What is CIG

- “Develops, supports, and disseminates community-accessible software for the geoscience community”
- Documented, validated, open-source, state-of-the-art codes
- Framework for code development, interfaces
- Strategic partnerships
- Education and Training

Why is CIG?

- 1950's – 1960's CFD, FEM codes
- 1970's: mantle convection (e.g.) amenable to existing methods → "hero" codes
- 1980's: "supercomputers," vector machines → hero codes on steroids
- 1990's: parallel supercomputers
  - Legacy code re-engineering
  - Need for fundamentally new codes

Why is CIG? (continued)

- By 2000's, hardware capabilities and advanced methods from computational science (applied math + computer science) began to outrun the "heroes"
- BUT, 1980's mantle convection workshop (e.g.): 2-D benchmarks also revealed deeper problems (basic physics, accuracy, validation)
- Software engineering primary limitation, NOT hardware speed, in many applications

Focus on Graduate Education

- Ex 1: here's this code…..
- Ex 2: go to LANL and work with so-and-so….
- Ex 3: why don't "we" write a new code…?
- PROBLEMS:
  - Reinventing wheels, spark plugs, pistons….
  - Validation, documentation, maintenance, upgrade
  - Educating students/PI's in CS + geophysics
  - FUNDING

Impasse (2000)

- Only solution must be community-based
- Major funding required, relatively high risk
- Failure of NASA/HPCC → benchmarks not effective
- Olson/Richards drink a lot of wine….

  → *Could the community agree on the need for a cultural change????*

Timeline for CIG

- 2001: draft white paper (shopping at NSF)
- July 2002: Granlibakken conference (boost from seismologists)
- March 2003: White paper submitted to NSF
- 2003-2004: community outreach, “center” proposals
- Sept. 2004: CIG proposal, Pasadena base

A Surprisingly Ambitious Concept

- CIG success would fundamentally alter the landscape of geophysics, earth science, and perhaps other similiarly defined communities of scientists: Contrast with
  - Climate modeling (community models)
  - Molecular dynamics (kingdoms)
  - Particle physics (accelerator-based)
  - Engineering (empirical orientation)
  - Astrophysics (cowboys)
- Time and patience required. Think of this as our LHC, drill ship, or genome project. In all these examples, the science/technology was probably more straightforward than the community organization.

CIG as an experiment in community self-organization

- It is very hard to change a culture that views itself as having had a successful history
- Surprises, unanticipated angles:
  - Crust vs. core community interest
  - Community building (skeptics, workshops, need for concrete examples)
  - Heroes (good) vs. hero codes (bad)
  - Physical vs. virtual center
  - 1-D geophysics surprisingly interesting – has little at all to do with high-performance computing!!

Where are we?

- Strategic Plan:
  - Software repository
  - Software framework
  - Organizing community participation
  - User training
- Transition from code re-engineering to new code development
- Community “ownership” increasing (workshops, codes)
- Ongoing concerns with representation, diversity
- Very Spartan operation: OK for now, but will require increased funding if perception of success continues to grow

This Meeting

- Friendly, frank exchange, not a dog-and-pony show
- Aim for understanding of deep issues
- Embrace CIG as a community/NSF partnership