



A cloud-native data repository for ocean, weather, and climate science.

R Y A N A B E R N A T H E Y

WHO AM I?



Physical Oceanographer

Ph.D. From MIT, 2012

Associate Prof. at Columbia / LDEO

<https://ocean-transport.github.io/>

Core developer of *Xarray*

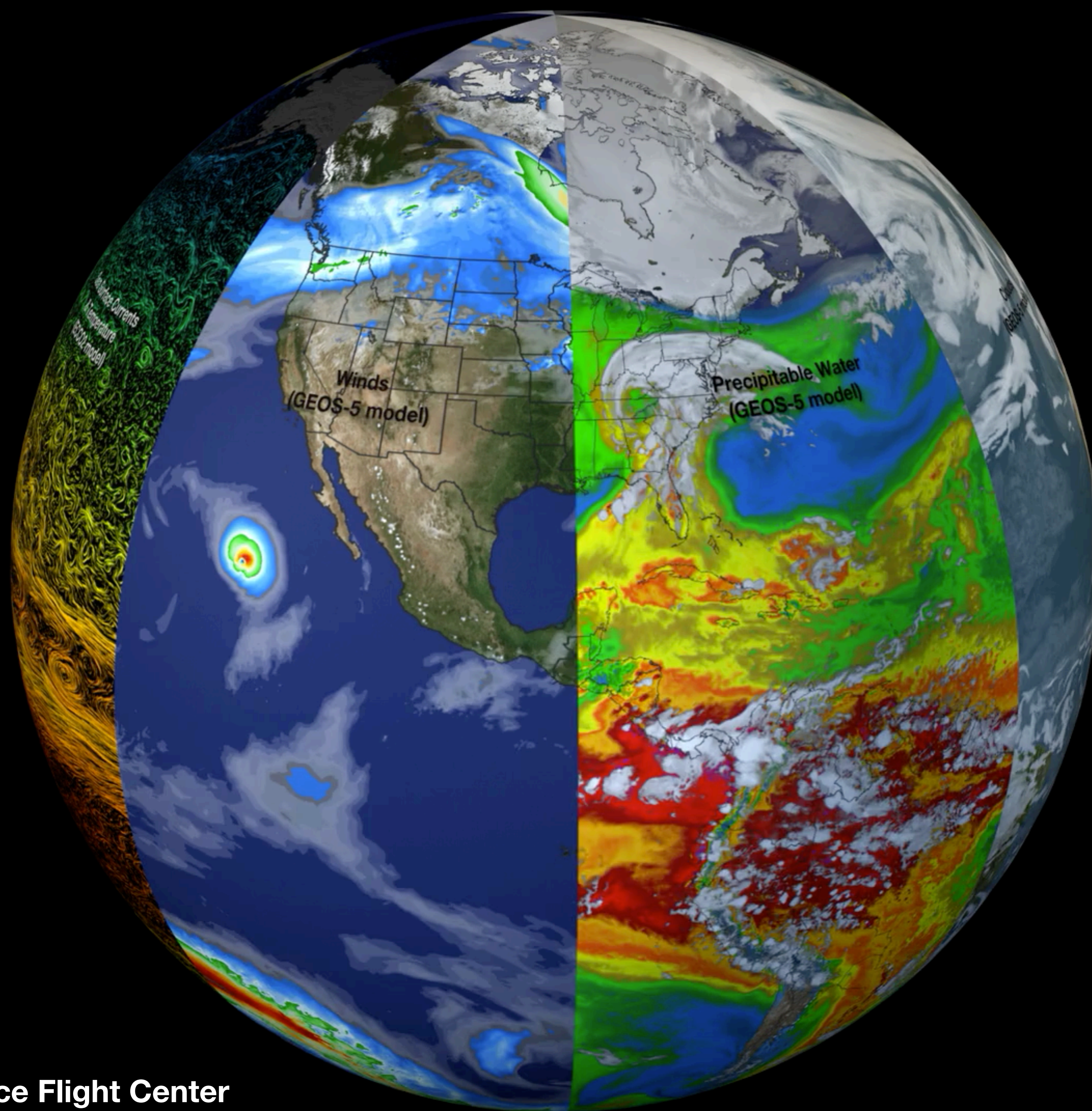
Core developer of *Zarr*

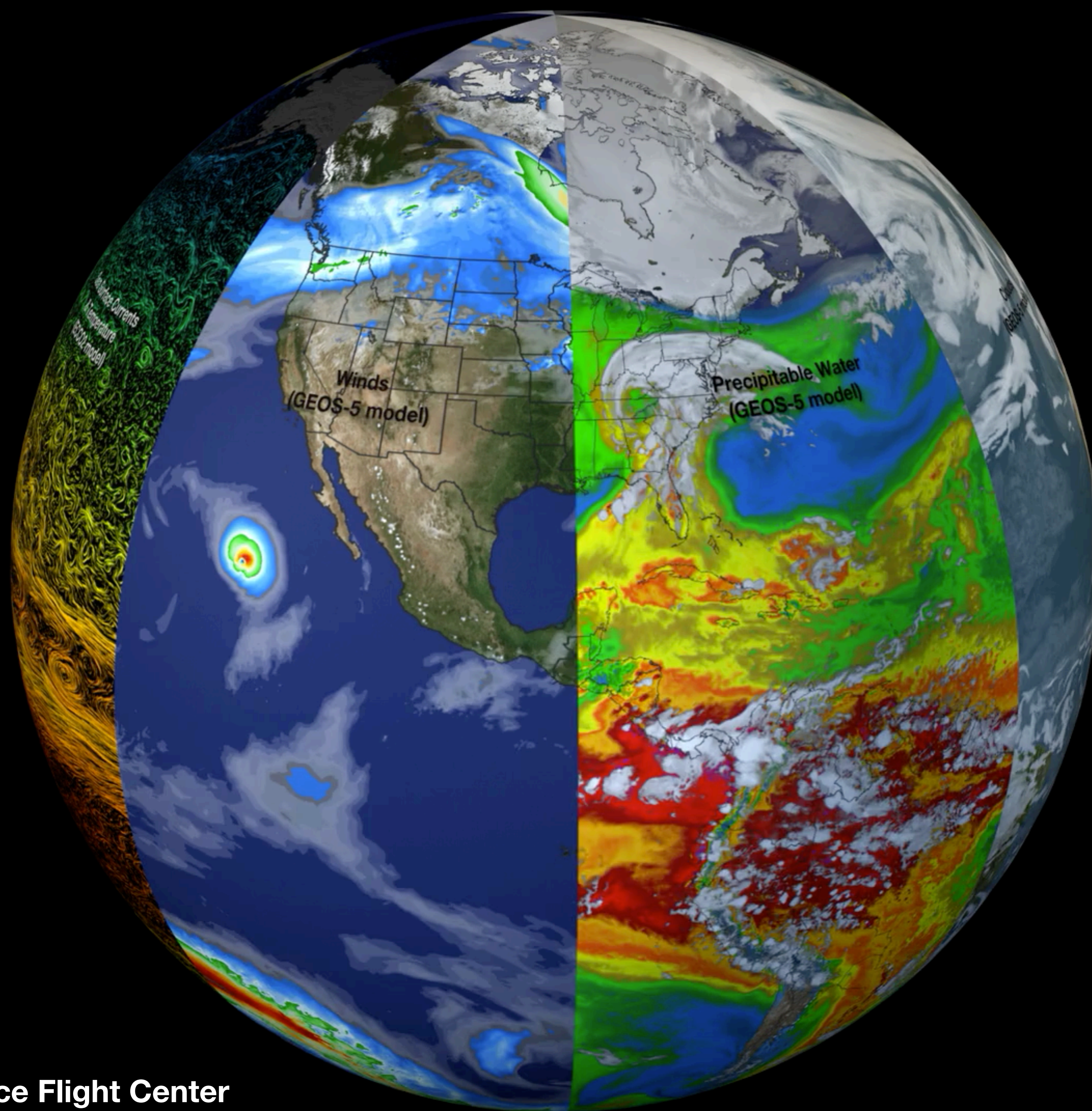
Co-founder of *Pangeo*

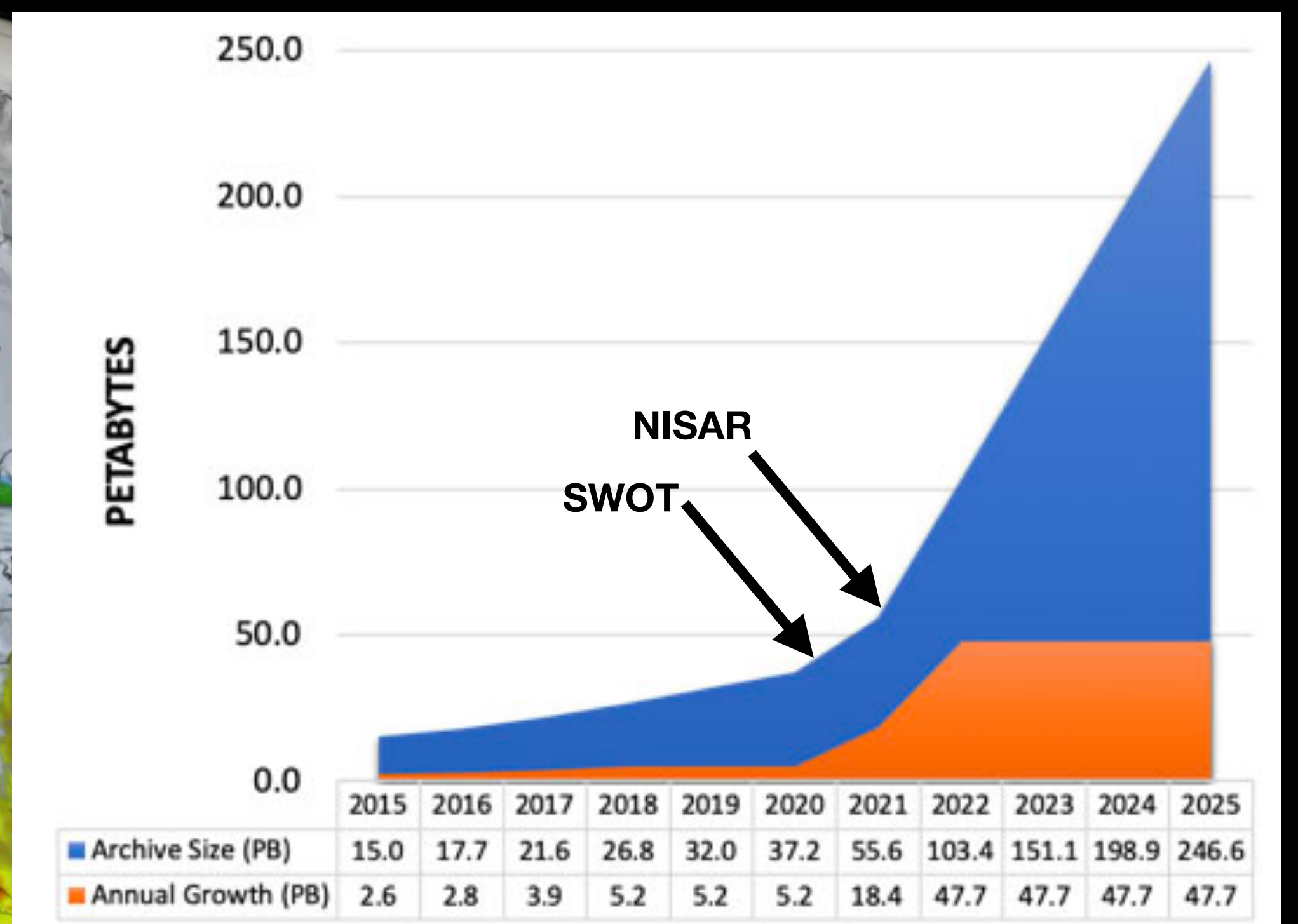
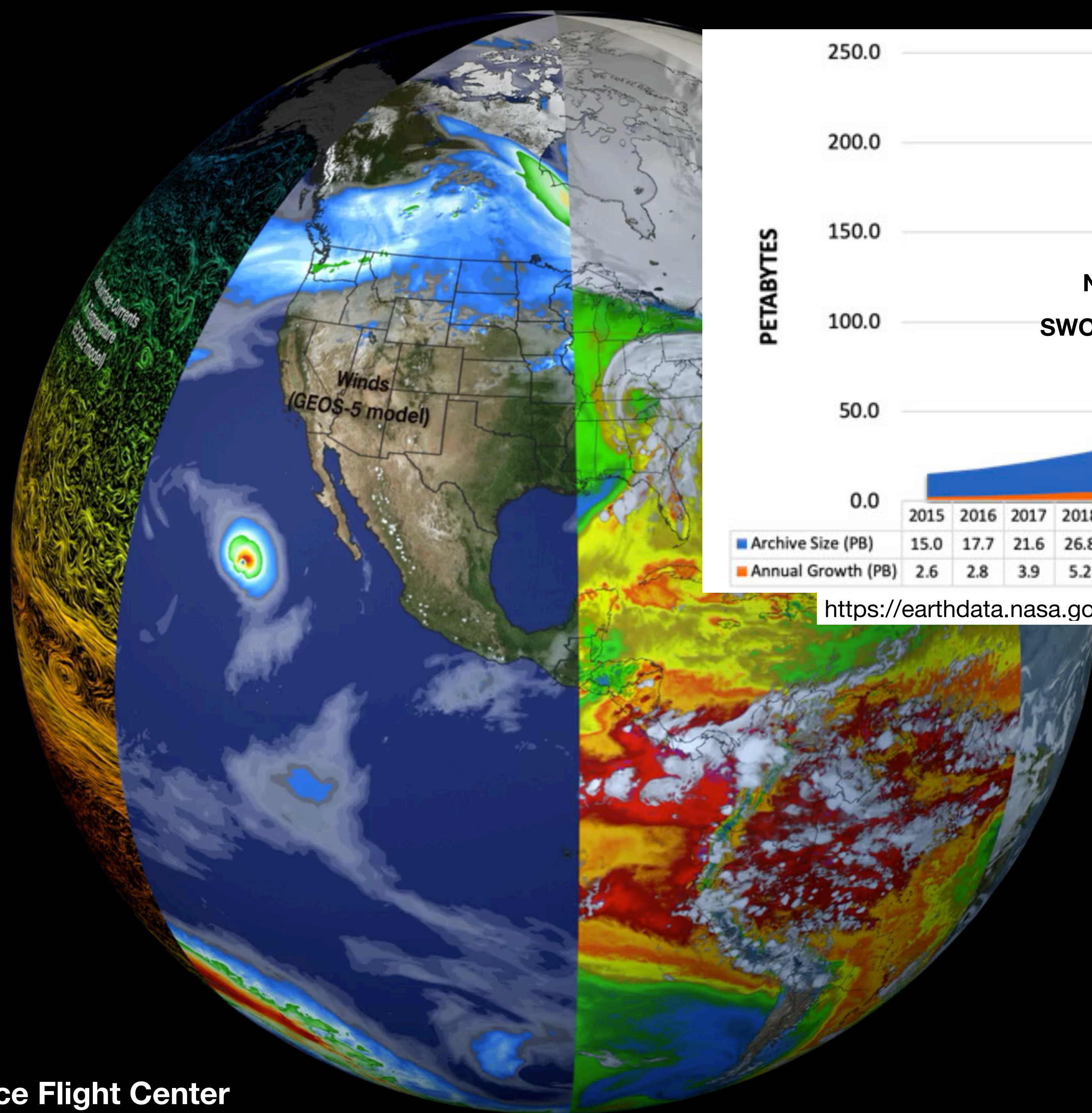
Open Source Advocate

THIS TALK

- *Problem:* Weather and climate data is large and complex 🤯
- *Solution:* Data-proximate computing in the cloud 😎
- *Problem:* Analysis-ready, cloud-optimized data is scarce 🤔
- *Solution:* **Pangeo Forge** 💪





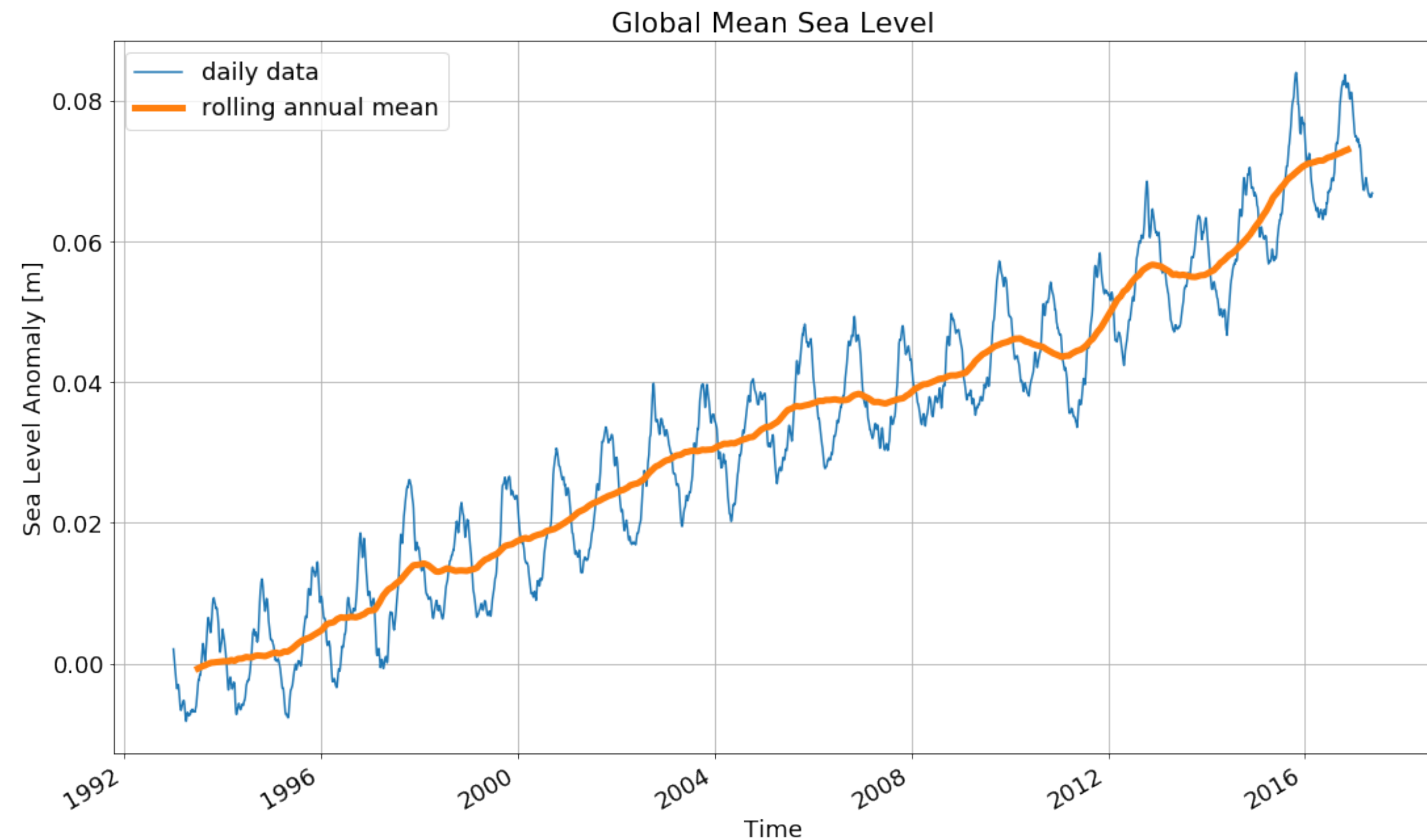


<https://earthdata.nasa.gov/eosdis/cloud-evolution>

WHAT SCIENCE DO WE WANT
TO DO WITH ALL THIS DATA?

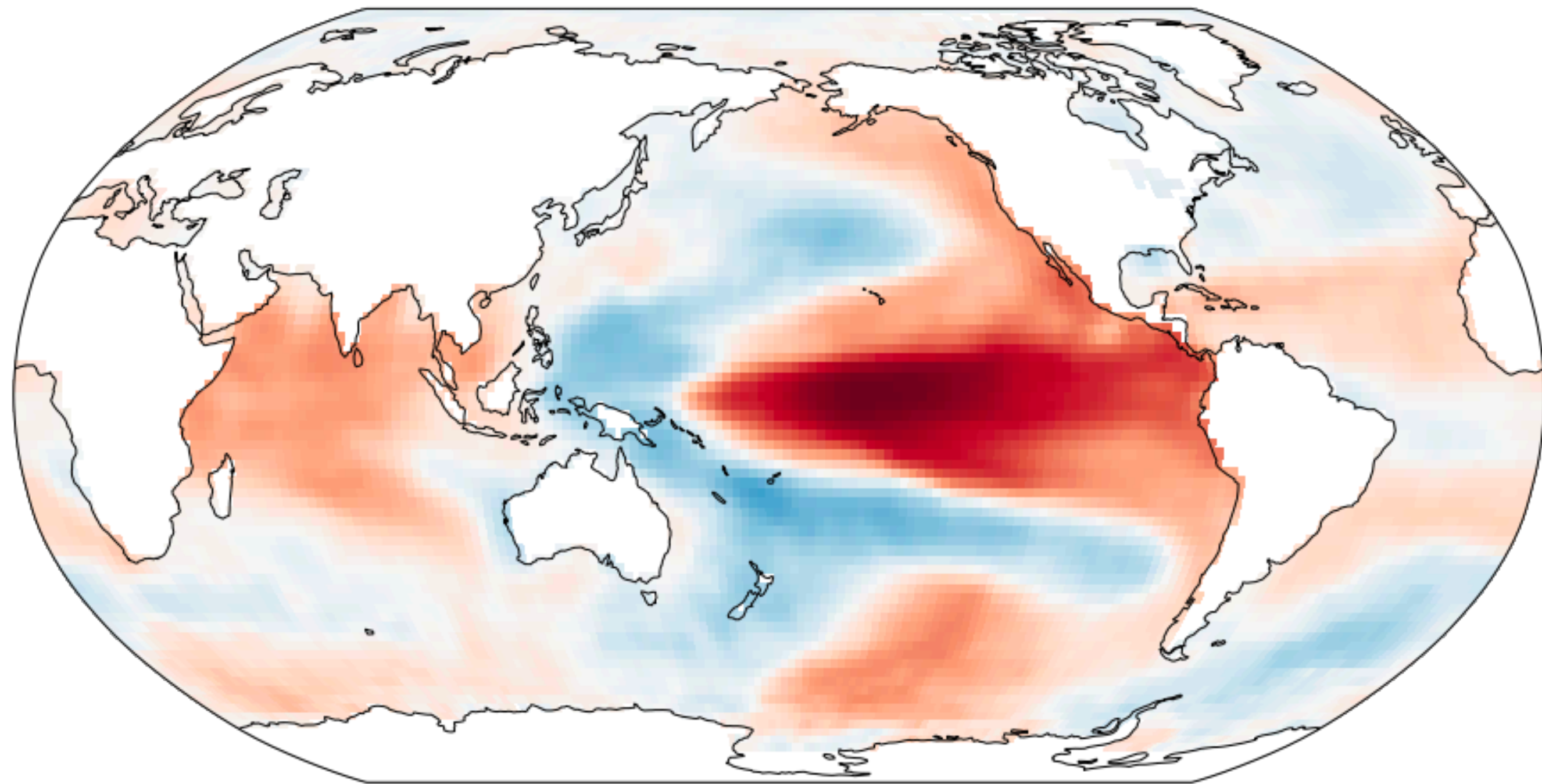
WHAT SCIENCE DO WE WANT TO DO WITH ALL THIS DATA?

Take the mean!



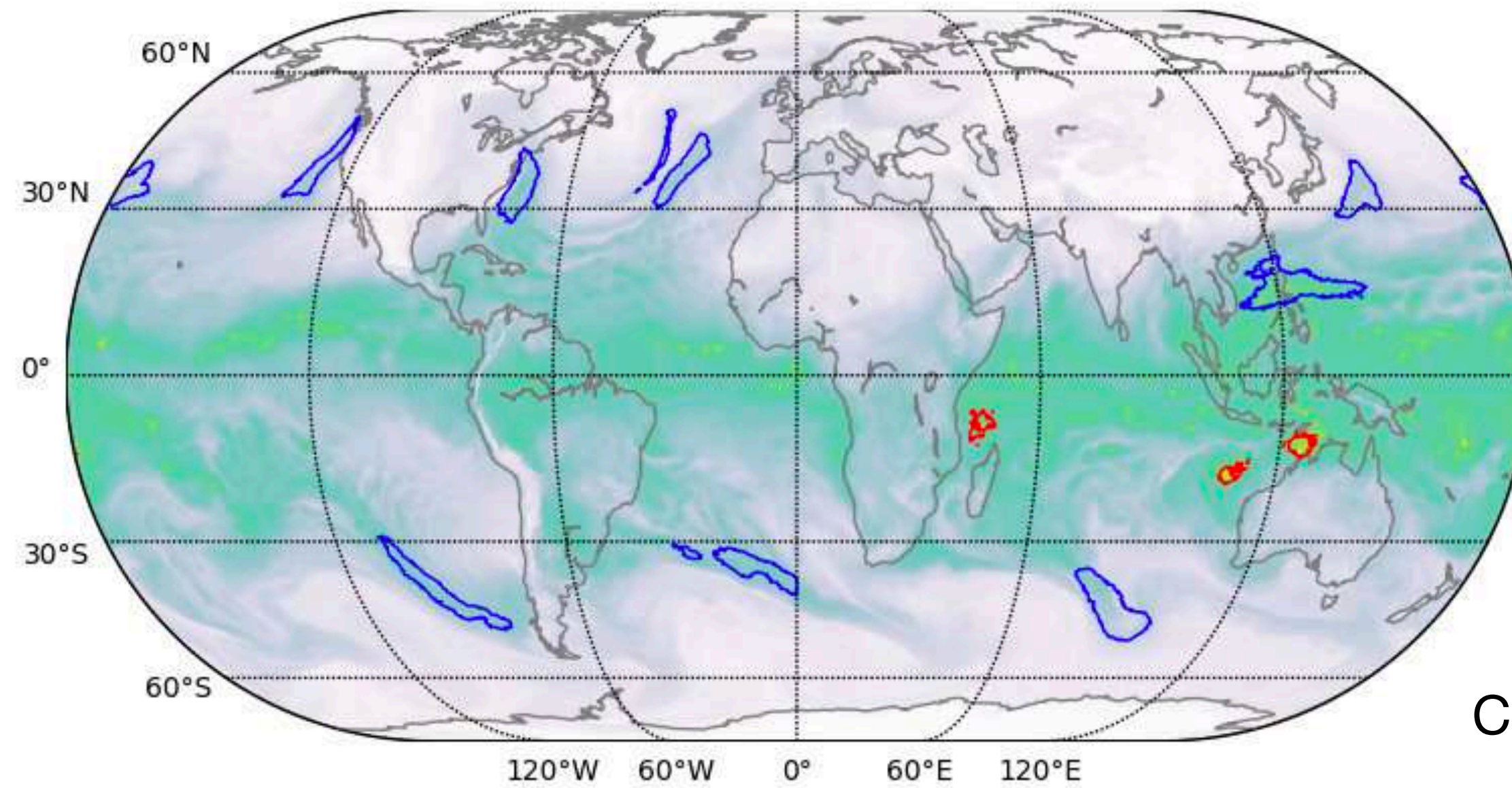
WHAT SCIENCE DO WE WANT TO DO WITH ALL THIS DATA?

Analyze
spatiotemporal
variability

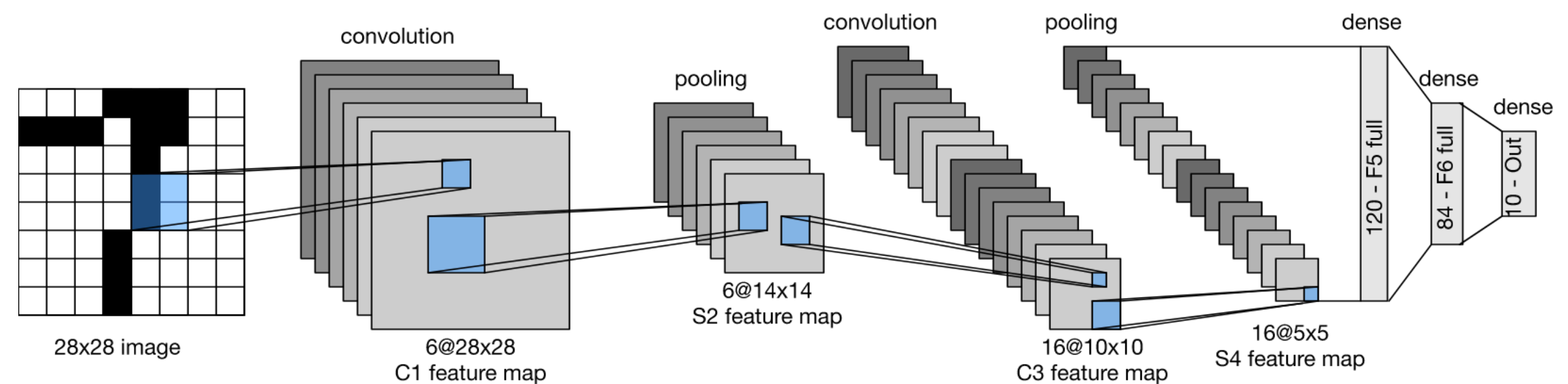


WHAT SCIENCE DO WE WANT TO DO WITH ALL THIS DATA?

Machine learning!



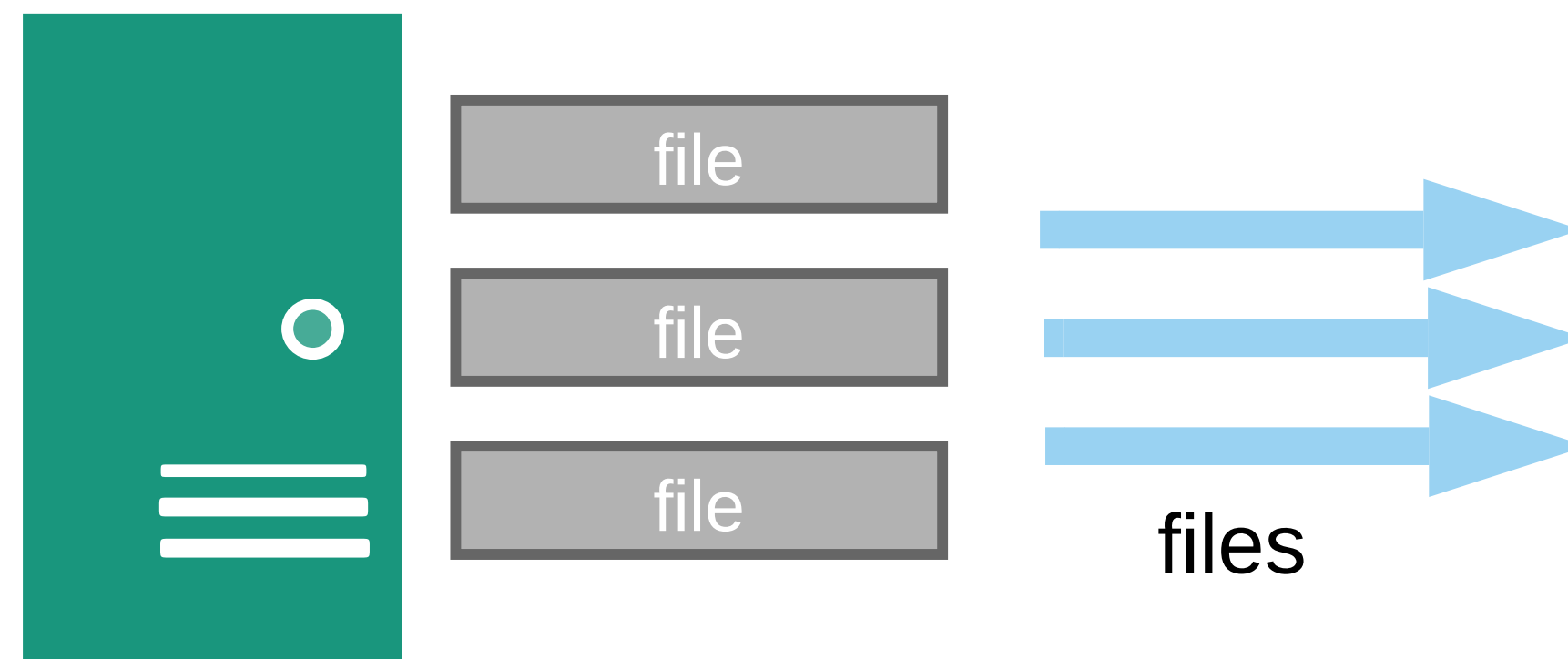
Credit: Berkeley Lab



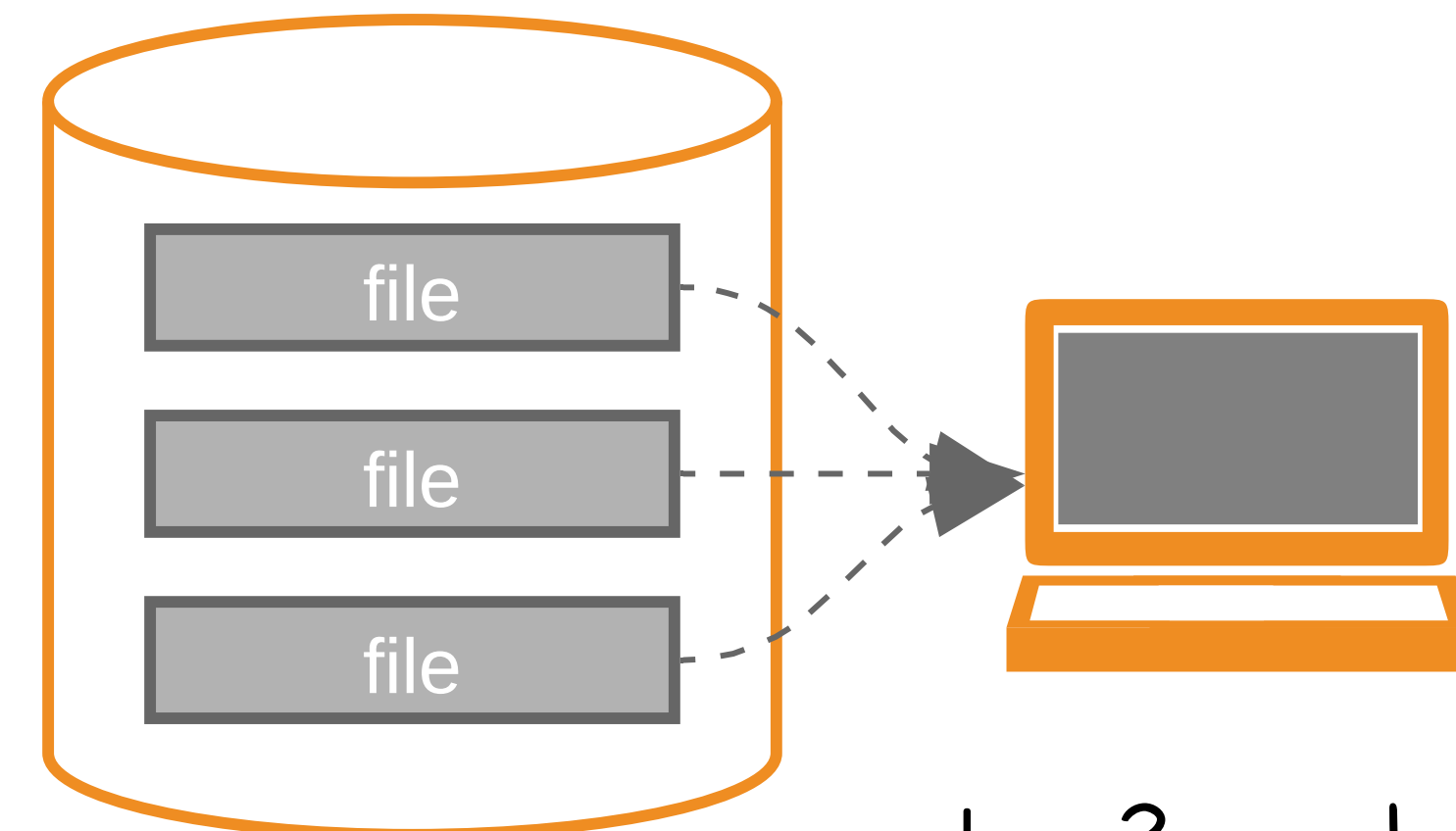
HOW?

THE "DOWNLOAD" MODEL

step 1: download



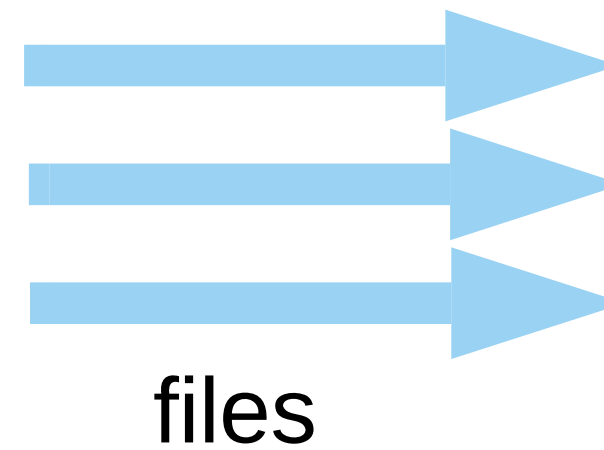
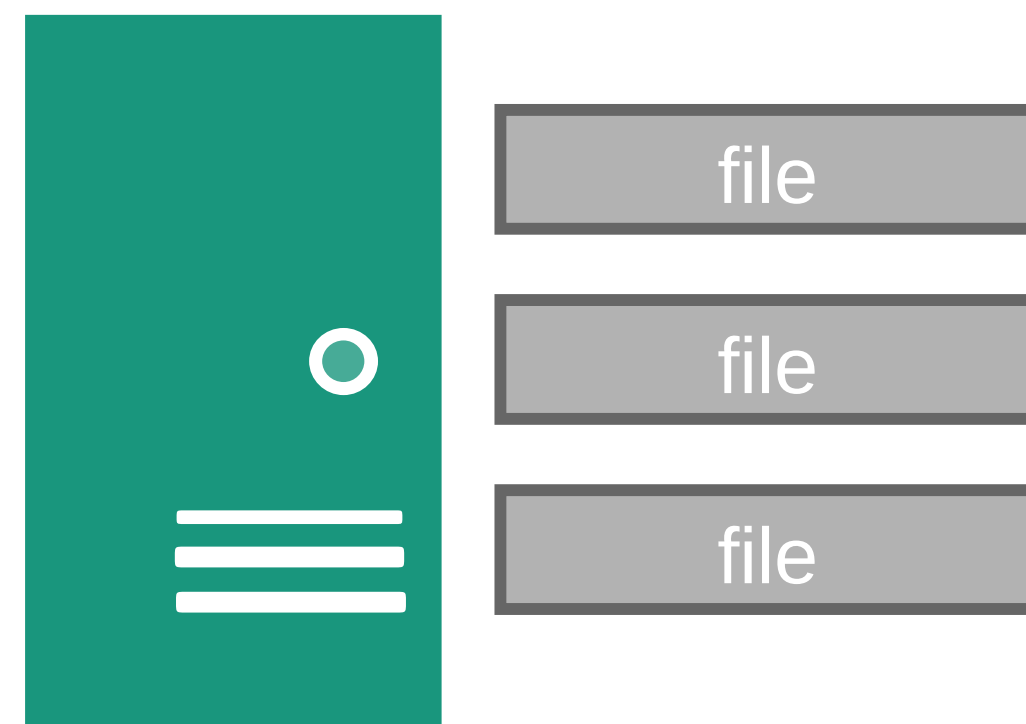
step 2: clean / organize



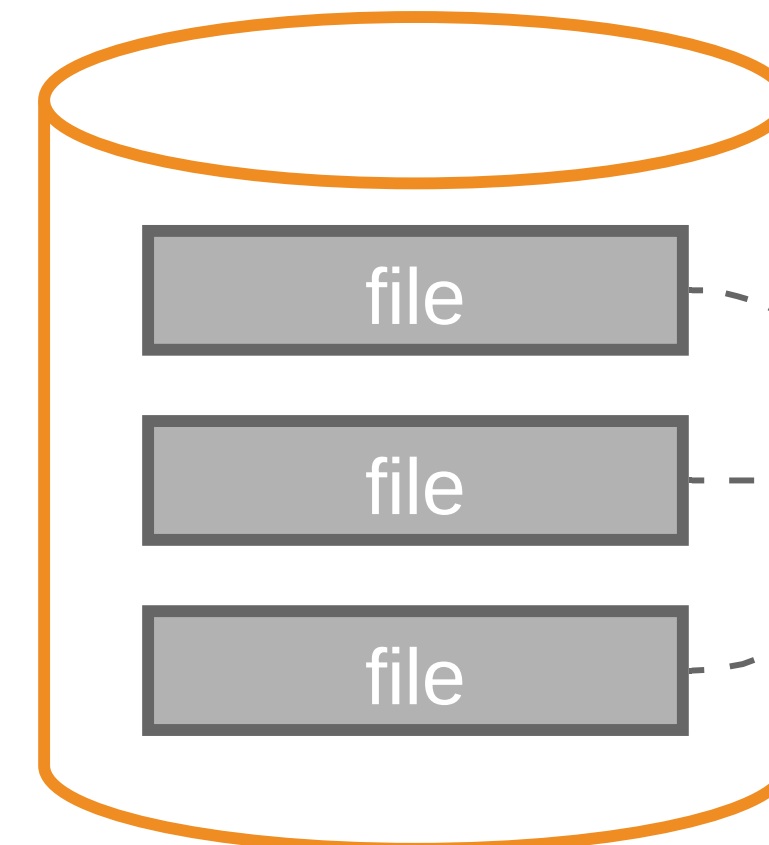
step 3: analyze

THE "DOWNLOAD" MODEL

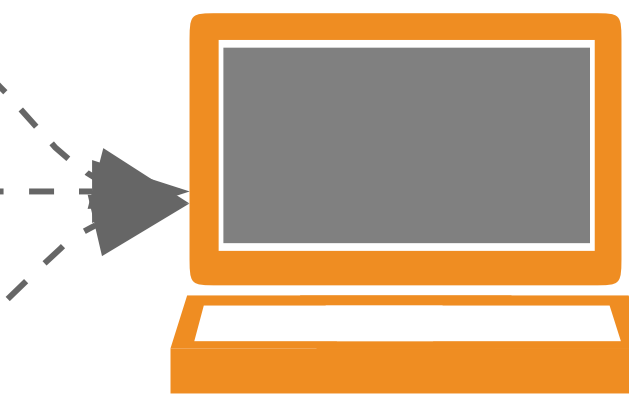
step 1: download



step 2: clean / organize



local disk



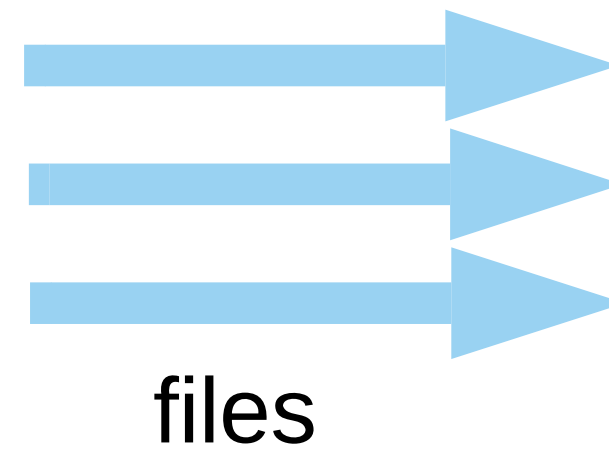
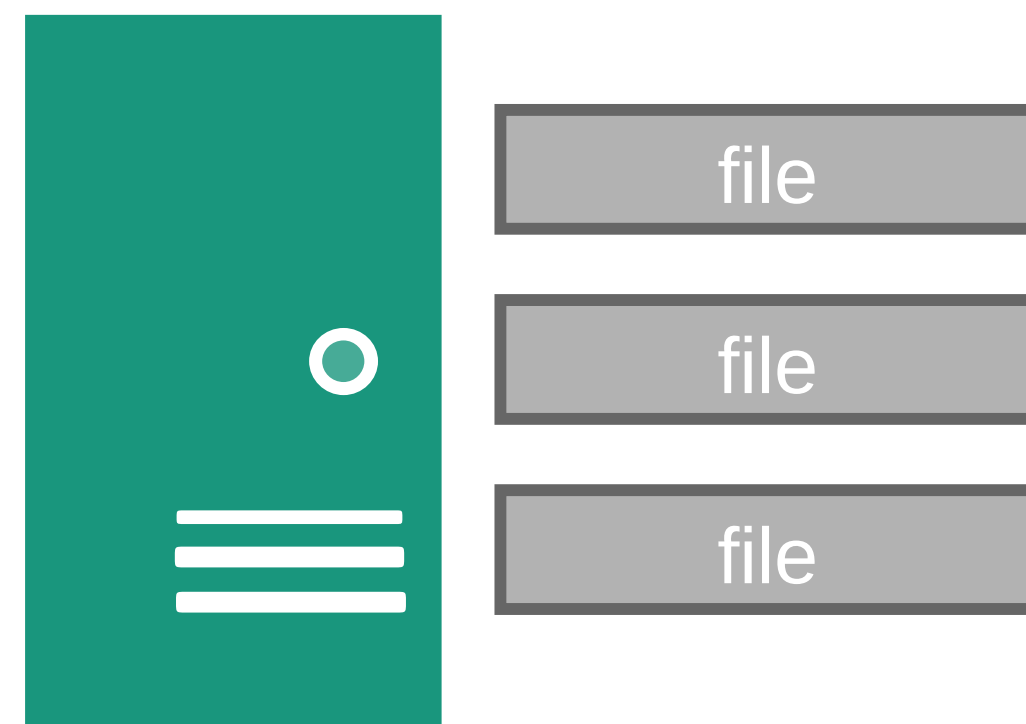
step 3: analyze

MB

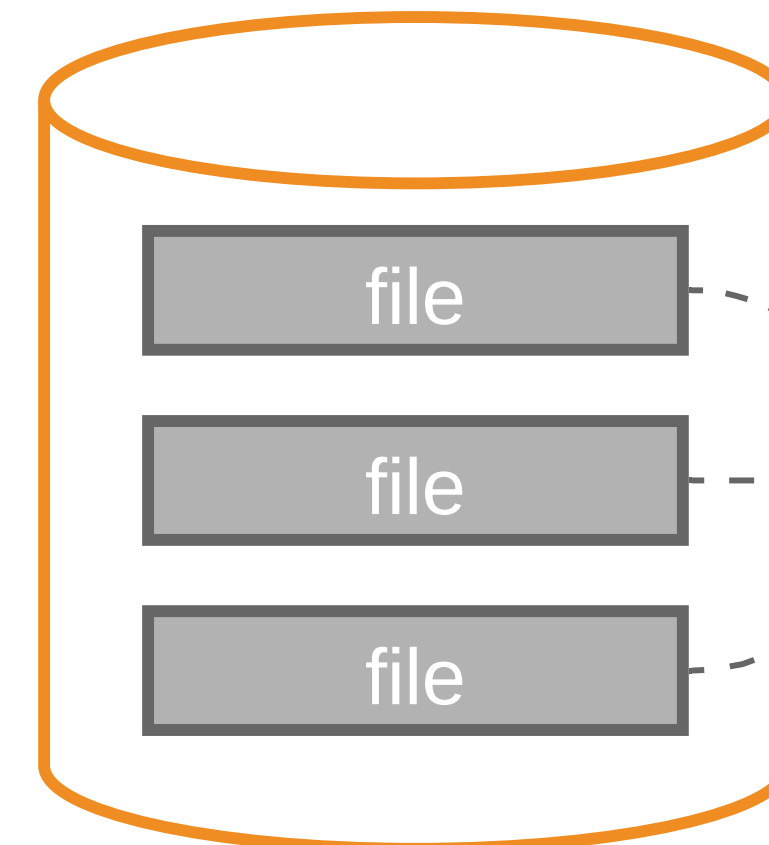


THE "DOWNLOAD" MODEL

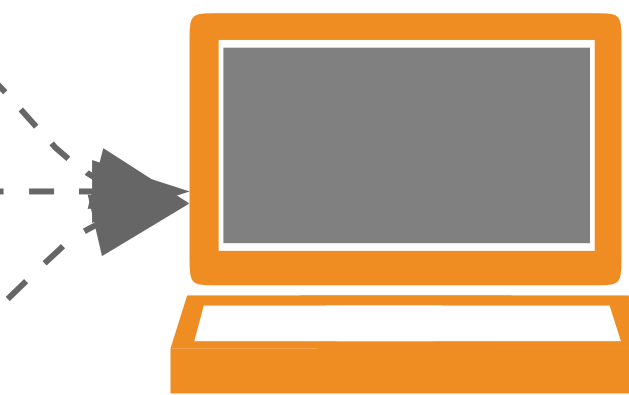
step 1: download



step 2: clean / organize



local disk



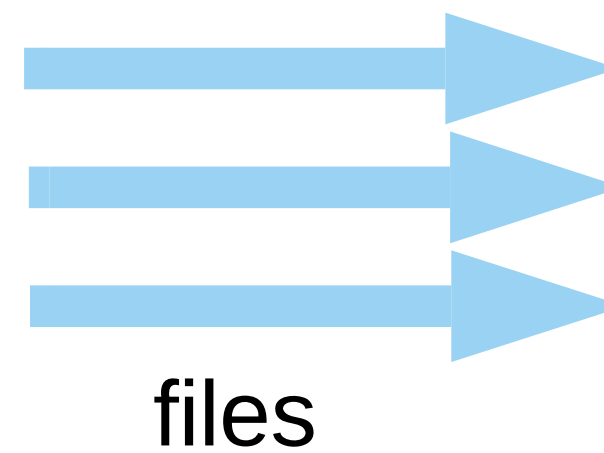
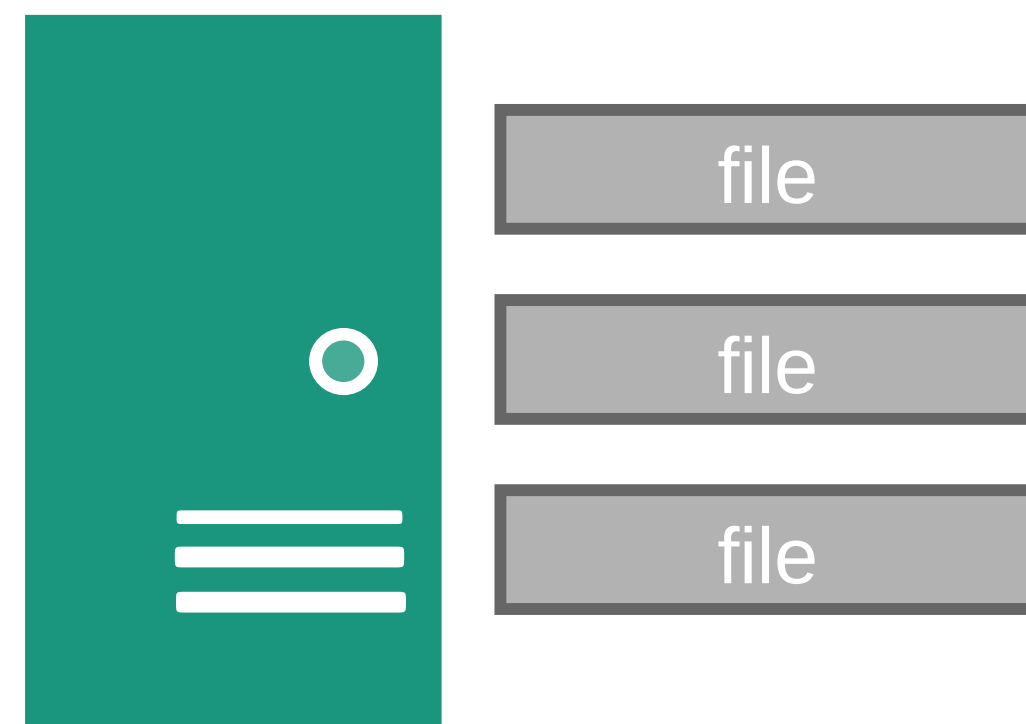
step 3: analyze

GB

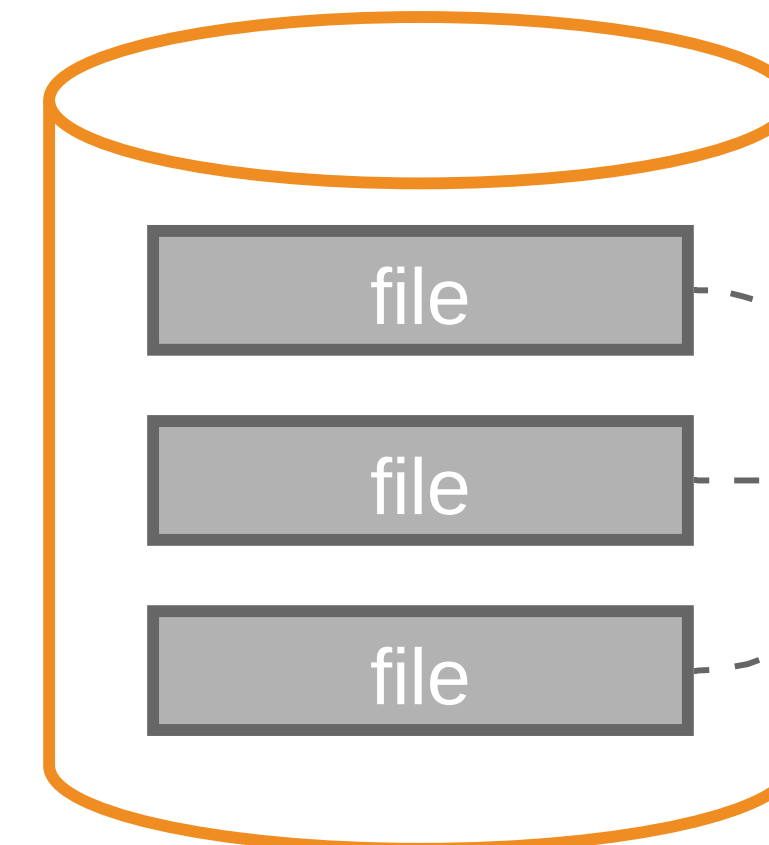


THE "DOWNLOAD" MODEL

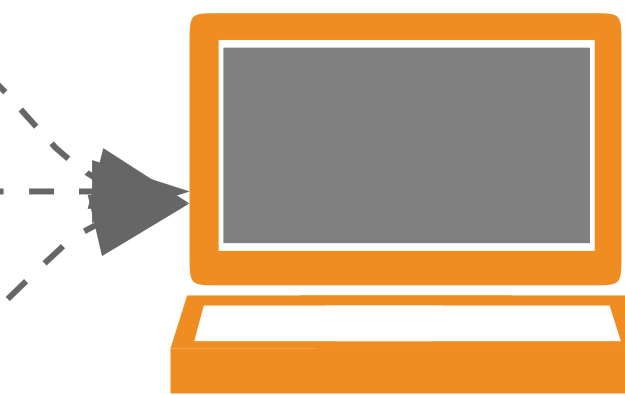
step 1: download



step 2: clean / organize



local disk



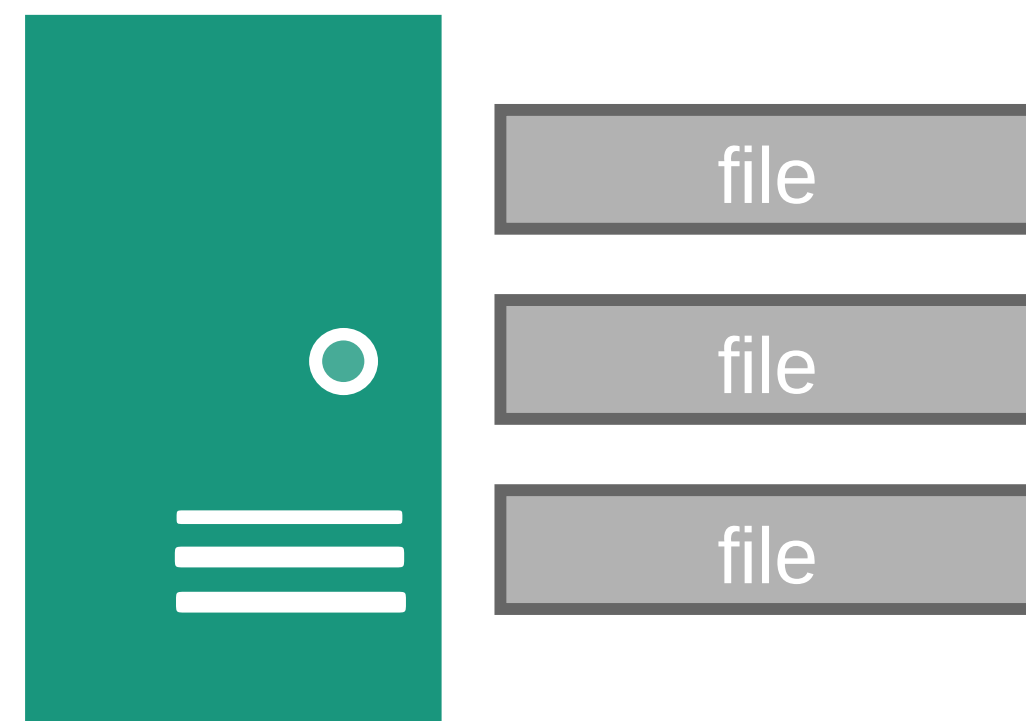
step 3: analyze

TB



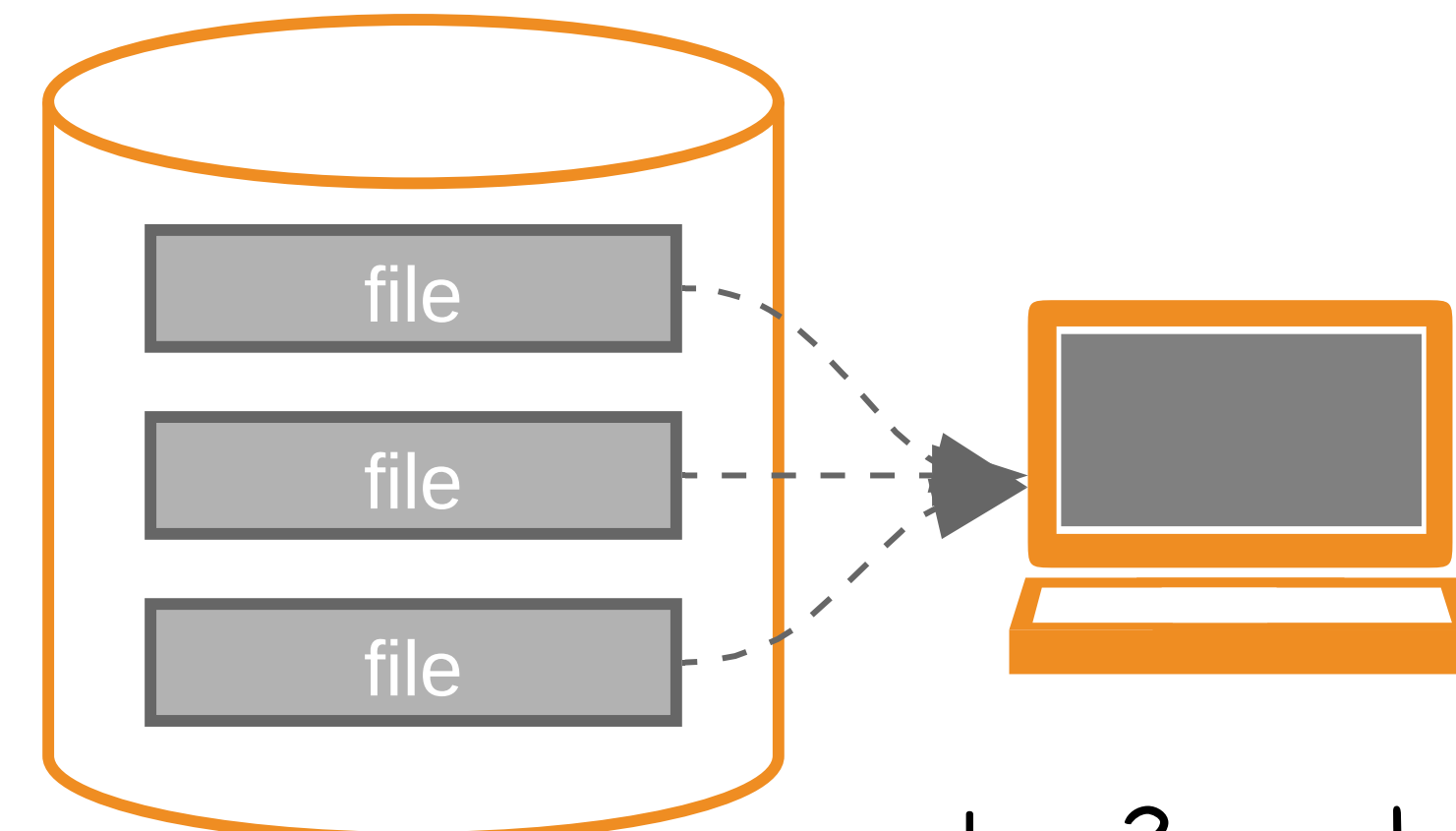
THE "DOWNLOAD" MODEL

step 1: download



files

step 2: clean / organize



local disk

step 3: analyze

PB



NEVER MIND...

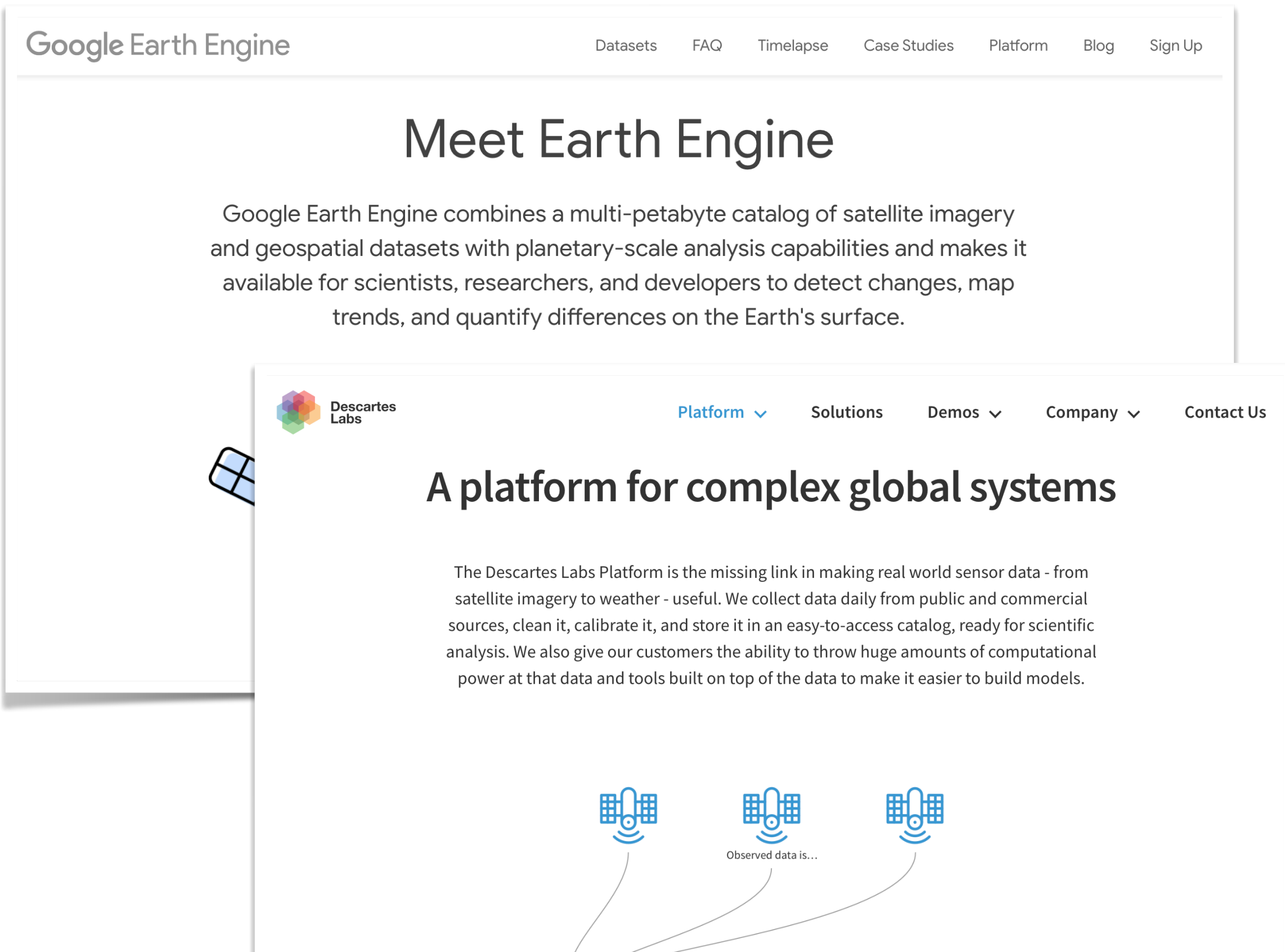
Let's “bring the compute to the data”!

HOW?

CLOUD-BASED ANALYSIS SOLUTIONS

Vertically Integrated
Proprietary Platforms

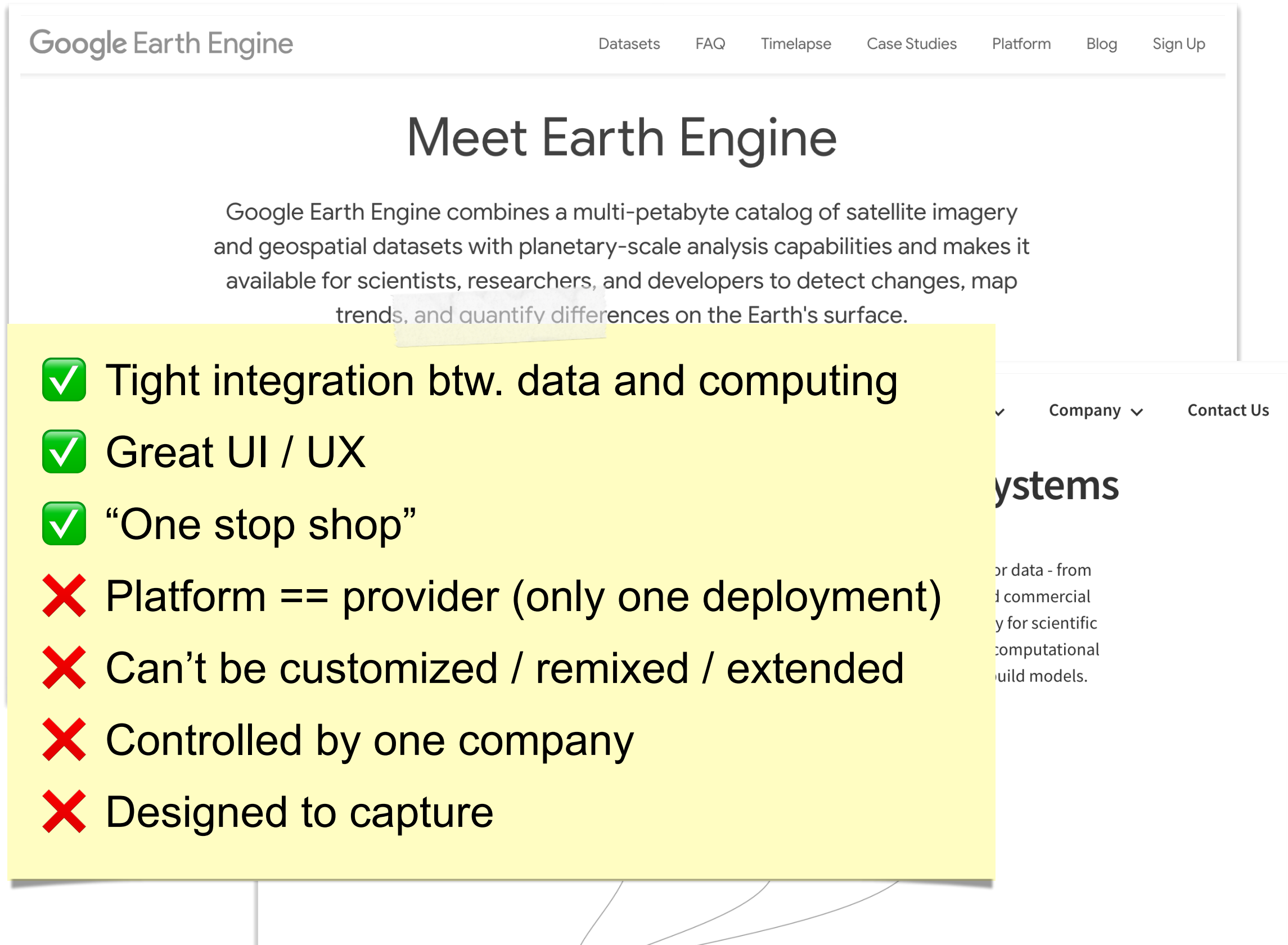
Open Source
Modular Architecture



...etc.

CLOUD-BASED ANALYSIS SOLUTIONS

Vertically Integrated Proprietary Platforms



Google Earth Engine

Datasets FAQ Timelapse Case Studies Platform Blog Sign Up

Meet Earth Engine

Google Earth Engine combines a multi-petabyte catalog of satellite imagery and geospatial datasets with planetary-scale analysis capabilities and makes it available for scientists, researchers, and developers to detect changes, map trends, and quantify differences on the Earth's surface.

- ✓ Tight integration btw. data and computing
- ✓ Great UI / UX
- ✓ “One stop shop”
- ✗ Platform == provider (only one deployment)
- ✗ Can't be customized / remixed / extended
- ✗ Controlled by one company
- ✗ Designed to capture

Open Source Modular Architecture



openEO

PANGEO

- ✓ Free software
- ✓ Community-driven development
- ✓ Deploy anywhere
- ✓ Interoperable
- ✗ Have to manage your own deployment
- ✗ Integration of data / computing can be hard
- ✗ UI / UX is less polished

PILLARS OF CLOUD NATIVE

1. Analysis-Ready, Cloud-Optimized Data

xarray.Dataset

► Dimensions: (latitude: 720, longitude: 1440, nv: 2, time: 8901)

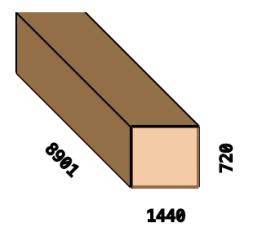
▼ Coordinates:

crs	()	int32	...
lat_bnds	(time, latitude, nv)	float32	dask.array<chunksizes=(...
latitude	(latitude)	float32	-89.875 -89.625 ... 89...
lon_bnds	(longitude, nv)	float32	dask.array<chunksizes=(...
longitude	(longitude)	float32	0.125 0.375 ... 359.625...
nv	(nv)	int32	0 1
time	(time)	datetime64[ns]	1993-01-01 ... 2017-05...
axis :	T		
long_name :	Time		
standard_na...	time		

▼ Data variables:

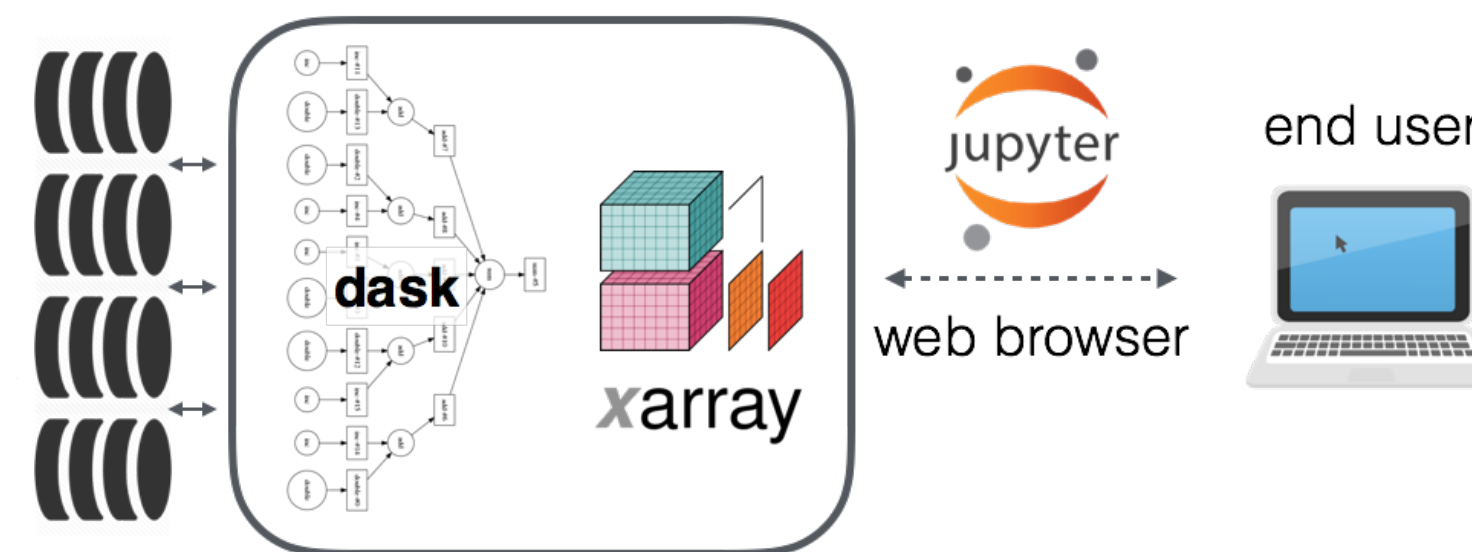
adt	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
-----	-----------------------------	---------	----------------------------

	Array	Chunk
Bytes	73.83 GB	41.47 MB
Shape	(8901, 720, 1440)	(5, 720, 1440)
Count	1782 Tasks	1781 Chunks
Type	float64	numpy.ndarray

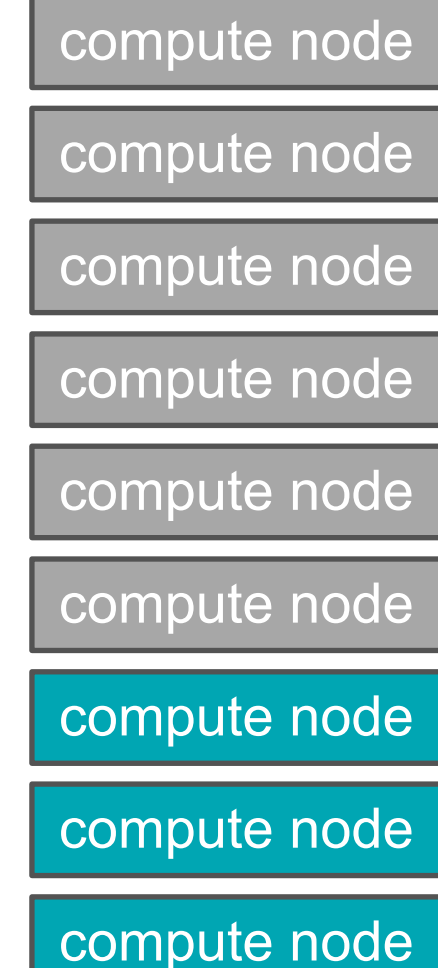


err	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
sla	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
ugos	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
ugosa	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
vgos	(time, latitude, longitude)	float64	dask.array<chunksizes=(...
vgosa	(time, latitude, longitude)	float64	dask.array<chunksizes=(...

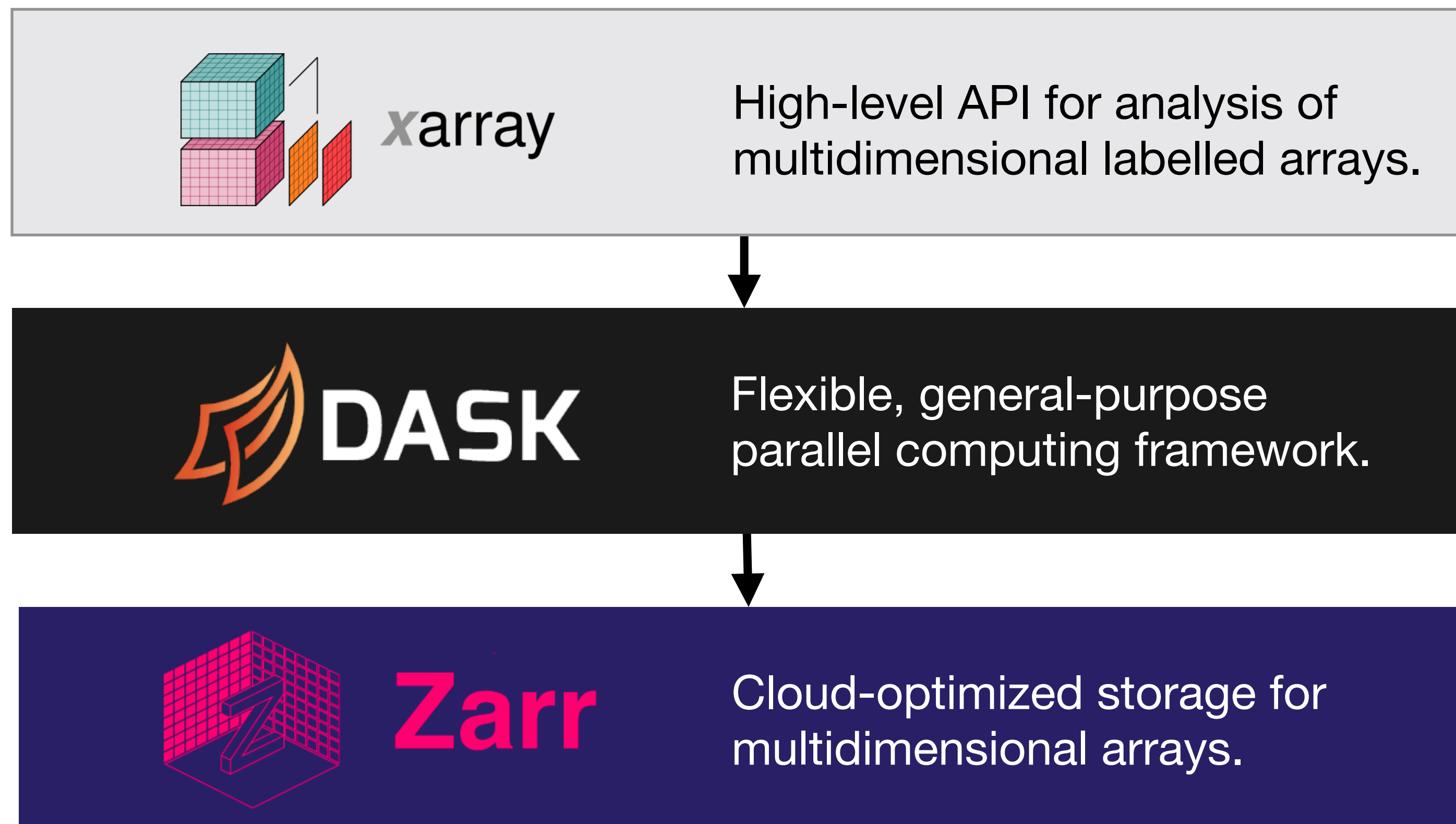
2. Data-Proximate Computing



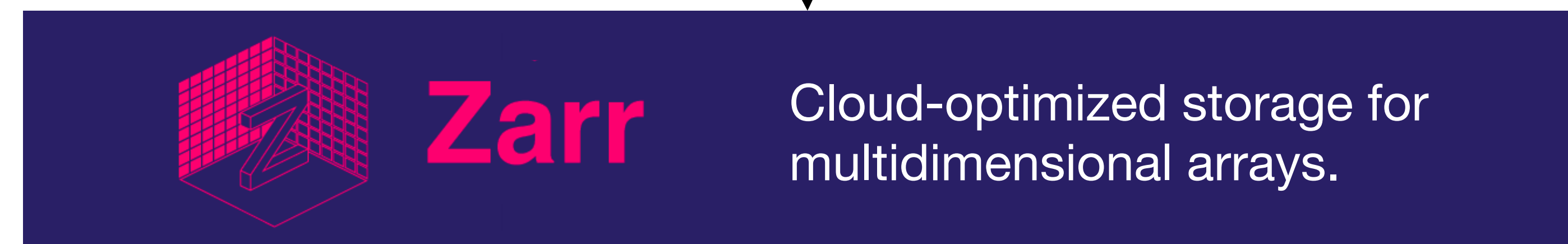
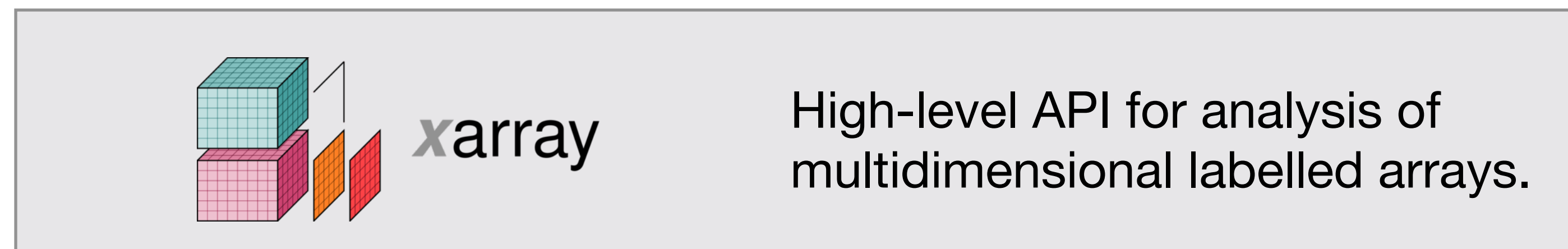
3. Elastic Distributed Processing



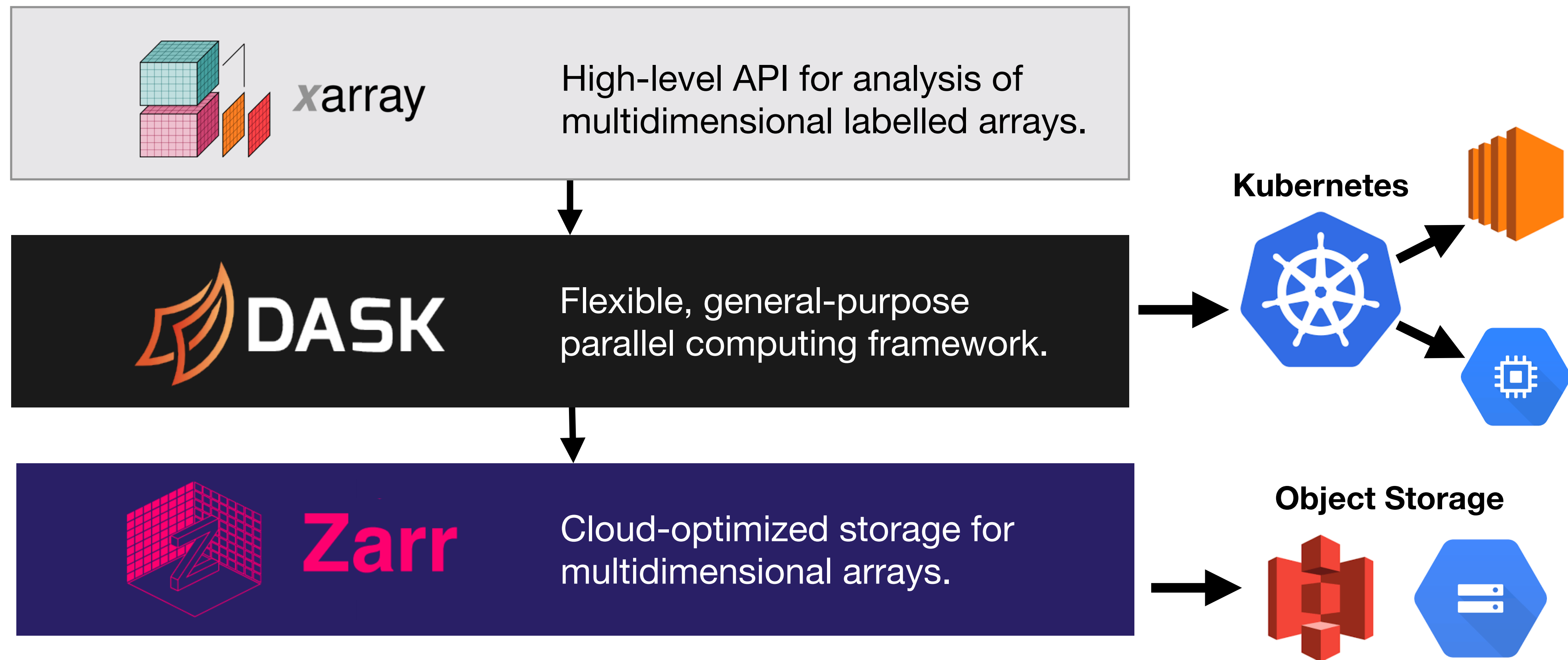
PANGEO CLOUD STACK



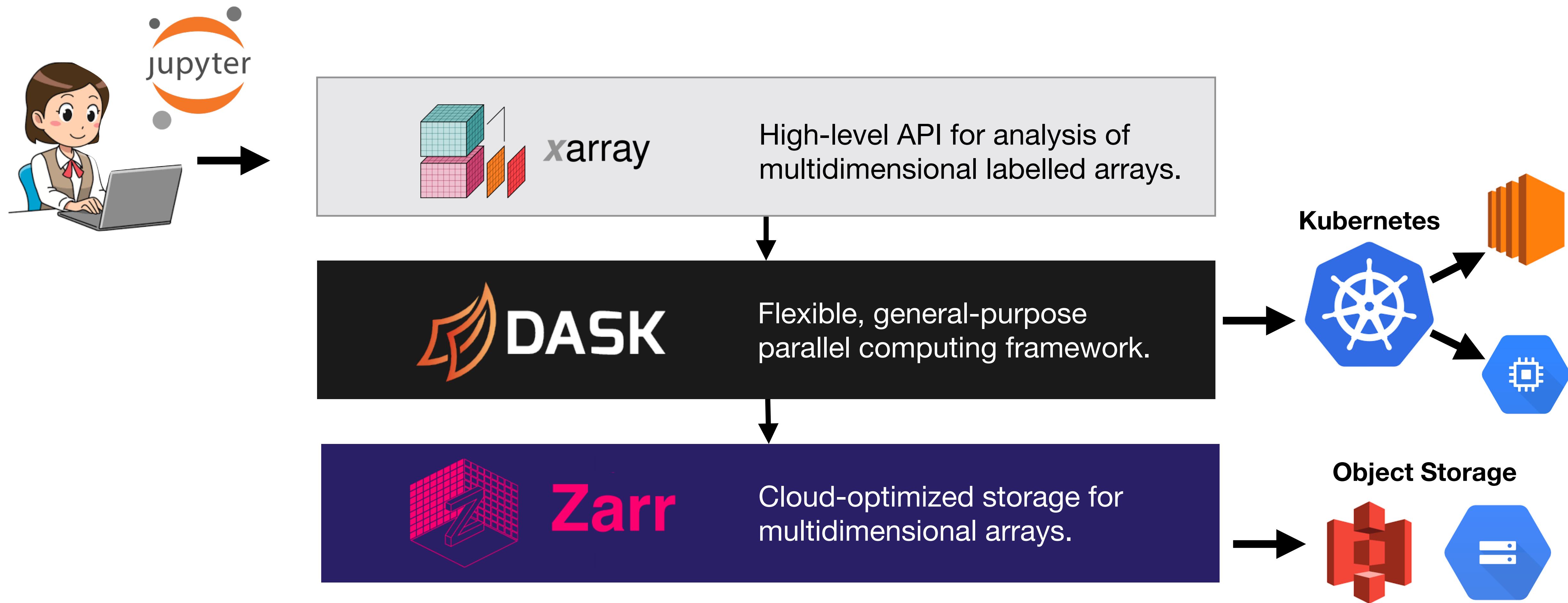
PANGEO CLOUD STACK



PANGEO CLOUD STACK



PANGEO CLOUD STACK

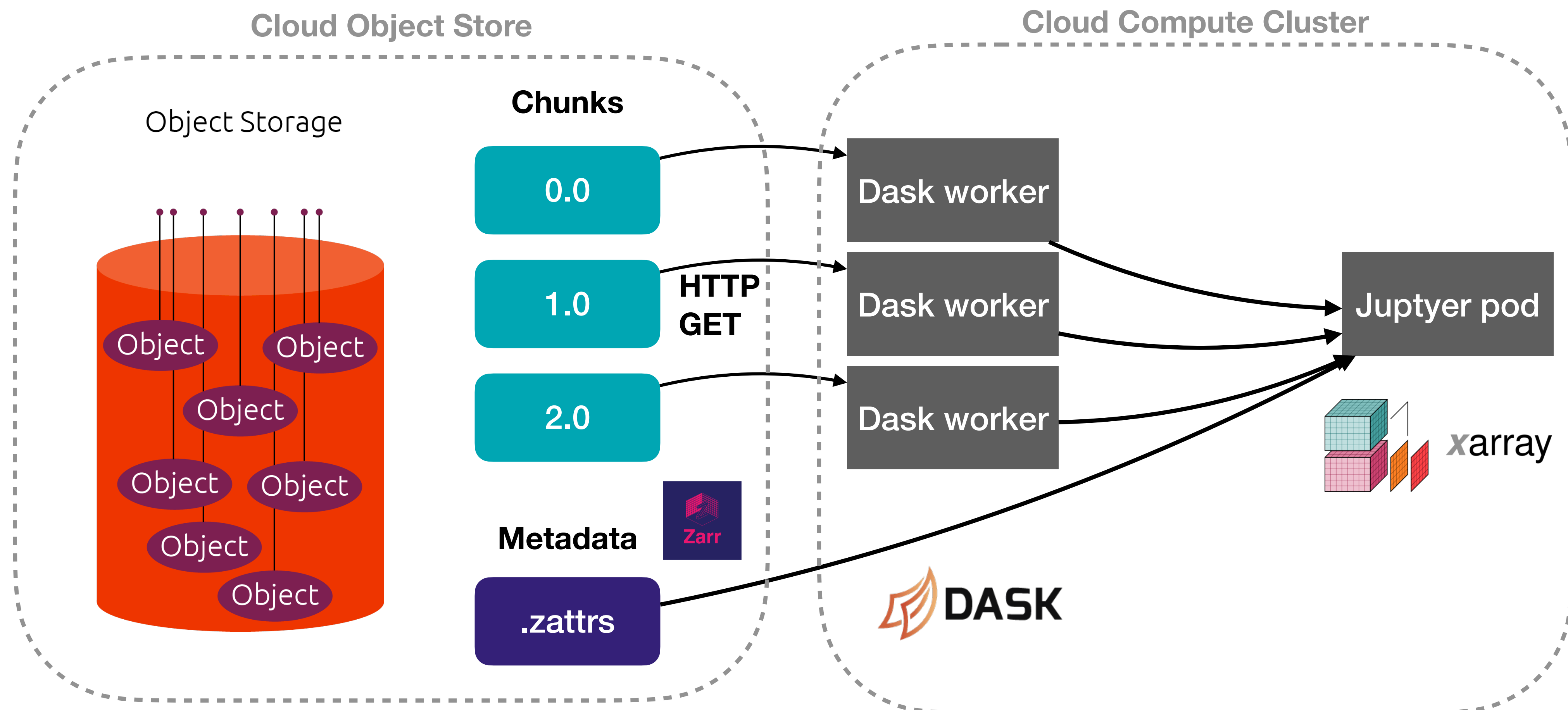


PANGEO CLOUD STACK



THE PANGEO CLOUD STACK

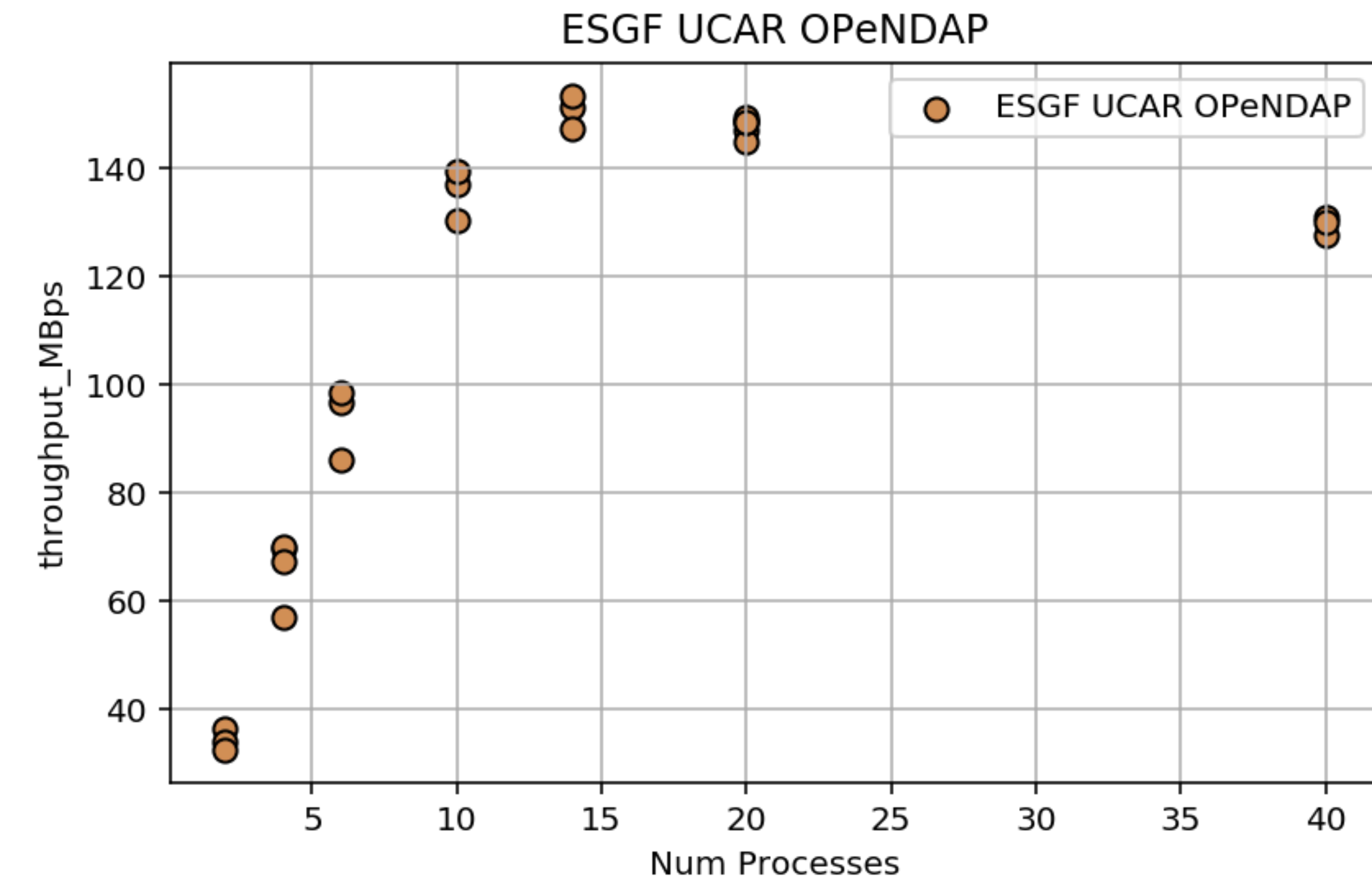
<http://pangeo.io/cloud.html>



CLOUD OPTIMIZED SCALES!

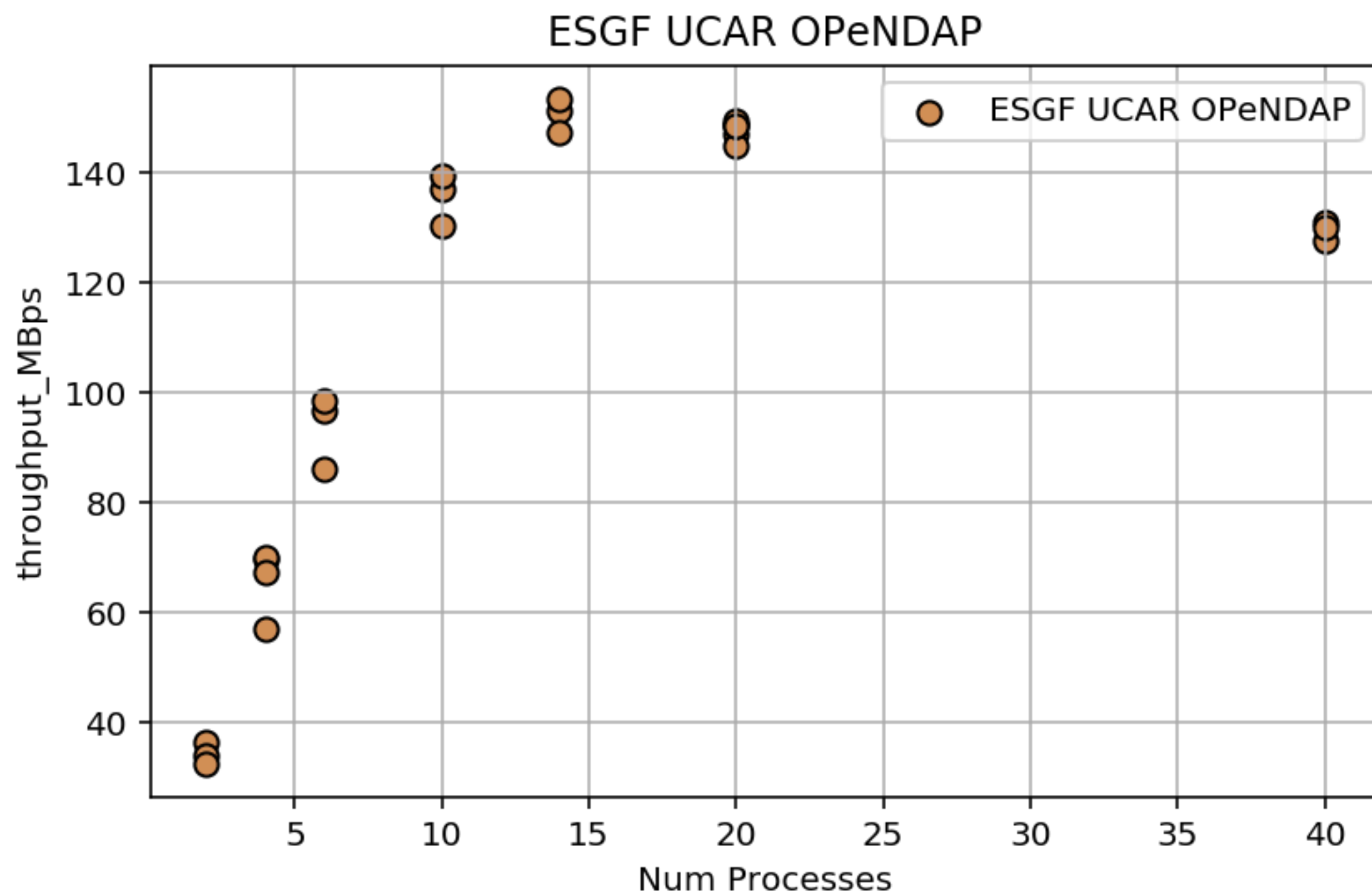
Legacy Server

Xarray + Dask + Zarr

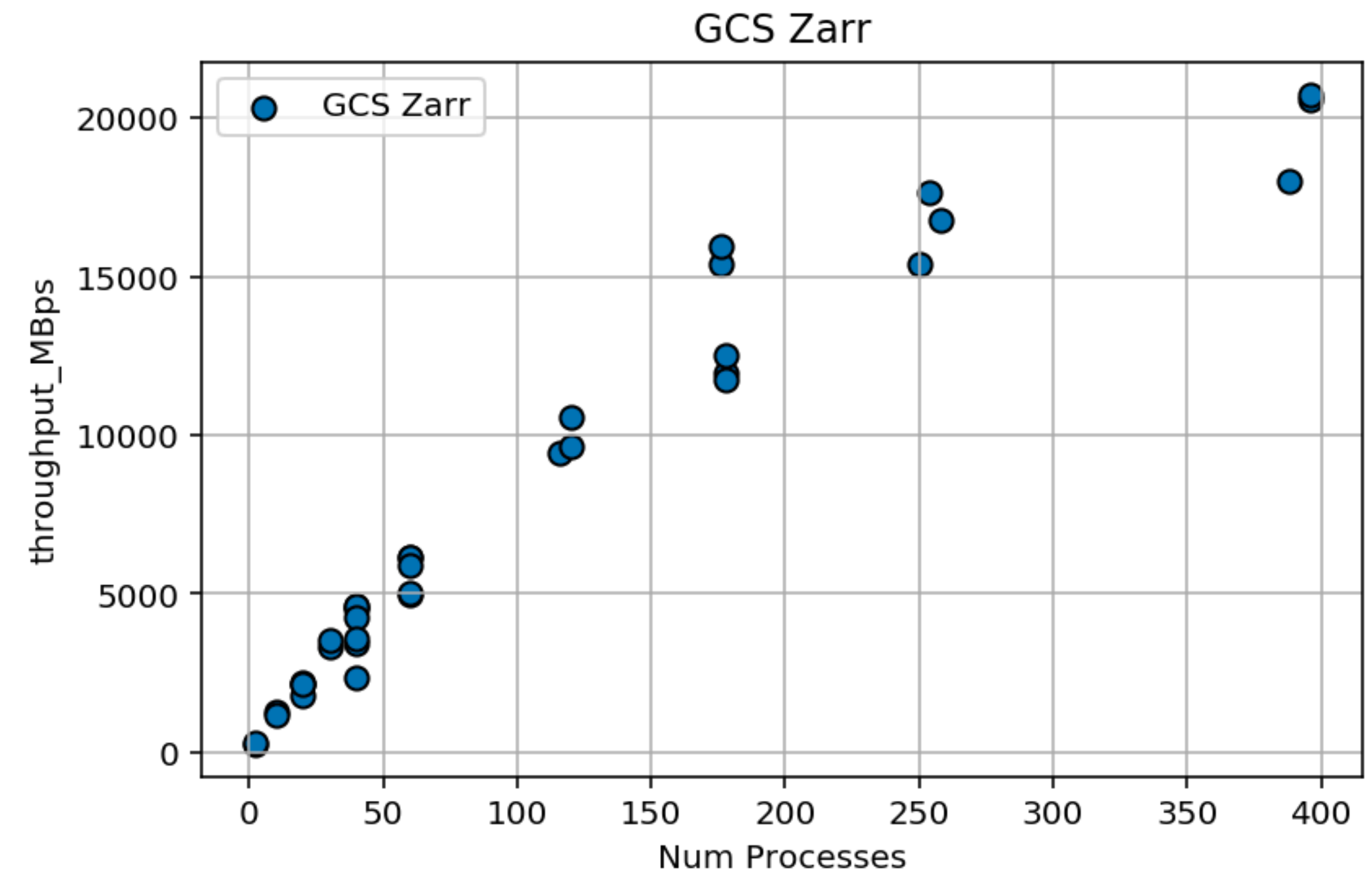


CLOUD OPTIMIZED SCALES!

Legacy Server



Xarray + Dask + Zarr

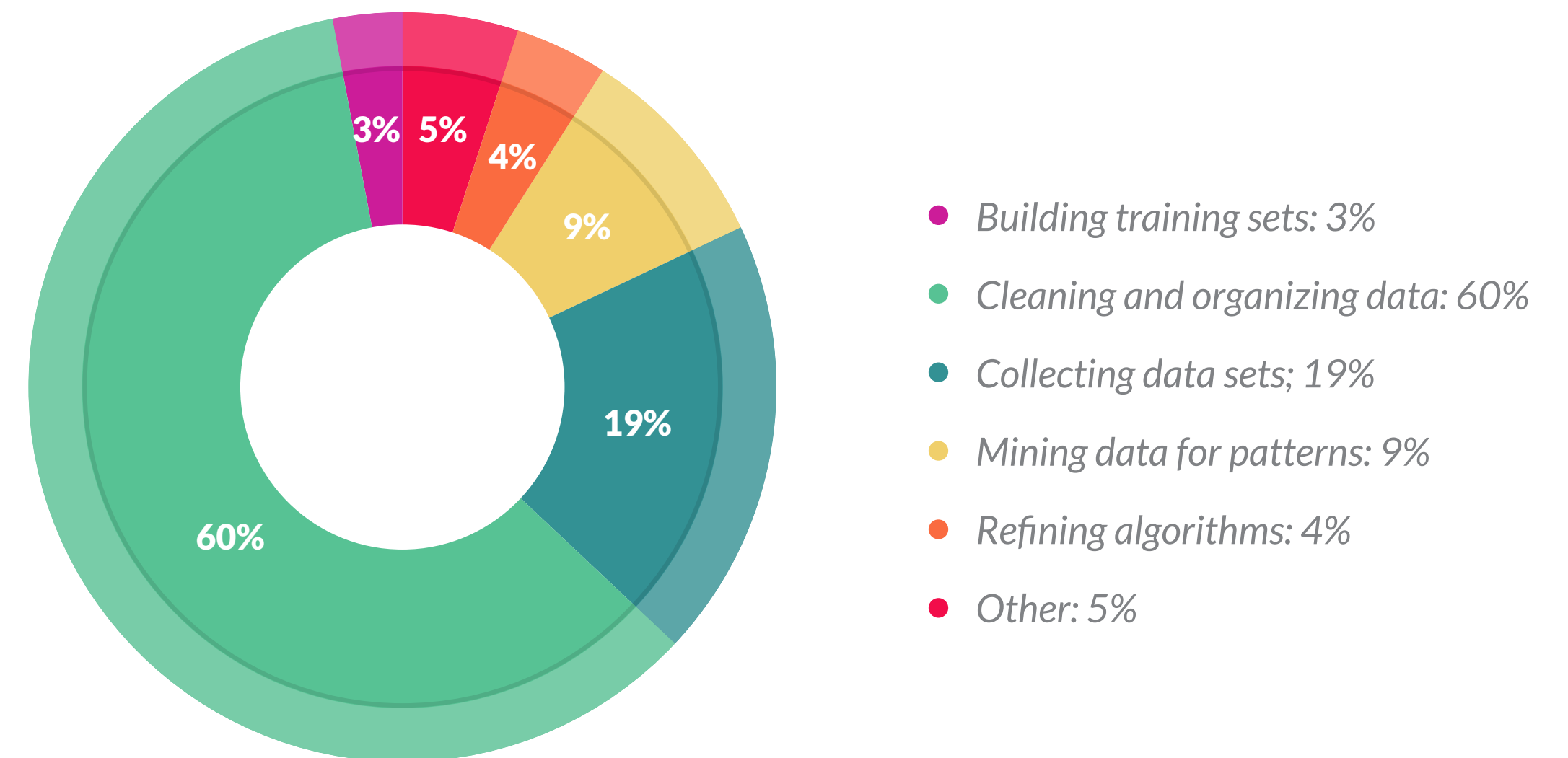


ARCO DATA

Analysis Ready, Cloud Optimized

What is “Analysis Ready”?

- Think in “Datasets” not “data files”
- No need for tedious homogenizing / cleaning steps
- Curated and cataloged

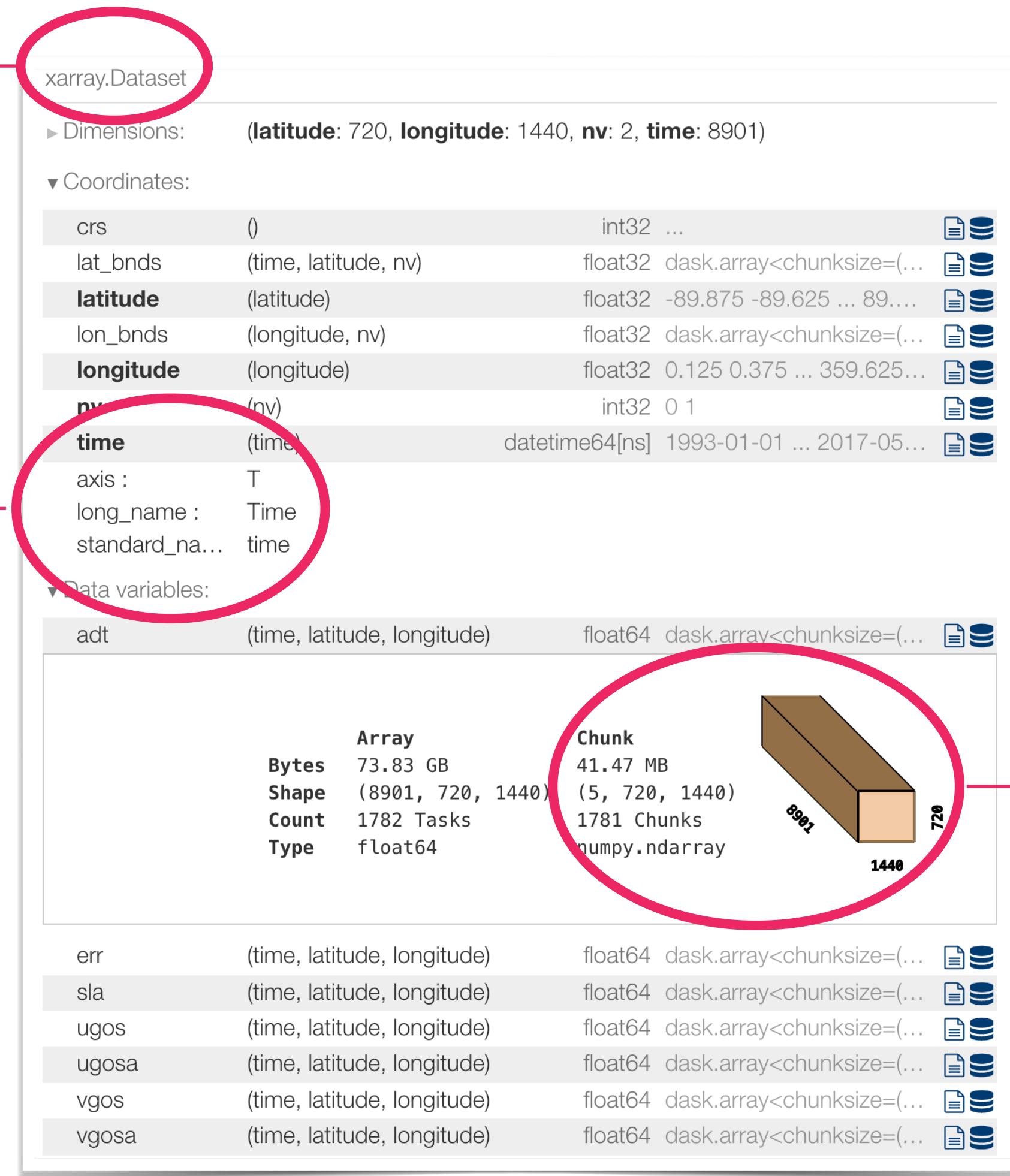


How do data scientists spend their time?
Crowdfunder Data Science Report (2016)

EXAMPLE OF ARCO DATA

Everything in one dataset object

Rich metadata



xarray.Dataset

► Dimensions: (latitude: 720, longitude: 1440, nv: 2, time: 8901)

▼ Coordinates:

crs	()	int32	...
lat_bnds	(time, latitude, nv)	float32	dask.array<chunksize=(...
latitude	(latitude)	float32	-89.875 -89.625 ... 89....
lon_bnds	(longitude, nv)	float32	dask.array<chunksize=(...
longitude	(longitude)	float32	0.125 0.375 ... 359.625...
nv	(nv)	int32	0 1
time	(time)	datetime64[ns]	1993-01-01 ... 2017-05...
axis :	T		
long_name :	Time		
standard_na...	time		

▼ Data variables:

adt	(time, latitude, longitude)	float64	dask.array<chunksize=(...
<div><div><div>Array</div><div>Bytes73.83 GB</div><div>Shape(8901, 720, 1440)</div><div>Count1782 Tasks</div><div>Typefloat64</div></div><div><div>Chunk</div><div>41.47 MB</div><div>(5, 720, 1440)</div><div>1781 Chunks</div><div>numpy.ndarray</div></div></div>			

err (time, latitude, longitude) float64 dask.array<chunksize=(...)

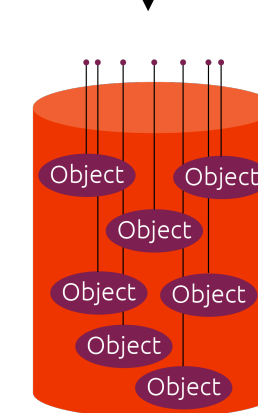
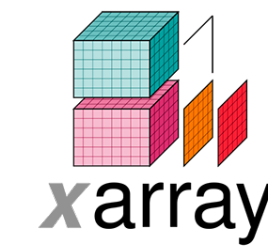
sla (time, latitude, longitude) float64 dask.array<chunksize=(...)

ugos (time, latitude, longitude) float64 dask.array<chunksize=(...)

ugosa (time, latitude, longitude) float64 dask.array<chunksize=(...)

vgos (time, latitude, longitude) float64 dask.array<chunksize=(...)

vgosa (time, latitude, longitude) float64 dask.array<chunksize=(...)



Chunked appropriately for analysis

https://catalog.pangeo.io/browse/master/ocean/sea_surface_height/

ARCO DATA

Analysis **R**eady, **C**loud **O**ptimized

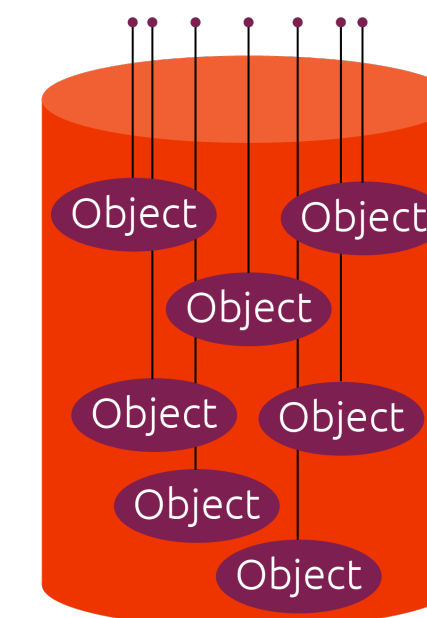
What is “Cloud Optimized”?

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

 pandas



 Parquet



Amazon S3



Azure Blob Storage

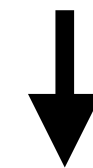


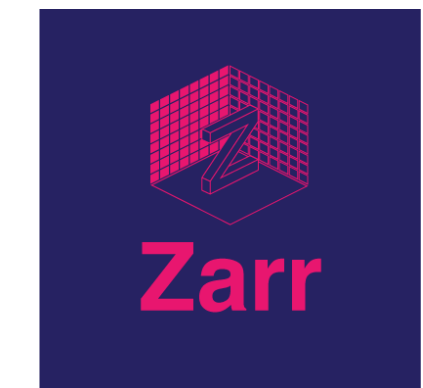
Google Cloud Storage

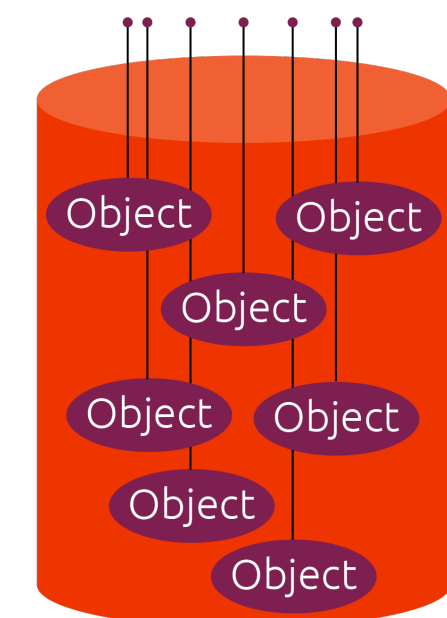


ceph

 xarray



 Zarr



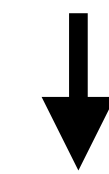
ARCO DATA

Analysis Ready, Cloud Optimized

What is “Cloud Optimized”?

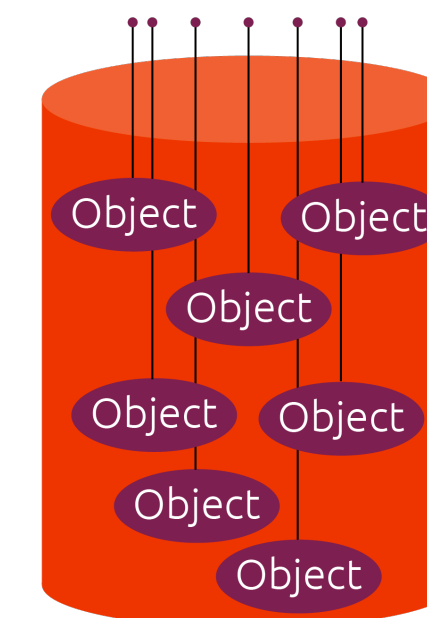
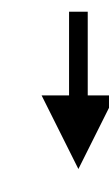
- Compatible with object storage (access via HTTP)
- Supports lazy access and intelligent subsetting
- Integrates with high-level analysis libraries and distributed frameworks

 **pandas**

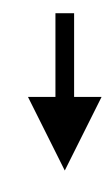


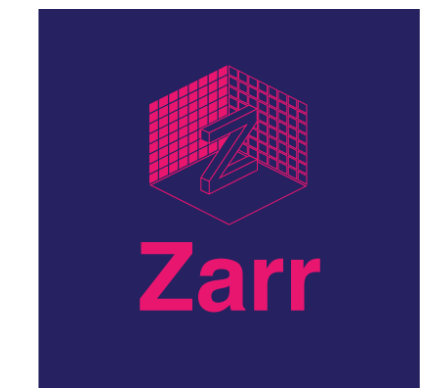
 **DASK**

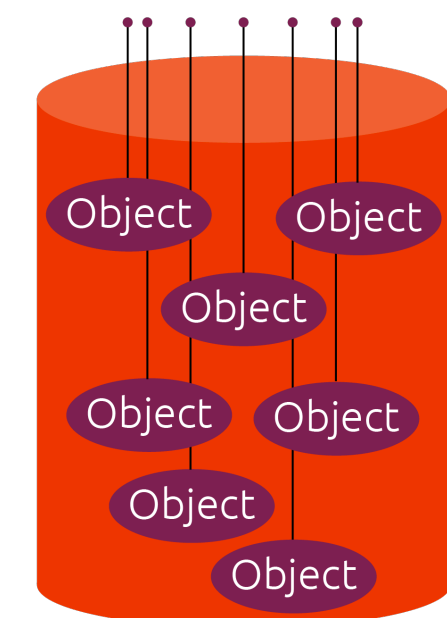
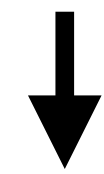
 **Parquet**



 **xarray**




 **Zarr**



PANGEO CLOUD DATA CATALOG

[CATALOG.PANGEO.IO](https://catalog.pangeo.io)

 Blog Forum

PANGEO CATALOG

master

MASTER

Pangeo Master Data Catalog

<https://raw.githubusercontent.com/pangeo-data/pangeo-datastore/master/intake-catalogs/master.yaml>

Child Catalogs

ocean

Pangeo Oceanography Dataset Catalog

atmosphere

Pangeo Atmospheric Science Dataset Catalog

climate

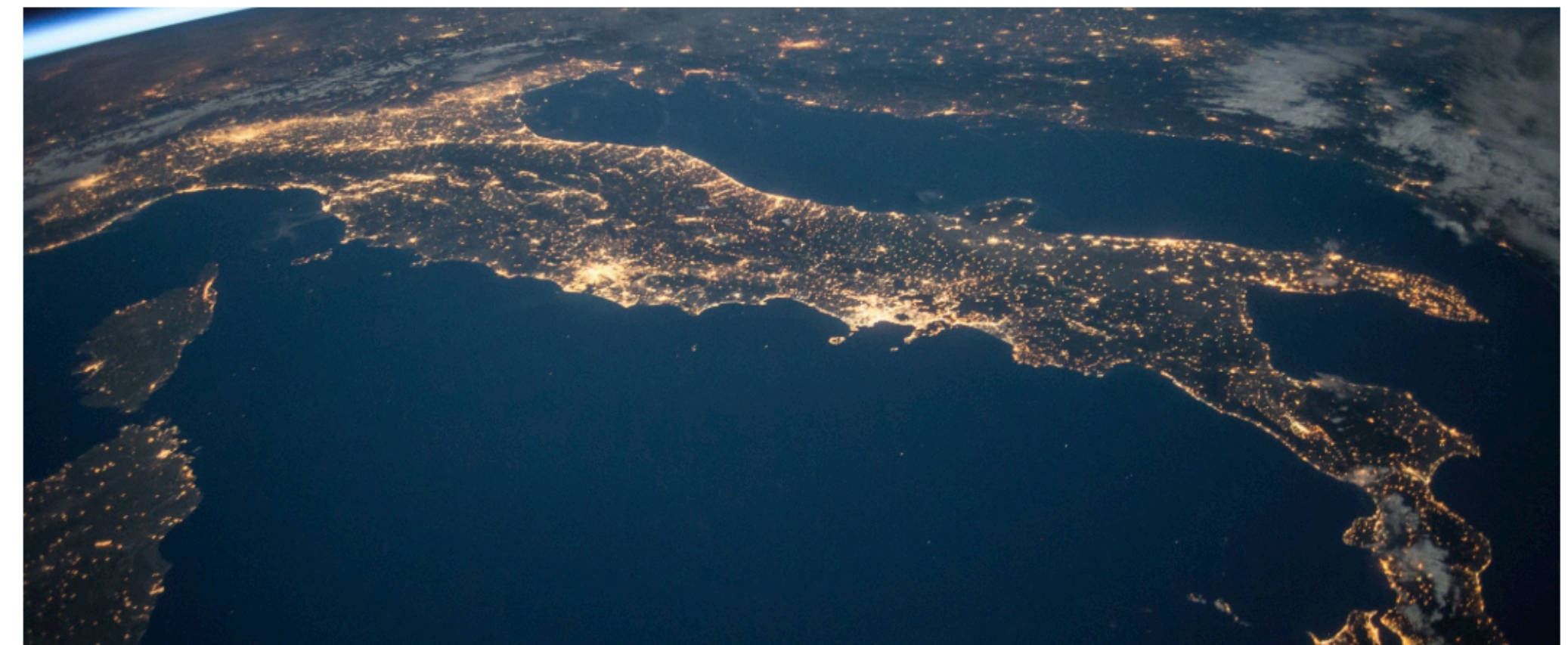
Pangeo Climate Dataset Catalog. Include model ensembles such as CMIP6 and LENS.

hydro

Pangeo Hydrology Dataset Catalog

DATA ANALYTICS

New climate model data now in Google Public Datasets



Shane Glass
Program Manager, Google
Cloud Public Dataset Program

December 9, 2019

Exploring [public datasets](#) is an important aspect of modern data analytics, and all this gathered data can help us understand our world. At Google Cloud, we maintain a collection of public datasets, and we're pleased to collaborate with the [Lamont-Doherty Earth Observatory](#) (LDEO) of Columbia University and the Pangeo Project [to host the latest climate simulation data in the cloud.](#)

PROBLEM:

Making ARCO Data is Hard!



To produce useful ARCO data, you must have:



Data Scientist

PROBLEM:

Making ARCO Data is Hard!



To produce useful ARCO data, you must have:

Domain Expertise:
How to find, clean, and
homogenize data



Data Scientist

PROBLEM:

Making ARCO Data is Hard!



To produce useful ARCO data, you must have:

Domain Expertise:
How to find, clean, and
homogenize data

Tech Knowledge:
How to efficiently produce
cloud-optimized formats



Data Scientist

PROBLEM:

Making ARCO Data is Hard!



To produce useful ARCO data, you must have:

Domain Expertise:
How to find, clean, and
homogenize data

Tech Knowledge:
How to efficiently produce
cloud-optimized formats

Compute Resources:
A place where to stage and
upload the ARCO data



Data Scientist

PROBLEM:

Making ARCO Data is Hard!



To produce useful ARCO data, you must have:

Domain Expertise:
How to find, clean, and
homogenize data

Tech Knowledge:
How to efficiently produce
cloud-optimized formats

Compute Resources:
A place where to stage and
upload the ARCO data

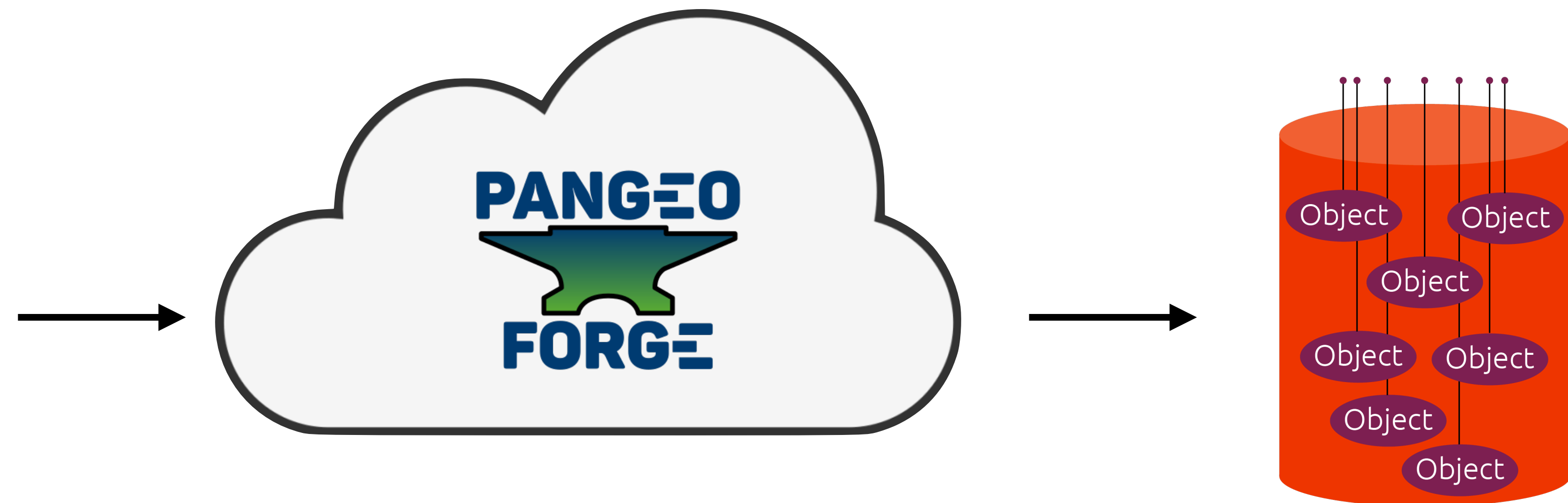
Communication Skills:
To explain to others how to
use the data



Data Scientist

PANGEO FORGE

Let's democratize the production of ARCO data!



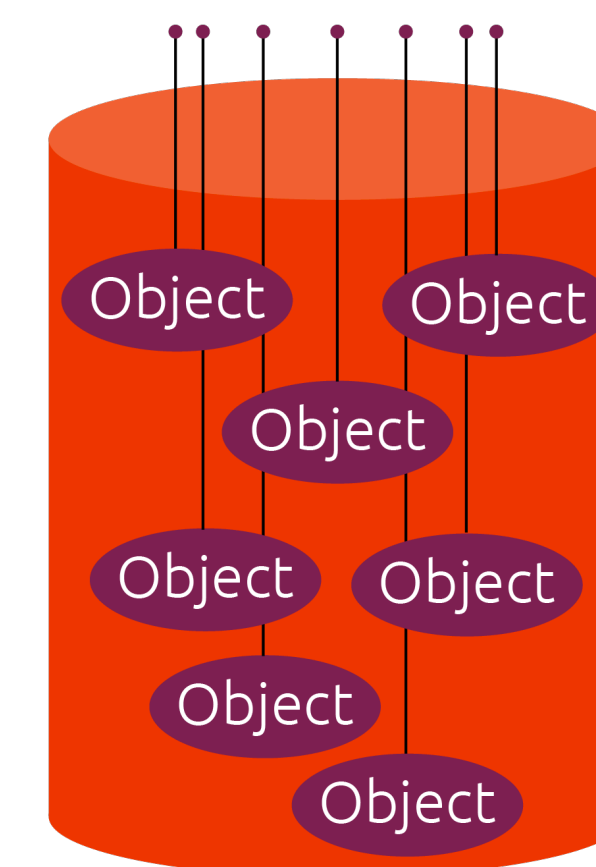
Data Scientist

PANGEO FORGE

Let's democratize the production of ARCO data!



Domain Expertise:
How to find, clean, and
homogenize data



Data Scientist

PANGEO FORGE RECIPES

Recipe defines:

- How to get the *inputs*
(e.g. 14000 daily netCDF files)
- How to combine the inputs
- Target format (e.g. Zarr)

<https://pangeo-forge.readthedocs.io/>

```
input_url_pattern = (  
    "https://www.ncei.noaa.gov/data/sea-surface-temperature-optimum-interpola  
    "/v2.1/access/avhrr/{yyyymm}/oisst-avhrr-v02r01.{yyyymmdd}.nc"  
)
```

```
recipe = NetCDFtoZarrSequentialRecipe(  
    input_urls=input_urls,  
    sequence_dim="time",  
    inputs_per_chunk=20  
)  
recipe
```

PANGEO FORGE BAKERIES

Recipes

<https://github.com/pangeo-forge/>

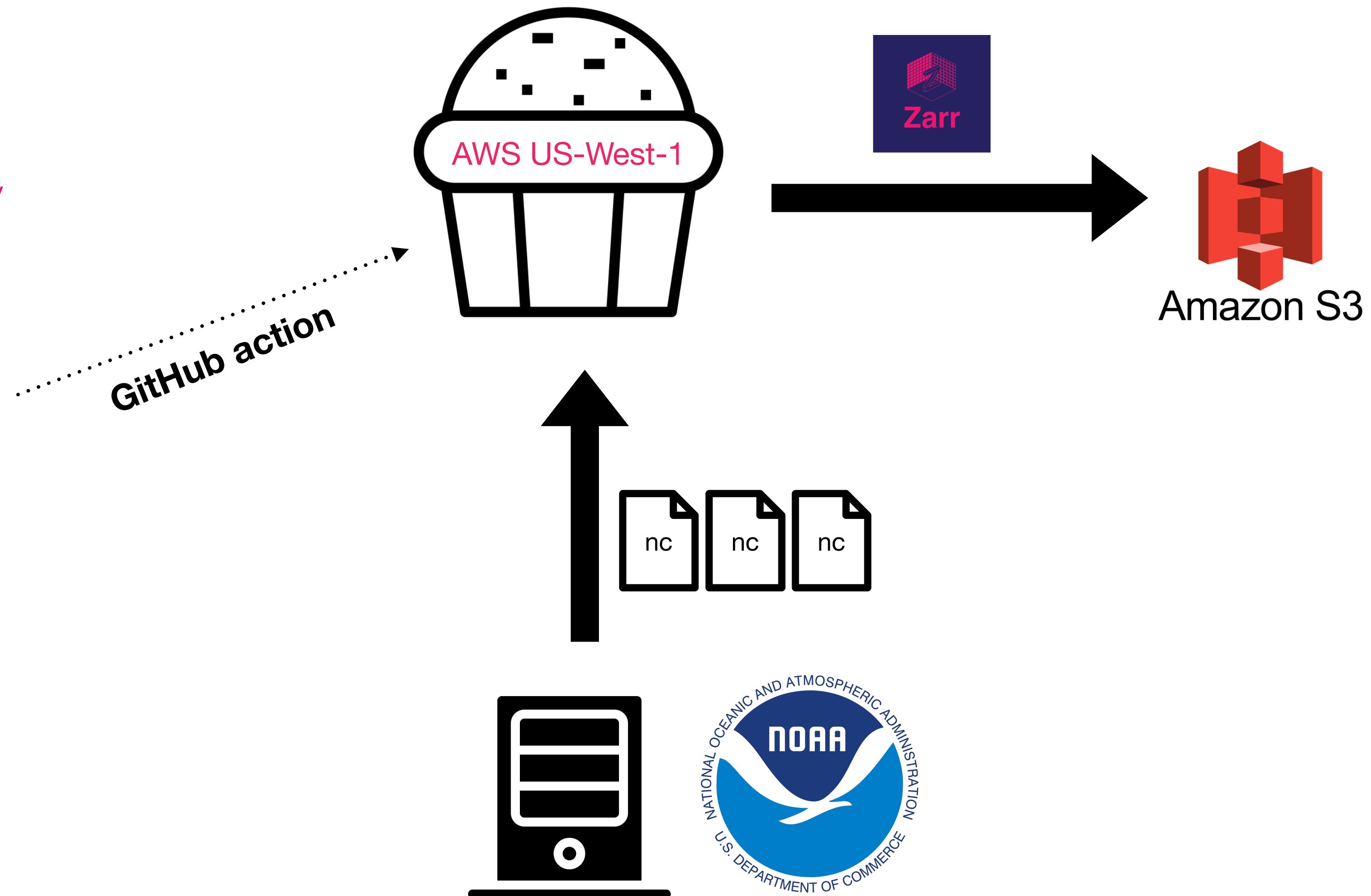
terraclimate-feedstock

A pangeo-smithy repository for the terraclimate dataset.

● Python Apache-2.0 3 2 1 3 Updated on Jan 1

noaa-oisst-avhrr-feedstock

● Python Apache-2.0 2 1 0 4 Updated on Jan 1



PANGEO FORGE BAKERIES

Recipes

<https://github.com/pangeo-forge/>

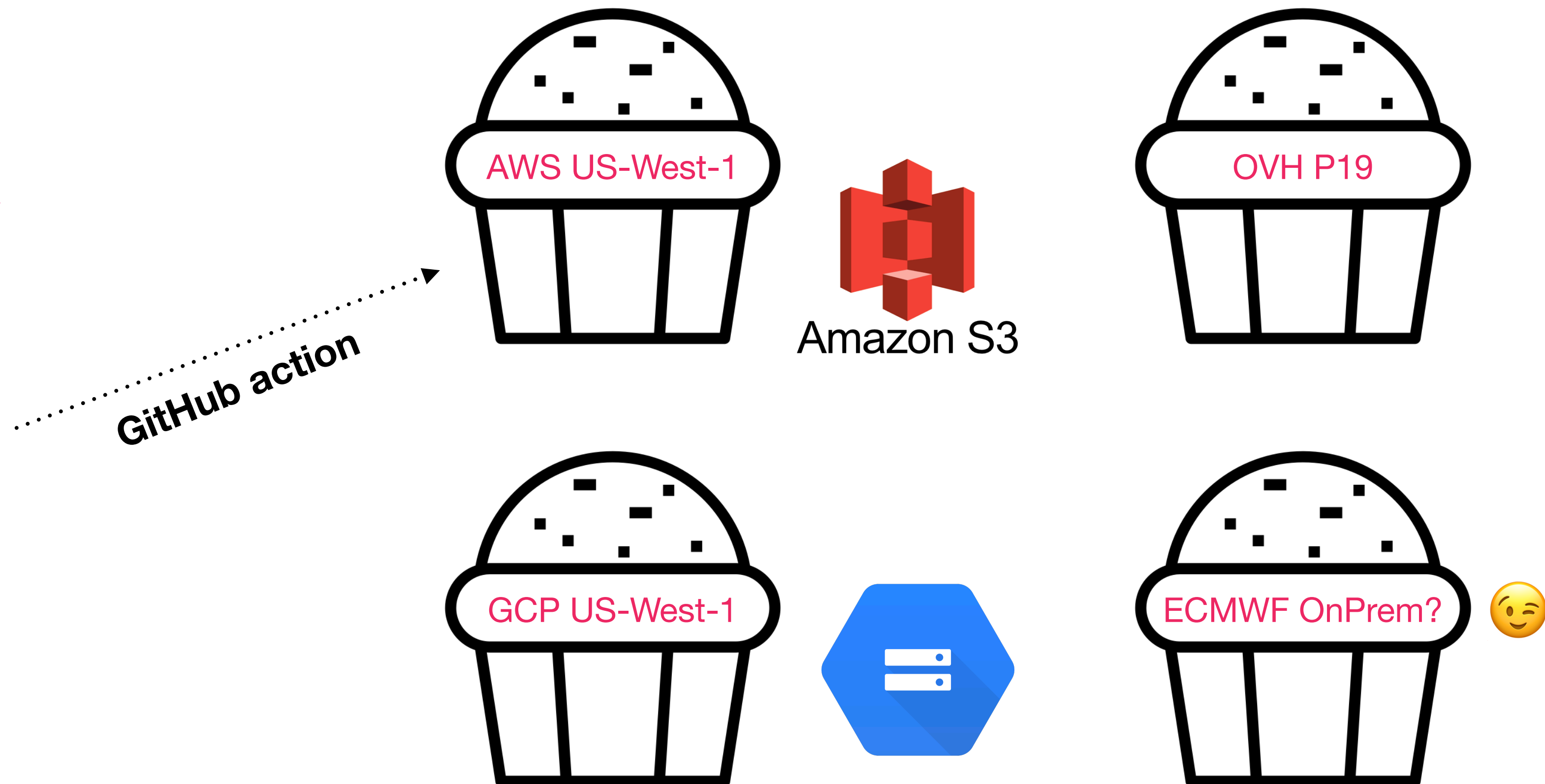
terraclimate-feedstock

A pangeo-smithy repository for the terraclimate dataset.

● Python 📄 Apache-2.0 🍴 3 ☆ 2 ⌚ 1 🔗 3 Updated on Jan 1

noaa-oisst-avhrr-feedstock

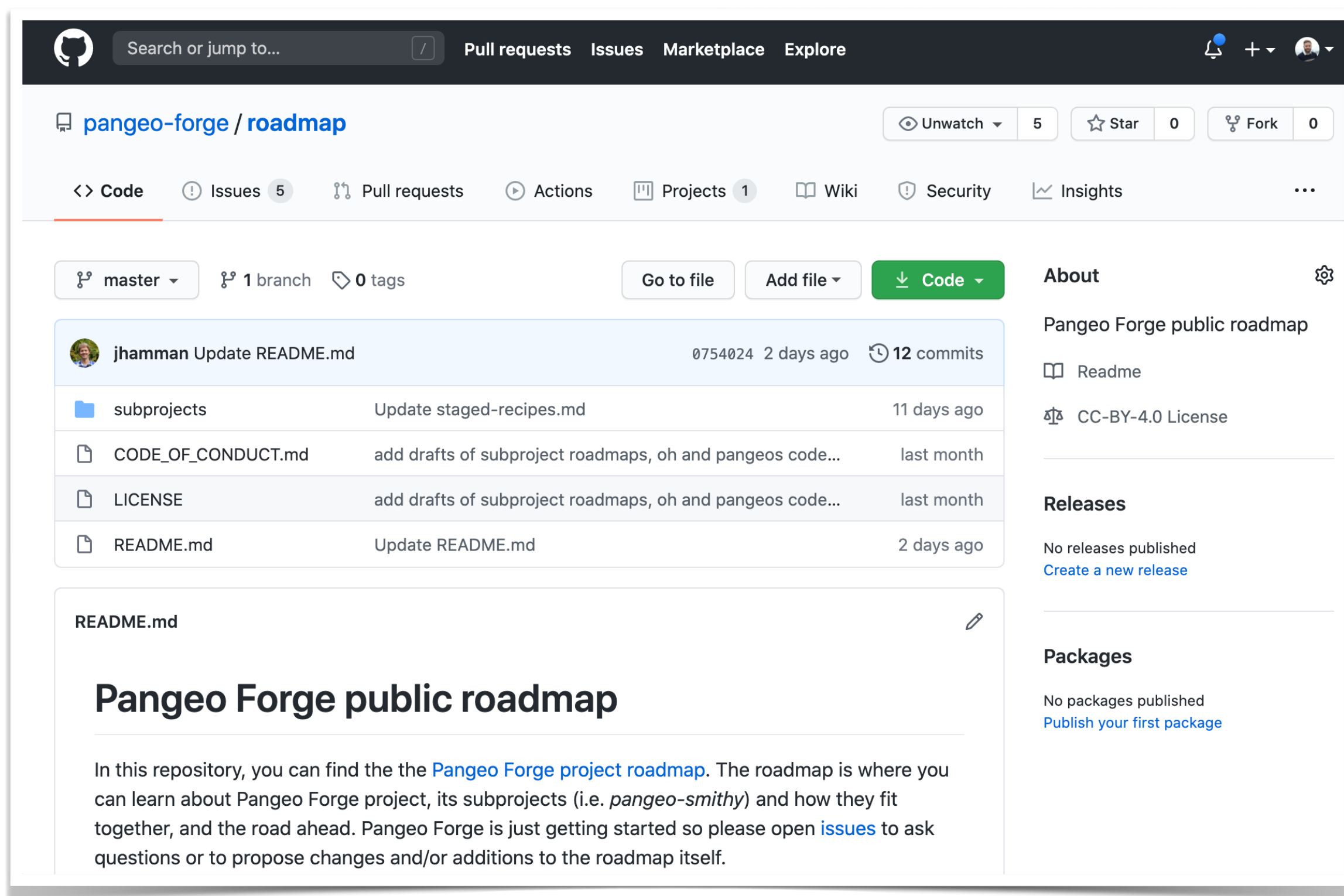
● Python 📄 Apache-2.0 🍴 2 ☆ 1 ⌚ 0 🔗 4 Updated on Jan 1



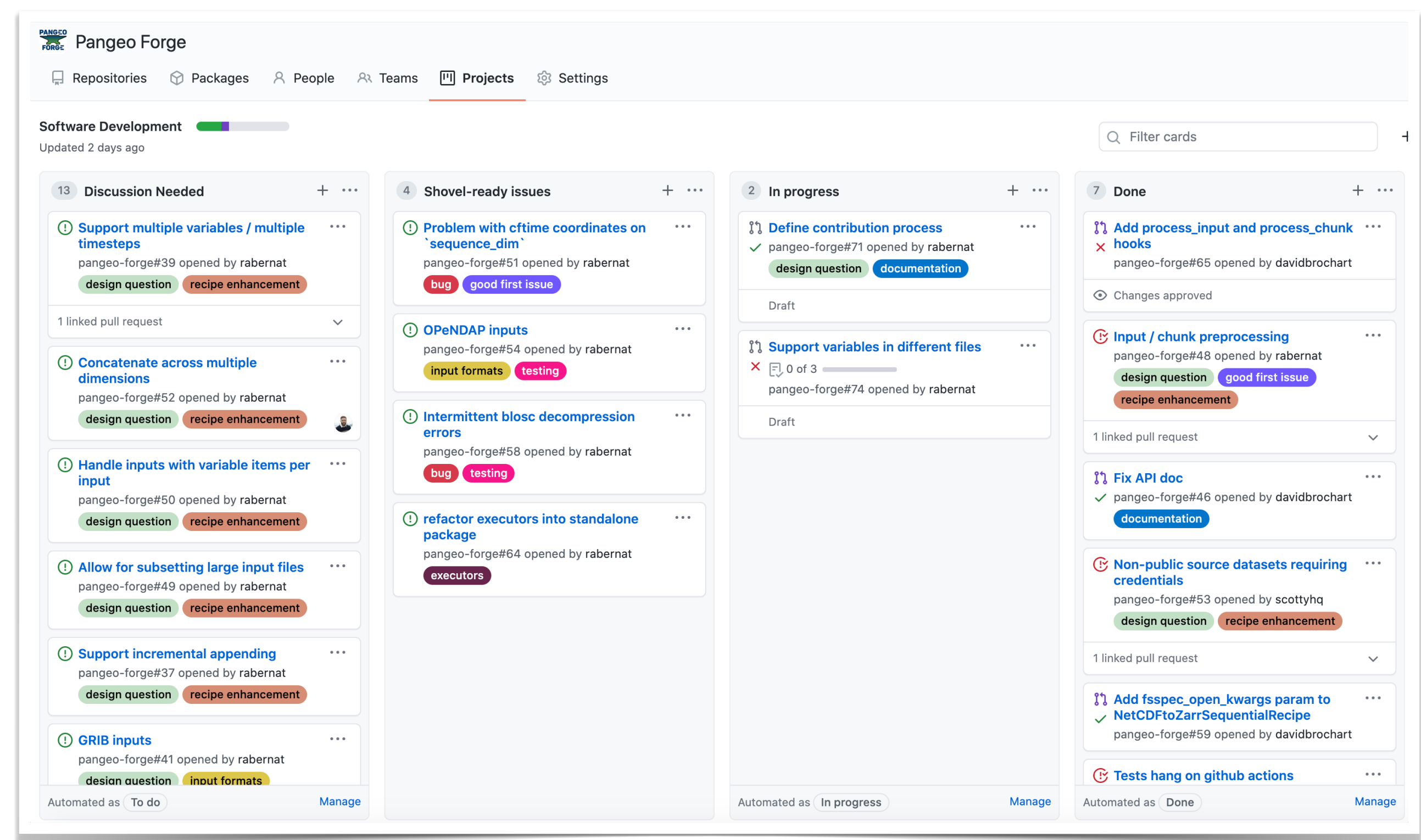
Bakeries are “Franchisable”
Will operate in a federation

PANGEO FORGE DEVELOPMENT

This is a 100% open project!



The screenshot shows the GitHub repository for Pangeo Forge, specifically the 'roadmap' page. The repository is owned by 'pangeo-forge' and has 5 issues, 1 pull request, and 0 stars. The 'Code' tab is selected, showing the 'master' branch with 1 branch and 0 tags. The 'About' section indicates the repository is the 'Pangeo Forge public roadmap' with a CC-BY-4.0 license. The 'Releases' section shows no releases published. The 'Packages' section shows no packages published. The 'README.md' file is displayed, containing the text: 'Pangeo Forge public roadmap. In this repository, you can find the the Pangeo Forge project roadmap. The roadmap is where you can learn about Pangeo Forge project, its subprojects (i.e. pangeo-smithy) and how they fit together, and the road ahead. Pangeo Forge is just getting started so please open issues to ask questions or to propose changes and/or additions to the roadmap itself.'



The screenshot shows the Pangeo Forge project dashboard. The dashboard is titled 'Pangeo Forge' and has tabs for 'Repositories', 'Packages', 'People', 'Teams', 'Projects', and 'Settings'. The 'Projects' tab is selected, showing a 'Software Development' project. The project is updated 2 days ago. The dashboard displays a Kanban board with columns for 'Discussion Needed', 'Shovel-ready issues', 'In progress', and 'Done'. Each column contains a list of issues with their status, labels, and assignees. The 'Discussion Needed' column has 13 issues, 'Shovel-ready issues' has 4, 'In progress' has 2, and 'Done' has 7. The issues are categorized by labels such as 'design question', 'recipe enhancement', 'bug', 'good first issue', 'input formats', 'testing', and 'documentation'.

<https://github.com/pangeo-forge/roadmap>

SUMMARY

- *Problem:* scientific data is large and complex 🤯
- *Solution:* data-proximate computing in the cloud 😎
- *Problem:* analysis-ready, cloud-optimized data is scarce 🤔
- *Solution:* **Pangeo Forge** 💪

We need your help!

<https://github.com/pangeo-forge/roadmap>

LEARN MORE

- “Cloud-Native Repositories for Big Scientific Data”
Abernathey et al. 2021. *Computing in Science and Engineering*
<https://doi.org/10.22541/au.160443768.88917719/v2>
- “Opening new horizons: How to migrate the Copernicus Global Land Service to a Cloud environment”. Abernathey et al., 2021. *Publications Office of the European Union*. <http://dx.doi.org/10.2760/668980>
- “Closed Platforms vs. Open Architectures for Cloud-Native Earth System Analytics”. Abernathey & Hamman, 2020.
<https://medium.com/pangeo/closed-platforms-vs-open-architectures-for-cloud-native-earth-system-analytics-1ad88708ebb6>