CESM/CISM Software Engineering Update

Bill Sacks (NCAR)
Land Ice Working Group
Software Engineering Liaison

In collaboration with
Bill Lipscomb, Jeremy Fyke & Steve Price (LANL)
Mariana Vertenstein (NCAR)
Tony Craig & Jon Wolfe (consultants)
Two-way Feedbacks Mostly Complete

CAM
(Community Atmosphere Model)

CLM
(Community Land Model)
10 glacier elevation classes

CISM
Surface Mass Balance
Surface Temperature
Icesheet Area
Surface Elevation

CICE / POP
(Sea Ice) / (Ocean)

Ice Calving
What's Still Needed for Dynamic Landunits in CLM

- Water & energy conservation
  - Basic code in place. Needs scientific review and some tweaking.
  - May be fundamentally reworked to conserve without the need for fictitious fluxes.

- Carbon & nitrogen conservation
  - Prototype code written; need to plug into CLM.
CISM2 in CESM

• CISM2 release version now in CESM

• Multi-year Greenland simulations give reasonable results
  ▸ CISM-only (TG compset)
  ▸ Fully coupled, with two-way feedbacks (BG1850C5L45BGCIS2 compset)

• Still needs tweaking of configuration settings & initial conditions
Moving Remapping into the Coupler

Currently: Remapping happens in CISM's glint package

Limitations

- Only works with regular lat/lon land grids
  - Would not work with CAM-SE grids
- Bilinear interpolation – not conservative
- Mapping happens in serial
- Any alternative ice sheet model (e.g., MPAS - Land Ice) needs to reimplement glint
- Ocean – land ice coupling would have to be done via the land grid

Solution: Move remapping into the CESM coupler

Complete on branches, will come to the trunk in a few weeks
Turning on CISM by Default in CESM2

• Most CESM runs do NOT include CISM or the calculation of surface mass balance

• We would like to include CISM as a diagnostic component beginning with CESM2 and all CMIP6 runs

• Some prerequisites:
  ▶ Coupling rework described earlier
  ▶ Variety of other rework to make it possible to run CISM with any CLM resolution
  ▶ Get CISM building with NAG compiler
Current Coupling

CLM: Fields passed on land grid, in multiple elevation classes

CICE: Fields passed on land grid, in multiple elevation classes

CISM: Regridding occurs within CISM (glint)

CAM

POP

Image credit: [http://www-personal.umich.edu/~paullric/research.html](http://www-personal.umich.edu/~paullric/research.html)
Moving Coupling into the Coupler

- CLM: Fields passed on land grid, in multiple elevation classes
- CICE
- CAM
- POP
- CISM: Fields passed on ice sheet grid

Regridding occurs within coupler

Image credit: http://www-personal.umich.edu/~paullric/research.html
Other Near-term Plans

• Allow deeper snow pack

• Put in a fix for the snow radiation absorption problem in Antarctica

• Develop a data GLC model, allowing simulations with prescribed transient glacier areas

• Generate new TG forcing datasets

• Improve CLM's diagnostic output capabilities
Development with Unit Tests

Leverages new unit testing framework in CESM

- Uses pFUnit
- CESM infrastructure developed by Sean Santos

\[ \beta_{ik} = \frac{b_{i,k+1} - b_{i,k-1}}{h_{i,k+1} - h_{i,k-1}} \]

```fortran
@Test
subroutine test_calc_vertical_gradient_ECmid(this)
    ! Test calc_vertical_gradient with an elevation class in the middle of the range
    ! (standard case, not an edge case). This uses a single grid cell.
    class(TestVertGradCalc2ndOrder), intent(inout) :: this
    type(vertical_gradient_calculator_2nd_order_type) :: calculator
    real(r8), parameter :: topo(1,3) = reshape([50._r8, 125._r8, 275._r8], [1,3])
    real(r8), parameter :: data(1,3) = reshape([11._r8, 12._r8, 13._r8], [1,3])
    real(r8) :: vertical_gradient(1)
    real(r8) :: expected_vertical_gradient(1)

    calculator = this%create_calculator(topo=topo, data=data)

    call calculator%calc_vertical_gradient(2, vertical_gradient)

    expected_vertical_gradient(1) = (data(1,3) - data(1,1)) / (topo(1,3) - topo(1,1))
    @assertEqual(expected_vertical_gradient, vertical_gradient, tolerance=tol)
end subroutine test_calc_vertical_gradient_ECmid
```