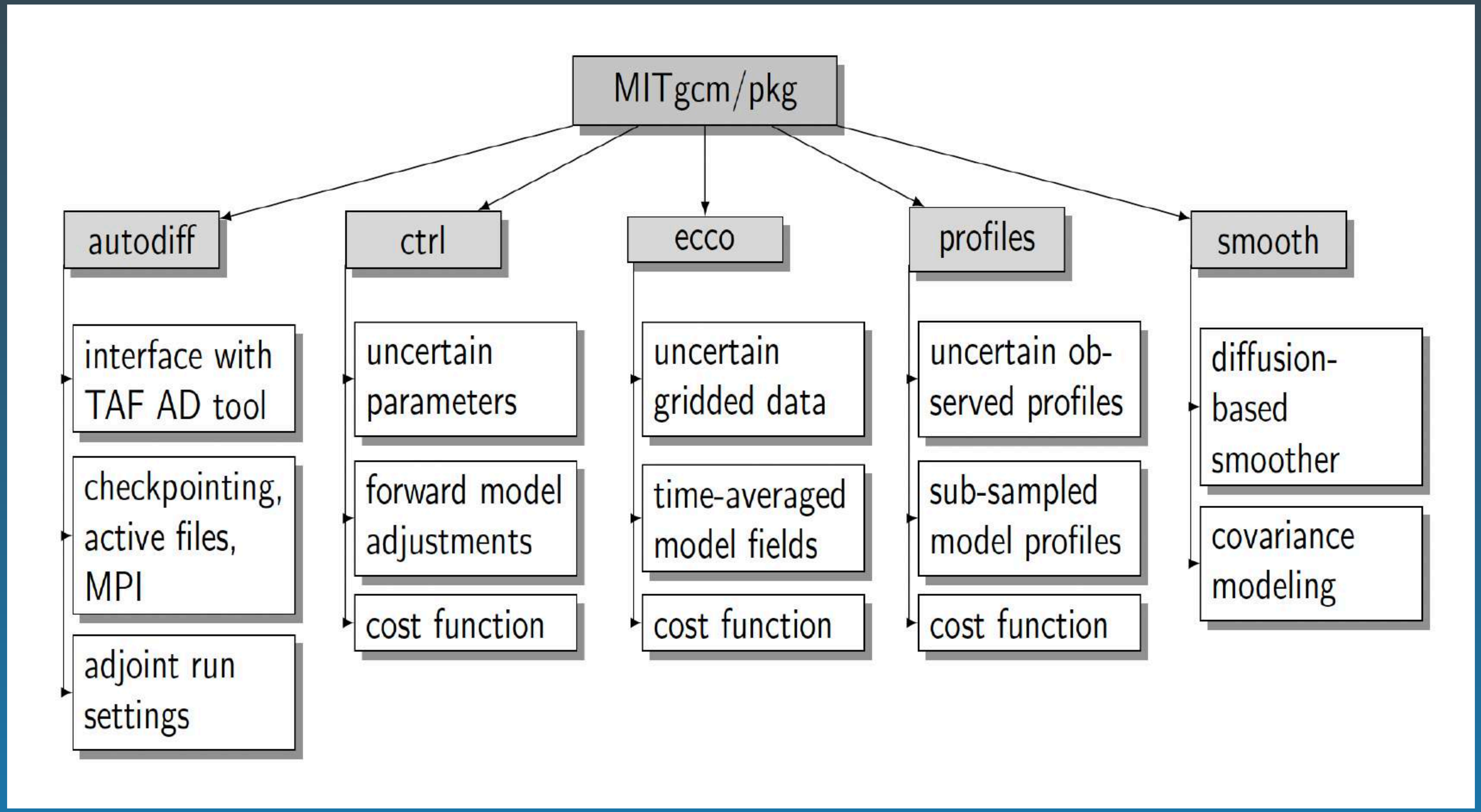
# ECCO with Julia , and Cyberinfrastructure

ECCO annual meeting 2024

Gaël Forget MIT 2024/03/21

# The ECCO4 framework




- The 2 km CS and LLC grids, as well as descending resolution grids
- The MITgcm estimation framework implementation charted in Fig. 5
- The model setup subjected to forward and adjoint daily regression tests
- The state estimate output, including model–data misfits
- The observational data input, including weights, to the state estimate
- The forward model input needed to re-compute the 1992–2011 solution
- The testreport_ecco.m tool to verify re-runs of the 1992–2011 solution
- The gcmfaces Matlab framework to analyze global, gridded solutions
- The MITprof Matlab framework to process and analyze in situ profiles
- The solution's standard analysis produced by gcmfaces and MITprof

**Including**

- ability to rerun forward and adjoint
- generic cost functions, controls
- LLC grids (LLC90, ..., LLC4320)

Forget, Campin, Heimbach, Hill, Ponte, & Wunsch, 2015 : ECCO version 4: an integrated framework for non-linear inverse modeling and global ocean state estimation, GMD, https://doi.org/10.5194/gmd-8-3071-2015

# ECCO cyber-infrastructure (1)

**ECCO v4**

latest

Search docs

CONTENTS:



## Contents:



**2016 : cloud services**

- Github (code, docs),
- Dataverse (input, output)
- running ECCO4 on AWS

**Cloud-based solutions for distributed climate modeling**


Nadya Vinogradova[1], Mark Shiffer[1], Gael Forget[2], and Chris Hill[2]

[1]Cambridge Climate Institute, MA

[2]Massachusetts Institute of Technology, MA

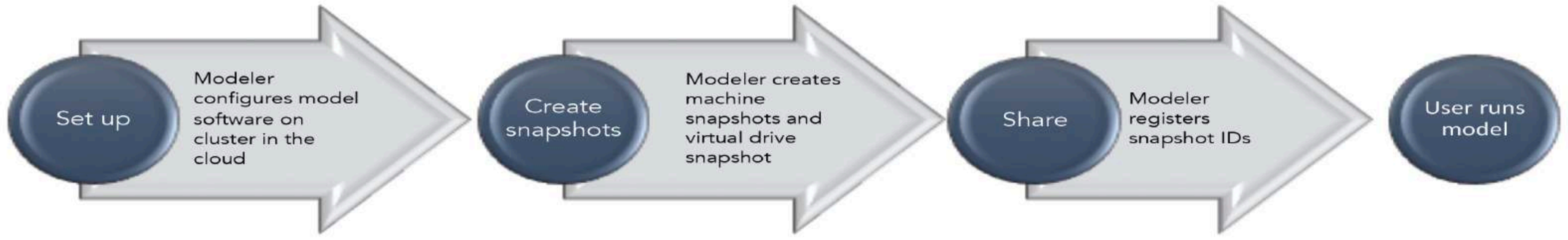




*Fig. 1: Flowchart of the framework*

HARVARD Dataverse

Add Data ▾ | Search ▾ | About | User Guide | Support | Sign Up | Log In

☑ Datasets (23)

☐ Files (3,163)

**Dataverse Category**

Research Project (3)

**Publication Year**

2024 (2)
2023 (1)
2019 (1)
2016 (22)

**Subject**

Earth and Environmental Sciences (26)

**Author Name**

Forget, Gaël (20)
Forget, Gael (3)

**Author Affiliation**

Massachusetts Institute of Technology (23)

**Ocean Heat Content**

Mar 7, 2024 - OCCA Version 2 Dataverse

Forget, Gael, 2024, "Ocean Heat Content", https://doi.org/10.7910/DVN/CAGYQL, Harvard Dataverse, V1

Estimates (OCCA2, ECCO4) of global ocean heat content (OHC) anomaly from 2004-2006 climatology. ECCO4 is a closed heat budget estimate. ECCO4 release 5 is used here that covers 1992-2019. OCCA2 was derived by 1. extending ECCO4 (r2) to 1980-2022 and 2. adding a gridded adjustment...

**OCCA Version 2 Dataverse** (Massachusetts Institute of Technology)

Mar 7, 2024

OCCA stands for OCean Comprehensible Atlas. OCCA2 is the combination of 1. a data-constrained heat budget estimate for 1980-2022, based on the ECCO4 and ERA5 reanalyses, and 2. monthly adjustments based on Argo over 2004-2022.

**Argo / MITprof 2023 release**

Oct 22, 2023

Forget, Gael, 2023, "Argo / MITprof 2023 release", https://doi.org/10.7910/DVN/7HLV09, Harvard Dataverse, V1

- MITprof files were created using ArgoData.jl - copy of the Argo GDAC downloaded on 20230415 was used

**Effective Ocean Heat Transport Estimates**

Jul 10, 2019 - ECCO Version 4 Release 2 Dataverse

Forget, Gael, 2019, "Effective Ocean Heat Transport Estimates", https://doi.org/10.7910/DVN/AVVGYX, Harvard Dataverse, V2

Estimates presented in the Figures and Tables of Forget, G. and Ferreira, D. (2019) Global ocean heat transport dominated by heat export from the tropical Pacific. Nature Geoscience. ISSN 1752-0894 (In Press)

**Documentation Of The ECCO Version 4 Release 2 Dataverse**

Jan 2, 2017 - ECCO Version 4 Release 2 Dataverse

Forget, Gaël, 2016, "Documentation Of The ECCO Version 4 Release 2 Dataverse", https://doi.org/10.7910/DVN/ODM2IQ, Harvard Dataverse, V3

See README.pdf

# ECCO4 integration with Darwin3

## Welcome to the CBIOMES-global documentation!

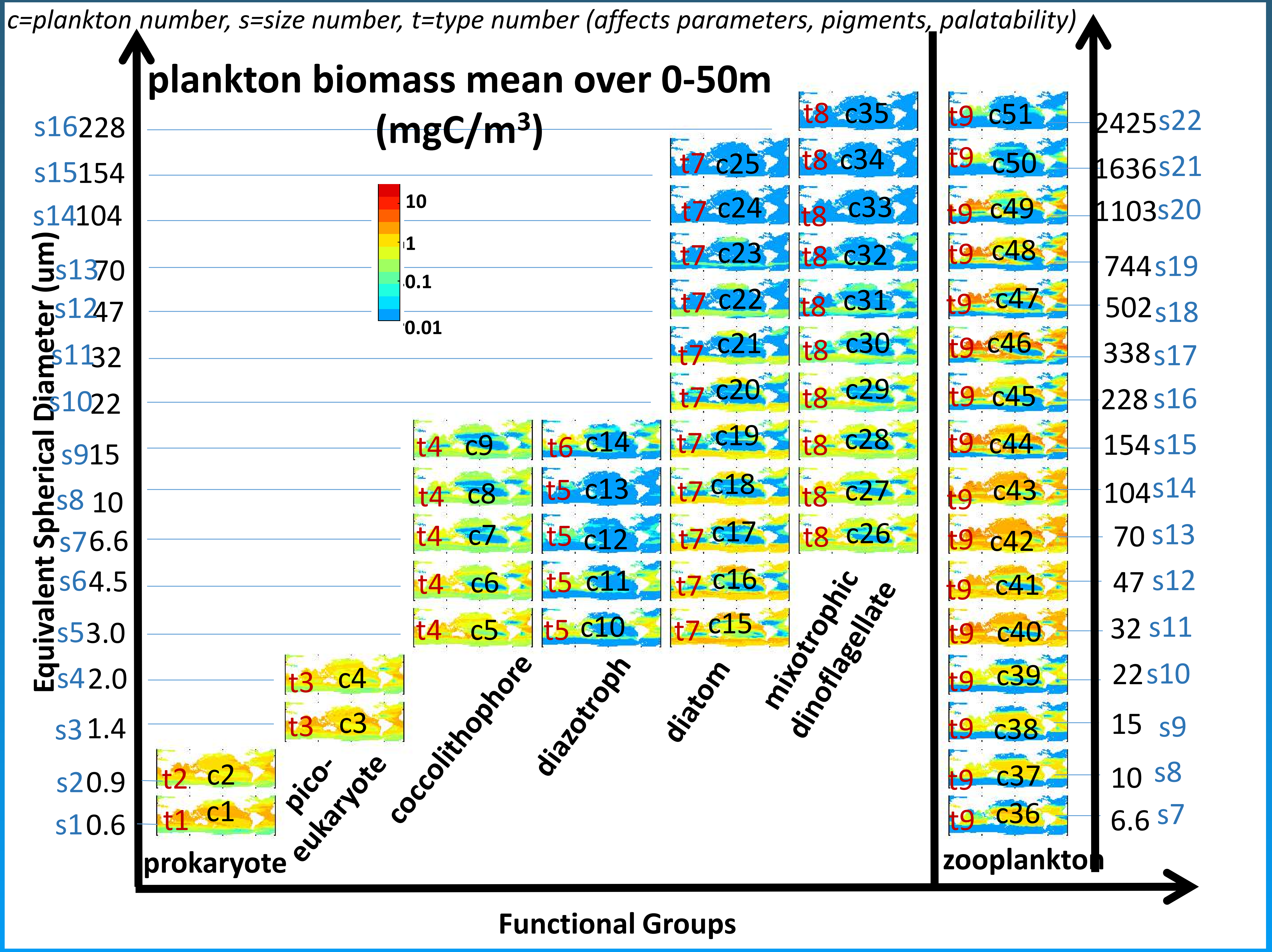
Here, you will learn about simple methods that are available to download, visualize, and compute *CBIOMES-global* solutions. The prototype solution, *CBIOMES-global (alpha version)*, is a global ocean state estimate that covers the period from 1992 to 2011. It is based on [FCH+15] for ocean physics and [DHJ+15] for marine biogeochemistry and ecosystems.

## Contents:



**2018 : CBIOMES-global**

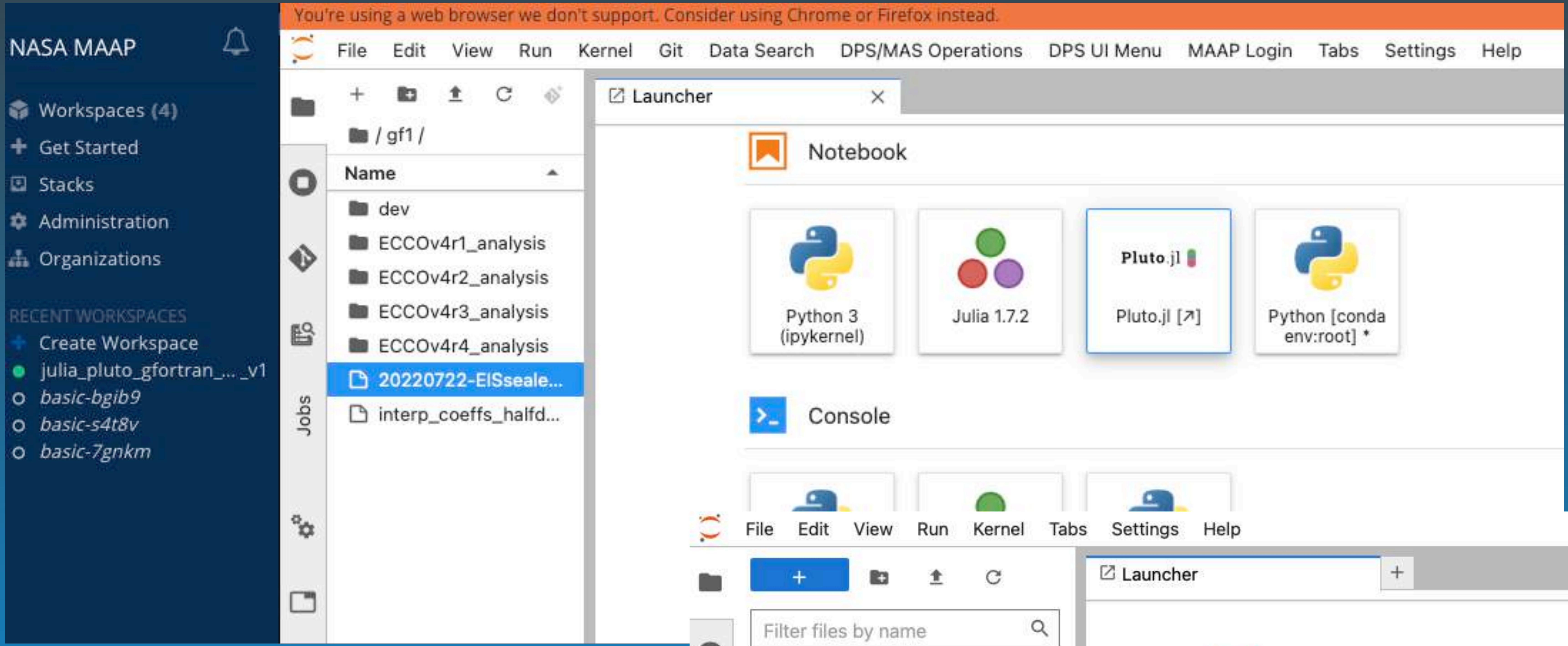
- ECCO4 (release 2)
- Darwin3 (Dutkiewicz et al 2015)

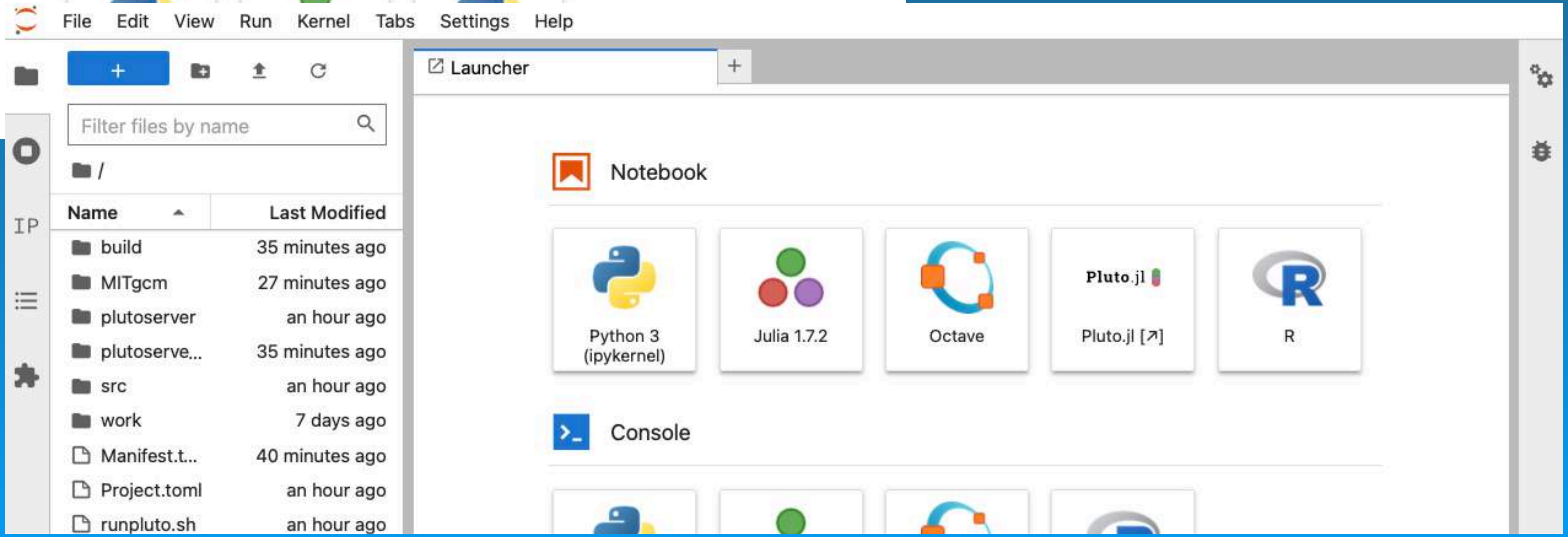


# ECCO cyber-infrastructure (2)

**2022 : ECCO-Docker**

- run & analyze ECCO anywhere, incl. NASA MAAP/EIS cloud
- Multi-language support (fortran, mpi, Julia, python, octave, R)




**docker run -p 8888:8888 gaelforget/ecco-docker**

# Why We Use Julia, 10 Years Later

14 February 2022 | **The Julia Community**

Exactly ten years ago today, we published "Why We Created Julia", introducing the Julia project to the world. At this point, we have moved well past the ambitious goals set out in the original blog post. Julia is now used by hundreds of thousands of people. It is taught at hundreds of universities and entire companies are being formed that build their software stacks on Julia.

**2022**

# JuliaEO23

- Julia for Beginners L. Alonso & S. Danisch
- Datacubes for High Resolution EO Data F. Cremer
- Data Visualizations with Makie.jl L. Alonso & S. Danisch
- Data Science, Big Data, and Cloud Native Solutions F. Gans & M. Klöwer
- Advanced Geodata Science & Geostatistical Learning J. Hoffimann
- Julia Showcases in Oceanography A. Barth & G. Forget
- Land Cover Classification of Earth Observation Images R. Schouten
- Machine Learning with Geospatial Data, What Can Go Wrong J. Hoffimann
- MBON, Ocean Biomes Monitoring, and Marine Ecosystem Modeling F. Muller Karger, A. Lima & G. Forget
- Raster Data Reading, Manipulating, and Visualising R. Schouten
- Julia Use Case for Change Detection F. Cremer
- Regional Ocean Data G. Forget, A. Valente & E. Castanho
- Processing LiDAR elevation point clouds using vector data in Julia M. Pronk
- Retrieving Satellite and Reanalysis Data from EO Servers N. Wong & A. Barth
- SAR Data Manipulation F. Cremer
- SARProcessing jl: Background and Method S. Lupemba & E. Lippert
- SARProcessing jl: Vision, State of Affairs, and Roadmap E. Lippert, S. Lupemba, K. Sorensen, & I. Szczesniak
- Simulating Oceanic Pathways of Plastics, Pollutants, or Marine Ecosystems G. Forget
- The Power of JuliaGeo M. Visser
- Working with SAR and InSAR Data K. Sorensen & S. Lupemba

https://www.youtube.com/@AIRCentre/playlists

**JuliaCon 2023 overview talk**

# Software 2024

🔭 Open science projects that I have recently designed or contributed to are listed below (*unfold each category for*

▼ Data (Julia)

- Dataverse.jl 📖 : interfaces to Dataverse APIs, collections, datasets, etc stars 7
- OceanRobots.jl 📖 : simulation and analysis of data generated by ocean robots stars 19
- ArgoData.jl 📖 : Argo data processing and analysis stars 15
- OceanColorData.jl 📖 : Ocean color data processing and analysis stars 9
- MarineEcosystemNotebooks : marine ecosystem observations and models stars 12
- OceanStateEstimation.jl 📖 : downloading, reading, displaying, and analyzing ocean state estimates stars 11

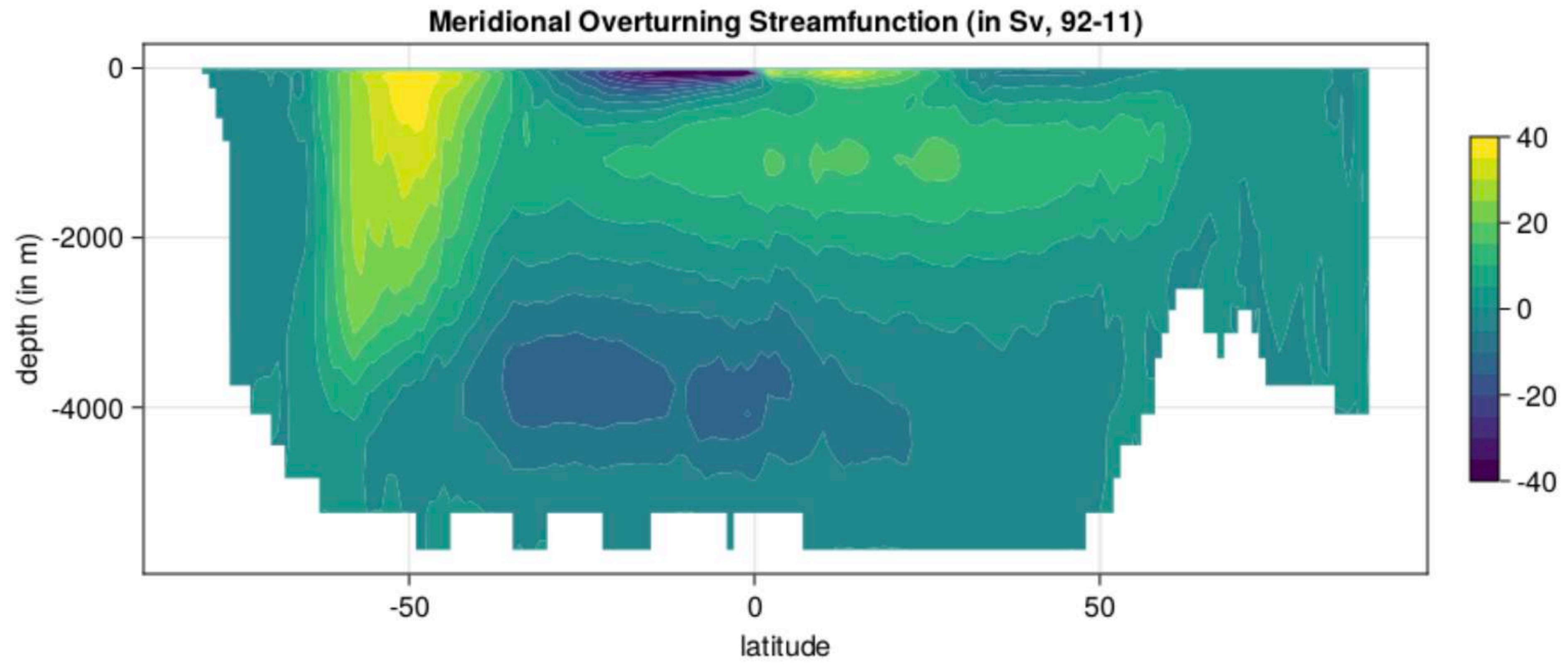
▼ Modeling (Julia)

- ClimateModels.jl 📖 : uniform interface to climate models of varying complexity and completeness stars 39
- MITgcm.jl 📖 : framework to interact with MITgcm (setup, run, output, plot, etc) stars 19
- MeshArrays.jl 📖 : gridded Earth variables, domain decomposition, and C-grid support stars 41
- IndividualDisplacements.jl 📖 : trajectory simulations for point particles in Ocean, Atmosphere, etc stars 31
- PlanktonIndividuals.jl 📖 : simulate the behaviors of an ensemble of phytoplankton individuals stars 28

# OceanStateEstimation.jl

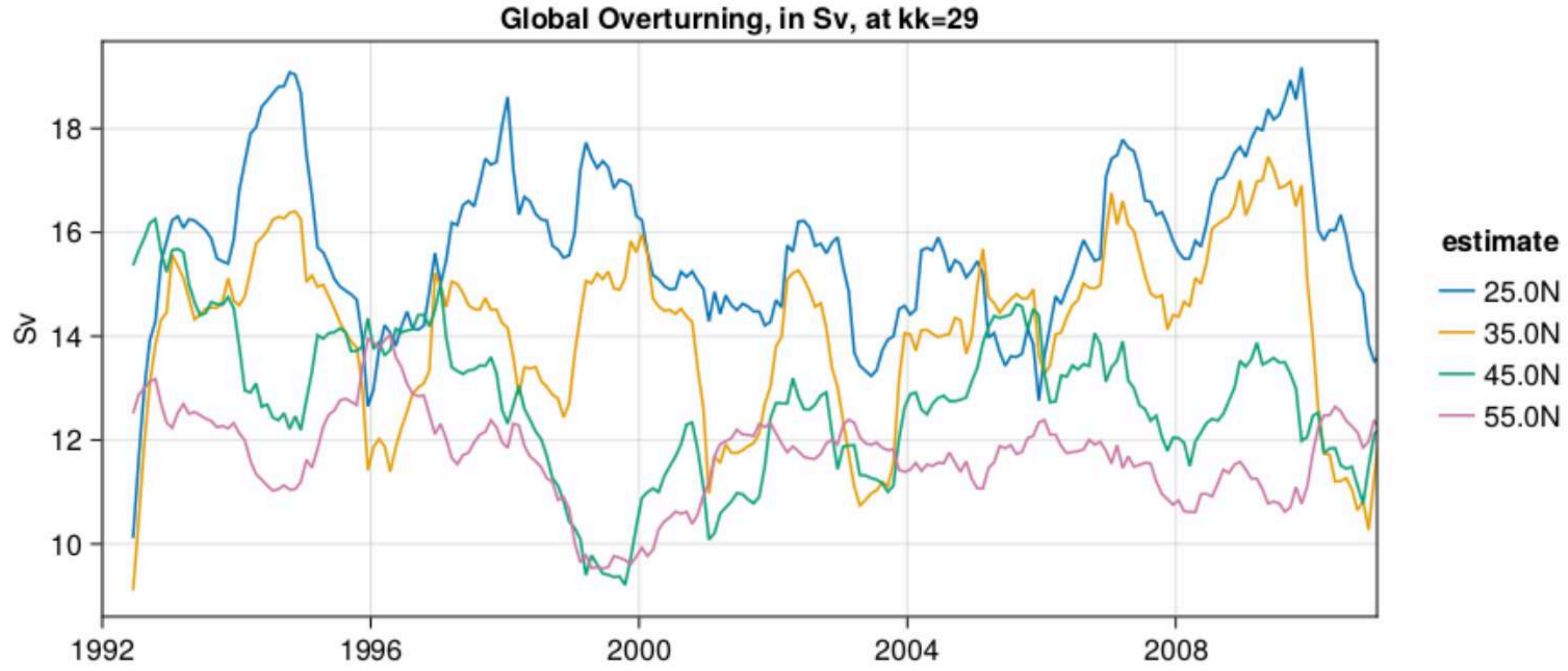
## Overturning Streamfunction

Edit or run this notebook




Save Plot for map in ECCOv4r2_analysis

## Overturning Time Series




### Table of Contents



The computation loop, over all months, can then be carried out as follows.

`OceanStateEstimation.ECCO_diagnostics.driver` — Function

```
driver(P)
```

Call main computation loop as specified by parameters `P`.

The main computation loop choice depends on the `P` parameter values. Methods include:

- `main_clim`
- `main_glo`
- `main_zonmean`
- `main_overturn`
- `main_MHT`
- `main_trsp`

source

# MITgcm.jl

**MITgcm.jl**

Search docs (Ctrl + /)

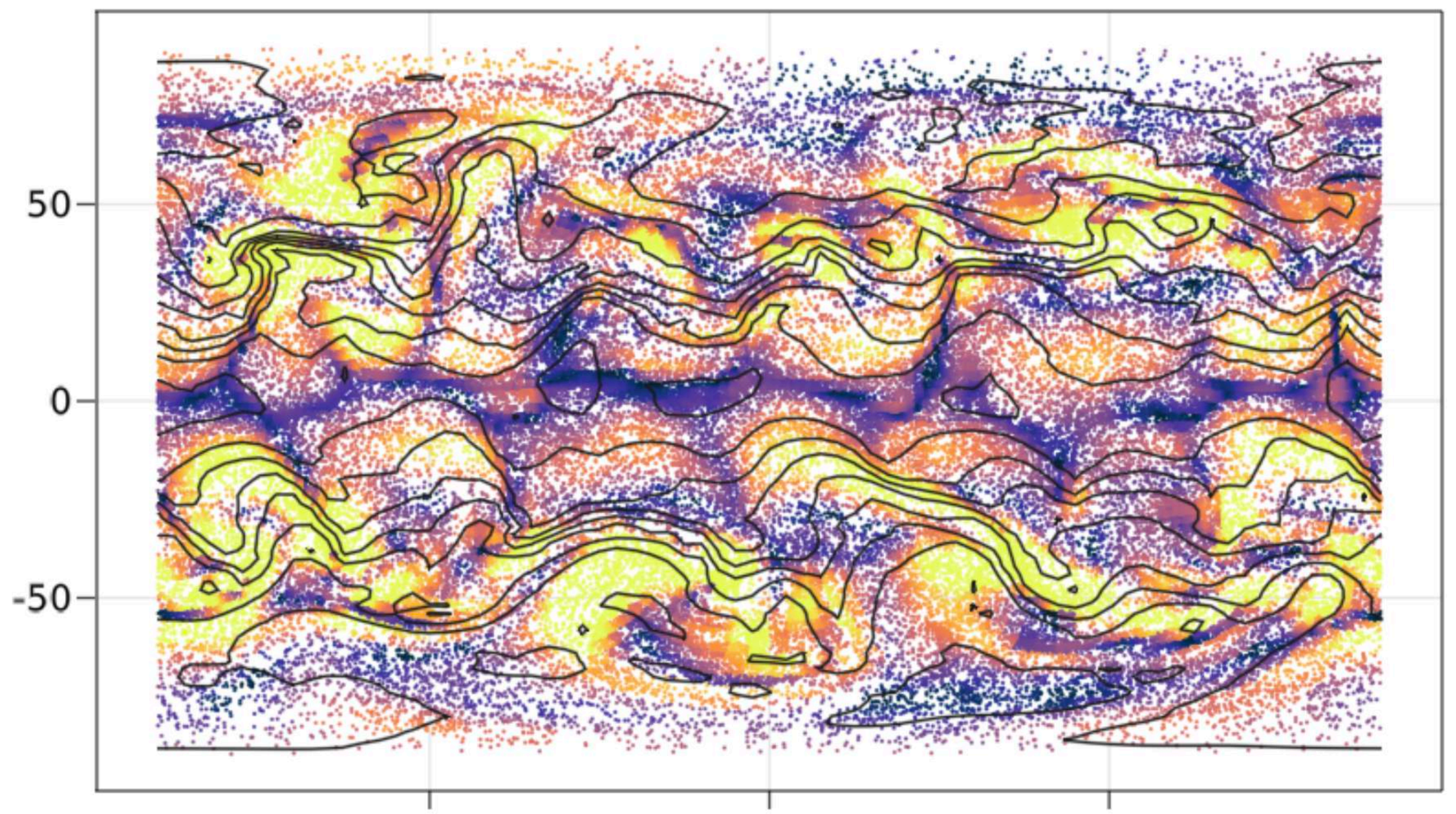
**Home**

- Main Features
- Table Of Contents

**Manual**

**Examples**

**API Reference**




## Run Model

Edit or run this notebook

The main model computation takes place via the `launch` function.

Once done, the next cell will display the model standard text file called `output.txt`

```
"Done with launch"

▾String[
  1: "(PID.TID 0000.0001) "
  2: "(PID.TID 0000.0001) // ======================================================="
  3: "(PID.TID 0000.0001) //                      MITgcm UV"
  4: "(PID.TID 0000.0001) //                      ========="
  5: "(PID.TID 0000.0001) // ======================================================="
  6: "(PID.TID 0000.0001) // execution environment starting up..."
  7: "(PID.TID 0000.0001) "
  8: "(PID.TID 0000.0001) // MITgcmUV version:  checkpoint68d"
  9: "(PID.TID 0000.0001) // Build user:        runner"
```

Table of Contents



- Scan Run Folder
- MDS Files (binary)
- MNC Files (netcdf)
- Grid Files (binary or netcdf)
- Standard Output (text)
- Input Files (text)
- Other Files

# ClimateModels.jl

ClimateModels.jl

Search docs (Ctrl + /)

**Home**

**User Manual**

**Examples**

- Workflows That Run Models
- Workflows That Replay Models
- JuliaCon 2021 Presentation
- Trying Out The Examples

**API reference**

## Workflows That Run Models

- Random Walk model (Julia) ➪ code link
- ShallowWaters.jl model (Julia) ➪ code link
- Oceananigans.jl model (Julia) ➪ code link
- Hector global climate model (C++) ➪ code link
- FaIR global climate model (Python) ➪ code link
- SPEEDY atmosphere model (Fortran90) ➪ code link
- MITgcm general circulation model (Fortran) ➪ code link

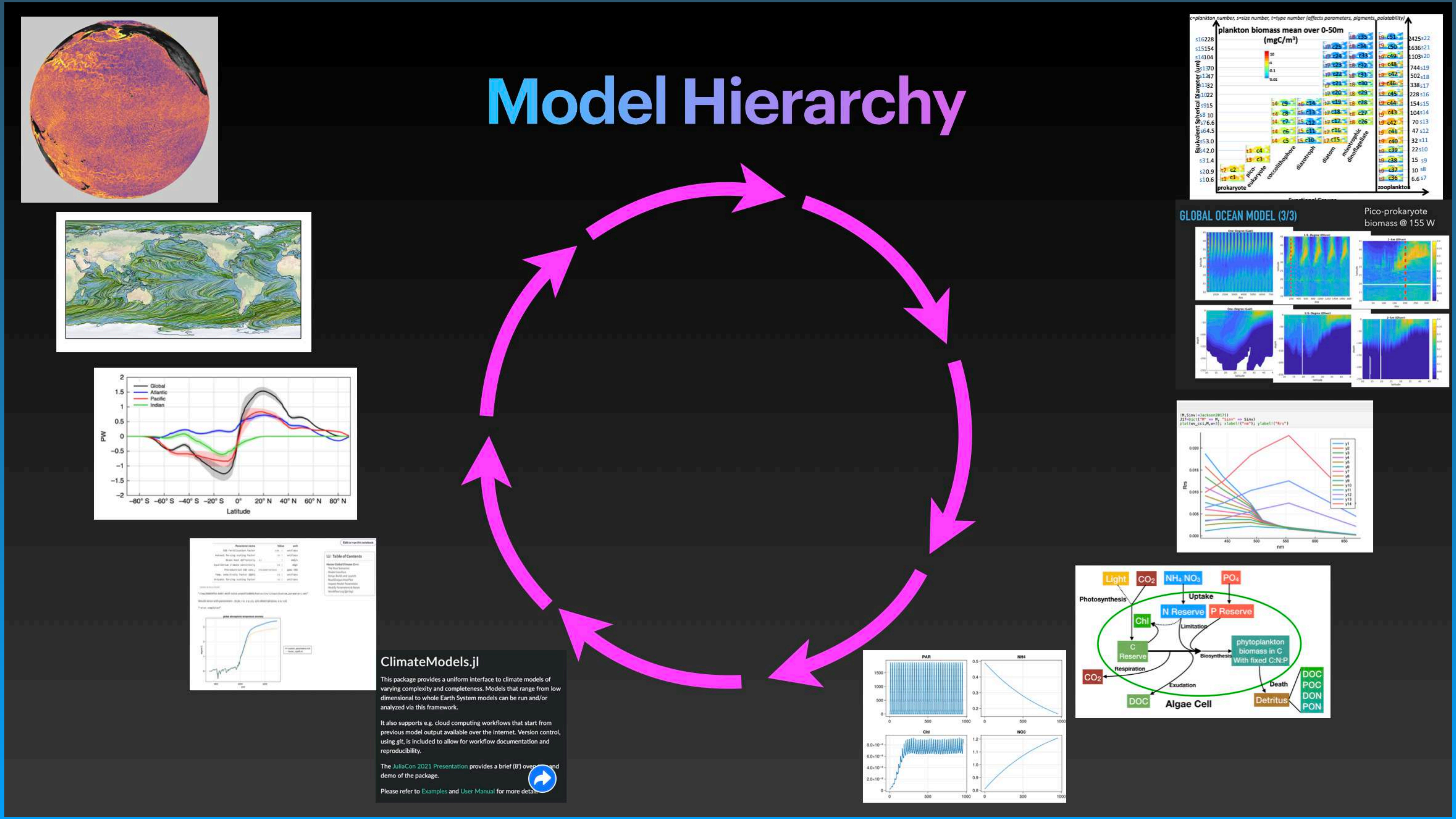
## Workflows That Replay Models

- IPCC report 2021 (NetCDF, CSV) ➪ code link
- CMIP6 model output (Zarr) ➪ code link
- ECMWF IFS 1km (NetCDF) ➪ code link
- ECCO version 4 (NetCDF) ➪ code link
- Pathway Simulations (binary, jld2) ➪ code link

JuliaCon 2021 Presentation (8')

JuliaCon 2023 Presentation (20')

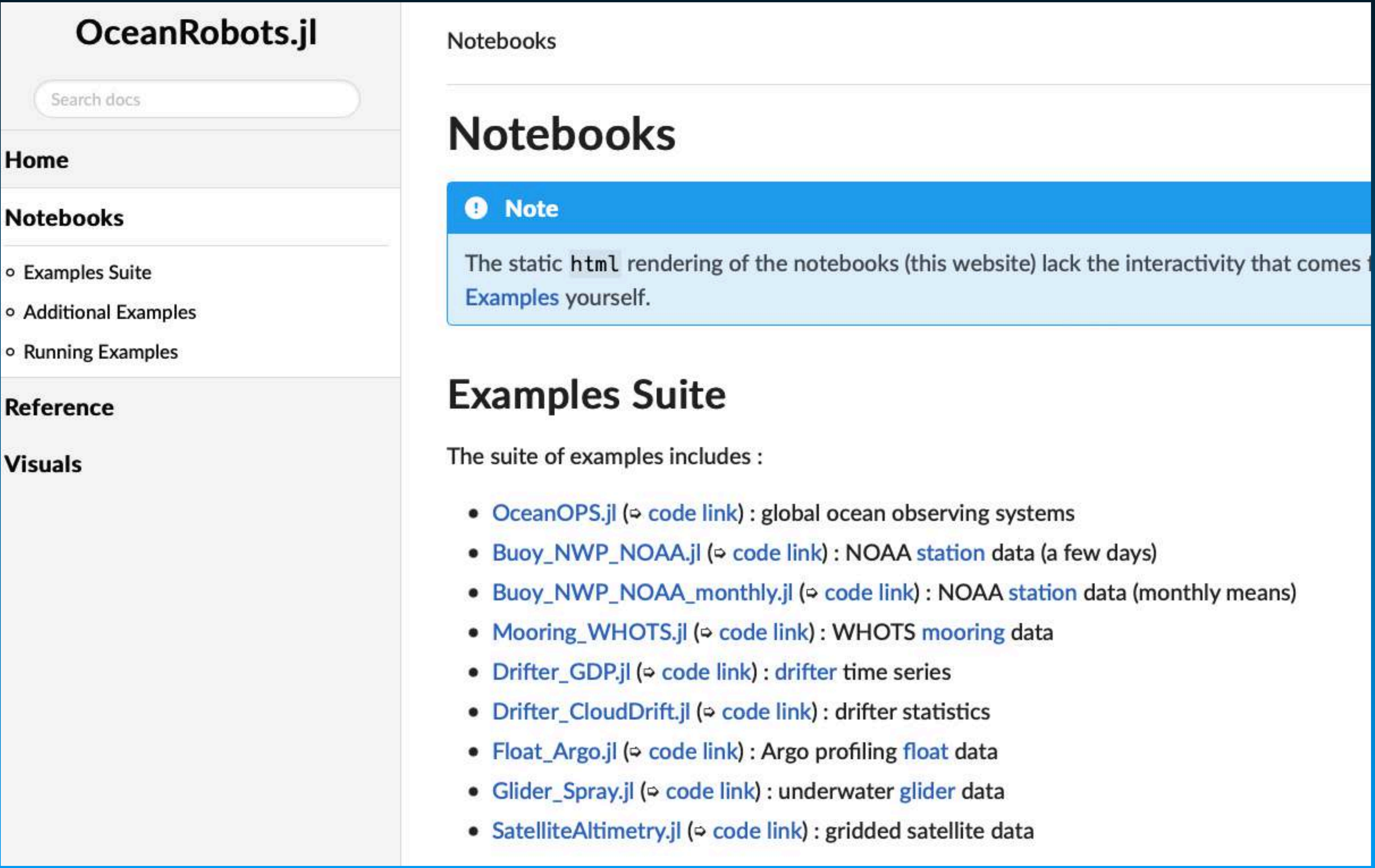
# Digital Twins for Ocean Robots and Marine Ecosystems




## US CLIVAR webinar

## JuliaCon 2023 talk

# OceanRobots.jl & ArgoData.jl



OceanRobots.jl

Search docs

**Home**

**Notebooks**

- Examples Suite
- Additional Examples
- Running Examples

**Reference**

**Visuals**

Notebooks

# Notebooks

**Note**

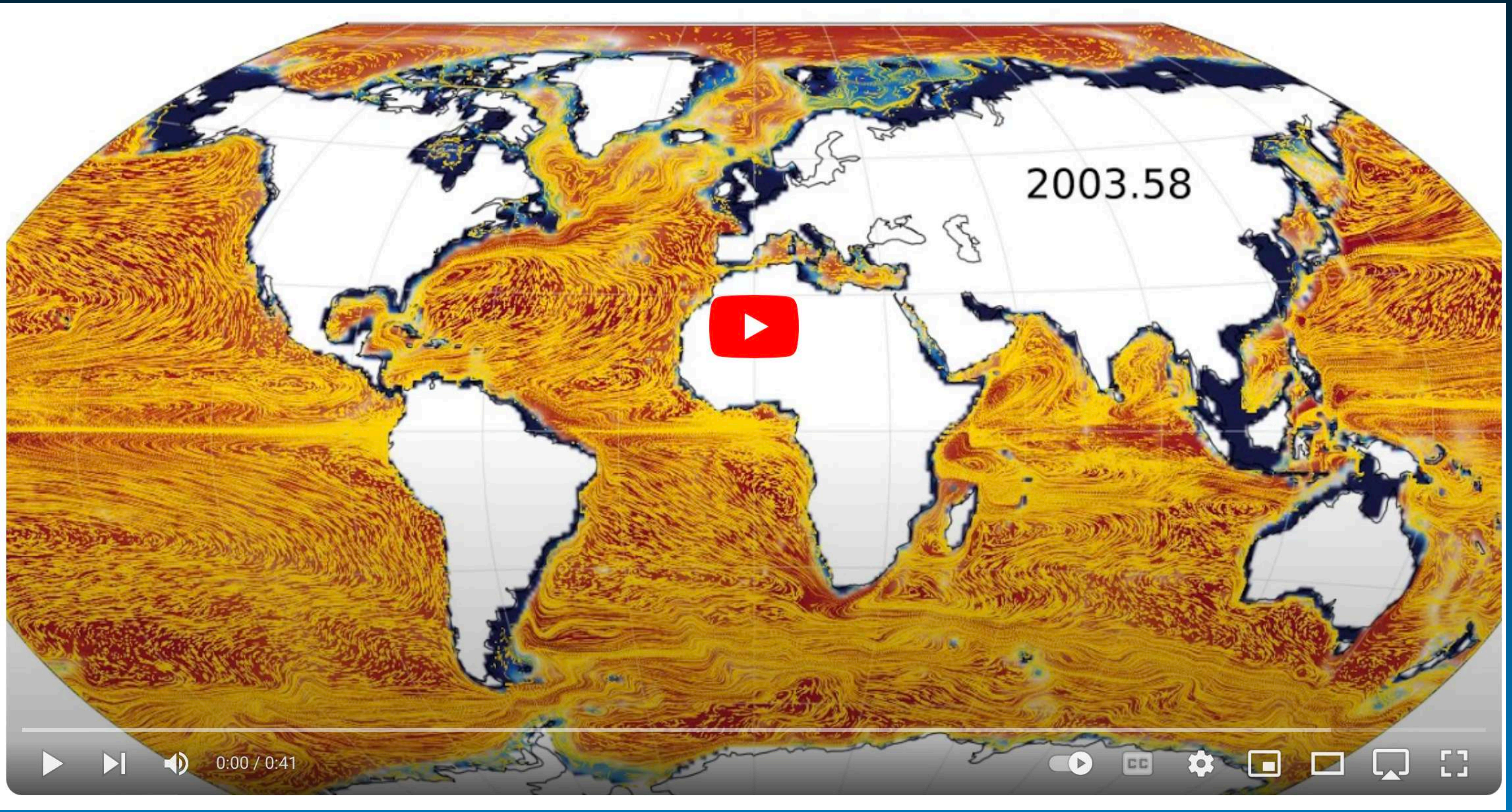
The static `html` rendering of the notebooks (this website) lack the interactivity that comes
Examples yourself.

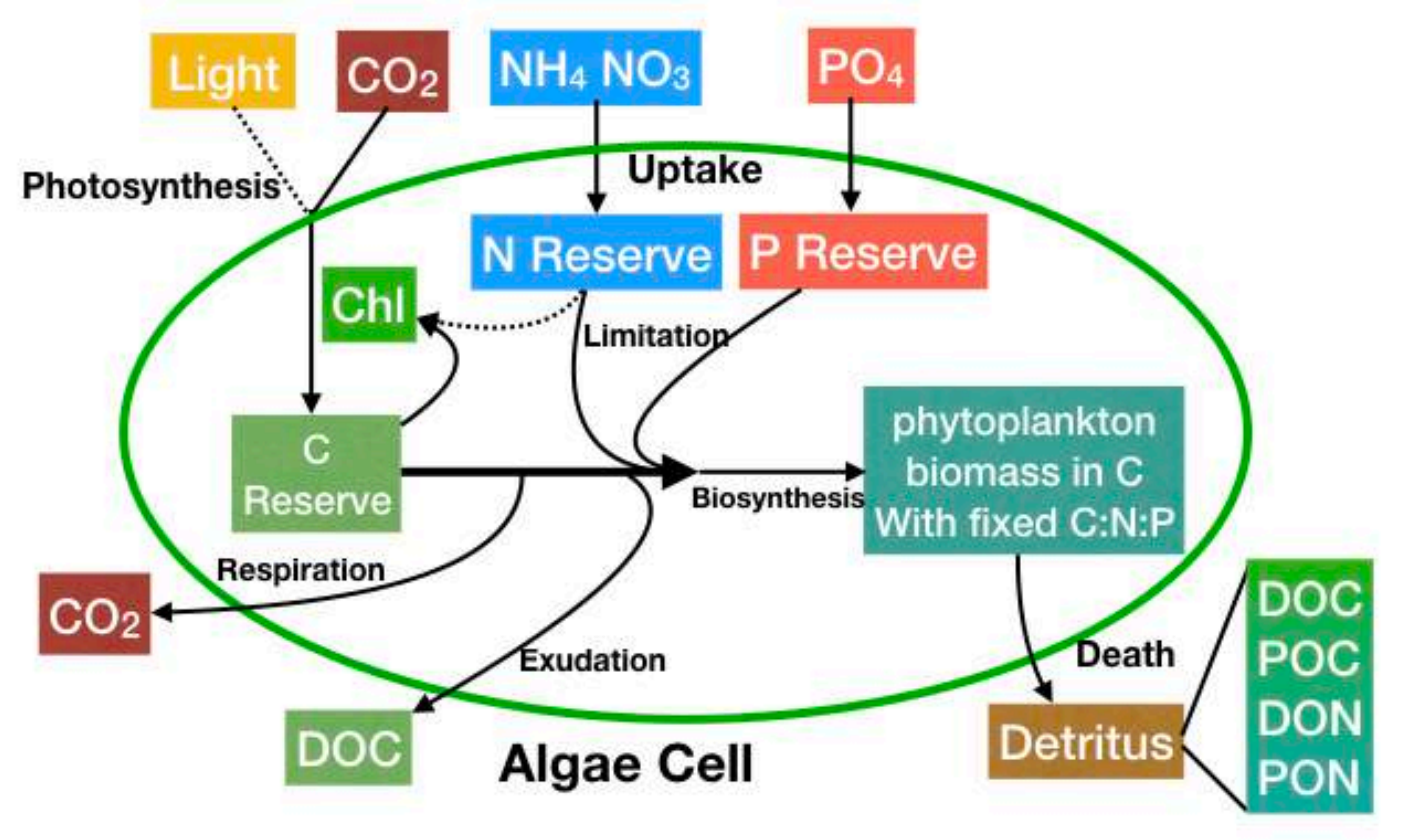
## Examples Suite

The suite of examples includes :

- OceanOPS.jl (➪ code link) : global ocean observing systems
- Buoy_NWP_NOAA.jl (➪ code link) : NOAA station data (a few days)
- Buoy_NWP_NOAA_monthly.jl (➪ code link) : NOAA station data (monthly means)
- Mooring_WHOTS.jl (➪ code link) : WHOTS mooring data
- Drifter_GDP.jl (➪ code link) : drifter time series
- Drifter_CloudDrift.jl (➪ code link) : drifter statistics
- Float_Argo.jl (➪ code link) : Argo profiling float data
- Glider_Spray.jl (➪ code link) : underwater glider data
- SatelliteAltimetry.jl (➪ code link) : gridded satellite data

# IndividualDisplacements.jl & PlanktonIndividuals.jl

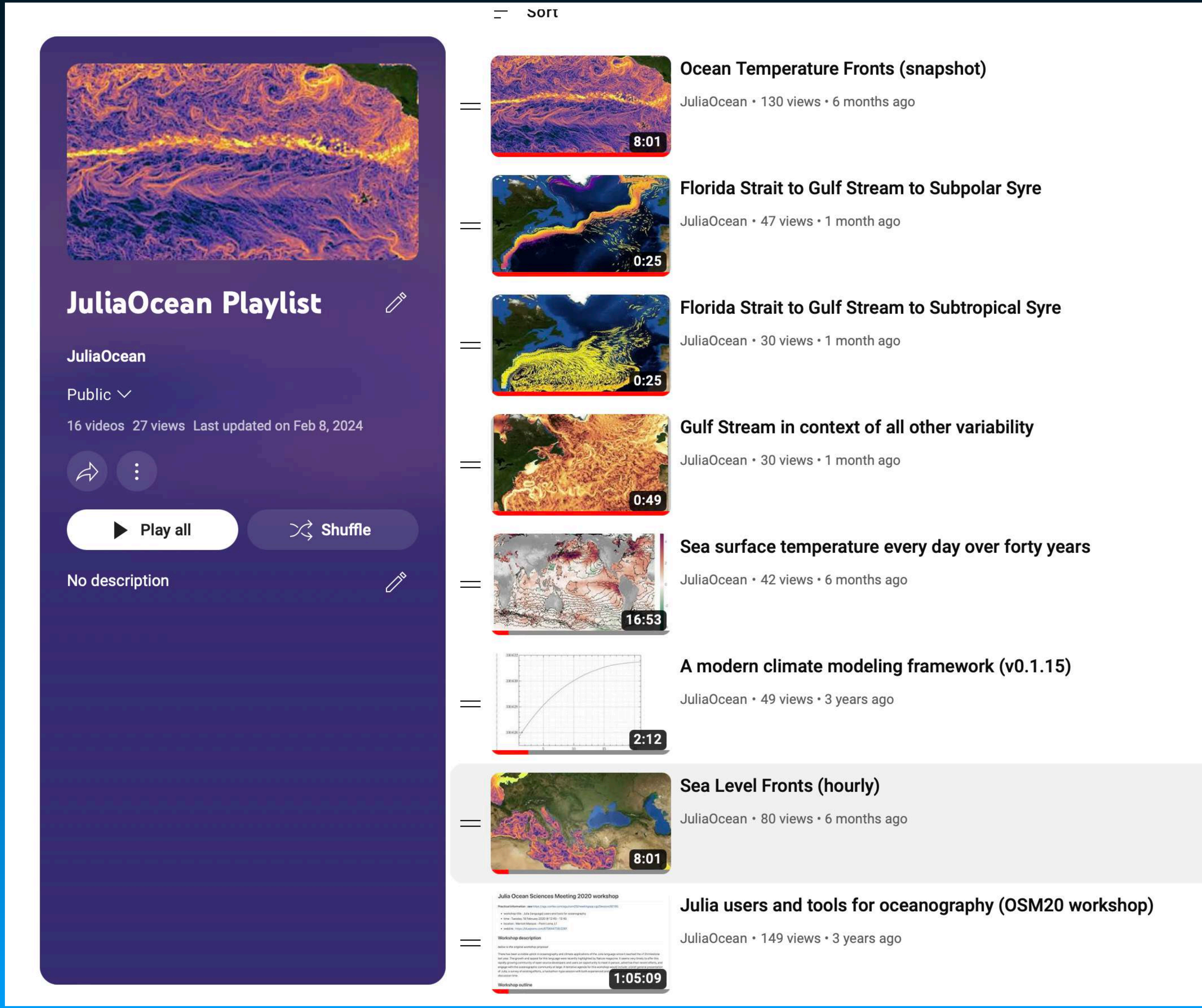



Forget, G., (2021). IndividualDisplacements.jl: a Julia package to simulate and study particle displacements within the climate system. Journal of Open Source Software, 6(60), 2813, https://doi.org/10.21105/joss.02813

Wu & Forget (2022). PlanktonIndividuals.jl: A GPU supported individual-based phytoplankton life cycle model. Journal of Open Source Software, 7(73), 4207, https://doi.org/10.21105/joss.04207

# Animations




## JuliaOcean playlist