EAR
Energy Aware Runtime

Energy monitoring and optimization made easy and accessible — powered by EAR

Julita Corbalan BSC team leader/ EAR CTO
julita.Corbalan@bsc.es
julita.corbalan@eas4dc.com, raphael.brochard@eas4dc.com

Agenda

- Why Energy efficiency in DC is needed
- The EAR roadmap for the ultimate goal of energy efficient data centers
  - Data Center Monitoring and Management
  - Dynamic Energy optimization
  - Node sharing support for big servers
  - Smart powercap
- Useful links

Why do we need energy efficient Data Centers

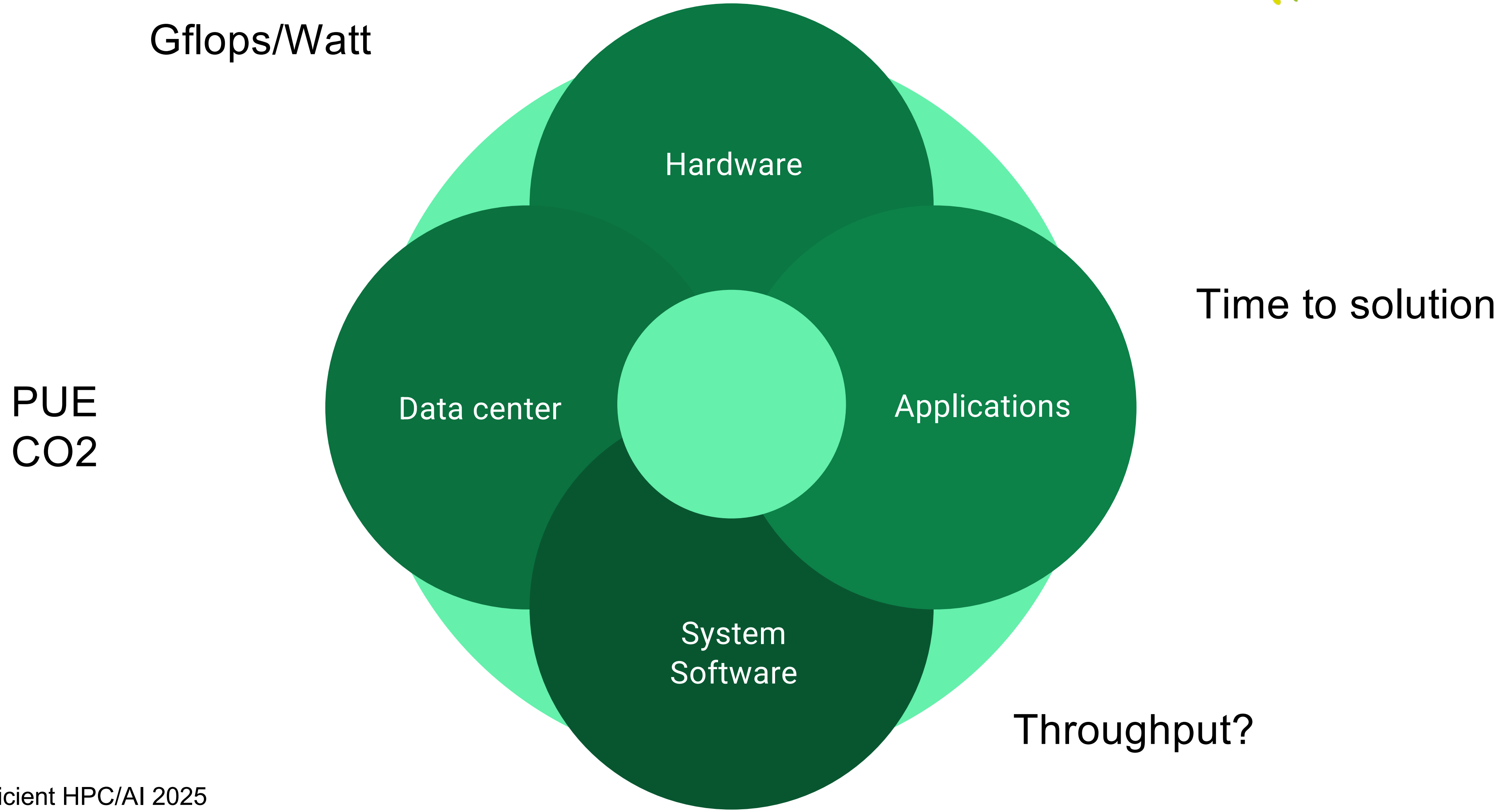
Power is a limited resource

- Massive increase in power requirements
  Real World Example:
  - Initial power available from public grid 8 MW
  - 100k H100 GPUs @700W each 70 MW
- Total data center power estimate ~150 MW
- Based on some sources (internet ☺)…
  - 2%- 3% of annual increase if we have a "eco" mind improving hardware and software technologies
  - 5% - 7% of annual increase if the utilization of AI , edge etc increases its utilization without the equivalent improvement in the technology efficiency

Is it only a matter of money? Do the market will regulate the “efficiency”?

- Can we avoid people using data centers , AI chats, cryptocurrencies, etc? Can we control?
    - Not
- What can we do then???
    - **DO OUR BEST !!!**

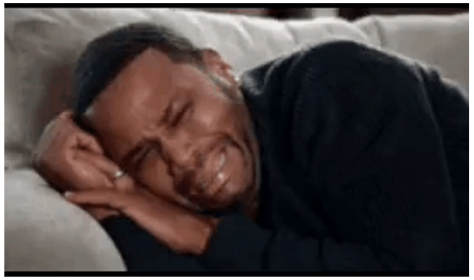
Energy "efficiency" in Data centers



Application efficiency: Power/Energy

- What about power
  - Much less expertise and tools since power is mostly a privileged metric
  - Many different APIs, more standard at the CPU level but anyway complex
  - **Different sources of power**
- What about new use cases
  - Workflows
  - Node sharing
  - Virtualization

And we have not commented yet about evaluation metrics!



System software for energy efficient Data centers

- System software tools and service helping application developers to improve their codes
  - Static optimization
    - Energy
    - **Resources (num nodes)… and then energy**
  - Dynamic optimization
    - Energy
    - **Resource (power/frequency) ….. and then energy**

Stakeholders perspective on Energy Optimization

- Users
  - Motivation: scientific results , **is that enough??**
  - Challenges:
    - Understand application behavior and actual impact (**metrics**)?
    - Limited skills/time/opportunity for optimizations (**transparent**)?

- Admins
  - Motivation: **keep the system running under the limits**
  - Challenges:
    - Cannot support individual users with energy optimization efforts (**automation**)
    - Operate the system within data center power limits (**hard power capping**)

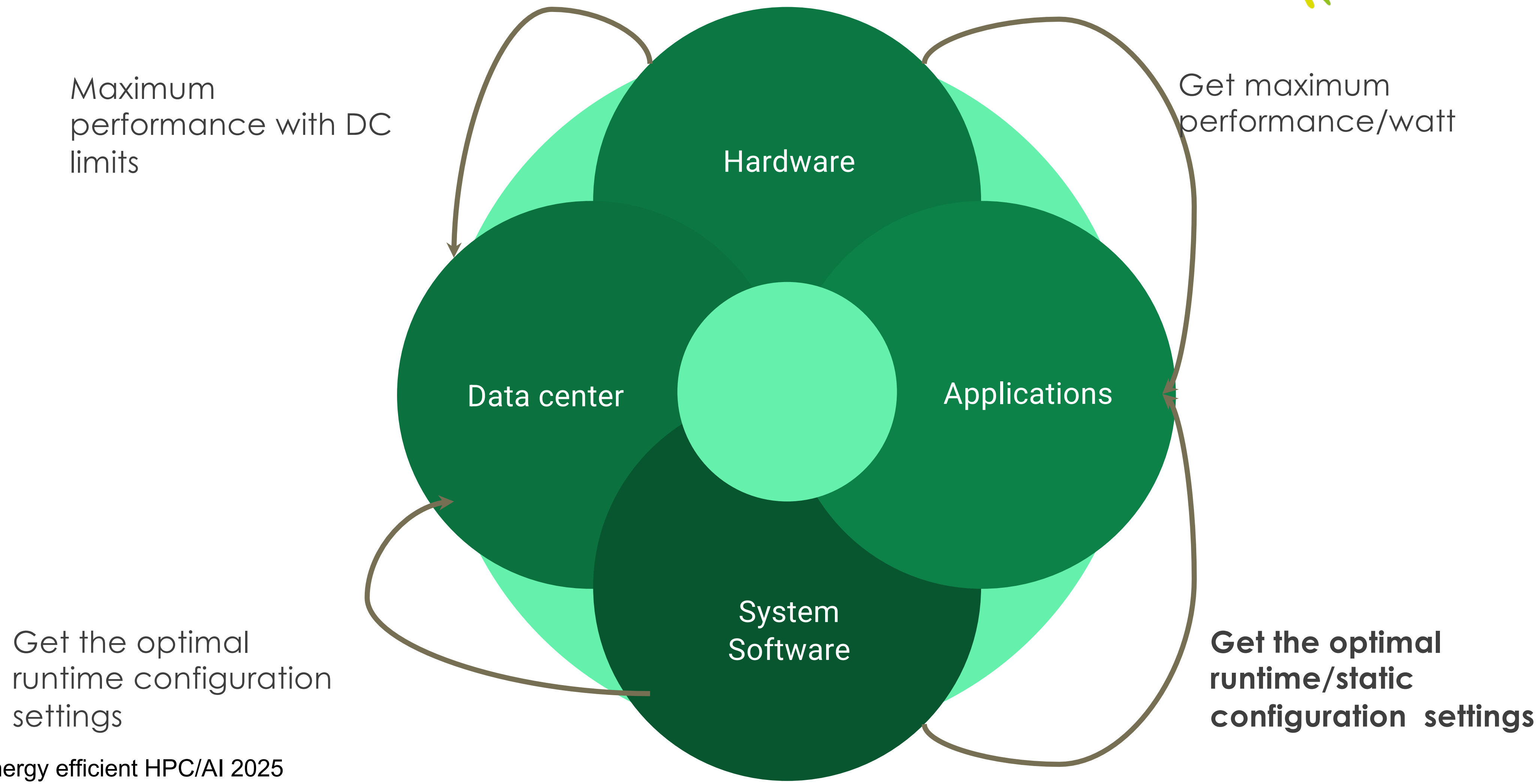
- Data center management
  - Motivation: **maximize work output within CAPEX/OPEX budgets**
  - Challenges:
    - Understand the workloads running in the data center (**analysis**)
    - Control energy budget (**energy optimization**)
    - Charge users for energy consumption (**accounting**)

Stakeholders perspective on Energy Optimization: Main challenges

- Users
  - Some users has expertise on resource optimisation
  - Others not
  - Most of them doesn't have on energy optimization
  - **So what is their benefit for being "energy efficient" ?? → It must be easy!**
- Admins
  - **For them is another tool , so another source of problems → It must be easy!**
- Data center
  - Reduce the electricity consumption and works under the power limits → **It is a must!**

It is pending to see how traditional scheduling policies (CPU hours) are replaced by KWH or Joules or GFLOPS/W targets or ....., energy efficiency metric, and the effect on users

Collaborative work for Energy efficient DC



Monitoring

- **Data Center monitoring**
  - Computational and Non-computational devices
  - Different granularities: ms… hours
  - Different sources of data: In-band, out of band, exporters, etc
  - Different scales: few nodes/devices…. Thousand
- **Application monitoring**
  - Different that jobs, fine grained: 1 job multiple applications
  - Metrics : CPU, GPUs, FPGAs, etc
  - Programming languages and frameworks
  - Schedulers
  - Virtualization
  - others…

Optimization

- Applications and system can be dynamically configured to adapt to the running ***workload***
- **What is a workload**? It depends on the context
  - Run vs Idle period → Select the best CPU governor. AFAIK there is nothing similar on the GPU
  - Running an application → **Select the best CPU/Memory/GPU frequency (and governor) resulting in the maximum performance/watt**
  - Running a workflow → Select the best per-application configuration taking into account the whole workflow
  - Data Center workload → Update the system optimization policy (and policy arguments) to maximize DC goals or constraints

Power/thermal/Cooling management

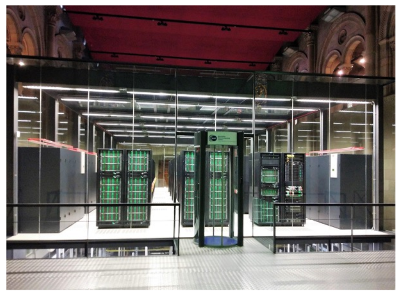
- Optimization can be seen as a ***ratio*** , power, thermal, cooling management is about infrastructure limits
  - Limits than cannot be exceeded
- This technology has been historically provided by Vendors since it is a “hardware” issue
- Several strategies
  - Preventing problems → limit always set by hardware
  - Predicting/Detecting → There is a risk
  - Dynamic settings → Adapting settings to the workload
  - ***Combining Prediction & Dynamic management***

The EAR roadmap to energy efficient Data Centers

Energy efficient HPC/AI 2025

EAR and the Stakeholders

- Users
  - **Automatic (easy)**
  - **Portable**, it's worth to do the effort of using EAR tools
  - HPC and AI frameworks, programming models etc supported
- Admins
  - It's a new tool. Yes, but very **similar to existing tools** (SLURM)
  - Very few dependencies
  - Powerful deployment and support for heterogeneous systems
- Data center management
  - Power management
  - Energy optimization
  - Non-computational devices(WIP)



EAR : System software for Energy efficiency

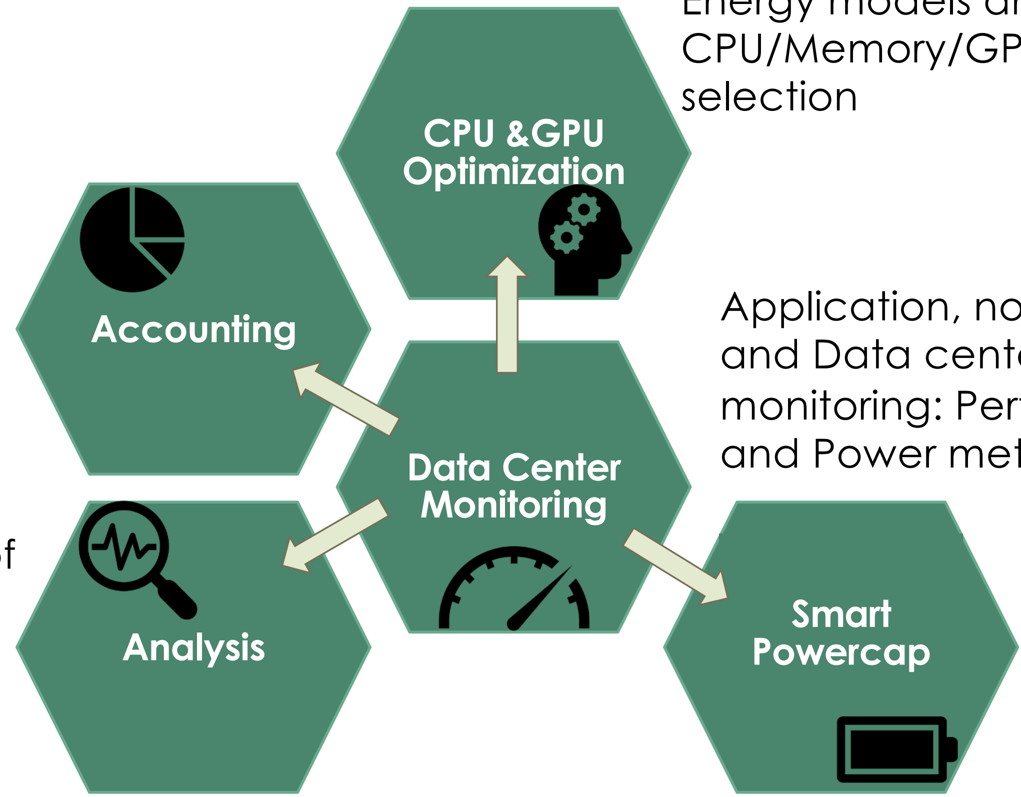
Energy models and policies for CPU/Memory/GPU frequency selection

Data reporting for accounting, billing, visualization and analytics

Application, node , cluster and Data center monitoring: Performance and Power metrics

Analysis and classification of application metrics for energy and power optimization

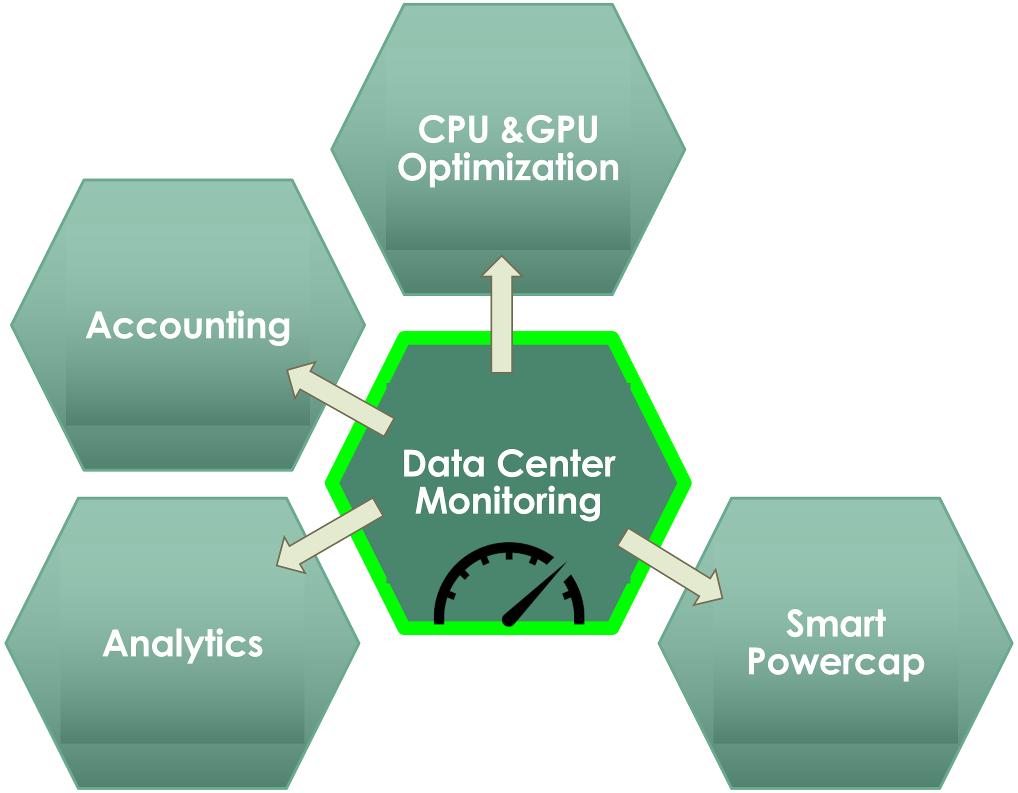
Power control to guarantee data center operational limits



Data Center Monitoring and Optimization

- Monitoring as source for
  - Optimizing
  - Management

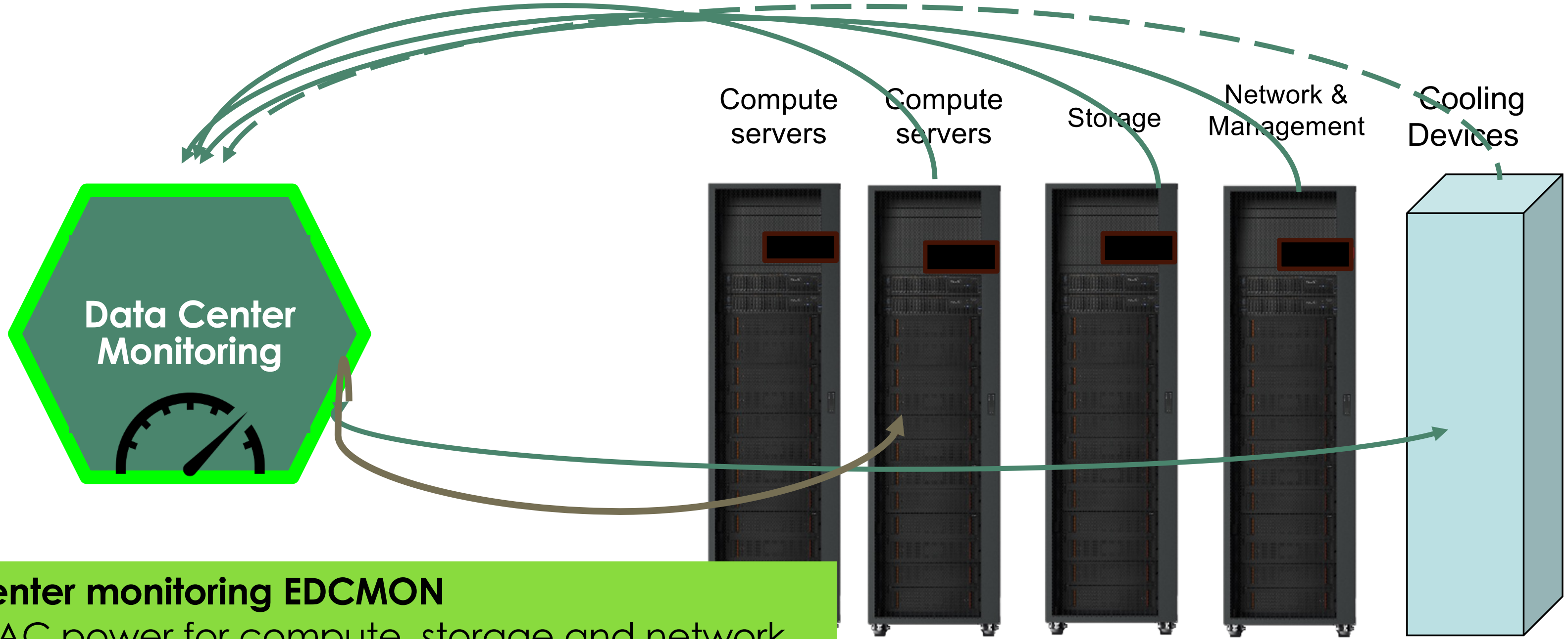
- Compatible with other tools
  - Sharing data
  - European software stack



EAR Energy Aware Runtime

REGALE project

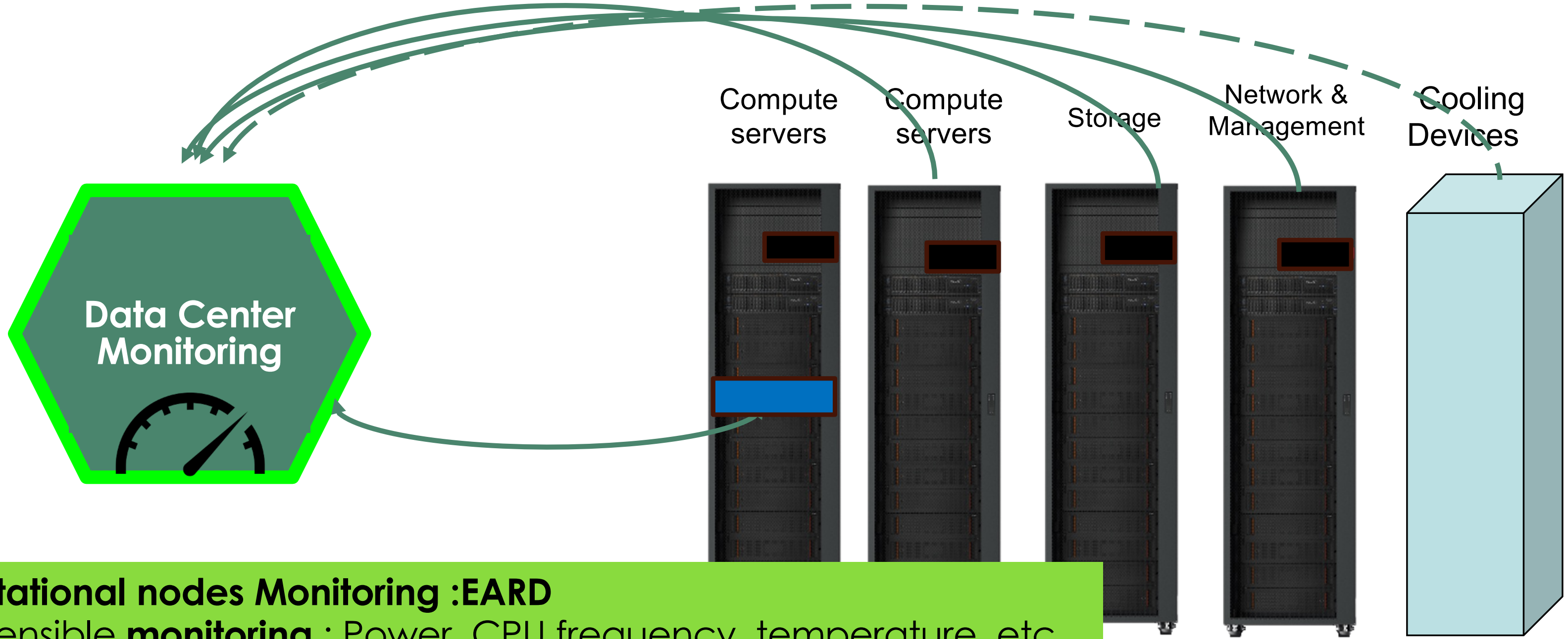
Rack, Node, Application Monitoring



- **Data Center monitoring EDCMON**
  - AC power for compute, storage and network through the PDUs
  - Report to DB
  - Visualization
  - Integration with EAR powercap service(WIP)

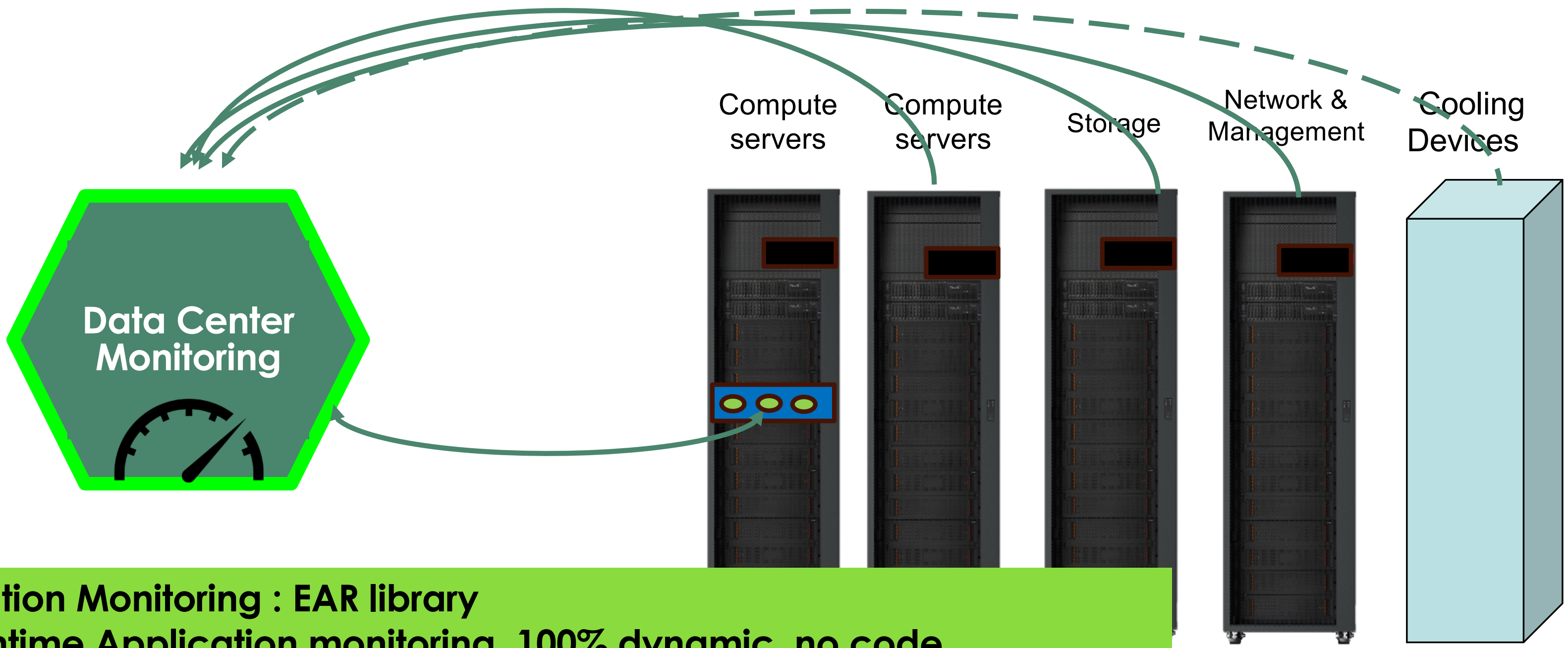
Rack, **Node**, Application Monitoring

EAR Energy Aware Runtime



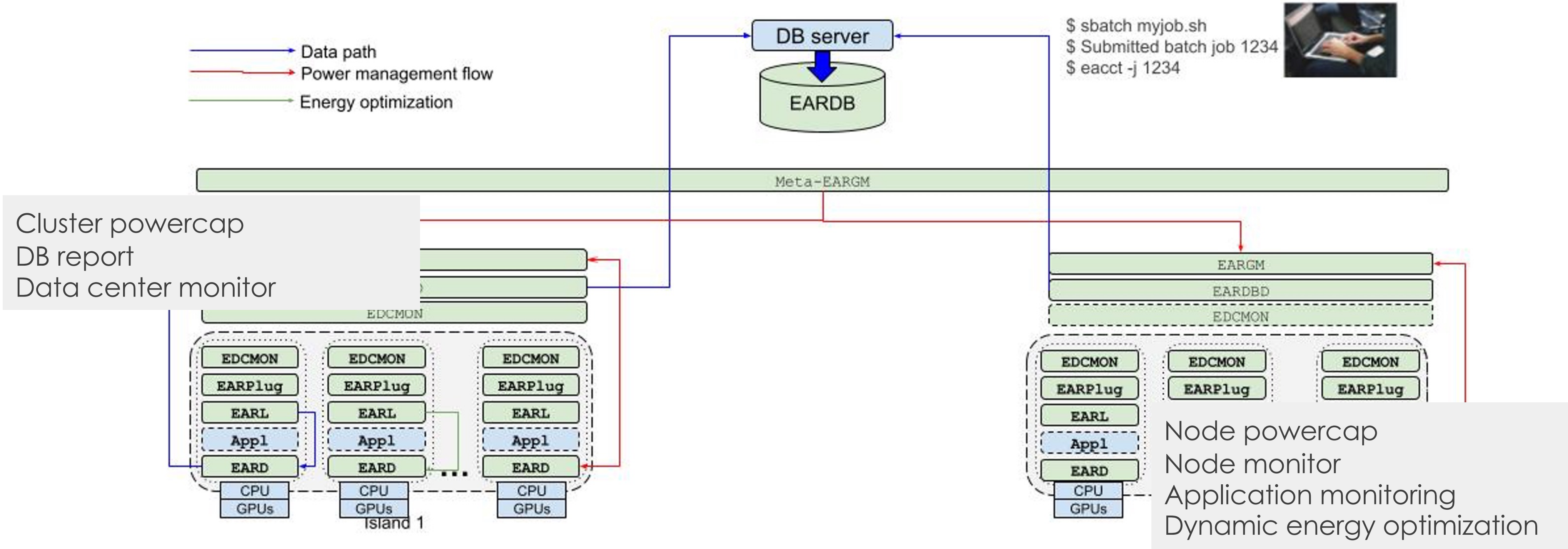
- **Computational nodes Monitoring :EARD**
  - Extensible **monitoring** : Power, CPU frequency, temperature, etc
  - Multiple sources of data: inband IPMI, redfish, NVIDIA, MSR, RAPL…
  - Extensible **report** : MariaDB, Postgres, Sysfs, Prometheus,…
  - **Anomalies** detection for power and temperature

Rack, Node, **Application** Monitoring



- **Application Monitoring : EAR library**
  - **Runtime Application monitoring, 100% dynamic, no code modifications**
  - **Collects the Application Runtime Signature**: Performance & Power

EAR architecture



- Software group of nodes
- Distributed features: Monitoring, optimization, power management, reporting
- Vendor independent

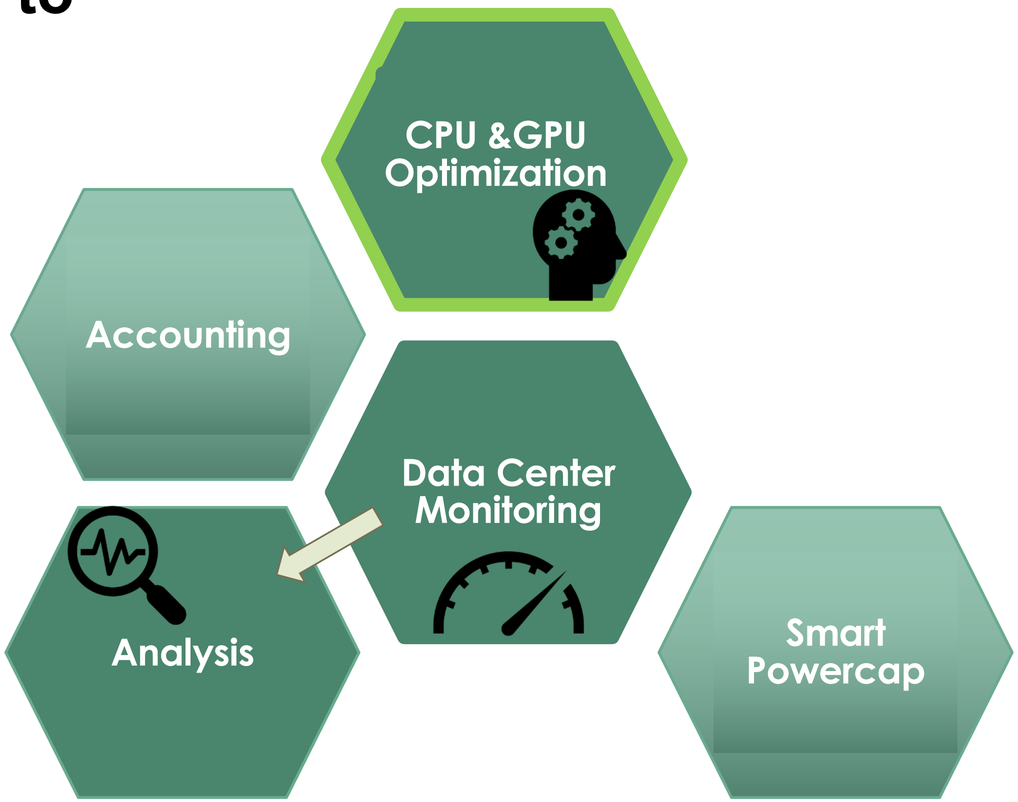
Dynamic energy optimization

Get more performance per Watt

Dynamic Energy Optimization: Adapt the power consumption to the application activity

- Runtime
- Transparent
- Application : Adapt configuration to **application** "Activity"
- Cluster: Adapt configuration to **Workload** "Activity"

- **Application high efficiency mind-set**

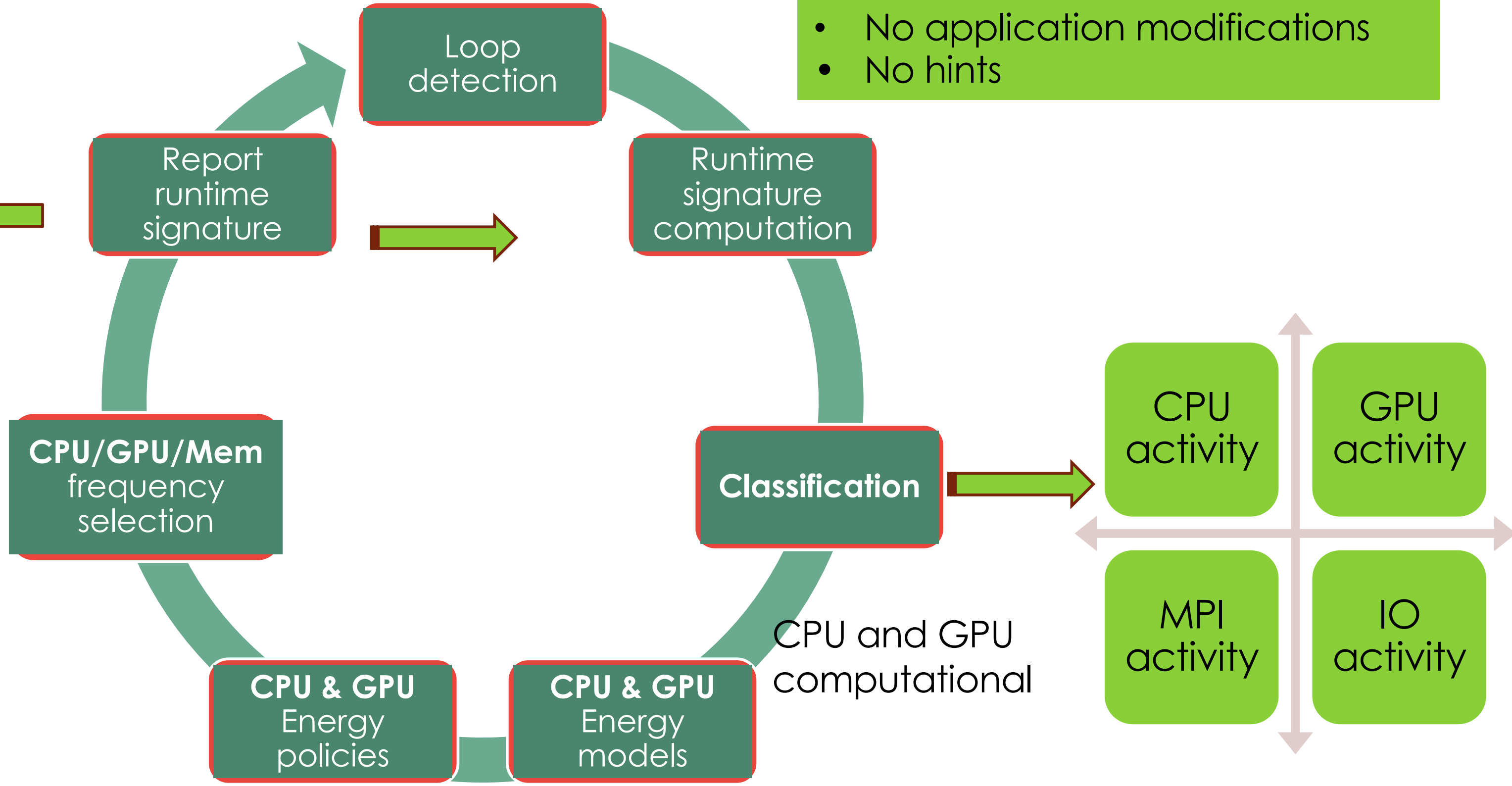


How to get more performance per watt

The EAR Library optimization loop

sbatch myapp.sh

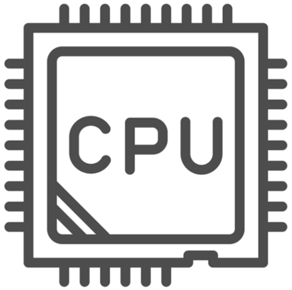
- 100% runtime
- No application modifications
- No hints



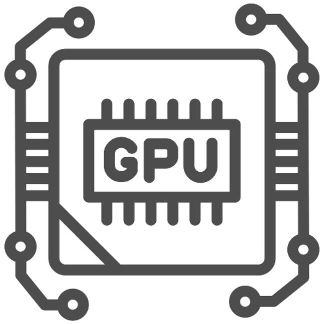
Application runtime characterization: Utilization vs “effective” utilization: The signature

EAR
Energy Aware Runtime

The key question is not whether the CPU or GPU is *used,* is **what is used for and how much does it consumes** to do the job…. And EAR knows!!!



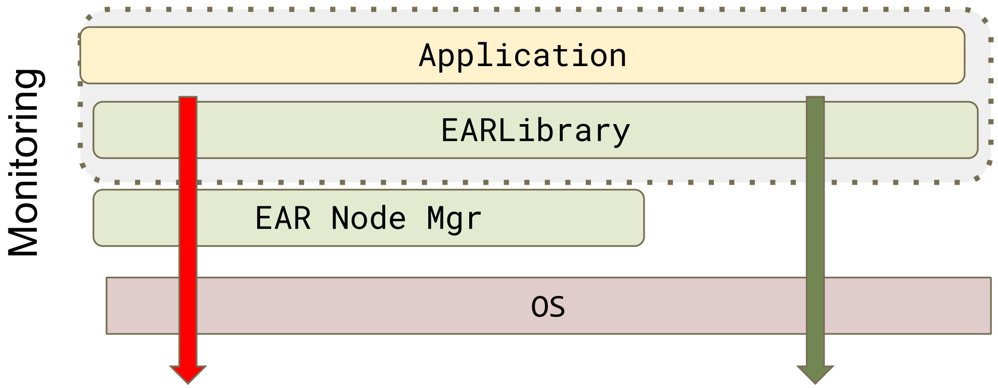
CPU activity : CPI, Memory BW, GFlops, IO, network
Power consumption: DC node power, DRAM and CPU power



GPU activity : GPU activity (tensor, nvlink etc)
Power consumption: DC node power, GPU power

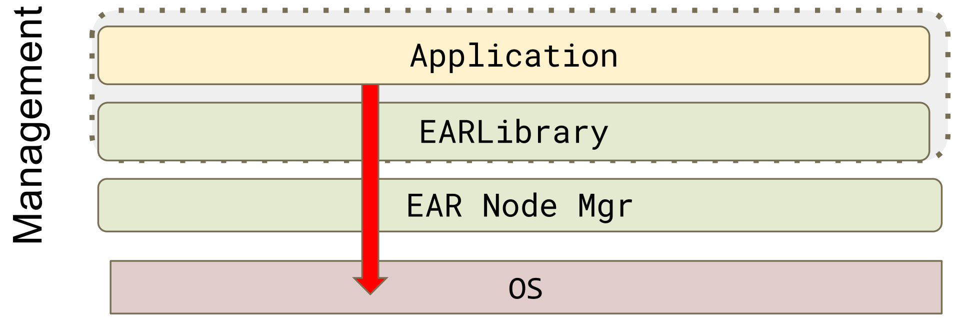
What about portability?

- EAR is very portable and flexible since all the APIs for monitoring and management are implemented in **two layers**
  - Generic
  - Device/API specific
- There is always a **dummy API** in case one API is not supported for a new architecture



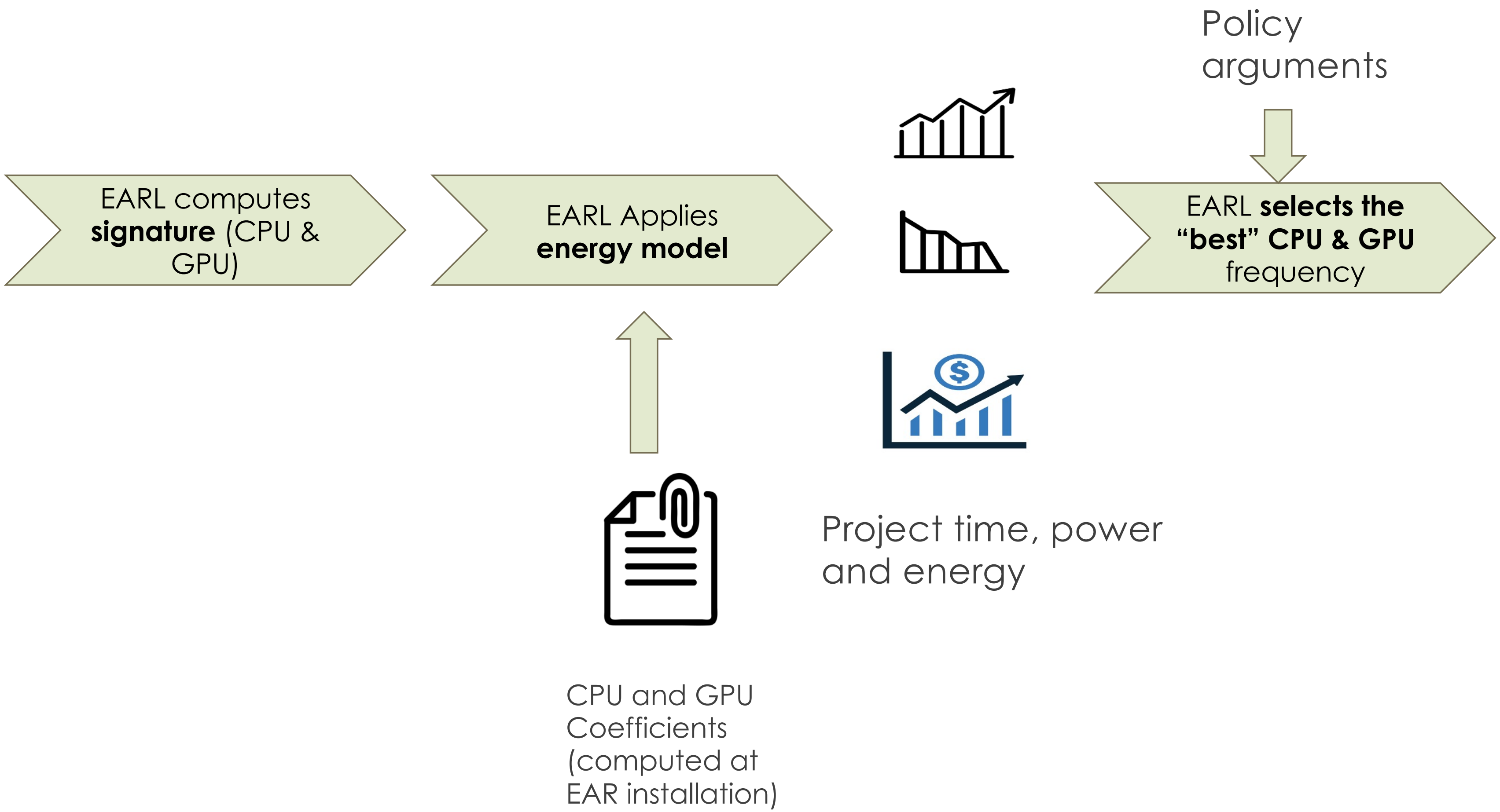
IPMI RAPL MSR HSMP REDFISH etc

Perf nvidia levelzero rsmi hwmon etc



- ➔ OS Cpufreq : userspace, intel_pstate
- ➔ AMD HSMP with virtual list of p-states
- ➔ GPU
  - Libnvdidia-ml, Levelzero, smi

Energy dynamic optimization



Min_energy_to_solution policy (CPU and **GPU**): Minimizes the **energy with a maximum performance (time) penalty defined as threshold**

Results on H100 - SXM5

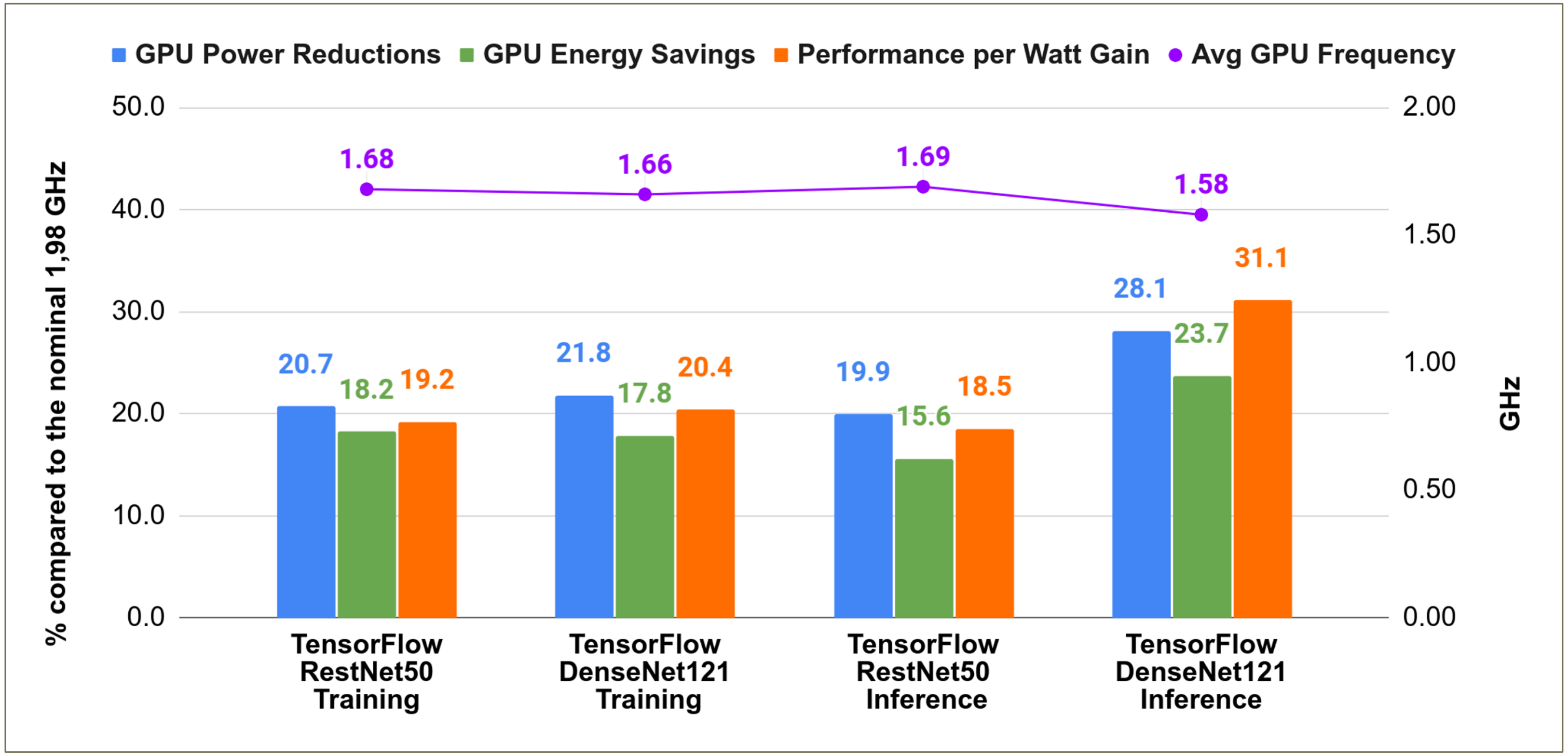
- 4x GPU
- TDP: 700 W
- Nominal Frequency: 1980 MHz

AI Applications on H100

Applications	Version	Model	# GPUS	MPI tasks	OpenMP Threads	Inputs (batch_size)
TensorFlow	2.15	Resnet 50	1	1	72	128
		DenseNet121	1	1	72	128
		VGG19	1	1	72	128
PyTorch	2.2	Resnet 50	1	1	8	256
		Resnet 50	4	4	4	256
		DenseNet121	1	1	8	128
		DenseNet121	4	4	4	128

EAR GPU optimization on AI applications (1 GPU)

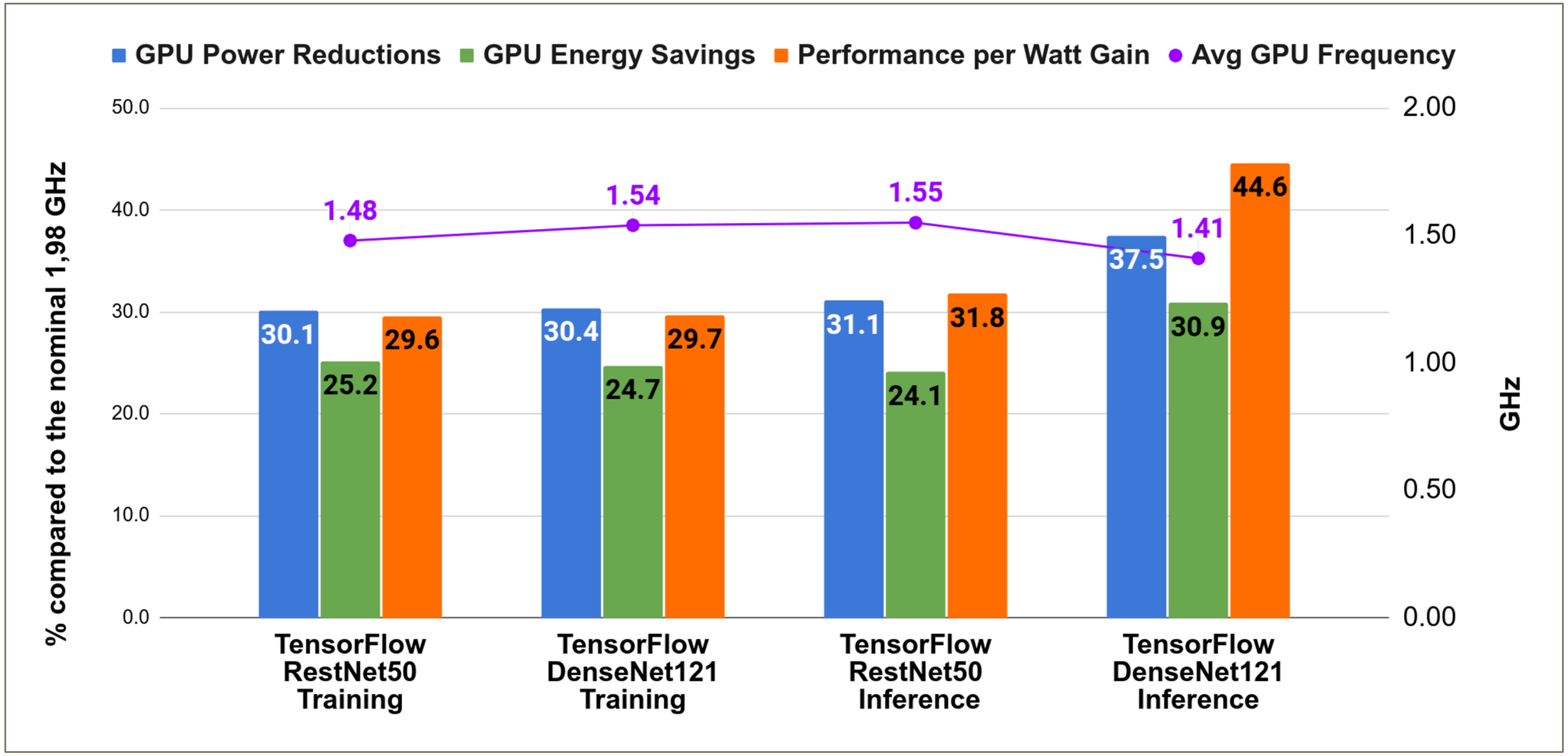
TensorFlow Benchmarks: GPU Energy savings and efficiency



EAR Optimization with 5% policythreshold: **Avg energy saving 18.8 %**

EAR GPU optimization on AI applications (1 GPU)

TensorFlow Benchmarks: GPU Energy savings and efficiency

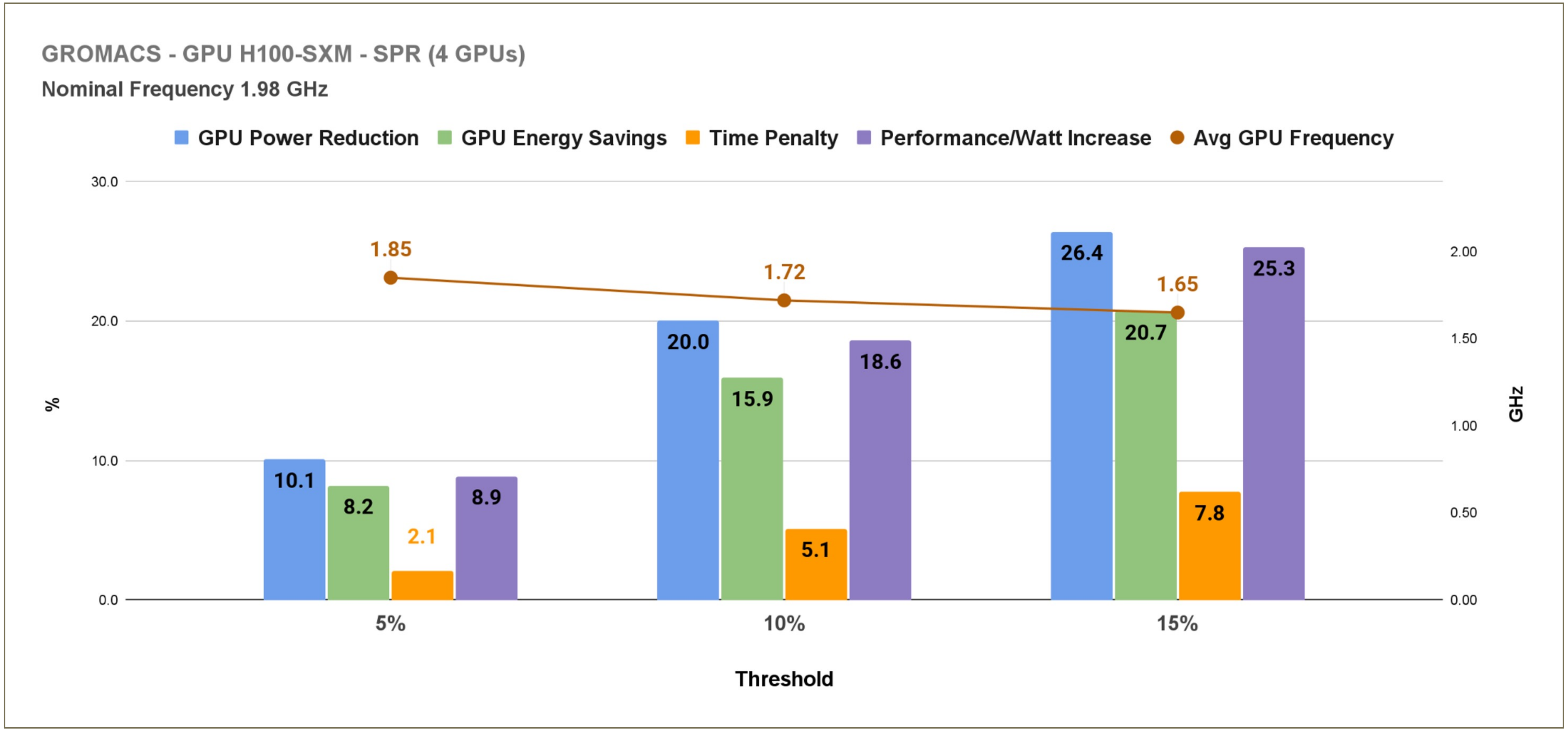


EAR Optimization with 10% threshold: **avg. energy saving 26.2%**

Gromacs on H100 SXM5 on 4 GPUs

- Processor: Xeon Platinum 8462Y+ (S:C:T = 2:32:2)
- GPU: H100 SXM5, TDP = 700 W/GPU
- Gromacs dataset: BenchPEP-h
- GPU min_energy policy:
- Benchmark configuration:
    - 4 GPUs with 4 MPI tasks and 16 OpenMP threads

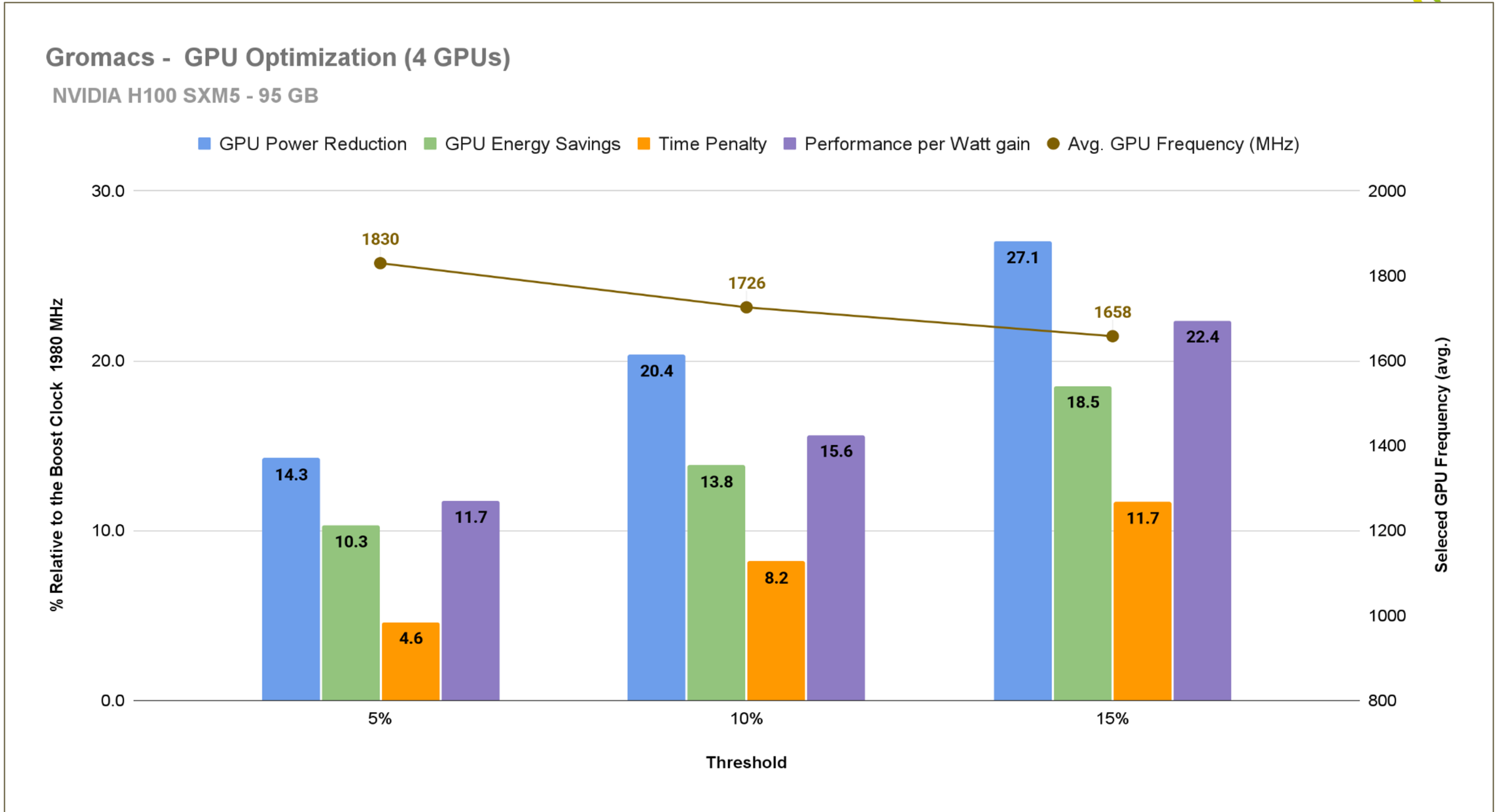
GPU savings with EAR Optimization



Gromacs on H100 SXM5 on 4 GPUs

- Processor: AMD EPYC 9334 (2x) 32 Cores/Socket
- GPU: H100 SXM5 (95 GB), TDP = 700 W/GPU
- Gromacs:
  - source: Nvidia NGC container
  - dataset: STMV benchmark
- GPU min_energy policy
- Benchmark configuration:
  - 4 GPUs with 2 thread-MPI tasks per GPU and 16 OpenMP threads per thread-MPI task

GPU savings with EAR Optimization



Energy efficient HPC/AI 2025

Node sharing support for big servers

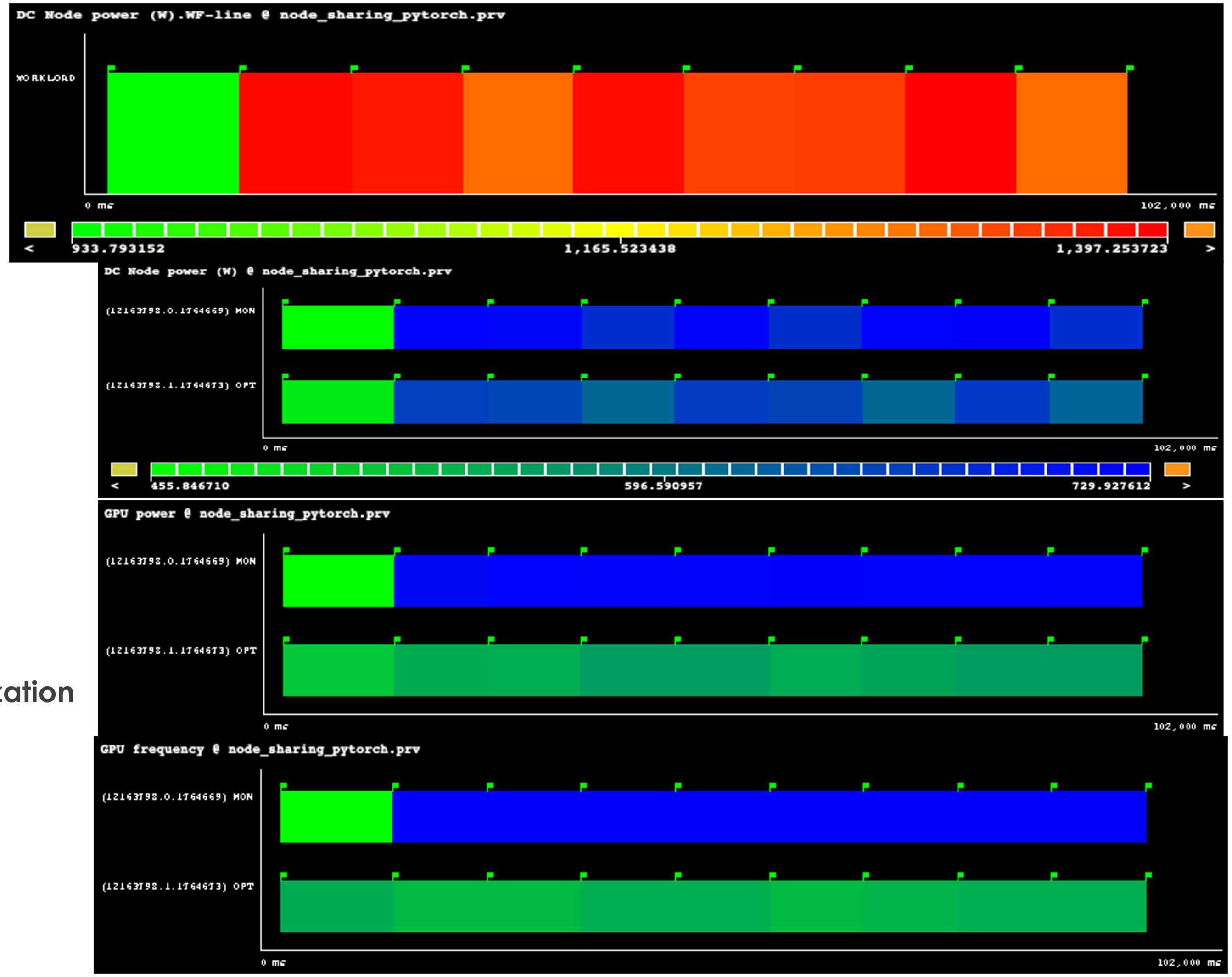
Node sharing support

- With the growing size of large CPU node such Granite Rapids from Intel and AMD Turin and powerful GPU nodes , the requirement of node sharing the get the maximum node utilization becomes more relevant
- EAR includes node sharing capabilities in order to monitor and optimize the running applications when sharing nodes
- Use cases supported are
  - Multiple applications sharing the node monitored and managed independently
  - Multiple applications sharing the node monitored and managed as a single entity

Node sharing example

- 2 instances of pytorch running at the same time , same node
- Each one asking for 1 GPU
  - 1 with monitoring
  - 1 with optimization
- EAR monitors each one independently and optimizes only one (as requested)
- Images generated with ear-job-visualization + Paraver

GPU power optimization



Total node power

Application power

No Optimization

Optimization

GPU power

No Optimization

Optimization

GPU frequency

No Optimization

Optimization

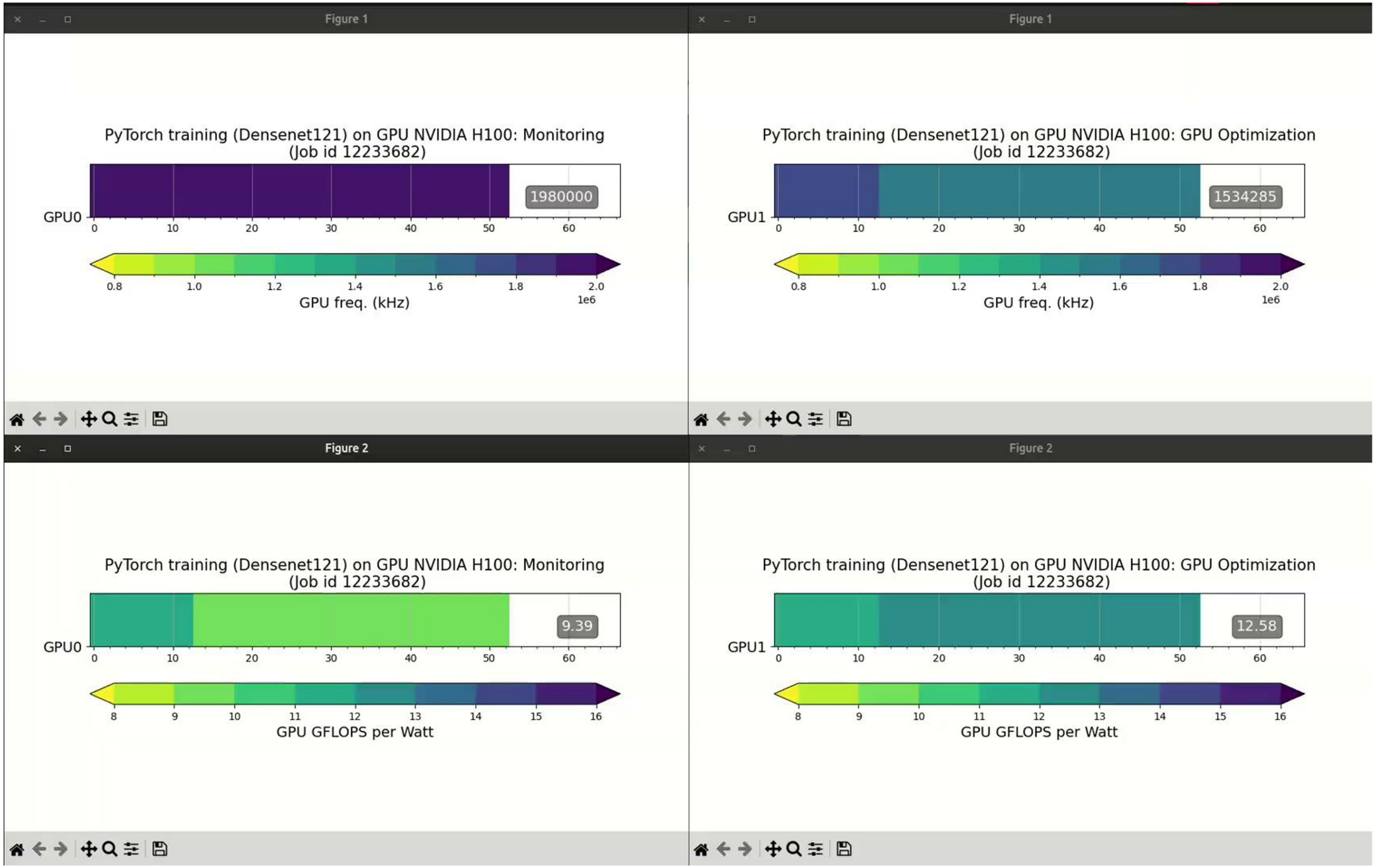


Figure 1
PyTorch training (Densenet121) on GPU NVIDIA H100: Monitoring (Job id 12233682)
GPU0
1980000
GPU freq. (kHz)
1e6
Figure 1
PyTorch training (Densenet121) on GPU NVIDIA H100: GPU Optimization (Job id 12233682)
GPU1
1534285
GPU freq. (kHz)
1e6
Figure 2
PyTorch training (Densenet121) on GPU NVIDIA H100: Monitoring (Job id 12233682)
GPU0
9.39
GPU GFLOPS per Watt
Figure 2
PyTorch training (Densenet121) on GPU NVIDIA H100: GPU Optimization (Job id 12233682)
GPU1
12.58
GPU GFLOPS per Watt

Smart powercap

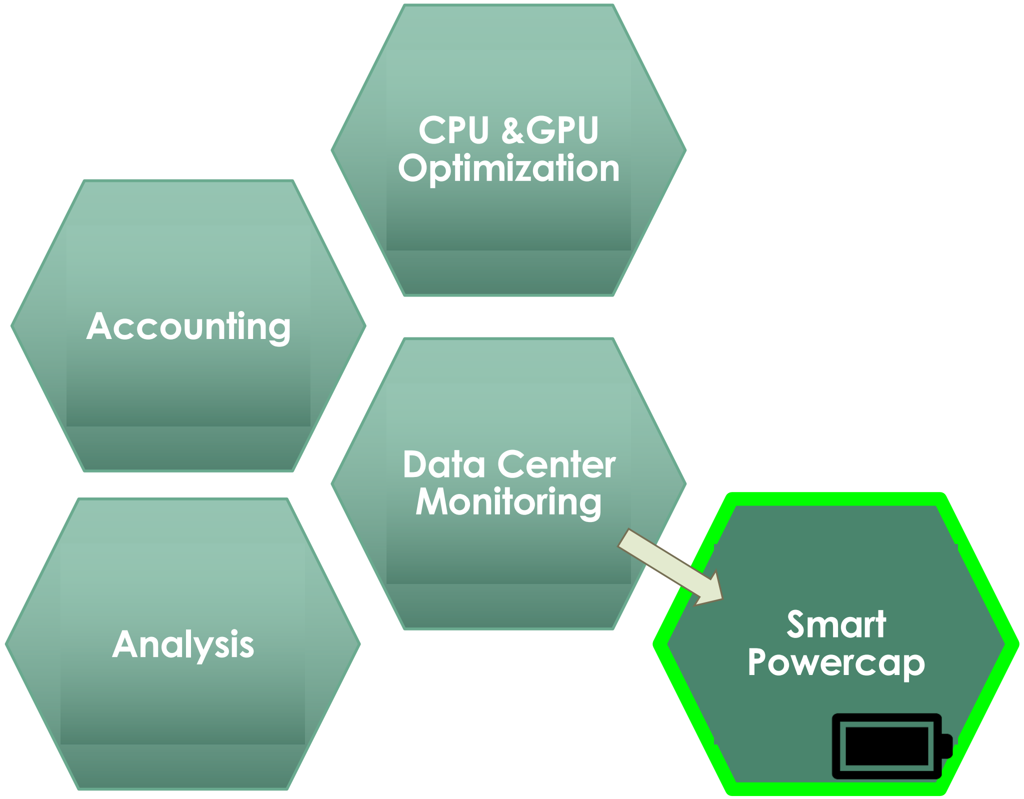
Dynamic powercap guided by application activity

Smart Powercap

- Node powercap
  - Heterogeneous
  - Dynamic
  - Application “driven”

- Cluster powercap
  - Heterogeneous
  - Dynamic
  - Node (Application) “driven”

- **Cluster high efficiency mind-set**

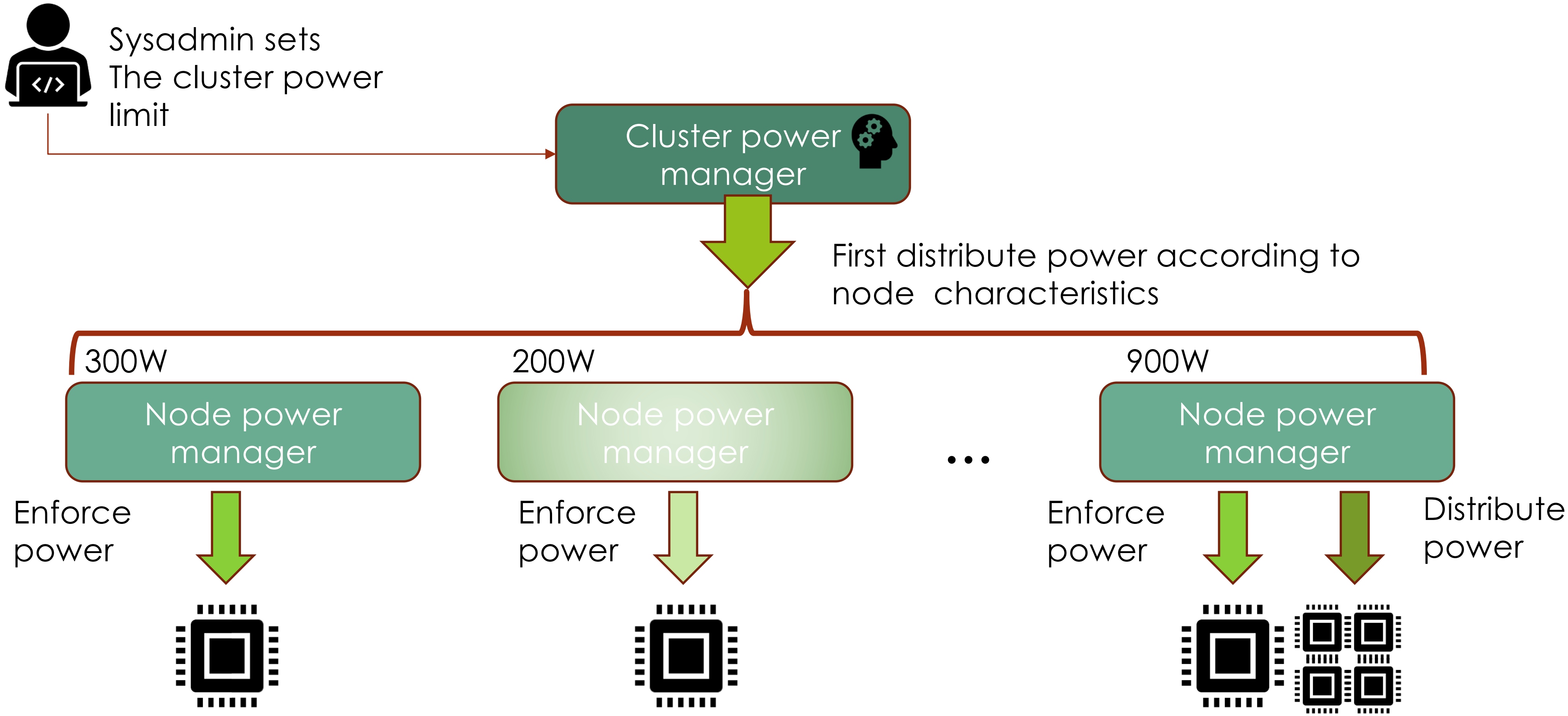


EAR smart powercap overview

- **Cluster power manager** distributes power to computational nodes
  - Hierarchical architecture for large scale clusters: 1 Meta- EARGM, N EARGMs
  - Two algorithms offered: **soft** (lightweight) and **hard** powercap
- **EAR Node powercap manager** enforces node power limit
  - Extensible through plugins: CPU, GPU
  - Dynamic **intra-node power re-allocation** based on application activity
- **EAR library informs** the EAR node powercap manager of application activity and power requirements
- Architecture
  - Software architecture: Compute nodes can be "grouped" with flexible & dynamic criteria
  - Vendor independent

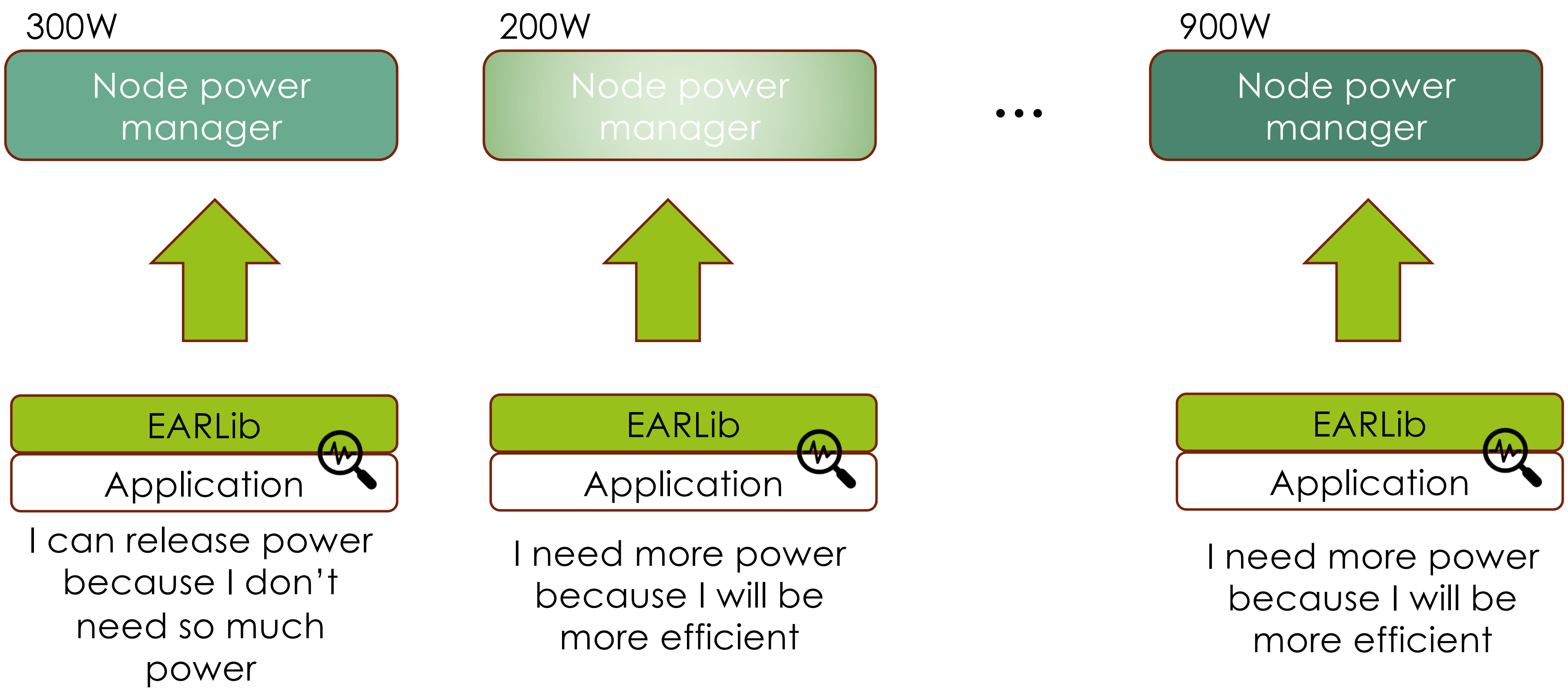
Powercap (I): Initial distribution

Cluster power manager distributes power and node power manager enforces power

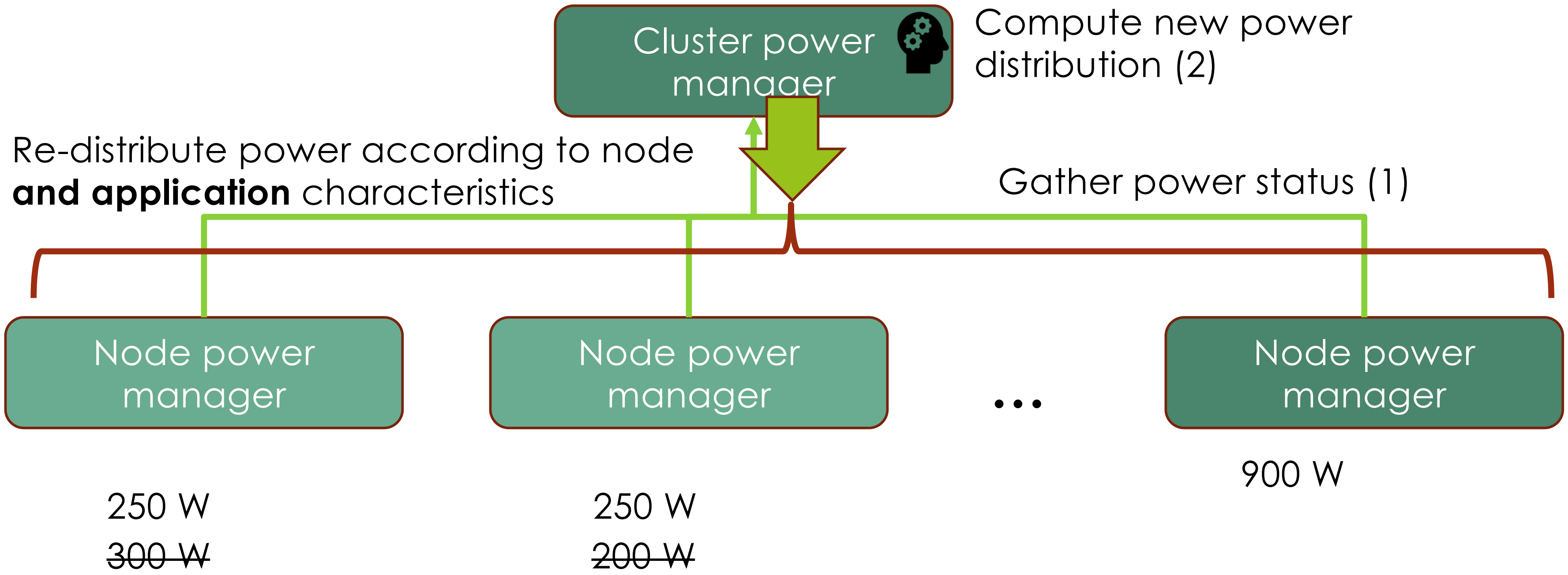


Powercap(II): Application feedback

Application (through EARlib) informs each node about its power needs



Powercap(III): Dynamic power reallocation



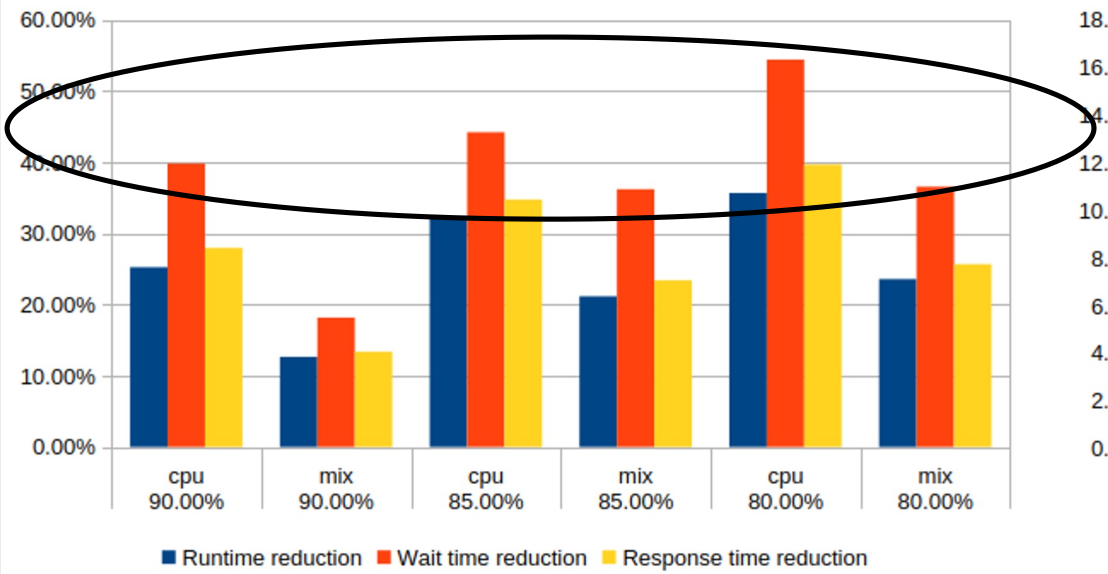
The "High Efficient Computing" mind-set

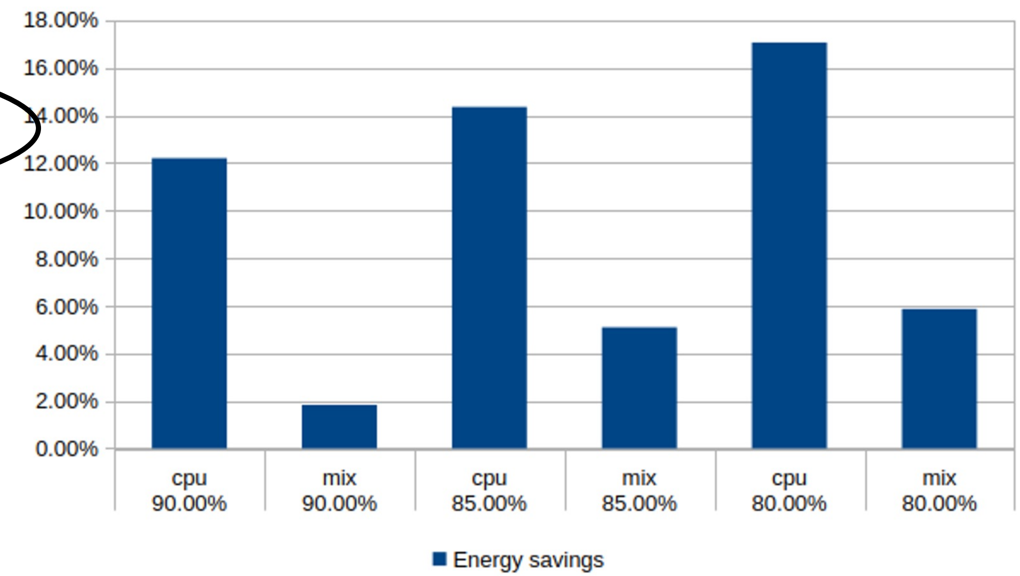
- Energy optimization seems to be against maximizing performance , but basically we look at individual applications and unlimited resources

- **This is not longer true if we focus on workloads and limited systems**

Energy/Power optimization is not a pain , it is a MUST

EAR cluster hard-powercap vs static powercap





- 10K nodes (simulation based results)
- SPR characterization CPU nodes

Reporting and visualization

- EAR CLI
- EAR visualization tools
- Grafana

EAR command for job accounting and node monitoring: eacct and ereport

- eacct is a linux command line tool to gather job metrics
- Regular users can gather their own data and managers can gather data from all the users
- Multiple filters and flags supported , including saving all the data in csv files (runtime and application signatures

```
[julitac@int3 TENSORFLOW_TEST]$ eacct -j 1460643
JOB-STEP USER       APPLICATION  POLICY NODES AVG/DEF/IMC(GHz) TIME POWER(W) GBS CPI   ENERGY(J)    GFLOPS/W IO(MBs) MPI% G-POW  G-FREQ  G-UTIL
1460643-5    julitac    rfm_TensorFlowSA MT     8     2.86/2.20/2.12   376.57     1343.82  21.43   0.29  4048335     0.0000    1960.8  44.6  868.93 1.410   89%/46%
1460643-4    julitac    rfm_TensorFlowSA ME     8     2.38/2.40/2.19   374.49     1321.24  21.71   0.30  3958278     0.0000    1971.8  46.7  870.58 1.410   90%/47%
1460643-3    julitac    rfm_TensorFlowSA MO     8     2.38/2.40/2.19   374.28     1322.08  21.90   0.30  3958678     0.0000    1972.7  46.2  869.99 1.410   90%/47%
1460643-2    julitac    rfm_TensorFlowSA MT     8     2.86/2.20/2.10   383.57     1329.94  21.08   0.29  4080972     0.0000    1924.8  44.5  856.43 1.410   88%/45%
1460643-1    julitac    rfm_TensorFlowSA ME     8     2.38/2.40/2.19   374.41     1321.40  21.77   0.29  3957900     0.0000    1972.1  46.2  871.06 1.410   90%/47%
1460643-0    julitac    rfm_TensorFlowSA MO     8     2.38/2.40/2.19   377.26     1313.15  21.66   0.29  3963175 0.0000    1957.1  46.4  863.03 1.410   90%/46%
```

Data visualization with Grafana

EAR
Energy Aware Runtime

Data visualization with Grafana

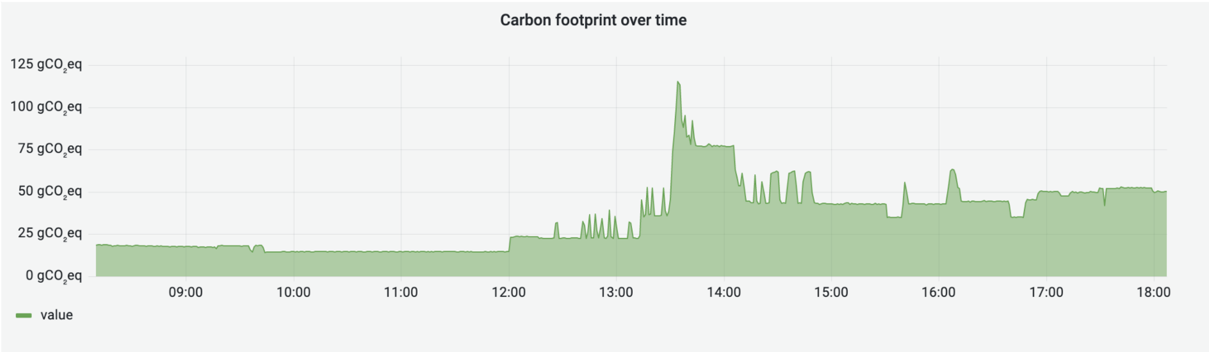
Jobs data can be seen with tools such as Grafana

Job metrics: CPU frequency, CPI, Memory bandwidth, Gflops, etc

Finished jobs

ID	Application ↑	Policy	Node power	Avg CPU frequency	Avg Mem frequency	CPI	GBS	GFlops	Elapsed time	MPI %	IO (MBS)	DRAM
230761	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	47
230759	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	47
230751	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	48
230749	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	48
230742	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	47
230658	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.17 GHz	0.47	156	125	6 min	2	0	47
230654	bt.D.x.ear.ME	min_energy	490 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	47
230650	bt.D.x.ear.ME	min_energy	489 W	2.18 GHz	2.18 GHz	0.47	156	125	6 min	2	0	48
230598	bt.D.x.ear.ME	min_energy	488 W	2.18 GHz	2.17 GHz	0.47	156	125	6 min	2	0	48
230761	bt.D.x.ear.MO	monitoring	528 W	2.73 GHz	1.85 GHz	0.54	169	135	6 min	2	0	49

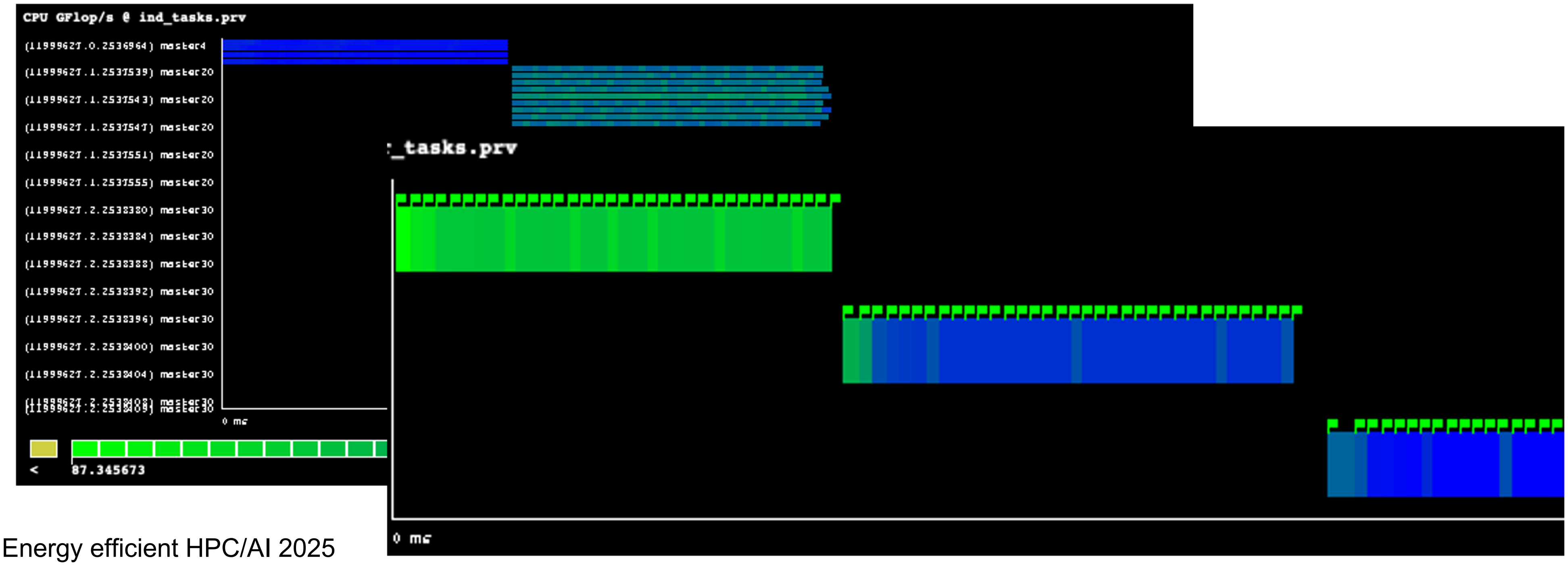
System metrics: Carbon footprint



EAR job visualizer

Helps the end users to understand their application behavior and their resources usage.

- Static images/Paraver traces with EAR metrics visualization for specific jobs over the time.



And then …What’s next?

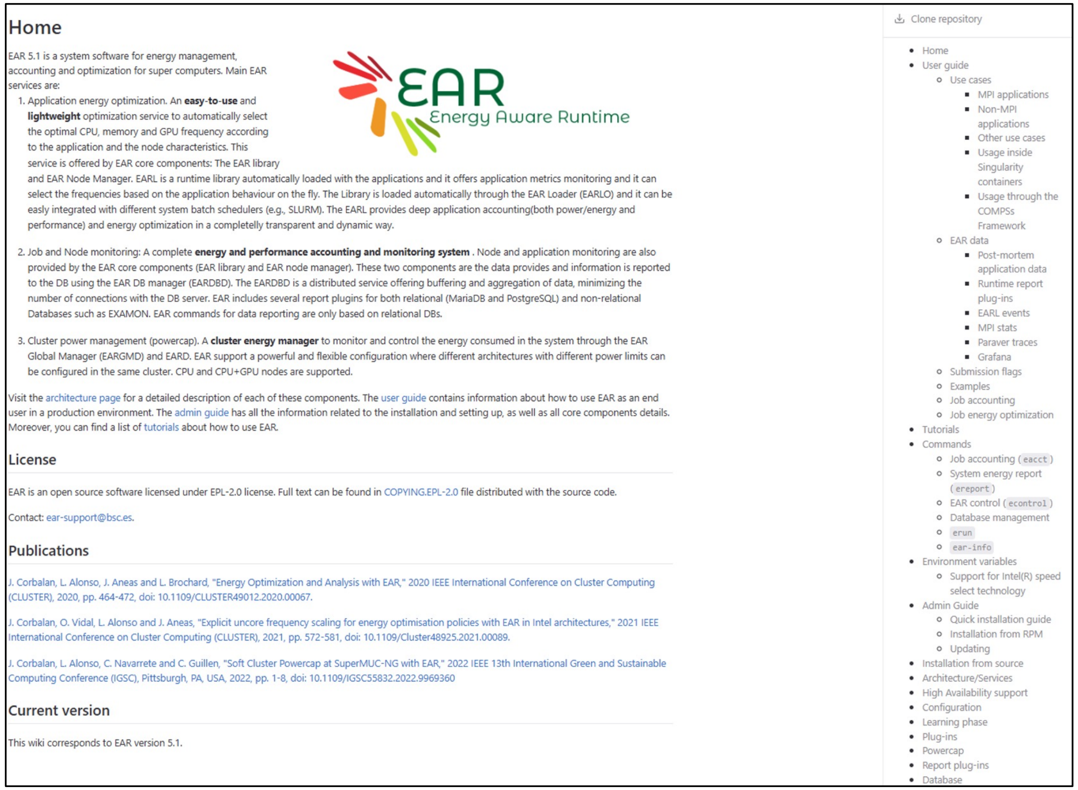
On going EU projects

- DYMAN: Dynamic data center cooling management
- ODISSEE
  - Energy efficient workflows for data science (CERN)
  - New architectures support for data science (Rhea, Maverick)
- DARE: Next generation european processors (OpenChip)
- SEANERGYS: European software stack for energy management

Useful links

Further information

- EAR Wiki (https://github.com/eas4dc/EAR/wiki)
- EAR GitHub repository (incl. CPU models) (https://github.com/eas4dc/EAR)
- Energy Aware Solutions (https://www.eas4dc.com/)



Home

EAR 5.1 is a system software for energy management, accounting and optimization for super computers. Main EAR services are:

1. Application energy optimization. An **easy-to-use** and **lightweight** optimization service to automatically select the optimal CPU, memory and GPU frequency according to the application and the node characteristics. This service is offered by EAR core components: The EAR library and EAR Node Manager. EARL is a runtime library automatically loaded with the applications and it offers application metrics monitoring and it can select the frequencies based on the application behaviour on the fly. The Library is loaded automatically through the EAR Loader (EARLO) and it can be easily integrated with different system batch schedulers (e.g., SLURM). The EARL provides deep application accounting(both power/energy and performance) and energy optimization in a completelly transparent and dynamic way.
2. Job and Node monitoring: A complete **energy and performance accounting and monitoring system** . Node and application monitoring are also provided by the EAR core components (EAR library and EAR node manager). These two components are the data provides and information is reported to the DB using the EAR DB manager (EARDBD). The EARDBD is a distributed service offering buffering and aggregation of data, minimizing the number of connections with the DB server. EAR includes several report plugins for both relational (MariaDB and PostgreSQL) and non-relational Databases such as EXAMON. EAR commands for data reporting are only based on relational DBs.
3. Cluster power management (powercap). A **cluster energy manager** to monitor and control the energy consumed in the system through the EAR Global Manager (EARGMD) and EARD. EAR support a powerful and flexible configuration where different architectures with different power limits can be configured in the same cluster. CPU and CPU+GPU nodes are supported.

Visit the architecture page for a detailed description of each of these components. The user guide contains information about how to use EAR as an end user in a production environment. The admin guide has all the information related to the installation and setting up, as well as all core components details. Moreover, you can find a list of tutorials about how to use EAR.

License

EAR is an open source software licensed under EPL-2.0 license. Full text can be found in COPYING.EPL-2.0 file distributed with the source code.

Contact: ear-support@bsc.es.

Publications

J. Corbalan, L. Alonso, J. Aneas and L. Brochard, "Energy Optimization and Analysis with EAR," 2020 IEEE International Conference on Cluster Computing (CLUSTER), 2020, pp. 464-472, doi: 10.1109/CLUSTER49012.2020.00067.

J. Corbalan, O. Vidal, L. Alonso and J. Aneas, "Explicit uncore frequency scaling for energy optimisation policies with EAR in Intel architectures," 2021 IEEE International Conference on Cluster Computing (CLUSTER), 2021, pp. 572-581, doi: 10.1109/Cluster48925.2021.00089.

J. Corbalan, L. Alonso, C. Navarrete and C. Guillen, "Soft Cluster Powercap at SuperMUC-NG with EAR," 2022 IEEE 13th International Green and Sustainable Computing Conference (IGSC), Pittsburgh, PA, USA, 2022, pp. 1-8, doi: 10.1109/IGSC55832.2022.9969360

Current version

This wiki corresponds to EAR version 5.1.

Clone repository