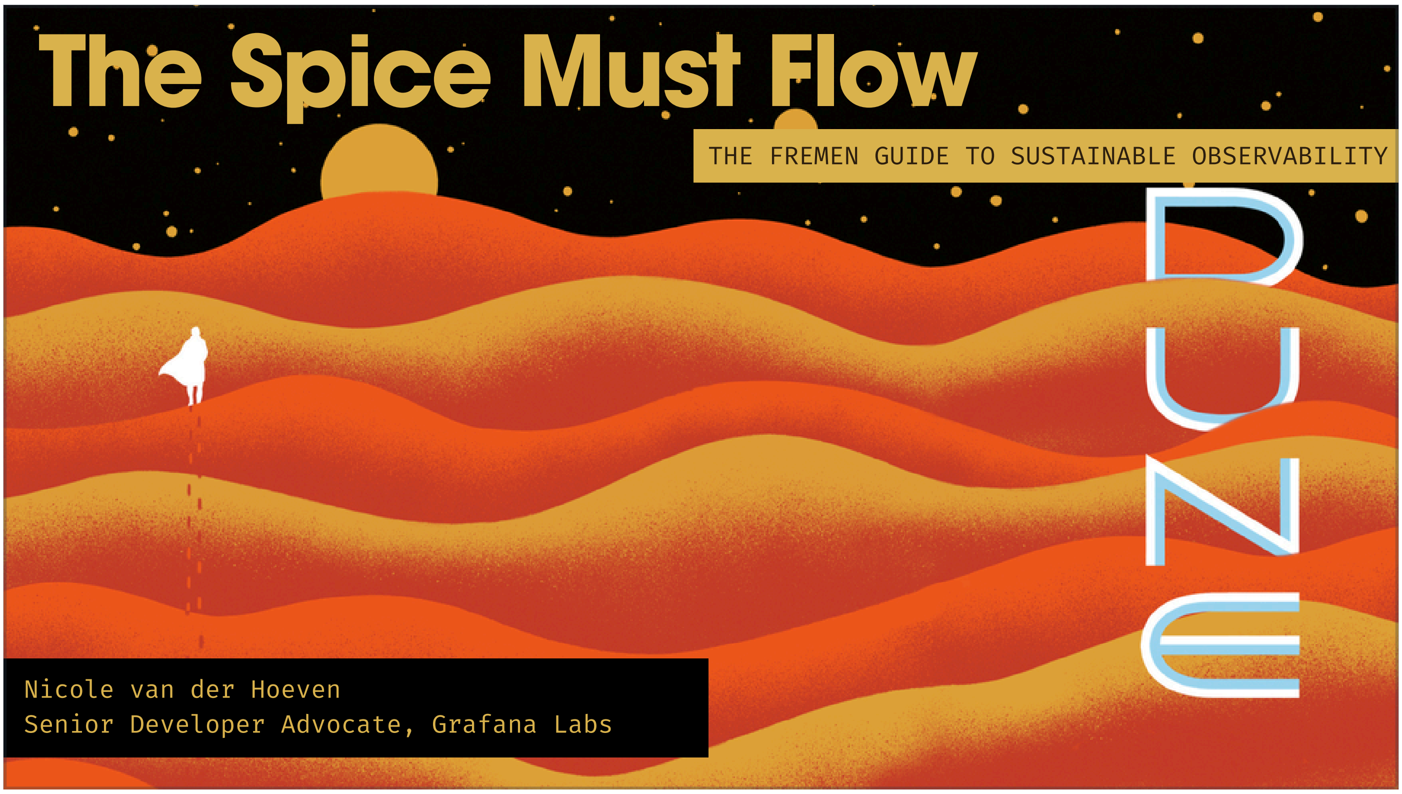


In Frank Herbert's classic sci-fi book Dune, there's a popular quote: “He (or she) who controls the spice controls the universe.”

Sounds like a temptingly simple proposition: control the spice, and you control the universe. The fine print is how, and what it will cost you.

In our world, the spice is data — telemetry — and sometimes it seems to flow endlessly. But just like everything else, it doesn't come free.

This is a story about observability, sustainability, and the discipline of finding balance between the two.

I'm Nicole van der Hoeven. I'm a Senior Developer Advocate at Grafana Labs, which means I get paid to learn how to do a lot of things in public. Here's what I've learned about sustainable observability. I'll provide a link to all my slides and everything I mention at the end!

I've also been a Dune fan for decades. In case you haven't been, let me give you a brief introduction to the universe of Dune.

The most striking characteristic about this universe is the reliance on something called spice melange, or spice for short. Spice is an orange/gold powder that heightens the senses to the point of giving humans prescient powers.

The Navigators of the Spacing Guild ingest a vaporized version of it to fold space and see paths through space-time, so space travel is powered by spice.

The Bene Gesserit is both a religious and a political faction. They're an order of nuns that use spice to predict the future and maneuver themselves into political control.

The Mentats, human computers, rely on spice to calculate and analyze information. That's especially relevant in a society like this one, where "thinking" computers have been banned (but that's another talk).

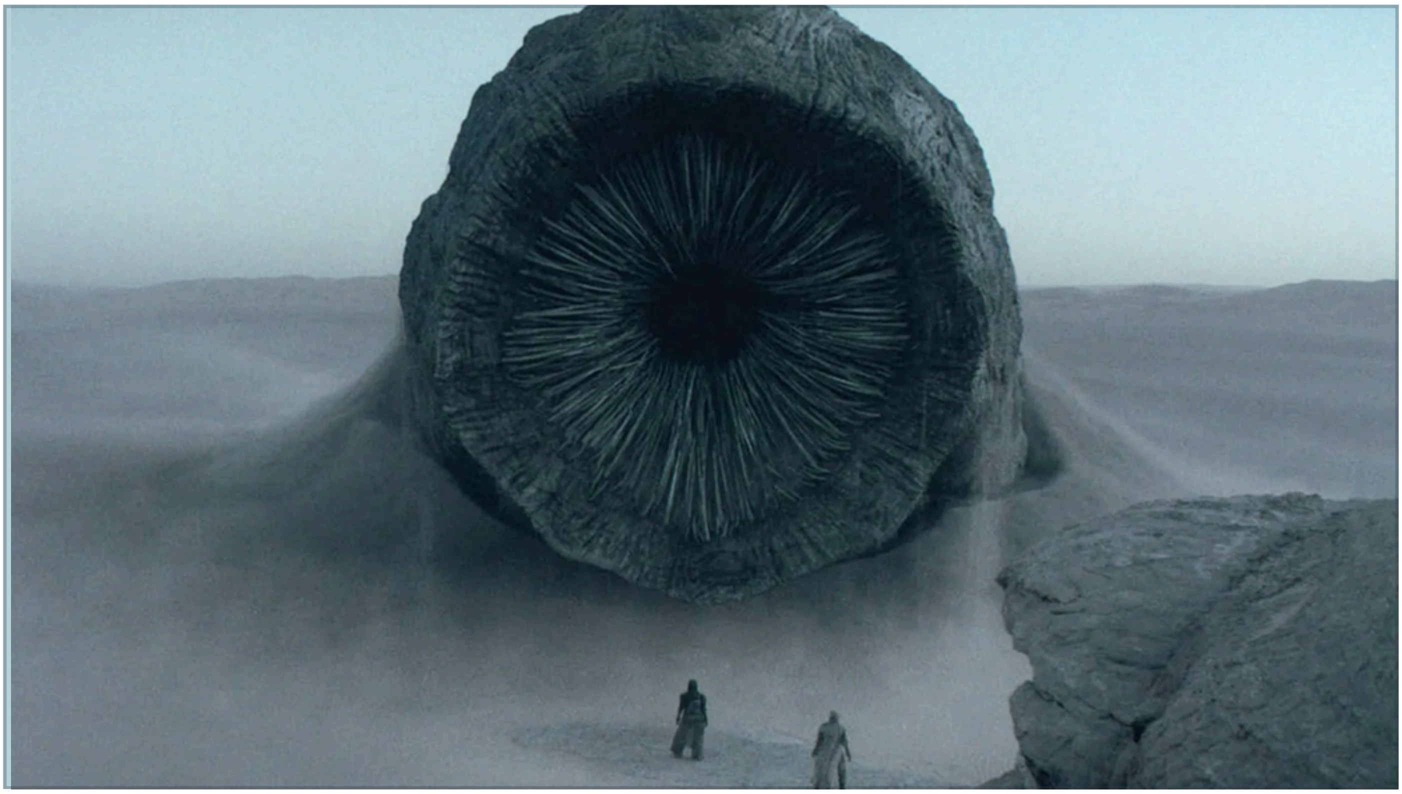
The Fremen, on the upper right, use spice in their religious rites and safeguard it as their most important economic resource.

And, of course, spice is heavily used recreationally. Have I mentioned it's addictive? No wonder it's the single most valuable substance in the Dune universe. On top of all the demand for it, it happens to be in short supply.

VERY short supply. In fact, spice melange, which is a natural resource, is only found on one planet: Arrakis, which is also called colloquially as Dune, is a desert planet that is the only source of spice in the whole universe.

It's home to the group of people I mentioned called the Fremen, who are very aware that their continued existence and importance in the world hinges on their continued production and harvest of spice.

So Arrakis is a fascinating mix of opposites: it's rich in spice, but as a desert planet, it's poor in water. As it turns out, there's a reason that those two go together.



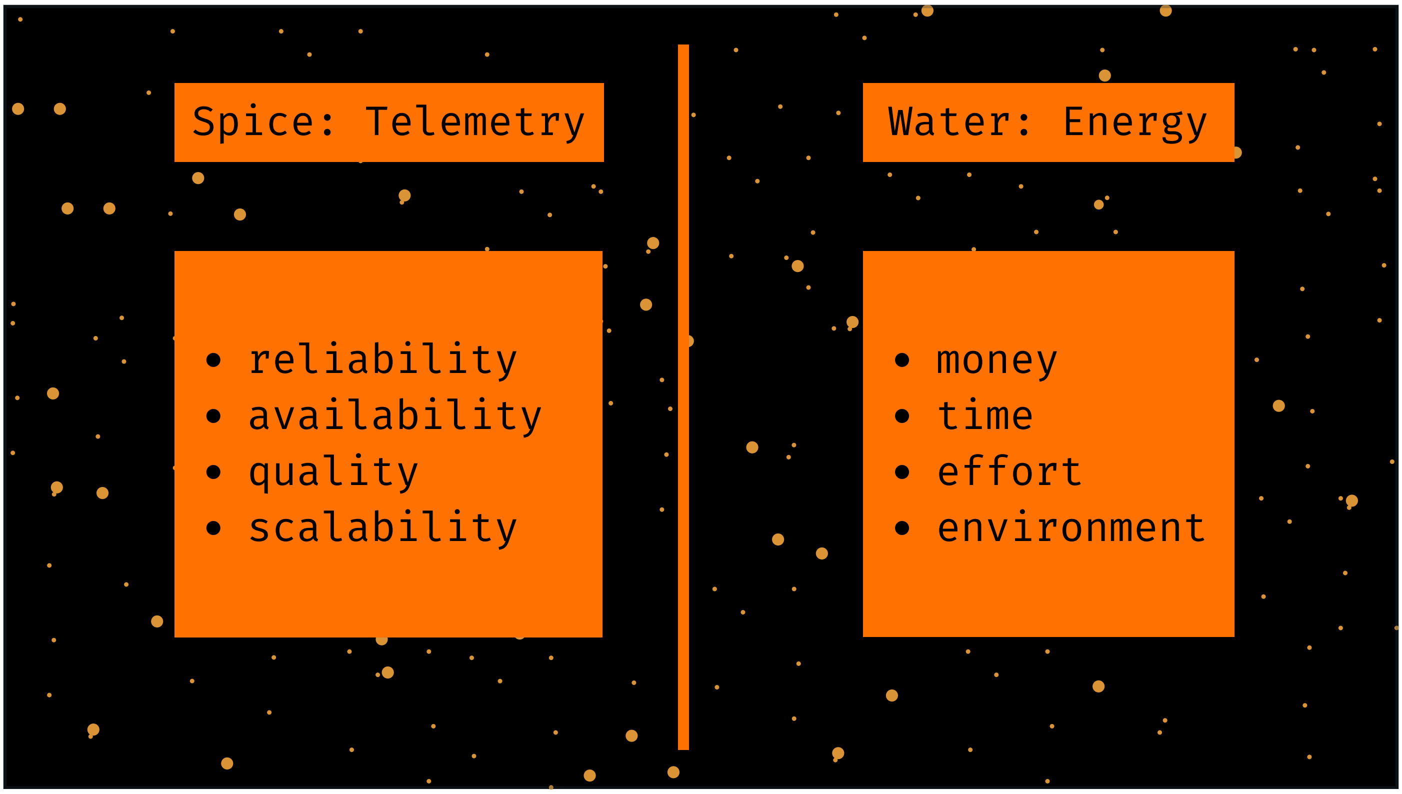
See, spice is created by sandworms, these massive sand-dwelling leviathans that drive up the price of spice by adding physical danger to the act of harvesting spice.

These sandworms nurture spice in different ways throughout their lifecycle. As sand trout, they create protective barriers that sequester sand away from water and create a pre-spice mass that eventually explodes to the surface. As adults, the sandworms' burrowing helps circulate the sand layers, spreading both pre-spice masses and sand trout so that the cycle can continue.

They're also quite indiscriminate about what they eat. They'll eat people, machines, vegetation... pretty much anything... except water.

Because that's the problem: water (the thing that sustains the humans living on the surface of Arrakis) kills sandworms. Too much water kills spice. Too much spice requires dry, desert-like conditions.

And so spice and water are in direct opposition withe one another, and Fremen, the native inhabitants of Arrakis, must live with the tension between two important resources: spice and water.



As builders and testers and deployers of software, we have a tension we must wrestle with, too. That tension is between telemetry and energy.

On the one hand, telemetry signals like logs, metrics, traces, and continuous profiles are what help us understand what's happening with our application. So naturally, we want as much of it as possible. All the data, all the time. We want more data than we know what to do with or and more than we can even foresee we'll need to use, so that we can be confident we'll be able to use that data to troubleshoot production issues. We want performance, speed, and as many nines as possible. Telemetry is what powers all that. But at what cost?

Our observability needs have gotten so big that our application stacks now have observability stacks that must also be observed. All that compute costs-- in terms of real dollars, time, and effort. And while we don't like to think of it, we all know that somewhere down the line, is a cost to the environment that we're paying.

But we want both. So engineers have to make decisions around this tradeoff all the time.

How do the Fremen do it?

The Fremen have something they call water discipline. They get so little water that what they do get is painstakingly saved.

On the top left is Zendaya showing off a Fremen stillsuit, an article of technical clothing designed not just to protect the body against the harsh desert environment but also to capture and recycle the body's moisture. In the Dune movies, she's often seen without the headpiece like in this photo because, well, the producers paid for that face, but real Fremen would never be barefaced in the open desert.

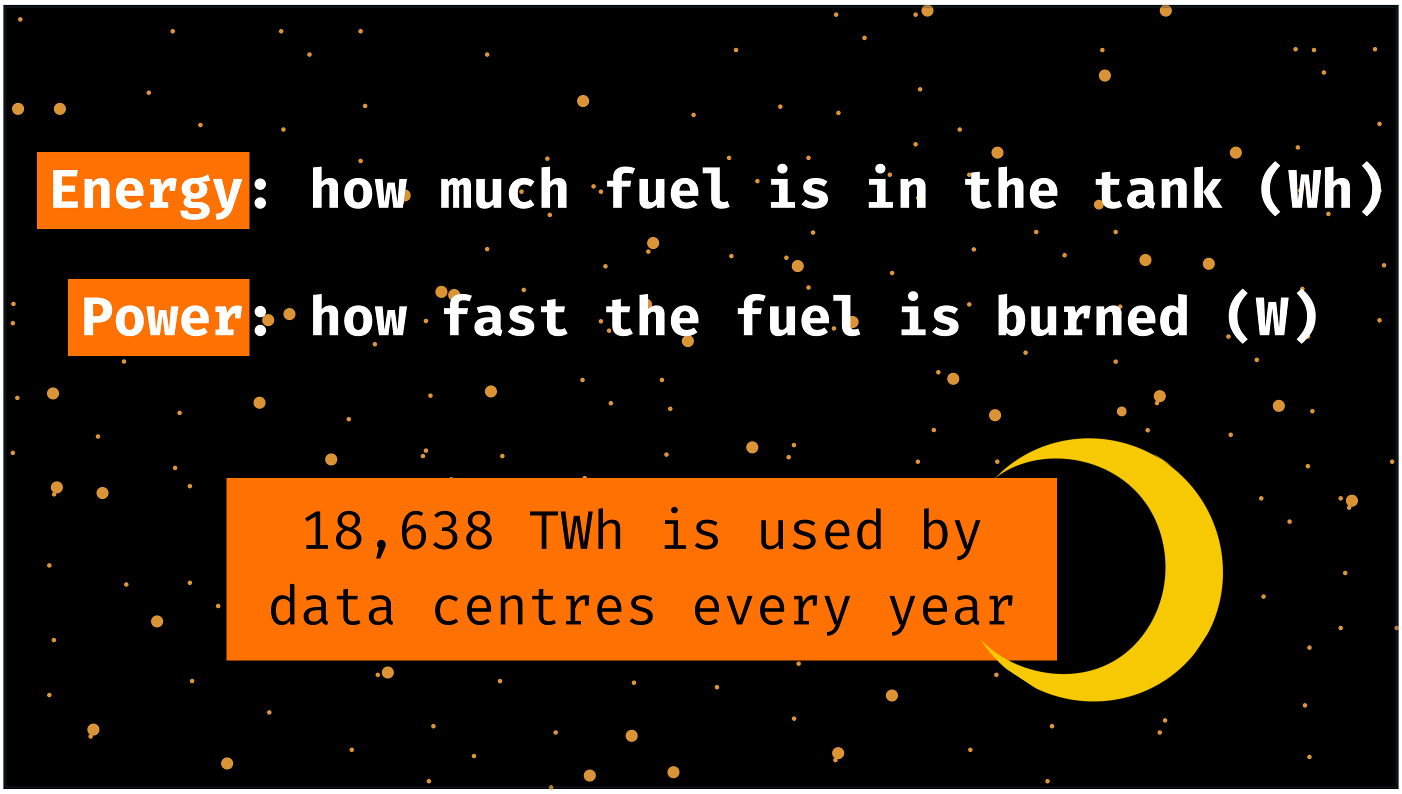
The bottom left shows a windcatcher that they deploy to capture trace moisture in the air.

At home in their underground villages called sietches, it is said that "The flesh belongs to the man, but his water belongs to the tribe." The Fremen all pool what water they have and keep them in tribal pools like these, carefully measured. This is the tribe's real wealth.

But Fremen don't just hoard water endlessly. At the bottom right is something called the water of life: an uncharacteristic ritual waste of water mixed with (what else?) concentrated spice that is passed around the sietch in an orgiastic celebration of life and togetherness.

The Fremen are not ignorant desert people. They might treat water with reverence, but they know the value of the spice their planet has, and they also cultivate the sandworms that keep their planet parched.

So let's talk about how we can walk this tightrope with them, and see how to adopt some water discipline.



Let's start by trying to quantify the environmental cost. What exactly are we to avoid here?

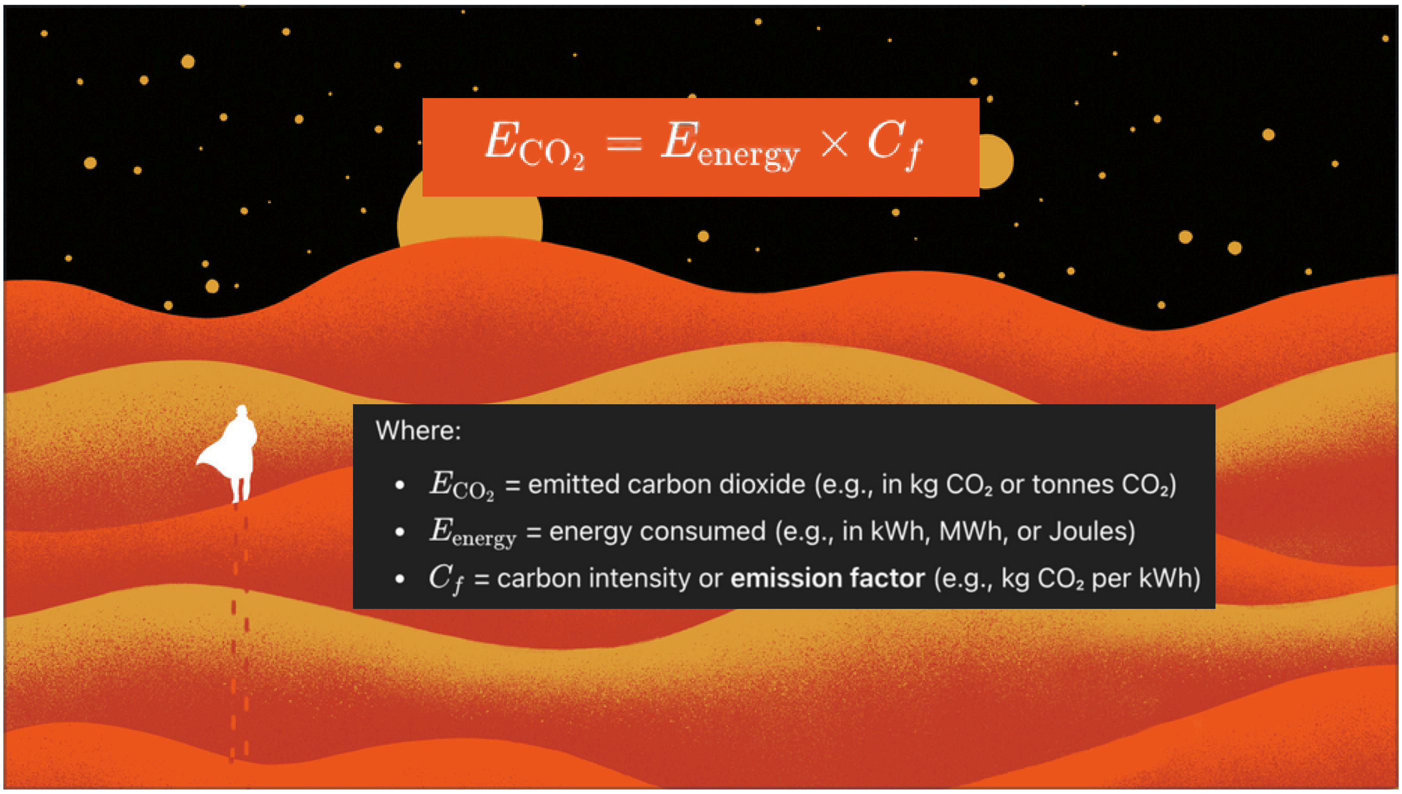
First, just a quick recap on the difference between energy and power. In terms of a gasoline-powered car, energy is how MUCH fuel is in the tank, and it's measured in Wh. Power is how fast that fuel is used up, and it's measured in terms of Watts.

So my Macbook Air requires at least 35W of power to charge. If I charge it at that rate for an hour, the energy consumption to charge it is 35 Wh.

When I was trying to look for figures about how much energy is being used for cloud computing, I really struggled to find numbers that specifically tied to cloud computing. So I settled for this figure on data centre usage instead: 18,638 TWh is used by data centres every year.

To put that into perspective, 1 TWh is 1 trillion Wh. To put that even more into perspective using something more day-to-day: If we could drive electric cars to the moon, this would be enough for 750,000 round trips.

But energy itself STILL isn't the direct cost to the environment.



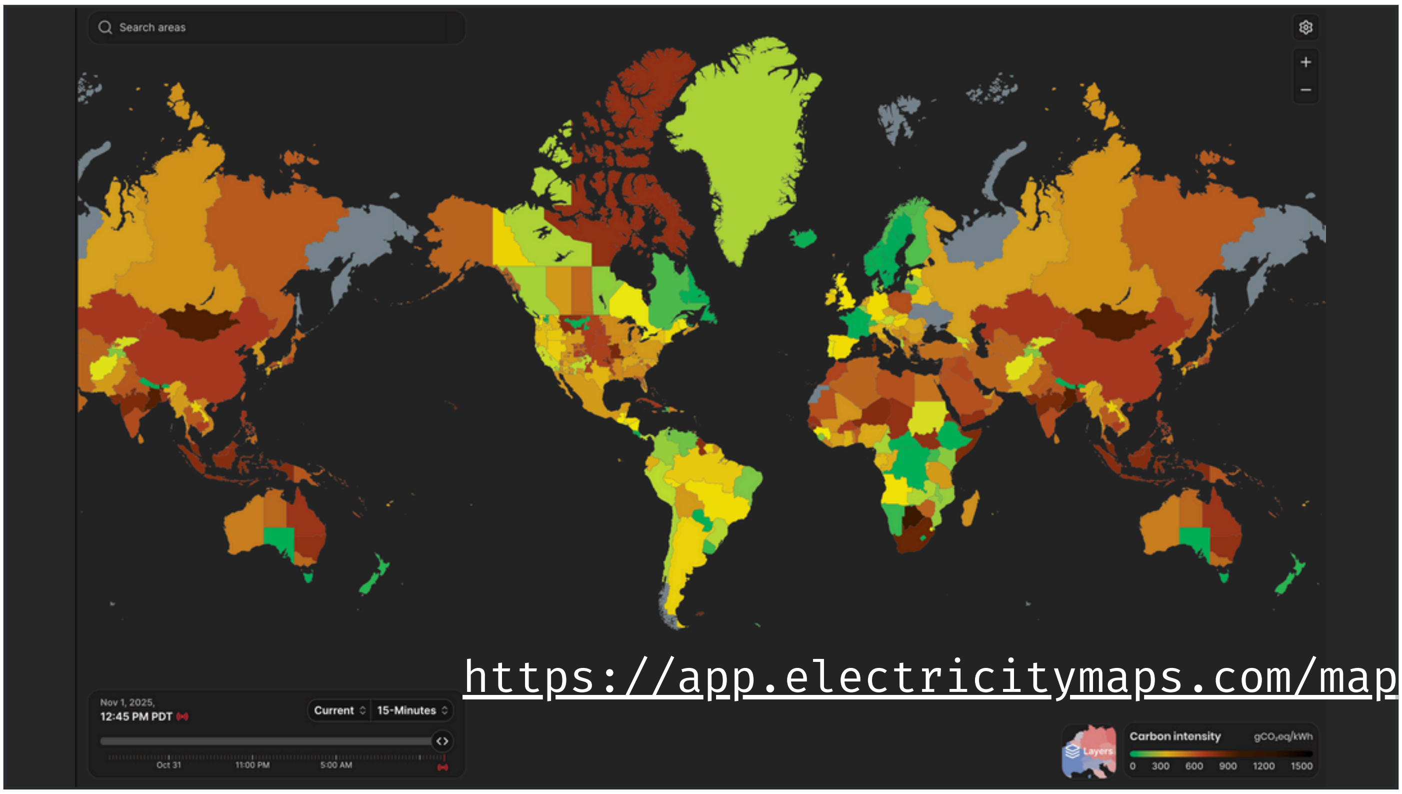
That would be CO2 emissions. The amount of CO2 emitted into the air has become the golden standard for environmental cost, but it CAN be derived from energy.

This equation means that to get the CO2 emitted from an activity, you can take the energy consumption and multiple it by Cf, which is called carbon intensity, or sometimes emission factor or carbon coefficient.

This carbon intensity number depends primarily on the source of the energy (like whether the energy was produced from coal or wind or solar sources) but also things like
- conversion efficiency (technology to convert into power can be more or less efficient)
- transmission efficiency (in electricity travelling through power lines)
- incidental/indirect emissions (such as costs to built the power plants for even a "zero-carbon" energy source)

(A high carbon intensity is bad, because you have to multiply it to energy to get the CO2 emitted.)

So now we know that there is a way to estimate the CO2 emissions based on power consumption. That still doesn't give us a good understanding of how the clusters we deploy affect the environment, though. But there are a few things we can do to approximate that.



Both AWS and GCP use these worldwide electricity maps. If you go to this site, you can see for yourself which countries and regions use greener sources than others. The darker the colour is, the worse the source is. The lighter it is, the better it is. The countries with the lightest colours are using alternative electricity sources to coal.

So when we're using a cloud provider, we can actually reduce the environmental harm we're doing just by choosing a different availability zone-- even without changing anything else in the size of our clusters. There may be some tradeoffs in latency, but luckily the cloud providers have thought of that. See, on top of what the countries have decided they want to do for their electricity sources, the big cloud providers also try to make it easy for us to choose greener sources using their own methods.

Carbon data across Google Cloud regions

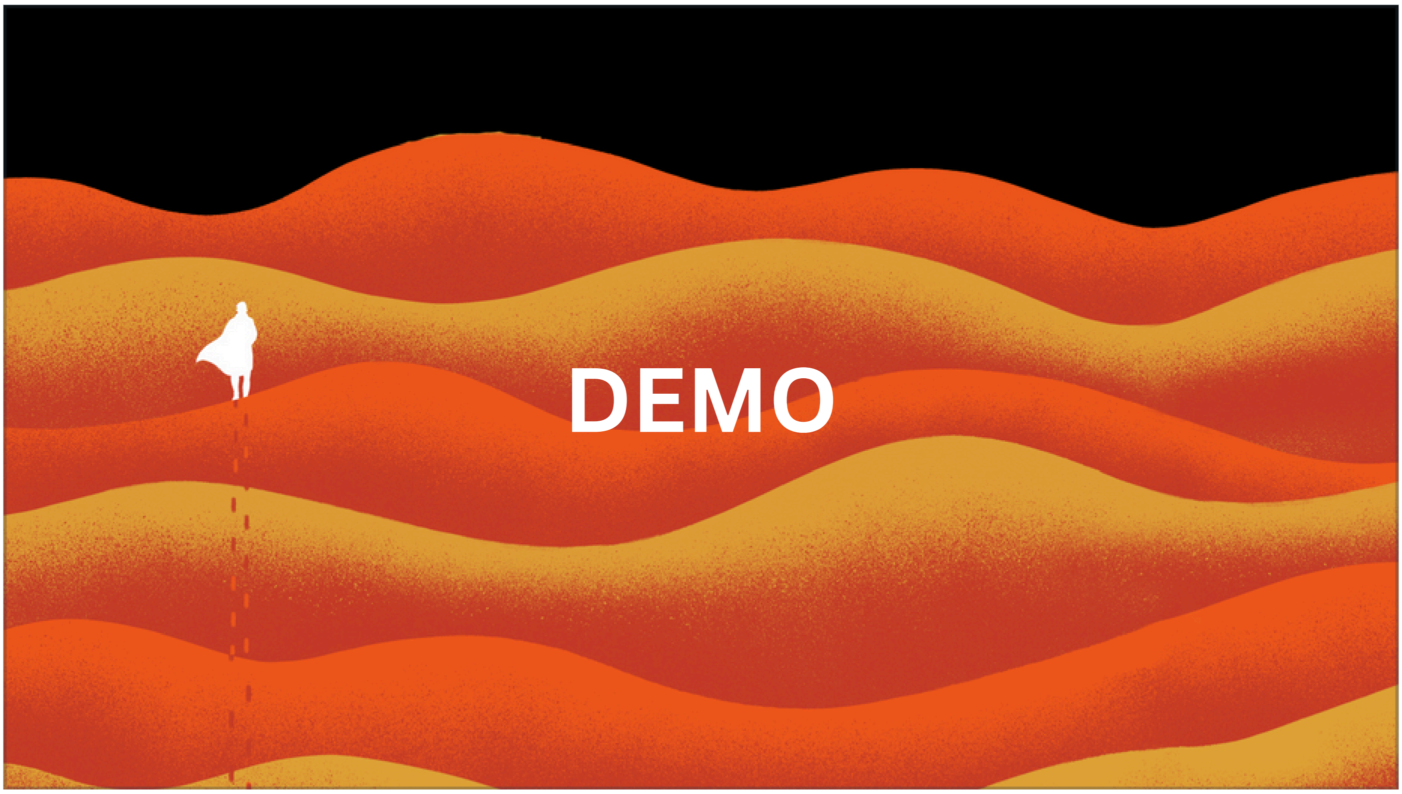
Google Cloud Region	Location	Google CFE%	Grid carbon intensity (gCO_2eq/kWh)	
africa-south1	Johannesburg	15%	657	
asia-east1	Taiwan	17%	439	
asia-east2	Hong Kong	1%	505	
asia-northeast1	Tokyo	17%	453	
asia-northeast2	Osaka	46%	296	
asia-northeast3	Seoul	37%	357	
asia-south1	Mumbai	9%	679	
asia-south2	Delhi	29%	532	
asia-southeast1	Singapore	4%	367	
asia-southeast2	Jakarta	18%	561	
australia-southeast1	Sydney	34%	498	
australia-southeast2	Melbourne	39%	454	
europe-central2	Warsaw	40%	643	
europe-north1	Finland	98%	39	Low CO_2
europe-north2	Stockholm	100%	3	Low CO_2
europe-southwest1	Madrid	87%	89	Low CO_2
europe-west1	Belgium	84%	103	Low CO_2
europe-west2	London	79%	106	Low CO_2
europe-west3	Frankfurt	68%	276	
europe-west4	Eemshaven	83%	209	Low CO_2
europe-west6	Zürich	98%	15	Low CO_2
europe-west8	Milan	73%	202	
europe-west9	Paris	96%	16	Low CO_2
europe-west10	Berlin	68%	276	
europe-west12	Turin	73%	202	
me-central2	Dammam	1%	382	
me-west1	Tel Aviv	7%	434	
northamerica-northeast1	Montréal	99%	5	Low CO_2
northamerica-northeast2	Toronto	84%	59	Low CO_2
northamerica-south1	Mexico	19%	305	
southamerica-east1	São Paulo	88%	67	Low CO_2
southamerica-west1	Santiago	92%	238	Low CO_2
us-central1	Iowa	87%	413	Low CO_2
us-east1	South Carolina	31%	576	
us-east2	Georgia	42%	340	
us-east4	Northern Virginia	62%	323	
us-south1	Dallas	94%	303	Low CO_2
us-west1	Oregon	87%	79	Low CO_2
us-west2	Los Angeles	63%	169	
us-west3	Salt Lake City	33%	555	
us-west4	Las Vegas	64%	357	

GCP releases this table that shows carbon intensities AND what they call "Carbon-Free Energy" Percentage (CFE%). Let's break that down.

We just talked about how carbon intensity is how much carbon electricity from that region TYPICALLY uses.

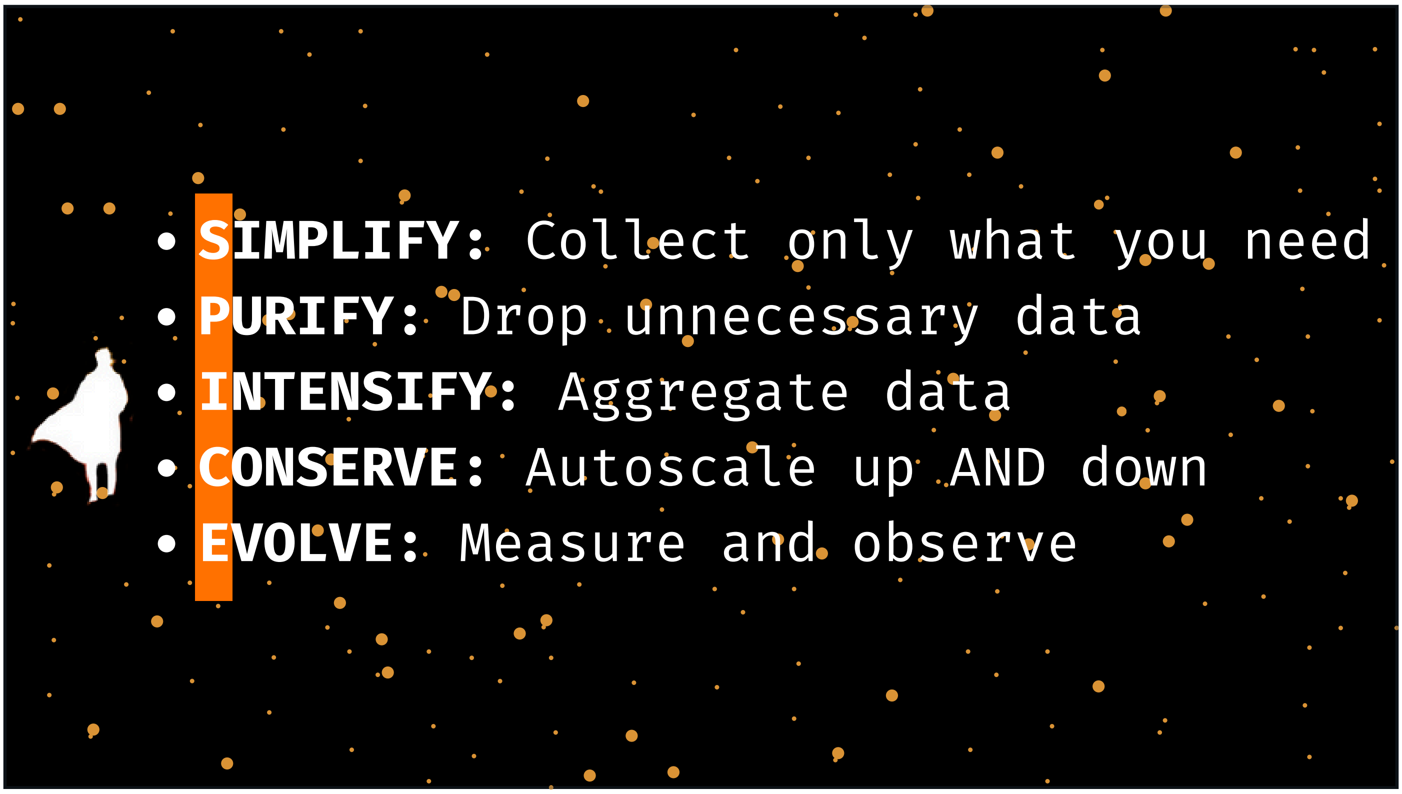
But Google also sometimes offset or use cleaner wind from that region, so they have their own metric, the CFE%, that expresses how clean the energy is that THEY specifically use for GCP regions. So it might be that the country doesn't use clean energy sources, but Google has been able to come in and create their own clean sources or offset them in some way.

Some notable regions here:
- Johannesburg is the worst region I could find from this perspective, with a carbon intensity of 657.
- Stockholm is looking pretty good at 100% clean energy-- a combination of regional efforts and Google's efforts.
- I was also gratified to learn that Iowa, which is what I chose by default for the demo I'm about to show you, actually has a poor carbon coefficient in general, but Google buys most of its energy from wind farms there. That made me feel a bit better.



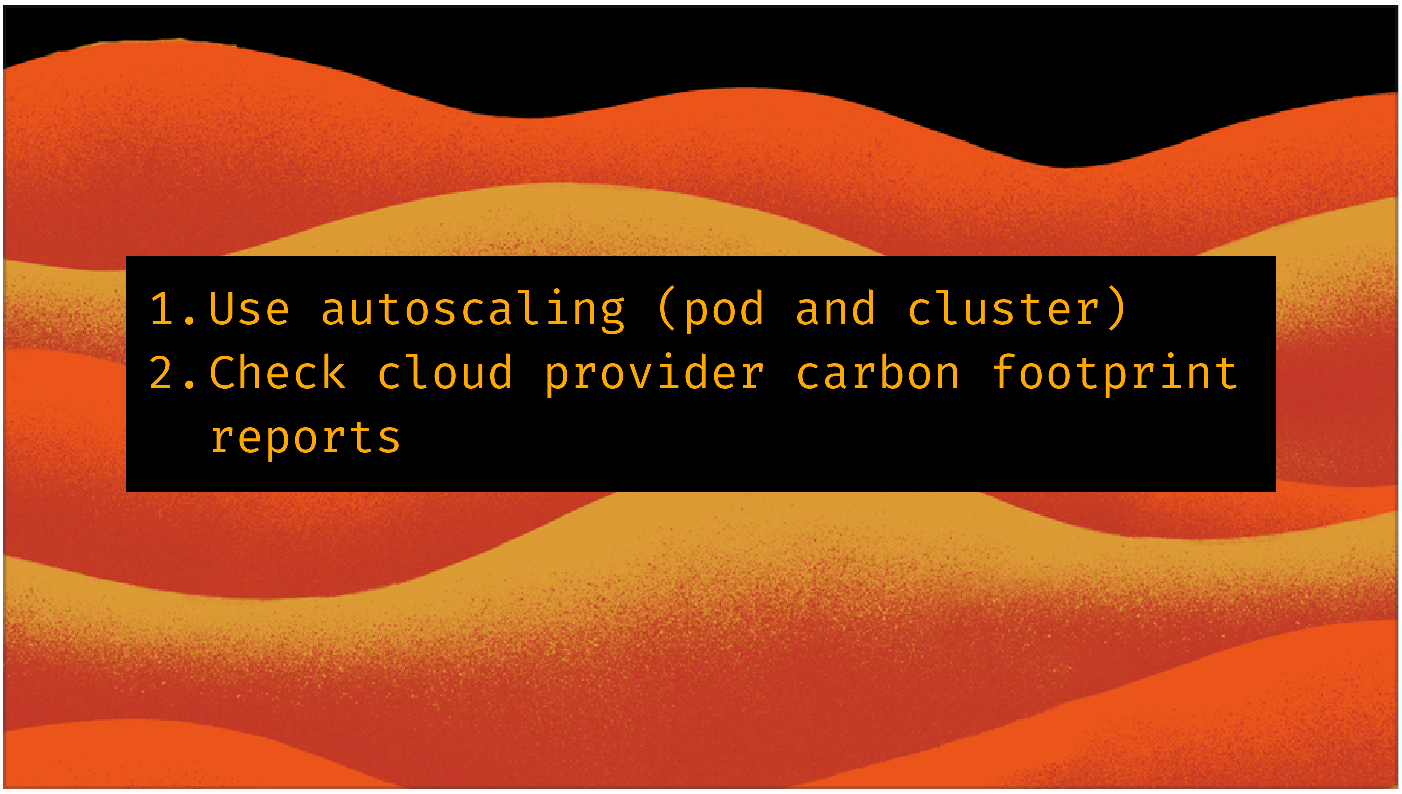
<10min> Demo (20 mins)
- Talk about the architecture of the application
- Show them the Kubernetes Config Monitoring dashboard. Note the number of nodes and pods.
- Play through one round of the game and show Faro instrumentation.
- Show them the Spice Runner App Stats dashboard
- Go through KEDA config and show what conditions it checks to scale
- Go through GKE Autoscaler and show what conditions it checks to scale
- Encourage them to play it themselves
- Show them as the dashboard updates live to show node and pod changes
- Show them Kepler stats

nicole.to/spicegame



Let's take a step back here and talk about some other ways to reduce our costs that I didn't get to show you. I've helpfully assembled them into an acronym for you.
- SIMPLIFY: Don't collect data/start compute that you don't need. Question whether you really need "info" level logs. Disable debug-level logging in production. Don't start new compute if you can use existing compute instances.
- PURIFY: Drop unnecessary data and resources. This is akin to filtering the water that you do get. You can do this by things like using intelligent sampling so that you don't have to save absolutely every raw data point of telemetry-- maybe it's okay to only save X%, or only keep traces that contain an error, or only data from a certain endpoint.
- INTENSIFY: Refine the water by aggregating data. Consider aggregating metrics at the telemetry collector level (that's actually what I do for some game metrics with Alloy).
- CONSERVE: This involves using both pod and cluster autoscaling, but also looking at data retention tiers if you're okay with saving older data at a lower granularity.
- And EVOLVE: There's a real cultural shift involved in more sustainable observability, and culture is hard to quantify. But it's amazing how much it changes the culture to have a single dashboard that everyone looks at. Observability specifically set up to measure environmental cost is a great way to get there.

Maybe that's a lot. Let me give you a tl;dr for that.



Here are two concrete things that you can do today that get you most of the way there.

1. First, consider cluster autoscaling. If you're on AWS, I recommend Karpenter-- internally at Grafana Labs, switching from Kubernetes Cluster Autoscaler to Karpenter resulted in idleness falling by 50% across the board. If you're using GCP, GKE Cluster Autoscaler worked pretty well and was easy to set up.
2. Secondly, major cloud providers give carbon footprint reports for your specific usage. I couldn't get our IT department to assign me the right role to view these reports for this demo, but they do exist. While they're still not going to be 100% accurate (remember that it's in cloud providers' best interests to legally deflate those numbers), they're still a good starting point.

Let's get back to Arrakis as we wrap up here, and see what we can learn from it.

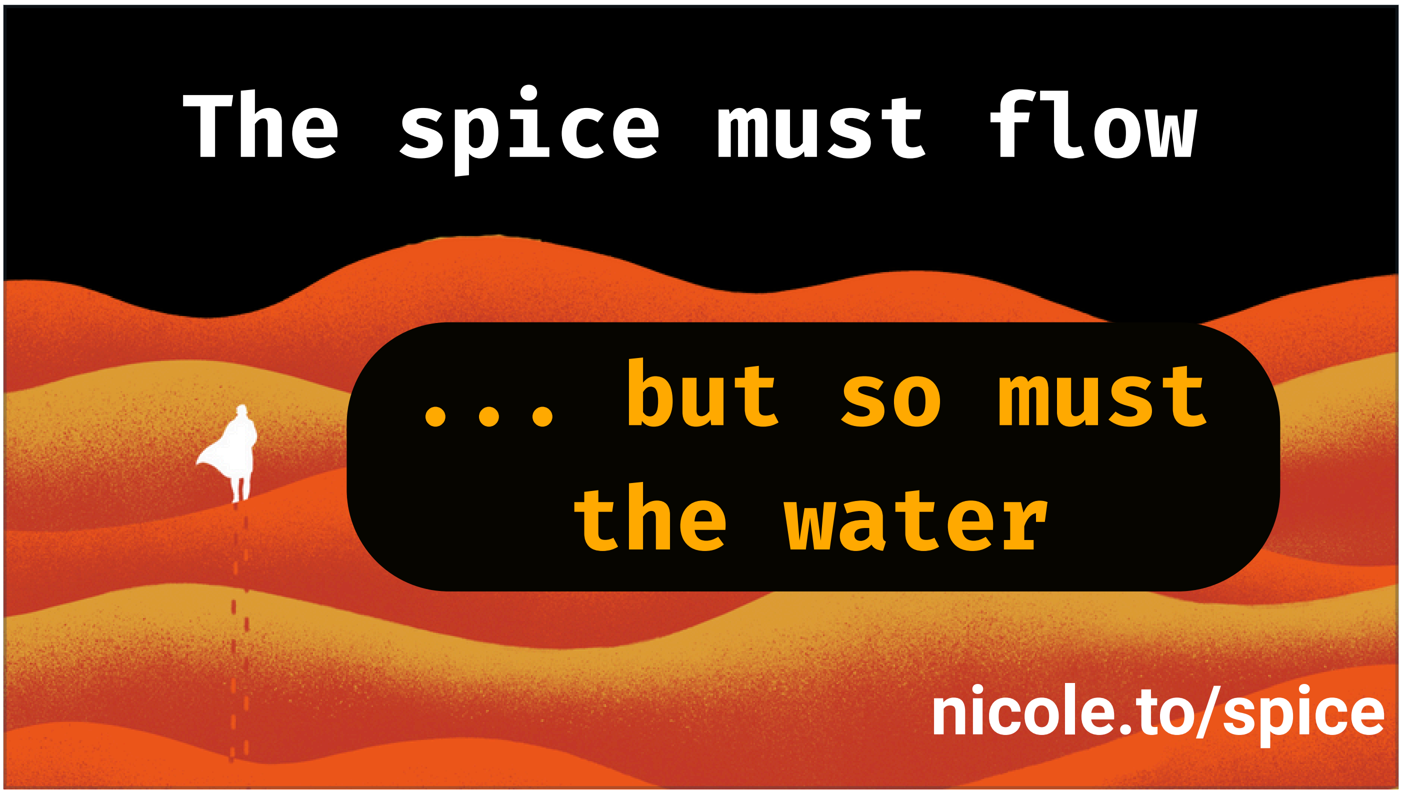
A long time before the first book (and certainly before the movies), Arrakis used to be a planet much like ours: blue, watery, and teeming with vegetation.

Eventually, the sandworms, presence of spice, and the desire to protect spice production from multiple fronts gradually made those waters run dry, and then it became the "Dune" that the series starts off with.

But what I've always liked about Dune is that it isn't your usual environmental cautionary tale: look at what we did to our planet. We've ruined it because of our greed. If only we cared more about our planet. I find that thinking overly simplistic, whether we're talking about Dune or our own planet.

Dune is not about the perils of turning out water world into a desert world. Actually, at the end of the Dune series, Arrakis ends up being a water world again, in part because of a desire to "save" the water. But it didn't work out well, because in the process, the spice on the planet was ruined forever, and along with it, its importance in the universe. It's not an either/or. Prioritizing water at whatever cost is just as overly idealistic and harmful as prioritizing spice at whatever cost.

If this were a cautionary tale, the warning is this: Sometimes optimising too much in either direction ruins everything.



So the lesson from Dune, from Arrakis and the Fremen, is one of balance.

We need to learn to balance our need for telemetry against the not insignificant effects to the environment. This doesn't mean we shouldn't STORE data. But it means we should be intentional about whether or not we HOARD it. It means making a conscious decision about what our tolerances are and using tools like autoscaling to allow for that play to expand in either direction and not have it be irreversible. It means having the environmental cost data TO make informed decisions in the first place.

So the lesson here is this: "The spice must flow... but so must the water."

Thanks for listening! Everything I've shown, mentioned, or demonstrated in this talk is on this link.