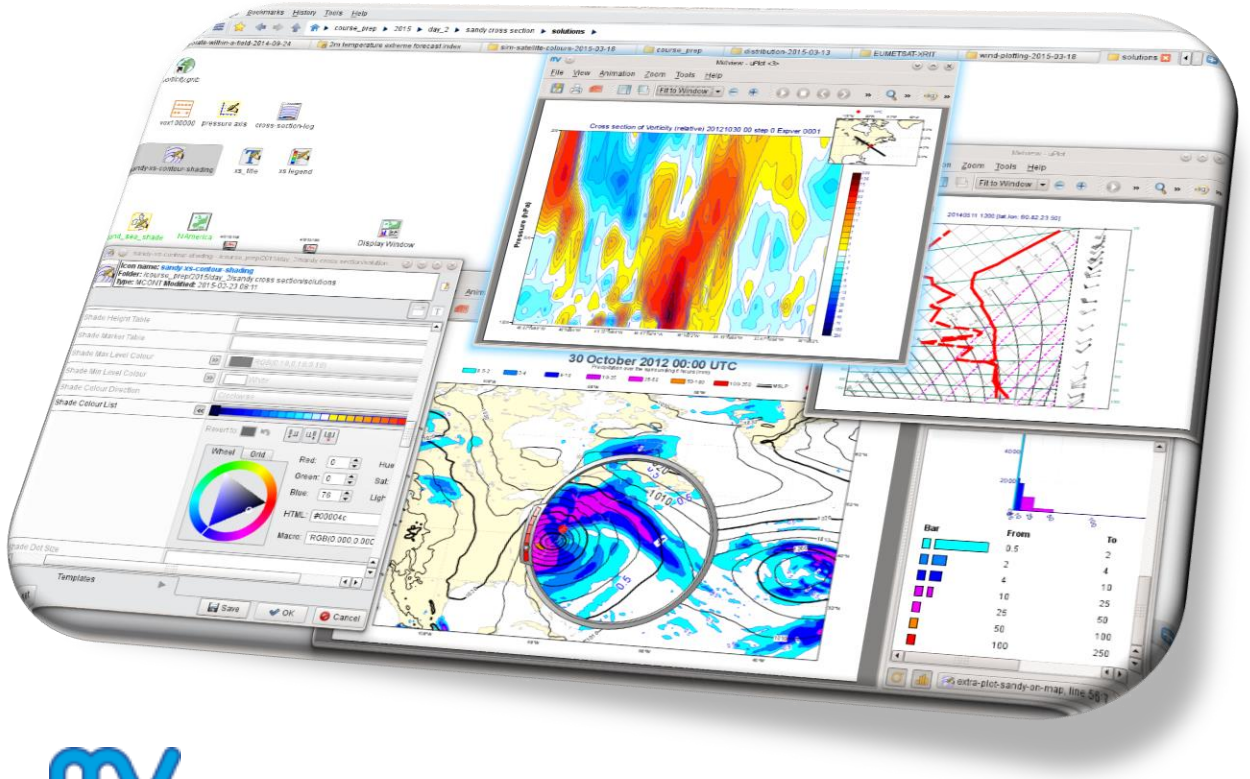
Metview Introduction

ECMWF

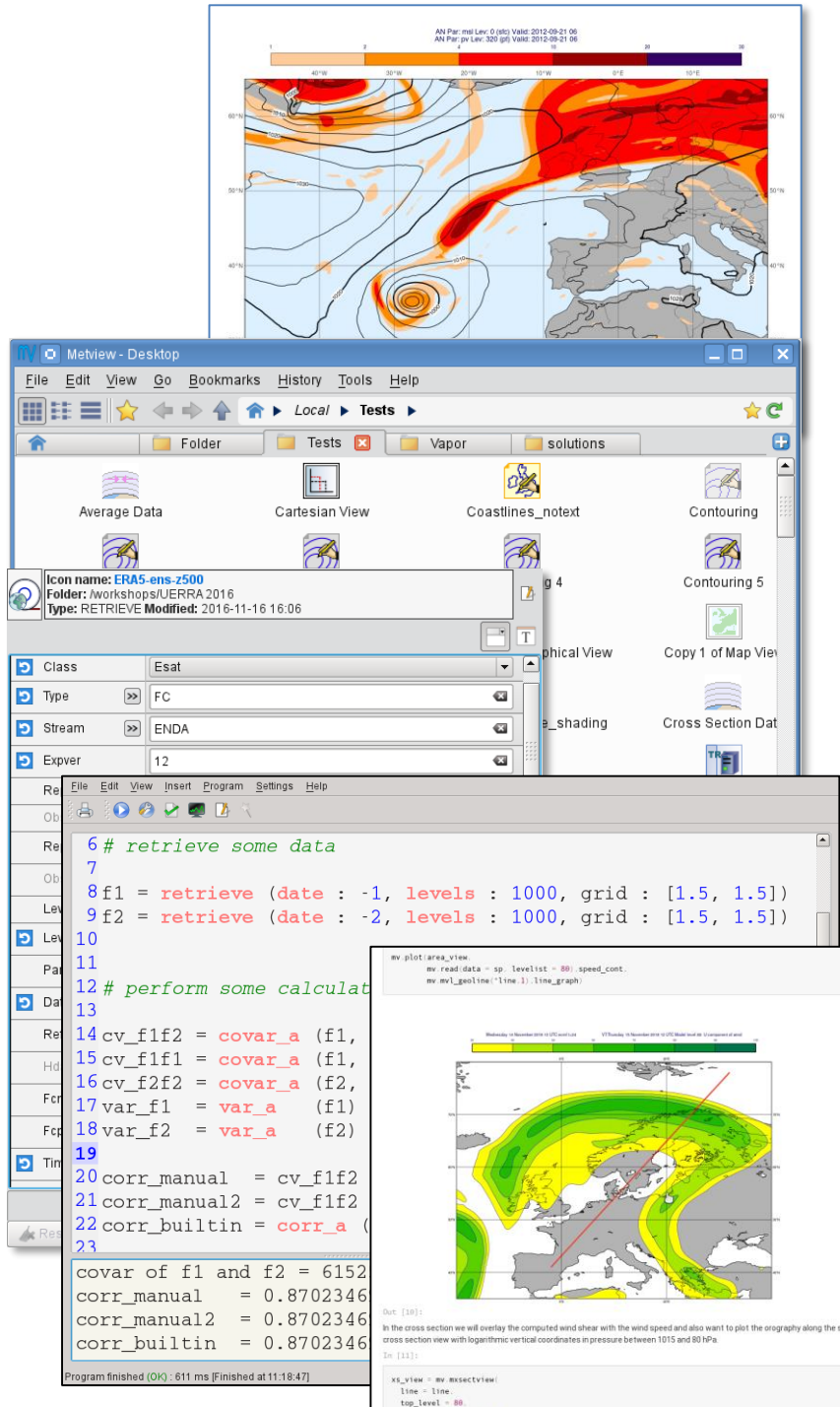
Iain Russell,
Sándor Kertész

Development Section, ECMWF

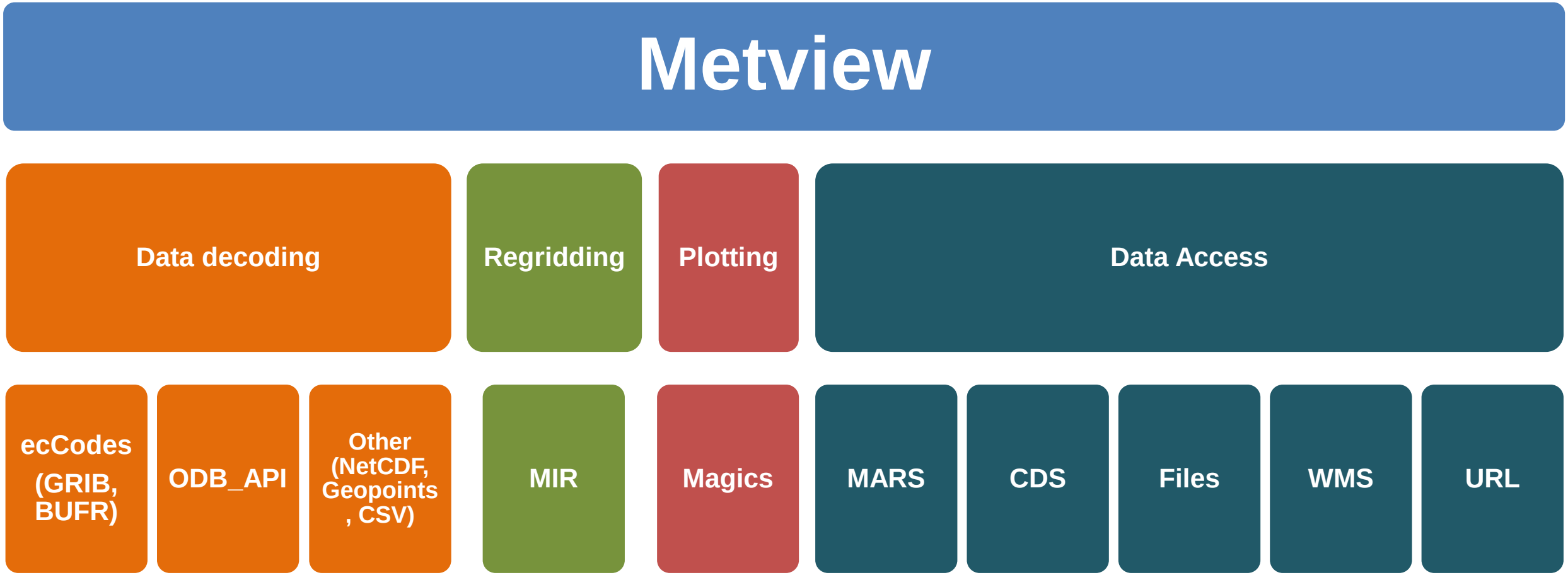


What is Metview?

- Workstation software for researchers and operational analysts
  - Runs on UNIX, from laptops to supercomputers (also being tested on Mac OS X)
- Retrieve/manipulate/visualise/examine meteorological data
- Can access MARS, either locally or through the Web API
- Open Source under Apache Licence 2.0
- Installed on ECMWF machines (module avail metview)
- Available via binary installations (RPM, DEB) and on conda
- Metview is a co-operation project with INPE (Brazil)

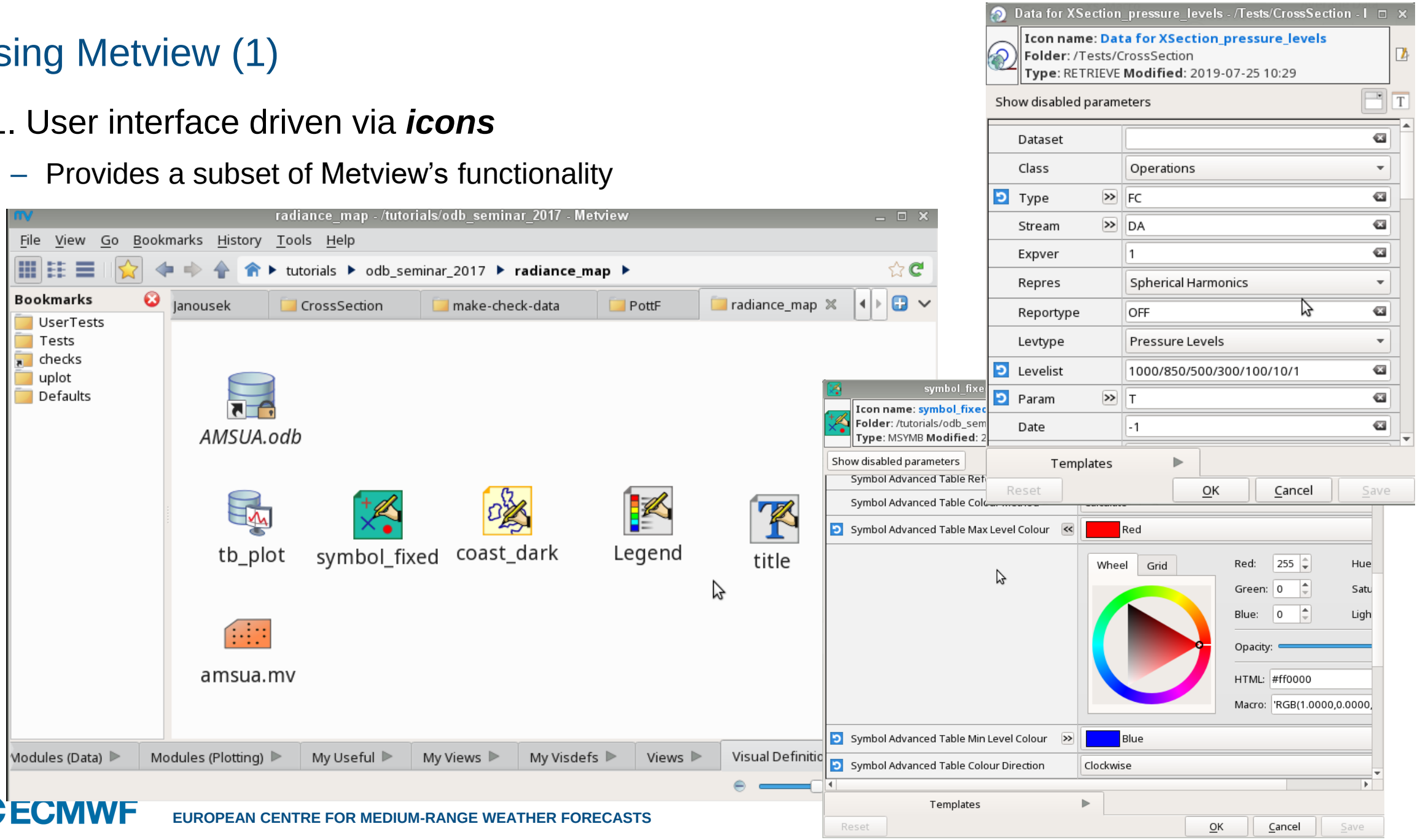


Built on top of ECMWF software packages



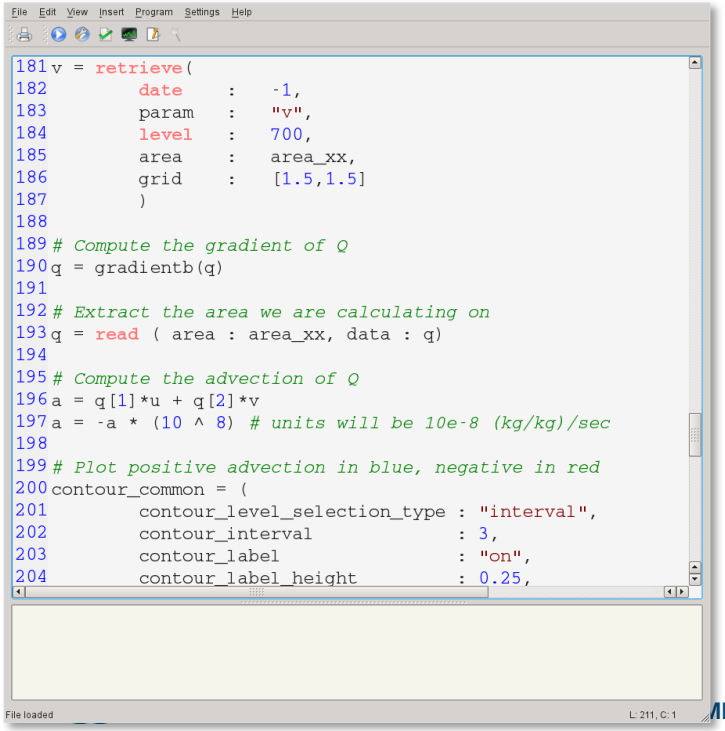
Using Metview (1)

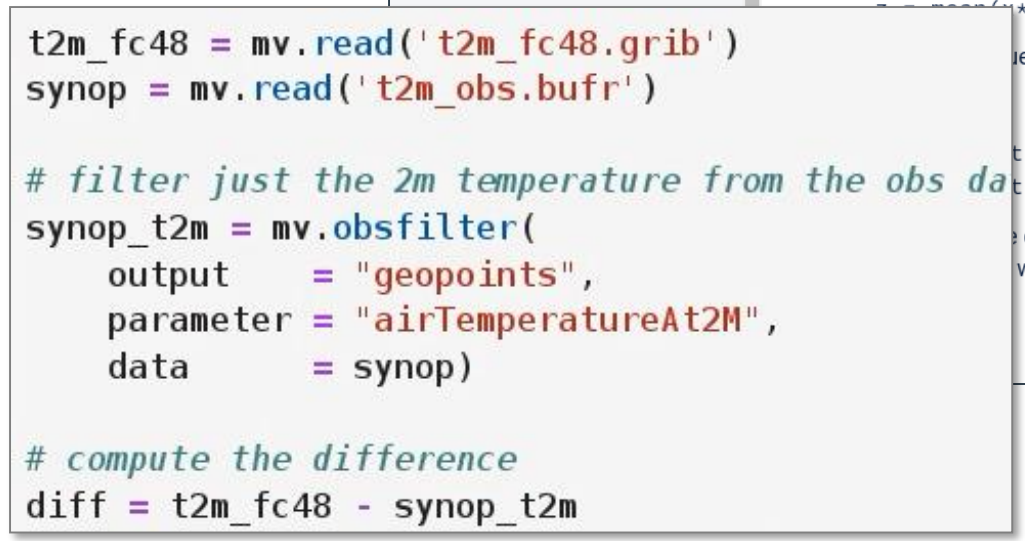
- 1. User interface driven via ***icons***
  - Provides a subset of Metview's functionality

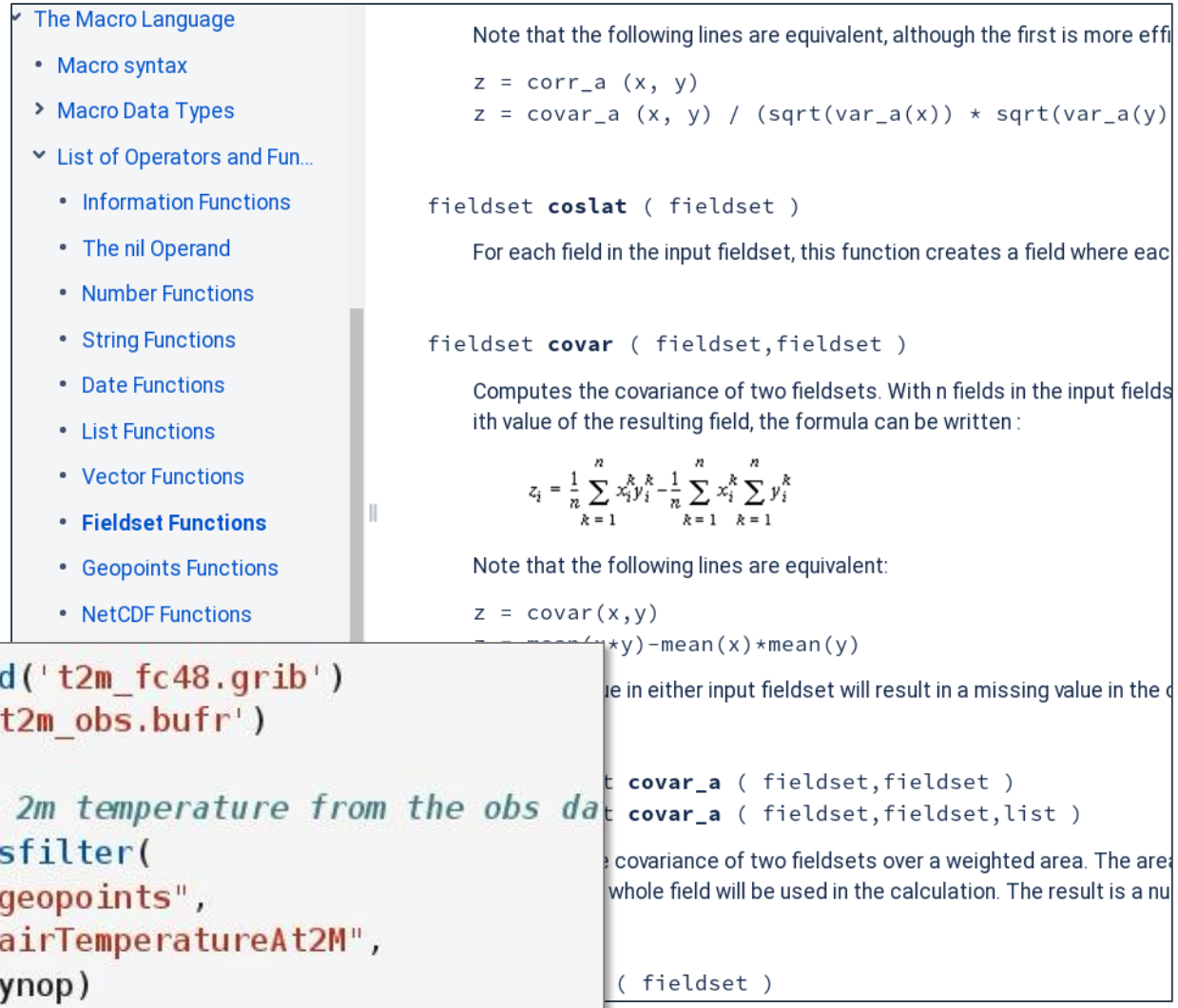


Using Metview (2)

- 2. Scripting (run from UI or command line)
  - Can do everything from icons, plus lots more
  - Macro language or Python interface – Python interface gives access to all Macro functions, and will gain more features and allows you to combine with other Python libraries and to work in Jupyter and other environment

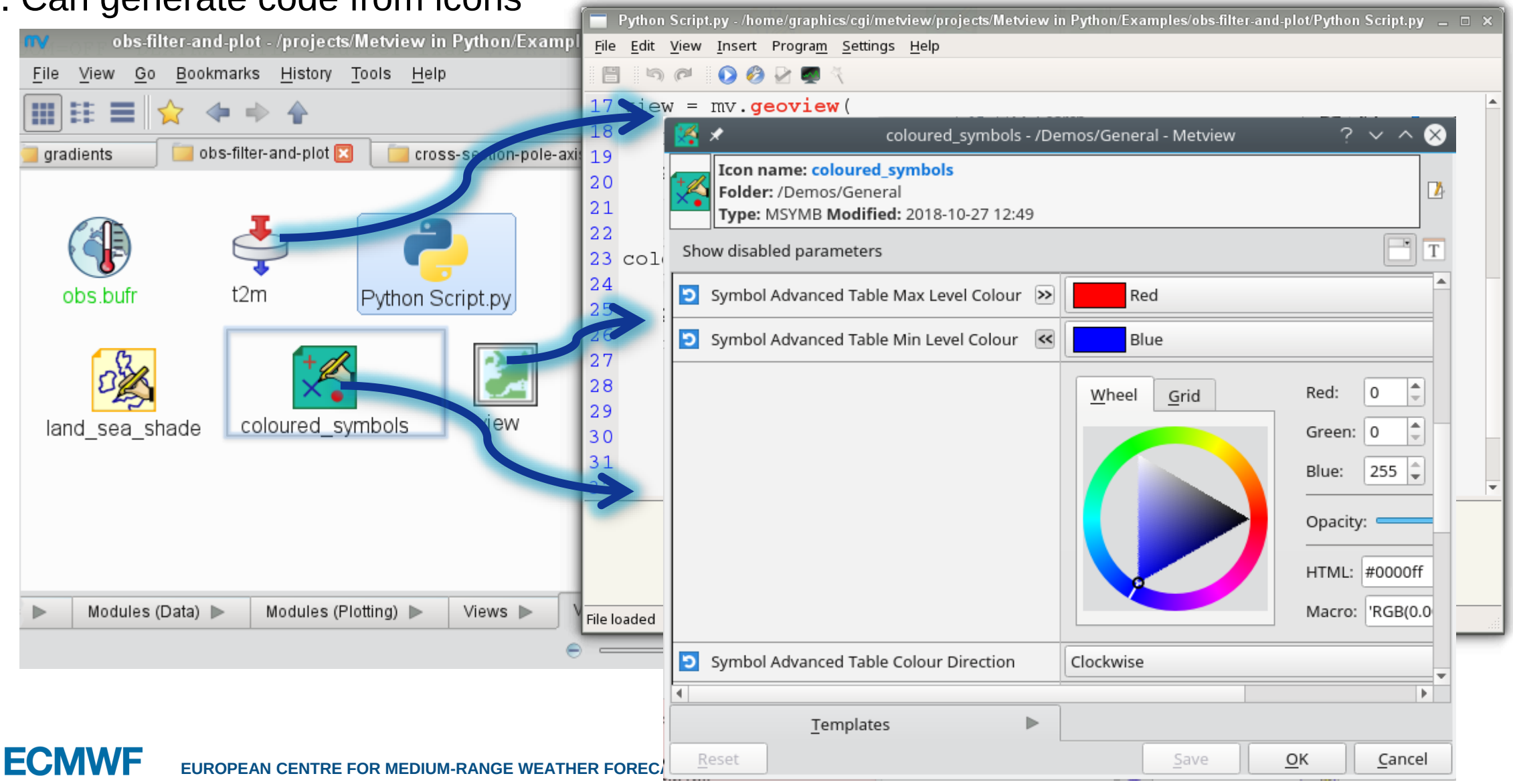






Using Metview (3)

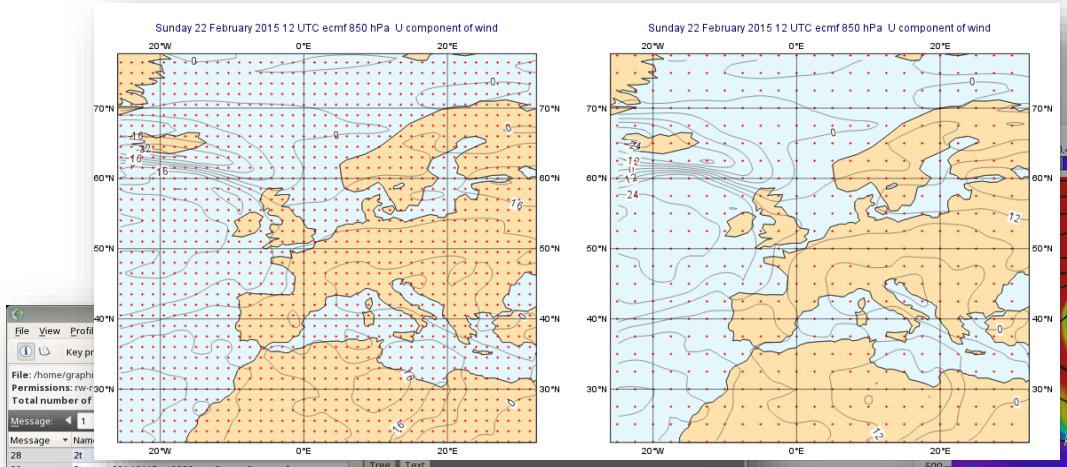
- 3. Can generate code from icons

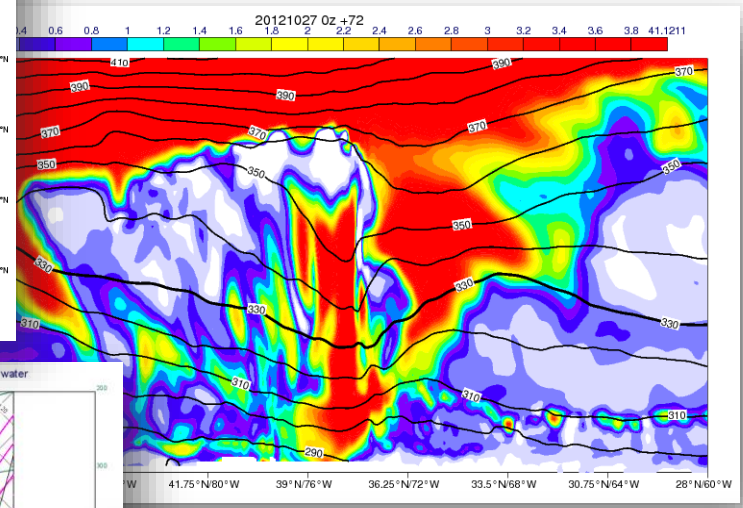


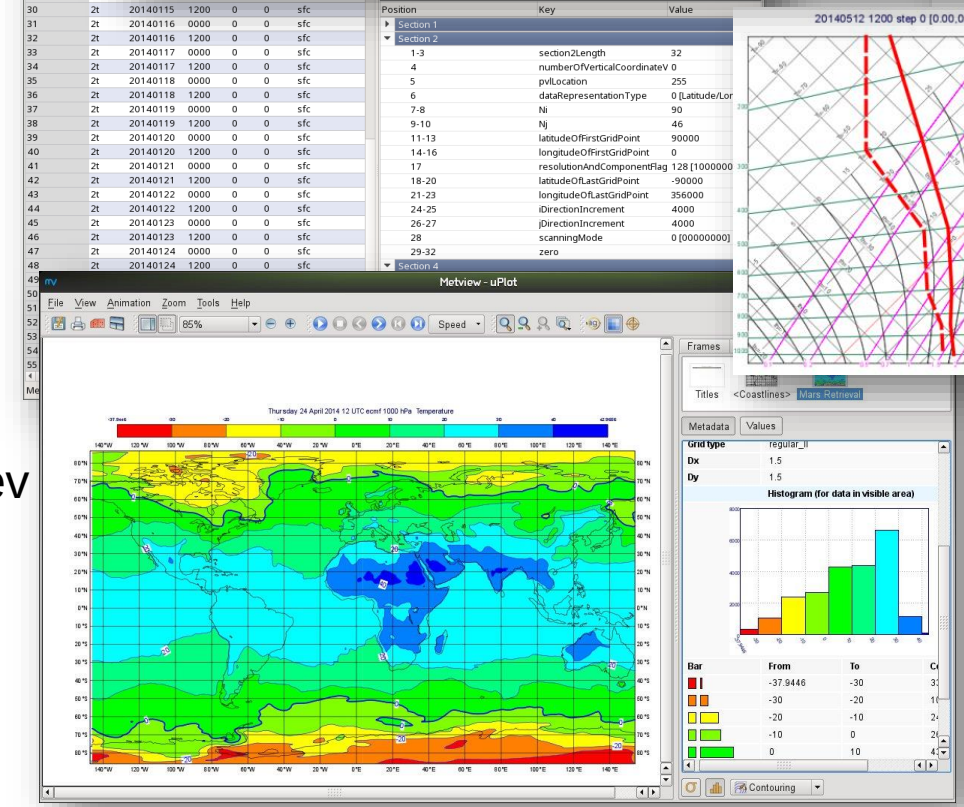
Now please do exercise Part 1 - Intro

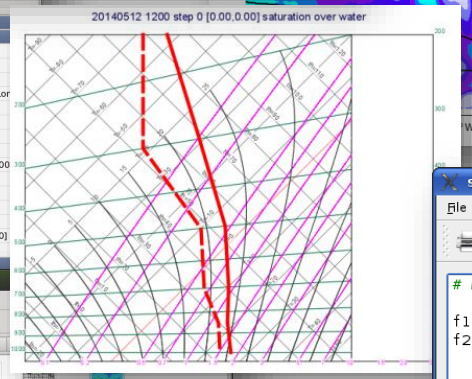
Metview + GRIB

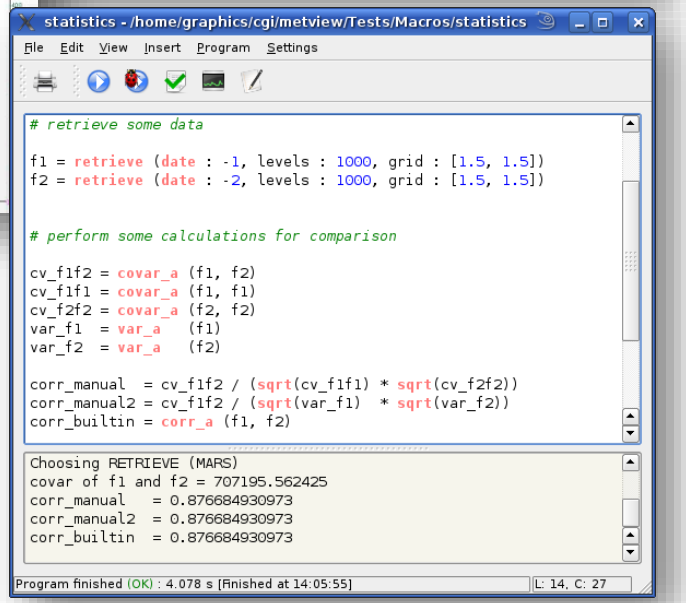
- Plot
- Examine
- Filter, regrid, masking
- Maths, Boolean
- Specialised:
  - Cross section
  - Thermodynamics
  - Gradient
  - Vertical integration
  - Model to pressure lev
  - Etc







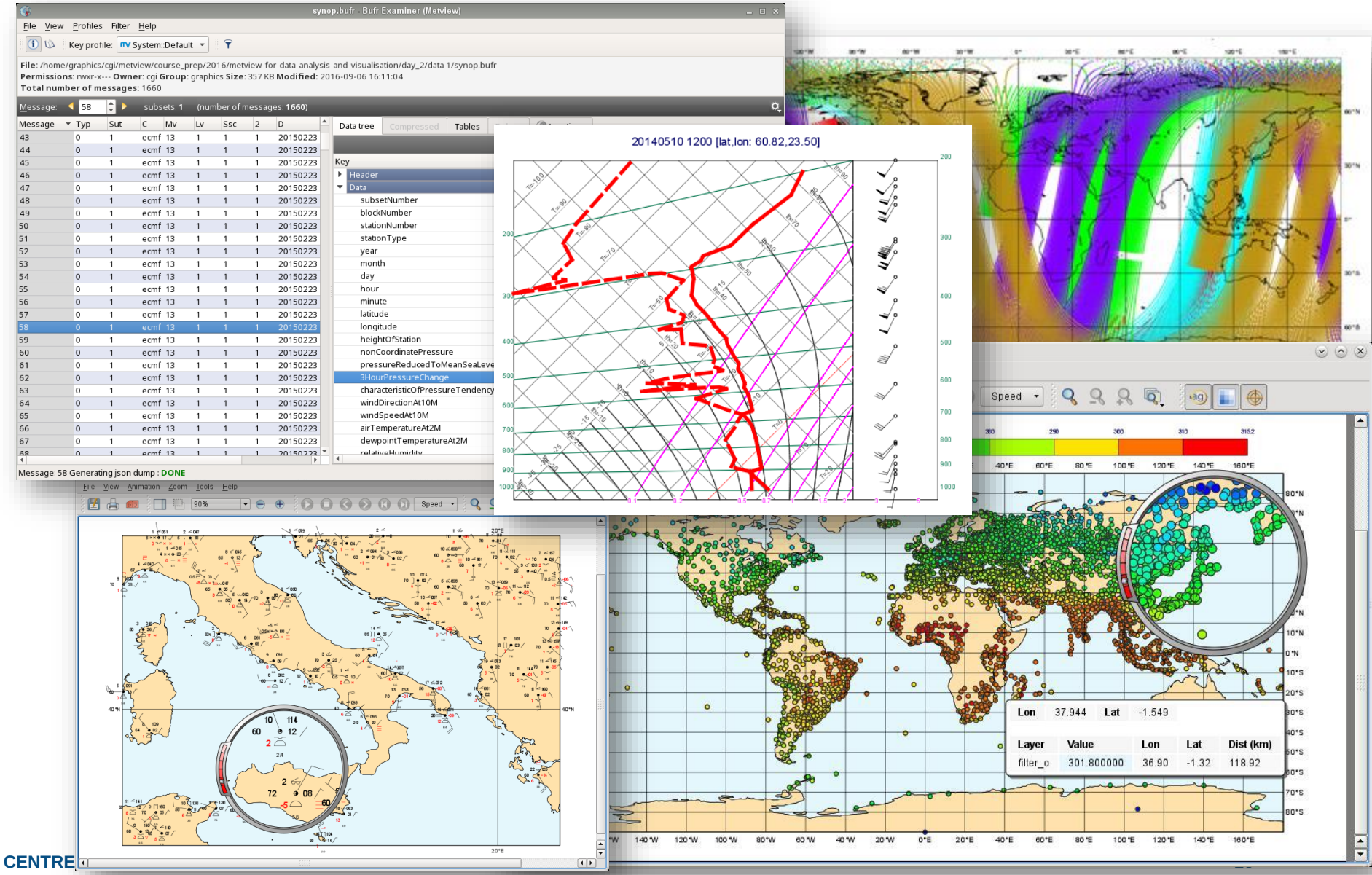




Now please do exercise Part 2 - GRIB

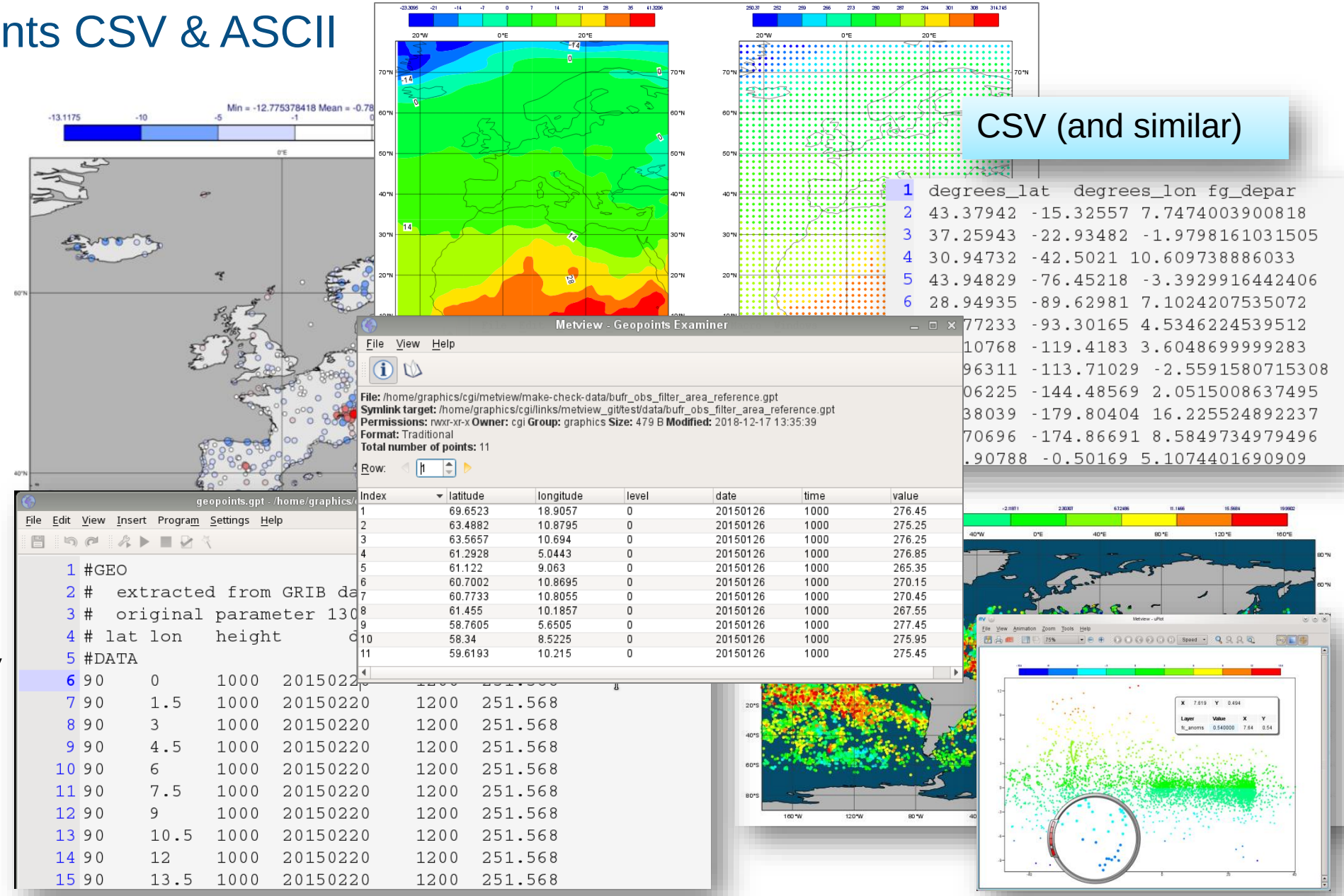
Metview + BUFR

- Plot
- Examine
- Filter
- Extract values
- Convert to Geopoints
- Thermodynamics



Metview + Geopoints CSV & ASCII

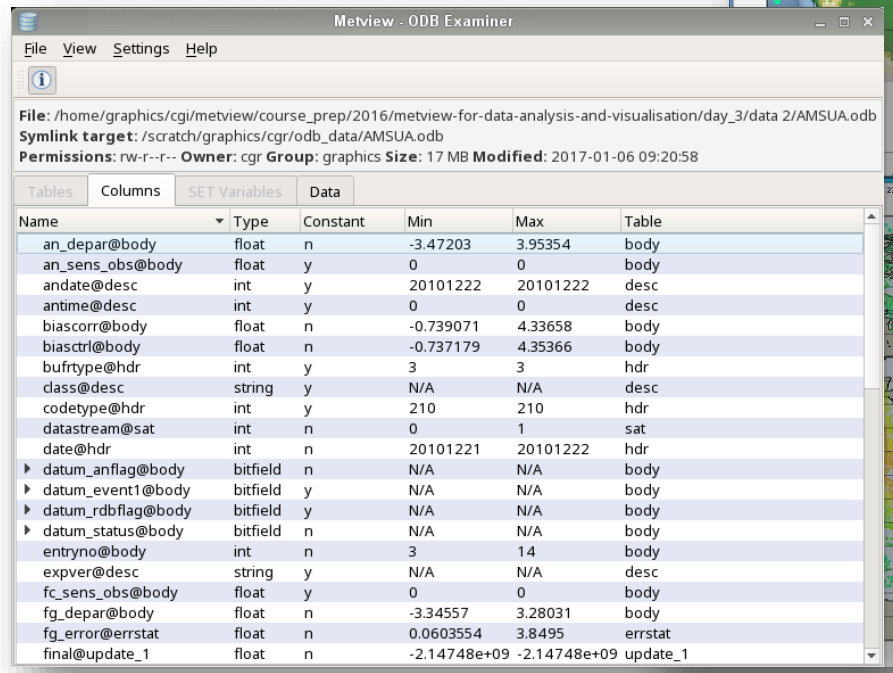
- Geopoints – geo-located values
- Plot
- Examine
- Filter
- Maths, Boolean
- Geo functions
- Convert between GRIB, BUFR and Geopoints
- Can also read CSV

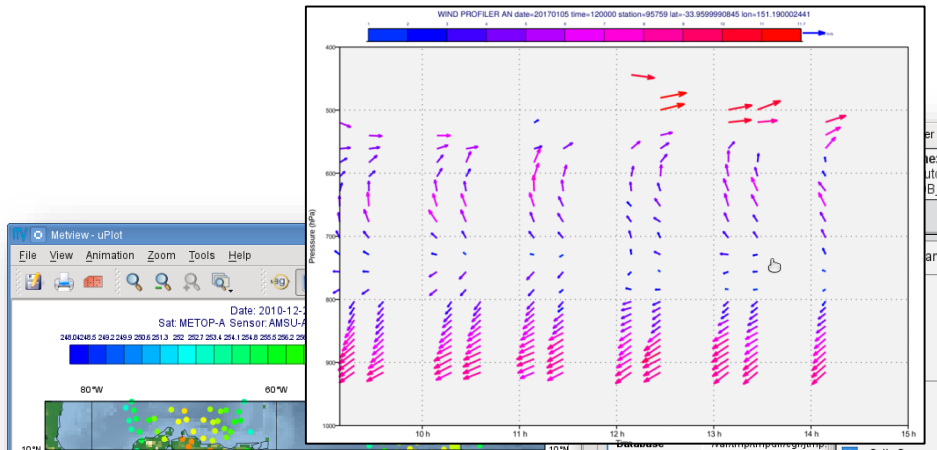


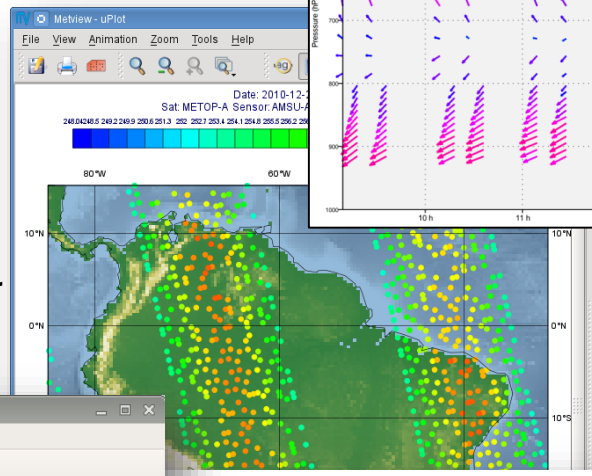
Now please do exercise Part 3 - BUFR

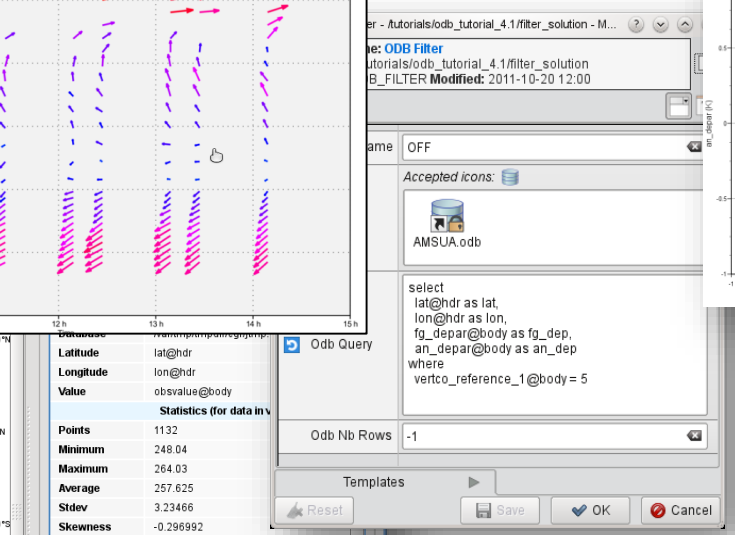
Metview + ODB

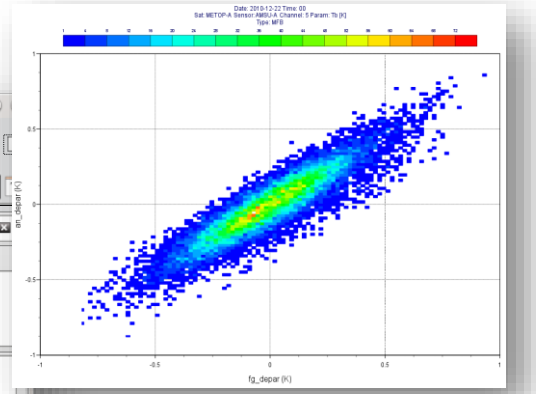
- Plot
- Examine
- Filter
- Convert to Geopoints & pandas

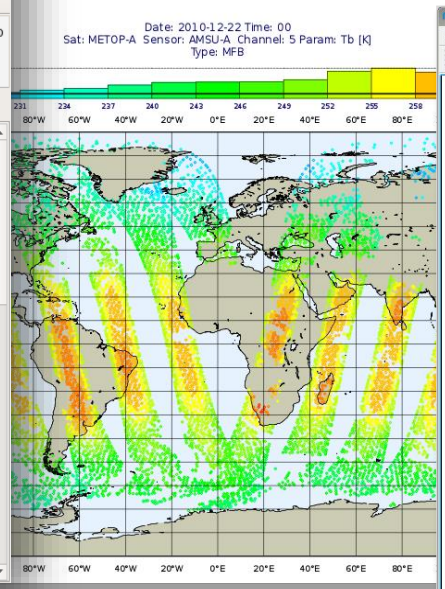


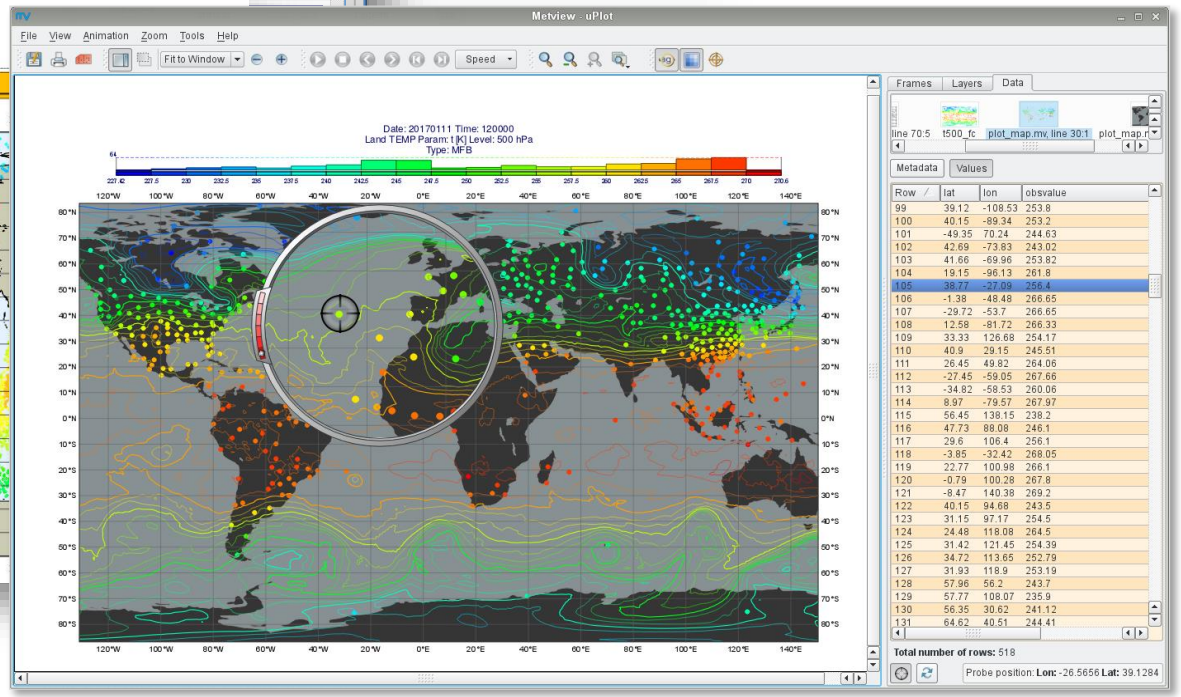








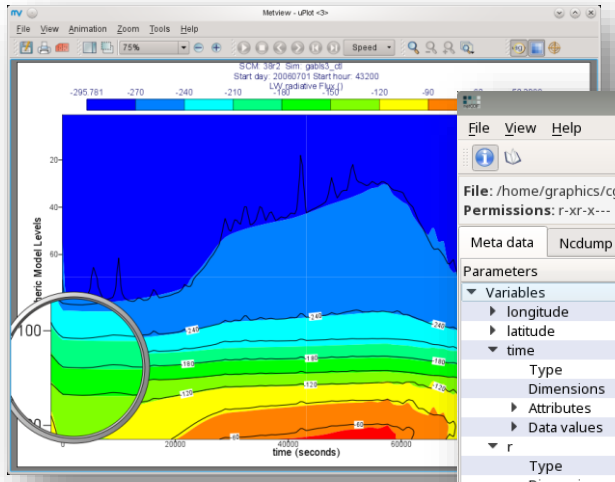


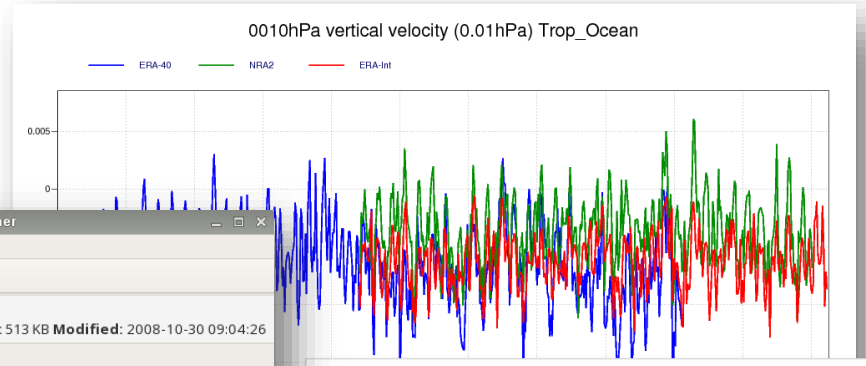


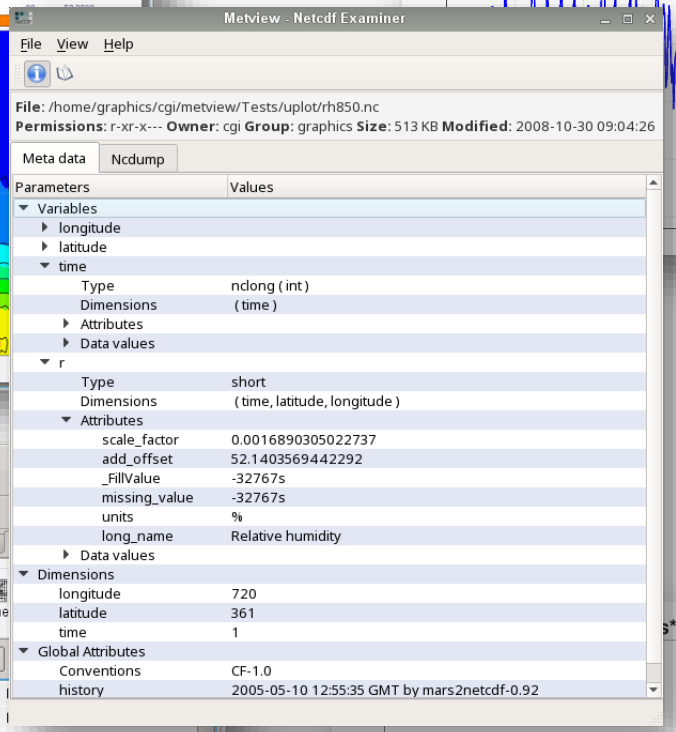
Now please do exercise Part 4 - ODB

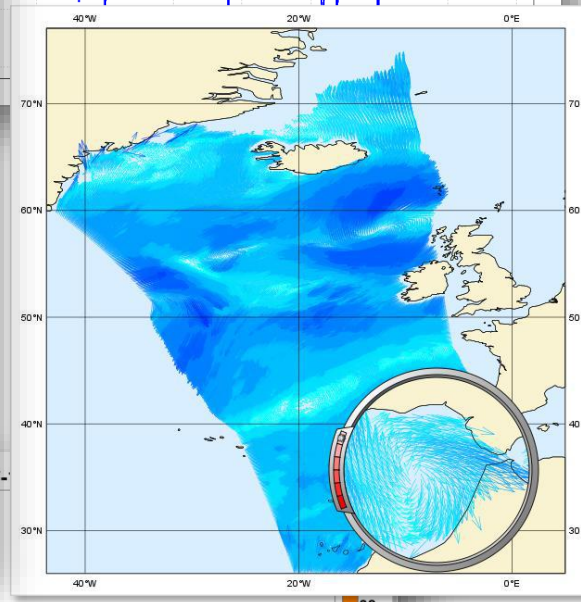
Metview + NetCDF

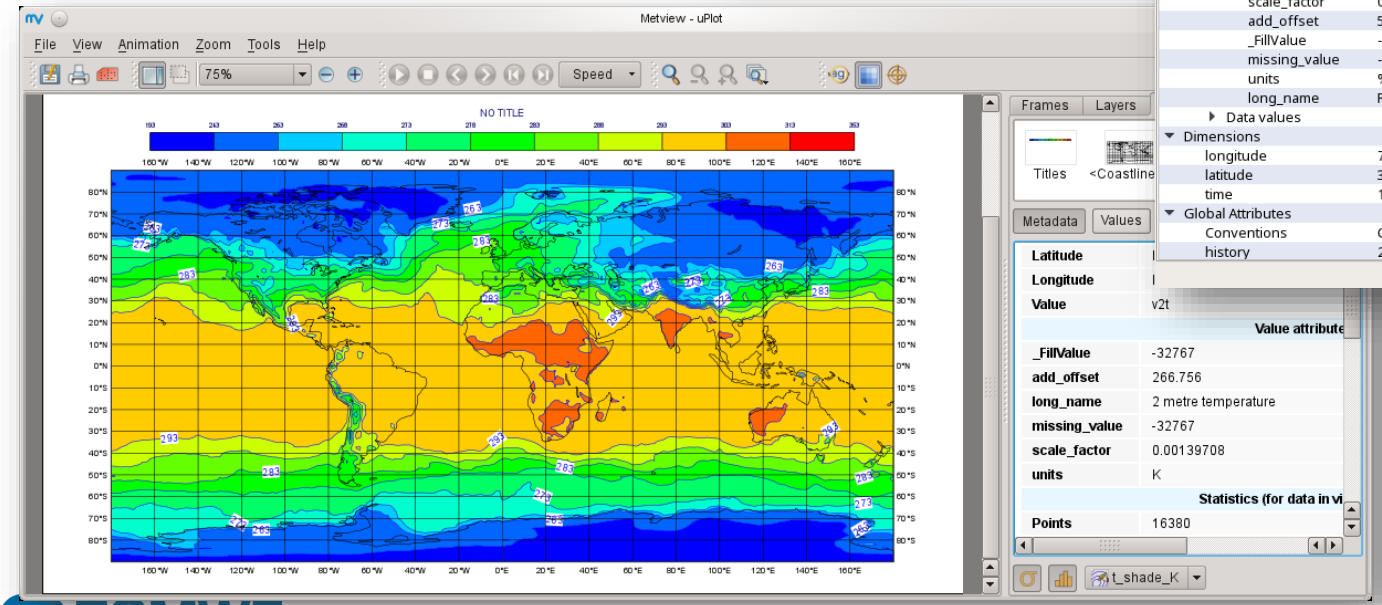
- Plot
- Examine
- Maths, Boolean

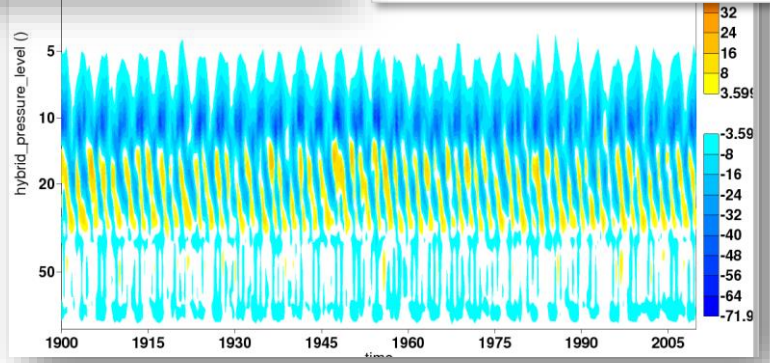








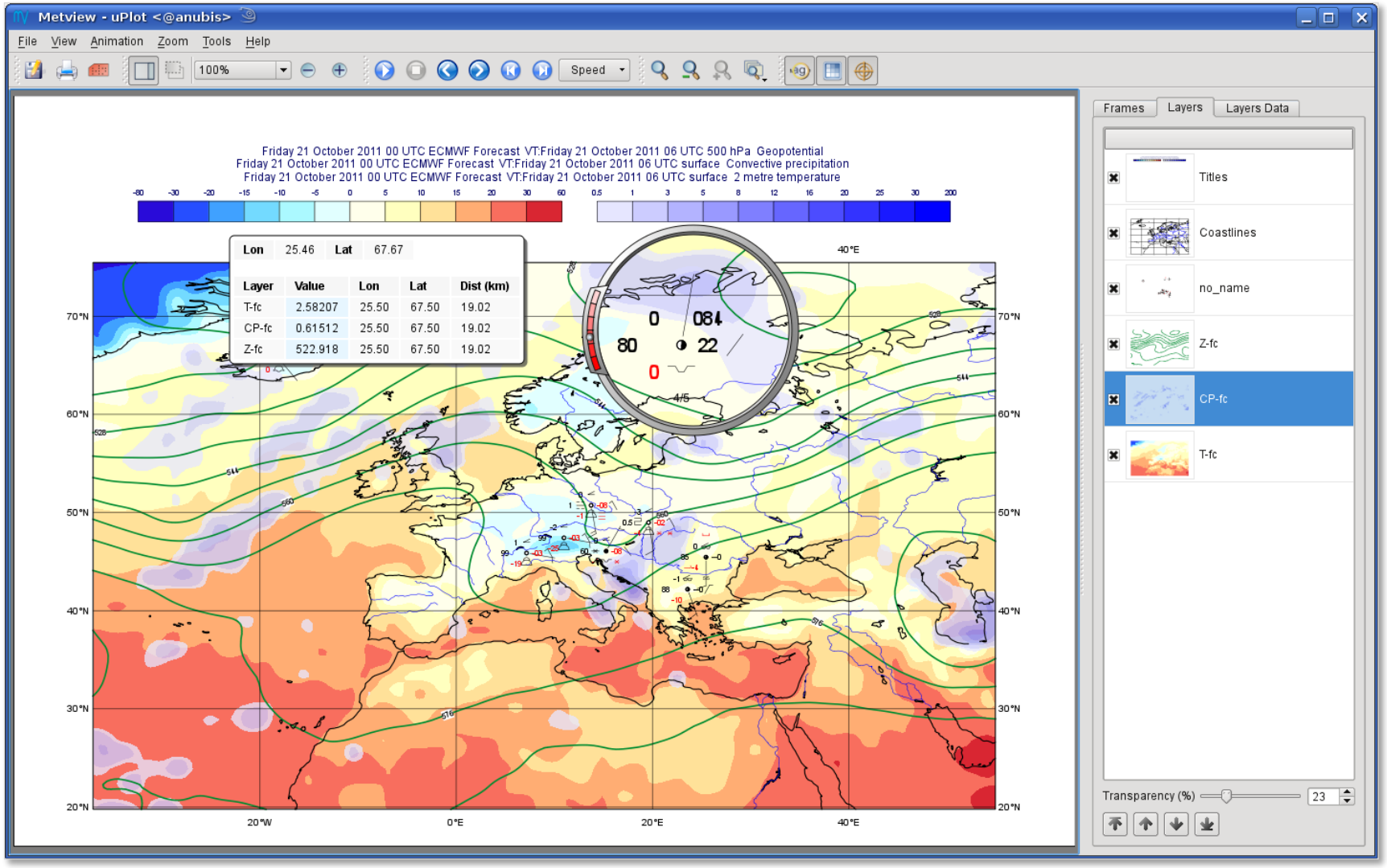




Now please do exercise Part 5 - NetCDF

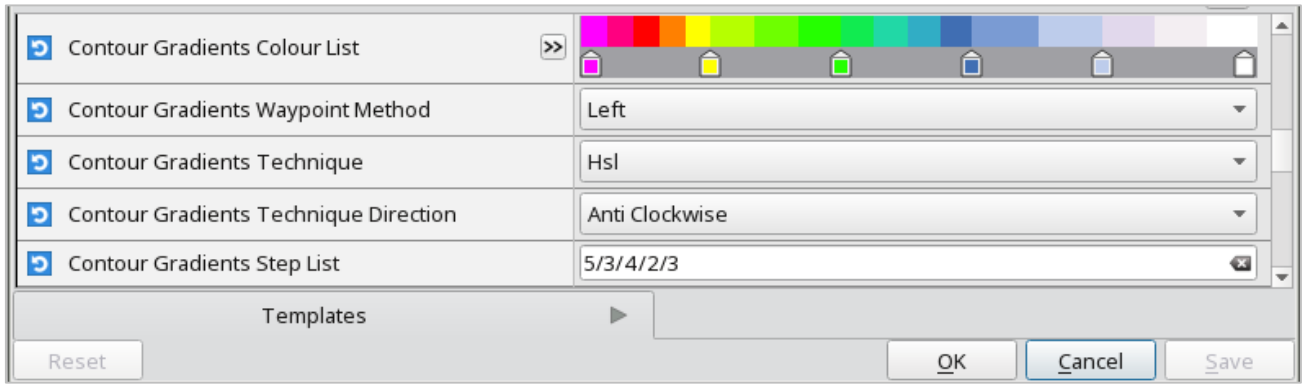
What we didn’t cover

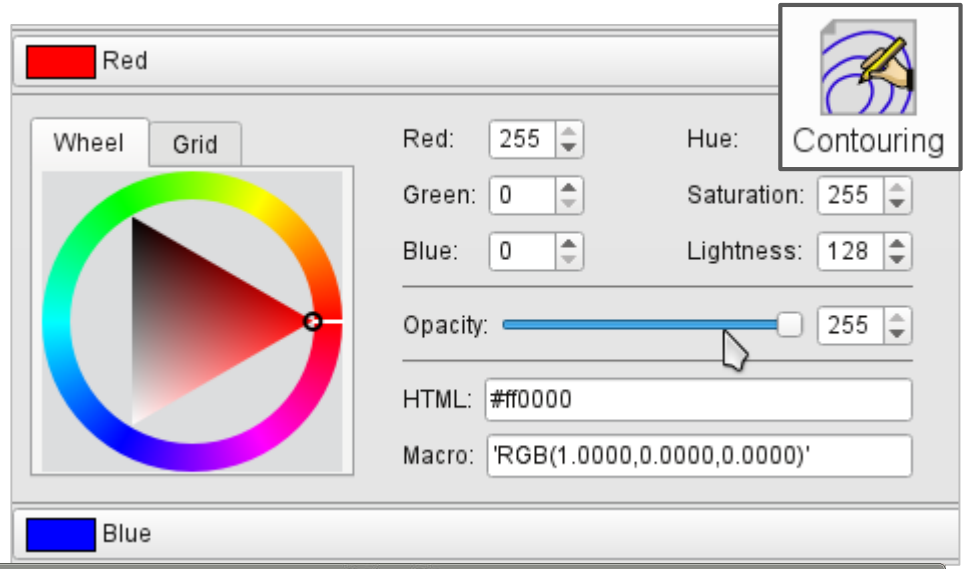
Visualisation - Overlay

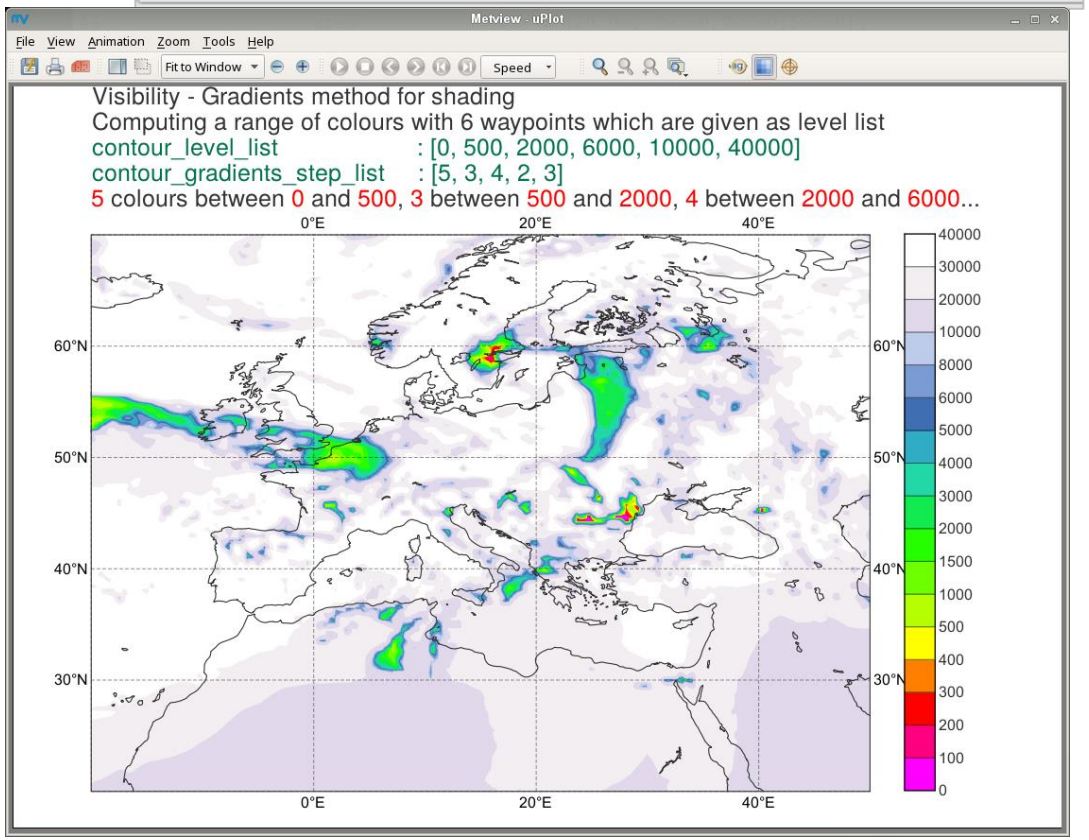


Contouring schemes

- Plenty of options for complete customisation of palettes

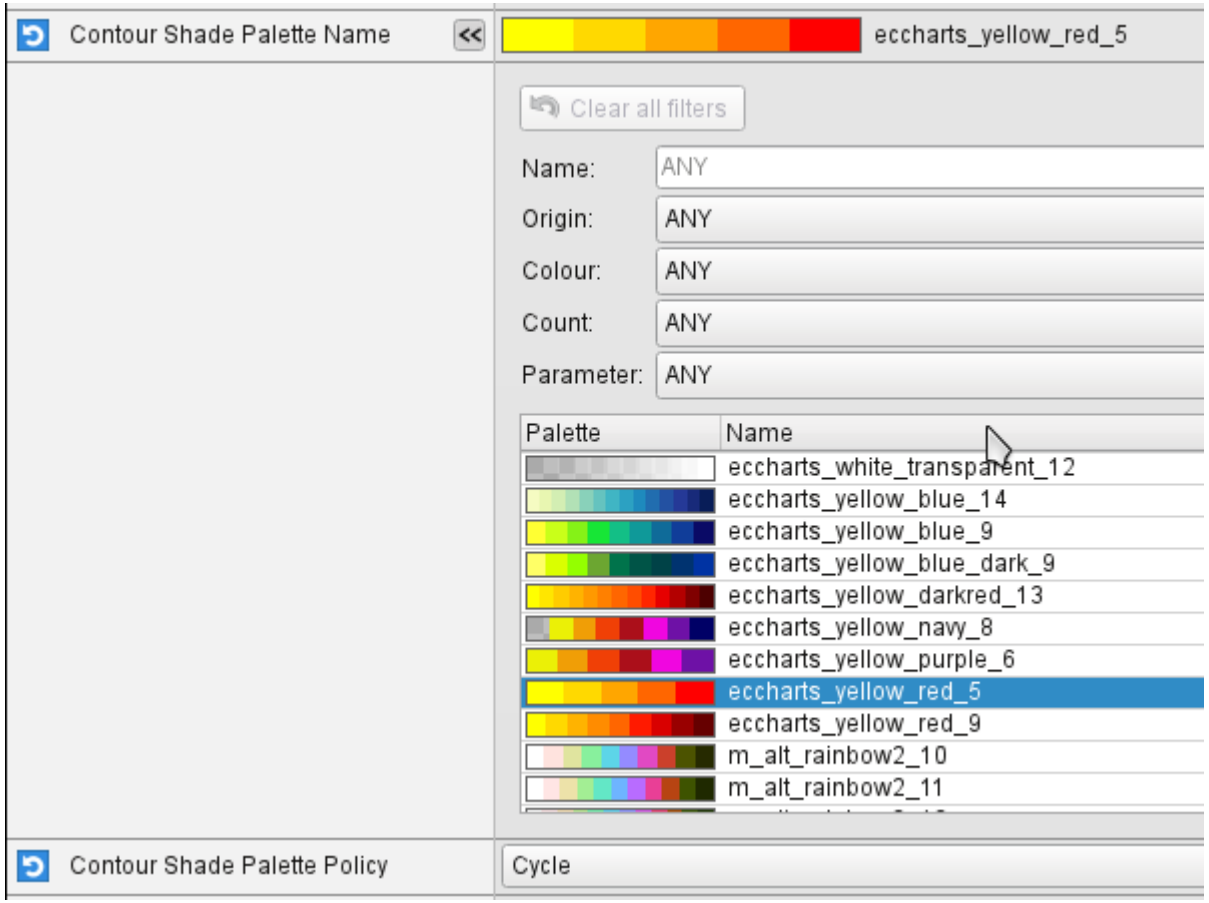


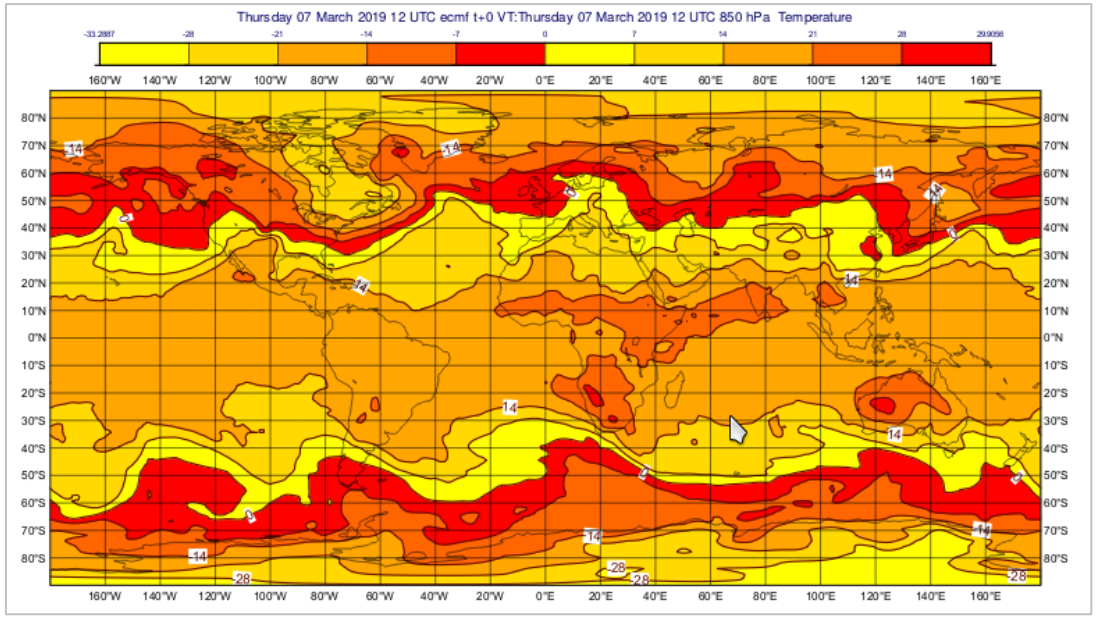




Contouring schemes

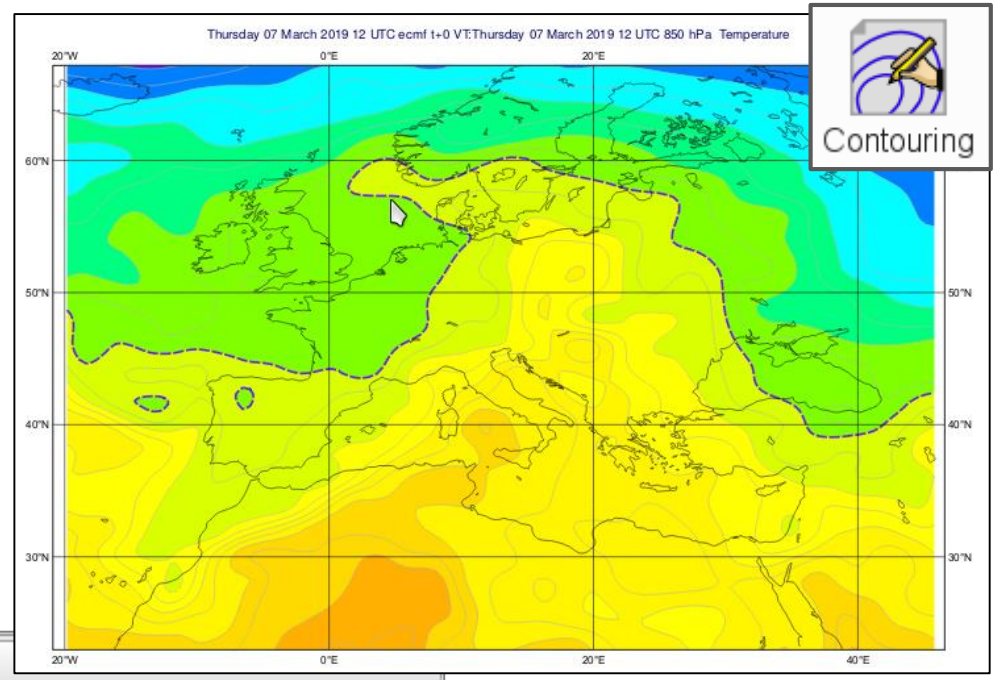
- A set of pre-defined palettes is also available
  - But you still have to supply the mapping between values and colours

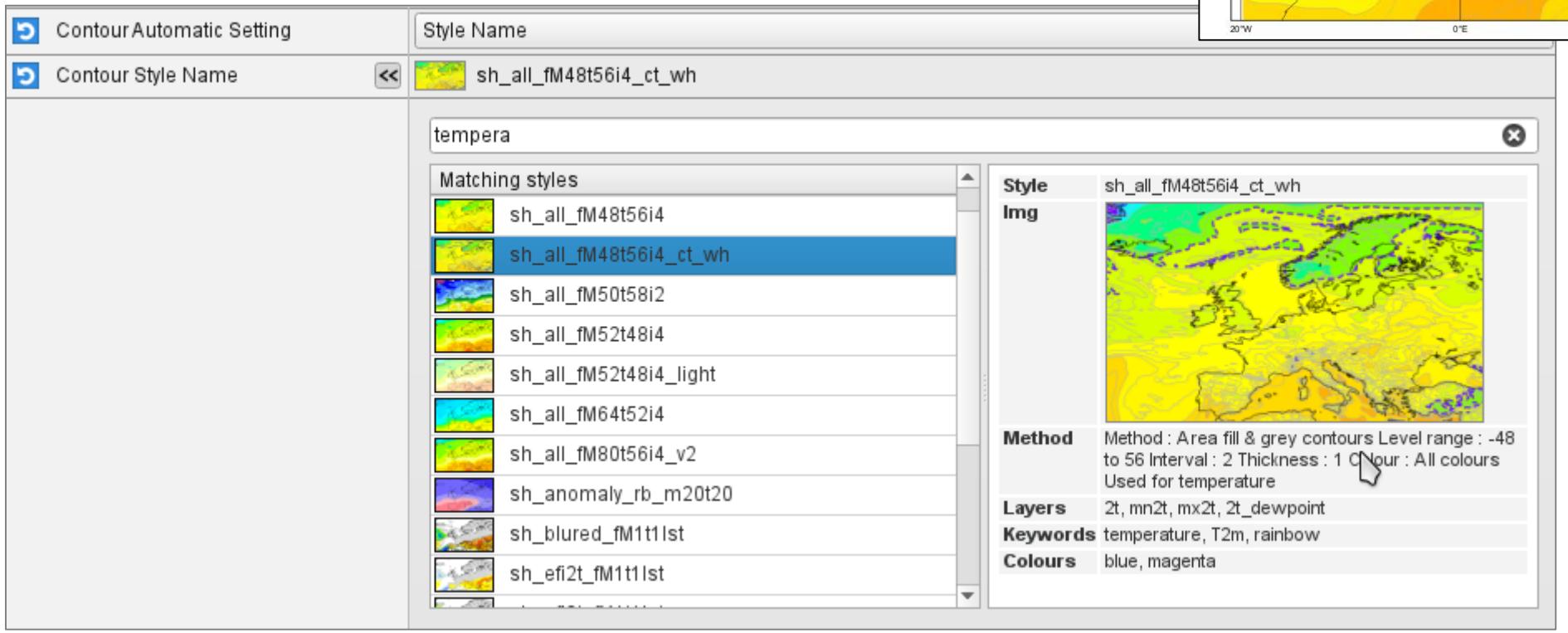




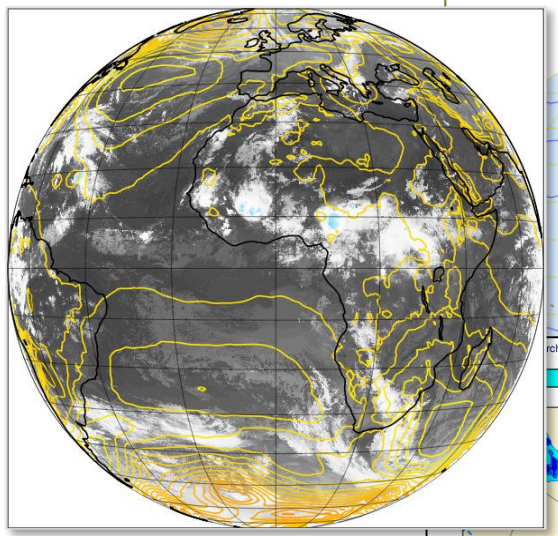
Contouring schemes

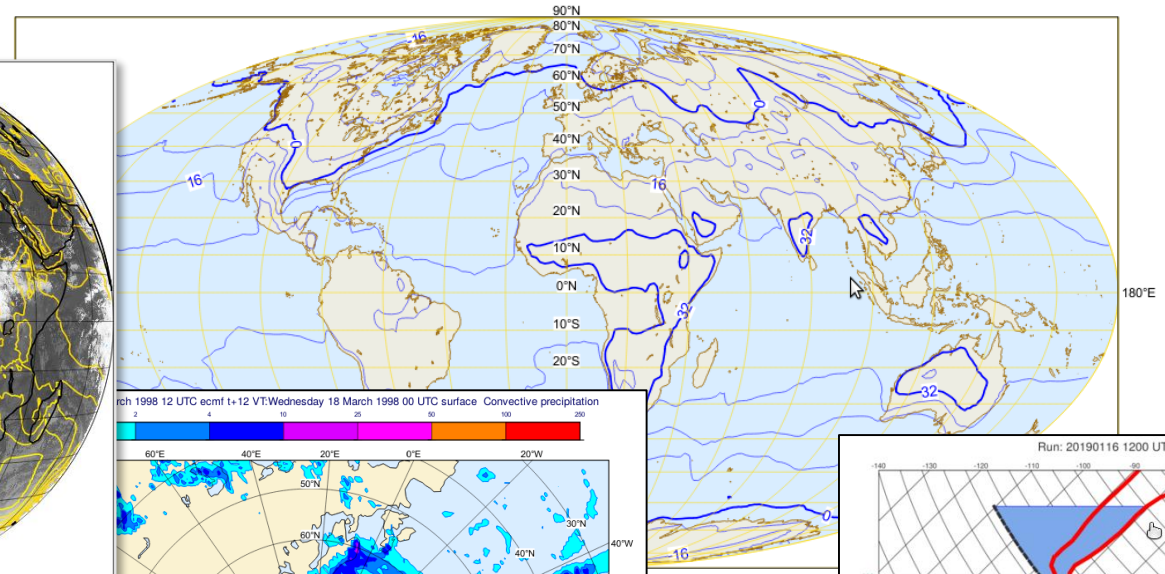
- Can select from pre-defined styles
  - the styles come from ecCharts
  - everything is done for you
  - or choose “Contour Automatic Setting = ECMWF” – style will be chosen based on meta-data

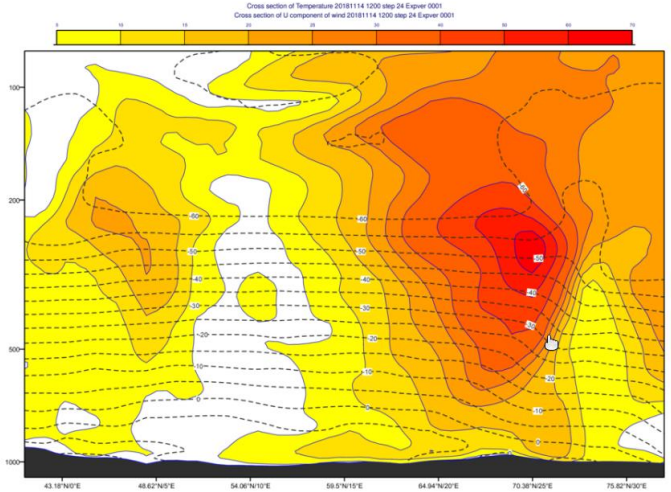


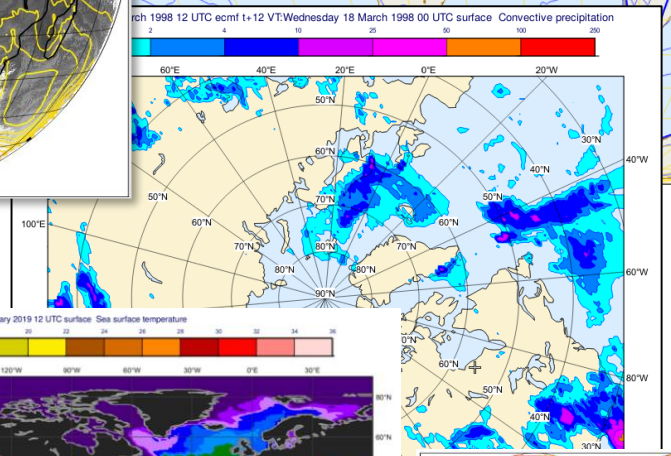


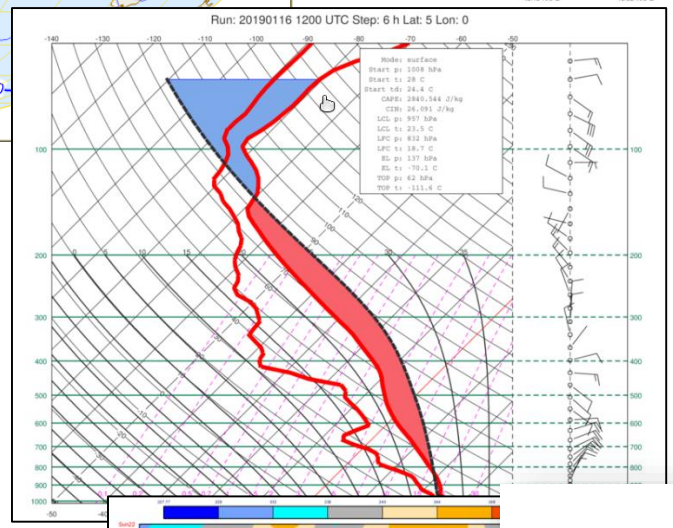
Views

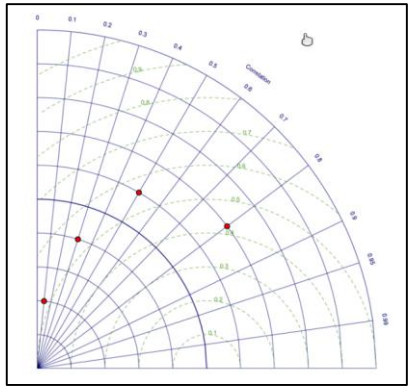


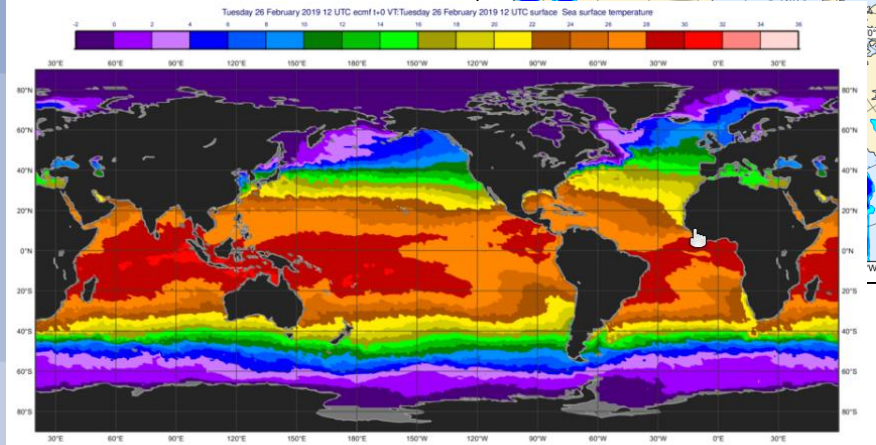


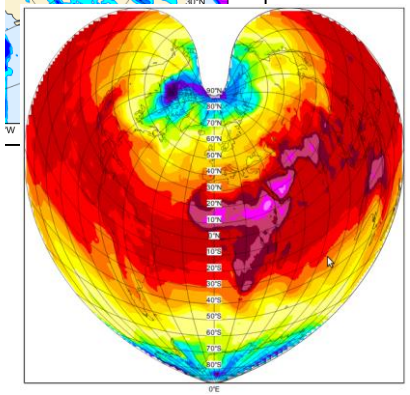


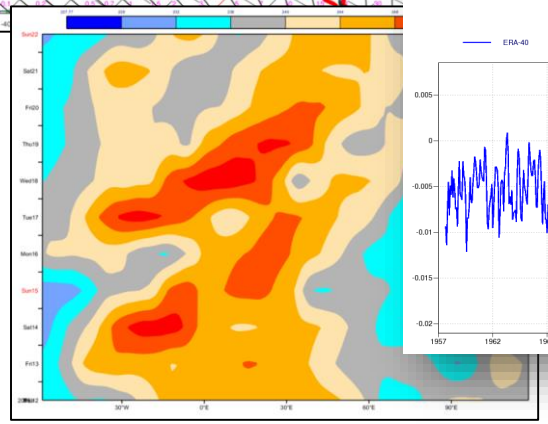


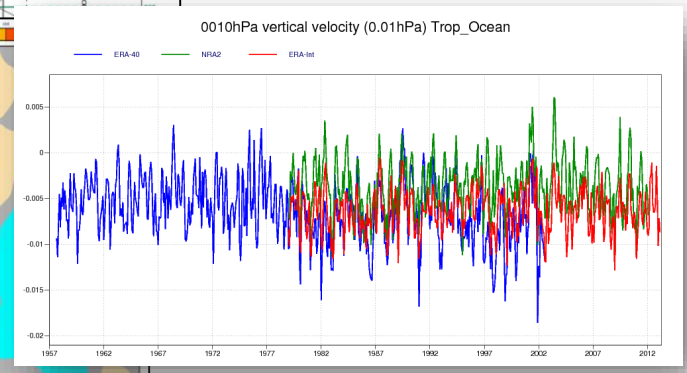










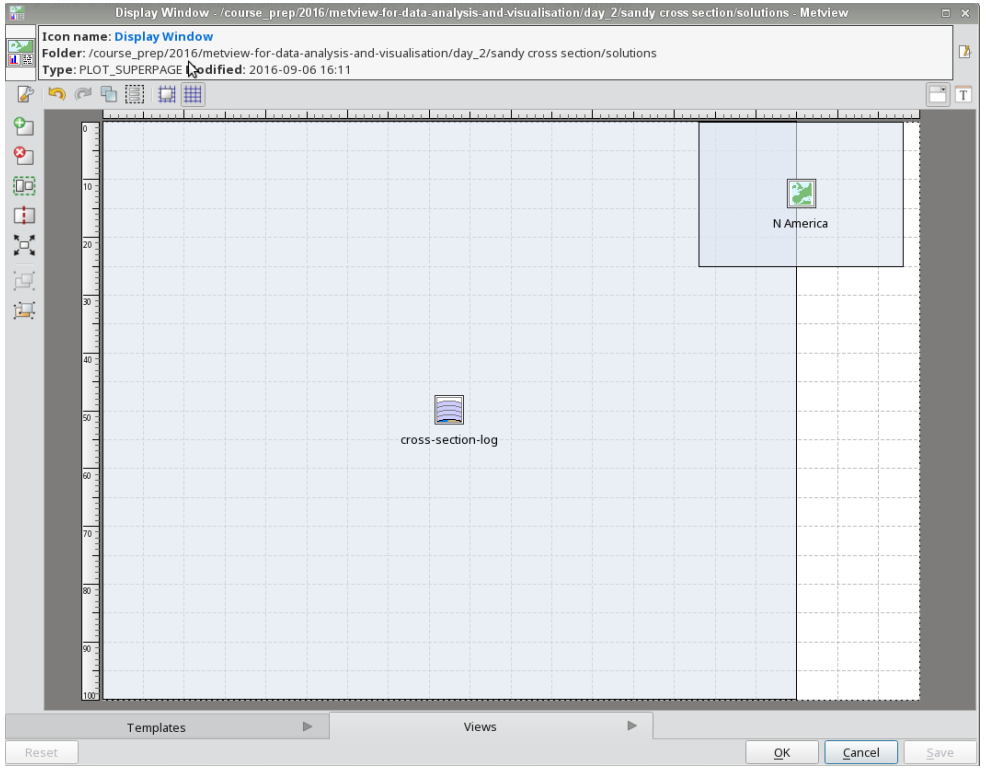


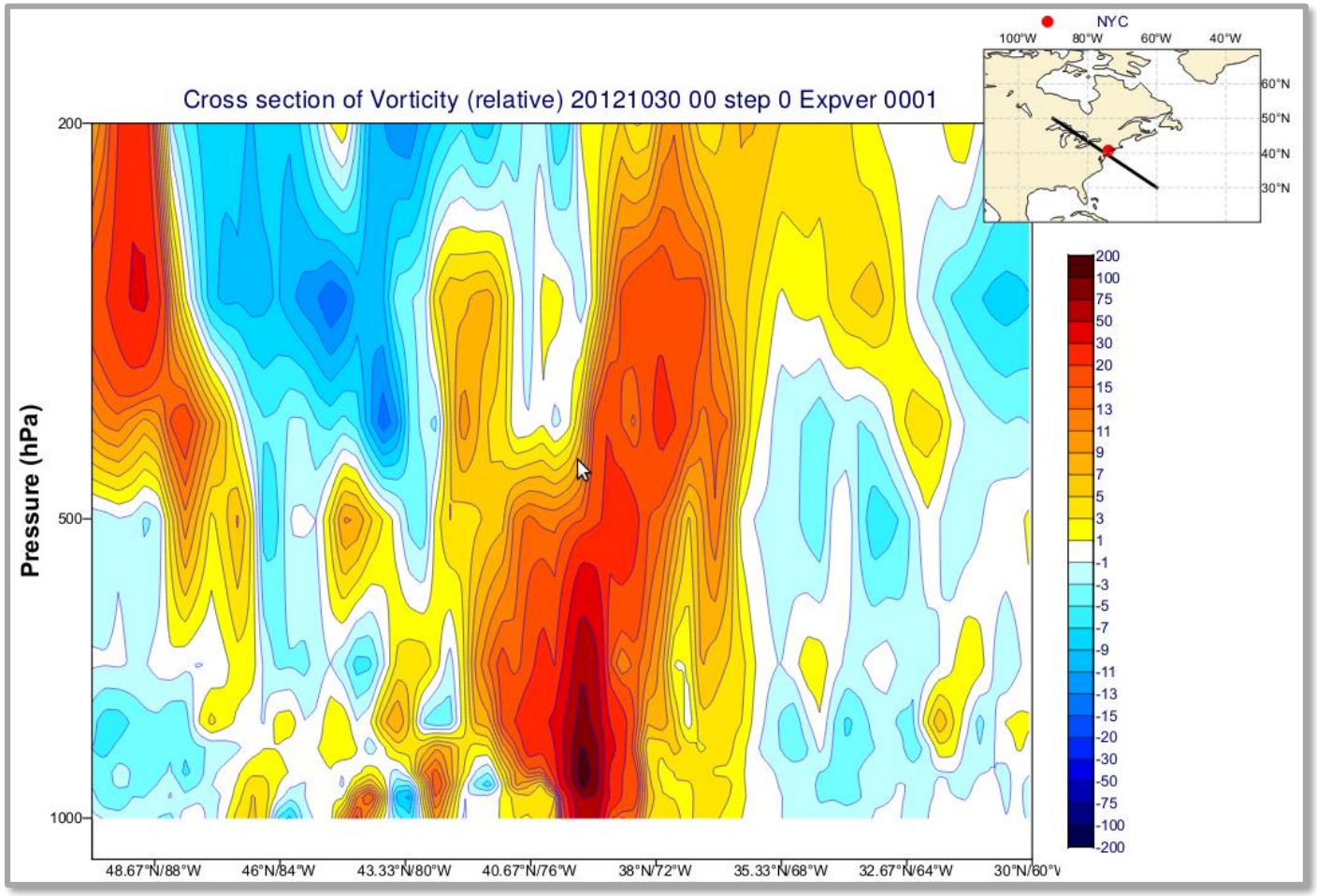
ECMWF

EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS

Visualisation - Layout

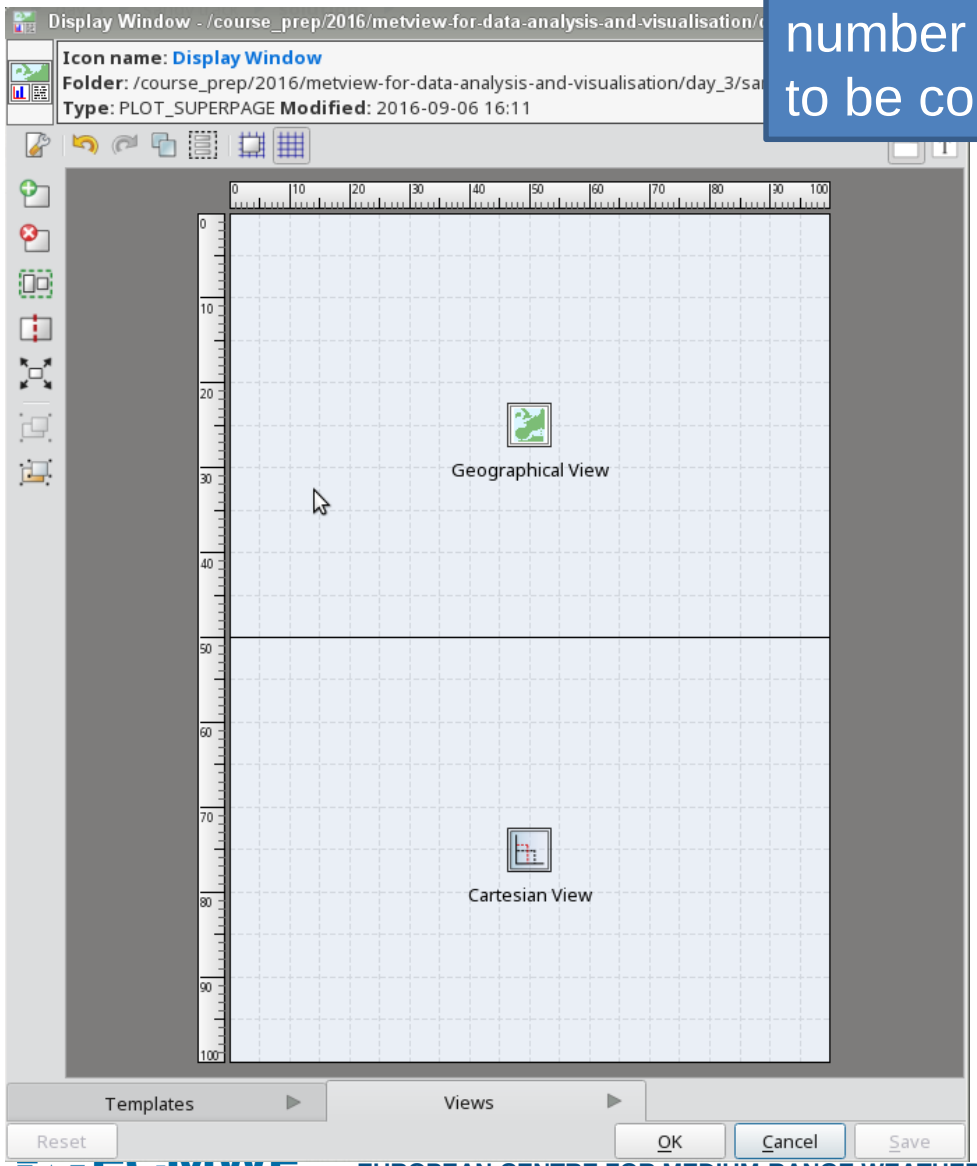
Layout editor allows any number of different views to be combined

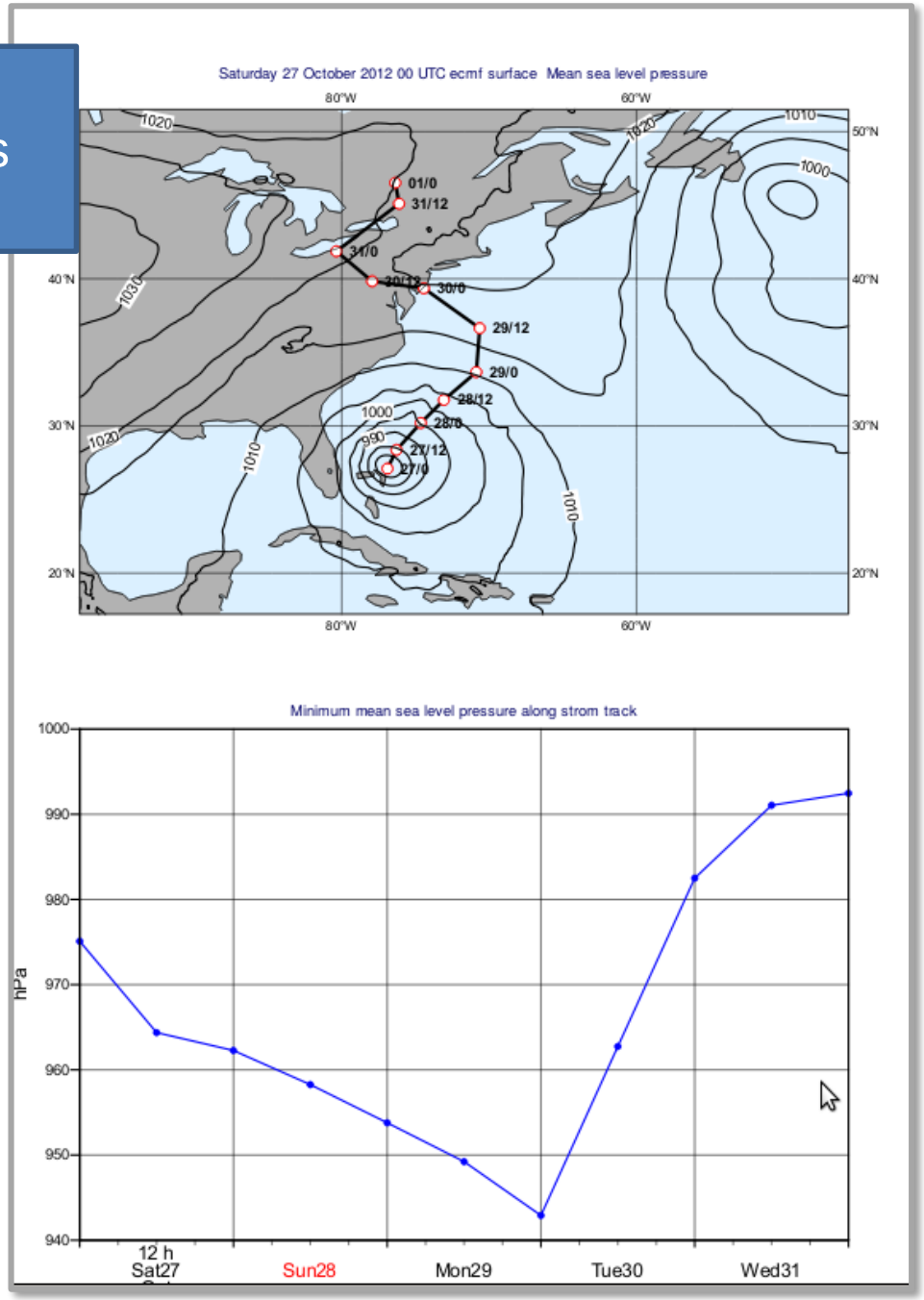




Visualisation - Layout

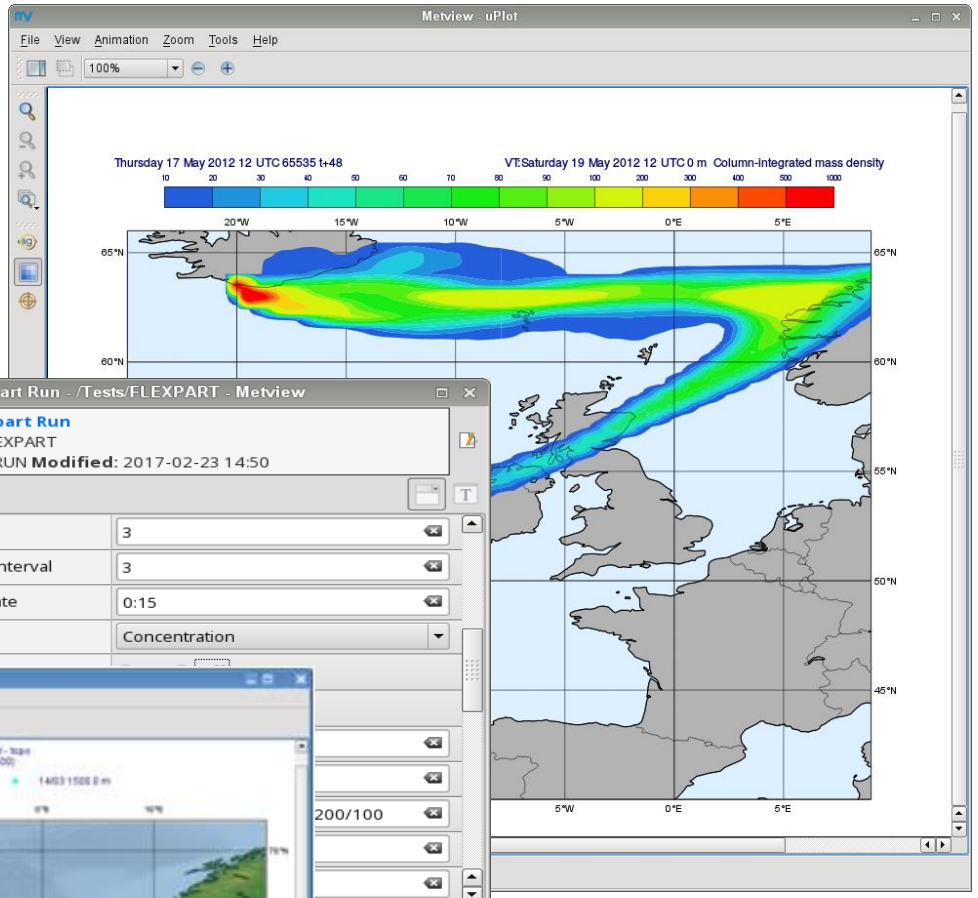
Layout editor allows any number of different views to be combined

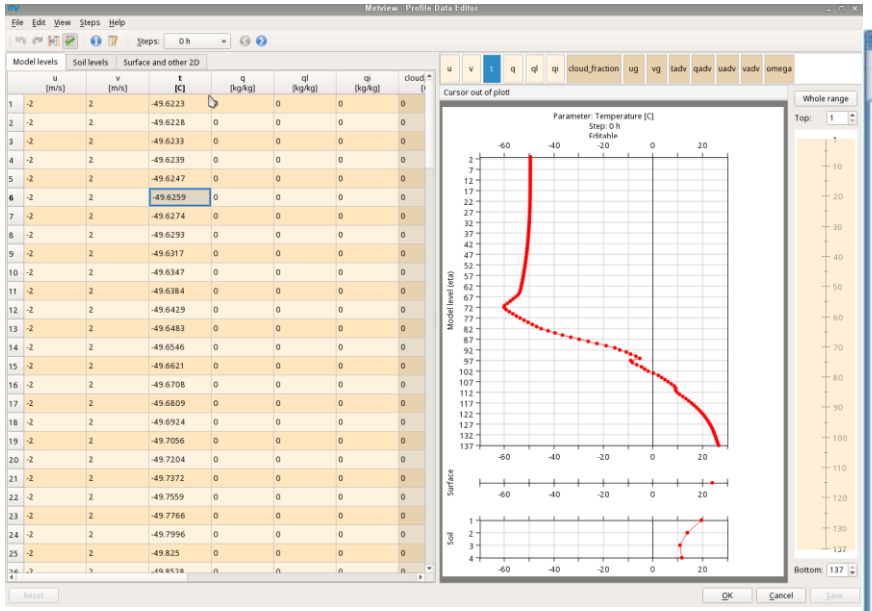


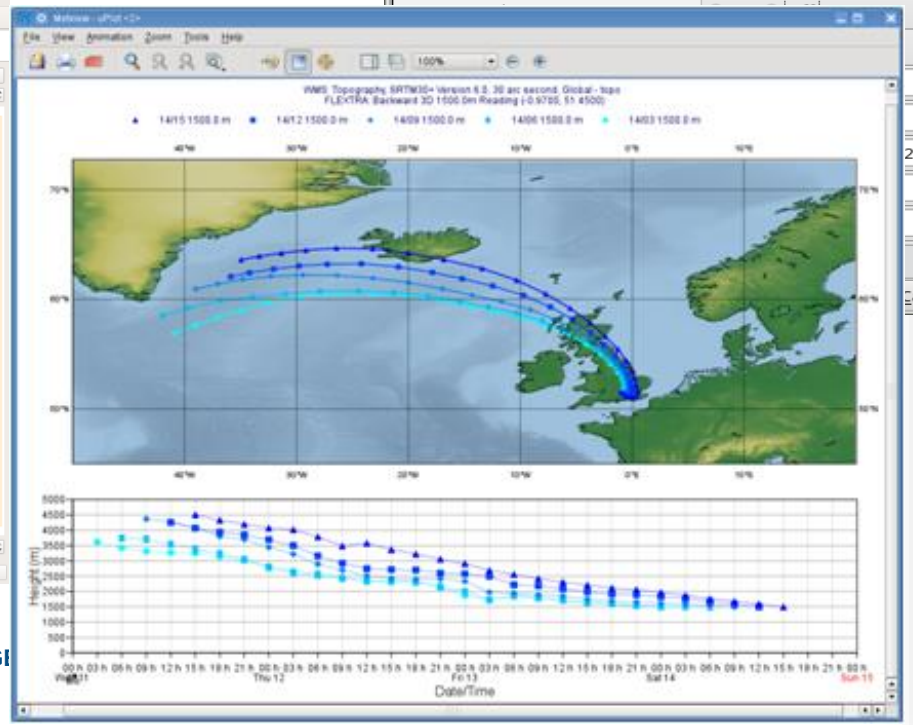


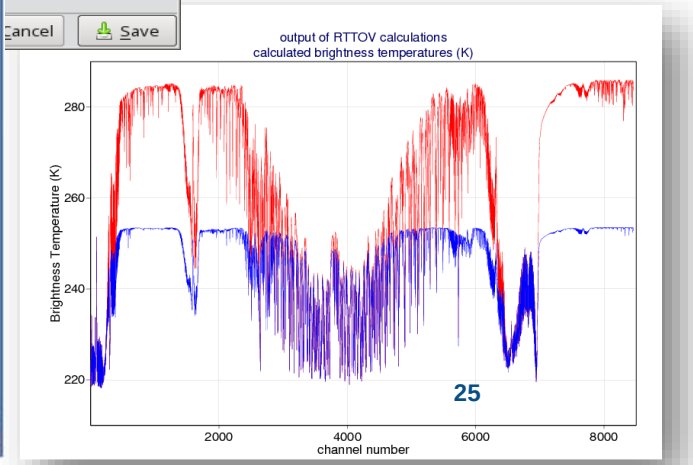
Running models

- Metview is able to prepare data for, run, and plot output from:
- FLEXTRA (trajectory)
- FLEXPART (particle dispersion)
- SCM (Single Column Model)
- RTTOV (Satellite)



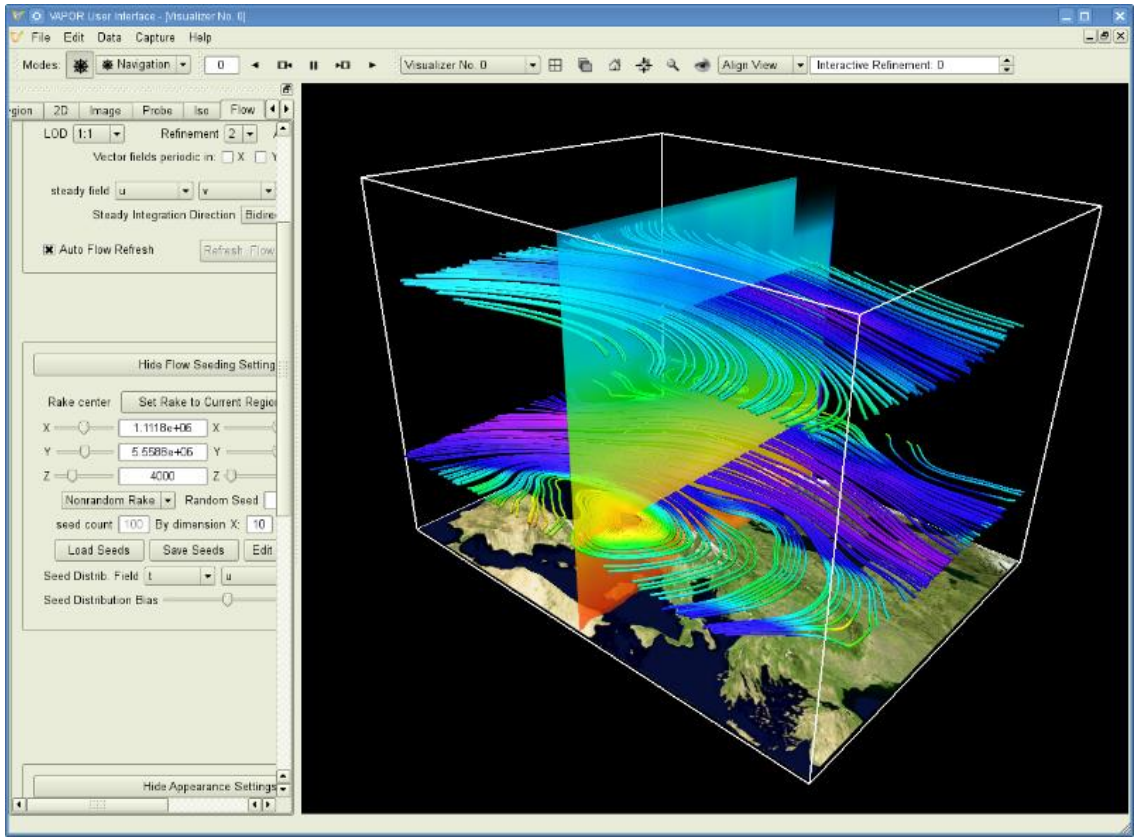




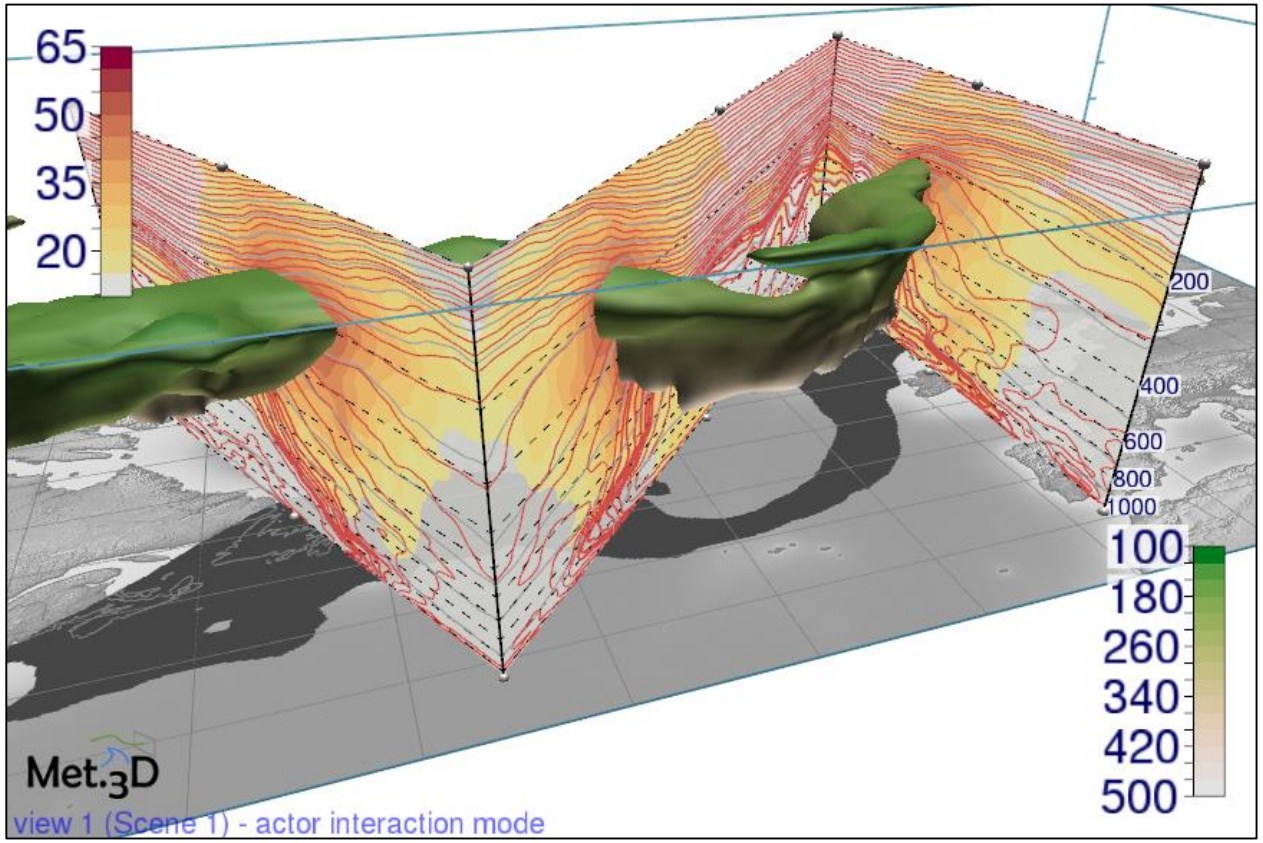


3D

- Metview can prepare data for, and launch:
- VAPOR, Met.3D



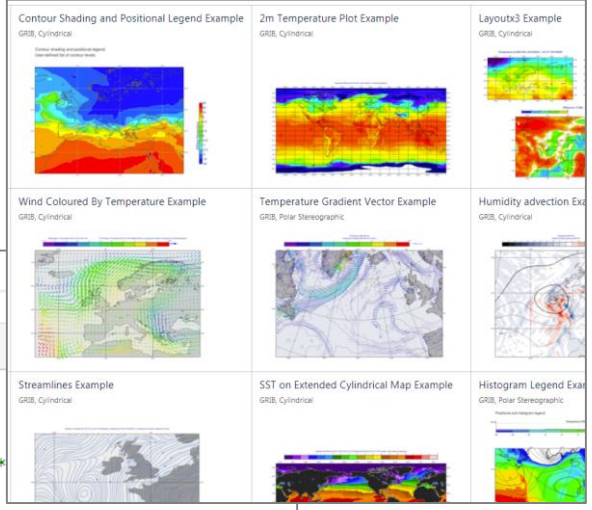
Imagery produced by VAPOR



Imagery produced by Met.3D (met3d.wavestoweather.de)

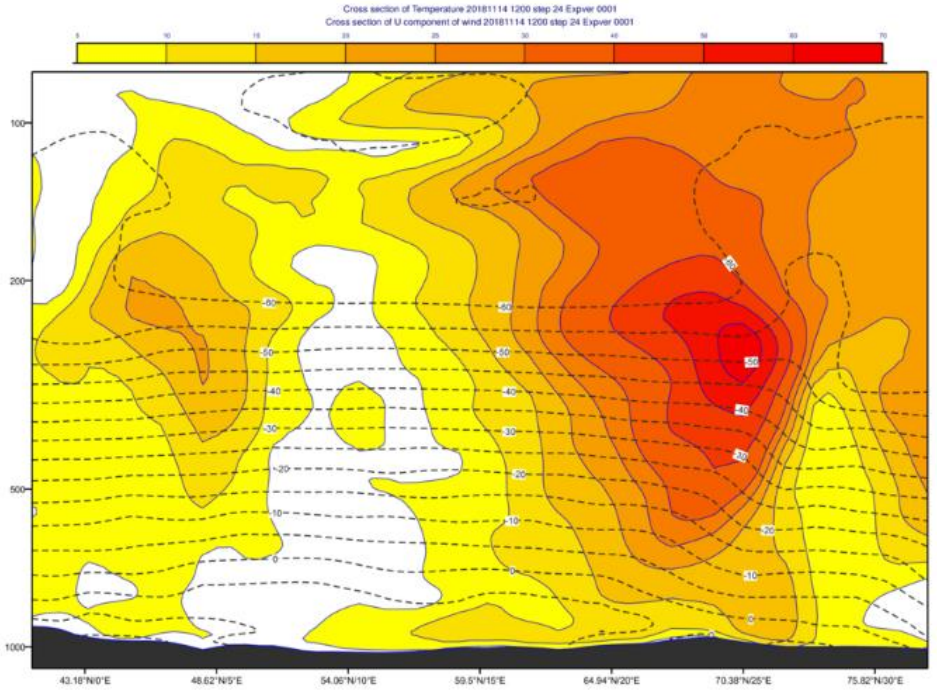
Examples

- See the Gallery for Macro and Python examples



Cross Section with Orography Example

Created by Sandor Kertesz, last modified on Feb 21, 2019



Download source and data
cross_section_orog.tar.gz

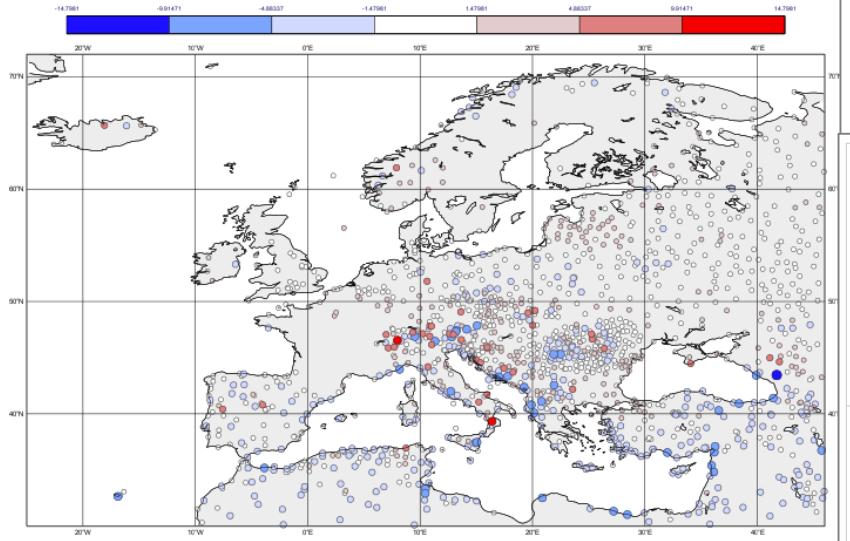
Macro | Python

Cross Section with Orography Example

```
#Metview Macro

#  ***************************** LICENSE START ***********************
#
#  Copyright 2019 ECMWF. This software is distributed under the terms
#  of the Apache License version 2.0. In applying this license, ECMWF does not
#  waive the privileges and immunities granted to it by virtue of its status as
#  an Intergovernmental Organization or submit itself to any jurisdiction.
#
#  ***************************** LICENSE END *************************************
#


# read grib file - contains model level data
fs = read(source : "fc_ml.grib")

# read temperature and scale it to C
t = read(data : fs, param : "t")
t = t - 273.16

# read wind components and compute speed
u = read(data : fs, param : "u")
v = read(data : fs, param : "v")
sp = sqrt(u*u + v*v)

# read log of surface pressure
lnsp = read(data : fs, param : "lnsp")

# define cross section line
line = [41,-2,78,32]

# define shading for wind speed
sp_cont = mcont(legend : "on",
                contour_automatics_settings : "style_name",
                contour_style_name : "sh_red_f5t70lst")
```

Examples

- See the Jupyter Notebooks for more Python



Out [14]:

We can easily convert this to a pandas dataframe for further analysis.

In [15]:

```
df = diff.to_dataframe()
```

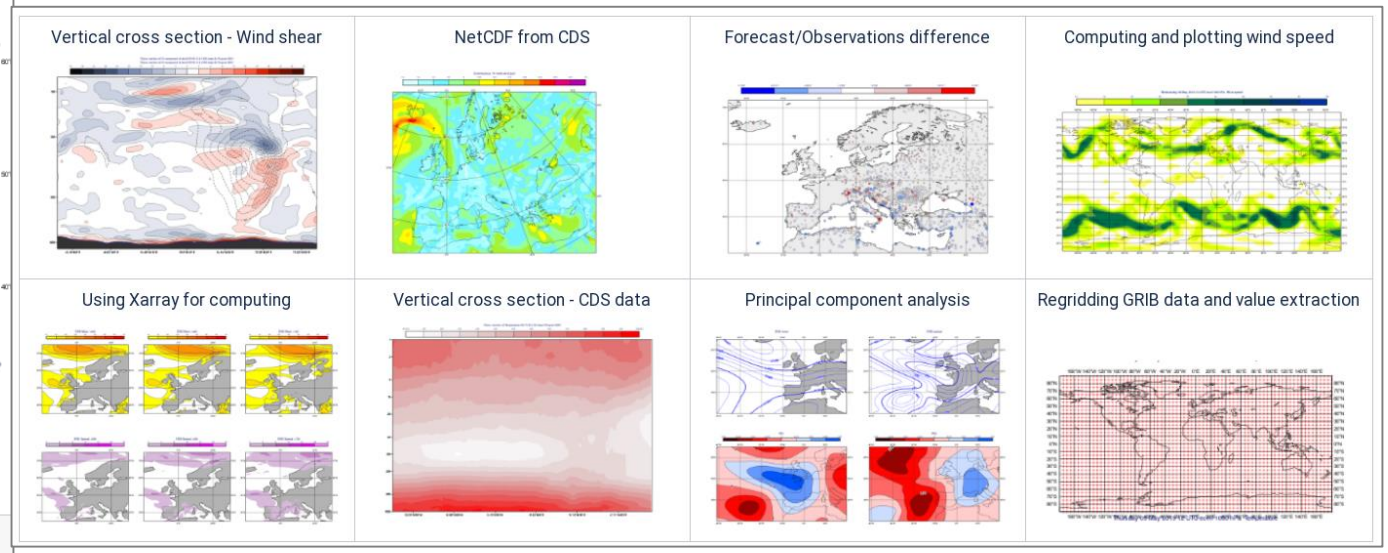
Print a summary of the whole data set:

In [16]:

```
df.describe()
```

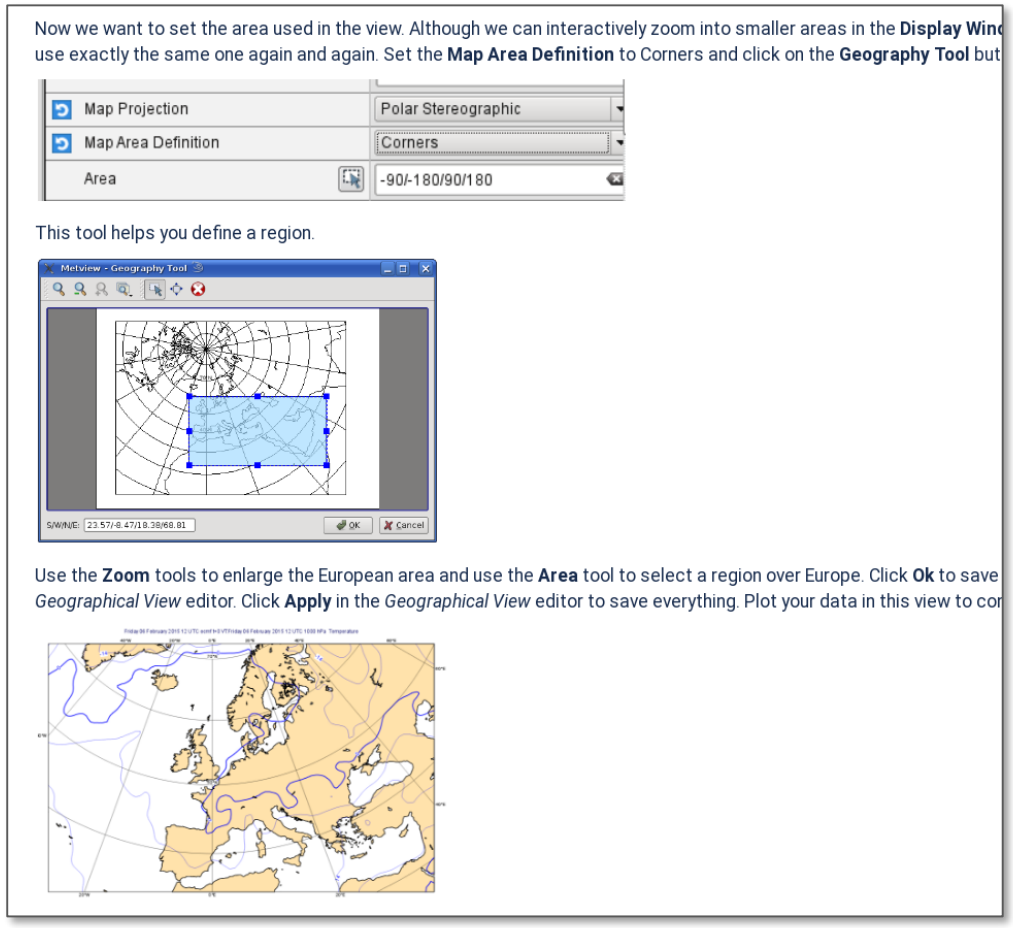
Out [16]:

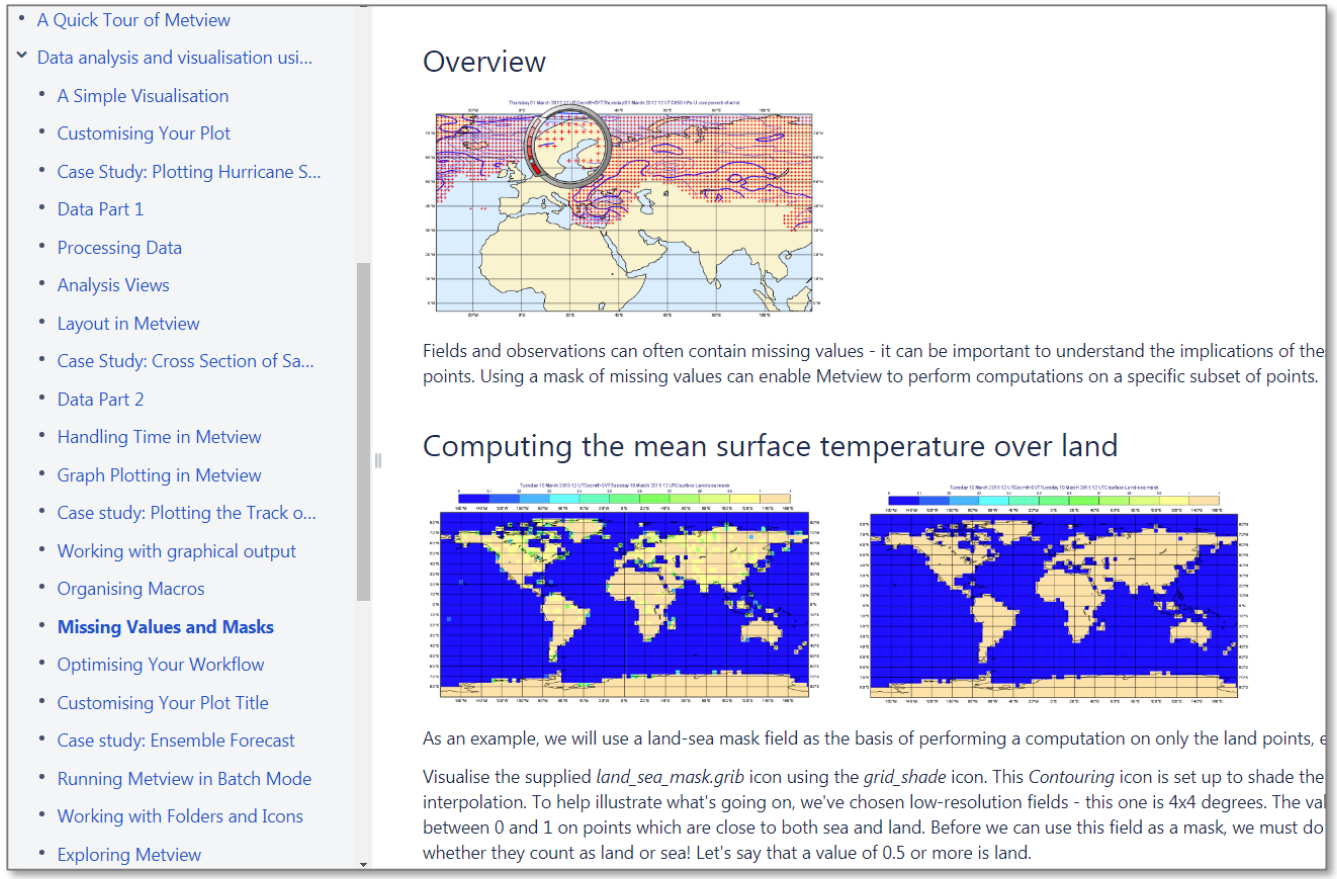
	latitude	longitude	value	level
count	1471.000000	1471.000000	1471.000000	1471.0
mean	46.557104	21.160707	-0.201723	0.0
std	8.350950	14.272239	2.417394	0.0
min	30.110000	-22.590000	-10.236664	0.0



Tutorials

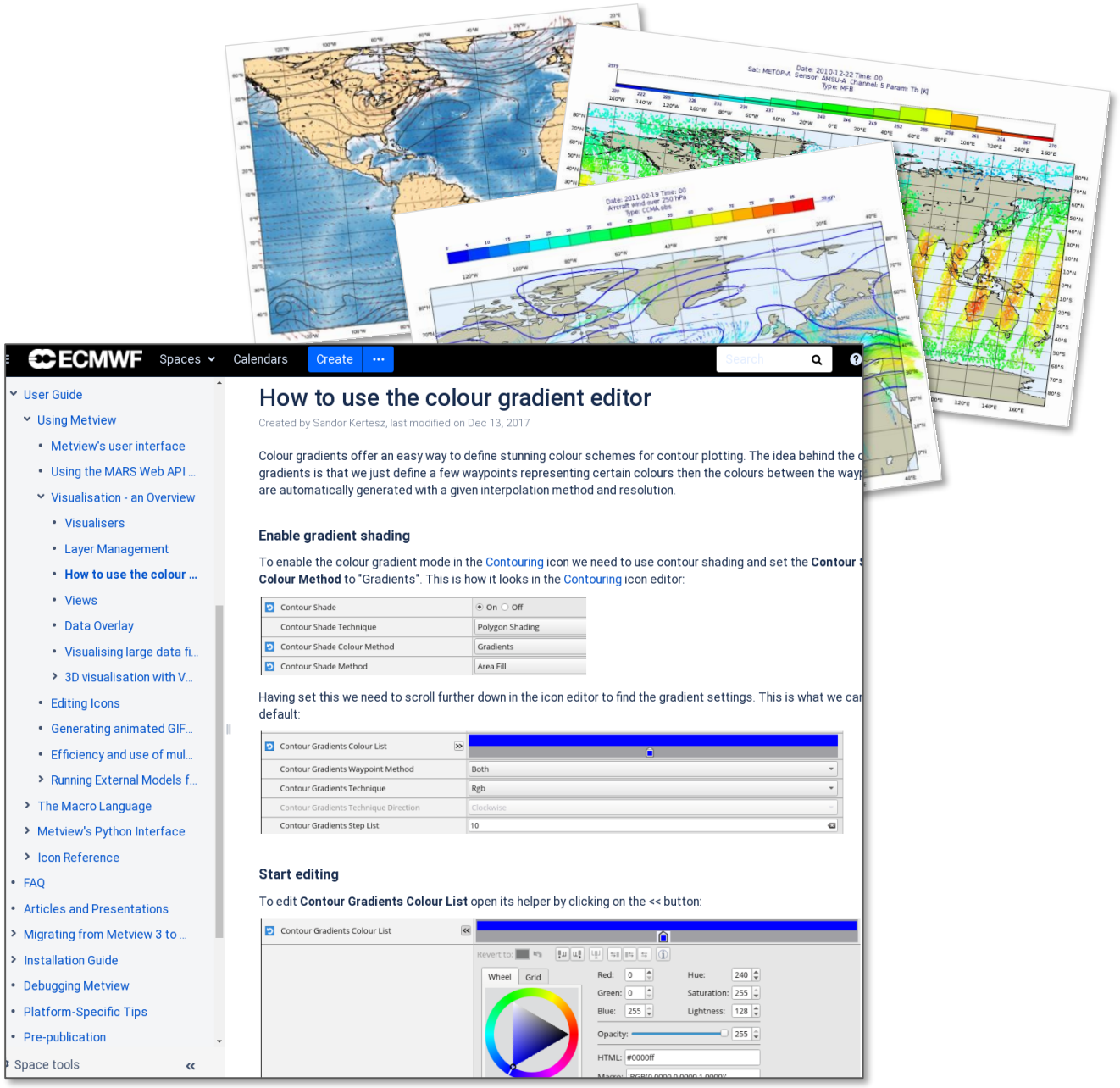
- Lots of material online including longer tutorials





For more information…

- Ask for help:
  - Software.Support@ecmwf.int
- Visit our web pages:
  - http://confluence.ecmwf.int/metview



Questions?