Welcome

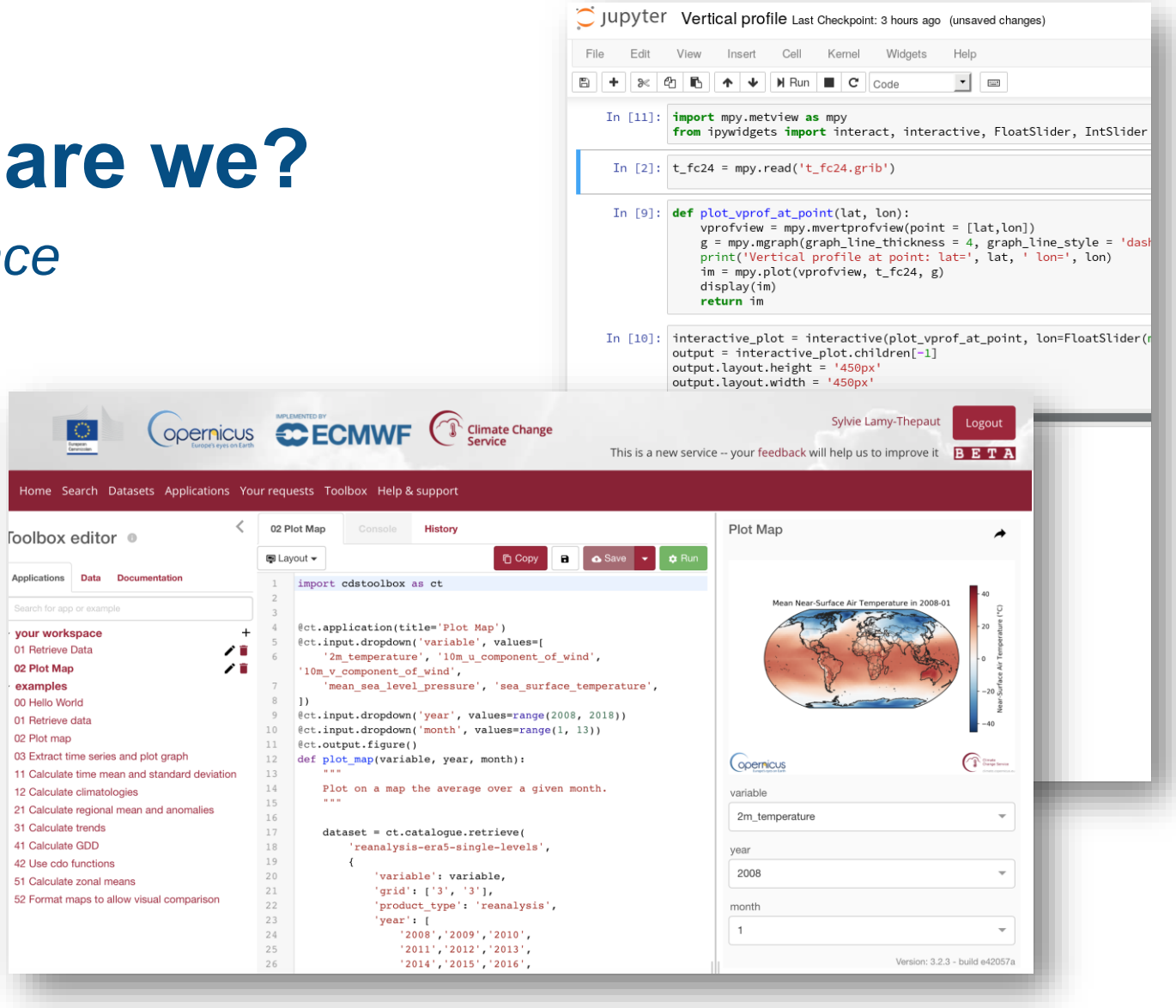
WORKSHOP ON **30–31 OCTOBER 2018**

Developing Python frameworks for earth system sciences

One year on … where are we?

What we learned then and done since

Stephan Siemen
Development Section, ECMWF



ECMWF

What we learned from last year's workshop

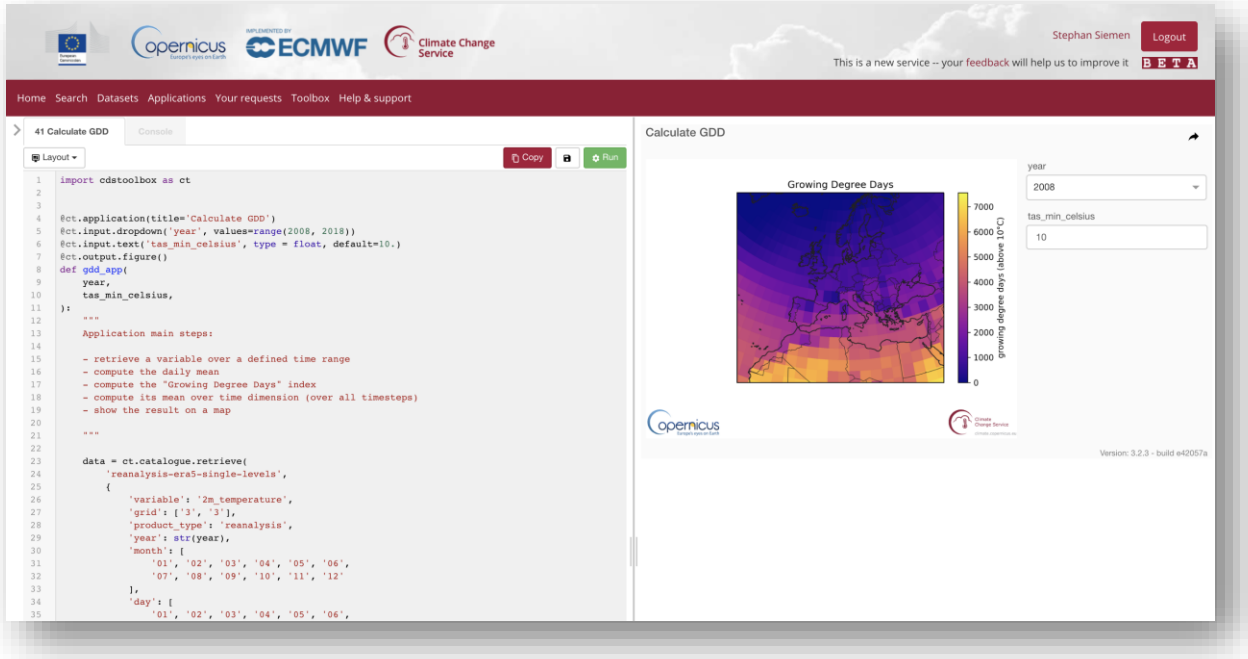
- There are already many good efforts and solutions out there
  - → Many good "wheels" which do not need to be reinvented
- Confirmation of our direction for developments
  - → Handle fields through *xarray*
- Building a community is more then just releasing software under Open Source
  - → 'Open Source' versus 'Open Development'

CDS (toolbox) release earlier this year

- New portal to find / download and work with Copernicus Climate data

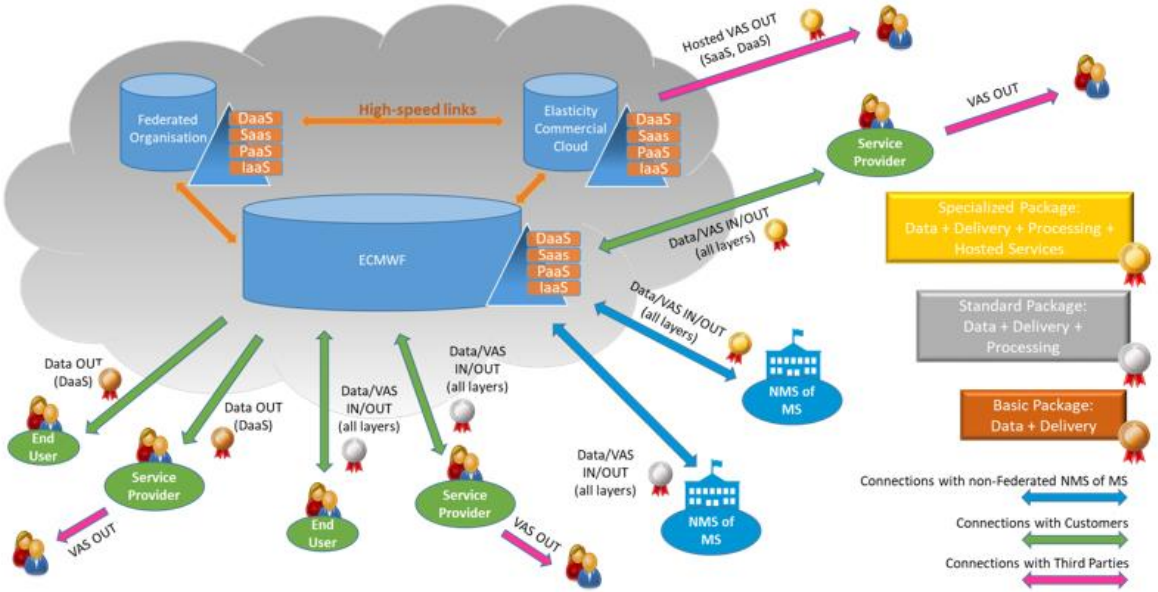
- Try it out yourself:

  https://cds.climate.copernicus.eu



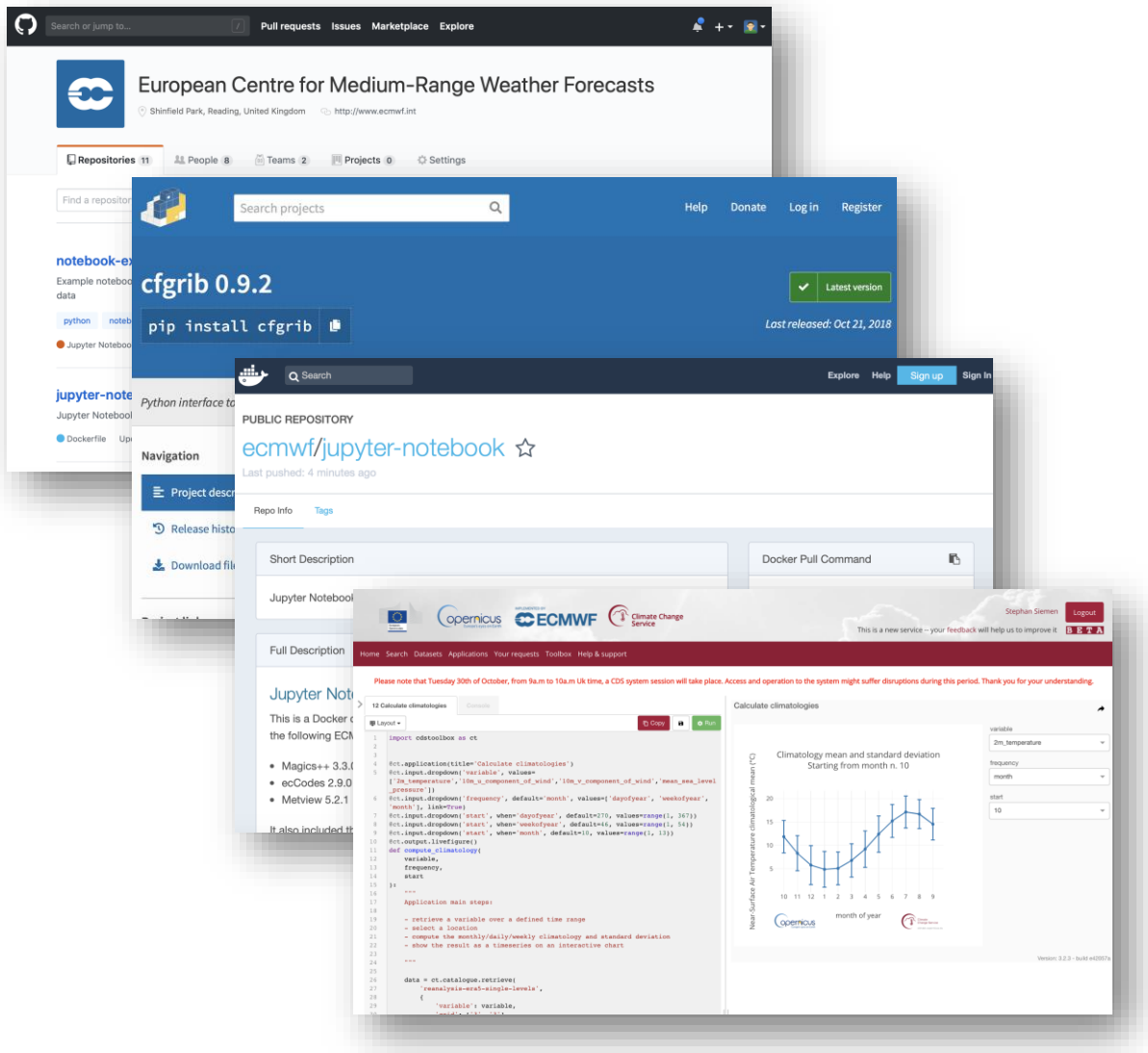
New opportunities through cloud services

- ECMWF looks together with its partners on providing private clouds close to data
  - *European Weather Cloud* with EUMETSAT
  - Copernicus *WEkEO* DIAS in co-operation with EUMETSAT & Mercator Ocean
- Making it easier for users to work with ECMWF forecast & Copernicus data
  - And Python will play an important role here …



Make software easily available within existing frameworks

- Source code & examples on GitHub (CaaS)
- Packages need be on PyPi and Conda
- ECMWF Python software on DockerHub
- CDS Toolbox - SaaS



We need to improve on the first generation Python interfaces

- SWIG is very limited
- Support Python3 with all our packages by end of 2018
- Need to support multiple Python version at once
- Very close to Fortran/C interfaces
  - Not taken into account any Pythonic ideas

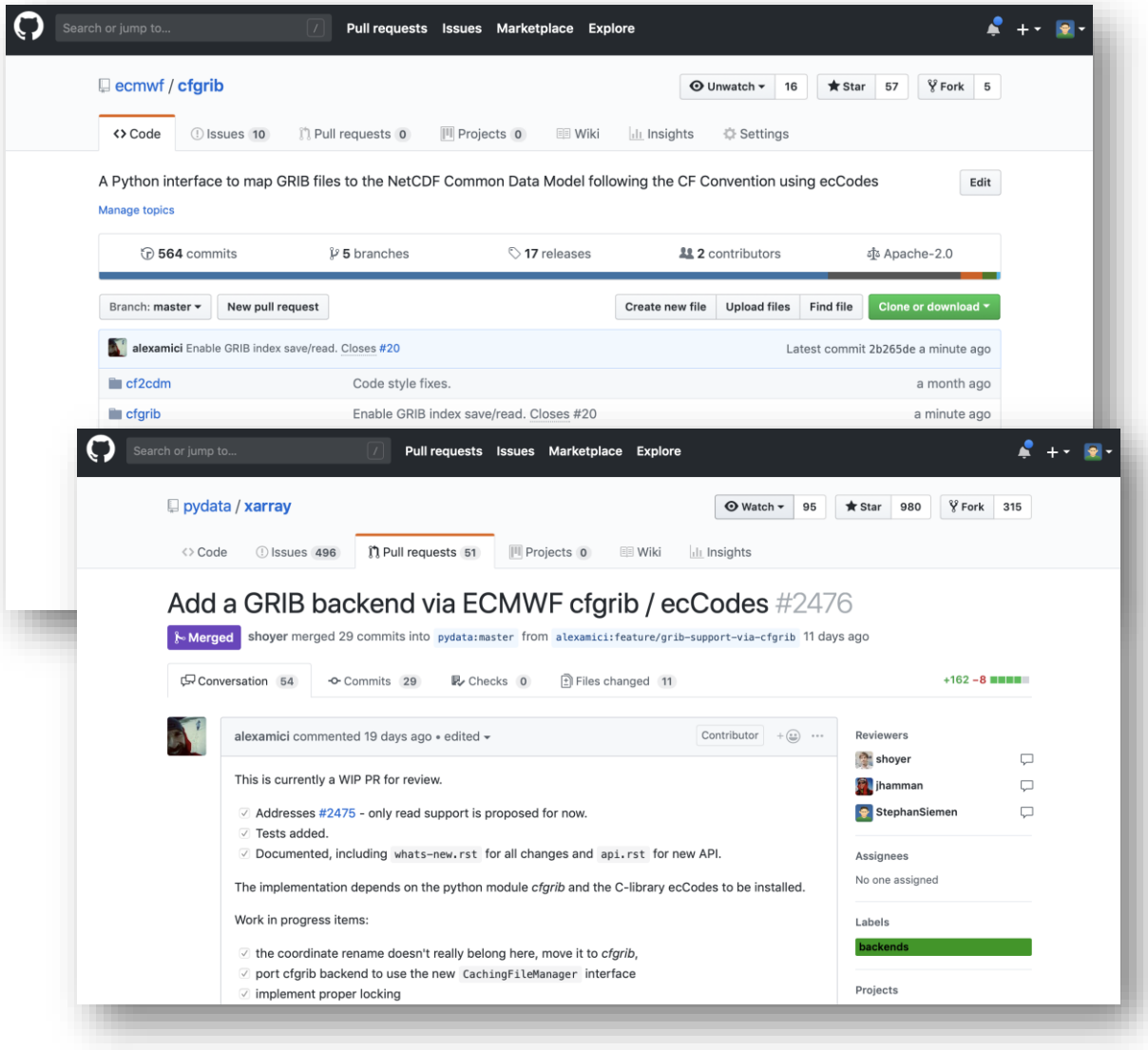
→Consequence: split Python interfaces from software packages

→ Package in PyPi

cfgrib – linking xarray and ecCodes

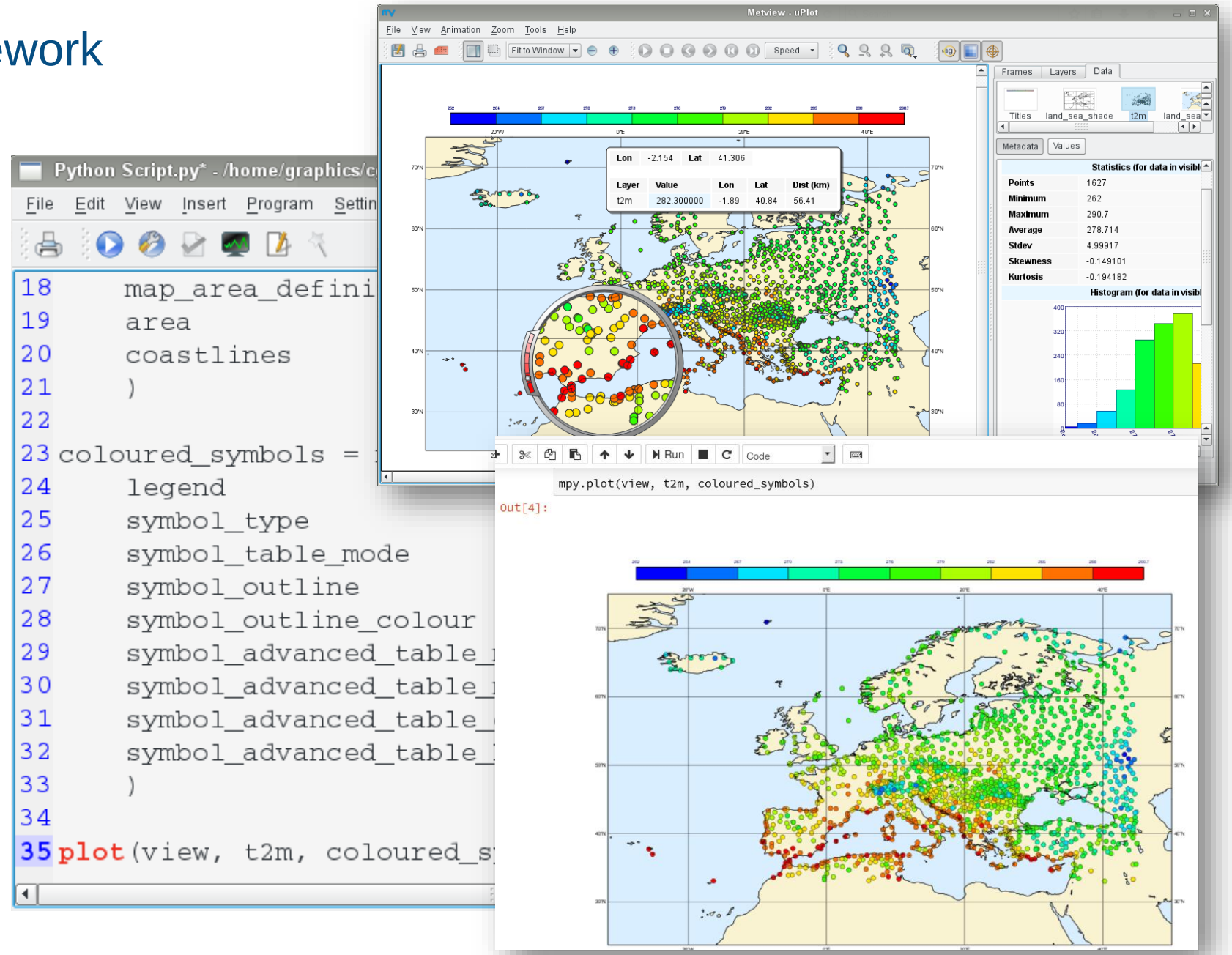
- To embrace xarray for all our field data, we needed to know that we could handle all our GRIB 1 & 2 data
  - Therefore it was important for us to have a solution based on ecCodes

- This work allows users of GRIB data now to have a high level access of our data and use all PyData community has to offer

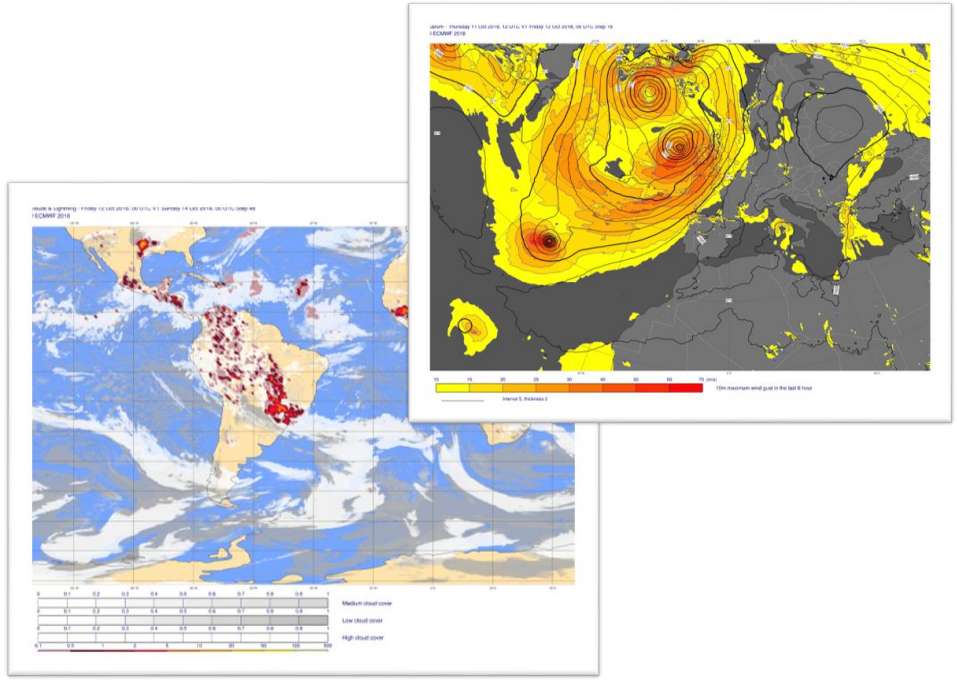


The Metview Python framework

- Aim is to allow users of Metview to use easily the power of Python but still have all functionality of Metview; including visualisation
- We have now beta release with all Metview functionality available from Python
- Close co-operation CDS toolbox



Improve automatic styling



- With more distributed (cloud) environments, it is even more of a challenge to keep a common (corporate) look-and-feel

- We need to make easy for users to apply these styles
  - Best automatic
  - Take meta data information from xarray
  - Detect internal representation of (NetCDF) data to perform geo referencing

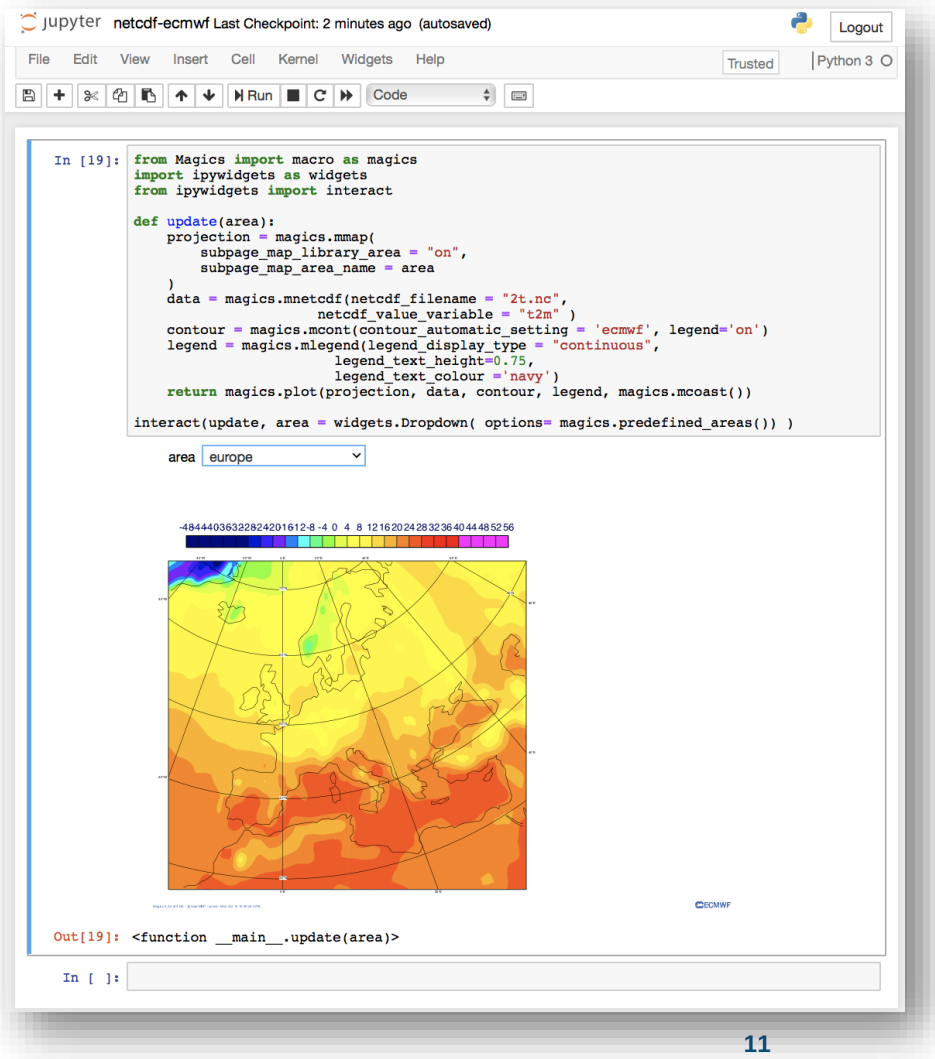
Challenges ahead …

- Would be nice if we could provide examples of our frameworks working together
  - E.g. Metview with MetPy

- What is the best way to offer a user environment? (especially in the cloud)
  - Jupyter/Lab?
  - Docker?
  - SaaS – JupyterHub / CDS toolbox?

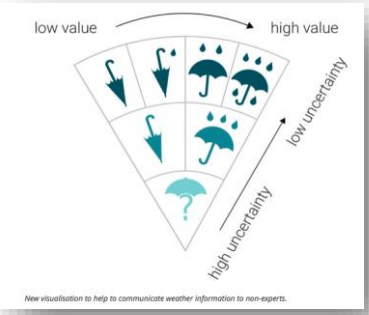
- Reproducible workflows → Workshop @ECMWF in 2019

- How we can scale heavy processing tasks
  - Dask & Pangeo … especially in the cloud

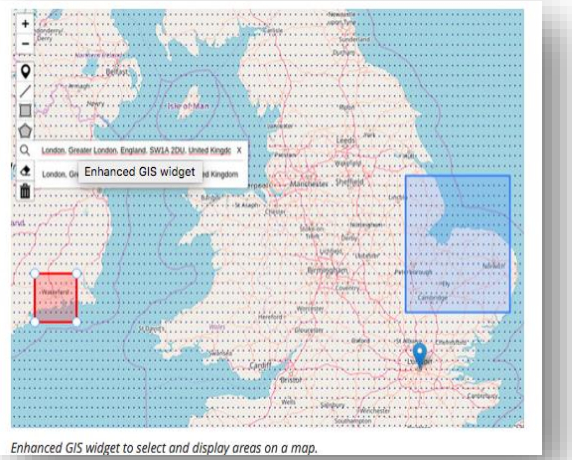


ECMWF Summer of Weather Code (ESoWC)

Innovative visualisations



Enhanced GIS widget



- First year was a great success to engage with new communities
- We look for new ideas by January 2019 …

  … perhaps our discussions will come up with some ideas …