VARIABLE RESOLUTION METHODS IN FV^3

Lucas Harris

and the GFDL FV^3 Team

FV3 Workshop in Taiwan

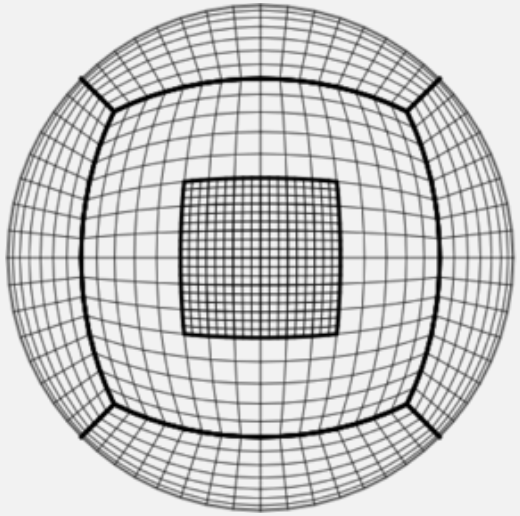
4 December 2017

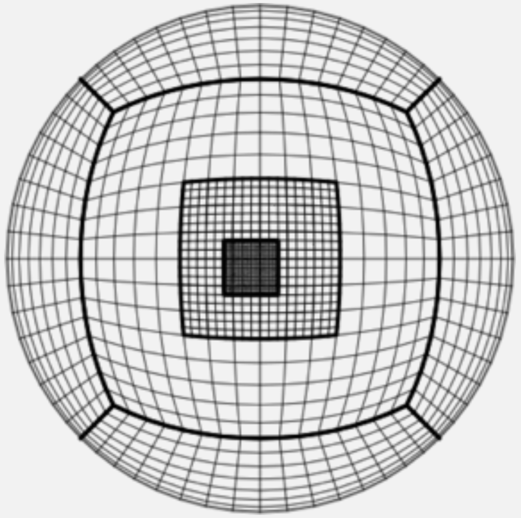
HIGH RESOLUTION MODELING: LIMITED AREA VS. GLOBAL MODELS

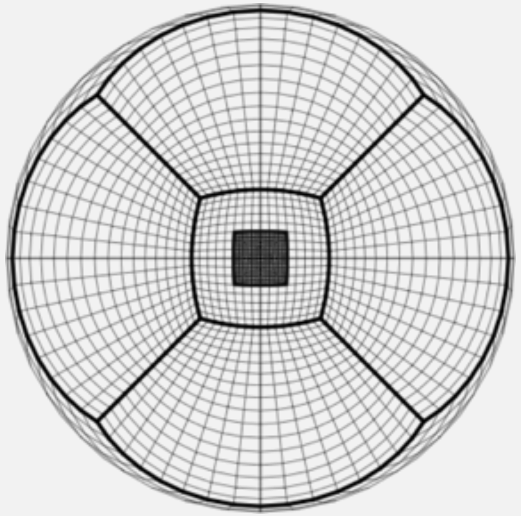
- Stand-alone regional models are commonly used for mesoscale simulation and regional climate modeling. But boundary errors creep in after a few days.
  - Require potentially inconsistent BCs from a global model.
  - No feedback onto large-scale
- Global models have no boundaries and are globally consistent,
  but global high resolution can be impractical
- **Solution: grid refinement of a global model!**

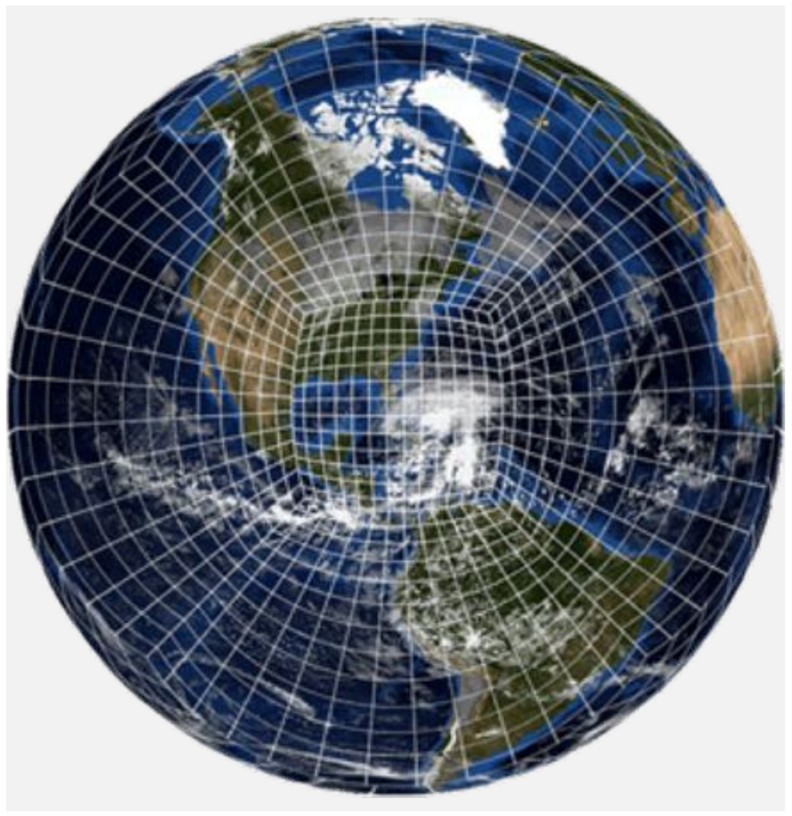
GRID REFINEMENT

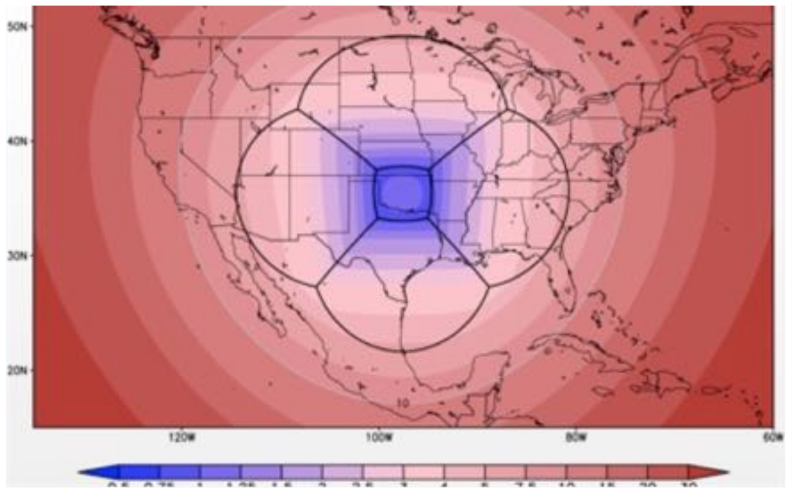
- FV^3 supports both grid stretching and two-way grid nesting
- Grids can be constructed on-the-fly within seconds.
- Both techniques have strengths and weaknesses:
  Combining the two leads to the best results











STRETCHED GRID

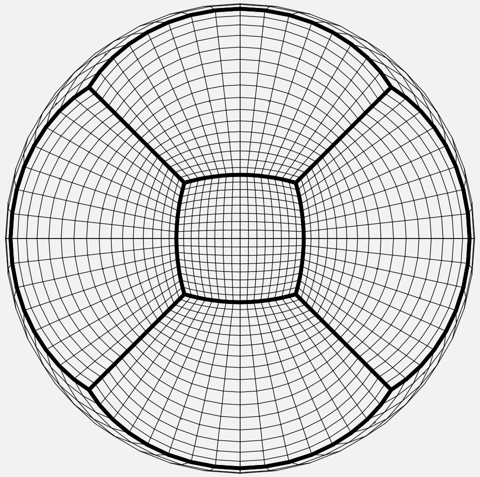
- **The simple, easy way to achieve grid refinement**
- Smooth deformation! Requires no changes to the solver
- No abrupt discontinuity.
- Capable of extreme refinement (80x!!) for easy storm-scale simulations on a full-size earth
- Requires a "compromise" tuning between coarse and fine regions. (A good scale-aware scheme can help here.)

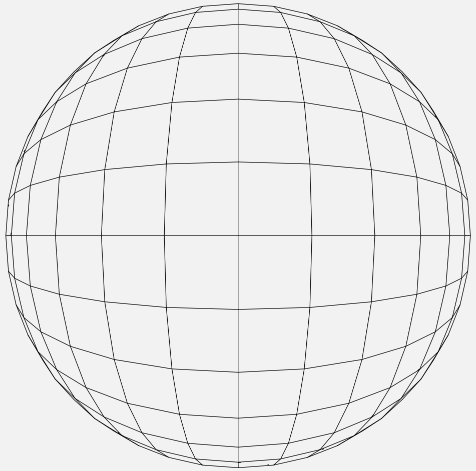
SCHMIDT TRANSFORMATION

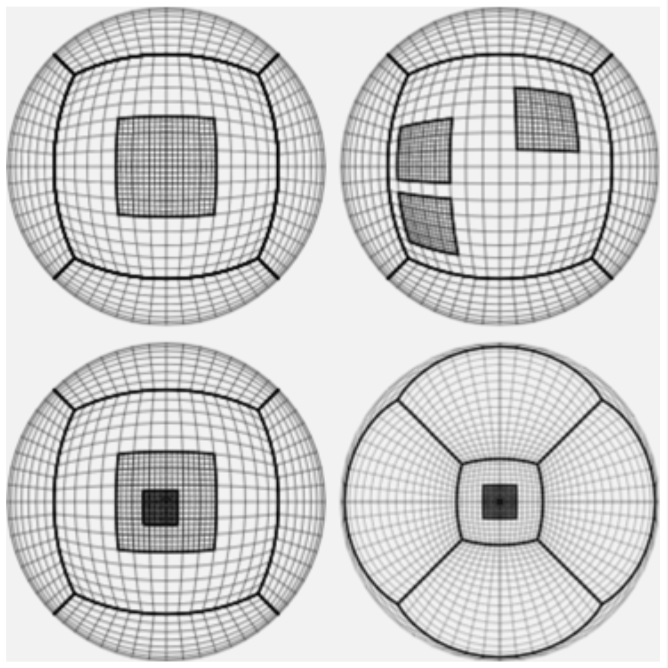
$$\sin\vartheta = \frac{D + \sin\theta}{1 + D\sin\theta}$$

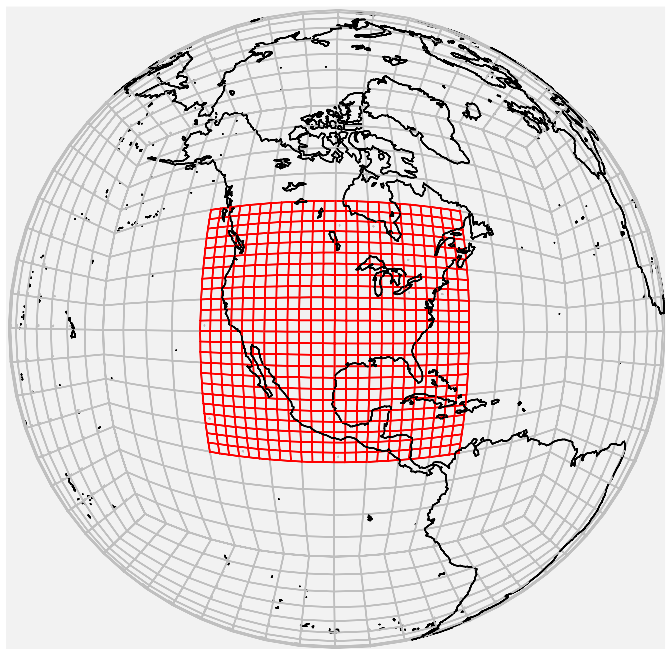
$$D = \frac{1 - c^2}{1 + c^2},$$

- Smoothly deforms the cubed sphere into a "truncated pyramid", with the high-resolution face at the top of the "pyramid"
- Transformation is **analytic**. Easily implemented and quickly executed
- The resulting pyramid can then be rotated to an arbitrary target point
  - Transformation does coarsen the opposing side of the sphere
  - Size of high-resolution region decreases with increasing stretching ratio









TWO-WAY GRID NESTING

- Simultaneous coupled, consistent global and regional solution. No waiting for a regional prediction!
- Different grids permit different parameterizations and timesteps; doesn't need a "compromise" for high-resolution region
- Flexible! Great possibilities for combining nesting and stretching.

GRID NESTING:
BOUNDARY CONDITIONS

- Strategy: fill halo (ghost) cells with boundary conditions
  Interior of solver needs no changes
- All variables linearly interpolated in space into nested grid halo.
  Correct upwind BCs "baked in" by FV's upstream-biased fluxes
  - BCs for all solution variables, as well as C-grid winds, and divergence
- Nonhydrostatic solver requires nonhydrostatic pressure, computed using the semi-implicit solver—consistent with interior algorithm

GRID NESTING: BOUNDARY CONDITIONS

- **Concurrent nesting**: BCs extrapolated in time so nest and coarse grids can run **simultaneously**.
  - BCs stepped forward every acoustic timestep
  - New BC data updated at the nest interaction frequency, usually vertical remap frequency
- Extrapolation is formally unstable but is not a problem in practice
  - Option to limit extrapolation to ensure positivity for scalars
- Two-time level extrapolation requires saving BCs across restarts to ensure run-to-run reproducibility

GRID NESTING
TOPOGRAPHY AND SMOOTHING

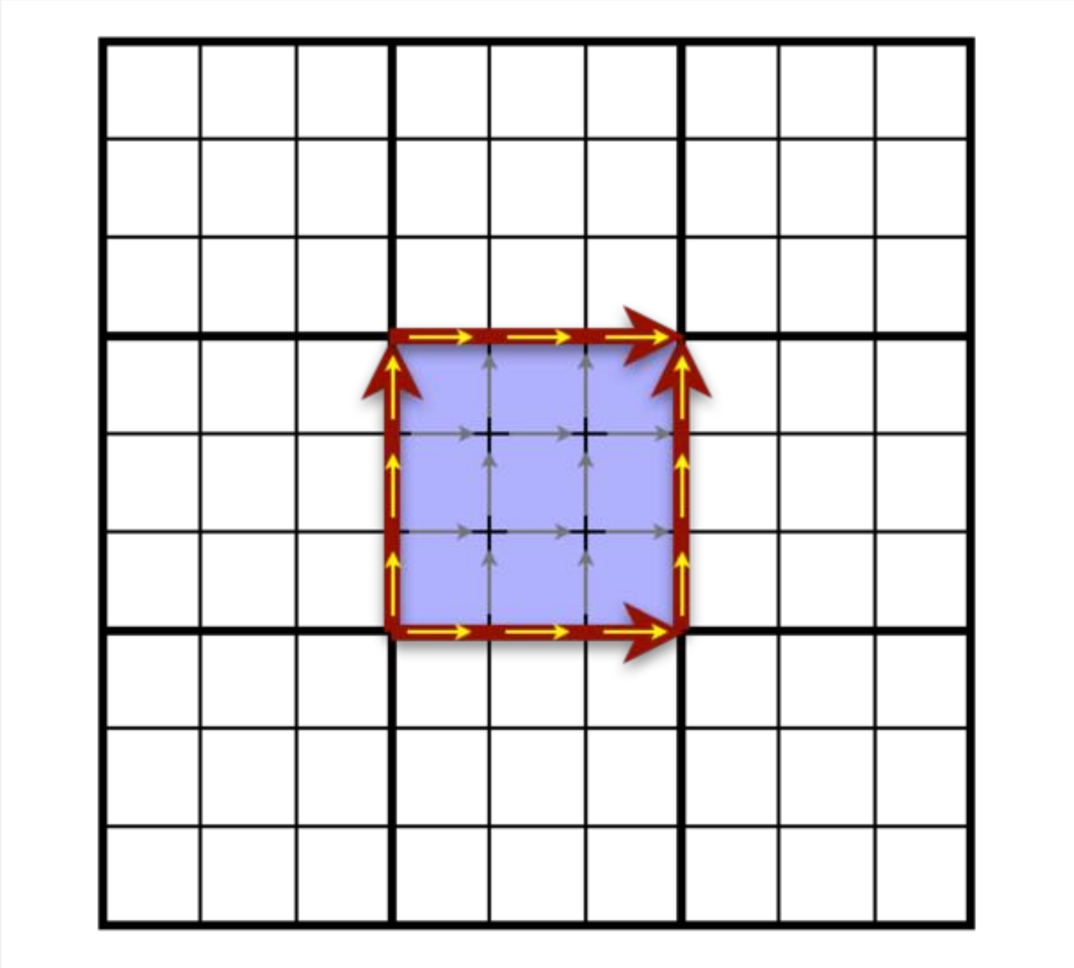
- FV^3 applies **no** additional diffusion or relaxation at the boundaries
  - Linear interpolation introduces some smoothing without creating new extrema
- For consistency, halo topography is linearly interpolated from the coarse grid, the same way as the solution variables
- Topography near the boundary is blended with the interpolated coarse-grid topography

TWO-WAY INTERACTION

- Two-way nesting: coarse grid solution periodically replaced (“updated”) by nested-grid solution where the grids coincide
- Essential for small-to-large-scale interaction (e.g. hurricanes, gravity-wave drag, small-scale orographic/coastal processes)
- Theory suggests two-way nesting yields a better nested-grid solution: Harris and Durran (2010), T.T. Warner et al (1997)

TWO-WAY INTERACTION

- Averaging update consistent with FV discretization
  - For consistency, the initialized coarse-grid topography is updated from the nested-grid using the same algorithm
- Cell average on scalars, including δz and w
- In-line average for winds, to conserve vorticity



MASS CONSERVATION AND TWO-WAY NESTING

- Conservation usually requires flux BCs at the nested-grid boundary
  These are difficult to implement with the time-extrapolation BC
- **Our approach:** Update everything except mass (δp) and tracers
- **Very simple!** Works regardless of BC and grid alignment
  - Two-way nesting over-specifies coarse-grid solution
    Less updating, less over-specification

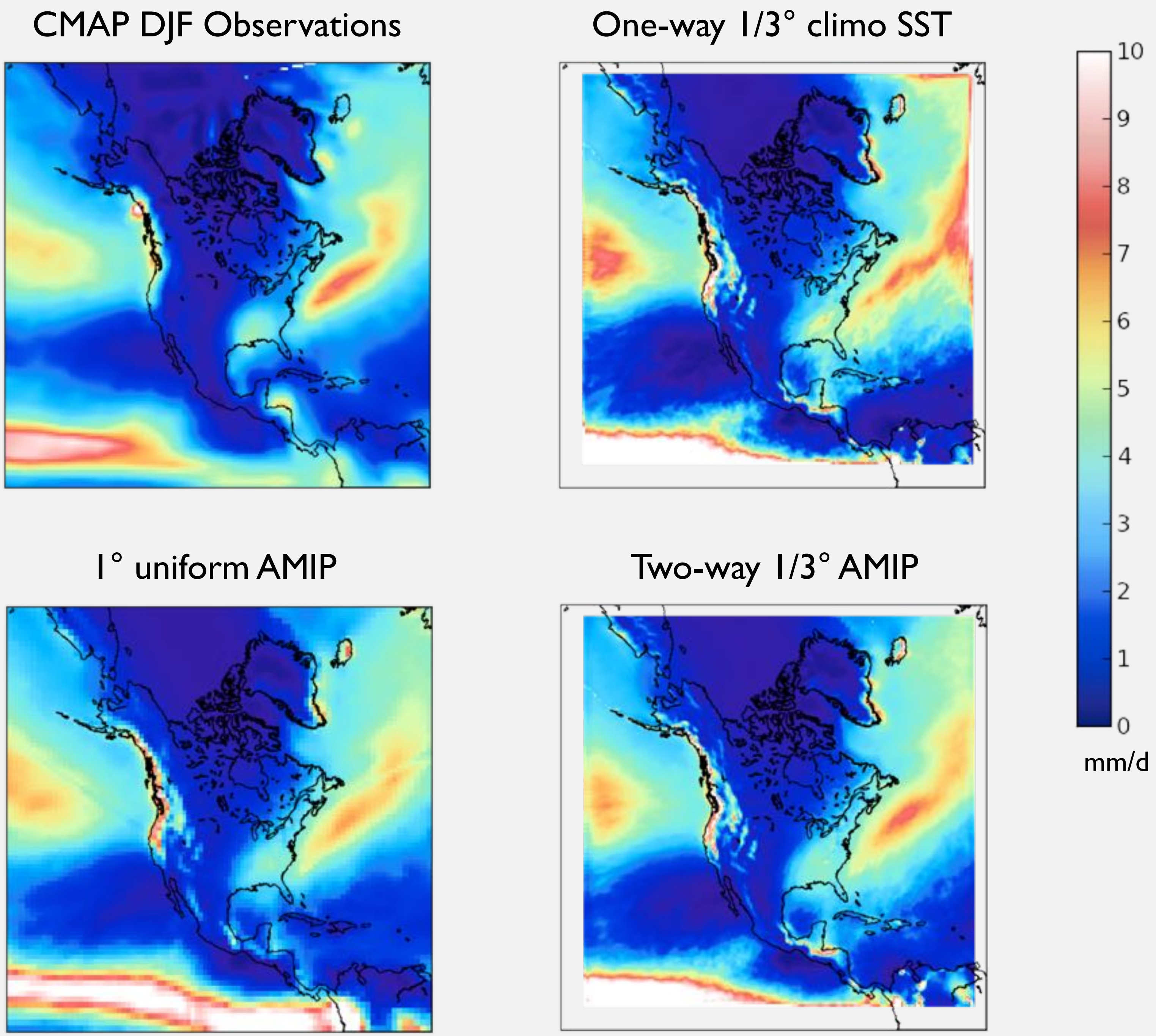
MASS CONSERVATION AND TWO-WAY NESTING

- **Our approach:** Update everything except mass (δp) and tracers
- Because δp is the vertical coordinate, we then need to remap the nested-grid data to the coarse grid's vertical coordinate
- Under development: "Renormalization-update" for tracers uses a layer-by-layer fixer to ensure tracer mass conservation

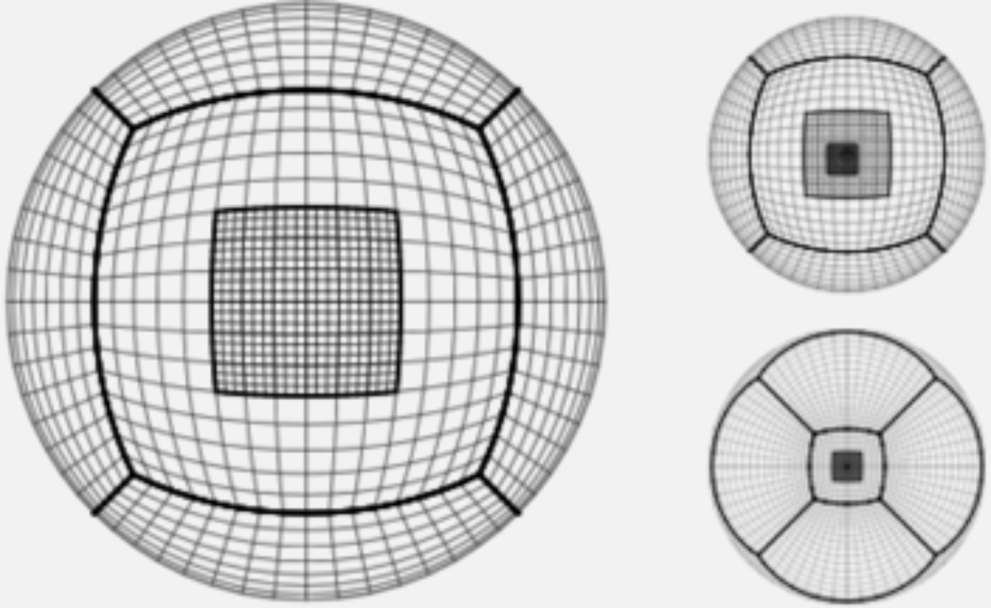
TWO-WAY VS. ONE-WAY GRID NESTING

- GFDL HiRAM climate model

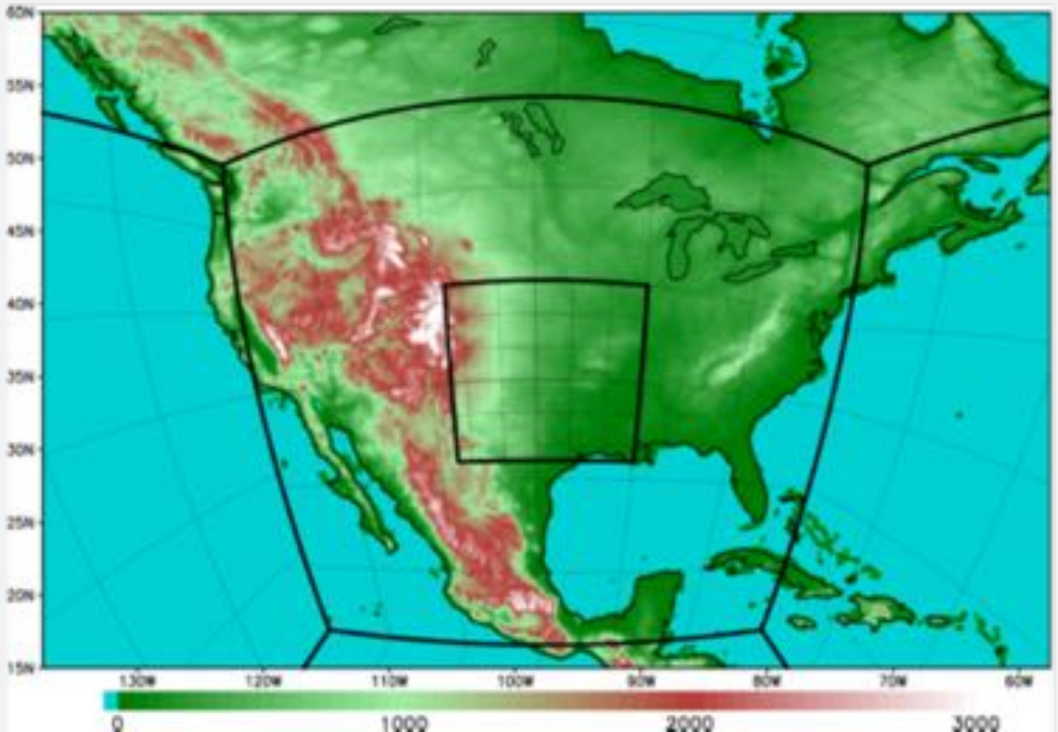
  c90 (1°) and
  c90n3 (1° and 1/3°)

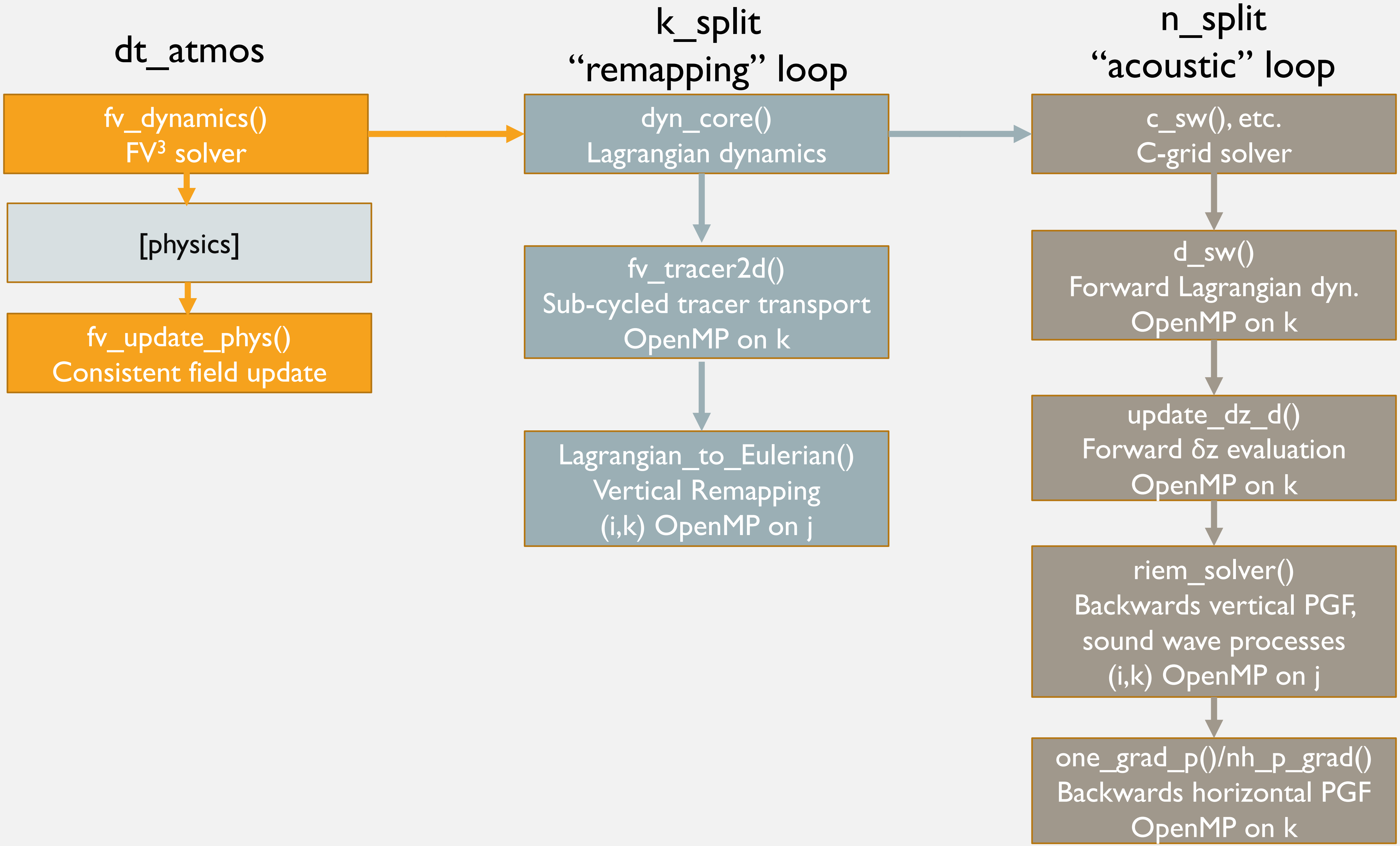


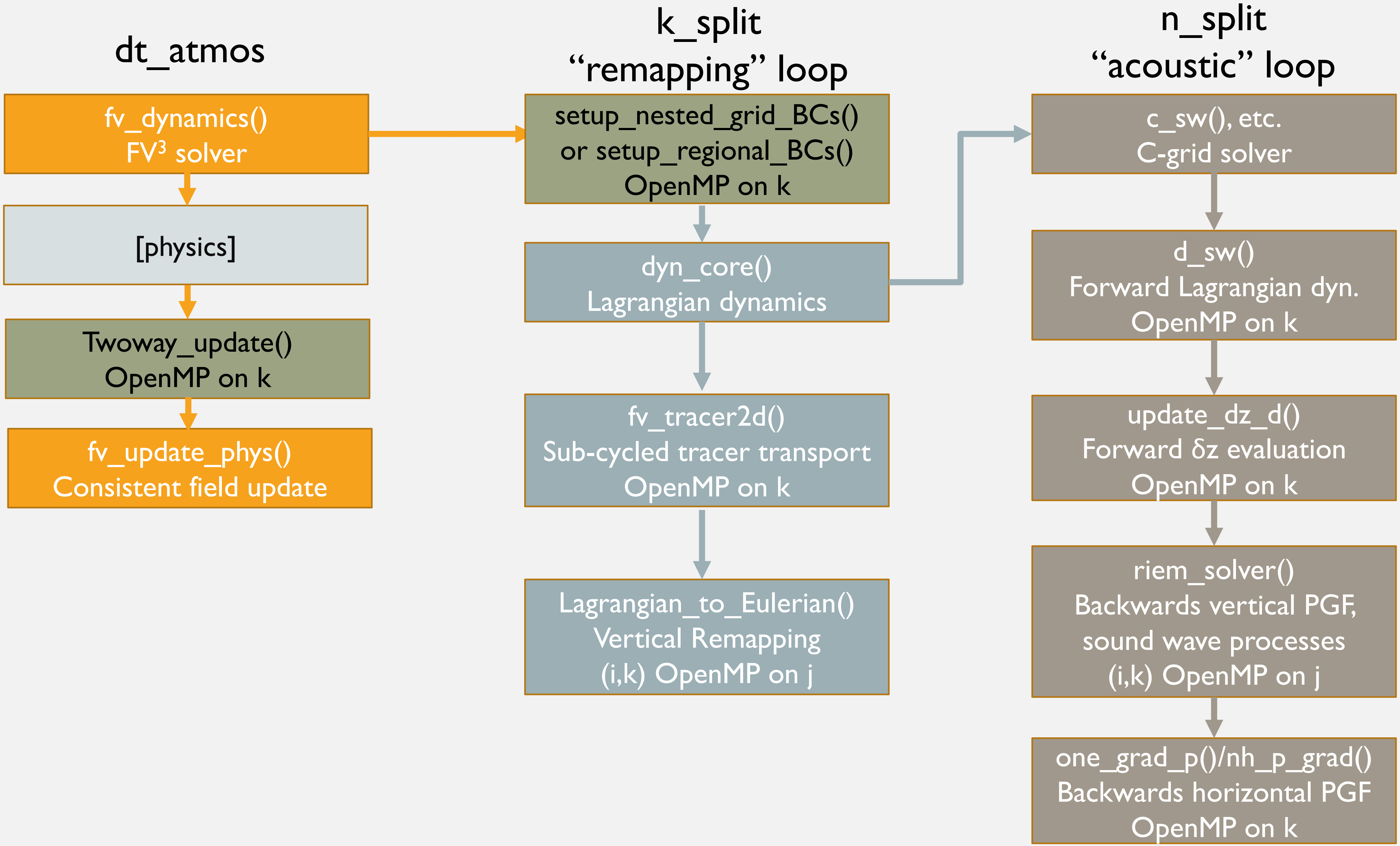
NESTING IMPLEMENTATION IN FV^3



- The nested grid is an additional tile beyond the six for the cubed sphere. There can exist many nests or nests within nests.
- Each nested grid has its own processor list and is integrated concurrently. This greatly aids load balancing between grids.




dt_atmos
fv_dynamics()
FV3 solver
[physics]
fv_update_phys()
Consistent field update
k_split
“remapping” loop
dyn_core()
Lagrangian dynamics
fv_tracer2d()
Sub-cycled tracer transport
OpenMP on k
Lagrangian_to_Eulerian()
Vertical Remapping
(i,k) OpenMP on j
n_split
“acoustic” loop
c_sw(), etc.
C-grid solver
d_sw()
Forward Lagrangian dyn.
OpenMP on k
update_dz_d()
Forward δz evaluation
OpenMP on k
riem_solver()
Backwards vertical PGF,
sound wave processes
(i,k) OpenMP on j
one_grad_p()/nh_p_grad()
Backwards horizontal PGF
OpenMP on k


dt_atmos
fv_dynamics()
FV3 solver
[physics]
Twoway_update()
OpenMP on k
fv_update_phys()
Consistent field update
k_split
"remapping" loop
setup_nested_grid_BCs()
or setup_regional_BCs()
OpenMP on k
dyn_core()
Lagrangian dynamics
fv_tracer2d()
Sub-cycled tracer transport
OpenMP on k
Lagrangian_to_Eulerian()
Vertical Remapping
(i,k) OpenMP on j
n_split
"acoustic" loop
c_sw(), etc.
C-grid solver
d_sw()
Forward Lagrangian dyn.
OpenMP on k
update_dz_d()
Forward δz evaluation
OpenMP on k
riem_solver()
Backwards vertical PGF,
sound wave processes
(i,k) OpenMP on j
one_grad_p()/nh_p_grad()
Backwards horizontal PGF
OpenMP on k

NESTED-GRID WORKFLOW

- Nested grids can be generated online, although orography, land-surface information, and initial conditions have to be generated.
- Modifications of standard NCEP tools allow offline generation of grid information and ICs
- Each grid gets its own namelist
  Any runtime parameter can be customized for the individual grids
- GFDL fregrid can perform conservative remapping of nested-grid output onto a regional regular latitude-longitude grid. It is also possible to use common software (Python, NCL, IDL, GrADS (?), etc) to plot the native nested grid.