PANGEO
A COMMUNITY-DRIVEN EFFORT FOR BIG DATA GEOSCIENCE
Assuming that ice forms part of the solid matrix, the effective saturation of soils, directions, and the terms

On the RHS of equation (15), the terms

water content

(i) the generalized Clapeyron equation is combined with the water retention curve to separate the total

Liquid water flow in partially frozen soils is driven by strong capillary pressure gradients that develop as ice

where

In contrast to equations (12) and (13) where

equation (14) the solid precipitation flux occurs only at the top of the snowpack (snowfall over bare ground and at times when the canopy is completely covered with snow). Note that in

the same constitutive functions can be used to relate

and

two terms of the Darcy flux are the capillary and gravity fluxes,

and

and


define rainfall, infiltration-excess runoff and saturation-excess runoff, respectively. Within the soil profile, the

what drives progress in geoscience?
REDUCING TIME TO SCIENCE WITH PANGEO (AN OUTLINE)

- Familiar software ecosystem
- Data-proximate deployments
- Scalability
- Emphasis on next-generation data storage formats for the geosciences
- Demonstration
THE BIG DATA GEOSCIENCE ERA IS NOW

• The geosciences are facing a data volume crisis
• From Earth System Models:
  • Higher resolution
  • More process representation
  • Larger ensembles
  • On track for exabytes by CMIP7

• From Remote Sensing Platforms:
  • New sensors / platforms
  • Continuous observations
  • Multiple versions of derived datasets
1. Software

- Few tangible incentives to share source code (funding agencies, journals)
- Lack of extensible development patterns; often it is easier to “home grow” your own solution, rather than using someone else’s.
- Result is that most geoscientific research is effectively unreproducible and prone to failure.

2. Data sprawl

- Inefficiencies of many copies of the same datasets (“dark replicas”)
- Lessons learned from the CMIP archives (CMIP3 was duplicated > 30x)

3. Local vs. High-performance vs. Cloud Computing

- Traditional scientific computing workflows are difficult to port from a laptop, to HPC, to the cloud
Growth of major programming languages
Based on Stack Overflow question views in World Bank high-income countries

Stack Overflow Traffic to Questions About Selected Python Packages
Based on visits to Stack Overflow questions from World Bank high-income countries

source: stackoverflow.com
SCIENTIFIC PYTHON FOR DATA SCIENCE

Credit: Stephan Hoyer, Jake Vanderplas (SciPy 2015)
XARRAY DATASET: MULTIDIMENSIONAL VARIABLES WITH COORDINATES AND METADATA

Data variables
used for computation

Coordinates
describe data

Indexes
align data

Attributes
metadata ignored by operations

“netCDF meets pandas.DataFrame”

Credit: Stephan Hoyer
import xarray as xr

ds = xr.open_dataset('NOAA_NCDC_ERSST_v3b_SST.nc')

ds

<xarray.Dataset>
Dimensions: (lat: 89, lon: 180, time: 684)
Coordinates:
  * lat  (lat) float32 -88.0 -86.0 -84.0 -82.0 -80.0 -78.0 -76.0 -74.0 ...
  * lon  (lon) float32 0.0 2.0 4.0 6.0 8.0 10.0 12.0 14.0 16.0 18.0 20.0 ...
  * time (time) datetime64[ns] 1960-01-15 1960-02-15 1960-03-15 ...
Data variables:
  sst  (time, lat, lon) float64 nan nan nan nan nan nan nan nan ...
Attributes:
  Conventions: IRIDL
  source: https://iridl.ldeo.columbia.edu/SOURCES/.NOAA/.NCDC/.ERSST/...
# select and plot data from my birthday
```python
ds.sst.sel(time='1982-08-07', method='nearest').plot()
```
# zonal and time mean temperature

ds.sst.mean(dim=('time', 'lon')).plot()
XARRAY: GROUPING AND AGGREGATION

```python
sst_clim = sst.groupby('time.month').mean(dim='time')
sst_anom = sst.groupby('time.month') - sst_clim
nino34_index = (sst_anom.sel(lat=slice(-5, 5), lon=slice(190, 240))
               .mean(dim=('lon', 'lat'))
               .rolling(time=3).mean(dim='time'))
nino34_index.plot()
```
XARRAY

https://github.com/pydata/xarray

- label-based indexing and arithmetic
- interoperability with the core scientific Python packages (e.g., pandas, NumPy, Matplotlib)
- out-of-core computation on datasets that don’t fit into memory (thanks dask!)
- wide range of input/output (I/O) options: netCDF, HDF, geoTIFF, zarr
- advanced multi-dimensional data manipulation tools such as group-by and resampling
Complex computations represented as a graph of individual tasks.

Scheduler optimizes execution of graph.

ND-Arrays are split into chunks that comfortably fit in memory.
EXAMPLE CALCULATION: TAKE THE MEAN!

### Multidimensional Array

- **x, 0, 0**
- **x, 0, 1**
- **x, 0, 2**
- **x, 1, 0**
- **x, 1, 1**
- **x, 1, 2**
- **x, 2, 0**
- **x, 2, 1**
- **x, 2, 2**
- **x, 3, 0**
- **x, 3, 1**
- **x, 3, 2**

### Serial Execution (a Loop)

- **Read chunk from disk** → **Reduce** → **Store** → **Read chunk from disk** → **Reduce** → **Store** → **Read chunk from disk** → **Reduce** → **Store** → **Reduce**
EXAMPLE CALCULATION: TAKE THE MEAN!

parallel execution (dask graph)
PANGEO PROJECT GOALS

• Foster collaboration around the open source scientific python ecosystem for ocean / atmosphere / land / climate science.

• Support the development with domain-specific geoscience packages.

• Improve scalability of these tools to handle petabyte-scale datasets on HPC and cloud platforms.
EarthCube Award Team

Lamont-Doherty Earth Observatory
Columbia University | Earth Institute
Ryan Abernathey, Chiara Lepore, Michael Tippet, Naomi Henderson, Richard Seager

NCAR
National Center for Atmospheric Research
Kevin Paul, Joe Hamman, Ryan May, Davide Del Vento

Anaconda
Powered by Continuum Analytics
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 recently funded ACCESS award to UW / NCAR 🎉

Yuvi Panda, Chris Holdgraf
Spending lots of time helping us make things work on the cloud
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 remote systems

web browser

Xarray provides data structures and intuitive interface for interacting with datasets

Distributed storage

Cloud / HPC

end user
<table>
<thead>
<tr>
<th>Build Your Own Pangeo</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Storage Formats</strong></td>
</tr>
<tr>
<td><strong>ND-Arrays</strong></td>
</tr>
<tr>
<td><strong>Data Models</strong></td>
</tr>
<tr>
<td><strong>Processing Mode</strong></td>
</tr>
<tr>
<td><strong>Compute Platform</strong></td>
</tr>
</tbody>
</table>
PANGEO DEPLOYMENTS

http://pangeo-data.org/deployments.html

Over 500 unique users since March!

 SCALE USING JOB QUEUE SYSTEM

 SCALE USING KUBERNETES
SHARING DATA IN THE CLOUD

Traditional Approach: A Data Access Portal

Data Granules (netCDF files)

- file.0001.nc
- file.0002.nc
- file.0003.nc
- file.0004.nc

Data Access Server

Client

Data Center

Internet
ON-DEMAND ANALYSIS-READY DATA

- **Too big to move**: assume data is to be used but not copied
- **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
Direct Access to Cloud Object Storage

Data Granules
(netCDF files or something new)

Cloud Object Storage

chunk.0.0.0
chunk.0.0.1
chunk.0.0.2
chunk.0.0.3

Catalog

Cloud Compute Instances

Client
Client
Client

Cloud Data Center
# Dask Scales Compute... Can the Storage Layer Keep Up?

<table>
<thead>
<tr>
<th>Cloud Optimized GeoTIFF</th>
<th>HDF + FUSE</th>
<th>HDF + Custom Reader</th>
<th>Build a Distributed Service</th>
<th>New Storage Format (e.g. zarr)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>pros</strong></td>
<td>fast, well-established</td>
<td>works with existing files, no changes to HDF lib needed</td>
<td>works with existing files, no complex FUSE tricks</td>
<td>offloads the problem to others, maintains stable API</td>
</tr>
<tr>
<td><strong>Cons</strong></td>
<td>data model not sophisticated enough</td>
<td>complex, low performance, brittle</td>
<td>Requires plugins to HDF library and tweaks to downstream libs</td>
<td>Complex, introduces intermediary, probably not free</td>
</tr>
</tbody>
</table>

By Matt Rocklin (Anaconda)

http://matthewrocklin.com/blog/work/2018/02/06/hdf-in-the-cloud
HOW TO SHARE A DATASET IN THE CLOUD


• Place your Big Data in cloud object storage in a self-describing, cloud-optimized format.

• Share a public path to your datasets (url/doi/ect)

```json
sea_surface:
  description: sea-surface altimetry data from The Copernicus Marine Environment
  driver: zarr
  args:
    urlpath: gcs://pangeo-data/dataset-duacs-rep-global-merged-allsat-phy-l4-v3-alt
  storage_options:
    token: anon

(EXAMPLE OF A "INTAKE" CATALOG)
HOW TO GET INVOLVED

HTTP://PANGEO-DATA.ORG

- Access and existing Pangeo deployment on an HPC cluster, or cloud resources (eg. pangeo.pydata.org)

- Adapt Pangeo elements to meet your projects needs (data portals, etc.) and give feedback via github: github.com/pangeo-data/pangeo

- Participate in open-source software development!
HANDS ON TIME

• Go to pangeo.pydata.org (requires GitHub credentials)
• Walk through xarray-data.ipynb
• Run a few of the examples
• Try some science of your own

(disclaimers about saving data, long term access, security, etc.)
MORE ON CLOUD NATIVE GEOSCIENCE

• Cloud Native Geospatial Part 2: The Cloud Optimized GeoTIFF
• Towards On-Demand Analysis Ready Data
  • https://medium.com/planet-stories
• Step-by-Step Guide to Building a Big Data Portal
  • https://medium.com/pangeo