Building a Geoscience Repository & Framework

Bill Appelbe, bill@vpac.org

Director, VPAC

(Victorian Partnership for Advanced Computing)

Outline

- VPAC – an interdisciplinary R&D partnership
- VPAC's geoscience projects

1. Modernise the state Departments of Primary Industry (DPI) and Sustainainabiiy and Enviroment (DSE) IT infrastructure
2. Build a geoscience software repository with CalTech
3. Build a software framework for the Access MNRF & US geoscience

VPAC

- A independent company owned by Victorian Universities
- Advanced Computing R&D centre
  - Collaborative, interdisciplinary R&D
    - national, and international
    - government, industry, and academia
  - Commercial, sustainable focus
- Opportunistic focus areas:

1. Geoscience
2. Bioinformatics & Life Sciences
3. Computational Engineering

VPAC Resources

- Staff
  - About 25 scientists and engineers
  - Multi-skilled, multidisciplinary; co-located
- IT
  - High Performance Computing (500+ CPUs)
  - Storage (10Terabytes by end of 2003)
  - High-bandwidth interconnection - GrangeNet
  - Visualization – collaborative with III VR Centre
- Opportunistic focus areas:
  1. Geoscience
  2. Bioinformatics & Life Sciences
  3. Computational Engineering

VPAC Key Drivers

- Focus on collaboration, collaboration, ...
  - Project/outcome driven
  - Partner with organizations in win-win projects
- Current major collaboration
  - Holden/GM
    - Innovation Centre; $10M p.a
  - IBM & Vic. Department of Primary Industry
    - Modernization of HPC, software support; $4M
  - Victorian Universities & U. of Queensland
    - ACCESS MNRF; $15M
  - Caltech & USA NSF & ARPA
    - Frameworks for reusable/adaptable scientific software

The DPI/DSE Project – IT Modernization

- Victoria invests heavily in state funded R&D for primary industry & environment
  - Mostly a “silo/desktop” mentality
    - Compartmentalized divisions in Forestry; Fisheries; Land Catchment, Conservation and Planning
    - Limited data archives/modeling capacity
- IT Modernization Project
  - Collaborative 3 year contract with IBM, RMIT’s I3
  - Install large 64/32 ***grid cluster***, managed storage, upgrade software, parallelize applications, cleanup data (fisheries), ...

The Geoscience Software Repository

- R&D in geoscience is hampered by poor software quality
  - Mostly “hero codes”, written by PhDs
  - Poorly documented
  - Hard to maintain, adapt, or extend
- Other scientific disciplines do NOT have this problem to the same extent!
  - Chemistry, meteorology, physics have well-developed and supported ***community codes***
    - NWChem, NAMD, MOM-3, etc.
  - Bioinformatics is “open source” driven

The Geoscience Software Repository (cont.)

- Why is geoscience software so far behind?
  1. Little government imperative to fund it
  2. Commercial software for seismic data processing very tightly held
  3. Geoscience R&D Community is very fragmented and silo-ised
  4. Geoscience physics models are poorly understood and coupled
     - No “Schrödinger's equation” of the solid earth

Solution to the Geoscience Software Quagmire

1. Build national and international collaboration
2. Develop an “international geoscience software repository”
   - Of well-documented and tested popular geoscience codes
   - Open source, to encourage improvement
     - Mostly free for academic use
3. Develop codes that are far easier to adapt and interface to allow model experimentation
   - A “software framework” for geoscience model development and integration

Solution Step #2
The Geoscience Software Repository

- It is now there in beta form
  - www.geoframework.org
  - Joint collaboration of Caltech and VPAC
  - More codes being added gradually
    - `Ellipsis/Citcom` – mantle/lithospheric modeling
    - `SpecFEM` – seismic modeling
    - `Snark` – mantle/lithospheric modeling
    - `FLAC3D` – small scale lithospheric modeling
    - `CascadeII` – surface processes: erosion/transport

Solution Step #3

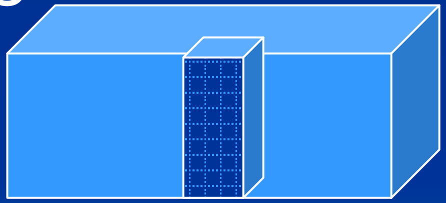
Software Frameworks for Geoscience

- There are two different needs for geoscience software model adaptability

1. Interface adaptation
   - Tie models together – model embedding
   - Simplify parametric model runs
     - Setting model parameters, boundary conditions, etc
2. Solver adaptation
   - Changing the internals of a model
     - rheology, solver technology

Solution Step #3
Interface Adaptation

- Most geoscience software models have inflexible Neanderthal interfaces
  - Set parameters by modifying the code, undocumented configuration file formats and switches
  - The software is unusable without an apprenticeship
- Geoscience models need embedding
  - High-resolution models (e.g., fault zones) embedded in low-resolution models (e.g., lithospheric plates)



  - You cannot put a rectangular box around a piece of crust!

Solution Step #3
Interface Adaptation

- In collaboration with Caltech we have developed “wrapper” libraries to simplify interface adaptation
  - `Pyre/Pythia` - written in Python
    - Open source, available in beta-form from VPAC or Caltech
  - Just replace a geoscience model Fortran or C ‘main program’ by a wrapper written in Pyre/Pythia
  - Now models can be plugged together, and use various data I/O formats
    - XML, textfiles, command-line options, embedded models

Solution Step #3
Solver Adaptation

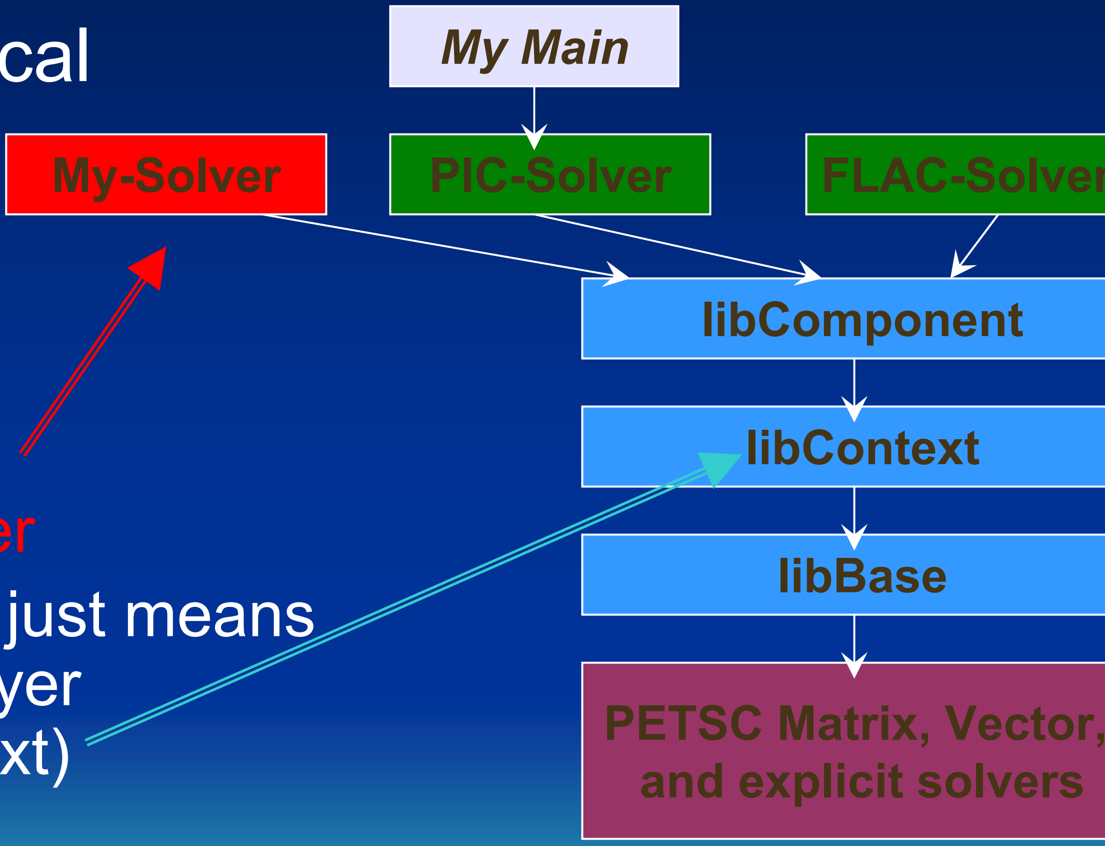
- Most scientific models boil down to a bunch of Partial Differential Equations (PDEs)
- Numerical analysts have developed countless techniques for solving these
  - Implicit (matrix solvers)
    - preconditioners, multigrid, ...
  - Explicit
    - grid, particle, ...
- In a community code, for well-studied PDEs, we can develop an “optimal solver”
  - This approach does not meet the geoscience need to tinker with both the PDEs and the solution methods

Solution Step #3
Solver Adaptation

- The solution is to develop ***scaleable parallel*** "building blocks" or "modules"; a ***framework*** that can be used to
  - Develop new solvers
  - Customize an existing solver written using the framework
- A beta version of this open source framework is available from VPAC, called ***StGermain***
  - Being used to build two production solvers for www.geoframework.org
    - `Snac` – a FLAC style explicit solver
    - `Snark-II` – a Particle-In-Cell combined explicit/implicit solver

Solution Step #3
Solver Adaptation - StGermain

- Layered/hierarchical oo-component architecture
  - Like the TCP/IP protocols
  - A new solver just means a new layer
  - A modified solver just means tinkering with a layer (usually the context)



Conclusion

- Geoscience is a real software challenge
  - R&D software models
  - Data set management and curation
  - Limited industry/government funding
    - where is the “geo-genome” project?
- We believe it needs collaboration of multi-skilled teams on focused international-scope projects