

An Interactive Demonstration of moving from GEMPAK to MetPy as the Primary Analysis and Visualization Tool of Atmospheric Scientists

Kevin H. Goebbert<sup>1</sup>, Ryan M. May<sup>2</sup>, Zach S. Bruick<sup>2,3</sup>  
<sup>1</sup> Valparaiso University, <sup>2</sup> UCAR/Unidata, <sup>3</sup> McKinsey & Company ACRE

Declarative Attributes

Gridded Data

- data
- time
- field
- level

Contours

- linecolor
- linestyle
- linewidth
- clabels
- contours

Filled-Contours

- colormap
- colorbar
- contours
- clabels
- image\_range

Images

- colorbar
- colormap
- image\_range

Barbs

- barblength
- skip
- pivot
- color
- earth\_relative

Map Elements

- area
- layout
- layers
- plots
- projection
- title

Panel Elements

- panels
- size
- show
- save

Observations

- data
- fields
- time
- time\_window
- colors
- locations
- formats
- vector\_field
- vector\_field\_length
- vector\_field\_color
- reduce\_points

# Synoptic-Dynamic Maps Made **Easy** Using MetPy and Jupyter Notebooks

```
In [1]: import matplotlib.pyplot as plt
plt.rcParams['font.size'] = 16

In [2]: from datetime import datetime
import metpy.plots as mplots
from metpy.units import units
import xarray as xr

# Set date for desired dataset
date = datetime(2012, 10, 31, 12)

# Open dataset from NCEI
ds = xr.open_dataset('https://www.ncei.noaa.gov/thredds/dodsC/gfs-g4-anl-files/'
                    f'{date:%Y%m}(date:%Y%m%d)/gfsanl_4_{date:%Y%m%d}_{date:%H}00_000.grb2').metpy.parse_cf()

# Subset Data to be just over CONUS
ds_us = ds.sel(lon=slice(360-150, 360-50), lat=slice(65, 20))

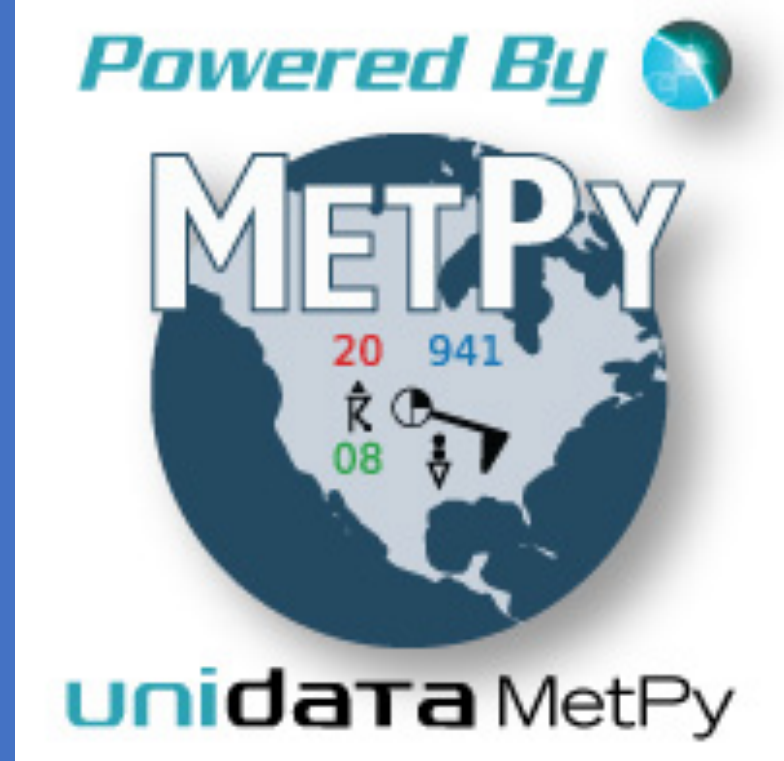
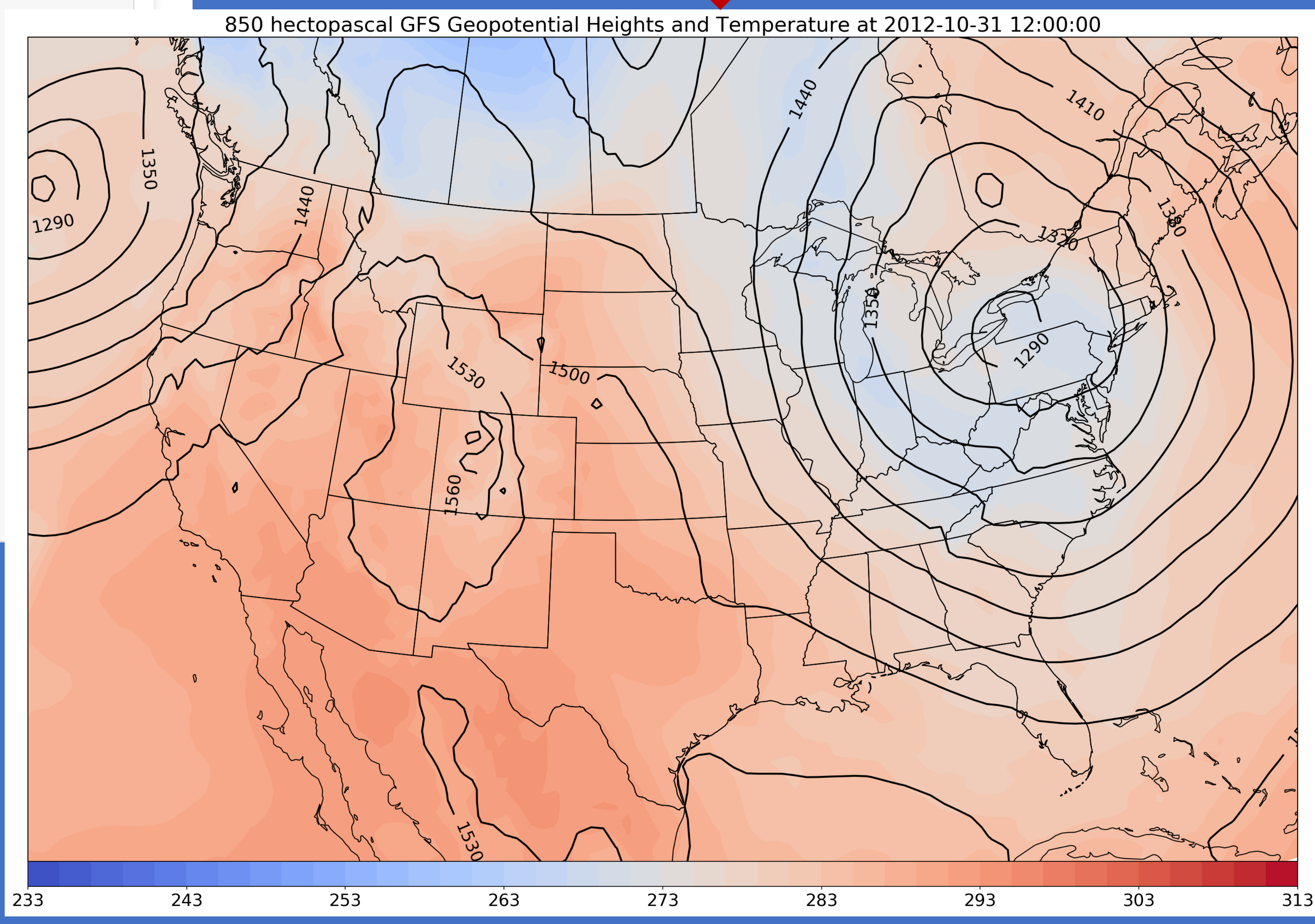
# Set Contour Plot Parameters
contour = mplots.ContourPlot()
contour.data = ds_us
contour.time = date
contour.field = 'Geopotential_height_isobaric'
contour.level = 850 * units.hPa
contour.linecolor = 'black'
contour.linestyle = '-'
contour.linewidth = 2
contour.clabels = True
contour.contours = list(range(0, 20000, 30))

# Set Color-filled Contour Parameters
cfill = mplots.FilledContourPlot()
cfill.data = ds_us
cfill.time = date
cfill.field = 'Temperature_isobaric'
cfill.level = 850 * units.hPa
cfill.contours = list(range(233, 314, 2))
cfill.colormap = 'coolwarm'
cfill.colorbar = 'horizontal'

# Panel for plot with Map features
panel = mplots.MapPanel()
panel.layout = (1, 1, 1)
panel.area = (-124, -72, 24, 53)
panel.projection = 'lcc'
panel.layers = ['coastlines', 'borders', 'states']
panel.title = f'{cfill.level} GFS Geopotential Heights and Temperature at {date}'
panel.plots = [cfill, contour]

# Bringing it all together
pc = mplots.PanelContainer()
pc.size = (24, 22)
pc.panels = [panel]

pc.save('850_hPa_declarative_example.png', dpi=150, bbox_inches='tight')
```



## Transition from GEMPAK to MetPy Declarative Syntax

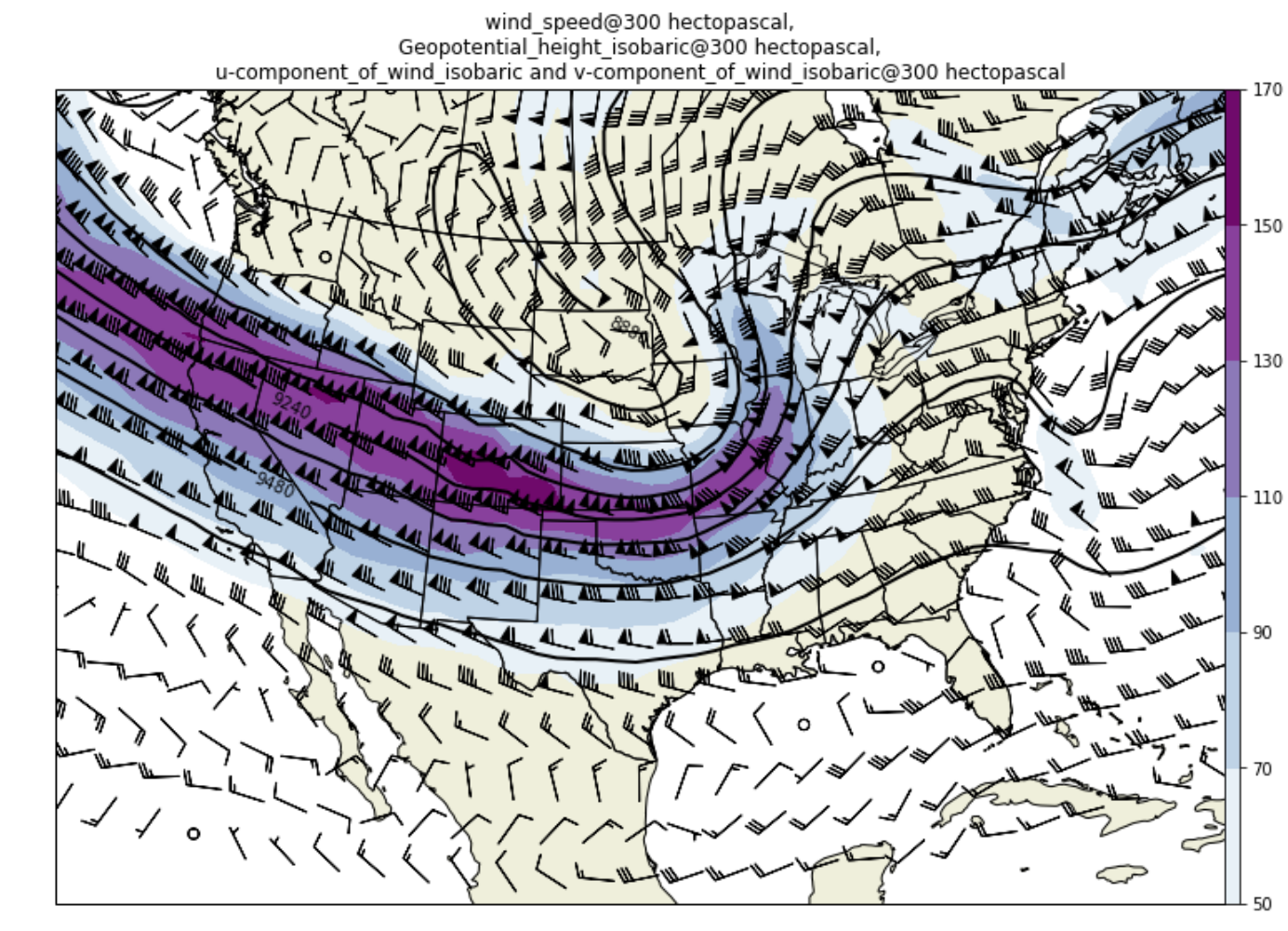
Request a Feature / Report a Bug



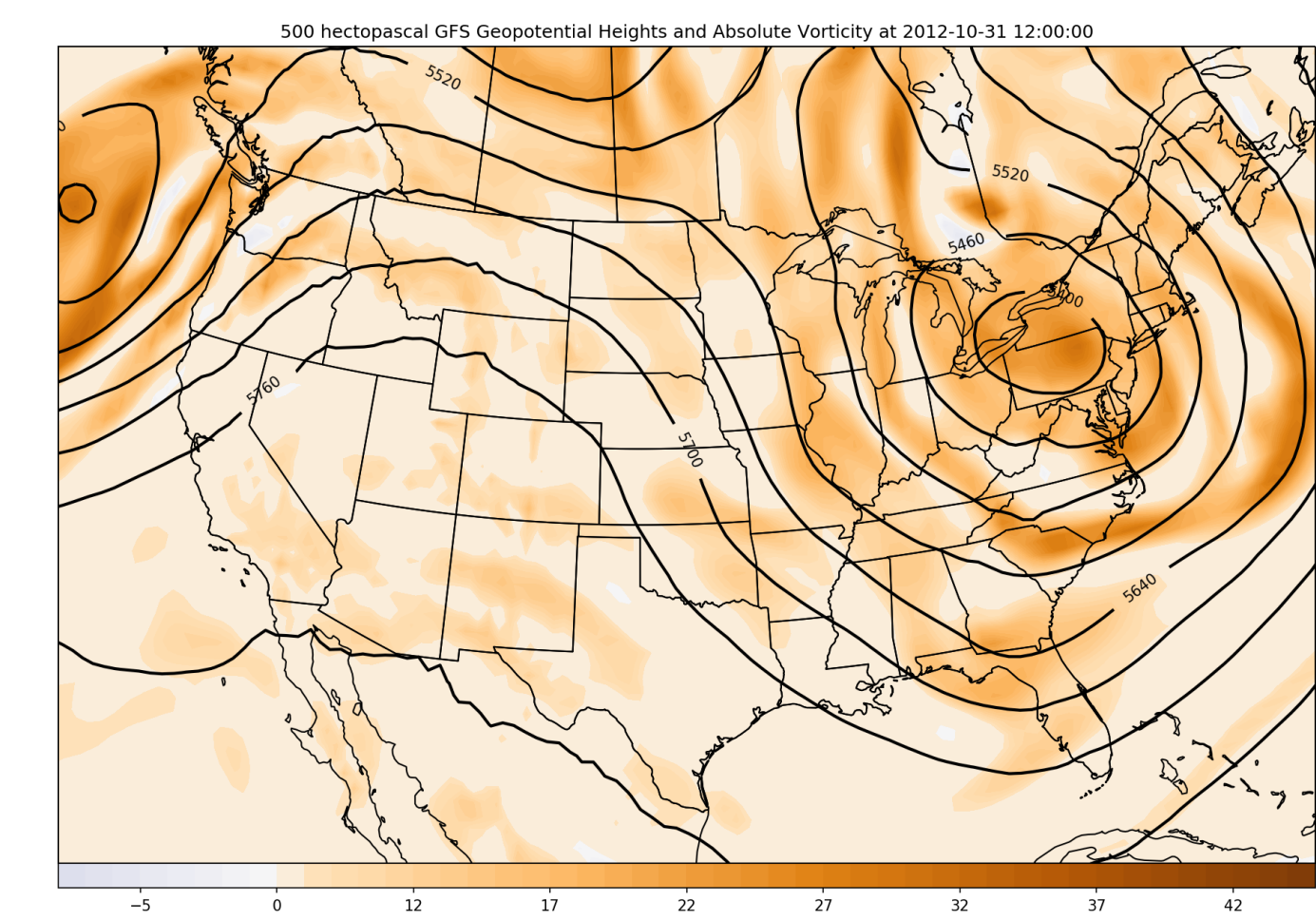
Submit a PR to Fix an Issue or add a Feature



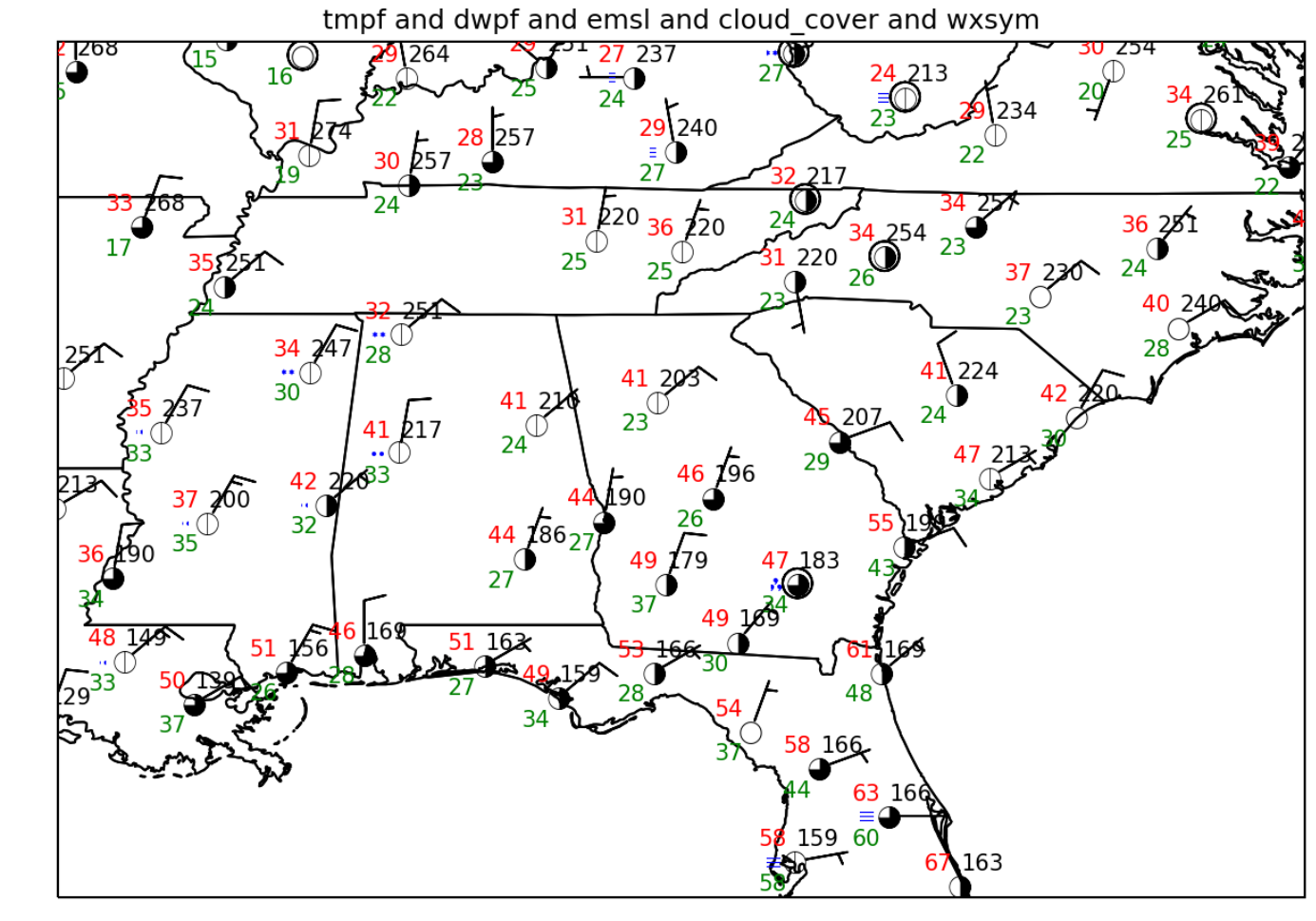
Examples



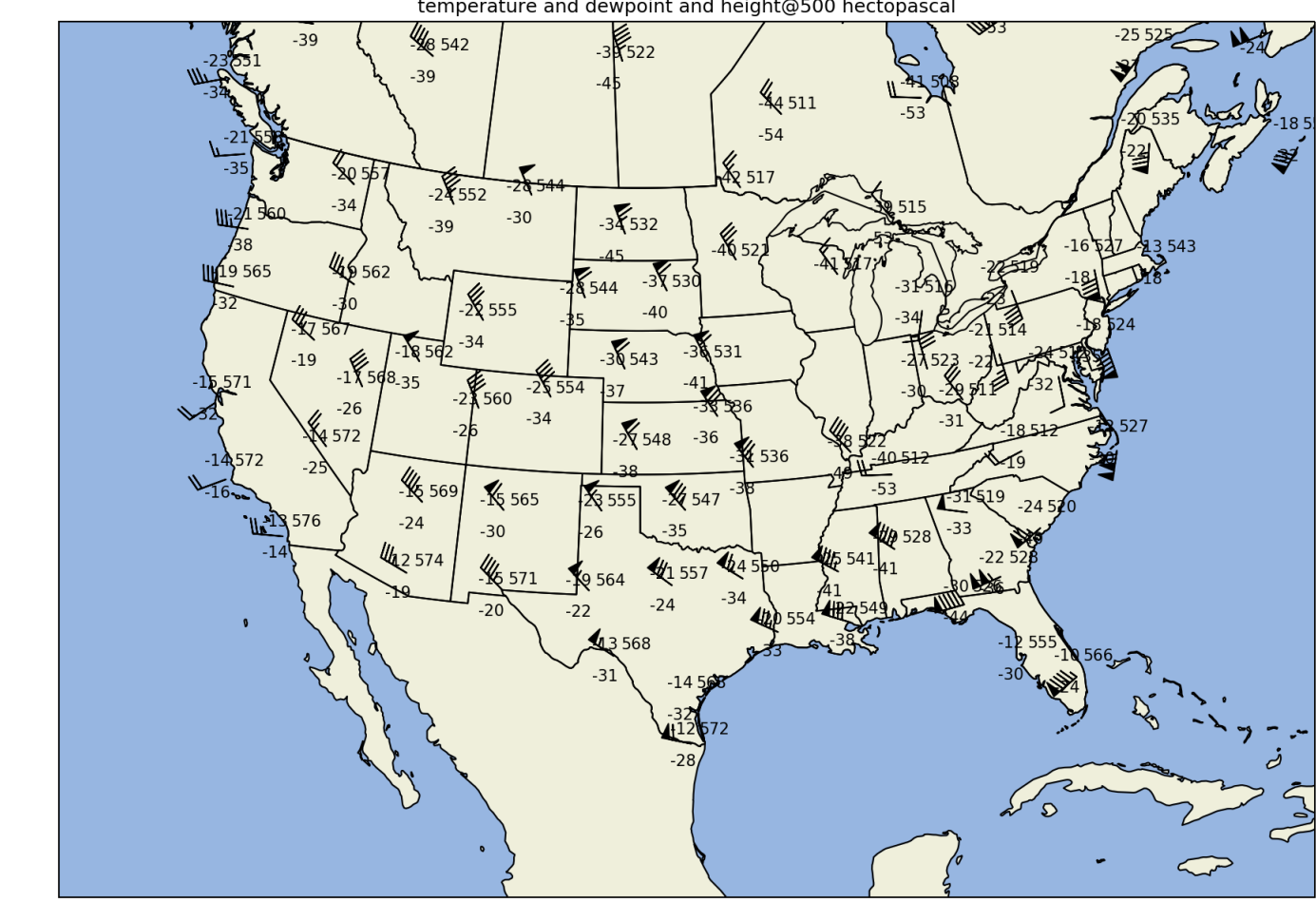
See Example



See Example



See Example



See Example



Funding Support: MetPy Development currently supported by Unidata (NSF-1344155) and an NSF grant (OAC-1740315)