Analysis and Display Meteorological Data using MeteoInfo

Yaqiang Wang

Chinese Academy of Meteorological Sciences

yaqiang.wang@gmail.com

MeteoInfo software package

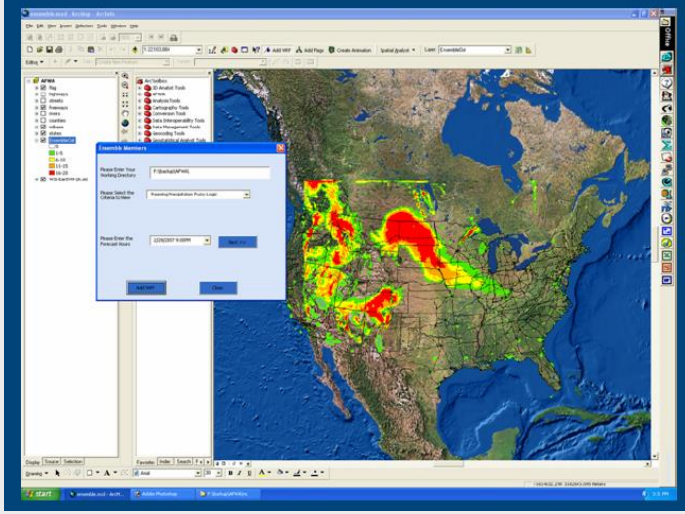
- Motivation: Scientific data analysis and visualization – especially for meteorological community
- MeteoInfo library: Java and C# version
- MeteoInfo GIS desktop: Java and C# version
- MeteoInfoLab: Java and Jython

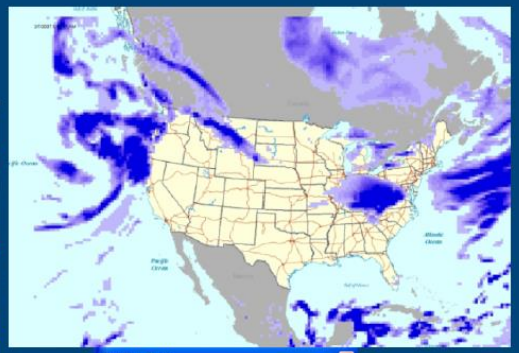
Commercial GIS software

- Very expensive
- Complex in operation
- Not support popular meteorological data formats conveniently

NetCDF in ArcGIS

- **Atmospheric community – ESRI collaboration**
- **Since 2006 release of ArcGIS 9.2 NetCDF CF format can be read in GIS**

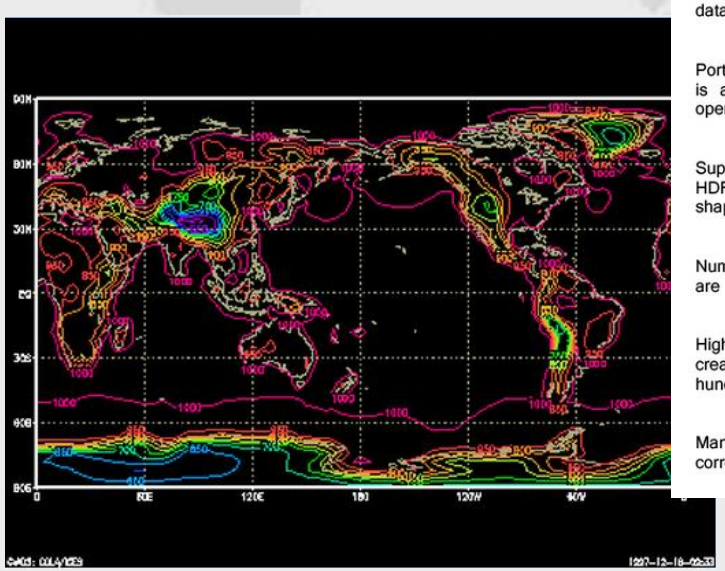


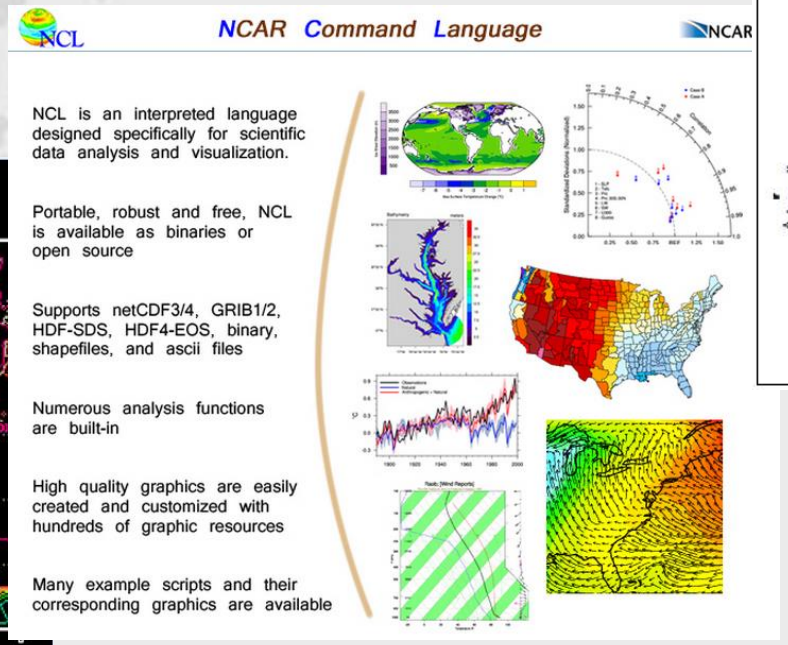


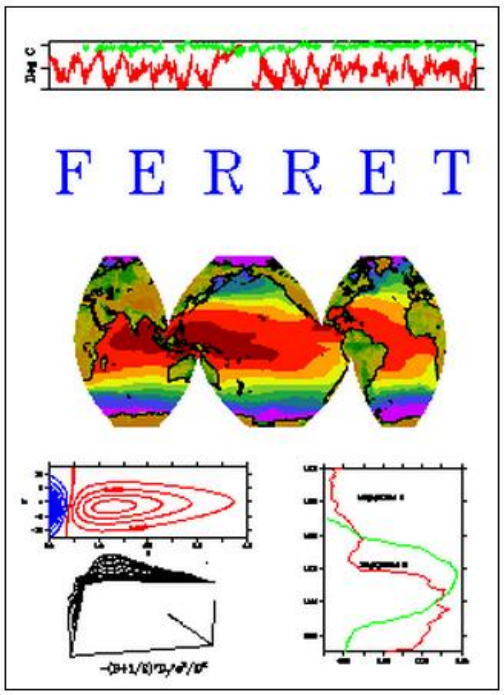
Visualization tools in atmospheric science community

GrADS, NCL, Ferret…

- Command-line interface
- Not closely linked with GIS community

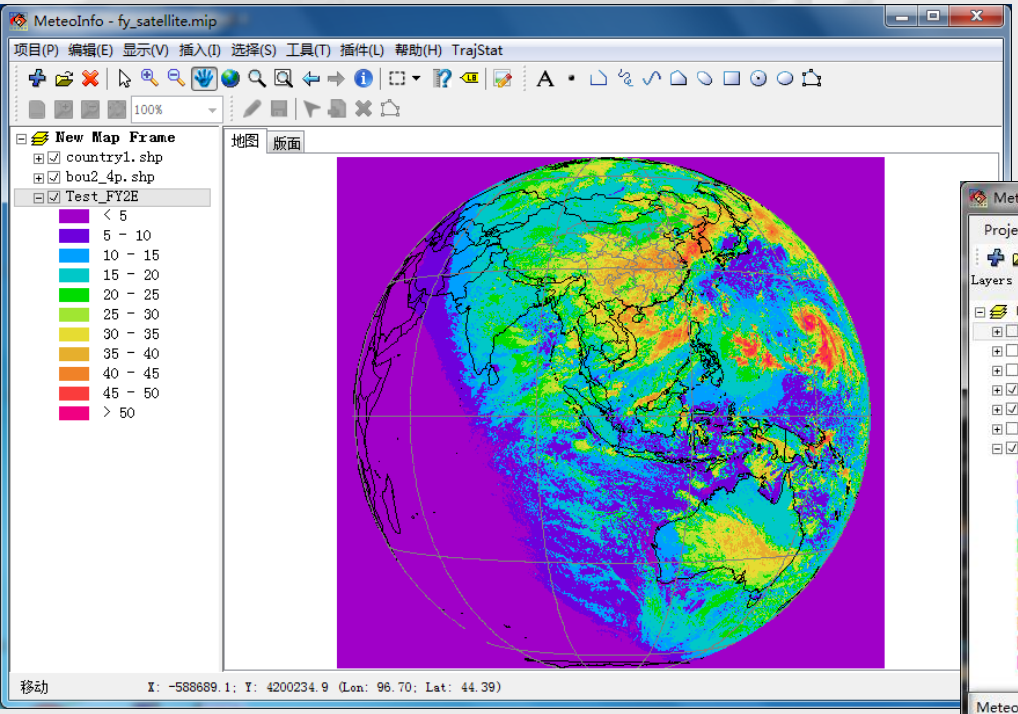


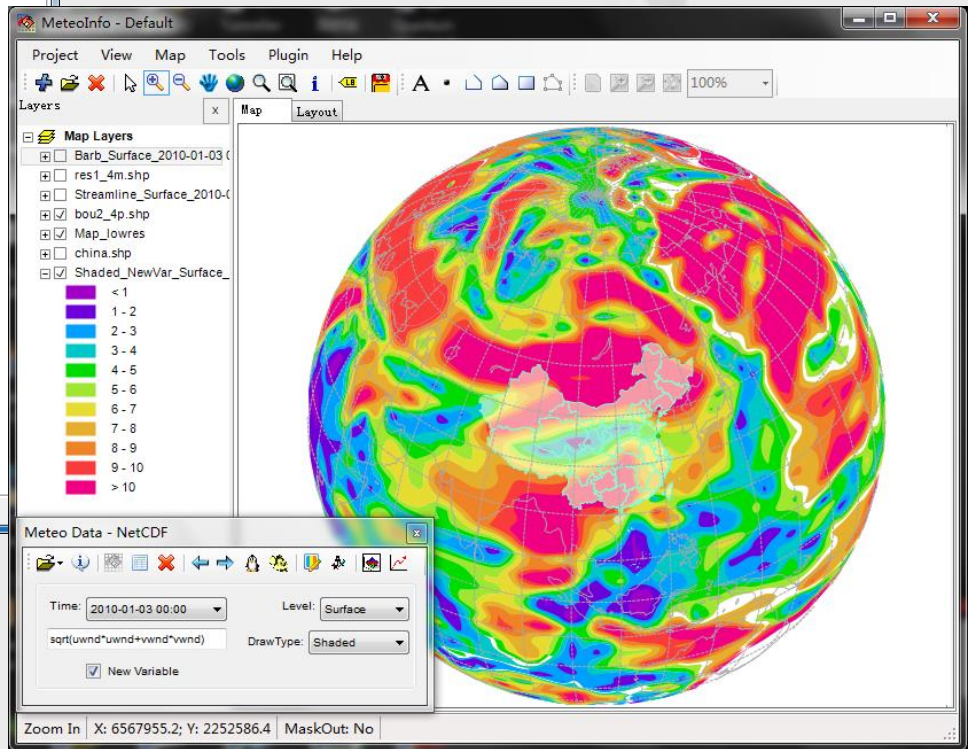




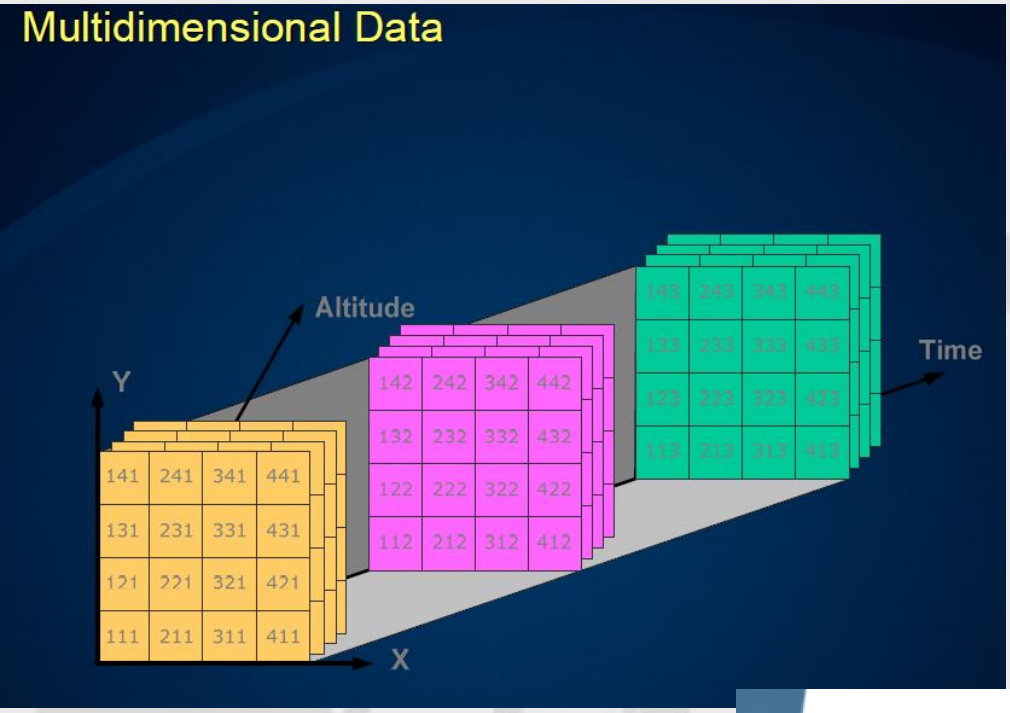
MeteoInfo Development

- GIS functions were developed from ground level.
- Two editions: Java and C#.
- MeteInfo could be downloaded freely from the website http://www.meteothinker.com
- Cross-platform: Windows, Unix, Linux, Mac OS.
- Export as EPS, PDF, EMF, PNG, GIF, JPEG …





Wang 2014 Meteorological Applications

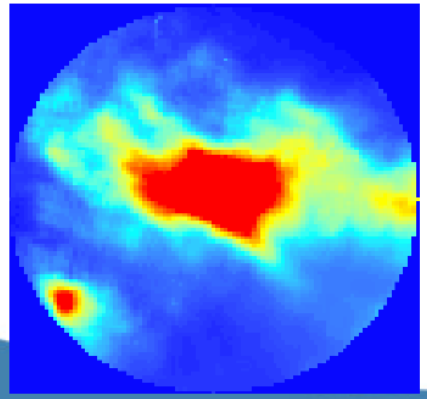


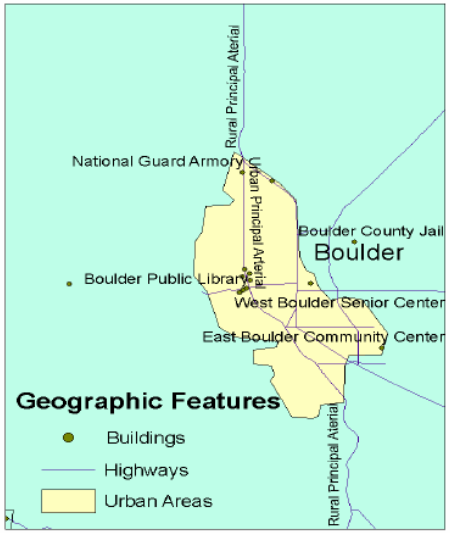
Data model

4D(meteorology) -> 2D, 2.5D (GIS)

To develop a comprehensive software tool which has the ability to handle both 4D meteorological data and traditional vector and raster GIS data.

- **Vector formats**
  - Discrete representation of reality (points, lines, polygons)
- **Raster formats**
  - Use square cells to model reality

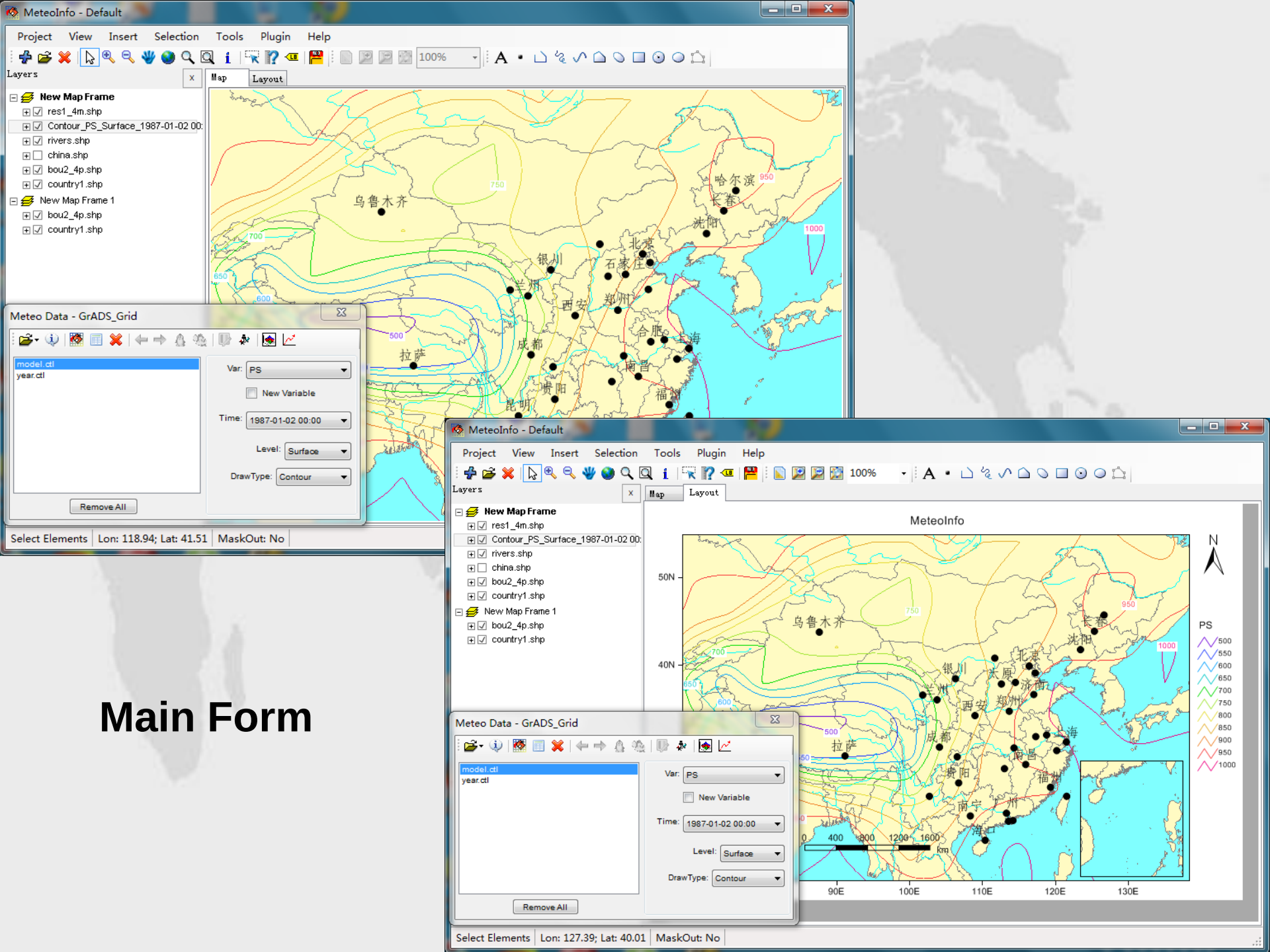




Supported data formats

Data format	Data type	File type	Decoding difficulty level	Development by	Users
NetCDF	Grid	Binary	Difficult, need API library	Unidata	Widely used, some conventions are used by atmospheric community
GRIB 1 and 2	Grid	Binary	Difficult	WMO	Atmospheric community
GrADS binary	Grid and station	Binary	Moderate	IGES	Atmospheric community
HDF EOS5	Grid and swath	Binary	Difficult, need API library	HDF group and NASA	Satellite community
ARL packed	Grid	Binary	Moderate	NOAA ARL	ARL model users
HYSPLIT output	Grid, station and trajectory	Binary (grid and station) or text (trajectory)	Easy	NOAA ARL	HYSPLIT model users
MICAPS	Grid, station and trajectory	Most are text	Easy	CMA	Atmospheric community in China
SYNOP	Station	Text	Moderate	WMO	Atmospheric community
METAR	Station	Text	Moderate	ICAO	Aviation and atmospheric communities
NOAA ISH	Station	Text	Moderate	NOAA	Atmospheric community
Longitude/latitude station	Station	Text	Easy	MeteoInfo	MeteoInfo users
ESRI ASCII grid	Grid	Text	Easy	ESRI	ArcGIS users
Surfer ASCII grid	Grid	Text	Easy	Golden Software	Surfer users

MeteoInfo GIS Desktop Application


MeteoInfo - Default
Project View Insert Selection Tools Plugin Help
Layers
Map Layout
New Map Frame
res1_4m.shp
Contour_PS_Surface_1987-01-02 00:
rivers.shp
china.shp
bou2_4p.shp
country1.shp
New Map Frame 1
bou2_4p.shp
country1.shp
Meteo Data - GrADS_Grid
model.ctl
year.ctl
Var: PS
New Variable
Time: 1987-01-02 00:00
Level: Surface
DrawType: Contour
Remove All
Select Elements
Lon: 118.94; Lat: 41.51
MaskOut: No
MeteoInfo - Default
Lon: 127.39; Lat: 40.01
MeteoInfo
PS
50N
40N
90E
100E
110E
120E
130E

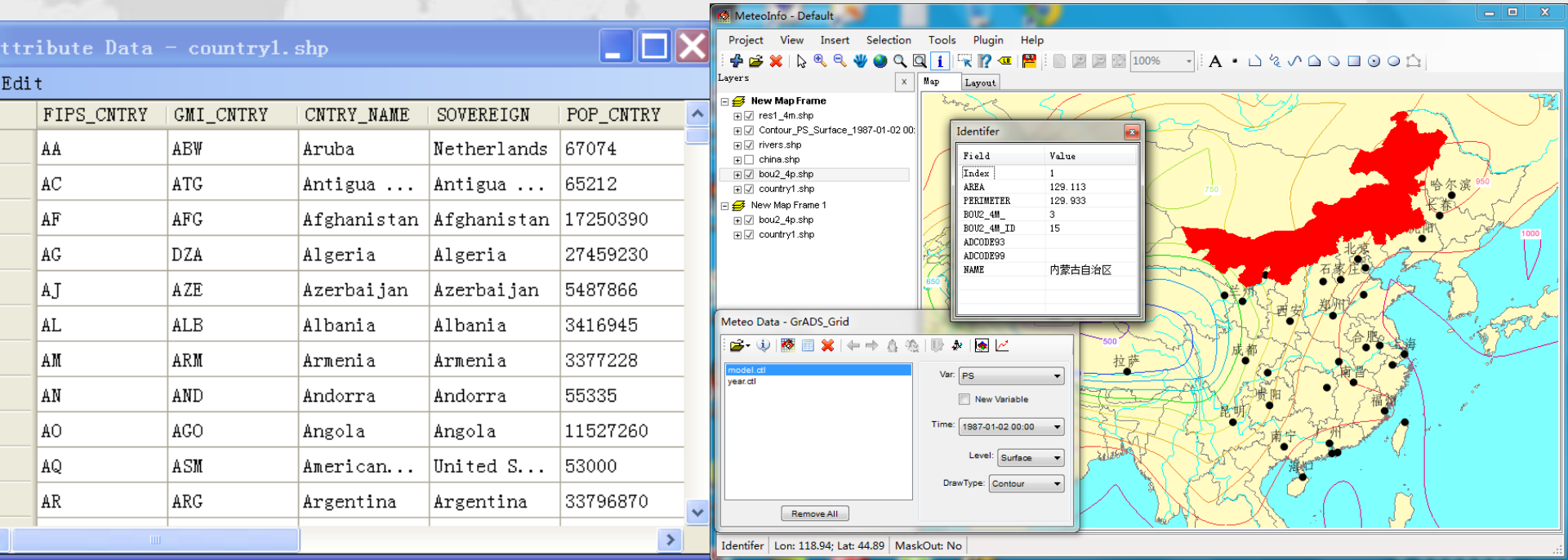
Main Form

Supported Map and Image Data

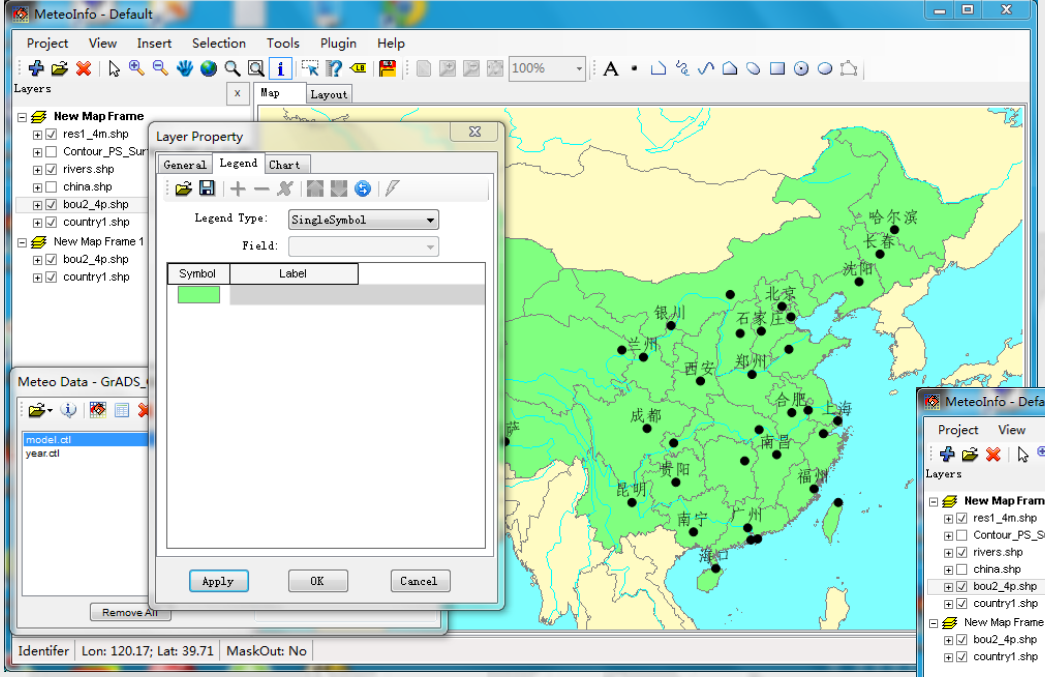
- ESRI shape file with point, polyline or polygon shape type. Projected shape file was not supported at present.
- Geographic map data of GrADS.
- Geographic map data of MICAPS.
- wmf map data. (Can be created by ‘Output Map Data’ function in MeteoInfo)
- Normal Image data: Bmp, Jpg, Gif, Tiff, Png

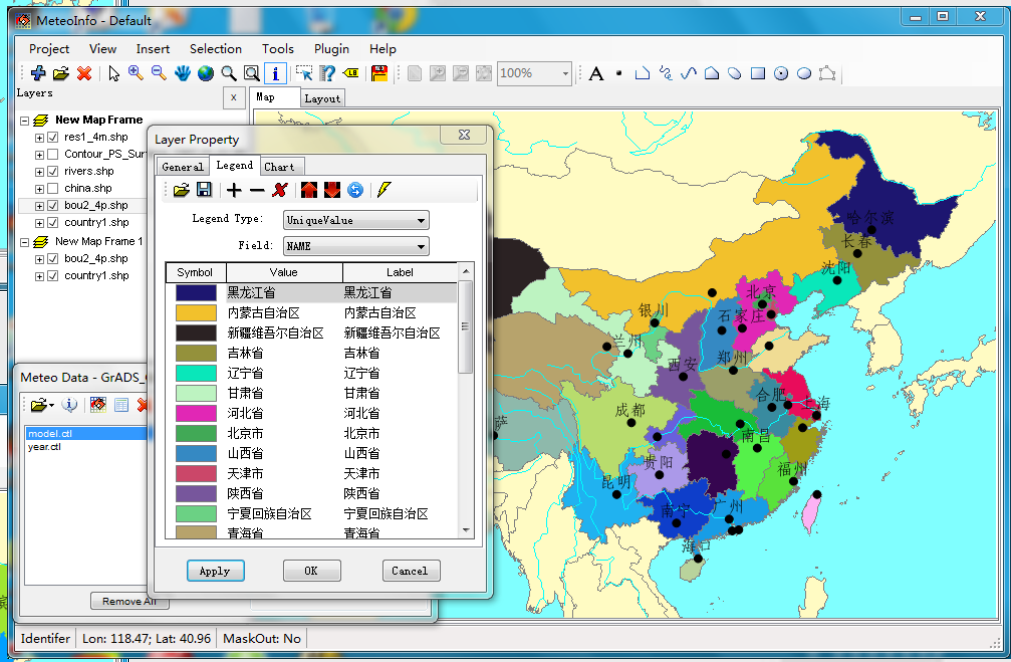
Layers and Project file

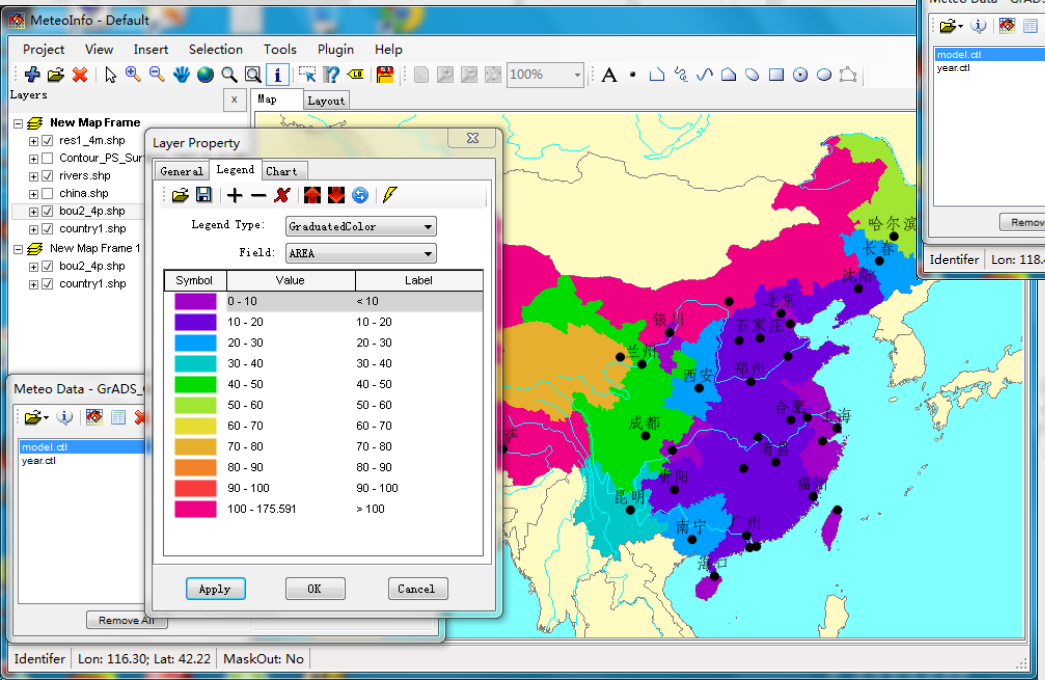
- All data were viewed as layers
- Vector layer (Point, Polyline, Polygon), has attribute data
- Image layer (geo-location setting, world file)
- Raster layer (Grid data)
- All layers saved as files and the setting of MapView and Layout could be saved as a project file (.mip).
- The layer created from meteorological data is only exist in memory. It can be saved as a shape file and then could be saved as a layer in the project file.

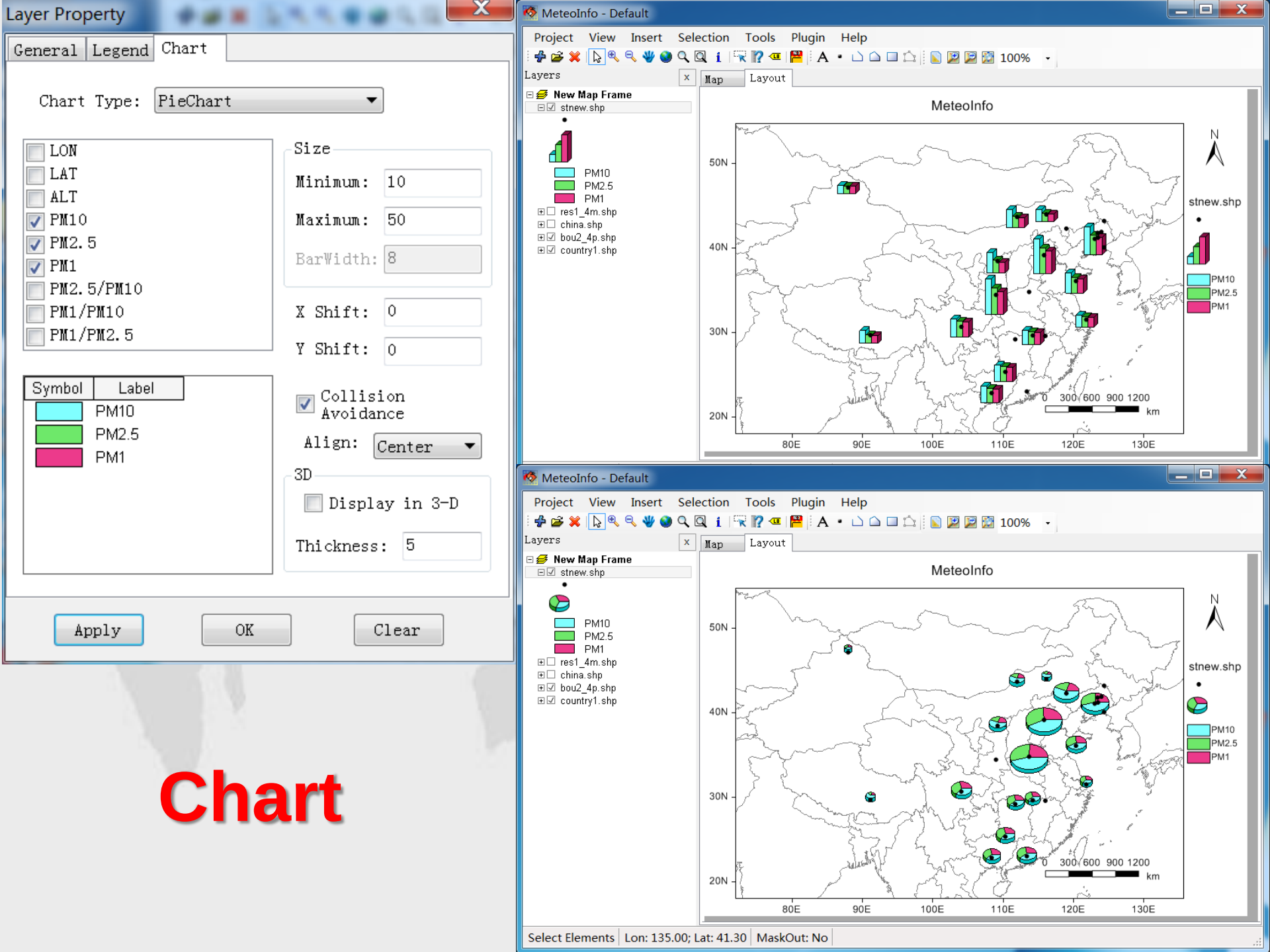


Legend scheme

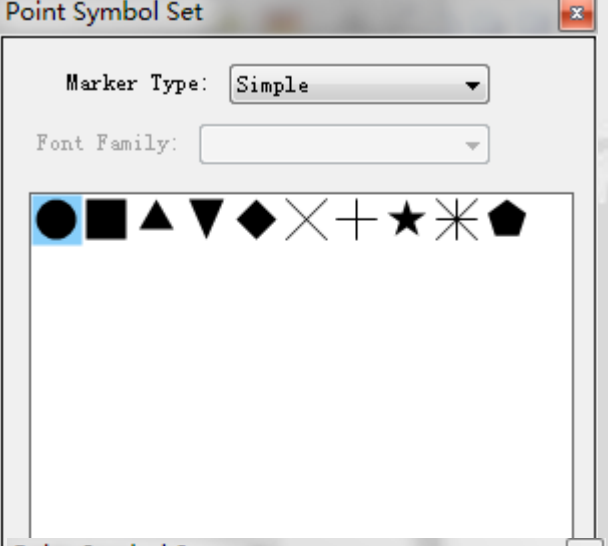


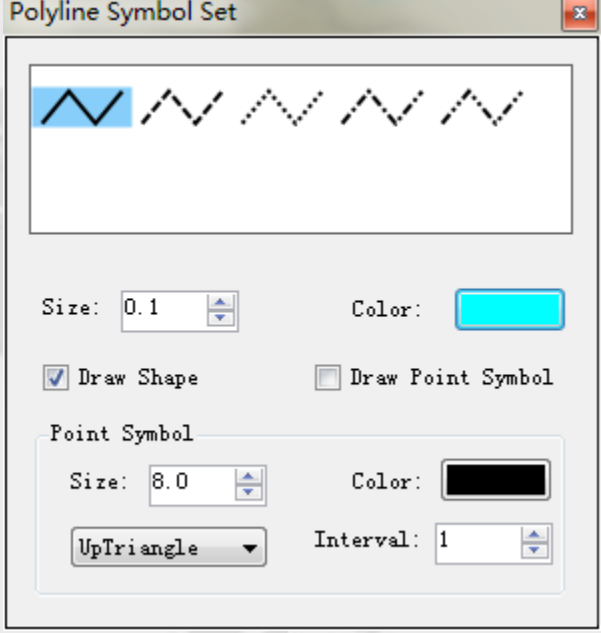


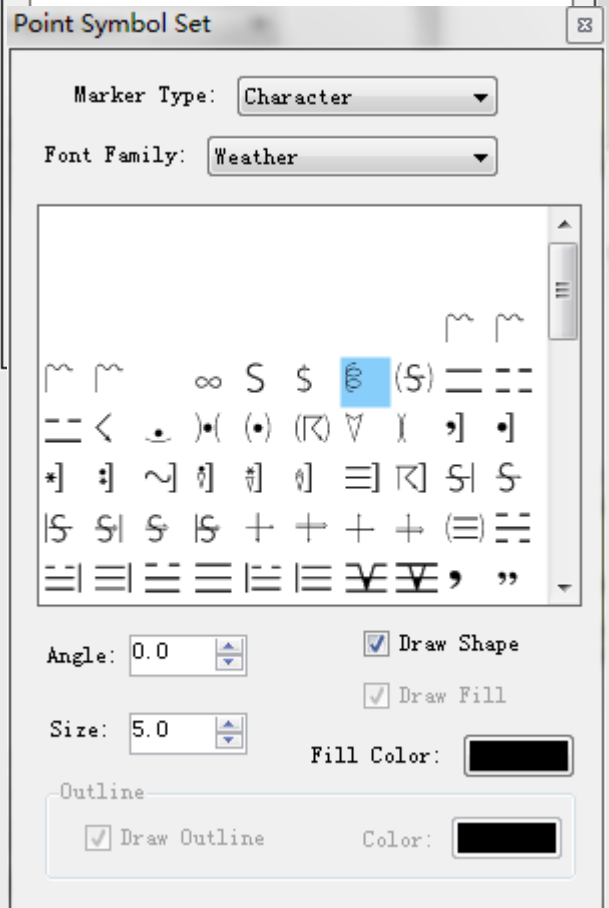


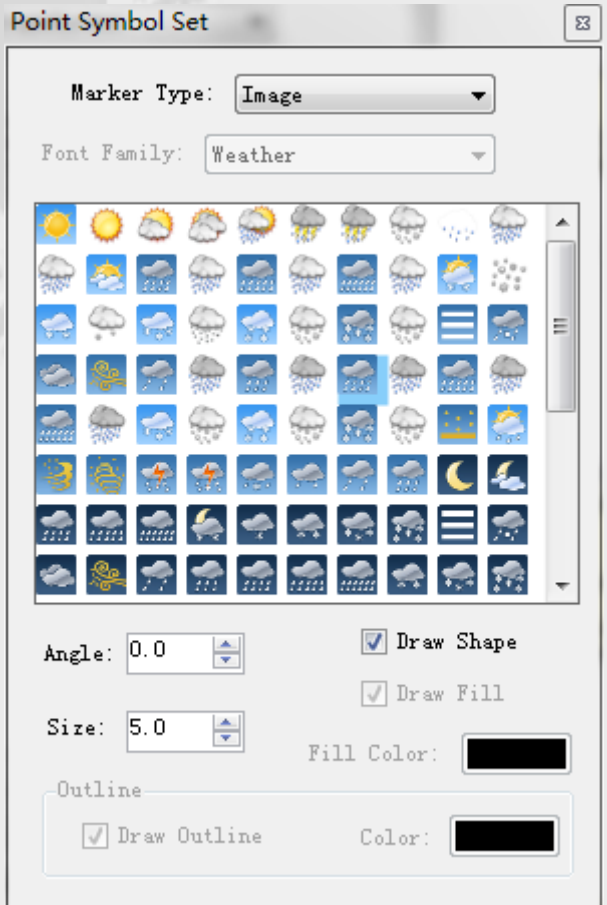

Layer Property
General
Legend
Chart
Chart Type: PieChart
LON
LAT
ALT
PM10
PM2.5
PM1
PM2.5/PM10
PM1/PM10
PM1/PM2.5
Size
Minimum: 10
Maximum: 50
BarWidth: 8
X Shift: 0
Y Shift: 0
Symbol
Label
PM10
PM2.5
PM1
Collision Avoidance
Align: Center
3D
Display in 3-D
Thickness: 5
Apply
OK
Clear
MeteoInfo - Default
Project
View
Insert
Selection
Tools
Plugin
Help
Layers
Map
Layout
New Map Frame
stnew.shp
PM10
PM2.5
PM1
res1_4m.shp
china.shp
bou2_4p.shp
country1.shp
MeteoInfo
50N
40N
30N
20N
80E
90E
100E
110E
120E
130E
N
0 300 600 900 1200
km
Select Elements
Lon: 135.00; Lat: 41.30
MaskOut: No
Chart

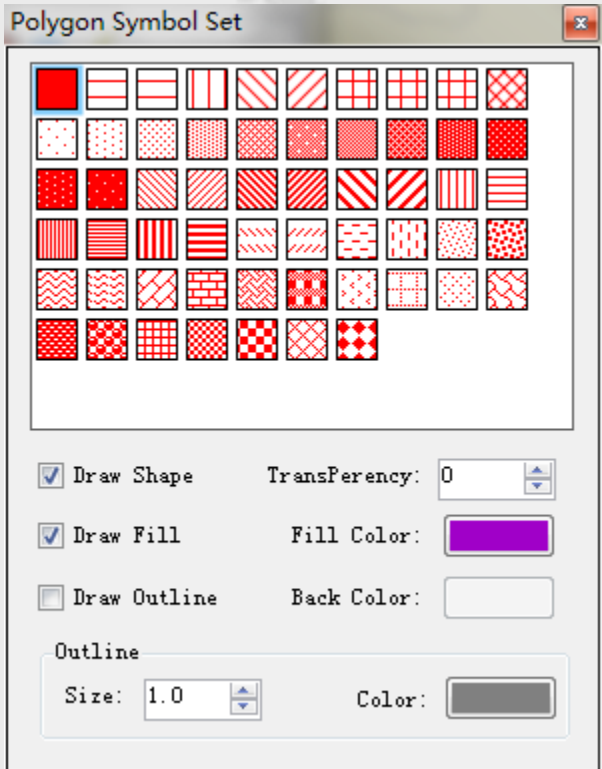
Symbol Set

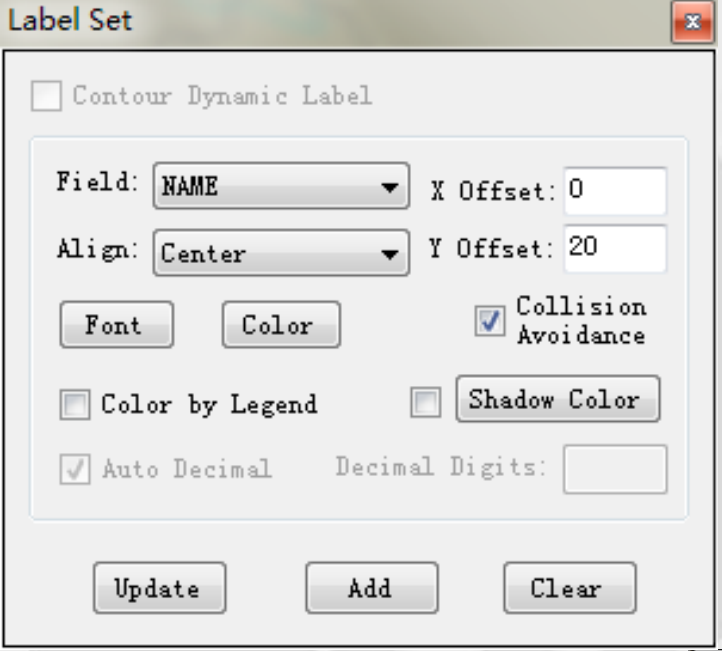

Point Symbol Set
Marker Type: Simple
Font Family:


Polyline Symbol Set
Size: 0.1
Color:
Draw Shape
Draw Point Symbol
Point Symbol
Size: 8.0
Color:
UpTriangle
Interval: 1


Point Symbol Set
Marker Type: Character
Font Family: Weather
Angle: 0.0
Draw Shape
Draw Fill
Size: 5.0
Fill Color:
Outline
Draw Outline
Color:


Point Symbol Set
Marker Type: Image
Font Family: Weather
Angle: 0.0
Draw Shape
Draw Fill
Size: 5.0
Fill Color:
Outline
Draw Outline
Color:


Polygon Symbol Set
Draw Shape
TransPerency: 0
Draw Fill
Fill Color:
Draw Outline
Back Color:
Outline
Size: 1.0
Color:



Label the shapes by the attribute data of a field

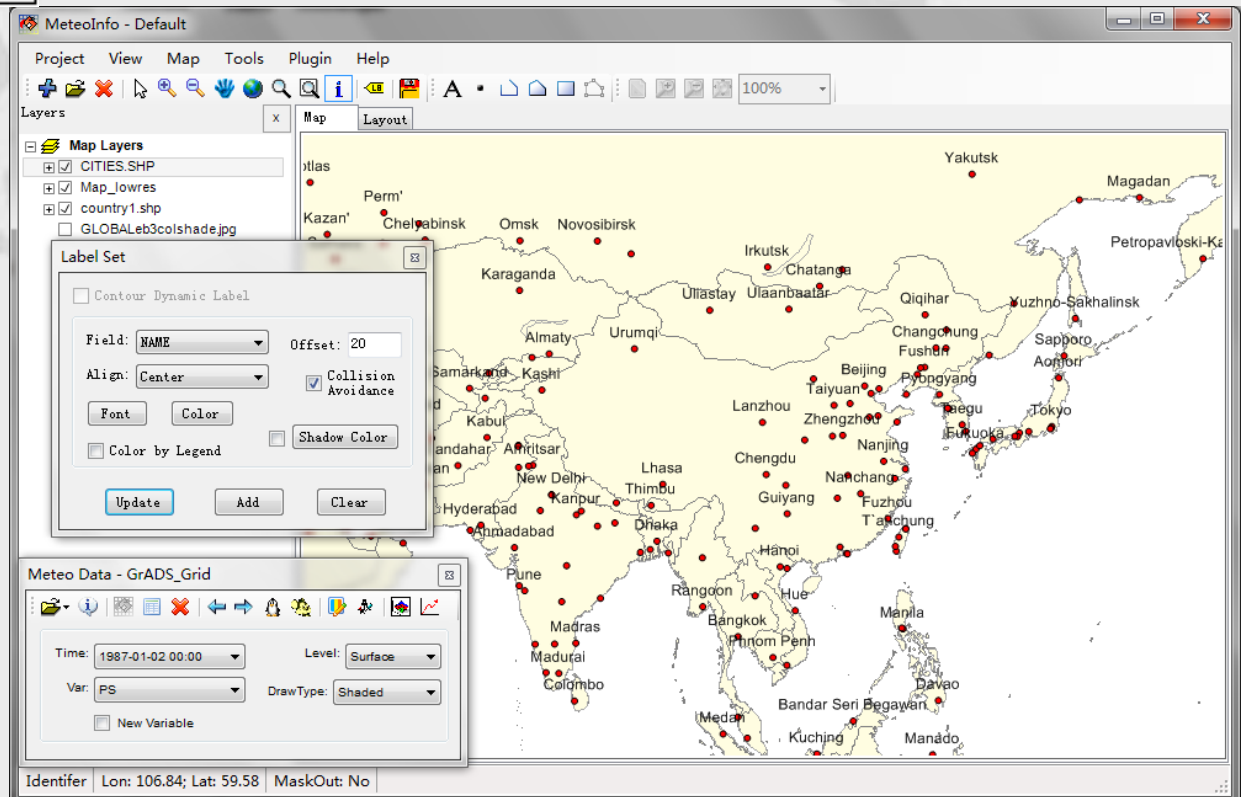
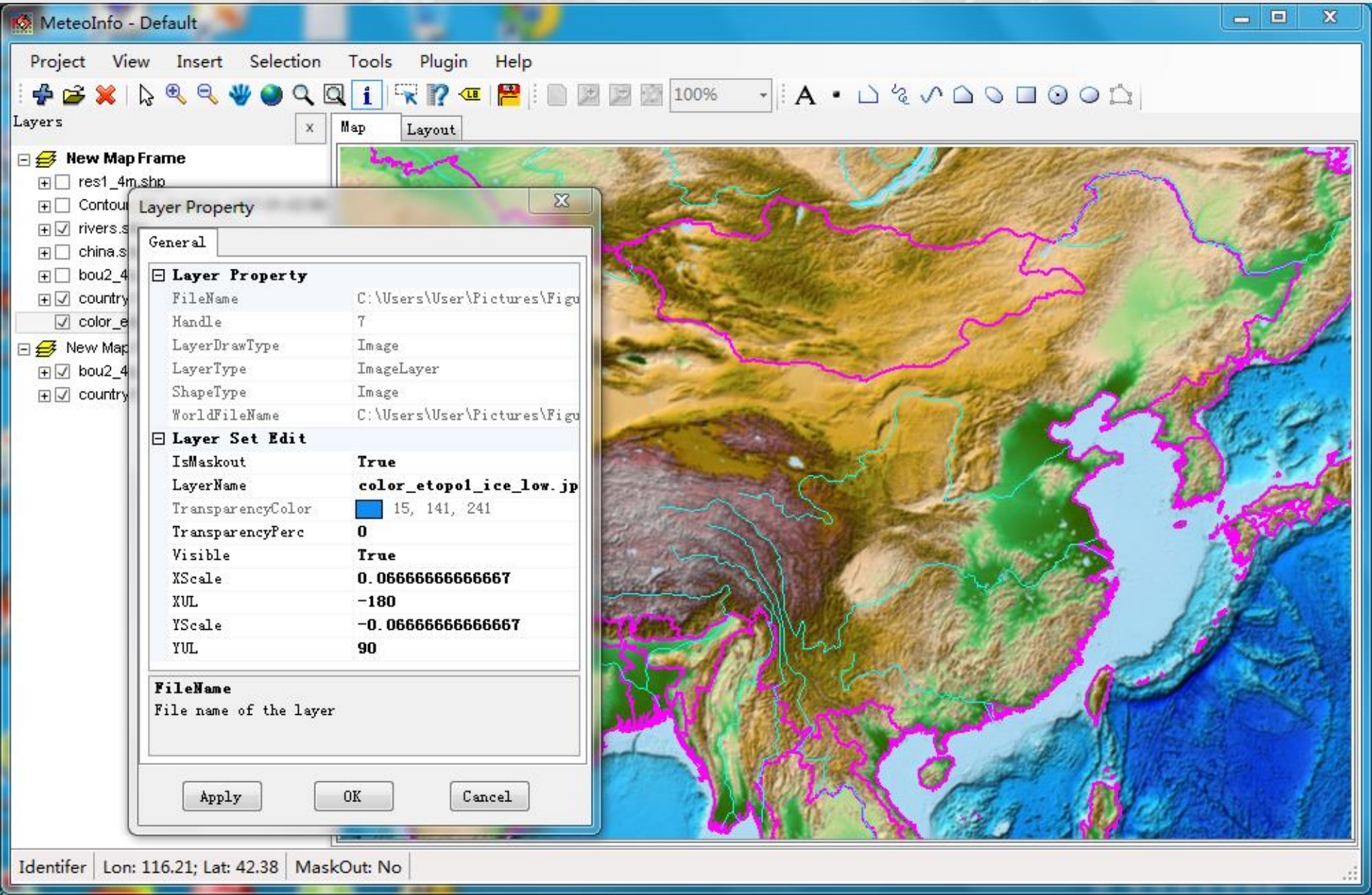


Image layer – after geo-located



Create meteorological data layers

Grid Data

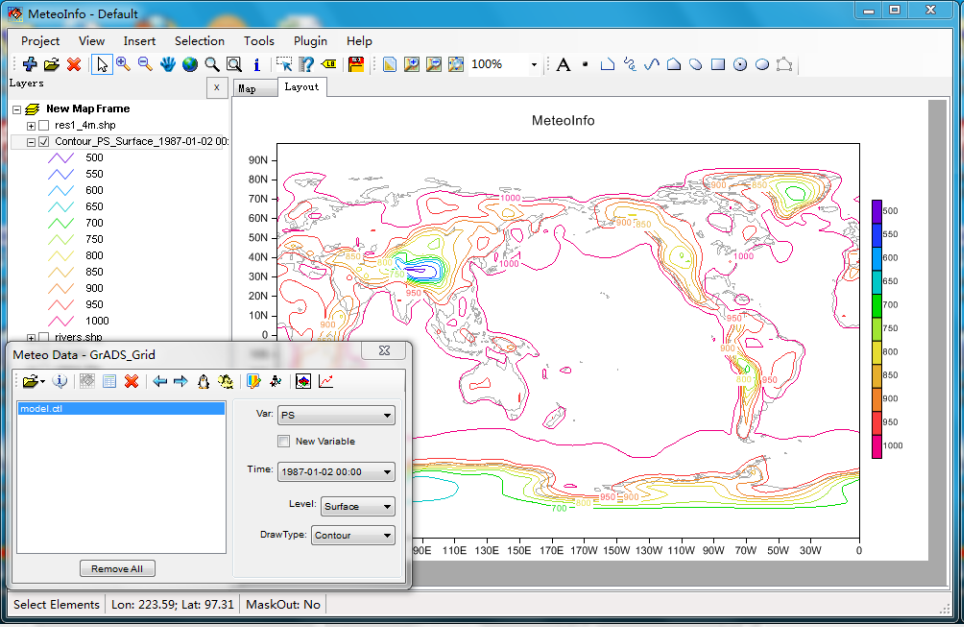
- Contour
- Shaded
- Grid Fill
- Grid Point
- Vector
- Barb
- Raster

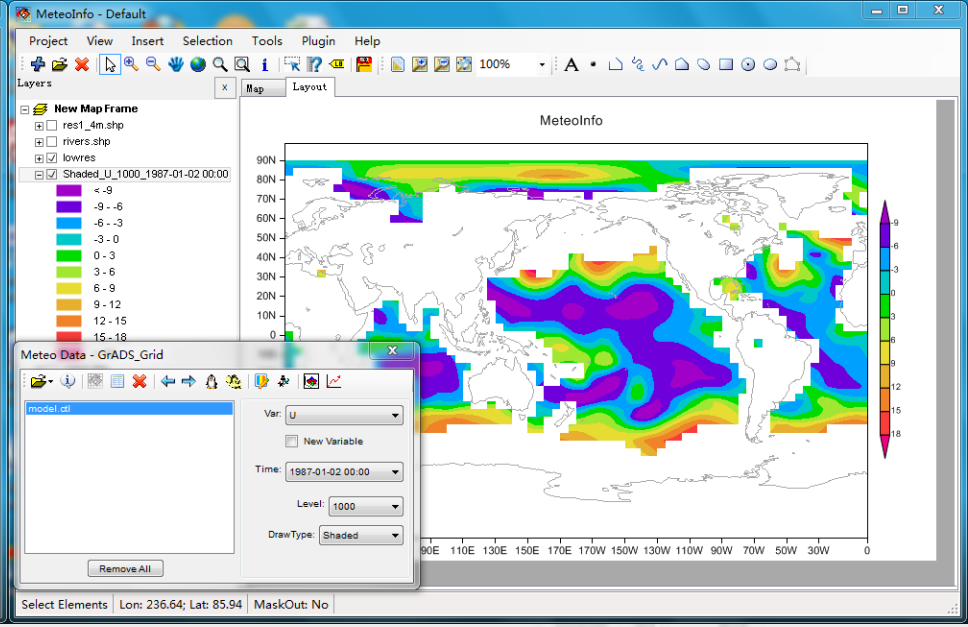
Station Data

- Station Point
- Contour
- Shaded
- Barb
- Weather Symbol
- Station Model
- Station Info

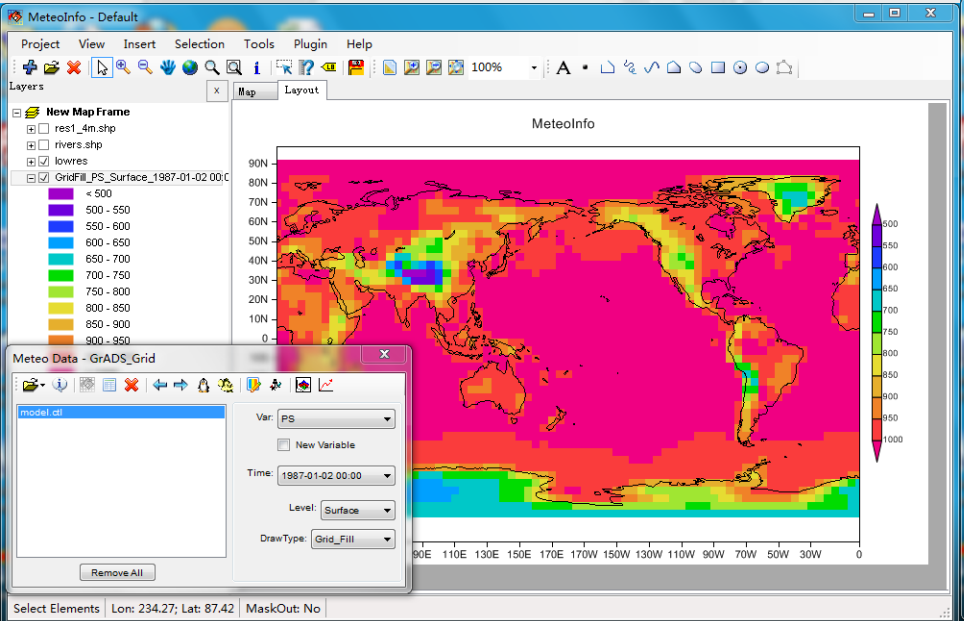
Line Data

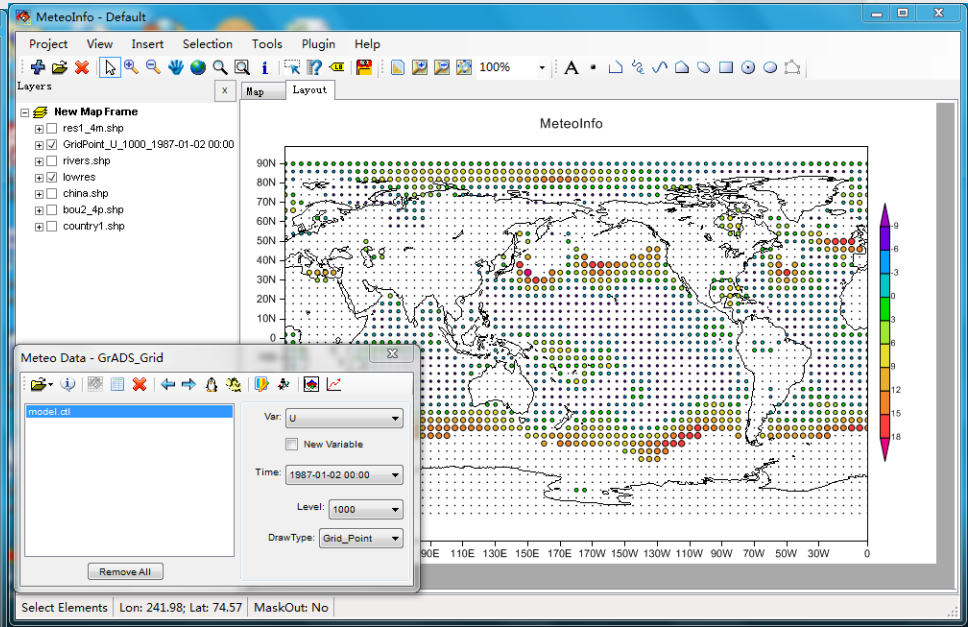
- TrajLine
- TrajStartPoint
- TrajPoint

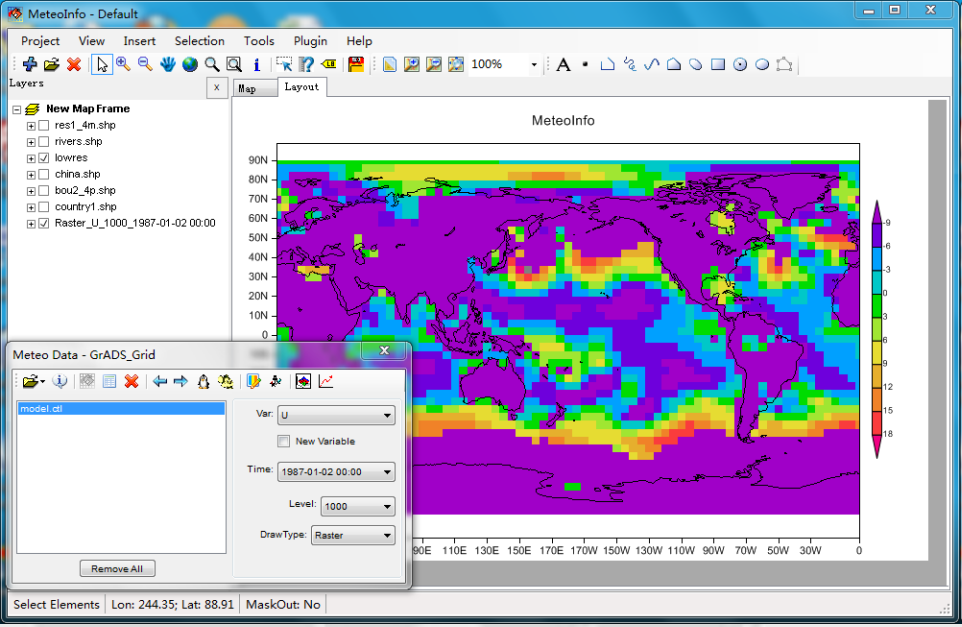


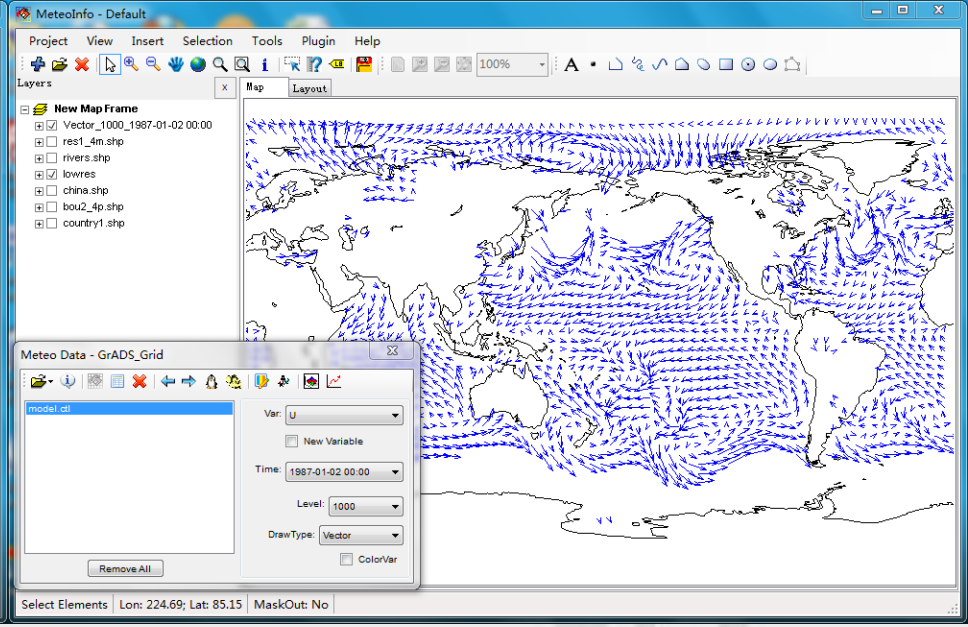


The layers created from grid data

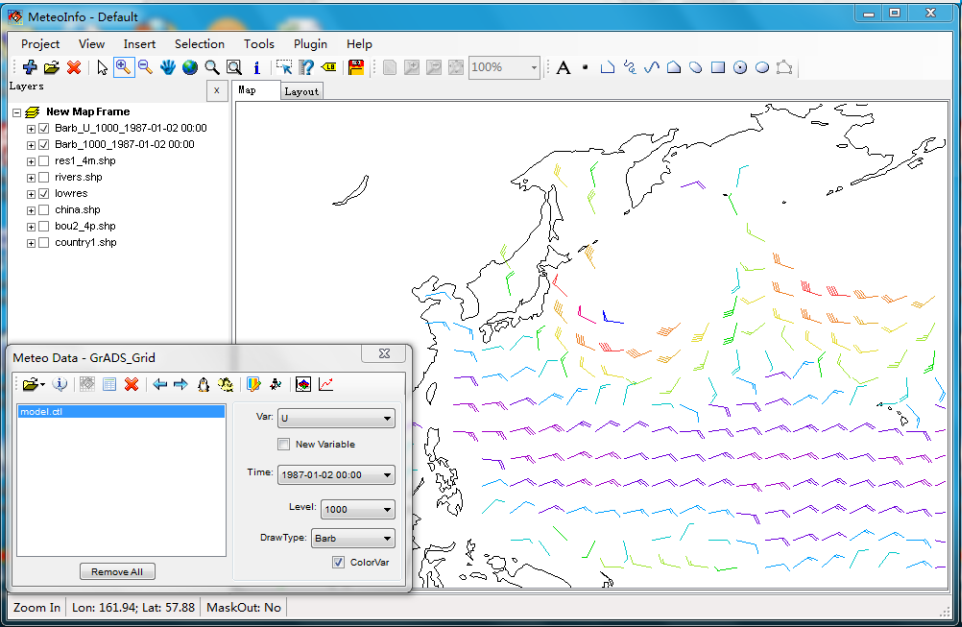


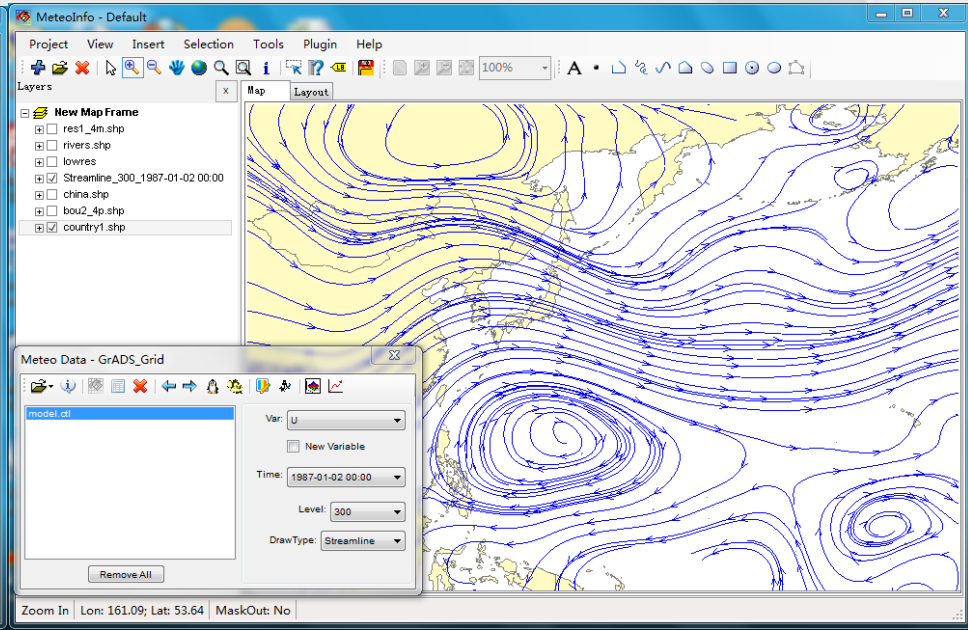


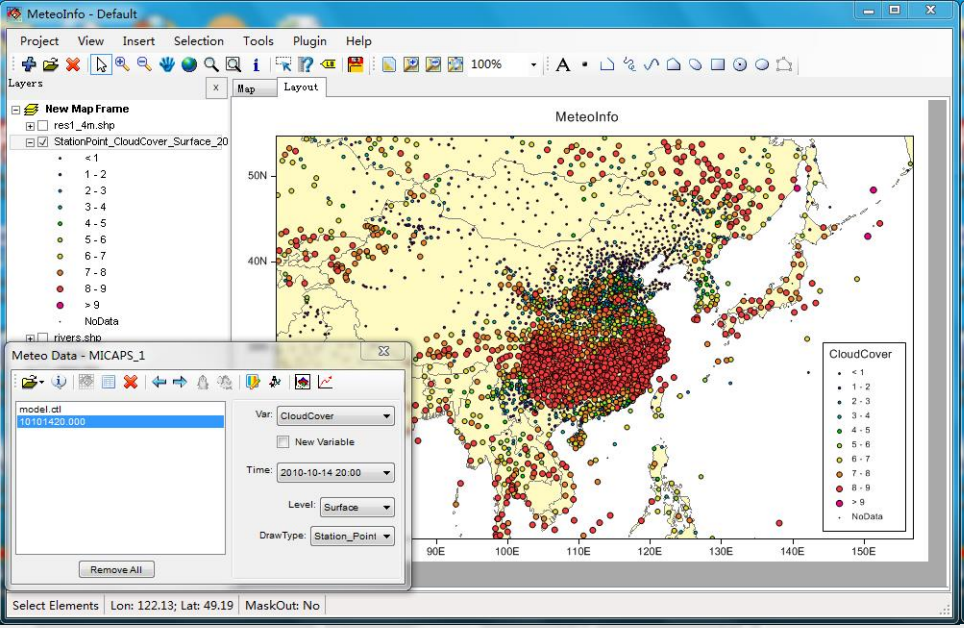


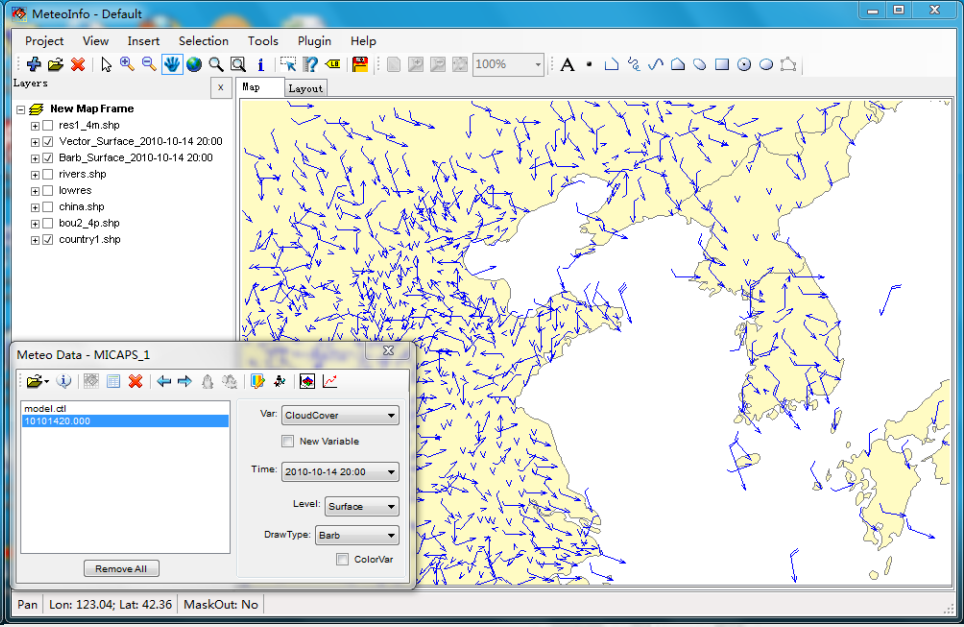


The layers created from grid data

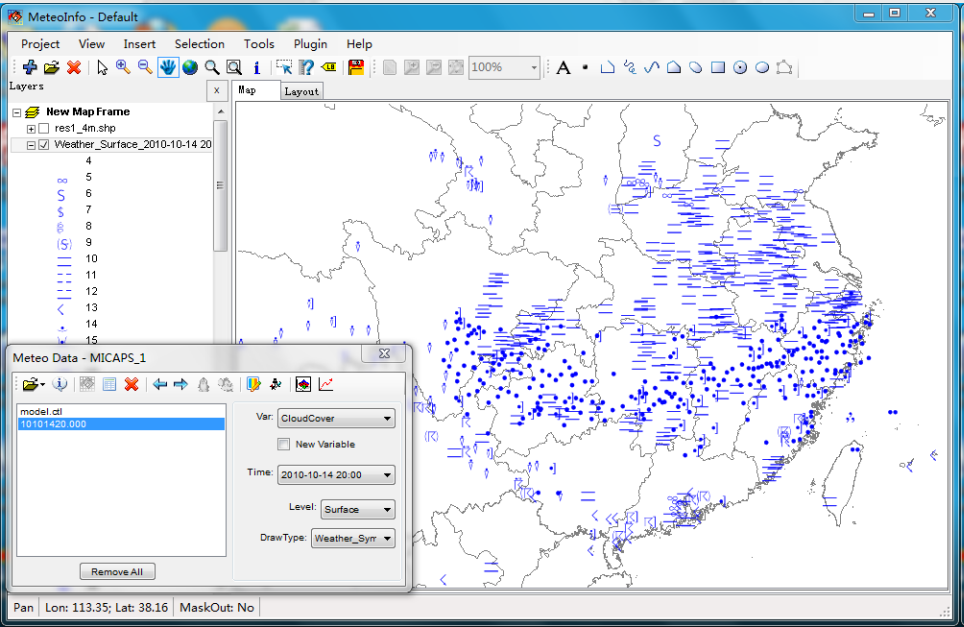


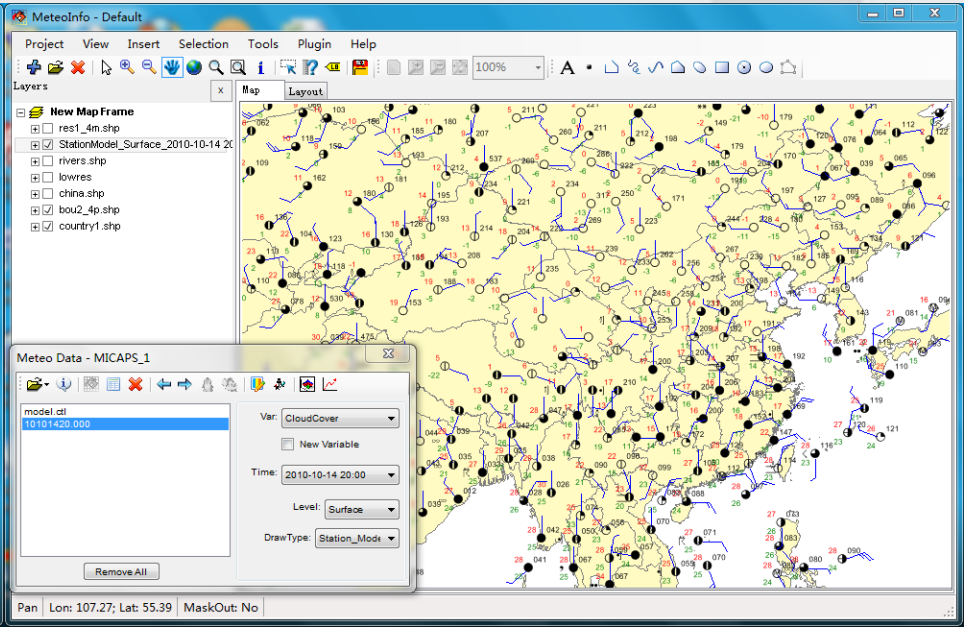






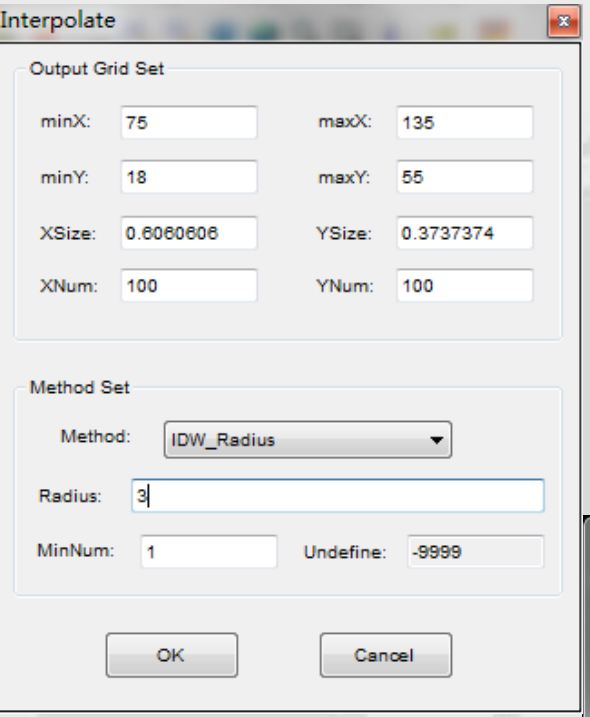
The layers created from station data

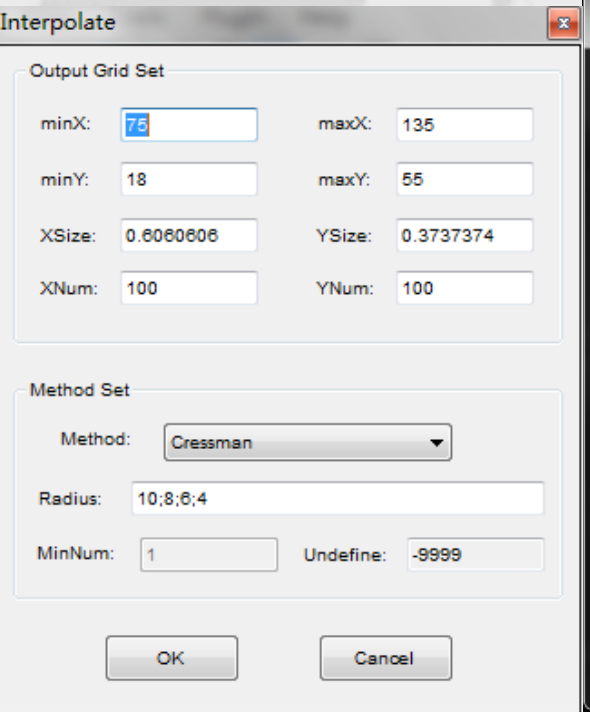


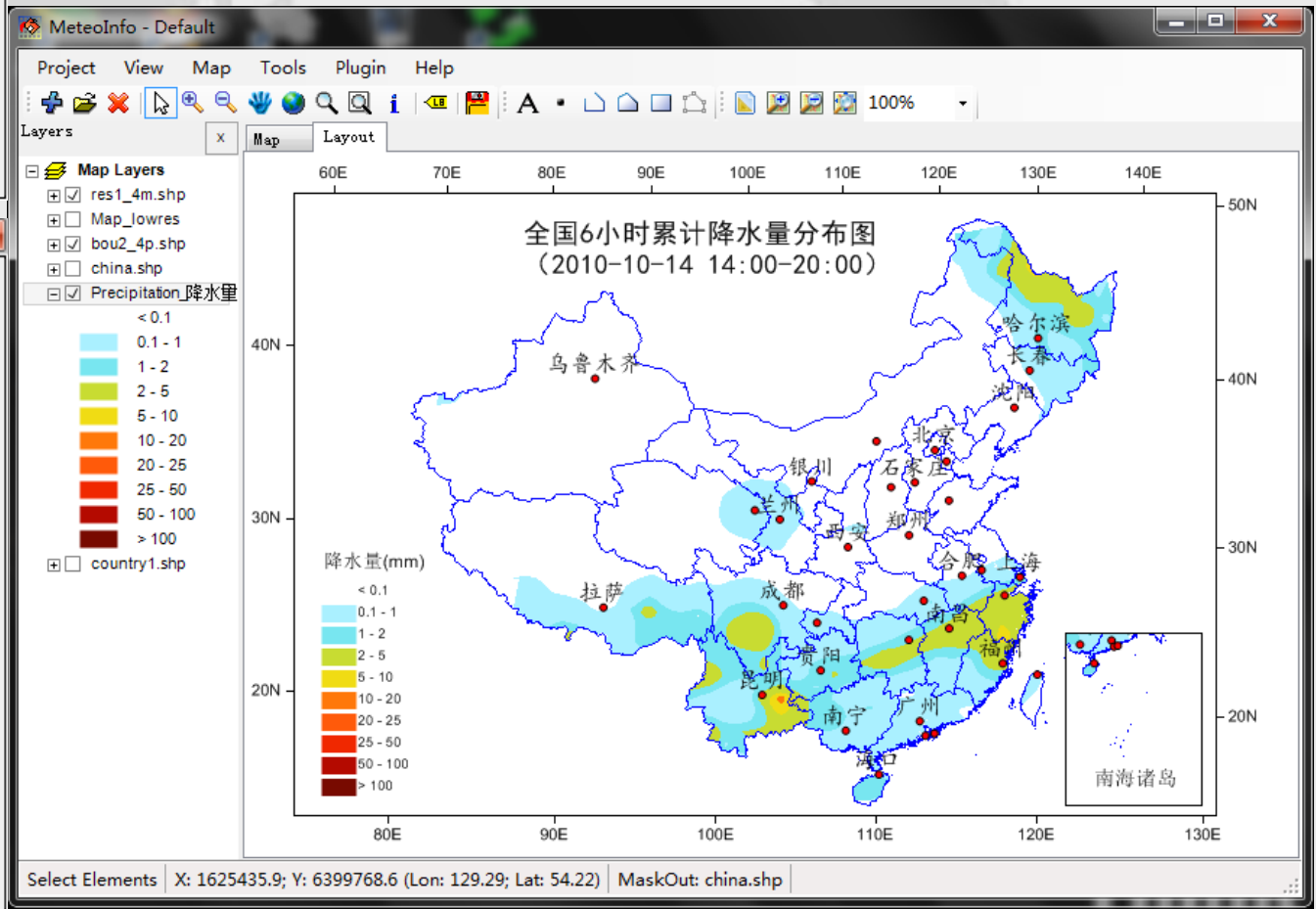


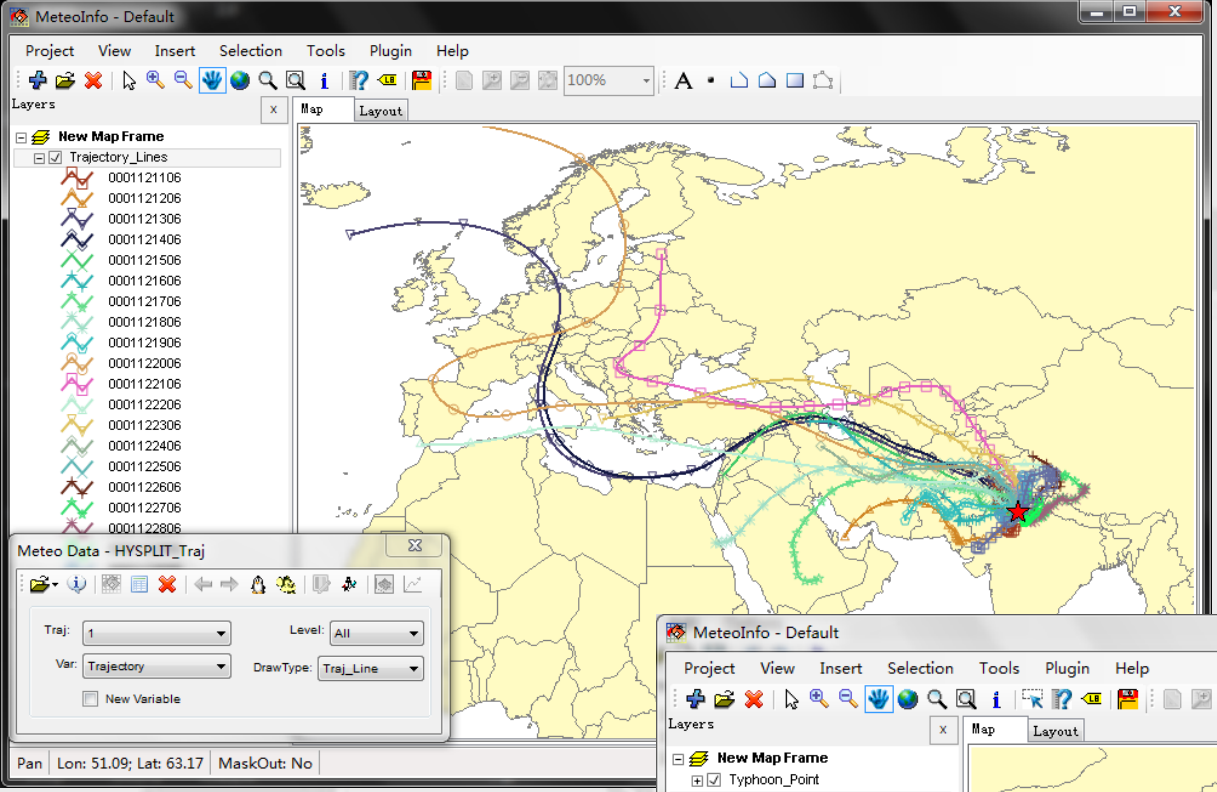
Interpolate station data to grid data, then create contour or shaded layer

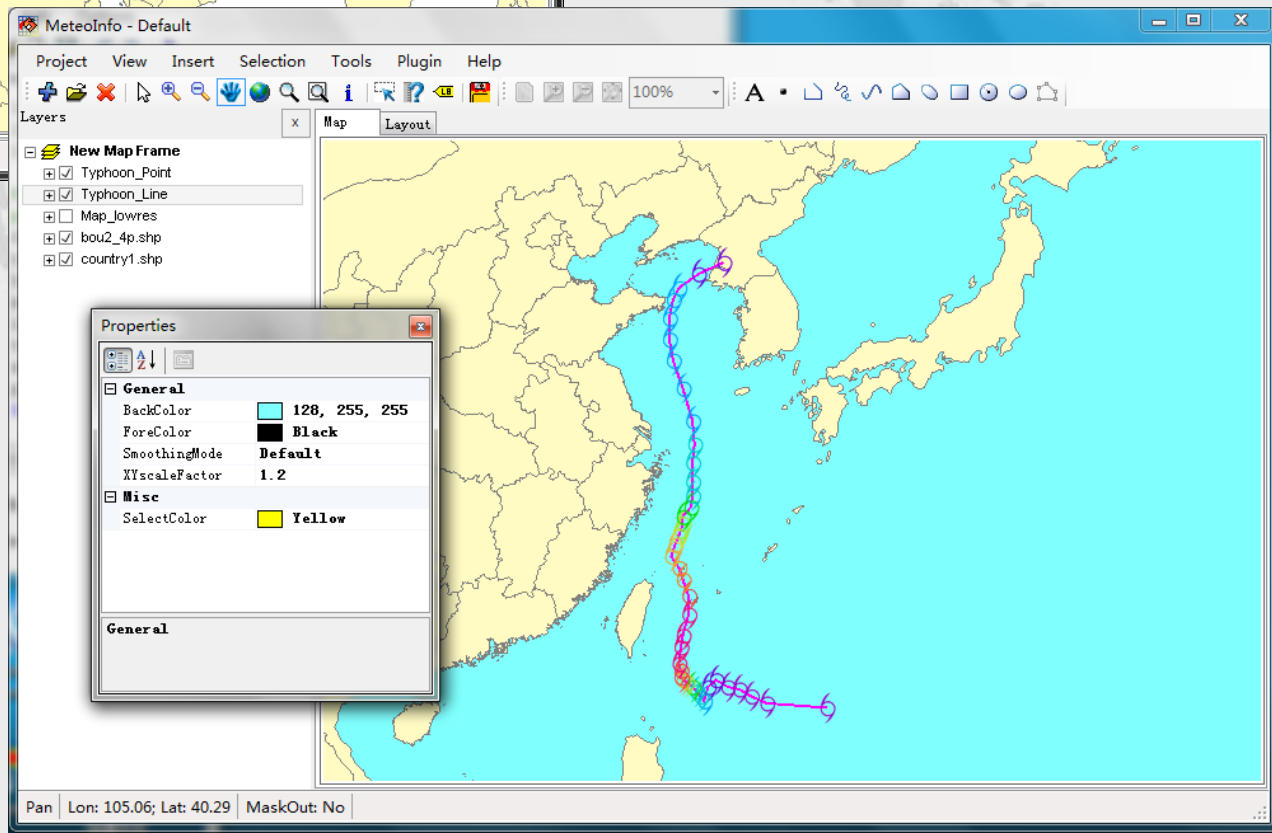
Cressman
IDW_Radius
IDW_Neighbors









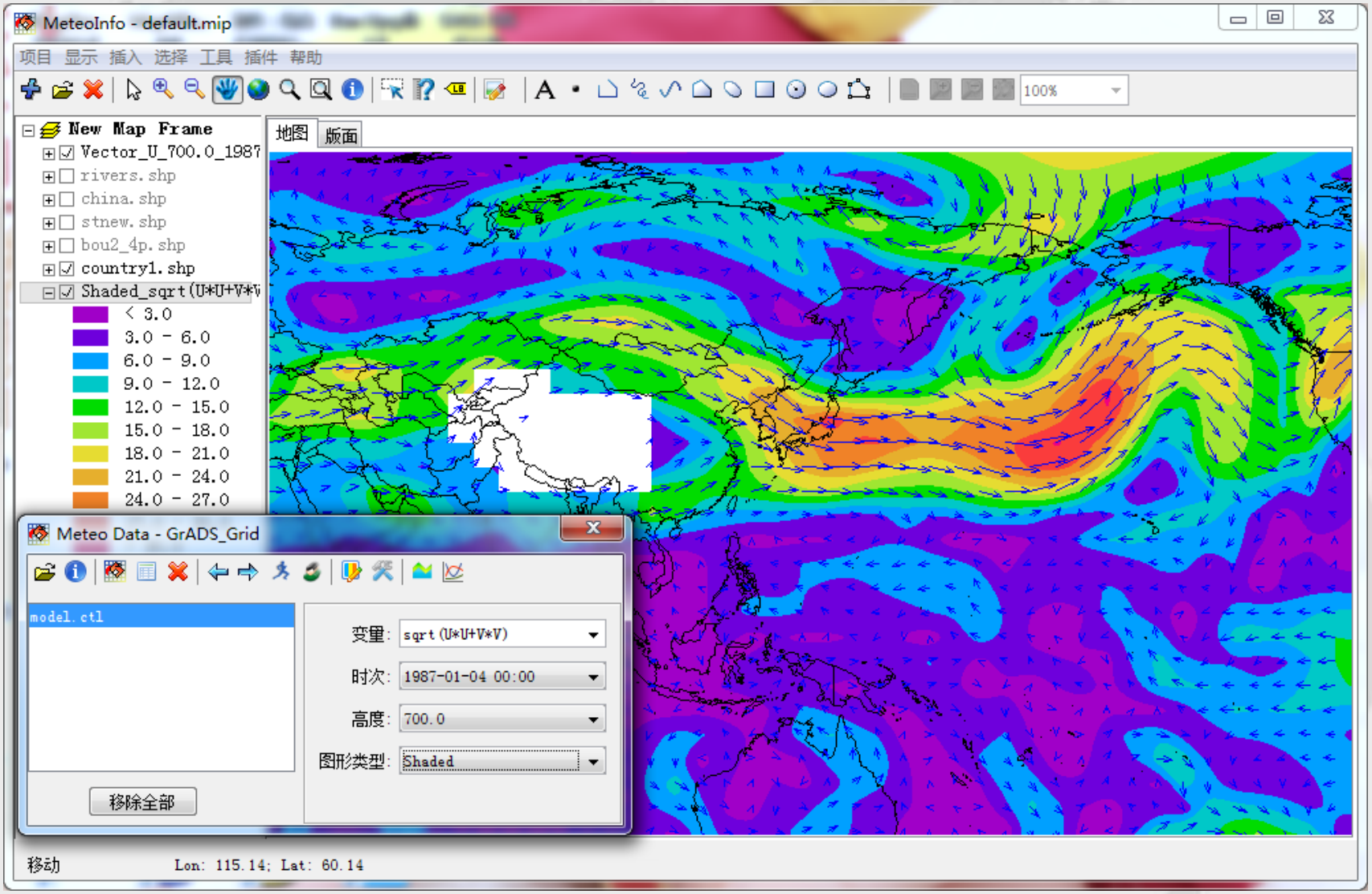


Line data

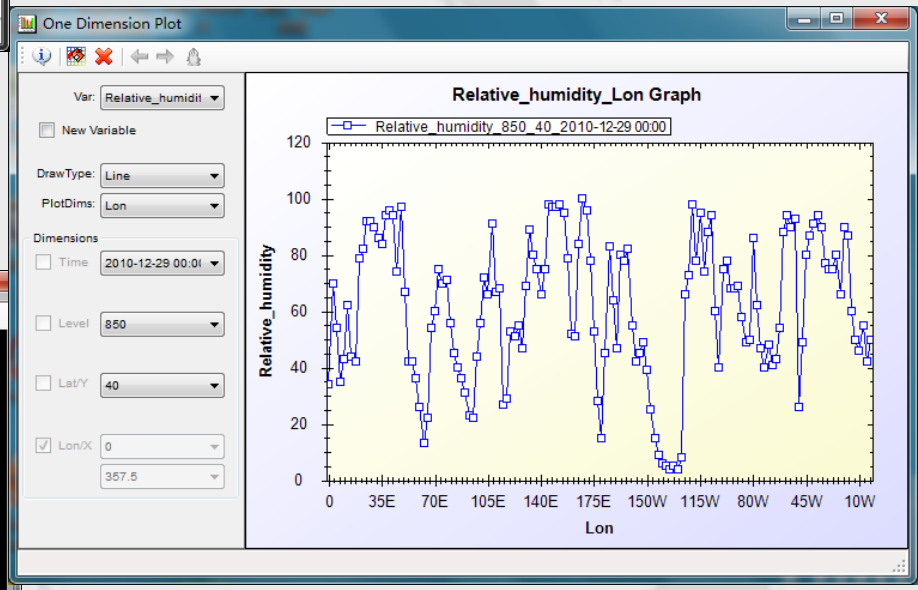
Trajectory

Typhoon

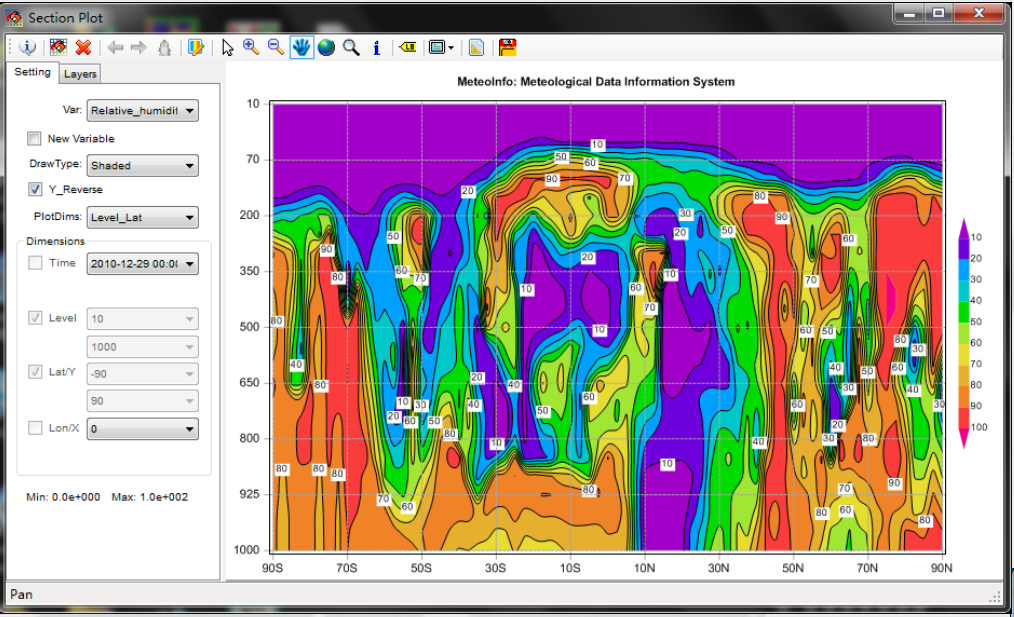
New variable created form existing variables

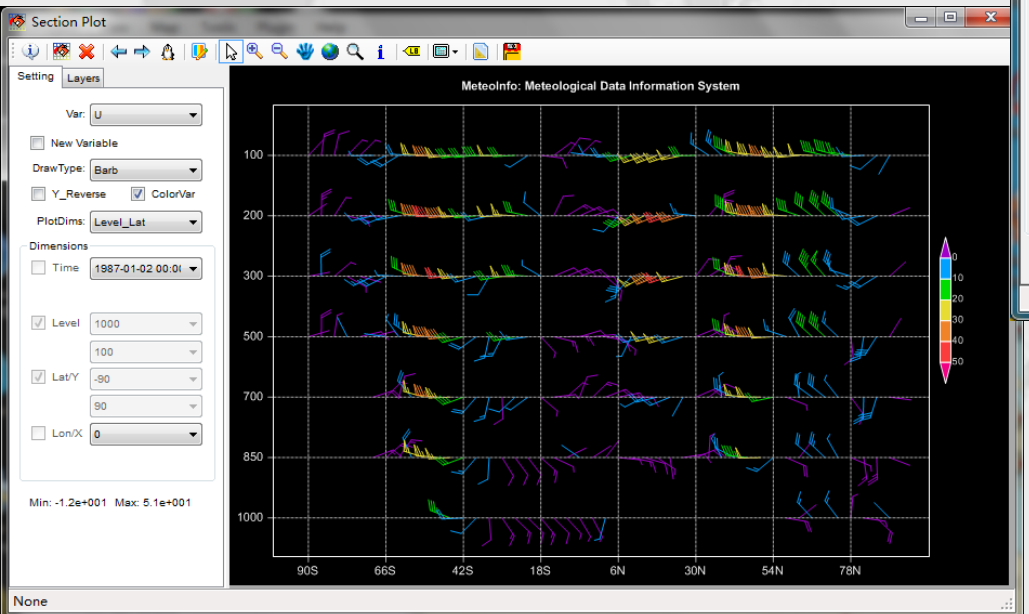


One dimension plot

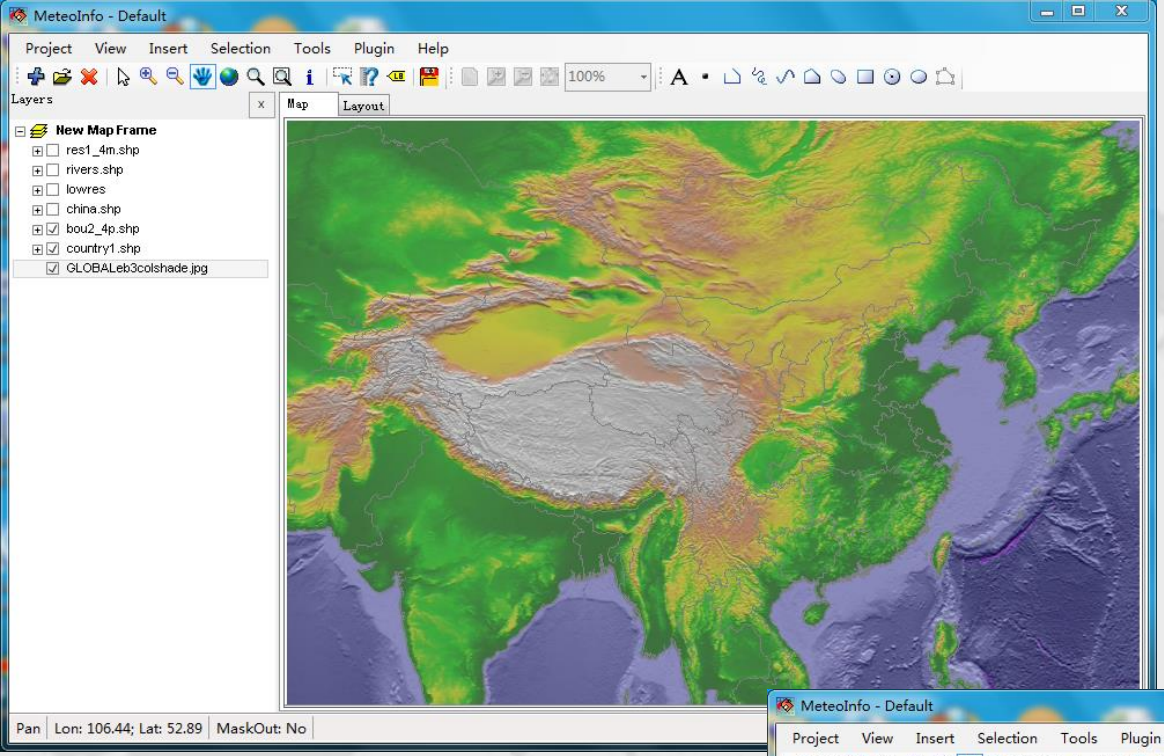


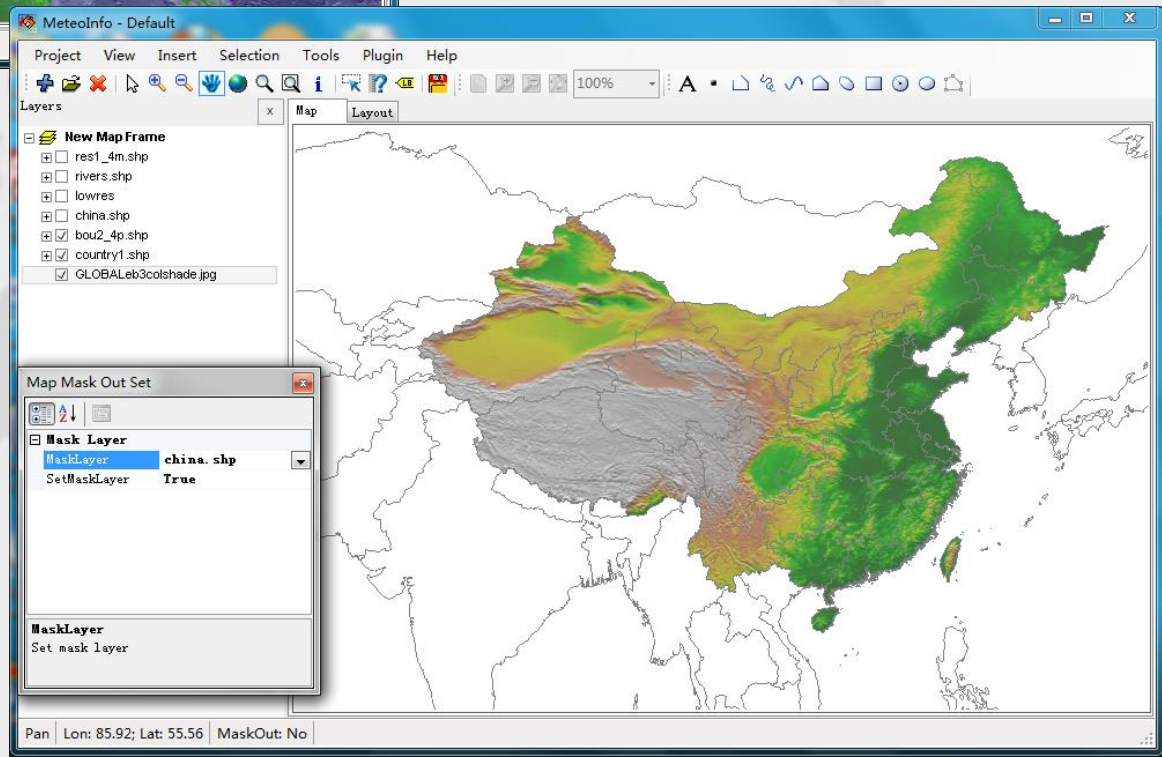
Section plot





Mask out layer

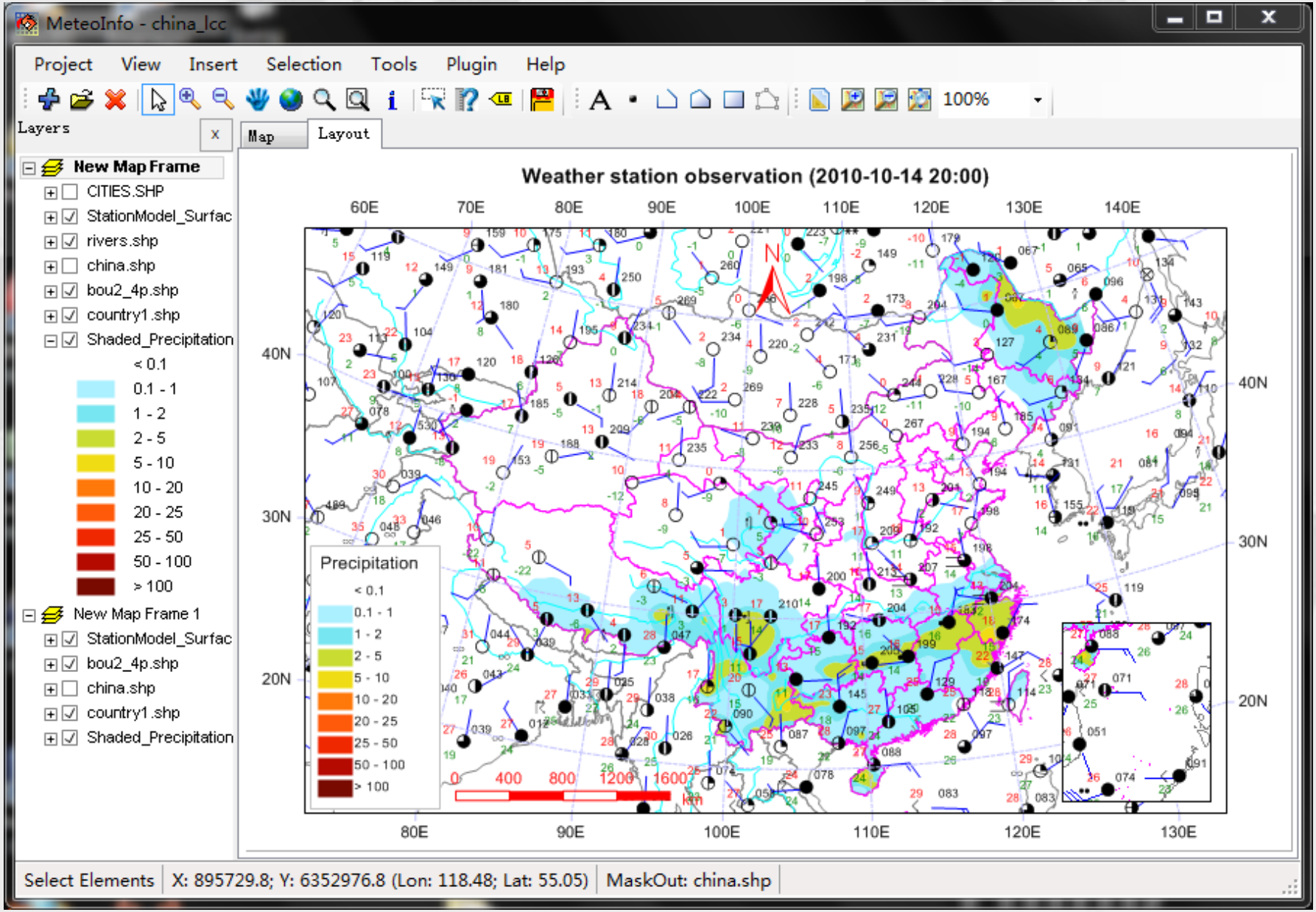




Each layer has ‘IsMaskOut’ property.

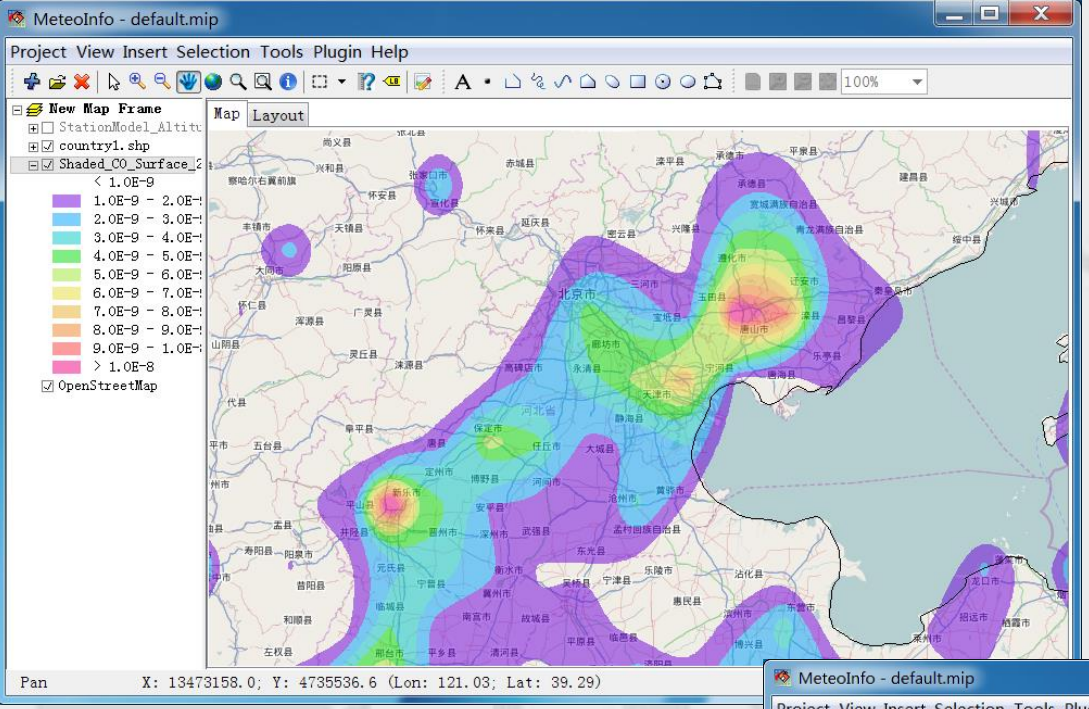
The mask out layer is only validate to the layers with true ‘IsMaskOut’ property. So you can control which layers will be masked out.

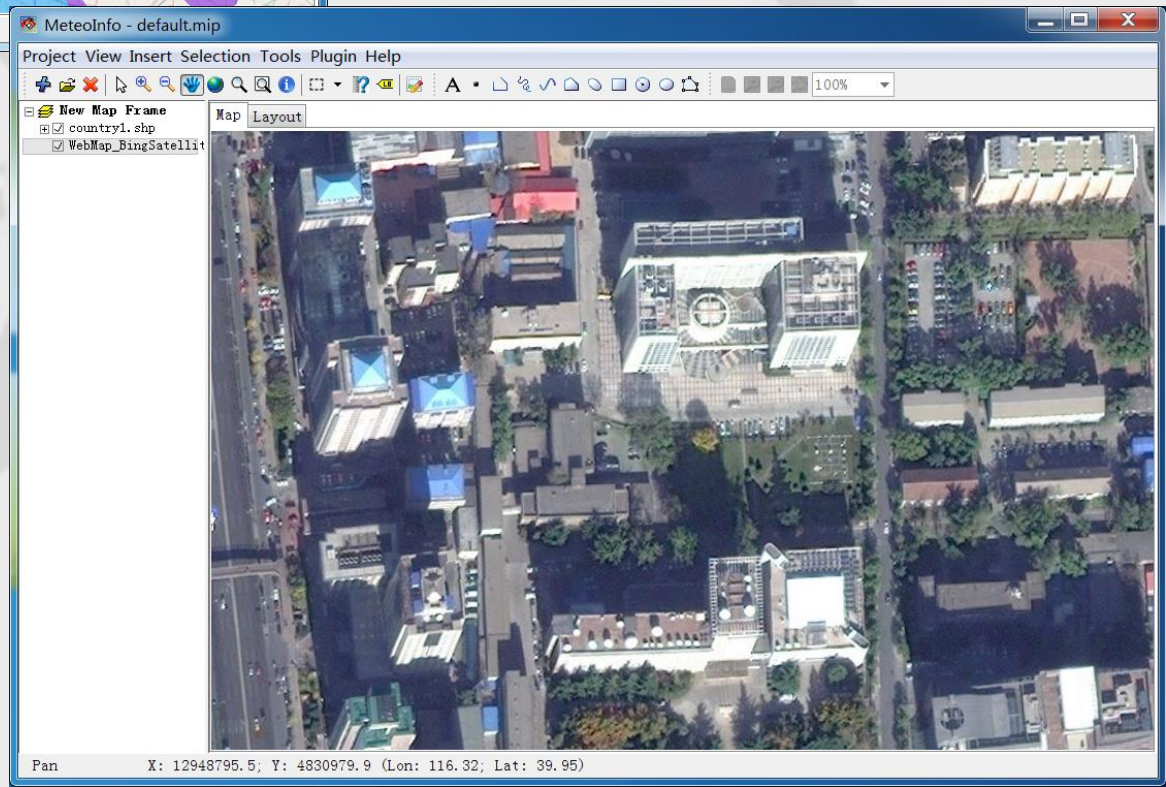
MeteoInfo Layout



Add web map layer

Mercator projection

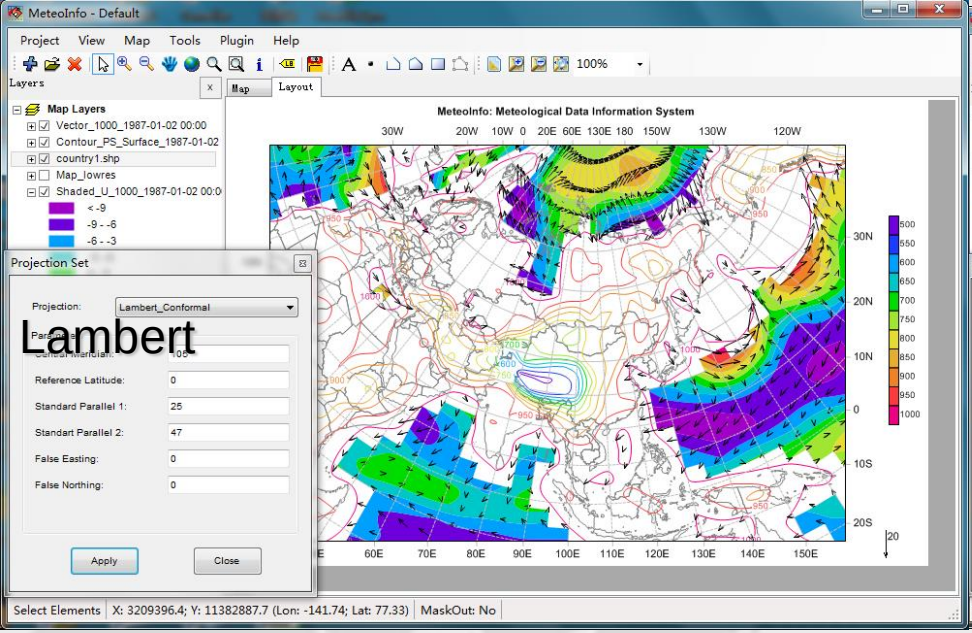


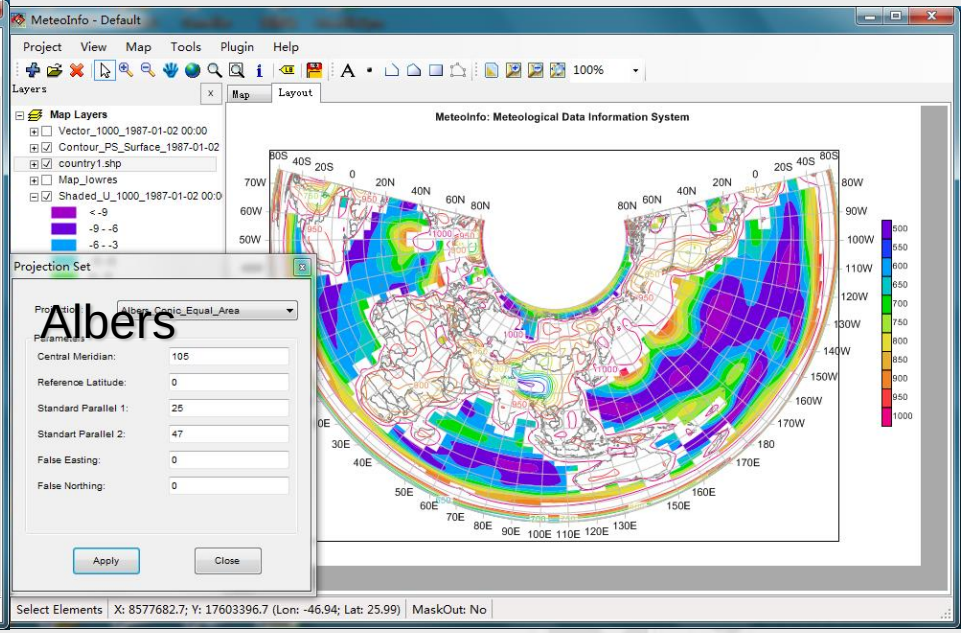


- OpenStreetMap
- OpenStreetMapQuestSatellite
- BingMap
- BingSatelliteMap
- BingHybridMap
- OviMap
- OviSatelliteMap
- OviTerrainMap
- OviHybridMap
- YahooMap
- YahooSatelliteMap
- YahooHybridMap

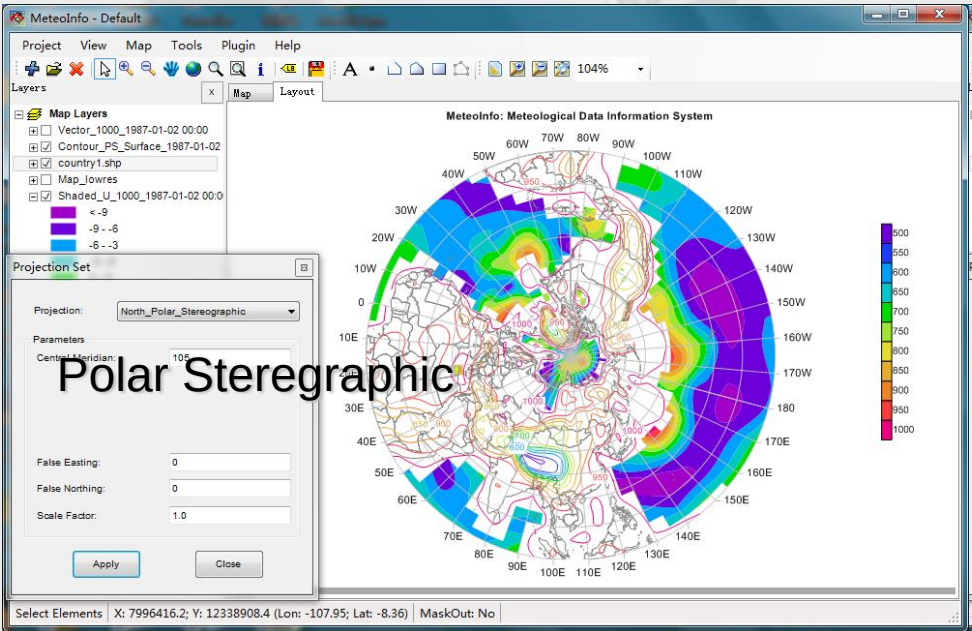
Map projection

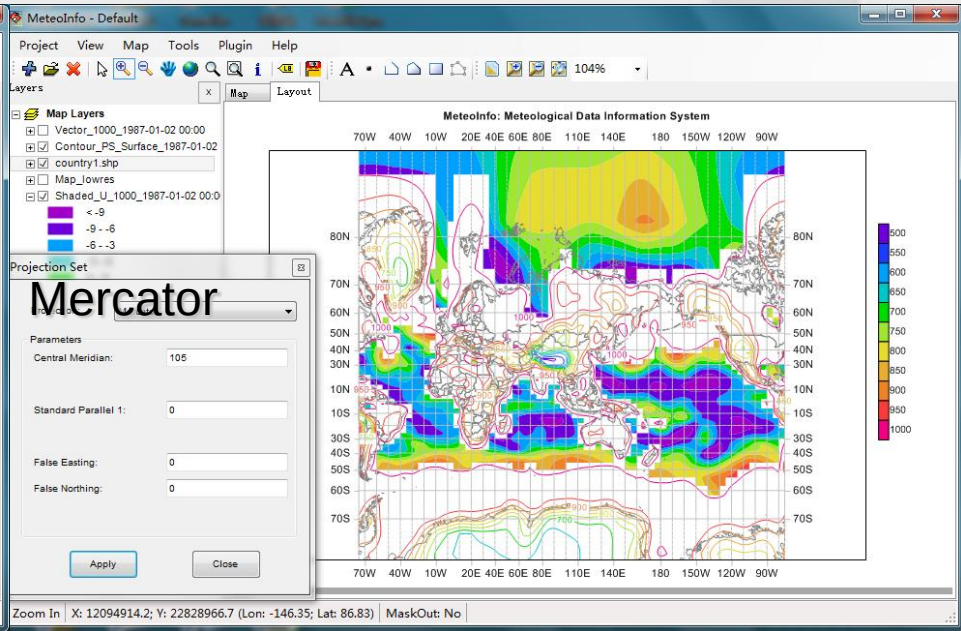
- Lambert_Conformal
- Albers_Conic_Equal_Area
- North_Polar_Stereographic
- South_Polar_Stereographic
- Mercator
- Robinson
- Mollweide
- Orthographic
- Oblique_Stereographic
- Transverse_Mercator

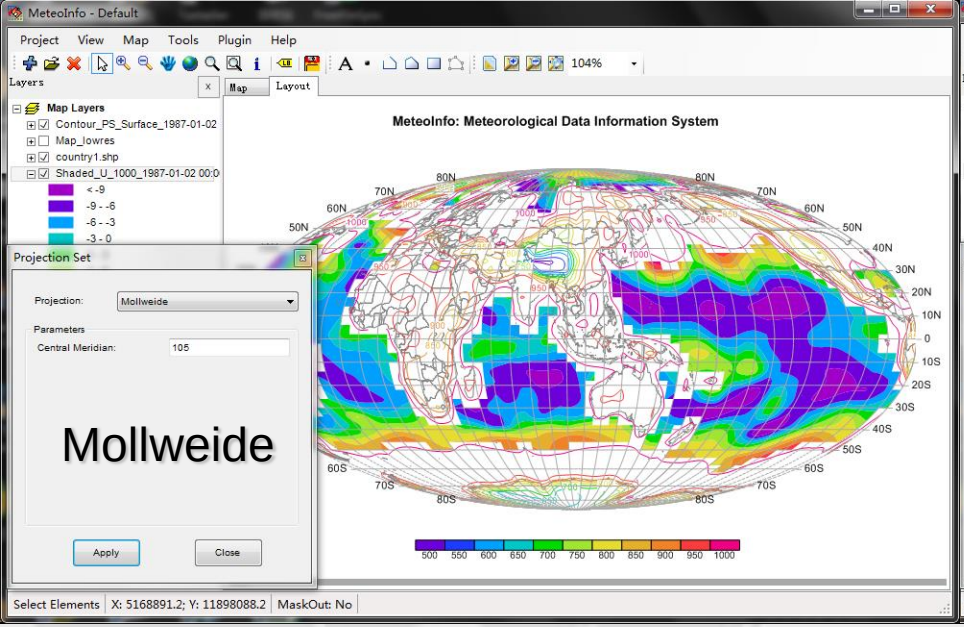


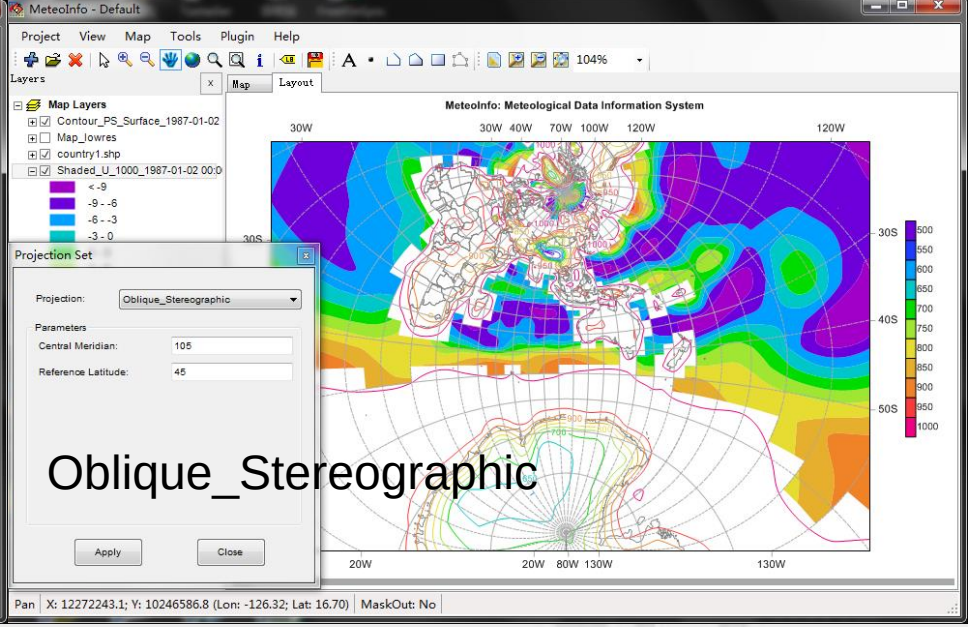


Map projection example

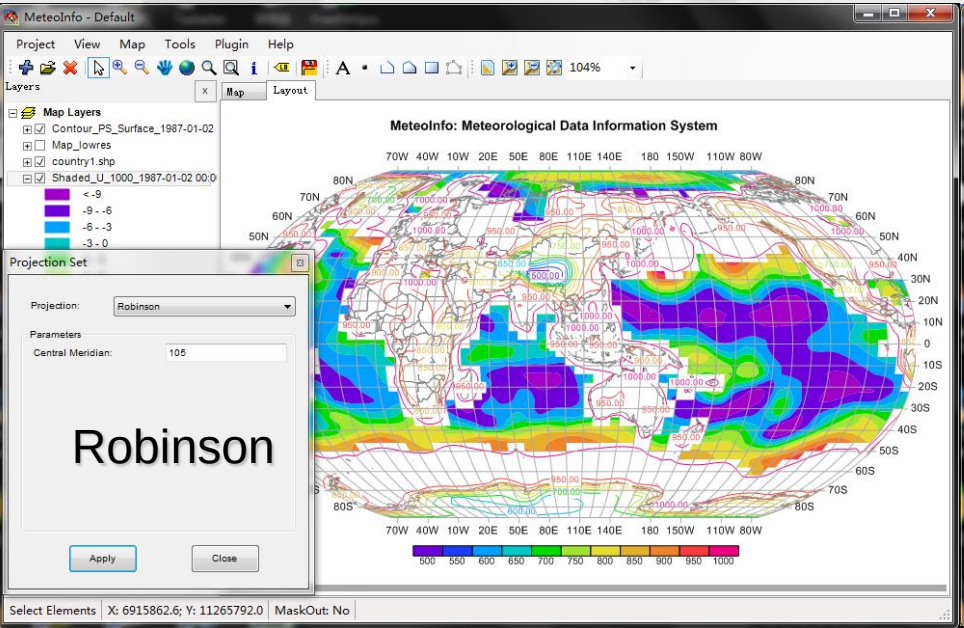


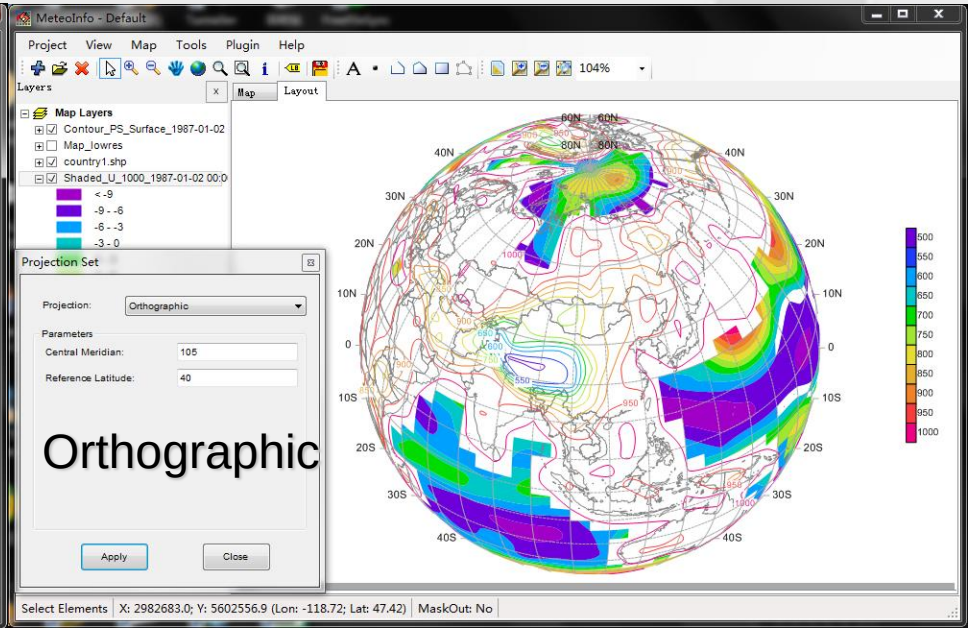


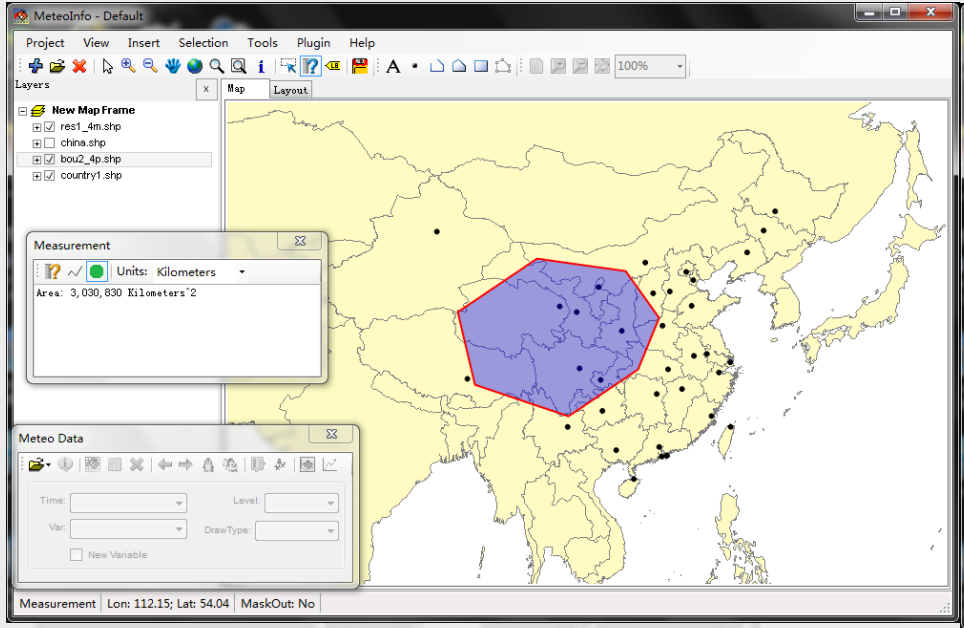


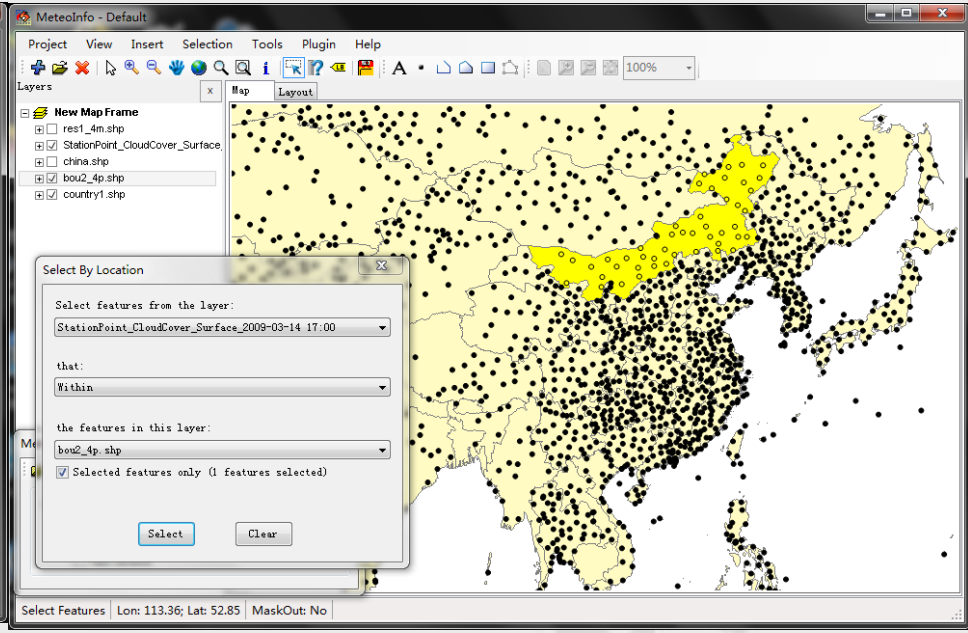


Map projection example

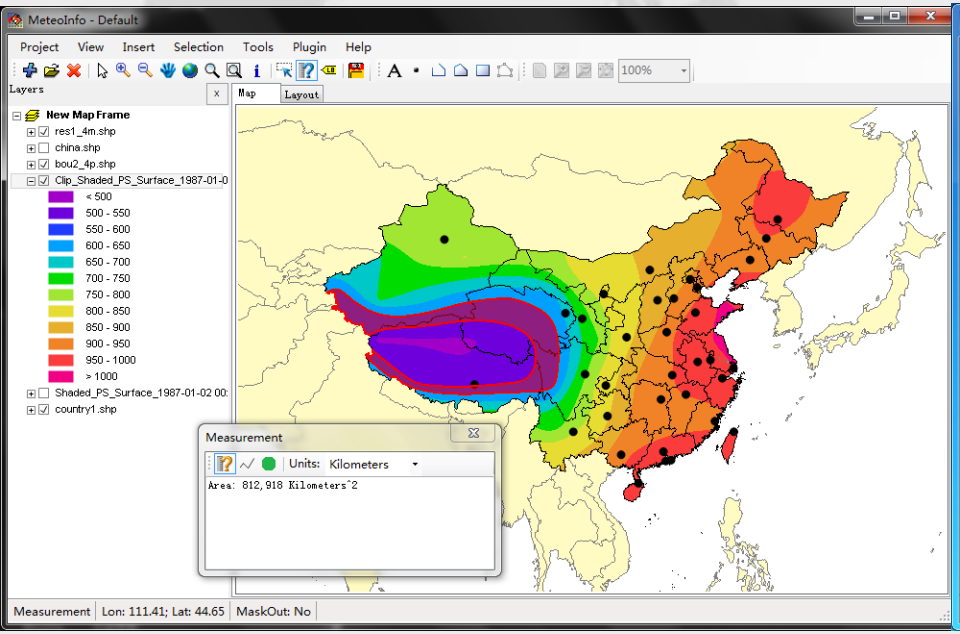


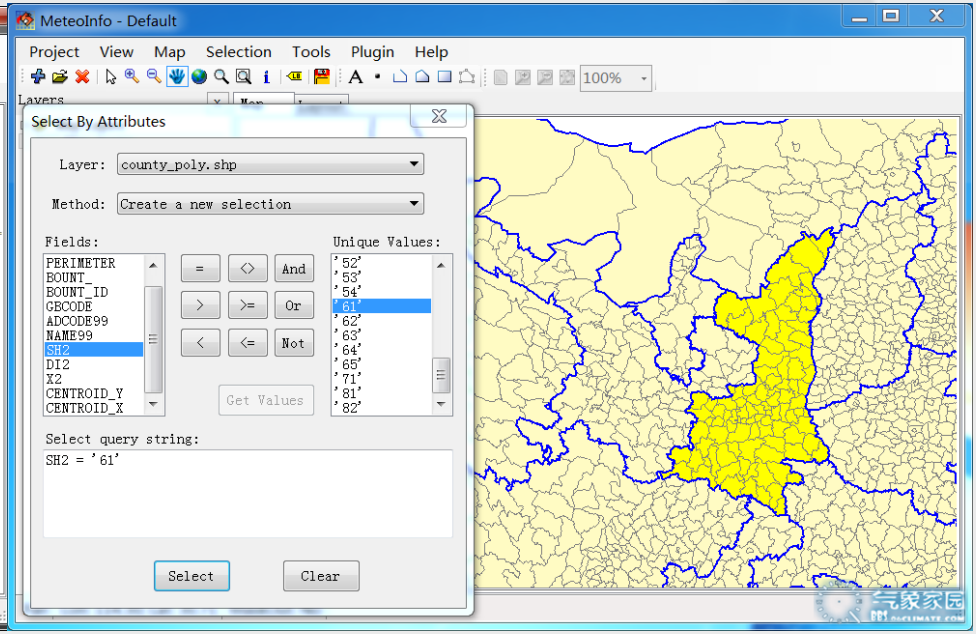


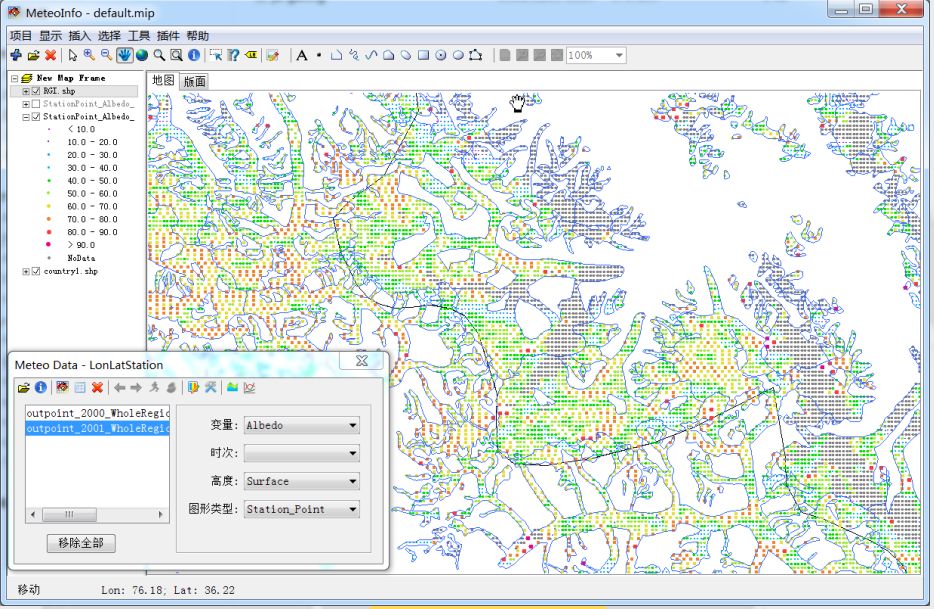




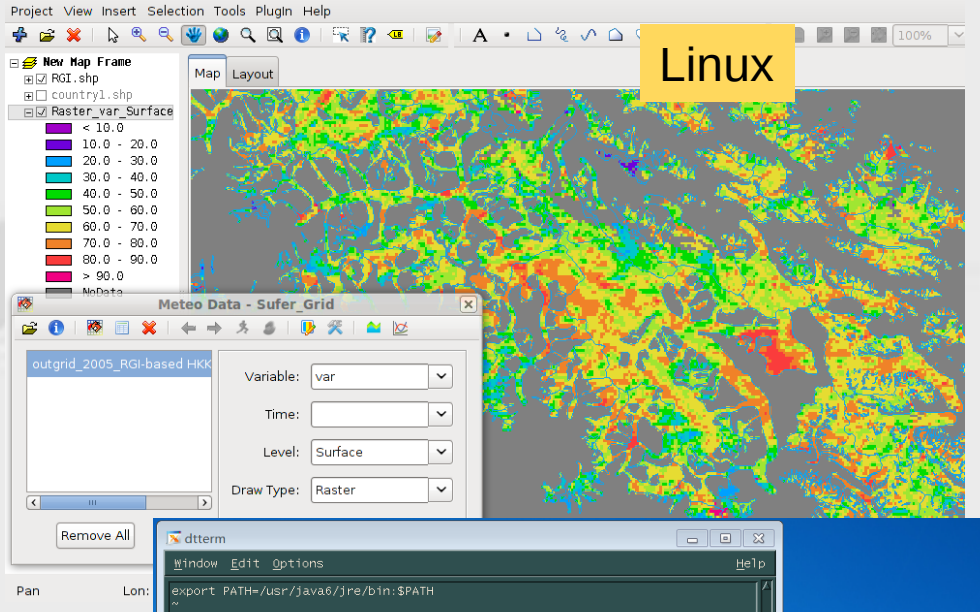
Analysis functions





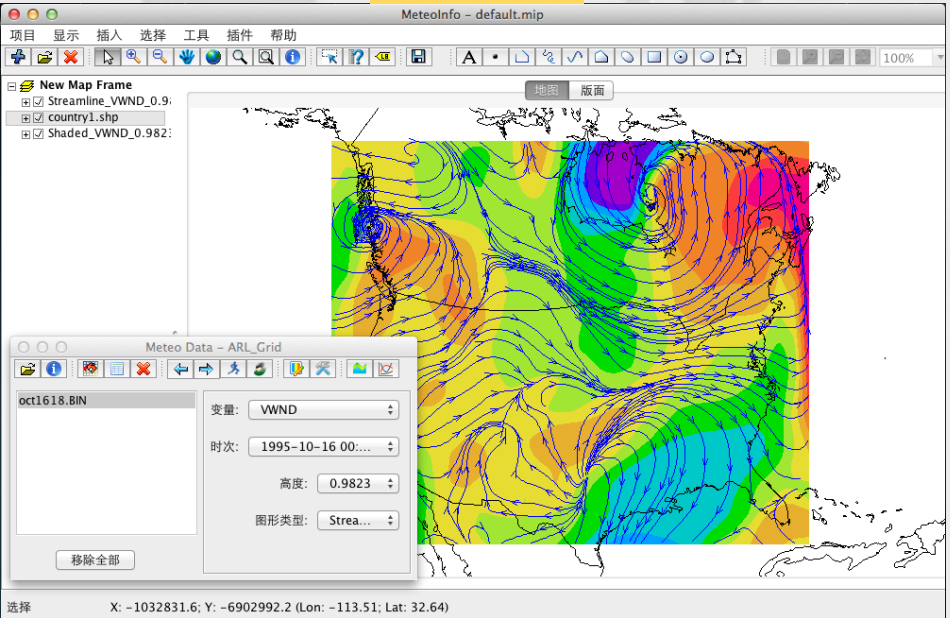


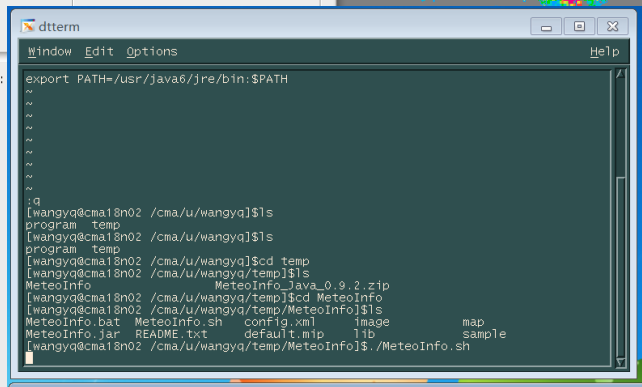
Windows



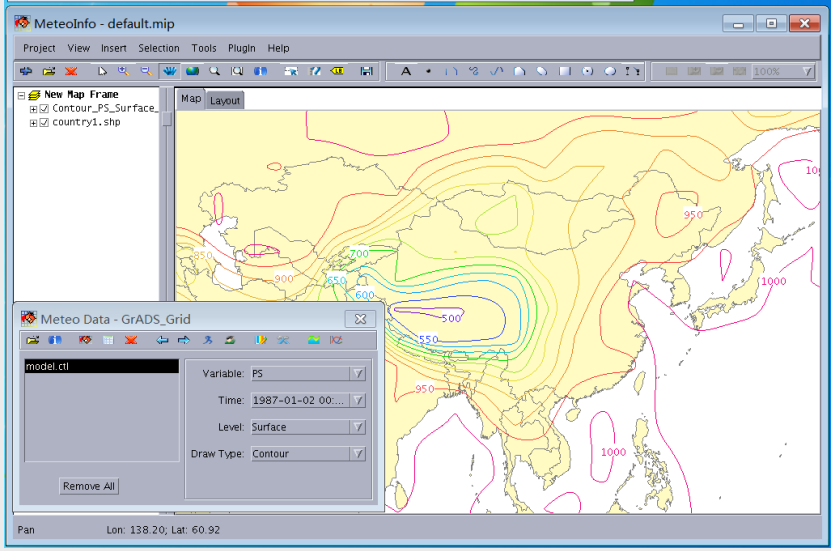
Cross platform

Mac OS



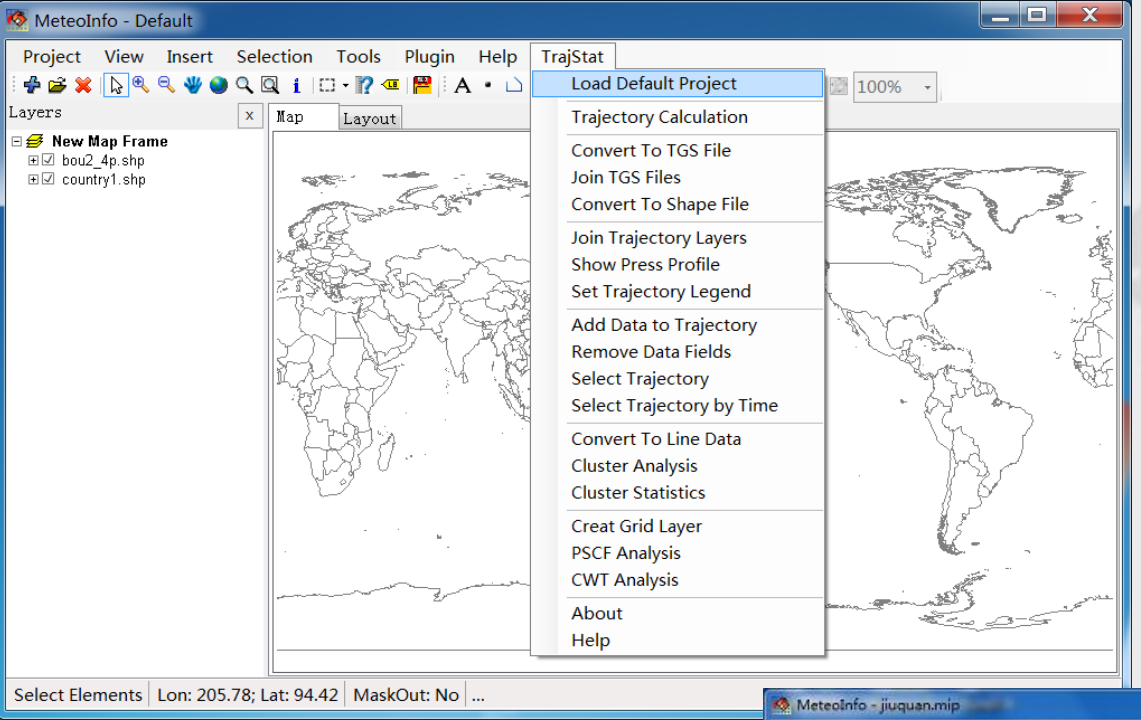


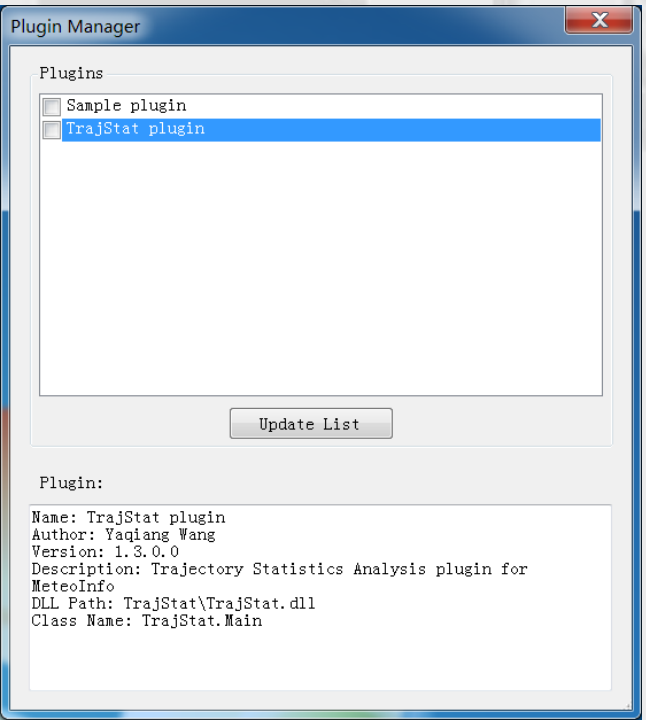
Unix

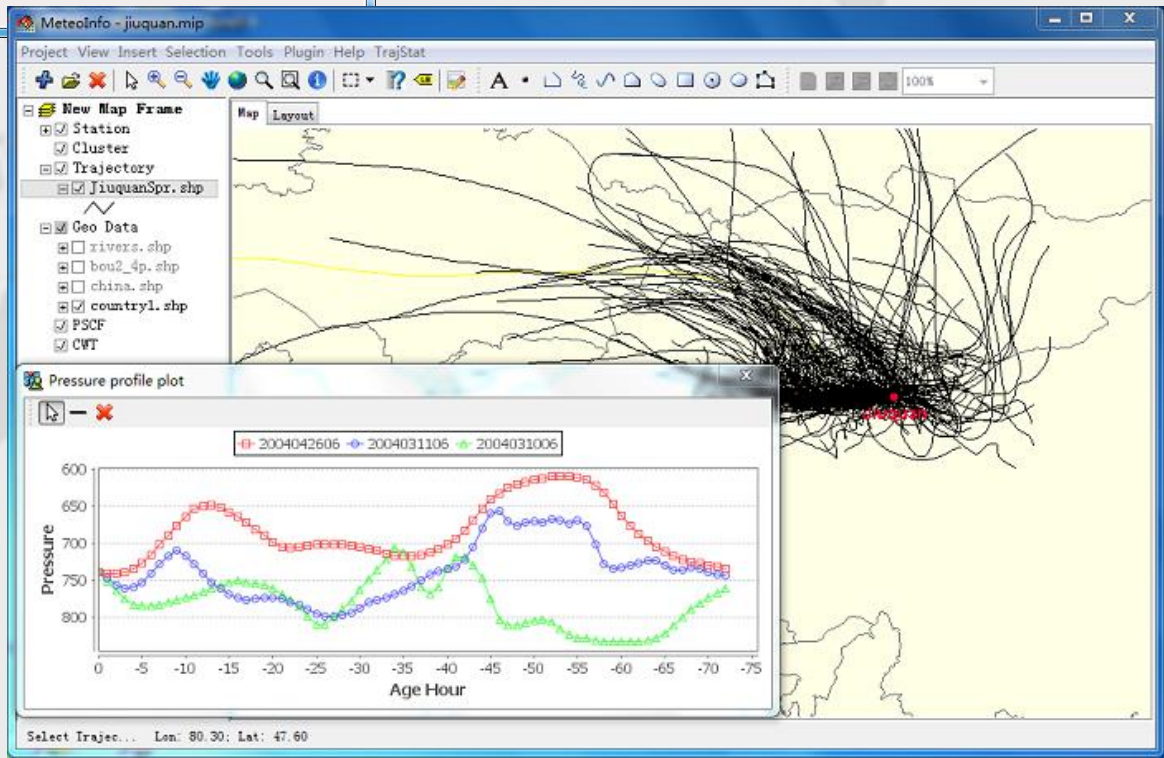


Plugin: Extend MeteoInfo functions

TrajStat (Wang 2009 EMS)



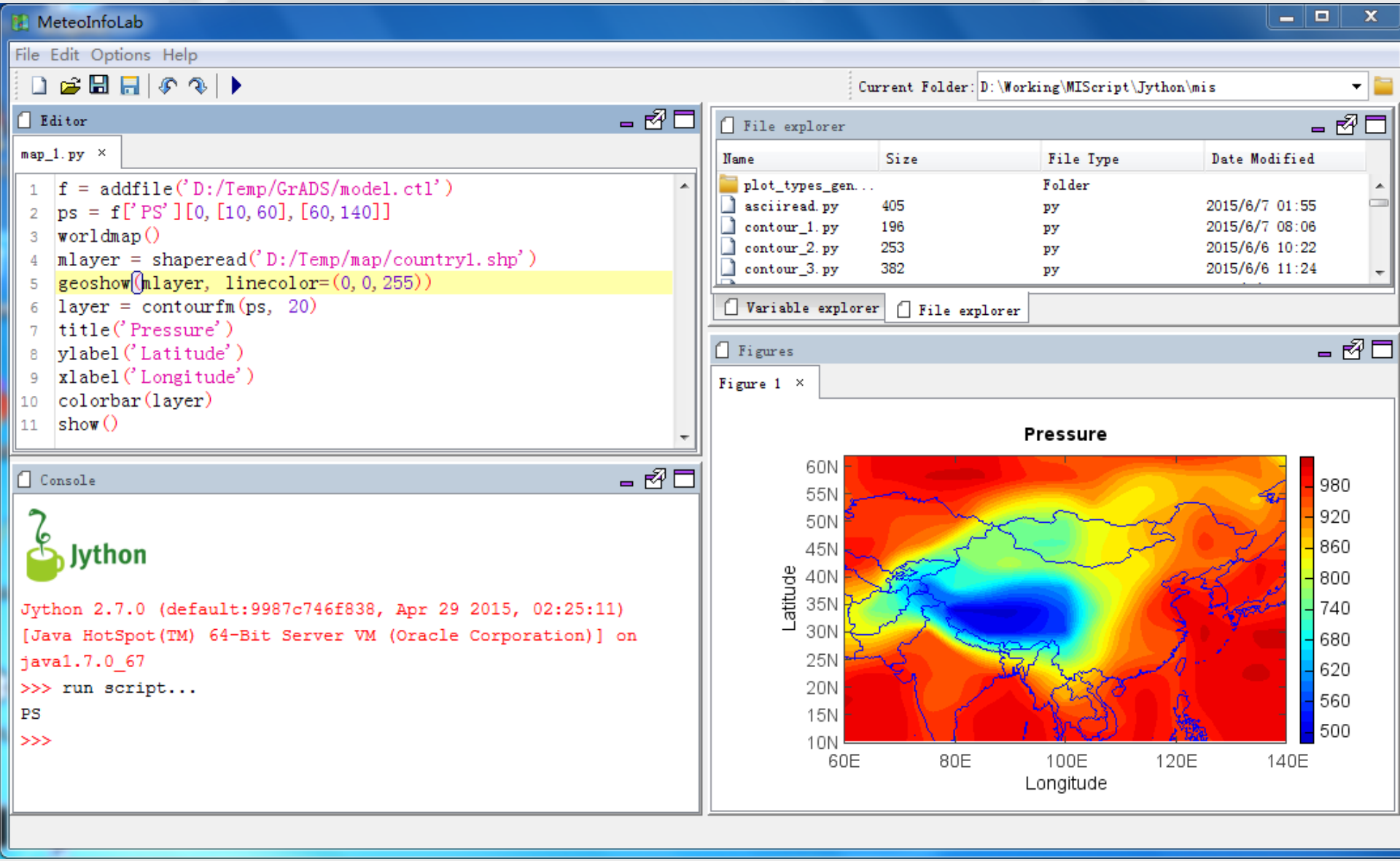




MeteoInfoLab (MeteoInfo Laboratory)

- Scientific data analysis and visualization for interactive command line and script.
- Developed using Java and Jython (Python branch) based on MeteoInfo library.
- Similar commands with MatLab (Matplotlib and numpy for Python) and NCL.
- Powered by Unidata Java netCDF library.
- Still in developing…

MeteoInfoLab



Sample for netCDF data

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

contour_2.py

```
1  ncf = addfile('D:/Temp/nc/cone.nc')
2  var = ncf['u']
3  udata = var[4, :, :]
4  levs = arange(0, 10, 1)
5  print levs
6  layer = contourf(udata, levs, cmap='BlAqGrYeOrRe')
7  title('Cone amplitude')
8  ylabel('Y')
9  xlabel('X')
10 colorbar(layer)
11 show()
12 print 'Finish.'
```

File explorer

Name	Size	File Type	Date Modified
asciiread.py	405	py	2015/6/7 01:55
contour_1.py	196	py	2015/6/7 08:06
contour_2.py	253	py	2015/6/6 10:22
contour_3.py	382	py	2015/6/6 11:24
cru_1.py	301	py	2015/6/7 10:47
FY2E.py	537	py	2015/6/7 10:47
geotiff.py	417	py	2015/6/7 11:44
hysplit_conc.py	330	py	2015/6/7 10:49

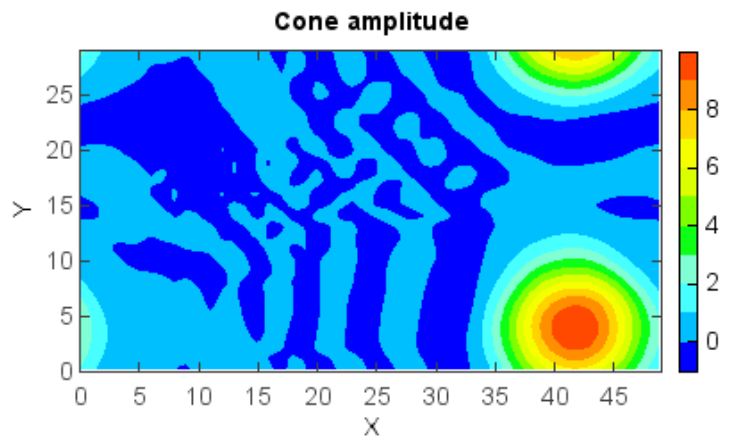
Variable explorer | File explorer

Console

```
[jar:file:/D:/MyProgram/Distribution/Java/Library/log4j-slf4j-i
SLF4J: Found binding in
[jar:file:/D:/MyProgram/java/MeteoInfoDev/MeteoInfoLib/dist/lib
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings fo
SLF4J: Actual binding is of type [org.slf4j.helpers.Log4jLogger
u
>>> run script...
u
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
Finish.
```

Figures

Figure 1



Sample for ASCII data

http://cdc.nmic.cn/sksj.do?method=ssrjscp

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

asciiread.py

```
1  rain = asciiread('D:/Temp/ascii/SURF_CLI_CHN_PRE_DAY_GRID_(
2  rain1 = rain[::-1,:]
3  x = arange1(72, 128, 0.5)
4  y = arange1(18, 72, 0.5)
5  worldmap()
6  mlayer = shaperead('D:/Temp/map/country1.shp')
7  geoshow(mlayer)
8  layer = contourfm(x, y, rain1, 20)
9  #layer = imshowm(x, y, rain1, 20)
10 title('Precipitation')
11 ylabel('Latitude')
12 xlabel('Longitude')
13 colorbar(layer)
14 show()
```

Console

```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
Finish.
>>> run script...
Varible count: 4
u
u dimension size: 3
>>> run script...
>>>
```

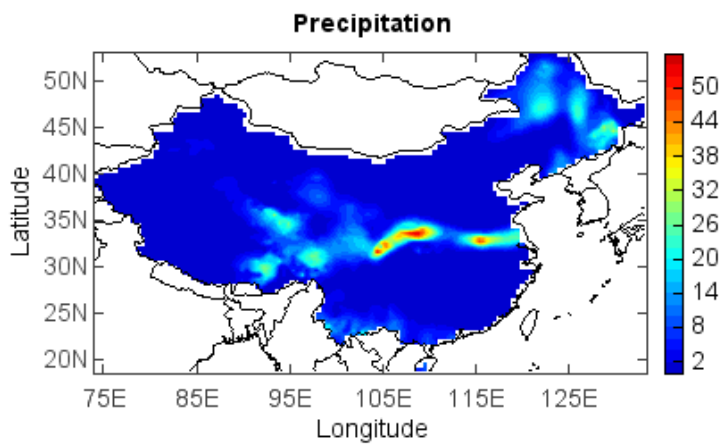
File explorer

Name	Size	File Type	Date Modified
plot_types_ge...		Folder	
asciiread.py	405	py	2015/6/7 01:55
contour_1.py	196	py	2015/6/7 08:06
contour_2.py	253	py	2015/6/6 10:22
contour_3.py	382	py	2015/6/6 11:24
cru_1.py	301	py	2015/6/7 10:47
FY2E.py	537	py	2015/6/7 10:47

Variable explorer | File explorer

Figures

Figure 1



Sample for geotiff data

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

geotiff.py

```
1  #Add data file
2  fn = 'D:/Temp/test/sample/ma040831.tif'
3  f = addfile(fn)
4  #Get data variable
5  v = f['var']
6  #Get data array
7  tbb = v[:,:]
8  #Plot
9  ax = worldmap()
10 proj = axesm(projinfo=f.proj, gridlabel=False)
11 clayer = shaperead('D:/Temp/map/bou2_4p.shp')
12 wlayer = shaperead('D:/Temp/map/country1.shp')
13 geoshow(clayer, linecolor=(0,0,255))
14 geoshow(wlayer)
15 layer = imshowm(tbb, proj=proj)
16 colorbar(layer)
17 show()
```

Console

```
The file does not exist: D:/Temp/Hdf/FY2E_TBB_IR1_NOM_20150126_
Traceback (most recent call last):
  File "<iostream>", line 5, in <module>
TypeError: 'NoneType' object is unsubscriptable
>>> run script...
var
```

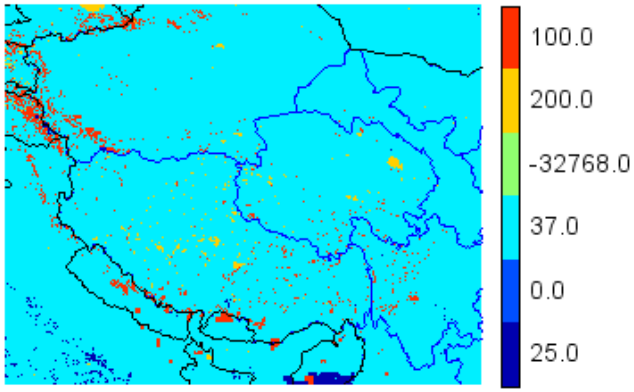
File explorer

Name	Size	File Type	Date Modified
asciiread.py	405	py	2015/6/7 01:55
contour_1.py	196	py	2015/6/7 08:06
contour_2.py	253	py	2015/6/6 10:22
contour_3.py	382	py	2015/6/6 11:24
cru_1.py	301	py	2015/6/7 10:47
FY2E.py	537	py	2015/6/7 10:47
geotiff.py	417	py	2015/6/7 11:44
hysplit_conc.py	330	py	2015/6/7 10:49

Variable explorer | File explorer

Figures

Figure 1



Sample for maskout

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

maskout.py

```
1  f = addfile('D:/Temp/GrADS/model.ctl')
2  ps = f['PS'][0,(10,60),(60,140)]
3  ax = worldmap()
4  m_china = shaperead('D:/Temp/map/china.shp')
5  m_world = shaperead('D:/Temp/map/country1.shp')
6  geoshow(m_china, visible=False)
7  geoshow(m_world, linecolor=(0,0,255))
8  layer = contourfm(ps, 12)
9  figmask(layer, m_china)
10 title('Pressure')
11 ylabel('Latitude')
12 xlabel('Longitude')
13 colorbar(layer)
14 show()
```

File explorer

Name	Size	File Type	Date Modified
map_3.py	236	py	2015/6/7 10:50
maskout.py	393	py	2015/6/7 10:51
micaps_extract.py	1kb	py	2015/5/14 09:58
micaps_extract1.py	1kb	py	2015/5/14 04:43

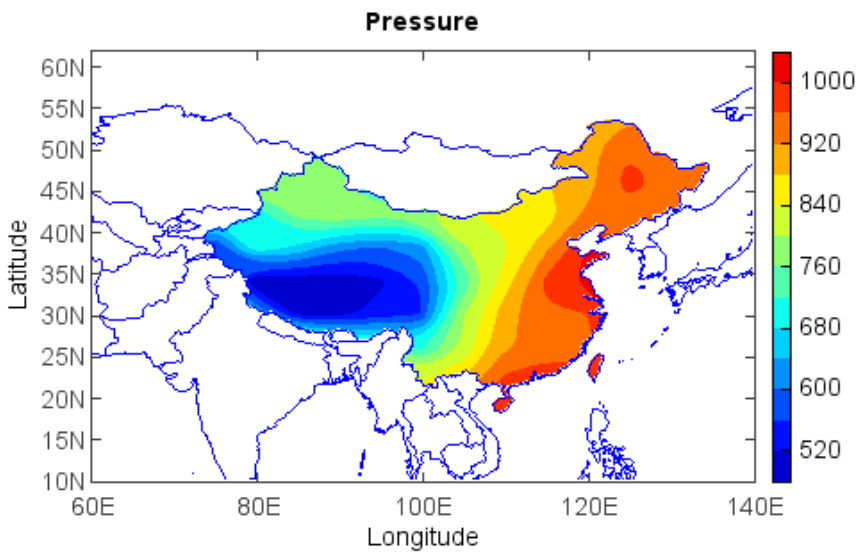
Variable explorer | File explorer

Console

```
>>> run script...
PRSS
>>> run script...
>>> run script...
PS
>>>
```

Figures

Figure 1



Sample for wind field

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

quiver_1.py

```
1  f = addfile('D:/Temp/GrADS/model.ctl')
2  u = f['U'][0,[500],[10,60],[60,140]]
3  v = f['V'][0,[500],[10,60],[60,140]]
4  speed = sqrt(u*u+v*v)
5  ax = worldmap()
6  mlayer = shaperead('D:/Temp/map/country1.shp')
7  geoshow(mlayer)
8  layer = quiverm(u, v, speed, 10, size=8)
9  title('Wind field')
10 ylabel('Latitude')
11 xlabel('Longitude')
12 colorbar(layer)
13 show()
```

Console

```
U
V
>>> run script...
U
V
>>>
```

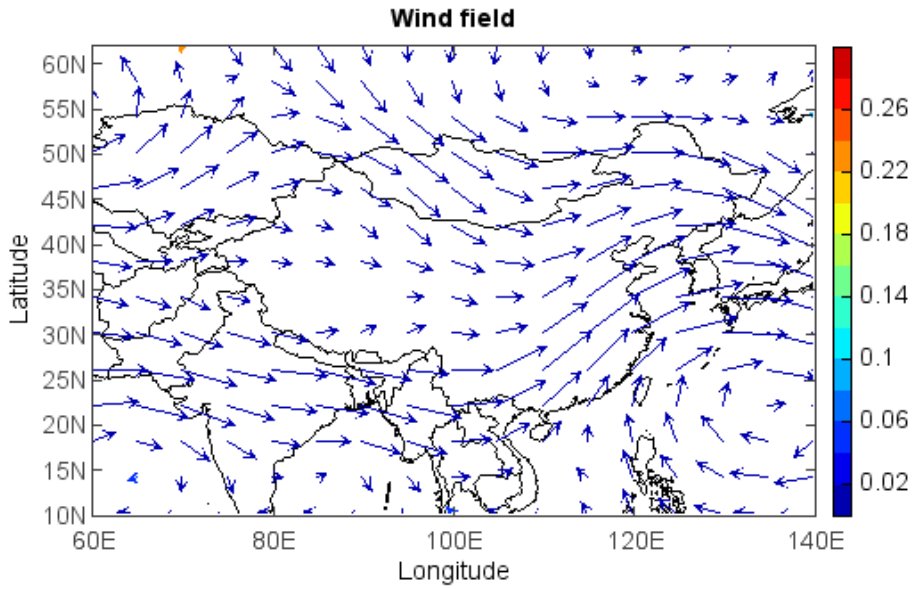
File explorer

Name	Size	File Type	Date Modified
polar_1.py	241	py	2015/6/7 10:52
quiver_1.py	388	py	2015/6/7 04:41
scatterm_1.py	277	py	2015/6/7 11:09

Variable explorer | File explorer

Figures

Figure 1



Sample for scatter

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

scatterm_1.py

```
1  f = addfile('D:/Temp/GrADS/model.ctl')
2  ps = f['PS'][0,(10,60),(60,140)]
3  worldmap()
4  mlayer = shaperead('D:/Temp/map/country1.shp')
5  geoshow(mlayer, linecolor=(0,0,255))
6  layer = scatterm(ps)
7  title('Pressure')
8  ylabel('Latitude')
9  xlabel('Longitude')
10 colorbar(layer)
11 show()
```

Console

```
PS
>>> run script...
PS
>>> run script...
PS
>>>
```

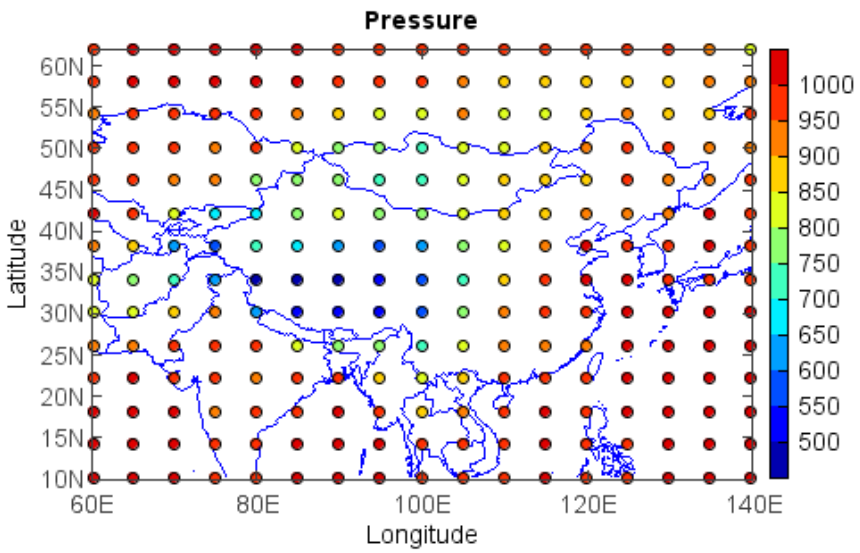
File explorer

Name	Size	File Type	Date Modified
scatterm_2.py	467	py	2015/6/7 11:10
scatterm_3.py	309	py	2015/6/7 11:10
scatterm_4.py	395	py	2015/6/7 11:11
seaice.py	345	py	2015/6/7 04:40

Variable explorer | File explorer

Figures

Figure 1



Sample for LaTeX

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\chart

Editor

latex_1.py

```
1  def  f(x, c):
2      m1 = sin(2*pi*x)
3      m2 = exp(-c*x)
4      return m1 * m2
5
6  x = linspace(0, 4, 100)
7  sigma = 0.5
8  plot(x, f(x, sigma), 'r', linewidth=2)
9  xlabel(r'$\rm{time} \ t$', fontsize=16)
10 ylabel(r'$\rm{Amplitude} \ f(x)$', fontsize=16)
11 title(r'$f(x) \ \rm{is \ damping \ with} \ x$', fontsize=16)
12 text(2.0, 0.5, r'$f(x) = \rm{sin}(2 \pi x^2) e^{\sigma x}$',
13 show()
```

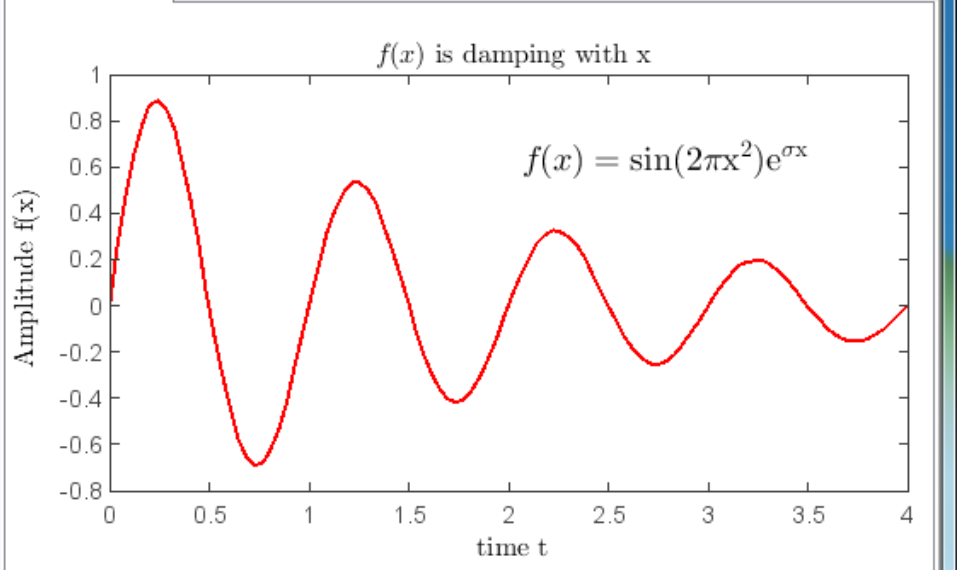
File explorer

Name	Size	File Type	Date Modified
chart_4.py	316	py	2015/4/24 11:11
chart_5.py	288	py	2015/4/24 11:12
chart_6.py	417	py	2015/4/15 02:54
chart_7.py	239	py	2015/6/1 11:58
get_runtim...	288	py	2015/4/18 06:06

Variable explorer | File explorer

Figures

Figure 1



Console

Jython

```
Jython 2.7b3 (default:e81256215fb0, Aug 4 2014, 02:39:51)
[Java HotSpot(TM) 64-Bit Server VM (Oracle Corporation)] on
java1.7.0_67
>>> run script...
Series index: 0
>>>
```

Sample for table data

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\mis

Editor

table-1.py

```
1  #Set directories
2  datadir = 'D:/KeyData/PMMUL/data'
3
4  #Set stations
5  stid = '54857'
6  station = 'Qingdao'
7
8  #Read data
9  fn = os.path.join(datadir, stid + '_PMMUL_DA.csv')
10 if os.path.exists(fn):
11     print fn
12     tsData = readtable(fn, delimiter=',')
13     d = tsData.coldata('PM10')
14     #print d
15     plot(d)
16     ylabel(r'$\rm{PM_{10}} \ (\mu g \ m^{-3})$')
17     show()
18
19 print 'Finished...'
```

Console

```
D:/KeyData/PMMUL/data\54857_PMMUL_DA.csv
PM10
Finished...
>>>
```

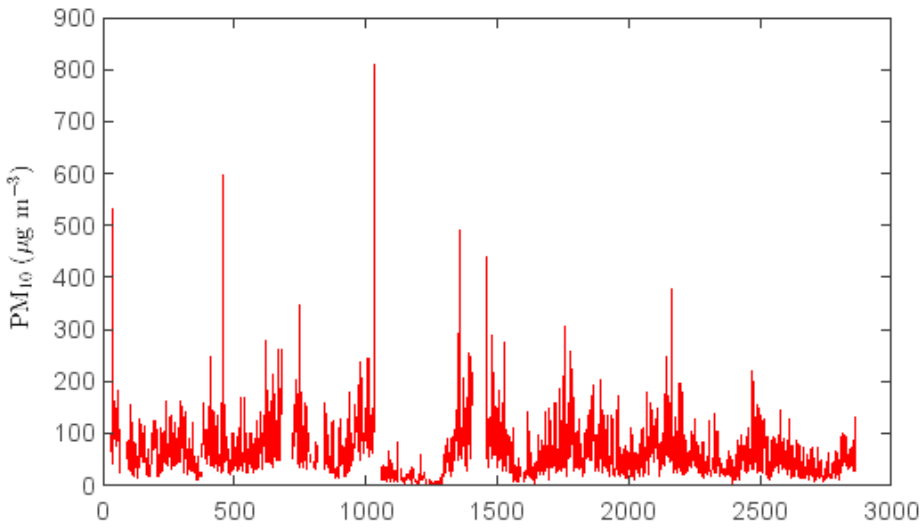
File explorer

Name	Size	File Type	Date Modified
seaice.py	345	py	2015/6/7 04:40
surfer_data.py	286	py	2015/6/7 11:11
table-1.py	346	py	2015/6/7 11:11
table.py	590	py	2015/5/15 11:19
test_1.py	152	py	2014/12/28 09:09

Variable explorer | File explorer

Figures

Figure 1



Sample for sub plot

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\chart

Editor

chart_4.py

```
1  def f(t):
2      return exp(-t) * cos(2*pi*t)
3
4  t1 = arange(0., 5., 0.1)
5  t2 = arange(0., 5., 0.02)
6
7  subplot(2,1,1)
8  plot(t1, f(t1), 'bo', t2, f(t2), 'k')
9
10 subplot(2,1,2)
11 plot(t2, cos(2*pi*t2), 'r--')
12 show()
```

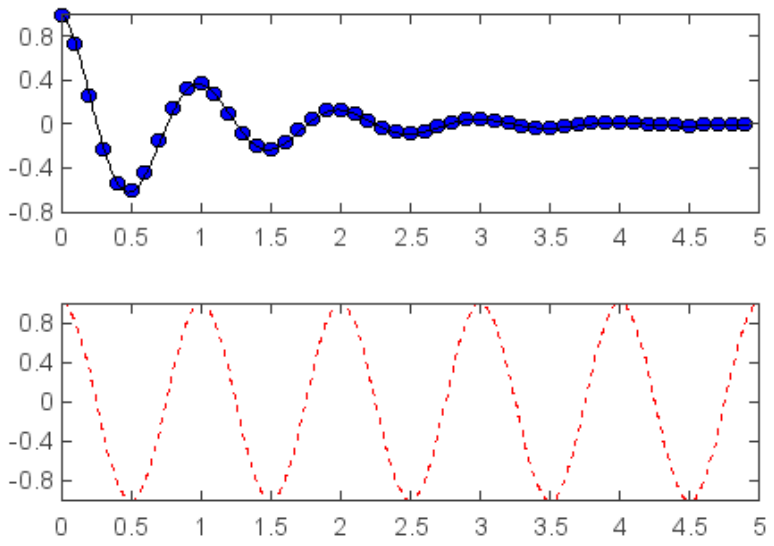
File explorer

Name	Size	File Type	Date Modified
chart_1.py	197	py	2015/6/7 08:56
chart_2.py	114	py	2015/6/6 10:00
chart_3.py	104	py	2015/6/6 09:45
chart_4.py	209	py	2015/6/7 08:56
chart_5.py	173	py	2015/6/7 08:56

Variable explorer | File explorer

Figures

Figure 1



Console

```
Series index: 0
Series index: 1
Series index: 0
>>>
```

Sample for double y axis

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\chart

Editor

yy_1.py × yyy_1.py ×

```
1  ax1 = axes()
2  yaxis(ax1, color='b')
3  t = arange(0.01, 10.0, 0.01)
4  s1 = exp(t)
5  plot(t, s1, 'b-', linewidth=2)
6  xlabel('time (s)')
7  ylabel('exp', color='b')
8
9  ax2 = twinx(ax1)
10 yaxis(ax2, color='r')
11 s2 = sin(2*pi*t)
12 plot(t, s2, 'r.')
13 ylabel('sin', color='r')
14 show()
```

Console

```
>>> run script...
Series index: 0
Series index: 0
>>>
```

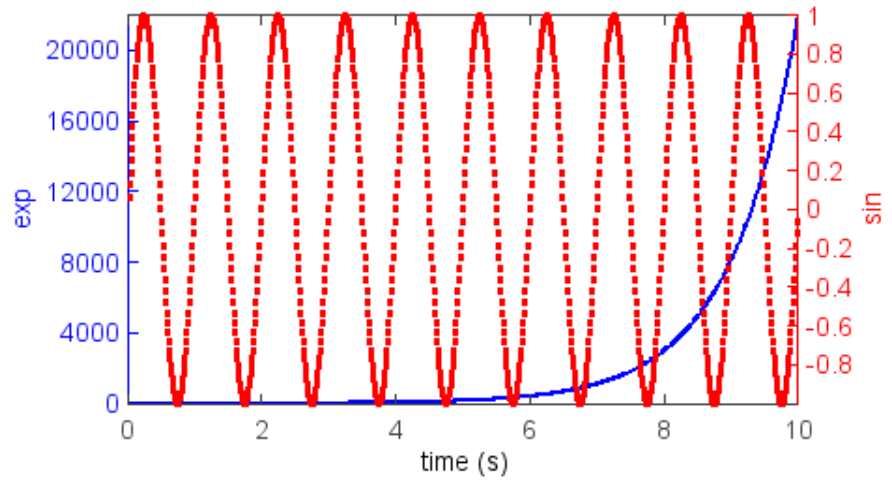
File explorer

Name	Size	File Type	Date Modified
test.py	136	py	2015/4/6 07:50
test1.py	198	py	2015/4/14 11:59
yy_1.py	224	py	2015/6/6 07:33
yyy_1.py	546	py	2015/6/7 08:56

Variable explorer | File explorer

Figures

Figure 1 ×



Sample for multiple y axis

MeteoInfoLab

File Edit Options Help

Current Folder: D:\Working\MIScript\Jython\chart

Editor

yy_1.py × yyy_1.py ×

```
1  ax1 = axes(position=[0.113, 0.15, 0.7, 0.8])
2  yaxis(ax1, color='b')
3  line1 = plot([0, 1, 2], [0, 1, 2], 'b-', label="D
4  xlabel('Distance')
5  ylabel('Density', color='b')
6  title('Mutiple Y Axis Sample')
7
8  ax2 = twinx(ax1)
9  yaxis(ax2, color='r')
10 line2 = plot([0, 1, 2], [0, 3, 2], 'r-', label="T
11 ylabel('Temperature', color='r')
12
13 ax3 = twinx(ax1)
14 yaxis(ax3, shift=60, color='g')
15 line3 = plot([0, 1, 2], [50, 30, 15], 'g-', label
16 ylabel('Velocity', color='g')
17 lines = [line1, line2, line3]
18 legend(lines)
19 show()
```

Console

```
>>> antialias(True)
>>>
```

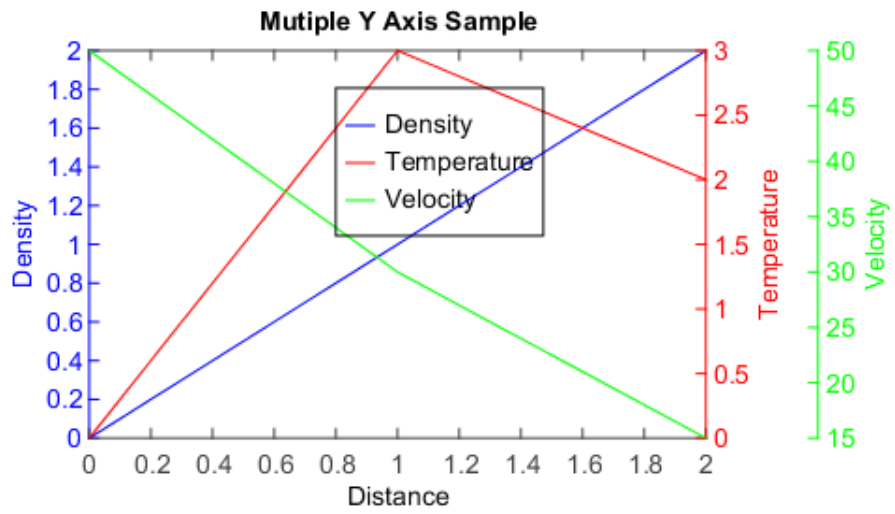
File explorer

Name	Size	File Type	Date Modified
test.py	136	py	2015/4/6 07:50
test1.py	198	py	2015/4/14 11:59
yy_1.py	224	py	2015/6/6 07:33
yyy_1.py	546	py	2015/6/7 08:56

Variable explorer | File explorer

Figures

Figure 1 ×



Sample for CALIPSO data plot

MeteoInfoLab

File Edit Options Apps Help

Current Folder: D:\Working\MIScript\Jython\mis\hdf

Editor

CALIPSO_L1.py

```
1  sys.path.append('D:/Working/MIScript/Jython/mis/hdf')
2  import CALIPSO_colors
3
4  # Add file
5  fn = 'D:/Temp/hdf/CAL_LID_L1-ValStage1-V3-01.2007-06-12T03-42-18ZN.hdf'
6  f = addfile(fn)
7
8  # Read data
9  vname = 'Total_Attenuated_Backscatter_532'
10 var = f[vname]
11 data = var[:1000,:]
12 #data = var[:,:]
13 data = rot90(data)
14 lats = f['latitude'][:1000,0]
15 #lats = f['latitude'][:,0]
16 latstrs = []
17 for lat in lats:
18     latstrs.append('%.1f' % lat)
19 height = f['metadata']['Lidar_Data_Altitudes']
```

Console

Jython

```
Jython 2.7.0 (default:9987c746f838, Apr 29 2015, 02:25:11)
[Java HotSpot(TM) 64-Bit Server VM (Oracle Corporation)] on java1.8.0_65
>>> run script...
>>> run script...
>>>
```

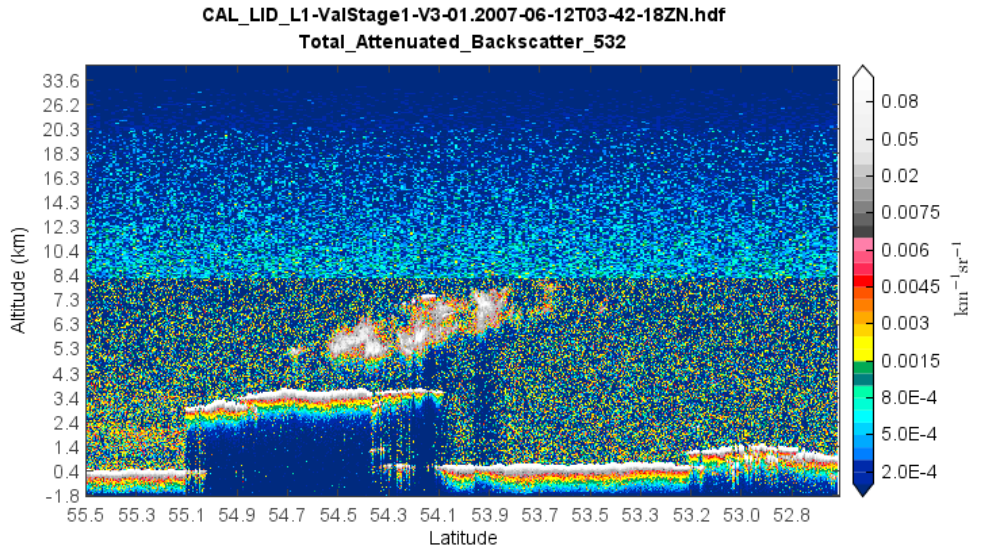
File explorer

Name	Size	File Type	Date Modified
AIRS_1.py	644	py	2016/2/18 09:35
AIRS_2.py	399	py	2015/12/27 03:33
AMSR_1.py	769	py	2016/2/18 09:35
AMSR_2.py	399	py	2016/1/3 12:36

Variable explorer | File explorer

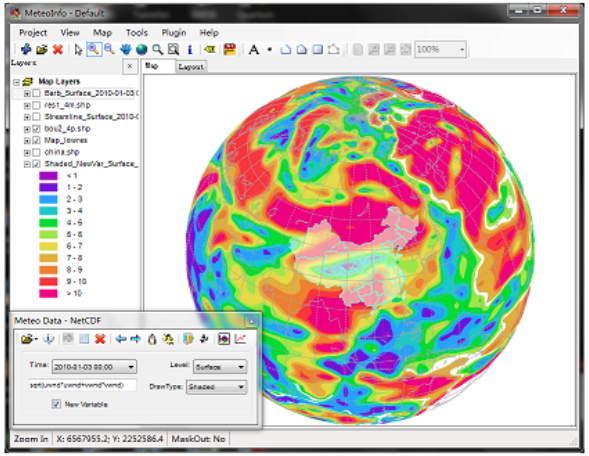
Figures

Figure 1

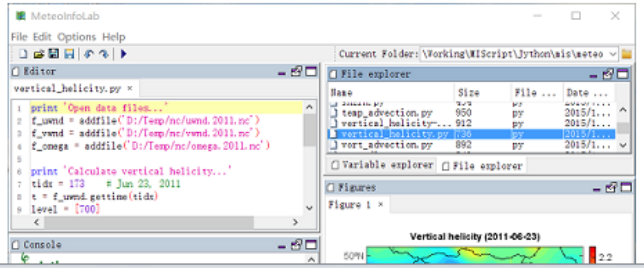


Introduction ¶

MeteoInfo is a freely available software designed to view and analyze meteorological and spatial data interactively. Some GIS functions were developed from ground level. It has two editions: Java and C#. MeteoInfo may be run in Windows, Mac OS, Linux and Unix. MeteoInfo can also be run automatically using MeteoInfo scripting with the IronPython language (C# edition) or Jython language (Java edition). The main functions are packed in the MeteoInfo class library, which could be used to conveniently develop the software.



MeteoInfoLab is a free software product developed using Java and Jython based on MeteoInfo Java library (Unidata netCDF Java library is used). The purpose is to provide an optional scientific computation and visualization tool similar with MatLab and/or NCL.



Quick search

Go

Enter search terms or a module, class or function name.

Website：http://www.meteothinker.com

Acknowledgement

- NetCDF Java: Available at www.unidata.ucar.edu/software/netcdf-java
- Jython: Available at http://www.jython.org/
- Proj4J: Available at http://trac.osgeo.org/proj4j/wiki
- Groovy: Available at http://groovy.codehaus.org/
- wContour: Available at http://www.meteothinker.com/
- L2FProd: Available at http://common.L2FProd.com
- RSyntaxTextArea: Available at http://fifesoft.com/rsyntaxtextarea/
- JLaTeXMath: Available at http://forge.scilab.org/index.php/p/jlatexmath/
- JavaHelp: Available at https://javahelp.java.net/
- BeanShell: Available at http://www.beanshell.org/
- FreeHEP VectorGraphics: Available at http://java.freehep.org/vectorgraphics/
- Docking Frames: Available at http://dock.javaforge.com/
- And more …

Thanks !

Welcome to use MeteoInfo !

http://www.meteothinker.com