

Installation of / Conversion to WRFV3

1. Decide on a BASEDIR, and make the directory if necessary.
2. **mkdir /path/to/yourBaseDir**
It is strongly recommended that you do *not* use an old directory used with WRFV2
3. Export BASEDIR. I suggest adding this to ~/.bashrc (or similarly for Cshell) as *everything* depends on it.

```
export BASEDIR=/path/to/yourBaseDir
```

4. Ensure that \$BASEDIR/bin is in your PATH
export PATH+=:\$BASEDIR/bin
5. Make sure you have enough space: Note that the unpacked geog is 16GiB! But see the notes at the end.

6. Untar raspGM.tgz, raspGM-bin.tgz, geog.tar.gz and rangs.tgz in \$BASEDIR

```
cd $BASEDIR  
tar xzf raspGM.tgz  
tar xzf raspGM-bin.tgz  
tar xzf geog.tar.gz  
tar xzf rangs.tgz
```

You should have at least

bin	GM	HTML rasp.site.parameters	lib
region.TEMPLATE		rasp.site.runenvironment	RUN.TABLES

7. If you will be designing your own domain, you will need the domain Wizard and Panoply; you may as well install them now. You will need Java6 to run them – install from your neighbourhood repo or from java.com, Then, in \$BASEDIR, do

```
tar xzf Wizard.tgz  
tar xzf Panoply.tgz
```

8. Examine \$BASEDIR/rasp.site.runenvironment. If you have an old version, amend the new copy in the light of this. If you already have the “rangs” database, adjust as necessary. Otherwise, you should untar rangs.tgz in \$BASEDIR

Note also the flag for “CURR_ONLY”. If true, files produced for a run for REGIONXYZ+1 yield files named *param.curr.HHMM1st.d2...* and not *param.curr+1.HHMM1st.d2...*

9. Examine rasp.site.parameters in the light of your old rasp.site.parameters, if you already have this. Note the addition of \$PROXY for curl; uncomment / adjust as necessary.
10. If you are doing a first-time installation from scratch, Go to the section **First Time Installation** below.

If you are converting from a working WRFV2 installation, continue with the steps below.

A) Conversion from a working WRFV2 Installation

1. If you have not already done so, complete all the Steps 1 to 9 above.
2. In directory \$BASEDIR, copy the whole of region.TEMPLATE to REGIONXYZ

```
cd $BASEDIR
cp -a region.template REGIONXYZ
```

3. Copy your old wrfsi.nl for REGIONXYZ to \$BASEDIR/REGIONXYZ
4. In directory REGIONXYZ, run **wrfsi2wps.pl** to make a new namelist.wps.template
cd REGIONXYZ
wrfsi2wps.pl
5. You may need to adjust wrfsi.nl – it tells you what to do. If you change wrfsi.nl, you will also need to copy it to your website if your website uses the cgi-bin utilities ij2latlon.PL or latlon2 ij.PL
6. Run **plotgrids**. This is only a confirmation that your region is what you expect; it is not critical if it should fail.

Ensure that the white line encloses your area of interest, and remember that there must be a five point region inside the white line for RASP, so it needs to be bigger than the actual area of interest – how much bigger depends on the resolution. If change is needed, adjust wrfsi.nl and return to Step 10. You *can* adjust namelist.wps.template, but this is not recommended as it may result in complaints from plotgrids, and you may need an accurate wrfsi.nl for your website.

7. Update namelist.input.template to your new namelist.wps.template by running **wps2input.pl** This also creates namelist.wps for you (needed by geogrid.exe).
8. Run **geogrid.exe** and check that it finishes with
!!
! **Successful completion of geogrid.** !
!!
If there is a problem, also check geogrid.log
9. Copy your old **rasp.run.parameters.REGIONXYZ** to directory REGIONXYZ
10. Copy your old **rasp.region_data.ncl** (AKA **rasp.ncl.region.data**) and **rasp.site_load*.ncl** files from your old NCL (or GM) directory to \$BASEDIR/GM You may wish to also save the old files as examples to help with your mods.
11. There is now a single library for ncl entitled **ncl_jack_fortran.so** which is in \$BASEDIR/GM/LIB

If you insist on running a 32-bit version of ncl, copy DrJack's

ncl_jack_fortran.so and **wrf_user_fortran_util_0.so** to
\$BASEDIR/GM/LIB

12. Run the program(!): do

runGM REGIONXYZ &

Note that you do not need to run this in the REGIONXYZ directory

Check progress with

tail -f \$BASEDIR/REGIONXYZ/LOG/GM.printout

B) First time Installation

1. If you have not already done so, complete all the Steps 1 to 9 at the start of this document.
2. Note the Section below on “Installation Testing”
3. Copy the whole of region.TEMPLATE to YOUR_DOMAIN_NAME

```
cd $BASEDIR  
cp -a region.TEMPLATE/* YOUR_REGION
```

Note the use of the -a flag!

4. Install **DomainWizard** from the RASP archive. Although available on the net, the archive version has been patched for RASP file locations. It is recommended to also install **Panoply**, from the RASP archive or from <https://www.giss.nasa.gov/tools/panoply/>

Note that you will need java6 run-time for both of these utilities. Install from your neighbourhood repo as necessary.

5. Run **domainwizard** to a shell prompt in directory \$BASEDIR to populate your new domain YOUR_REGION.
 - First time out, configure the setup: Select
\$BASEDIR/bin for WPS programs
\$BASEDIR/geog for the Geography database
\$BASEDIR for Domains
 - Select “Open or Delete a domain”. Click Next (at Bottom-Right). Ignore the Error. Select YOUR_REGION in the box at the left. Click Next
 - Locate the locality of your region on the map and Click-Drag to define the region (the MOAD – Mother Of All Domains). Note that this must be larger than the region you actually want.
 - Select “Type” to be Lambert Conformal
 - Play with the values on the right and / or drag the box
 - Click “Update Map”. Note that you cannot adjust any parameters in Projection Options after you have done this, although you can adjust the MOAD. If you want to change the Projection Options, you must “Start Over”, so you may wish to note the current values.
 - Click the “Nests” tab at the top-right and click New and OK. Drag the Nest domain to what you want. Note that the must always be 5 points around it which are inside the MOAD. When you’re happy click Next.

- You can ignore the next screen: click Next.
 - Click “geogrid” under Run / Step 1
 - Ensure that the output finishes with
“*** Successful completion of program geogrid.exe”
 - Quit the program - “Actions | Close”.
 - It all sounds simple :-)
6. In directory YOUR_REGION, move (or copy) namelist.wps to namelist.wps.wzd and similarly for namelist.input

```
cd $BASEDIR/YOUR_REGION
mv namelist.wps namelist.wps.wzd
mv namelist.input namelist.input.wzd
```

WARNING: If you do not do this you may lose your results from domainwizard!

7. Still in this directory, run wzd2wps.pl

```
wzd2wps.pl
```

Note that this will fail if you do not have a namelist.wps.wzd

8. And then run wps2input

```
wps2input.pl
```

9. Edit the file rasp.run.parameters.REGIONXYZ for your region. This will entail:
- Renaming the file to rasp.region.parameters.YOUR_REGION
 - Changing the string “REGIONXYZ” to “YOUR_REGION” throughout the file.
 - Adjusting the GRIB parameters: if you are using subarea GRIBFILES (recommended!), you will need to specify the limits, as indicated. See below for how to determine these.
 - Adjust LSEND & LSAVE as required. It is suggested to set LSEND to 0 for testing.
 - Set the GRIB file list parameters. “0Z” is the “Initialisation Time”; “+3” is the offset from that, so, for example, ‘12Z+18’ is 06Z on the next day.
 - The FORECAST_PERIODHRS is the time from the start of the first GRIBFILE to the end of the last.
10. Edit the file \$BASEDIR/GM/rasp.region_data.ncl Details are given at the head of the file.
11. You may wish to edit \$BASEDIR/rasp.site_load*.ncl at some point.
12. Run the program(!): do
- ```
runGM REGIONXYZ &
```

Note that you do not need to run this in the REGIONXYZ directory.  
Check progress with

```
tail -f $BASEDIR/YOUR_REGION/LOG/GM.printout
```

## Determining Values for sub-region grib files

Panoply can provide accurate values for the extreme values of Latitude and Longitude for your domain. Note that this *must* be for the MOAD (d01)

- Run panoply:

```
panoply $BASEDIR/YOUR_REGION/geo_em.d01.nc
```

- “Create Plot” for some parameter, for example HGT\_M,
- Adjust the map (Map tab at bottom) so that the entire MOAD is visible
- Determine the locations of the extremes of Lat & Lon. Note that this may not be where expected!
- Create a “plot” of XLAT\_M
- Select the “Array 1” tab at the top, adjust the format to “%.3f” at the bottom and determine the max & min latitudes. Note that the layout of the numbers is as on the plot.
- Repeat the above two steps for XLONG\_M
- Round the values obtained to define a ***larger*** region – so further East, further North, etc.
- Use these values in rasp.run.parameters.YOUR\_REGION

If you get missing values at the metgrid stage when running rasp, it probably because you are using GFSA and the area you are requesting for the grib files is not large enough to cover the entire MOAD.

## Installation Testing

If you have never run RASP before, you are strongly advised to run a ready-to-go test region to ensure that all the infrastructure required is in place.

If your region is in the USA, test by running the PANOCHE region. To do this, ship and unpack PANOCHE.tgz in \$BASEDIR and run it:

```
tar xzf PANOCHE.tgz
runGM PANOCHE &
tail -f $BASEDIR/PANOCHE/LOG/GM.printout
```

Otherwise, test by running one of the UK regions; the quickest to run is UK12. To do this, ship and unpack UKregions.tgz in \$BASEDIR and run it:

```
tar xzf UKregions.tgz
runGM UK12 &
tail -f $BASEDIR/UK12/GM.printout
```

## General Notes on RASP – WRFV3 (RASP-V3 ?)

- The directory structure has changed *completely* from that of WRFV2:

All executables are now in are now in \$BASEDIR/bin (although GM.pl is still copied to \$BASEDIR/REGIONXYZ from \$BASEDIR/bin/GM-master.pl)

Everything related to REGIONXYZ is now in \$BASEDIR/REGIONXYZ

- GM.{printout,stdout,stderr} are in \$BASEDIR/REGIONXYZ/LOG
  - GRIB files are in \$BASEDIR/REGIONXYZ/GRIB
  - All NCL logfiles are also in \$BASEDIR/REGIONXYZ/LOG
  - All HTML output is in \$BASEDIR/REGIONXYZ/HTML This is a link to \$BASEDIR/HTML, which could also be a link (say to /var/www/html).
  - The OUT directory (in \$BASEDIR/REGIONXYZ) no longer has a REGIONXYZ subdirectory
- You will need a bucketload of OS-supplied utilities, but runGM checks for them. Be aware that the code requires ncl V6. If you have compiled this and/or it is installed in a non-standard place (e.g. /opt/bin), make sure that this is in your \$PATH. You will still need to install the perl module “Proc::Background”. See Step 5 of <http://www.drjack.info/twiki/bin/view/RASPop/ProgramInstall> if you don’t know how.
  - This code does not do Windowed runs, nor Threaded runs (yet?)
  - You need the “geog” database only if you are building a domain, specifically if you are going to run geogrid.exe. If you have, in \$BASEDIR/REGIONXYZ, the following files in **bold**, *appropriate for your region*, you do not need it:

|                      |                                |
|----------------------|--------------------------------|
| <b>geo_em.d01.nc</b> | <b>namelist.wps</b>            |
| <b>geo_em.d02.nc</b> | <b>namelist.input.template</b> |
| geogrid.log          | <b>namelist.input</b>          |
| wps_show_dom.png     | <b>namelist.input.template</b> |
| wrfsl.nl             |                                |

The non-bold files are optional, but recommended. Note that you can prepare these files on another machine, and ship them to a “production” machine.

- This V3.6.1 setup uses “adaptive time\_step” for wrf.exe. Tests have indicated that this “fully automatic” mode can sometimes crash. The scripts provide a namelist.input.template that limits time\_step to 10\*dx where dx is the domain grid resolution in km. If you still get crashes, you can revert to a manual time\_step setting (set use\_adaptive\_time\_step to “.false” and adjust the value of time\_step). Alternatively you can adjust max\_time\_step.

You may need to read

[http://www2.mmm.ucar.edu/wrf/users/docs/user\\_guide\\_V3/users\\_guide\\_chap5.htm#adaptive](http://www2.mmm.ucar.edu/wrf/users/docs/user_guide_V3/users_guide_chap5.htm#adaptive) (and understand it!) Search “adaptive time step” when viewing that page.

- This version supports ETA and GFSN (0.5 degree resolution only) and GFSA (at 0.25 degree resolution, with all 48 levels – i.e. the main parameters and the additional parameters). GFSA is **highly recommended** for use outside the USA.