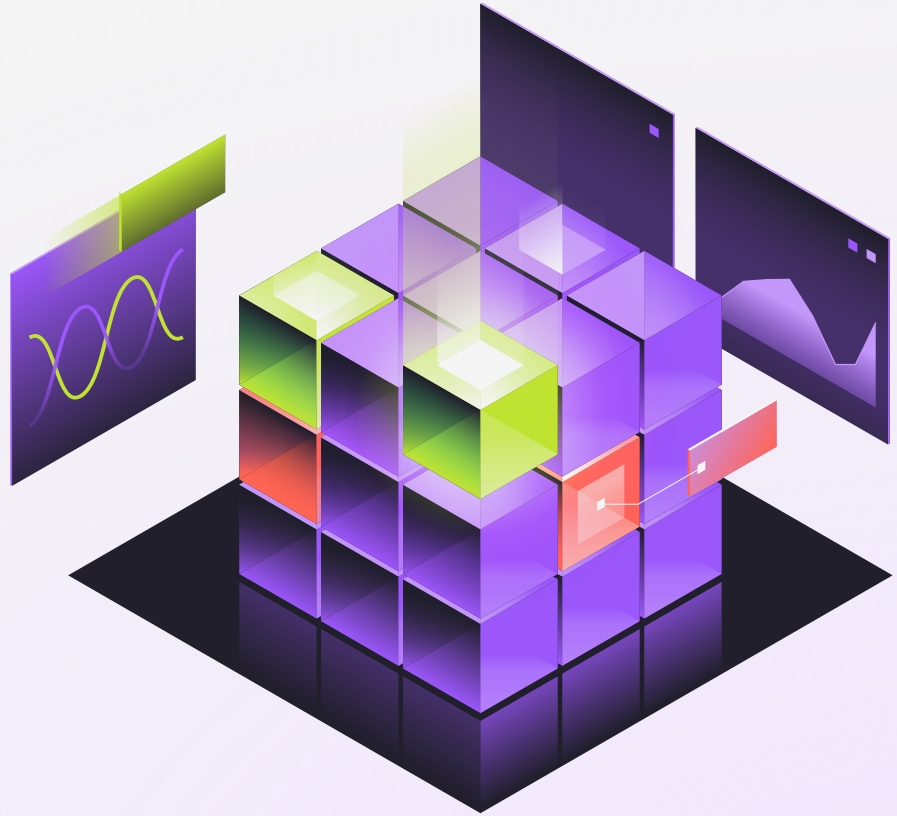
earthmover

from exploration to production

flexible data repositories with Arraylake and Icechunk



Julius Busecke · Forward Deployed Engineer, Earthmover PBC

FTP Early Career Researcher Webinar Series · July 2026

MY JOURNEY

🌊 Ocean & Climate Scientist

PhD in Physical Oceanography at Columbia University

Postdoc at Princeton, then research scientist at Lamont-Doherty

Studied ocean mixing and biogeochemistry in high-resolution models and CMIP6

👷 Open Source Developer

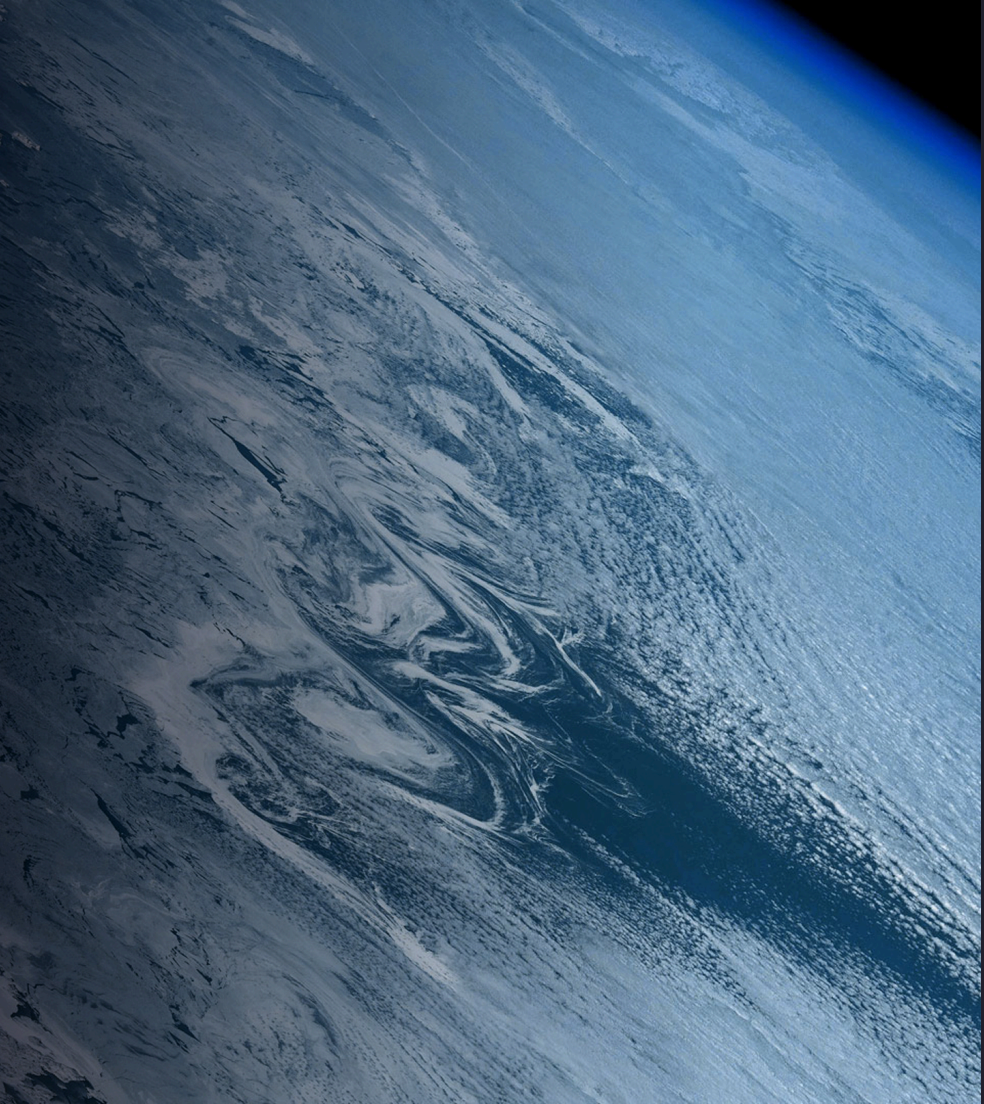
Maintainer of **xGCM** and **xMIP**, longtime **Pangeo** contributor

Data and open science roles at **NSF LEAP** and **M2LInES**

Made global climate model archives (CMIP6) analysis-ready in the cloud

🛠️ Forward Deployed Engineer

At Earthmover, embedded with ARIA-FTP teams as your technical contact



earthmover

Our Mission

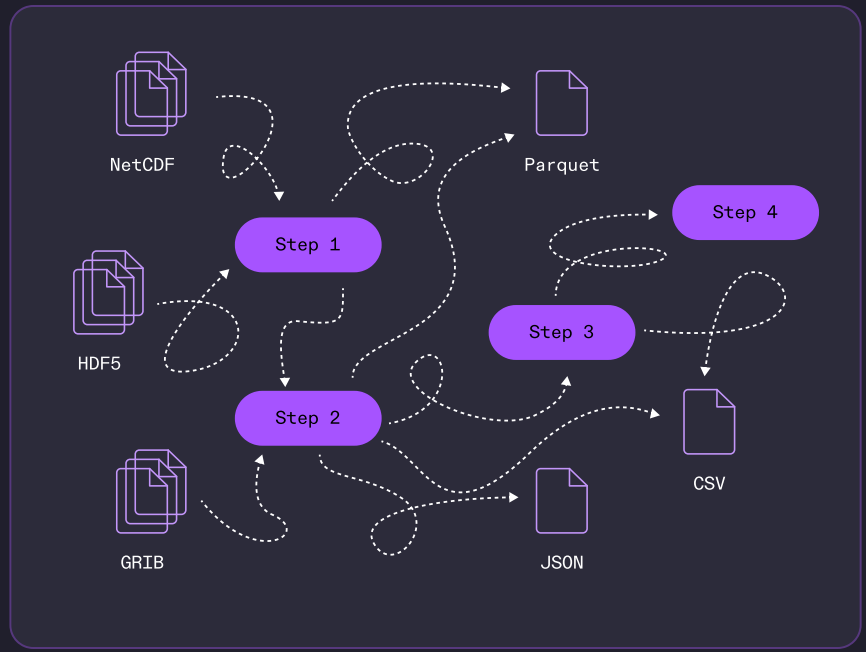
To empower **people** to use **scientific data** to solve humanity's **greatest challenges.**

- Founded in 2022 as a **public benefit corporation** by **Ryan Abernathey** and **Joe Hamman.**
- 14 people spread across US, Europe, and South America.
- "Open core" business model - investing in **community open source** and providing a **managed cloud data platform** for institutions.
- Serving clients in **private sector** (energy / commodities, insurance, AI startups), **academia**, and **government.**

THE PROBLEM

science-scale data is stuck in files

- Millions of netCDF and HDF5 files spread across HPC scratch, FTP servers, and laptops
- "Download first, analyze later" stops working once the data outgrows your disk
- No versioning and no shared source of truth: `final_v2_REALLY_final.nc`
- Every collaborator rebuilds the same pipeline from scratch



THE ALTERNATIVE

from files to arco data cubes

Humans and agents don't want to deal with thousands of individual files. They want ARCO data cubes.

Analysis Ready

> *All timesteps, variables, ensemble members, etc. are organized into a single logical object (data cube).*

Cloud Optimized

> *Applications can efficiently stream data directly from cloud object storage with low latency and high throughput.*

Zarr is the most widely used ARCO data format for Earth-system-model data.

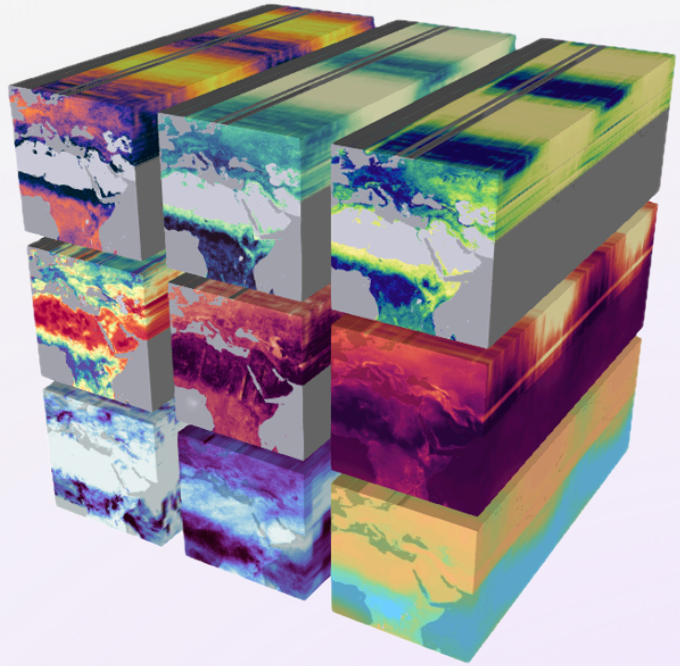
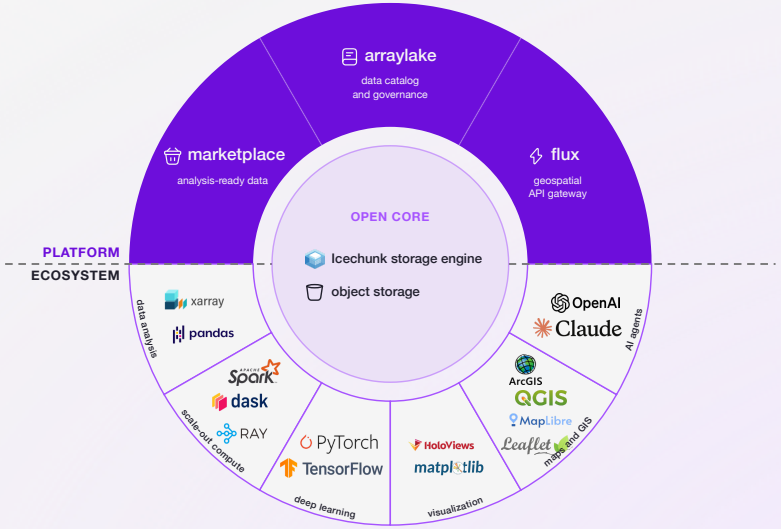


Image credit: Mahecha et al. 2020. https://doi.org/10.5194/esd-11-201-2020

WHAT WE'RE BUILDING

one platform, from first commit to operational service



Icechunk storage engine → **Arraylake** catalog → **Flux** maps & APIs → **Marketplace**

ICECHUNK

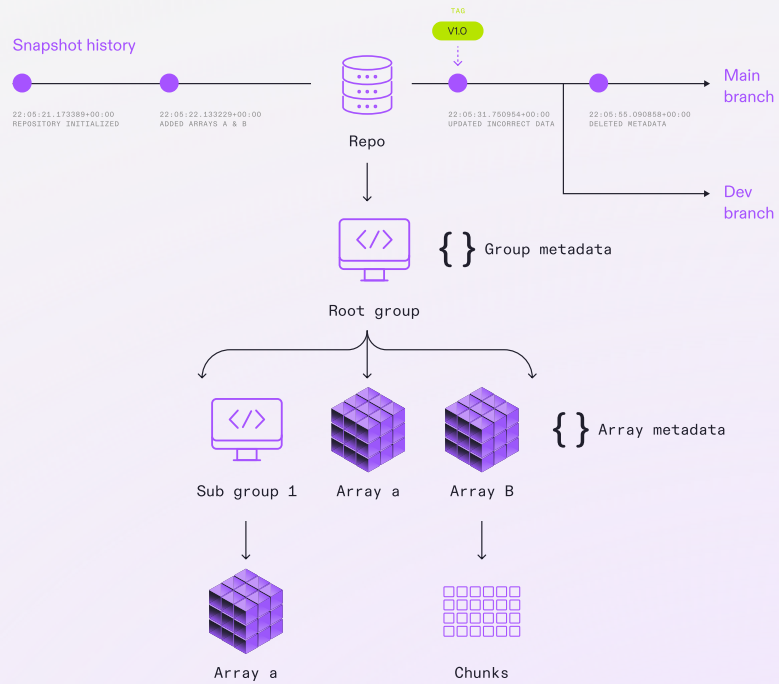
zarr for production

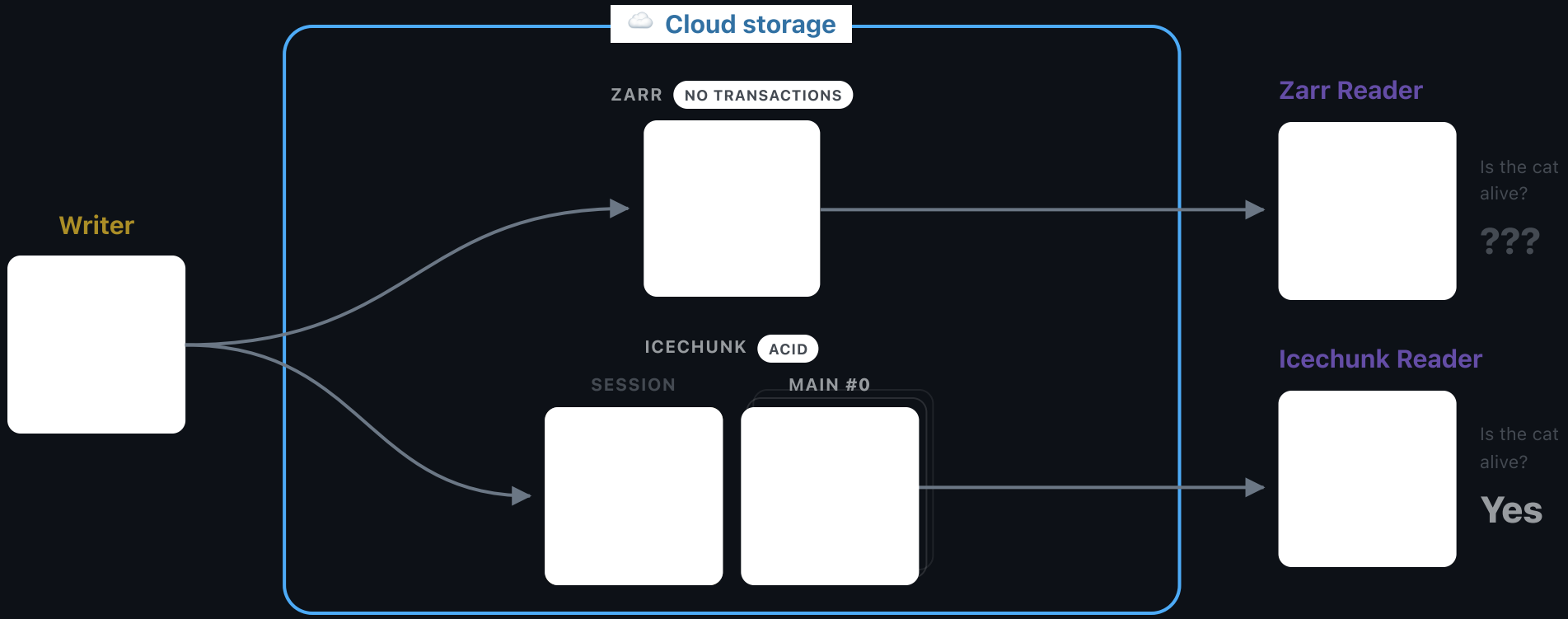
A specialized **Zarr Store** which supports:

- **ACID transactions** in a distributed context
- **Version history** (snapshots, branches, tags) with optional expiration and garbage collection
- Complex metadata operations (move, shift, reindex)
- **Virtual chunks:** pointers to blobs of numerical data in external files (NetCDF, GRIB)
- **Open source** (Apache 2.0), **serverless**

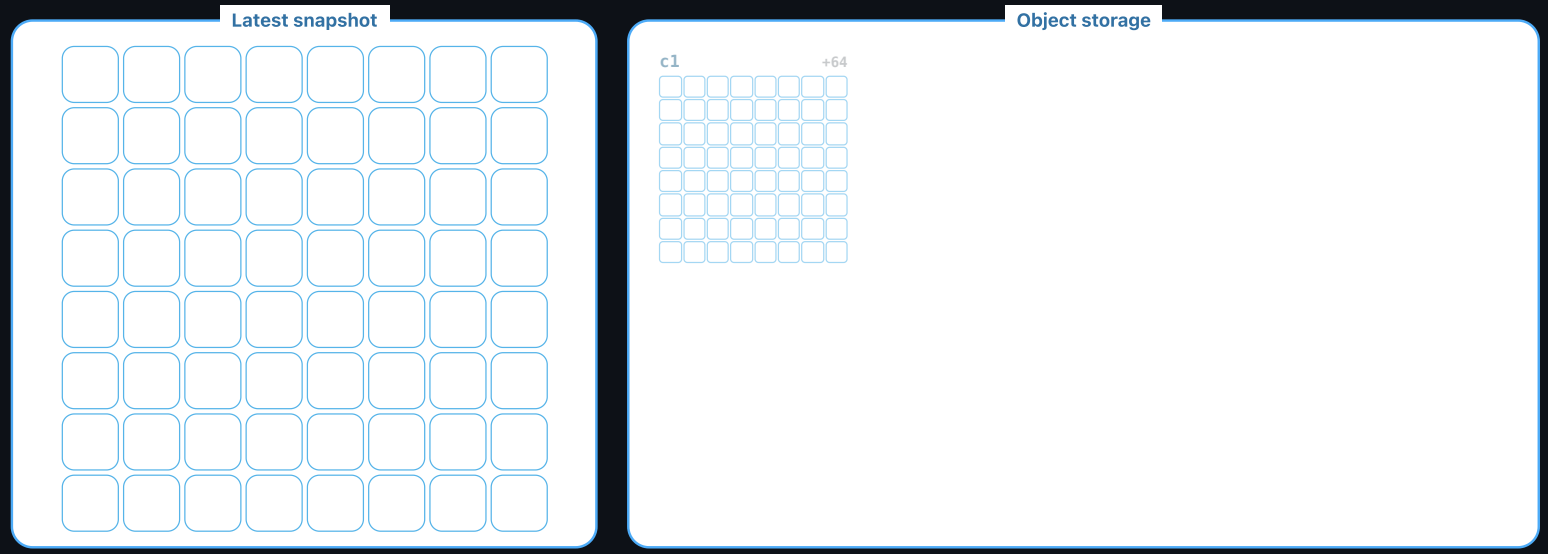
Implemented in Rust with a **high-performance I/O layer.**

https://icechunk.io




Cloud storage
ZARR
NO TRANSACTIONS
Writer
ICECHUNK
ACID
SESSION
MAIN #0
Zarr Reader
Is the cat alive?
???
Icechunk Reader
Is the cat alive?
Yes

STORAGE USED	FULL-COPY ALTERNATIVE	SAVED
4.0 GB	4.0 GB	—

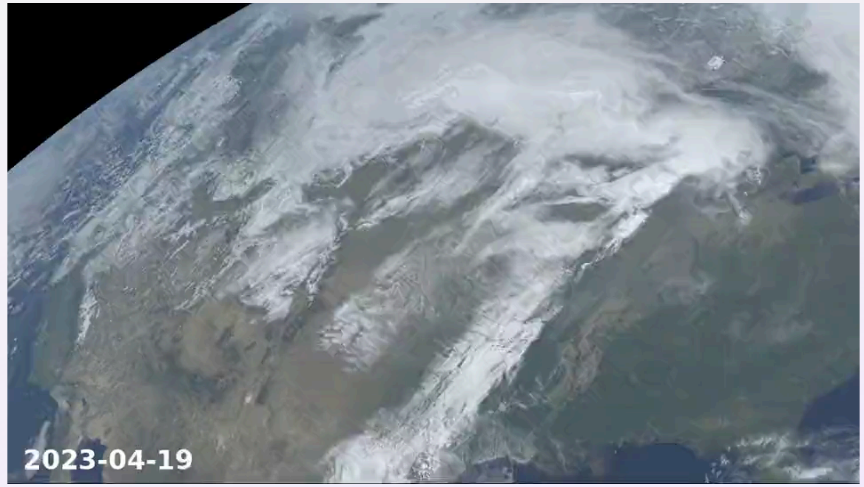


ICECHUNK

virtual chunks: bring the archives you already have

- A virtual chunk is a lightweight pointer, `(file url, byte offset, length)`, into an existing netCDF, HDF5, or GRIB file
- You get Icechunk versioning and cloud performance without copying or converting a single byte
- Proven at scale: the full GOES-16 satellite archive (about 115 TB in 380,000 netCDF files) indexed as over 7 billion virtual chunk references
- Existing archives can be indexed in place*

* reads go through the archive's original storage, so performance depends on it; hot subsets can be materialized later



NATIVE READER BILL

$0

< 1 s

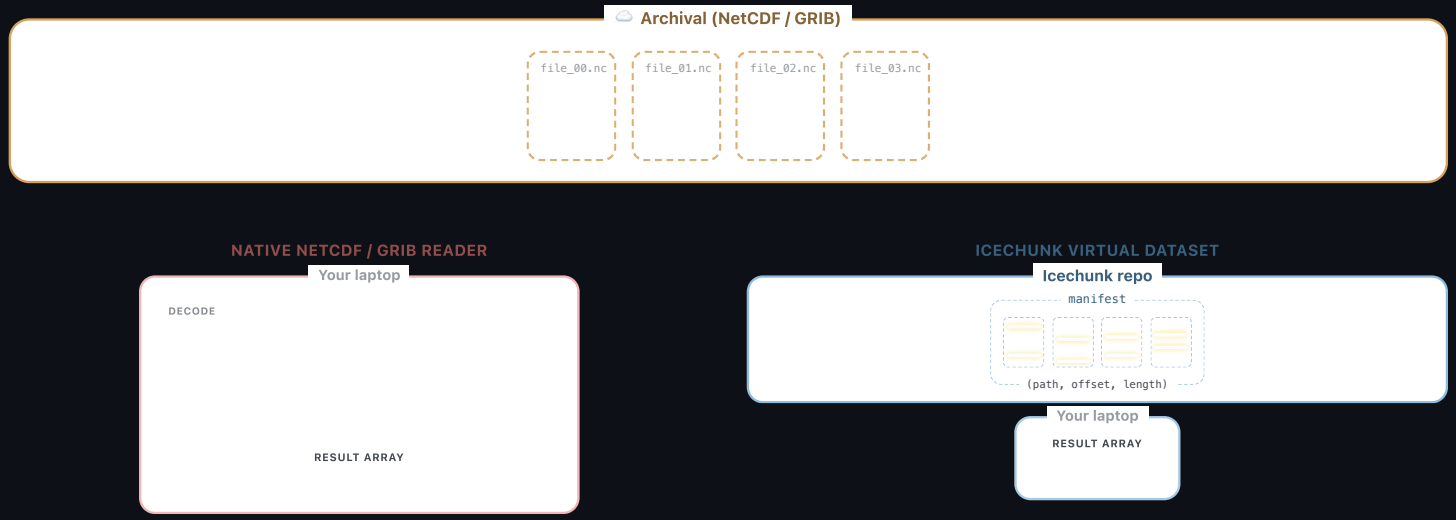
0 B downloaded

VIRTUAL CHUNKS BILL

$0

< 1 s

0 B downloaded



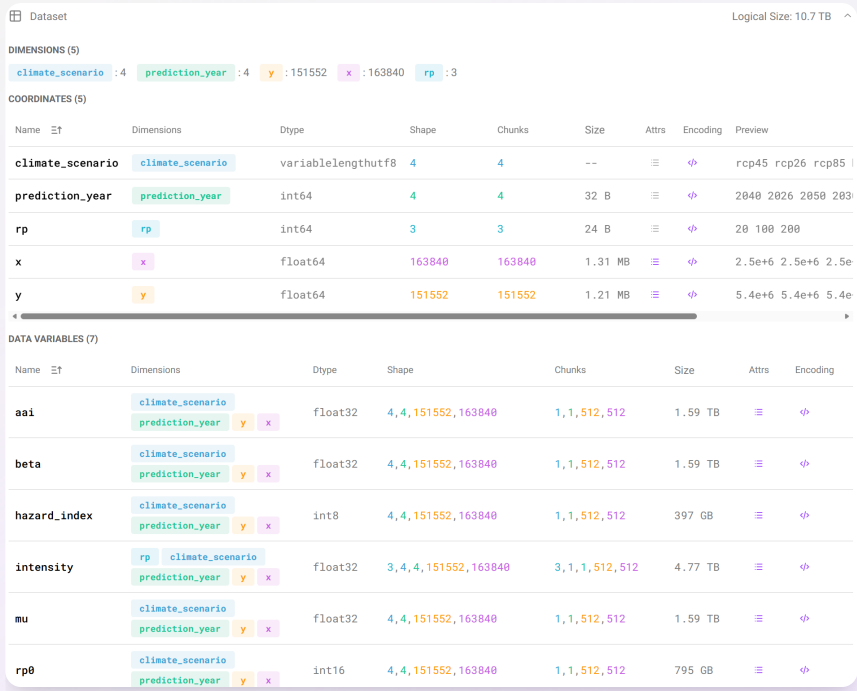
ARRAYLAKE

your data, organized and shared

ARRAYLAKE

a catalog for your team

- Organizations, repositories, branches and commits, all browsable in a web app
- Searchable metadata instead of README archaeology
- Access control per repository, plus API tokens for scripts and apps
- Your ARIA data stays readable after the programme ends



A 10.7 TB climate risk dataset in the Arraylake catalog

ARRAYLAKE

one login, any cloud

No more juggling S3 keys and tokens per storage system. Log in once; Arraylake vends short-lived, scoped credentials for wherever the data lives:

```python
import arraylake as al
import xarray as xr

client = al.Client()
client.login()  # once

# open data from wherever you have internet
ssp126 = client.get_repo("earthmover-aria/CMIP6.….ssp126.…ta")
ssp245 = client.get_repo("earthmover-aria/CMIP6.….ssp245.…ta")
ssp585 = client.get_repo("earthmover-aria/CMIP6.….ssp585.…ta")
tipmip = client.get_repo("earthmover-aria/tipmip-demo")

for repo in [ssp126, ssp245, ssp585, tipmip]:
    ds = xr.open_zarr(repo.readonly_session("main").store)
    ds.ta.isel(plev=0).mean(["lon", "lat"]).plot()
```

scenario	repo	data lives on	format
SSP126	JASMIN	Google Cloud	Zarr (virtual)
SSP245	JASMIN	AWS	netCDF (virtual)
SSP585	JASMIN	AWS	Zarr (virtual)
TIPMIP	JASMIN	JASMIN	Icechunk

One plot, four repos, three clouds, three formats.

Under 1 minute on 32 cores in eu-west-2.

you
`arraylake auth login`
once

→

Arraylake
checks your permissions, vends short-lived credentials

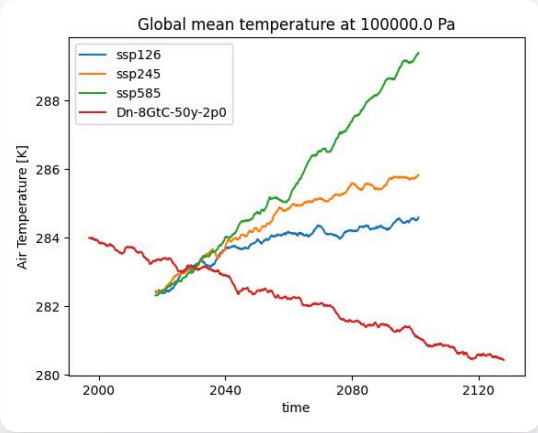
→

JASMIN aws CLOUDFLARE

ARRAYLAKE

one login, any cloud

No more juggling S3 keys and tokens per storage system. Log in once; Arraylake vends short-lived, scoped credentials for wherever the data lives:



scenario	repo	data lives on	format
SSP126	JASMIN	Google Cloud	Zarr (virtual)
SSP245	JASMIN	AWS	netCDF (virtual)
SSP585	JASMIN	AWS	Zarr (virtual)
TIPMIP	JASMIN	JASMIN	Icechunk

One plot, four repos, three clouds, three formats.

Under 1 minute on 32 cores in eu-west-2.

you
`arraylake auth login`
once

→

Arraylake
checks your permissions, vends short-lived credentials

→

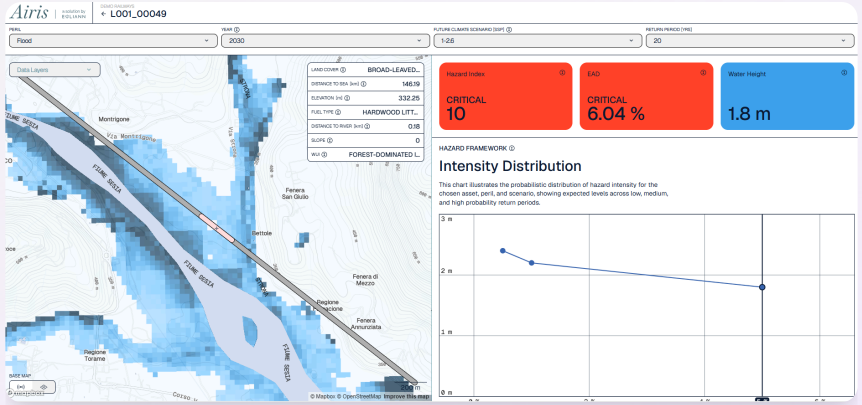
JASMIN aws CLOUDFLARE

ARRAYLAKE

flux: from repository to map or api in minutes

- Any dataset becomes a standard web service, with zero server code
- **EDR:** "this variable, at this point, at this time" over plain HTTP
- **WMS and map tiles:** drop into any web map or GIS tool
- **OPeNDAP:** MATLAB, ArcGIS, QGIS, no Python required
- Real model grids work: curvilinear, tripole, unstructured meshes

EDR = Environmental Data Retrieval, an OGC (Open Geospatial Consortium) standard web API.



A flood-risk platform whose map tiles come straight from Flux (from our blog)

MARKETPLACE

subscribe once, always up to date

FROM EXPLORATION TO PRODUCTION

building the early warning system, together

DATA SOURCES

your data, one platform

observations

Seismometer records from glaciers, hydrographic profiles, AMOC metrics, laser scintillometry

satellites & reanalysis

Sentinel-1 SAR (synthetic aperture radar), ice velocity maps, CARRA-2 Arctic reanalysis (2.5 km)

model output

CMIP6, UKESM and TerraFIRMA overshoot runs, TIPMIP (Tipping Points Modelling Intercomparison Project)

derived products

Early warning indicators, benchmarking metrics, ensemble statistics

live feeds

Marketplace subscriptions: ECMWF IFS forecasts, ERA5, HRRR

your experiments

Whatever your project produces next, from a few kilobytes of station data to petabytes of model output

SHARING

share your data, one step at a time

Sharing on Arraylake is not all or nothing. Widen the circle as your confidence grows:

a colleague

grant one person read access

→

your team

roles for a whole group at once

→

?

something between your team and the whole world

→

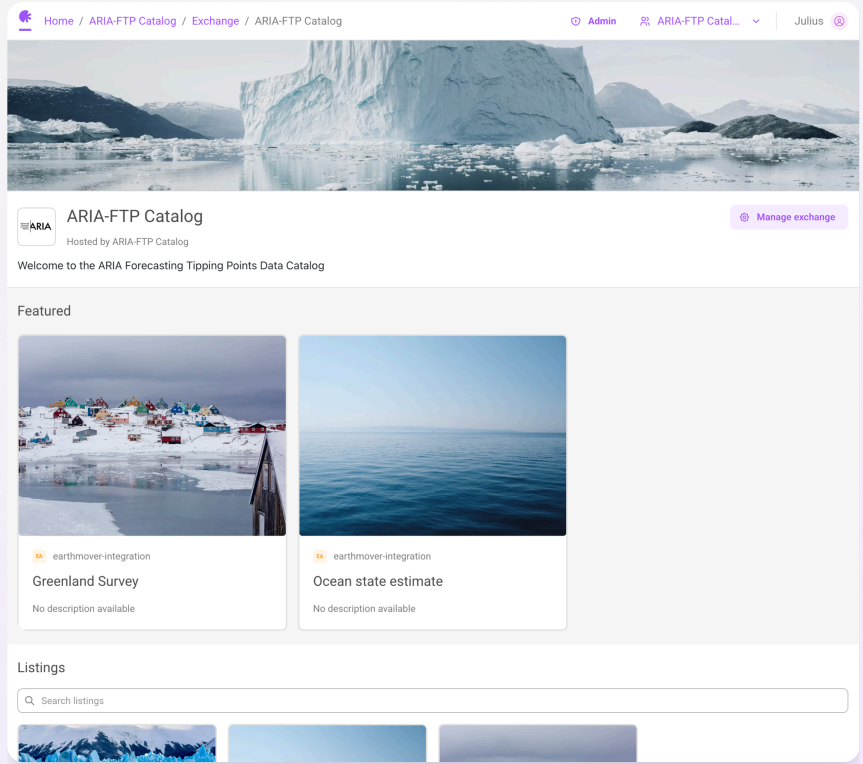
the marketplace

publish to the world, open or priced

SHARING

coming soon: the aria data exchange

- One catalog for the whole programme, run by the programme
- Member teams choose **read** (subscribe) or **post** (publish) rights
- Listings stay members-only until you decide otherwise
- New marketplace features carry over to the exchange automatically



AFTER THE SCIENCE

what happens after the science?

- An early warning system needs more than papers: **dashboards, alerts, APIs.** Start building them early.
- The same repository serves your notebook and your production app
- **Vibe coding is your friend:** AI assistants write the boilerplate, and the Arraylake MCP server lets them browse and query your data directly
- This dashboard reads live from the VERIFY team's repo and updates on every commit

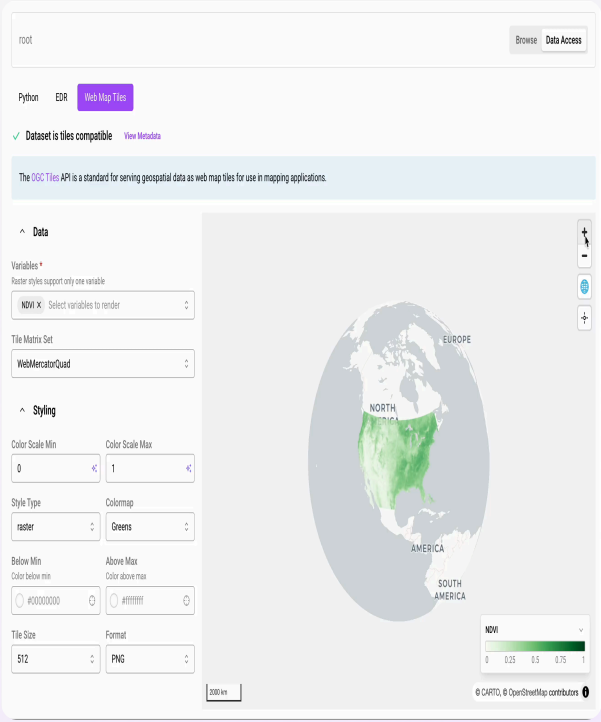
huggingface.co/spaces/jbusecke-earthmover/aria-verify-demo

MCP = Model Context Protocol

AFTER THE SCIENCE

one repository, every workflow

- The repository you commit to during research is the same one your apps read from later
- **Flux takes the hard part off your plate:** serving web tiles, answering point queries, scaling to many users
- Your app just calls a URL. An API token is all it needs.
- Rebuild or throw away apps as often as you like. The data underneath stays put.



NEXT STEPS

what you get from us, and how to start

your own org

an Arraylake organization for your team, set up together

storage

the JASMIN object store, or a managed bucket to start

the simulation catalogue

programme-wide data discovery

support

onboarding and engineering help. That's my job.

1. **Create or join your team's Arraylake org.** We set it up with you.
2. **Connect storage:** the JASMIN object store, or a managed test bucket from us.
3. **Run the quickstart** at docs.earthmover.io and commit your first dataset.
4. **Book an onboarding session**, or just message your team's Slack channel.

Interested in ingesting data? Let us know. We are looking for folks to help with TIPMIP.

Questions, requests, complaints: **julius@earthmover.io**

earthmover

earthmover.io · icechunk.io · docs.earthmover.io

julius@earthmover.io