CPAS User Guide - Result Visualization

ClusterTech Limited

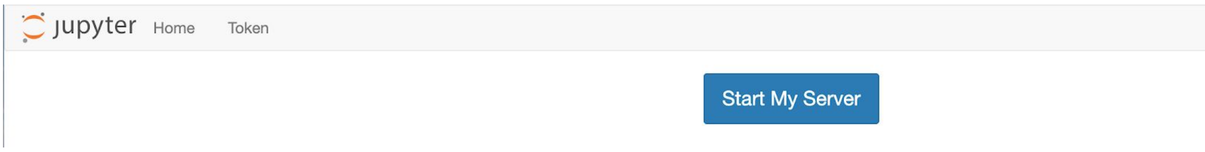
July 2020

0. Before you start

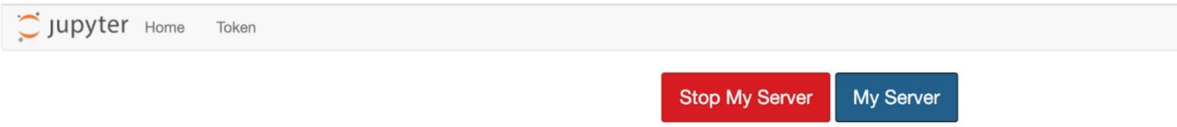
0.1 Spawn your Jupyter server

After you sign in CPAS, click “Visualization” on the top menu.

You shall see

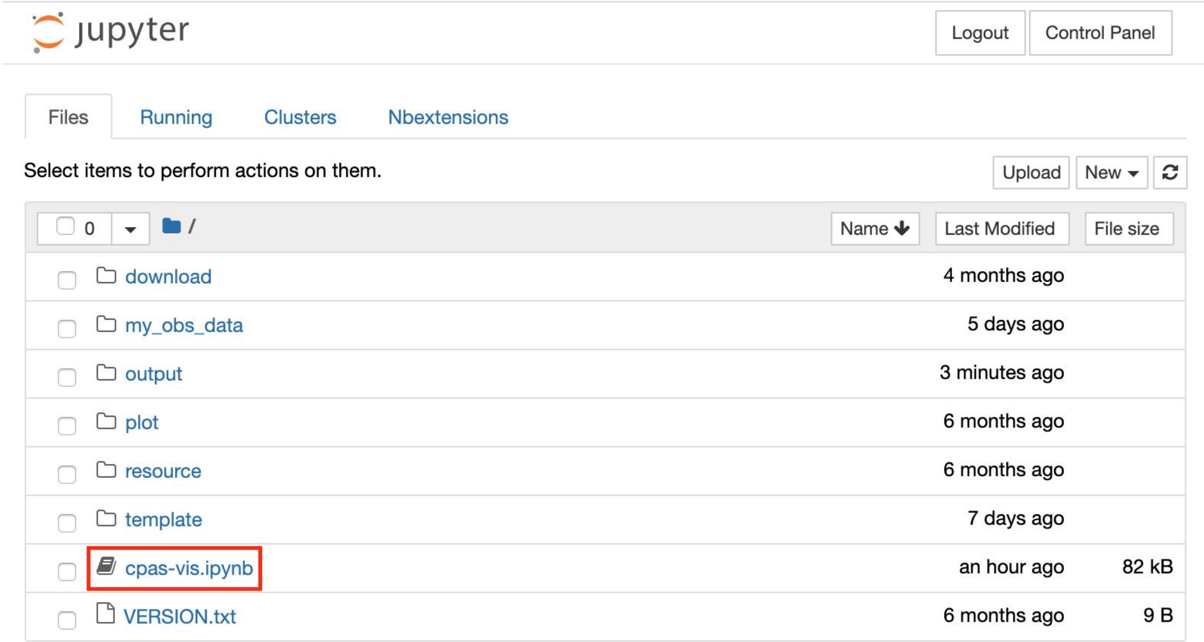


or

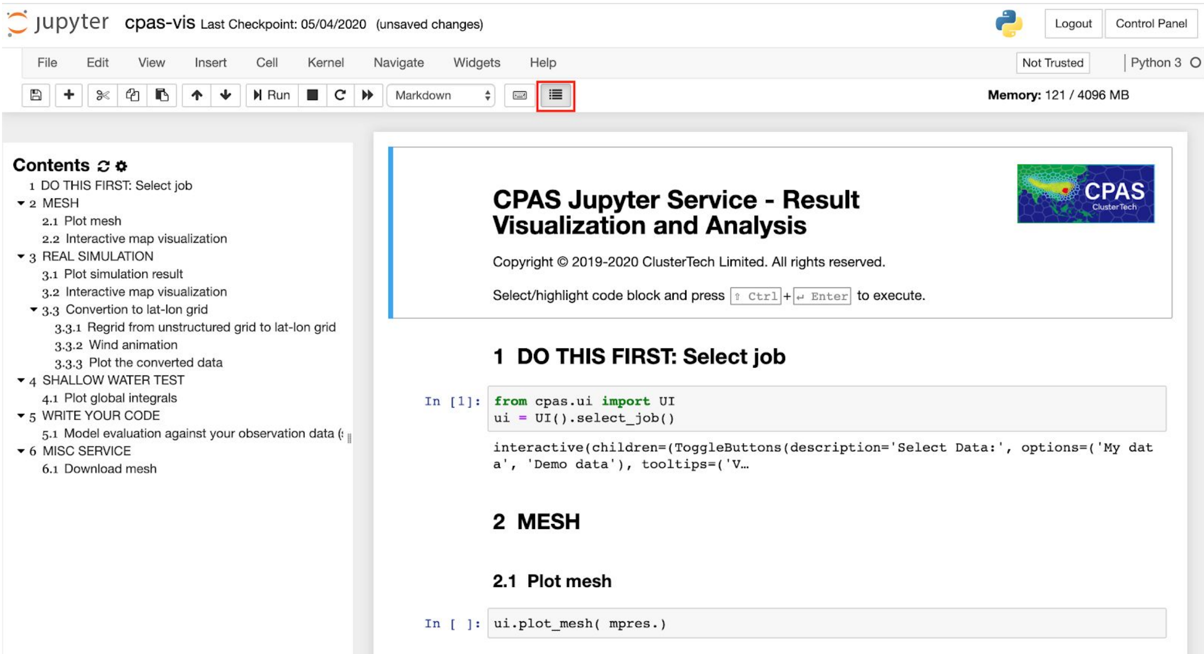


Note: If you see an error message like “401 Unauthorized”. Go back to console.cpas.earth (click the “Back” button in your browser), sign-in and try again.

Click “Start My Server” or “My Server”, and wait a few seconds for the progress bar for spawning the server to complete, you will see:



Double click “cpas-vis.ipynb” to open it. You should see:

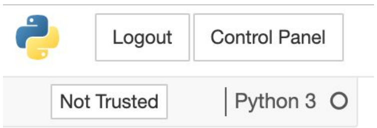


If you cannot see the left Table of Contents, click the button on the toolbar.

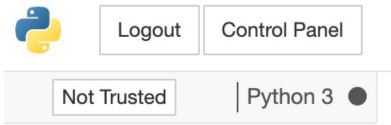
0.2 Tips on Jupyter system usage

0.2.1 Kernel busy

There is a circle beside the label “Python 3” near the top right corner. If the circle turns black, it means that the Python 3 kernel in the backend is processing the data. Just wait.



Kernel Idle

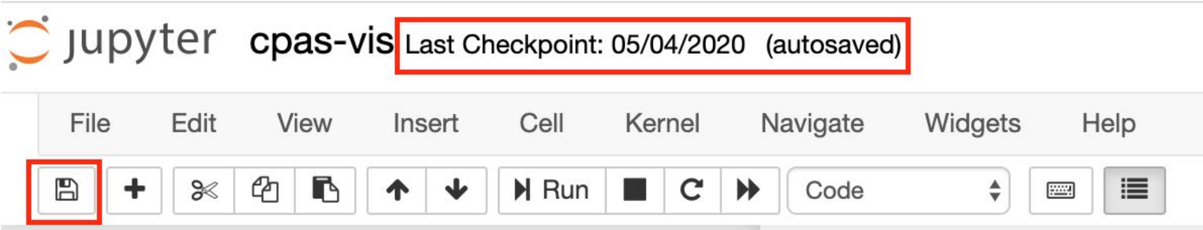


Kernel Busy

If you think the Jupyter system is not responsive, please kindly check whether the kernel in the backend is busy processing.

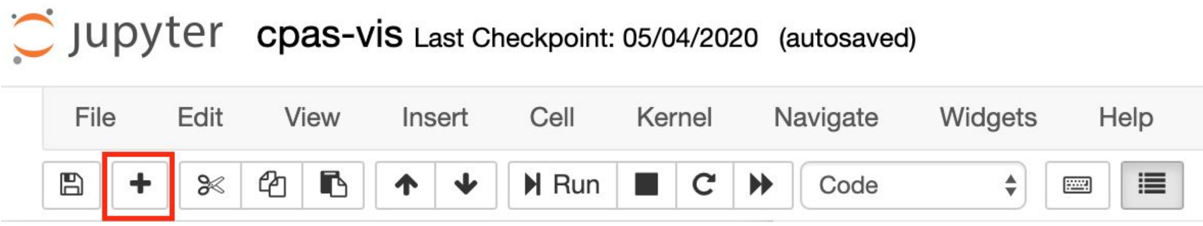
0.2.2 Save / Autosave

The content of your Notebook can be edited. Your edited content can be manually saved by clicking the “save” button, or autosaved by the system.



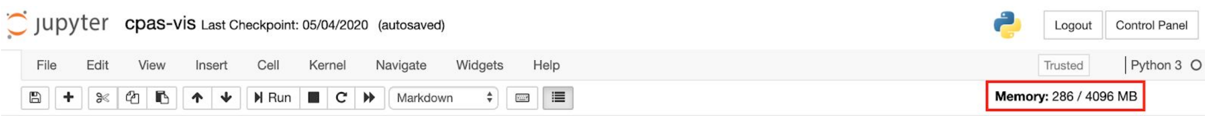
0.2.3 Add you code block

You are free to add your code block by clicking the “+” button.



0.2.4 Memory Usage

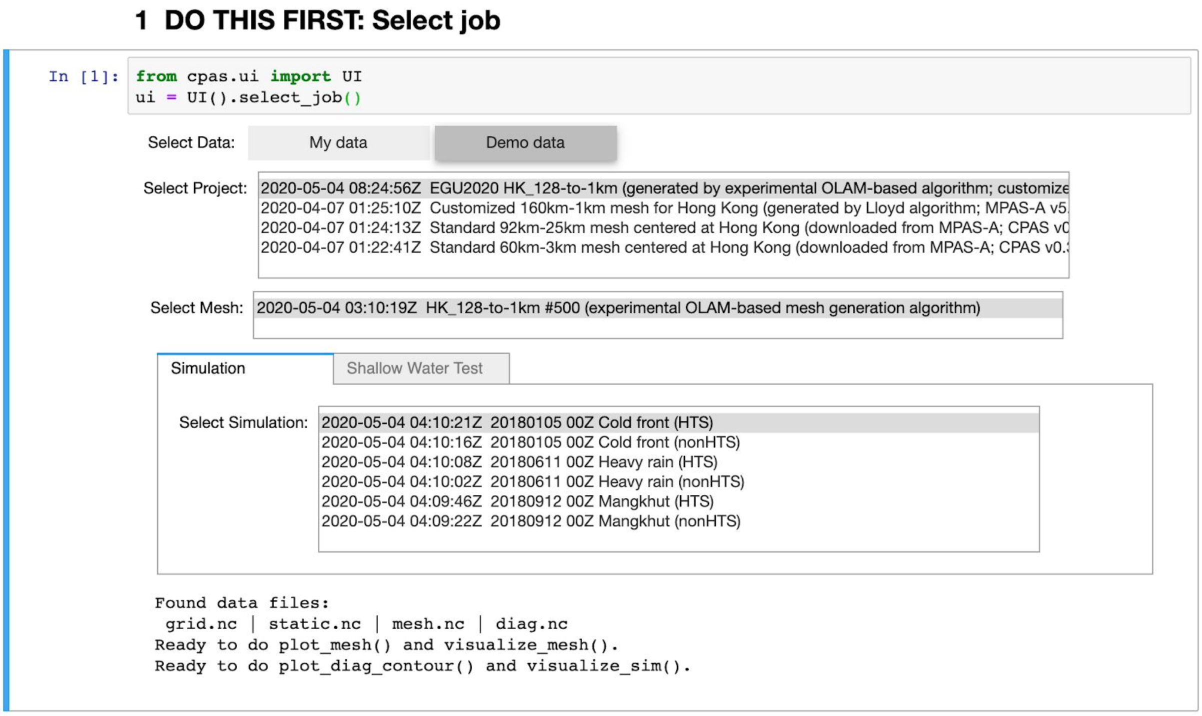
You may note your usage of memory and the limit (in accordance to your subscription level).



If you run out of memory, consider changing to a higher subscription level. Your Jupyter server may be killed if your memory usage exceeds the limit. Restart the kernel in that case.

1. Select Job

Following the instructions on the notebook, highlight the code block with select_job() and press Ctrl+Enter. A set of widgets for selecting the jobs you previously ordered will be shown. You may select the Mesh Generation / Real Simulation / Shallow Water Test job you submitted (My data) or from public data available (Demo data).



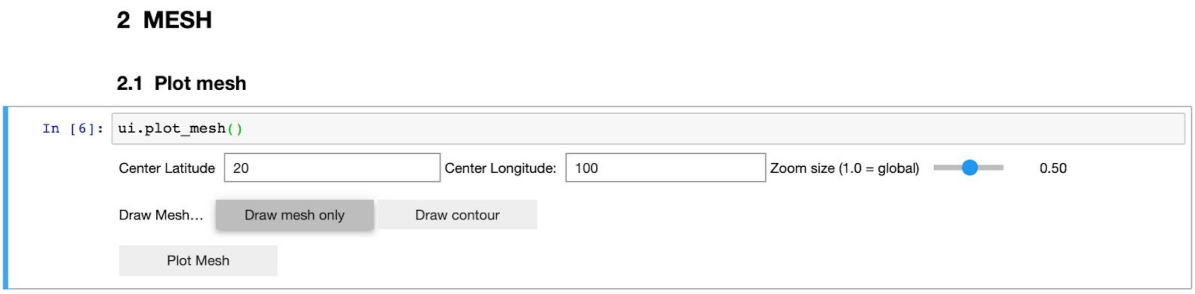
Note: The "My data" tab would not appear until you have a successful job submission and the Server restarted to mount your result directory.

The demo data consists of some data from ClusterTech's R&D work presented on the [Publication and Public Presentations](#) page.

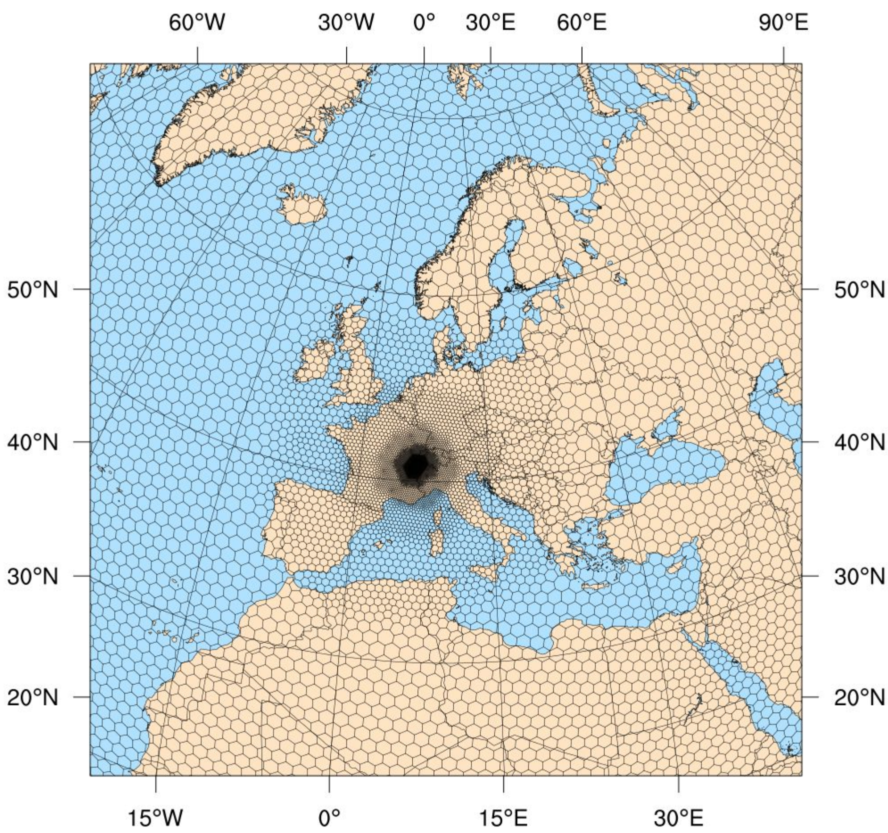
2. Mesh visualization

2.1 Plot mesh

Highlight the code block with plot_mesh() and press Ctrl+Enter. A set of widgets will appear for you to specify the options on the scope to plot. Enter the latitude and longitude for the center and drag the slider to zoom.



Have “Draw mesh only” selected by default, click the “Plot Mesh” button. After a while, the plot will appear.



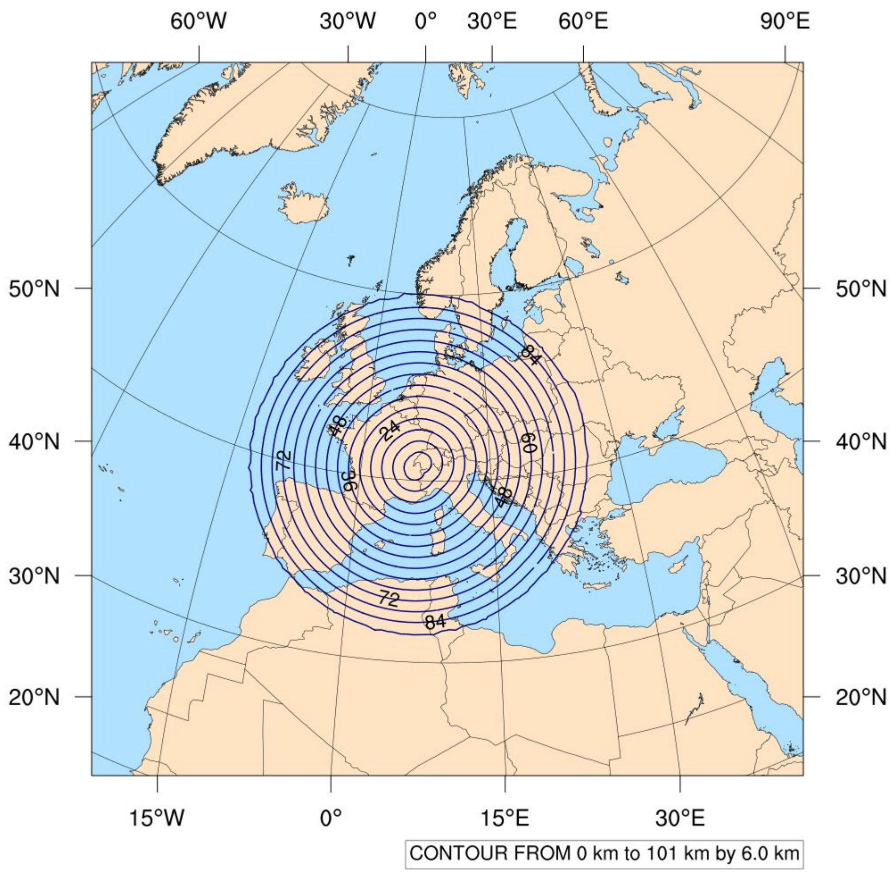
For more details of geographical information, please use visualization of mesh in an interactive map.

If you choose “Draw contour”, and click “Plot Mesh” again, Center Latitude 50 Center Longitude: 10 Zoom size (1.0 = global) 0.50

Draw Mesh… Draw mesh only Draw contour

Plot Mesh

A plot with contours of resolution will appear soon.

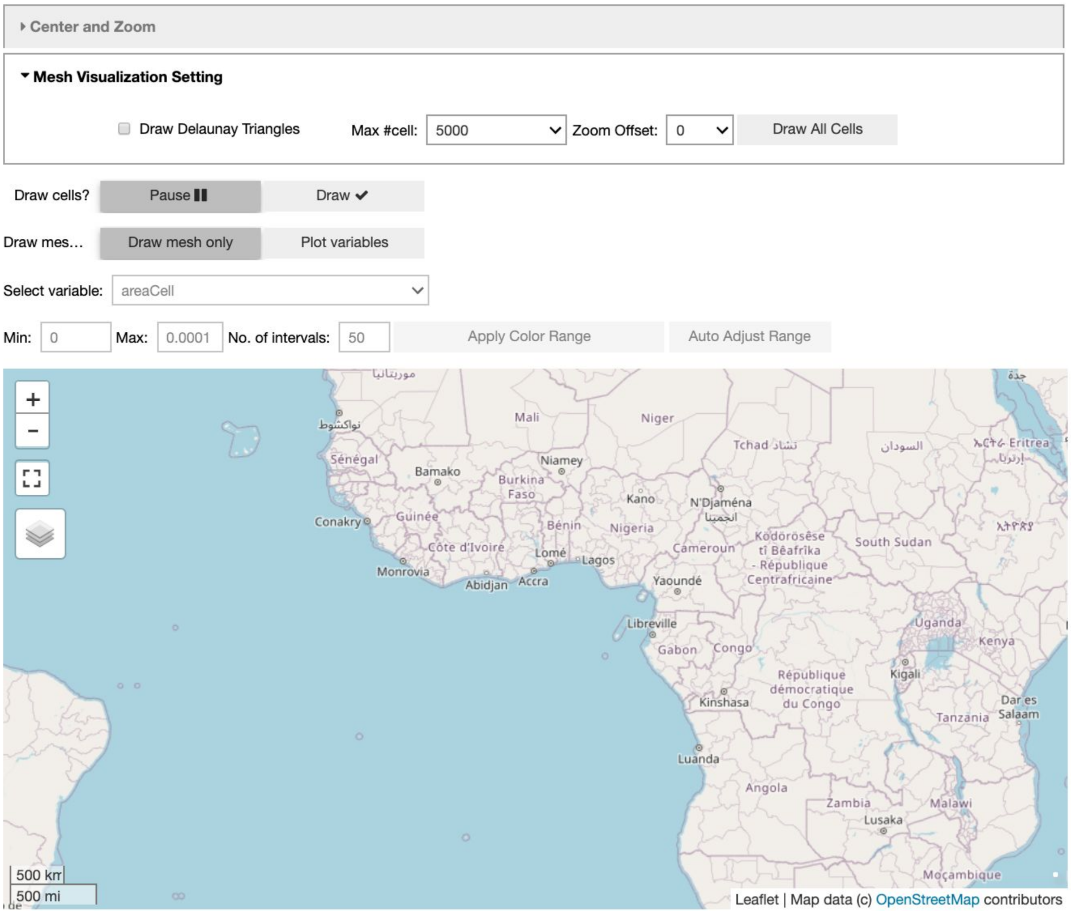


2.2 Visualize mesh in interactive map

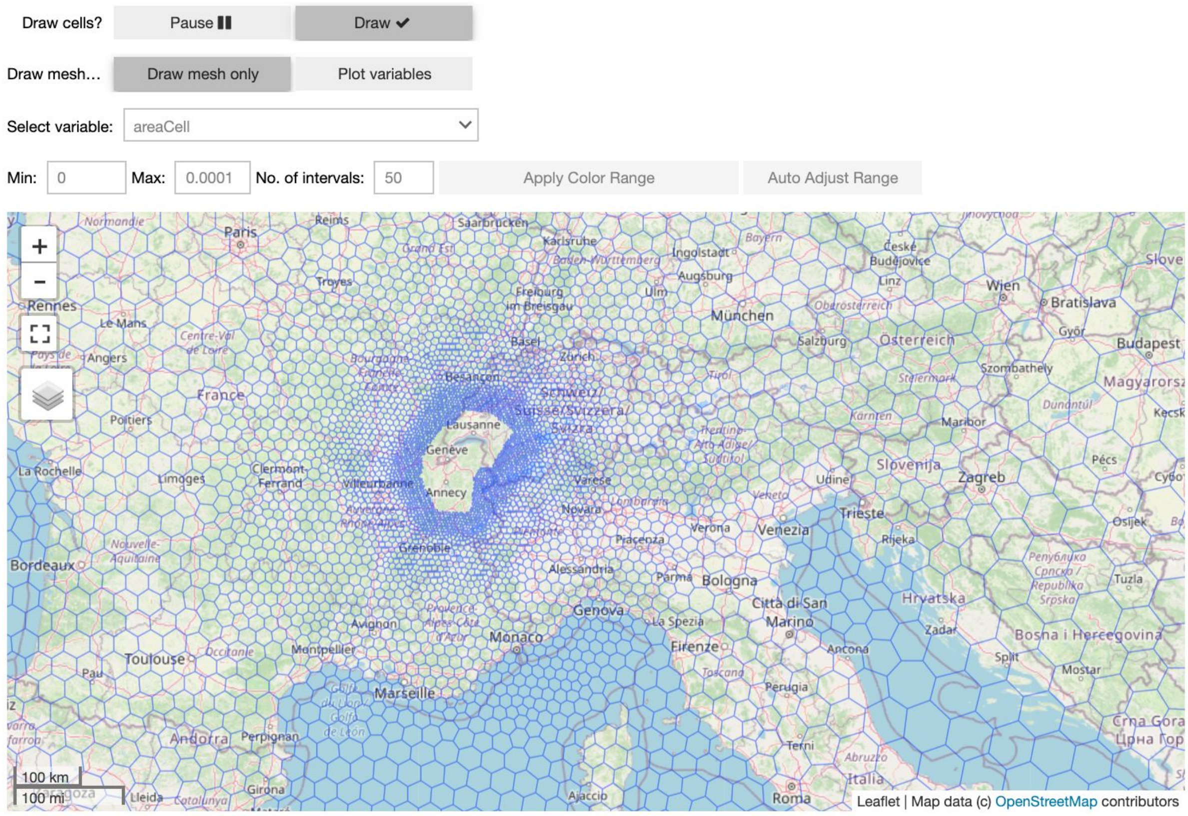
Highlight the code block with visualize_mesh() and press Ctrl+Enter. A progress bar on loading will appear, and after loading completes, you will see:

2.2 Interactive map visualization

```
In [8]: ui.visualize_mesh(height='500px')
```

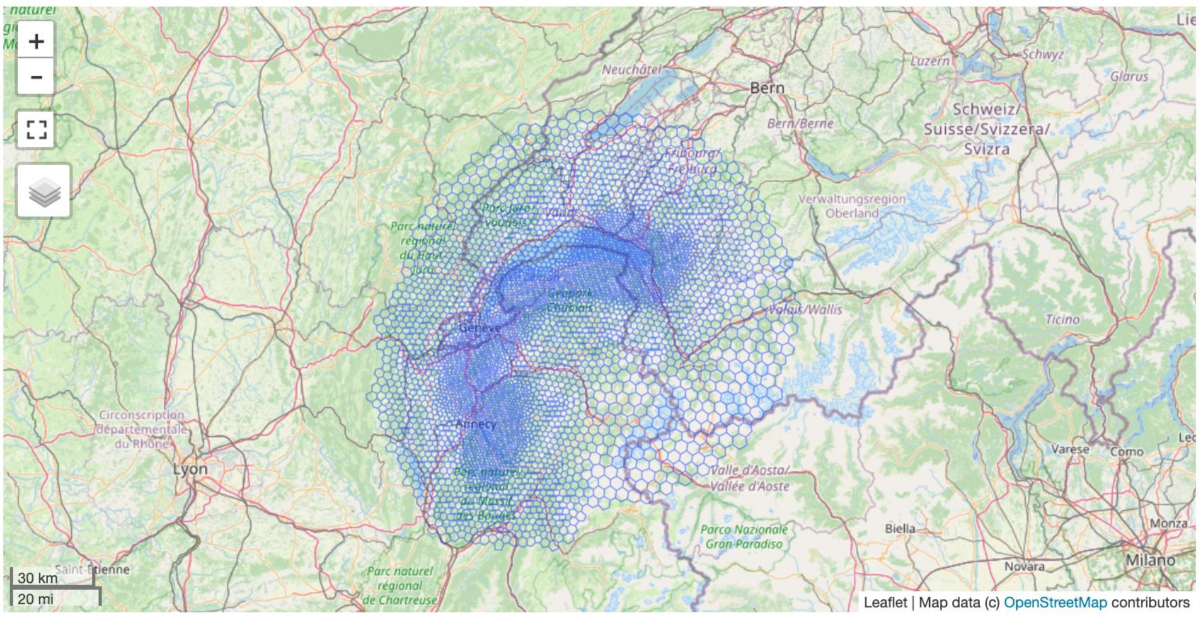


The interactive map is in Pause state. You are expected to drag and zoom in the map to the location you are interested. Then, click the Draw button to toggle to auto-redraw the state.

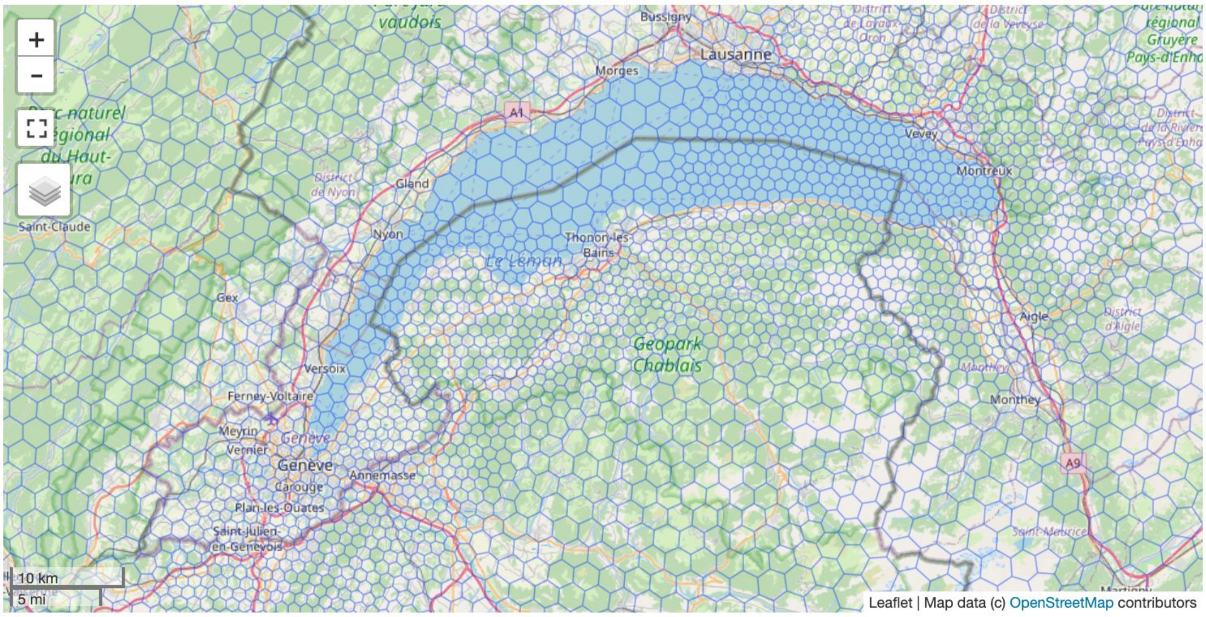


With the initial default setting, drawing of the cells on the map may take a few seconds. However, if regions with no cells drawn appear, it's because we have set a limit on the maximum number of cells to draw in order to keep your browser responsive. The drawing algorithm selects cells of appropriate sizes to draw, and avoids drawing cells that stick together and form a patch.

Zooming in, the system will take some time to redraw and you will see results similar to:

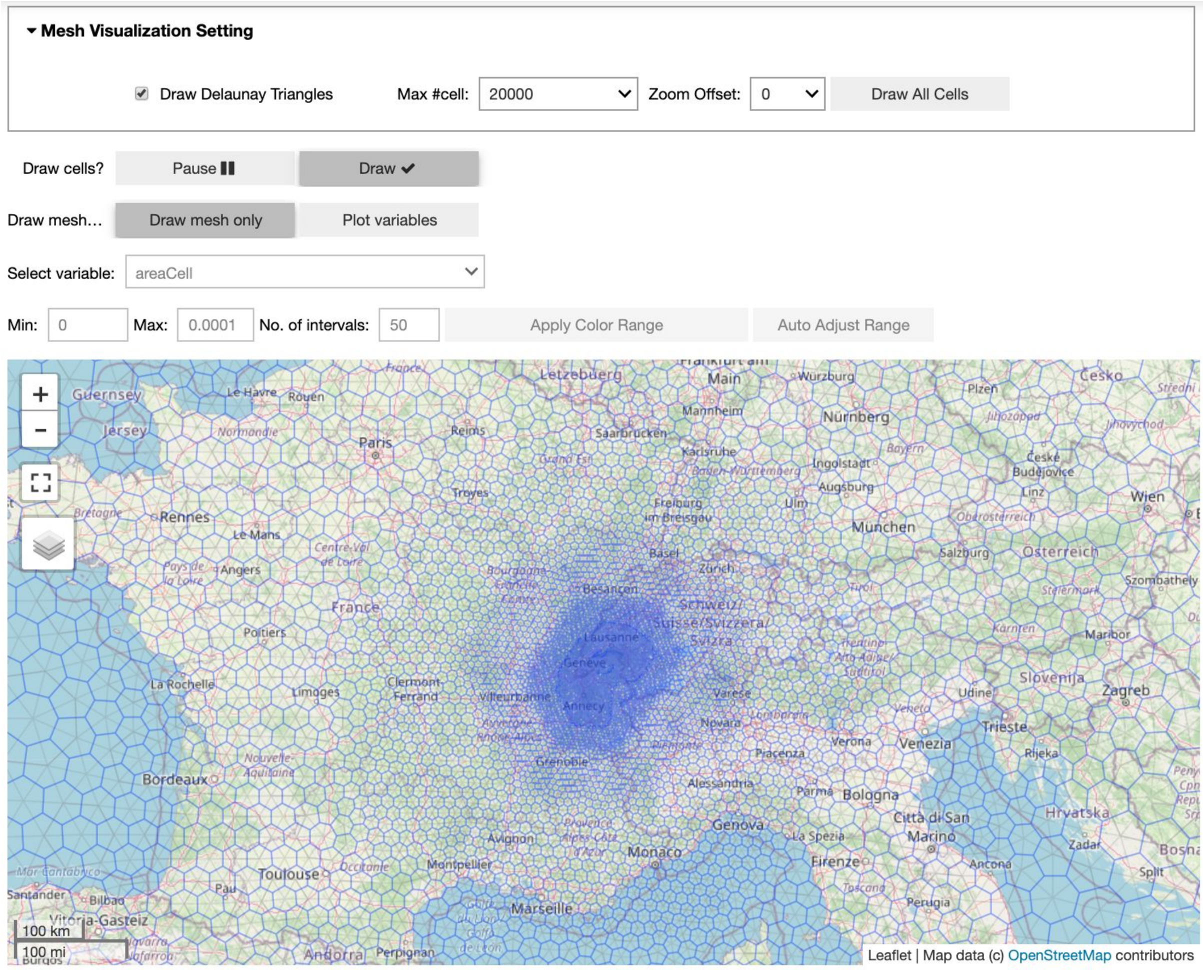


All cells of interest appear now, and you can continue to zoom in. You can also click the “Pause” button before zooming in in order to make the browser respond faster.



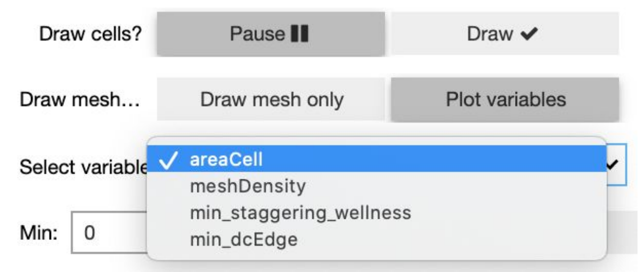
(This is Geneva Switzerland! CPAS technology was firstly announced here at Meteorological Technology World EXPO 2019.)

In case you would like to fine-tune the interaction behaviour, look into the “Mesh Visualization Setting” accordion. Max #cell is the limit mentioned earlier. Setting it to a larger value plots more cells but slows down browser response. “Zoom Offset” controls the choice of sizes of cells to appear. The “Draw All Cells” button disregards the limit on the number of cells, but will be very small if the number of cells is large. Last but not least, “Draw Delaunay Triangles” (light grey) helps you inspect the quality of the mesh (whether the triangles are close to equilateral, or if any obtuse triangle exists).

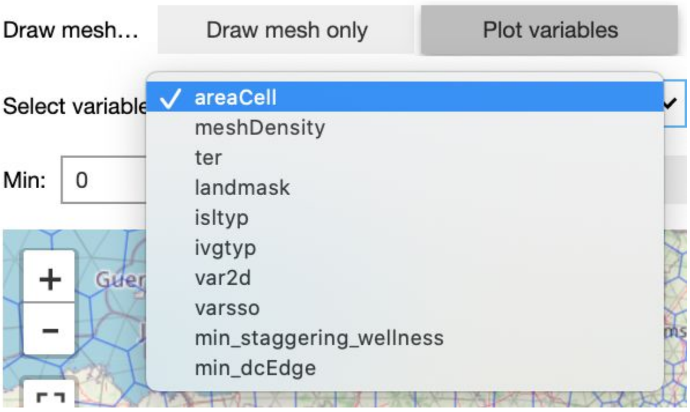


2.3 Visualize static data in interactive map

If you choose “Plot variables”, you may see



And if you have already executed a Real Simulation job with the mesh, you may see more variables:

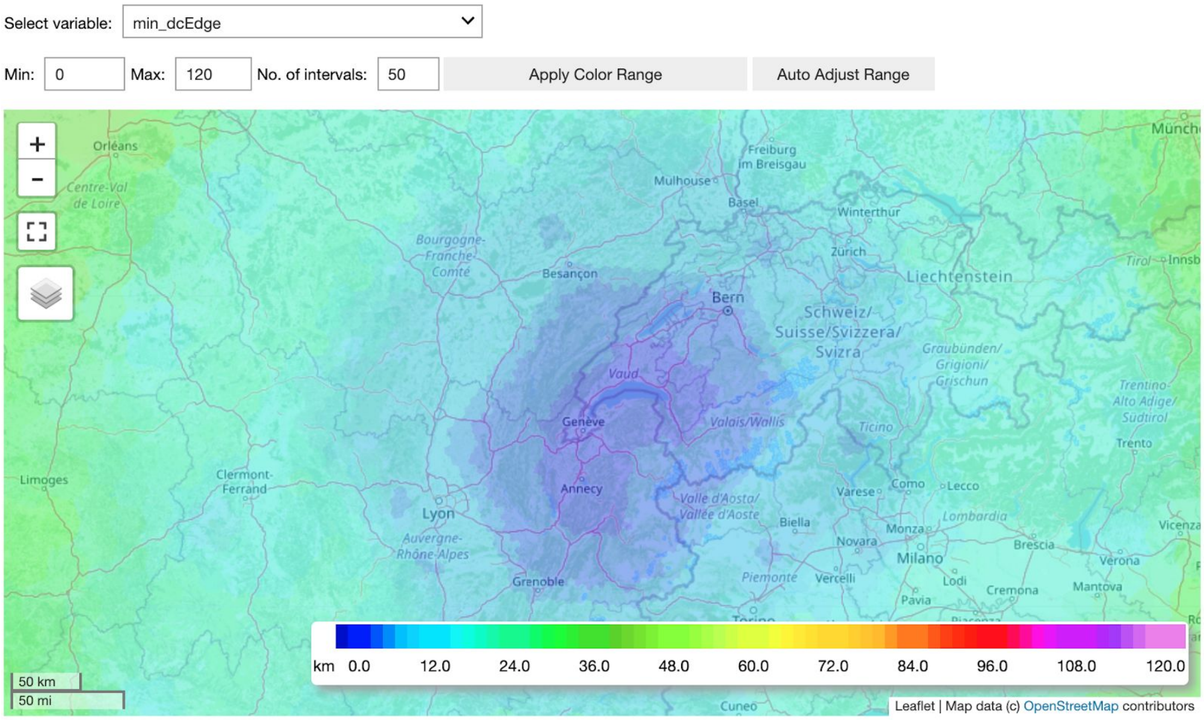


Variable	Unit	Description	Reference
areaCell	km^2	Area in square meters for a given cell of the primary mesh.	MPAS Mesh Specification
meshDensity	unitless	The value of the generating density function at each cell center.	MPAS Mesh Specification
ter	m	Terrain height	MPAS-Atmosphere Model User's Guide
landmask	unitless	Land-ocean mask (1=land ; 0=ocean)	MPAS-Atmosphere Model User's Guide
isltyp	unitless	Dominant soil category	MPAS-Atmosphere Model User's Guide
ivgtyp	unitless	Dominant vegetation category	MPAS-Atmosphere Model User's Guide
var2d	m^2	Variance of orography	MPAS-Atmosphere Model User's Guide
varsso (experimental)	m^2	Variance of sub-grid-scale orography.	WPS V4 Geographical Static Data Downloads Page
min_staggering_wellness	unitless	For each edge, the staggering wellness is defined as $1.0 - \frac{\lvert dv1Edge - dv2Edge \rvert}{dvEdge}$ For a cell, min_staggering_wellness is the smallest staggering wellness among its edges. The range is from 0.0 to 1.0. 0.0 means ill staggering (obtuse or	Similar metrics are used in https://doi.org/10.1175/MWR-D-12-00236.1 and https://doi.org/10.1016/j.jcp.2018.07.025

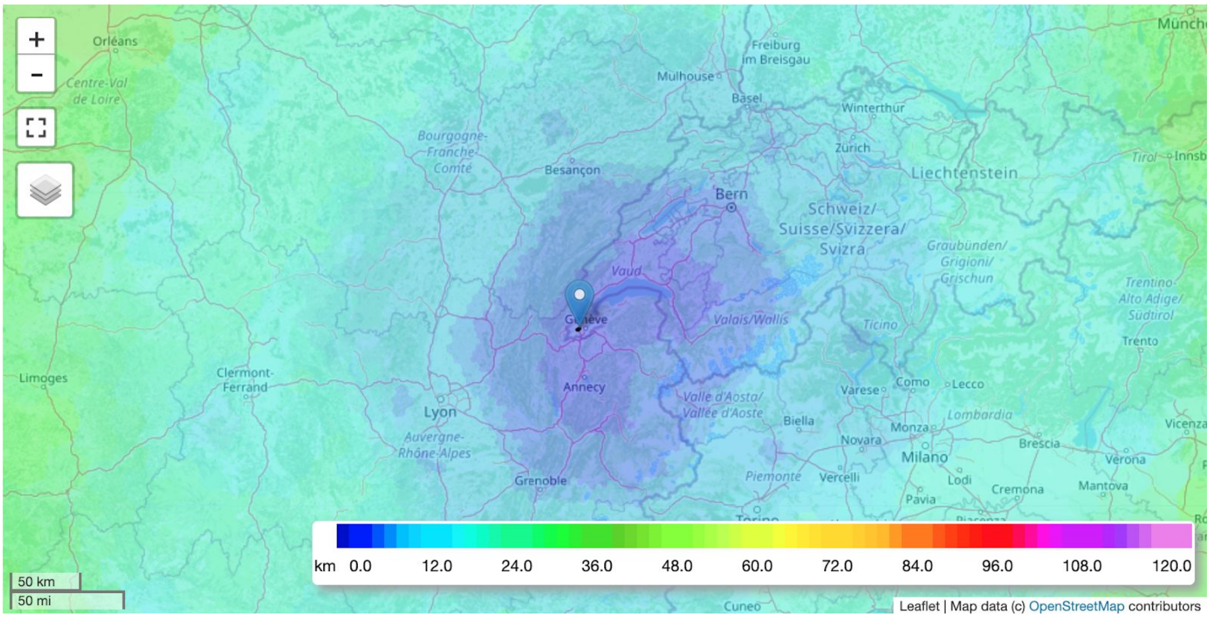
		right-angled Delaunay triangle occurs); 1.0 means perfect staggering.	
min_dcEdge	km	The shortest grid spacing with its neighbours, i.e. the smallest dcEdge associated with a cell.	[MPAS Mesh Specification]()

Table 1: Variables about mesh

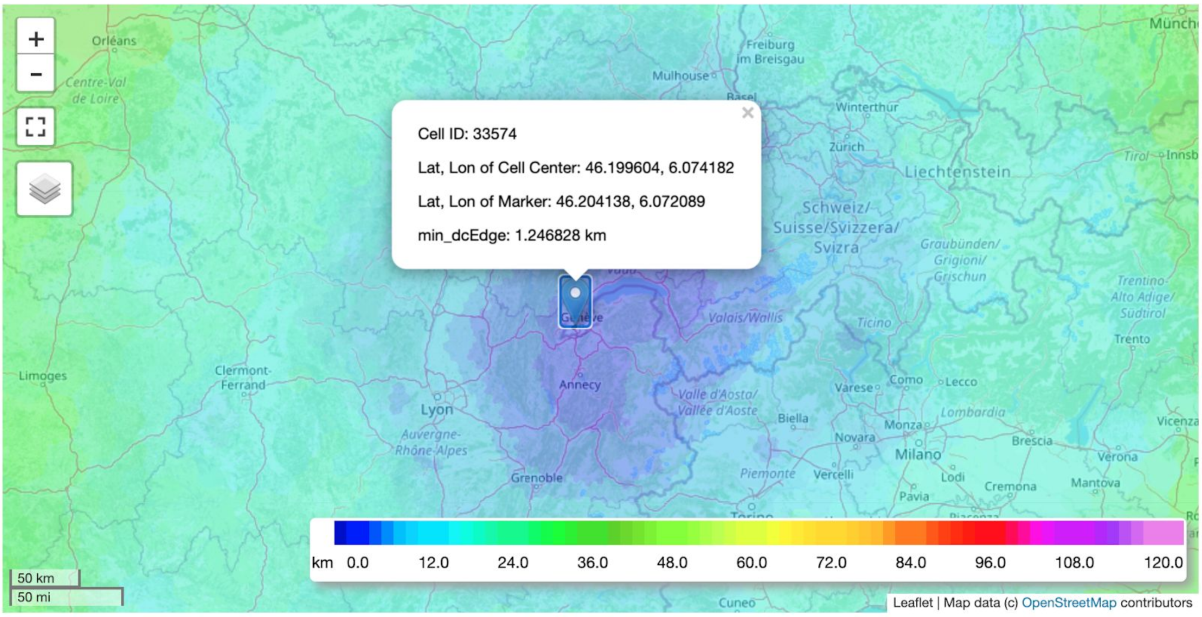
Take min_dcEdge as an example,



If you click on a position in a map panel, a marker will appear.



Click at the marker, detailed information would appear:



3. Simulation results visualization

The following list of variables may be available for plotting as contours or visualized in an interactive map.

Variable	Unit	Description	Reference
t2m	°C	2-meter temperature, converted to degree Celsius	MPAS-Atmosphere Model User's Guide
q2	kg kg^{-1}	2-meter specific humidity	MPAS-Atmosphere Model User's Guide
u10	m s^{-1}	10-meter zonal wind	MPAS-Atmosphere Model User's Guide
v10	m s^{-1}	10-meter meridional wind	MPAS-Atmosphere Model User's Guide
wspd10	m s^{-1}	10-meter wind speed, derived from u10 and v10 for CPAS users' convenience.	
mslp	hPa	Mean sea level pressure	core_atmosphere registry
rainc	mm	Accumulated convective precipitation	MPAS-Atmosphere Model User's Guide
rainnc	mm	Accumulated total grid-scale precipitation	MPAS-Atmosphere Model User's Guide
temperature_<xxx>hPa	°C	Temperature at <xxx> hPa pressure level	core_atmosphere registry
relhum_<xxx>hPa	%	Relative humidity at <xxx> hPa pressure level	core_atmosphere registry
uzonal_<xxx>hPa	m s^{-1}	Zonal wind at <xxx> hPa pressure level	core_atmosphere registry
umeridional_<xxx>hPa	m s^{-1}	Meridional wind at <xxx> hPa pressure level	core_atmosphere registry
wspd_<xxx>hPa	m s^{-1}	Wind speed at <xxx> hPa pressure level, derived from uzonal_<xxx>hPa and umeridional_<xxx>hPa for CPAS users' convenience.	

height_<xxx>hPa	m	Height at <xxx> hPa pressure level	core_atmosphere registry
w_500hPa	m s^{-1}	Vertical wind at 500hPa. Good for visualizing convection.	core_atmosphere registry
delta_rainc	mm	Difference of rainc of this time slide from the previous time slide.	
delta_rainnc	mm	Difference of rainnc of this time slide from the previous time slide.	
rainsum	mm	rainc + rainnc	
delta_rainsum	mm	Difference of rainsum of this time slide from the previous time slide.	
olrtoa	W m^{-2}	All-sky top-of-atmosphere outgoing longwave radiation flux	MPAS-Atmosphere Model User's Guide
vorticity_500hPa	s^{-1}	Relative vorticity at vertices	MPAS-Atmosphere Model User's Guide

Table 2: Variables about real simulation.

For variables with <xxx>hPa suffix, usually the following pressure levels are available.

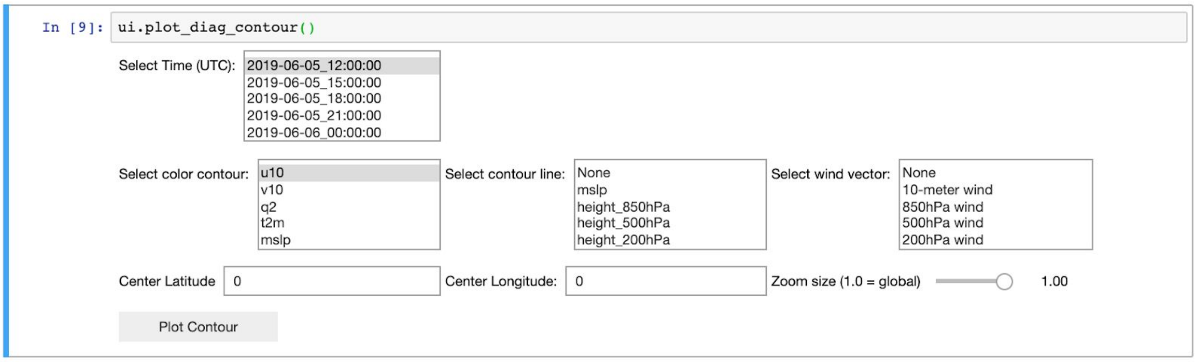
Pressure level	**Typical analysis usage**
850hPa	Low level thermal advection, vapour transport
500hPa	Vorticity, ridge and trough movement
200hPa	Jet stream, convergence and divergence

3.1 Plot contours

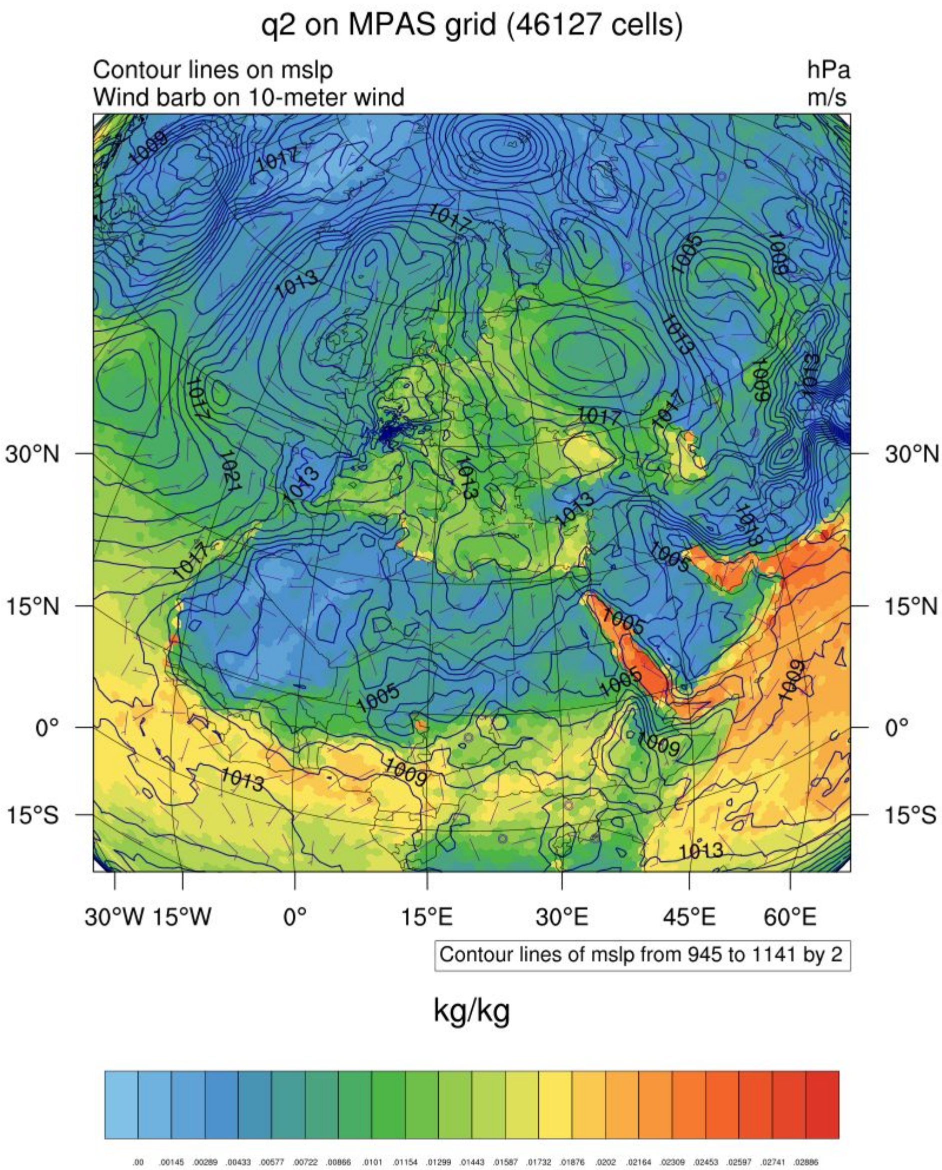
Highlight the code block with plot_mesh() and press Ctrl+Enter. A set of widgets will appear for you to specify the options on the scope to plot. Select the time slide to plot. Select up to 3 variables you may plot as color contour, line contour and wind vector. Enter the latitude and longitude for the center and drag the slider to zoom.

3 REAL SIMULATION

3.1 Plot simulation result

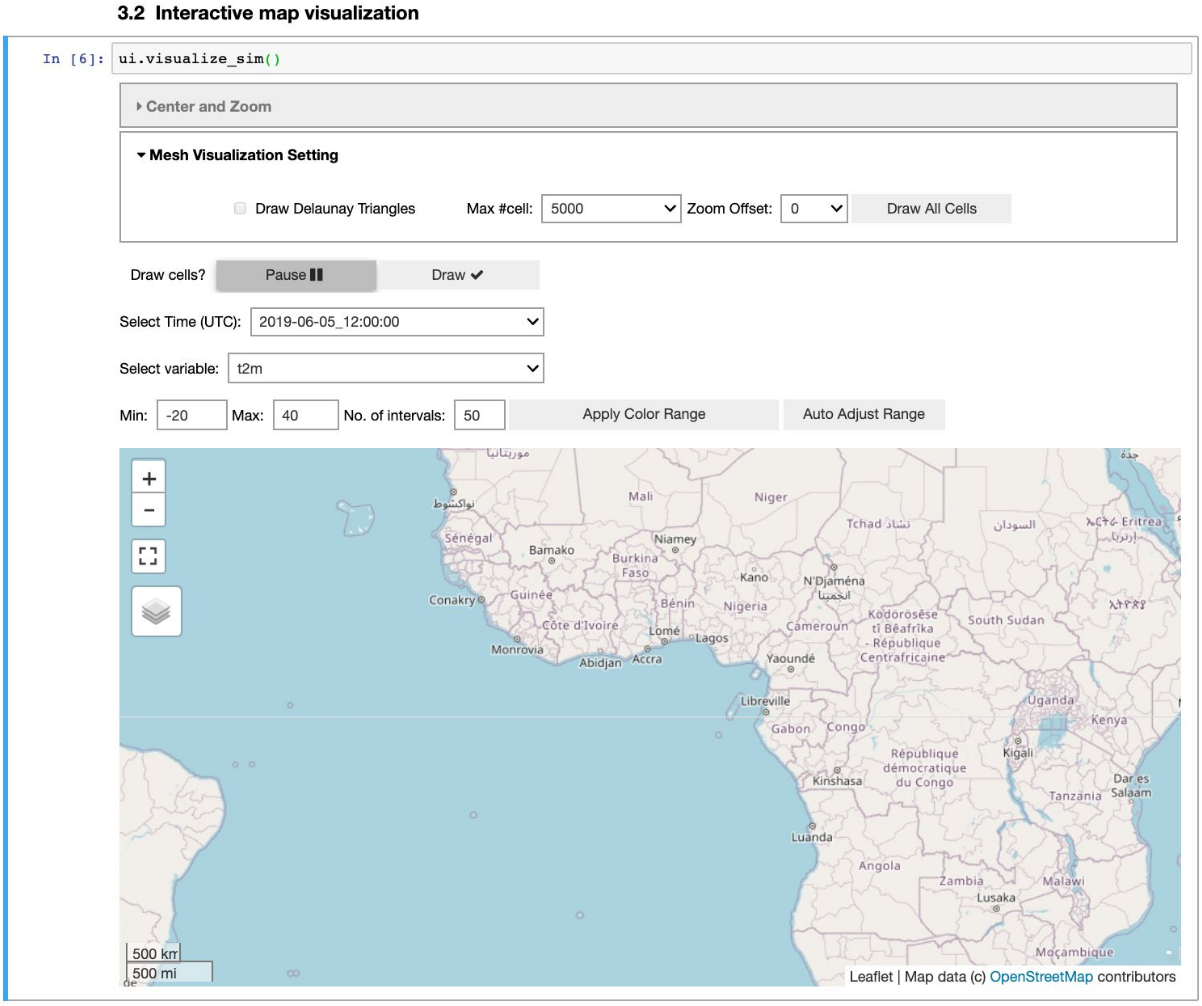


Then click the “Plot Mesh” button. The plot will appear soon.

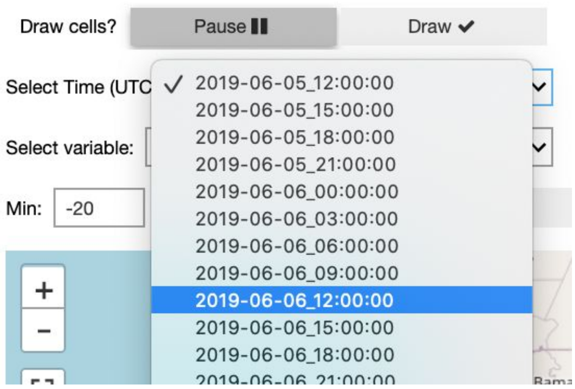


3.2 Visualize simulation results in interactive map

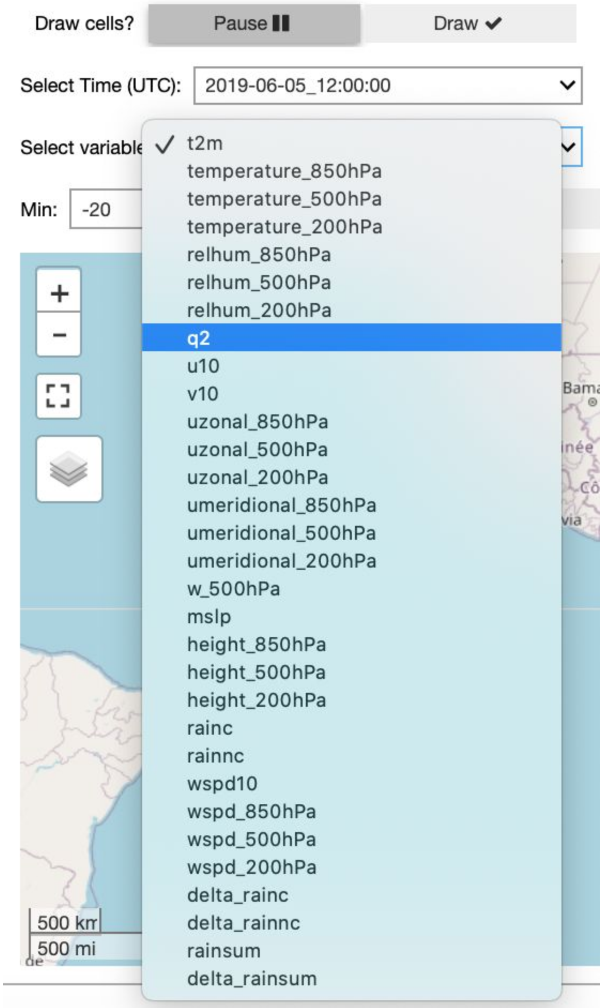
Highlight the code block with visualize_sim() and press Ctrl+Enter. A progress bar on loading will appear, and after loading completes, you will see:



Pulling down the “Select Time (UTC)” box, you will see time slides that your Real Simulation job outputs diagnostic data. Select the time slide you want to visualize.

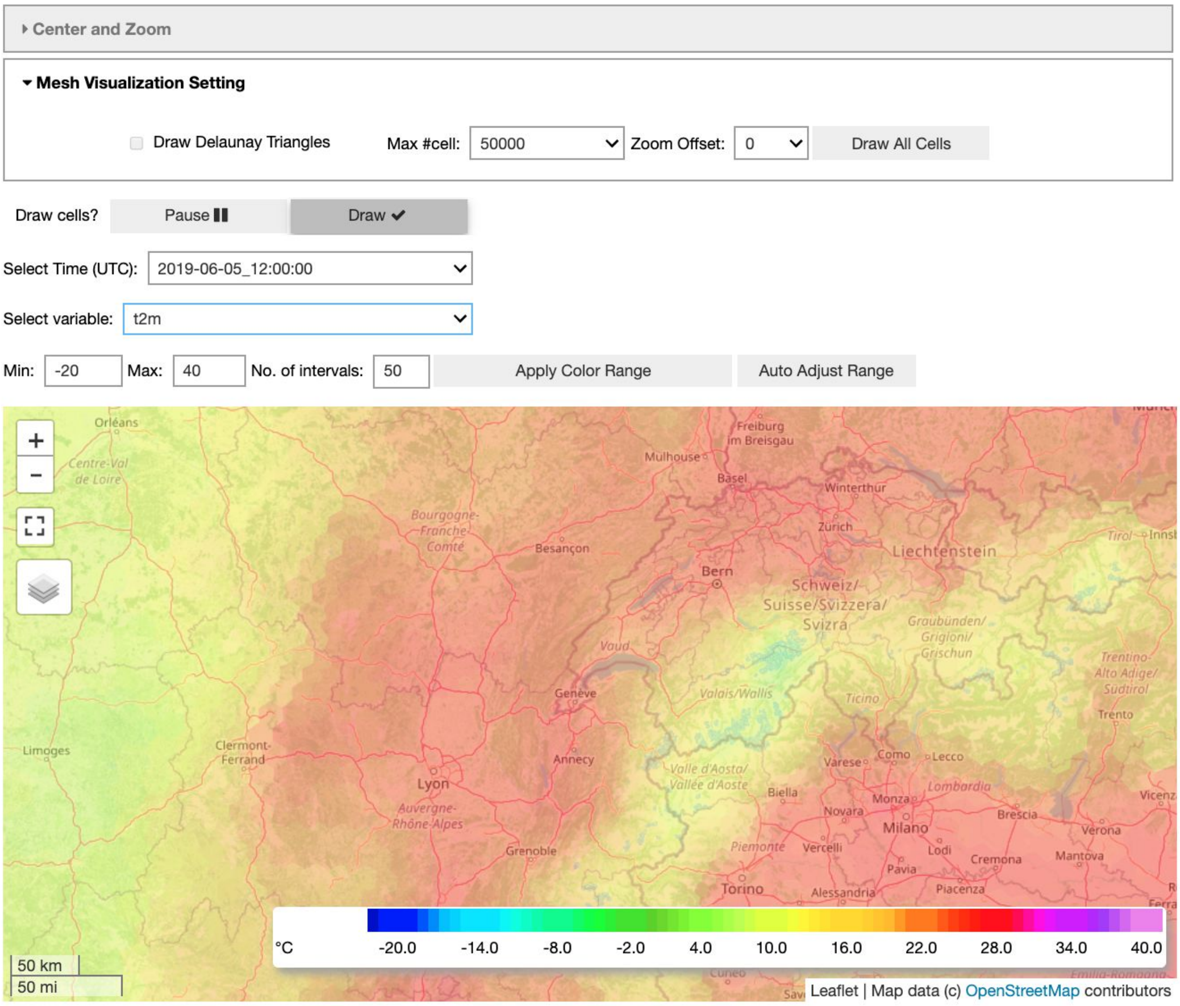


Pull down the “Select variable” pull down box. You will see a list of variable names that may be available to visualize. Select the variable (see Table 2) you want to visualize.

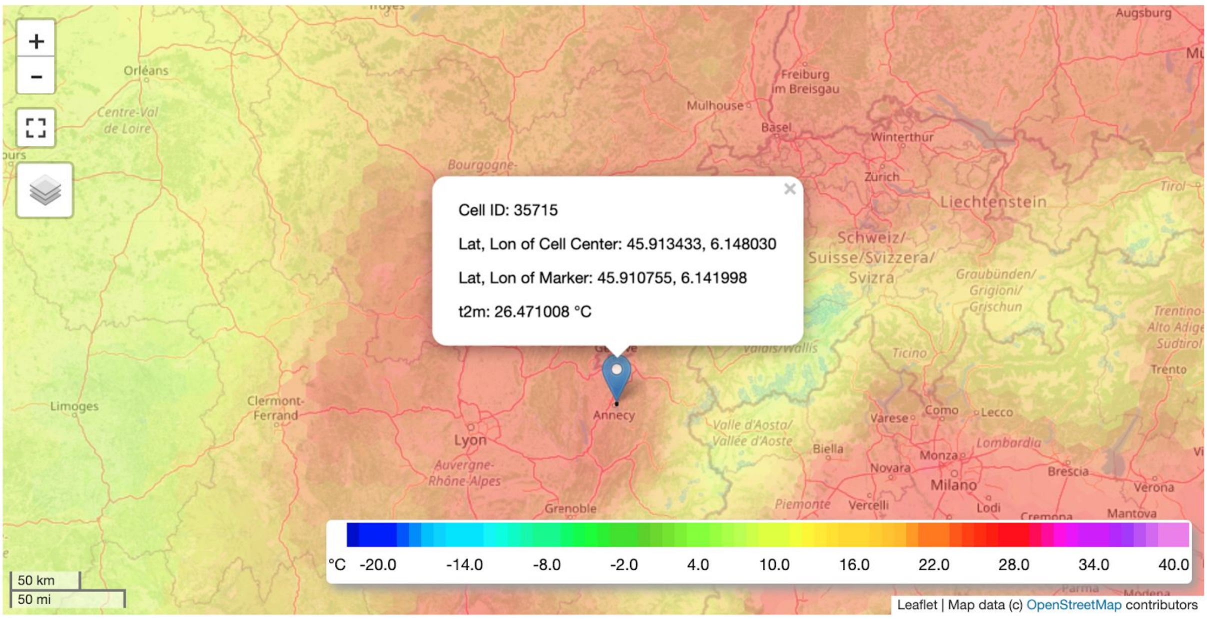


Other operations are similar to those for visualizing mesh. An example of visualizing t2m is below:

```
ui.visualize_sim()
```



Click on the map to position the marker, and display the detailed information:



3.3 Conversion to lat-lon grid

Many analysis and visualization tools are for lat-lon grid, for example, the pretty ipyleaflet Velocity package for visualizing wind with animation.

3.3.1 Regrid from unstructured grid

Read the code in the cell:

3.3.1 Regrid from unstructured grid to lat-lon grid

This generates a separate nc file with listed variables in lat-lon grid.

```
In [10]: # Set configs for conversion (e.g. output filename, variables, target grid)
         output_fname = 'output/diag_latlon.nc'
         var_list = ['mslp', 't2m', 'q2', 'u10', 'v10',
                     'uzonal_850hPa', 'umeridional_850hPa',
                     'uzonal_500hPa', 'umeridional_500hPa',
                     'uzonal_200hPa', 'umeridional_200hPa']
         # Other available variables:
         # 'height_850hPa', 'temperature_850hPa', 'relhum_850hPa',
         # 'height_500hPa', 'temperature_500hPa', 'relhum_500hPa',
         # 'height_200hPa', 'temperature_200hPa', 'relhum_200hPa',
         # 'w_500hPa', 'rainc', 'rainnc'

         ui.convert_mpas(output=output_fname,
                         nlat=180, nlon=360,
                         startlat=-90.0, startlon=-180.0,
                         endlat=90.0, endlon=180.0,
                         var=var_list)

         Created the regridded nc file successfully: output/diag_latlon.nc
```

It specifies the output file path, the list of variables that needs to be converted and the invocation of convert_mpas() with input arguments. After execution, a message would be printed.

3.3.2 Wind animation

Read the code in the cell:

3.3.2 Wind animation

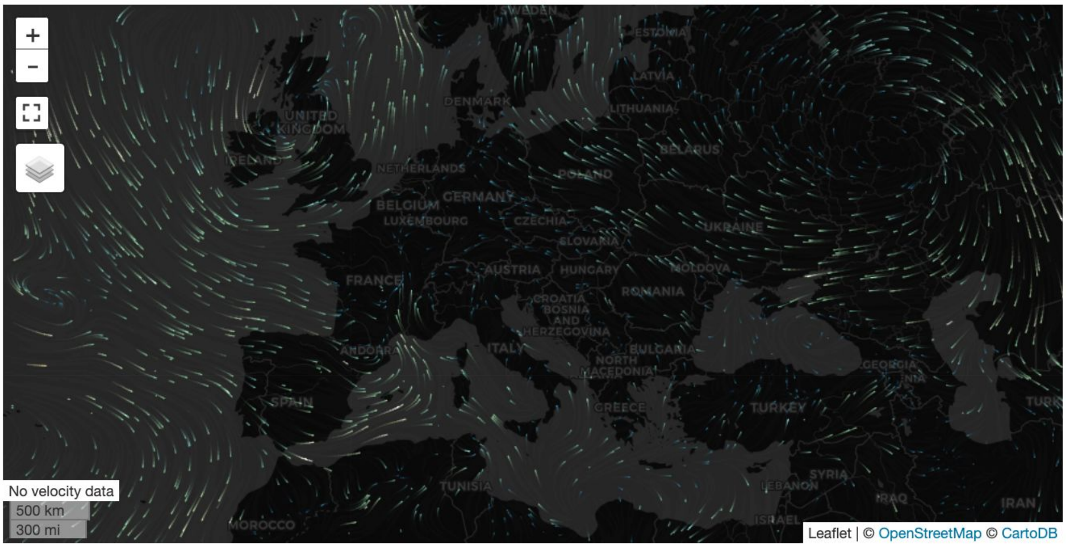
In [12]:
```
# This is a sample code block for wind animation
from cpas.wind import *

wind_field = WindMap(output_fname)
display(wind_field)
```

▾ Center and Zoom

Center Lati… 46.89985 Center Lo… 14.91497 Zoom level: 4

Select time step: 0

Select vertical level: surface

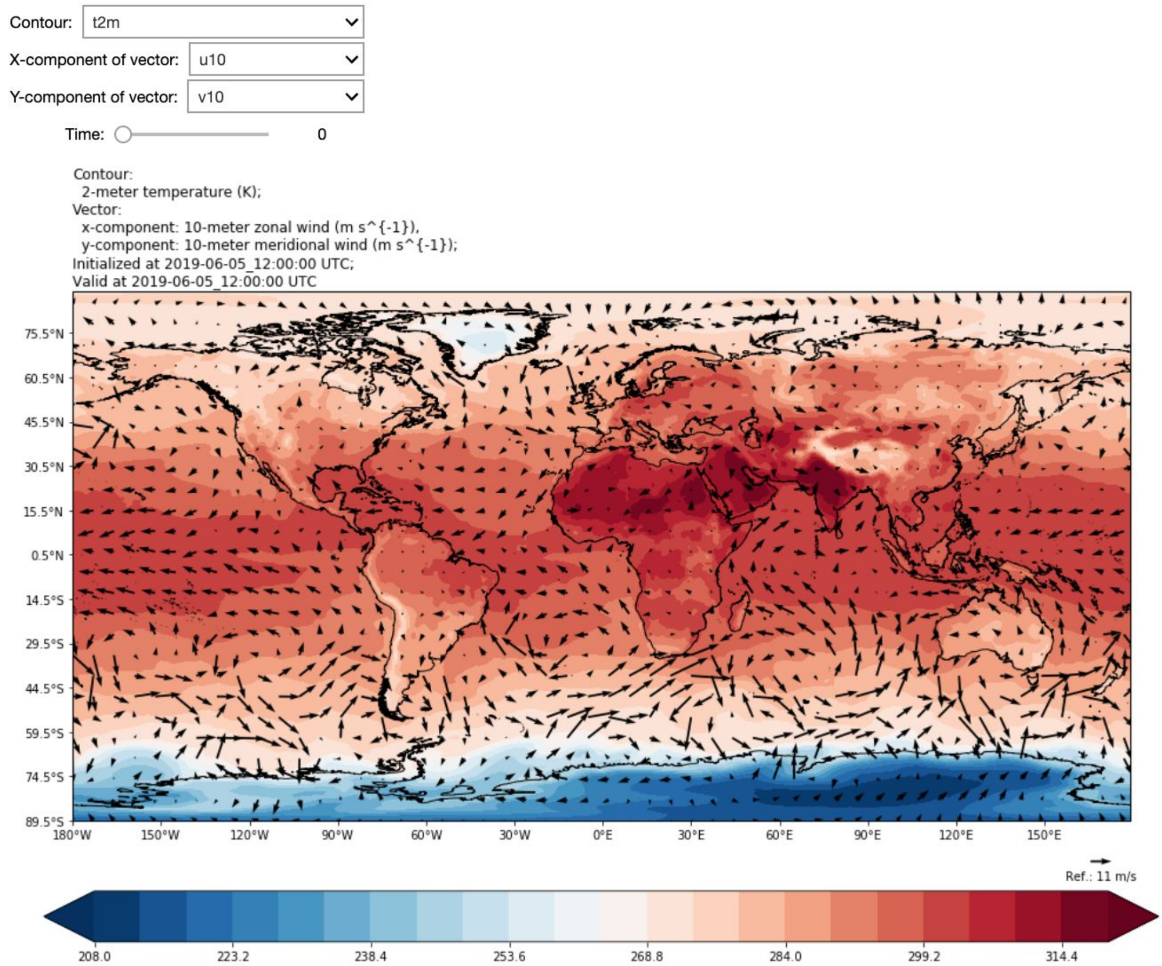


It assumes the wind data was converted into lat-lon grid data in the specified file path. Executing the code would produce the wind animation.

3.3.3 Plot the converted data

Have a look at the long piece of code and modify if necessary. Executing the code, some widgets and a plotting would appear. They should be self-explanatory to operate.

```
output = interactive_plot.children[-1]
interactive_plot
```

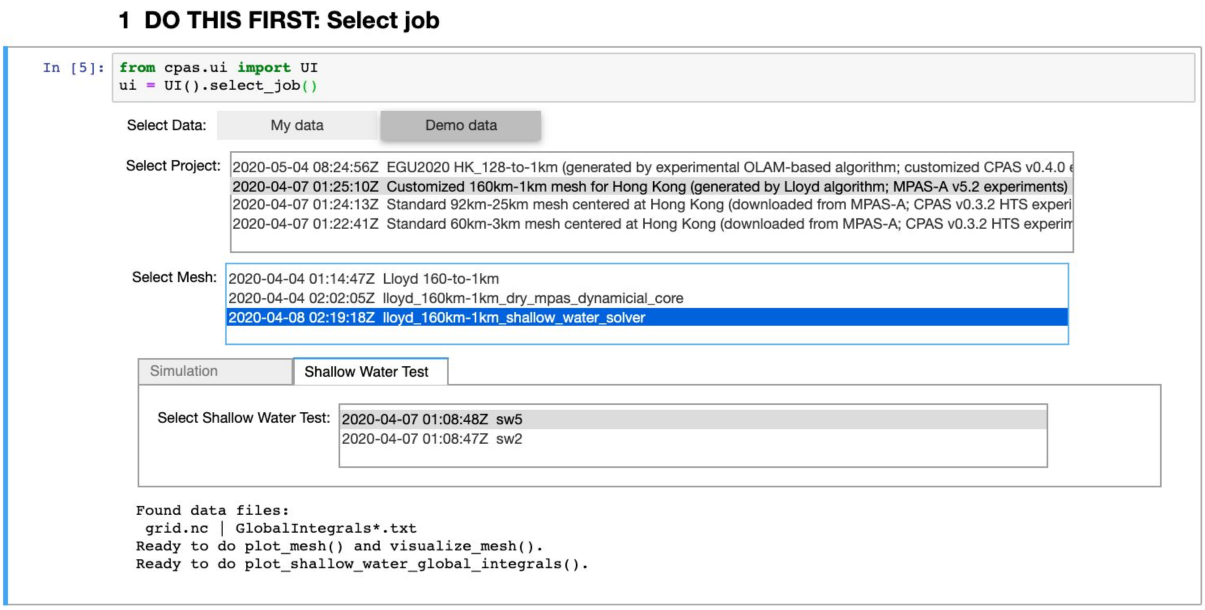


4. Shallow water test result analysis

To verify validity of Atmospheric models, shallow water tests are often used. Researchers often publish shallow water test results in academic journals or presentations. MPAS-A bundles shallow water tests in core_sw, which are also described in Ringler et al (2011) and Williamson et al (1992). The shallow water dynamic core, core_sw, can be configurated to output global integrals into text files.

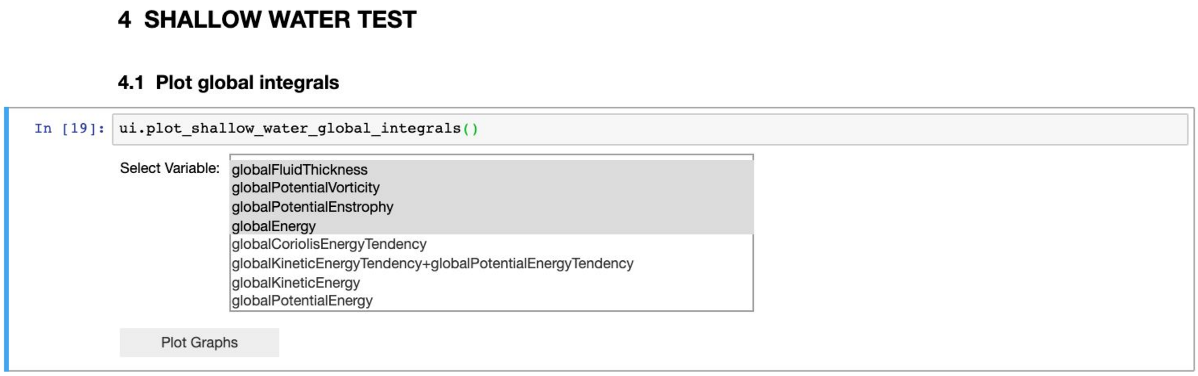
4.1 Plot global integrals

In the widgets shown by select_job(), select the “Shallow Water Test" tab.

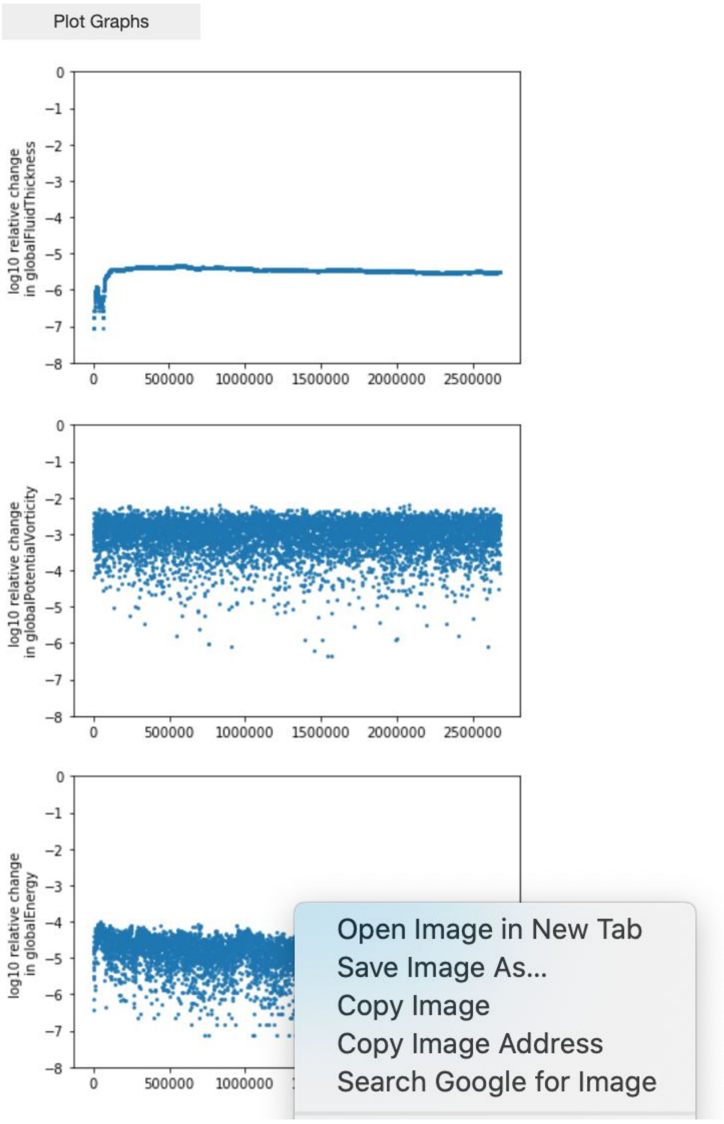


The Demo data shown above contains some Shallow Water Test results. Note that the message tells GlobalIntegrals*.txt are found.

Execute the code cell with plot_shallow_water_global_integrals().



Select the variables you are interested in (multiple items allowed). Click the “Plot Graphs” button. Graphs would appear after a while.

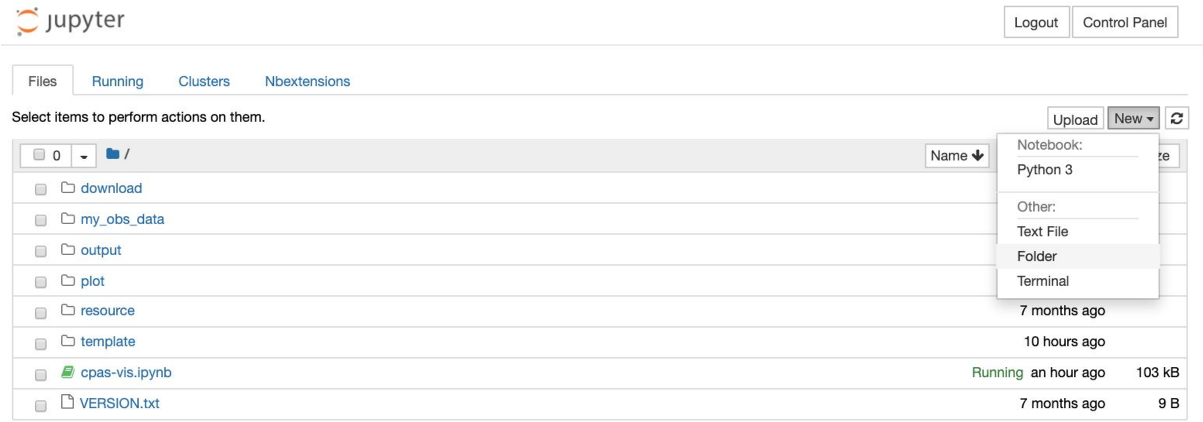


For your convenience, you may right click on a graph and use “Save Image As…” to save it to your local computer.

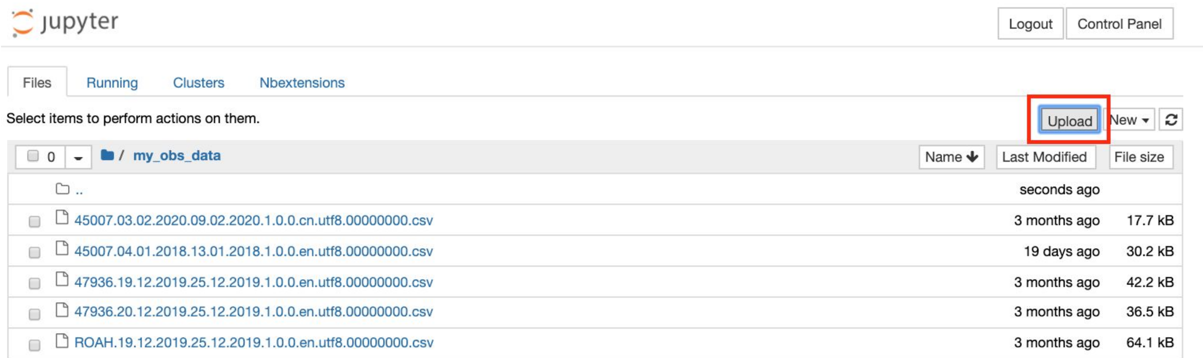
5. Model evaluation against your observation data

The cpas Python package provides the ModelEvaluation class for your convenience of evaluating model predictions against observation data.

Your observation data needs to be uploaded to the Jupyter system first.



Use “New”, “Folder” as shown above to create a folder on the Jupyter system. The “my_obs_data” has already been created in the above example. Use the “Upload” button to upload your observation data file.



Here is an example of of observation data in csv format:

```
# Weather station Hong Kong (airport), Hong Kong, WMO_ID=45007,selection from 04.01.2018 till 13.01.2018, all days
# Encoding: UTF-8
# The data is provided by the website "Reliable Prognosis", rp5.ru
# If you use the data, please indicate the name of the website.
# For meteorological parameters see the address http://rp5.ru/archive.php?wmo_id=45007&lang=en
#
"Local time in Hong Kong (airport)";"T";"Po";"P";"Pa";"U";"DD";"Ff";"ff10";"ff3";"N";"WW";"W1";"W2";"Tn";"Tx";"Cl";"Nh";"H";"Cm";"Ch";"VV";"Td";"RRR";"tR";"E";"Tg";"E'";"sss"
"13.01.2018 23:00";"14.2";"768.7";"769.3";"0.6";"66";"Wind blowing from the east";"5";"";"";"10%  or less, but not 0";"";"";"";"";"";"Stratocumulus other than Stratocumulus cumulogenitus.";"10%  or less, but not 0";"600-1000";"No Altocumulus, Altostratus or Nimbostratus.";"No Cirrus, Cirrocumulus or Cirrostratus.";"17.0";"7.8";"";"";"";"";"";"";
"13.01.2018 20:00";"13.8";"768.1";"768.7";"0.9";"67";"Wind blowing from the east";"5";"";"";"no clouds";"";"";"";"";"17.3";"";"";"";"";"";"16.0";"7.8";"No precipitation";"12";"";"";"";"";
```

Read the code in the code cell:

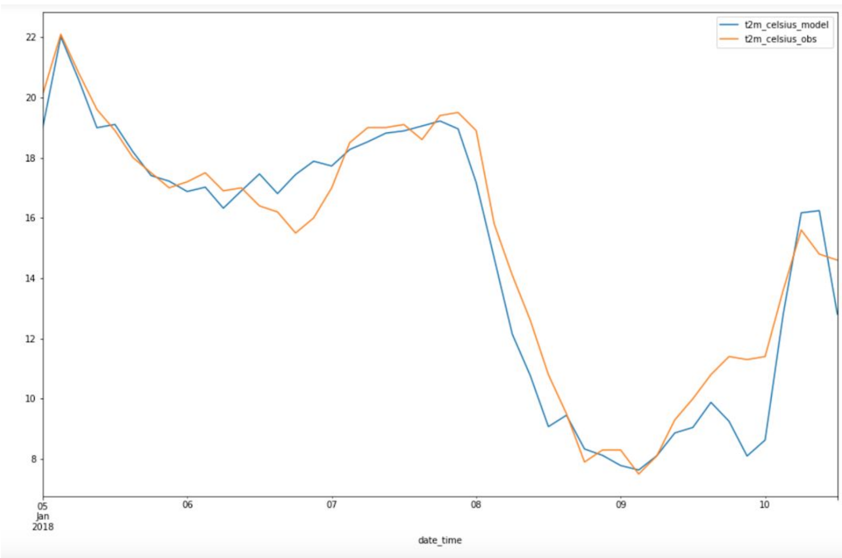
5.1 Model evaluation against your observation data (sample code)

Free free to change the code for your observation data upload. See this blog article.

```
In [ ]: import numpy as np
        import pandas as pd
        from datetime import timedelta
        from cpas.evaluation import ModelEvaluation, csv2df

        (my_lat, my_lon) = (22+18.534/60, 113+54.84/60)  # Hong Kong (airport)
        local_time_header = 'Local time in Hong Kong (airport)'  # Column header in CSV file
        local_time_offset = 8
        my_csv = 'my_obs_data/45007.04.01.2018.13.01.2018.1.0.0.en.utf8.00000000.csv'

        raw_obs_df = csv2df(my_csv, sep=';', comment='#', index_col=False)
        obs_df = pd.DataFrame()
        obs_df['date_time'] = pd.to_datetime(raw_obs_df[local_time_header], format='%d.%m.%Y %H:%M') + timedelta(hours=-local_t
        (obs_df['lat'], obs_df['lon']) = (my_lat, my_lon)
        obs_df['t2m_celsius'] = raw_obs_df['T']

        me = ModelEvaluation(ui.mesh_ncfile, ui.diag_ncfile, obs_df)
        me.met_combine()
        comp = me.get_comparison()

        mask = ~np.isnan(comp['t2m_celsius_model'])
        filtered = comp[mask]
        filtered.xs(my_lat, level='lat').xs(my_lon, level='lon').plot(y=['t2m_celsius_model', 't2m_celsius_obs'], figsize=(16,
```

You need to load your observation data into a pandas DataFrame. Beware of conversion between local time and UTC, as UTC is used in ModelEvaluation for matching time. If you reuse this piece of code, please remember to change the variables for specifying the lat, lon, header label and file path for your case.

Executing this piece of code, something like the following plotting would appear:



6. Download mesh

Run the cell of code with download_mesh().

6 MISC SERVICE

6.1 Download mesh

```
In [2]: ui.download_mesh()
```

Out[2]: By downloading the mesh, I agree to the No Unauthorized Redistribution of Generated Mesh Files clause of the Terms and Conditions.
download/mesh.tar.gz

Read the Terms and Conditions. If you agree with it, you may click the link to download the file.

7 API documentation

You may use the question mark (?) built-in magic of Jupyter's IPython kernel to check API documentation.

7 API Signature and Docstring

Put cursor after the dot, hit `⇥ Tab`, select method, hit `^ Ctrl` + `↵ Enter`

```
In [1]: from cpas.ui import UI
        ?UI.visualize_mesh
```

```
Signature: UI.visualize_mesh(self, height='500px', width='100%', stroke_opacity=0.5)
Docstring:
Use an interactive map and widgets to visualize the mesh

Parameters
----------
height : str
    The height of the interactive map to display.

width : str
    The width of the interactive map to display.

stroke_opacity : float
    Opacity for drawing strokes of cell edges

Returns
-------
none
```