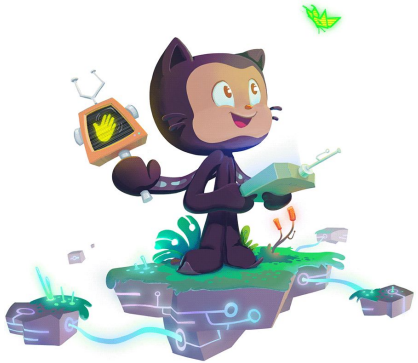
Measuring energy consumption in the complex world of DevOps



Quentin Perez, Associate Professor at INSA Rennes
Romain Lefeuvre, Ph.D student at the University of Rennes

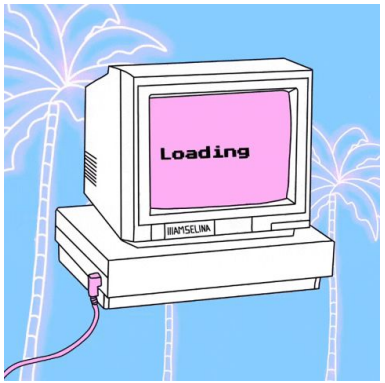
Information and Communication Technologies (ICTs) CO2 Emissions

World energy consumption: 2,848 TWh in 2019 (IEA)

⇒ +1.7% compared to 2018

⇒ +44% compared to 2000

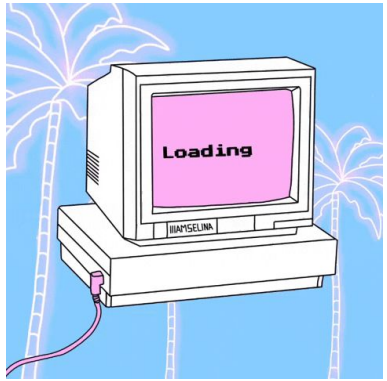
ICTs CO2 emissions: 1,8% to 3,9% of global CO2 emissions in 2020



The figures given here are taken from the publication by Freitag *et al.*, *"The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations"*, 2021 [Freitag2021]

Information and Communication Technologies (ICTs) CO2 Emissions

ICTs CO2 emissions: 1,8% to 3,9% of global CO2 emission in 2020 [Freitag 2021]

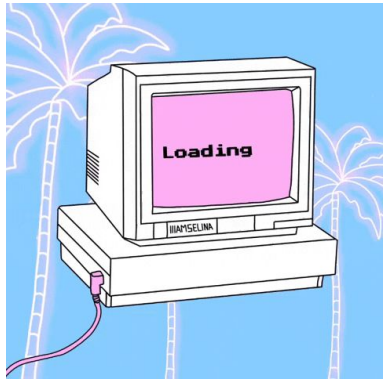


CO2 emissions from aviation in 2018: 2,4%

[*Lee et al., "The contribution of global aviation to anthropogenic climate forcing for 2000 to 2018", 2018*]

Information and Communication Technologies (ICTs) CO2 Emissions

ICTs CO2 emissions: 1,8% to 3,9% of global CO2 emission in 2020 [Freitag2021]



CO2 emissions from aviation in 2018: 2,4%

[*Lee et al., "The contribution of global aviation to anthropogenic climate forcing for 2000 to 2018", 2018*]

C02 emissions for ICTs are very hard to quantify "precisely"

Quantifying ICTs' CO2 Emissions

Software is continuously evolving

C02 emissions for ICTs are very hard to quantify “precisely”, especially for software

- Scope: software, peripherals, users, networks, hardware workload, etc.
- Share of the material life cycle considered: manufacture, use, recycling, reuse, etc.
- Electrical mix
- Share of software life cycle

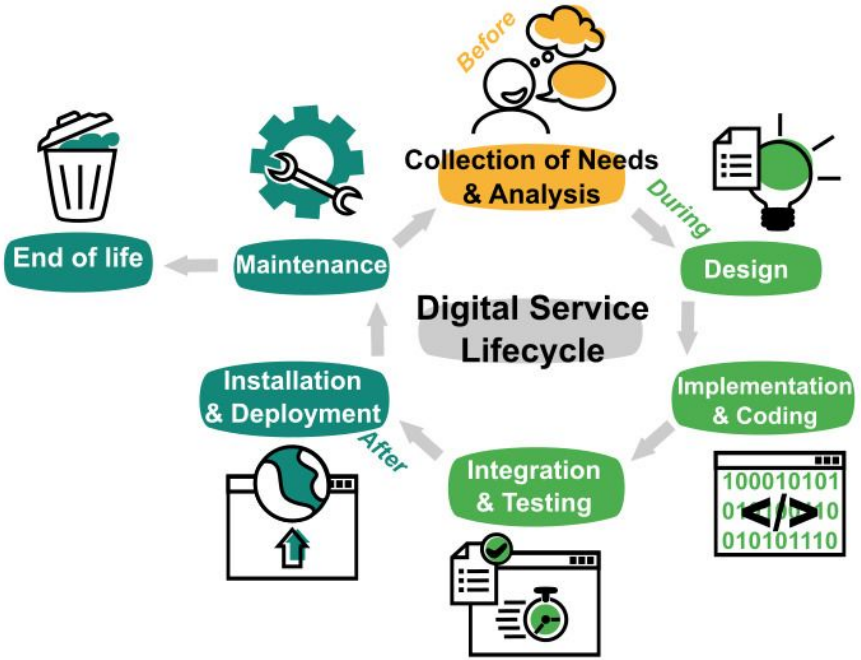


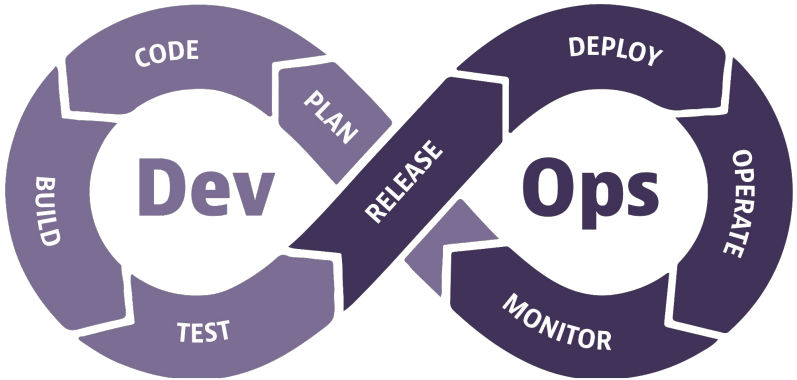
Figure source - Bonamy, & Boudinet, & Bourgès, & Dassas, & Lefèvre, & Ninassi, & Vivat,. (2022). L'écoconception d'un service numérique : des actions pour réduire l'impact environnemental du numérique. Bulletin 1024. 59-68. 10.48556/SIF.1024.19.59.

Quantifying ICTs' CO2 Emissions

C02 emissions for ICTs are very hard to quantify “precisely”, especially for software

- Scope: software, peripherals, users, networks, hardware workload, etc.
- Share of the material life cycle considered: manufacture, use, recycling, reuse, etc.
- Electrical mix
- Share of software life cycle

Iterative and incremental cycle makes calculations more complex

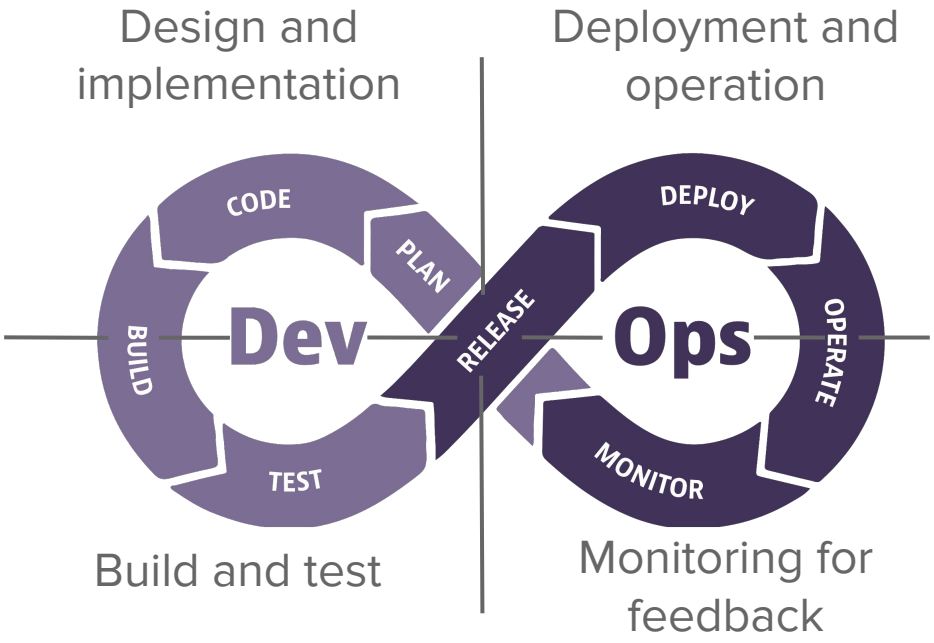


Quantifying ICTs' CO2 Emissions

C02 emissions for ICTs are very hard to quantify “precisely”, especially for software

- Scope: software, peripherals, users, networks, hardware workload, etc.
- Share of the material life cycle considered: manufacture, use, recycling, reuse, etc.
- Electrical mix
- Share of software life cycle

Iterative and incremental cycle makes calculations more complex

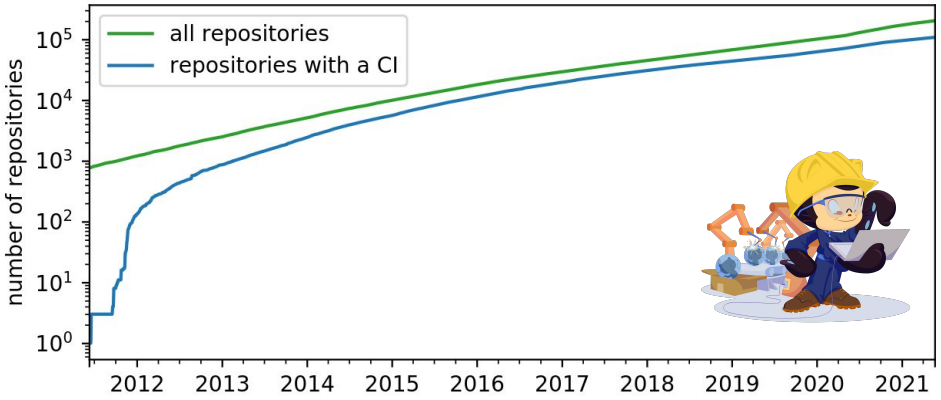


An example of DevOps practice: CI/CD systems

Information and Communication Technologies (ICT) ⇒ **1.8%** to **3.9%** of the greenhouse gases [1]

Global acceleration of software development and delivery ⇒ continuous testing and deployment of software systems. [2]

Democratization of CI/CD pipelines with Gitlab and GitHub services in clouds (Github Actions)⇒ in 2022, **30% of the Github projects have a pipeline of CI/CD** [3]

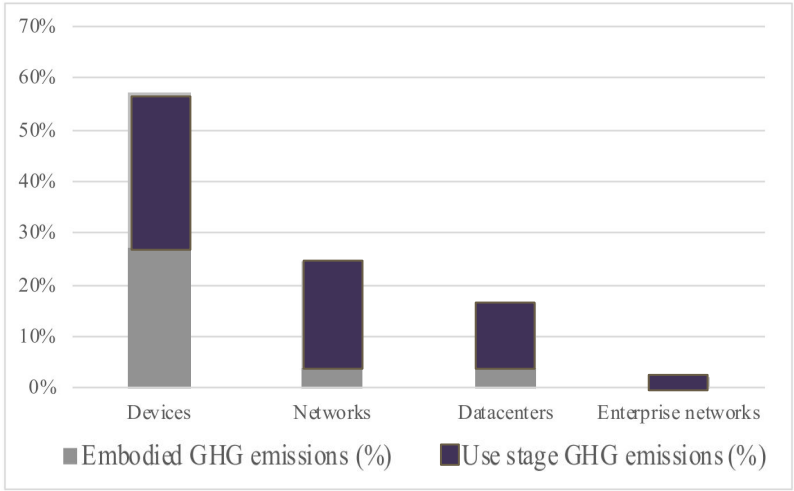


[1] Freitag et al., "The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations", 2021
[2] Jez Humble and Gene Kim. 2018. Accelerate: The science of lean software and devops: Building and scaling high performing technology organizations. IT Revolution.
[3] Mehdi Golzadeh, Alexandre Decan, and Tom Mens. 2022. On the rise and fall of CI services in GitHub. In IEEE International Conference on Software Analysis, Evolution and Reengineering, SANER 2022, Honolulu, HI, USA, March 15-18, 2022. IEEE, 662–672

Energy efficiency and CO2 emission

ICT share on global electricity usage :

- **4%** in **2020** [1]
- might represent **21%** in **2030** [2]



Total ICT sector carbon footprint 2020 [2]

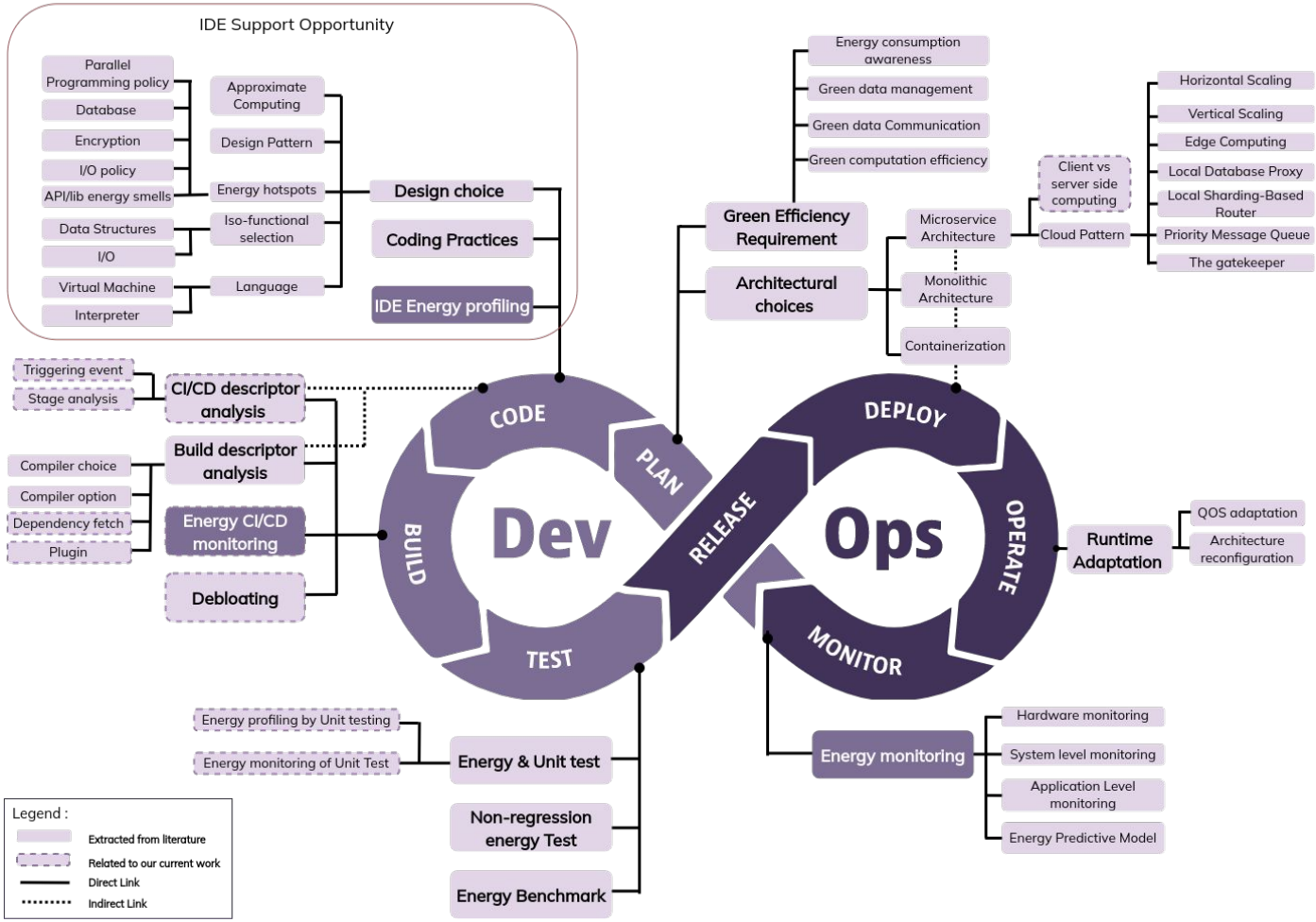
Energy efficiency is **necessary**, but it's only one of the requirements for sustainable software.

Jevons paradox ...

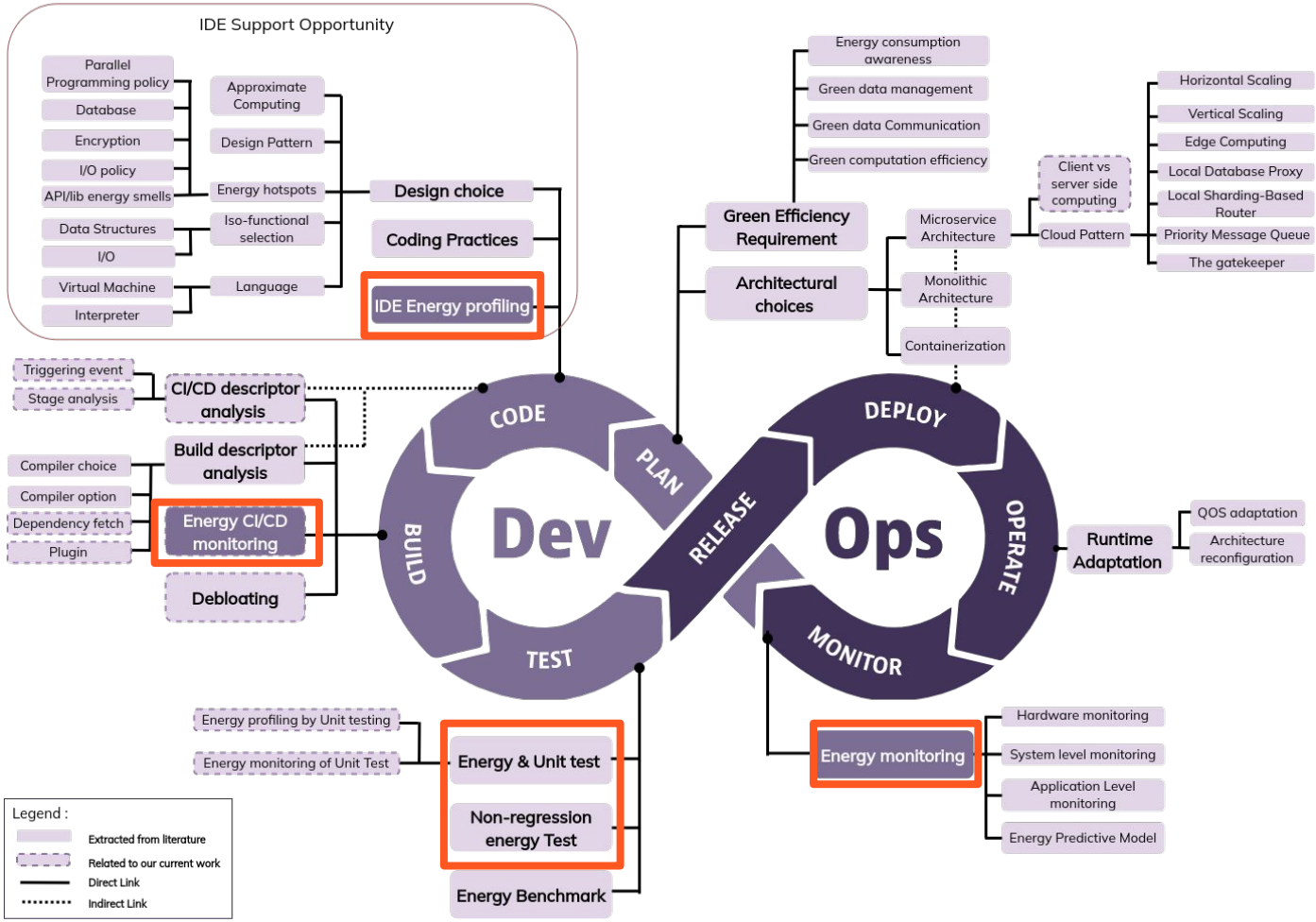
[1] A. S. G. Andrae and T. Edler, “On Global Electricity Usage of Communication Technology: Trends to 2030,” Challenges, vol. 6, no. 1, Art. no. 1, Jun. 2015

[2] J. Malmodin, N. Lövehagen, P. Bergmark, and D. Lundén, “ICT Sector Electricity Consumption and Greenhouse Gas Emissions – 2020 Outcome.” Rochester, NY, Apr. 20, 2023. doi: 10.2139/ssrn.4424264.

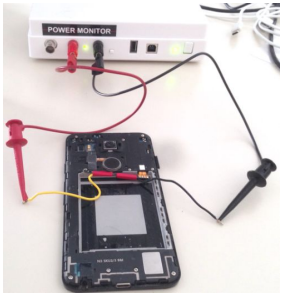
Energy efficiency preoccupation within DevOps lifecycle



Energy efficiency preoccupation within DevOps lifecycle



Two major ways of measuring energy: Hardware level



Power meter — Electrical socket

Power Distributed Units — Group of electrical socket

Monitor Circuit-Breaker — Electrical Circuit

Hardware Level

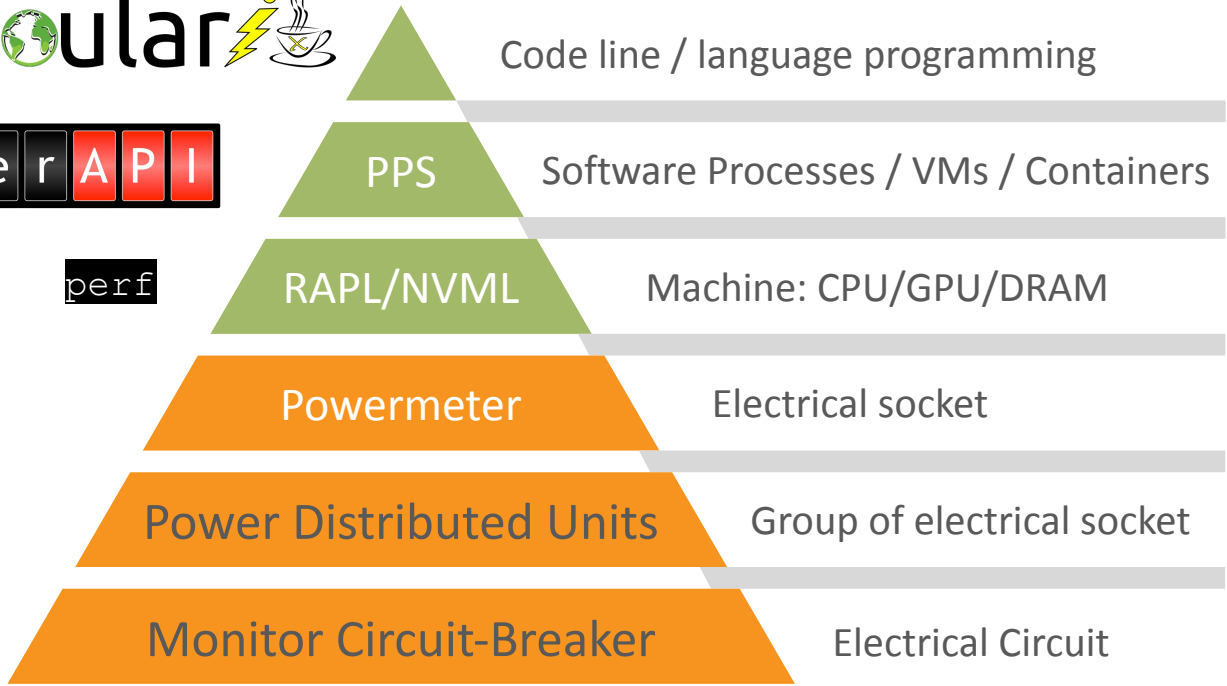
Two major ways of measuring energy: Software level

KEPLER

Joular

PowerAPI

perf



Hardware Level

Software Level

Power Profiling Software (PPS)

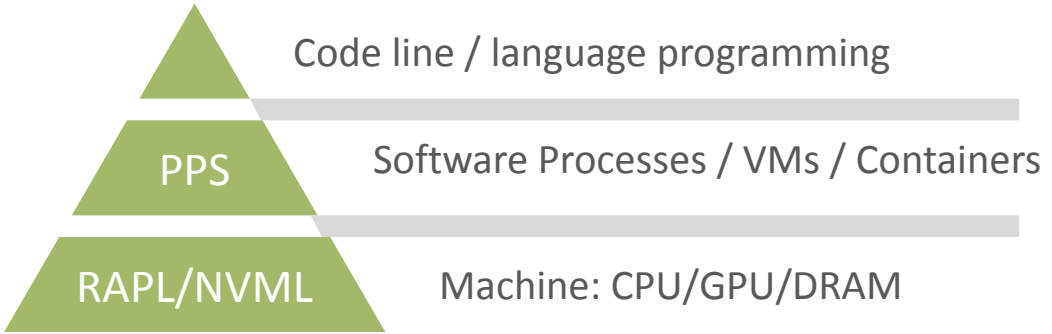
"Tools returning the power profile of a program and that are based on the Intel RAPL and/or Nvidia NVML interface."

Intel RAPL and Nvidia NVML are interfaces providing by hardware manufacturers on their chips to measure the power at a *t* moment

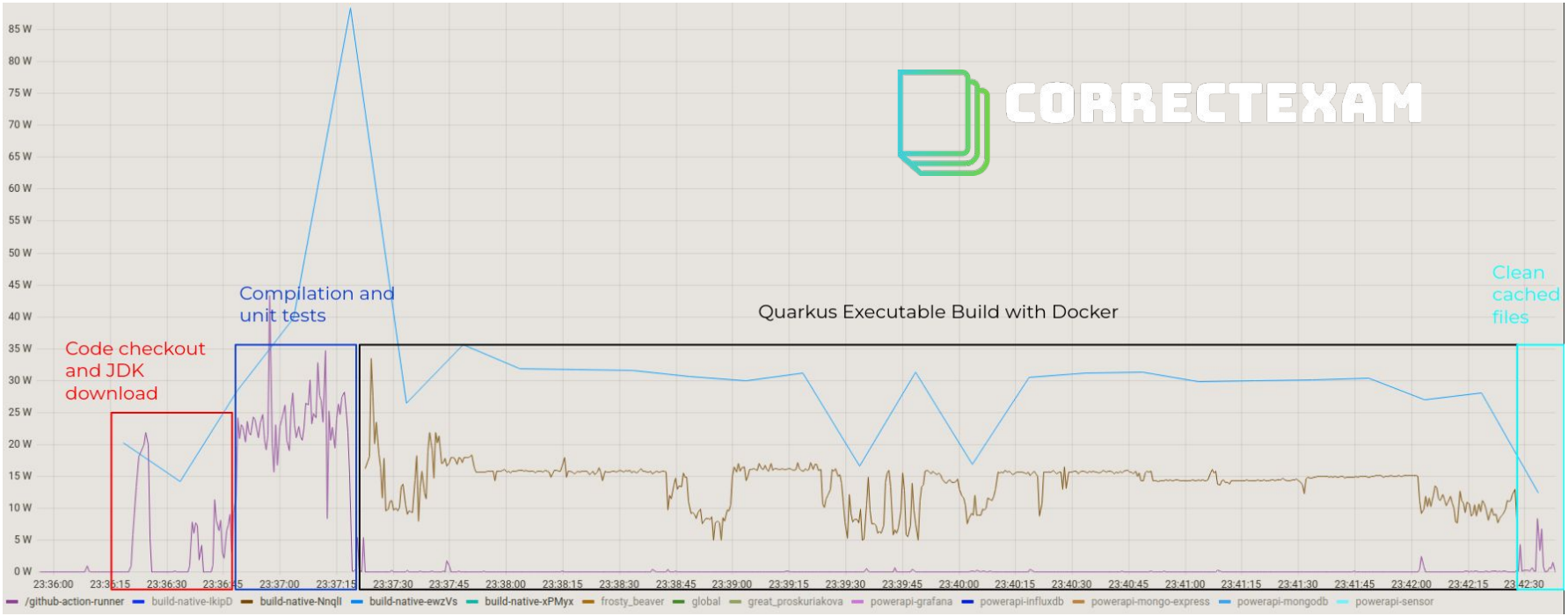
https://inria.hal.science/hal-04030223v2/document#table.caption.1

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.orgerie@irisa.fr
§ *OVHCloud, France* - benjamin.fichel@ovhcloud.com



Power Profiling Software: an example of profile

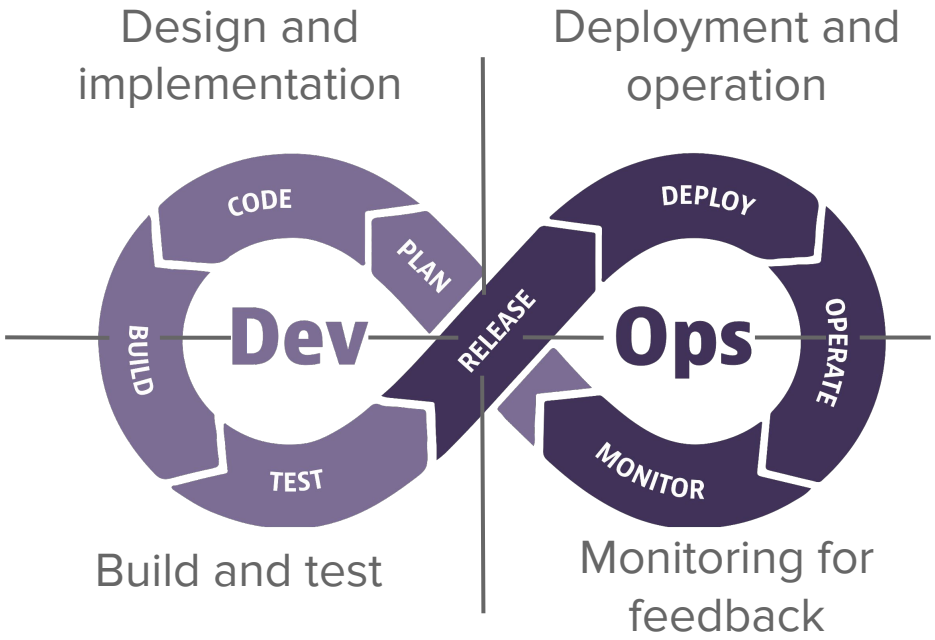


Consumption profile of maven.yml workflow from the CorrectExam project

https://github.com/correctexam/corrigeExamBack/blob/deploy/.github/workflows/maven.yml

Why and where use a power profiler in the DevOps cycle?

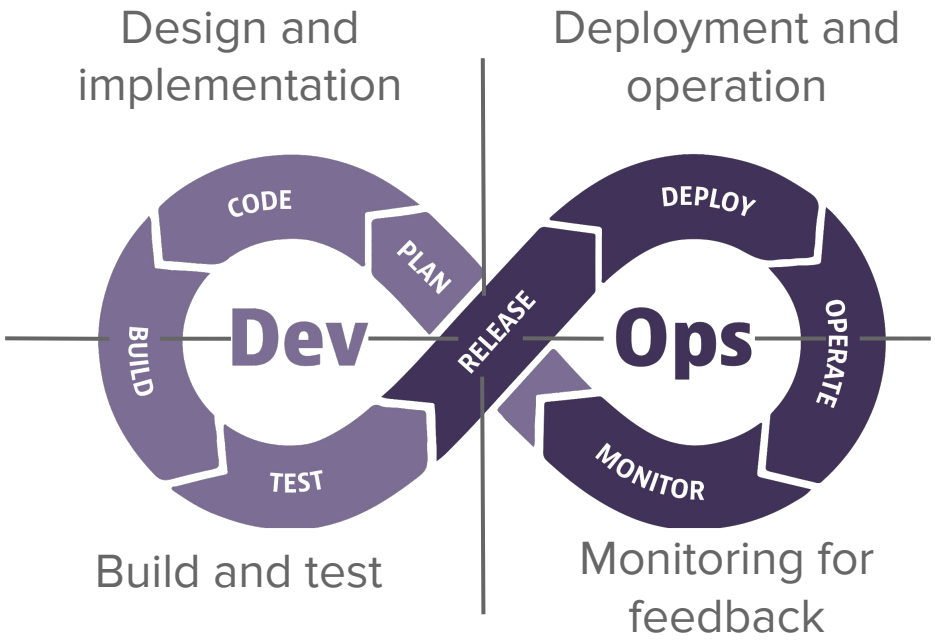
Why? ⇒ Monitoring allows us to identify **where** to focus our energy consumption efforts.



Why and where use a power profiler in the DevOps cycle?

Why? ⇒ Monitoring allows us to identify **where** to focus our energy consumption efforts.

For a given program used by millions of users, we prefer to optimize its code rather than the “build/test” stage, as the latter is used more sporadically by a smaller number of people compared to the number of users.



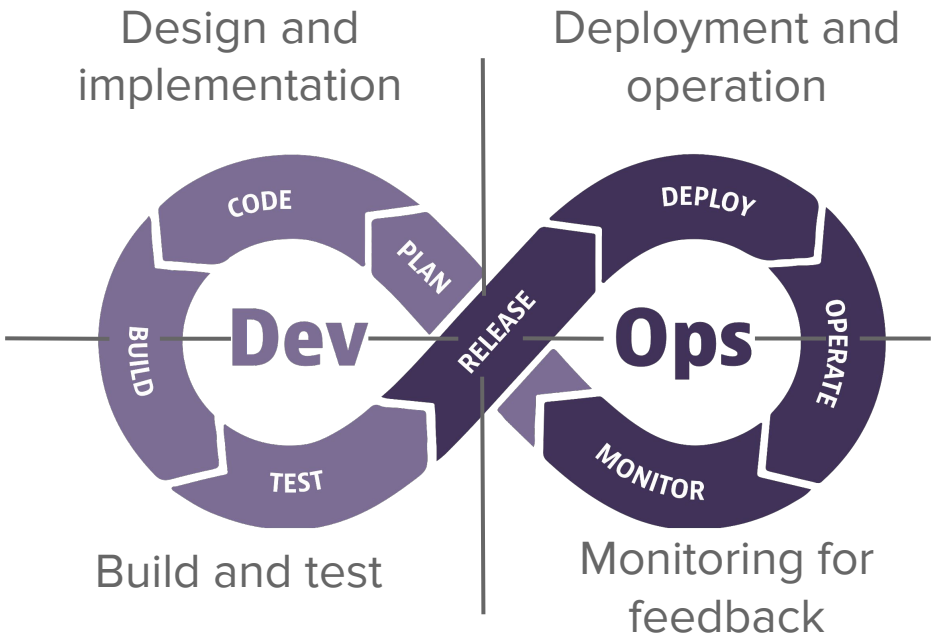
Why and where use a power profiler in the DevOps cycle?

Why? ⇒ Monitoring allows us to identify **where** to focus our energy consumption efforts.

⇒ **The best energy is that which is not spent!**

Whenever possible, **energy consumption** should be considered at the **design time**

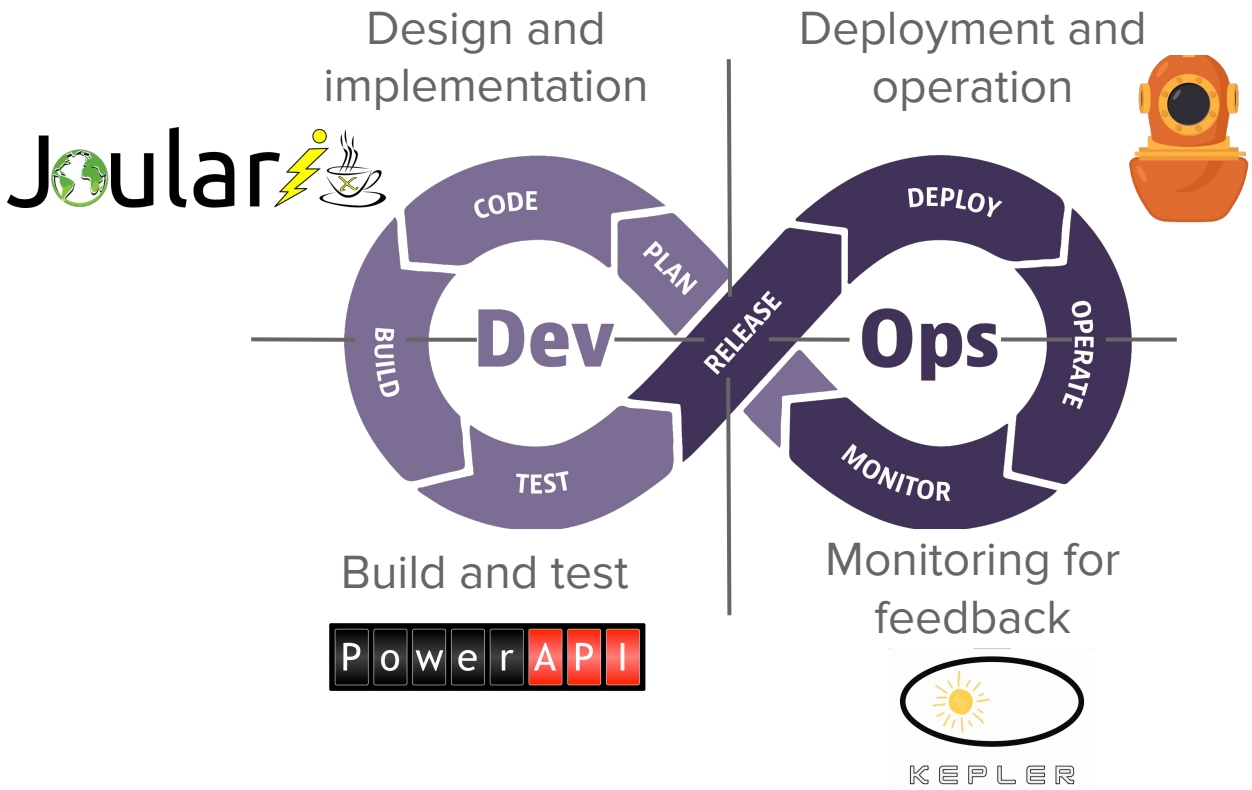
⇒ **requirements!**



Why and where use a power profiler in the DevOps cycle?

The **where** depends on the energy spend in the different phase of the lifecycle.

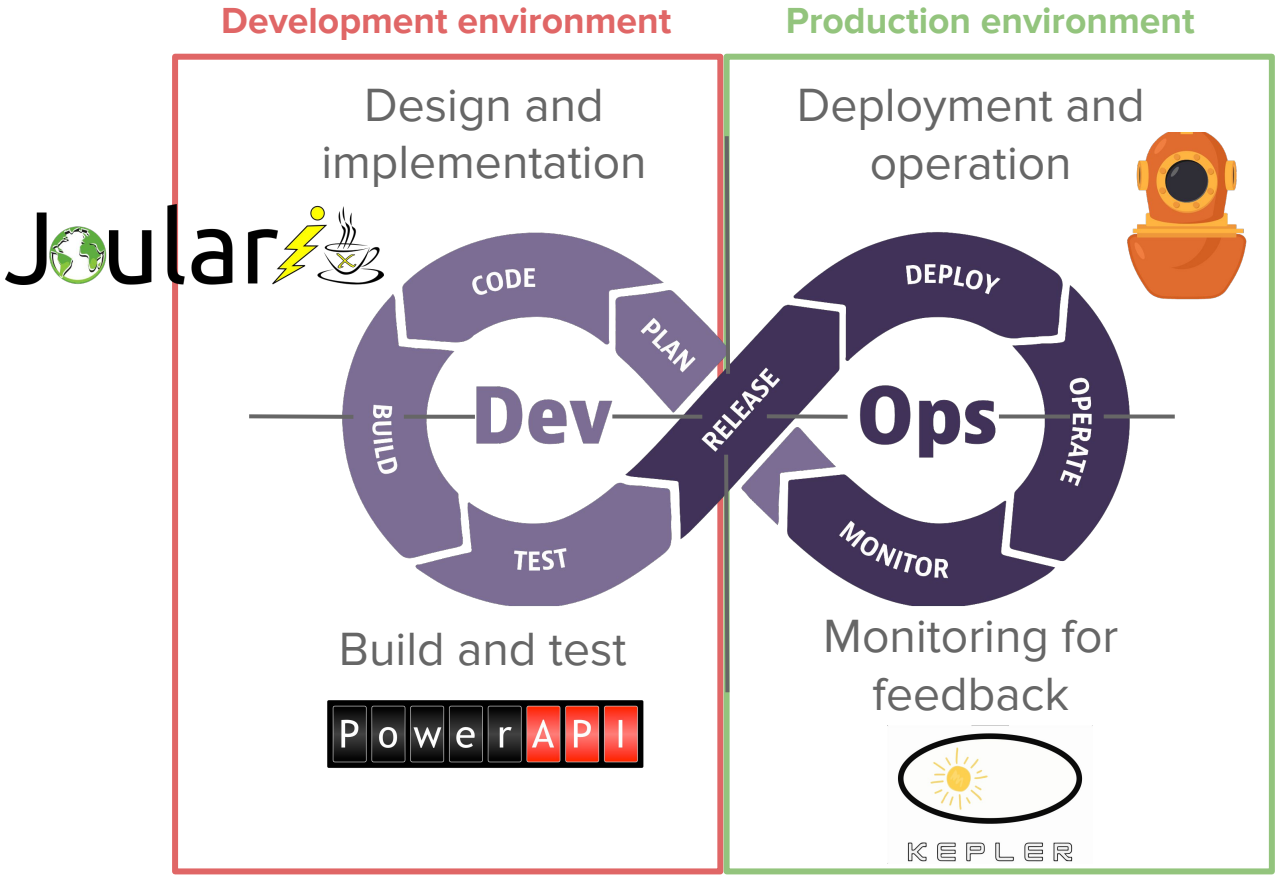
There is a solution of energy monitoring for each phase



Why and where use a power profiler in the DevOps cycle?

The **where** depends on the energy spend in the different phase of the lifecycle.

There is a solution of energy monitoring for each phase



Comparison of Power Profiling Software

	JoularJX	Scaphandre	PowerAPI	Kepler
Lifecycle	Design and implementation	Monitoring Continuous Delivery	Build and test	Monitoring
Ease of use	Fair	Good	Poor	Good
Quality of documentation	Fair	Good	Good	Good
Default sampling frequency	None - Code level power consumption	0.1 Hz	1 Hz	0.5 Hz
Virtualization support	False	True (k8s, Docker, VMs based on QEMU)	True (Docker)	True (k8s)

M. Jay, V. Ostapenco, L. Lefevre, D. Trystram, A. -C. Orgerie and B. Fichel, "An experimental comparison of software-based power meters: focus on CPU and GPU," *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, Bangalore, India, 2023, pp. 106-118, doi: 10.1109/CCGrid57682.2023.00020.

Take Away

- Software programs are particular products due to their immateriality
- Agile practices, and consequently DevOps, are widely adopted.
- There is a need for quantifying and monitoring before optimizing.
- Variety of monitoring tools with different levels of granularity and sampling rates is adapted to the different phases.