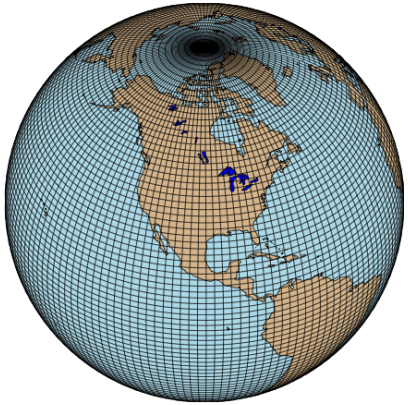
Project Rajin and UXarray: community tools for the analysis of kilometer scale climate and weather model outputs

John Clyne
Acting Director, Technology Development Division
National Center for Atmospheric Research (NCAR)

20th ECMWF workshop on high performance computing in meteorology
Bologna, Italy
October 12, 2023

This presentation is based upon work supported by the National Science Foundation under Grant No.2126458

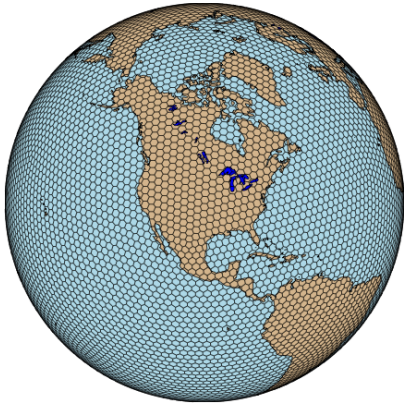
Transition to unstructured meshes in high resolution climate and global weather modeling



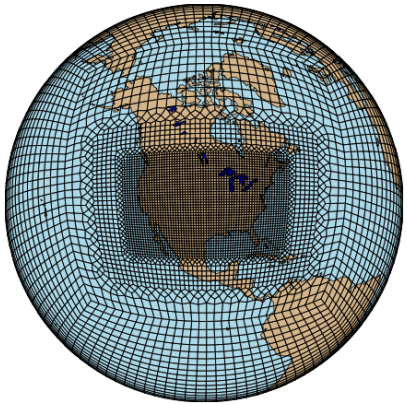
structured lat-lon grid

After more than two decades of development and evaluation the Climate and global weather modeling communities are transitioning from simple structured grids to more complex, but scalable, unstructured grids upon which governing equations of state are solved.

These new models are capable of operating at kilometer scale (*storm resolving*) resolutions.



Icosahedral grid



Variable resolution, cubed sphere grid

Challenges for analysis of model outputs

PROJECT raijin

1. No widely used convention for the storage of unstructured grid data
    - Every model saves data (and metadata) differently
    - No common internal (in RAM) data structure either

1. Few analysis tools capable of working directly with unstructured grids data
    - Resampling to structured grids has numerous pitfalls

1. Global storm resolving resolution models generating LOTS of data
    - Further exacerbating problems with limited set of tools that operate directly on unstructured meshes

1. Analysis operations that are trivial and efficient on structured data can become complex and computationally expensive on unstructured meshes
    - E.g. Efficiently finding the cell containing a point in an unstructured mesh requires an acceleration data structure such as a kd-tree

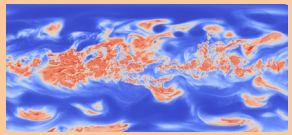
Project Raijin Goals

Extensible, scalable, open source software for analysis on unstructured grids

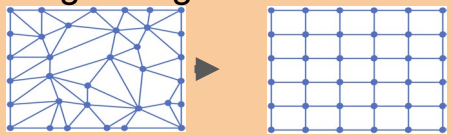
Computational operators

$\mathcal{F}(x)$ $\iint$ $\frac{\partial x}{\partial y}$ $\mathcal{L}(x)$

Plotting



Regridding



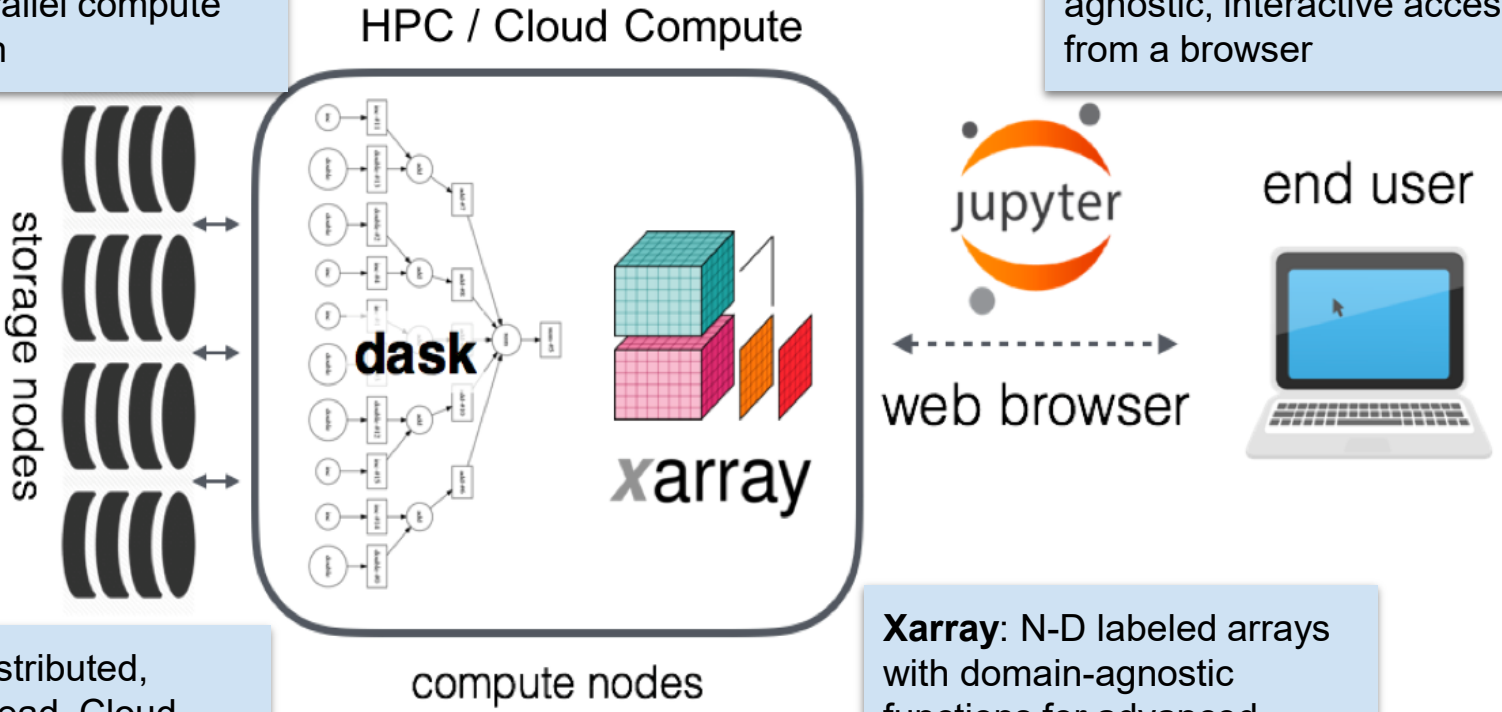
+

Sustainably, community owned

Pangeo: A community platform for Big Data geoscience

Dask: parallel compute for Python

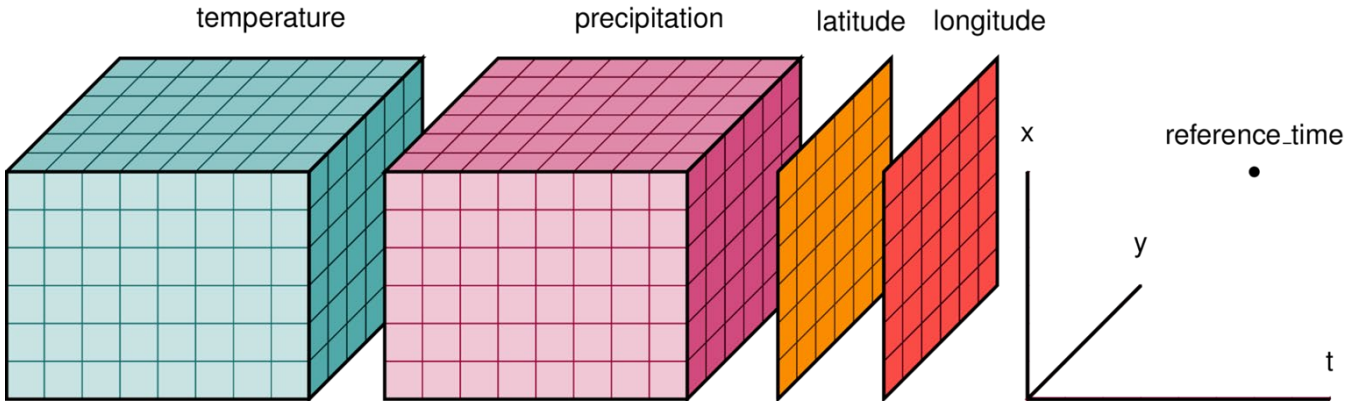
JupyterHub: platform agnostic, interactive access from a browser



Globally distributed, Analysis Read, Cloud Optimized (**ARCO**) data

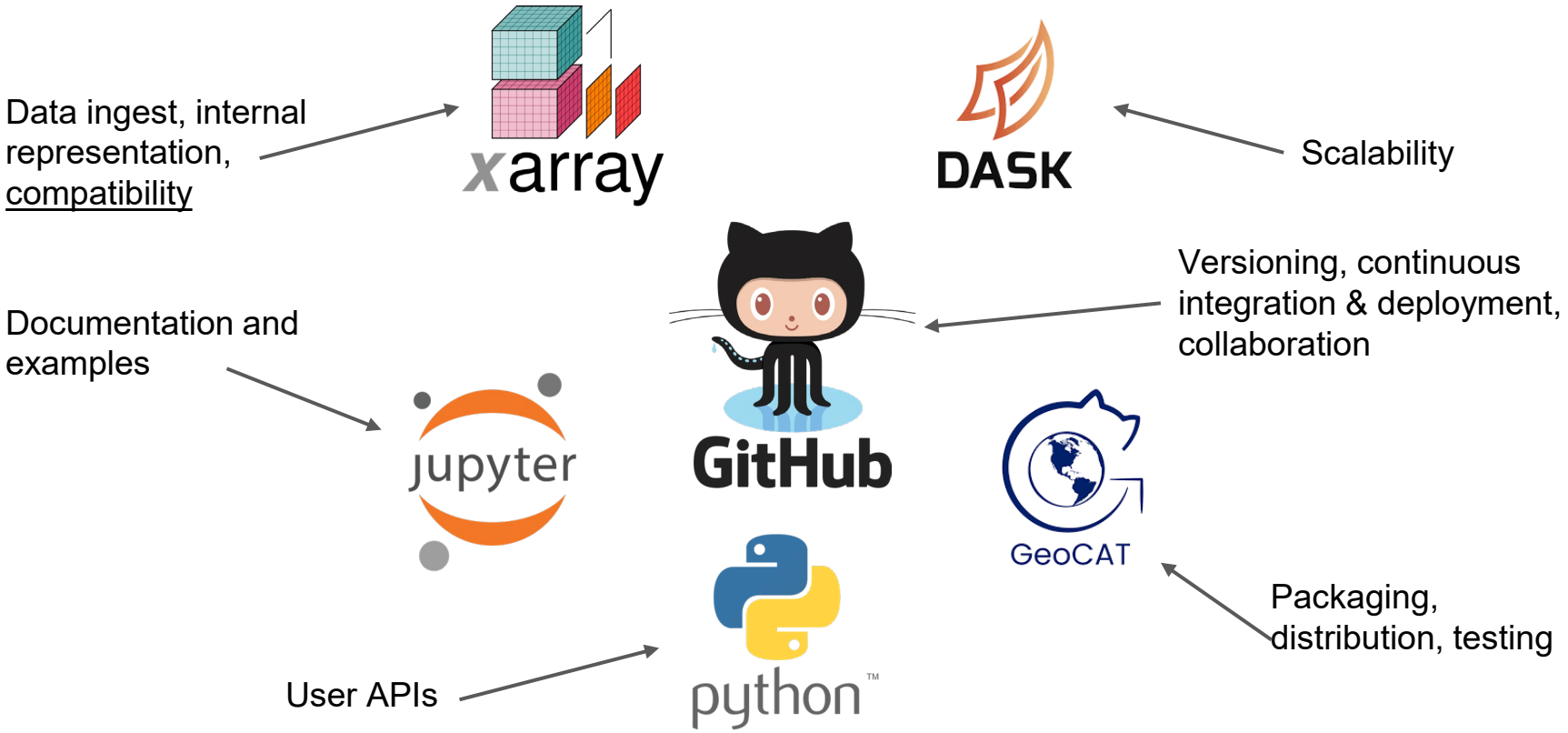
Xarray: N-D labeled arrays with domain-agnostic functions for advanced analytics and visualization

Xarray: The Structured Scientific Data Model
What makes it so popular with geoscientists?

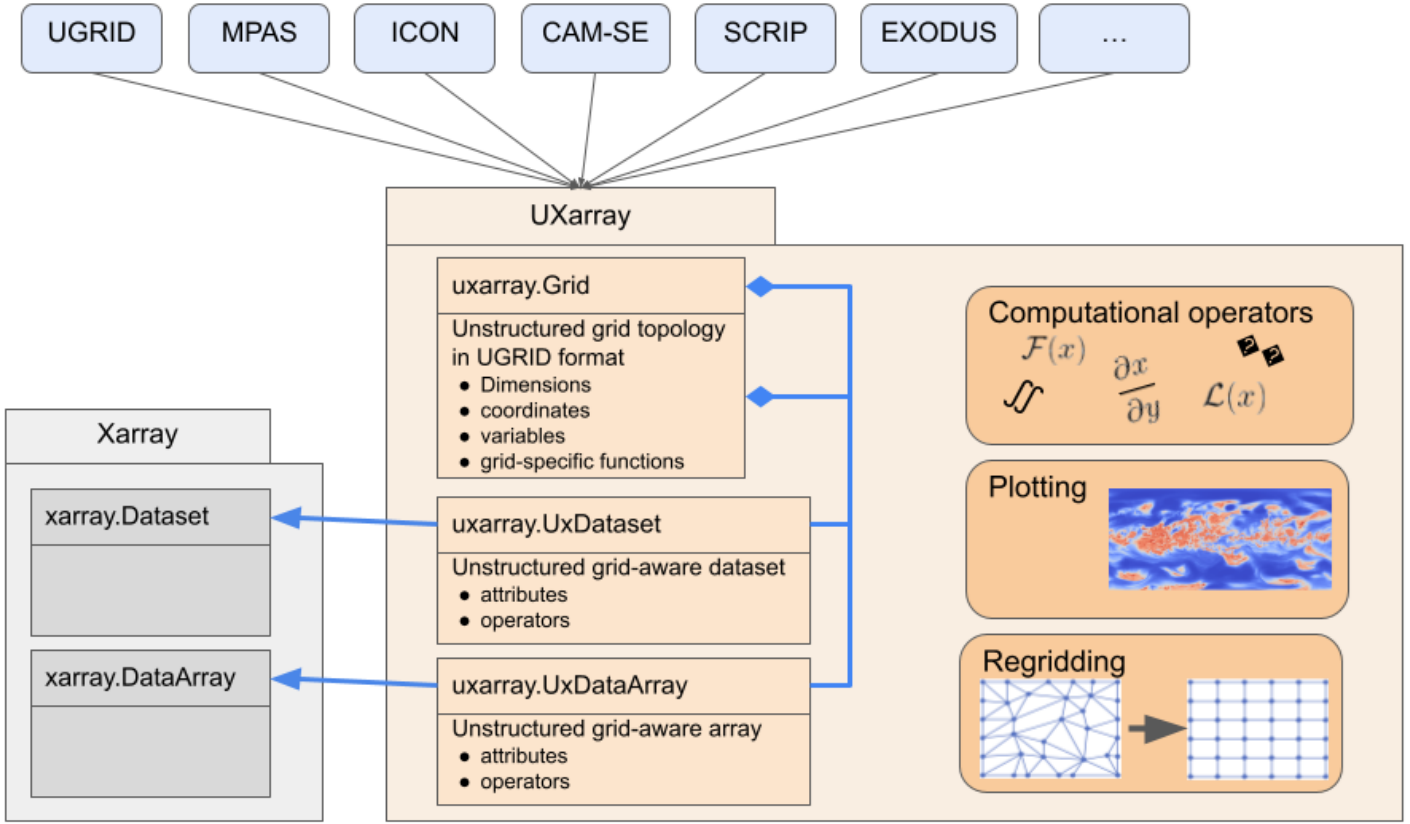


- **Convenience**: Select values by label, not integer location
- **Popular geoscience I/O backends**: NetCDF, GRIB, OpenDap, HDF, Zarr
- **Interoperability**: works with the scientific Python ecosystem including NumPy, Dask, Pandas, and Matplotlib
- **Performant**: Operators are vectorized, implemented in compiled code, and easily parallelized (usually) with Dask
- **Ease of use**: E.g. overloaded operators, such as `plot()`, that just “do the right thing”

Core technologies from Pangeo



UXarray: a specialization (class extension) of Xarray that supports unstructured grids and is Pangeo compatible



A sampling of UXarray public class methods

Operators inherited as is from Xarray (grid info not needed!)

```
Import uxarray as ux
…
ux.argmin()
ux.mean()
ux.where()
```

Operators reimplemented from Xarray

```
Import uxarray as ux
…
ux.integrate()
ux.to_netcdf()
ux.plot()
```

New operators added by UXarray

```
Import uxarray as ux
…
ux.to_geodataframe()
ux.calc_total_face_area()
ux.to_polycollection()
```

For users already using Xarray, UXarray should look pretty familiar!

Co-design of UXarray

Using Agile methodologies (iterative development), NCAR coordinates efforts of Software Engineers at NCAR, Argonne National Lab, and U.C. Davis.

Science partners at NCAR, U.C. Davis, Penn. State University meet with development team monthly (or more frequently)

Science teams' roles:

- Integration of domain knowledge
- Requirements and prioritization
- Usability
- Verification and validation

SIMA

EarthWorks

Building the Raijin community

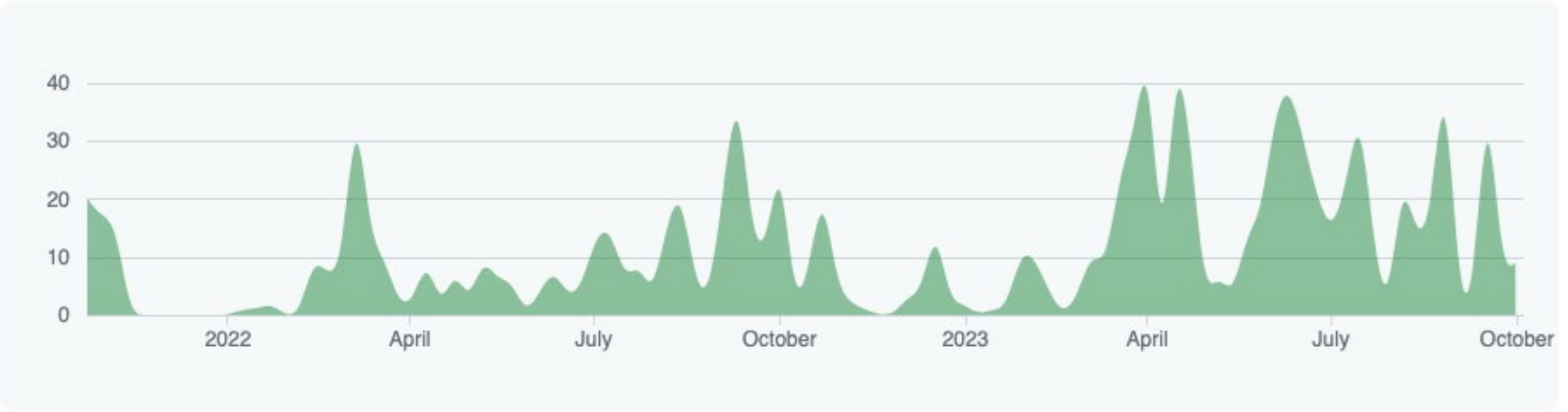
- Everything on *public* GitHub
- Open Source (Apache 2.0)
- Build on Pangeo community AND Pangeo software stack
- Detailed contributors guide + code of conduct
- Open discussion on all topics (GitHub Issues and Discussion forum)
- Socialize major design decisions
- CI/CD infrastructure
- Advocacy (conferences, discussion forums, events)
- A common programming language understood by scientists and software developers: Python

CI	CI PASSING COVERAGE 76%
Docs	DOCS PASSING
Package	conda-forge v2023.10.0 PYPI V2023.10.0
License	LICENSE APACHE-2.0
Citing	DOI 10.5281/zenodo.8404110

Building the Raijin community

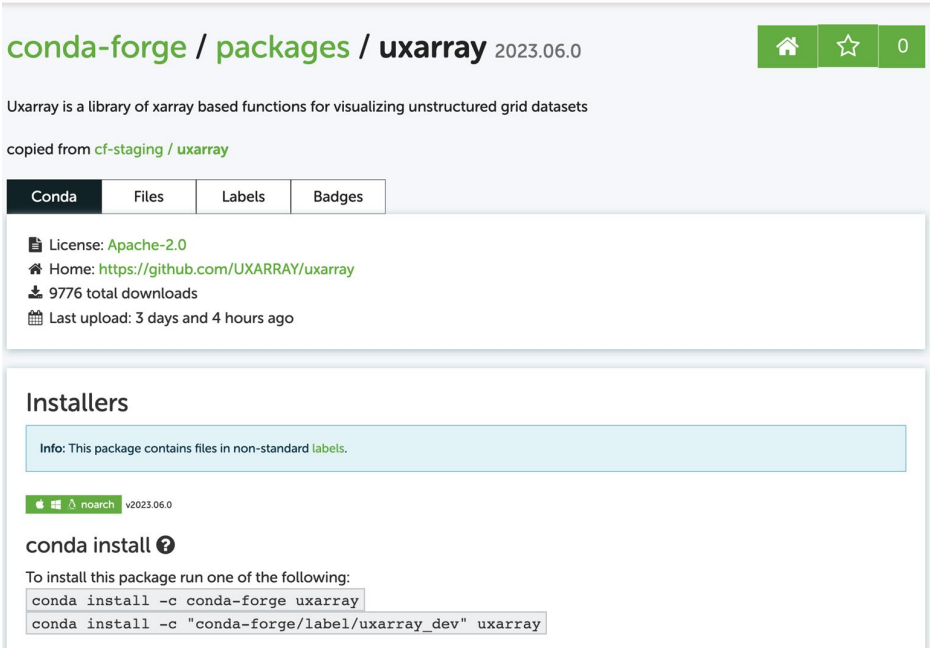
Activity to date:

- 11 external code contributors (347 “commits”)
- 15 participants in API design discussion
- 16 people with feature requests
- DOE and Earthworks partnerships



Current status

- Functioning internal data model (UXarray)
- Public facing website
  - https://raijin.ucar.edu/
- Continuous Integration pipeline
- Detailed contributor's guide
- Monthly releases on conda
- Comprehensive user documentation
- Detailed examples (Jupyter Notebooks)



Current status

Supported models / file formats:

- UGRID, MPAS, CAM-SE, SCRIP, EXODUS

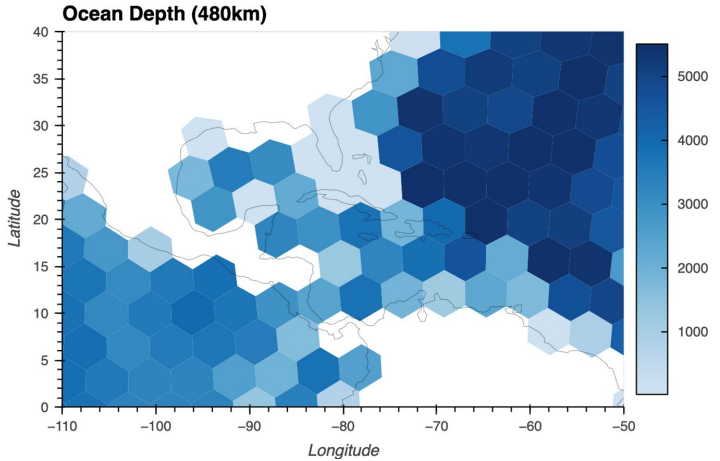
Grid inspection

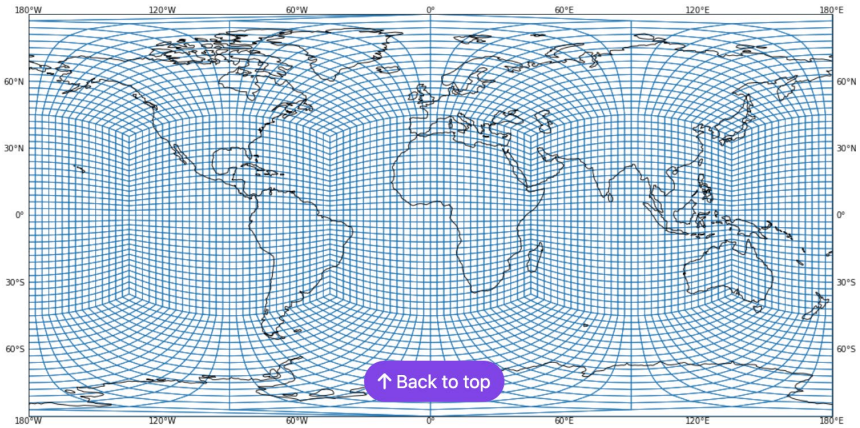
Computational operators

- Integration

Plotting

- Graphic primitive generation





Future work

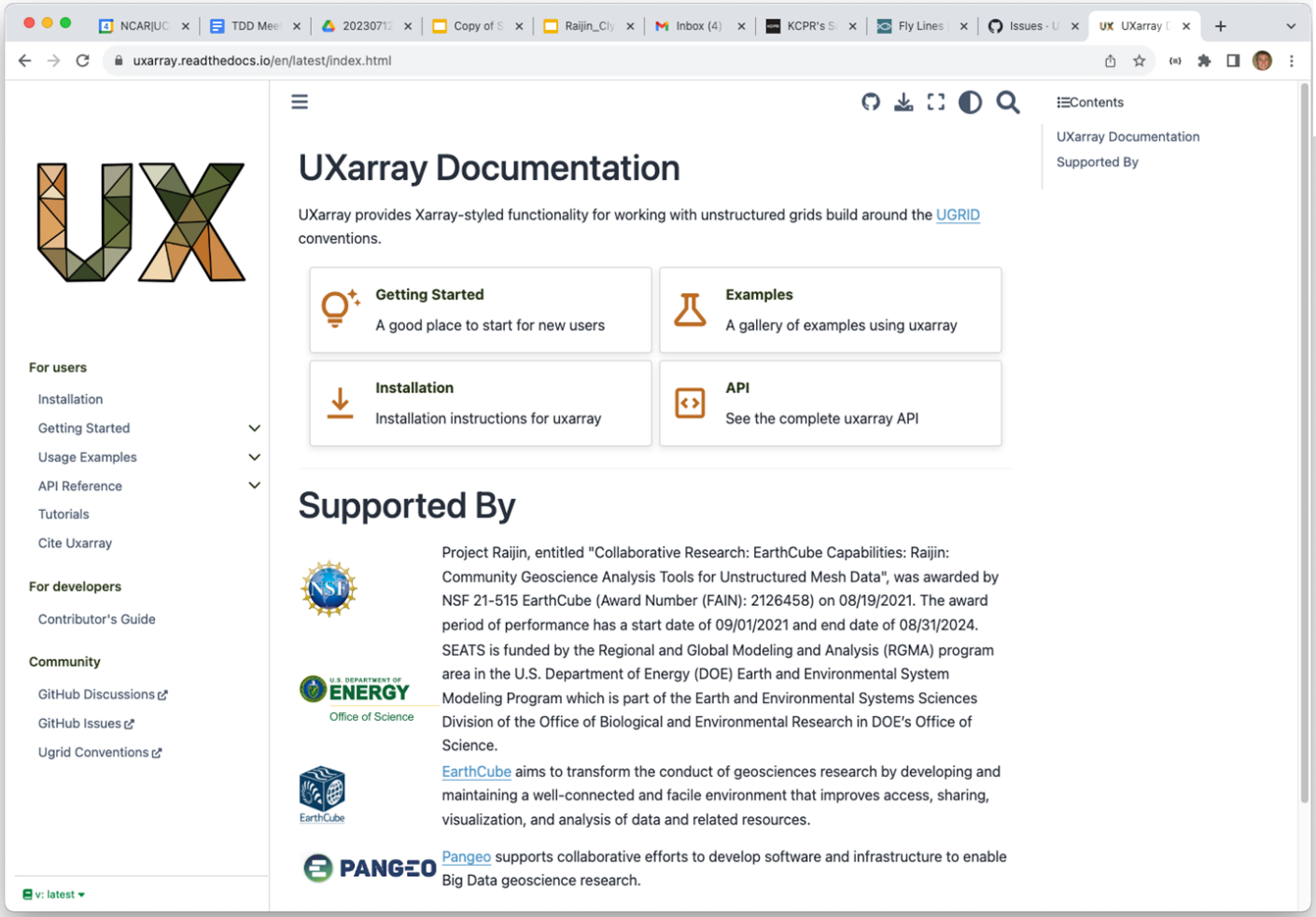
Work in progress

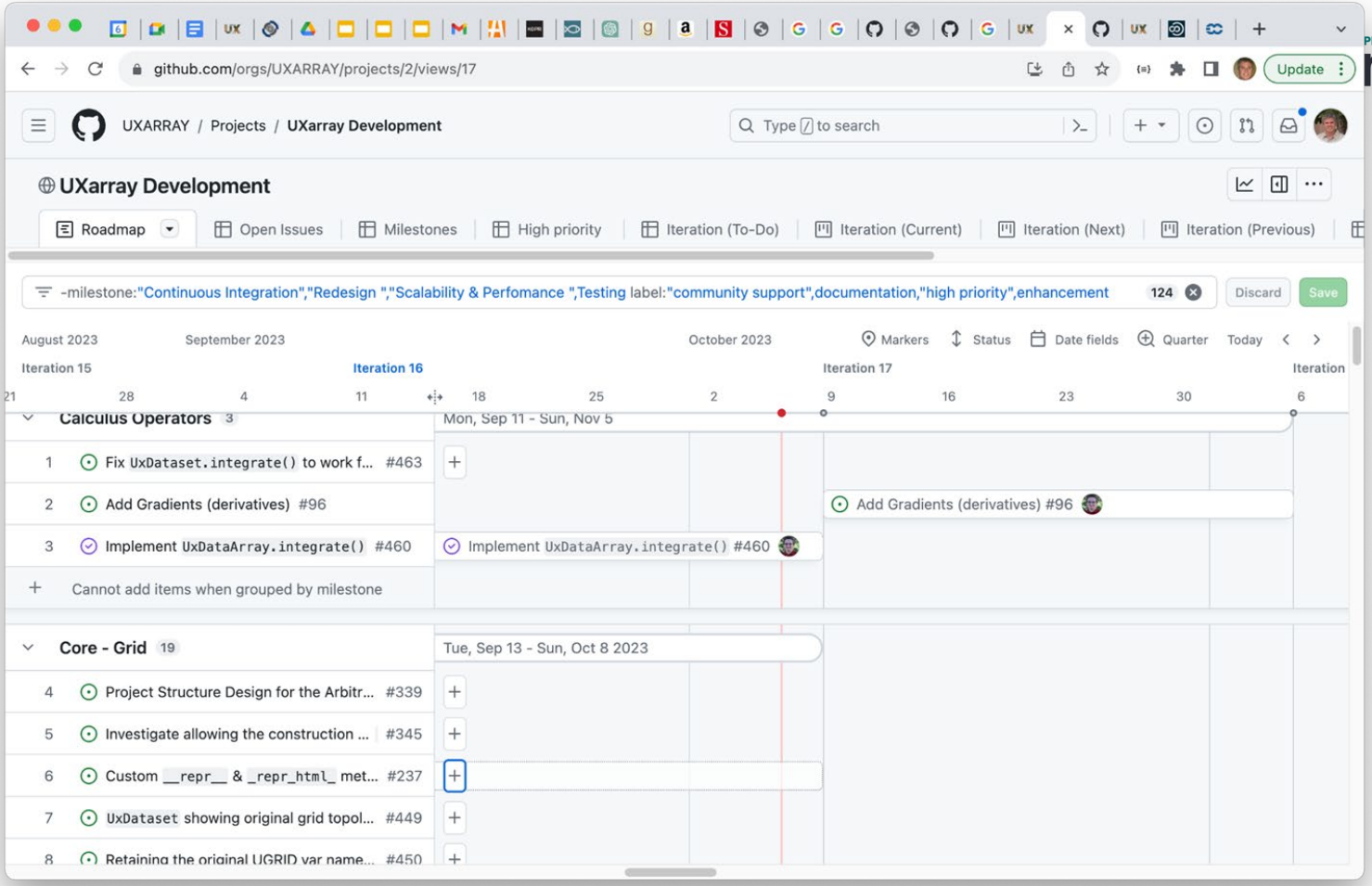
- Derivatives
- Global and zonal means (conservative and non-conservative)
- Integrated plotting
  - E.g. `uxarray.plot()`

Longer term

- New operators (computational functions)
- Regridding
- ICON support (reader)
- Subsetting

Release notes

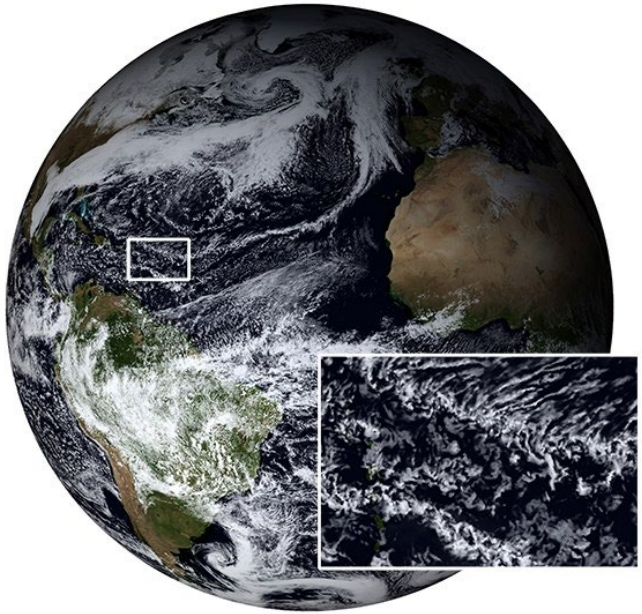

uxarray.readthedocs.io/en/latest/index.html
UXarray Documentation
UXarray provides Xarray-styled functionality for working with unstructured grids build around the UGRID conventions.
Getting Started
A good place to start for new users
Examples
A gallery of examples using uxarray
Installation
Installation instructions for uxarray
API
See the complete uxarray API
Supported By
Project Raijin, entitled "Collaborative Research: EarthCube Capabilities: Raijin: Community Geoscience Analysis Tools for Unstructured Mesh Data", was awarded by NSF 21-515 EarthCube (Award Number (FAIN): 2126458) on 08/19/2021. The award period of performance has a start date of 09/01/2021 and end date of 08/31/2024.
SEATS is funded by the Regional and Global Modeling and Analysis (RGMA) program area in the U.S. Department of Energy (DOE) Earth and Environmental System Modeling Program which is part of the Earth and Environmental Systems Sciences Division of the Office of Biological and Environmental Research in DOE's Office of Science.
EarthCube aims to transform the conduct of geosciences research by developing and maintaining a well-connected and facile environment that improves access, sharing, visualization, and analysis of data and related resources.
Pangeo supports collaborative efforts to develop software and infrastructure to enable Big Data geoscience research.
For users
Installation
Getting Started
Usage Examples
API Reference
Tutorials
Cite Uxarray
For developers
Contributor's Guide
Community
GitHub Discussions
GitHub Issues
Ugrid Conventions
v: latest
Contents
UXarray Documentation
Supported By


github.com/orgs/UXARRAY/projects/2/views/17
Update
UXARRAY / Projects / UXarray Development
Type / to search
UXarray Development
Roadmap
Open Issues
Milestones
High priority
Iteration (To-Do)
Iteration (Current)
Iteration (Next)
Iteration (Previous)
-milestone:"Continuous Integration","Redesign ","Scalability & Perfomance ",Testing label:"community support",documentation,"high priority",enhancement
124
Discard
Save
August 2023
September 2023
October 2023
Markers
Status
Date fields
Quarter
Today
Iteration 15
Iteration 16
Iteration 17
Iteration
21
28
4
11
18
25
2
9
16
23
30
6
Calculus Operators 3
Mon, Sep 11 - Sun, Nov 5
1 Fix UxDataset.integrate() to work f... #463
2 Add Gradients (derivatives) #96
3 Implement UxDataArray.integrate() #460
Add Gradients (derivatives) #96
Implement UxDataArray.integrate() #460
Cannot add items when grouped by milestone
Core - Grid 19
Tue, Sep 13 - Sun, Oct 8 2023
4 Project Structure Design for the Arbitr... #339
5 Investigate allowing the construction ... #345
6 Custom __repr__ & _repr_html_ met... #237
7 UxDataset showing original grid topol... #449
8 Retaining the original UGRID var name... #450

What about scalability?

Interactive visualization of 3.75km MPAS data from a laptop

- 83,886,080 nodes
- 41,943,042 cells
- Jupyter Notebook
- Bokeh + Datashader



Data courtesy of Falko Judt, NCAR

Methods for Visualizing Unstructured Grid Data

Overview

Unstructured grids are a powerful tool to store Geoscience data. Unlike traditional, structured grids, unstructured grids have flexible geometries and variable resolution. This makes them incredibly useful for filling in irregularly shaped domains like Earth's oceans, or for achieving high resolutions in localized regions. However, working with unstructured datasets comes with additional challenges. The grids are made up of various shapes with varying sizes, so many datasets store additional information that describes their grid's geometry. Before we can plot our data, we must convert this connectivity information into a format compatible with plotting software. In this notebook, we will discuss and compare various ways in which we can visualize unstructured datasets.

Imports

[1]:
```
# Recognition of unstructured grids and data handling
import uxarray as ux

# General Plotting
import cartopy.crs as ccrs

# Plotting with HoloViz
import holoviews as hv
# import hvplot.pandas
import geoviews.feature as gf
```

Dataset Overview

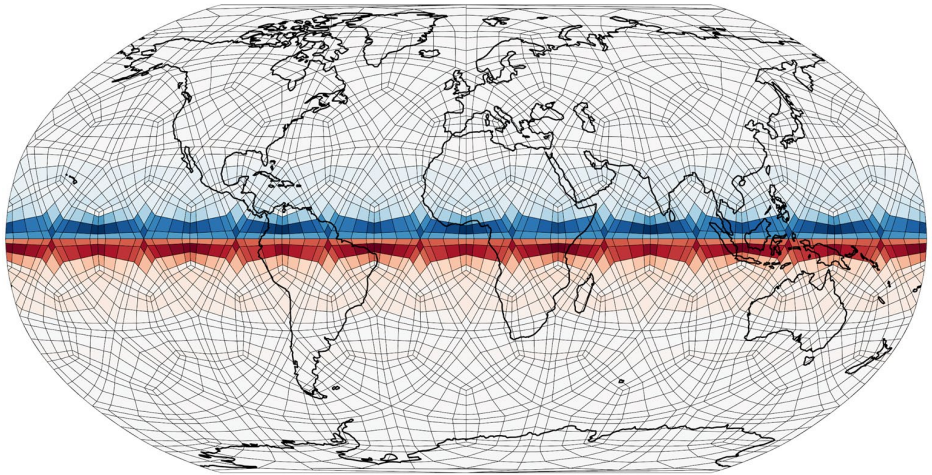
We will be visualizing data, courtesy of NCAR's Falko Judt, and were produced as part of the DYAMOND initiative: http://dx.doi.org/10.1186/s40645-019-0304-z.

The global data sets used in this example are from the same experiment, but run at several resolutions from 30km to 3.75km. Due to their size, the higher resolution data sets are only distributed with two variables in them:

Community Geoscience Analysis Tools for Unstructured Grids

PROJECT raijin

UXarray for visualization



Get involved!

Send us email
projectraijin@googlegroups.com

Start or contribute to a UXarray discussion
https://github.com/UXARRAY/uxarray/discussions

Find out more
https://raijin.ucar.edu

U.S. DEPARTMENT OF ENERGY
Office of Science
NSF
EarthCube
Transforming Geosciences Research

Acknowledgements

NSF Earth Cube program (award #2126458)

U.S. Department of Energy

Pangeo community

Our growing list of contributors on GitHub!

Development of UXarray is supported by: the Regional and Global Modeling and Analysis (RGMA) program area in the Earth Environmental System Modeling Program of the U.S. Department of Energy's (DOE) Office of Science AND the Earthcube program of the U.S. National Science Foundation (NSF)