Installing WRF & WPS

Kelly Keene

NCAR/MMM

January 2015

Installing Steps

- *Check system requirements*
- Installing Libraries
- Download source data
- Download datasets
- Compile WRFV3
- Compile WPS

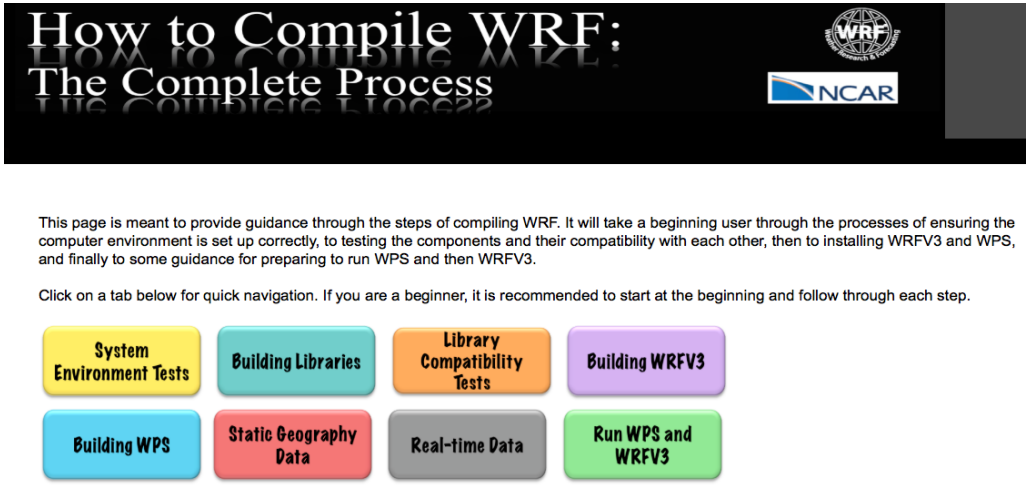
System Requirements

- On what kinds of systems will WRF run?
  - Generally any 32- or 64-bit hardware, running a UNIX-like operating system
  - You may also use dual-booting into a UNIX-like OS (e.g., Windows with Linux build parallel)

- Examples of acceptable systems:
  - Laptops, desktops, and clusters running Linux
  - Laptops and desktops running MacOS X
  - Clusters running Linux or AIX (Cray, IBM, SGI, etc.)

Check System Requirements

- Webpage: http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compilation_tutorial.php



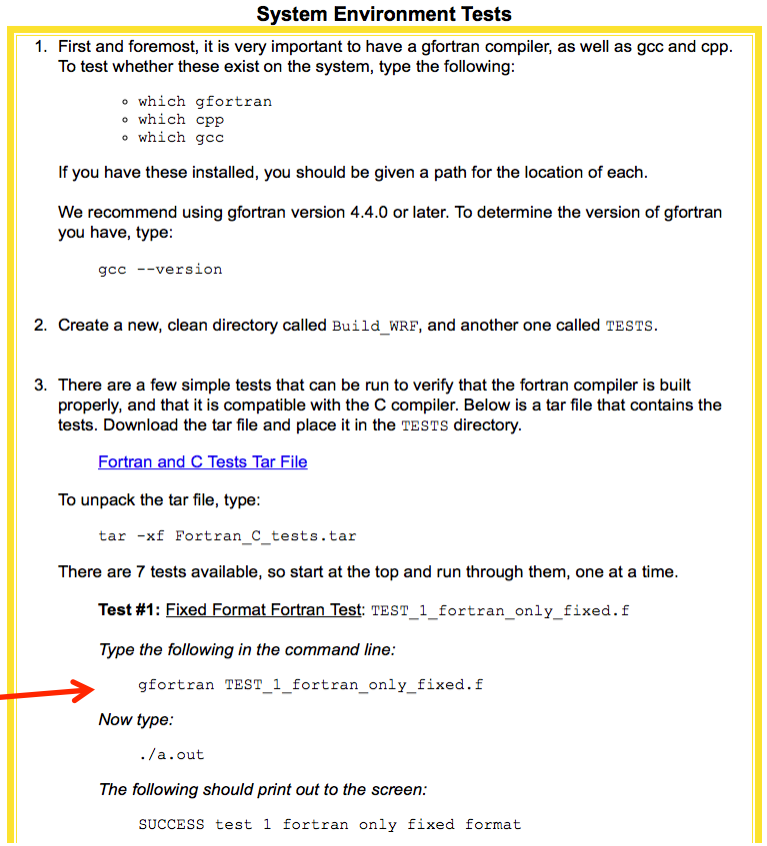
Check System Requirements

- It is mandatory to have a Fortran (e.g., gfortran) compiler, gcc, and cpp on your system. Check by typing:
  - which gfortran
  - which cpp
  - which gcc
  - If installed, you will be given a path for each
- Fortran compiler should be version 4.4.0, or later

  Check this by typing

  (for csh):

  ```
  gcc –version
  ```
- Tests available for checking that your fortran compiler is built properly, and that it is compatible with the C compiler.



System Environment Tests

1. First and foremost, it is very important to have a gfortran compiler, as well as gcc and cpp. To test whether these exist on the system, type the following:

   - `which gfortran`
   - `which cpp`
   - `which gcc`

   If you have these installed, you should be given a path for the location of each.

   We recommend using gfortran version 4.4.0 or later. To determine the version of gfortran you have, type:

   ```
   gcc --version
   ```

2. Create a new, clean directory called `Build_WRF`, and another one called `TESTS`.

3. There are a few simple tests that can be run to verify that the fortran compiler is built properly, and that it is compatible with the C compiler. Below is a tar file that contains the tests. Download the tar file and place it in the `TESTS` directory.

   Fortran and C Tests Tar File

   To unpack the tar file, type:

   ```
   tar -xf Fortran_C_tests.tar
   ```

   There are 7 tests available, so start at the top and run through them, one at a time.

   Test #1: Fixed Format Fortran Test: `TEST_1_fortran_only_fixed.f`

   Type the following in the command line:

   ```
   gfortran TEST_1_fortran_only_fixed.f
   ```

   Now type:

   ```
   ./a.out
   ```

   The following should print out to the screen:

   ```
   SUCCESS test 1 fortran only fixed format
   ```

Additional Necessary Requirements

- Scripting languages (testing available in test package):
  - csh
  - perl
  - sh

- UNIX commands:

ar	head	sed	awk
hostname	sleep	cat	ln
sort	cd	ls	tar
cp	make	touch	cut
mkdir	tr	expr	mv
uname	file	nm	wc
grep	printf	which	gzip
rm			

Installing Steps

- Check system requirements
- *Installing Libraries*
- Download source data
- Download datasets
- Compile WRFV3
- Compile WPS

Installing Libraries

- NetCDF (needed by WRF and WPS)
- Optional libraries for GRIB2 met data support
  - JasPer (JPEG 2000 “lossy” compression library)
  - PNG (“lossless” compression library)
  - Zlib (compression library used by PNG)
- Optional MPI library (for building in parallel):
  - MPICH2

Installing Libraries

- Installation of these libraries (MPICH2, NetCDF, JasPer, zlib, and libpng) is NOT part of the WPS and WRF installation scripts
- VERY IMPORTANT!
  - Make sure these libraries are installed using the same compilers as will be used to install WRF and WPS
- Downloads for the libraries, with installation instructions, and library compatibility tests are also included on the compilation website

Installing Libraries: NetCDF

```
setenv DIR directory-where-your-tar-files-are
setenv CC gcc
setenv CXX g++
setenv FC gfortran
setenv FCFLAGS -m64      # FCFLAGS may be needed on some systems
setenv F77 gfortran
setenv FFLAGS -m64       # FFLAGS may be needed on some systems

tar xzvf netcdf-4.1.3.tar.gz       # no '.gz' if downloaded to most Macs
cd netcdf-4.1.3
./configure --prefix=$DIR/netcdf --disable-dap  --disable-netcdf-4 --
disable-shared
make
make install
setenv PATH $DIR/netcdf/bin:$PATH
setenv NETCDF $DIR/netcdf
cd ..
```

Installing Libraries: MPICH2

- In principle, any implementation of the MPI-2 standard should work with WRF; however, we have the most experience with MPICH

- *Assuming environment variables for netCDF install are already set:*

```
tar xzvf mpich-3.0.4.tar.gz   # no '.gz' if downloaded to most Macs
cd mpich-3.0.4
./configure --prefix=$DIR/mpich
make
make install
setenv PATH $DIR/mpich/bin:$PATH
cd ..
```

Installing Libraries: zlib

- *Assuming environment variables from netCDF install are already set:*

```
tar xzvf zlib-1.2.7.tar.gz          # no '.gz' if downloaded to most Macs
cd zlib-1.2.7
./configure --prefix=$DIR/zlib
make
make install
cd ..
```

Installing Libraries: libpng

- *Assuming environment variables from netCDF install are already set*

```
tar xzvf libpng-1.2.50.tar.gz      # no '.gz' if downloaded to most Macs
cd libpng-1.2.50
./configure --prefix=$DIR/libpng
make
make install
cd ..
```

Installing Libraries: JasPer

- *Assuming environment variables from netCDF install are already set*

```
tar xzvf jasper-1.900.1.tar.gz     # no '.gz' if downloaded to most Macs
cd jasper-1.900.1
./configure --prefix=$DIR/jasper
make
make install
cd ..
```

Installing Steps

- Check system requirements
- Installing Libraries
- *Download source data*
- Download datasets
- Compile WRFV3
- Compile WPS

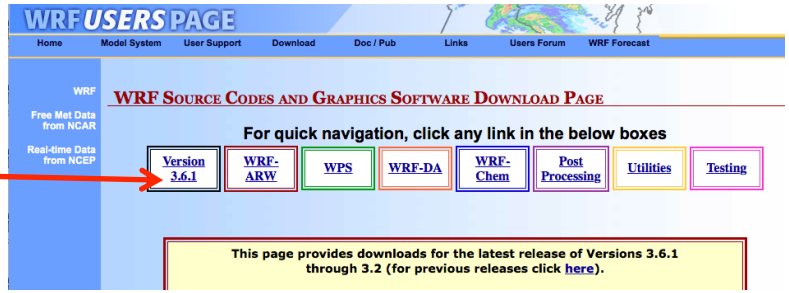
Download WRF & WPS Code

- Download WRF & WPS source code from:

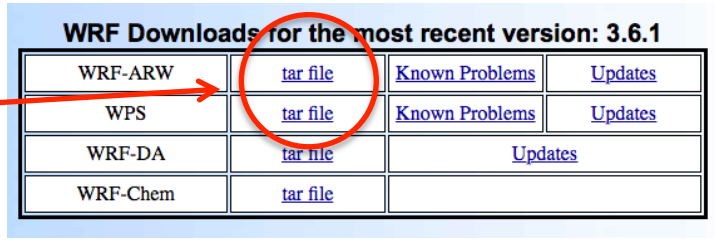
  http://www2.mmm.ucar.edu/wrf/users/download/get_source.html

  – Click 'New User,' register and download, or
  – Click 'Returning User,' enter your email, and download

Step 1:
Click here for the latest released code (recommended)

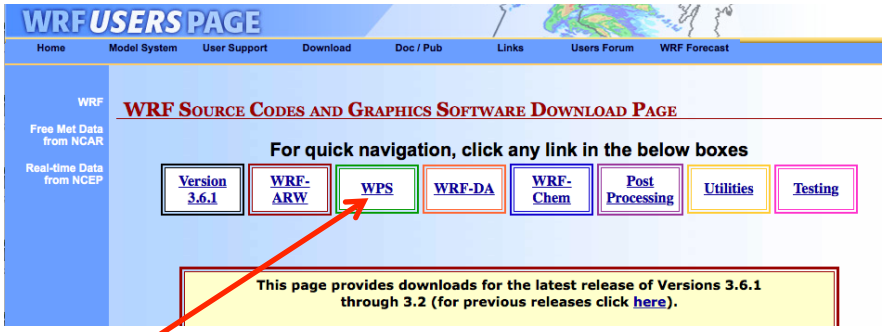


Step 2:
Click on tar files to download



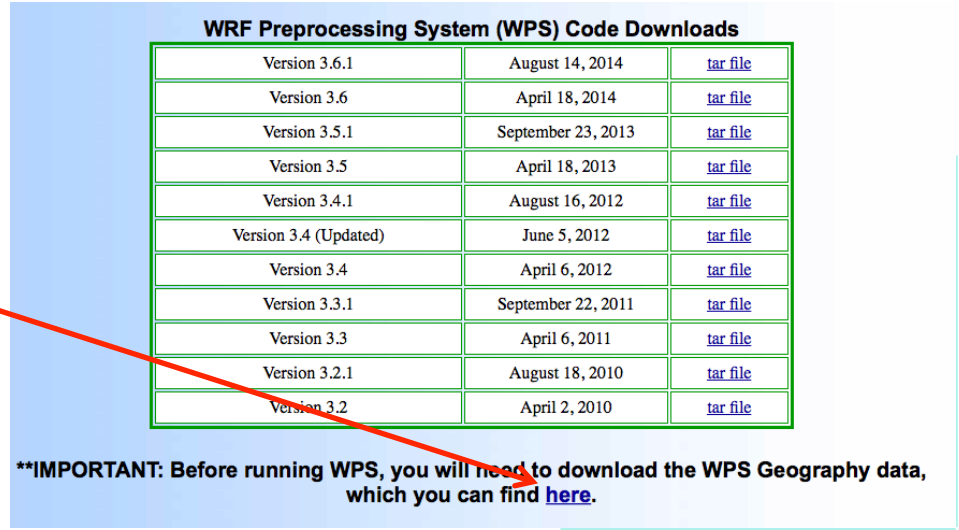
Download Static Geographical Data

- From the WRF Download page: http://www.mmm.ucar.edu/wrf/users/download/get_sources.html



Step 1: Click 'WPS' box

Step 2: Click 'here' to get geography data



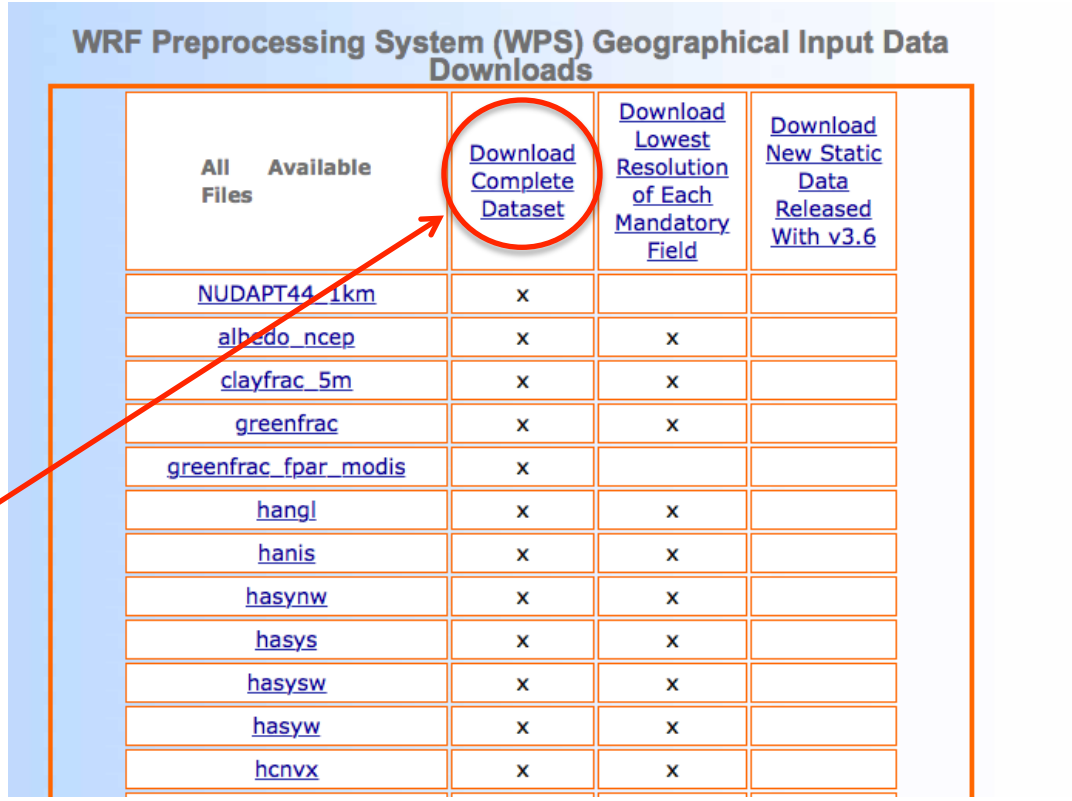
Download Static Geographical Data

- Geographical Input and Data Download Page:

http://www.mmm.ucar.edu/wrf/users/download/get_sources_wps_geog.html

geog.tar.gz
~ 15 GB when uncompressed

This is the one you want



WRF Preprocessing System (WPS) Geographical Input Data Downloads

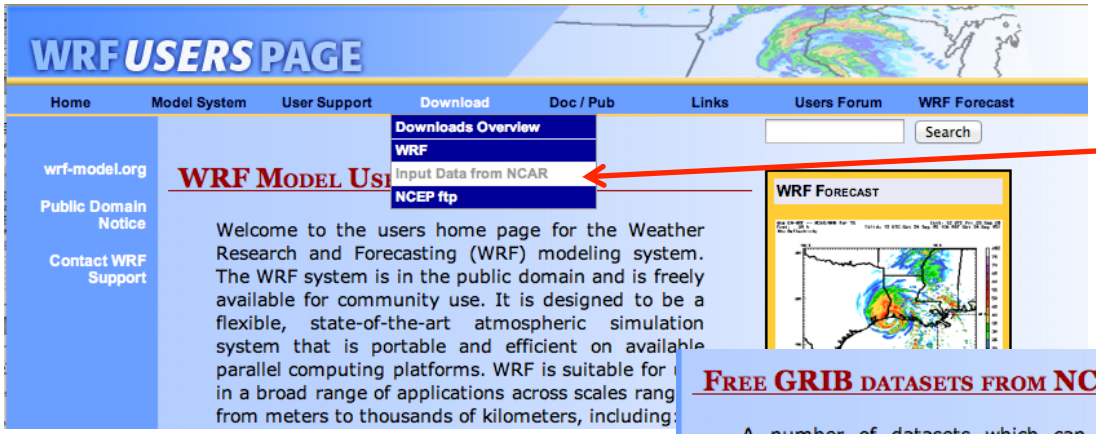
All Available Files	Download Complete Dataset	Download Lowest Resolution of Each Mandatory Field	Download New Static Data Released With v3.6
NUDAPT44_1km	x		
albedo_ncep	x	x	
clayfrac_5m	x	x	
greenfrac	x	x	
greenfrac_fpar_modis	x		
hangl	x	x	
hanis	x	x	
hasynw	x	x	
hasys	x	x	
hasysw	x	x	
hasyw	x	x	
hcnvx	x	x	

Installing Steps

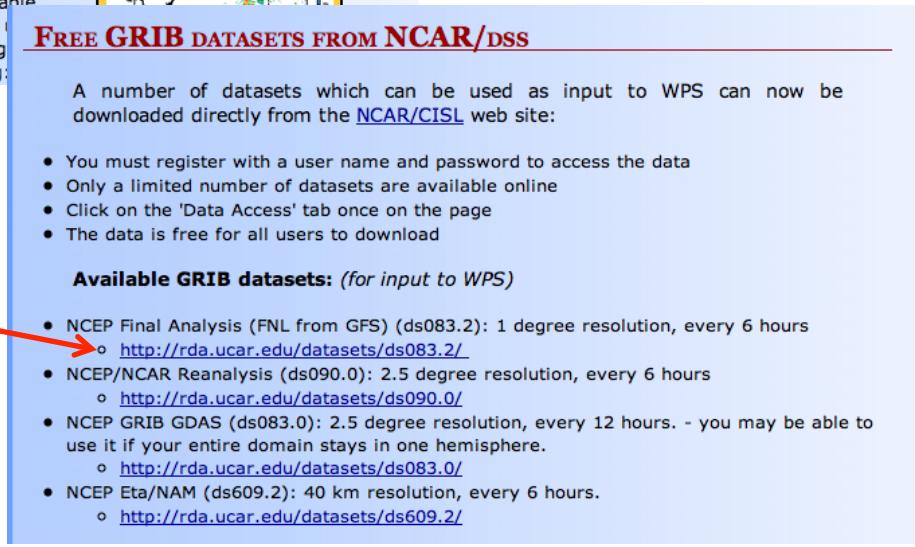
- Check system requirements
- Installing Libraries
- Download source data
- *Download datasets*
- Compile WRFV3
- Compile WPS

Download Datasets

- From the WRF Users' page: http://www.mmm.ucar.edu/wrf/users/



Step 1: Click Download, then scroll down and click 'Input Data from NCAR'

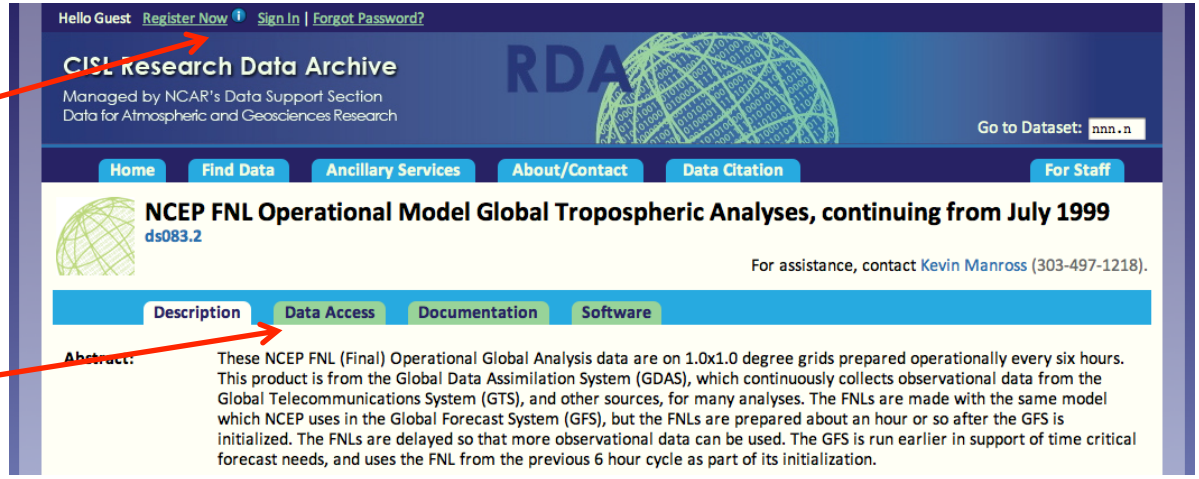


Step 2: Click the dataset you wish to use (for this example, we will use 'FNL from GFS')

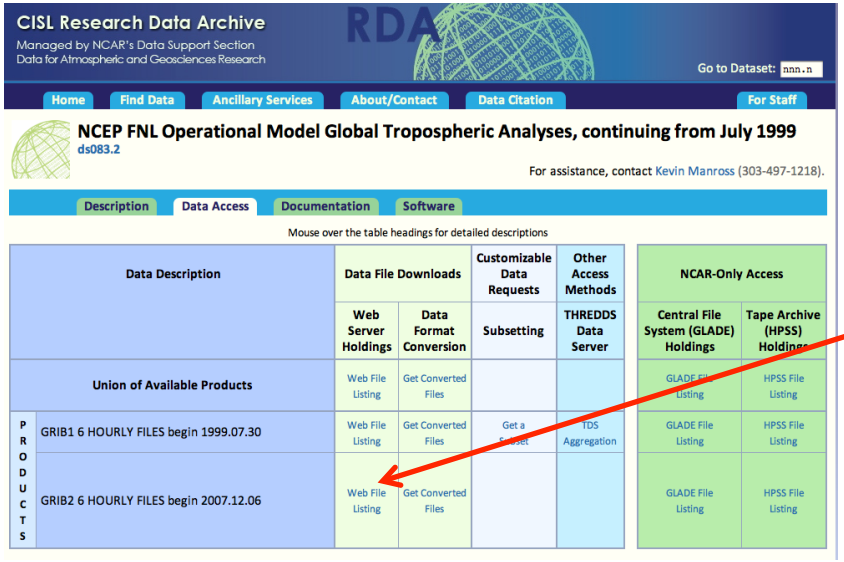
*Note: NOMADS site has several types of useful data: http://nomads.ncdc.noaa.gov

Download Datasets (continued)

Step 3: Register, or sign in, if you already have an account



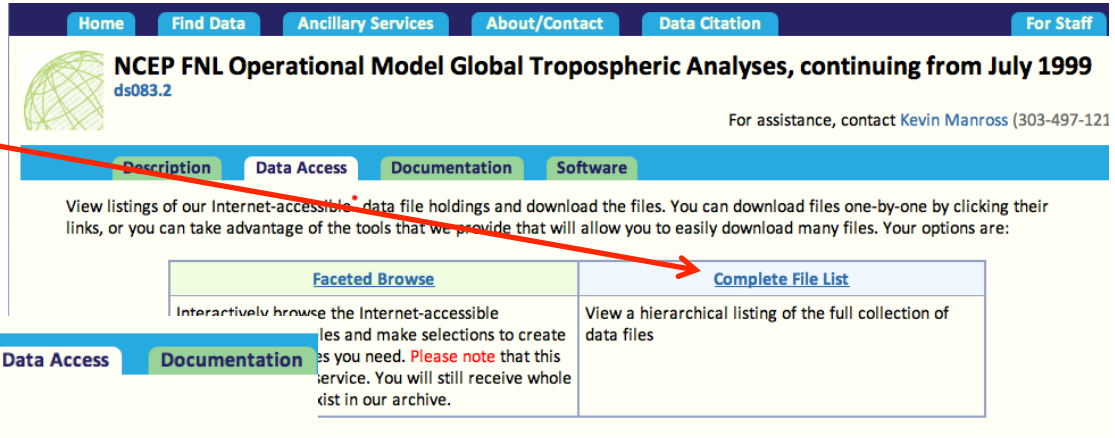
Step 4: Click 'Data Access'

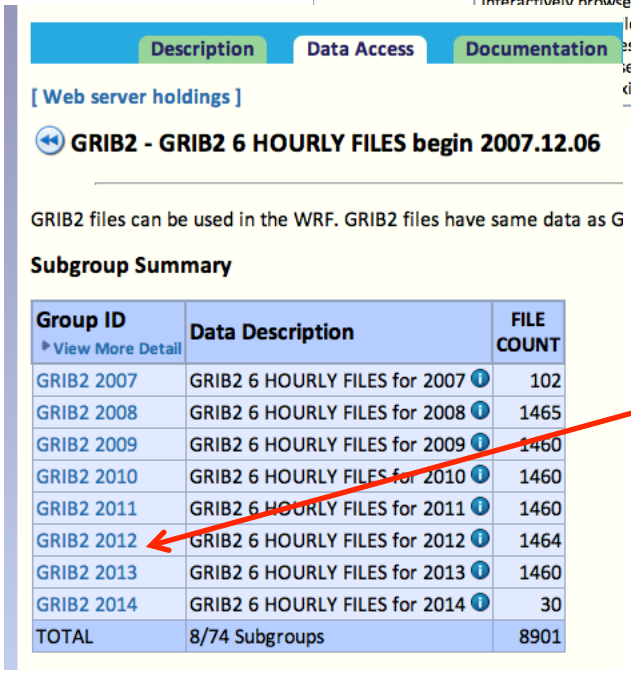


Step 5: Click 'Web File Listing' for the span of years you need

Download Datasets (continued)

Step 6: Click 'Complete File List'

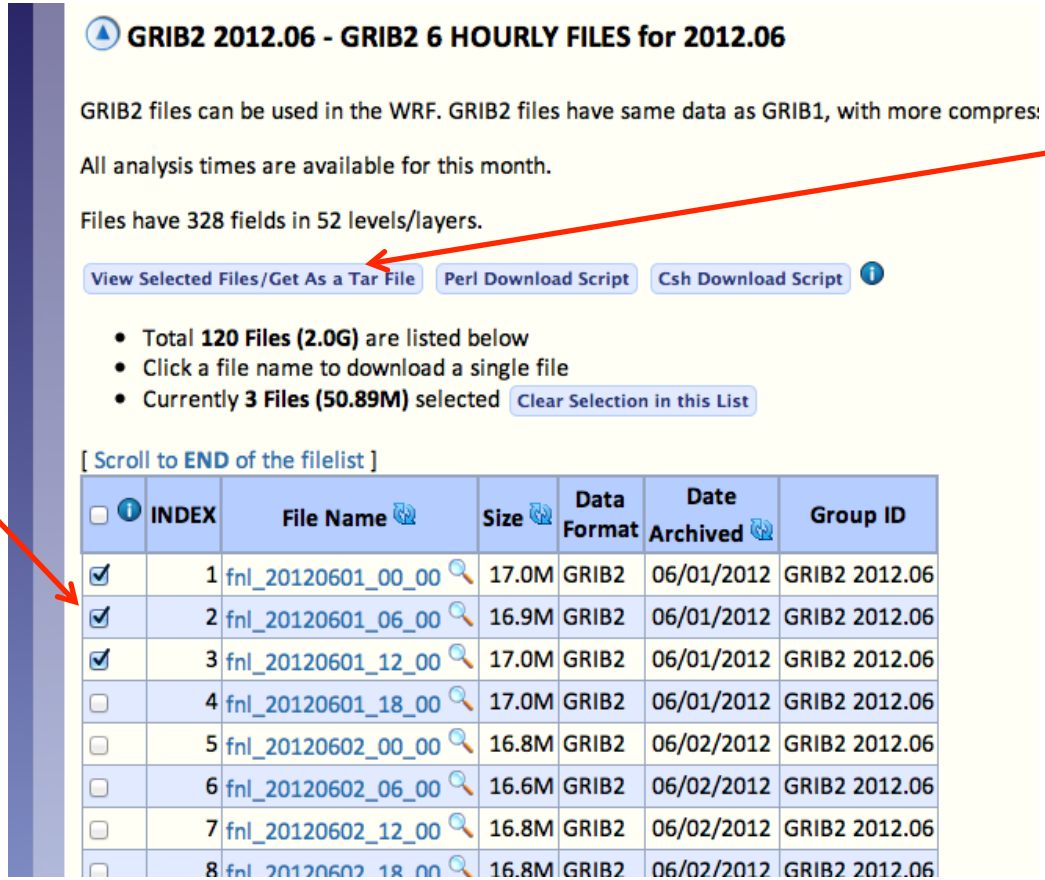




Step 7: Click the year you need. After this, You will click the month you need (not shown)

Download Datasets (continued)

Step 8: Click a box for each time span that you need



GRIB2 2012.06 - GRIB2 6 HOURLY FILES for 2012.06

GRIB2 files can be used in the WRF. GRIB2 files have same data as GRIB1, with more compres

All analysis times are available for this month.

Files have 328 fields in 52 levels/layers.

View Selected Files/Get As a Tar File | Perl Download Script | Csh Download Script

- Total **120 Files (2.0G)** are listed below
- Click a file name to download a single file
- Currently **3 Files (50.89M)** selected Clear Selection in this List

[Scroll to **END** of the filelist]

	INDEX	File Name	Size	Data Format	Date Archived	Group ID
☑	1	fnl_20120601_00_00	17.0M	GRIB2	06/01/2012	GRIB2 2012.06
☑	2	fnl_20120601_06_00	16.9M	GRIB2	06/01/2012	GRIB2 2012.06
☑	3	fnl_20120601_12_00	17.0M	GRIB2	06/01/2012	GRIB2 2012.06
☐	4	fnl_20120601_18_00	17.0M	GRIB2	06/01/2012	GRIB2 2012.06
☐	5	fnl_20120602_00_00	16.8M	GRIB2	06/02/2012	GRIB2 2012.06
☐	6	fnl_20120602_06_00	16.6M	GRIB2	06/02/2012	GRIB2 2012.06
☐	7	fnl_20120602_12_00	16.8M	GRIB2	06/02/2012	GRIB2 2012.06
☐	8	fnl_20120602_18_00	16.8M	GRIB2	06/02/2012	GRIB2 2012.06

Step 9: Once you have chosen All your times, click on the 'View Selected Files/Get As a Tar File' button
To download one tar file with all your Dates/times

Installing Steps

- Check system requirements
- Installing Libraries
- Download source data
- Download datasets
- *Compile WRFV3*
- Compile WPS

Before Compiling

- Check where your netCDF library and include file are. If it is not in the usual location (i.e., */usr/local/netcdf),* then use the NETCDF environment variable to set the path. e.g., for a C-shell environment:

```
setenv NETCDF /where-netcdf-is
```

- Unpack your remaining tar files:

You should now have the following directories:

```
gunzip WPSV3.6.1.TAR.gz
tar -xf WPSV3.6.1.TAR  ------------->   WPS

gunzip WRFV3.6.1.TAR.gz
tar -xf WRFV3.6.1.TAR  ------------->   WRFV3

gunzip geog.tar.gz
tar -xf geog.tar  ------------------->  WPS_GEOG
```

*Useful Tip:
Create another directory inside here, called 'TAR_FILES' and place all the tar files in there to keep things organized.

Compile WRFV3

- It is important to compile WRFV3 **first**, before WPS
  - WPS makes use of the external I/O libraries in the *WRFV3/external* directory
  - The libraries are built when WRF is installed

- Two steps to compiling:

  1) Create a configuration file for your computer and compiler

  ```
  ./configure
  ```

  2) Compile the code

  ```
  ./compile test_case >& log.compile
  ```

Step 1: Configure for WRFV3

- Inside the WRFV3/ directory, type: **`./configure`**

```
checking for perl5... no
checking for perl... found /usr/bin/perl (perl)
Will use NETCDF in dir: /usr/local/netcdf
PHDF5 not set in environment. Will configure WRF for use without.
configure:  WRF operating system set to "Linux" via environment variable $WRF_OS
configure:  WRF machine set to "i686" via environment variable $WRF_MACH
$JASPERLIB or $JASPERINC not found in environment, configuring to build without grib2 I/
    O...
------------------------------------------------------------------------
Please select from among the following supported platforms.

  1.  Linux i486 i586 i686, gfortran compiler with gcc  (serial)
  2.  Linux i486 i586 i686, gfortran compiler with gcc  (smpar)
  3.  Linux i486 i586 i686, gfortran compiler with gcc  (dmpar)
  4.  Linux i486 i586 i686, gfortran compiler with gcc  (dm+sm)
  5.  Linux i486 i586 i686, g95 compiler with gcc  (serial)
  6.  Linux i486 i586 i686, g95 compiler with gcc  (dmpar)
  7.  Linux i486 i586 i686, PGI compiler with gcc  (serial)
  8.  Linux i486 i586 i686, PGI compiler with gcc  (smpar)
  9.  Linux i486 i586 i686, PGI compiler with gcc  (dmpar)
  10.  Linux i486 i586 i686, PGI compiler with gcc  (dm+sm)
  11.  Linux x86_64 i486 i586 i686, ifort compiler with icc  (serial)
  12.  Linux x86_64 i486 i586 i686, ifort compiler with icc  (smpar)
  13.  Linux x86_64 i486 i586 i686, ifort compiler with icc  (dmpar)
  14.  Linux x86_64 i486 i586 i686, ifort compiler with icc  (dm+sm)

Enter selection [1-16] :

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

- Output from configuration: a file called ‘configure.wrf’

Step 2: Compile WRFV3

- In the WRFV3/ directory, type:

  `./compile em_case >& log.compile`

Important in case there are compile problems

Where **`em_case`** is one of the following
(type **`./compile`** to see all options)

`em_real` (3d real case)

3d Ideal:
- **`em_quarter_ss`**
- **`em_b_wave`**
- **`em_les`**
- **`em_heldsuarez`**
- **`em_tropical_cyclone`**

2d Ideal:
- **`em_hill2d_x`**
- **`em_squall2d_x`**
- **`em_squall2d_y`**
- **`em_grav2d_x`**
- **`em_seabreeze2d_x`**

`em_scm_xy` (1d ideal)

- Compilation should take at most 20-30 mins

Successful Compilation

- If the compilation is successful, you should find these executables in **`WRFV3/main`** (non-zero size):

  Real data case:

  `wrf.exe` – model executable

  `real.exe` – real data initialization

  `ndown.exe` – one-way nesting

  `tc.exe` – for tc bogusing (serial only)

  Ideal case:

  `wrf.exe` – model executable

  `ideal.exe` – ideal case initialization

  ***Note: Each ideal case compile creates a different executable, but with the same name**

- These executables are linked to 2 different directories (**`WRFV3/run`** and **`WRFV3/test/em_real`**). You can go to either place to run WRF.

Installing Steps

- Check system requirements
- Installing Libraries
- Download source data
- Download datasets
- Compile WRFV3
- *Compile WPS*

Compile WPS

- Once WRFV3 is successfully compiled, change directory to WPS prior to compiling

```
cd ../WPS
```

- Two steps to compiling WPS, as well:

  1) Create a configuration file for your computer

```
./configure
```

  2) Compile the code

```
./compile >& log.compile
```

Step 1: Configure for WPS

- Inside the WPS/ directory, type: **`./configure`**

```
Will use NETCDF in dir: /usr/local/netcdf-pgi
$JASPERLIB or $JASPERINC not found in environment. Using /usr/local for library paths...
------------------------------------------------------------------------
Please select from among the following supported platforms.
   1.  Linux i486 i586 i686, PGI compiler     (serial)
   2.  Linux i486 i586 i686, PGI compiler     (serial_NO_GRIB2)
   3.  Linux i486 i586 i686, PGI compiler     (dmpar)
   4.  Linux i486 i586 i686, PGI compiler     (dmpar_NO_GRIB2)
   5.  Linux i486 i586 i686, Intel compiler     (serial)
   6.  Linux i486 i586 i686, Intel compiler     (serial_NO_GRIB2)
   7.  Linux i486 i586 i686, Intel compiler     (dmpar)
   8.  Linux i486 i586 i686, Intel compiler     (dmpar_NO_GRIB2)
   9.  Linux i486 i586 i686, g95    (serial)
  10.  Linux i486 i586 i686, g95    (serial_NO_GRIB2)
  11.  Linux i486 i586 i686, g95    (dmpar)
  12.  Linux i486 i586 i686, g95    (dmpar_NO_GRIB2)
  13.  Linux i486 i586 i686, gfortran     (serial)
  14.  Linux i486 i586 i686, gfortran    (serial_NO_GRIB2)
  15.  Linux i486 i586 i686, gfortran    (dmpar)
  16.  Linux i486 i586 i686, gfortran    (dmpar_NO_GRIB2)
Enter selection [1-16] :
```

- Choose to compile WPS **serially** (even if you compiled WRFV3 in parallel), unless you plan to use very large domains

  *NOTE: If you compile WPS in parallel, ungrib.exe must still be run serially

- Output from configuration: a file called 'configure.wps'

Step 2: Compile WPS

- In the WPS/ directory, type:

```
./compile >& log.compile
```

- Compilation should only take a few minutes

- If successful, these executables should be in your WPS/ directory (and they are linked, respectively, from their source code directories):

```
geogrid.exe -> geogrid/src/geogrid.exe
ungrib.exe -> ungrib/src/ungrib.exe
metgrid.exe -> metgrid/src/metgrid.exe
```

./clean -a

- The ‘./clean –a’ command is something that should be used when you have made corrections to your configure.wrf file, configure.wps file, or any changes to the registry. If you have made any of these changes, or if you plan to recompile your code from scratch, you must issue a ‘clean –a’ before recompiling.

- If you made any changes to any subroutines within the code, you will need to recompile your code, but you do NOT need to issue the ‘clean –a’ command, nor do you need to reconfigure. You will simply just recompile. This compilation should take a lot less time than a clean compile.

Questions?