Slurm UG Meeting - Site report:
Dresden University of Technology
ZIH

Ulf.Markwardt@tu-dresden.de

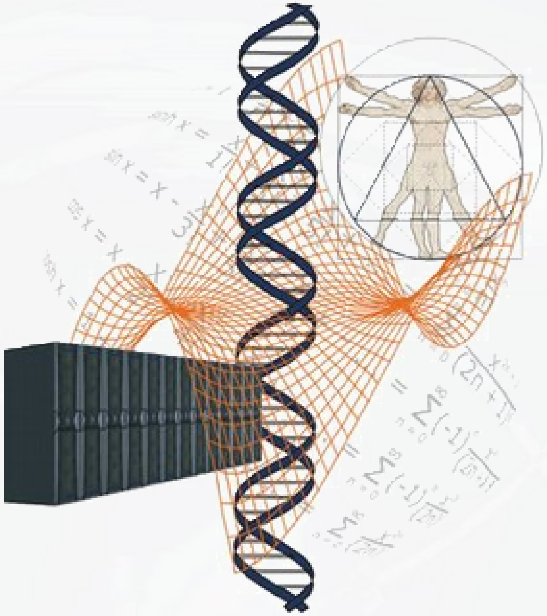
ZIH
Center for Information Services &
High Performance Computing

Dresden University of Technology

- Founded in 1828: one of the oldest technical universities in Germany
- 14 faculties and a number of specialized institutes
- More than 36.500 students, about 4000 employees, 438 professors
- One of the largest computer science faculties in Germany
- 200 million Euro annual third party funding
- 2012: University of Excellence

Center for Information Services and HPC (ZIH)

- Central Scientific Unit at TU Dresden
- Competence Center for „Parallel Computing and Software Tools“
- Strong commitment to support real users
- Development of algorithms and methods: Cooperation with users from all departments
- Providing infrastructure and qualified service for TU Dresden and Saxony



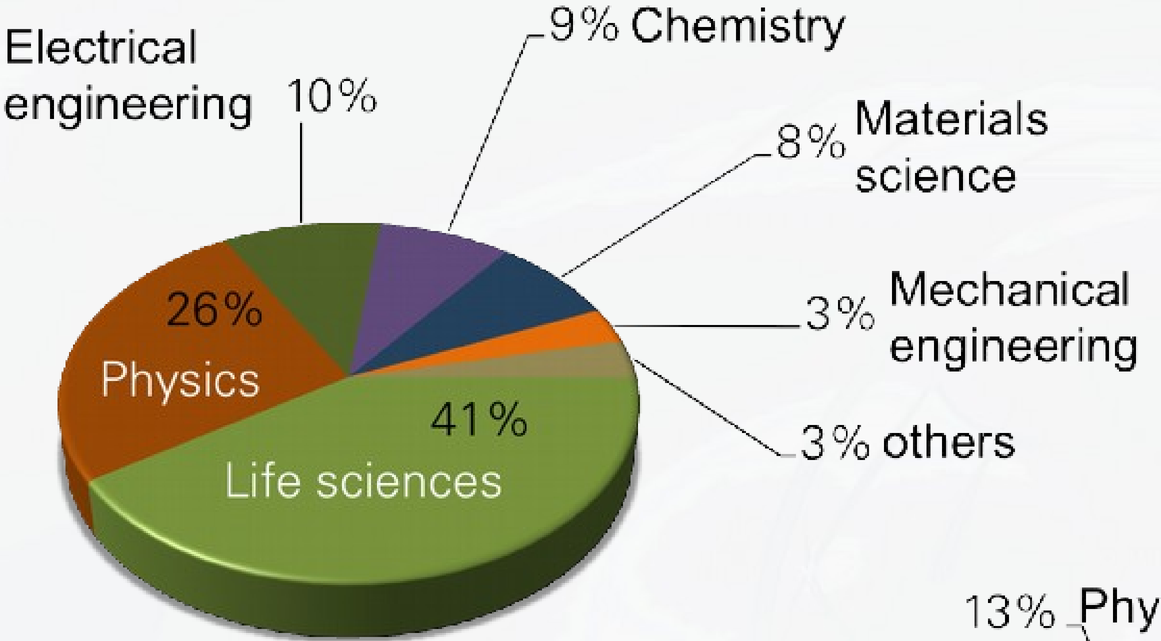
Research Topics

- Scalable software tools to support the optimization of applications for HPC systems
- Performance and energy efficiency analysis for innovative computer architectures
- Data intensive computing and data life cycle

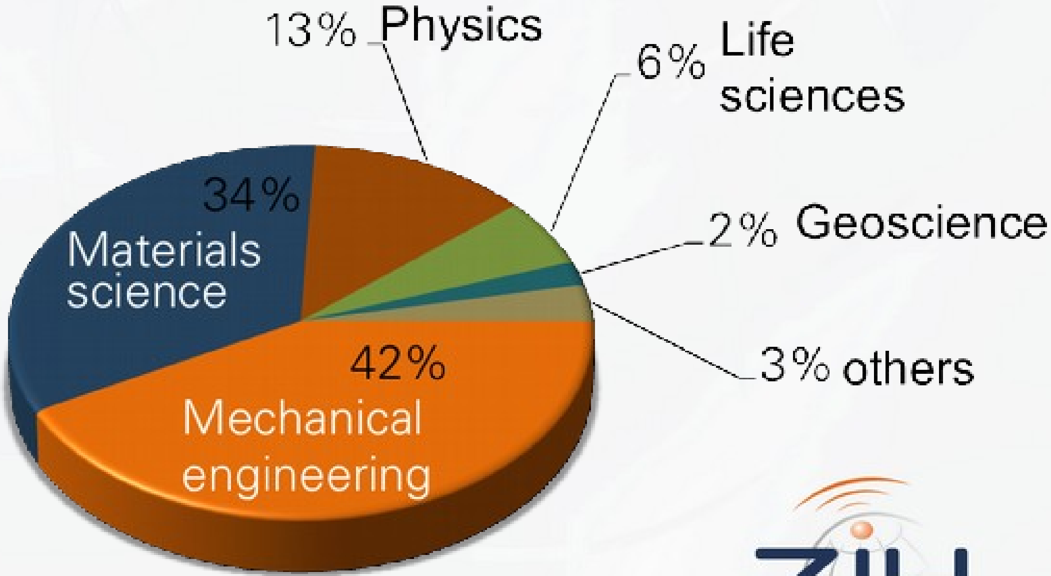
  Distributed computing and cloud computing
- Data analysis, methods and modeling in life sciences
- Parallel programming, algorithms and methods

HPC Users

- ~650 accounts
- ~150 active projects



Capacity computing



Capability computing

Bull HPC System

- 1. Phase 2013
  first contact with SLURM
  - 200 TFLOP (6000 cores)
  - Intel Sandy Bridge
  - <300 kW

- 2. Phase 2014
  - 1000 TFLOP (20000 cores)
  - Intel Haswell
  -

Research in the Field of Energy Efficiency @ ZIH

- Started 2009 with two projects:
  eeClust, Cool Computing
- Currently four active projects:
  Cool Computing II, HAEC, HDEEM *(Bull –TUD cooperation)*, Score-E
- Research Topics:
  - Power consumption instrumentation at different hardware levels
  - Integration of power monitoring in performance analysis tools
  - Modeling of energy consumption
  - Optimization of energy efficiency for applications
  - Optimization of system energy efficiency

Performance and Energy Efficiency

Cool Computing – ZIH contribution

- Event based recording of energy management in application traces
- Graphical presentation and analysis - Vampir Performance Analysis Suite

Temperature

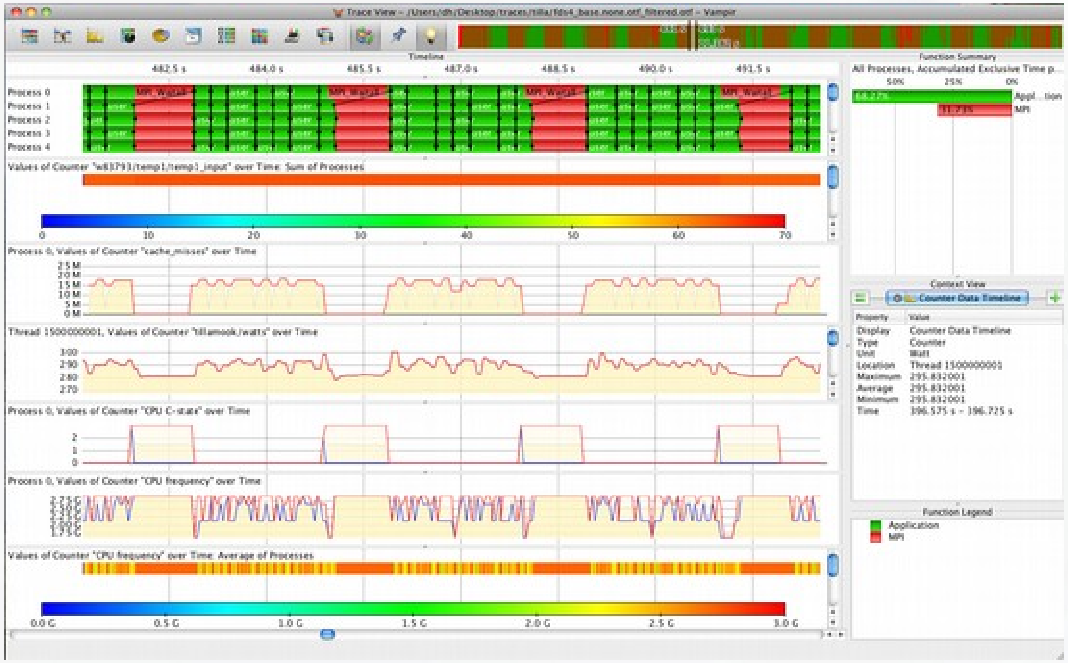
Cache Misses

Power consumption

Proc. sleep state

Proc. frequency

Avg. proc. frequency



Trace View – /Users/dh/Desktop/traces/tilla/fds4_base.none.otf_filtered.otf – Vampir

Values of Counter "w83793/temp1/temp1_input" over Time: Sum of Processes

Process 0, Values of Counter "cache_misses" over Time

Thread 1500000001, Values of Counter "tillamook/watts" over Time

Process 0, Values of Counter "CPU C-state" over Time

Process 0, Values of Counter "CPU frequency" over Time

Values of Counter "CPU frequency" over Time: Average of Processes

Function Summary

All Processes, Accumulated Exclusive Time p...

Context View

Property	Value
Display	Counter Data Timeline
Type	Counter
Unit	Watt
Location	Thread 1500000001
Maximum	295.832001
Average	295.832001
Minimum	295.832001
Time	396.575 s – 396.725 s

Function Legend

🟩 Application

🟥 MPI

- Evaluation of computer systems
- Energy-saving techniques, e.g. for Linux Kernel CPU frequency switching

TECHNISCHE UNIVERSITÄT DRESDEN

ZIH Center for Information Services & High Performance Computing

Comparison of Power Measurement Techniques

- “Power Measurement Techniques on Standard Compute Nodes: A Quantitative Comparison” (D. Hackenberg et al., ISPASS 2013)
- Compares RAPL (Intel), APM (AMD), ZES LMG, two IPMI solutions, and a National Instruments DAC
- SLURM uses RAPL or IPMI measurements

RAPL

- Is modeled – not measured!
- Accuracy depends on workload
- Does not cover devices

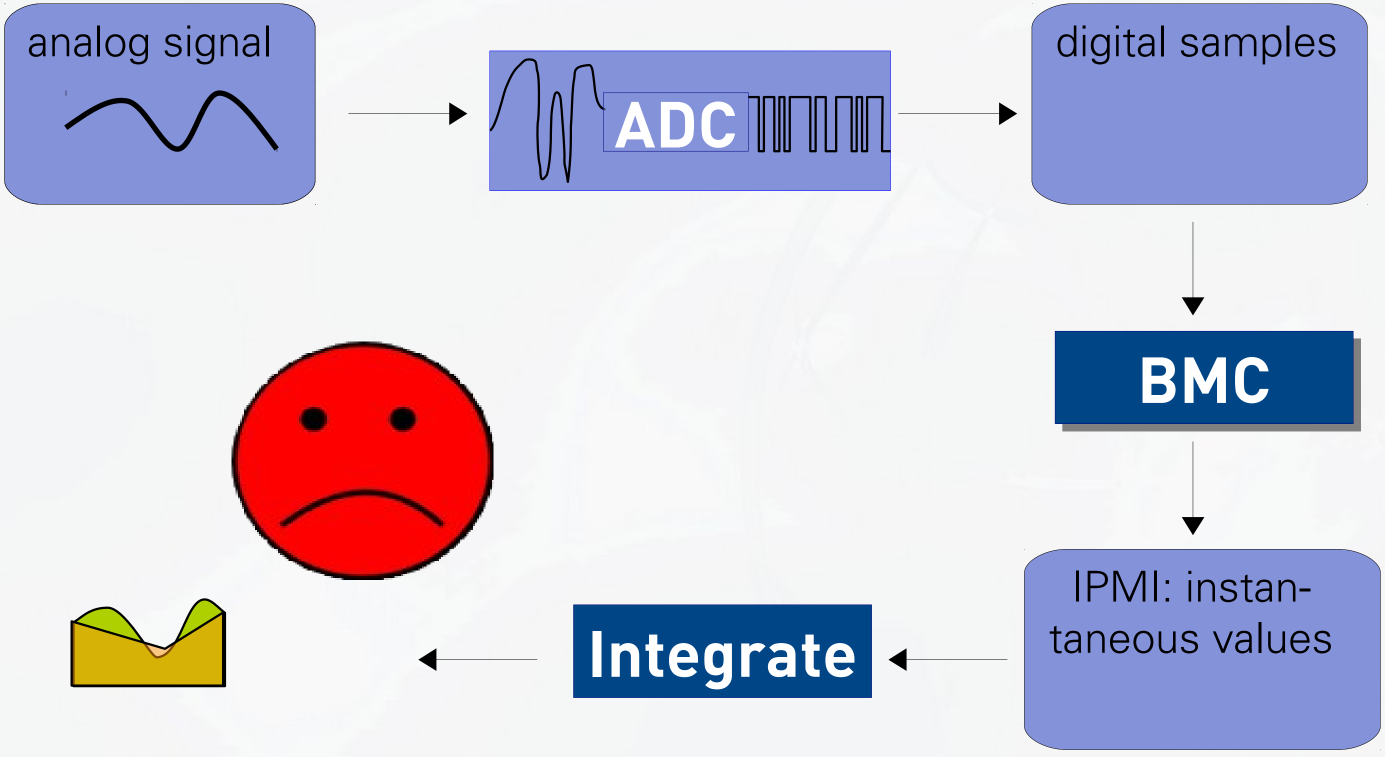
IPMI

- Lack of temporal resolution
- Often provides instantaneous measurements

Energy "Measurement" – where can things go wrong?

- You cannot directly measure electrical energy

$$E = \int_{t_a}^{t_b} u(t)\,i(t)\,dt \cong \sum_{t=t_a}^{t_b} u(t)\,i(t)\,\Delta t$$



analog signal

ADC

digital samples

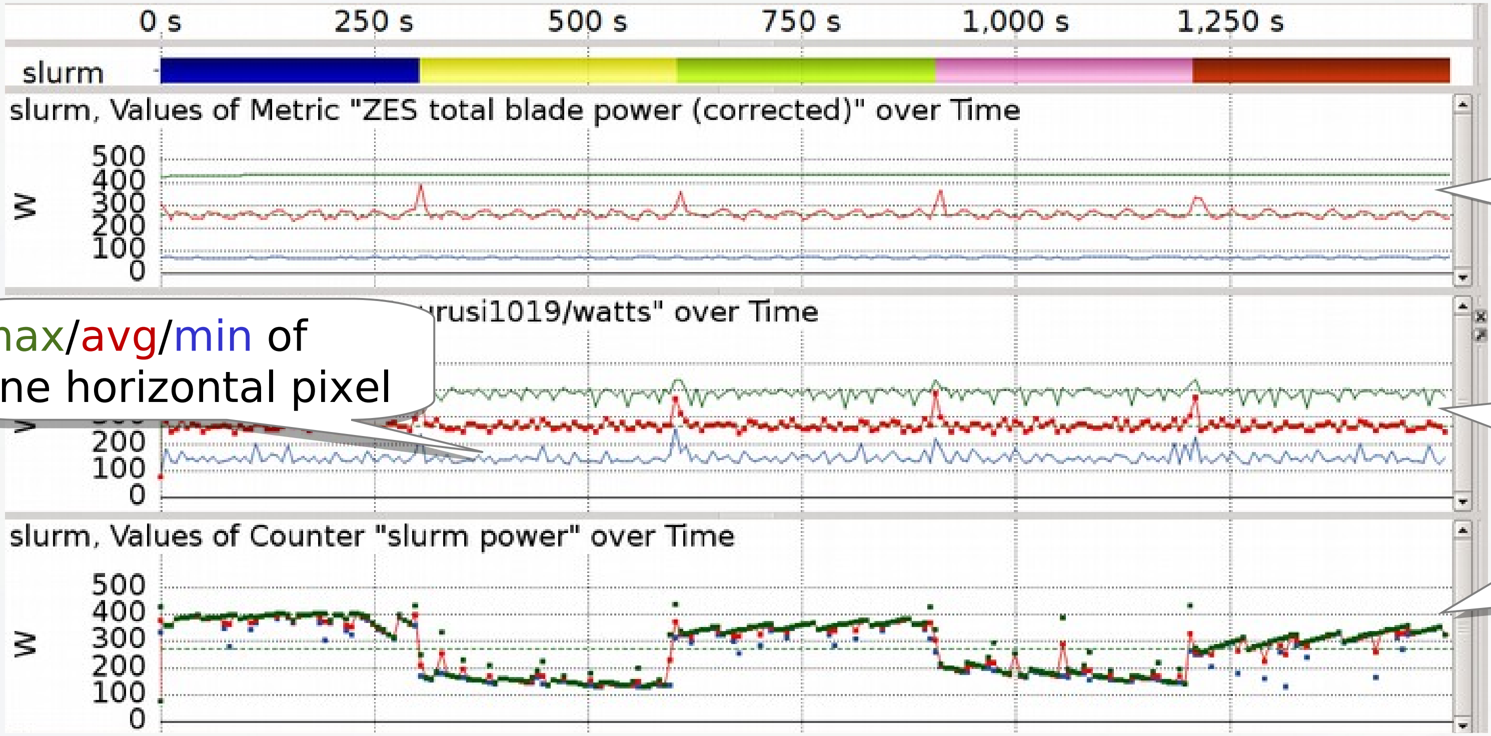
BMC

IPMI: instantaneous values

Integrate

Aliasing Effects on Energy Accounting

- Using the default 3s sampling interval in SLURM / IPMI
  - Synthetic high/low load workload with regular intervals
- Energy reported by SLURM ranges from 48 to 111kJ (5 identical job steps)
  - Reference measurement 78±1 kJ per step


0 s
250 s
500 s
750 s
1,000 s
1,250 s
slurm
slurm, Values of Metric "ZES total blade power (corrected)" over Time
W
500
400
300
200
100
0
rusi1019/watts" over Time
200
100
0
slurm, Values of Counter "slurm power" over Time
W
500
400
300
200
100
0

Correct Information

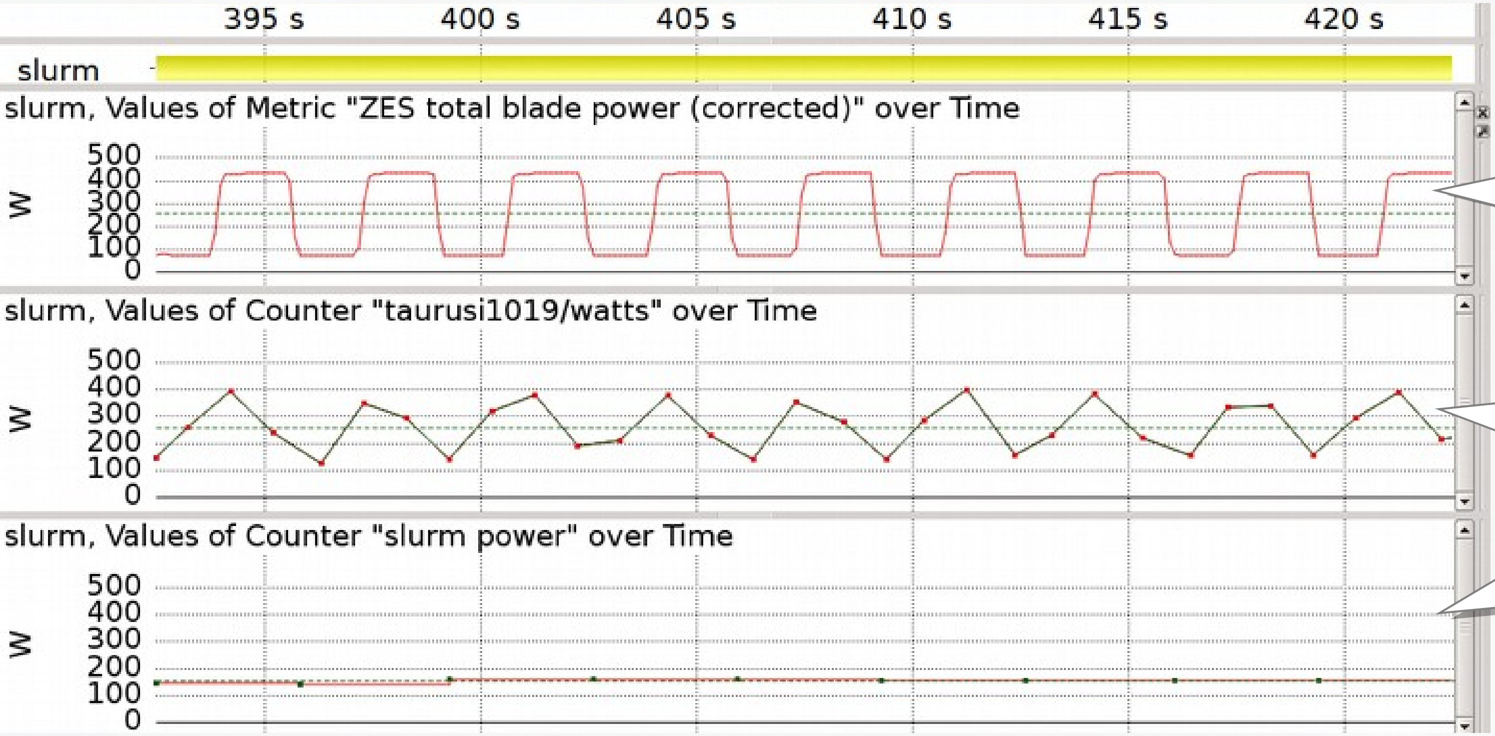
max/avg/min of one horizontal pixel

IPMI (1s) Information

Slurm (3s) Information

Aliasing Effects on Energy Accounting

- Using the default 3s sampling interval in SLURM / IPMI
  - Synthetic high/low load workload with regular intervals
- Energy reported by SLURM ranges from 48 to 111kJ (5 identical job steps)
  - Reference measurement 78±1 kJ per step



slurm, Values of Metric "ZES total blade power (corrected)" over Time

slurm, Values of Counter "taurusi1019/watts" over Time

slurm, Values of Counter "slurm power" over Time

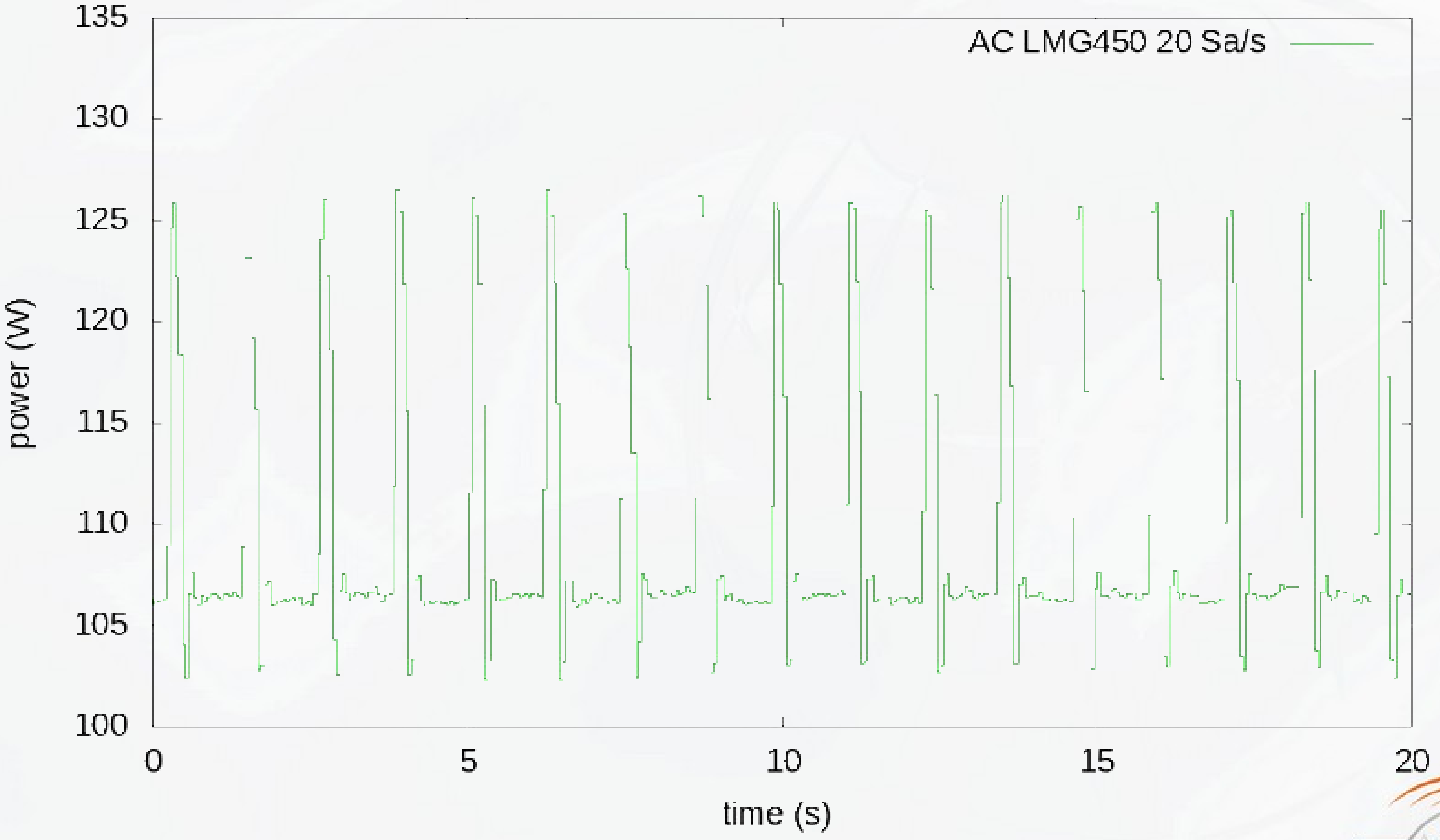
Correct Information

IPMI (1s) Information

Slurm (3s) Information

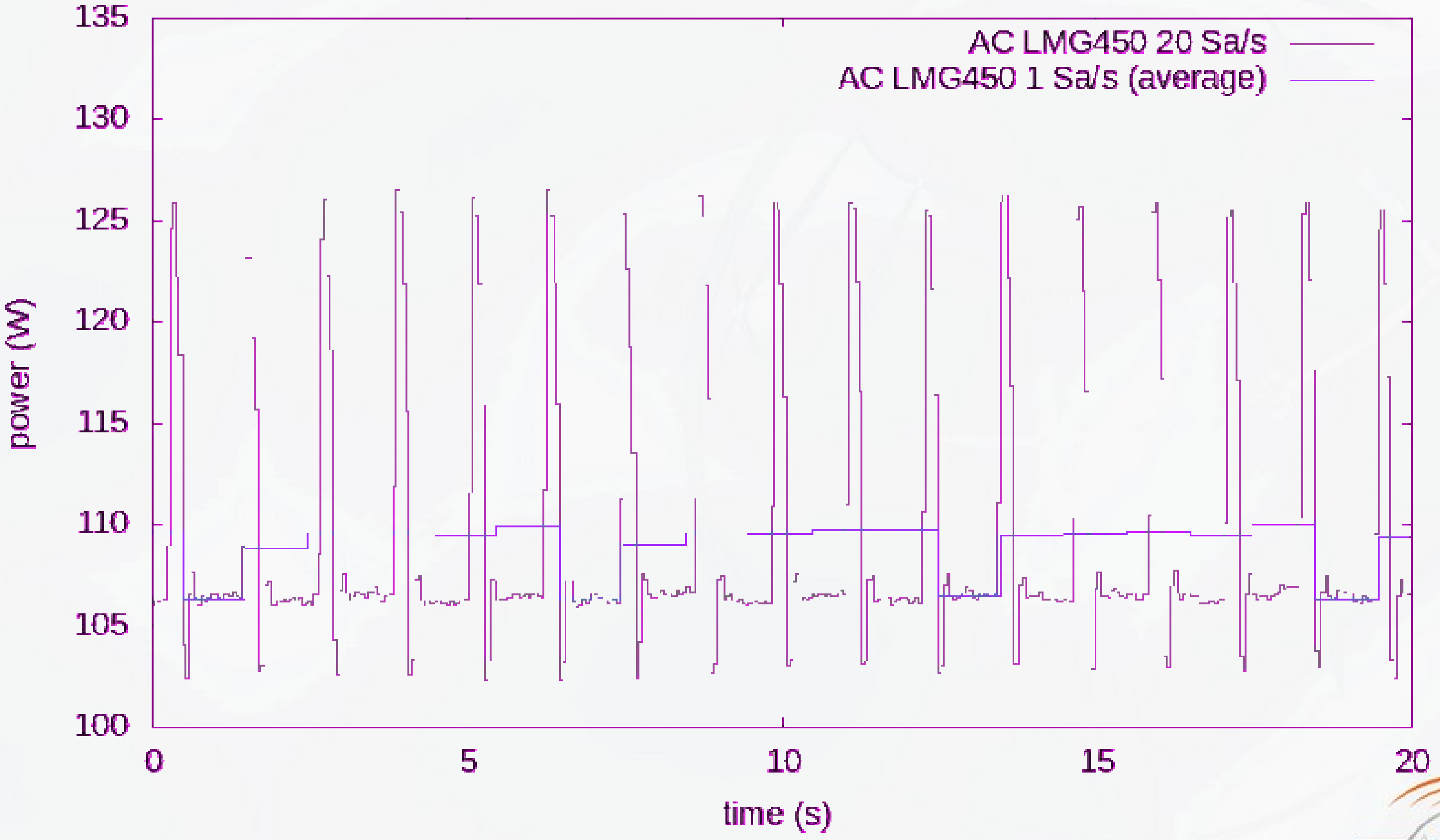
IPMI power samples

- Workload: provide a pulse of high CPU load at ~0.9 Hz
- Measured with: ZES LMG 450, 20 Sa/s



IPMI power samples

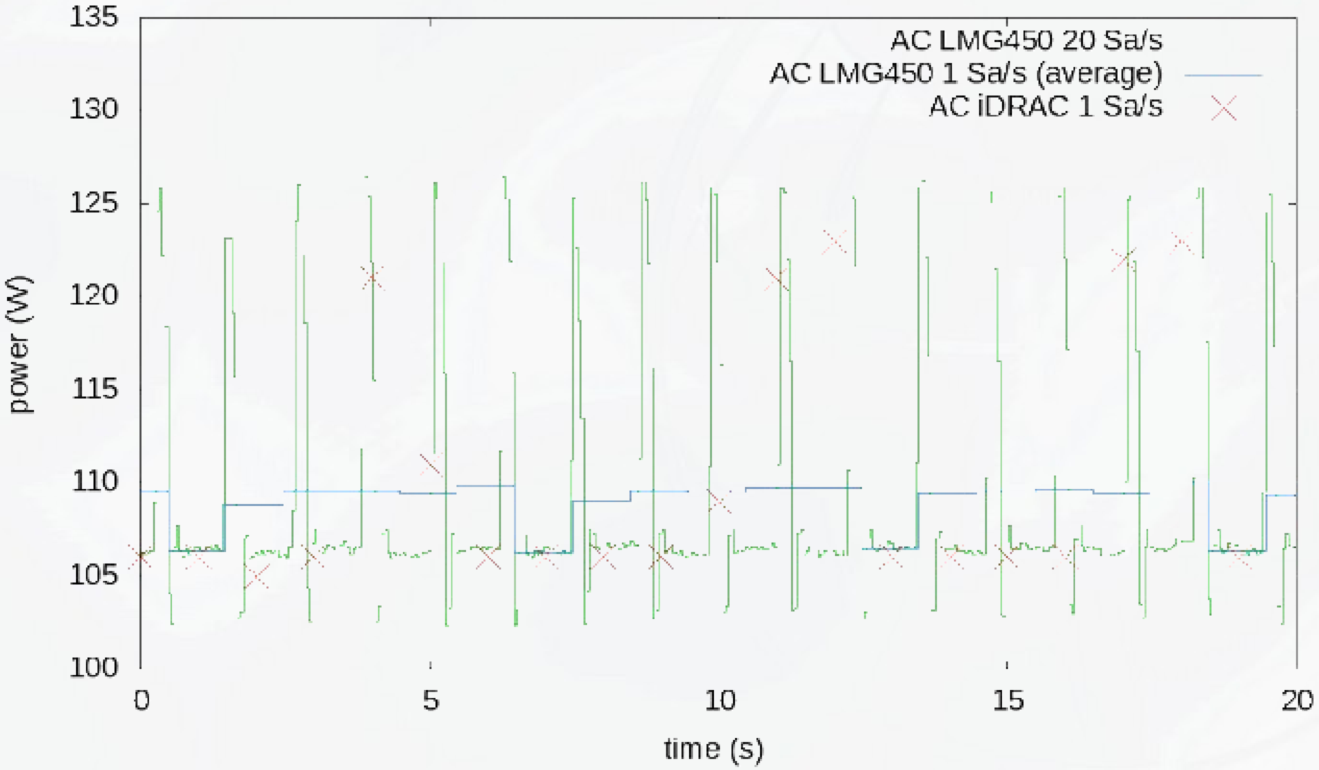
- Workload: provide a pulse of high CPU load at ~0.9 Hz
- Measured with: ZES LMG 450, 20 Sa/s / 1 Sa/s;



IPMI power samples

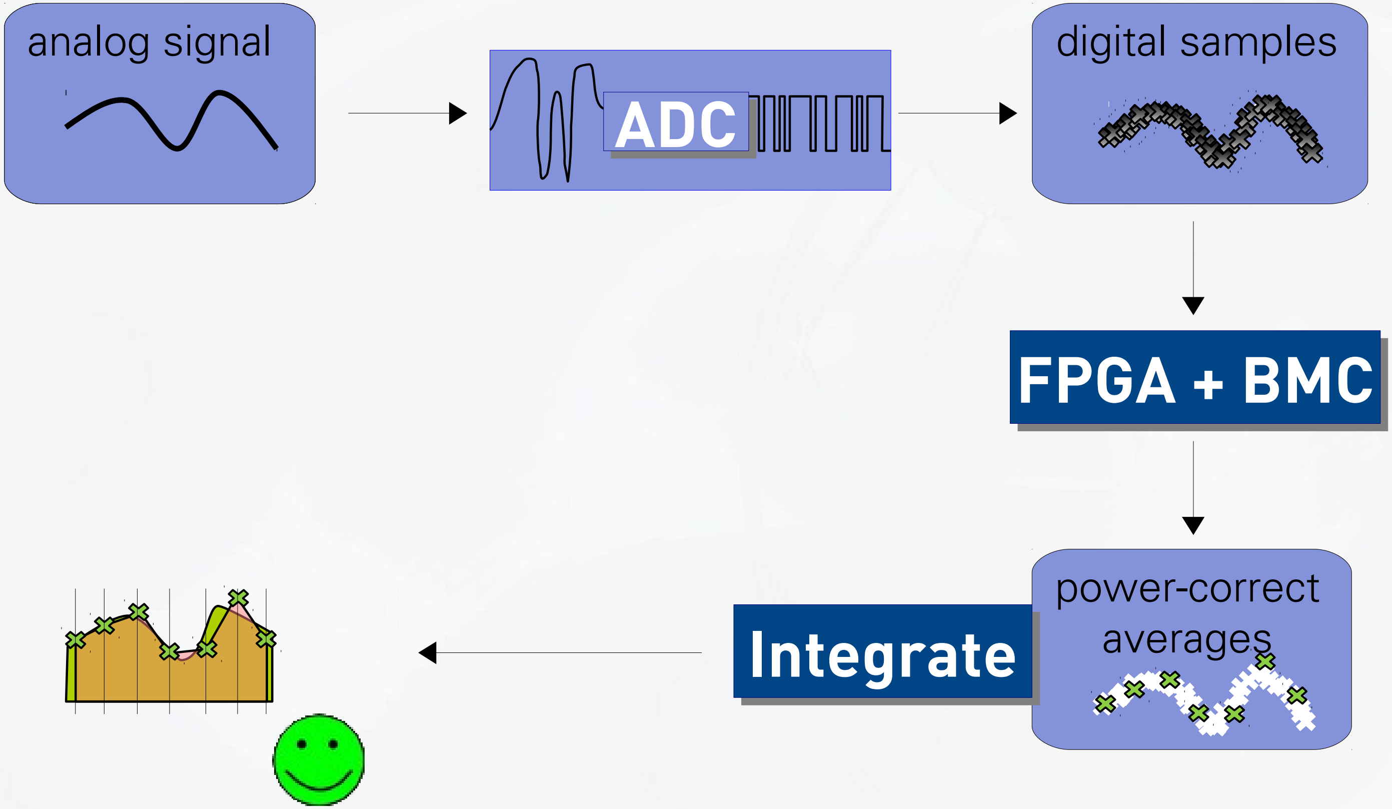
- Workload: provide a pulse of high CPU load at ~0.9 Hz
- Measured with: ZES LMG 450, 20 Sa/s / 1 Sa/s;

  AC iDRAC (Dell PSU via IPMI), 1 Sa/s
- PSU measurement does not integrate power consumption over time



HDEEM cooperation (BULL – TUD)

- FPGA supported measurements
- Higher internal sampling rate, better averaging



analog signal

ADC

digital samples

FPGA + BMC

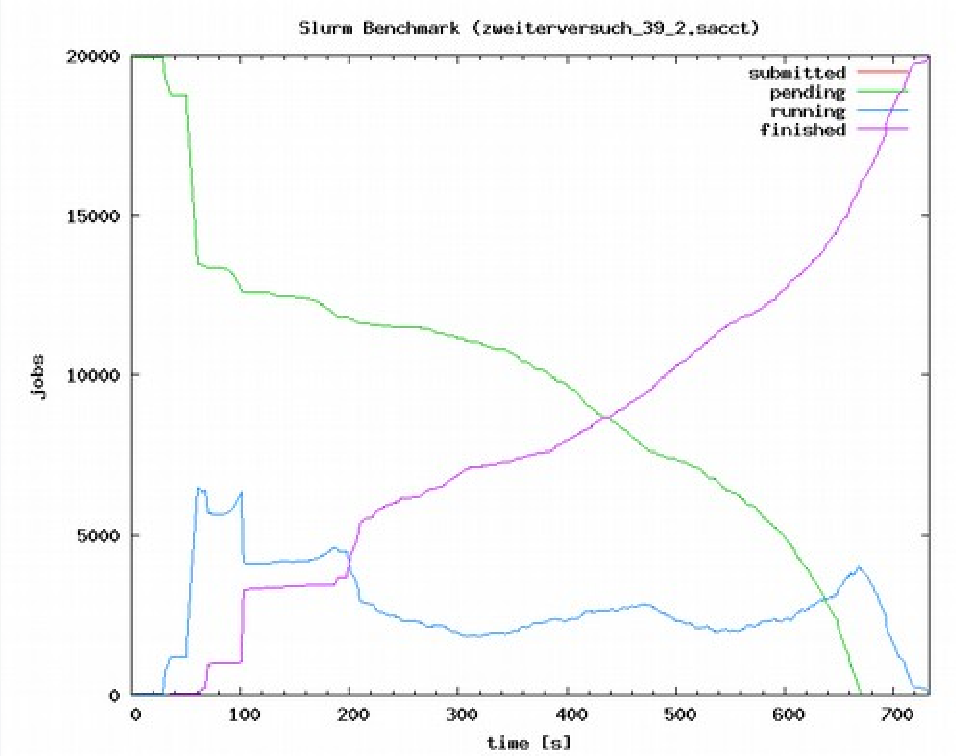
power-correct averages

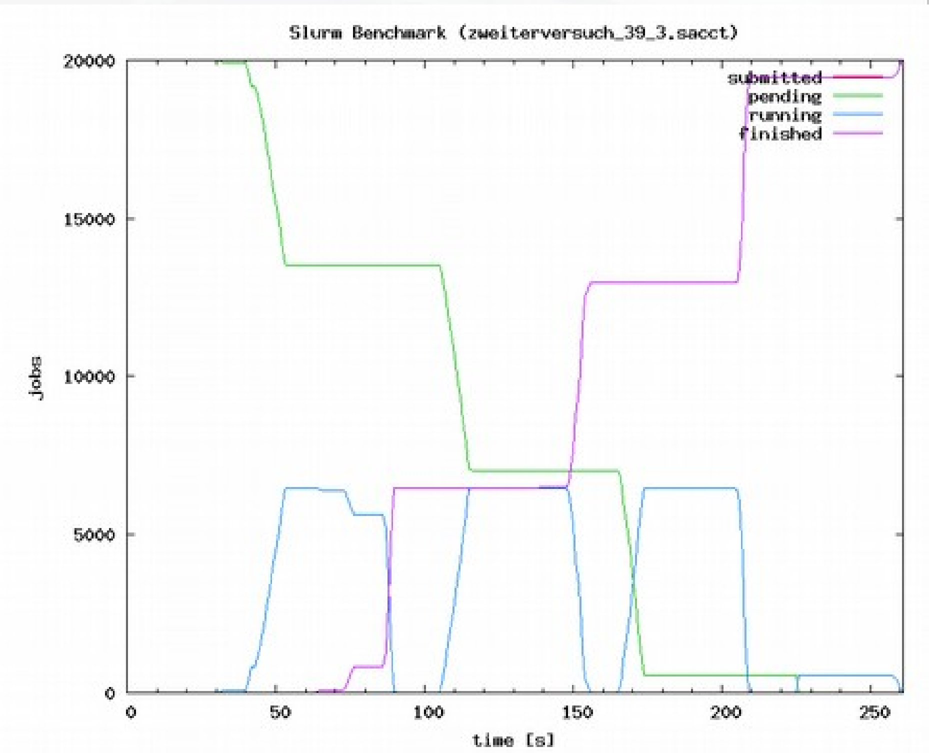
Integrate

SLURM in Production

- We need large job array for single-core jobs (100.000) with low impact from scheduler

  example: 20.000 jobs 'sleep 30' on 5.000 cores (120s net) :





Comments

- The batch system should give a better pending reason than
  Cannot, have not – and especially not for you !
  E.g. full system reservation could be mentioned as a reason.
- Multi-objective scheduling
  - Fair share Dona Crawford: memory - the most precius resource.
  - Minimize fragmentation with respect to CPU / memory
  - How to customize SLURM for CPU / memory usage?
- Replay engine would be great!
  Simulation instead of understanding :-)

Thank you

- ... for your attention,
- ... for good discussions,
- ... for the support !