A Case Study of Energy Aware Scheduling on SuperMUC

Axel Auweter, Leibniz Supercomputing Centre of the Bavarian Academy of Sciences and Humanities
Luigi Brochard, IBM Systems Technology Group

LRZ, SuperMUC, and Energy Efficiency…

Leibniz Supercomputing Centre

„Twin Cube“ with computing rooms (3,160m² for IT equipment, 6,393m² for Infrastructure, 2x10MW Power supply)

SuperMUC

9,216 Thin Nodes (2x Xeon E5-2680 8C), 288 TB RAM – 205 Fat Nodes (4x Xeon E7-4870 10C), 52 TB RAM
Infiniband FDR10 Interconnect, 10 PB Storage (200 GB/s bandwidth)

Energy Efficiency at LRZ

Energy Efficient Hardware

- Use latest semiconductor technology
- Use energy saving processor and memory technologies
- Consider using special hardware or accelerators designed for specific scientific problems or numerical algorithms

Energy Efficient Infrastructures

- Reduce power losses in the power supply chain
- Improve cooling technology
- Reuse waste heat from IT systems

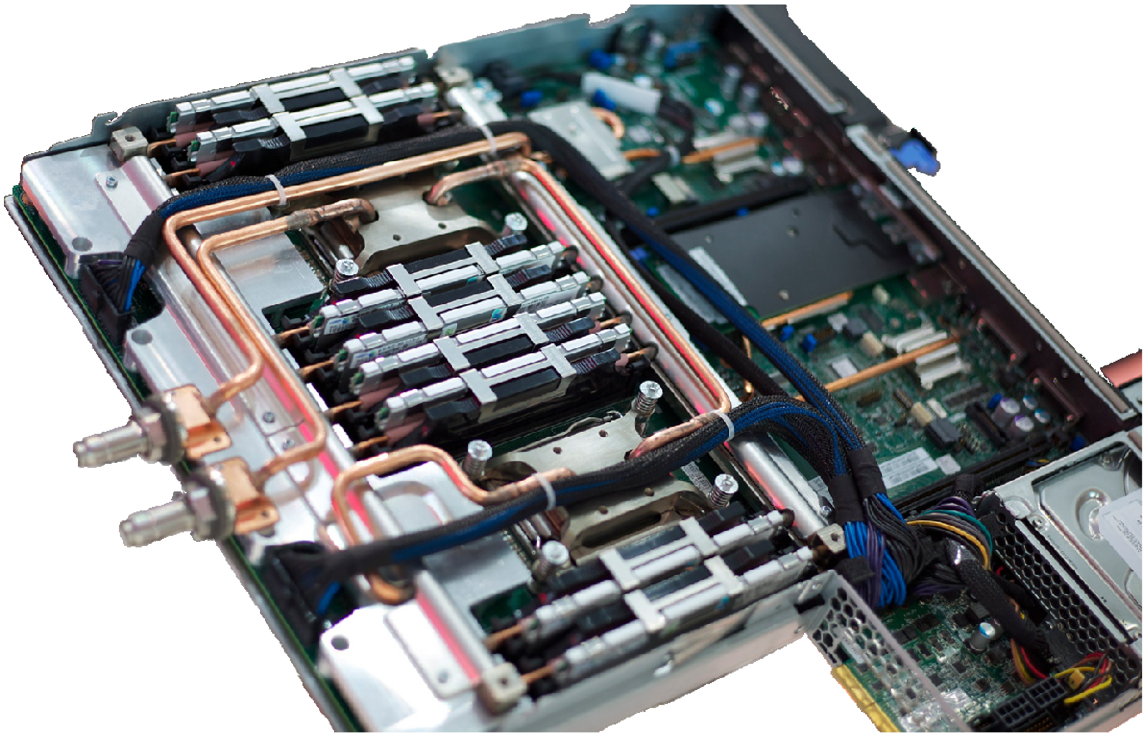
Energy Aware Management Software

- Monitor the energy consumption of the compute system and the building infrastructure
- Use energy aware system software to exploit the energy saving features of the platform

Energy Efficient Applications

- Use the most efficient algorithms
- Use best libraries
- Use most efficient programming paradigm

Direct Liquid Cooling



- **Heat flux to water > 90%**
- **Very little need for chilled water**
- **Power advantage over air-cooled node:**
  - Warm water cooled: ~10% (Cold water cooled: ~15%)
  - Due to lower component temperature and no fans
- **Typical operating conditions:**
  - T_{air} = 25-30°C
  - T_{water} = 18-45°C

Energy Aware Scheduling…

Energy Aware Scheduling

Monitor and report Energy-to-Solution for all applications

Use Voltage and Frequency Scaling capabilities of CPUs

Power and Performance Prediction of applications

Power off / suspend unused nodes

Who of you has tried to boot an Infiniband cluster with ~ 10.000 nodes?

It‘s not that easy!
Suspend/Resume on Supe[...]
is still work-in-progres[...]

It‘s all about the applicability of the features for production environments!

Energy Aware Scheduling

Monitor and report Energy-to-Solution for all applications

Use Voltage and Frequency Scaling capabilities of CPUs

Power and Performance Prediction of applications

Power off / suspend unused nodes

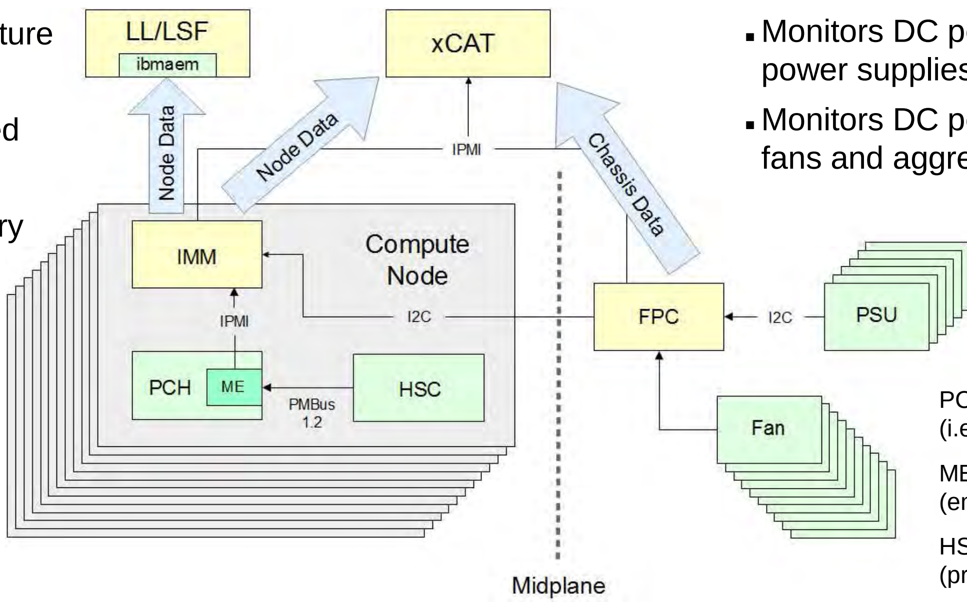
Monitor & Report Energy-to-Solution for Applications

IMM = Integrated Management Module (Node-Level Systems Management)

- Monitors DC power consumed by node as a whole and by CPU and memory subsystems
- Monitors inlet air temperature for node
- Caps DC power consumed by node as a whole
- Monitors CPU and memory subsystem throttling caused by node-level throttling
- Enables or disables power savings for node

FPC = Fan/Power Controller (Chassis-Level Systems Mgmt)

- Monitors AC power consumed by individual power supplies and aggregates to chassis level
- Monitors DC power consumed by individual power supplies and aggregates to chassis level
- Monitors DC power consumed by individual fans and aggregates to chassis level



PCH = Platform Controller Hub (i.e., south bridge)

ME = Management Engine (embedded in PCH, runs Intel NM firmware)

HSC = Hot Swap Controller (provides power readings)

Energy Aware Scheduling

Monitor and report Energy-to-Solution for all applications

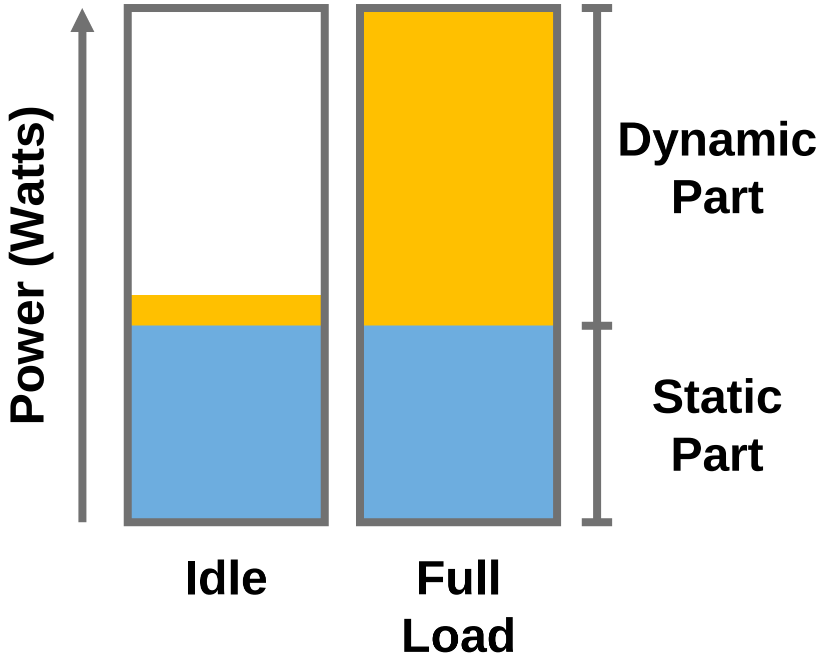
Use Voltage and Frequency Scaling capabilities of CPUs

Power and Performance Prediction of applications

Power off / suspend unused nodes

Predicting Runtime and Power Consumption of Applications under Frequency Scaling…

Modeling Compute Node Power Consumption



- Simplified Model for Compute Node Power
- Dynamic Part:
  - Processor
  - Memory
- Static Part:
  - South Bridge
  - NIC
  - …

Predicting Power Consumption at CPU Frequency f_n

$$\underbrace{PWR(f_n)}_{\substack{\textbf{Node Power}\\ \textbf{consumption at}\\ \textbf{CPU frequency } f_n}} = A_n * \underbrace{GIPS(f_0)}_{\substack{\textbf{Giga Instructions}\\ \textbf{per second at ref.}\\ \textbf{frequency } f_0}} + B_n * \underbrace{GBS(f_0)}_{\substack{\textbf{Gigabytes per}\\ \textbf{second transferred}\\ \textbf{at ref. frequency } f_0}} + C_n$$

Platform-specific power coefficients for predicting the node power at frequency f_n

Generation of Platform Coefficients A_n, B_n, and C_n

$$PWR(f_n) = A_n * GIPS(f_0) + B_n * GBS(f_0) + C_n$$

- Assemble a list of compute kernels with a large spectrum of GIPS and GBS characteristics
- For each kernel: measure GIPS and GBS at f_0
- For each kernel, for each CPU frequency f_n:
  - Execute kernel
  - Measure average node power $PWR(f_n)$
- Approximate A_n, B_n, C_n for each frequency f_n to satisfy the equation for all kernels

Predicting Runtime at CPU Frequency f_n

$$\underbrace{CPI(f_n)}_{\substack{\text{Cycles Per}\\ \text{Instruction at CPU}\\ \text{frequency } f_n}} = D_n * \underbrace{CPI(f_0)}_{\substack{\text{Cycles Per}\\ \text{Instruction at ref.}\\ \text{frequency } f_0}} + E_n * \underbrace{TPI(f_0)}_{\substack{\text{Memory Transactions}\\ \text{Per Instruction at ref.}\\ \text{frequency } f_0}} + F_n$$

$$TIME(f_n) = TIME(f_0) * \frac{CPI(f_n)}{CPI(f_0)} * \frac{f_0}{f_n}$$

Energy Aware Scheduling

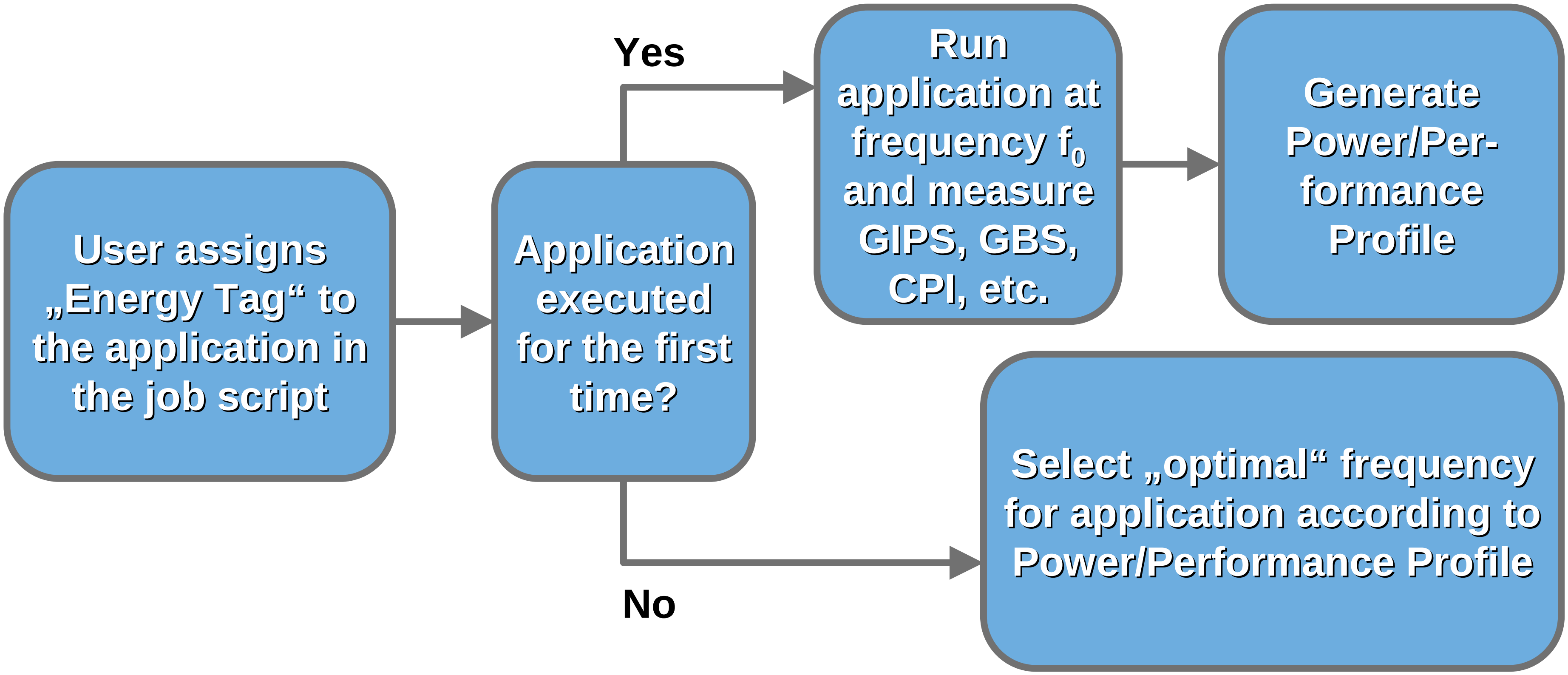
Monitor and report Energy-to-Solution for all applications

Use Voltage and Frequency Scaling capabilities of CPUs

Power and Performance Prediction of applications

Power off / suspend unused nodes

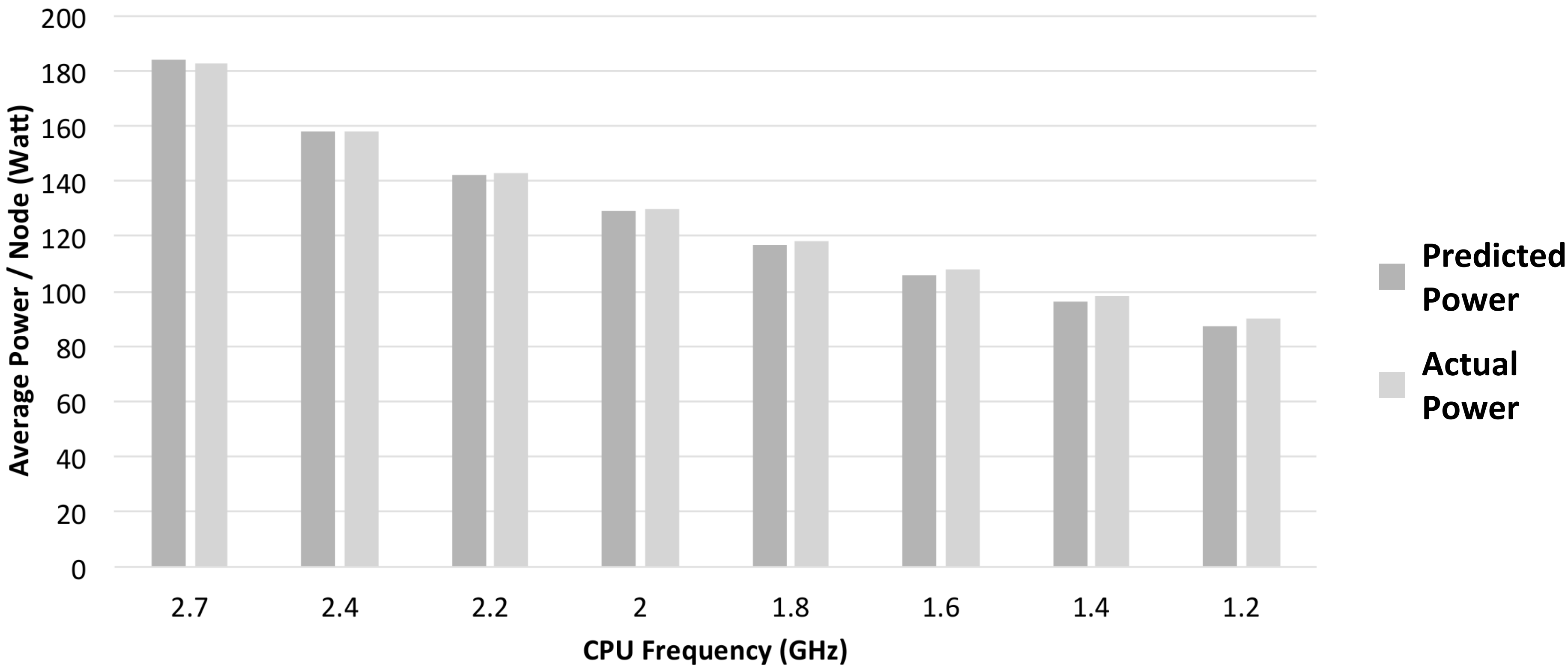
IBM LoadLeveler EAS Implementation



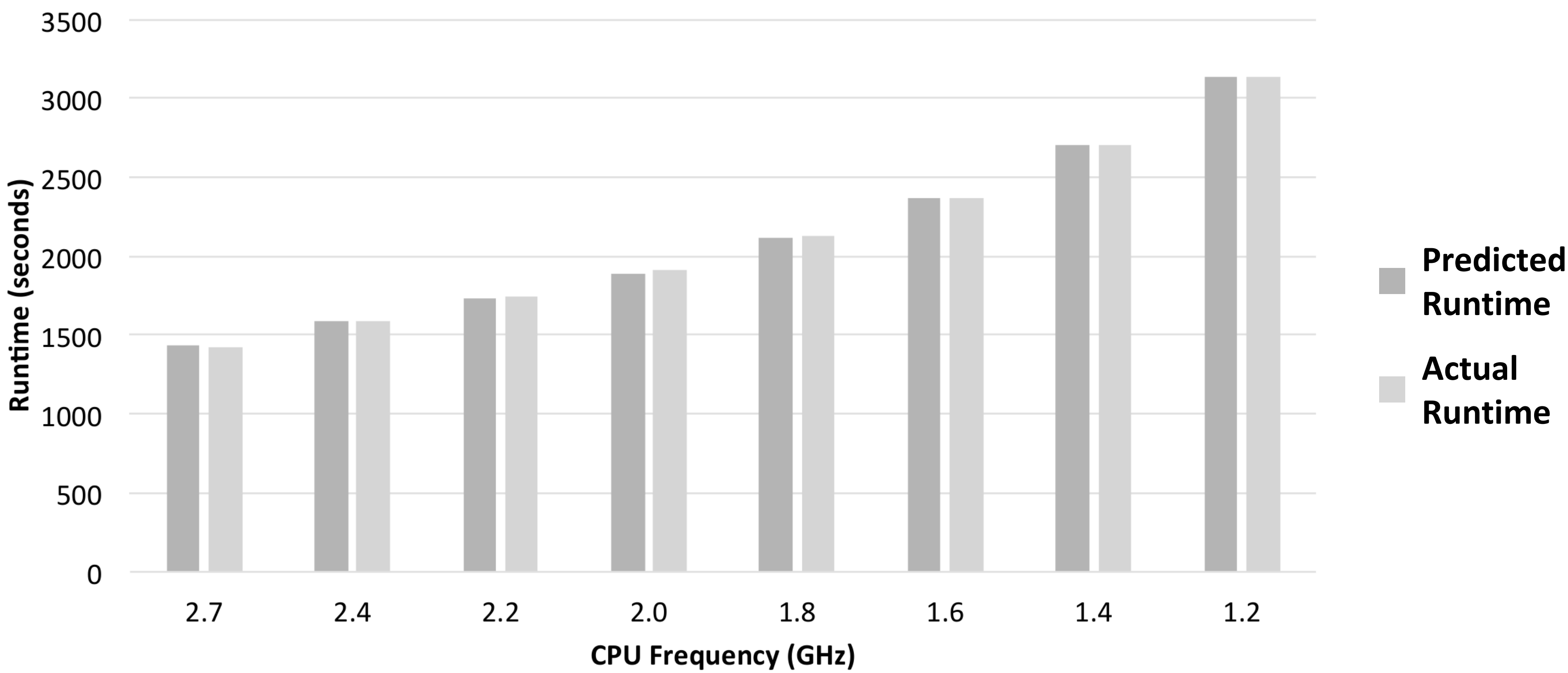
Validation of the Prediction Model

- LoadLeveler predicts runtime and power consumption for applications running at CPU frequency f_n
- We can compare the predictions to actual measurements
- 4 Application Benchmarks:
  - Quantum Espresso
  - Gadget
  - Seissol
  - WaLBerla
- 2 Synthetic Benchmarks:
  - Parallel Dense Matrix Multiply

Gadget Power Prediction



Gadget Runtime Prediction



Results Overview (**W**orstcase **P**rediction **E**rror)

Quantum Espresso

Nodes: 16
Parallelization: Hybrid
(4 MPI Tasks, 4 OpenMP Threads)
WPE Power: 1.4%
WPE Runtime: 4.6%

Gadget

Nodes: 8
Parallelization: Hybrid
(4 MPI Tasks, 4 OpenMP Threads)
WPE Power: 2.7%
WPE Runtime: 0.7%

PMatMul

Nodes: 64
Parallelization: MPI only
(1024 MPI Tasks)
WPE Power: 0.9%

Seissol

Nodes: 16
Parallelization: Hybrid
(1 MPI Tasks, 16 OpenMP Threads)
WPE Power: 2.6%
WPE Runtime: 2.6%

WaLBerla

Nodes: 64
Parallelization: MPI only
(1024 MPI Tasks)
WPE Power: 2.4%
WPE Runtime: 1.8%

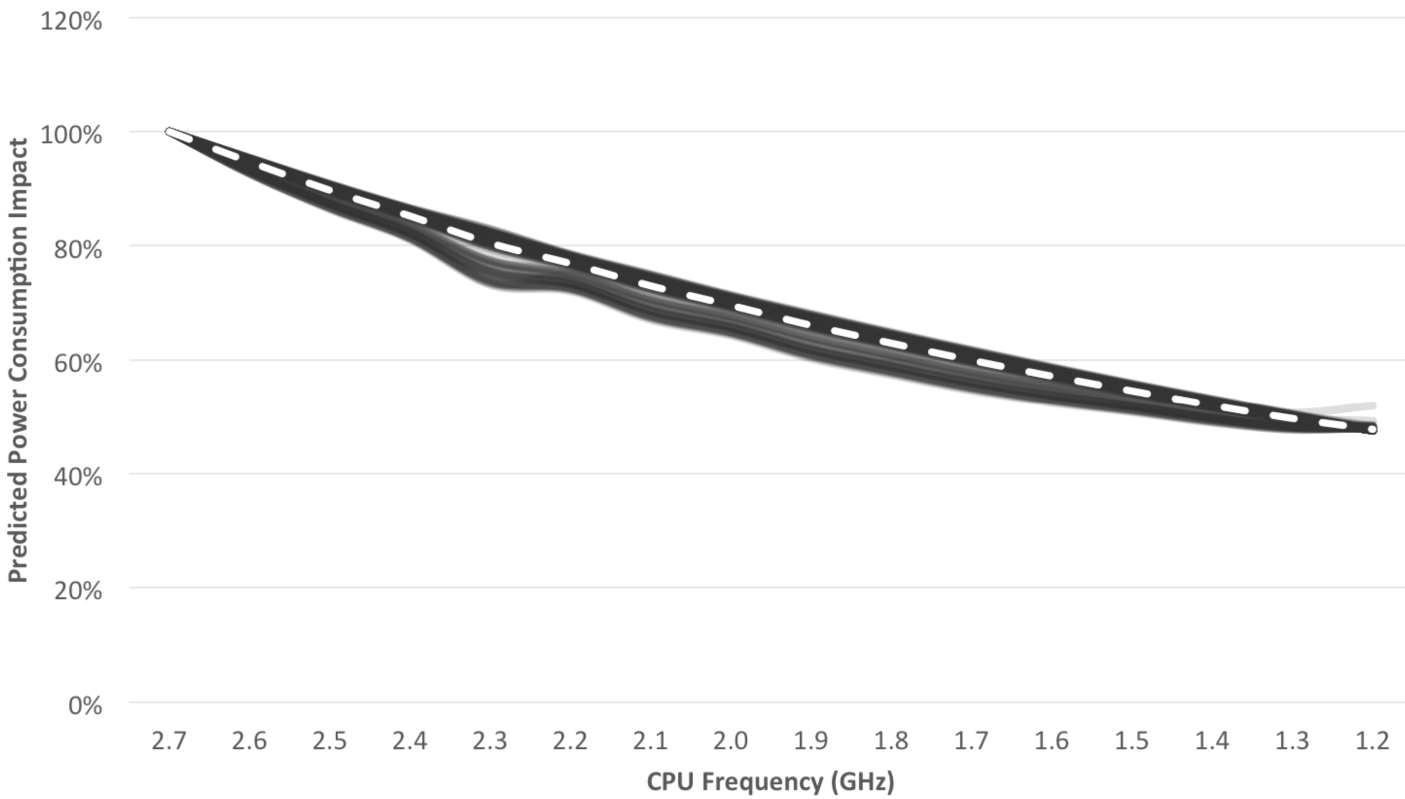
Predictions are accurate!
Let's analyze the LRZ application portfolio…

Analyzing the LRZ Application Portfolio…

What is a reasonable default frequency for unknown applications?

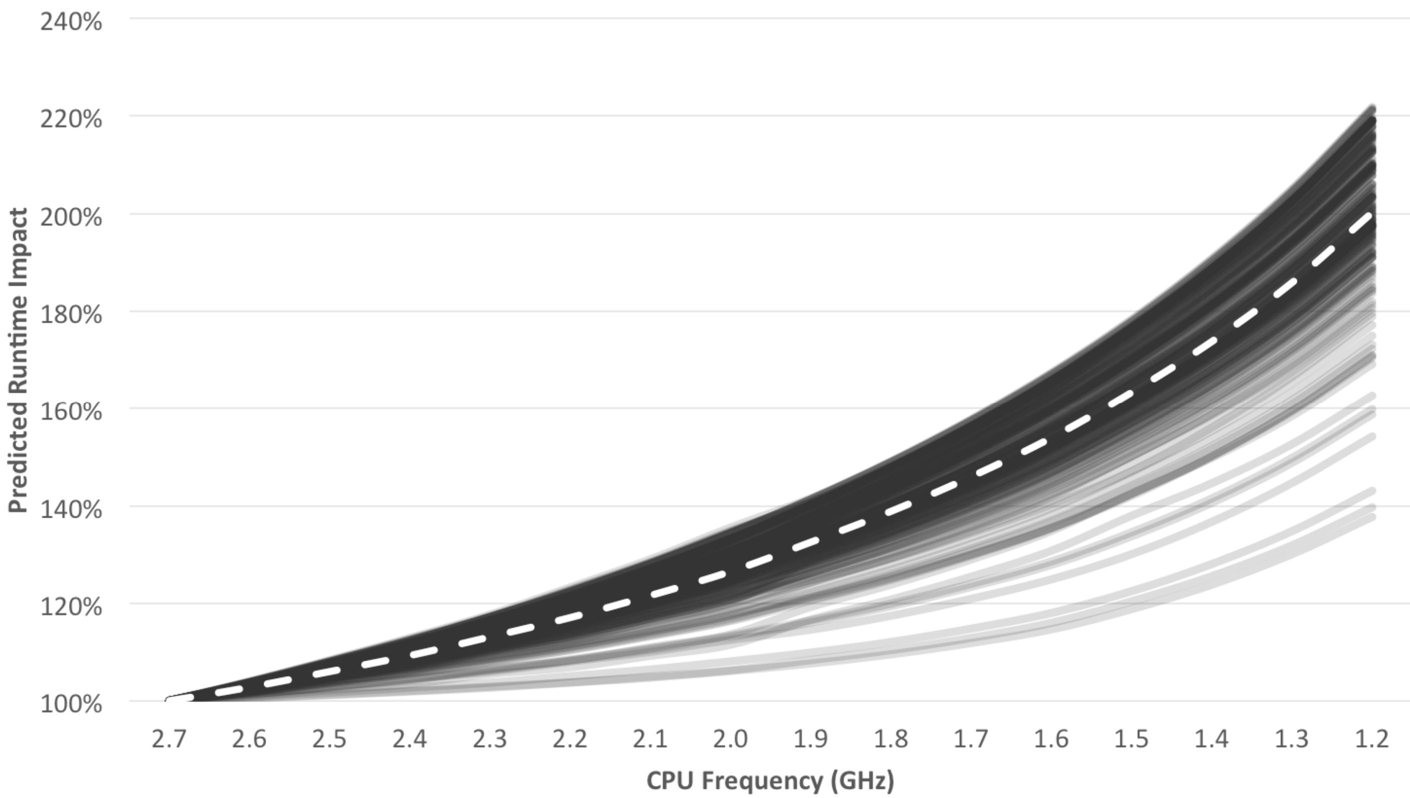
What is an optimal tradeoff between energy savings and runtime increase?

SuperMUC Application Power Consumption



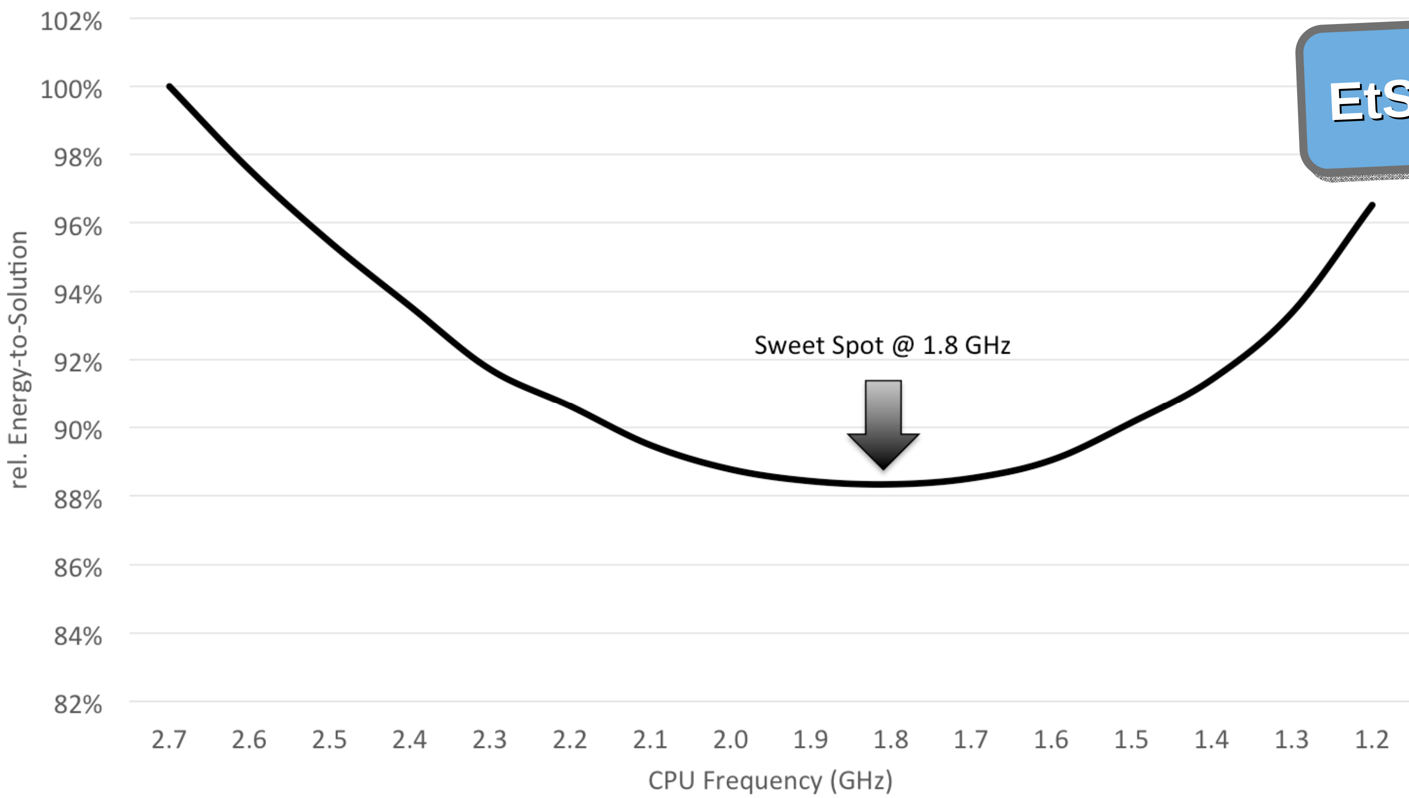
SuperMUC application power consumption under CPU frequency scaling.

SuperMUC Application Runtime



SuperMUC application runtime under CPU frequency scaling.

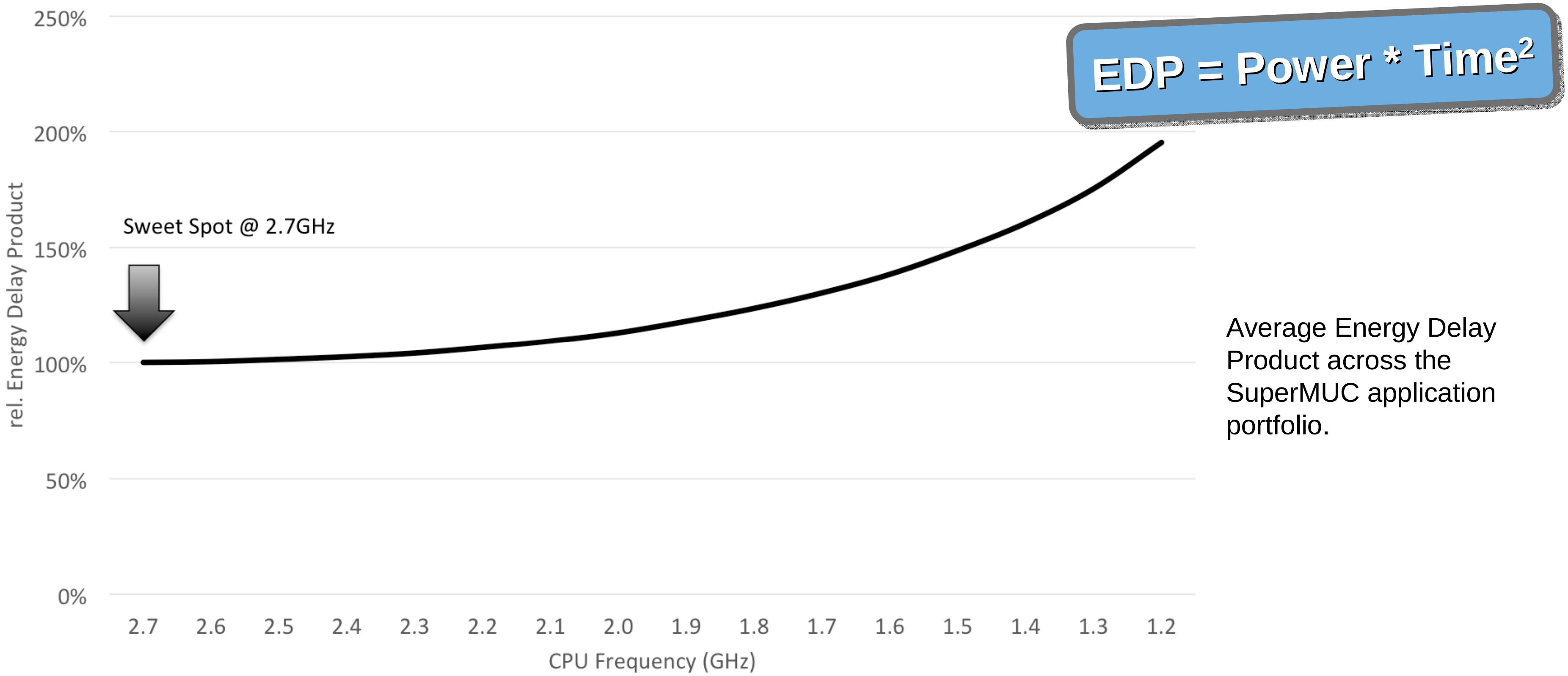
SuperMUC Energy-to-Solution



EtS = Power * Time

Average Energy-to-Solution across the SuperMUC application portfolio.

SuperMUC Energy Delay Product



Average Energy Delay Product across the SuperMUC application portfolio.

The LRZ Energy Aware Scheduling Policy

- Policy Decisions:
  - All applications run at 2.3 GHz by default
  - If the runtime decreases by at least 5% when going from 2.3 to 2.5 GHz → Run at 2.5 GHz
  - If the runtime decreases by at least 12% when going from 2.3 to 2.7 GHz → Run at 2.7 GHz
- Justifications & Benefits:
  - Defaulting to 2.3 GHz is a reasonable setting
  - Policy offers an incentive to use EAS and to tune applications
  - Well-optimized applications will run at full speed

Conclusions

First implementation of Energy Aware Scheduling in a production environment!

The prediction quality of LoadLeveler is quite good!

We have come to a reasonable policy to trade off energy efficiency and performance!

Thanks & Kudos

- Co-Authors:
  - Raj Panda, François Thomas (IBM)
  - Arndt Bode, Matthias Brehm, Nicolay Hammer, Herbert Huber, Torsten Wilde (LRZ)
- Application Code Support:
  - Volker Springel & Team (MPI)
  - Ulrich Rüde & Team (Uni Erlangen)
  - Martin Käser & Team (LMU)
  - Carlo Cavazzoni & Team (CINECA)
- All other colleagues at LRZ & IBM for their valuable input & support