**Settings for WRF & WRFChem….**

***ssh -X fkaragulian@sub2***

***mike machine (WRF)***

You can login to the wrfmachine (10.102.14.39) with:

U: fkaragulian p: fkaragulian123\_wrf

mtg is mounted under /home/fkaragulian/mtg/ with your permissions (i.e you can see your folder and shared folder).

Vineeth you are

U: vkvalappil p: vkvalappil123\_wrf

export LC\_LIBRARY\_PATH=/apps/netcdf/installed/lib

module load gcc/4.9.2

module load openmpi/1.8.4

export LC\_LIBRARY\_PATH=/apps/libpng/libpng-1.4.13/lib:$LD\_LIBRARY\_PATH

source "/home/fkaragulian/WRFV3/wrfchem\_env.txt"

export PATH=/apps/cdo/cdo-1.7.2/bin:$PATH

module load ncview

export PATH=/apps/ncl/ncl-6.3.0/bin:$PATH

export NCARG\_ROOT=/apps/ncl/ncl-6.3.0/

**download met data (GFS)……….use Mike’s & Vineeth bash scripts**

**cd into the directory where you want to download GFS data**

#!/bin/bash  
  
date\_day=20160310  
date\_month=`date +%Y%m -d$date\_day`  
hours=( `seq -f%03.0f 0 6 120` )  
initial=( 06  ) # 06 00  
  
for i in ${initial[@]}; do  
 for h in ${hours[@]}; do  
  #<ftp://nomads.ncdc.noaa.gov/GFS/Grid4/201603/20160310/>  
  echo <ftp://nomads.ncdc.noaa.gov/GFS/Grid4/$date_month/$date_day/gfs_4_$date_day>"\_"$i"00"\_$h.grb2  
  wget -c <ftp://nomads.ncdc.noaa.gov/GFS/Grid4/$date_month/$date_day/gfs_4_$date_day>"\_"$i"00"\_$h.grb2  
 done  
done

**$ chmod +x** ./download\_gfs.sh

**To run🡪 $** ./download\_gfs.sh

**To run🡪 $** ./download\_GFS\_today.sh

**$ nohup sh download\_GFS0.25.sh 20170915 06 &**

$ cd /disk3/fkaragulian/WRF\_UAE/WPS/

Make changes in the *namelist.wps* file

### introduced subdomain for only UAE area

*&geogrid*

*parent\_id = 1,* ***1****, 2,*

*parent\_grid\_ratio = 1,* ***3****, 3,*

*i\_parent\_start = 1,* ***65****, 31 ,95,*

*j\_parent\_start = 1,* ***65****, 40, 90,*

*e\_we = 187,* ***250****, 127, 106,*

*e\_sn = 184,* ***172****, 142, 109,*

**to plot-check domains**

[*http://www2.mmm.ucar.edu/wrf/OnLineTutorial/Class/cases/nest2.php*](http://www2.mmm.ucar.edu/wrf/OnLineTutorial/Class/cases/nest2.php)

**dlink namelist.wps in the /util directory to diplay the domain**

**cd /home/fkaragulian/WRF\_UAE/WPS/util/**

**ln -sf /home/fkaragulian/WRF\_UAE/WPS/namelist.wps .**

**ncl plotgrids\_new.ncl**

**ndown**

[*http://www2.mmm.ucar.edu/wrf/OnLineTutorial/Class\_York2013/cases/ndown.htm*](http://www2.mmm.ucar.edu/wrf/OnLineTutorial/Class_York2013/cases/ndown.htm)

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

To include erodibility map (EROD)

*If you run WRF-Chem, link the GEOGRIB table to GEOGRIB.TBL\_ARW\_CHEM*

**ln -svf GEOGRID.TBL.ARW\_CHEM GEOGRID.TBL**

**without chem use:**

**ln -svf GEOGRID.TBL.ARW GEOGRID.TBL**

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

$ cd /disk3/fkaragulian/WRF\_UAE/WPS/

**./geogrid.exe**

will generate *geo\_em.d01.nc*

will generate *geo\_em.d02.nc*

**link met data**

(data older than 2017)

./link\_grib.csh /home/fkaragulian/WRF\_UAE/forcing\_data/gfs\*

ln -sf ungrib/Variable\_Tables/Vtable.GFS Vtable

for 2017 data, use:

./link\_grib.csh /home/fkaragulian/WRF\_UAE/forcing\_data/20170915/gfs\*

ln -sf ungrib/Variable\_Tables/Vtable.GFS\_new Vtable

**./ungrib.exe**

will generate *FILE:2017-03-29\_06*

**./metgrid.exe**

will generate *met\_em.d01.2017-03-28\_12:00:00.nc*

will generate *met\_em.d02.2017-03-28\_12:00:00.nc* first date only

$ cd /disk3/fkaragulian/WRF\_UAE/WRFV3/test/em\_real/

**link ungribbed met files…..in the working directory**

ln -sf ../../../WPS/met\_em\* .

ln -sf /home/fkaragulian/WRF\_UAE/WPS/met\_em\* .

link the VEGPARM.TBL from /home/fkaragulian/WRFV3/run/

**ln -sf /home/fkaragulian/WRF\_UAE/WRFV3/run/VEGPARM.TBL\*** .

update NAMELIST.INPUT (WRF-Chem) for two domains

update NAMELIST.INPUT (WRF met) for two domains

**~~~~~~ run below only for WRF with MET data ~~~~~~~~~~~~~~~~~~~~~~~**

**Run real.exe**

bsub -n 6 -J job\_name -o vk.%J.out -e vk.%J.err -q normal mpirun **./real.exe**

bsub -n 96 -q general -W 600 -J example -o example.%J.out -e example.J.err "mpirun **./real.exe"**

will generate *wrfinput\_d01* (initialization file)

will generate *wrfinput\_d02* (initialization file)

will generate wrfbdy\_d01 (boundary conditions file)

**Run wrf.exe**

bsub -n 6 -J job\_name -o vk.%J.out -e vk.%J.err -q normal mpirun **./wrf.exe**

bsub -n 96 -q general -W 600 -J example -o example.%J.out -e example.J.err "mpirun **./wrf.exe**"

bsub –n 48 –q general –W 20 –J example –o example.%J.out –e example.%J.err mpirun –lsf hello.a

will generate *wrfout\_d01\_2017-03-12\_06:00:00* (these are .nc files)

will generate *wrfout\_d02\_2017-03-12\_06:00:00* (these are .nc files)

**check SUCCESS!!!**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**## ## ## ### ## ### ## ## ## ## ### ## ### ## ## ## ## ### ## ### ## ## ##**

**WRF-CHEM**

Untar wrf code.

Then untar wrfchem code inside wrf code.

tar -xvzf WRFV3.7.1.TAR.gz

tar -xvzf WRFV3-Chem-3.7.1.TAR.gz

WRFV3.7.1.TAR.gz  
WRFV3-Chem-3.7.1.TAR.gz

export LD\_LIBRARY\_PATH=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/hdf/h4dir/lib:/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/hdf/h5dir\_1.8.13/lib:/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/hdf/h5dir/lib:/home/fkaragulian/local/lib:/home/fkaragulian/gfortran/installed\_noquadmath/lib64:/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/lib:/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/bin/mpich-install/lib/:$LD\_LIBRARY\_PATH

export JASPERLIB=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/lib

export JASPERINC=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/include

export EM\_CORE=1

export WRF\_CHEM=1

export NETCDF=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local

export WRF\_KPP=1

export YACC='/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/bin/yacc -d'

export PATH=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/bin/mpich-install/bin/:/home/fkaragulian/local/ncl\_nodap/bin:/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/bin:/home/fkaragulian/gfortran/installed\_noquadmath/bin:$PATH:/home/fkaragulian/bin

export FLEX\_LIB\_DIR=/home/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/lib

module load gcc/4.9.2

module load openmpi/1.8.4

Then

**./configure  
./compile em\_real 2>&1 |tee compile.log**

-----------------------------------------------------------------------------------------

Use **./clean -a** to clean previous compilation

**Compilation of Emission Inventory**

Go to directory:

/disk3/fkaragulian/WRFV3/test/em\_real/PREP-CHEM-SRC-1.5/bin/build/

edit file:

Include.mk.gfortan

Add:

**NETCDF=/apps/netcdf/netcdf-4.1.3-usr-gcc** for NETCDF

**HDF5=/apps/hdf5/hdf5-1.8.13** for HDF5

Go to directory

/disk3/fkaragulian/WRFV3/test/em\_real/PREP-CHEM-SRC-1.5/src/

edit file:

edgar\_emissions.f90

go to lines from 840 to 845:

add spaces to the end of the names 'ENERGY', 'INDUSTRY', and 'TRANSPORT', until the length of all the names be the same as each other and equal to the length of 'AGRICULTURE' and 'RESIDENTIAL' (i.e. equal to 11 characters). the final modification must be like bellow:

**CODE:**[**SELECT ALL**](http://forum.wrfforum.com/viewtopic.php?f=39&t=9199)

841         'AGRICULTURE',&  
842         'ENERGY     ',&  
843         'INDUSTRY   ',&  
844         'RESIDENTIAL',&  
845         'TRANSPORT  '/)

Download EGDAR/HTAP emission data

cd /home/fkaragulian/WRFV3/test/em\_real/

wget <http://www.acom.ucar.edu/webt/wrf-chem/processors/EDGAR-HTAP.tgz>

extract the EDGAR/HTAP data:

tar -xvzf EDGAR-HTAP.tgz

**anthro\_emiss (from mike)**

Download **anthro\_emiss** from this page:

<https://www2.acom.ucar.edu/wrf-chem/wrf-chem-tools-community>

<http://www.acom.ucar.edu/wrf-chem/download.shtml>

….. also EDGAR-HTAP

Load anthro\_emiss in HCP

Untar “ANTHRO.tar”

**$ cd /home/fkaragulian/ANTHRO/src/**

……read the README.anthro\_emis

Set the environmental variable for the right fortran compiler

export FC=/apps/gcc/gcc-4.9.2/bin/gfortran

export NETCDF\_DIR=/apps/netcdf/netcdf-4.1.3-usr-gcc

export NETCDF\_DIR=/disk3/fkaragulian/wrf\_libraries\_gfort\_noquadmath/local/

proceed with the compilation of anthro\_emiss

modify the MAKEfile in /home/fkaragulian/ANTHRO/src/

compile anthro\_emiss….

./make\_anthro

it will build the executable anthro\_emis

unzip EDGAR-HTAP.tar in a separate folder

cd /home/fkaragulian/EDGAR-HTAP/

look at the file “**MOZCART.inp**” as example of input file for the emission ( /home/fkaragulian/EDGAR-HTAP/INP-Examples/)

Execution DATE\_TIME of the WRFChem will be taken from the files:

wrfinput\_d01 in the /home/fkaragulian/WRF\_UAE/WRFV3/test/em\_real/ directory…that is the one indicated in the namelist.input (from GFS met data)

**make a copy of the MOZCART.inp in the ANTHRO/src/ directory**

**change date in the MOZCART.inp according to the date of the namelist.input (met)**

export LD\_LIBRARY\_PATH=/apps/netcdf/netcdf-4.1.3-usr-gcc/lib/:$LD\_LIBRARY\_PATH

To run anthro\_emis, execute the command:

**$ cd /home/fkaragulian/ANTHRO/src/**

**./anthro\_emis < MOZCART.inp > MOZCART\_FK.out**

….it will generate files:

**wrfchemi\_00z\_d01**

**wrfchemi\_12z\_d01**

**download wes-coldens.tar to process MOZCART configuration**

**untar wes-coldens.tar in a new directory:** tar –xvf wes\_coldens

**compile the MOZCART source code**

**change to gfortaran processor in the make\_util and in the makefile and in the in the /home/fkaragulian/WRFV3/wesely/ directory**

**$ cd /disk3/fkaragulian/WRFV3/wesely/**

./make\_util wesely (will generate wesely)

./make\_util exo\_coldens (will generate exo\_coldens)

**update the namelist.input (in the wesely folder) as follow:**

&control  
 wrf\_dir = ' /home/fkaragulian/WRFV3/test/em\_real/'

wrf\_dir = '/home/fkaragulian/WRF\_UAE/WRFV3/test/em\_real/'  
 domains = 1,  
/

run exo\_coldens with the updated namelist.input

**./exo\_coldens < namelist.input**

Will generate ***exo\_coldens\_d01***

to generate wrfchemi\_00z\_d02 and wrfchemi\_00z\_d02:

* change the **namelist.input** file (met). Move the nested information to

the mother domain column (e\_ew; e\_sn; dx = 4000; dy=4000)

* Move (rename) the met **wrfinput\_d02** to **wrfinput\_d01** (make a copy first of the previous wrfinput\_d01…the coarse one)
* cp file.doc newfile.doc
* Go to cd /home/fkaragulian/ANTHRO/src/ and rename wrfchemi\_00z\_d01 and wrfchemi\_12z\_d01 into **wrfchemi\_00z\_d01\_coarse** and **wrfchemi\_12z\_d01\_coarse**
* Run **anthro\_emis** for wrfchemi\_00z\_d01 and wrfchemi\_12z\_d01 (this time will be for the nested domain)

**./anthro\_emis < MOZCART.inp > MOZCART\_FK.out**

* Rename wrfchemi\_00z\_d01 and wrfchemi\_12z\_d01 into **wrfchemi\_00z\_d02** and **wrfchemi\_12z\_d02** and set back the **wrfchemi\_00z\_d01** and **wrfchemi\_12z\_d01.**
* Link the output from **anthro\_emis (**wrfchemi\_00z\_d02 and wrfchemi\_12z\_d02 (for the nested domain) and wrfchemi\_00z\_d01 and wrfchemi\_12z\_d01 for the main domain to /home/fkaragulian/WRFV3/test/em\_real/ (see below)
* $ cd /home/fkaragulian/WRFV3/test/em\_real/

ln -sf /home/fkaragulian/ANTHRO/src/wrfchemi\_\* .

* Also link met files

ln -sf /home/fkaragulian/WRF\_UAE/WPS/met\_em\* .

* update the **NAMELIST.INPUT.CHEM** in /disk3/fkaragulian/WRFV3/test/em\_real/. Move the nested information to the mother domain column (e\_ew; e\_sn; dx = 4000; dy=4000)
* first run
* Go to **cd /disk3/fkaragulian/WRFV3/wesely/** and rename exo\_coldens\_d01 to **exo\_coldens\_d01\_coarse**

(First run **./exo\_coldens < namelist.input** with the d01 domain)

* run exo\_colden with the updated namelist.input.chem **./exo\_coldens < namelist.input.** It will generate a new **exo\_coldens\_d01** (finer for the nested domain).
* Finally, RENAME all the new \_d01 into \_d02: the resulting wrfchemi\_d01 into **wrfchemi\_d02** and rename wrfchemi\_d01\_coarse into wrfchemi\_d01. SAME FOR exo\_coldens: exo\_coldens\_d01\_coarse will be renamed exo\_coldens\_d01 and the new exo\_coldens\_d01 Will be renamed exo\_coldens\_d02.

### link all the wrfchemi\_d01 and wrfchem\_d02, and exo\_coldens\_d01, exo\_coldens\_d02 to /home/fkaragulian/WRFV3/test/em\_real/ ………see below

**!!!! reload the original input.namelist (met) & input.namelist.chem (chem) for the 2 domain configuration**

source "/home/fkaragulian/WRFV3/wrfchem\_env.txt"

$ cd /home/fkaragulian/WRFV3/test/em\_real/

**link ungribbed met files…..in the working directory** em\_real/

ln -sf /home/fkaragulian/WRF\_UAE/WPS/met\_em\* . (generated above with ./metgrid.exe above)

**link emission files wrfchemi\_00z\_d01 generated from ANTRO**

ln -sf /home/fkaragulian/ANTHRO/src/wrfchemi\_\* .

update the **NAMELIST.INPUT.CHEM** in /disk3/fkaragulian/WRFV3/test/em\_real/

namelist.input.chem MUST be renamed into namelist.input or just link the “name” of namelist.input to namelist.input.chem

ln -sf namelist.input.chem namelist.input

**cd /home/fkaragulian/WRFV3/test/em\_real/**

**link exo\_coldens into working wrf directory**

ln -sf /home/fkaragulian/WRFV3/wesely/exo\_coldens\_\* .

**######### RUN the WRFChem model ####################**

**Run real.exe**

bsub -n 96 -J job\_name -o vk.%J.out -e vk.%J.err -q normal mpirun **./real.exe**

bsub -n 96 -q general -W 600 -J example -o example.%J.out -e example.J.err "mpirun **./real.exe"**

will generate *wrfinput\_d01* (initialization file)

will generate *wrfinput\_d02* (initialization file)

will generate wrfbdy\_d01 (boundary conditions file)

**Run wrf.exe (WRFChem also)**

bsub -n 96 -J job\_name -o vk.%J.out -e vk.%J.err -q normal mpirun **./wrf.exe**

bsub -n 96 -q general -W 600 -J example -o example.%J.out -e example.J.err "mpirun **./wrf.exe**"

will generate *wrfout\_d01\_2017-03-27\_06:00:00* (these are .nc files)

to see al jobs: bjobs

to kill a job: bkill xxxxxx

bjobs –u all al(from everybody)

bhosts (to see if processors are open or closed)

**NEW JOB SUBMISSION**

**Scripts and documentation**

This example, config.sh, requests 48 cores on the general queue for 20 minutes, to run the executable hello.a.

  ######### config.sh ########

  #BSUB -n 48

  #BSUB –q general

  #BSUB –W 20 #Run limit in minutes

  #BSUB -J example

  #BSUB -o example.%J.out

  #BSUB -e example.%J.err

  mpirun –lsf hello.a

This can then be submitted by executing Submit.sh file given below:

  ######### Submit.sh ########

  bsub < config.sh

Alternatively, this can be submitted directly from the command line via the command:

bsub –n 48 –q general –W 20 –J example –o example.%J.out –e example.%J.err mpirun –lsf ./wrf.exe

bsub -n 96 -q general -W 500 -J example -o example.%J.out -e example.J.err "mpirun ./wrf.exe"

**WRF post-processing**

#### visualize .nc files #####################################

Load environ. variab

source "/home/fkaragulian/WRFV3/wrfchem\_env.txt"

module load ncview

use **ncdump** to see layers in the .nc “wrfout\_d01\_2017-03-27\_06:00:00”

ncdump -h *wrfout\_d01\_2017-03-27\_06:00:00*

to visualize the .nc wrfchem output file:

ncview *wrfout\_d01\_2017-03-27\_06:00:00*

ncview *"/home/fkaragulian/WRFV3/test/em\_real/wrfout\_d01\_2017-03-27\_06:00:00\_FK\_TRIAL"*

### check variable included in the model run

/home/fkaragulian/WRFV3/Registry/

open **registry.chem**

**#### POST PROCESSING WRF-Chem ######################**

**### Climate Data Operator tool ######**

**# cdo tool**

export PATH=/apps/cdo/cdo-1.7.2/bin:$PATH

**cdo program**

cdo infov wrfout\_d01\_2017-03-27\_06:00:00\_FK\_TRIAL | grep -i "DUST\_1" > federico.txt

**cdo infov wrfout\_d01\_2015-03-31\_00:00:00 | grep -i "SWDOWN"**

cdo infov wrfpost.nc |grep SW\_d|less

**########** ncl tool 🡪 **TO RUN ###########**

**ncl 'file\_in="wrfout\_d01\_2017-09-15\_07:00:00"' wrfpost\_dust\_20161010.ncl**

**Air Quality data {PM10, PM2.5, CO, O3, NO2, SO2)**

**ncl 'file\_in="wrfout\_d01\_2017-09-15\_06:00:00"' wrfpost\_dust\_20170927\_airquality.ncl**

**Extinction coefficients…**

**ncl 'file\_in="wrfout\_d01\_2015-03-31\_00:00:00"' EXTCOF55.ncl**

**to save all the headers of the wrf output:**

**ncdump -h wrfout\_d02\_2015-03-31\_00:00:00 > wrfout\_header.txt**

to search for a specific variable:

**ncdump -h wrfout\_d01\_2015-03-31\_00:00:00 |less**

press /**P25** (P25 is the name of the variable)

to esc press **q**

#########################################################

Nice GUIDE for namelist settings in WRF-Chem

<https://github.com/hanschen/WRFDA/tree/master/run>

<file:///C:/Users/fkaragulian/Downloads/wrfsolar_users_guide.pdf>

to simulate the solar radiation in the presence of AOD (Aerosols at 550 nm) make the following changes in the WRF-chem **namelist.input** in the physics session:

&physics

aer\_opt = 2, using J. A. Ruiz-Arias method

The following aerosol options allow RRTMG and new Goddard radiation schemes to see it, but the aerosols are constant during the model integration:

aer\_aod550\_opt = 1,

aer\_aod550\_val = 0.12

input constant value for AOD at 550 nm from namelist. In this case, the value is read from aer\_aod550\_val = 0.12 or any value at user’s choice.

aer\_angexp\_opt = [1,2,3] :

1 = input constant value for Angstrom exponent from namelist. In

aer\_type = [1,2,3] : 1 for rural, 2 is urban and 3 is maritime.

*OPTIONAL:* Turn on the aerosol optical properties (e.g., aer\_op\_opt, opt\_pars\_out) to obtain the extinction coefficient in the wrfout file and rerun the simulation. Examine the EXTCOEF fields. Are these fields non-zero? Do the higher values correlate with locations of higher aerosol concentration? The vertical sum of each EXTCOEF field multiplied by the layer depth is the computed Aerosol Optical Depth (AOD) at a particular wavelength.

dz(1) = z\_at\_w(2) - z\_at\_w(1)  
dz(2) = z\_at\_w(3) - z\_at\_w(2)  
...  
dz(e\_vert-1) = z\_at\_w(e\_vert) - z\_at\_w(e\_vert-1)  
AOD = SUM[EXTCOEF55(1)\*dz(1) + EXTCOEF55(2)\*dz(2) + ... + EXTCOEF55(n)\*dz(e\_vert-1)]

- Now you will need to make a txt file called "myoutfields.txt". Inside that file, you will need to declare the variables that you wish to output. For this exercise, we are going to ask the model to output RTHCUTEN, RTHBLTEN, RTHRATEN, andH\_DIABATIC. Inside the "myoutfields.txt" file, you will simply need to type the following:

+:h:0:RTHCUTEN,RTHBLTEN,RTHRATEN,H\_DIABATIC

Dear Federico,

To compute the dust-emission and dust-deposition you need to read these variables:

for **dust-emission**: dustload\_1, dustload\_2, dustload\_3,dustload\_4,dustload\_5

for **dust-deposition**  : setvel\_1, setvel\_2,setvel\_3,setvel\_4, setvel\_5, drydep\_1,drydep\_2,drydep\_3,drydep\_4,drydep\_5

where 1 - for bin1; 2 - for bin2; 3 - for bin3 and so on

these variables by default are not switch on in wrfout , for this you need to find them in Registry.chem file and enable them  and after that you have to recompile the wrfchem(don't forget to do ./clean -a before compiling)

the another opportunity to obtain these variables without editing Registry file and recompiling wrf- chem  is to add name of all variables which you want to **my\_variables\_d01.txt** and add in **namelist.input** this line : **iofields\_filename = 'my\_variables\_d01.txt'**in **&time\_control** section

**to calculate total mass you have to sum all bin and convert to comfortable unit like kg/m2.**

**##########################################################################################################################################**

**Initializing WRF-Chem with previous run (auxinput12)**

Link the saved 12 hour forecast output file to your run directory and name the input file "wrf\_chem\_input\_d01"

**ln -sf Output/wrfout\_d01\_2010-07-14\_12:00:00 wrf\_chem\_input\_d01**

auxinput12\_inname = 'wrf\_chem\_input',

io\_form\_auxinput12 = 2,

...

chem\_in\_opt = 1,

to move files in linux

you are in the directory where wrfoutput files are located

mv wrfout\_d0\* /research/cesam/AirQuality/WRF\_outputs/

Operational WRF-Chem

**Format gfs data**

[**http://nomads.ncep.noaa.gov/cgi-bin/filter\_gfs\_0p25.pl?file=gfs.t06z.pgrb2.0p25.f072&all\_lev=on&all\_var=on&subregion=&leftlon=20.00&rightlon=120.00&toplat=40.00&bottomlat=00.00&dir=%2Fgfs.2017092700**](http://nomads.ncep.noaa.gov/cgi-bin/filter_gfs_0p25.pl?file=gfs.t06z.pgrb2.0p25.f072&all_lev=on&all_var=on&subregion=&leftlon=20.00&rightlon=120.00&toplat=40.00&bottomlat=00.00&dir=%2Fgfs.2017092700)

**sh execute\_Model.sh 2017092700 (morning)**

**sh execute\_Model.sh 2017092712 (afternoon)**

**sh execute\_Model.sh 2017092800**

**sh execute\_Model.sh 2017100100**

**sh execute\_Model.sh 2017100406**

**sh execute\_Model.sh 2017100800**

**$ echo $date**

**sh execute\_Model.sh 2017100900**

########################################### Post Processing (to run for each output hour ##################################################################

#dir=/research/cesam/AirQuality/WRF\_outputs/2017092800

#cd ${dir}/

#files=wrfout\_d0\*

# echo ${files}

#for i in ${files[@]}; do

# output=( `echo ${i##\*/}|sed 's/wrfout/wrfpost/'` )

#echo file\_in is $i

#echo file\_out is $output

#ncl 'file\_in="'$i'"' 'file\_out="./'$output'.nc"' ./wrfpost\_dust\_20170927\_airquality.ncl

# ncl 'file\_in="'$i'"' 'file\_out="./'$output'.nc"' /home/fkaragulian/bin/wrfpost\_dust\_20170927\_airquality.ncl

# done

**# new smaller domain**

&geogrid

parent\_id = 1,1,2,

parent\_grid\_ratio = 1,3,3,

i\_parent\_start = 1, 65,31,95,

j\_parent\_start = 1, 42,40,90,

e\_we = 180,209,127,106,

e\_sn = 120,140,142,109,

geog\_data\_res = '30s','30s', '30s'

end\_second = 00, 00, 00,

interval\_seconds = 21600,

input\_from\_file = .true.,.true.,.true.,

**history\_outname = '/research/cesam/AirQuality/WRF\_outputs/test/wrfout\_d<domain>\_<date>',**

history\_interval = 60, 60, 60,

frames\_per\_outfile = 1, 1, 1000,

########################################

Copy data from the day before when a WRF\_Chem run fail

bash copyoldData.sh 2017110600 (today date)

###########################################

date=`date +%Y%m%d`00

echo $date

2017120300 (this is the date at time 00)

${date:0:4}

2017 (this is the year)

echo ${date:4:2}

12 (this is the month)

echo ${date:6:2}

03 (this is the day)

echo ${date:8:2}

00 (this is the hour)

echo 2010-${date:4:2}-01\_${date:8:2}:00:00

2010-12-01\_00:00:00

echo 2010-${date:4:2}-04\_${date:8:2}:00:00

2010-12-04\_00:00:00