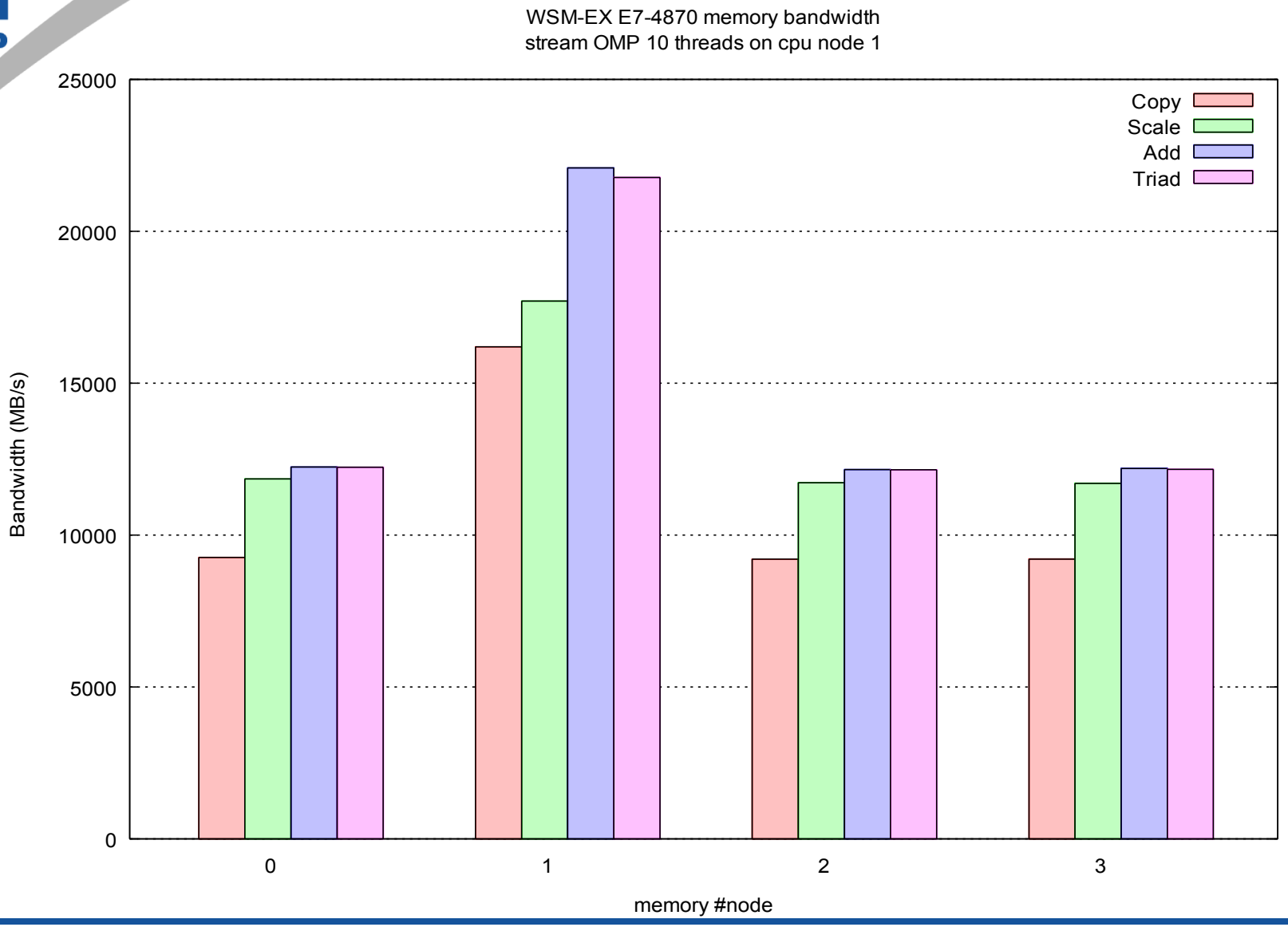
Memory bandwidth measurements



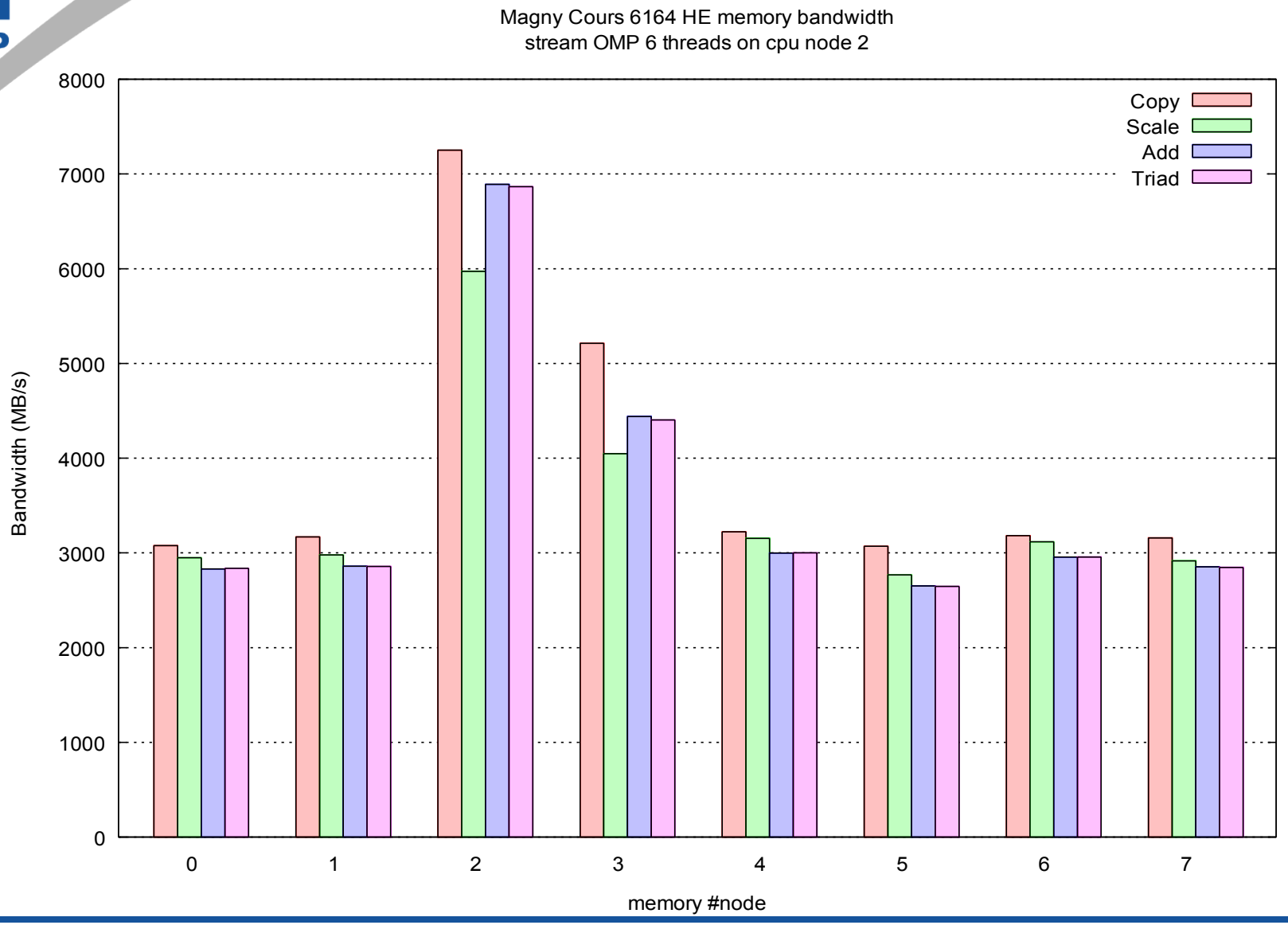
Memory bandwidth measurements



Memory bandwidth measurements



Memory bandwidth measurements



Managing OpenMP on NUMA systems

- OpenMP eases parallel application development
  - Perfect for SMP systems from previous generations
    - Flat memory model
  - What do NUMA systems change?

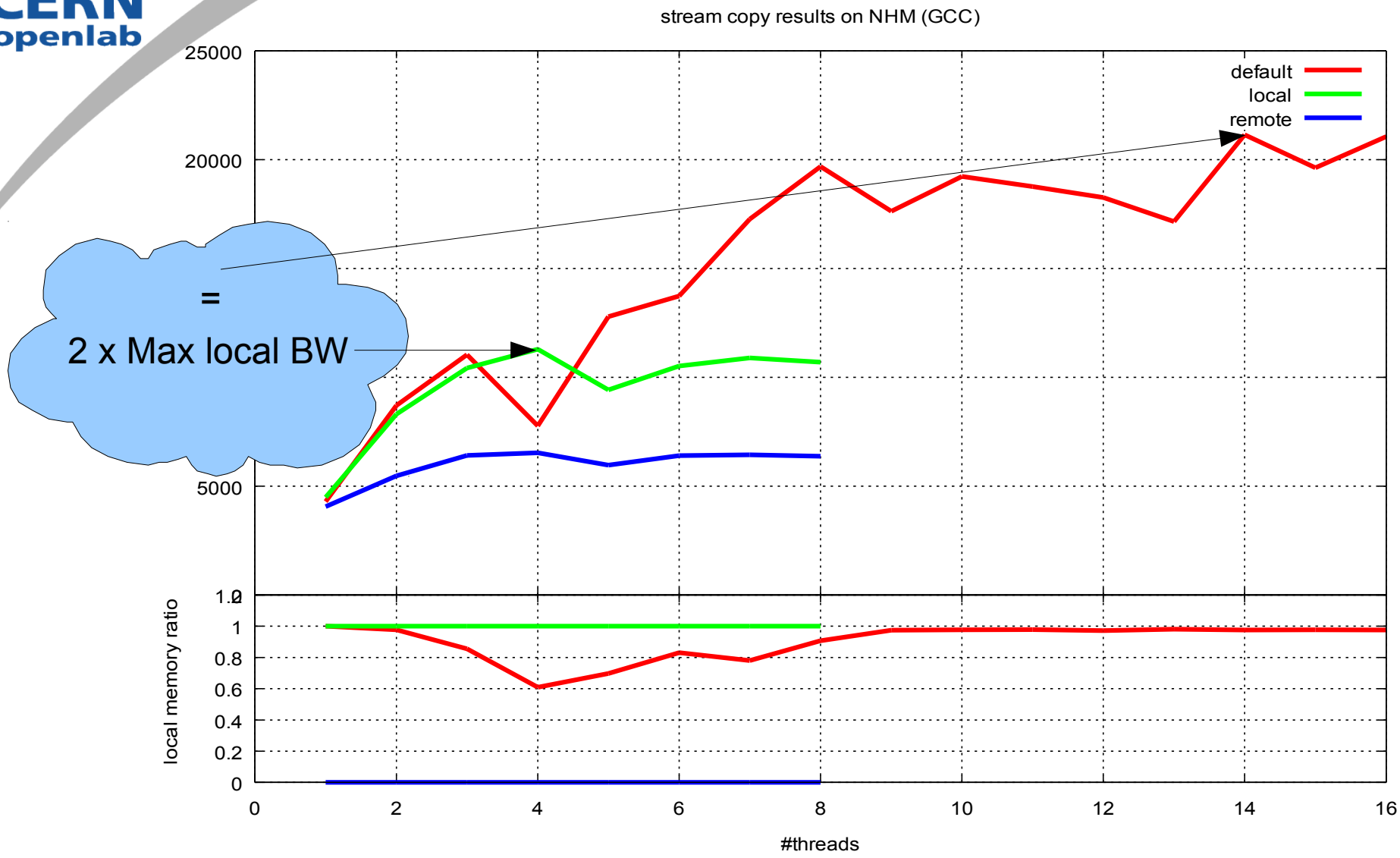
stream

- Stream is a simple bandwidth benchmark, a lot of implementations are available
  - Single threaded
  - **OpenMP**
  - MPI
  - Customs
- 3 large arrays of doubles defined (N=20000k)
  - Using 20000000*8*3=457MB of data
  - 4 operations performed and timed on those arrays:
    - Copy: c[j] = a[j]
    - Scale: b[j] = scalar * c[j]
    - Add: c[j] = a[j] + b[j]
    - Triad: a[j] = b[j] + scalar * c[j]

OpenMP on NUMA systems

- First compile stream omp with GCC
- Using CPUSET, stream is constrained to run on cores of one numa node and use:
  - Local memory: allocating memory on the same numa node
  - Remote memory: allocating memory on the other numa node
- How does stream behaves when running on the system without any constraint

OpenMP on NUMA systems: cpusets

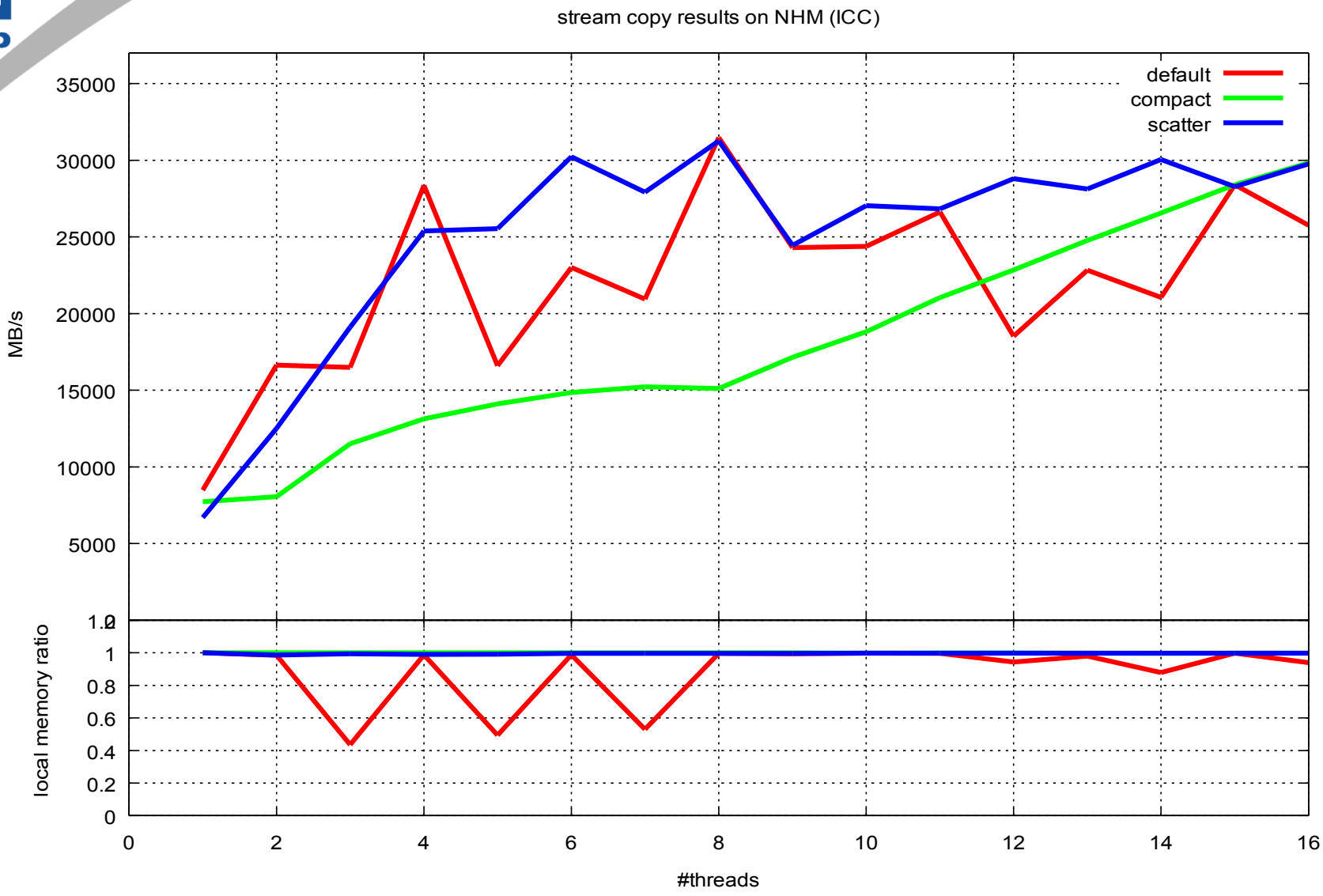


OpenMP on NUMA systems

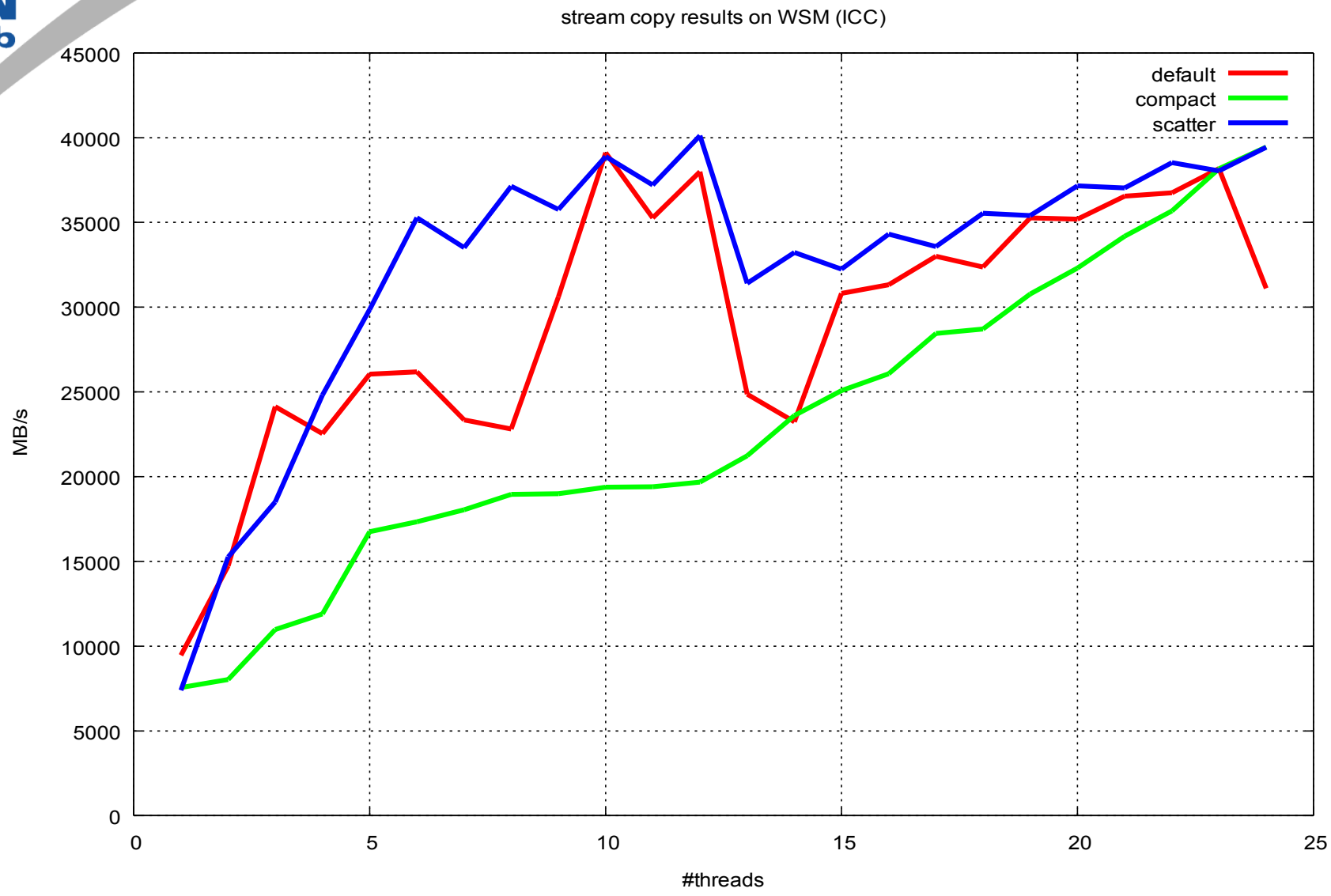
- Using ICC, environment variable KMP_AFFINITY allows to control omp threads placement
  - **Verbose**: allows to extract the scheduling information
  - **granularity=core**: control placement at the core level
  - Type:
    - **Compact**: assigns the OpenMP thread <n>+1 to a free thread context as close as possible to the thread context where the <n> OpenMP thread was placed
    - **Scatter**: distributes the threads as evenly as possible across the entire system (opposite of compact)

- Compare execution and scheduling using KMP_AFFINITY:
  - Unset
  - KMP_AFFINITY=verbose,granularity=core,compact
  - KMP_AFFINITY=verbose,granularity=core,scatter

OpenMP on NUMA systems: KMP_AFFINITY



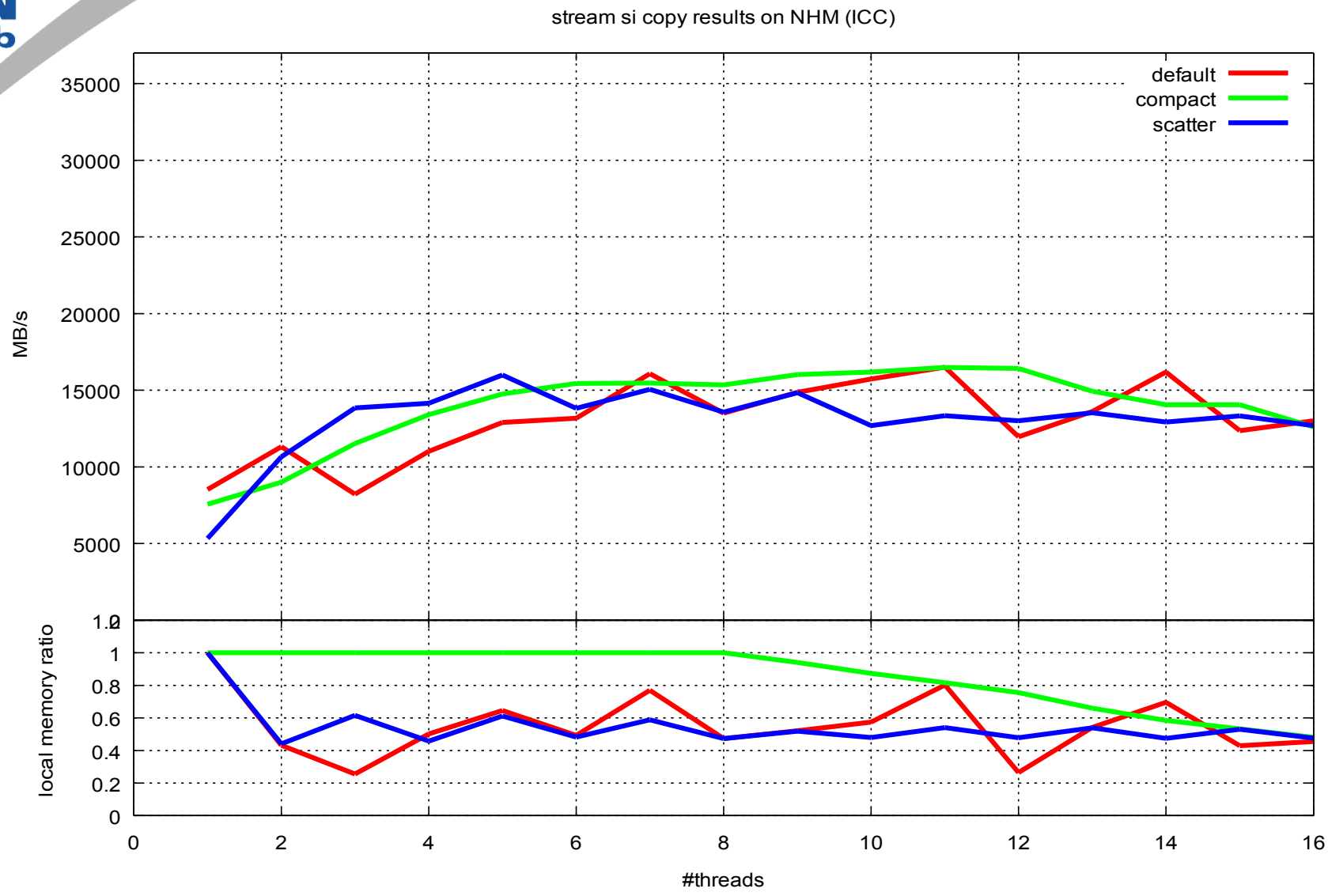
OpenMP on NUMA systems X5670



OpenMP on NUMA systems

- A hidden “detail”:
  - During all the previous measurements, memory was initialized in parallel
  - What happens if memory is initialized by the master thread?

OpenMP on NUMA systems serial initialization



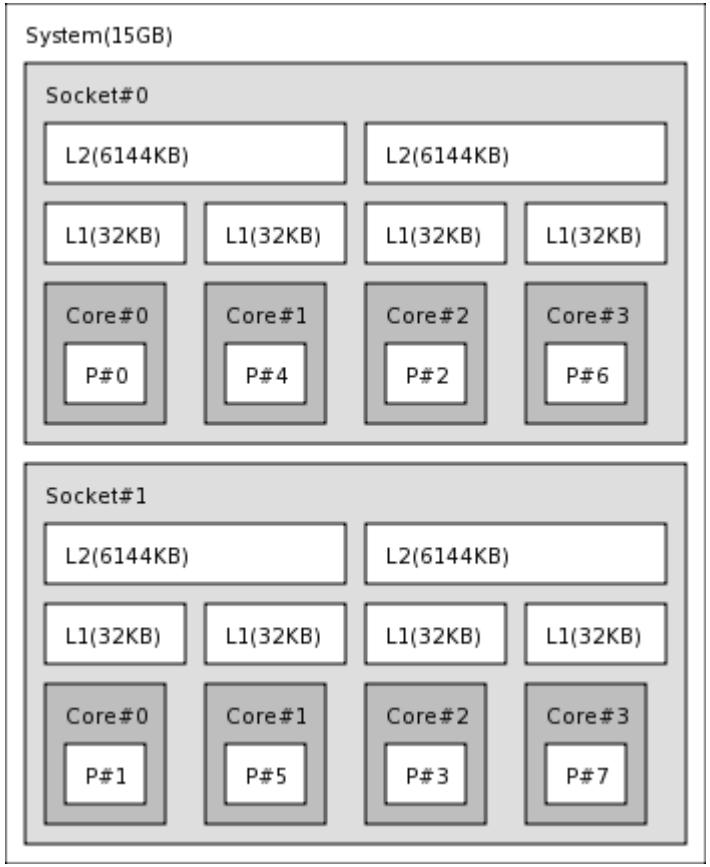
OpenMP on NUMA systems

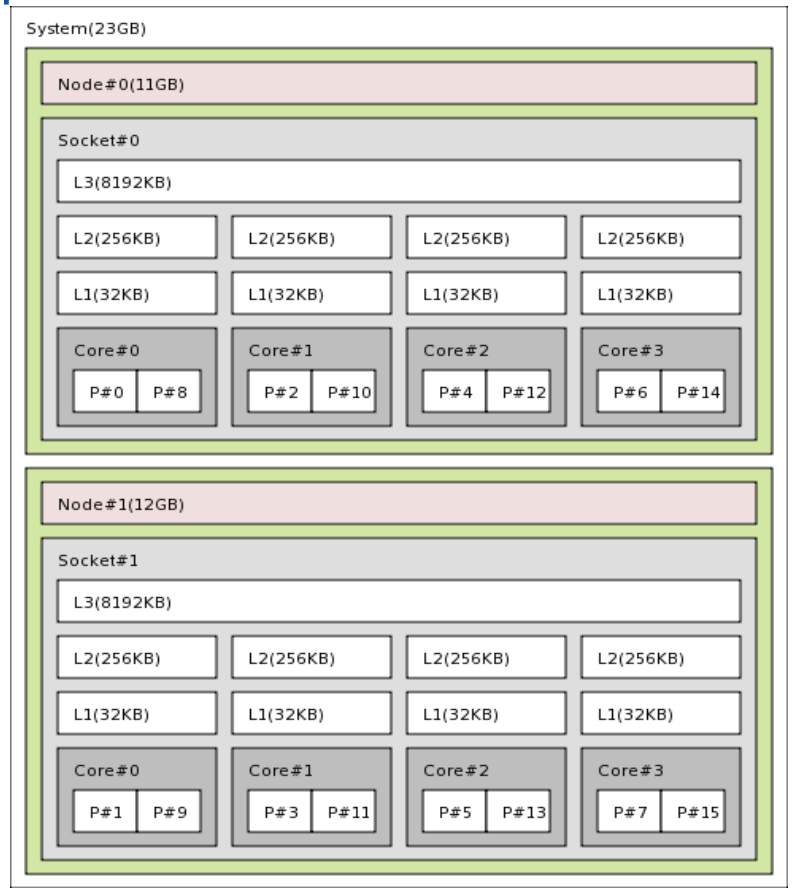
- **Parallel memory initialization** is primordial because Linux and most operating systems use memory in place: where it was first touched
  - Memory migration patches exist but are not in the kernel
- Affinity can be interesting:
  - Using a numa DP system as 2 SMP systems in one server (compact, cpuset, numactl)
  - Scatter, can help
  - Explicit scheduling is possible

Need to extract the topology of the system

Extracting system topology

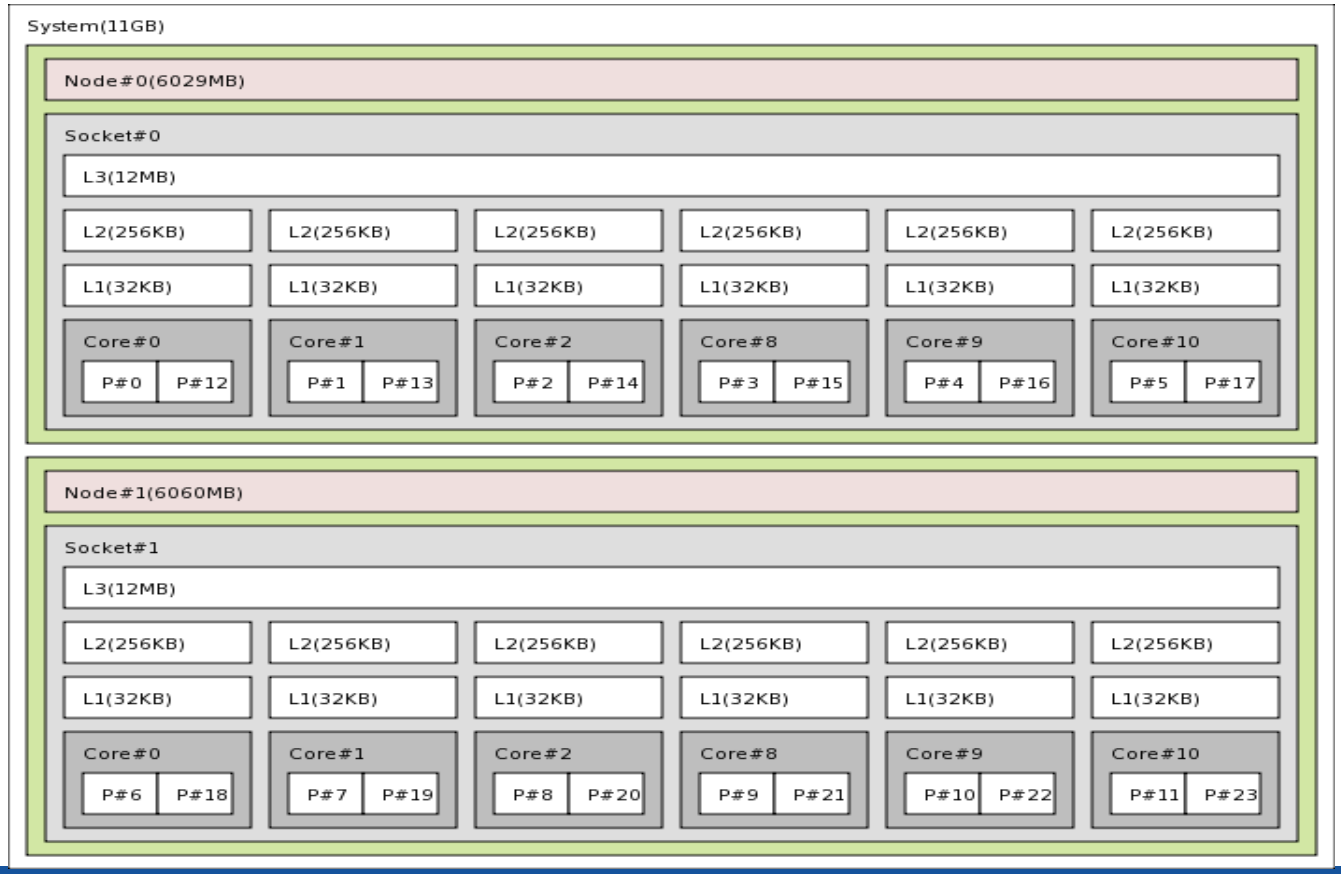
- Hwloc
  - Displays the topology (lstopo)
  - Offers some bindings to cpusets





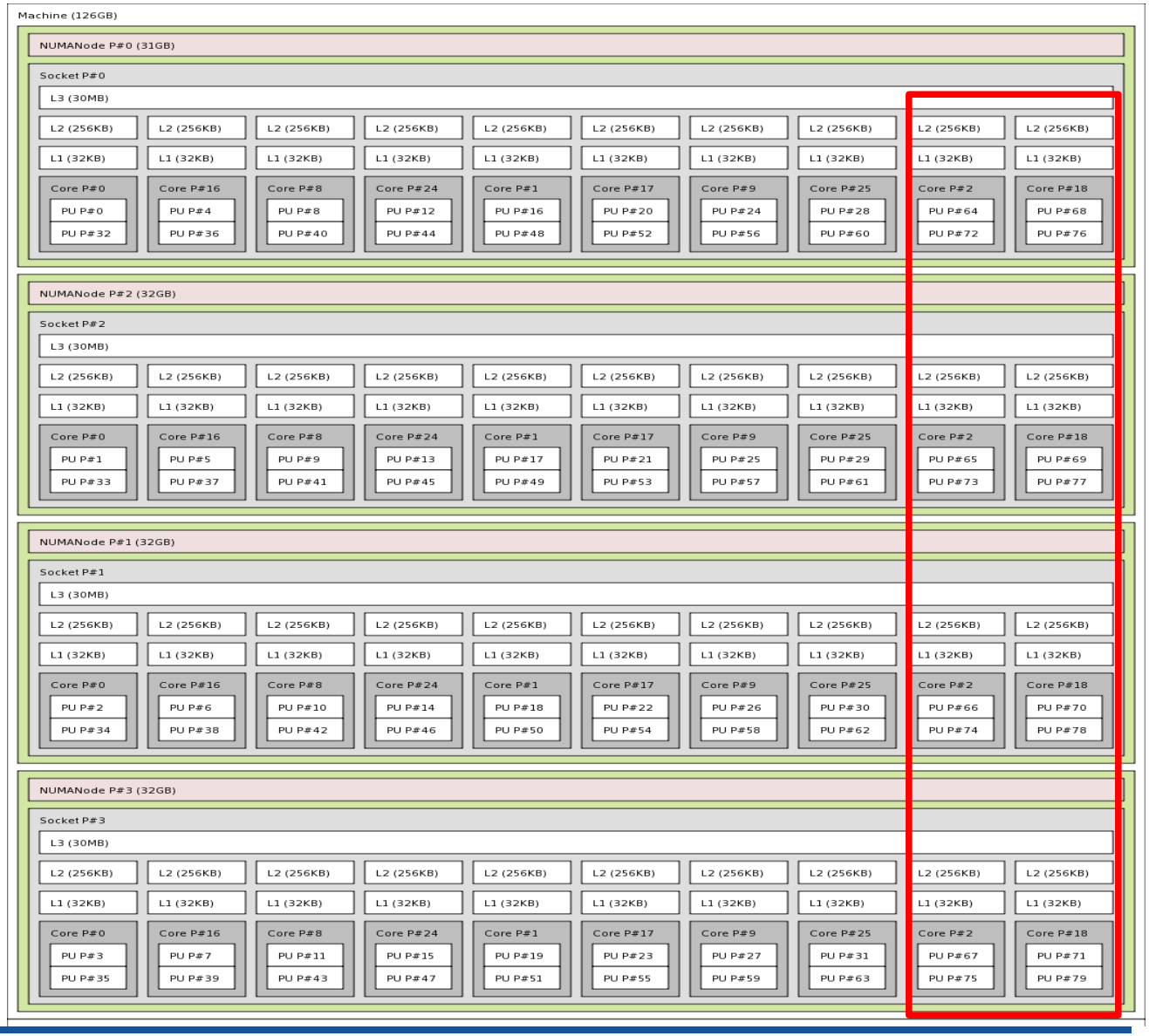
Extracting system topology

- Westmere:
  - 6 cores, 2 QPI links per socket, Integrated Memory Controller, 32nm



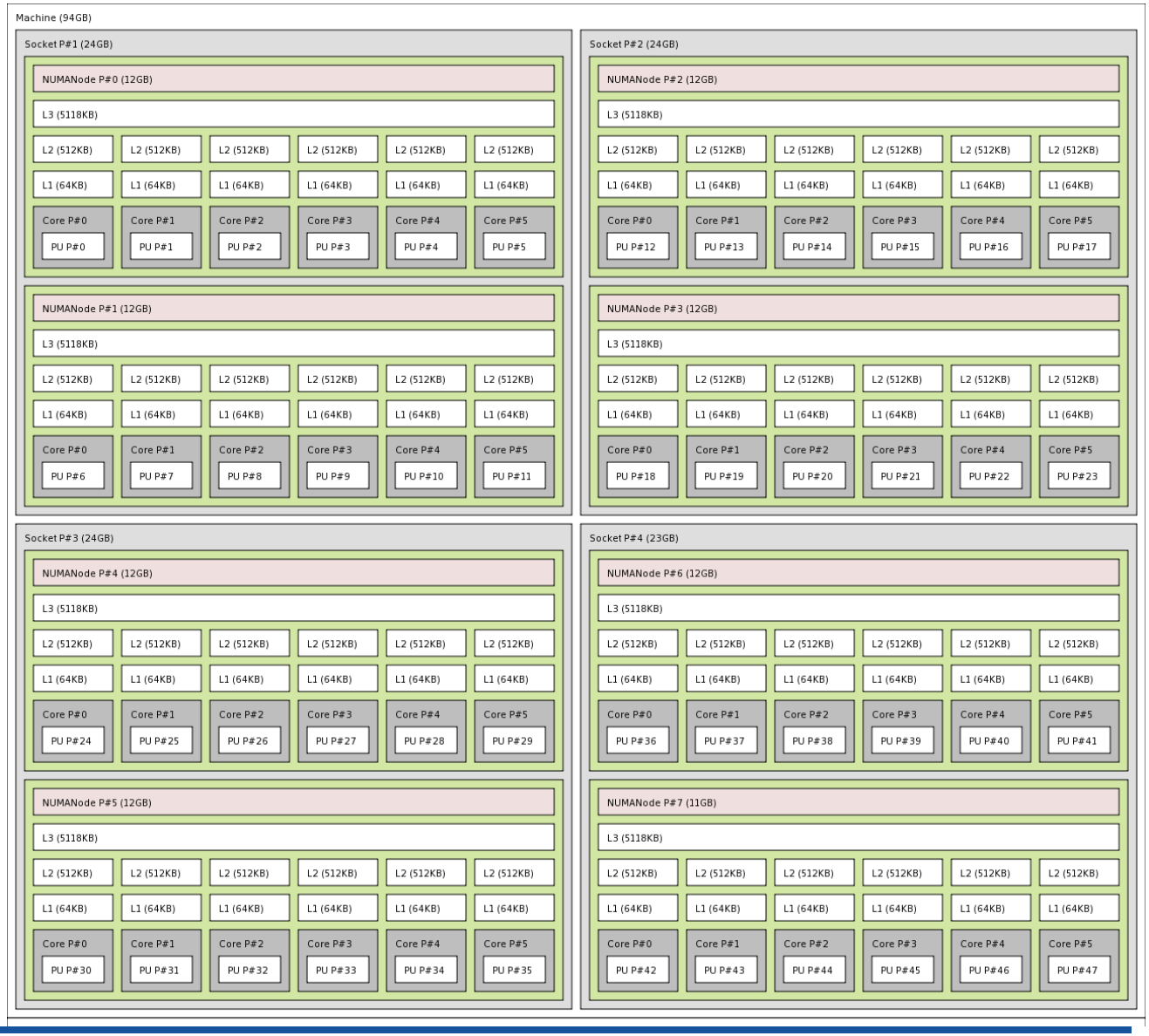
Extract the topology

- WSM-EX
  - 40 cores
  - 4 sockets



Extract the topology

- Magny-Cours
  - 48 cores
  - 4 sockets



Questions ?