

**CPC Search**

CPC search

Go

[HOME](#) > [Monitoring\\_and\\_Data](#) > [Oceanic and Atmospheric Data](#) > [Reanalysis: Atmospheric Data](#) > wgrib2

**About Us**

[Our Mission](#)  
[Who We Are](#)

**Contact Us**

[CPC Information](#)  
[CPC Web Team](#)

## wgrib2: wgrib for GRIB-2

### Utility to read and write grib2 files

#### Introduction

wgrib2 is not simple upgrade of wgrib for grib2 files. wgrib can slice and dice grib1 files. wgrib2 is more like four drawers of kitchen utensils as well as the microwave and blender. This kitchen miracle was done by a more developer friendly design and the contributions of many people. Some functionality include,

- inventory and read grib2 files
- create subsets
- create regional subsets by cookie cutter or projections
- export to ieee, text, binary, CSV, netcdf and mysql
- write of new grib2 fields
- parallel processing by using threads (OpenMP)
- parallel processing by flow-based programming
- [fortran](#) and C interface

#### OpenMP Configuration

OpenMP is used to speed up wgrib2 by running loops over multiple cores. With gcc and icc, OpenMP is on by default. In general, you set the number of threads to be less or equal to the number of CPU cores available. Usually I set the number to 3 on a quad-core machine. That leaves one core for unrelated work. So I added "export OMP\_NUM\_THREADS=3" to my .bashrc. I also added "export OMP\_WAIT\_POLICY=PASSIVE" to my .bashrc in order to reduce CPU utilization at the expense of run time.

#### Compiling: NetCDF Configuration

Wgrib2 can be configured with (1) no NetCDF support, (2) NetCDF3 support or (3) NetCDF4 support. The advantages and disadvantages are,

- Option 1: No NetCDF support
  - fast compile
  - small executable (3.9 MB as of 2014)
  - compile works on 32+ bit machines
  - cannot read nor write NetCDF files
  - avoid library conflicts in wgrib2lib when calling program uses netcdf or hdf
  - makefile configuration: USE\_NETCDF3=0, USE\_NETCDF4=0
- Option 2: NetCDF3 support (default in makefile)
  - fast compile
  - modest increase in executable size (5.7 MB vs 3.9 MB as of 2014)
  - compile may fail on 32-bit machines
  - library conflict in wgrib2lib when calling program that use netcdf
  - makefile configuration: USE\_NETCDF3=1, USE\_NETCDF4=0
- Option 3: NetCDF4 support
  - slow compile time (hours on an Intel Apollo Lake)
  - 3.5x increase in executable size (13.5 MB vs 3.9 MB as of 2014)
  - compile has not failed on RedHat 6 (gcc 4.x and intel 17.03)
  - compile has failed on Cray X-30, worked on X-40.

- compile has failed on many newer Ubuntu systems and worked on others.
- requires fortran90 compiler
- library conflict in wgrib2lib when calling program that use netcdf
- makefile configuration: USE\_NETCDF3=0, USE\_NETCDF4=1
- prompts for downloading netcdf4/hdf5 libraries

1/2019: Debian and Fedora and many others have patches to make hdf5-1.10.4 work (needed for netCDF4). After you download hdf5-1.10.4, apply the patches. Then you may be able to compile wgrib2 with netCDF4 support.

## Execution: Segment Faults

Wgrib2 is used in NCEP operations and by many daily processing jobs at CPC. Seg-faults are pretty uncommon to wgrib2. The common problems are in related to experimental/alpha features and the following 3 problems.

The jpeg2000 library (Jasper) could seg-fault with larger grids. This problem was fixed with an Ubuntu/Debian update to Jasper (fall 2015) which was incorporated into wgrib2 v2.0.3. If your copy of wgrib2 is using another version of the Jasper library, then your copy of wgrib2 may be vulnerable to this problem.

Wgrib2 can decode grib using g2clib. Using the g2clib that was not included with the wgrib2 src code is could cause a seg-fault. Wgrib2 had patches to g2clib that were not fully incorporated until g2clib v1.4.0. In addition, g2clib can be compiled either with a 32 or 64-bit interface. The default makefile assumes that a 32-bit interface is used on 32-bit processors and a 64-bit interface is used 64-bit processors. Third-parties often followed NCEP's lead and used a 64-bit interface on 32-bit processors and a 32-bit interface on 64-bit processors.

Finally, the third problem is a compiler bug and only affects

1. wgrib2 v2.0.4
2. wgrib2 was compiled with OpenMP support
3. number of threads is greater than one
4. creating simple packed grib messages
5. all the grid points are undefined
6. compiled with older versions of gcc such as 2.4.7

This problem appears in gcc 2.4.7's handling of unsigned int looping variables when OpenMP is enabled. The problem is fixed by gcc 2.5.0.

There a potential for seg-faults by reading using specially crafted png or jpeg2000 compressed grib files. For example, a few days after the release of wgrib2 v2.0.3, the pnglib was upgraded to fix a seg-fault when reading a specially crafted png file.

## News

2/12/2019: [wgrib2 v2.0.8](#) is released, highlights: -ens\_processing, rot lat-lon grid support, -ndates

Tested systems with netcdf3 turned on

- gcc/gfortran: Ubuntu, RedHat
- clang (no fortran): Ubuntu
- icc/iftor: Redhat, Suse
- Windows 10: Cygwin64

## Some Good Stuff

### Usage

- each option call corresponds to a subroutine call
- type wgrib2 to see primary options
- inventory format is specified on command line by options
- if no "inv" option is specified, -s is used

## Joining the development effort

- Source code: knowledge of C and some grib-2

- Changes to existing source code should use the same licence as the original code
- New code (files) must either be GNU or public domain.
- Bug reports are important
- Contact [wesley.ebisuzaki@noaa.gov](mailto:wesley.ebisuzaki@noaa.gov)

## Contributions by

- Wesley Ebisuzai: many modules
- Reinoud Bokhorst: tosubmsg, checksum
- DWD (Gregor Schee, Daniel Lee and others): AEC compression
- Jaakko Hyvätti: gribtab
- John Howard: callable wgrib2
- Dusan Jovic: staggered grids, proj4 code
- Kristian Nilssen: netcdf module
- Karl Pfeiffer: georeferencing
- Pablo Romero: unix\_time
- Manfred Schwarb: many modules
- Arlindo da Silva: openGrADS, bbox
- Niklas Sondell: mysql module
- Sam Trahan: satellite tables
- Sergey Varlamov: netcdf module improvements, georeferencing updates
- thanks to the people who report the bugs and more who provide the fixes!

## Documentation

- [problems-error message](#)
- [Slides for talk to CUNY students 6/2016: grib1/grib2](#)
- [intro\\_grib2.pdf using GFS fcst](#)
- [Understanding the default inventory](#)
- [converting from wgrib to wgrib2 in scripts](#)
- [Powerpoint presentation](#) 4/6/2011 at NCEP
- [Changes in wgrib2 from 5/2012 to 1/2015 \(pdf\)](#)
- [submessages](#)
- [writing a simple function](#)
- [selecting fields to decode](#)
- [option types: init inv out misc](#)
- [bin, ieee, text format](#)
- [small fast databases](#)
- [usage questions](#)
- [using pipes](#) and [multiprocessing](#)
- [compile questions](#)
- [limitations](#)
- [make wgrib2 faster](#)
- [calling wgrib2 from C](#)
- [user defined grib tables](#)
- [special files](#)
- [some tricks](#)
- [some tricks for NCEP users](#)
- [one-line tricks](#)

## The options

- [the common options](#)
- [all the commands](#)

## Some solutions

- [Time interpolation of two grib files](#)
- interpolation to new grid: [new\\_grid](#), [Lat-Lon nearest neighbor](#), [Lat-Lon by Cressman](#)
- CSV (common separated value) file: [-csv](#), [-spread](#)
- [Windows version of wgrib2](#)
- [Fast averaging](#)

## Selecting Fields/Records/Message

**Select/Ignore by name/level/time/etc**

- [-match](#) process records that match a posix extended regular expression
- [-not](#) process records not matching a regular expression
- [-match\\_inv](#) the inventory used by -match and -not
- [-i](#) reads inventory from stdin for record selection

**Select by number (better to use previous method)**

- [-d](#) dump specific record
- [-for](#) select a range of records to process (nth message)
- [-for\\_n](#) select a range of records to process (nth message/submessage)

**Selective Processing: if options**

After selecting the fields to process, you can refine the processing by the "if" options. For example, you selected the winds and temperatures. You can use the "if" to send the winds to one file and the temperatures to another file. In general it is faster to use the -match/-not options because there is less processing of the data.

- [-if](#) if this then that
- [-not\\_if](#) if not this then that
- [-fi](#) closes -if (often an output option closes the if)

**Individual Grid Point Data**

- [-ij](#) print value at grid point (i, j)
- [-ijlat](#) print lat, lon, value at grid point (i, j)
- [-ilat](#) print lat, lon, value of Nth grid point
- [-lon](#) prints the lat-lon, and value of the grid point nearest the specified lat-lon
- [-max](#) prints the maximum value
- [-min](#) prints the minimum value
- [-stats](#) prints some statistics about the fields
- [-V](#) verbose inventory (shows stats)

**Regridding, Interpolating to new grids**

Wgrib2 has the ability to convert grib files from one grid to another. The conversion uses a user-selected interpolation scheme: bilinear (default), bicubic, nearest neighbor and budget. The supported grids (7/2010) include lat-lon, gaussian, Lambert conformal and polar stereographic. This capability uses the NCEP IPOLATES library and is an optional package.

Note: the interpolation uses scalar and vector interpolation schemes. For the vector quantities the V field must immediately follow the corresponding U field.

- [new\\_grid](#)
- [-new\\_grid\\_interpolation](#) set interpolation type used by -new\_grid
- [wgrib2m](#) fast regrid using multiple processes

**Exporting data to other programs**

- [-netcdf](#): write data in netcdf format
- [-mysql](#): export data to a mysql database
- [-mysql\\_speed](#): export data to a mysql database
- [-spread](#): write data for spreadsheets
- [-csv](#): write in column separated values, another one for spreadsheets
- [-text](#): data in text format
- [-bin](#): data native binary floating point
- [-ieee](#): data in big endian IEEE format
- [-ijbox](#): write a rectangular grid of data
- [-AAIG](#): arcinfo ascii grid, GIS

For a short list of options, type "wgrib -h"

For a complete list, type "wgrib -help all"

To search for an option, type "wgrib -help keyword"

**Writing grib2**

Writing grib2 is problem with no perfect solution. Some people want speed, some people want ease of use. Some people want it to run the code on a hundred processors and other want to run it on Windows XP. Here is a list of techniques that involve wgrib2. Here is the [background](#) for techniques that use the import function.

1. The basic [-import\\_ieee/bin/text](#) option.
2. using [-set\\_metadata](#)
3. [Example of converting HDF to grib2](#)

## Machines able to run wgrib2

- o linux: 64-bit with IPOLATES
  - Redhat 6 Enterprise: primary development system
  - Redhat 5 Enterprise: remove compile option "-Werror=format-security" from the makefile. This is a mandatory compile option for some distributions and is just a good idea. However, the gcc in RH5 doesn't support this option.
  - Ubuntu: 64-bit with IPOLATES
  - Ubuntu: clang and gfortran
- o Redhat linux: 32-bit with IPOLATES, not tested recently, use netcdf4 instead of netcdf3
- o Ubuntu: 32-bit with IPOLATES, not tested recently
- o Mandriva linux (old report)
- o AIX: use makefile, some fiddling with libraries is necessary, not tested recently
- o Solaris, needs gnu make and gcc (old report)
- o Solaris-10
- o HPUX, needs changes to makefiles (old report)
- o Windows: using Cygwin system produces 32-bit binaries. using netcdf4
- o Windows: using Cygwin system produces 64-bit binaries
- o Windows: compiled MingW (not recent), Watcom C, icc/ifort
- o Windows/linux subsystem (ubuntu): compiled with gcc/gfortran
- o Intel-based Mac
- o The IPOLATES library has been tested with on AIX fortran, g95, gfortran, ifort, and openf95

The makefile works on Redhat and Ubuntu (with needed installed options). For other systems, you may have to modify the makefile. The makefile requires gnu make is a common version of make.

## System dependencies: 32 vs 64 bit, Windows vs Linux/UNIX

The wgrib2 source code is written to be portable; there are no issues with big vs little endian or the size of the integer as long as it meets the minimum C requirement (32-bits). For windows, the cygwin's gcc supports the POSIX standard and allows the standard source code to be compiled. It is possible to compile the code with many of the POSIX extensions turned off was done when compiling with the Watcom compiler.

Wgrib2 can be compiled with 32-bit systems. With a 32-bit version, memory limitations will be important and the message size needs to be much smaller than 2GB. File size is limited to 2GB unless you restrict yourself to sequential reads and writes. (No -i or -submsg options, no using the byte location.)

Wgrib2 can be compiled for 64-bit systems. On most systems (types, not boxes), files can be larger than 2GB and the grib message can be as large as 2GB. (Many options support up to 4GB.)

For most Windows 64-bit C compilers, "long int" is defined as 32 bits. Consequently the ISO C random-access routine is limited to 2GB files. The 64-bit cygwin gcc works the way it should and is recommended for windows compilations.

The main difference between the windows and linux versions is that windows treats binary and text files differently. Consequently you can send a grib or binary data to STDOUT in unix/linux with no problem. Under windows, the results are implementation dependent. In addition, POSIX1 and POSIX2 functions may not be supported or may function differently. Consequently certain options may work differently.

- o The maximum number of open files is system dependent.
- o The maximum number of arguments to wgrib2 may be limited wgrib2 or by the system.

- The netcdf conversion is done in memory, so the speed is dependent on the free memory available.
- the accuracy of the trig functions depends on the compiler
- wgrib2 is usually compiled in fast mode rather than ieee mode.

## Source Code and Compling Hints

The wgrib2 source code is written to the POSIX standard. Features requiring POSIX2, such as regular expressions, can usually be turned off in the makefile. The wgrib2 code can be compiled with 32 or 64 bit pointers and integers. However, the code has to be compiled in a like manner for all the libraries. Some packages are optional (netcdf, mysql) and enabling these options can really increase the executable size.

While compiling wgrib2, you may see warnings about unknown pragmas. Pragmas are "comments" that are used when compiling an OpenMP version of the code. The default is to compile with OpenMP turned on. However, in some environments, turning off OpenMP is preferred.

2/2019: The HDF5 library has problems being compiled with newer gcc compilers. The only consisten method of adding NetCDF4 to wgrib2 is by using old compilers (RedHat 6) or using the Intel compilers. The work around to enable NetCDF3 support and use an external utility to convert the files to Netcdf4.

NetCDF3 has problem being compiled on 32-bit machines.

- FTP directory [by ftp by http](#)
- README [by ftp by http](#)
- Changes [by ftp by http](#)
- [compiling directions and questions](#)
- wgrib2.tgz [by ftp by http](#)
- Windows XP/Vista/7/8 32-bits [by ftp by http](#)
- Windows Vista/7/8/10 64-bits [by ftp by http](#)
- [NCAR's installation guide](#) using clang and OS-X
- [OS-X installation from author of rNOMADS](#) using gcc and OS-X

## Precompiled code from External Sites

There are many sites with precompiled versions of wgrib2. This list is neither exhaustive nor an endorsement of the sites. I have not tested the wgrib2 executables from these sites and YMMV.

- [OpenGrads](#): Darwin, Freebsd, Linux, Windows(cygwin)
- [Fedora Project](#)
- [RPMs](#): Centros, Fedora, OpenSUSE, RedHat, SUSE
- [MacPorts](#)

## Other programs that use wgrib2 (incomplete list)

- [g2ctl](#) a control maker for [GraADS](#)
- [atl\\_g2ctl](#), [alt\\_gmp](#) alternative versions of g2ctl/gribmap for for [GraADS](#)
- grib-filter/g2subset cgi-bin programs for Nomads (NOAA OperationalModel Archive & Distribution System)
- [g2grb.gs](#) enables GrADS to write grib2 files
- [rNOMADS](#) R interface for NOAA weather data
- [grb1to2.pl](#) grib1 to grib2 converter
- [wgrib2api](#), a fortran API for reading/writing grib2

## Status

The code is being developed on a Redhat linux x86\_64 platform in standard (POSIX) C. The wgrib2 source code requires an integer size of 32 or more bits. Incompatibilites tend to be related to problems in compiling of the external libraries (ex. netcdf, jasper).

[Change logfile](#)

## Bugs

9/2018: netcdf4 will not compile with Ubuntu, upgrade to hdf5 v1.10.4

11/2016: wgrib2 v2.0.5 AEC library doesn't compile, upgrade library source code

8/2015: wgrib2 v2.0.0-v2.0.2 -ieee does not work with OpenMP. Fixed 2.0.3

8/2015: complex packing: problem when creating if grid values are all undefined. Fixed 2.0.3

5/2015: callable wgrib2 and fatal errors. The error handling code can cause a seg fault on subsequent calls to wgrib2. Problem has been fixed v2.0.3

1/2014: Jasper library bug? Grib files that have large grids and that are compressed with jpeg-2000 can have problems. Have seen this problem in GrADS, wgrib2 and it has been reported in WCT. update: 11/2016: this problem was fixed by using Debian/Ubuntu version of the Jasper library included with wgrib2 2.0.3.

Please report them to [wesley.ebisuzaki@noaa.gov](mailto:wesley.ebisuzaki@noaa.gov). When you report bugs, try to make them reproducible on a linux machine and include sample data.

## Distribution

The source code modules for wgrib2 are either in the public domain or under the GNU licence depending on the authors of the various modules. Wgrib2 uses libraries that are in the public domain, under various GNU licences, the Image Power JPEG-2000 Public Licence\*, libpng licence\*, the zlib licence\*, the netcdf licence\*, HDF5 licence\*, MySQL licence\* and perhaps others. How about one licence to rule them all?

\* optional package

---

NOAA/ National Weather Service  
National Centers for Environmental Prediction  
Climate Prediction Center  
5830 University Research Court  
College Park, Maryland 20740  
Climate Prediction Center Web Team  
Page last modified: Dec 29, 2016

[Disclaimer](#)

[Privacy Policy](#)