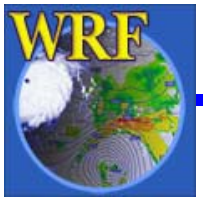
Post-processing Tools

Cindy Bruyère



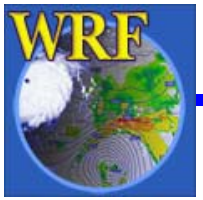
Graphical Packages

- **NCL** UG: 9-2
  - Graphical package
- **ARWpost** UG: 9-28
  - Converter (GrADS & vis5d)
- **RIP4** UG: 9-19
  - Converter and interface to graphical package NCAR Graphics
- **WPP** UG: 9-35
  - Converter (GrADS & GEMPAK)
- **VAPOR** UG: 9-50
  - Converter and graphical package
  - *Support: VAPOR*

unidata.ucar.edu

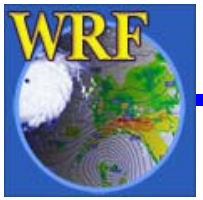
- **IDV**
  - GRIB (from WPP)
  - GEMPAK (from wrf2gem)
  - vis5d (from ARWpost)
  - CF complaint data (from wrf_to_cf)
  - *Support: unidata*
- **GEMPAK**
  - Data from wrf2gem or WPP
  - *Support: unidata*

MatLab / IDL



Graphical Packages

	NCL	RIP4	ARWpost (GrADS / Vis5D)	WPP	VAPOR
Directly ingest WRF data	*Y*	N *converter*	N / *(Y)* *converter*	N *converter*	N *converter*
Intermediate files	*N*	lots	large file	Y	large file
WPS DATA	*Y*	*Y*	*Y*	N	N
wrfinput data	*Y*	*Y*	*Y*	N	N
Idealized data files	*Y*	*Y*	*Y*	N	N
Input format	*netCDF*	*netCDF*	*netCDF / GRIB1*	*netCDF / binary*	*netCDF*
Vertical Output Coordinate	eta pressure height	eta pressure height	eta pressure height	pressure	eta
Software required *(All binaries are free)*	*NCL*	*NCARG*	*GrADS/vis5d*	*GrADS / GEMPAK*	*VAPOR*
Diagnostics	*some*	*> 100*	*some*	*some*	*limited*

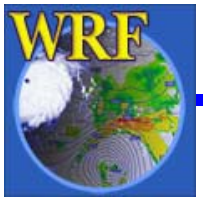


ARWpost / RIP4 / WPP

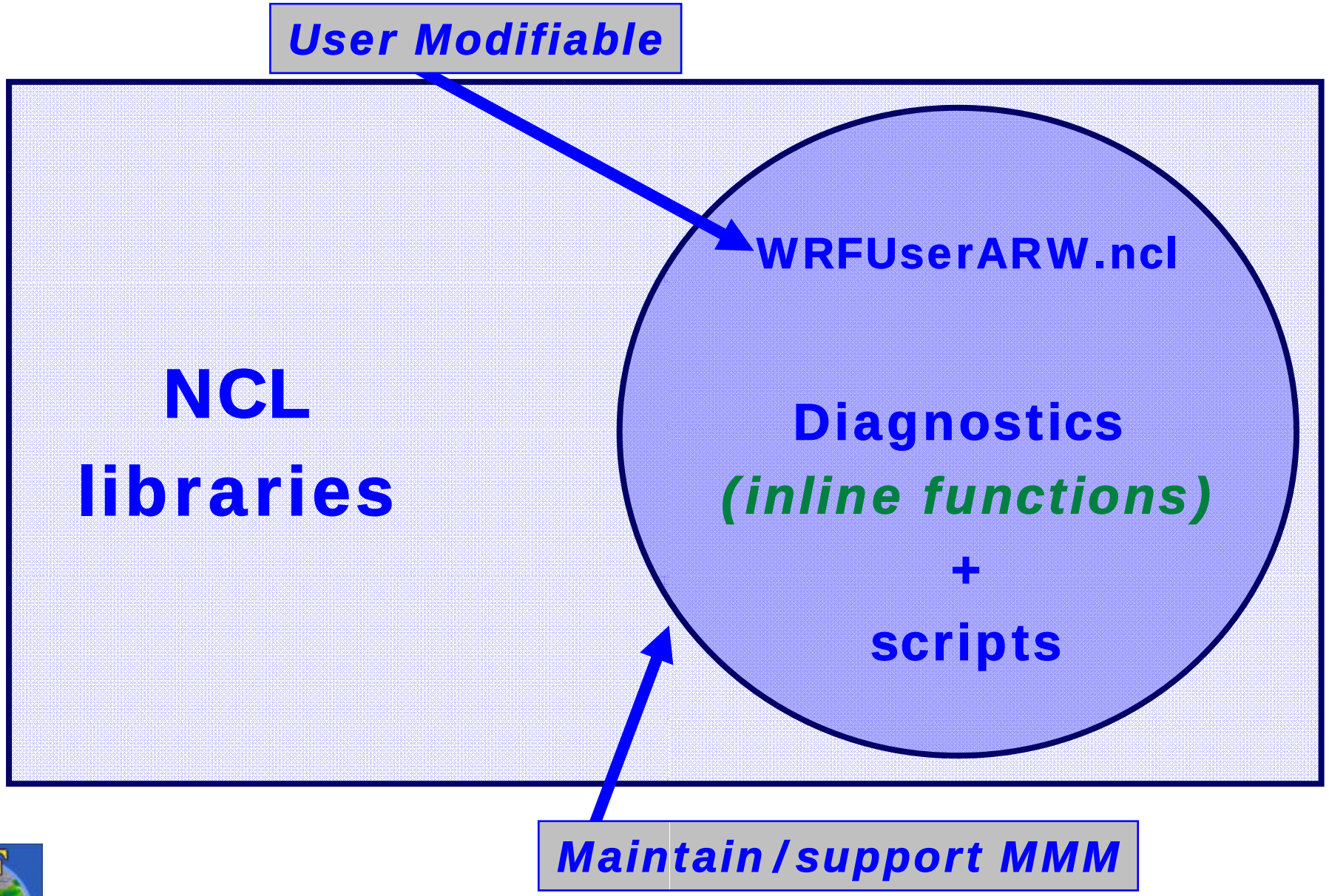
- Download code
- ./configure & ./compile
- Edit namelist
  - Namelist format differ between these three programs
- Obtain a graphical display package
- Run code and display output

NCL

- NCAR Command Language
- http://www.ncl.ucar.edu
- Read WRF model data directly
- Generate a number of graphical plots
  - Horizontal, cross-section, skewT, meteogram, panel

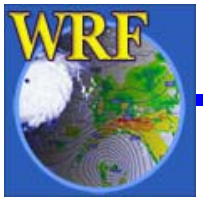


NCL & WRF


User Modifiable
WRFUserARW.ncl
NCL
libraries
Diagnostics
(inline functions)
+
scripts
Maintain / support MMM

Download NCL

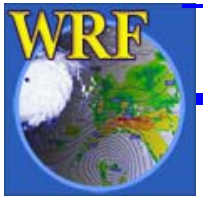
- *http://www.ncl.ucar.edu/Download*
  - Fill out short registration form *(there is a short waiting period)*
  - Read and agree to OSI-based license
  - Download binaries
- NCARG_ROOT environment variable
  - *setenv NCARG_ROOT /usr/local/ncl*
- NCL version 5.1.0 *or later*



Home Directory → ~/.hluresfile ← **Very Important**

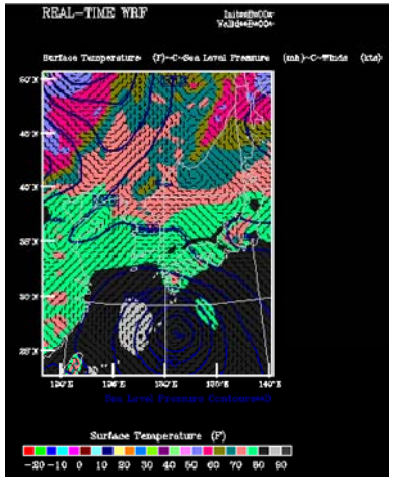
- Required by NCL libraries
- Must be in your "~/" directory *(home directory)*
- Control
  - color table ; font
  - white/black background
  - size of plot
  - control characters

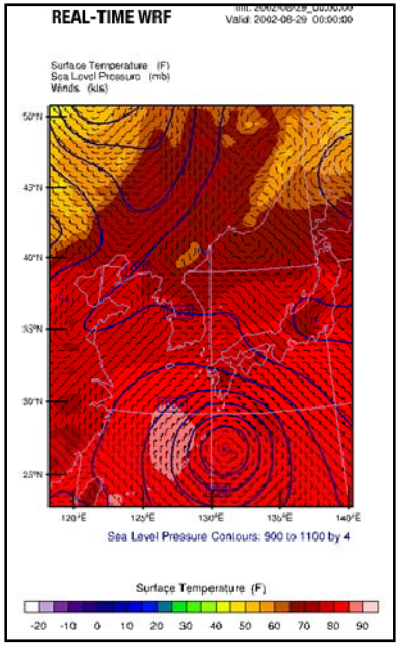
- *http://www.ncl.ucar.edu/Document/Graphics/hlures.shtml*



~/.hluresfile

```
*wkColorMap              : BlAqGrYeOrReVi200
*wkBackgroundColor       : white
*wkForegroundColor       : black
*FuncCode                : ~
*TextFuncCode            : ~
*Font                : helvetica
*wkWidth                 : 900
*wkHeight                : 900
```

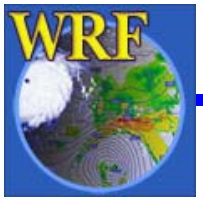




http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/.hluresfile

Generate Plots

- Create a script
  - *wrf_real.ncl*
    (start with a sample script)
- Set NCARG_ROOT environment variable:
  - *setenv NCARG_ROOT /usr/local/ncl*
- Ensure you have an ~/.hluresfile file
- Run NCL script
  - *ncl wrf_real.ncl*



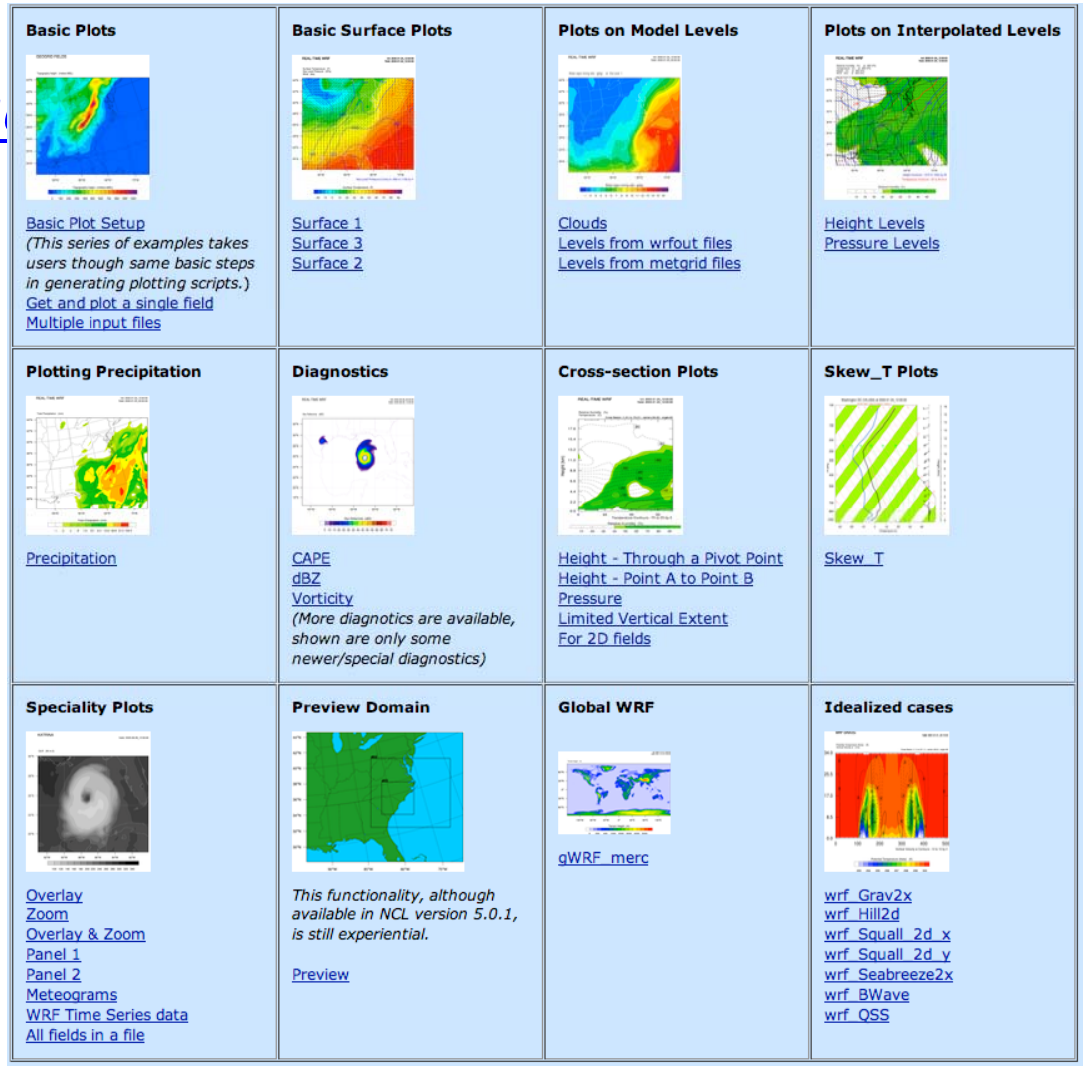
Generate Plots: *A good start – OnLine Tutorial*

http://www.mmm.ucar.e

wrf/OnLineTutorial/

Graphics/

NCL/index.html



Basic Plots

Basic Plot Setup
(This series of examples takes users though same basic steps in generating plotting scripts.)
Get and plot a single field
Multiple input files

Basic Surface Plots

Surface 1
Surface 3
Surface 2

Plots on Model Levels

Clouds
Levels from wrfout files
Levels from metgrid files

Plots on Interpolated Levels

Height Levels
Pressure Levels

Plotting Precipitation

Precipitation

Diagnostics

CAPE
dBZ
Vorticity
(More diagnotics are available, shown are only some newer/special diagnostics)

Cross-section Plots

Height - Through a Pivot Point
Height - Point A to Point B
Pressure
Limited Vertical Extent
For 2D fields

Skew_T Plots

Skew_T

Speciality Plots

Overlay
Zoom
Overlay & Zoom
Panel 1
Panel 2
Meteograms
WRF Time Series data
All fields in a file

Preview Domain

This functionality, although available in NCL version 5.0.1, is still experiential.

Preview

Global WRF

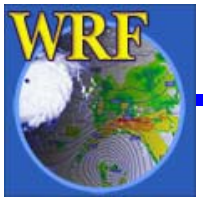
gWRF_merc

Idealized cases

wrf_Grav2x
wrf_Hill2d
wrf_Squall_2d_x
wrf_Squall_2d_y
wrf_Seabreeze2x
wrf_BWave
wrf_QSS

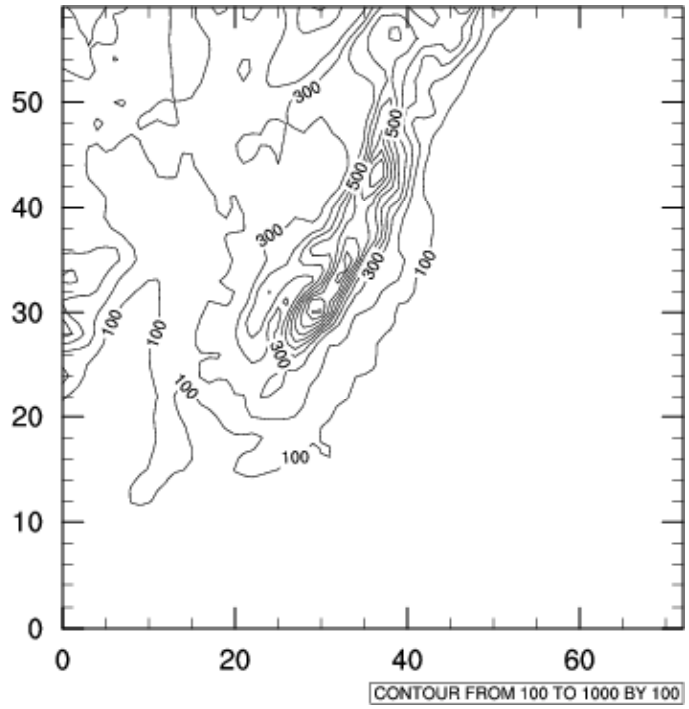
Creating a Plot : NCL script

```
load ncl library scripts

begin

  ; Open graphical output
  ; Open input file(s)

  ; Read variables

  ; Set up plot resources & Create plots
  ; Output graphics

end
```



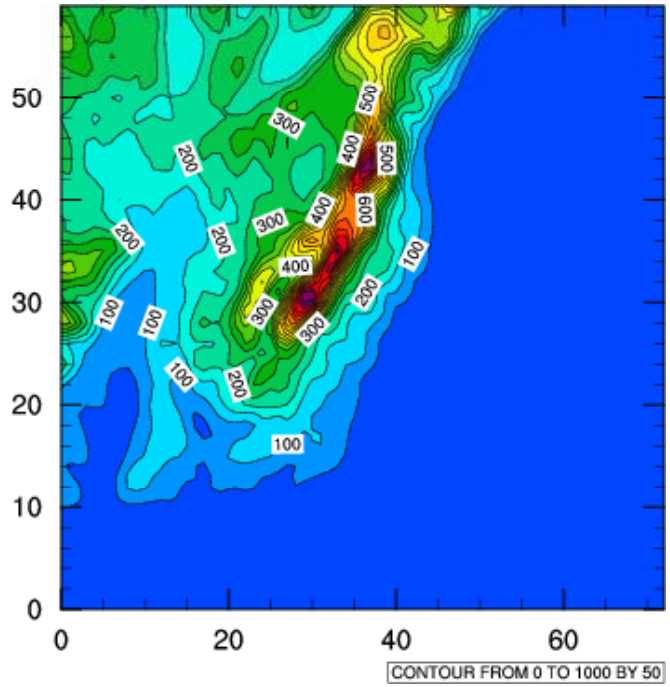
Generate Plots

```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"

begin
  a = addfile("./geo_em.d01.nc","r")
  wks = gsn_open_wks("pdf","plt_ter1")
  ter = a->HGT_M(0,:,:)
  plot = gsn_contour(wks,ter,True)
end
```



Generate Plots

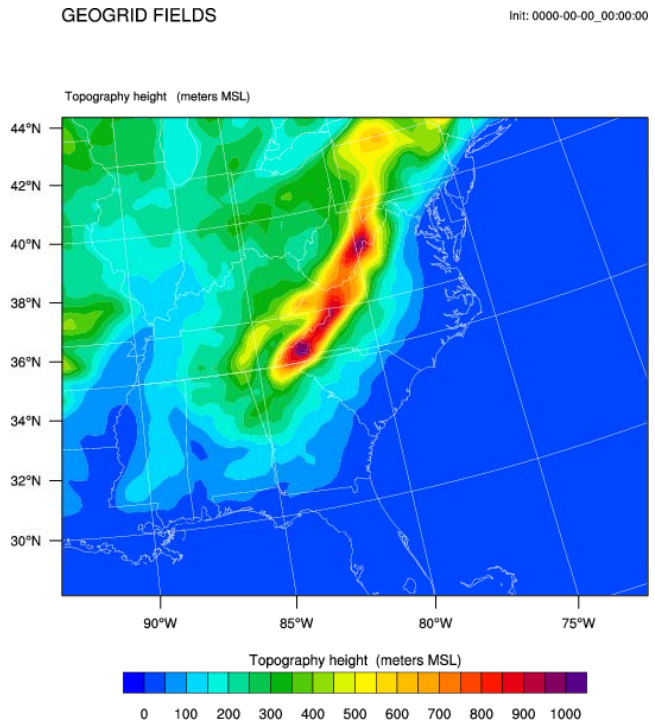
```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"

begin
  a = addfile("./geo_em.d01.nc","r")
  wks = gsn_open_wks("pdf","plt_ter1")
  ter = a->HGT_M(0,:,:)
  res = True
  res@cnFillOn = True
  res@gsnSpreadColors = True
  res@cnLevelSelectionMode = \
     "ManualLevels"
  res@cnMinLevelValF = 0.
  res@cnMaxLevelValF = 1000.
  res@cnLevelSpacingF = 50.
  plot = gsn_contour(wks,ter,res)
end
```



Basic NCL resources - *there are over 1400 controlling contours, labelbars, legends, maps, etc.*

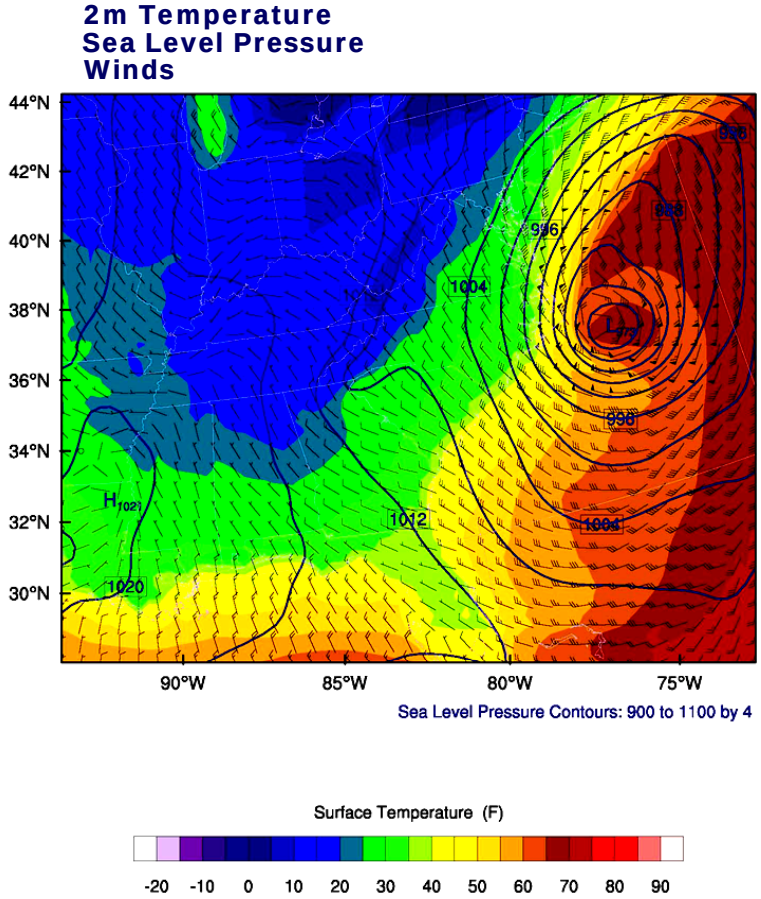
Generate Plots

```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/wrf/WRFUserARW.ncl"

begin
  a = addfile("./geo_em.d01.nc","r")
  wks = gsn_open_wks("pdf","plt_ter5")
  opts = True
  opts@MainTitle = "GEOGRID FIELDS"


  ter = wrf_user_getvar(a,"HGT_M",0)
  res = opts
  res@cnFillOn = True
  res@ContourParameters = (/0.,1000.,50./)
  contour = wrf_contour(a,wks,ter,res)
  pltres = True
  mpres = True
  plot = wrf_map_overlays(a,wks,(/contour/),\
    pltres,mpres)
end
```



Generate Plots

```
slp = wrf_user_getvar(a,"slp",5)
t2  = wrf_user_getvar(a,"T2",5)
u10 = wrf_user_getvar(a,"U10",5)
v10 = wrf_user_getvar(a,"V10",5)

os@cnLineColor = "NavyBlue"
c_slp = wrf_contour(a,wks,slp,os)
ot@cnFillOn = True
c_tc = wrf_contour(a,wks,t2,ot)
ov@NumVectors = 47
vec = wrf_vector(a,wks,u10,v10,ov)

plot = wrf_map_overlays(a, wks, \
      (/c_tc,c_slp,vec/), pltres, mpres)
```

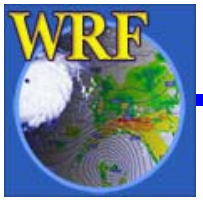


Special WRF Functions

- wrf_user_getvar
  Get fields from input file

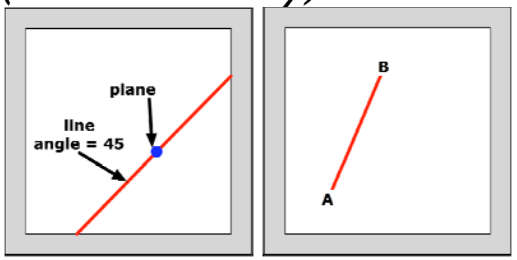
  ter = wrf_user_getvar(a,"HGT",0)
  t2 = wrf_user_getvar(a,"T2",-1)
  slp = wrf_user_getvar(a,"slp",1)

  avo/pvo: Absolute/Potential Vorticity,
  cape_2d: 2D mcape/mcin/lcl/lfc, *cape_3d*: 3D cape/cin,
  dbz/mdbz: Reflectivity, *geopt/geopotential*: Geopotential,
  p/pres/pressure: Pressure,*rh/rh2*: Relative Humidity,
  slp: Sea Level Pressure, *td/td2*: Dew Point Temperature,
  tc/tk: Temperature, *th/theta*: Potential Temperature,
  z/height: Height, *ua/va/wa*: wind on mass points,
  uvmet/uvmet10: wind rotated to earth coordinates



Special WRF Functions

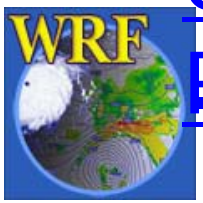
- **wrf_contour / wrf_vector**
  Create line/shaded & vector plots
- **wrf_map_overlays / wrf_overlays**
  Overlay plots created with wrf_contour and wrf_vector
- **wrf_map**
  Create a map background – not used often
- **wrf_user_intrp3d / wrf_user_intrp2d**
  Interpolate horizontally to a given pressure or height level (3d data only)
  Interpolate vertically along a given line
- **wrf_user_ll_to_ij / wrf_user_ij_to_ll**
  Convert: lat/lon ⟷ ij
- **wrf_user_list_times**
  Get list if times available in input file
- **wrf_user_unstagger**
  Unstaggers an array



Resources

- The special WRF functions have unique resources: http://www.mmm.ucar.edu/wrf/OnLineTutorial/Graphics/NCL/NCL_functions.htm

- All general NCL resources can also be used to control the plot: http://www.ncl.ucar.edu/Document/Graphics/Resources

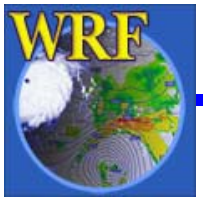


Calling FORTRAN code from NCL

- Easier to use F77 code, but works with F90 code
- Need to isolate definition of input variables and wrap it with special comment statements:

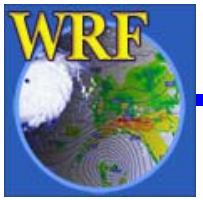
```
C NCLFORTSTART
C NCLEND
```

- Use a tool called `WRAPIT` to create a `*.so` file
- Load `*.so` file in NCL script with `"external"` statement
- Call Fortran function with special `"::"` syntax
- Must preallocate arrays!



myTK.f

```
C NCLFORTSTART
      subroutine compute_tk (tk,pressure,theta, nx, ny, nz)
        implicit none
        integer nx,ny,nz
        real    pi, tk(nx,ny,nz)
        real    pressure(nx,ny,nz), theta(nx,ny,nz)
C NCLEND
      integer i,j,k

        do k=1,nz
          do j=1,ny
          do i=1,nx
             pi=(pressure(i,j,k) / 1000.)**(287./1004.)
             tk(i,j,k) = pi*theta(i,j,k)
          enddo
          enddo
        enddo

      end
```

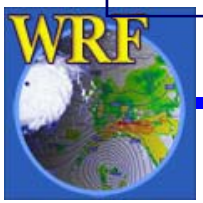


myTK.so – Create & use in NCL script

```
% WRAPIT myTK.f
```

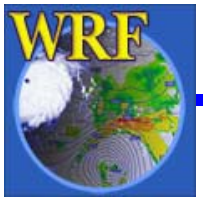
This will create a "myTK.so" file

```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
load "$NCARG_ROOT/lib/ncarg/nclscripts/wrf/WRFUserARW.ncl"
external myTK "./myTK.so"

begin
    t = wrf_user_getvar(a,"T",5)
    t = t + 300
    p = wrf_user_getvar(a,"pressure",5)

    ; Must preallocate space for output arrays
    dim = dimsizes(t)
    tk = new( dimsizes(t), typeof(t) )

    ; Remember, Fortran/NCL arrays are ordered differently
        myTK :: compute_tk (tk,p,t,dim(2),dim(1),dim(0))
end
```



FORTRAN 90 code

- Can use simple FORTRAN 90 code
- Your FORTRAN 90 program may not contain any of the following features:
  - pointers or structures as arguments,
  - missing/optional arguments,
  - keyword arguments, or
  - recursive procedure.



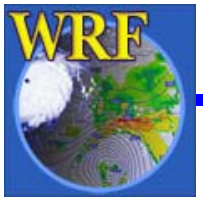
Utilities

UG: Chapter 3 & 10

- Graphics
- Designing a model domain
- Data

grib 1&2	intermediate format	netcdf
netcdf		

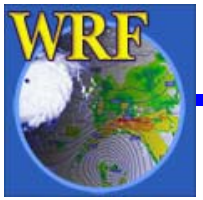
- netCDF tools
- Other utilities



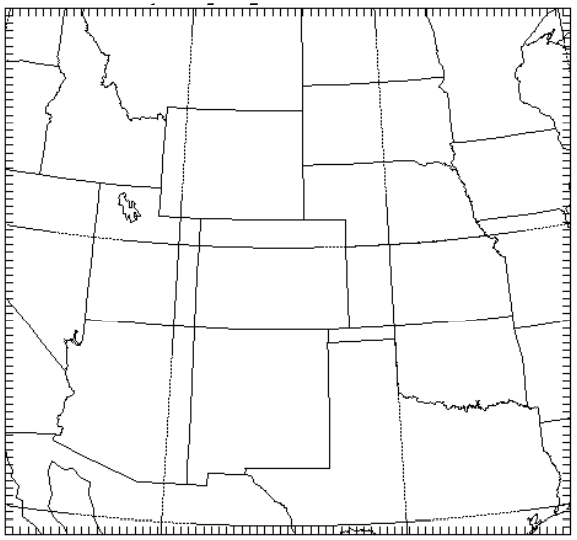
Graphics : ImageMagick

- Convert graphical files from one format to another
  - Many options available (*rotate* frames, *trim* white space, etc.)
  - Can be used for files with single or multiple frames
  - Cannot deal with **.ncgm** files
  - http://www.imagemagick.org

convert *file*.pdf *file*.png
convert *file*.png *file*.bmp
convert *file*.pdf *file*.gif
convert *file*.ras *file*.png



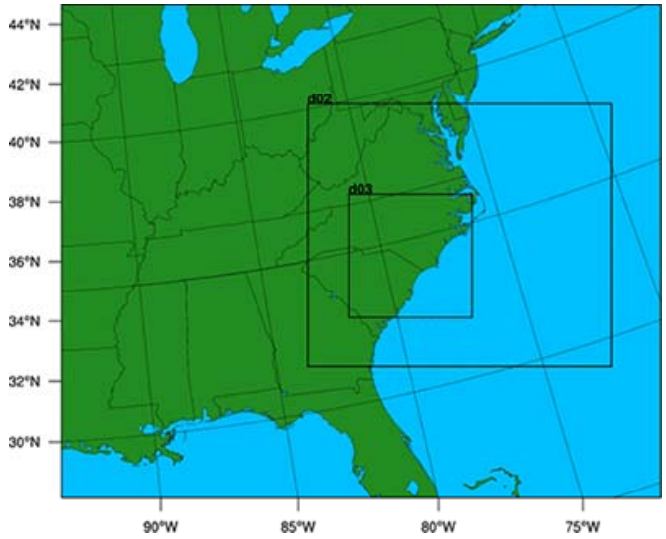
WRF Model Domain Design

- *WPS util/ directory*
  - ***plotgrids.exe***
  - *reads namelist information to generate plot*
  - *create an NCAR Graphics file called 'gmeta'*
  - 

- *NCL*

mp =
wrf_wps_dom (wks, mpres, lnres, txres)

```
mpres@max_dom   = 3
mpres@ref_lat   = 34.83
mpres@ref_lon   = -81.03
mpres@dx        = 30000.
mpres@dy        = 30000.
mpres@map_proj  = "lambert"
```



GRIB

- *http://dss.ucar.edu/docs/formats/grib/gribdoc/*

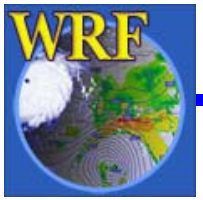
- g1print.exe & g2print.exe
  - Show data available in GRIB1 & GRIB2 files
  - Available from **util**/ directory in WPS

- grib2ctl.pl
  - Create .ctl and .idx files, so one can plot GRIB files with GrADS *(available on web)*

- wgrib *(for GRIB 1 data files)*

  wgrib –v file

  wgrib –V file
  http://www.cpc.ncep.noaa.gov/products/wesley/wgrib.html



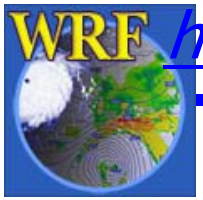
GRIB2

- Documentation
  http://www.nco.ncep.noaa.gov/pmb/docs/grib2/grib2_doc.shtml

- GRIB2 – GRIB1 parameter conversion table
  http://www.nco.ncep.noaa.gov/pmb/docs/grib2/GRIB2_parmeter_conversion_table.html

Product	Category	Parameter		Parameter
0	2	2	U	33
0	2	3	V	34

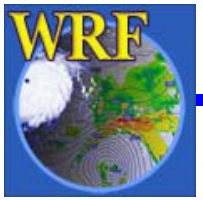
- wgrib2
  http://www.cpc.ncep.noaa.gov/products/wesley/wgrib2/



Utility: g1print and g2print

The *g1print* and *g2print* programs list the contents of a GRIB1 or GRIB2 file:

rec num	Prod Disc	Cat	Param num	Lvl code	Lvl one	Lvl two	Name	Time	Fcst hour
1	0	3	5	100	100000	0	HGT	2006-08-16_12:00:00	00
2	0	3	5	100	97500	0	HGT	2006-08-16_12:00:00	00
3	0	3	5	100	95000	0	HGT	2006-08-16_12:00:00	00
4	0	3	5	100	92500	0	HGT	2006-08-16_12:00:00	00
5	0	3	5	100	90000	0	HGT	2006-08-16_12:00:00	00
6	0	3	5	100	85000	0	HGT	2006-08-16_12:00:00	00
7	0	3	5	100	80000	0	HGT	2006-08-16_12:00:00	00
8	0	3	5	100	75000	0	HGT	2006-08-16_12:00:00	00
9	0	3	5	100	70000	0	HGT	2006-08-16_12:00:00	00
10	0	3	5	100	65000	0	HGT	2006-08-16_12:00:00	00

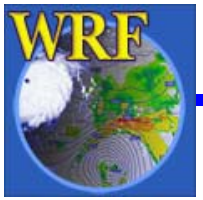


WPS Intermediate Files

- Output format of ungrid

- WPS util/ directory
  - **plotfmt.exe** (*graphical interface to view intermediate file)*
  - ***rd_intermediate.exe***

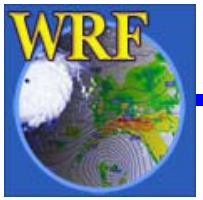
- Create your own intermediate files
  - *example if you have input data in netCDF format*
  - ***http://www.mmm.ucar.edu/wrf/OnLineTutorial/WPS/IM_files.htm***



Utility: rd_intermediate

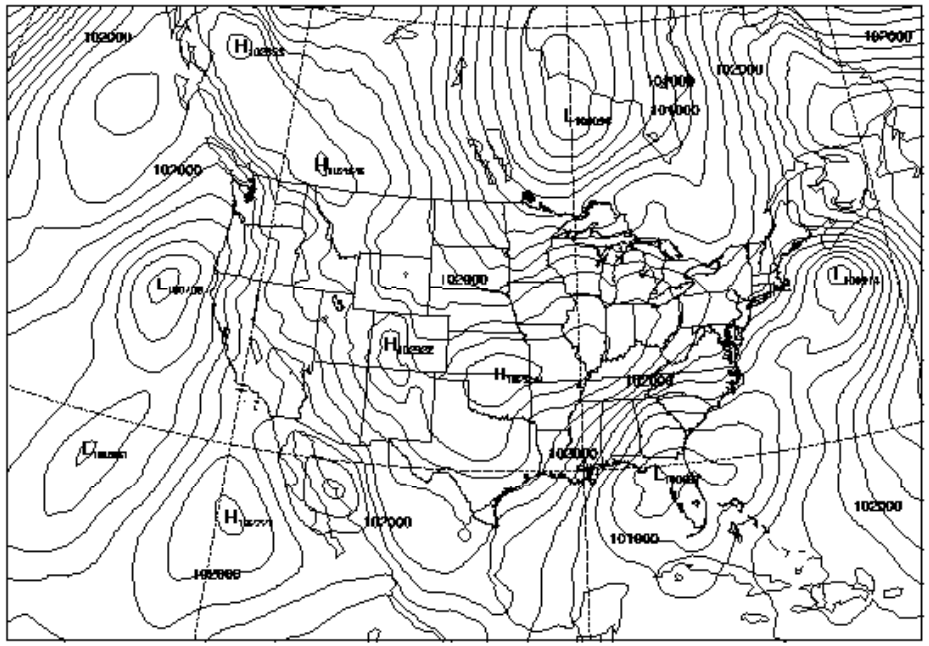
The rd_intermediate lists information about the fields found in an intermediate-format file

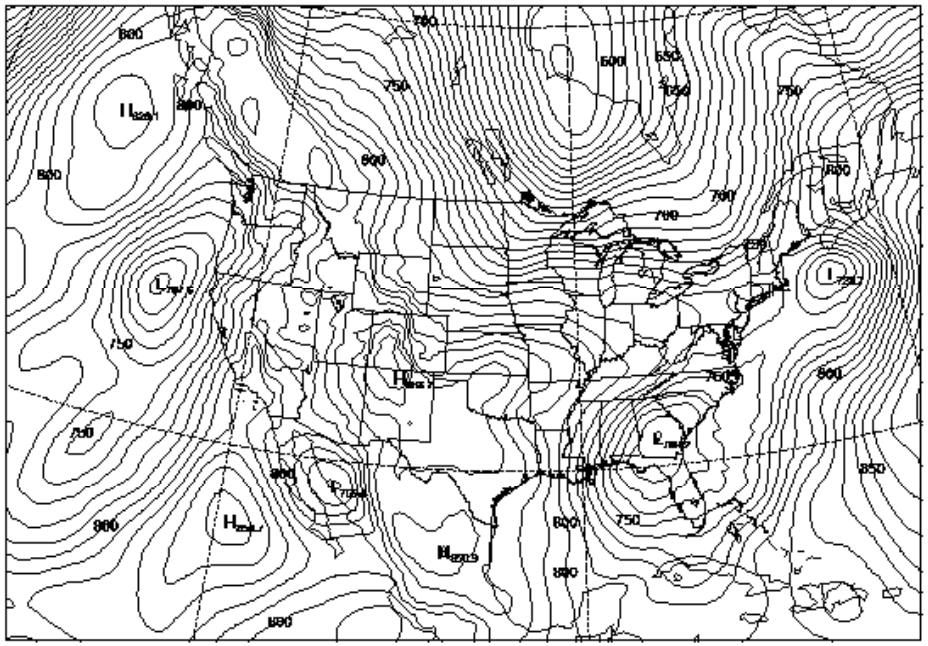
```
================================================
FIELD = TT
UNITS = K DESCRIPTION = Temperature
DATE = 2000-01-24_12:00:00 FCST = 0.000000
SOURCE = unknown model from NCEP GRID 212
LEVEL = 200100.000000
I,J DIMS = 185, 129
IPROJ = 1
  REF_X, REF_Y = 1.000000, 1.000000
  REF_LAT, REF_LON = 12.190000, -133.459000
  DX, DY = 40.635250, 40.635250
  TRUELAT1 = 25.000002
DATA(1,1)=295.910950
================================================
```



Utility: plotfmt

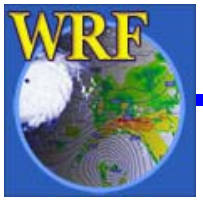
The plotfmt program plots the fields in the ungrib intermediate-formatted files





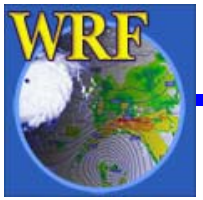
netCDF

- netCDF stands for *network Common Data Form*
- netCDF is one of the current supported data formats chosen for WRF I/O API
  - WRF I/O supports netCDF *(not fully CF compliant - `climate and Forecast Metadata Convention`)*/ binary/GRIB/HDF
  - Most support graphical packages currently only support netCDF file format
- *http://www.unidata.ucar.edu* (***documentation***)
- http://www.unidata.ucar.edu/software/netcdf/fguide.pdf ***(writing Fortran programs to read/write netCDF files)***



netCDF

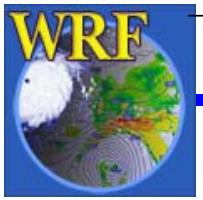
- Advantages of using netCDF?
  - Platform-independent (*big_endian / little_endian*)
  - A lot of software already exist which can be used to process netCDF data
- netCDF operators
  - http://nco.sourceforge.net/
  - Stand alone programs to, which can be used to manipulate data *(performing grid point averaging / file differencing / file 'appending')*



NCO tools

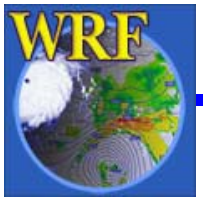
http://nco.sourceforge.net/

- ncdiff
  - Difference two file
    ncdiff input1.nc input2.nc output.nc
- ncrcat *(nc cat)*
  - Write specified variables / times to a new file
    ncrcat -v RAINNC wrfout RAINNC.nc*
    ncrcat -d Time,0,231 -v RAINNC wrfout RAINNC.nc*
- ncra *(nc average)*
  - Average variables and write to a new file
    ncra -v OLR wrfout OLR.nc*
- ncks *(nc kitchen sink)*
  - Combination of NCO tools all in one *(handy: one tool for multiple operations)*



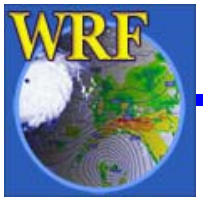
Change fields in netCDF file

```
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"

Begin

  a = addfile("./met_em.d01.2000-01-24_12:00:00.nc",”w")

  sst = a->SST      ; read a field
  sst = sst + 10    ; change the field
  a->SST = sst      ; write the field

end
```



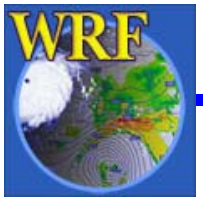
netCDF : Utilities

- ncdump
  - reads a netCDF dataset and prints information from the dataset
  - ncdump –h *file*
    print header (inc. list of variables in the file)
  - ncdump –v *VAR file*
    print data of the variable VAR
    ncdump –v Times *file*



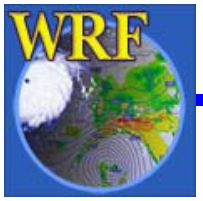
netCDF : *ncdump -h*

```
netcdf wrfinput_d01 {
dimensions:
        Time = UNLIMITED ; // (1 currently)
        DateStrLen = 19 ;
        west_east = 73 ;
        south_north = 60 ;
        west_east_stag = 74 ;
        bottom_top = 27 ;
        south_north_stag = 61 ;
        bottom_top_stag = 28 ;
variables:
        char Times(Time, DateStrLen) ;
        float LU_INDEX(Time, south_north, west_east) ;
                LU_INDEX:FieldType = 104 ;
                LU_INDEX:MemoryOrder = "XY " ;
         LU_INDEX:description = "LAND USE CATEGORY" ;
                LU_INDEX:units = "" ;
                LU_INDEX:stagger = "" ;
……
……
……

global attributes:
                :TITLE = " OUTPUT FROM REAL_EM V3.0 PREPROCESSOR";
                :START_DATE = "2000-01-24_12:00:00" ;
                :SIMULATION_START_DATE = "2000-01-24_12:00:00" ;
                :WEST-EAST_GRID_DIMENSION = 74 ;
                :SOUTH-NORTH_GRID_DIMENSION = 61 ;
                :BOTTOM-TOP_GRID_DIMENSION = 28 ;
                :DX = 30000.f ;
                :DY = 30000.f ;
                :GRIDTYPE = "C" ; :MP_PHYSICS = 3 ;
                :SF_SFCLAY_PHYSICS = 1 ;
                :SF_SURFACE_PHYSICS = 1 ;
                :BL_PBL_PHYSICS = 1 ;
                :CU_PHYSICS = 1 ;
……
……
……
```



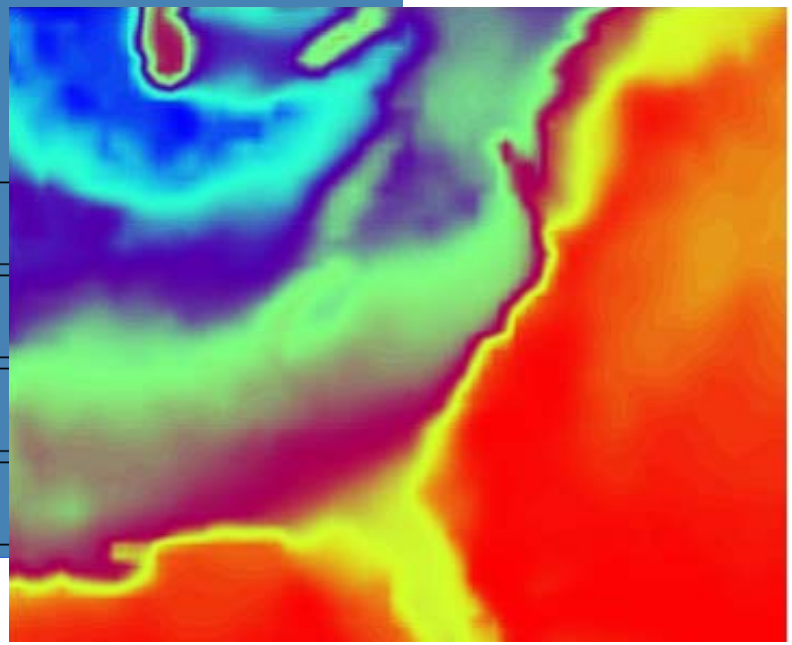
netCDF : *ncdump -v Times*

```
netcdf wrfinput_d01 {
dimensions:
        Time = UNLIMITED ; // (1 currently)
        DateStrLen = 19 ;
        west_east = 73 ;
        south_north = 60 ;
        west_east_stag = 74 ;
        bottom_top = 27 ;
        south_north_stag = 61 ;
        bottom_top_stag = 28 ;
variables:
        char Times(Time, DateStrLen) ;
        float LU_INDEX(Time, south_north, west_east) ;
                LU_INDEX:FieldType = 104 ;
                LU_INDEX:MemoryOrder = "XY " ;
         LU_INDEX:description = "LAND USE CATEGORY" ;
                LU_INDEX:units = "" ;
                LU_INDEX:stagger = "" ;
……
……

global attributes:
                :TITLE = " OUTPUT FROM REAL_EM V3.0 PREPROCESSOR";
                :START_DATE = "2000-01-24_12:00:00" ;
                :SIMULATION_START_DATE = "2000-01-24_12:00:00" ;
                :WEST-EAST_GRID_DIMENSION = 74 ;
                :SOUTH-NORTH_GRID_DIMENSION = 61 ;
                :BOTTOM-TOP_GRID_DIMENSION = 28 ;
                :DX = 30000.f ;
                :DY = 30000.f ;
……
……

data:

 Times =
  "2000-01-24_12:00:00",
  "2000-01-24_18:00:00",
  "2000-01-25_00:00:00",
  "2000-01-25_06:00:00"
```



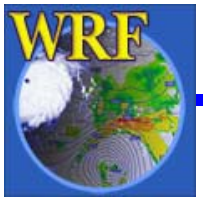
ncview

http://meteora.ucsd.edu/~pierce/ncview_home_page.html



Other Utilities

- Developed / Supported by NCAR
- FORTRAN program
  - Easy to use
  - Easy to add your own code
  - Only for netCDF datasets
- http://www.mmm.ucar.edu/wrf/users/utilities/util.htm



Other Utilities

- **read_wrf_nc**
  - Display data inside a wrfout netCDF file
  - Specific points; min/max of fields; time series; edit data in file
- **iowrf**
  - Thinning of netCDF data; extracting a area; destaggering grid
- **p_interp**
  - Interpolate to pressure levels
- **v_interp**
  - Add vertical levels in wrf input and boundary files
  - For use with ndown
- **wrf_to_cf**
  - User contributed code *(not fully supported)*