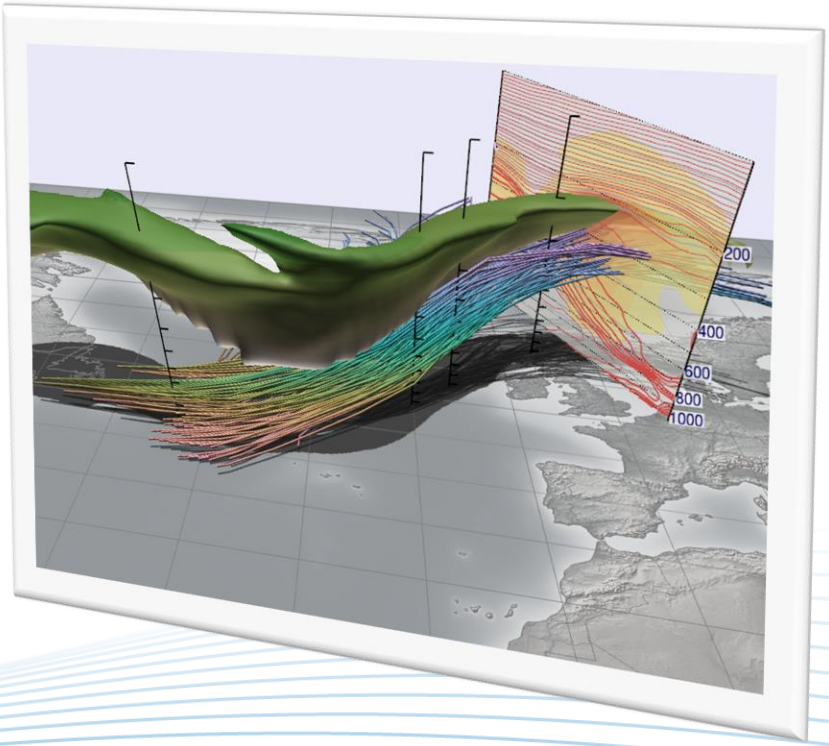
Met.3D
introductory tutorial

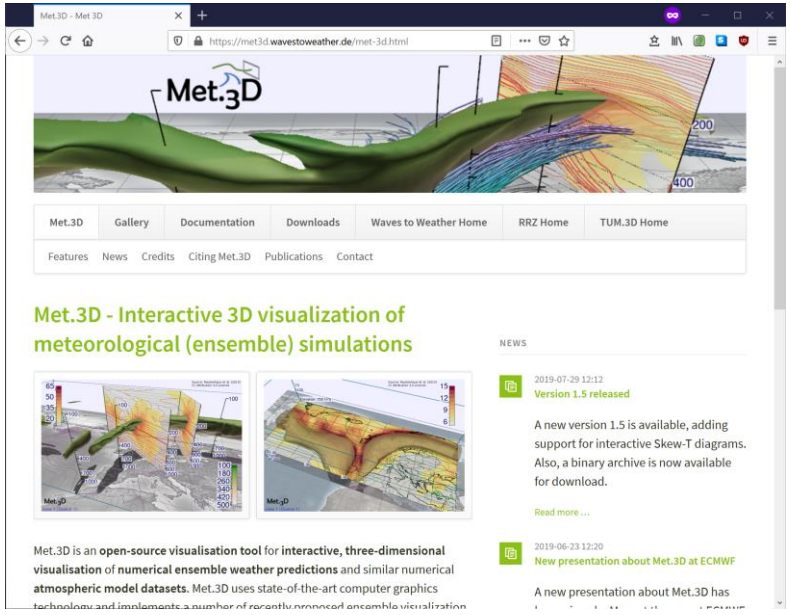
Marc Rautenhaus

Universität Hamburg, Regional Computing Center Visual Data Analysis Group,
Centre for Earth System Research and Sustainability (CEN)



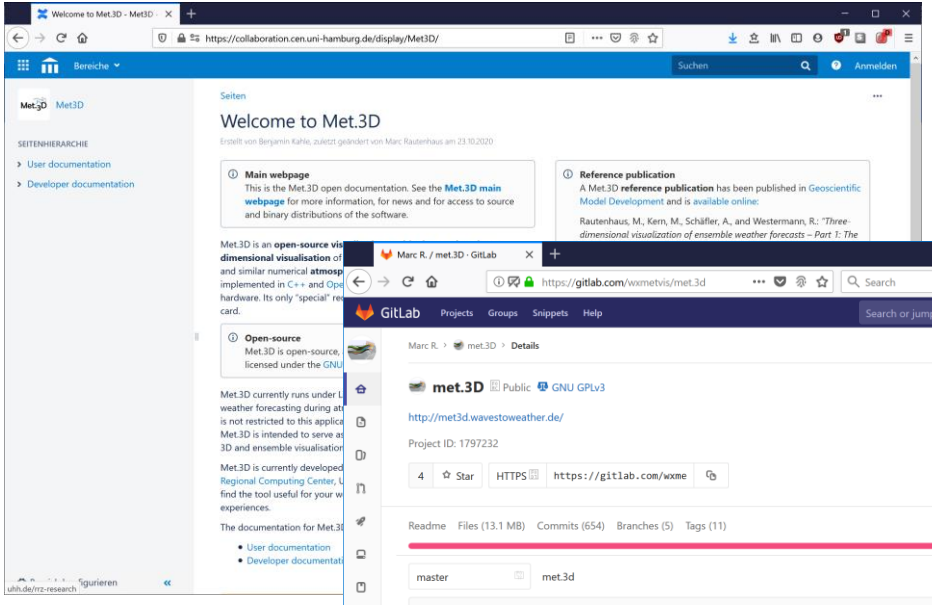
Version 2021.03.20

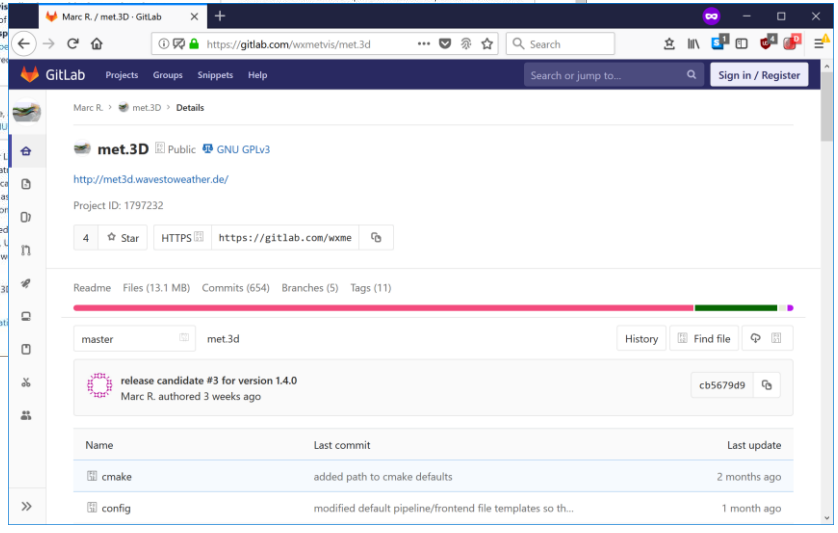
Before you start: Met.3D online resources



Website:
met3d.wavestoweather.de

Documentation:
https://collaboration.cen.uni-hamburg.de/display/Met3D/



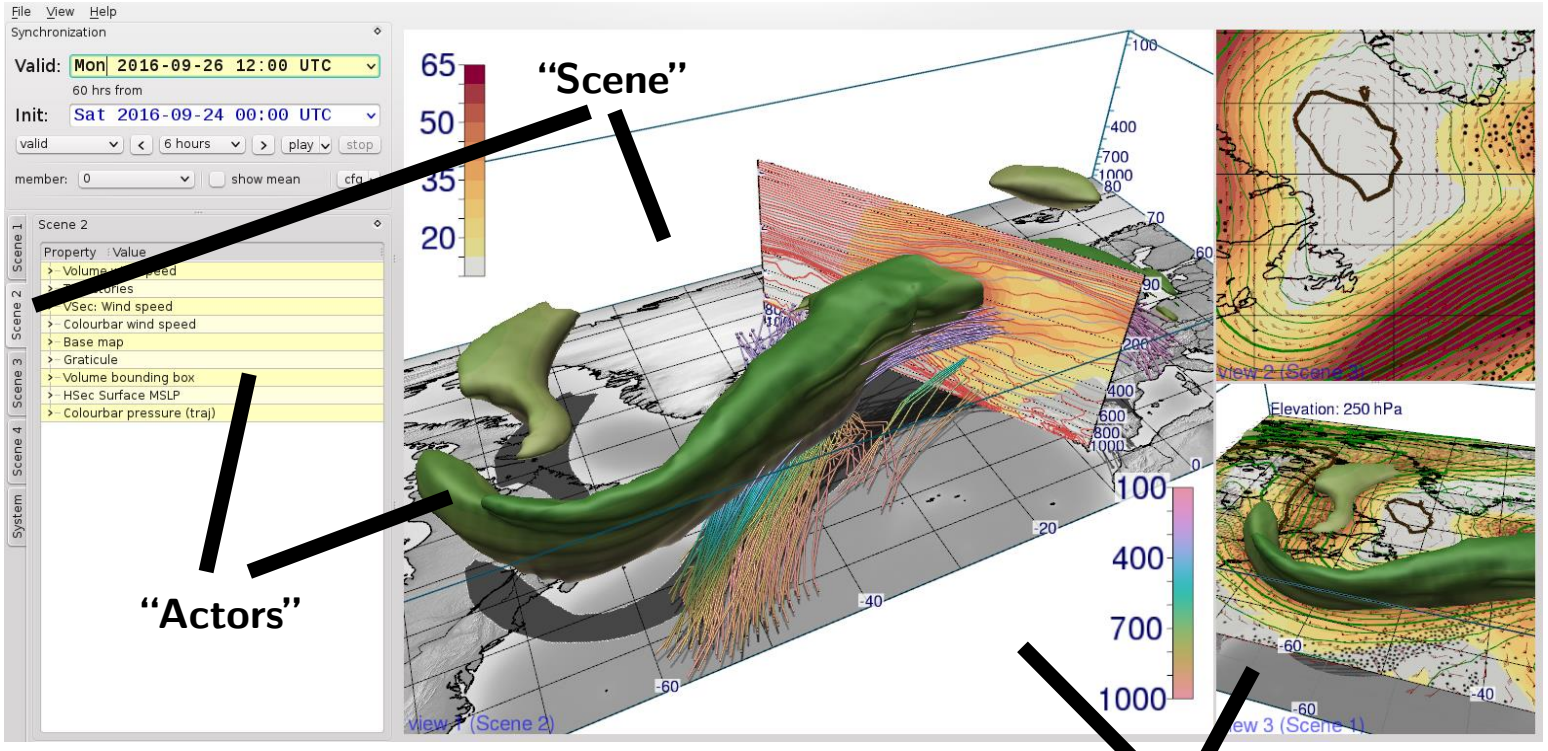


Repository:
gitlab.com/wxmetvis/met.3d

Basics: Met.3D frontend elements

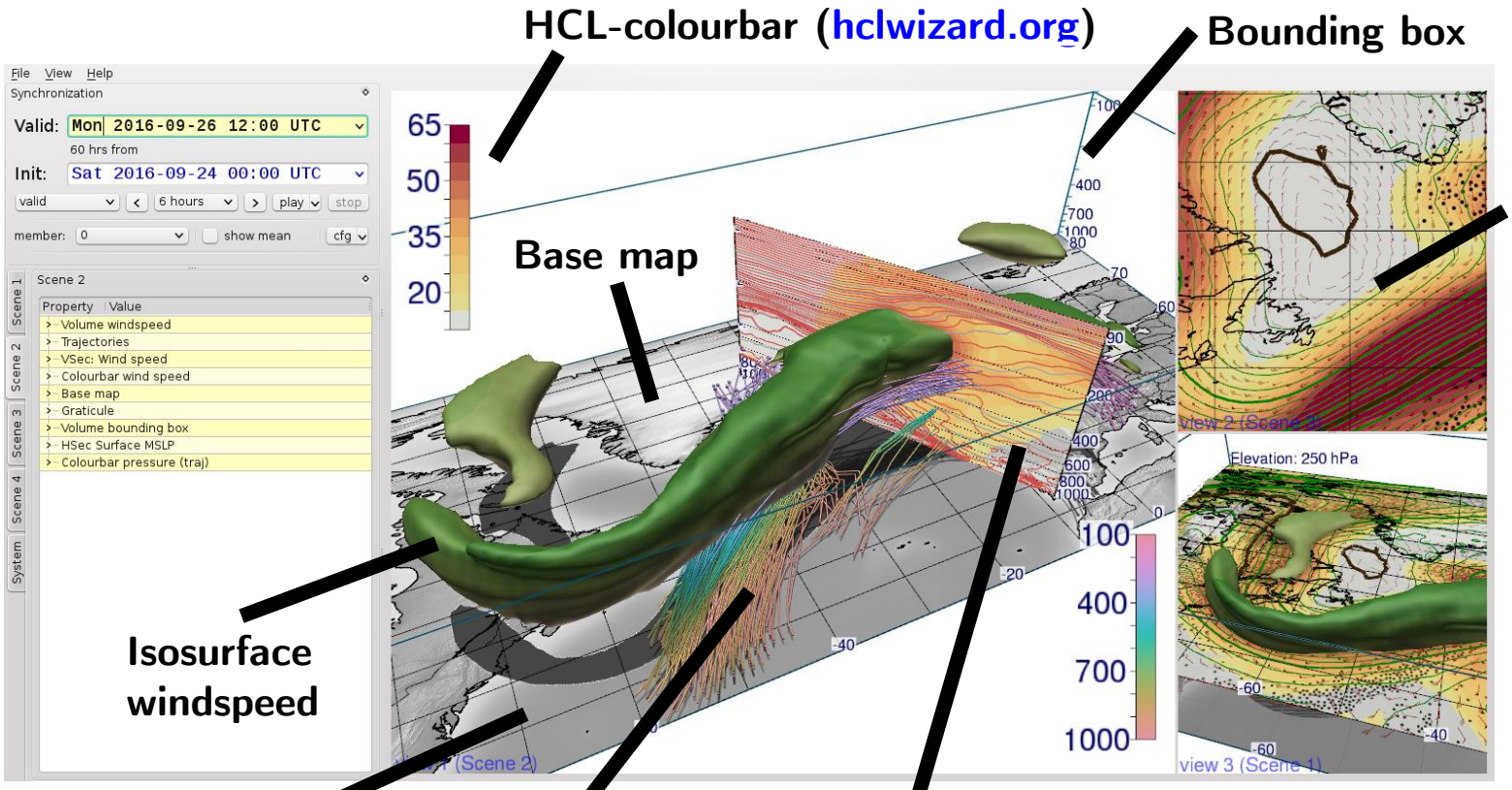
Time and ensemble navigation

Parameters for actors and views



Different 3D projection options

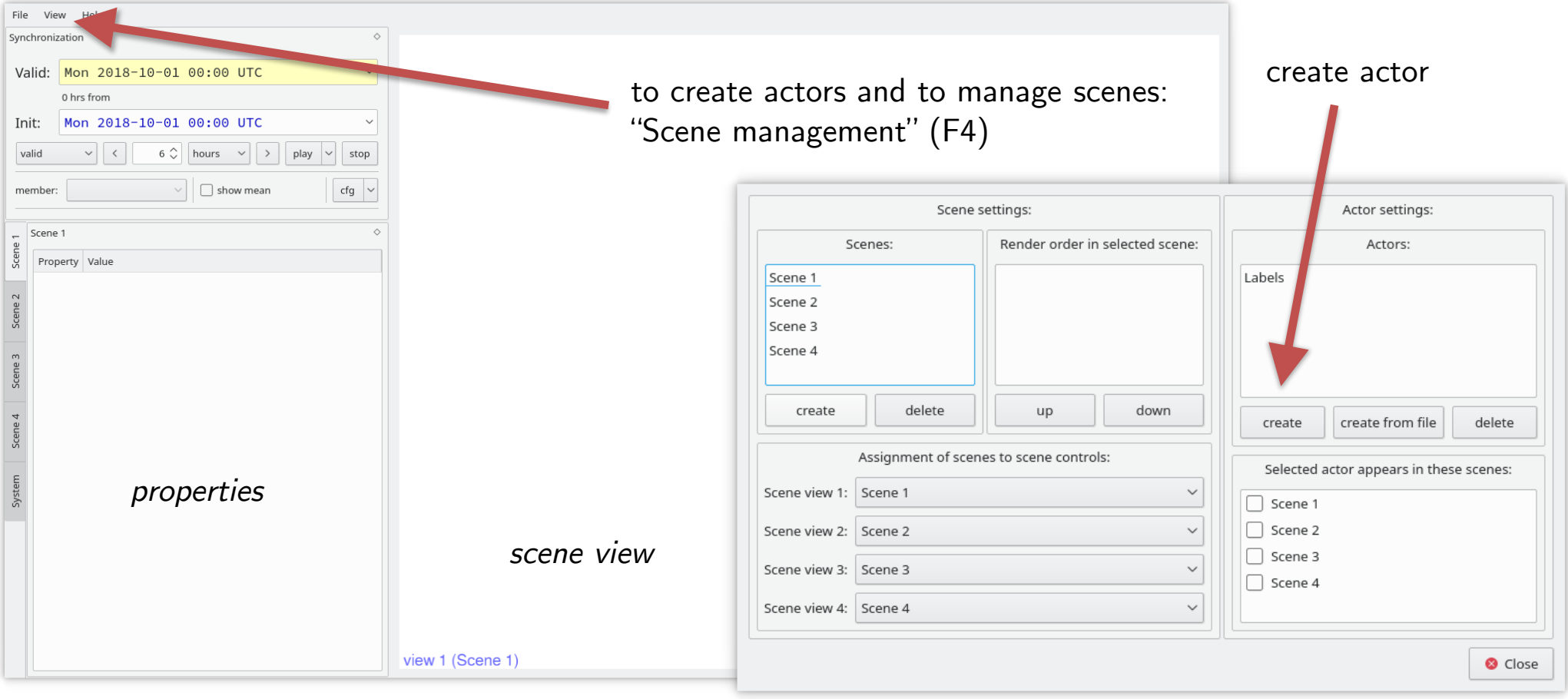
Basics: Met.3D frontend elements



Tutorial content: you will learn...

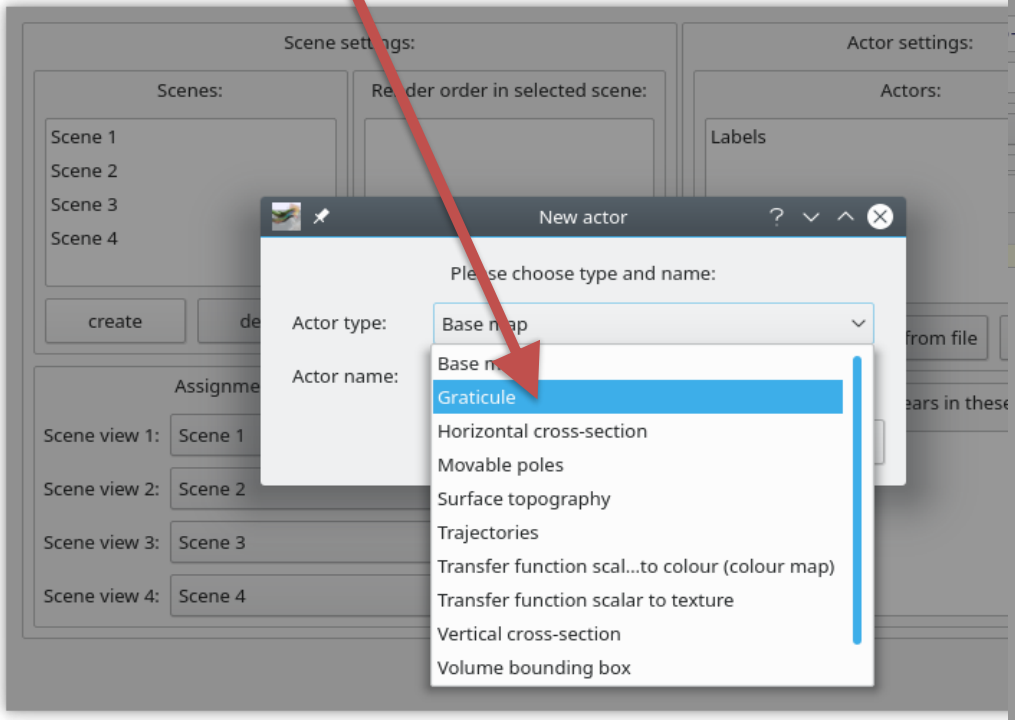
1. Actors 1: create graticule, basemap etc.
2. Saving individual actors, session management
3. Loading datasets
4. Actors 2: horizontal section with data
5. Basic time and ensemble navigation, adjusting the colour bar
6. Actors 3: vertical section
7. System settings: multiple views, vertical scaling, camera interaction, annotations
8. Actors 4: 3D isosurface and vertical poles
9. Transfer functions
10. Basic ensemble statistics
11. Saving images and creating animations
12. Direct volume rendering
13. Data format: typical NetCDF layout
14. Working with Met.3D...

Create actors

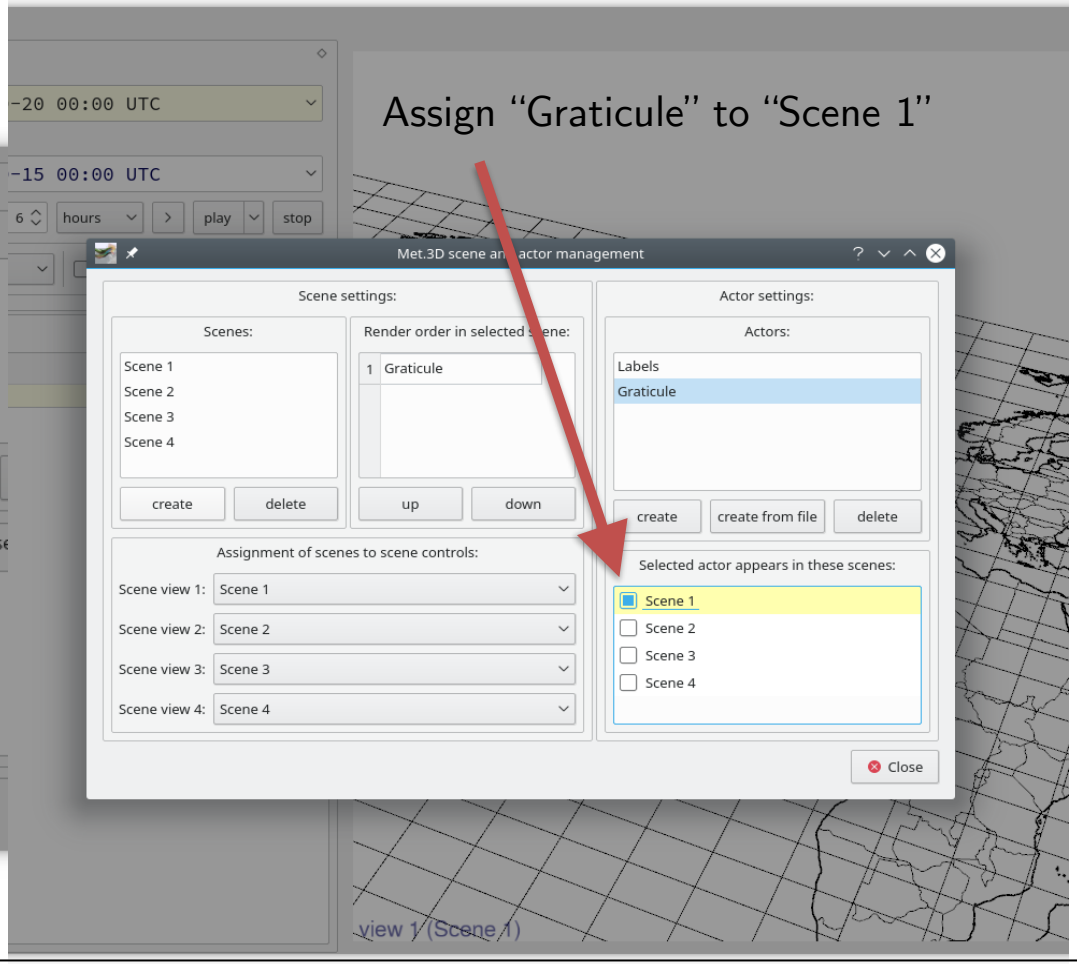


Create an actor: Graticule

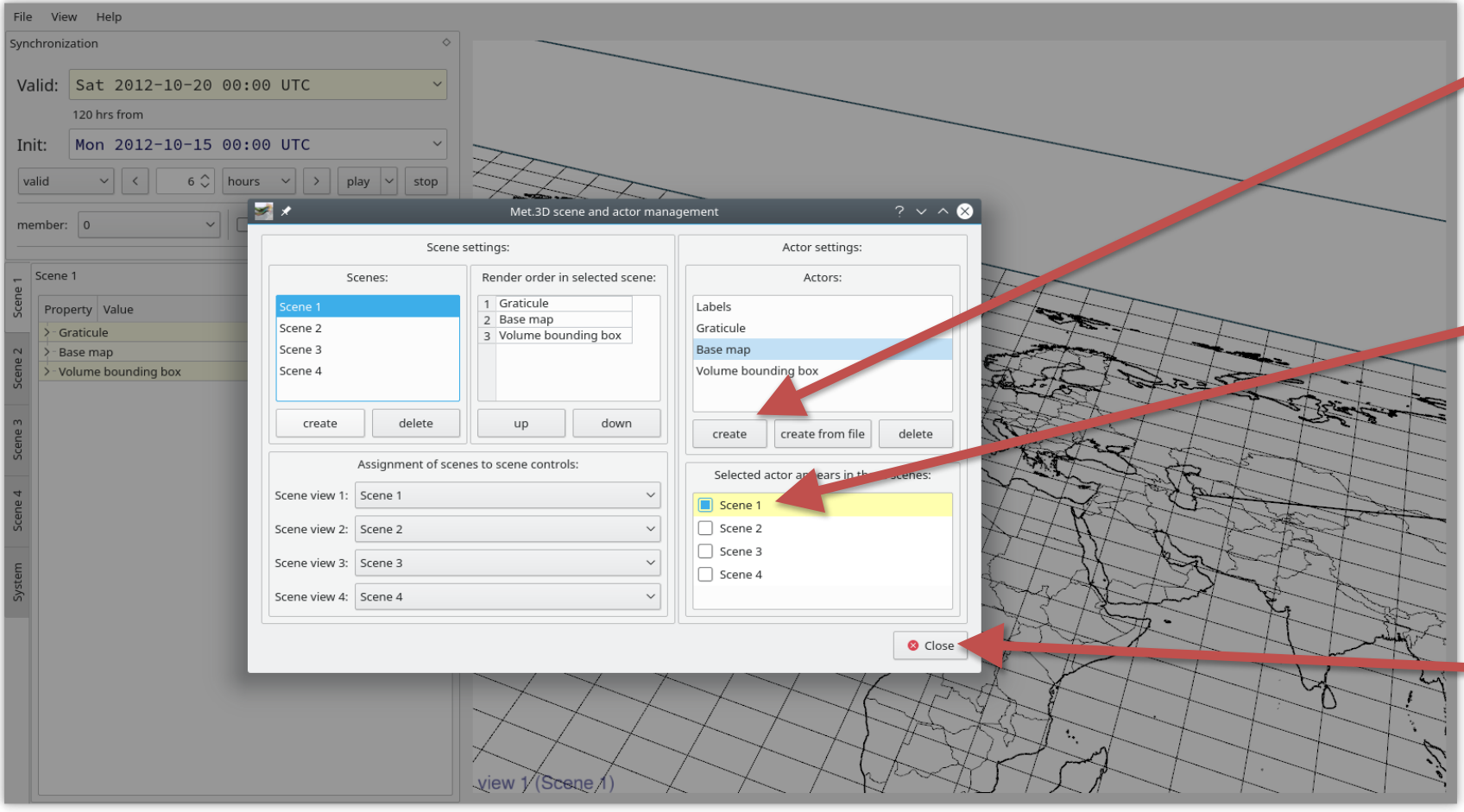
Choose “Graticule”, confirm with “ok”



Assign “Graticule” to “Scene 1”



Create actors: add “basemap” and “volume bounding box”

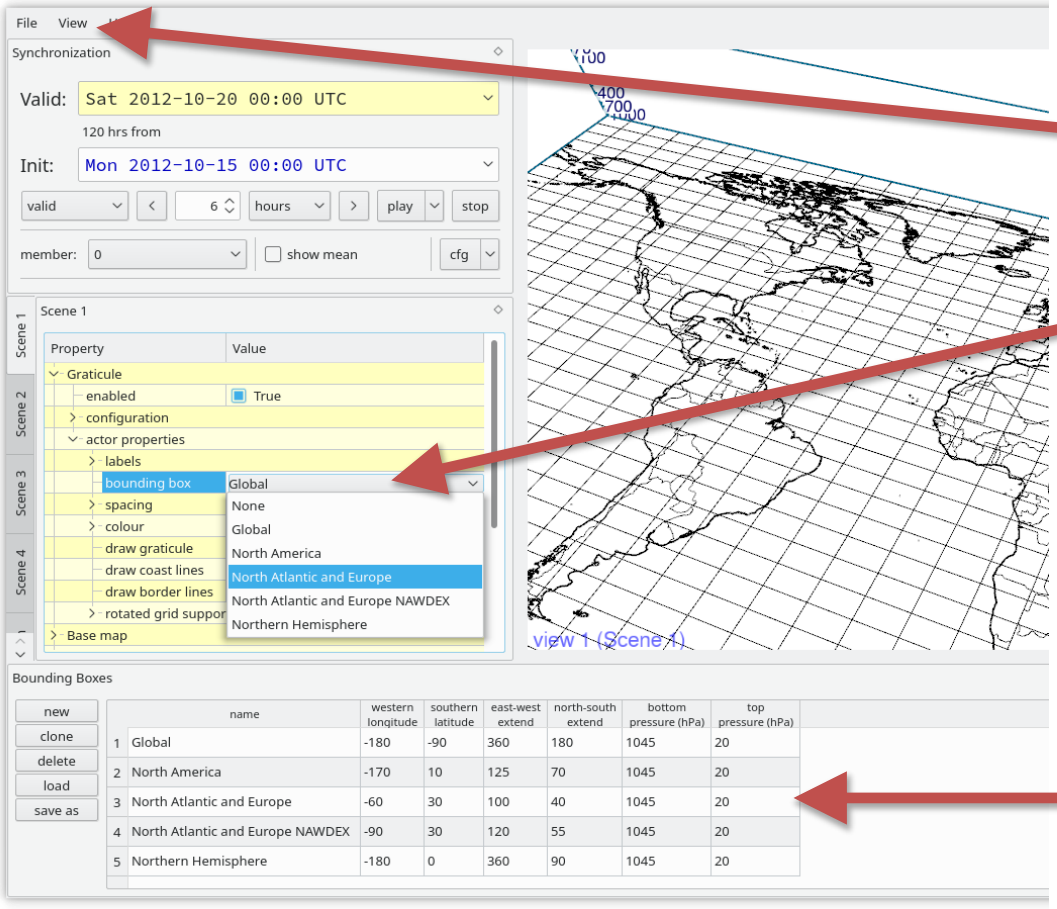


create new actors

assign actors to scene

close dialog

Adjusting the visible region (bounding box)



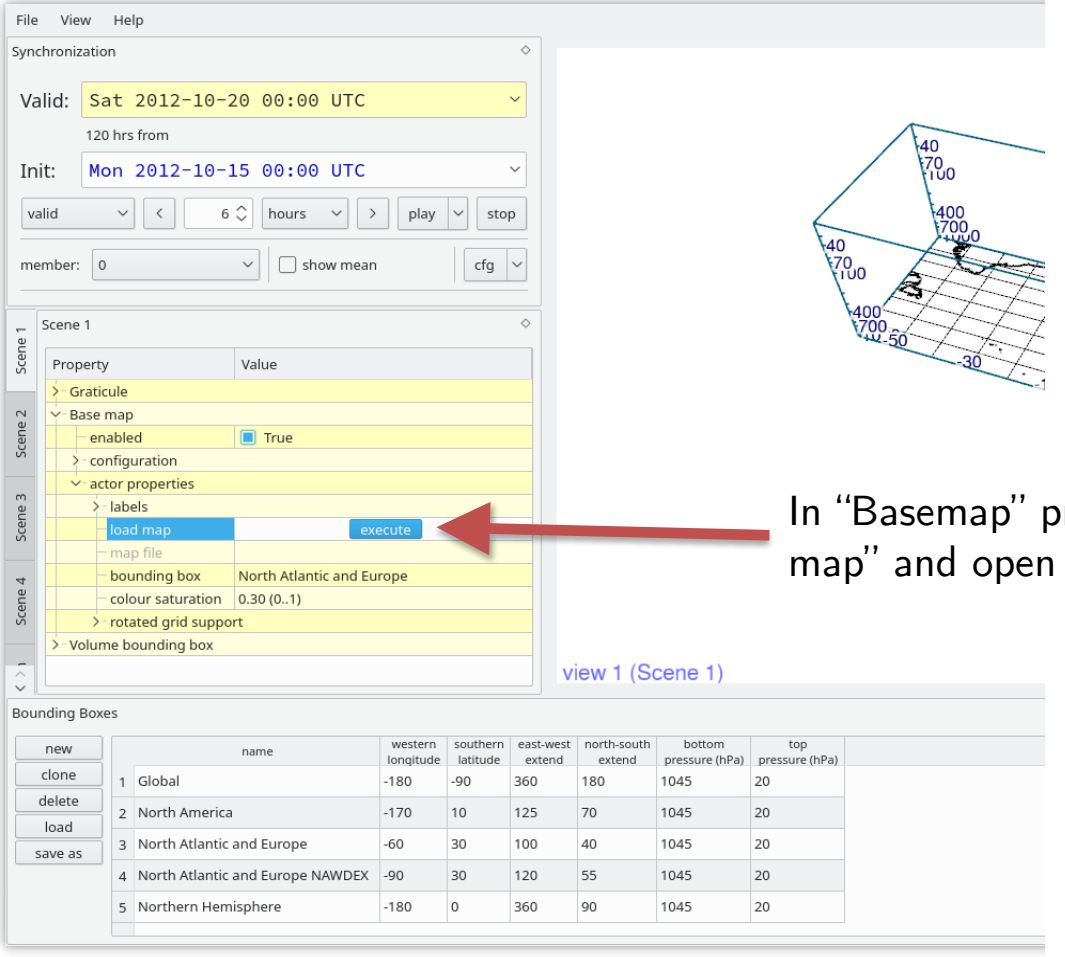
1) View -> Show Bounding Boxes (or F3) shows list of bounding boxes, parameters can be adjusted

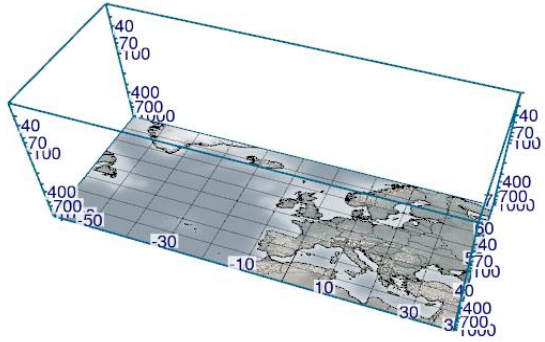
2) For each actor, the bounding box that should be used can be chosen

   Here: Select “North Atlantic and Europe”

3) Parameters could be set here; F3 closes the view

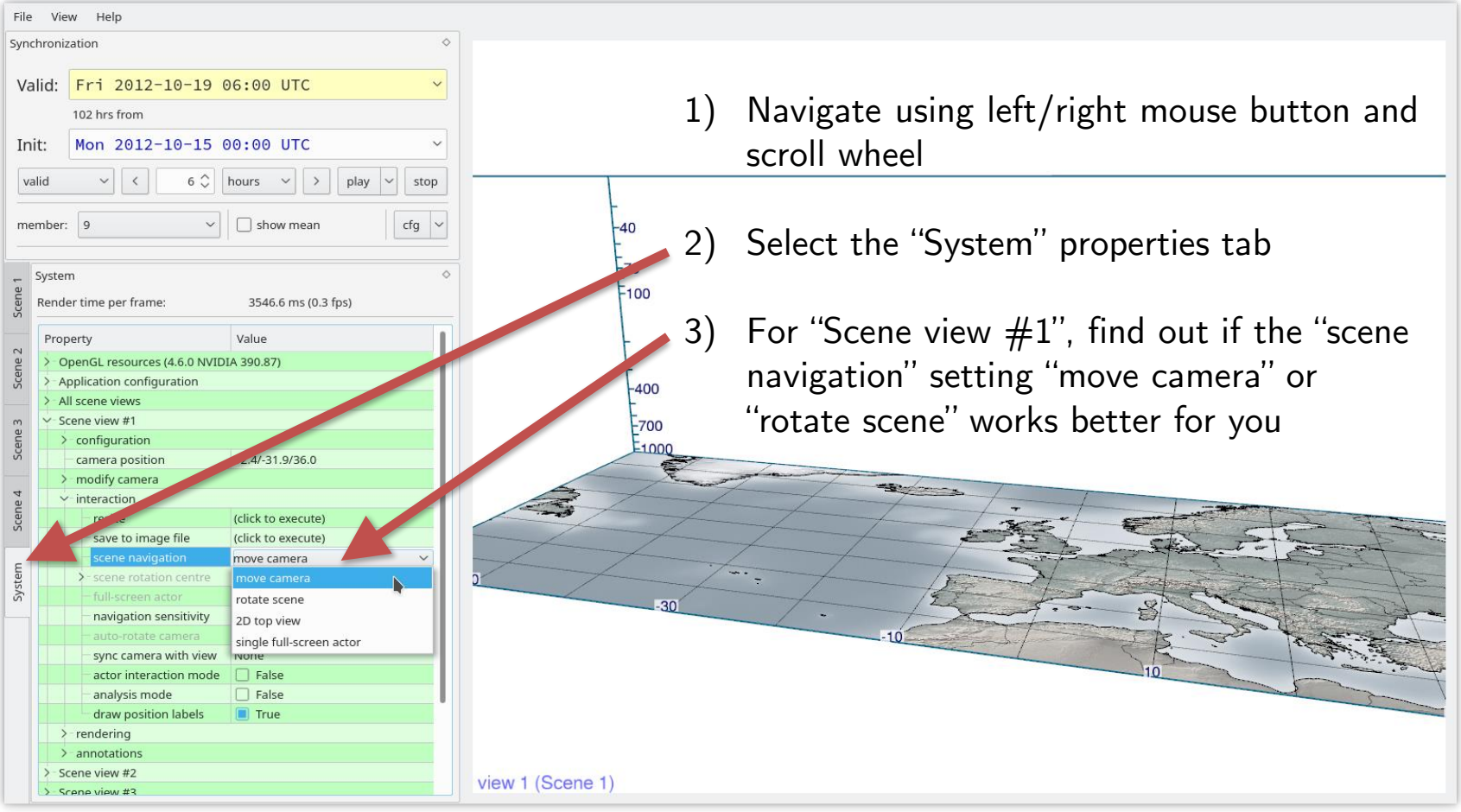
Basemap: load map file



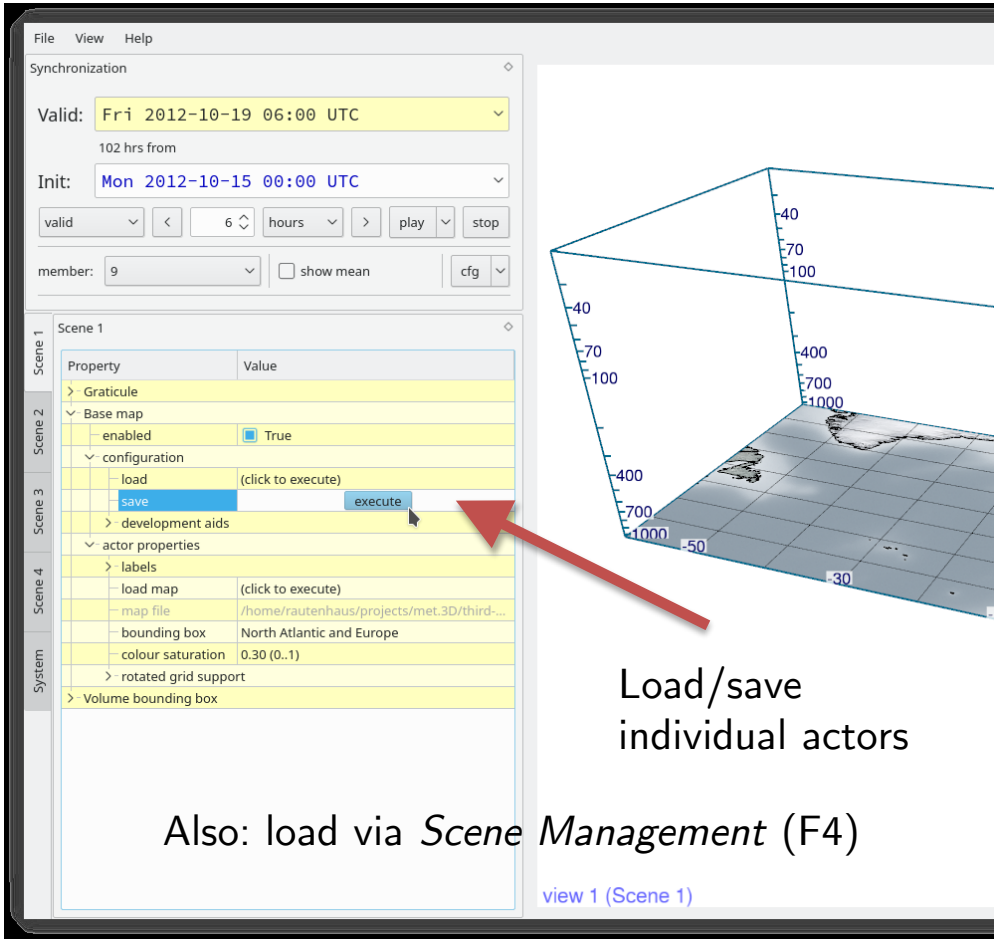


In “Basemap” properties, select “load map” and open a GeoTIFF file

Get accustomed to 3D navigation

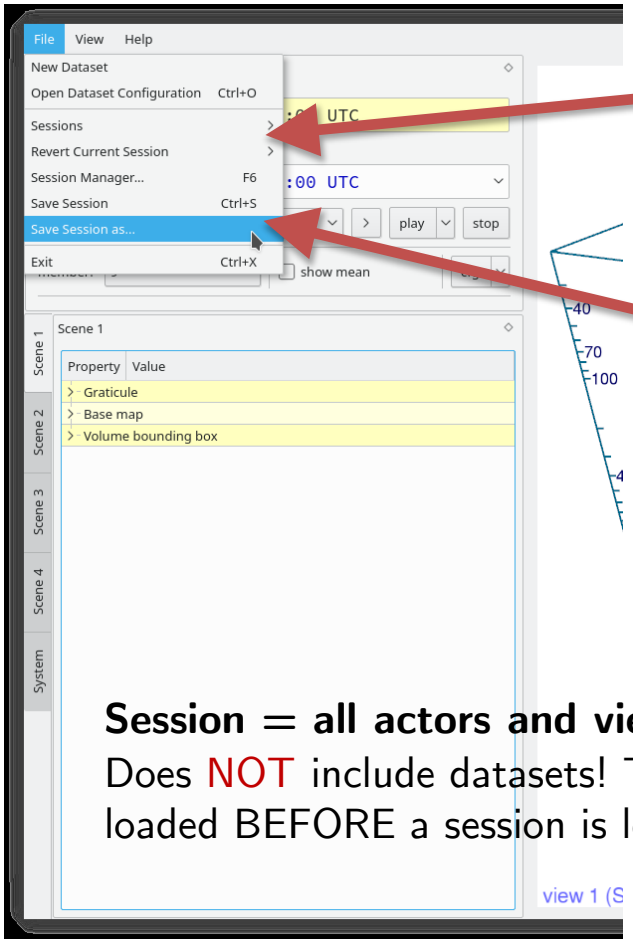


Saving individual actors and entire visualization sessions



Load/save individual actors

Also: load via *Scene Management* (F4)



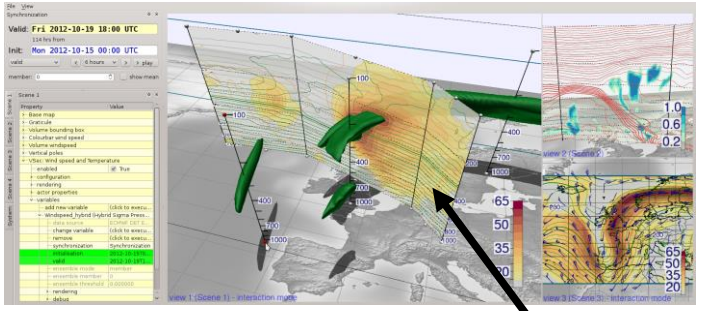
Load/revert/ switch session

Save current session

"**Ctrl-S**" (see terminal output)

Session = all actors and viewport settings.
Does NOT include datasets! These need to be loaded BEFORE a session is loaded.

How to get data into Met.3D: basic architecture



"ensemble-aware" pipeline

forecast

dataset

data processing pipeline

async. request

data item

actor ("forecast product")

CPU

shader program

GPU

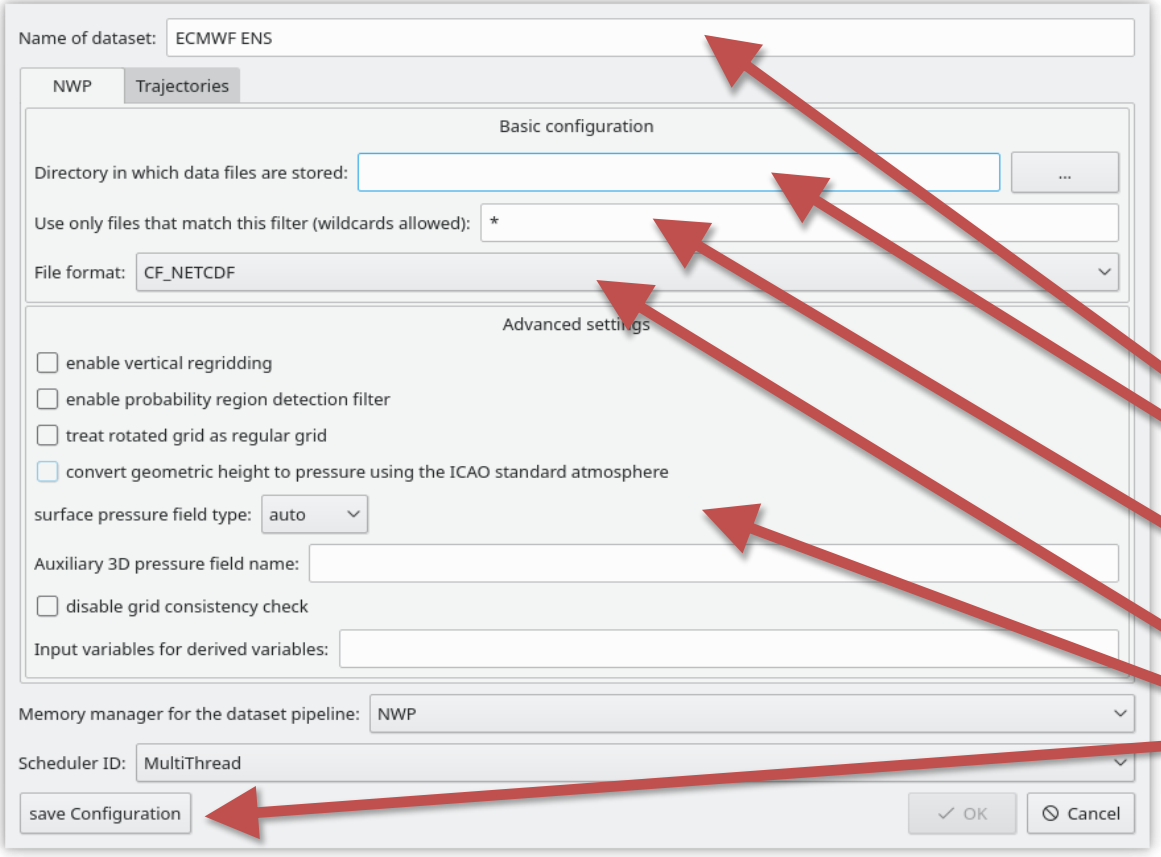
modular

caching

multithreading

GPU based visualization

Load a dataset

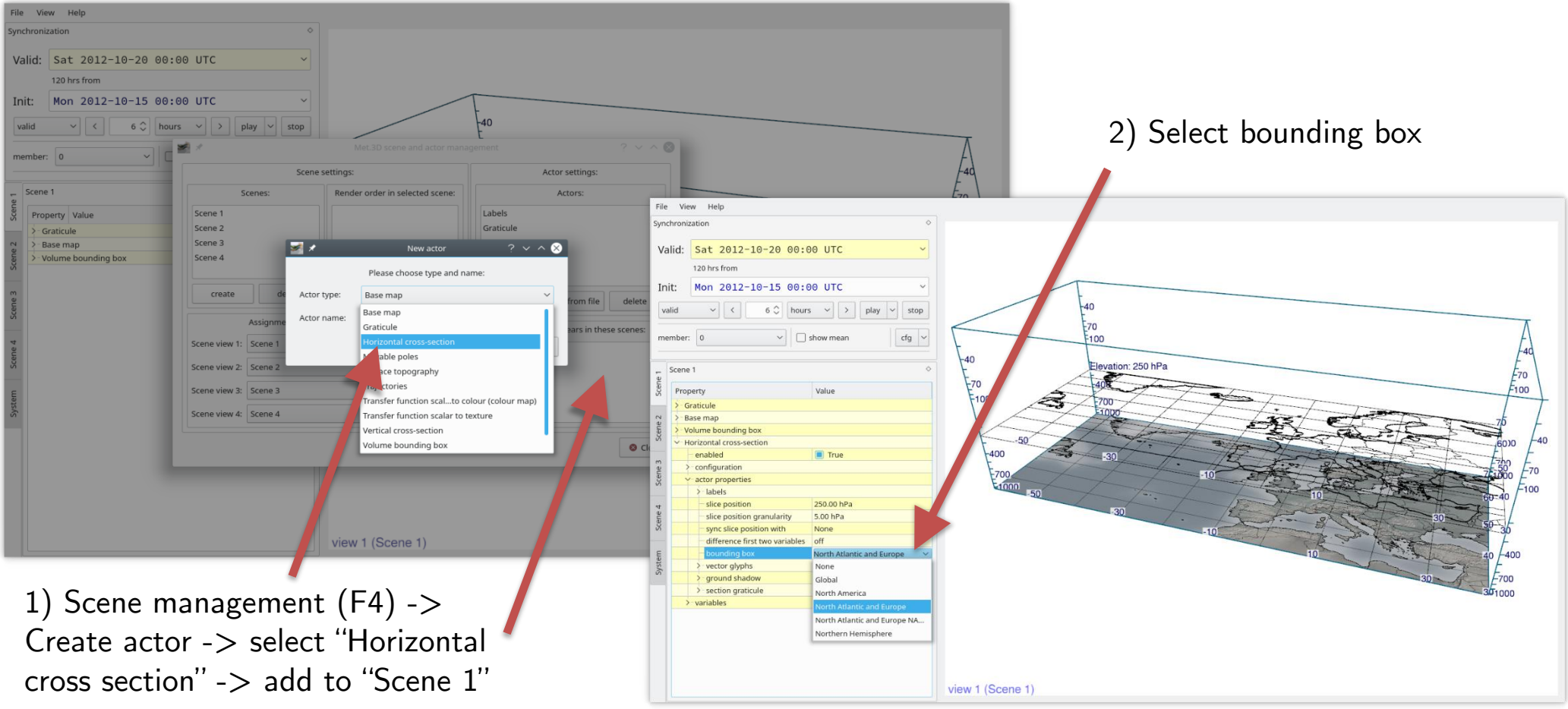


A "*dataset*" describes a single simulation or an ensemble of simulations. All variables and timesteps within a dataset need to be defined on the same grid. Data can be in a single file or distributed over multiple files.

1) File -> New Dataset
2) Specify name of dataset
3) Choose directory in which file(s) are stored
4) Specify a "filter" name that matches all the files that belong to the dataset
5) Select data format
6) Additional parameters
7) Save configuration to reload later

!! **Check terminal output for messages!**

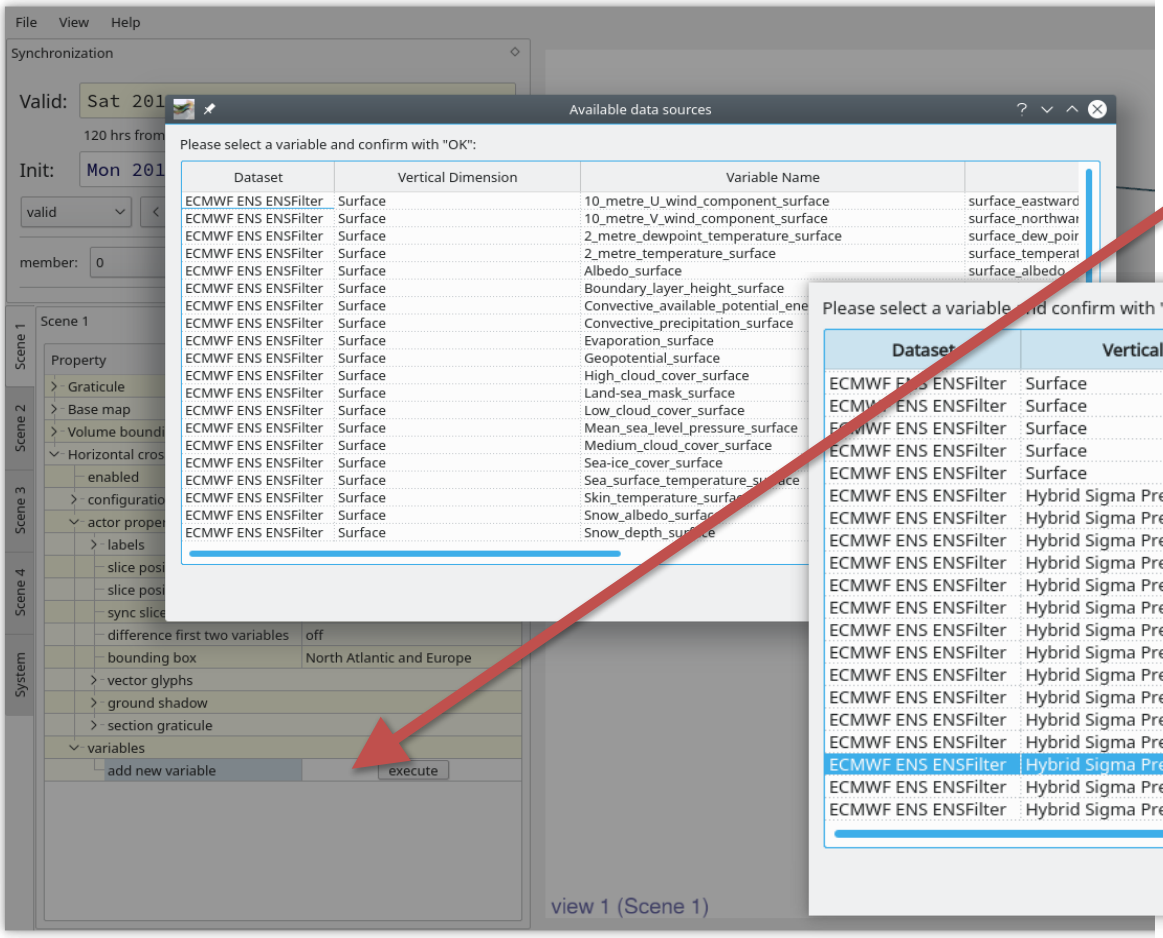
Create a horizontal section (map) with wind speed data



2) Select bounding box

1) Scene management (F4) -> Create actor -> select “Horizontal cross section” -> add to “Scene 1”

Create a horizontal section (map) with wind speed data



A horizontal section can display multiple "variables" from a dataset

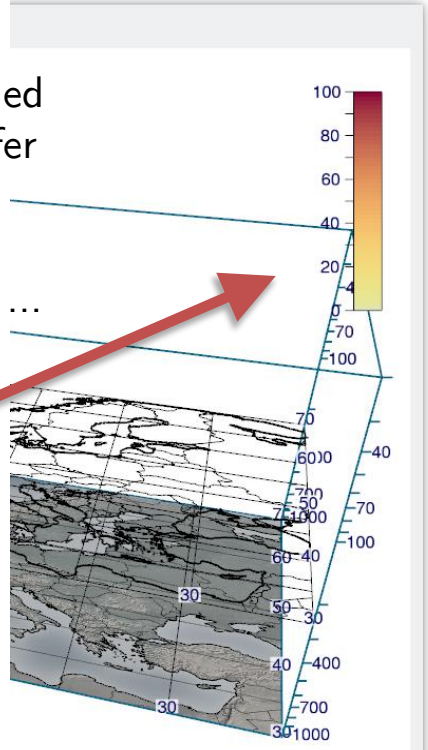
3) Select "add new variable"
4) Choose "wind speed"

Create a horizontal section (map) with wind speed data

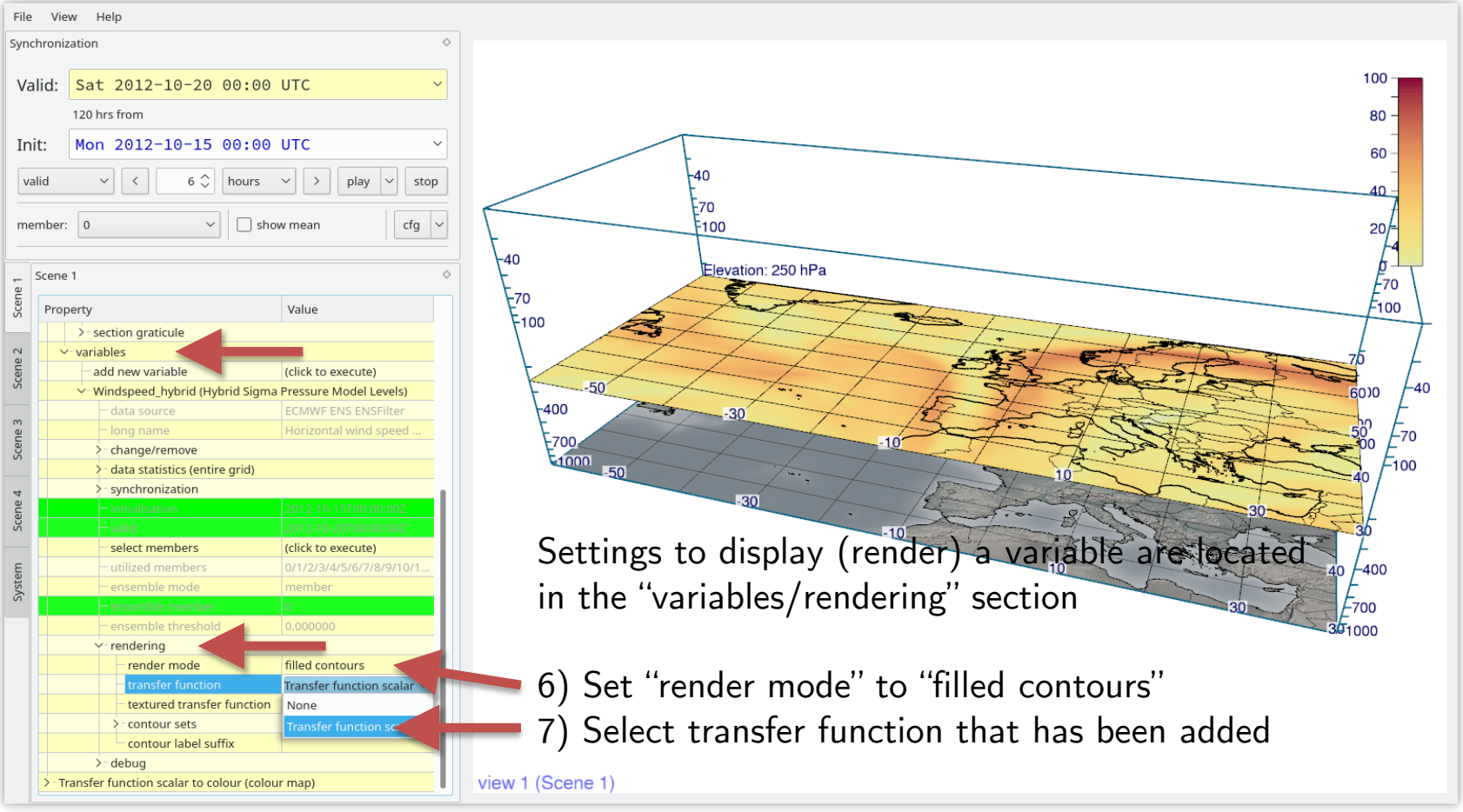
5) To show data as filled contours, add a “transfer function/colour map”:

F4 -> create actor -> …

This is also an actor.



Create a horizontal section (map) with wind speed data

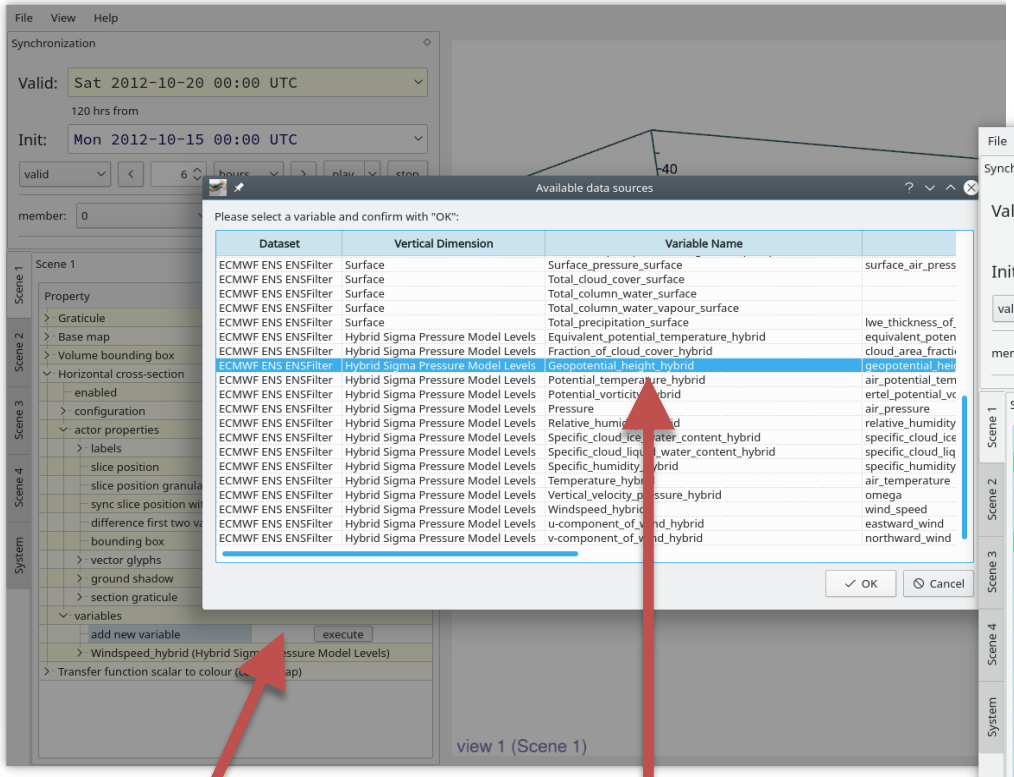


Settings to display (render) a variable are located in the "variables/rendering" section

6) Set "render mode" to "filled contours"

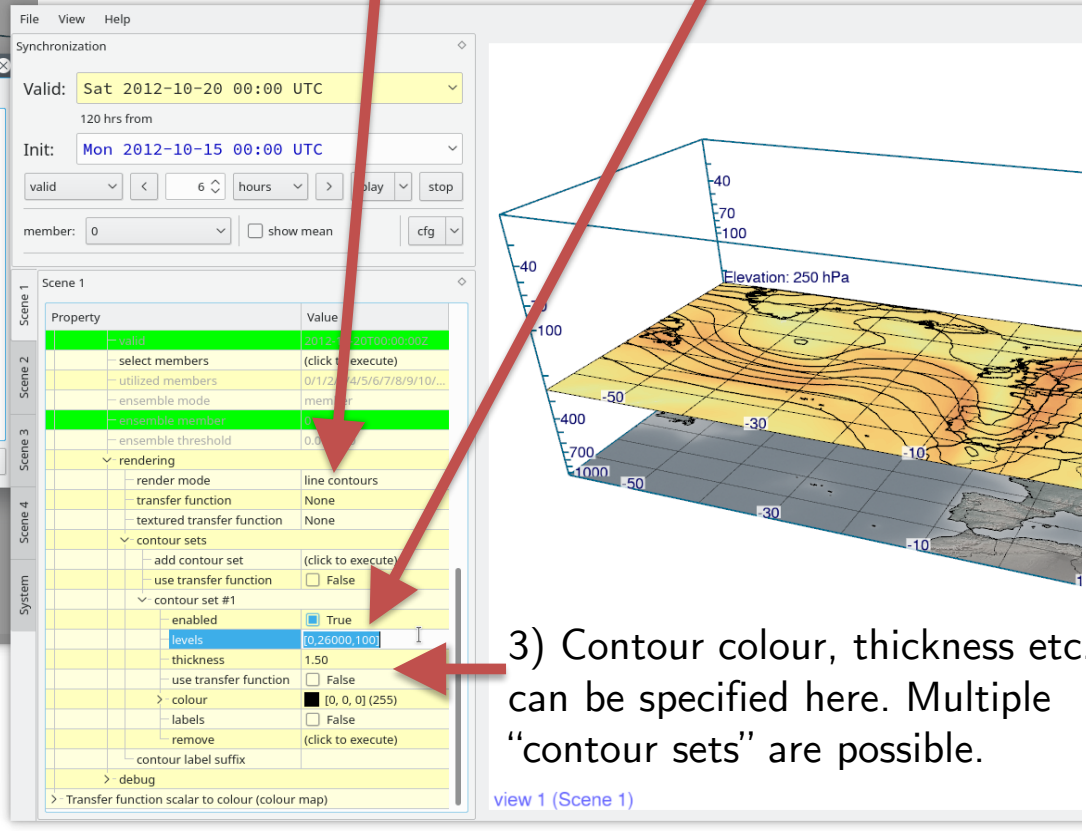
7) Select transfer function that has been added

Add a second variable: Geopotential height



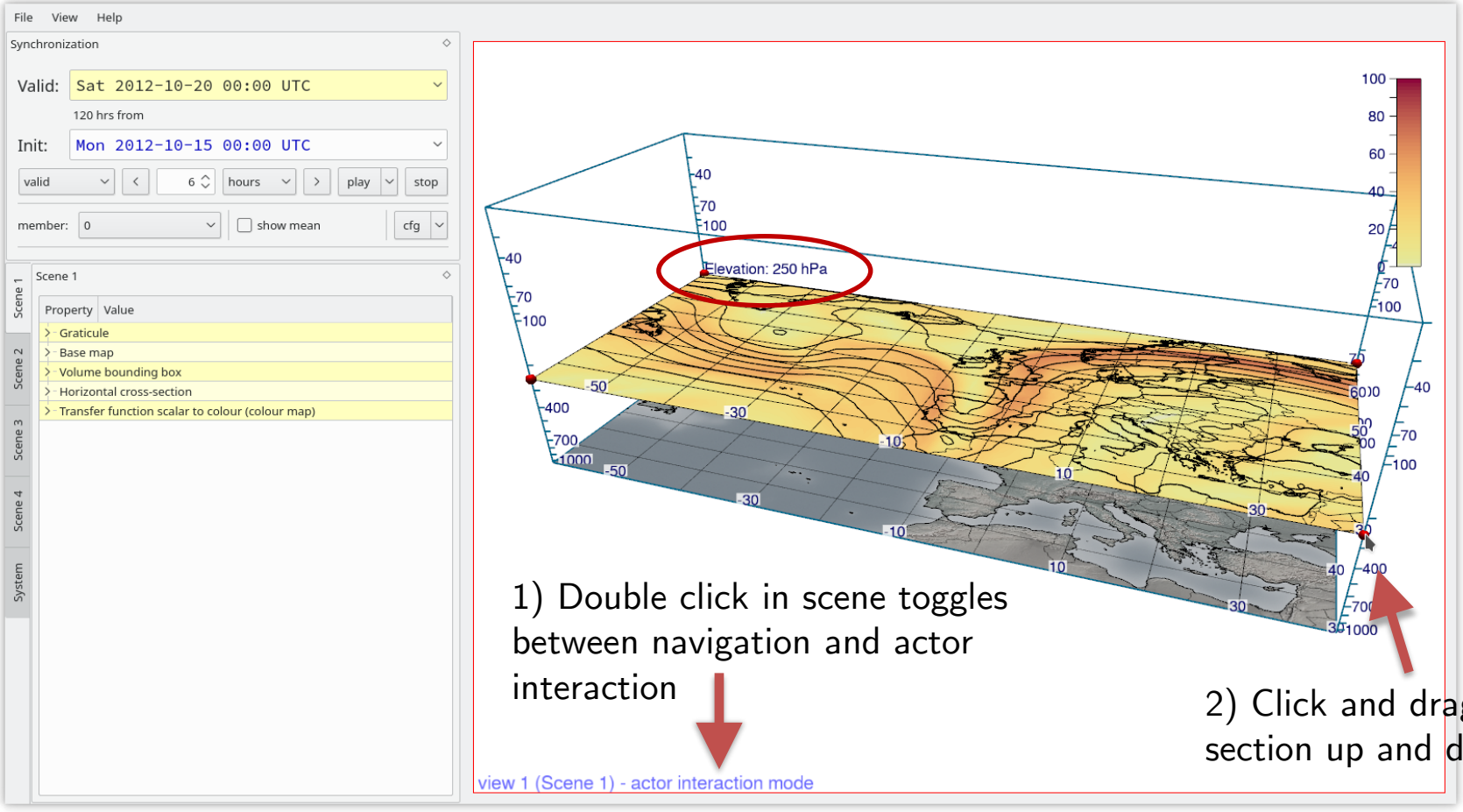
1) Add new variable -> select "Geopotential height" on model levels

2) Now choose "render mode" = "line contours", then specify contour levels as "[from, to, step]"

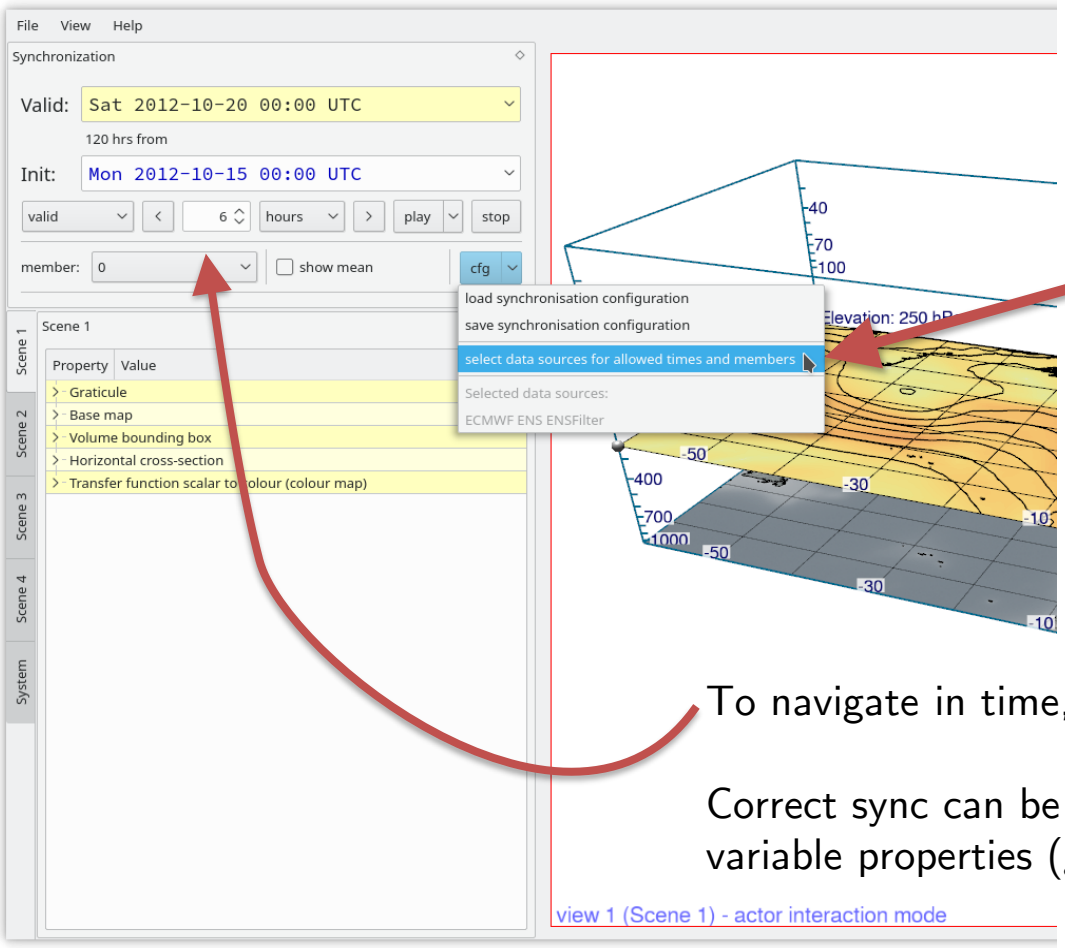


3) Contour colour, thickness etc. can be specified here. Multiple "contour sets" are possible.

“Actor interaction mode”: move horizontal section



Time navigation

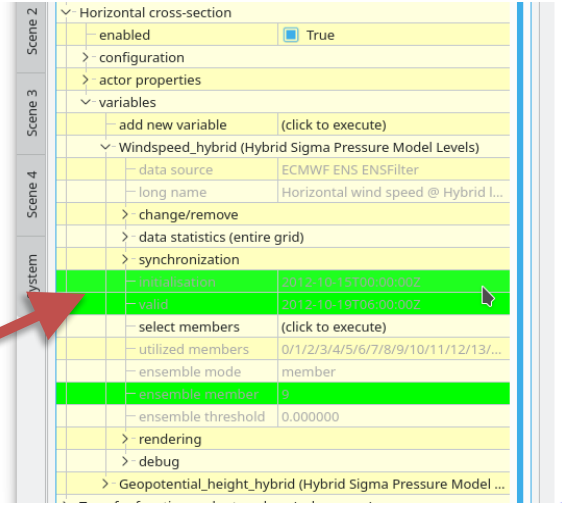


By default all variables are synchronized (can be disabled in actor properties).

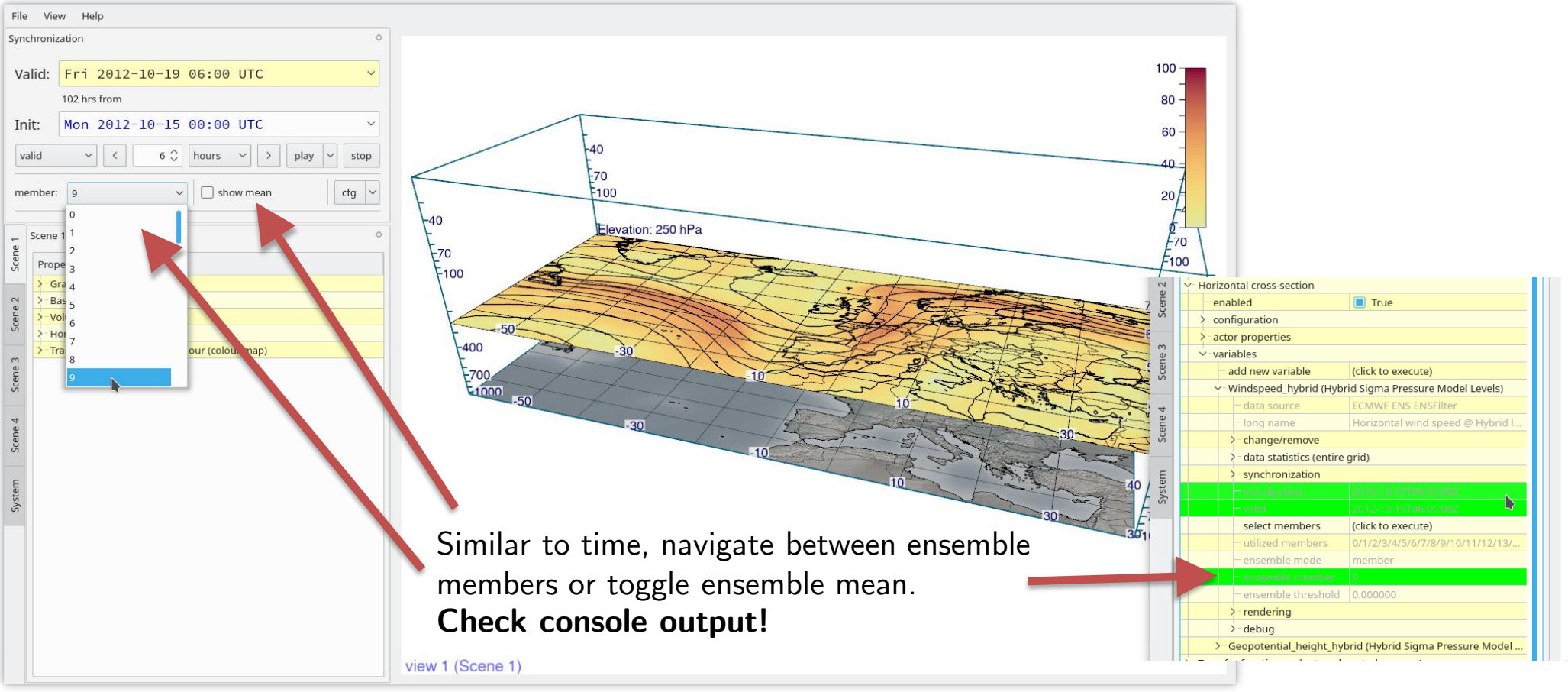
To restrict possible dates and times to values in dataset, "connect" dataset with sync control.

To navigate in time, use GUI elements.

Correct sync can be checked in actor variable properties (green = ok):

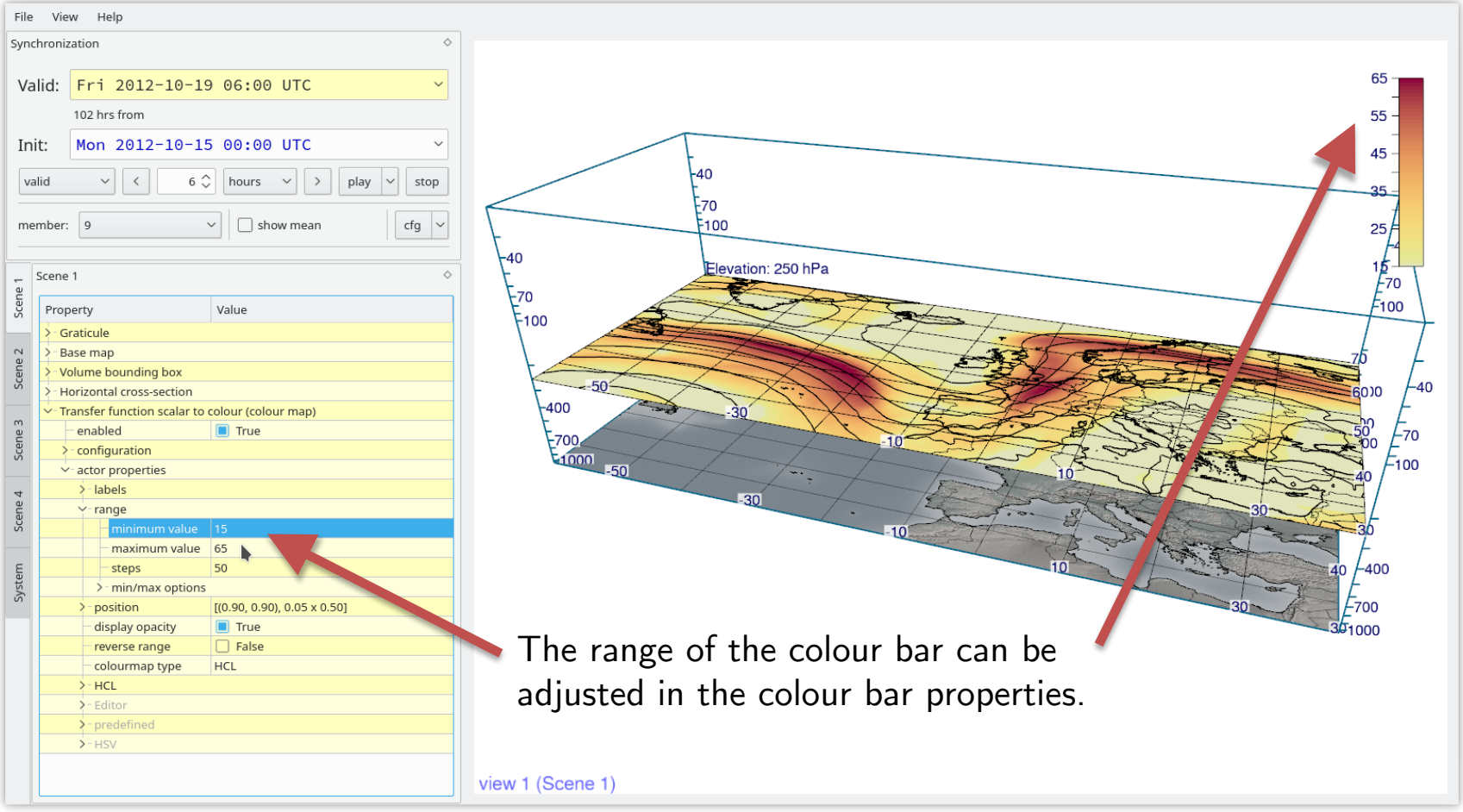


Ensemble navigation



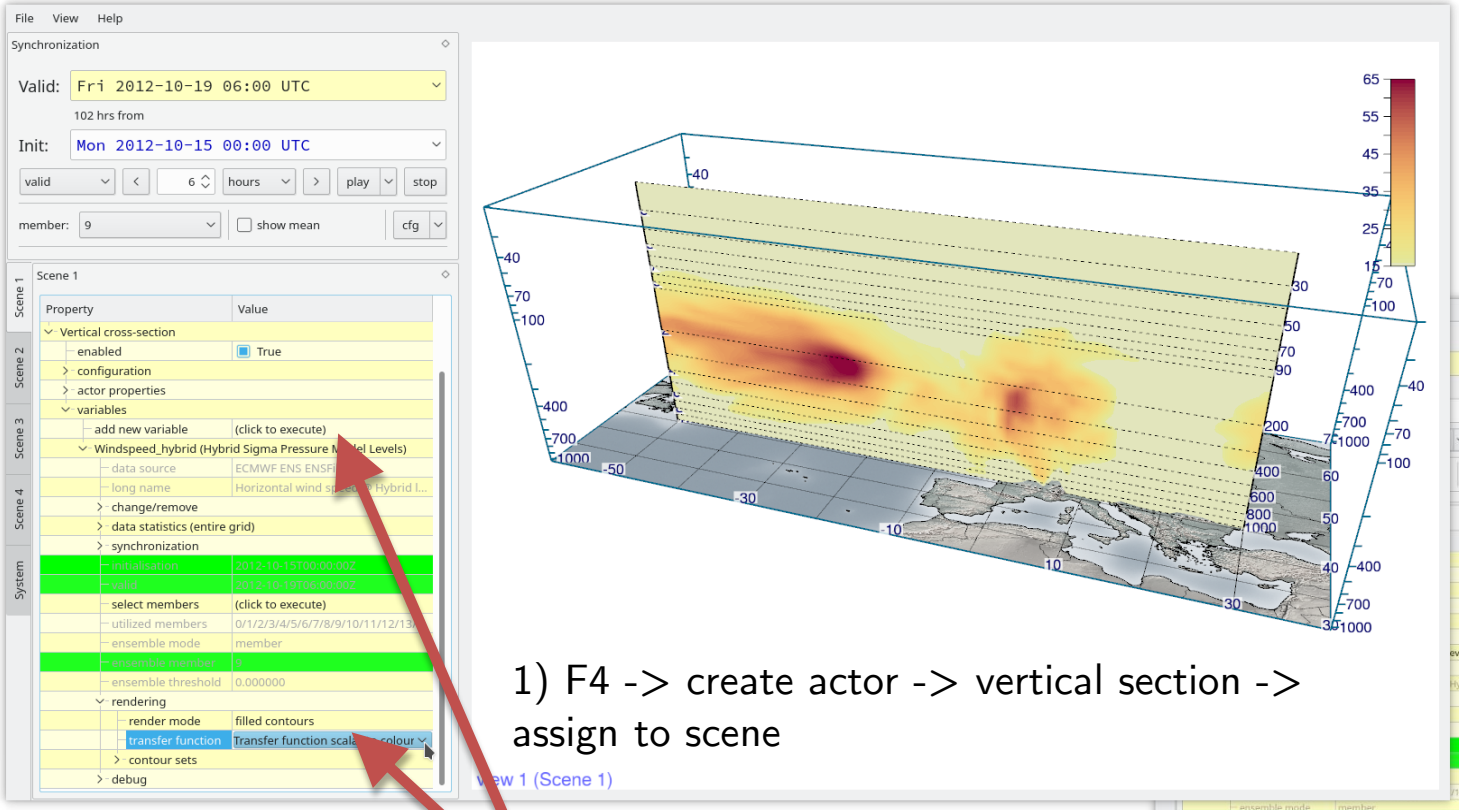
Similar to time, navigate between ensemble members or toggle ensemble mean.
Check console output!

Adjusting colour bar range



The range of the colour bar can be adjusted in the colour bar properties.

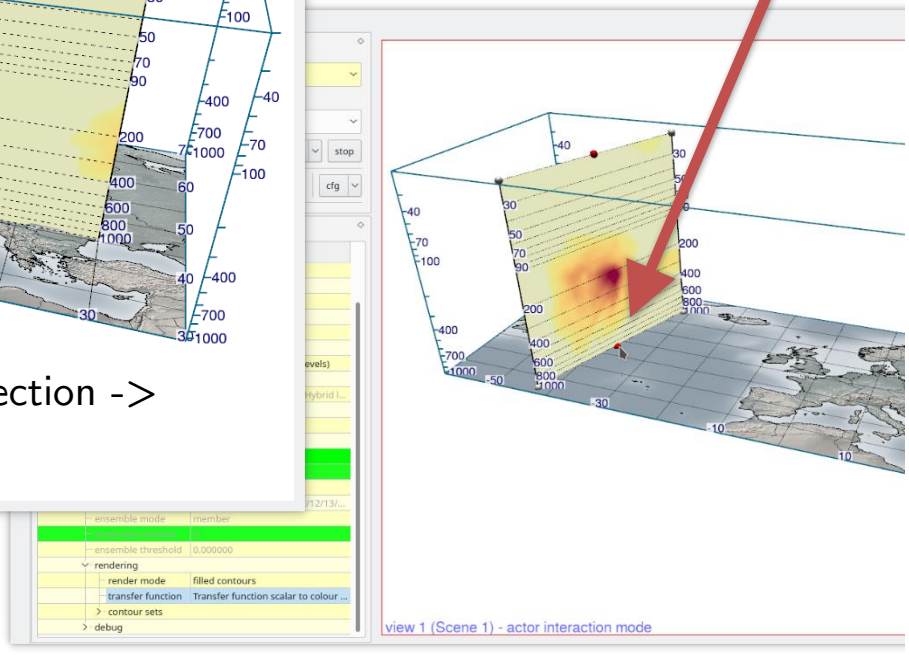
Similar: vertical sections



1) F4 -> create actor -> vertical section -> assign to scene

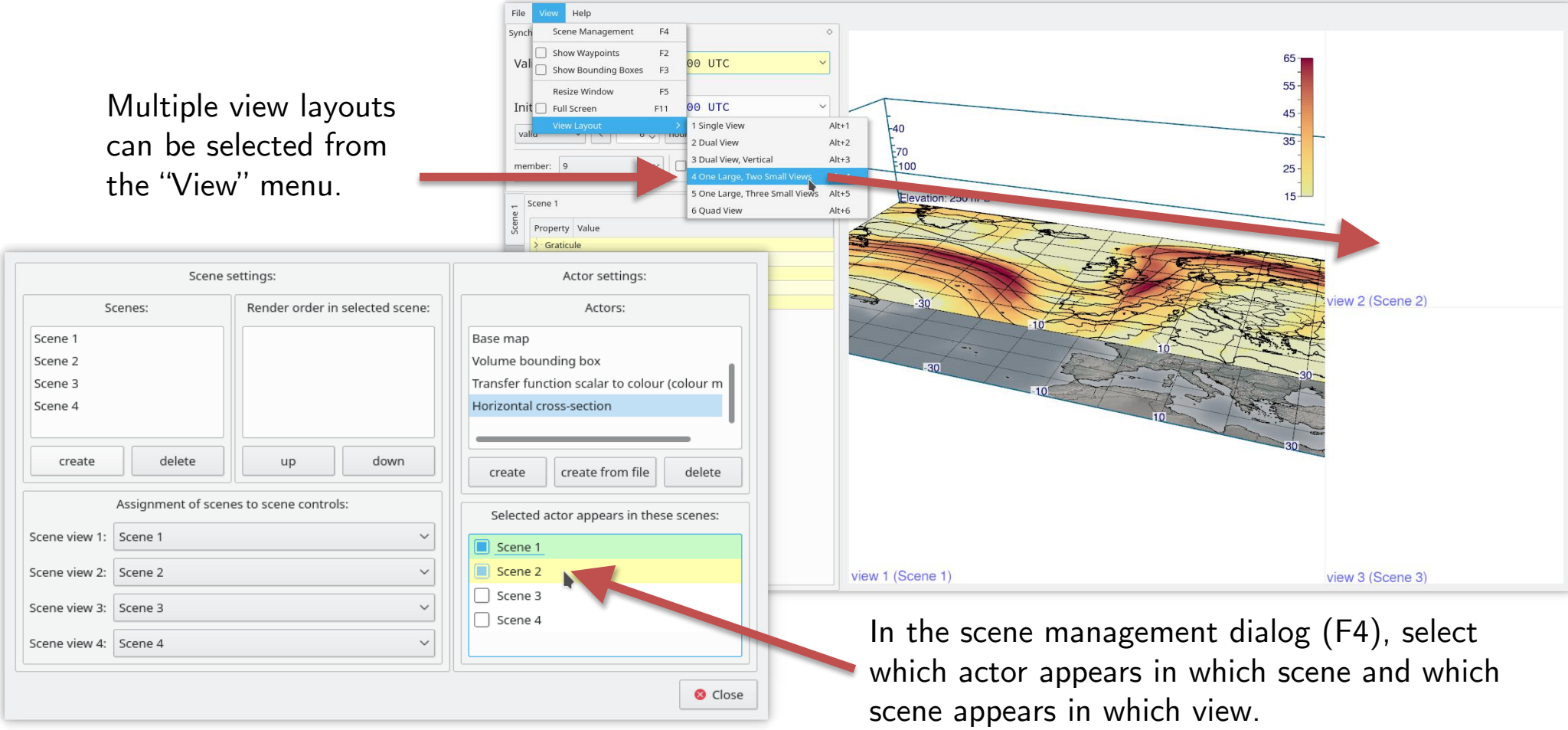
2) Add variable (here: wind speed) -> render mode = "filled contours" -> select transfer function

3) Toggle actor interaction with double click -> click and drag handles



System settings: multiple views, vertical scaling etc.

Multiple view layouts can be selected from the "View" menu.

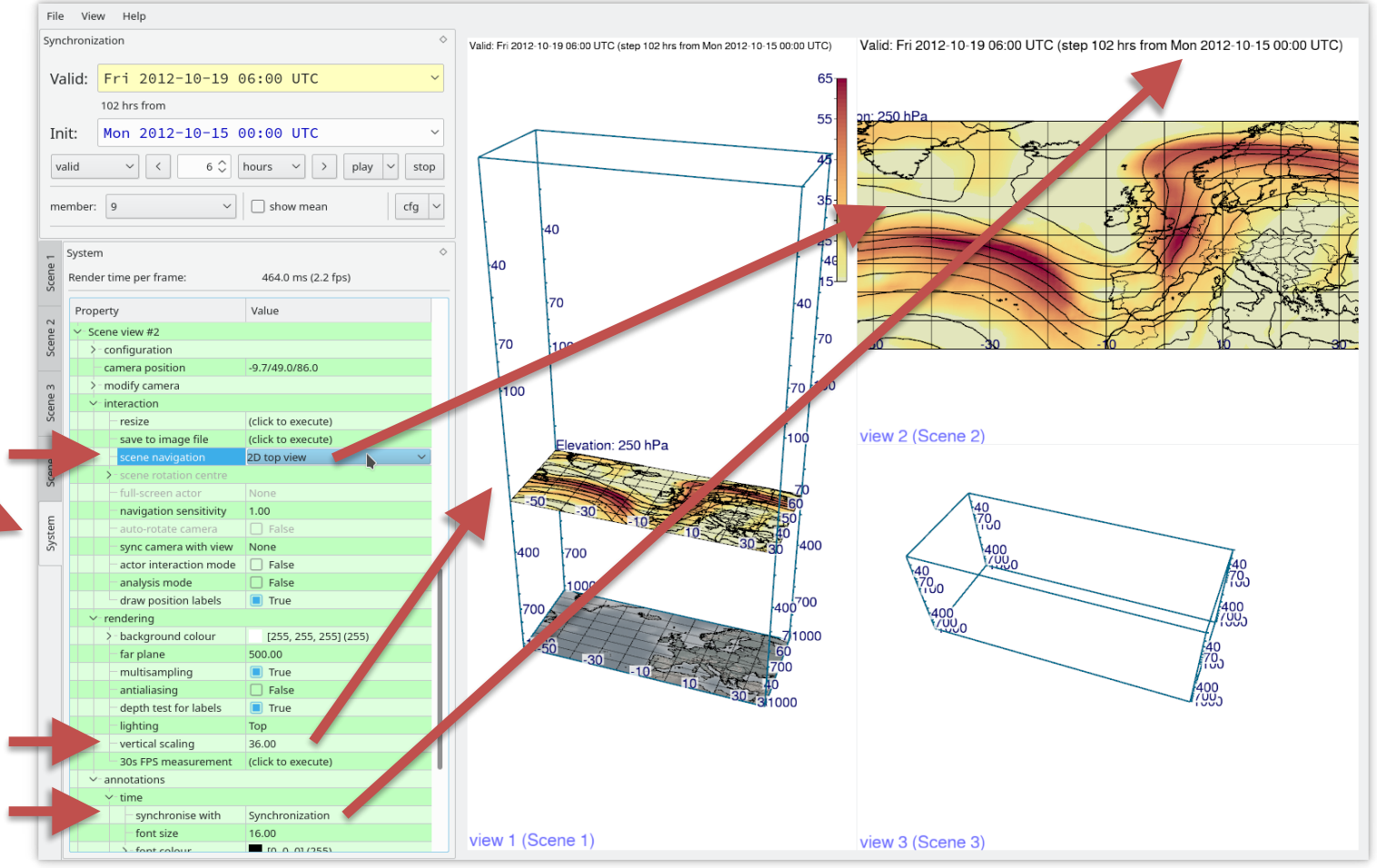


In the scene management dialog (F4), select which actor appears in which scene and which scene appears in which view.

System settings: multiple views, vertical scaling etc.

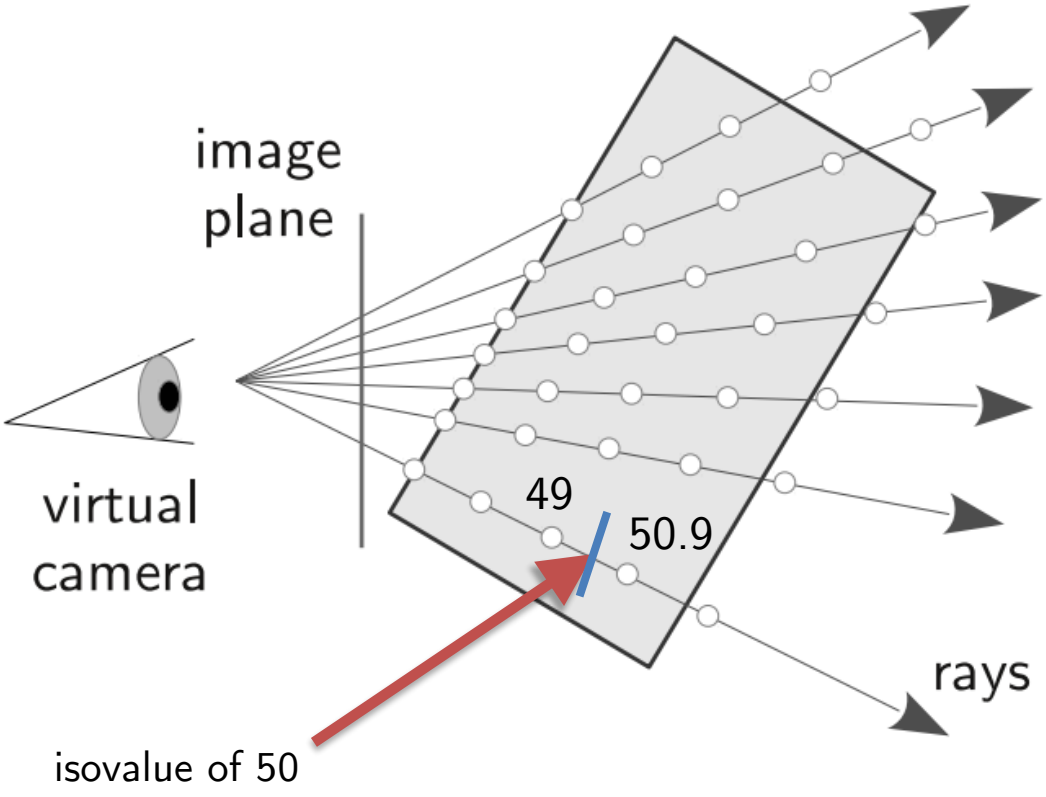
In the "System" tab, properties for each of the scene views can be set.

Examples: type of scene navigation, vertical scaling of the 3D view, some annotations, synchronizing the camera viewpoint between two views.



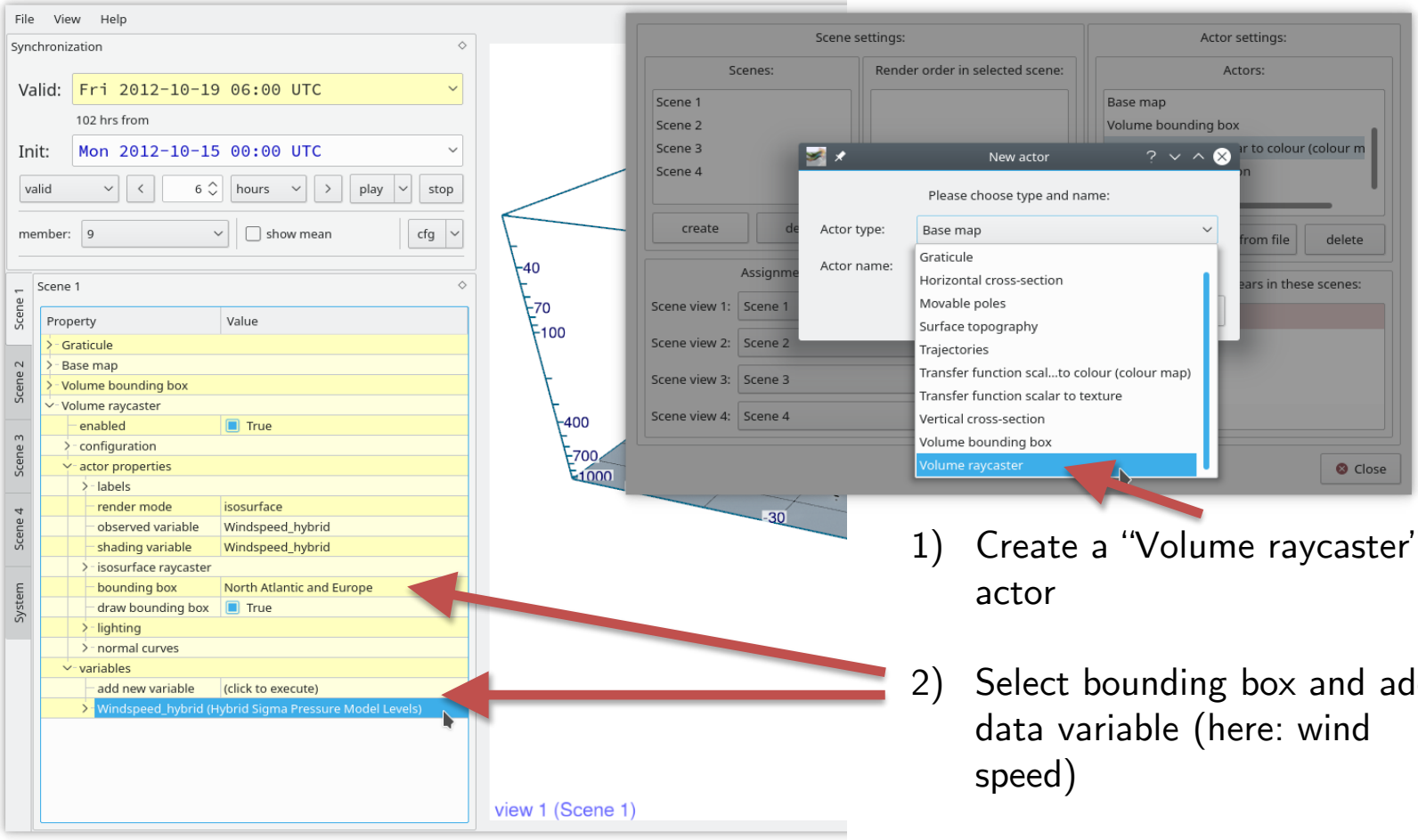
Met.3D: Isosurfaces

3D isosurfaces and direct volume rendering (DVR) are implemented using "raycasting", i.e. a ray is "shot" from the camera through each image pixel and through the data volume to obtain the required visualization information.



Met.3D: Isosurfaces

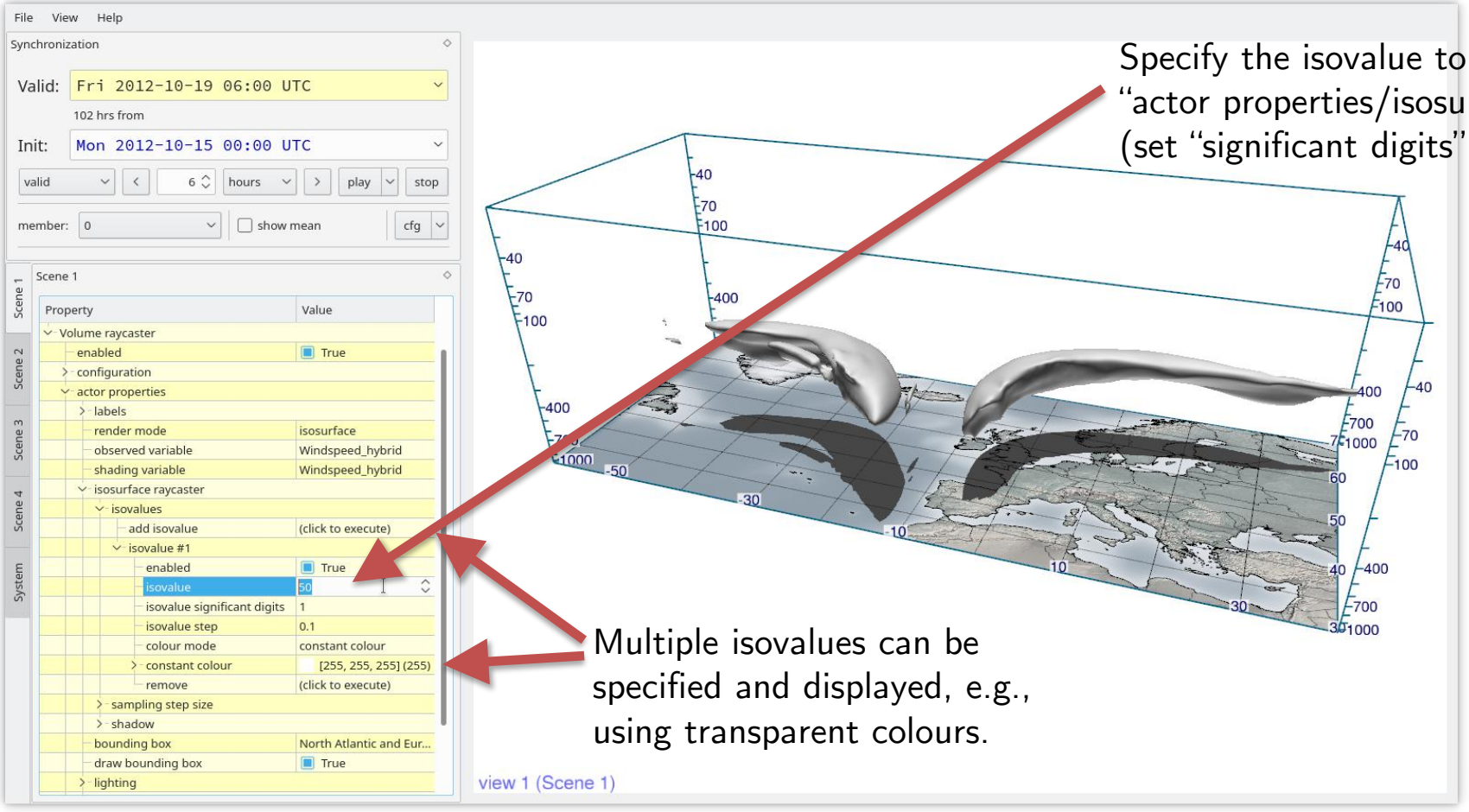
3D isosurfaces and direct volume rendering (DVR) are implemented using "raycasting", i.e. a ray is "shot" from the camera through each image pixel and through the data volume to obtain the required visualization information.



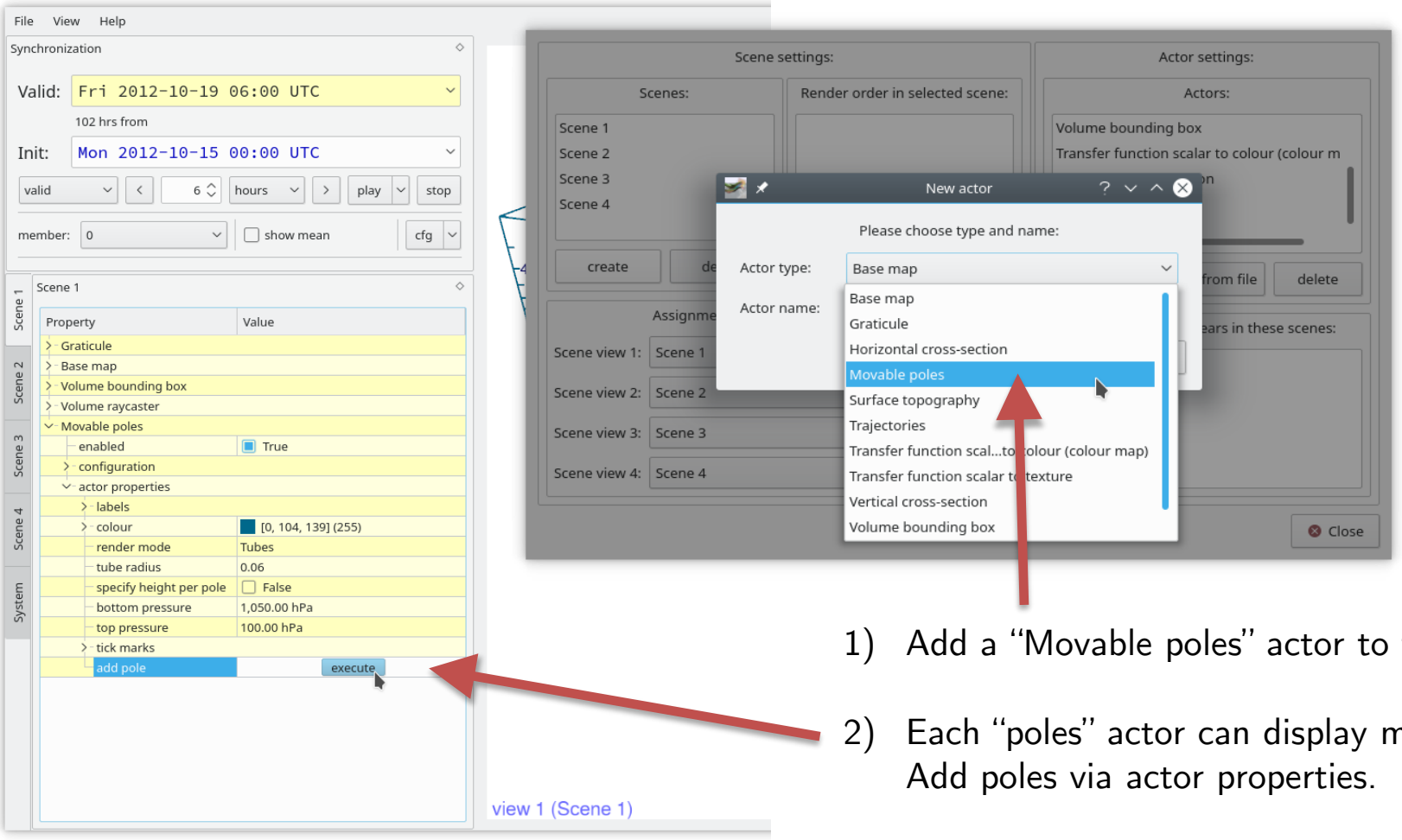
1) Create a "Volume raycaster" actor

2) Select bounding box and add data variable (here: wind speed)

Met.3D: Isosurfaces



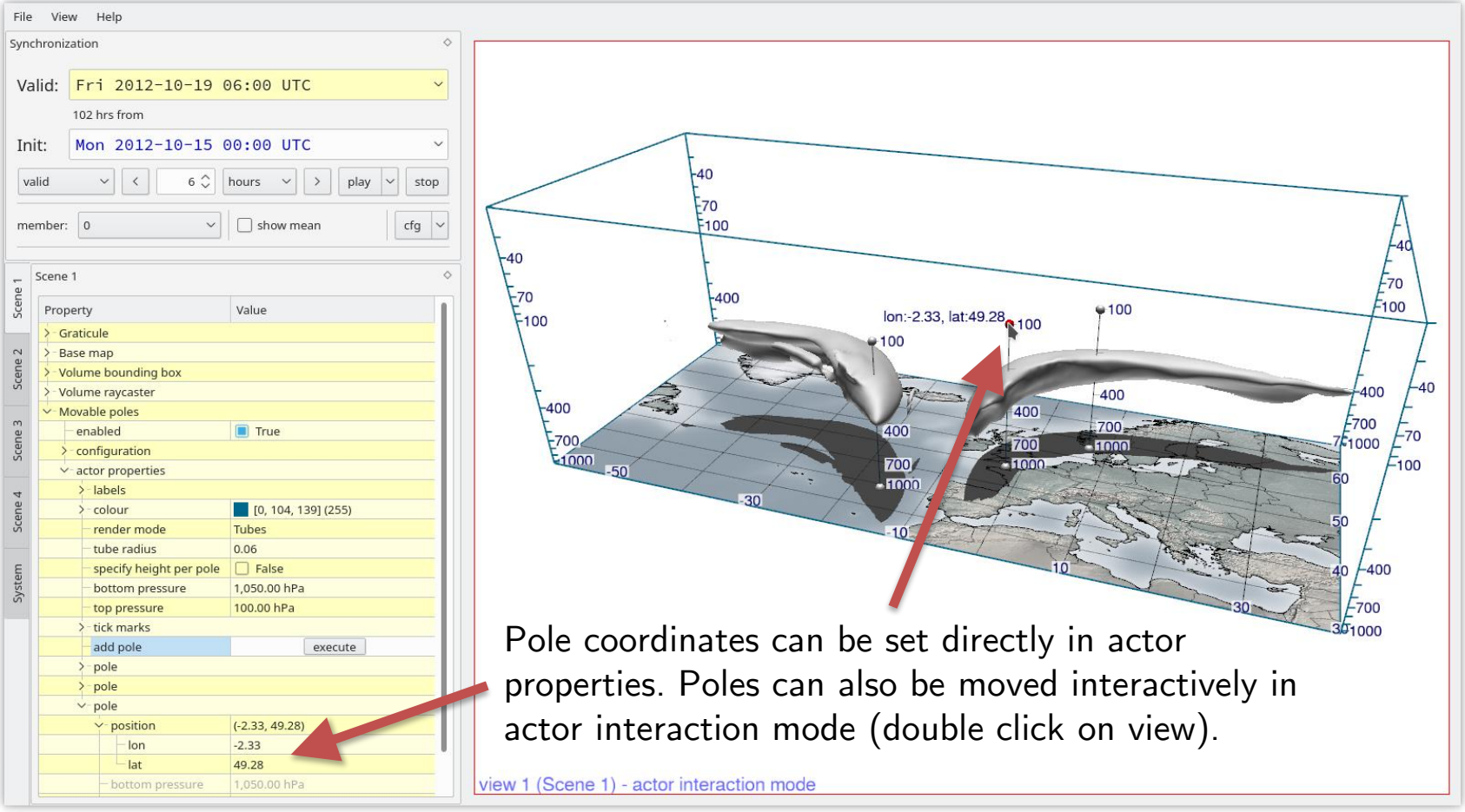
Improve spatial perception: add poles



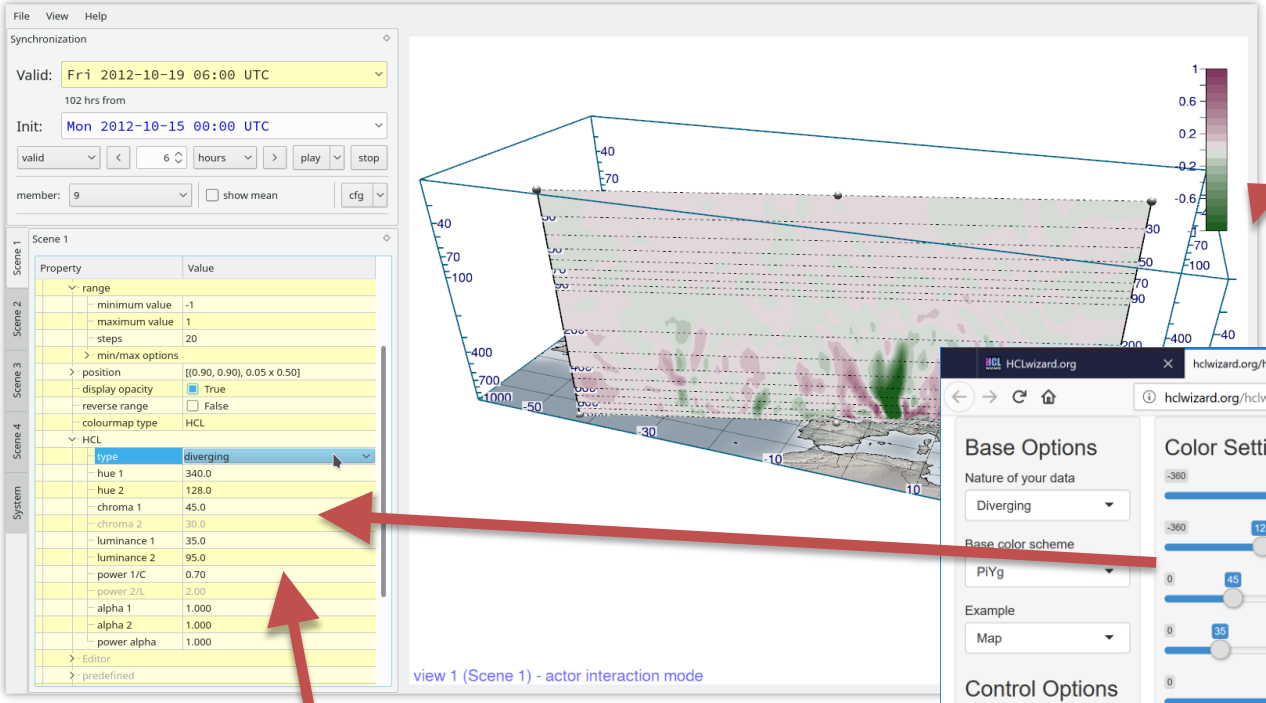
1) Add a "Movable poles" actor to the scene.

2) Each "poles" actor can display multiple poles.
   Add poles via actor properties.

Improve spatial perception: add poles

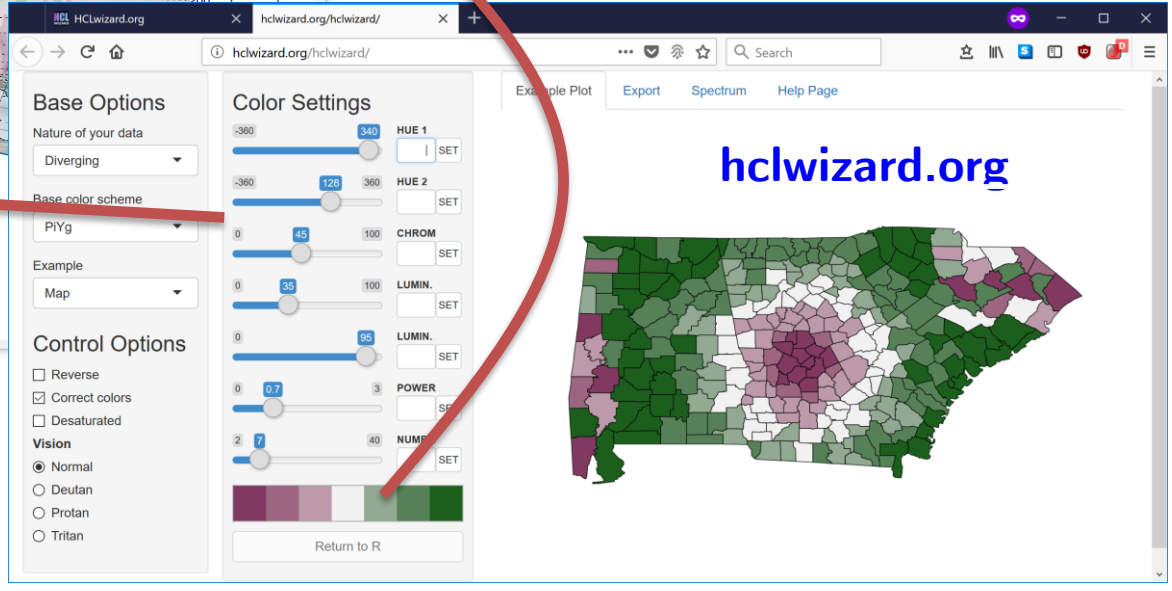


Transfer functions (colour bars): HCL



Hue-Chroma-Luminance (HCL) is a perceptually linear colour space → no misleading colour gradients

1) Choose/create HCL colour scale with "HCL Wizard" online



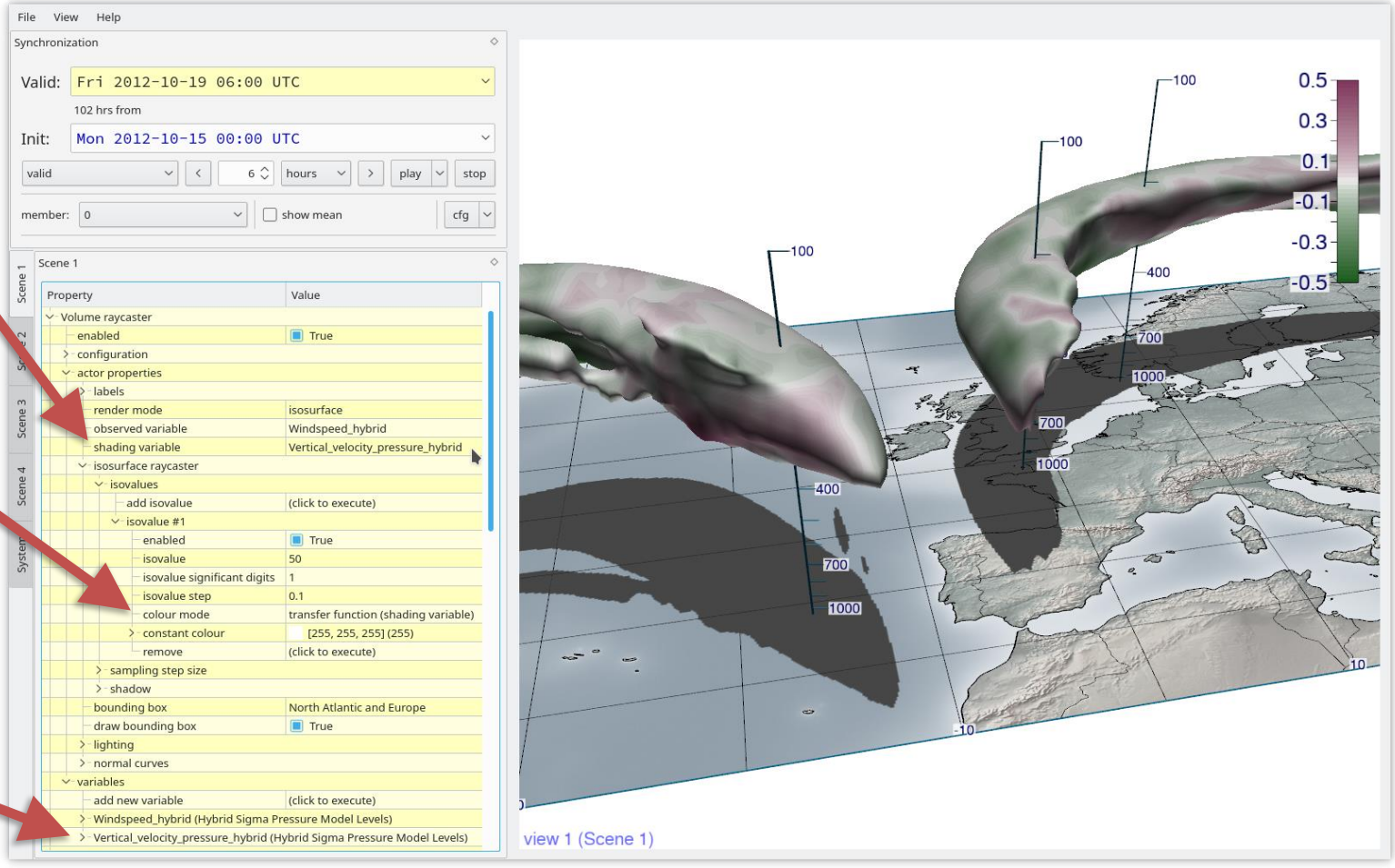
2) Copy parameters to Met.3D transfer function actor

Colour an isosurface using a second variable

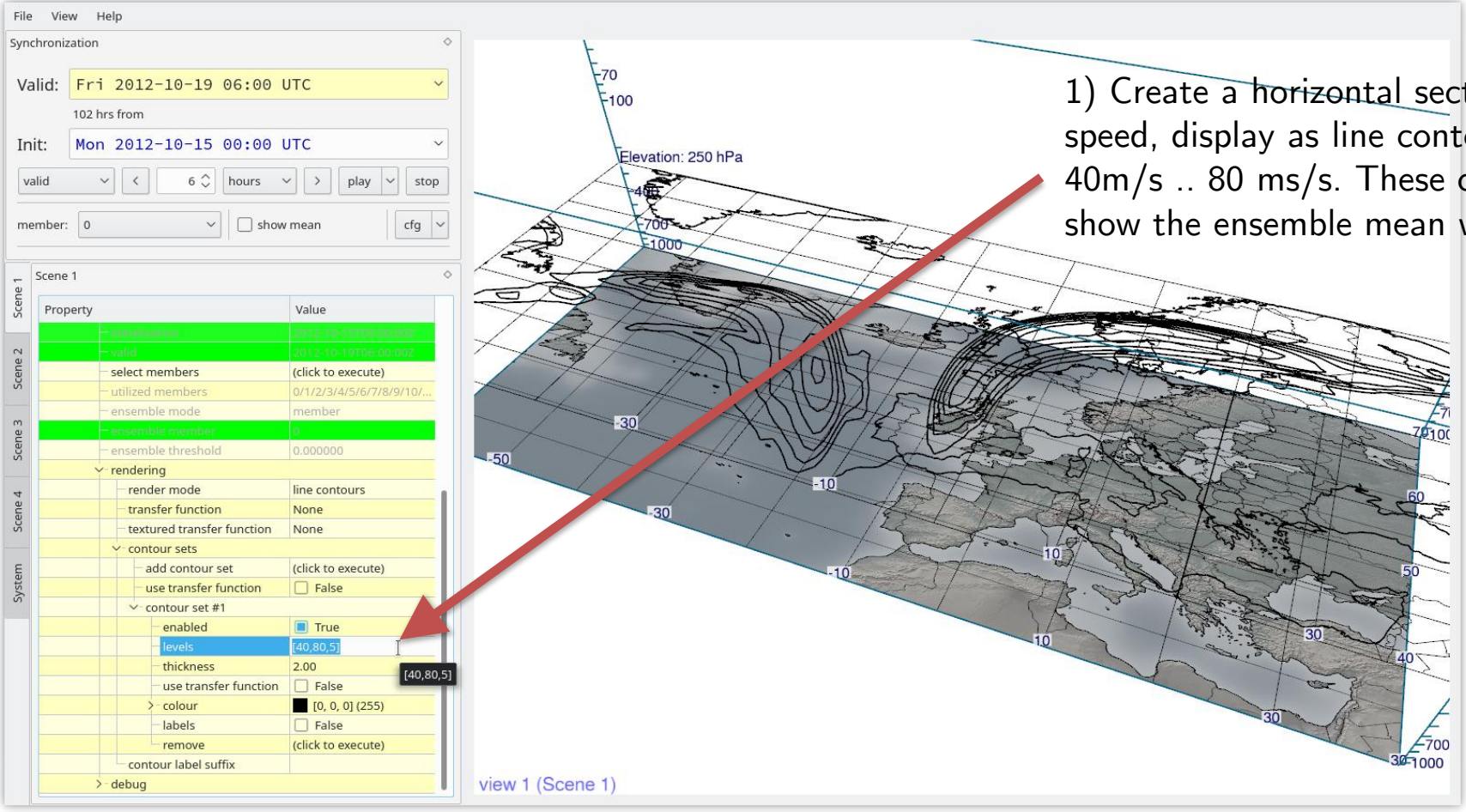
2) To colour ("shade") the isosurface, select the new variable as "shading variable".

3) For the isovalue of interest, set "colour mode" to "transfer function".

1) Add a new data variable (here: vertical velocity). In the variable's properties, in the "render mode" group, select a transfer function.

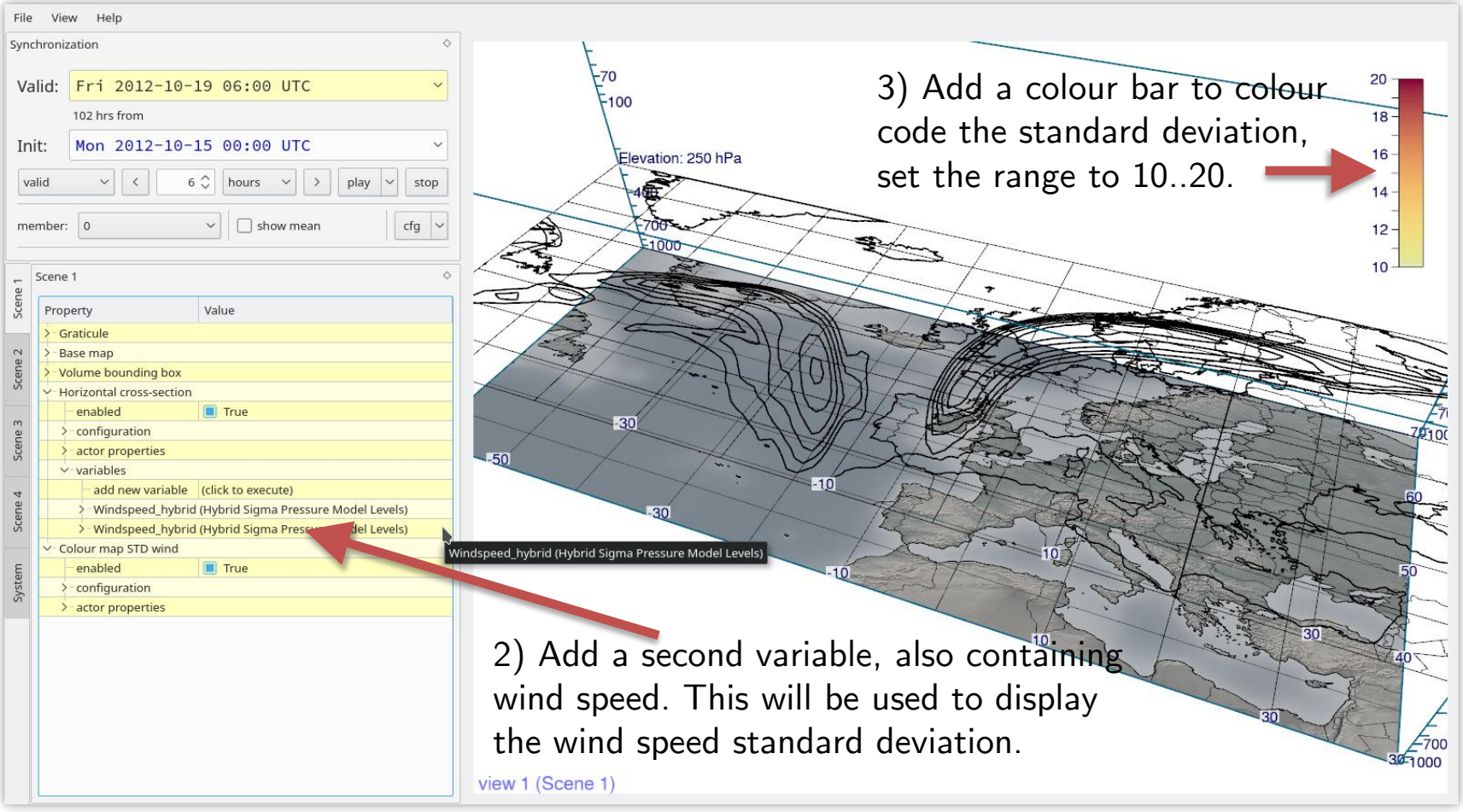


Ensemble statistics: Wind speed mean+std.dev

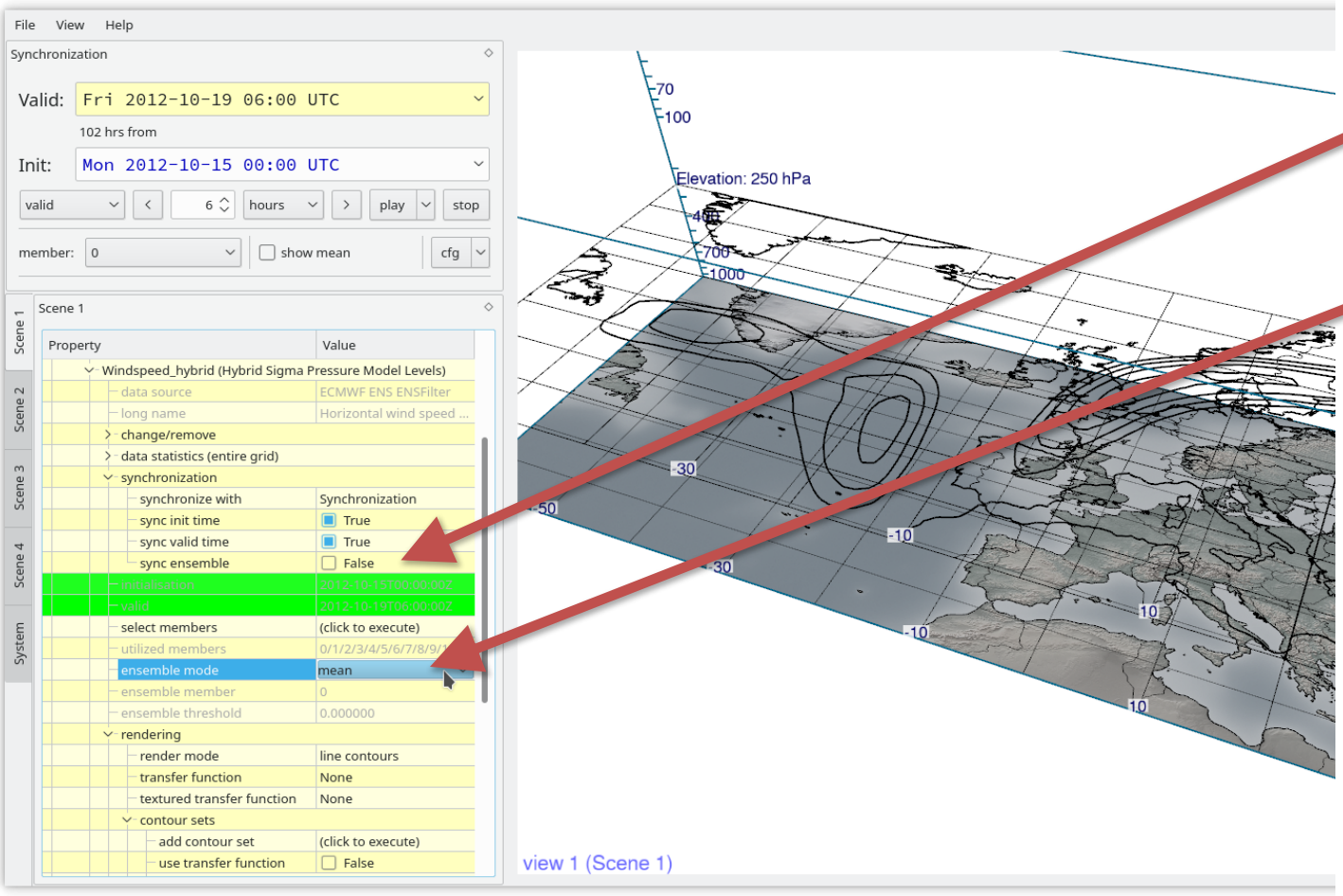


1) Create a horizontal section, add wind speed, display as line contours in the range 40m/s .. 80 ms/s. These contours will later show the ensemble mean wind speed.

Ensemble statistics: Wind speed mean+std.dev

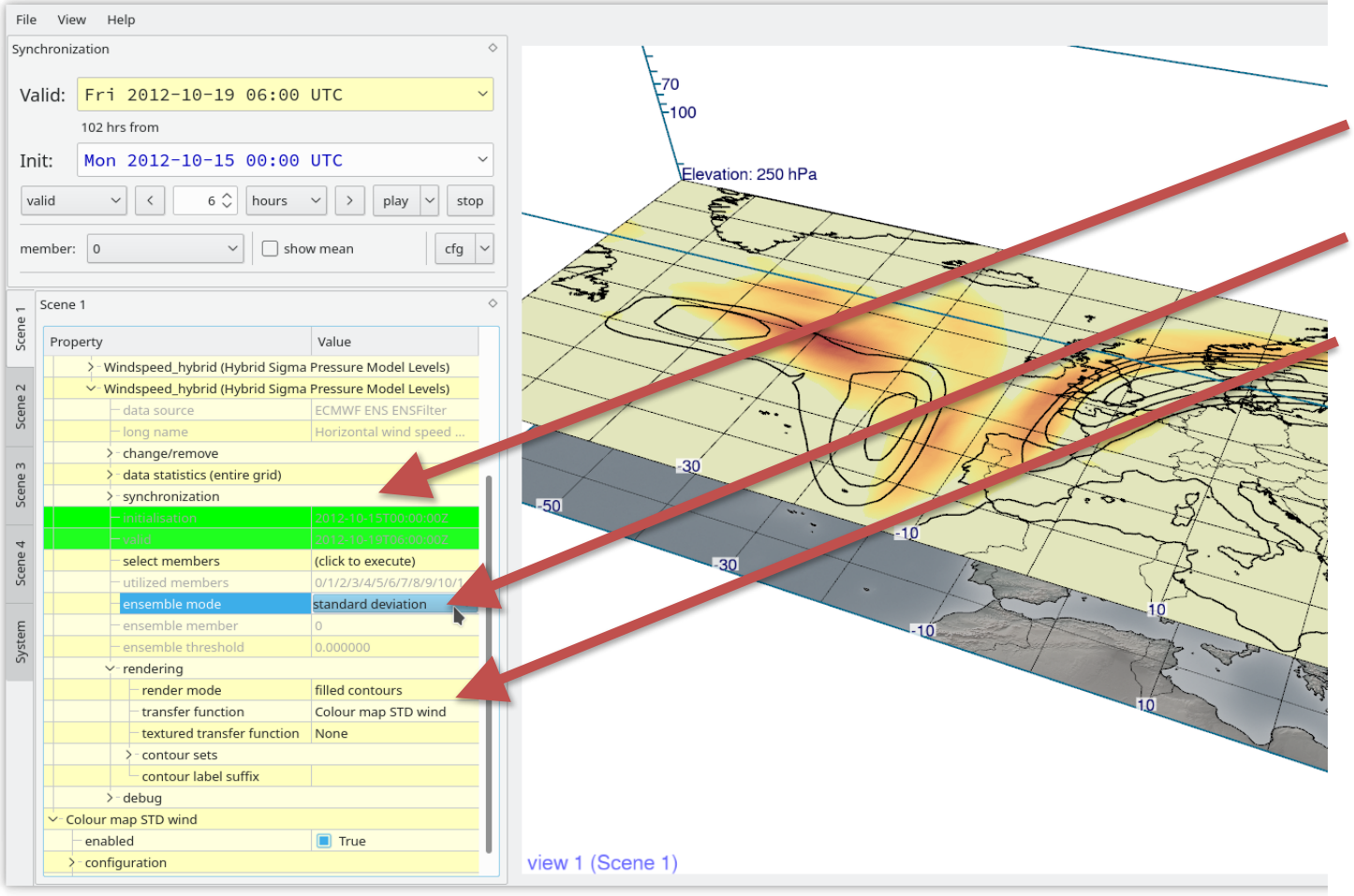


Ensemble statistics: Wind speed mean+std.dev



4) For the first wind speed variable (contours), disable ensemble synchronization to allow local setting of ensemble mode, then set its ensemble mode to “mean”.

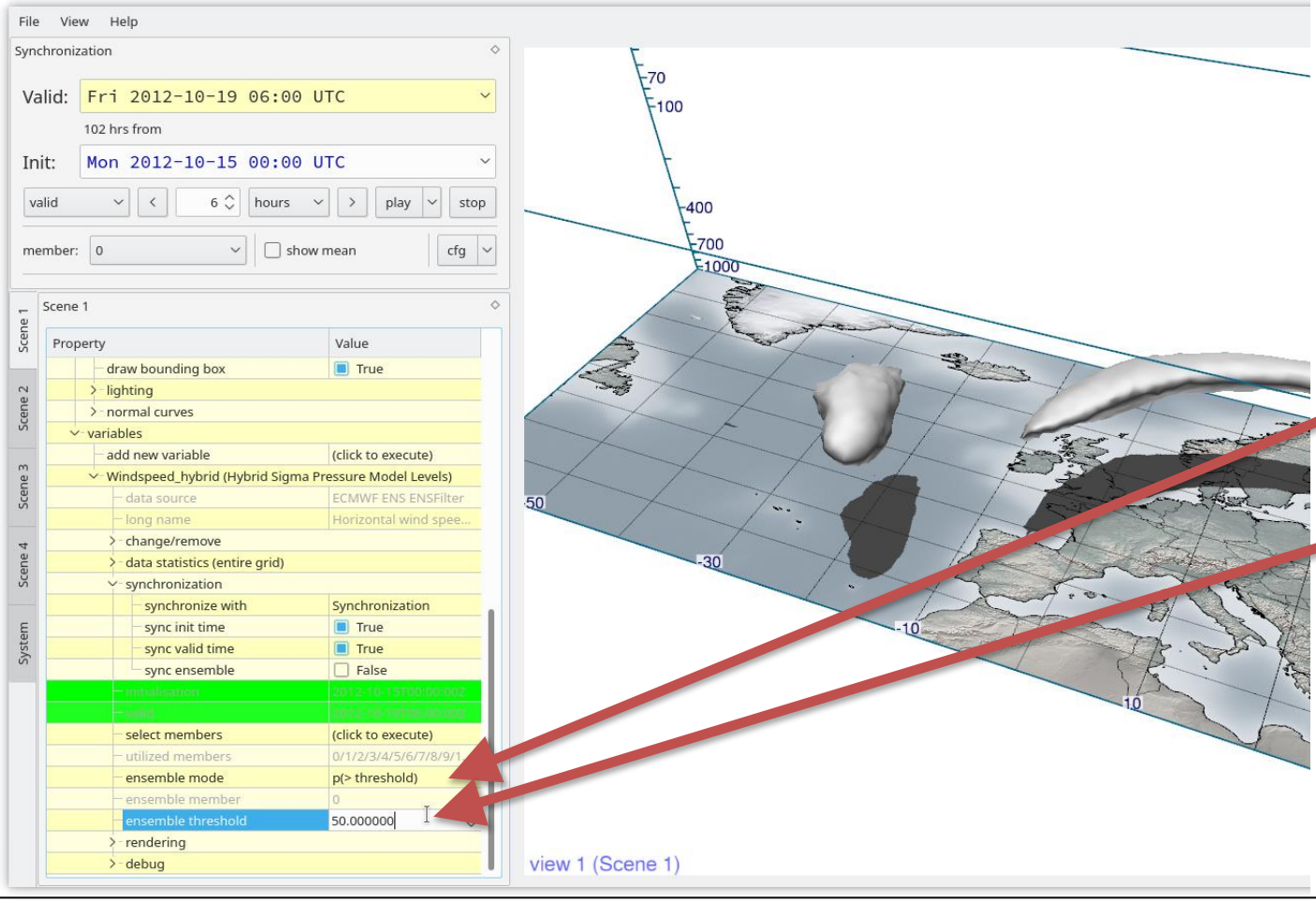
Ensemble statistics: Wind speed mean+std.dev



5) For the second wind speed variable (std.dev.), also disable ensemble synchronization. Then set its ensemble mode to "standard deviation"; for rendering select "filled contours" and connect to the colour map.

Investigate vertical structure and temporal development of std.dev.

Ensemble statistics: probabilities

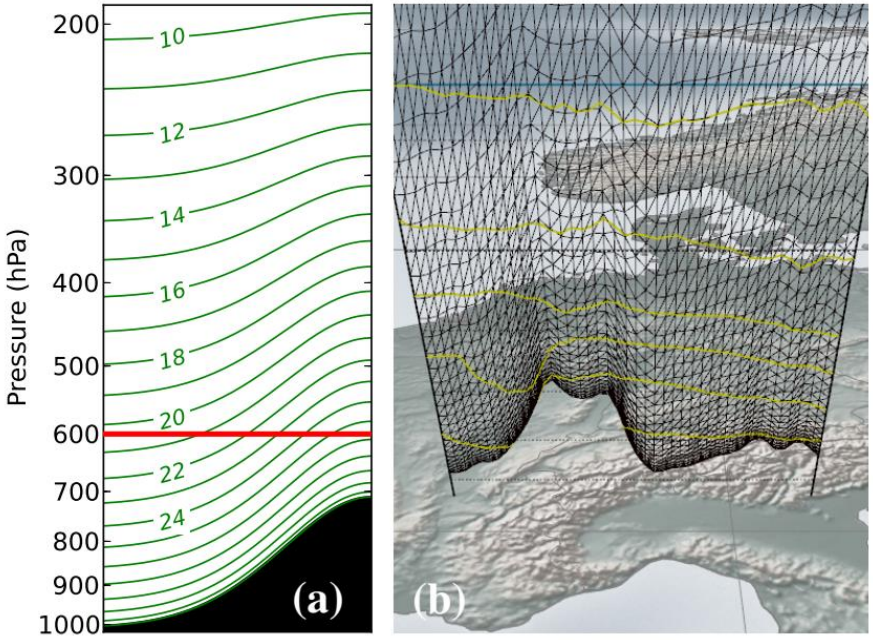


All actors that display simulation data can also display these basic ensemble statistics.

Another example: Create a volume raycaster, add wind speed, disconnect ensemble synchronization, set the ensemble mode to "p > threshold" (= probability that threshold is exceeded), set the threshold to 50m/s, set the isovalue to 0.5.

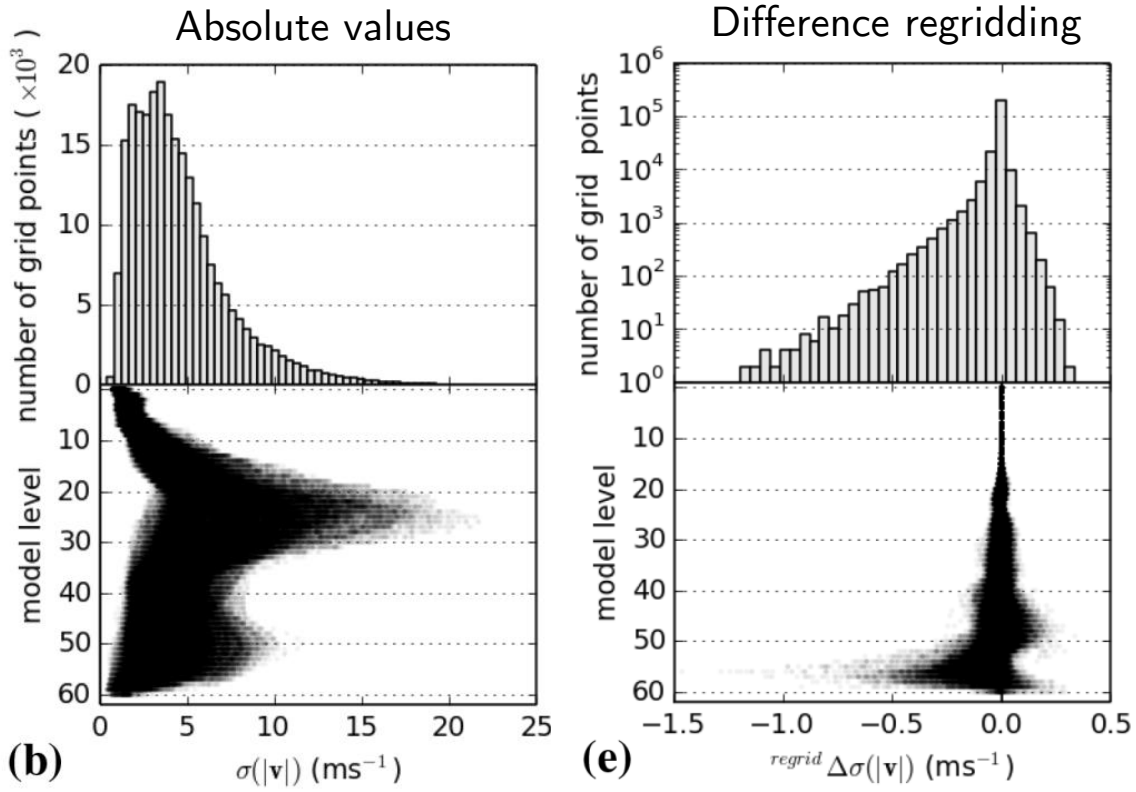
3D regions are shown in which the probability that the wind speed exceeds 50m/s is larger than 50%.

Ensemble statistics: Caution with hybrid grids



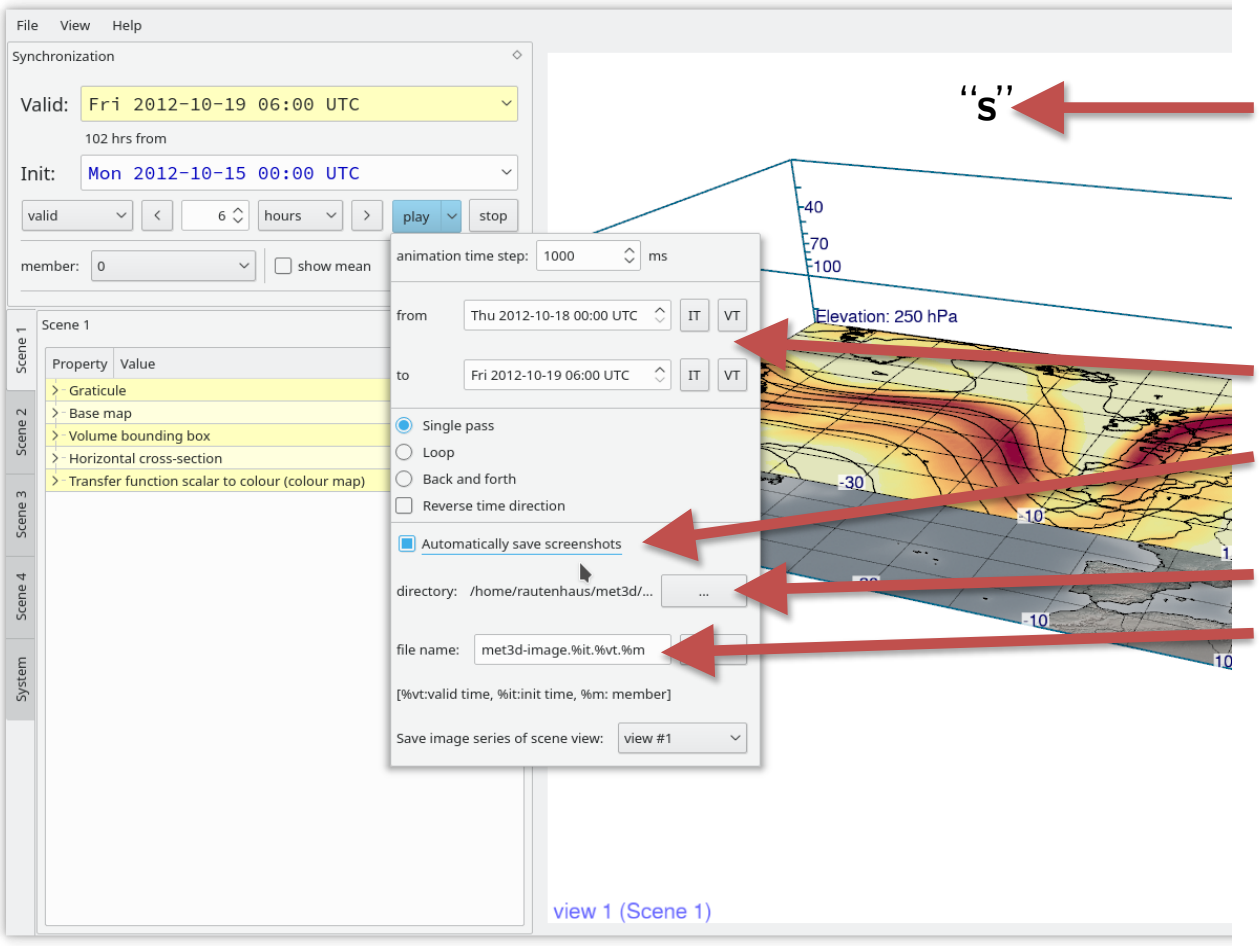
Vertical position of grid points is slightly offset in each member.

Standard deviation of wind speed:



Rautenhaus, Kern, Schäfler, Westermann (Geosci. Model Dev. 2015)

Saving images



To save individual images: Click on the scene view and press "s". A "save-file-dialog" will appear.

Entire time series can be saved from the animation control. Set animation parameters (from which time to which time) and enable "automatically save screenshots".

Select a directory for the images and a file name template (init and valid time and ensemble member can be automatically replaced).

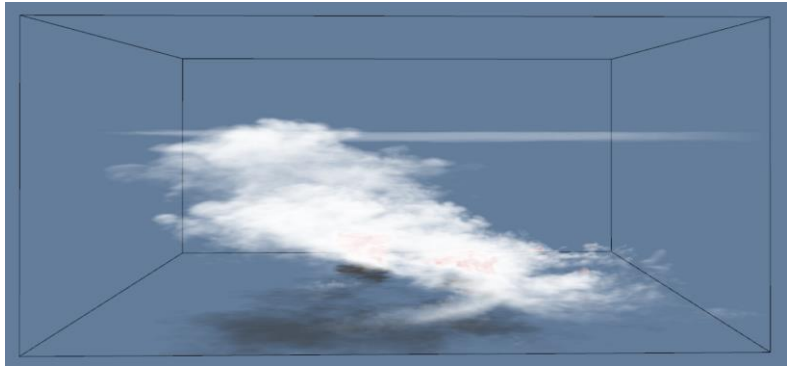
When running the animation, all images will be saved.

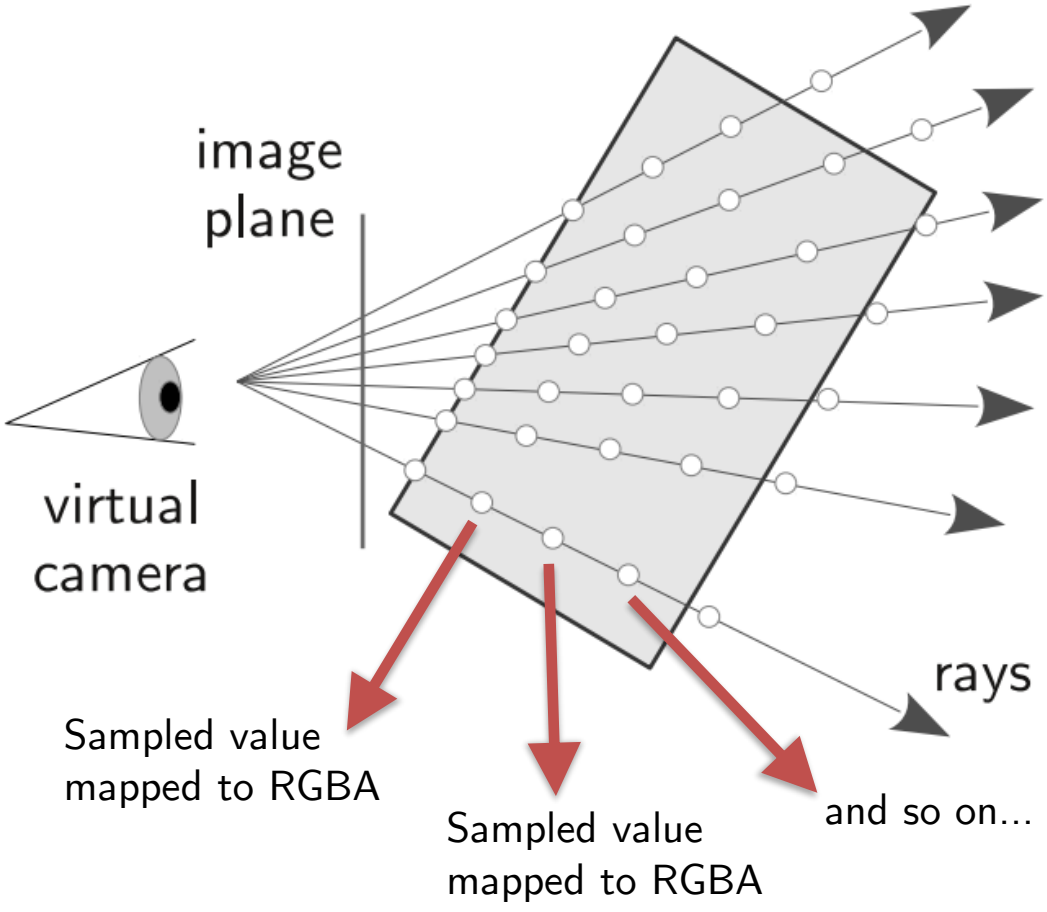
Direct volume rendering

Direct volume rendering implements a simple "emission-absorption model":

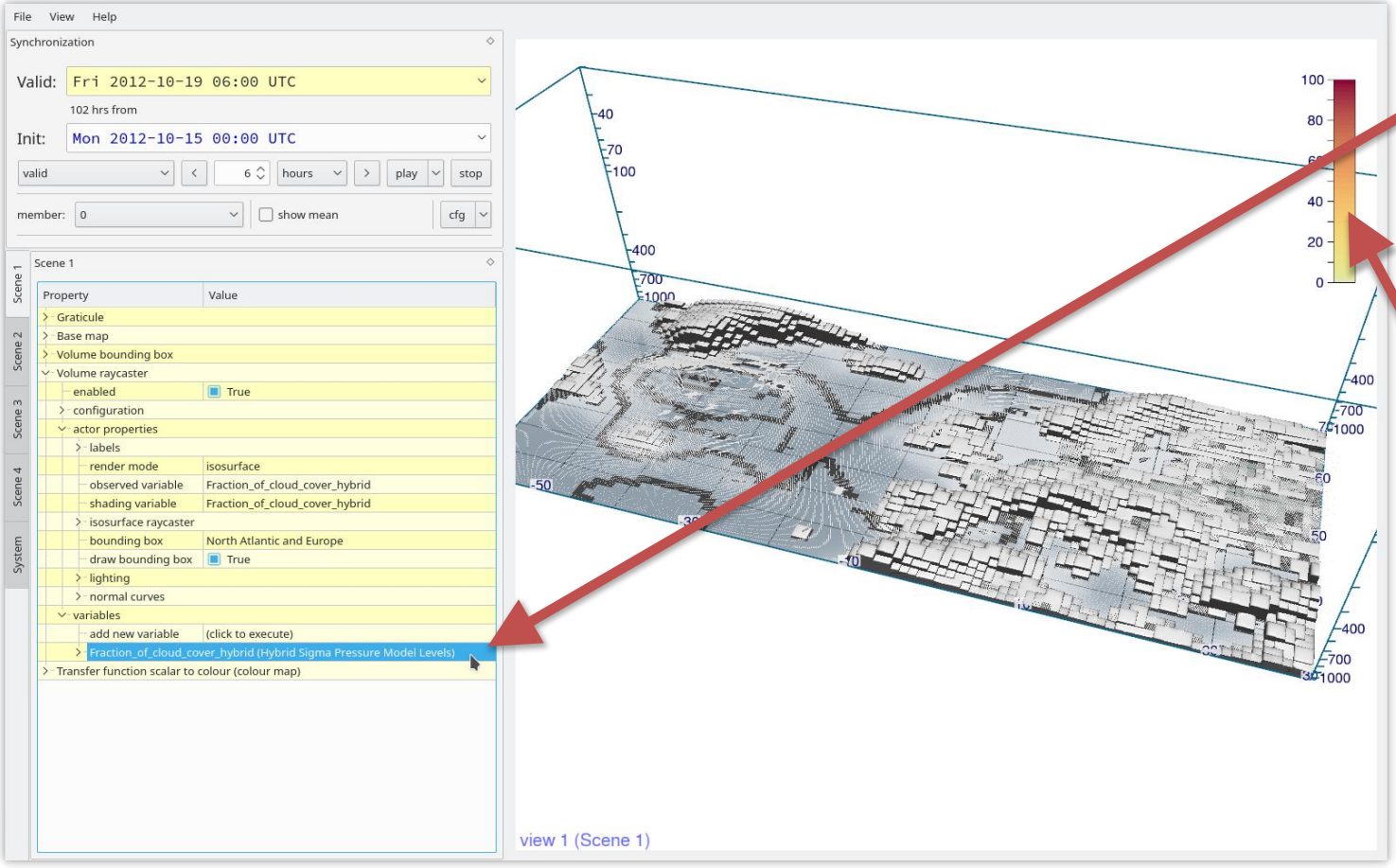
Rays are casted through the data volume; at each sample location the encountered scalar value is mapped to colour (emission) and opacity (absorption).

The sample values are "merged" ("alpha blending") to obtain the pixel colour.





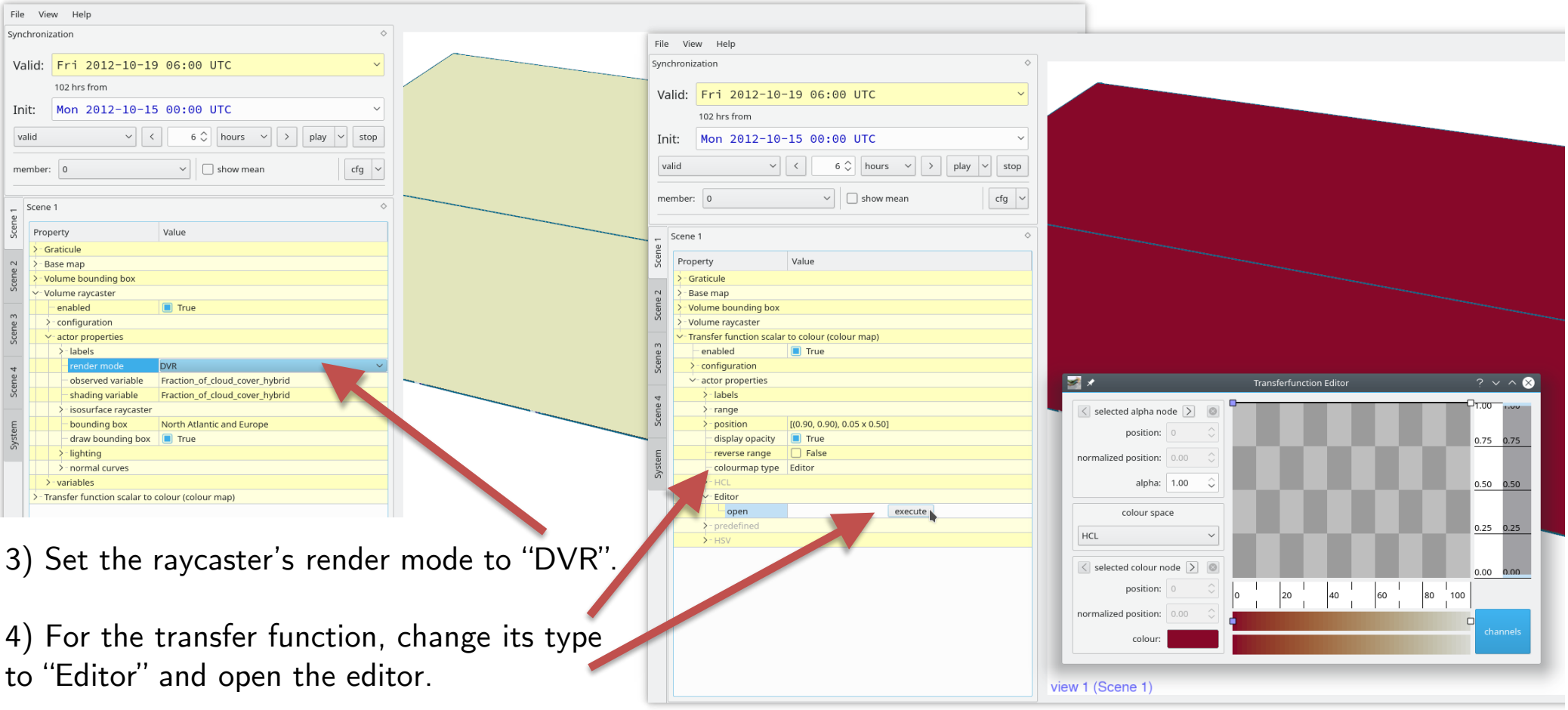
Direct volume rendering



1) Create a new "volume raycaster" actor, select a bounding box and add a variable. Here, we use "fraction of cloud cover".

2) Add a new transfer function, connect it to the cloud cover variable (in the variables properties in "rendering").

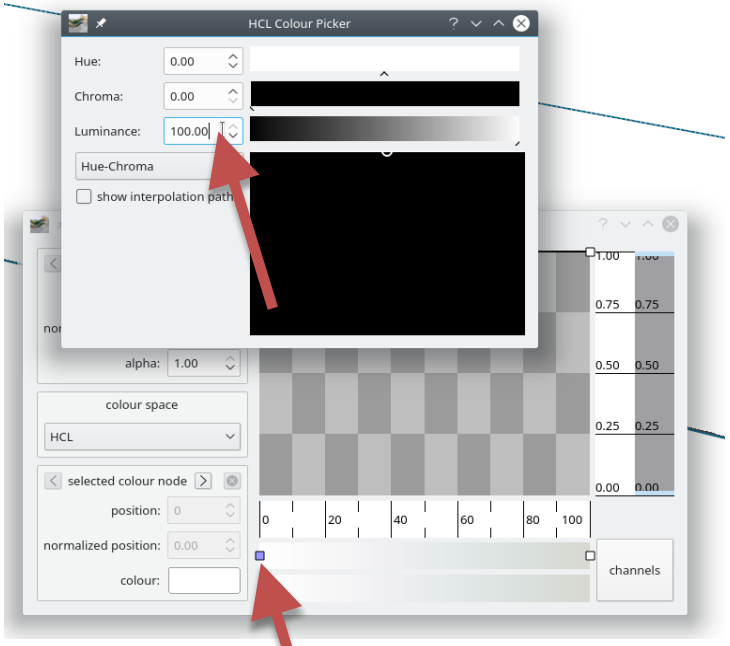
Direct volume rendering



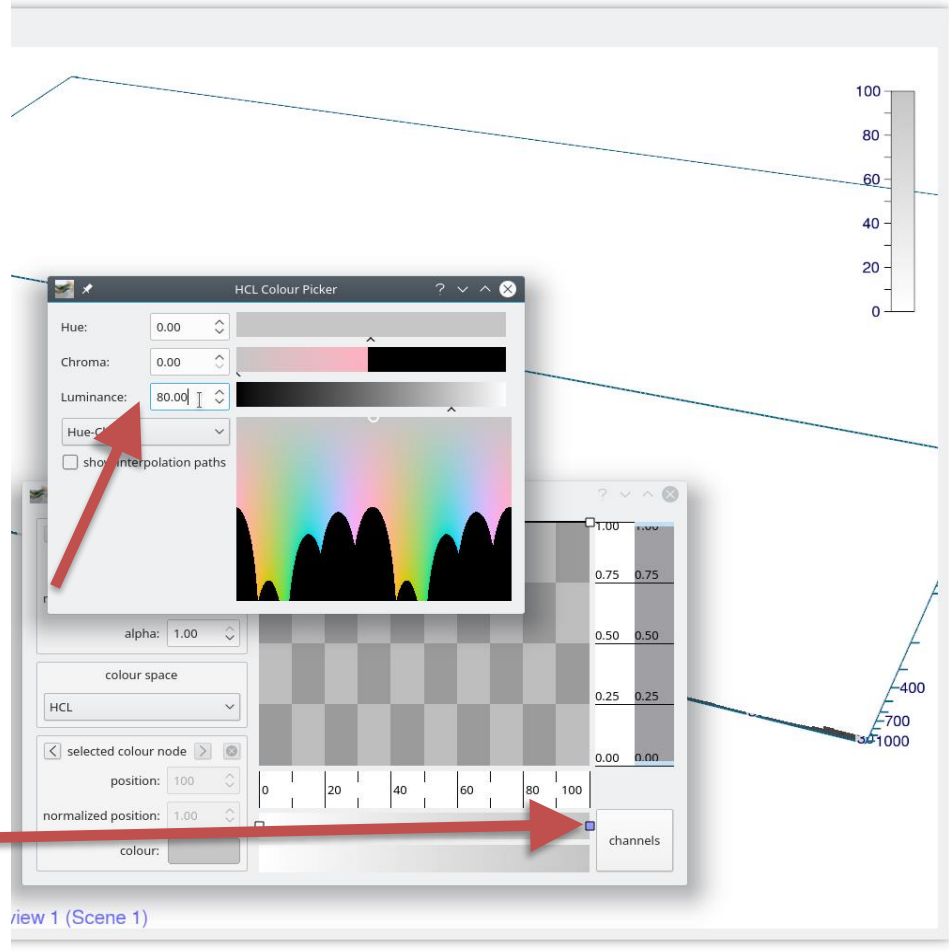
3) Set the raycaster's render mode to "DVR".

4) For the transfer function, change its type to "Editor" and open the editor.

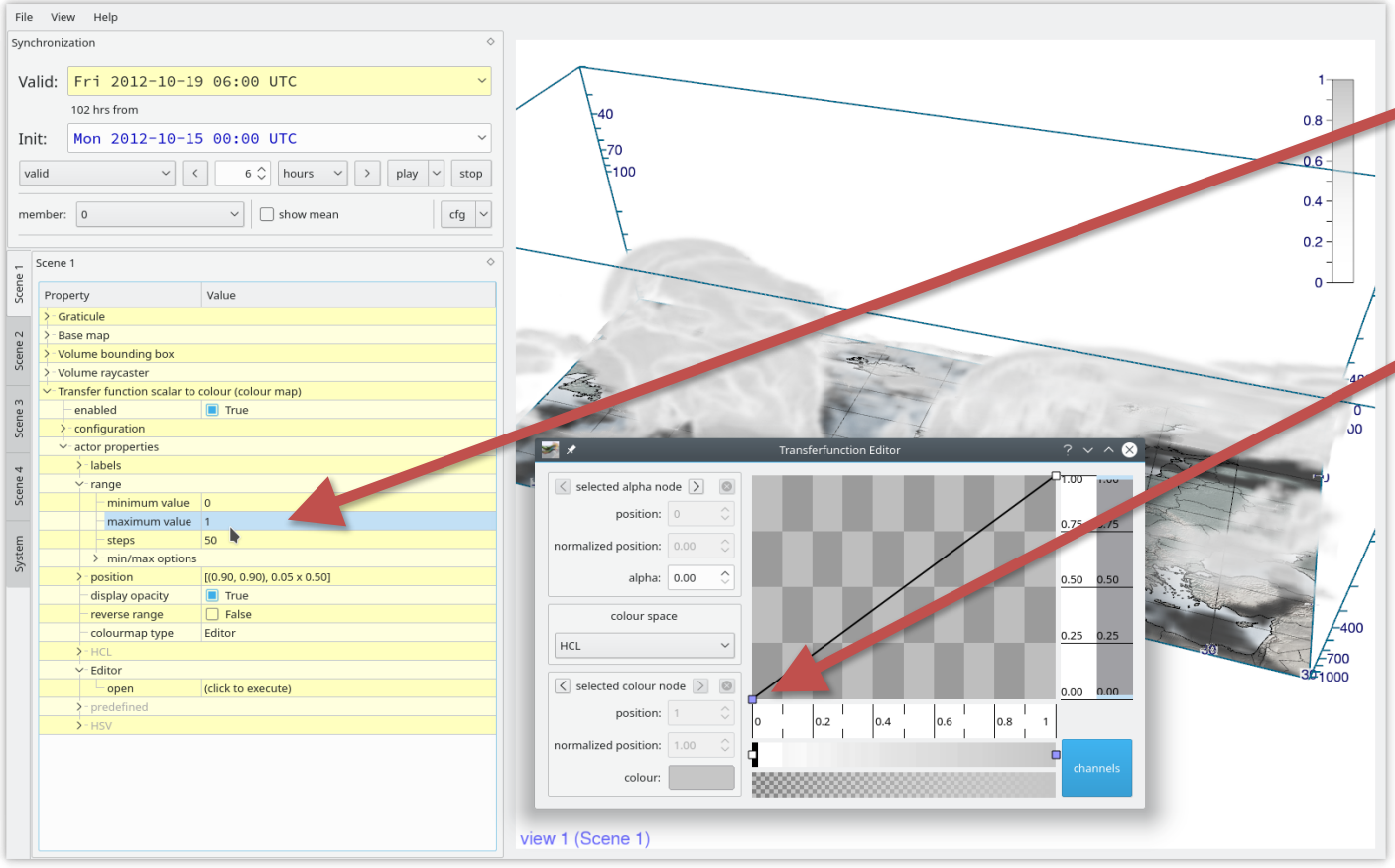
Direct volume rendering



5) Change the default colour map to a "white-to-grey" scale: Double click on the left colour node, set HCL values for "white" (0-0-100). Repeat for the right colour node, set a grey value (0-0-80).



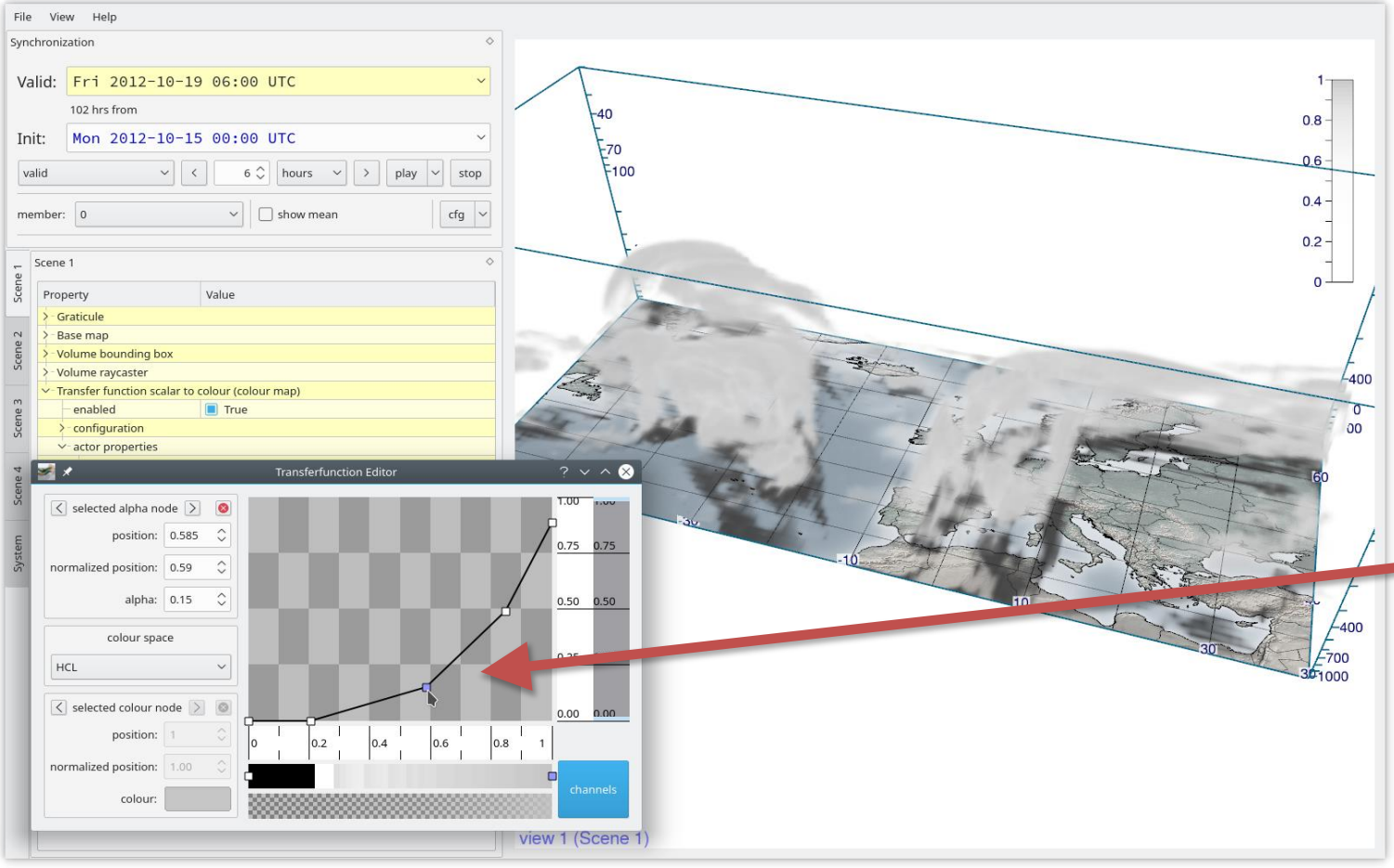
Direct volume rendering



6) Change the colour map's range to 0..1 (this is the range of cloud cover values).

7) Drag the left opacity node to 0.

Direct volume rendering



8) Adjust the opacity scale to adjust the "absorption model" and to change the display.

How to get data into Met.3D

Met.3D can read data in NetCDF that comply with the CF-conventions and that

- are defined on a regular or rotated lon-lat grid, or a regular projected grid in the horizontal

- are defined on levels of constant pressure or hybrid sigma-pressure model levels, or that provide 3D pressure fields

GRIB files from ECMWF can be read.
Trajectory data needs to be in NetCDF.

More information:
https://collaboration.cen.uni-hamburg.de/display/Met3D/Data+Handling

```
netcdf somegriddeddata.pl {
dimensions:
    lon = 101 ;
    lat = 41 ;
    isobaric = 12 ;
    time = 1 ;
    ens0 = 51 ;
variables:
    float lat(lat) ;
            lat:units = "degrees_north" ;
    float lon(lon) ;
            lon:units = "degrees_east" ;
    float isobaric(isobaric) ;
            isobaric:units = "hPa" ;
            isobaric:long_name = "Isobaric surface" ;
            isobaric:positive = "down" ;
    int time(time) ;
            time:units = "Hour since 2012-10-15T00:00:00.000Z" ;
            time:standard_name = "time" ;
    int ens0(ens0) ;
            ens0:standard_name = "ensemble_member_id" ;

    float Geopotential_isobaric(time, ens0, isobaric, lat, lon) ;
            Geopotential_isobaric:long_name = "Geopotential @ Isobaric surface" ;
            Geopotential_isobaric:units = "m2.s-2" ;
    float Temperature_isobaric(time, ens0, isobaric, lat, lon) ;
            Temperature_isobaric:long_name = "Temperature @ Isobaric surface" ;
            Temperature_isobaric:units = "K" ;

    ...
}
```

Thanks for your interest in using Met.3D!

Met.3D is a project with limited resources, hence bugs and issues become known but often cannot be fixed immediately. Look at the online issue tracker: https://gitlab.com/wxmetvis/met.3d/issues

We are working on improving the software! Please help us do so:

- Talk to us when you use Met.3D.
- Talk to us when you run into any kind of problems with the software.
- Talk to us when you have any special ideas/requests/...
- Let us know when you would like to use a Met.3D image for any presentation or publication. In particular in the beginning we may be able to help improve it! Also, please acknowledge Met.3D as the image source – acknowledgements will help us acquire further funding.
- Report bugs that you discover.
- If you want to contribute, that would be great, too!

If you are interested in using some of the "prototype-level-code" (jets, fronts, clustering, ..) talk to us as well!