2005-2006 Work Plan

Plans and Priorities, 2005

Working group documents defining plans and priorities for 2005-2006.

Proposal for Model Archiving and Querying Facility

Draft of a CIG proposal to provide a model archiving and querying facility.

At the short-term crustal dynamics workshop, one of the priorities that emerged was the need to compare output from physics codes for benchmark problems. Typically, the results are computed with different modeling codes (and output formats), different computational meshes, and different basis functions. We would like to compare computed fields at a specified set of points or metrics of those fields (e.g., norms, RMS misfit).

To provide this functionality, we require tools and methodologies with the following features:

1. A submission and storage (archiving) facility, including the ability to specify metadata describing the submitted results and physics code that generated them.
2. A mechanism for calculating a field from simulation output over a set of points whose coordinates and topology are independent from the simulation. This would allow comparison over data fields over the same set of points from different meshes. The simulation output may involve > 1M points, so the implementation must be efficient.

We would like to develop an initial implementation of this infrastructure for two specific codes presently being used by the short-term crustal dynamics group – LithoMop and GeoFEST. This prototype should provide the basic functionality to permit comparisons between models (i.e., different codes, meshes, and/or basis functions) for the short-term crustal dynamics benchmarks.

We request from CIG:

1. A mechanism to submit and archive results from simulations for benchmark problems. This mechanism should include a method for attaching metadata that describes the code and result to the simulation output.
2. Development of an efficient method for querying simulation results at user specified sets of points. While the queries should honor the basis functions and shape of the simulation code, it should be independent of the original simulation code to allow comparison of results from commercial codes (e.g. Abaqus). It is likely that the FIAT package being developed by Robert Kirby at the University of Chicago could be used to provide the basis function and shape information for any given element type.

CIG working groups will likely also need this tool/methodology for their benchmarking exercises.

rmore, this facility will be useful in evaluating the efficiency of different element types and mesh ions, and in regression testing.

Proposal for a Mesh Preprocessing Package

f a CIG proposal to develop a software package to read output from mesh generation software into a on data structure, partition the mesh, and refine it, yielding a mesh suitable for physics codes running in l.

the major workflow bottlenecks when setting up finite-element models of large, crustal dynamics tions is the process of transforming the output from mesh generation software into a form that the s codes can handle. A related obstacle to simulating large problems is that the mesh generation software se complex geophysical domains runs in serial, thereby limiting the size of problems that can be ted. These problems are encountered in any parallel modeling code that requires an unstructured mesh, e creation of a separate package for mesh importing, partitioning, and refinement would be of enormous to all such codes. The availability of this package would remove the burden of performing these tasks e modeling codes, greatly simplifying their structure and improving the speed of development. The op and EqSim finite-element codes (soon to be merged into PyLith) are examples of codes that would greatly from this package, as the parallel versions will require meshes that are already partitioned and . Other codes, such as GeoFEST, could take advantage of the standardization of a mesh importing even if they do not require partitioning and refinement capabilities.

r to facilitate simulation of crustal dynamics problems in realistic geologic structure using a variety of generation software and modeling codes, we need software infrastructure with the following features:

- The ability to load mesh information from the most widely used mesh generation software (e.g., LaGriT and Cubit) into a common data structure.
- An interface to allow modification of the mesh topology with a specific implementation for inserting cohesive elements into a mesh to create fault surfaces via crack-like interfaces in the interior of a volume.
- The ability to partition a mesh and its metadata, given the number of desired computational nodes. Partitioning would also include identification of *ghost* nodes.
- The ability to globally refine a mesh without degrading the element quality.

requirements of this package, we are assuming a minimal number of features in the mesh generation for the complex geologic structure in crustal dynamics. We assume that the mesh generation software coordinates, element connectivities (nodes in each element), and groups of nodes and elements some ID (integer or string). Some mesh generation software may offer additional abilities, such as the de groups of element faces, and it may be useful for the proposed package to take advantage of such available.

e functionality of this package (object for storing and handling mesh topology) appears to be already Sieve software package under development by Dmitry Karpeev and Matt Knepley at ANL.

of Workflow for Short-Term Crustal Dynamics Modeling

orkflow in short-term crustal dynamics modeling. The different software packages are color coded
ne availability. The CIG effort is focused on the physics codes.

Vorkflow for Short-Term Crustal Dynamics Modeling

2005-2007 priorities

scussion of working group priorities for 2005-2007.