Running WPS/geogrid in Docker: A basic guide to generating a WRF-Hydro domain

Programs and data you will need to download and install:

- WPS Docker container: contains the `geogrid.exe` executable you need
- WPS `geog` data: the raw input data needed to create an input file. Can be downloaded here:
  - http://www2.mmm.ucar.edu/wrf/users/download/get_sources_wps_geog.html
  - This download information is for reference only. A subset of the data needed for the training will be provided.

Input you will need to provide:

- Domain Coordinates: These will be used in the WRF Preprocessing System (WPS) steps as input in the file 'namelist.wps' which will then help produce geogrid files which will then be used by WRF-Hydro
  - If you do not already have coordinates, you can get them from the WRF Domain Wizard https://esrl.noaa.gov/gsd/wrfportal/DomainWizard.html
  - It is a graphical user interface (GUI) which enables users to easily define and localize domains (cases) by selecting a region of the Earth and choosing a map projection. Users can also define nests using the nests editor, edit namelist.input, run the WPS programs (geogrid, ungrib, and metgrid) through the GUI, and visualize the NetCDF output.
- For this training, pre-determined domain coordinates will be used. Plus, we will play around a little straight from the namelist.wps.

Workflow

1. Move to the WPS directory and look at what's there:

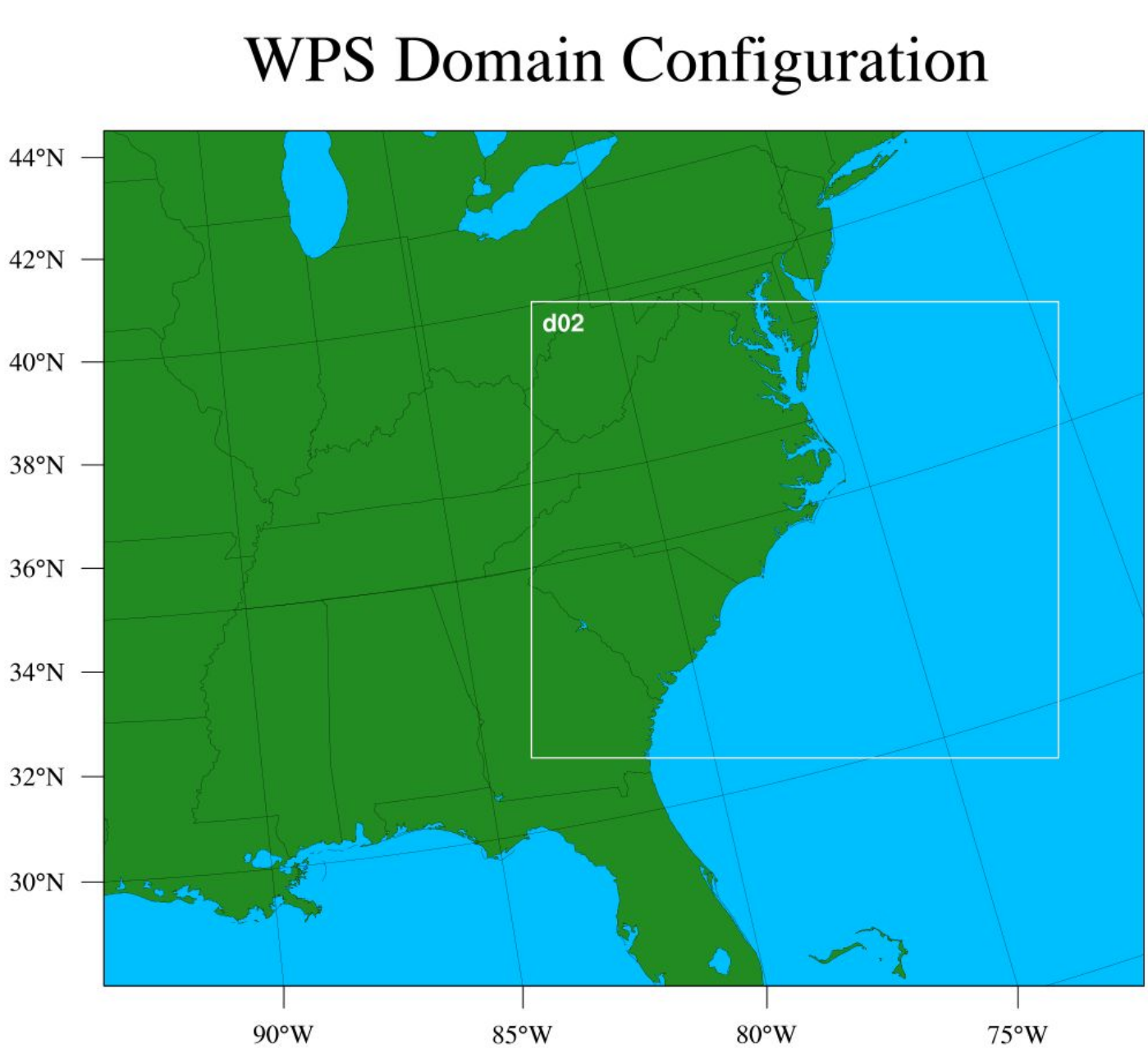
```
cuahsi:~$ cd WRF_WPS/WPS
cuahsi:~/WRF_WPS/WPS$ ls
```

Note three things: geogrid.exe, geogrid/, namelist.wps

```
cuahsi@4383dbf6fc8a:~/WRF_WPS/WPS$ ls
README      configure.wps  log.all                  namelist.wps.global
arch        foo            metgrid                 namelist.wps.nmm
clean       geogrid        namelist.wps            ungrib
compile     geogrid.exe    namelist.wps.all_options  util
configure   link_grib.csh  namelist.wps.fire
cuahsi@4383dbf6fc8a:~/WRF_WPS/WPS$
```

2. Use the available plot tool to view the domain in the default '`namelist.wps`'

```
cuahsi:~/WRF_WPS/WPS$ ncl util/plotgrids_new.ncl
```



You should get a display that looks like above.

3. Take a look at `namelist.wps` and see what modifies the domain

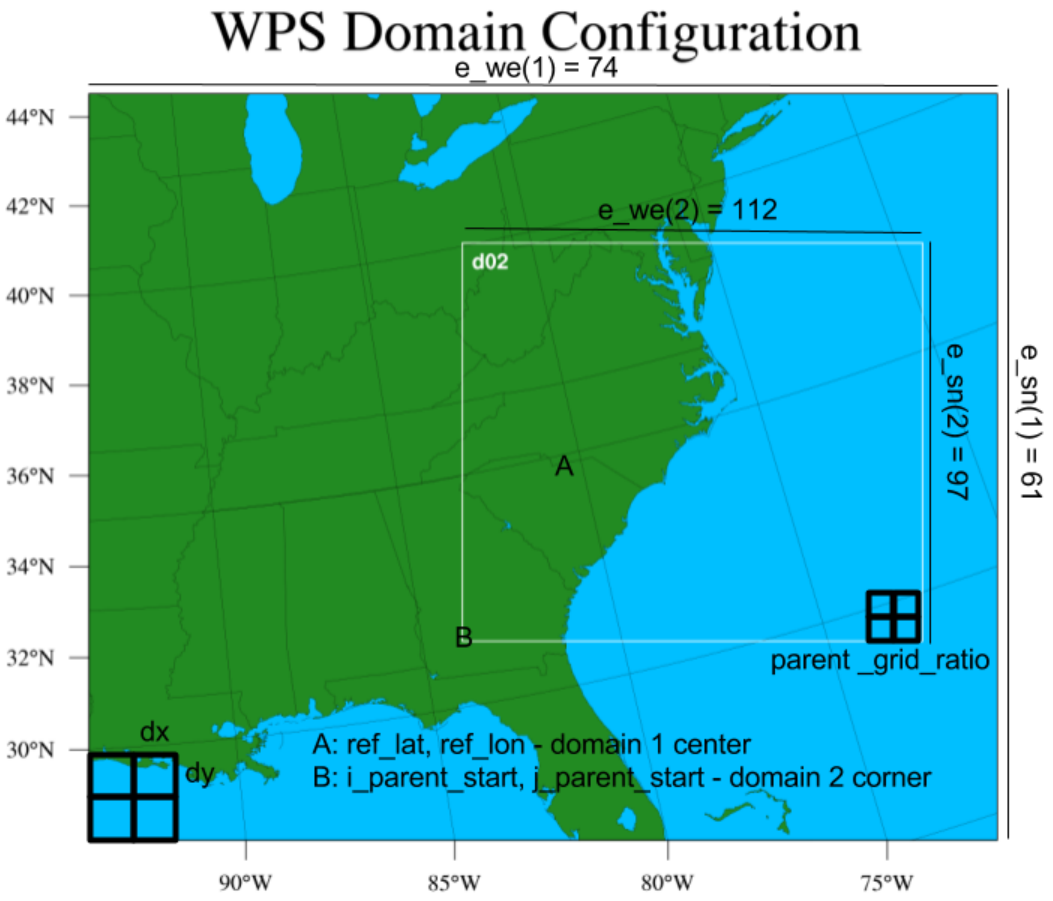
```
cuahsi:~/WRF_WPS/WPS$ more namelist.wps
```

Key components:

`max_dom` : how many domains?
`map_proj, truelat1, truelat2, stand_lon, ref_lat, ref_lon`: map projections definitions
`dx, dy`: model resolution (for lambert, in meters)
`e_we, e_sn`: model dimensions (how many grid points)
`parent_grid_ratio`: for nested domain, how many nested grids per parent grid
`i_parent_start, j_parent_start`: where the lower left corner of nest is relative to parent

```
&share
 wrf_core = 'ARW',
 max_dom = 2,
 start_date = '2006-08-16_12:00:00','2006-08-16_12:00:00',
 end_date   = '2006-08-16_18:00:00','2006-08-16_12:00:00',
 interval_seconds = 21600
 io_form_geogrid = 2,
/

&geogrid
 parent_id         =   1,   1,
 parent_grid_ratio =   1,   3,
 i_parent_start    =   1,  31,
 j_parent_start    =   1,  17,
 e_we              =  74, 112,
 e_sn              =  61,  97,
 !
 geog_data_res = 'default','default',
 dx = 30000,
 dy = 30000,
 map_proj = 'lambert',
 ref_lat   =  34.83,
 ref_lon   = -91.03,
 truelat1  =  30.0,
 truelat2  =  60.0,
 stand_lon = -98.0,
 geog_data_path = '/glade/p/work/wrfhelp/WPS_GEOG/'
/
```



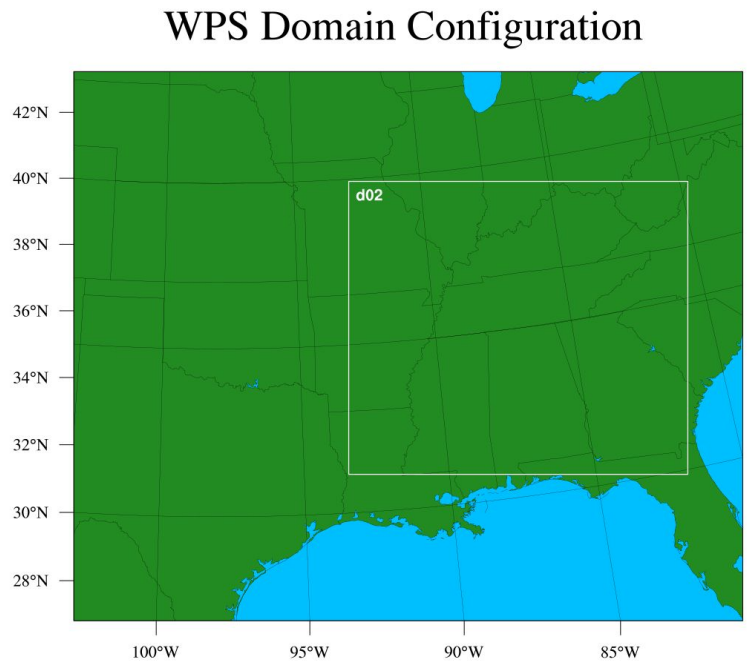
Just for fun, let's modify the `namelist.wps`

```
cuahsi:~/WRF_WPS/WPS$ vi namelist.wps
```

Change `ref_lon` to `-91.03` (for example); entire configuration shifts 10 degrees west.

By using the arrow keys, place the cursor over the '`8`' and press '`r`' followed by '`9`', then save using '`:wq`'. Now, rerun the ncl script:

```
cuahsi:~/WRF_WPS/WPS$ ncl utils/plotgrids_new.ncl
```



4. Set up the test domain environment

```
cuahsi:~/WRF_WPS/WPS$ ln -s ~/ClassMaterials/data/wps_cutout/helper_files
cuahsi:~/WRF_WPS/WPS$ ls helper_files
cuahsi:~/WRF_WPS/WPS$ cp helper_files/namelist.wps.wrf_hydro_training .
cuahsi:~/WRF_WPS/WPS$ cp namelist.wps.wrf_hydro_training namelist.wps
```

Take a quick look:

```
cuahsi:~/WRF_WPS/WPS$ more namelist.wps
```

```
&share
 wrf_core = 'ARW',
 max_dom = 1,
 start_date = '2006-08-16_12:00:00','2006-08-16_12:00:00',
 end_date   = '2006-08-16_18:00:00','2006-08-16_12:00:00',
 interval_seconds = 21600
 io_form_geogrid = 2,
/

&geogrid
 parent_id         =   1,   1,
 parent_grid_ratio =   1,   3,
 i_parent_start    =   1,  31,
 j_parent_start    =   1,  17,
 e_we              =  337, 112,
 e_sn              =  336,  97,
 geog_data_res     = 'usgs_30s+default','2m',
! geog_data_res     = 'usgs_30s+gtopo_30s+default','2m',
 dx = 1000,
 dy = 1000,
 map_proj = 'lambert',
 ref_lat   =   39.52699  !,   39.52697963917897
 ref_lon   = -104.9146   !,  -104.9146023321557
 truelat1  =  30.0,
 truelat2  =  60.0,
 stand_lon = -97.0,
 geog_data_path = '/home/cuahsi/ClassMaterials'
```

Reduce the data to what is needed (only for training):

```
cuahsi:~/WRF_WPS/WPS$ cp helper_files/GEOGRID.TBL.ARW.wrf_hydro_training
geogrid
cuahsi:~/WRF_WPS/WPS$ cd geogrid
cuahsi:~/WRF_WPS/WPS/geogrid$ cp GEOGRID.TBL.ARW.wrf_hydro_training GEOGRID.TBL
cuahsi:~/WRF_WPS/WPS/geogrid$ cd ..
```

5. Create the domain

Run `geogrid.exe`:

```
cuahsi:~/WRF_WPS/WPS$ ./geogrid.exe
cuahsi:~/WRF_WPS/WPS$ ls
```

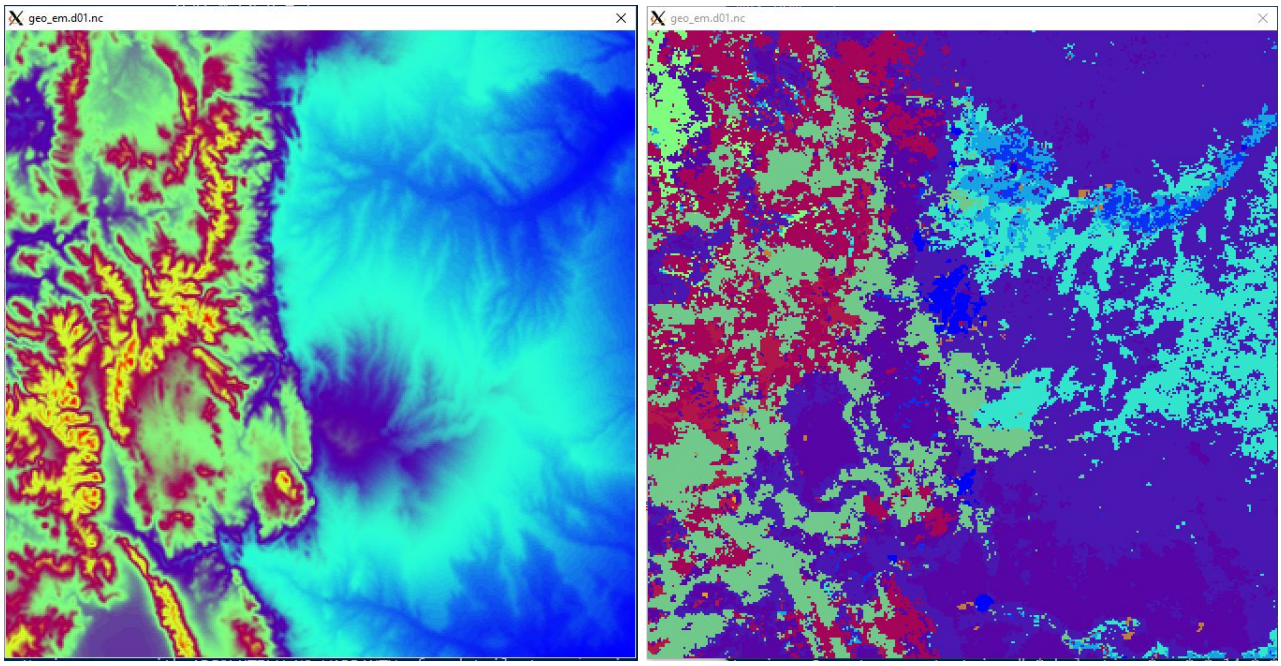
```
cuahsi@58bee99d6a8a:~/WRF_WPS/WPS$ ls -1
README
arch
clean
compile
configure
configure.wps
foo
geo_em.d01.nc
geogrid
geogrid.exe
geogrid.log
```

```
cuahsi:~/WRF_WPS/WPS$ ncdump -h geo_em.d01.nc | more
```

```
netcdf geo_em.d01 {
dimensions:
        Time = UNLIMITED ; // (1 currently)
        DateStrLen = 19 ;
        west_east = 336 ;
        south_north = 335 ;
        south_north_stag = 336 ;
        west_east_stag = 337 ;
        land_cat = 24 ;
        soil_cat = 16 ;
        month = 12 ;
variables:
        char Times(Time, DateStrLen) ;
        float XLAT_M(Time, south_north, west_east) ;
                XLAT_M:FieldType = 104 ;
                XLAT_M:MemoryOrder = "XY " ;
                XLAT_M:units = "degrees latitude" ;
                XLAT_M:description = "Latitude on mass grid" ;
                XLAT_M:stagger = "M" ;
                XLAT_M:sr_x = 1 ;
                XLAT_M:sr_y = 1 ;
        float XLONG_M(Time, south_north, west_east) ;
                XLONG_M:FieldType = 104 ;
                XLONG_M:MemoryOrder = "XY " ;
                XLONG_M:units = "degrees longitude" ;
                XLONG_M:description = "Longitude on mass grid" ;
                XLONG_M:stagger = "M" ;
                XLONG_M:sr_x = 1 ;
                XLONG_M:sr_y = 1 ;
```

```
cuahsi:~/WRF_WPS/WPS$ ncview geo_em.d01.nc
```

Choose `HGT_M`, and then `LU_INDEX`




```
cuahsi:~/WRF_WPS/WPS$ cp geo_em.d01.nc ~/ClassMaterials/data/
```

Important Local Environment Settings

1. Location of the input data

```
cuahsi:~/WRF_WPS/WPS$ more namelist.wps
```

```
map_proj = 'lambert',
ref_lat   =   39.52699  !,   39.52697963917897
ref_lon   = -104.9146   !,  -104.9146023321557
truelat1  =  30.0,
truelat2  =  60.0,
stand_lon = -97.0,
geog_data_path = '/home/cuahsi/ClassMaterials/data/wps_cutout'
/
```

For our training set-up:

```
geog_data_path = '/home/cuahsi/ClassMaterials/data/wps_cutout'
```

Or, in your environment, you may have:

```
geog_data_path = '/d1/barlage/data/geog_data'
```

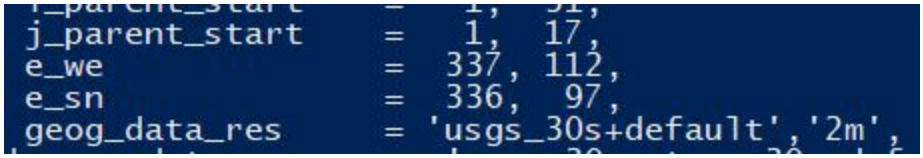
Advanced Usage

1. Change the source of the input data

```
geog_data_res = 'default','default',
```

For our training set-up:

```
cuahsi:~/WRF_WPS/WPS$ more namelist.wps
```




```
geog_data_res = 'usgs_30s+default','2m',
```

Note that we are using a different land cover dataset (USGS) and then the default for the other data. The '`2m`' is not used since we are only creating one domain.

Test using a different terrain dataset:

```
cuahsi:~/WRF_WPS/WPS$ vi namelist.wps
```

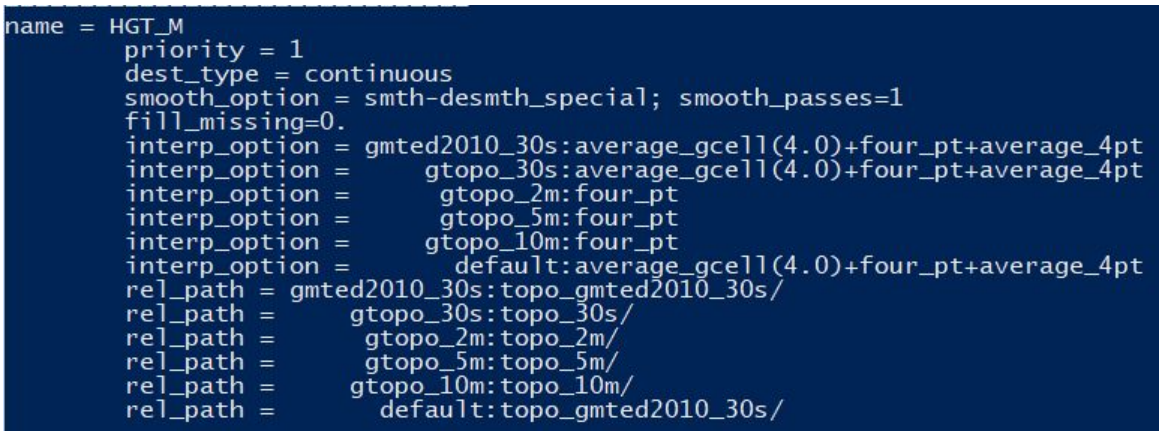
By using the arrow keys, place the cursor over the '`+`' and press '`i`' followed by '`+topo_30s`', then save using '`:wq`'.

```
geog_data_res = 'usgs_30s+topo_30s+default','default',
```

Now, rerun `geogrid`:

```
cuahsi:~/WRF_WPS/WPS$ ./geogrid.exe
```

```
cuahsi:~/WRF_WPS/WPS$ more geogrid/GEOGRID.TBL
```



```
cuahsi:~/WRF_WPS/WPS$ ls ~/ClassMaterials/data/wps_cutout
```

```
cuahsi@30650fe737a0:~/WRF_WPS/WPS$ ls ~/ClassMaterials/data/wps_cutout/
greenfrac_fpar_modis  lai_modis_10m  soiltemp_1deg     soiltype_top_30s  topo_gmted2010_30s
helper_files          landuse_30s    soiltype_bot_30s  topo_30s
```

2. Projection options in '`namelist.wps`'

Map projection / value of `map_proj`	Projection parameters

`Lambert Conformal / 'lambert'`	`truelat1`
	`truelat2` (optional)
	`stand_lon`

Mercator / `'mercator'`	`truelat1`
Polar stereographic / `'polar'`	`truelat1` `stand_lon`

Regular latitude-longitude, or cylindrical equidistant / `'lat-lon'`	`pole_lat`
	`pole_lon`
	`stand_lon`

Appendix: Definition of namelist.wps useful variables from:

http://www2.mmm.ucar.edu/wrf/users/docs/user_guide_V3.9/users_guide_chap3.html

PARENT_ID	A list of MAX_DOM integers specifying, for each nest, the domain number of the nest's parent; for the coarsest domain, this variable should be set to 1. Default value is 1.
PARENT_GRID_RATIO	A list of MAX_DOM integers specifying, for each nest, the nesting ratio relative to the domain's parent. No default value.
I_PARENT_START	A list of MAX_DOM integers specifying, for each nest, the x-coordinate of the lower-left corner of the nest in the parent unstaggered grid. For the coarsest domain, a value of 1 should be specified. No default value.
J_PARENT_START	A list of MAX_DOM integers specifying, for each nest, the y-coordinate of the lower-left corner of the nest in the parent unstaggered grid. For the coarsest domain, a value of 1 should be specified. No default value.
S_WE	A list of MAX_DOM integers which should all be set to 1. Default value is 1.
E_WE	A list of MAX_DOM integers specifying, for each nest, the nest's full west-east dimension. For nested domains, e_we must be one greater than an integer multiple of the nest's parent_grid_ratio (i.e., e_we = n*parent_grid_ratio+1 for some positive integer n). No default value.
S_SN	A list of MAX_DOM integers which should all be set to 1. Default value is 1.
E_SN	A list of MAX_DOM integers specifying, for each nest, the nest's full south-north dimension. For nested domains, e_sn must be one greater than an integer multiple of the nest's parent_grid_ratio (i.e., e_sn = n*parent_grid_ratio+1 for some positive integer n). No default value.

GEOG_DATA_RES	A list of MAX_DOM character strings specifying, for each nest, a corresponding resolution or list of resolutions separated by + symbols of source data to be used when interpolating static terrestrial data to the nest's grid. For each nest, this string should contain a resolution matching a string preceding a colon in a rel_path or abs_path specification (see the description of GEOGRID.TBL options) in the

	GEOGRID.TBL file for each field. If a resolution in the string does not match any such string in a rel_path or abs_path specification for a field in GEOGRID.TBL, a default resolution of data for that field, if one is specified, will be used. If multiple resolutions match, the first resolution to match a string in a rel_path or abs_path specification in the GEOGRID.TBL file will be used. Default value is 'default'.
DX	A real value specifying the grid distance in the x-direction where the map scale factor is 1. For ARW, the grid distance is in meters for the 'polar', 'lambert', and 'mercator' projection, and in degrees longitude for the 'lat-lon' projection; for NMM, the grid distance is in degrees longitude. Grid distances for nests are determined recursively based on values specified for parent_grid_ratio and parent_id. No default value.
DY	A real value specifying the nominal grid distance in the y-direction where the map scale factor is 1. For ARW, the grid distance is in meters for the 'polar', 'lambert', and 'mercator' projection, and in degrees latitude for the 'lat-lon' projection; for NMM, the grid distance is in degrees latitude. Grid distances for nests are determined recursively based on values specified for parent_grid_ratio and parent_id. No default value.
MAP_PROJ	A character string specifying the projection of the simulation domain. For ARW, accepted projections are 'lambert', 'polar', 'mercator', and 'lat-lon'; for NMM, a projection of 'rotated_ll' must be specified. Default value is 'lambert'.
REF_LAT	A real value specifying the latitude part of a (latitude, longitude) location whose (i,j) location in the simulation domain is known. For ARW, ref_lat gives the latitude of the center-point of the coarse domain by default (i.e., when ref_x and ref_y are not specified). For NMM, ref_lat always gives the latitude to which the origin is rotated. No default value.
REF_LON	A real value specifying the longitude part of a (latitude, longitude) location whose (i, j) location in the simulation domain is known. For ARW, ref_lon gives the longitude of the center-point of the coarse domain by default (i.e., when ref_x and ref_y are not specified). For NMM, ref_lon always gives the longitude to which the origin is rotated. For both ARW and NMM, west longitudes are negative, and the value of ref_lon should be in the range [-180, 180]. No default value.
TRUELAT1	A real value specifying, for ARW, the first true latitude for the Lambert conformal projection, or the only true latitude for the Mercator and polar stereographic projections. For NMM, truelat1 is ignored. No default value.
TRUELAT2	A real value specifying, for ARW, the second true latitude for the Lambert conformal conic projection. For all other projections, truelat2 is ignored. No default value.
STAND_LON	A real value specifying, for ARW, the longitude that is parallel with the y-axis in the Lambert conformal and polar stereographic projections. For the regular latitude-longitude projection, this value gives the rotation about the earth's geographic poles. For NMM, stand_lon is ignored. No default value.
GEOG_DATA_PATH	A character string giving the path, either relative or absolute, to the directory where the geographical data directories may be found. This path is the one to which rel_path specifications in the GEOGRID.TBL file are given in relation to. No default value.