PANGEO

A COMMUNITY-DRIVEN EFFORT FOR
BIG DATA GEOSCIENCE

WHAT WOULD YOU LIKE TO HAVE AND WHY?

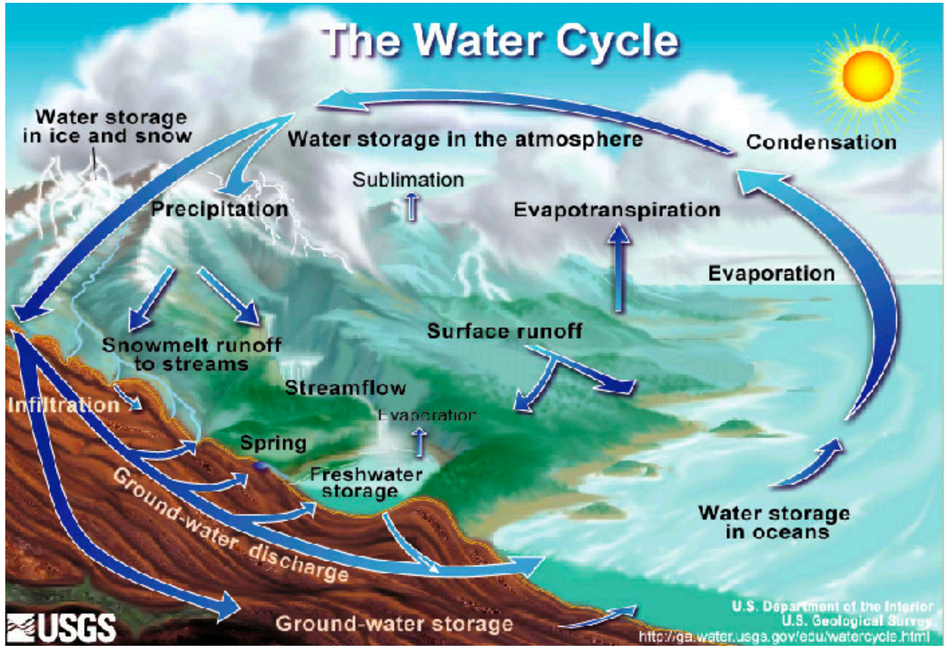
Pangeo's vision for scientific computing in the big-data era

PANGEO'S WEBSITE

pangeo-data.org

HELLO!

- Who am I?
  - Joe Hamman, Ph.D., P.E.
  - I am a scientist at the National Center for Atmospheric Research
  - I study the impacts of climate change on the water cycle.
  - I contribute to open-source projects like Pangeo, Xarray, Dask, and Jupyter



Github: @jhamman
Twitter: @HammanHydro
Web: joehamman.com

EARTHCUBE AWARD TEAM

EARTHCUBE

Google Cloud Platform

Lamont-Doherty Earth Observatory
COLUMBIA UNIVERSITY | EARTH INSTITUTE

Ryan Abernathey, Chiara Lepore, Michael Tippet, Naomi Henderson, Richard Seager

Kevin Paul, Joe Hamman, Ryan May, Davide Del Vento

Matthew Rocklin

OTHER CONTRIBUTORS

Jacob Tomlinson, Niall Roberts, Alberto Arribas

Developing and operating Pangeo environment to support analysis of UK Met office products

Rich Signell

Deploying Pangeo on AWS to support analysis of coastal ocean modeling

Justin Simcock

Operating Pangeo in the cloud to support Climate Impact Lab research and analysis

Supporting Pangeo via SWOT mission and funded ACCESS award to UW / NCAR

Yuvi Panda, Chris Holdgraf

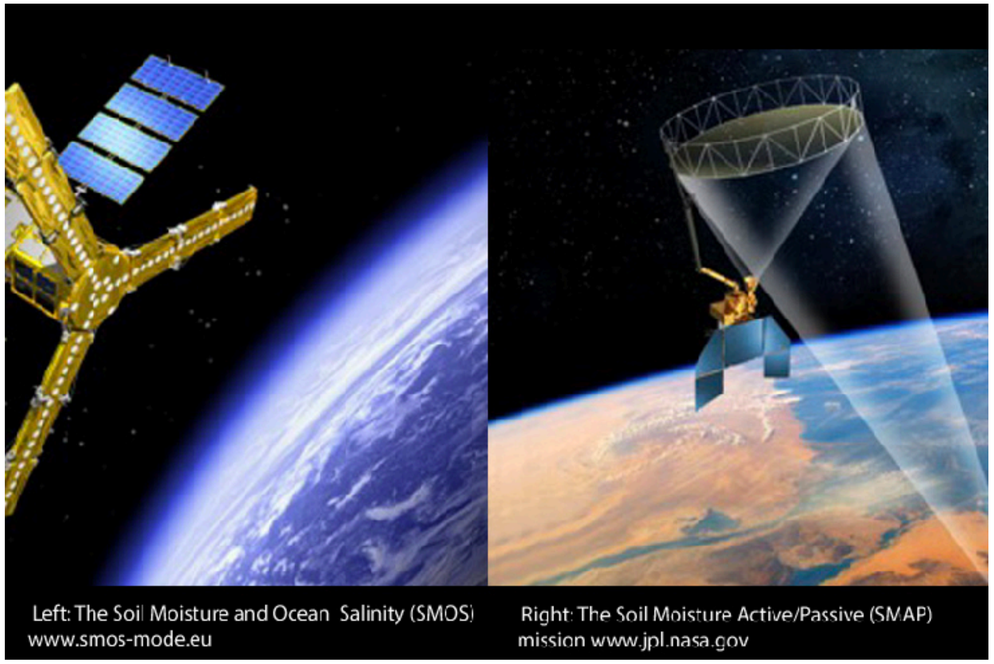
Spending lots of time helping us make things work on the cloud

PANGEO

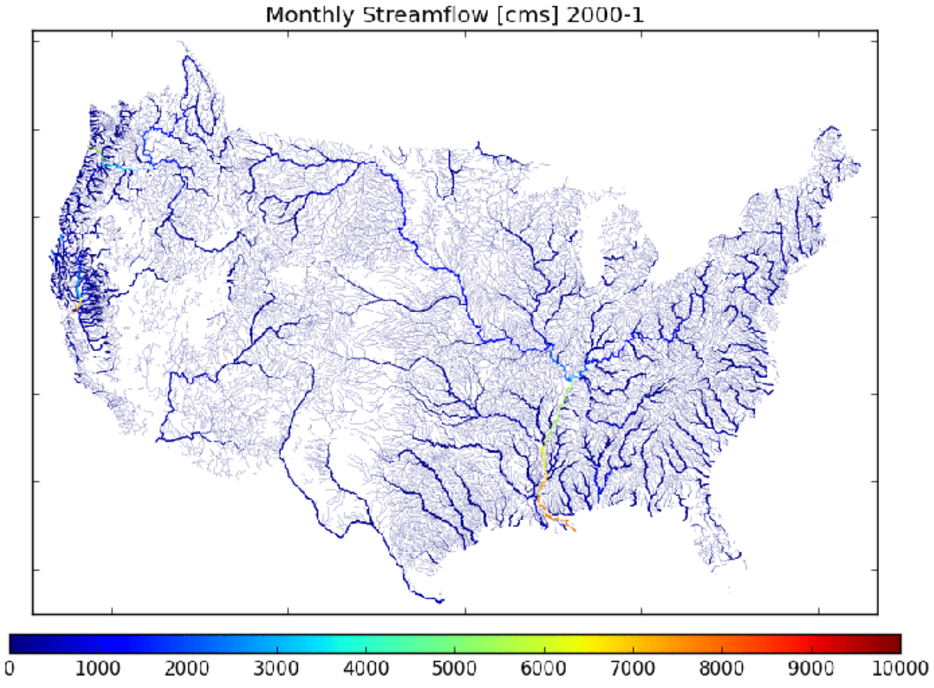
BIG DATA IN THE GEOSCIENCES

We use our observations to test our models… and our models to test our observations

Observations



Simulations



WHAT IS PANGEO?

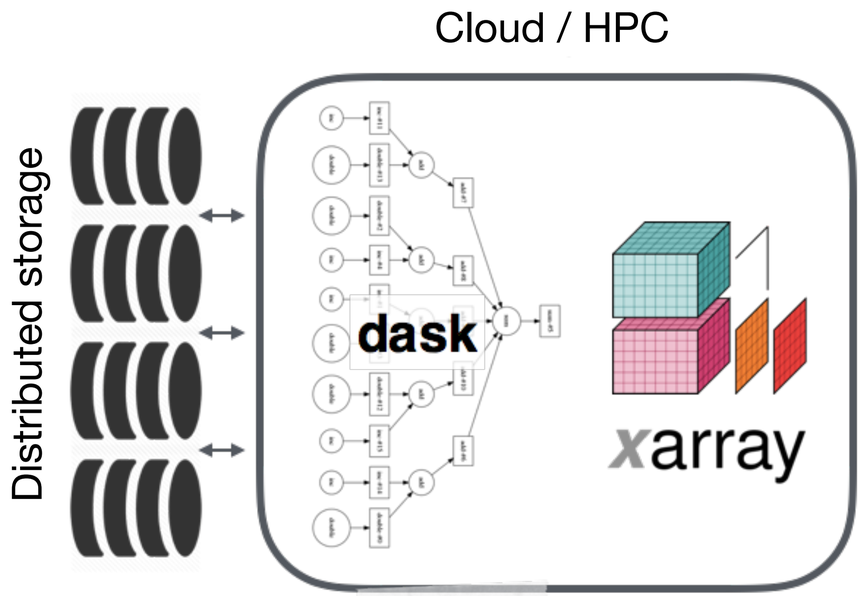
Pangeo is a community working to develop software and infrastructure to enable big-data geoscience.

- **Mission:** To cultivate an ecosystem in which the next generation of open-source analysis tools for the big-data geosciences can be developed, distributed, and sustained.

- **Vision:**
  - Open and collaborative development
  - Tools for scaling computations from small to very large datasets
  - Frameworks for moving scientific analysis to the data
  - Welcoming and inclusive development culture

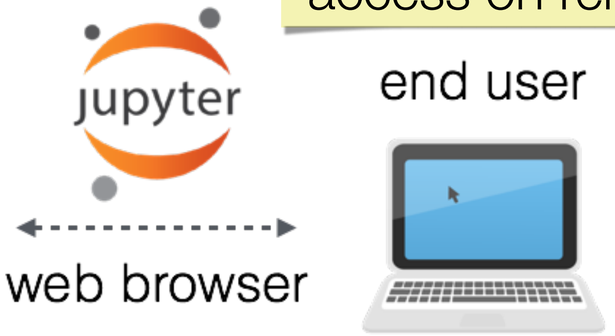
PANGEO ARCHITECTURE

"Analysis Ready Data" stored on globally-available distributed storage.



Parallel computing system allows users deploy clusters of compute nodes for data processing.

Dask tells the nodes what to do.

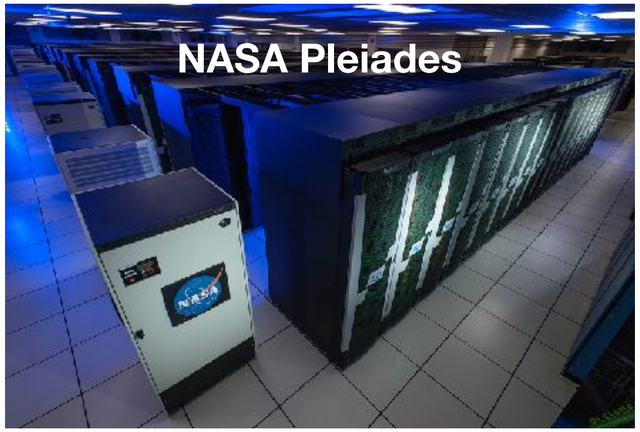


Jupyter for interactive access on remote systems

Xarray provides data structures and intuitive interface for interacting with datasets

PANGEO DEPLOYMENTS

HTTP://PANGEO-DATA.ORG/DEPLOYMENTS.HTML



NCAR Cheyenne

(SCALE USING JOB QUEUE SYSTEM)

PANGEO.PYDATA.ORG

Google Cloud Platform

Over 500 unique users since March!

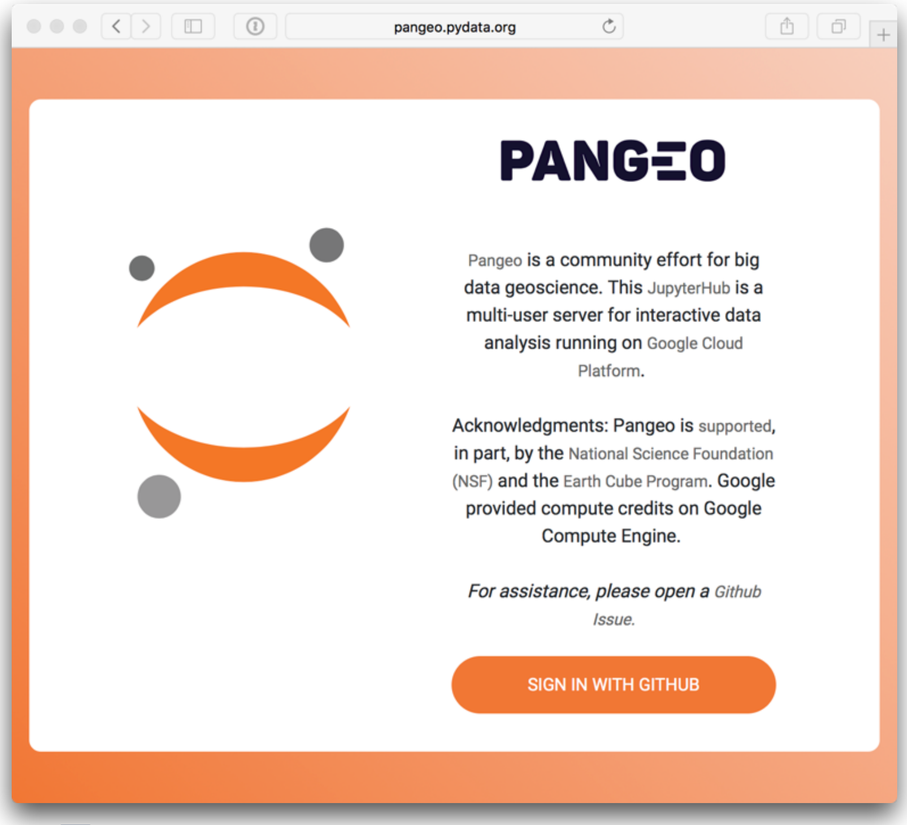
Microsoft Azure

aws

(SCALE USING KUBERNETES)

PANGEO

PANGEO.PYDATA.ORG



- **What is pangeo.pydata.org?**
  - Multi-user JupyterHub running on Google Cloud Platform
  - Zero-to-jupyterhub deployment using Kubernetes
  - Dask scales using “Dask-Kubernetes”
- **Why the cloud?**
  - Highly scalable (storage, compute, user access)
  - Easy to customize
  - Cost effective

Google Cloud Platform

PANGEO

WHAT WOULD YOU LIKE TO HAVE AND WHY?

1. Scalable data-proximate computing
2. **Cloud optimized data formats**
3. **Machine readable catalogs**
4. **Transparent data portals**
5. Helpful scientific IT administrators with cloud-native experience
6. On demand derived data products

CLOUD OPTIMIZED DATA FORMATS FOR MULTI-DIMENSIONAL DATA

- WHAT DO WE WANT?
  - **Self-describing**: data and metadata packaged together
  - **On-demand**: data can be read/used in its current form from anywhere
  - **Analysis-ready**: no pre-processing required
- WHY THE CLOUD?
  - **Too big to move**: assume data is to be used but not copied
  - **Easy to share:** reduces duplicate storage
  - **Scalable:** storage and throughput during computation

GeoTiff files stored in cloud object store support http byte range requests.

?

We don't know what the Cloud Optimized NetCDF will be.

MACHINE READABLE DATA CATALOGS

- Data discovery and access is too hard.
- We need better ways to index/ search existing metadata catalogs
- It should be much easier to use catalogs to get actual data
- As a scientist, 3 lines of code is about the right amount of effort to starting working with a dataset.

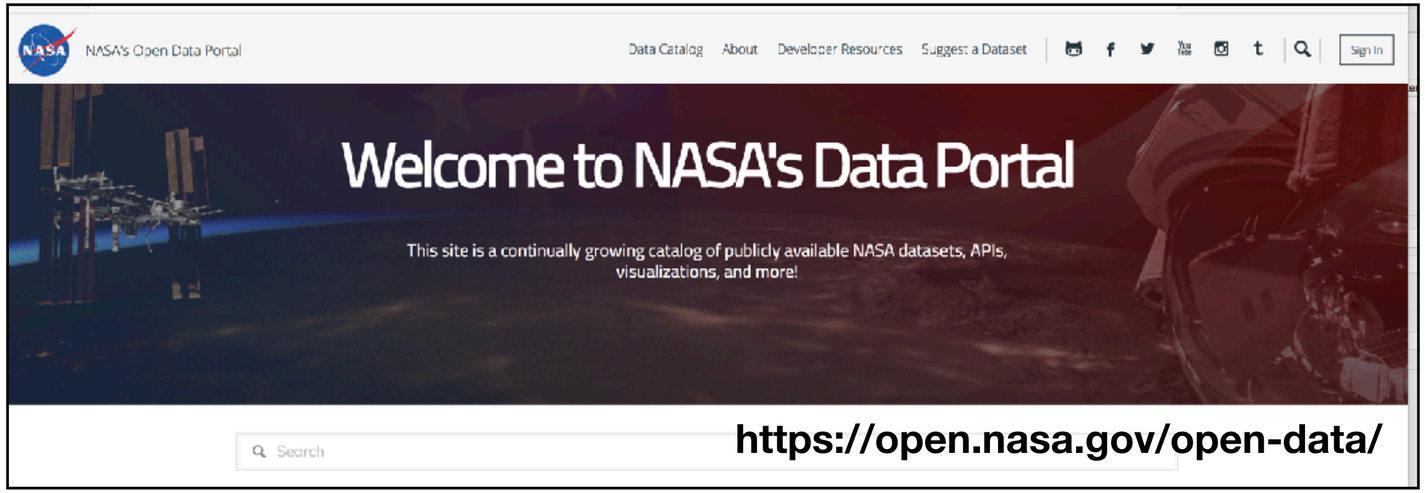
```
In [4]: # Load with intake catalog service
        import intake
        cat = intake.Catalog('https://raw.githubusercontent.com/pangeo-data/pangeo/master/gce/catalog.yaml')
        ds = cat.gmet_v1.read_chunked()

In [5]: # Print dataset
        ds

Out[5]: <xarray.Dataset>
        Dimensions:    (ensemble: 100, lat: 224, lon: 464, time: 12054)
        Coordinates:
          * ensemble   (ensemble) int64 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 ...
          * lat        (lat) float64 25.06 25.19 25.31 25.44 25.56 25.69 25.81 25.94 ...
          * lon        (lon) float64 -124.9 -124.8 -124.7 -124.6 -124.4 -124.3 ...
          * time       (time) datetime64[ns] 1980-01-01 1980-01-02 1980-01-03 ...
        Data variables:
            elevation  (lat, lon) float64 dask.array<shape=(224, 464), chunksize=(224, 464)>
            mask       (lat, lon) int32 dask.array<shape=(224, 464), chunksize=(224, 464)>
            pcp        (ensemble, time, lat, lon) float64 dask.array<shape=(100, 12054, 224, 464), chunksize=(1, 366, 224, 464)>
            t_max      (ensemble, time, lat, lon) float64 dask.array<shape=(100, 12054, 224, 464), chunksize=(1, 366, 224, 464)>
            t_mean     (ensemble, time, lat, lon) float64 dask.array<shape=(100, 12054, 224, 464), chunksize=(1, 366, 224, 464)>
            t_min      (ensemble, time, lat, lon) float64 dask.array<shape=(100, 12054, 224, 464), chunksize=(1, 366, 224, 464)>
            t_range    (ensemble, time, lat, lon) float64 dask.array<shape=(100, 12054, 224, 464), chunksize=(1, 366, 224, 464)>
        Attributes:
            history:                  Version 1.0 of ensemble dataset, created Decem...
            institution:              National Center for Atmospheric Research (NCAR...
            nco_openmp_thread_number: 1
            references:               Newman et al. 2015: Gridded Ensemble Precipita...
            source:                   Generated using version 1.0 of CONUS ensemble ...
            title:                    CONUS daily 12-km gridded ensemble precipitati...
```

Example Python snippet demonstrating the use of Intake (https://intake.readthedocs.io) catalog package with Xarray, Jupyter, and GoogleCloud

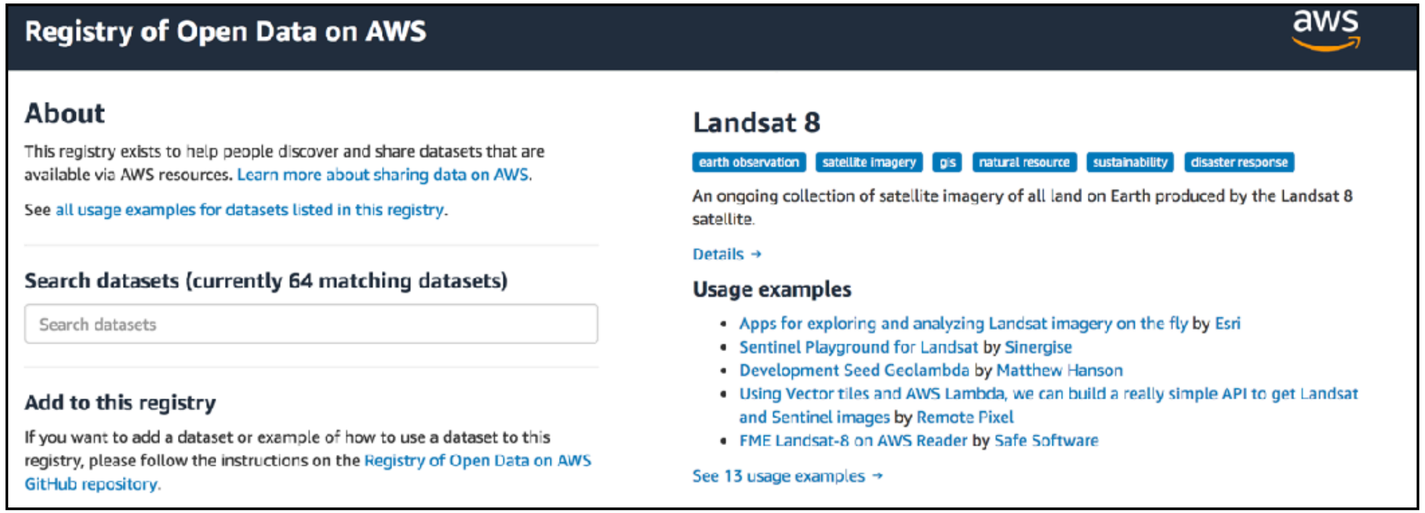
PANGEO

TRANSPARENT DATA PORTALS

- WHAT DO WE WANT?
  - **Intuitive**: organization needs to easy to understand
  - **Simple**: prioritize direct/easy access to datasets instead of fancy interfaces
- WHY?
  - **Manual searches and check boxes don't scale**: portals should supply machine readable data catalogs
  - **Easy to automate:** Batch queries and retrievals



https://open.nasa.gov/open-data/



https://registry.opendata.aws

PANGEO

PANGEO-DATA.ORG

PANGEO.PYDATA.ORG