New open source libraries enable browser visualization of Zarr and other xarray-readable formats, such as NetCDF.

slides

Benchmarking revealed the significance of a chunk size and evaluated performance impacts of other variations.

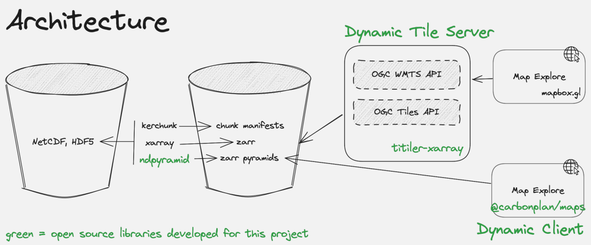
Authors: Aimee Barciauskas (aimee@ds.io), Sean Harkins, Vincent Sarago (Development Seed), Max Jones, Anderson Banihirwe, Kata Martin (CarbonPlan)

Motivation

Visualization is crucial for exploring and understanding Earth data. While web browsers provide a widely accessible platform for data exploration, the immense scale of geospatial data complicates fast rendering. Browsers struggle to quickly read, reproject, and tile this data, impacting user experience.

Drawbacks of pre-generated map tiles

The need to visualize large geospatial datasets has led to the use of pre-generated static map tiles, which allow for quick visualization but come with significant drawbacks. The main issue is that data providers control the data's appearance, limiting user customization like adjusting color scales or combining variables ("band math"). Providers also face challenges like high storage costs and the need to continually update tiles with new data.



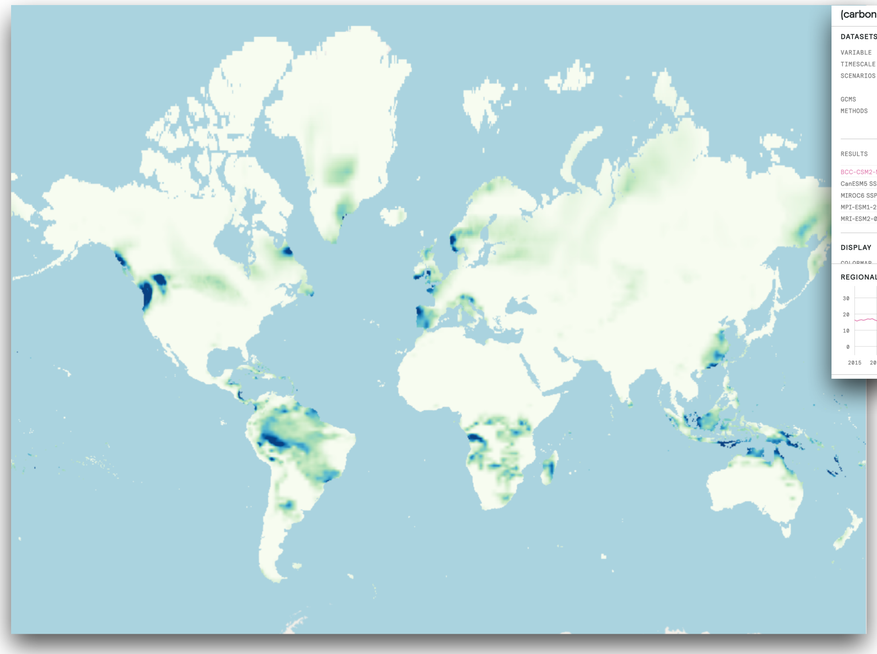
New Methods: Dynamic Tile Server and Dynamic Client

Recent developments in dynamic tiling allow for on-demand map tile creation, traditionally using Cloud-Optimized GeoTIFFs (COGs). With the rise of the Zarr data format for large-scale N-dimensional data, there's increasing demand for browser-based visualization of Zarr. The conventional Zarr chunk size of about 100MB is too large for efficient browser fetching.

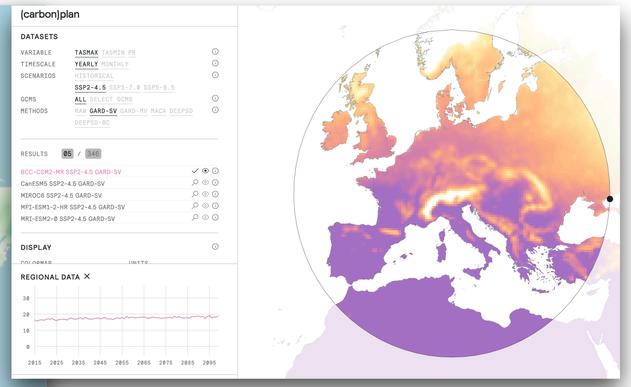
However, there are two options for visualizing Zarr data:

1. A tile server (titiler-xarray) using rio_tiler's XarrayReader can render tiles from any xarray-readable formats, including Zarr, netCDF4/HDF5, and others. However, this approach still necessitates running a server, while the second option,
2. The dynamic client reads Zarr directly in the browser client and uses webGL to render map tiles.

Dynamic tiler: `@developmentseed/titiler-xarray`



Dynamic client: `@carbonplan/maps`



development SEED

(carbon)plan

IMPACT Interagency Implementation and Advanced Concepts Team

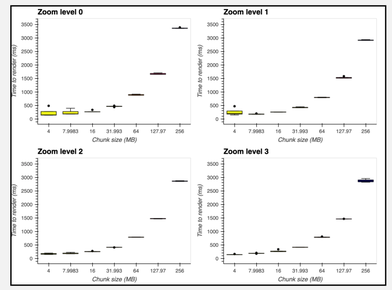
NASA

Benchmarking Results

nasa-impact.github.io/zarr-visualization-report/

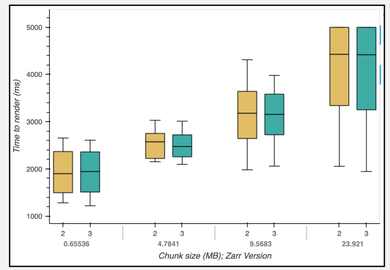
Dynamic Tile Server

Chunk size



Dynamic client

Zarr version



Summary

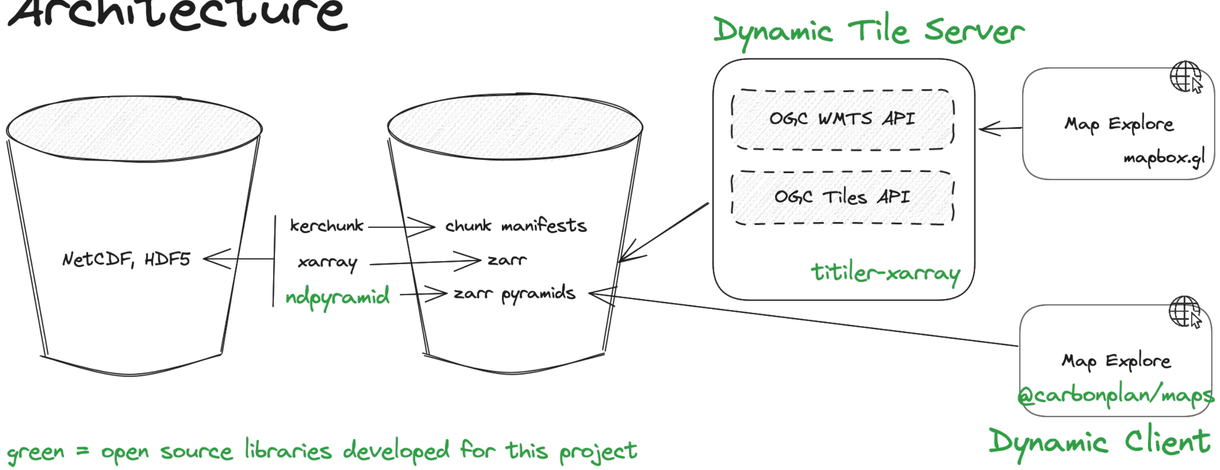
- For both methods, chunk size was found to significantly impact performance and pyramid generation will greatly improve performance for high resolution datasets.
- For the tile server, pre-processing may be **unnecessary** if the data has minimal coordinate chunking, and small chunk sizes (~10MB).

New open source libraries enable browser visualization of Zarr and other xarray-readable formats, such as NetCDF.

Benchmarking revealed the significance of a chunk size and evaluated performance impacts of other variations.

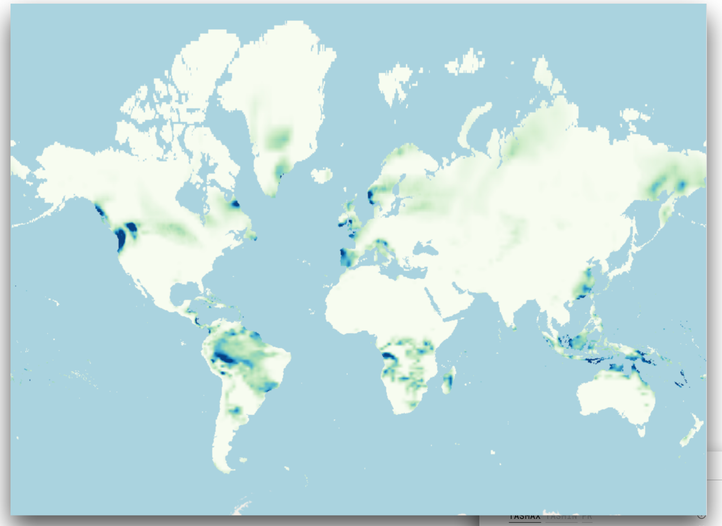
Authors: Aimee Barciauskas (aimee@ds.io), Sean Harkins, Vincent Sarago (Development Seed), Max Jones, Anderson Banihirwe, Kata Martin (CarbonPlan)

Architecture

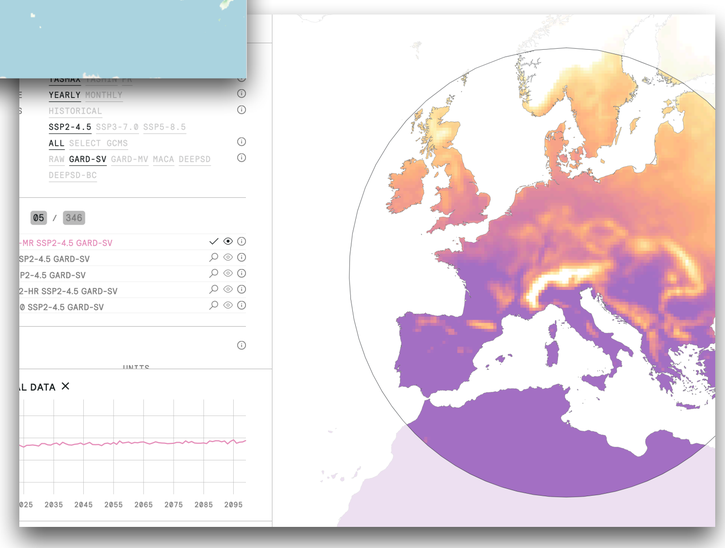


green = open source libraries developed for this project

Dynamic tiler: `@developmentseed/titiler-xarray`



Dynamic client: `@carbonplan/maps`



developmentSEED

(carbon)plan

IMPACT Interagency Implementation and Advanced Concepts Team

NASA