***This is a workflow for generating a real WRF simulation on Meso3 from WPS through visualization with RIP.***

Execute:

ssh meso3.chpc.utah.edu

cd /

cd /compute1/scratch/eneemann/WRFV3

*In separate ssh session (also on meso3 makes it easier to copy files later, if running on scratch space):*

**Get Initial/Boundary Condition Data**

Copy “getnam.csh” file from directory below into your own directory. Edit it for dates you need and download NAM data:

/uufs/chpc.utah.edu/common/home/horel-group3/horel\_data/eneemann/Uintah\_Basin\_Runs/WPS\_Feb\_2013

Execute: ./getnam.csh

\*\*\*you now have the NAM data you need for your simulation\*\*\*

- check to see that all files are in your directory

**Set Environment & Compile WRF**

Navigate to WRFV3 folder

Execute:

./clean –a

source /uufs/chpc.utah.edu/sys/pkg/pgi/std\_rh6/etc/pgi.csh

source /uufs/chpc.utah.edu/sys/pkg/mpich2/std\_pgi\_rh6/etc/mpich2.csh

setenv NETCDF /uufs/chpc.utah.edu/sys/pkg/netcdf/4.0pg

setenv WRF\_EM\_CORE 1

setenv WRFIO\_NCD\_LARGE\_FILE\_SUPPORT 1

./configure

\*\*\*select configuration option 7, then 1 for nesting\*\*\*

./compile em\_real >& out.txt

vi out.txt

\*\*\*check out.txt file

* read bottom lines to ensure compile was completed (should say “build completed”)
* search for words “fatal” and “severe”; if 0 occurences noted, WRF probably compiled properly
* check WRFV3/run directory and ensure “wrf.exe” and “real.exe” were created

**Write namelists for both WPS and WRF**

Can copy namelist files from path below, then edit for your case

/uufs/chpc.utah.edu/common/home/u0818471/public\_html/Documents/2014/Continuity/WRF\_modified\_code

Copy namelist.wps into your WPS folder

Copy namelist.input into your WRFV*3*/run directory

- Ensure dates, times, domains, dimensions, and geographical information (lat, lon) match in both files.

**Run Geogrid**

- Navigate to the WPS folder of your WRF build

- Execute the command: ./geogrid.exe

- Will take a few minutes

- Open geo\_em.d0*x*.nc (*x* indicates domain)

- Use ncview to open the file with this command: ncview geo\_em.d0*x*.nc

- Look at the variables to ensure they are accurate. Check land use, land mask, the hgt variables, etc. If all is well proceed.

**Run Ungrib**

- Set the correct Vtable for the dataset you are using as boundary conditions

- In the WPS folder link the Vtable to a variable called Vtable with the command:

ln -sf ungrib/Variable\_Tables/Vtable.*xxx* Vtable

(*xxx* indicates the model data source, look in the folder in the link command for all possibilities)

- This will often be “NAM” or “GFS”

- Use the script ./link\_grib.csh to link your real data files (NAM, etc.) to the WPS directory

- The command to run the script is: ./link\_grib.csh filepath/header\*

- Check the folder to ensure you see linked files with the header GRIBFILE.AAA

- Execute the command: ./ungrib.exe

- Will take a few minutes

- When complete you should see:

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

! Successful completion of ungrib. !

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

- Look for files with date time format below that match the dates and times of your data in the WPS folder:

YYYY-MM-DD\_HH

**Run Metgrid**

- Execute the command: ./metgrid.exe

- When complete you should see:

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

! Successful completion of metgrid. !

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

- Look for files with format below that match the domains, dates and times of your data in the WPS folder:

met\_em.d0*x.*YYYY-MM-DD\_HH.nc (*x* indicates domain)

- Open these files with ncview as above to ensure the data looks reasonable, there will be meteorological variables to examine after this step

- Move or copy the met\_em.d0x.YYYY-MM-DD\_HH.nc files from the WPS directory to the WRFV3/run directory

- Navigate to WRFV3/test/em\_real folder and link “met\_em\*” files there from WRFV3/run (test/em\_real is the folder you will run WRF from and the files need to be linked there)

- Execute the command: ln –sf insert\_path\_here/run/met\_em.d0\*.2013-02\* .

\*\*\*example for Feb 2013 case\*\*\*

- Make sure files are linked with “ls –l” command

**Run Real**

- Execute the command: mpirun -np 4 ./real.exe

- Open rsl.out.0000

- Look for errors in the file

- Execute the command: tail rsl.out.000

- should see “SUCCESS COMPLETE REAL…” at bottom

- Look for the files below in the directory

wrfbdy\_d01

wrfinput\_d0*x* (*x* indicates domain)

**Run WRF**

- Delete all rsl files (command: rm rsl\*), they are also written by wrf.exe and the files can be less useful for trouble shooting if wrf.exe tries to overwrite them as opposed to creating them

- Check namelist.input one more time to make sure everything is correct for your WRF run

- Set one more environment variable with the command:

setenv OMP\_NUM\_THREADS 8

- Run WRF with the command:

mpirun -np 15 ./wrf.exe &

- This is the command used on meso3. It runs the wrf on meso3 in parallel over 15 cores, talk to others in the group for specifics on how to run on your compute nodes

- You can check run’s progress by: tail rsl.error.0000

- Check rsl.out.*xxxx* or rsl.error.*xxxx* for errors (*xxxx* indicates node

- As run goes along or when complete, check to see if output files were created: wrfout\_d0*x.*YYYY-MM-DD\_HH:MM:SS

**Run RIPDP**

- Write .in files for RIP

- Move wrfout\_d0*x.*YYYY-MM-DD\_HH:MM:SS(*x* indicates domain)to your RIP directory or specify the path to the wrfout\_d0*x.*YYYY-MM-DD\_HH:MM:SS files in the ripdp command

- Execute the ripdp command: ./ripdp\_wrfarw -n ‘filename.in’ ‘model data set name’ all ‘data file name or path’

- ‘filename.in’ is the name of your .in file, this .in file should only include times

- ‘model data set name’ is the path/name for the files

- ‘data file name or path’ is either the first portion of the wrfout file name including domain or the file path to the wrfout data including domain

- After running RIPDP, check the RIP directory for a list of files with your selected file name

**Run RIP**

- Write.in files for RIP that specify the visualization you want to accomplish

- Execute the rip command: ./rip ‘model data set name’ ‘filename’

- ‘model data set name’ is the path/dataset name for the files

- ‘filename’ is the name of your .in file, no need to add .in