# Visualizing JMA High-Resolution GSM Data with GrADS

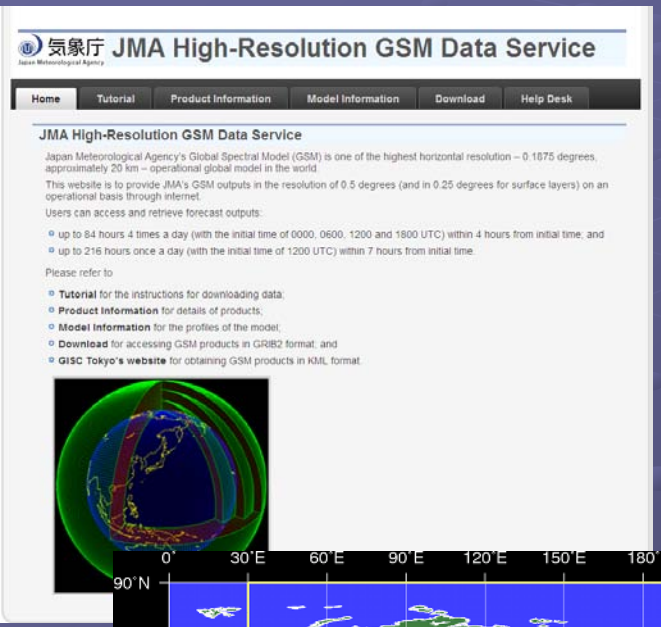
Numerical Prediction Division
Information Infrastructure Department, JMA

Last updated on 20 April 2021

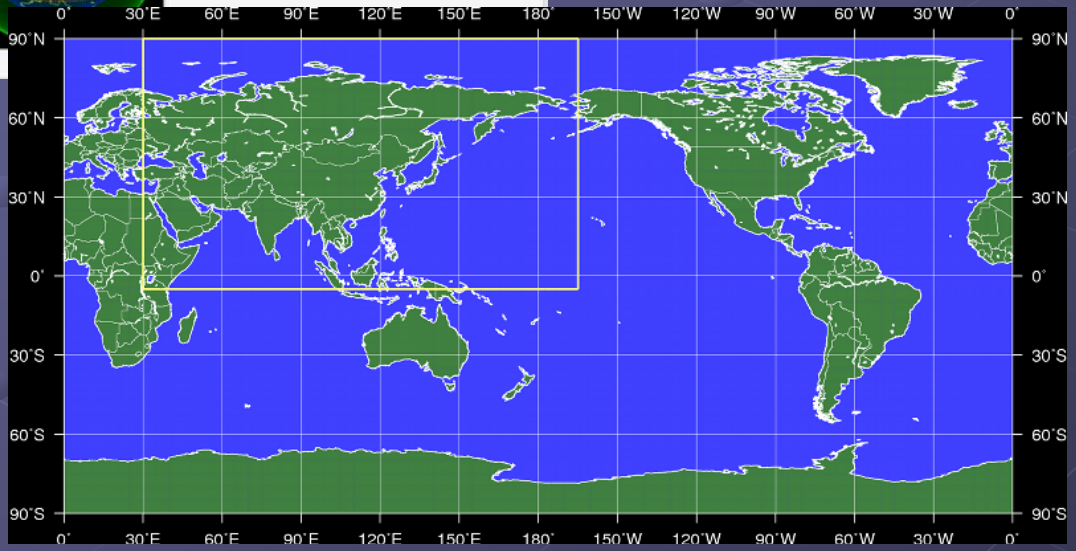
# Contents

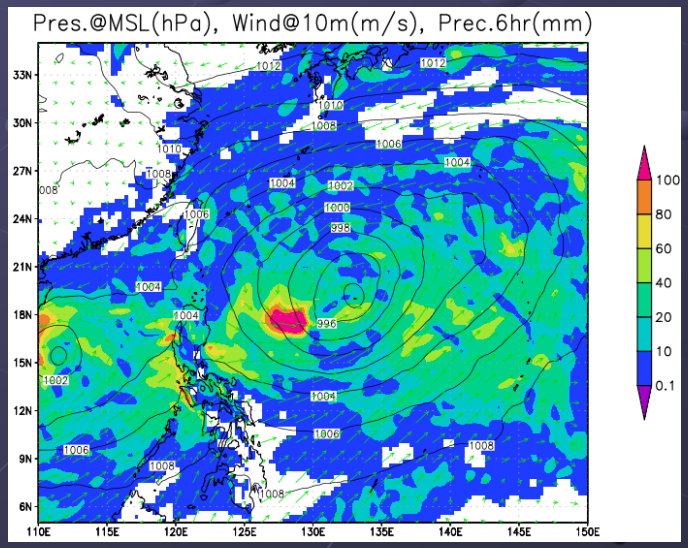
- Getting Meteorological Data
  - JMA High Resolution GSM Data Service
- Preparing for Visualization
- Visualizing Data with GrADS
  - Basic interactive operation
  - Customizing images
  - Converting images
- Scripting with GrADS

# JMA High-Resolution GSM Data Service




- Target: NMHSs
  - Mainly in WMO Region II
- Model: GSM(TL959)
- Data Format: GRIB2
- Preservation Term: Latest 14 days



# Profile (Global)

| | Upper Air | Surface | |
|---|---|---|---|
| Resolution | 0.5deg (720 x 361) | | 0.25deg (1440 x 721) |
| Region | Lon: 0E – 180 – 0W, Lat: 90S – 90N | | |
| Initial Time | 00, 06, 12, 18 UTC (4 times per day) | | |
| Forecast Time | 00 to 132 hours by 3 hours (any initials)<br>132 to 264 hours by 6 hours (00, 12UTC initials) | | |
| Levels | 21 (1000, 975, 950, 925, 900, 850, 800, 700, 600, 500, 400, 300, 250, 200, 150, 100, 70, 50, 30, 20, 10 hPa)<br>* All levels in one file | 1 (surface) | |
| Elements | U, V, T, Rh, Z, Omg<br>Stream Function, Velocity Potential (at 850, 250 hPa)<br>Vorticity (at 500 hPa) | U, V, T, Rh, Ps, Psea, Rain, Cloudiness (total, high, middle, low) | |
| File size | 50MB | 4MB | 17MB |

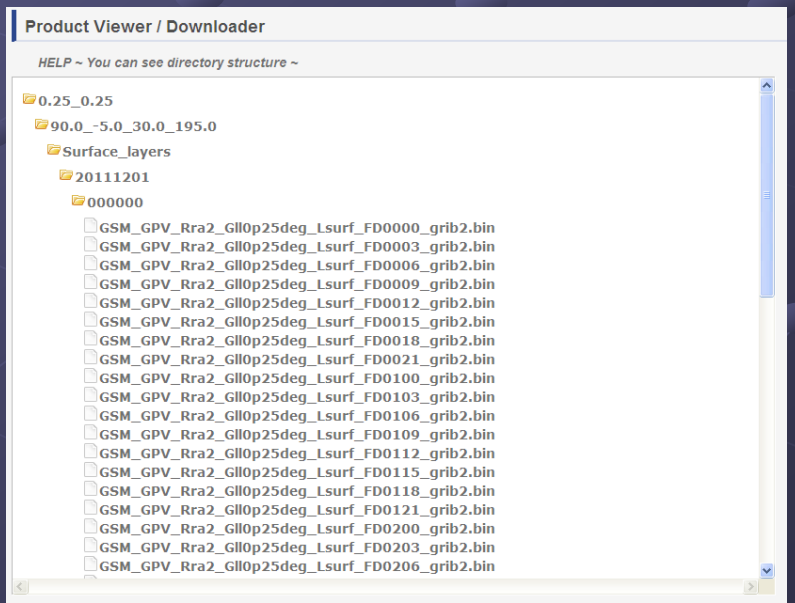
# Profile (WMO Region II)

| | Upper Air | Surface | |
|---|---|---|---|
| Resolution | 0.5deg (331 x 191) | | 0.25deg (661 x 381) |
| Region | Lon: 30E – 180 – 165W, Lat: 5S – 90N | | |
| Initial Time | 00, 06, 12, 18 UTC (4 times per day) | | |
| Forecast Time | 00 to 132 hours by 3 hours (any initials)<br>132 to 264 hours by 6 hours (00, 12UTC initials) | | |
| Levels | 21 (1000, 975, 950, 925, 900, 850, 800, 700, 600, 500, 400, 300, 250, 200, 150, 100, 70, 50, 30, 20, 10 hPa)<br>* Separated 21 files by levels | 1 (surface) | |
| Elements | U, V, T, Rh, Z, Omg<br>Stream Function, Velocity Potential (at 850, 250 hPa)<br>Vorticity (at 500 hPa) | U, V, T, Rh, Ps, Psea, Rain, Cloudiness (total, high, middle, low) | |
| File size | 500 – 700KB | 1MB | 4MB |

# Directories of Server（1）

0.25_0.25/ 0.25deg. resolution

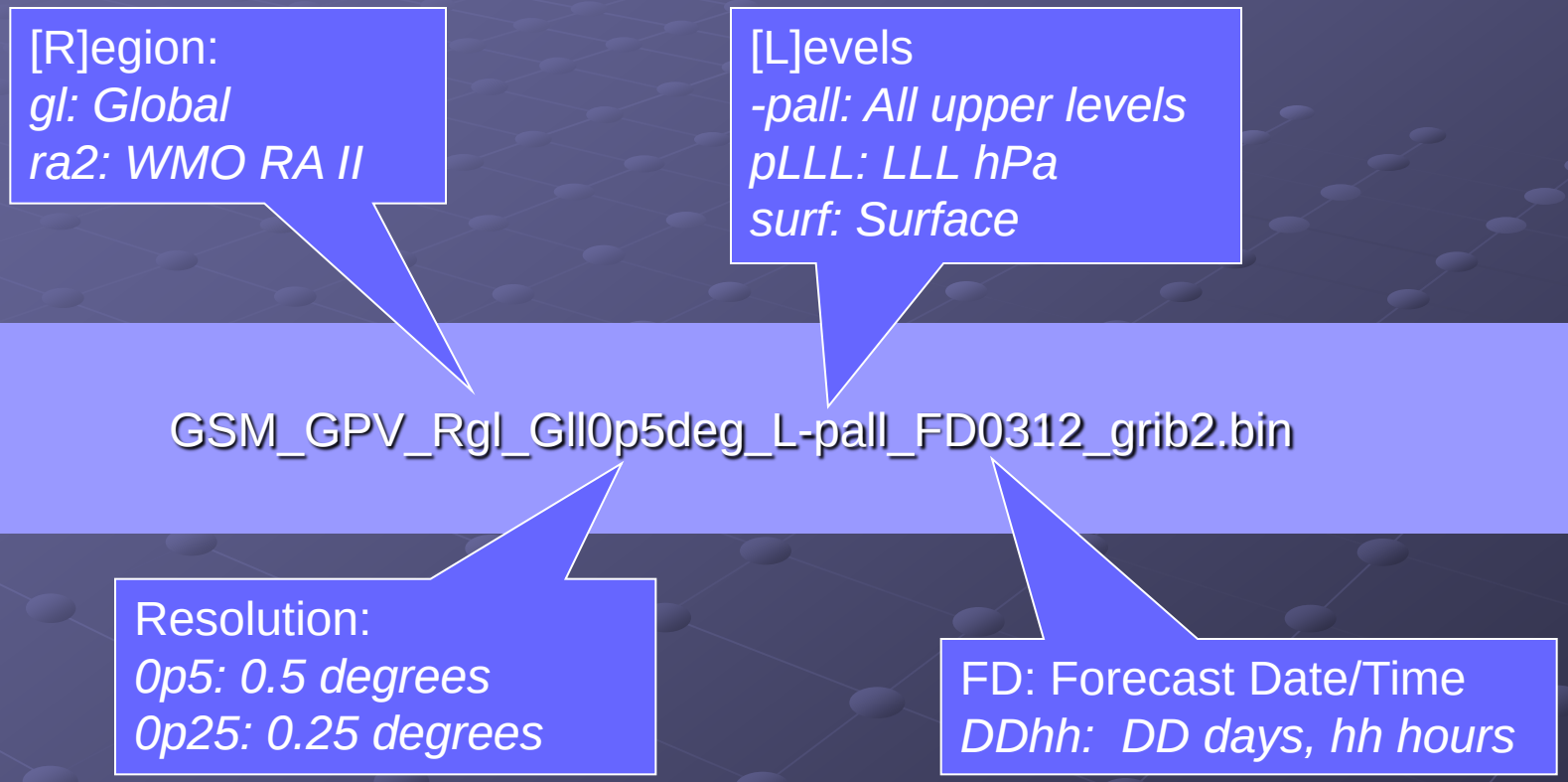
- 90.0_-5.0_30.0_195.0/ WMO Region II
  - Surface_layers / Surface
    - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
      - GSM_GPV_Rra2_Gll0p25deg_Lsurf_FD0000_grib2.bin
      - …
- 90.0_-90.0_0.0_359.75/ Global region
  - Surface_layers / Surface
    - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
      - GSM_GPV_Rgl_Gll0p25deg_Lsurf_FD0000_grib2.bin
      - …



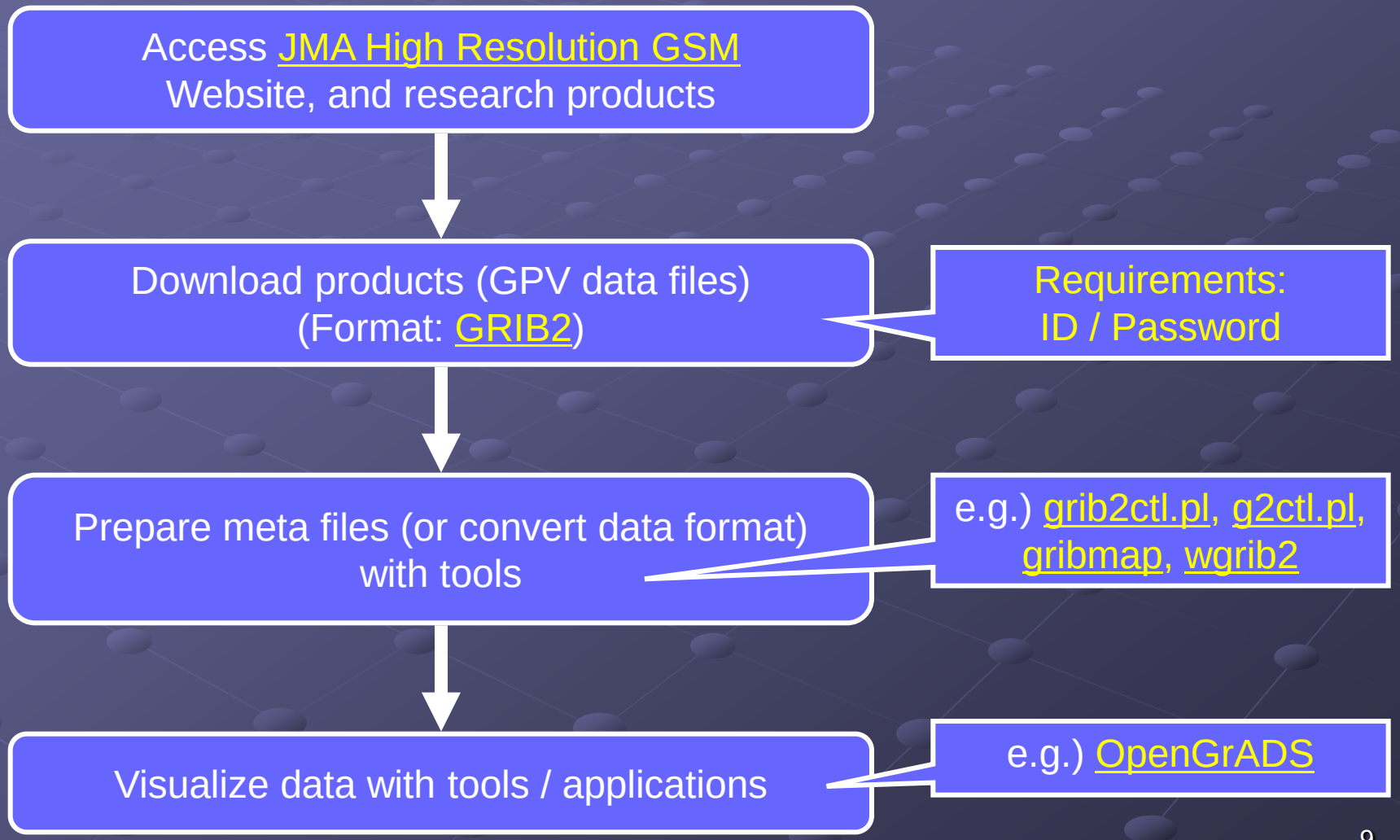
# Directories of Server（2）

- 0.5_0.5/ 0.5deg. resolution
  - 90.0_-5.0_30.0_195.0/ WMO Region II
    - Surface_layers / Surface
      - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
        - GSM_GPV_Rra2_Gll0p5deg_Lsurf_FD0000_grib2.bin
        - …
    - 1000hPa/ Level 1000hPa only
      - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
        - GSM_GPV_Rra2_Gll0p5deg_Lp1000_FD0000_grib2.bin
        - …
    - …
  - 90.0_-90.0_0.0_359.5/ Global region
    - Surface_layers / Surface
      - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
        - GSM_GPV_Rgl_Gll0p5deg_Lsurf_FD0000_grib2.bin
        - …
    - Upper_air_layers / 1000hPa – 10hPa Levels in one file
      - 20111201/000000/ Initial Date/Time (format:YYYYMMDDhhmmss)
        - GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0000_grib2.bin
        - …

# Naming rule of data files



# Workflow for processing meteorological data



# What's GRIB?

- GRIB (GRIdded Binary)
  - CBS(Commission for Basic Systems)/WMO Data format
    - Described in WMO Manual on Codes No.306
  - For the storage and the exchange of Weather Products in Gridded Binary Form
  - Editions:
    - GRIB(GRIB1): Used world-wide for NWP
    - GRIB2: Latest edition (much different in format)

For details: http://www.wmo.int/pages/prog/www/WMOCodes.html

# Tools for GRIB/GRIB2

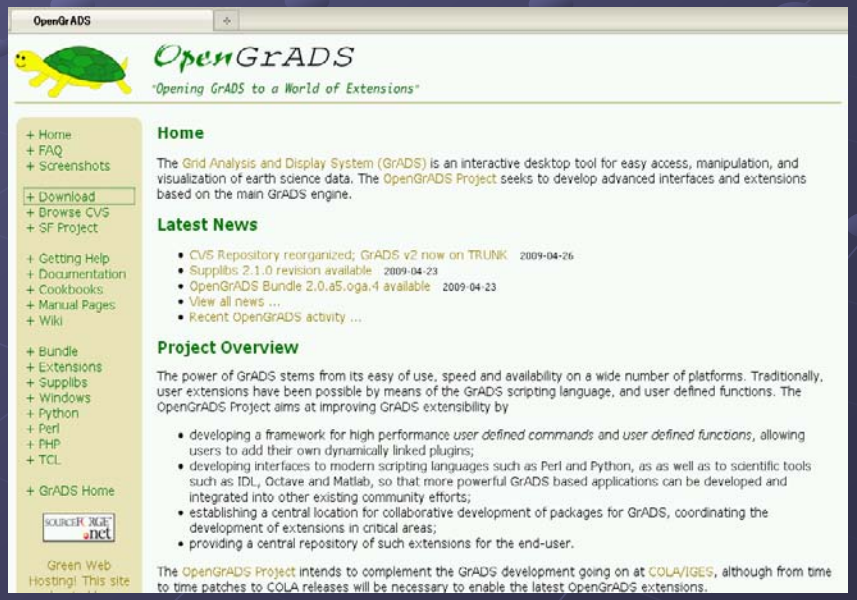
- Decoding tools
  - wgrib : decoding / converting GRIB data file
  - wgrib2 : decoding / converting GRIB2 data file
- Preprocessing tools for GrADS
  - grib2ctl.pl : making GrADS control file from GRIB data file
  - g2ctl.pl : making GrADS control file from GRIB2 data file
  - gribmap : making GrADS index file from GRIB or GRIB2 data file

Note: ‘wgrib’, ‘wgrib2’, and ‘gribmap’ are installed with GrADS (‘Windows superpack’ edition).

‘grib2ctl.pl’ and ‘g2ctl.pl’ are Perl scripts. They require Perl.

# What's GrADS?

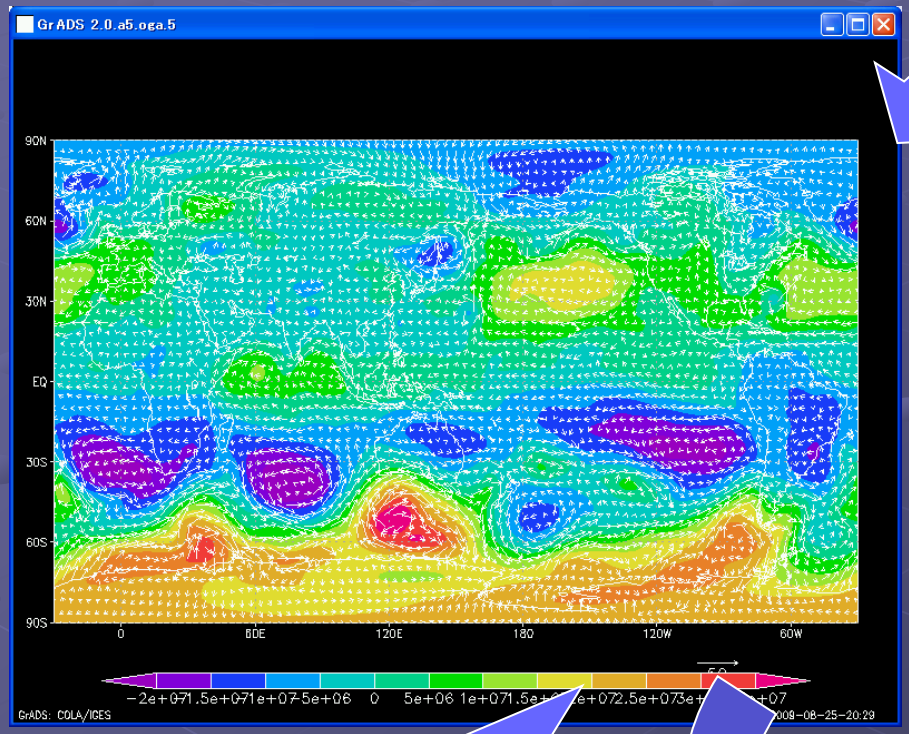
- GrADS(The Grid Analysis and Display System)
  - Interactive desktop tool
  - Commonly used for easy access, manipulation, and visualization of earth science data
  - CUI(Character User Interface) / GUI(Graphical User Interface) Application
- OpenGrADS
  - Based on 'GrADS'
  - Open Source
    - Available sources/binaries of several OSs
  - Some useful Extensions (User Defined Commands/Functions)




For details: http://www.iges.org/grads (GrADS), http://opengrads.org/ (OpenGrADS)

# What's GrADS?

Display Window




Enter commands, and
Set visualizing properties

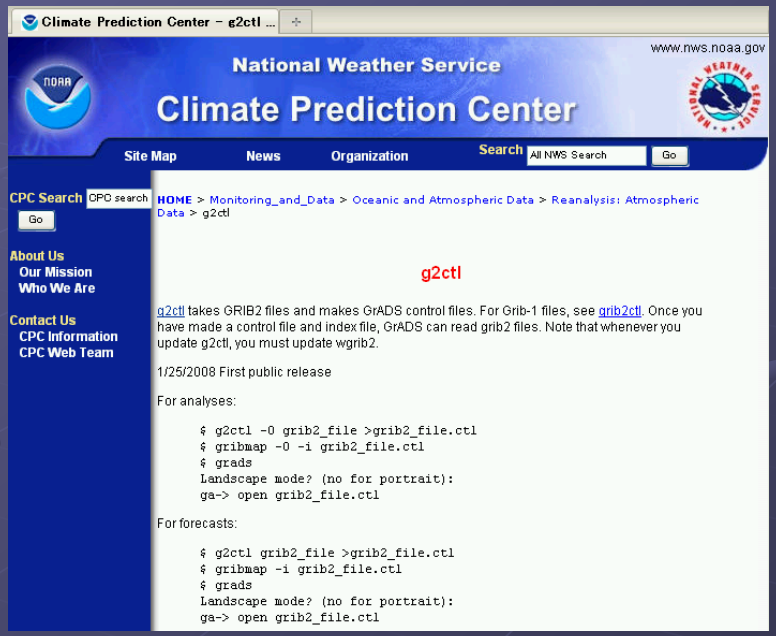
```
POSIX Shell (bash)
  Error ocurred at column 1
DISPLAY error:  Invalid expression
  Expression = skip(uvelprs,5,5)
ga-> d skip(ugrdprs,5,5);vgrdprs;mag(ugrdprs,vgrdprs)
ga-> c
ga-> set gxout shaded
ga-> d strm850mb
Contouring: -2.5e+07 to 4e+07 interval 5e+06
ga-> d skip(ugrdprs,5,5);vgrdprs;
Syntax Error:  Expected operand or '('
  Error ocurred at column 1
DISPLAY error:  Invalid expression
  Expression =
ga-> d skip(ugrdprs,5,5);vgrdprs
ga-> c
ga-> d strm850mb
Contouring: -2.5e+07 to 4e+07 interval 5e+06
ga-> d skip(ugrdprs,10,10);vgrdprs
ga-> c
ga-> set lon -30 330
LON set to -30 330
ga-> d strm850mb
Contouring: -2.5e+07 to 4e+07 interval 5e+06
ga-> d skip(ugrdprs,8,8);vgrdprs
ga->
```

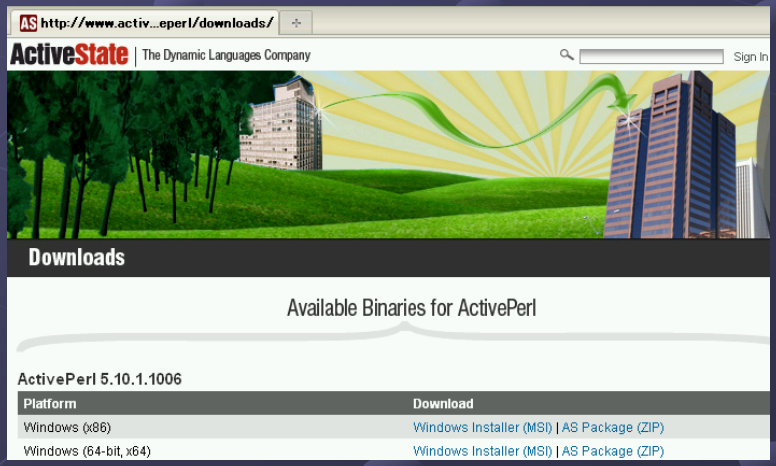
Visualize image in synchronization with commands

Console Window

# Installing grib2ctl.pl, g2ctl.pl

- Download Perl scripts from “CPC/NOAA” website, and allocate them adequately
  - grib2ctl.pl
    - http://www.cpc.noaa.gov/products/wesley/grib2ctl.html
  - g2ctl.pl
    - http://www.cpc.noaa.gov/products/wesley/g2ctl.html
- Download and Install Perl
  - ActivePerl (for Windows)
    - http://downloads.activestate.com/ActivePerl/Windows/5.10/ActivePerl-5.10.1.1006-MSWin32-x86-291086.msi



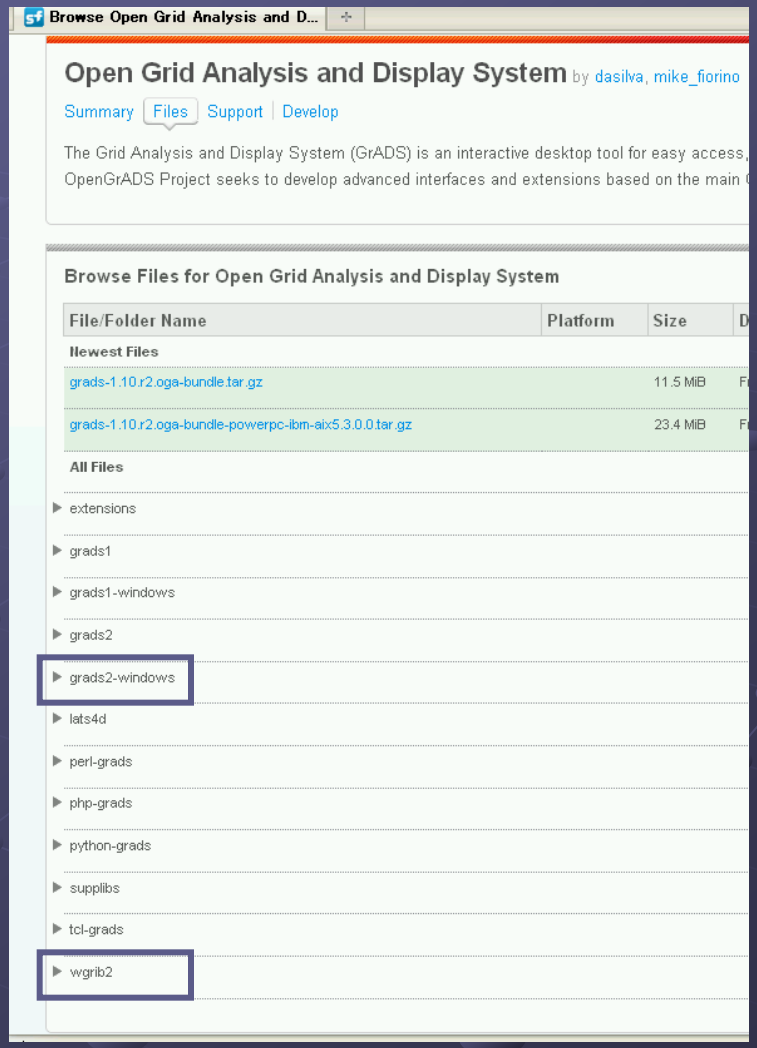



# Installing GrADS (Windows)

- Download installation binary (.exe) from "SourceForge" website
  - Filename:

    grads-2.0.*-win32_superpack.exe
- Execute the binary
  - Automatic installation
  - Follow installer's guidance

For details:
http://opengrads.org/wiki/index.php?title=Installing_GrADS_v2.0_on_Microsoft_Windows

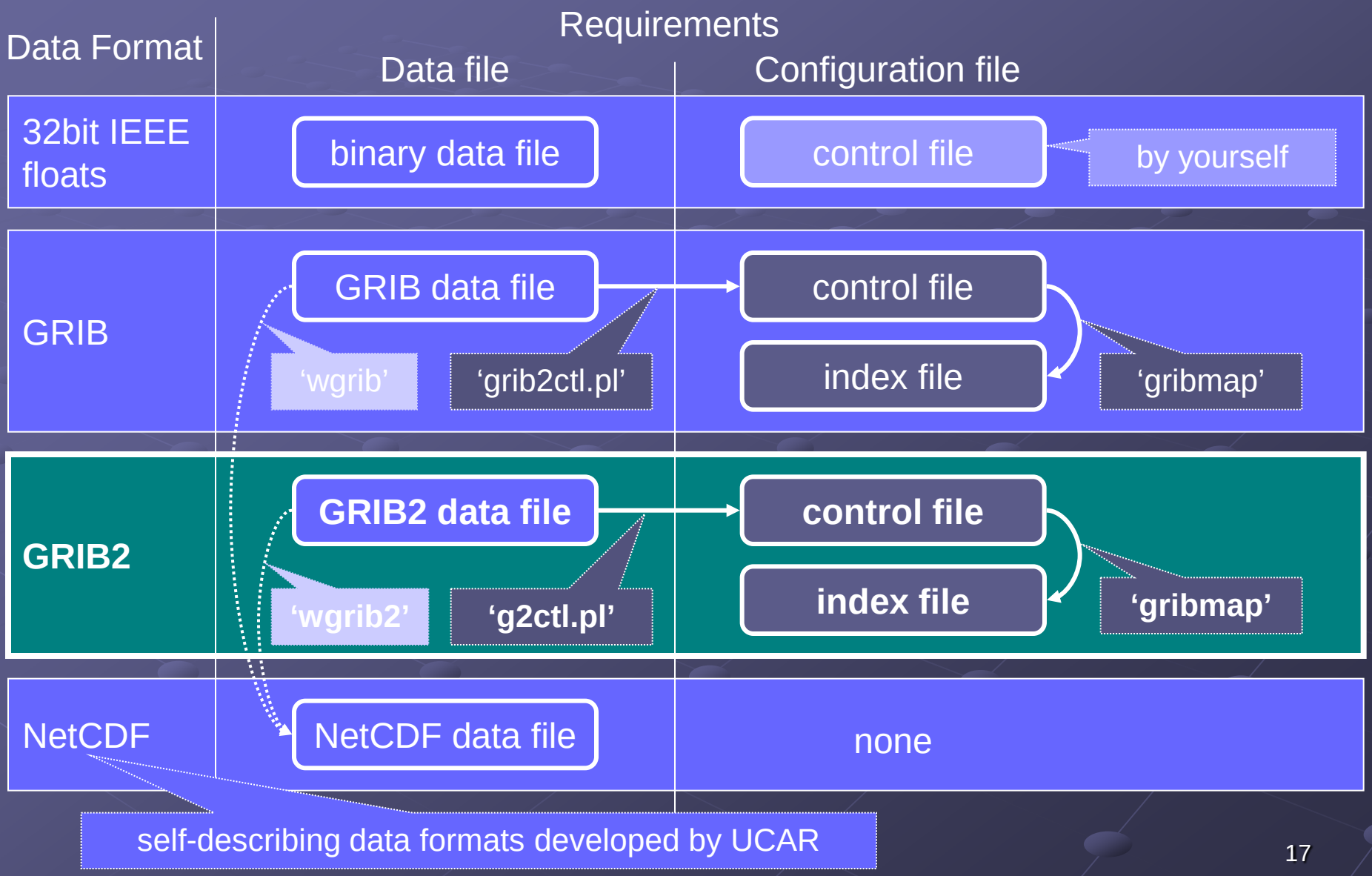


# Installed GrADS Directories

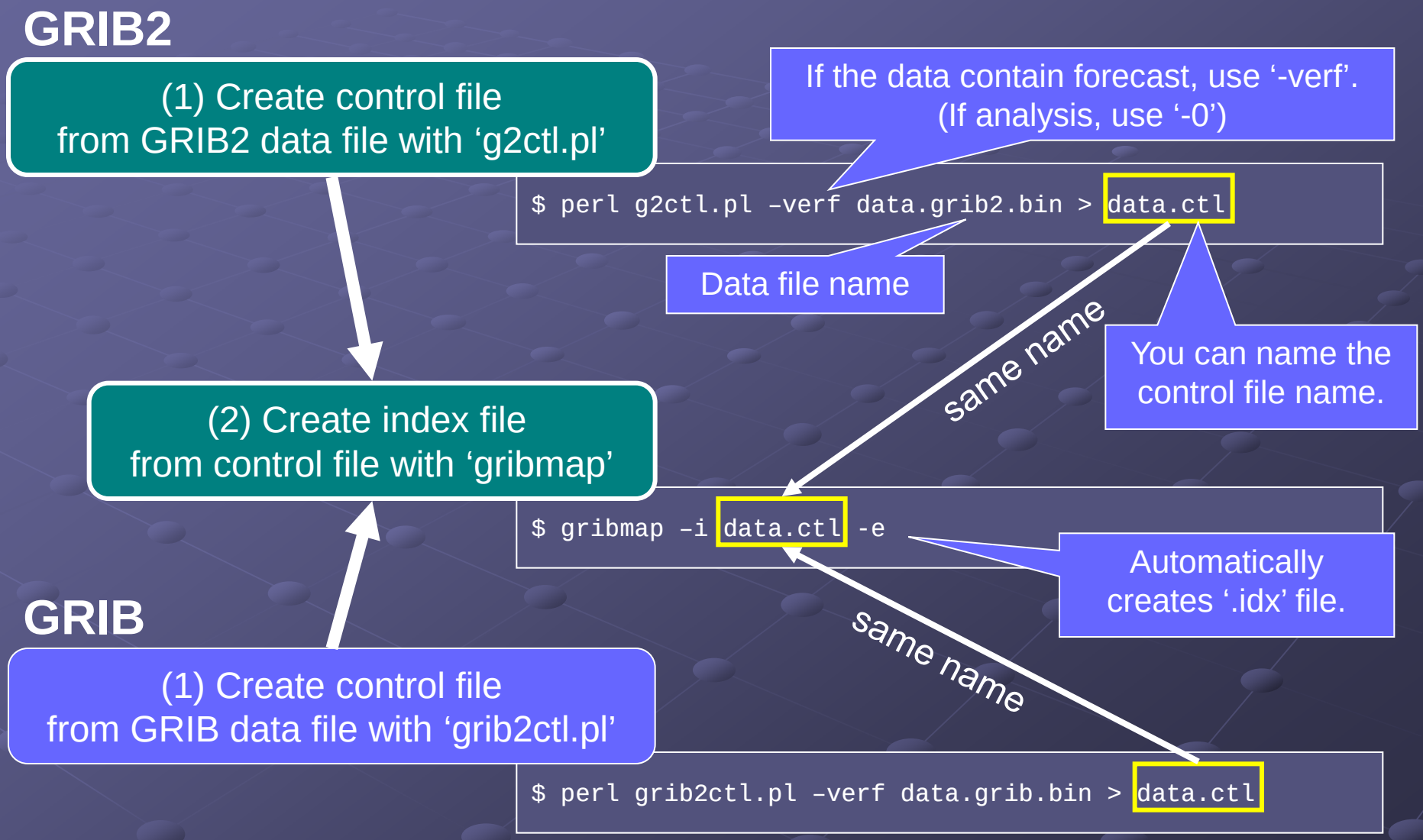
GrADS_ROOT/ — e.g.) C:¥OpenGrADS

- Contents/ --- map data bases, GrADS fonts
  - Resources/
    - SampleDatasets/ --- sample data files
    - Documentation/ --- user's guide, references, and so on.
    - Scripts/ --- GrADS utility scripts
    - SupportData/ --- fonts, coastal data
    - Xming/ --- bundled Xming X window server
  - Cygwin/ --- programs, DLLs
- Classic/ --- Shortcuts for conventional user.

# Requirements for visualizing with GrADS



# Preparation for GRIB / GRIB2



# Control file (GRIB2): Sample

```
dset  ^GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin
index ^GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin.idx
undef 9.999E+20
title GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin
* produced by g2ctl v0.0.4n
* griddef=1.1:0:(720 x 361):grid_template=0:lat-lon grid:(720 x 361) units 1e-006
  input WE:NS output WE:SN res 48 lat 90.0 to -90.0 by 0.5 lon 0.0 to 359.5 by 0.5
  #points=259920:winds(N/S)
dtype grib2
ydef 361 linear -90.000000 0.5
xdef 720 linear 0.000000 0.500000
tdef 1 linear 12Z22aug2009 1mo
* PROFILE hPa
zdef 21 levels 100000 97500 95000 92500 90000 85000 80000 70000 60000 50000 40000 30000
               25000 20000 15000 10000 7000 5000 3000 2000 1000
options pascals
vars 11
HGTprs     21,100  0,3,5 ** (1000 975 950 925.. 50 30 20 10) Geopotential Height [gpm]
RELV500mb   0,100,50000  0,2,12 ** 500 mb Relative Vorticity [1/s]
RHprs      21,100  0,1,1 ** (1000 975 950 925.. 50 30 20 10) Relative Humidity [%]
STRM850mb   0,100,85000  0,2,4 ** 850 mb Stream Function [m^2/s]
STRM200mb   0,100,20000  0,2,4 ** 200 mb Stream Function [m^2/s]
TMPprs     21,100  0,0,0 ** (1000 975 950 925.. 50 30 20 10) Temperature [K]
UGRDprs    21,100  0,2,2 ** (1000 975 950 925.. 50 30 20 10) U-Component of Wind [m/s]
VGRDprs    21,100  0,2,3 ** (1000 975 950 925.. 50 30 20 10) V-Component of Wind [m/s]
VPOT850mb   0,100,85000  0,2,5 ** 850 mb Velocity Potential [m^2/s]
VPOT200mb   0,100,20000  0,2,5 ** 200 mb Velocity Potential [m^2/s]
VVELprs    21,100  0,2,8 ** (1000 975 950 925.. 50 30 20 10) Vertical Velocity (Pressure) [Pa/s]
ENDVARS
```

# ol file (GRIB2): Sample

"^" indicates same directory with control file.

```
dset ^GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin
index ^GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin.idx
undef 9.999E+20
title GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin
* produced by g2ctl v0.0.4n
* griddef=1.1:0:(720 x 361):grid_template=0:lat-lon grid.(720 x 361) units 1e-006
  input WE:NS output WE:SN res 48 lat 90.0 to -90.0 by 0.5 lon 0.0 to 359
  #points=259920:winds(N/S)
dtype grib2
ydef 361 linear -90.000000 0.5
xdef 720 linear 0.000000 0.500000
tdef 1 linear 12Z22aug2009 1mo
* PROFILE hPa
zdef 21 levels 100000 97500 95000 92500 90000 85000 80000 70000
               25000 20000 15000 10000 7000 5000 3000 2000 1000
options pascals
vars 11
HGTprs     21,100  0,3,5 ** (1000 975 950 925.. 50 30 20 10) Geopotential Height [gpm]
RELV500mb   0,100,50000  0,2,32 ** 500 mb Relative Vorticity [1/s]
RHprs      21,100  0,1,1 ** (1000 975 950 925.. 50 30 20 10) Relative Humidity [%]
STRM850mb   0,100,85000  0,2,4 ** 850 mb Stream Function [m^2/s]
STRM200mb   0,100,20000  0,2,4 ** 200 mb Stream Function [m^2/s]
TMPprs     21,100  0,0,0 ** (1000 975 950 925.. 50 30 20 10) Temperature [K]
UGRDprs    21,100  0,2,2 ** (1000 975 950 925.. 50 30 20 10) U-Component of Wind [m/s]
VGRDprs    21,100  0,2,3 ** (1000 975 950 925.. 50 30 20 10) V-Component of Wind [m/s]
VPOT850mb   0,100,85000  0,2,5 ** 850 mb Velocity Potential [m^2/s]
VPOT200mb   0,100,20000  0,2,5 ** 200 mb Velocity Potential [m^2/s]
VVELprs    21,100  0,2,8 ** (1000 975 950 925.. 50 30 20 10) Vertical Velocity (Pressure) [Pa/s]
ENDVARS
```

Data file path

Index file path

Definition of 'undefined value' (=No Data)

Title

Comment

Grid definition for Y-axis (grid size, latitude of southern end, interval)

Grid definition for X-axis (grid size, longitude of western end, interval)

Grid definition for Time-axis (size, initial date/time, interval)

Grid definition for Vertical-axis (grid size, levels)

Number and information of elements

# Preparation for time-series dataset

- How to treat separated data files by forecast time
  - Concatenate GRIB files by “cat” command before making Control file

```
GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0000_grib2.bin
GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0003_grib2.bin
GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0006_grib2.bin
   :
   :
GSM_GPV_Rgl_Gll0p5deg_L-pall_FD0312_grib2.bin
```

Command “cat” :
1) Display contents of file
2) Concatenate files

```
$ cat GSM_GPV_Rgl_Gll0p5deg_L-pall_*_grib2.bin
 > GSM_GPV_Rgl_Gll0p5deg_L-pall_grib2.bin
```

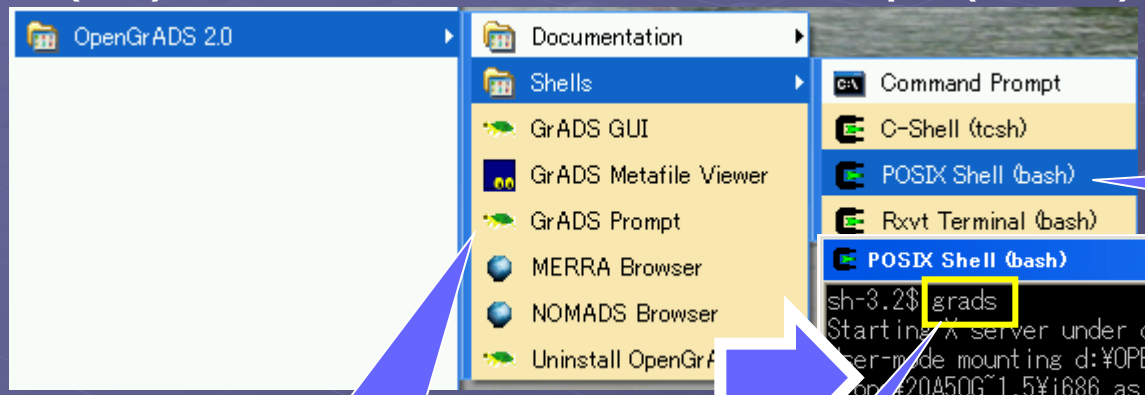
```
GSM_GPV_Rgl_Gll0p5deg_L-pall_grib2.bin
```

```
$ g2ctl ……
$ gribmap ……
```

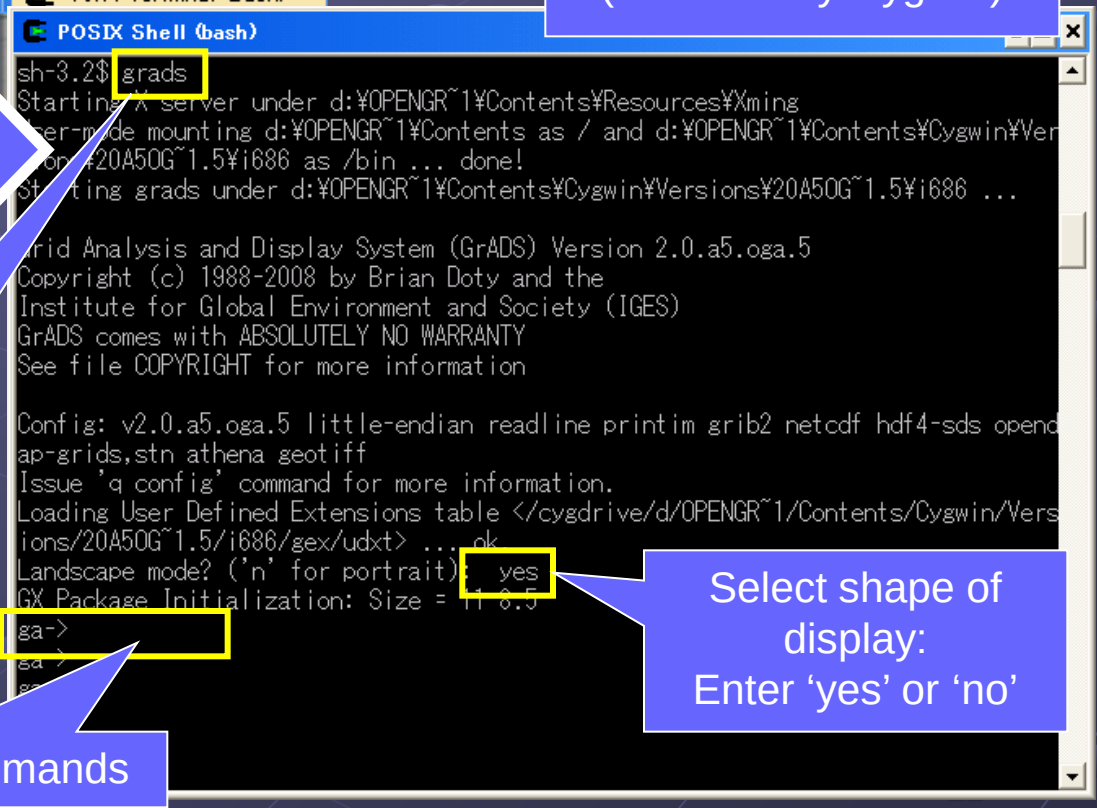
# Visualizing by GrADS CUI

## - Lesson 1 : How to launch GrADS -

### ( 1 ) Launch Command Prompt (Shell) and GrADS




Command prompt which enables using 'bash' for Unix/Linux.
(Powered by Cygwin)




Launch GrADS command prompt directly.

Enter 'grads'

Select shape of display:
Enter 'yes' or 'no'

Enter GrADS commands

# Visualizing by GrADS CUI

## - Lesson 2 : How to open data and visualize them -

( 2 ) Open control file. ( “ga->” means GrADS command prompt)

```
ga-> open data.ctl
```

Control file name that was created by g2ctl.pl

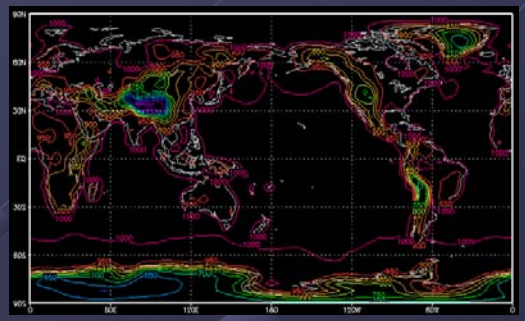
( 3 ) Check available elements.

```
ga-> q ctlinfo  (query ctlinfo)
```

***Italic words*** depend on your environment / setting

( 4 ) Visualize for an element (e.g. Surf. Pres.).

```
ga-> d pressfc (display pressfc)
```



Element name is shown in results of ‘q ctlinfo’.

( 5 ) Clear screen.

```
ga-> c (clear)
```

To reset settings, use ‘reinit’.

# Visualizing by GrADS CUI

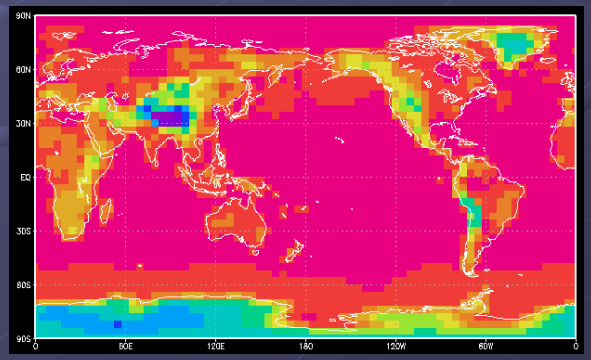
## - Lesson 3 : How to draw by various rendering methods -

( 6 ) Use ‘contour’ option (default).

```
ga-> set gxout contour
ga-> d pressfc
```

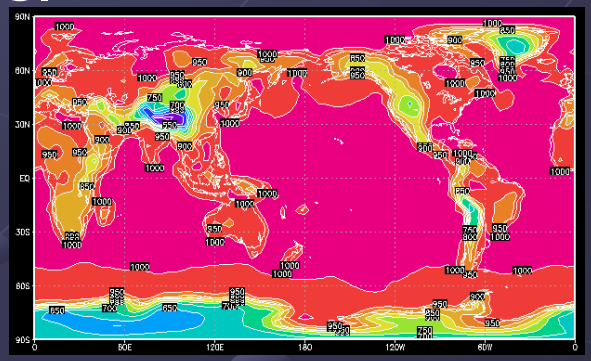
( 7 ) Use ‘grid-fill’ option.

```
ga-> set gxout grfill
ga-> d pressfc
```




( 8 ) Overlay ‘shaded’ and ‘contour’ images.

```
ga-> set gxout shaded
ga-> d pressfc
ga-> set gxout contour
ga-> d pressfc
```



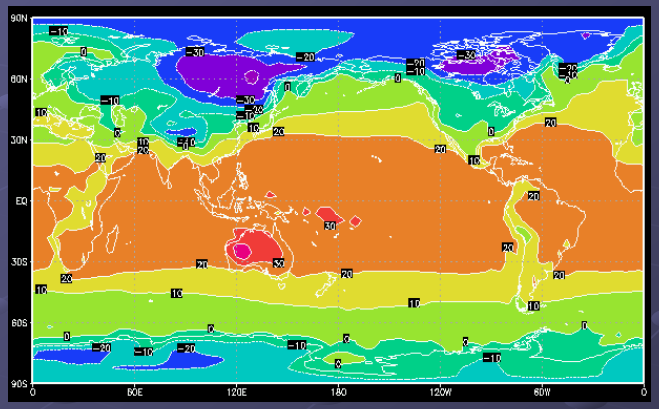
# Visualizing by GrADS CUI

## - Lesson 4 : How to change latitude and longitude -

( 6 ) Set latitude range

```
ga-> set LAT -15 25
```

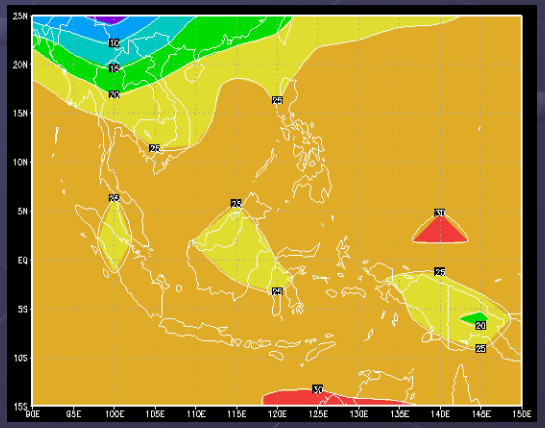
Latitude Range (South North)
( South Latitude: negative value )



( 7 ) Set longitude range

```
ga-> set LON 90 150
```

Longitude Range (West East)
( West Longitude: negative value )



# Visualizing by GrADS CUI

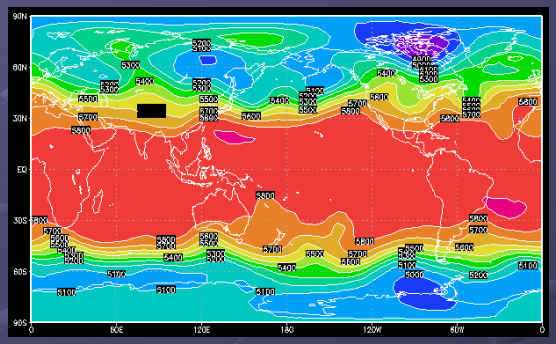
## - Lesson 5 : How to change level -

( 8 ) Change level.

```
ga-> set LEV 500
```

( 9 ) Visualize data (e.g. Geopotential height)

```
ga-> d hgtprs    # Draw Z at 500hPa
```



## - Lesson 6 : How to change target time -

( 10 ) Change target time.

```
ga-> set t 2  (set time 00Z02oct2009)
```

( 11 ) Visualize data (e.g. Geopotential height)

```
ga-> d hgtprs
```

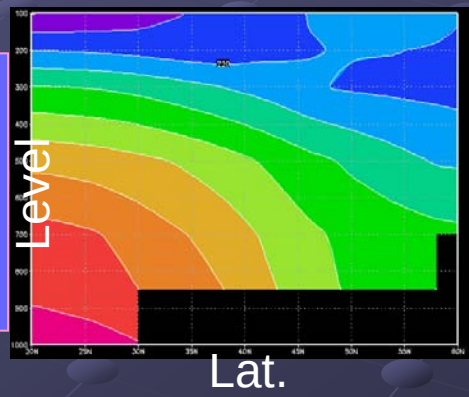
# Visualizing by GrADS CUI

## - Lesson 7 : How to visualize section -

GrADS treats 4 dimensions (latitude and longitude, level and time). If you set these to range value or single value, you can draw many kinds of figures.

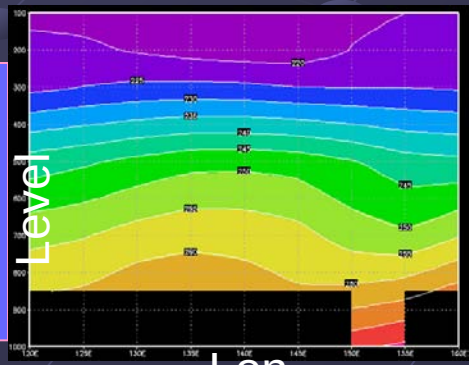
### ( 12 ) Visualize with longitude fixed

```
ga-> set LON 140    # Fix as 140E
ga-> set LAT 20 60    # 20N – 60N
ga-> set LEV 1000 100    # 1000hPa – 100hPa
ga-> d tmpprs    # Valuable: Lat., Level
```




### ( 13 ) Visualize with latitude fixed

```
ga-> set LON 120 160    # 120E – 160E
ga-> set LAT 40    # Fix as 40N
ga-> set LEV 1000 100    # 1000hPa – 100hPa
ga-> d tmpprs    # Valuable: Lon., Level
```



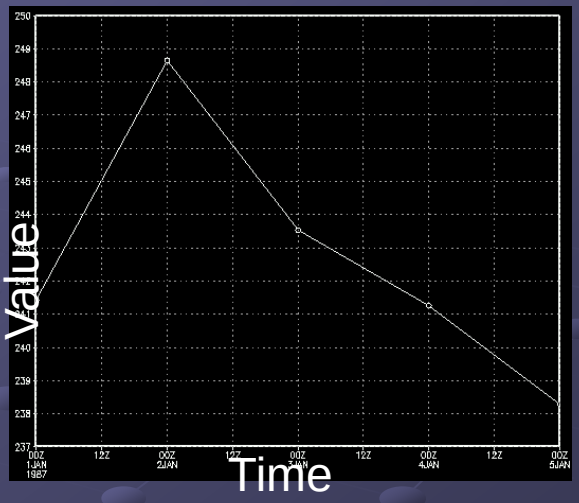
# Visualizing by GrADS CUI

## - Lesson 8 : How to visualize time graph -

( 14 ) Visualize time-series graph

```
ga-> set LON 140
ga-> set LAT 40
ga-> set LEV 500
ga-> set t 1 5
ga-> d tmpprs
              # Valuable: 1 dimension(Time)
```

Fix as Single value




## - Lesson 9 : How to visualize plane map -

( 15 ) Visualize plane map

```
ga-> set LON 20 60
ga-> set LAT 120 160
ga-> set LEV 500
ga-> set t 1
ga-> d tmpprs
      # Valuable: 2 dimensions(Lon.,Lat.)
```

Fix as Single value

# Visualizing by GrADS CUI

## - Lesson 10 : How to visualize wind -

( 16 ) Visualize wind strength

Zonal Component

Meridional Component

```
ga-> d mag(ugrdprs,vgrdprs) # Wind Magnitude
```

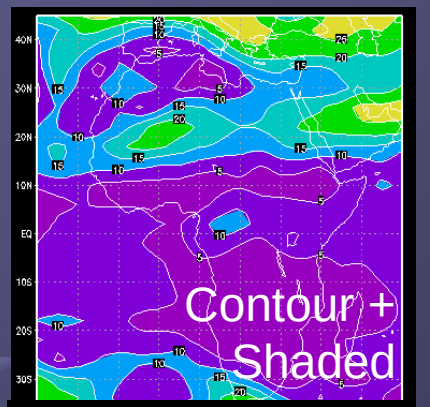
Note: mag(ugrdprs, vgrdprs) = sqrt(ugrdprs*ugrdprs+vgrdprs*vgrdprs)

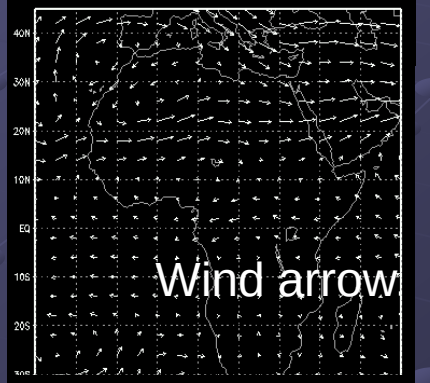
( 17 ) Visualize wind arrow (as vector)

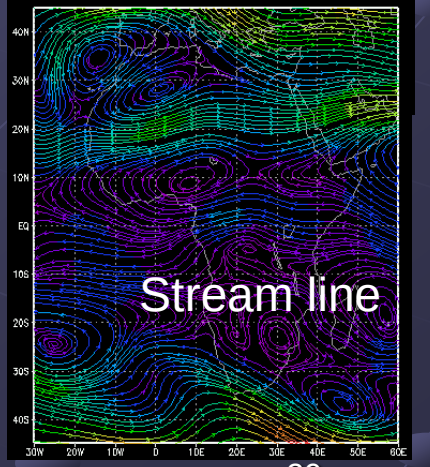
```
ga-> set gxout vector
ga-> d ugrdprs;vgrdprs
```

( 18 ) Visualize stream line (color indicates wind strength)

```
ga-> set gxout stream
ga-> d ugrdprs;vgrdprs;mag(ugrdprs,vgrdprs)
```



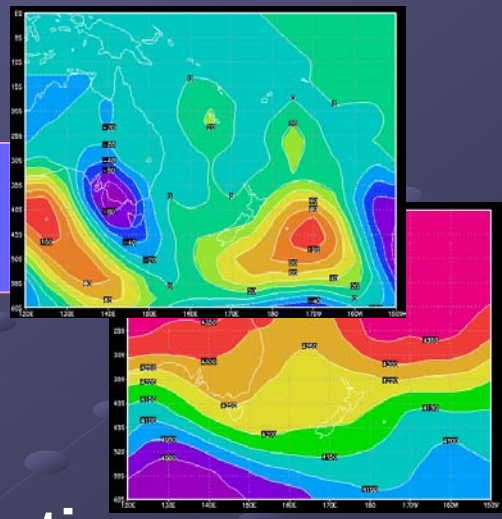



# Visualizing by GrADS CUI

## - Lesson 11 : How to calculate / visualize the differences -

( 19 ) Calculate and visualize the differences

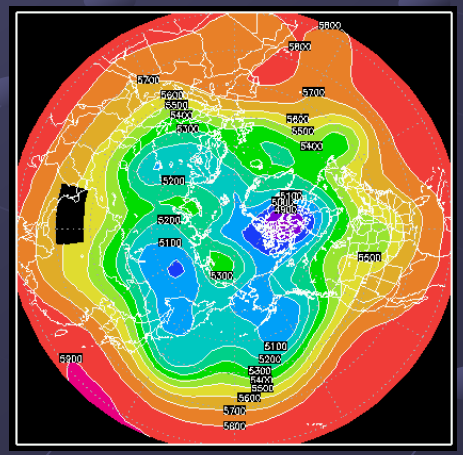
```
ga-> d hgtprs(t=2)-hgtprs(t=1)
ga-> d hgtprs(lev=500)-hgtprs(lev=850)
```



## - Lesson 12 : How to use various map projections -

( 20 ) Visualize by using Northern Hemisphere Polar Stereo projection

```
ga-> set mproj nps        # Projection
ga-> set mpdset mres      # Resolution
ga-> set lat 20 90        # Region
ga-> d hgtprs
```



# Export images by GrADS CUI

**- Lesson 13 : How to export into image file -**

( 21 ) Export as GrADS metafile,
and Convert into image (e.g. format: PNG)

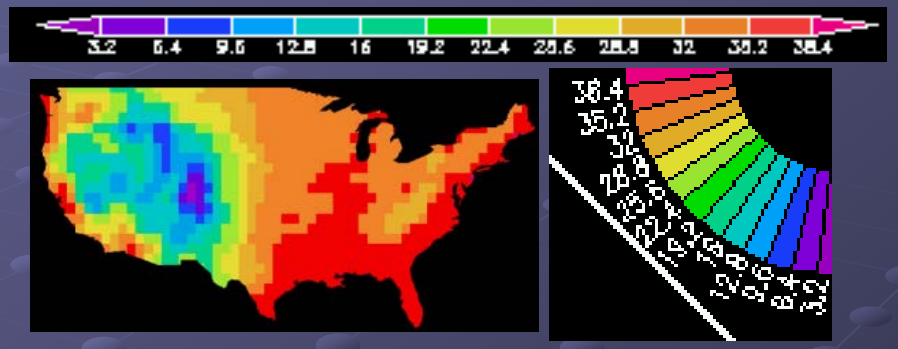
```
ga-> enable print image.gm
ga-> print
ga-> disable print
ga-> !gxyat –o image.png image.gm
     (or exit GrADS, and type
      $ gxyat –o image.png image.gm)
```

The suffixes “.gm” or “.gmf” are typical for GrADS metafile.
Format of image file: PNG, PostScript, PDF, SVG (Scalable Vector Graphics)

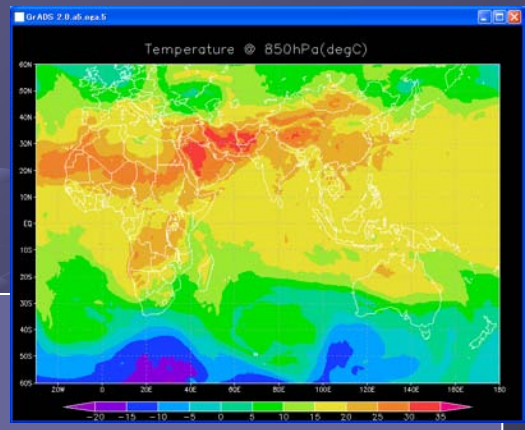
# GrADS Script Library

- Useful GrADS Utilities
  - cbarn.gs, cbarc.gs, …
    - Draw legends
  - map.gs
    - Automates settings for a variety of useful map projections
  - basemap.gs
    - Overlays a land or ocean mask that exactly fits the coastal outlines

For details:
http://www.iges.org/grads/gadoc/library.html




# GrADS Scripting Language




- User facility to program GrADS operations
  - Automatic production
  - Saving procedures
- Run by GrADS

```
ga-> run script.gs
   or
$ grads –c script.gs
```

For details:
http://www.iges.org/grads/gadoc/script.html

```
*GrADS script sample 1:

'open ./tl959_p-all_0p5.ctl'

'set lon -30 180'
'set lat -60 60'
'set lev 850'

'set mpdset mres'
'set gxout shaded'
'set grads off'

'set strsiz 0.2'
'set string 1 c 3'
'draw string 5.25 8 Temperature @ 850hPa(degC)'

'd tmpprs - 273.15'

'run cbarn.gs'
```

* … : Comments
'…' : GrADS statements

Exchange (K) to (degC)

Call GrADS library

# Sample: Overlay data and print image

```
* GrADS script sample 2:

'open ./tl959_surf_0p25.ctl'

'set lon 110 150'
'set lat 5 35'
'set mpdset hires'

'set gxout shaded'
'set clevs 0.1 10 20 40 60 80 100'
'set cmin 0.1'
'd maskout(apcpsfc, apcpsfc - 0.1)'
'cbarn'

'set gxout contour'
'd prmslmsl / 100'

'set gxout vector'
'd skip(ugrd10m, 5, 5);vgrd10m'

'draw title Pres.@MSL(hPa), Wind@10m(m/s), Prec.6hr(mm)'

'enable print tmp.gm'
'print'
'disable print'

'!gxyat -i tmp.gm -o output_sample2.png'
```

Use high-resolution coastal map dataset

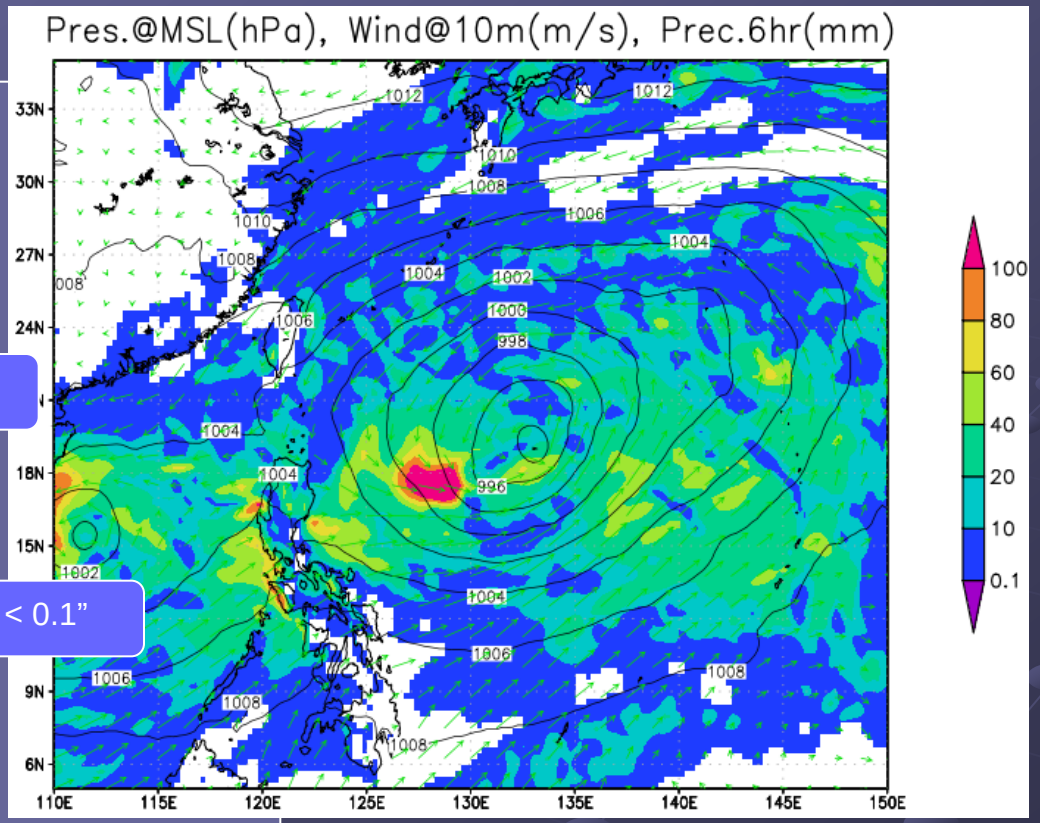
Change intervals of contour

Not draw if “apcpsfc < 0.1”

Alias of ‘run cbarn.gs’

Export as GrADS meta image

Convert image into PNG format image



# Sample: Using arguments, prompts

```
* GrADS script sample 3:

function main(args)
  ctl_file = "./tl959_p-all_0p5.ctl"

  if subwrd(args, 1) != ""
    level = subwrd(args, 1)
  else
    prompt "Enter Level : "
    pull level
  endif

  'open 'ctl_file
  'set lon 30 210'
  'set lat -60 60'
  'set lev 'level
  'set gxout shaded'
  'd hgtprs'

  'set gxout vector'
  'd skip(ugrdprs,5,5);vgrdprs'

  'set strsiz 0.2';'set string 1 c 3'
  'draw string 5.25 8 Z,U,V @ 'level'hPa(degC)'
  'run cbarn.gs 0.8 1 10.25 4.25'

  prompt "Press any key to quit."
  pull answer

  'quit'
return
```

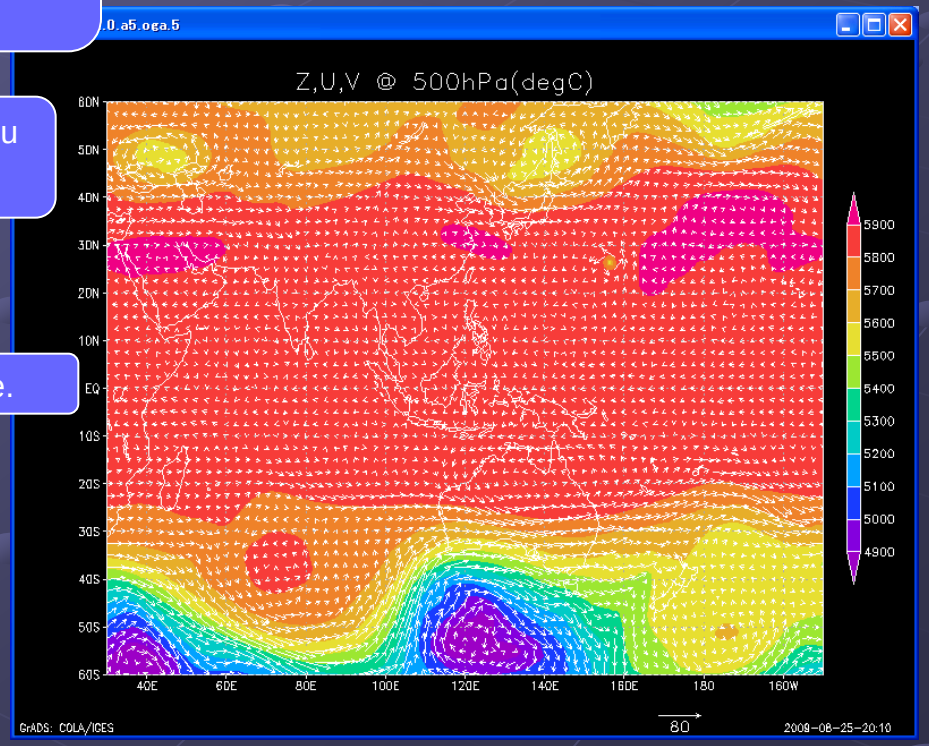
Arguments

If you run script by "run sample3.gs 850", level will be automatically set as "level = 850".

If you don't set arguments, you can enter level manually.

Config and draw title.

```
ga-> run sample3.gs
Enter Level : 500
Press any key to quit.
```



# Sample: Extracting values at 1 grid point

```
* GrADS script sample 4:

function main()

  ctl_file = "./tl959_p-all_0p5.ctl"
  out_file = "./output_sample4.txt"

  say "Geopotential Height (gpm)"
  output = write(out_file, "Geopotential Height (gpm)")

  'open 'ctl_file
  'set lon 140'
  'set lat 36'
  'set t 1'
  'set gxout value'

  Z = 1

  while(Z<=21)
    'set z 'Z
    zlev = subwrd(result, 4)
    'd hgtprs'
    height = subwrd(result, 4)
    say zlev' 'height
    output = write(out_file, zlev' 'height, append)

    Z = Z + 1
  endwhile

  'quit'
return
```

Write data to out_file

Loop 21 times
(Z=1, 2, 3, … ,21)

```
Geopotential Height (gpm)
1000 28.7195
975 252.417
950 480.678
925 713.9
          :
          :
```

output_sample4.txt

```
POSIX Shell (bash)
ga-> run sample3.gs
Geopotential Height (gpm)
1000 28.7195
975 252.417
950 480.678
925 713.9
900 952.657
850 1447.85
800 1967.55
700 3093.19
600 4364.55
500 5828.22
400 7560.92
300 9693.23
250 10975.8
200 12470.6
150 14300
100 16753.6
70 18878.9
50 20956.2
30 24201.3
20 26830.5
10 31470.1
No hardcopy metafile open
GX package terminated
```

# Sample: Extracting values at 1 grid point

```
* GrADS script sample 5:
```

```
* GrADS script sample 5:

function main()

  ctl_file = "./tl959_p-all_0p5.ctl"
  out_file = "./output_sample5.txt"

  say "Geopotential Height (gpm)"
  output = write(out_file, "Geopotential Height (gpm)")

  'open 'ctl_file
  'set lon 140'
  'set lat 36'
  'set t 1'

  'set z 1 21'
  'set gxout print'
  'set prnopts %10.2f 1 1'
  'd hgtprs'

  output = write(out_file, result)
  output = close(out_file)

  'quit'
return
```

Output single value

Format outputs

Output at once without ‘while’ loop

```
Geopotential Height (gpm)
      28.71
     252.41
     480.67
     713.90
          :
          :
```

output_sample5.txt

```
POSIX Shell (bash)
Geopotential Height (gpm)
Printing Grid -- 21 Values -- Undef = 9.999e+20
     28.72
    252.42
    480.68
    713.90
    952.66
   1447.85
   1967.55
   3093.19
   4364.55
   5828.22
   7560.92
   9693.23
  10975.77
  12470.55
  14300.04
  16753.58
  18878.86
  20956.16
  24201.27
  26830.52
  31470.05

sh-3.2$
```

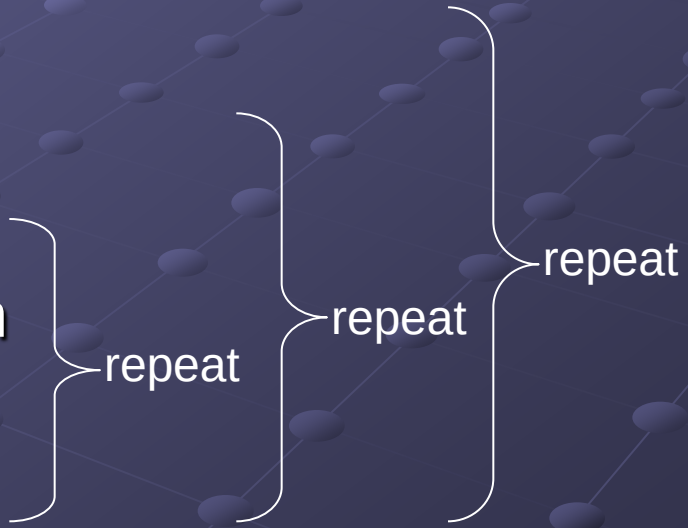
# Reference: GRIB(GRIB Edition 1)

- Format
  - SECTION0: Indicator (the first 4byte is the ASCII ‘GRIB’)
  - SECTION1: Product definition
  - SECTION2: Grid description
  - SECTION3: Bitmap
  - SECTION4: Binary data
  - SECTION5: End (the last 4byte is the ASCII ‘7777’)

For details: http://www.wmo.int/pages/prog/www/WMOCodes/GRIB.html

# Reference: GRIB2(GRIB Edition 2)

- Format
  - SECTION0: Indicator (the first 4byte is the ASCII ‘GRIB’)
  - SECTION1: Identification
  - SECTION2: (Local use section)
  - SECTION3: Grid definition
  - SECTION4: Product definition
  - SECTION5: Data Representation
  - SECTION6: Bitmap
  - SECTION7: Data
  - SECTION8: End (the last 4byte is the ASCII ‘7777’)




For details: http://www.wmo.int/pages/prog/www/WMOCodes/GRIB.html

# References: Preparation for time-series dataset (Analysis)

- How to treat many initials by one Control file (Analysis)
  - Edit Control file, and apply template for “dset” before making Index file

Example:

```
Data/Anal/Hist/Daily/20090930.grib2
Data/Anal/Hist/Daily/20091001.grib2
Data/Anal/Hist/Daily/20091002.grib2
Anal_Hist_Daily.ctl
```

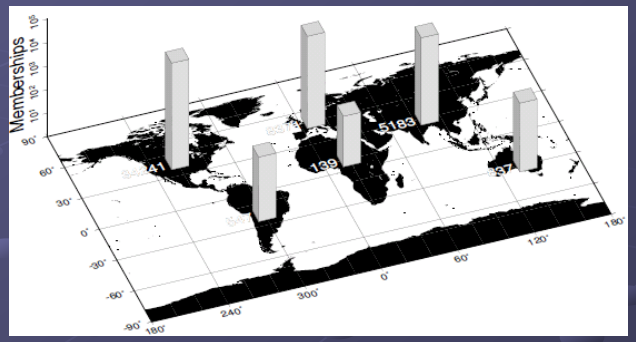
```
%y4 : year  4 digits
%m2 : month 2 digits
%d2 : date  2 digits
%h2 : hour  2 digits
```

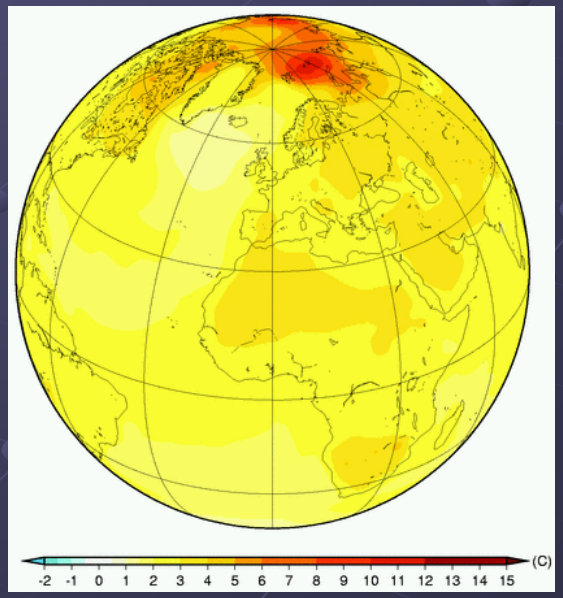
```
dset  ^Data/Anal/Hist/Daily/%y4%m2%d2.grib2
index ……
   :
options pascals template
   :
```

Contents of “Anal_Hist_Daily.ctl”

# Reference: GMT

- GMT (Generic Mapping Tool)
  - Software package for reading script (batch) files, generating fine PostScript images
  - Easier than GrADS when visualizing Station-Data
  - Interactive operation is not supported.
    - ‘iGMT’ would be enable interactive operation?
  - GRIB/GRIB2 are not supported.




For details:
http://gmt.soest.hawaii.edu/

# Reference: NCARG / PyNGL

- NCARG(NCAR Graphics)
  - Software package for scientific visualization
  - NCAR Command Language (NCL; Fortran/C library)
  - Difficult to master than GrADS/GMT
- PyNGL
  - Python wrapper for NCL
  - Easier than using NCL directly
  - Interactive operation is not supported by NCARG and PyNGL
  - Binary is available for only Linux, Solaris and Macintosh.

For details:
http://www.ncarg.ucar.edu/
http://www.pyngl.ucar.edu/