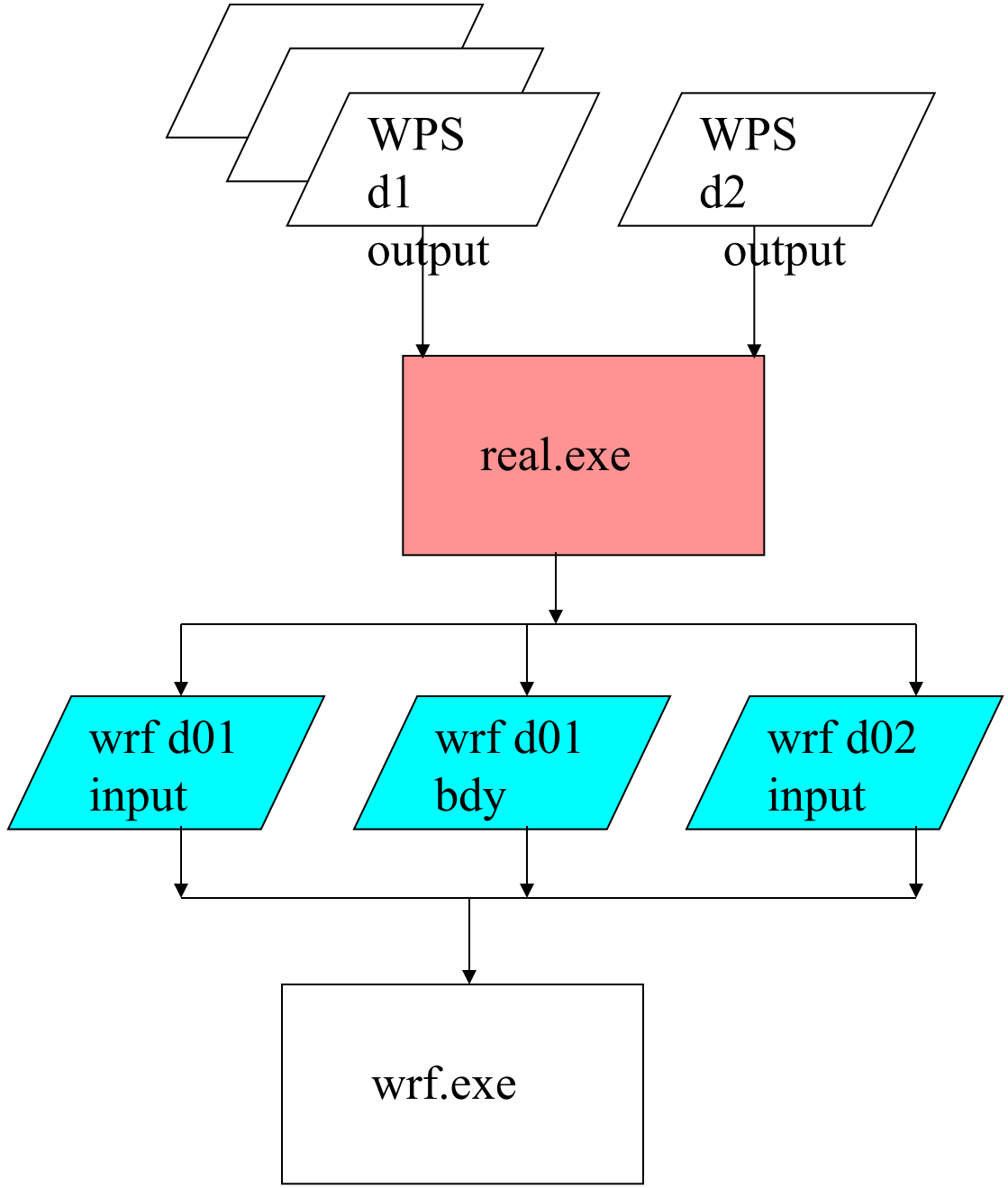
Real

Description of General Functions

Dave Gill
gill@ucar.edu



Real program in a nutshell

- Function
- Required input variables
- Base State for ARW
- Standard generated output
- Optional output
- Vertical interpolation
- Soil level interpolation
- Water temperature initialization
- Sea-ice initialization
- Land/Water mask
- Nested processing in real for ARW

Function

- The ARW WRF model pre-processor is ***real.exe***
- The real.exe program is available ***serial*** or ***DM parallel*** (primarily for aggregate memory purposes, as opposed to timing performance)
- This program is automatically generated when the model is built and the requested use is for a real data case
- The real.exe program takes data ***from WPS*** and transforms the data ***for WRF***
- Similar to the ARW idealized data pre-processor, real.exe is tightly coupled to the WRF model through the ***Registry***

Function

- ***3D forecast*** or simulation
- ***Meteorological input*** data that primarily originated from a previous forecast or analysis, probably via the WPS package
- Anticipated ***utilization of physics*** packages for microphysics, surface conditions, radiation, convection, and boundary layer (ARW: maybe usage of nudging capabilities)

Function

- A non-Cartesian ***projected domain***
  - ARW: Lambert conformal, Mercator, polar stereographic, rotated latitude/longitude (global or regional)
- Selection of ***realistic static fields*** of topography, land use, vegetation, and soil category data
- Requirement of ***time dependent*** lateral boundary conditions for a regional forecast

Function

- Not referring to the ***Variational*** or the ***Digital Filtering*** usage of Initialization

- Generation of ***diagnostics*** necessary for assumed WRF model input
- Input field ***adjustment*** for consistency of static and time dependent fields (land mask with soil temperature, etc.)
- ARW: computation of ***reference*** and ***perturbation*** fields
- Generation of ***initial*** state for each of the requested domains
- Creation of a ***lateral boundary file*** for the most coarse domain
- ***Vertical interpolation*** for 3d meteorological fields and for sub-surface soil data

Required input Variables

- The metgrid program typically provides meteorological data to the real program.
- ARW **mandatory**:
  - 3d and surface: horizontal winds, temperature, relative humidity, geopotential height
  - 3d soil: soil temperature
  - 2d fields: surface pressure, sea-level pressure, land mask
- ARW **optional** (but desirable):
  - 3d soil: soil moisture
  - 2d fields: topography elevation of input data, SST, sea-ice, skin temperature

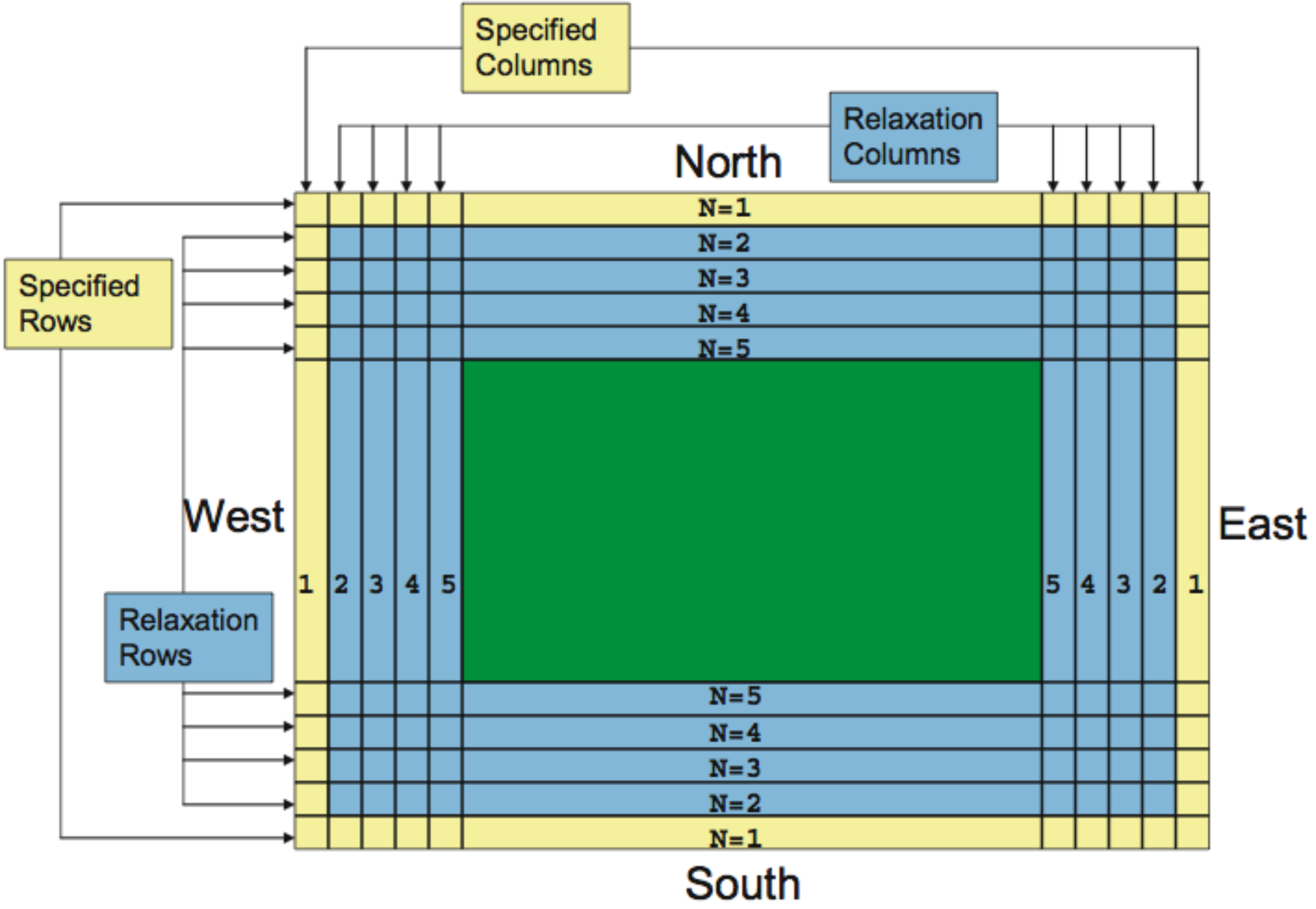
Base State for ARW

- Several of the mass-point fields are separated into a time-independent **base state** (also called a reference state) and a perturbation from the base state
- The base state fields are only **functions of the topography** and a few user-selectable constants
- If the **topography changes**, such as with a moving nest, the base state fields are modified
- Feedback for **2-way nesting** also impacts base state fields through **topographic averaging**
- No base state computations are required prior to the real program

Standard Generated Output

- For regional forecasts, the real program generates both an both an initial (***wrfinput_d01***) and a lateral boundary (***wrfbdy_d01***)
- The boundary file is not required for global forecasts with ARW
- The initial condition file contains a single time period of data
- If ***n*** times were processed with real, the lateral boundary file contains ***n-1*** time slices
- These files contain data **used directly** by the WRF model
- The ARW initial condition file may be ingested by the **WRFDA** code

Real-Data Lateral Boundary Condition: Location of Specified and Relaxation Zones



Optional Output

- The ARW allows users to provide a lower boundary condition file (***wrflow_d01***), containing slowly changing files such as SST, sea-ice, greenness fraction

Optional Output

- ARW users may request that each time period of data generate an “initial condition” file, typically for diagnostic purposes (***wrfinput_d01.2000-01-24_18:00:00***)

Optional Output

- ARW users who select to implement analysis nudging in the WRF model (also known as grid nudging) generate extra input fields for the model (***wrffdda_d01***)

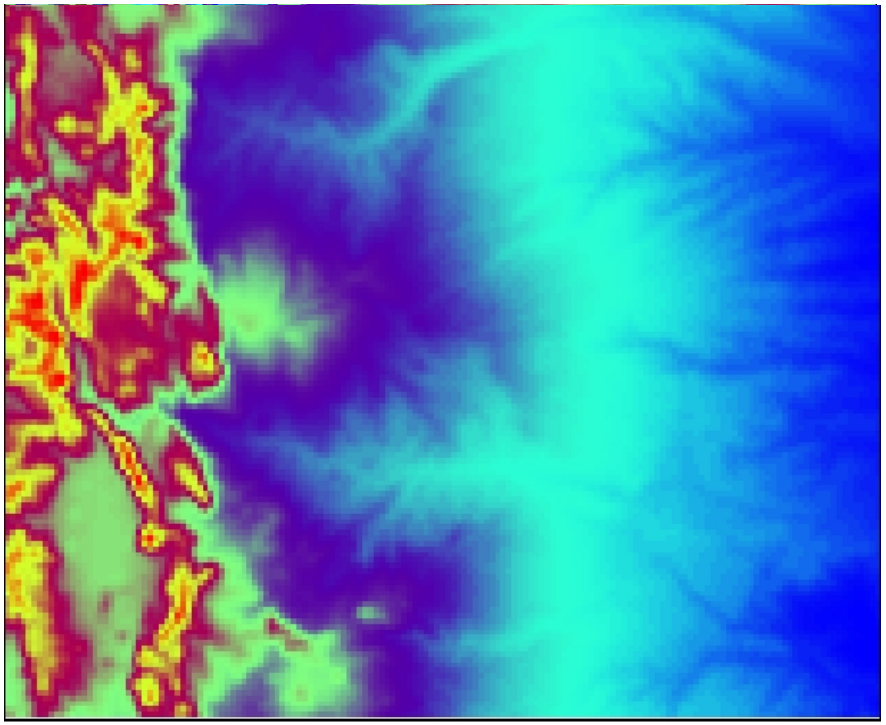
Vertical Interpolation

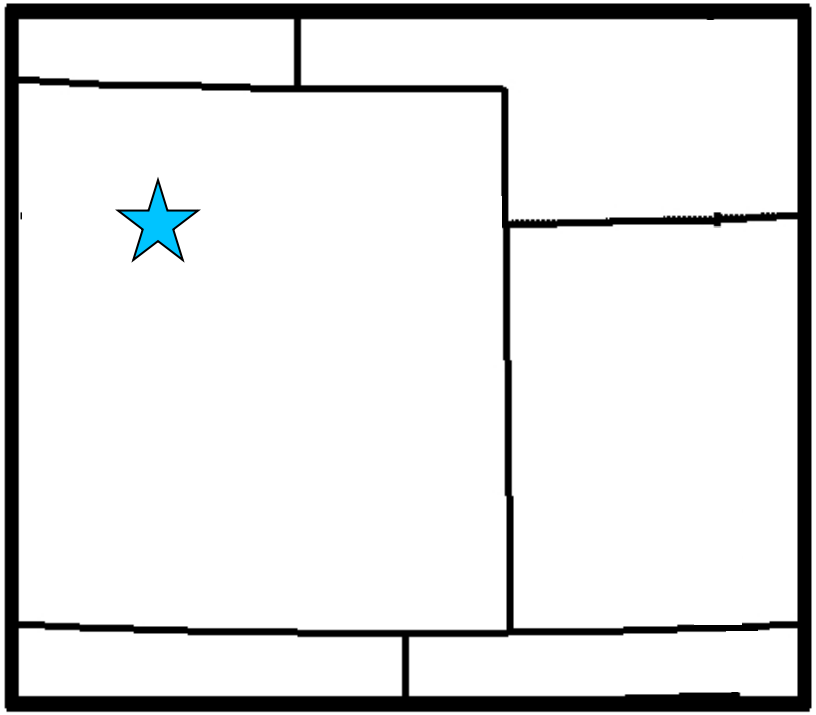
- A number of vertical interpolation options are available to users
- The options can have a significant impact on the initial conditions passed to the model
- More information is contained in the info file ***README.namelist*** in the ***run*** directory
- Options are located in the ***&domains*** namelist record of ***namelist.input***

Vertical Interpolation

- Impact: *Expected region of changes*
- *Non-standard setting*
- Which level is being viewed

- Topography and domain for difference plots, 160x140, 4 km, input = 40 km NAM

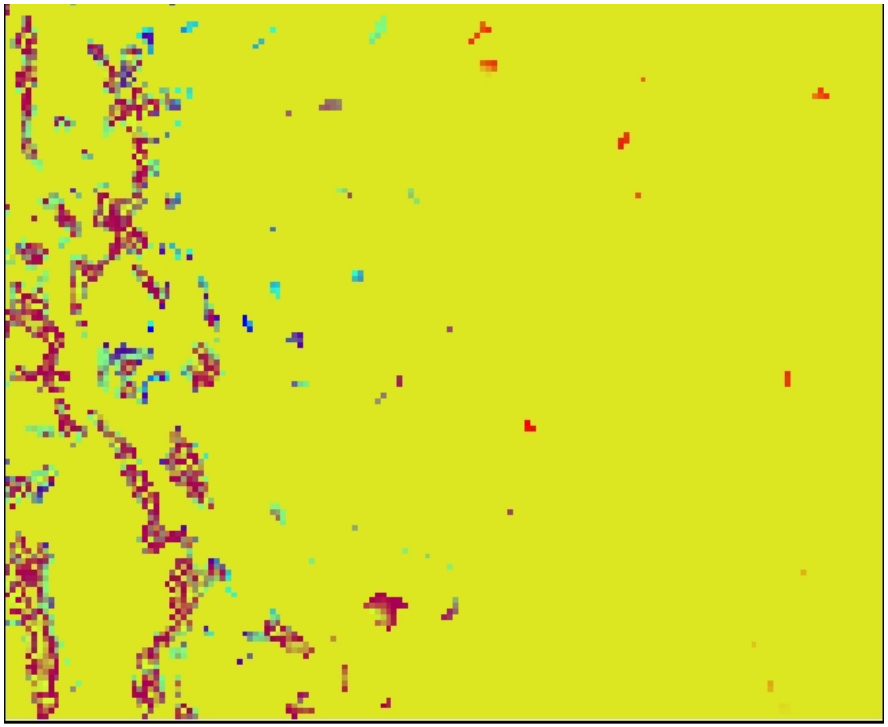




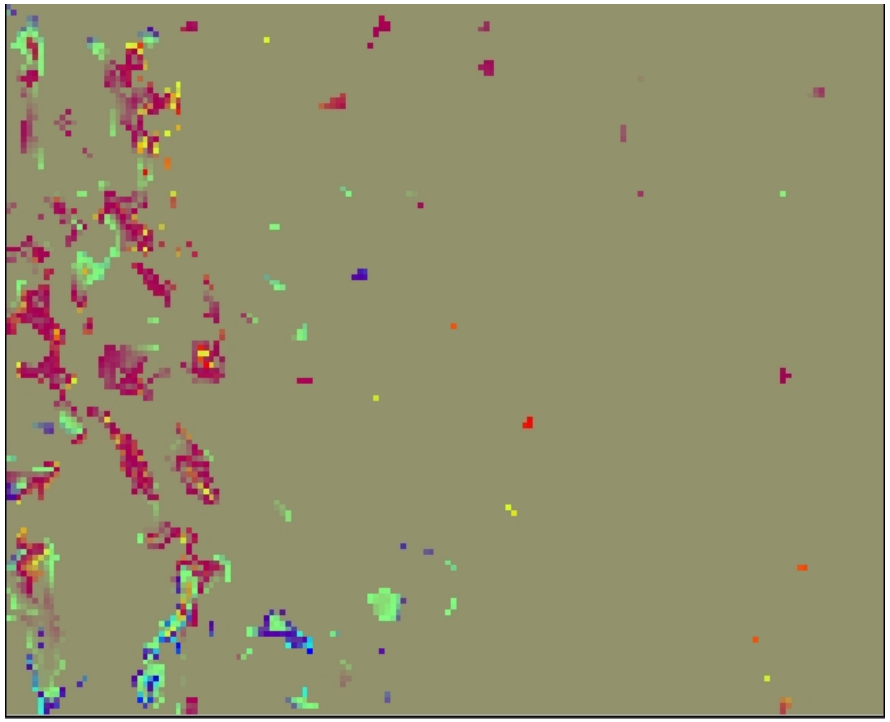
Vertical Interpolation

- Impact: few lowest levels only
- force_sfc_in_vinterp = 0
- η level 1

- Theta (-8 K blue, 0 K yellow)



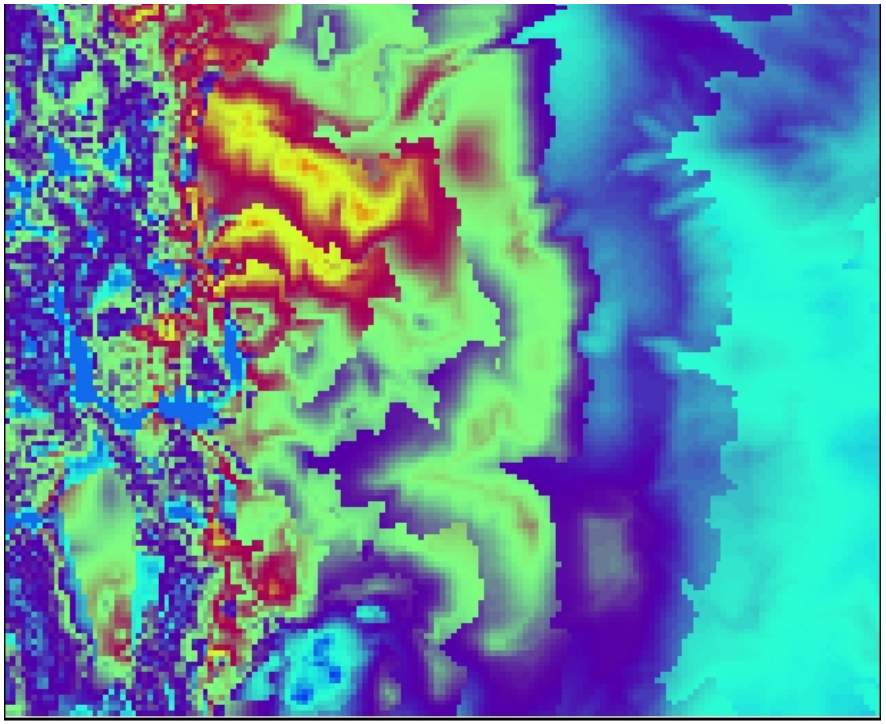
- U (-3 m/s blue, 2 m/s red)



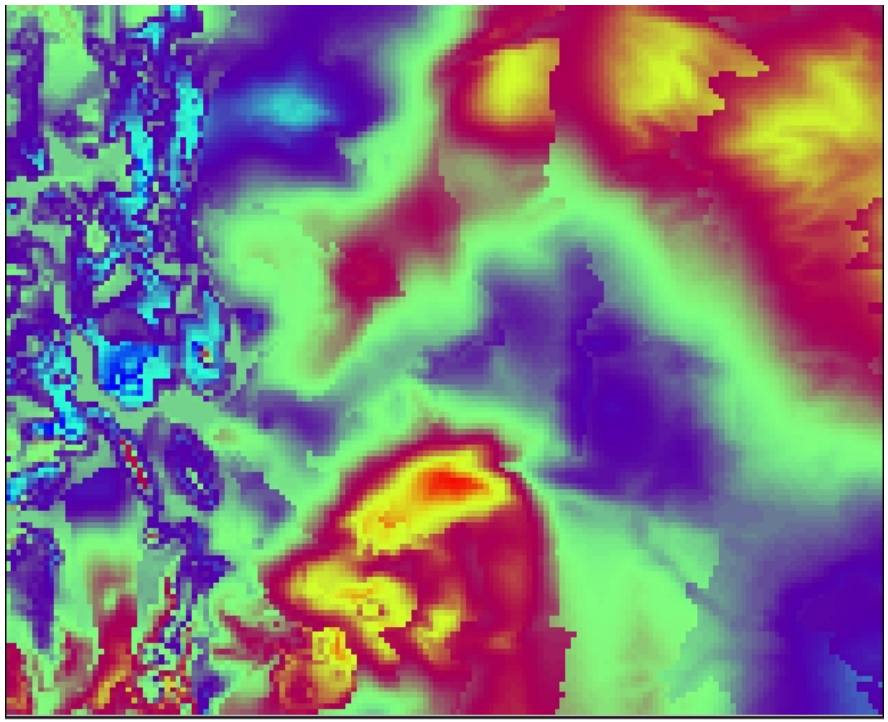
Vertical Interpolation

- Impact: few lowest levels only
- force_sfc_in_vinterp = 6
- η level 4

- Theta (0 K blue, 10 K red)



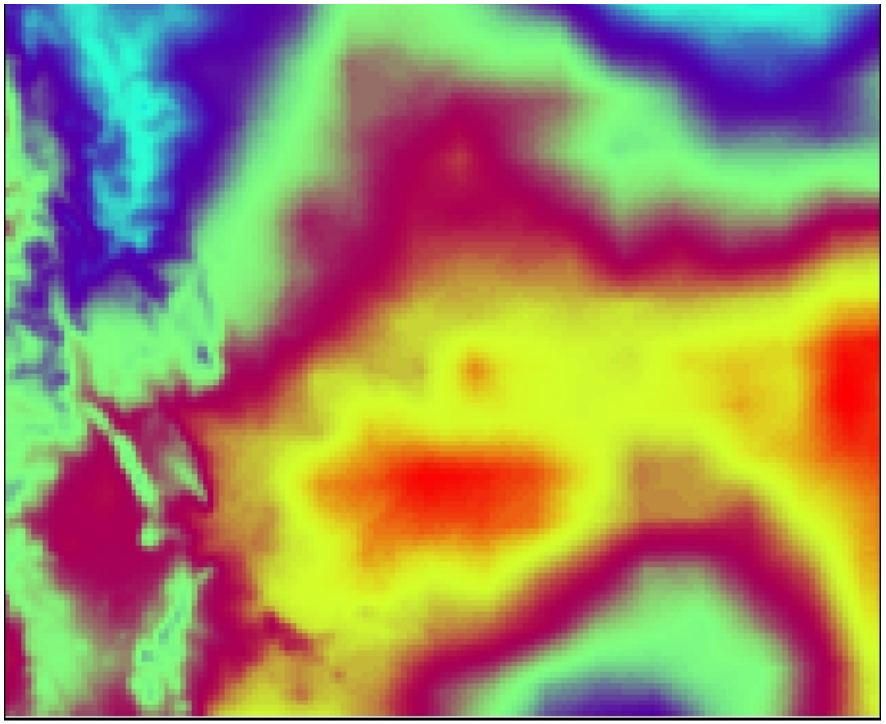
- U (-5 m/s blue, 6 m/s red)



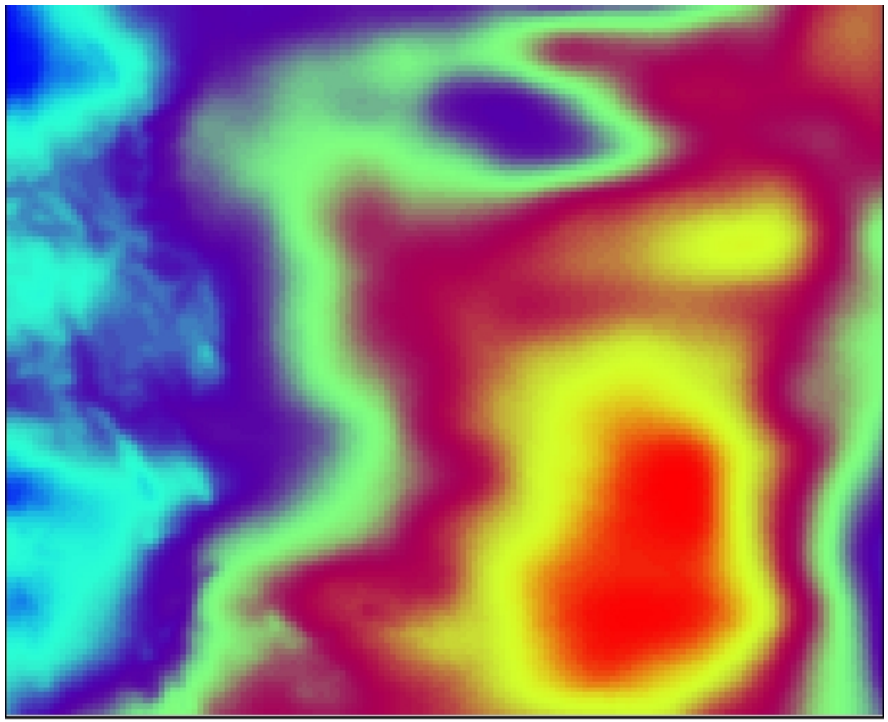
Vertical Interpolation

- Impact: above first 4 levels, most near tropopause
- lagrange_order = 2
- η level TOP

- Theta (0.7 K blue, 1.6 K red)



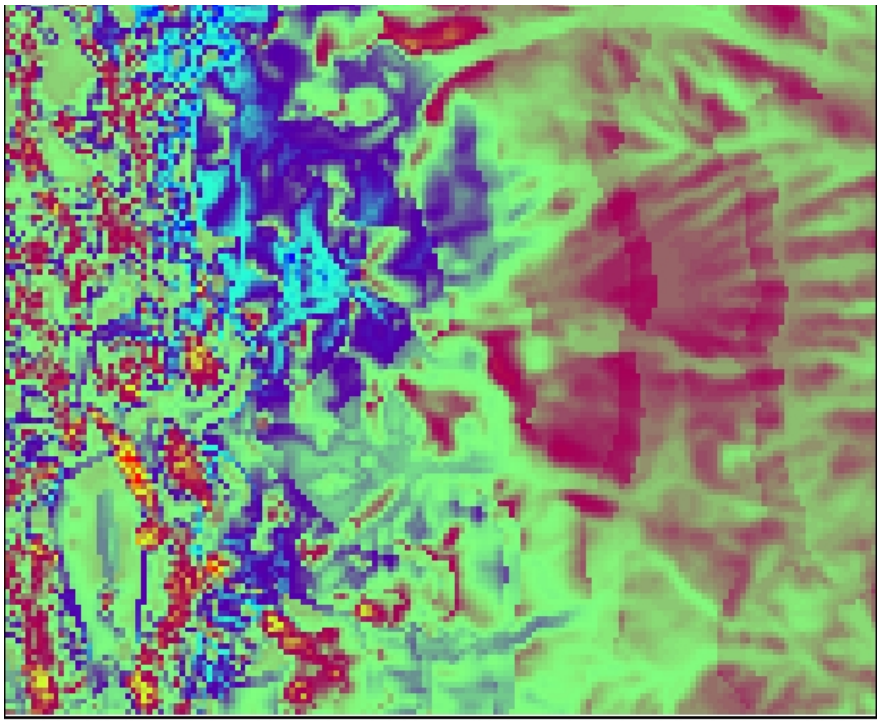
- U (0.4 m/s blue, 1.4 m/s red)



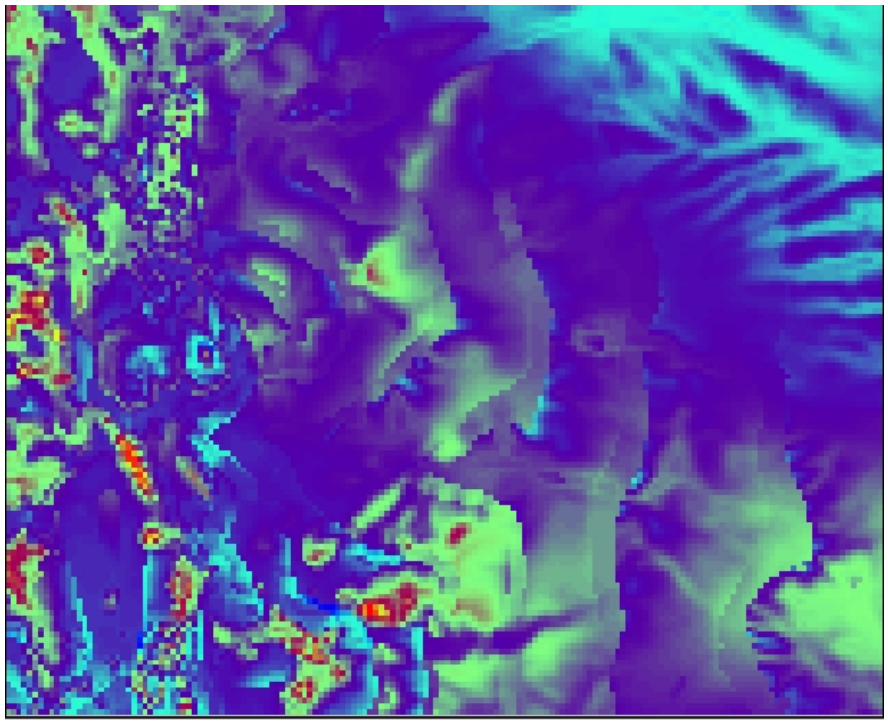
Vertical Interpolation

- Impact: lowest level only
- lowest_lev_from_sfc = T
- η level 1

- Theta (-10 K blue, 8 K red)



- U (-3 m/s blue, 7 m/s red)



Vertical Interpolation

- Impact: outer few rows and column, amplitude damps upward
- smooth_cg_topo = T
- η level 1

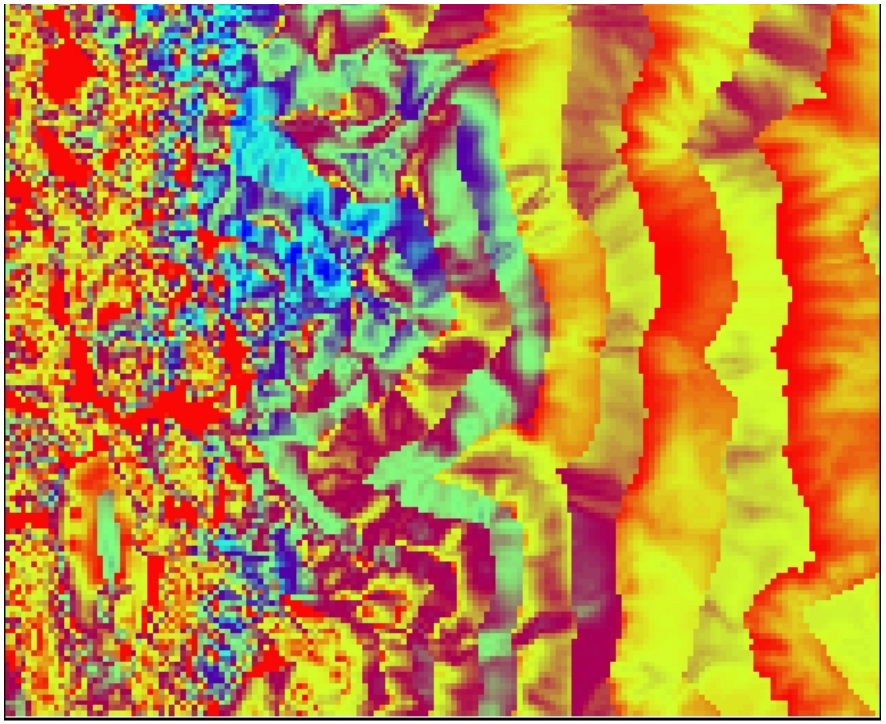
- Theta (-10 K blue, 9 K red)

- U (-6 m/s blue, 6 m/s red)

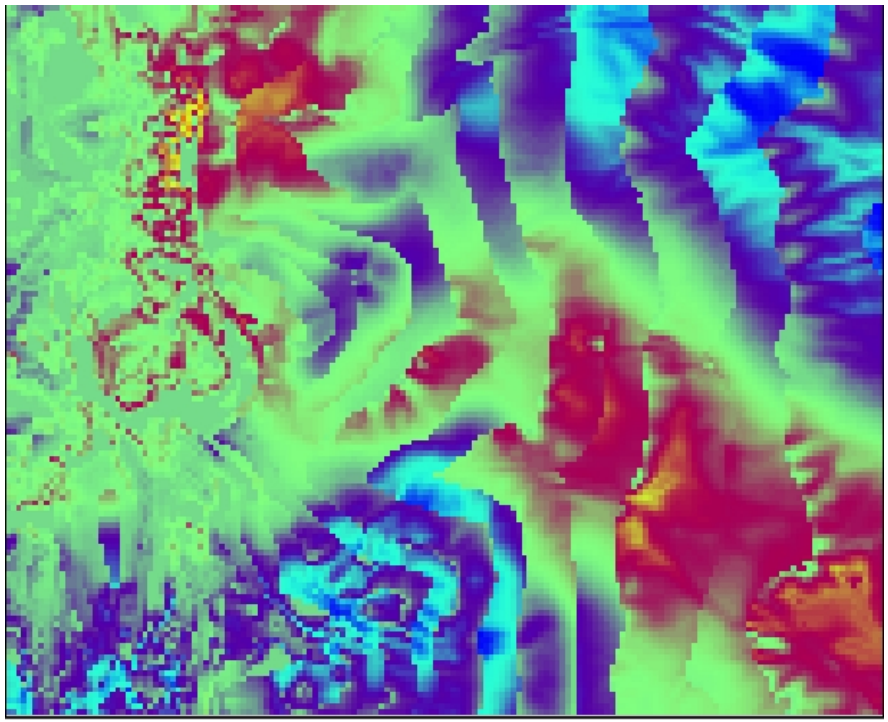
Vertical Interpolation

- Impact: lowest few levels
- use_surface = F
- η level 1

- Theta (-11 K blue, 0 K red)



- U (-3 m/s blue, 4 m/s red)



Soil Level Interpolation

- The WRF model supports several Land Surface schemes:
  - sf_surface_physics = 1, Slab scheme
  - 5 layers
  - Defined with thicknesses: 1, 2, 4, 8, 16 cm

Soil Level Interpolation

- The WRF model supports several Land Surface schemes:
  - sf_surface_physics = 2, Unified Noah scheme
  - 4 layers
  - Defined with layers: 0-10, 10-40, 40-100, 100-200 cm

Soil Level Interpolation

- The WRF model supports several Land Surface schemes:
  - sf_surface_physics = 3, RUC scheme (wilting perturbation)
  - 6 levels
  - Defined at levels: 0, 5, 20, 40, 160, 300 cm

Soil Level Interpolation

- The WRF model supports several Land Surface schemes:
  - sf_surface_physics = 7, PX scheme
  - 2 layers
  - Defined with layers: 0-1, 1-100 cm

Soil Level Interpolation

- The real program accepts soil temperature and soil data from metgrid with an **arbitrary vertical distribution** (though it is explicitly defined in the ungrib Vtable via the naming convention)
- Vertical **interpolation is linear in depth** below ground, where "layers" are assumed defined at their mid-point
- Temperature extrapolation:
  - Near or at the surface uses the skin temperature
  - Below the deepest input soil level uses the annual mean temperature (assumed to be at 300 cm)
- Moisture extrapolation uses the closest level

Soil Level Interpolation

- Mismatches in the land sea mask and the masked fields are typical when the **input sources heterogeneous**, though this is mostly handled in the metgrid program
- The “first” level is near or at the model surface, and the “last” level is the deepest of the soil information

Water Temperature Initialization

- **Two types of water temperatures** are input by the ungrib program
  - Identified as a water temperature (SST)
  - Identified as a "ground" temperature, but over water (SKINTEMP)
- The real program is able to **preferably use an SST** over a water body, if the input field exists
- An **in-land water body** capability in WPS is supported in the real program, with both the USGS and the MODIS sources
  - Locations identified as in-land water bodies use a daily-mean 2-m air temperature (if one exists)

Sea-Ice Initialization

- Most first-guess sources of data (such as GFS) **provide a sea-ice** field
- Originally, these were **only flag values**:
  - 0 = no sea-ice
  - 1 = sea-ice
- Some data sets provide a **fractional sea-ice** field
- ARW, users may set an arbitrary SST temperature in the real program, below which the water points are turned to ice

Land Water Mask

- The **distinction** between land and water in the real program follows almost entirely from that **defined by WPS**
- Several **masked fields rely upon this definition**:
  - Land: soil temps, soil moisture, vegetation fraction
  - Water: sea-ice, SST
- If inadequate data exists (usually to support a declaration of a land point), it may be **turned to a water point**
- After all adjustments, **insure** that SST, skin temperature, land mask, soil temp and moisture, and sea ice **all agree**

Nested Processing in real for ARW

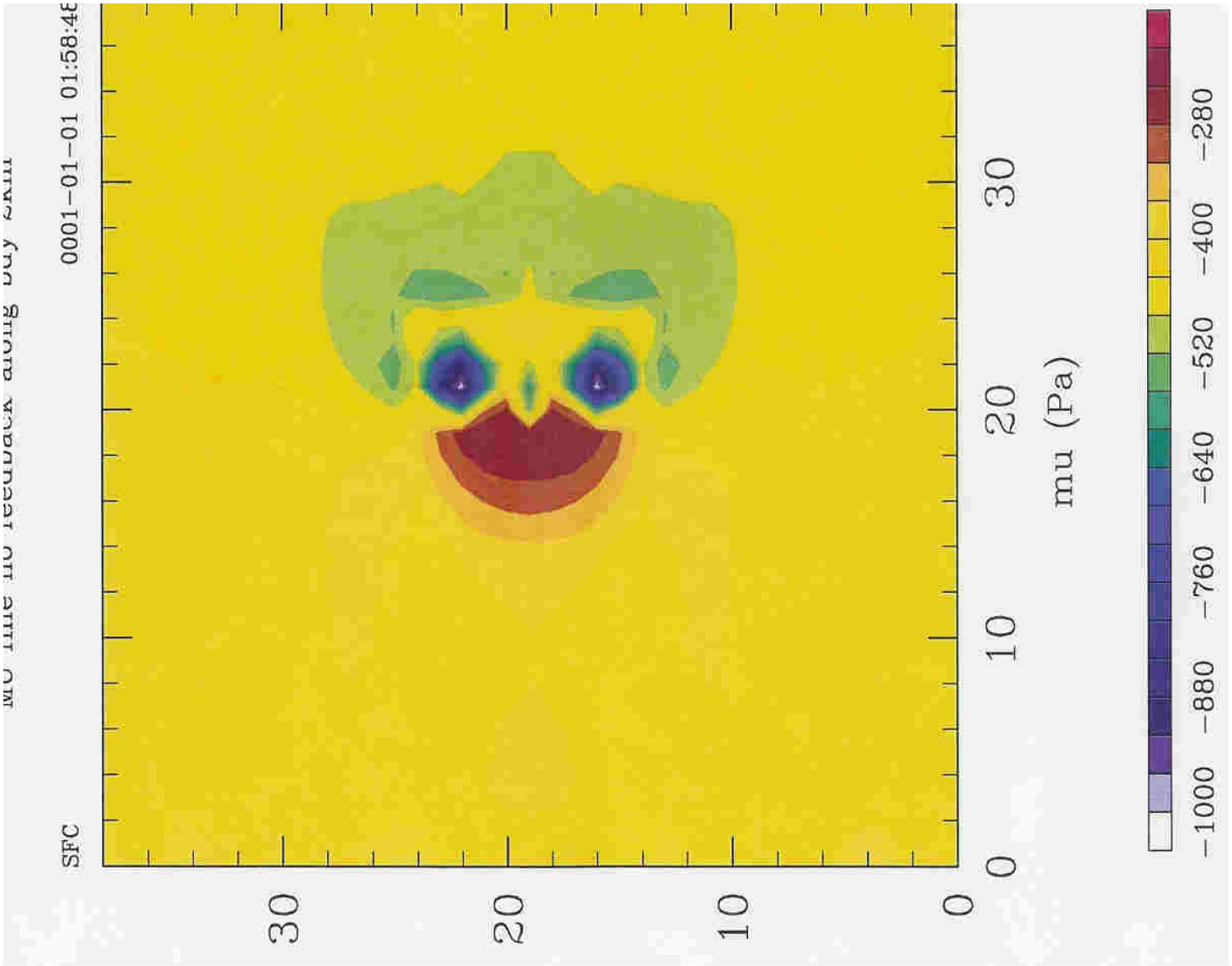
- May read **multiple domain input files** from metgrid
- Requires only the **initial time for the fine domains**, unless doing analysis nudging or SST update
- **No horizontal interpolation** from coarse to fine
- **No consistency checks** between domains (handled in the feedback step for the WRF model)
- A ***wrfinput_d0x*** file is created for each processed input domain
- A **lateral boundary file** is created only for the **most coarse domain**

Real program in a nutshell

- Function
- Required input variables
- Base State for ARW
- Standard generated output
- Optional output
- Vertical interpolation
- Soil level interpolation
- Water temperature initialization
- Sea-ice initialization
- Land/Water mask
- Nested processing in real for ARW

ARW Nesting

Dave Gill



Nesting Basics - What is a nest

- A nest is a ***finer-resolution*** model run. It may be ***embedded*** simultaneously within a coarser-resolution (parent) model run, or ***run independently*** as a separate model forecast.
- The nest ***covers a portion*** of the parent domain, and is driven along its ***lateral boundaries*** by the parent domain.
- Nesting enables running at finer resolution without the following problems:
  - Uniformly high resolution over a large domain - **prohibitively expensive**
  - High resolution for a very small domain with **mismatched** time and spatial lateral boundary conditions

Nesting Basics - ARW

- One-way nesting via multiple model forecasts
- One-way nesting with a single model forecast, without feedback

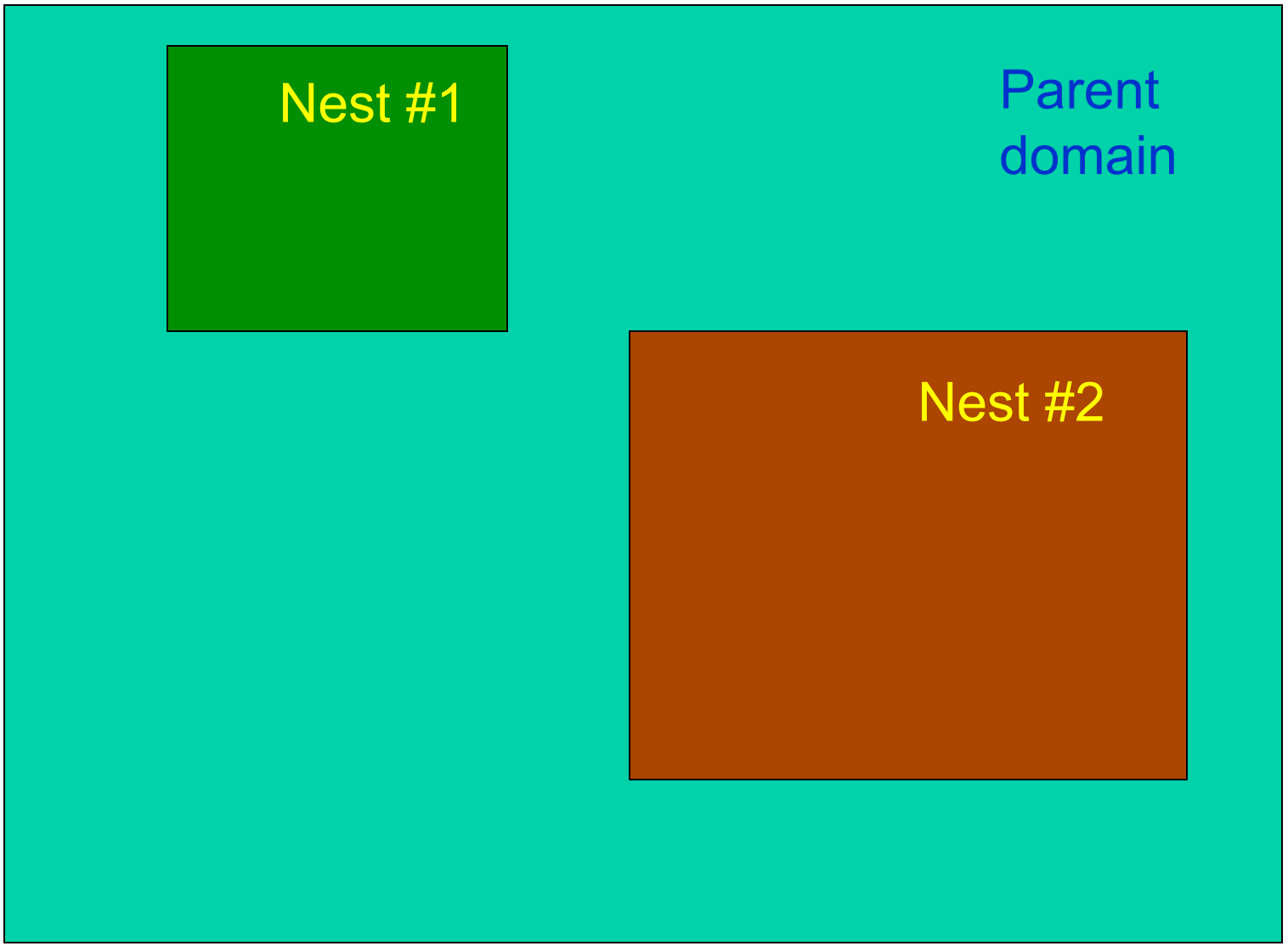
- One-way/two-way nesting with a single input file, all fields interpolated from the coarse grid
- One-way/two-way nesting with multiple input files, each domain with a full input data file
- One-way/two-way nesting with the coarse grid data including all meteorological fields, and the fine-grid domains including only the static files

- One-way/two-way nesting with a specified move for each nest
- One-way/two-way nesting with an automatic move on the nest determined through 500 mb low tracking

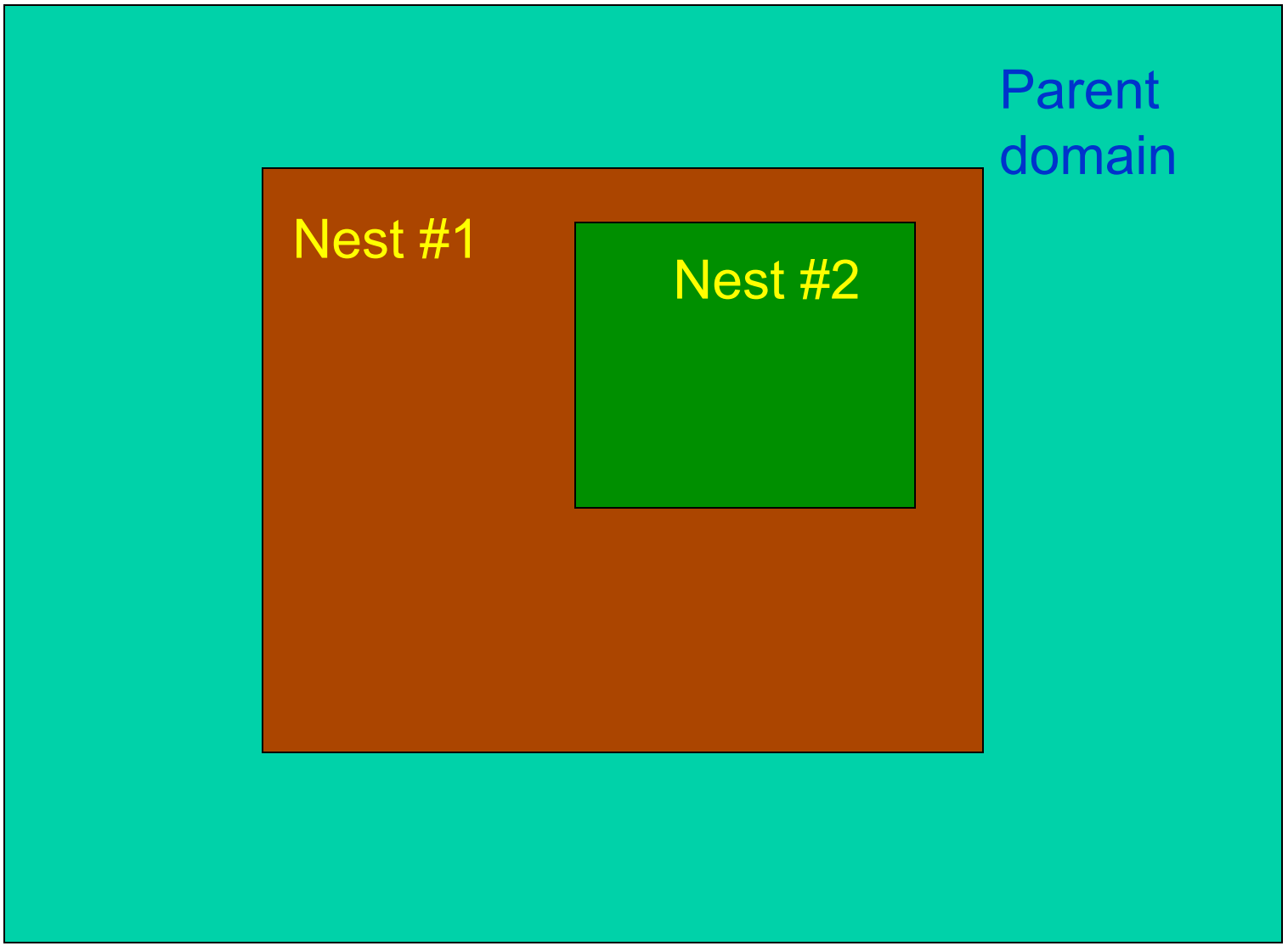
Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting

Two nests on the same “level”, with a common parent domain

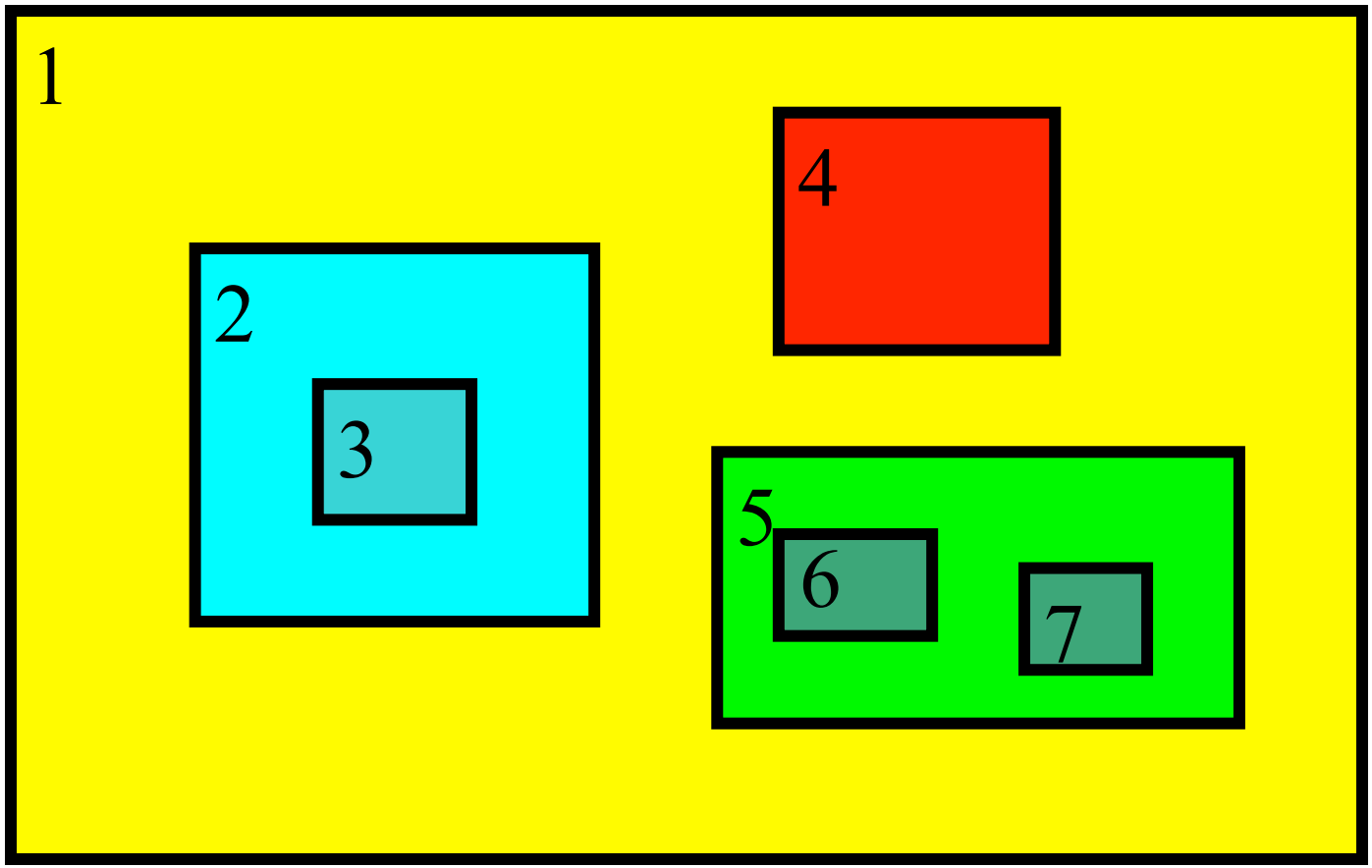


Two levels of nests, with nest #1 acting as the parent for nest #2



These are all OK

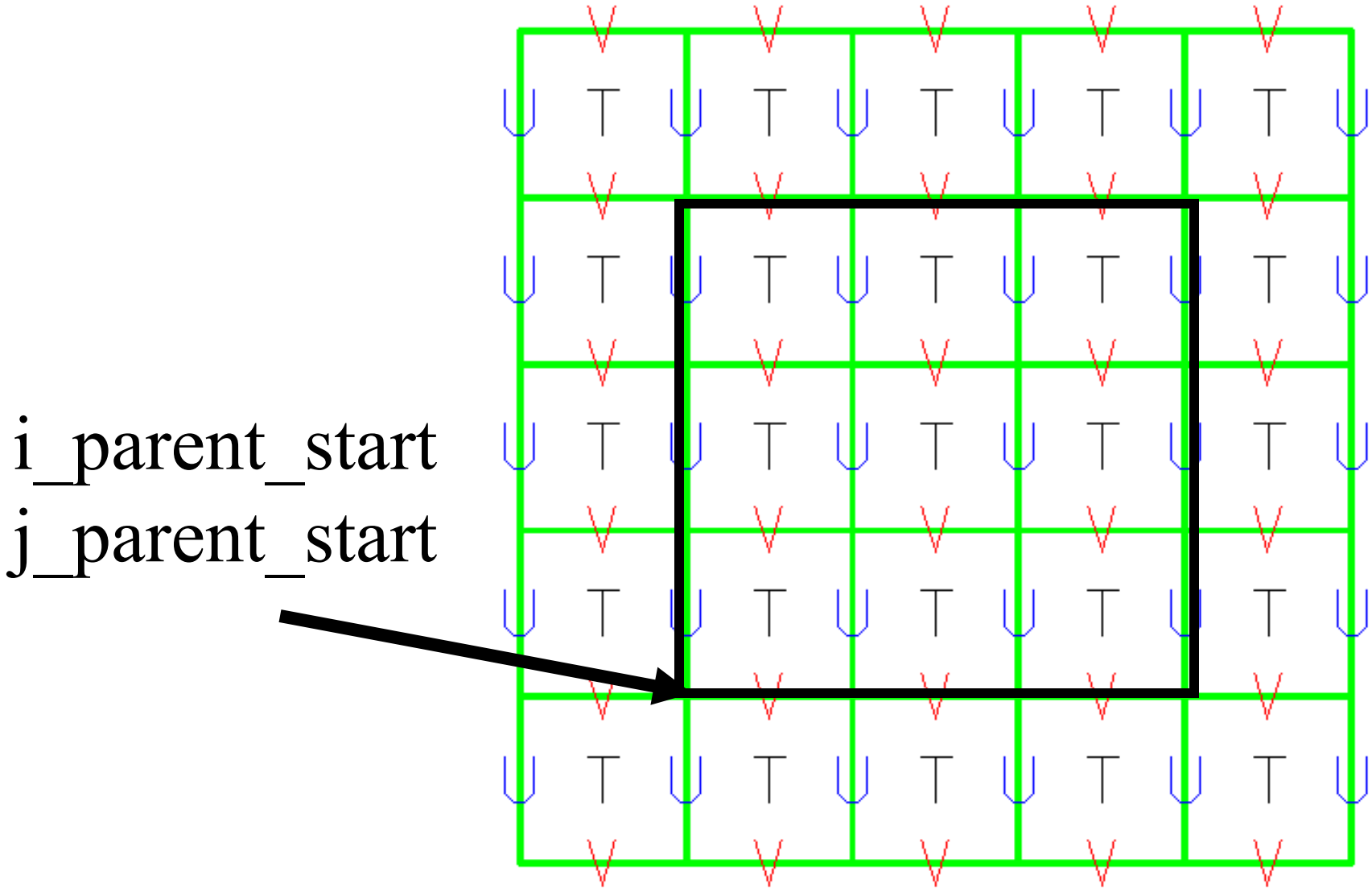
Telescoped to any depth
Any number of siblings



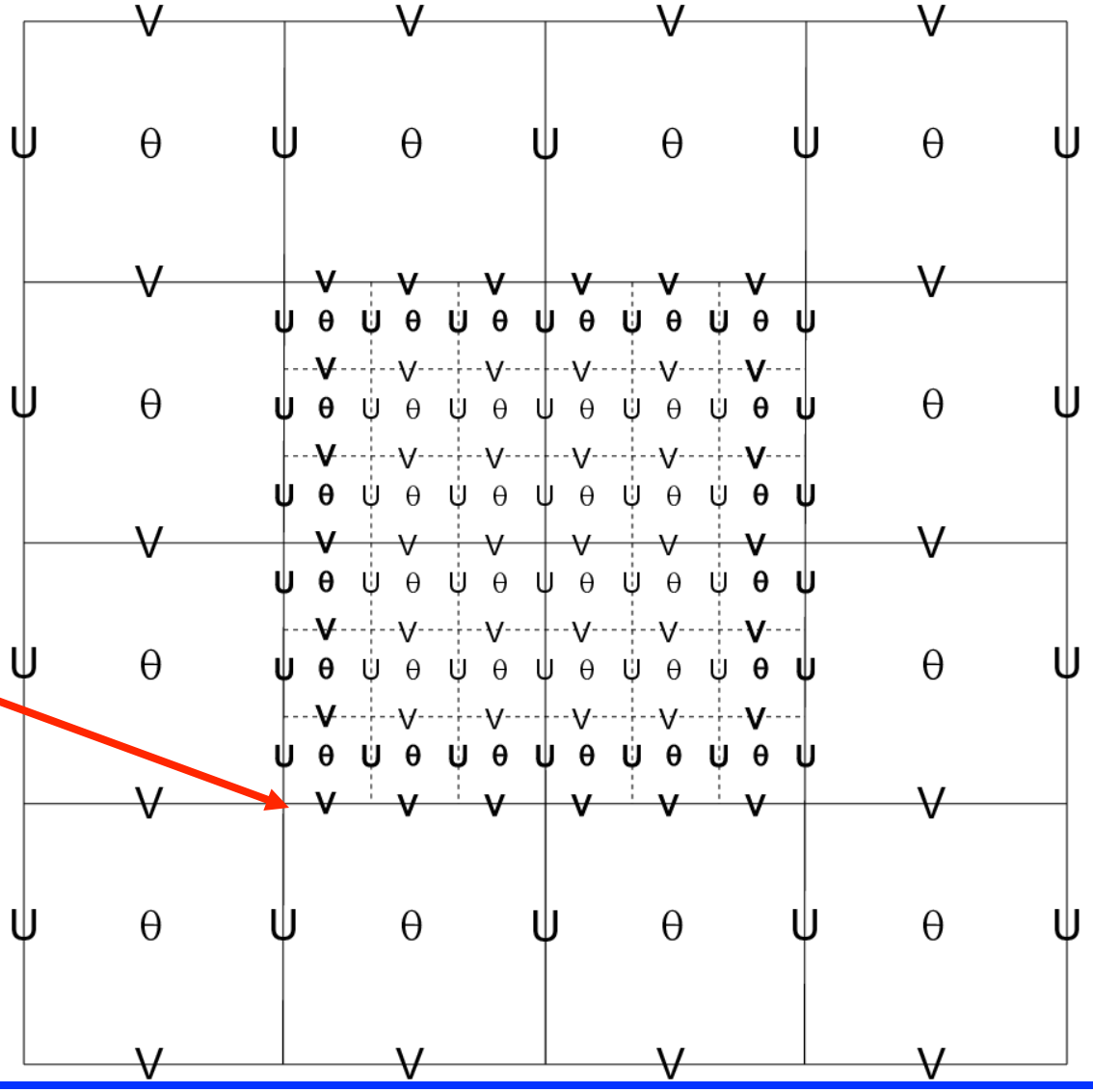
Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting

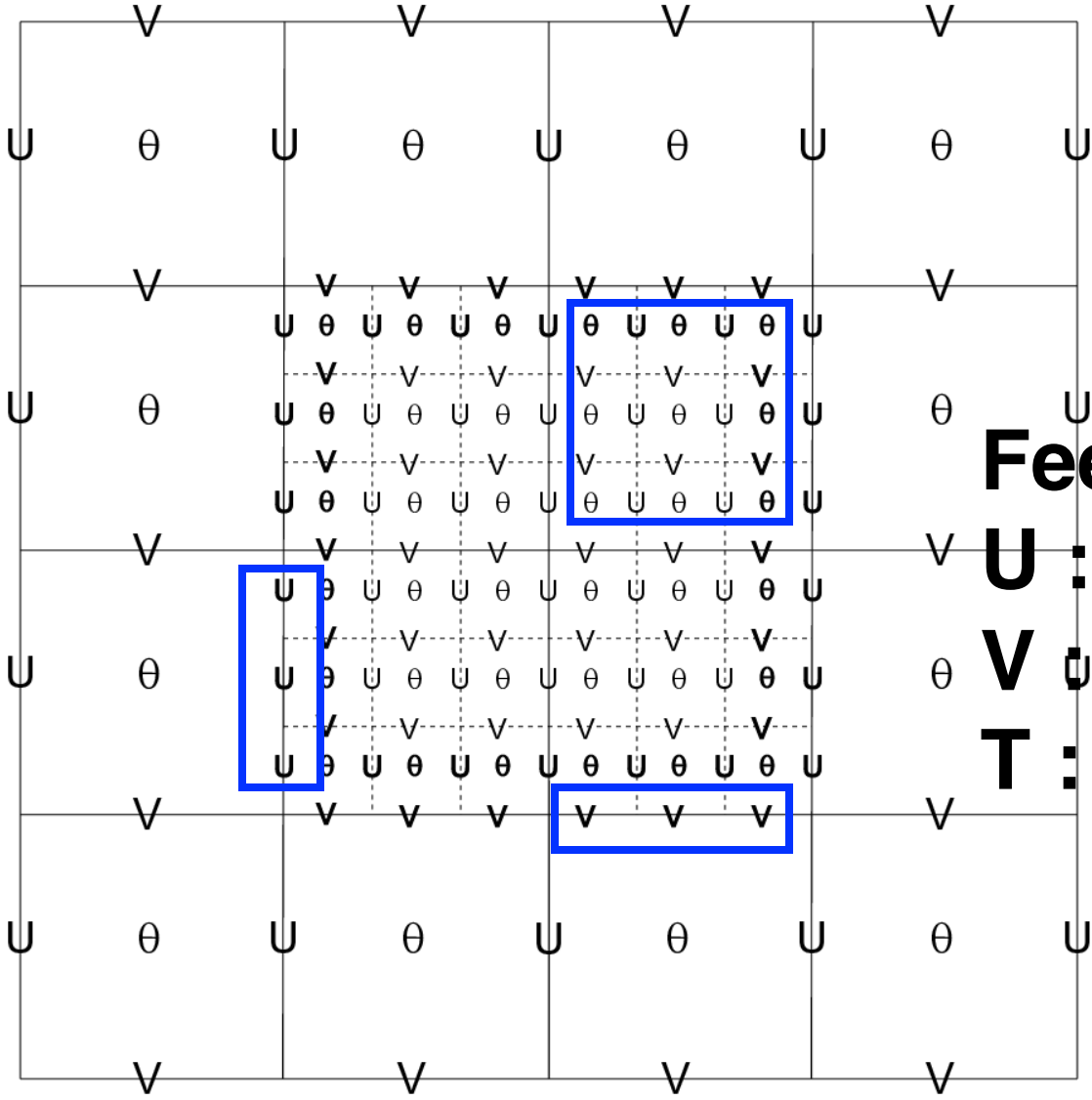
ARW Coarse Grid Staggering



ARW Coarse Grid Staggering 3:1 Ratio



ARW Coarse Grid Staggering 3:1 Ratio



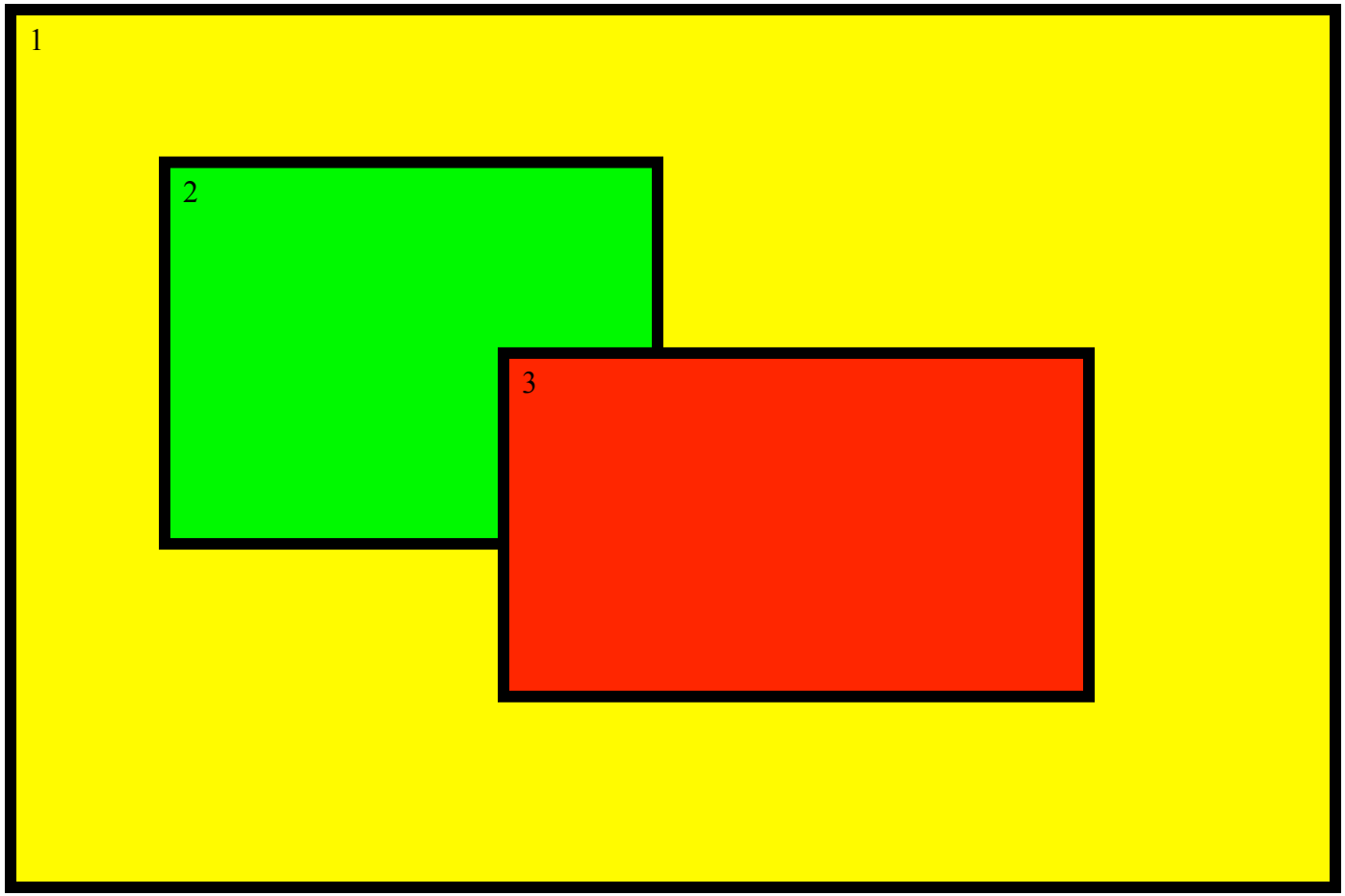
Feedback:
U : column
V : row
T : cell

Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting

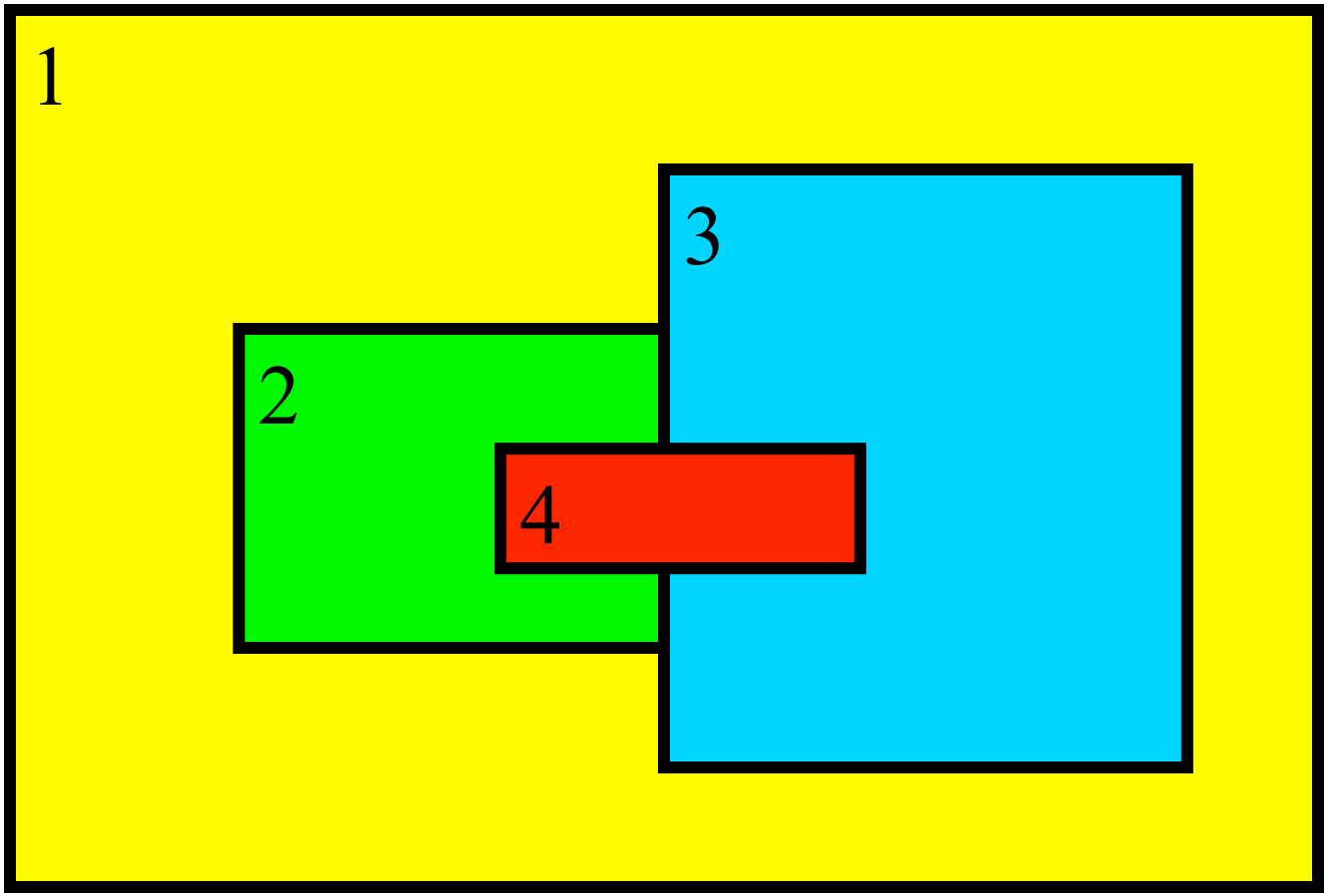
Not OK for 2-way

Child domains *may not* have overlapping points in the parent domain (1-way nesting excluded).



Not OK either

Domains have one, and only one, parent -
(domain 4 is NOT acceptable even with 1-way nesting)



Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting

Nesting Performance

- The size of the nested domain may need to be chosen with computing performance in mind.
- Assuming a 3:1 ratio and the same number of grid cells in the parent and nest domains, the fine grid will require 3x as many time steps to keep pace with the coarse domain.
- A simple nested domain forecast is approximately 4x the cost of just the coarse domain.
- Don't be *cheap* on the coarse grid, doubling the CG points results in only a 25% nested forecast time increase.

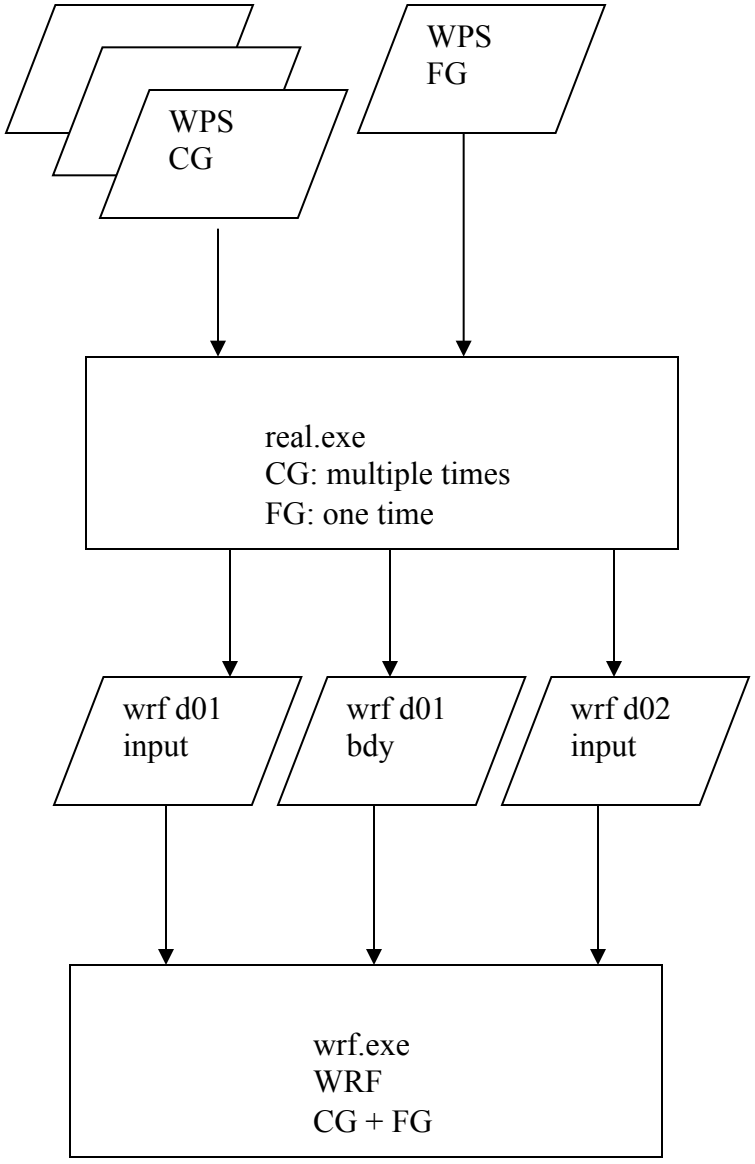
ARW: 2-Way Nest with 2 Inputs

Coarse and fine grid domains must start at the same time, fine domain may end at any time

Feedback may be shut off to produce a 1-way nest (cell face and cell average)

Any integer ratio for coarse to fine is permitted, odd is usually chosen for real-data cases

Options are available to ingest only the static fields from the fine grid, with the coarse grid data horizontally interpolated to the nest



ARW: 2-Way Nest with 2 Inputs

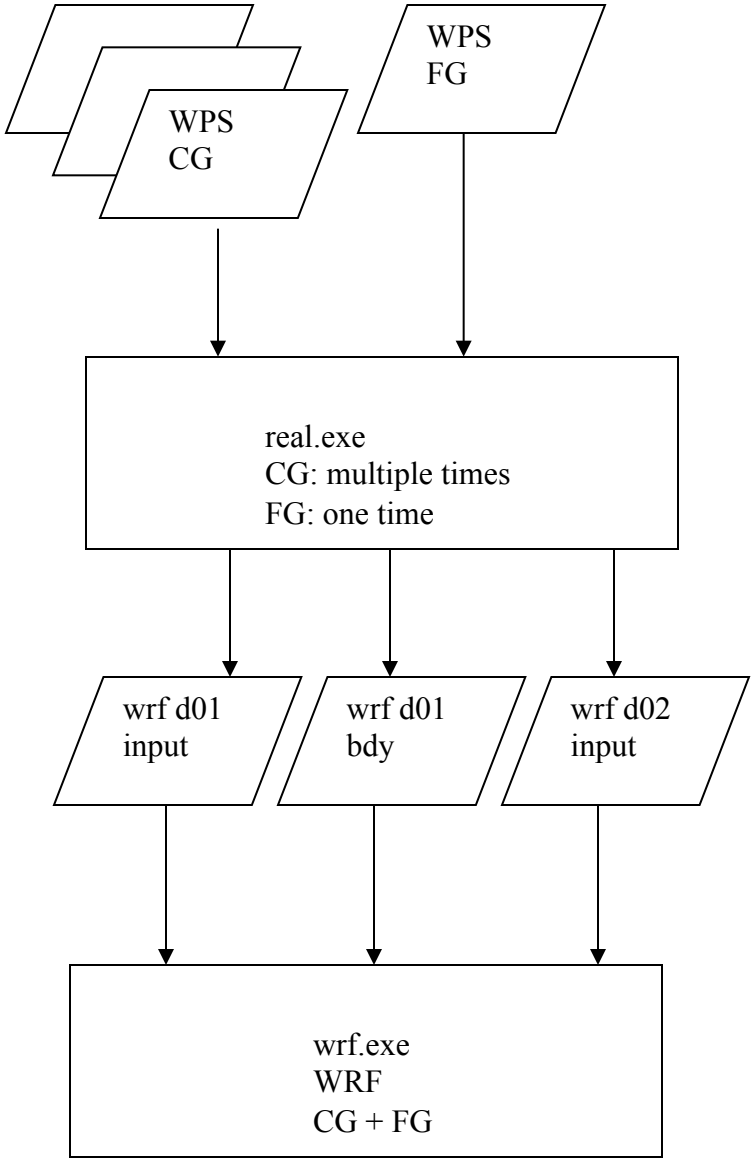
Vertical nesting requires ndown

Usually the same physics are run on all of the domains (excepting cumulus)

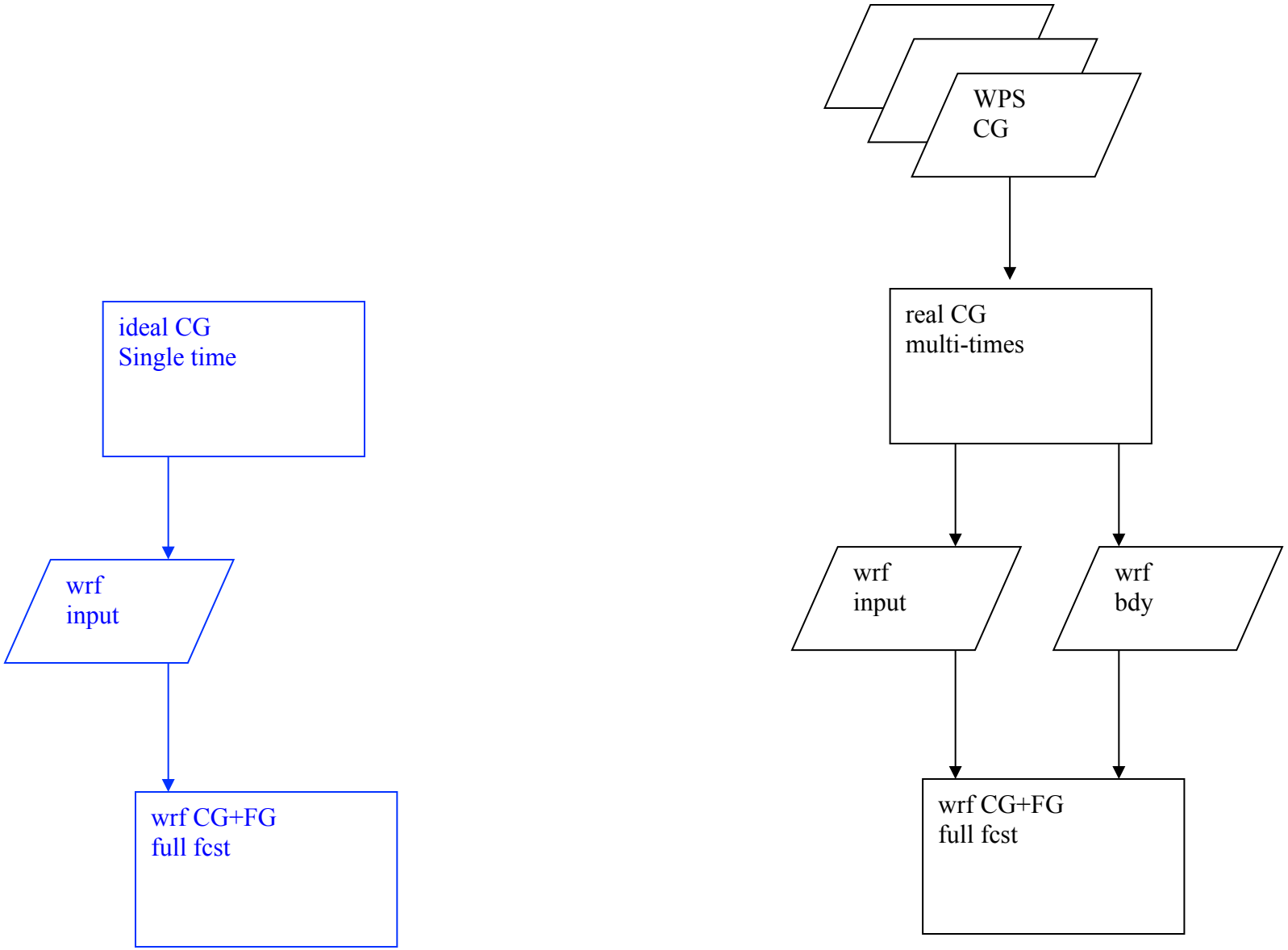
The grid distance ratio is not strictly tied to the time step ratio

Topography smoothly ramps from coarse grid to the fine grid along the interface along the nest boundary

All fine grids must use the nested lateral boundary condition



ARW: 2-Way Nest with 1 Input



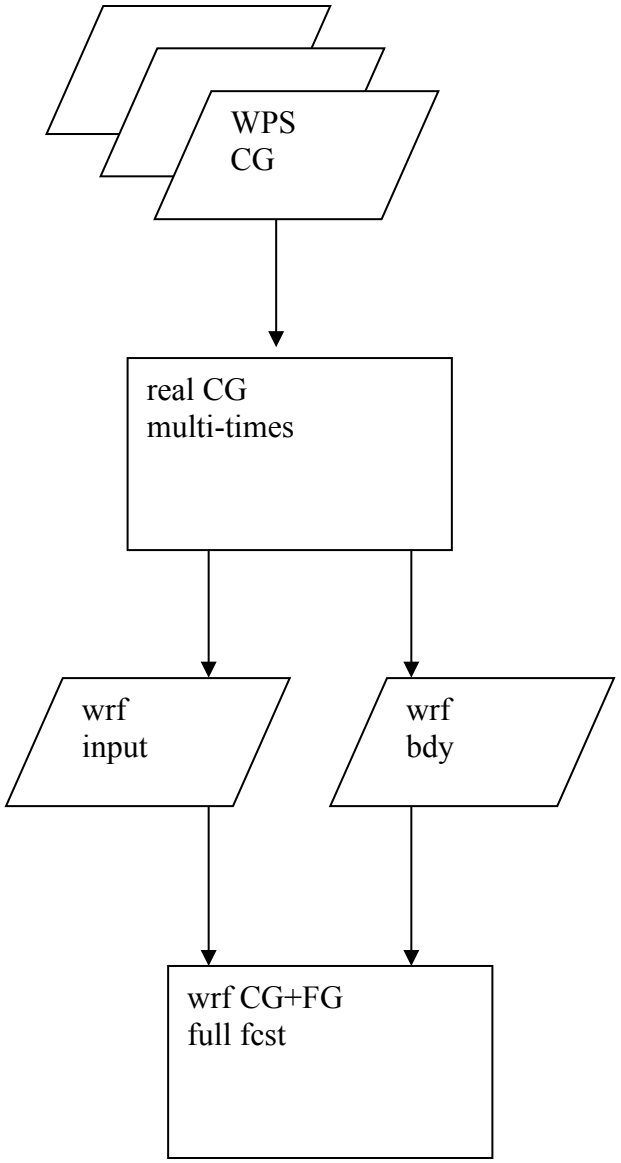
ARW: 2-Way Nest with 1 Input

A single namelist column entry is tied to each domain

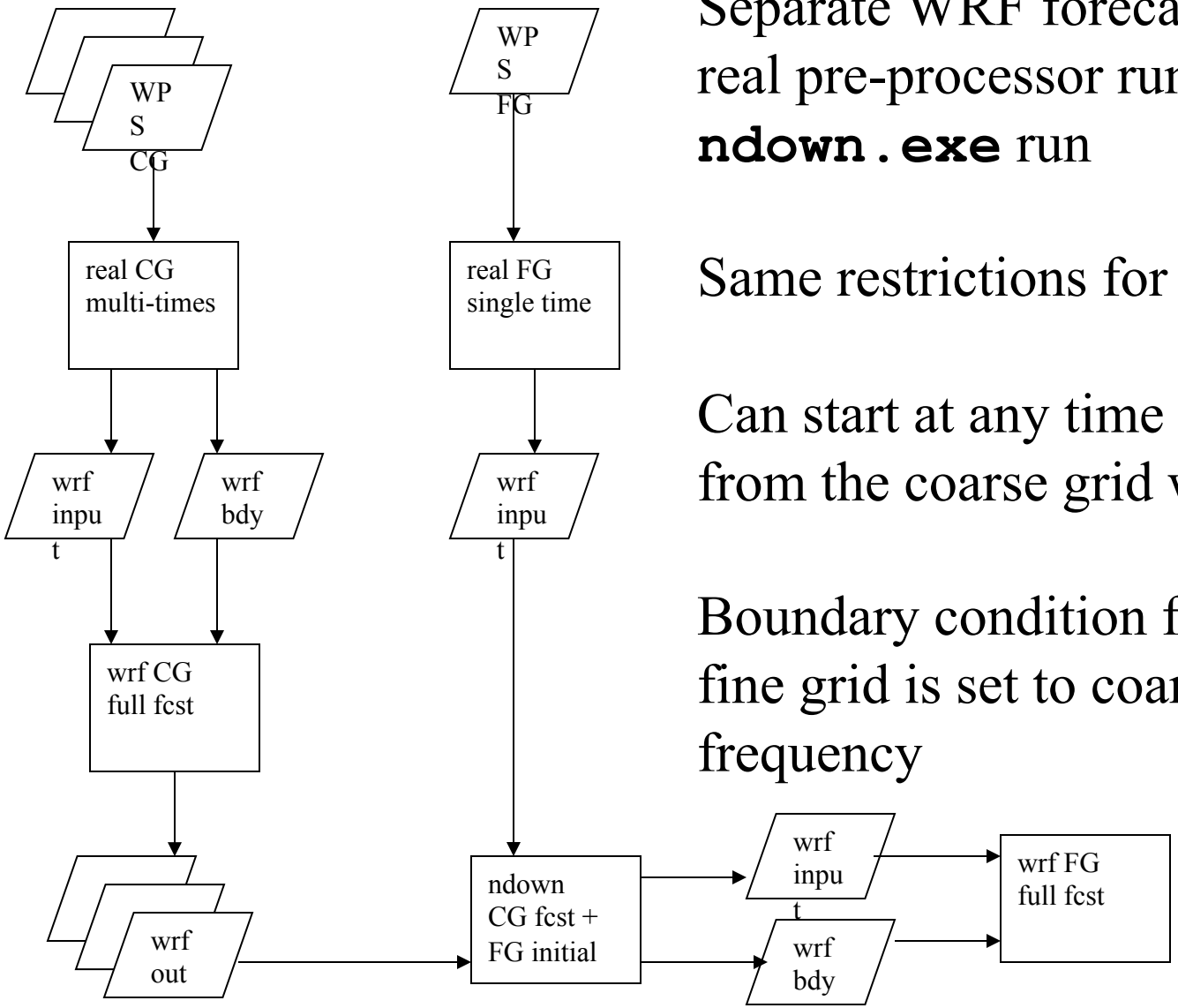
The horizontal interpolation method, feedback, and smoothing are largely controlled through the Registry file

For a 3:1 time step ratio, after the coarse grid is advanced, the lateral boundaries for the fine grid are computed, the fine grid is advanced three time steps, then the fine grid is fed back to the coarse grid (recursively, depth first)

Helpful run*.tar files are located in the **`./WRFV3/test/em_real`** directory



ndown: 1-Way Nest with 2 Inputs



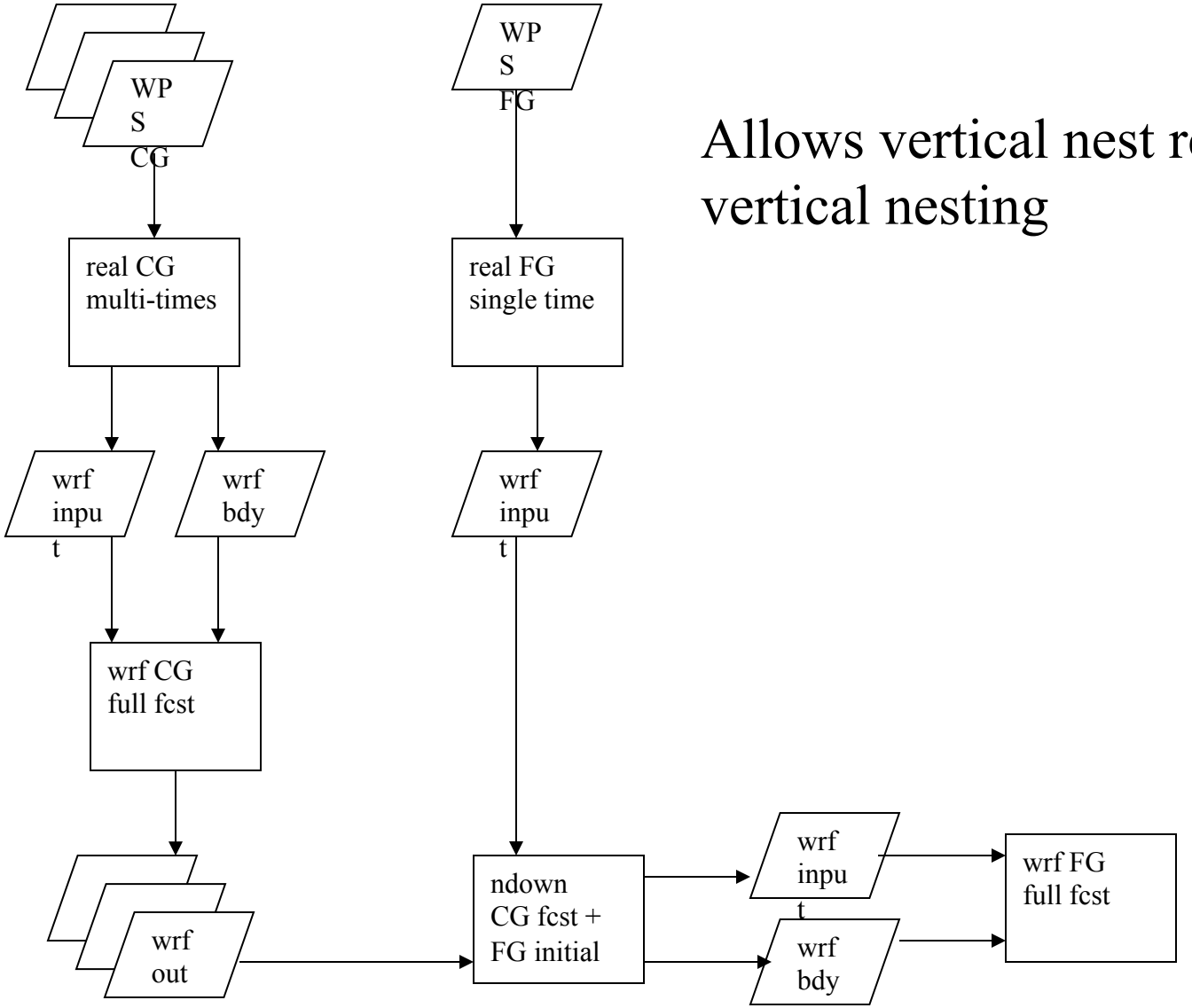
Separate WRF forecast runs, separate real pre-processor runs, intervening **`ndown.exe`** run

Same restrictions for nest ratios

Can start at any time that an output time from the coarse grid was created

Boundary condition frequency for the fine grid is set to coarse grid output frequency

ndown: 1-Way Nest with 2 Inputs



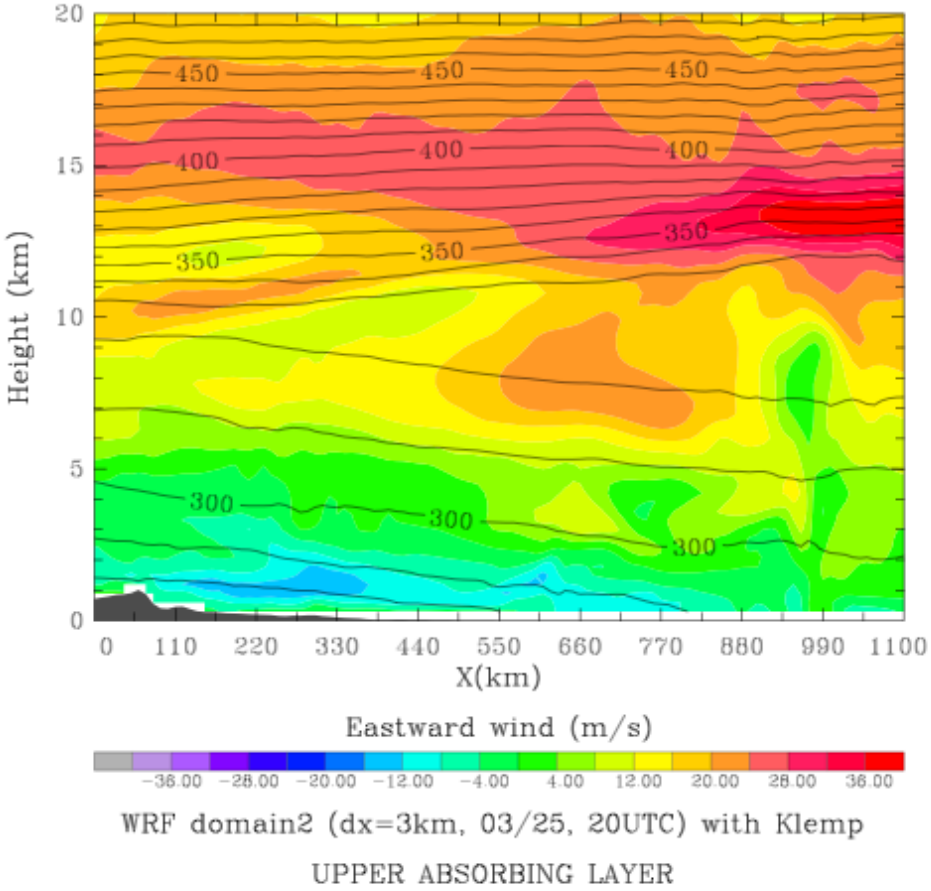
Allows vertical nest refinement ratio: vertical nesting

West East Cross section

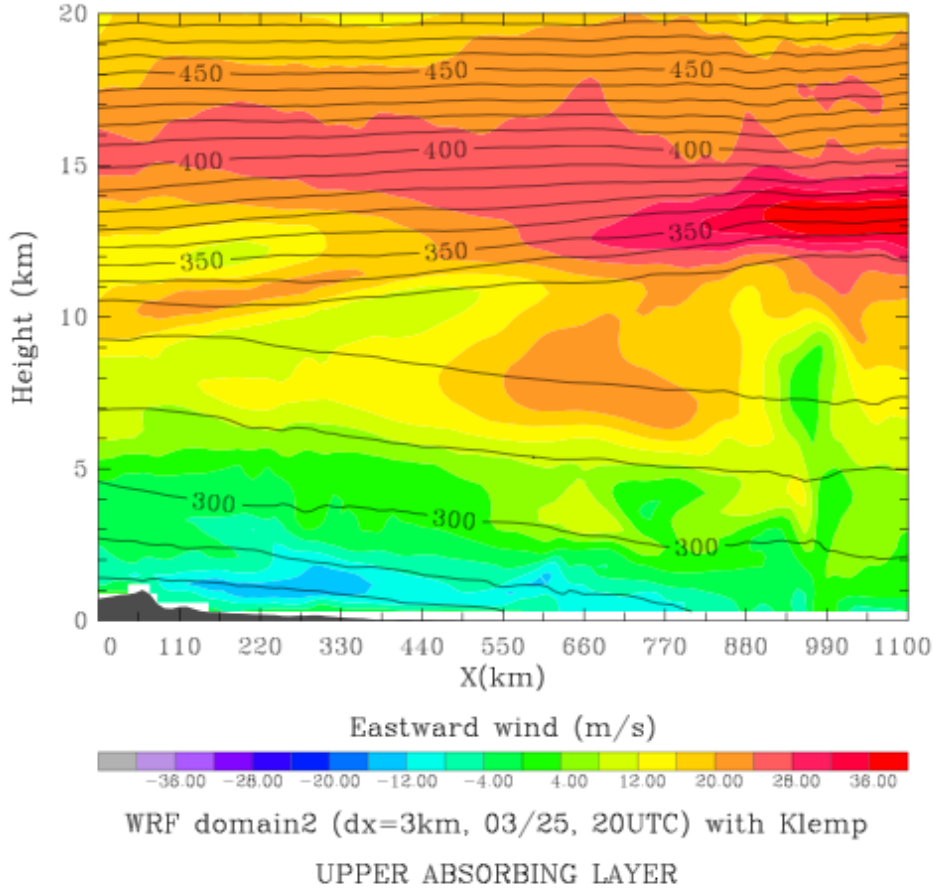
Shaded: v; Contour: theta

6-h Forecast, from Mohamed Moustaoui

Standard Levels



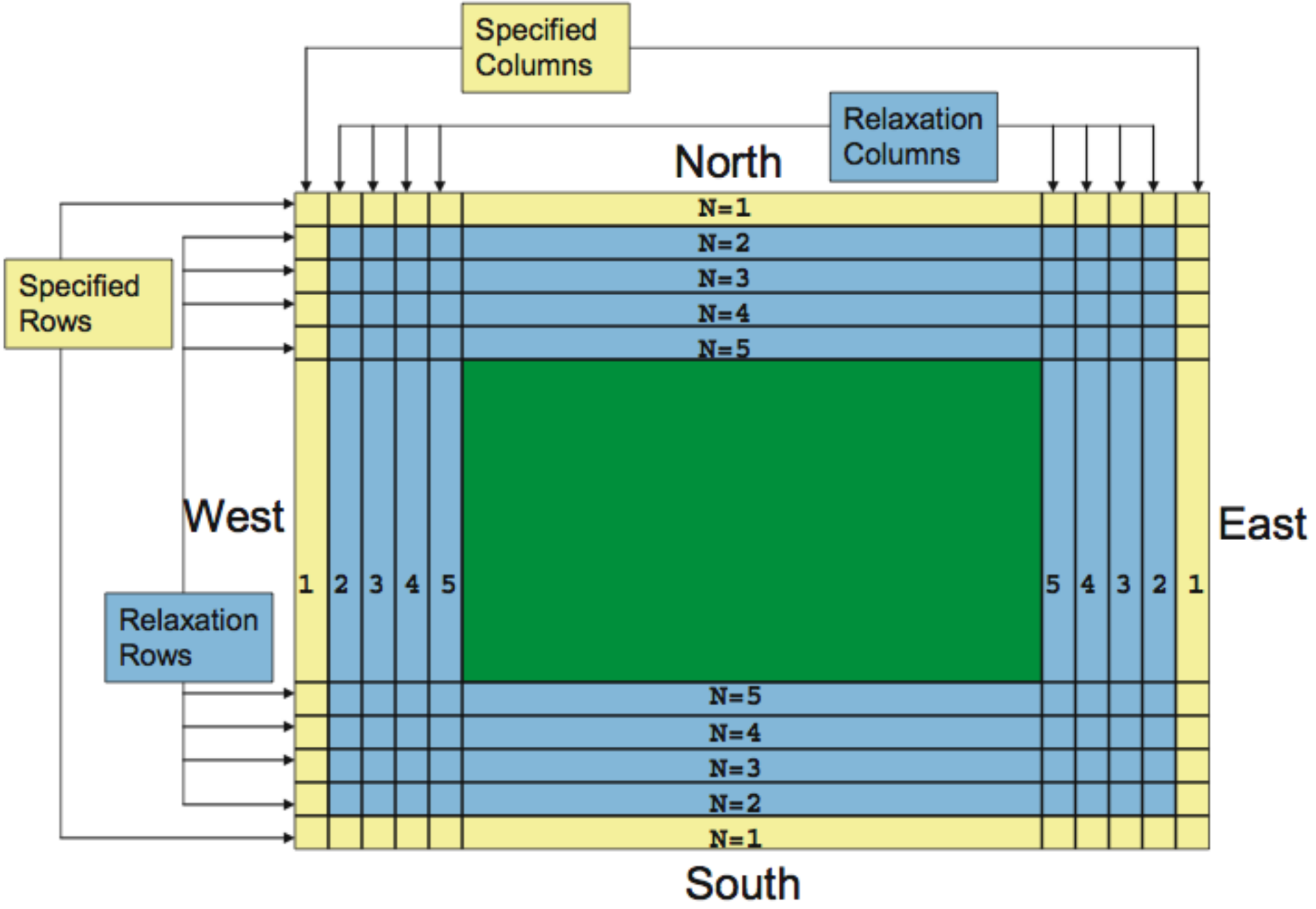
3x Refinement



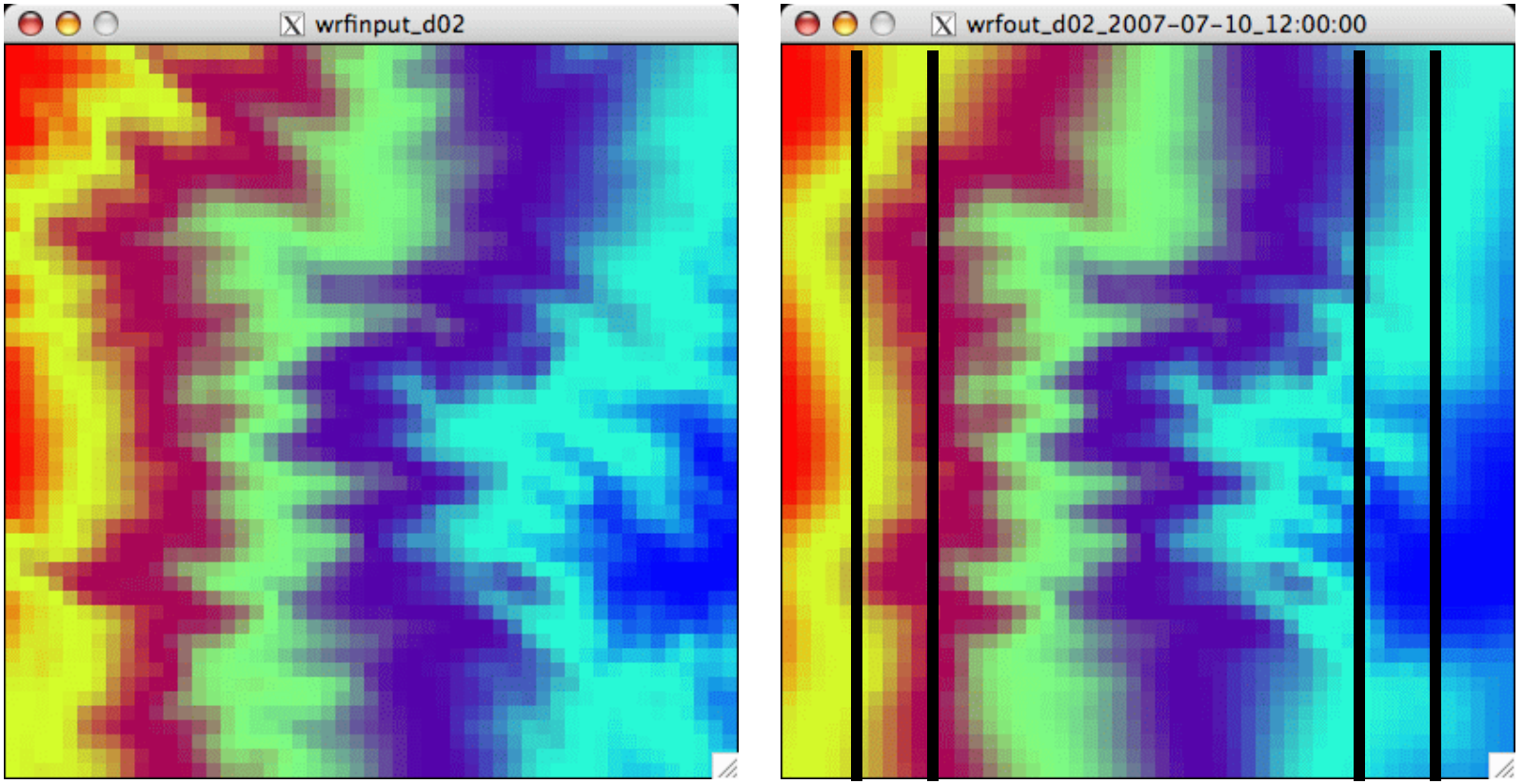
Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting
- Nest logic in WRF source code
- Nest information in the Registry

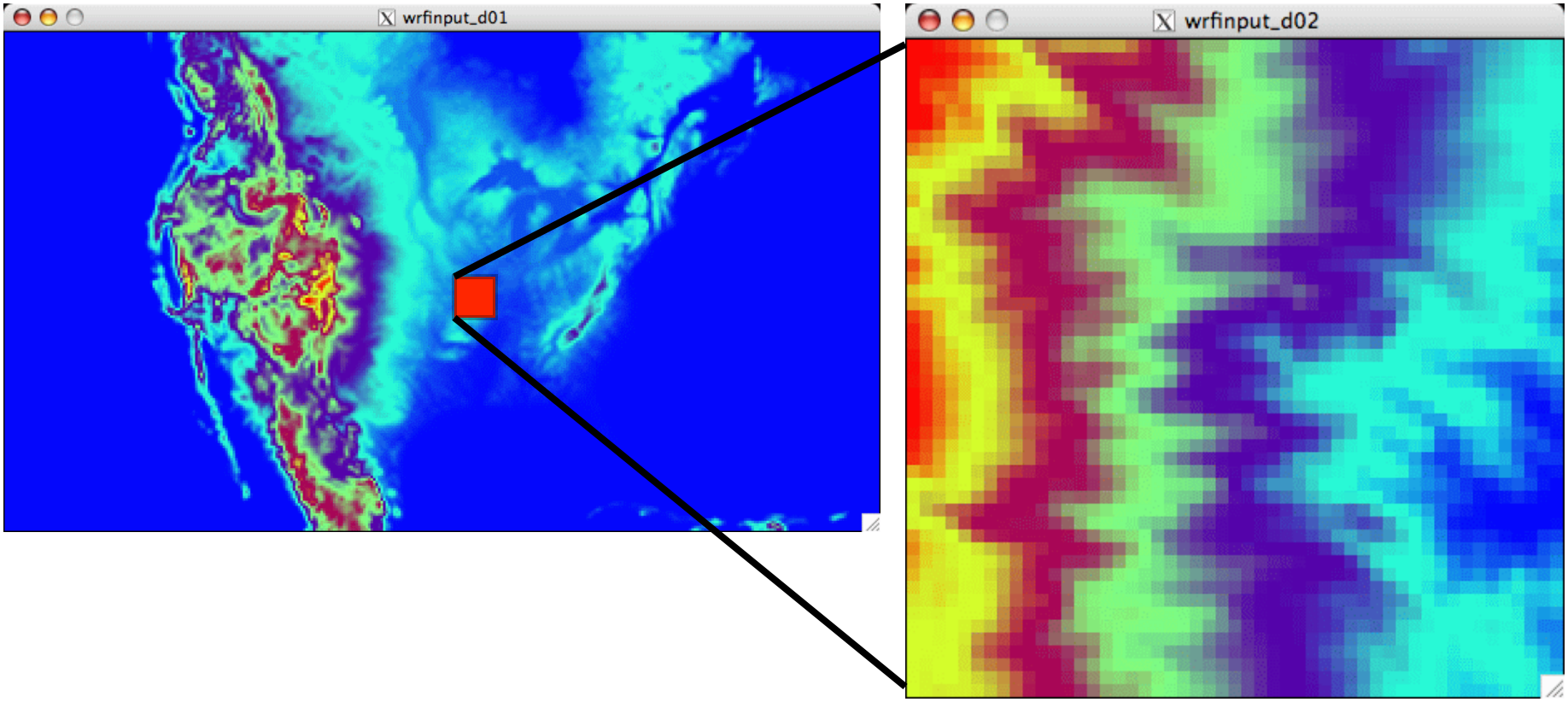
Real-Data Lateral Boundary Condition: Location of Specified and Relaxation Zones



ARW Lateral Smoothing

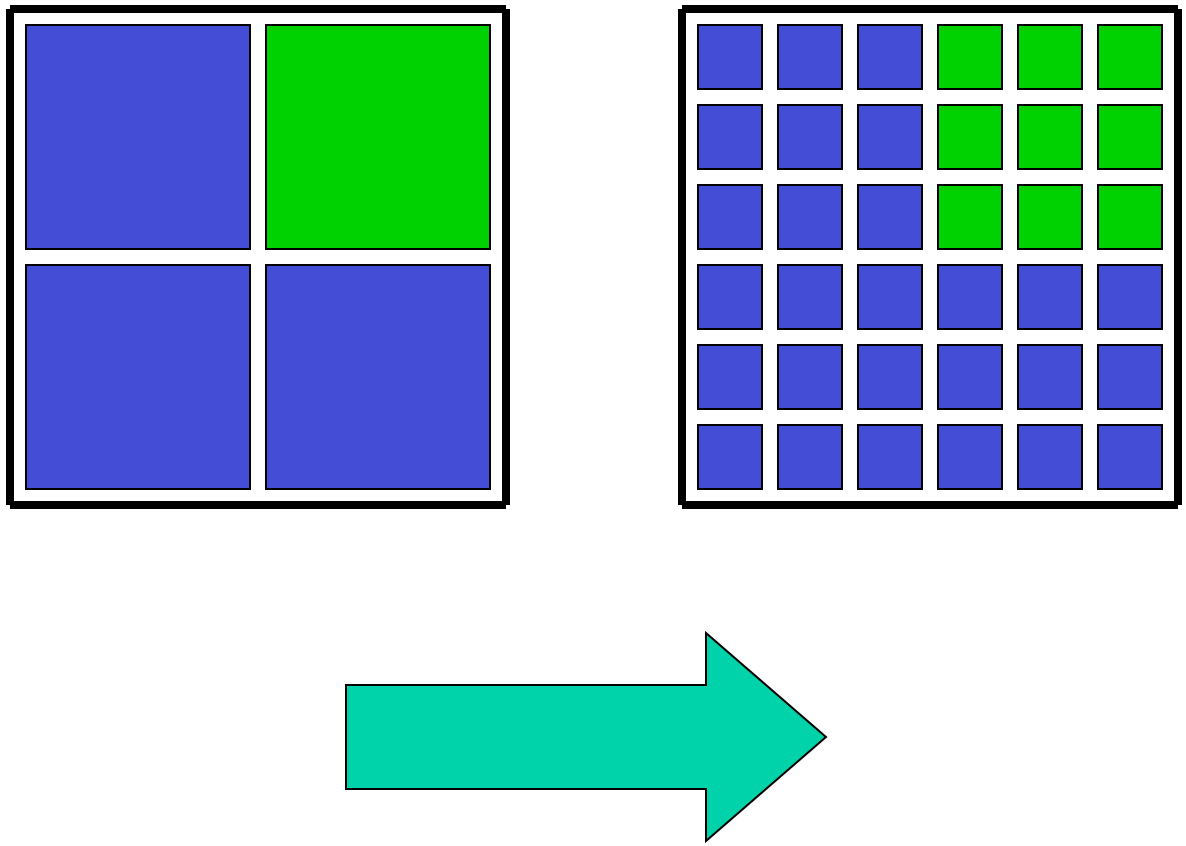


Intermediate Domains

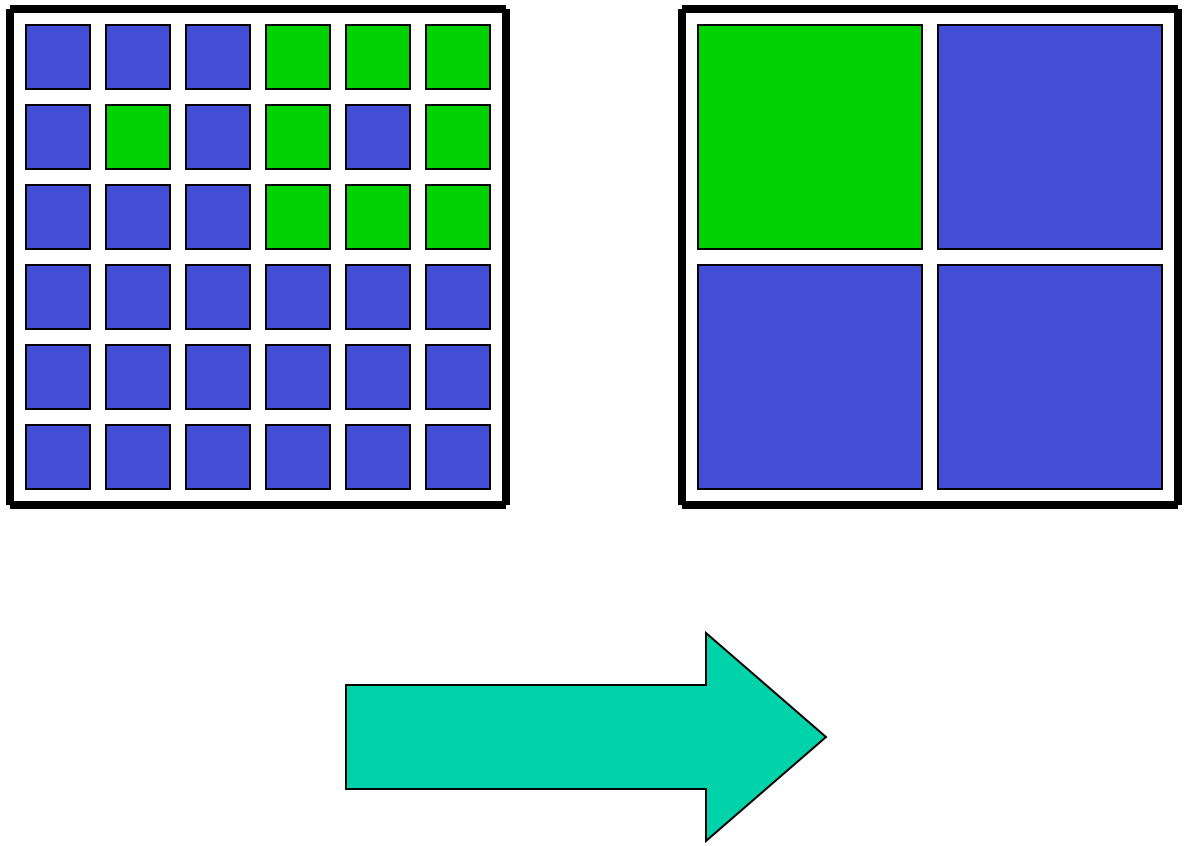


The intermediate domain between a parent and a child is the resolution of the coarse grid over the size of the fine grid. It allows the model to re-decompose the domain among all of the processors.

ARW Masked Interpolation



ARW Masked Feedback



Some Nesting Hints

- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting
- Nest logic in WRF source code
- Nest information in the Registry

What are those “usdf” Options

```
state real u ikjb dyn_em 2 X \
  i01rhusdf=(bdy_interp:dt) \
  "U" "x-wind component" "m s-1”
```

“f” defines what lateral boundary forcing routine (found in **share/interp_fcn.F**) is utilized, colon separates the additional fields that are required (fields must be previously defined in the Registry)

What are those "usdf" Options

```
state real landmask ij misc 1 -  \
  i012rhd=(interp_fcnm)u=(copy_fcnm)\
  "LANDMASK" "LAND MASK (1=LAND, 0=WATER)"
```

"u" and "d" define which feedback (up-scale) and horizontal interpolation (down-scale) routines (found in **share/interp_fcn.F**) are utilized

Default values (i.e. not a subroutine name listed in the parentheses) assume non-masked fields

At compile-time, users select options

What are those “usdf” Options

```
state real ht ij  misc  1  - i012rhdus  "HGT" \
   "Terrain Height"   "m"
```

“s” if the run-time option for smoothing is activated, this field is to be smoothed - only used for the parent of a nest domain, smoothing is in the area of the nest, excluding the outer row and column of the nest coverage

Whether or not smoothing is enabled is a run-time option from the namelist

Special IO Stream #2 Fields

```
state real msft ij  misc  1 -  \   i012rhdu=
   (copy_fcnm)  "MAPFAC_M"   \
   "Map scale factor on mass grid" "”

state real msfu ij  misc  1 X  \  i012rhdu=
   (copy_fcnm)  "MAPFAC_U"   \
   "Map scale factor on u-grid" "”

state real msfv ij  misc  1 Y  \   i012rhdu=
   (copy_fcnm)  "MAPFAC_V"   \
   "Map scale factor on v-grid" ""
```