5 REGRID

5.1 Purpose

The purpose of the REGRID package is to transfer meteorological analyses from some native grid and format, to the horizontal MM5 grid in a format useful for the MM5 system programs. For anyone familiar with older versions of the MM5 system, the REGRID package replaces the obsolete DATAGRID program.

REGRID is the second step in the flow diagram of the MM5 modeling system (Fig 1.1). It accepts input from the TERRAIN program, and creates files ready for RAWINS, LITTLE_R, or INTERP. These files are used as the first guess to an objective analysis (RAWINS or LITTLE_R), or as analyses which are to be directly interpolated to the MM5 model levels (INTERP).

5.2 Structure

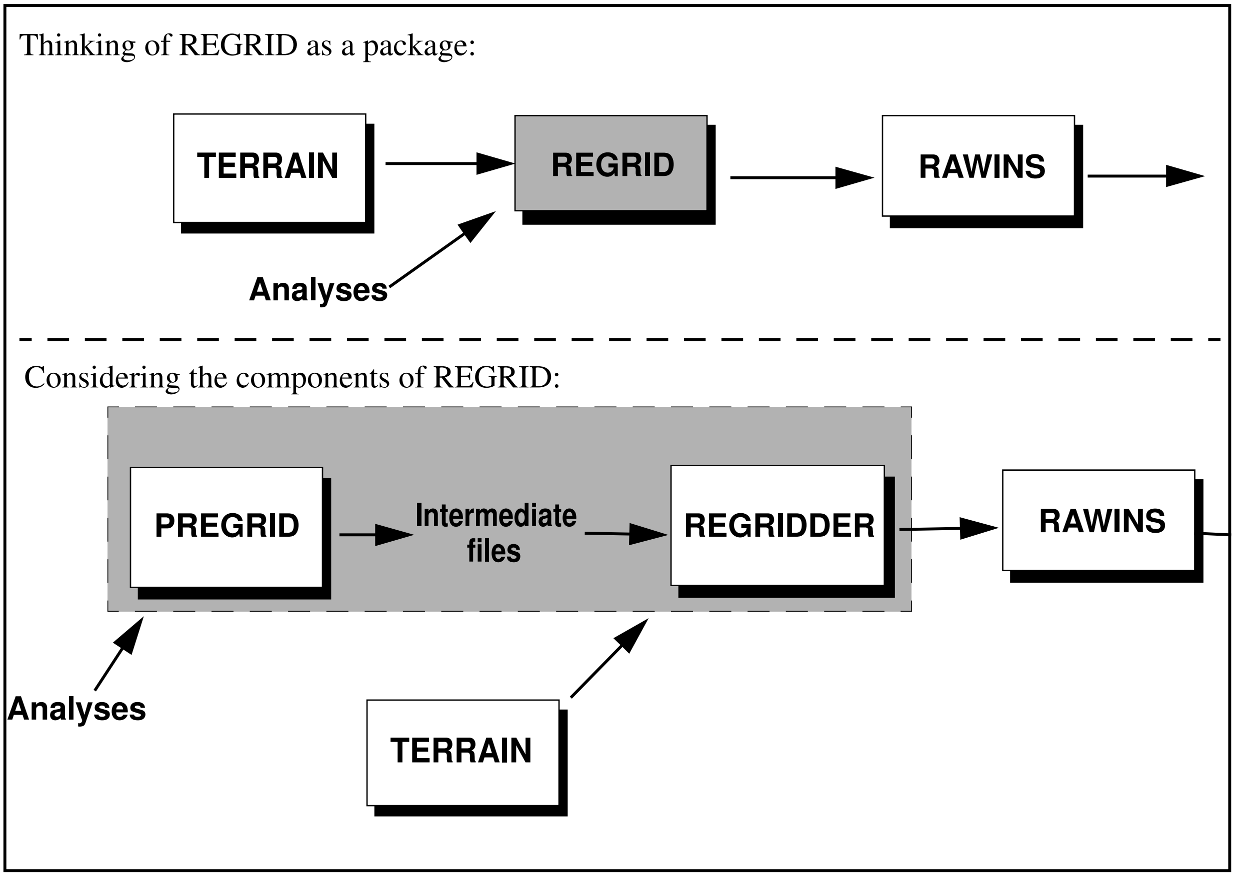
The REGRID package is not a single program (the way we like to think about most of the other programs of the MM5 modeling system), but a set of programs which divides the tasks into two broad categories:

1) Reading the original meteorological analyses.
2) Interpolating the analyses to the MM5 grid.

The main division of REGRID is into two sub-packages, called “pregrid” and “regridder”. The pregrid portion handles the data ingest (stage 1), and the regridder portion handles interpolation and output (stage 2). Data are sent from pregrid to regridder by files written in a relatively simple specified format. The pregrid tasks are further subdivided into programs which read specific data formats, while the regridder tasks are managed in a single program.

This division separates the fairly messy and very data-specific task of reading the data from the more general task of interpolation. By this division, REGRID can be more easily expanded to ingest more data sets, and users can more easily ingest their own data sets for use in the MM5 system.

5.3 A schematic



5.4 Input to pregrid

The pregrid program expects to find files of meteorological analyses. Currently, pregrid can read many GRIB-formatted data sets and several GRIB and non-GRIB data sets that have traditionally been available to MM5 users.

Most of the individual pregrid programs also expect to find tables which tell the pregrid program what fields to access from the input files. These are referred to as “Vtables” and are discussed in greater detail below.

A Fortran namelist file passes user-specified options to pregrid. For pregrid, this is mostly date information.

5.5 Input to regridder

The regridder program expects to find in the files from pregrid the fields of temperature, horizontal wind components, relative humidity, height of pressure levels, sea-level pressure, sea-surface temperature, and snow-cover data. Other fields may be used as well, interpolated and passed on to the rest of the modeling system. When you set up and run the pregrid programs, you should verify that the files you pass to regridder contain the necessary fields.

From the TERRAIN files, regridder finds terrain, land-use, and map data.

A Fortran namelist file passes user-specified options to regridder.

5.6 Output from regridder

The regridder program creates a file called "datagrid". This file contains the data at every time period for a single domain. For now, users should run the program pablum, which converts the datagrid file to a file called "datagrid_old_format".

5.7 Intermediate Data Format

Key to the REGRID package is the data format used for passing data from pregrid to regridder. Data are passed from the pregrid programs to regridder via files written in the format described in this section.

5.7.1 General format description

Data are written as horizontal slabs defined at a given pressure level (or the surface). Each horizontal slab contains a single level of a single variable (e.g., 500 mb RH, surface T, etc.). Any number of horizontal slabs may be written in to single file. The slabs in a given file are not necessarily all from the same data source, or all on the same map projection, but they all represent the same time.

5.7.2 File Naming conventions

Each file contains data for a single time. The file names consist of a prefix (possibly denoting the source of data), followed by a colon, followed by a time-stamp in the form YYYY-MM-DD_HH. Regridder uses the file names as discussed below.
For example, analyses from the ON84-formatted data from NCEP for 16 June 1999 at 12 UTC may be written to a file called "ON84:1999-06-16_12"

5.7.3 File format

The files are written as unformatted FORTRAN 32-bit IEEE records. Four records are used for every horizontal slab:

-Record 1: IFV
-Record 2: HDATE, XFCST, FIELD, UNITS, DESC, XLVL, NX, NY, IPROJ

Record 3: Depends on value of IPROJ:

if (IPROJ == 0) (Cylindrical equidistant grid)
-Record 3: STARTLAT, STARTLON, DELTALAT, DELTALON
if (IPROJ == 3) (Lambert conformal grid)
-Record 3: STARTLAT, STARTLON, DX, DY, XLONC, TRUELAT1, TRUELAT2

if (IPROJ == 5) (Polar-stereographic grid)
    -Record 3: STARTLAT, STARTLON, DX, DY, XLONC, TRUELAT1

    -Record 4: SLAB

where:

integer	:: IFV	:The PREGRID format version number, currently 3
char*24	:: HDATE	:The time, in format “YYYY-MM-DD_HH:mm:ss ”
real	:: XFCST	:Forecast time (in hours) of the data in the slab
char*9	:: FIELD	:Field name, those with special meaning are described below
char*25	:: UNITS	:Units describing the field in the slab
char*46	:: DESC	:Text description of the field
real	:: XLVL	:Pressure-level (Pa) of the data. 200100 Pa indicates surface data; 201300 Pa indicates sea-level data
integer	:: NX	:Slab dimension in the X direction
integer	:: NY	:Slab dimension in the Y direction
integer	:: IPROJ	:Flag denoting the projection. Recognized values are: 0: Cylindrical Equidistant (regular lat/lon) grid 3: Lambert conformal projection 5: Polar stereographic projection
real	:: STARTLAT	:Starting latitude (degrees)
real	:: STARTLON	:Starting longitude (degrees)
real	:: DELTALAT	:Latitude increment (degrees) for lat/lon grid
real	:: DELTALON	:Longitude increment (degrees) for lat/lon grid
real	:: DX	:Grid-spacing in x (km at TRUELAT1 (and TRUELAT2))
real	:: DY	:Grid-spacing in y (km at TRUELAT1 (and TRUELAT2))
real	:: XLONC	:Center longitude of the projection
real	:: TRUELAT1	:Standard latitude used for polar stereo. and Lambert conf. projections
real	:: TRUELAT2	:Second standard latitude value used for Lambert conf. projection
real	:: SLAB	:Two-dimensional array (NX,NY) of data

5.7.4 Special field names

The variable FIELD indicates the physical variable in the slab. Certain values of FIELD are recognized by regridder for specific treatment. Slabs identified by an unrecognized values of FIELD are simply interpolated horizontally and written out by regridder. Recognized field names are:

T	-	Expected for input to RAWINS
U	-	Expected for input to RAWINS
V	-	Expected for input to RAWINS
RH	-	Expected for input to RAWINS
HGT	-	Expected for input to RAWINS
PMSL	-	Expected for input to RAWINS
SST	-	Expected for input to RAWINS
SNOWCOVR		
SOILT010		
SOILT200		
SOILT400		

SOILM010
SOILM200
SEAICE
LANDSEA
SOILHGT
WEASD

5.8 Pregrid VTables

Pregrid is intended to read a wide variety of data sets. Since many data sets are archived and distributed in GRIB format, and the non-GRIB data sets we read use many of the same ideas for describing given fields, it is convenient to use the GRIB data sets as an example.

The GRIB format describes each field by several code numbers. However, we cannot include the code tables in the program code itself, because these code numbers are not consistent from one data set to another. Also, pregrid must have the capability to ingest analyses that we have not anticipated. Therefore, we have to supply coded information to the program through some sort of table. The pregrid VTables are the means to do this.

The body of the VTables consists of one or more lines describing the fields we want to extract from the analysis files. A couple of examples are in order:

```
GRIB  | Level | Level | Level| REGRID     | REGRID    | REGRID                 |
Code  | Code  |   1   |   2  | Name       | Units     | Description            |
------+-------+-------+------+-----------+-----------+-----------------------+
11    | 100   |   *   |      | T          | K         | Temperature            |
11    | 105   |   2   |      | T          | K         | Temperature at 2 m     |
------+-------+-------+------+-----------+-----------+-----------------------+
```

The GRIB Code is the code number identifying the variable to access. For example, in NCEP GRIB files, temperature is generally coded as 11.

The Level Code is the code number identifying the type of level on which the variable is expected. For example, GRIB Level Code 100 refers to pressure levels, and GRIB Level Code 105 refers to a fixed height (in meters) above the ground.

Level 1 is the GRIB code for the value of the level. An asterisk (*) means to get data from every level of the type defined by the Level Code.

Level 2 is often needed for types of levels (such as averages or depths) which are defined by two values.

REGRID Name is the character string identifying the field to the rest of the modeling system.

REGRID Units are the units used for this field in the rest of the modeling system

REGRID Description is a text description of the field.

There are a few subtleties to the VTables. A situation that sometimes occurs is that we want a field that is not included in the source files, but may be derived from fields which are in those files. One example is relative humidity. Some data sets may archive specific humidity instead. Yet we can derive RH from specific humidity, temperature, and pressure. We want to write out RH, but not write out specific humidity. So we need to ask for specific humidity in the Vtables.

The signal in the VTables that a certain field is not to be written out is a blank REGRID Description. Since we want to write out relative humidity, we include the relative humidity in the VTables in the usual way (with no GRIB Code since it wouldn't be found anyway). There is coded into the program the conversion from specific humidity to relative humidity, so pregrid will create the relative humidity.

GRIB Code	Level Code	Level 1	Level 2	REGRID Name	REGRID Units	REGRID Description
11	100	*		T	K	Temperature
51	100	*		SPECHUMD	kg kg{-1}	
	100	*		RH	%	Relative Humidity

There are several VTables already set up for certain data sets that we have commonly accessed. Most of these are found in the directory pregrid/grib.misc. If you want to access a different GRIB-formatted data set, you must first determine which variables are included in that data set, and find the appropriate code numbers that are used by that data set. If you want to extract additional variables from a data set, you are responsible for finding the appropriate GRIB Code and Level Code numbers. You may find NCEP Office Note 388, a description of the GRIB format, useful. This document can be found in many places on the internet, including

ftp://nic.fb4.noaa.gov/pub/nws/nmc/docs/gribed1

5.9 Pregrid program functioning

The pregrid programs first read the namelist to determine the starting and ending times of the period of interest, and to find the desired time-interval of data. It then reads the VTable to determine which variables to extract from the source files. Then for each source file, the program scans through the data, pulling out all analyses which fall between the starting and ending times, and which are listed in the VTable. These analyses are written to files (named by time). Once a record with a time greater than the user-specified ending time has been read, processing on that file stops and the next file is opened. (I.e., records in the source file are assumed to be sorted by time.) This cycle repeats until all the source files have been scanned.

Once that cycle is finished, the files are reread and derived fields are computed. Temporal interpolation is performed as necessary to fill in missing time periods. The final files are written.

5.10 Handy pregrid utility programs

We have created a handful of handy programs you should be aware of.

gribprint [- v | -V] file

Scans through a GRIB-formatted file, printing out a few details of each GRIB record.
With the -v option, prints more details of the GRIB header.
With the -V option, prints way too much of the actual data values.
This program is made automatically when you issue the top-level make. It is found in the pregrid/util directory.

plotfmt file

Makes plots of each field in the file which has been created by pregrid. This program requires NCAR Graphics. To make this program, go to the util directory, compile plotfmt.F using NCAR-Graphics and loading library libpgu.a.

get_ncep.deck

Job deck to be submitted to the NCAR Crays, to download archives from the GRIB-formatted NCEP GDAS analyses. Found in directory pregrid/ncep.grib.
Be sure to check the catalog for missing analyses.

get_on84.deck

Job deck to be submitted to the NCAR Crays, to download archives from the ON84-formatted NCEP GDAS analyses. Found in directory pregrid/on84. Be sure to check the catalog for missing analyses.

get_nnrp.deck

Job deck to be submitted to the NCAR Crays, to download archives from the NCEP/NCAR Reanalysis project. Found in directory pregrid/nnrp.

get_era.deck

Job deck to be submitted to the NCAR Crays, to download archives from the ECMWF Reanalysis project. Found in directory pregrid/era.

5.11 How to run REGRID

1) Get the source code. The current REGRID release resides on NCAR's anonymous FTP site, ftp://mesouser/newprogs/regrid.tar.gz. There may be a regrid.tar available elsewhere for the tutorial. Uncompress ("gunzip regrid.tar.gz") and untar ("tar -xvf regrid.tar") the file. This creates a top-level REGRID directory called, strangely enough, REGRID.

2) Make the executables. To do this, go into the REGRID directory, and type "make". That doesn't work, but it does give you a list of architectures on which we've run REGRID. We've build in appropriate options for each of these architectures. Issue the make command for the particular architecture you are running on (e.g., "make sgi" or "make dec"). Watch while your machine builds all the executables.

3) Get the analysis files. It may be convenient to put these files in a directory of their own. For users of NCAR's machines interested in historical cases, investigate the get_*** programs mentioned in "Handy utility programs".

4) Set up to run pregrid. The "pregrid.csh" shell in the pregrid subdirectory is handy. This is discussed below. If you need to create your own Vtables, do it now.

5) Run pregrid.csh: "pregrid.csh"

6) Check your output: Make sure that pregrid created files for every time between your starting and ending dates. Check the printout to see which fields are available at which times.

7) Set up to run regridder: Get your terrain output file. Go to the regridder subdirectory, and edit the namelist for your specific case.

8) Run regridder. "regridder". This creates a file "datagrid". If necessary, and for now it probably is necessary, run pablum: "pablum". This reads file "datagrid" and creates "datagrid_old_format".

5.12 pregrid.csh

A shell script has been created, called pregrid.csh, as a higher-level user interface for the pregrid programs. The top part of the pregrid.csh script looks something like this (variables the user may have to set are noted with a vertical bar to the left.

```
###########################################################################
#!/bin/csh -f
set echo
#
# Put your input files for pregrid into the directory you specify as DataDir:
#
   set DataDir = /tmp1/username/data
#
# Select the source of 3-d analyses:
#
#   set SRC3D = ON84   # Old ON84-formatted NCEP GDAS analyses
#   set SRC3D = NCEP   # New GRIB-formatted NCEP GDAS analyses
    set SRC3D = GRIB   # Many GRIB-format data sets
#
# Tell the program where you have put the analysis files, and what you have
# called them:
#
   set InFiles = ( ${DataDir}/GRIB.1994DEC \
                   ${DataDir}/GRIB.1995JAN )
#
# Select the source of SST analyses
#
#  set SRCSST = ON84
#  set SRCSST = NCEP
#  set SRCSST = NAVY
   set SRCSST = $SRC3D
#
# Tell the program where the files with SST analyses are.  Do this only if SST
# analyses are coming from files not named above in InFiles.
#
  set InSST = ( )
#
# Select the source of snow-cover analyses (entirely optional)
#
#  set SRCSNOW = $SRC3D
#  set SRCSNOW = ON84
#  set SRCSNOW = GRIB

# Set this one only if snow-cover analyses are from files not listed in InData

#   set InSnow = ( ${DataDir}/ON84.1994DEC ${DataDir}/ON84.1995JAN )
#
# Select the source of soil model analyses (entirely optional)
#
#   set SRCSOIL = $SRC3D

# Set this one only if snow-cover analyses are from files not listed in InData

   set InSoil = ()
#
#  Build the Namelist
#
if ( -e ./pregrid.namelist ) then
   rm ./pregrid.namelist
endif
cat << End_Of_Namelist | sed -e 's/#.*//; s/  *$//' > ./pregrid.namelist
&record1
#
# Set the starting date of the time period you want to process:
#
```

```
 START_YEAR  = 1994   # Year (Four digits)
 START_MONTH = 12     # Month ( 01 - 12 )
 START_DAY   = 30     # Day ( 01 - 31 )
 START_HOUR  = 00     # Hour ( 00 - 23 )

 END_YEAR  = 1995   # Year (Four digits)
 END_MONTH = 01     # Month ( 01 - 12 )
 END_DAY   = 03     # Day ( 01 - 31 )
 END_HOUR  = 00     # Hour ( 00 - 23 )
#
# Define the number of time periods, and the time interval to process.
#
 INTERVAL =  43200 # Time interval (seconds) to process.
                   # This is most sanely the same as the time interval for
                   # which the analyses were archived, but you can really
                   # set this to just about anything, and pregrid will
                   # interpolate in time and/or skip over time periods for
                   # your regridding pleasure.

/
End_Of_Namelist

if ( ! $?SRC3D ) then        #\
   set SRC3D                 # \
endif                        #  \
if ( ! $?SRCSNOW ) then      #   \
   set SRCSNOW               #    \
endif                        #     \ Do not change these lines.
if ( ! $?SRCSST ) then       #     /        Please.
   set SRCSST                #    /
endif                        #   /
if ( ! $?SRCSOIL ) then      #  /
   set SRCSOIL               # /
endif                        #/

#
#  Tell the Pregrid programs which Vtables to use.  Note that this is done
#  only if you have selected GRIB-formatted input.
#

if ( $SRC3D == GRIB ) then
   set VT3D = ../grib.misc/Vtable.xxxx3D
endif
if ( $SRCSST == GRIB ) then
   set VTSST = ../grib.misc/Vtable.xxxxSST
endif
if ( $SRCSNOW == GRIB ) then
   set VTSNOW = ../grib.misc/Vtable.xxxxSNOW
endif
if ( $?SRCSOIL ) then
   if ( $SRCSOIL == GRIB ) then
      set VTSOIL = ../grib.misc/Vtable.xxxxSOIL
   endif
endif

########################################################################
########################################################################
######                                                            ######
######                  END USER MODIFICATION                     ######
######                                                            ######
```

```
###########################################################################
###########################################################################
```

The rest of the shell performs some file shuffling to put files in places that the pregrid programs expect. The shell links the source files to files of specific names which the pregrid programs expect. The shell builds a file called "VTable" from the individual Vtables named by the user in the shell. The shell then executes the program, and moves the final output files to the pregrid directory.

5.13 The regridder Namelist options

The regridder program is run entirely through the namelist file. There are no values required at compile-time to specify the horizontal or vertical extent of the input (the intermediate format previously described) or output (MM5 system DATAGRID format) data. The regridder namelist is separated into four namelist records, with similar data within each record.

5.13.1 RECORD1

The first namelist record handles the temporal information required by the regridder program: basically, when do I start, when do I stop, and how many intermediate steps are to be taken between those bounding times. This namelist record is identical to that of pregrid. (see pregrid.csh, above).

5.13.2 RECORD2

The second record for regridder deals with information concerning the vertical levels. The user defines the top of the analysis and which "new" levels to add to the first-guess data (through vertical interpolation from the surrounding layers, linear in pressure).

```
&record2
 ptop_in_Pa                  = 10000 , /
 new_levels_in_Pa            = 97500 , 95000 ,         90000 ,
                               87500 ,         82500 , 80000 ,
                               77500 , 75000 , 72500 ,
                               67500 , 65000 , 62500 ,
                               57500 , 55000 , 52500 ,
                               47500 , 45000 , 42500 ,
                               37500 , 35000 , 32500 ,
                               27500 ,         22500 ,
                               17500 ,         12500          /
```

5.13.3 RECORD3

The third record is used to input the pregrid output names to the regridder program. The file names include the root of the file name (up to but not including the ":", and may include directory information). The character string after the ":" is the date, which is internally generated by the regridder program based on the information provided in RECORD1.

For example, to input the file "../test/FILE:1996-07-30_00:00:00", data would be entered as given below. Multiple files for the same time may be used as input. It is typical for the sea-surface temperature to be defined in a file different than the wind fields, for example. The user appends as many files names are required on the root_nml line (a limit of 20 is currently enforced).

The terrain_file_name is the file name of the output file from the terrain program.

```
&record3
 root                              = `../test/FILE' ,
 terrain_file_name                 = './terrain' /
```

5.13.4 RECORD4

The fourth record controls the print-out in the regridder program. Until something is going wrong, keep everything turned off.

```
&record4
 print_echo                        = .FALSE. ,
 print_debug                       = .FALSE. ,
 print_mask                        = .FALSE. ,
 print_interp                      = .FALSE. ,
 print_link_list_store             = .FALSE. ,
 print_array_store                 = .FALSE. ,
 print_header                      = .FALSE. ,
 print_output                      = .FALSE. ,
 print_file                        = .FALSE. /
```

5.14 REGRID tar File

The regrid.tar file contains the following files and directories:

REGRID/configure.rules	Rules for make
REGRID/pregrid	pregrid directory
REGRID/regridder	regridder directory
REGRID/Makefile	Makefile to create REGRID executables.