

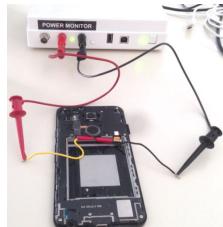
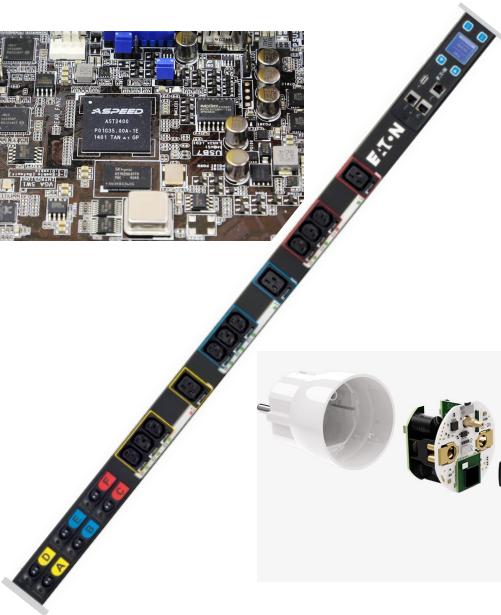
Green Computing - CM 4

Moyens de mesure de l'énergie matériels et logiciels - Méthodes de mesure

Quentin Perez
quentin.perez@insa.fr
4INFO
2023-2024

Deux grands moyens de mesure

Matériels



Logiciels



perf



carbontracker

Moyens de mesure matériels

Divers moyens pour diverses applications

- Baseboard Management Controller (BMC)
- Wattmètre (prise ampèremétrique)
- Power Distributed Units (PDUs)
- Disjoncteurs industriels monitorés
- Compteur électrique



Divers moyens pour diverses applications

⇒ *Dispositifs expérimentaux*

- Baseboard Management Controller (BMC)
- Wattmètre (prise ampèremétrique)
- Power Distributed Units (PDU)
- Disjoncteurs industriels monitorés
- Compteur électrique

Il est possible de placer des *dispositifs expérimentaux* spécifiques à divers endroits



Divers moyens pour diverses applications

⇒ *Dispositifs expérimentaux*

- Baseboard Management Controller (BMC)
- Wattmètre
- Power D
- Disjoncteur
- Compteur

**Attention aux fréquences d'
échantillonnage des différents
systèmes**
⇒ **Influence sur la précision
des mesures**

Il est possible de placer des *dispositifs expérimentaux* spécifiques à divers endroits



Baseboard Management Controller (BMC)

Puce dédiée sur la carte mère d'un serveur ou d'un équipement réseau (routeur, switch, etc.) permettant la remontée d'informations notamment matérielles via Ethernet ou port série.

- Température
- Voltage
- Débit des interfaces réseaux
- Courant
- Puissance
- Vitesse des ventilateurs
- etc.

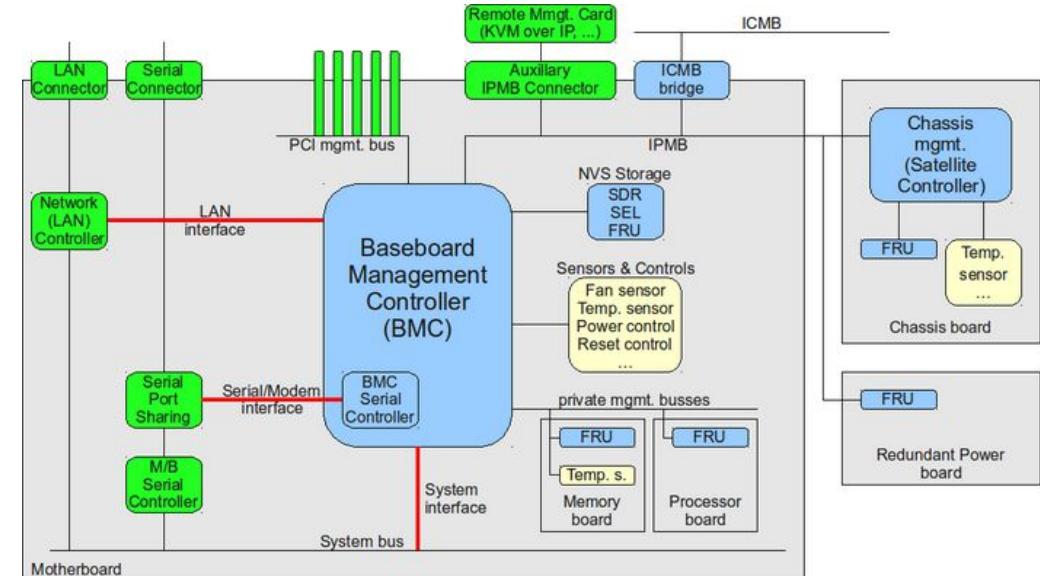


Baseboard Management Controller (BMC)

Puce dédiée sur la carte mère d'un serveur ou d'un équipement réseau (routeur, switch, etc.) permettant la remontée d'informations notamment matérielles via Ethernet ou port série.

- Température
- Voltage
- Débit des interfaces réseaux
- Courant
- Puissance
- Vitesse des ventilateurs
- etc.

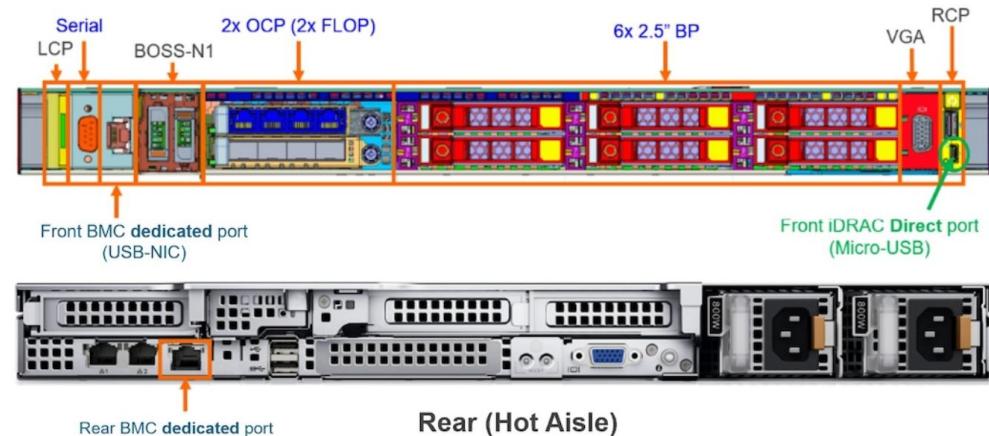
Intéressant en GL \Rightarrow permet de mesurer la conso physique d'une machine au plus près des composants



Baseboard Management Controller (BMC)

Puce dédiée sur la carte mère d'un serveur, d'un équipement réseau (routeur, switch, etc.) permettant la remontée d'informations notamment matérielles via Ethernet ou port série.

- Température
- Voltage
- Débit des interfaces réseaux
- Courant
- Puissance
- Vitesse des ventilateurs
- etc.



Wattmètre (prise ampèremétrique)

Prise capable de mesurer le courant, la tension et la puissance à intervalles réguliers. Peut-être communicante ou non selon le type :

- limité à 1 appareil
- prend compte la puissance totale de l'appareil en entrée
- simple à utiliser mais compliqué à mettre en œuvre pour des dizaines d'équipements
- peut-être peu précis (basse fréquence d'échantillonnage)
- souvent limité à 16A

Intéressant en GL ⇒ permet de mesurer la conso sur une machine pour faire des validations/estimations

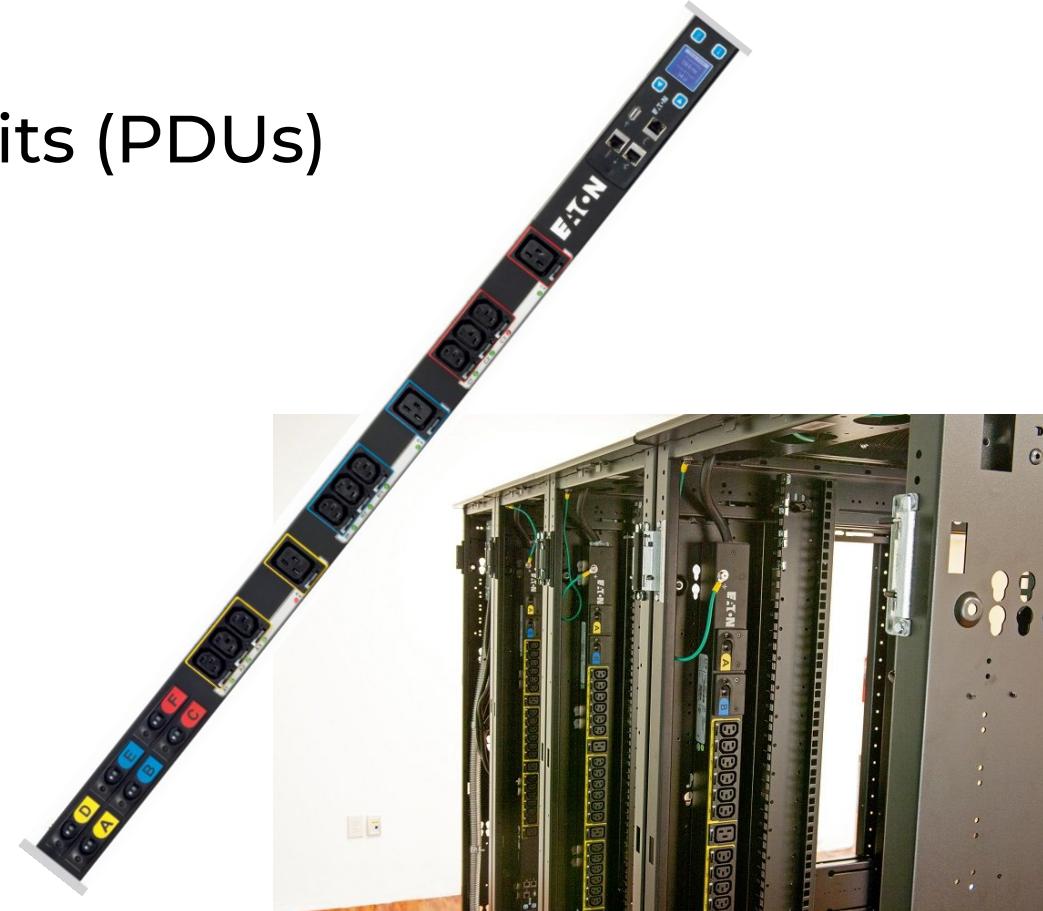


Power Distributed Units (PDUs)

Unités de distribution d'énergie rackables et dédiées à l'alimentation des équipements IT en datacenter. Remontée d'information par Ethernet.

- Permet d'alimenter plusieurs équipements sur une rampe (max 11kW) ou 3 phases (max 22kW) [[fiche technique Eaton](#)]
- **Précis** mais attention à la fréquence d'échantillonnage
- Permet d'avoir la consommation à l'**échelle d'une baie mais aussi d'une prise**

Intéressant GL ⇒ permet de mesurer la consommation d'un serveur ou d'une baie "conditions réelles", c-à-d en entrée d'alimentation/baie et à l'intérieur d'un rack.



Disjoncteurs industriels monitorés

Disjoncteurs capables de mesurer le courant dans les phases les parcourant

- Permet de visualiser les courants/puissances importantes en jeu
- Puissance à l'échelle de plusieurs baies voir d'une partie du DC.

Pas d'intérêt sur un aspect conception logiciel, car bien trop grosse granularité.



Compteurs industriels monitorés

Compteurs capables de mesurer le courant dans les phases les parcourant

- Permet de capturer les courants/puissances entrantes en DC

Pas d'intérêt sur un aspect conception logiciel, car trop (giga) grosse granularité.

Dispositifs expérimentaux

L'ensemble des dispositifs de mesure matériels qui ne rentrent pas dans les catégories précédentes. Ils permettent de faire par exemple :

- la mesure de courant en sortie de batterie sur smartphone

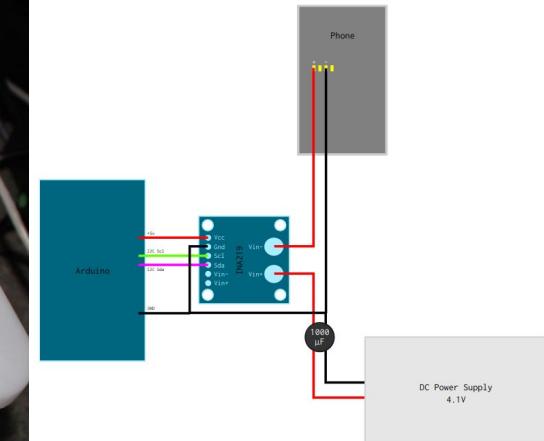
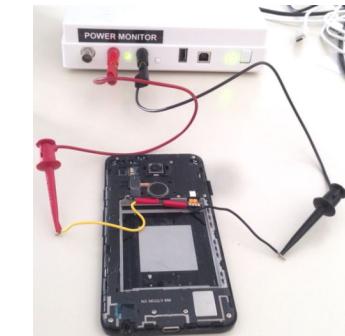
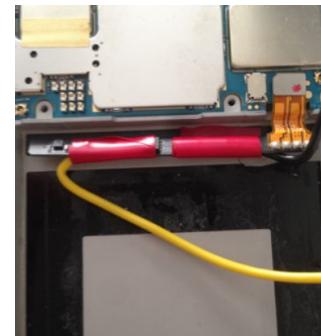
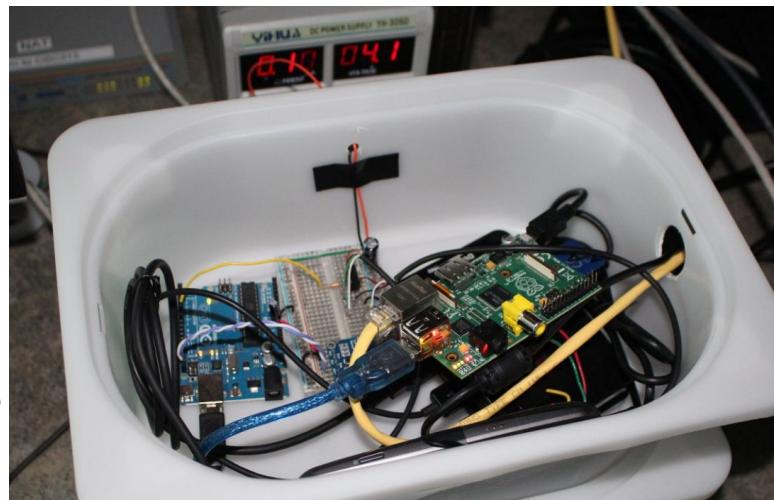


Figure 4: INA219 Circuit for power measurement from a Galaxy Nexus smartphone

[Hindle et al., GreenMiner: A Hardware Based Mining Software Repositories Software Energy Consumption Framework]

Dispositifs expérimentaux

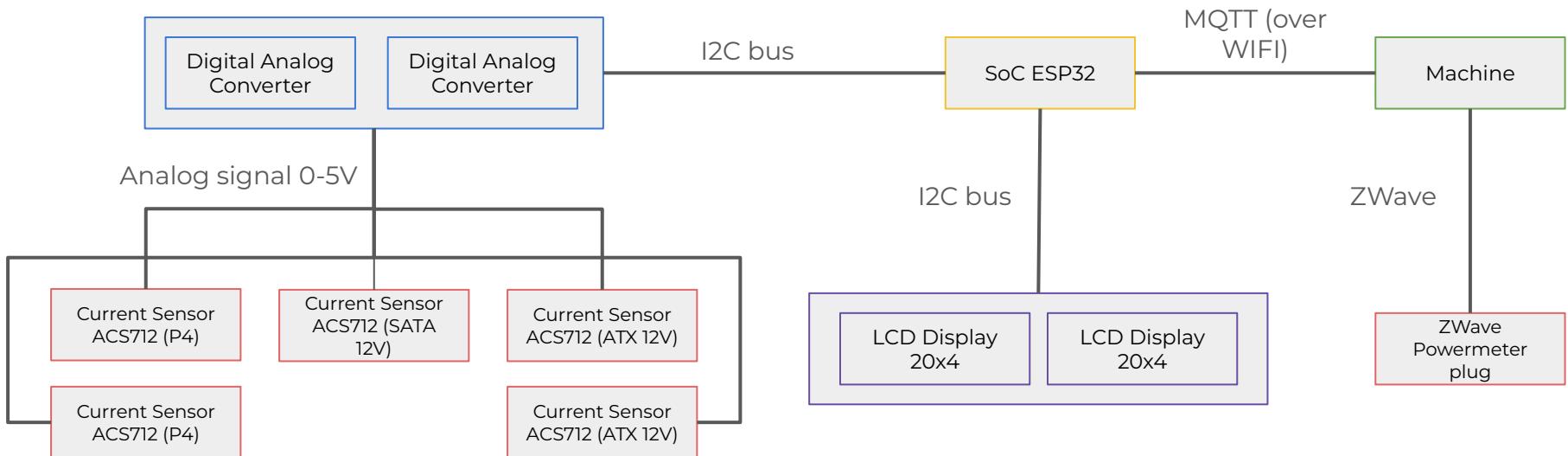
L'ensemble des dispositifs de mesure matériels qui ne rentrent pas dans les catégories précédentes. Ils permettent de faire par exemple :

- la mesure de puissance sur une alimentation ATX : **moment démo** (sans filet, à cru et au galop !)



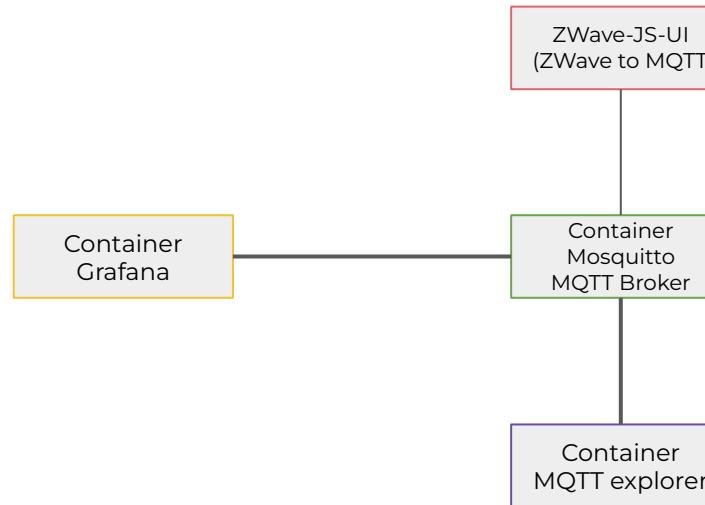
Dispositifs expérimentaux - mesures des puissances sur alimentation ATX

- Architecture matérielle



Dispositifs expérimentaux - mesures des puissances sur alimentation ATX

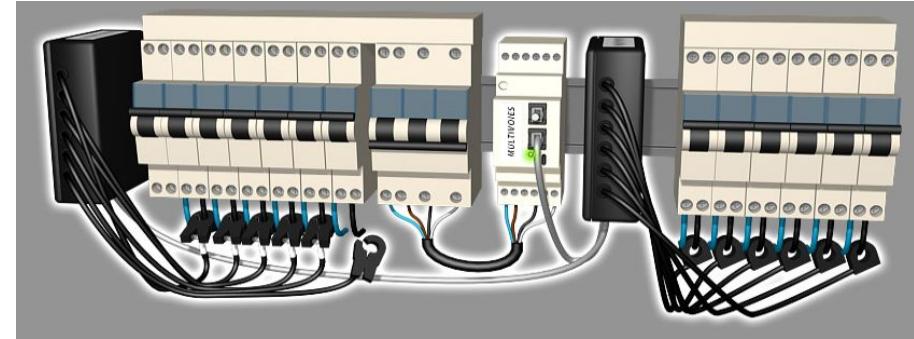
- Architecture logicielle



Dispositifs expérimentaux

L'ensemble des dispositifs de mesure matériels qui ne rentrent pas dans les catégories précédentes. Ils permettent de faire par exemple :

- de la mesure de courant et de tension par effet hall en outillant les tableaux ou les baies



Système Multivoies de Omégawatts

Utilisé dans des publications de mesure de la consommation de serveurs/logiciels:

- Wang, “Évaluation et modélisation de l’impact énergétique des centres de donnée en fonction de l’architecture matérielle/logicielle et de l’environnement associé”, 2020

Moyens de mesure logiciels

Mesure de l'énergie d'un CPU Intel

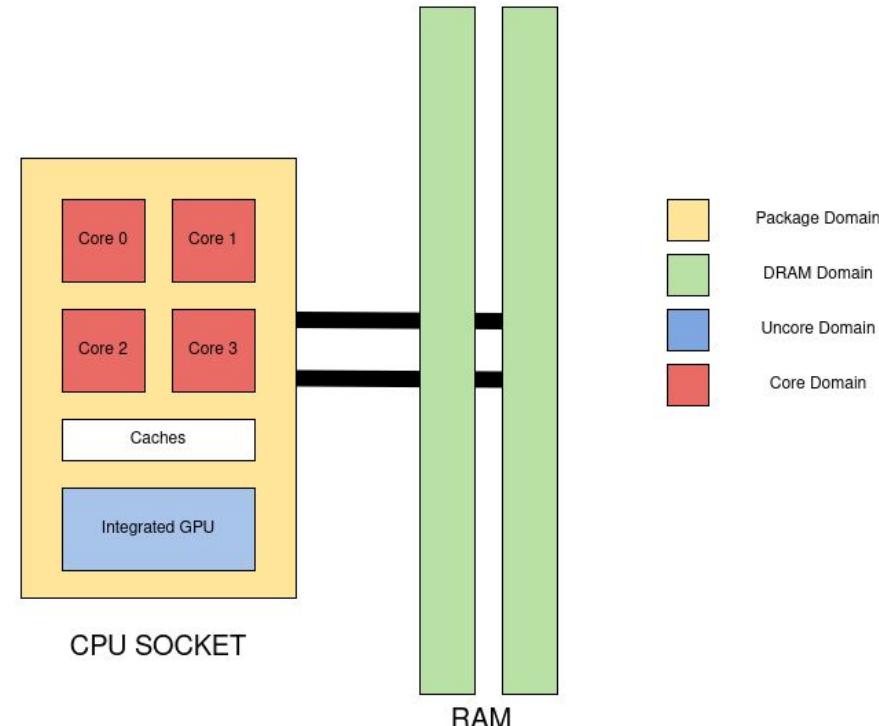
Il existe des registres spécifiques aux processeurs Intel permettant de mesurer la consommation d'énergie depuis que le processeur est allumé : **RAPL (Running Average Power Limit)**

Introduite en 2012 par Intel dans la génération Sandy Bridge

RAPL permet d'obtenir la consommation d'énergie du CPU en micro-joules à différents endroits du processeur.

La fréquence d'actualisation de RAPL est d'environ 1000Hz

Les valeurs RAPL sont aisément lisibles sur un système Linux

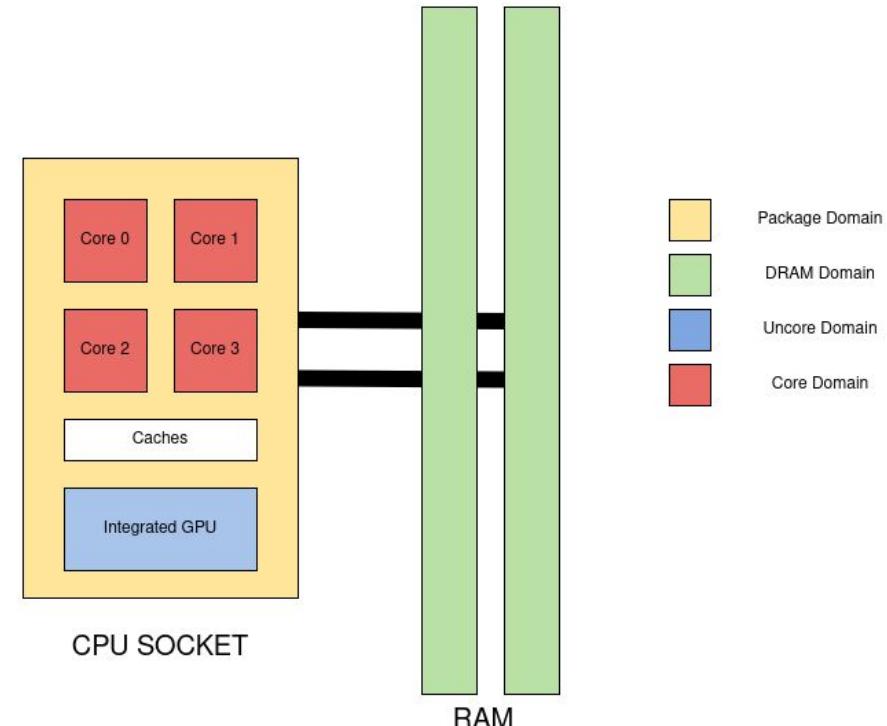


Mesure de l'énergie d'un CPU Intel

RAPL permet d'obtenir la consommation d'énergie du CPU en micro-joules à différents endroits du processeur :

- **Core** : énergie consommée par les coeurs du CPU
- **Uncore** : énergie consommée par l'integrated GPU
- **Package** : qui mesure **Core** + **Uncore** + cache mémoire L1/L2/L3
- **Psys** : qui mesure la consommation du die complet

Les valeurs RAPL sont aisément lisibles sur un système Linux sur :
"/sys/class/powercap/intel-rapl"



Mesure de la puissance d'un GPU NVIDIA

La mesure de puissance sur des GPU NVIDIA peut se faire via 2 moyens :

- par la ligne de commande via nvidia-smi [[documentation NVIDIA-smi](#)]

nvidia-smi ⇒ retourne les informations du GPU y compris la puissance sous forme de tableau lisible par l'utilisateur

```
nvidia-smi --query-update=gpu.name,timestamp,power.utilization.gpu,utilization.memory --format=csv -l  
1
```

⇒ retourne en continu les informations passées dans `--query-update` au format CSV (1 colonne par information)

- par l'API C fournie dans la bibliothèque NVIDIA *Management Library* (NVML) [[documentation NVML](#)]

Comparaison de wattmètres logiciels

Nous allons faire une comparaison des wattmètres logiciels en se basant sur la publication suivante : [\[Jay23\]](#)

An experimental comparison of software-based power meters: focus on CPU and GPU

Mathilde Jay^{*†}, Vladimir Ostapenco[†], Laurent Lefevre[†], Denis Trystram^{*}, Anne-Cécile Orgerie[‡], Benjamin Fichel[§]

** Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, Grenoble, France*

{mathilde.jay,denis.trystram}@univ-grenoble-alpes.fr

† Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, Lyon, France - {laurent.lefevre,vladimir.ostapenco}@ens-lyon.fr

‡ Univ. of Rennes, Inria, CNRS, IRISA, Rennes, France - anne-cecile.berger@irisa.fr

§ OVHCloud, France - benjamin.fichel@ovhcloud.com

Comparaison de wattmètres logiciels

Nous allons faire une comparaison des wattmètres logiciels en se basant sur la publication suivante : [\[Jay23\]](#)

An experimental comparison of software-based power meters: focus on CPU and GPU

Mathilde Jay^{*†}, Vladimir Ostapenco[†], Laurent Lefevre[†], Denis Trystram^{*}, Anne-Cécile Orgerie[‡], Benjamin Fichel[§]

** Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, Grenoble, France*

{mathilde.jay,denis.trystram}@univ-grenoble-alpes.fr

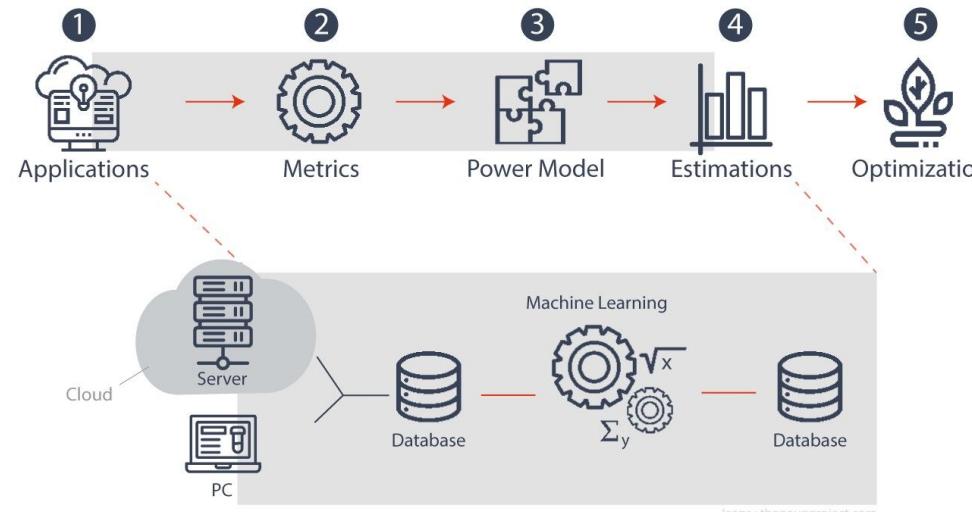
† Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, Lyon, France - {laurent.lefevre,vladimir.ostapenco}@ens-lyon.fr

‡ Univ. of Rennes, Inria, CNRS, IRISA, Rennes, France - anne-cecile.berger@irisa.fr

§ OVHCloud, France - benjamin.fichel@ovhcloud.com

Comparaison de wattmètres logiciels

Power profiling software = outil logiciel permettant d'obtenir des profils de puissance de processus/conteneurs/VMs (si intégré sur le temps ⏳ ⇒ énergie ⚡)



Comparaison de wattmètres logiciels - Energy Scope

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Developement				
Citation	[40]	[41]	[42]	[46]
First (latest) release date	Jul. 2019 (Aug. 2022)	Dec. 2020 (May 2021)	2021	Sept. 2009 (Jan. 2023)
Maintenu	Oui	Oui	Non	Oui

Comparaisons de wattmètres logiciels - Perf

Perf = outil Linux en ligne de commande permettant de profiler les performances d'un logiciel et de reporter sa consommation en se basant sur les valeurs de RAPL (*Running Average Power Limit*) [[Documentation](#)]

Installation avec APT :

```
apt install linux-tools-common linux-tools-generic linux-tools-`uname -r`
```

Utilisation :

```
perf stat -a -e /power/energy-[domain]/ [command to monitor]
```

[domain] = une valeur parmi : {pkg, cores}

Comparaisons de wattmètres logiciels - Perf

Perf = outil Linux en ligne de commande permettant de profiler les performances d'un logiciel et de reporter sa consommation en se basant sur les valeurs de RAPL (*Running Average Power Limit*) [[Documentation](#)]

Installation avec APT :

```
apt install linux-tools-common linux-tools-generic linux-tools-`uname -r`
```

Exemple d'utilisation :

```
perf stat -a -e /power/energy-core/ sysbench cpu --threads=15 run
```

Comparaison de wattmètres logiciels - Energy Scope

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Development				
Citation	[40]	[41]	[42]	[46]
First (latest) release date	Jul. 2019 (Aug. 2022)	Dec. 2020 (May 2021)	2021	Sept. 2009 (Jan. 2023)
Maintenu	Oui	Oui	Non	Oui

Comparaison de wattmètres logiciels - Energy Scope

Energy Scope = est un logiciel de monitoring de l'énergie et de profilage prenant en compte la consommation d'énergie des CPUs et GPUs Nvidia

[Jay23](#)

Impossible d'en dire plus, le site web d'Energy Scope n'existe plus...

Comparaison de wattmètres logiciels - Scaphandre

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Development				
Citation	[40]	[41]	[42]	[46]
First (latest) release date	Jul. 2019 (Aug. 2022)	Dec. 2020 (May 2021)	2021	Sept. 2009 (Jan. 2023)
Maintenu	Oui	Oui	Non	Oui

Comparaisons de wattmètres logiciels - Scaphandre

Scaphandre = outil de monitoring processus **et VMs** (sur base KVM/QEMU) capable d'estimer la puissance CPU en fonction de sa charge et des métriques d'utilisation des processus. Outil développé par la société Hubblo et open-source [Github Scaphandre]



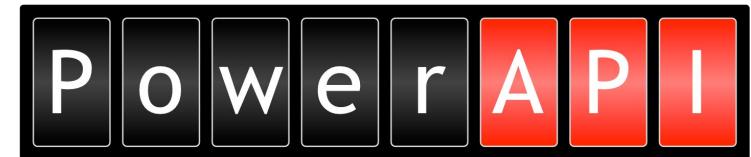
Comparaison de wattmètres logiciels - PowerAPI

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Development				
Citation	[40]	[41]	[42]	[46]
First (latest) release date	Jul. 2019 (Aug. 2022)	Dec. 2020 (May 2021)	2021	Sept. 2009 (Jan. 2023)
Maintenu	Oui	Oui	Non	Oui

Comparaisons de wattmètres logiciels - PowerAPI

PowerAPI = outil de monitoring
processus **et conteneurs Docker**
capable d'estimer la puissance CPU
grâce au RAPL et à plusieurs
métriques CPU et processus
(cgroup). Crée par l'équipe de
recherche Spirals de l'Université de
Lille.

Nous allons étudier en détails son
fonctionnement dans ce cours et
allons l'utiliser en TP.



Comparaison de wattmètres logiciels - Compatibilité et environnement matériel

Les 2 premiers critères à prendre en compte pour l'utilisation d'un wattmètre sont :

- **la compatibilité matérielle**
- **le type de virtualisation utilisé si besoin (Docker / VMs)**

		Power profiling software			
		<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Environment					
Hardware compatibility		Intel RAPL	Intel RAPL	Intel RAPL, Nvidia NVML	Intel RAPL
Scope		CPU, DRAM, process	CPU, DRAM, process	CPU, DRAM, GPU	CPU, DRAM
Virtualization support		Yes	Yes	No	No
Job management support		No	No	OAR, SLURM	No

Comparaison de wattmètres logiciels - Capacités

Puis, définir le **périmètre (granularité)** à montrer :

- **CPU**
- **GPU**
- **DRAM**
- **Processus / conteneur / VM**

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Environment				
Hardware compatibility	Intel RAPL	Intel RAPL	Intel RAPL, Nvidia NVML	Intel RAPL
Scope	CPU, DRAM, process	CPU, DRAM, process	CPU, DRAM, GPU	CPU, DRAM
Virtualization support	Yes	Yes	No	No
Job management support	No	No	OAR, SLURM	No

Comparaison de wattmètres logiciels - Fonctionnement

Ensuite, se poser la question de l'
échantillonnage
(sampling frequency) ⇒
impact sur la précision en fonction de la durée d'exécution de l'objet d'étude

Power profiling software				
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
Functional				
Hardware technology used	RAPL	RAPL	RAPL, NVML	RAPL
Software power model used	Regression based on perf events	CPU usage based		
Default sampling frequency (Hz)	1 10Hz max	0.1	2	10
Online reporting	Yes	Yes	No	Yes
Power profiling	Yes	Yes	Yes	Yes

Comparaison de wattmètres logiciels - Facilité déploiement/utilisation

Enfin, réfléchir à :

- **la complexité de déploiement,**
- **aux implications** (notamment au **niveau administration système**)
- **la manière de stocker les données collectées**

	Power profiling software			
	<i>Power API</i>	<i>Scaphandre</i>	<i>Energy Scope</i>	<i>Perf</i>
User-friendliness				
Availability of source code (License)	Yes (BSD 3-Clause)	Yes (Apache 2.0)	No	Yes (GNU GPL)
Ease of use	Poor	Fair	Good	Good
Quality of documentation	Good	Good	Good	Good
Configurability	Good	Good	Good	Good
Resulting data format	MongoDB, InfluxDB, Prometheus, CSV, Socket, File	Prometheus, Warp10, Riemann, JSON, Stdout	JSON	CSV, Stdout, File
Data visualisation possibilities	Grafana (InfluxDB, Prometheus)	Grafana (Prometheus)	Custom Dashboard	

Le wattmètre SmartWatts de PowerAPI

SmartWatts et PowerAPI

PowerAPI [[site PowerAPI](#)] est une suite d'outils développés par l'équipe de recherche Spirals et notamment Romain Rouvoy (PU à l'Université de Lille).

En 2020 est publié une publication présentant SmartWatts [[Fieni20](#)] : un wattmètre intégré à PowerAPI permettant de monitorer et profiler des processus et des conteneurs.

SMARTWATTS: Self-Calibrating Software-Defined Power Meter for Containers

Guillaume Fieni, Romain Rouvoy*, Lionel Seinturier

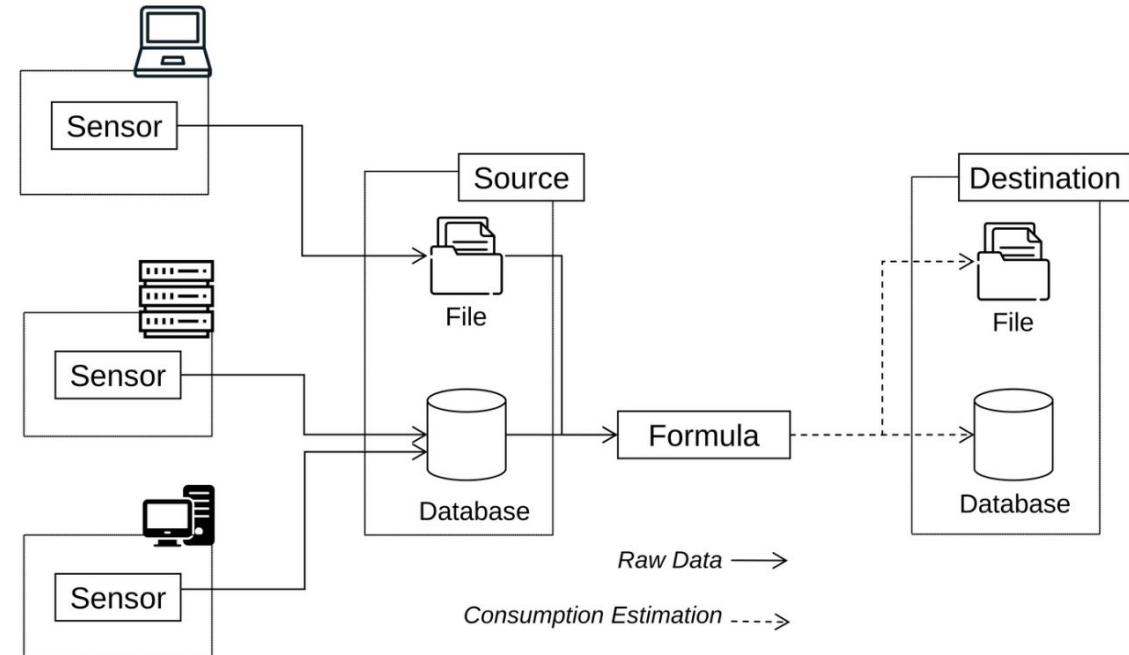
*Univ. Lille / Inria / *IUF*

firstname.lastname@univ-lille.fr

Architecture de PowerAPI

Architecture à 4 composants :

- Un ou plusieurs capteurs (sensors)
- Une ou plusieurs sources (source données brutes)
- Un composant de calcul (formula)
- Une destination pour les données de sortie

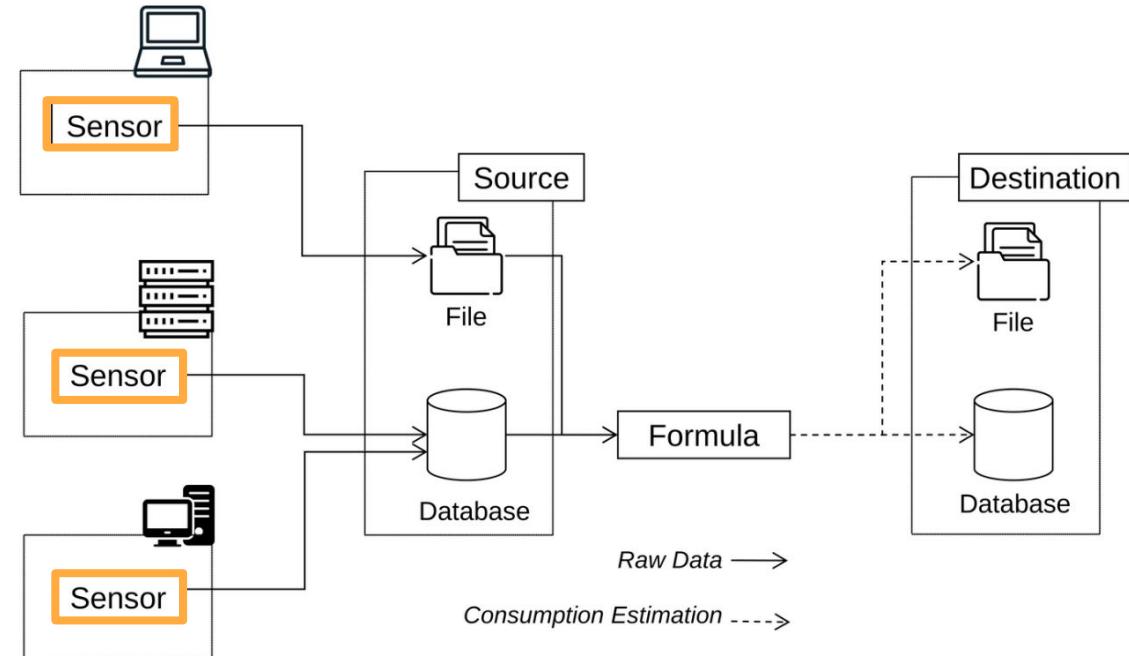


[[PowerAPI Overview](#)]

Architecture de PowerAPI

Capteur (Sensor) ⇒

processus logiciel
permettant de collecter
des données brutes
(métriques) concernant
le CPU et les processus
monitorés.

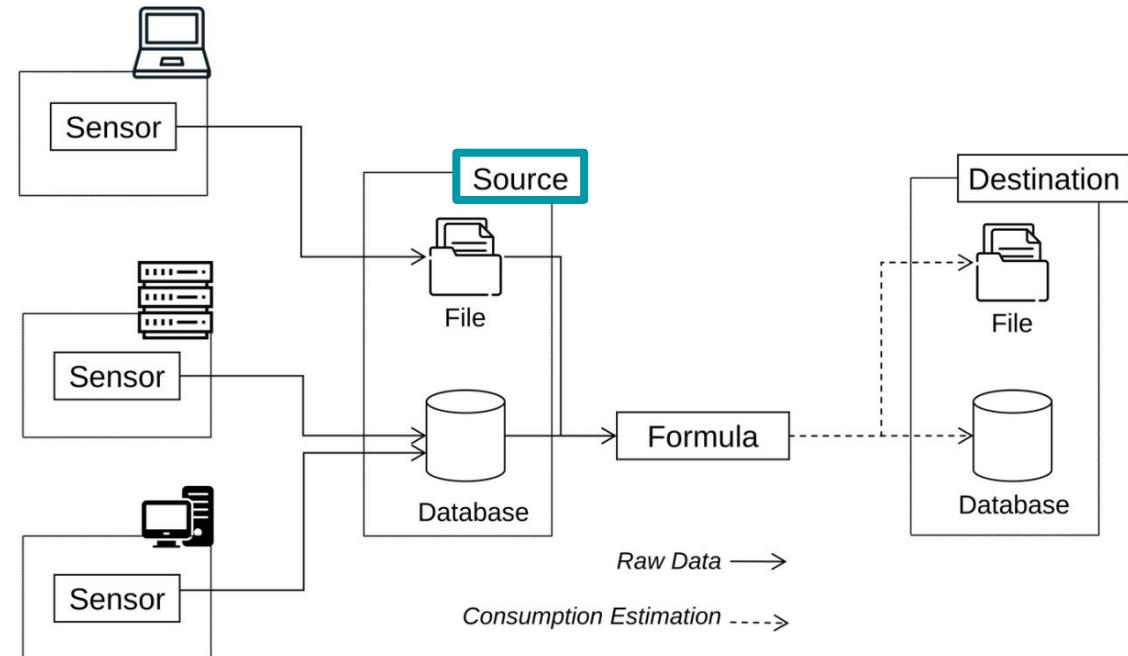


[[PowerAPI Overview](#)]

Architecture de PowerAPI

Source ⇒ composant permettant de stocker les données brutes. La source peut-être :

- une base de données (MongoDB, InfluxDB)
- un composant de monitoring (Prometheus)
- un fichier (CSV)

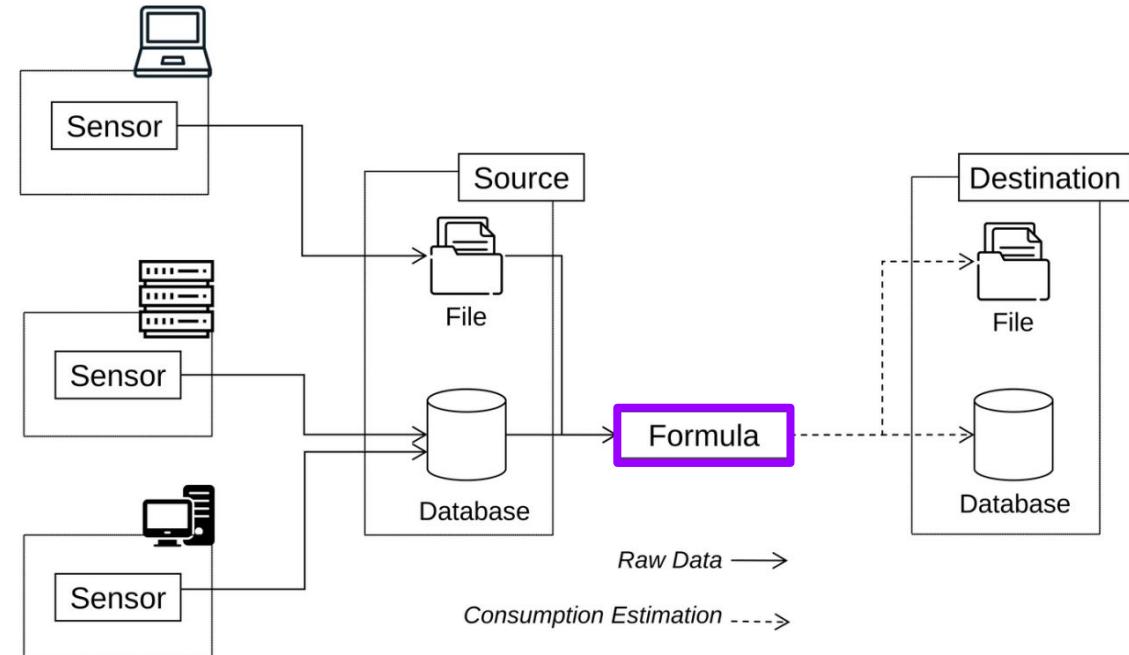


[[PowerAPI Overview](#)]

Architecture de PowerAPI

Formula ⇒ composant permettant de calculer les puissances à partir des données brutes.

Les puissances en sorties sont calculées à partir d'un modèle *Ridge Regression* (nous verrons ça en détails dans les slides suivants).



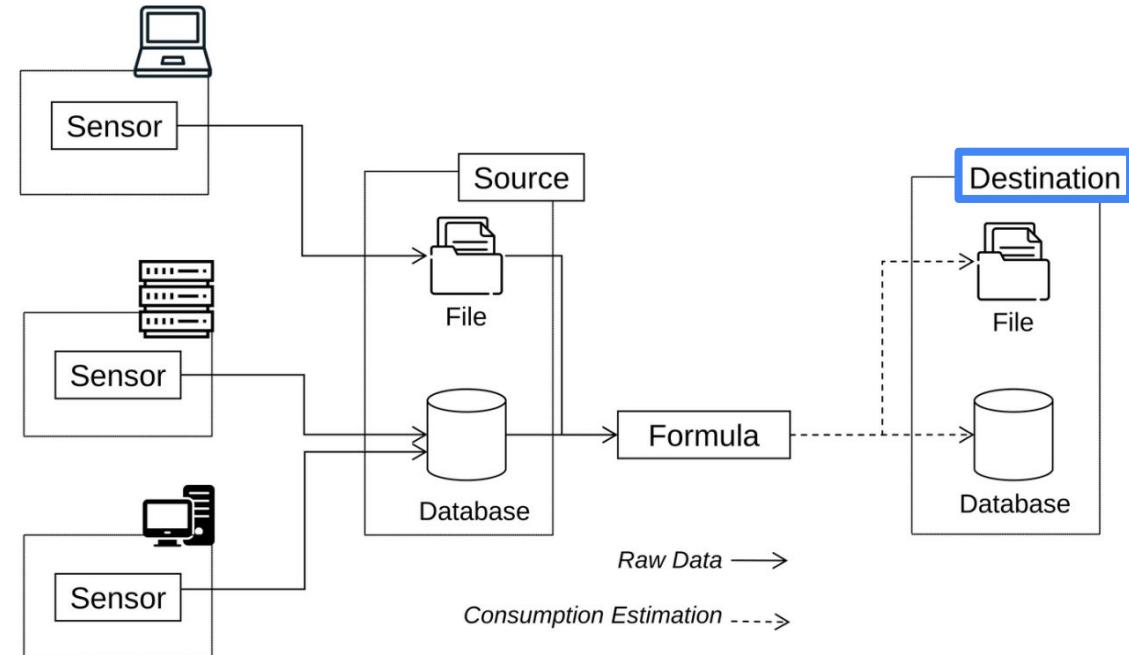
[[PowerAPI Overview](#)]

Architecture de PowerAPI

Destination ⇒

composant permettant de stocker les données traitées. La destination peut-être :

- une base de données (MongoDB, InfluxDB)
- un composant de monitoring (Prometheus)
- un fichier (CSV)



[[PowerAPI Overview](#)]

Architecture de PowerAPI - HWPC Sensor

HardWare Performance Counter (HWPC) Sensor monitore le CPU, des registres spécifiques CPU, les events des cgroups ainsi que la consommation CPU.

- nécessite l'outil perf
- nécessite cgroup V1

Architectures	Group	Events
Intel Sandy Bridge and newer, AMD Zen 2	rapl	RAPL_ENERGY_PKG , RAPL_ENERGY_DRAM
Intel Sandy Bridge and newer, AMD Zen 2	msr	TSC , APERF , MPERF
Intel Skylake, Whiskey Lake, Coffe Lake	core	CPU_CLK_THREAD_UNHALTED:REF_P , CPU_CLK_THREAD_UNHALTED:THREAD_P , LLC_MISSES , INSTRUCTIONS_RETIRED
Intel Sandy Bridge, Comet Lake	core	CPU_CLK_UNHALTED:REF_P , CPU_CLK_UNHALTED:THREAD_P , LLC_MISSES , INSTRUCTIONS_RETIRED
AMD Zen 2	core	CYCLES_NOT_IN_HALT , RETIRED_INSTRUCTIONS , RETIRED_UOPS

Architecture de PowerAPI - HWPC Sensor

HardWare Performance Counter (HWPC) Sensor monitore le CPU, des registres spécifiques CPU, les events des cgroups ainsi que la consommation CPU.

- nécessite l'outil perf ⇒ vu dans les slides précédents
- nécessite les cgroups en V1 (et pas V2) ⇒ fonctionnalité du noyau Linux permettant de monitorer, limiter et isoler l'usage des ressources (CPU, mémoire, disque, etc.) à des processus. Inventé en 2006 par deux ingénieurs chez Google (Paul Ménage et Rohit Seth).

Architecture de PowerAPI - HWPC Sensor

HWPC Sensor est utilisable via :

- l'exécutable [[Github HWPC](#)]
- l'image Docker [[DockerHub](#)], fortement conseillé !

Prend en paramètre un fichier de configuration permettant de définir la source de stockage des données brutes ainsi que le paramétrage des events

Fréquence d'échantillonnage par défaut : 1Hz, évaluée à 10Hz au maximum [[Jay23](#)]

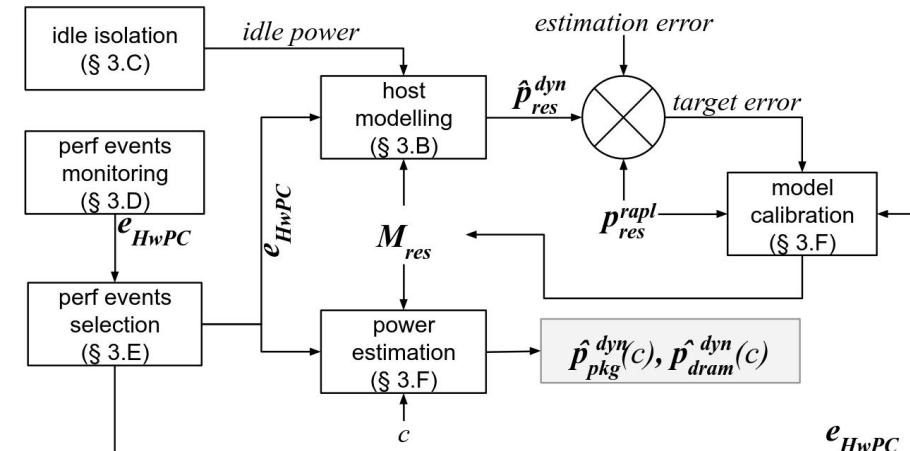
```
{
  "name": "sensor",
  "verbose": true,
  "frequency": 500,
  "output": {
    "type": "mongodb",
    "uri": "mongodb://127.0.0.1",
    "database": "db_sensor",
    "collection": "report_0"
  },
  "system": {
    "rapl": {
      "events": [ "RAPL_ENERGY_PKG" ],
      "monitoring_type": "MONITOR_ONE_CPU_PER_SOCKET"
    },
    "msr": {
      "events": [ "TSC", "APERF", "MPERF" ]
    }
  },
  "container": {
    "core": {
      "events": [
        "CPU_CLK_THREAD_UNHALTED:REF_P",
        "CPU_CLK_THREAD_UNHALTED:THREAD_P",
        "LLC_MISSES",
        "INSTRUCTIONS_RETIRED"
      ]
    }
  }
}
```

Architecture de PowerAPI - SmartWatts

Composant de calcul de la puissance  utilisant les métriques collectées par le HWPC-sensor.

La fréquence d'échantillonnage des puissances est liée à la fréquence de collecte du HWPC-Sensor.

Ajuste un modèle au début de la phase de traitement (*Ridge Regression*) pour estimer la puissance. Calcul une erreur d'estimation pendant le traitement et corrige le modèle si besoin.

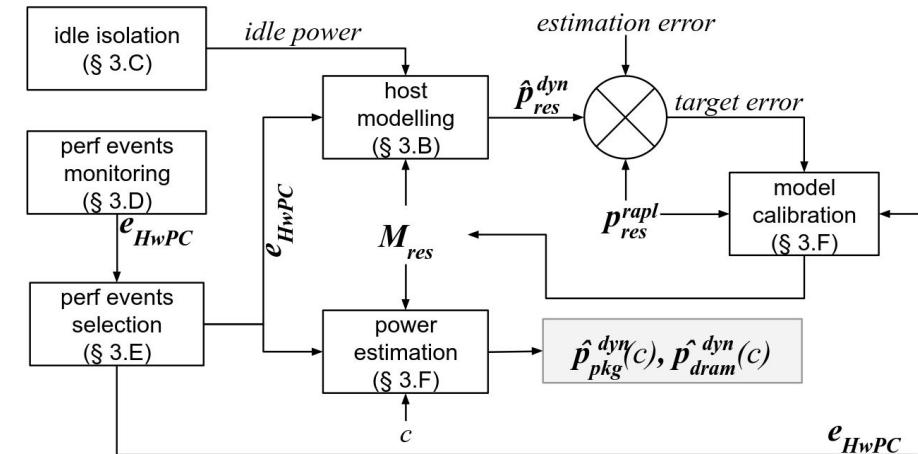


Architecture pour la prédiction des puissances de SmartWatts [Fieni20]

Architecture de PowerAPI - SmartWatts

Composant de calcul de la puissance ⚡ utilisant les métriques collectées par le HWPC-sensor.

⚠ Attention, phase “d’entraînement” sur une centaine de mesures brutes ⇒ pas de puissances estimées sur celles-ci ⇒ phase “d’échauffement”



Architecture pour la prédiction des puissances de SmartWatts [Fieni20]

Architecture de PowerAPI - SmartWatts

SmartWatts est utilisable via :

- l'exécutable [[Github](#) SmartWatts]
- l'image Docker [[DockerHub](#)], fortement conseillé !

Prend en paramètre un fichier de configuration permettant de définir la source des données brutes et la destination des puissances

Fréquence d'échantillonnage par défaut : 1Hz, évaluée à 10Hz au maximum [[Jay23](#)]

```
{  
    "verbose": $bool,  
    "stream": $bool ,  
    "sensor-report-sampling-interval" : $int,  
    "input": {  
        $puller_name: {  
            "model": $type_of_report,  
            "type": $type_of_database,  
            $database_parameters  
        }  
        ... #(Multiple pullers can be used)  
    },  
    "output": {  
        $pusher_name: {  
            "type": $type_of_database  
            $database_parameters  
        }  
        ... #(Multiple pushers can be used)  
    },  
    $processors_if_required  
    $formula_parameters  
}
```

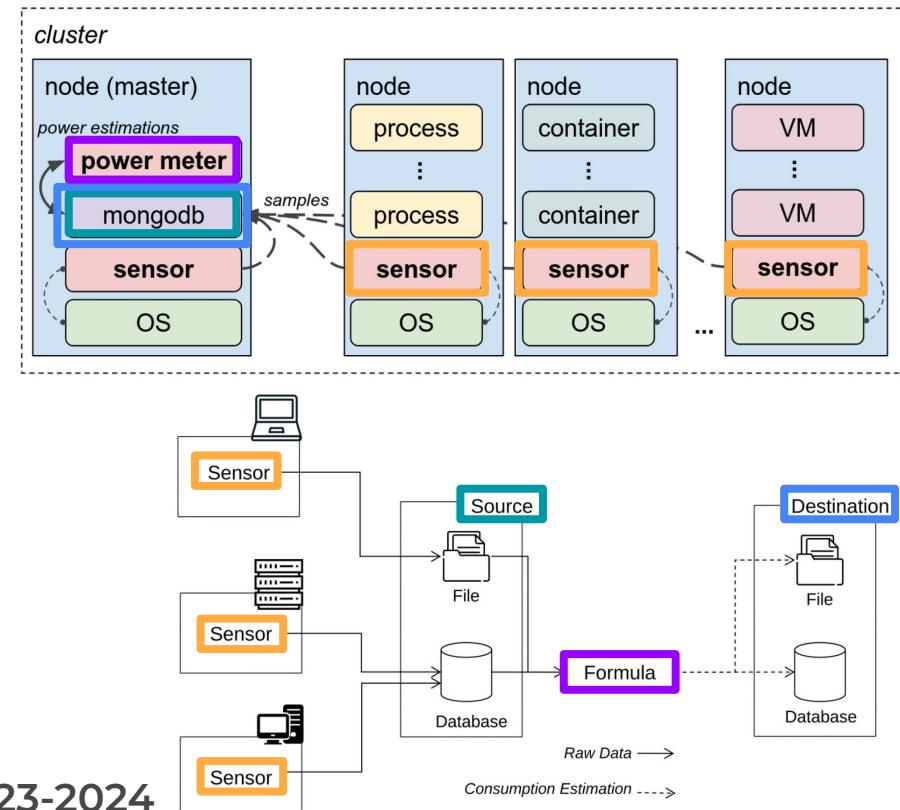
Exemples de déploiement de PowerAPI - Cluster de pré-production / production

Ici sur un cluster de machine (par exemple en pré-production / production) :

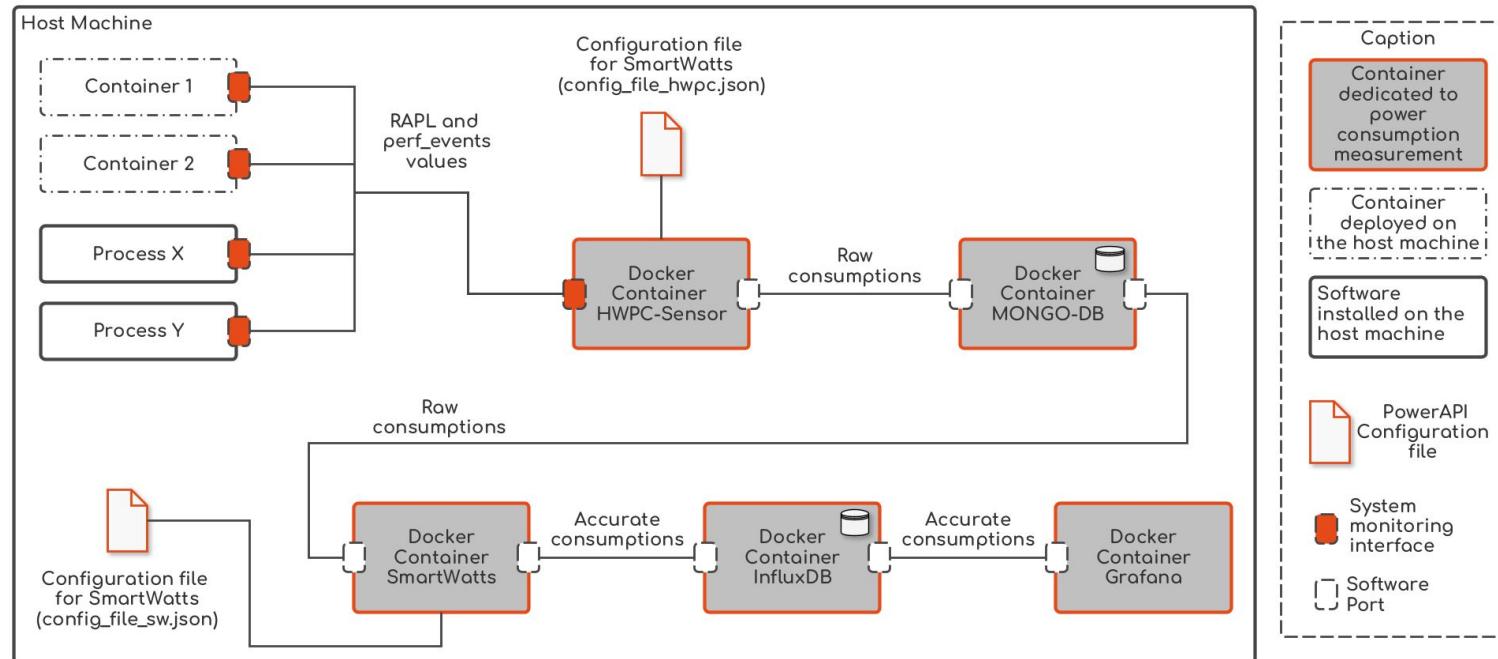
- un **HWPC-sensor** par machine (node)
- une machine (master) de **collecte des données brutes** (MongoDB) + **calcul des puissances** (Smartwatts) + **injection des résultats dans une base de données** (MongoDB)

Il est possible de dissocier les éléments sur 3 machines :

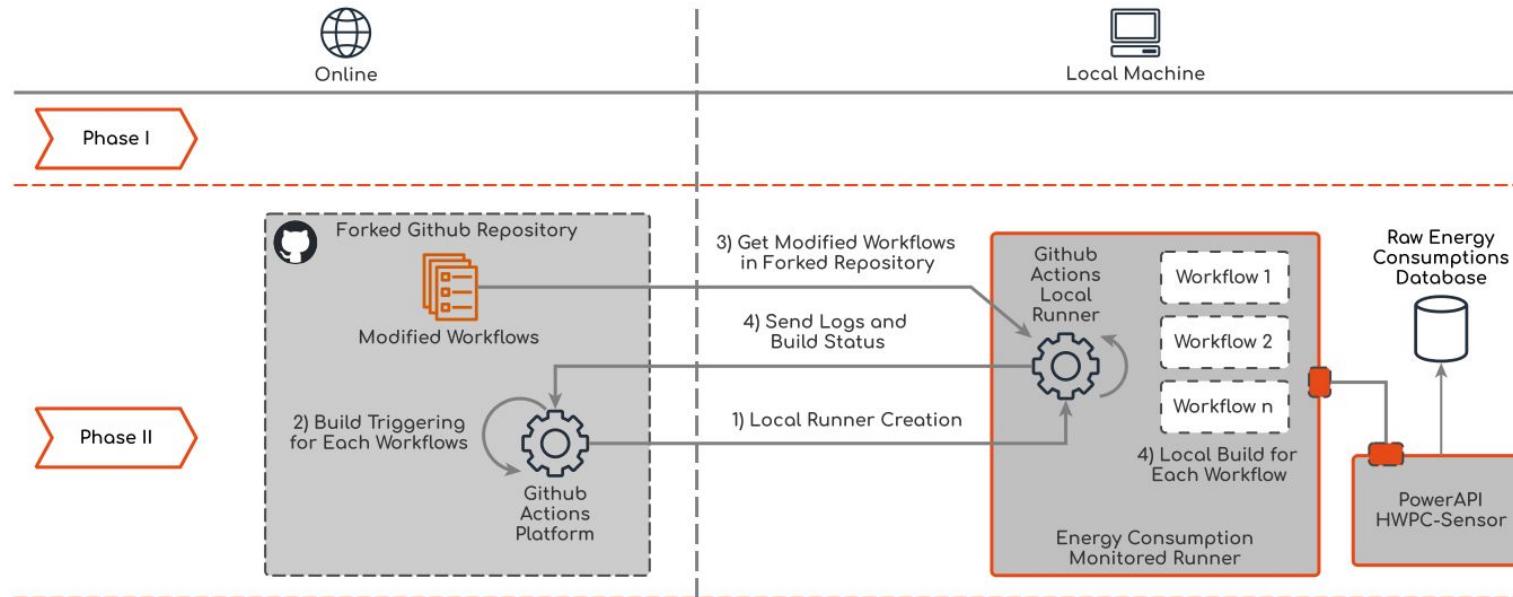
- une machine de collecte des données (**source**)
- une machine de calcul avec SmartWatts (**formula**)
- une machine de stockage des puissances (**destination**)



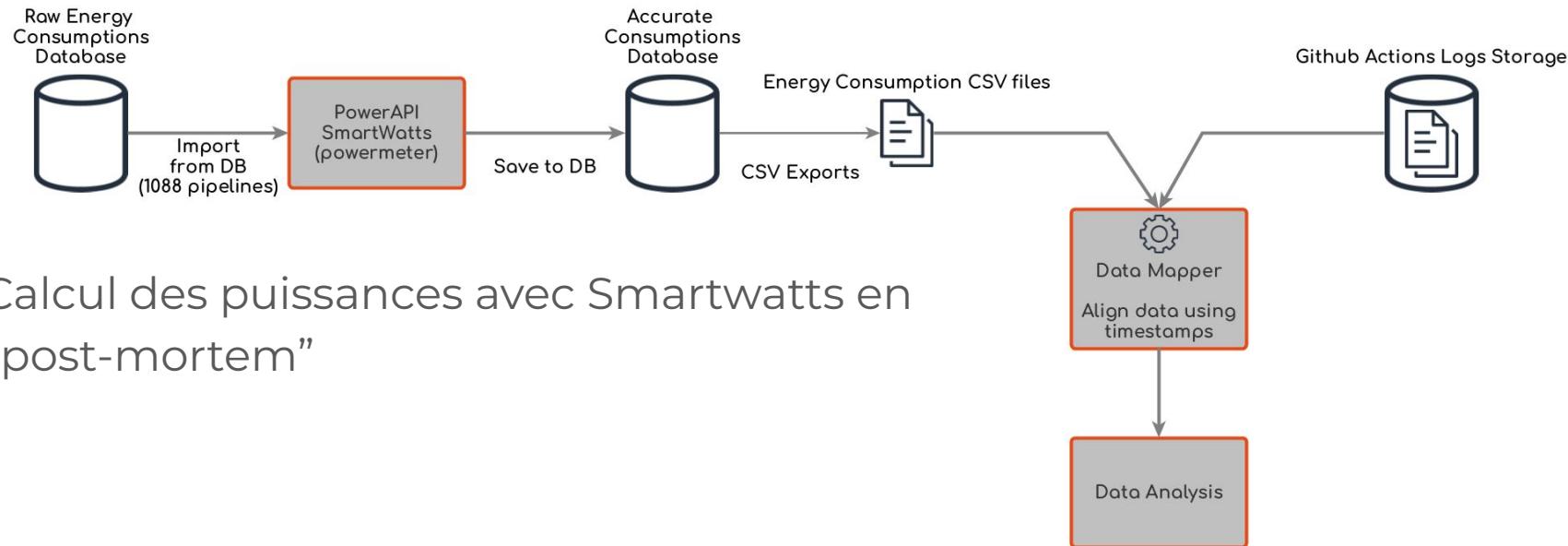
Exemples de déploiement de PowerAPI - Mesure de consommation de processus / conteneurs avec une stack Docker PowerAPI



Exemples de déploiement de PowerAPI - Mesure de consommation de pipelines DevOps avec une stack Docker PowerAPI



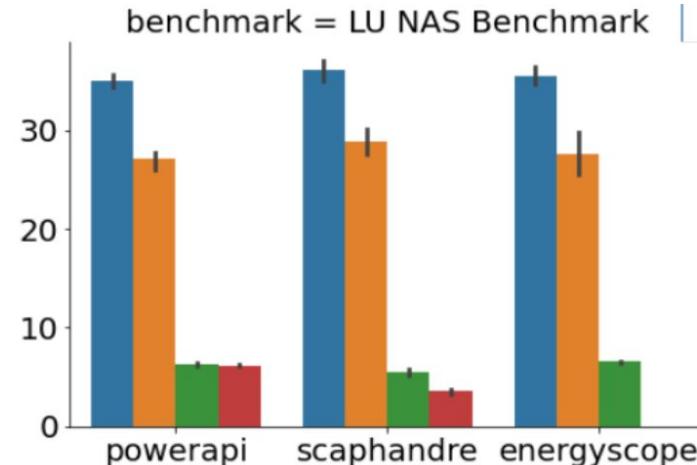
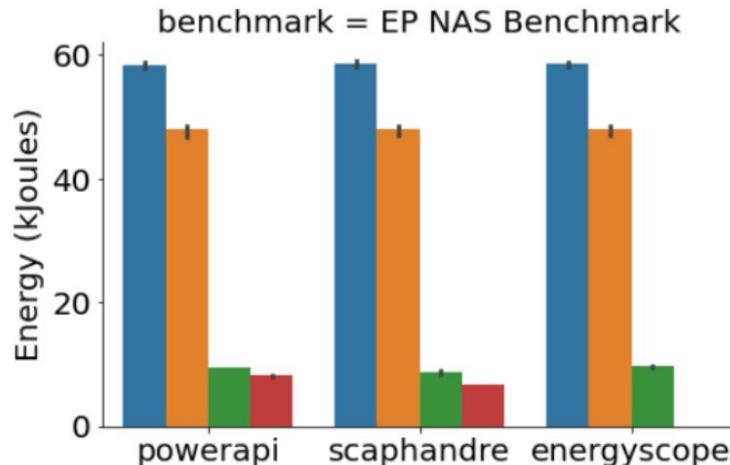
Exemples de déploiement de PowerAPI - Mesure de consommation de pipelines DevOps avec une stack Docker PowerAPI



Calcul des puissances avec Smartwatts en
“post-mortem”

Mesures matérielles VS wattmètres logiciels

[[Ostapenco et al., "Une comparaison expérimentale de wattmètres logiciels : focus sur le CPU". 2022](#)]

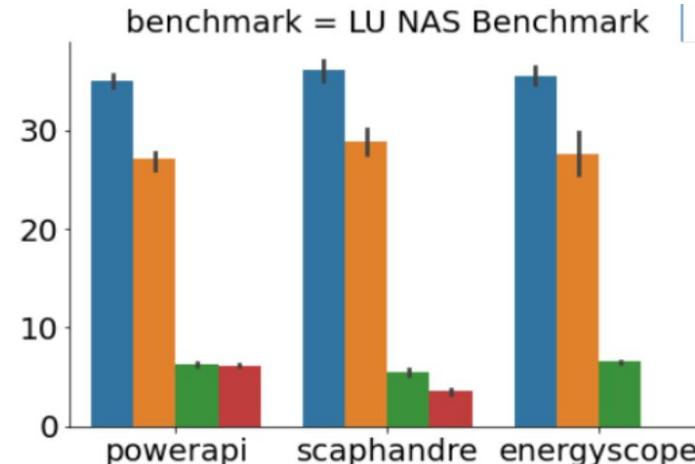
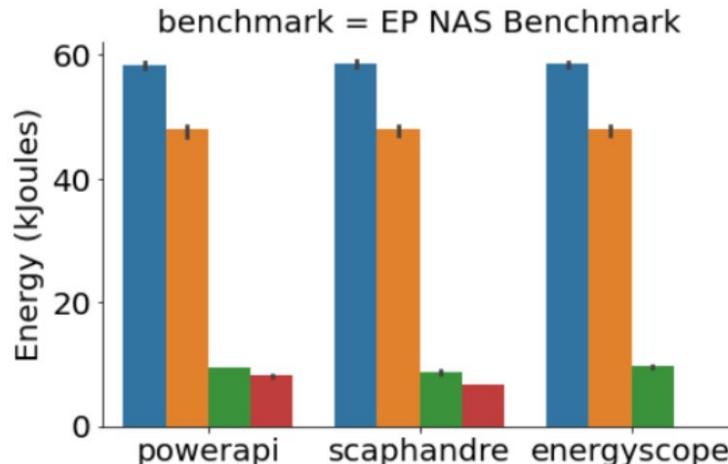


Le wattmètre logiciel ne va prendre en compte qu'une petite partie de la consommation (CPU et RAM sur les serveurs).

Il ne tient pas compte d'autres aspects comme les I/Os réseau ou disque, ou encore les autres périphériques connectés sur la carte mère et potentiellement utilisés également.

Mesures matérielles VS wattmètres logiciels

[\[Ostapenco et al., "Une comparaison expérimentale de wattmètres logiciels : focus sur le CPU", 2022\]](#)

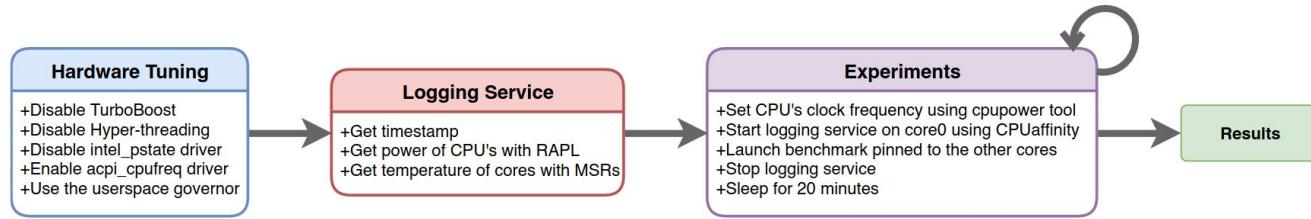


Cependant, les wattmètres logiciels sont à une granularité qui intéresse fortement le développeur 🧑‍💻 ! L'approche matérielle est donc complémentaire à l'approche purement logicielle !

Tests matériels avec mesures logicielles - Des CPUs **verts** pas si **verts** !

Comparaison matérielle sur l'
énergie entre 2 types de CPU
sur des serveurs Dell
PowerEdge R630 :

- **Xeon E5-2630L v4**
(Broadwell, 1.80GHz
nominal frequency, 2
CPUs/node, 10
cores/CPU) ⇒ **Low
Power Processor**
- **Xeon E5-2630 v4**
(Broadwell, 2.20GHz
nominal frequency, 2
CPUs/node, 10
cores/CPU)



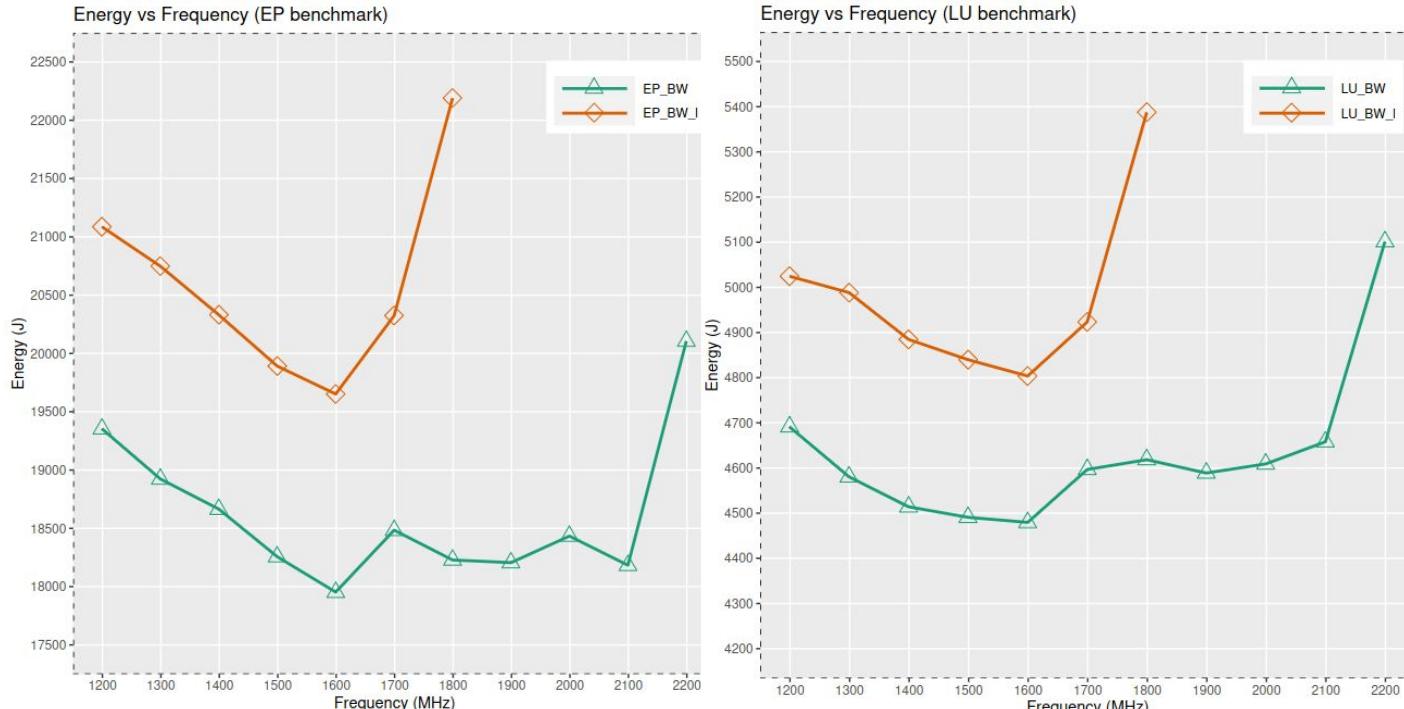
Protocole d'expérimentation utilisé

[Rao Vaddina et al., "Experimental Workflow for
Energy and Temperature Profiling on HPC Systems",
2022]

Tests matériels avec mesures logicielles - Des CPUs **verts** pas si **verts** !

Comparaison matérielle sur l'énergie entre 2 types de CPU sur des serveurs Dell PowerEdge R630 :

- **Xeon E5-2630L v4**
⇒ Low Power Processor ⇒ BW_1
- **Xeon E5-2630 v4**
⇒ BW



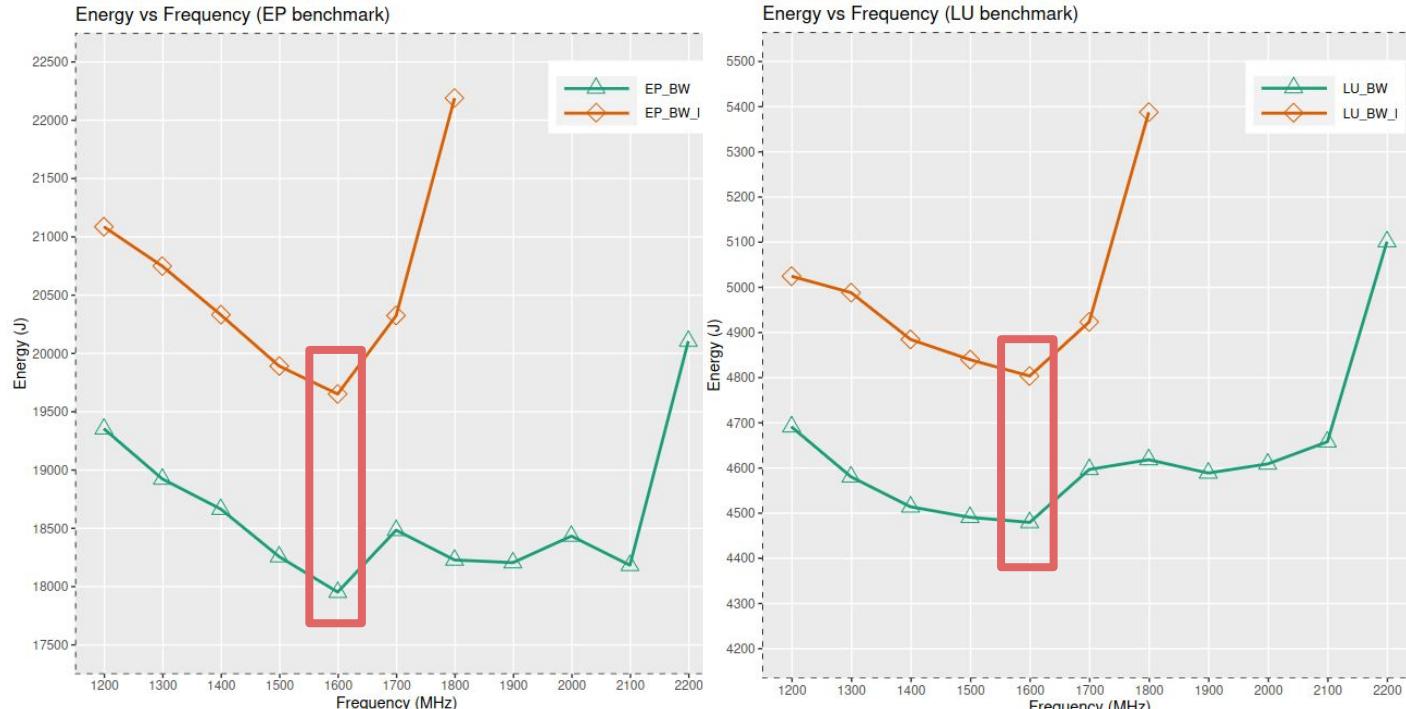
[Kao Vannara et al., Experimental Evaluation for
Energy and Temperature Profiling on HPC Systems”,
2022]

Tests matériels avec mesures logicielles - Des CPUs **verts** pas si **verts** !

Comparaison matérielle sur l'énergie entre 2 types de CPU sur des serveurs Dell PowerEdge R630 :

- **Xeon E5-2630L v4**
⇒ Low Power Processor ⇒ BW_1
- **Xeon E5-2630 v4**
⇒ BW

Point d'inflexion où la consommation est minimale à une certaine fréquence

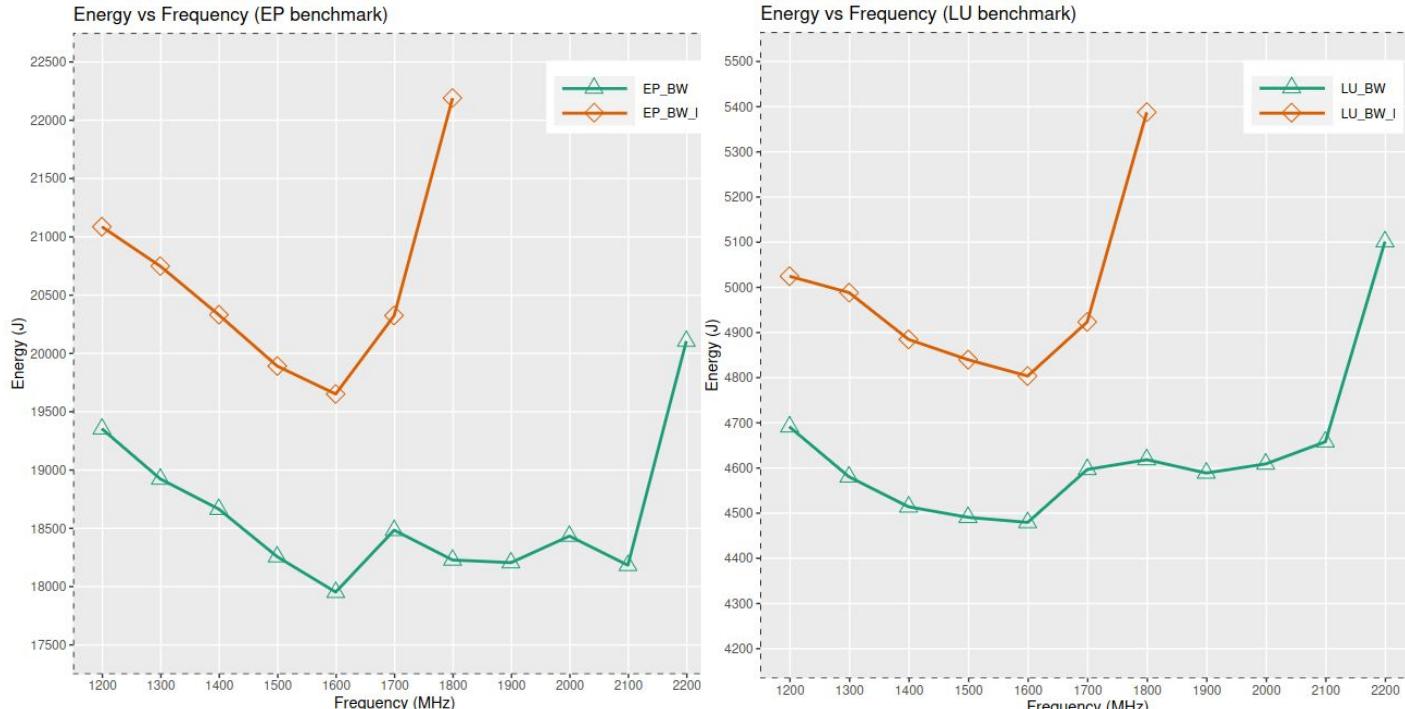


[Kao Vannan et al., Experimental Evaluation for
Energy and Temperature Profiling on HPC Systems”,
2022]

Tests matériels avec mesures logicielles - Des CPUs **verts** pas si **verts** !

Comparaison matérielle sur l'énergie entre 2 types de CPU sur des serveurs Dell PowerEdge R630 :

- **Xeon E5-2630L v4**
⇒ Low Power Processor ⇒ BW_1
- **Xeon E5-2630 v4**
⇒ BW

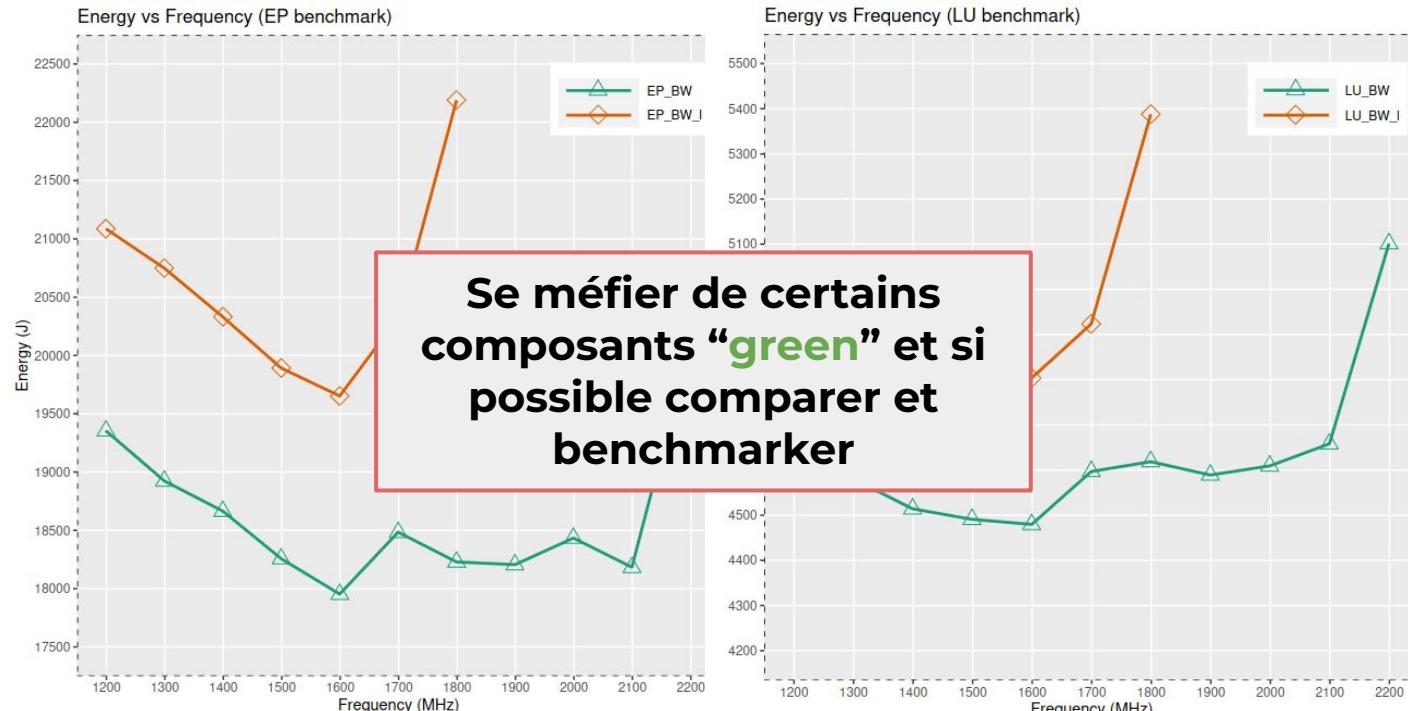


[Kao Vannan et al., Experimental Evaluation for
Energy and Temperature Profiling on HPC Systems”,
2022]

Tests matériels avec mesures logicielles - Des CPUs **verts** pas si **verts** !

Comparaison matérielle sur l'énergie entre 2 types de CPU sur des serveurs Dell PowerEdge R630 :

- Xeon E5-2630L v4
⇒ Low Power Processor ⇒ BW_1
- Xeon E5-2630 v4
⇒ BW

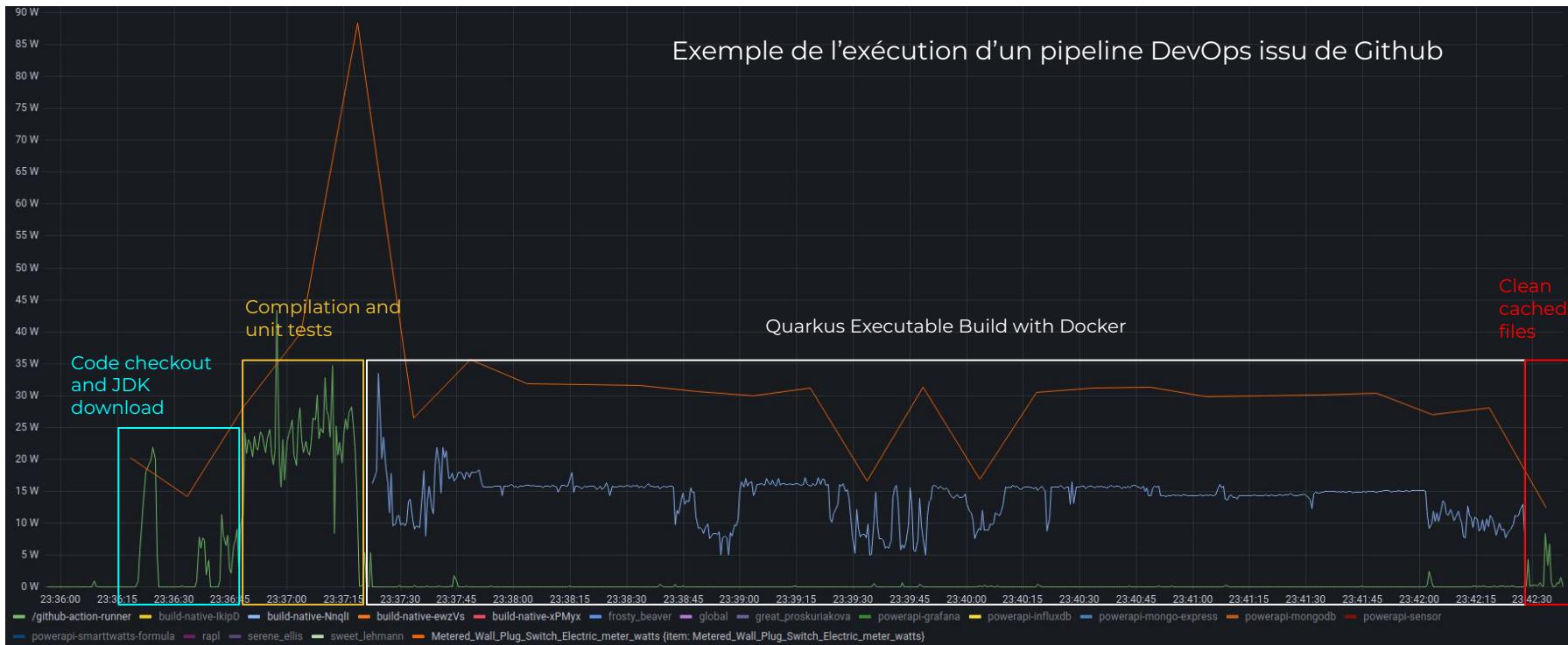


[Kao Vannan et al., Experimental Evaluation for
Energy and Temperature Profiling on HPC Systems”,
2022]

Complémentarité des mesures matérielles et logicielles



Complémentarité des mesures matérielles et logicielles



Écarts de mesures (matérielles/logicielles) et types de tâches logicielles

Tâches intensives pour le CPU/GPU ⇒ mesurables avec les approches matérielles et logicielles



Tâches intensives en I/Os ⇒

- DRAM ⇒ mesurables avec les **approches logicielles et matérielles**
- Disque ⇒ mesurables uniquement avec des **approches matérielles**
- Réseau ⇒ mesurables uniquement avec des **approches matérielles** (et difficile à quantifier pour du Web)



Tâches intensives I/Os et CPU/GPU ⇒ **demande une complémentarité des deux approches**



Estimateurs CO₂ logiciels

Estimateur CO2 (basé sur des mesures logicielles)

Estimateur CO2 = outil logiciel permettant d'estimer les émissions de CO2 d'un logiciel/code à partir des collectes de mesures (RAPL / NVML) et d'une valeur CO2 définie.

Carbontracker

Carbontracker est un estimateur permettant d'évaluer les émissions de CO₂ des modèles de Deep Learning entraînés sur CPU (RAPL) ou GPU (NVML) [[Github Carbontracker](#)]

L'énergie consommée sur les epochs est multipliée par un PUE afin d'avoir une estimation “globale” en tenant compte des “rendements”

Les émissions sont la multiplication de l'énergie par l'intensité carbone.

consumption E as

$$E = \text{PUE} \sum_{e \in \mathcal{E}} \sum_{d \in \mathcal{D}} P_{avg,de} T_e \quad (2)$$

where $P_{avg,de}$ is the average power consumed by device $d \in \mathcal{D}$ in epoch $e \in \mathcal{E}$, and T_e is the duration of epoch e .

$$\text{Carbon Footprint} = \text{Energy Consumption} \times \text{Carbon Intensity.} \quad (3)$$

[[Anthony et al., “Carbontracker: Tracking and Predicting the Carbon Footprint of Training Deep Learning Models”, 2020](#)]

Carbontracker

Carbontracker est un estimateur permettant d'évaluer les émissions de CO₂ des modèles de Deep Learning entraînés sur CPU (RAPL) ou GPU (NVML) [[Github Carbontracker](#)]

⇒ Fréquence d'échantillonnage des mesures = **1 mesure toutes les 10 secondes**

consumption E as

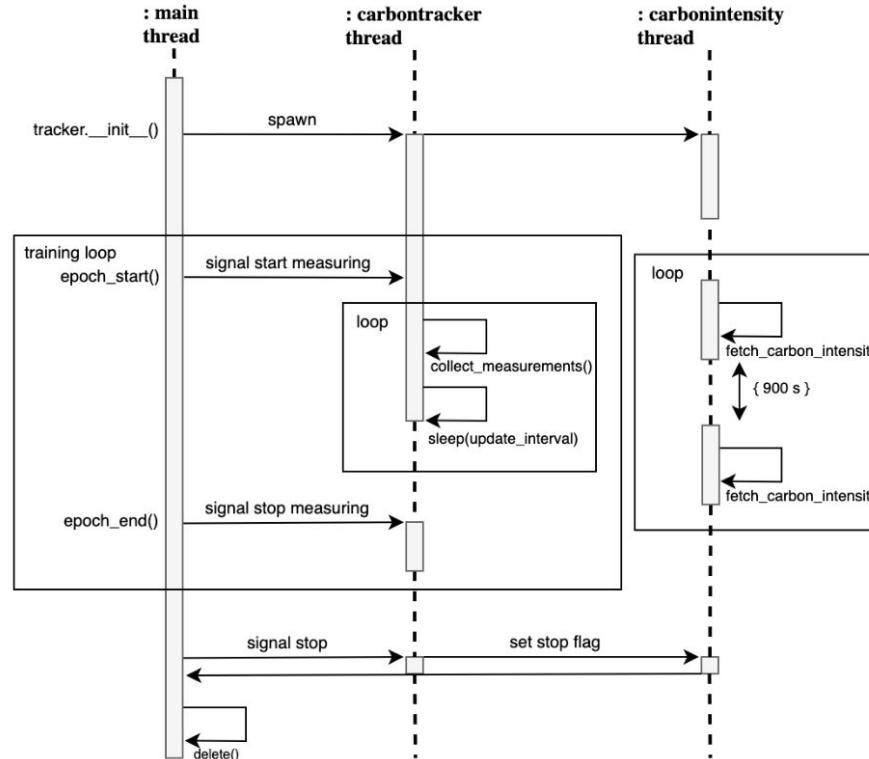
$$E = \text{PUE} \sum_{e \in \mathcal{E}} \sum_{d \in \mathcal{D}} P_{avg,de} T_e \quad (2)$$

where $P_{avg,de}$ is the average power consumed by device $d \in \mathcal{D}$ in epoch $e \in \mathcal{E}$, and T_e is the duration of epoch e .

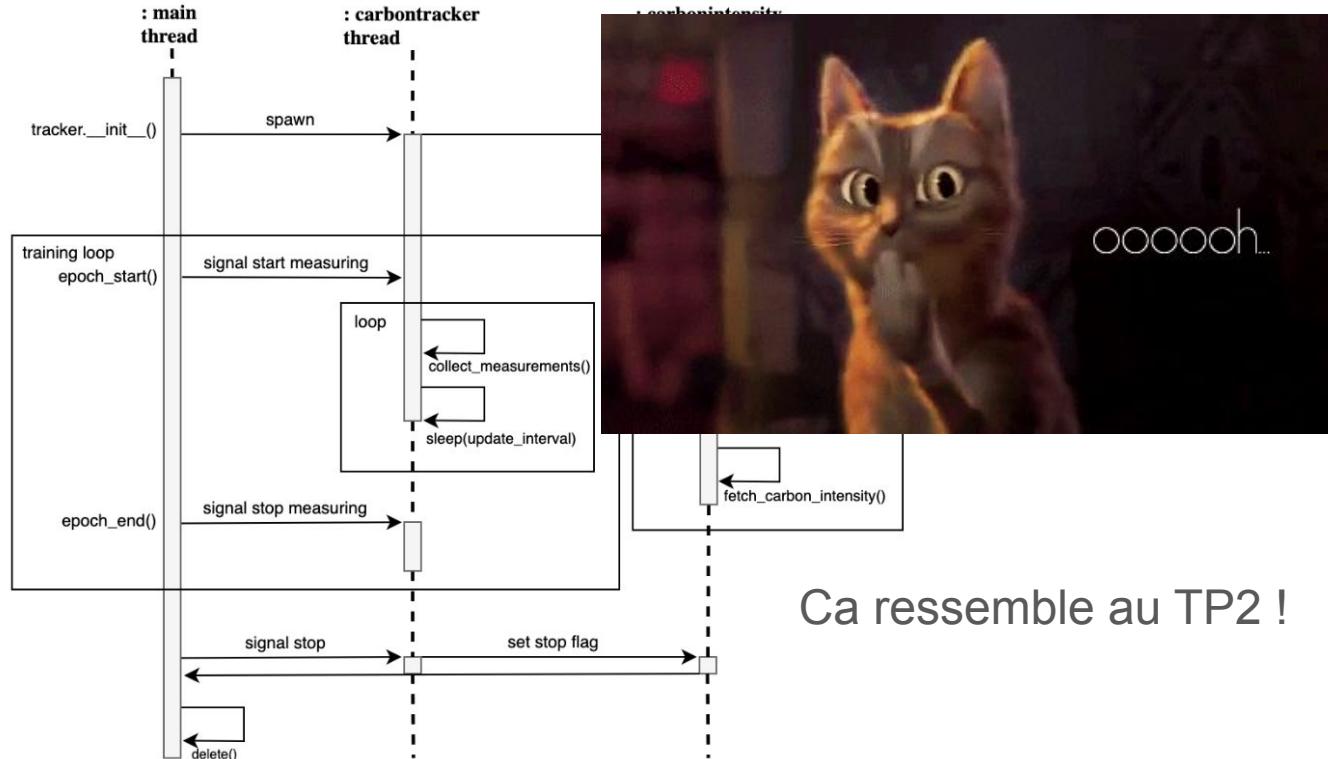
$$\text{Carbon Footprint} = \text{Energy Consumption} \times \text{Carbon Intensity.} \quad (3)$$

[[Anthony et al., "Carbontracker: Tracking and Predicting the Carbon Footprint of Training Deep Learning Models", 2020](#)]

Carbon Tracker - mécanisme de capture de l'énergie



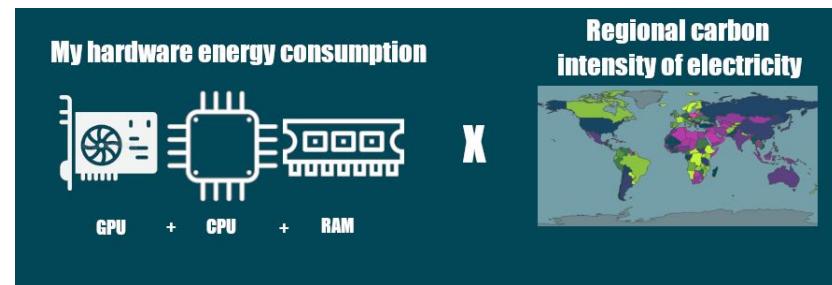
Carbon Tracker - mécanisme de capture de l'énergie



Ca ressemble au TP2 !

Code Carbon

Code Carbon est plus qu'un simple estimateur CO₂ c'est une plateforme complète permettant d'évaluer les émissions des sous différents angles et régions du monde. [[site web](#)]



Code Carbon

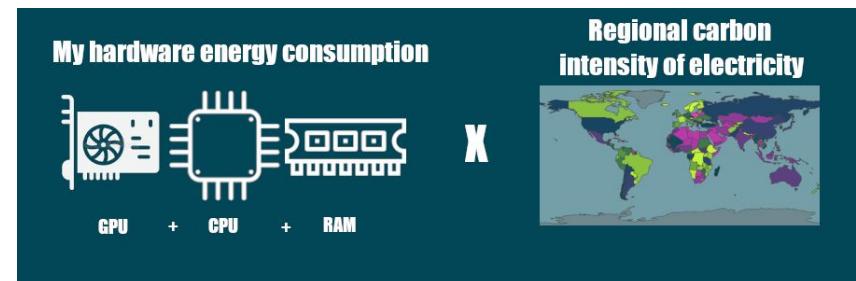
<https://youtu.be/uSHKUzjKOsc>



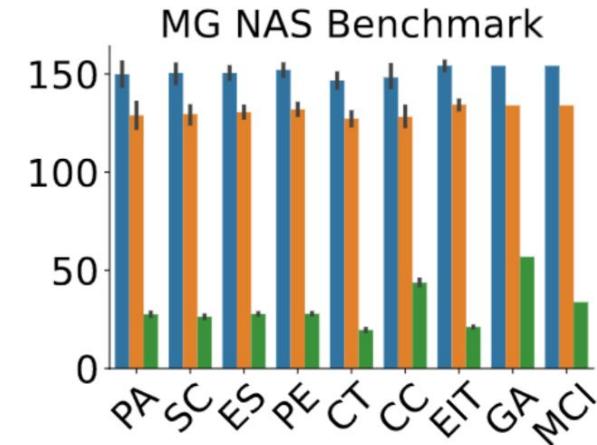
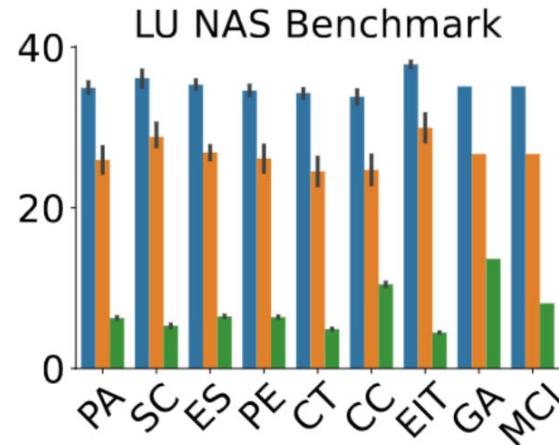
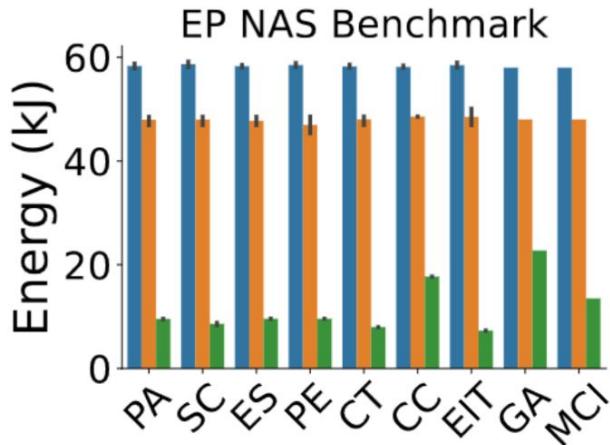
Code Carbon

Utilise le RAPL et NVML pour capture la puissance. Utilise également le mix énergétique pour calculer au mieux les émissions.

⇒ Fréquence d'échantillonnage des mesures = **1 mesure toutes les 15 secondes**



Comparaison des estimateurs CO2/energy profilers



PA = PowerAPI

External power meter

SC = Scaphandre

CT = Carbontracker

CC = CodeCarbon

(a) [CPU Ber

"Code Carbon tends to report power consumption values almost twice as high as other tools based on the same interface for obtaining consumption values (Intel RAPL)" [Jay23] ⇒ Se méfier des sur/sous-estimations des estimateurs/profilers

Conclusion

- Divers moyens de mesures (matériels ou logiciels) pour des granularités différentes
- Complémentarité des moyens de mesure matériels et logiciels
- Les profilers/estimateurs purement logiciels ne prennent en compte que la consommation d'une partie de la machine dans un contexte de tâche particulier
- Bien sélectionner son profiler/estimateur est déterminant afin d'éviter les sur-estimations/sous-estimations
- Prendre en compte le contexte global pour l'utilisation
 - quel composant logiciel et matériel ?
 - quelle fréquence de mesure ?
 - sur quel type de machine ?
 - tenir compte de la complexité de mise en œuvre et du stockage des données