

Computational Infrastructure for Geodynamics

**CIG Planning Document Version 1.0**

November 21, 2003

Please comment on this document publicly by visiting  
<http://geodynamics.org>

1. CIG Announcement
2. CIG People
3. Envisioning CIG
  - a. Open Questions
  - b. Crustal Deformation/Earthquake Simulation
  - c. Coupling surface processes and the lithosphere (in progress)
  - d. Coupled mantle dynamics, magma migration and geochemistry
  - e. Global mantle convection coupled to lithosphere
  - f. Geodynamo with Data Assimilation:
  - g. Geodynamic modeling/Computational Seismology Interface
4. Management and Oversight
5. Working Group Matrix

## **CIG People**

### **Coordinators:**

- [Peter Olson](#) (Johns Hopkin) Geodynamo
- [Mark Richards](#) (UC, Berkeley) Mantle convection

### **Proposal Writing Committee**

- [Louise Kellogg](#) (Chairperson; UC, Davis) Chemical Geodynamics
- [Jeremy Bloxham](#) (Harvard) Geodynamo
- [Michael Gurnis](#) (Caltech) Geodynamics
- [Bradford H. Hager](#) (MIT) Crustal dynamics
- [Marc Spiegelman](#) (LDEO) Magma Migration
- [Sean Willett](#) (U. Washington) Mountain Building
- [Michael E. Wysession](#) (Wash. U.) Seismology

## 2. Announcement for CIG: A Computational Infrastructure for Geodynamics

Purpose:

The purpose of the CIG is to leverage the current state of the art in scientific computing into a suite of open source tools and codes that serve the greater geodynamics community from model developers to end-users of models. The emphasis is on developing toolkits that ease model development and foster the interchange of ideas and algorithms. If successful, this effort promises to revolutionize the way our community computes by providing tools and services that allow us to focus more on the science and less on the nitty-gritty (science-neutral) parts of high performance computation. The goal is to empower a larger community of students, researchers, educators with the tools for quantitative hypothesis testing.

What we are:

The CIG plans to have a small core (~5-7) of dedicated software architects and engineers guided by scientific objectives driven by the Geodynamics community. We will partner with science-neutral activities from the Computer Science/Scientific Computation communities (e.g. SciDac, Pyre, etc) to implement current "best-practices" into community-specific toolkits for scientific computation in solid-earth sciences. This Software Development Team will provide full software service to the community in terms of programming, documentation and full-time support. However, guidance for the programmers will come from working groups (see below) whose emphasis is to identify (and balance) common components needed across the disciplines, with specific requirements of individual disciplines.

What we are not:

We will not be a body that will "anoint" specific models as "community models" or deliver complex "black-boxes" for end users. Given the nature of our discipline, we are well aware of the pitfalls of that approach. Nevertheless, one of the natural products of CIG will be repositories of modular, well-documented and supported codes that end-users can use immediately to test ideas. These models will be anything but black-boxes, however, in that they will allow the user to get under the hood and explore and implement alternative solvers, rheologies and new physics with unprecedented ease and understanding.

Why a center:

The emphasis of the CIG is on the computational infrastructure. However, it is clear from many successful software initiatives that a useful model for efficient code development is to use small teams of programmers working in close

contact with one another. For this reason we propose a Center for Computational Infrastructure in Geodynamics (the CCIG) to house the core group of developers and provide a single point of contact for support, visiting and consultation. The products of this center and community interaction with it will be virtual and distributed on the web. We are committed to an open-source philosophy. If designed correctly, the location of the center will be irrelevant.

Why "Working Groups":

Much of the effort of the center will be in implementing science-neutral coding frameworks into useful toolkits for the entire geodynamics community. The design of the CIG is to support and enable our science not to direct it. Nevertheless, to be successful, this project needs significant scientific input to identify the features of available and future computational infrastructures that are most appropriate for our community. Without a close coupling between scientific needs and computational resources, this project will fail to develop tools that will actually be used. For this reason we envision a set of active working groups representing core constituencies in scientific computing and solid earth science (e.g. mantle convection, geodynamo, seismology) that help set priorities and direction to the software development team. The goal of these working groups is to identify key components that are of general use across the disciplines. In particular, we seek to identify components that allow better integration and exchange of ideas, algorithms and data across traditional discipline boundaries (for example, components/models that would allow ready exploration of the chemical or seismic signatures of various mantle convection models, or tools for multi-scale global/regional mesh interaction). The need for such working groups seems clear although their specific structure and scope is still up for discussion. Thus...

An invitation:

We will participate in the *Computational Geoinformatics Town Meeting* During the Fall AGU meeting on Monday, Dec. 8, 2004 from 7:30 P.M. to 9:00 P.M. followed by a Workshop January 16-17 at LAX to provide the community an opportunity to respond to this manifesto and bring together interested parties from both the scientific computation and geodynamics communities to discuss and design a workable CIG. The goals of the workshop will be 1) to develop a better understanding of the state of the art and opportunities in scientific computation and 2) to discuss fundamental earth scientific problems and identify potential tools that could revolutionize our science and enable us to reach these goals. If you are interested, please remain engaged with the rest of our community, by talking with your colleagues, visiting and posting your views on the web site (<http://geodynamics.org>), e-mailing & calling us, and attending the town meeting and workshop.

### 3. Envisioning CIG

#### *a. Open Questions*

What software will CIG actually produce and maintain? What kind of software system (tool kit) do we envision for the future? How do we break free from 'heroic code' development and create a system of re-usable and module software? What will a tool kit look like which has components which can be more easily maintained, validated, and extended while allowing us to do problems which are we can yet do?

The writing committee has been trying to address these questions and quickly came up against the overall hierarchy of software: the low-level bits which we could potentially refer to as infrastructure, parts which are specific to a geodynamics project, and on top of this would lie code which binds things together called superstructure. This sounds simple enough, but the answers are very individual and depend on many factors, including the kinds of problems which might be contemplated for the future, the way data might be linked to models and the way which models might be linked to one another. On the next several pages, we give some examples of our work.

## b. CIG Crustal Deformation/Earthquake Simulation Flow Chart

The sample problem consists of 3 parts, all of which need to incorporate realistic geometries and material properties. These could be idealized, e.g. for numerical experiments or could be region-specific, for comparison to data (geodesy, strong ground motion waveform, stress inference, . . . .)

- 1) Calculation of stress and strain intersection through the interseismic period – includes viscoelastic relaxation in response to loading by “tectonics” and earthquake displacements.
- 2) Calculating of seismic ruptures resulting from the stress evolution in (1), and forcing (1) with the resulting coseismic displacements.
- 3) Calculating the seismic wave propagation through the system resulting from (2).

Clearly, these 3 problems, each with distinct physics and time scales, would be linked at the Superstructure level. Each of the individual components represents what could also be a “stand-alone” problem that could be of interest in its own right that can also be decomposed into the boxes we are discussing. I label each component as S (Superstructure), G (Geodynamic Specific), I (Infrastructure) and L (Library).

For simplicity, I go through a proposed organizational scheme below for only (1). I have started with the flow chart for GeoFEST provided by Greg Lyzenga, modifying it to fit in my view of the CIG structure.

- a) Overall program control (S)
  - i. Make the code, link the modules
  - ii. Open files
  - iii. Obtain the mesh
  - iv. Get initial and boundary conditions
  - v. Get initial elastic solution
  - vi. Step through time (viscoelastic, etc.)
  - vii. Close files
  - viii. Process results
  
- b) Example: iii) Obtain the mesh
  - Get fault surfaces (G)
    - Existing fault data base (I), or
    - User provided fault geometry (G)
      - Call surface generation routine (I)
  - Get volumetric properties (elastic constants, viscosities, . . . ) (G)
    - Existing properties data base?
      - Regular grid (I), or
      - Irregular grid (I), or
      - Analytic function (I)
    - User-provided representation (G)

Analytic function? (I)  
Register fault surfaces and *volumes* (I)  
    Triangles           *tetrahedral*  
    Rectangles         *hexahedra*  
    Polynomials        *polynomials*  
    Note: any surface should talk to any volume  
Generate coarse model description (I)  
    Surfaces -> closed volumes (I)  
    Mesh (e.g., LAGrit, L)  
Refine mesh (I)  
    Allocate memory (I)  
    Partition (L, e. g., METIS)

- c) Example: v) Get initial elastic solution  
    Call elastic solver of desired type - direct, multigrid, PCCG, . . . (G)  
    Elastic solver assembles and solves problem (G or I, depending)  
        “standard” solvers migrate to I  
        “research” solvers start in G
- d) Example: vi) Step through time, choose solver (G)  
    Loop over time steps (G)  
        Increment simulation time  
        Check for earthquake (S)  
        Choose rheology (G)  
        Solve for displacements and stresses (G or I, as for elastic)  
        Output results, if desired (I)  
            Distributed, centralized, visual, . . . .  
    End loop
- e) Example: viii) Process results  
    Visualization (I)

**c. Coupling surface processes and the lithosphere (in progress)**

d. Coupled mantle dynamics, magma migration and geochemistry

# **CIG components Example:**

## **Coupled mantle dynamics, magma migration and geochemistry**

### **Marc Spiegelman**

**Scientific Overview** A major scientific challenge for both geodynamics and geochemistry is to understand how to use observations of chemical variability to infer dynamic processes in the Earth's interior over the age of the earth. One of the key processes that controls the distribution and composition of observable geochemistry is the production and transport of magma in the earth's interior. Magma dynamics controls both the mixing and fractionation of chemical constituents in the mantle as well as how the solid mantle is sampled to produce the arrays of composition we observe at the surface. Magma dynamics is inherently embedded in the mantle convection as melting is principally driven by decompression and upwelling. Magmatism is also a fundamental feature of plate boundaries and could contribute to the dynamics of plate formation.

**Science Driven Computational Needs:** To couple solid and fluid flow into a testable theory of mantle dynamics requires a theoretical and computational framework that can consistently integrate both large scale mantle convection and smaller scale magma dynamics. From a computational point of view, this requires at least the following components:

1. Multi-scale mesh technology for embedding high-resolution partially molten regions in large scale mantle convection codes.
2. Interoperability with the full range of solid flow solvers used in mantle convection. In particular, these solvers should include a range of rheological models and produce *accurate* dynamic pressures which affect the flow of fluids (the latter constraint is not always easy).
3. Modular codes for calculating fluid/magma production and transport dynamics.
4. Accurate advection and advection/reaction/diffusion schemes for passive and active chemical transport.
5. Interaction with thermodynamic/parameterized models of melting
6. portals to geochemical and partitioning data and databases

**Schematic Examples:** With such a set of modular components, users could develop a suite of customized models to explore a range of problems from the simplest process models to full multi-scale global chemical evolution models that embed high-resolution magma migration into large scale mantle convection models. The following diagrams describe, and attempt to convey graphically such a hierarchy of models.

## Process Model #1: Simple parameterized melting

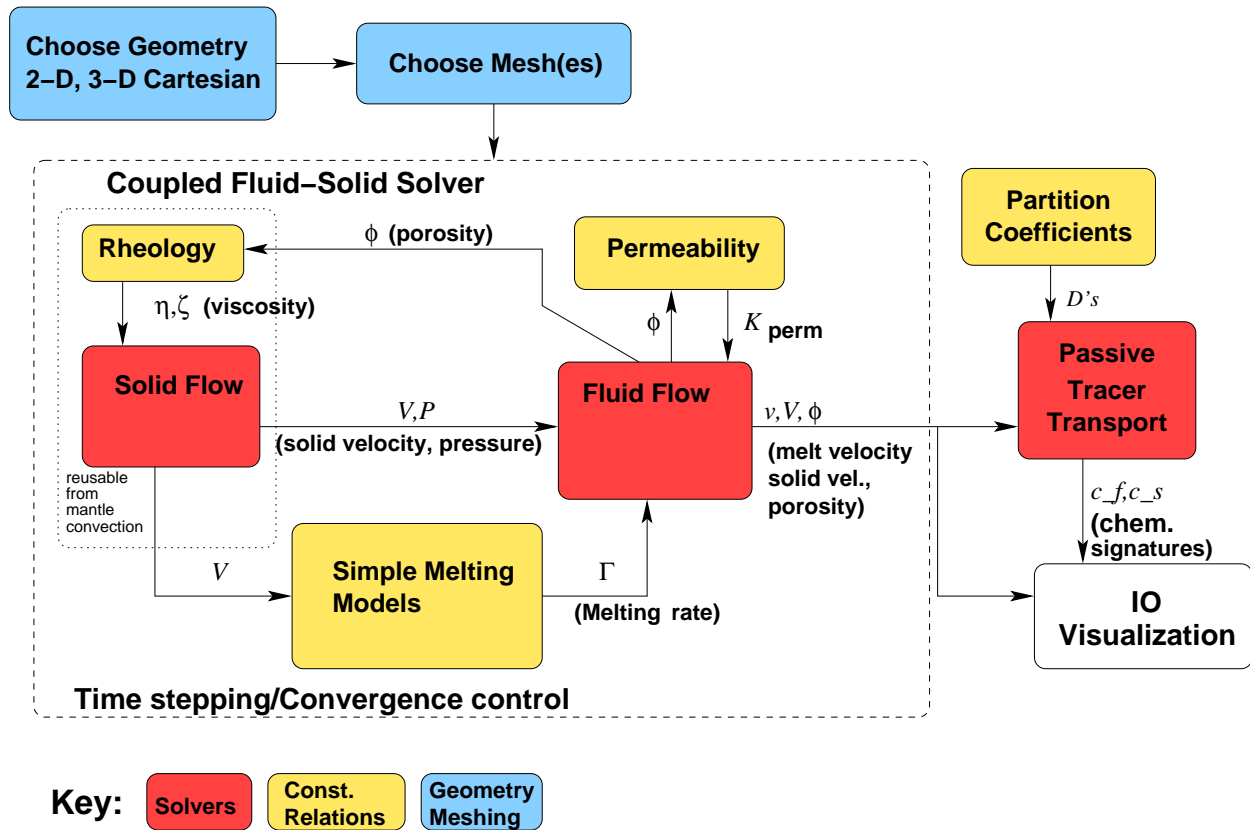


Figure 1: Work flow diagram of a simple model problem to explore the basic physics of coupled fluid-solid flow. Simplified models are essential for developing insight into the physics as well as form a test-bed for developing and testing algorithms and advanced solvers. The model depicted here explores magma-migration in a simple cartesian geometry by coupling an incompressible solid-flow solver (reused from a convection code) with a melt-transport code. In this model, melting is just driven by adiabatic upwelling which only requires knowledge of the solid upwelling field. This model however, allows feedback to solid flow through a porosity dependent solid shear viscosity. An additional tracer model is added to explore the possible geochemical signature of this model.

## Process Model #2: Reactive Flow Solver

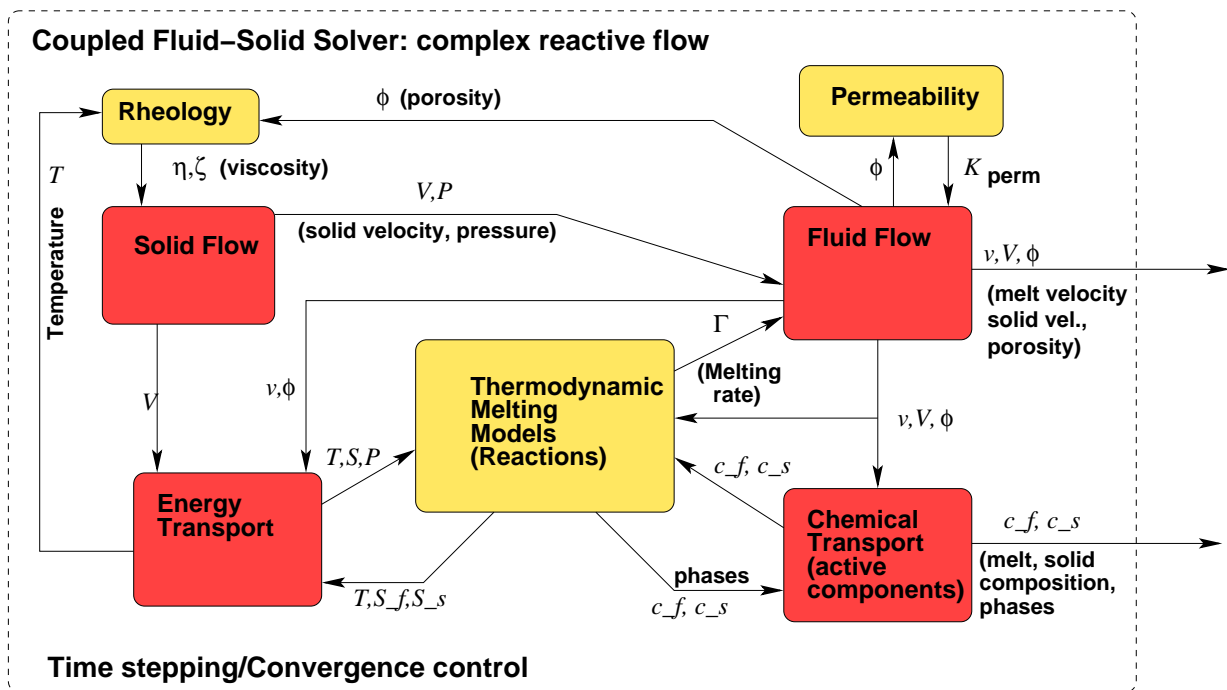


Figure 2: Details of a more advanced reactive flow solver that includes thermodynamic reactions between solids and fluids. The actual algorithm and solver design are likely to be written by the community scientist. The interface design, interoperable framework, support and documentation will come from the core CIG design group to ensure future reusability in other applications.

## Regional Model: MORs, Arcs, Plumes

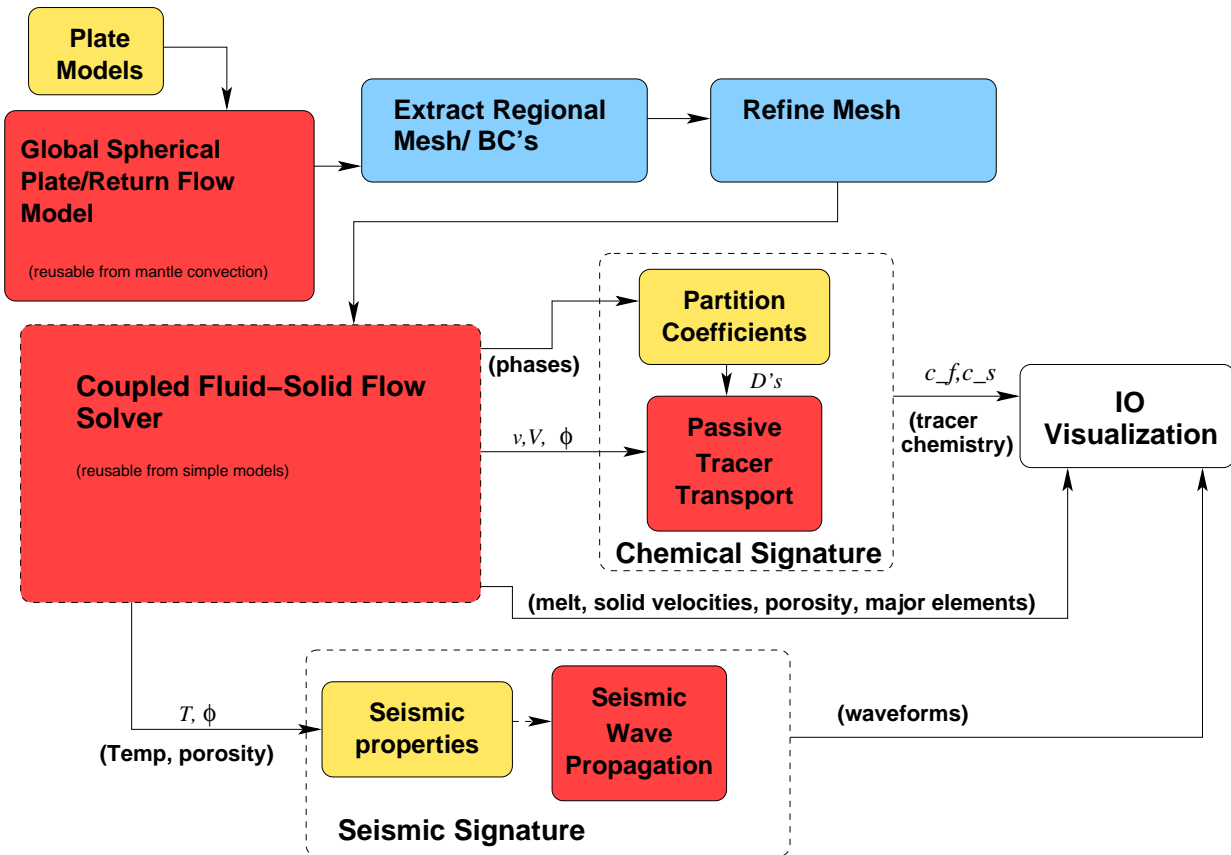


Figure 3: A schematic for a higher level user code for exploring possible geochemical and geophysical signatures of regional mantle dynamics. Such codes could be developed for specific sections of the global mid-ocean ridge system, Island Arcs or mantle plumes. The large scale geometry and coarse scale flow boundary conditions for any region could be extracted from a variety of instantaneous plate driven return flow models (or any mantle convection code). The grid would then be locally refined and local solid and melt flow fields calculated (using one of the coupled fluid-solid solvers developed from simpler problems). Output of these models could then be put into trace element and seismic wave propagation codes to explore their possible observable signatures.

## Global Convection/Chemical Evolution Model

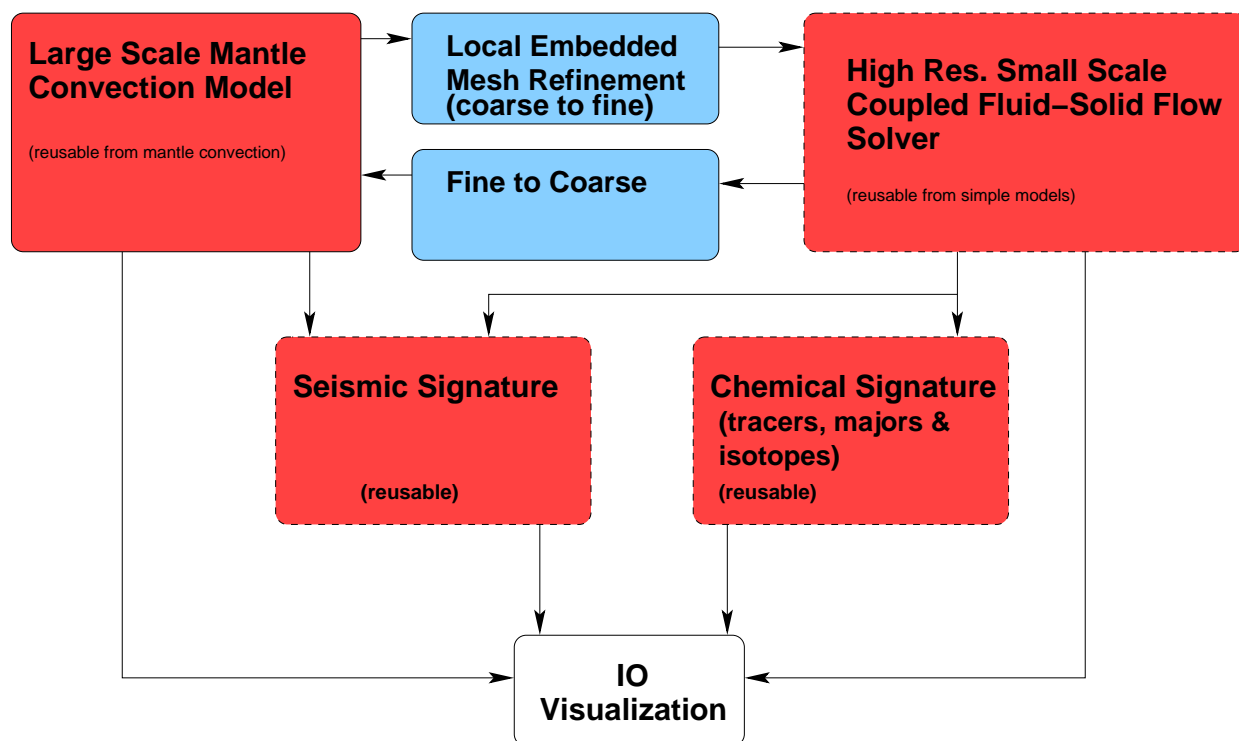


Figure 4: Schematic of a fully dynamic global convection/chemical evolution model. This model would couple a large scale mantle convection code with high-resolution fluid-solid solvers embedded in partially molten regions. Such a code could be used to explore the long-term isotopic evolution of the earth's interior by accurately calculating distributions of solid and melt compositions through geologic time. Such models could then be compared to global statistical distributions of geochemistry.

## **Breakdown of components:**

### **Superstructure**

- Tools for Job initiation and run archiving
- Interface design for inter-module communication
- Standardized IO routines
- Run-time and post-processing visualization
- Integration with External databases (e.g. plate models, partition coefficients, geochemical data)

### **Geoscience Specific modules:**

#### **Solvers:**

- Incompressible and compressible solid flow
- Magma/fluid transport
- Energy/Enthalpy/Temperature transport
- Active and passive chemical transport

#### **Constitutive relations modules**

- solid flow: rheology, viscosity
- fluid flow: permeability
- Thermodynamics: latent heats, reactions/reaction rates, Parameterized melting models
- Tracers: Partition coefficients

#### **Geometry/Boundary Condition modules**

- Global to regional mesh extraction
- Embedded regional mesh generation
- Plate geometry/ return flow models for BC's, IC's

### **Infrastructure:**

- Multi-scale/multi-level meshing
- Inter-grid communication data structures and routines
- efficient parallel linear/non-linear equation solvers (e.g. PETSc, HYPRE, TOPS)

## e. Global mantle convection coupled to lithosphere

Problem statement: Routine set-up and execution of earth models with mantle convection (as a creeping fluid) coupled to either the lithosphere or lithosphere/crust (as a visco-elastoplastic material). The objective is to predict global plate motions (& the degree to which a plate is 'rigid') as a function of time. The model is time-dependent with initial conditions for mantle & plate structure, temperature & composition, and preexisting rheology of plate boundaries at age  $\tau_i$ . However, the conditions of plate boundaries, including their locations, evolution and strength, are subsequently determined dynamically. The initial conditions will come from a 'library' of existing mantle models at different geological ages and surface constraints from a paleo-geographic system. The model is tested against plate motions and plate boundary deformations (from the paleo-geographic system) and present day mantle structure.

### Work & Data Flow

- I. Set-up computational model
  - A. Write own time-loop to sequence interleaving of solvers
  - B. Choose Eulerian spherical mesh (equal area in map view, grid refinement radially) for mantle
  - C. Choose Lagrangian spherical shell mesh for lithosphere, but with grid refinement near plate boundaries at  $\tau_i$ .
  - D. Choose solver for compressible or incompressible flow in mantle.
  - E. Choose solver for visco-elastoplastic deformation in lithosphere.
  - F. Choose solver for heat transport in Eulerian mantle.
  - G. Choose solver for heat transport in Lagrangian lithosphere.
  - H. Choose regridding module that will work with mesh 'C' & solver 'E'.
  - I. Choose (and/or modify) exchanger module which will couple solvers 'D' & 'E'.
- II. Set-up data flow
  - A. From paleogeographic system choose plate boundaries at  $\tau_i$ .
  - B. From paleogeographic system choose ages of oceanic plates at  $\tau_i$ .
  - C. Using plate structure at  $\tau_i$ , establish boundary between lithosphere & mantle (i.e. either at a constant earth radius or base of elastic lithosphere)
  - D. Map different boundaries into faults, regions of existing plastic failure, or damage.
  - E. From 'library' of earth models choose mantle structure (this could be based on tectonic representations, sequence of earlier inverse calculations, etc.)
- III. Solution sequence
  - A. Model starts with internal buoyancy (mantle & lithospheric structure) and preexisting plate boundaries at  $\tau_i$ .
  - B. Normal Stokes flow convection proceeds in mantle with large time step.

- C. Using exchanger, pass tractions (or alternatively velocities) from mantle model to lithosphere model along the interface between the two domains.
  - D. Lithospheric deformation proceeds with imposed tractions (or velocities) on boundaries.
  - E. If necessary, regrid lithospheric mesh.
  - F. Using exchanger, pass velocity (or alternatively tractions) from lithosphere to mantle
  - G. Loop Back to 'B' Stokes flow.
- IV. Post processing
- A. At each time step, determine 'plates' based on surface deformation field
  - B. For these 'plates' determine plate pole and velocity & compare with values in paleogeographic system.
  - C. For non-plate regions compare deformation field to geologic constraints in paleogeographic system.
  - D. From 'C' feed results to collaborators for new field tests, or
  - E. Using mismatches in plate velocities and plate boundary deformations feed results into inversion procedure to extract plate boundary constitutive relation, etc.
  - F. For predicted mantle structure, feed results into MEW's 'Geodynamic-Seismic interface'

# Mantle convection coupled to crust-lithosphere

## Goal: Predict global plate motions & deformations in 'non-plate' regions since 50 Ma

### Superstructure

- Coupler & Exchanger for Eulerian viscous mantle::Lagrangian elastoplastic plate.
- Simulation monitors, incl. visualization.
- Physical constants.
- Portal for plate kinematics, geological strain fields & seismic tomography.

### Geodynamic Specific

- Time-loop & frequency of 'mesh trades'
- Specify IC (plate boundaries and mantle)
- Determine plate poles & compare w/ observed  $f(\text{time})$
- Determine strain field in resulting 'non-plate regions'
- Constitutive eqs. (mantle, lithosphere, crust).

### Infrastructure

- Mesher (Eulerian, Spherical ~equal area)
- Mesh (Lagrangian spherical cap)
- Re-mesher (Lagrangian spherical cap)
- Solver (Spherical, compressible Stokes flow)
- Solver (Spherical, energy eq.)
- Solver (Spherical Shell, elastoplastic)
- Solver (Spherical shell, energy eq.)
- Particle advector
- Particle to spherical-shell mesh

Library: PETSc, MPI.

## f. Geodynamo with Data Assimilation

# Dynamo Code: Earth System Modeling Framework Implementation

Main Program

use Framework\_Module  
Type variables, data structures, etc.  
Input parameters

Initialize\_Framework  
Create\_Application\_Component  
Get\_Layout (*processor elements etc.*)  
Create\_Gridded\_Components

Create\_Grid  
Create\_Distributed\_Grid (*grid distribution over processing elements*)  
Create\_Arrays (*velocity, magnetic field, temperature arrays*)  
Create\_Fields (*associate arrays with the grid*)  
Create\_Bundle (*package the velocity, magnetic, and temperature fields together*)  
Clock\_Setup  
Alarm\_Setup (*to coordinate data assimilation, output, etc*)  
Create\_State  
Initialize\_Gridded\_Components

Dynamo

Register\_With\_Framework  
Get\_Grid  
Get\_Layout  
Create\_States  
Initialize\_States (*for each solver, for data assimilator, etc*)

Module

Timeline

Run\_Gridded\_Component (*Dynamo*)  
Run\_Gridded\_Component (*Data Assimilator*)  
Clock\_Advance  
Output\_States

Destroy\_States, etc.

Destroy State, Grid\_Components, Layout, Application, etc.

Key:

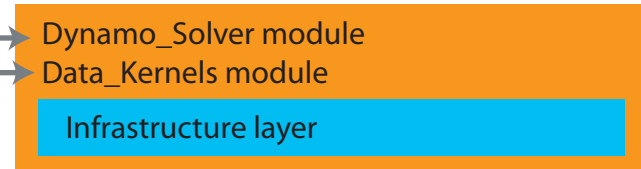
Superstructure

User Code

Infrastructure

Note:

This is where the almost all the user written code fits in.



## **g. Geodynamic modeling/Computational Seismology Interface**

This example involves the capability of taking the output from a geodynamics model of Earth deformation, and using it as the input for 2-D or 3-D synthetic seismogram programs. With the incorporation of actual seismic data, this method could provide a means of forward-modeling Earth structure.

1. Choose or obtain an Earth model.
  - a. The model can be 2-D or 3-D.
  - b. The model can be in terms of a variety of material properties such as elastic constants, temperatures, viscosities, etc.
  - c. Models can also be chosen ad-hoc or by altering previous models to examine certain physical phenomena.
  - d. The model can be intended to represent realistic Earth conditions in the present (or past or future), or can be hypothetical.
2. Convert Earth model into input model for seismic codes.
  - a. Map geodynamic model parameters into geometry for seismic code, regardless of format (mesh type, grid type) of geodynamic model.
  - b. This mesh parameter conversion should be able to work for 3-D to 2-D, and from 2-D to 3-D. There will be instances where a 3-D geodynamics code could be tested with a 2-D synthetic seismogram code along a particular great circle path. There will also be instances where output from a 2-D geodynamics code could be approximated as quasi-3-D for testing with a 3-D synthetic seismogram code.
  - c. Conversion of geodynamic model into density ( $\rho$ ), attenuation parameters ( $Q_\alpha$ ,  $Q_\beta$ ) and elastic constants ( $C_{mn}$ ). There should be a variety of options available for making this conversion, based on a variety of scaling laws between thermochemical and rheological material properties.
  - d. In certain instances, it may be useful because of the benefit in computational time to average the geodynamic model into a radially-symmetric model and test it with a 1-D synthetic seismogram codes like reflectivity or normal mode methods.
3. Generate synthetic seismograms using 2-D or 3-D synthetic seismogram codes.
  - a. Users should have a choice of trying different seismic codes, depending upon the physical problems being investigated.
  - b. For direct testing of Earth structure, synthetic seismic source locations are chosen ( $\theta$ ,  $\phi$ ,  $z$ ) to correspond with known source locations. Arbitrary source locations could also be chosen if hypothetical physical structures are being investigated.
  - c. Set of digital time series (output from synthetic seismic codes) are organized and archived in a database as a function of model type, source location, moment tensor, source time function, earthquake origin time, station name, source-station distance and azimuth, etc.
4. Comparison of seismic data and actual seismic data, in the case of real-Earth investigations.

- a. When realistic Earth structures are investigated, compare synthetic seismic data with actual seismic data.
  - b. Make correction to data or synthetics to allow for comparison (appropriate band-pass filtering, sampling frequency, signal length, instrument response, etc.).
  - c. Waveforms. Use a fitness criterion (such as cross-correlation, arrival times, amplitudes, etc.) to determine the fitness of the synthetic seismic data, in comparison with actual data. This can be done with body and surface waves.
  - d. Secondary seismic data. Compare data and synthetics using secondary seismic data such as mode frequencies and Q values (for multiplets and singlets, in the case where modes are generated using the geodynamic output model), boundary topography of discontinuities, migration stacks, etc.).
5. Use fitness assessments of synthetic data as a selection criterion in choosing geodynamic model boundary conditions and model parameters for forward-modeling particular or general Earth structure.

## 4. Management and Oversight

### *CCIG Management & Oversight* DRAFT ONLY (version #2) -- 10/18/03

A ‘community-centric’ management structure (with similarities to IRIS) is essential to maximize the utility of the cyber infrastructure development and maintenance we propose. As computational geodynamics moves forward, the center needs the expertise, vision, and guidance of the community if it is to remain a nimble and evolving organization. Authority and decision making would be shared between an executive committee, a set of working groups, and an on-site director. Outside advice would be sought from a Scientific Advisory Board. The director would be responsible for the day-to-day management and in turn would be responsive to the executive committee. The center staff would work together closely as a team under the guidance of a senior software engineer. Vision in information technology will be as essential as vision in geophysics and it is for this reason that the center will need a chief software architect.

The Scientific Advisory Board provides advice and input from the larger scientific community. The advisory board will consist of 10 to 12 scientists from educational, non profit, and governmental institutions with a major commitment to research in computational geodynamics and related fields. No more than two members will be from a single institution. Members of the board will be elected by unanimous vote of the Executive Committee. They would hold fixed terms of two years in duration. The composition of the advisory is expected to represent the primary scientific disciplines encompassed by computational geodynamics and closely related fields. The Scientific Advisory Board will meet annually to review the work of the Working Groups and provide advice as requested by the Executive Committee.

The Executive Committee (EC) provides for long the term guidance of the center, scientifically and managerially. The membership of the committee would be composed of a chairperson and regular members equal in number to the number of working groups. We propose that each regular member is the chairperson of a working group. In this way, the executive remains fully integrated with the activities and needs of the working groups and is in a unique position to balance the needs of the different disciplines which CCIG represents. In addition, there would also be two *ex officio* members of the executive committee, the center director and the chief software architect. The most important duty of the executive committee is to weigh the competing objectives and needs of the working groups. Each year, after these needs are determined, it will determine the ideal composition of the software development team in terms of required skills. With this composition of SDT skills, the EC will allocate to the working groups individuals in the SDT or fractional FTE’s with specific skills.

The needs of the different working groups may often not overlap and at times they may be in conflict with one another. Further, it will be essential that the EC weighs needs transcending specific working groups (visualization and software frameworks

being two such overarching needs). It is essential that the EC maintain scientific prioritization in allocation of resources to different disciplines rather than proportionate distribution based upon the number of working groups. Competing objectives of the working groups, the needs of their respective disciplines, and needs which transcend a single working group will all likely evolve over time as new software is conceptualized, developed, and finally disseminated through the community. As time progresses, it is likely that the needs for computational infrastructure within the different disciplines will change and that the composition of respective working groups will need to be revised. It will be the duty of the executive committee to continuously review which disciplines have working groups and to merge, split, add, and/or remove working groups as needed.

The working groups themselves would provide guidance within the specific sub-disciplines of computational geodynamics<sup>1</sup> (including, global geodynamics, computational seismology, crustal tectonics, and the geodynamo). The principal duty of a working group will be to assess the competing objectives and needs of the sub-discipline and then to provide specific instructions to the software development team (SDT). On a year-by-year basis, the executive committee, as described above, will allocate a set amount of resources (FTE's of different skills within the SDT) to the working group. With the chairperson acting as the normal (but not exclusive) conduit to the SDT, the needs of the sub-discipline would be given to the SDT as specific instructions. The members of the working groups would be composed of active members within the sub-discipline. They would hold fixed terms of 2-years in duration, and the chairman of a working group shall not hold two consecutive terms. In this way, a substantial fraction of active members of the community will move through these committees.

---

<sup>1</sup> The writing group is considering the different ways to organize working groups: scientific (e.g. the sub-disciplines of computational geophysics -- the idea which was originally considered), technical, or some combination of the two.

## 5. Working Group Matrix

The writing committee has discussed the possibility of a unique arrangement in which each working group committee member is in two groups: a scientifically themed working group and a technical working group. We make one such matrix of working groups available for initial consideration.

	Meshes/ Grids	solver	Framework/ Superstructures/ Visualization
Geodynamo			
Long-term <sup>1</sup>			
Short-term <sup>2</sup>			

1. Long term refers to long term in the geological sense (mantle convection, magma migration, mountain building, etc.). Seismic tomography, etc.

2. Short is neotectonics, the seismic cycle, earthquake rupture.