

Carbon-Aware computing

Reducing the carbon intensity of software

Anders Lybecker | @Lybecker



I am a principal engineering lead at Microsoft

Vestas





Challenges

- Wind Turbine Simulation
 - Optimal placement of turbines in windfarm
 - Structural optimizations
- Existing on-premise HPC cluster
 - Expensive to maintain
 - Industry growth at 300%
 - Complexity of turbine

Requirements Distilled



Scale to 400K
cores



Inexpensive



As green as
possible



The IT industry emits more carbon than the aviation industry

What can I as a software engineer do?

Table 4. Normalized global results for Energy, Time, and Memory

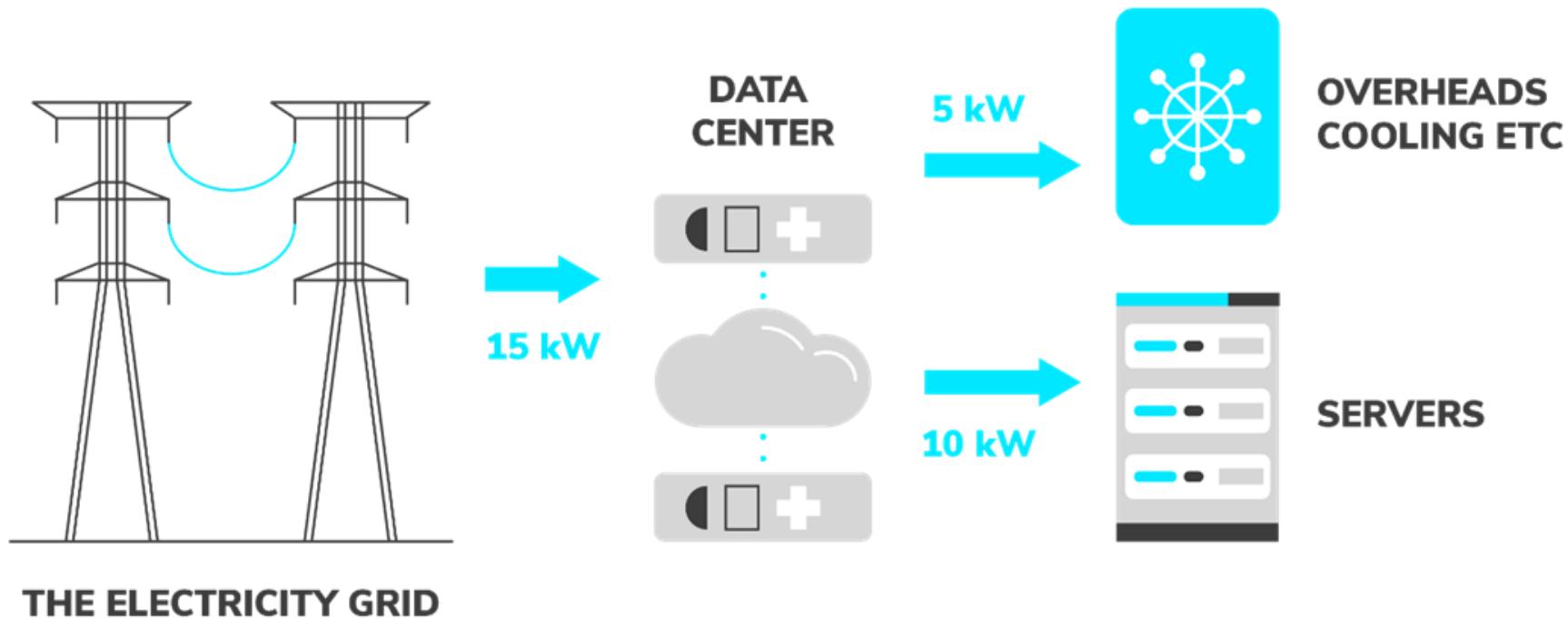
Total			
	Energy	Time	Mb
(c) C	1.00	1.00	1.00
(c) Rust	1.03	1.04	1.05
(c) C++	1.34	1.56	1.17
(c) Ada	1.70	1.85	1.24
(v) Java	1.98	1.89	1.34
(c) Pascal	2.14	2.14	1.47
(c) Chapel	2.18	2.83	1.54
(v) Lisp	2.27	3.02	1.92
(c) Ocaml	2.40	3.09	2.45
(c) Fortran	2.52	3.14	2.57
(c) Swift	2.79	3.40	2.71
(c) Haskell	3.10	3.55	2.80
(v) C#	3.14	4.20	2.82
(c) Go	3.23	4.20	2.85
(i) Dart	3.83	6.30	3.34
(v) F#	4.13	6.52	3.52
(i) JavaScript	4.45	6.67	3.97
(v) Racket	7.91	11.27	4.00
(i) TypeScript	21.50	26.99	4.25
(i) Hack	24.02	27.64	4.59
(i) PHP	29.30	36.71	4.69
(v) Erlang	42.23	43.44	6.01
(i) Lua	45.98	46.20	6.62
(i) Jruby	46.54	59.34	6.72
(i) Ruby	69.91	65.79	7.20
(i) Python	75.88	71.90	8.64
(i) Perl	79.58	82.91	19.84



Energy Efficiency

Consume the least amount of electricity possible

Power Usage Effectiveness



PUE 1.5

Design Decisions



Saturate CPU	✓	✓
Rightsizing VM	✓	✓
Auto scale up and down	✓	✓
Using unused VMs	✓	✓

Carbon Awareness

CO₂

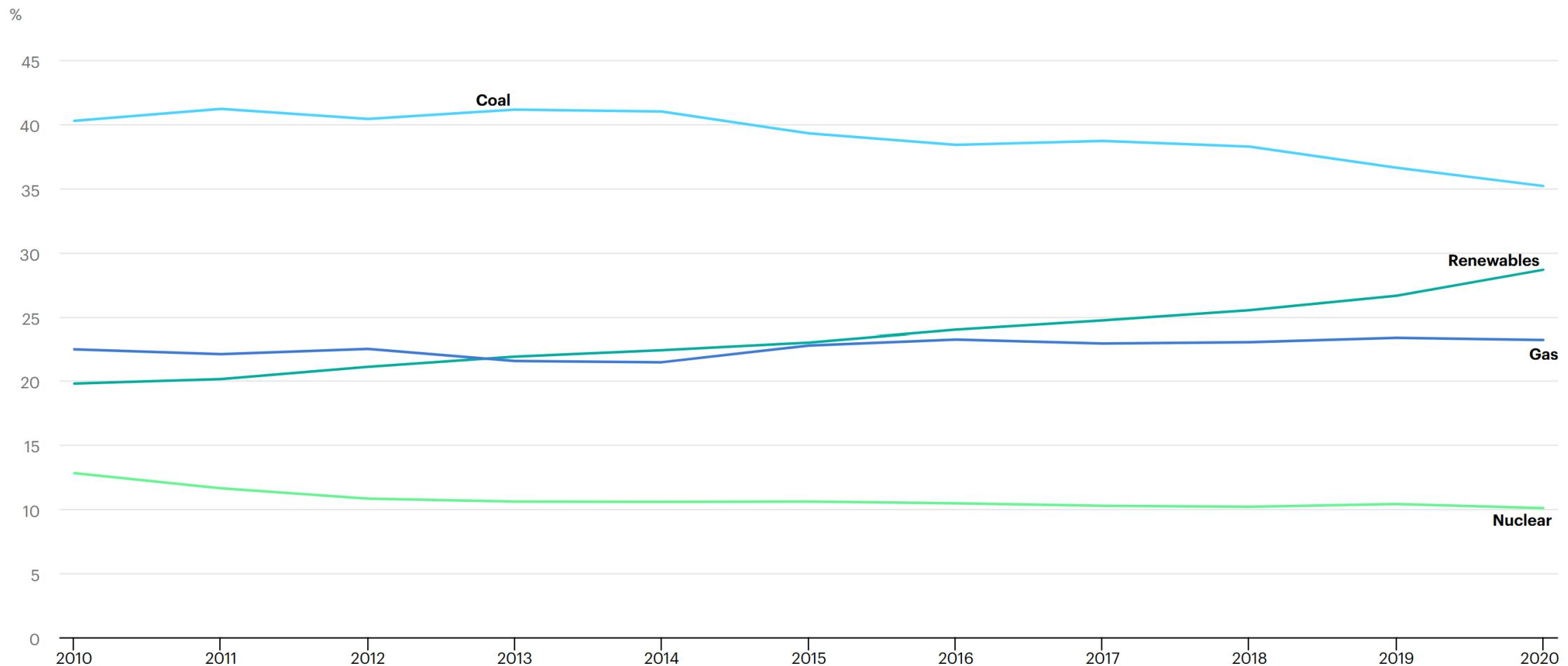
CO₂eq

Carbon intensity

CO₂eq/kWh

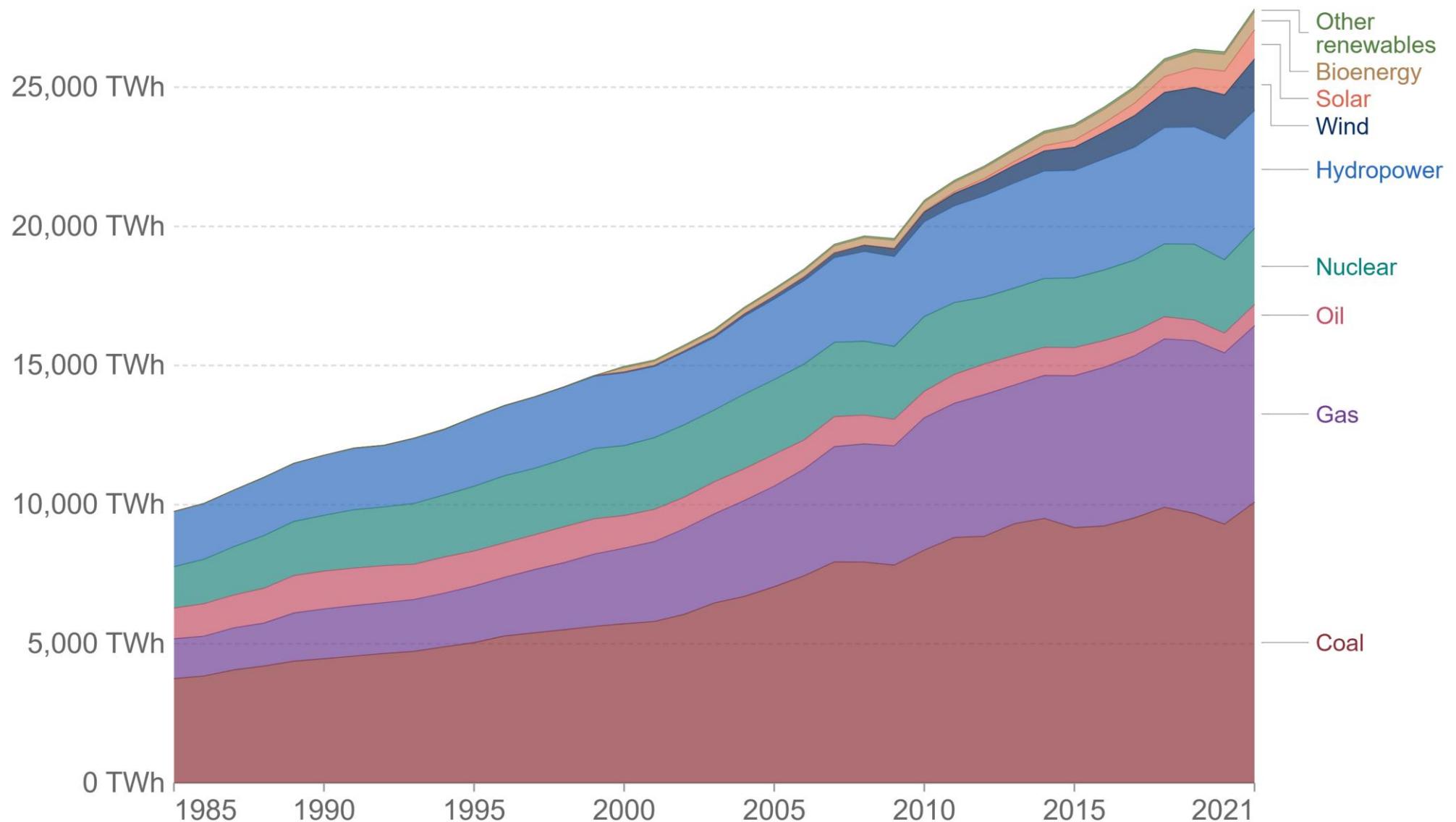
Electricity mix

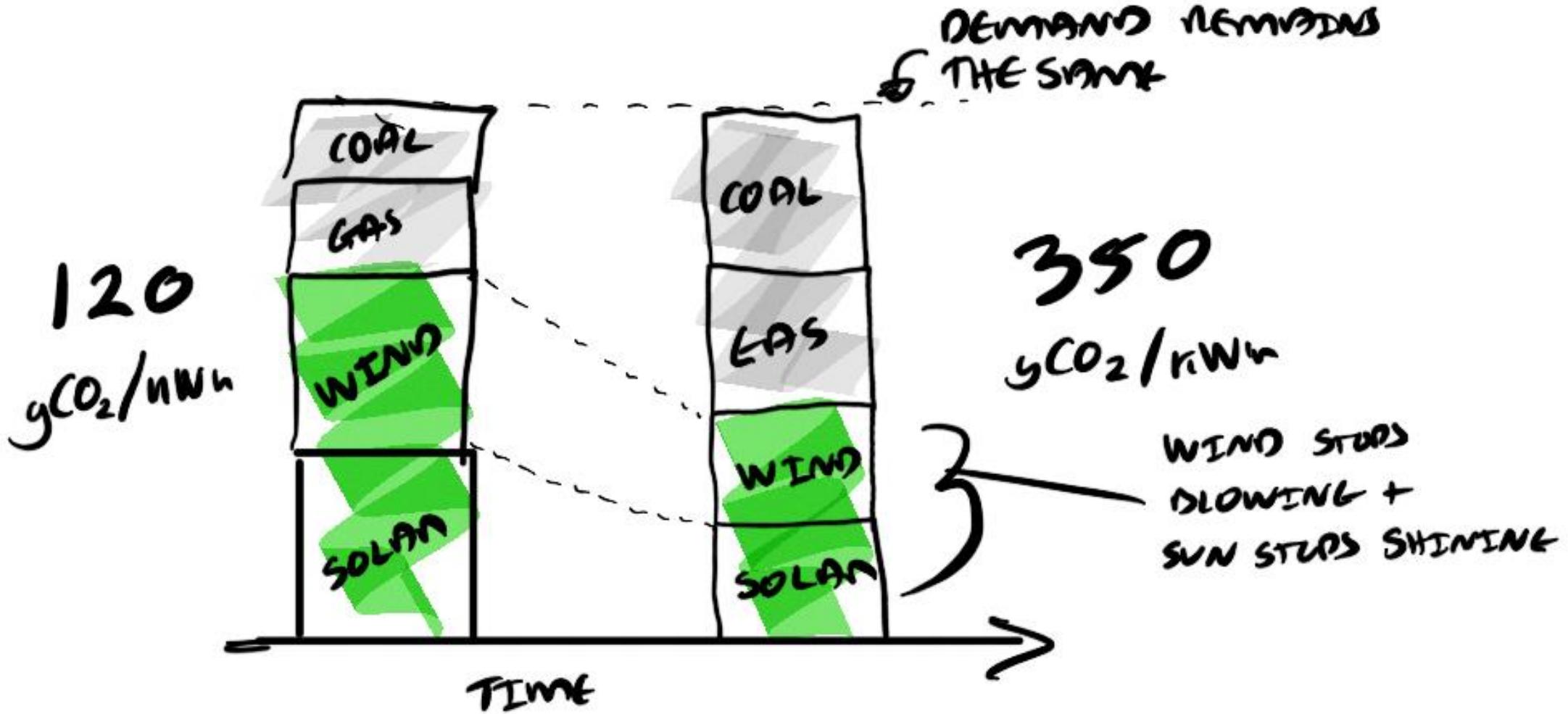
Electricity mix

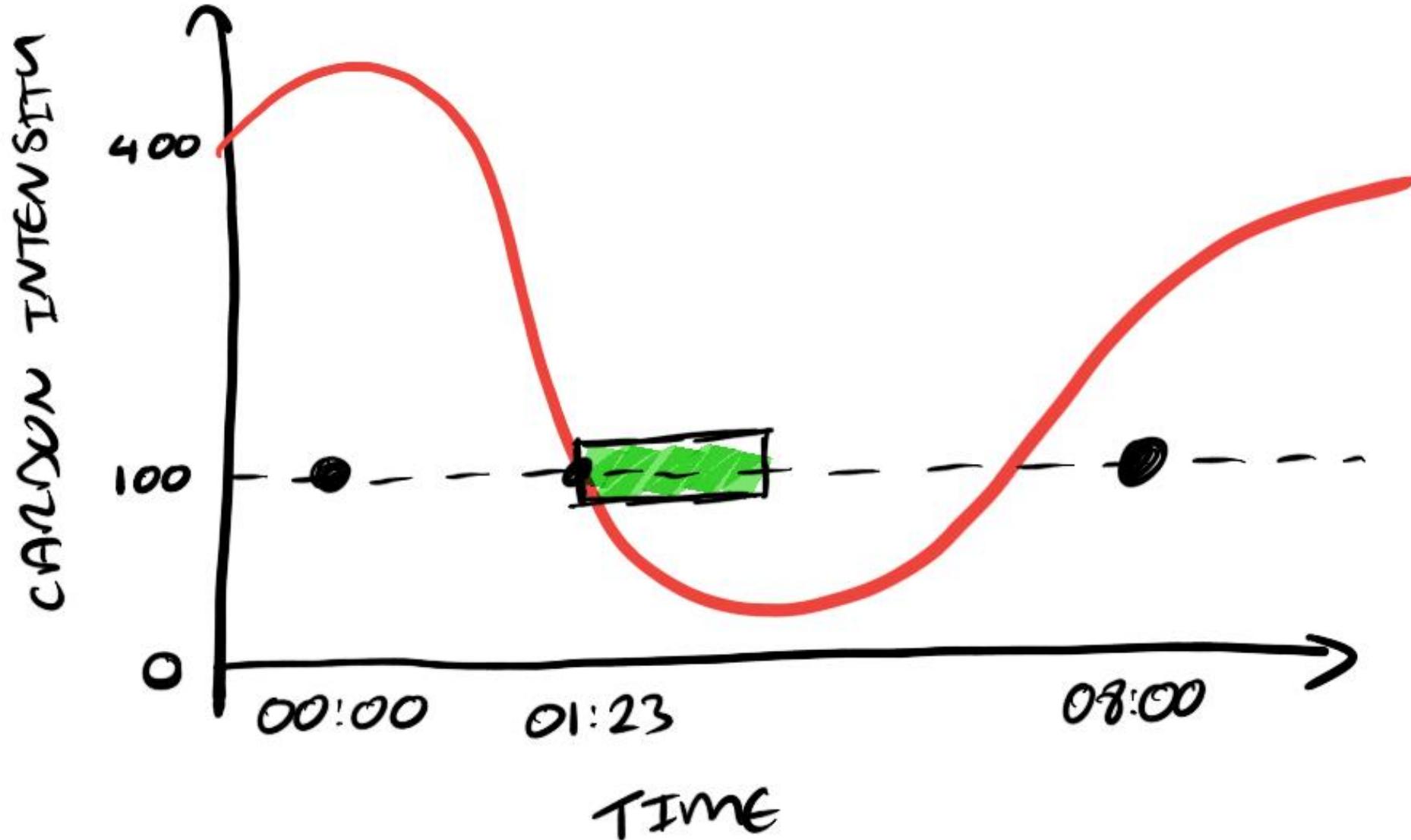


Source: International Energy Agency

Electricity mix

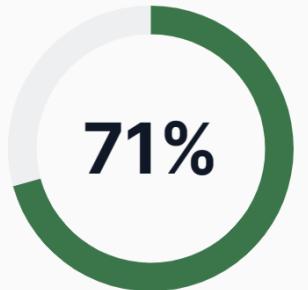
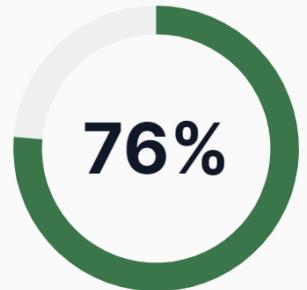






Netherlands

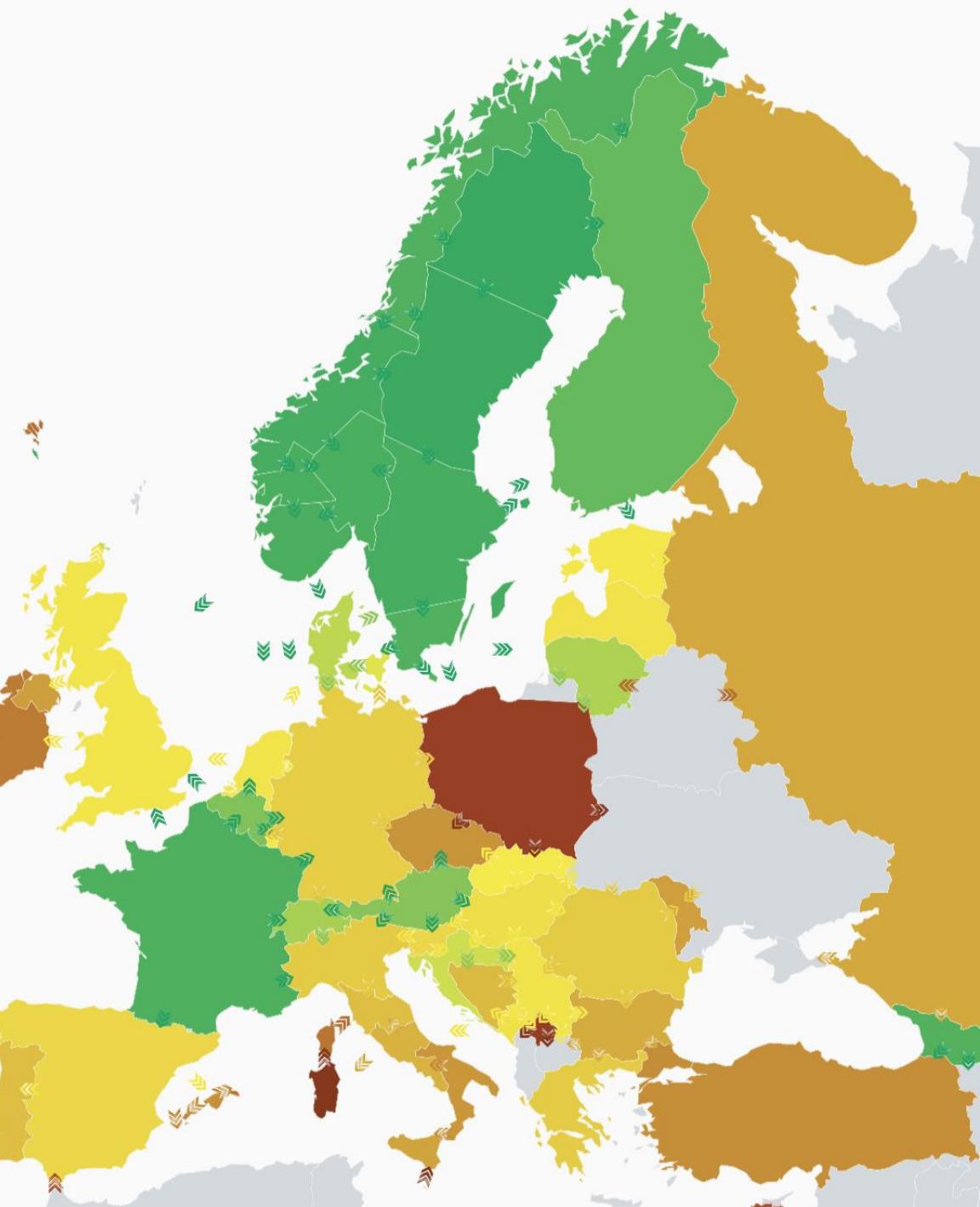
May 30, 2023, 10:00 AM



Carbon Intensity

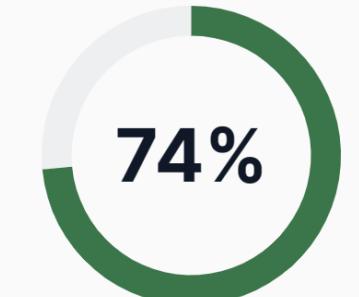
Low-carbon

Renewable



South Central Sweden

May 30, 2023, 10:00 AM



Carbon Intensity

Low-carbon

Renewable

Design Decisions



	Cost	Environment
Saturate CPU	✓	✓
Rightsizing VM	✓	✓
Auto scale up and down	✓	✓
Using unused VMs	✓	✓
Time shifting		✓
Location shifting	✓	✓

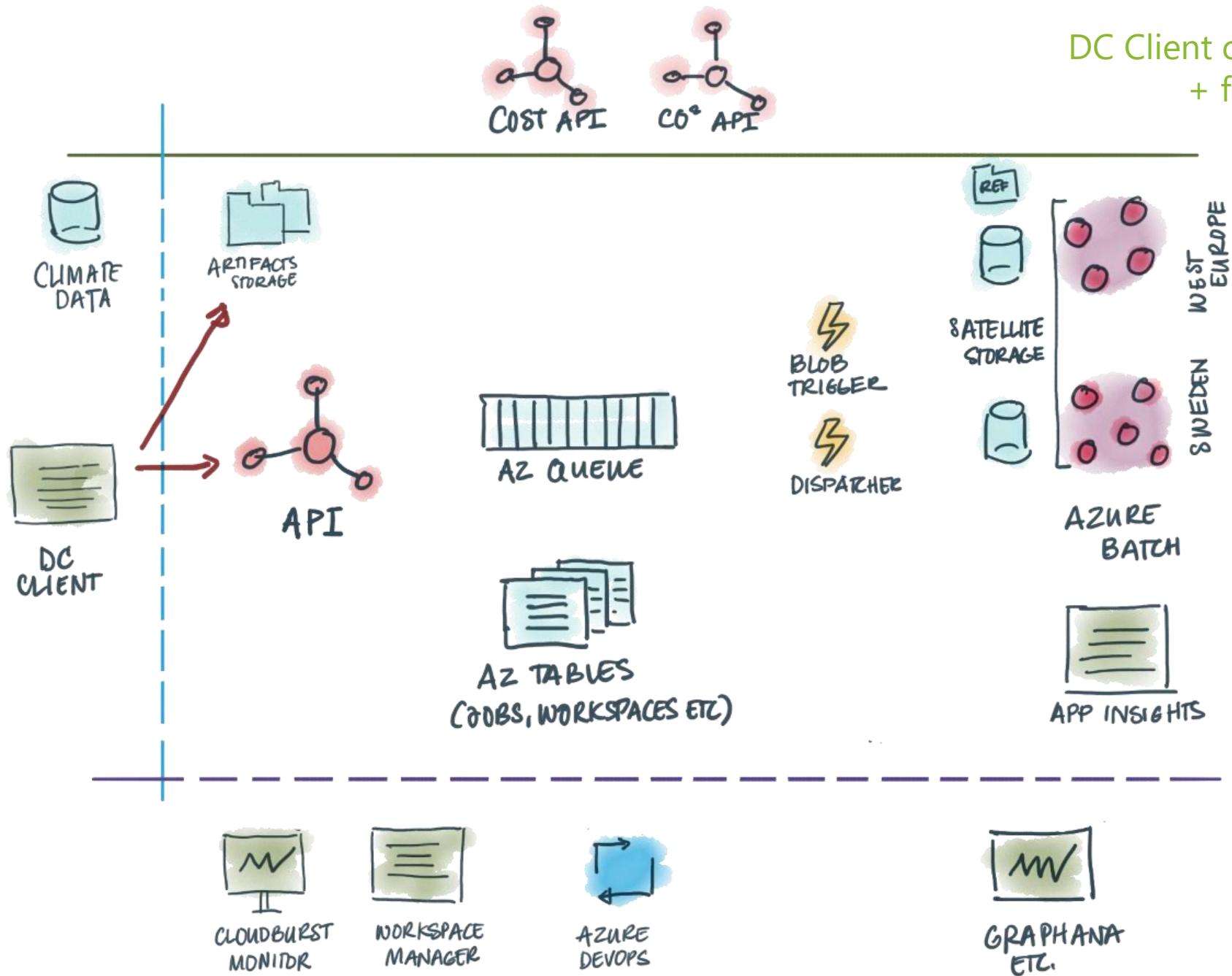


Time & location shifting

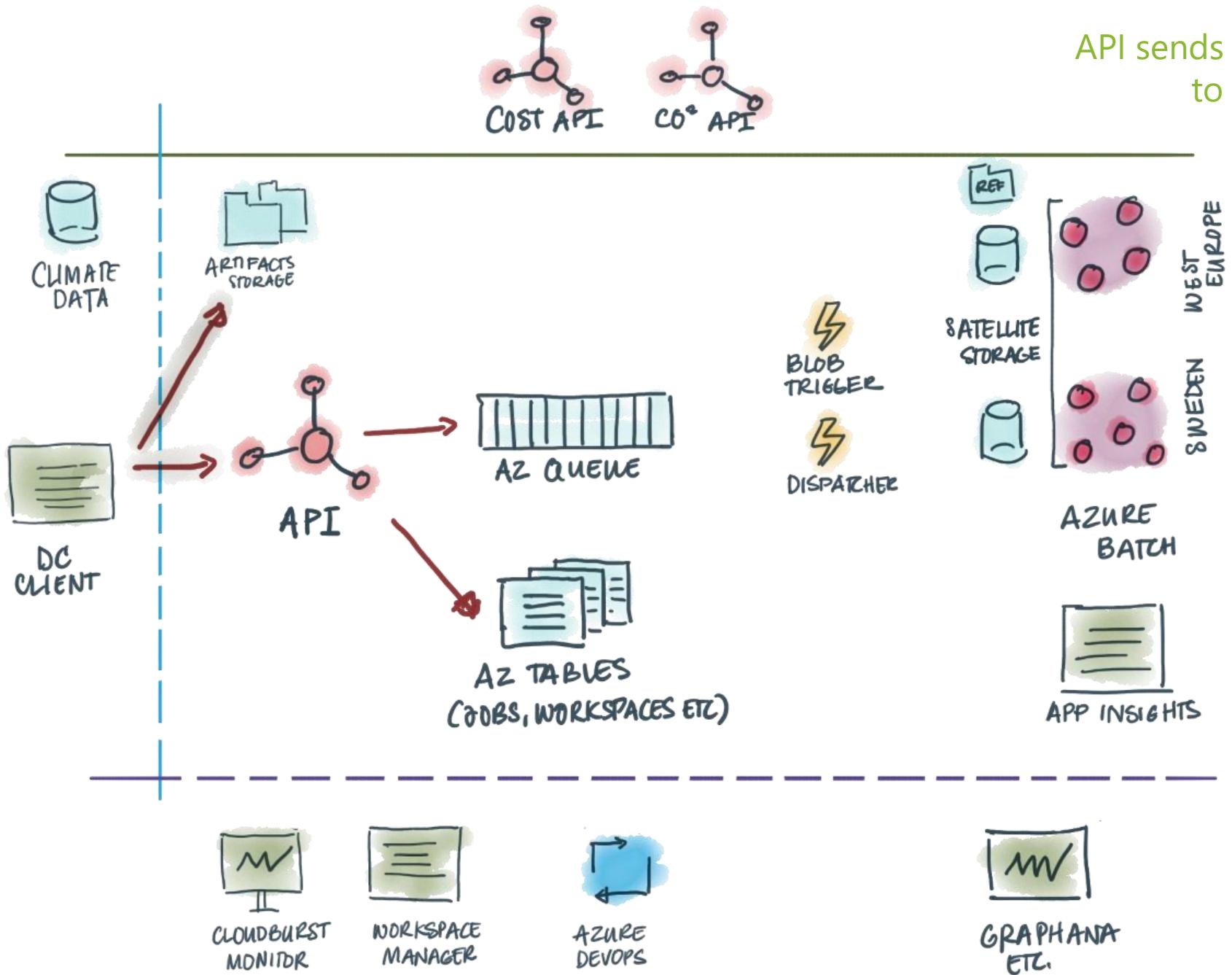
Coordination, data & time



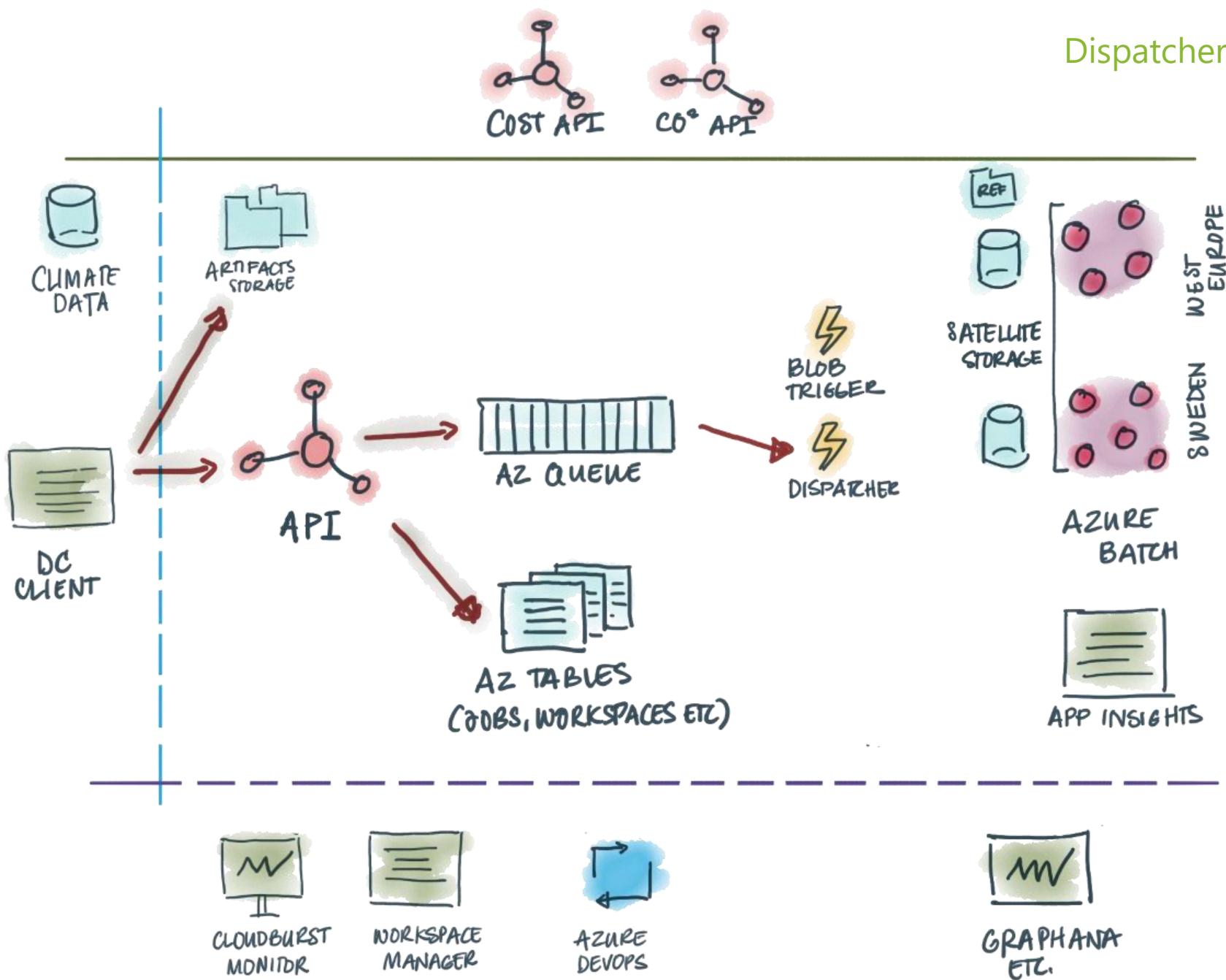
DC Client calls API with job spec
+ files in artifact storage



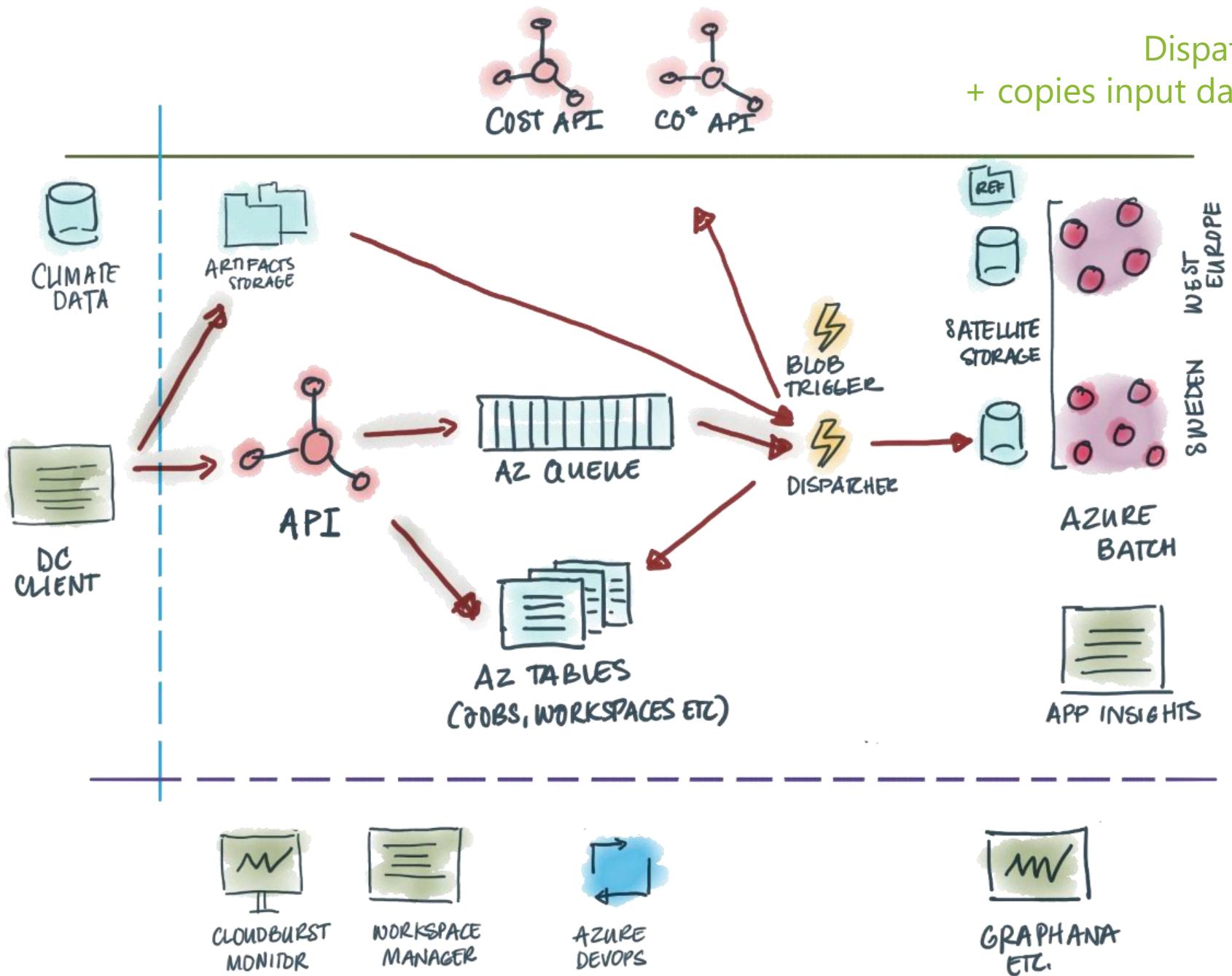
API sends a job create message
to the dispatcher queue



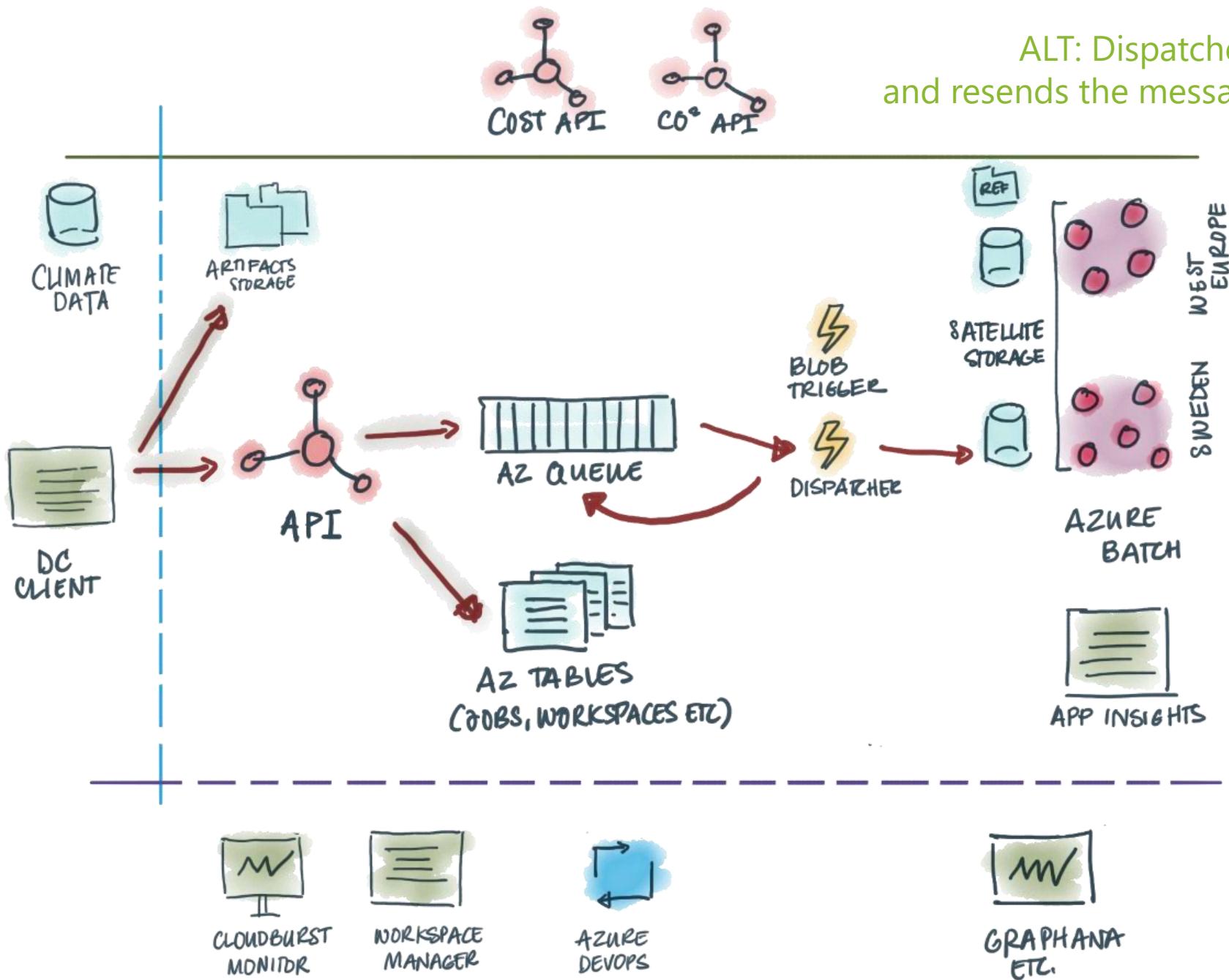
Dispatcher handles the message



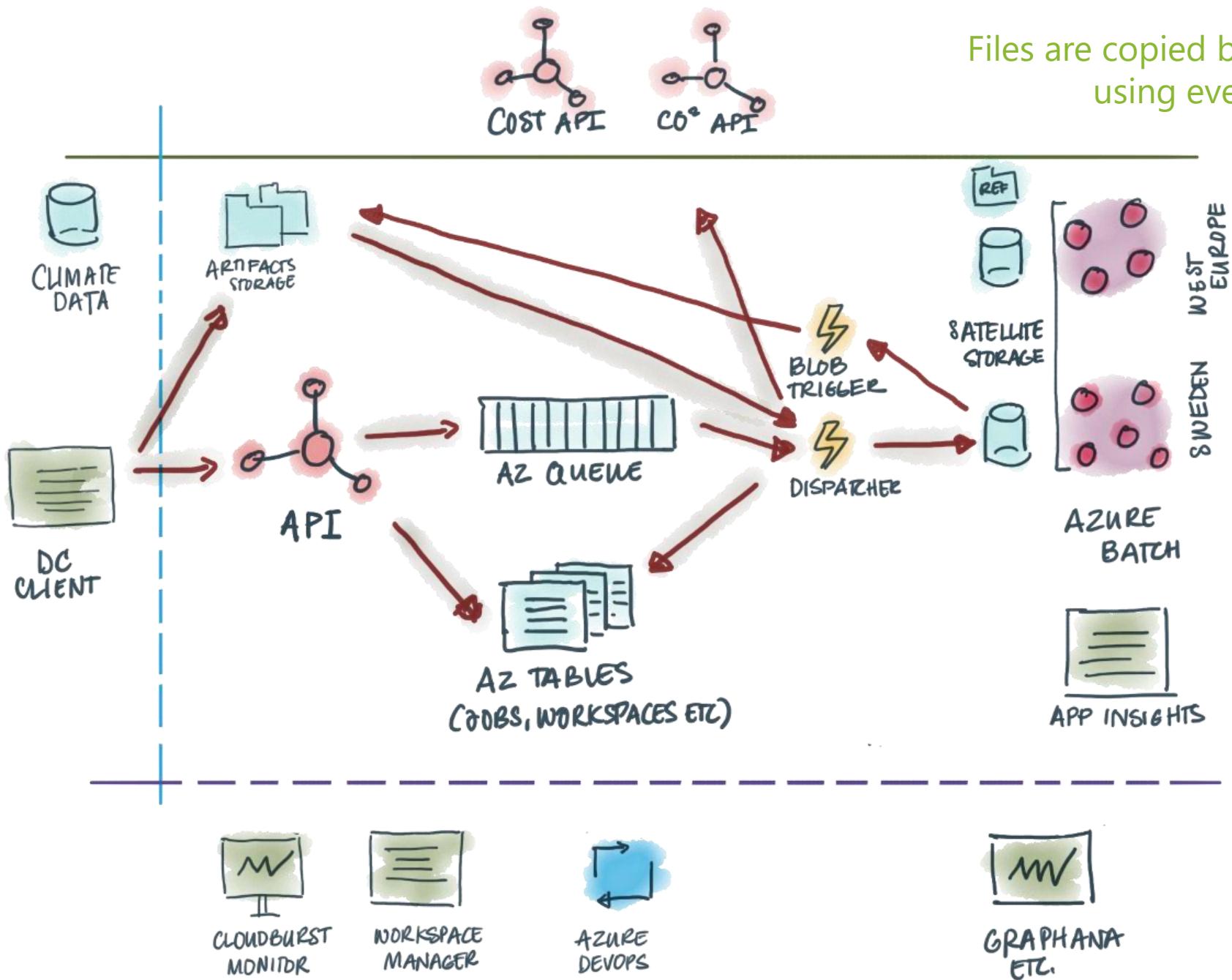
Dispatcher selects the pool
+ copies input data to satellite storage



ALT: Dispatcher needs to scale pool
and resends the message with a 1 min delay



Files are copied back on task complete
using event grid (BlobCreated)

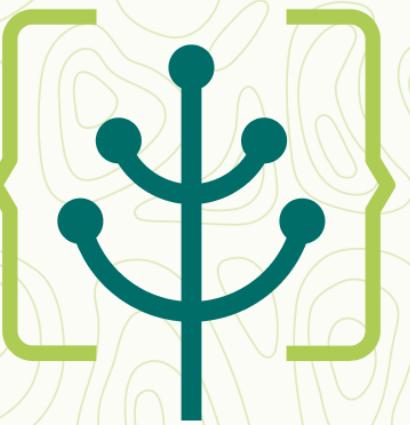




12-16% reduction with
time shifting within a day

30-90% reduction with
location shifting within EU

Tools



**Green
Software
Foundation**



Measuring: What you can't measure, you can't improve



Software Carbon Intensity calculation

One hour carbon emissions

- 400 W server
400 W/hour = 0.4 kWh
- 1000 kg embodied carbon for server with 4-year lifespan
 $1000 / (4 * 365 * 24) = 28\text{g CO}_2\text{eq}/\text{hour}$
- 216 carbon intensity
- 398 orders

Dell PowerEdge R6515

- AMD EPYC 7313P 3.0GHz
- 16GB RDIMM
- 480GB SSD SATA



* not including cooling, network etc.

$$\text{SCI} = ((\text{E} * \text{I}) + \text{M}) \text{ per R}$$

Energy consumed by software in kWh

Carbon emitted per kWh of energy, gCO₂/kWh

Carbon emitted through the hardware that the software is running on

Functional Unit; this is how software scales, for example per user or per device

E 0.4 kWh server

I 216 carbon intensity

M 28 g embodied carbon

R 398 orders

SCI = ((E * I) + M) per R

SCI = ((0.4 *216) +28) per 368

SCI = ((86.4) + 28) per 368

SCI = (114.4 g CO₂) per 368

SCI = 0.31 g CO₂/Order

$$\text{SCI} = ((\text{E} * \text{I}) + \text{M}) \text{ per R}$$

Energy consumed by software in kWh

Carbon emitted per kWh of energy, gCO₂/kWh

Carbon emitted through the hardware that the software is running on

Functional Unit; this is how software scales, for example per user or per device

Carbon Aware SDK

CarbonAware

API Documentation for Carbon Aware SDK

GET /emissions/bylocations/best Calculate the best emission data by list of locations for a specified time period.

GET /emissions/bylocations Calculate the observed emission data by list of locations for a specified time period.

GET /emissions/bylocation Calculate the best emission data by location for a specified time period.

GET /emissions/forecasts/current Retrieves the most recent forecasted data and calculates the optimal marginal carbon intensity window.

POST /emissions/forecasts/batch Given an array of historical forecasts, retrieves the data that contains forecasts metadata, the optimal forecast and a range of forecasts filtered by the attributes [start...end] if provided.

GET /emissions/average-carbon-intensity Retrieves the measured carbon intensity data between the time boundaries and calculates the average carbon intensity during that period.

POST /emissions/average-carbon-intensity/batch Given an array of request objects, each with their own location and time boundaries, calculate the average carbon intensity for that location and time period and return an array of carbon intensity objects.

Green-Software-Foundation / carbon-aware-sdk Public

Code Issues 56 Pull requests 7 Discussions Actions Projects 4 Security Insights

dev 12 branches 1 tag Go to file Add file Code

Willmish Merge pull request #324 from Green-Software-Foundation/stale... 2d8ef30 5 days ago 857 commits

.devcontainer Updates to docker file for generation and file system format chan... 7 months ago

.github use "stale" label for PRs as well 5 days ago

.vscode Update launch.json 'cwd' 5 months ago

docs Merge branch 'dev' into publish-openapi-spec 2 weeks ago

images Adding SOGS banner 3 months ago

samples Merge pull request #256 from microsoft/173/sdkAccessBoundaries 2 weeks ago

scripts Merge pull request #264 from microsoft/224/add-ps-scripts last month

src DCO Remediation Commit for Kristjana Popovski <kropovs@mic... 2 weeks ago

.gitignore Merge branch 'dev' into java-client 2 months ago

CONTRIBUTING.md Fix .md violations on existing files 2 months ago

LICENSE Create LICENSE 6 months ago

README.md Merge branch 'dev' into 223/getting-started-docs last month

action.yml Update action.yml 10 months ago

custom.markdownlint.jsonc Updates given latest gsf 2 months ago

entrypoint.sh Update entrypoint.sh 10 months ago

README.md

Carbon Aware SDK

You can reduce the carbon footprint of your application by just running things at different times and in different locations. That is because not all electricity is produced in the same way. Most is produced through burning fossil fuels, some is produced using cleaner sources like wind and solar.

When software does more when the electricity is clean and do less when the electricity is dirty, or runs in a location where the energy is cleaner, we call this **carbon aware software**.

The Carbon Aware SDK helps you build the carbon aware software solutions with the intelligence to use the greenest energy sources. Run them at the greenest time, or in the greenest locations, or both! Capture consistent telemetry and report on your emissions reduction and make informed decisions.

With the Carbon Aware SDK you can build software that chooses to run when the wind is blowing, enable systems to follow the sun, moving around the world to where energy is the greenest, and create tools that give insights and help software innovators to make greener software decisions. All of this helps reduce carbon emissions.

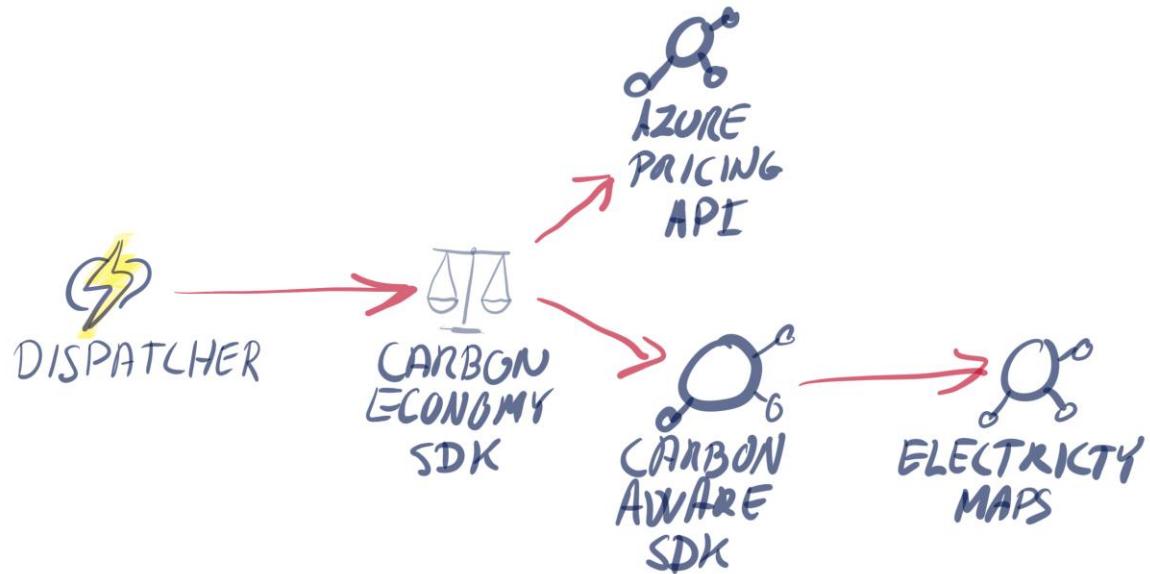
**Pay 5% extra for 50%
carbon reduction**



**Pay 50% extra for 5%
carbon reduction**



Carbon Economy SDK



Screenshot of the GitHub repository for the Carbon Economy SDK.

Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

Lybecker / carbon-economy-sdk (Public)

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main · 1 branch · 0 tags Go to file Add file Code About No description, website, or topics provided.

Lybecker Release b3bb981 1 minute ago 2 commits

- CarbonEconomy.Tests Release 1 minute ago
- CarbonEconomy Release 1 minute ago
- .gitignore Release 1 minute ago
- CarbonEconomy.sln Release 1 minute ago
- LICENSE Initial commit 47 minutes ago
- README.md Release 1 minute ago

README.md

Carbon Economy SDK

You can reduce the carbon footprint of your application by just running things at different times and in different locations. That is because not all electricity is produced in the same way. Most is produced through burning fossil fuels, some is produced using cleaner sources like wind and solar.

Reducing carbon emissions might incur a cost making it a business decision if the reduction is worth the cost. This SDK helps you to make informed decisions, enabling your system to answer challenges like below:

- 5% increase in cost, but 50% reduction in carbon emissions is worth it.
- 50% increase in cost, but 5% reduction in carbon emissions is not worth it.

Note: The SDK builds upon the [Carbon Aware SDK from Green Software Foundation](#). To compile and use this SDK, you need to compile and reference the Carbon Aware SDK.

Background story

This SDK was built in collaboration with [Microsoft](#) and [Vestas Wind Systems](#). Vestas needed to run very large wind turbine simulations in the cloud. For Vestas to meet their sustainability goals and become carbon neutral by 2030 without carbon offsets, they need a way to run their simulations in the most carbon efficient way.

Using the Carbon Economy SDK, Vestas can now run their simulations in the most carbon efficient way. This is done by running the simulations in the cloud at the time of day when the electricity is produced using the least carbon intensive sources. This is done by using the Carbon Economy SDK to evaluate the carbon emissions of the electricity in the different cloud regions at different times of the day. The simulation is then run in the cloud region with the lowest carbon emissions at the time of day when the electricity is produced using the least carbon intensive sources.

Two approaches were used:

- *Time shifting*: delaying the simulation to a time when the electricity is produced using the least carbon intensive sources. Resulting in a 8-12% reduction in carbon emissions.
- *Location shifting*: running the simulation in a cloud region with the lowest carbon emissions. Resulting in up to 90% reduction in carbon emissions.

Carbon Economy SDK Formula

$$\sqrt{((1 - w_e) * price)^2 + (w_e * emissions)^2}$$

Carbon Economy SDK Example

Emission weight: **0.25**

Location	Emission rating	Unit price	Norm. rating	Norm. price	Calc weight
Sweden Central	29	\$1.932	0	1	0.750
Norway East	42	\$1.824	0.107	0.870	0.653
West Europe	100	\$1.103	0.587	0	0.147
UK South	150	\$1.713	1	0.736	0.606

Emission weight: **0.75**

Location	Emission rating	Unit price	Norm. rating	Norm. price	Calc weight
Sweden Central	29	\$1.932	0	1	0.250
Norway East	42	\$1.824	0.107	0.870	0.232
West Europe	100	\$1.103	0.587	0	0.440
UK South	150	\$1.713	1	0.736	0.772

[General Settings](#)[Settings > Job Processing](#)[User Settings](#)

Choose the balance that works for you

[Rulebook](#)

How much should we optimize for greenness?

[Job Processing](#)

How much should we optimize for budget?



You are willing to pay 1% more for each 1% CO2 reduction

In the previous month, in only about 4% of cases, the greener option was cheaper, or priced the same.



Month over Month Numbers

CO2 Reduction



Budget Spending



Queue Length



Avg. Execution Time



Global Settings

Maximum number of standard instances

This number limits how many of standard nodes can run in the workspace, and everything else is waiting for Spot instances to become available.

Maximum number of cores

The maximum number of nodes in a Workspace across Azure Regions including spot instances.

Minimum discount for a spot instance

You can set a minimum discount (default 50%) compared to a normal prices node

 YES

Ovrule maximum number of standard instances to process High priority jobs

The admin can set a number of full cost instances in the settings. However this number might get overrule in case high priority job gets in

A wide-angle photograph of a glacial lagoon at sunset. The water is a deep blue, reflecting the sky and the surrounding landscape. Numerous icebergs of various sizes are scattered across the water. In the background, there are snow-covered mountains under a sky with wispy clouds.

Our future,
our responsibility

questions?