WRF System Overview

Cindy Bruyere
Janice Coen
Michael Duda
Jimy Dudhia
Dave Gill
John Michalakes
Bill Skamarock
Wei Wang

What is WRF?

Weather **R**esearch and **F**orecasting Model

- A supported community model
  - Free and shared resource
  - Distributed development
  - Centralized support

What is WRF?

Weather **R**esearch and **F**orecasting Model

- Development is led by
  - NCAR
  - NOAA/ESRL
  - NOAA/NCEP/EMC
- Partnerships: AFWA, FAA, collaborations with universities, government agencies in the US and overseas

What is WRF?

Weather **R**esearch and **F**orecasting Model

- The WRF system refers to all of the associated pre- and post-processors that accompany the dynamical model.

Modeling System Components

- WRF Pre-processing System
  - Real-data interpolation for NWP runs (WPS)
  - Program for adding more observations to analysis (obsgrid)
- WRF Model
  - Programs for real (real.exe) and idealized initializations (ideal.exe)
  - Numerical integration program (wrf.exe)

Modeling System Components

- Graphics and verification tools
- WRFDA
- WRF-Chem
- WRF-Fire – wildland model for surface fires

2956
3960
2221
2970
1980
1486
990
751.6
Date/Time: 0001-01-01_00:00:00
17.01
990
1980
2970
3960

What can WRF be used for?

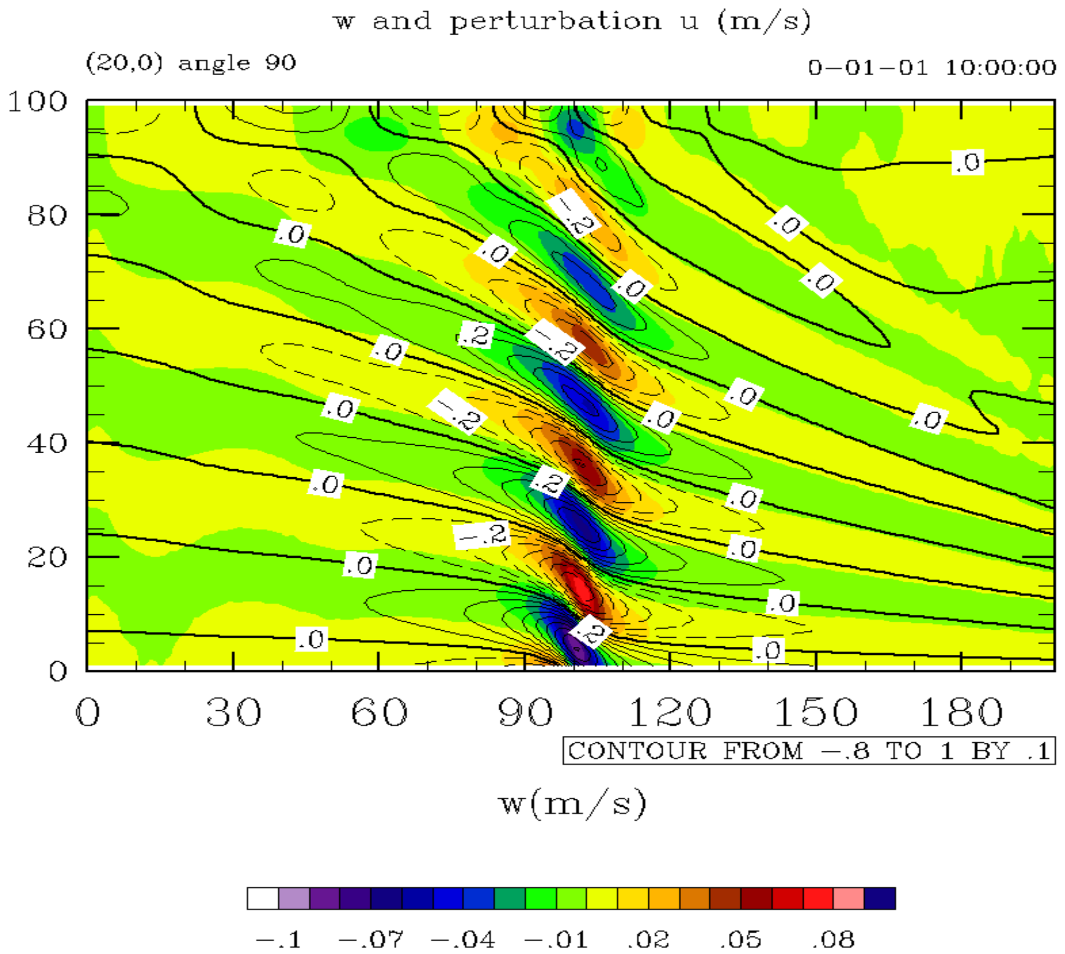
- Research
  - Atmospheric physics/parameterization
  - Case-study
  - Data assimilation
  - Regional climate and seasonal time-scale

What can WRF be used for?

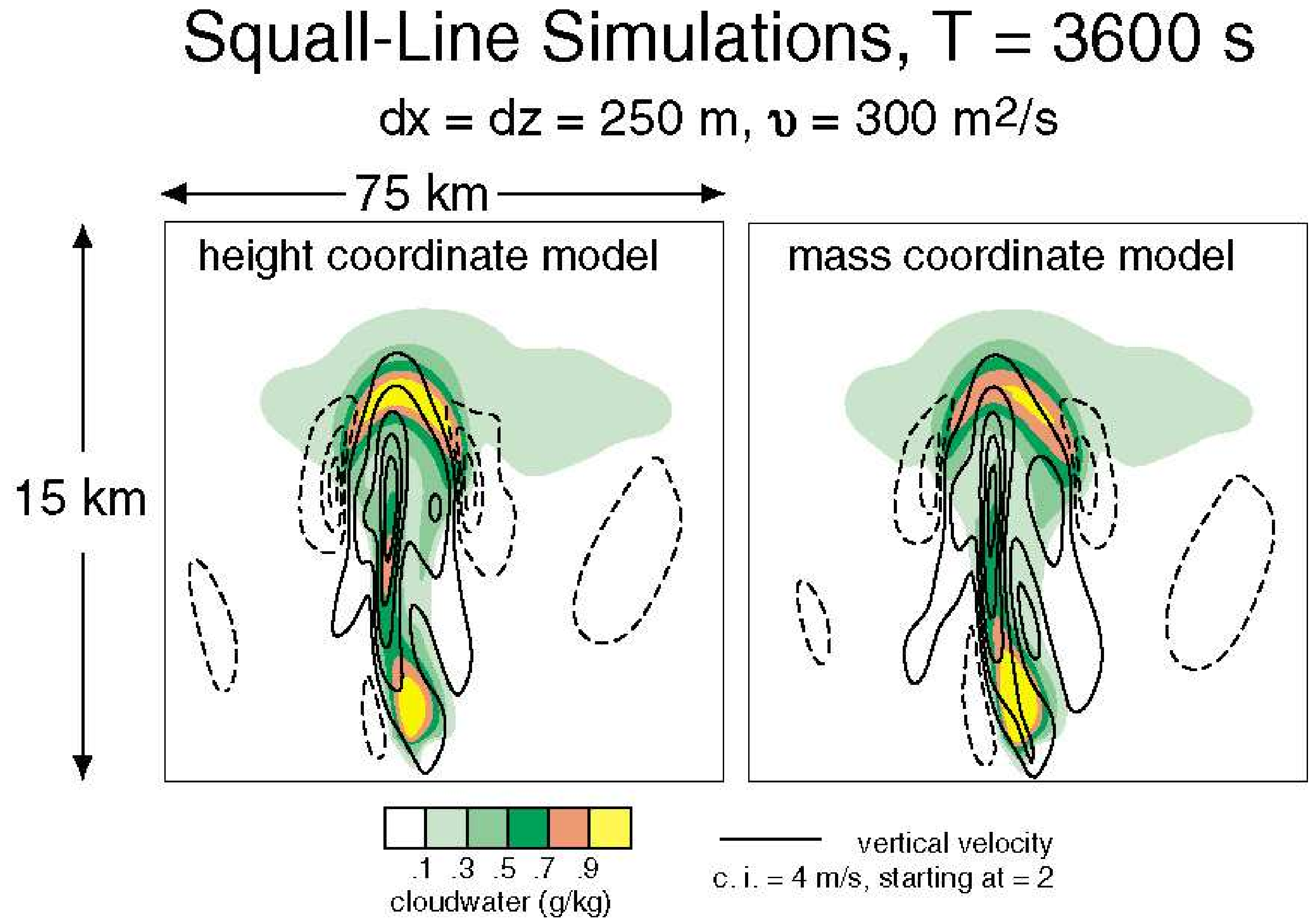
- Teaching dynamics and NWP
- Idealized simulations at many scales: large eddy simulations (meters) to global waves (thousands of km)

What can WRF be used for?

(dx = 2 km, dt = 20 s, T=10 hr)



What can WRF be used for?

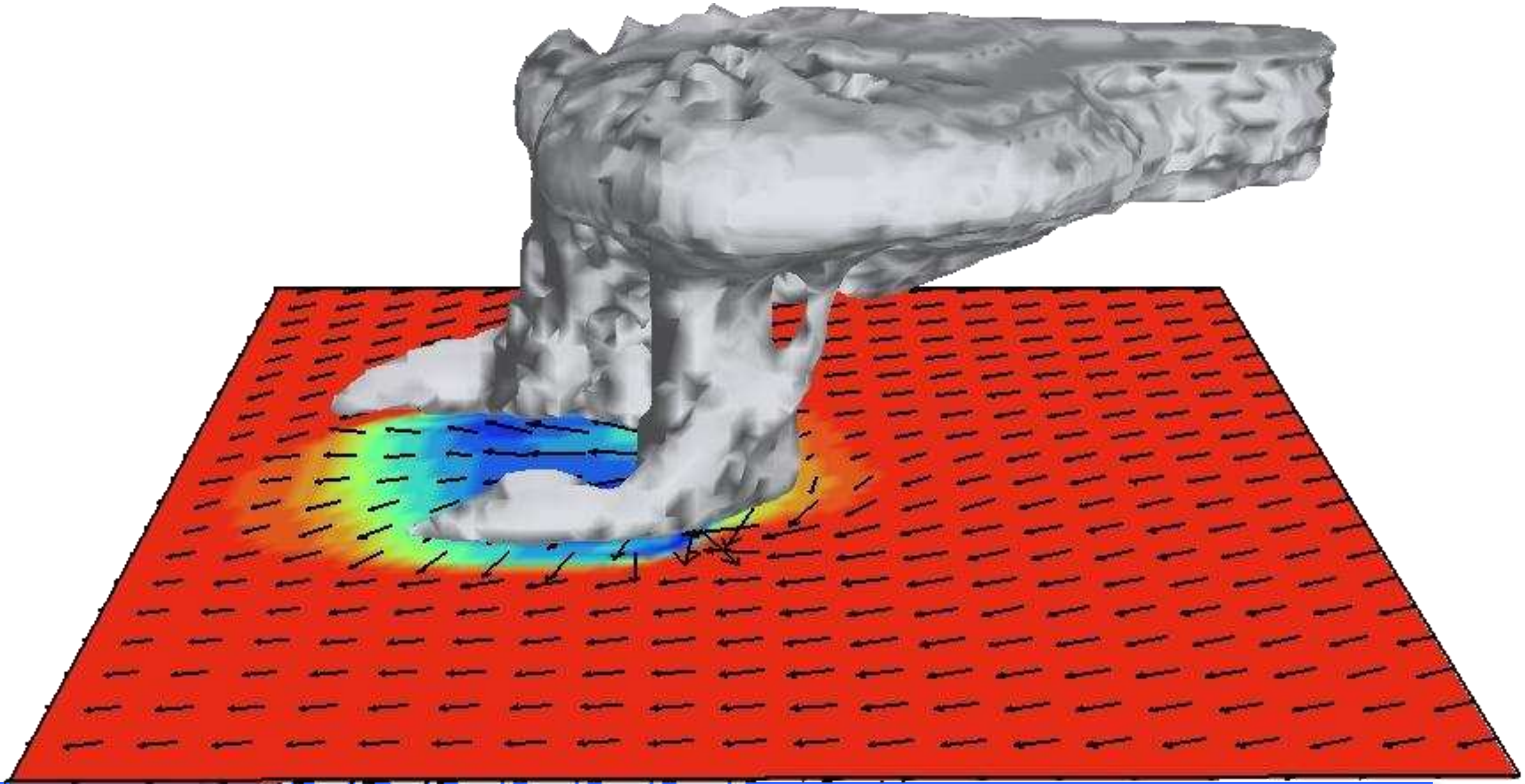


What can WRF be used for?

Height coordinate model

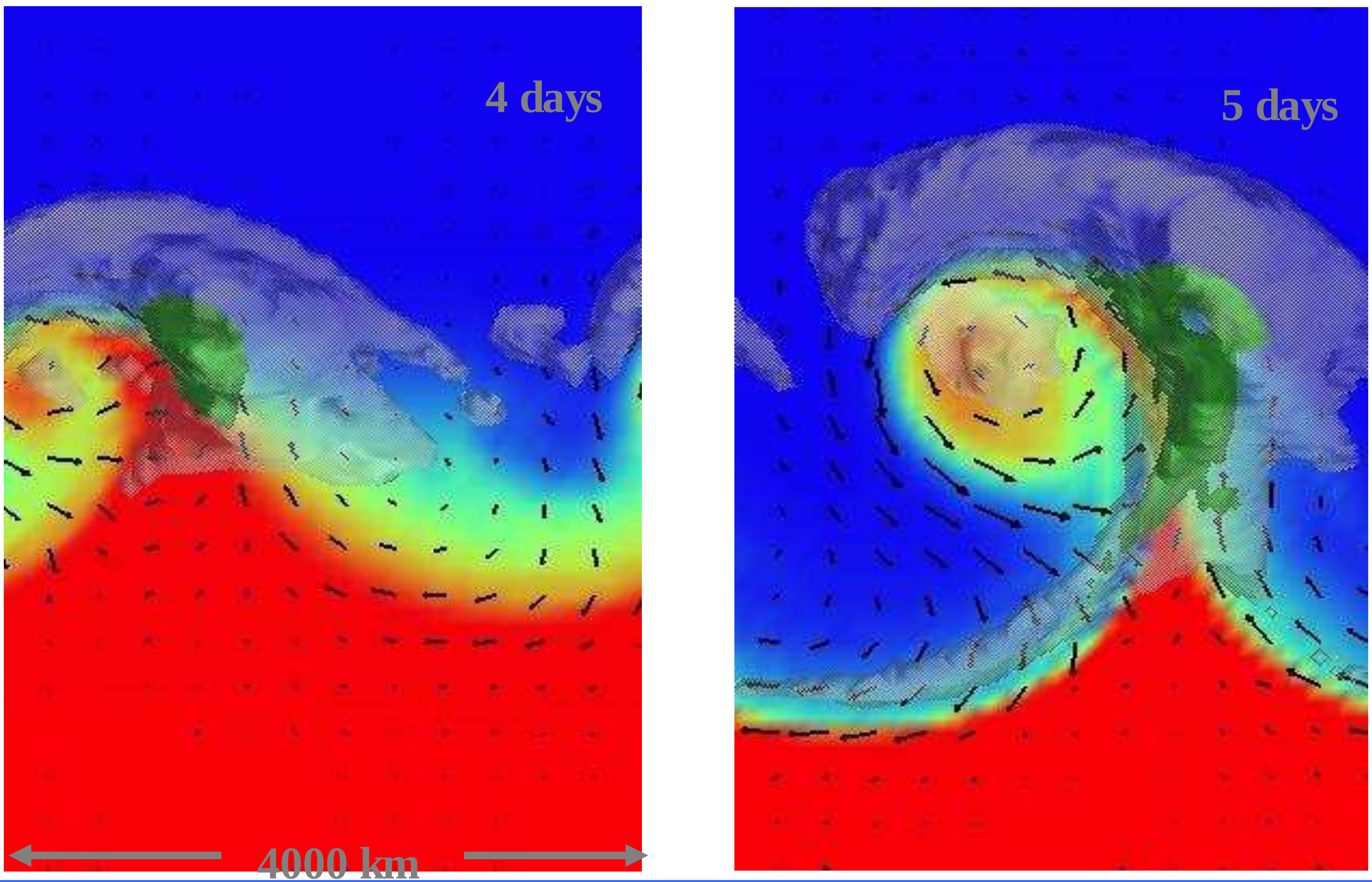
(dx = dy = 2 km, dz = 500 m, dt = 12 s, 160 x 160 x 20 km domain)

Surface temperature, surface winds and cloud field at 2 hours



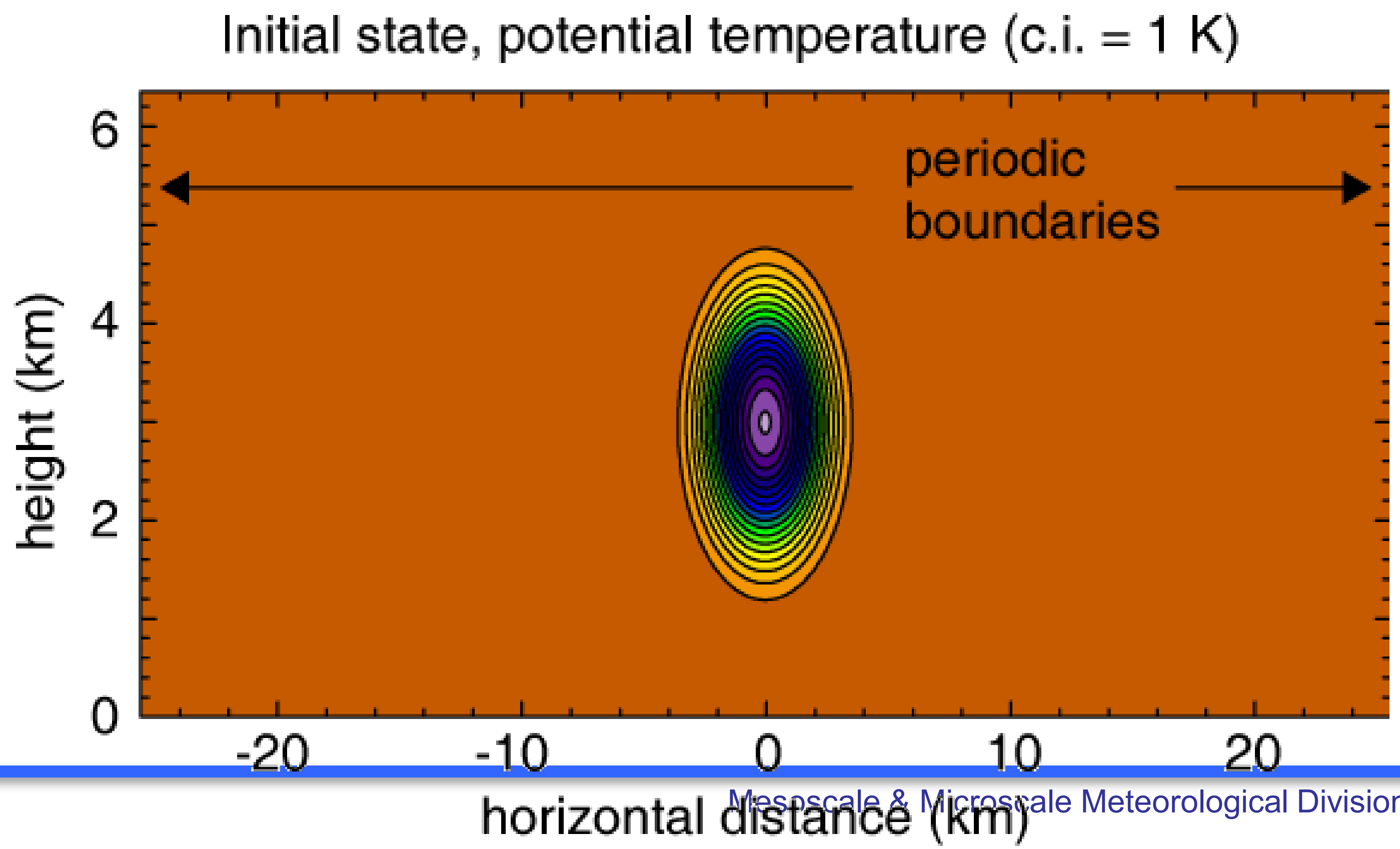
What can WRF be used for?

Height model (dx = 100 km, dz = 250 m, dt = 600 s)
Surface temp, surface winds, cloud and rain water

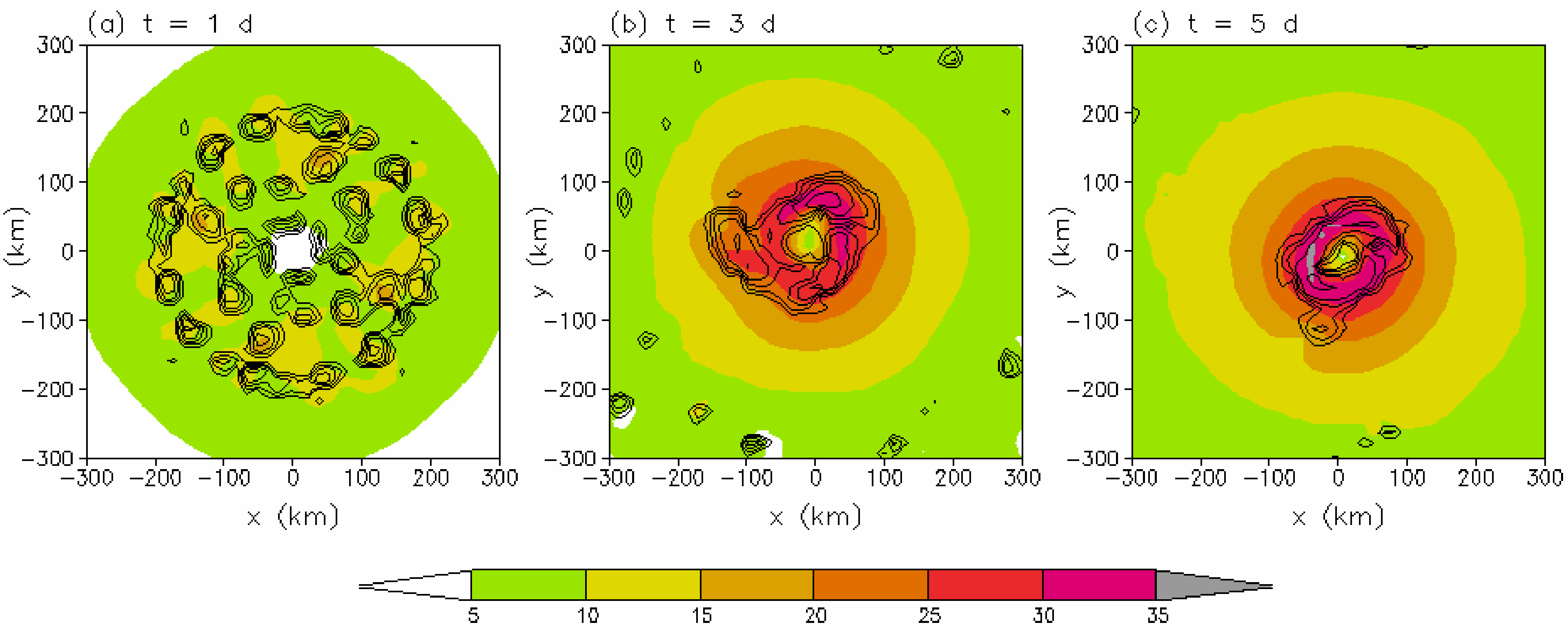


What can WRF be used for?

2D channel (x , z ; 51.2 x 6.4 km)
Initial state: theta = 300 K (neutral) + perturbation (max = 16.2 K)
Eddy viscosity = 75 m**2/s**2 (constant)



What can WRF be used for?



colors = 10-m windspeed (m/s)
contours = reflectivity (every 10 dBZ)

Real-Data Applications

- Need time-independent information for chosen *domain* (simulation grid area)
- GEOGRID program
  - Map projection information
    - 2d gridded latitude, longitude, Coriolis parameter, map-scale factors, etc.
  - Topographic information
    - 2d gridded elevation, vegetation and soil categories, etc.

Real-Data Applications

- Need initial conditions (initial analysis time)
- UNGRIB and METGRID programs
  - 3d fields of horizontal wind, temperature, geopotential height, relative humidity
  - 2d fields of surface or sea-level pressure, surface temperature, relative humidity, horizontal winds
  - Time-sensitive land-surface fields: snow-cover, soil temperature, soil moisture

Real-Data Applications

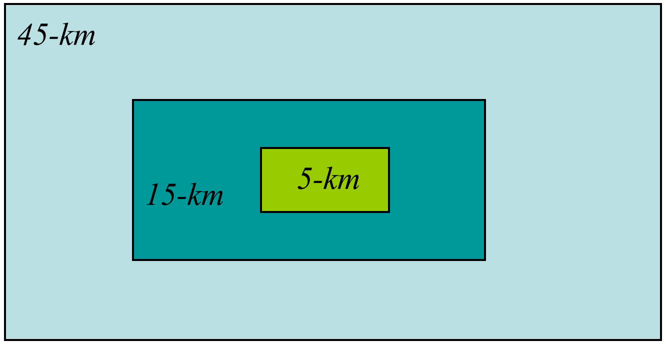
- Regional domains need *specified* lateral boundary conditions at later times (e.g. every 6 hours) through forecast period
  - 3d fields of horizontal wind, temperature, geopotential height, water vapor
  - 2d field of surface pressure
- Long simulations (> 1 week) also need lower boundary condition at later analysis times
  - 2d fields of sea-surface temperature, sea-ice, vegetation fraction, other slowly varying fields

Real-Data Applications

- Lateral Boundary Conditions (linear in time)
  - The *wrfbdy* file contains gridded information at model points in a zone around the domain
  - The boundary fields are linearly time-interpolated from boundary times to the current model time
  - This specifies the outer values, and is used to nudge the next several interior points
- Lower Boundary Condition (step-wise)
  - New SSTs (and other fields) are read in and overwritten at each analysis time from *wrflowinp* file

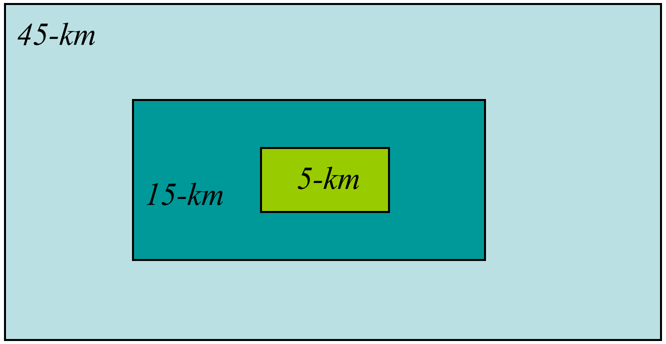
Nesting

- Running multiple domains with increasing resolution in nested areas
- Parent has *specified* boundary conditions from *wrfbdy* file
- *Nested* boundary conditions come from parent



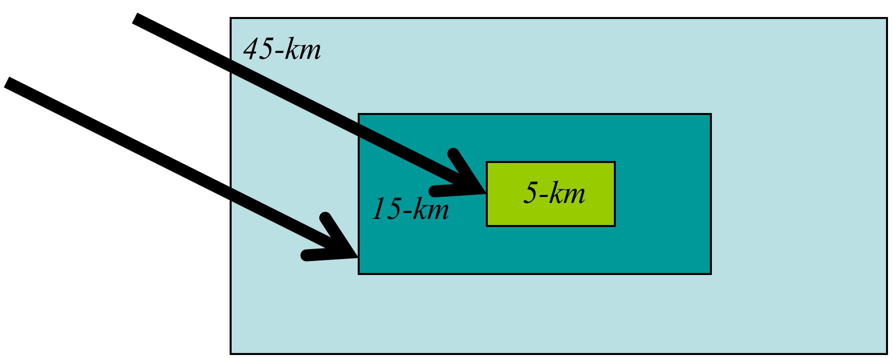
Nesting

- Can select either 1-way to 2-way nesting



Nesting

- Can select either 1-way to 2-way nesting
- 1-way CG impacts FG
- 2-way CG impacts FG, FG impacts CG



Nesting (Two-Way)

- Lateral boundary condition is provided by parent domain at every parent step
- Feedback: Interior of nest overwrites overlapped parent area

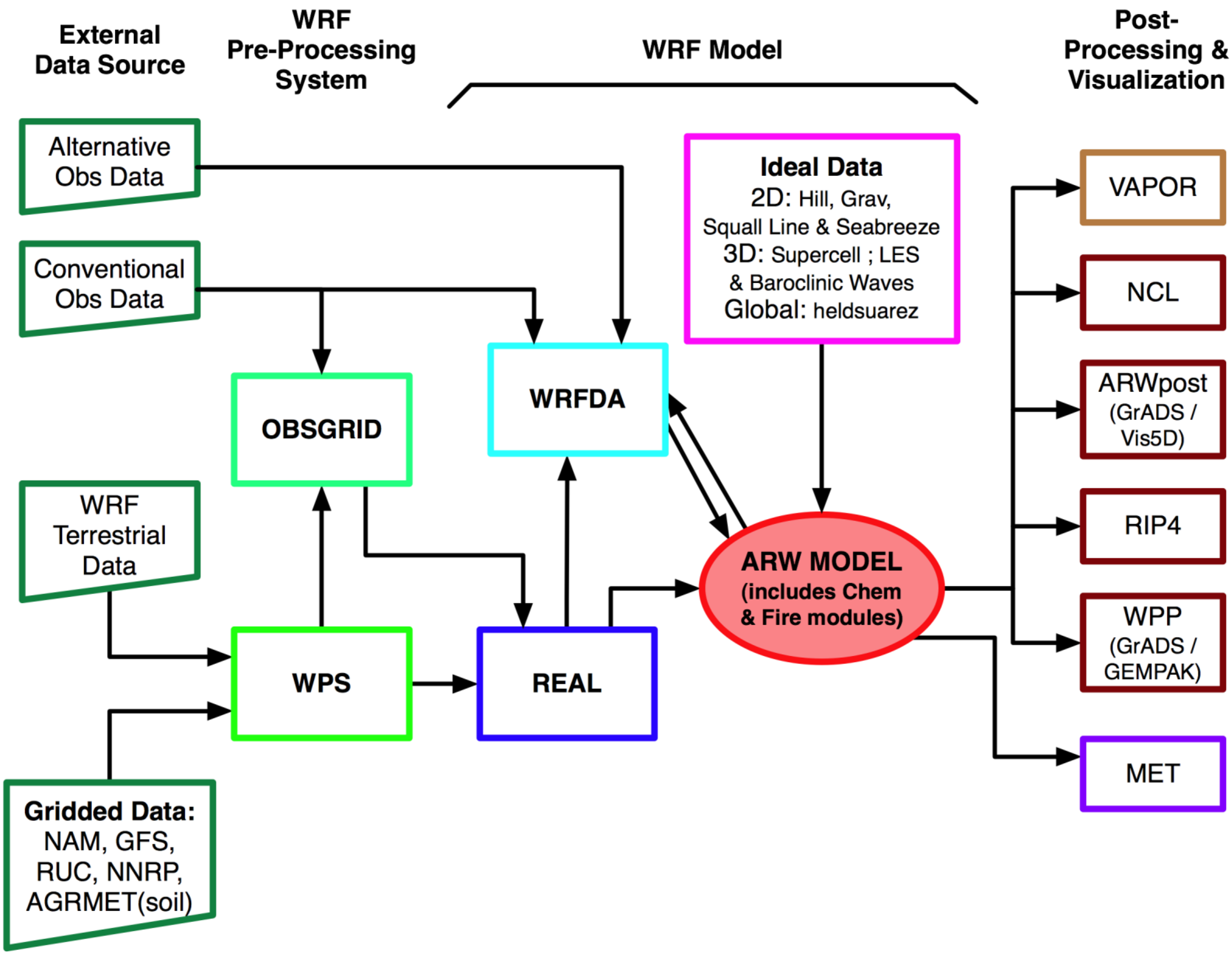
Nesting (Two-Way)

- Sequence
  - Parent domain runs a time-step to t+dt
  - Nest boundaries from beginning and end of time-step interpolated
  - Nest runs multiple steps using time-interpolated parent info at nest boundaries
  - After nest reaches t+dt, feedback overwrites parent in overlapped region

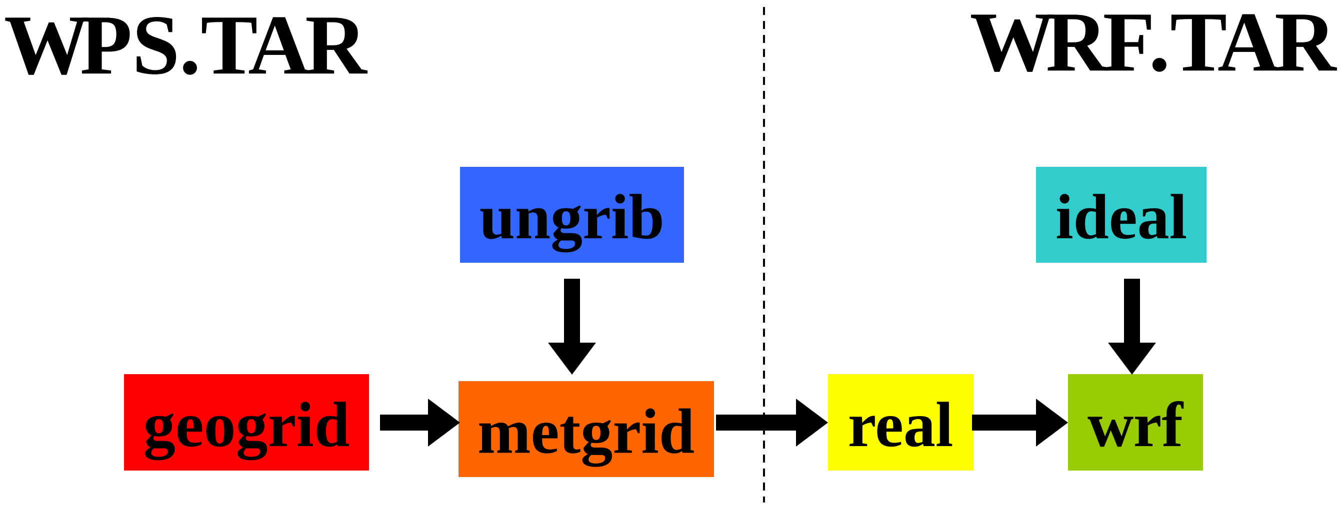
Nesting (One-Way)

- Same as two-way nesting but with no feedback

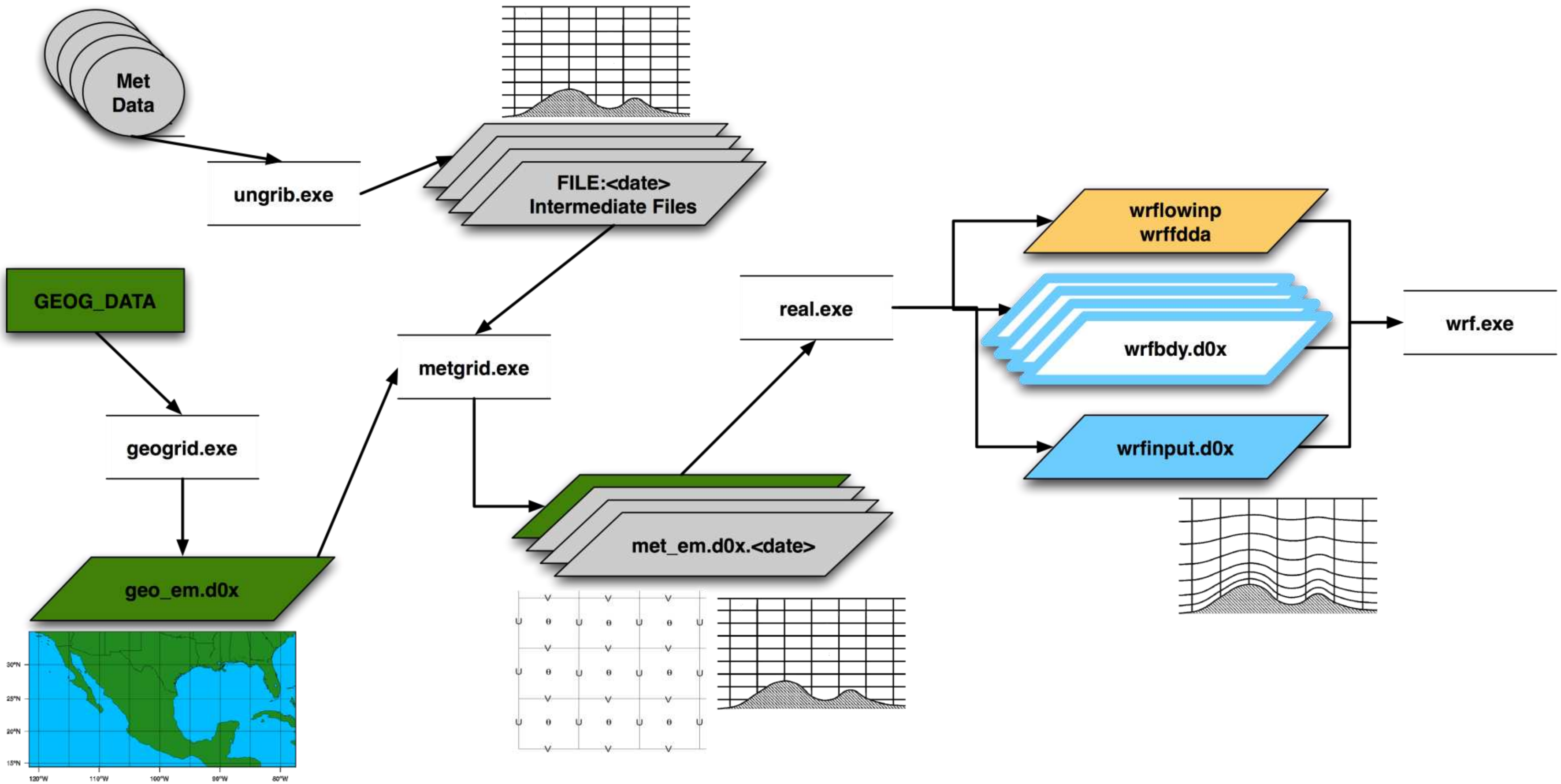
WRF Modeling System Flow Chart



WPS and WRF Program Flow



Data Flow


Met
Data
ungrib.exe
FILE:<date>
Intermediate Files
GEOG_DATA
metgrid.exe
geogrid.exe
geo_em.d0x
met_em.d0x.<date>
real.exe
wrflowinp
wrffdda
wrfbdy.d0x
wrfinput.d0x
wrf.exe

WPS Functions

- GEOGRID
- UNGRIB
- METGRID

WPS Functions

- GEOGRID
  - Define simulation area for domain(s)
  - Produce terrain, landuse, soil type etc. on the simulation domain ("static" fields)
- UNGRIB
- METGRID

WPS Functions

- GEOGRID
  - Define simulation area for domain(s)
  - Produce terrain, landuse, soil type etc. on the simulation domain ("static" fields)
- UNGRIB
  - De-grib GRIB files for meteorological data (e.g. u, v, T, RH, SLP)
- METGRID

WPS Functions

- GEOGRID
  - Define simulation area for domain(s)
  - Produce terrain, landuse, soil type etc. on the simulation domain (“static” fields)
- UNGRIB
  - De-grib GRIB files for meteorological data (e.g. u, v, T, RH, SLP)
- METGRID
  - Horizontally interpolate meteorological data to WRF model grid

WRF real.exe functions

- REAL
  - Inputs data from WPS and outputs data for use by the WRF model
  - Creates initial IC and boundary condition BC files for real-data cases
  - Vertically interpolates to model levels
  - Computes a vertical hydrostatic balance
  - Vertical interpolation of soil
  - Various data consistency checks

WRF Model

- WRF
  - Uses IC and lateral BC from REAL
  - Runs the model simulation with run-time selected namelist switches: physics choices, time-step, length of simulation
  - Outputs various history streams and restart files

Dynamical Core

- Basic Dynamical Equations:
  - Advection
  - Coriolis
  - Pressure gradient terms
  - Buoyancy
  - Diffusion

Dynamical Core

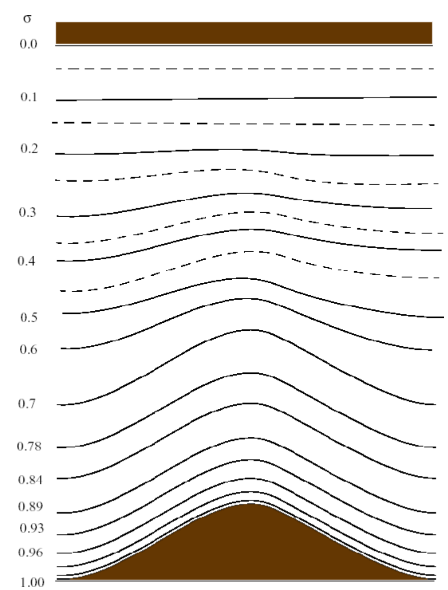
- Finite differencing:
  - Staggered grid-structure
  - Time-stepping method
  - Numerical filters

ARW Dynamics

Key features:

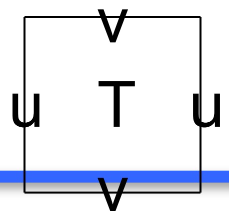
- Fully compressible, non-hydrostatic (with hydrostatic option)
- Mass-based terrain following coordinate, η

$$\eta = \frac{(\pi - \pi_t)}{\mu}, \qquad \mu = \pi_s - \pi_t$$



where π is hydrostatic pressure,
μ is column mass

- Arakawa C-grid staggering



ARW Model

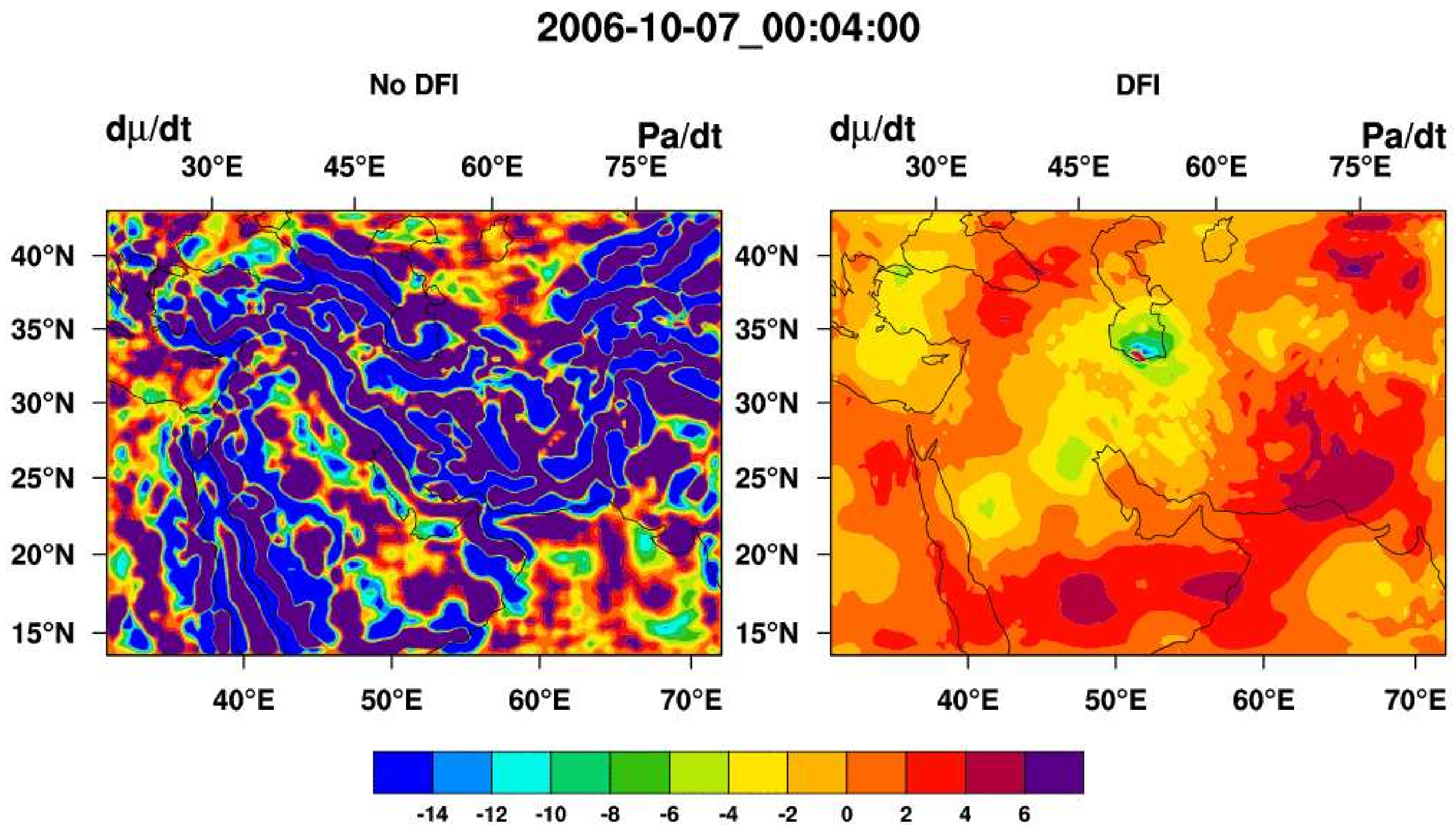
Key features:

- 3rd-order Runge-Kutta time integration scheme
- High-order advection scheme
- Scalar-conserving (positive definite option)
- Nesting

ARW Model

Key features:

- Choices of lateral boundary conditions suitable for real-data and idealized simulations
- Full physics options to represent atmospheric radiation, surface and boundary layer, and cloud and precipitation processes
- Grid-nudging and obs-nudging (FDDA)
- Digital Filter Initialization (DFI) option


2006-10-07_00:04:00
No DFI
DFI
dμ/dt
Pa/dt
dμ/dt
Pa/dt
30°E
45°E
60°E
75°E
40°N
35°N
30°N
25°N
20°N
15°N
40°E
50°E
60°E
70°E
-14
-12
-10
-8
-6
-4
-2
0
2
4
6

Graphics and Verification Tools

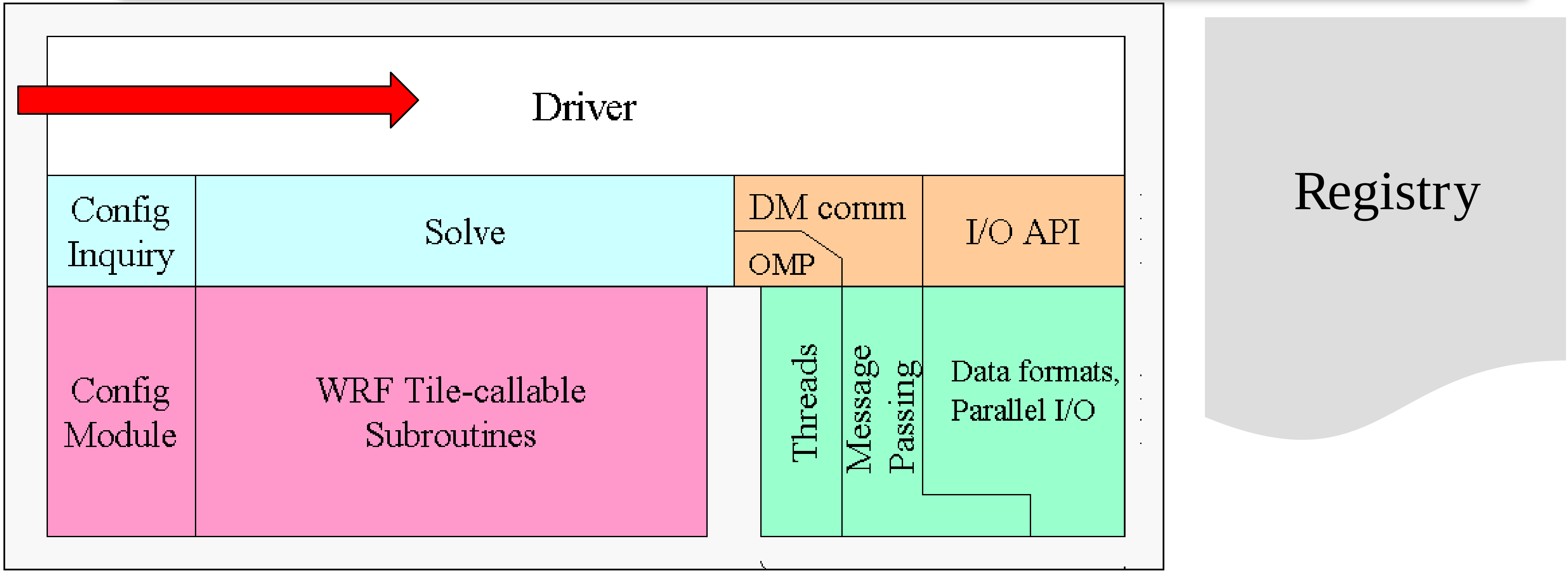
- NCAR Graphics Command Language (NCL)
- RIP4 (Read, Interpolate and Plot)
- Unified Post-Processor (UPP)
- ARWpost
- VAPOR (3D visualization tool)
- IDV (3D visualization tool)

- MET (Model Evaluation Toolkit)

Basic Software Requirement

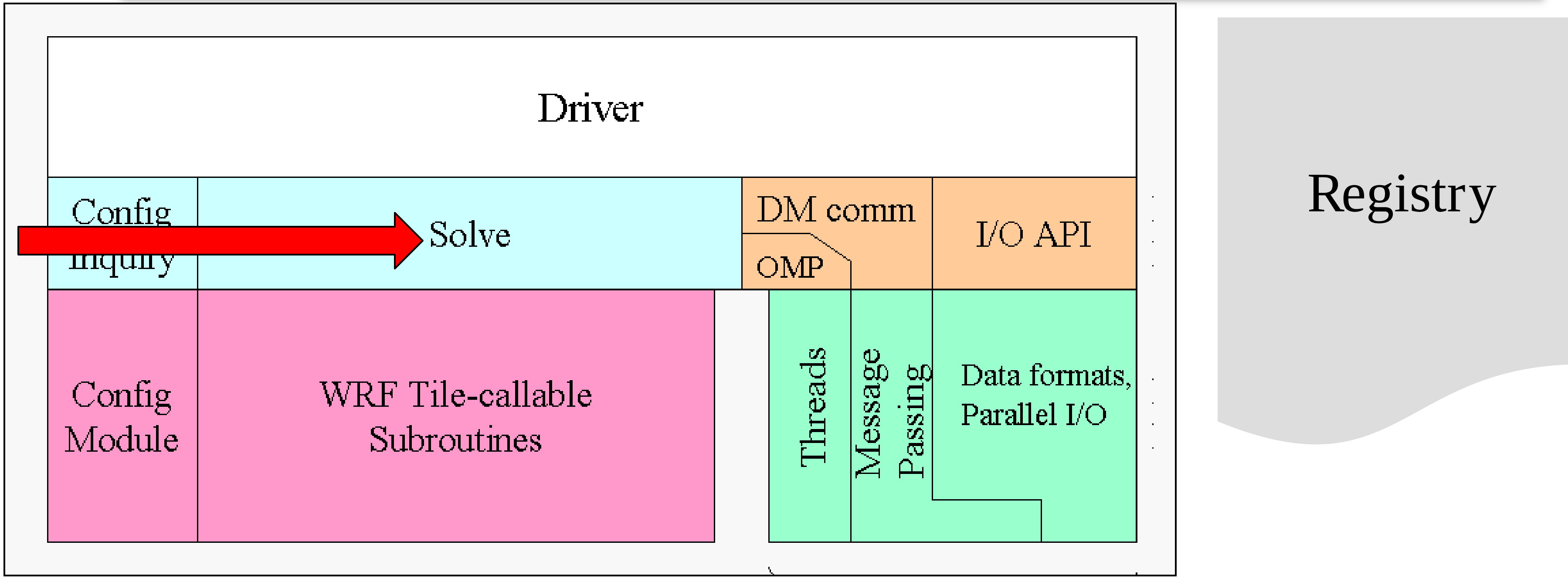
- Fortran compiler
  - Code adheres relatively closely to the standard
- C compiler
  - "Registry"-based automatic Fortran code generation (for argument lists, declarations, nesting functions, I/O routines)
- sh, csh, perl
  - configure/compile scripts
- netcdf library
  - for I/O (other I/O formats semi-supported)
- Public domain mpich/OpenMPI for MPI
  - if using distributed memory option

Code Layers



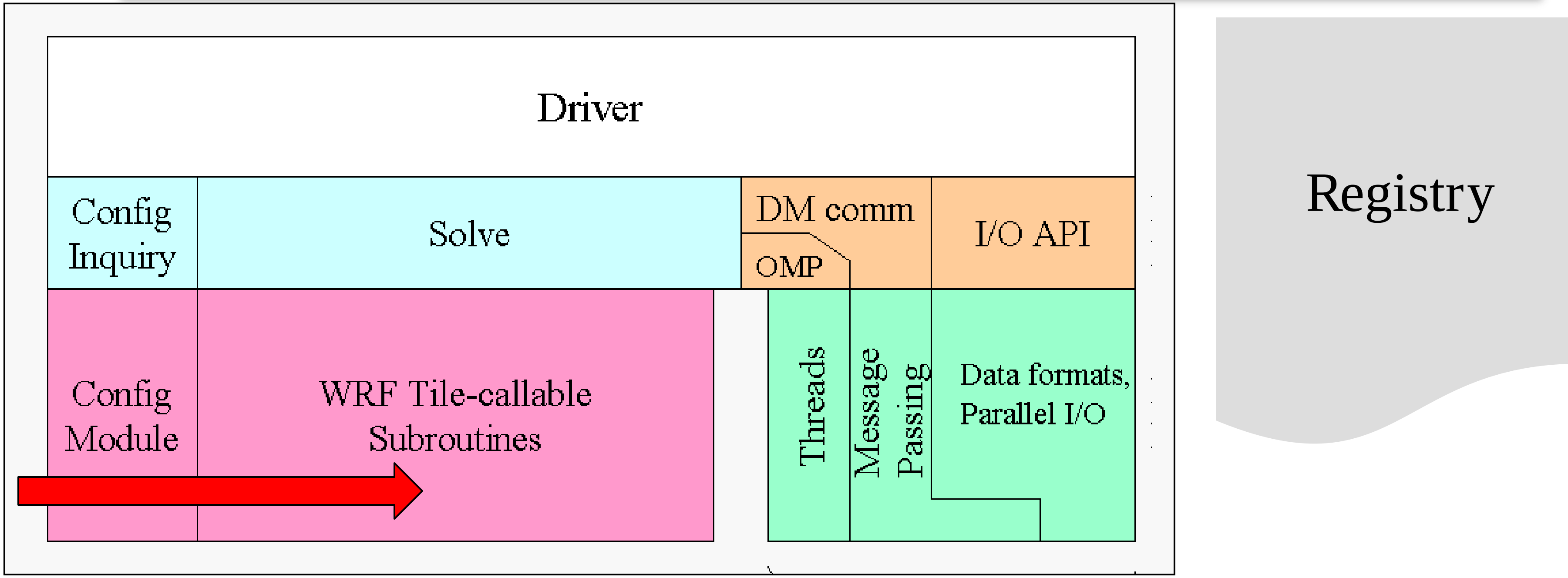
Top-level (framework): allocates space, handles nested domains and interpolation/feedback functions, time-stepping, solver calls, and i/o file contents and calls

Code Layers



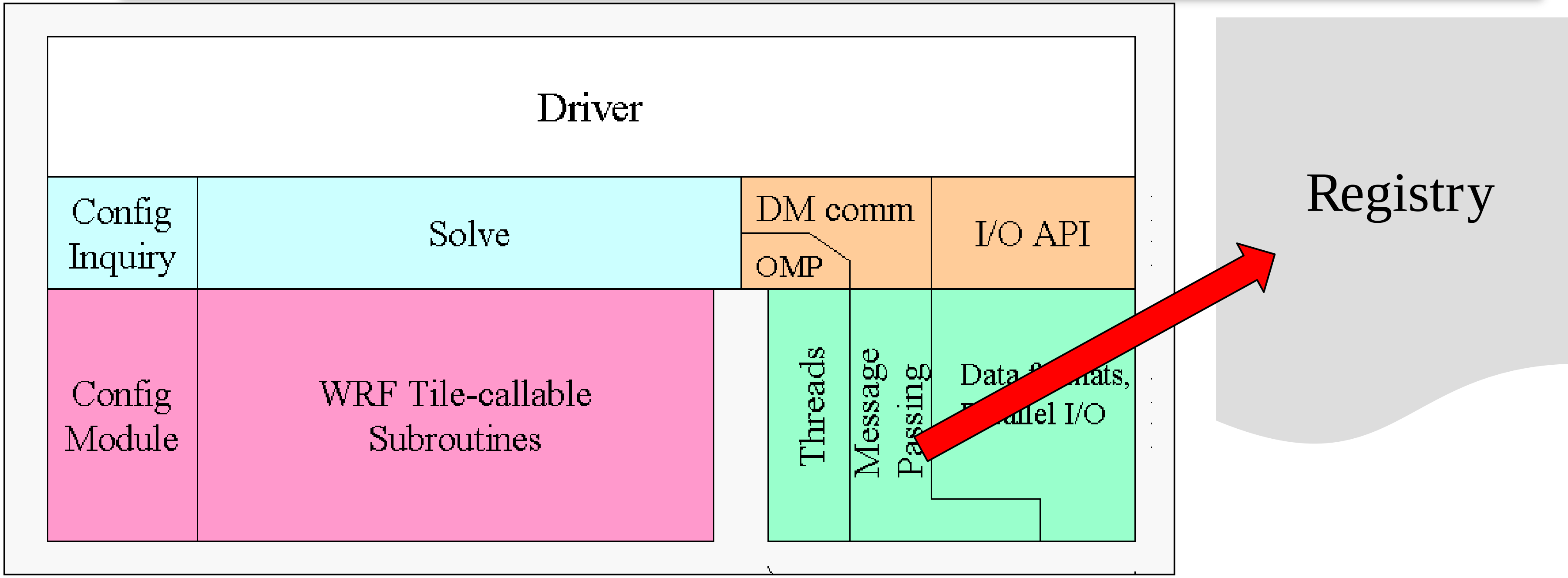
Intermediate level: “start” routine for initial calls,
“solve” routine for run-time advancing, communications

Code Layers



Low-level: science code in plain Fortran (no MPI or I/O calls)

Code Layers

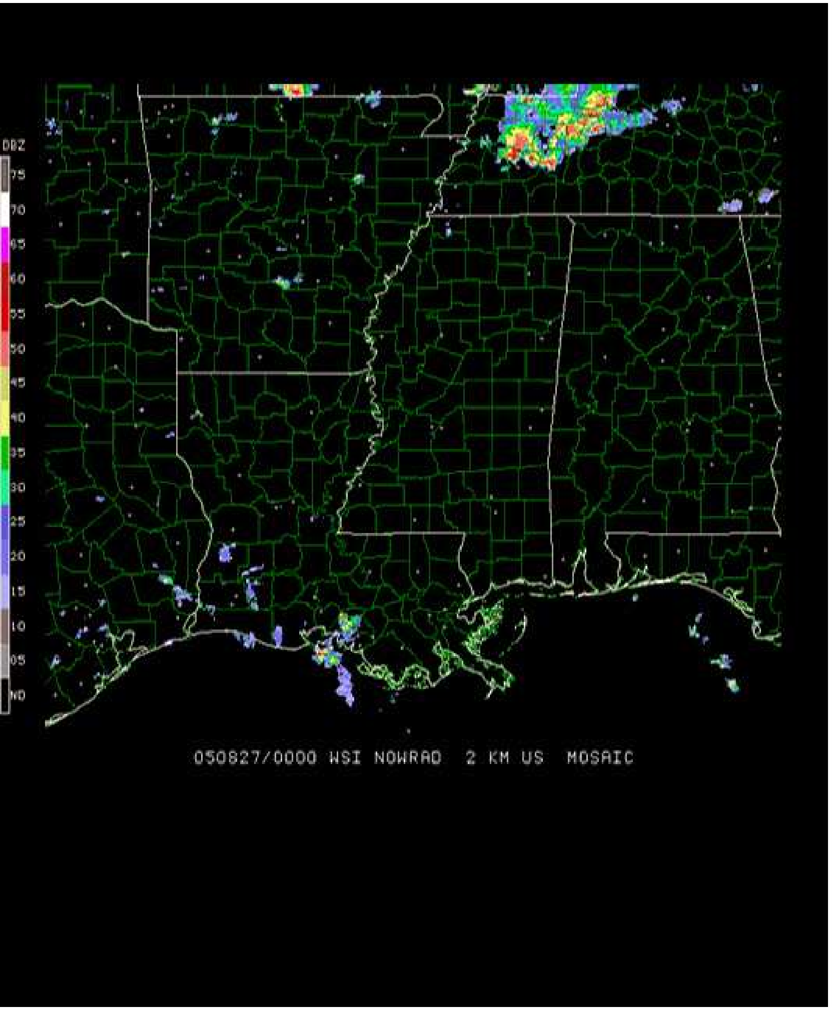


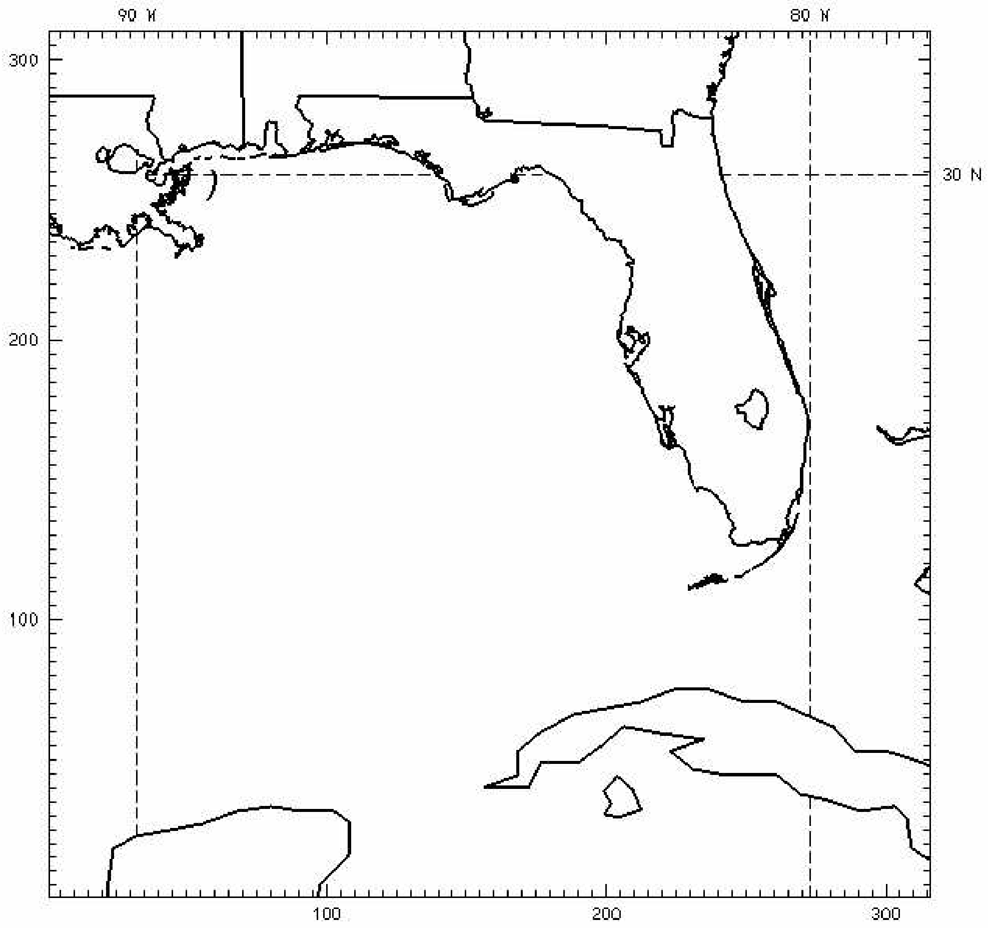
Active data dictionary; primary purpose is to make adding variables simple and safe. Controls all WRF data requiring IO and communications: dimensions, name, nesting, staggering, and time-level information.

User Support

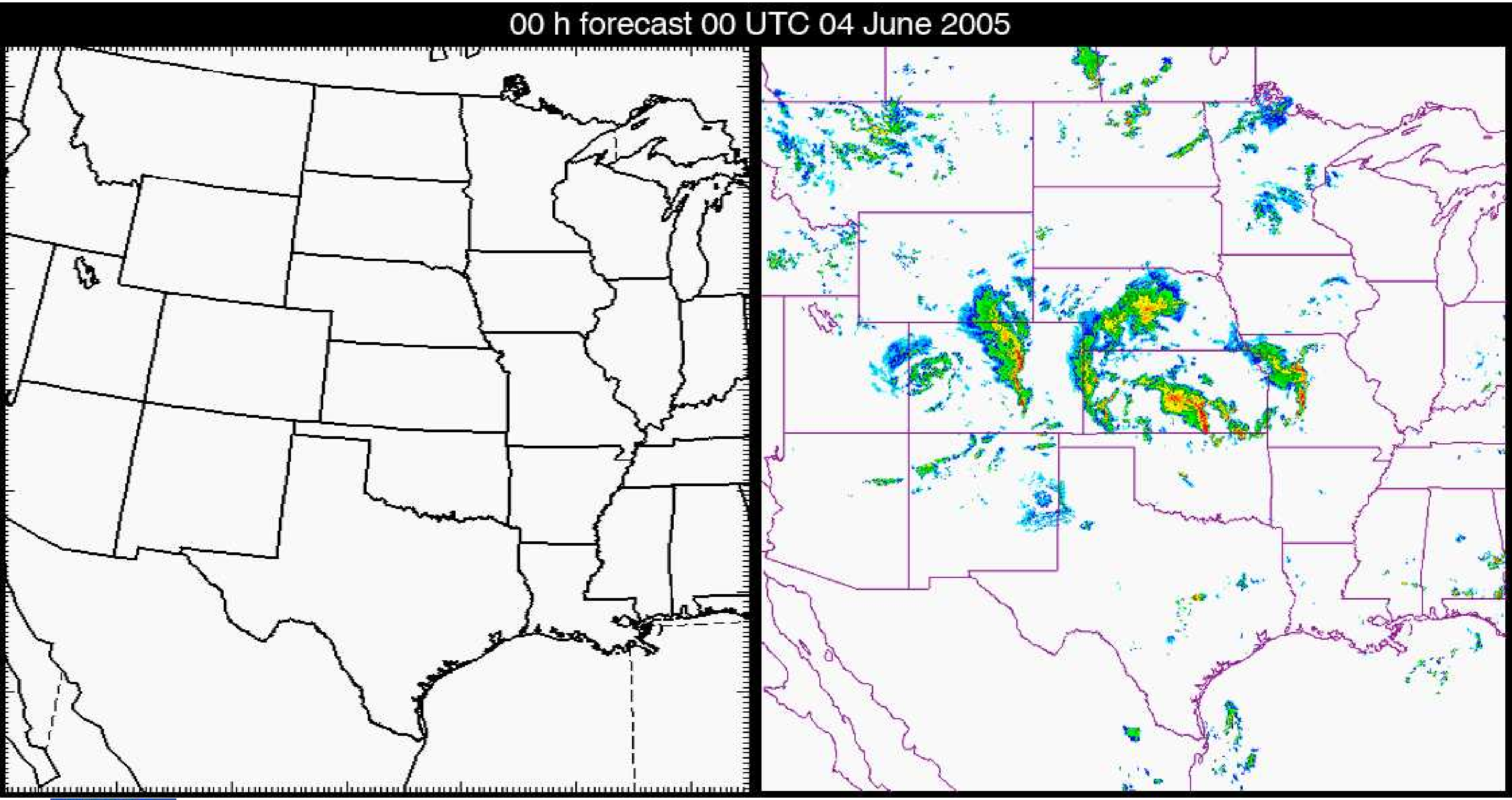
- Email: wrfhelp@ucar.edu
- User Web pages:
  http://www.mmm.ucar.edu/wrf/users/
  - Events schedule (tutorials, workshops)
  - WRF software download
  - Documentation
    - User's Guide
    - Technical Note

ARW Hurricane Katrina Simulation (4km)



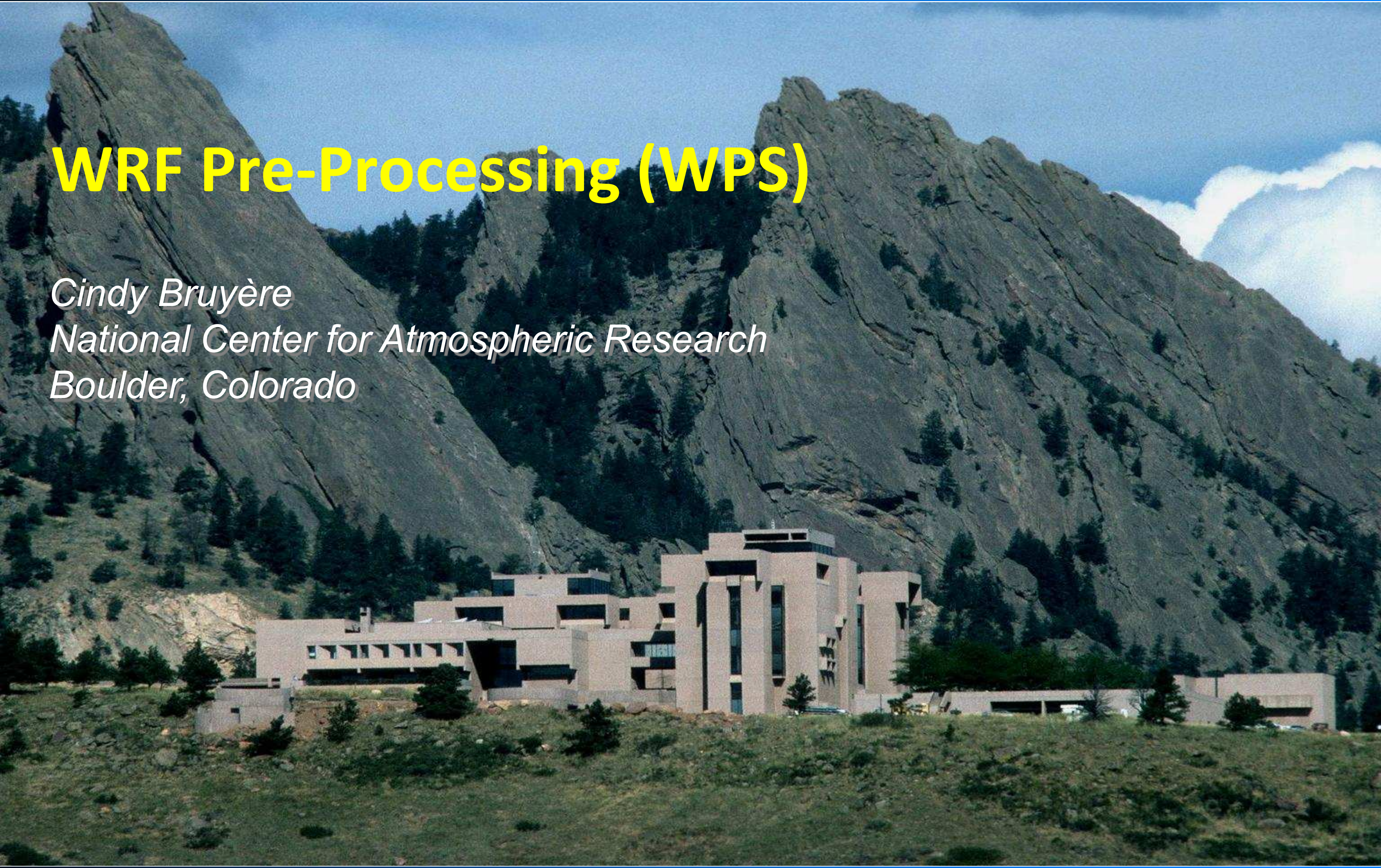


ARW Convective-scale Forecasting (4km)



Mesoscale & Microscale Meteorological Division / NCAR

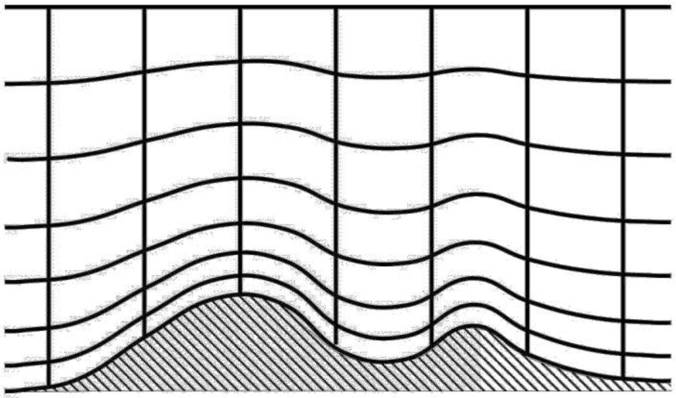
NCAR is Sponsored by NSF and this work is partially supported by the Willis Research Network and the Research Partnership to Secure Energy for America

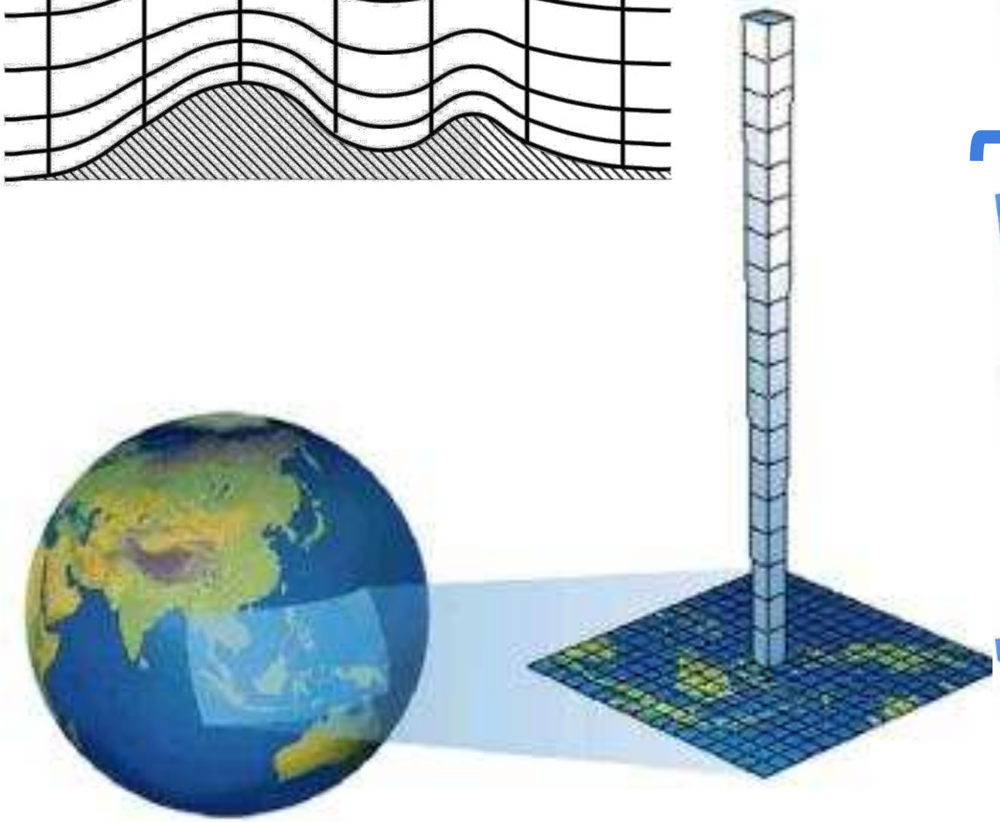


WRF Pre-Processing (WPS)

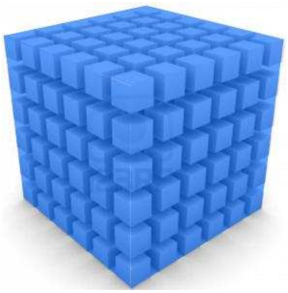
Cindy Bruyère
National Center for Atmospheric Research
Boulder, Colorado

Requirements for WRF Model

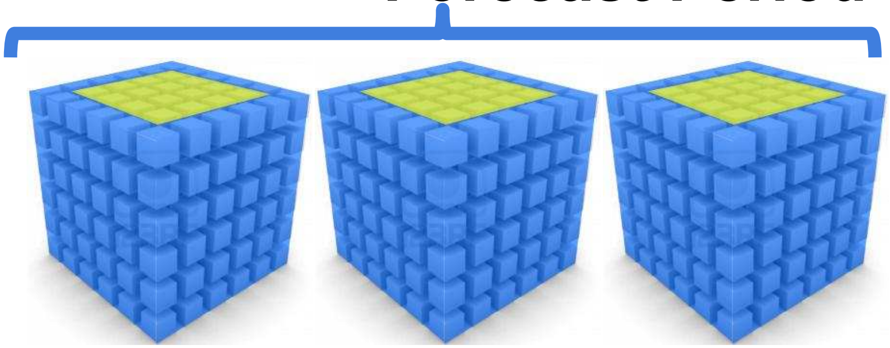




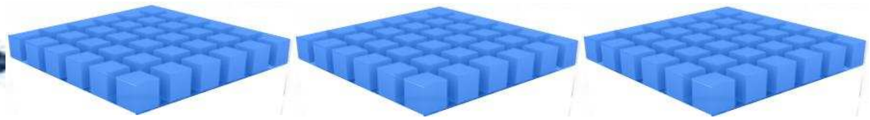
Initial Conditions



Forecast Period

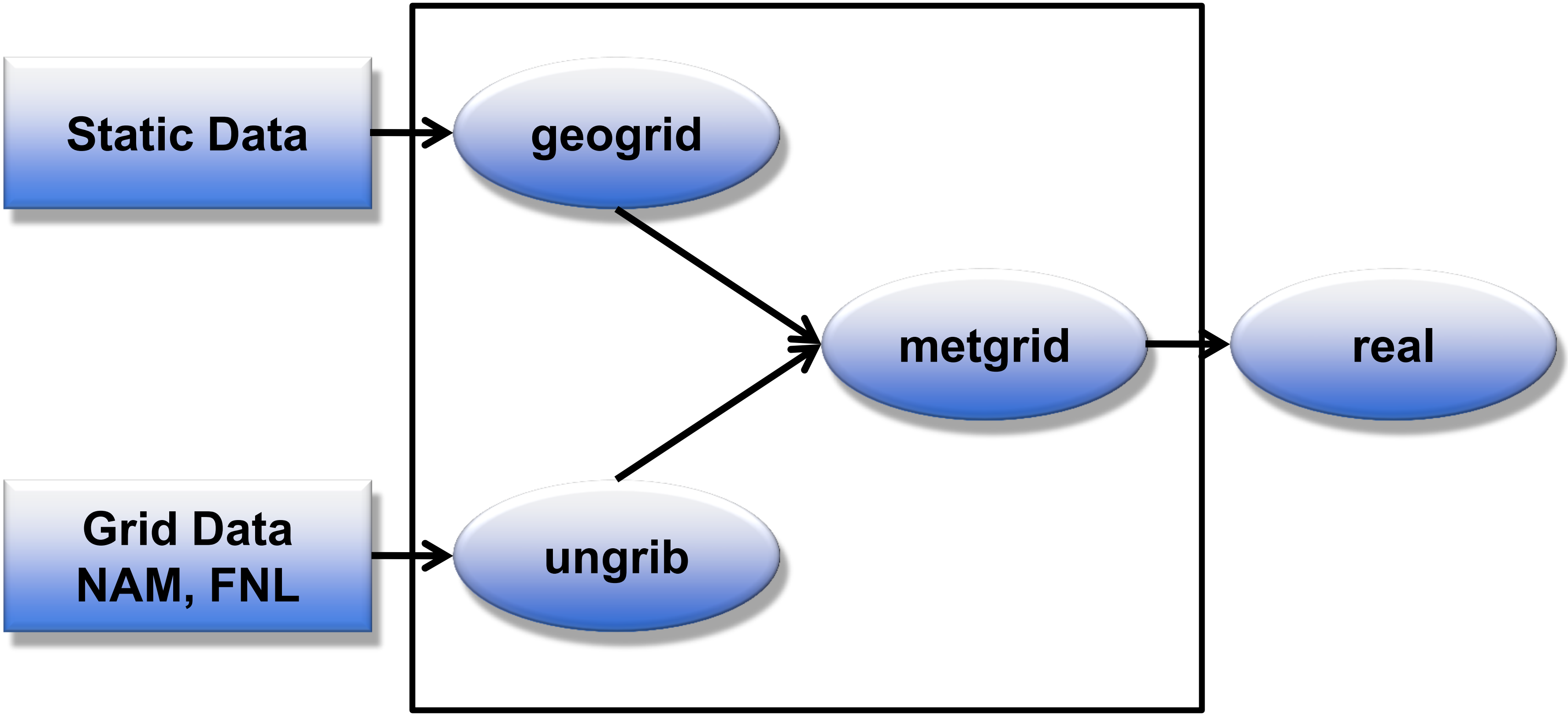


Boundary Conditions



Low Boundary Conditions

WRF Pre-Processing System (WPS)



External GRIB Data sources

- North American AWIP data
  http://dss.ucar.edu/datasets/ds609.2/

- North American NAM data
  http://www.emc.ncep.noaa.gov/

- North American Regional Reanalysis data
  http://dss.ucar.edu/pub/narr/

- NNRP (R1)
  http://dss.ucar.edu/datasets/ds090.0/

- NCEP / DOE Reanalysis II (Jan 1979 to Dec 2010)
  http://search.dss.ucar.edu/datasets/ds091.0/

External GRIB Data sources

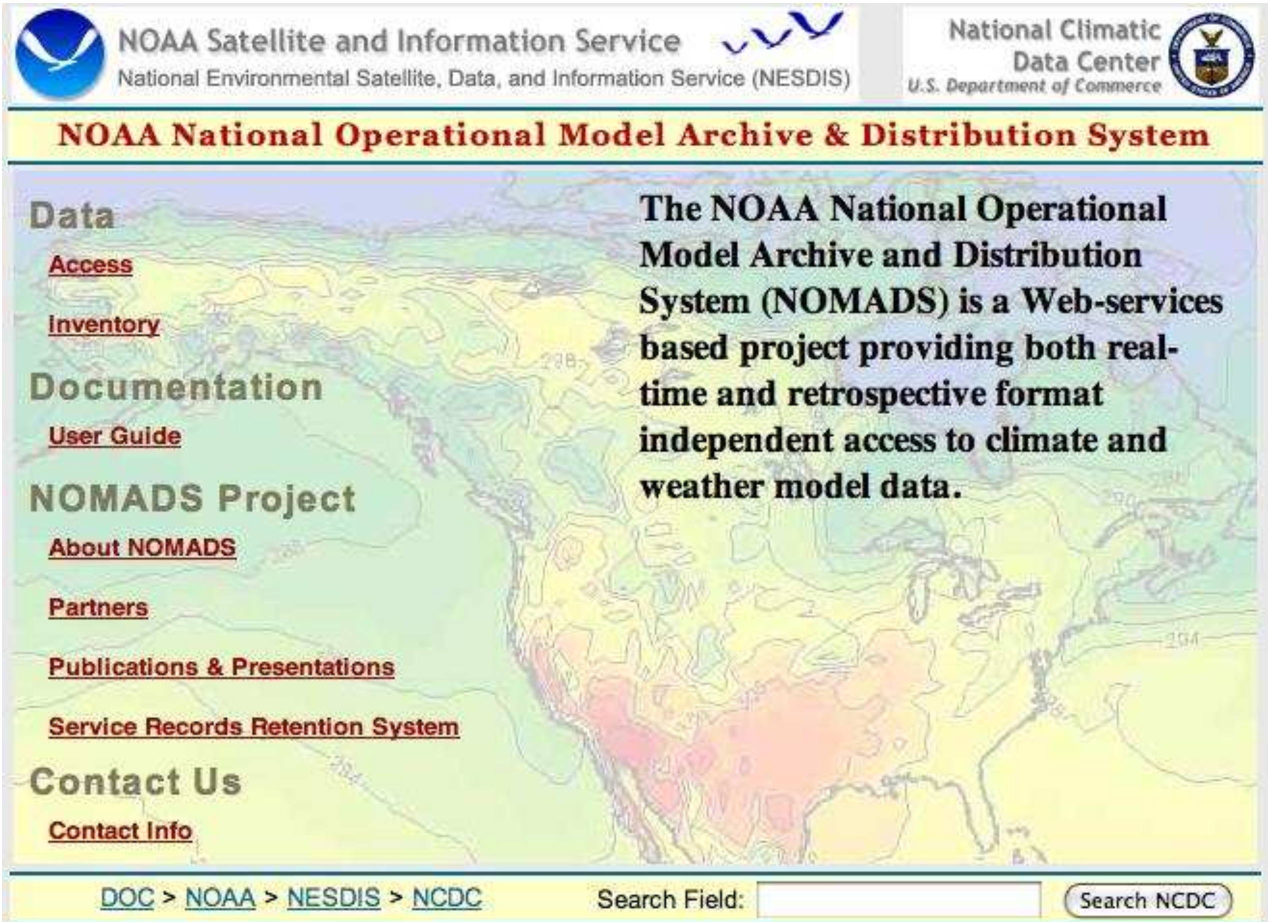
- GFS Real-time 1^o data
  http://www.emc.ncep.noaa.gov/
- Final Analysis (FNL)
  http://dss.ucar.edu/datasets/ds083.2
- GFS 0.5^o Global data (Dec 2002 to present)
  http://dss.ucar.edu/datasets/ds335.0/
- SST data
  http://polar.ncep.noaa.gov/sst/oper/Welcome.html

External GRIB Data sources

- ERA Interim Data (~ 0.7o Global data; Jan 1979 to Sep 2011) http://search.dss.ucar.edu/datasets/ds627.0/

- NCEP Climate Forecast System Reanalysis (CFSR) (~38km, global data; Jan 1979 to Dec 2010) http://dss.ucar.edu/pub/cfsr.html

http://nomads.ncdc.noaa.gov



NAM
GFS
RUC
CFS
NARR
R1/R2
SST

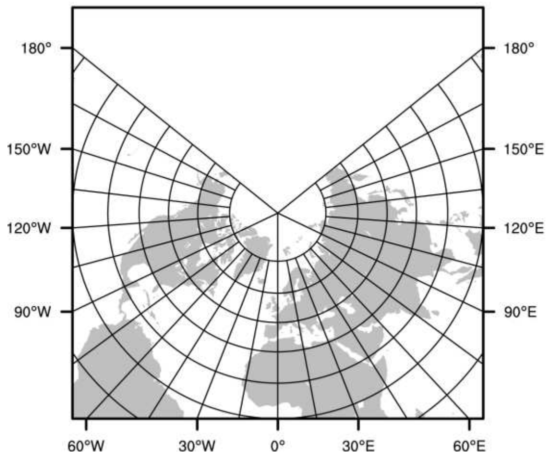
Purpose of the WPS

- The purpose of the WPS is to prepare input to WRF for real-data simulations:

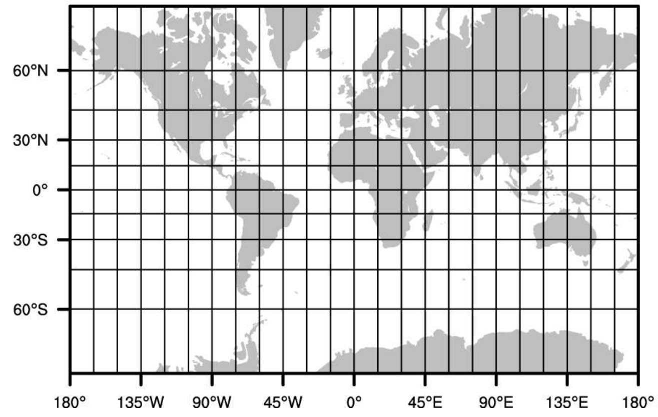
  1. Defines simulation coarse domain and nested domains
  2. Computes latitude, longitude, map scale factors, and Coriolis parameters at every grid point
  3. Interpolates time-invariant terrestrial data to simulation grids (e.g., terrain height and soil type)
  4. Interpolates time-varying meteorological fields from another model onto simulation domains

WPS - geogrid

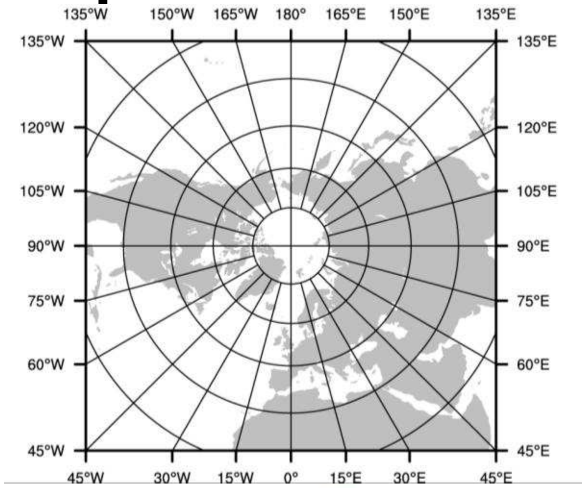
lambert

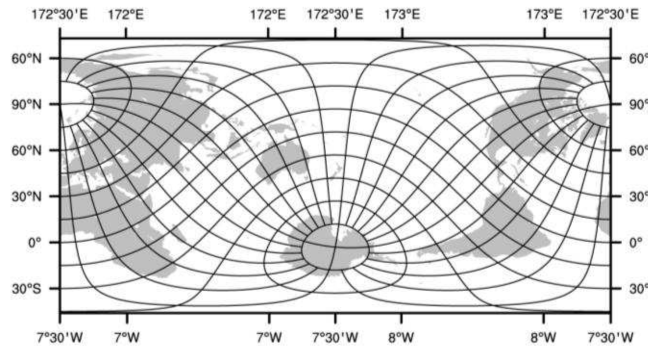


mercator

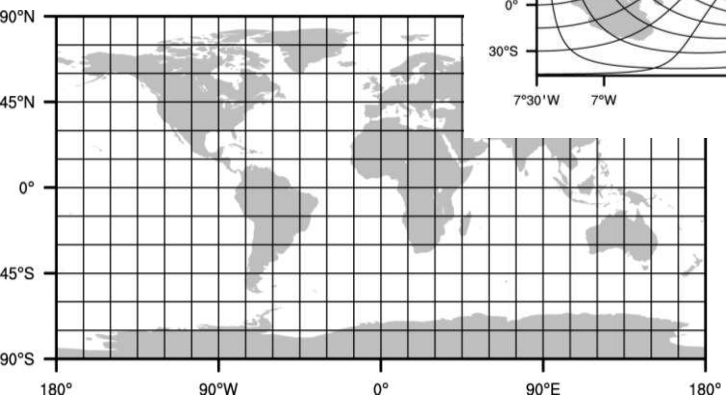


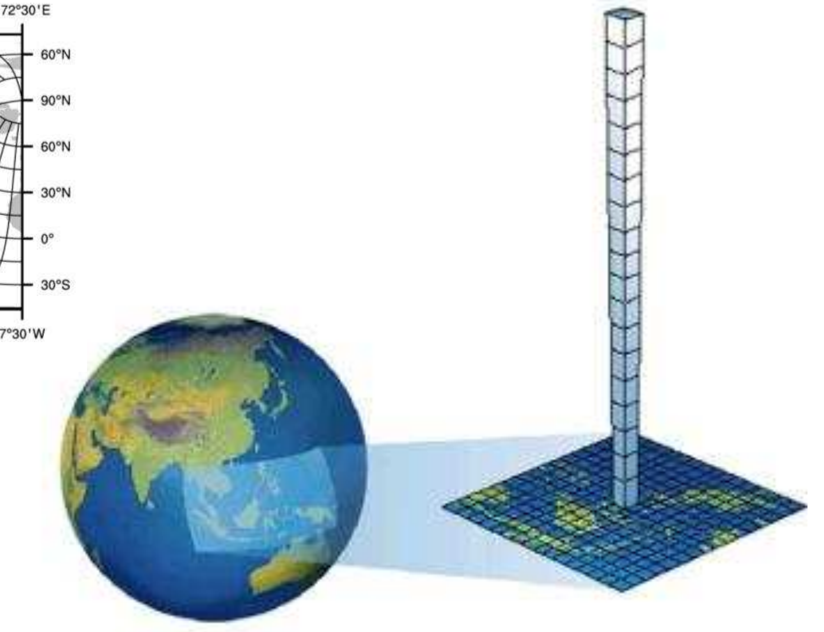
ps



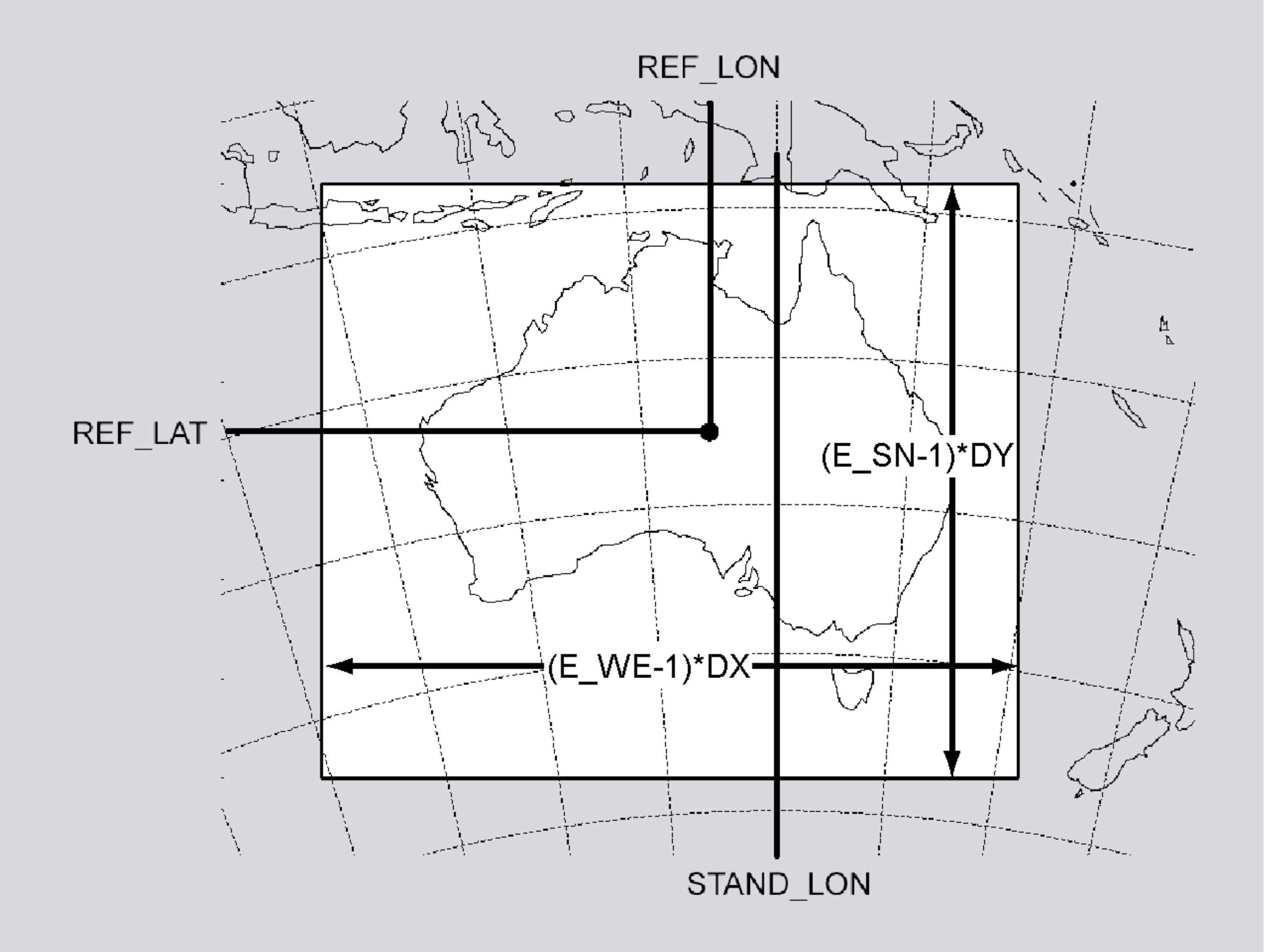


lat-lon

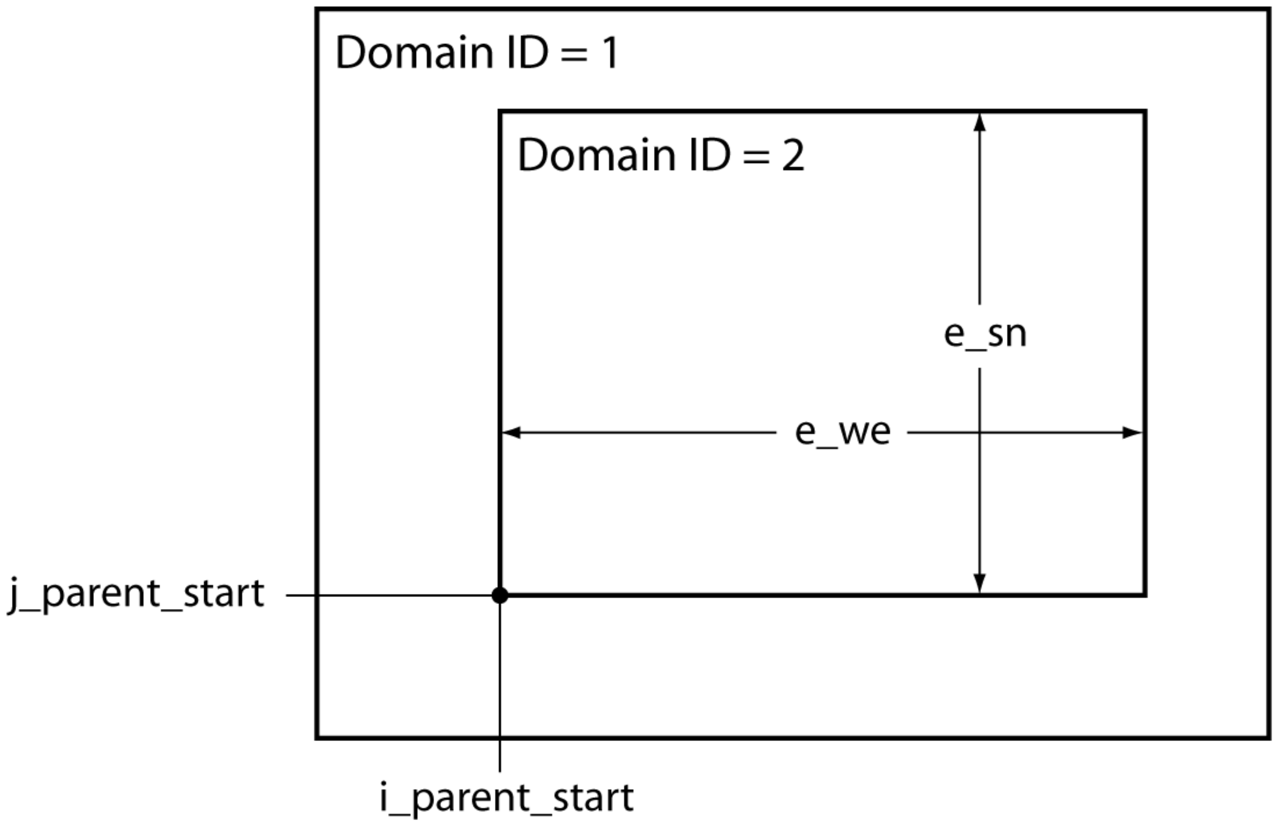




Geogrid: Defining ARW Domains



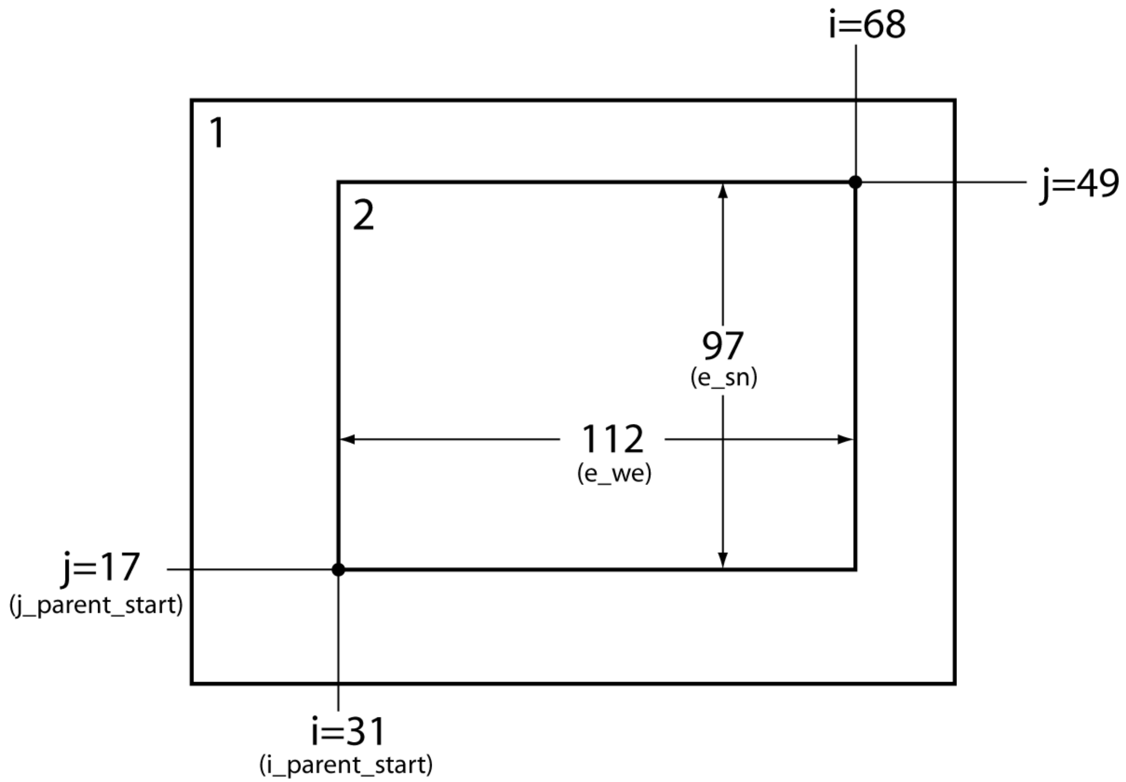
Geogrid: Defining Nested Domains



The grid spacing (*dx)* of domain 2 is determined by grid spacing of domain 1 and the *parent_grid_ratio*

Geogrid: Nesting example

Assuming *parent_grid_ratio* = 3



In ARW, nest dimensions must be

(*n*parent_grid_ratio* + 1)

for some integer *n*

112 = 3*n+1 for n=37

97 = 3*n+1 for n=32

Geogrid: Defining Model Domains

- Define projection of domains using a subset of the following parameters
  - **MAP_PROJ**: ‘lambert’, ‘mercator’, ‘polar’, or ‘lat-lon’
  - **TRUELAT1**: First true latitude
  - **TRUELAT2**: Second true latitude (*only for Lambert conformal*)
  - **POLE_LAT, POLE_LON**: Location of North Pole in WRF computational grid (*only for ‘lat-lon’*)
  - **STAND_LON**: The meridian parallel to y-axis
- All parameters reside in the file *namelist.wps*

See p. 3-9 and 3-43

Geogrid: Defining Model Domains

- Define the area covered (dimensions and location) by coarse domain using the following:
  - **REF_LAT**, **REF_LON**: The (lat,lon) location of a known location in the domain (*by default, the center point of the domain*)
  - **DX**, **DY**: Grid distance where map factor = 1
    - For Lambert, Mercator, and polar stereographic: **meters**
    - For (rotated) latitude-longitude: **degrees**
  - **E_WE**: Number of velocity points in west-east direction
  - **E_SN**: Number of velocity points in south-north direction

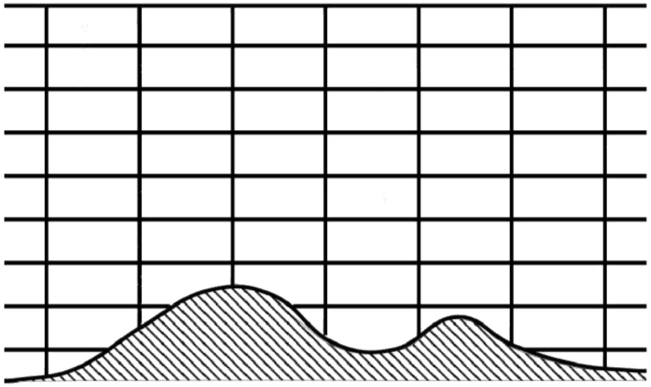
See p. 3-13 and 3-42

Geogrid: Defining Nested Domains

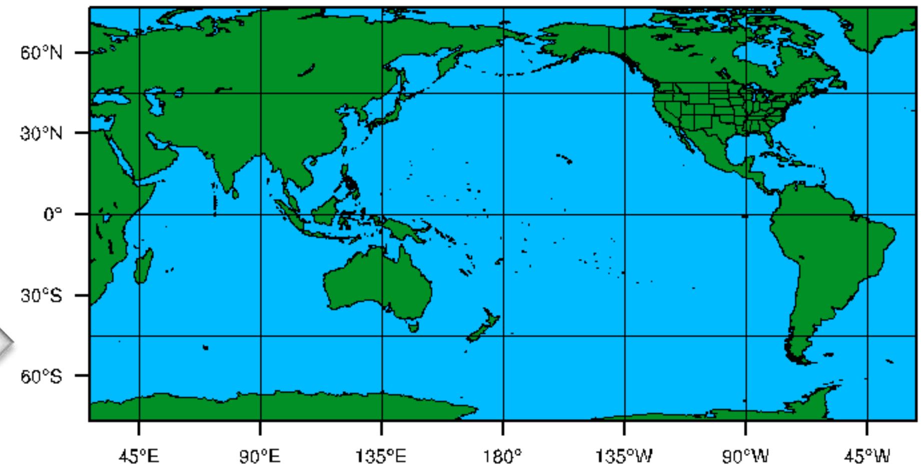
- Define the dimensions and location of nested domains using:
  - **PARENT_ID**: Which domain is the parent?
  - **PARENT_GRID_RATIO**: What is the ratio of grid spacing in parent to grid spacing in this nest?
  - **I_PARENT_START**: *i*-coordinate in parent of this nest's lower-left corner
  - **J_PARENT_START**: *j*-coordinate in parent of this nest's lower-left corner
  - **E_WE**: Number of velocity points in west-east direction
  - **E_SN**: Number of velocity points in south-north direction

See p. 3-20 and 3-42

WPS - ungrib



grib data
Vtable

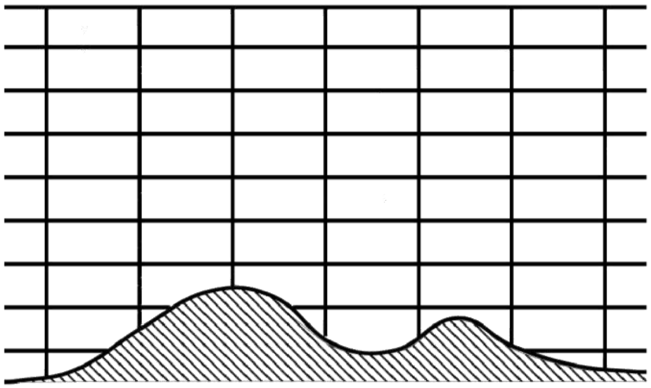


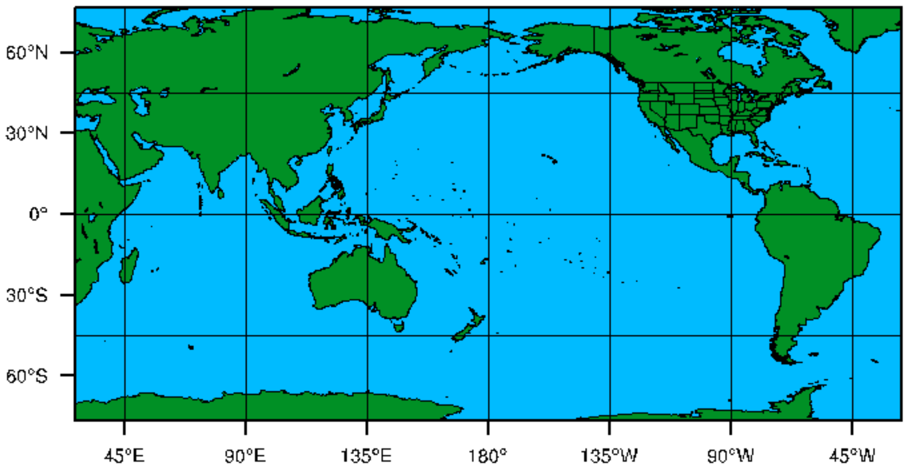
Intermediate Files
Binary Data
Same projections as global data
Only fields requested in Vtables

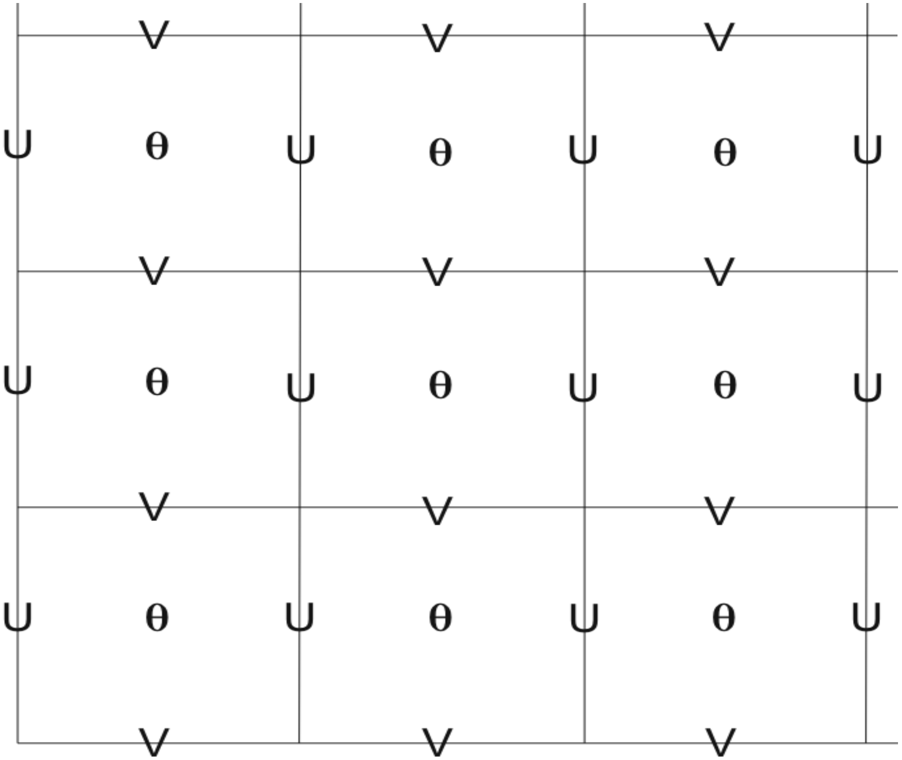
Ungrib: Example Vtable

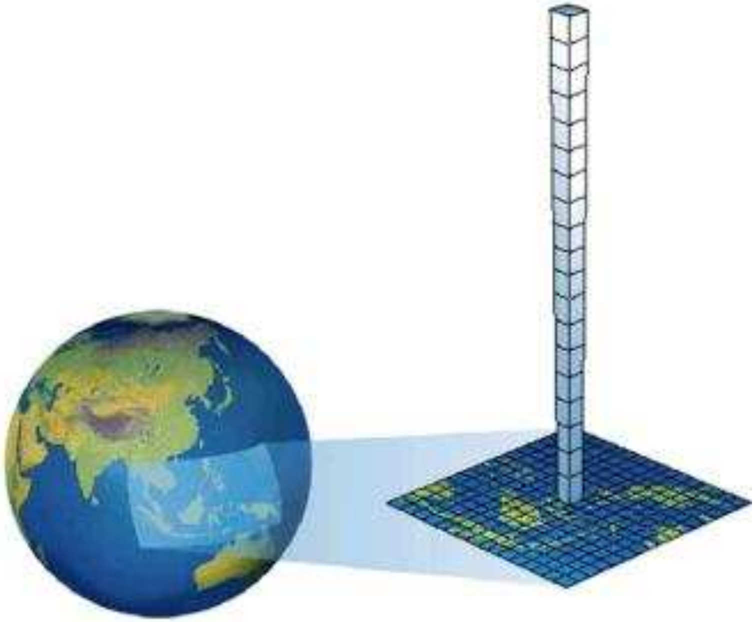
```
GRIB1| Level| From |  To  | UNGRIB   |  UNGRIB | UNGRIB
Param| Type |Level1|Level2| Name     |  Units  | Description
-----+-----+-----+-----+---------+--------+----------------------------------------
  11 | 100  |   *  |      | T        | K       | Temperature
  33 | 100  |   *  |      | U        | m s-1   | U
  34 | 100  |   *  |      | V        | m s-1   | V
  52 | 100  |   *  |      | RH       | %       | Relative Humidity
   7 | 100  |   *  |      | HGT      | m       | Height
  11 | 105  |   2  |      | T        | K       | Temperature       at 2 m
  52 | 105  |   2  |      | RH       | %       | Relative Humidity at 2 m
  33 | 105  |  10  |      | U        | m s-1   | U                 at 10 m
  34 | 105  |  10  |      | V        | m s-1   | V                 at 10 m
   1 |   1  |   0  |      | PSFC     | Pa      | Surface Pressure
 130 | 102  |   0  |      | PMSL     | Pa      | Sea-level Pressure
 144 | 112  |   0  |  10  | SM000010 | kg m-3  | Soil Moist 0-10 cm below grn layer
(Up)
 144 | 112  |  10  |  40  | SM010040 | kg m-3  | Soil Moist 10-40 cm below grn layer
 144 | 112  |  40  | 100  | SM040100 | kg m-3  | Soil Moist 40-100 cm below grn layer
 144 | 112  | 100  | 200  | SM100200 | kg m-3  | Soil Moist 100-200 cm below gr layer
  85 | 112  |   0  |  10  | ST000010 | K       | T 0-10 cm below ground layer (Upper)
  85 | 112  |  10  |  40  | ST010040 | K       | T 10-40 cm below ground layer (Upper)
  85 | 112  |  40  | 100  | ST040100 | K       | T 40-100 cm below ground layer (Upper)
  85 | 112  | 100  | 200  | ST100200 | K       | T 100-200 cm below ground layer
(Bottom)
  91 |   1  |   0  |      | SEAICE   | proprtn | Ice flag
  81 |   1  |   0  |      | LANDSEA  | proprtn | Land/Sea flag (1=land,2=sea in GRIB2)
   7 |   1  |   0  |      | HGT      | m       | Terrain field of source analysis
  11 |   1  |   0  |      | SKINTEMP | K       | Skin temperature (can use for SST
also)
  65 |   1  |   0  |      | SNOW     | kg m-2  | Water equivalent snow depth
 223 |   1  |   0  |      | CANWAT   | kg m-2  | Plant Canopy Surface Water
 224 |   1  |   0  |      | SOILCAT  | Tab4.213| Dominant soil type category
 225 |   1  |   0  |      | VEGCAT   | Tab4.212| Dominant land use category
-----+-----+-----+-----+---------+--------+----------------------------------------
```

WPS - metgrid



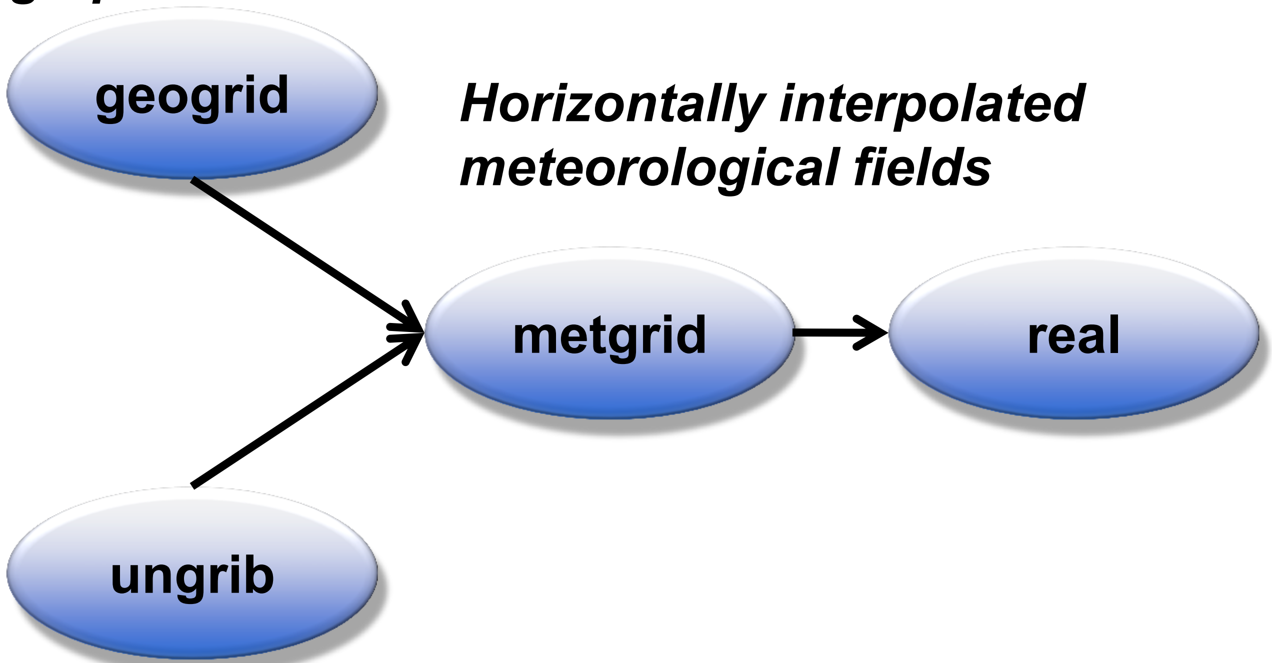






WRF Pre-Processing System (WPS)

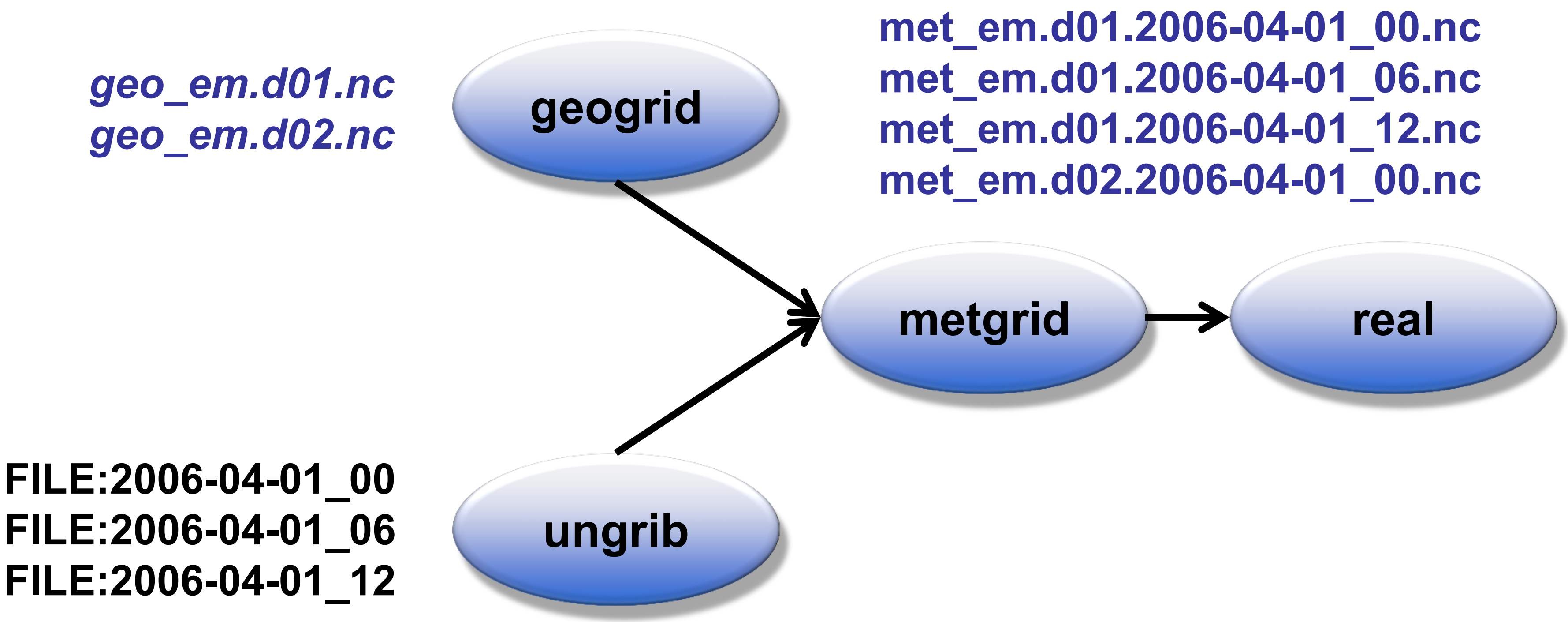
Domain definitions and interpolated geographical data



Horizontally interpolated meteorological fields

Meteorological fields from GRIB files

WRF Pre-Processing System (WPS)



namelist.wps

```
&share
   max_dom = 2,
   start_date  = '2006-04-01_00:00:00',  '2006-04-01_00:00:00',
   end_date    = '2006-04-01_12:00:00',  '2006-04-01_00:00:00',
   interval_seconds = 21600

 &geogrid
   parent_id                   =   1,      1,
   parent_grid_ratio           =   1,      3,
   i_parent_start              =   1,     20,
   j_parent_start              =   1,     17,
   e_we                        =  220,   181,
   e_sn                        =  175,   181,
   dx                          = 15000,
   dy                          = 15000,
   map_proj                    = 'lambert',
   ref_lat            = 37.0,
   ref_lon            = -97.0,
   truelat1                    = 45.0,
   truelat2                    = 30.0,
   stand_lon                   = -97.0,
   geog_data_path  = '/data/static/geog/’
```

RUN WPS

- ./geogrid.exe
  (get static data and set up namelist)

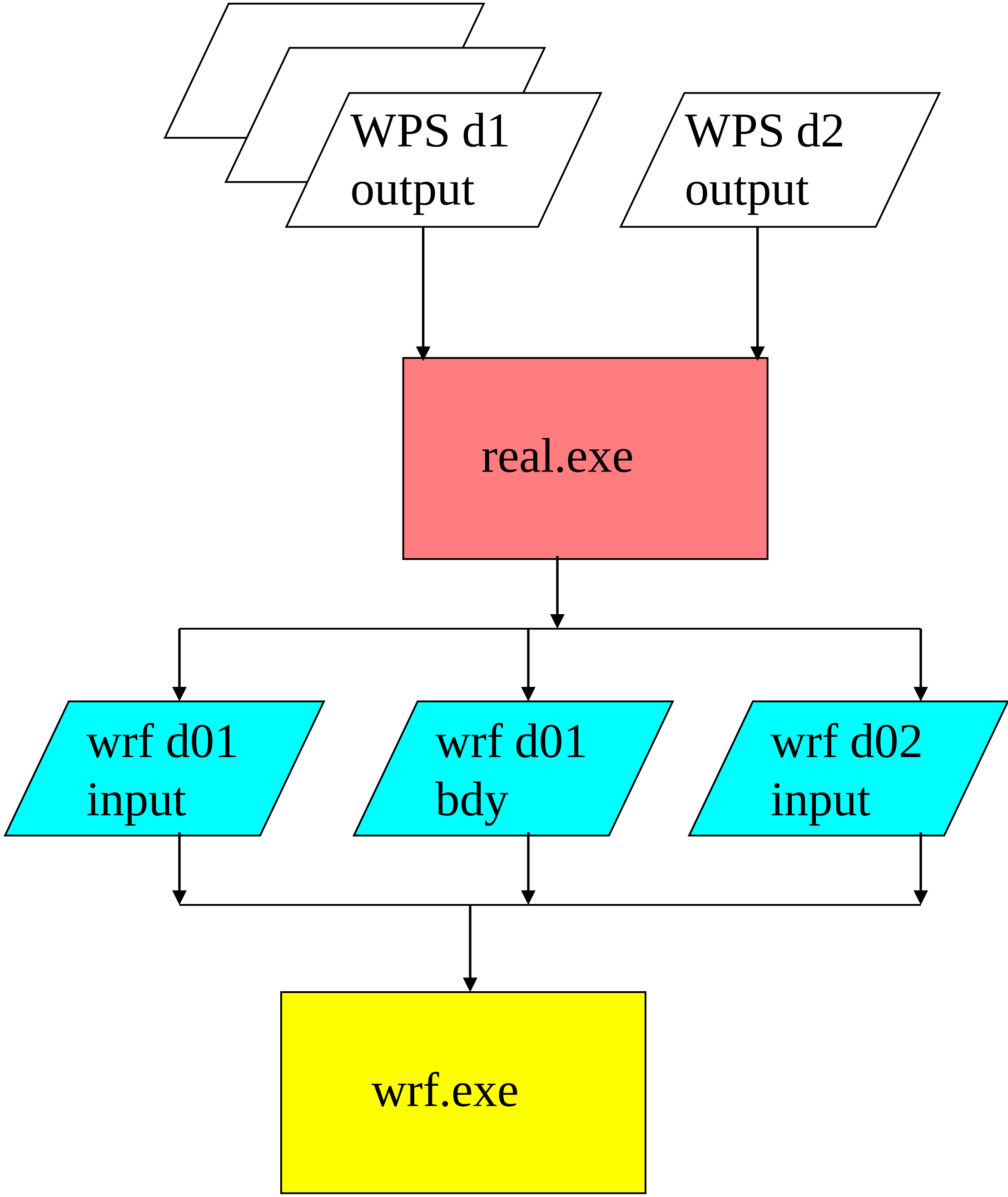
- ln –s ungrib/Variable_Tables/Vtable.***XXXXX*** Vtable
  link_grib.csh   [grib-data]
      GRIBFILE.???
  ./ungrib.exe

- metgrid.exe

Real and WRF

Description of General Functions

Jimy Dudhia
Dave Gill
Bill Skamarock



Real program in a nutshell

- Function
- Required input variables
- Base State
- Standard generated output
- Vertical interpolation
- Soil level interpolation
- Water temperature initialization
- Sea-ice initialization
- Land/Water mask

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

- 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

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

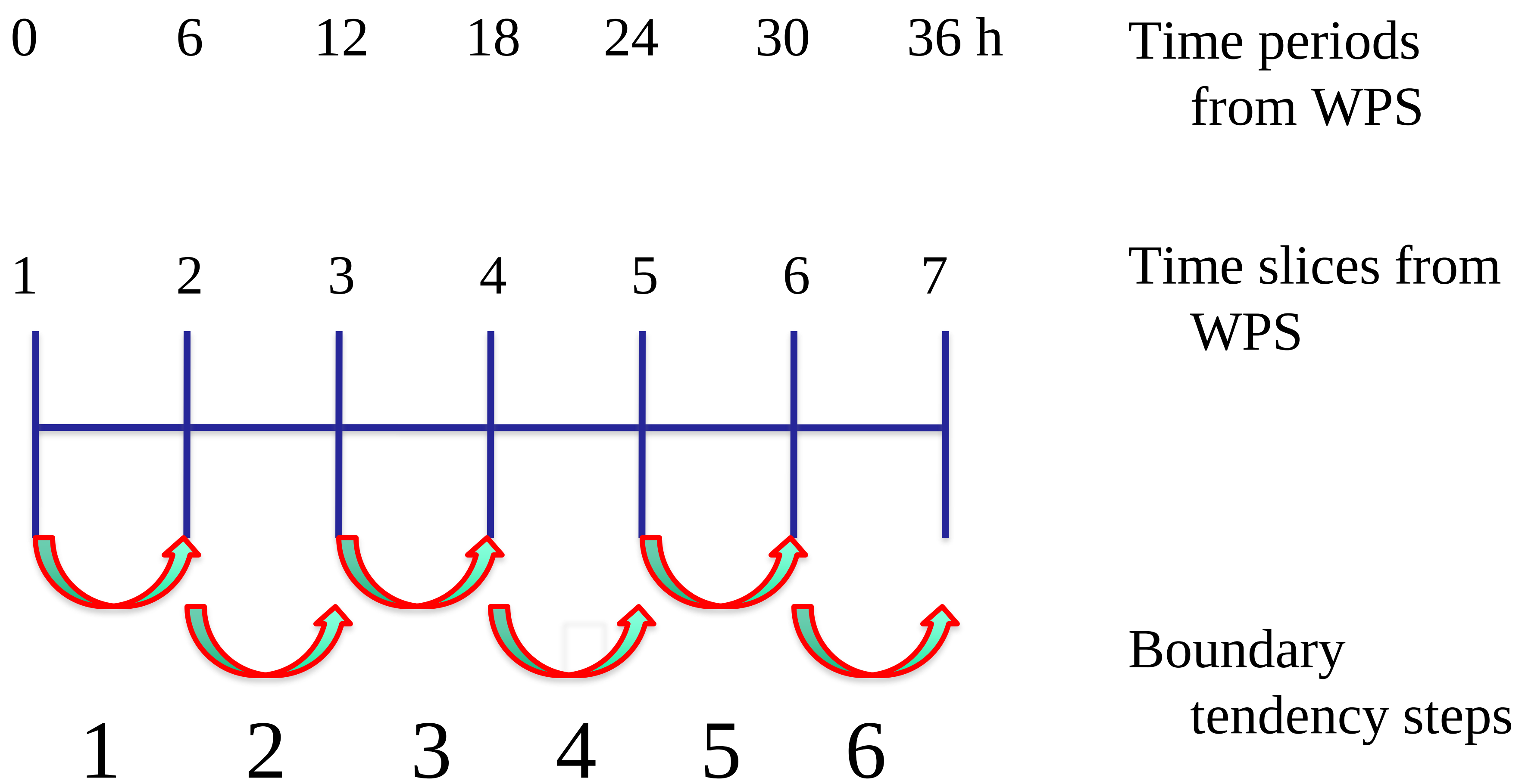
Base State

- 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
- No base state computations are required prior to the real program

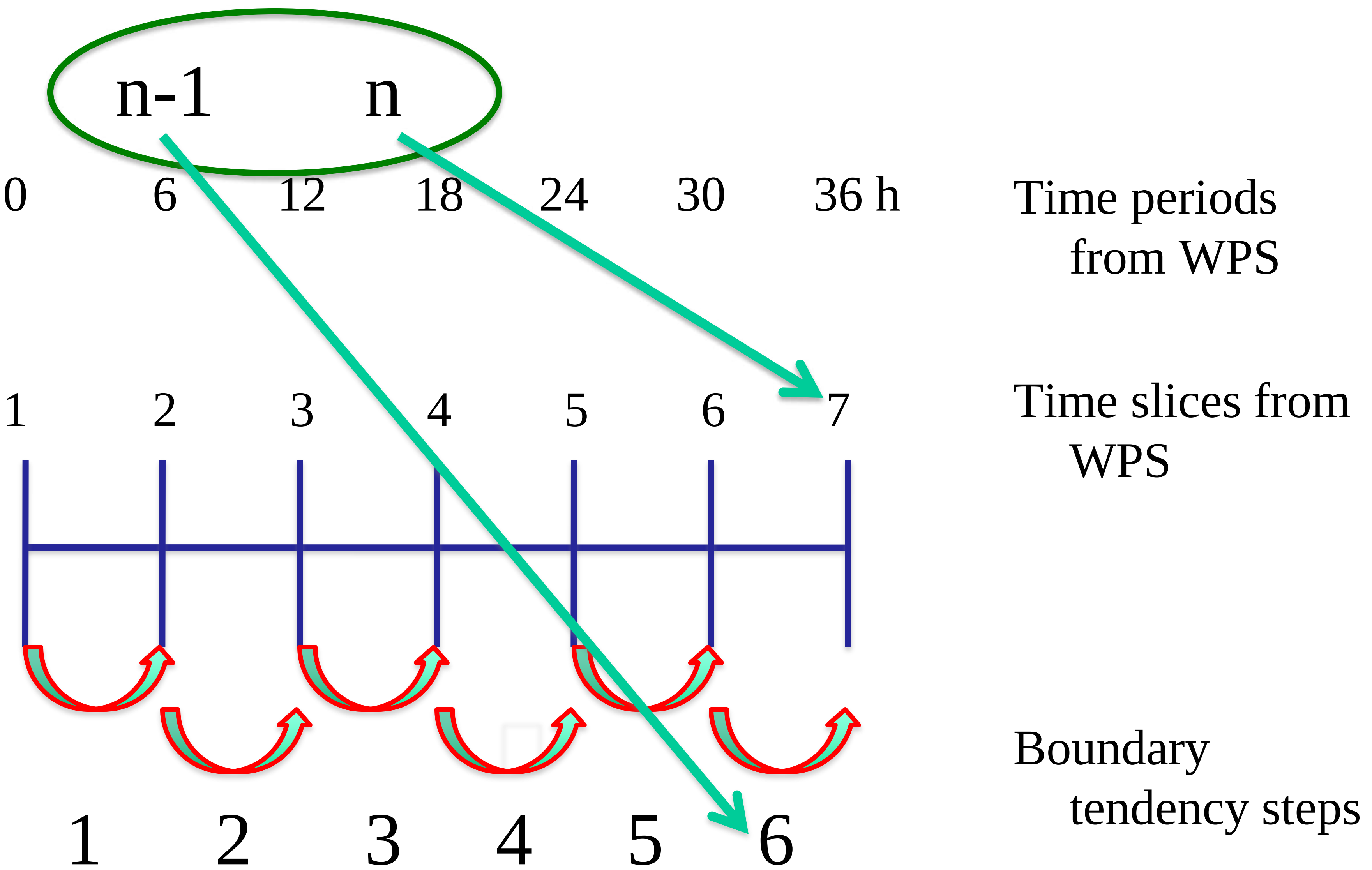
Standard Generated Output

- For regional forecasts, the real program generates both an both an initial (**wrfinput_d01**) and a lateral boundary (**wrfbdy_d01**)
- The initial condition file contains a single time period of data
- These files contain data used directly by the WRF model
- If **n** times were processed with WPS and real, the lateral boundary file contains **n-1** time slices

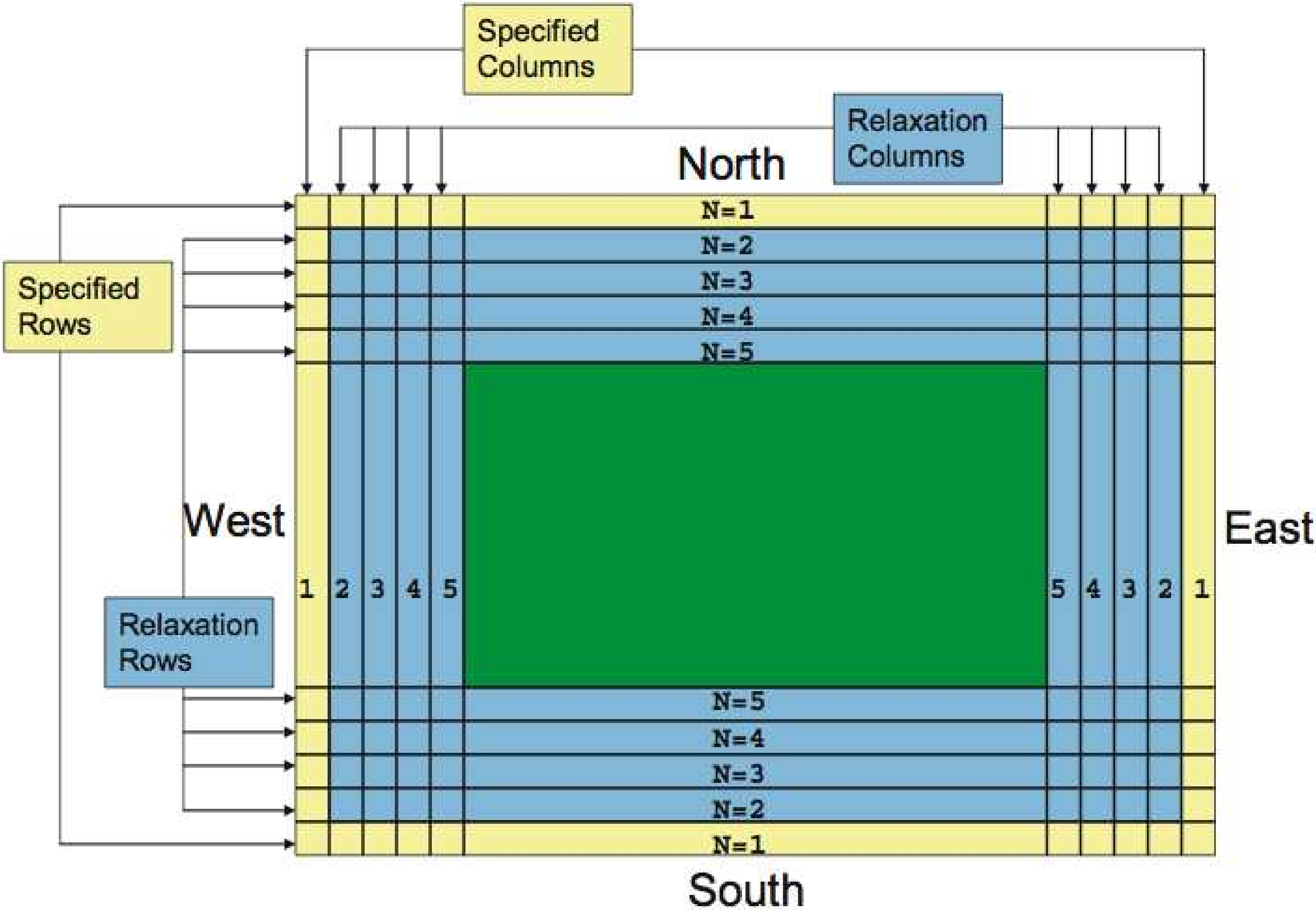
Lateral Boundary Condition Times



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

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

User specifies the selected η surfaces in the namelist (or can be computed)

Vertically interpolate input fields in pressure to the η surfaces in dry pressure: default all variables log

Soil Level Interpolation

Noah

Layers Mid point

000 – 010 cm -- 005 cm

010 – 040 cm -- 025 cm

040 – 100 cm -- 070 cm

100 – 200 cm – 150 cm

RUC

Levels

000 cm

005 cm

020 cm

040 cm

160 cm

300 cm

Soil Level Interpolation

- The real program accepts soil temperature and moisture 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
- Orientation:
  - the “first” level is near or at the model surface
  - the “last” level is the deepest of the soil information

Water Temperature Initialization

- Two general 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

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

The Advanced Research WRF (ARW) Dynamics Solver

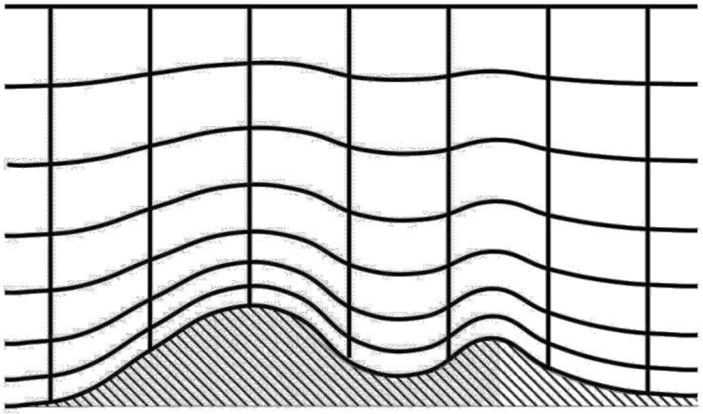
Bill Skamarock

Jimy Dudhia

ARW Dynamical Solver

- Vertical coordinate
- Equations / variables
- Time integration scheme
- Grid staggering
- Advection scheme
- Time step parameters
- Filters
- Boundary conditions
- Nesting
- Map projections

Vertical Coordinate and Prognostic Variables



Hydrostatic pressure π

Column mass $\mu = \pi_s - \pi_t$

Vertical coordinate $\eta = \dfrac{(\pi - \pi_t)}{\mu}$

Layer mass $\mu \Delta \eta = \Delta \pi = g \rho \Delta z$

Conserved state (prognostic) variables:

$\mu, \quad \mathrm{U} = \mu \mathrm{u}, \quad \mathrm{V} = \mu \mathrm{v}, \quad \mathrm{W} = \mu \mathrm{w}, \quad \Theta = \mu \theta$

Non-conserved state variable: $\phi = \mathrm{gz}$

Time Integration in ARW

3rd Order Runge-Kutta time integration

advance $\phi^t \rightarrow \phi^{t+\Delta t}$

$$\phi^{*} = \phi^t + \frac{\Delta t}{3} R\left(\phi^t\right)$$

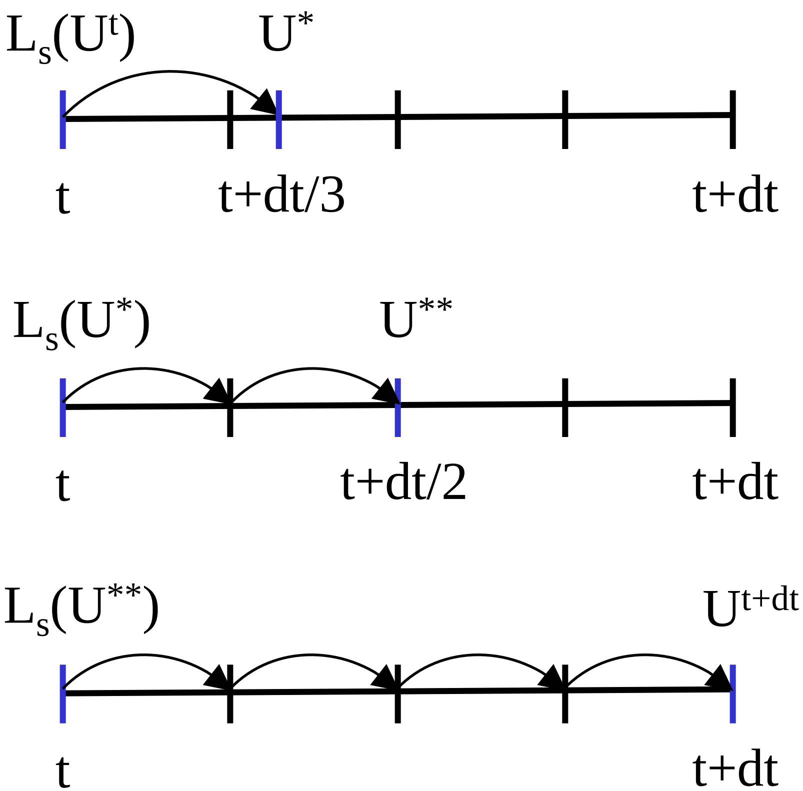
$$\phi^{**} = \phi^t + \frac{\Delta t}{2} R\left(\phi^{*}\right)$$

$$\phi^{t+\Delta t} = \phi^t + \Delta t\, R\left(\phi^{**}\right)$$

Time-Split Runge-Kutta Integration Scheme

$$U_t = L_{fast}(U) + L_{slow}(U)$$

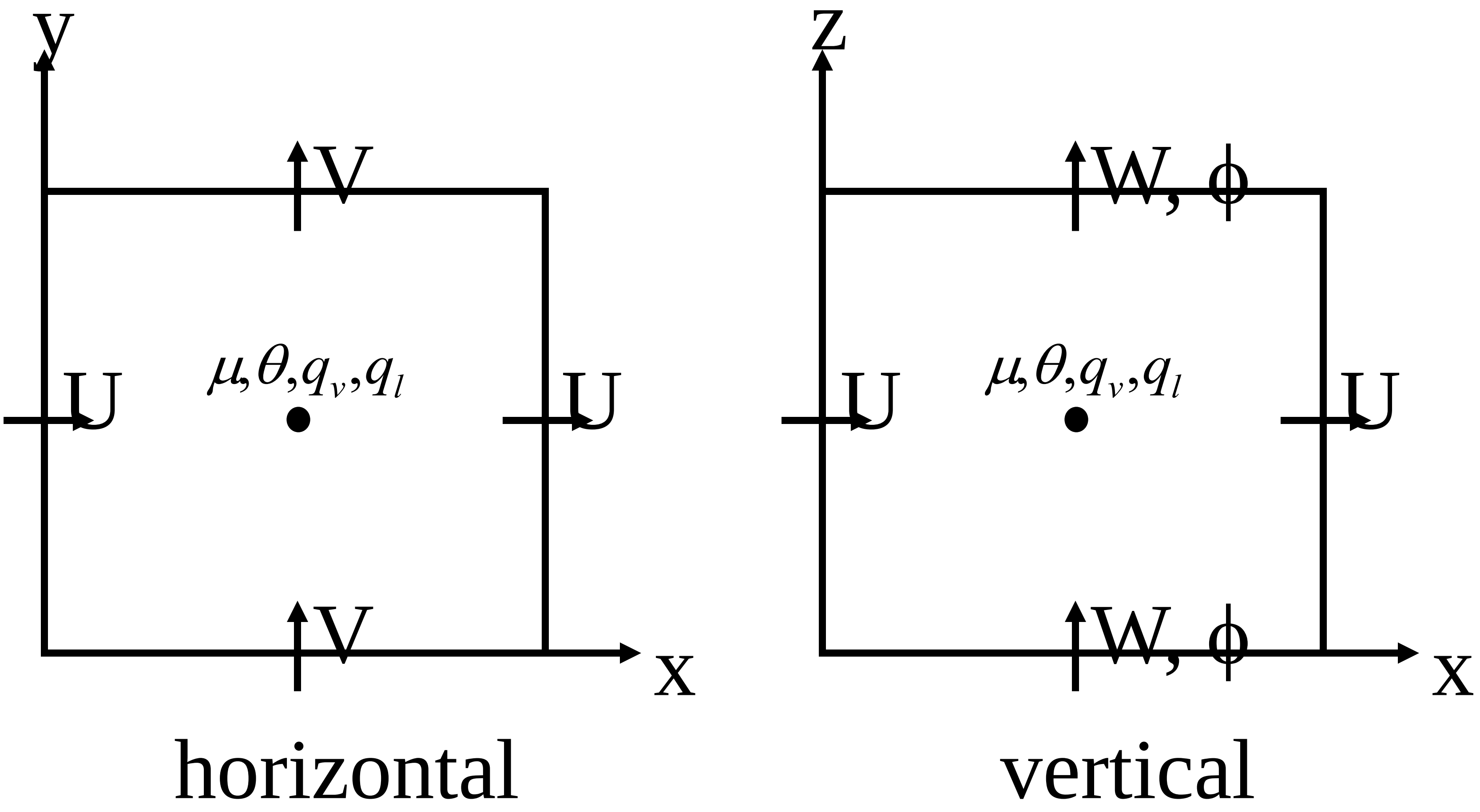
3rd order Runge-Kutta, 3 steps



- RK3 is 3rd order accurate for linear eqns, 2nd order accurate for nonlinear eqns.
- Stable for centered and upwind advection schemes.
- Stable for Courant number U*dt/dx < 1.73
- Three $L_{slow}(U)$ evaluations per timestep.

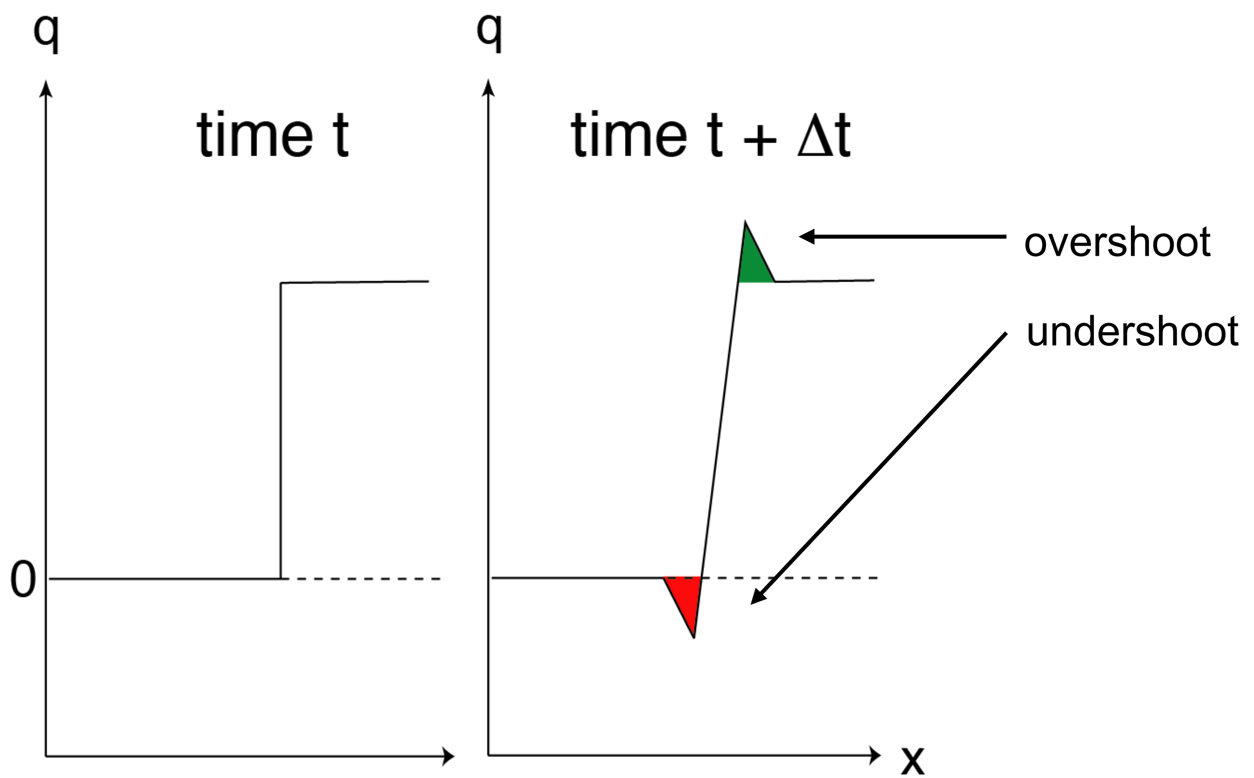
ARW model, grid staggering

C-grid staggering

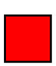
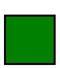


Moisture Transport in ARW

1D advection

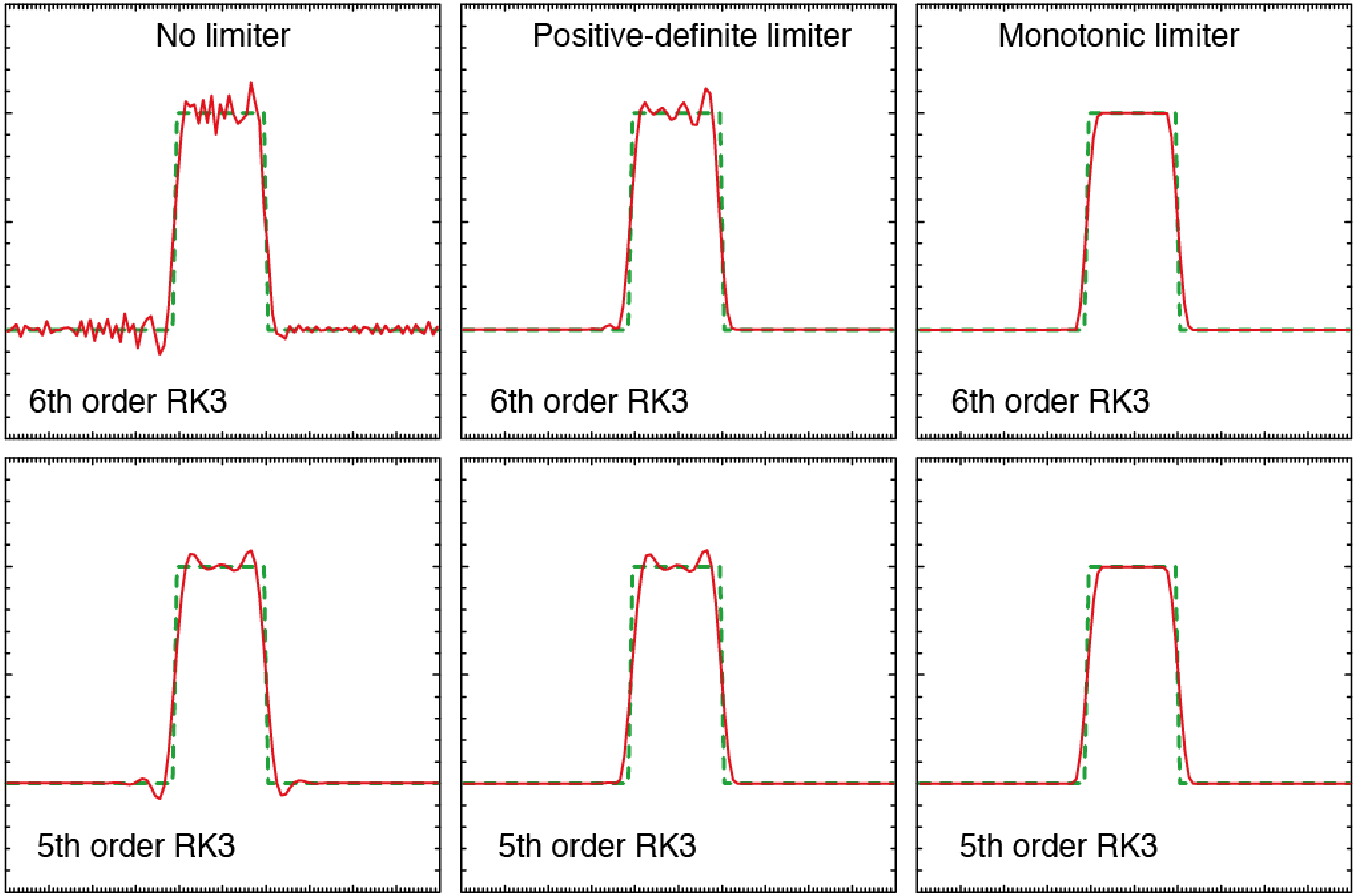


ARW scheme is conservative,
but not positive definite nor monotonic.

Removal of negative q  results in spurious source of q .

PD/Monotonic Limiters in ARW - 1D Example
Top-Hat Advection

1D Top-hat transport Cr = 0.5, 1 revolution, 200 steps



ARW Model: Dynamics Parameters

Guidelines for time step

Δt in seconds should be about 6*Δx (grid size in kilometers). Larger Δt can be used in smaller-scale dry situations, but time_step_sound (default = 4) should increase proportionately if larger Δt is used.

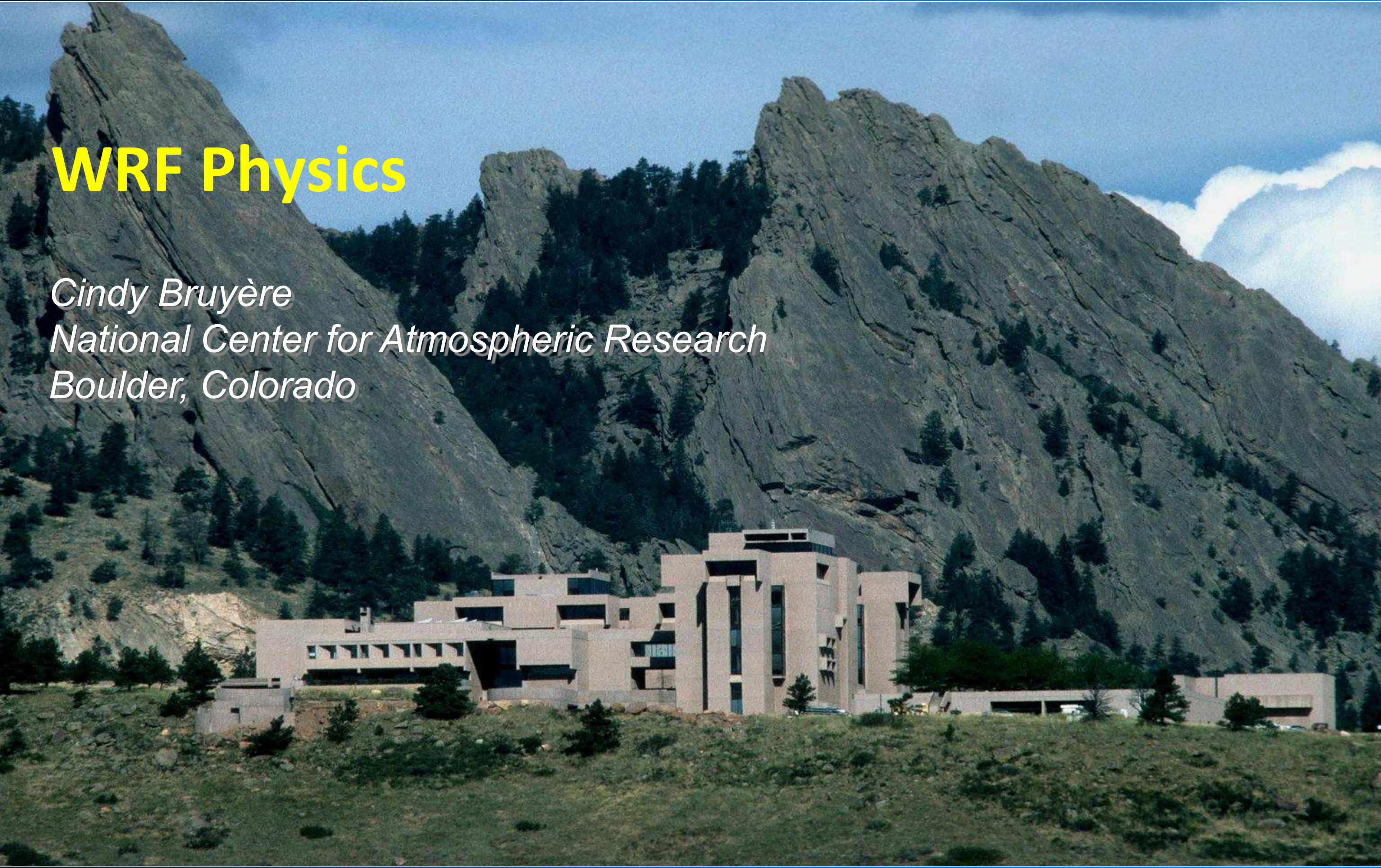
WRF ARW Model Integration Procedure

Begin time step

- Runge-Kutta loop (steps 1, 2, and 3)
  - (i) advection, p-grad, buoyancy, physics
  - Acoustic step loop
    - (i) advance U,V
    - (ii) advance μ, Θ□□□□
    - (iii) advance w, ϕ□□
  - End acoustic loop
  - Advance scalars
- End Runge-Kutta loop
- Adjustment physics (currently microphysics)

End time step

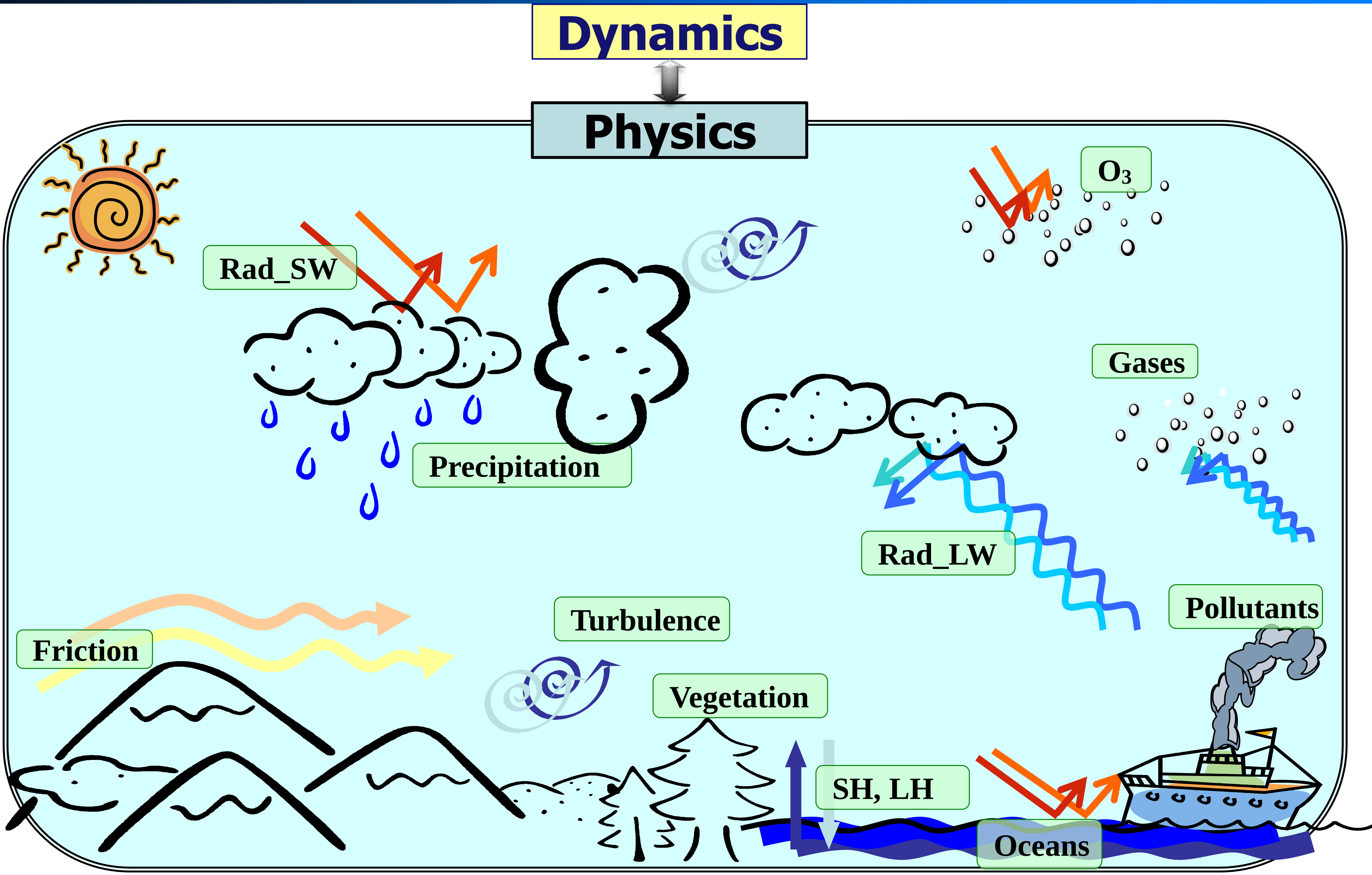
NCAR is Sponsored by NSF and this work is partially supported by the Willis Research Network and the Research Partnership to Secure Energy for America



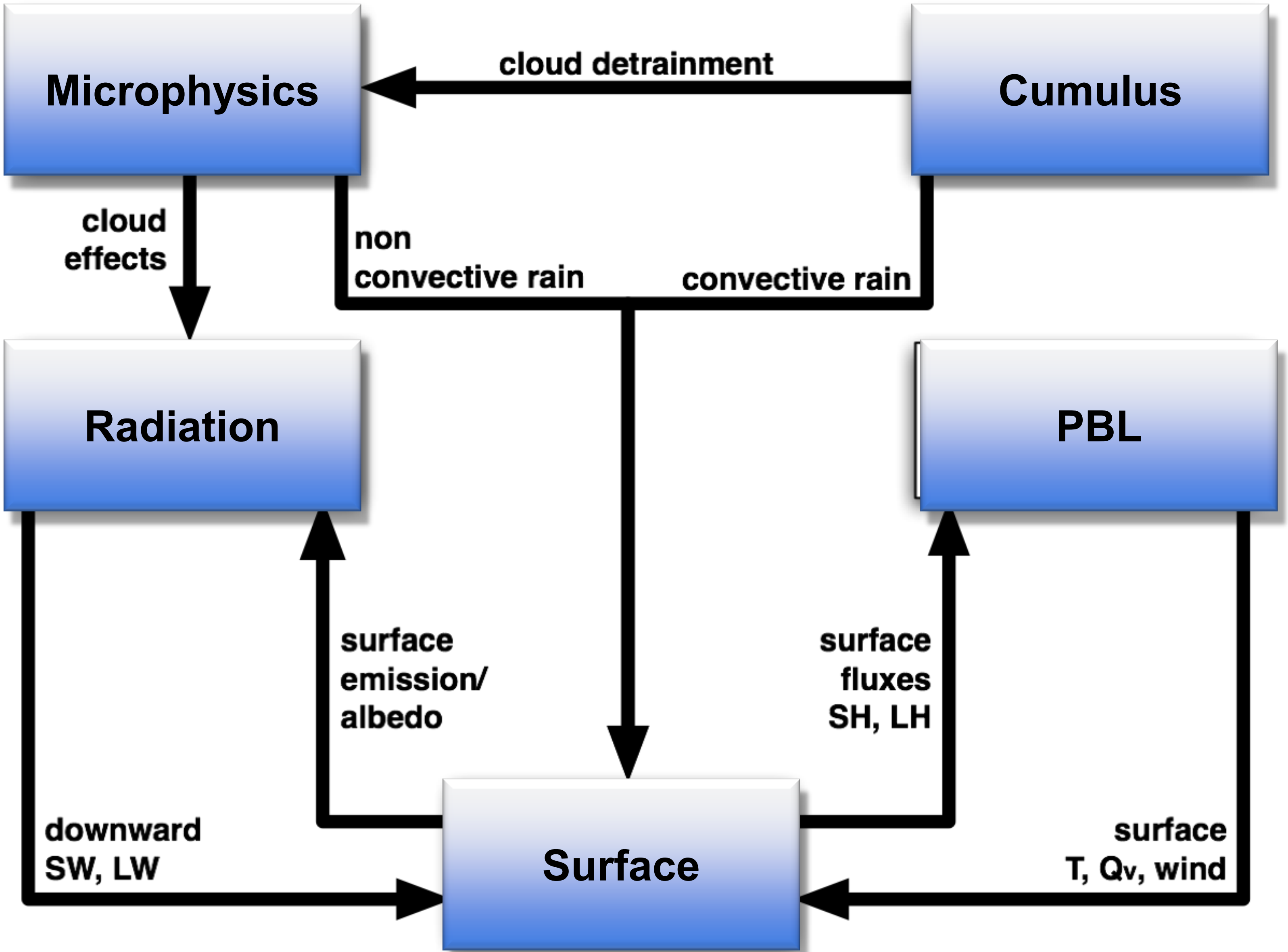
WRF Physics

Cindy Bruyère
National Center for Atmospheric Research
Boulder, Colorado

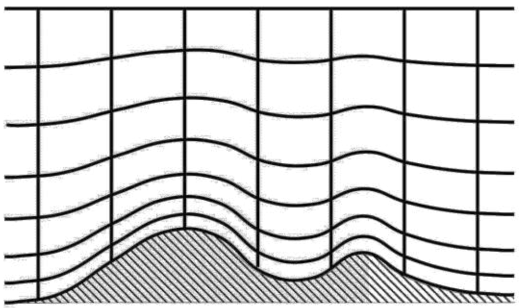
Introduction

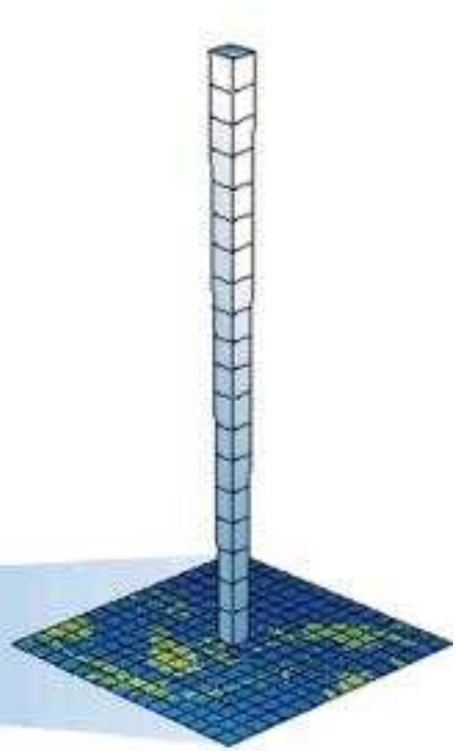


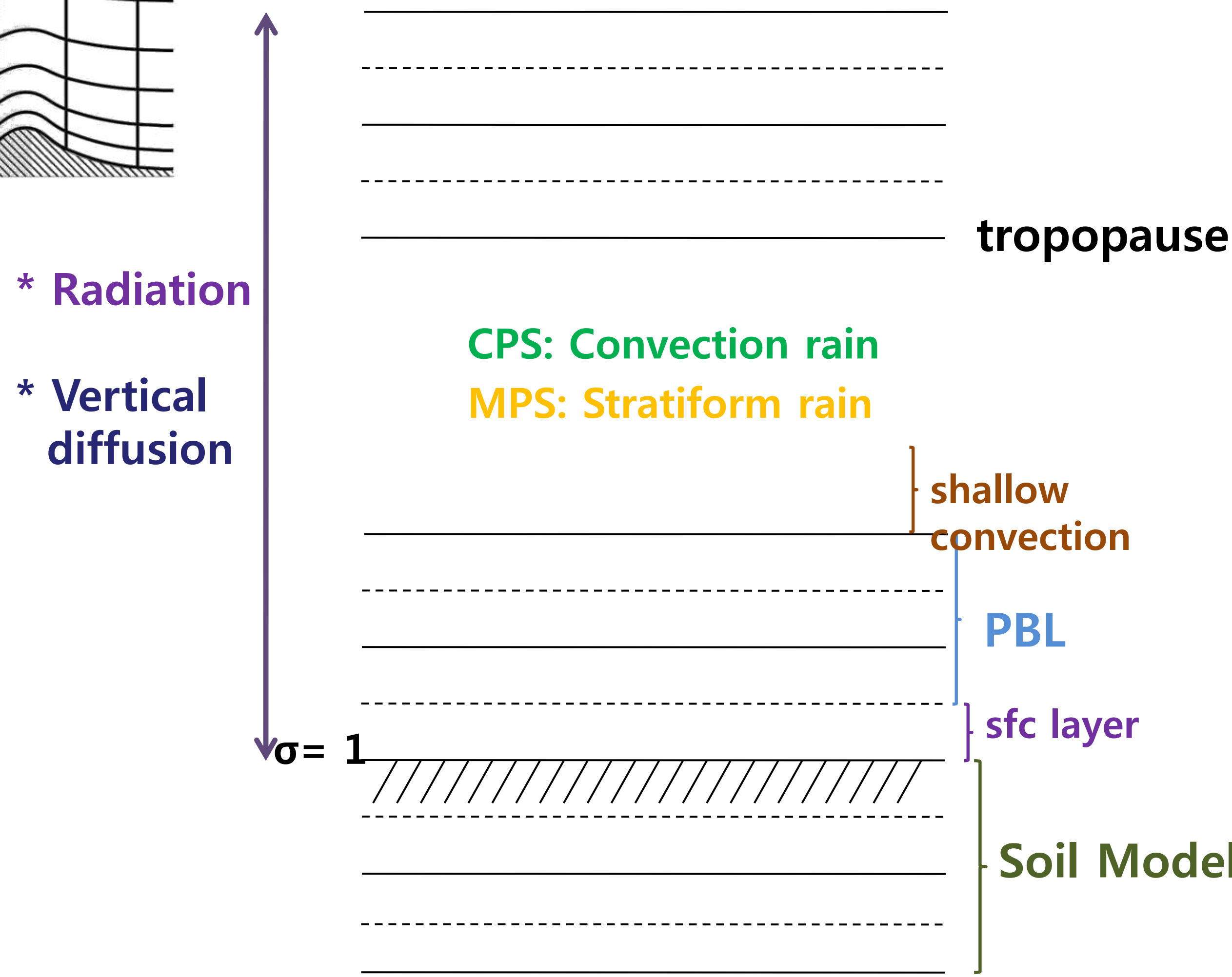
Interactions



Vertical

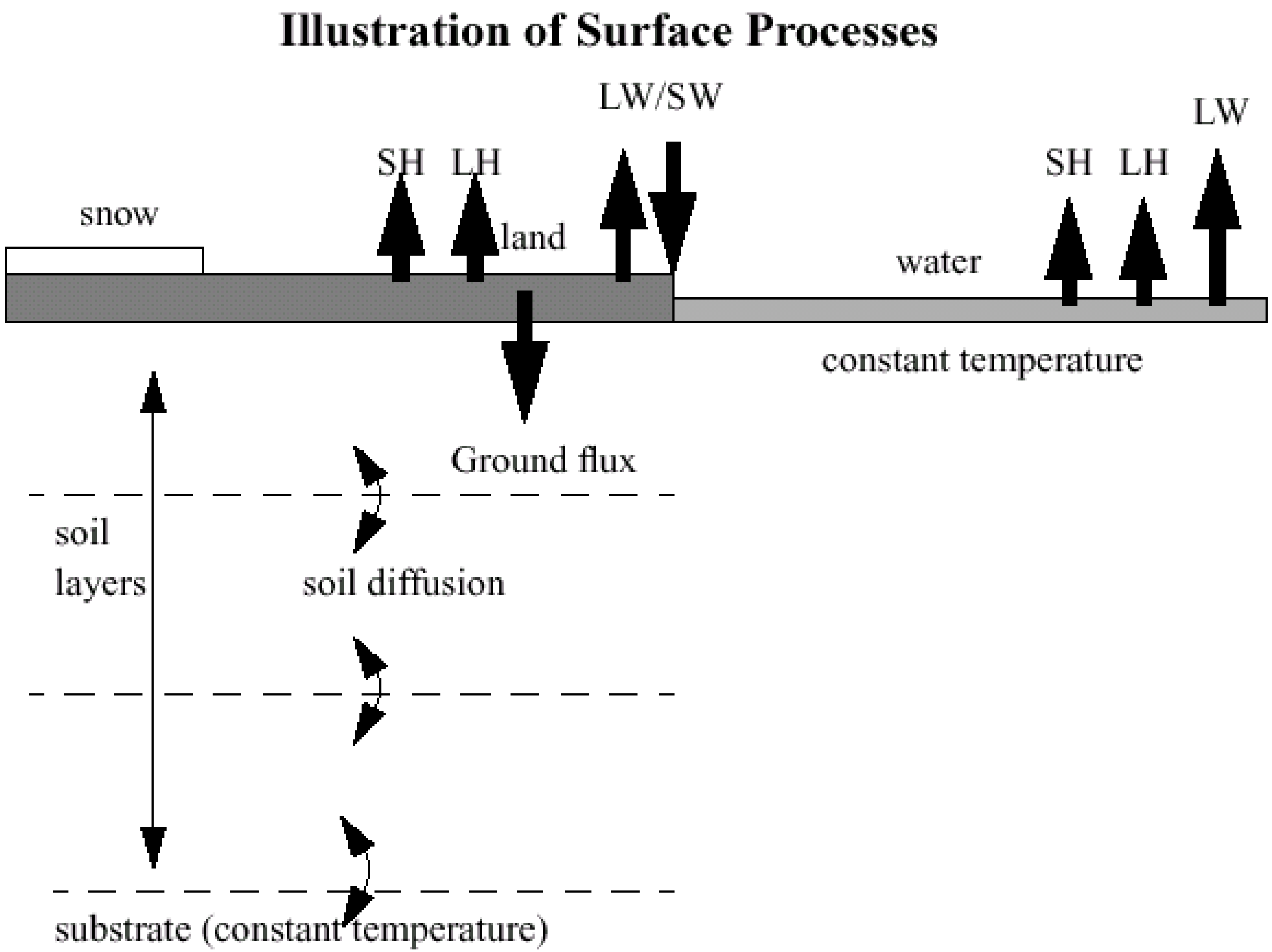






* Radiation
* Vertical diffusion
tropopause
CPS: Convection rain
MPS: Stratiform rain
shallow convection
PBL
sfc layer
σ= 1
Soil Model

Atmosphere – Surface Interaction



WRF Physics

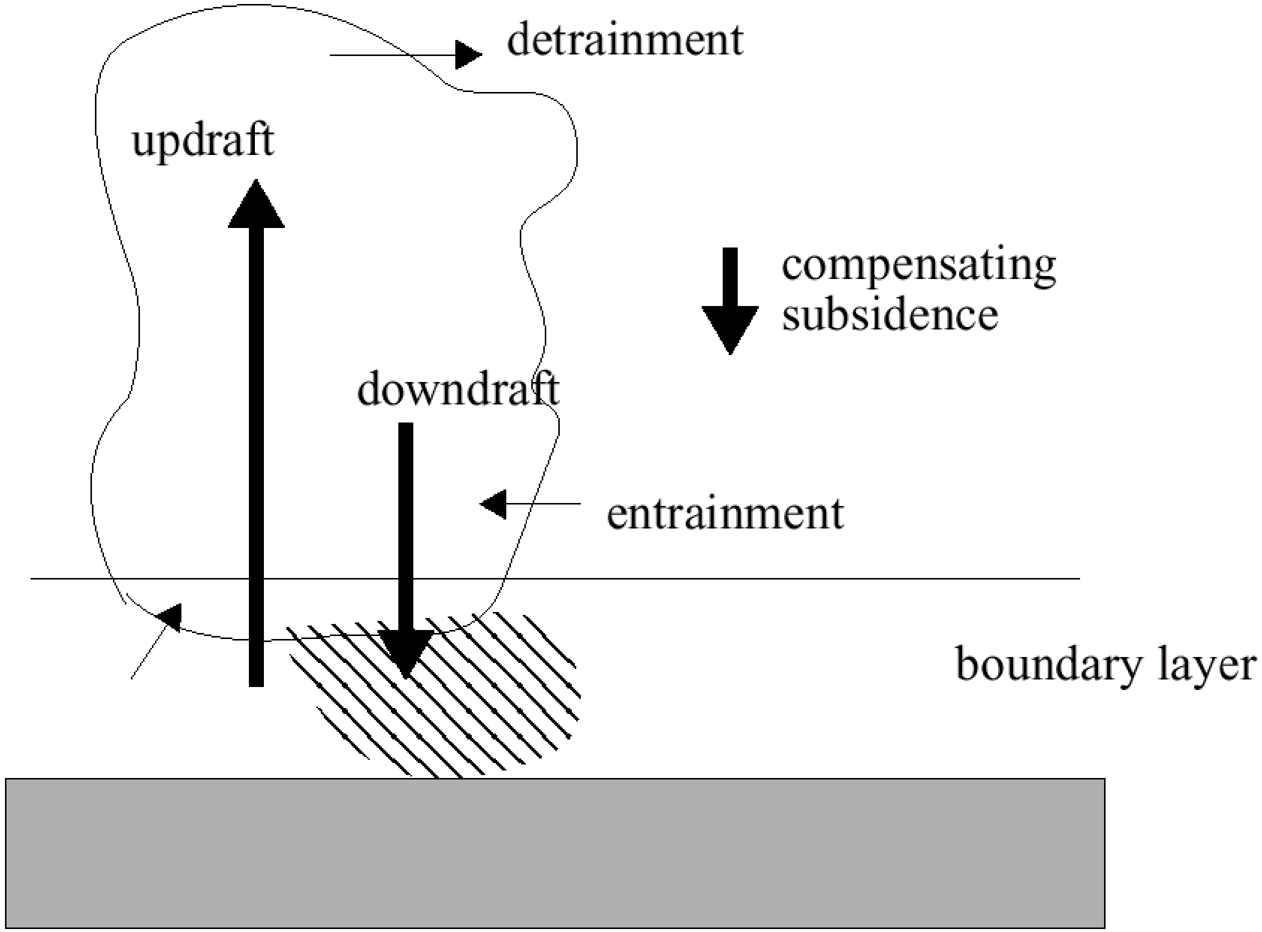
- Cumulus parameterization (cu_physics)
- Microphysics (mp_physics)
- Radiation
  - Longwave (ra_lw_physics)
  - Shortwave (ra_sw_physics)
- PBL (bl_pbl_physics)
- Surface
  - Surface layer (sf_sfclay_physics)
  - Land/water surface (sf_surface_physics)

Cumulus Parameterization

Provides

- Atmospheric heat and moisture/cloud tendency profiles
- Surface sub-grid-scale (convective) rainfall

Cumulus Processes



- Schemes work in individual columns that are considered convectively unstable

Cumulus Schemes

- Use for grid columns that completely contain convective clouds
- Re-distribute air in column to account for vertical convective fluxes
  - Updrafts take boundary layer air upwards
  - Downdrafts take mid-level air downwards
- Some schemes have non-precipitation shallow convection extensions
- Schemes have to determine
  - When to trigger a convective column
  - How fast to make the convection act

Triggers

- Clouds only activate in columns that meet certain criteria
  - Presence of some convective available potential energy (CAPE) in sounding
  - Not too much convective inhibition (CIN) in sounding (cap strength)
  - Minimum cloud depth from parcel ascent

Cumulus schemes in WRF

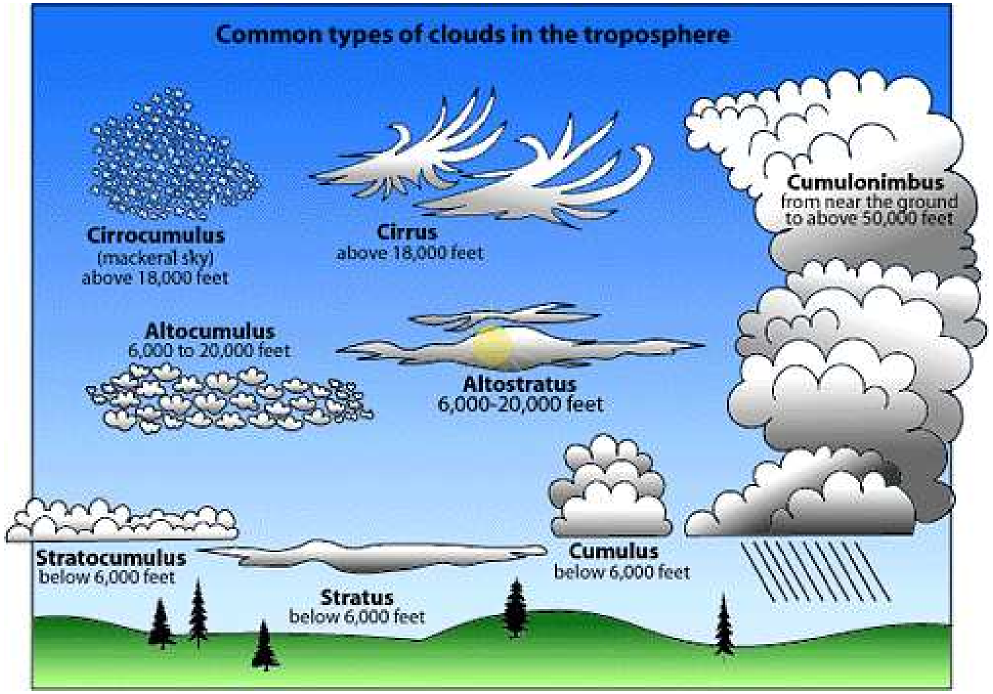
cu_physics	Scheme	Moisture Tendencies	Momentum Tendencies	Shallow Convection
1	Kain-Fritsch Eta	Qc Qr Qi Qs	no	yes
2	Betts-Miller-Janjic	-	no	yes
3	Grell-Devenyi	Qc Qi	no	no
4	Old Simplified Arakawa-Schubert	Qc Qi	no	yes
5	Grell-3	Qc Qi	no	yes
6	Tiedtke	Qc Qi	yes	yes
7	Zhang-McFarlane	Qc Qi	yes	no
14	New SAS	Qc Qi	yes	yes
84	New SAS (HWRF)	Qc Qi	no	yes
99	Old Kain-Fritsch	Qc Qr Qi Qs	no	no

Cumulus scheme

Recommendations about use

- For dx ≥ 10 km: probably need cumulus scheme
- For dx ≤ 3 km: probably do not need scheme
  - However, there are cases where the earlier triggering of convection by cumulus schemes help
- For dx=3-10 km, scale separation is a question
  - Few schemes are specifically designed with this range of scales in mind
- Issues with 2-way nesting when physics differs across nest boundaries (seen in precipitation field on parent domain)
  - best to use same physics in both domains or 1-way nesting

Microphysics

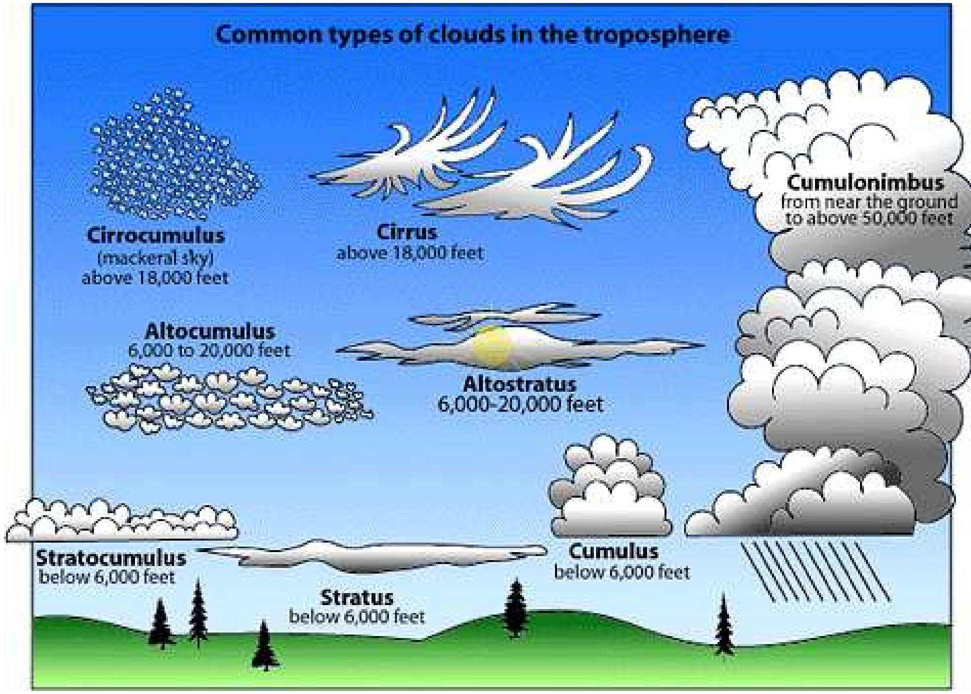


Provides

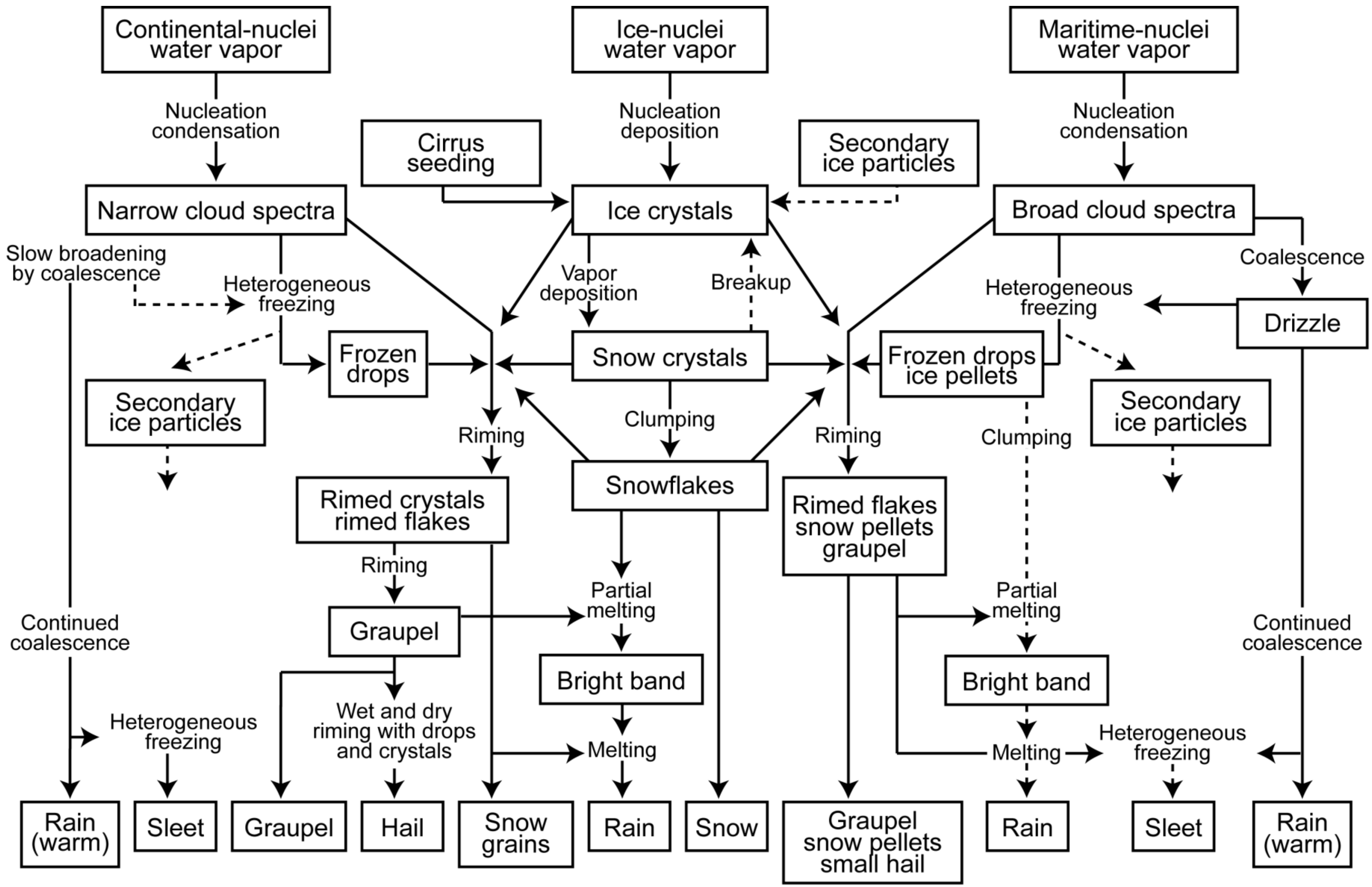
- Atmospheric heat and moisture tendencies
- Microphysical rates
- Surface resolved-scale rainfall

Resolved clouds

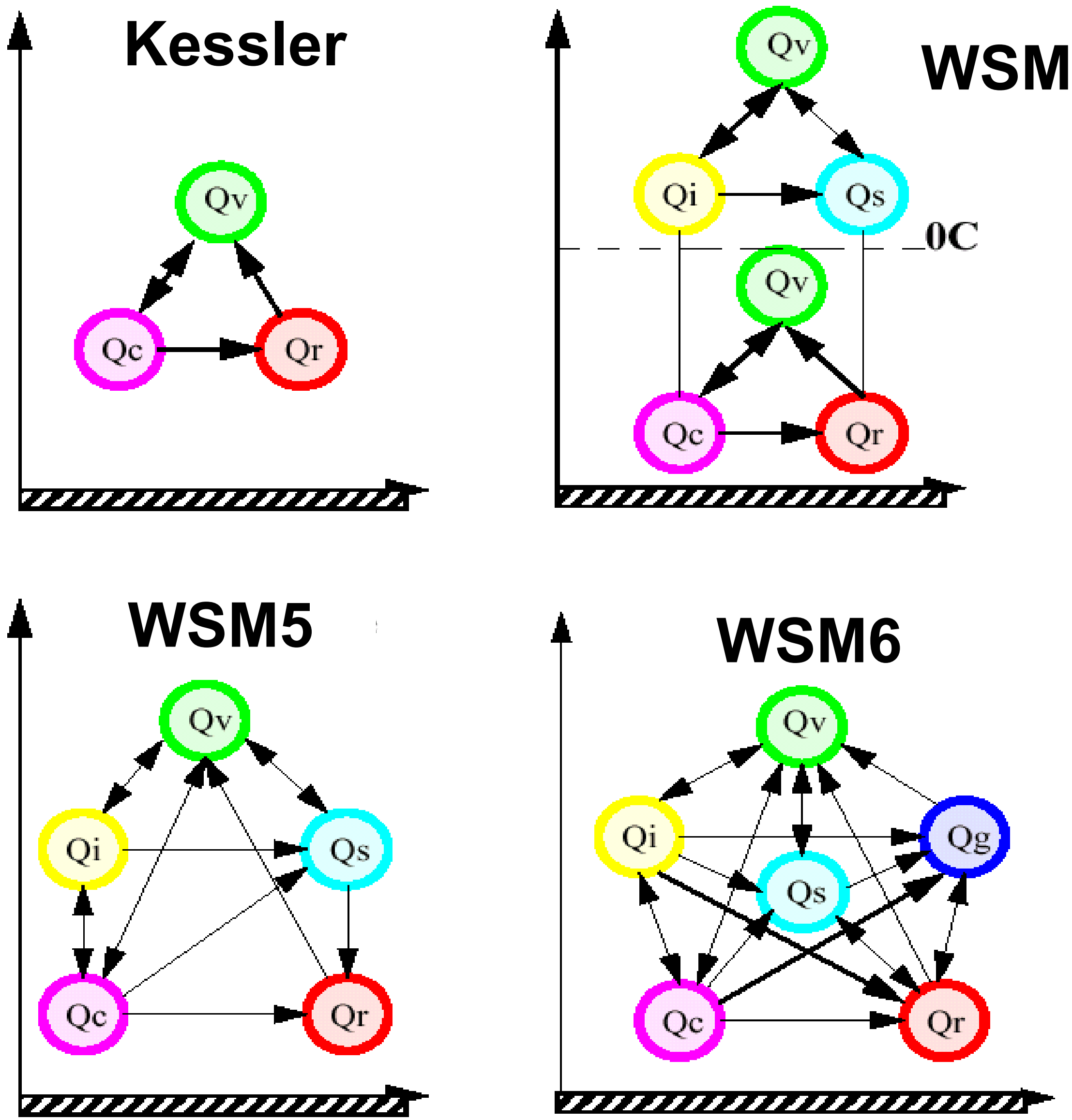
- Formed by radiative, dynamical or convective processes
- Model only considers grid-scale average so will not resolve fine-scale structures



Microphysics Parameterization



Microphysical Interactions in WRF



WRF Microphysics Options

- Range of levels of sophistication
  - Warm rain (i.e. no ice) – Kessler (idealized)
  - Simple ice (3 arrays) – WSM3
  - Mesoscale (5 arrays, no graupel) – WSM5
  - Cloud-scale single-moment (6 arrays, graupel) – WSM6, Lin, Goddard, SBU, Eta-Ferrier
  - Double-moment (8-13 arrays) – Thompson, Morrison, Milbrandt-Yau, WDM5, WDM6

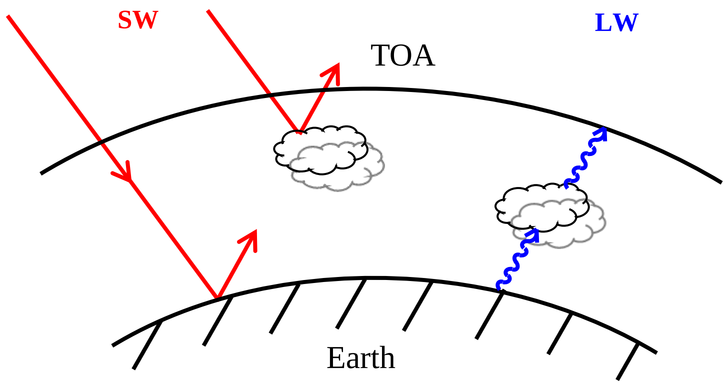
Microphysics

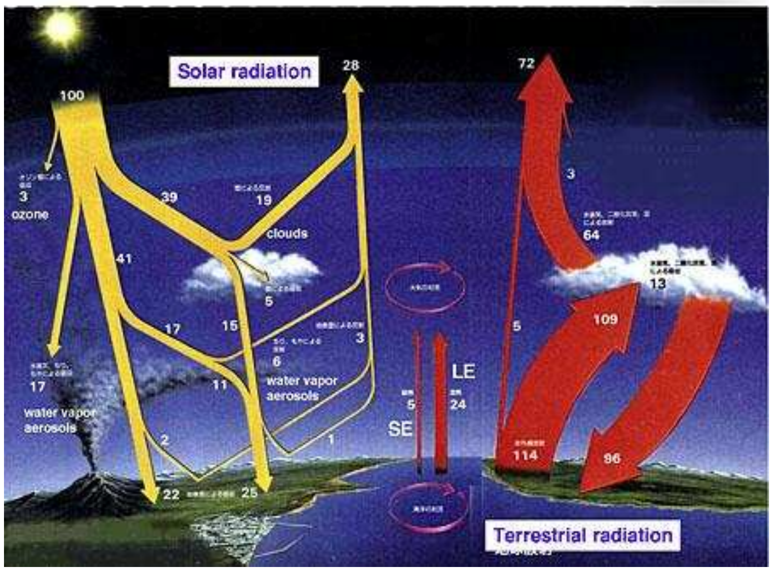
- Latent heat release from
  - Condensation, evaporation, deposition, sublimation, freezing, melting
- Particle types
  - Cloud water, rain drops, ice crystals, snow, graupel (also hail in some)
    - Total mass contributes to liquid loading in dynamics
- Processes
  - Aggregation, accretion, growth, fall-out

Microphysics Schemes in WRF

1	Kessler	Qc Qr	
2	Lin (Purdue)	Qc Qr Qi Qs Qg	
3	WSM3	Qc Qr	
4	WSM5	Qc Qr Qi Qs	
5	Eta (Ferrier)	Qc Qr Qs (Qt*)	
6	WSM6	Qc Qr Qi Qs Qg	
7	Goddard	Qc Qr Qi Qs Qg	
8	Thompson	Qc Qr Qi Qs Qg	Ni Nr
9	Milbrandt 2-mom	Qc Qr Qi Qs Qg Qh	Nc Nr Ni Ns Ng Nh
10	Morrison 2-mom	Qc Qr Qi Qs Qg	Nr Ni Ns Ng
13	SBU-YLin	Qc Qr Qi Qs	
14	WDM5	Qc Qr Qi Qs	Nn Nc Nr
16	WDM6	Qc Qr Qi Qs Qg	Nn Nc Nr
17	NSSL 2-mom	Qc Qr Qi Qs Qg Qh	Nc Nr Ni Ns Ng Nh
18	NSSL 2-mom+ccn	Qc Qr Qi Qs Qg Qh	Nc Nr Ni Ns Ng Nh Nn

Radiation

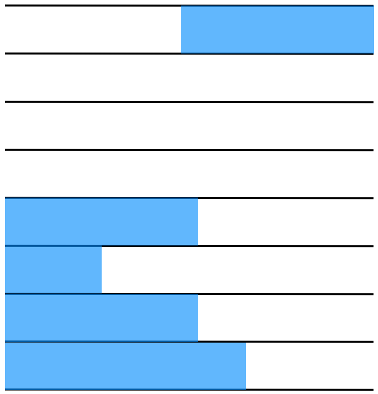




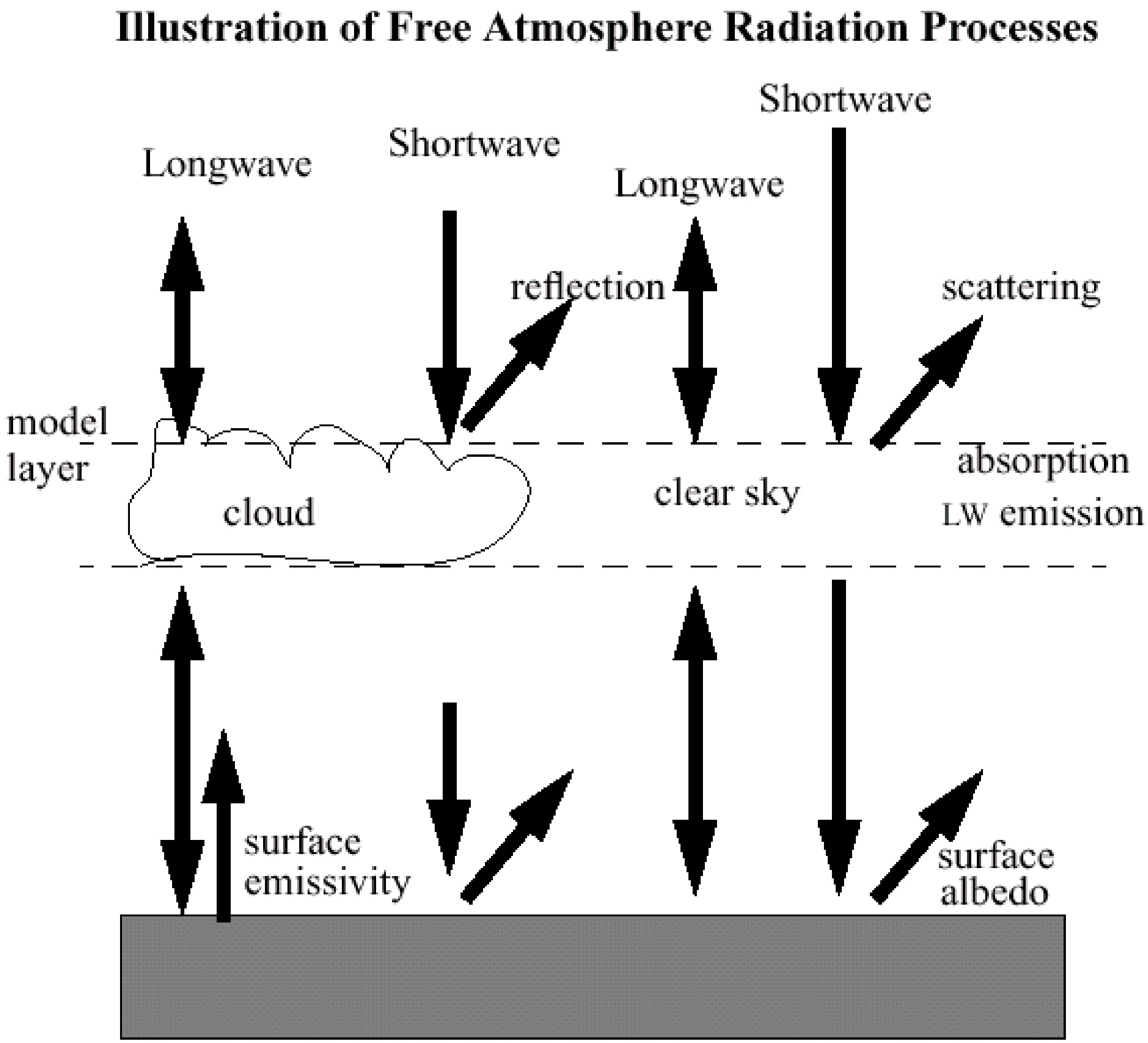
Provides

- Atmospheric temperature tendency profile
- Surface radiative fluxes

Radiation Processes



Maximum Random Overlap



Longwave Radiation Schemes

ra_lw_physics	Scheme	Microphysics Interaction	Cloud Fraction	CO2 *(current 392)*
1	RRTM	Qc Qr Qi Qs Qg	1/0	330
3	CAM	Qc Qi Qs	Max-rand overlap	yearly (A2)
4	RRTMG	Qc Qr Qi Qs	Max-rand overlap	379
5	New Goddard	Qc Qr Qi Qs Qg	1/0	337
7	FLG (UCLA)	Qc Qr Qi Qs Qg	1/0	345
31	Held-Suarez	none	none	none
99	GFDL	Qc Qr Qi Qs	Max-rand overlap	fixed

- CO2 – well-mixed, specified constant in whole atmosphere (CAM has year-dependent table of values - *hardcoded*)

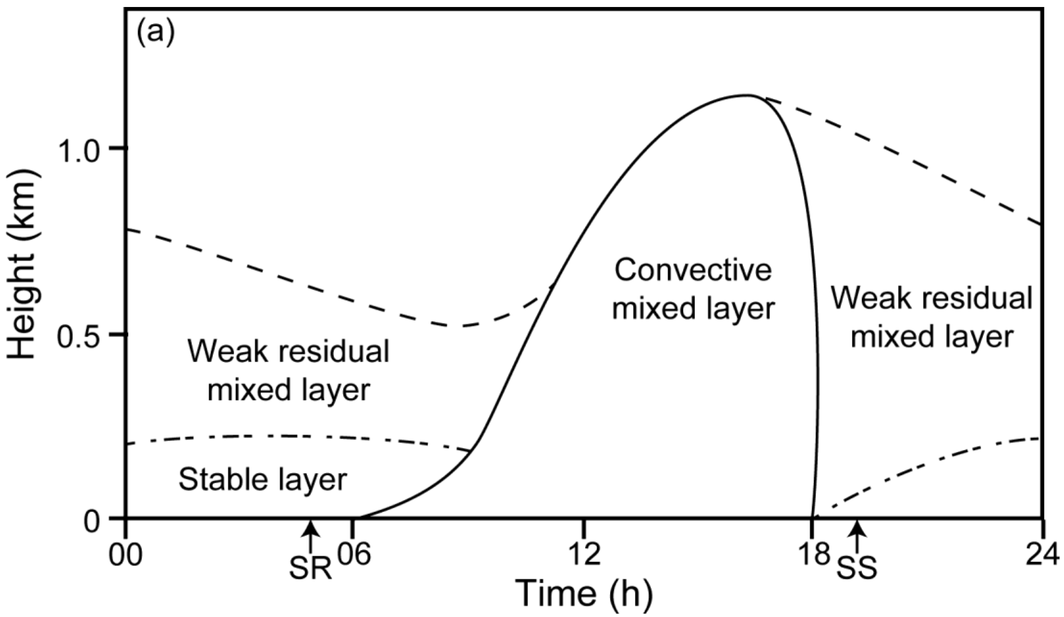
Shortwave Radiation Schemes

ra_lw_physics	Scheme	Microphysics Interaction	Cloud Fraction	Ozone
1	Dudhia	Qc Qr Qi Qs Qg	1/0	none
2	GSFC	Qc Qi	1/0	5 profiles
3	CAM	Qc Qi Qs	Max-rand overlap	Lat/month
4	RRTMG	Qc Qr Qi Qs	Max-rand overlap	1 profile
5	New Goddard	Qc Qr Qi Qs Qg	1/0	5 profiles
7	FLG (UCLA)	Qc Qr Qi Qs Qg	1/0	5 profiles
99	GFDL	Qc Qr Qi Qs	Max-rand overlap	Lat/date

Shortwave Radiation Schemes

- Ozone *(same as for LW Radiation)*
  - Ozone heating maintains warm stratosphere
  - Important for model tops above about 20 km (50 hPa)
  - CAM has monthly, zonal, pressure-level data
  - Others use single profiles
- SW radiation is an important component of surface energy balance
- Surface albedo reflection based on land-surface type and snow cover

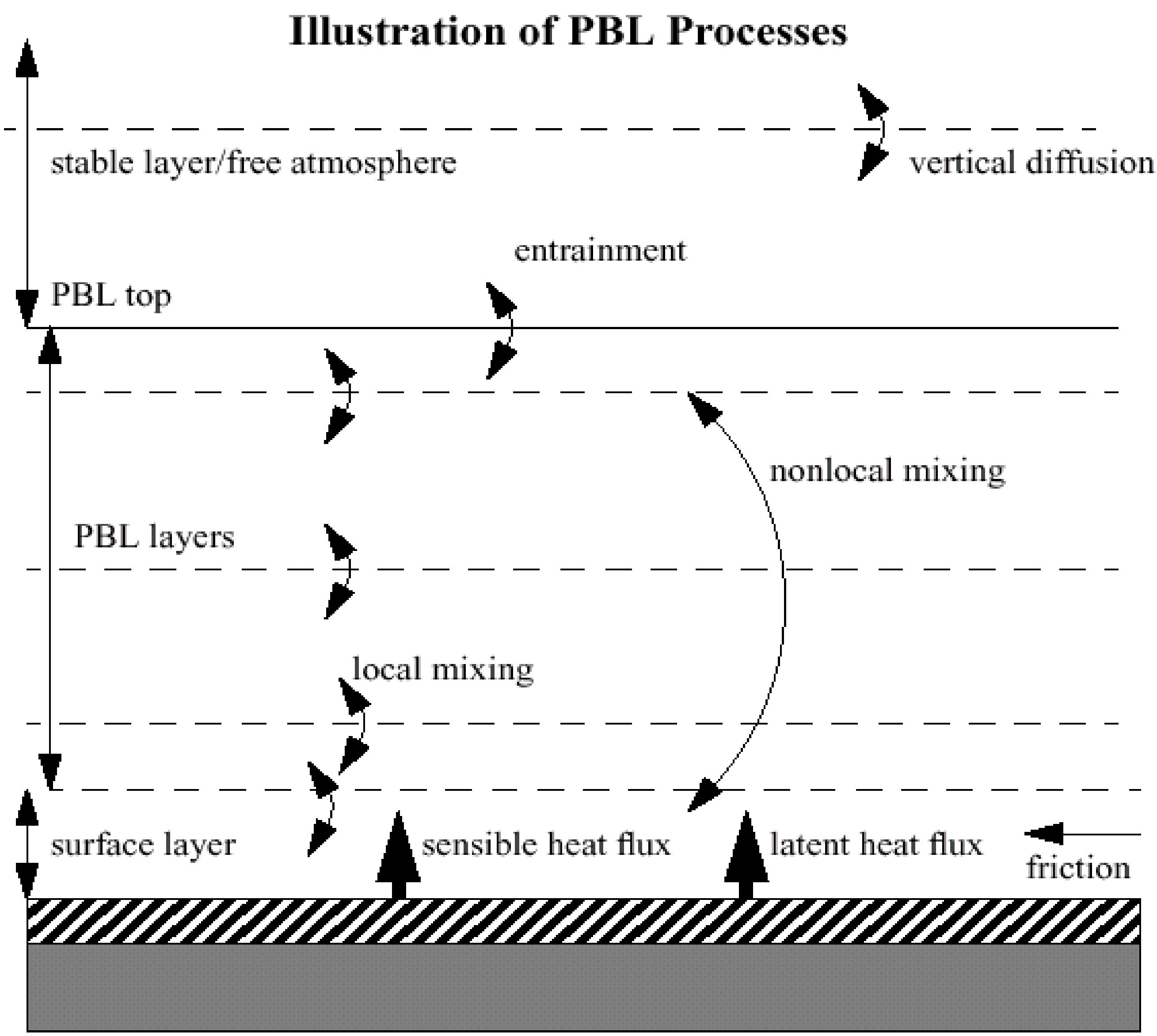
Planetary Boundary Layer



Provides

- Boundary layer fluxes (heat, moisture, momentum)
- Vertical diffusion in whole column

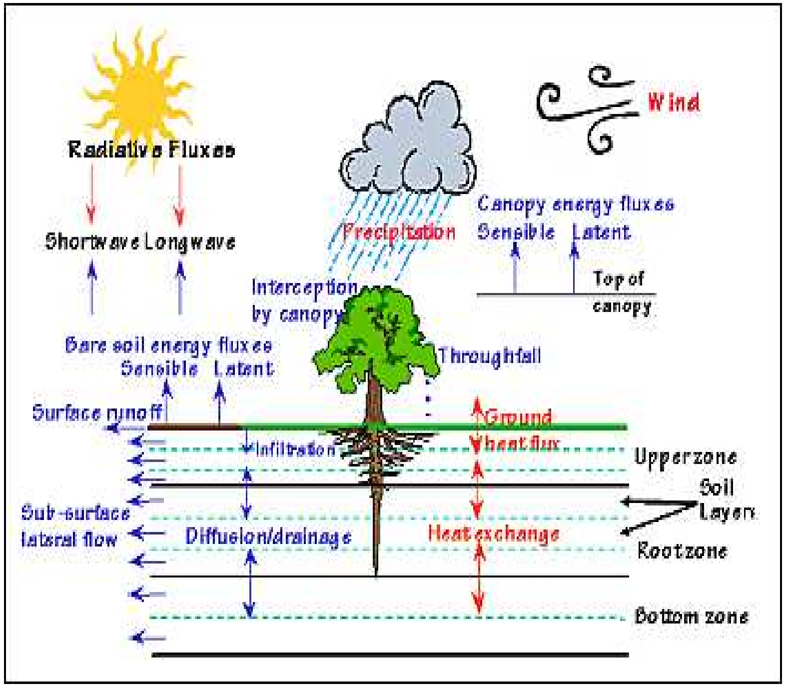
PBL Processes



WRF PBL Options

- Purpose is to distribute surface fluxes with boundary layer eddy fluxes and allow for PBL growth by entrainment
- Classes of PBL scheme
  - Turbulent kinetic energy prediction (Mellor-Yamada Janjic, MYNN, Bougeault-Lacarrere, TEMF, QNSE, CAM UW)
  - Diagnostic non-local (YSU, GFS, MRF, ACM2)
- Above PBL all these schemes also do vertical diffusion due to turbulence

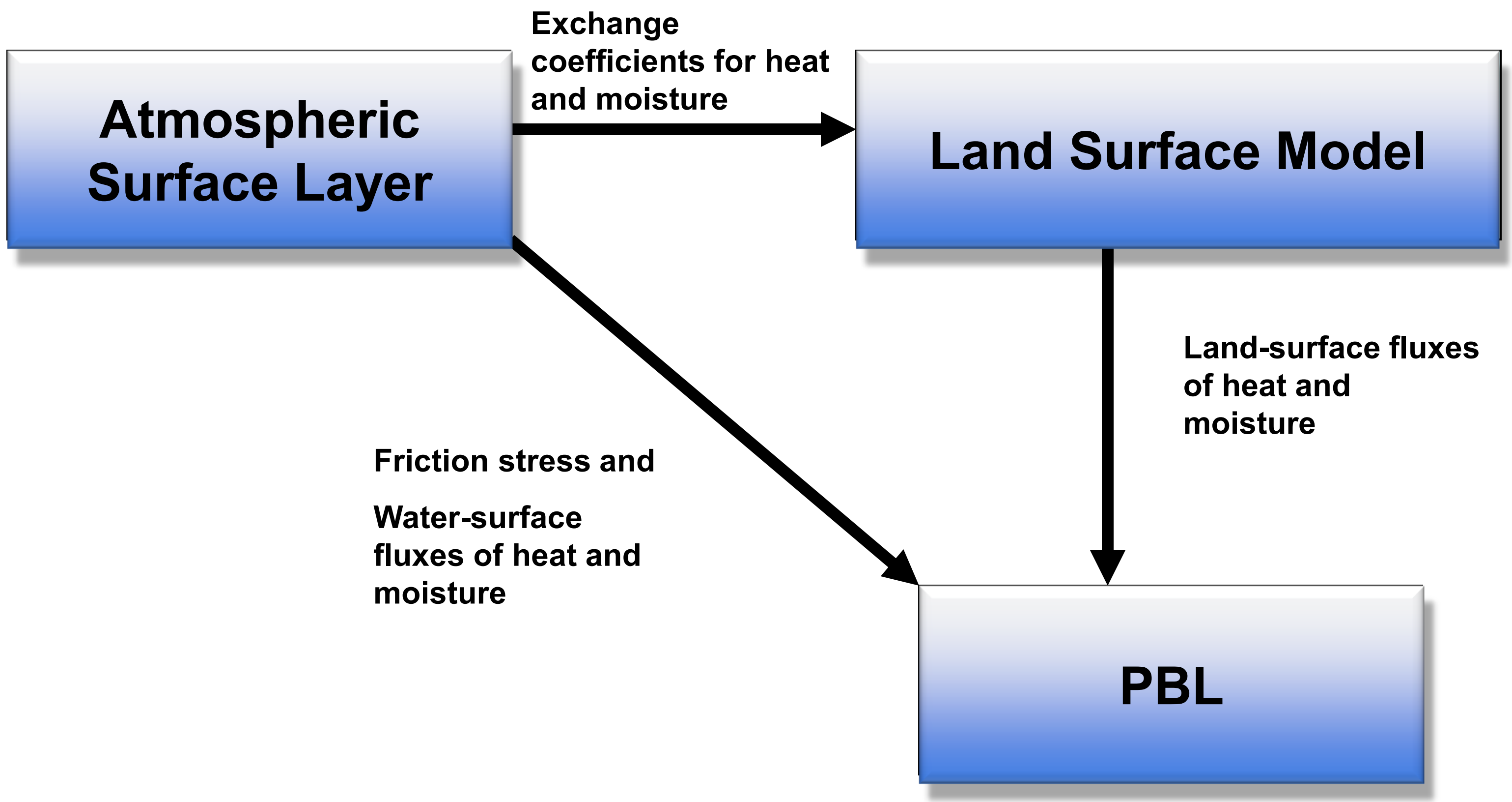
Surface schemes



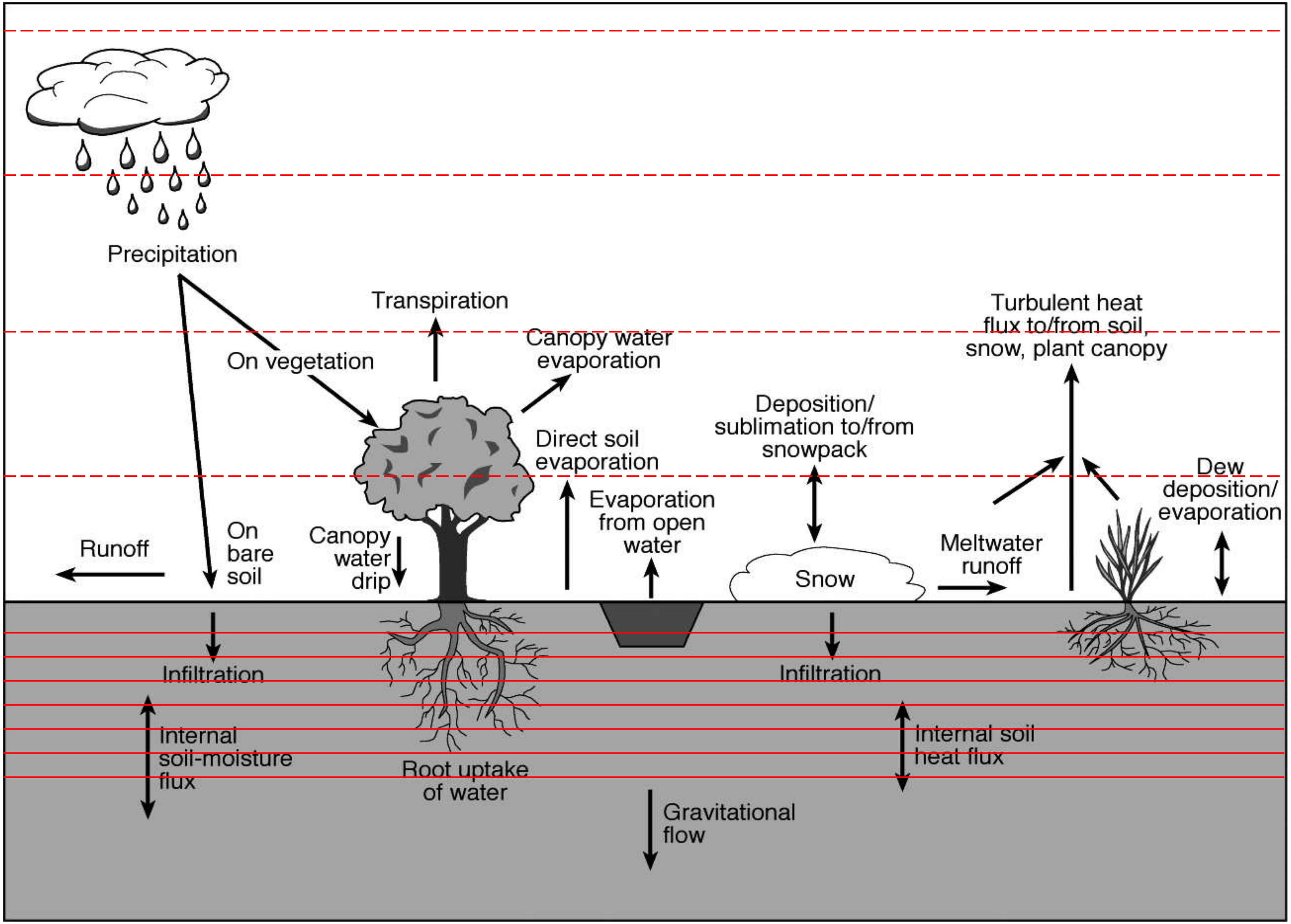
Provides

- Surface layer of atmosphere diagnostics (exchange/transfer coeffs)
- Land Surface: Soil temperature /moisture /snow prediction /sea-ice temperature

Surface Physics Components



Land-Surface Model Processes



WRF Surface Layer Options

- Surface layer calculates heat, moisture and momentum fluxes
- Use similarity theory to determine exchange coefficients and diagnostics of 2m T and q and 10 m winds
- Provide exchange coefficient to land-surface models
- Provide friction velocity to PBL scheme
- Provide surface fluxes over water points
- Schemes have variations in stability functions, roughness lengths

Roughness Lengths

- Roughness lengths are a measure of the “initial” length scale of surface eddies.
- Roughness length depends on land-use type
- For water points roughness length is a function of surface wind speed

WRF Land-Surface Model Options

- Simple 5-layer soil model
  - No vegetation or snow cover prediction, just thermal diffusion in soil layers
- Noah, RUC, and PX Land-Surface Models
  - Sophisticated vegetation model and snow cover prediction
- The models
- Predicts soil temperature and soil moisture in layers (4 for Noah, 6 for RUC, 2 for PX)
- Predicts snow water equivalent on ground.

Snow Cover

- LSMs include fractional snow cover and predict snow water equivalent development based on precipitation, sublimation, melting and run-off
- Frozen soil water is also predicted by some schemes

Initializing LSMs

- Noah and RUC LSM require additional fields for initialization
  - Soil temperature
  - Soil moisture
  - Snow liquid equivalent
- These are in the GRIB files, but are not from observations
- They come from “offline” models driven by observations (rainfall, radiation, surface temperature, humidity wind)

Initializing LSMs

- There are consistent model-derived datasets for Noah and RUC LSMs
- But, resolution of mesoscale land-use means there will be inconsistency in elevation, soil type and vegetation
- This leads to spin-up as adjustments occur in soil temperature and moisture
- This spin-up can only be avoided by running offline model on the same grid (e.g. HRLDAS for Noah)
- Cycling land state between forecasts also helps, but may propagate errors (e.g in rainfall effect on soil moisture)

sst_update=1

- Reads lower boundary file periodically to update the sea-surface temperature (otherwise it is fixed with time)
- For long-period simulations (a week or more)
- Sea-ice can be updated
- Vegetation fraction update is included
  - Allows seasonal change in albedo, emissivity, roughness length in Noah LSM

&physics

Seven major physics categories:

- **mp_physics**: 0,1,2,3,4,5,6,8,10
- **ra_lw_physics**: 0,1,3,99
- **ra_sw_physics**: 0,1,2,3,99
- **sf_sfclay_physics**: 0,1,2
- **sf_surface_physics**: 0,1,2,3,99 (set before running **real**)
  - ucm_call = 0,1
- **bl_pbl_physics**: 0,1,2,99
- **cu_physics**: 0,1,2,3,99

Basic namelist Options

Dave Gill
Wei Wang

What is a namelist?

- A Fortran namelist contains a list of *runtime* options for the code to read in during its execution. Use of a namelist allows one to change runtime configuration without the need to recompile the source code.
- Fortran 90 namelist has very specific format, so edit with care:

  `&namelist-record` - start

  `/` - end
- As a general rule:

  Multiple columns: domain dependent

  Single column: value valid for all domains

&time_control

```
run_days            = 0,
run_hours           = 24,
run_minutes         = 0,
run_seconds         = 0,
start_year          = 2000, 2000, 2000,
start_month         = 01,   01,   01,
start_day           = 24,   24,   24,
start_hour          = 12,   12,   12,
start_minute        = 00,   00,   00,
start_second        = 00,   00,   00,
end_year            = 2000, 2000, 2000,
end_month           = 01,   01,   01,
end_day             = 25,   25,   25,
end_hour            = 12,   12,   12,
end_minute          = 00,   00,   00,
end_second          = 00,   00,   00,
interval_seconds    = 21600
history_interval    = 180,  60,   60,
frame_per_outfile   = 1000, 1000, 1000,
restart_interval    = 360,
restart             = .false.,
```

domain 1 option

nest options

Notes on `&time_control`

- *run_** time variables:
  - Model simulation length: *wrf.exe* and domain 1 only
- *start_** and *end_** time variables:
  - Program *real* will use WPS output between these times to produce lateral (and lower) boundary file
  - They can also be used to specify the start and end of simulation times for the coarse grid if *run_** variables are not set (or set to 0).

Notes on `&time_control`

- *interval_seconds*:
  - Time interval between WPS output times, and lateral BC (and lower BC) update frequency
- *history_interval*:
  - Time interval in <u>minutes</u> when a history output is written
  - The time stamp in a history file name is the time when the history file is first written, and multiple time periods may be written in one file. e.g. a history file for domain 1 that is first written for 1200 UTC Jan 24 2000 is

`wrfout_d01_2000-01-24_12:00:00`

Notes on `&time_control`

- *frame_per_outfile*:
  - Number of history times written to one file.
- *restart_interval*:
  - Time interval in minutes when a restart file is written.
  - By default, restart file is not written at hour 0.
  - A restart file contains only one time level data, and its valid time is in its file name, e.g. a restart file for domain 1 that is valid for 0000 UTC Jan 25 2000 is

    `wrfrst_d01_2000-01-25_00:00:00`
- *restart:*
  - whether this is a restart run

Notes on *restart*

- What is a *restart* run?
  - A restart run is a continuation of a model run.
- How to do a *restart* run:
  - In the first run, set *restart_interval* to a value that is within the model integration time.
  - A restart file will be created. e.g.

    `wrfrst_d01_2000-01-25_00:00:00`
- When doing a restart run:
  - Set *restart* = .true.,
  - Set start times to restart times in namelist

&time_control

```
io_form_history      = 2,
io_form_restart      = 2,
io_form_input        = 2,
io_form_boundary     = 2,
debug_level          = 0,
```

IO format options:
- = 1, binary
- = 2, netcdf (most common)
- = 4, PHDF5
- = 5, Grib 1
- =10, Grib 2
- =11, pnetCDF

For large file:
`io_form_restart = 102` : write output in patch sizes: fast for large grids and useful for restart file

Debug print control: Increasing values give more prints.

&domains

```
time_step                  = 180
time_step_fract_num        = 0,
time_step_fract_den        = 1,
max_dom                    = 1,
e_we                       = 74,     112,    94,
e_sn                       = 61,     97,     91,
e_vert                     = 28,     28,     28,
num_metgrid_levels         = 21
num_metgrid_soil_levels    = 4
dx                         = 30000, 10000, 3333,
dy                         = 30000, 10000, 3333,
eta_levels                 = 1.0,0.996,0.99,0.98,… 0.0
p_top_requested            = 5000,
```

nest options

Notes on **&domains**

- *time_step, time_step_fract_num, time_step_frac_den*:
  - Time step for model integration in seconds.
  - Fractional time step specified in separate integers of numerator and denominator.
  - ARW: 6*DX (DX is grid distance in km): for DX=10km, time_step = 60
- *e_we, e_sn, e_vert*:
  - Model grid dimensions (staggered) in X, Y and Z directions.
- *num_metgrid_levels*:
  - Number of *metgrid* (input) data levels.
- *num_metgrid_soil_levels*:
  - Number of soil data levels in the input data

  Found by typing **`ncdump -h met_*.d01.<date> | more`**
- *dx, dy*:
  - grid distances: in meters for ARW.

Notes on `&domains`

- *p_top_requested*:
  - Pressure value at the model top.
  - Constrained by the available data from WPS.
  - Default is 5000 Pa (recommended as lowest Ptop)

- *eta_levels*:
  - Specify your own model levels from 1.0 to 0.0.
  - If not specified, program *real* will calculate a set of levels

Where do I start?

- Always start with a *namelist* template provided in a test case directory, whether it is an ideal case, or real data case.
  - A number of namelist templates are provided in *test/test_<case>/* directories

  For example: in *test/em_real/*, there are

    namelist.input.4km ~ 4 km grid size

    namelist.input.jun01 ~ 10 km grid size

    namelist.input.jan00 ~ 30 km grid size

Where do I start?

- For different applications, please refer to pages 5-25 to 5-27 of the User's Guide:
  - 2 or 4 km convection-permitting runs
  - 20 – 30 km, 2 – 3 day runs
  - Antarctic region
  - Tropical storm forecasting
  - Regional climate

Where do I start?

- Use document to guide the modification of the namelist values:
  - **`run/README.namelist`**
  - **`test/em_real/examples.namelist`**
  - User's Guide, Chapter 5 (online version has the latest)
  - Full list of namelists and their default values can be found in Registry files: Registry.EM_COMMON (ARW), and registry.io_boilerplate (IO options, shared)

To run a job in a different directory..

- Directories *run/* and test_<*case*>/ are convenient places to run, but it does not have to be.
- Copy or link the content of these directories to another directory, including physics data files, wrf input and boundary files, wrf namelist and executables, and you should be able to run a job anywhere on your system.

Check Output

Output After a Model Run

- Standard out/error files:

  `wrf.out`, or **`rsl`**.* files

- Model history file(s):

  `wrfout_d01_<date>`

- Model restart file(s), optional

  `wrfrst_d01_<date>`

Output from a multi-processor run

The standard out and error will go to the following files for a MPI run:

`mpirun -np 4 ./wrf.exe` ➔

```
rsl.out.0000          rsl.error.0000
rsl.out.0001          rsl.error.0001
rsl.out.0002          rsl.error.0002
rsl.out.0003          rsl.error.0003
```

There is one pair of files for each processor requested

What to Look for in a standard out File?

Check run log file by typing

`tail wrf.out`, or

`tail rsl.out.0000`

You should see the following if the job is successfully completed:

`wrf: SUCCESS COMPLETE WRF`

How to Check Model History File?

- Use **ncdump**:

  ```
  ncdump -v Times wrfout_d01_<date>
  ```

  to check output times. Or

  ```
  ncdump -v U wrfout_d01_<date>
  ```

  to check a particular variable (U)

- Use **ncview** or **ncBrowse** (great tools!)
- Use post-processing tools (see talks later)

What is in a *wrf.out* or *rsl* file?

- A printout of information from namelist file: *namelist.output*
- Time taken to compute one model step:

```
Timing for main: time 2000-01-24_12:03:00 on domain   1:    3.25000 elapsed seconds.
Timing for main: time 2000-01-24_12:06:00 on domain   1:    1.50000 elapsed seconds.
Timing for main: time 2000-01-24_12:09:00 on domain   1:    1.50000 elapsed seconds.
Timing for main: time 2000-01-24_12:12:00 on domain   1:    1.55000 elapsed seconds.
```

- Time taken to write history and restart file:

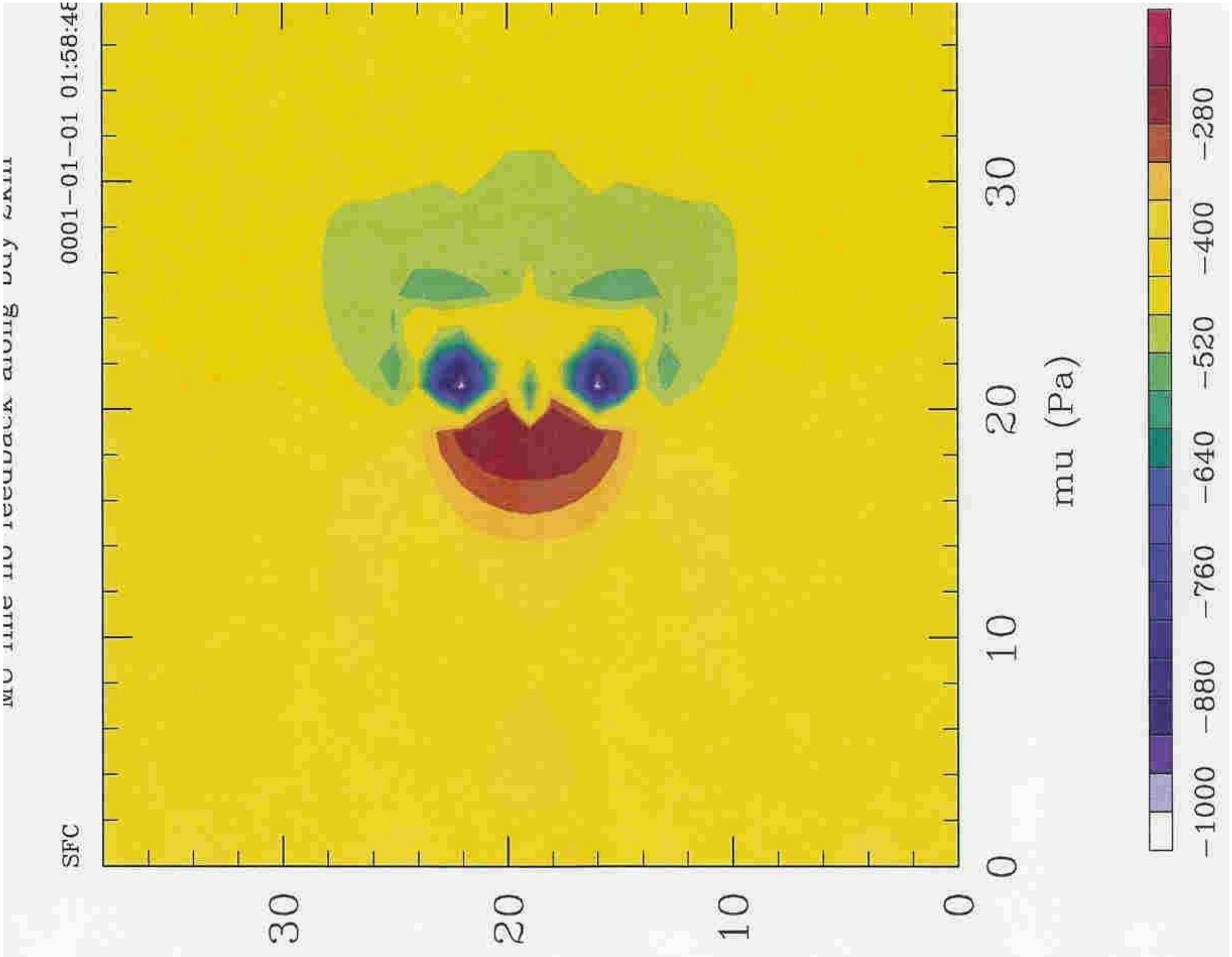
```
Timing for Writing wrfout_d01_2000-01-24_18:00:00 for domain   1:   0.14000 elapsed
      seconds.
```

- Any model error prints: (example from ARW run)

```
5 points exceeded cfl=2 in domain 1 at time 4.200000 MAX AT i,j,k: 123 48 3
      cfl,w,d(eta)= 4.165821
```

➡ An indication the model has become numerically unstable

ARW Nesting



- Nesting definitions
- WRF nesting capabilities
- Domains, staggering, feedback
- Nesting performance
- Masked interpolation and feedback
- Some suggestions

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

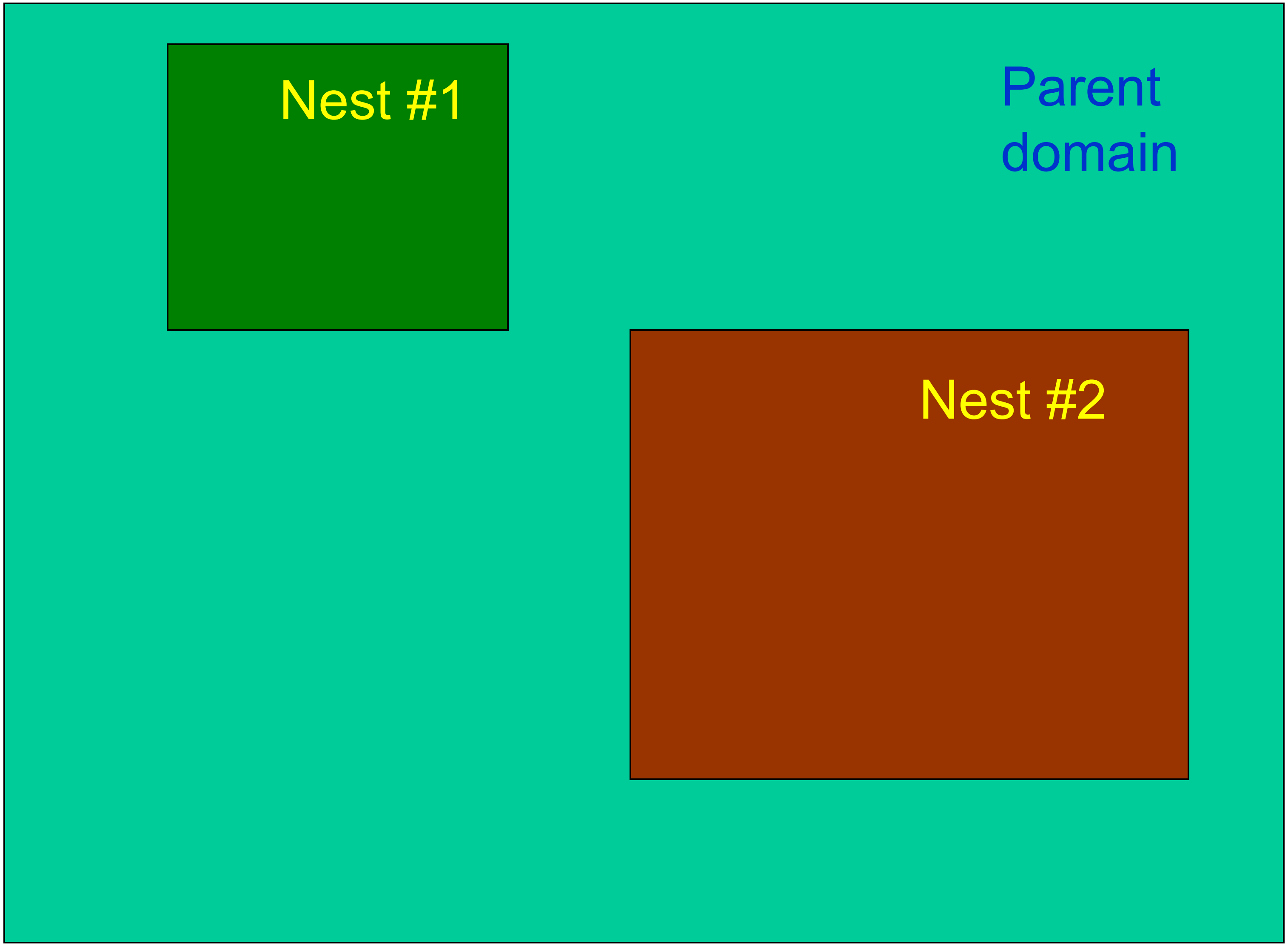
Nesting Basics - ARW

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

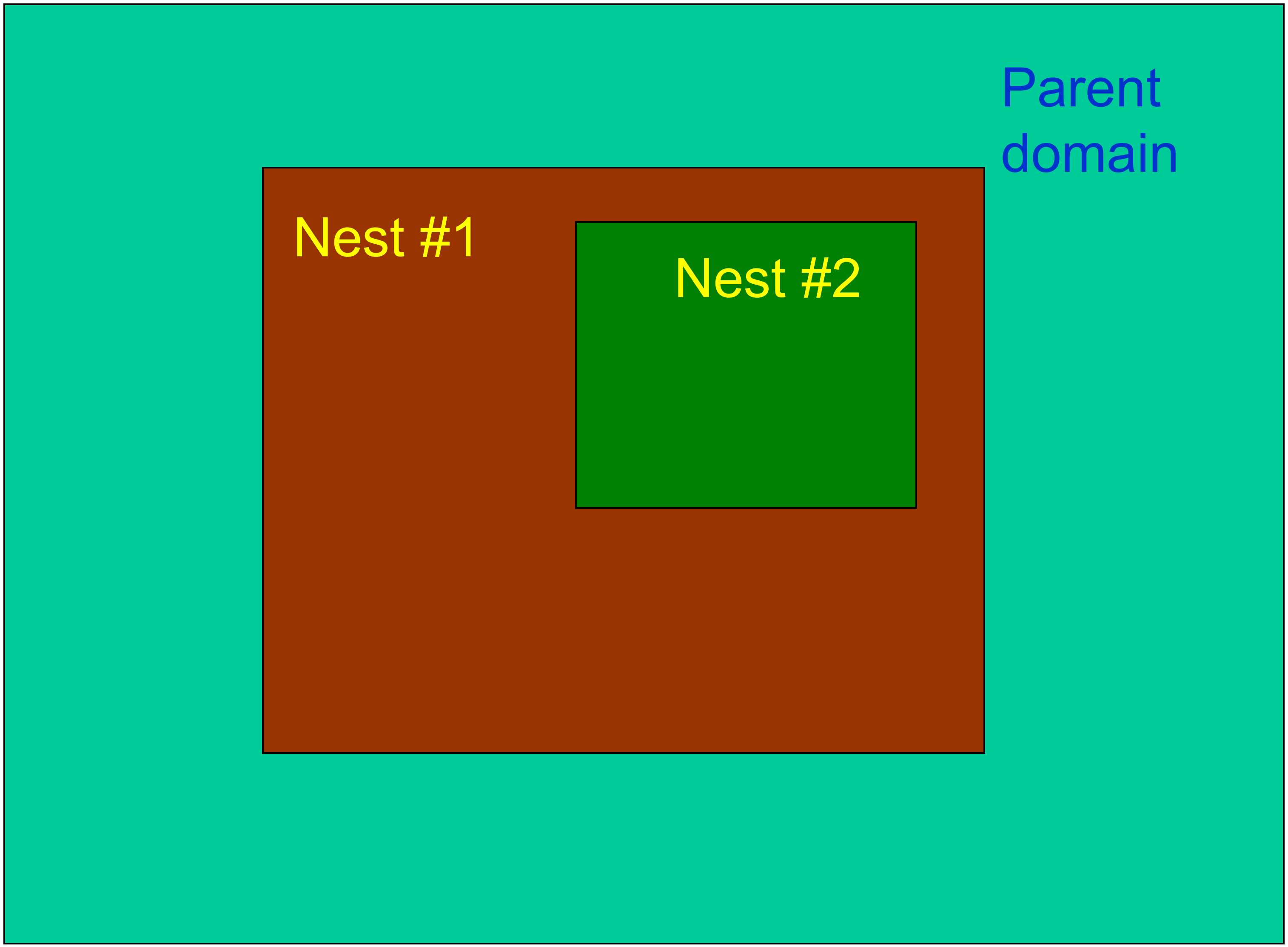
- One-way/two-way nesting with a single input file, all fields interpolated from the coarse grid
- One-way/two-way nesting with multiple input files, each domain with a full input data file
- One-way/two-way nesting with the coarse grid data including all meteorological fields, and the fine-grid domains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

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



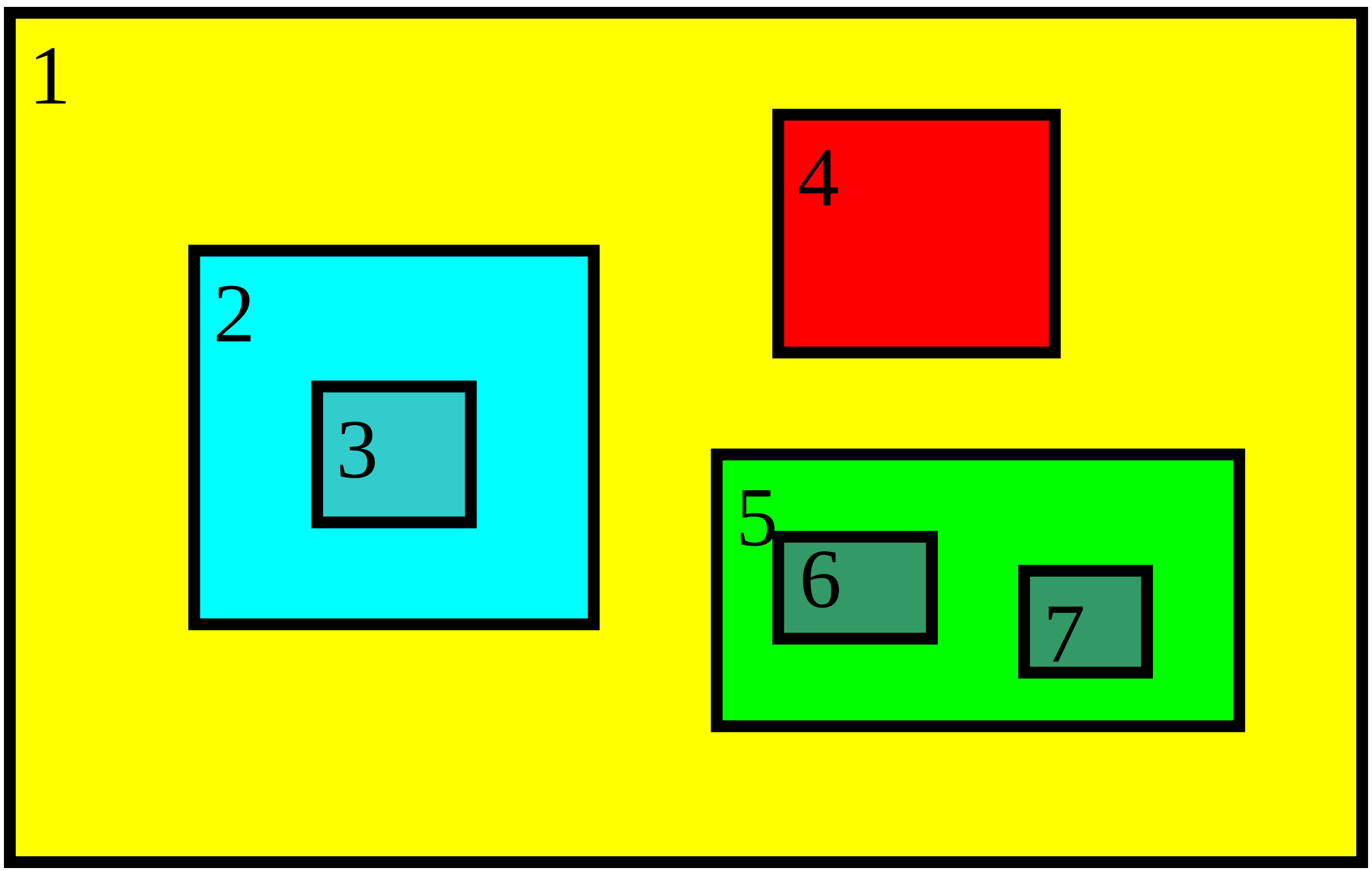
Two levels of nests, with nest #1 acting as the parent for nest #2



These are all OK

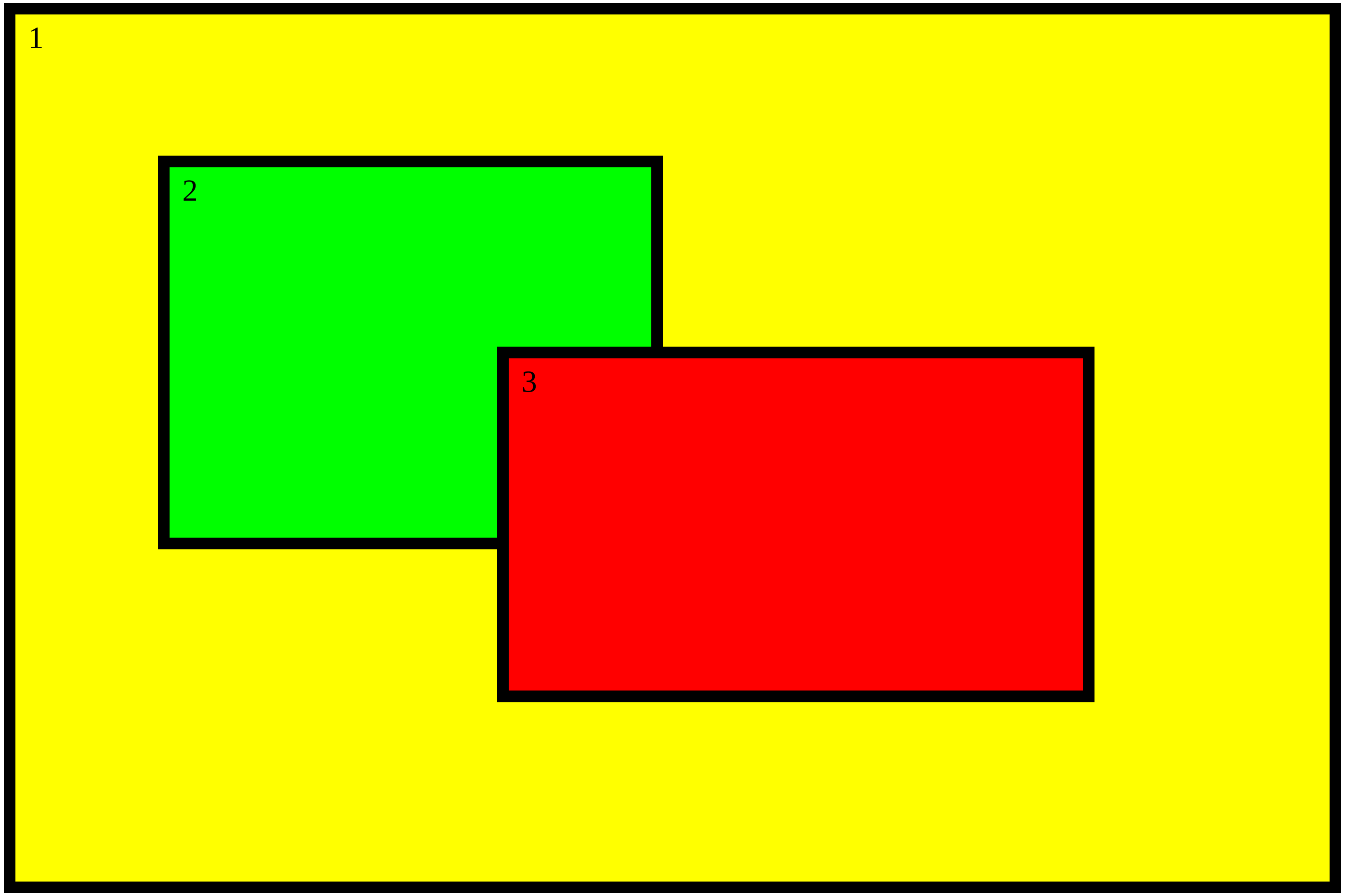
Telescoped to any depth

Any number of siblings



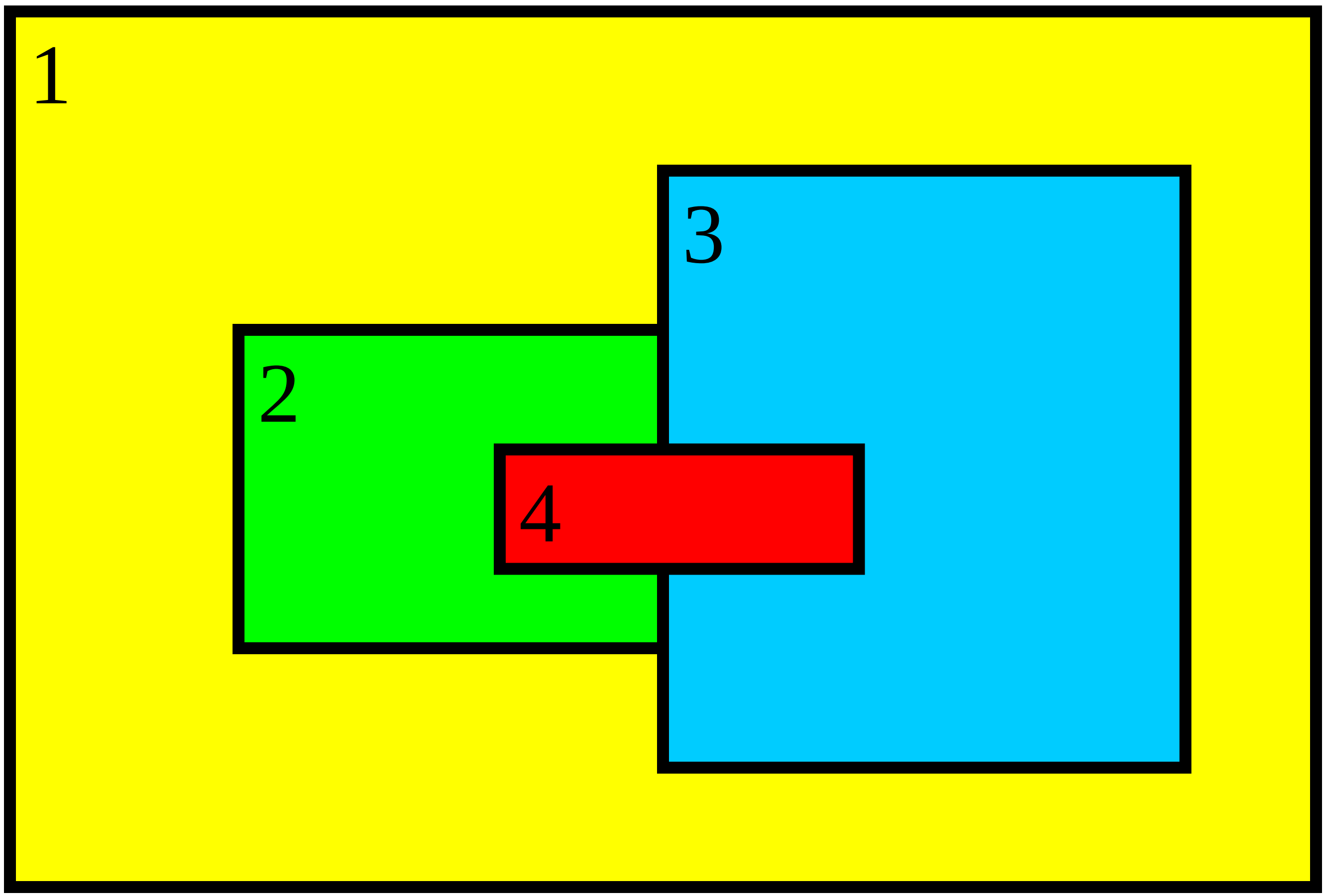
Not OK for 2-way

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)

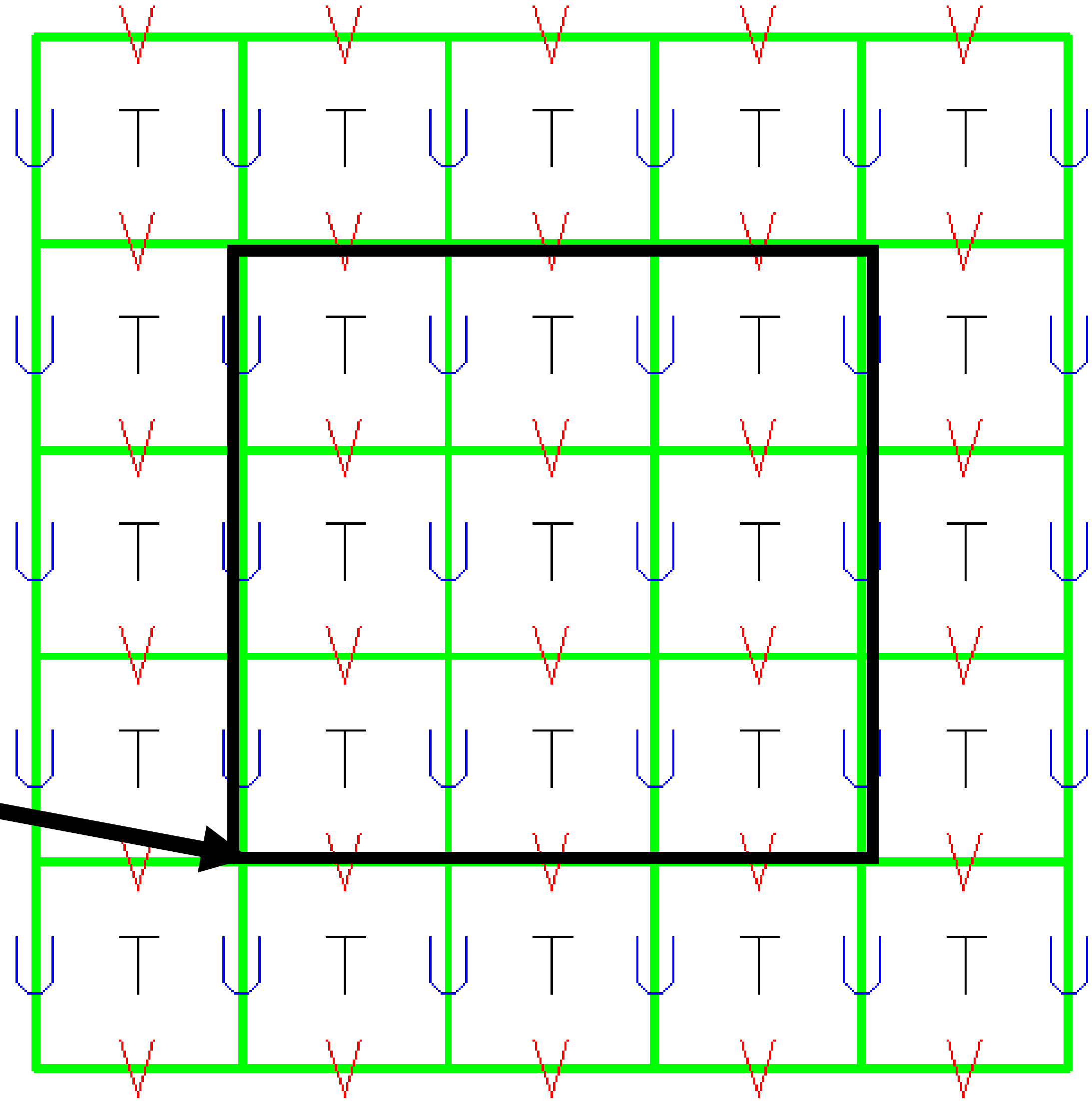


WRF Coarse-Fine Overlap

- The rectangular fine grid is coincident with a portion of the high-resolution grid that covers the entire coarse grid cell
- The nested domain can be placed anywhere within the parent domain and the nested grid cells will exactly overlap the parent cells at the coincident cell boundaries.
- Coincident parent/nest grid points eliminate the need for complex, generalized remapping calculations, and enhances model performance and portability.
- The grid design was created with moving nests in mind.

ARW Coarse Grid Staggering

i_parent_start
j_parent_start



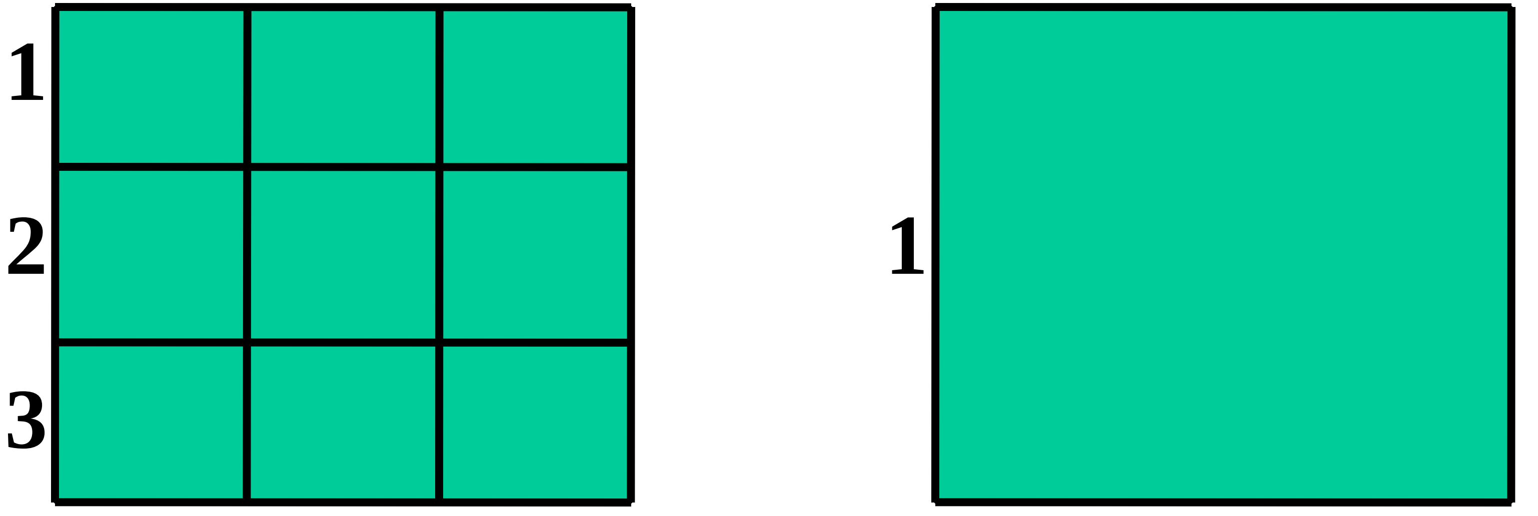
ARW Coarse Grid Staggering 3:1 Ratio

Feedback:

U : column

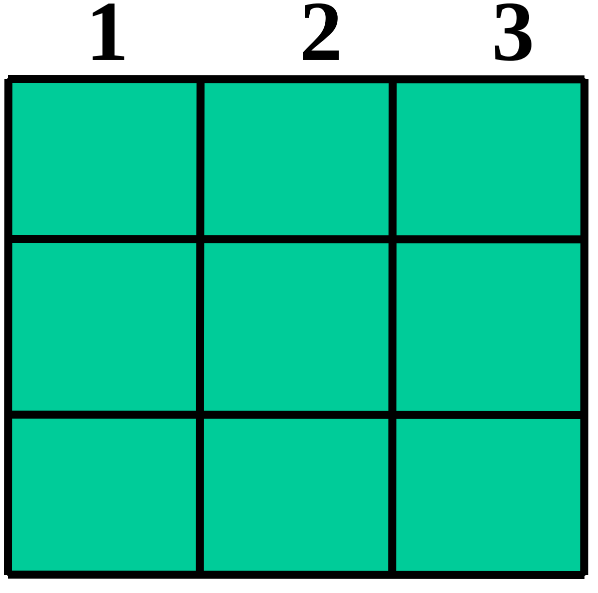
V : row

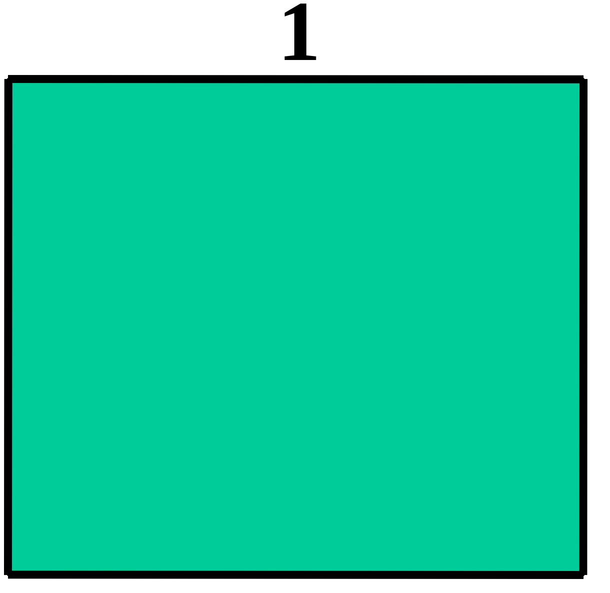
T : cell



ARW Coarse Grid Staggering 3:1 Ratio

Feedback:
U : column
V : row
T : cell





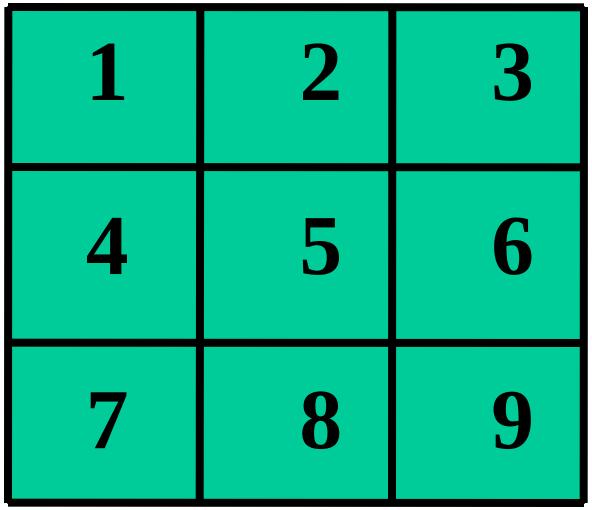
ARW Coarse Grid Staggering 3:1 Ratio

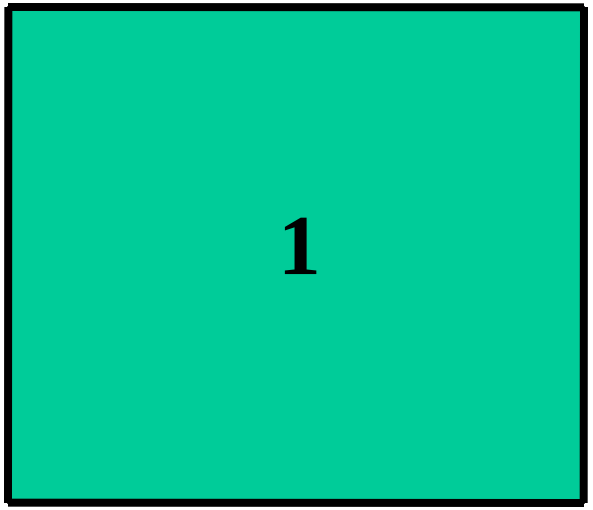
Feedback:

U : column

V : row

T : cell





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.

Nesting Performance

- Example: assume 3:1 nest ratio

If the nest has the same number of grid cells, then the **amount of CPU** to do a single time step for a coarse grid (CG) and a fine grid step (FG) is **approximately the same**.

Since the fine grid (3:1 ratio) has 1/3 the grid distance, it requires 1/3 the model time step. Therefore, the FG requires 3x the CPU to catch up with the CG domain.

Nesting Performance

- Example: assume 3:1 nest ratio

If you try to cover the SAME area with a FG domain as a CG domain, you need **(ratio)^2 grid points**.

With the associated FG time step ratio, you require a **(ratio)^3**.

With a 3:1 ratio, a FG domain covering the same area as a CG domain **requires 27x CPU**.

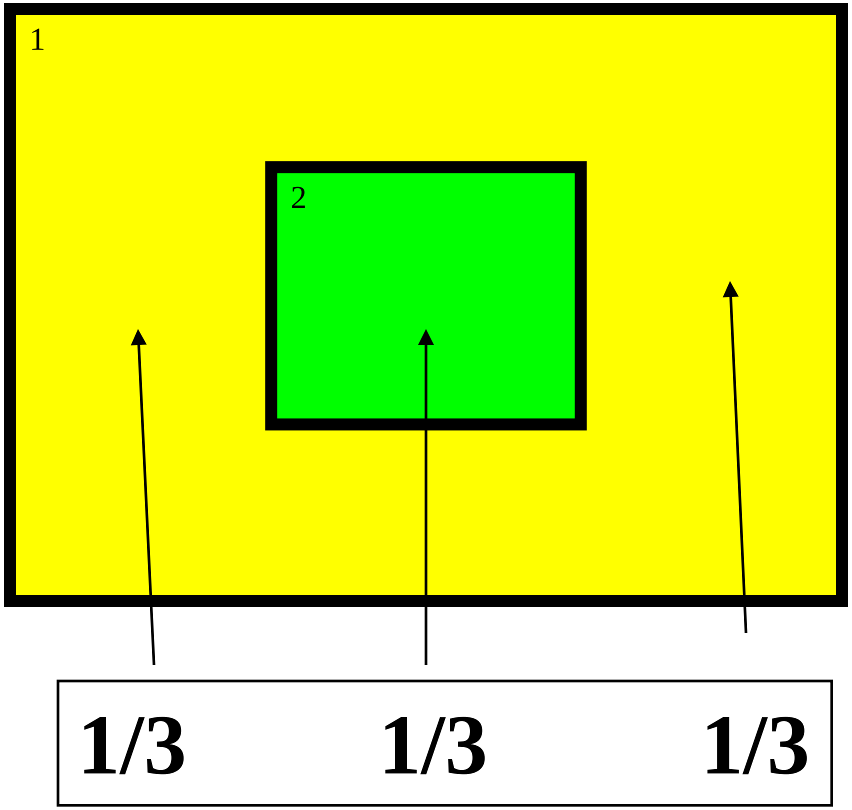
Nesting Performance

- Example: assume **10:1 nest ratio**

To change your test case from 50-km resolution to a finer 5-km resolution would be **1000x more** expensive.

Nesting Suggestions

- The minimum distance between the nest boundary and the parent boundary is FOUR grid cells
- You should have a **MUCH** larger buffer zone
- It is not unreasonable to have approximately 1/3 of your coarse-grid domain surrounding each side of your nest domain



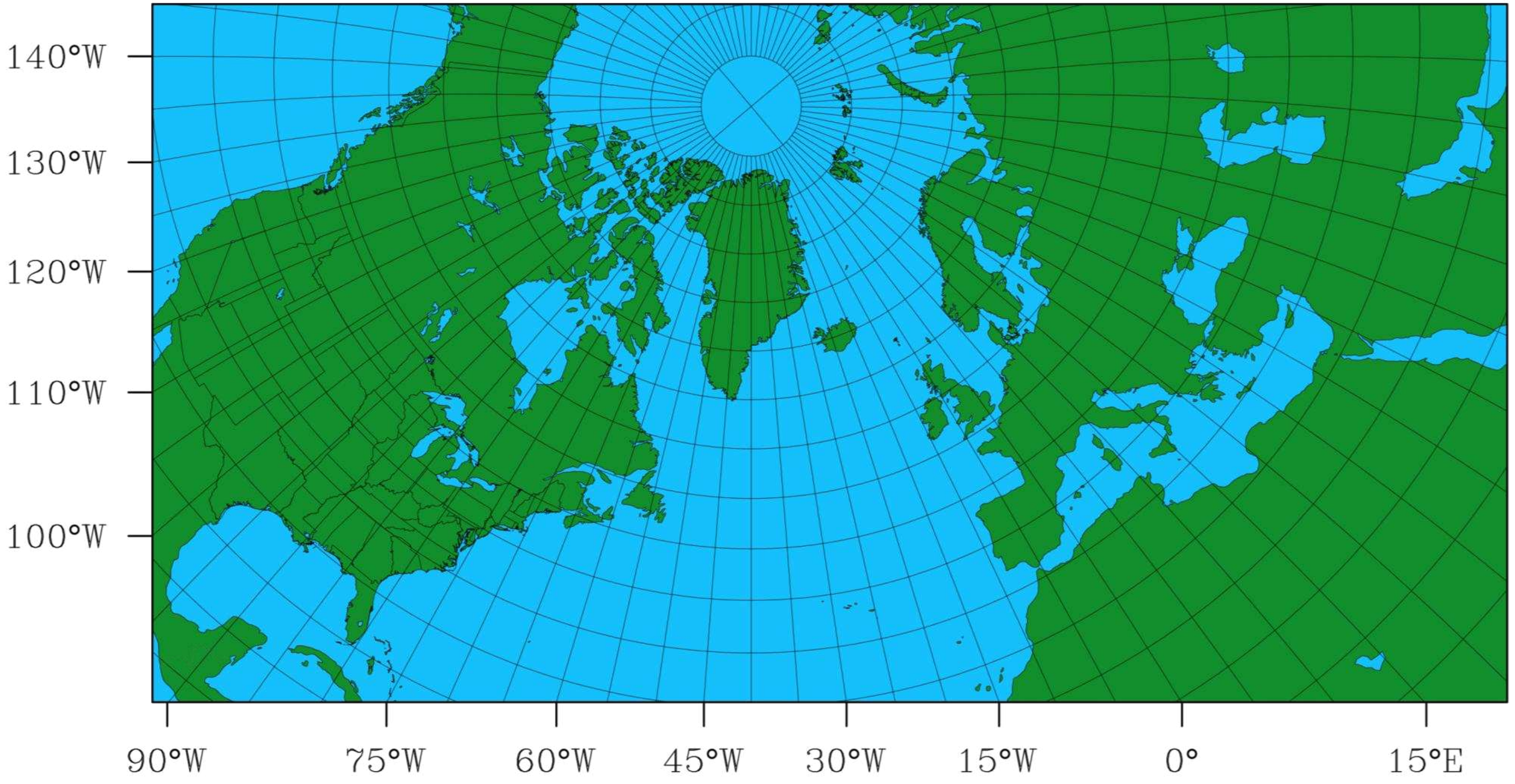
Nesting Suggestions

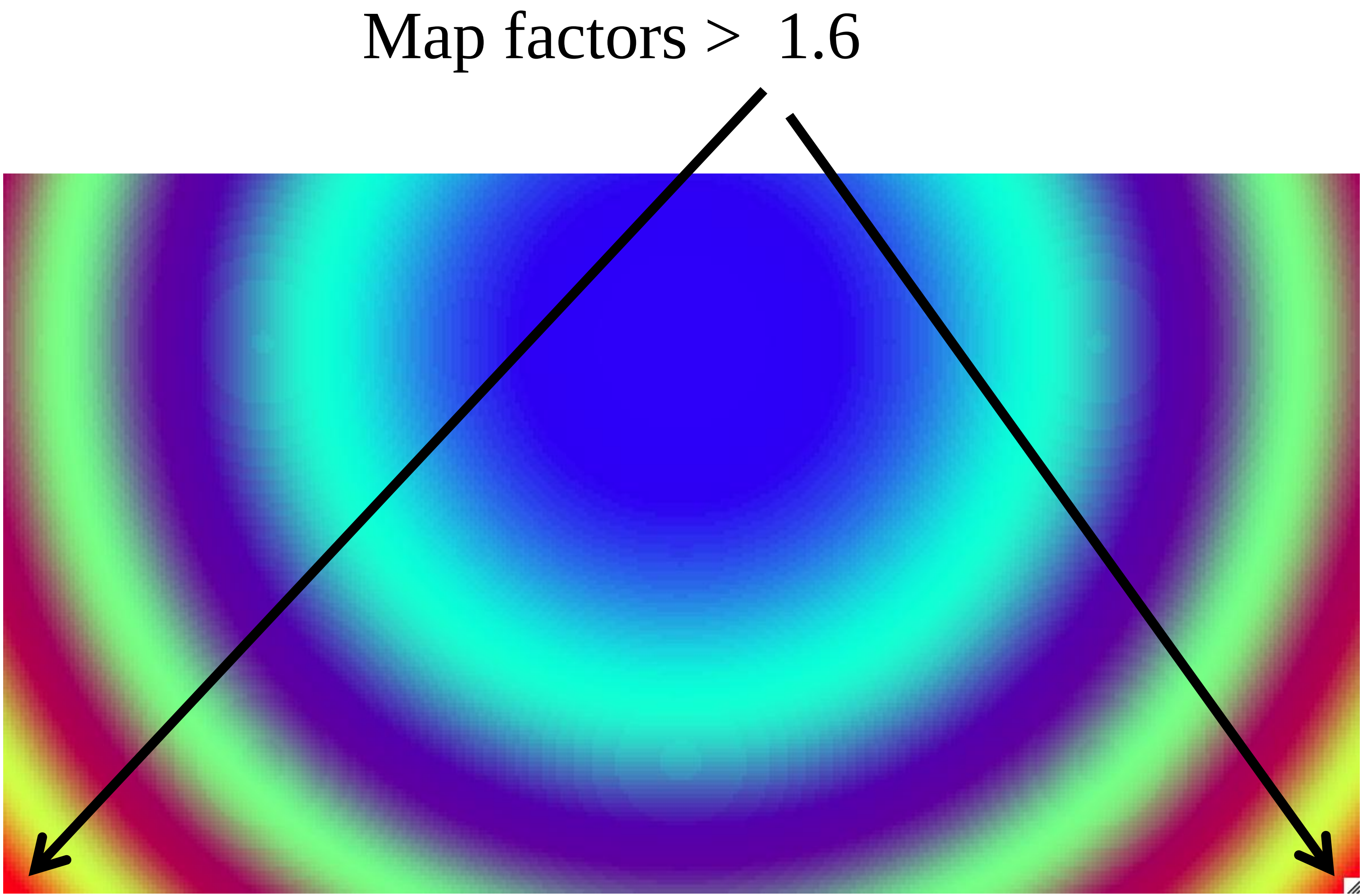
- Start with designing your inner-most domain. For a traditional forecast, you want everything important for that forecast to be entirely contained inside the domain.
- Then start adding parent domains at a 3:1 or 5:1 ratio. A parent should not have a smaller size (in grid points). Keep adding domains until the most coarse WRF grid has a no more than a 3:1 to 5:1 ratio to the external model (first guess) data.

Nesting Suggestions

- Larger domains tend to be better than smaller domains.
- A 60 m/s parcel moves at > 200 km/h. A 2-km resolution grid with 100x100 grid points could have most of the upper-level initial data swept out of the domain within a couple of hours.

Nesting Suggestions




Map factors > 1.6

Nesting Suggestions

- The most-coarse domain may have a geographic extent that causes large map factors.

```
time_step              = 300 (BLOWS UP)
dx                     = 45000,15000,5000
grid_id                = 1,    ,2    ,3
parent_id              = 0,    ,1    ,2
parent_grid_ratio      = 1,    ,3    ,3
parent_time_step_ratio = 1,    ,3    ,3
```

Nesting Suggestions

- Reducing the time step so that the coarse grid is stable makes the model too expensive. 1.6x

```
time_step                = 180 (STABLE, PRICEY)
dx                       = 45000,15000,5000
grid_id                  = 1,    ,2    ,3
parent_id                = 0,    ,1    ,2
parent_grid_ratio        = 1,    ,3    ,3
parent_time_step_ratio   = 1,    ,3    ,3
```

Nesting Suggestions

- Only reduce the time step on the coarse grid, and keep the fine grid time steps at their approx original values.

```
time_step              = 180 (STABLE, CHEAP)
dx                     = 45000,15000,5000
grid_id                = 1,    ,2    ,3
parent_id              = 0,    ,1    ,2
parent_grid_ratio      = 1,    ,3    ,3
parent_time_step_ratio = 1,    ,2    ,3
```

Nesting Suggestions

- Model time step is always proportional to the time step of the most coarse grid.
- The coarse grid is the only grid impacted with large map factors: dt(s) = 6*dx(km) but the nominal grid distance needs to be scaled: dt(s) = dx(km) / MAX (map factor)
- Reducing the coarse grid time step does not significantly reduce model performance if you can tweak the time step ratio.

Nesting Suggestions

- The time step ratio and grid distance ratio are not necessarily identical, and may used effectively when large map factors in the coarse grid domain force a time step reduction for stability.
- If map factors are causing stability troubles, it is usually only the most coarse grid that is impacted since the fine grid is usually in the middle of the domain.

Nesting Suggestions

- Set up domain first to provide good valid forecast, then deal with efficiency
- Selecting a set of domains with the reason “it is all I can afford” gets you into trouble
- Numerically stable and computationally expedient do not imply scientifically or physically valid

Nesting Suggestions

- Assume a coarse grid at 12-km going down to about 50-m resolution, using a 3:1 ratio
- If each domain is about the same size, then if the 12-km domain is "1" unit of time, then the relative cost per domain to complete a CG time step:

Domain 2	4000 m	= 3x
Domain 3	1333 m	= 9x
Domain 4	450 m	= 27x
Domain 5	150 m	= 81x
Domain 6	50 m	= 729x

Nesting Suggestions

- You cannot simply use “hundreds” of processors to play catch-up, as most domains are not large enough to effectively use that many processors.
- You need to match the expensive domains with an appropriate number of processors, with a reasonable expectation of completion.

Nest namelist Options

&time_control

```
run_days          = 0,
run_hours         = 24,
run_minutes       = 0,
run_seconds       = 0,
start_year        = 2000, 2000, 2000,
start_month       = 01,   01,   01,
start_day         = 24,   24,   24,
start_hour        = 12,   12,   12,
start_minute      = 00,   00,   00,
start_second      = 00,   00,   00,
end_year          = 2000, 2000, 2000,
end_month         = 01,   01,   01,
end_day           = 25,   25,   25,
end_hour          = 12,   12,   12,
end_minute        = 00,   00,   00,
end_second        = 00,   00,   00,
interval_seconds = 21600
```

First column: domain 1 option

These control the start and end times of the nests. They can be different from the parent domain, but must fit in the time window of the parent domain

&time_control

```
interval_seconds      = 21600
history_interval      = 180,  60,   60,
frame_per_outfile     = 1000, 1000, 1000,
restart_interval      = 360,
```

History output may be split into multiple files

- History files are written separately for each domain
- History intervals may be different for different domains
- restart files are also written one per domain

&time_control

Nest input option:

```
input_from_file  = .true., .true., .true.,
fine_input_stream   = 0, 2, 2,
```

Specify what fields to use in nest input: they can be all (0), or data specified in I/O stream 2 in Regsitry (2).
Limited use: if a nest starts at a later time, or have a updated analysis only on domain 1.
(*used less common*)

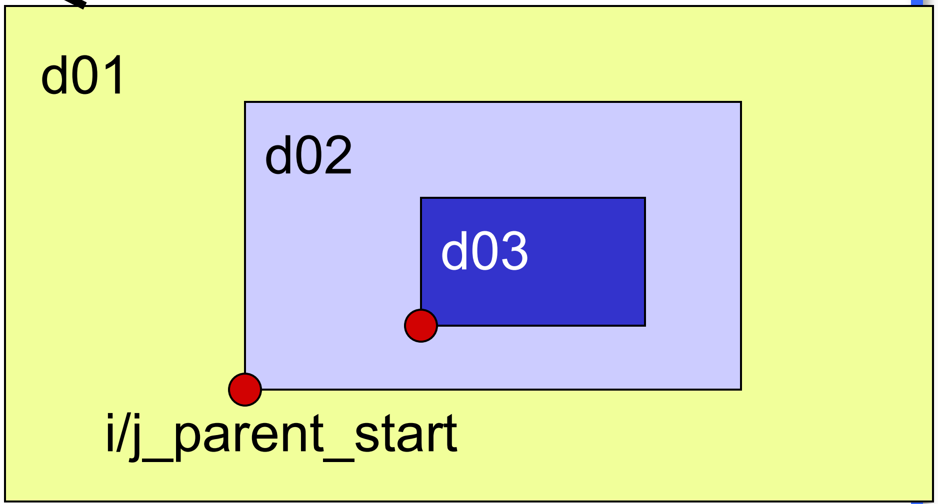
Whether to produce in *real* and use nest wrfinput files in *wrf*. This is usually the case for real-data runs. For idealized nest runs, set it to .false. .
(*commonly used*)

&domains

```
max_dom    = 3,
e_we       = 74, 112,94,
e_sn       = 61, 97, 91,
e_vert     = 28, 28, 28,
grid_id          = 1,  2,  3,
parent_id        = 0,  1,  2,
i_parent_start = 0, 31, 30,
j_parent_start = 0, 17, 30,
```

Activate nests: no. of domains to run

Dimensions of all domains; same as in WPS.



Make sure the nest domain parameters match those defined in WPS

&domains

Nest domain parameters:

```
dx = 30000., 10000., 3333.33,
dy = 30000., 10000., 3333.33,
parent_grid_ratio = 1, 3, 3,
parent_time_step_ratio = 1,3,3,
```

For fractional grid distance, use at least 2 decimal places

All 4 variables must be specified. *Grid ratio* can be any integer, and *time step ratio* can be different from grid ratio. Grid distance is in meters, even for lat/lon map projection.

&domains

```
feedback      = 1,
smooth_option = 2,
```

When feedback is on, this option can be selected to smooth the area in the parent domain where the nest is. Valid values are 0,1,2.

Whether a nest will overwrite parent domain results. Setting feedback=0 ➔ 'one-way' nesting in a concurrent run.

&bdy_control

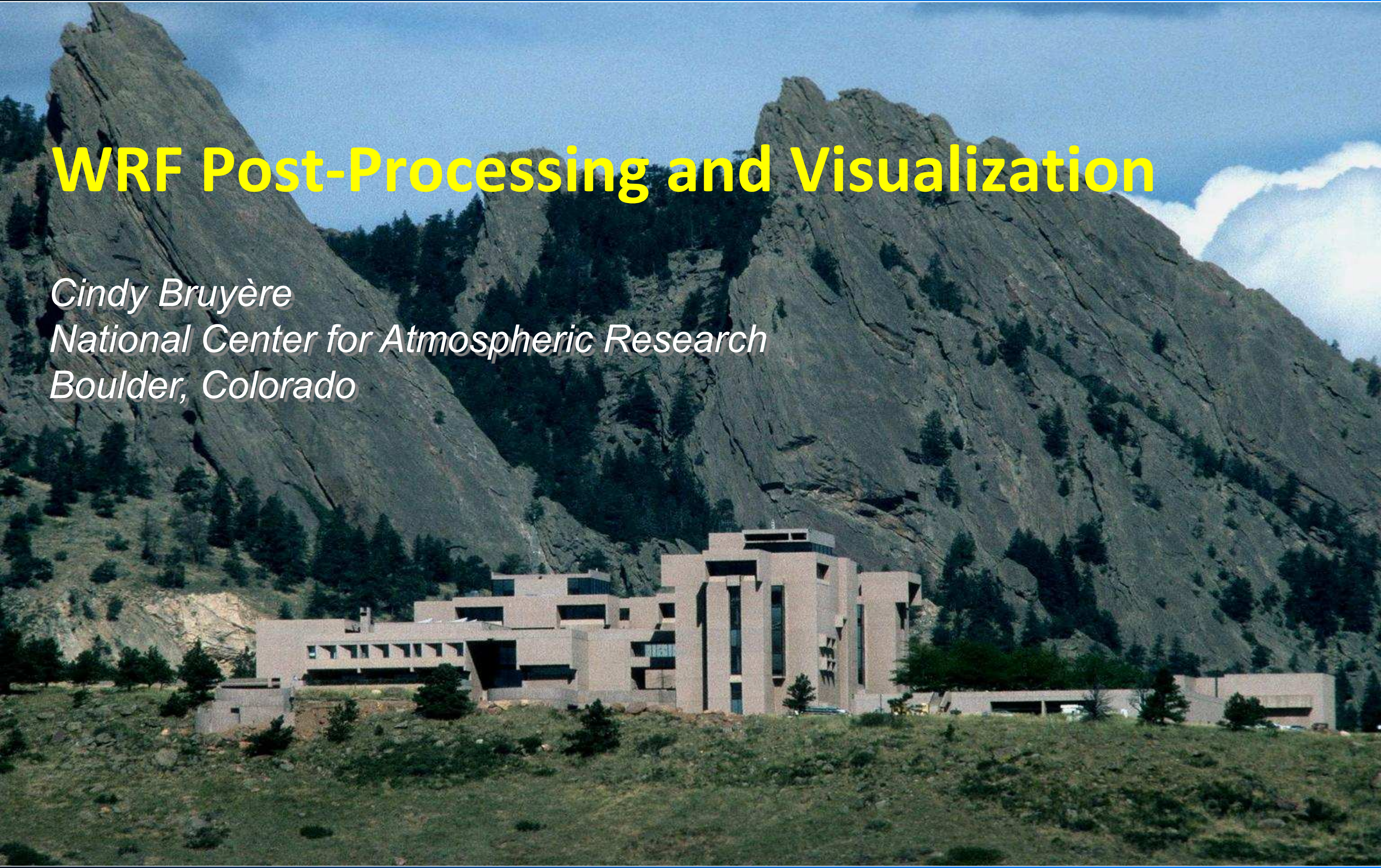
```
spec_bdy_width = 5,
spec_zone      = 1,  (ARW only)
relax_zone     = 4,  (ARW only)
specified      = .T.,.F.,.F.,
nested         = .F.,.T.,.T.,
```

Boundary condition option for domain 1.

May change *relax_zone* and *spec_bdy_width*

Boundary condition option for nests.

NCAR is Sponsored by NSF and this work is partially supported by the
Willis Research Network and the Research Partnership to Secure Energy for America



WRF Post-Processing and Visualization

Cindy Bruyère
National Center for Atmospheric Research
Boulder, Colorado

Graphical Packages

- **NCL** UG: 9-2
  - Graphical package
- **ARWpost** UG: 9-28
  - Converter (GrADS)
- **RIP4** UG: 9-19
  - Converter and interface to graphical package NCAR Graphics
- **UPP** UG: 9-35
  - Converter (GrADS & GEMPAK)
- **VAPOR** UG: 9-58
  - Converter and graphical package
  - Support: VAPOR
- **IDV** unidata.ucar.edu
  - GRIB (from WPP)
  - GEMPAK (from wrf2gem)
  - vis5d (from ARWpost)
  - CF complaint data (from wrf_to_cf)
  - Support: unidata
- **GEMPAK**
  - Data from wrf2gem or UPP
  - Support: unidata

MatLab / IDL / R / ferret

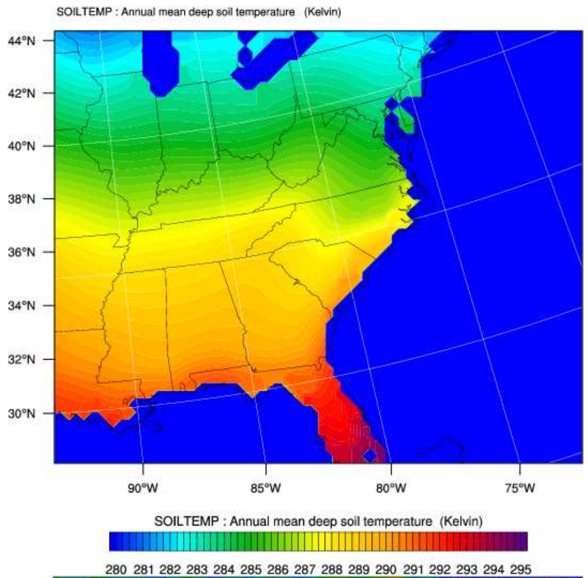
Graphical Packages

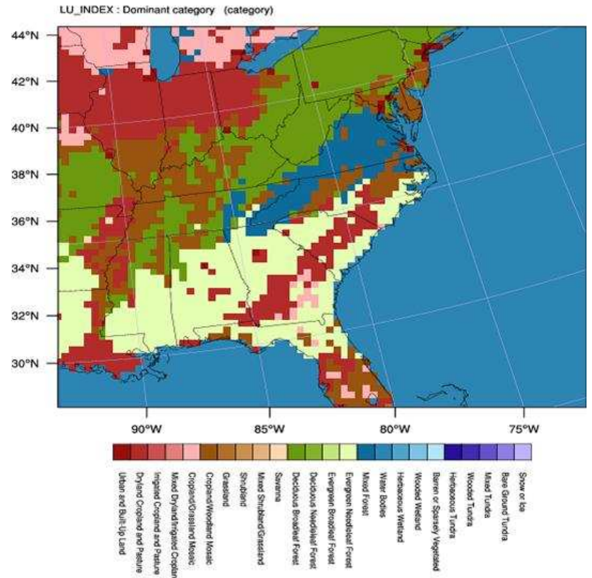
	NCL	RIP4	ARWpost (GrADS)	UPP	VAPOR	IDV
Directly ingest WRF data	***Y***	**N** *converter*	**N / *(Y)*** *converter*	**N** *converter*	**N / *(Y)*** *converter*	**N / *(Y)*** *converter*
Intermediate files	***N***	lots	large file	**Y**	large file	**Y**
WPS DATA	***Y***	***Y***	***Y***	**N**	**N**	**Y**
wrfinput data	***Y***	***Y***	***Y***	**N**	**N**	**Y**
Idealized data files	***Y***	***Y***	***Y***	**N**	**N**	**Y**
Input format	*netCDF*	*netCDF*	*netCDF*	*netCDF / binary*	*netCDF*	*netCDF*
Vertical Output Coordinate	eta pressure height	eta pressure height	eta pressure height	pressure	eta	if pre-processed
Software required *(All binaries are free)*	*NCL*	*NCARG*	*GrADS*	*GrADS / GEMPAK*	*VAPOR*	*JAVE*
Diagnostics	*some*	*> 100*	*some*	*some*	*limited*	*limited*

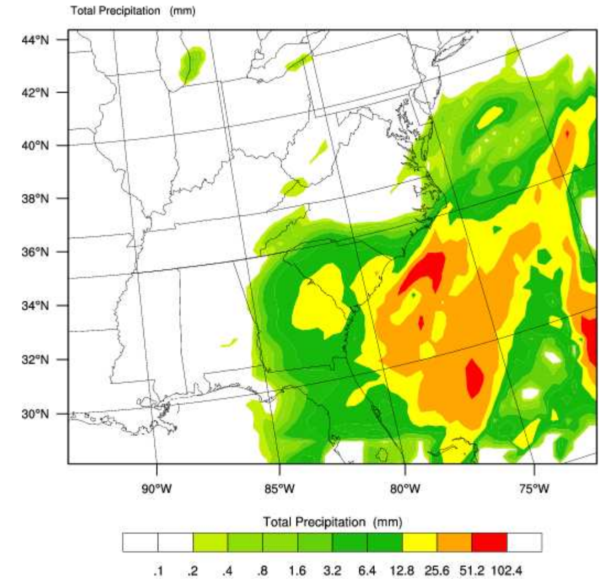
NCL

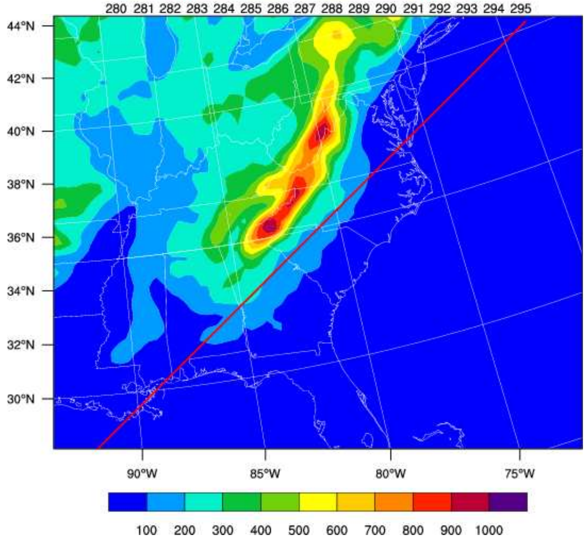
- NCAR Command Language
- http://www.ncl.ucar.edu
- Read WRF-ARW data directly
- Download
  - http://www.ncl.ucar.edu/Download
  - Fill out short registration form *(there is a short waiting period)*
  - Get version 5.1.1 or ***later*** *(current v6.1.0-beta)*
- NCARG_ROOT environment variable
  - setenv NCARG_ROOT /usr/local/ncl

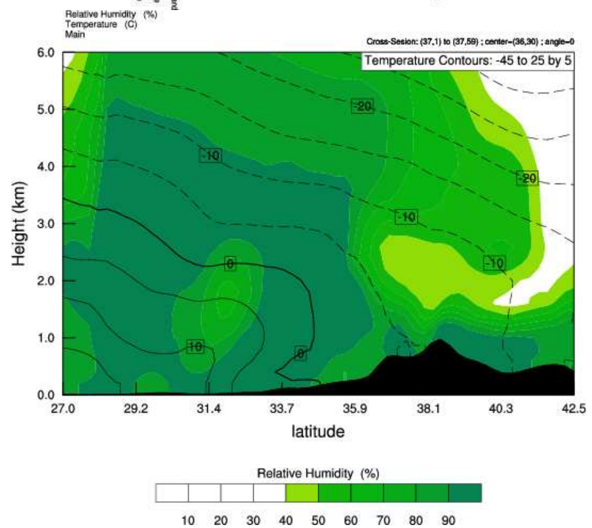
Examples

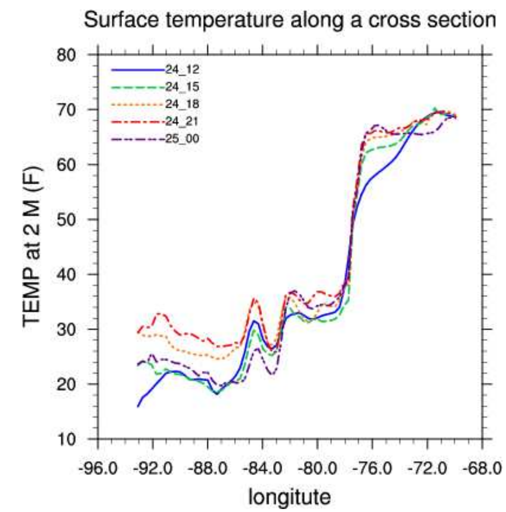

SOILTEMP : Annual mean deep soil temperature (Kelvin)
44°N
42°N
40°N
38°N
36°N
34°N
32°N
30°N
90°W
85°W
80°W
75°W
SOILTEMP : Annual mean deep soil temperature (Kelvin)
280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295


LU_INDEX : Dominant category (category)
44°N
42°N
40°N
38°N
36°N
34°N
32°N
30°N
90°W
85°W
80°W
75°W

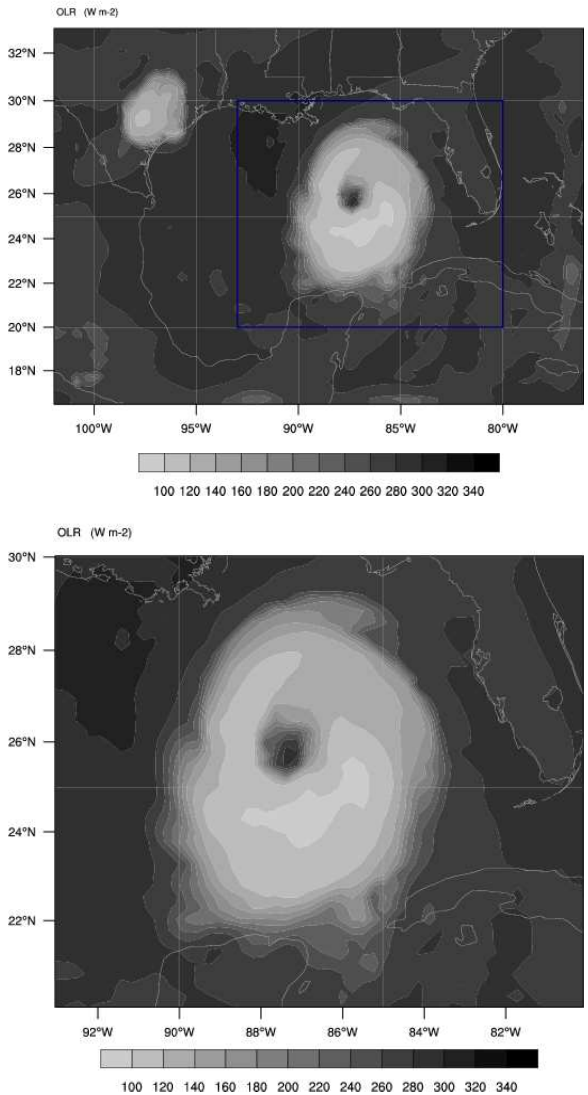

Total Precipitation (mm)
44°N
42°N
40°N
38°N
36°N
34°N
32°N
30°N
90°W
85°W
80°W
75°W
Total Precipitation (mm)
.1 .2 .4 .8 1.6 3.2 6.4 12.8 25.6 51.2 102.4


44°N
42°N
40°N
38°N
36°N
34°N
32°N
30°N
90°W
85°W
80°W
75°W
100 200 300 400 500 600 700 800 900 1000

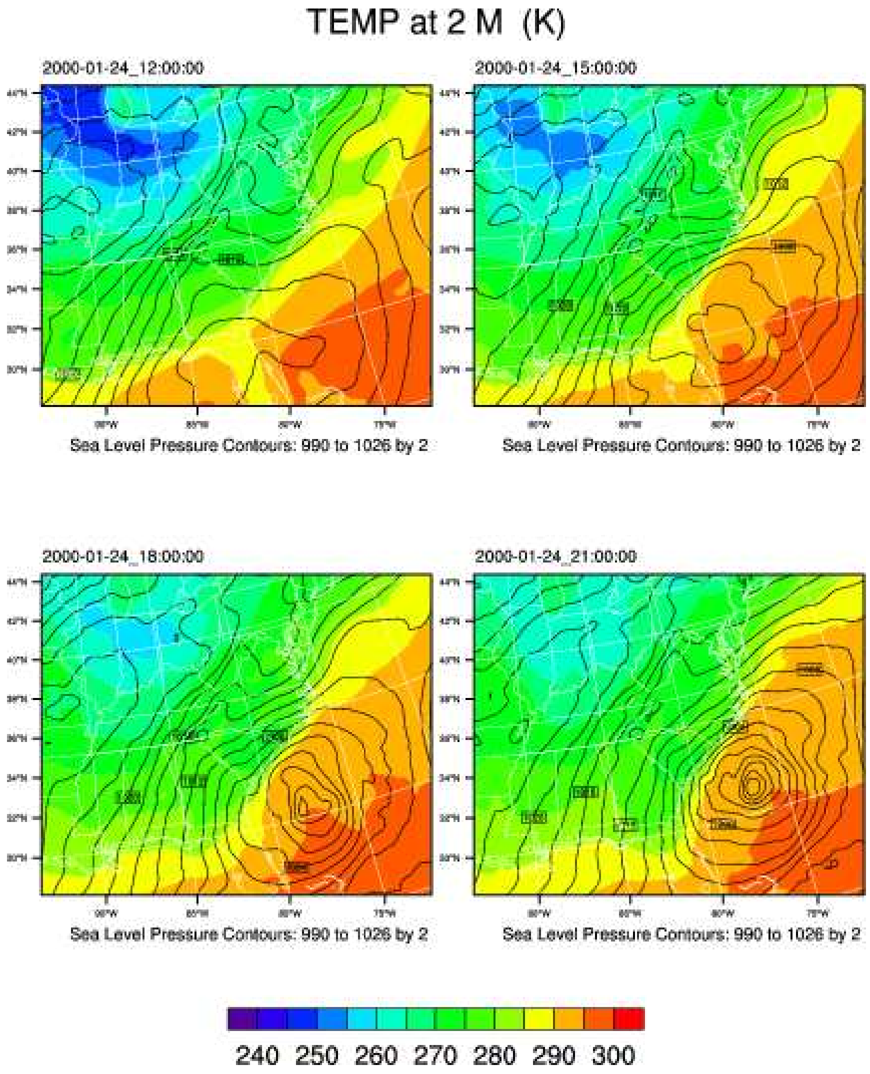

Relative Humidity (%)
Temperature (C)
Main
Temperature Contours: -45 to 25 by 5
Height (km)
latitude
27.0 29.2 31.4 33.7 35.9 38.1 40.3 42.5
Relative Humidity (%)
10 20 30 40 50 60 70 80 90


Surface temperature along a cross section
24_12
24_15
24_18
24_21
25_00
TEMP at 2 M (F)
longitute
-96.0 -92.0 -88.0 -84.0 -80.0 -76.0 -72.0 -68.0

Examples

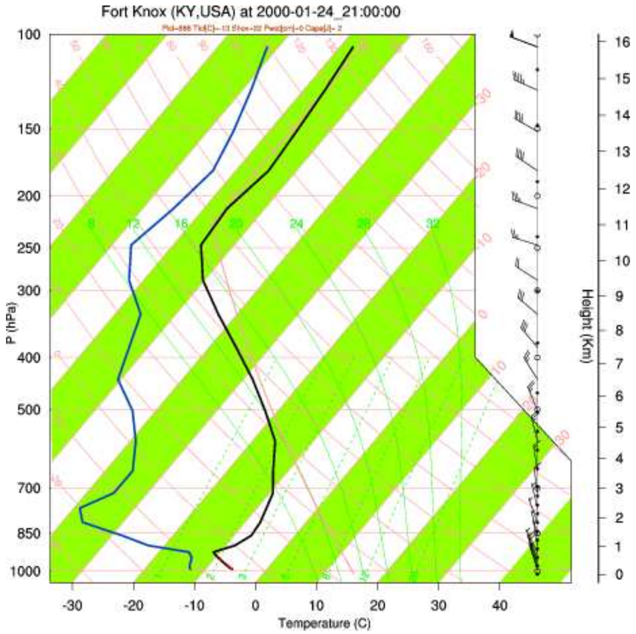


OLR (W m-2)
32°N
30°N
28°N
26°N
24°N
22°N
20°N
18°N
100°W 95°W 90°W 85°W 80°W
100 120 140 160 180 200 220 240 260 280 300 320 340
OLR (W m-2)
30°N
28°N
26°N
24°N
22°N
92°W 90°W 88°W 86°W 84°W 82°W
100 120 140 160 180 200 220 240 260 280 300 320 340

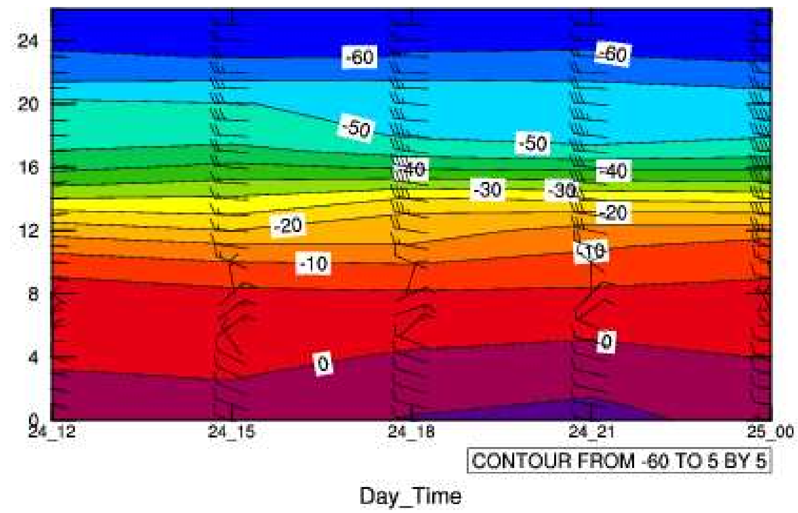


TEMP at 2 M (K)
2000-01-24_12:00:00
2000-01-24_15:00:00
Sea Level Pressure Contours: 990 to 1026 by 2
Sea Level Pressure Contours: 990 to 1026 by 2
2000-01-24_18:00:00
2000-01-24_21:00:00
Sea Level Pressure Contours: 990 to 1026 by 2
Sea Level Pressure Contours: 990 to 1026 by 2
240 250 260 270 280 290 300

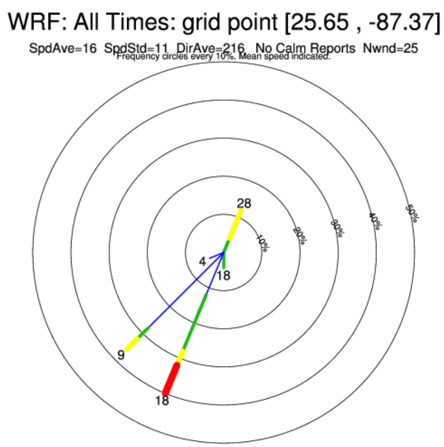
Examples



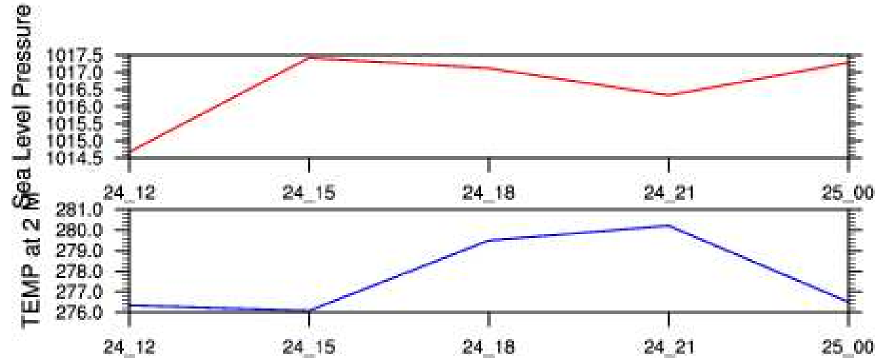
Fort Knox (KY,USA) at 2000-01-24_21:00:00
P (hPa)
Temperature (C)
Height (Km)



CONTOUR FROM -60 TO 5 BY 5
Day_Time

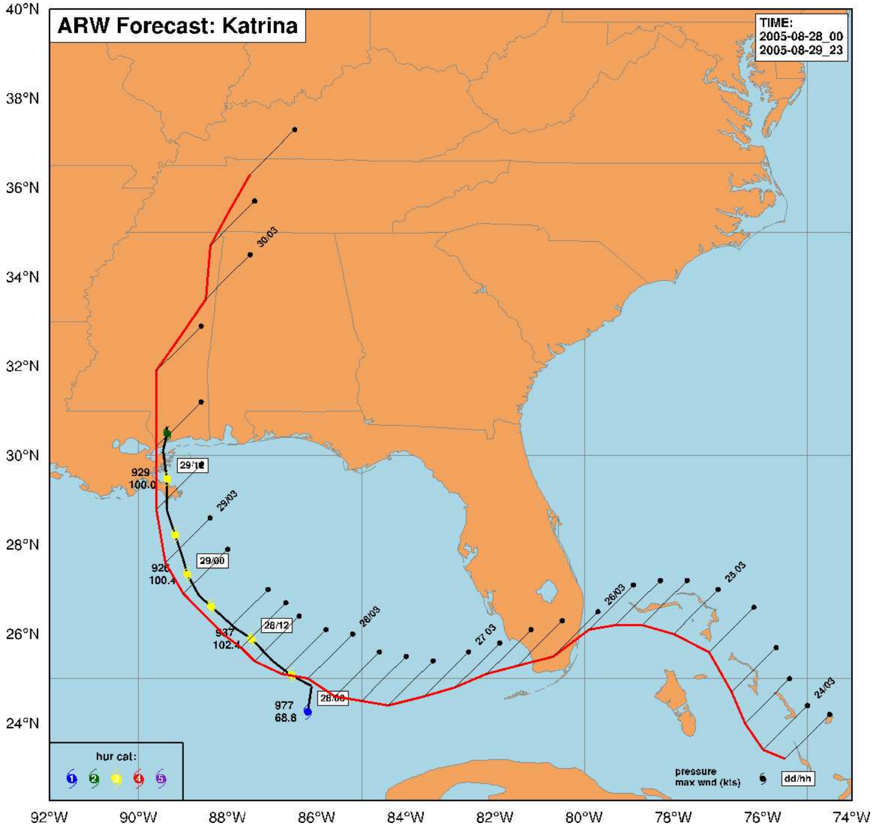


WRF: All Times: grid point [25.65 , -87.37]
SpdAve=16 SpdStd=11 DirAve=216 No Calm Reports Nwnd=25
Frequency circles every 10%. Mean speed indicated.

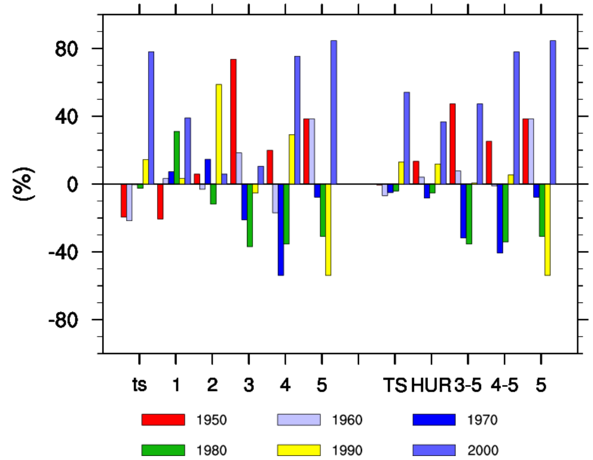


Sea Level Pressure
TEMP at 2 M

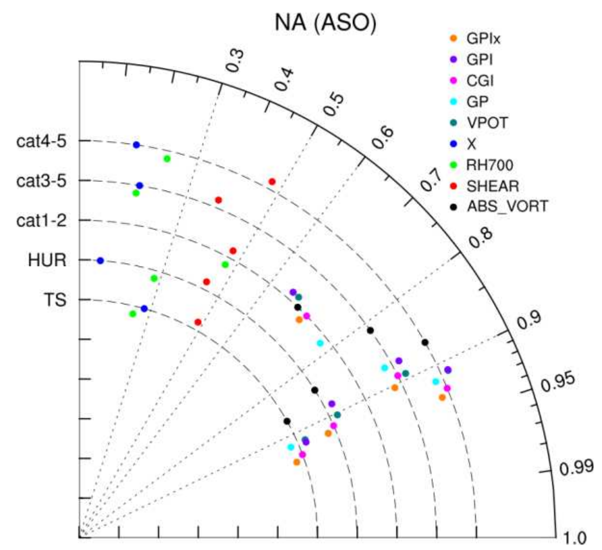
Examples



ARW Forecast: Katrina
TIME:
2005-08-28_00
2005-08-29_23
40°N
38°N
36°N
34°N
32°N
30°N
28°N
26°N
24°N
92°W
90°W
88°W
86°W
84°W
82°W
80°W
78°W
76°W
74°W
hur cat:
pressure
max wnd (kts)
dd/hh



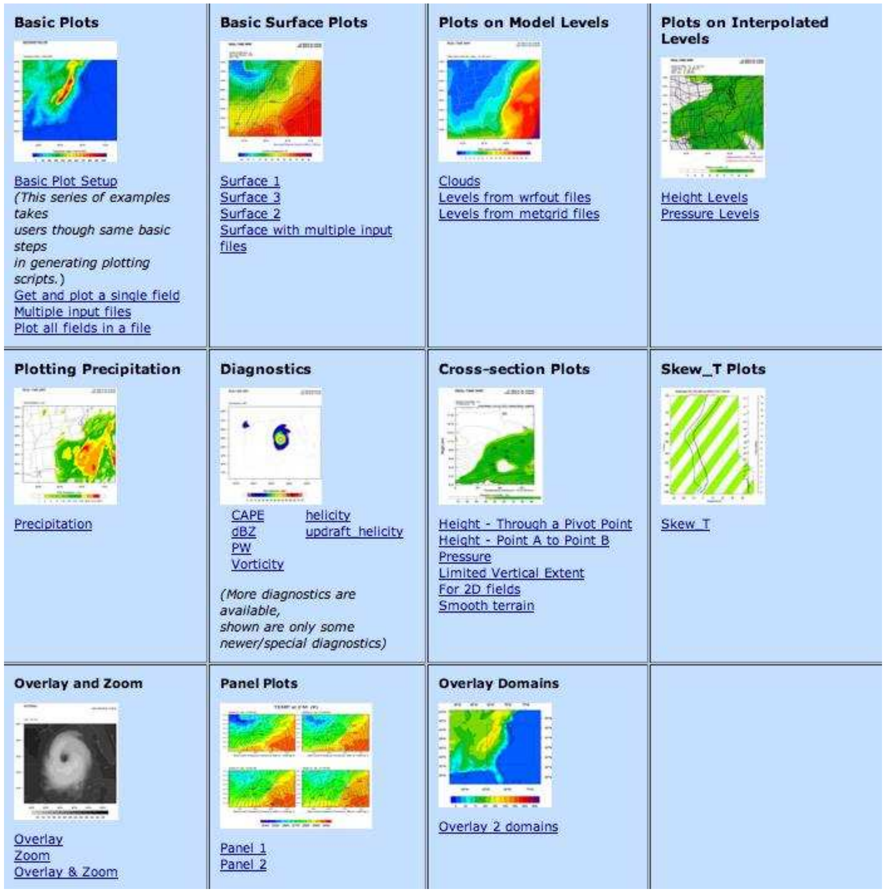
80
40
0
-40
-80
(%)
ts
1
2
3
4
5
TS
HUR
3-5
4-5
5
1950
1960
1970
1980
1990
2000

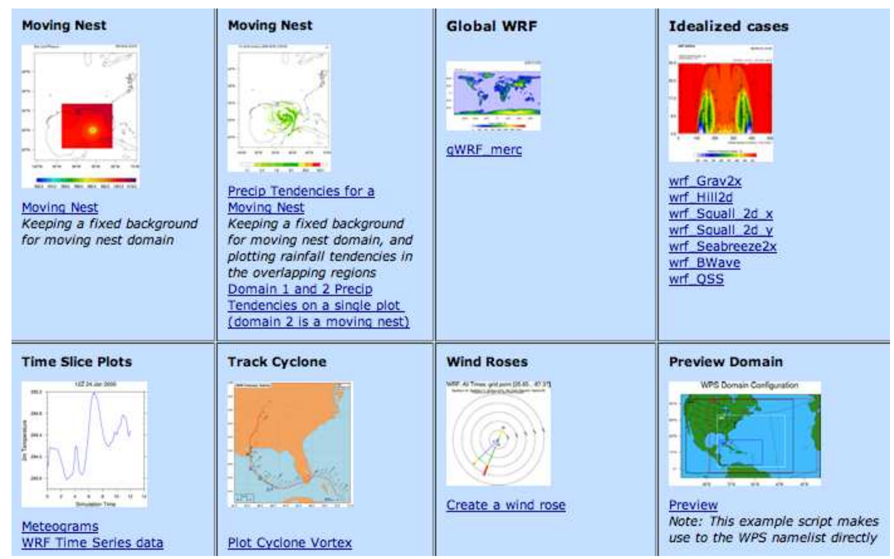


NA (ASO)
GPIx
GPI
CGI
GP
VPOT
X
RH700
SHEAR
ABS_VORT
cat4-5
cat3-5
cat1-2
HUR
TS
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0.95
0.99
1.0

Generate Plots: *A good start - OnLine Tutorial*

http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/index.html





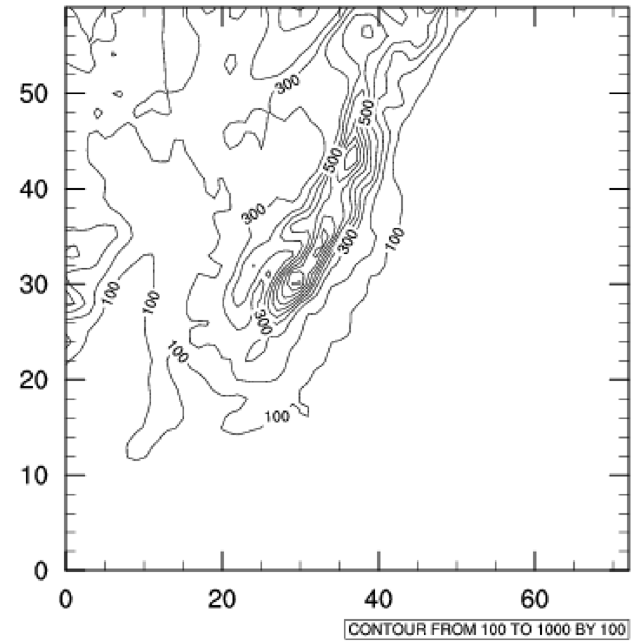
Creating a Plot : NCL script

```
load ncl library scripts

begin

  ; Open graphical output
  ; Open input file(s)

  ; Read variables

  ; Set up plot resources & Create plots
  ; Output graphics

end
```

Generate Plots

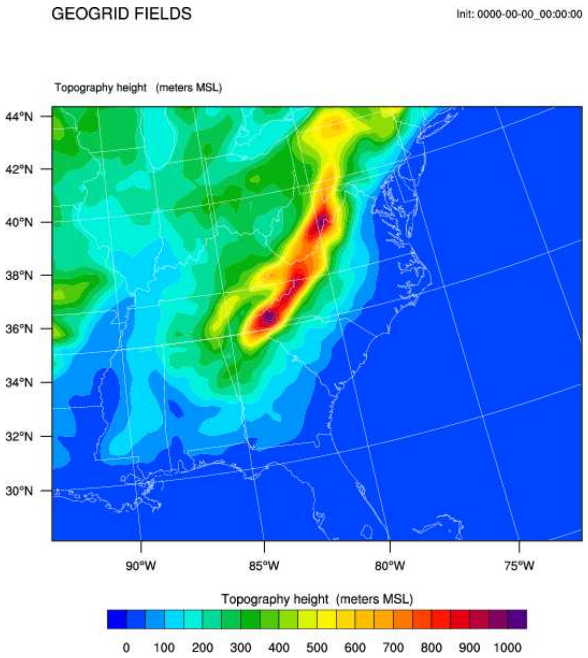
```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"

begin
  a = addfile("./geo_em.d01.nc","r")
  wks = gsn_open_wks("pdf","plt_ter1")
  ter = a->HGT_M(0,:,:)
  plot = gsn_contour(wks,ter,True)
end
```

```
> ls wrfout*
  wrfout_d01_2005-10-08_00:00:00

a = addfile("./wrfout_d01_2005-10-
08_00:00:00.nc","r")
```

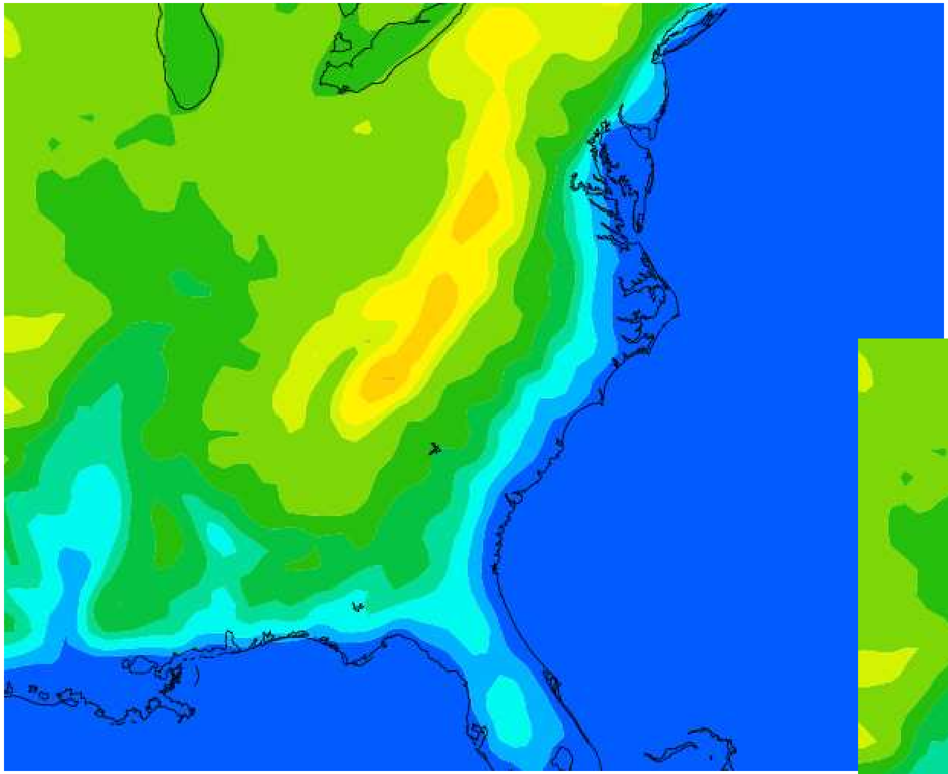


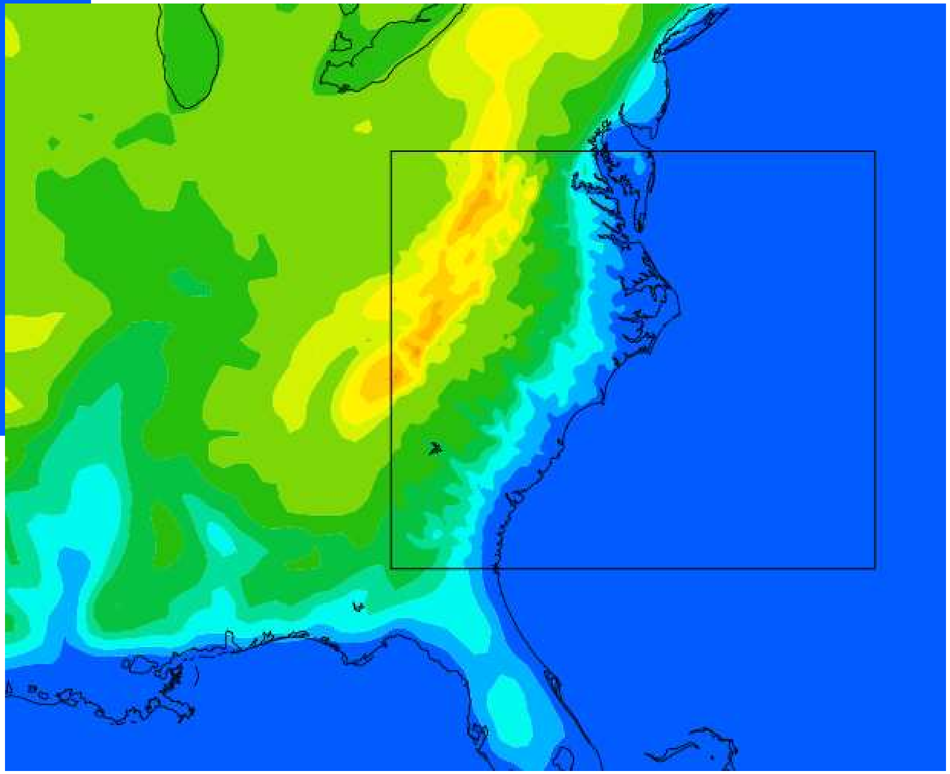
Generate Plots

```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/wrf/WRFUserARW.ncl"

begin
  a = addfile("./geo_em.d01.nc","r")
  wks = gsn_open_wks("pdf","plt_ter1")
  ter = a->HGT_M(0,:,:)

  res = True
  res@MainTitle = "GEOGRID FIELDS"
  res@cnFillOn = True
  res@ContourParameters = (/0.,1000.,50./)
  contour = wrf_contour(a,wks,ter,res)
  pltres = True
  mpres = True
  plot = wrf_map_overlays(a,wks,(/contour/),\
    pltres,mpres)
end
```

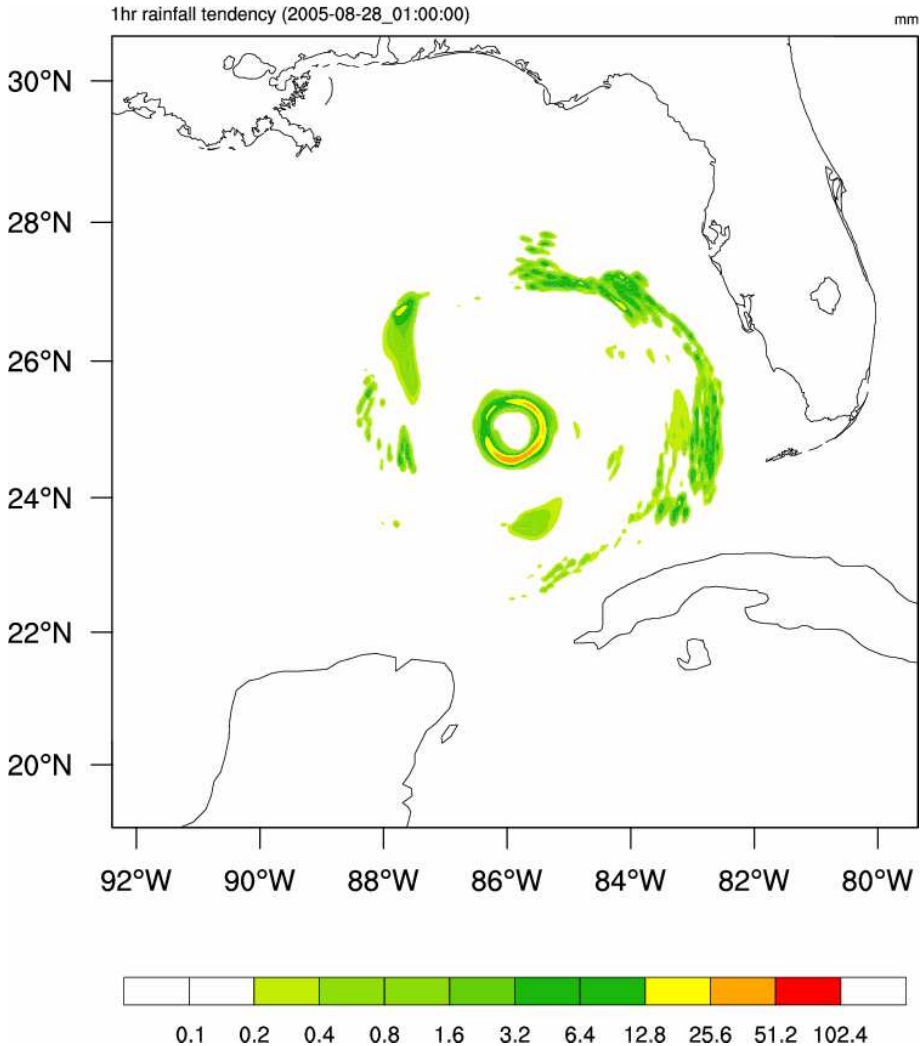


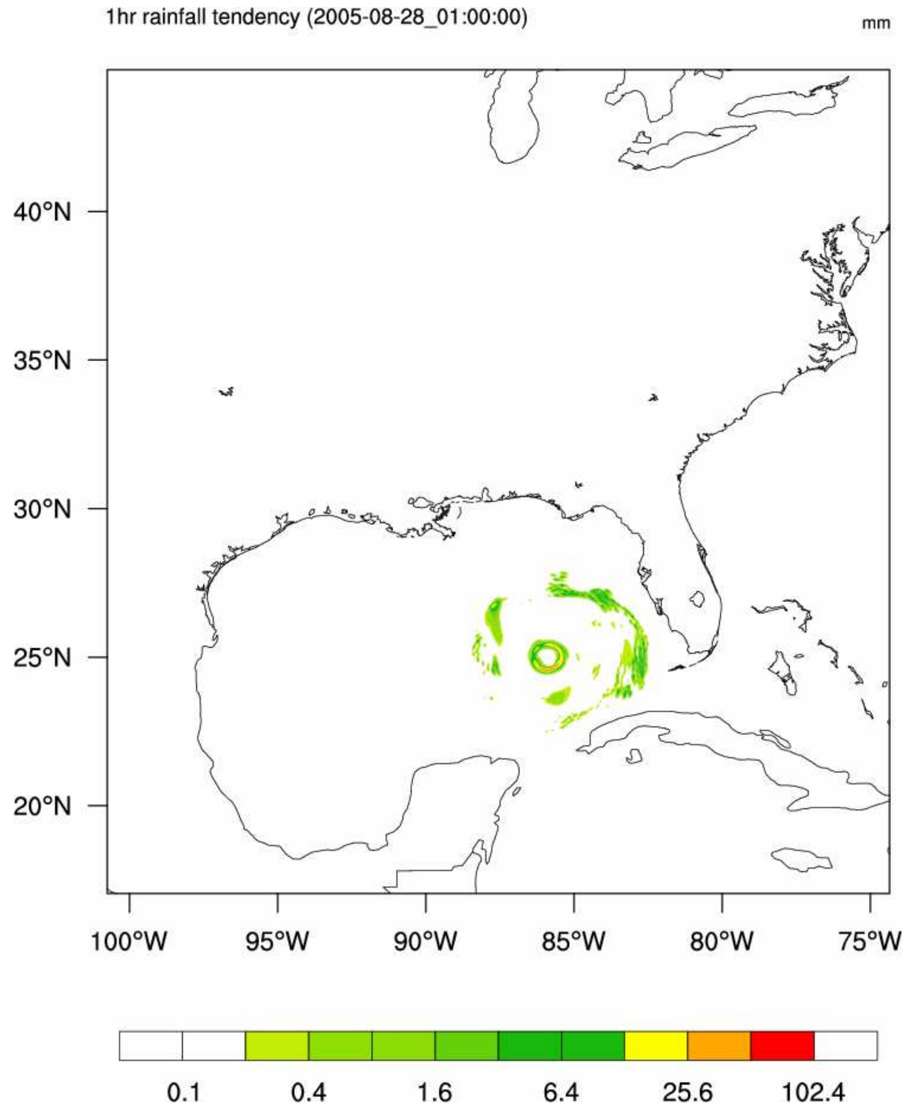
Over Domains





Moving Nests

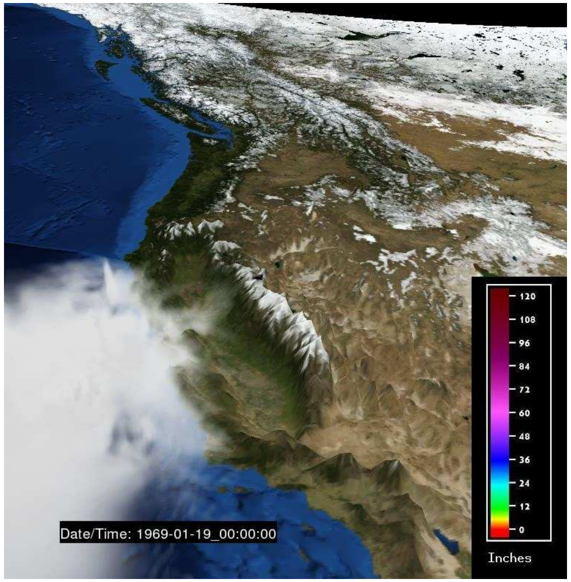


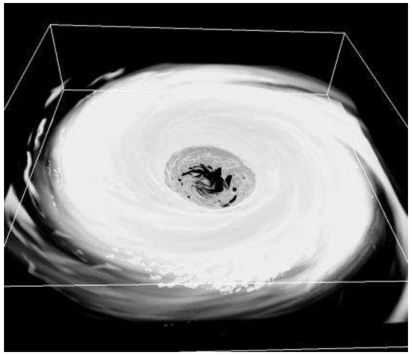


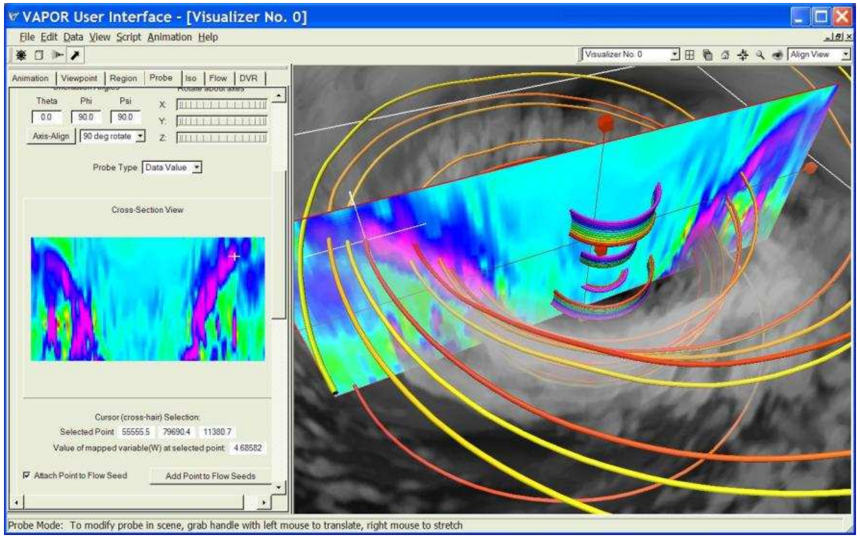
Visualization and Analysis Platform for Oceanic, atmospheric and solar Research

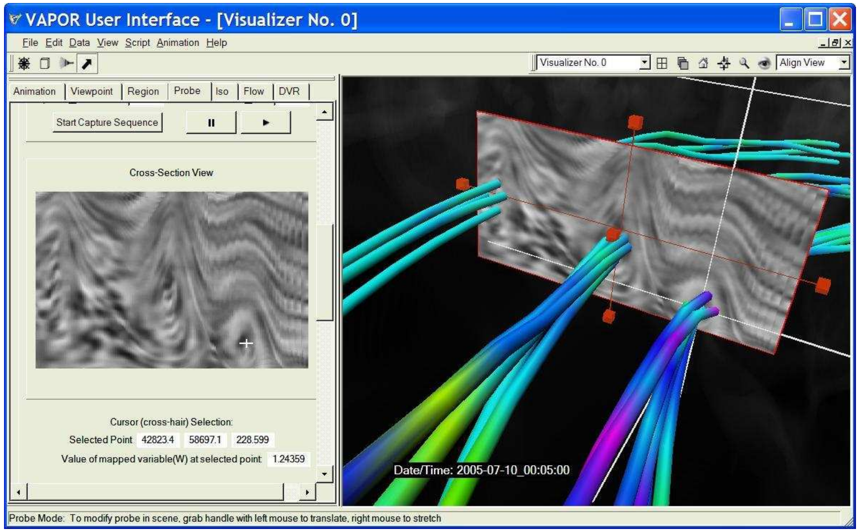
Alan Norton
alan@ucar.edu
vapor@ucar.edu
National Center for Atmospheric Research

VAPOR visualization of WRF-ARW data









Integrated Data Viewer

Yuan Ho and Julien Chastang
Unidata Program Center/UCAR

Supported Data Sources

- Data Types:
  - Gridded model output
  - Satellite imagery
  - Radar data
  - Point observations
  - Balloon soundings
  - NOAA Profiler Network winds
  - Aircraft Tracks
  - Fronts
  - GIS data (WMS, shapefile)
  - Quick Time movies
  - Web Cams
- Vertical Coordinates
  - Pressure
  - Height/Depth
  - Other (2D only)
- Sample of Supported Formats:
  - netCDF
  - GRIB
  - Vis5D
  - KML
  - CSV
  - GEMPAK grid
  - ADDE
- Access Methods:
  - Local files
  - HTTP
  - ADDE, TDS and OPeNDAP servers
  - WMS

ADDE = Abstract Data Distribution Environment
TDS (THREDDS) = Thematic Realtime Environmental Distributed Data Services

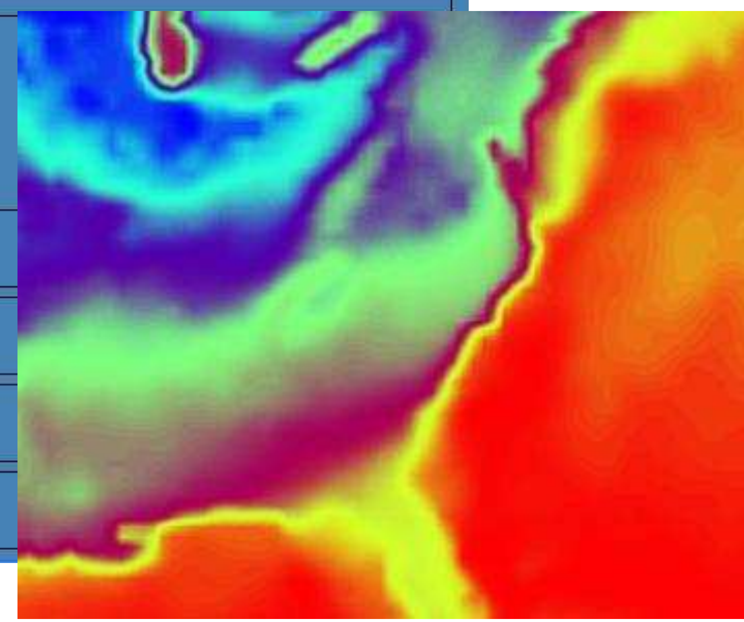
Wrfout output fields *(ncdump -h)*

ALBBCK	ALBEDO	CANWAT	CF1	CF2
CF3	CFN	CFN1	COSALPHA	DN
DNW	DZS	E	EDT_OUT	EMISS
F	FNM	FNP	GLW	GRAUPELNC
GRDFLX	HFX	HGT	HGT_SHAD	ISLTYP
ISTEP	IVGTYP	LANDMASK	LH	LU_INDEX
MAPFAC_M	MAPFAC_MX	MAPFAC_MY	MAPFAC_U	MAPFAC_UX
MAPFAC_UY	MAPFAC_V	MAPFAC_VX	MAPFAC_VY	MAX_MSTFX
MAX_MSTFY	MF_VX_INV	MU	MUB	NEST_POS
OLR	P_TOP	**P**	**PB**	PBLH
PH	**PHB**	POTEVP	PRATEC	PSFC
Q2	QCLOUD	QFX	QNDROPSOURCE	QRAIN
QVAPOR	RAINC	RAINCV	RAINNC	RDN
RDNW	RDX	RDY	RESM	RHOSN
SEAICE	SFROFF	SH2O	SINALPHA	SMOIS
SNOPCX	SNOW	SNOWC	SNOWH	SNOWNC
SOILTB	SR	SST	SWDOWN	**T**
T2	TH2	Times	TMN	TSK
TSLB	**U**	**U10**	UDROFF	UST
V	**V10**	VEGFRA	**W**	X
XICEM	XLAND	XLAT	XLAT_U	XLAT_V
XLONG	XLONG_U	XLONG_V	ZETATOP	ZNU
ZNW	ZS			

Total Geopotential, staggered (PH+PHB)	Total Pressure in Pa (P+PB)
Wind components, grid relative, staggered (U & V)	Total Potential Temperature (T+300)
10m wind components, grid relative, mass points (U10 & V10)	Surface temperature in K (T2)

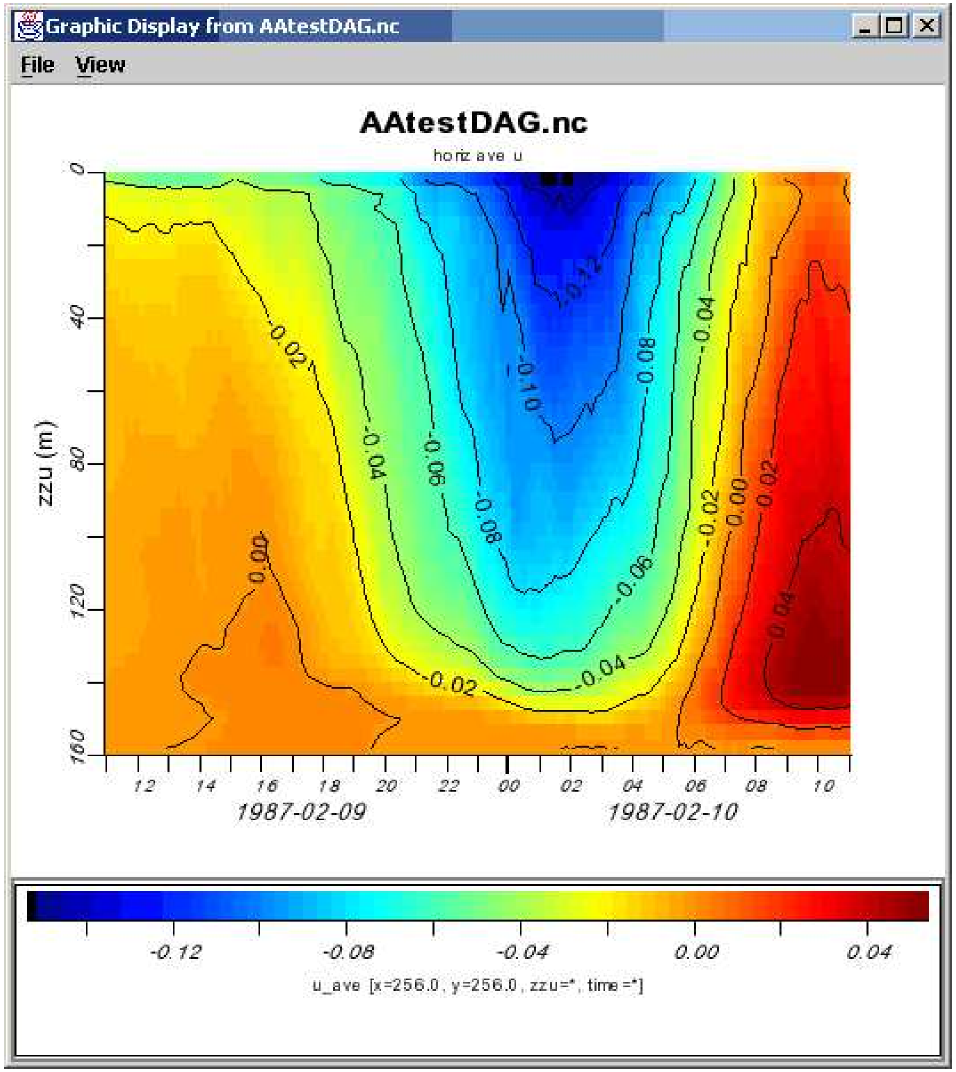
ncview

http://meteora.ucsd.edu/~pierce/ncview_home_page.html



ncBrowse

http://www.epic.noaa.gov/java/ncBrowse/



Panoply

http://www.giss.nasa.gov/tools/panoply/