Resource and Job Management on HPC systems with Slurm

State of the Art

Yiannis Georgiou David Glesser

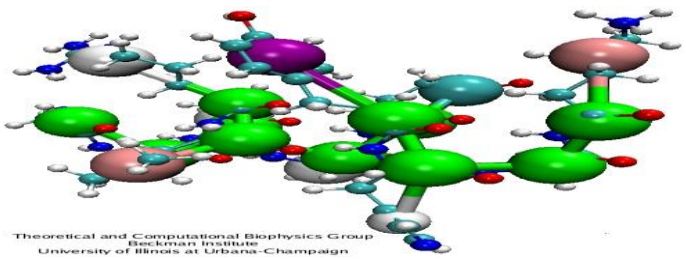
R&D Bull - Atos Technologies

IEEE Cluster

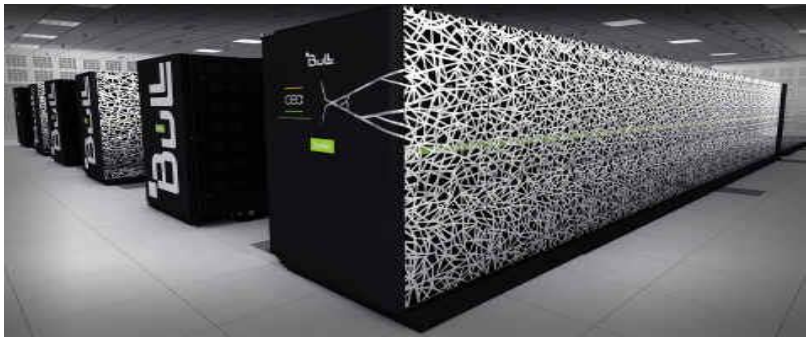
13-09-2016

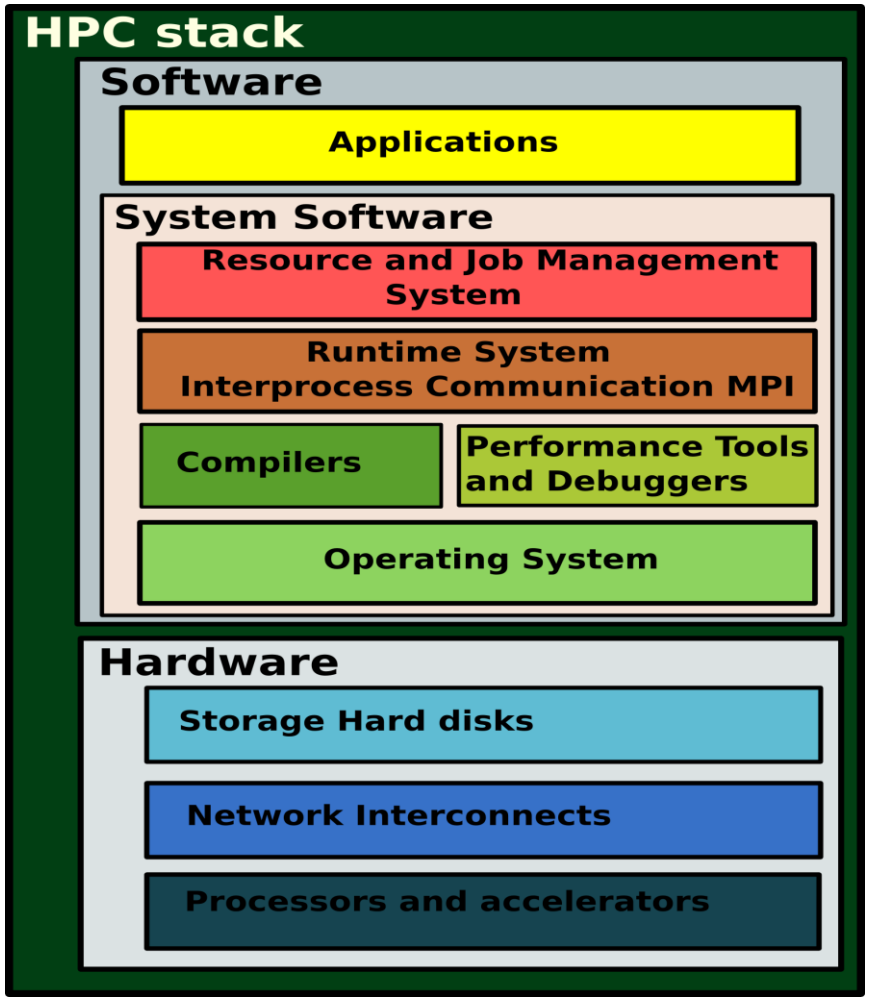
Bull atos technologies

High Performance Computing Systems



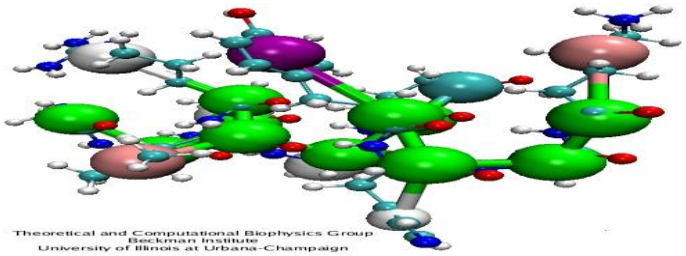
- **System Software:**
- Operating System, Runtime
- System, Resource Management,
- I/O System, Interfacing to External
- Environments



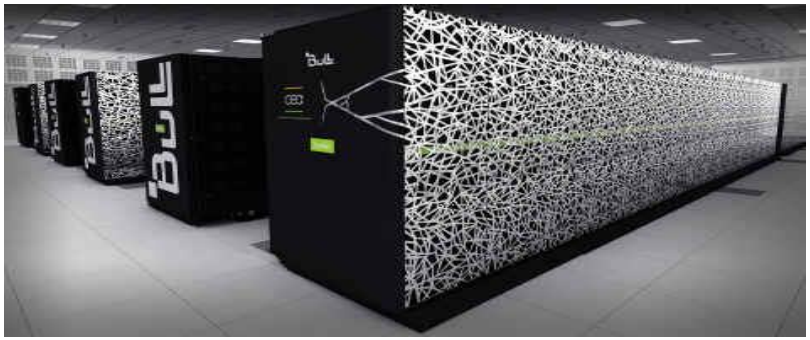


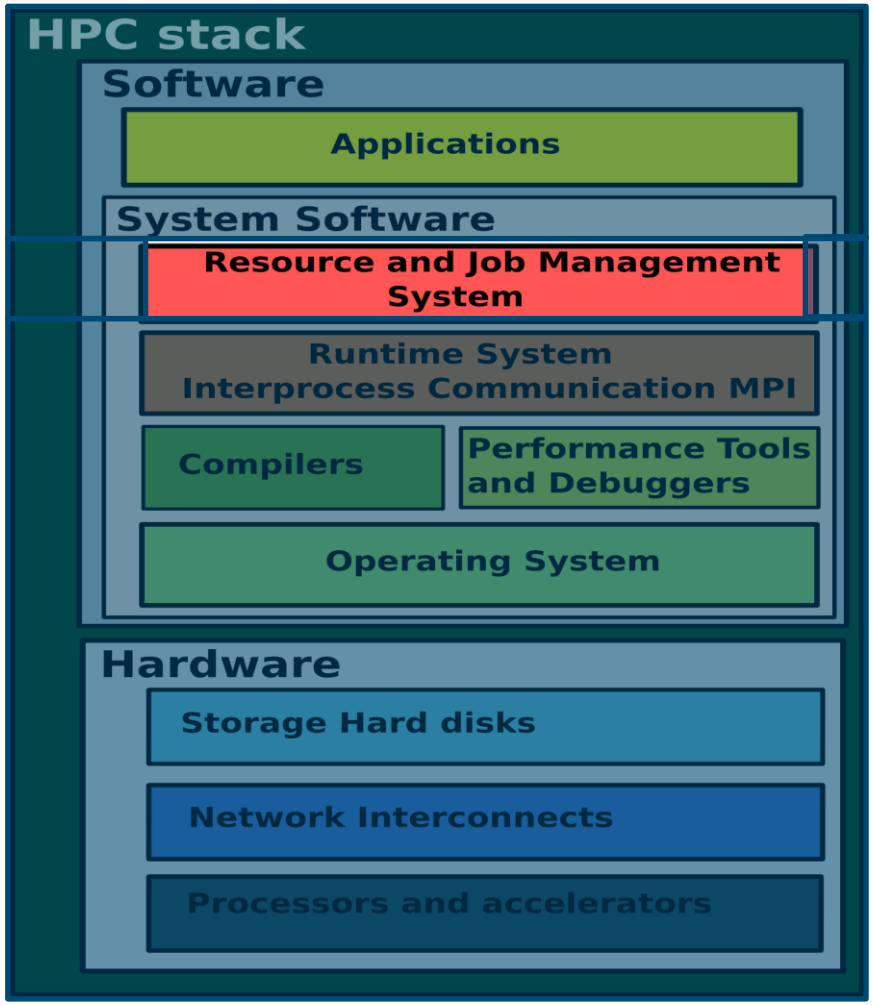
Bull
atos technologies

High Performance Computing Systems



- **System Software:**
- Operating System, Runtime
- System, Resource Management,
- I/O System, Interfacing to External
- Environments

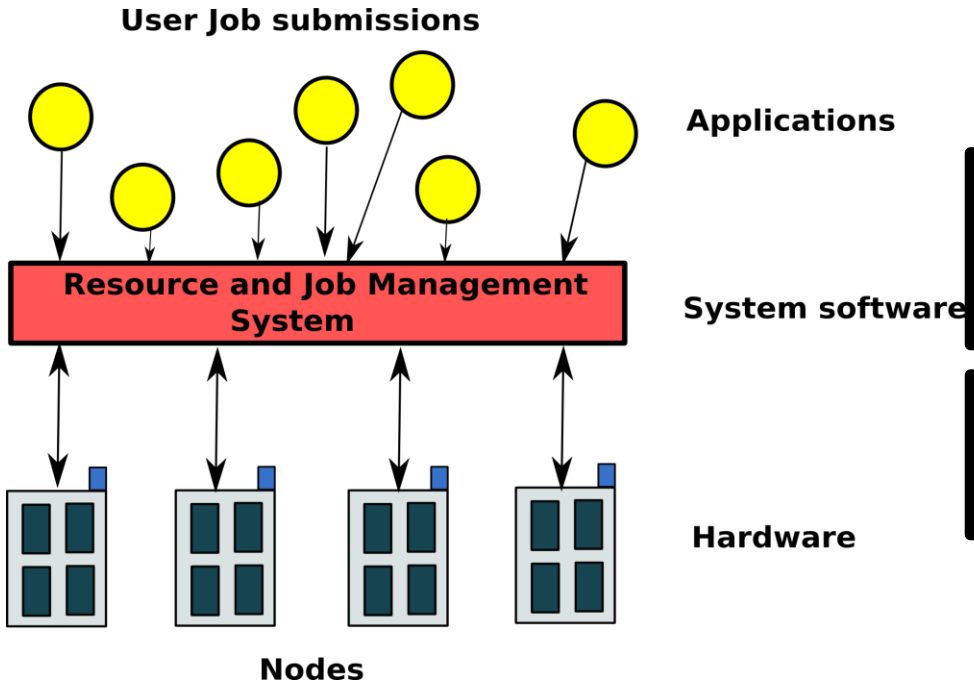




Bull
atos technologies

Resource and Job Management System

•The goal of a Resource and Job Management System (RJMS) is to satisfy users' demands for computation and assign resources to user jobs with an efficient manner.

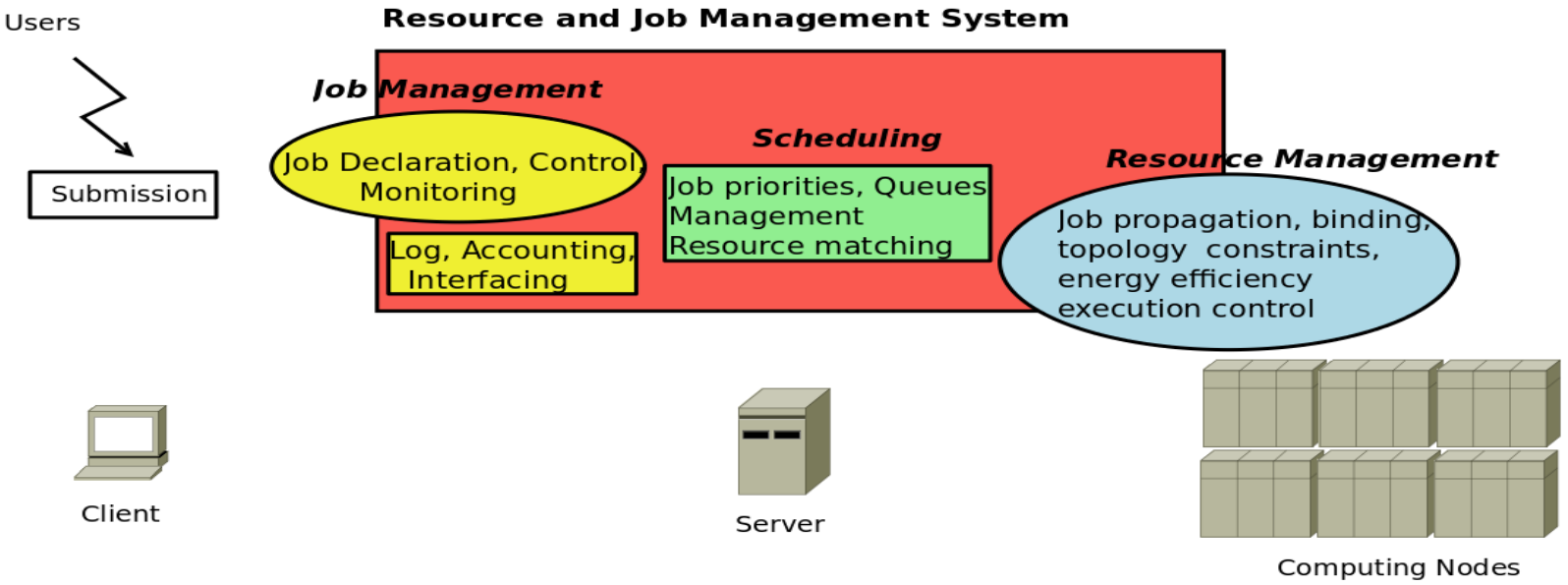


RJMS Importance
Strategic Position, responsibility for the overall system performance

Direct and constant knowledge of resources and application needs

Resource and Job Management System Layers

This assignement involves three principal abstraction layers:
- **Job Management:** declaration of a job and demand of resources and job characteristics,
- **Scheduling**: matching of the jobs upon the resources,
- **Resource Management :** launching and placement of job instances upon the computation resources along with the job's control of execution



SLURM scalable and flexible RJMS

- **Scalability**: Designed to operate in a heterogeneous cluster with up to tens of millions of processors.
- **Performance**: Can accept 1,000 job submissions per second and fully execute 500 simple jobs per second (depending upon hardware and system configuration).
- **Free and Open Source**: Its source code is freely available under the GNU General Public License.
- **Portability**: Written in C with a GNU autoconf configuration engine. While initially written for Linux, Slurm has been ported to a diverse assortment of systems.
- **Power Management**: Job can specify their desired CPU frequency and power use by job is recorded. Idle resources can be powered down until needed.
- **Fault Tolerant**: It is highly tolerant of system failures, including failure of the node executing its control functions.
- **Flexibility**: A plugin mechanism exists to support various interconnects, authentication mechanisms, schedulers, etc.

https://**github**.com/**SchedMD**/**slurm**

Bull
atos technologies

SLURM History and Facts

- Initially developed in LLNL since 2003, passed to SchedMD in 2011
- Multiple enterprises and research centers have been contributing to the project (LANL, CEA, HP, BULL, BSC, CRAY etc)
- Large international community, active mailing lists
  - Contributions (various external software and standards are integrated upon SLURM)
- As of the June 2016 **Top500** supercomputer list, Slurm is being used on **five of the ten** most powerful computers in the world including the **no2 system, Tianhe-2** with 3,120,000 computing cores.

- **Bull** does research, development and support of Slurm since 2005.
  - Provides Slurm as the default RJMS of the delivered platforms
  - All developed code dropped in open-source

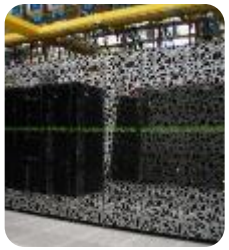
BULL current largest HPC supercomputers

CURIE - 2011
1st PRACE Petascale supercomputer
Intel E5 "Early Bird"
150 GB/s Lustre
2 PFlops peak

OCCIGEN 2015
TIER0 Supercomputer, CINES
DLC technology
2.1 PFlops peak
250 + 300GB/s – Lustre & DMF

TAURUS 2013-2014
1st BULL PetaFlops Supercomputer in Germany
1 PFlops peak
Lustre

DKRZ 2014-2016
Climate research
DLC technology
3 PFlops
45 PB @ 480 GB/s
Lustre + HPSS



CARTESIUS 2013-2014
1st Bull Petascale Supercomputer in Netherland
DLC technology
1.3 PFlops
8PB @ 220 GB/s - Lustre

HELIOS 2011-2014
ITER Community
1.7PFlops peak
X86 + PHI
+100GB/s – Lustre

BEAUFIX PROLIX 2013-2014-2015
1st Intel E5 v3 supercomputer in production ww
DLC technology
1 PFlops peak
Extension to 5 PFlops in 2016

SANTOS DUMONT 2015
Largest supercomputer in Latin America
DLC Technology
1 PFlops peak
Mobull

Bull
atos technologies

Bull sequana
the exascale generation of supercomputer

- **Open and modular platform designed for the long-term**
  - To integrate current and future technologies
  - Multiple compute nodes: Xeon-EP, Xeon Phi, Nvidia GPUs, other architectures…
- **Scales up to tens of thousands of nodes**
  - Large building blocks to facilitate scaling
  - Large systems with DLC: 250-64k nodes
- **Embedding the fastest interconnects**
  - Multiple Interconnects: BXI, InfiniBand EDR-HDR
  - Optimized interconnect topology for large basic cell / DLC (288 nodes)
  - Fully non-blocking within Cell
- **Ultra-energy efficient**
  - Enhanced DLC – up to 40°C for inlet water and ~100% DLC

Bull
atos technologies

Objectives

- Provide the material for a complete tutorial upon administration and usage of Slurm
  - Docker containers and step-by-step procedure for deployment of a small Slurm cluster on your PC.
  - Slides for installation, configuration and usage best-practices
  - Training through exercises
- State of the art on Resource and Job Management Systems and in particular on Slurm
  - Overview of our latest and ongoing research and development activities on the subject
  - Focus on Power Management, Energy Efficiency, Heterogeneous Resources, Performance Evaluation

Overview

- Tutorial: Material and First Steps
- Ongoing Works Towards Exascale
  - Heterogeneous Resources Support
  - Power/Energy Monitoring and Control
    - Measurement System, Energy Accounting and Power Profiling, User level control of power and energy
  - Power adaptive and Energy aware scheduling
    - User Incentives for energy aware scheduling, System-level control of power and energy, Power adaptive scheduling, Energy budget control

Slurm Administration and Usage Hands-On

Tutorial: Download code and Demo

Download from:
https://rjms-bull.github.io/slurm-tutorial

- Follow installation steps of a Slurm cluster on your PC based on Docker in github
- Start training following the tutorial's slides

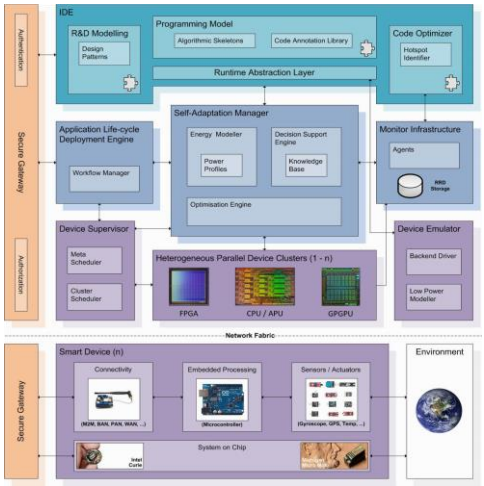
- Demo

Resource and Job Management State of the art

Support of Heterogeneous Resources and Power Management

Considering energy efficiency from code design to execution for heterogeneous architectures

- Atos/Bull leading the Tango project (Transparent Heterogeneous hardware Architecture deployment for eNergy Gain in Operation)
- Extension of currently available programming models and resource and job management systems to support complex heterogeneous architectures
- Code optimizer engine with the aim of optimizing code mapping. to reduce power consumption by the application.
- Power-awareness integrated in the whole software development optimization and execution process



Karim Djemame, Django Armstrong, Richard E. Kavanagh, Jean-Christophe Deprez, Ana Juan Ferrer, David Garcia Perez,Rosa M. Badia, Raúl Sirvent, Jorge Ejarque, Yiannis Georgiou: **TANGO: Transparent heterogeneous hardware Architecture deployment for eNergy Gain in Operation.** CoRRabs/1603.01407 (2016)

Slurm and SPMD support

- Slurm provides a SPMD (Single Program Multiple Data) environment.

  srun –N4 –Cgreen –gres=gpu myapp

- All nodes in an allocation have identical resources
  - 4 nodes are allocated, all have the feature green, all have a gpu

- All tasks execute the same application
  - myapp launched on all nodes.

Motivation for heterogeneous resources and MPMD

- In some cases it is desirable to have nodes with different characteristics as part of the same job execution.
  - A node with lots of memory for the serial startup/wrapup phase.
  - Lots of nodes with GPU for the parallel phase.
  - Nodes with Fast I/O to store the results.
  - And these nodes run different executables that are part of the same MPI_Comm_World
- Kind of like multiple sruns scheduled so the run at the same time.

Support for heterogeneous resources and MPMD

- The job specification language is extended to support the demand of heterogeneous resources
- New functionality to be available in Slurm version 17.02
- Support of MPMD - MPI with tight integration of srun
- Demo

Power/Energy Monitoring and Control

Power and Energy Management

- Issues that we wanted to deal with:
- Attribute **power and energy data** to HPC components
- Calculate the **energy consumption of jobs**
- Extract **power** consumption **time series of jobs**
- **Control** the Power and Energy usage of jobs

Power and Energy Measurement System

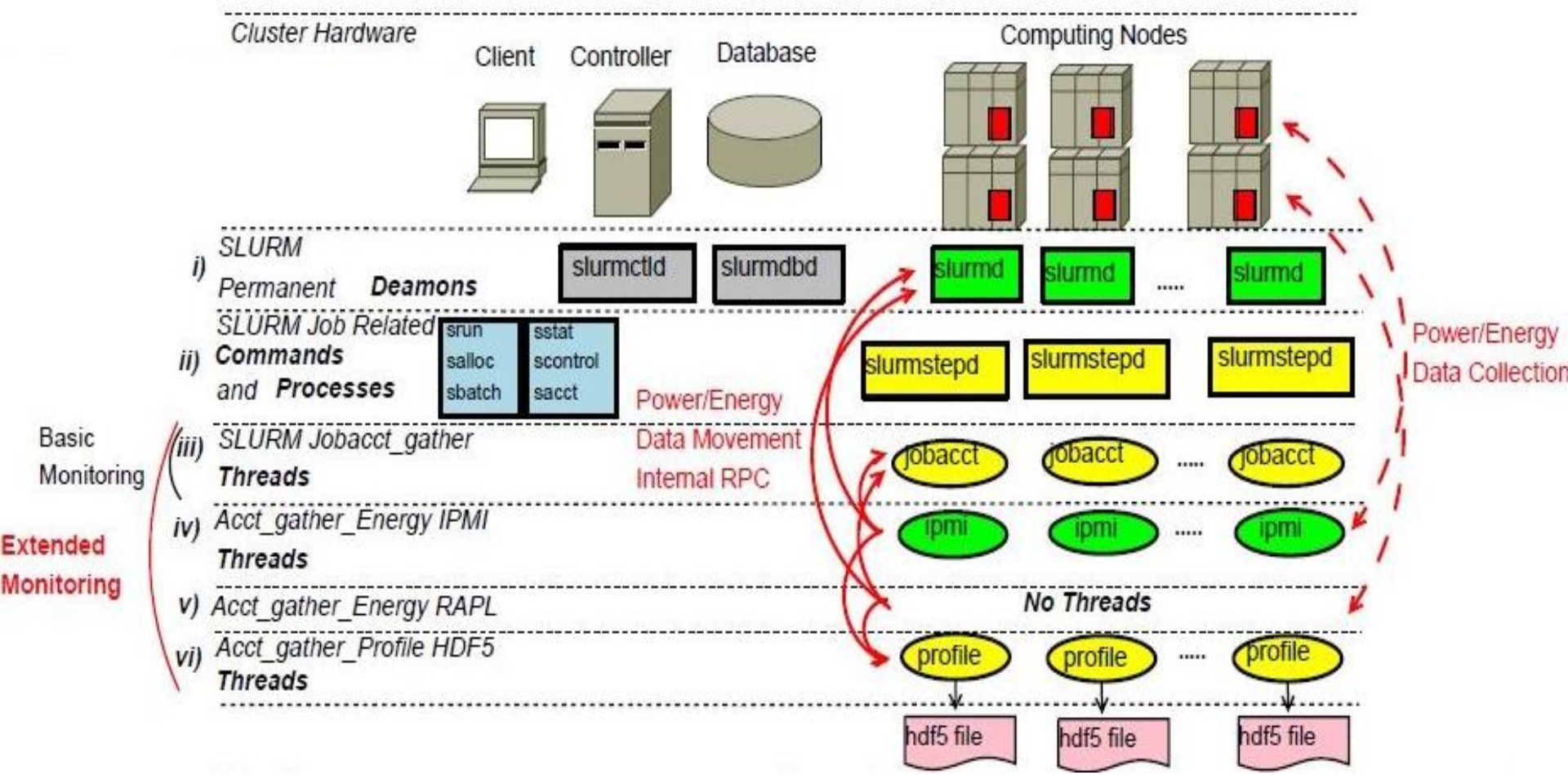
```
[root@cuzco108 bin]# $ scontrol show n=mo38 | grep ConsumedJoules
   CurrentWatts=105 LowestJoules=105 ConsumedJoules=17877

[root@cuzco108 bin]# sacct -o
"JobID%5,JobName,AllocCPUS,NNodes%3,NodeList%22,State,Start,End,Elapse
d,ConsumedEnergy%9"
JobID    JobName  AllocCPUS NNodes              NodeList      State
Start                End    Elapsed ConsumedEnergy
----- ---------- ---------- --- ---------------------- ---------- ----
--------------- ------------------ ---------- ---------
  127    cg.D.32         32   4     cuzco[109,111-113]  COMPLETED
2013-09-12T23:12:51 2013-09-12T23:22:03   00:09:12   490.60KJ

[root@cuzco108 bin]# cat extract_127.csv
Job,Step,Node,Series,Date_Time,Elapsed_Time,Power
13,0,orion-1,Energy,2013-07-25 03:39:03,0,126
13,0,orion-1,Energy,2013-07-25 03:39:04,1,126
13,0,orion-1,Energy,2013-07-25 03:39:05,2,126
13,0,orion-1,Energy,2013-07-25 03:39:06,3,140
```

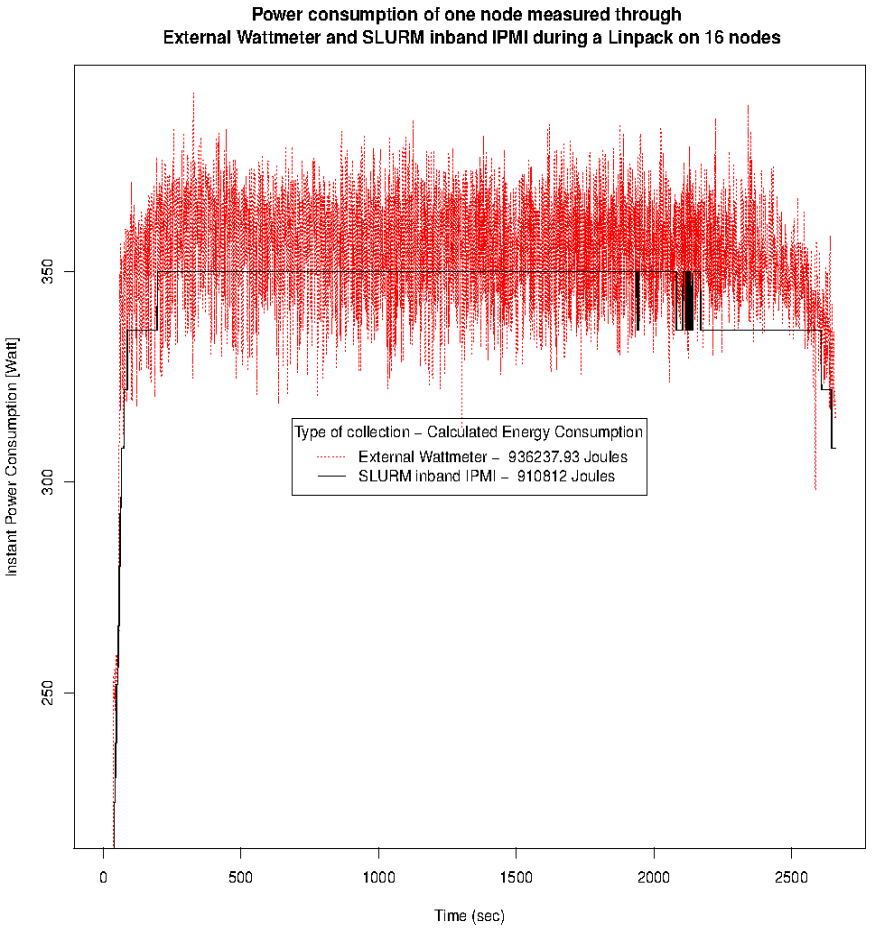

Yiannis Georgiou, Thomas Cadeau, David Glesser, Danny Auble, Morris Jette and Matthieu Hautreux
Energy Accounting and Control with SLURM Resource and Job Management System
(In proceedings of ICDCN 2014)

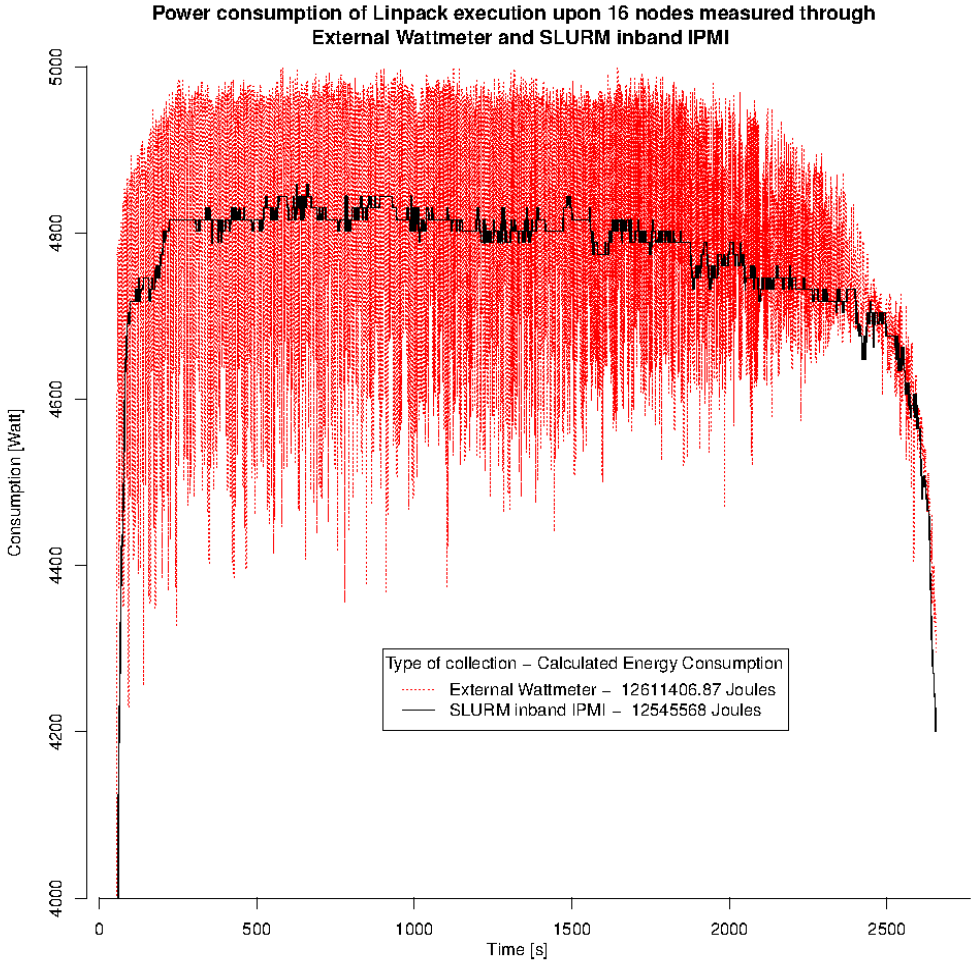
Energy Accounting and Power Profiling Architecture



Bull
atos technologies

Power and Energy Measurement System

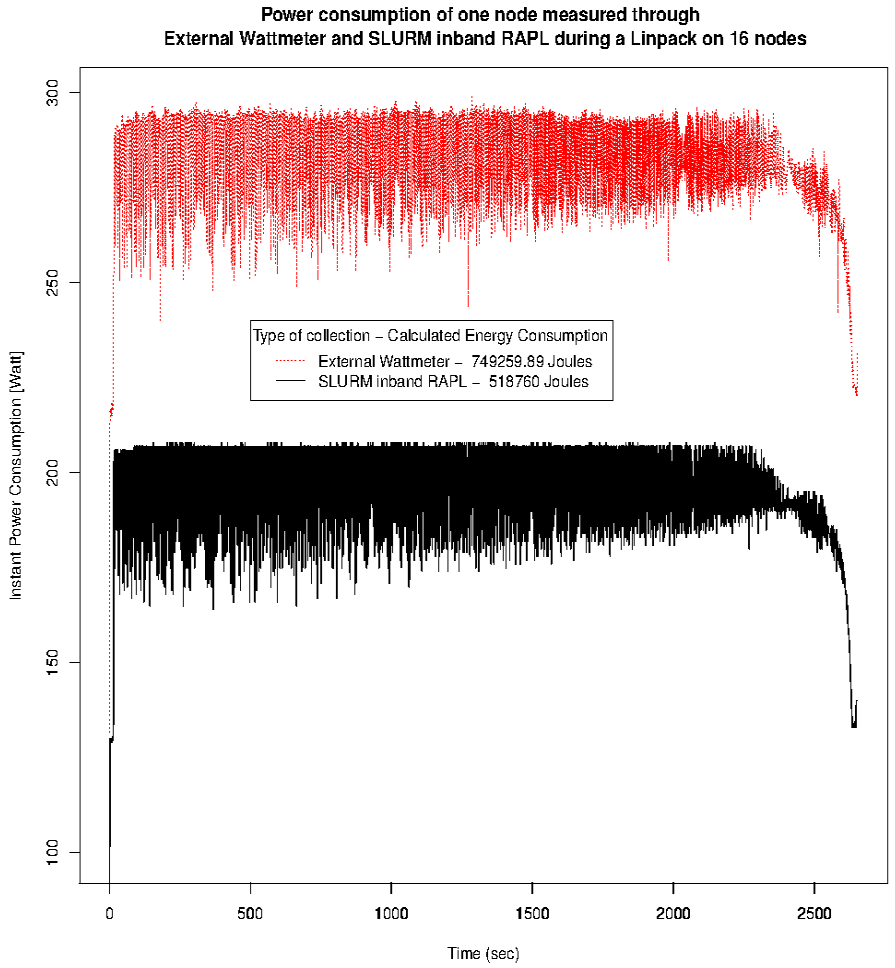


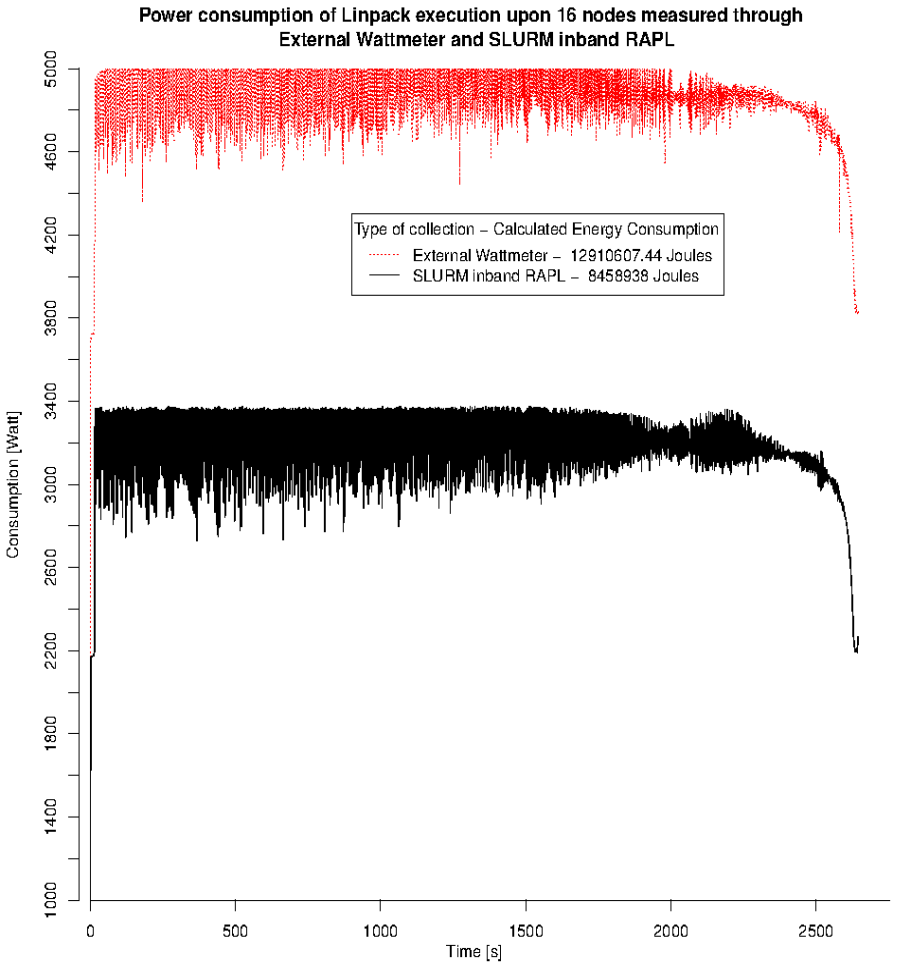


Yiannis Georgiou, Thomas Cadeau, David Glesser, Danny Auble, Morris Jette and Matthieu Hautreux
Energy Accounting and Control with SLURM Resource and Job Management System
(In proceedings of ICDCN 2014)

Bull
atos technologies

Power and Energy Measurement System



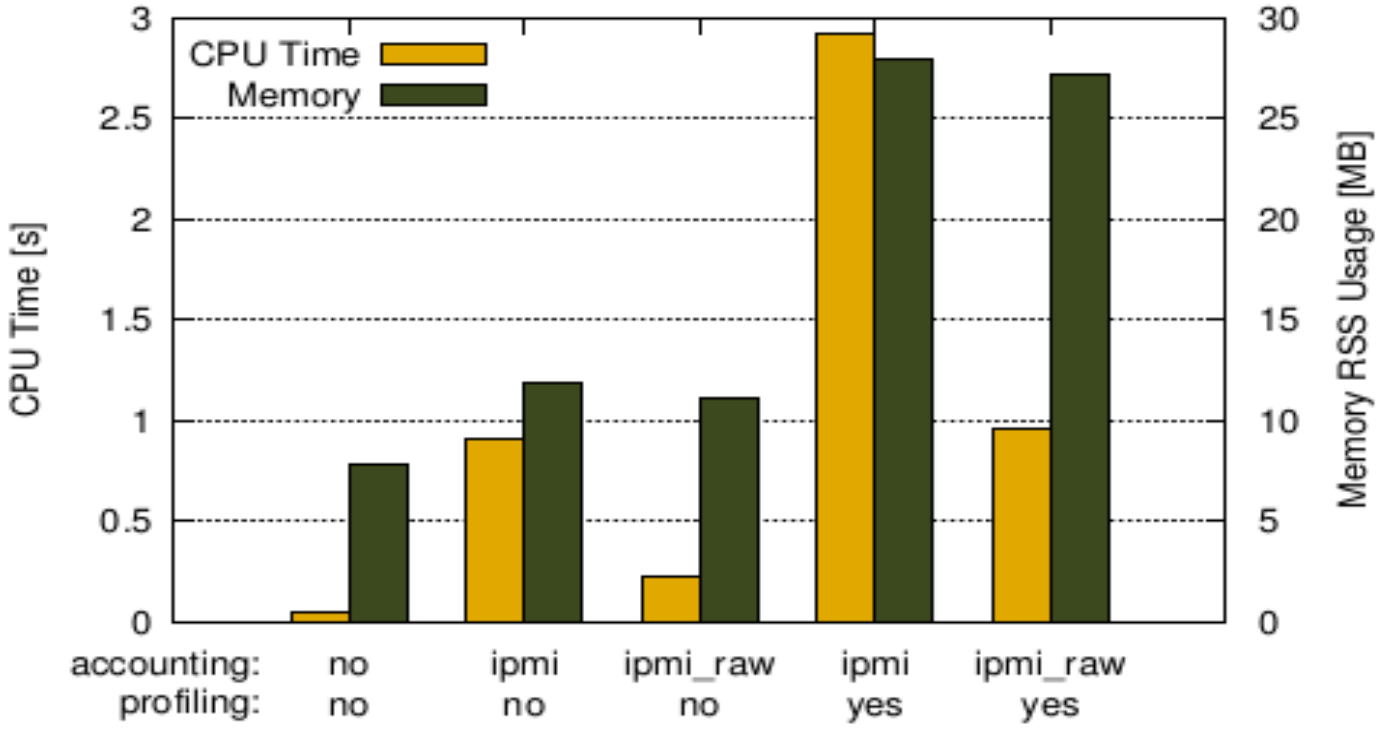


Yiannis Georgiou, Thomas Cadeau, David Glesser, Danny Auble, Morris Jette and Matthieu Hautreux
Energy Accounting and Control with SLURM Resource and Job Management System
(In proceedings of ICDCN 2014)

Bull atos technologies

Optimizations of Power and Energy Measurement System

- Based on TUD/BULL - BMC firmware optimizations
  - sampling to 4Hz
  - No overhead for accounting



HDEEM
High Definition Energy Efficiency Monitoring

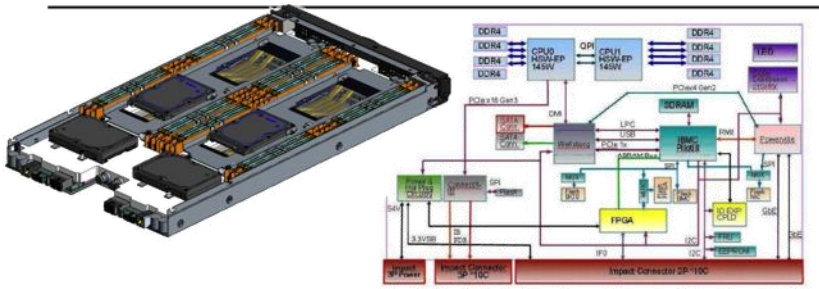
TECHNISCHE UNIVERSITÄT DRESDEN

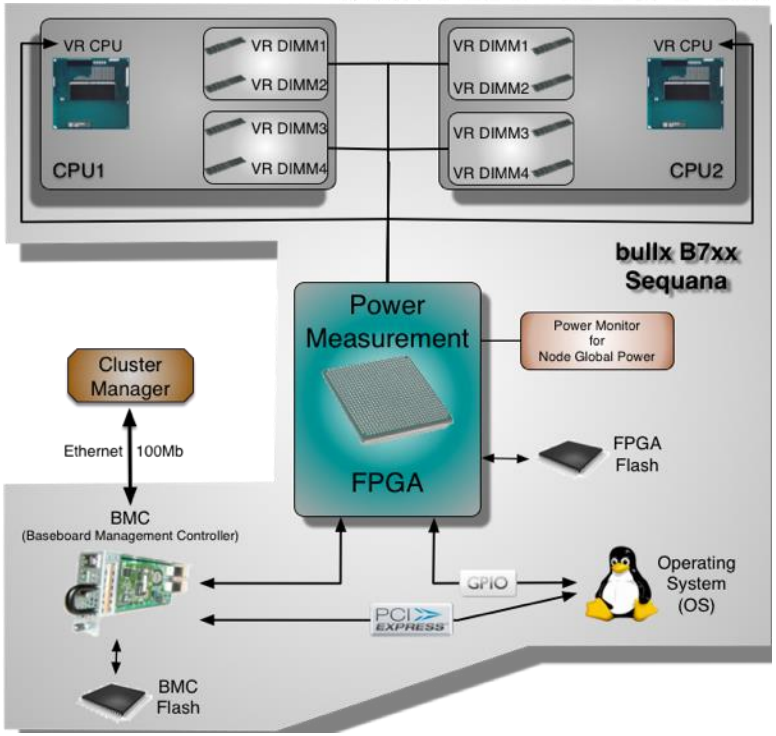
Daniel Hackenberg, Thomas Ilsche, Joseph Schuchart, Robert Sch¨ne, Wolfgang E. Nagel, Marc Simon, Yiannis Georgiou
HDEEM: High Definition Energy Efficiency Monitoring
In proceedings E2SC-2014

Bull atos technologies

FPGA for power measurement

HDEEM
High Definition Energy Efficiency Monitoring



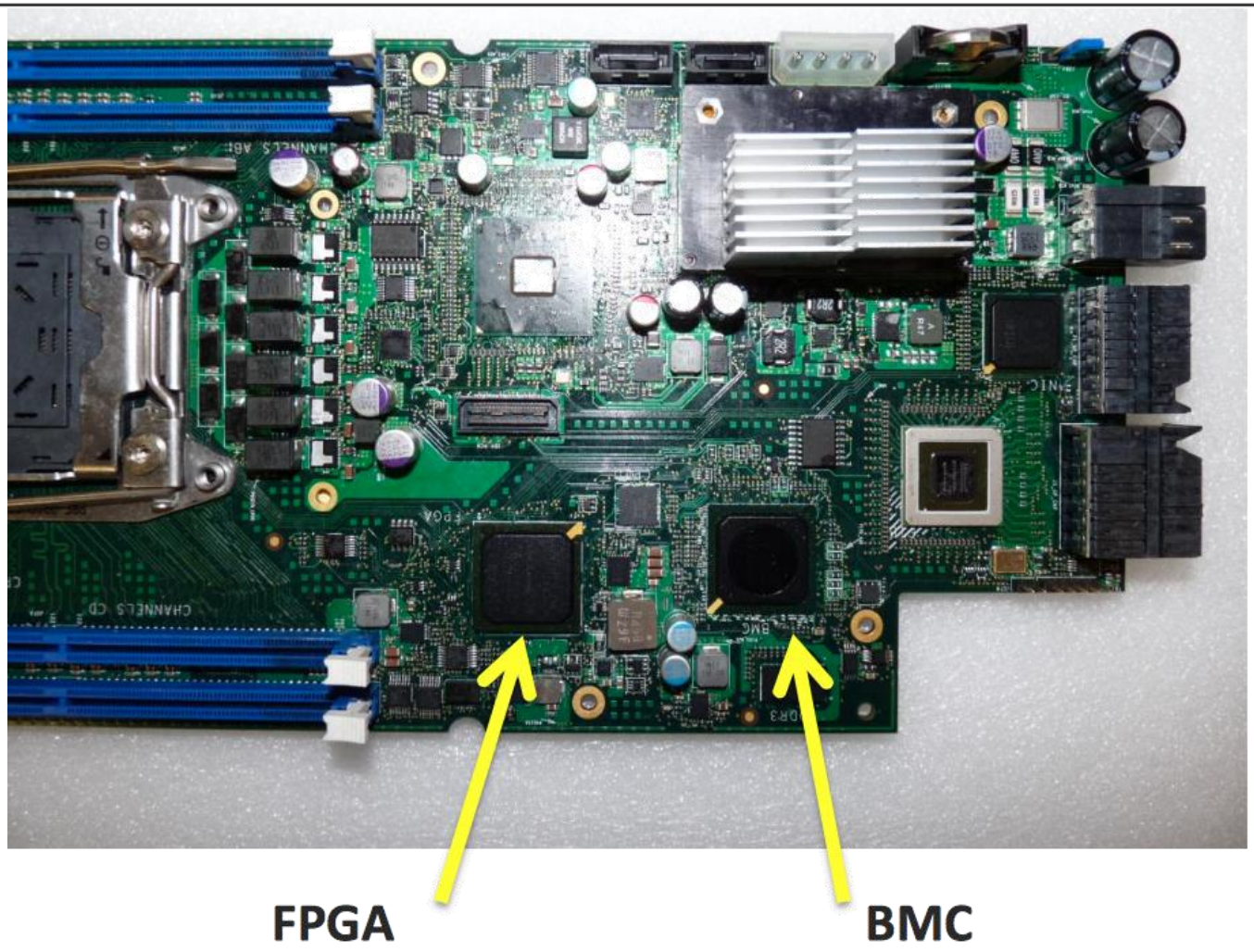


- On **bullx B7xx and Bull Sequana** platform a power measurement **FPGA** is integrated in each compute node
- Provides a sampling up to:
  - **1000 sample per second** for global power including sockets, DRAM, SSD and on-board
  - 100 sample per second for voltage regulators (VR) – 6 VR: one per socket + 4 for DRAM (one / 2 lanes)
- High accuracy with **2-3%** of uncertainty after calibration
  - 2% for blades
  - 5% for VR
- **Time stamped** measurements

Bull
atos technologies

Hardware implementation
bullx B7xx

HDEEM
High Definition Energy Efficiency Monitoring



Bull
atos technologies

Power and Energy through SLURM IPMI-RAW plugin

- High Definition energy efficiency monitoring based on new FPGA architecture supported through ipmi-raw
  - Improved accuracy for both power profiling per components (100Hz) and nodes (1000Hz)
  - Improved precision for energy consumption per job based on nodes (1000Hz) measurements
  - Decrease overhead on the application (CPU and Memory) since the collection is done internally within the FPGA
- To be released in upcoming slurm version

HDEEM
High Definition Energy Efficiency Monitoring

Bull
atos technologies

Accounting – Profiling
Support of multiple energy sensors

- Support for one sensor per node (until 14.11)

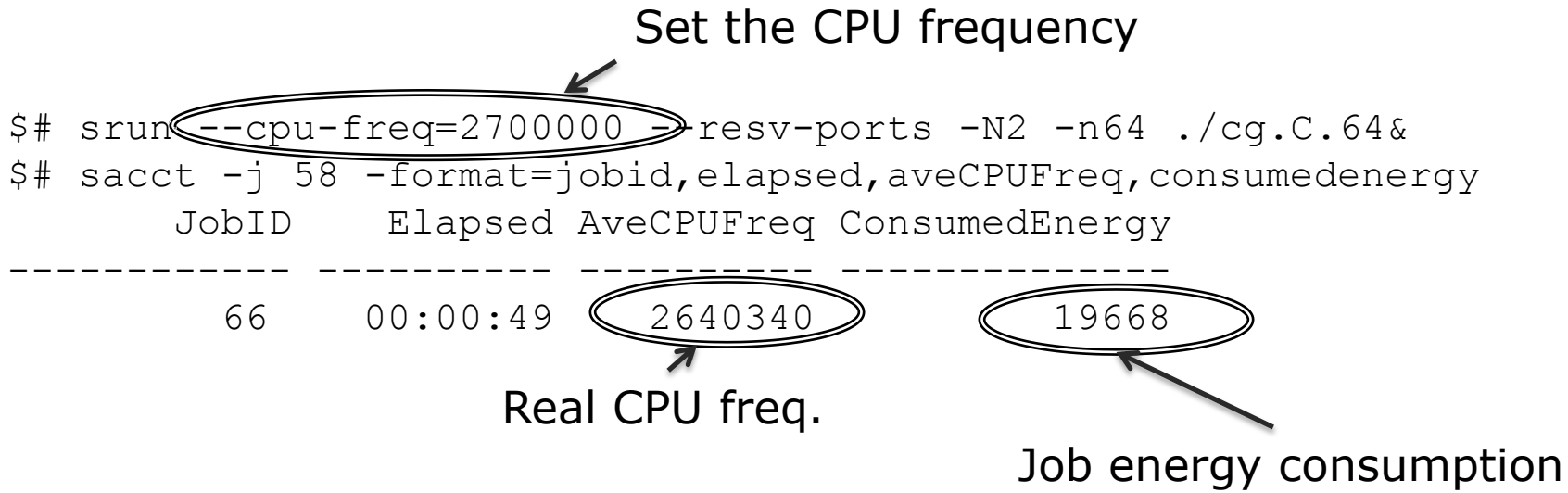
```
$ ipmi-sensors
62 | Power          | Current      | 175.80     | W          | 'OK'
```

- Support for multiple sensors per node (from 15.08)

```
$ipmi-sensors
85 | CPU0 Pwr           | Power Supply        | 10.00    | W    | 'OK'
86 | CPU1 Pwr           | Power Supply        | 6.00     | W    | 'OK'
87 | CPU0 DIM01 Pwr     | Power Supply        | 2.00     | W    | 'OK'
88 | CPU0 DIM23 Pwr     | Power Supply        | 0.00     | W    | 'OK'
89 | CPU1 DIM01 Pwr     | Power Supply        | 1.00     | W    | 'OK'
90 | CPU1 DIM23 Pwr     | Power Supply        | 0.00     | W    | 'OK'
91 | Blade Pwr          | Power Supply        | 112.00   | W    | 'OK'
```

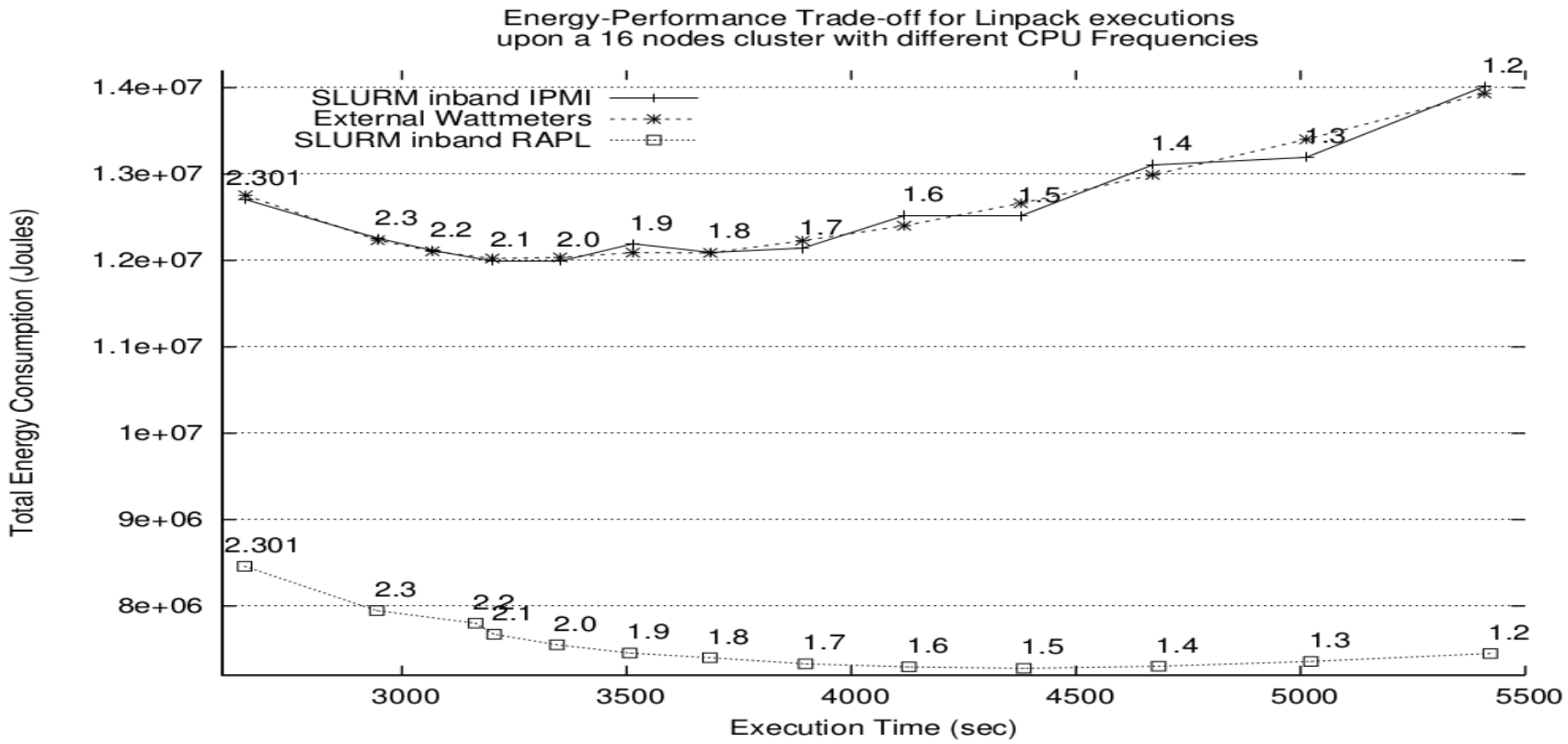
User-level control of power and energy through CPU Frequency setting parameter

- Job "--cpu-freq" option now supports minimum frequency (in addition to maximum frequency and governor) and supported for salloc and sbatch (for power adaptive scheduling)
- --cpu-freq =<p1[-p2[:p3]]>
  - p1 is current options or minimum frequency
  - optional p2 is maximum
  - optional p3 is scaling governor
- New configuration parameter "CpuFreqGovernors" identifies allowed governors



User-level: Find the good configuration…

- Using Slurm Energy Accounting to find the right trade off
- Specifying the optimal CPU using srun parameter



Bull
atos technologies

Power Adaptive and Energy aware Scheduling

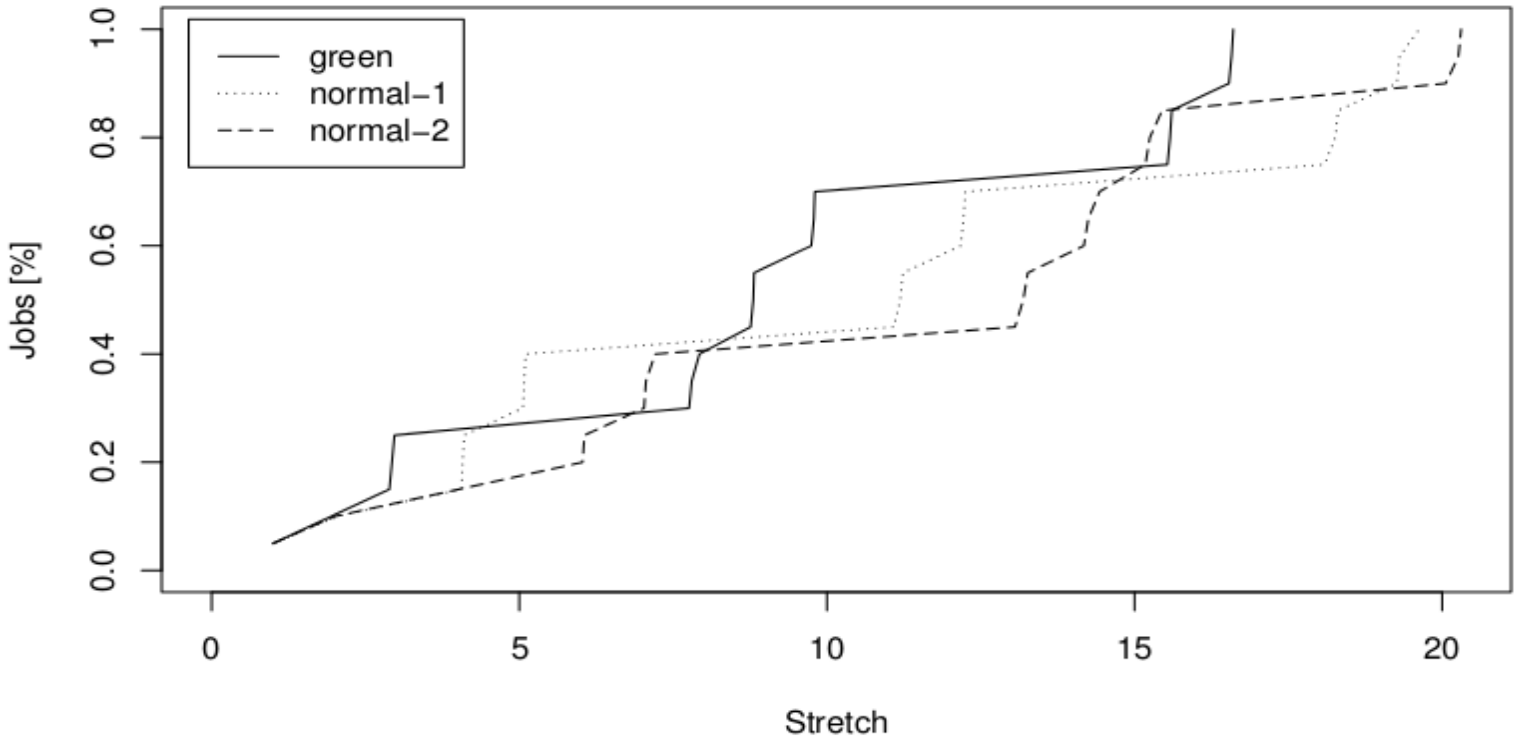
Energy Fairsharing

- Fairsharing is a common scheduling prioritization technique
- Exists in most schedulers, based on past CPU-time usage
- Our goal is to do it for **past energy usage**
- Provide **incentives to users** to be more energy efficient
  - Based upon the energy accounting mechanisms
  - Accumulate past jobs energy consumption and align that with the shares of each account
  - Implemented as a new multi-factor plugin parameter in SLURM

- Energy efficient users will be favored with lower stretch and waiting times in the scheduling queue

Energy Fairsharing

Cumulated Distribution Function for Stretch with EnergyFairShare policy running a submission burst of 60 similar jobs with Linpack executions by 1 energy-efficient and 2 normal users (ONdemand and 2.3GHz)



Yiannis Georgiou, David Glesser, Krzysztof Rzadca, Denis Trystram
A Scheduler-Level Incentive Mechanism for Energy Eciency in HPC
(In proceedings of CCGRID 2015)

Power adaptive scheduling

- The implementation appeared in Slurm v15.08 has the following characteristics:
- Based upon layouts framework
  - -for internal represantation of resources power
  - -Only logical/static represantation of power
  - -Fine granularity down to cores
- Power Reductions take place through following techniques
- coordinated by the scheduler:
  - –Letting Idle nodes
  - –Powering-off unused nodes (using default SLURM mecanism)
  - –Running nodes in lower CPU Frequencies (respecting –-cpu-freq allowed frequencies)

Set/Modify/View Powercap Value

- Initially with parameter in slurm.conf

```
[root@nd25 slurm]#cat /etc/slurm.conf |grep Power
PowerParameters=cap_watts=INFINITE
```

- Dynamically with scontrol update

```
[root@nd25 slurm]#scontrol update powercap=1400000
```

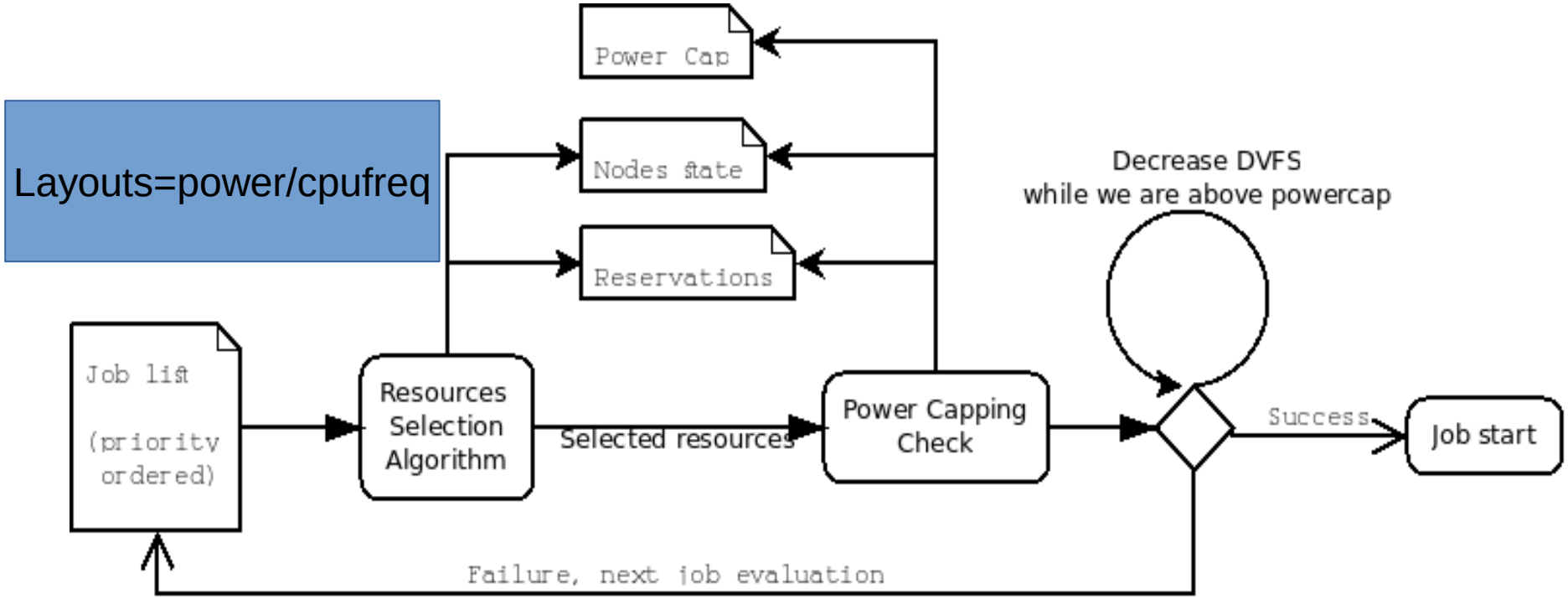
- In advance with watts reservation (scontrol create

```
[root@nd25 slurm]#scontrol create res FLAG=ANY_NODES starttime=now+11minutes
duration=16 Watts=532224 Users=root
```

- View with scontrol show

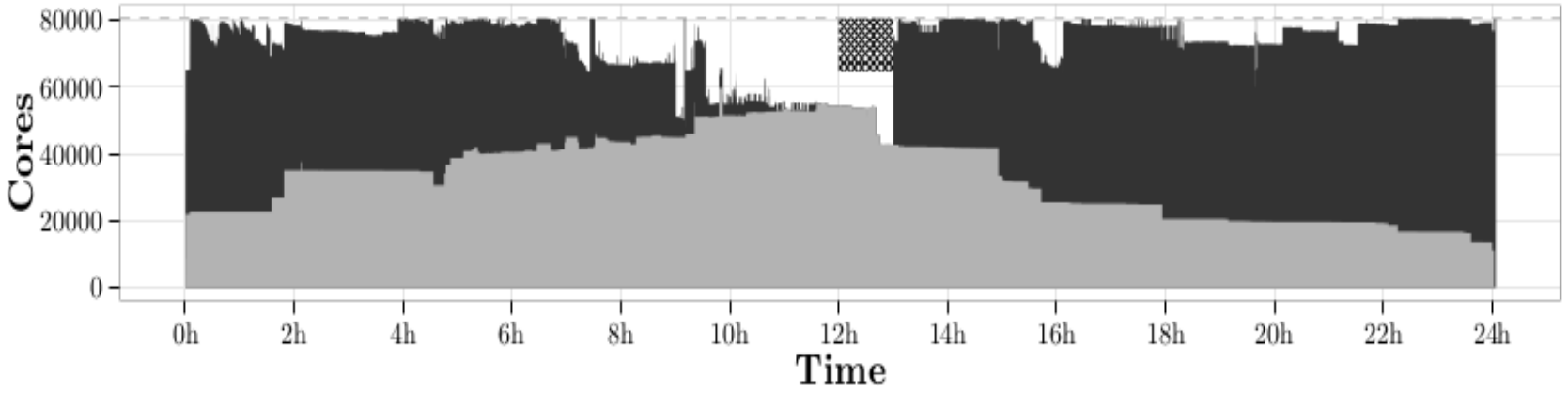
```
[root@nd25 slurm]#scontrol show powercap
MinWatts=564480 CurrentWatts=809934 PowerCap=INFINITE PowerFloor=0
PowerChangeRate=0AdjustedMaxWatts=1774080 MaxWatts=1774080
```

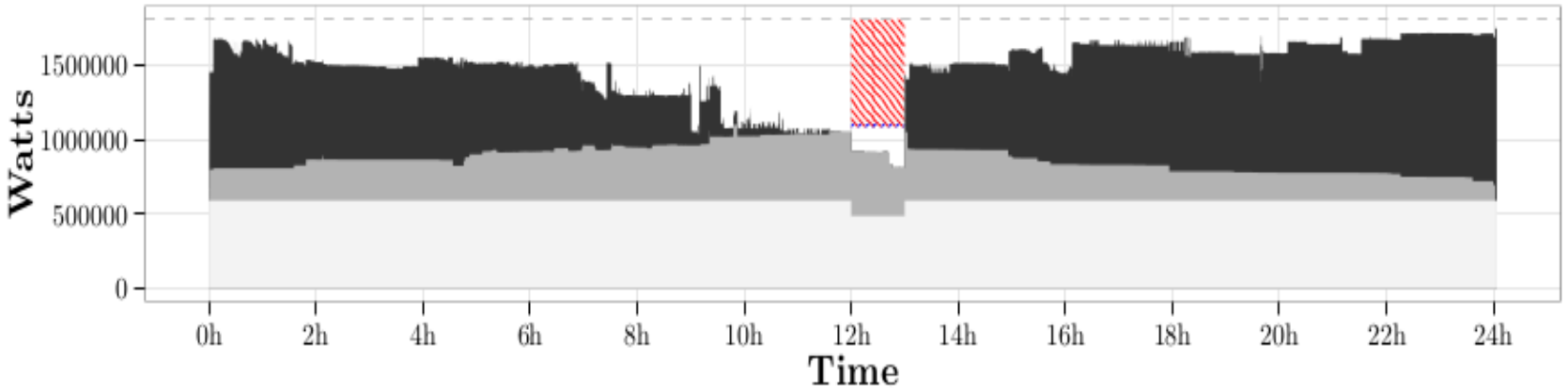
Power adaptive scheduling – algorithm extended version –



- Reductions through DVFS, idle and shut-down nodes (if power-save mode activated)
- Considering core level power consumption

Power adaptive scheduling

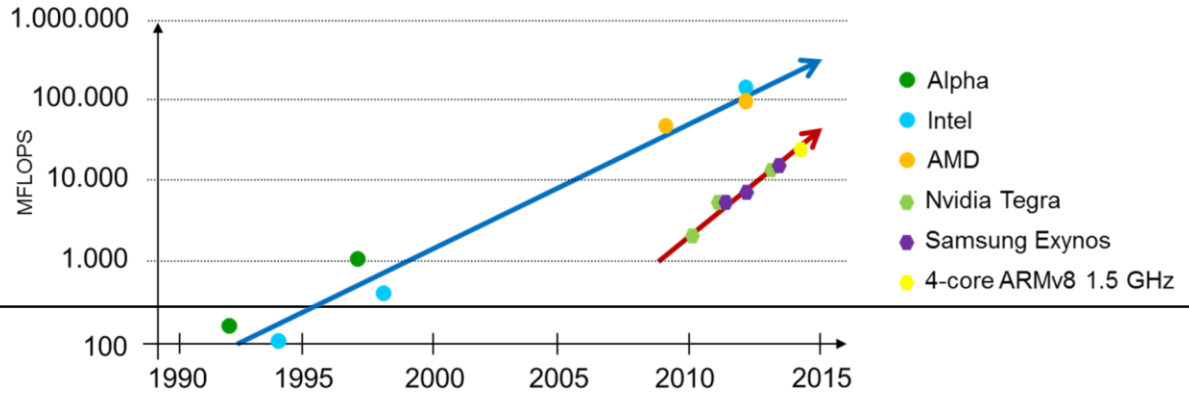




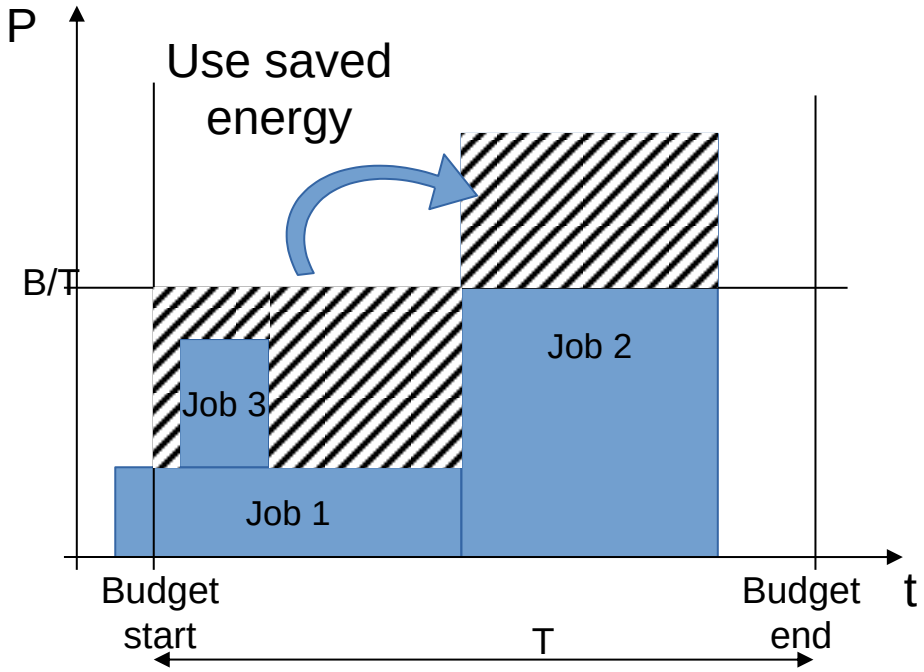
Yiannis Georgiou, David Glesser, Denis Trystram
Adaptive Resource and Job Management for limited power consumption
In proceedings of IPDPS-HPPAC 2015

MontBlanc project targets pre-exascale systems for 2020

- BULL leading the MontBlanc project
- European approach towards energy efficient high performance
- To design a well-balanced architecture and to deliver the design for an ARM based SoC or SoP (System on Package) capable of providing pre-exascale performance
- To introduce new high-end ARM core and accelerators implementations to efficiently support HPC applications.
- To develop the necessary software ecosystem for the future SoC.



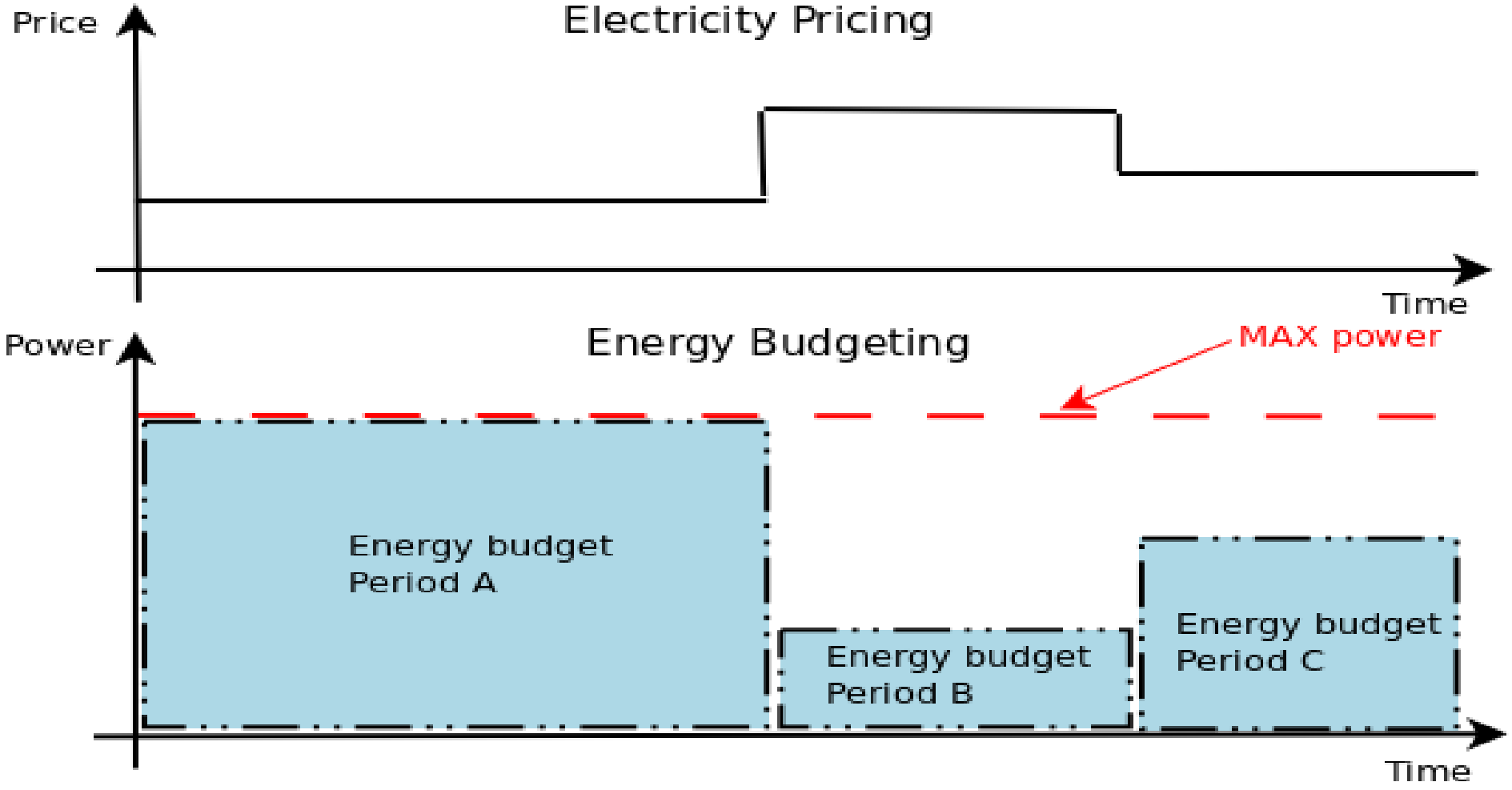
Energy budget



- Energy budget scheduling: **control the energy consumption** during a time interval
- Based on powercap and backfilling
  - Limit power consumption to respect budget
  - Backfill a job if it does not delay jobs with higher priorities
- Can be used with opportunistic shutdown mechanism to reduce consumption of non used nodes
- Use a basic model for energy prediction, the algorithm adapt itself in a second time to the real energy consumption

- Evaluated on a simulator, implementation in SLURM in progress
- For low to medium energy budget, the **performance/energy available increases**

Towards Energy Budget Control



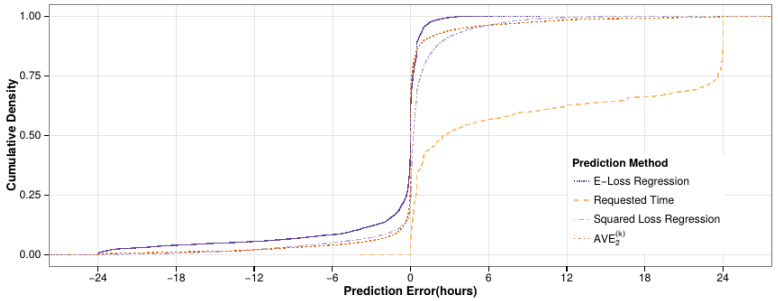
Bull
atos technologies

Helping the scheduler by analyzing user behaviors

- *Improving Backfilling by using Machine Learning to Predict Running Times*
- by Gaussier, Glesser, Reis, Trystram

- A machine learning algorithm that learns running times of jobs in order to improve the scheduling algorithm

- Users have in average
- thier results 28% sooner
- than with actual used
- algorithms



- Next steps:
  - Use Monte Carlo Tree Search to take (some) scheduling decision
  - Modelize user behavior to help scheduling and administrators

Simunix, a platofrm simulator

- Objective: **simulate an RJMS** (like Slurm)

- Existing simulator are
  - Too far from reality
  - Intricated within one specific version of a specific RJMS
- Idea: **simulate the underlying platform** and then **run the real code**

- **Intercept** blocking and platform related calls
  - pthread_mutex, gettimeofday, recv, select, poll…
  - To do this: modify dynamically the GOT (where the linker store the address of shared functions)
  - Redirect the calls to calls of our simulator

- We use Simgrid a **simulator framework** for distributed software

- Work in progress (we can simulate basic C programs but not Slurm (1M LoC…))

Thanks

yiannis.georgiou@atos.net
david.glesser@atos.net

Bull
atos technologies