

Green Metrics Tool

measure and compare the energy-use of software

 **GREEN** CODING;

What?

What is the Green Metrics Tool

- Open Source tool to measure the energy / co2 consumption of software
 - embodied carbon
 - network (WIFI, fixed-line, mobile)
 - CPU energy
 - DRAM energy
 - Hard-Disk energy

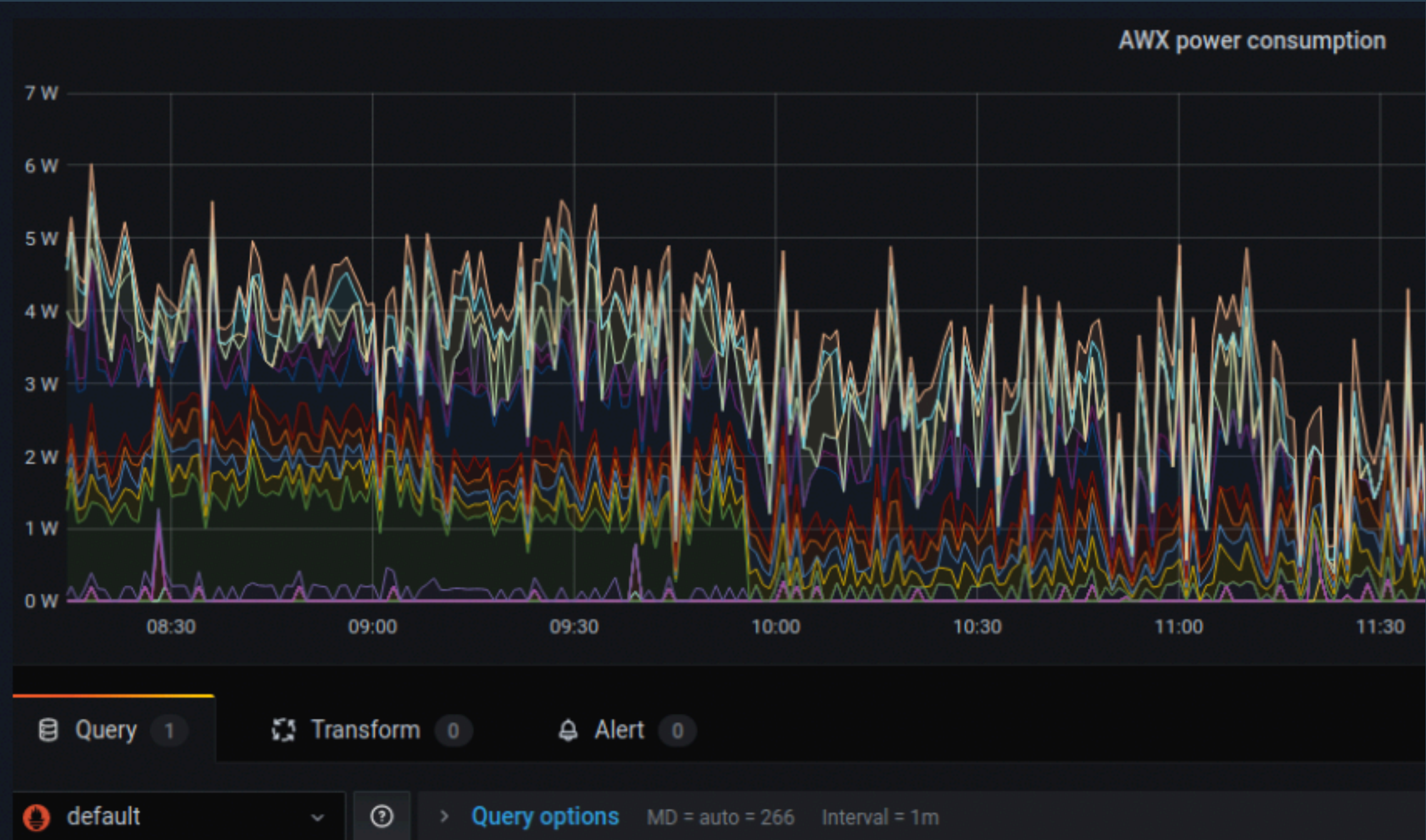
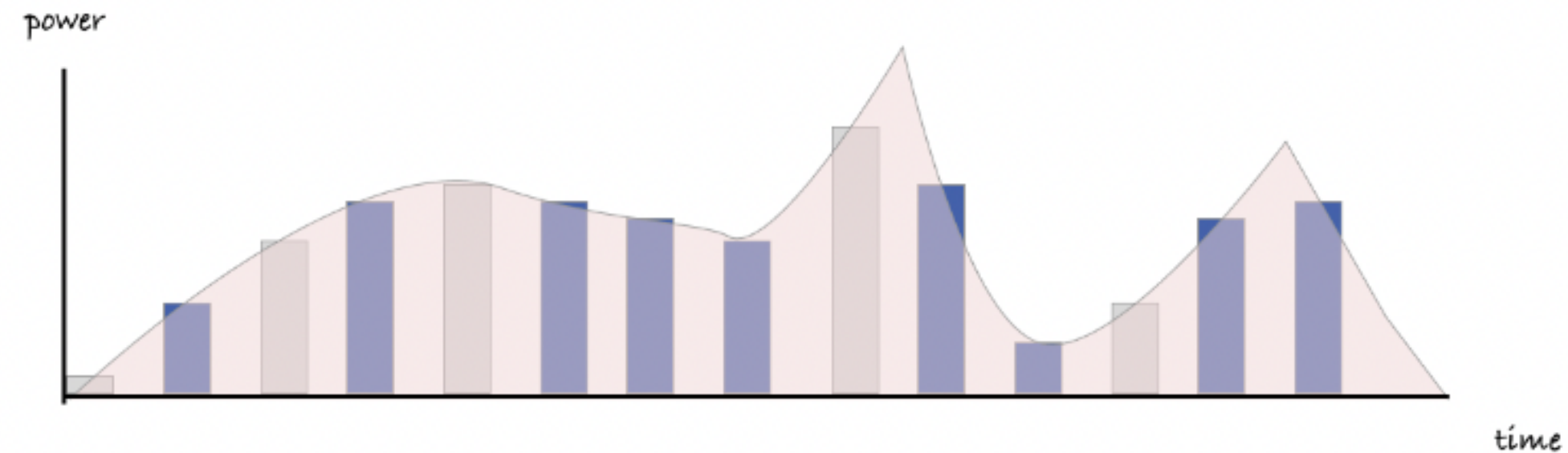
Why?

Aren't there tools already?

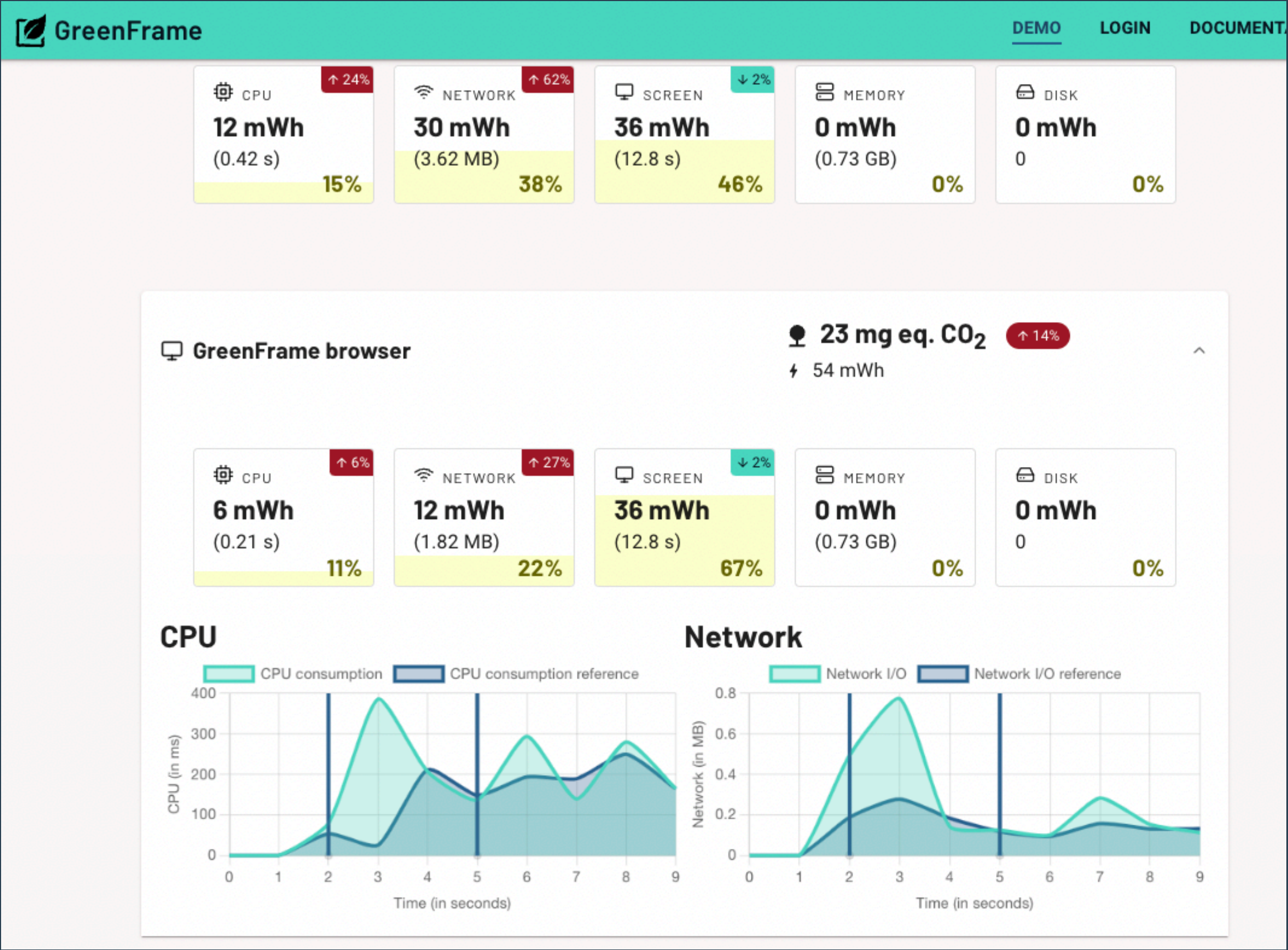
- Scaphandre
- Greenframe.io
- Blauer Engel für Software
- perf_events
- PAPI
- AMD uProf
- Intel Power Gadget
- Academic research ... lots of! But never with actual raw data ...
- more?

Scaphandre

```
Host: 9.391334 W      Core      Uncore      DRAM
Socket0 9.392 W      1.497082 W
Top 5 consumers:
Power  PID  Exe
4.808363 W  642  "/usr/sbin/dockerd"
4.808363 W  703  "/usr/bin/docker-containerd"
4.808363 W  1028 "/usr/local/bin/redis-server"
0 W  1  "/usr/lib/systemd/systemd"
0 W  2  ""
```



Greenframe .io



Comparison of current industry tools

What current tools are lacking

Scaphandre	<u>Greenframe.io</u>	Cloud Carbon Footprint
👍 Free and Open Source	👍 Comparison between iterations	👍 Free and Open Source
👍 CPU / DRAM real measurements	👍 Nice visual interface	👍 Estimate cloud infrastructures
👍 command line app support	👍 Network included	👍 Embodied Carbon
👍 Raw Data export to Prometheus	👍 Usage scenario for apps	👍 Nice visual interface
😞 No comparison between iterations	🚫 Methodology unknown	👍 Recommendations
😞 No visual interface	🚫 No real measurements	🚫 Granularity on "Service" level
🚫 No usage scenario for measurement	🚫 Not Open Source (outdated version on github)	🚫 No real measurements
🚫 No estimate for cloud infrastructure	🚫 Non-Free	🚫 No Network Energy
🚫 No Network Energy	🚫 No Raw Data export	🚫 No comparison between iterations
	🚫 No estimate for cloud infrastructure	🚫 Only apps on cloud infrastructure

Why? Why? (aka Why #2)

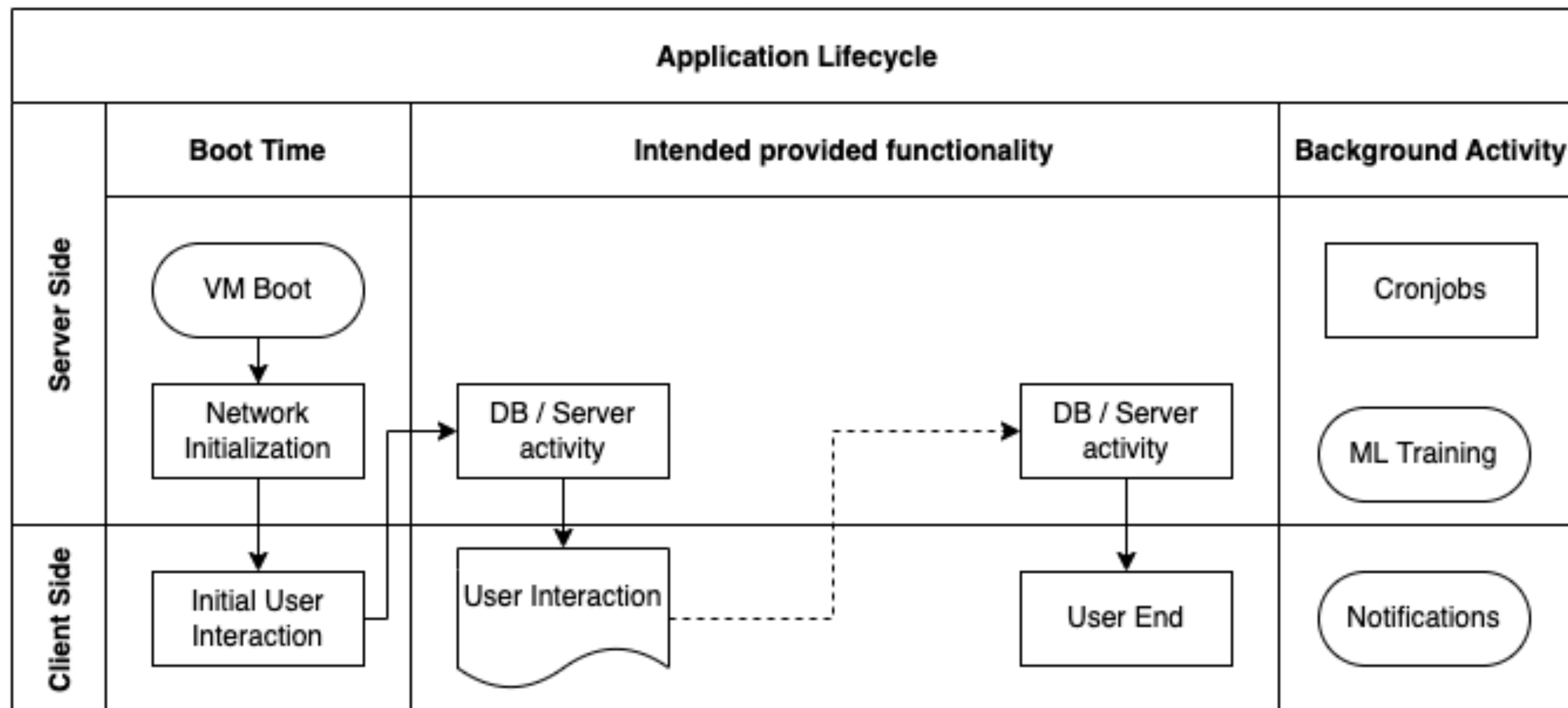
Why did we build the Green Metrics Tool?

- Community and open data approach for measurements
- Carbon awareness approach for software
- Compare measurements of different software
- Real energy measurements DC / AC
- Application lifecycle
 - Boot times
 - Pre-standardized standard usage scenarios

We want such a tool to exist
and create a green software eco-system

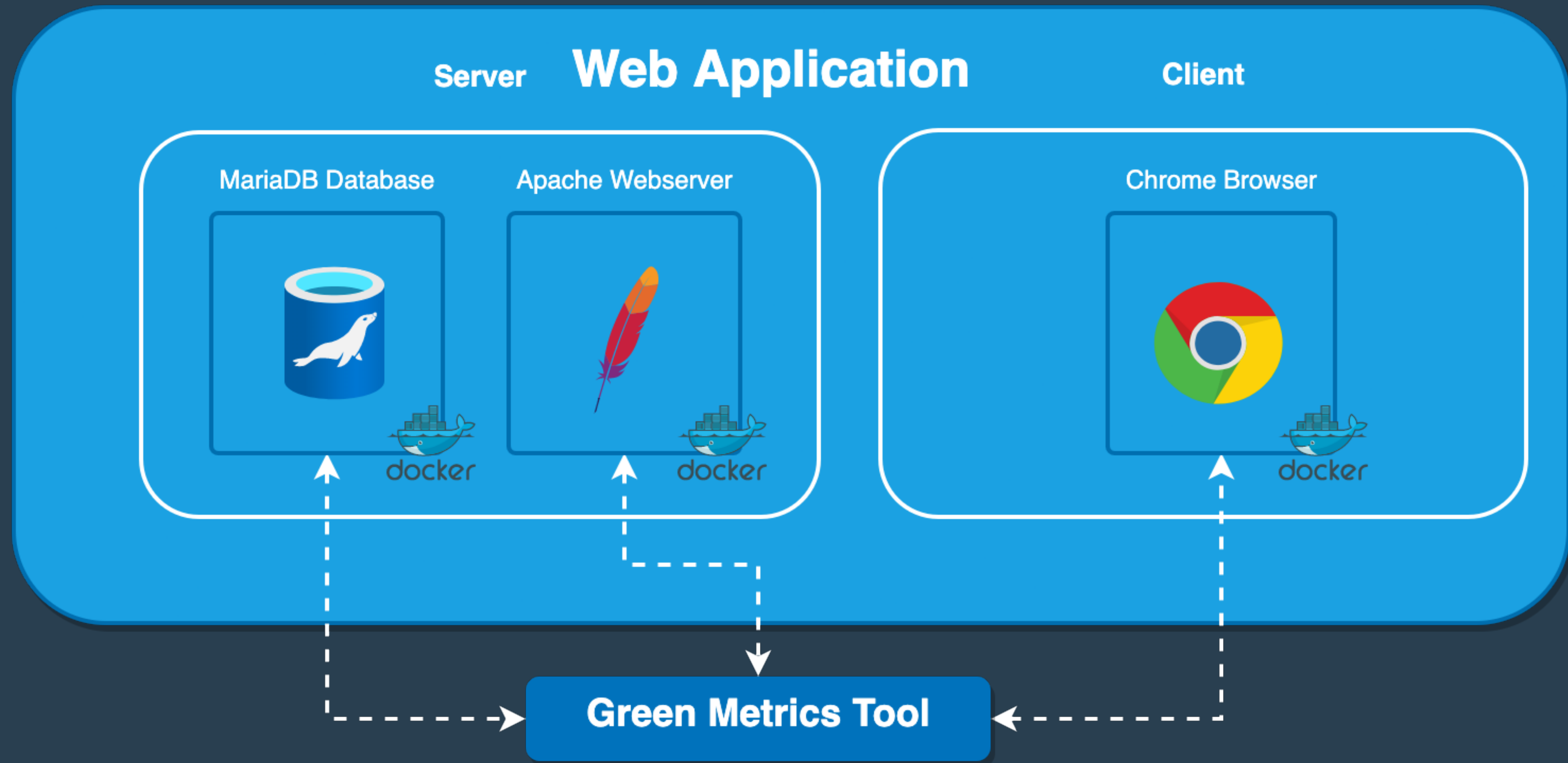
Concepts of the Green Metrics Tool

Application Lifecycle & Standard Usage Scenarios



Concepts of the Green Metrics Tool

Adoption of container approach. Every functionality is a container



Demo Time

Understanding Tools without visuals is hard!

Demo Time

Agenda

- Documentation
 - Metric Reporters. Can used separately. Everyone is like little Scaphandre
- Case Studies
- CO2 formulas
- Example apps
- Web interface and Open Data API

Provide developers with answers

Even around small decisions

- Energy efficiency of Wordpress vs. static site?
- Is podman more effective than docker for building?
- Is Flask better than FastAPI than Django for our workflow?
- Hardest goal to achieve: recommendations
 - When to make energy optimizations
 - What kind of saving do I want to have? 1 g of co2? 1 ton of co2? What is my budget?
 - When should I switch architectures?

Other approaches of Green Coding Berlin

- Green Software eco-system with awareness around digital energy consumption
- Case studies
- Integration into CI Pipelines (Badges)
- Inline Measurements in current stacks
- Show energy differences between architectures
- Create Transfer Models to estimate energy of target infrastructure with measuring
- Integration into OSS frameworks like RedwoodJS, Django, RoR etc.

Want deeper dives and more details?

Follow Green-Coding.org

- Check out our website and blog & newsletter: <https://www.green-coding.org>
- Meetup group: <https://www.meetup.com/green-coding>
- Demo Open Data Repository: <https://metrics.green-coding.org>
- Our tool: <https://github.com/green-coding-berlin/green-metrics-tool>
- <https://www.linkedin.com/in/arne-tarara> / arne@green-coding.org
- If you wanna present your green software case, please hit us up!