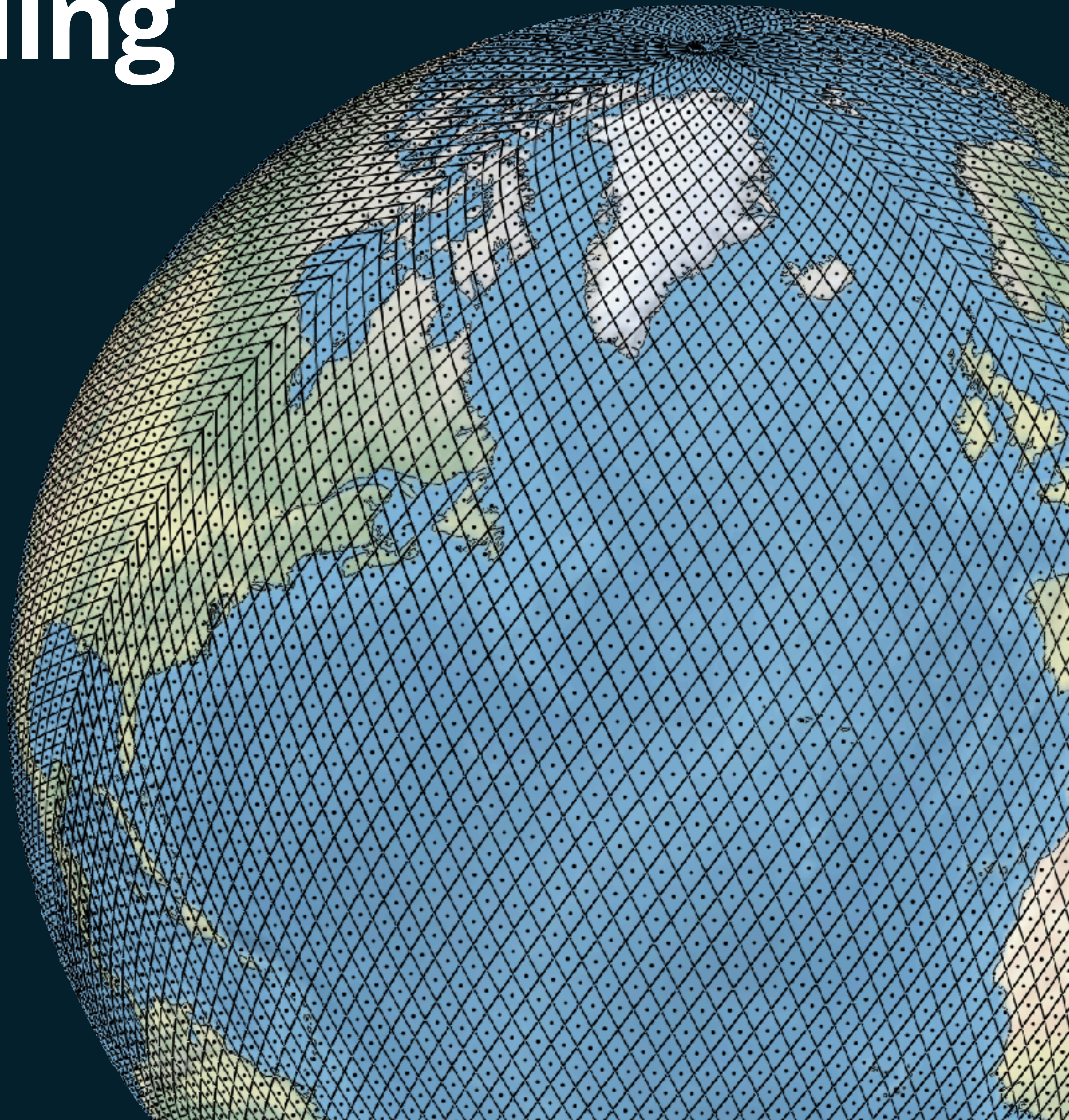
Interactive climate modelling

An introduction to SpeedyWeather

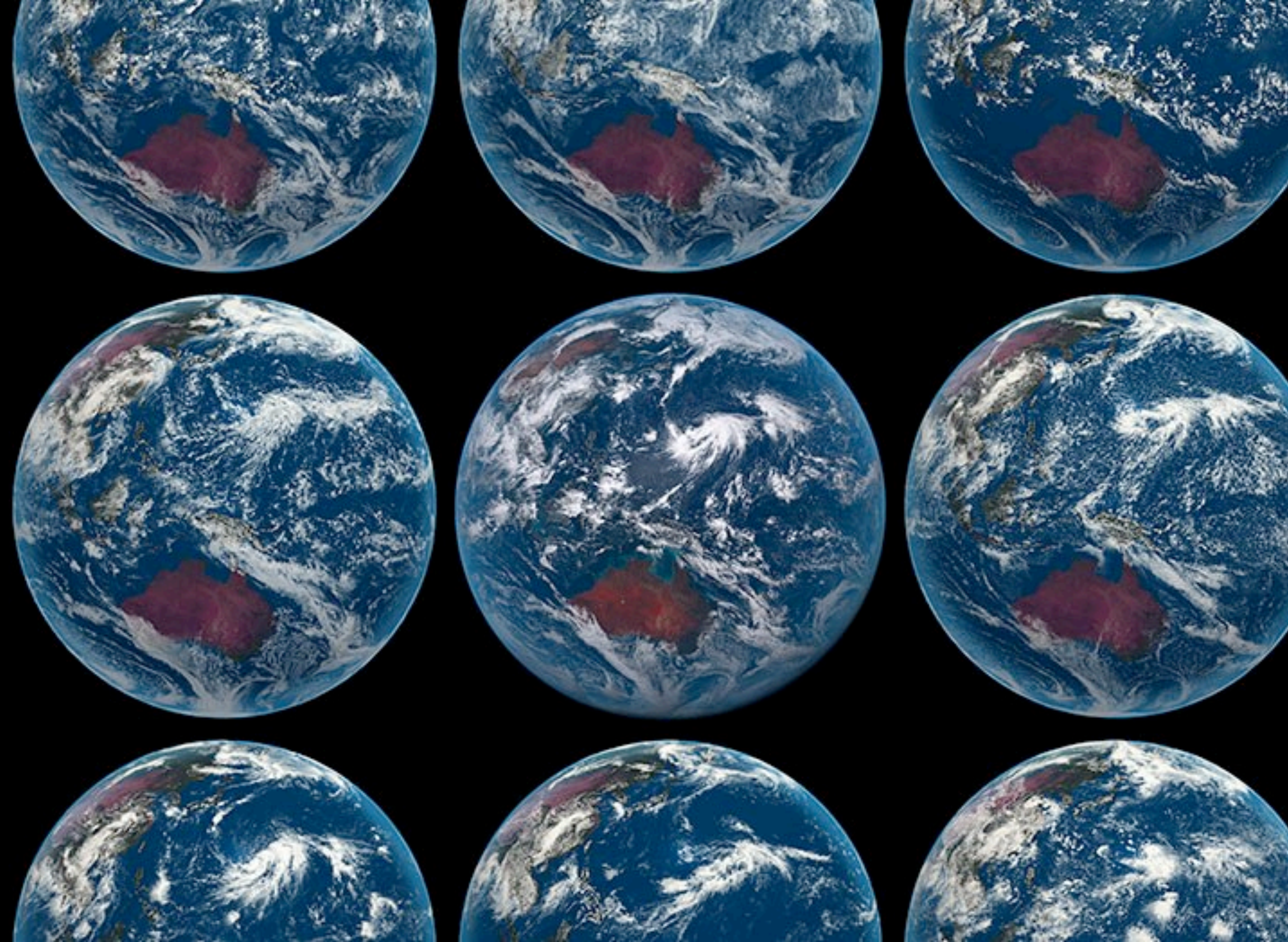


Session 3: Earth-system modelling
Edinburgh, Feb 2026

Milan Klöwer and **many colleagues**
University of Oxford

Atmospheric models

A silent revolution



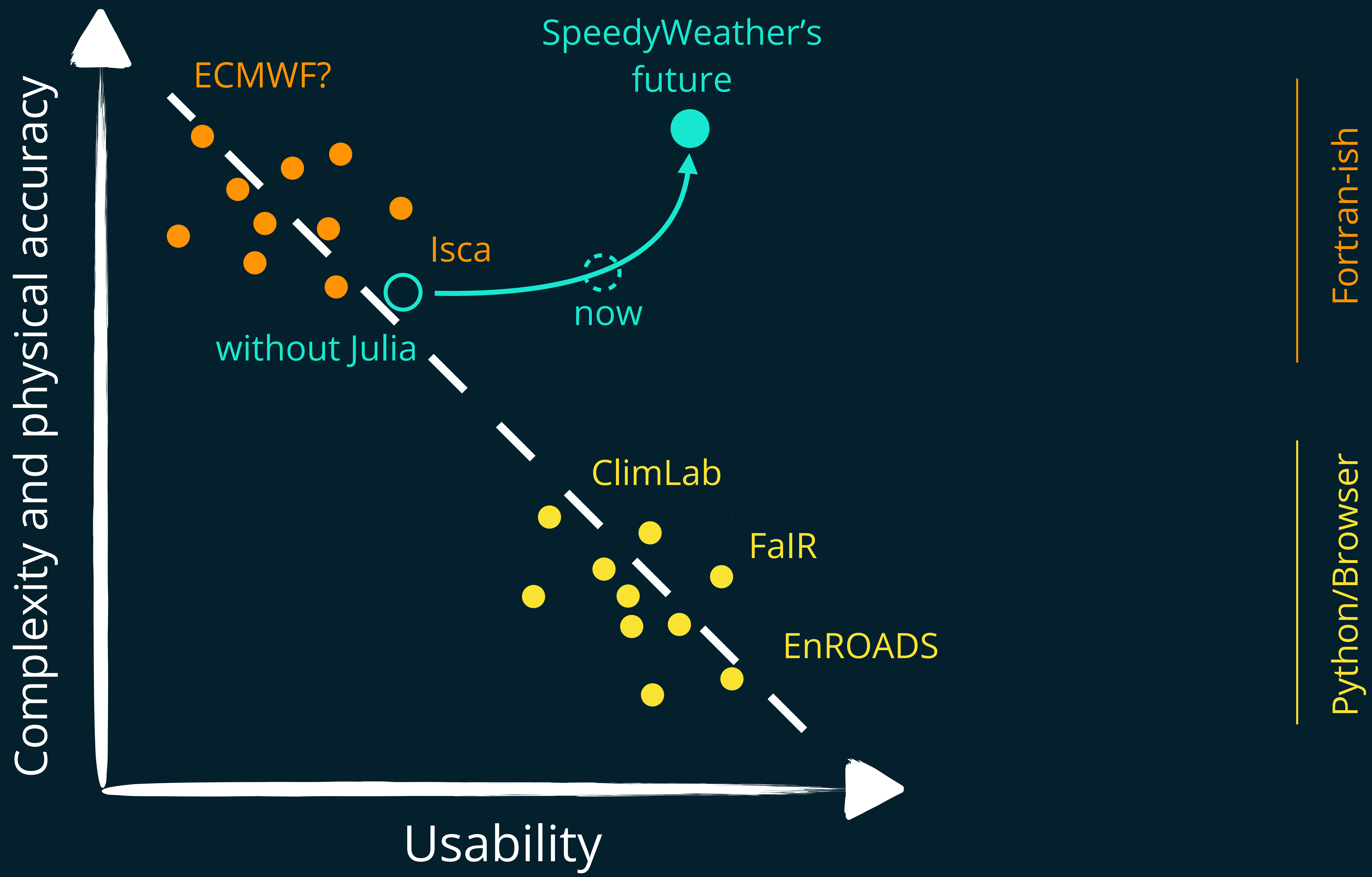
Stevens et al., 2021

A vision:

Design a research project on Monday
Implement it on Tuesday
Run simulations on Wednesday
Analyse and visualise them on Thursday
Start writing the paper on Friday

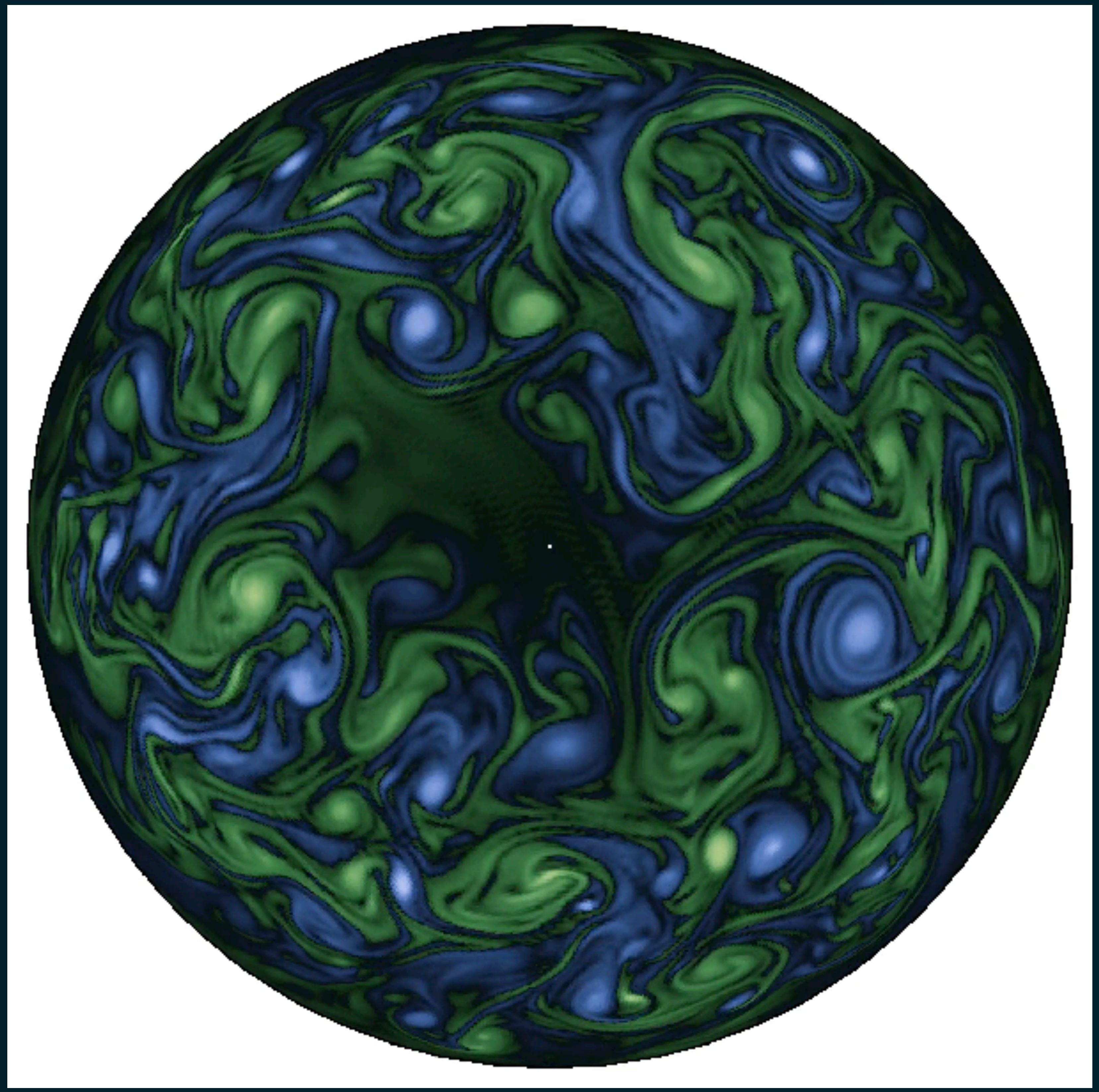
Why is this currently not possible?

Two “language”-problem of weather and climate models

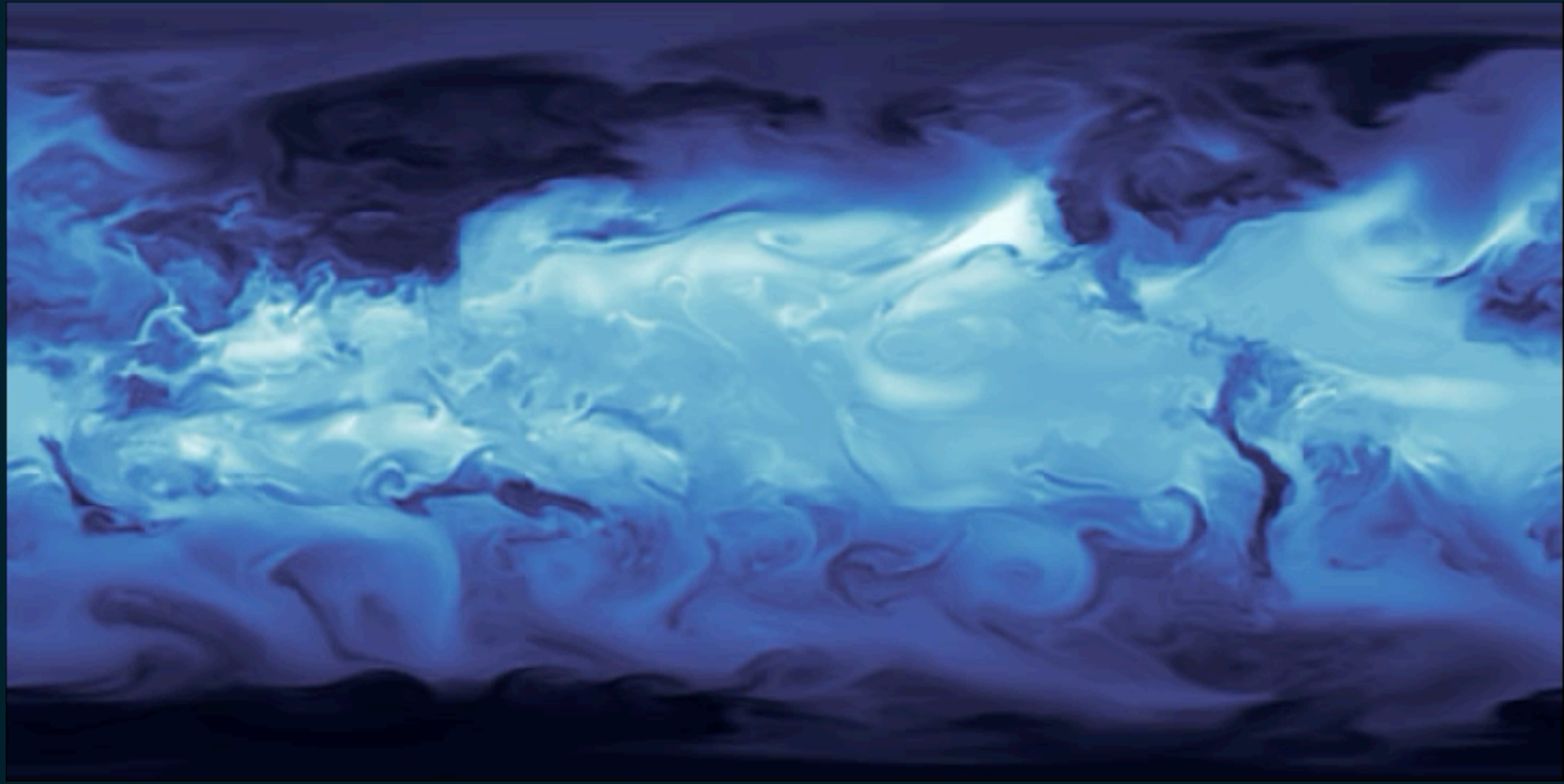


SpeedyWeather: Play atmospheric modelling like it's LEGO

From idealised 2D turbulence to 3D atmosphere-climate simulations

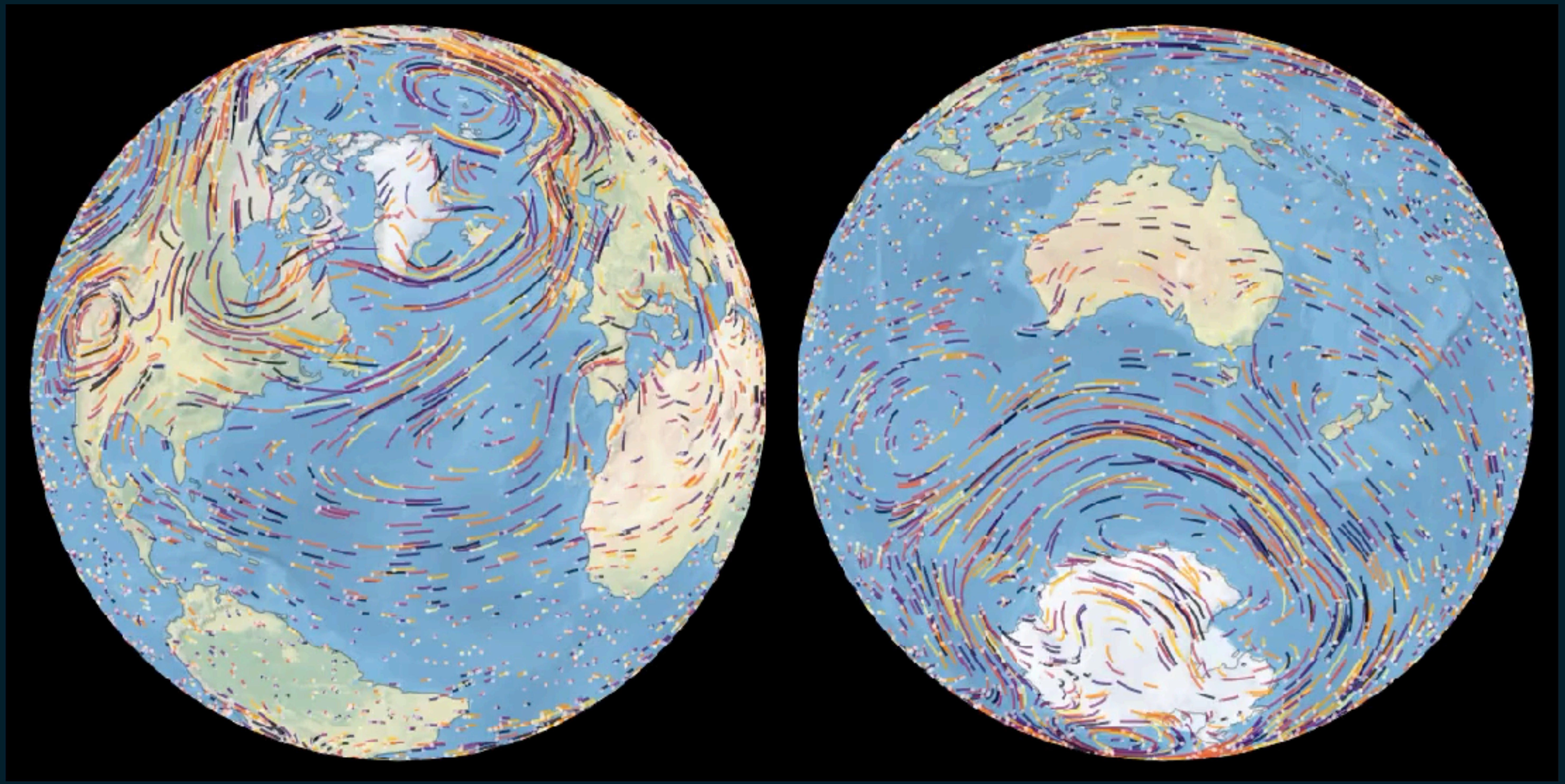


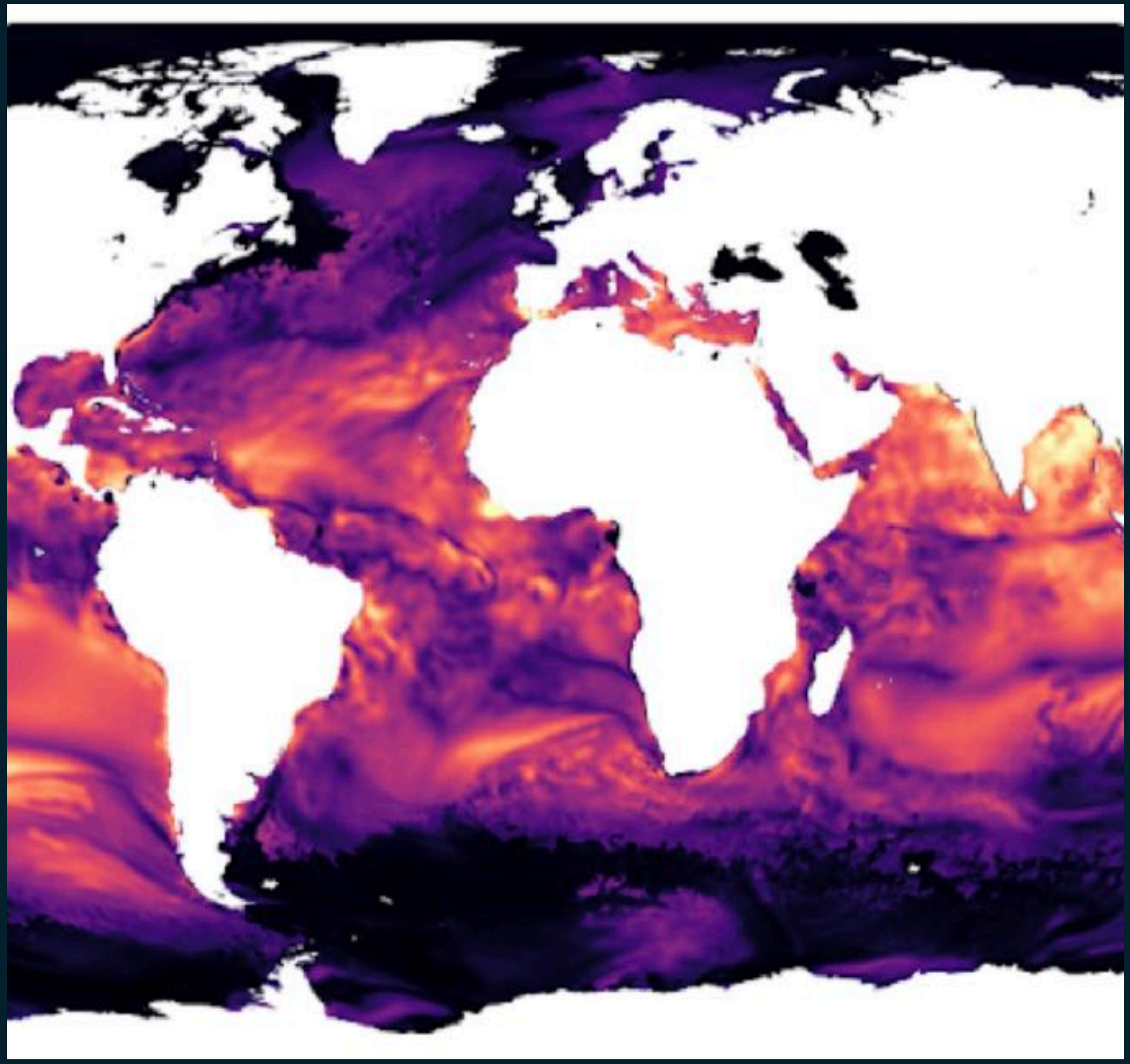
Humidity at 50km resolution



Pier Luigi Vidale: "*unthinkable on such a small laptop … like playing a video game*"

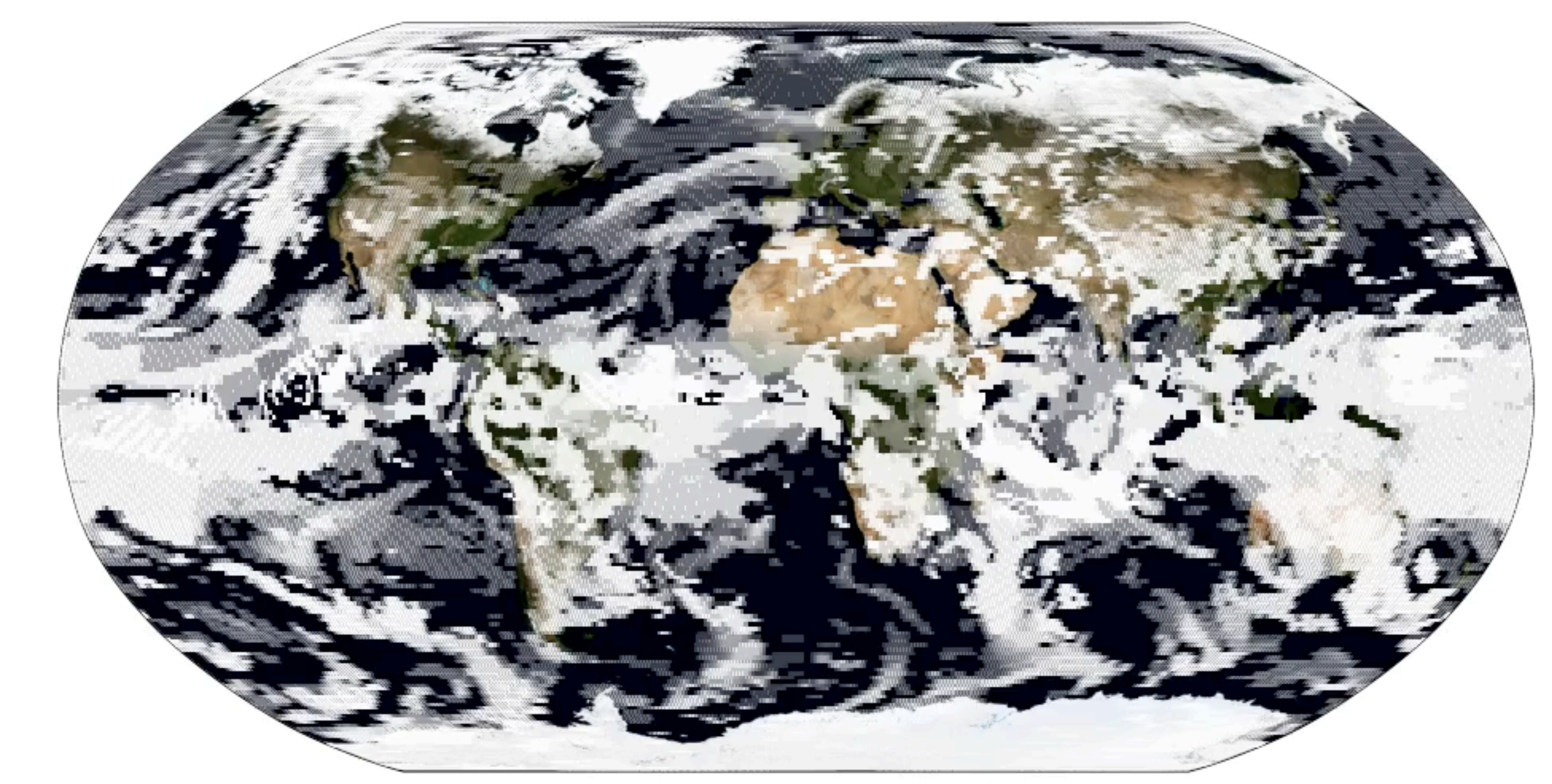
Particle advection





Heat flux when coupled to Oceananigans

Cloud cover



It’s complicated!

Coriolis

Gravity

Pressure gradient

Momentum advection

Viscous forces

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + 2\boldsymbol{\Omega}\times\mathbf{u}\right) = \rho\mathbf{g} - \nabla p + \mu\nabla^2\mathbf{u}$$

Conservation of mass (compressible)

$$\frac{\partial \rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0$$

Ideal gas law, equation of state

$$p = \rho R T$$

$$\frac{\partial T}{\partial t} + (\mathbf{u}\cdot\nabla)T - \frac{RT}{c_p}\frac{D\ln p}{Dt} = \frac{Q}{c_p}$$

Heat sources

Temperature advection

Adiabatic conversion

What’s missing? Humidity, clouds, precipitation, snow, radiation, surface fluxes, ocean, land, …

Simpler: 2D barotropic vorticity

Momentum advection

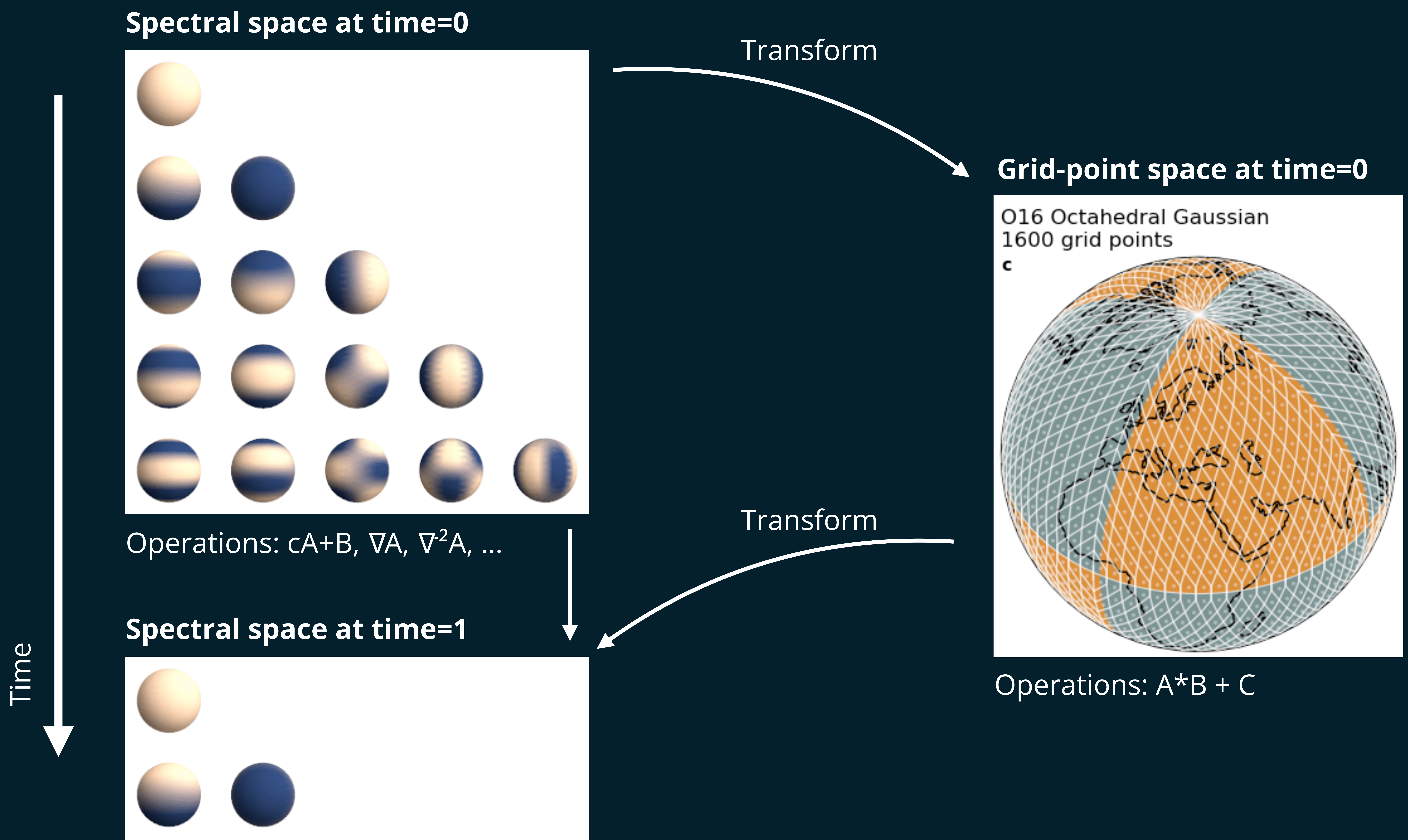
Forcing: energy in

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} + 2\boldsymbol{\Omega} \times \mathbf{u} = -\frac{1}{\rho}\nabla p + F + D$$

Coriolis

Dissipation: energy out

Simulates wind and vortices (chaos!) but no waves
(Rossby though!), temperature, humidity, ...



Spectral space at time=0
Transform
Grid-point space at time=0
O16 Octahedral Gaussian
1600 grid points
c
Operations: cA+B, ∇A, ∇⁻²A, ...
Transform
Operations: A*B + C
Spectral space at time=1
Time

Solving equations spectrally

Apply spectral transform S

- *As much as possible in spectral space*
- *Linear terms and especially gradient operators*
- *A*B of two spatially varying fields in grid*
- *Transforms are expensive, use sparingly*

$$\frac{\partial \zeta}{\partial t} = \underbrace{F_\zeta + \nabla \times (\mathbf{F}_\mathbf{u}}_{\text{Forcing/Drag}} + \underbrace{\mathbf{u}_\perp(\zeta}_{\text{Advection}} + \underbrace{f}_{\text{Coriolis}})) + \underbrace{\nu \nabla^{2n} \zeta}_{\text{Diffusion}}$$

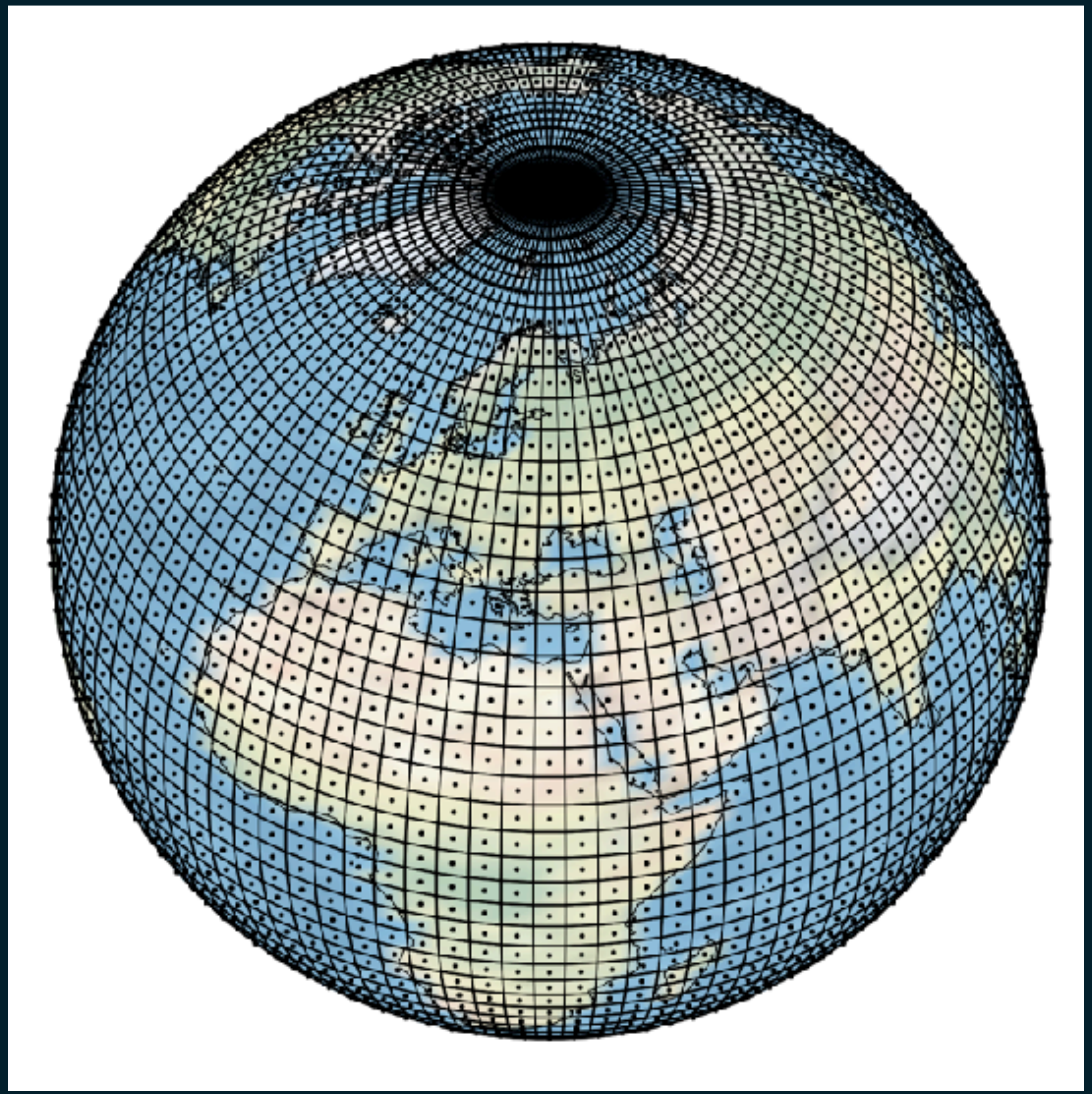
$$\underbrace{\frac{\partial \hat{\zeta}}{\partial t} = \hat{F}_\zeta + \nabla \times S}_{\text{Spectral}} \underbrace{\left(\mathbf{F}_\mathbf{u} + \mathbf{u}_\perp(\zeta + f)\right)}_{\text{Grid}} + \underbrace{\nu \nabla^{2n} \hat{\zeta}}_{\text{Spectral}}$$

Algorithm

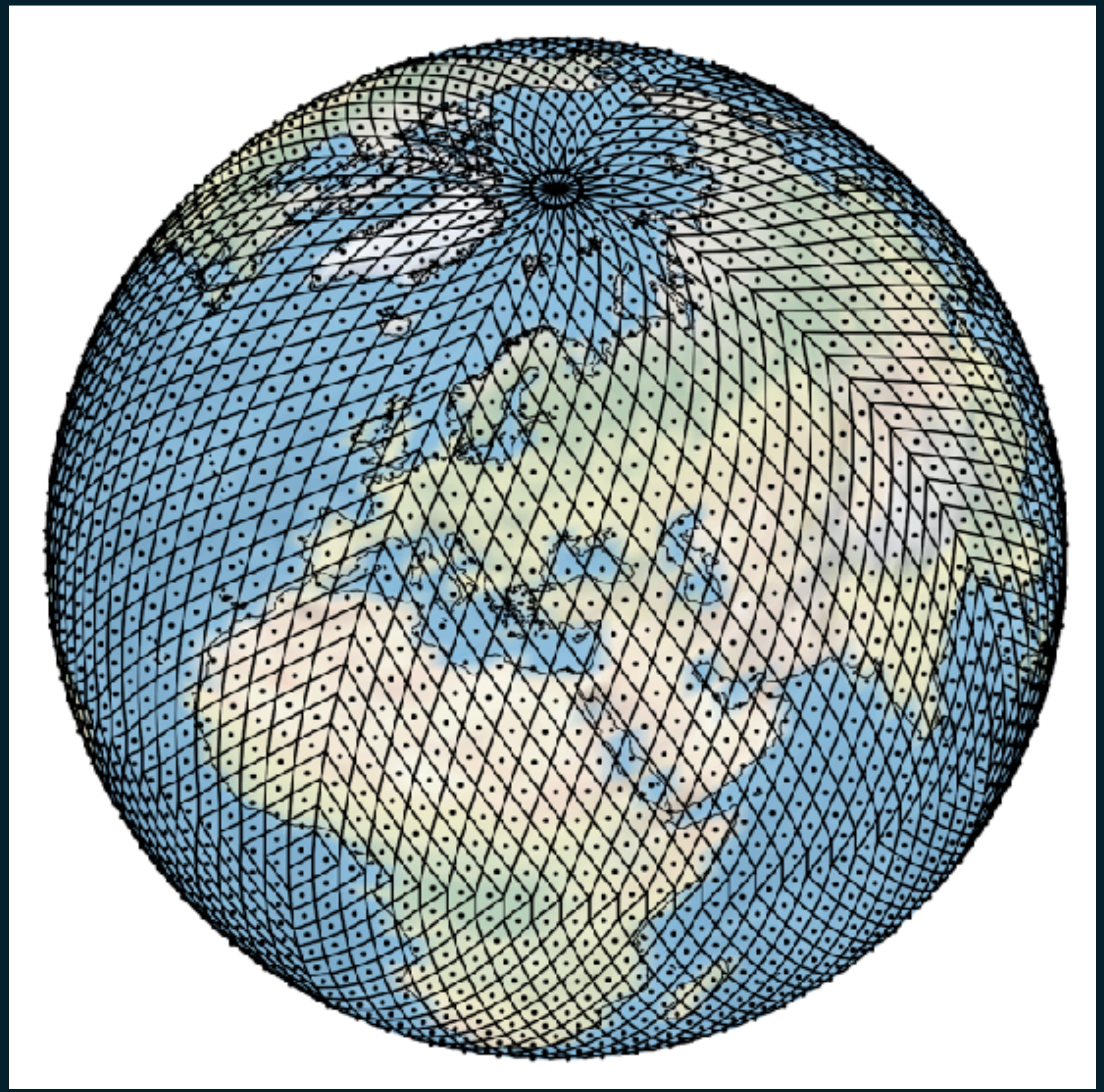
1. Start with $\hat{\zeta}$ in spectral space, inverse transform to ζ
2. Get u,v from $\hat{\zeta} = \nabla^{-2}\hat{\Psi}$ and $(\hat{u}, \hat{v}) = \nabla^{\perp}\hat{\Psi}$ via an inverse transform
3. Do in grid space $F_u + v(\zeta + f), \quad F_v - u(\zeta + f)$
4. Transform to spectral, apply curl, add other terms
5. Time step in spectral space $\hat{\zeta}_{i+1} = \hat{\zeta}_{i-1} + \Delta t \; d\zeta_i$
6. Repeat from 2

Grids

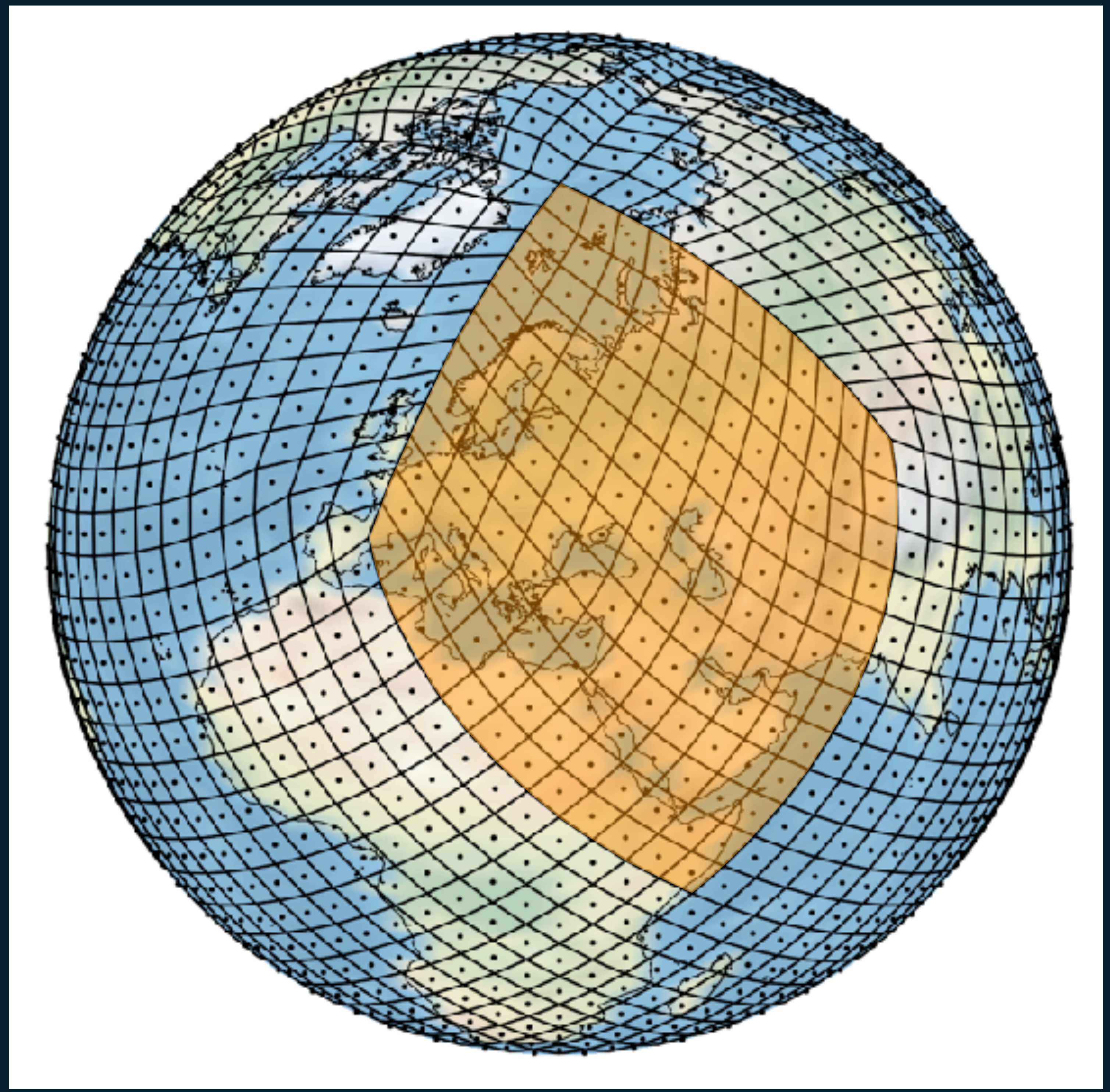
Suited for a spherical harmonic transform



A 96x48 regular longitude-latitude grid, 4608 grid points



ECMWF's octahedral Gaussian grid, 3168 grid points



HEALPix grid, equal-area grid, 1728 grid points

Requirements for *ring* grids
- iso-latitude (by Legendre polynomials)
- equi-longitude (FFT)

Live demo