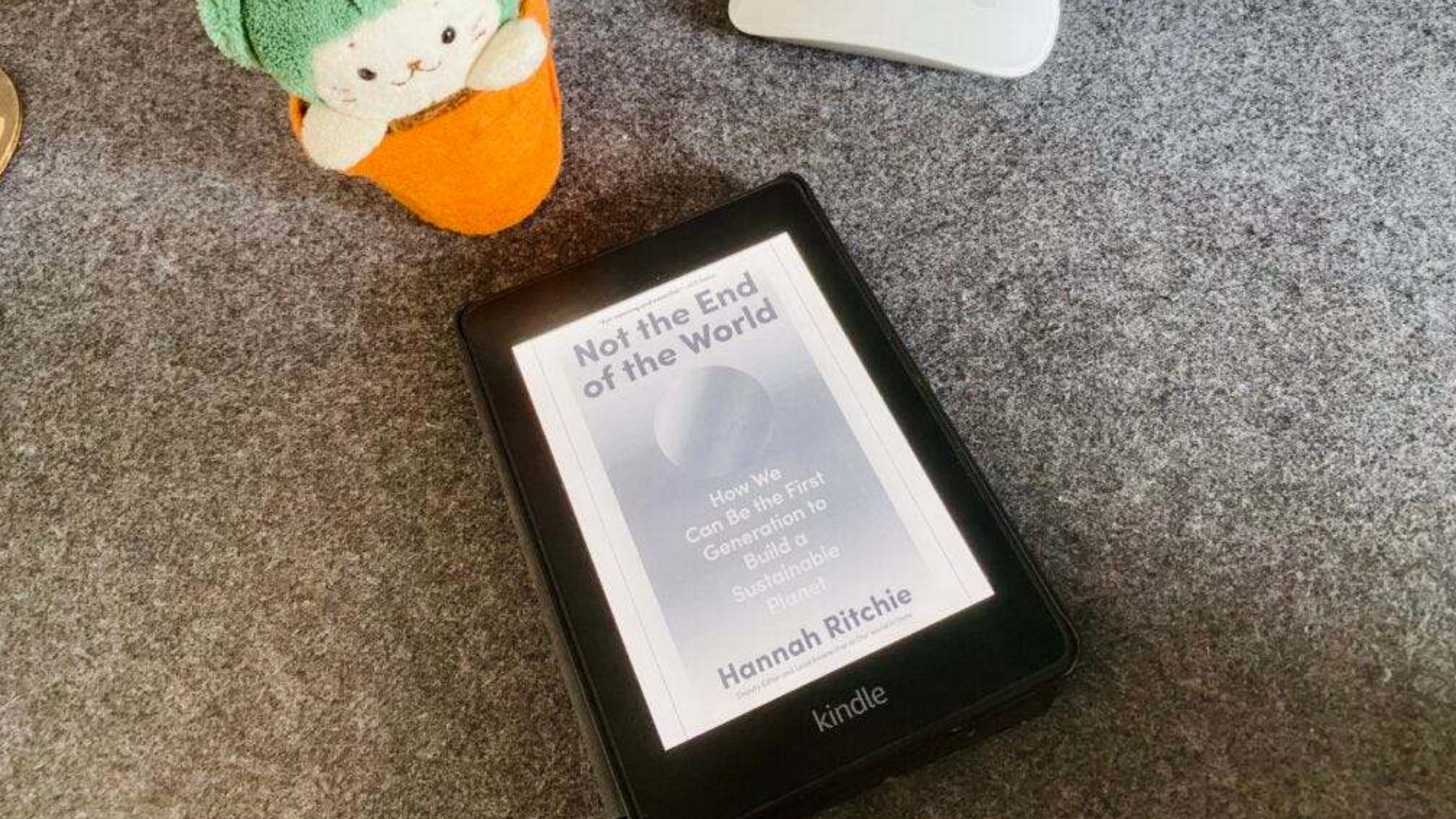
The State of the Green IT

At the beginning of 2024

Artur Skowroński

VIRTUSLAB


kindle


Not the End of the World
How We Can Be the First Generation to Build a Sustainable Planet
Hannah Ritchie
kindle


GLOBAL WARMING PROOF
IT'S COLD OUTSIDE


I'M WRITING FAKE PRESS RELEASES FOR IMAGINARY NEW GREEN ENERGY TECHNOLOGIES.
Dilbert.com DilbertCartoonist@gmail.com
SCIENTISTS SAY THAT BY 2040 YOU WILL BE ABLE TO POWER YOUR ENTIRE HOME WITH THE BREEZE FROM YOUR REFRIGERATOR DOOR.
3291-11 ©2011 Scott Adams, Inc./Dist. by UFS, Inc.
NOW HOW WILL I KNOW WHICH GREEN BREAK-THROUGHS ARE REAL?
SERIOUSLY? YOU THINK THERE ARE REAL ONES?

Do you recall GDPR ?

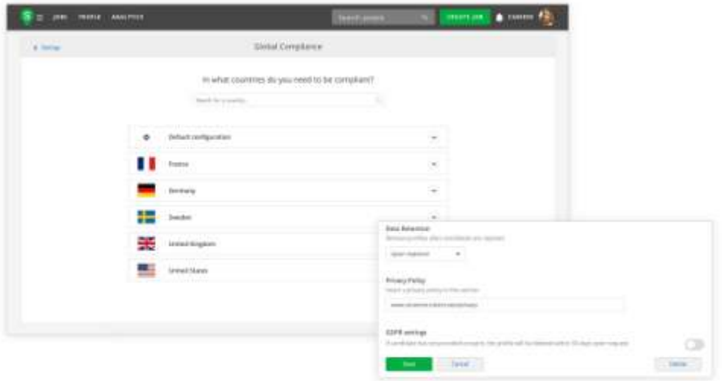

Cruella Breville
@Bad_Kate
DO YOU STILL WANT TO GET OUR EMAILS AFTER GDPR TAKES EFFECT?
Przetłumacz wpis

Globally Compliant Recruiting

Align your recruiting with national and international regulations while ensuring data privacy and local compliance everywhere your business operates. SmartRecruiters offers enterprise-grade functionality designed with global compliance & diversity in mind, allowing your organization to focus on hiring the best talent while ensuring safe & secure data processing and unbiased hiring practices.

Safeguard Candidate Data

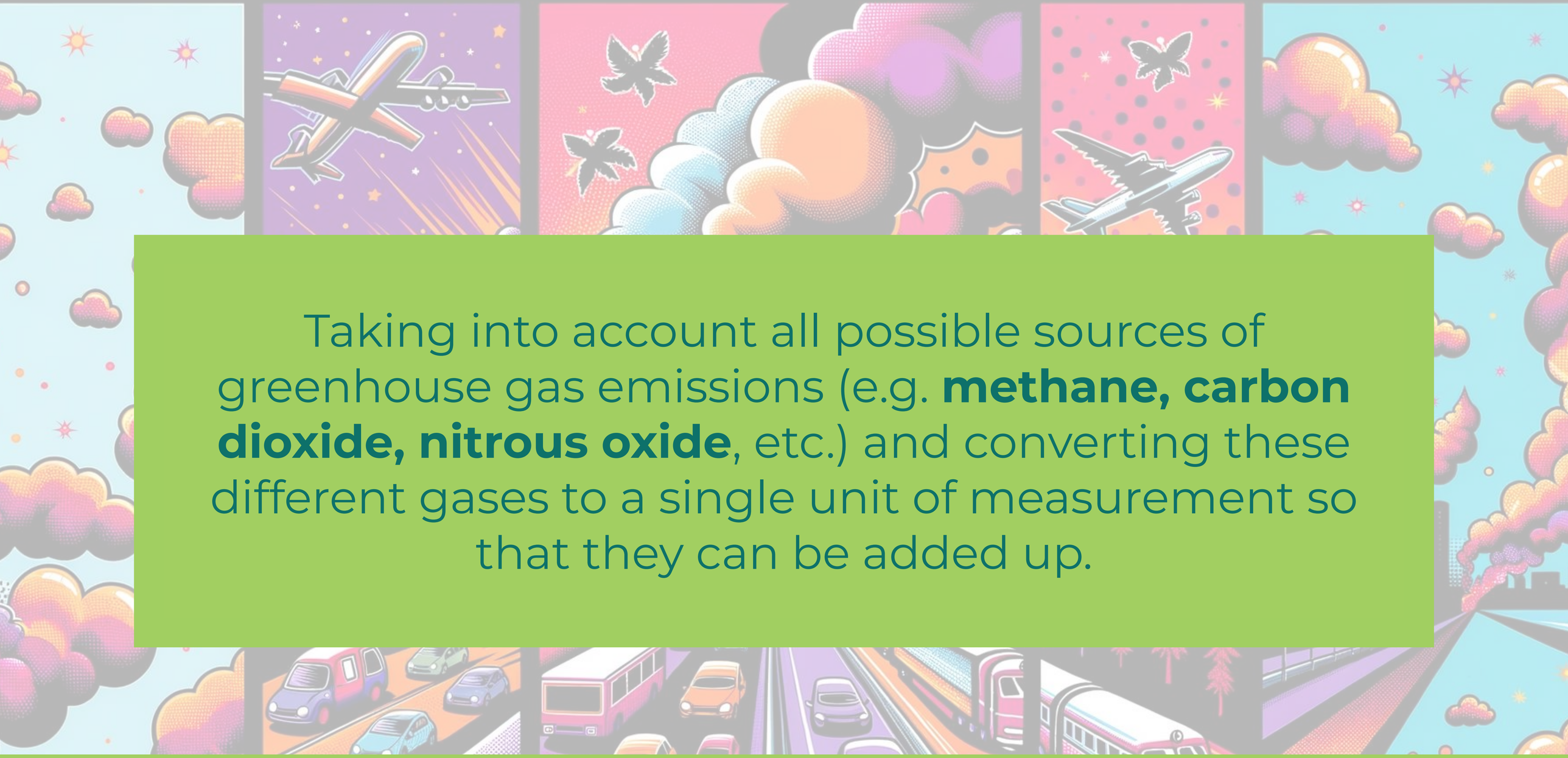
Handle candidate data responsibly, regardless of location, for every job you create. SmartRecruiters' compliance support allows you to narrow in on country-level data retention rules and region-specific legislation details, like the CCPA, to ensure your organization meets all local & international data privacy requirements.




OH YOU ARE A
"GDPR EXPERT"?
PLEASE TELL ME
HOW HIGH FINES ARE

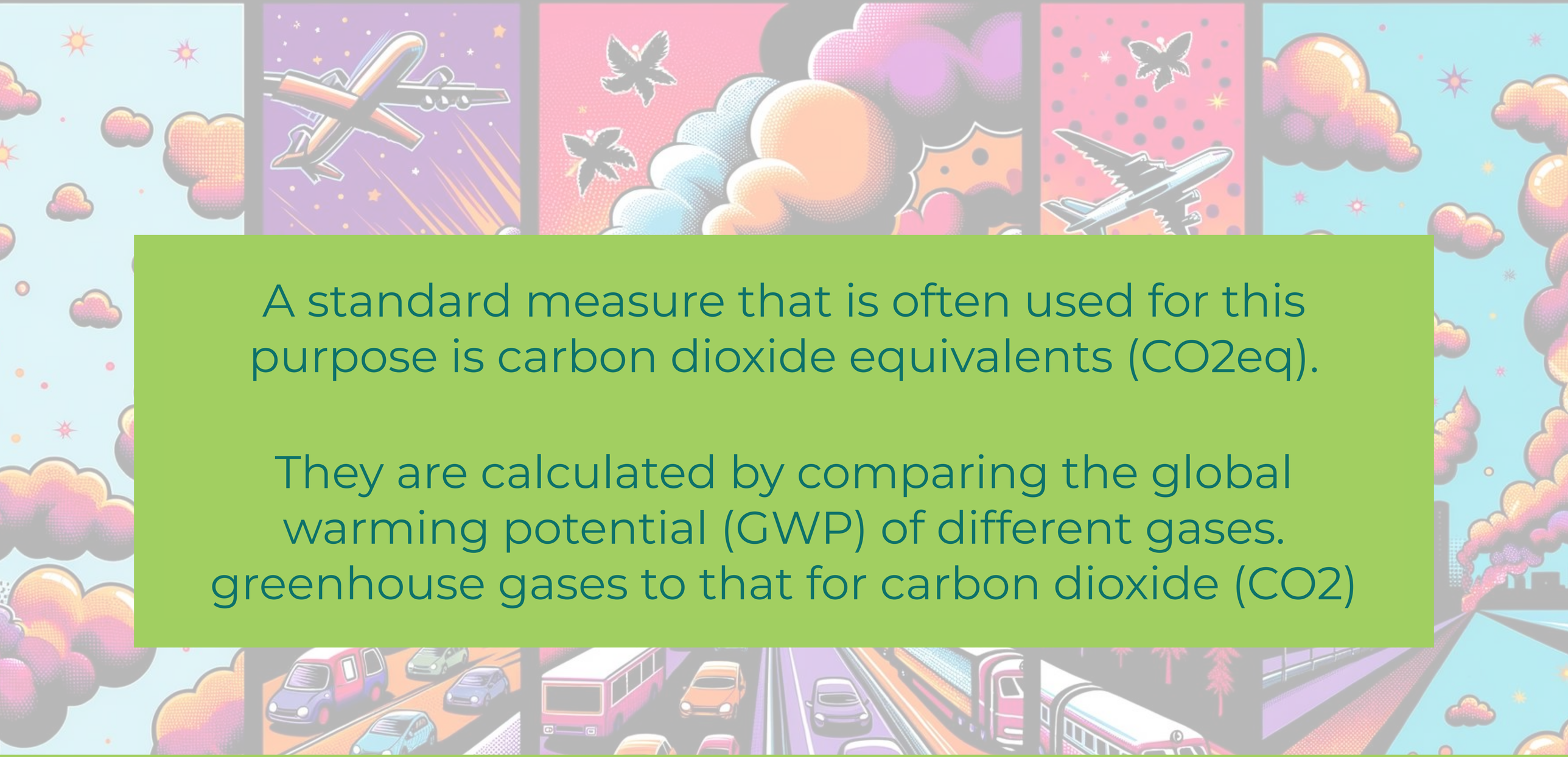
What are we talking about when we talk about Green Technology?

Carbon Footprint



Taking into account all possible sources of greenhouse gas emissions (e.g. **methane, carbon dioxide, nitrous oxide**, etc.) and converting these different gases to a single unit of measurement so that they can be added up.

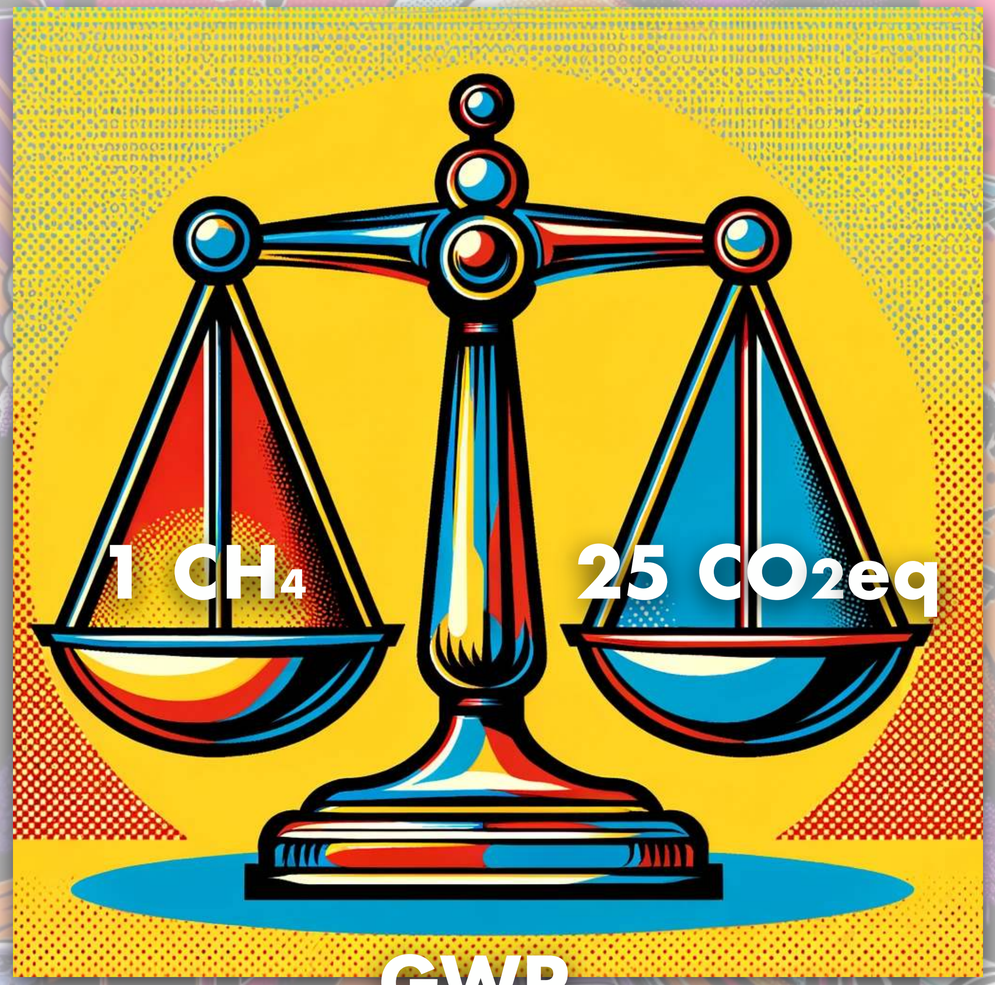
LCA Approach (Life Cycle Assessment)



A standard measure that is often used for this purpose is carbon dioxide equivalents (CO2eq).

They are calculated by comparing the global warming potential (GWP) of different gases. greenhouse gases to that for carbon dioxide (CO2)

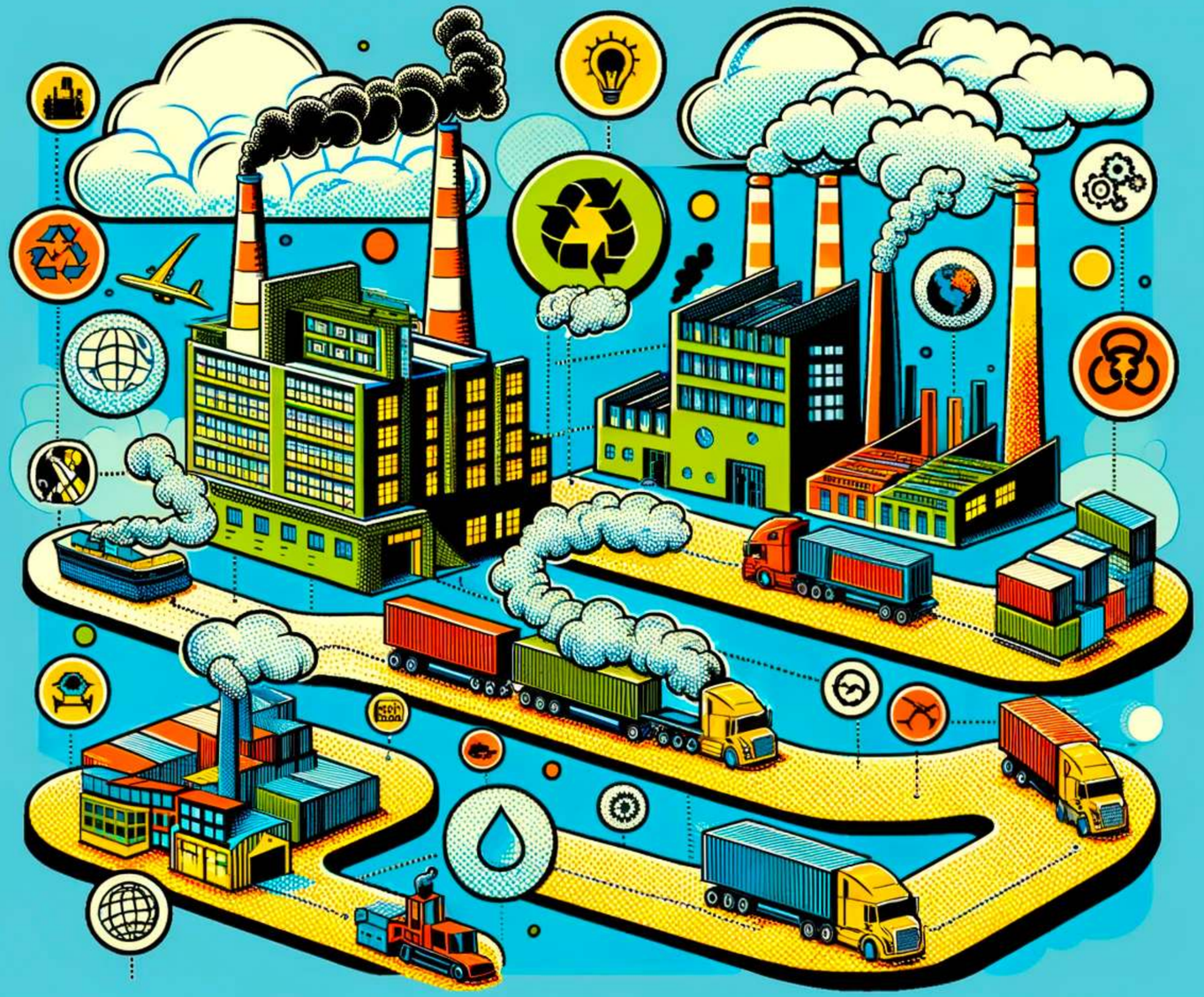
LCA Approach (Life Cycle Assessment)



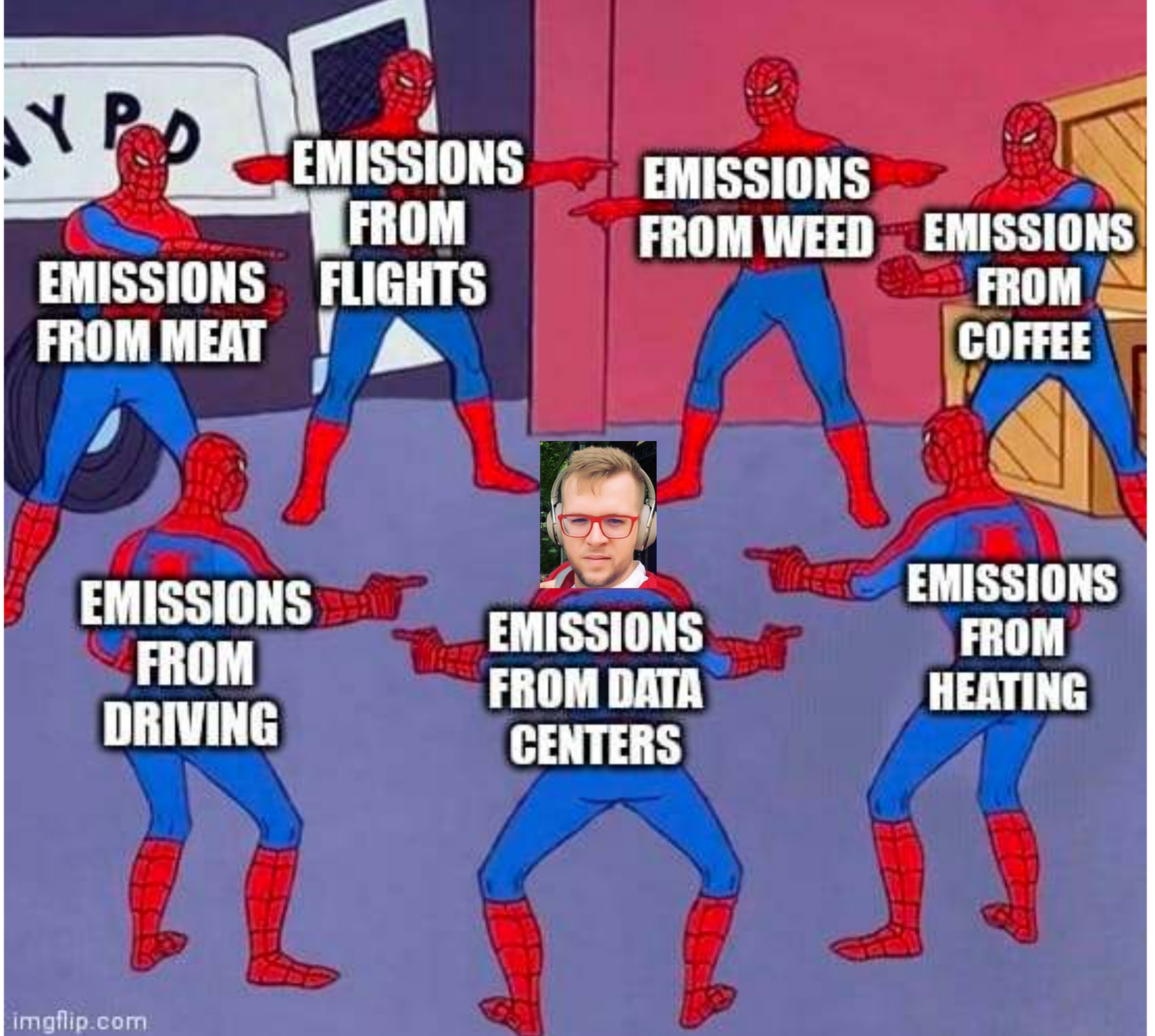
GWP

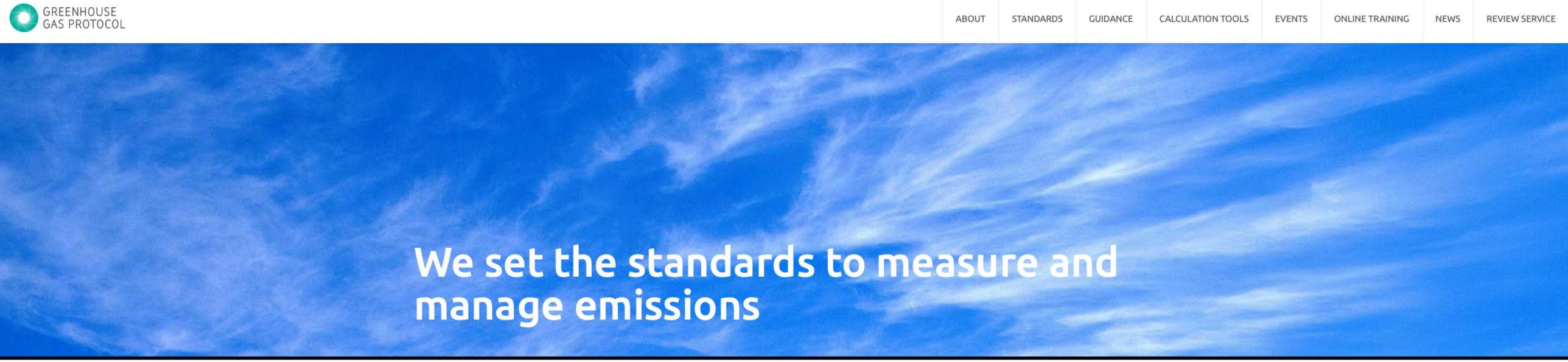
LCA Approach (Life Cycle Assessment)

All right, but which **emissions** should we really count?

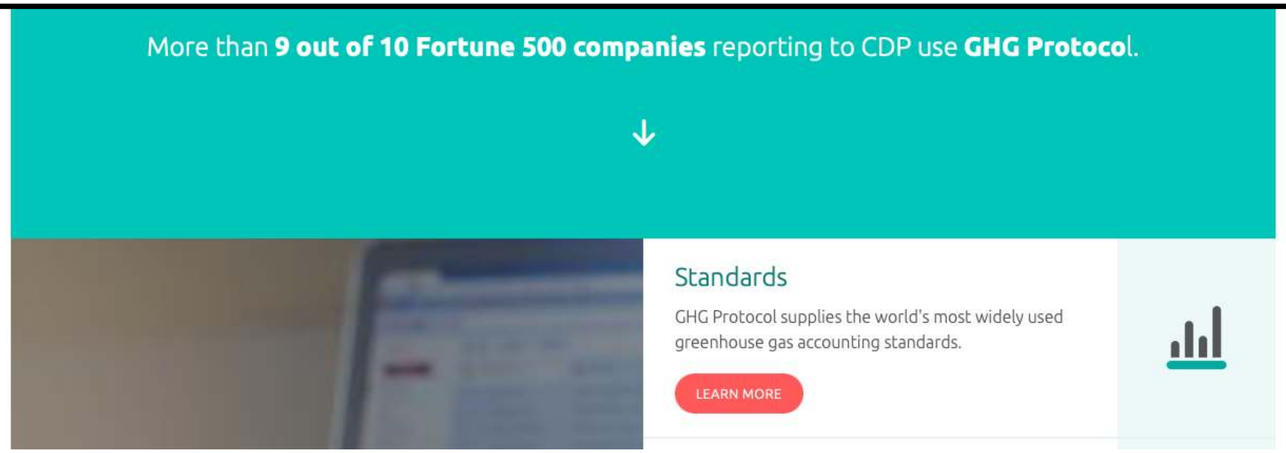


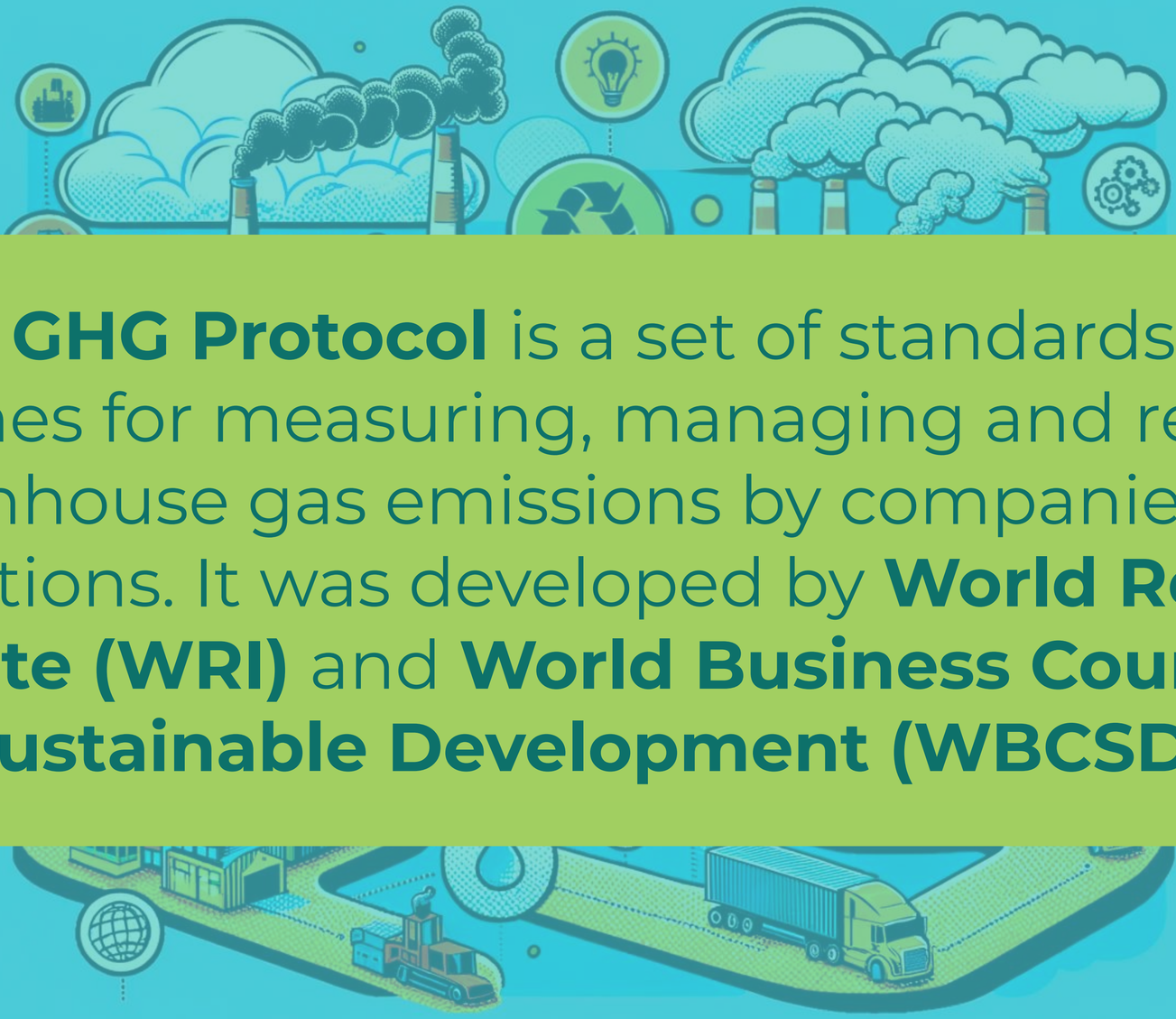
"Transferring" environmental impacts
AKA **creative accounting**


EMISSIONS FROM MEAT
EMISSIONS FROM FLIGHTS
EMISSIONS FROM WEED
EMISSIONS FROM COFFEE
EMISSIONS FROM DRIVING
EMISSIONS FROM DATA CENTERS
EMISSIONS FROM HEATING
imgflip.com



Greenhouse Gas Protocol





The GHG Protocol is a set of standards and guidelines for measuring, managing and reporting greenhouse gas emissions by companies and organisations. It was developed by **World Resources Institute (WRI)** and **World Business Council for Sustainable Development (WBCSD).**

Greenhouse Gas Protocol

These are greenhouse gas emissions that come directly from sources owned or controlled by the company.

Examples include exhaust fumes emitted from the chimneys of a factory it owns or the burning of fuel by company cars.

Scope 1 - Direct emissions

These include emissions from the production of electricity, heat or steam that the company buys and consumes.

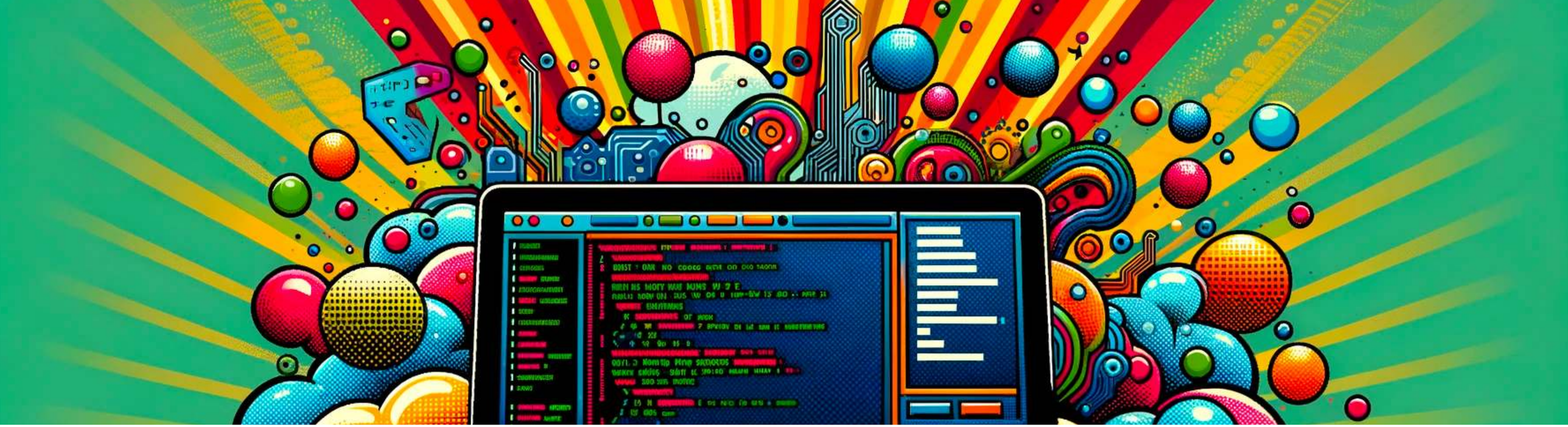
Although these emissions do not come directly from the company's operations, they are related to the company, as this energy is needed for its operations.

Scope 2 - Indirect Emissions

This is the broadest category and includes all other greenhouse gas emissions that result from a company's operations but come from sources the company does not own or control.

This could be, for example, the production of materials that the company buys from suppliers, or how customers use the products the company sells. It is also, for example, aircraft emissions.

Scope 3 - Other indirect emissions



How does Software fit into this?

GHG Scope	2	3
Private Cloud	Energia	Hardware
Public Cloud	-	Energia + Hardware
Hybrid Cloud	Energia + Hardware	Energia + Hardware
Front End		Energia + Hardware Użytkownika

Carbon Footprint

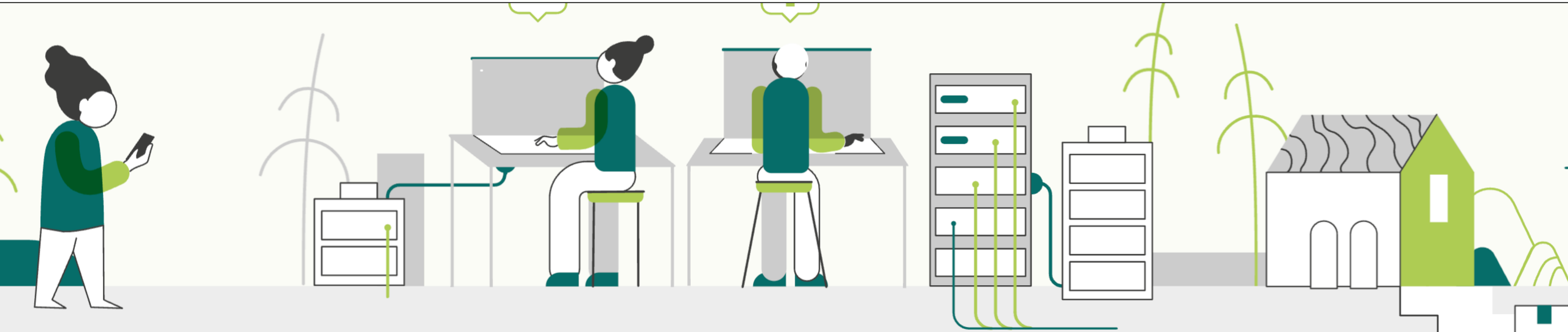
We are building a trusted ecosystem of people, standards, tooling and best practices for

GREEN SOFTWARE

Sign up to our newsletter...

Sign up

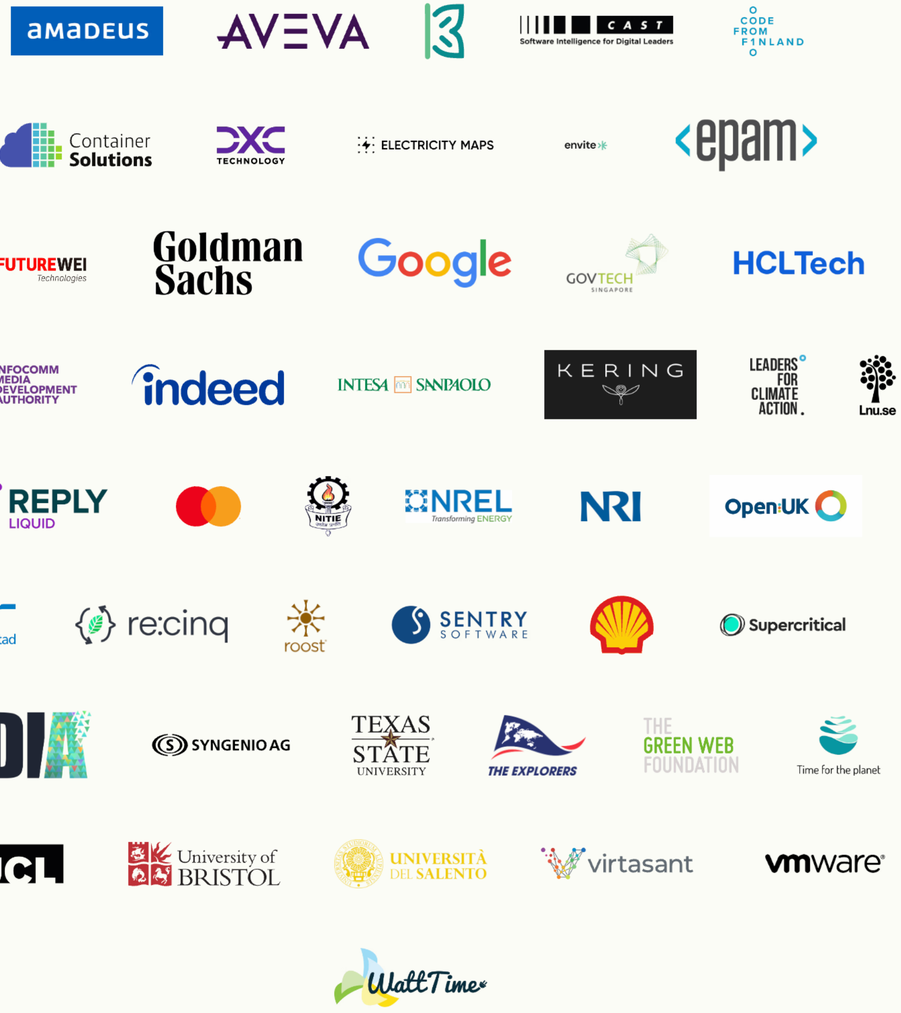
Green Software Foundation



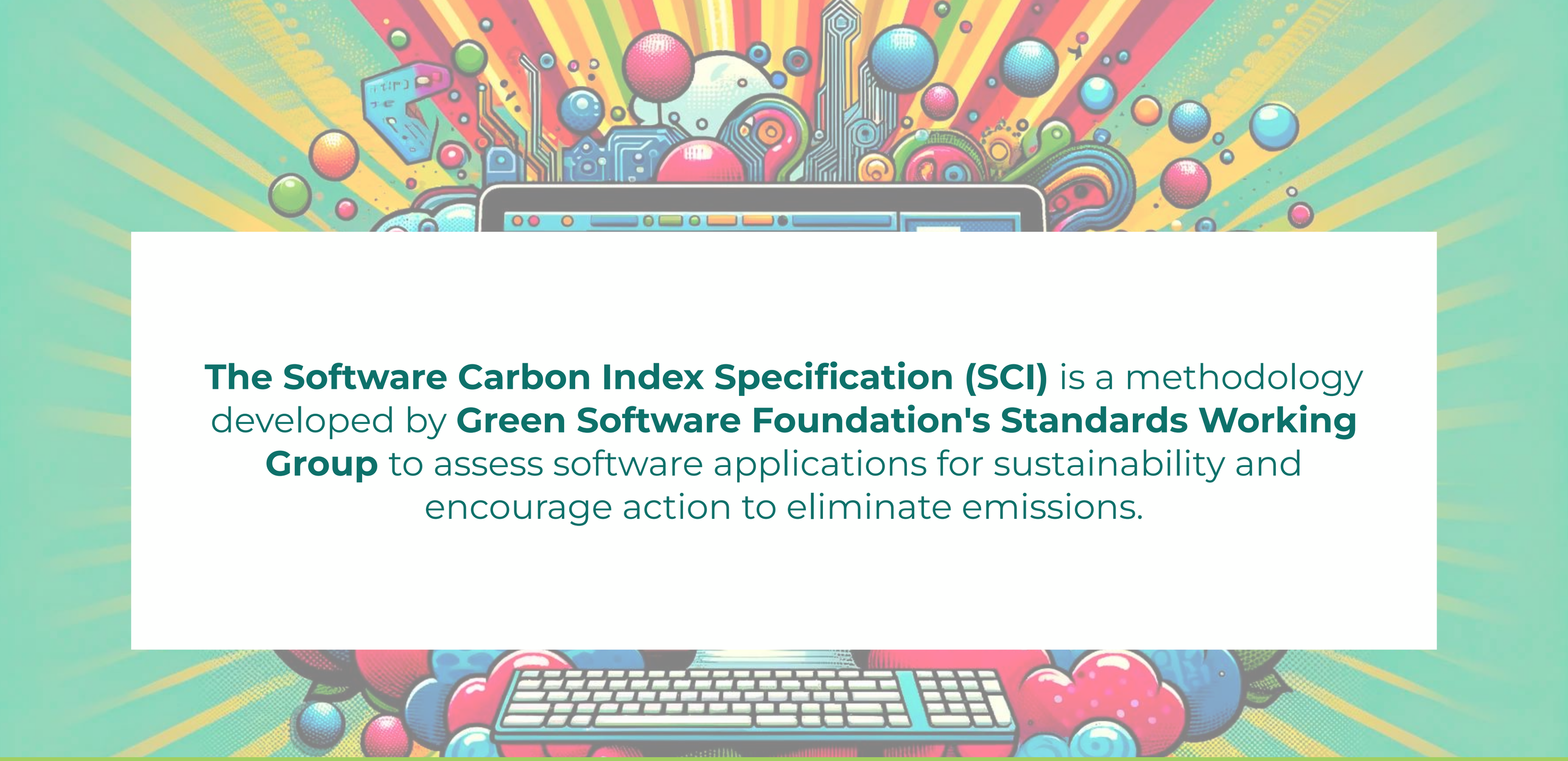
OUR STEERING MEMBERS

Green Software Foundation

OUR GENERAL MEMBERS



Green Software Foundation



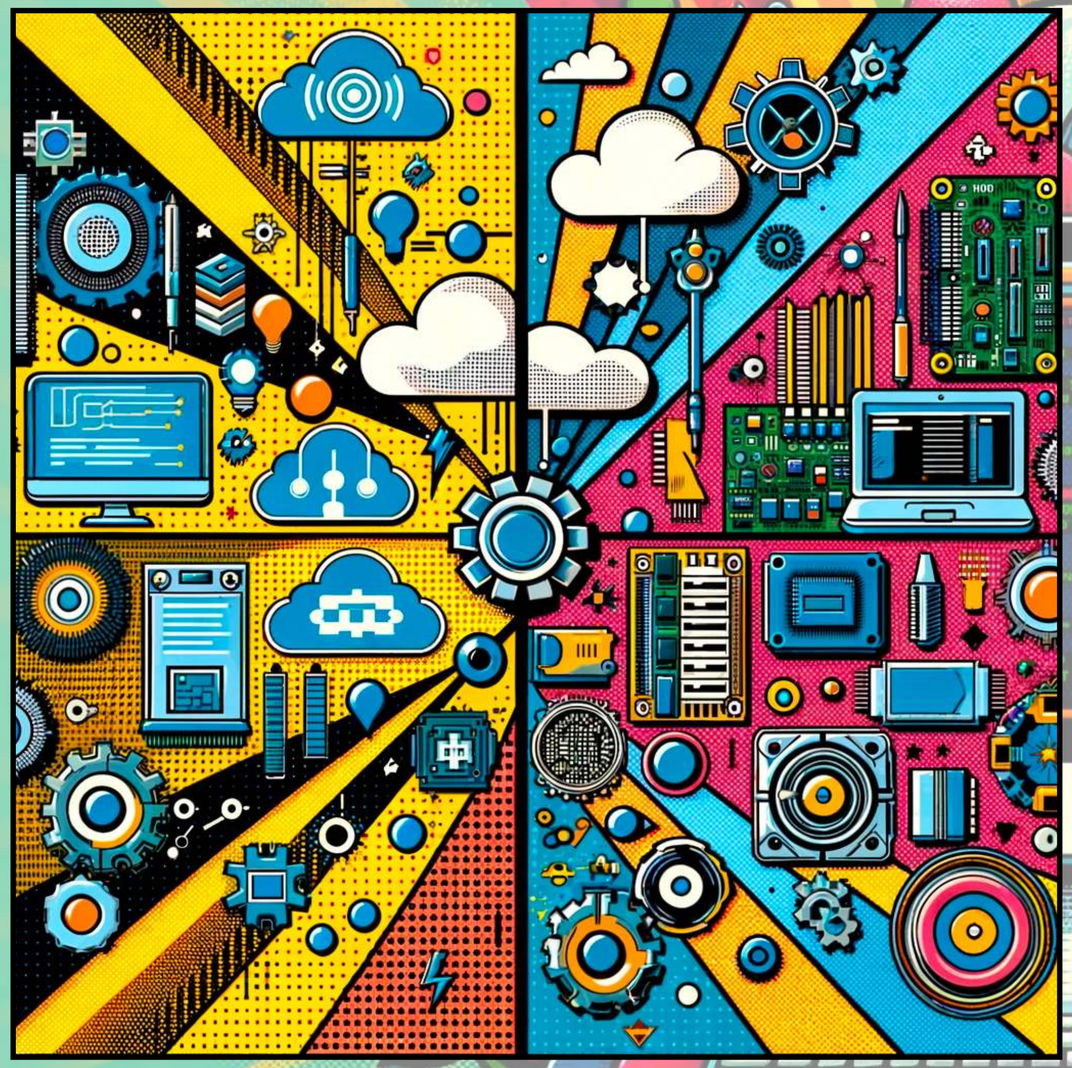
The Software Carbon Index Specification (SCI) is a methodology developed by **Green Software Foundation's Standards Working Group** to assess software applications for sustainability and encourage action to eliminate emissions.

Software Carbon Intensity specification

While the GHG protocol calculates total emissions, SCI is about calculating the emission factor.

SCI is more like measuring fuel consumption per mile, while the GHG protocol is more like the total carbon footprint of a car manufacturer.

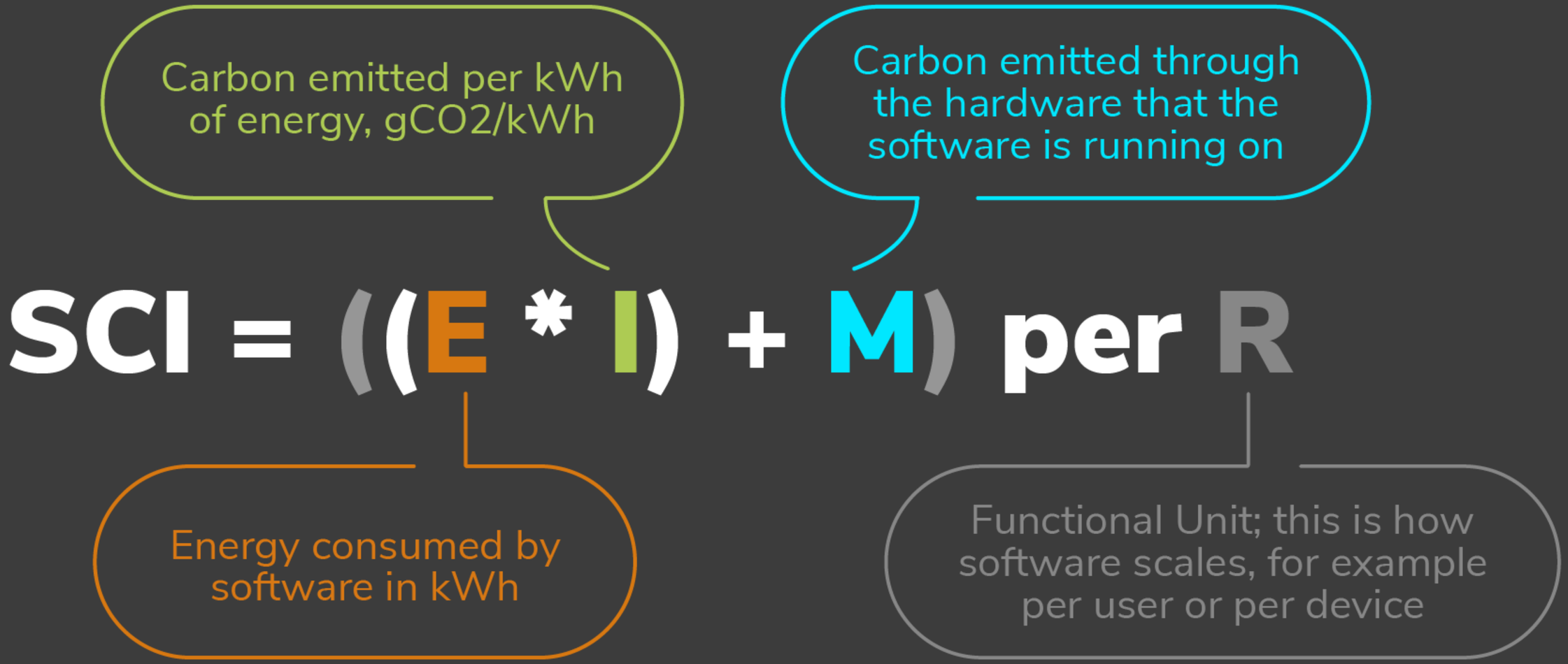
Software Carbon Intensity specification



Rather than dividing software carbon emissions into Scope 1-3, it divides them into **operational emissions** (carbon emissions from running the software) and **embedded emissions** (carbon emissions from the physical resources required to run the software).

It is therefore based on intensity rather than total quantity.

Software Carbon Intensity specification



Green Software Foundation

greensoftware.foundation

Web Sustainability Guidelines (WSG) 1.0

Draft Community Group Report 10 October 2023

Latest published version:

https://w3c.github.io/sustyweb/

Latest editor's draft:

https://w3c.github.io/sustyweb/

Editors:

Alexander Dawson

Tim Frick (Mightybytes)

Feedback:

GitHub w3c/sustyweb (pull requests, new issue, open issues)

Implementation:

Sustainable Web Design

Supplements:

At A Glance

Quick Reference

---

https://w3c.github.io/sustyweb/

- Minimize environments and reduce data transfer by compressing files, managing duplicate data, storing data according to visitor needs, optimizing caching, refreshing data only when the visitor needs it, and using CDN's and edge caching.

  In relation to **code minimization**, make sure you:

  - Focus on removing bottlenecks and reducing compute load on a visitor's device by removing rendering blockages, using code splitting, managing dependencies, taking advantage of native features, rigorously assessing third-party code / services and minifying everything.
  - Hunt out valueless code by tree shaking your code, avoiding duplication, fewer and simpler queries, and (sparingly) using the latest JavaScript and API's.

  In relation to **content and assets**, make sure you:

  - Avoid unnecessary assets, but when they are needed, take a sustainable approach to images, media, animations, typefaces, and documents.
  - Provide alternatives to visual and audio content, as well as facilitate speech-based browsing of your content.
  - Write with purpose, in an accessible, easy to understand format.

Draft for a new RFC to expose carbon emissions in HTTP responses #71

bertysentry started this conversation in **Standards Working Group**

bertysentry 3 weeks ago

Help!

I proposed a draft of an RFC with the HTTP group of the IETF, that defines a new HTTP response header to provide the amount of CO2 (scope 2) emitted for the processing of that HTTP request. This header would be optional. I just want to make sure everybody uses the same way to communicate this information.

This would help organizations, applications, and clients to assess their scope 3 emissions when they leverage Web services:

- A user visiting a Web site, doing a Google search, a ChatGPT conversation.
- An application leveraging 3rd-party REST APIs.
- An infrastructure relying on SaaS platforms, online S3 storage, etc.

Unfortunately, I'm getting a lot of pushback from the HTTP working group. My draft can definitely be improved, and I'd like to keep the discussion going. Most importantly, some of the people at the HTTP working of the IETF requested experts to chime in and provide guidance (why would this new HTTP header would be beneficial, and what's the best way to achieve its goal).

The discussion is happening in a specific issue on the HTTP group's GitHub: httpwg/admin#52.

Anyone who could answer some of the questions on this thread is welcome! I need some backup here, and comments and suggestions from the GSF to get this RFC adopted.

Thank you!

@NAMRATA-WOKE @seanmcilroy29 @atg-abhishek @Henry-WattTime

↑ 1

Category

✅ Standards Working Group

Labels

Opportunity 🌱

1 participant

STATE OF GREEN SOFTWARE

Green Software Foundation

CULTURE

There is a business case for making software greener

CULTURE

Green Software is a priority for software practitioners, but still lacks leadership support

KNOWLEDGE TOOLS

Measuring carbon emissions is crucial for scaling sustainable AI

KNOWLEDGE TOOLS

Responsible AI is green AI

KNOWLEDGE TOOLS

Green Software is vital for a net zero future

CULTURE

Software legislation has quadrupled in the last decade

KNOWLEDGE

Green software is central to tech ethics

CULTURE

63% of CEOs do not rate sustainability as a top priority

KNOWLEDGE

Green software should be part of higher education

CULTURE

92% of software practitioners are concerned about climate change

CULTURE KNOWLEDGE

Open-source solutions are essential to greening software and ICT

TOOLS

Carbon-aware software is central to decarbonization

Green Software Foundation

64% of software practitioners are measuring their carbon impact

Developers and other ICT stakeholders are already working towards climate goals. According to the SOGS survey, 15.6% of over 2,000 software practitioners work for an organization that always measures the carbon impact of their software, and 49.3% say that their organization sometimes measures the carbon impact of their software.

Digital does not equal green

Software directly contributes to CO2 emissions shaping atmospheric changes and extreme weather events related to global warming. The cloud alone has a larger carbon footprint than the airline industry, increasing by 9% YOY. Beyond CO2 emissions, software has a range of environmental impacts, from water used for manufacturing or server cooling to land use for data centers.

Software emissions are equivalent to air, rail, and shipping combined

Software-related CO2 emissions account for 4-5% of global emissions. This is equivalent to the emissions of all aviation, shipping, and rail combined.

The current gold standard paper on the climate and transformative impact of ICT indicates that precise estimates of software-related CO2 emissions vary according to methodology, but generally agree as to the main sources: digital devices, data centers, and networks. Software development itself can be energy intensive, such as training LLMs (large language models) and other AI models. This is why Hugging Face calculates the carbon cost of AI using a life cycle analysis, taking into account both training and use over time along with the larger computing infrastructure.

As a result, the Cloud now has a greater carbon footprint than the airline industry. A single data center can consume the equivalent electricity of 50,000 homes. At 200 terawatt hours (TWh) annually, data centers collectively devour more energy than some nation-states. Today, the electricity utilized by data centers accounts for 0.3 percent of overall carbon emissions, and if we extend our accounting to include networked devices like laptops, smartphones, and tablets, the total shifts to 2 percent of global carbon emissions.

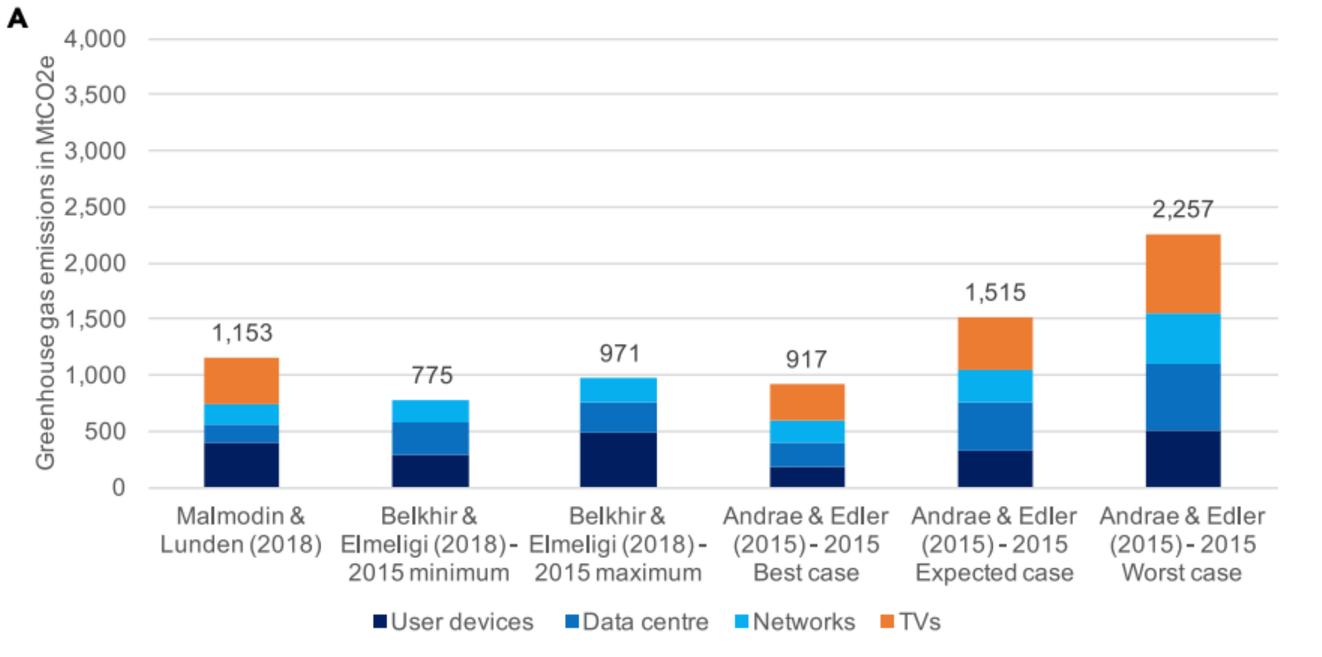
The study notes that the information and communication industry now accounts for about 1.5 percent of the total global carbon footprint.

"If trends continue, ICT will account for as much as 14 percent of the total global footprint by 2040, or about half of the entire transportation sector worldwide," Balkhir said.

Review

The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations

Charlotte Freitag,[1] Mike Berners-Lee,[1] Kelly Widdicks,[2,*] Bran Knowles,[2] Gordon S. Blair,[2] and Adrian Friday[2]
[1]Small World Consulting, Gordon Manley Building, Lancaster Environment Centre, Lancaster University, Lancaster, Lancashire LA1 4YQ, UK
[2]School of Computing and Communications, InfoLab21, Lancaster University, Lancaster, Lancashire LA1 4WA, UK
*Correspondence: k.v.widdicks@lancaster.ac.uk
https://doi.org/10.1016/j.patter.2021.100340



Green Software Foundation

Green Software Verified Facts #47

danielvaughan started this conversation in **Community Working Group**

danielvaughan on Aug 10

We have been preparing internal material on sustainability. We want to get three solid facts to present but it is hard to get facts that are verifiable.

For example suppose we want to make a statement about the carbon footprint of data centres.

There seems to be many figures and comparisons e.g. .They emit as much carbon as the aviation industry. However when you go looking for the source and the detail it seems it is easy to get a confused and inaccurate statement. For example:

- The subject of a statement can be the "Digital World", the ICT Industry or Date Centers
- The measure could be emissions or electricity consumption
- The timeframe can be now or a projection for the future

Therefore it is easy for someone in a talk to say: "Data centers produce 20% of the world's carbon emissions" where the original fact could have been something like "The Digital World is projected to use 20% of the world's electricity by 2025".

The risk is we alienate people with incorrect facts. I remember years ago we had an environment group at work that sent an email saying a TV left on standby uses as much electricity as when it is on. It ended up with a load of replies from people saying that was nonsense. In reality the original stat was probably something like a TV left on standby for 23 hours uses the same energy as being watched for an hour.

Could we maintain a repository of "GSF Facts" with citations that people could use safely in the knowledge that they can be backed up?

↑ 12 👍 8

Cloud Computing

ENVIRONMENT JUNE 19, 2007 / 3:47 PM / UPDATED 16 YEARS AGO

Google aims to go carbon-neutral by end 2007

By Timothy Gardner

3 MIN READ

NEW YORK (Reuters) - Google Inc. GOOG.O aims to voluntarily cut or offset all of its greenhouse emissions by the end of the year, the Web search leader said on Tuesday .

TECH / GOOGLE / SCIENCE

Google now purchases more renewable energy than it consumes as a company

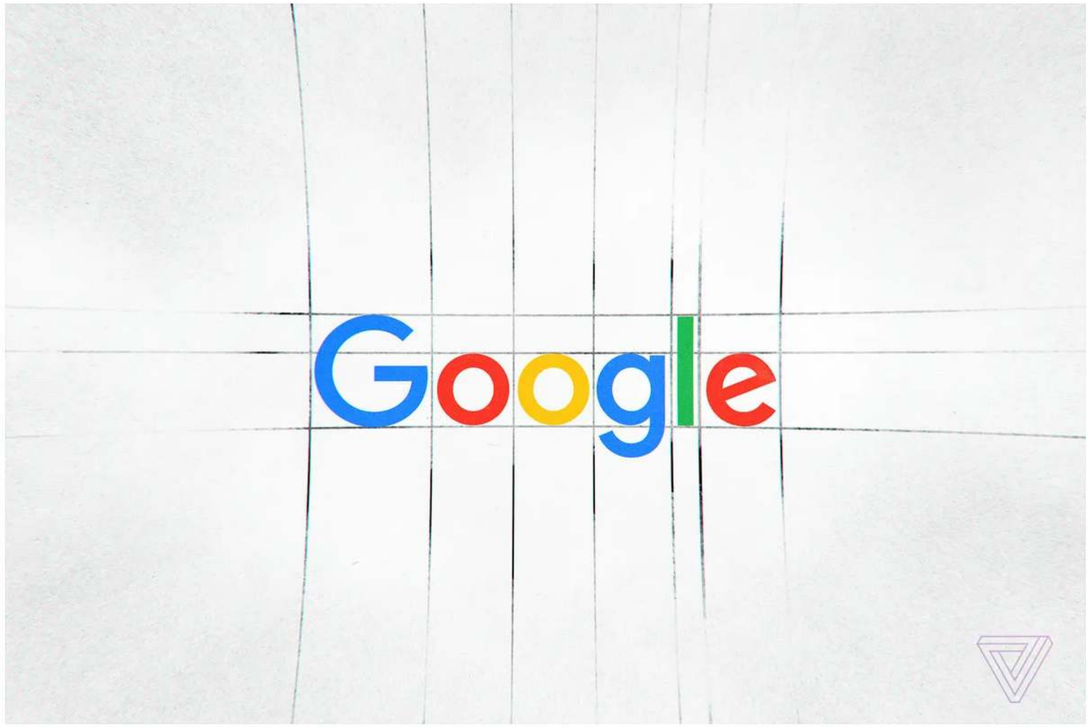


Illustration by Alex Castro / The Verge

By Dani Deahl

Apr 4, 2018 at 7:51 PM GMT+2 | 0 Comments / 0 New

Google announced in a blog post that it now purchases more renewable energy than it consumes as a company. Google began these efforts in 2017, with the goal of purchasing as much renewable energy as it uses across its 13 data centers and all of its office complexes.

SCIENCE / ENVIRONMENT / TECH

Google announced one of the biggest green pledges from tech yet

Google CEO Sundar Pichai Photo by Justin Sullivan/Getty Images

/ It will run on carbon-free energy by 2030

By Justine Calma, a science reporter covering the environment, climate, and energy with a decade of experience. She is also the host of the Hell or High Water podcast.

Sep 14, 2020 at 5:36 PM GMT+2 | 0 Comments / 0 New

Google just made one of Big Tech's most ambitious environmental commitments: it will work to run its operations purely on carbon-free energy by 2030. It also announced that as of today, it has purchased enough carbon offsets to essentially cancel out all the planet-heating carbon dioxide emissions the company has released since it was founded in 1998.

	Carbon Neutral *offsets emissions* achieved by purchasing carbon offsets that reduce or prevent global emissions	**100% Renewable** *reduces emissions* achieved by purchasing enough renewable energy to match annual electricity use	**24/7 Carbon-free** *eliminates emissions** achieved by sourcing clean energy for every location and every hour of operation
Google's progress	**Since 2007**	**Since 2017**	**By 2030**
Helps combat climate change	✅	✅	✅
Encourages full-scale transformation of electric grids	❌	❌	✅
Directly reduces carbon emissions associated with electricity use	❌	✅	✅
Eliminates all carbon emissions associated with electricity use	❌	❌	✅
Matches *annual* electricity consumption with clean energy	❌	✅	✅
Matches *hourly* electricity consumption with clean energy	❌	❌	✅
Directly increases amount of clean energy on *some* electric grids where a company operates	❌	✅	✅
Directly increases amount of clean energy on *all* electric grids where a company operates	❌	❌	✅

To offset emissions that could not be reduced immediately, Google invested in external projects that absorb CO2 or reduce emissions elsewhere.

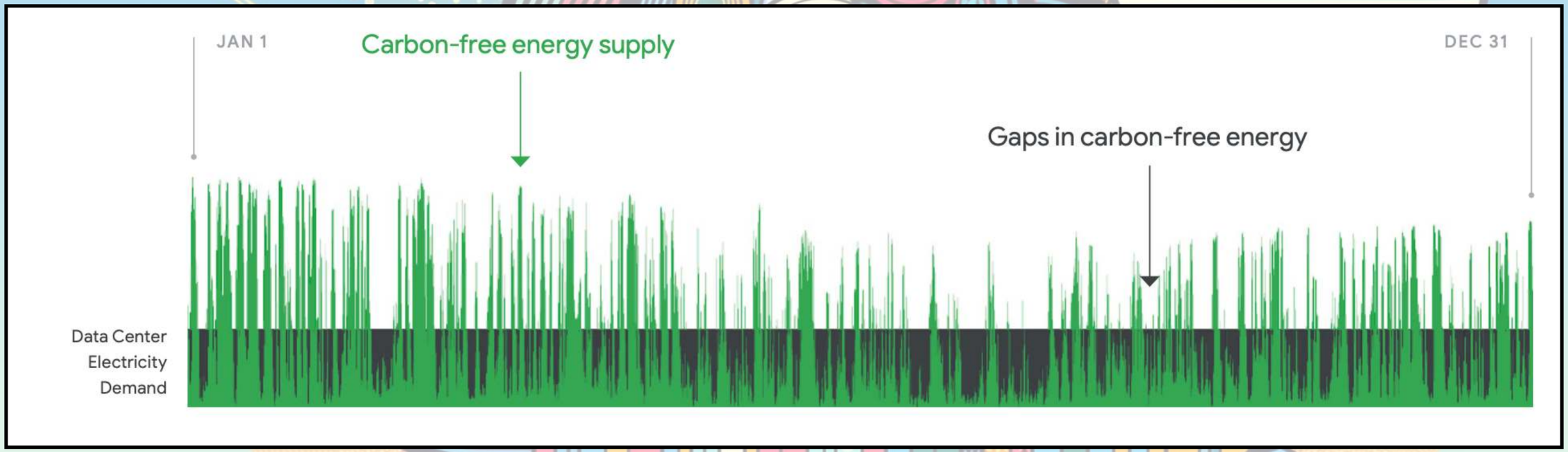
These projects included financing wind farms, solar installations, the purchase of carbon credits and forestry initiatives that bind CO2 through tree photosynthesis.

Carbon Neutral in 2007

Zero Emission in 24-hour operation

This does not mean that every operation of the company was powered directly by renewable sources, but the company purchased enough renewable energy (or renewable energy credits) to offset its total energy consumption.

100% renewable energy sources in 2017



100% renewable energy sources in 2017

Zero Emissions 1h

Not only would the energy used by Google come from 100% renewable sources, but **all of the company's processes, products** and **operations** would produce no carbon dioxide or other greenhouse gas emissions.

Carbon Free in 2030

Programming languages

Towards a Green Ranking for Programming Languages *

Marco Couto, Rui Pereira
HASLab/INESC TEC
Universidade do Minho
Portugal
marco.l.couto@inesctec.pt,
ruipereira@di.uminho.pt

Francisco Ribeiro, Rui Rua
HASLab/INESC TEC
Universidade do Minho
Portugal
{fribeiro,rrua}@di.uminho.pt

João Saraiva
HASLab/INESC TEC
Universidade do Minho
Portugal
saraiva@di.uminho.pt

Energy Efficiency across Programming Languages

How Do Energy, Time, and Memory Relate?

Rui Pereira
HASLab/INESC TEC
Universidade do Minho, Portugal
ruipereira@di.uminho.pt

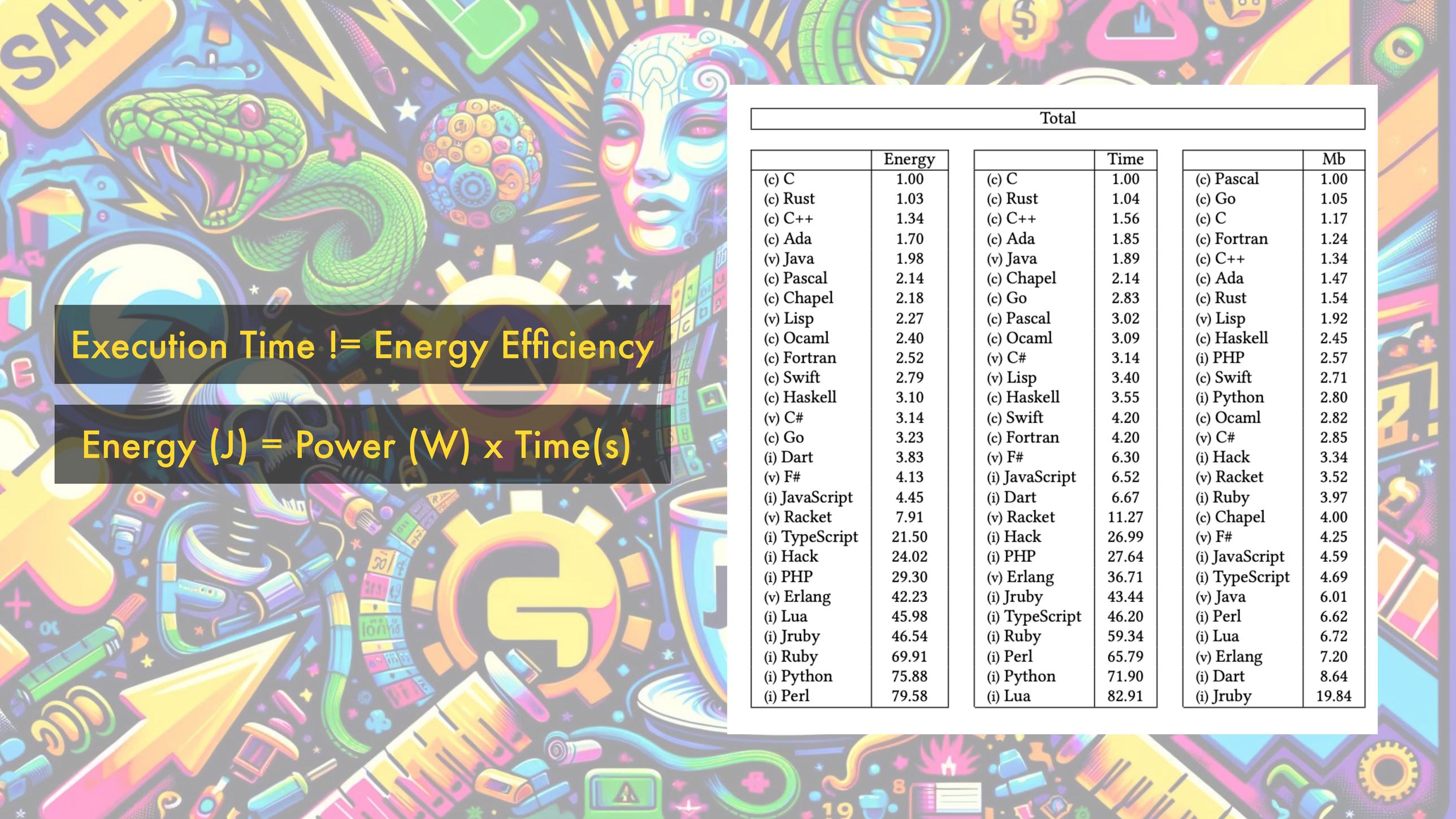
Marco Couto
HASLab/INESC TEC
Universidade do Minho, Portugal
marco.l.couto@inesctec.pt

Francisco Ribeiro, Rui Rua
HASLab/INESC TEC
Universidade do Minho, Portugal
fribeiro@di.uminho.pt
rrua@di.uminho.pt

Jácome Cunha
NOVA LINCS, DI, FCT
Univ. Nova de Lisboa, Portugal
jacome@fct.unl.pt

João Paulo Fernandes
Release/LISP, CISUC
Universidade de Coimbra, Portugal
jpf@dei.uc.pt

João Saraiva
HASLab/INESC TEC
Universidade do Minho, Portugal
saraiva@di.uminho.pt



Execution Time != Energy Efficiency

Energy (J) = Power (W) x Time(s)

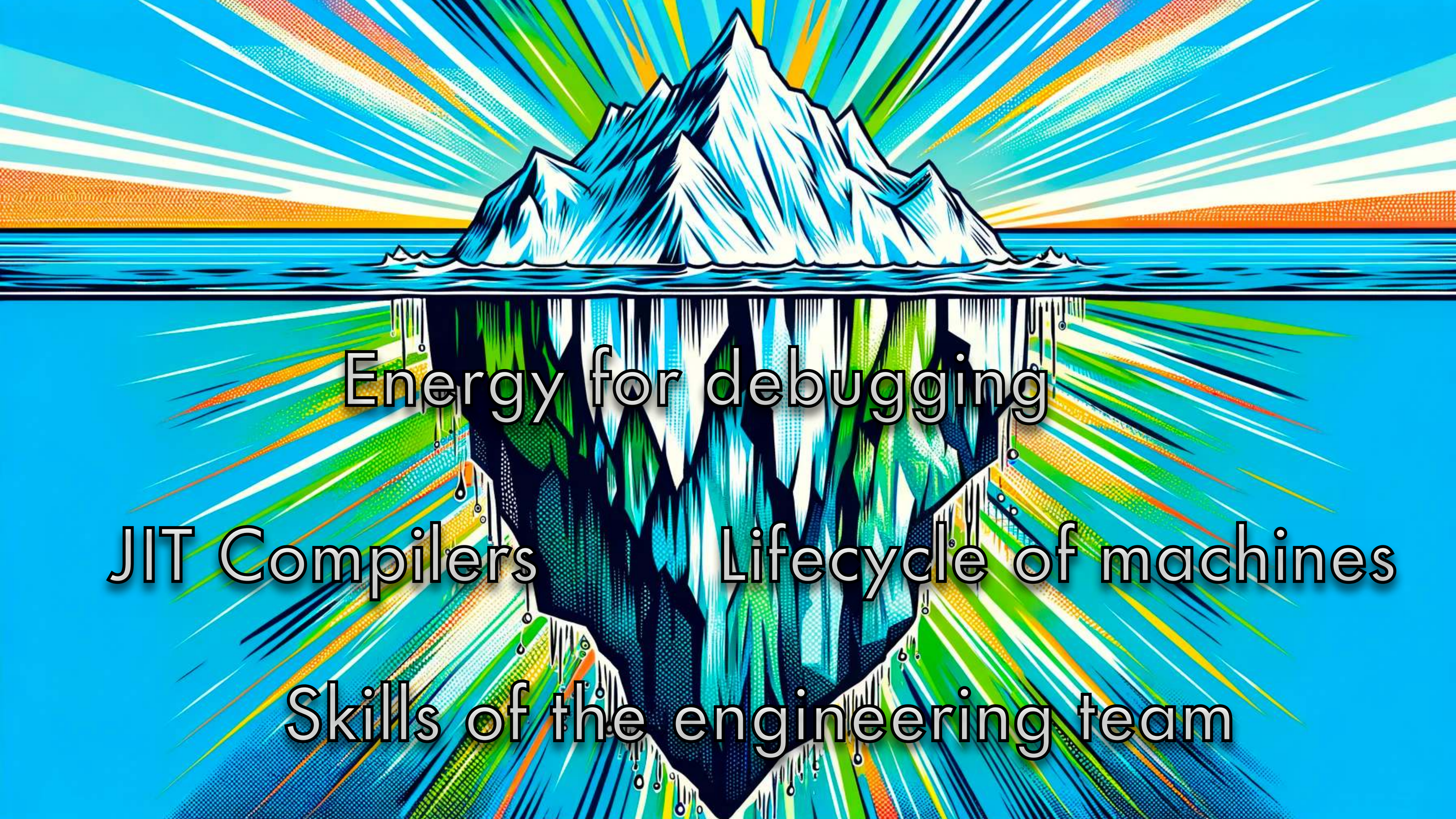
Total

	Energy
(c) C	1.00
(c) Rust	1.03
(c) C++	1.34
(c) Ada	1.70
(v) Java	1.98
(c) Pascal	2.14
(c) Chapel	2.18
(v) Lisp	2.27
(c) Ocaml	2.40
(c) Fortran	2.52
(c) Swift	2.79
(c) Haskell	3.10
(v) C#	3.14
(c) Go	3.23
(i) Dart	3.83
(v) F#	4.13
(i) JavaScript	4.45
(v) Racket	7.91
(i) TypeScript	21.50
(i) Hack	24.02
(i) PHP	29.30
(v) Erlang	42.23
(i) Lua	45.98
(i) Jruby	46.54
(i) Ruby	69.91
(i) Python	75.88
(i) Perl	79.58

	Time
(c) C	1.00
(c) Rust	1.04
(c) C++	1.56
(c) Ada	1.85
(v) Java	1.89
(c) Chapel	2.14
(c) Go	2.83
(c) Pascal	3.02
(c) Ocaml	3.09
(v) C#	3.14
(v) Lisp	3.40
(c) Haskell	3.55
(c) Swift	4.20
(c) Fortran	4.20
(v) F#	6.30
(i) JavaScript	6.52
(i) Dart	6.67
(v) Racket	11.27
(i) Hack	26.99
(i) PHP	27.64
(v) Erlang	36.71
(i) Jruby	43.44
(i) TypeScript	46.20
(i) Ruby	59.34
(i) Perl	65.79
(i) Python	71.90
(i) Lua	82.91

	Mb
(c) Pascal	1.00
(c) Go	1.05
(c) C	1.17
(c) Fortran	1.24
(c) C++	1.34
(c) Ada	1.47
(c) Rust	1.54
(v) Lisp	1.92
(c) Haskell	2.45
(i) PHP	2.57
(c) Swift	2.71
(i) Python	2.80
(c) Ocaml	2.82
(v) C#	2.85
(i) Hack	3.34
(v) Racket	3.52
(i) Ruby	3.97
(c) Chapel	4.00
(v) F#	4.25
(i) JavaScript	4.59
(i) TypeScript	4.69
(v) Java	6.01
(i) Perl	6.62
(i) Lua	6.72
(v) Erlang	7.20
(i) Dart	8.64
(i) Jruby	19.84

Time & Memory	Energy & Time	Energy & Memory	Energy & Time & Memory
C • Pascal • Go	C	C • Pascal	C • Pascal • Go
Rust • C++ • Fortran	Rust	Rust • C++ • Fortran • Go	Rust • C++ • Fortran
Ada	C++	Ada	Ada
Java • Chapel • Lisp • Ocaml	Ada	Java • Chapel • Lisp	Java • Chapel • Lisp • Ocaml
Haskell • C#	Java	OCaml • Swift • Haskell	Swift • Haskell • C#
Swift • PHP	Pascal • Chapel	C# • PHP	Dart • F# • Racket • Hack • PHP
F# • Racket • Hack • Python	Lisp • Ocaml • Go	Dart • F# • Racket • Hack • Python	JavaScript • Ruby • Python
JavaScript • Ruby	Fortran • Haskell • C#	JavaScript • Ruby	TypeScript • Erlang
Dart • TypeScript • Erlang	Swift	TypeScript	Lua • JRuby • Perl
JRuby • Perl	Dart • F#	Erlang • Lua • Perl	
Lua	JavaScript	JRuby	
	Racket		
	TypeScript • Hack		
	PHP		
	Erlang		
	Lua • JRuby		
	Ruby		


Energy for debugging
JIT Compilers
Lifecycle of machines
Skills of the engineering team

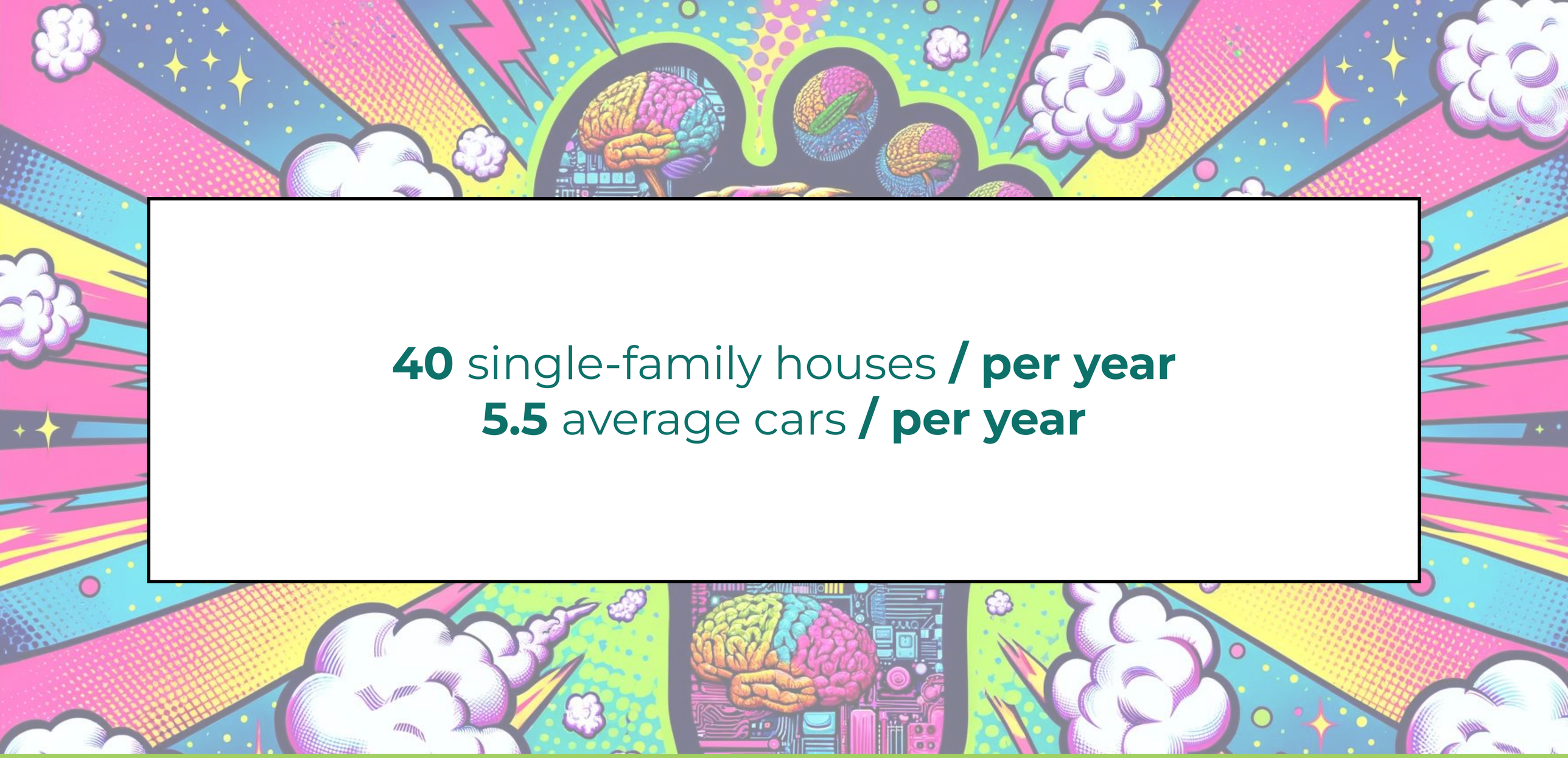
AI

Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model

Alexandra Sasha Luccioni
Hugging Face
sasha.luccioni@hf.co

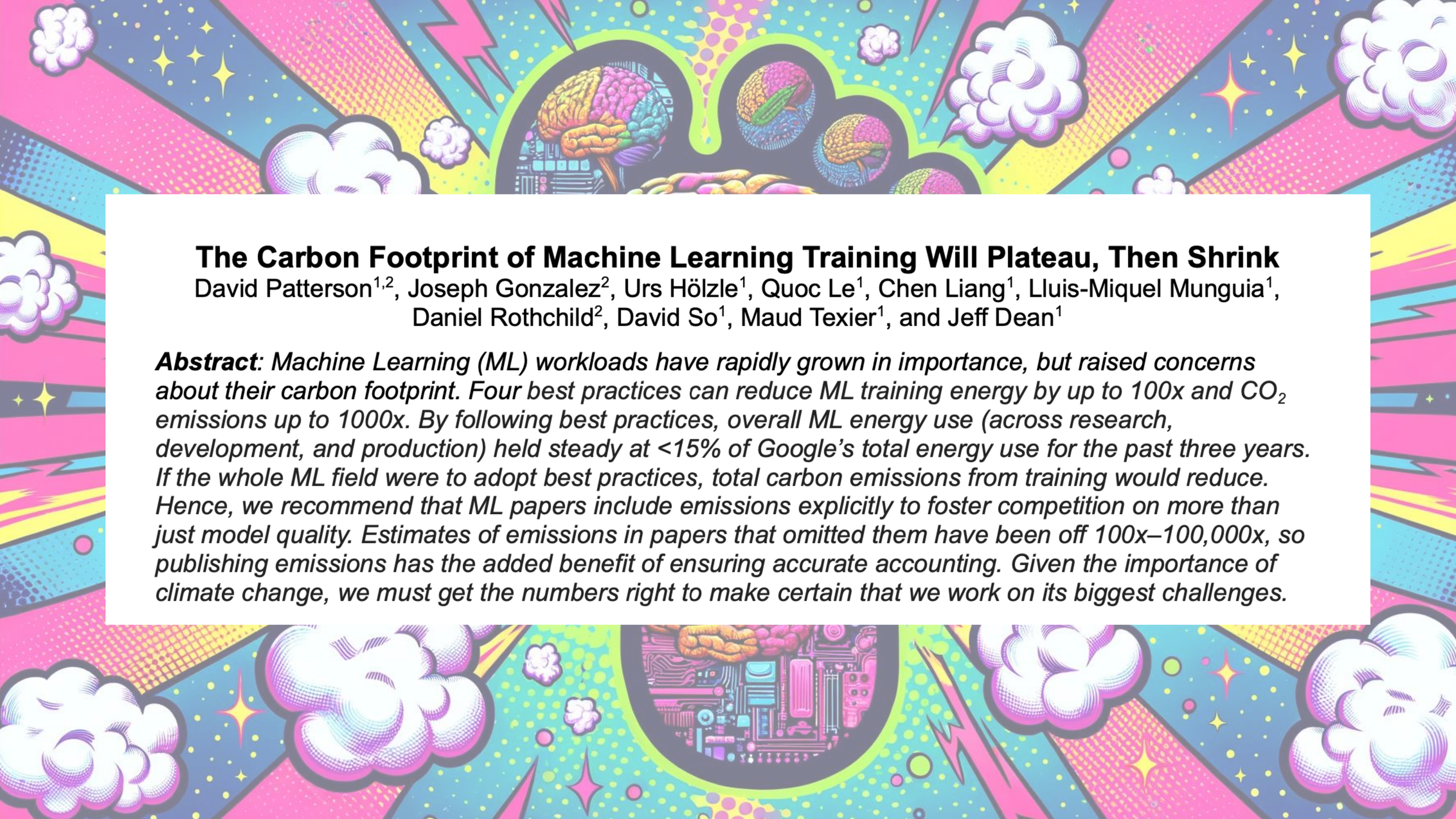
Sylvain Viguier
Graphcore
sylvainv@graphcore.ai

Anne-Laure Ligozat
LISN & ENSIIE
anne-laure.ligozat
@lisn.upsaclay.fr



40 single-family houses **/ per year**
5.5 average cars **/ per year**

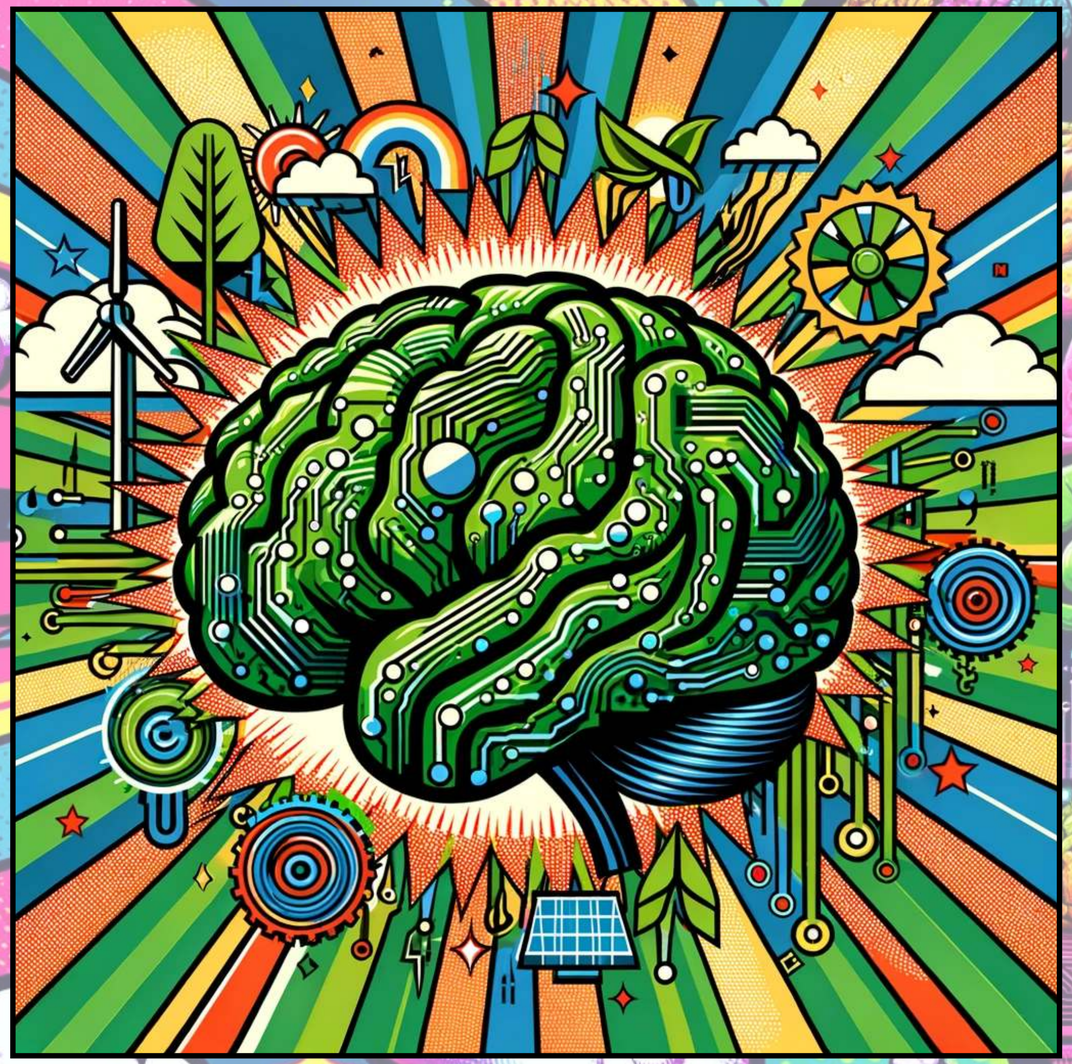
433,196 kWh i **24.69** ton CO2eq



The Carbon Footprint of Machine Learning Training Will Plateau, Then Shrink

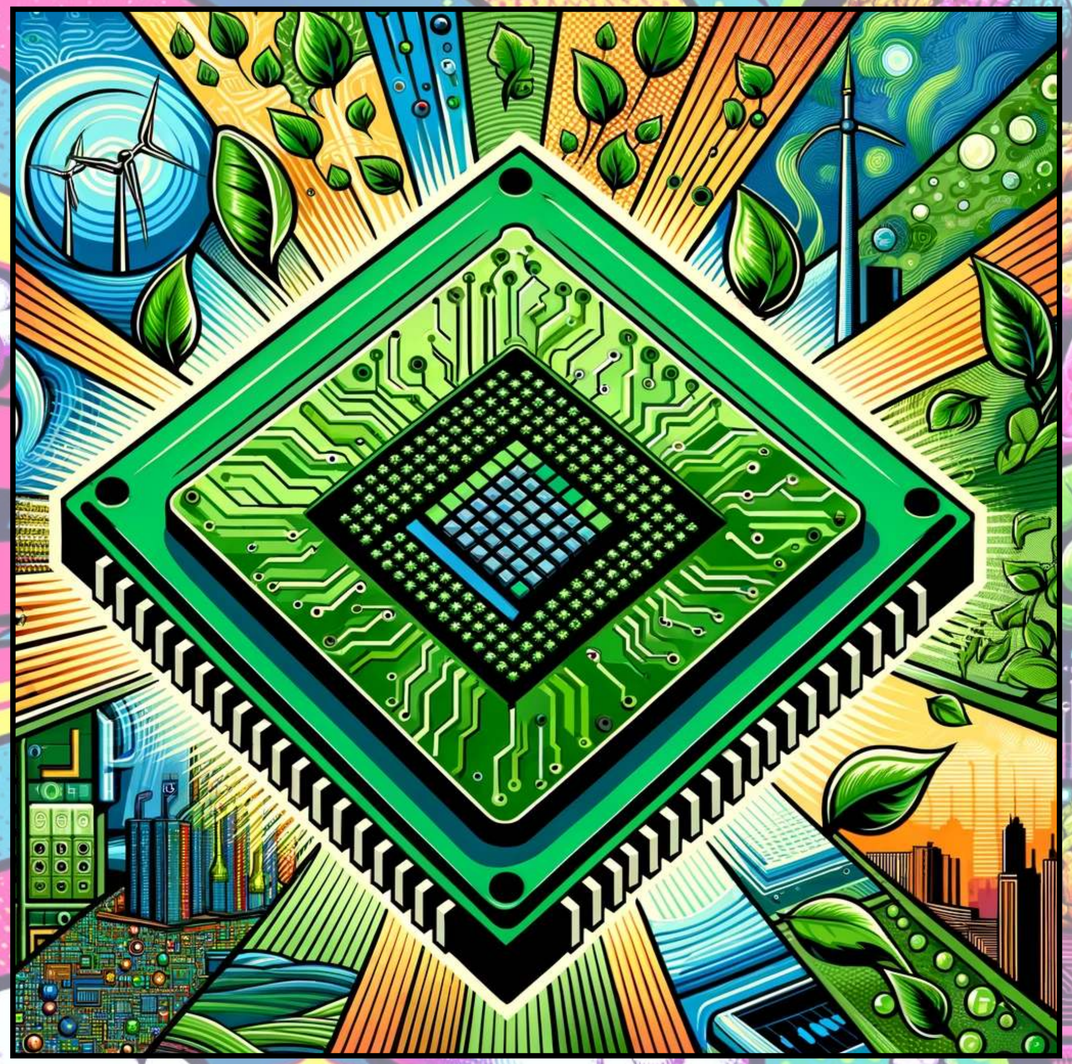
David Patterson[1,2], Joseph Gonzalez[2], Urs Hölzle[1], Quoc Le[1], Chen Liang[1], Lluis-Miquel Munguia[1], Daniel Rothchild[2], David So[1], Maud Texier[1], and Jeff Dean[1]

***Abstract**: Machine Learning (ML) workloads have rapidly grown in importance, but raised concerns about their carbon footprint. Four best practices can reduce ML training energy by up to 100x and CO_2 emissions up to 1000x. By following best practices, overall ML energy use (across research, development, and production) held steady at <15% of Google's total energy use for the past three years. If the whole ML field were to adopt best practices, total carbon emissions from training would reduce. Hence, we recommend that ML papers include emissions explicitly to foster competition on more than just model quality. Estimates of emissions in papers that omitted them have been off 100x–100,000x, so publishing emissions has the added benefit of ensuring accurate accounting. Given the importance of climate change, we must get the numbers right to make certain that we work on its biggest challenges.*



Model

Choosing efficient architectures for machine learning models while developing ML quality, such as sparse models versus dense models, can improve performance by around 5-10 times.



Machine

Using processors optimised for ML training, such as TPUs or new GPUs (e.g. V100 or A100), compared to general purpose processors, can improve perform

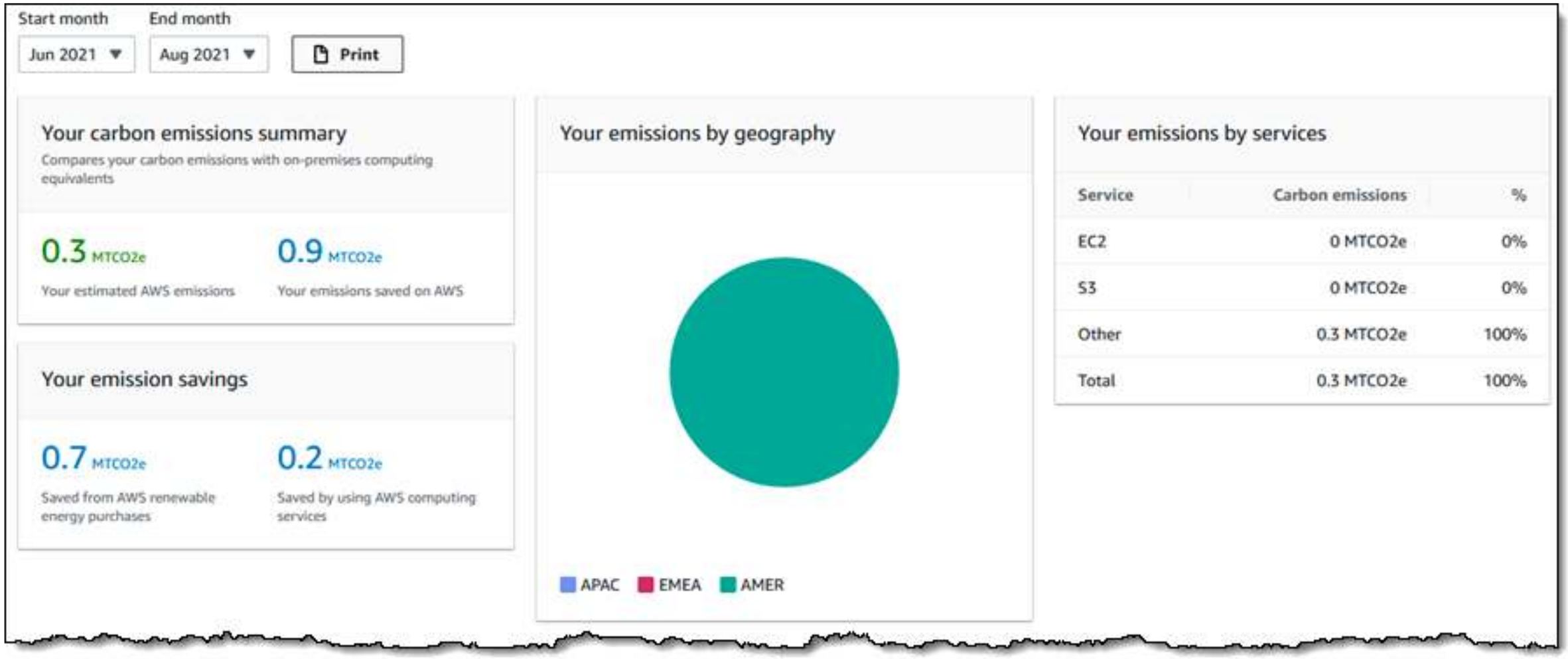
Mechanisation

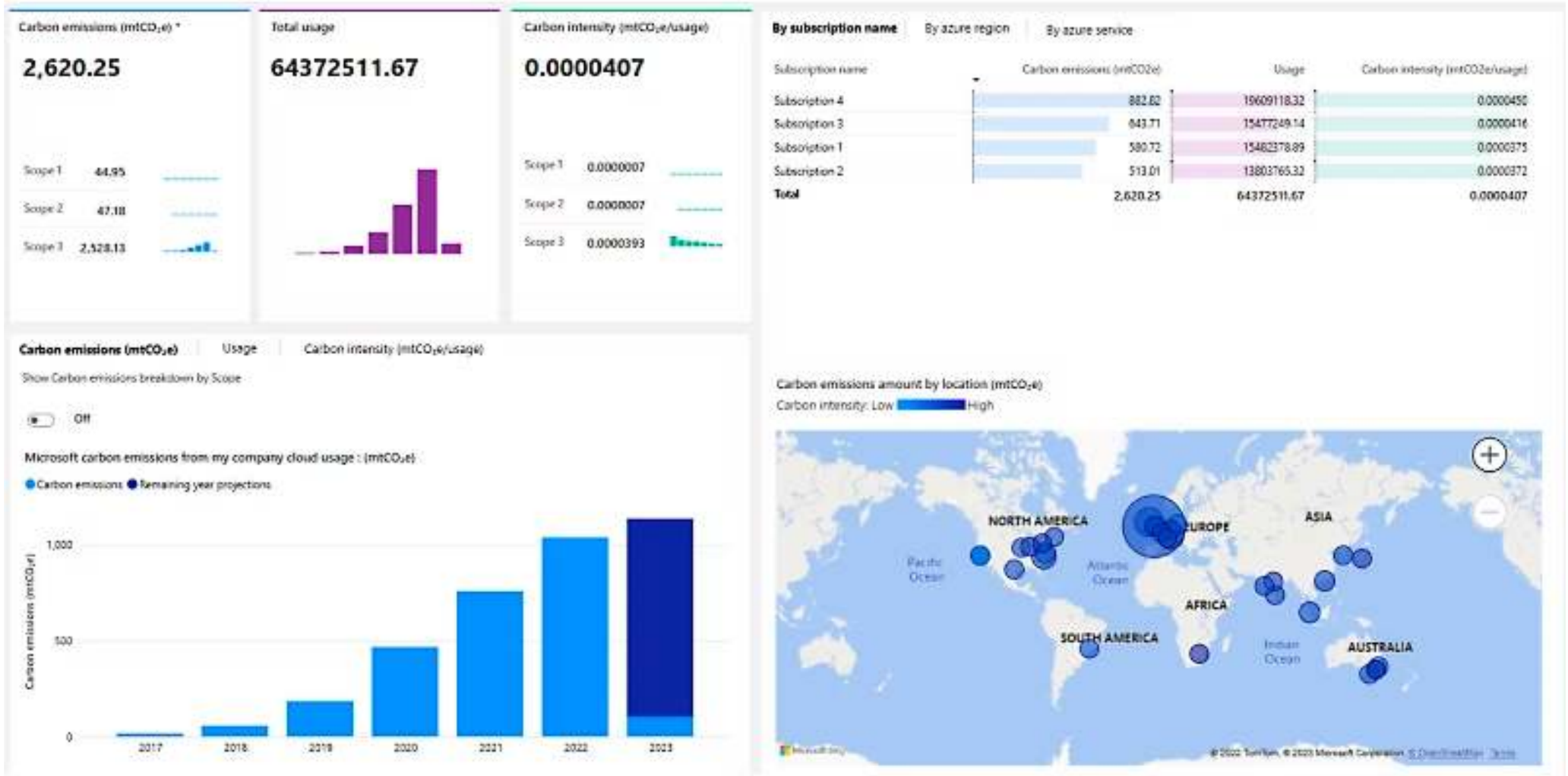
Cloud computing instead of local computing improves the energy efficiency of data centres, reducing energy costs by a factor of 1.4-2.

Map

Cloud computing allows ML practitioners to select the location with the cleanest energy, further reducing the overall carbon footprint by factors of 5-105.

Tooling


Start month
End month
Jun 2021
Aug 2021
Print
Your carbon emissions summary
Compares your carbon emissions with on-premises computing equivalents
0.3 MTCO2e
Your estimated AWS emissions
0.9 MTCO2e
Your emissions saved on AWS
Your emission savings
0.7 MTCO2e
Saved from AWS renewable energy purchases
0.2 MTCO2e
Saved by using AWS computing services
Your emissions by geography
APAC
EMEA
AMER
Your emissions by services
Service
Carbon emissions
%
EC2
0 MTCO2e
0%
S3
0 MTCO2e
0%
Other
0.3 MTCO2e
100%
Total
0.3 MTCO2e
100%


Carbon emissions (mtCO2e) *
2,620.25
Scope 1 44.95
Scope 2 47.18
Scope 3 2,528.13
Total usage
64372511.67
Carbon intensity (mtCO2e/usage)
0.0000407
Scope 1 0.0000007
Scope 2 0.0000007
Scope 3 0.0000393
By subscription name
By azure region
By azure service
Subscription name
Carbon emissions (mtCO2e)
Usage
Carbon intensity (mtCO2e/usage)
Subscription 4 882.82 19609118.32 0.0000450
Subscription 3 643.71 15477249.14 0.0000416
Subscription 1 580.72 15482378.89 0.0000375
Subscription 2 513.01 13803765.32 0.0000372
Total 2,620.25 64372511.67 0.0000407
Carbon emissions (mtCO2e)
Usage
Carbon intensity (mtCO2e/usage)
Show Carbon emissions breakdown by Scope
Off
Microsoft carbon emissions from my company cloud usage : (mtCO2e)
Carbon emissions
Remaining year projections
Carbon emissions (mtCO2e)
1,000
500
0
2017
2018
2019
2020
2021
2022
2023
Carbon emissions amount by location (mtCO2e)
Carbon intensity: Low
High
NORTH AMERICA
EUROPE
ASIA
AFRICA
SOUTH AMERICA
AUSTRALIA
Pacific Ocean
Atlantic Ocean
Indian Ocean

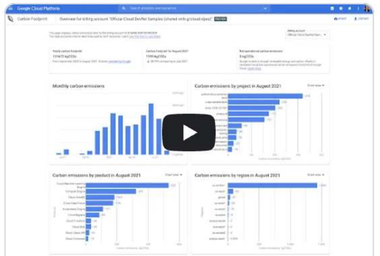
Microsoft

Carbon Footprint

Measure, report, and reduce your cloud carbon emissions.

Go to console

- Include gross carbon emissions data in reports and disclosures
- Visualize carbon insights via dashboards and charts
- Reduce the gross emissions of cloud applications and infrastructure



VIDEO

Watch this session and learn how to measure carbon emissions on Google Cloud

8:02

BENEFITS

Accurately measure your gross carbon footprint

View the gross, location-based emissions that derive from your Google Cloud usage, providing

Track the emissions profile of cloud projects

Monitor your gross cloud emissions over time by project, product, and region—giving IT

Share detailed methodology with reviewers

Our detailed calculation methodology is published so that

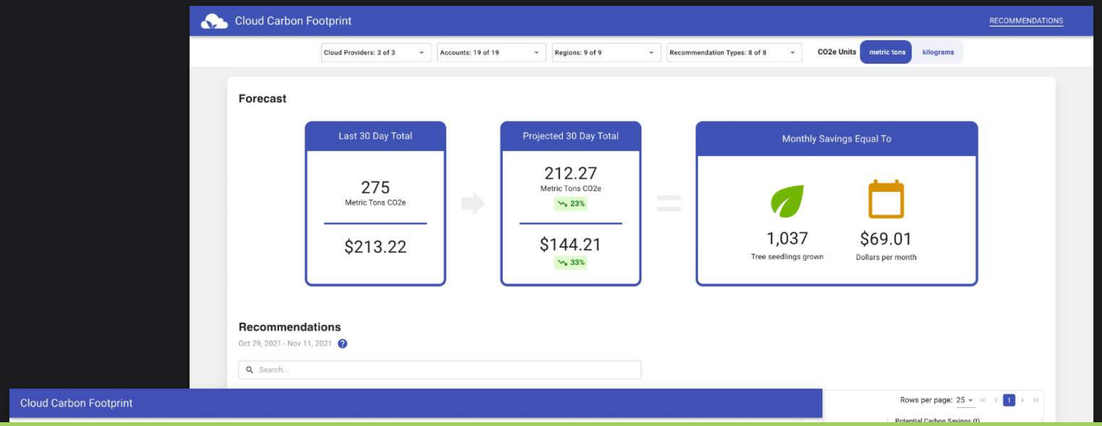
Cloud Carbon Footprint

Free and Open Source

Cloud Carbon Emissions Measurement and Analysis Tool

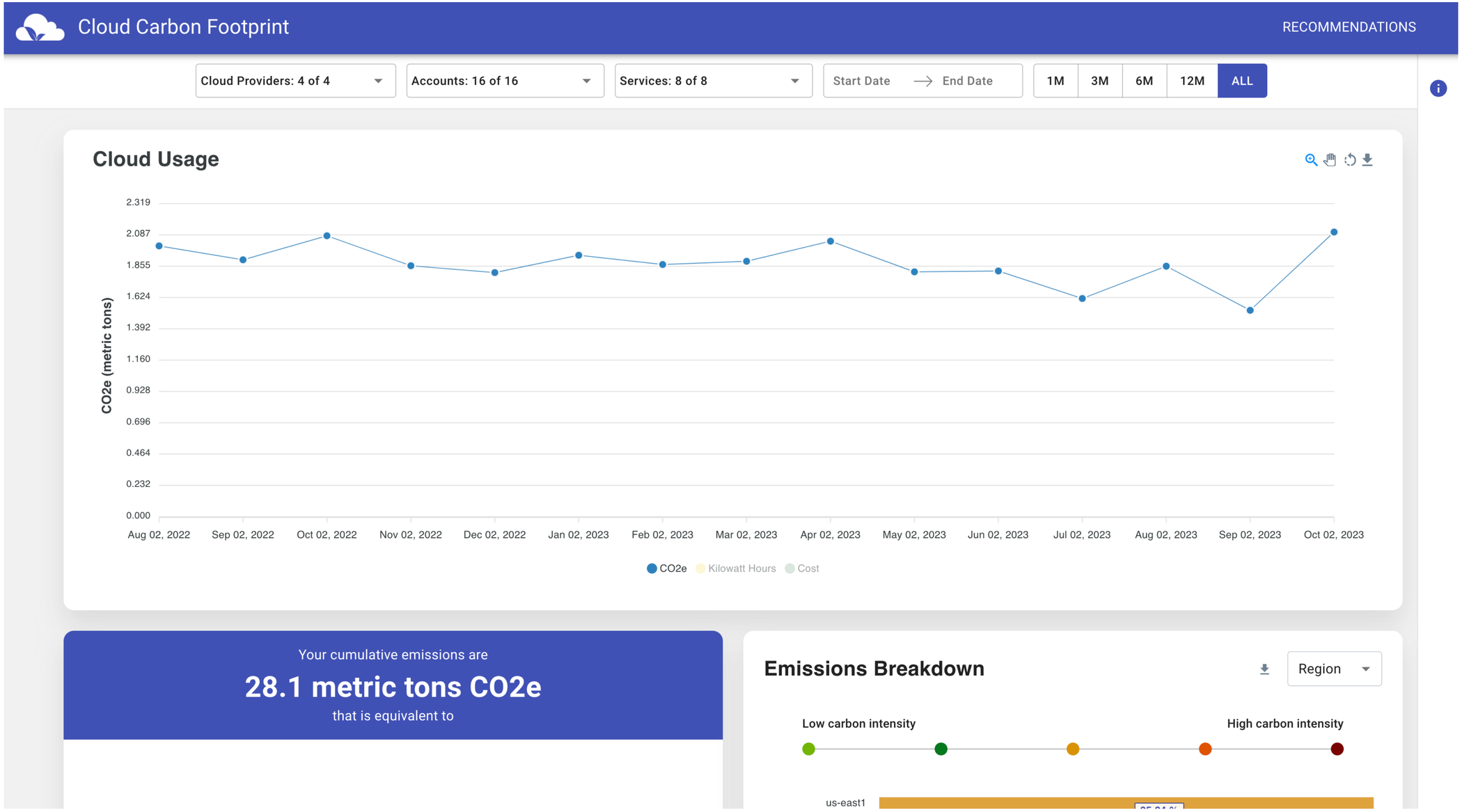
Understand how your cloud usage impacts our environment and what you can do about it

TRY DEMO NOW



Get to know the carbon

Carbon Footprint



Carbon Footprint

52. Cloud Carbon Footprint

Trial

Cloud Carbon Footprint (CCF) is an open-source tool that estimates carbon emissions for cloud workloads across the major cloud service providers. It queries cloud APIs for resource usage data and uses multiple sources to track carbon emissions. Following a published methodology, CCF combines these into emission estimates and provides a visualization of the data over time. Cloud providers have started adding similar offerings to their platforms, but organizations are still deploying CCF because it has all of the following features: It's open-source, designed to be extended, works across multiple clouds and has a transparent, published methodology. In addition, it also includes estimates for scope 2 and scope 3 emissions — for electricity use and hardware production, respectively. In our experiments, the estimates between different tools have varied, which is not a huge surprise given that all tools in this space make estimates and multiply estimated numbers. However, settling on one tool, taking a baseline and improving from that baseline is the key usage scenario we've come across, and tools like Kepler may reduce the need for estimates in the future. CCF also delivers GCP and AWS-sourced optimization recommendations, which not only help reduce your cloud carbon footprint but can also become part of a wider cloud cost optimization strategy. Thoughtworks is a significant contributor to CCF.

Cloud Carbon Footprint

JoularJX

License GPLv3 Made with Java

JoularJX is a Java-based agent for software power monitoring at the source code level.

Detailed documentation (including user and reference guides) are available at: https://joular.github.io/joularjx/.

🚀 Features

- Monitor power consumption of each method at runtime
- Uses a Java agent, no source code instrumentation needed
- Uses Intel RAPL (powercap interface) for getting accurate power reading on GNU/Linux, our research-based regression models on Raspberry Pi devices, and a custom program monitor (based on Intel Power Gadget) for accurate power readings on Windows
- Provides real-time power consumption of every method in the monitored program
- Provides total energy for every method on program exit

JoularJX for Java

- On Windows, JoularJX uses a custom power monitor program that uses Intel Power Gadget API on Windows, and therefore require installing the Intel Power Gadget tool and using a supported Intel CPU.
- On PC/server GNU/Linux, JoularJX uses Intel RAPL interface through powercap, and therefore requires running on an Intel CPU or an AMD Ryzen CPU.
- On macOS, JoularJX uses `powermetrics`, a tool bundled with macOS which requires running with `sudo` access. It is recommended to authorize the current users to run `/usr/bin/powermetrics` without requiring a password by making the proper modification to the `sudoers` file.

- On Windows, JoularJX uses a custom power monitor program that uses Intel Power Gadget API on Windows, and therefore require installing the Intel Power Gadget tool and using a supported Intel CPU.
- On PC/server GNU/Linux, JoularJX uses Intel RAPL interface through powercap, and therefore requires running on an Intel CPU or an AMD Ryzen CPU.
- On macOS, JoularJX uses `powermetrics`, a tool bundled with macOS which requires running with `sudo` access. It is recommended to authorize the current users to run `/usr/bin/powermetrics` without requiring a password by making the proper modification to the `sudoers` file.

Is there “The Greenest JVM”?

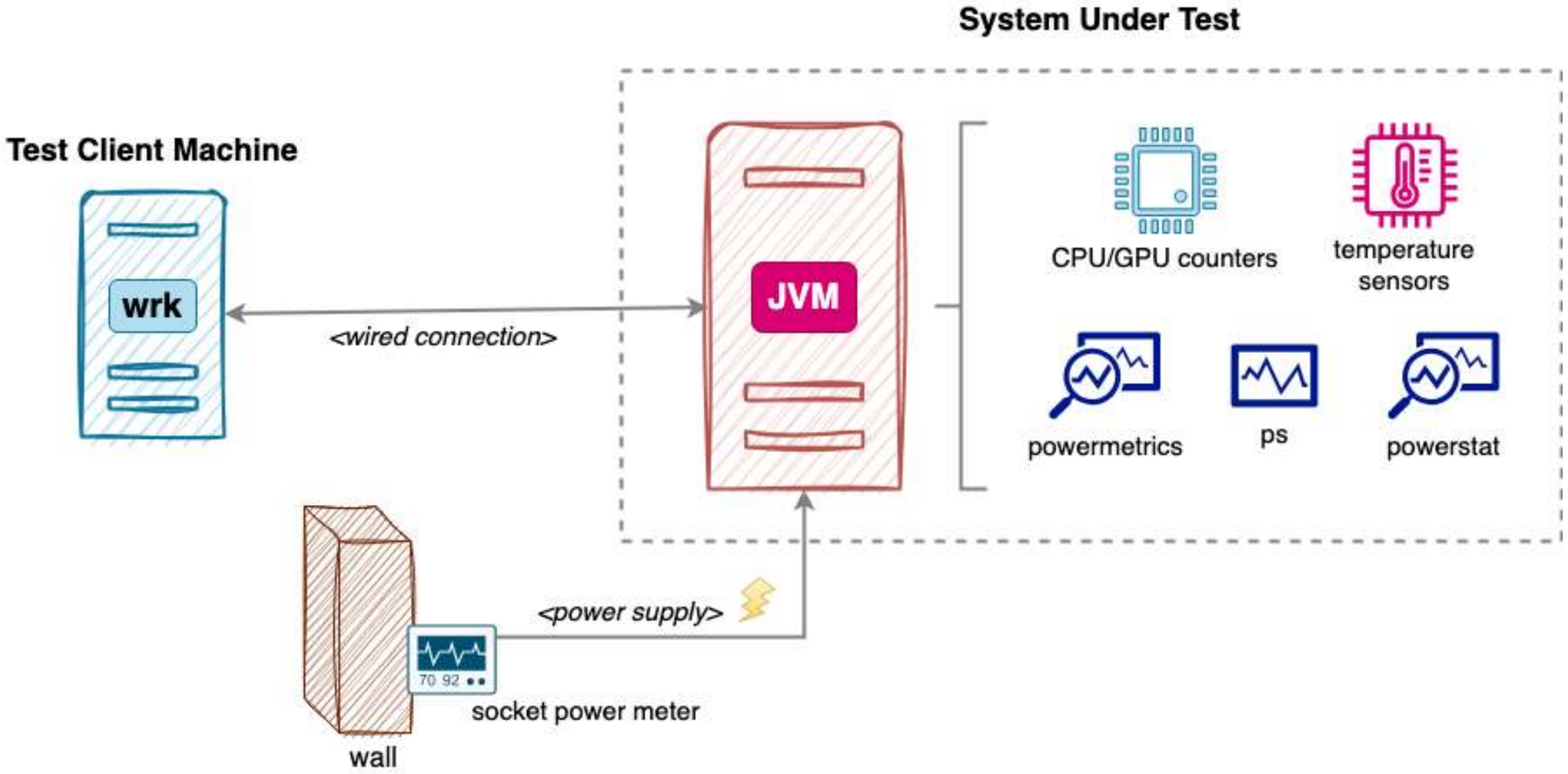
ENERGY EFFICIENCY

Analyzing JVM Energy Consumption for JDK 21: An Empirical Study

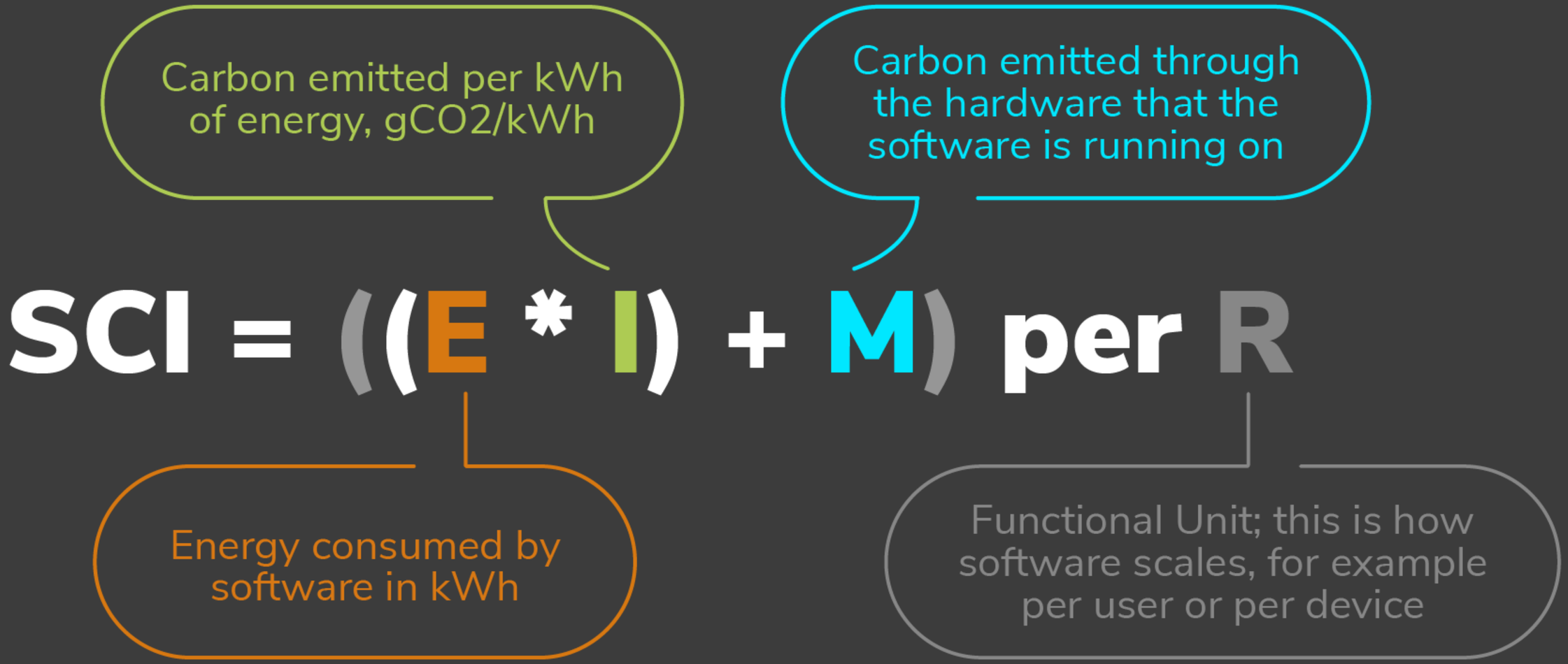
By **Ionut Balosin**

MAR 28, 2024 aot, azul prime, coding guidelines, eco friendly, energy, energy efficiency, graalvm, green code, hotspot vm, intel, java, java virtual machine, JDK21, jit, joule, jvm energy, logging, measurements, memory access patterns, power, power consumption, quarkus, rapl, running average power limit, sorting algorithms, spring, string concatenation, throughput, throwing exception, virtual calls, virtual threads, wall power, watt, wrk, wrk2

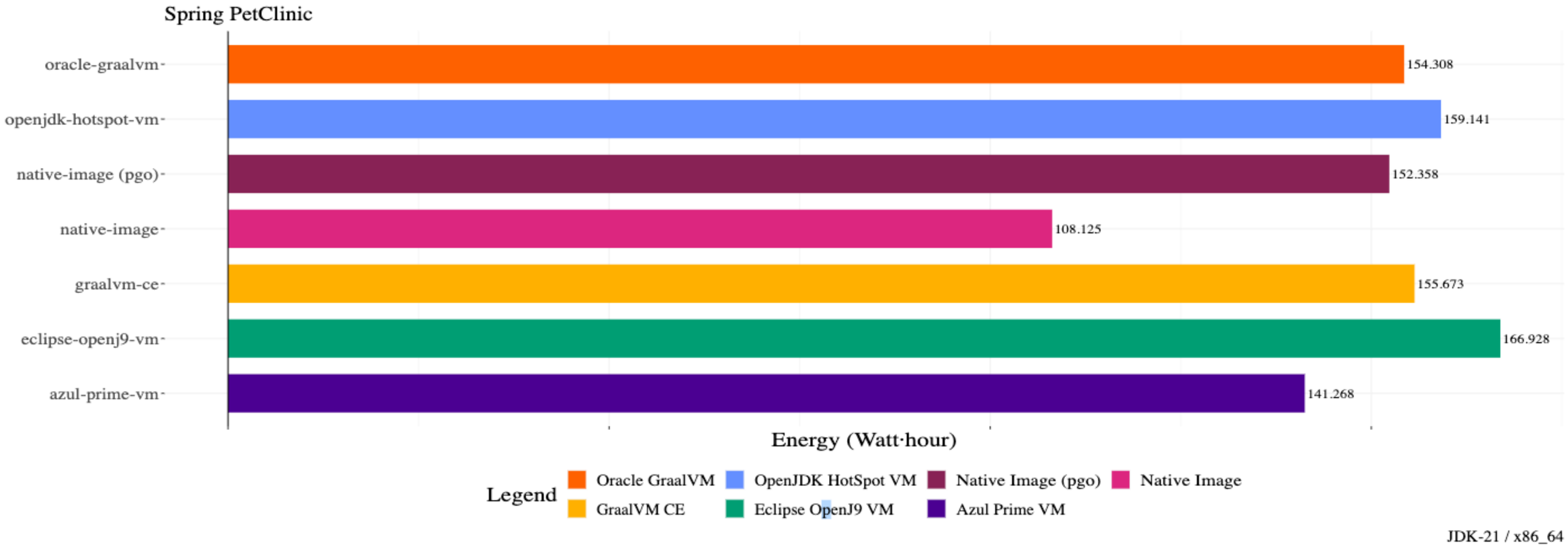
Analyzing JVM Energy Consumption for JDK 21



JoularJX for Java



No.	JVM distribution	Architecture	Total Energy (Watt·sec)	CO_2 Emission Factor (gCO_2eq/kWh)	CO_2 Emissions (gCO_2)
1	Oracle GraalVM Native Image	x86_64	1,819,642.816	168	84.917
2	Azul Prime VM	x86_64	1,860,940.776	168	86.844
3	Eclipse OpenJ9 VM	x86_64	1,929,280.408	168	90.033
4	Oracle GraalVM Native Image (PGO)	x86_64	2,086,482.972	168	97.369
5	Oracle GraalVM	x86_64	2,092,295.995	168	97.640
6	GraalVM CE	x86_64	2,136,435.392	168	99.700
7	OpenJDK HotSpot VM	x86_64	2,217,293.347	168	103.474

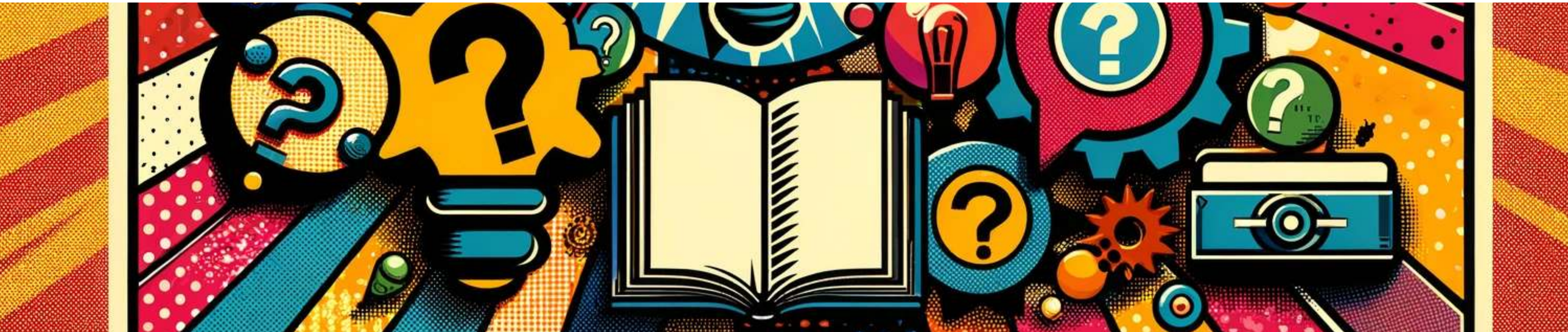


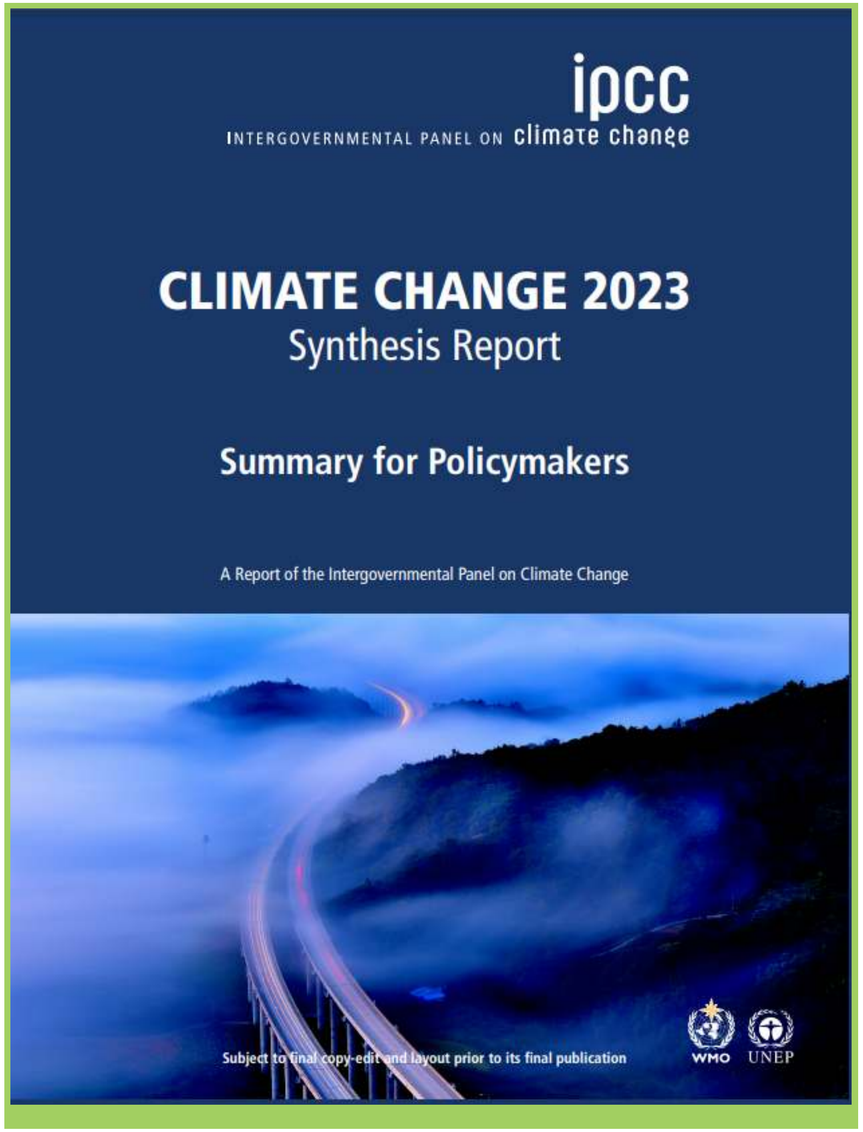
JoularJX for Java

The build energy consumption is one aspect to consider, but we also need to evaluate runtime efficiency, which includes both energy consumption and performance. In terms of runtime efficiency, the Oracle GraalVM Native Image with PGO enabled showed the highest efficiency. While its normalized energy consumption score was relatively moderate compared to other JVMs, it delivered the best overall normalized throughput across all tested use cases.

Oracle GraalVM Native Image without PGO, Azul Prime VM, Oracle GraalVM, GraalVM CE, and OpenJDK HotSpot VM exhibited similar efficiency in terms of power consumption versus performance, with marginal differences. Eclipse OpenJ9 VM, on the other hand, also demonstrated moderate overall energy consumption but exhibited the lowest performance in terms of throughput.

Thoughts with which I would like to leave you





Climate change is a real problem

“Premature optimization is the root of all evil." **Sir Tony Hoare**

What Hoare and Knuth are really saying is that software engineers should worry about other issues (such as good algorithm design and good implementations of those algorithms) before they worry about micro-optimizations such as how many CPU cycles a particular statement consumes.

It is interesting to look at how some software engineers have perverted those good software engineering concepts to avoid the work associated with writing efficient code.

“Premature optimization is the root of all evil." **Sir Tony Hoare**

Developers time more valuable than servers time?

In practice, what we can do - write efficient applications

Jevons paradox

charging station in Skei, Norway. The country has the world's highest rate of electric car adoption. | Sean Gallup/Getty Images

FUTURE PERFECT

Why Norway — the poster child for electric cars — is having second thoughts

Electric cars are crucial, but not enough to solve climate change. We can't let them crowd out car-free transit options.

By David Zipper | Oct 31, 2023, 7:00am EDT

Materials used in the presentation

- Energy Efficiency across Programming Languages
- Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model
- Google: Operating on 24/7 Carbon-Free Energy by 2030
- The Carbon Footprint of Machine Learning Training Will Plateau, Then Shrink
- 2023 State of Green Software
- GHG Protocol
- Web Sustainability Guidelines (WSG) 1.0
- Software Carbon Intensity (SCI) Specification Project
- The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations
- Analyzing JVM Energy Consumption for JDK 21: An Empirical Study

Thank You!

@ArturSkowronski

jvm weekly

VIRTUSLAB