Open science the cloud-native way

Tom Nicholas
(and many others)

tom@earthmover.io
@TomNicholas

About me

2012 Undergrad physics at St Catz

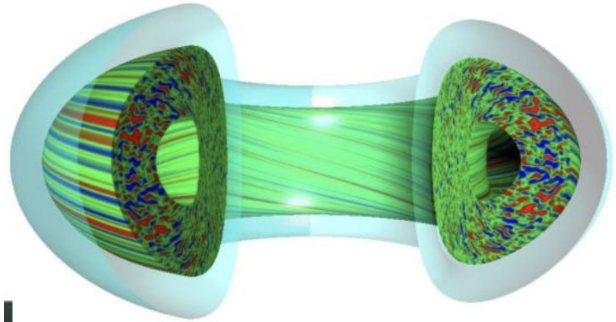
2016 Ph.D. in Plasma Physics for Nuclear Fusion at CCFE

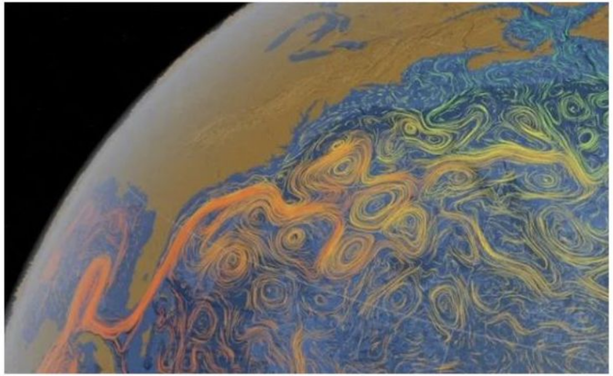
2018 Open-source contributor (Xarray maintainer)

2021 Pivot to oceanography at Columbia Uni.

2023 Joined [C]Worthy (non-profit)

2025 Joined Earthmover (startup)

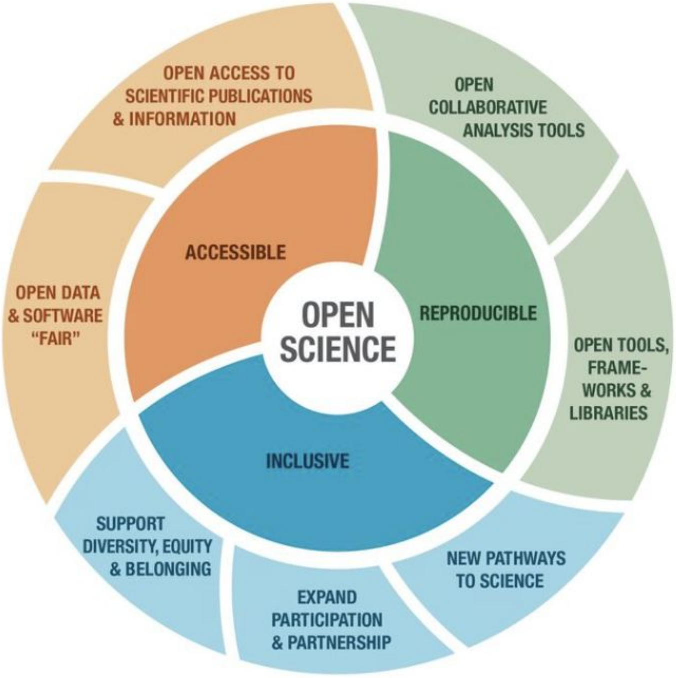




What I want to convince you of:

- Cloud services are the future of open science
- Cloud object storage is the future of scientific data distribution
- Much of science can utilize the same architecture and software
- Still some platform services needed to fully realize the dream

THE OPEN SCIENCE VISION



https://earthdata.nasa.gov/esds/open-science

```
for 👩🏽‍🔬 in everyone_in_the_world:
    for 👨🏻‍🔬 in everyone_else_in_the_world:
        📄 = (👩🏽‍🔬 + 👨🏻‍🔬).collaborate()
```

This would transform the 🌎 by allowing all of humanity to participate in the scientific process.

What are the barriers to realizing this vision?

❤️ 🥰 FAIR 🥰 ❤️

FAIR = Findable, Accessible, Interoperable, Reusable

FAIR is great. Nobody disagrees with FAIR.

But making data-intensive scientific workflows FAIR is easier said than done. FAIR does not specify the protocols, technologies, or infrastructure.

FAIR is not a platform.

An abbreviated history of meteorological data sharing

- We’ve been trying to do open science for a while
  - *it is confidently hoped that […], there will always be* ***a full and free interchange of materials [data]****, and a frequent and friendly intercourse between the departments: for it is evident that much of the success of the plan proposed will depend upon this interchange, …”*
    - Lt. Matthew Fontaine Maury, 1853, Brussels
- tl;dr: each new data sharing technology changes science
  - Telegraph
  - Internet-connected servers
  - Cloud??

cloud | open data

Cloud-native platforms are the natural evolution of atmosphere-ocean open-data practice

June 9, 2025

Open-data practice in ocean/atmosphere sciences is approximately 170 years old! While it is easy to exclaim, "weather/climate are global, of...

Read more

Deepak Cherian
Forward Engineer

The dream: Frictionless reproducibility

- Donoho (2024) - “Data science at the singularity”
  - “**frictionless open services**, essentially offering immediate, permissionless, complete access to each relevant digital artifact, programmatically from a single line of code.”
  - “Empirical machine learning is today’s leading adherent field; its hidden superpower is adherence to frictionless reproducibility practices”

Weights & Biases

GitHub

NIGHTINGALE OPEN SCIENCE

kaggle

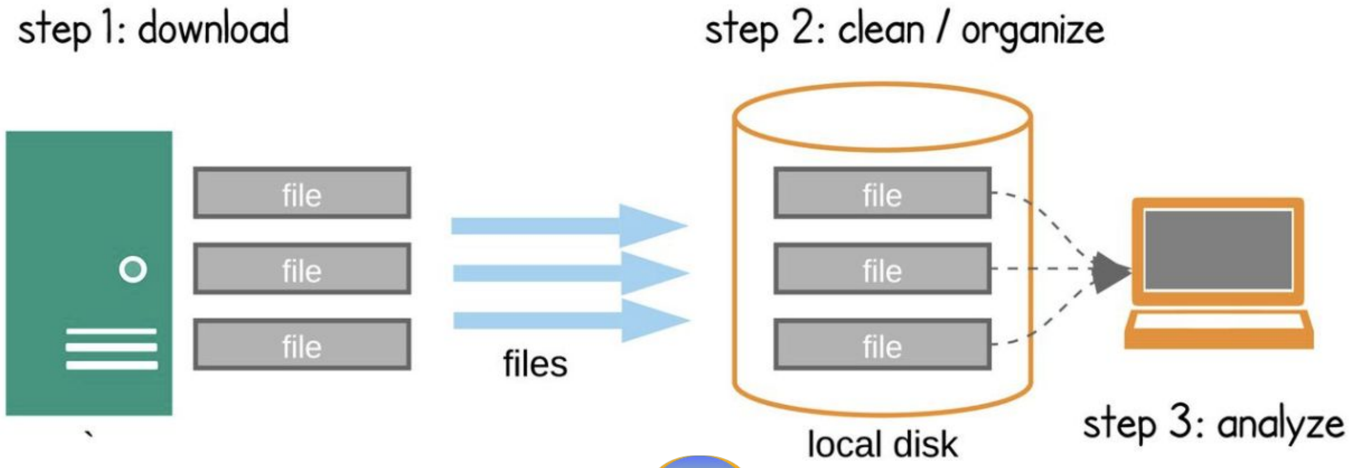
jupyterhub

OSF

CodaLab Competitions

Big data is big hard

- Hard to distribute
- Hard to analyse
- The download model



-

Cloud as a revolutionary technology

- Object storage solves the big data distribution problem
- Scalable on-demand scalable compute makes big data analysis possible
    - Key: Separation of storage and compute

Why servers suck for sharing data

- Falls over under heavy load
- Requires FTE just to keep it running
- No standardization of output

Actual image of THREDDS when I ask for a PB of environmental data

Object storage is great for sharing data

- Scales storage almost indefinitely
- Scales throughput almost indefinitely
- Always-on
- Access from anywhere
- Cloud provider runs the infrastructure for you

ceph

Google Cloud Storage

Amazon S3

Microsoft Azure
Blob Storage

10
01

CLOUDFLARE
R2 SUPPORT

“Cloud-native data repositories”

- Sharing data via object storage is simple

Step-by-Step Guide to Building a Big Data Portal

Ryan Abernathey Follow 8 min read · Jun 12, 2018

- Step 1: Place your Big Data in cloud object storage in a self-describing, cloud-optimized format.
- Step 2: You’re done. There is no step 2.

Requires cloud-optimized data formats

What is "Cloud Optimized"?

- Compatible with object storage (access via HTTP)
- Supports lazy access and intelligent subsetting
- Integrates with high-level analysis libraries and distributed frameworks

cloud native | fundamentals | icechunk

Fundamentals: What is Cloud-Optimized Scientific Data?

April 17, 2025

Why naively lifting scientific data to the cloud falls flat. Scientific formats predate the cloud There are exabytes of scientific...

Read more

Tom Nicholas
Forward Engineer

I like Zarr

- Chunked, compressed, multidimensional arrays
- Highly scalable
- Self-describing
- Domain-agnostic

- Can even access other formats as if they were in Zarr (i.e. “virtual Zarr” / Kerchunk)

fundamentals icechunk zarr

Fundamentals: What Is Zarr? A Cloud-Native Format for Tensor Data

May 20, 2025

Why scientists, data engineers, and developers are turning to Zarr Often the biggest bottleneck in your workflow isn’t your code...

Read more

Lindsey Nield
Software Engineer

But what about databases?

- Traditional serverful databases have some very useful properties:
  - Version control (“time travel”)
  - Schema evolution (“alter however you like”)
  - Isolatable transactions (”multiplayer mode”)
    - Crucial: Allows scientist to read data *whilst provider is updating it*

PostgreSQL

Icechunk / Iceberg provide database features

- Provide
  - Version control
  - Schema evolution
  - Transactions

- All just via object storage, no running server required!!!

In other words, Icechunk is like “Git for Zarr”

Cataloguing the buckets

- Still need to find all the object storage bucket URLs
- Need a *catalog*
- Can build yourself
- Or buy
    - e.g. Earthmover’s ArrayLake platform

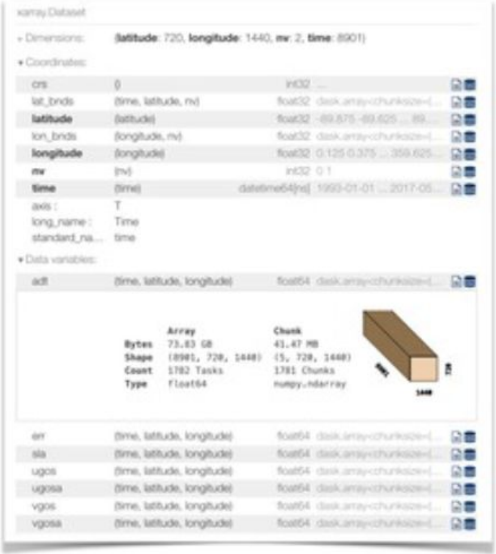
Data Catalog

Arraylake provides a central catalog for your organization's scientific data. With a native understanding of multidimensional array data and the Zarr data model,

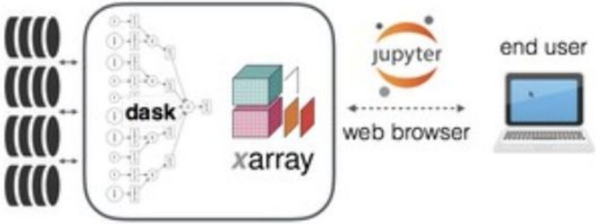
Arraylake makes it easy to know what data you have, how it got there, and how it's changing over time. To learn more about how data are organized in Arraylake, check out the Arraylake data model.

Cloud compute

Analysis-Ready, Cloud-Optimized Data



Data-Proximate Computing

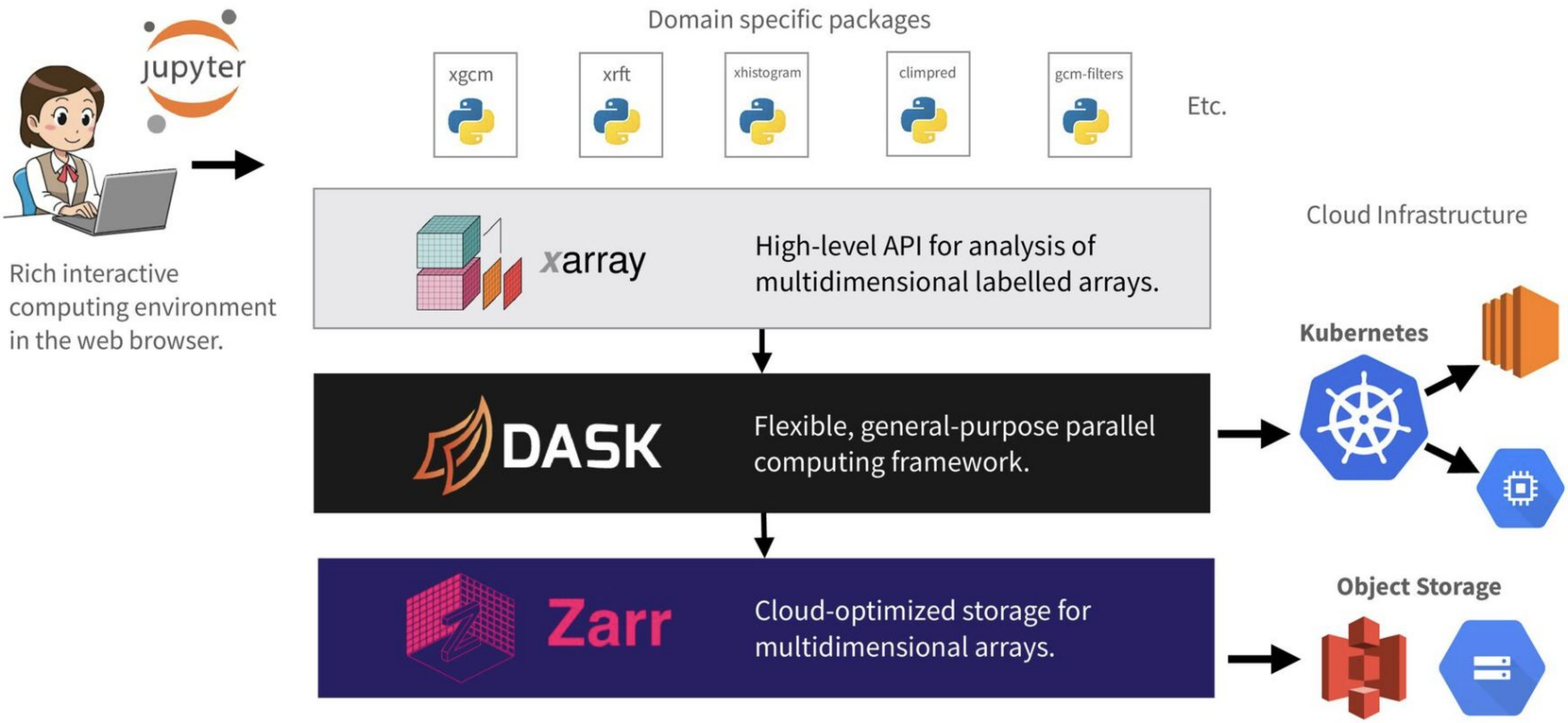


On Demand, Scalable Distributed Computing

Interactive computing environments

- Jupyter, IPython, RStudio

THE PANGEO CLOUD-NATIVE STACK

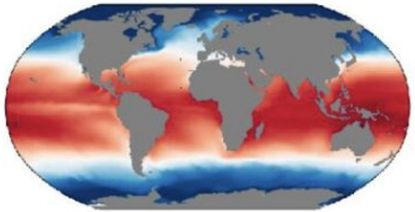


Pangeo community

- Academia
  - NASA, NOAA, USGS, NSF, NCAR
- Non-profits
  - 2i2c, CarbonPlan, [C]Worthy, ClimateMatch
- Companies
  - Earthmover, Coiled, Jupiter Intelligence

Community-driven development

Scientific users / use cases



- Define science questions
- Use software / infrastructure
- Identify bugs / bottlenecks
- Provide feedback to developers

Open-source software libraries

jupyter zarr xarray DASK

- Contribute widely the the open source scientific python ecosystem
- Maintain / extend existing libraries, start new ones reluctantly
- Solve integration challenges

GitHub

HPC and cloud infrastructure

- Deploy interactive analysis environments
- Curate analysis-ready datasets
- Platform agnostic

Google Cloud Platform Microsoft Azure aws

Not just for geoscience!

- Core libraries are domain-agnostic
  - (Xarray, Dask, Zarr, Icechunk…)
- Multidimensional arrays are very common abstraction
- Sees use in
  - Climate science
  - Meteorology
  - Oceanography
  - Glaciology
  - Satellite earth observation
  - Biomedical imaging
  - Genomics
  - Neuroscience
  - Radar astronomy
  - Fusion plasma physics
  - Finance
  - Machine Learning

Open for contributions!

- Openly licenced
- Contributions welcome!
  - (I got into this by fixing something in Xarray that annoyed me)

Takeaways

- Cloud services are the future of open science
- Cloud object storage is the future of scientific data distribution
- Much of science can utilize the same architecture and software

Bonus topics

- Xarray
- Zarr
- Icechunk
- VirtualiZarr
- Cubed and serverless array computing
- Cataloging