



RIP4 (ARW & NMM)

Cindy Bruyère

RIP4

Read / Interpolate / Plot (Version 4)

Originally written for MM5 input

ARW WRF output: version 4.0

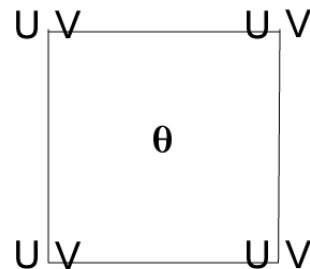
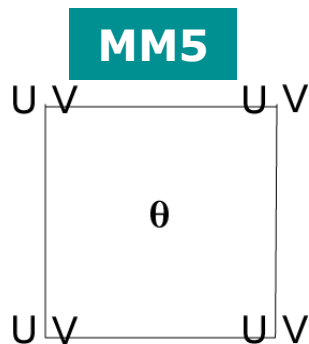
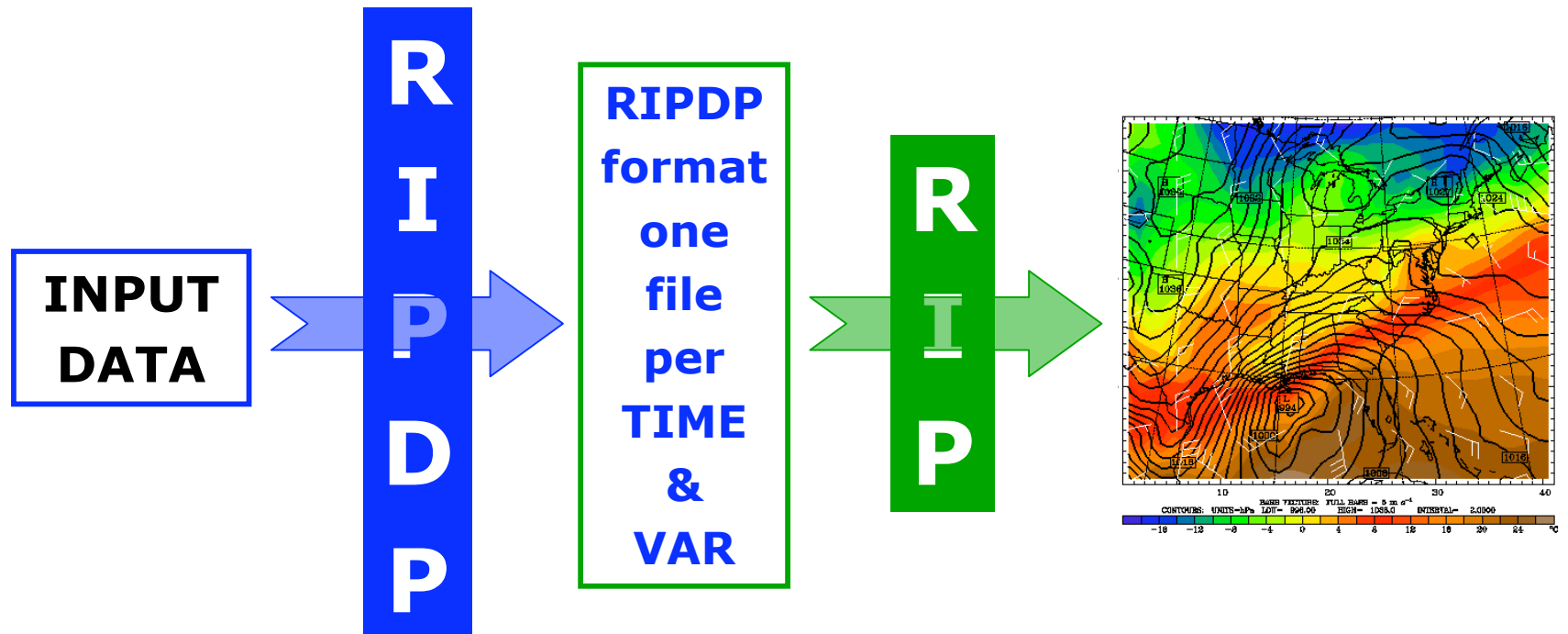
ARW idealized data: version 4.1

WPS (ARW WRF): version 4.2

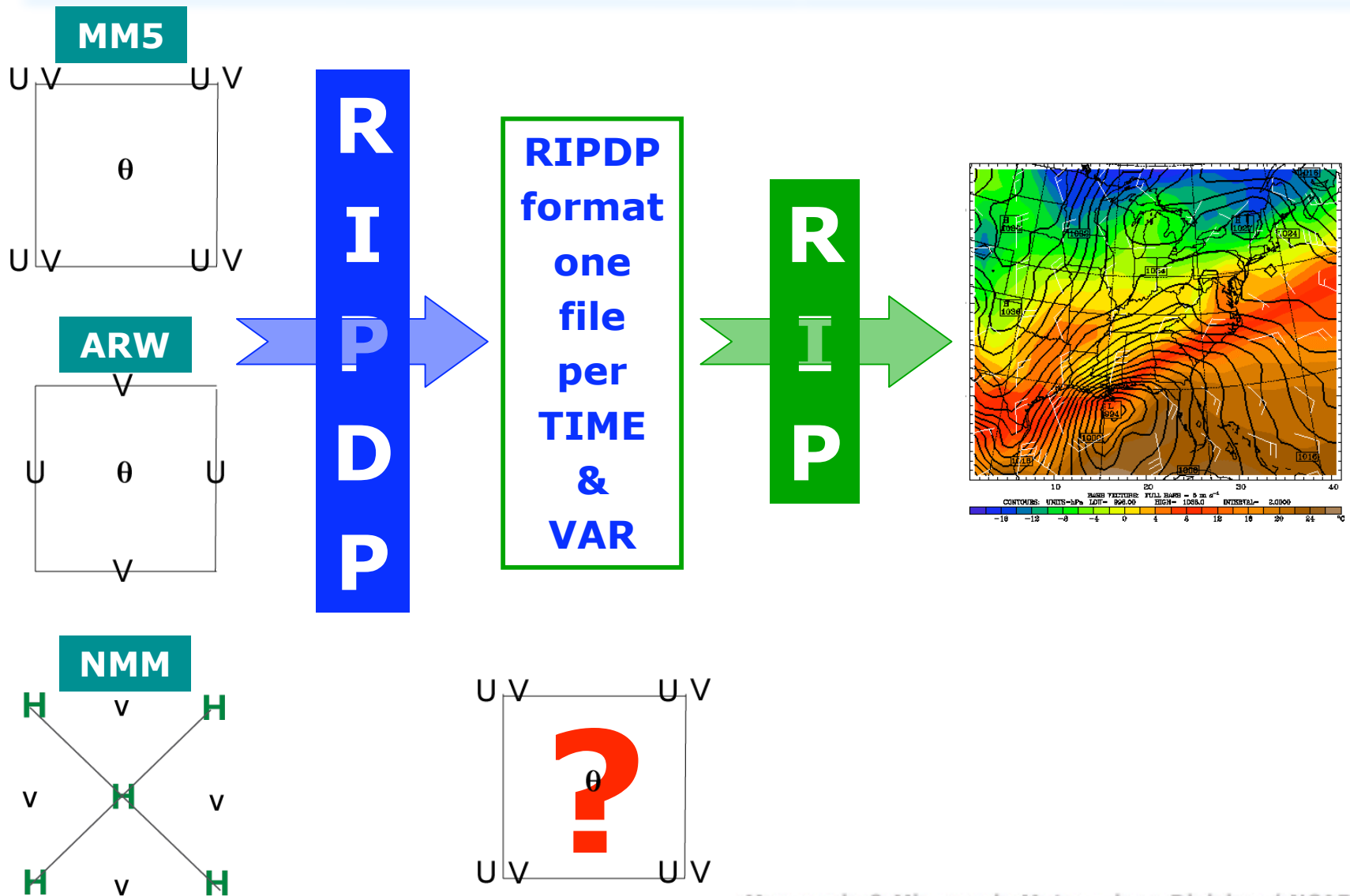
NMM WRF output: version 4.3

Mark Stoelinga (UW/NCAR) & MMM/NCAR Staff

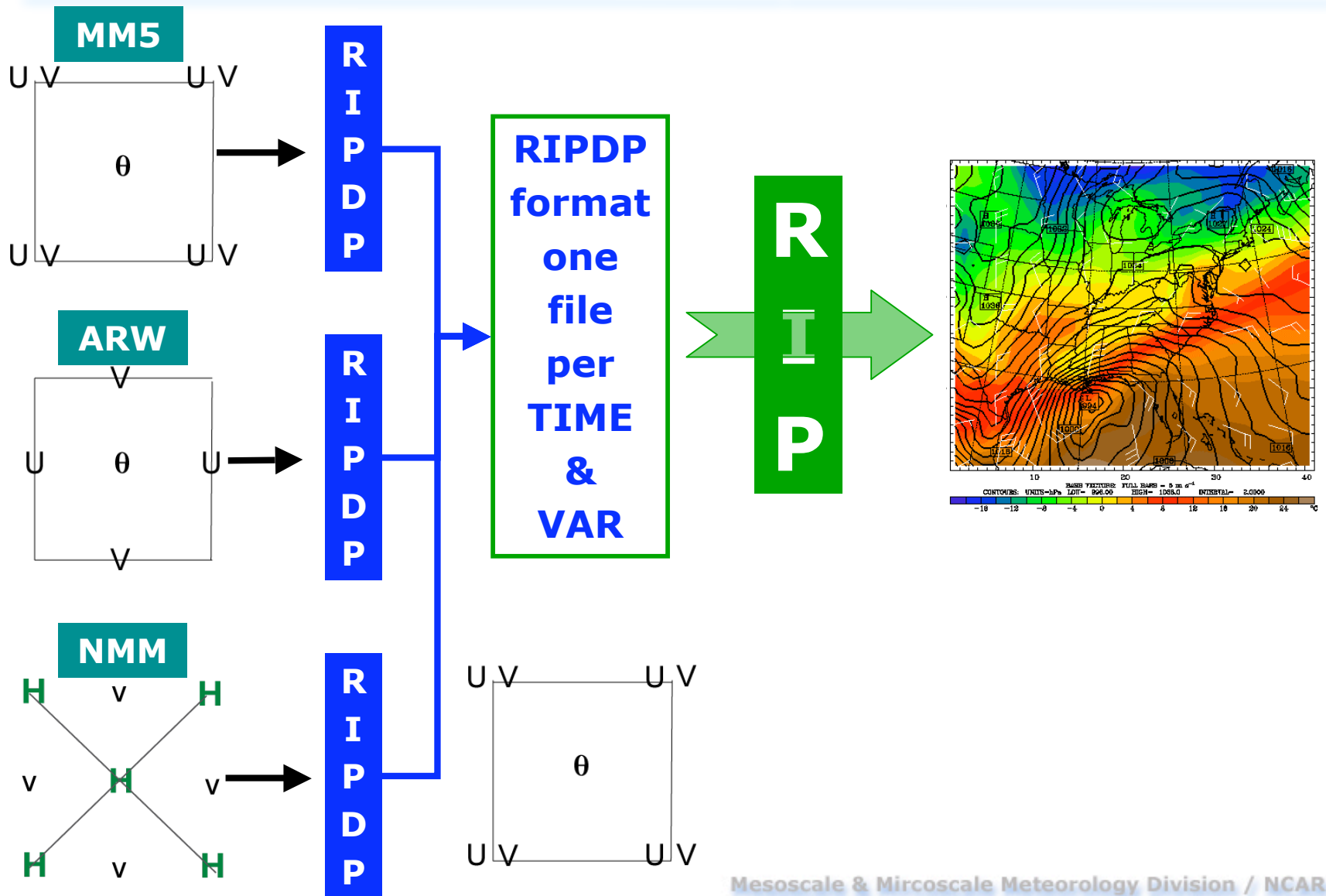
RIP4



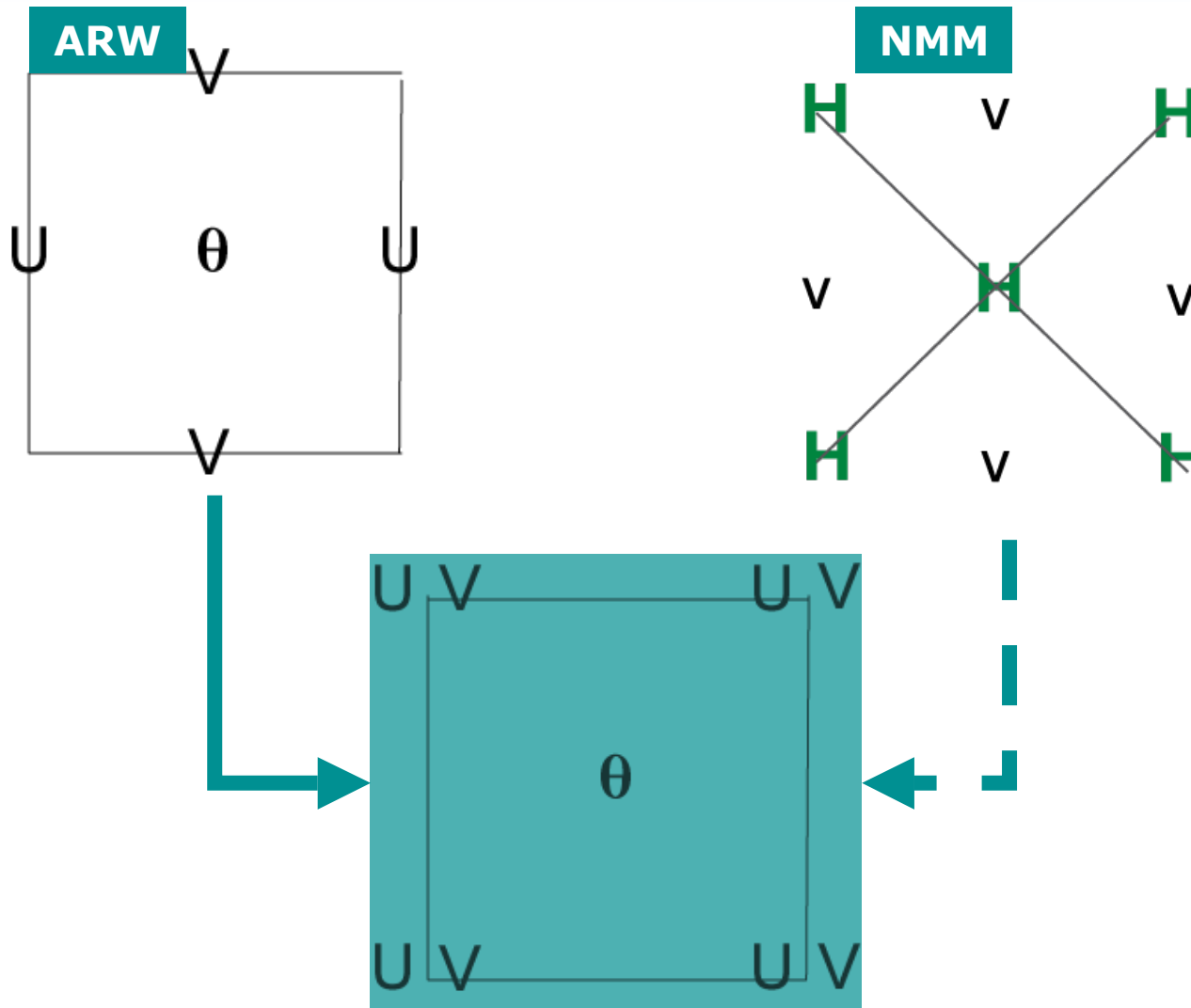
RIP4 - Grids



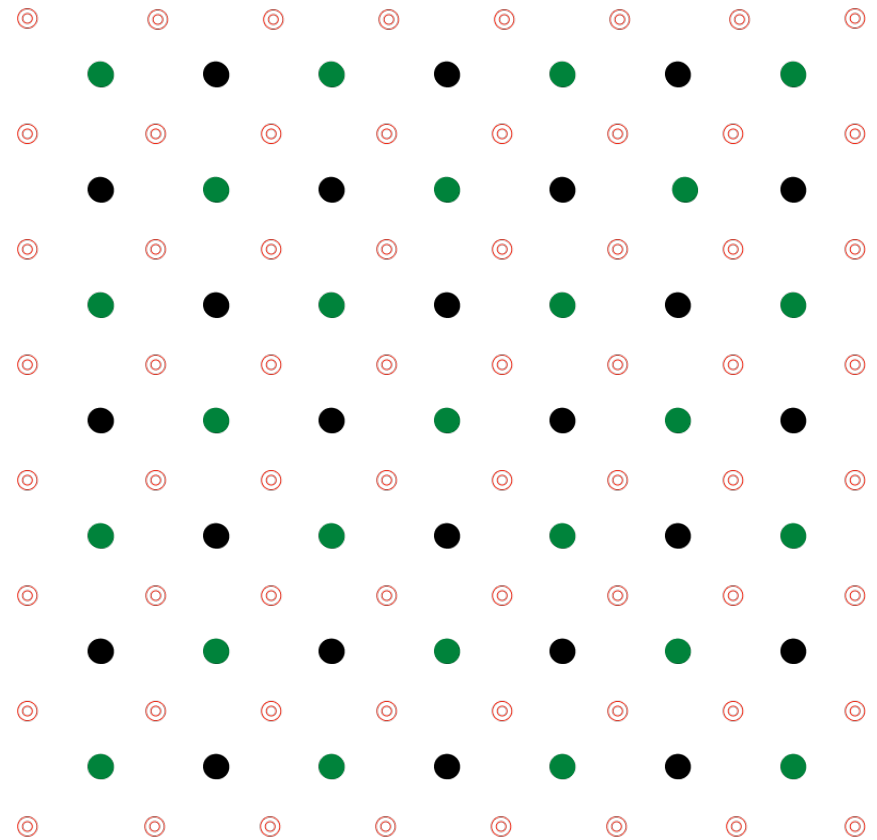
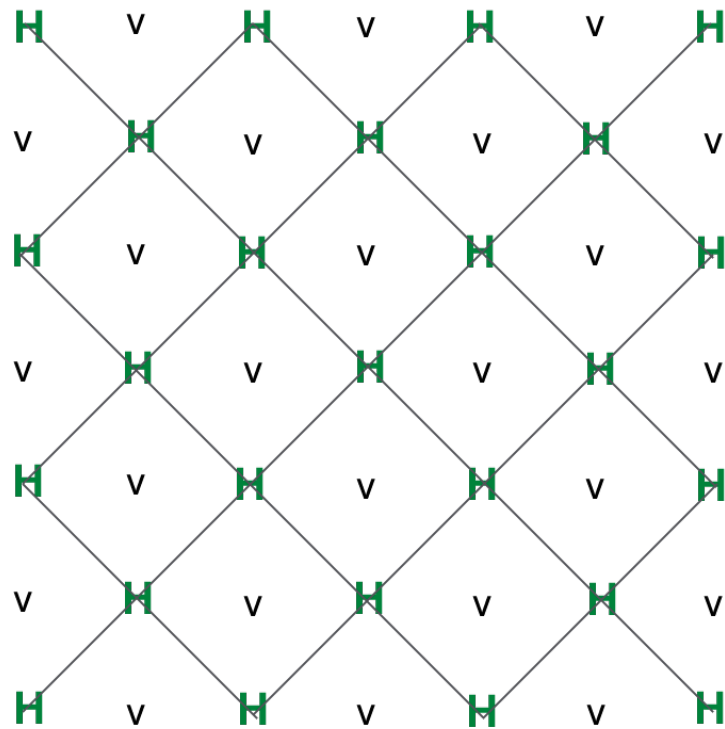
RIP4 - Grids



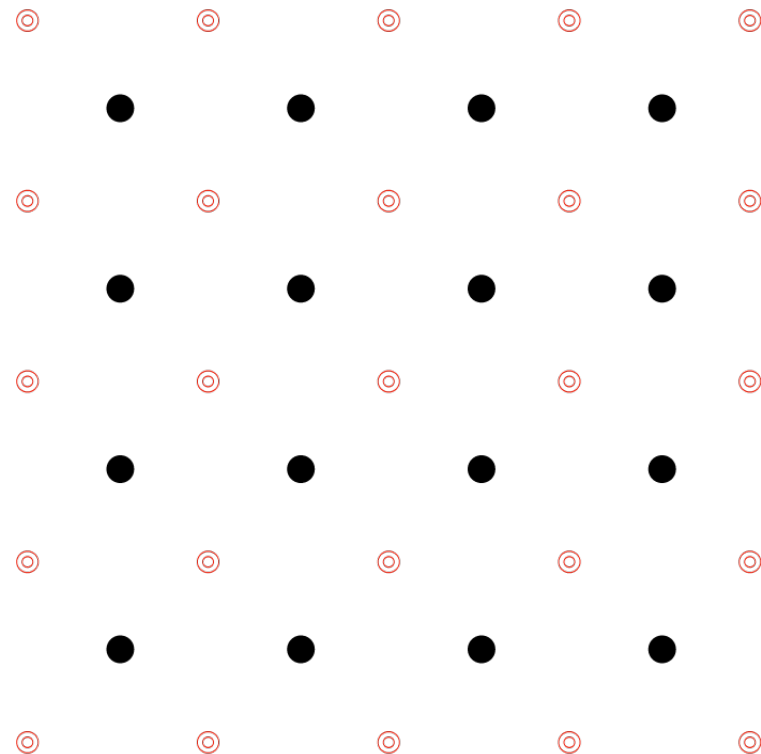
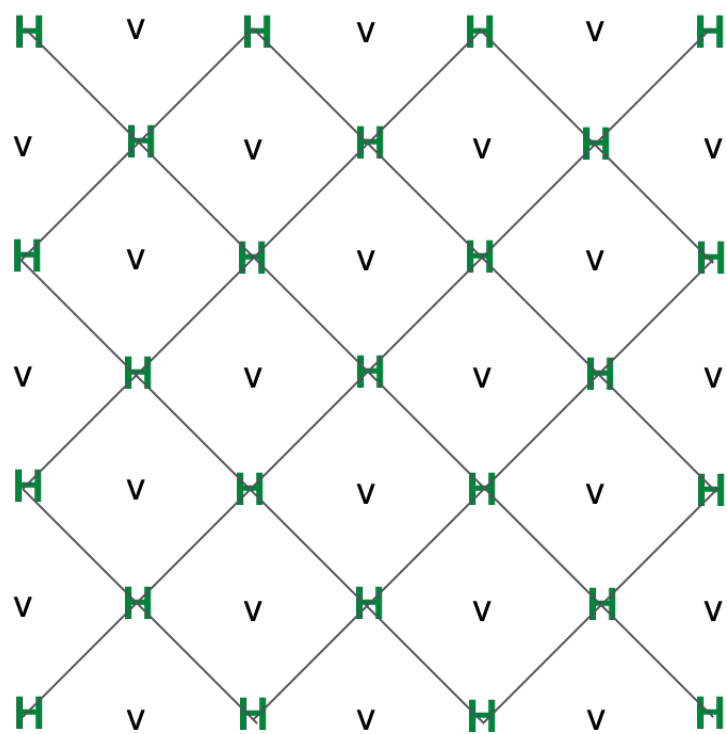
RIP4 - WRF Grids



RIP4 - NMM Grid (*iinterp 0*)



RIP4 - NMM Grid (*iinterp 1*)



new projection ; no direct relationship

general

- **Requires NCAR Graphics low-level routines**
<http://ngwww.ucar.edu>
- **Documentation:**
 - In program tar file under the Doc/ directory
 - <http://www.mmm.ucar.edu/wrf/users/docs/ripug.htm>

download RIP4

- From wrf-model.org web site
(*rip4.tar.gz*):

http://www.mmm.ucar.edu/wrf/users/download/get_source.html

- OnLine Tutorial:

<http://www.mmm.ucar.edu/wrf/users/graphics/RIP4/RIP4.htm>

RIP4 on your computer

- set **RIP_ROOT** environment variable

setenv RIP_ROOT /usr/\$USER/RIP4

- Edit **Makefile** to define paths to netCDF library and include file on your computer:

NETCDFLIB and *NETCDFINC*

- make **<machine type>** (*it'll make suggestions*)

make linux (example)

- RIP4 has 2 parts (**RIPDP** and RIP)

ripdp_mm5

ripdp_wrfarw
ripdp_wrfnmm

ripdp

- **ripdp_wrfxxx**
RIP Data Preparation for WRF (ARW / NMM)
- **RIPDP** converts different input file formats (*WRF - netCDF*) into RIP input format (*B - grid*)
- **RIPDP** puts each **Variable** at each **Time** into a separate file – **LOTS** of files

└─→ **mkdir RIPDP**

running ripdp

Optional

```
ripdp_wrfxxx [-n namelist-file] \
<model_data_name> [basic/all] \
<input_file1 input_file2>
```

Example

```
ripdp_wrfarw RIPDP/arw all wrfout
ripdp_wrfnmm RIPDP/nmm basic wrfout
```

use directory as part of the
model_data_name

ripdp namelist

```
ptimes=0,-72,1,ptimeunits='h',tacc=90.,  
discard='LANDMASK',H2SO4',
```

- **Use namelist to add control**

- **ptimes** (*times for ripdp to process*)
0,1,2,3,4,5,6 (0,1,2,3,4,5,6)
0,-6,1 (0,1,2,3,4,5,6)
0, 2,-4,1, 6 (0, 2,3,4, 6)
- **tacc**: *input files not on exact times*
- **discard**: fields if 'all' is selected on the command line

NMM only

ripdp namelist

- **iinterp = 1:** interpolate to a new B-grid
- **dskmcib:** grid spacing, in km, of the coarse domain on which the new B-grid will be based
- **miycorsib, mjxcorsib:** number of grid points in the y and x directions of new B-grid
- **nprojib:** map projection number (0: none/ideal, 1: LC, 2: PS, 3: ME, 4: SRCE) of new B-grid
- **xlatcib, xloncib:** central latitude and longitude of new B-grid
- **truelat1ib, truelat2ib:** two true latitudes of new B-grid
- **miyib, mjxib:** number of grid points in the y and x directions, of the fine domain
- **yicornib, xjcornib:** coarse domain y and x locations of the lower left corner point of the fine domain
- **dskmib:** grid spacing, in km, of the fine domain

rip

- read the output generated by *ripdp*
- read **User Input File (UIF)** (*rip_sample.in*)
 - **First** section is a list of general parameters (*namelist format*)
 - **Second** section is a series of plots in the Plot Specification Table (PST)
- generate meta file

running rip

- Edit the **User Input File (UIF)**
- **setenv NCARG_ROOT /usr/local/ncarg**
- **setenv RIP_ROOT *your-rip-directory***

running rip

- **rip [-f] model-data-set-name **
rip-execution-name
- created by
ripdp
- User Input File (UIF)

Example

rip [-f] RIPDP/xxx rip_sample.in

use directory as part of the
model_data_set_name

[output]
metacode

[rip_sample.out]
rip_sample.ncgm

rip UIF

```
&userin  
.....  
&end
```

Namelist controlling general parameters

```
&trajcalc  
.....  
&end
```

Namelist for trajectory calculations
Only used if itrajcalc=1, in userin namelist

```
=====
```

```
Plot Specification Table
```

```
=====
```

```
feld= .....  
feld= .....  
=====
```

**Frame specification
group (FSG)**

```
=====
```

```
feld= .....  
feld= .....  
=====
```

Plot specification line (PSL)

```
=====
```

**Plot
Specification
Table (PST)**



rip namelist - userin

- **Use namelist to control**
 - processing times, intervals
 - title information
 - text quality on a plot
 - whether to do time series, trajectory, or to write output for Vis5D
- **Full explanation for namelist variables is available in the user document**

rip namelist - userin

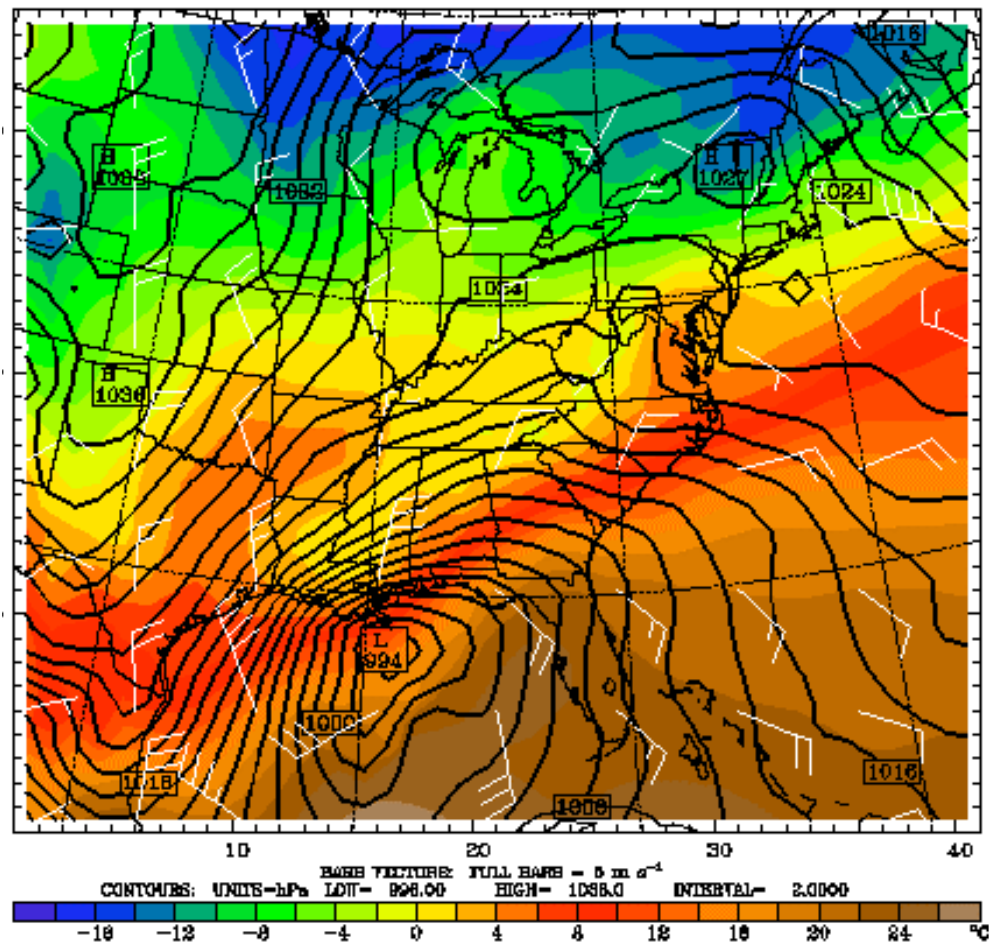
- **idotitle** – first part of first title line
- **titlecolor** – color of title lines
- **ptimes, ptimeunits** – times to process
- **tacc** – tolerance for processing data
- **timezone** –display of local time
- **iusedaylightrule** – 1 applied, 0 not applied
- **iinittime** – plotting of initial time
- **ivalidtime** – plotting of valid time
- **inearsth** – plot times as 2 / 4 digits
- **flmin, frmax, fbmin, ftmax** – frame size
- **ntextq** – text quality

rip namelist - userin

- **ntextcd** – text font
- **fcoffset** – 12 means hour 12 of the MM5 forecast is considered hour 0 by you
- **idotser** – generate time series output
- **idescriptive** – more descriptive titles
- **icgmsplit** – split metacode into several files
- **maxfld** – reserve memory for RIP (10-15)
- **itrajcalc** – 0, 1 ONLY when doing trajectory calculations (*use also namelist trajcalc*)
- **imakev5d** – 0, 1 generate Vis5D data

creating a plot

Temperature @ lowest sigma level
Sea Level Pressure
Winds @ lowest sigma level



creating a plot

```
&userin
```

```
.....
```

```
&end
```

```
&trajcalc
```

```
.....
```

```
&end
```

```
=====
```

```
----- Plot Specification Table -----
```

```
=====
```

```
feld=tmc; ptyp=hc; vcor=s; levs=1fb; >
```

```
  cint=2; cmth=fill; >
```

```
  cosq=32,light.violet,-16,blue, >
```

```
  0,yellow,16,orange,32,light.gray
```

```
feld=slp; ptyp=hc; cint=2; linw=2
```

```
feld=uuu,vvv; ptyp=hv; vcmx=1; >
```

```
  colr=white;intv=5
```

```
feld=map; ptyp=hb
```

```
feld=tic; ptyp=hb
```

```
=====
```


summary: how to run RIP4?

- **Compile the code**
`make <machine type>`
- **Run ripdp_wrfxxx**
Create a new directory for the output
- **Set environment variables**
`setenv NCARG_ROOT /usr/local/ncarg`
`setenv RIP_ROOT your-rip-directory`
- **Edit the User Input File (UIF)**
- **Run rip**