

You already have what you need

Open Source GreenOps in Practice

GreenIO London - October 2026

Julien Nioche

Diana Todea

digital**Pebble**



VictoriaMetrics





How to get **actionable** data about impact of cloud usage?

Hint: not from the cloud providers

Example: AWS Sustainability Console

- Good progress in the last 18 months but...
- Min granularity is a month (year for water)
- Services: 4 values only (EC2, S3, CloudFront, other)
- Opacity of data (carbon intensity factors? energy?)
- Can't see user tags → can't attribute impacts
- No actionable insights

⇒ **Not a tool for greenops**



Open Source?

Cloud Carbon Footprint



- Hard to set up
- Slow
- No user tags
- Aging data and methodology
 - nothing for AI or water usage
- No active development
- Dropped by Thoughtworks

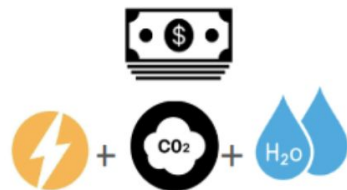


SPRUCE

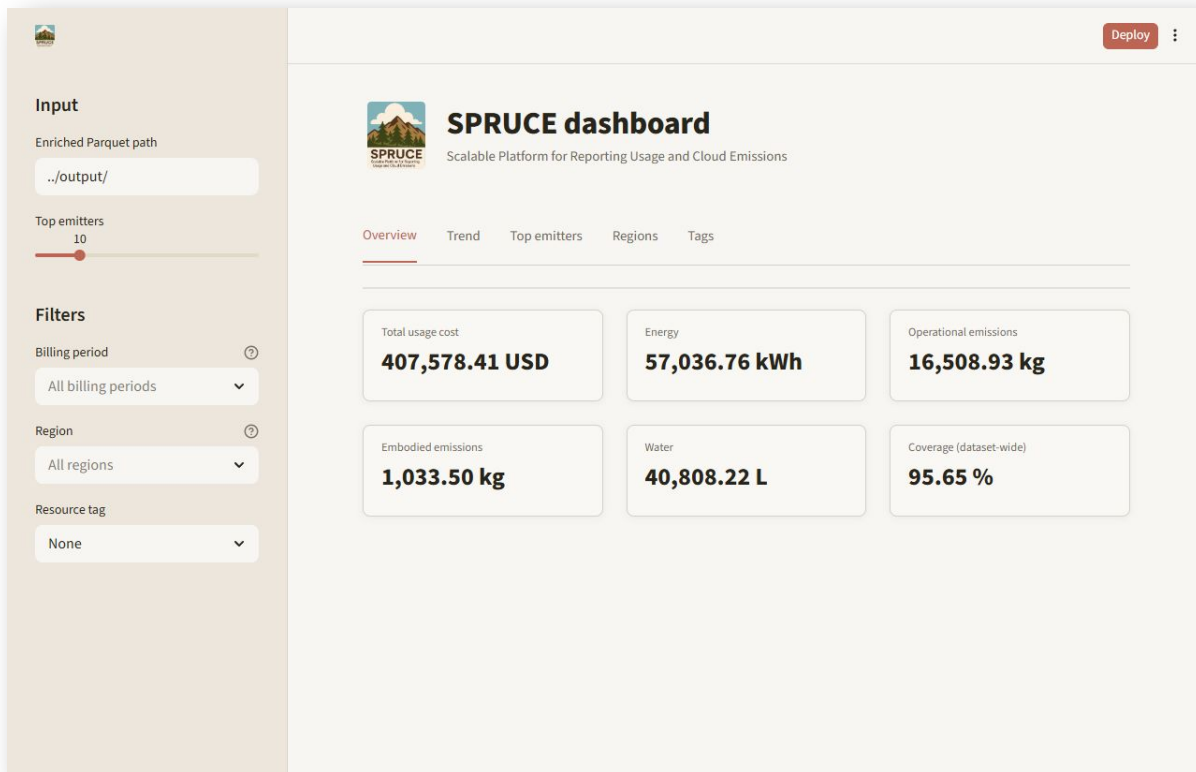
Scalable Platform for Reporting
Usage and Cloud Emissions

SPRUCE

- <https://github.com/DigitalPebble/spruce>
- <https://opensourcegreenops.cloud/>
- Enrichment pipeline for cloud usage reports
 - AWS + Azure (native and FOCUS)
- Combines third-party data and libraries
- Open Source (Apache License 2.0) 🌸
- v1.3 (15 releases in ~1 year)
- Covers >80% of the costs for a typical user



Dashboard



Static reports(html, pdf, md)

Top Emitters by Service

product_code	service_code	operation	co2_usage_kg	energy_kwh	co2_embodied_kg	water_usage_l
AmazonEC2	AmazonEC2	FargateTask	4108.45	9279.53	None	25588.6
AmazonEC2	AmazonEC2	RunInstances	3285.43	6690.29	1036.26	18697.0
AmazonS3	AmazonS3	GlacierInstantRetrievalStorage	1780.19	2946.98	None	7725.34
AmazonEC2	AWSDataTransfer	RunInstances	1411.0	2848.33	None	7967.46
AmazonS3	AmazonS3	OneZoneIAStorage	654.23	1317.62	None	3686.18
AmazonS3	AmazonS3	StandardStorage	551.49	1133.16	None	3161.07
AmazonS3	AWSDataTransfer	GetObject	280.72	663.11	None	1818.96
AmazonEC2	AmazonEC2	CreateVolume-Gp3	233.57	487.85	None	1362.71
AmazonEC2	AmazonEC2	RunInstances-SV001	212.66	428.31	13.4	1198.24
AmazonEC2	AWSDataTransfer	FargateTask	183.41	370.89	None	1037.04
AmazonDocDB	AmazonDocDB	CreateCluster	128.21	258.21	None	722.36
AmazonS3	AWSDataTransfer	UploadPart	97.54	196.73	None	550.27
AmazonEC2	AWSDataTransfer	InterZone-In	93.16	187.29	None	523.69
AmazonEC2	AWSDataTransfer	InterZone-Out	91.38	183.69	None	513.63
AmazonEC2	AmazonEC2	CreateVolume-Gp2	53.65	126.83	None	348.18
AWSELB	AWSDataTransfer	LoadBalancing	48.52	113.35	None	311.32
AmazonS3	AmazonS3	IntelligentTieringIAStorage	44.65	89.93	None	251.58
AmazonS3	AmazonS3	StandardIAStorage	24.99	50.33	None	140.81
AmazonEC2	AmazonEC2	CreateSnapshot	20.28	48.05	None	136.02
AmazonECR	AmazonECR	TimedStorage-ByteHrs	19.9	40.07	None	112.11

Top Instance Types

instance_type	co2_usage_kg	co2_embodied_kg	water_usage_l
i3en.2xlarge	1002.08	794.66	5646.13
g5.2xlarge	581.92	23.39	3278.79
g1e.2xlarge	531.84	15.01	2996.7

Query with Athena / DuckDB

```
select line_item_product_code, product_servicecode,  
       round(sum(operational_emissions_co2eq_g),2) as co2_usage_g,  
       round(sum(energy_usage_kwh),2) as energy_usage_kwh  
from enriched_curs where operational_emissions_co2eq_g > 0.01  
group by line_item_product_code, product_servicecode order by co2_usage_g desc;
```

line_item_product_code	product_servicecode	co2_usage_kg	energy_usage_kwh
AmazonEC2	AmazonEC2	5411.49	17501.57
AWSELB	AWSDataTransfer	1.82	5.67
AmazonS3	AWSDataTransfer	1.42	4.6
AmazonEC2	AWSDataTransfer	0.7	2.36
AmazonECR	AWSDataTransfer	0.07	0.28



HELP WITH SQL?

Sustainable open source

- Aspiration ->



- Build a healthy community of contributors
 - No reliance on a single individual or organisation
- Users!
- Be part of a wider ecosystem
 - Don't reinvent the wheel
 - Contribute back

Ecosystem



Why SPRUCE?

- It's free — no procurement gate
- Your data does not have to leave your environment
- Open methodology: numbers you can understand and challenge
- Gives you a reference point when you do evaluate commercial vendors

Why SPRUCE?

- Tooling is not the main problem, it is what you do with the data
 - How you implement a greenops programme in your organisation
- SPRUCE is an enabler
 - Gets you to the real problems quicker
- Showcase the value of GreenOps
 - Early engagement, early wins



Observability

Building a Green O11y pipeline

- Why Green telemetry is important?
- Result: create a workflow where cloud cost, operational energy, carbon emissions, water consumption and other sustainability indicators become first-class observability signals.

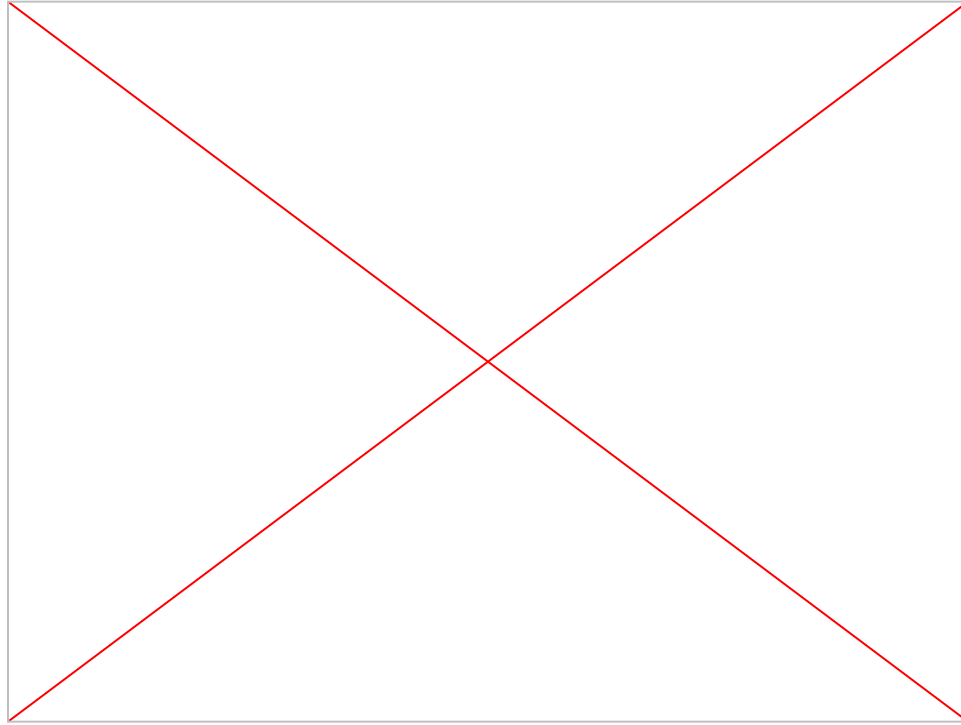
Architecture overview

- Azure FOCUS Export -> Billing data source
- SPRUCE -> Normalization + sustainability enrichment
- Python exporter -> Prometheus metric generation
- VictoriaMetrics -> Time-series storage
- Grafana -> Visualization



 <https://github.com/didiViking/greenops-observability-reference-architecture> 

Observe your Green Oily pipeline



GSF and OpenTelemetry



- The need for standardization is important in OSS and cloud-native
- Green Software Foundation is currently translating SCI (Software Carbon Intensity standard) to OpenTelemetry through semantic conventions
- Carbon as a new observability signal



 <https://github.com/Green-Software-Foundation/sci-otel> 



Thanks!

julien@digitalpebble.com

diana@victoriametrics.com

Who am I?



digitalPebble



Who am I?

