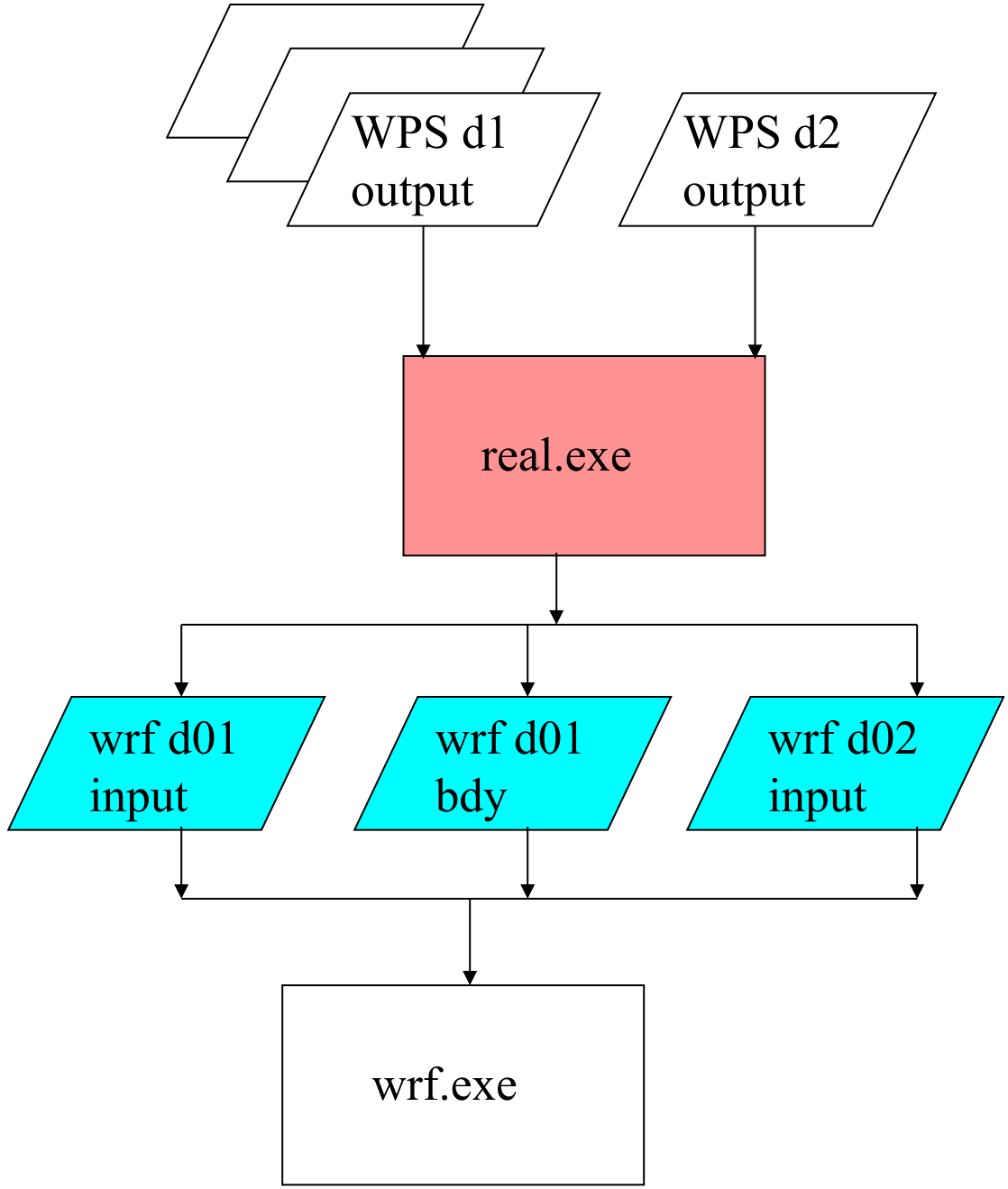
REAL

Description of
General Functions

Dave Gill

gill@ucar.edu



Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

Function

- The 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 transform 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 (maybe usage of nudging capabilities)

Function

- A non-Cartesian ***projected domain***
  - 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

- 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

Function

- **Run-time options**
  - specified in the Fortran namelist file (namelist.input for real and WRF)
- **Compile-time options**
  - Changes inside of the source code
  - Compiler flags
  - CPP ifdefs
  - Modifications to the Registry file

Standard Input Variables

- The metgrid program typically provides meteorological data to the real program.
- Coordinate:
  - The real program is able to input and correctly process any ***strictly monotonic*** vertical coordinate
    - Isobaric: OK
    - Sigma: OK
    - Hybrid: OK

Standard Input Variables

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

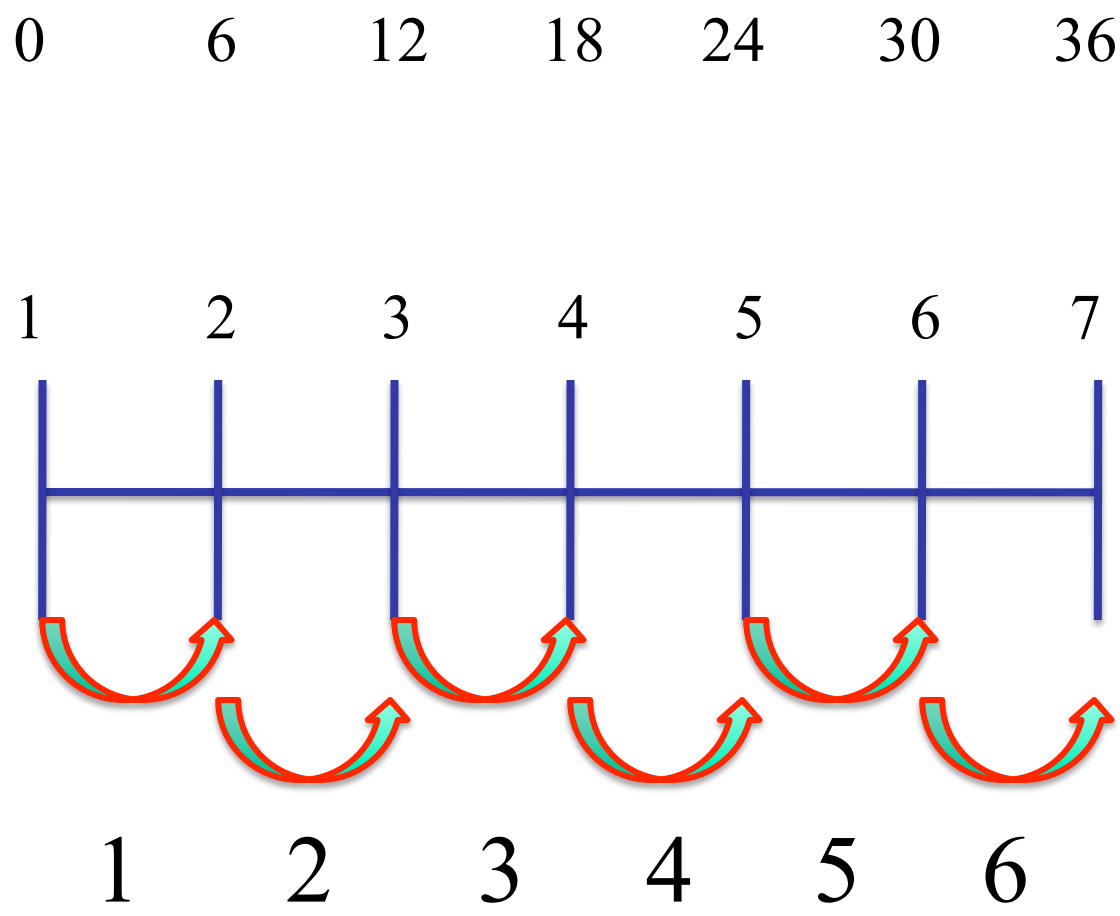
Base State

- 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 – ***inside of the WRF model***
- No base state computations are required ***prior to the real program***

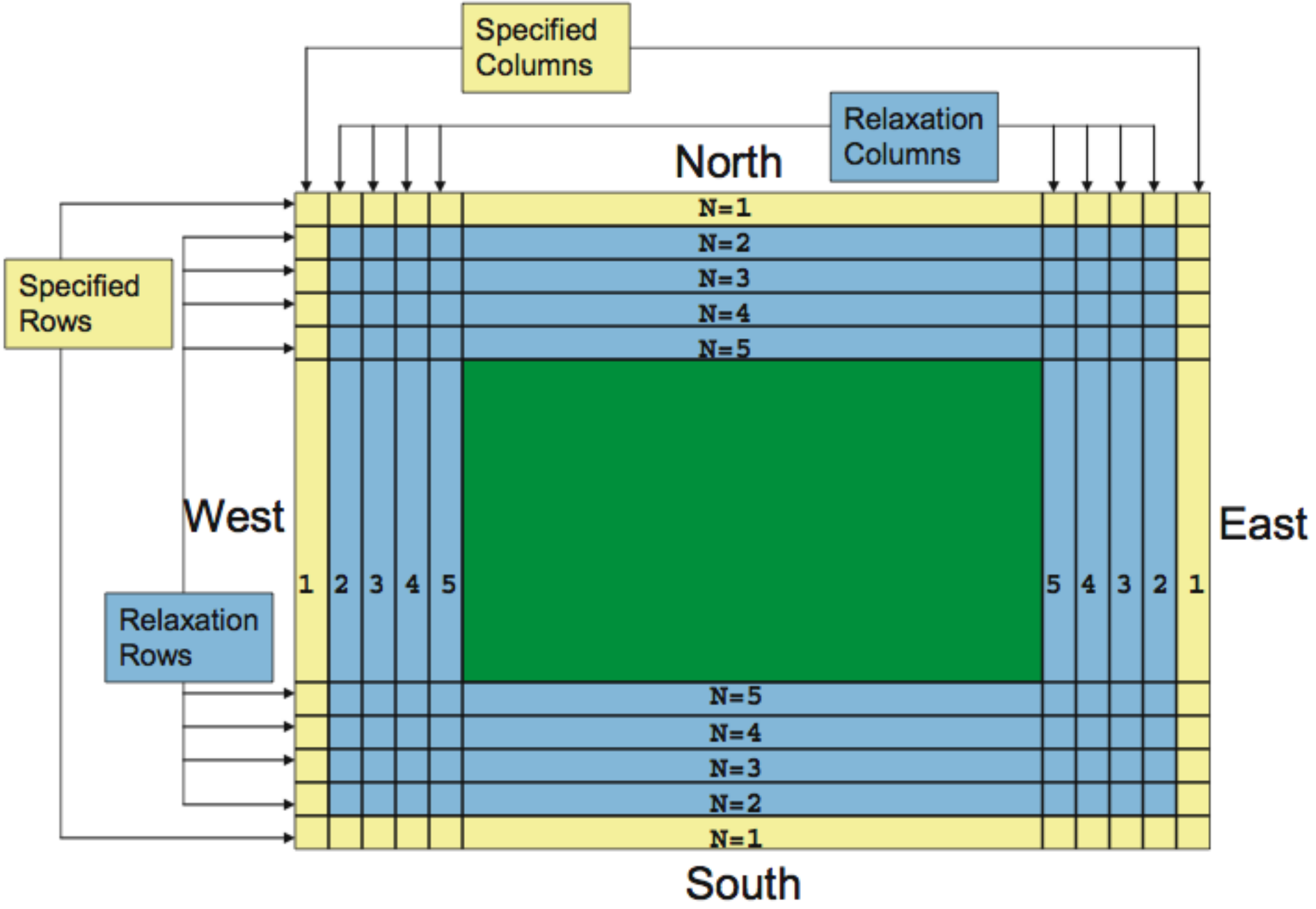
Standard Generated Output

- 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 (look at MPAS for global simulations)
- The ***initial condition*** file contains a ***single time period*** of data
- These files contain data used directly by the WRF model
- The initial condition file may be ingested by the ***WRFDA*** code (referred to as a *cold-start*)
- If ***n*** times were processed with WPS and real, the lateral boundary file contains ***n-1*** time slices

Lateral Boundary Condition Times



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



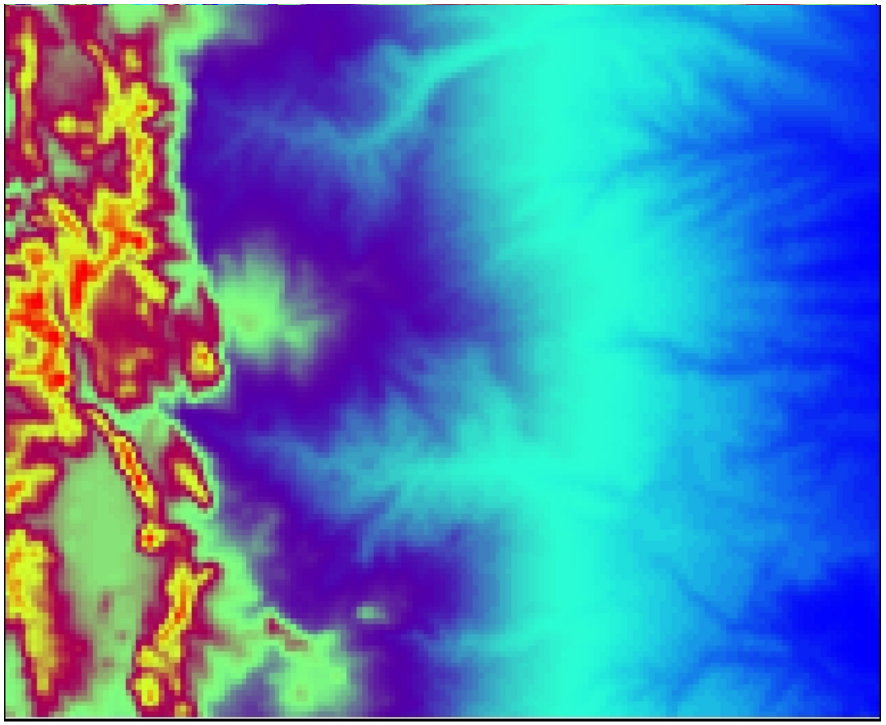
Vertical Interpolation

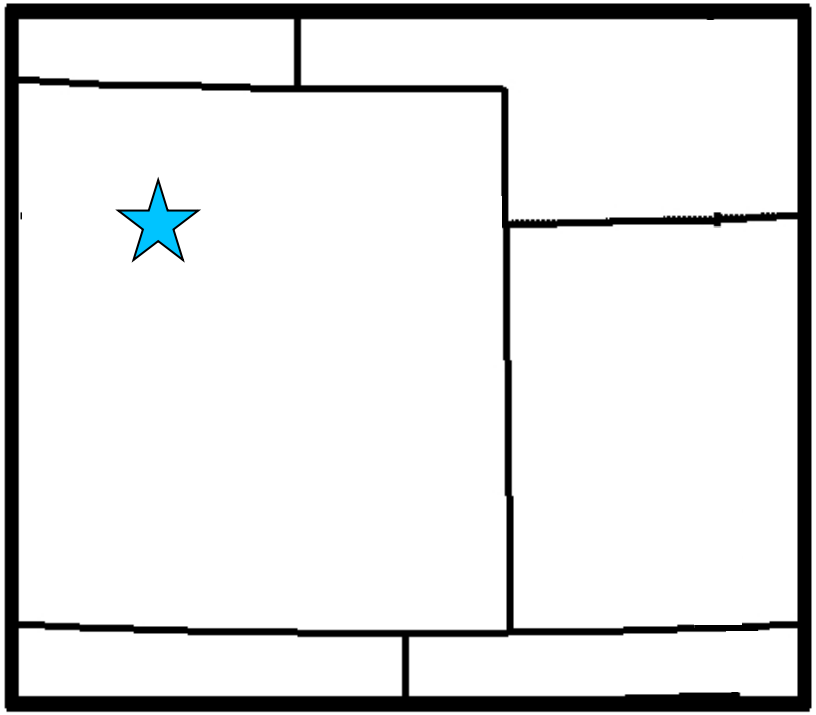
- A number of vertical interpolation options are available to users
- The options can have a significant impact on the initial conditions passed to the model
- More information is contained in the info file ***README.namelist*** in the ***run*** directory
- Options are located in the ***&domains*** namelist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

Make sure input data is vertically ***ordered*** as expected

Input 3-D pressure and T, topo, Z, moisture used to compute total ***surface pressure***

Compute target ***vertical coordinate*** using normalized dry column pressure pressure

The ***eta surfaces*** may be computed or selected

Vertically interpolate input fields in pressure to the **eta** surfaces in dry pressure: default all variables linear in log(pressure)

Vertical Interpolation

- Select reasonable `eta` levels, or let the real program do it for you
- Verify that the ***"thicknesses" are acceptable***, generally about the same value in the free-atmosphere and less than 1000 m
- It is ***SAFEST to NOT initially choose `eta` values***
  - Initially, ***select the number*** of `eta` levels
  - ***Plot profiles*** of the resultant heights
  - ***Adjust the hlevels*** accordingly

- A few namelist options, the terrain elevation, and eta levels completely define the model coordinate for the WRF code

Vertical Interpolation

- The ***eta surfaces*** are computed with a few NML parameters:

```
&domains
e_vert              = 50,    50,   50
p_top_requested     = 1000,

&dynamics
base_temp           = 290.
iso_temp            = 200
```

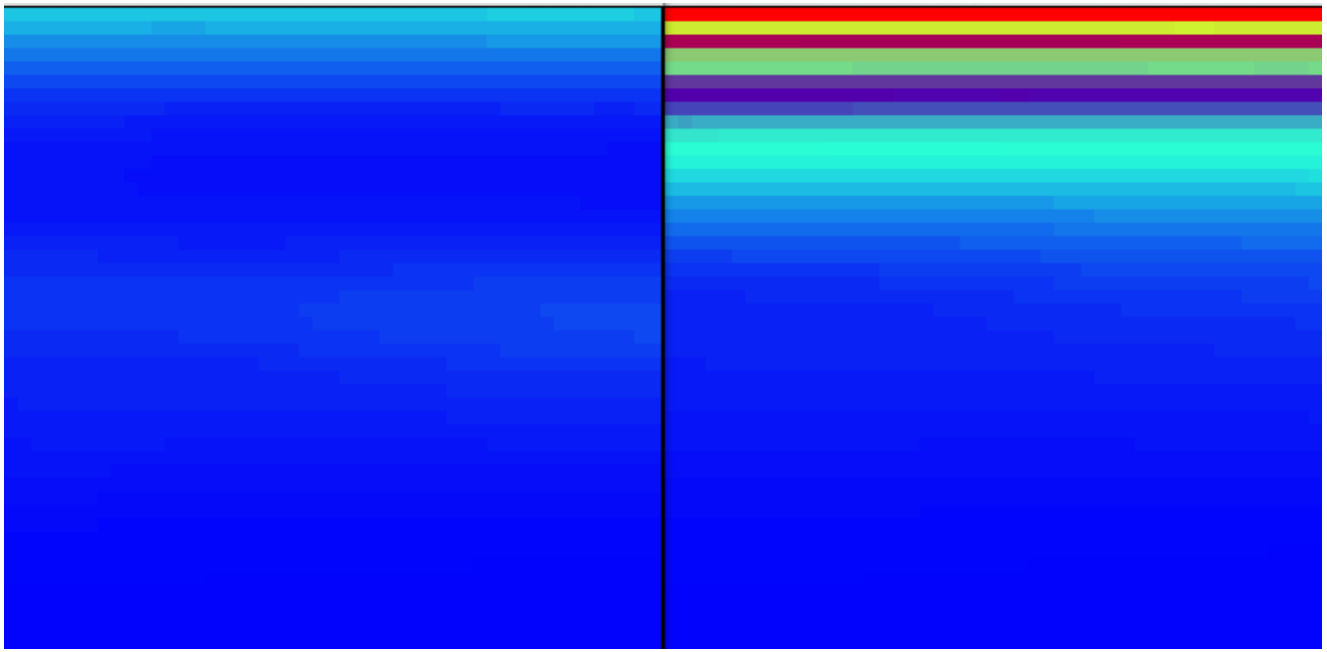
Vertical Interpolation

Vertical cross sections of THICKNESS of each model layer, with 50 vertical levels above the PBL, ptop = 10 hPa.

Uniform layers

Exaggerated Stretching

720-820 m



2000 m

1000 m

800 m

575 m

Physical Parameterization Settings

- The real program and the WRF model are tightly coupled
- Most physical parameterization settings in the namlist.input are IGNORED by real
- EXCEPT
  - ***sf_surface_physics***
  - Land surface model (processes soil temperature and soil moisture)
  - Different schemes in WRF use ***differing numbers of layers***
  - The layers are defined in real from the metgrid output

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

What are the required, optional variables?

From whence do they come?

What are the restrictions on metgrid vertical coordinates?

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

What defines the base state?

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

What are the mandatory files for success?

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

How does the user change the vertical coordinate?

Are there recommendations?

Real program in a nutshell

- Function
- Standard input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation

Why is the surface layer scheme special compared to the other physics options?