

# Power, Energy, and Carbon-Aware Computing on Heterogeneous Systems (PEACHES)

Kerstin I. Eder<sup>\*1</sup>, Timo Hönig<sup>\*2</sup>, Maja Hanne Kirkeby<sup>\*3</sup>,  
Daniel Mosse<sup>\*4</sup>, Max Plauth<sup>\*5</sup>, and Jonas Juffinger<sup>†6</sup>

- 1 University of Bristol, GB. [cskie@bristol.ac.uk](mailto:cskie@bristol.ac.uk)
- 2 Ruhr-Universität Bochum, DE. [timo.hoenig@rub.de](mailto:timo.hoenig@rub.de)
- 3 Roskilde University, DK. [majaht@ruc.dk](mailto:majaht@ruc.dk)
- 4 University of Pittsburgh, US. [mosse@cs.pitt.edu](mailto:mosse@cs.pitt.edu)
- 5 UltiHash – Berlin, DE. [max.plauth@icloud.com](mailto:max.plauth@icloud.com)
- 6 TU Graz, AT. [jonas.juffinger@iaik.tugraz.at](mailto:jonas.juffinger@iaik.tugraz.at)

---

## Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 24351 “Power, Energy, and Carbon-Aware Computing on Heterogeneous Systems (PEACHES)”.

**Seminar** August 25–30, 2024 – <https://www.dagstuhl.de/24351>


**2012 ACM Subject Classification** Computer systems organization → Heterogeneous (hybrid) systems; General and reference; General and reference → Evaluation; General and reference → Measurement; General and reference → Metrics; Hardware → Power and energy; Software and its engineering → Operating systems

**Keywords and phrases** Carbon, Energy, Operating Systems, Power, Ressource Efficiency, Systems

**Digital Object Identifier** 10.4230/DagRep.14.8.36

## 1 Executive Summary

*Kerstin I. Eder (University of Bristol, GB)*  
*Timo Hönig (Ruhr-Universität Bochum, DE)*  
*Maja Hanne Kirkeby (Roskilde University, DK)*  
*Daniel Mosse (University of Pittsburgh, US)*  
*Max Plauth (UltiHash – Berlin, DE)*

**License**  Creative Commons BY 4.0 International license  
© Kerstin I. Eder, Timo Hönig, Maja Hanne Kirkeby, Daniel Mosse, and Max Plauth

The increasing carbon footprint of computing systems represents a critical and immediate challenge for the computing community, particularly as global digitalization accelerates. As computational demands grow – driven by data-intensive applications such as artificial intelligence – addressing the environmental impact of computing has become essential. The Dagstuhl Seminar 24351 on Power, Energy, and Carbon-Aware Computing on Heterogeneous Systems (PEACHES) focused on advancing the field’s understanding of how carbon emissions can be effectively measured, managed, and reduced across all layers of heterogeneous computing systems.

This seminar (PEACHES) brought together leading researchers and practitioners from the fields of computer science, software engineering, and environmental sustainability. The seminar participants explored 5 essential topics through continued group discussions. The

---

\* Editor / Organizer

† Editorial Assistant / Collector



Except where otherwise noted, content of this report is licensed under a Creative Commons BY 4.0 International license

Power, Energy, and Carbon-Aware Computing on Heterogeneous Systems (PEACHES), *Dagstuhl Reports*, Vol. 14, Issue 8, pp. 36–52

Editors: Kerstin I. Eder, Timo Hönig, Maja Hanne Kirkeby, Daniel Mosse, and Max Plauth



Dagstuhl Reports

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

participants explored boundaries, challenges, and possible new methods and techniques for: 1) obtaining **Carbon Transparency**, 2) reaching **Net-Zero in the Age of AI and Machine Learning**, and 3) achieving **Carbon-Aware Computing, Storage, and Communication**. In addition, we discussed and explored **Disruptive Paradigms**, focusing on innovative approaches needed to achieve net-zero carbon goals, rather than relying on incremental improvements. The seminar also featured enlightening **Carbon-aware Computing Hackathons**, introducing and discussing the latest research tools in the area, e.g., to reduce energy consumption via undervolting and tracking carbon emissions in applications. The seminar contributes to advancing the understanding of how software innovations can help mitigate the global environmental impact of computing.

## 2 Table of Contents

### Executive Summary

*Kerstin I. Eder, Timo Hönig, Maja Hanne Kirkeby, Daniel Mosse, and Max Plauth* 36

### Overview of Talks

Performance is not efficiency <i>Gustavo Alonso</i> . . . . .	40
Tape-based storage reduces CO2 emissions <i>Pierre Bennorth</i> . . . . .	40
Heterogeneous Computing for Energy with the TeamPlay Coordination Language <i>Clemens Grelck</i> . . . . .	41
Optimizing Green Coding Practices: Measurement Accuracy and Best Practices <i>Geerd-Dietger Hoffmann</i> . . . . .	42
Quantifying the potential of link sleeping in ISP networks <i>Romain Jacob</i> . . . . .	42
Cloud Server Design for Lower Carbon <i>Fiodar Kazhamiaka</i> . . . . .	43
Energy Labelling in Software <i>Maja Hanne Kirkeby</i> . . . . .	43
10 years of GCC from an energy perspective <i>Michael Kirkedal Thomsen and Maja Hanne Kirkeby</i> . . . . .	44
Trace & Tag: Towards Carbon-Aware Memory Placement <i>Sven Köhler and Timo Hönig</i> . . . . .	44
Working towards a standardized reporting and assessment of the energy efficiency of ML models training and inference <i>Silverio Martínez-Fernández</i> . . . . .	45
CXL Memory Expansion: A Closer Look on Actual Platforms <i>Vinicius Petrucci</i> . . . . .	46
Reducing Storage Emissions based on Deduplication <i>Max Plauth</i> . . . . .	46
The Cost of a Secure OS <i>Fabian Rauscher, Daniel Gruss, and Timo Hönig</i> . . . . .	47
Green AI: The more does not (always) lead to the better! <i>June Sallou</i> . . . . .	47
Executable explanations of control software for wind turbines <i>Sibylle Schupp</i> . . . . .	48
How to quantify the energy consumption of applications <i>Tamar Eilam</i> . . . . .	49
Energy-efficient systems through non-volatile memory <i>Manuel Vögele, Timo Hönig, and Wolfgang Schröder-Preikschat</i> . . . . .	49

**Working groups**

Hackathon: CPU Undervolting  
*Jonas Juffinger* . . . . . 50


Hackathon: Carbond  
*Robin Ohs, Benedict Herzog, Gregory Stock, Henry Janson, Timo Hömig, Luis Gerhorst, and Andreas Schmidt* . . . . . 50

**Participants** . . . . . 52

## 3 Overview of Talks

### 3.1 Performance is not efficiency

*Gustavo Alonso (ETH Zürich, CH)*

License  Creative Commons BY 4.0 International license  
© Gustavo Alonso

Performance measures how well a system completes a task in terms of throughput and latency, typically with the aim of maximizing the first and minimizing the latter. Efficiency measures how well the resources needed to accomplish a task are used focusing on reducing, e.g., utilization and idle times. The IT industry is focused almost exclusively on performance as seen in the common service level agreements used in industry and in benchmarks. Efficiency plays a role mainly in terms of overall monetary costs and does not necessarily include environmental impacts or externalized costs. In the talk I will discuss the difference between performance and efficiency in relation to Green IT with the goal of identifying the biggest targets for potential savings.

### 3.2 Tape-based storage reduces CO2 emissions

*Pierre Bennorth (CopenCloud – Karlslunde, DK)*

License  Creative Commons BY 4.0 International license  
© Pierre Bennorth

In today's digital age, vast amounts of data are stored on servers, much of which is rarely accessed but continuously consumes energy. At CopenCloud, we have developed a solution that categorizes data into two distinct groups: frequently accessed data stored on hard drives for instant access, and infrequently used data stored on energy-efficient tape drives. This approach allows us to significantly reduce energy consumption and CO2 emissions, with up to 90% savings compared to traditional storage methods.

Our CO2 reduction claims have been validated by the esteemed Danish consultancy Rambøll, whose report [1] confirms that our tape-based storage reduces CO2 emissions by 90% compared to hard drives. The report also highlights that globally, 5.6 zettabytes of data are ideal for long-term tape storage, and around 80% of organizational data can benefit from this method.

Additionally, for every 17 terabytes stored on tape, we can save one ton of CO2 annually. To aid organizations in optimizing their data storage, we've developed a Directory Scanner that identifies data suitable for long-term storage. We also provide a CO2 report for ESG accounts, ensuring our clients can track their environmental impact.

Our solution prioritizes security, featuring write protection via S3 and Glacier Protocol, making it resilient against ransomware attacks. Furthermore, our platform is designed to minimize energy consumption while enabling faster uploads and downloads. We are collaborating with Maja H. Kirkeby on the project "Software's energy consumption at the end-user device" to have these capabilities independently verified.

#### References

- 1 Rambøll, "CopenCloud Rapport Intelligent data storage can optimize organizations' costs and lower their climate footprint." Available at: [https://copencloud.com/docs/CopenCloud\\_Report\\_EN.pdf](https://copencloud.com/docs/CopenCloud_Report_EN.pdf), 2023. Accessed: 2024-09-12.

### 3.3 Heterogeneous Computing for Energy with the TeamPlay Coordination Language

Clemens Grellck (Friedrich-Schiller-Universität Jena, DE)

**License** © Creative Commons BY 4.0 International license  
© Clemens Grellck

**Joint work of** Clemens Grellck, Sebastian Altmeyer, Benjamin Rouxel, Julius Roeder, Andy D. Pimentel, Lukas Miedema, Ulrik Pagh Schultz, Jesper Holst, Ole Jørgensen, Benny Åkesson

**Main reference** Julius Roeder, Benjamin Rouxel, Sebastian Altmeyer, Clemens Grellck: “Towards Energy-, Time- and Security-Aware Multi-core Coordination”, in Proc. of the Coordination Models and Languages – 22nd IFIP WG 6.1 International Conference, COORDINATION 2020, Held as Part of the 15th International Federated Conference on Distributed Computing Techniques, DisCoTec 2020, Valletta, Malta, June 15-19, 2020, Proceedings, Lecture Notes in Computer Science, Vol. 12134, pp. 57–74, Springer, 2020.

**URL** [https://doi.org/10.1007/978-3-030-50029-0\\_4](https://doi.org/10.1007/978-3-030-50029-0_4)

Heterogeneous compute architectures are key to energy savings, provided that software chooses to components of software systems on the individually best suited hardware components available. High-performance embedded devices and other battery-powered devices from mobile phones to laptops are typically equipped with various instances of multiple types of CPU and GPU cores with highly different performance/energy trade-offs and characteristics. The open question is how applications can be mapped to such architectures under global objectives such as meeting deadlines or achieving required performance levels while using the minimal amount of energy possible.

The TeamPlay coordination language is geared towards describing componentised applications, their dependencies, constraints, and objectives on a high level of abstraction. As a coordination language TeamPlay delegates the implementation of software components of reasonable size and complexity to established programming languages for programming-in-the-small and entirely focusses on coordination aspects such as time-wise and spatial scheduling, dependencies and communication, etc, all in line with the aforementioned global objectives. Compiler and runtime system middleware are charged with establishing transparency about the inevitable trade-offs and actually run applications on the heterogeneously parallel system architectures targeted.

#### References

- 1 J. Roeder, B. Rouxel, S. Altmeyer, C/ Grellck. *Towards Energy-, Time- and Security-aware Multi-core Coordination*. 22nd International Conference on Coordination Models and Languages (COORDINATION 2020), virtual from Valletta, Malta, Lecture Notes in Computer Science 12134, pp. 57-74.
- 2 B. Rouxel, U. Pagh Schultz, B. Åkesson, J. Holst, O. Jørgensen, C. Grellck. *PReGO: a generative methodology for satisfying real-time requirements on COTS-based systems: Definition and experience report*. 19th ACM SIGPLAN International Conference on Generative Programming: Concepts and Experiences (GPCE 2020), virtual from Chicago, USA, ACM, 2020.
- 3 J. Roeder, B. Rouxel, S. Altmeyer, C. Grellck. *Energy-aware Scheduling of Multi-version Tasks on Heterogeneous Real-time Systems*. 36th ACM/SIGAPP Symposium on Applied Computing (SAC 2021), pp. 500–510, ACM, 2021.
- 4 B. Rouxel, S. Altmeyer, C. Grellck. *YASMIN: a Real-time Middleware for COTS Heterogeneous Platforms*. 22nd ACM/IFIP International Middleware Conference (MIDDLEWARE 2021), pp. 298-309, virtual from Montréal, Canada, ACM, 2021.
- 5 J. Roeder, A.D. Pimentel, C. Grellck. *GCN-based Reinforcement Learning Approach for Scheduling DAG Applications*. Artificial Intelligence Applications and Innovations, 19th IFIP WG 12.5 International Conference (AIAI 2023), León, Spain, IFIPAICT Vol. 676, pp. 121-134, Springer, 2023.

### 3.4 Optimizing Green Coding Practices: Measurement Accuracy and Best Practices

*Geerd-Dietger Hoffmann (Green Coding Solution – Berlin, DE)*

**License** © Creative Commons BY 4.0 International license  
© Geerd-Dietger Hoffmann

In the quest for sustainable and efficient software, accurate energy consumption measurement is paramount. This talk addresses common pitfalls and best practices in green coding, specifically focusing on the importance of understanding the tools and techniques used for energy measurement in software development. Through real-world examples and critical analysis, we explore the limitations of popular measurement tools like Intel RAPL and the Gude Power Meter, and discuss the necessity of consistent temperature conditions, disabling CPU features like Turbo Boost and Hyper-Threading, and the challenges of using CPU utilization as a reliable metric. Moreover, we emphasize the significance of validating machines for anomalies, automating processes, and considering the overhead of measurement setups. Practical advice on using specialized operating systems, non-standard sampling intervals, and handling external network traffic is also provided. This talk invites discussion and collaboration on improving green coding practices, ultimately aiming to refine how we measure and reduce the environmental impact of software development.

### 3.5 Quantifying the potential of link sleeping in ISP networks

*Romain Jacob (ETH Zürich, CH)*

**License** © Creative Commons BY 4.0 International license  
© Romain Jacob

**Main reference** Lukas Röllin, Romain Jacob, Laurent Vanbever: “A Sleep Study for ISP Networks: Evaluating Link Sleeping on Real World Data”, 2024.

**URL** <https://doi.org/10.3929/ethz-b-000680859>

It is widely accepted that energy proportionality, i.e., the ability to scale the power drawn by a system with its workload, is an enabler for energy efficiency. Putting parts of a system to sleep – turning them off or into some low-power mode – when they are not needed is the B-A-BA of proportionality. In networks, a simple part one can turn off is a link: if traffic volume is low, one can often route traffic away from some links that can be put to sleep to save energy. But how to select the links one can safely turn off without causing congestion elsewhere in the network? And how much energy would that actually save?

Our ongoing research addresses these questions. We found that, in lowly loaded networks, simple heuristics appear sufficient to decide which links to turn off. Using real topology and traffic data from two small ISPs (Switch (CH) and SURF (NL)), we found that about 1/3 of links can be turned off on average. As I will discuss in the talk, the conversion into expected energy savings requires further research.

### 3.6 Cloud Server Design for Lower Carbon

*Fiodar Kazhamiaka (Microsoft – Redmond, US)*

**License** © Creative Commons BY 4.0 International license  
© Fiodar Kazhamiaka

**Joint work of** Fiodar Kazhamiaka, Jaylen Wang, Daniel Berger, Fiodar Kazhamiaka, Celine Irvine, Chaojie Zhang, Esha Choukse, Kali Frost, Rodrigo Fonseca, Brijesh Warriar, Chetan Bansal, Jonathan Stern, Ricardo Bianchini, Akshitha Sriraman

**Main reference** Jaylen Wang, Daniel S. Berger, Fiodar Kazhamiaka, Celine Irvine, Chaojie Zhang, Esha Choukse, Kali Frost, Rodrigo Fonseca, Brijesh Warriar, Chetan Bansal, Jonathan Stern, Ricardo Bianchini, Akshitha Sriraman: “Designing Cloud Servers for Lower Carbon”, in Proc. of the 51st ACM/IEEE Annual International Symposium on Computer Architecture, ISCA 2024, Buenos Aires, Argentina, June 29 – July 3, 2024, pp. 452–470, IEEE, 2024.

**URL** <https://doi.org/10.1109/ISCA59077.2024.00041>

Cloud servers can be designed to optimize for carbon emissions. Hardware developments such as CXL and dense/efficient CPU cores offer improved performance/Watt and the ability to give a second life to older components with high embodied carbon, such as DIMMs. The talk covers the design of such servers, and a framework for evaluating the carbon savings at the datacenter, including features such as application performance, customer adoption, maintenance implications, and workload packing implications.

### 3.7 Energy Labelling in Software

*Maja Hanne Kirkeby (Roskilde University, DK)*

**License** © Creative Commons BY 4.0 International license  
© Maja Hanne Kirkeby

**Joint work of** Maja Hanne Kirkeby, Kerstin I. Eder, E. B. Unna-Lindhard, N. Müllenborn

This presentation explores the energy consumption associated with Information and Communication Technology (ICT) systems, with a particular focus on how software influences overall energy use. It proposes the introduction of energy labels for software and software modules. ICT’s contribution to global greenhouse gas (GHG) emissions is substantial, accounting for an estimated 1.8 % to 2.8 % of total emissions, primarily driven by the operational phases of these systems [2]; e.g., the GHG emissions was evaluated to be 1.4 % in 2020 for the operational phase [1]. Existing research demonstrates that optimization at the software level can lead to significant energy savings. For example, replacing specific JavaScript libraries in web applications can reduce energy consumption by up to 30 % [3]. Despite the potential for considerable energy savings, a large portion of software developers do not prioritize energy efficiency during development, with only about 10% attempting to measure energy consumption [4]. We propose the creation of energy labels for software, inspired by the EU energy labels for household products. These labels are intended to provide relevant, comparable, and accurate information on energy efficiency, thereby enabling users and developers to make informed decisions. The EU labels have been recognized as “one of the most cost-effective ways to enhance security of energy supply...” [5]. Our preliminary study on WordPress plugins – software modules – indicates that (1) it is feasible to assess individual software modules independently when investigating the energy consumption of web-based software, (2) it is possible to measure and determine their impact on the energy consumption of the entire system, and (3) this approach allows for the first comparison of energy consumption among individual modules offering similar functionalities [6]. The study suggests that it is feasible to develop energy labels for software in the future, providing relevant, comparable, and accurate information on energy efficiency. We envision creating energy labels for functionally



equivalent modules and, in the long run, standardizing the process for creating such labels to be applicable across different levels within the software stack, effectively driving reductions in the environmental impact of software.

### References

- 1 Jens Malmodin, Nina Lövehagen, Pernilla Bergmark, Dag Lundén, “ICT sector electricity consumption and greenhouse gas emissions – 2020 outcome”, *Telecommunications Policy*, Volume 48, Issue 3, 2024, <https://doi.org/10.1016/j.telpol.2023.102701>.
- 2 C. Freitag, M. Berners-Lee, K. Widdicks, B. Knowles, G. Blair, and A. Friday, ”The climate impact of ICT: A review of estimates, trends, and regulations,” *December*, 2020.
- 3 A. P. Miettinen and J. K. Nurminen, “Analysis of the Energy Consumption of JavaScript Based Mobile Web Applications,” in *International Conference on Green IT*, 2010, doi: 10.1007/978-3-642-16644-0\_12.
- 4 C. Pang, A. Hindle, B. Adams, and A. E. Hassan, “What do programmers know about the energy consumption of software?,” *PeerJ PrePrints*, vol. 3, p. e1094, 2015.
- 5 Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee, and the Committee of the Regions, “Energy Efficiency Plan 2011,” *Issue 1*, 2011.
- 6 E. B. Unna-Lindhard and N. Müllenborn, “Energy consumption of WordPress plugins,” 2022.

## 3.8 10 years of GCC from an energy perspective

*Michael Kirkedal Thomsen (University of Copenhagen, DK), Maja Hanne Kirkeby (Roskilde University, DK)*

**Joint work of** Michael Kirkedal Thomsen, Ken Friis Larsen, Maja Hanne Kirkeby, Lars-Bo Husted Vadgaard  
**License** © Creative Commons BY 4.0 International license  
 © Michael Kirkedal Thomsen and Maja Hanne Kirkeby

In this work, we present an investigation into 10 years of GCC versions from an energy perspective. Specifically, we employ Intel’s Running Average Power Limit (RAPL) technology to compare the energy consumption of different versions of GCC when they compile a relevant subset of the SPEC benchmark. Initial results show no significant improvement and may suggest that older GCC versions are more efficient at compiling. However, more measurements are needed to obtain a clear statistical result.

## 3.9 Trace & Tag: Towards Carbon-Aware Memory Placement

*Sven Köhler (Hasso-Plattner-Institut, Universität Potsdam, DE) and Timo Hönig (Ruhr-Universität Bochum, DE)*

**License** © Creative Commons BY 4.0 International license  
 © Sven Köhler and Timo Hönig  
**Joint work of** Sven Köhler, Wenzel Lukas, Wenzel, Timo Hönig, Andreas Polze

Memory subsystems have long been overlooked consumers of operational energy in computing systems. Novel memory technologies, like NVRAM, further introduce wear-and-tear behaviour, making embodied carbon a factor in memory placement decisions.

With cMemento we aim to supply system software with knowledge about workload behaviour and hardware resources alike for more carbon-aware steering of allocations.

One key requirement is an understanding of a workload’s sub-page memory access patterns. Hence, we propose Heimdall, a novel memory device that occupies part of the physical address space. It is backed by other DRAM in the system but can snoop on and redirect requests on the memory bus. Our prototype runs on an AMD Zynq 7000 SOC, with an ARM Cortex-A and tightly integrated Xilinx FPGA.

Heimdallr can simulate heterogenous memory devices, using the FPGA’s block RAM. We present an architecture and operating system interface to trace and process access patterns on the memory bus, without impeding the overall bandwidth. Although we add about 90% latency while the tracing is enabled, an early evaluation shows that we can use the gained knowledge to move the most promising allocations to the block RAM – saving an average of 11% in operational energy permanently.

This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Individual Research Grant 502228341 (Memento) as part of the Priority Programme on Disruptive Memory Technologies (SPP 2377).

### 3.10 Working towards a standardized reporting and assessment of the energy efficiency of ML models training and inference

*Silverio Martínez-Fernández (UPC Barcelona Tech, ES)*

**License** © Creative Commons BY 4.0 International license  
© Silverio Martínez-Fernández

**Main reference** Silverio Martínez-Fernández, Xavier Franch, Francisco Durán: Towards green AI-based software systems: an architecture-centric approach (GAISSA). SEAA 2023: 432-439

**URL** <https://doi.org/10.1109/SEAA60479.2023.00071>

Software-related CO2 emissions from the information and communications technology sector currently account for 2.1%–3.9% of global emissions. With the latest advancements in Machine Learning (ML) systems, this percentage of global emissions is estimated to increase. This raises the question: how to create more awareness on the sustainability impact of training and inference of ML models? In this talk, I navigate through the capabilities of the GAISSALabel tool [1] from the GAISSA project [2]. The tool shows how to report green ML metrics, to measure and assess the energy efficiency in ML systems, and shows recommendations on how to reduce their carbon emissions (e.g., reducing ML model size, using lightweight architectures, early stopping, optimization tactics as quantization and pruning, energy-efficient ML serving, and when to deploy in cloud or edge). Efforts to standardize the reporting of ML carbon emissions are needed (e.g., ISO 20226, SCI from GSF). Besides reporting, the aggregation of the results of empirical studies from the green software engineering community, in order to create theories with consolidated and quantified software tactics to reduce emissions, is key.

#### References

- 1 Pau Duran, Joel Castaño, Cristina Gómez, Silverio Martínez-Fernández: GAISSALabel: A Tool for Energy Labeling of ML Models. SIGSOFT FSE Companion 2024: 622-626
- 2 Silverio Martínez-Fernández, Xavier Franch, Francisco Durán: Towards green AI-based software systems: an architecture-centric approach (GAISSA). SEAA 2023: 432-439

### 3.11 CXL Memory Expansion: A Closer Look on Actual Platforms

*Vinicius Petrucci (Micron Technology – Austin, US)*


License  Creative Commons BY 4.0 International license  
© Vinicius Petrucci

The proliferation of data-intensive workloads has led to a significant increase in the volume of data that computing systems need to handle. The growing significance of memory as a bottleneck has resulted in increased total cost of ownership (TCO) for servers. Existing methods to enhance bandwidth involve raising the data rate; while scaling up capacity is achieved through techniques like 2DPC or increasing DRAM density, each of which are coming at the expense of the other. An alternative approach is to scale out by adding more CPUs or through 3D stacking memory packaging technology, but these approaches come at the expense of much higher TCO.

In this talk I will discuss the potential use cases of Compute Express Link (CXL) technology. CXL is an open industry standard interconnect offering coherency and memory semantics using high bandwidth, low latency connectivity between processors, accelerators, memory, storage, and other IO devices. It helps address the above challenges by scaling up through increased bandwidth and capacity per core in the system. CXL is gaining faster adoption because it introduces load/store semantics to PCI Express (PCIe) physical layer, enabling expansion of both capacity and bandwidth at access latency comparable to remote non uniform memory access (NUMA) node DRAM memory. This addresses the CPU pin count challenge and avoids the TCO associated with adding new servers and processors just to get additional bandwidth and capacity.

### 3.12 Reducing Storage Emissions based on Deduplication

*Max Plauth (UltiHash – Berlin, DE)*

License  Creative Commons BY 4.0 International license  
© Max Plauth  
URL <https://ultihash.io/whitepaper>

Until recently, the emissions caused by storage resources were largely disregarded, with both academic and industry efforts having largely focused on emissions attributed to compute resources. To keep up with the excessive demand for storage resources across disciplines and industries however, all available opportunities need to be explored to enable data volumes to grow more sustainably. While research efforts on reducing storage emissions are still in an early stage, they largely focus on three general strategies to reduce storage emissions: By lowering and/or shifting power, by increasing the density of storage devices, or by extending the lifetime of storage hardware [1].

With the development of the UltiHash object storage, we are proposing an additional strategy to reduce storage emissions. We propose that using fine-grained deduplication, we can reduce the number of storage devices required to store data. With this approach, we are accepting slightly increased operational emissions caused by spending CPU time once to reduce both operational and embodied emissions of storage devices over the retention time of the stored data.

Our approach differs from existing deduplication mechanisms in two distinctive ways: first, using a variable fragment size enables us to detect more fine grained occurrences of redundancies. Second, instead of operating on a limited scope such as a single file system or a single file server, the scope of deduplication spans across a distributed storage cluster, aiming even towards federations of storage clusters.


While deduplication rates of our approach highly depend on the properties of the dataset to be deduplicated, initial tests have yielded data deduplication rates of up to 70% for uncompressed datasets, and even deduplication rates  $\geq 10\%$  for compressed media files.

## References

- 1 Sara McAllister, Fiodar Kazhamiaka, Daniel S. Berger, Rodrigo Fonseca, Kali Frost, Aaron Ogus, Maneesh Sah, Ricardo Bianchini, George Amvrosiadis, Nathan Beckman, Gregory R. Ganger, “A call for research on storage emissions”, *3rd Workshop on Sustainable Computer Systems (HotCarbon)*, 2024

### 3.13 The Cost of a Secure OS

*Fabian Rauscher (TU Graz, AT), Daniel Gruss (TU Graz, AT), and Timo Hönig (Ruhr-Universität Bochum, DE)*

**License**  Creative Commons BY 4.0 International license  
© Fabian Rauscher, Daniel Gruss, and Timo Hönig

In the security community it is commonly assumed that performance and energy are correlated and it is sufficient to just measure performance overheads to assess the overall costs of a security measure. We investigated whether this is true for the Linux kernel and found that in a significant number of cases energy and performance costs are actually not correlated. We conclude that to assess the energy costs of security, it is currently unavoidable to perform actual energy measurements.

### 3.14 Green AI: The more does not (always) lead to the better!

*June Sallou (TU Delft, NL)*

**License**  Creative Commons BY 4.0 International license  
© June Sallou

**Joint work of** Nienke Nijkamp, June Sallou, Niels van der Heijden, Luís Cruz  
**Main reference** Nienke Nijkamp, June Sallou, Niels van der Heijden, Luís Cruz: “Green AI in Action: Strategic Model Selection for Ensembles in Production”, in Proc. of the 1st ACM International Conference on AI-Powered Software, Alware 2024, Porto de Galinhas, Brazil, July 15-16, 2024, ACM, 2024.

**URL** <https://doi.org/10.1145/3664646.3664763>

The integration of Artificial Intelligence (AI) into software systems has significantly enhanced their functional capabilities, albeit at the cost of increased energy demands. Ensemble learning, which combines predictions from multiple models to derive a single output, exacerbates this issue by intensifying cumulative energy consumption. We address these trade-off challenges through an industrial case study of an AI-enabled system using ensemble learning for document information extraction. Our findings demonstrate that our approach of selecting a subset of models by optimising the GreenQuotientIndex, which accounts for the trade-off between accuracy and resource usage, can lead to resource savings of up to 99%. Additionally, this approach reveals unexpected gaps in system optimisation related to

accuracy. Selecting a subset of AI models can even increase accuracy by up to a factor of 2.7. These insights underscore the necessity of defining sustainability as a critical requirement in software development, rather than defaulting to a higher number of models without careful consideration.

### 3.15 Executable explanations of control software for wind turbines

*Sibylle Schupp (TU Hamburg, DE)*

License  Creative Commons BY 4.0 International license  
© Sibylle Schupp

Cyber-physical energy systems (CPES) are digitally controlled systems in which the degree of heterogeneity and complexity has become so high that their particular behavior is no longer necessarily transparent – not even to developers or users with technical background. One way of providing more transparency is by giving an explanation of the particular behavior of interest; and for that, the subsystem in question is in the best positions, as it best could explain itself. So that a self-explanation is operationizable by its addressee, it needs to be geared towards that specific addressee. Also, it has to be provided timely and in a context-sensitive fashion, and in particular at a level of abstraction different from the behavior that should be explained.

Consider as example a wind turbine controller. Characteristic for controllers is the feedback loop, in which the input depends non-trivially on the output as well as the interaction with other control systems, e.g., the controller for the collective pitch angle interacts with the controller for the yaw. An addressee of an explanation of that subsystem might be a maintainer or an optimizer or fault searcher as well as an operator of a client system. Now assume the addressee observes an unexpected output value and thus is in need of an explanation. So that the addressee can properly respond, it must not consider one such parameter just in isolation. On the other hand, it cannot be expected to understand the full control-theoretic details of its dependences. Instead, the addressee needs a simplified, abstract explanation of the system behavior, here: of the turbine controller.

For reducing the complexity of control software while keeping mutual dependencies of the feedback loop, we exploit that the differential equations of the control problem can be piecewise approximated by simple (linear) functions and that in an online setting those simplifying “control patterns” can be swapped out when outdated. The overall soundness of the approximation then depends on the “control patterns”, the choice of time windows, and the safe and timely swapping of patterns.

The talk motivates the abstract explanation pattern introduced by the CAUSE (Concepts and Algorithms for—and Usage of—Self-Explaining Systems) Research Training Group and, for wind turbines, illustrates this explanation pattern on its control software where explanations are executable networks of timed automata.

### 3.16 How to quantify the energy consumption of applications

*Tamar Eilam (BM TJ Watson Research Center – Yorktown Heights, US)*

License © Creative Commons BY 4.0 International license  
© Tamar Eilam

Improving the efficiency of software means running more computing using less resources, and energy. However, in order to achieve this goal we need to first be able to quantify the energy consumption of applications. Quantification is hard for multiple reasons: in realistic environments multiple applications run on shared infrastructure. First, there is no way to measure the energy for individual applications, hence we need to approximate it. Second, in many cases there is no access to the actual on-line power reads of the machine (e.g., in cloud environment). Third, applications behave differently on different hardware and configuration leading to combinatorial explosion. In this talk I will describe the approach we took with our open source tool Kepler to address these challenges.

#### References

- 1 <https://sustainable-computing.io>

### 3.17 Energy-efficient systems through non-volatile memory

*Manuel Vögele (Ruhr-Universität Bochum, DE), Timo Hönig (Ruhr-Universität Bochum, DE), and Wolfgang Schröder-Preikschat (Universität Erlangen-Nürnberg, DE)*

License © Creative Commons BY 4.0 International license  
© Manuel Vögele, Timo Hönig, and Wolfgang Schröder-Preikschat

**Main reference** Jonas Rabenstein, Dustin T. Nguyen, Oliver Giersch, Christian Eichler, Timo Hönig, Jörg Nolte, Wolfgang Schröder-Preikschat: “Back to the Core-Memory Age: Running Operating Systems in NVRAM only”, in Proc. of the Architecture of Computing Systems – 36th International Conference, ARCS 2023, Athens, Greece, June 13-15, 2023, Proceedings, Lecture Notes in Computer Science, Vol. 13949, pp. 153–167, Springer, 2023.

**URL** [https://doi.org/10.1007/978-3-031-42785-5\\_11](https://doi.org/10.1007/978-3-031-42785-5_11)

Non-volatile RAM (NVRAM) is fast, byte-addressable, non-volatile memory that can replace traditional DRAM in systems. Doing so improves the non-functional properties of a system, including lower energy usage, faster storage and higher resiliency. This talk focuses on the energy usage aspect.

There are several opportunities to save energy by using NVRAM. First, opposed to DRAM, NVRAM does not need to be refreshed, leading to low idle power consumption. Second, by getting rid of all volatile memory in our systems, they become able to rapidly power off and resume their operation. This can be used to shut off machines during even short periods of idle time, while staying able to quickly respond to requests. Third, a lot of persistency mechanisms in operating systems are no longer necessary, as NVRAM provides that persistence for free. This means we can declutter our operating systems, getting rid of energy intensive I/O operations in the process. However, there are still challenges to overcome to realize those savings. Today’s systems are built with DRAM in mind. Booting into NVRAM is challenging and slow firmware hinders rapid suspend/resume mechanisms. Finally, introducing a new class of memory into a system requires the operating system to perform placement decisions to ensure the heterogeneous memory resources are used most efficiently.

This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Individual Research Grant 465958100 (NEON).

## 4 Working groups

### 4.1 Hackathon: CPU Undervolting

*Jonas Juffinger (TU Graz, AT)*

**License** © Creative Commons BY 4.0 International license  
© Jonas Juffinger

**Joint work of** Jonas Juffinger, Stepan Kalinin, Daniel Gruss, Frank Mueller

**Main reference** Jonas Juffinger, Stepan Kalinin, Daniel Gruss, Frank Mueller: “SUIT: Secure Undervolting with Instruction Traps”, in Proc. of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, ASPLOS 2024, La Jolla, CA, USA, 27 April 2024- 1 May 2024, pp. 1128–1145, ACM, 2024.

**URL** <https://doi.org/10.1145/3620665.3640373>

CPU undervolting, running a CPU with a lower voltage than specified by the manufacturer, can save up to 20% of CPU power consumption. However, CPU undervolting also negatively impacts systems’ reliability and security, which is not tolerable in most use cases. In our work, SUIT, we showed how CPU undervolting could be made secure to allow the usage of it on a wide scale. In this hackathon, we prepared some experiments to measure the energy and performance impact of CPU undervolting. We measured up to 20% reduction in energy consumption without any performance impact. Another experiment could be used to play with the undervolting offset and experience the possible reliability issues. Finally, we prepared experiments to show the possible security issues of CPU undervolting and how it can be used steal cryptographic keys. About 30 people participated in the hackathon.

### 4.2 Hackathon: Carbond

*Robin Ohs (Universität des Saarlandes – Saarbrücken, DE), Benedict Herzog, Gregory Stock, Henry Janson, Timo Hönig (Ruhr-Universität Bochum, DE), Luis Gerhorst, and Andreas Schmidt (Universität des Saarlandes – Saarbrücken, DE)*

**License** © Creative Commons BY 4.0 International license  
© Robin Ohs, Benedict Herzog, Gregory Stock, Henry Janson, Timo Hönig, Luis Gerhorst, and Andreas Schmidt

**Main reference** Andreas Schmidt, Gregory Stock, Robin Ohs, Luis Gerhorst, Benedict Herzog, Timo Hönig: “carbond: An Operating-System Daemon for Carbon Awareness”, in Proc. of the 2nd Workshop on Sustainable Computer Systems, HotCarbon 2023, Boston, MA, USA, 9 July 2023, pp. 2:1–2:6, ACM, 2023.

**URL** <https://doi.org/10.1145/3604930.3605707>

The carbon intensity of the grid is influenced by the mix of carbon emissions generated by various electricity production methods. To utilize this intensity metric for carbon-aware computing, a common approach is to query data providers such as “Electricity Maps” or “WattTime” to obtain the average grid intensity of the desired location. Although this is an effective initial approach, it underestimates the actual carbon intensity of the utilized energy due to inefficiencies within the system itself coming from the power supply unit (AC to DC conversion) and, if battery powered, from the storage of the energy. The presented tool Carbond is an operating system daemon that aims to calculate a system intensity that approximates more accurately the intensity of the energy you are currently using. The system intensity is calculated by considering the above-mentioned inefficiencies and an estimated embodied intensity of the battery. This hackathon explains the principles and origins of

these inefficiencies and showcases the practical use of Carbond. For this, the participants measure the required energy of an energy-intensive workload and then connect this energy demand to actual emissions.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) project numbers 502228341 (“Memento”), 465958100 (“NEON”), and 389792660 as part of TRR 248 – CPEC4.



## Participants

- Gustavo Alonso  
ETH Zürich, CH
- Antonio Carlos Schneider  
Beck Filho  
Federal University of Rio Grande  
do Sul, BR
- Pierre Bennorth  
CopenCloud – Karlslunde, DK
- Liliana Cucu-Grosjean  
INRIA – Paris, FR & StatInf –  
Paris, FR
- Ada Diaconescu  
Telecom Paris, FR
- Kerstin I. Eder  
University of Bristol, GB
- Tamar Eilam  
IBM TJ Watson Research Center  
– Yorktown Heights, US
- Michael Engel  
Universität Bamberg, DE
- João Paulo Fernandes  
New York University –  
Abu Dhabi, AE
- Clemens Grelck  
Friedrich-Schiller-Universität  
Jena, DE
- Daniel Gruss  
TU Graz, AT
- Timo Hönig  
Ruhr-Universität Bochum, DE
- Geerd-Dietger Hoffmann  
Green Coding Solution –  
Berlin, DE
- Romain Jacob  
ETH Zürich, CH
- Jonas Juffinger  
TU Graz, AT
- Fiodar Kazhamiaka  
Microsoft – Redmond, US
- Maja Hanne Kirkeby  
Roskilde University, DK
- Michael Kirkedal Thomsen  
University of Copenhagen, DK
- Sven Köhler  
Hasso-Plattner-Institut,  
Universität Potsdam, DE
- David Kohnstamm  
Leafcloud – Amsterdam, NL
- Julia Lawall  
INRIA – Paris, FR
- Silverio Martínez-Fernández  
UPC Barcelona Tech, ES
- Daniel Mosse  
University of Pittsburgh, US
- Frank Mueller  
North Carolina State University –  
Raleigh, US
- Jukka K. Nurminen  
University of Helsinki, FI
- Robin Ohs  
Universität des Saarlandes –  
Saarbrücken, DE
- Vinicius Petrucci  
Micron Technology – Austin, US
- Max Plauth  
UltiHash – Berlin, DE
- Fabian Rauscher  
TU Graz, AT
- June Sallou  
TU Delft, NL
- Joao Saraiva  
University of Minho, PT
- Andreas Schmidt  
Universität des Saarlandes –  
Saarbrücken, DE
- Gunnar Schomaker  
Universität Paderborn, DE
- Wolfgang Schröder-Preikschat  
Universität Erlangen-  
Nürnberg, DE
- Sibylle Schupp  
TU Hamburg, DE
- Jennifer Switzer  
University of California –  
San Diego, US
- Manuel Vögele  
Ruhr-Universität Bochum, DE
- Samuel Xavier-de-Souza  
Federal University of Rio Grande  
do Norte, BR

