Green Coding Workshop

Green Coding Berlin

</> GREEN CODING;

Who we are

https://www.green-coding.berlin

Agenda

24.11.2023

- **PART 1 -**
  - Hello [5 Mins]
  - Requirements / Systems we use [10 Mins]
    - Note: We offer breakouts for professional practitioners! (Talk to Didi and Dan :))
  - Basics [15 Mins]
    - Where do software emissions come from
    - What is energy / power / CO2
  - How to get to power metrics in a system [10 Mins]
  - Easy tools to get first energy readings (perf_events, scaphandre, XGBoost ML, SDIA Model etc.) [30 min]
  - Cloud [30 Minutes]
- **PART 2**
  - Quick presentation GMT - Eco-CI - PowerHOG
    - Hands-On in Groups

</> GREEN CODING;

Requirements

For the workshop

- You must have a Github account.
- Linux System? -> Live Systems?
- Windows Systems? -> WSL2
- macOS Systems?
- Basic Install for everybody:
  - sudo apt update
  - sudo apt install curl git stress-ng -y
  - # or
  - brew install stress-ng
  - brew install curl

GREEN CODING;

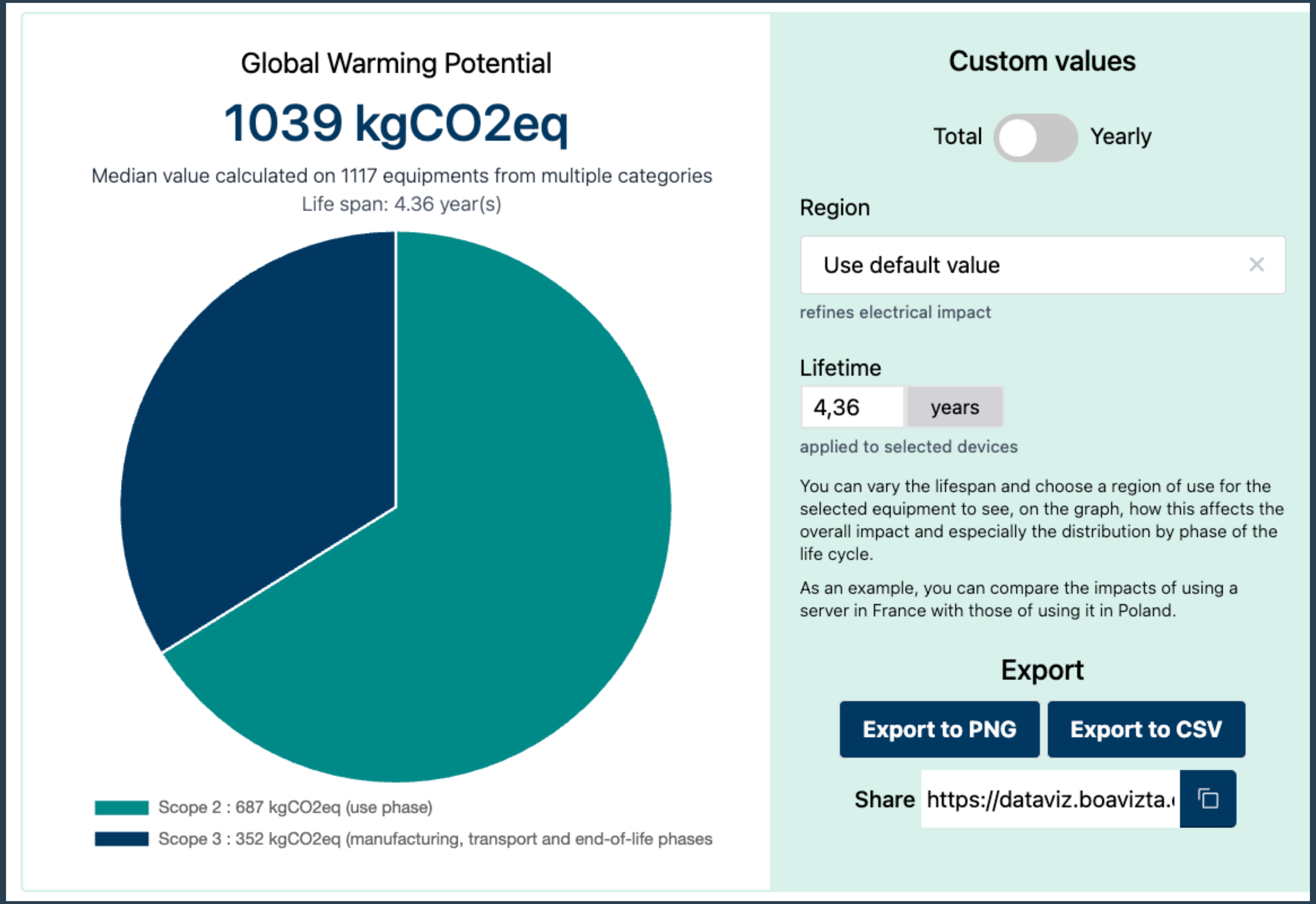
Where do software emissions come from?

Components

- Operational emissions - We will do this hands on!
  - Energy (and thus CO_2 through fossil fuels)
- Embodied Emissions - Done through data sheets
  - Carbon (workshop focus)
  - Water consumption
  - Land use
  - Toxic Metals
  - ...

Embodied Carbon
Using Life-Cycle-Assessment databases

- **Boavizta**
  https://dataviz.boavizta.org/manufacturerdata
- **Microsoft**
  https://tco.exploresurface.com/sustainability/calculator
- **Dell**
  Example: https://www.delltechnologies.com/asset/en-us/products/servers/technical-support/Full_LCA_Dell_R740.pdf
- ... many more



Source: https://dataviz.boavizta.org/manufacturerdata

Quick recap on energy and CO2

Technical details - What you must know for this workshop

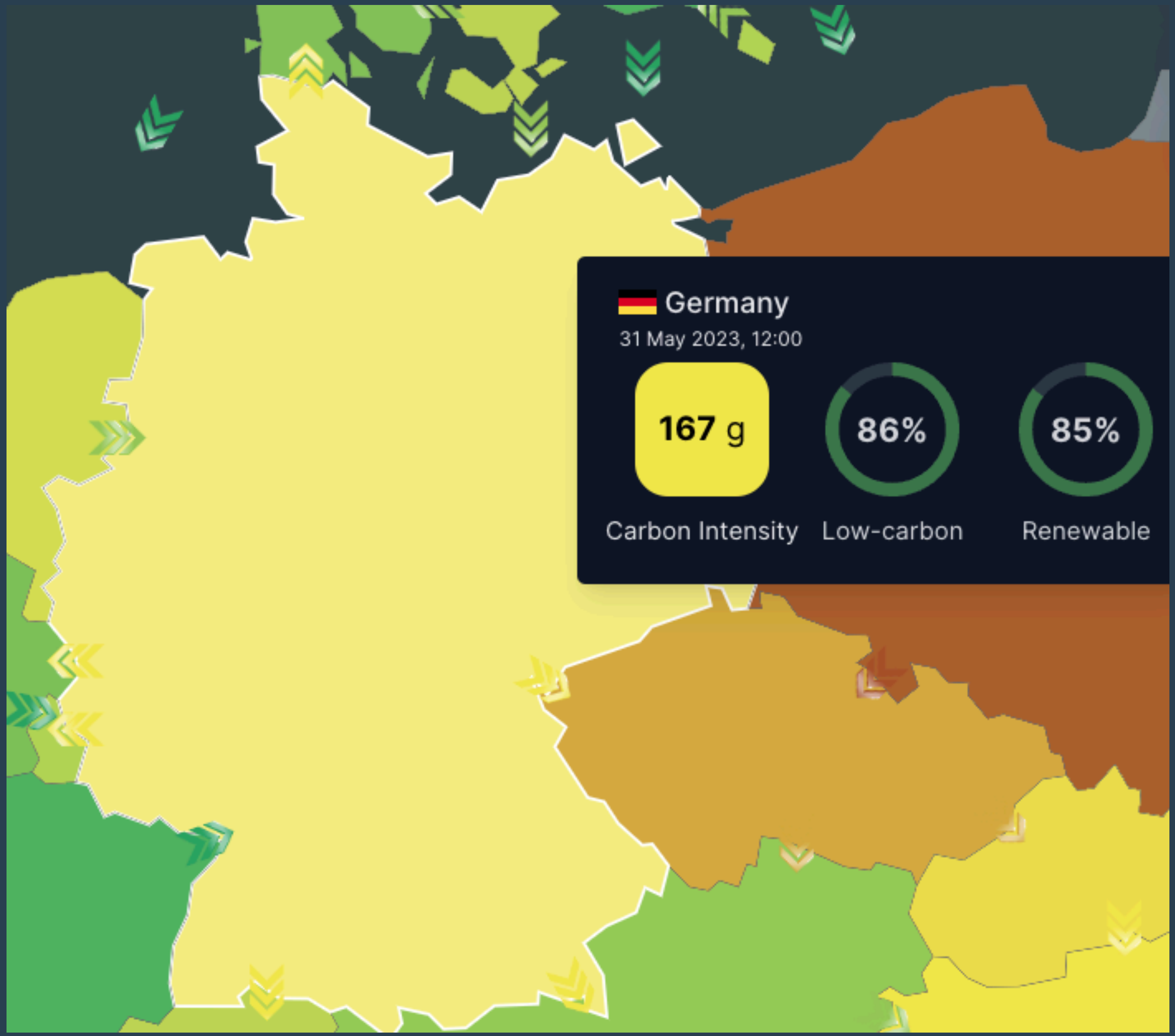
- What is a TDP?
  - https://ark.intel.com/content/www/us/en/ark/products/96900/intel-xeon-processor-e7-8894-v4-60m-cache-2-40-ghz.html
- What is a kWH?
  - Watts * usage time
- From TDP to kwH
  - https://www.green-coding.berlin/co2-formulas/#from-specs-to-kwh
- What is a Joule?
  - https://www.green-coding.berlin/co2-formulas/#from-joules-to-kwh
- From kwH to CO2e / Grid Carbon Intensity
  - https://app.electricitymaps.com/map
- From Network to CO2e
  - https://www.green-coding.berlin/co2-formulas/#gigabytes-to-kwh

</> GREEN CODING;

Getting from energy to CO2

Using grid emission factors

- **Electricitymaps**
  https://www.electricitymaps.com/
- **Bundesnetzagentur**
  https://www.smard.de/home
- **Wattime**
  https://www.watttime.org/
- **Carbon-Aware-SDK**
  https://github.com/Green-Software-Foundation/carbon-aware-sdk
- ... many more

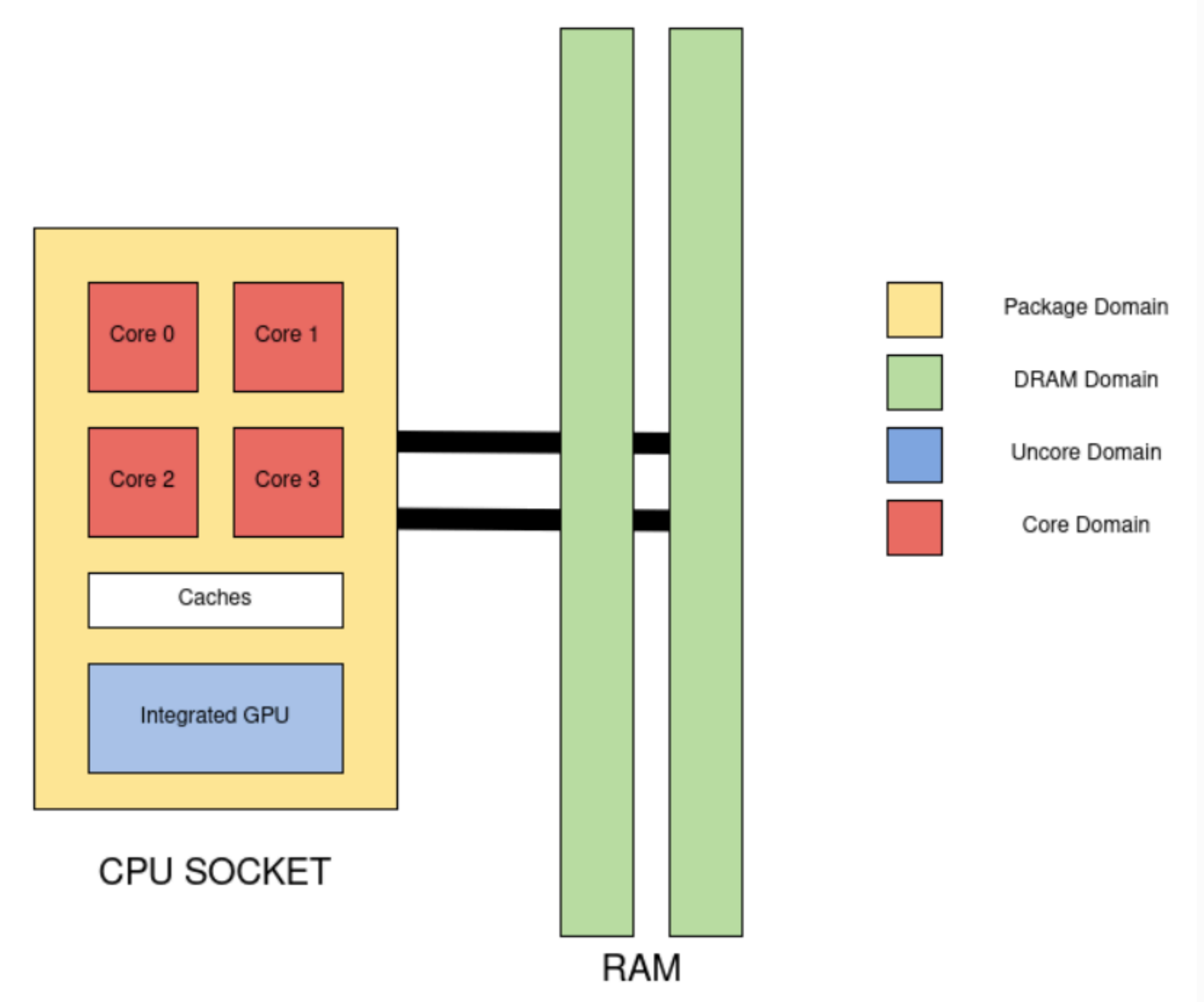


Source: https://app.electricitymaps.com/zone/DE

GREEN CODING;

How do we measure energy?

Two easy methods: Wall-Plug vs. Hardware/Software-Interfaces. Servers: IPMI



Intel RAPL
Source: https://pyjoules.readthedocs.io/en/stable/devices/intel_cpu.html

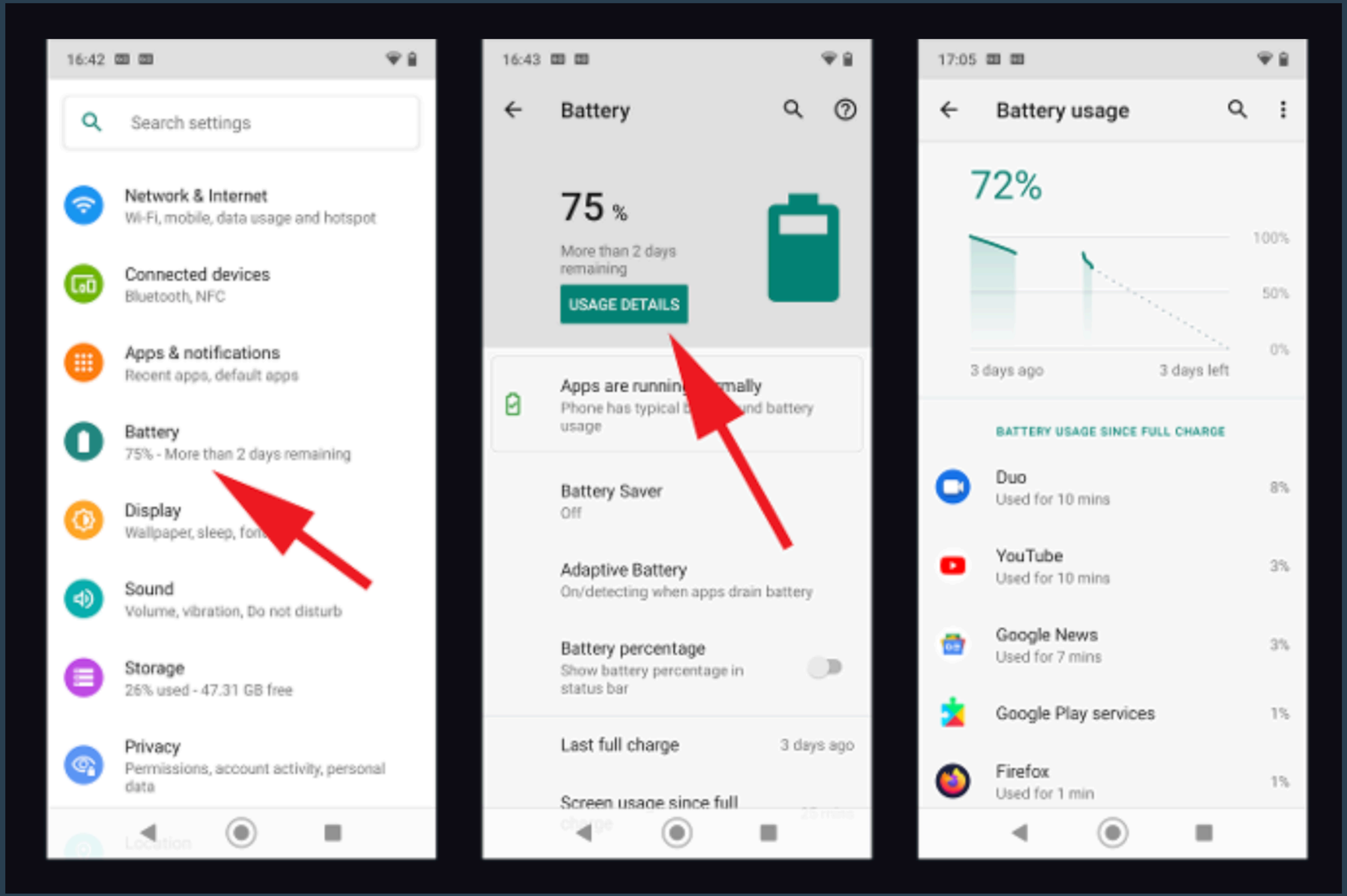
Wall-Plug power meter

GREEN CODING;

How do we measure energy?

A possible alternative: Through battery drain on mobile devices

Example: Coconut Battery for macOS / iOS



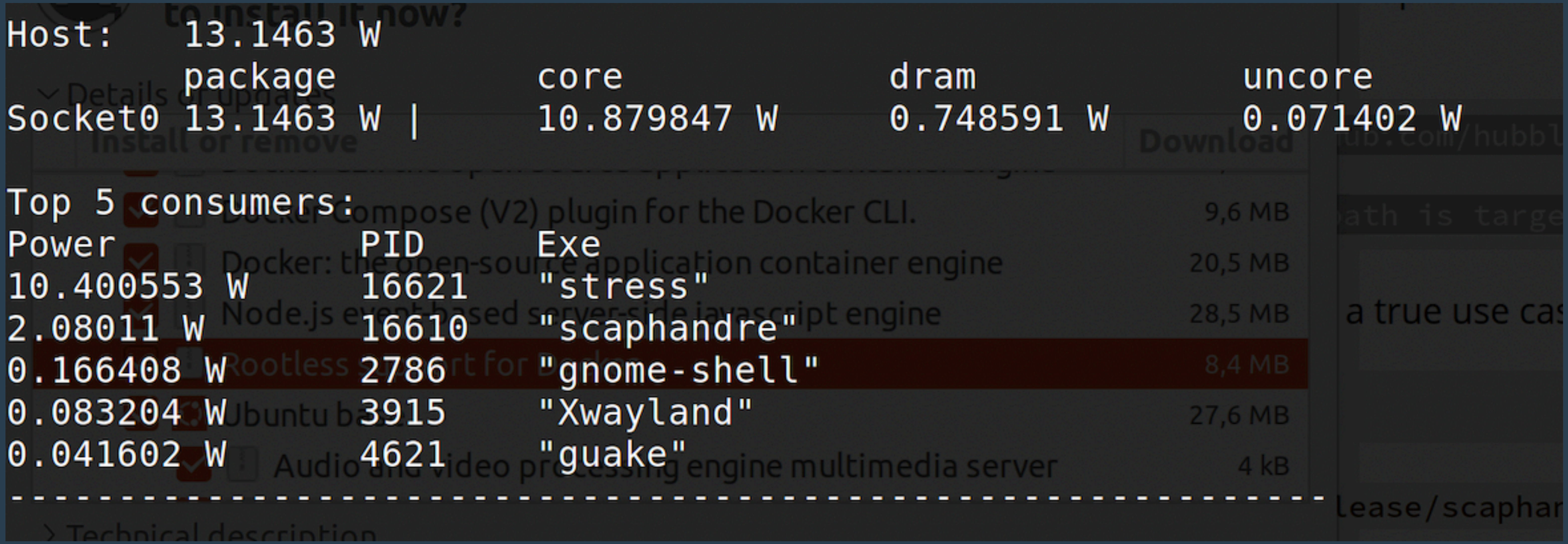
Android Battery usage (model)

GREEN CODING;

Scaphandre - Hubblo

open-source RAPL based command line tool

- Neat feature: Can split by process



GREEN CODING;

Let's run Scaphandre!

via https://hubblo-org.github.io/scaphandre-documentation/tutorials/compilation-linux.html

- ## Could not get it working with current version ...
- # from https://www.rust-lang.org/tools/install
- curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
- source "$HOME/.cargo/env"

- git clone https://github.com/hubblo-org/scaphandre.git
- cd scaphandre
- cargo build # binary path is target/debug/scaphandre
- git checkout v0.5.0
- sudo ./target/debug/scaphandre run stdout -t 0

- # But if you have docker in rootless mode:
- sudo docker run -v /sys/class/powercap:/sys/class/powercap -v /proc:/proc -ti hubblo/scaphandre stdout -t 15

- # now we run stress to see changes in separate terminal
- stress-ng -c 1

perf_events

- sudo apt install linux-tools-$(uname -r)
- perf list | grep power # to see what we have available on the system
- perf stat -e power/energy-pkg/ # to read package

- ## Mini Benchmark
- perf stat -e power/energy-pkg/ sleep 10 # to get system baseline over ten seconds
- perf stat -e power/energy-pkg/ stress-ng -c 1 -t 10 # to get system baseline over ten seconds

- # Look at IPC
- perf stat -e instructions,cache-misses stress-ng -c 1 -t 1 # to get system baseline over ten seconds

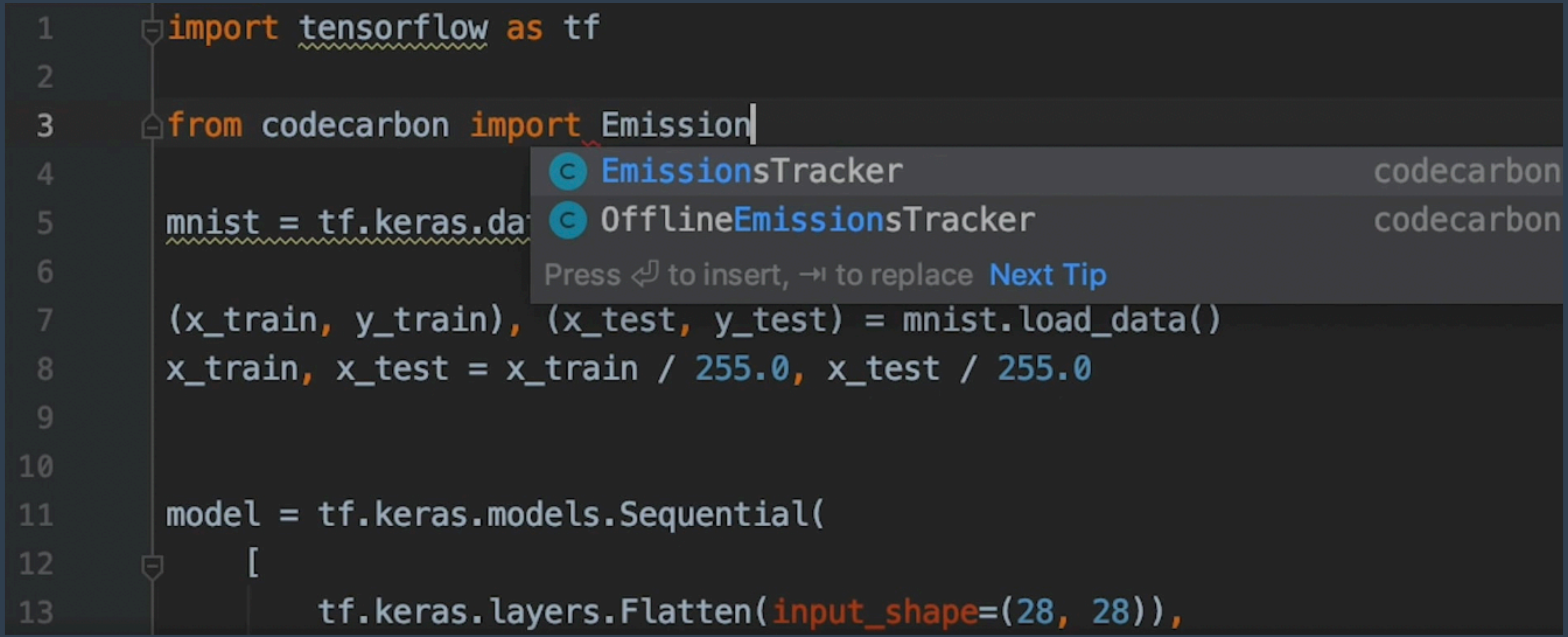
- # Look at "default" defailed view
- perf stat -d stress-ng -c 1 -t 1 # to get system baseline over ten seconds

GREEN CODING;

codecarbon.io

Slide only due to time constraints :)

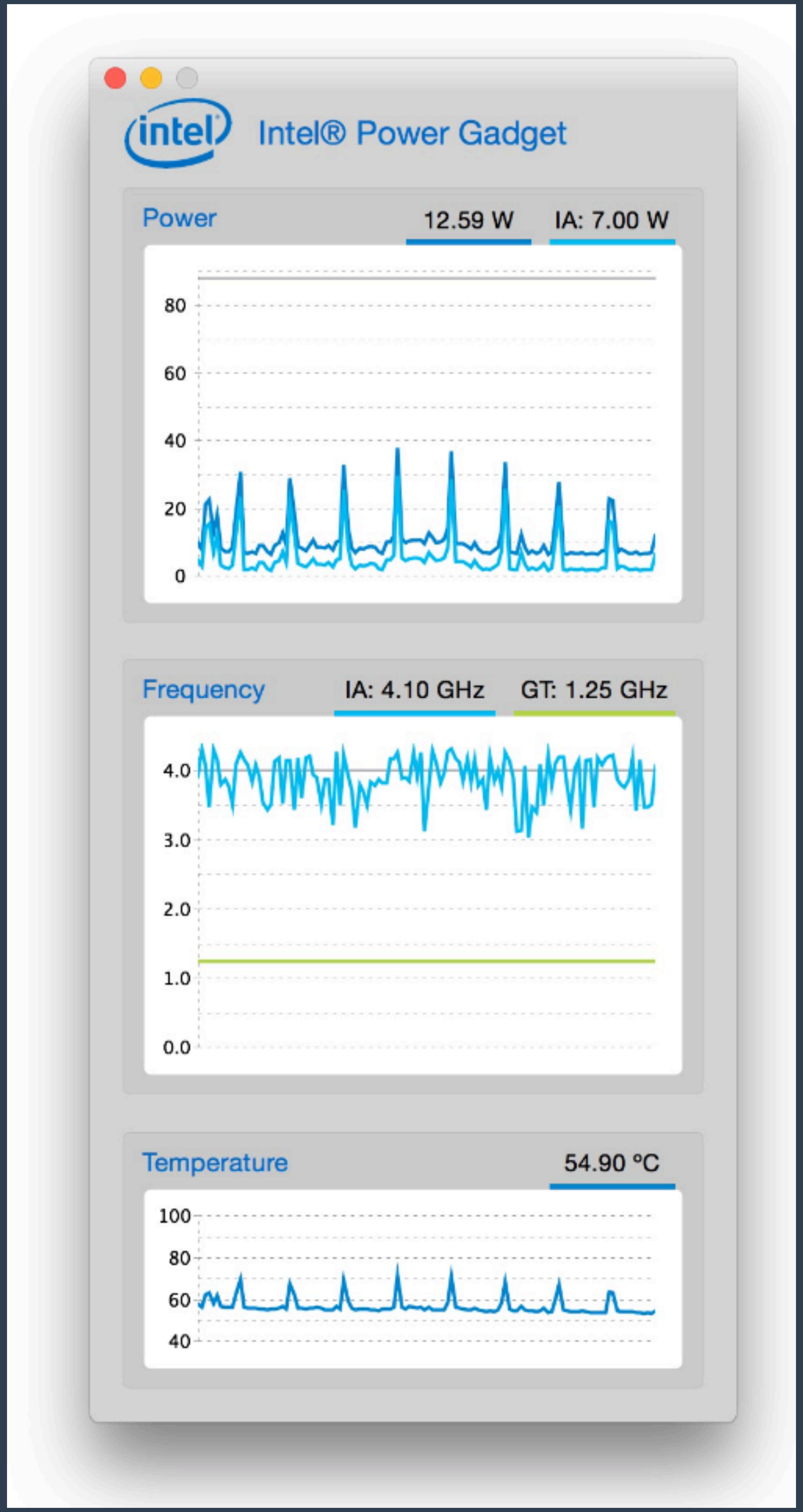
- Python
- RAPL-based
- NVIDIA GPU support



Windows Tools

Intel Power Gadget

- Not distributed anymore.
- But you can try: https://www.computerbild.de/download/Intel-Power-Gadget-24653156.html



GREEN CODING;

Cloud environments

SDIA Model - 1/3

- Assumes that 65% of the machines total energy is related to the CPU. We can extrapolate from there
  - **total_machine_power = ((cpu_utilization * TDP) /0.65) * CPU_CHIPS**

Cloud environments

SDIA Model - 2/3

- Assumes that 65% of the machines total energy is related to the CPU. We can extrapolate from there
  - **total_machine_power = ((cpu_utilization * TDP) / 0.65) * CPU_CHIPS**
- Example for 12% CPU Utilization and 2 chips with a 160 W TDP:
  - (0.12 * 160) / 0.65 * 2 = **59.08 W**

Cloud environments

SDIA Model 3/3

- That, more or less, is for instance what CloudCarbonFootprint does. However, they integrate more components like memory etc. with static offsets.
  - https://www.cloudcarbonfootprint.org/docs/methodology/#energy-estimate-watt-hours
- For hard disks they use 0.001 kWh/Gb for instance
- How can we improve that ...?

Cloud environments

XGBoost estimation 1/3

- Using ML Models based on power curves of actual machines
  - Non-Linear!
- https://www.spec.org/power_ssj2008/results/https://www.spec.org/power_ssj2008/results/

Cloud environments

XGBoost estimation 2/3

- Using ML Models based on power curves of actual machines
  - Non-Linear!
- Caveats:
  - CPU Frequency is needed to be assumed constant
- See our article on this in detail:
  - https://www.green-coding.berlin/case-studies/cpu-utilization-usefulness/
  - https://www.green-coding.berlin/case-studies/hyper-threading-and-energy/
  - etc.

Cloud environments

XGBoost estimation 3/3

- Let's install it!
  - https://github.com/green-coding-berlin/spec-power-model

</> GREEN CODING;

Appetizer for deep dives

How to get good measurements?

- Architectures (Mikrocontrollers vs. Multi-Tasking systems)
- Stable systems (Timers, Services, Processes)
- Temperature
- Component scalings (HyperThreading, Turbo Boost, PowerCaps)
- Calibration (Resource congestion / Headroom)
- Overhead
- ... (watch our blog :))

Part #2

In breakout groups

Green Metrics Tool Linux&WSL / Eco-CI / macOS

Green Metrics Tool - Cluster Setup

Current machines in the Green Metrics Tool Cluster

- Fujitsu ESPRIMO P956 - Blue Angel compatible (Ubuntu)
- Fujitsu TX1330 M2 - Single-Tenant Server (Ubuntu)
- Quanta Leopard - Multi-Tenant Server - SoftAWERE compatible (Ubuntu)
- Intel Mac 13" Q3-2015
- M1 Mac 13" Q1-2022