German Research School
for Simulation Sciences

Uncovering degraded application performance with LWM^2

Aamer Shah, Chih-Song Kuo, Lucas Theisen, Felix Wolf
November 17, 2014

JÜLICH
FORSCHUNGSZENTRUM

RWTH AACHEN
UNIVERSITY

German Research School for Simulation Sciences

Motivation: Performance degradation

- Internal factors:
  - Inefficient use of hardware resources
  - Uneven work load distribution
  - Inefficient communication pattern
  - Etc.
- External factors:
  - Operating system jitter
  - Network interference from other applications
  - I/O interference from other applications
  - Inefficient process-to-compute-node mapping
  - I/O subsystem anomalous behavior
  - Etc.

German Research School for Simulation Sciences

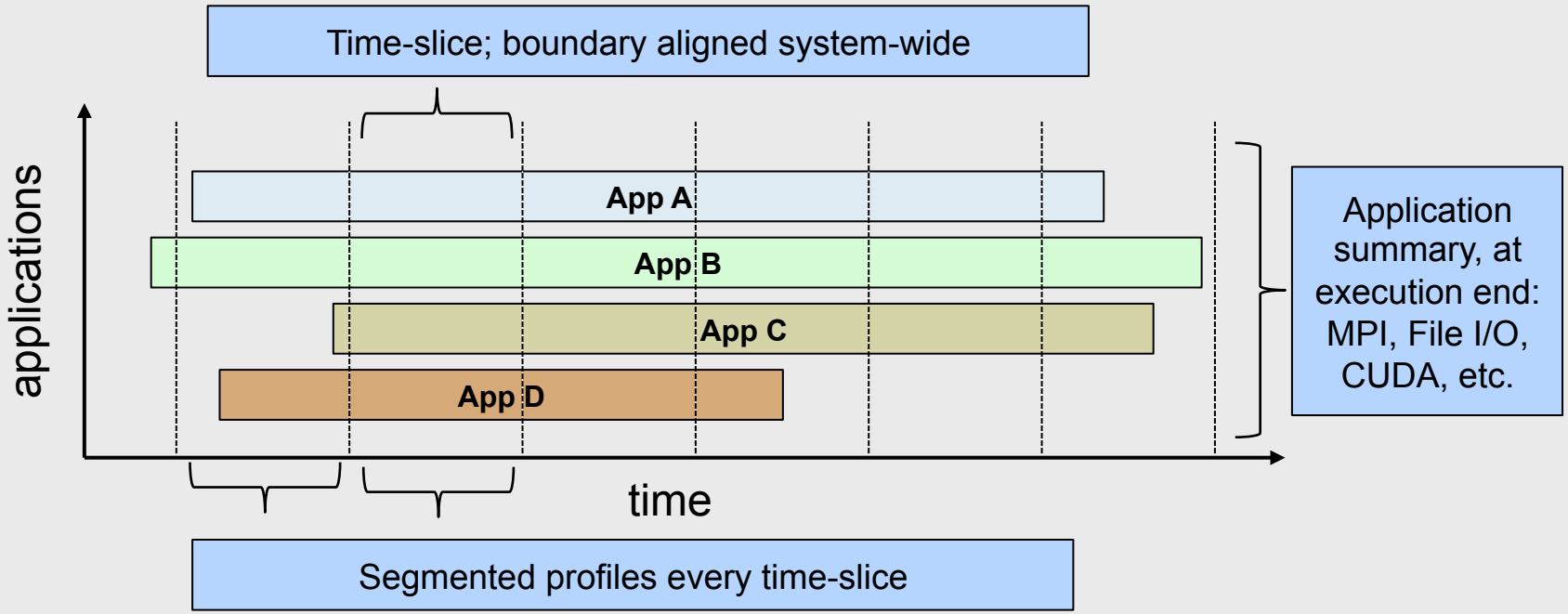
LWM^2: Introduction

- LWM^2: Light-Weight Monitoring Module
  - Lightweight profiler
  - Supports: MPI, File I/O, OpenMP and CUDA
- Easy to use
  - No code recompilation or relinking
  - Uses library preloading to profile application
- Compact output
  - Application performance summary on console
  - Generates output files with more detailed information
  - Command line utility available to read the output files

➢ Main objective is to identify performance degradation from external sources by monitoring system resources

Time-slices

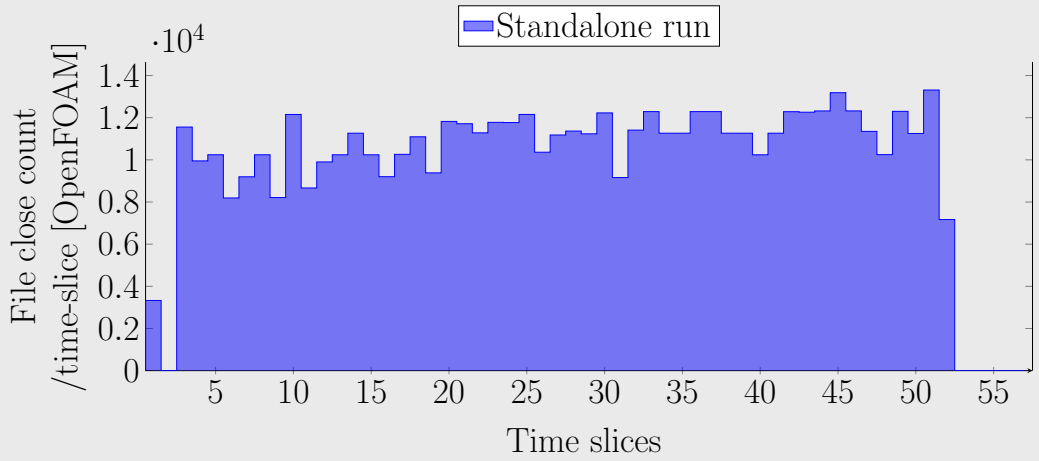
- LWM^2 also generates segmented profiles at fixed time intervals, called time-slices
  - Time-slice boundaries are synchronized system-wide



German Research School for Simulation Sciences

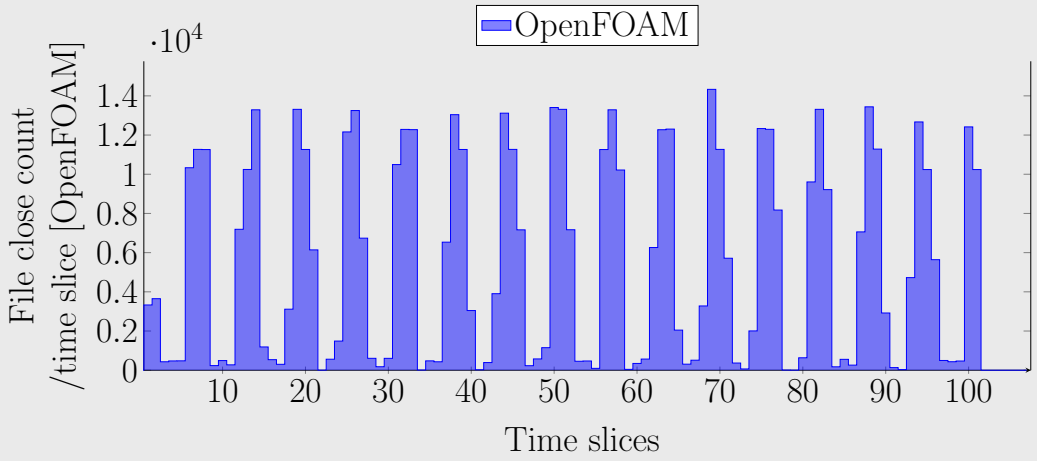
Inter-application interference

- Time-slices allow comparing of performance across applications
  - Can identify cases of inter-application interference
  - OpenFOAM: creates large number of checkpoint files during execution
  - Executed alone and against a periodic file-write-benchmark



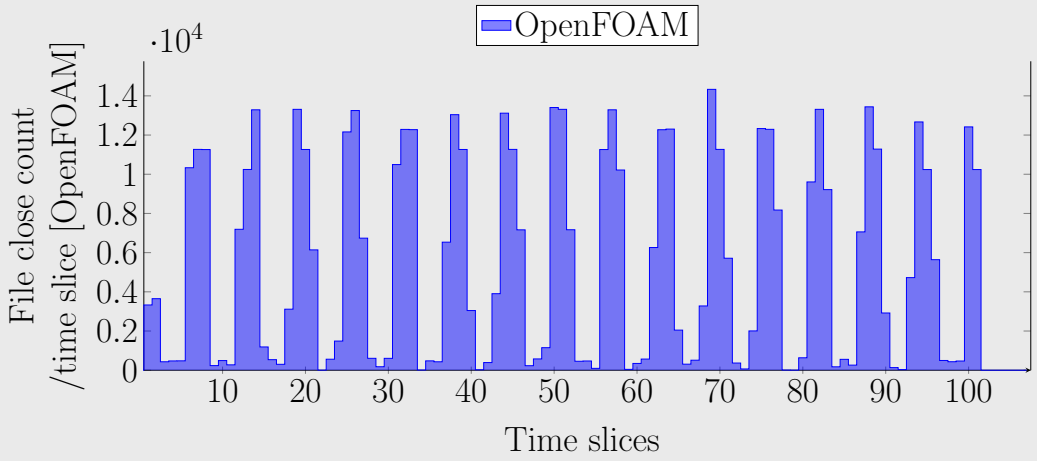
German Research School
for Simulation Sciences

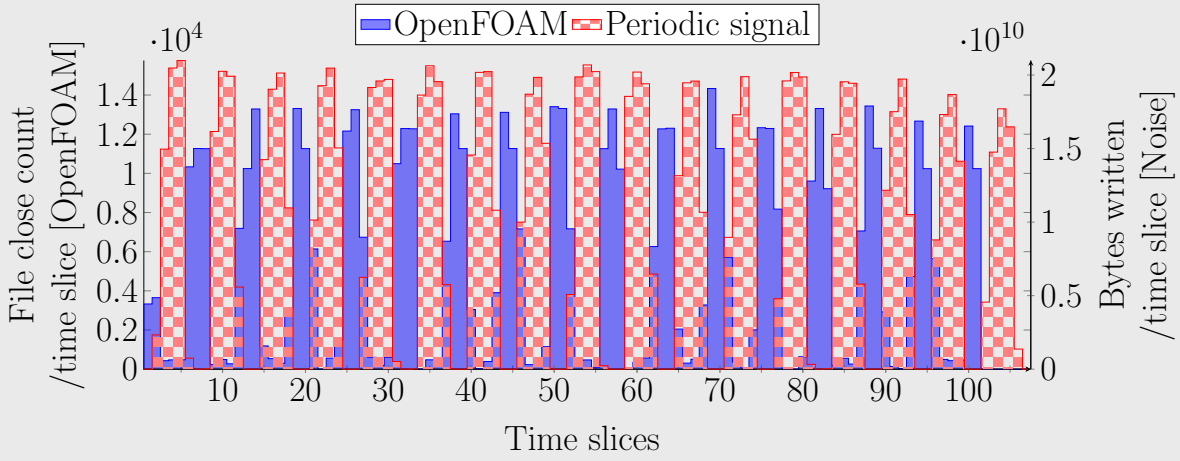
Inter-application interference



German Research School for Simulation Sciences

Inter-application interference

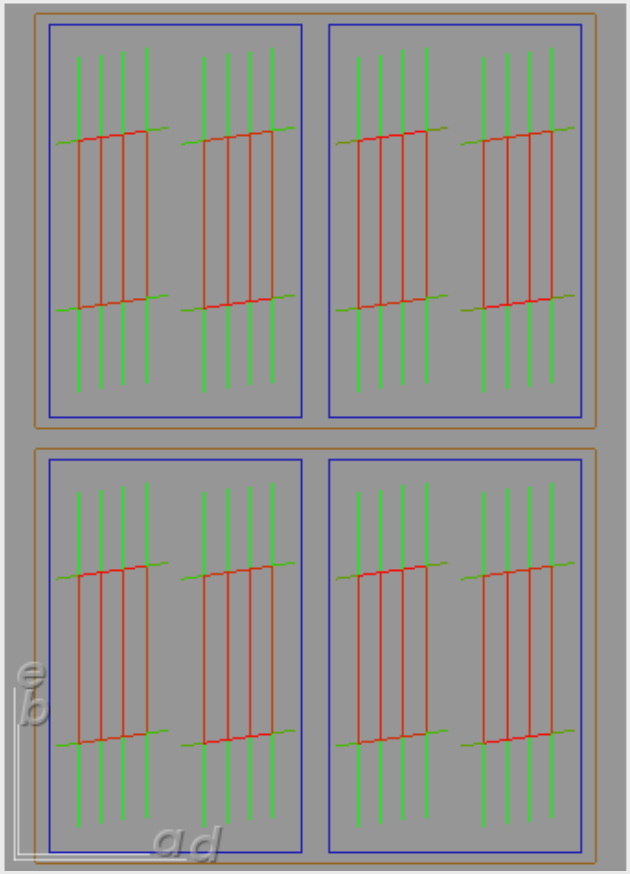




German Research School for Simulation Sciences

Network monitoring on BG/Q

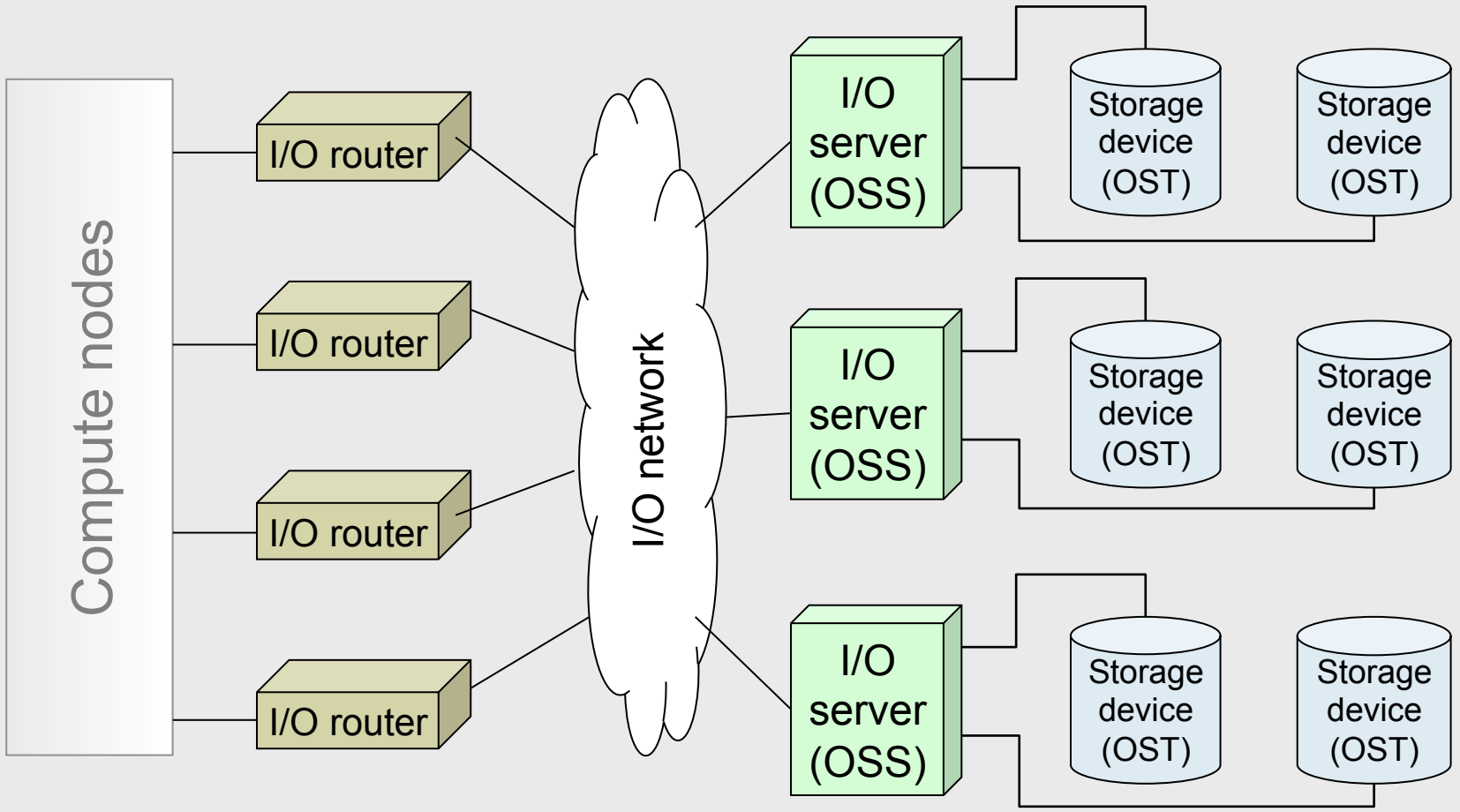
- Each compute node on BG/Q system has 11 network links
  - 2 x 5D for communication
  - 1 for I/O
- For each link, LWM^2 captures
  - Link traffic: number of 32 bytes packet sent
  - Node contention: packet arrival rate, average queue length
- Provide a separate tool (VisTorus) to visualize the network traffic[1]
- Identify hot links and bottlenecks



[1] Will be presented in VPA'14 workshop on Friday (Nov 21)

German Research School for Simulation Sciences

I/O subsystem structure



German Research School for Simulation Sciences

Enhanced I/O monitoring

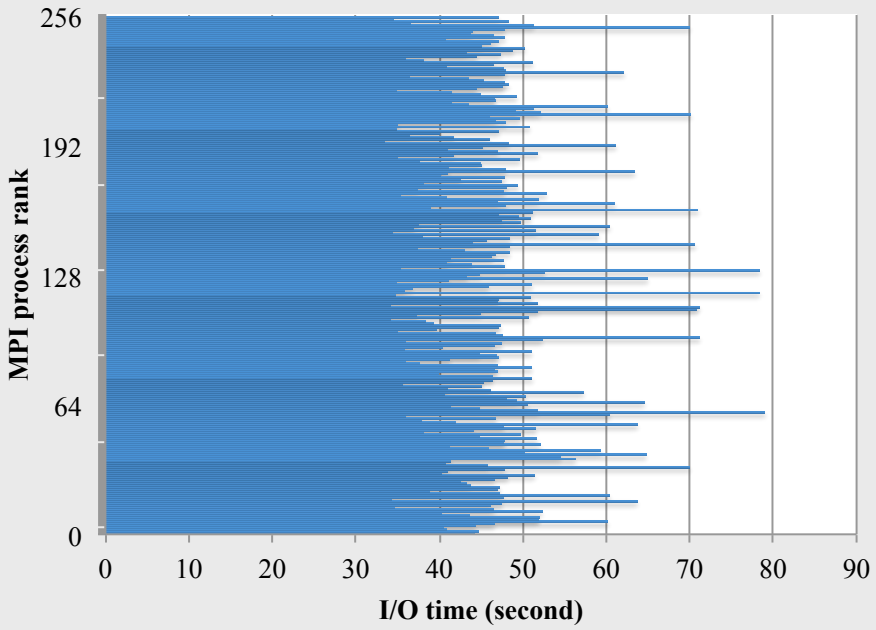
- Two components added for enhanced I/O monitoring
- Global server load monitoring
  - Monitor the overall load on the I/O servers
  - Profiles the Infiniband counters of the I/O servers
- Identifies I/O performance degradation due to high I/O subsystem load

- Lustre OST reads/writes monitoring
  - Monitor reads and writes to individual OSTs
  - Metrics aggregated together for the same OSS
  - Monitoring done at compute node level
- Identifies distribution of reads and writes on I/O subsystem
- Identifies I/O subsystem anomalies

German Research School for Simulation Sciences

I/O server imbalance

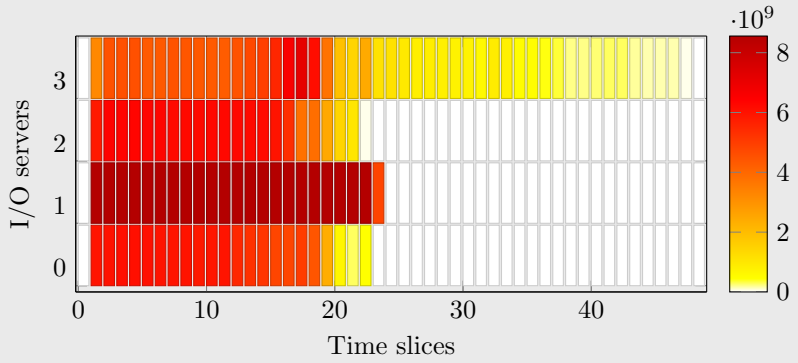
- Benchmark:
  - All processes simultaneously write to their own file
  - Each process writes 1MB of data, 2048 times
  - Observed large difference in I/O time of each process



German Research School for Simulation Sciences

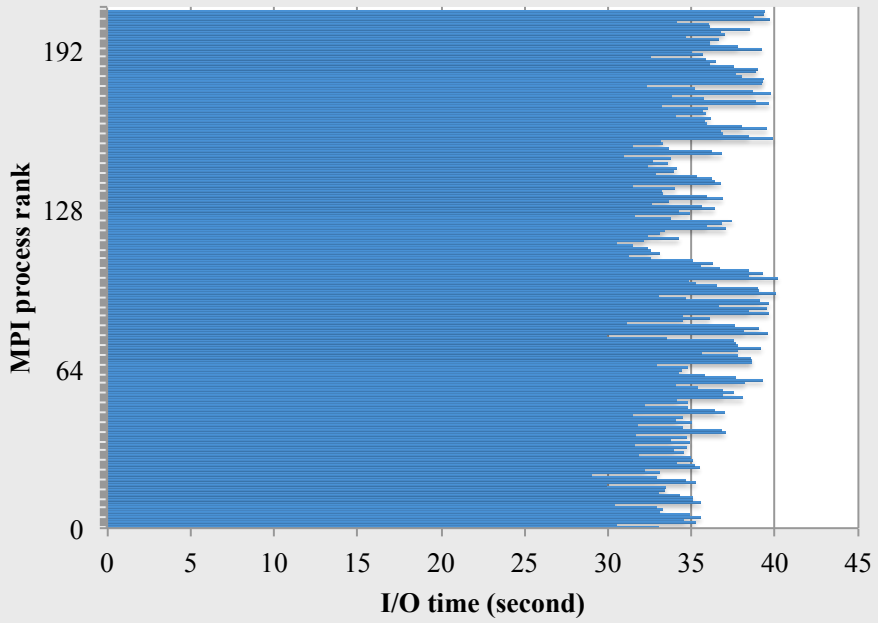
I/O server imbalance

- One I/O server had low write throughput (for that execution)
- All slow processes wrote to that server
- One of the reasons identified was that large number of writes were directed to that I/O server



I/O server imbalance

- A balanced distribution of writes lead to balanced I/O time among processes
  - Programmatically specifying a dedicated OST for each process



German Research School for Simulation Sciences

Conclusion

- External factors add to variance and performance degradation of applications
- LWM^2 can identify interference from external factors
  - Usage of time-slices to compare performance data across applications and subsystems
  - Profile BG/Q network counters to identify hot links
  - Monitor I/O subsystem to identify server-side imbalance and other anomalies
- LWM^2 available at: **`https://jay.grs.rwth-aachen.de/hg/lwm2`**

References

- A. Shah, F. Wolf, S. Zhumatiy, and V. Voevodin. Capturing inter-application interference on clusters. In IEEE International Conference on Cluster Computing (CLUSTER), 2013, pages 1–5, 2013.
- C.-S. Kuo, A. Shah, A. Nomura, S. Matsouka, and F. Wolf. How file access patterns influence interference among cluster applications. In IEEE International Conference on Cluster Computing (CLUSTER), pages 1–8, 2014.
- C.-S. Kuo. I/O subsystem as a source of inter-application interference on supercomputers. Master's thesis, German Research School for Simulation Sciences, 2014.
- L. Theisen, A. Shah, and F. Wolf. Down to earth – how to visualize traffic on high-dimensional torus networks. In Proc. of VPA: First workshop on Visual Performance Analysis, held in conjunction with Supercomputer 2014, New Orleans, LA, pages 1–6, 2014.