

Mayavi

Intro to Mayavi

- Mayavi allows for plotting of three-dimensional data (i.e. $x, y, z + \text{value}$)
- Provides (<http://docs.enthought.com/mayavi/mayavi/overview.html#introduction>):
 - Visualization of scalar, vector and tensor data in 2 and 3 dimensions.
 - Easy scriptability using Python.
 - Easy extendibility via custom sources, modules, and data filters.
 - Reading several file formats: VTK (legacy and XML), PLOT3D, etc.
 - Saving of visualizations.
 - Saving rendered visualization in a variety of image formats.
 - Convenient functionality for rapid scientific plotting via mlab (see mlab: Python scripting for 3D plotting).
- Installing:
 - If you are not using one of the python distributions, you may need to install it manually
 - Fedora: `yum install Mayavi`
 - Note: you may need to enable the updates-testing repo to get the latest version because of a change to VTK

Starting Python for Mayavi

- Mayavi is interactive
 - See http://docs.enthought.com/mayavi/mayavi/mlab_running_scripts.html#running-mlab-scripts
- Interactive use:
 - `ipython --gui=wx --pylab=wx`
 - This will use a different output window to render the results
- In scripts:
 - Use `mlab.show()` to show the output window and allow for interaction

Interactivity

- From http://docs.entthought.com/mayavi/mayavi/mlab_case_studies.html:

...it is important to remember that Mayavi is an interactive program, and that the properties of these objects can be modified interactively, as described in Changing object properties interactively. It is often impossible to choose the best parameters for a visualization before hand. Colors, contour values, colormap, view angle, etc... should be chosen interactively. If reproducibles are required, the chosen values can be added in the original script.

Moreover, the mlab functions expose only a small fraction of the possibilities of the visualization objects. The dialogs expose more of these functionalities, that are entirely controlled by the attributes of the objects returned by the mlab functions. These objects are very rich, as they are built from VTK objects. It can be hard to find the right attribute to modify when exploring them, or in the VTK documentation, thus the easiest way is to modify them interactively using the pipeline view dialog and use the record feature to find out the corresponding lines of code. See Organisation of Mayavi visualizations: the pipeline to understand better the link between the lines of code generated by the record feature and mlab. .

Some Basics

- Mayavi is a big library. We'll look at some simple examples that show some basics.
- See the docs for more examples/detail:
 - <http://docs.enthought.com/mayavi/mayavi/mlab.html>