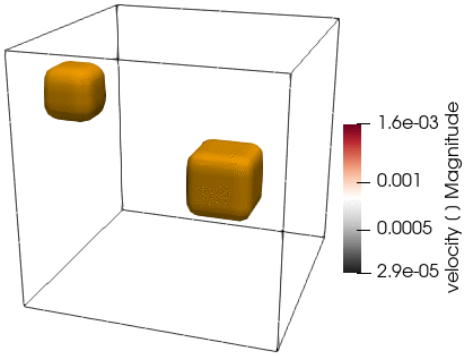
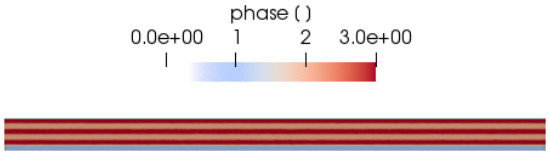
LaMEM short course

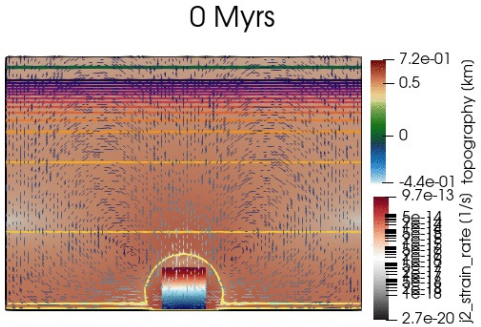
16-20 02 2026 Heidelberg

Nicolas Riel- nriel@uni-mainz.de

Objective of the course







- Learn how to use a geodynamic modeling code to simulate of wide range of 2D/3D scenario

- To do so, you will use
  - A geodynamic modeling code (LaMEM – 90%)
  - A landscape evolution code (FastScape – 10%)
  - A programming environment (Visual studio)
  - A programming language (Julia)
  - Visualization tools (Paraview, and PlotlyJS (a julia package))
  - A tool to generate density diagram used in LaMEM (MAGEMinApp)

Structure/Organization of the short course

Monday

Morning
- Introduction
- Installation
- First model

Afternoon
- Julia introduction
- Create first setup(s)

Tuesday

Morning
- Building up complexity

Afternoon
- Advanced setup: plume

Wednesday

Morning
- Fold and thrust

Afternoon
- Rifting
- GeophysicalModelGenerator

Thursday

Morning
- Landscape evolution modelling

Afternoon
- Advanced setup: subduction

Friday

Morning and afternoon
- build your own setup

Some tips (based on experience)

- Nothing we are going to do is “complicated” on its own
  - *You don’t need advanced mathematical background*
  - *Programming experience will help but is not necessary,*
  - *Copy and paste and modify as much as possible!*

- Complexity arises from chaining several simple tasks

- If you don’t do it yourself, you are not going to learn
  - *There is not way around having bugs and solving them*

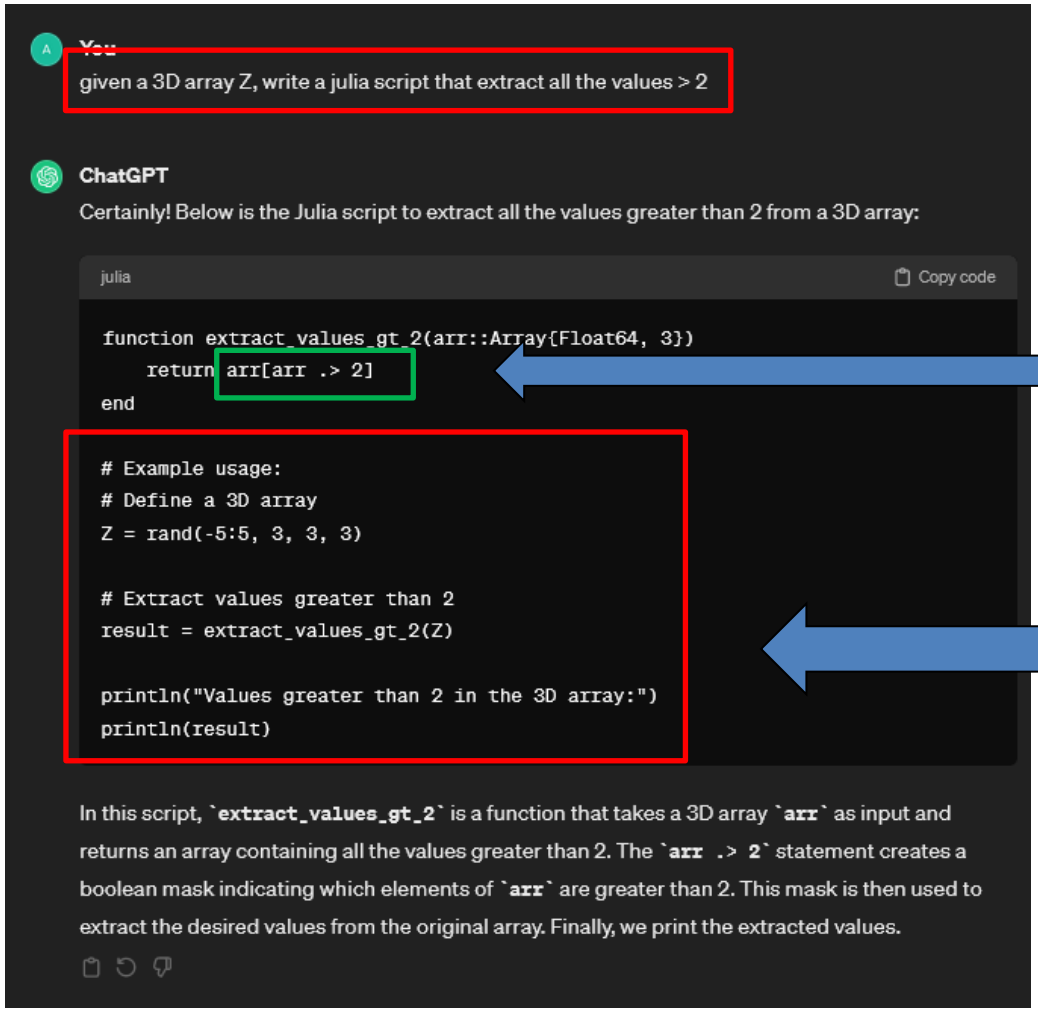
- When there is a bug, try to understand the error message
  - *Start at the beginning of the error message!*
  - *If this still does not help, try to come back to a working version*
  - *Ask for help!*

- Spend a bit of time making your scripts readable for you
  - *Making it look good helps*
  - *Adding comments is important*

- Modeling is like Lego: you can do a lot, but you are constrained by the building blocks!

Some tips (based on experience)

- Use ChatGPT when you get stuck for common task.
  Mind that ChatGPT is nothing more than a complex interpolation. If there are a lot of data, the answer may be useful



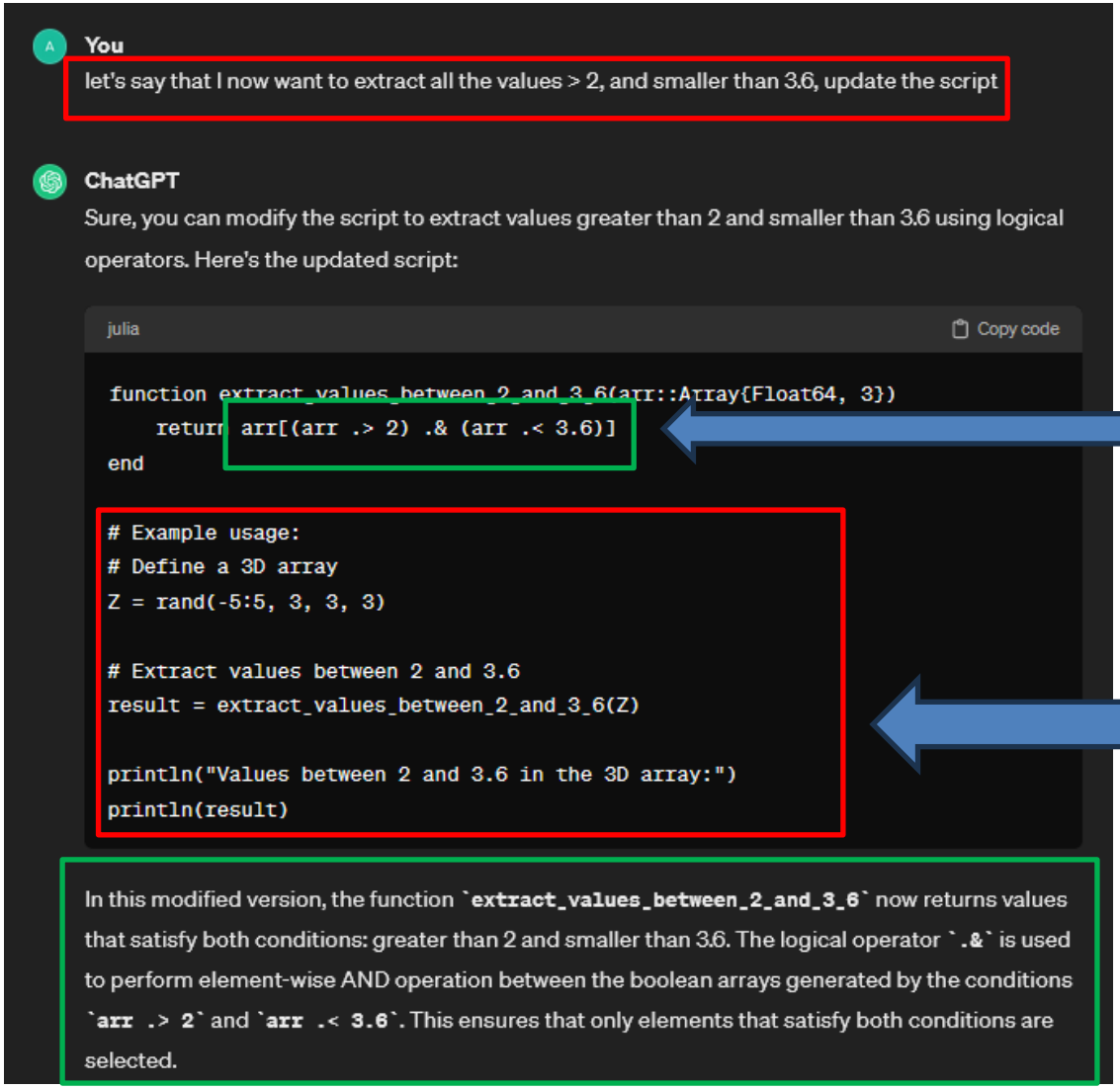
This is how to do!

Mostly useless

Some tips (based on experience)

- Once you have a prompt that is useful update it

For instance:



This is how to do!

Mostly useless

Very useful

Some tips (based on experience)

- ChatGPT knows nothing about how to set a LaMEM model up, or any other scientific modelling software in general

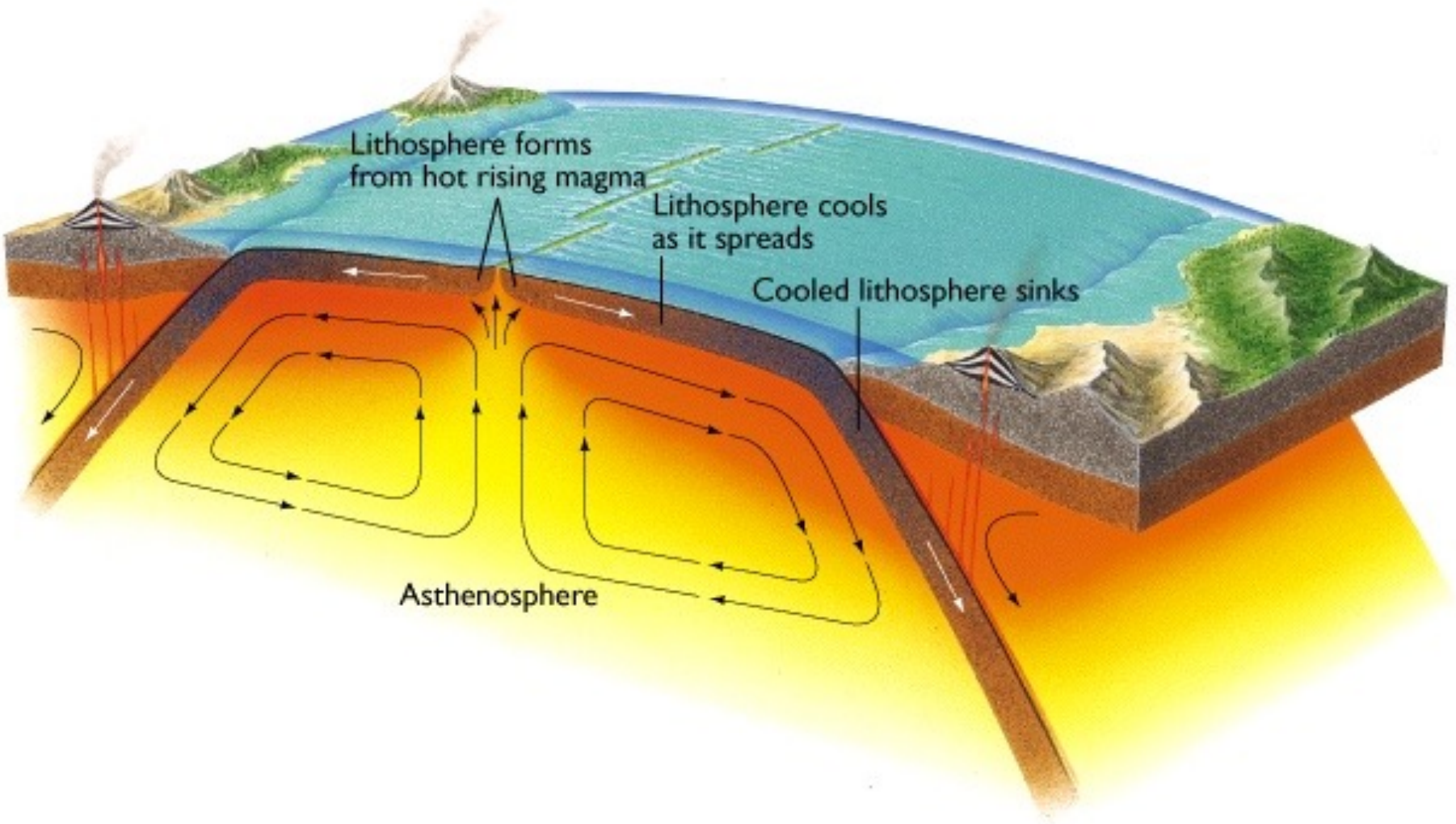
- It is however useful to help with Julia programming e.g.
    - How to navigate between different Julia terminal modes such as calculation mode, package manager, shell and help
    - Help for array creation, access and modification which will use extensively
    - May be helpful to better understand Julia related error message (in some cases)
    - Can also potentially help in finding out some syntax mistakes

LaMEM

Lithosphere and Mantle Evolution Model

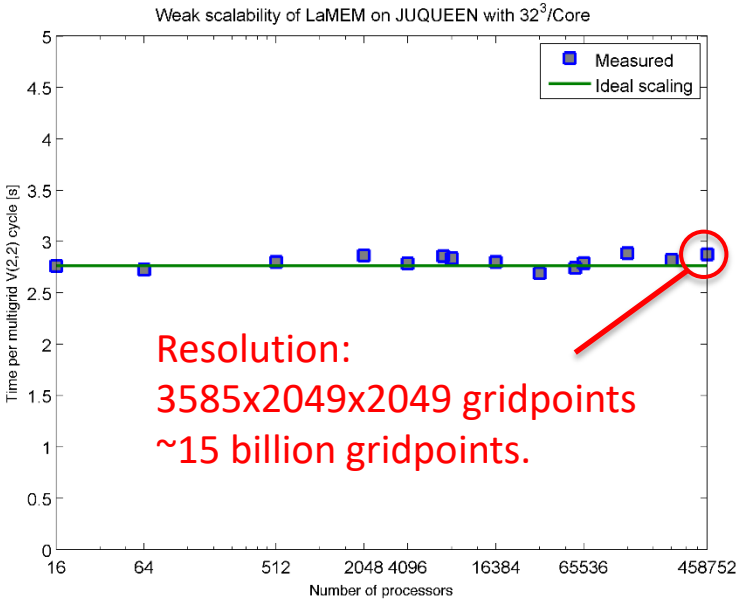
General overview

Geodynamic modelling



What is LaMEM?

- **L**ithosphere **a**nd **M**antle **E**volution **M**odel
- 3D thermo-mechanical code, written in C/PETSc.
- Nonlinear visco-elasto-visco-plastic rheologies
- MPI-parallel; runs on 1-458’752 processors.
- Staggered finite difference method (faster than FEM).
- Can use a large variety of (multigrid) solvers (Galerkin GMG, AMG, Coupled/decoupled)
- Marker-and-cell method, free surface, (coupled to simple erosion model).
- Phase transitions
- Polygonal meshes (geomIO) to create (complex) 3D input geometries.



Equations

$$\frac{\partial u_i}{\partial x_i} = -\frac{1}{K}\cdot\frac{dP}{dt}$$ conservation of mass

$$-\rho g_i = -\frac{\partial P}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}$$ conservation of momentum

$$\dot{\varepsilon}_{ij} = \frac{1}{2\eta_{eff}}\tau_{ij} + \frac{1}{2G}\frac{D\tau_{ij}}{Dt} + \dot{\lambda}\frac{\partial Q}{\partial \sigma_{ij}}$$ visco-elasto-plastic rheology

$$\rho c \frac{DT}{Dt} = \frac{\partial}{\partial x_i}\left(k\frac{\partial T}{\partial x_i}\right) + H_{sources}$$ conservation of energy

"Stokes" flow

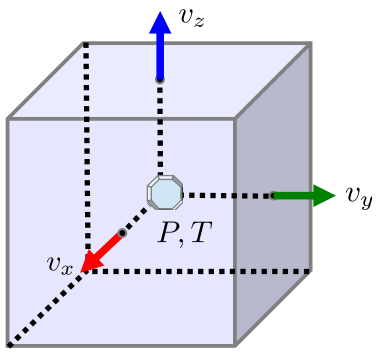
$$\frac{\partial u_i}{\partial x_i} = 0$$

$$-\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\eta_{eff}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right) = -\rho g_i$$

$$\longrightarrow \begin{pmatrix} \mathbf{K} & \mathbf{G} \\ \mathbf{G}^{\mathbf{T}} & \mathbf{0} \end{pmatrix}\begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

- Complexities mainly from the nonlinear rheology.
- Typically need to be solved numerically.
- No time derivative in Stokes equations!
  - For given density and viscosity distribution, velocity is given everywhere

Parallel staggered grid layout implementation



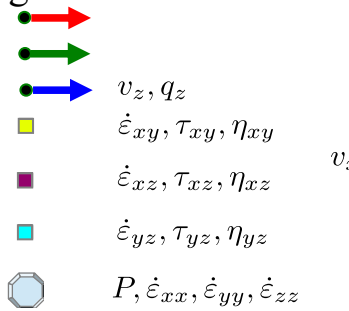
Equations:

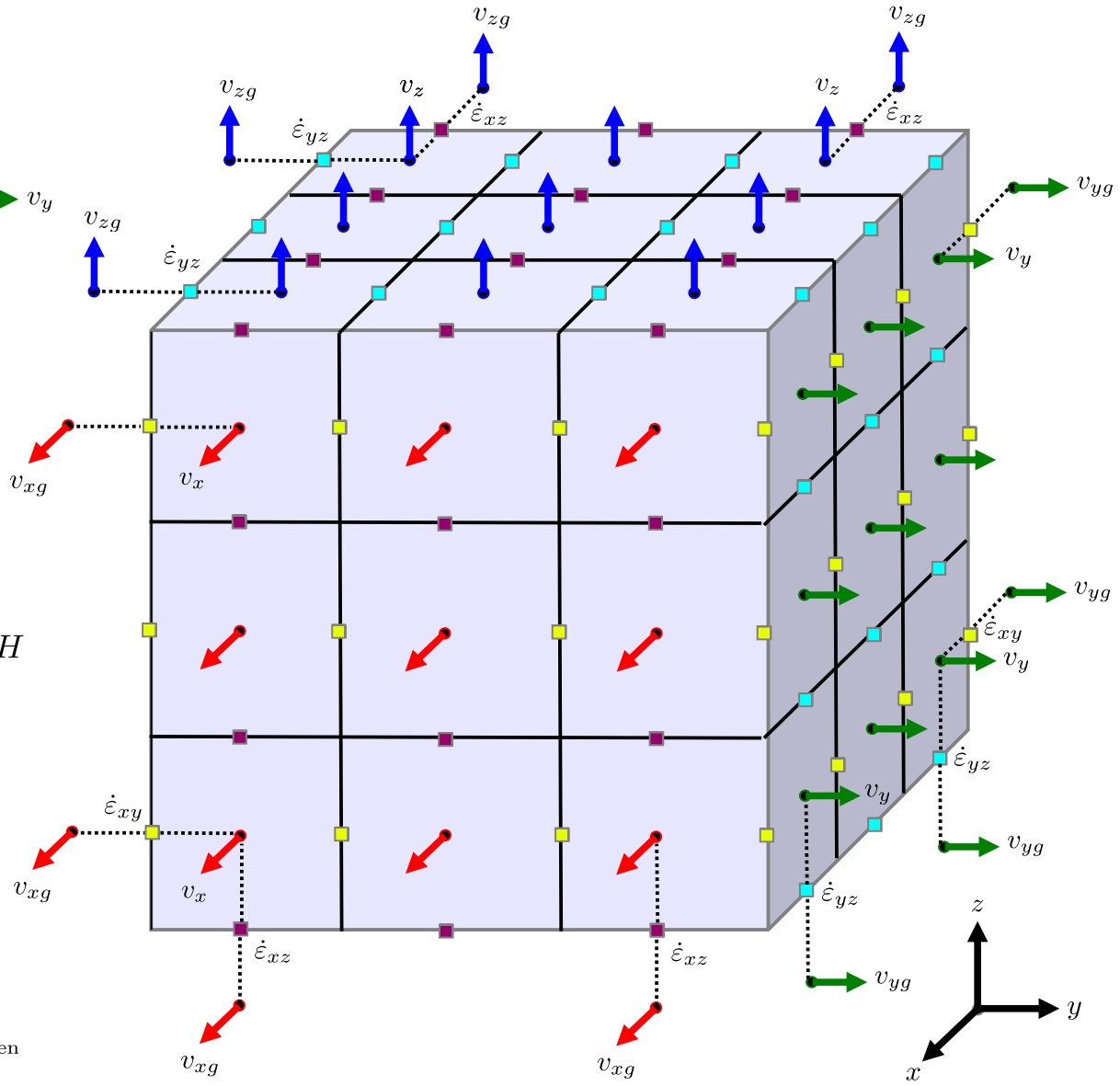
$$\frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial P}{\partial x_i} + \rho g_i = 0$$

$$\frac{1}{\rho}\frac{D\rho}{Dt} + \frac{\partial v_i}{\partial x_i} = 0$$

$$\rho C_p \frac{DT}{Dt} = \frac{\partial}{\partial x_i}\left(\lambda \frac{\partial T}{\partial x_i}\right) + H$$

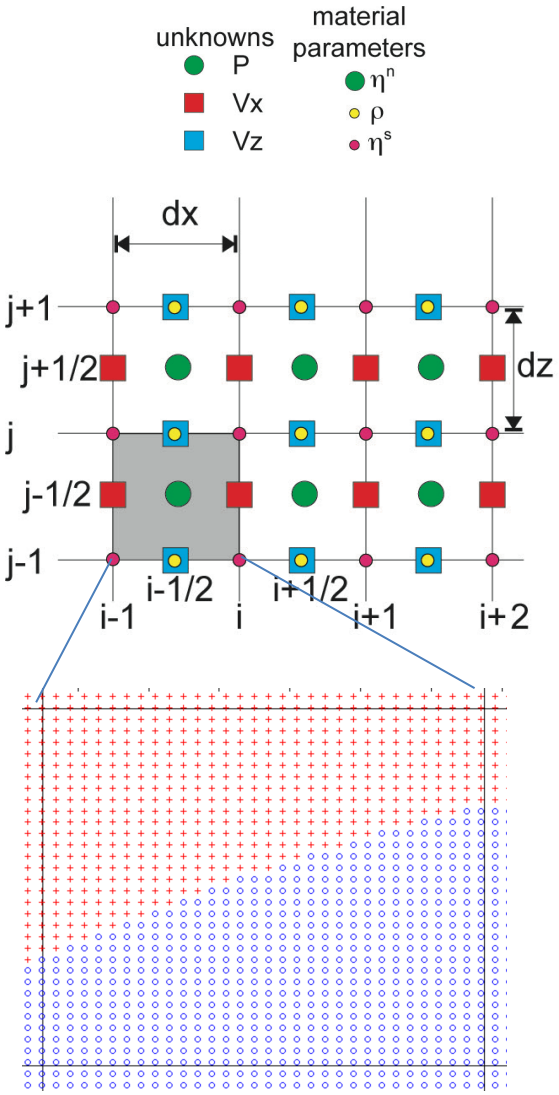
Legend:





Marker & Cell approach

- Rock types are tracked by markers (called “Particles” in LaMEM).
- Each particle:
  - Is of a certain rocktype (or “phase ID”)
  - Contains info such as temperature, stress etc.
- Are advected with the flow
- During every timestep, the particles are mapped back to the computational grid, and vice versa



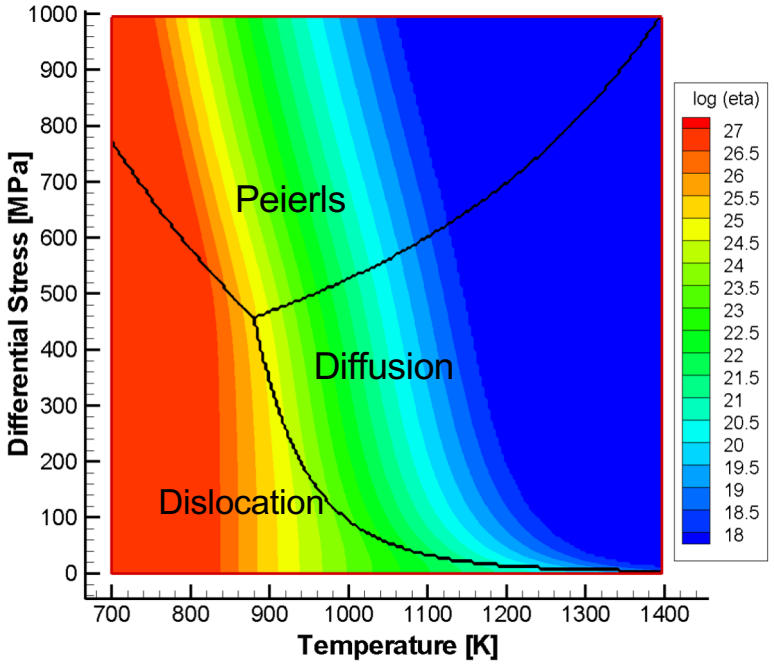
Nonlinear visco-elasto-plastic rheology

Drucker-Prager yield stress (brittle faulting)

$$\tau_Y = \mu P + c$$

Diffusion, Dislocation and Peierls creep

$$A_l = B_l \exp\left[-\frac{E_l + pV_l}{RT}\right],$$
$$A_n = B_n \exp\left[-\frac{E_n + pV_n}{RT}\right]$$
$$A_p = \frac{B_p}{(\gamma\tau_p)^s} \exp\left[-\frac{E_p + pV_p}{RT}(1-\gamma)^q\right]$$



Effective creep viscosities

$$\eta_l = \frac{1}{2}(A_l)^{-1}$$
$$\eta_n = \frac{1}{2}(A_n)^{-\frac{1}{n}}(\dot{\varepsilon}^*_{II})^{\frac{1}{n}-1}$$
$$\eta_p = \frac{1}{2}(A_p)^{-\frac{1}{s}}(\dot{\varepsilon}^*_{II})^{\frac{1}{s}-1}$$

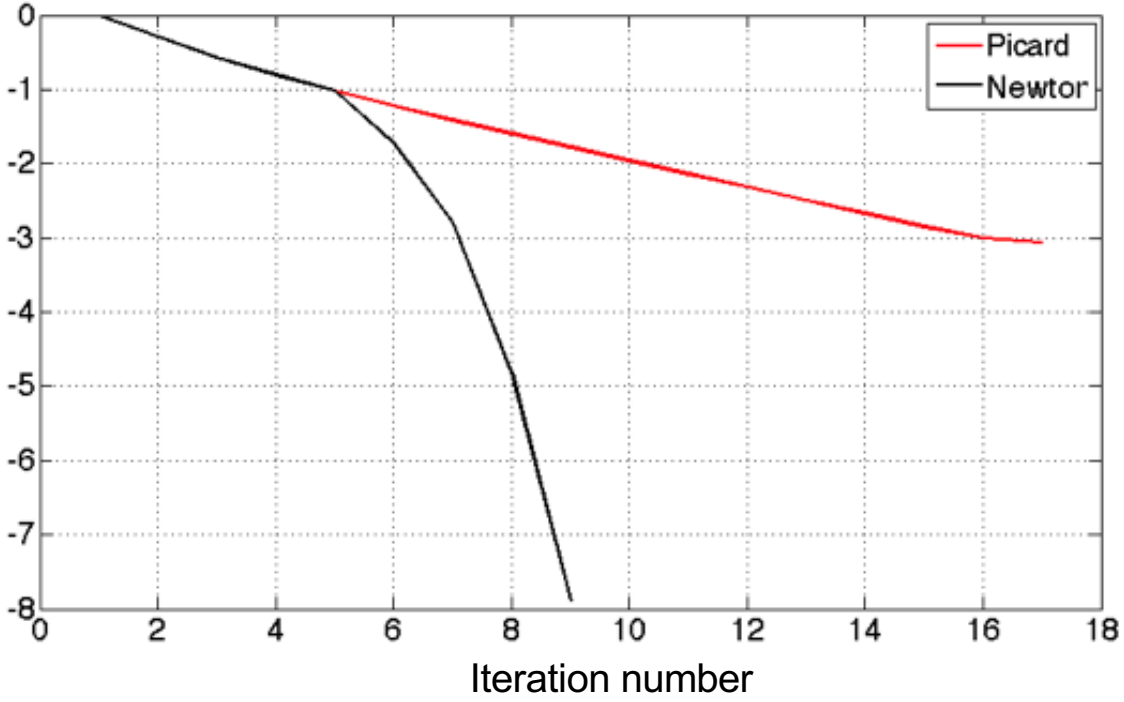
Peierls effective exponent

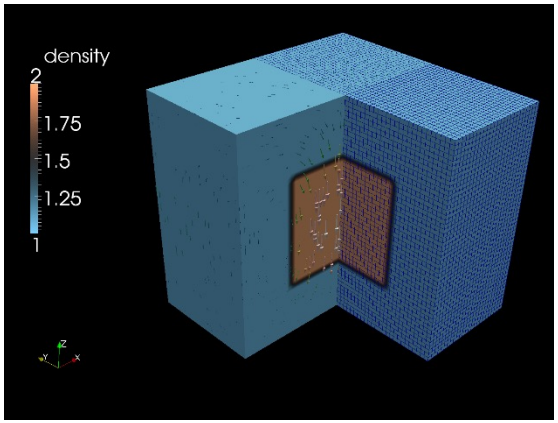
$$s = \frac{E_p + PV_p}{RT}(1-\gamma)^{q-1} q\,\gamma$$

Picard vs. Newton

Quasi-linear residual form:

Picard fixed-point iteration: $J(x) \approx A(x)$

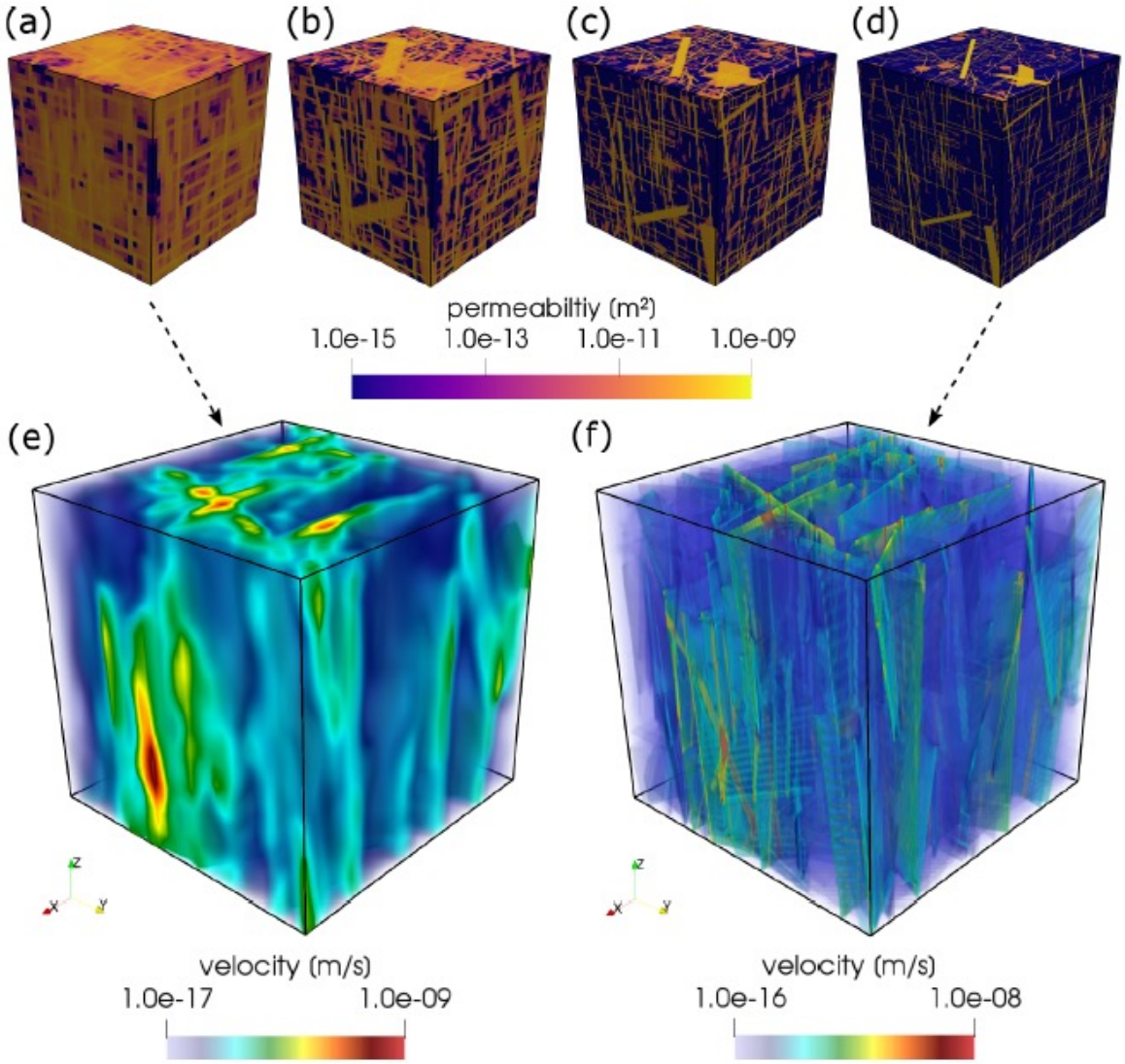




- Newton much faster than Picard for nonlinear materials
- Picard approximation facilitates convergence at the initial stages
- Switching to Picard can improve Newton convergence

Some recent LaMEM applications

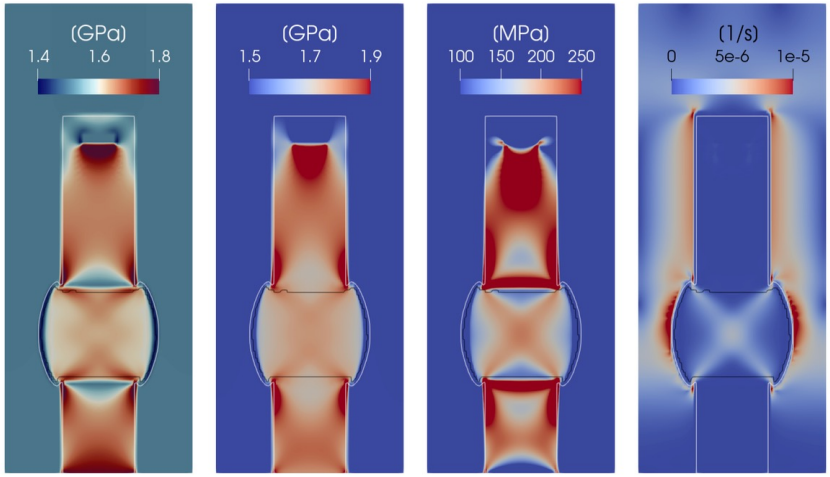
Fracture network permeability study



Kottwitz et al., 2021 – Solid Earth

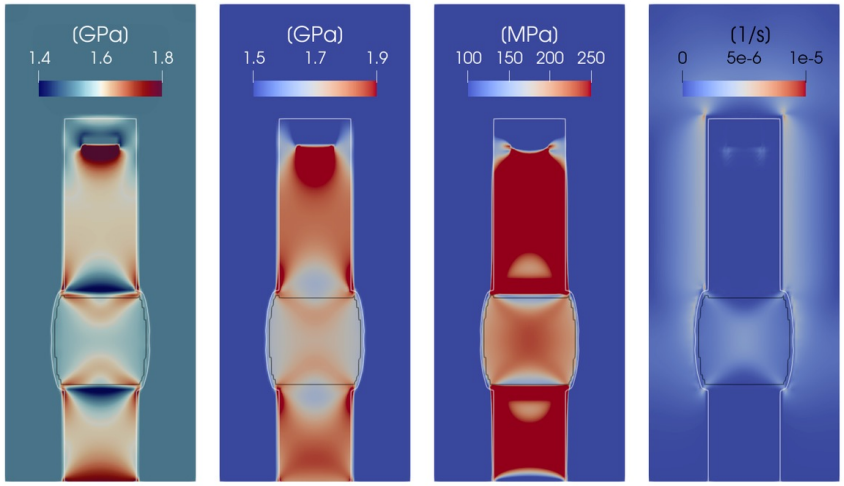
High-pressure rock deformation experiments

2D



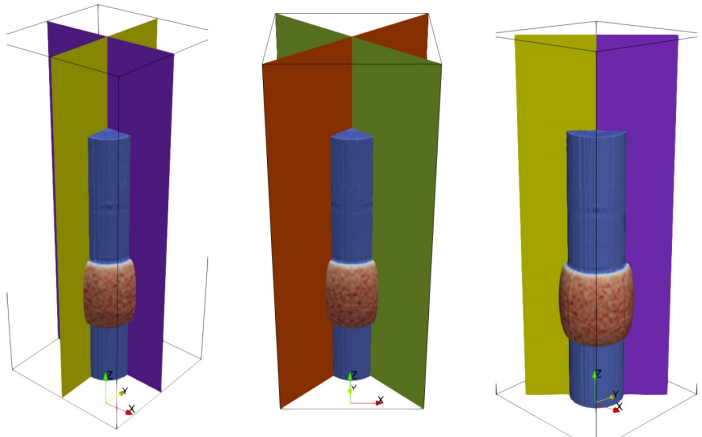
(a) pressure (b) σ_1 (c) $\Delta\sigma$ (d) strain rate

3D



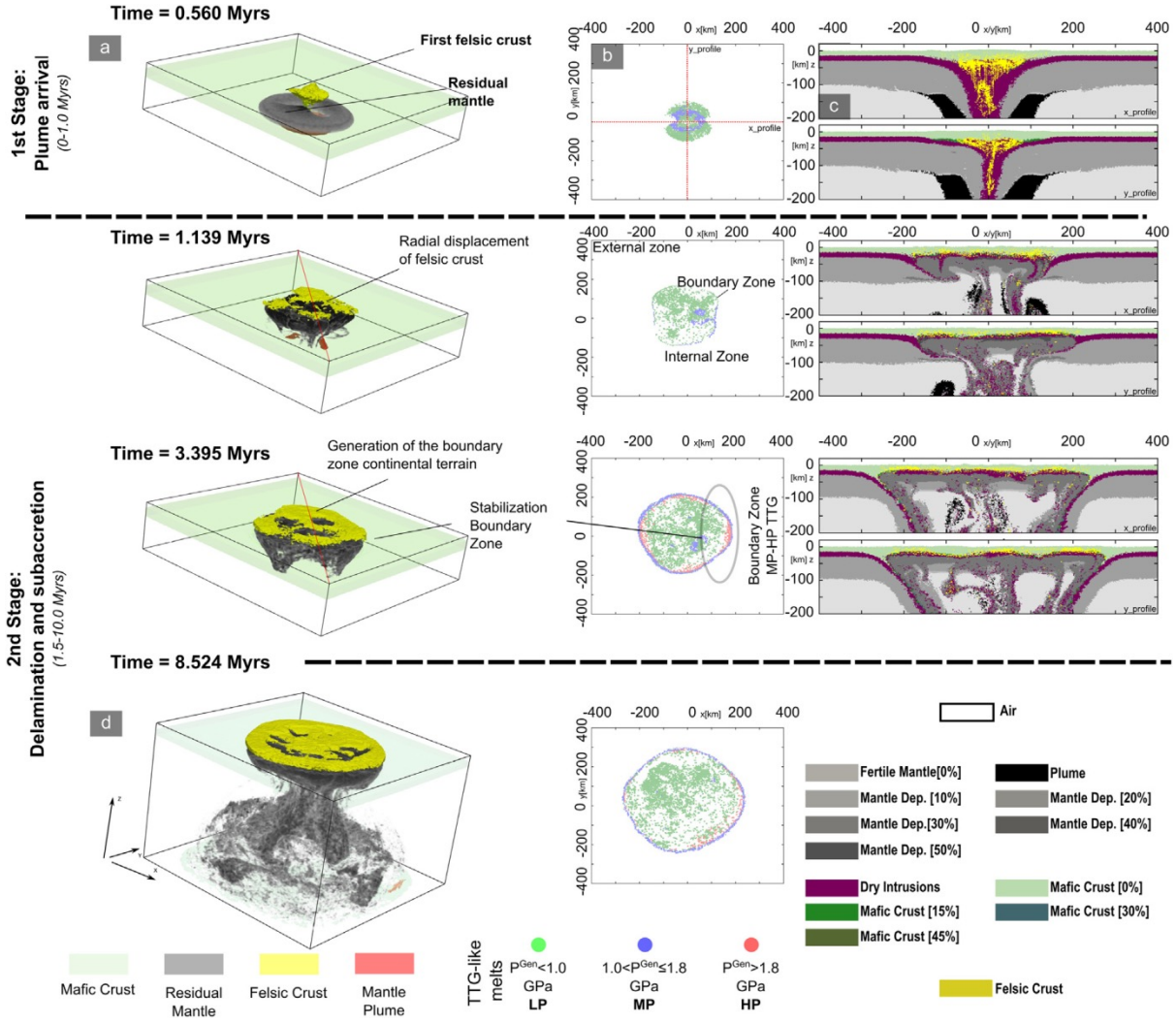
(a) pressure (b) σ_1 (c) $\Delta\sigma$ (d) strain rate

Kilian Herrmanns
(BSc Thesis, Heidelberg)



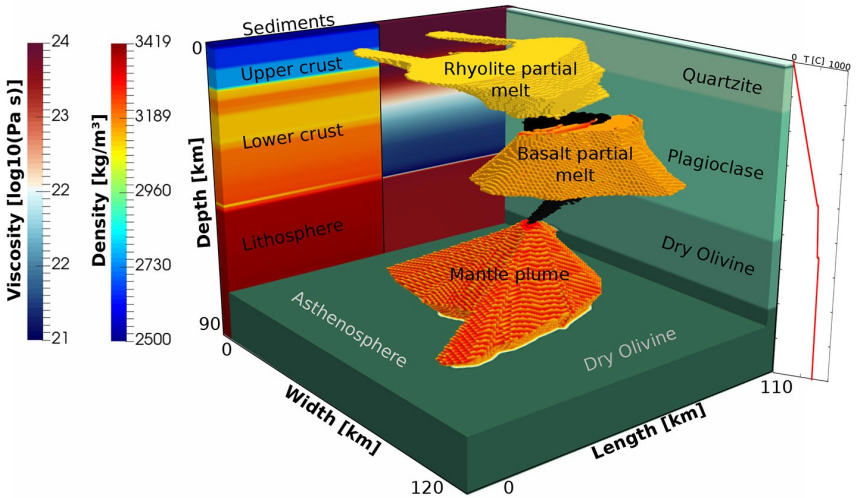
(a) X and Y planes (b) Diagonal planes [110] and [-110] (c) Display of plane orientation for A6

Plume - LID interaction during the Archean

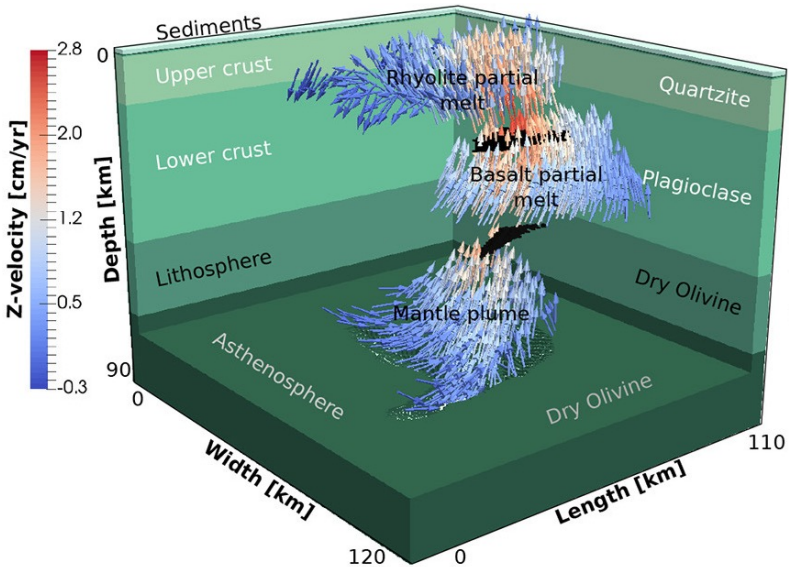


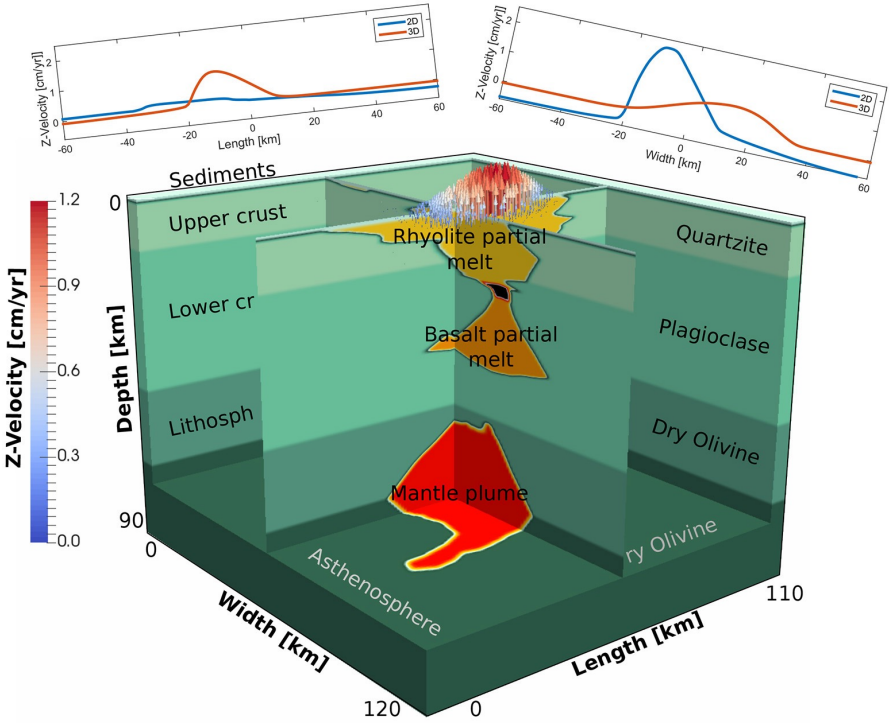
Piccolo et al., 2020 – Gondwana research

Yellowstone Magmatic System



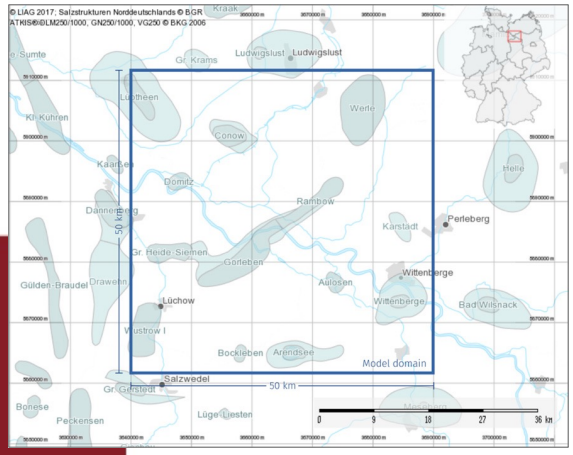
Reuber et al., 2018, Frontiers in Earth Sciences

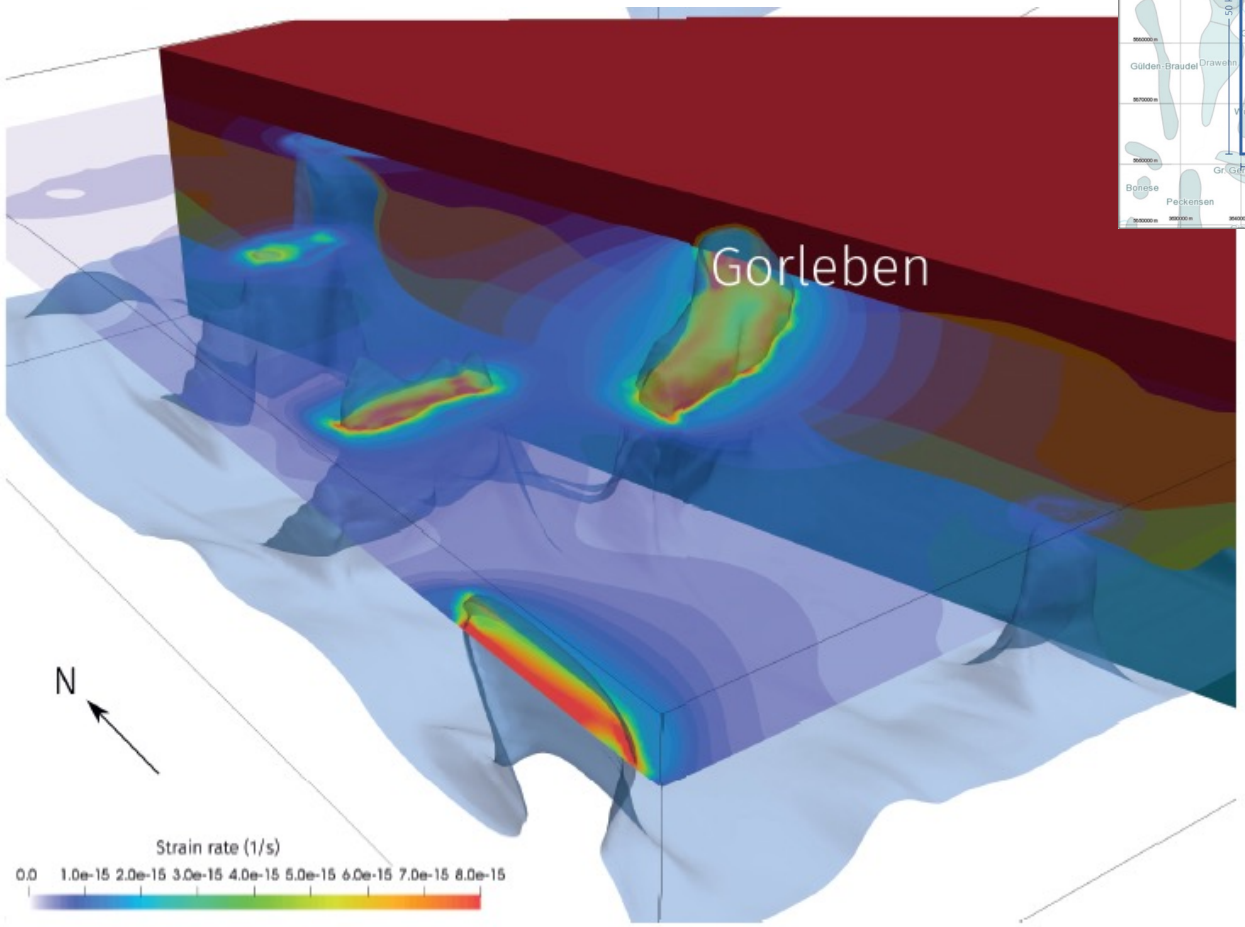




Gorleben salt dome

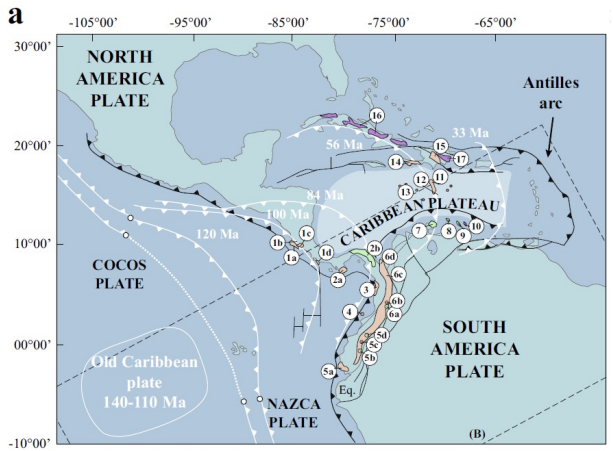
- Salt cavities as potential storage of radioactive waste?

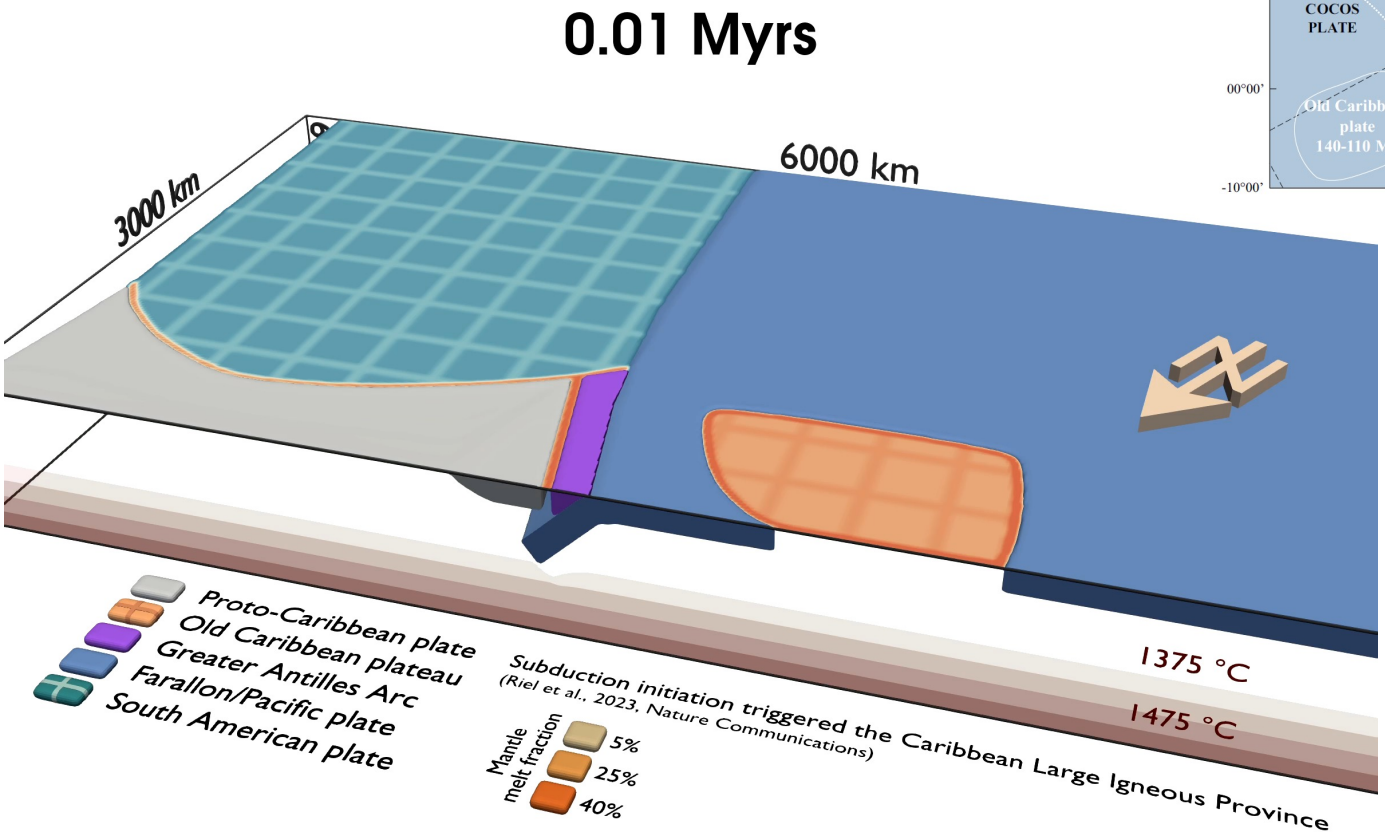




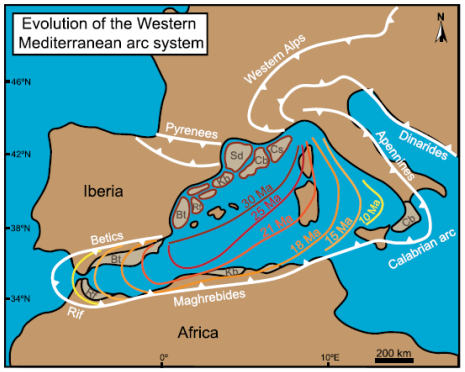
Baumann et al., 2018

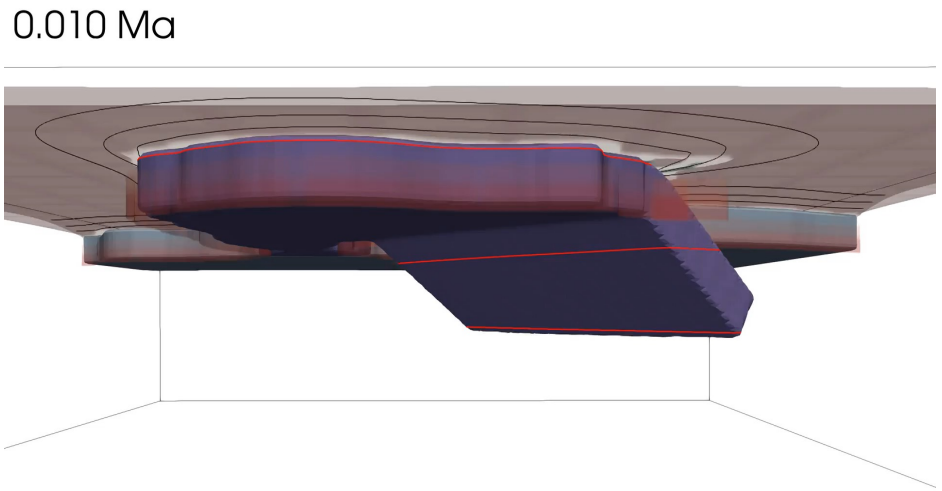
Formation of Caribbean LIP

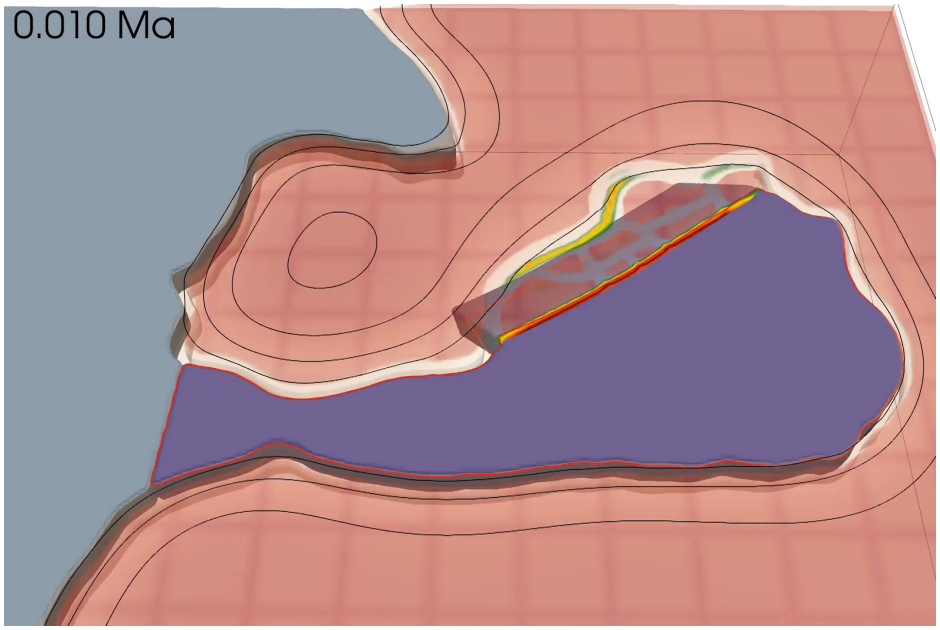




Subduction invasion - Gibraltar

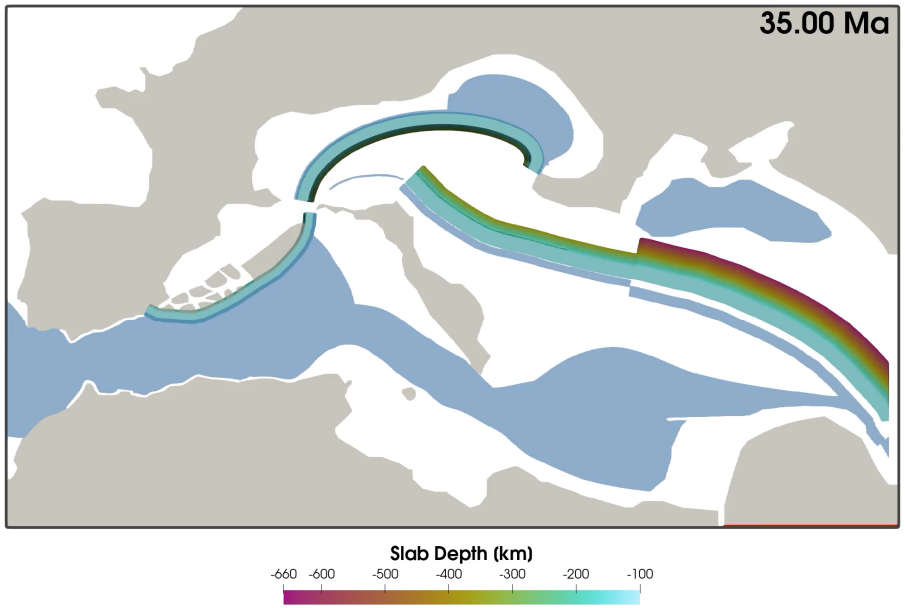


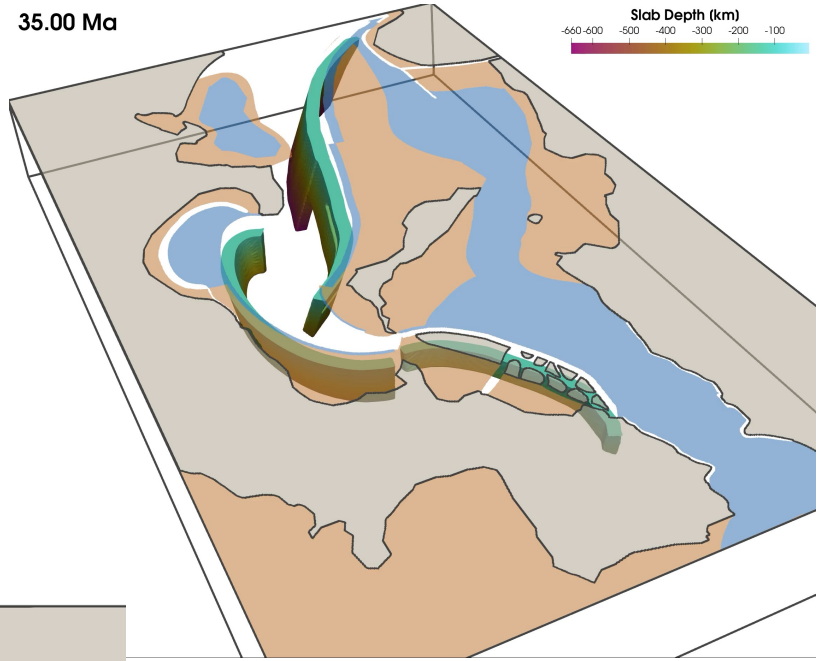


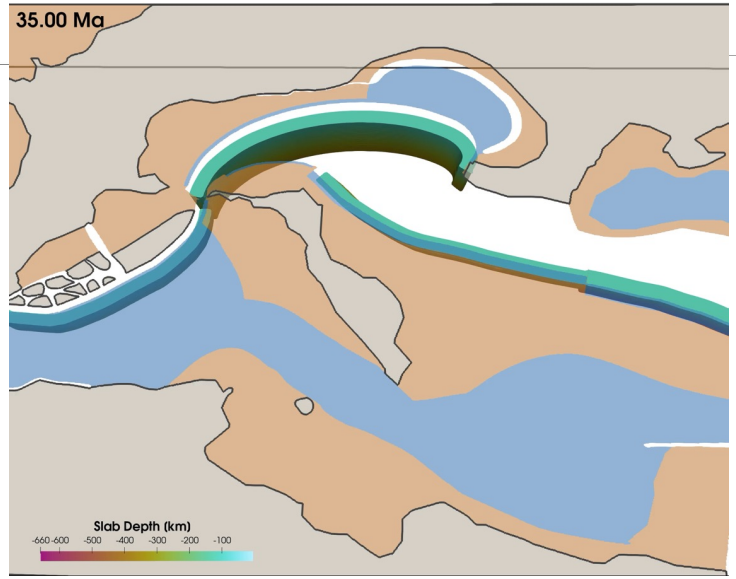


Duarte, Riel et al ., 2024 – Geology

Mantle Dynamics in the Mediterranean and Plate Motion of the Adriatic Microplate



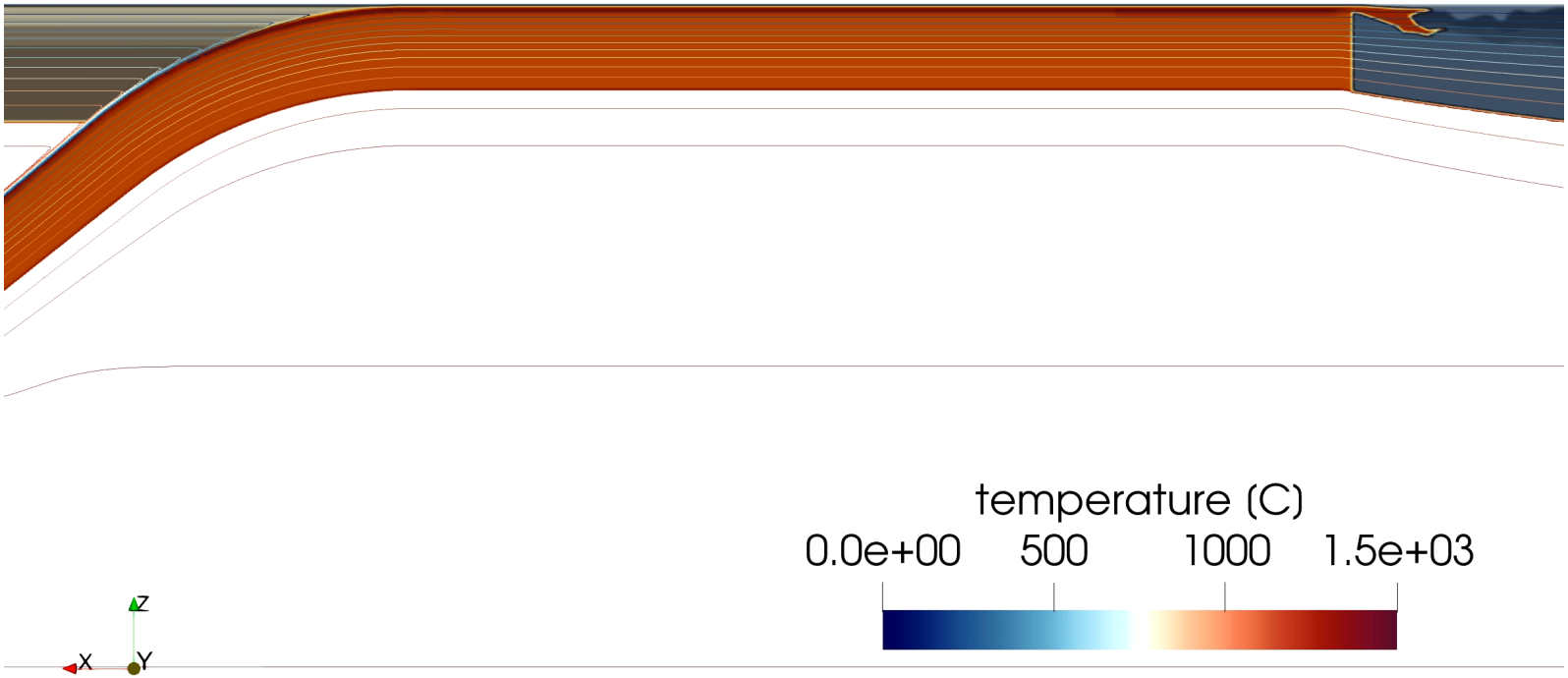




Schuler et al., 2025 G-cubed

Role of lower crust in Orogen build-up

Rodrigues et al, in prep



Ridge opening and mantle window

Rojas-Agramonte, to be submitted