



# Introduction to the Job Performance Monitoring JobMon at NHR@KIT

Begatim Bytyqi, Holger Obermaier | 15. Apr. 2026

- 1. JobMon - Overview**
- 2. Demo Application Workloads**
- 3. Login Page**
- 4. Welcome Page**
- 5. Jobs page**
- 6. Spider Plot**
- 7. Per job page**
- 8. Roofline Plot**

# JobMon

## History

- Started as a research project as part of J. Schmitt's Bachelor's thesis in 2021/2022
- Contributions from employees (B. Bytyqi, H. Obermaier) and student assistants (D. Schild, J. Schmitt, F. Wedler)
- Basis for D. Schild's master's thesis in 2024/2025

## Overview


- Web Service: JobMon [↗](#)
- Access with HoreKa user account
- Same networks restrictions apply as for HoreKa (VPN)
- Visualizes performance metrics
- Aggregates performance metrics for improved clarity
- Metrics are continuously collected on cluster nodes
- Metrics are stored ⇒ Performance changes over time can be tracked.

# Demo Application Workloads

## Categories

- *Compute Bound*
  - Performance is limited by floating point performance
  - Benchmark *DGEMM*[↗](#) performs a matrix matrix multiplication
- *Memory bound*
  - Performance is limited by memory bandwidth
  - Benchmark *Stream*[↗](#) / *BabelStream*[↗](#) performs vector computations
  - Benchmark *HPCG* performs the conjugate gradients method with a sparse matrix
- *Communication bound*
  - Performance is limited by interconnect bandwidth or latency
  - Benchmark *OSU Micro-Benchmarks*[↗](#) performs MPI point to point communication
- *IO bound*
  - Performance is limited by filesystem throughput or metadata rate
  - Benchmark *IOR*[↗](#) performs parallel IO operations

# Login page



☾

## Welcome to Jobmon

Jobmon monitors jobs running on the High Performance Computing system HoreKa. Jobmon visualizes the usage of the computing system and helps you understand the performance of your job.

### Login

Login with OIDC is recommended for most users

Login with OIDC

Login with local account

# Welcome Page

Navigation bar with icons for home, calendar, search, dark mode, and refresh.

Search user/job



## Welcome to Jobmon

Jobmon monitors jobs running on the High Performance Computing system HoreKa. Jobmon visualizes the usage of the computing system and helps you understand the performance of your job.


### Your Activities

Running jobs  
**2**


Jobs in the last 24 hours  
**5**

Jobs in the last week  
**5**

### Your account


 Username: bq0742  
Role: user

### Your jobs

 This system monitors jobs. You can take a look at them in by using on the link below.

[Your Jobs](#)

### Need Help?

 You can find further information about jobmon by using the link below.

[Further Information](#)



# Jobs page

⇒ Overview of the personal HoreKa batch jobs

## Filter options

- Batch partition
- Number of nodes or GPUs
- Running or finished jobs
- Job execution time
- Exit code ⇒ successful or non-successful job run
- Tags assigned to the job ⇒ group jobs by your needs

# Jobs page - Filter

Users:

Exit Code:

Jobs per page:

From:

To:

Sort by:

Nb. nodes

Nb. GPUs

Toggle view



# Jobs page - Spider Plot

## Spider Plot

- Plot that characterizes the overall job performance
- High level view on performance limitations
- Allows categorization as:
  - Compute bound
  - Memory bound
  - Communication bound
  - IO bound
- Average and maximum values for the metrics:
  - CPU floating point operations per second (FLOPs)
  - CPU memory bandwidth
  - GPFS IO operations
  - GPFS metadata operations
  - GPU utilization
  - GPU memory utilization
  - InfiniBand bandwidth

# Spider Plot - Compute Bound Workl. (cpuonly)

Id: **3015837**

User: bq0742 (hk-project-scs)

Partition: cpuonly

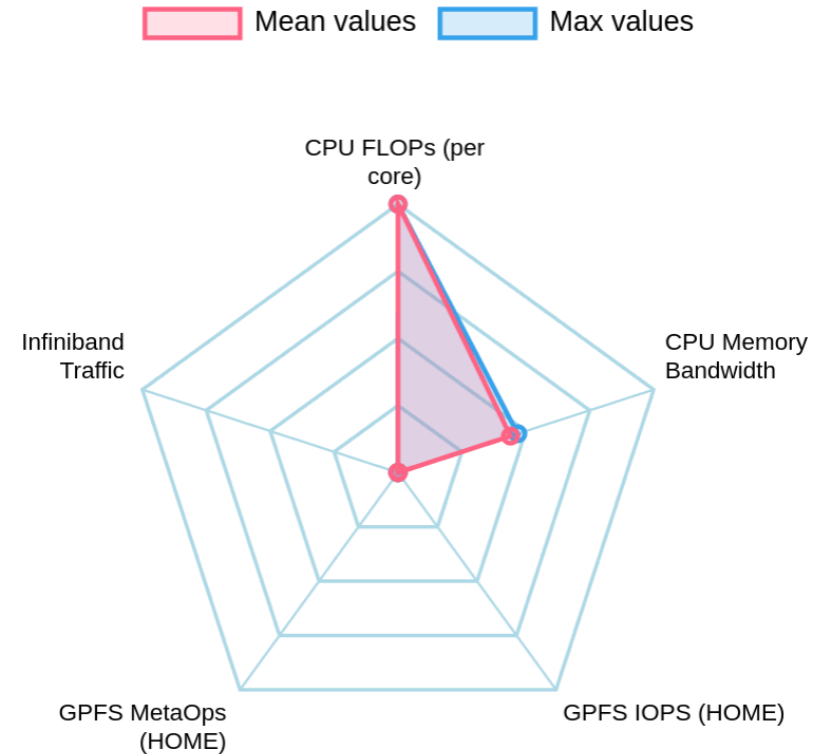
Name: cpu.dgemm

Nodes: 1

Start: 27.3.2025, 16:25:13

End: 27.3.2025, 17:21:28

Exit code: **0**



# Spider Plot - Memory Bound Workl. (cpuonly)

Id: **3015838**

User: bq0742 (hk-project-scs)

Partition: cpuonly

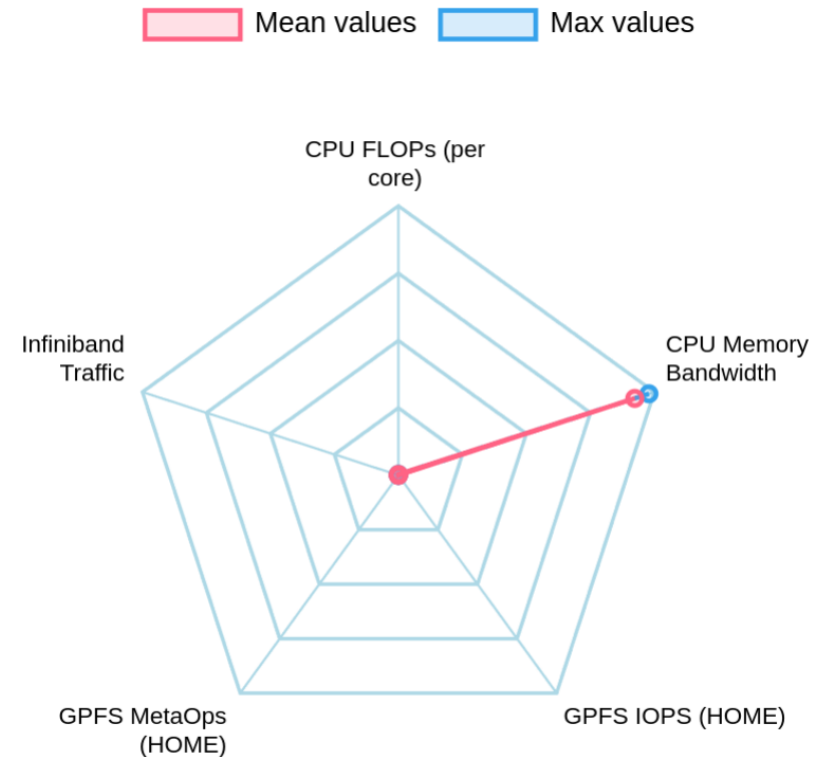
Name: cpu.stream

Nodes: 1

Start: 27.3.2025, 16:25:13

End: 27.3.2025, 17:27:08

Exit code: **0**



# Spider Plot - Communication Bound Workl. (cpuonly)

Id: **3019847**

User: bq0742 (hk-project-scs)

Partition: cpuonly

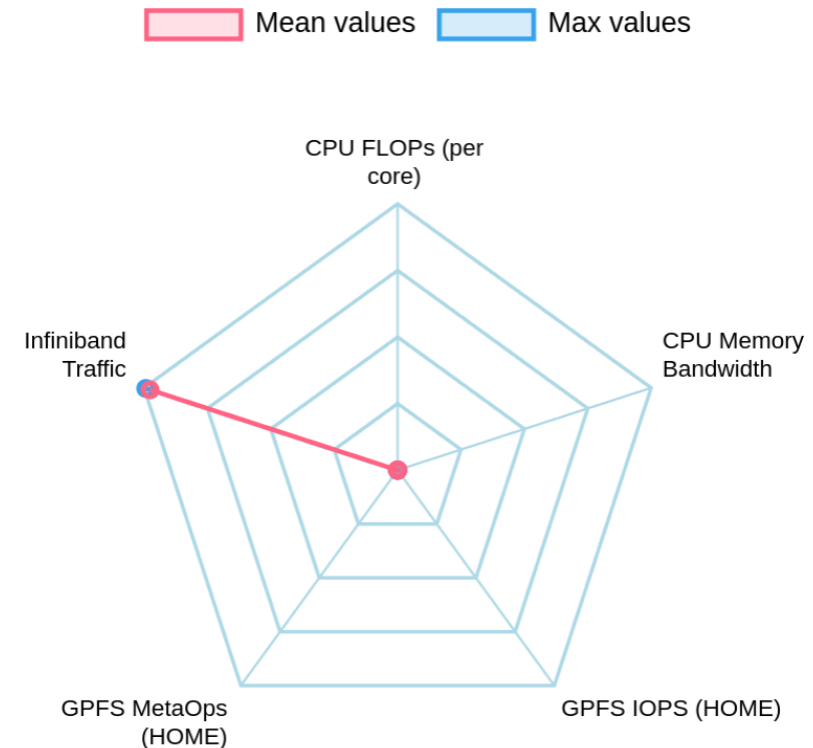
Name: cpu.omb,mpi,node

Nodes: 2

Start: 28.3.2025, 15:30:33

End: 28.3.2025, 16:28:00

Exit code: **0**



# Spider Plot - Compute Bound Workl. (accelerated)

Id: **3015828**

User: bq0742 (hk-project-scs)

Partition: accelerated

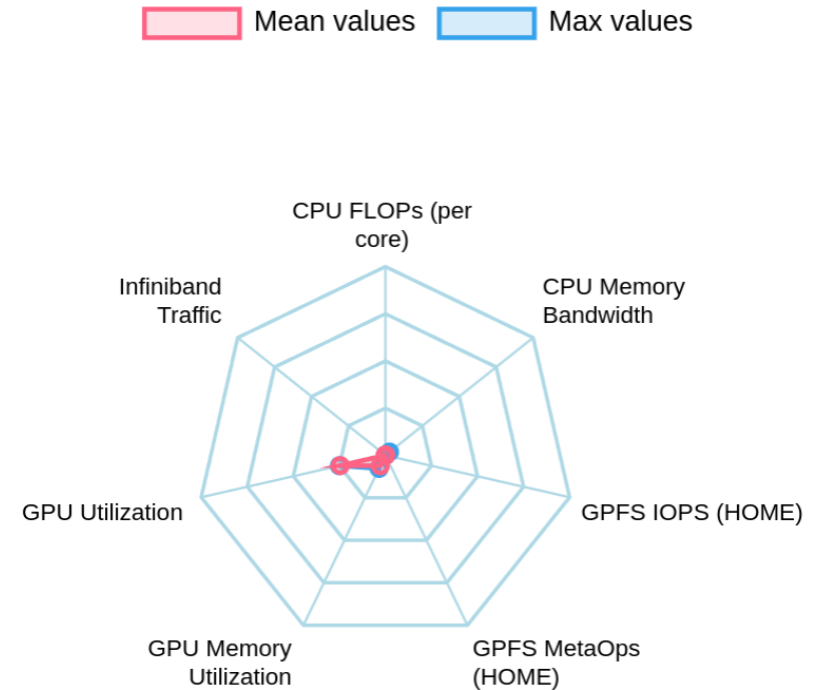
Name: gpu.dgemm

Nodes: 1                  GPUs: 4

Start: 26.3.2025, 13:58:52

End: 26.3.2025, 14:57:36

Exit code: **0**



# Spider Plot - Memory Bound Workl. (accelerated)

Id: **3013359**

User: bq0742 (hk-project-scs)

Partition: accelerated

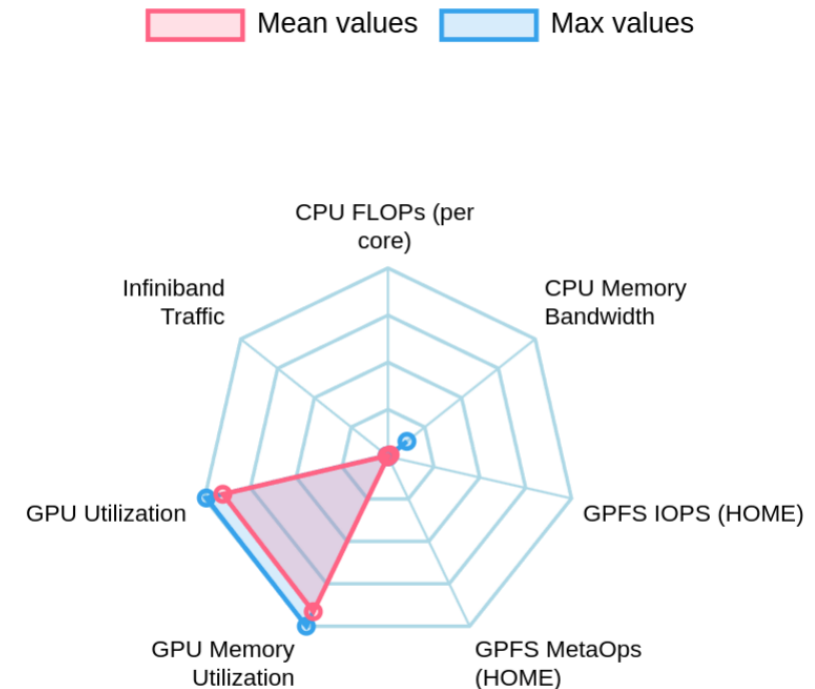
Name: gpu.hpcg,apptainer

Nodes: 1                  GPUs: 4

Start: 25.3.2025, 11:56:12

End: 25.3.2025, 13:02:09

Exit code: **0**



# Spider Plot - Comm. Bound Workl. (accelerated)

Id: **3016899**

User: bq0742 (hk-project-scs)

Partition: accelerated

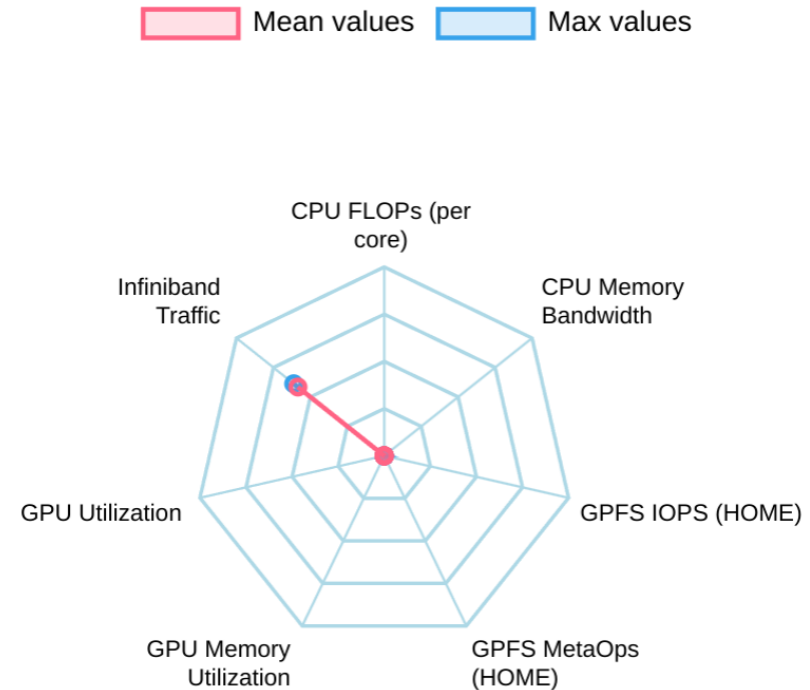
Name: gpu.osu,apptainer

Nodes: 2                  GPUs: 2

Start: 27.3.2025, 10:36:13

End: 27.3.2025, 10:49:12

Exit code: **0**



# Per job page

## Configuration options

- Select subset of nodes
- Select time range
- Select subset of metrics
- Set tags ⇒ Mark jobs for quick access

## Toggles

- *Automatic Scaling* ⇒ Select y-axis limits depending on metrics limits
- *Quantile View* ⇒ Only plot quantiles
- *Changepoints* ⇒ Identify times where performance metric behavior changes

# Per job page - Options, Toggles, ...

Home | Search user/job

Id: **3019847**  
User: bq0742 (hk-project-scs)  
Partition: cpuonly  
Name: cpu.omb,mpi,node  
Nodes: 2  
Start: 28.3.2025, 15:30:33  
End: 28.3.2025, 16:28:00  
Exit code: **0**  
course

Mean values | Max values

**Roofline plot**

Nodes (2)

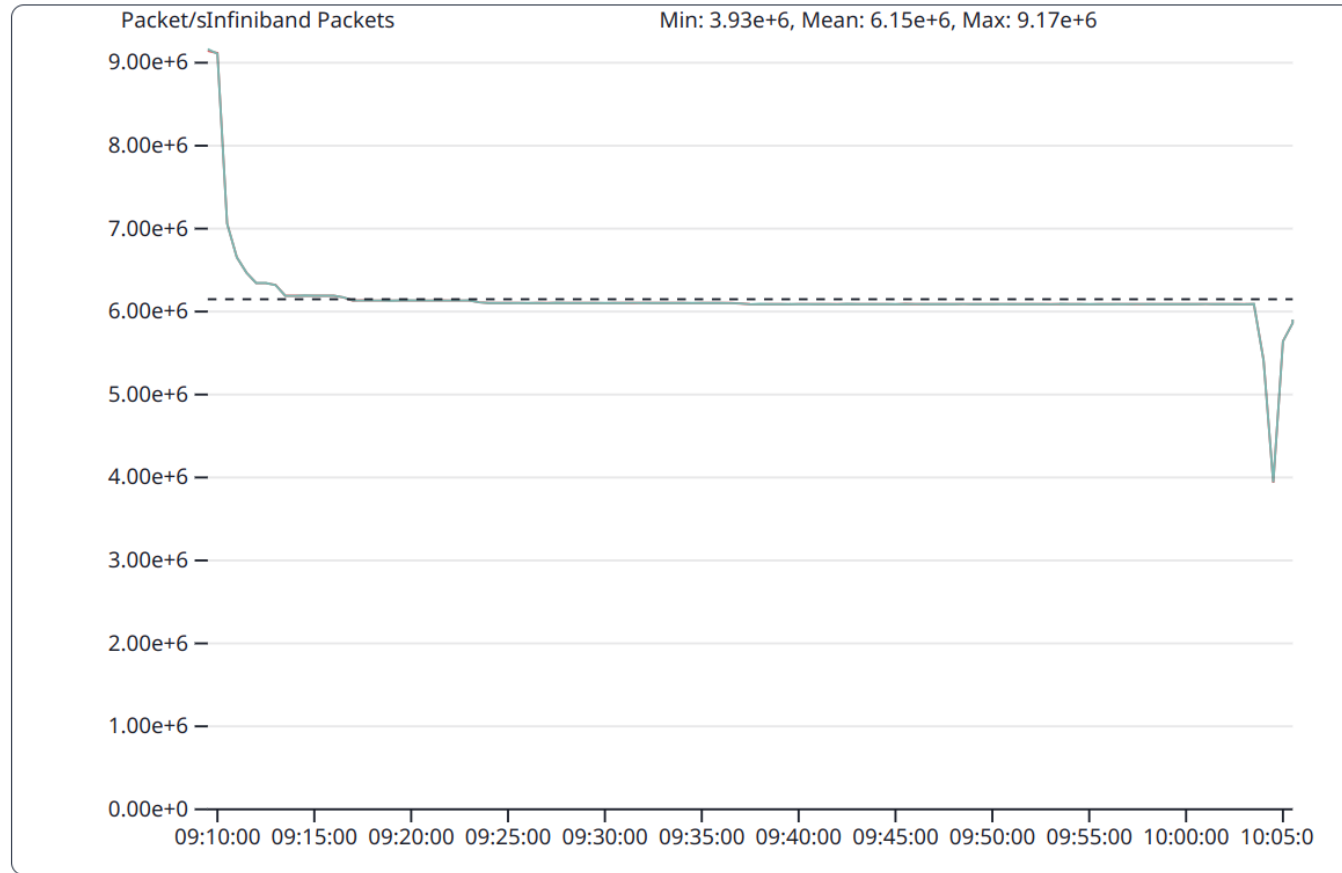
Metrics

28.3.2025, 15:30:33 | 28.3.2025, 15:44:54 | 28.3.2025, 15:59:16 | 28.3.2025, 16:13:38 | 28.3.2025, 16:28:00

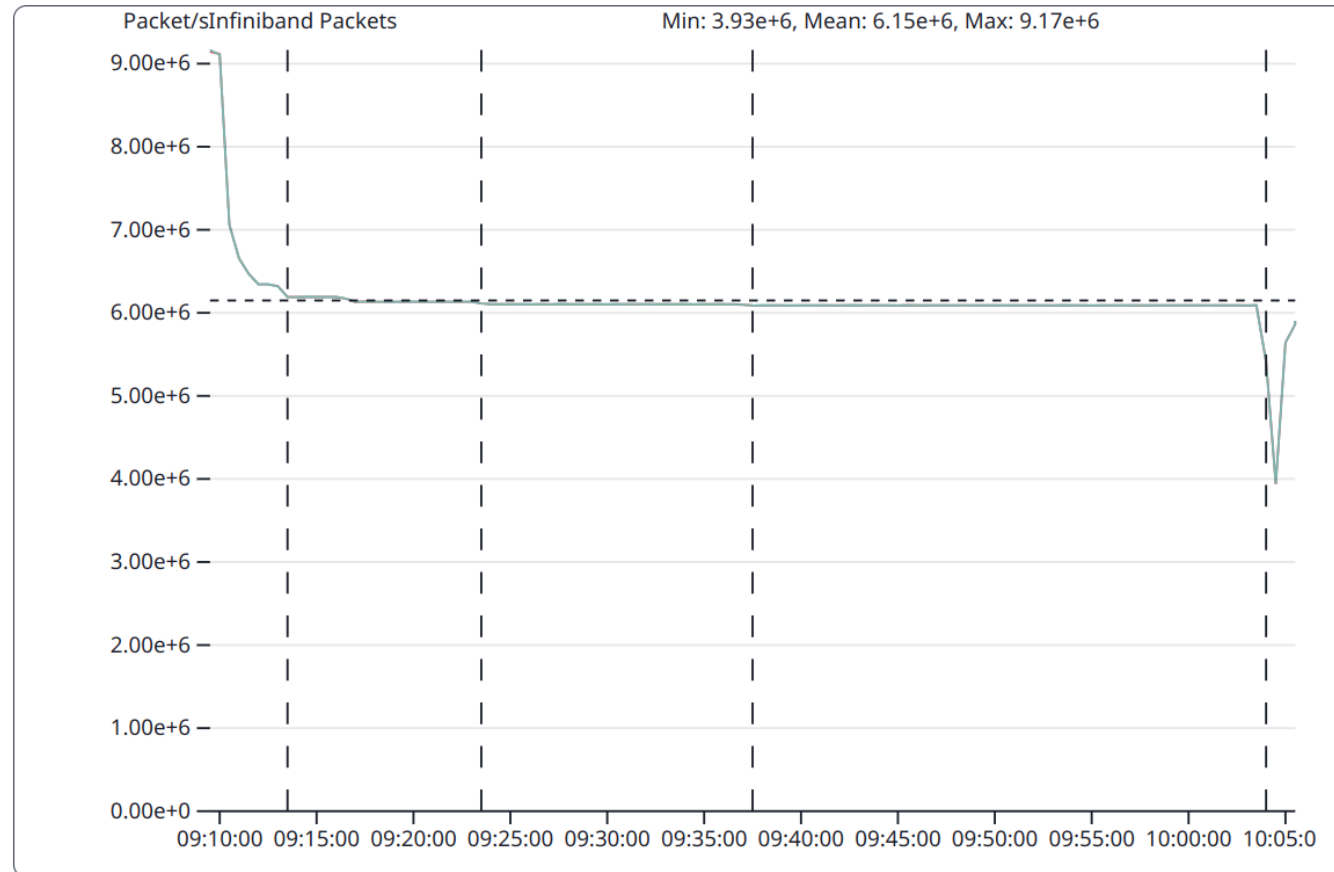
Changepoints  | Quantile View  | Automatic Scaling

Filesystem | Memory | Energy | Interconnect | Performance | Temperature

# Toggle: Changepoints Off - Infiniband Packets



# Toggle: Changepoints On - Infiniband Packets



# Toggle: Changepoints - Insight

## Insight

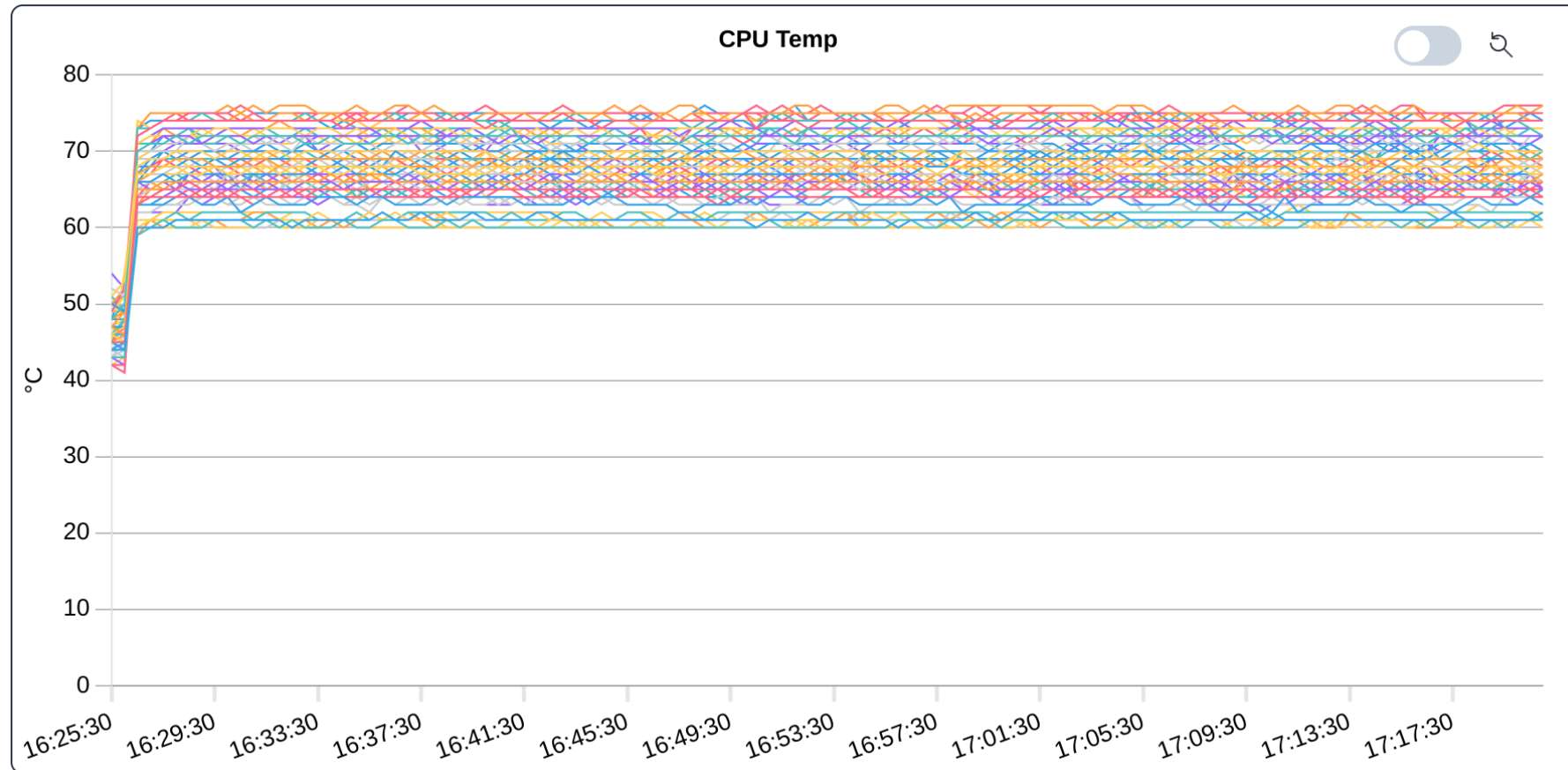
- OSU Micro-Benchmark performs communication with increasing message size
- Message size influences performance
  - ⇒ Visible as steps in the graph
- Steps can be detected by *changepoint algorithm*
- *Changepoints* marked by vertical lines

# Per Job Page - Toggle: Quantile View

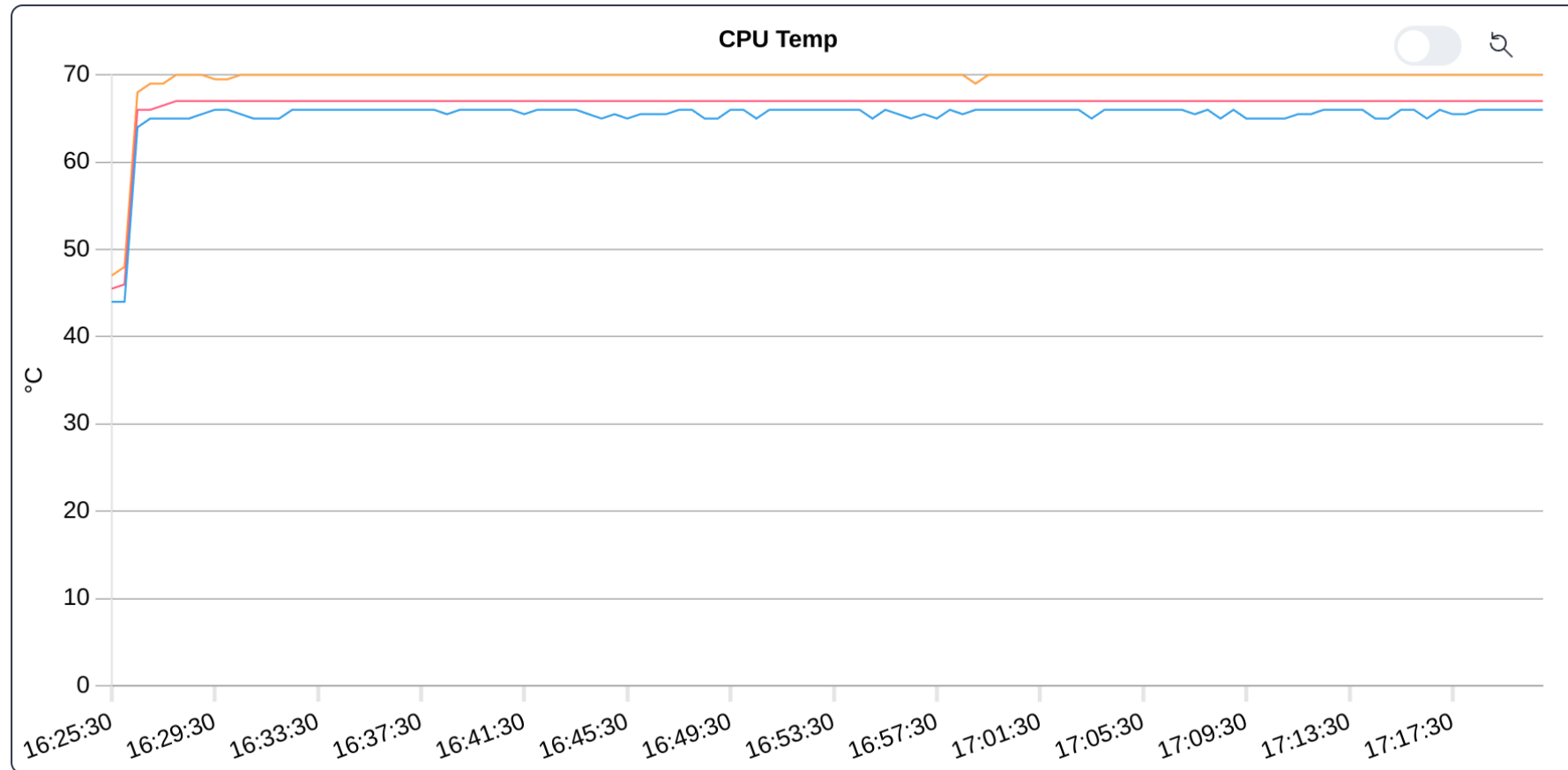
## Quantile View

- Improve clarity when too many graphs are displayed in one diagram
- Only three graphs (25% / 50% / 75% Quantile = Quartiles) are drawn
- 25% Quantile  $\Rightarrow$  25% of the measured values are below this graph
- 50% Quantile  $\Rightarrow$  Median
- Difference between upper and lower Quantile  $\Rightarrow$  Measure for the spread of the metrics

# Toggle: Quantile View Off - CPU Temperature



# Toggle: Quantile View On - CPU Temperature

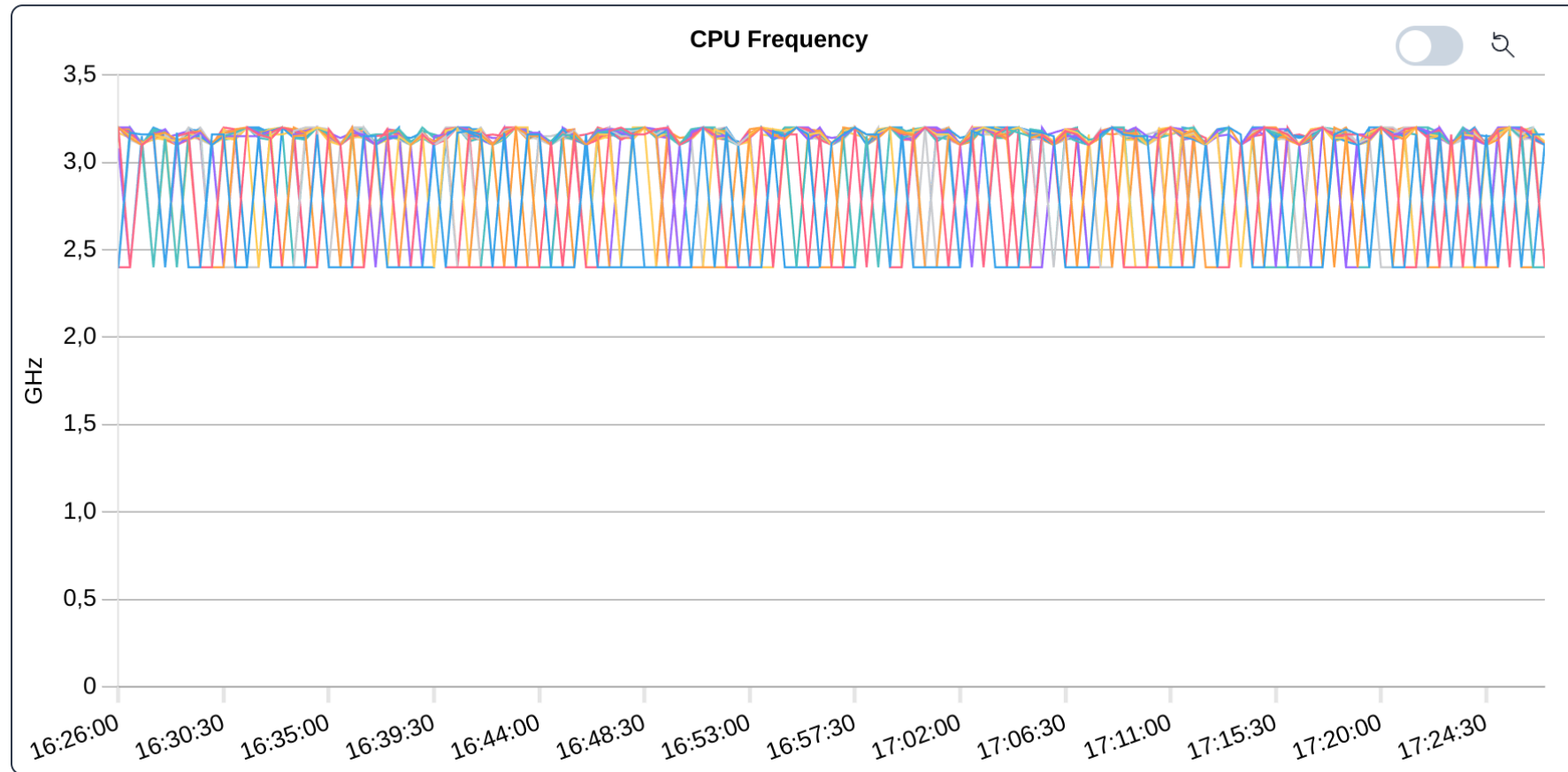


# Quantile View - Insight

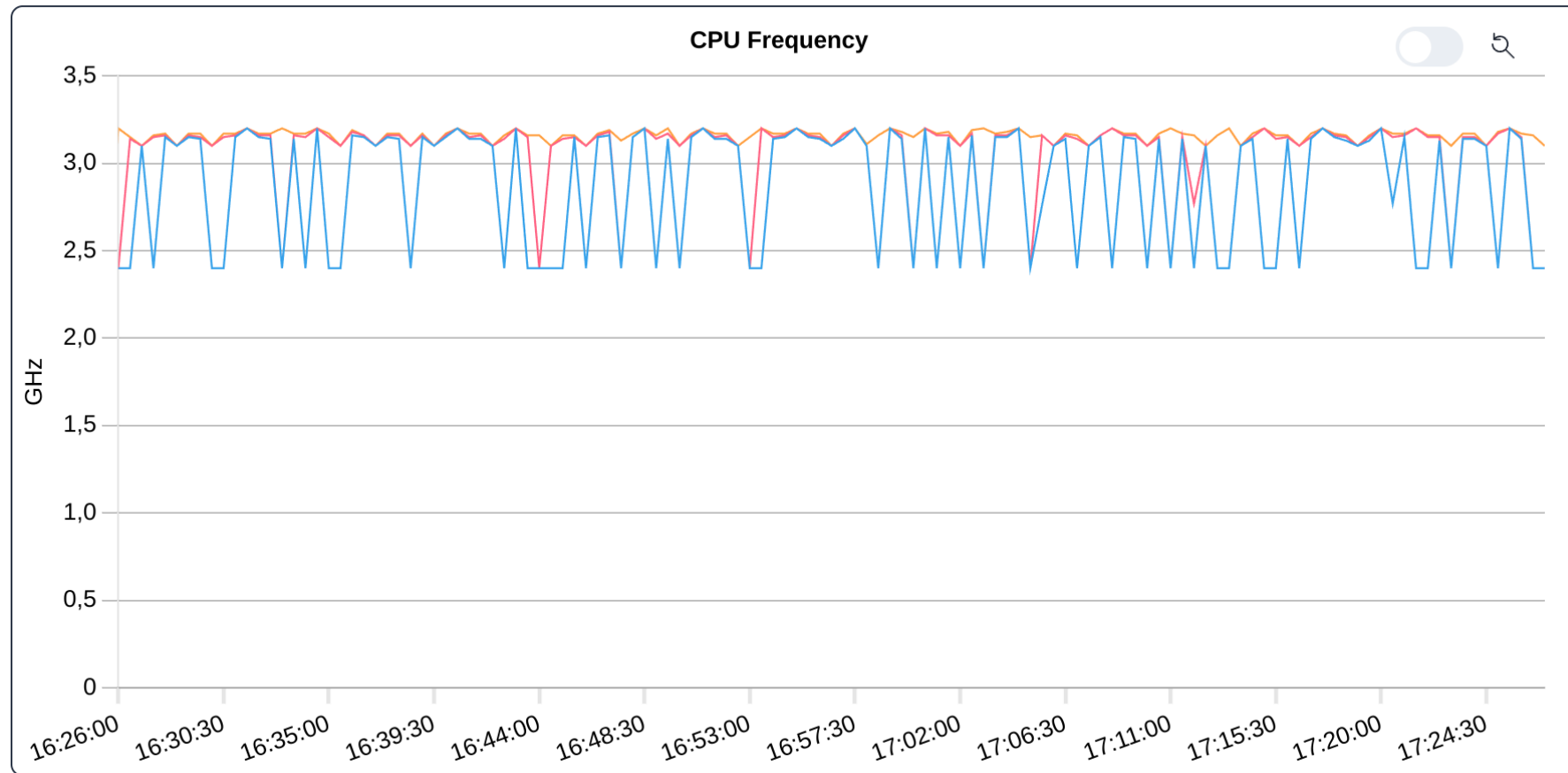
## Insight

- CPU temperatures are collected per hardware thread  
⇒ Diagram appears very cluttered
- Quantile View shows the distribution of the CPU temperatures much clearer

# Toggle: Quantile View Off - CPU Frequency



# Toggle: Quantile View On - CPU Frequency



# Quantile View - CPU Frequency

## Insight

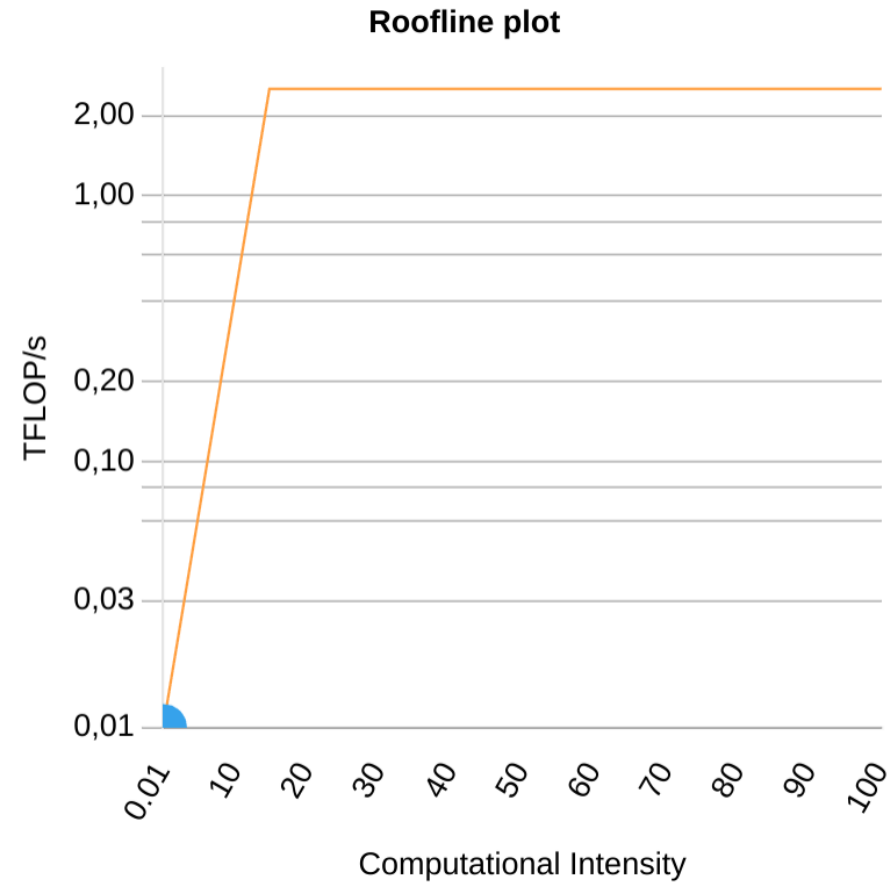
- CPU frequencies are collected per CPU core  
⇒ Diagram appears very cluttered
- Quantile View shows the distribution of the CPU frequencies much clearer

# Per Job Page - Roofline Plot

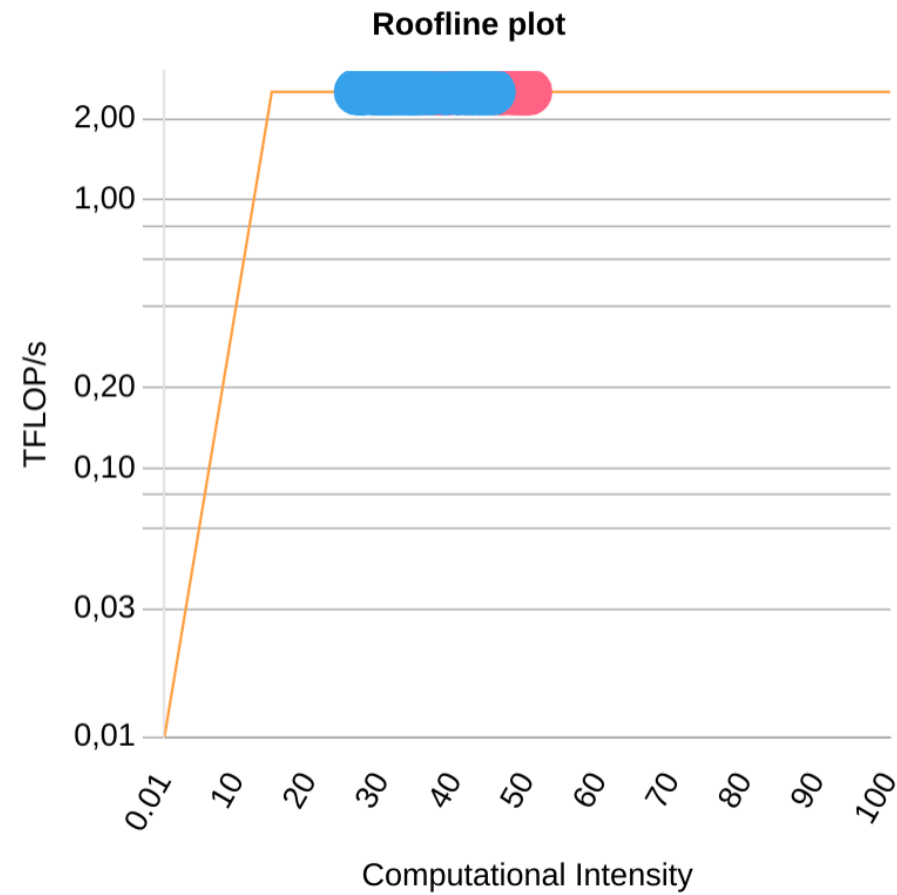
## Roofline Plot

- Diagram with:
  - y-Axis: Floating point operations per second (FLOP/s)
  - x-Axis: Computational intensity (FLOP/s / byte)
- Roofline shows two performance limiting factors:
  - For *low* computational intensity: Memory bandwidth
  - For *high* computational intensity: Processor peak performance
- Plot point:
  - For each CPU package
  - For each measurement interval

# Roofline Plot - Stream



# Roofline Plot - DGEMM



# Roofline Plot - Insight

## Insight

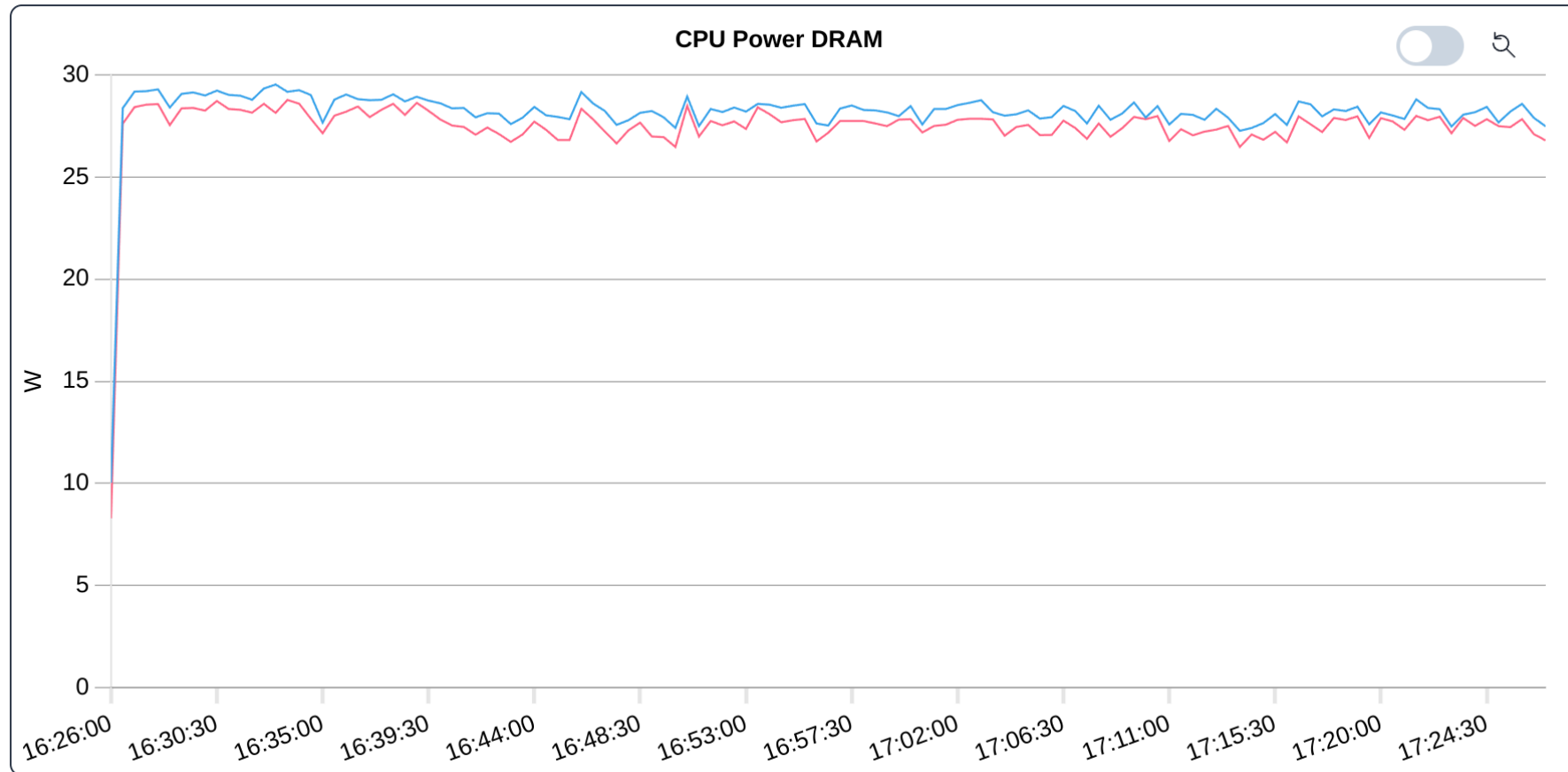
- Stream (memory bound)
  - ⇒ Low computational intensity
  - ⇒ Limiting factor: Memory bandwidth
- DGEMM (compute bound)
  - ⇒ High computational intensity
  - ⇒ Limiting factor: Processor peak performance

# Per Job Page - Performance Category Energy

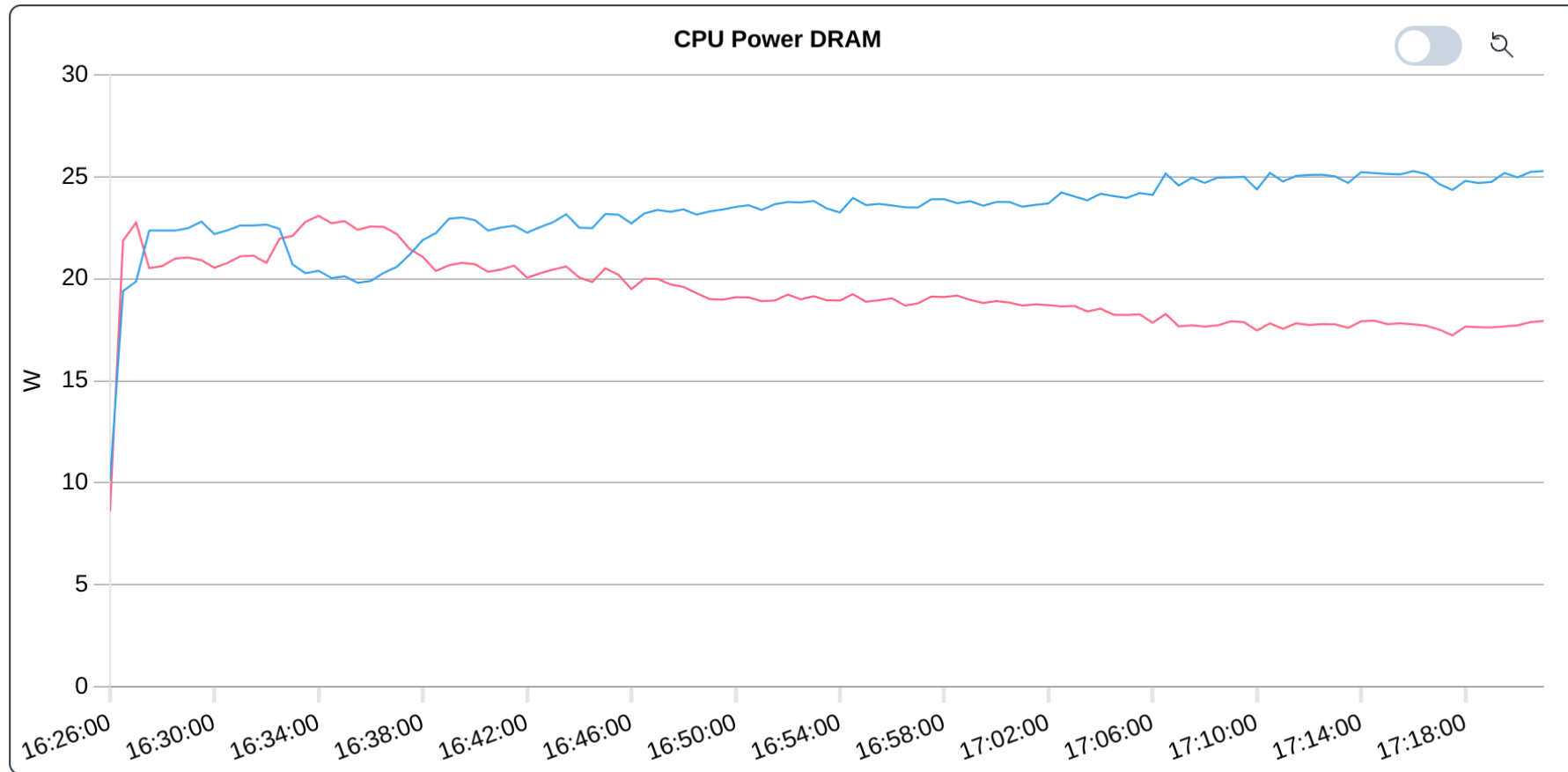
## Performance Category Energy

- CPU power consumption of DRAM channels and the package
- GPU power consumption
- Server system power consumption

# Performance Category Energy - Stream



# Performance Category Energy - DGEMM

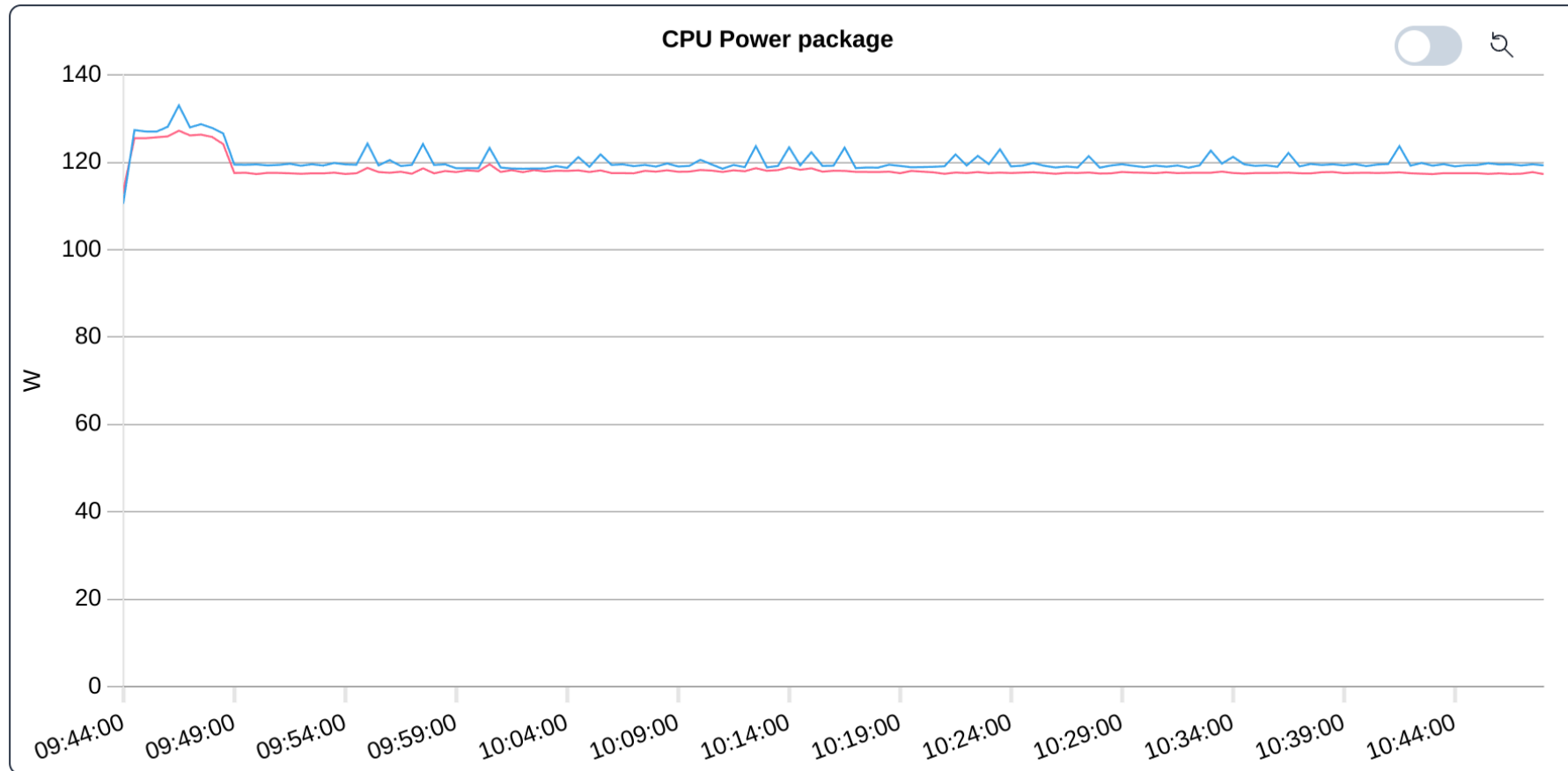


# Performance Category Energy - Insight

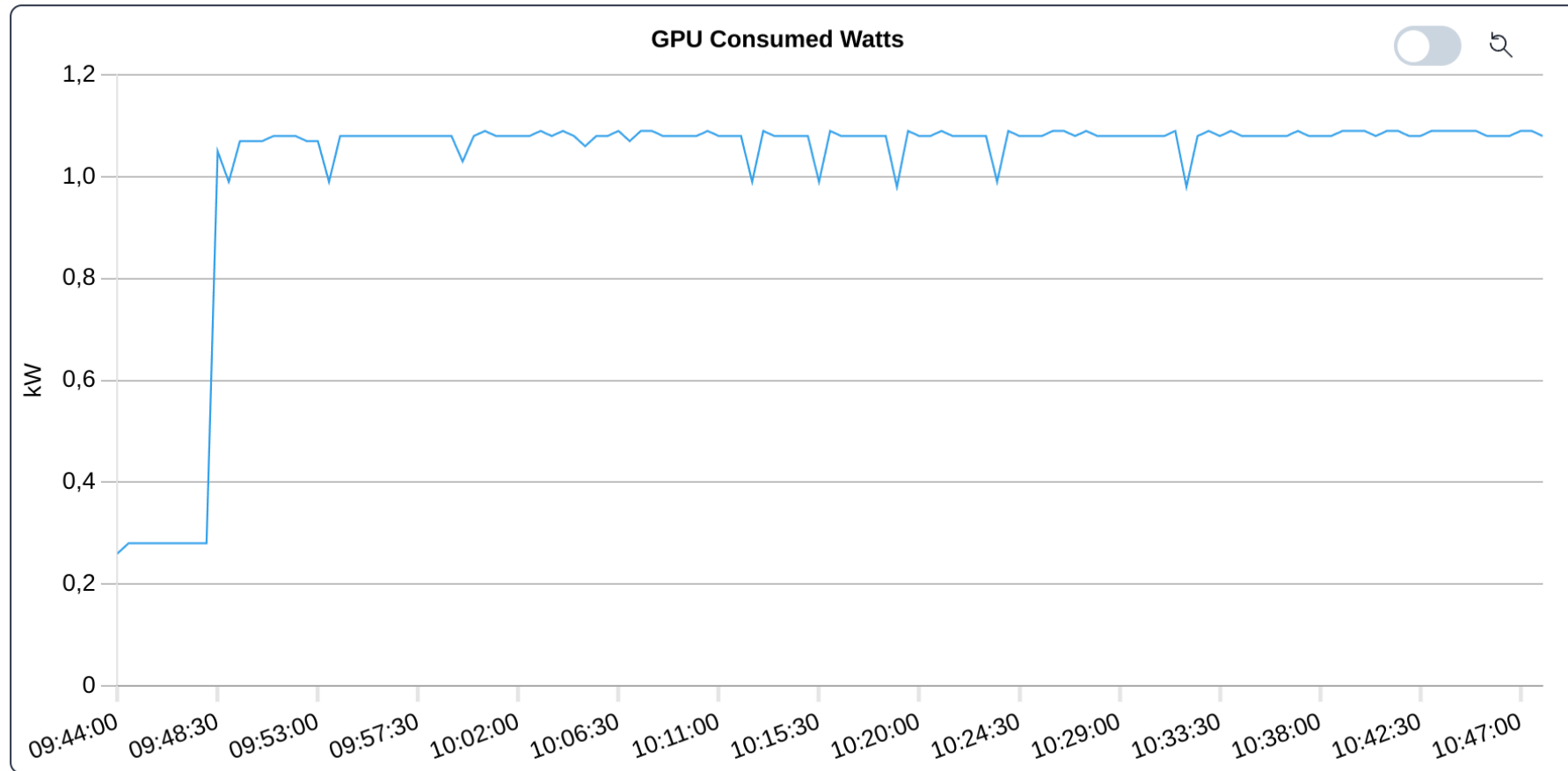
## Insight

- Stream (memory bound)
  - ⇒ constantly high pressure on the DRAM subsystem
  - ⇒ Constantly high energy consumption
- DGEMM (compute bound)
  - ⇒ Less pressure on the DRAM subsystem
  - ⇒ Varying power consumption over time

# Performance Category Energy - HPCG



# Performance Category Energy - HPCG



# Performance Category Energy - Insight

## Insight

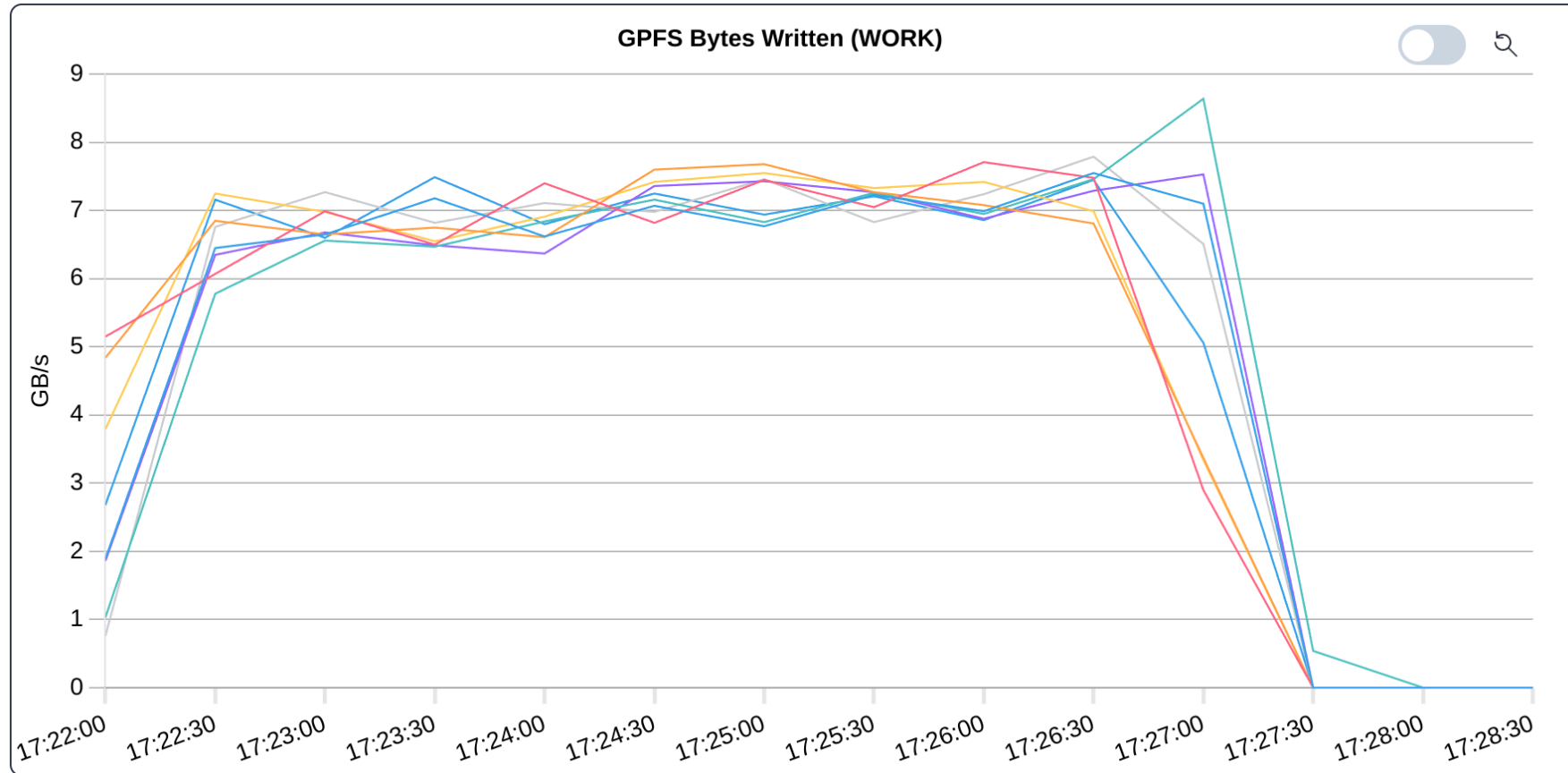
- GPU-HPCG benchmark
  - Preparation phase executed on the CPUs
  - Computation phase executed on the GPUs
- CPU package power consumption: Higher in preparation phase than in the compute phase
- GPU power consumption: Higher in compute phase than in preparation phase

# Per Job Page - Performance Category Filesystem

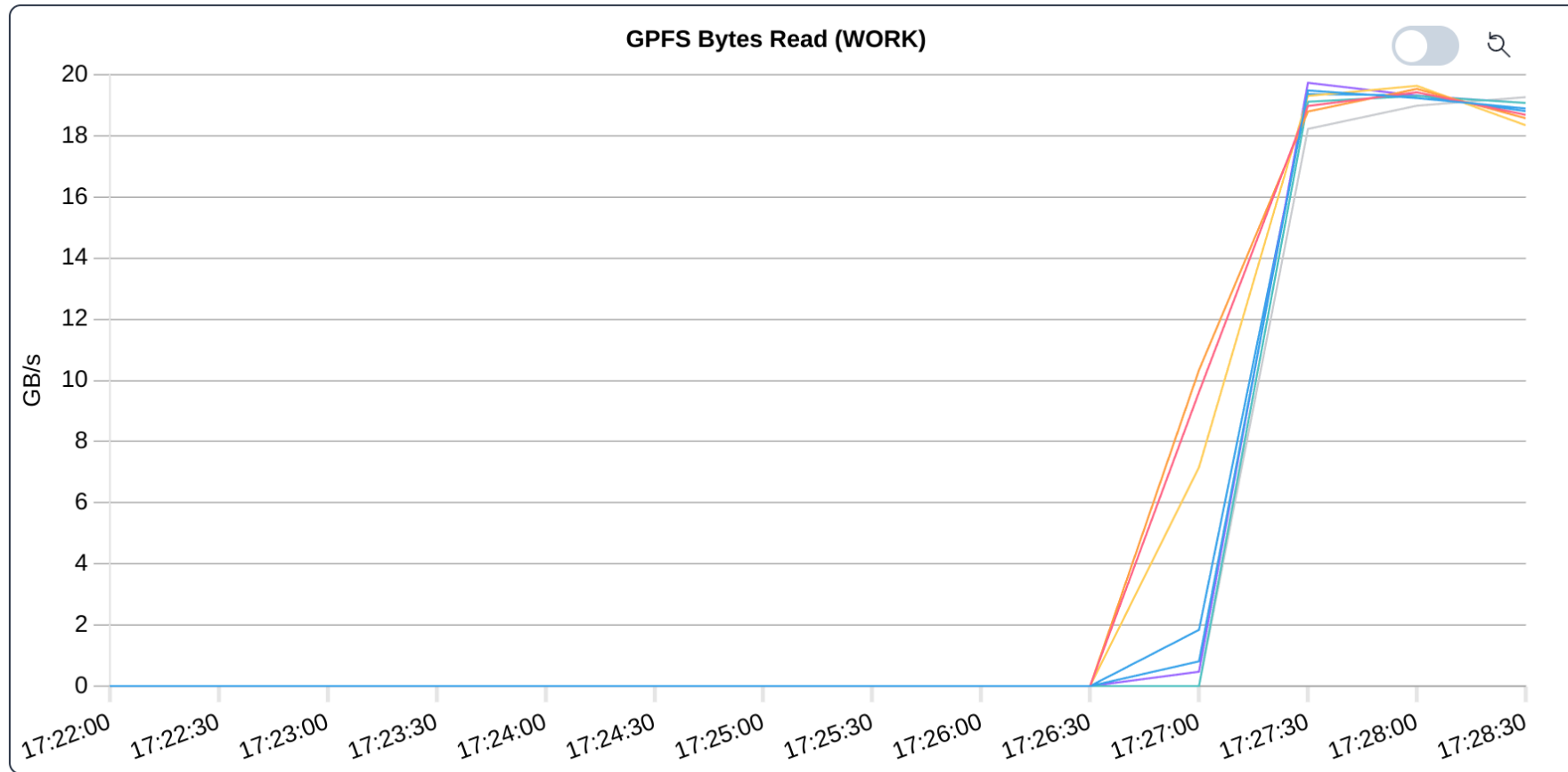
## Performance Category Filesystem

- Meta data operation
  - Number of file open / closes
  - Number of directory reads
  - Number of inode updates
  - ...
- IO throughput
  - Bytes read / written
  - Number of read / writes

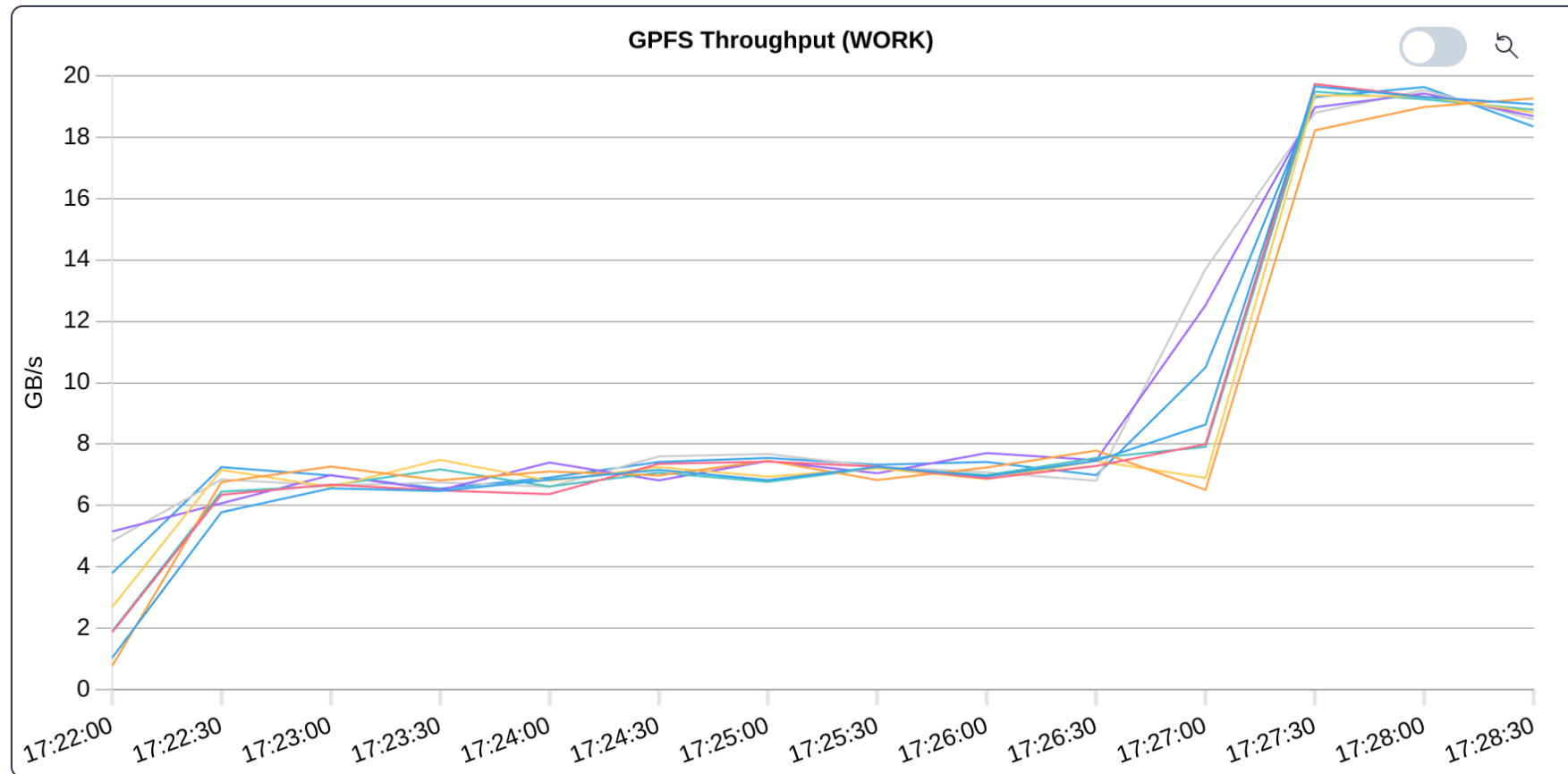
# Performance Category Filesystem - IOR



# Performance Category Filesystem - IOR



# Performance Category Filesystem - IOR



# Performance Category Filesystem - Insight

## Insight

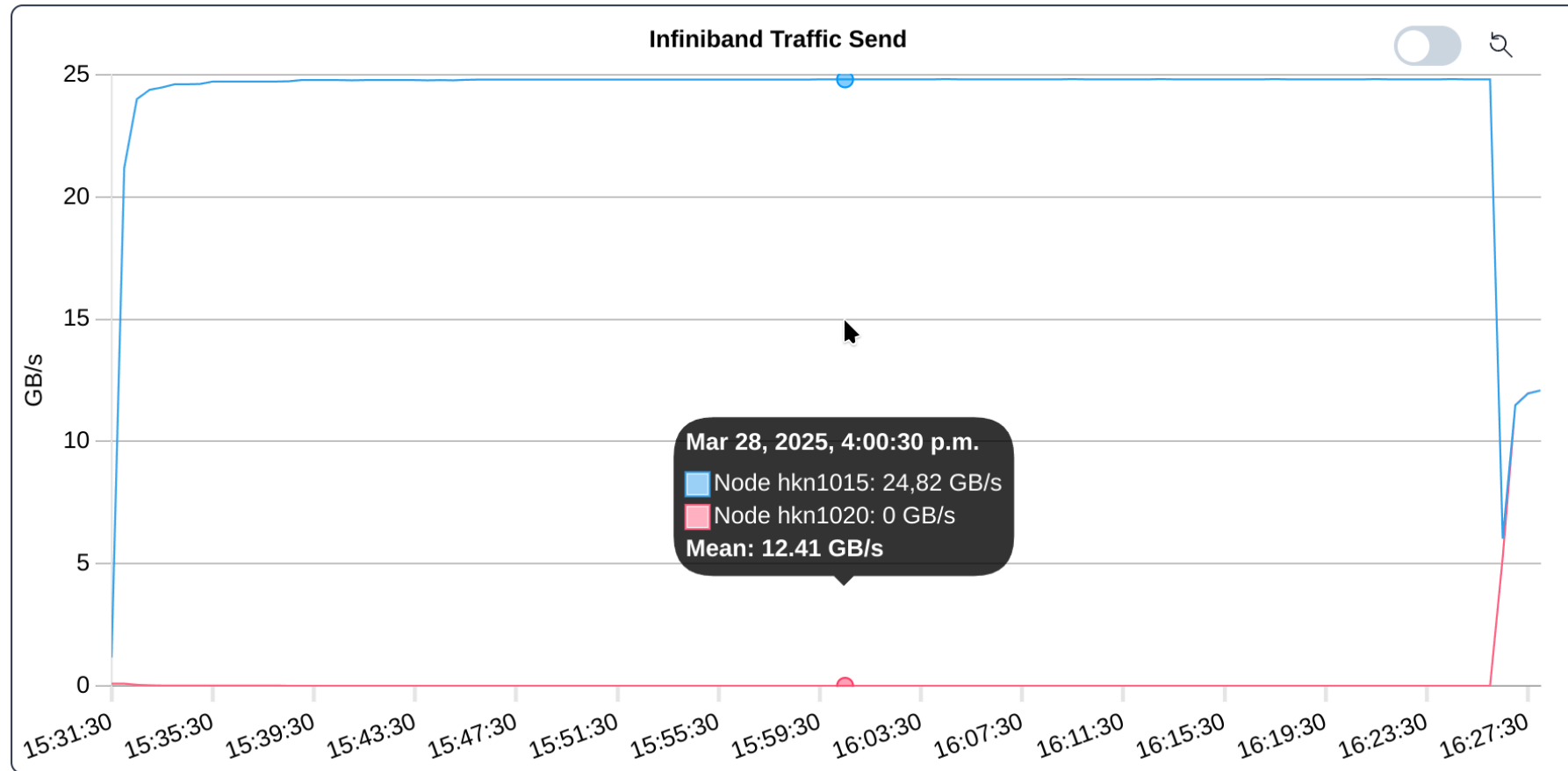
- Parallel IO: Performed from multiple nodes
- Two phases:
  - Phase 1: Files are written
  - Phase 2: Files are read
- Read throughput is higher than write throughput

# Per Job Page - Performance Category Interconnect

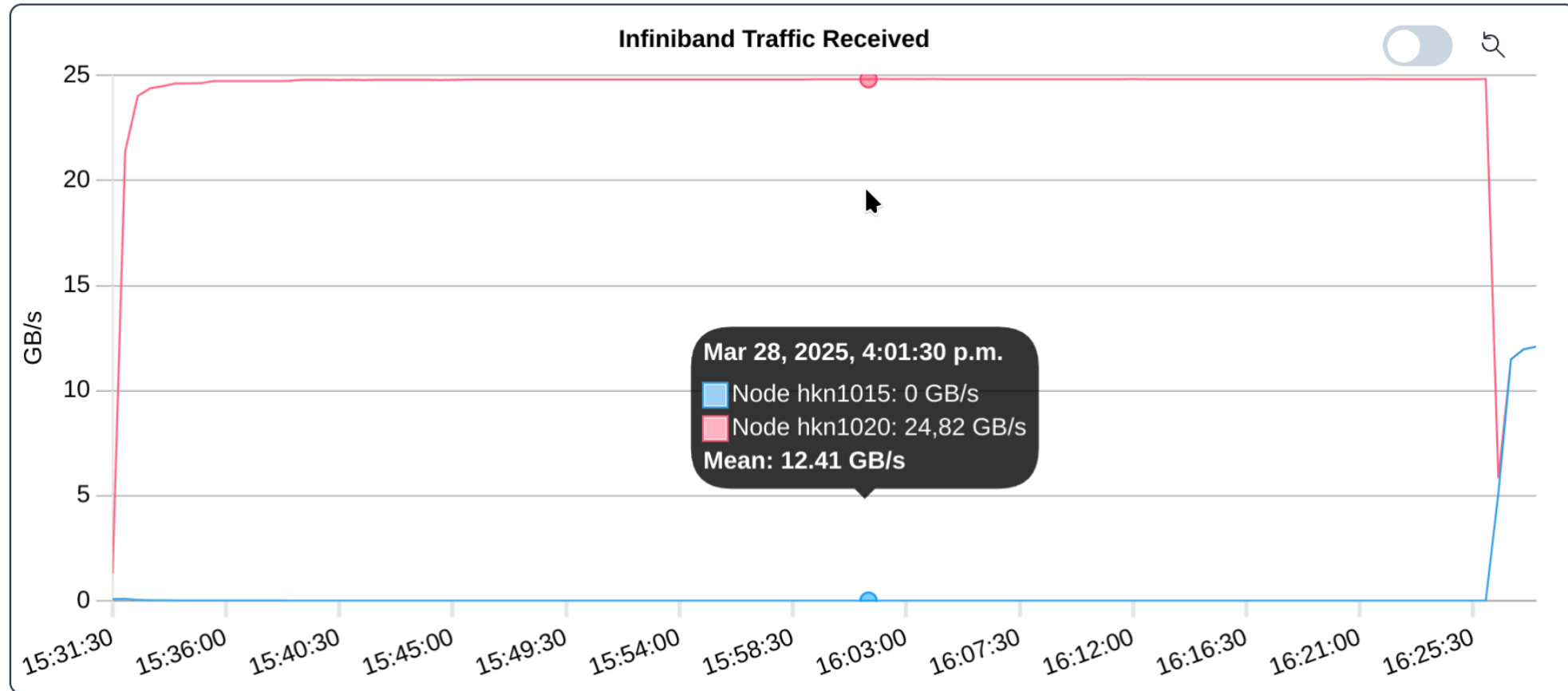
## Performance Category Interconnect

- InfiniBand bandwidth
  - Sent
  - Received
  - Aggregated sent and received
- InfiniBand number of packages
  - Sent
  - Received
  - Aggregated sent and received

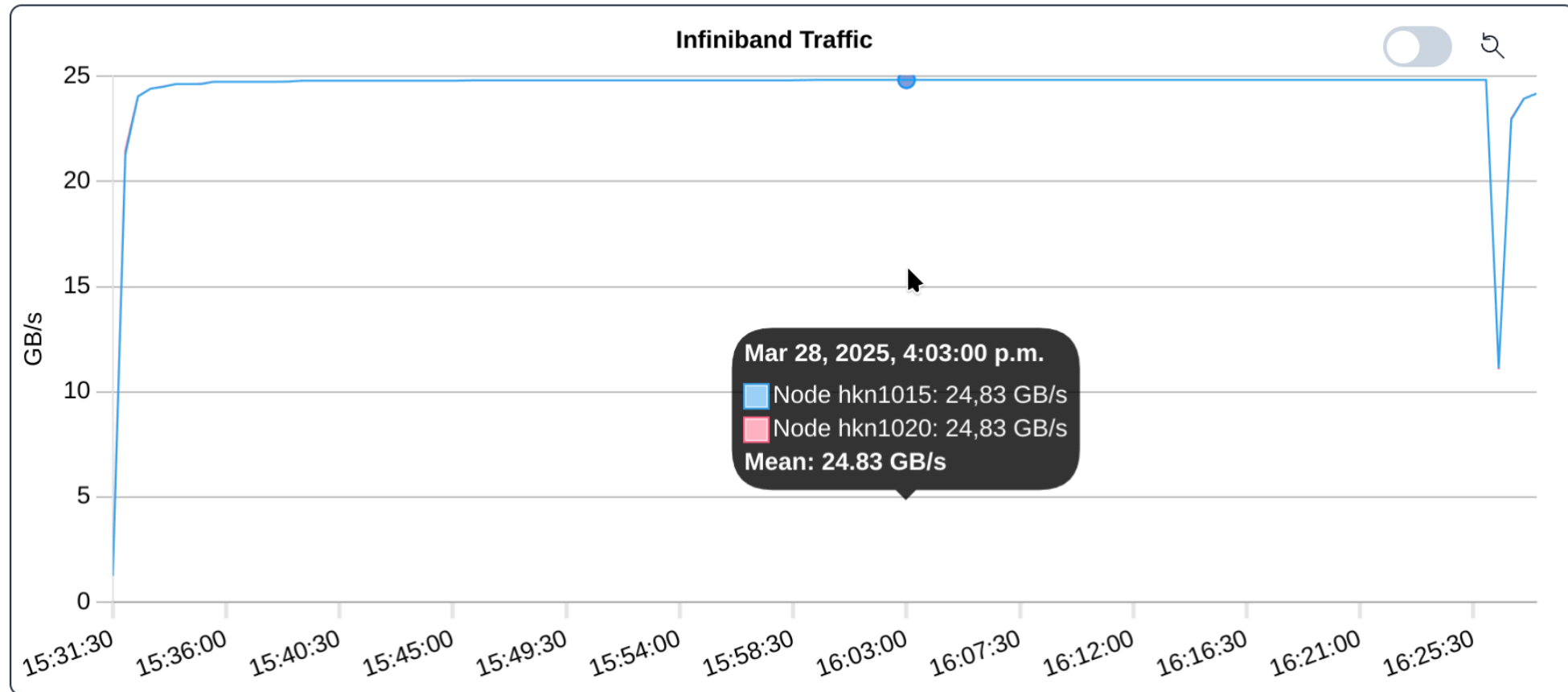
# Performance Category Interconnect - OMB



# Performance Category Interconnect - OMB



# Performance Category Interconnect - OMB



# Performance Category Interconnect - Insight

## Insight

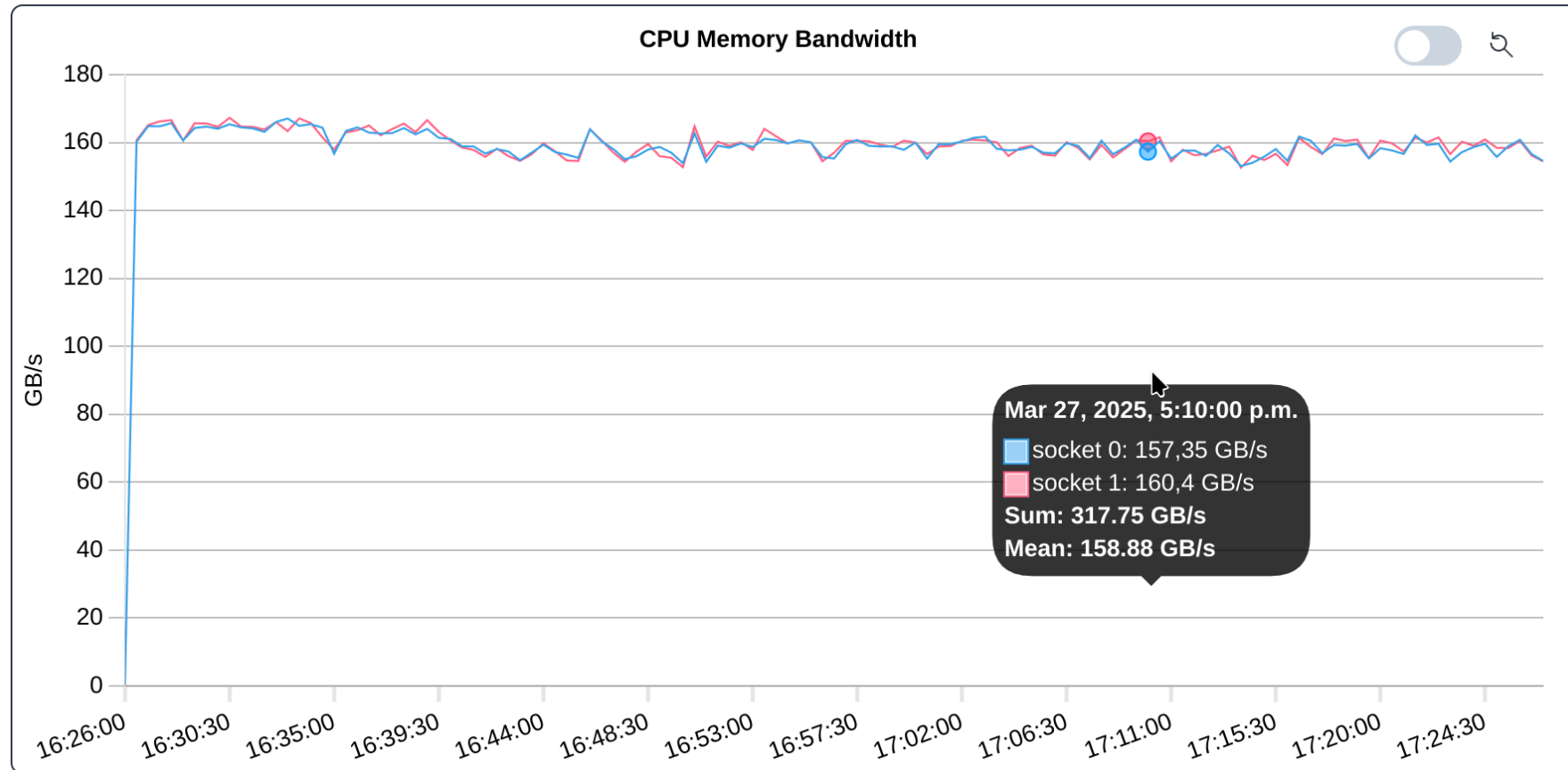
- OSU Micro-Benchmark performs MPI point to point communication:
  - Node hkn1015 only sends data (receive bandwidth is zero)
  - Node hkn1020 only receives data (send bandwidth is zero)

# Per Job Page - Performance Category Memory

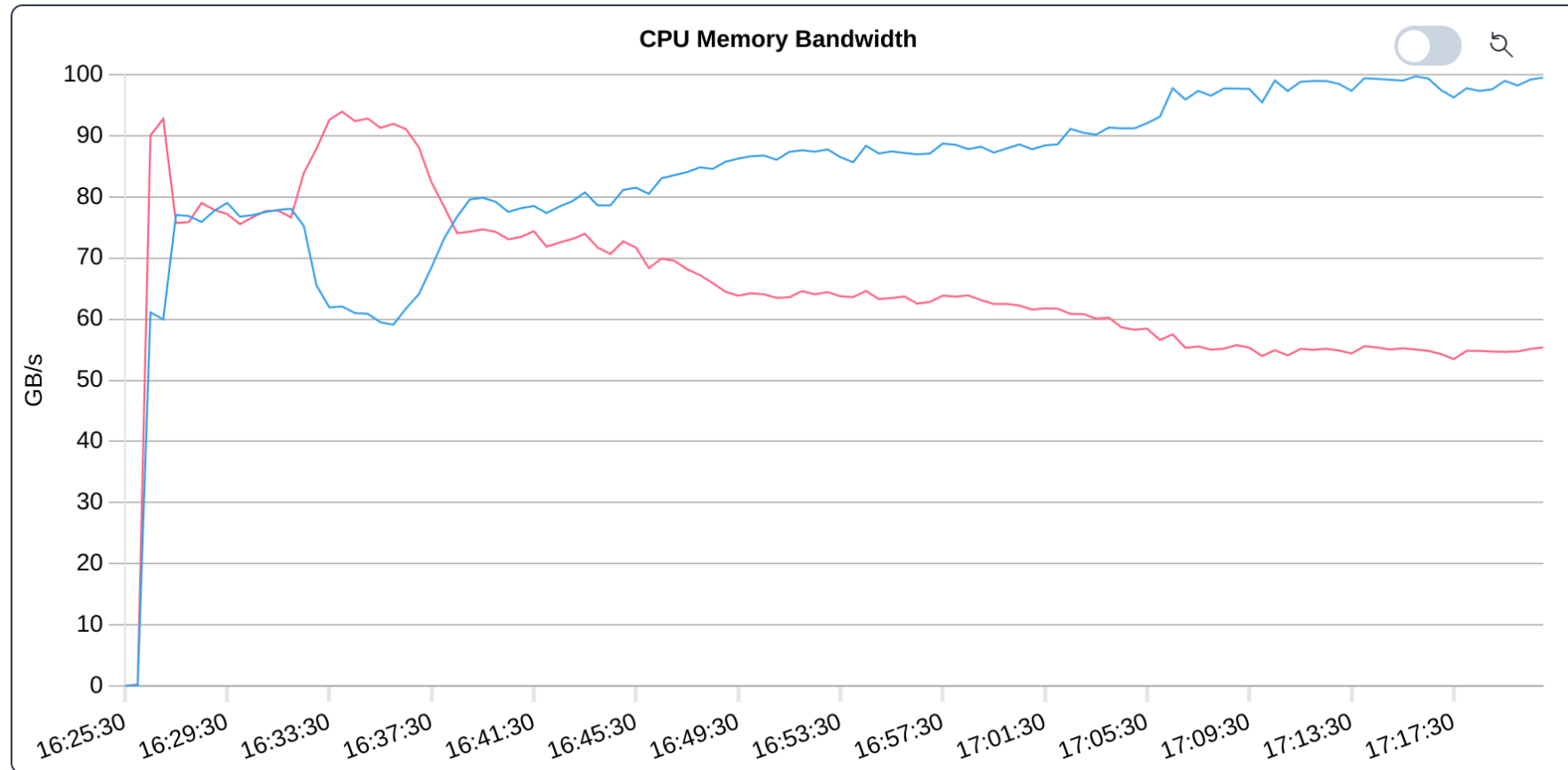
## Performance Category Memory

- Amount of memory used
  - On the system
  - On the GPU
- CPU memory bandwidth
- GPU memory
  - Utilization (in %)
  - Frequency

# Performance Category Memory - Stream



# Performance Category Memory - DGEMM

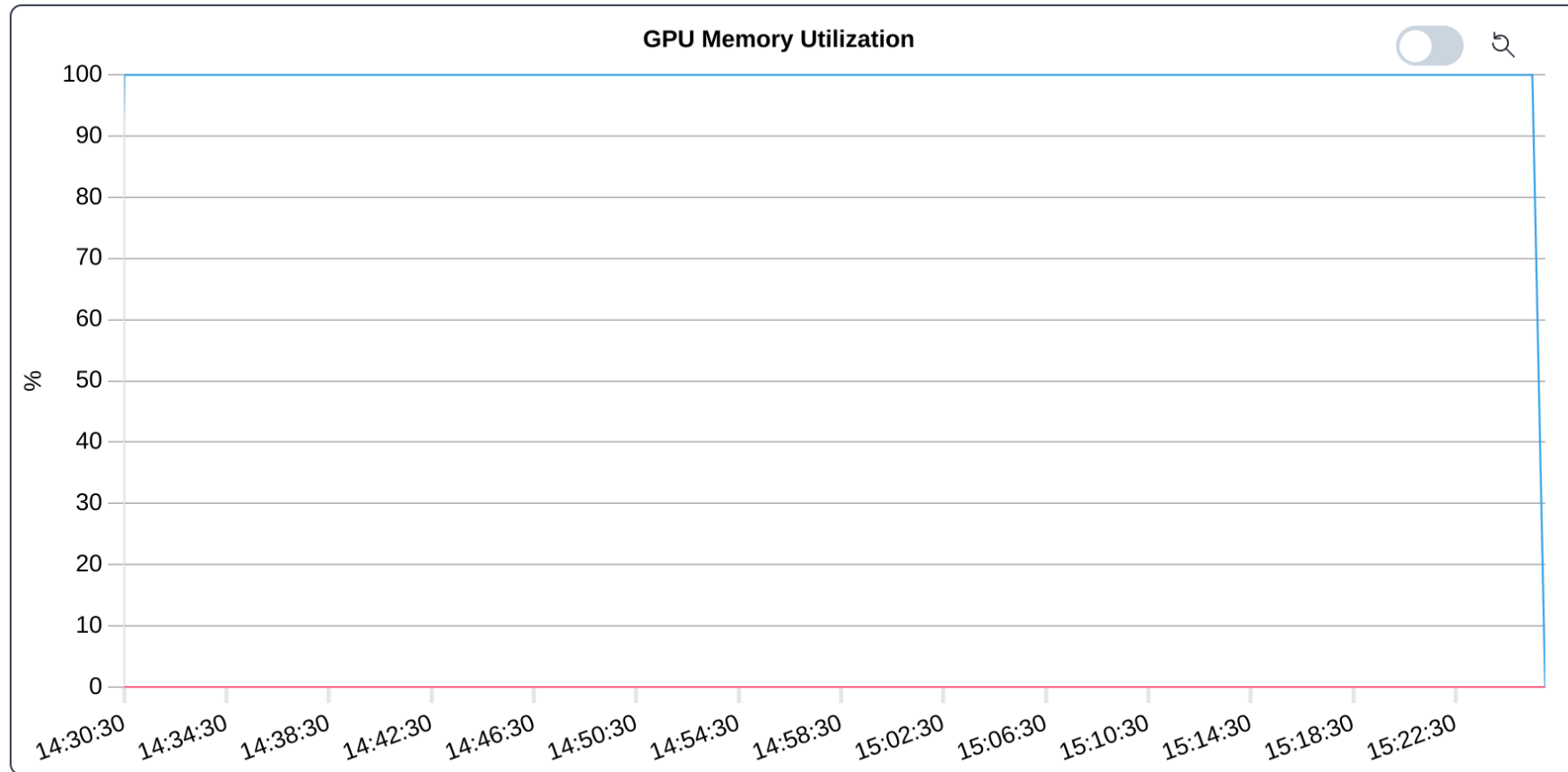


# Performance Category Memory - Insight

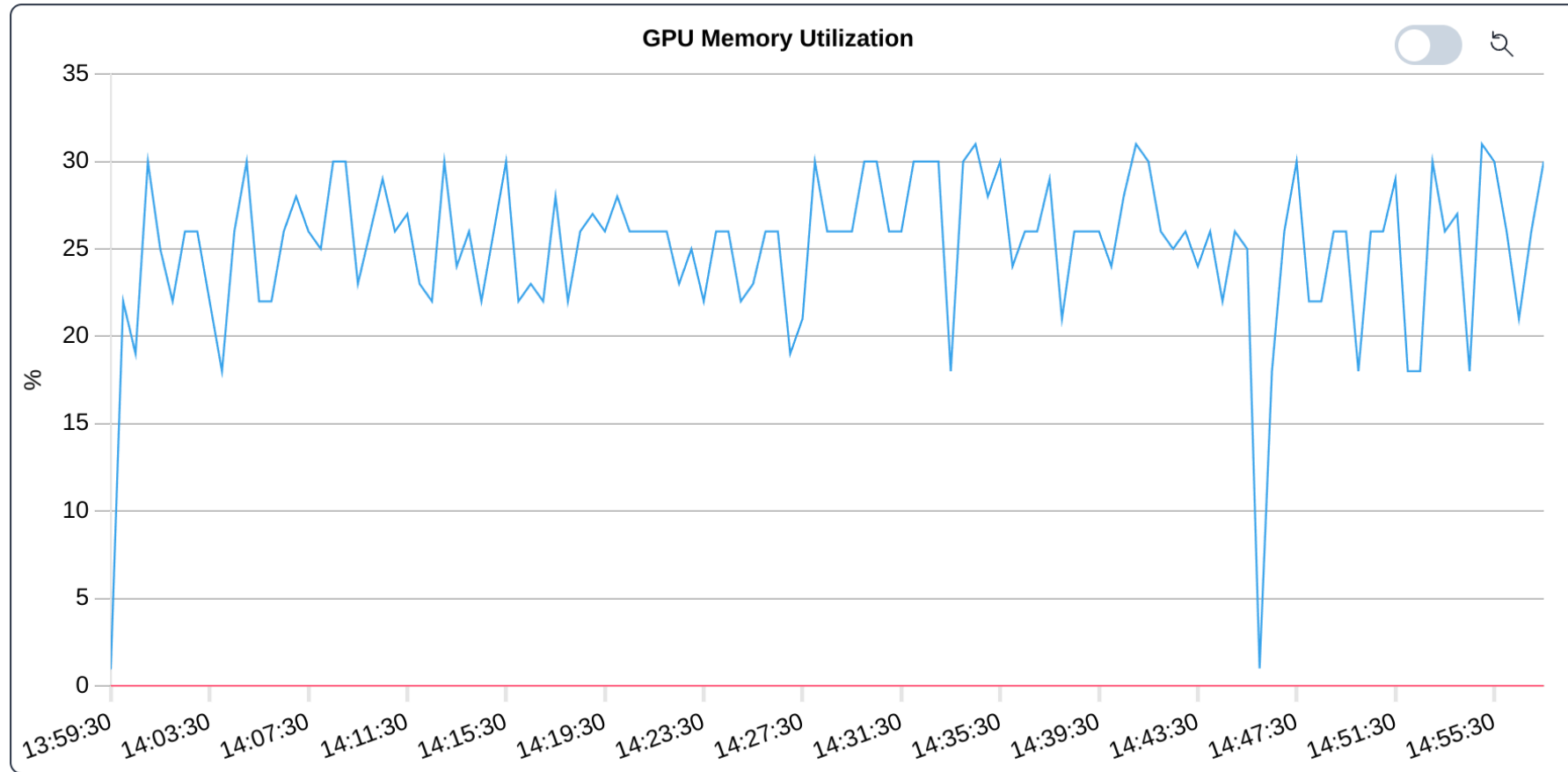
## Insight

- Stream (memory bound):
  - ⇒ Constantly high pressure on the memory subsystem
- DGEMM (compute bound):
  - ⇒ Less pressure on the memory subsystem
  - ⇒ Varying bandwidth over time

# Performance Category Memory - BabelStream



# Performance Category Memory - DGEMM



# Performance Category Memory - Insight

## Insight

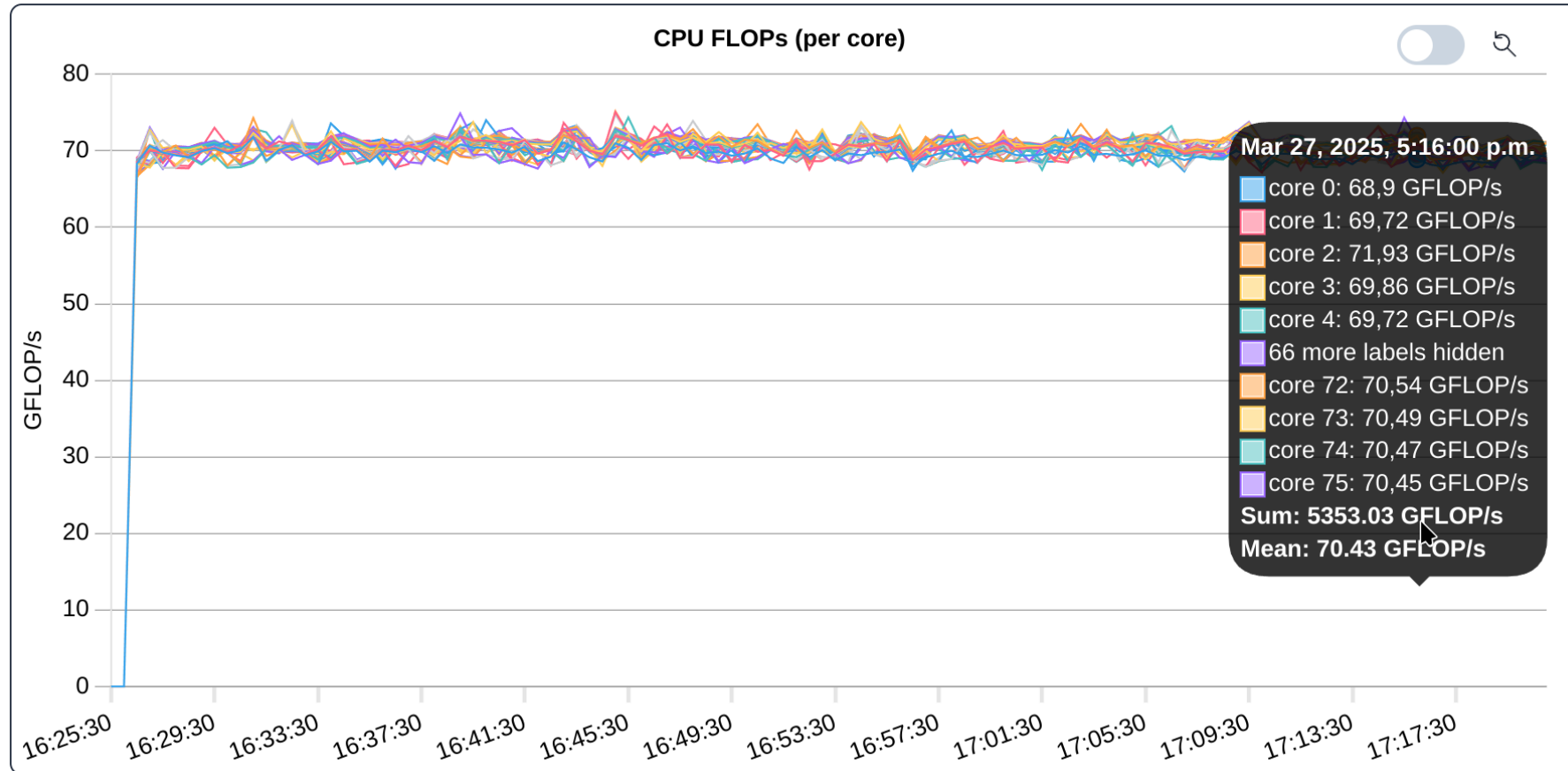
- BabelStream (memory bound):
  - ⇒ Fully utilizes the memory subsystem of the GPU
- GPU-DGEMM (compute bound):
  - ⇒ Less pressure on the GPU memory subsystem
  - ⇒ Varying utilization over time.

# Per Job Page - Performance Category Performance

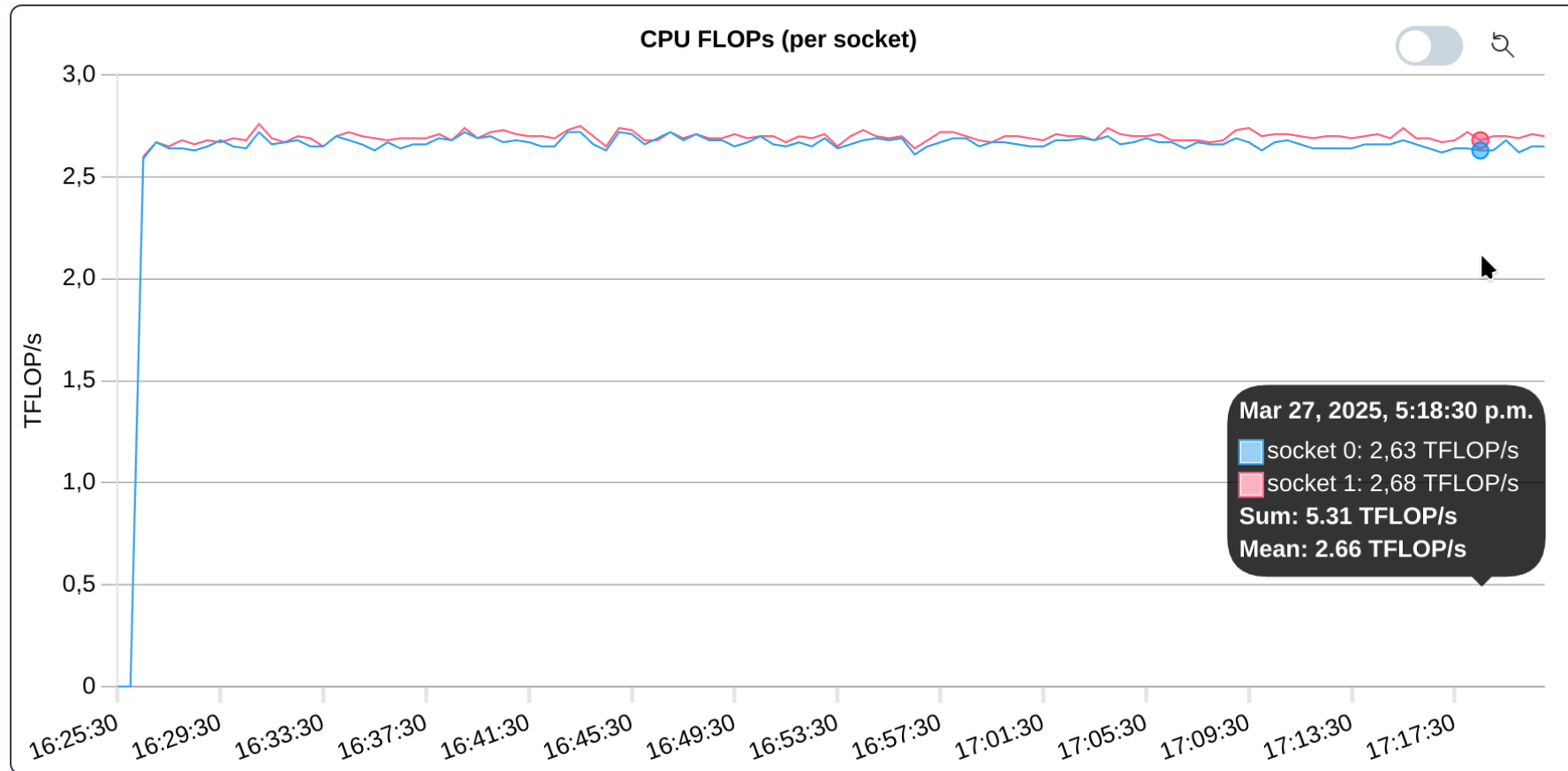
## Performance Category Performance

- Floating point operation per second (FLOP/s), collected per *hardware thread*, aggregated per *core* or per *socket*
- Instructions per cycle (IPC), collected per *hardware thread*, aggregated per *core* or per *socket*
- CPU time spend in kernel and in user space
- One minute Linux load average
- GPU utilization
- CPU and GPU frequency

# Performance Category Performance - DGEMM



# Performance Category Performance - DGEMM



# Performance Category Performance - Insight

## Insight

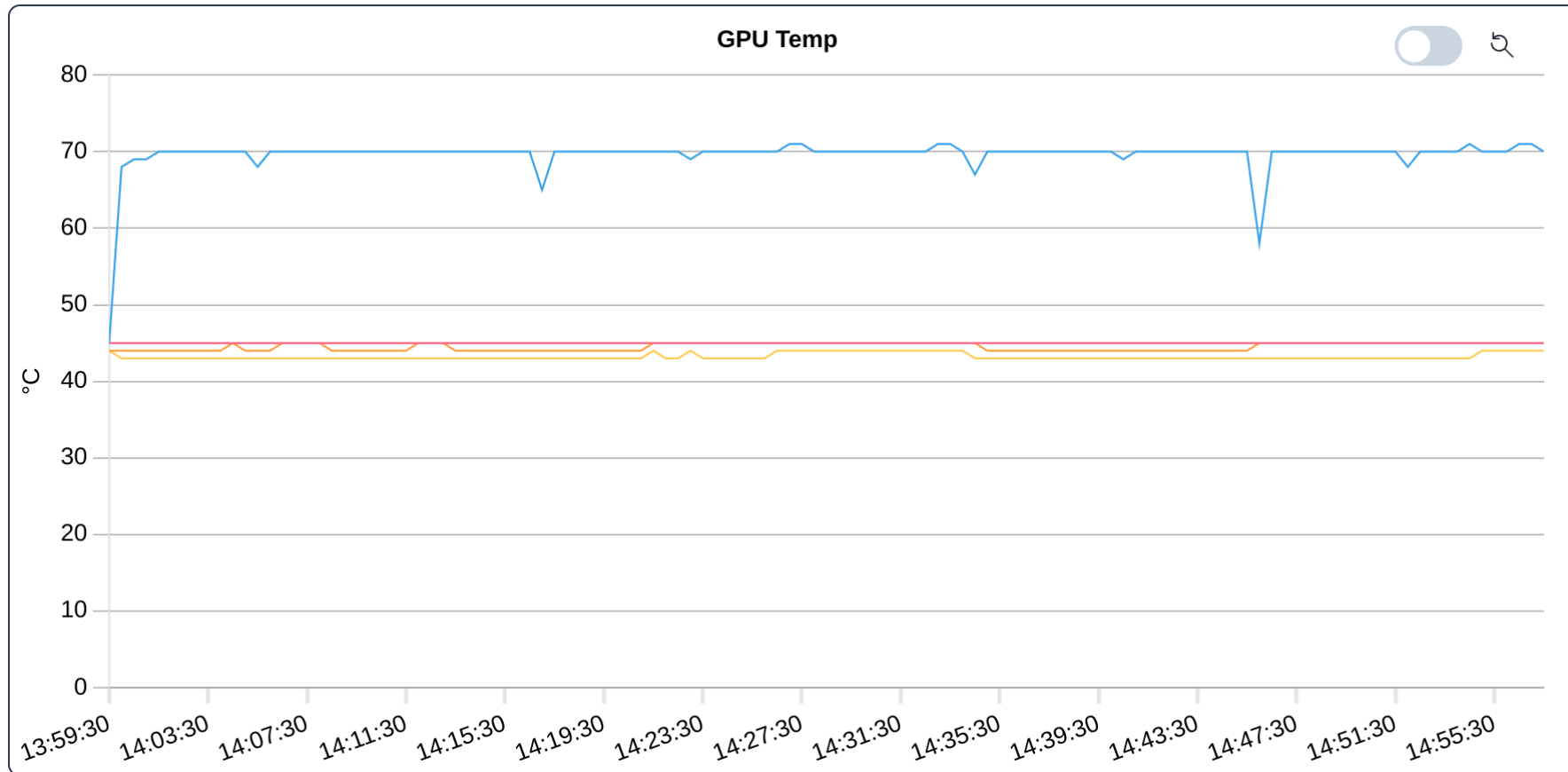
- Floating point operation per second (FLOP/s) are collected per hardware thread
- Examine even utilization of cores
  - ⇒ Aggregate per core
- Examine even utilization of CPU sockets
  - ⇒ Aggregate per socket
- Summed FLOP/s is the same in both diagrams

# Per Job Page - Performance Category Temperature

## Performance Category Temperature

- CPU
- GPU

# Performance Category Temperature - DGEMM

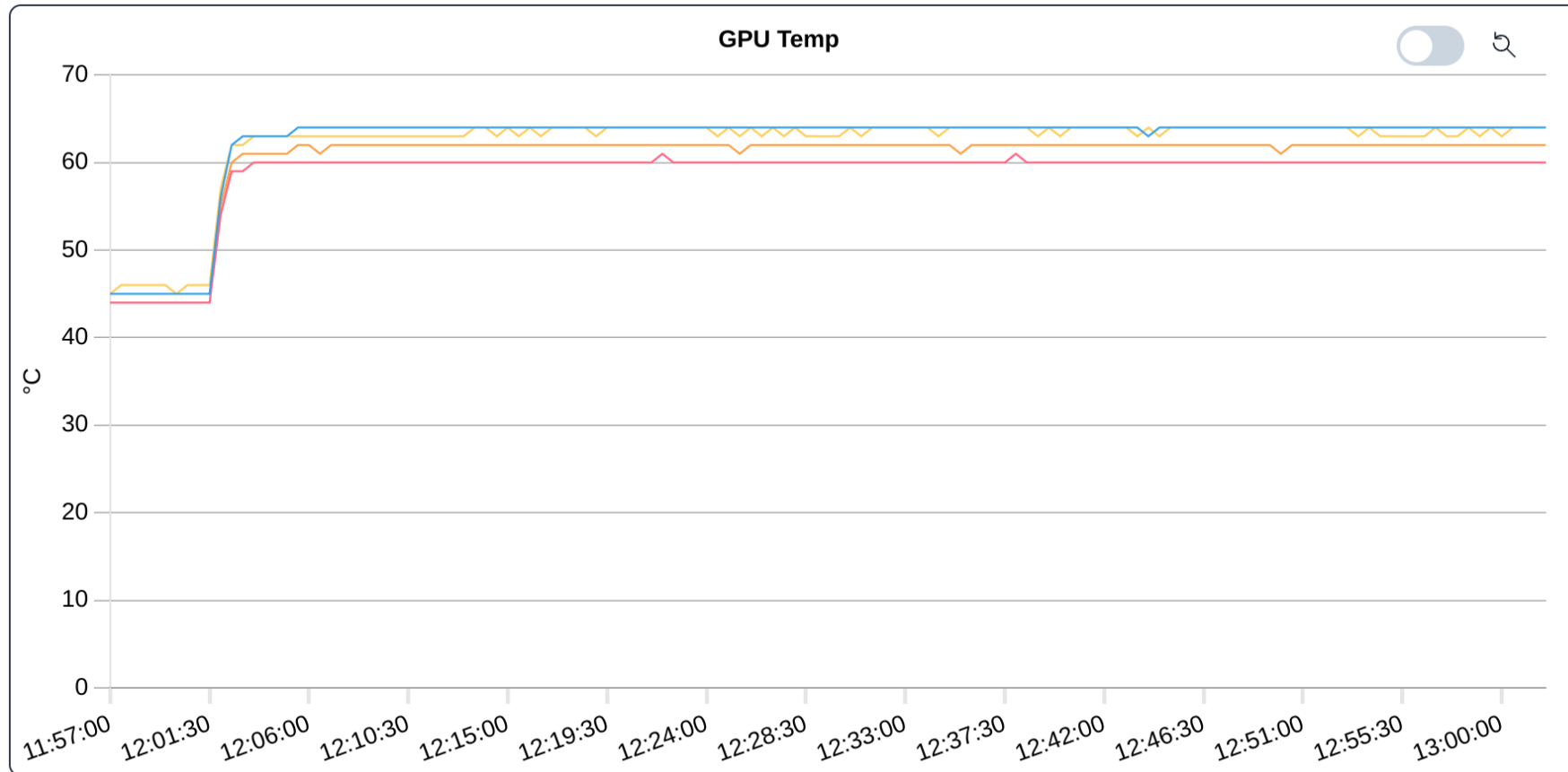


# Performance Category Temperature - Insight

## Insight

- GPU-DGEMM only utilizes one GPU
- Only this GPU gets hot
- Other GPUs maintain lower temperatures

# Performance Category Temperature - HPCG



# Performance Category Temperature - Insight

## Insight

- GPU-HPCG
  - Preparation phase executed on the CPUs
    - ⇒ GPUs not utilized
    - ⇒ Low temperature
  - Computation phase executed on the GPUs
    - ⇒ Higher temperature

# Per Job Page - Additional Features

## Additional Features

- For multi-node jobs there is a configuration option to select the per-node aggregation function used (e.g. average, sum, maximum)
- Live view of running jobs
- Download CSV file
  - All metrics
  - Use in spread sheet application or Python
- Outlook
  - Availability for Cluster uc3
  - Automatic job analyzer
    - ⇒ Assign tags for detected characteristics
  - Job sharing (e.g. with your work group)