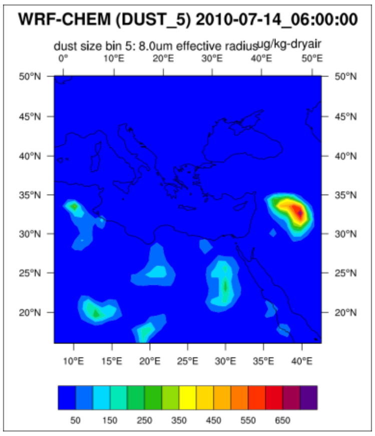
สถาบันสารสนเทศทรัพยากรน้ำ
Hydro - Informatics Institute

KPP

WRF-Chem



How to Install WRF-Chem Model

Version 4.4 and KPP

On Ubuntu 22.04.

Compute System

Technological development and Digitization

Hydro Informatics Institute (HII)

Thailand.

Compute System.	22 Jun 2022

CONTENTS

How to Install WRF-Chem Model Version 4.4 and KPP On Ubuntu 22.04.

1. Objective.

To understand about WRF-Chem with KPP installation.

2. Scope.

Installation WRF-Chem Model Version 4.4 and KPP On Ubuntu.

3. Definition.

Ubuntu is linux operating system.

WRF-Chem Model is the Weather Research and Forecasting (WRF) model coupled with Chemistry. The model simulates the emission, transport, mixing, and chemical transformation of trace gases and aerosols simultaneously with the meteorology. The model is used for investigation of regional-scale air quality, field program analysis, and cloud-scale interactions between clouds and chemistry.

Kinetic PreProcessor(KPP) is a software tool that assists the computer simulation of chemical kinetic systems. The concentrations of a chemical system evolve in time according to the differential law of mass action kinetics. A numerical simulation requires an implementation of the differential laws and a numerical integration in time.

4. Device.

Linux Ubuntu 22.04 type server have available space is 80G and memory more than 32G.

5. Operating Procedures.

5.1. open ssh client and connect to your linux ubuntu with username ubuntu.

```
ssh ubuntu@192.168.122.229
```

show ubuntu version.

```
$ cat /etc/os-release
```

```
NAME="Ubuntu"
VERSION_ID="22.04"
VERSION="22.04 LTS (Jammy Jellyfish)"
```

show default home directory.

```
$ pwd
```

```
/home/ubuntu
```

show user name.

```
$ whoami
```

```
ubuntu
```

5.2. update ubuntu software.

update repository.

```
$ sudo apt update
```

5.3. install preferred software.

Install preferred software by command.

```
$ sudo apt install csh m4 build-essential nasm cmake unzip libxmu-dev libcairo-dev libbz2-dev libxaw7-dev libx11-dev xorg-dev flex bison subversion liburi-perl evince tcsh cpp m4 quota cvs libomp-dev python3-pip freeglut3-dev libjpeg-dev file
```

5.4. install compiler.

install compiler by command.

```
$ sudo apt install gcc g++ gfortran
```

and show compiler version.

```
$ gcc -v
$ g++ -v
$ cpp --version
$ gfortran -v
```

5.5. Directory Structure.

```
/                     root directory of linux
|--home                      home directory of linux
  |--ubuntu                  default ubuntu home directory of linux
    |--Build_WRF             main WRF directory.
      |--LIBRARIES           WRF library directory.
      |--geos                Geography Data and mount to /data/geos
      |--DATA                Directory for Real-time Data.
      |--src                 Directory for Source Code.
```

5.6. Install Preferred Library.

Preferred Library include

- mpich
- zlib
- hdf5
- curl
- netcdf
- libpng
- Jasper

5.6.1 Install mpich Library.

Create source code directory and change to directory.

```
$ mkdir -p /home/ubuntu/Build_WRF/src
$ mkdir -p /home/ubuntu/Build_WRF/LIBRARIES
$ cd /home/ubuntu/Build_WRF/src
```

Download Source Code.

```
$ wget https://www.mpich.org/static/downloads/4.0.2/mpich-4.0.2.tar.gz
```

Uncompress Source Code and Compile.

```
$ tar -xzvf mpich-4.0.2.tar.gz
$ cd mpich-4.0.2

$ FFLAGS=-fallow-argument-mismatch FCFLAGS=-fallow-argument-mismatch ./configure --
prefix=/home/ubuntu/Build_WRF/LIBRARIES/mpich

$ make -j4
$ make install
```

create PATH variable in file .profile.

```
$ vi /home/ubuntu/.profile
```

```
### mpich
export PATH=/home/ubuntu/Build_WRF/LIBRARIES/mpich/bin:$PATH
```

update PATH variable.

```
$ source /home/ubuntu/.profile
```

5.6.2 install zlib library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

download source code.

```
$ wget http://www.zlib.net/fossils/zlib-1.2.12.tar.gz
```

uncompress source code and compile.

```
$ tar xzvf zlib-1.2.12.tar.gz
$ cd zlib-1.2.12
$ ./configure --prefix=/home/ubuntu/Build_WRF/LIBRARIES/zlib
$ make -j4
$ make install
```

5.6.3 Install hdf5 Library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

Download Source Code

```
$ wget https://hdf-wordpress-1.s3.amazonaws.com/wp-
content/uploads/manual/HDF5/HDF5_1_12_2/source/hdf5-1.12.2.tar.gz
```

Uncompress Source Code and Compile.

```
$ tar xzvf hdf5-1.12.2.tar.gz
$ cd hdf5-1.12.2
$ ./configure --prefix=/home/ubuntu/Build_WRF/LIBRARIES/hdf5 --with-
zlib=/home/ubuntu/Build_WRF/LIBRARIES/zlib --enable-fortran --enable-fortran2003 --
enable-cxx --with-default-api-version=v18

$ make -j4
$ make install
```

Create PATH variable in file .bash_profile.

```
$ vi /home/ubuntu/.profile
```

```
### hdf5
export PATH=/home/ubuntu/Build_WRF/LIBRARIES/hdf5/bin:$PATH
export LD_LIBRARY_PATH=/home/ubuntu/Build_WRF/LIBRARIES/hdf5/lib:$LD_LIBRARY_PATH
```

update PATH variable.

```
$ source /home/ubuntu/.profile
```

5.6.4 install curl library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

download source code.

```
$ wget https://curl.se/download/curl-7.83.1.tar.gz
```

uncompress source code and compile.

```
$ tar xzvf curl-7.83.1.tar.gz
$ cd curl-7.83.1
$ ./configure --prefix=/home/ubuntu/Build_WRF/LIBRARIES/curl --with-
zlib=/home/ubuntu/Build_WRF/LIBRARIES/zlib --without-ssl
$ make -j4
$ make install
```

5.6.5. Install netcdf Library.

Change to directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

Download Source Code.

```
$ wget ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-c-4.8.0.tar.gz
$ wget ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-fortran-4.5.3.tar.gz
```

Uncompress Source Code and Compile.

```
$ tar xzvf netcdf-c-4.8.0.tar.gz
$ cd netcdf-c-4.8.0

$ CFLAGS=-fPIC CPPFLAGS='-I/home/ubuntu/Build_WRF/LIBRARIES/hdf5/include -
I/home/ubuntu/Build_WRF/LIBRARIES/curl/include' LDFLAGS='-
L/home/ubuntu/Build_WRF/LIBRARIES/hdf5/lib -
L/home/ubuntu/Build_WRF/LIBRARIES/curl/lib ' ./configure --
prefix=/home/ubuntu/Build_WRF/LIBRARIES/netcdf --enable-netcdf-4 --enable-netcdf4 --
enable-shared --enable-dap

$ make -j4
$ make install
$ cd ..
$ tar xzvf netcdf-fortran-4.5.3.tar.gz
$ cd netcdf-fortran-4.5.3

$ CPPFLAGS='-I/home/ubuntu/Build_WRF/LIBRARIES/netcdf/include' LDFLAGS='-
L/home/ubuntu/Build_WRF/LIBRARIES/netcdf/lib' FCFLAGS='-m64' ./configure --
prefix=/home/ubuntu/Build_WRF/LIBRARIES/netcdf

$ make -j4
$ make install
```

Create PATH variable in file .profile.

```
$ vi /home/ubuntu/.profile
```

```
### netcdf
export PATH=/home/ubuntu/Build_WRF/LIBRARIES/netcdf/bin:$PATH
export NETCDF=/home/ubuntu/Build_WRF/LIBRARIES/netcdf
export LD_LIBRARY_PATH=/home/ubuntu/Build_WRF/LIBRARIES/netcdf/lib:$LD_LIBRARY_PATH
```

update PATH variable.

```
$ source /home/ubuntu/.profile
```

5.6.6. Install libpng Library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

Download Source Code

```
$ wget https://jaist.dl.sourceforge.net/project/libpng/libpng16/1.6.37/libpng-1.6.37.tar.gz
```

Uncompress Source Code and Compile.

```
$ tar xzvf libpng-1.6.37.tar.gz
$ cd libpng-1.6.37

$ CPPFLAGS='-I/home/ubuntu/Build_WRF/LIBRARIES/netcdf/include' FCFLAGS='-
m64' ./configure --prefix=/home/ubuntu/Build_WRF/LIBRARIES/libpng

$ make -j4
$ make install
```

5.6.7. Install Jasper Library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

Download Source Code.

```
$ wget https://www.ece.uvic.ca/~frodo/jasper/software/jasper-2.0.10.tar.gz
```

Uncompress Source Code and Compile.

```
$ tar -xvzf jasper-2.0.10.tar.gz


$ cmake -G "Unix Makefiles" -H/home/ubuntu/Build_WRF/src/jasper-2.0.10 -
B/home/ubuntu/Build_WRF/src/jasper-2.0.10-build -
DCMAKE_INSTALL_PREFIX=/home/ubuntu/Build_WRF/LIBRARIES/jasper


$ cd /home/ubuntu/Build_WRF/src/jasper-2.0.10-build
$ make install
```

Create PATH variable in file .profile.

```
$ vi /home/ubuntu/.profile
```

```
#### jasper
export PATH=/home/ubuntu/Build_WRF/LIBRARIES/jasper/bin:$PATH
export LD_LIBRARY_PATH=/home/ubuntu/Build_WRF/LIBRARIES/jasper/lib:$LD_LIBRARY_PATH
```

update PATH variable.

```
$ source /home/ubuntu/.profile
```

5.6.8 Install ncl Library.

change to source code directory.

```
$ cd /home/ubuntu/Build_WRF/src
```

Install miniconda.

```
$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
$ sh ./Miniconda3-latest-Linux-x86_64.sh
```

Enter..., yes, ENTER, yesl

Install ncl.

```
$ source /home/ubuntu/.bashrc
$ conda create -n ncl_stable -c conda-forge ncl
$ source activate ncl_stable
```

Test ncl.

```
$ ncl -V
```

Show version 6.6.2 is good.

Create variable in file .profile.

```
$ vi /home/ubuntu/.bashrc
```

```
#### ncl
source activate ncl_stable
```

5.7. Install WRF-Chem.

5.7.1 Install WRF-Chem and KPP.

Variable setup.

```
$ ulimit -s unlimited
$ export MALLOC_CHECK_=0
$ export EM_CORE=1
$ export NMM_CORE=0
$ export WRF_CHEM=1
$ export WRF_KPP=1
$ export YACC='/usr/bin/yacc -d'
$ export FLEX=/usr/bin/flex
$ export FLEX_LIB_DIR=/usr/lib/x86_64-linux-gnu
$ export KPP_HOME=/home/ubuntu/Build_WRF/WRFV4.4/chem/KPP/kpp/kpp-2.1
$ export WRF_SRC_ROOT_DIR=/home/ubuntu/Build_WRF/WRFV4.4
$ export PATH=$KPP_HOME/bin:$PATH
$ export SED=/usr/bin/sed
$ export WRFIO_NCD_LARGE_FILE_SUPPORT=1
```

Change to directory.

```
$ cd /home/ubuntu/Build_WRF/
```

Download Source Code.

```
$ wget -c https://github.com/wrf-model/WRF/releases/download/v4.4/v4.4.tar.gz -O wrf-4.4.tar.gz
```

Uncompress Source Code and Configure WRF.

```
$ tar -xvzf wrf-4.4.tar.gz
$ cd WRFV4.4/chem/KPP/kpp/kpp-2.1/src
$ /usr/bin/flex scan.l
$ cd /home/ubuntu/Build_WRF/WRFV4.4
```

Edit configure file line 919

$ vi configure	
Change from	**To**
if ["$USENETCDFPAR" == "1"] ; then	if ["$USENETCDFPAR" = "1"] ; then

Run configure wrf

```
$ ./configure
```

select 34. (dmpar)   GNU (gfortran/gcc)

```
   Enter selection [1-75] : 34
```

and nesting is 1

```
   Compile for nesting? (1=basic, 2=preset moves, 3=vortex following) [default 1]: 1
```

Complie kpp.

```
$ ./compile 2>&1 | tee compile_kpp.log
```

Show message below is good.

Complie em_real mode.

```
$ ./compile em_real 2>&1 | tee compile_wrf.log
```

Wait 30-50 minutes for finish test by

```
$ ls -lah main/*.exe
```

If you see real.exe and wrf.exe then correct. Else check Error in compile_wrf.log file.

5.7.2 Install WPS.

Change to directory.

```
$ cd /home/ubuntu/Build_WRF/
$ ln -sf WRFV4.4 WRF
```

Download Source Code.

```
$ wget -c https://github.com/wrf-model/WPS/archive/refs/tags/v4.4.tar.gz -O wps-4.4.tar.gz
```

Uncompress Source Code.

```
$ tar -xvzf wps-4.4.tar.gz
```

Define variable.

```
$ cd WPS-4.4
$ export JASPERLIB=/home/ubuntu/Build_WRF/LIBRARIES/jasper/lib
$ export JASPERINC=/home/ubuntu/Build_WRF/LIBRARIES/jasper/include
```

Configure WPS.

```
$ ./configure
```

select 3. Linux x86_64, gfortran   (dmpar)

Enter selection [1-40] : **3**

edit configure.wps file. Change DM_FC to mpif90 and Append -lgomp in WRF_LIB.

```
$ vi configure.wps
```

```
COMPRESSION_LIBS    = -L/home/ubuntu/Build_WRF/LIBRARIES/jasper/lib -ljasper -lpng -lz
COMPRESSION_INC     = -I/home/ubuntu/Build_WRF/LIBRARIES/jasper/include

DM_FC               = mpif90
WRF_LIB     =     -L$(WRF_DIR)/external/io_grib1 -lio_grib1 \
                  -L$(WRF_DIR)/external/io_grib_share -lio_grib_share \
                  -L$(WRF_DIR)/external/io_int -lwrfio_int \
                  -L$(WRF_DIR)/external/io_netcdf -lwrfio_nf \
                  -L$(NETCDF)/lib -lnetcdff -lnetcdf -lgomp
```

Complie WPS.

```
$ ./compile 2>&1 | tee compile_wps.log
```

Wait for finish test by

```
$ ls -lah *.exe
```

If you see geogrid.exe metgrid.exe and ungrib.exe then correct. Else check Error in compile_wps.log file.

5.8. Create Directory for Geography Data.

Change Directory.

```
$ cd /home/ubuntu/Build_WRF/
```

Downlaod Geography Data.

```
$ wget http://www2.mmm.ucar.edu/wrf/src/wps_files/albedo_modis.tar.bz2
$ wget http://www2.mmm.ucar.edu/wrf/src/wps_files/geog_complete.tar.gz
$ wget http://www2.mmm.ucar.edu/wrf/src/wps_files/maxsnowalb_modis.tar.bz2
$ wget http://www2.mmm.ucar.edu/wrf/src/wps_files/topo_2m.tar.bz2
```

Uncompress Geography Data.

```
$ tar -xvzf geog_complete.tar.gz
$ cd geog
$ tar -xvjf ../albedo_modis.tar.bz2
$ tar -xvjf ../maxsnowalb_modis.tar.bz2
$ tar -xvjf ../topo_2m.tar.bz2
```

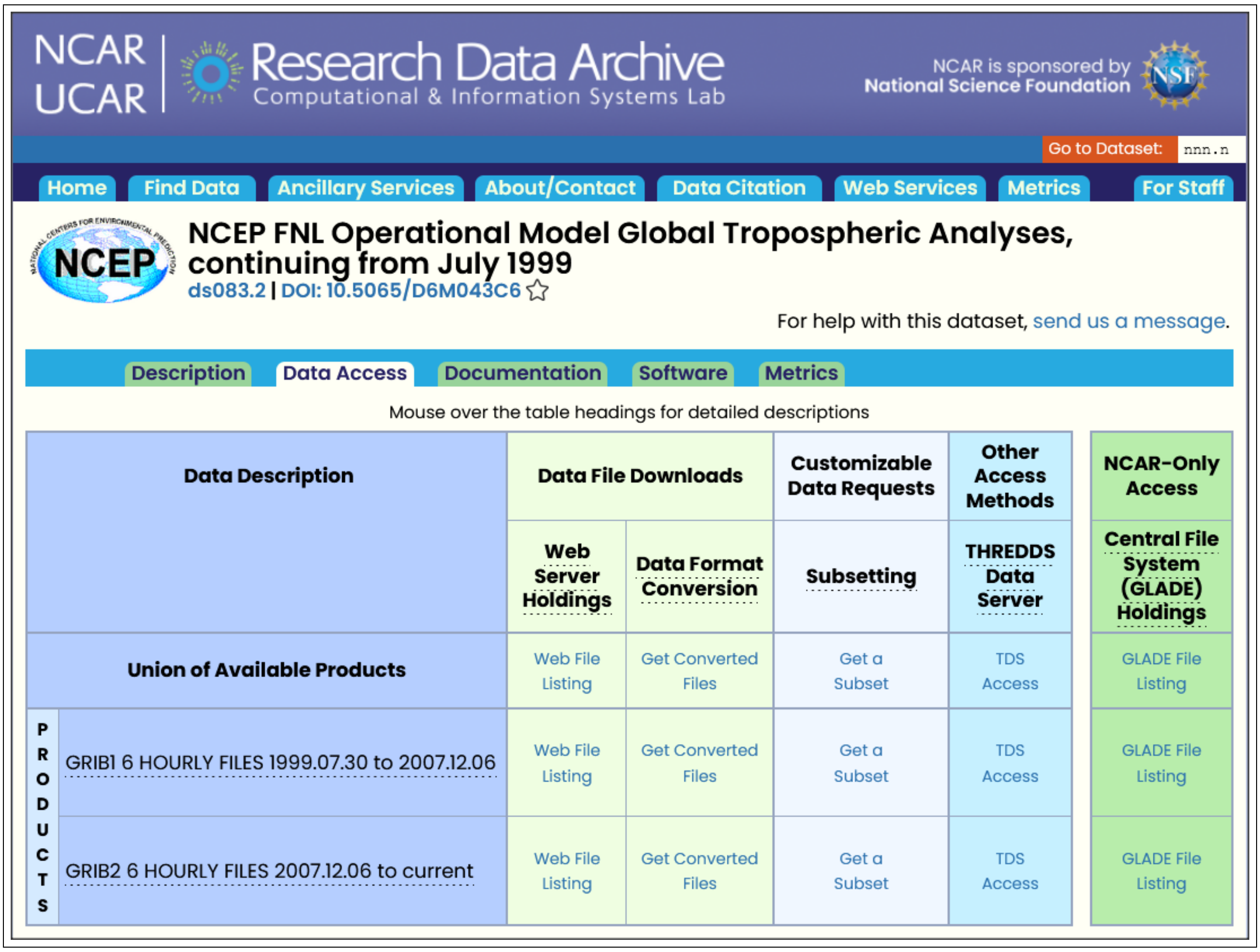
5.9. Create directory for Input Data.

Create Real-time Data Directory and Change to Directory.

```
$ mkdir -p /home/ubuntu/Build_WRF/data/fnl
$ cd /home/ubuntu/Build_WRF/data/fnl
```

Download Data from

https://rda.ucar.edu/datasets/ds083.2/index.html#sfol-wl-/data/ds083.2?g=22010



NCAR UCAR Research Data Archive Computational & Information Systems Lab

NCAR is sponsored by National Science Foundation

Go to Dataset: nnn.n

Home | Find Data | Ancillary Services | About/Contact | Data Citation | Web Services | Metrics | For Staff

NCEP FNL Operational Model Global Tropospheric Analyses, continuing from July 1999

ds083.2 | DOI: 10.5065/D6M043C6

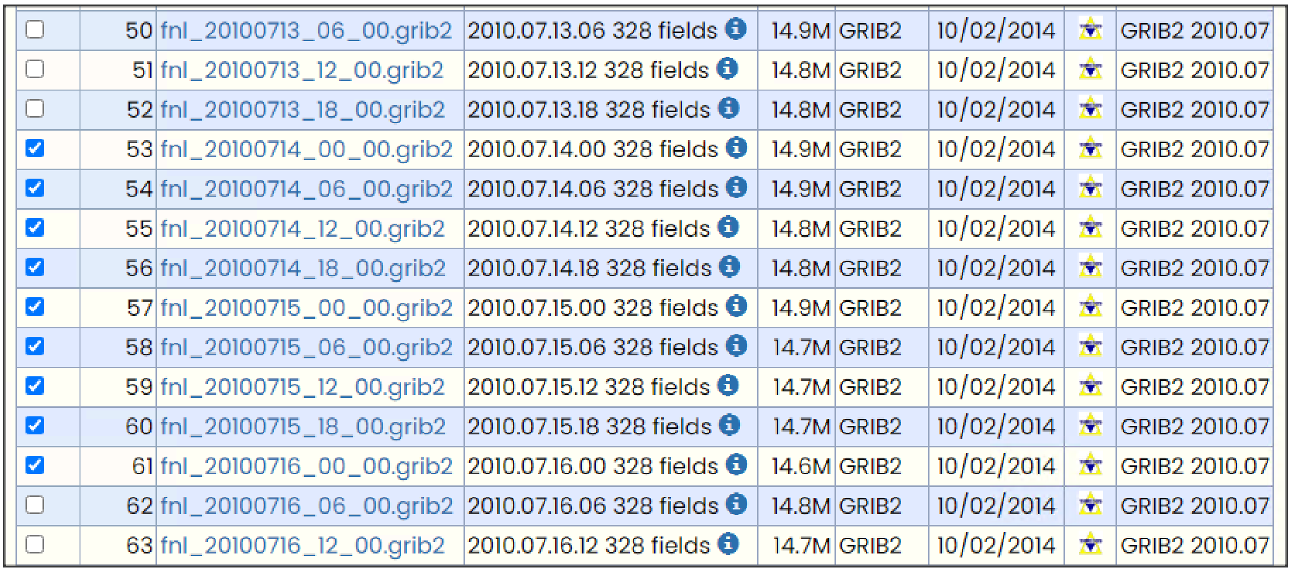
For help with this dataset, send us a message.

Description | Data Access | Documentation | Software | Metrics

Mouse over the table headings for detailed descriptions

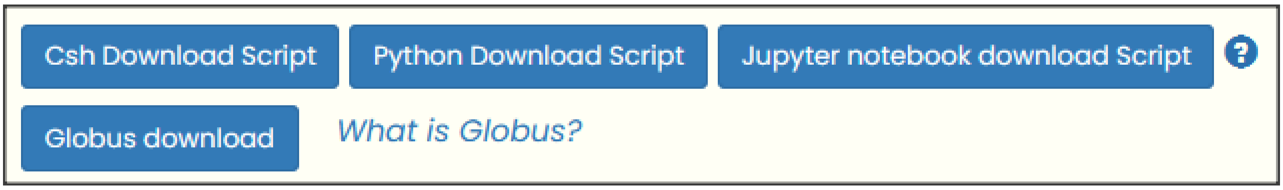
	Data Description	Data File Downloads		Customizable Data Requests	Other Access Methods	NCAR-Only Access
		Web Server Holdings	Data Format Conversion	Subsetting	THREDDS Data Server	Central File System (GLADE) Holdings
	Union of Available Products	Web File Listing	Get Converted Files	Get a Subset	TDS Access	GLADE File Listing
PRODUCTS	GRIB1 6 HOURLY FILES 1999.07.30 to 2007.12.06	Web File Listing	Get Converted Files	Get a Subset	TDS Access	GLADE File Listing
	GRIB2 6 HOURLY FILES 2007.12.06 to current	Web File Listing	Get Converted Files	Get a Subset	TDS Access	GLADE File Listing

Click Web File Listing



□	50	fnl_20100713_06_00.grib2	2010.07.13.06 328 fields ℹ	14.9M	GRIB2	10/02/2014	★	GRIB2 2010.07
□	51	fnl_20100713_12_00.grib2	2010.07.13.12 328 fields ℹ	14.8M	GRIB2	10/02/2014	★	GRIB2 2010.07
□	52	fnl_20100713_18_00.grib2	2010.07.13.18 328 fields ℹ	14.8M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	53	fnl_20100714_00_00.grib2	2010.07.14.00 328 fields ℹ	14.9M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	54	fnl_20100714_06_00.grib2	2010.07.14.06 328 fields ℹ	14.9M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	55	fnl_20100714_12_00.grib2	2010.07.14.12 328 fields ℹ	14.8M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	56	fnl_20100714_18_00.grib2	2010.07.14.18 328 fields ℹ	14.8M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	57	fnl_20100715_00_00.grib2	2010.07.15.00 328 fields ℹ	14.9M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	58	fnl_20100715_06_00.grib2	2010.07.15.06 328 fields ℹ	14.7M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	59	fnl_20100715_12_00.grib2	2010.07.15.12 328 fields ℹ	14.7M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	60	fnl_20100715_18_00.grib2	2010.07.15.18 328 fields ℹ	14.7M	GRIB2	10/02/2014	★	GRIB2 2010.07
☑	61	fnl_20100716_00_00.grib2	2010.07.16.00 328 fields ℹ	14.6M	GRIB2	10/02/2014	★	GRIB2 2010.07
□	62	fnl_20100716_06_00.grib2	2010.07.16.06 328 fields ℹ	14.8M	GRIB2	10/02/2014	★	GRIB2 2010.07
□	63	fnl_20100716_12_00.grib2	2010.07.16.12 328 fields ℹ	14.7M	GRIB2	10/02/2014	★	GRIB2 2010.07

Check fnl 2010714_00 to 20100716_00



Select Download Script and copy **download_ds083.2.csh** to directory

/home/ubuntu/Build_WRF/data/fnl

Change Directory to WPS directory.

```
$ cp /home/ubuntu/download_ds083.2.csh /home/ubuntu/Build_WRF/data/fnl
$ cd /home/ubuntu/Build_WRF/data/fnl
$ csh ./download_ds083.2.csh
$ ls -al fnl*
```

```
-rw-rw-r-- 1 ubuntu ubuntu 14924785 Jun 27 05:53 fnl_20100714_00_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14925135 Jun 27 05:53 fnl_20100714_06_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14828592 Jun 27 05:53 fnl_20100714_12_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14802287 Jun 27 05:53 fnl_20100714_18_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14898435 Jun 27 05:53 fnl_20100715_00_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14708117 Jun 27 05:53 fnl_20100715_06_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14657174 Jun 27 05:53 fnl_20100715_12_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14739475 Jun 27 05:53 fnl_20100715_18_00.grib2
-rw-rw-r-- 1 ubuntu ubuntu 14587482 Jun 27 05:53 fnl_20100716_00_00.grib2
```

5.10. Run WRF-Chem.

5.10.1. Running WPS

Change Directory to WPS directory.

```
$ cd /home/ubuntu/Build_WRF/WPS-4.4
```

Change GEOGRID.TBL

```
$ cd geogrid
$ rm -rf GEOGRID.TBL
$ ln -svf GEOGRID.TBL.ARW_CHEM GEOGRID.TBL
$ cd ..
```

Create namelist.wps file.

```
$ rm -rf namelist.wps
$ vi namelist.wps
```

```
&share
 wrf_core = 'ARW',
 max_dom = 1,
 start_date = '2010-07-14_00:00:00',
 end_date = '2010-07-16_00:00:00',
 interval_seconds = 10800,
 io_form_geogrid = 2,
/


&geogrid
 parent_id         =   1,
 parent_grid_ratio =   1,
 i_parent_start    =   1,
 j_parent_start    =   1,
 e_we              =   41,
 e_sn              =   41,
 geog_data_res     =   '10m',
 dx        = 100000,
 dy        = 100000,
 map_proj  =  'lambert',
 ref_lat   =  35.0,
 ref_lon   =  25.0,
 truelat1  =  30.0,
 truelat2  =  40.0,
 stand_lon =  25.0,
 geog_data_path = '/home/ubuntu/Build_WRF/geog'
/
 ref_x     =  20.5
 ref_y     =  20.5

&ungrib
 out_format = 'WPS',
 prefix = 'FILE',
/

&metgrid
 fg_name = 'FILE'
 io_form_metgrid = 2,
/
 opt_ignore_dom_center = .true.,

&mod_levs
 press_pa = 201300 , 200100 , 100000 ,
             95000 ,  90000 ,
             85000 ,  80000 ,
             75000 ,  70000 ,
             65000 ,  60000 ,
             55000 ,  50000 ,
             45000 ,  40000 ,
             35000 ,  30000 ,
             25000 ,  20000 ,
             15000 ,  10000 ,
              5000 ,   1000
/
```

Create Geography Data.

```
$ ./geogrid.exe
```

Wait for finish test by

```
$ ls -lah geo_em.d01.nc
```

If you see geo_em.d01.nc is coreect.

link Real-time Data top WPS.

```
$ ./link_grib.csh /home/ubuntu/Build_WRF/data/fnl/fnl_201007*
```

link Vtable

```
$ ln -sf ungrib/Variable_Tables/Vtable.GFS Vtable
```

create grib file.

```
$ ./ungrib.exe
```

Wait for finish test by

```
$ ls -lah FILE*
```

If you see FILE* is coreect.

create met file.

```
$ ./metgrid.exe
```

Wait for finish test by

```
$ ls -lah met_em.*
```

If you see met_em.* is coreect.

5.10.2. Running WRF.

Change Directory to WRF directory.

```
$ cd /home/ubuntu/Build_WRF/WRF/test/em_real/
```

link met file from WPS to WRF.

```
$ ln -sf /home/ubuntu/Build_WRF/WPS-4.4/met_em* .
```

remove namelist file.

```
$ rm -rf namelist.input
```

Edit namelist.input

```
$ vi namelist.input
```

```
&time_control
 run_days                            = 2,
 run_hours                           = 0,
 run_minutes                           = 0,
 run_seconds                           = 0,
 start_year                          = 2010,
 start_month                           = 07,
 start_day                           = 14,
 start_hour                          = 00,
 start_minute                          = 00,
 start_second                          = 00,
 end_year                            = 2010,
 end_month                             = 07,
 end_day                             = 16,
 end_hour                            = 00,
 end_minute                            = 00,
 end_second                            = 00,
 interval_seconds                      = 10800,
 input_from_file                      = .true.,
 history_interval                    = 60,
 frames_per_outfile                    = 72,
 restart                          = .false.,
 restart_interval                   = 0,
 io_form_history                     = 2,
 io_form_restart                     = 2,
 io_form_input                       = 2,
 io_form_boundary                       = 2,
 auxinput6_inname                       = 'wrfbiochemi_d01',
 auxinput7_inname                       = 'wrffirechemi_d<domain>',
 auxinput8_inname                       = 'wrfchemi_gocart_bg_d<domain>',
 auxinput12_inname                       = 'wrf_chem_input',
auxinput5_interval_m                   = 86400,
 auxinput7_interval_m                   = 86400,
 auxinput8_interval_m                   = 86400,
 io_form_auxinput2                     = 2,
 io_form_auxinput5                     = 0,
 io_form_auxinput6                     = 0,
 io_form_auxinput7                     = 0,
 io_form_auxinput8                     = 0,
 io_form_auxinput12                     = 0,
 debug_level                         = 0,
 auxinput1_inname                       = "met_em.d<domain>.<date>",
 auxinput13_inname                       = 'wrfchemv_d<domain>',
 auxinput13_interval_m                   = 86400,
 io_form_auxinput13                     = 0,
/

 &dfi_control
/

 &domains
 time_step                          = 600,
 time_step_fract_num                     = 0,
 time_step_fract_den                    = 1,
 max_dom                              = 1,
 s_we                              = 1,
 e_we                              = 41,
s_sn                             = 1,
 e_sn                             = 41,
 e_vert                            = 31,
 num_metgrid_levels                   = 27,
 num_metgrid_soil_levels              = 4,
 dx                              = 100000,
 dy                              = 100000,
 grid_id                          = 1,
 parent_id                          = 0,
 i_parent_start                      = 1,
 j_parent_start                      = 1,
 parent_grid_ratio                    = 1,
 parent_time_step_ratio                = 1,
 p_top_requested                      = 10000,
 feedback                           = 1,
 smooth_option                        = 0,
 p_top_requested                      = 10000,
 zap_close_levels                    = 50,
 interp_type                         = 1,
 t_extrap_type                       = 2,
 force_sfc_in_vinterp                 = 0,
 use_levels_below_ground               = .true.,
 use_surface                          = .true.,
 lagrange_order                       = 1,
 sfcp_to_sfcp                         = .true.,
 /

 &physics
 num_land_cat                         = 21,
 mp_physics                           = 4,
progn                             = 0,
 ra_lw_physics                        = 1,
 ra_sw_physics                    = 2,
 radt                          = 30,
 sf_sfclay_physics                = 1,
 sf_surface_physics                = 2,
 bl_pbl_physics                   = 1,
 bldt                          = 0,
 cu_physics                      = 5,
 cu_diag                        = 1,
 cudt                          = 0,
 ishallow                       = 0,
 isfflx                        = 1,
 ifsnow                        = 1,
 icloud                        = 1,
 surface_input_source               = 1,
 num_soil_layers                   = 4,
 sf_urban_physics                  = 0,
 mp_zero_out                      = 2,
 mp_zero_out_thresh                  = 1.e-12,
 maxiens                         = 1,
 maxens                          = 3,
 maxens2                         = 3,
 maxens3                         = 16,
 ensdim                          = 144,
 cu_rad_feedback                   = .true.,
/

 &fdda
/

 &dynamics
 rk_ord                         = 3,
 w_damping                        = 1,
 diff_opt                       = 1,
 km_opt                          = 4,
 diff_6th_opt                    = 0,
 diff_6th_factor                 = 0.12,
 base_temp                         = 290.
 damp_opt                          = 0,
 zdamp                          = 5000.,
 dampcoef                          = 0.01,
 khdif                       = 0,
 kvdif                       = 0,
 non_hydrostatic                    = .true.,
 moist_adv_opt                       = 2,
 scalar_adv_opt                     = 2,
 chem_adv_opt                          = 2,
 tke_adv_opt                       = 2,
 time_step_sound                       = 4,
 h_mom_adv_order                        = 5,
 v_mom_adv_order                        = 3,
 h_sca_adv_order                      = 5,
 v_sca_adv_order                      = 3,
/

 &bdy_control
 spec_bdy_width                       = 5,
 spec_zone                         = 1,
relax_zone                         = 4,
 specified                         = .true.,
 nested                            = .false.,
/

 &grib2
/

 &namelist_quilt
 nio_tasks_per_group = 0,
 nio_groups = 1,
/



 &chem
 kemit                              = 1,
 chem_opt                              = 401,
 bioemdt                              = 0,
 photdt                              = 0,
 chemdt                               = 10,
 io_style_emissions                     = 0,
 emiss_opt                             = 3,
 emiss_opt_vol                          = 0,
 emiss_ash_hgt                          = 20000.,
 chem_in_opt                            = 0,
 phot_opt                              = 0,
 gas_drydep_opt                          = 0,
 aer_drydep_opt                          = 1,
 bio_emiss_opt                          = 0,
ne_area                              = 0,
 dust_opt                             = 1,
 dmsemis_opt                            = 0,
 seas_opt                             = 0,
 depo_fact                             = 0.25,
 gas_bc_opt                             = 0,
 gas_ic_opt                            = 0,
 aer_bc_opt                             = 1,
 aer_ic_opt                            = 1,
 gaschem_onoff                          = 0,
 aerchem_onoff                          = 1,
 wetscav_onoff                          = 0,
 cldchem_onoff                          = 0,
 vertmix_onoff                         = 1,
 chem_conv_tr                           = 0,
 conv_tr_wetscav                         = 0,
 conv_tr_aqchem                        = 0,
 biomass_burn_opt                      = 0,
 plumerisefire_frq                   = 30,
 have_bcs_chem                         = .false.,
 aer_ra_feedback                      = 0,
 aer_op_opt                          = 0,
 opt_pars_out                        = 0,
 diagnostic_chem                      = 0,
/
```

create real case.

```
$ mpirun -np 1 ./real.exe
```

Wait for finish test by

```
$ tail rsl.error.0000
```

If you see real_em: SUCCESS COMPLETE REAL_EM INIT is coreect.

And see file wrfbdy_d01 wrfinput_d01.

```
$ ls -alh wrfbdy_d01 wrfinput_d01
```

If you see wrfbdy_d01 and wrfinput_d01 is coreect.

Run WRF

```
$ mpirun -np 2 ./wrf.exe
```

Wait for finish test by

```
$ tail rsl.error.0000
```

If you see wrf: SUCCESS COMPLETE WRF is coreect.

And see file wrfout_*

```
$ ls -alh wrfout_*
```

If you see wrfout_* is coreect.

5.10.3. list data inside wrfout.

```
$ ncdump -h wrfout_d01*
$ ncdump -v DUST_5 wrfout_d01*
```

5.10.4. Create pdf from wrfout.

Change Directory

```
$ cd /home/ubuntu/Build_WRF/WRF/test/em_real
```

Create ncl script.

```
$ vi plot_dust_5.ncl
```

```
;************************************************
; WRF: DUST_5
;***********************************************
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_csm.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/wrf/WRF_contributed.ncl"


begin
  f     = addfile ("wrfout_d01_2010-07-14_00:00:00", "r")


  wks = gsn_open_wks("pdf" ,"WRF_DUST_5")      ; ps,pdf,x11,ncgm,eps
  gsn_define_colormap(wks,"BlAqGrYeOrReVi200") ; select color map


  res                        = True            ; plot mods desired
  res@gsnMaximize            = True            ; uncomment to maximize size
  res@gsnSpreadColors        = True            ; use full range of colormap
  res@cnFillOn               = True            ; color plot desired
  res@cnLinesOn              = False           ; turn off contour lines
  res@cnLineLabelsOn         = False           ; turn off contour labels
  WRF_map_c(f, res, 0)                    ; reads info from file


  res@tfDoNDCOverlay       = True
  res@pmTickMarkDisplayMode = "Always"        ; turn on tickmarks


  times = wrf_user_getvar(f,"times",-1)       ; get all times in the file
  ntimes = dimsizes(times)                    ; number of times in the file


  do nt=0,ntimes,6
    x = f->DUST_5(nt,0,:,:)                  ; (Time,level, south_north, west_east)
    res@tiMainString      = "WRF-CHEM (DUST_5) " + times(nt)
    res@gsnLeftString     = x@description
    plot                  = gsn_csm_contour_map(wks,x(:,:),res)
  end do
end
```

Create PDF

```
$ ncl plot_dust_5.ncl
```

Output is **WRF_DUST_5.pdf,** you can open pdf file from pdf viewer or download to your laptop or type “**evince WRF_DUST_5.pdf**”.

6. Conclusion.

We can setup WRF-Chem and KPP. And thank you website

- https://wiki.harvard.edu/confluence/pages/viewpage.action?pageId=228526205
- https://github.com/whatheway/WRFCHEM-4.4-install-script-linux-64bit/blob/main/WRF_CHEMKPP_INSTALL_64BIT.sh
- https://ruc.noaa.gov/wrf/wrf-chem/tutorialexercises/tutorialexercises001.html
- https://www.ncl.ucar.edu/Training/Tutorials/WRF_Users_Workshop/WRF_NCL.shtml

for information.

Thank you.