MONASH University Science

www.sci.monash.edu

Underworld – what have we learned ?

Louis Moresi, Mirko Velic

Steve Quenette, John Mansour, Julian Giordani

Monash University

School of Geosciences &

School of Mathematical Sciences

Monash eResearch Centre

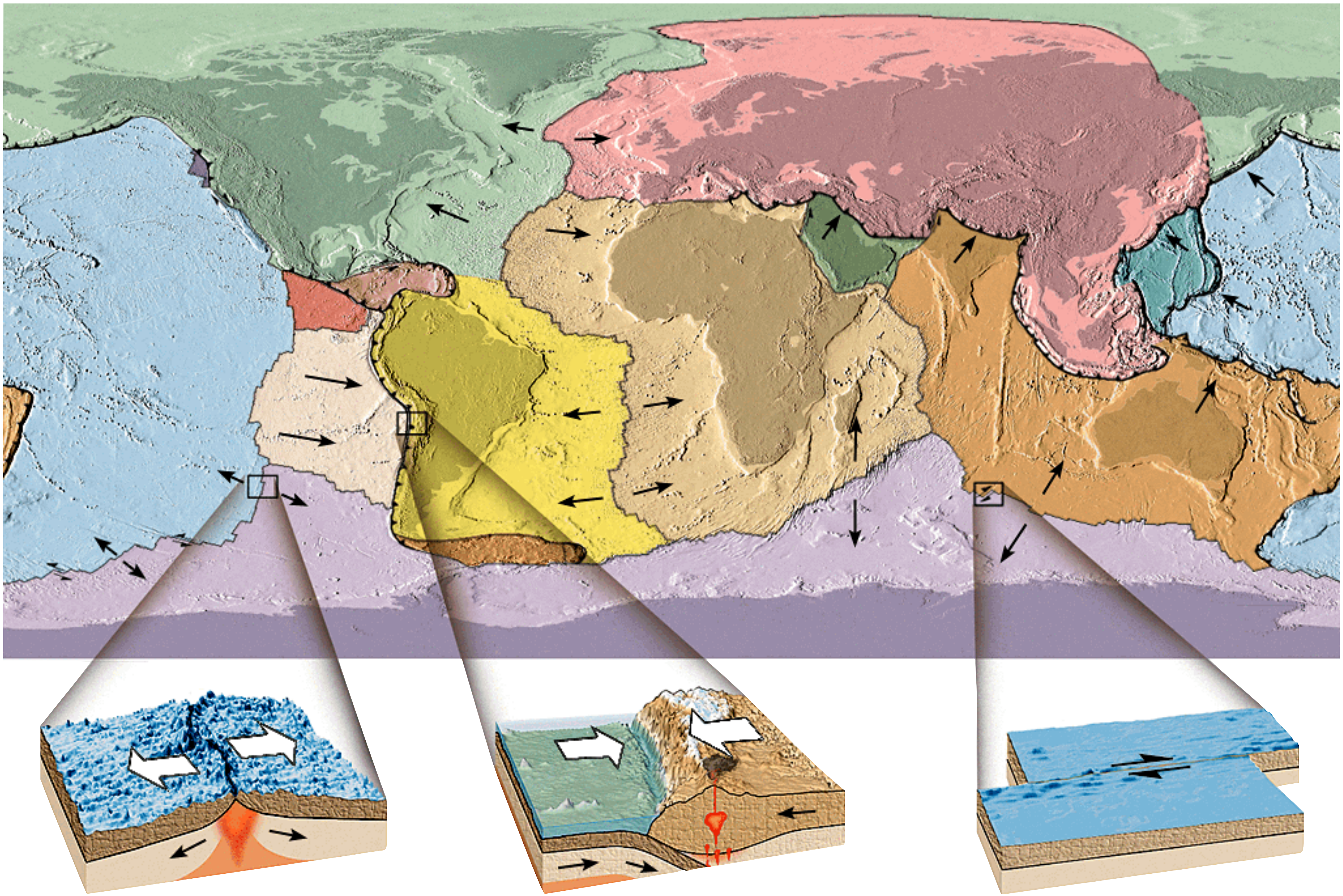
Auscope

Simulation, Analysis, Modelling Group

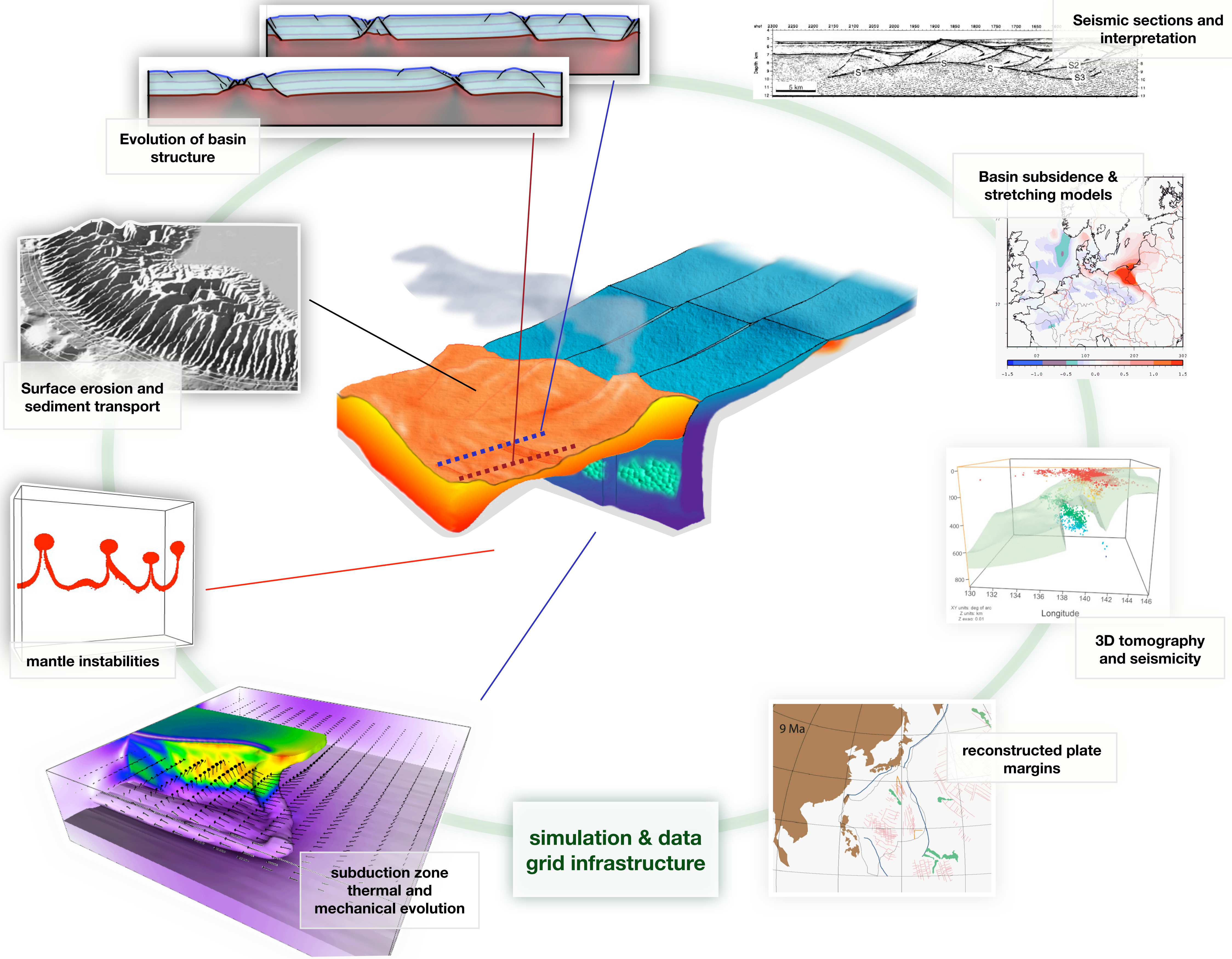
NeCTAR

eResearch Tools "Geology from Geodynamics"

Plate tectonics — a multi-scale problem



Integrated Earth modelling



Equations

Ugly !!

$$\tau_{ij} = 2\eta D_{ij} = \eta\left(u_{i,j} + u_{j,i}\right)$$

$$\tau_{ij,j} - p_{,i} = f_i$$

$$u_{i,i} = 0$$

$$f = \mathbf{g}\rho_0\alpha\Delta T$$

Stokes flow, incompressible
Elliptic problem - multigrid

$$T_{,t} + u_k T_{,k} = (\kappa T_{,k})_{,k} + Q$$

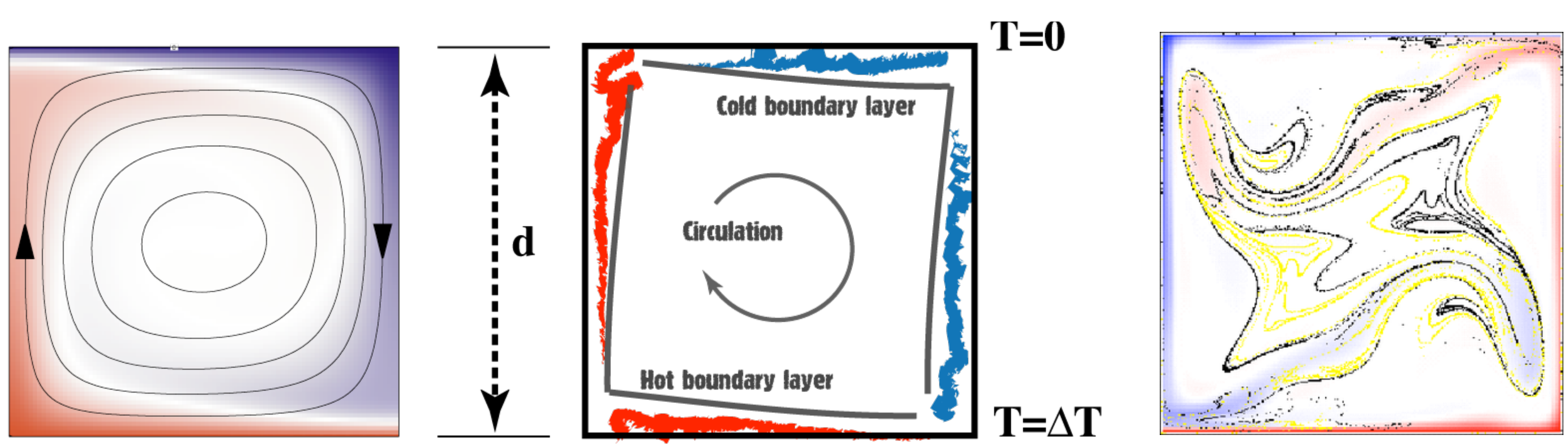
Advection / diffusion

Challenges: High strain accumulation during fluid-like deformation

This is a Rayleigh-Bénard convection model which evolves to a straightforward balance between thermal diffusion and thermal advection in narrow boundary layers.

At modest Rayleigh number, the structure which develops is steady despite strongly developed convective flow.

This system can be solved very efficiently on a fixed mesh



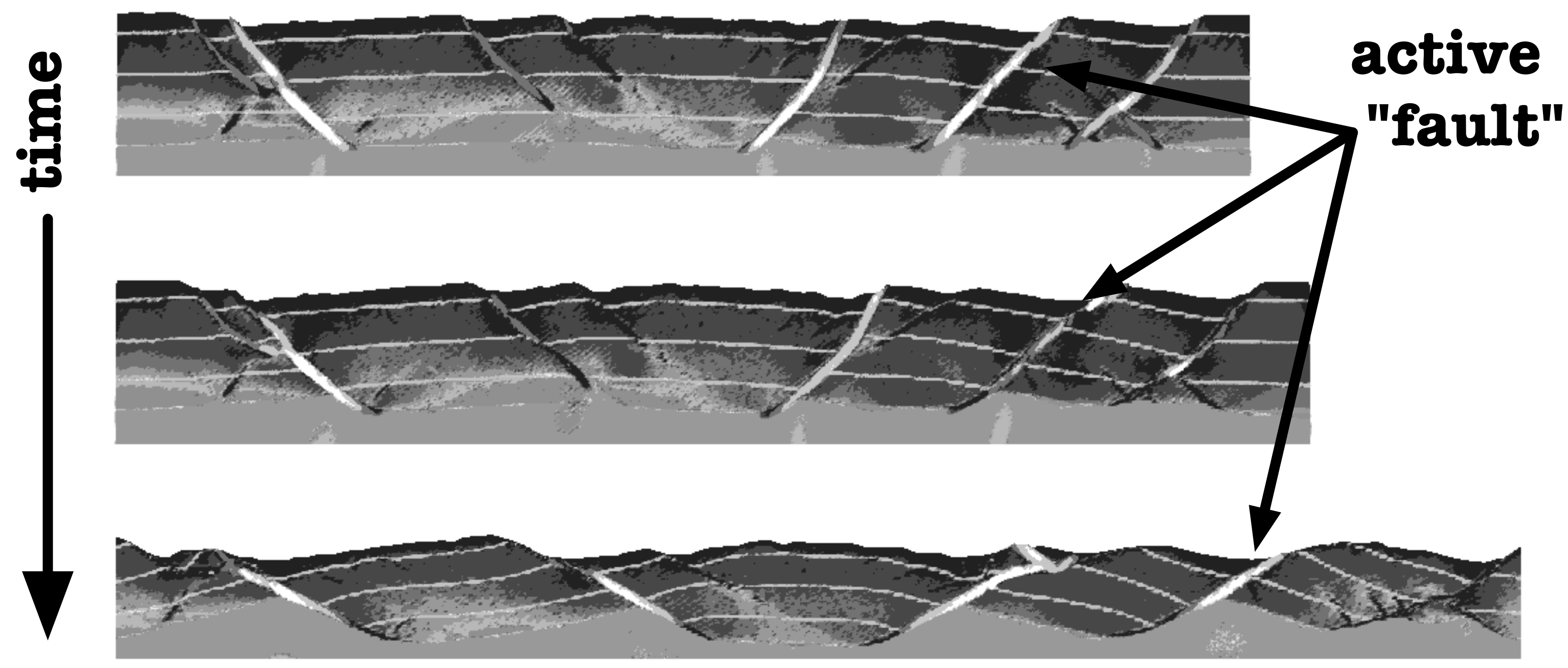
$\mathrm{Ra} = 10^7$

Challenges: Strain-dependence of lithospheric deformation

This is a simulation of continental crust being stretched in response to far field stresses imposed by plate motions.

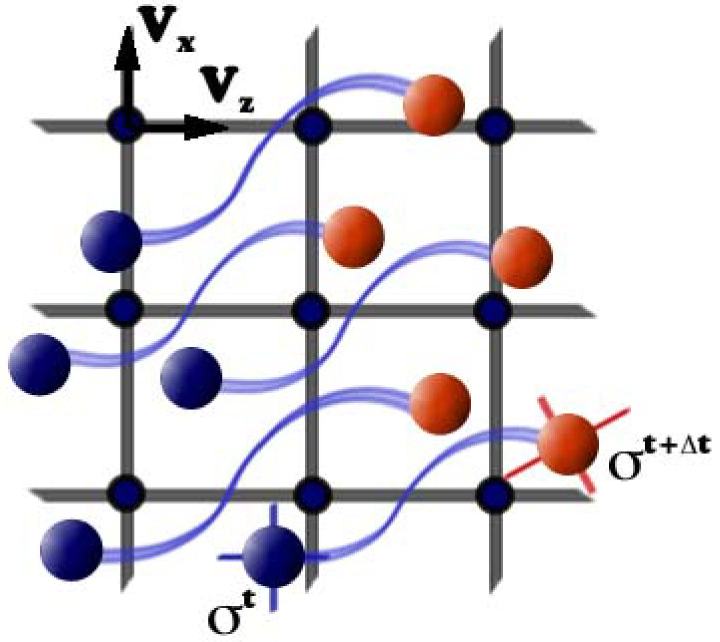
At modest strain, the deformation will often localise onto faults which can be very long-lasting structures; very fine scale in width, but with large lateral dimension and relatively weak.

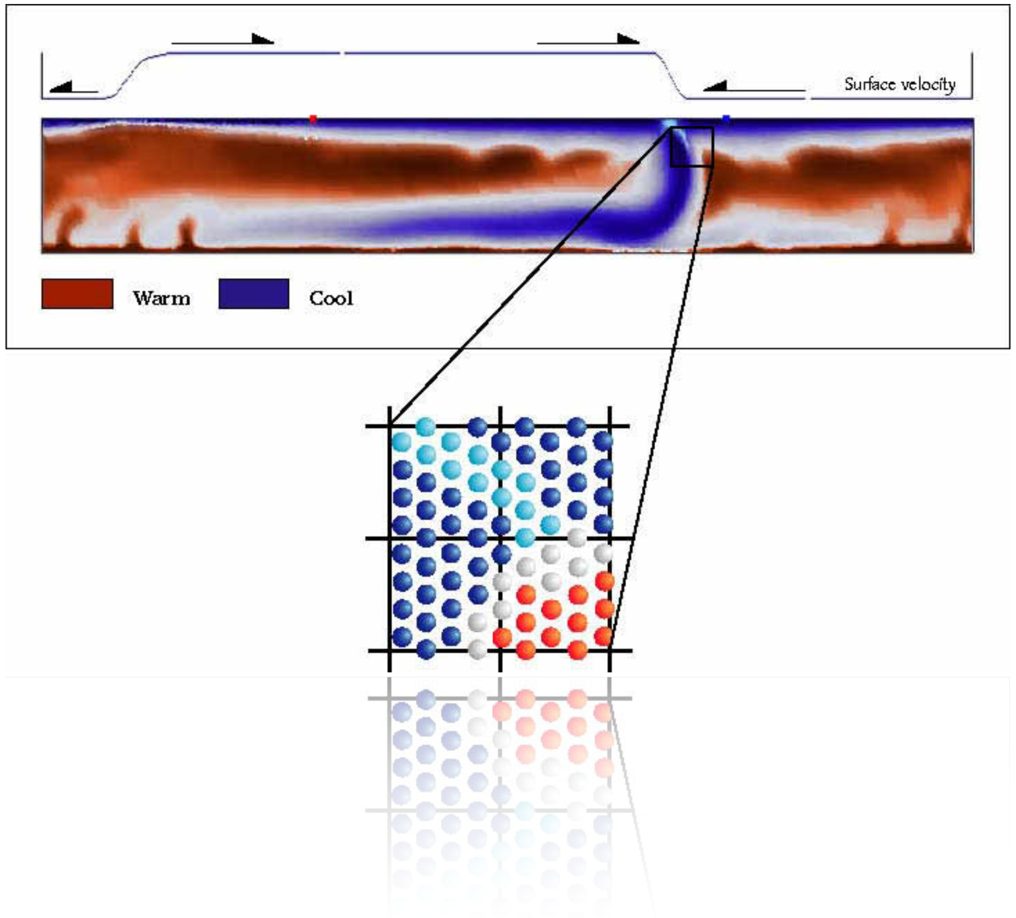
The history dependence of shear deformation is tractable if we use a Lagrangian reference frame.



Lagrangian History & Efficient Fluid solvers

In the **material point method** we can keep a mesh which is computationally efficient for diffusion-dominated problems (including Stokes flow) and material points — a.k.a. particles — for tracking history variables.

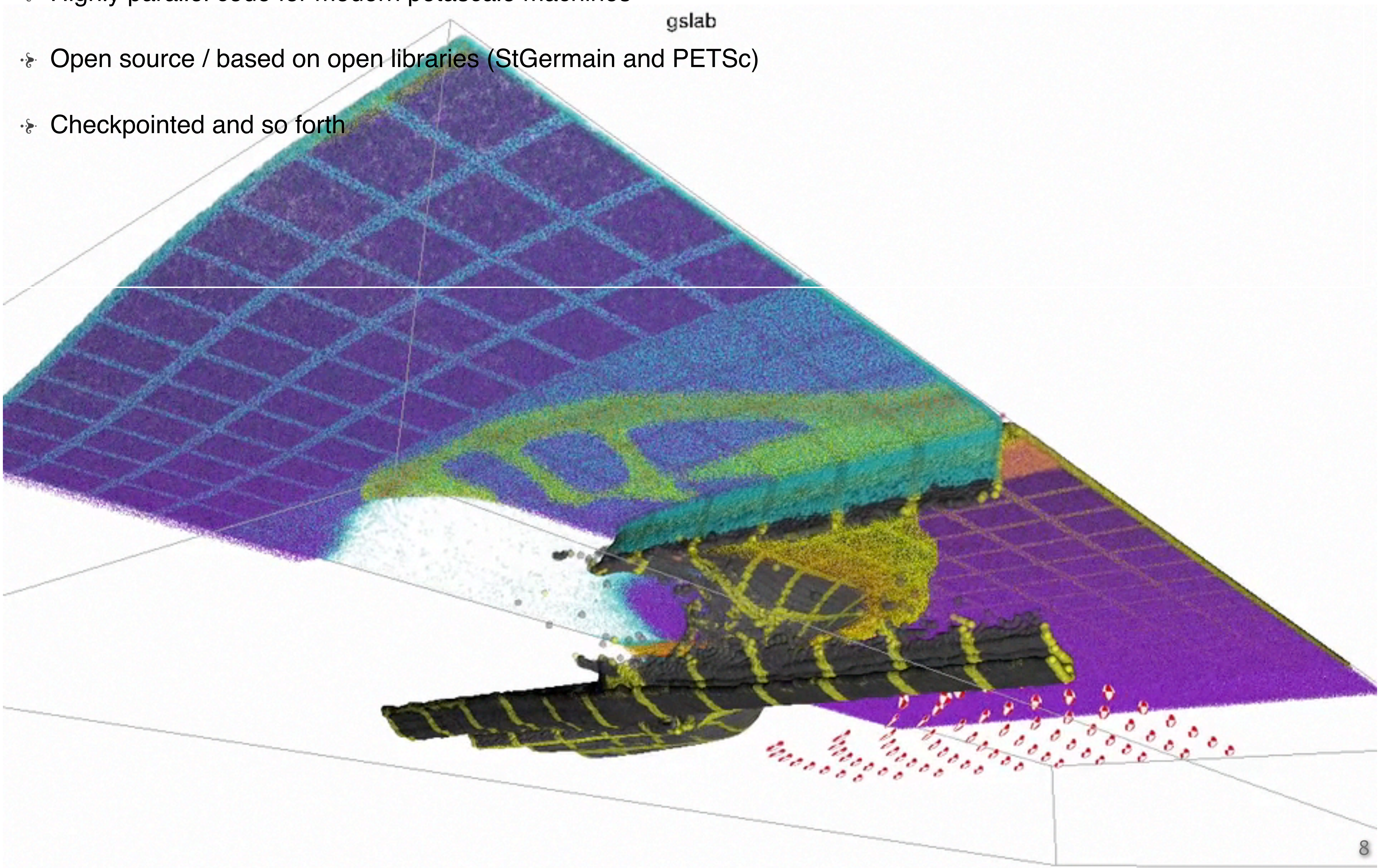




This is the technique implemented in Underworld and leads to a very natural approach to many “difficult” issues in geological thermal / mechanical models (www.underworldproject.org)

Underworld for plate scale models with material history

- Underworld — finite element models with tracking of small-scale physics
- Highly parallel code for modern petascale machines
- Open source / based on open libraries (StGermain and PETSc)
- Checkpointed and so forth



Underworld — aims / successes

We wanted to build a code that would be widely used and useful.

Easy to install	✗	✓	
Robust	✓	✓	More important than efficient
Easy to understand and use	✗	?	Better to build something effective
Highly parallel / efficient	✓	✓	
Very general	✓	✓	
Easy to maintain	✗	?	Better to build something effective

Community code

No more heroes ?	✗	?	Why knock the heroes ?
Vibrant community of users	✓	✓	
Everyone open source !	✓	?	Work with everyone !
Linked in to other workflows	✗	✓	

Robust solution is possible

The general approach is like this

mg configuration
accuracy ?

"outer" solve for p

$$(G^T \tilde{K}^{-1} G)p = (G^T \tilde{K}^{-1})F$$

"inner" mg solve for v*

$$\tilde{K} v_i^* = f_i^*$$

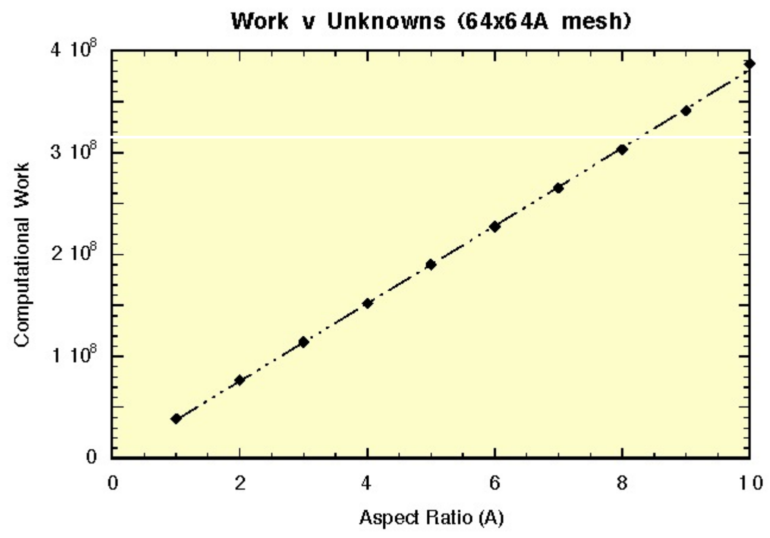
solve for v

$$Kv = F - Gp$$

which ksp / pc ?
accuracy ?
penalty ?

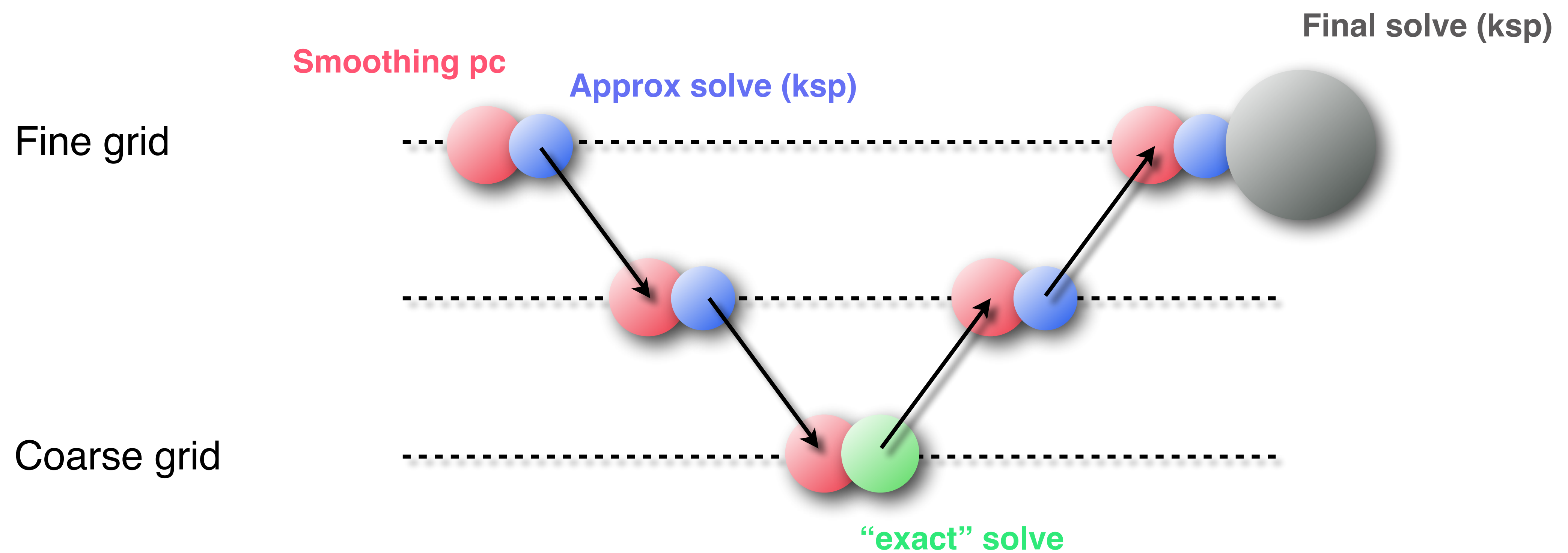
which ksp ?
accuracy ?
penalty ?

Multigrid "Inner solve"



Geometric multigrid

- Multiple, nested grids sharing common nodes
- Solution is obtained by combining solutions on all grids
- Ideal for elliptic problems in which information propagation is not local and is instantaneous



"V cycle" is MG preconditioner on final solver

Robust and general preconditioners are hard to find

Augmented Lagrangian Approach — Takes advantage of the variational nature of the FE formulation. To add a constraint it is always possible to introduce via Lagrange multipliers. Because "exact" compliance with the incompressibility constraint produces a very stiff ill-conditioned system, we can instead use an approximate form and use it to improve iterative convergence

$$Ku + \lambda G^T W^{-1} G u + Gp = F$$

Minimization drives this term to O(1/λ)

Replaces the stiffness matrix in the Schur complement

$$\tilde{K} = Ku + \lambda G^T W^{-1} G u$$

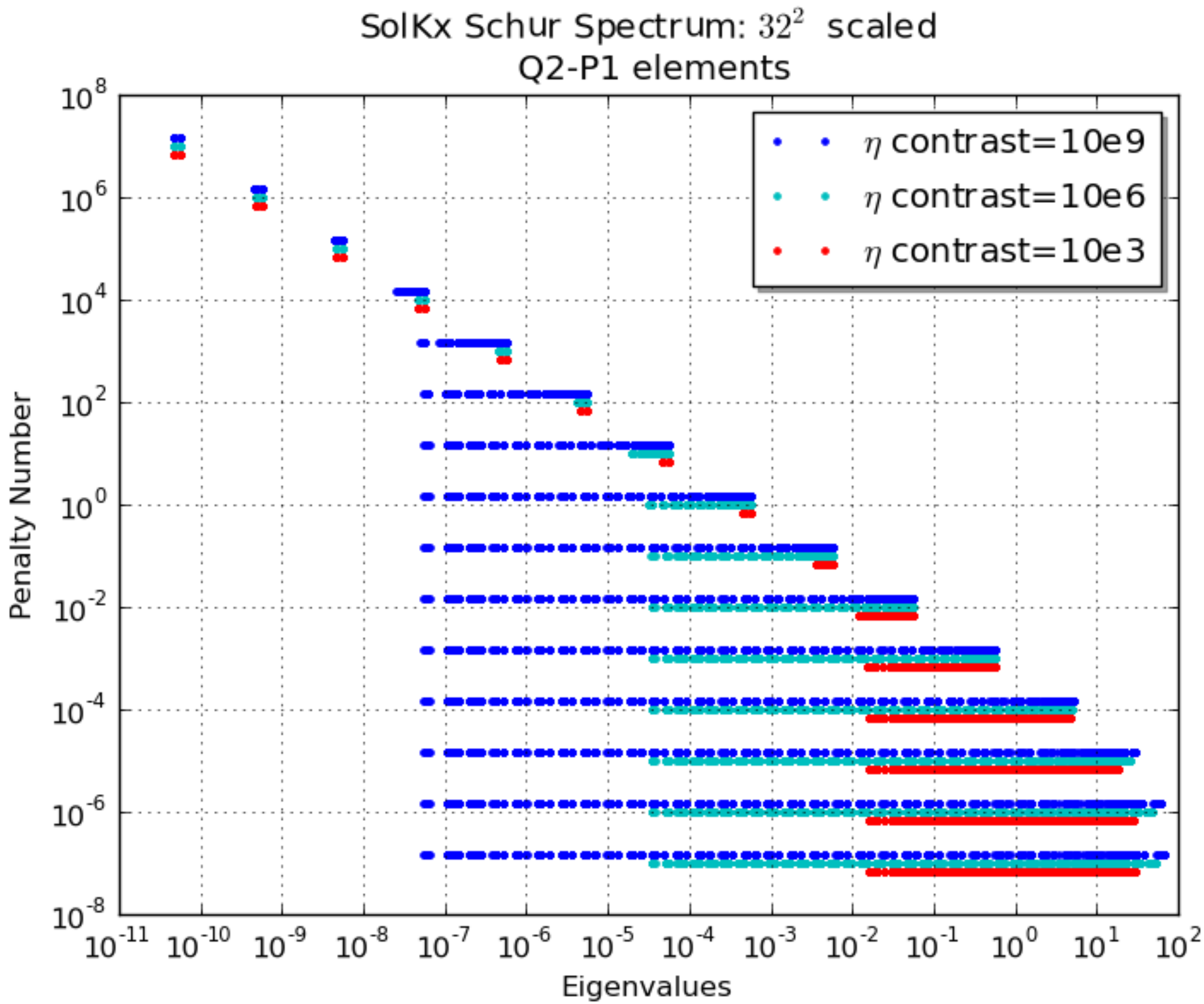
The equivalent augmented Lagrangian (AL) formulation [11] is given by

$$\begin{pmatrix} A + \gamma B^T W^{-1} B & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} \hat{f} \\ g \end{pmatrix}, \quad \text{or} \quad \widehat{\mathcal{A}} x = \hat{b}, \tag{10}$$

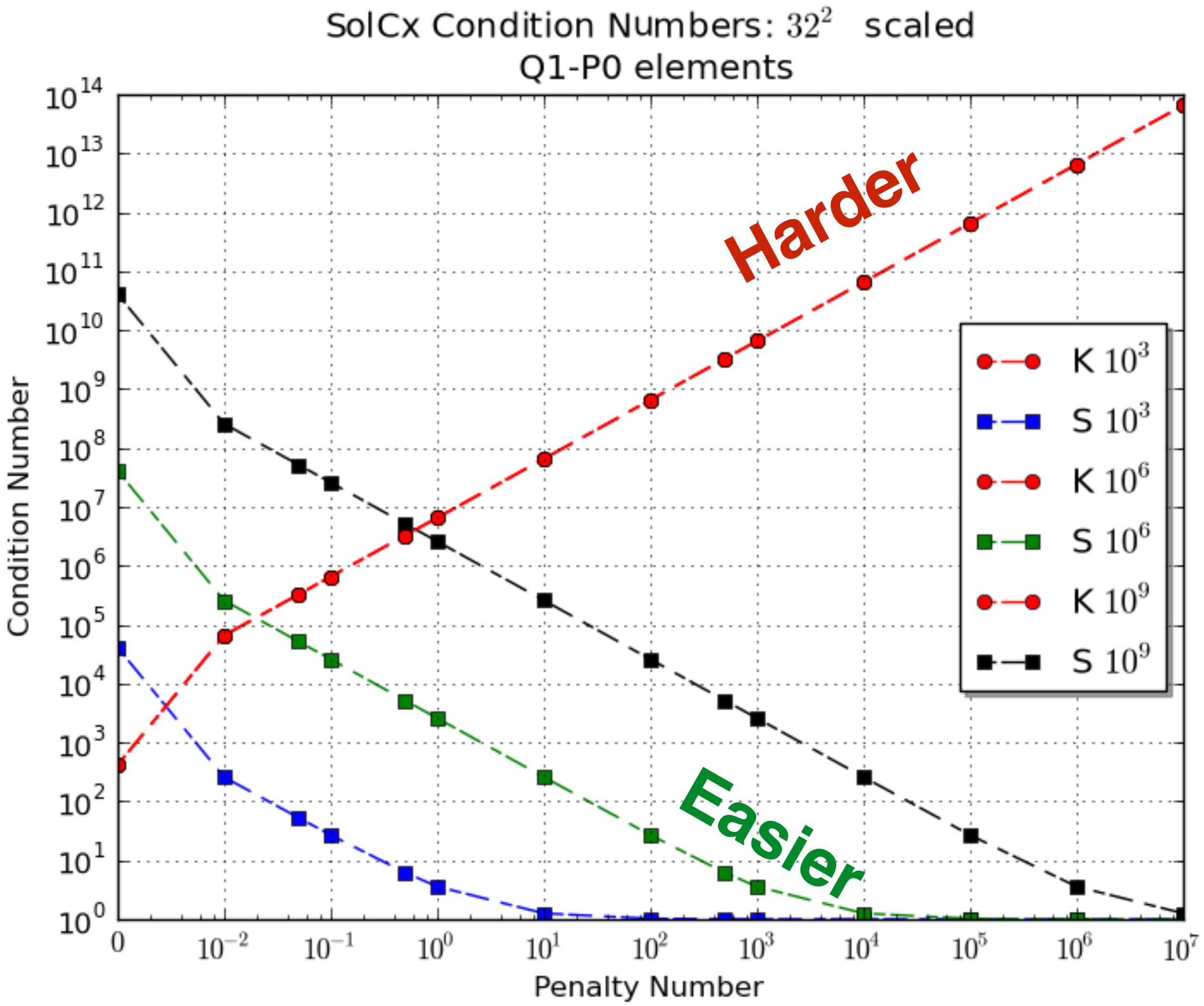
where $\hat{f} := f + \gamma B^T W^{-1} g$, W is symmetric positive definite, and $\gamma > 0$. A good choice of W is the pressure mass matrix, M_p; in practice, we use the main diagonal of M_p instead, in order to maintain sparsity in $A + \gamma B^T W^{-1} B$. The choice of γ is important and will be discussed below.

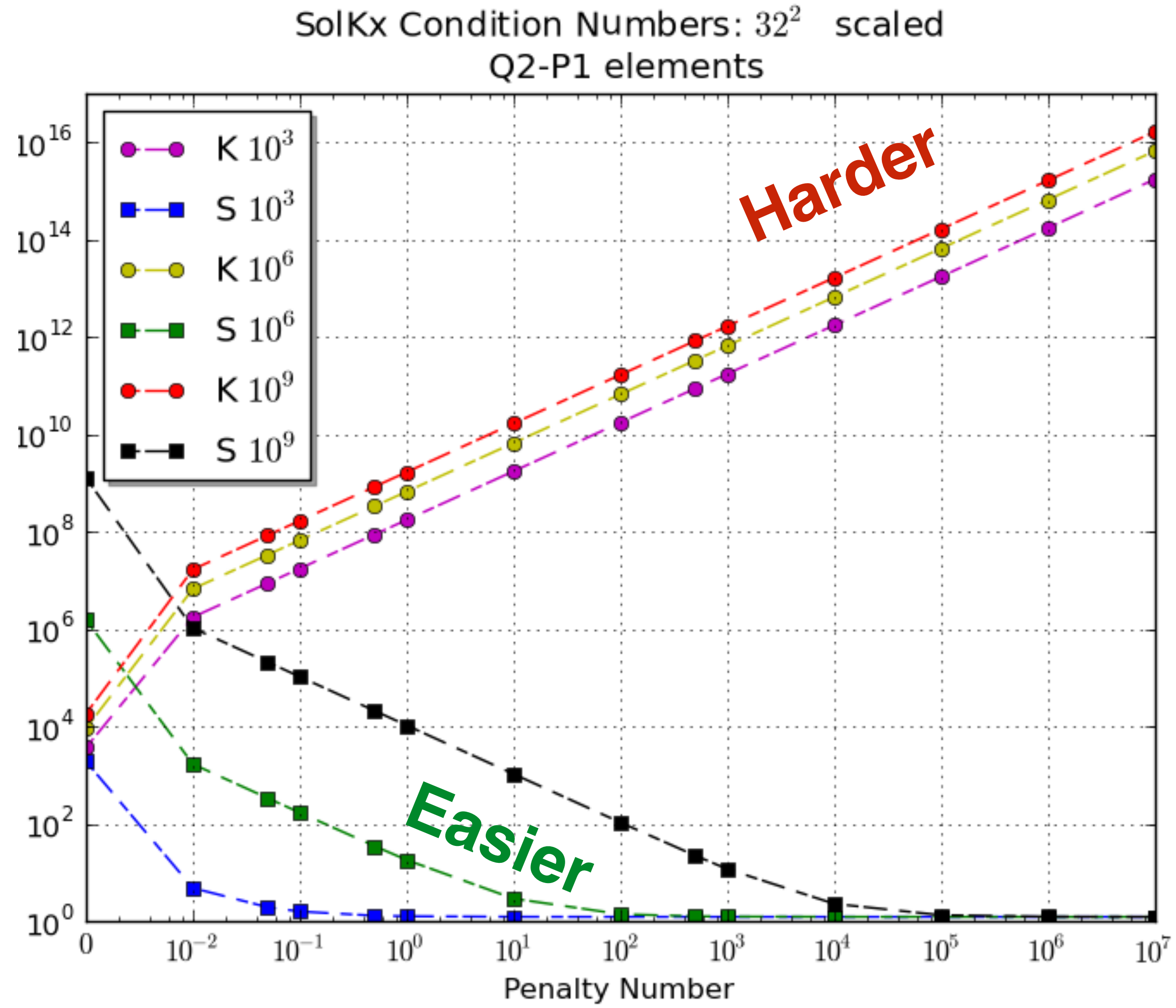
The use of the AL formulation (10) instead of the original one (9) can be justified in various ways; see for instance the discussion in [9, 10]. Here we justify this choice by the observation that preconditioning (10) allows us to circumvent the delicate issue of finding good approximations for the pressure Schur complement $BA^{-1}B^T$ or its inverse, which is crucial when constructing preconditioners for the non-augmented system (9).

Augmented Lagrangian makes the condition number problem go away

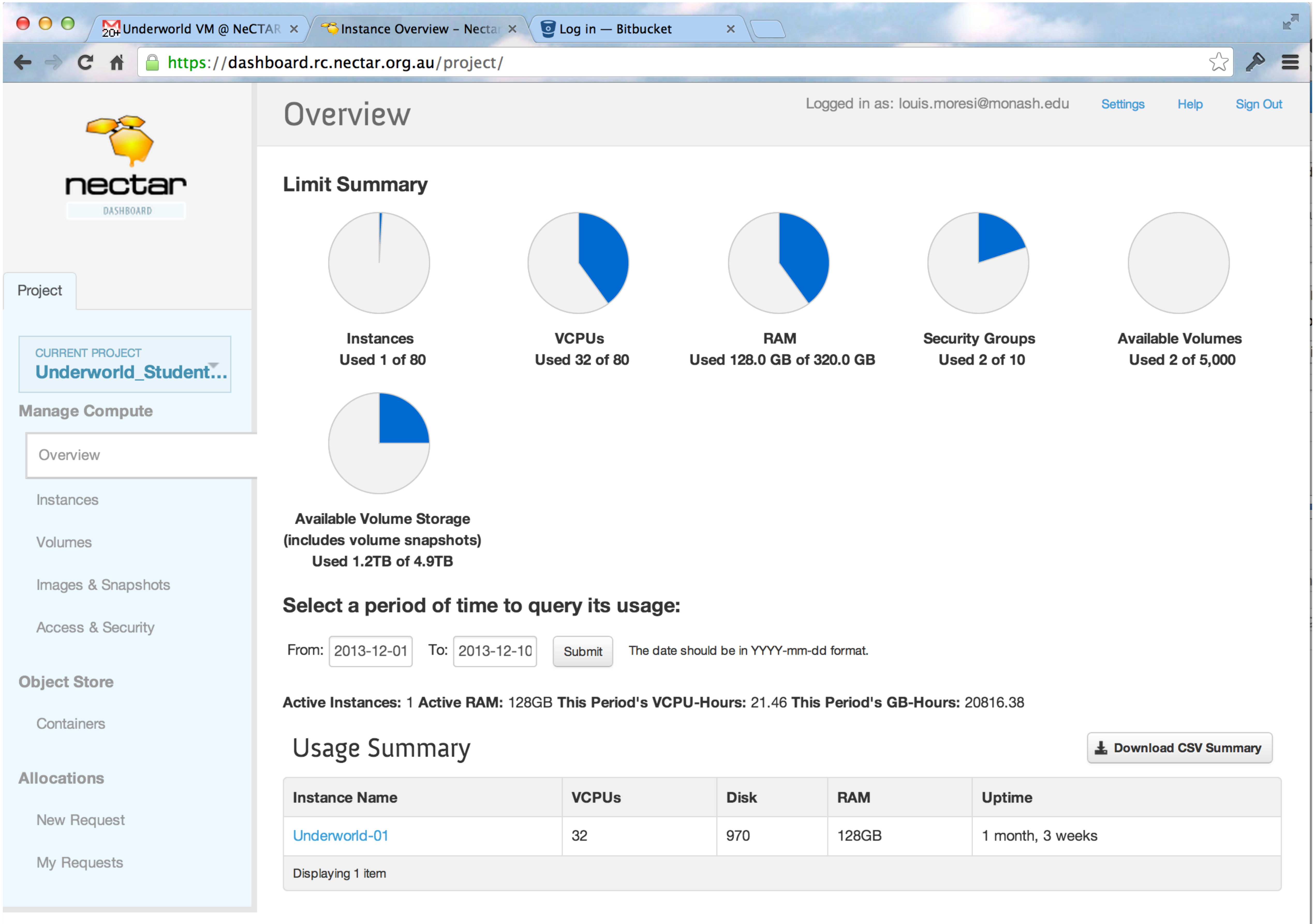


But it's a trade off between the inner and outer problems

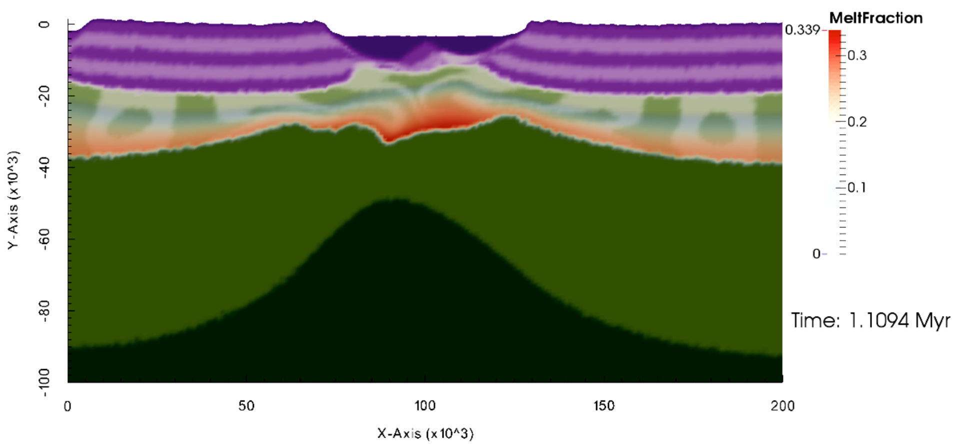


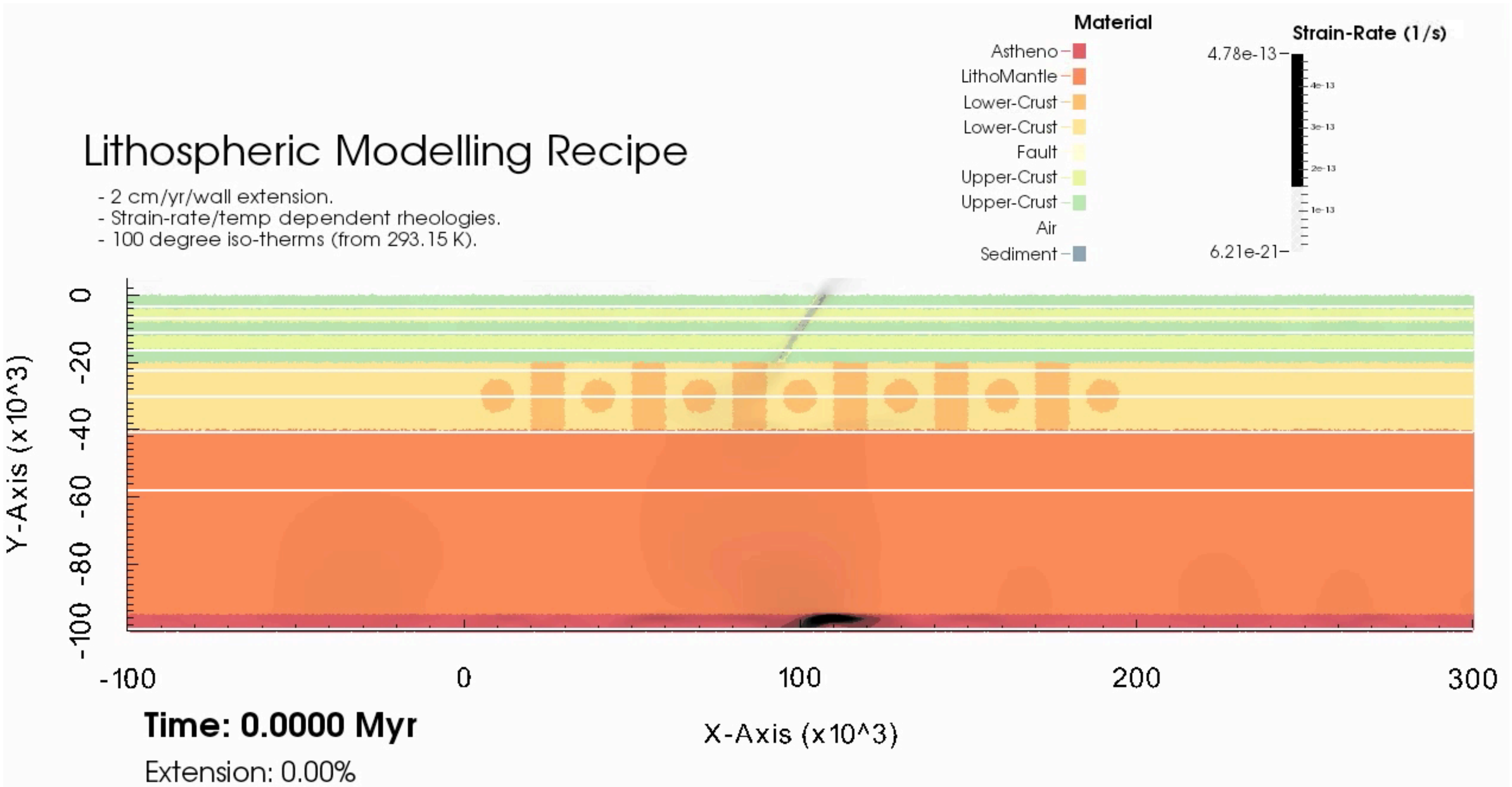


Nectar — Underworld in the Cloud (no installation)

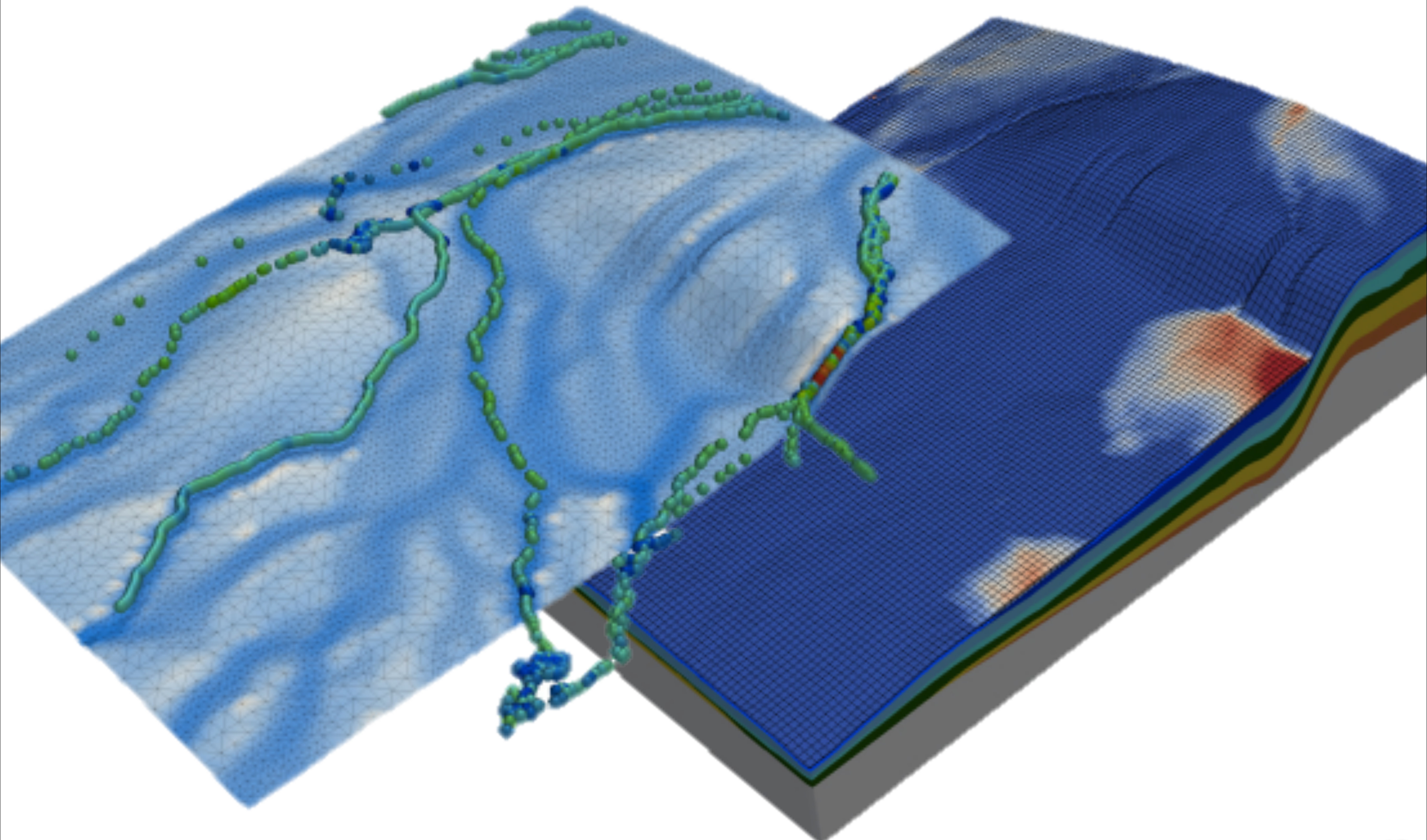


ELLIPSIS modelling toolkit for UNDERWORLD

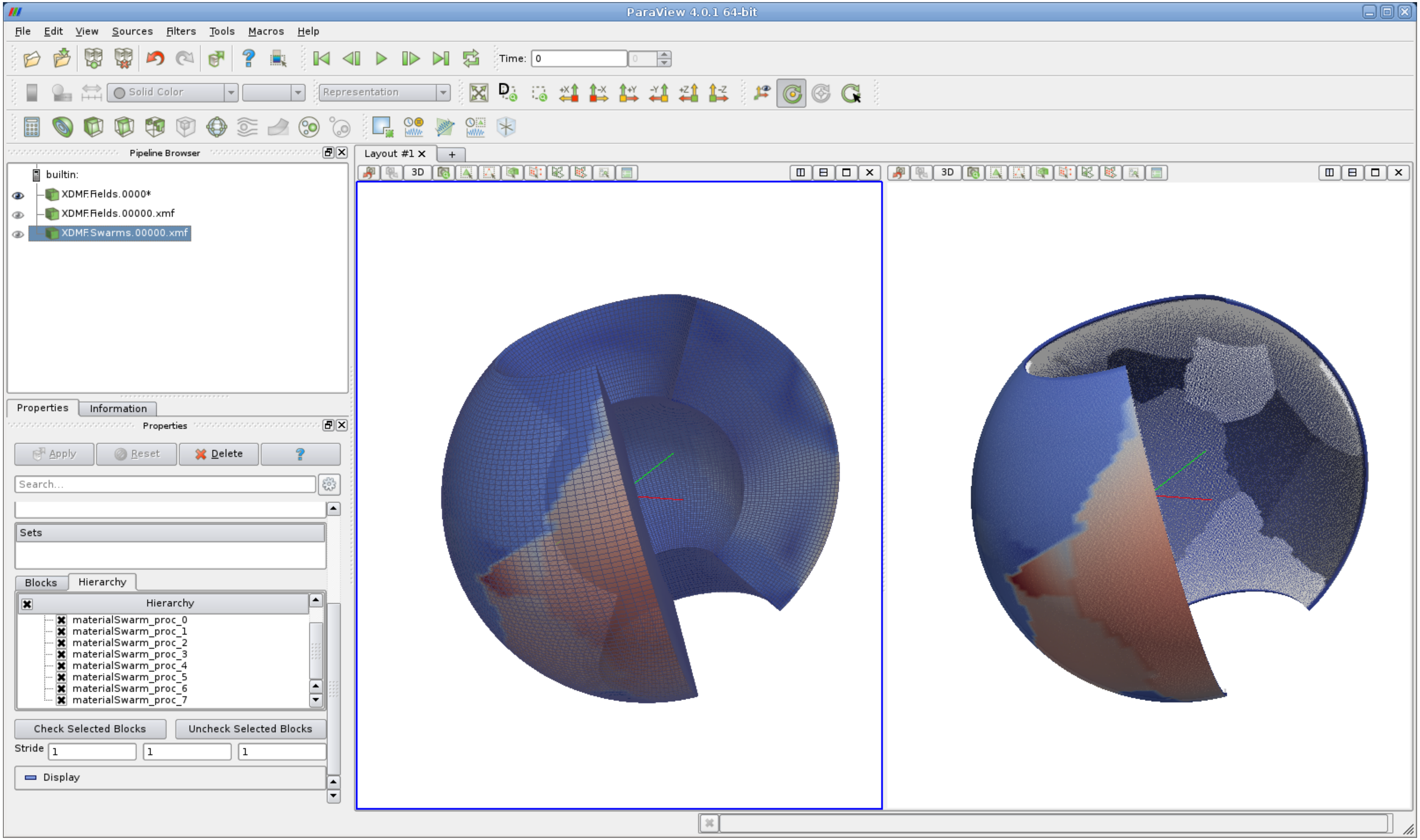




Coupling with LECODE



Underworld and gPlates



Underworld — aims / successes

We wanted to build a code that would be widely used and useful.

Easy to install	✗	✓	
Robust	✓	✓	More important than efficient
Easy to understand and use	✗	?	Better to build something effective
Highly parallel / efficient	✓	✓	
Very general	✓	✓	
Easy to maintain	✗	?	Better to build something effective

Community code

No more heroes ?	✗	?	Why knock the heroes ?
Vibrant community of users	✓	✓	
Everyone open source !	✓	?	Work with everyone !
Linked in to other workflows	✗	✓	