wgrib2: changes since 5/2012

Operational wgrib2 on NCEP's WCOSS is 5/2012 version

Brief look at the changes since then

Wesley Ebisuzaki 1/2015

Changes: grib table

Grib table has been updated

Codes will fail if they use the generic name such as

--- short generic names ---
varM_N_O
M=discipline
N=parameter category
O=parameter number

--- long generic names ---
var discipline=0 master_table=2 parmcat=1 parm=190
var discipline=0 center=99 local_table=1 parmcat=1 parm=195

Changes: g2clib optional

wgrib2 used g2clib for decoding grib.

Fedora and Redhat versions of wgrib2 were failing because they were using the opn g2clib without my patches. (Policy)

g2clib fails when finds unknown template.

g2clib is not parallelized

	OLD	NEW
-g2clib=0	internal	internal
-g2clib=1 (default)	g2clib	internal with CF bug
-g2clib=2 (new)	N/A	g2clib (optional)

g2clib has CF bug, decimal scaling is ignored with constant field

g2clib is now optional for general distribution, may be included with NCEP distributions

Changes: Geolocation

Geolocation: determine the latitude and longitudes of the grid points. Need to go from spherical earth to ellipsoidal.

Internal code:	donated+some iplib-based, spherical
gctpc	USGS, orphaned? ellipsoidal
Proj4	Open source, gold standard ellipsoidal, uses config, OpenMP?

	OLD	NEW
default	internal (spherical)	gctpc *
-gctpc 1	gctpc	gctpc
-proj4 1 (new)	N/A	Proj4 (optional)

* transition in progress, reasons: ellipsoidal, no config, OpenMP
internal codes still used for Gaussian, rotated lat-lon

Changes: extended names

Names: want unique names to identify the fields

Old: HGT, TMP, PRES, UGRD

New:

Aerosols are called MASSMR (mass mixing ratio)

Need to know type and size

":TMP:2 m above ground:" is not TMP2m when error field

Names → extended names

MASSMR.aerosol=Dust_Dry.aerosol_size_>=2e-06,<3.6e-06.

TMP.analysis/forecast_error

-set_ext_name 1

Upgraded functionality: internal

-new_grid: e-grid, Mercator
(many): support for ensembles
(many): support for more tables
(many): bug fixes

Callable wgrib2 (v2.0.2)

Development project with outside user
C code can call wgrib2
Can transfer data by arguments
Only need to open file once
Can be used for read and writing grib
New

Upgraded functionality: external

grb1to2.pl	grib1 to grib2 converter, perl script that calls wgrib and wgrib2. Easy to build?
fast_grib2_mean.sh	many times faster than previous script
wgrib2m	script to parallelize wgrib2 by running multiple copies of wgrib2
g2grb.gs	GrADS script to write grib2 files Netcdf, HDF -> grib2
alt_g2ctl	alternative g2ctl/gribmap (GrADS)
rNOMADS	package for R for reading NCEP grib, by Daniel Bowman
grib-filter/g2subset	user can download regridded fields
fort_grib_wrt	fortran callable routines to write grib2

fort_grib_wrt:

Fortran callable routines to write grib. Based on having an existing grib2 file (template). Templates are made by regridding existing grib files (wgrib2) and modifying metadata (wgrib2).

Step 1: open a channel (file number)
Step 2a: create metadata string (based on wgrib2 inventory)
ex. ‘d=2015010212:HGT:500 mb:12 hour fcst:’
Step 2b: write grib message
Step 3: close channel

Fort_grib_wrt: Step 1

call grb_wopen(ichannel, file, template_file, grid_size,
n_templates, compress, options, ierr)

integer :: ichannel	integer 1..4, channel
character (len=*) :: file	name of the output file
character (len=*) :: template_file	name of grib2 file with one or more templates
integer :: grid_size(n_templates)	grid size for each template
integer :: n_templates	num. of templates
character (len=*) ;; compress	compression: c1, c2, c3, j
character (len=*) ;; options	wgrib2 options (see doc)
integer :: ierr	returns 0 if no errors

fort_grib_wrt: Step 2b

call grb_write(ichannel, template, data, metadata, ierr)

integer :: ichannel	integer
integer :: template	template number
real :: data(ndata)	real, values of the grid. default: WE:SN, WE:NS (see doc) undefined value = 9.999e20
character (len=*) :: metadata	metadata line
integer :: ierr	returns 0 if no errors

metadata is really simple .. wgrib2 inventory
'd=2015010100:HGT:500 mb:6 hour fcst:'
call grb_wrt(1,1,data,'d=2015012100:TMP:200 mb:12 hour fcst', ierr)

fort_grib_wrt: Step 3

Close channel

call grb_close(ichannel, ierr)

integer :: ichannel — integer, channel
Integer :: ierr — integer, 0 for no errors

fort_grib_wrt: Status (1/2015)

ss2gg (sigma spectral to Gaussian grid) was modified to write grib2 using these routines. Modifications were painless and ss2gg is now a light-weight post processor.

To handle more esoteric fields like aerosols would require updates to the -set_metadata option in wgrib2.

Reading grib from fortran (current)

```
call system("grep "match string" file.inv | wgrib2 -i file.grb
        -bin file.bin"
open(unit=10,file="file.bin" ....)
read(10) grid
close(10)
```

Not high performance .. but ok for climate, models like GFS

Reading grib from fortran (future)

Callable wgrib2

Should be possible to write a higher performance fortran reading API using callable wgrib2.

Operational issues

Interpolation: testing as NAM post-processor

Attractions: Can select type of interpolation

```
-net_grid_interpolation bilinear \
-if ":(SOTYP|VEGTYP):" -new_grid_interpoaltion neighbor -fi
```

Attraction: Can run faster (multiple processors by wgrib2m)

Complex packing with bitmap: NAM

Added (v2.0.2)

Complex packing with bitmaps is inferior because it produces larger files and is slower to read (wgrib2)

Compatible with some codes

Operational issues: updated wgrib2

Parts of NCO have said that versioning of utilities is a good idea and needs approval within NCO. They wanted of bundle of new utilities with the new hardware.

For now, need to include version of wgrib2 with package that you want to installed.