AuScope & Lagrangian-Eulerian consistent AMR

Steve Quenette

Louis Moresi,

Luke Hodkinson & Dave May

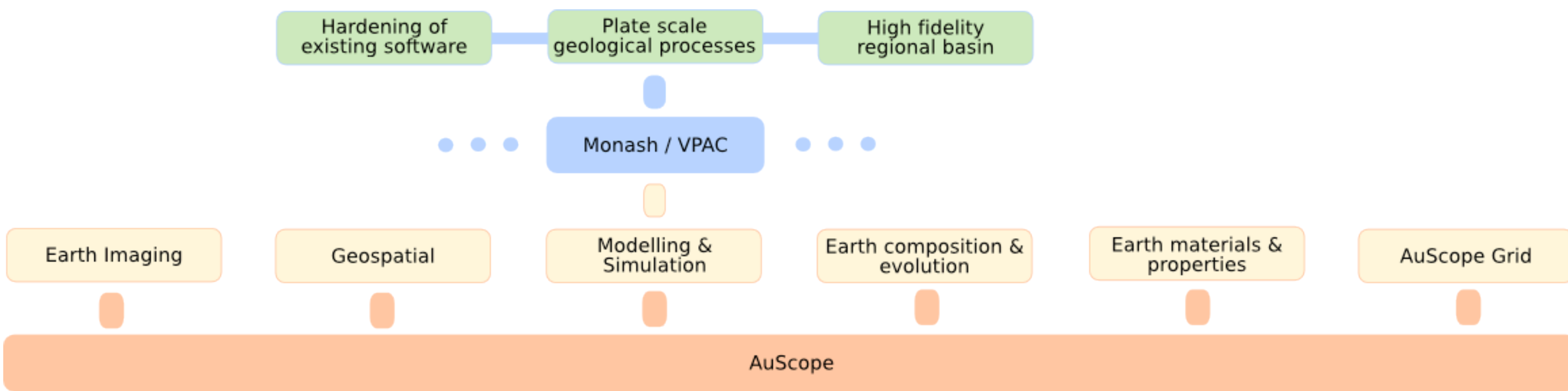
CIG - AMR Workshop October 2007

Outline

- *About us & motivations*
- *Lagrangian - Eulerian consistent AMR*

- *Provide the capability of:*

"Structure and Evolution of the Australian Continent"



- *With respect to us:*
  - *Software as infrastructure*
  - *Fabricating numerical and geophysics research*

Examples of supporting research

Existing: Underworld

- **Aspect**: geophysics
  - Isolated toolbox of rheologies and workflow revolving about Stokes flow
  - Long-term geodynamics - large deformation
- **Target models**:
  - Mantle, slab, basin, plumes, lithospheric, …
- **Targeted numerics**:
  - FEM
  - material point history (PIC)
  - Multigrid

Bleeding-edge: Mayhem

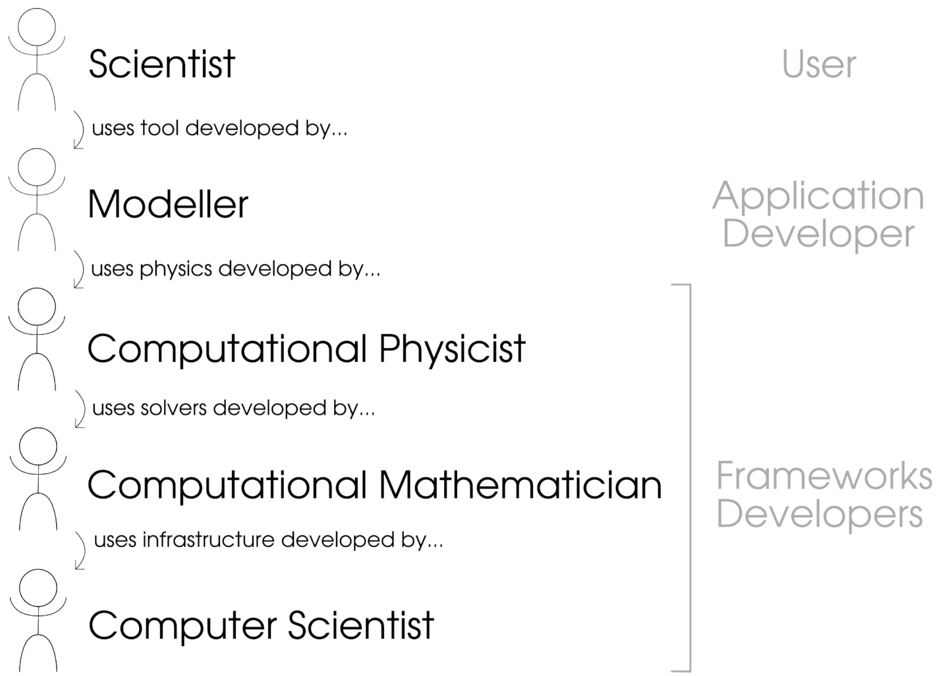
Dave May – dave.mayhem23@gmail.com

- **Aspect**: numerics
  - Research into AMR techniques optimised for Stokes and FEM with material point methods (Lagrangian-Eulerian consistent AMR)
- **Origin**: isolated research code
  - Serial
  - Not applied to involved geophysical problem
- **Objective**:
  - Consolidate into framework
  - plugin into existing phenomena models

Development model

- We're interested in an environment where:
  - Numerical schemes & physics can change with minimal impact on existing phenomena models
    - Hardware proofing (bandwidth, memory models)
    - Enabling multiphysics
    - Enabling scaling
- Our solution:
  - StGermain
    - Aspect oriented
    - "Composition" of phenomena models by isolated numerics and physics
    - Enables layered frameworks & expectation alignment

The language

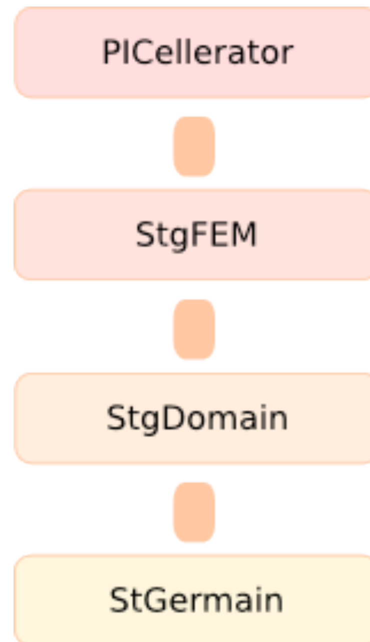



```
<struct name="components" mergeType="merge">

        <struct name="mantleShape">
                <param name="Type">Box</param>
                <param name="startX"> minX </param
                <param name="startY"> 0.0 </param>
                <param name="startZ"> minZ </param
        </struct>

        <struct name="mantleShape2">
                <param name="Type">Union</param>
                <list name="shapes">
                        <param>mantleShape</param>
                        <nparam>weakZoneShape</npa
```

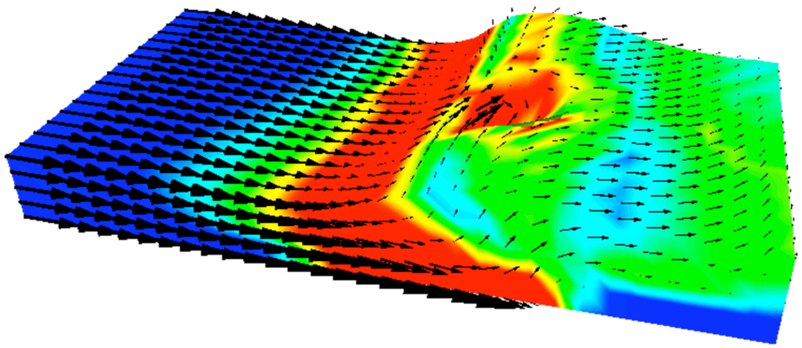
- *Spans C & XML*
- *Abstraction of concepts at all levels*
  - *CP: MoresiMulhous*
  - *CM: Field*
  - *CS: Component*
  - *...*
- *Interchangability*

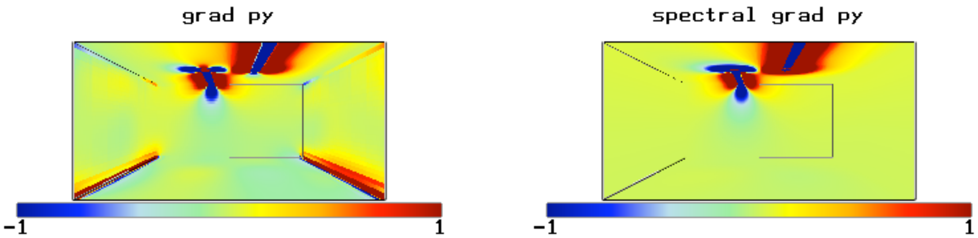
The bain: distributed mem parallel

- Domain & Discretisation
  - Meshing
    - Structured – 1 to 3d decomposition
    - Unstructured - less mature
    - Incidence graph technique
    - Render out to flat arrays (Fortran like FEM)
  - Particles
    - Complicated to optimise
  - FEM
    - Abstract out fields (bundles)
    - (have had versions with optimal numbering)
- Summary
  - Expensive to develop.
  - Years of use.
  - Its all book keeping!



CIG





<table><tr><td>

Existing: GALE

- GALE solves the Stokes and heat transport equations with a large selection of viscous and plastic rheologies.
- *Target models:*
  - orogenesis, rifting, and subduction, …
- *Targeted numerics:*
  - *Underworld (FEM,PIC) + free surface + …*

</td><td>

Bleeding-edge: MADDs

- Explore how magma dynamics interacts with mantle convection and/or long-term tectonics
- *Target models:*
  - *mor, …*
- *Targeted numerics:*
  - *Present thinking… Mixed FEM(PIC)-FV, >= quadratic*

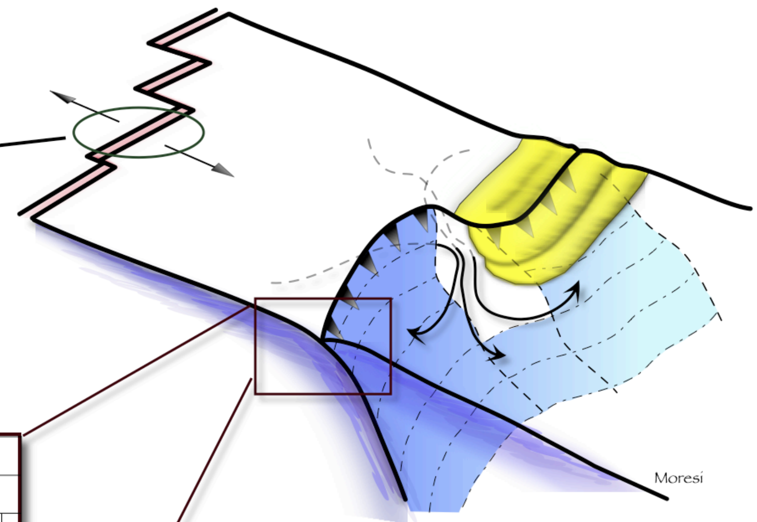
</td></tr></table>

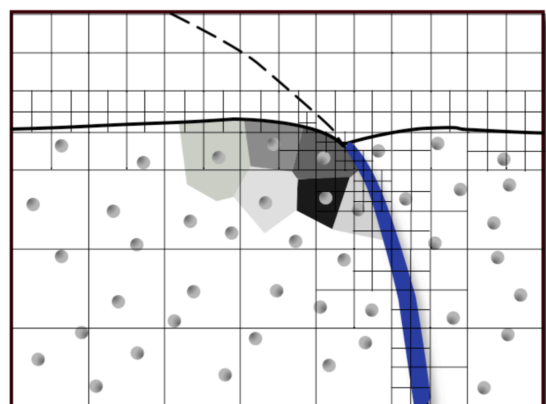
Ok, ok, but what about AMR?

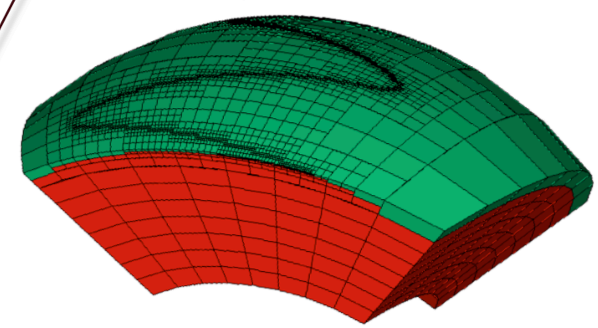
Some geophysical context

- *Either at the scale of:*
  - *Rifting & subduction*
  - *Graven*
- *Embedded within a greater lithospheric & mantle context*
- *Material point vs mesh density at a given point*

$$\dot{\varepsilon}_{ij} = \frac{\tau_{ij}}{2\eta(T, |\dot{\varepsilon}|, \phi)} + \frac{1}{2\eta_s}\Lambda_{ijkl}\tau_{kl} + \frac{1}{2\eta_s}\Lambda'_{ijkl}\tau_{kl}$$



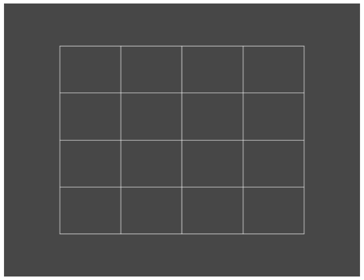


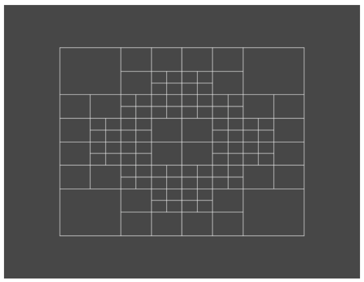


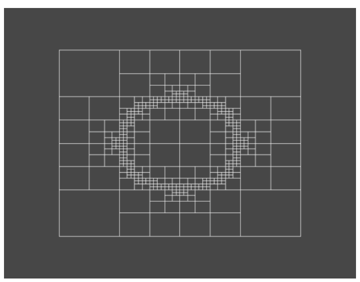
The pieces

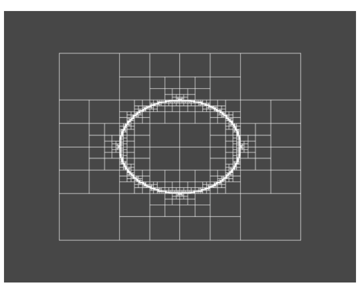
1. Distributed memory parallel meshing infrastructure
   - Mixed tree & flat array based system
2. AMR aware FEM book-keeping
   - Refinement models
3. AMR aware PIC
   - Global Voronoi
4. AMR aware Multigrid, levelsets, …

Status









- *Implemented the distributed memory parallel meshing infrastructure*
- *C example…*

```
int newCells[4];
AdjTopology* topo;
AdjSet* coords;

topo = AdjTopology_New();
newCells[0] = AdjTopology_MakeQuad( topo );
AdjTopology_RefineQuad( topo, newCells[0], newCells );
AdjTopology_RefineQuad( topo, newCells[0], newCells );
```

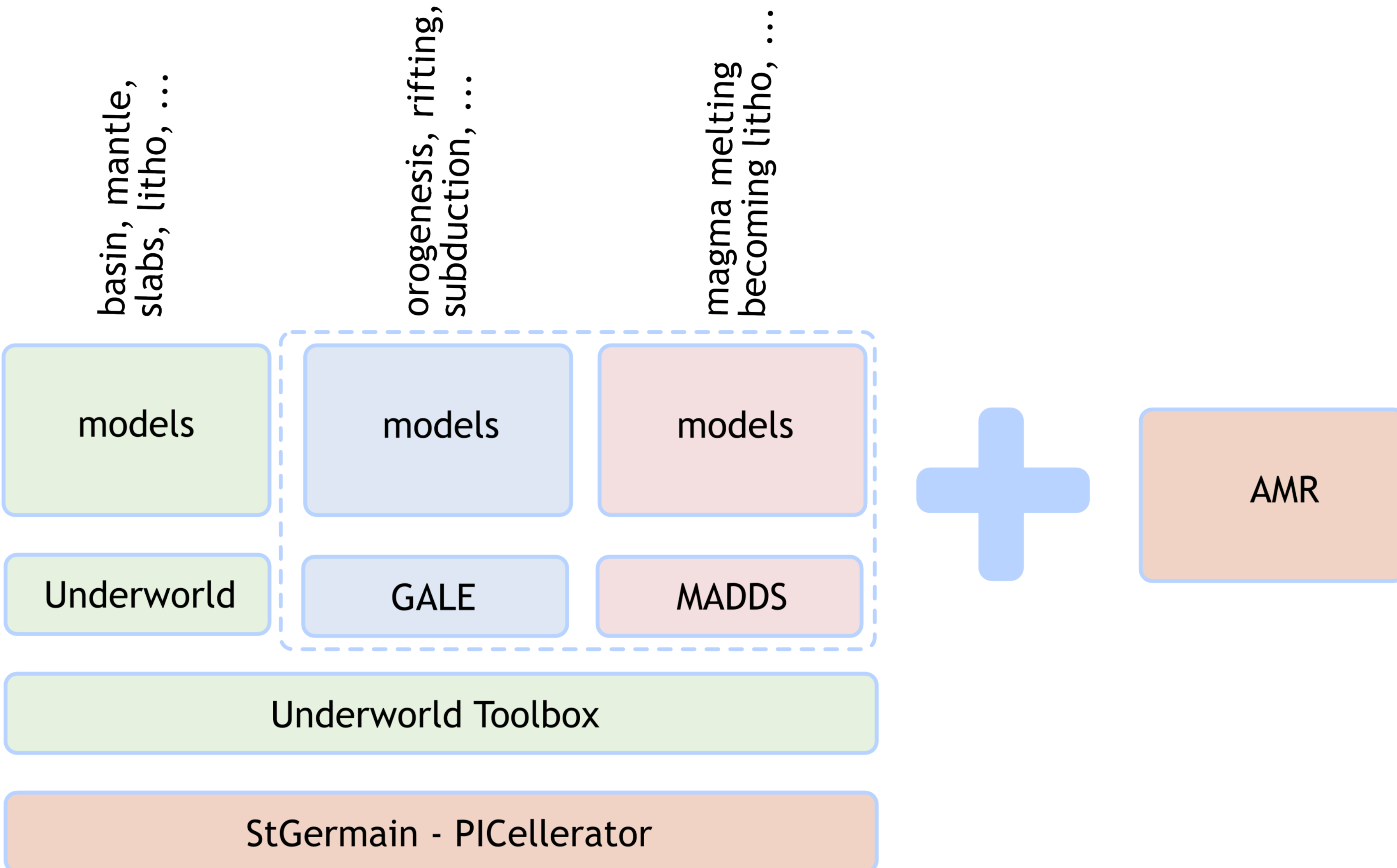
- *From an API that looks like…*

```
int MakeVertex( void* self );
void KillVertex( void* self, int id );
int Lift( void* self, int dim, int nSubCells, int* subCells );
void Unlift( void* self, int dim, int id );
int Join( void* self, int dim, int leftID, int rightID );
```

```
void RefineHexa( void* self, int cell, int *newCells );
void RefineQuad( void* self, int cell, int *newCells );
void RefineEdge( void* self, int cell, int *newCells );
```

```
void Update( void* self );
```

Desired outcome…



Steve Quenette,
Head of CSD