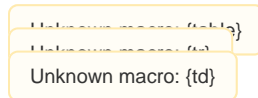


CLM Fine Mesh Project



CLM Fine Mesh Project

!! Goals

Implement a fine mesh capability in CLM3. This will let CLM run on an independent grid. Provide capability for downscaling and upscaling between the coarse external grid and the fine internal grid.

- Phase I: implement infrastructure, simple "copy-only" between coarse and fine grids – **COMPLETE** – Phase IIa: implement downscaling and upscaling – **COMPLETE** –
- Phase IIb: implement nested grids – **DEFER** –
 - !! Process
TC implements/tests. MV guides/reviews. GB and AH provide guidance, reviews, and clarify requirements as needed.
 - !! Requirements
 - works with clm standalone, cam/clm, ccsm.
 - bfb or roundoff if fine mesh and coarse mesh grids are same
 - supports coarse grids that can be defined via four corner points with only two unique latitudes and two unique longitudes including reduced grid. clm3 will support only regular lat/lon fine grids. both grids must be on the same coordinate system. unstructured or nested grids should be considered during implementation but implementation is deferred to phase IIb.
 - coarse and fine grids are generated external to clm3 and are read.
 - coarse/fine mapping must conserve water, energy, mass.
 - implementation is consistent with current clm3 design. minimize code changes. code is well commented. code provides useful diagnostic output
 - user's guide, etc are updated as needed
 - external tools (grid generation and surface dataset generation) are consistent with clm needs (netcdf, fields, fill, etc).
 - meets current portability requirements. performance acceptable.
 - history file writing required for fine mesh only
 - consistent with current and future coupling interfaces (CCSM contract, CLM/CAM Ipcoupling, CLM/CAM MCT rearranger, CCSM/ESMF/MCT future)
 - !! Design
 - follow implementation by Andrea Hahmann in ccm/bats/lsm
 - surface dataset and coarse grid are independent datasets.
 - supports regular lat/lon grid with dx constant and dy(j) constant or variable. unstructured or nested grids is deferred.
 - coarse to fine mapping is determined at startup using nearest neighbor mapping. only real land gridcells are used on the fine mesh if there is at least one real land gridcell for a coarse gridcell. if there are no real land gridcells, all fake fine gridcells are used for that coarse gridcell.
 - Phase I downscaling is just a copy of coarse grid data to fine grid data. upscaling is just an area average of active fine grid cells. downscaling and upscaling are done taking into account areas of cells for conservation. renormalize fine mesh areas as required with nearest neighbor mapping.
 - coarse/fine mapping is done only for "coupled" fields initially.
 - decomp of fine grid determined using coarse gridcells
 - surface dataset on fine grid
 - include grid information and surface information
 - every cell is a land cell, fill with swamp as needed. future surface datasets can have more sophisticated fill methods including setting fill type to glaciers for lats < 79S.
 - provides real/fake land cell mask
 - coarse grid
 - includes grid info including landfrac on coarse grid (set by mapped ccsm ocean grid or other method)
 - zenith angle in clm set by lat/lon of coarse grid for consistency in coupling.
 - Phase IIa downscaling will address topography
 - Nested grids in finemesh.
 - support up to triply nested grids. they must be perfect integer divisions of each other (edges must match). this will be done by reading up to 3 surface datasets.
 - modify surface dataset generation to generate nested grids
 - check edge matching
 - somehow specify via namelist the nested grid regions. use lat/lon?
 - infrastructure must support the nested grids. 1d domain type?
 - all history output to be written on finest mesh initial
- h2. Tests
- Phase I Tests:
 - T42 coarse grid, T42 fine grid, bfb or roundoff with baseline {color:blue}>-- BFB PASS --- T42 coarse grid, T42-double deg fine grid, bfb or roundoff with baseline for a special surface dataset where T42-double surface dataset is equivalent to T42 surface dataset -- **COMPLETE** –
 - T42 coarse grid, 0.5 deg fine grid, science looks good for realistic fine surface dataset – **COMPLETE** –
- Phase II Tests:
 - FV testing
 - scientific test runs (output on mss/TCRAIG/csm/\$CASE)
 - fm.T42base.amip0 - standard T42 amip run – **COMPLETE** --- fm.T42base.clim0 - standard T42 clim run – **COMPLETE** –
 - fm.T42half.amip0 - downscaling just a copy of cam forcing at all finemesh grid points – **COMPLETE** --- fm.T42half.clim0 - downscaling just a copy of cam forcing at all finemesh grid points – **COMPLETE** –
 - fm.T42half.clim0d1 - downscaling uses constant lapse rate, 0.0065 and we adjust t, pbot, th, q, vp, psrf, and rho. – **COMPLETE** --- fm.T42half.clim0d2 - same as "d1" but we use cam lapse rate from bottom four layers, (t4-t1)/(z4-z1) instead of constant. -- **COMPLETE** –

- ~~fm.T42half.clim0d3 - "d2" plus rain/snow redistribution based on local finemesh temperature (computed from topo and lapse) with transition from -5C to 0C. diagnose implied heat flux. - COMPLETE~~
- ~~fm.T42half.clim0d4 - "d3" plus add the diagnosed phase change heat flux back into the latent heat flux sent to the atmosphere to formally conserve heat across system. - COMPLETE~~
- fm.T42half.clim0d5 - "d4" plus add precip disaggregation scheme with mu=0.2. - COMPLETE
- ~~fm.T42half.clim0d6 - "d1" plus add precip disaggregation scheme with mu=0.2. or "d5" with lapse rate and rain/snow redistribution turned off. this will lead to identically ZERO heat flux added to the land to atmosphere latent heat term. - COMPLETE~~

h2. Phase II

Phase IIa Tasks:

- Implement sub-grid terrain (topography) into the finemesh model. Use a "standard" lapse rate to adjust temperature, pressure, and humidity from the coarse to fine grids. - COMPLETE
- Set coarse and finemesh topography - COMPLETE
- Upgrade downscaling using cam lapse rates. - COMPLETE in test system, not on trunk

Phase IIb Tasks:

- Make testbed testbed parameters
 - color: Color value is invalid
- Write coarse grid history output - COMPLETE
- Testbed testbed testbed testbed in clm
 - color: Color value is invalid
- Modify surface dataset generation to generate nested grids - COMPLETE
- Add numlon to coarse grid capability
- Make domain datatype private
- Modify edges(4) to be edgen, edgee, edges, edgew
- areaini_pt capability
- decomp/gridmap in InitDecomp
 - color: Color value is invalid
- Modify gridmap_type so it's 1d and more memory efficient - COMPLETE, fme12, bfb
- review decomp for memory scaling - COMPLETE, fme12, bfb
 - reduce global arrays and global xpe arrays - COMPLETE
 - clm3 datatype now local only - COMPLETE
 - domains now local except at initialization - COMPLETE
 - decomps being removed slowly in favor of gsmaps {color:blue} - MOSTLY COMPLETE
 - vegxy, wtxy - COMPLETE
- refactor DTM for accuracy and accuracy - COMPLETE
- implement MCT for parallel mapping and scatter/gather.

Unknown macro: {color=blue}

- COMPLETE --

- implement parallel I/O - UNDERWAY

- update atm driver to use parallel datm7 - COMPLETE

Phase IIc Tasks:

- Implement rain/snow redistribution based on temperature, lapse rate, and topography - COMPLETE
- Implement heat flux adjustment due to rain/snow redistribution - COMPLETE
- Implement repeatable random number generator - COMPLETE
- Implement precip downscaling parameterizations - COMPLETE
- Phase II Concerns:
 - Must move test system mods to clm trunk version - COMPLETE
 - Additional field/s (ie. lapse rate) coupling must be implemented in ccsm and cam in production systems - STILL TO BE DONE

Phase II downscaling/upscaling Implementation Details:

- topographic adjustment. update finemesh hgt, t, pbot, th, q, vp, psrf, qbot, egcm, and rhos based on topography and lapse rates. two lapse rates tested, constant 0.0065 and lapse rate sent from cam based on bottom 4 cam layers.
- rain/snow redistribution. redistribute rain/snow on finemesh gridcells based on local finemesh temp with transition from -5C (100% snow) to 0C (100% rain). Compute phase change heat flux and send this back to cam as a term in the latent heat flux.
- precipitation disaggregation. redistribute precip on finemesh gridcells based on "random" chance. both the change of precip and the amount of precip are based on random numbers. the amount follows a log distribution, $P \sim \log(1-rn)$. at least one finemesh cell must have precip for any coarse cell at each timestep. mu, the chance of precip on any cell is set to 0.20. normalization is implemented to conserve mass between coarse and overlying finemesh gridcells.

h2. Phase I:

Phase I Tasks:

- Code review, design discussion and review (10 hrs) - COMPLETE(30)
- Establish baseline run capability of clm, cam, ccsm (10 hrs) - COMPLETE(10)
- Setup CLM3 branch - COMPLETE(0)
- Keep WIKI updated, communicate (16 hrs) - ONGOING(8)
- Remain in contact with Rob on Ipcoupling rewrite - COMPLETE(6)
- Modify surface dataset generation and README, add corner points to surface dataset grid info (16 hrs) - COMPLETE(15)
- Determine coarse grid dataset format and implement tool. Hopefully just use gen_domain or mkgrid tools with few changes (16 hrs) - COMPLETE(12)
- Modify gridcell datatype; separate a2l, l2a datatype. Add extgrid datatypes. (12 hrs) - COMPLETE(12)
- Add coarse grid reader in CLM (6 hrs) - COMPLETE(6)
- Modify Typelnit (4 hrs) - COMPLETE(10)
- Modify Decomp (2 hrs)
- Implement intercoarse grid mapping taking into account Phase II needs to add downscaling/upscaling (8 hrs) - COMPLETE(8)
- Implement intercoarse grid mapping (4 hrs)
- Review verify interface (8 hrs) - COMPLETE(2)
- Implement testbed testbed testbed testbed to handle coarse/fine code (10 hrs)

- Add area and normalized area to clm history file (2 hrs) ~~COMPLETE(1)~~
 - Generate coarse grid datasets and fine grid surface datasets for resolutions TBD
 - color: Color value is invalid
 - Test ctm, cam, ccsm on bluesky, tempest, bangkok (60 hrs) ~~COMPLETE(75)~~
 - Prepare for checkin/handoff (4 hrs) – ~~COMPLETE(4)~~ –
 - Phase I Tasks Added:
 - Add coarse grid dataset fields
 - color: Color value is invalid
 - Review and modify handling of grid data in clm3 ~~COMPLETE(15)~~
 - Review and clean up ccsmMod.F90
 - color: Color value is invalid
 - Review and clean up xtmMod.F90, use of AreaMod, use of domainMod ~~COMPLETE(12)~~
 - Other code cleanup and infrastructure implementation ~~COMPLETE(12)~~ - Repository migration, merging, and associated testing ~~COMPLETE(8)~~
 - Add history field variables to coarse grid
 - color: Color value is invalid
 - fix domain/area needs to be adjusted by coarse/finemesh area overlap ~~COMPLETE~~
 - verify single gridpoint capability in surface dataset generation ~~COMPLETE~~ - generate new independent coarse grid landfrac file. generated by mkggrid in addition to the gridfile. remove landfrac from general gridfile
- ~~COMPLETE~~
- ~~implement new optional namelist for fatmndfrc in clm3. if it's input, it's used. if not, the get frac off atm grid file or surface dataset. COMPLETE~~
- Unknown macro: {color=blue}
- NOT NEEDED
- modify clm/cam/ccsm scripts for new clm namelist and datasets ~~COMPLETE~~
- Phase I Tasks Deferred:
- Unstructured/nested grid capability in fine mesh and clm3 in general
 - Coupling with WRF
 - Add numlon to coarse grid capability
 - Make domain private
 - Modify edges(4) to be edgen, edgee, edges, edgew
 - areaini_pt capability
 - clean up circular logic related to decomp/gridmap in InitDecomp
 - modify gridmap_type so it's 1d and more memory efficient
 - dynpft implementation in mksrfddata and clm3
 - verify independent finemesh works with FV, code works with coarse and finemesh same for FV
- Phase I Concerns:
- Conservation of mass, heat, water between coarse and fine grid using nearest neighbor? Must implement area corrected nearest neighbor mapping. – ~~COMPLETE~~ –
- Phase I Implementation Details:
- Code is bfb with baseline code and fully backward compatible
 - Code seems to produce roundoff errors in clm and cam when running with a finegrid double resolution surface dataset where the double resolution surface dataset is identically the same as the baseline surface dataset would be. coarse grid is original grid.
 - One new clm namelist added, "fatmgrid". This is the coarse gridfile.
 - When running same coarse and finemesh, users can just specify the surface dataset. The coarse grid will default to the grid that's in the surface dataset. This provides backward compatibility.
 - When running different coarse and finemesh, use the fatmgrid namelist to set the coarse grid and the fsurdat to set the finemesh grid/surface dataset.
 - Both coarse and fine grids are read by the same tool and the dataset format is based on the original surface dataset. They are both netcdf and none of the input field names have changed.
 - mkggriddata now produces a gridfile (sufficient for use as the coarse gridfile) from several different inputs including cam topo files, ccsm domain files, clm surface datasets, or the navy topo file.
 - mksurfddata now produces a surface dataset that has the additional PFTDATA_MASK field and filled wetland. The input is a mkggriddata gridfile. The code is bfb backward compatible. The file writes out all the fields from the input gridfile plus the surface dataset. This allows all surface datasets to be used either as coarse grid files or fine surface datasets.
 - The surface dataset is a superset of the grid dataset. Everything on the grid dataset is also on the surface dataset in addition to surface data. A surface dataset can be used as input for the coarse or finemesh grid.
- !! User Guide (updated as of clm3_5_03)
- CLM now has two grids internally, the coarse grid and the finemesh grid. CLM is always run on the finemesh grid. The coarse and finemesh grids can be identical. The grids are specified via namelist input. There are five "grid" namelist inputs now. Only fsurdat is required. The five namelist input are
 - fsurdat - defines the finemesh grid and surface dataset (required).
 - flndtopo - defines the finemesh topography. if not set, topography will not impact downscaling.
 - fatmgrid - defines the coarse grid, lat/lon. if not set, the coarse grid will be identical to the finemesh grid.
 - fatmndfrc - defines the coarse grid land fraction. if not set, the land fraction will be set in the fatmgrid (if set) or the fsurdat file.
 - fatmtopo - defines the coarse grid topography. if not set, topography will not impact downscaling.
 - To run non-finemesh cases, just set the fsurdat namelist and leave the others unused.
 - There are currently limited checks to compare the coarse and finemesh resolutions but it's recommended that either the coarse and finemesh grids be identical or that the finemesh grid be at least 4x the resolution of the coarse grid. In other cases, the simplistic coarse/finemesh mapping could produce relatively large errors of the forcing on the finemesh grid.
 - If the finemesh and coarse grids match each other in size and lat/lon within 1.0e-3 degrees at all gridpoints, then the coarse and finemesh grids are assumed to be identical and the grids will be forced to be exactly identical. This implicitly turns off downscaling for grids that might be identical except for roundoff type differences in their grid description.
 - All scripts run out-of-the-box and are bfb backward compatible for cases where the coarse grid and finemesh grid are identical.
 - For cases where the coarse and finemesh grids are different, the following is done
 - Downscaling is implemented using the nearest neighbor overlap scheme of Hahmann.
 - Surface forcing is adjusted for topographic variability on the finemesh grid using a constant lapse rate for the variables t, pbot, th, q, vp, and psrf. Eventually this is expected to be upgraded to more realistic forcing lapse rates.

- Precipitation disaggregation is carried out in the finemesh grid using the scheme of Hahmann. In the current implementation, any gridcell has a 20% chance of having precipitation ($\mu=0.2$) and the precipitation amount at any gridcell is set based on a random log weighted function.
 - Finemesh precipitation will be refractionated based on the local surface temperature. Precipitation will be 100% snow if $\text{temp} < -5\text{C}$ and 100% rain if $\text{temp} > 0\text{C}$. Between -5C and 0C , both snow and rain will exist based on a linear interpolation.
 - precip is normalized at each coarse gridcell such that mass is conserved.
 - the heat flux associated with the precipitation refractionation is added to the latent heat flux term diagnosed in the clm model and exported.
- For more detailed information about the implementation, see the developers guide below.
!! Developers Guide (updated to clm3_5_03)
 The high level calling tree of the grid and decomp initialization follows. Detailed implementation details are below.

```

call initialize1
  call control_init
  !--- read lat/lon into global arrays ---
  call surfrd_get_latlon(alatlon)
  call surfrd_get_latlon(llatlon)
  ! compare lat/lon, if "close" then set identical
  !--- initialize coarse/fine grid decompositions
  call decompInit_atm
  call map_setgadm
    ! compute overlap indices
  call decompInit_lnd
  !--- read grid info into local arrays ---
  call surfrd_get_grid(adomain)
  call surfrd_get_frac(adomain)
  call surfrd_get_topo(adomain)
  call surfrd_get_grid(ldomain)
  call surfrd_get_topo(ldomain)
  !--- read surface dataset ---
  allocate(vegxy,wtxy)
  call surfrd
  !--- initialize finemesh subgrids, gridcells, landunits, columns, pfts ---
  call decompInit_glcp
call initialize2
  call map_setmapsFM
    ! set coarse/finemesh mapping
    ! normalize coarse/finemesh areas
    ! normalize coarse/finemesh topography
  call initClmtype
    ! allocate clmtype data
  call init_atm2lnd_type
    ! initialize datatypes for downscaling and upscaling
  call init_adiaq_type
    ! initialize coarse/finemesh diagnostic arrays
  call initGridCells
    ! initialize clmtype data
  deallocate(vegxy,wtxy)

```

- The finemesh mask is computed in file areaMod.F90, subroutine map_setgadm. The algorithm implemented is
 - For each finemesh gridcell, compute the nearest coarse grid cell by testing whether that finemesh cell is in the bounds of the north, south, east, and west edges of the coarse gridcell.
 - Keep only coarse gridcells that have some land fraction.
 - For active coarse gridcells, the finemesh cells that are considered "fake" are thrown out as long as there is at least one "real" grid cell. The fake/real mask is read in from the surface dataset and stored in the domain field, pftm. This field is diagnosed when the surface dataset is generated. All finemesh gridcells are "real" except gridcells that are set to swamp when they contain no land on the high resolution raw surface dataset.
 - If there are no real land cells, then keep all the fake finemesh cells associated with an active coarse gridcell.
- Coupling fields are stored in the datatypes, atm2lnd_type and lnd2atm_type as defined in clm_atmlnd.F90. The instantiations of these datatypes is also in clm_atmlnd.F90 and they are called atm_a2l, atm_l2a, clm_a2l, and clm_l2a. The atm2lnd (a2l) types are for downscaling from the coarse to fine grid. The lnd2atm (l2a) types are used for upscaling from the fine to coarse grid.
- The actual downscaling code is in subroutine clm_mapa2l and the upscaling is in subroutine clm_mapl2a in file clm_atmlnd.F90.
- The grid overlaps are computed in file areaMod.F90, subroutine map_setgadm. The global finemesh array, gadm, contains the coarse grid cell associated with each finemesh gridcell. With the nearest neighbor algorithm, each finemesh cell is associated with only one coarse grid cell.
- A finemesh area normalization is computed in file areaMod.F90, subroutine map_setmapsFM. Basically, a normalized area is computed such that the sum of the normalized finemesh areas associated with a coarse gridcell are equal to the area of the coarse gridcell. This field is needed later for conservation adjustments and to compute normalized topography and land fraction. The field is called nara in the domain datatype.
- The finemesh topography is corrected such that the area weighted average matches the coarse grid topographic height using the following algorithm (search for "DOWNSCALING" in areaMod.F90, subroutine map_setmapsFM). This is field topo in the domain datatype.
 - The topography over ocean is set to a minimum of 0.0 where ocean is defined as any finemesh gridcell with landfrac = 0.
 - If the finemesh average topography is zero, then the coarse topographic height is added to each finemesh cell.
 - If the coarse topographic height is greater than zero, then the finemesh topography is normalized using a multiplicative factor.
 - If the coarse topographic height is less than or equal to zero, then the finemesh topography is normalized using an additive factor.

- The finemesh land fraction is computed from the coarse grid land fraction and adjusted by the normalized areas in order to support conservation of fields later. This is field frac in the domain datatype.
- The upscaling computation is just a simple normalized area weighted average of the finemesh cells to the coarse mesh. This is computed via a call to subroutine map_maparrayl in subroutine clm_mapl2a in file clm_atmInd.F90. The map1d_l2a is a datatype that holds the local mapping information and it's weights include all the normalized area corrections.
- The downscaling computation is done in subroutine clm_mapa2l in file clm_atmInd.F90 using the following algorithm
 - The coarse fields are mapped using a simple "copy" algorithm to the finemesh via a call to subroutine map_maparrayl in subroutine clm_mapa2l. The map1d_a2l datatype holds the local mapping information for the coarse to finemesh mapping. Basically, each finemesh cell has a single coarse mesh source and a weight of 1.0.
 - Topographic downscaling of the fields t, pbot, th, q, vp, and psrf is computed first and corrections are based on the lapse rate 0.0065. If topography files are not specified in input, no lapse rate corrections will be carried out.
 - Precipitation disaggregation is then computed for each finemesh cell. This requires a reproducible random number generator (for exact restart requirements), so the random numbers are generated by extracting end bits from real values of temperature and psrf as well as using grid information to produce spatial variability. This method has not been robustly tested for true randomness.
 - At each finemesh gridcell, there is a 20% (mu) chance that precipitation will fall at any timestep in that gridcell with the following caveats. First, because each finemesh gridcell has a 20% chance on average, the number of finemesh gridcells that contain precipitation at any timestep can be greater than or less than 20% but the long-term average should be 20%. Second, each coarse gridcell must have at least one finemesh gridcell that contains precipitation at all timesteps. If none of the finemesh gridcells associated with a coarse gridcell are precipitation gridcells, then the finemesh gridcell with the smallest random number is set to be a precipitation gridcell.
 - Each finemesh precipitation gridcell receives a percentage of the coarse gridcell precipitation based on a random log weighted function. The sum of the finemesh precipitation is normalized such that it equals the amount of precipitation on the associated coarse gridcell.
 - The finemesh precipitation is disaggregated such that finemesh gridcells with temperatures less than -5C are 100% snow and gridcells with temperatures above 0C are 100% rain. Both snow and rain exist between -5C and 0C and the distribution is linearly varying with temperature. The heat flux associated with the phase change is diagnosed and later added to the latent heat flux term in the land model. This variable is named adiaq_lflux and is added to the latent heat flux in subroutine clm_map2gcell in file clm_atmInd.F90.
 - A few parameters are set in subroutine clm_mapa2l. Namely mu, the chance that any finemesh gridcell has precipitation; lapse, the lapse rate; and precip_max, a threshold to limit the amount of precipitation in any gridcell associated with the random log function.
 - In addition, there are several comments in subroutine clm_mapa2l that allow the user to modify code to turn off topographic effects, change the precipitation disaggregation either in distribution or amounts, or to turn off the refractionation.
- NOTE: In the future, the lapse rate will be an externally computed quantity and it will be a forcing quantity in the coupling implementation. This requires coordination with external atmosphere and coupling models. Until that is implemented, the constant lapse rate value, 0.0065 is used.