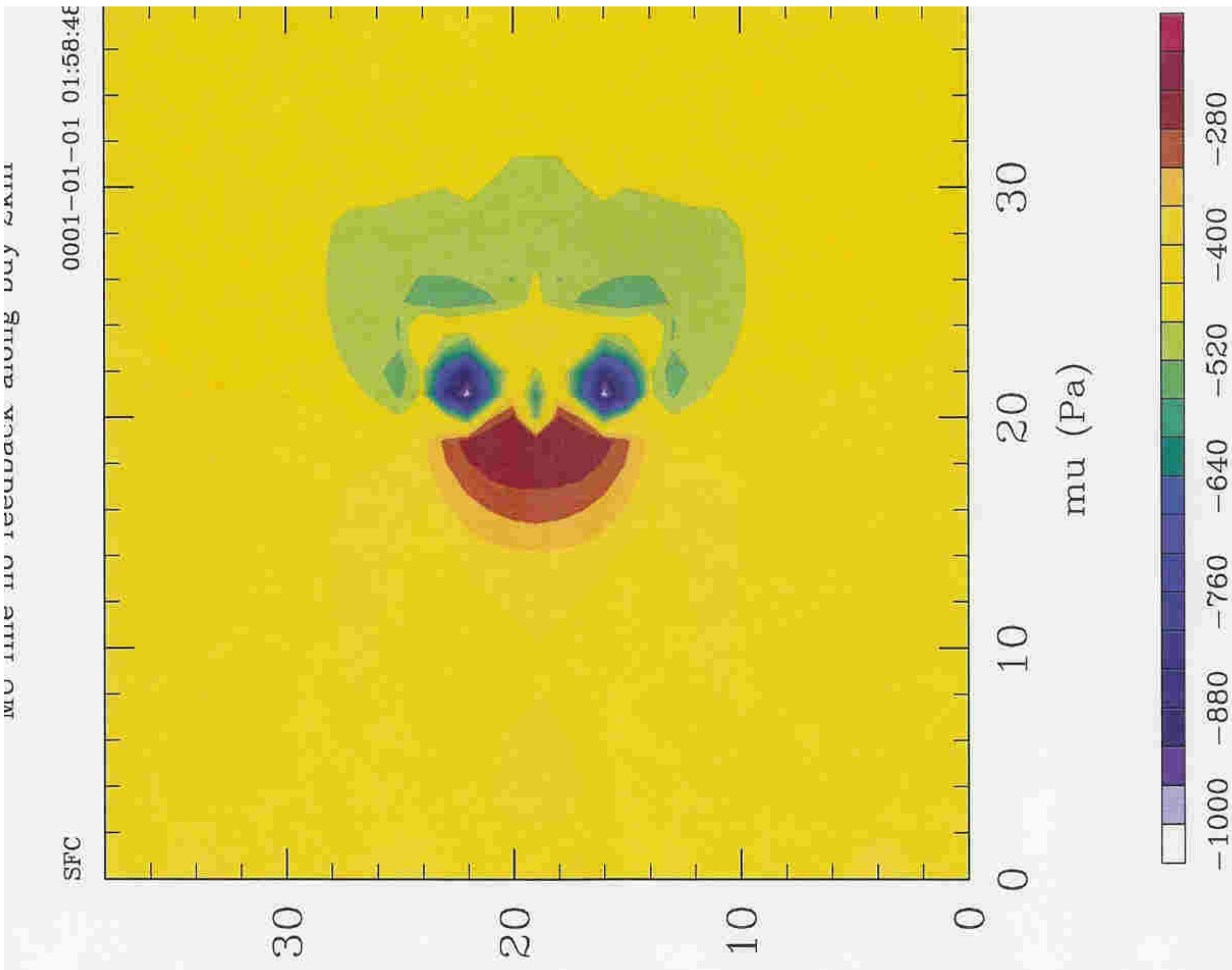
# ARW Nesting

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

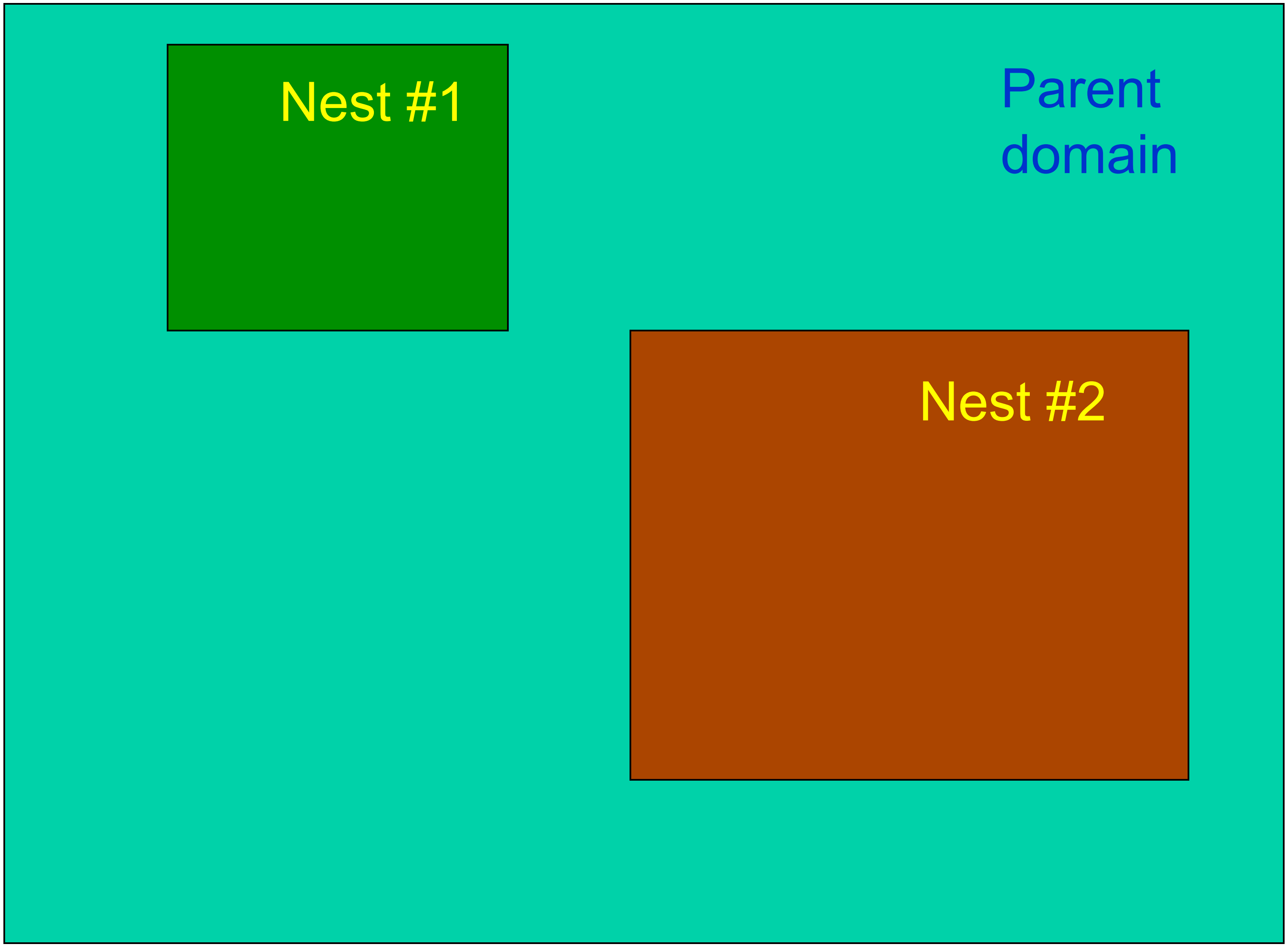
- One-way/two-way nesting with a single input file, all fields interpolated from the coarse grid
- One-way/two-way nesting with multiple input files, each domain with a full input data file
- One-way/two-way nesting with the coarse grid data including all meteorological fields, and the fine-grid domains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

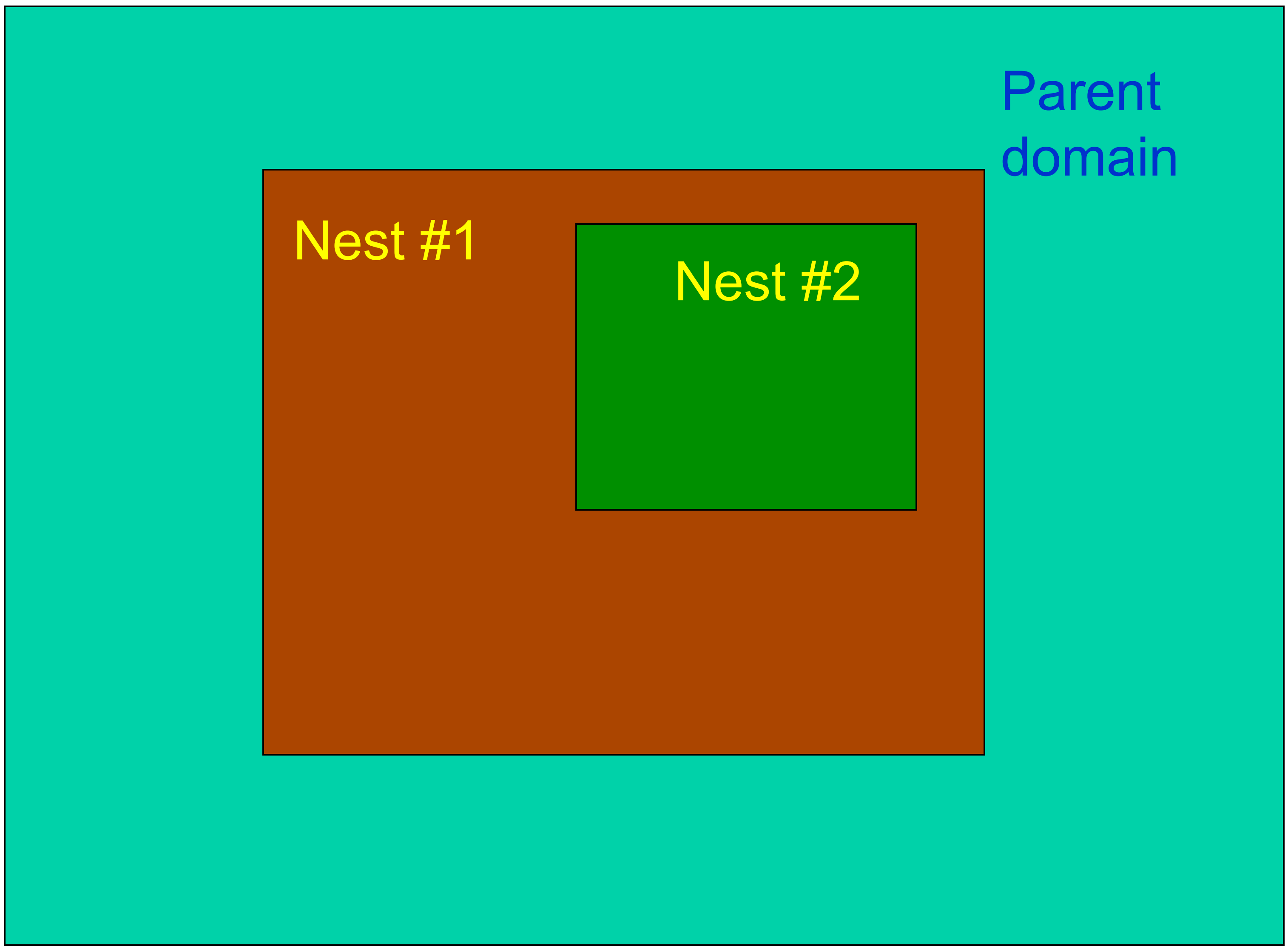
# Some Nesting Hints

- 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

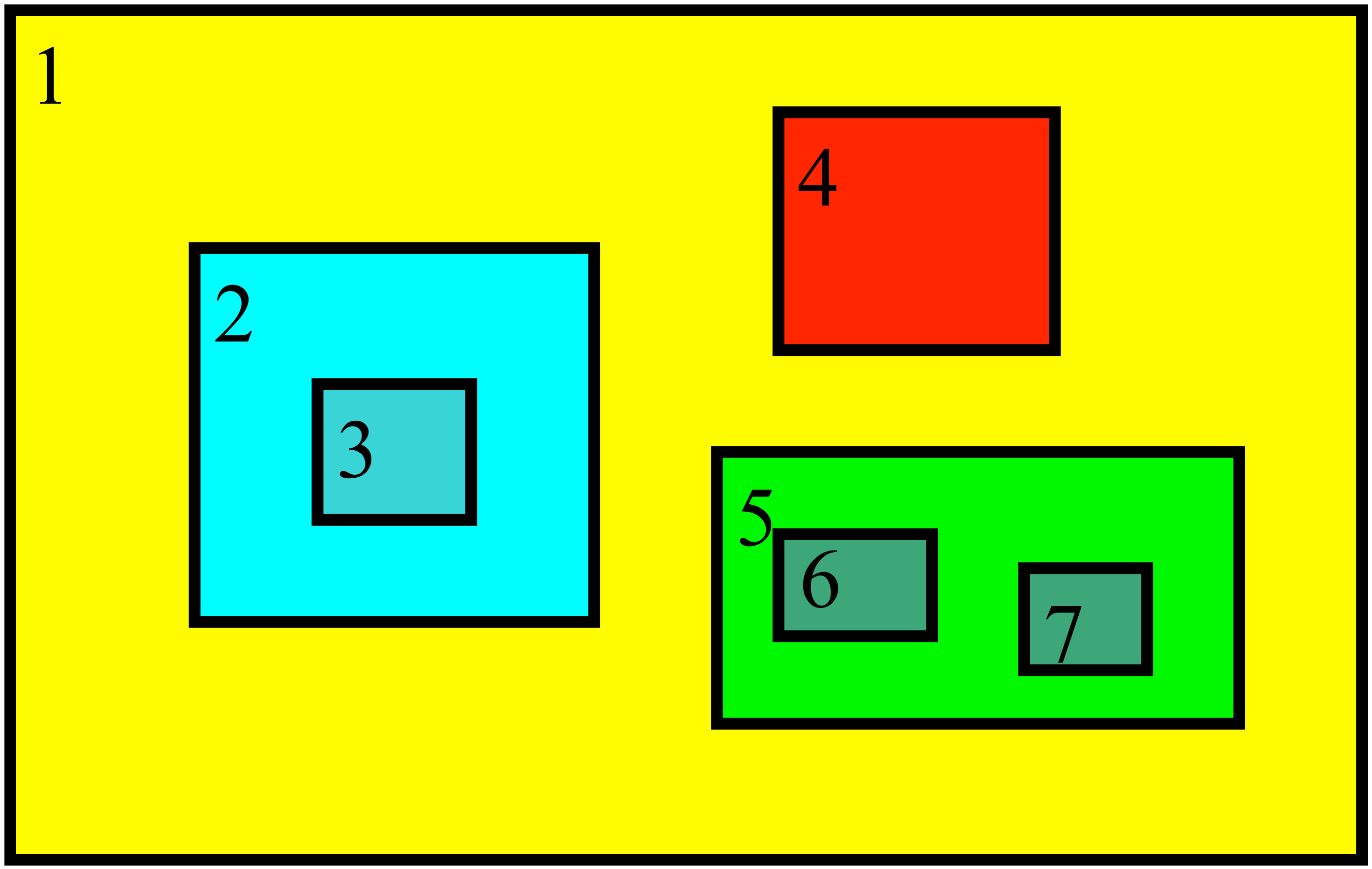


# Two levels of nests, with nest #1 acting as the parent for nest #2



# These are all OK

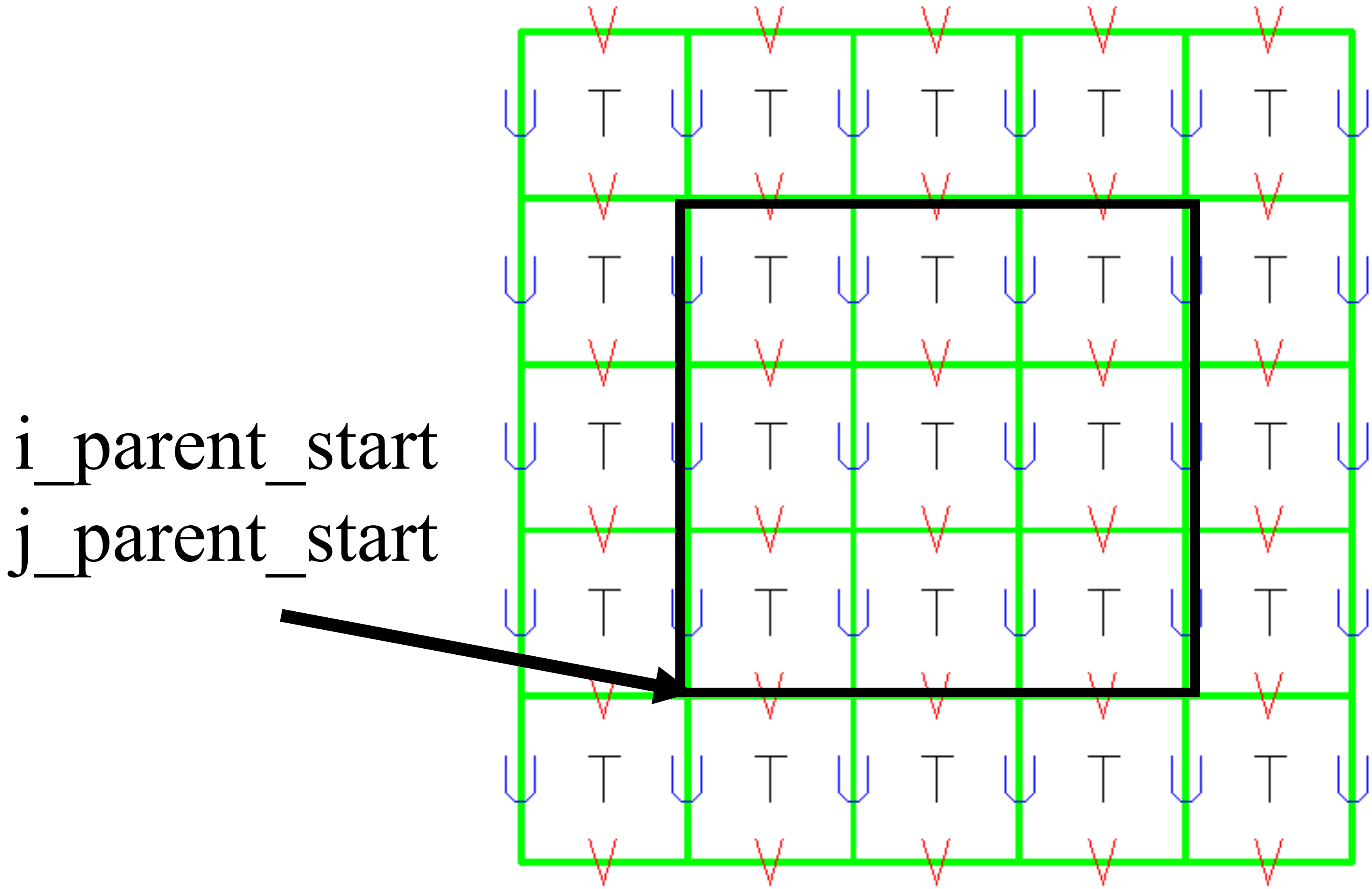
Telescoped to any depth
Any number of siblings



# Some Nesting Hints

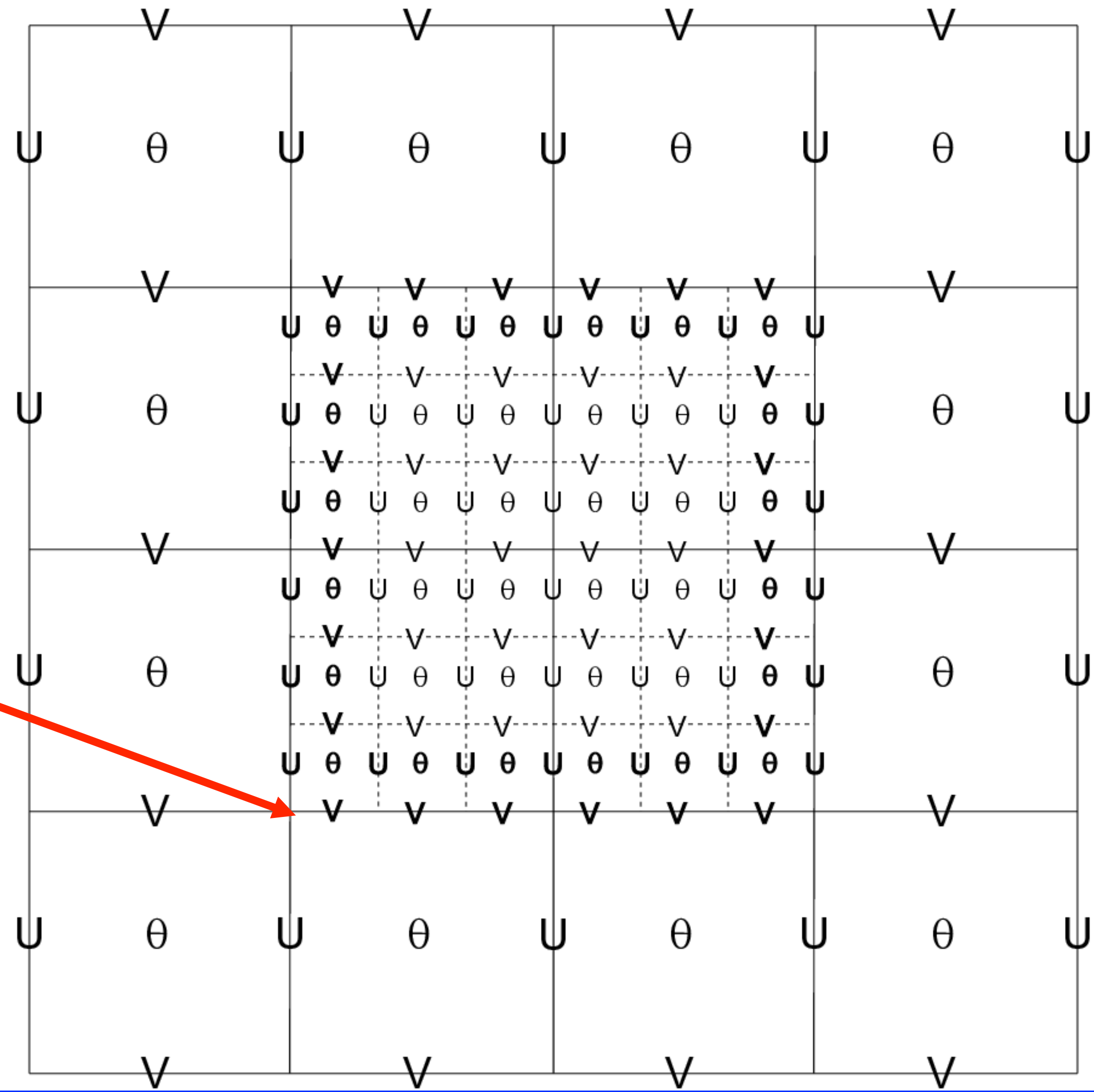
- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting

# ARW Coarse Grid Staggering



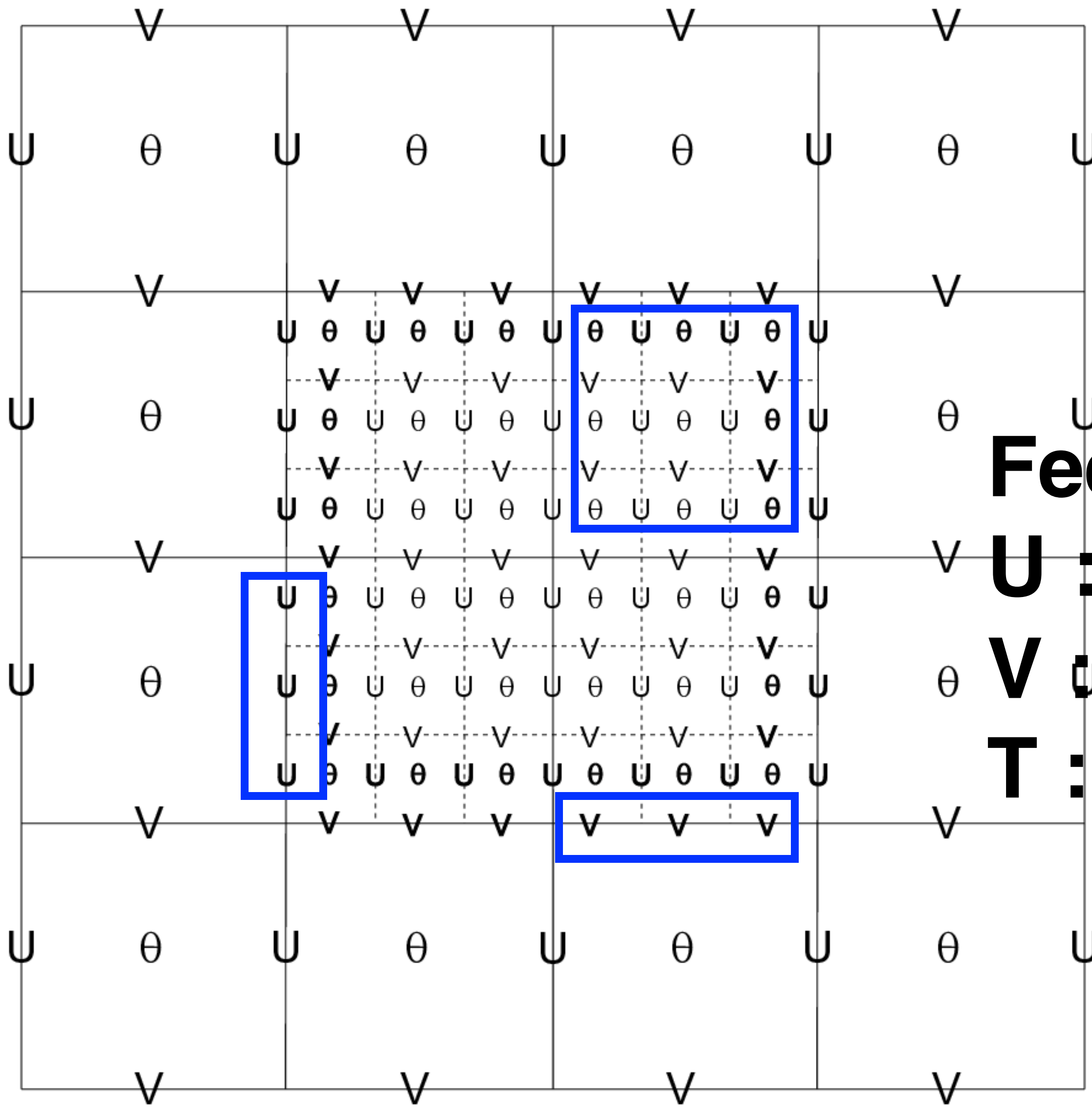
# ARW Coarse Grid Staggering 3:1 Ratio

**Starting Location I = 31**




**CG … 30 31 32 33 34**

# ARW Coarse Grid Staggering 3:1 Ratio



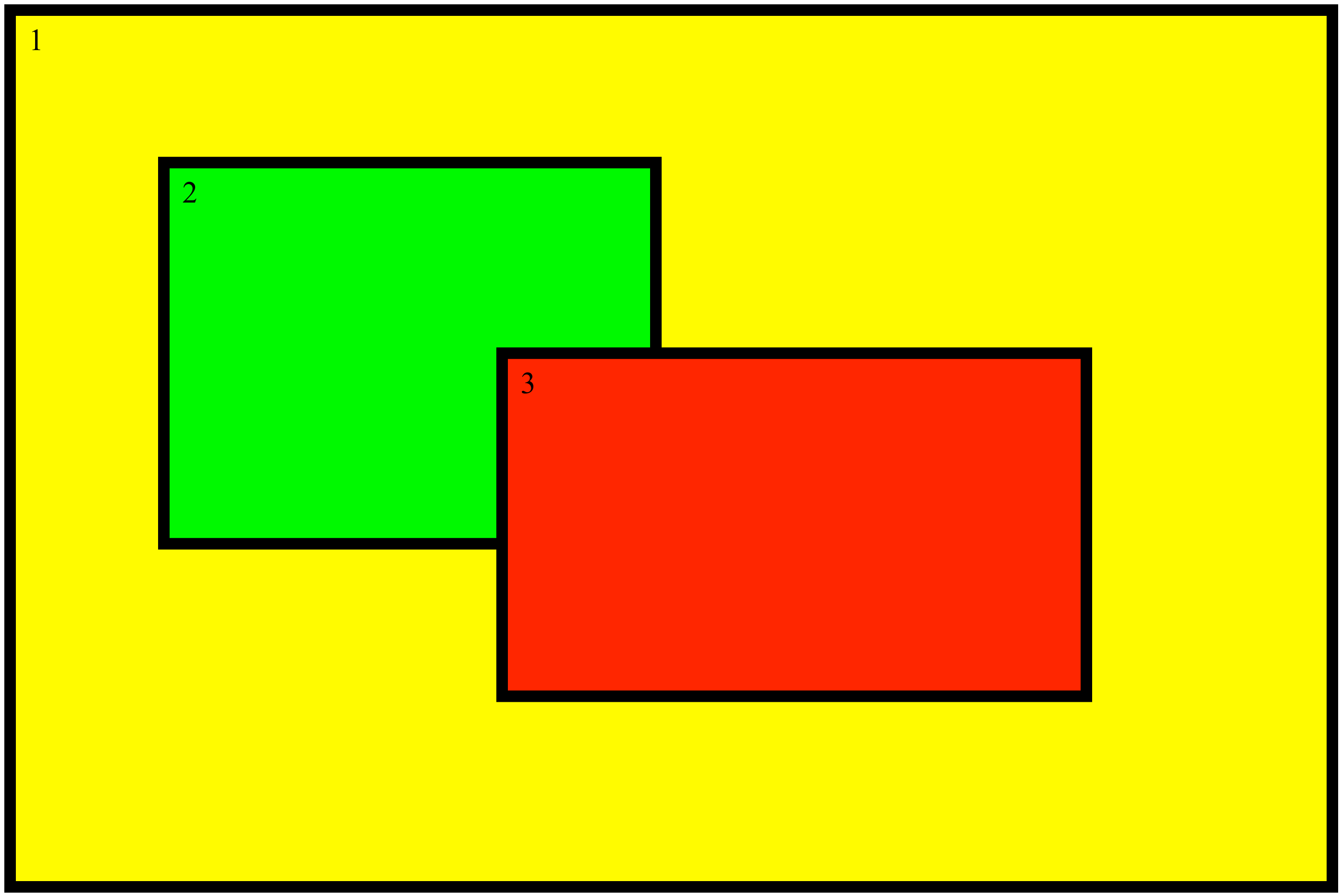
**Feedback:**
**U : column**
**V : row**
**T : cell**

# Some Nesting Hints

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

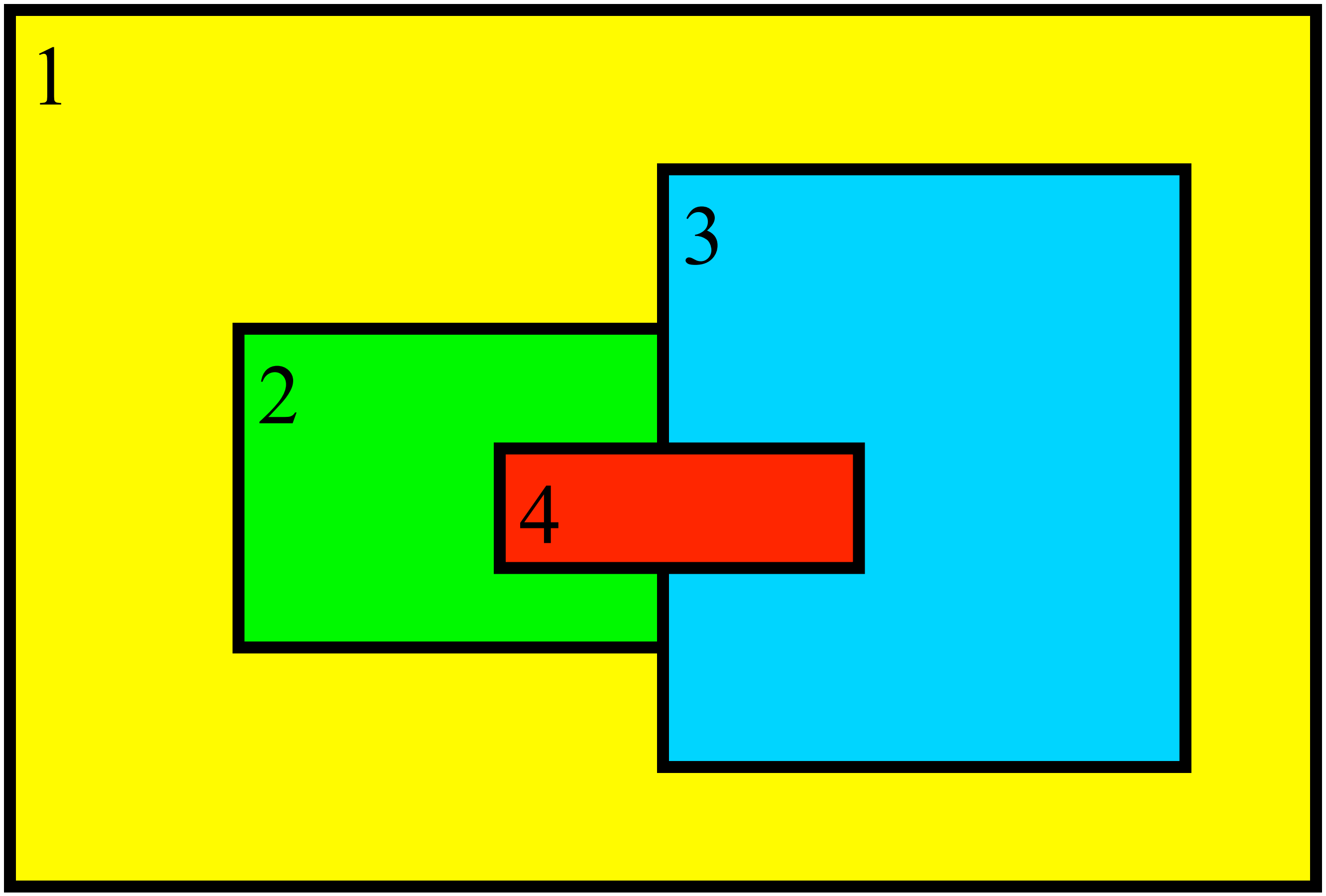
# Not OK for 2-way

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.

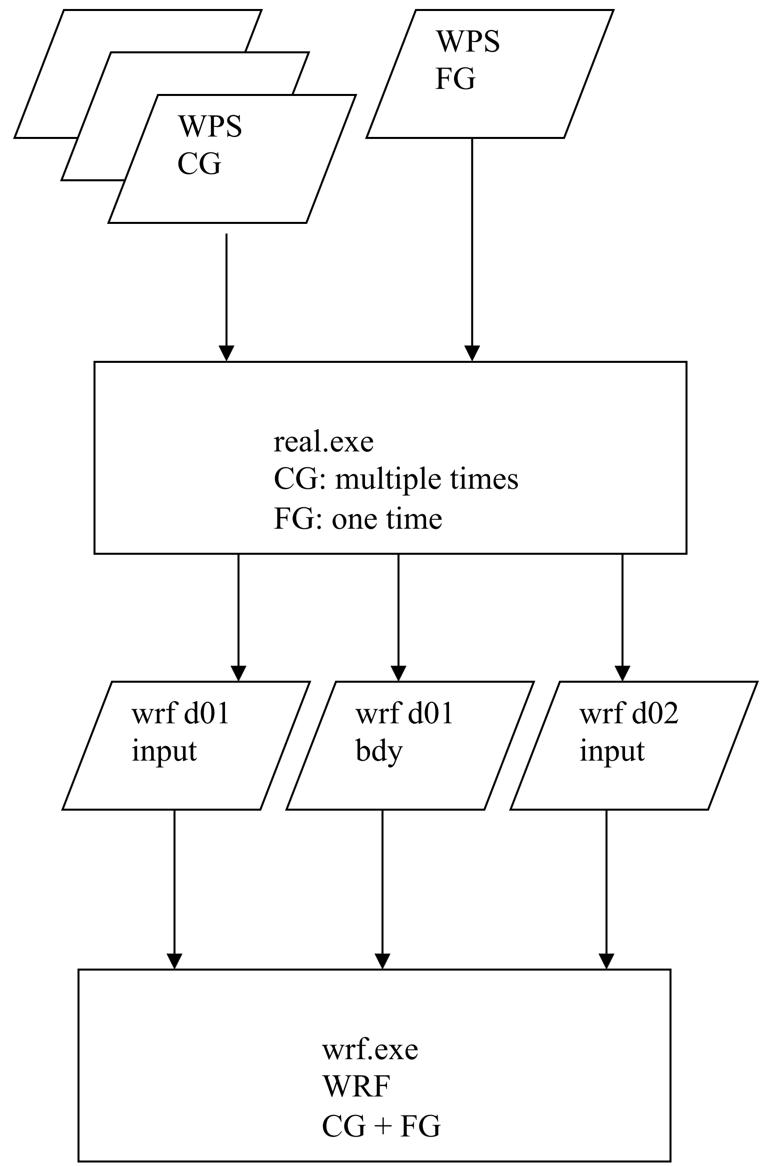
# ARW: 2-Way Nest with 2 Inputs

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

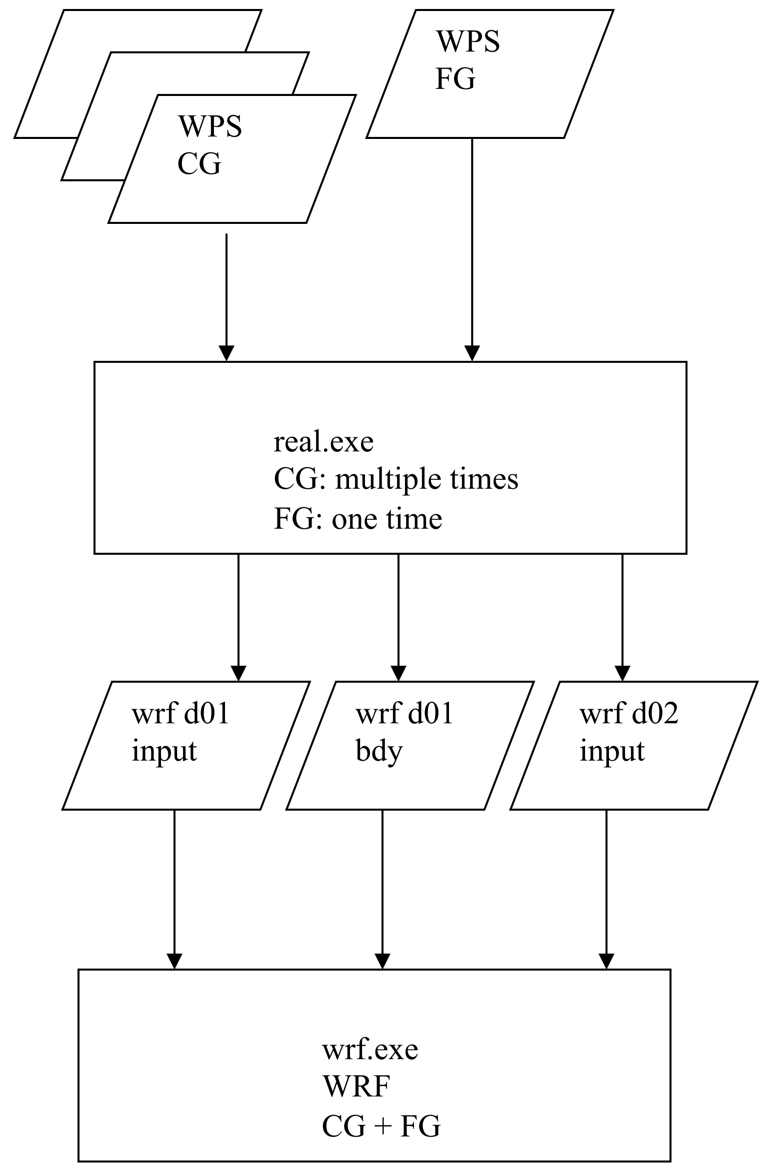
Vertical nesting requires ndown

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

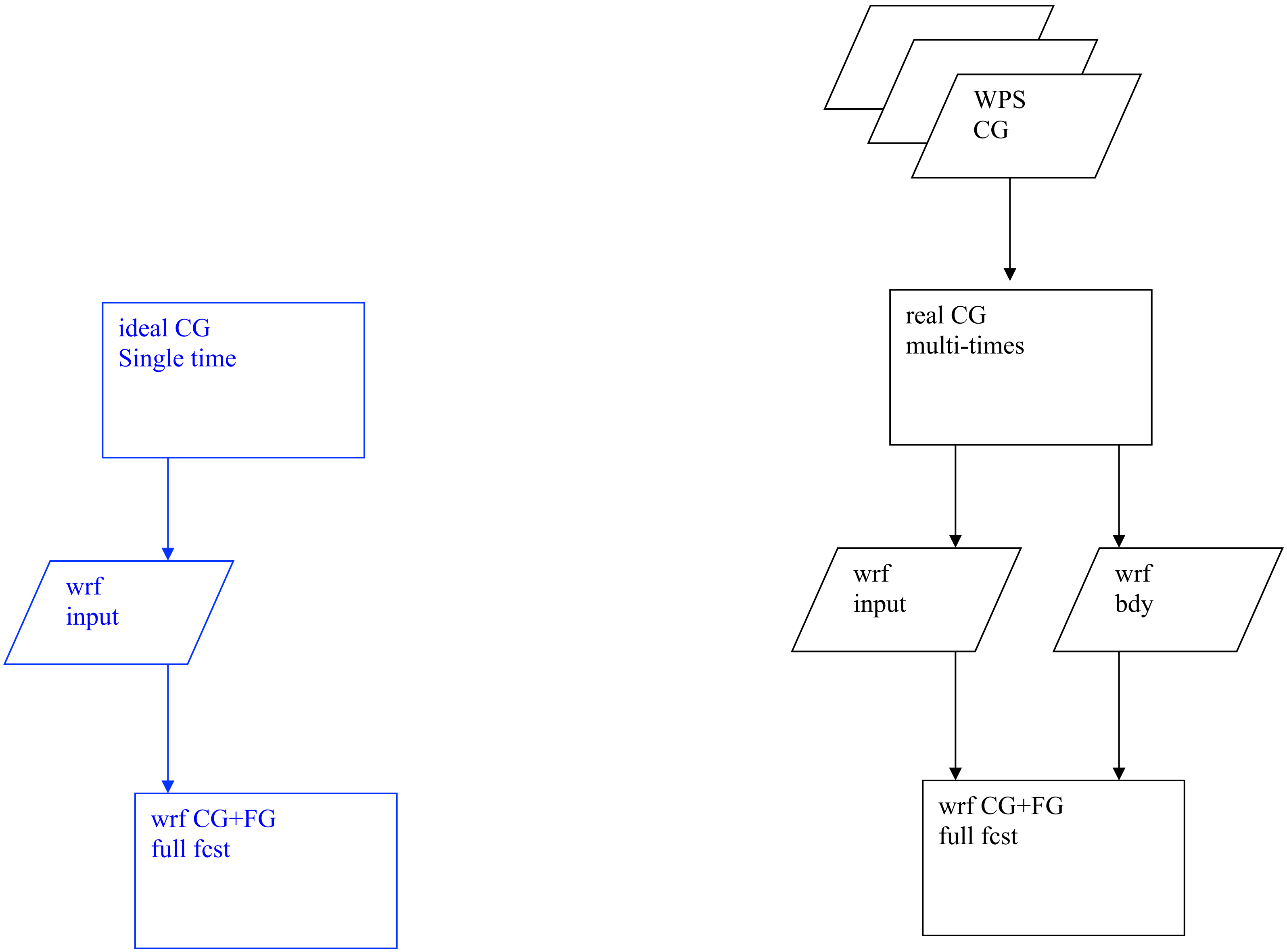
The grid distance ratio is not strictly tied to the time step ratio

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



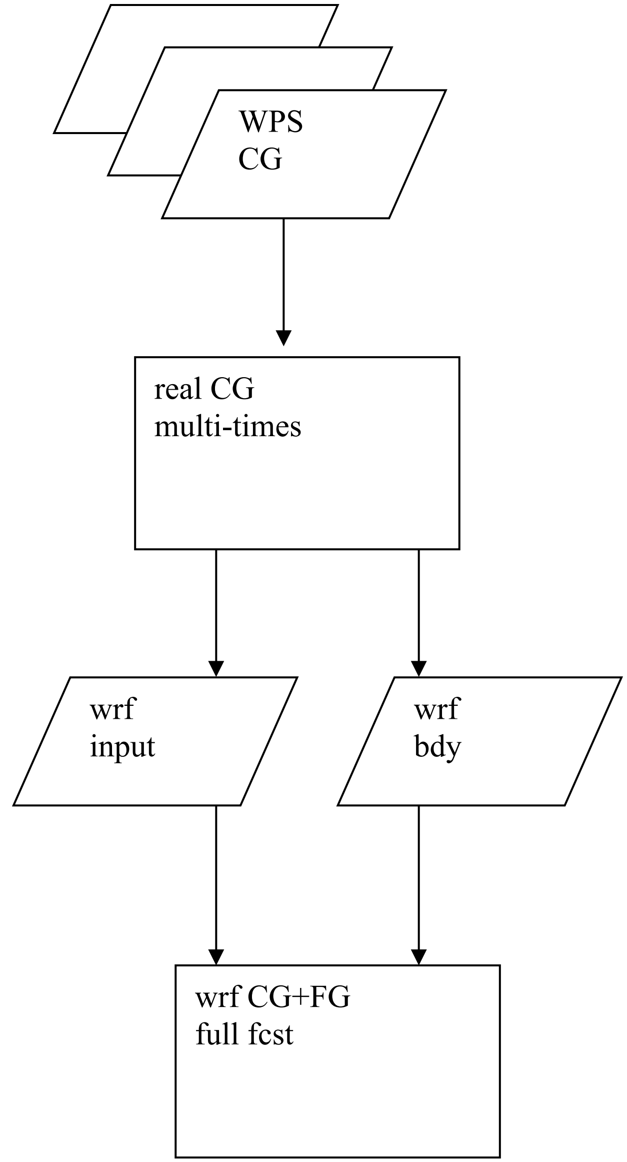
# ARW: 2-Way Nest with 1 Input

A single namelist column entry is tied to each domain

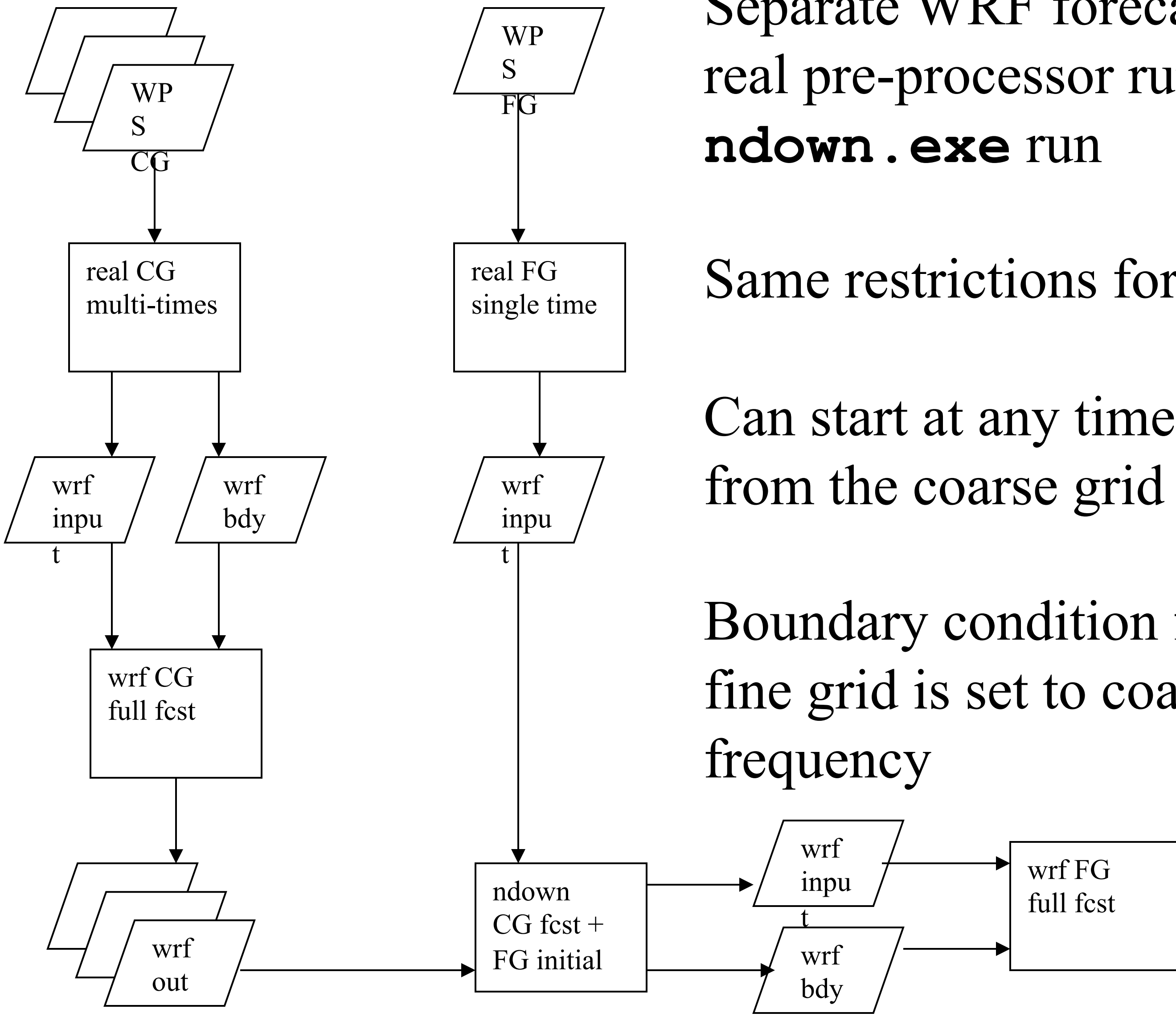
The horizontal interpolation method, feedback, and smoothing are largely controlled through the Registry file

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



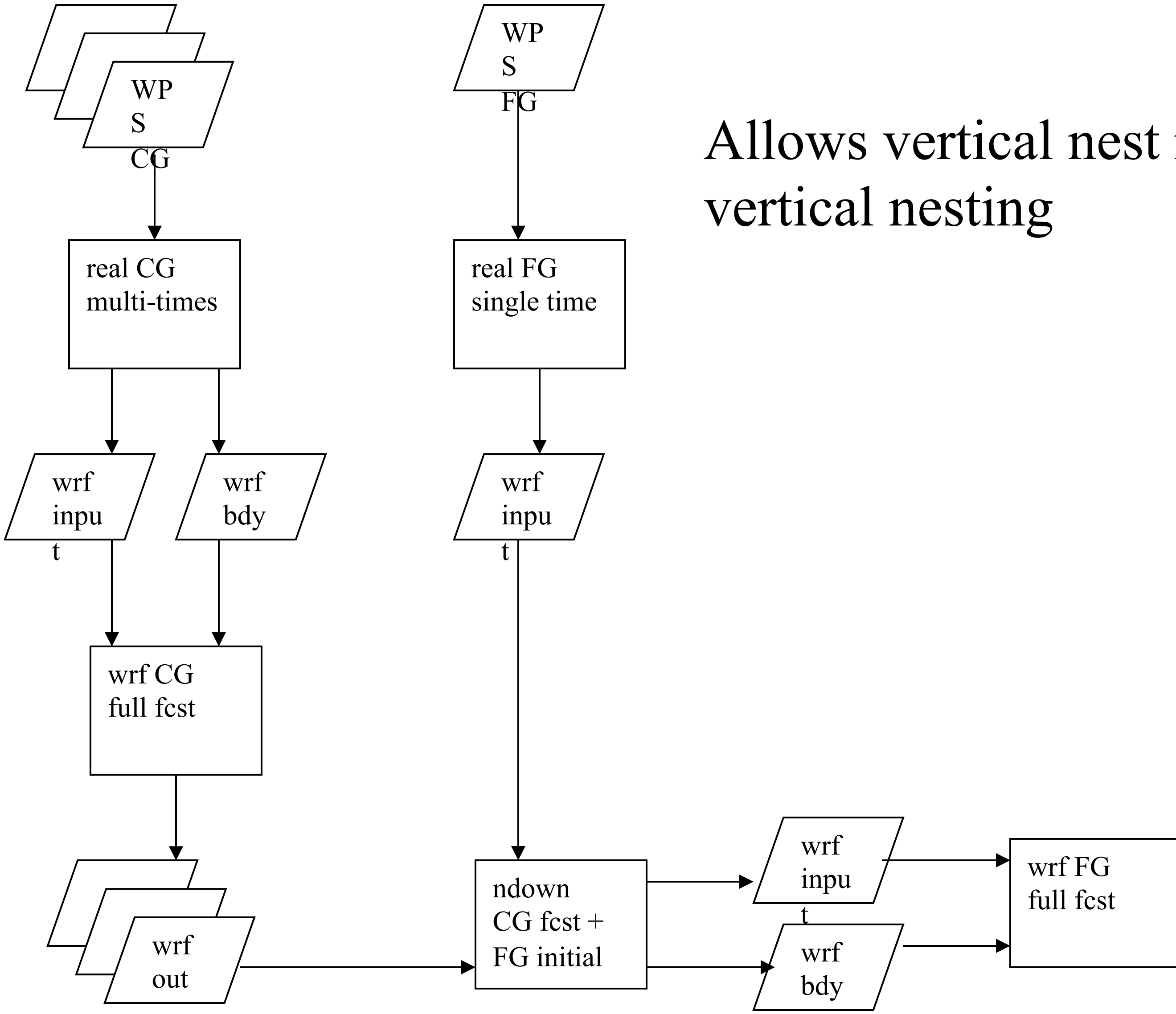

Separate WRF forecast runs, separate real pre-processor runs, intervening **`ndown.exe`** run

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

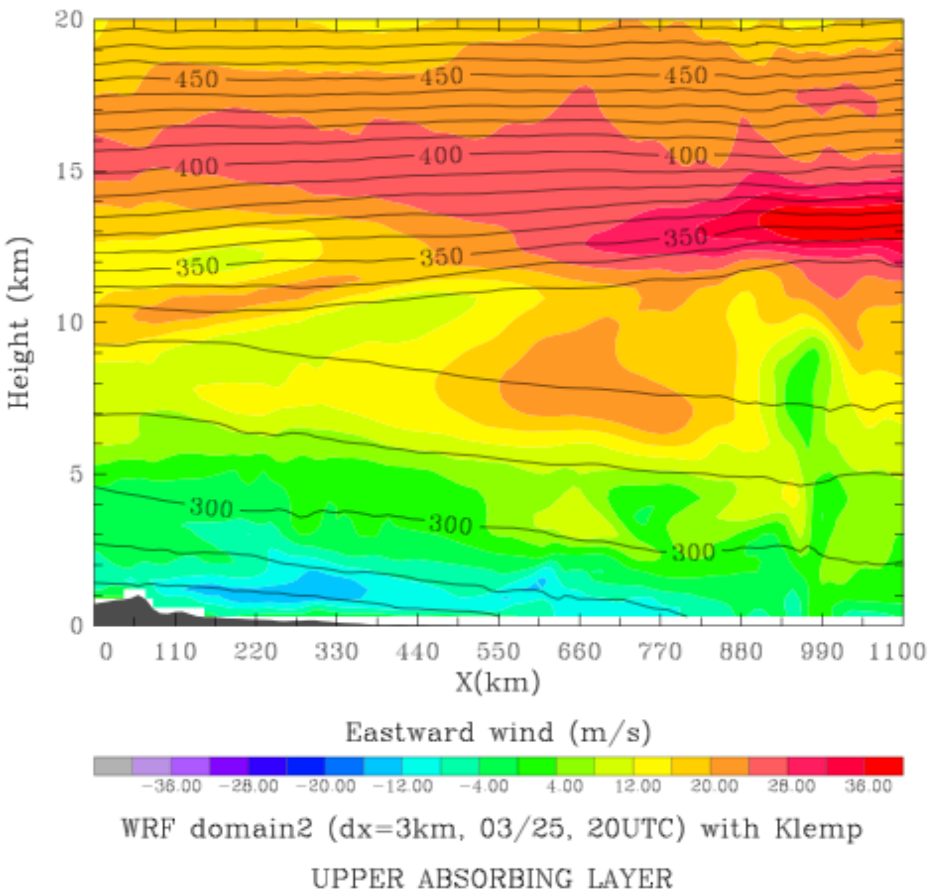
# ndown: 1-Way Nest with 2 Inputs



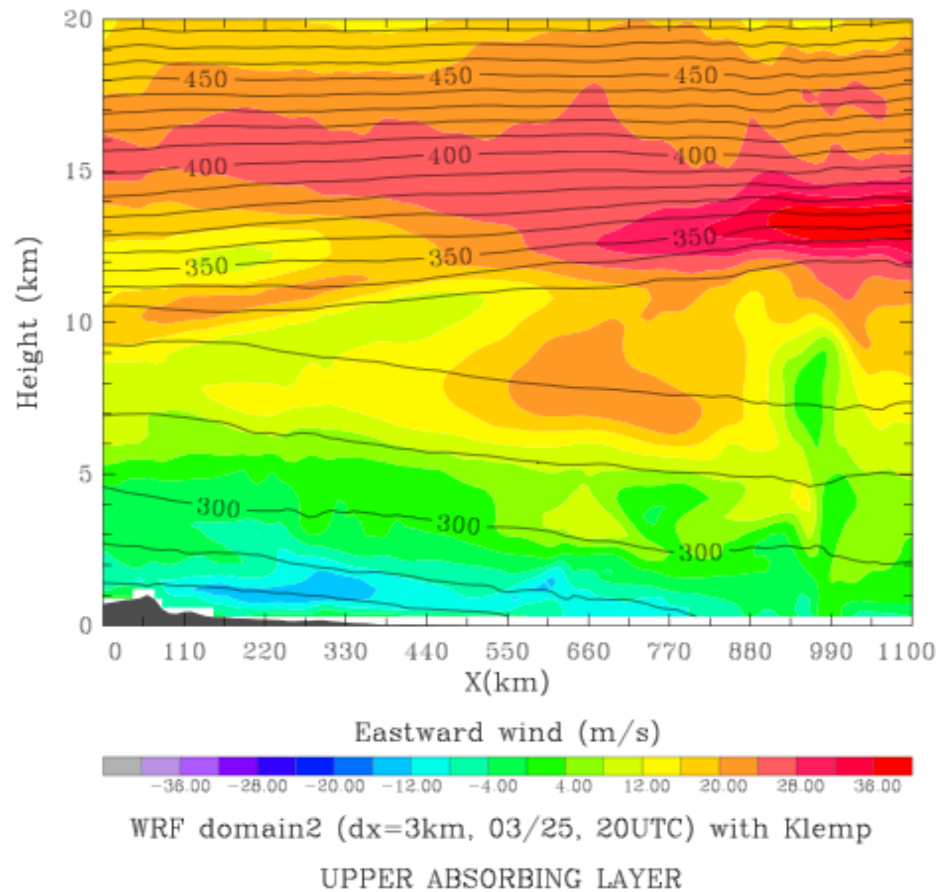

Allows vertical nest refinement ratio: vertical nesting

# West East Cross section
# Shaded: v; Contour: theta
# 6-h Forecast, from Mohamed Moustaoui

## Standard Levels



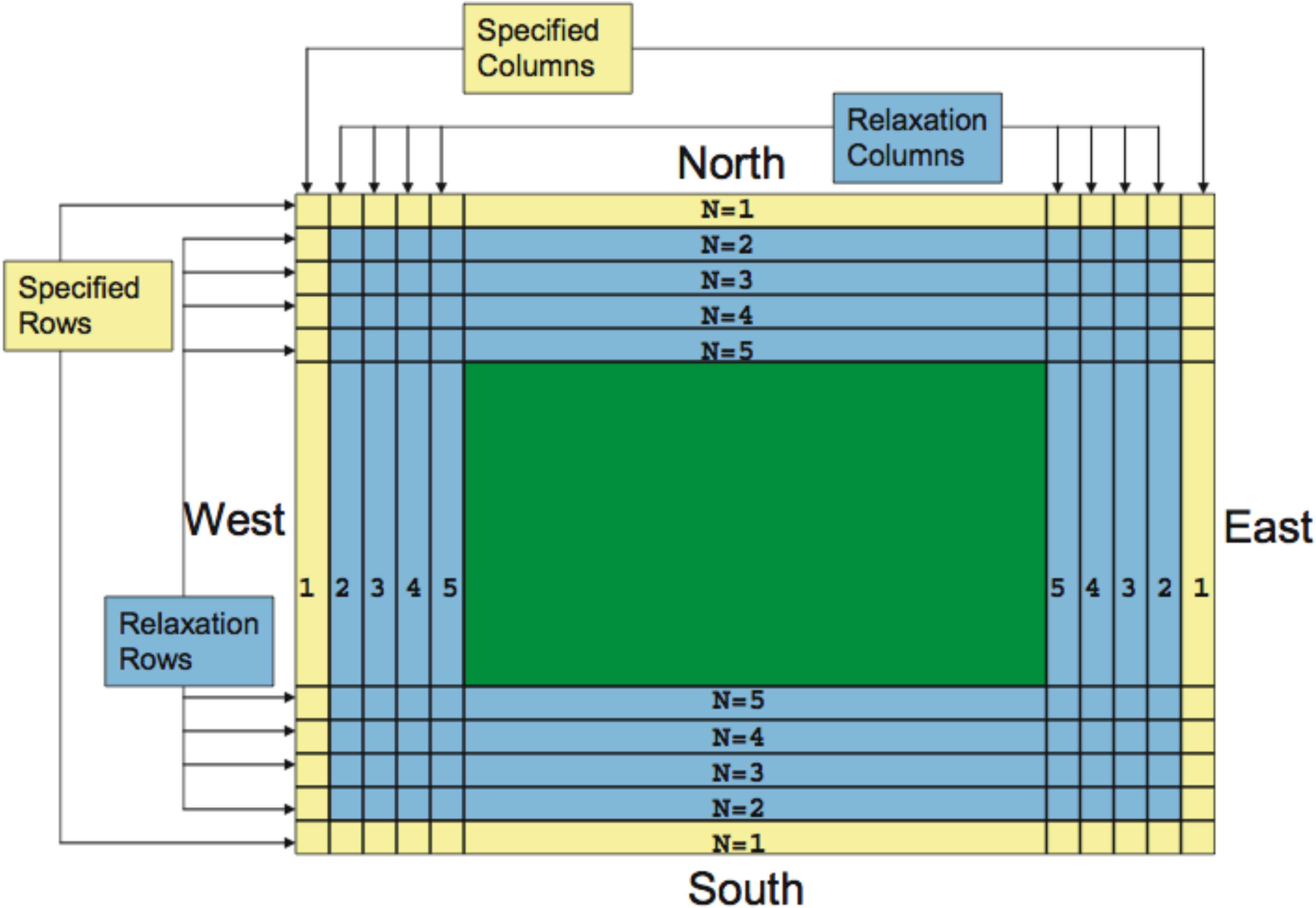

## 3x Refinement



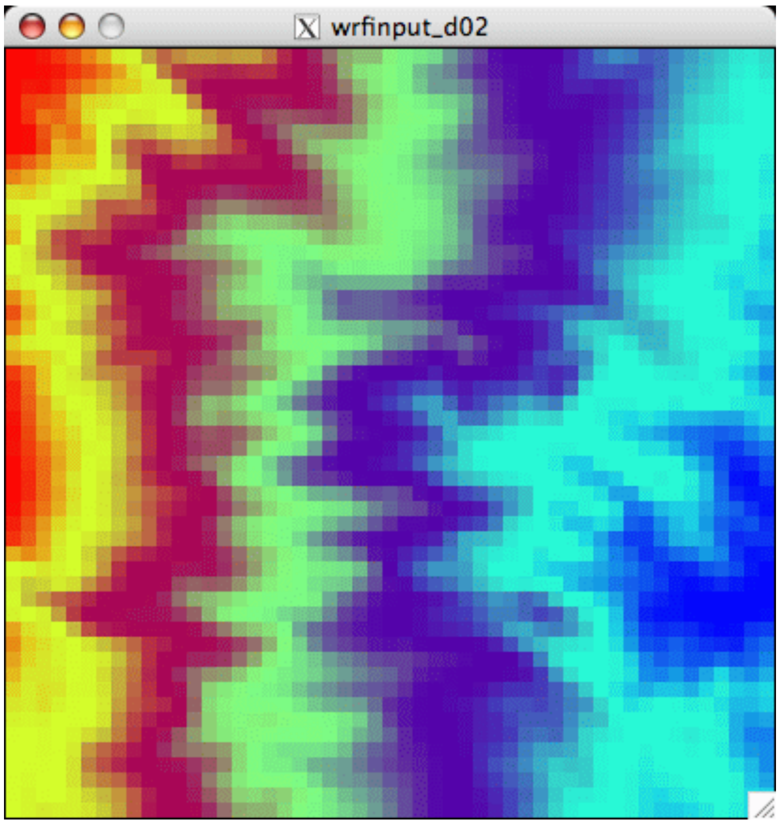
# Some Nesting Hints

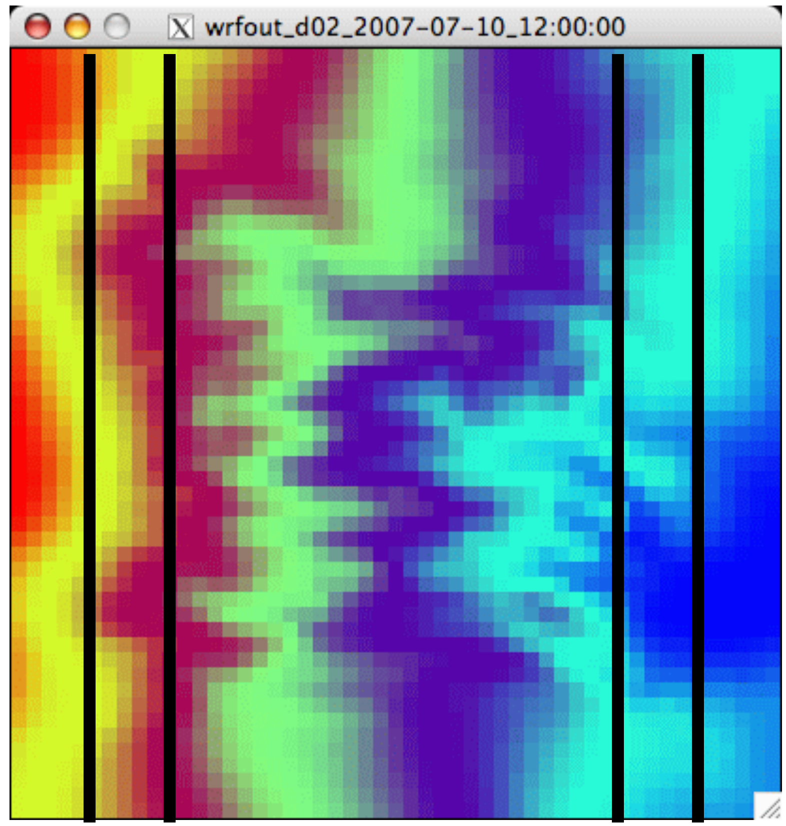
- Allowable domain specifications
- Defining a starting point
- Illegal domain specifications
- 1-way *vs* 2-way nesting
- Nest logic in WRF source code
- Nest information in the Registry

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

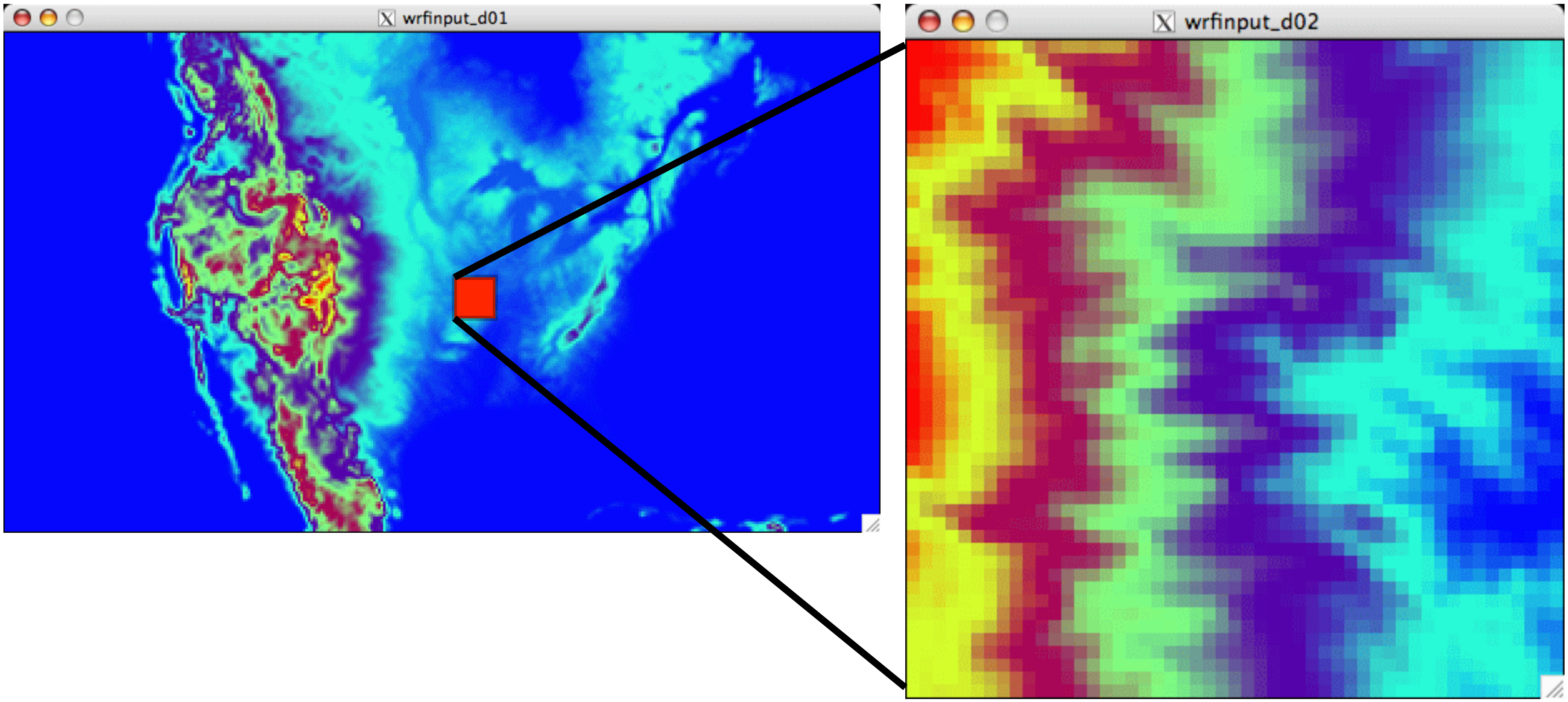


# ARW Lateral Smoothing



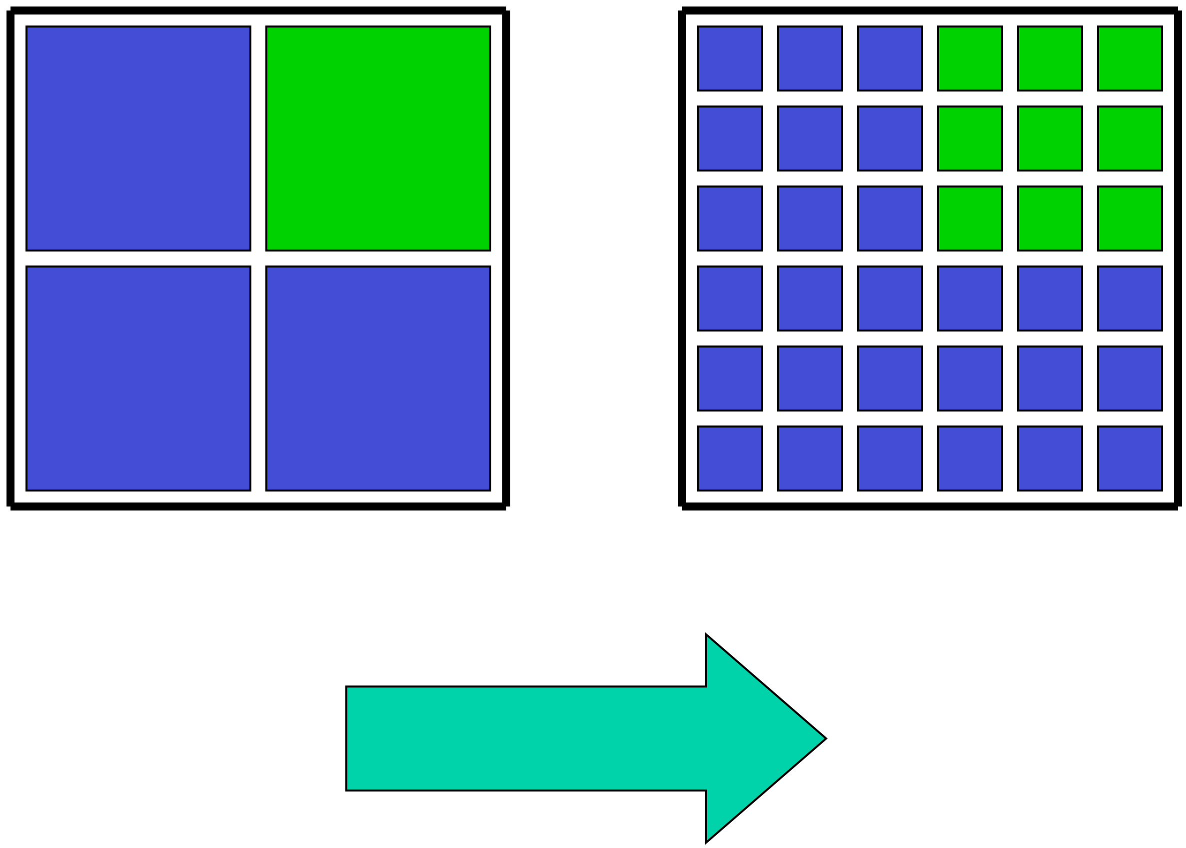



# Intermediate Domains

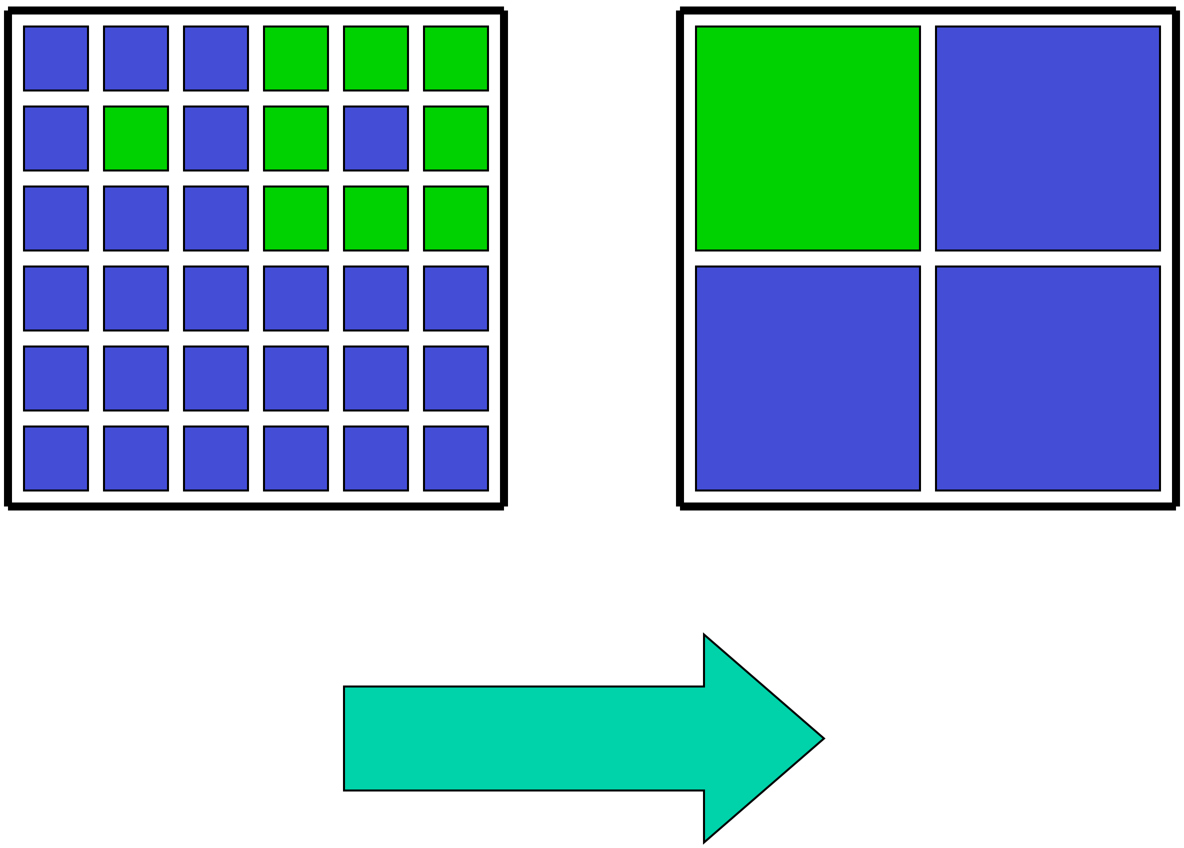



The intermediate domain between a parent and a child is the resolution of the coarse grid over the size of the fine grid. It allows the model to re-decompose the domain among all of the processors.

# 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" ""
```