Carbon, Power, and Sustainability in ATLAS Computing

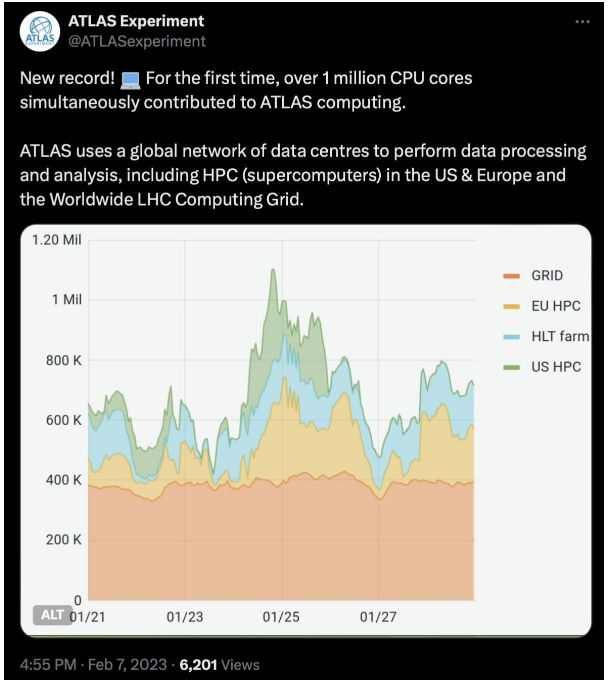
WLCG Environmental Sustainability Workshop

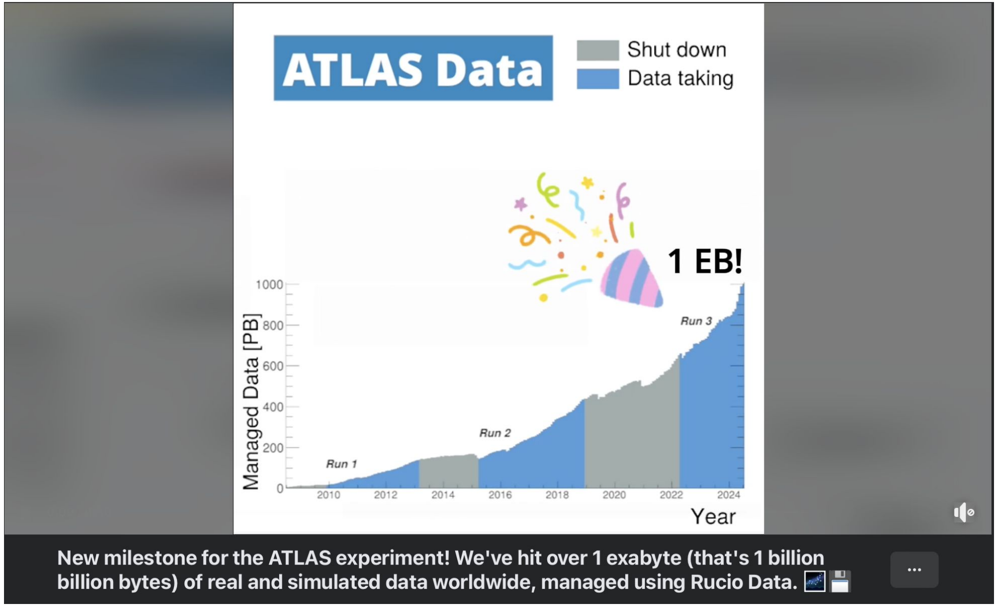
12 December 2024

Zach Marshall (LBNL) on behalf of the ATLAS Computing Activity

ATLAS computing resource scale

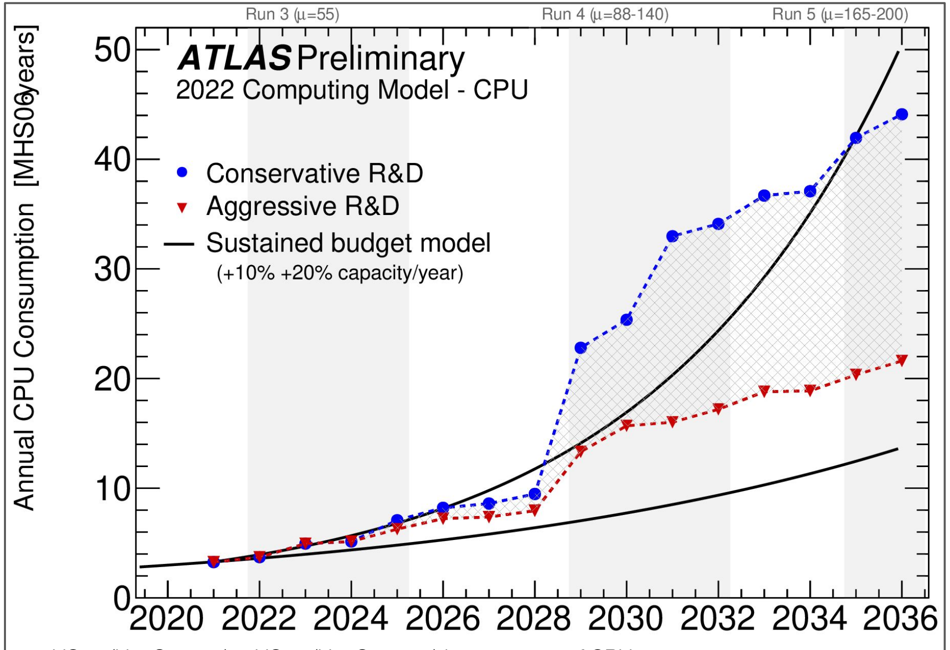
- ATLAS operates ~700k cores of compute (peaks at 1M cores), 400 PB of disk, 650 PB of tape
- Distributed over ~100 sites worldwide (much more info)
- Combination of high-throughput computing centers (WLCG sites), high-performance computing (HPC) systems, cloud computing (e.g. Google, AWS), volunteer computing (ATLAS@Home)
    - Peaks mostly from large HPC systems (not HEP-specific systems) like Vega, Perlmutter, etc



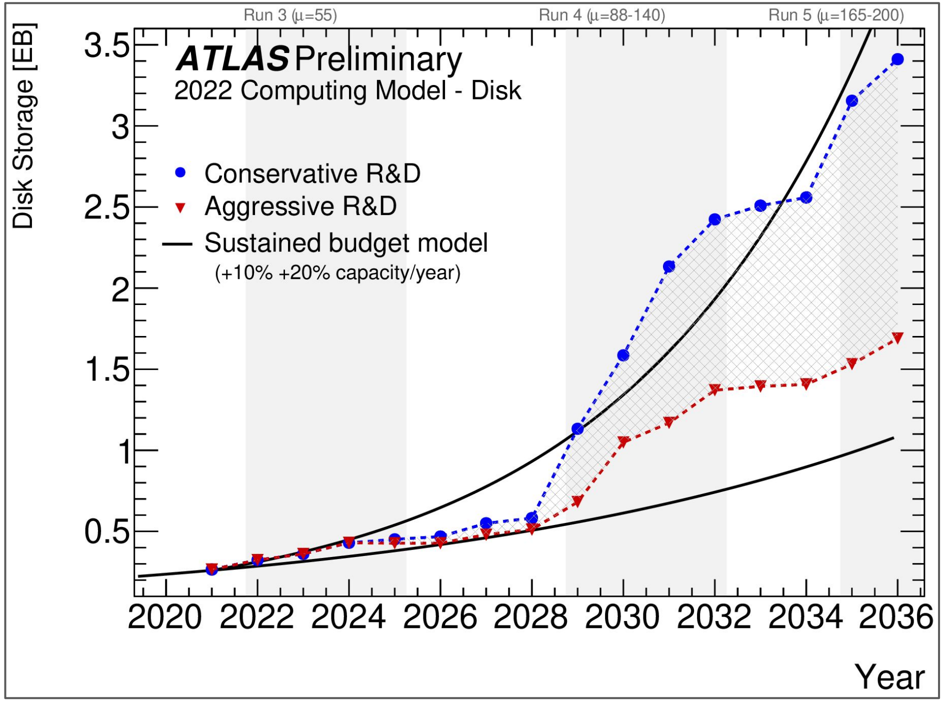


Computing resource extrapolation

- As a part of the HL-LHC upgrades, computing resources required to expand significantly
- 3–5x compute, disk, and tape by 2031; another >2x by 2041
- Requests for more resources made 12–18 months in advance
    - In-year, Jevons paradox applies; over several years, opportunity for reduction through optimization



HS06 (HepSpec06) = HS23 (HepScore23) is a measure of CPU power.
A modern CPU core is ~12 HS23



Building awareness

- Summary sent when a user’s grid job finishes
- Now includes an **average** carbon estimate
  - Links to a page explaining that number
  - Along with some background and some equivalents for comparison
  - More info later today from Fernando et al.
- Averaged for several reasons
  - Inaccuracy of and missing site-specific data (work ongoing to improve these data)
  - CPU doesn’t sit idle (moving a job would not reduce the experiment’s total footprint)
  - Users pushing on a single site could cause other problems (including waste)
  - Faster code → lower carbon footprint
- **Not** intended for “shame”; a reminder that Grid use is relevant to the environment
  - Code optimization and success rate matter!
- Also tracked for production campaigns and reported back to production managers →

PanDA

Summary of TaskID: 41161761

Detail	Value
Created	2024-09-11 12:21:24
Ended	2024-09-11 12:22:30.624502
Final Status	done

Total Number of Inputs

Category	Count
Succeeded	1
Failed	0
Cancelled	0
Total	1

Error Dialog: None

Datasets

In	[]
Out	[]
Log	['user.zmarshal.test.4a25be90-fb43-4d42-bdde-854c86b41bea.9-x86_64-bash-pythonrecommended-x509.log/']

Parameters

Command	prun --exec=myPunJob.sh --outDS=user.zmarshal.test.4a25be90-fb43-4d42-bdde-854c86b41bea.9-x86_64-bash-pythonrecommended-x509 --noBuild

Estimated Carbon Footprint for the Task

Category	gCO2
Succeeded	0.01 gCO2
Failed	0 gCO2
Cancelled	0 gCO2
Total	**0.01 gCO2**

More details on estimation: https://panda-wms.readthedocs.io/en/latest/advanced/carbon_footprint.html

Panda Problem Reporting

- The eGroup for help request: hn-atlas-dist-analysis-help@cern.ch
- The Discourse forum for distributed computing help: https://atlas-talk.web.cern.ch/c/distributed-computing-help
- The JIRA portal for software bug: https://its.cern.ch/jira/browse/ATLASPANDA

ATLAS PanDA

120 tasks listed. Estimated CO_2 emission in total: 2 t, including 129 kg by failed jobs.

Reminders Early On

- Now discussed in the ATLAS analysis tutorial
- How to run a grid job (was always there)
- How to run and examine a *failing* grid job
  - We still find that users are not familiar with what goes wrong on the grid, and have considerable difficulty debugging failing jobs
  - We have found cases 'in the wild' of hundreds of retries of failing jobs
- How to look at the carbon impact of your (failing) grid jobs
  - Again, meant for information and reflection, not for shame — and as encouragement to test jobs carefully!

ATLAS Software Documentation | Athena | Analysis software | Software infrastructure | Internal links

A Simple Grid Job

Last update: 23 Aug 2024 [History] [Edit]

The PanDA client package contains a number of tools you can use to submit and manage analysis jobs on `PanDA`.

While `pathena` is used to submit Athena user jobs to `PanDA`, more general jobs (e.g. ROOT and Python scripts) can be submitted to the grid by using `prun`.

Finally, `pbook` is a python-based bookkeeping tool for all `PanDA` analysis jobs.

Detailed information about each of the tools can be found in the above page.

Setup

If you are working on lxplus, the client is already installed, so to use it you only need to do:

```
setupATLAS
lsetup panda
```

Here we have set up the cvmfs software environment and asked to set up the Panda Clients.

To keep your grid work consolidated, create a new directory within `tutorial` called `GridTutorial`:

```
cd tutorial
mkdir GridTutorial
cd GridTutorial
```

Run a 'Hello World' job with prun

From your `GridTutorial` directory, create a new directory for a simple `prun` test and navigate into the new directory:

⭐ Bonus Exercise 2

Under normal circumstances, you should not send any jobs to the Grid that do not work. For this one case, to see what happens, try modifying your HelloWorld program to crash. For example, you can change it to:

```
#!/usr/bin/env python3
print("Hello world!")
Hello World!
```

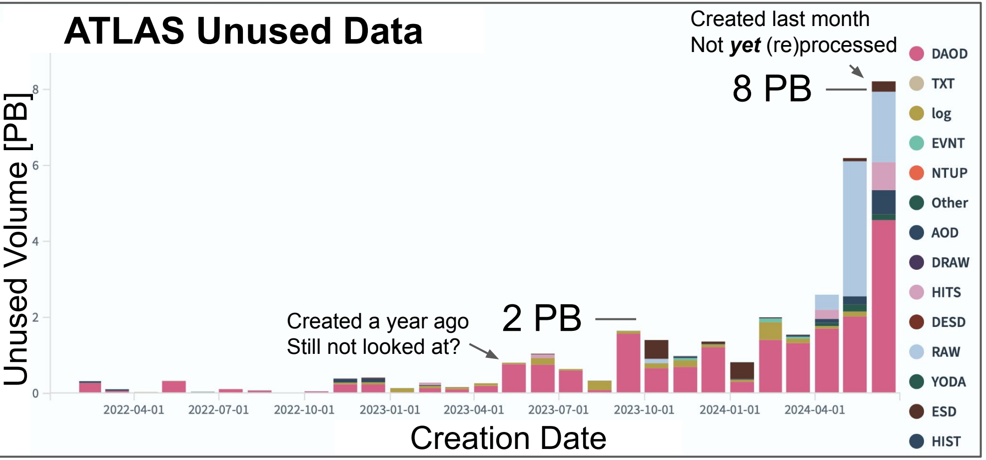
💡 On the task monitoring page, as well as in the email you get from PanDA when your task is complete, you will notice it reports a carbon footprint for your job. You can read more about that here. This particular failure probably used as much carbon as about 30 seconds of breathing, so don't worry too much about it. Still, this is a good reason to test your jobs carefully, learn how to optimize your code, and generally avoid wasting CPU on the grid!

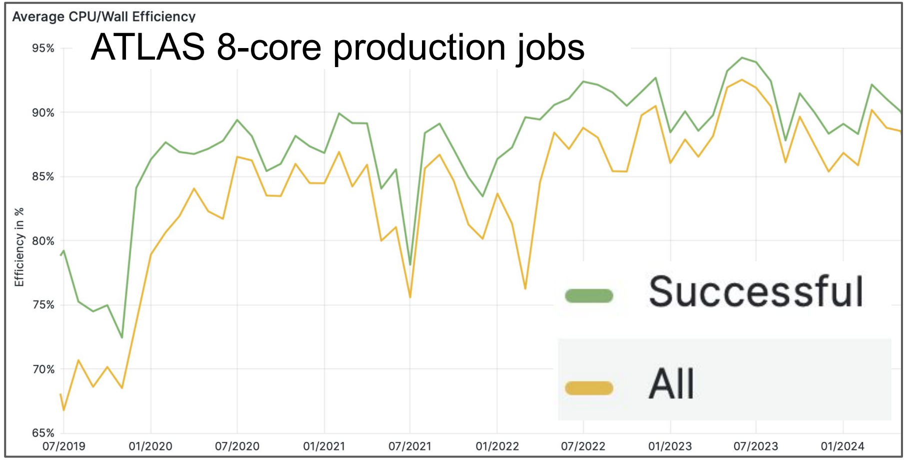
Computing model, policies, and tools

- Many aspects of the computing model affect carbon
  - Until recently, considered mostly in financial terms
  - Additional CPU is easier to find than additional storage; mostly attempts to reduce storage requirements
- Data carousel: use tape as a more active storage medium
  - Designed to reduce disk usage; also reduces carbon as tape is lower-carbon (in all ways) than disk
- Data reproduction: to save disk, reproduce little-used data when required
  - Able now to reproduce data on-demand, usually in <48h
  - Reduces need for archival storage, “just in case” storage; similar action with intermediate data formats
  - Being **reexamined with carbon in mind**
  - Very **rough** estimate suggests if our reproduction rate is <10%, we save carbon by **deleting** the data
  - Need to get some site numbers and understand from monitoring what the real rates are

Waste, loss, and unused data

- *Using* carbon for important science is "allowed" — *wasting* carbon is never ok!
- Constantly monitoring *unused data* in the production system
  - Requests made, bug found, reproduced before the data were looked at
  - Processing done based on a too-inclusive pattern (mc*)
- Steady progress to improve CPU/wall efficiency to over 90% (below for 8-core production jobs)
  - Impact of failed jobs is visible; errors on copying output (after all the CPU has been consumed) are killer!
  - Constant effort to reduce serial portions of many-core jobs as well (wasted CPU and power)





Job Failures

- Significant investigation of job failures recently as well
- Found that it is **very** important to de-correlate issues to avoid victim-blaming
    - A site might have special resources, and so be sent special tasks… that fail a lot. Not the site's fault!
    - The opposite is also true: our failure rate at the CERN Tier 0 is below 2%! But…
- Still, some very strong correlations that merit closer examination
- Also re-examining re-try policies
    - Automatic identification of jobs that will fail repeatedly so that they aren't retried
    - Automatic re-brokering of jobs that have site-failures so that they have a new chance to succeed

Resources utilisation and savings by HammerCloud

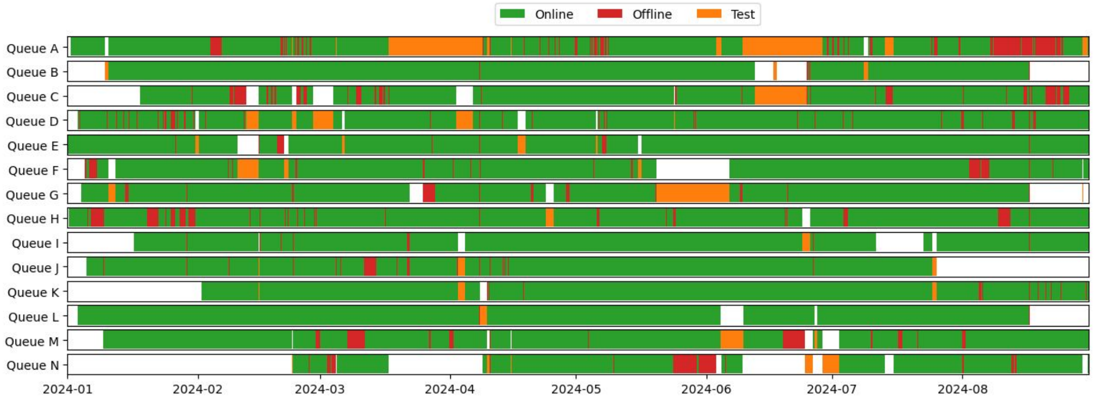
- HammerCloud is used by ATLAS to submit test jobs to grid sites
- HammerCloud consumes **0.15%** of the ATLAS Grid (Jan – Sept 2024)
- HammerCloud runs a variety of workloads:

  - **Analysis Functional Tests**
  - **Production Functional Tests**
  - **HEPScore23** measurements
  - **multi-core** functional tests
  - **Pilot** development tests
  - **ALRB** development tests

About ⅓ of processing is for Auto-exclusion
→ 0.05% of the ATLAS HS23 is used for Auto-exclusion testing

Resources utilisation and savings by HammerCloud

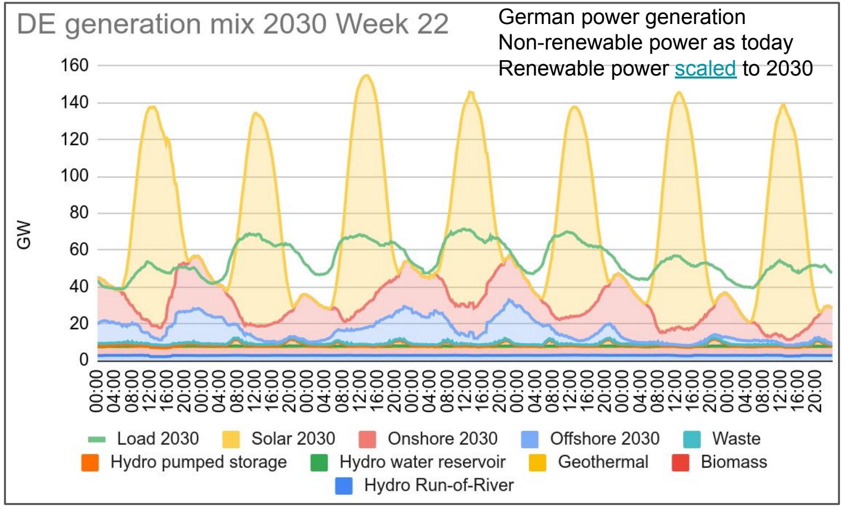
- Auto-exclusion **reduces waste** by **preventing failing sites** from running jobs
- Typical availability profile for PanDA queues:



- ~4.6% of the total time, queues are Auto-excluded
- Savings depend on the site (/configuration):
  - If the node is idle when excluded, **~50% energy savings**
  - If the node is used productively by another VO, **100% energy savings**
- Auto-excluding resources prevents 2–4% of wasted energy annually
  → Sustainability efforts in HammerCloud should focus on efficient, accurate Auto-exclusion

Software development for Sustainability

- Goes without saying that we should improve our software
- Some testing of **checkpointing** to see if we can pause processing during scheduled outages, peak load periods, and brown energy periods
  - Likely in the future to see periods of *very* green energy, and periods of less green energy
- Need to also improve **validation** to ensure that we don't produce buggy samples that must be replaced (ongoing effort in the collaboration to improve CI/CD/validation)



Collaboration for Sustainability

- Sustainability is not an ‘ATLAS problem’, or even an ‘LHC problem’
- ‘Sustainability’ focus sessions at ATLAS Software and Computing weeks since mid-2023
- Great success reaching out to people who are interested in this problem space
  - Dan Schien and Paul Shabajee at Uni Bristol
  - Varsha Rao and Andrew Chien at U. Chicago
  - Folks at the EE HPC WG, Lefdal Mine, Lancium
  - Got to also invite the LHCb folks and HEP sustainability group
- We’d be happy to engage more with others on these issues

Sustainability Summary and Future Work

- Effort ongoing to build a complete carbon footprint model for ATLAS Computing
  - Studies ongoing in storage footprint and storage carbon optimization, frequency scaling and checkpointing, networking aspects, cooling systems, site power consumption, platform adoption…
  - **Many of these studies are being shown at this workshop!**
  - Clear that limited models / information risk *harmful* recommendations
- Connected effort to make users, developers, operators, admins more aware of the environment
- Goal is to make useful recommendations for policies and to sites towards a more sustainable computing model as we approach the HL-LHC
  - What is the *carbon impact* of GPU usage? The payoff point in terms of GPU load? (30–60%?)
  - In carbon terms, what is the optimal storage configuration? (RAID? Background task configuration?)
  - How should we think about the tradeoff between *disk*, *tape*, *network,* and *CPU*?
  - What is the "optimal" approach to take towards old hardware?
  - Some of these should probably be **WLCG recommendations**
- Closely watching extrapolations that affect these recommendations
  - De-carbonization of the power grid will likely emphasize embodied carbon
  - Hardware landscape is constantly evolving (ARM, RISC-V, GPUs, Grace Hopper, …)
  - Improved chip performance (flops/W) is coming with larger packages and less modularity and repairability — potential impacts on site operations and hardware lifetimes

Thank you!

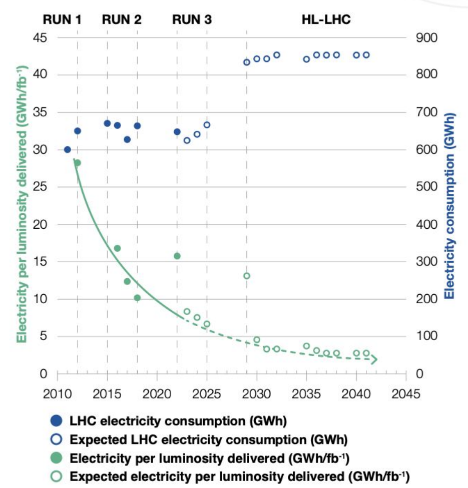
And thanks to Michael Boehler, Alex Lory, John McGowan, and Rod Walker for some of the plots and material

BERKELEY LAB

ATLAS EXPERIMENT

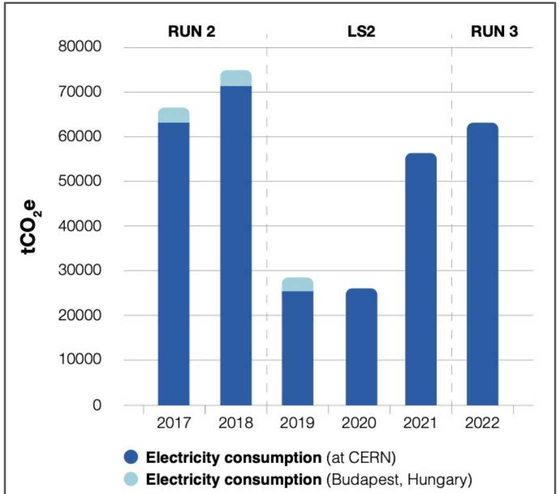
CERN Environmental Reports

- Much more attention to carbon footprints recently (CERN reports in 2021 and 2023)
- Computing is ~5% of CERN's footprint when the LHC runs (accelerator cooling ~80%)
  - CERN mostly draws power from the French (nuclear, quite green) grid
  - Data center in Hungary that ran until 2019 had a visible impact on energy consumption (now moving into the new Prevessin Data Center)



ELECTRICITY INTENSITY OF THE LHC

Quantity of electricity used to run the LHC per unit of luminosity delivered, showing that less and less electricity has been needed over time to produce the same amount of data and, hence, scientific output. During the year after each long shutdown, while the machines are being brought back online and progressively ramped up, the luminosity delivered is not at its maximum – as seen in 2022 and 2029.

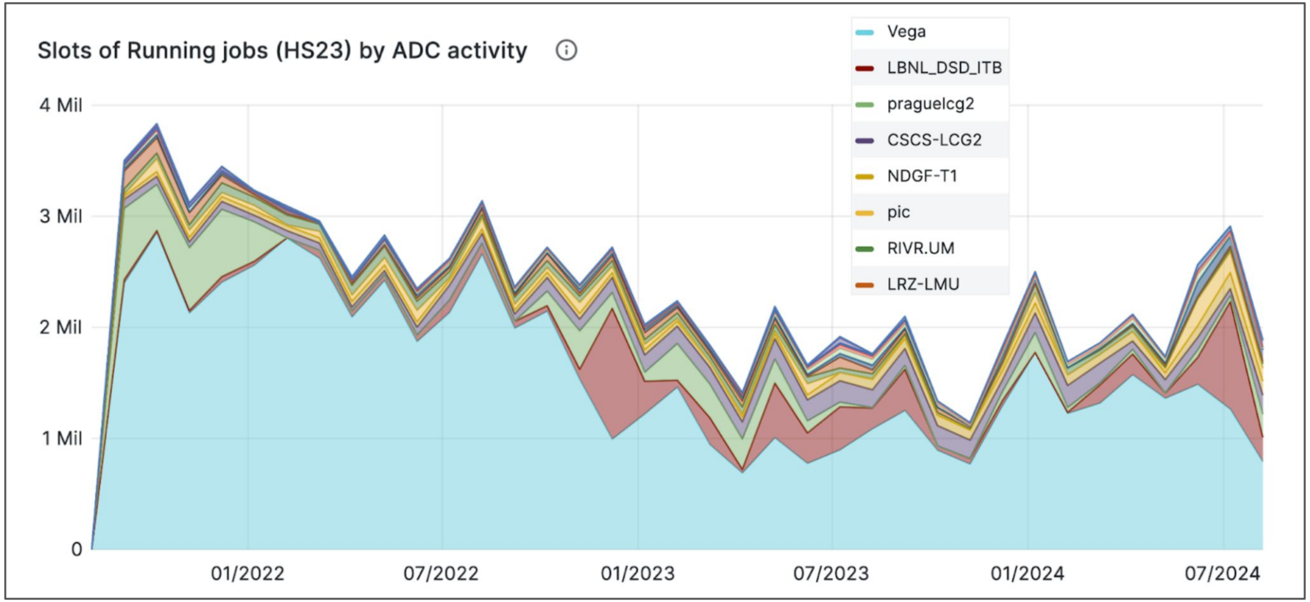


CERN SCOPE 2 EMISSIONS FOR 2017–2022

Emission calculations for electricity follow a location-based methodology, with average yearly emission factors taken from ADEME Base Empreinte©. From 2017 to 2019, CERN operated a data centre at the Wigner Centre in Budapest, Hungary, for which the emissions are also shown. The location-based emission factors used for Hungary were taken from Bilan Carbone® V8.4.

High Performance Computing Systems

- These systems would likely be built with or without us there
  - Their use, in some sense, ‘saves’ carbon worldwide
- With good software, we can fill the ‘gaps’ in these systems (and other systems with similar running profiles) and improve their usage fraction — a win for everyone



New data centers

- Most important to data centers is Power Usage Efficiency (PUE)
  - Power into a data center ÷ power to IT elements; 1.1 is *good*; 1.5+ is *typical*
  - Operational carbon is (Power to IT elements) * (PUE) * (Power Grid Carbon Intensity)
- Steel / concrete buildings (like data centers) typically cost 200–800 kgCO_2eq/m^2 of space (1,2)
- We can then write down how long it takes for a new data center to “pay off” its carbon footprint
- Taking typical examples (500 kgCO_2eq/m^2, 10,000 m^2, 5 MW data center):

In California (0.12 kgCO_2eq/kWh) changing PUE from 1.2 to 1.1:

~12 years to pay off

At CERN (0.075 kgCO_2eq/kWh) changing PUE from 1.6 to 1.1:

<5 years to pay off

- In other words: you should **almost certainly** build a new data center!
  - Also: physical buildings are likely O(10%) of the total carbon footprint of computing
- Water cooling and waste-heat usage can reduce an effective PUE to ~0.6
  - Effective PUE: (power into data center – power *out of* data center) ÷ power to IT elements
  - These are **obvious** additions that any new data center should incorporate if close to a population!