university of groningen

faculty of science and engineering

digital lab

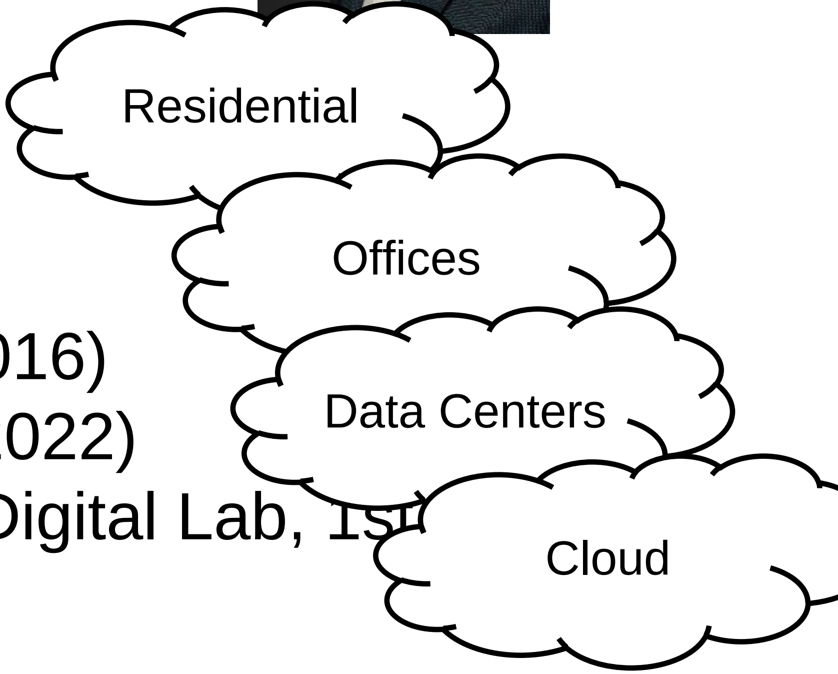
Sustainability at Scale

dr. Brian Setz (b.setz@rug.nl)
University of Groningen, Digital Lab
SummerSOC 2025

Who am I?

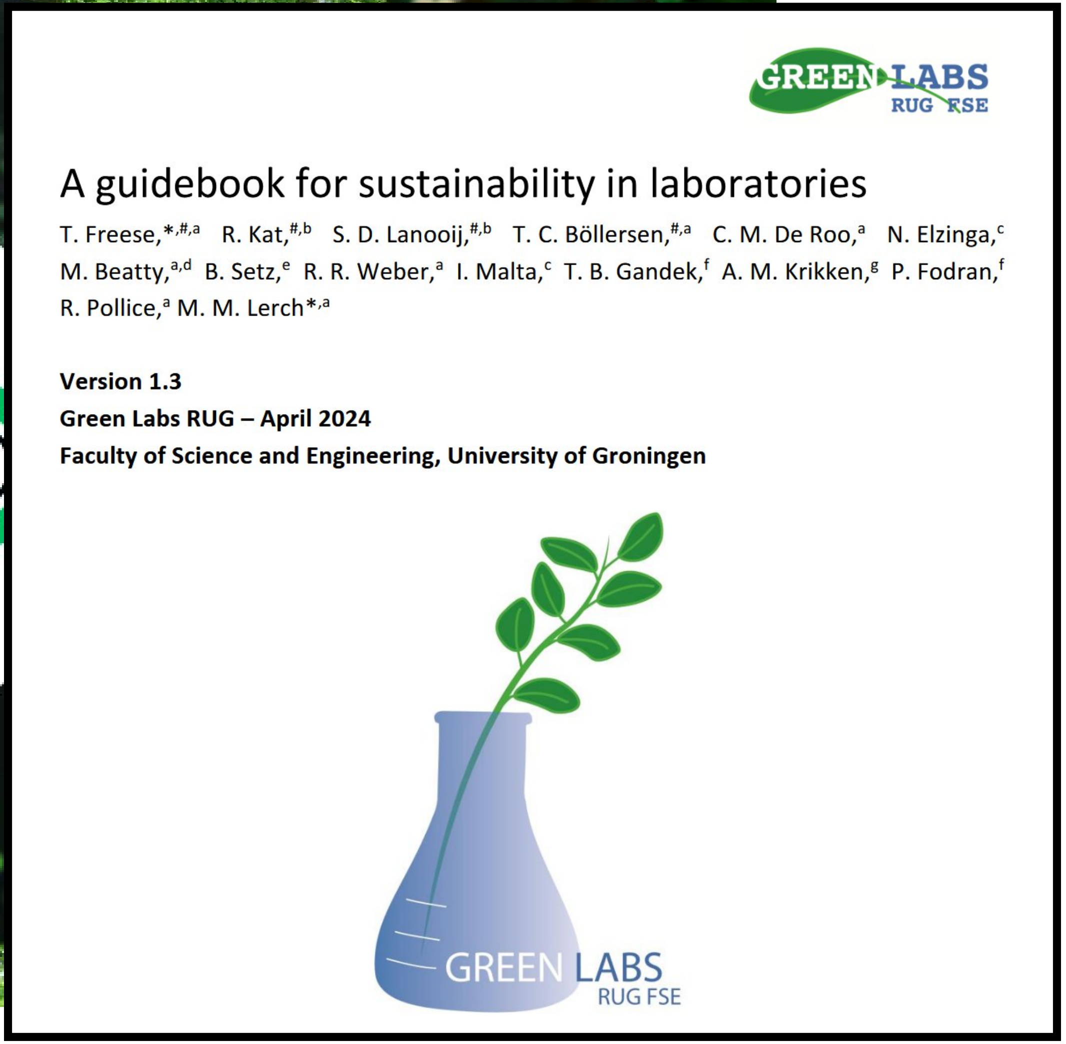
- Brian Setz, Head of the Digital Lab @ UG
  - PhD in Computer Science, Universität Stuttgart

- Sustainability
  - HERTZ & eDIANA, energy efficient buildings (2011)
  - UG GreenMind Awards (2012, 2014)
  - SURFsara Sustainability & ICT grant (2015)
  - Founding shareholder Sustainable Buildings B.V. (2016)
  - NWO NextGenSmartDC, smart data centers (2016-2022)
  - LEAF (2024) + GreenDiSC (2025) Accreditation for Digital Lab, 1s UG



Green DiSC, **D**igital **S**ustainability **C**ertification (2025)

Awareness



A guidebook for sustainability in laboratories

T. Freese,[*,#,a] R. Kat,[#,b] S. D. Lanooij,[#,b] T. C. Böllersen,[#,a] C. M. De Roo,[a] N. Elzinga,[c] M. Beatty,[a,d] B. Setz,[e] R. R. Weber,[a] I. Malta,[c] T. B. Gandek,[f] A. M. Krikken,[g] P. Fodran,[f] R. Pollice,[a] M. M. Lerch*,[a]

Version 1.3

Green Labs RUG – April 2024

Faculty of Science and Engineering, University of Groningen

A Quick Recap

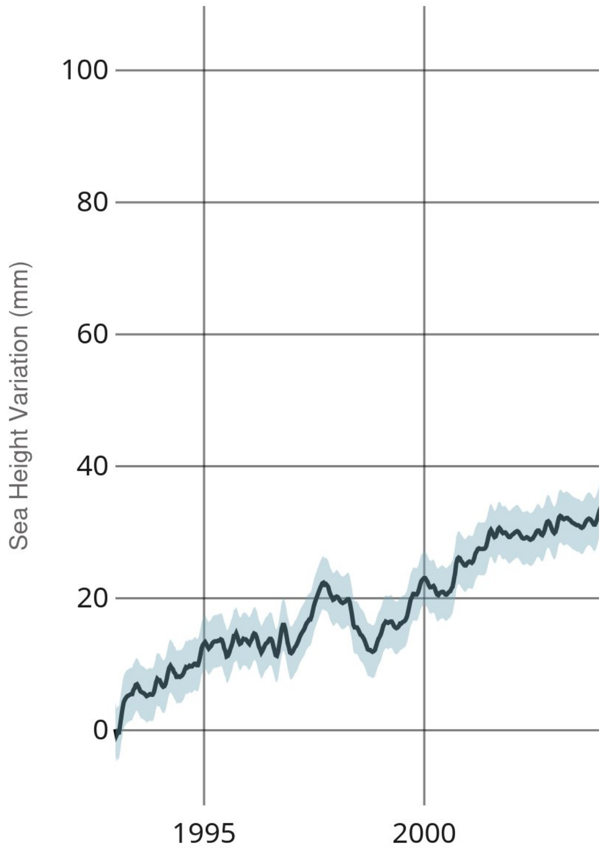
SUSTAINABILITY

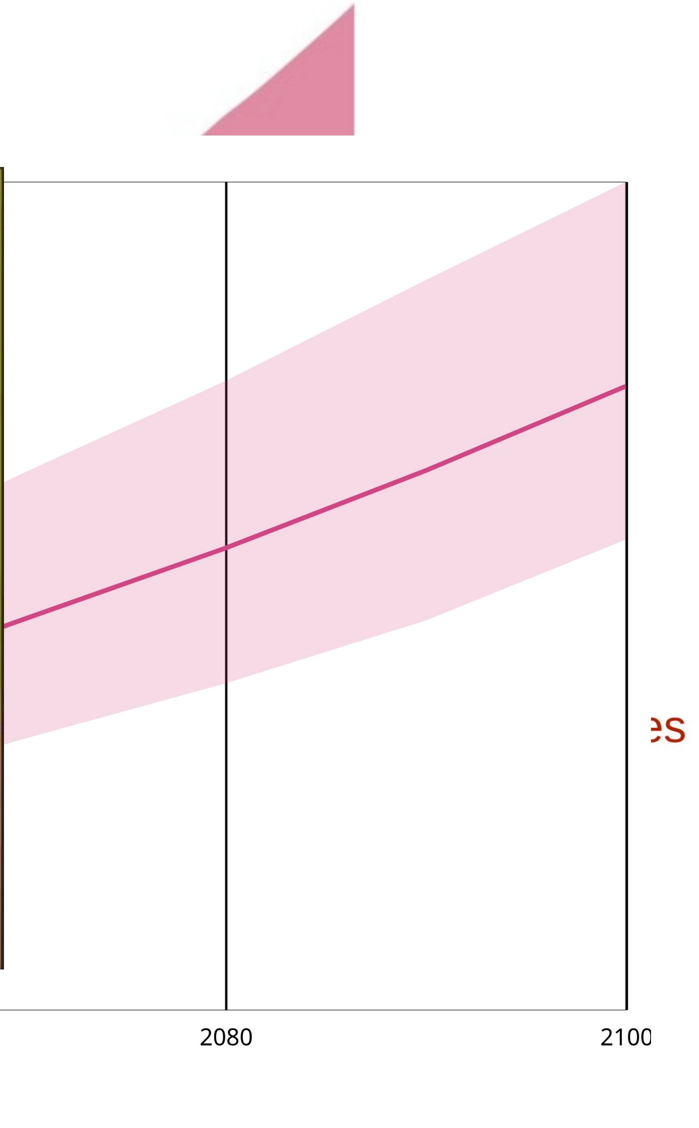
Paris Agreement

› In 2015, ~200 nations agreed to limit global warming to 1.5degC by 2030

The three degrees world

› On track for a 3degC increase

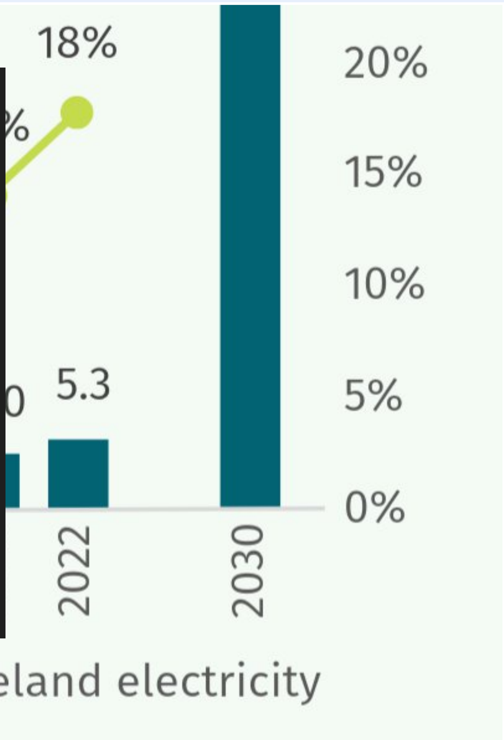




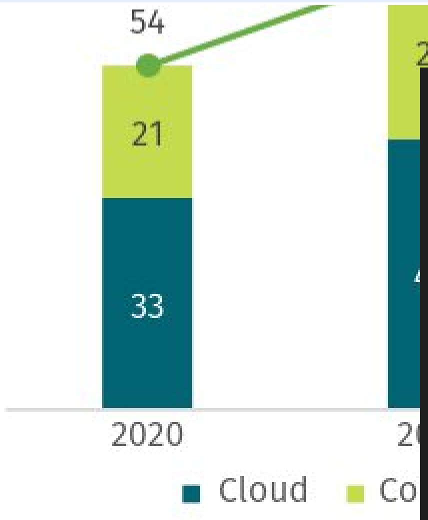
Source: climate.nasa.gov

What about IT? Are we doing better?

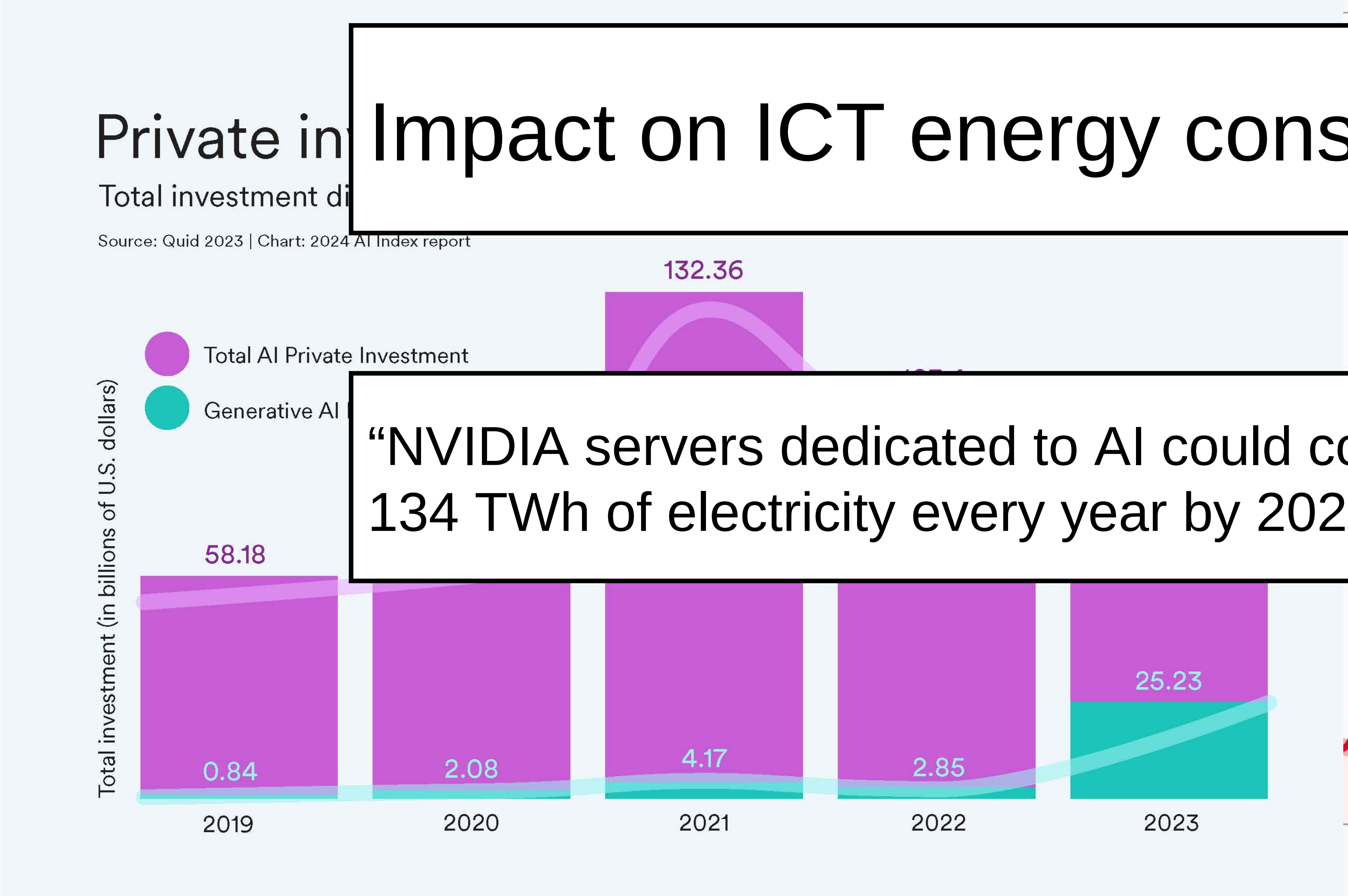
› World Bank Report (2024)

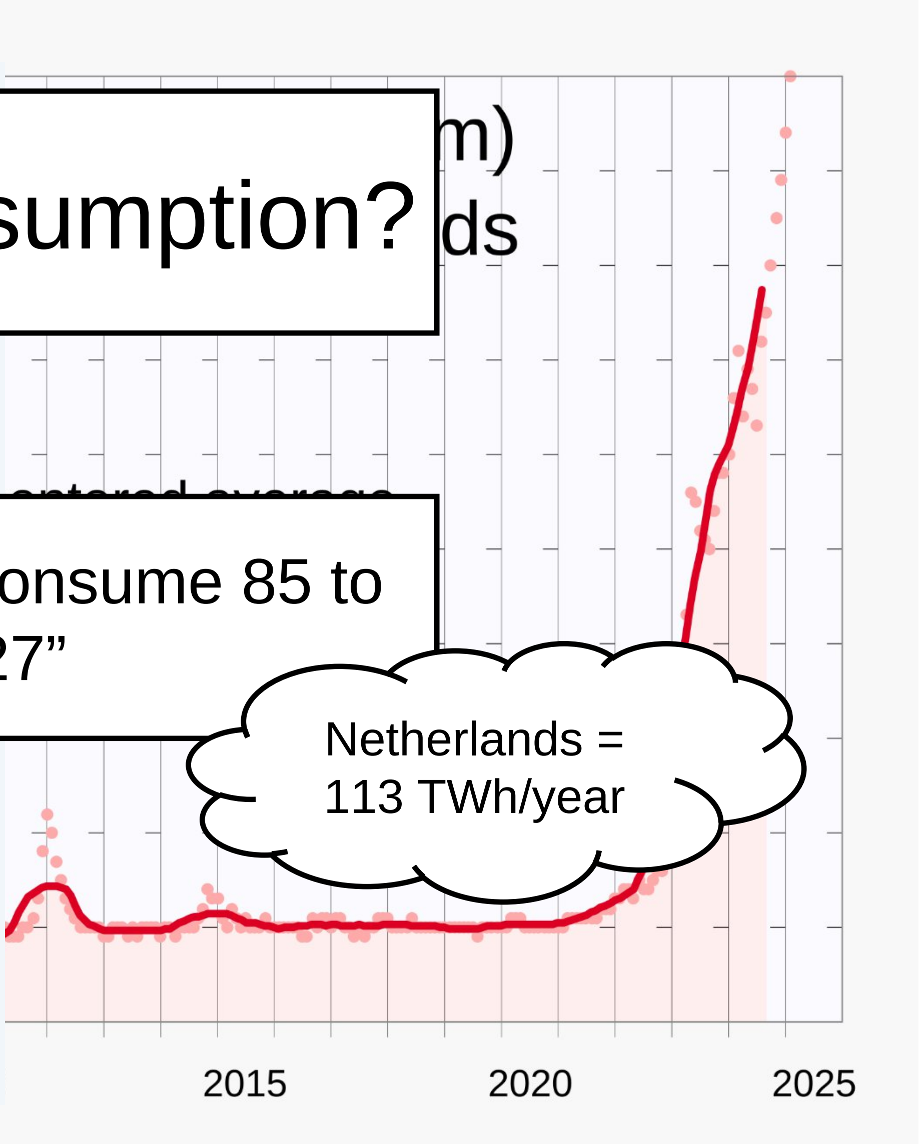


	2020	2021	2022
World Energy Con.	1094 TWh	1196 TWh	1183 TWh **(+8.2%)**
World Emissions	568 mil tCO$_2$e	589 mil tCO$_2$e	567 mil tCO$_2$e **(-0.4%)**



GenAI “Boom” of 2023





Impact on ICT energy consumption?

“NVIDIA servers dedicated to AI could consume 85 to 134 TWh of electricity every year by 2027”

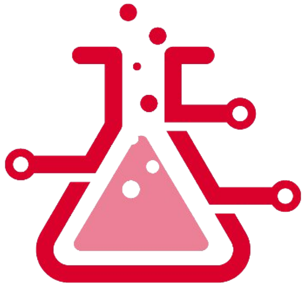
Netherlands = 113 TWh/year

Supporting Education in Digital World

DIGITAL LAB

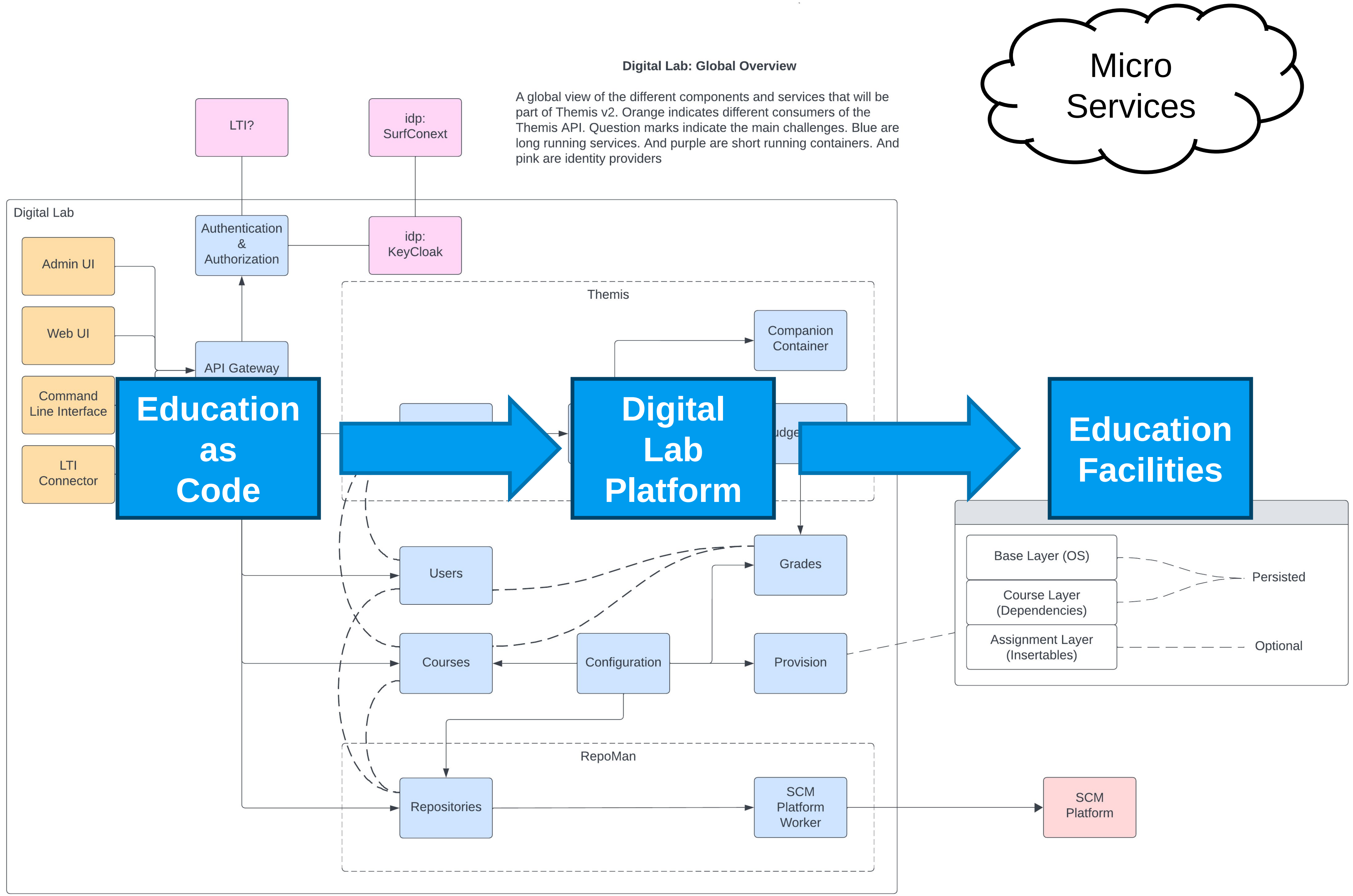
Who are we?

- Innovation by modernizing the university’s curriculum through the **design and development of a digital tool suite** that supports *education* activities and enables scalability of courses through automation.



- From **19** supported courses in 2023 to **55** courses in 2025

- Services:
  - **Themis**, automated assessment of programming assignments
  - **Repository Management**, organization of student source code repositories
  - **Virtual Labs**, compute resources for students


Micro Services
Digital Lab: Global Overview
A global view of the different components and services that will be part of Themis v2. Orange indicates different consumers of the Themis API. Question marks indicate the main challenges. Blue are long running services. And purple are short running containers. And pink are identity providers
LTI?
idp: SurfConext
Digital Lab
Authentication & Authorization
idp: KeyCloak
Admin UI
Web UI
Command Line Interface
LTI Connector
API Gateway
Themis
Companion Container
Education as Code
Digital Lab Platform
Education Facilities
Users
Grades
Courses
Configuration
Provision
Base Layer (OS)
Course Layer (Dependencies)
Assignment Layer (Insertables)
Persisted
Optional
RepoMan
Repositories
SCM Platform Worker
SCM Platform

Themis

Submit

You can upload mu

Submission: Comparing grades /

Practical 2		
Comparing grades	Matter and antimatter	Rules of Acquisition
Passed: 50.00	Wrong output: 20.00	Passed: 35.00
Timeout: 18.00	Runtime error: 20.00	Wrong output: 4.00
Timeout: 10.00	None	Wrong output: 4.00
Passed: 50.00	Passed: 25.00	Passed: 35.00
Missed deadline: 50.00	None	None
Passed: 50.00	Passed: 25.00	Passed: 35.00
Passed: 50.00	Wrong output: 22.00	Passed: 35.00
Passed: 50.00	Wrong output: 20.00	Passed: 35.00
Wrong output: 2.00	Wrong output: 22.00	Passed: 35.00
Passed: 50.00	Runtime error: 23.00	Wrong output:
Passed: 50.00	Wrong output: 11.00	Passed: 35.00
Timeout: 10.00	Runtime error: 8.00	Passed: 35.00
Passed: 50.00	Runtime error: 12.00	Passed: 35.00
None	Runtime error: 11.00	None
Passed: 50.00	Runtime error: 23.00	Passed: 35.00

Judgement #0

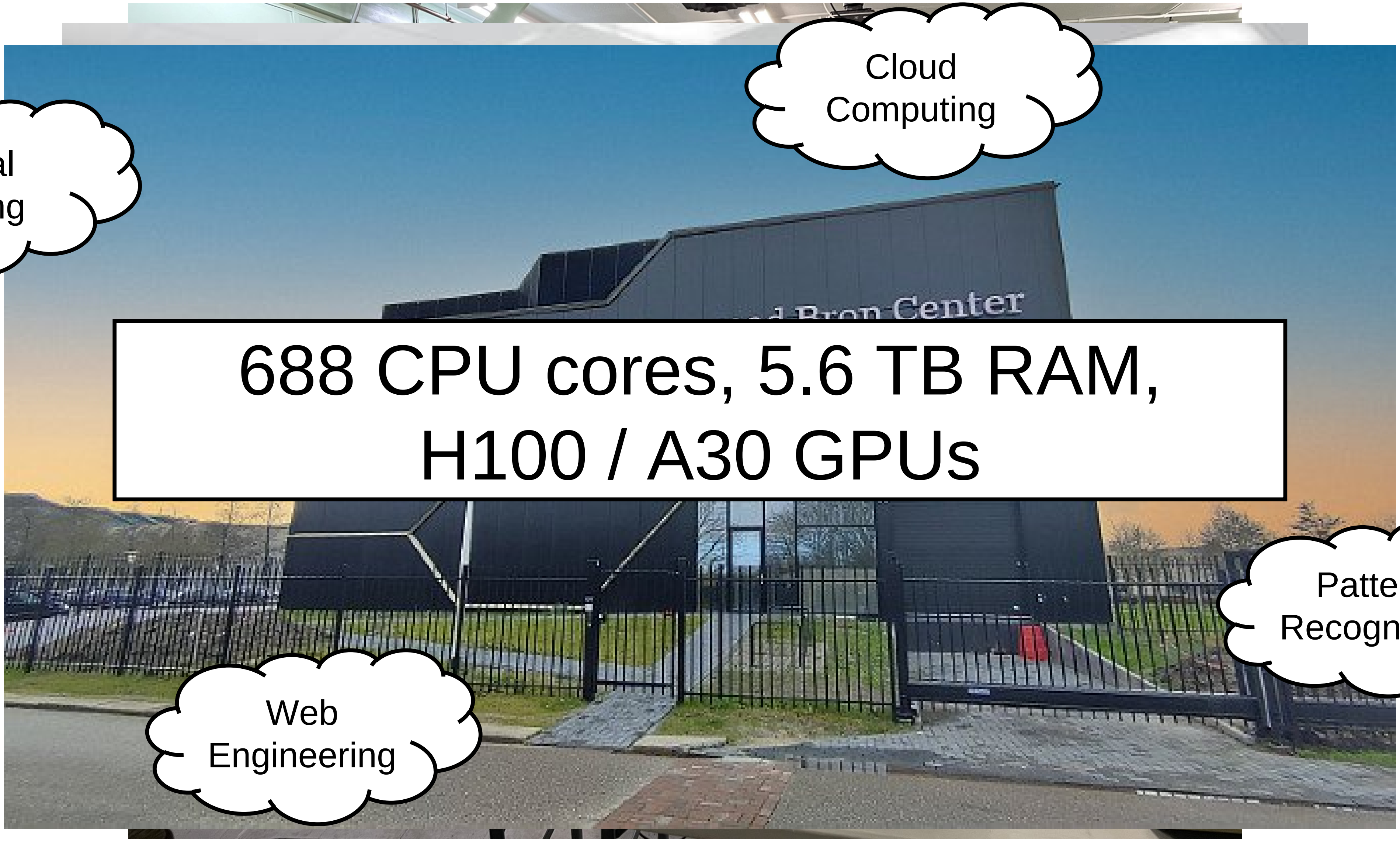
Started: Wed Sep 18 2024 20:09:19 GMT+0200

Duration: 8 sec 335.000 ms

1	Passed | 1 Point Test files: Input | Expected output | Your output | Error output | Difference | Resource Usage
2-valgrind	Passed | 1 Point Test files: Input | Expected output | Your output | Error output | Difference | Resource Usage
3	Passed | 1 Point Test files: Input | Expected output | Your output | Error output | Difference | Resource Usage
4	Passed | 1 Point Test files: Input | Expected output | Your output | Error output | Difference | Resource Usage
5	Passed | 1 Point Test files: Input | Expected output | Your output | Error output | Difference | Resource Usage
6	Passed | 1 Point Test files: Input | Expected output | Your output | Error output |
	Error output |
	Error output |
	Passed Test files: Input | Expected output | Your output | Error output |

Carbon footprint of a submission, bring awareness

Virtual Lab, a cloud-native approach to labs



Ethical Hacking

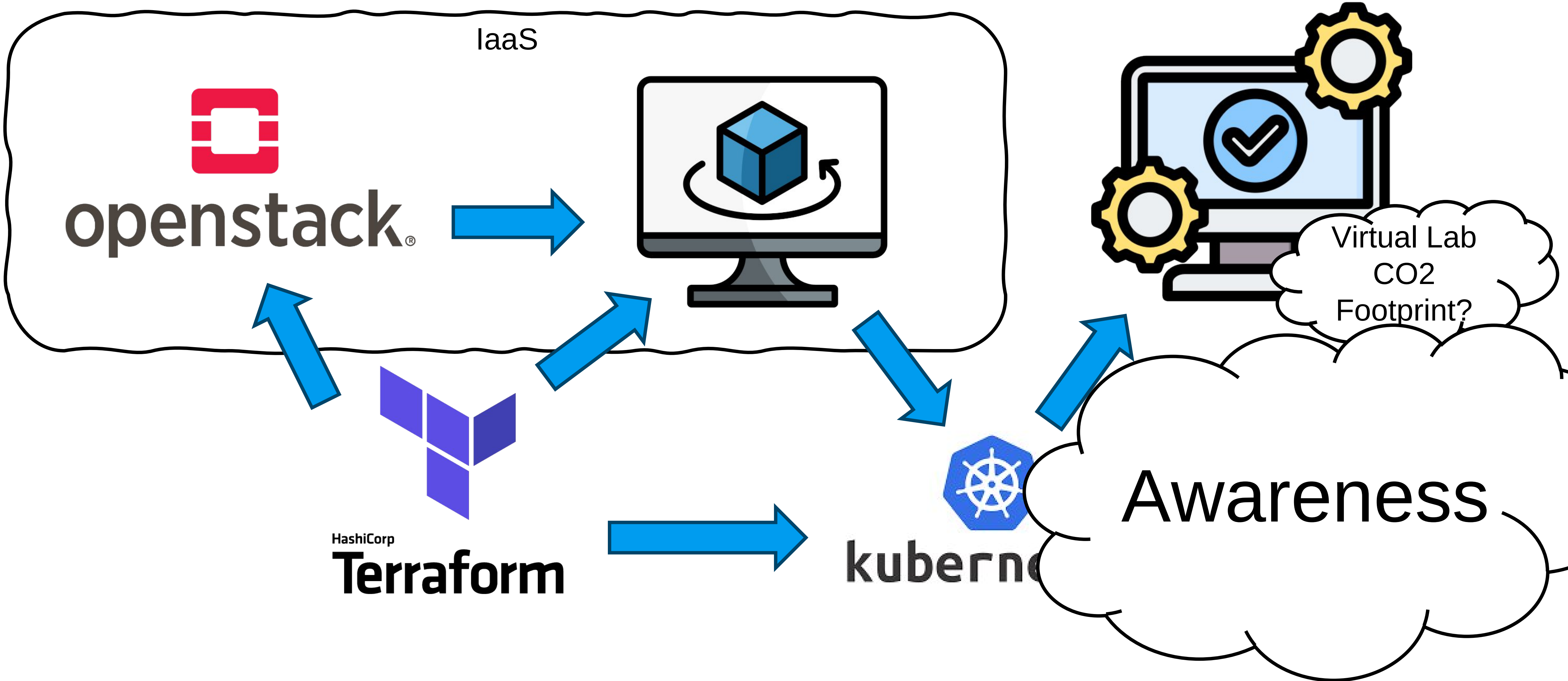
Cloud Computing

688 CPU cores, 5.6 TB RAM, H100 / A30 GPUs

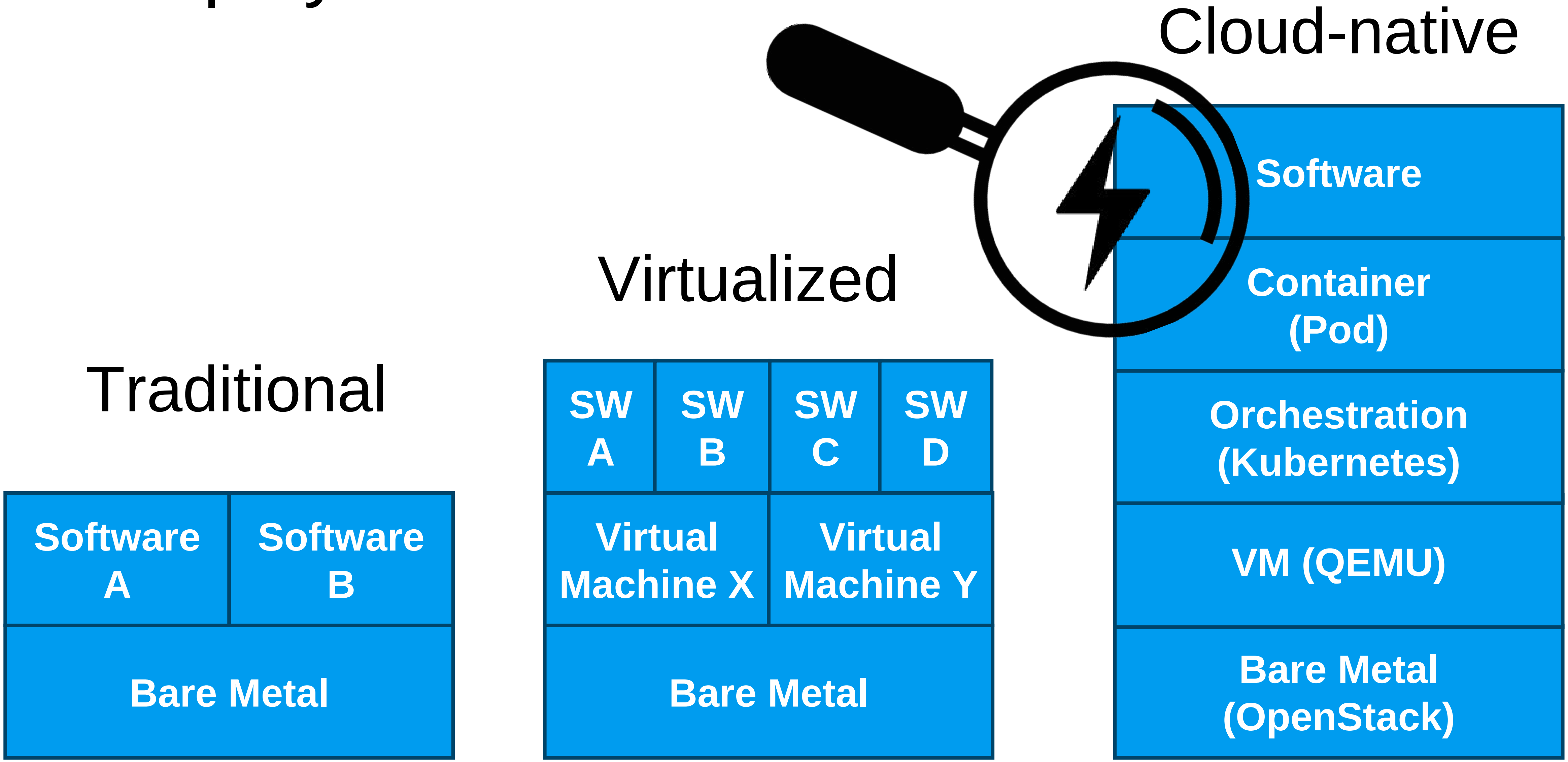
Pattern Recognition

Web Engineering

Virtual Lab at Scale



Deployment Models



Lars Andringa, Brian Setz, and Vasilios Andrikopoulos. 2025. Understanding the Energy Consumption of Cloud-native Software Systems. In Proceedings of the 16th ACM/SPEC International Conference on Performance Engineering (ICPE '25). Association for Computing Machinery, New York, NY, USA, 309–319. https://doi.org/10.1145/3676151.3719371

ICPE2025, Artifacts Available

UNDERSTANDING THE ENERGY CONSUMPTION OF CLOUD-NATIVE SOFTWARE SYSTEMS

Research Questions

- What is the relation between load and energy consumption in typical cloud-native applications across different abstraction layers?
  - How can cloud-native applications' energy consumption be observed using existing solutions?
  - What level of accuracy can an observability stack achieve in terms of energy estimation?

Profiling

Billing

Sustainability

Layers of Abstraction

RU = Resource Usage
EC = Energy Consumption

Software

Orchestration (Kubernetes)

VM (QEMU)

OpenStack

Bare Metal

Tool	BM		OS		VM		K8s		App	
	RU	EC	RU	EC	RU	EC	RU	EC	RU	EC
Prometheus Node Exporter	✓	✓	✗	✗	✓	✗	✗	✗	✗	✗
Telegraf	✓	✓	✓	✗	✓	✗	✓	✗	✓	✗
OProfile	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Linux top	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Linux sar	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Linux dstat	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Glances	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗
Monasca	✓	✗	✓	✗	✗	✗	✓	✗	✗	✗
Ceilometer	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
OpenStack-Exporter	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
cAdvisor	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
Kube State Metrics	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Kubernetes Metrics Server	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
Prometheus Kubernetes Agent	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
Resource Metrics API	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Kepler	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗
Scaphandre	✓	✓	✗	✗	✓	✓	✗	✓	✗	✗
Jeager	✓	✓	✓	✗	✓	✗	✓	✗	✓	✗
Prometheus	✓	✓	✓	✗	✓	✓	✓	✓	✓	✗

Tool Selection

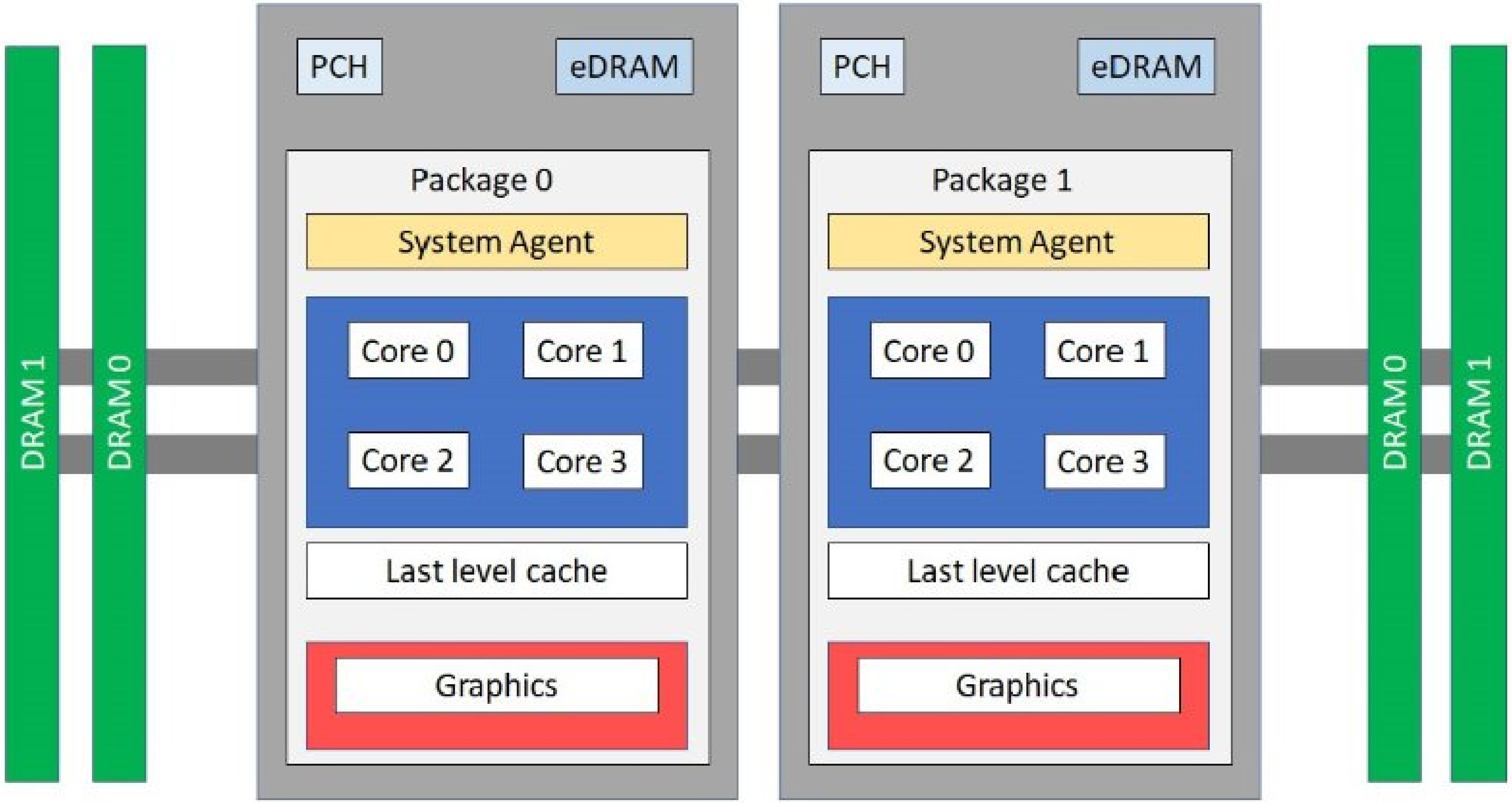
	Resource Usage	Energy Consumption
Software	Open Telemetry	none available!
Orchestration (Kubernetes)	cAdvisor	Kepler
VM (QEMU)	node_exporter	Scaphandre
OpenStack	none worked!	none available!
Bare Metal	node_exporter	RAPL

RAPL

- Running Average Power Limit (RAPL), designed by Intel & adopted by AMD

- Original purpose: balance performance vs. efficiency by enforcing limits

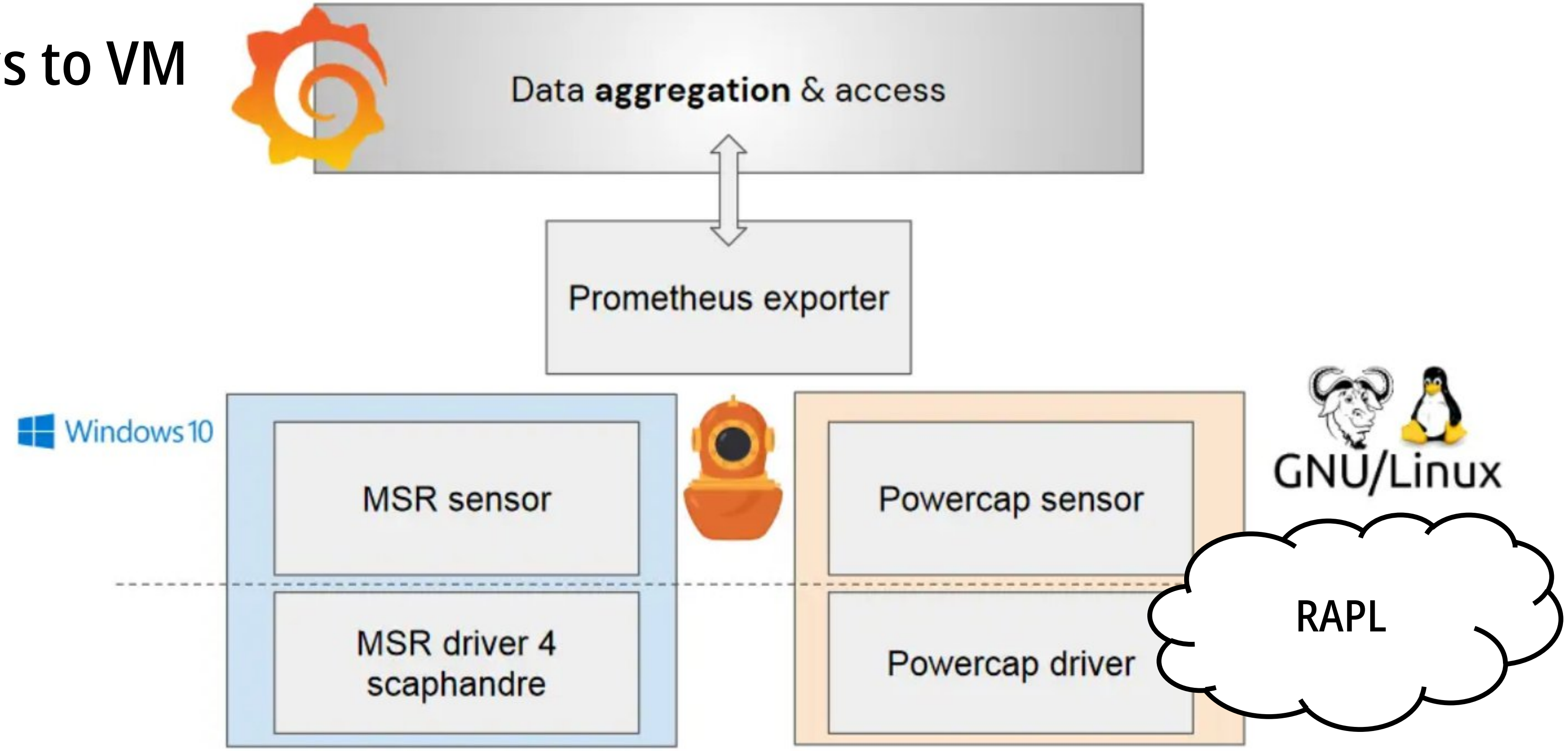
- Package
- Core
- Uncore
- DRAM



Scaphandre

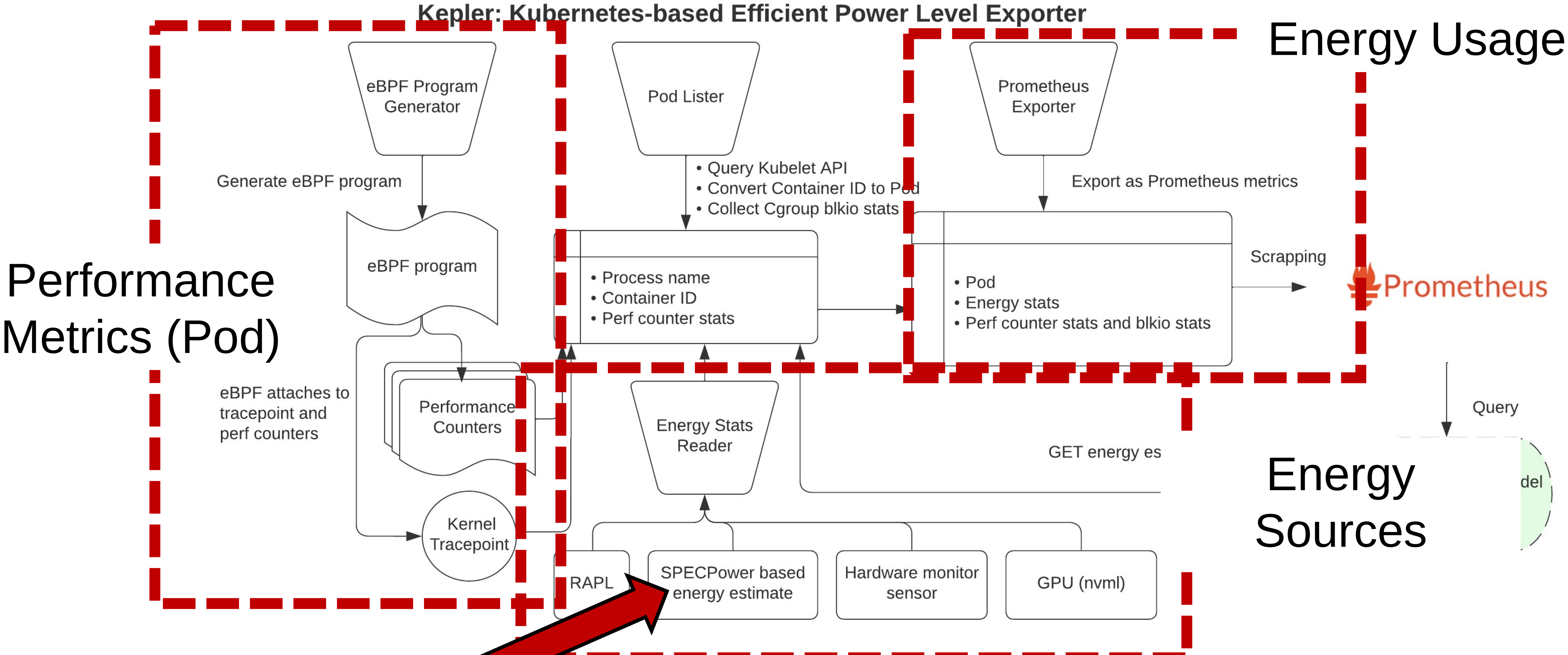
- Energy Consumption Metrics Agent
  - Bare Metal
  - Virtual Machines (from host)

- Exposes host metrics to VM



Kepler

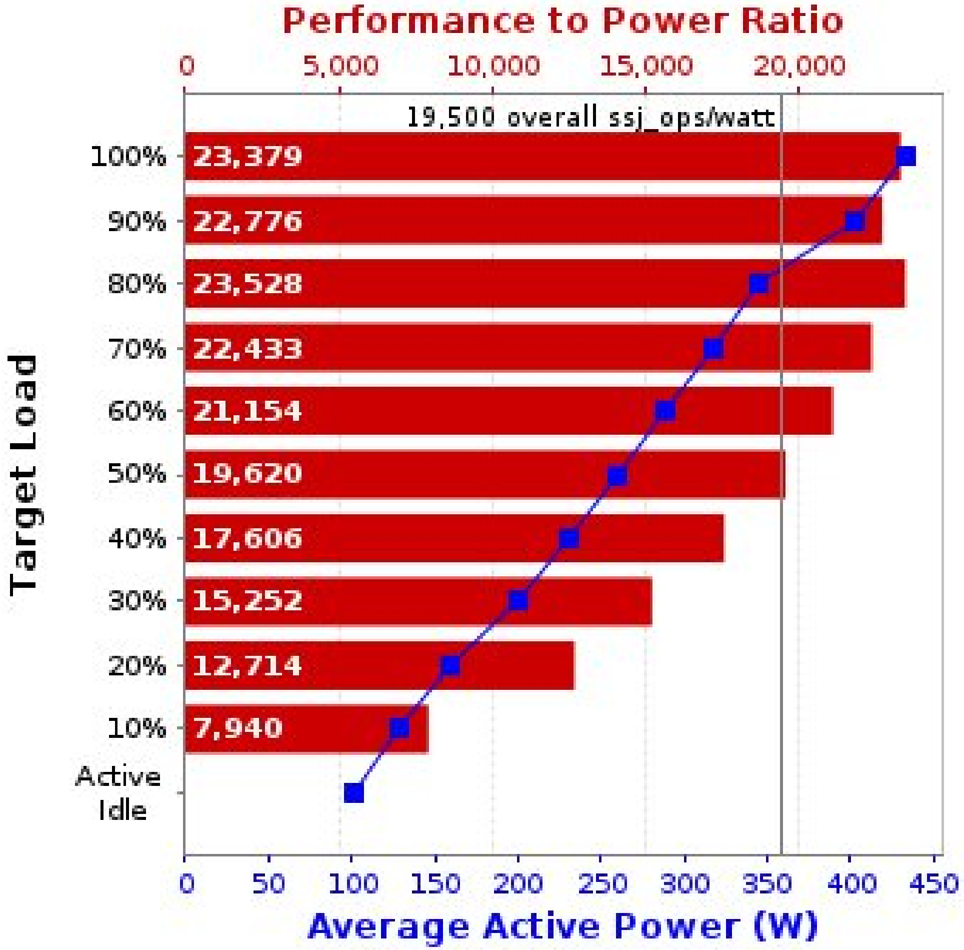
- Kubernetes Efficient Power Level ExporteR (Kepler)
  - Energy consumption of pods and nodes



Naïve consumption mapping

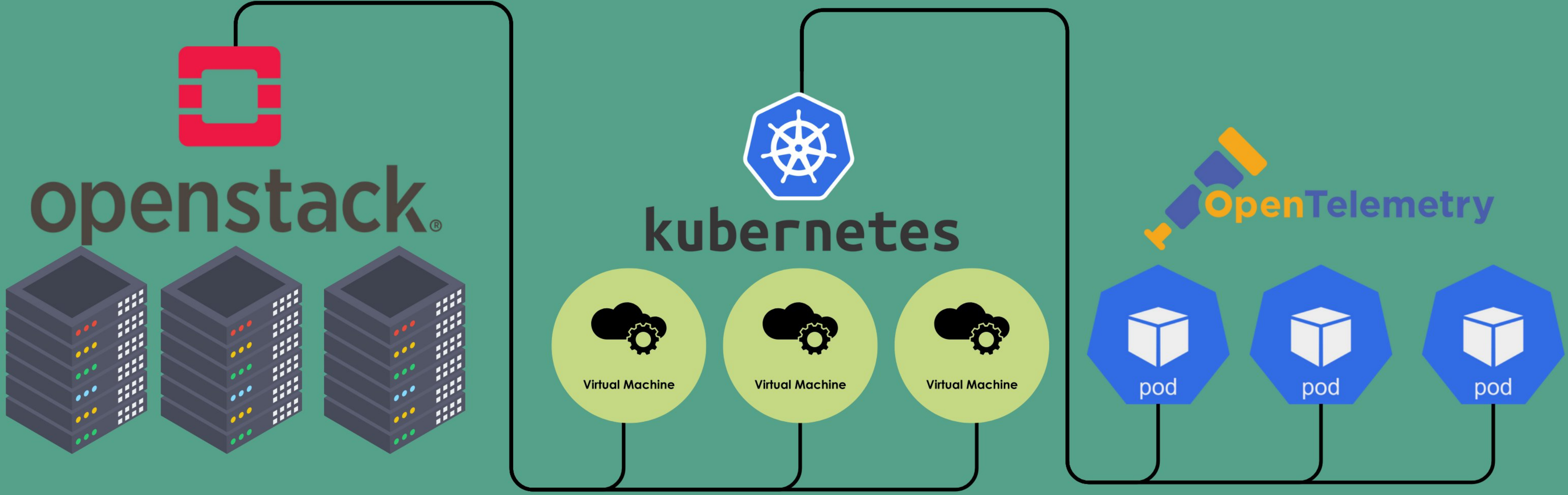
$$VM\_Energy\_Consumption = BM\_Energy\_Consumption \cdot \frac{VM\_CPU\_Utilization}{BM\_CPU\_Utilization}$$

$$Pod\_Energy\_Consumption = VM\_Energy\_Consumption \cdot \frac{Pod\_CPU\_Utilization}{VM\_CPU\_Utilization}$$

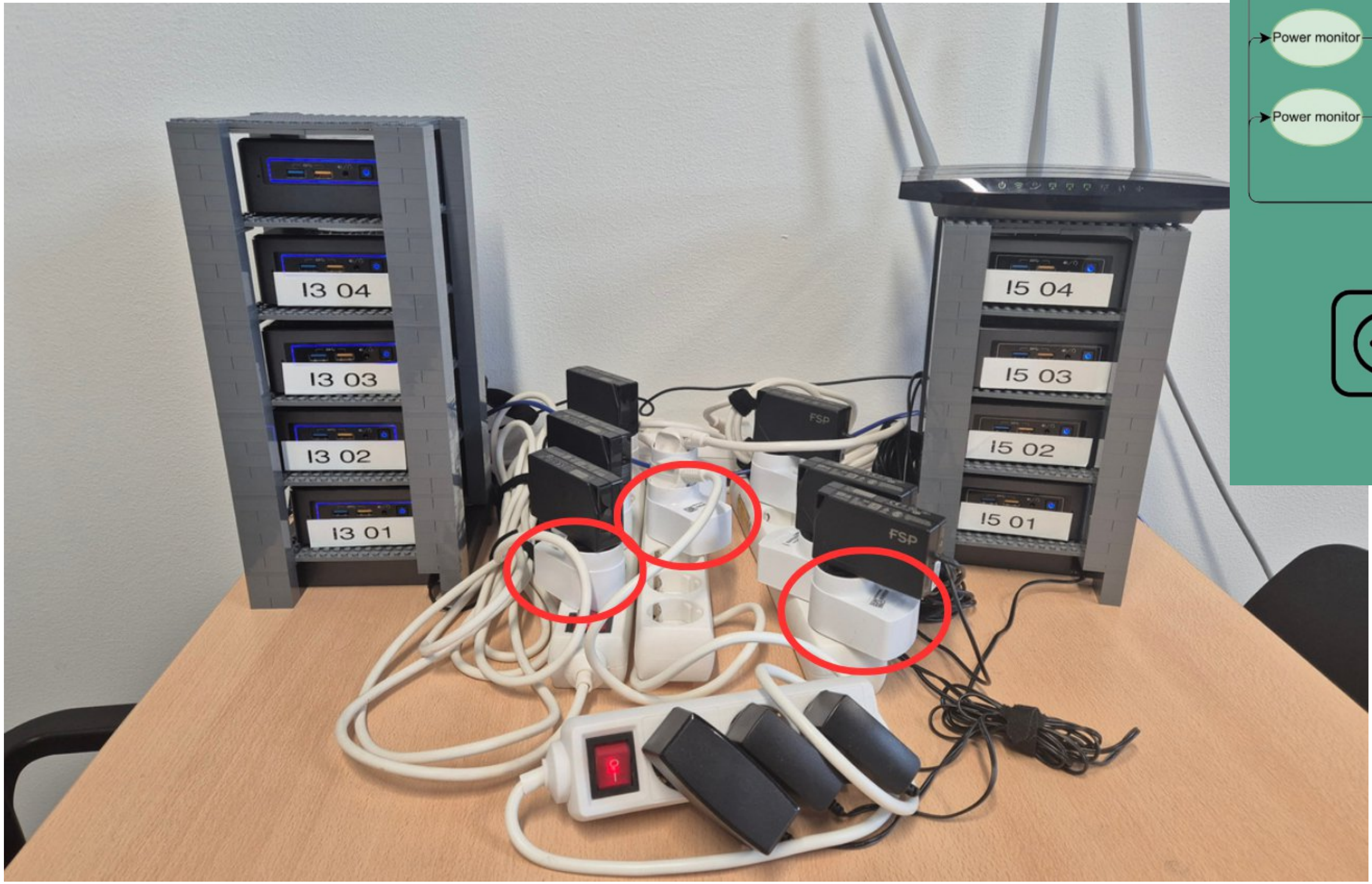


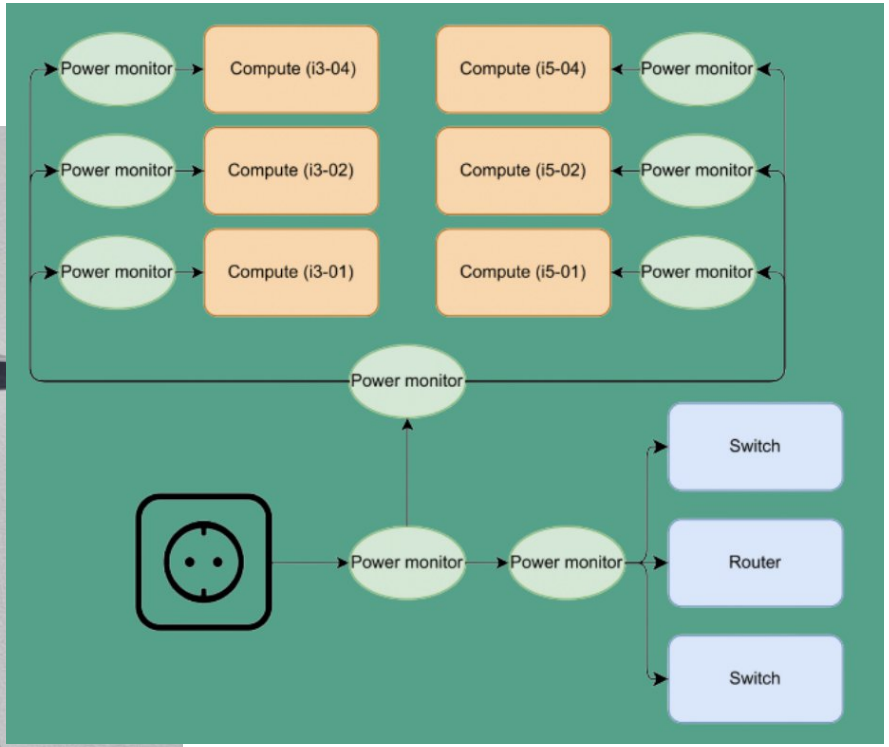
Intel(R) Xeon(R) Gold 6780E, 2.20GHz, 144 Cores

System Under Test



System Under Test





What data did we collect?

Open
Data Set!

- Physical
  - Energy Consumption (Bare Metal + Networking)
- Bare Metal
  - Energy Consumption (RAPL)
  - CPU, Mem, Disk
- Virtual Machine
  - Energy Consumption (Scaphandre)
  - CPU, Mem, Disk
- Kubernetes
  - Energy Consumption (Kepler)
  - CPU, Mem, Disk
- Application
  - Requests

https://doi.org/10.5281/zenodo.14967419

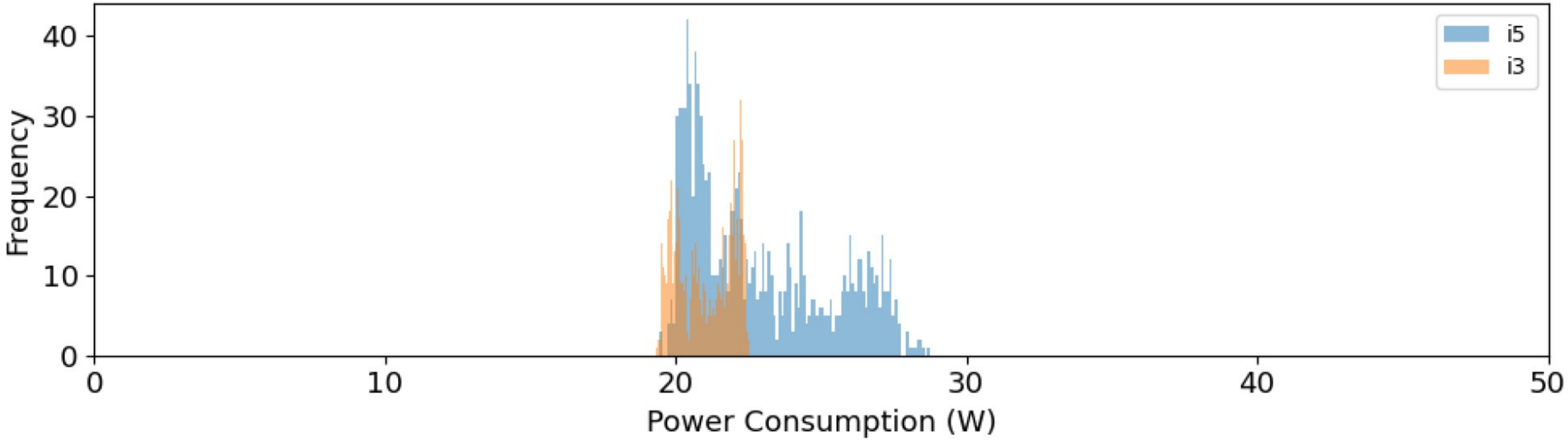
Experiments

- **Constant Load**
  - 6 different levels of constant load on the application
  - No autoscaling
- **Linear Load**
  - Linear scaling of load on the application
  - With horizontal pod autoscaling
- **Direct Load**
  - Run pods with an exact load (200 mCPU)
  - With linear scaling by spawning more pods

How accurate is RAPL?

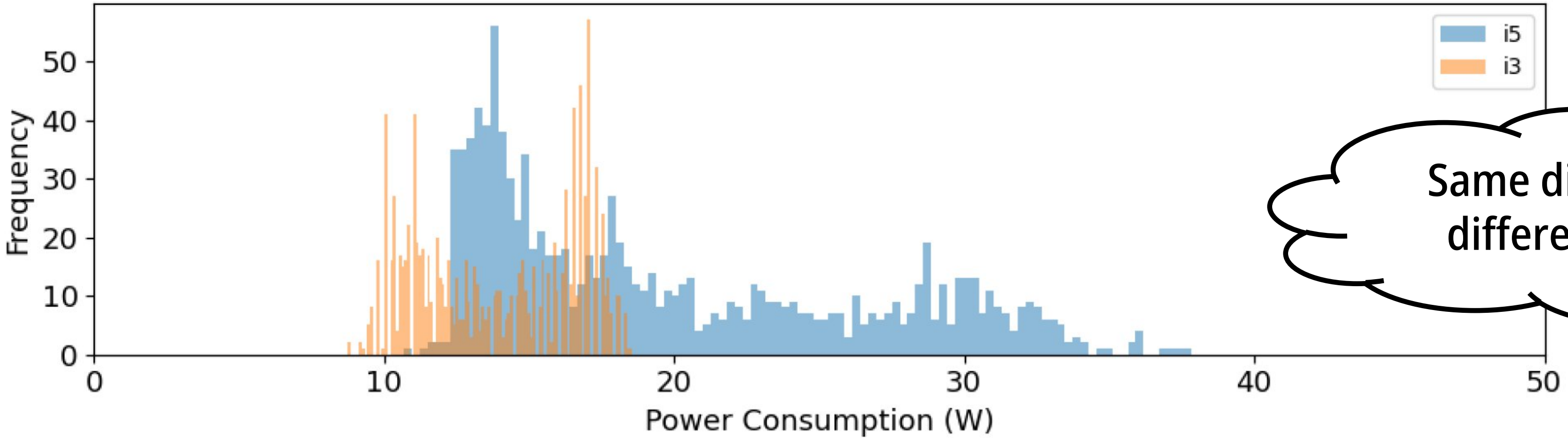
RAPL = BM

RAPL



(a) RAPL.

Power Plugs



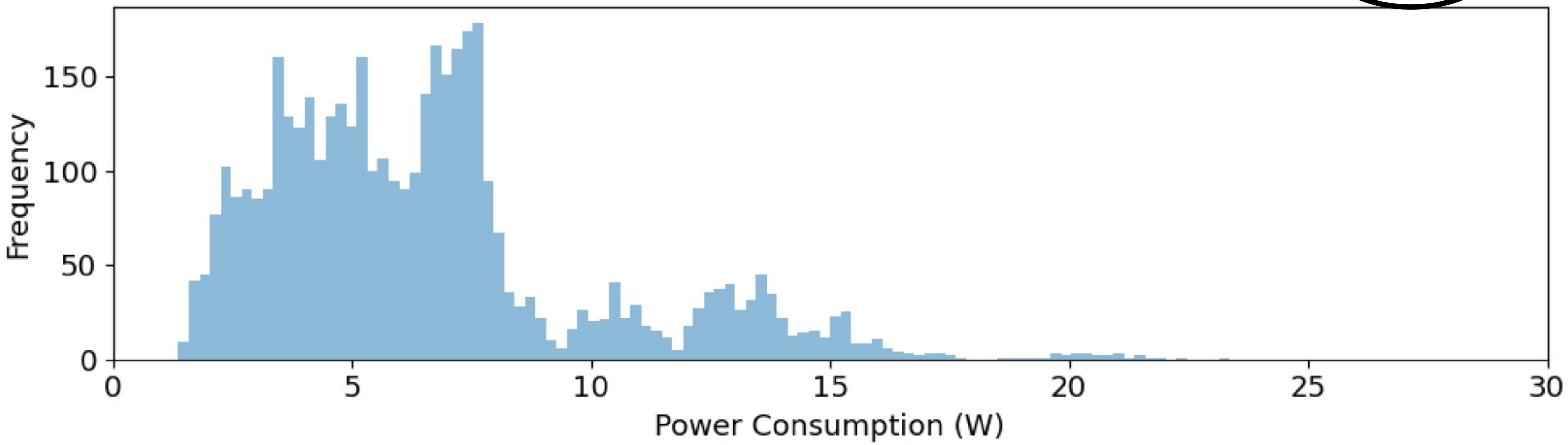
(b) Power plugs.

Same distribution, different scaling

How accurate is Scaphandre?

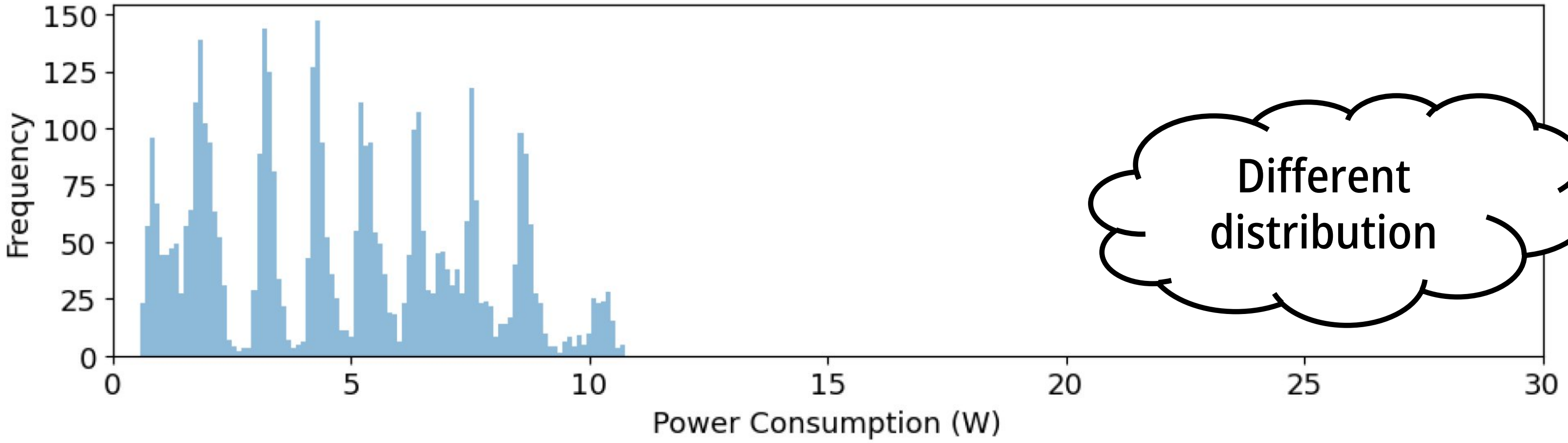
Scaph. = VM

VM mapping



(a) Virtual machine power consumption mapping.

Scaphandre

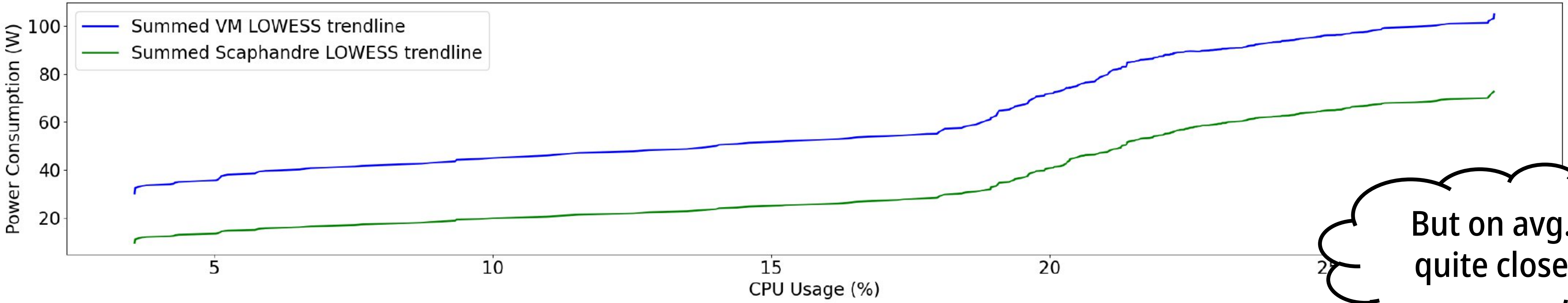


Different distribution

(b) Scaphandre power consumption estimation.

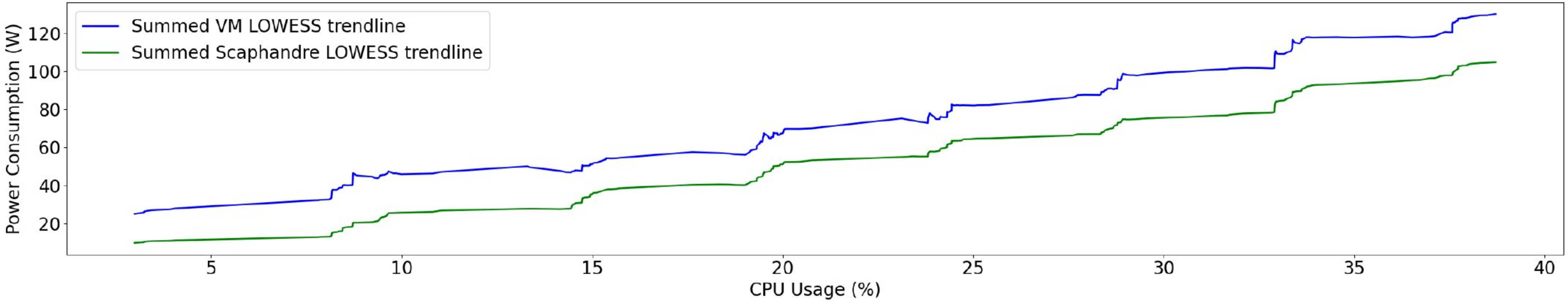
Naïve vs. Scaphandre

Blue = VM mapping
Green = Scaphandre



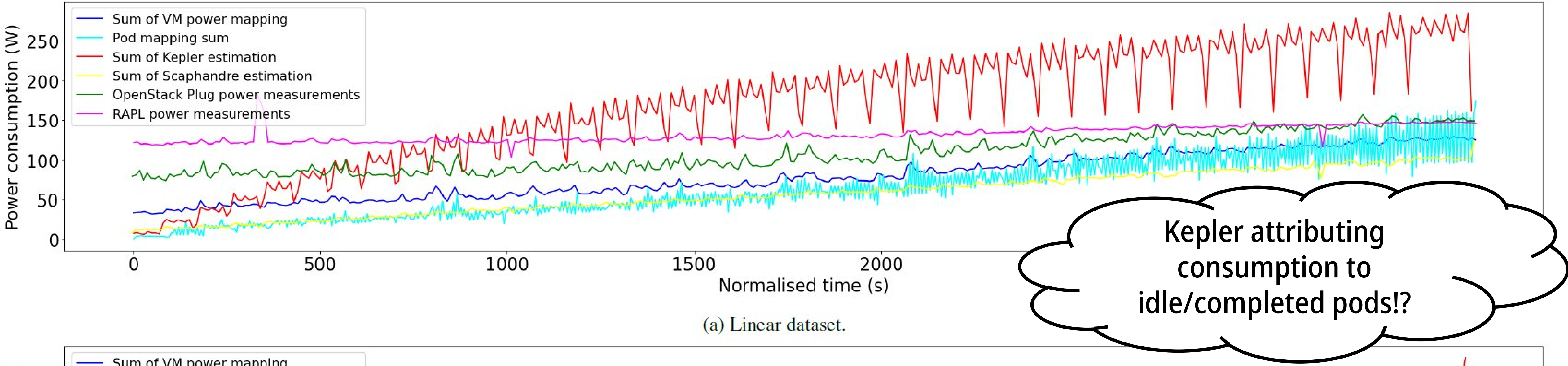
But on avg.
quite close

(a) Linear dataset.

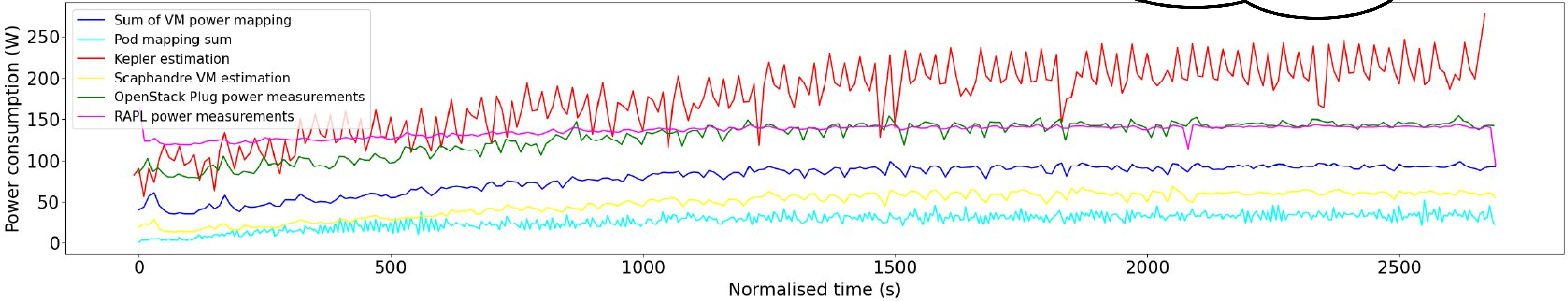


(b) Direct dataset.

How accurate is Kepler using Power Model?



(a) Linear dataset.



(b) Direct dataset.

Findings

- We have the tools for multi-layer monitoring of cloud-native environments!
- But, the tools for energy estimations are inaccurate
  - Especially RAPL, and Kepler
- Monitoring stack functions on private clouds, but what about public clouds?
- More research needed into Kepler

Bjorn Pijnacker, Brian Setz, and Vasilios Andrikopoulos. 2025. Container-level Energy Observability in Kubernetes Clusters. arXiv preprint arXiv:2504.10702. https://arxiv.org/abs/2504.10702

ICT4S 2025

CONTAINER-LEVEL ENERGY OBSERVABILITY IN KUBERNETES CLUSTERS

Research Question

- Previous work showed anomalies in the data produced by Kepler
- We have not found a systematic evaluation of Kepler’s accuracy beyond an initial evaluation
- Despite this, Kepler is used in a number of publications

How can we accurately estimate the power usage of Kubernetes containers based on external measurements?

Experiment Design

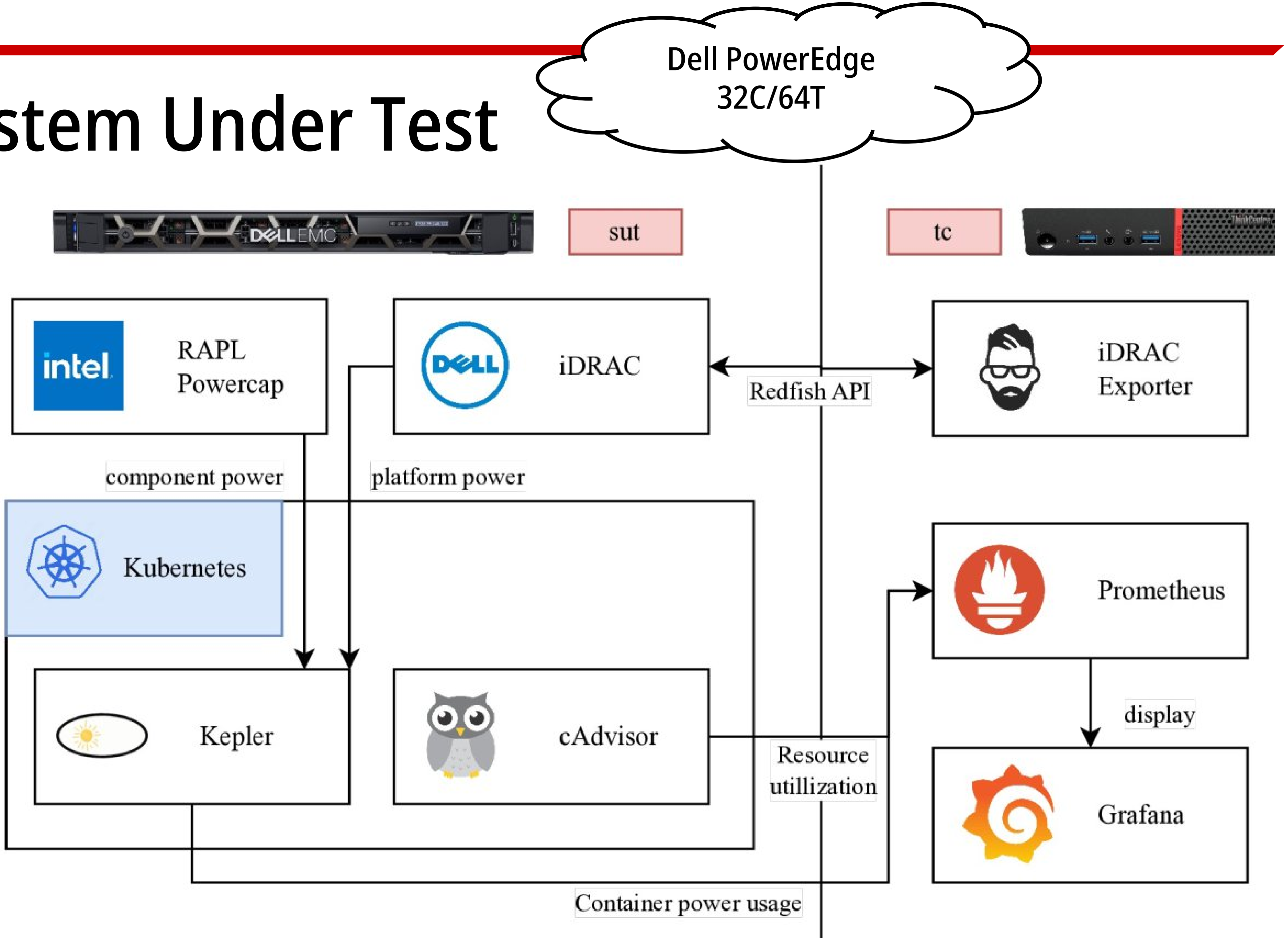
iDRAC data
(platform power)

KEPLER

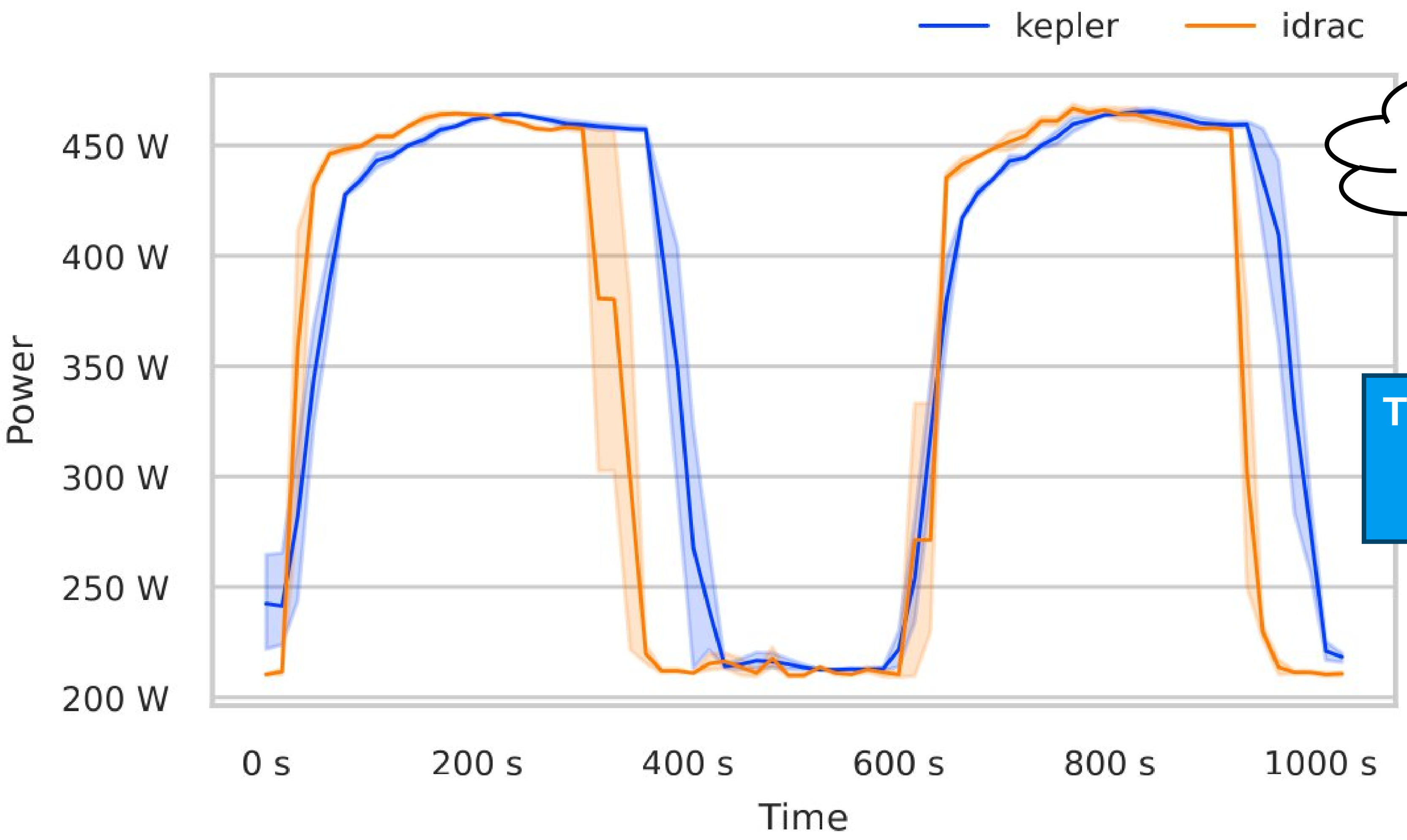
RAPL data
(core & package power)

Stressor Pod	Idle Pod	Idle Pod	Idle Pod	Idle Pod
Stress	Idle			
Kubernetes Namespaces				

System Under Test



How accurate is Kepler using BM data?



Orange = BM
Blue = Kepler

There is lag in the data, but otherwise totals are accurate

university of groningen

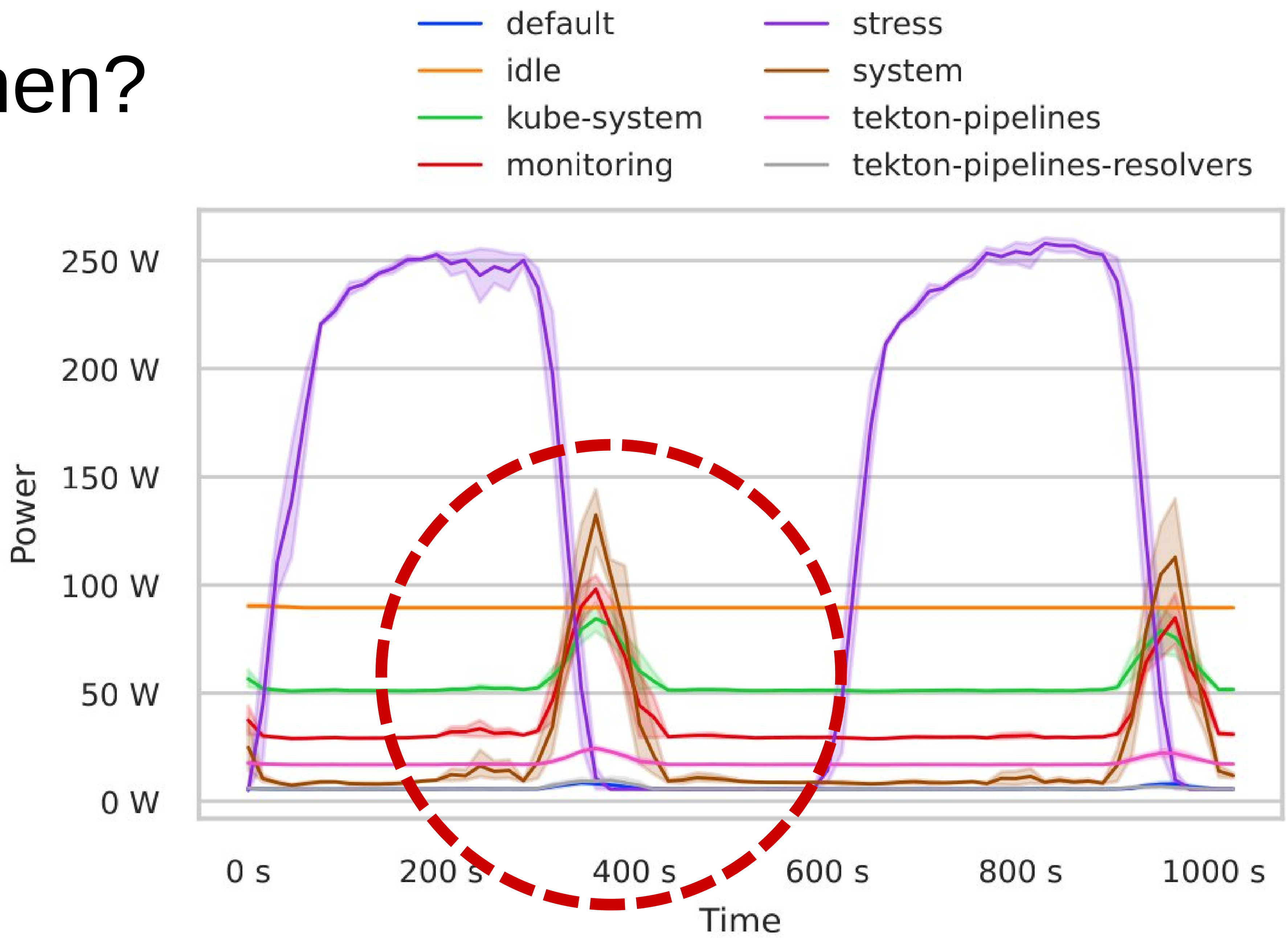
All is well then?

Experiment #1

Purple = Stressor

Other namespaces peak after stressor completes

Idle is constantly attributed +/- 100W while during stress test there is no load!



university of groningen

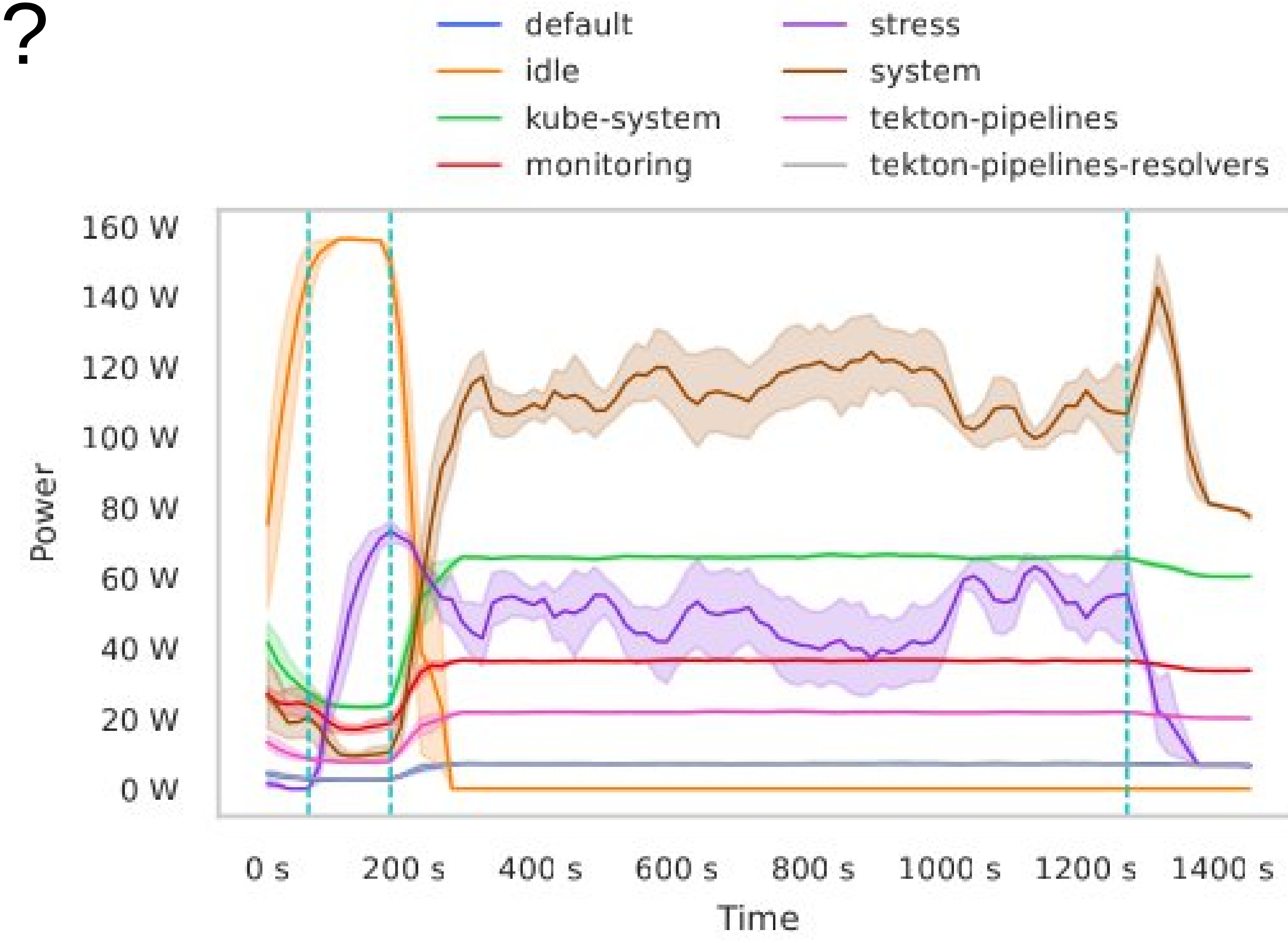
All is well then?

Experiment #2

Purple = Stressor
Orange = Idle
Brown = System

Idle has 64 inactive pods, at t=200, inactive pods are deleted

After t=200, stress power attribution drops, system power attribution increases

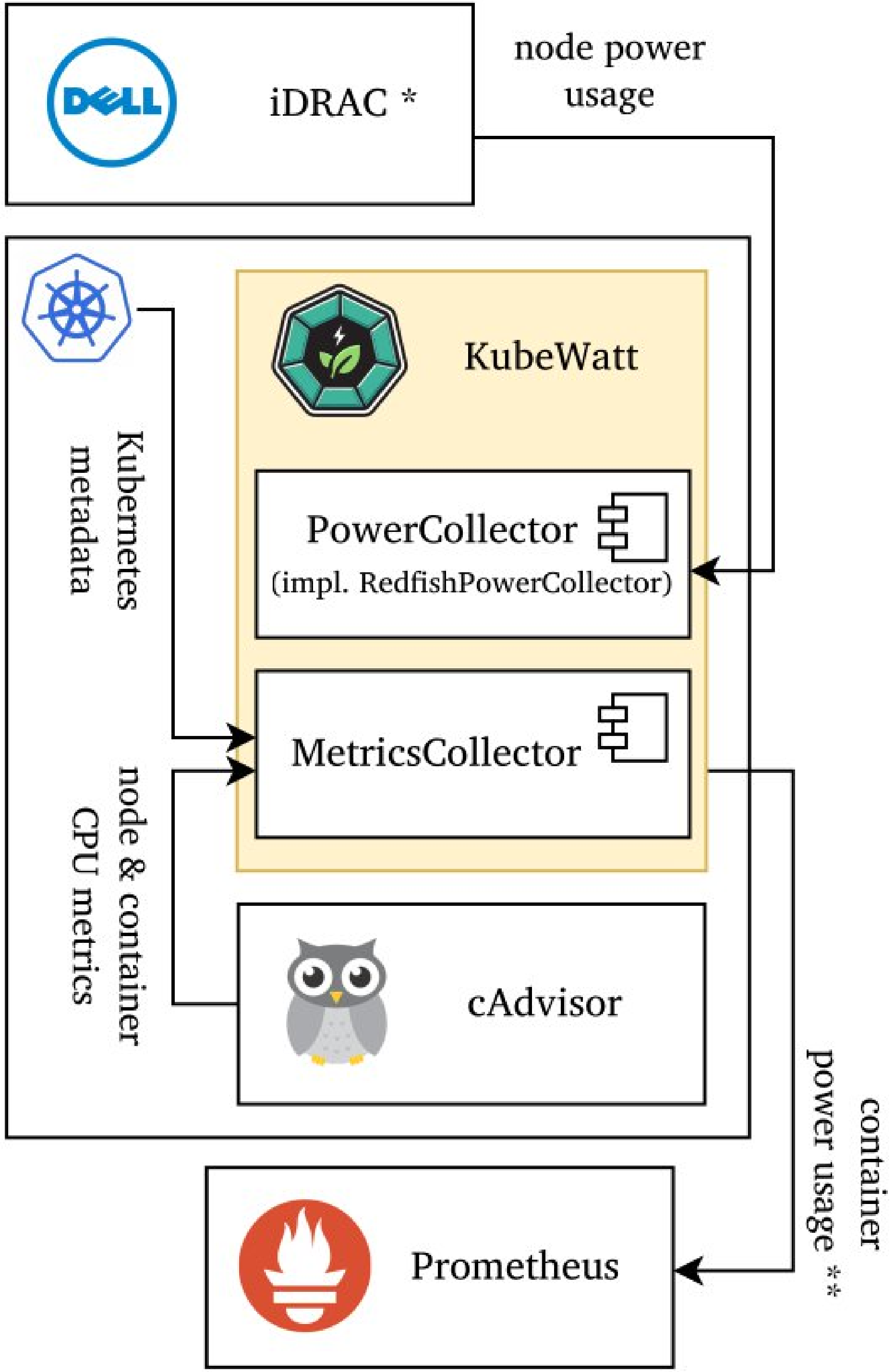


KubeWatt

› Proof of concept to determine if Kepler's shortcomings are inherent to the problem
  - Static Power
  - Dynamic Power

› Init modes to determine static power + train power model
  - Empty cluster
    - Over a period of time collect utilization an power draw
  - Existing cluster
    - Collect utilization and power draw until reaching enough variability (20-80% CPU usage)

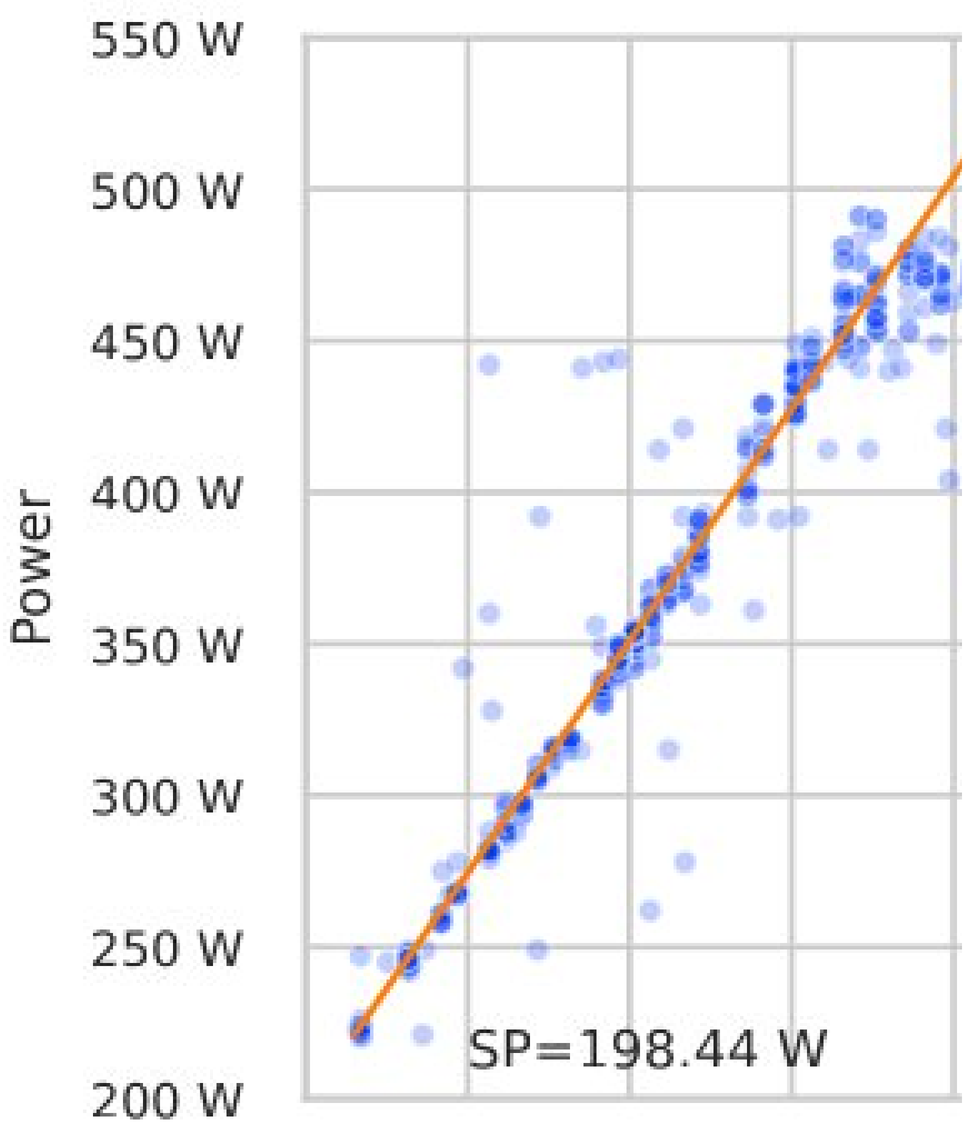
https://github.com/bjornpijnacker/kubewatt



Power Model

- Init mode: existing cluster
  - Data gathering, 3.5h to 13h until enough variability in the data
  - Static power estimated at 198.44 W
    - iDRAC reported 199.1W
  - R2 = 0.92

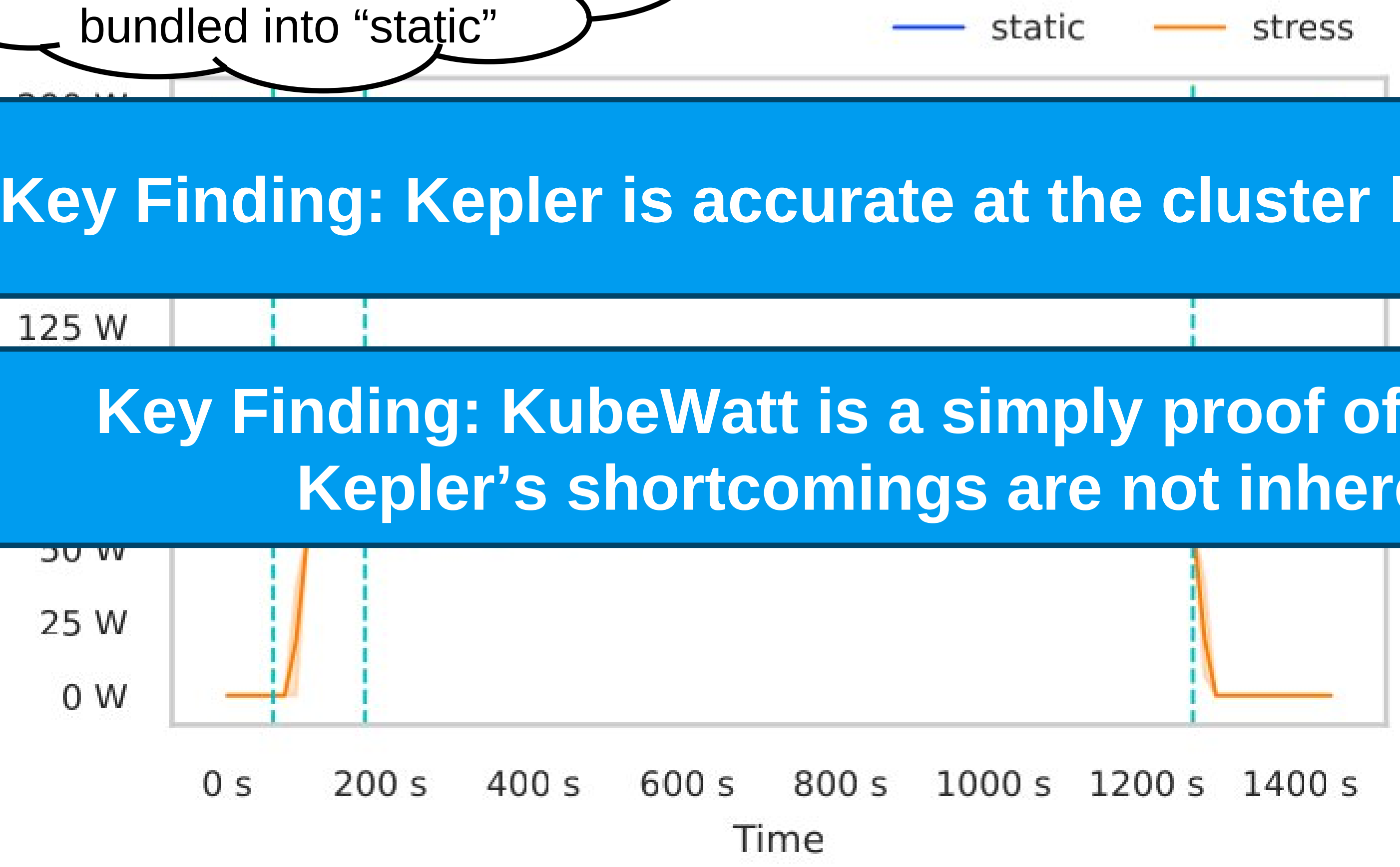
- After fitting a linear model we no longer need iDRAC data!

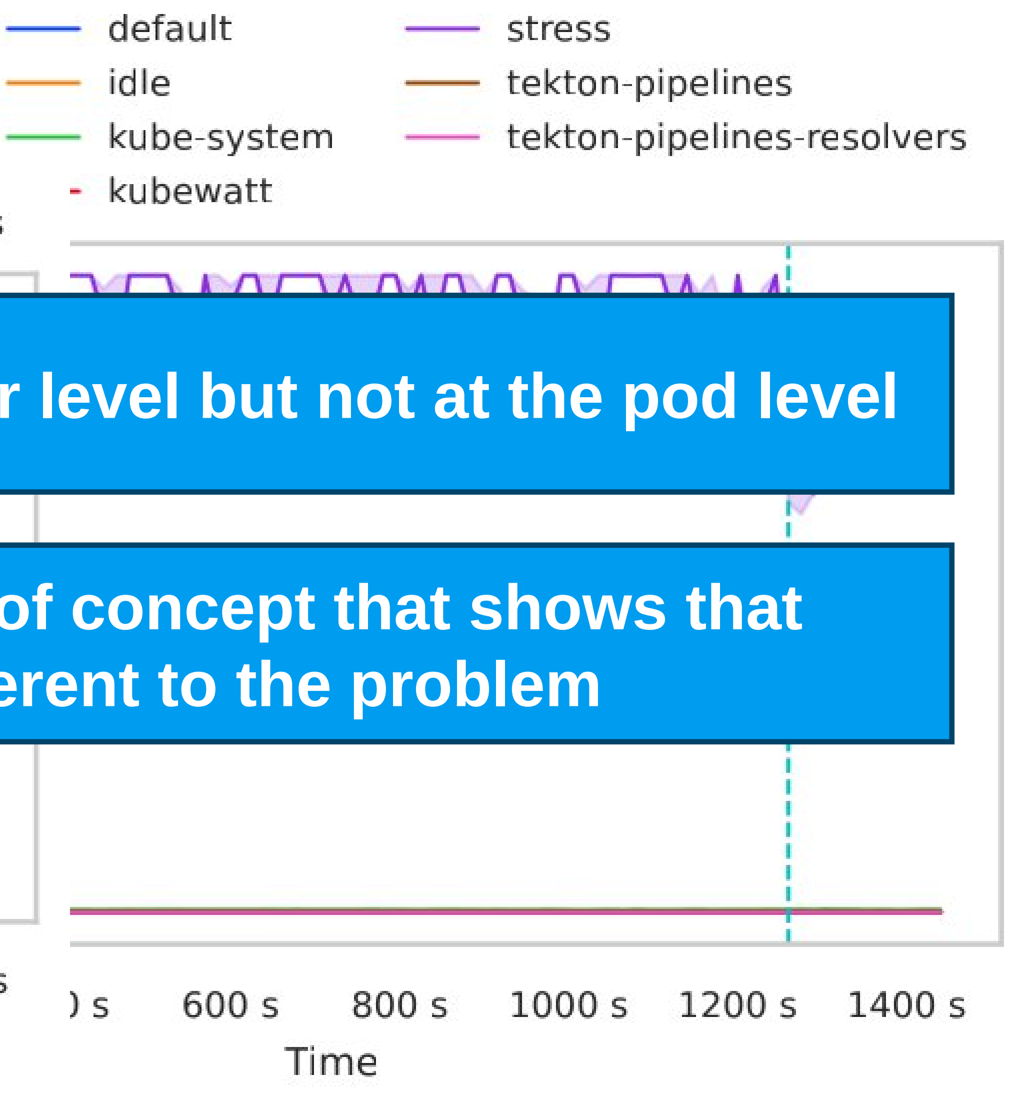


How does KubeWatt perform?

Repeat Experiment #2

Control plane namespaces are bundled into “static”





Key Finding: Kepler is accurate at the cluster level but not at the pod level

Key Finding: KubeWatt is a simply proof of concept that shows that Kepler’s shortcomings are not inherent to the problem

Take-home message

WHAT IS NEXT?

Conclusion

- Set out to build a monitoring stack using existing technology aimed at large scale cloud-native deployments to bring awareness to end-users

- Key technology exists but has critical flaws
  - **RAPL**, inaccurate, wrong scaling

How can we achieve true full-stack energy monitoring of cloud-native applications on public and private clouds?

- Proposed KubeWatt, but only solves a small part of the puzzle (k8s)

- Evaluated on “private clouds”, full BM access, what about public clouds?