Plotting in Gnuplot and Vislt

Stephen Millmore

Laboratory for Scientific Computing, University of Cambridge

May 13, 2019

<ロ> <同> <同> < 同> < 同>

Laboratory for Scientific Computing

э



Outline

Gnuplot

Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

Laboratory for Scientific

Computing

2 / 155

O Python scripting in Vislt



What should you use to make a plot?

Whatever is easiest!

• Gnuplot - very good for 1D, good for simple 2D set ups, not recommended for 3D

Laboratory for Scientific

Computing

- Vislt not recommended for 1D, very good for 2D and 3D, especially multiple materials and AMR
- These two will be covered, but other options exist:
- Inbuilt plotting tools, e.g. those in Matlab or Mathematica (or Excel!)
- Paraview (coming soon?)
- Python, e.g. pyplot
- By hand?



Outline

Gnuplot

Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

Laboratory for Scientific Computing

3 / 155

3 Python scripting in Vislt



Advantages of gnuplot

- It's free
- Plot and axis appearance are easily customisable
- 1D plots are very clear and easy to manipulate
- 2D plots work well, either as a surface rendering or a colour map image harder to use if you don't want to plot the entire domain

Laboratory for Scientific

Computing

- In-built data analysing tools (e.g. maxima and minima)
- Can either plot through command line, or from a file



Gnuplot via terminal

G N U P L O T Version 5.0 patchlevel 3 last modified 2016-02-21 Copyright (C) 1986-1993, 1998, 2004, 2007-2016 Thomas Williams, Colin Kelley and many others gnuplot home: http://www.gnuplot.info faq, bugs, etc: type "help FAQ" immediate help: type "help" (plot window: hit 'h') ferminal type set to 'qt' jnuplot>

- > gnuplot
- Loads up gnuplot within a terminal, ready for commands
- Typing help will bring up an overall description of gnuplot, and a list of further topics for which help is available

5 / 155

Laboratory for Scientific

Computing

• In general, gnuplot help is well written with useful examples



Plotting commands and shortcuts

- There tend to be two forms of each gnuplot command, long (verbose) and short (shorthand)
- gnuplot> plot 'testData.dat' with linespoints linewidth 2, 'testData.dat' using 1:3 with lines linetype 4, f(x) title 'line'

۲

gnuplot> p 'testData.dat' w lp lw 2, '' u 1:3 w l lt 4, f(x) t 'line'

The two examples above do exactly the same thing





Laboratory for Scientific 6 / 155 Computing

Basic commands



Laboratory for Scientific

Computing



Ordering of commands

- In general, commands must be in the correct order or an error is given
- The general format is: plot <range> <file or function> <what is plotted> <style commands>
- For example:

gnuplot> p [1:50] 'testData.dat' every 2 u 1:3 lw 2 t 'penguin' w lp

イロン 不同 とくほう イヨン

8 / 155

Laboratory for Scientific

Computing



Using the plot window

- This assumes you've loaded gnuplot with the default 'qt' terminal other operating systems (e.g. Macs) might differ
- The terminal window is interactive, usually used just for zooming in and out, but may wish to do other things
- Zoom: +/- buttons at the top or +/keys ('=' acts as '+' too) or right click and drag a region you want to zoom in on
- Reset view to default: magnifying glass button with '1', or 'a'
- Toggle grid: grid button or 'g'
- Other interactive options include replot (if plotfile is still being output) and window save options





Laboratory for Scientific 9 / 155 Computing

Other plot window commands

- Not always useful, but sometimes hit by accident, and can be undone by pressing the same button!
- (1'/2') alter format of coordinate information in the bottom left of the window
- '6' echoes button commands to terminal screen (as gnuplot commands)
- '7' alter size ratio of the screen
- 'q' quit window
- 'r' toggle ruler (a cross-hair that appears at the cursor location)
- 'h' help window for these commands
- 'l' toggle logarithmic y-axis
- 'L' toggle logarithmic <axis closest to the cursor>
- 'm' toggle mouse interactivity
- arrow keys move plot window in direction of arrow





Other commands than plot

gnuplot> help <topic>

gnuplot> h <topic>

Brings up the help pages for a specific topic (cannot necessarily use shortcut for the topic name)

gnuplot> splot

gnuplot> sp

surface plot (2D plot command)

gnuplot> replot

gnuplot> rep

replot last plot or splot command

gnuplot> load <file>

gnuplot> l <f<u>ile></u>

load a file (should contain gnuplot commands)

- gnuplot> print <arg> gnuplot> pr <arg> print output of argument (e.g. function evaluation)
- gnuplot> set <arg> gnuplot> se <arg>

set environment variable

gnuplot> unset <arg> gnuplot> unse <arg>

stop an environment variable from being shown

gnuplot> <var> = <value> define a variable, or a function



Laboratory for Scientific Computing

3



Defining constants is straightforward, and gnuplot has basic maths commands

```
qnuplot> pi = 4*atan(1)
gnuplot> pr pi
 .14159265358979
```

- Note pi is defined already, but can be overwritten
- Take care when defining constants through integers

```
gnuplot> ratio = pi/3
qnuplot> pr ratio
1.0471975511966
qnuplot> ratio = 7/3
gnuplot> pr ratio
2
gnuplot> ratio = 7./3.
gnuplot> pr ratio
2.3333333333333333333
```



3

 $\frac{12}{155}$

Functions

- Functions are defined intuitively, and can be functions of multiple variables
- When plotting, however, the argument must be 'x' in 1D, 'x,y' in 2D





Plotting from files

• Plotting from multiple files is straightforward

gnuplot> p 'testData.dat' w lp, '' u 1:3 w lp, 'testData2.dat' w lp

- The shortcut '' specifies the same file as the previous command is to be used
- Files that gnuplot can read are moderately flexible, this is 'testData.dat'

# So	can	speed	(mm/s),	Simulation	data,	measured	data
24,	0.0	00205	2617	0.000201	158		
31,	0.0	000194	7424	0.000195	516		
I sl	houl	ldn't	be typing	g here			
47,	0.0	000168	558	0.000180	904		
94,	0.0	000147	3605	0.000157	702		
188	, 0.	.00014	21181	0.000123	312		

- The **u** or **using** command selects which columns of data are plotted not specifying is equivalent to **u 1:2** the first column does not have to be the *x* value, e.g. **u 3:1**
- Columns can be separated by spaces, tabs or commas (at least), '#' comments a line in a manner that gnuplot may be able to recognise, garbage lines are ignored entirely



Indexed file format

• It is possible to have multiple data quantities within a single file, e.g. output from multiple time steps

```
# Scan speed (mm/s), Simulation data, measured data
24. 0.0002052617
                         0.00020158
31, 0.0001947424
                        0.00019516
47. 0.000168558
                        0.00018004
94, 0.0001473605
                        0.00015702
188, 0.0001421181
                        0.00012312
# Scan speed (mm/s), Simulation data, measured data
        0.0001948399
31
                         0.0002405
41
        0.0001868327
                         0.0002281
62
        0.0001574733
                         0.0002080
124
        0.0001521352
                        0.0001780
248
        0.0001467972
                         0.0001400
```

- In order for gnuplot to distinguish these, each separate entry must be separated by *at least* two lines of blank space
- The index or i option specifies which will be used (starting with 0)

gnuplot> <u>p</u> 'testData3.dat' u 1:2 i 1 w lp, '' u 1:3 index 0 w lp



イロト イポト イヨト イヨト

2D data format

٩	2D data should (ideally) be set out as <x coords=""> <y coords=""> <data at="" x,y=""></data></y></x>	1 1 1 1	1 2 3 4 5	0.1 0.2 0.3 0.4 0.5	
•	For rectangular grids, the format shown should be used (or inverse for x and y)	2 2 2 2 2	1 2 3	0.3	
٩	After each y -sweep, a blank line must be left to allow for a grid to be plotted	2 3 3	5 1 2	0.0 0.7 0.7 0.8	

1 Line





Ξ.

16 / 155



UNIVERSITY OF CAMBRIDGE

Laboratory for Scientific Computing

< □ > < □ > < □

plot <range> <file or function> <what is plotted> <style commands>

- One way to control the range of the plot is to specify it after the plot or splot command
 - gnuplot> p [0:2*pi] f(x), g(x), h(x,1)
 gnuplot> p [0:2*pi][0:1] f(x), g(x), h(x,1)
 gnuplot> p [][0:1] f(x), g(x), h(x,1)
 gnuplot> sp [1:3][1:3][0:1] 'testData4.dat' w lp
- A single entry will always adjust the x-axis range, but all axes can be controlled (or ignored)
- This will control the ranges of *all* subsequent plots, attempting to specify different ranges later will result in a gnuplot warning

イロト 不得 とくほ とくほ とうほう

17

Laboratory for Scientific

Computing



plot <range> <file or function> <what is plotted> <style commands>

- These are ways to use the data from a file, beyond plotting just the raw numbers
- Here we consider index, using and every
- index has already been described there is little more to it
- using or u picks columns, but we can also operate on these columns

gnuplot> p 'testData.dat' u (1e-3*\$1):2 w lp gnuplot> p 'testData.dat' u (1e-3*\$1):(\$3-\$2) w lp

• every or ev selects points to plot, see help for full 2D file options

gnuplot> p 'testData.dat' u 1:2 ev 2 w lp
gnuplot> p 'testData.dat' u 1:2 ev 2::2 w lp



Laboratory for Scientific Computing

• One additional note for the using command - the ternary '?:' operator is supported

p 'testData.dat' u 1:(\$1 > 50 ? sin(\$2) : 1/0) w lp

• Additionally, attempting to plot 1/0 is rendered empty by gnuplot, e.g. this can be used to plot data for a specific sign of a level set function

・ロト ・回ト ・ヨト ・ヨト

Laboratory for Scientific

Computing

э

19



Style commands

UNIVERSITYOE

CAMBRIDGE

plot <range> <file or function> <what is plotted> <style commands>

- We consider three different style commands, with (and associated styles), axis and title
- with or w controls the look of the data plotted
- axis or ax controls the axis data is plotted on for 1D plots, there are four possibilities, x1, x2, y1, y2, useful for showing values of two very different variables

gnuplot> p 'testData.dat' u 1:2 axis x1y1 w lp, 'testData2.dat' u 1:2 axis x2y2 w lp gnuplot> <u>p</u> 'testData.dat' u 1:2 axis x1y1 w lp, 'testData2.dat' u 1:2 ax x2y1 w lp

• title or t sets the title of the data within the legend

gnuplot> p 'testData.dat' u 1:2 w lp title 'Good data' gnuplot> p 'testData.dat' u 1:2 w lp t 'Good data' gnuplot> p 'testData.dat' u 1:2 w lp notitle

notitle means the plot will not appear in the legend

Laboratory for Scientific Computing

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □



Style commands - with

• gnuplot> help with gives a list of potential styles

Synt	ax: with ≺style	> { {\lins {() { 	estyle ls linetype linewidth linecolor pointtype pointsize Fill fs - pohidden3d] poalette}}	<pre>s <line_style>} lt <line_type> lt <line_type> lw <line_width <colorspec:="" <point_size="" <point_type="" fillstyle="" lc="" ps="" pt="">} {nocontours} + </line_width></line_type></line_type></line_style></pre>	-) >} > > >> nosurface}	
wher Press	e <style></style>					

- Not all are applicable to all data (e.g. error bars needs error data as columns)
- Refer to gnuplot> test to see the styles and colours



۲

gnuplot> set <variable> allows various aspects of the plot to be altered

gnuplot> help set reveals the quantity of options available (we shall not detail ٥ them all!)

Subtopics available f	for set:		
angles	аггом	autoscale	bars
bmargin	border	boxwidth	cbdata
cbdtics	cblabel	cbntics	cbrange
cbtics	clabel	clip	cntrlabel
cntrparam	color	colorbox	colorsequence
contour	dashtype	data	datafile
date_specifiers	decimalsign	dgrid3d	dunny
encoding	fit	fontpath	format
function	grid	hidden3d	history
Press return for more			
historysize	isosamples	key	label
linetype	link	lmargin	loadpath
locale	log	logscale	macros
mapping	margin	margins	missing
monochrome	mouse	multiplot	mx2tics
mxtics	my2tics	mytics	nztics
object	offsets	origin	output
palette	parametric	paxis	pm3d
pointintervalbox	pointsize	polar	print
psdir	raxis	rmargin	rrange
rtics	samples	size	style
surface	table	term	terminal
termoption	tics	ticscale	ticslevel
time_specifiers	timefmt	timestamp	title
tmargin	trange	urange	view
vrange	x2data	x2dtics	x2label
x2mtics	x2range	x2tics	x2zeroaxis
xdata	xdtics	xlabel	xmtics
xrange	xtics	xyplane	xzeroaxis
y2data	y2dtics	y2label	y2mtics
y2range	y2tics	y2zeroaxis	ydata
ydtics	ylabel	ymtics	yrange
Press return for more			
ytics	yzeroaxis	zdata	zdtics
zero	zeroaxis	zlabel	zmtics
zrange	ztics	zzeroaxis	





Axis manipulation

• The range of data plotted can be adjusted through e.g. set xrange or set xr



By default, axes are not labelled, labels are set through e.g. set xlabel or set xl

> <u>qnupl</u>ot> set xlabel 'Height ({/Symbol m}m)' onuplot> set vlabel 'Pressure (atm)' qnuplot> set y2label 'Density (kg/m^3)'

• Control of the numbered tics on the axis is through e.g. set xtics

anuplot> set v2tics gnuplot> set ytics nomirror qnuplot> set xtics (24, 31, 47, 94, 188)



Laboratory for Scientific Computing

3

23 155

• It can be useful to give a plot a title, this is done through set title

gnuplot> set title 'Plot of a Penguin'

• It is also possible to draw arrows, or lines, on a plot, through set arrow - this can be useful for marking the location of a feature, e.g. a discontinuity

gnuplot> set arrow from 0.0,0.5 to 100,0.5 gnuplot> set arrow from 50,0.5 to 100,0 nohead

・ロン ・回 と ・ ヨ と ・ ヨ と …

24 155

Laboratory for Scientific

Computing



- There are three typical ways to plot 2D data through splot :
- 1) The default, height-mapped grid
- 2) Contours
- 3) Colour map
- For the second two options, it may be desirable to remove the grid entirely, and ensure a top-down view

イロン イロン イヨン イヨン

Laboratory for Scientific

Computing

э

25 155

 This is achieved through gnuplot> set view map gnuplot> unset surface





• Contours are turned on through gnuplot> set contour or

gnuplot> set cont

- By default, contours are placed at the base of the three dimensional box containing the surface grid
- gnuplot> set contour surface or gnuplot> set contour both can change this
- Default contour levels are chosen by gnuplot, control over these levels is through gnuplot> set cntrparam <options>

gnuplot> set cntrparam levels discrete 0,1,2,3,4 gnuplot> set cntrparam levels incremental 0,1,4

Laboratory for Scientific

Computing

3

26 155

• The contour style (line width etc.) is controlled through the plot style - it is not straightforward to independently control the colour of the contours



Palette-mapped 3d (pm3d)

- gnuplot> set pm3d allows a surface to be plotted as a 2D colour mapped image
- By default, the image will be plotted on the bottom 'surface' of the three dimensional box containing the surface grid





Laboratory for Scientific Computing

27 155

Altering the pm3d style

• The colour range is set through gnuplot> set cbrange [<low>:<high>] - this is independent of the zrange setting

(日)

28 155

Laboratory for Scientific

Computing

- The colour scheme is changed through gnuplot> set palette <options>
- Options here are numerous, recommended examples are at: http://gnuplot.sourceforge.net/demo_5.2/pm3dcolors.html
- The location of the colour box is altered through gnuplot> set colorbox <options>



Output to file

- Outputting a plot to a file requires two things; setting the output name (obviously) and setting the correct terminal for the output
- To set an output, simply use e.g. gnuplot> set output 'Filename.png'
- In order to generate the file in the desired format, the terminal type needs to be changed through gnuplot> set terminal <options> or

gnuplot> set term <options>

Subtopics available for set term:							
cairolatex	canvas	cgm	context				
corel	dumb	dxf	eepic				
emf	emtex	epscairo	epslatex				
fig	gif	hpgl	jpeg				
latex	lua	nf	mp				
pcl5	pdfcairo	png	pngcairo				
рор	postscript	pslatex	pstex				
pstricks	push	qms	qt				
size	svg	tek40xx	tek410x				
texdraw	tgif	tikz	tkcanvas				
tpic	vttek	wxt	×11				
Press return for m	ore:						
xlib	xterm						



29 155

Terminal options

- Some of these terminal options will open a new window when selected, e.g. 'qt' or 'x11', others will generate a file, e.g. 'pngcairo' or 'postscript'
- Selecting the terminal also allows for size and font choices for the overall plot to be made

gnuplot> set term postscript enhanced color

gnuplot> set terminal pngcairo size 700,500 enhanced font 'Verdana,20'

30 155

Laboratory for Scientific

Computing

gnuplot> set term postscript "Helvetica" 12 enhanced color portrait

- The enhanced mode is what allows for text formatting, e.g. superscripts
- Practical note when outputting to files, the first plot will create the file, whilst for a second plot, gnuplot will attempt to append this to the file (which may not make sense)
- This can be avoided if the set output command is used again (even to the same filename)



Multiple plots in one window

- Gnuplot allows this through gnuplot> set multiplot, this is likely to be used with output to a file, rather than the default terminal
- Once multiplot mode is on, the size and origin of each plot must be chosen, e.g. gnuplot> set size 0.4 0.2
 gnuplot> set origin 0.1 0.1
- The bottom-left corner is (0,0) and the top-right corner is (1,1) always (regardless of the dimensions of the window)
- Some care may be needed if the width of the axis numbers do not match the size defined may be the entire box containing the plot

・ロト ・回ト ・ヨト ・ヨト

31 155

Laboratory for Scientific

Computing



- Log scaling of the axes can be done through gnuplot> set logscale <axes>
- The log scaling can be specified, if something other than base 10 (default) is required
- When plotting functions, it may be that they are not well resolved (e.g. a highly oscillatory function), this is due to gnuplot not sampling the function frequently enough
- If necessary, gnuplot> set samples <number> will force gnuplot to take a specified number of samples
- For 2D surfaces, gnuplot> set isosamples <number> achieves this (note a lot of sampling will lead to long plot times)



32

- All commands shown so far can be entered through the command line, though they can also be entered through a gnuplot script
- These are useful for saving plot commands, either to avoid forgetting, or to use upon restarting or on new files
- Scripts can either a complete process (loading, setting output file and closing), or partial process (e.g. setting constants and environment variables)

イロト 不得 とくほ とくほ とうほう

33 155

Laboratory for Scientific

Computing

- A complete process can be run through:
 - > gnuplot PlotFile.gp
- Whist a partial process is loaded within a gnuplot instance: gnuplot> load PlotCommands.gp



Script example 1 - how to split lines

set term postscript enhanced color set output '100 z24 sqrtlaw.eps'

set format x "%3.1e" set format y "%3.1e"

set xtics 6e-4,2e-4,2e-3

f(x) = 0.102 * x + 1.1e-05g(x) = 0.0262 * x - 2.34e-5

set key outside right

set xlabel 'sqrt(time)'
set ylabel 'radius'

```
p'crossingAadius_24e-o.dut' u (sqrt($1)):3 w lp t'1000, 0!evels', \
/crossingAadius_24e-o.dut' u (sqrt($1)):3 w lp t'400, 1!evels', \
/crossingAadius_24e-o.dut' u (sqrt($1)):3 w lp t '800, 0!evels', \
/crossingAadius_24e-o.dut' u (sqrt($1)):3 w lp t '800, 1!evels', \
/crossingAadius_24e-o.dut' u (sqrt($1)):3 w lp t'400, 0!evels', \
/f(x) w l 10 lw 6 t'0:02x'
```

set xlabel 'time^{0.38}'
set xtics 2e-3,10e-4,10e-3

set output '100_z24_38law.eps'



Laboratory for Scientific Computing

э

34

(日)

Script example 2 - do loops

```
set terminal pngcairo size 1400,260 enhanced font 'Verdana,14'
set pm3d
unset surface
set view map
set size ratio -1
set cbrange [0:5e6]
unset kev
set origin -0.04, -0.01
set xrange [0:0.4]
set yrange [0:0.05]
set xtics out nomirror
set vtics out nomirror
unset x2tics
unset y2tics
set output 'DynamicVillaPressure 0.png'
sp 'DynamicVillaARP 2D test 2D 0.dat' u 1:2:7 w lp
do for [t=1:97] {
  x = 20.*cos(2*pi*2e6*t*1e-7)
  set arrow from 0.0.0.5 to x.0.5
  set output 'DynamicVillaPressure '.t.'.png'
  sp 'DynamicVillaARP 2D test 2D '.t.'.dat' u 1:2:7 w lp
```



Laboratory for Scientific Computing

э

35 155
Outline

Gnuplot

Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

Laboratory for Scientific

Computing

35 / 155

Python scripting in Vislt



Advantages and disadvantages of Vislt

Advantages

- It's free
- It is created by a group with a multiphysics AMR code, and as a result (LLNL), has many very useful features for our multiphysics AMR code
- 2D and 3D plots are handled well, and there is a lot of flexibility as to what is plotted

イロト イポト イヨト イヨト

Laboratory for Scientific

Computing

36

Disadvantages

- It is not the most stable software
- Can be more work to compile and generate output for



Data files for Vislt

- Unlike gnuplot, Vislt cannot read tabulated files it expects more information about what is in the file
- However, the list of other file formats that can be used is huge! https://www.visitusers.org/index.php?title= Detailed_list_of_file_formats_VisIt_supports
- General output is beyond the scope of this lecture
- Large codes (group code, AMReX) should already output Vislt readable files, test codes only need limited functionality (VTK)
- Standard output formats, such as HDF5, allow Vislt to deal with data stored on patches, in different materials, as well as allowing for parallel input/output



VTK output

- One of the simplest formats of Vislt-friendly output is VTK (visualization toolkit)
- In many ways, a tabulated data structure with additional header information
- Test codes will need to output this information correctly https://vtk.org/wp-content/uploads/2015/04/file-formats.pdf



Part 1: Header	Part 4: Geometry/topology. Typ		
Part 2: Title (256 characters maximum, termi- tated with newline \n character)	STRUCTURED_POINTS STRUCTURED_GRID UNSTRUCTURED_GRID		
Part 3: Data type, either ASCII or BINARY	POLYDATA RECTILINEAR_GRID FIELD		
	11110		

Part 5: Dataset attributes. The number of data items *n* of each type must match the number of points or cells in the dataset. (If type is FIELD, point and cell data should be omitted.

Type is one of:



Laboratory for Scientific Computing

38

(日) (同) (三) (三)

Example vtk script

Much of the work we do uses the RECTILINEAR_GRID output type, hence we consider this for an example

```
write(1001, '(a26)') "# vtk DataFile Version 3.0"
write(1001, '(a26)') "Data produced by mf evolve"
write(1001, '(a5)') "ASCII"
write(1001, '(a24)') "DATASET RECTILINEAR_GRID"
write(1001. '(a10.3i6)') "DIMENSIONS". &
     npx, npv, 1
write(1001, '(a13,i6,a6)') "X COORDINATES", npx, " FLOAT"
do i = 1, npx
  write(1001, '(es21.12E3, a1)', advance = 'NO') q(nnt, i, 1, 1), ' '
end do
write(1001. *)
write(1001, '(a13,16,a6)') "Y COORDINATES", npv, " FLOAT"
do i = 1, npy
  write(1001, '(es21.12E3, a1)', advance = 'NO') q(nnt, 1, i, 2), ' '
end do
write(1001. *)
write(1001, '(a13,i6,a6)') "Z_COORDINATES", 1
                                                     . " FLOAT"
write(1001, '(11)') 0
write(1001, '(a10,i12)') "POINT DATA", npx * npy
do k = nv1, nv2
  write(1001, *)
  out string = "SCALARS "//base name
  write(n_string, '(i1,a6)') k, " FLOAT"
  write(tmp_string,'(i3)') len_trim(out_string) + len_trim(n_string)
  format string = "(a"//adjustl(trim(tmp string))//")"
  write(1001, format string) adjustl(trim(out string))//adjustl(trim(n string))
  write(1001, '(a20)') "LOOKUP_TABLE default"
  do i = 1, npv
   do i = 1. npx
       if (abs(g(nnt, i, j, k+2)) .le. 1.d-16) then
          write(1001, '(es21.12E3)') 0.d0
          write(1001, '(es21.12E3)') q(nnt, i, j, k+2)
      end if
    end do
  end do
end do
```

close(1001)



- First information about the data is inserted into the file
- Then the *x*-coordinates, *y*-coordinates and, if applicable, *z*-coordinates
- Then the data, if you have multiple variables, these follow one after another
- Each variable is given a name and a data type
- The data points are then output in the specific order (z-loop then y-loop the x-loop)

Laboratory for Scientific Computing

Opening Vislt

- When opening Vislt, two windows (hopefully) open
- The window on the left controls all aspects of the plots you make, we refer to this as the 'Vislt window', or the 'main window'
- The window on the right contains the plot, you can have multiple of these







- **Open** and **Close** self explanatory, though closing a file requires *all* plots from it to be deleted
- Vislt can open a file series, hence you can **Reopen** once you have more output will cause errors if file is being written (or Vislt is feeling grumpy)
- Multiple files can be opened as a database, by default Vislt groups numerical files of the same file type automatically
- Active source shows the files/databases you currently have open, and you select which one you want to plot

41 155

Laboratory for Scientific

Computing







- Plots appear in the order they were created in the main window
- This order determines what appears 'on top' - the most recent plot will always on top
- Sometimes, Vislt is a bit cleverer e.g. contours will always appear on top of colour plots in 2D
- **Replace** will replace a highlighted plot with the current **Active source**, maintaining all formatting
- Overlay will duplicate the current highlighted plot, but use the current Active source instead
- Note multiple plots can be selected (hold shift or control)



Laboratory for Scientific Computing



- Add a new plot
- Add Operators to an existing plot, altering how it looks
- Delete a plot entirely, or Hide/Show to toggle its visibility
- Draw a plot, once it has been added, operated upon, or changed significantly
- Variables list, from all the variables available in your output file
- By default, Vislt often sets its main window size such that the **Variables** menu is hidden (it can be accessed through the '\lapsilon' button)

<ロ> <同> <同> < 回> < 回>

Laboratory for Scientific

Computing

э



Apply to	🔘 active window	۲	all windows	
🛃 Apply operators to all plots				
🞯 Apply subset selecti	ons to all plots			

- These are three options which select how you modify plots
- Operators are added through the **Operators** menu, subset selections are available once you've made a plot (covered later)
- Vislt can have multiple plots open, and visible, at once, and you may wish to alter them all in the same way (e.g. revolve)
- These options are on by default (though not on an LSC machine), but are better turned off (I think)

・ロト ・回ト ・ヨト ・ヨト

44 155

Laboratory for Scientific

Computing

• You can also alter operators across all windows open (off by default)



At the top of the plot window are a series of buttons

: □ □ □ □ □ □ □ □ □ □ □ □ □ □ ■ ■ # ¥ ¥ ∅ ∅ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ₩ ↓ ↓ ↓ □ > > > = 3 ≥ 2 ≥ ⊕ ≪ +z +, +, +; ⊭ <u>≪</u> ∕ ∦ ₩ ♥ ♥ !!!

<ロ> <同> <同> < 回> < 回>

Laboratory for Scientific

Computing

- Numbered by position from the left
- 1 Activate window, for when you have multiple windows open
- 2 New (blank) window
- 3 Clone window (and all plots on it)
- 4 Delete window (also achieved by the 'x' button)
- 14 Reset view (when you've zoomed in, and want to return)
- 21-25 Change frames also in the main viewer



At the top of the plot window are a series of buttons

- Numbered by position from the left
- 1 Navigate mode click and drag the plot, slides the plot in 2D, revolves in 3D (holding ctrl will translate the plot in 3D)
- 2 Zoom mode use mouse to draw a zoom region
- 3/4/5 Zone/Node/Spreadsheet pick Click on a point, and cell centred, node centred or spreadsheeted data will be given about it
- 6 Lineout 2D only, click and drag a line and a 1D slice along that line will be plotted in a new window

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □

46

Laboratory for Scientific

Computing



Saving the window

Default (ctrl-s) might not save where you want it to, nor with a useful name (default filename is 'visit')

😑 🗇 🗇 Set so	we options								
Filename									
Filename pen	guin			Family					
e Output file	Output files to current directory								
Output directory									
/home/stevem	illmore/lsc_ho	me/TexStuff/N	otes/VisltG	uide/Figs					
Format options									
File type	png			•					
Quality		-0-	Progre	ssive					
Compression to	vpe PackBits								
de Binary	Stereo								
te									
Aspect ratio and	resolution								
Aspect fatto s	creen ratio	•							
Width 1	1024		t 1024						
Screen cap	ture								
e 🗌 Multi-windo	w save								
Tiled		Advanced							
Window 1			Omit v	window					
Width 1	28	Height	128						
Position (X)		Position (Y)	0						
Layer 1		Transparency		0%					
Save	ave and Dismi	55							
Apply			Post	Dismiss					

UNIVERSITY OF

CAMBRIDGE

- $\bullet \ \ \mathsf{File} \to \mathsf{Set} \ \mathsf{save} \ \mathsf{options}$
- Family will consecutively number each plot (default on)
- Default is to **Output files to current directory**, '...' button lets you choose directory
- Various formatting options, traditional graphics format, but also 'curve' - can be used to save 1D output, and be read by gnuplot
- Screen capture save picture at current screen resolution (often suitable for presentations) - buggy in some older versions

47 155

Laboratory for Scientific

Computing

Saving movies (in theory)

	😣 🗈 Save movie wizard
	Choose format Choose movie formats and resolutions.
Movie type	Format and resolution Output
Would you like to create a new simple movie or use your previous	Format MPEG movie
settings? Use my previous movie settings New simple movie New template movie	Width 1007 is lock aspect Width 1007 is lock aspect Scale S
< <u>Back</u> <u>Next</u> Cancel	d Othe Screen Capture (Back Next > Cancel

• This can also be used to save some or all of the images in the current database



э

Saving movies (in theory)

😣 🗈 Save movie	mijard			😣 🗈 Save movie	e roijatð
Choose length Choose movie start	/end time and frames per second.		•	Choose filename Choose the outpu	t directory and base filename for your movie(s).
Frames per second	10)	l	Output directory	/home/stevemillmore/
First frame	0			Base filename	movie
Last frame	153		Ľ.		
Frame stride	1		10		
Initial frame value	0		п		
ib.			(Ab		
d			d		
ıb			ıb		
d			d		
1È			1p		
d			d		
· · · · · · · · · · · · · · · · · · ·			н		
	< <u>Back</u> Cancel				< <u>Back</u> <u>Next</u> Cancel

 Frames per second only does something if you successfully make a movie, not images

<ロ> <同> <同> < 同> < 同> < 同> - < 同> - < 同> - < 同 > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - < □ > - <

Laboratory for Scientific Computing

э



Saving movies (in theory)

g 🗈 Sabe mobie wizard	8 @ Sabe modie wijard	
E-mail notification Do you want to be notified by E-mail when your movie completes?	Choose method Choose when and how you would like Visit to create your movies.	
🔿 Yes 🕘 No	Now, use currently allocated processors	
	 Now, use a new instance of Vislt 	
e E-mail address	e 🔷 Later, tell me the command to run	
< Back Cancel	< <u>Back</u> <u>Finish</u> Cancel	

 Finish will bring up an additional window (visit terminal instance) running the movie script, and may result in a movie - if it fails, it will still give the files it saved



-

< ≣⇒

- Saving a session allows you to save the plots you've already made, including operators, so that you can work on them later
- Useful if making a complex plot, especially if Vislt keeps crashing on you
- Two restore options 'restore session' and 'restore session with sources'
- The first will reload exactly as you had the session before
- The second allows you to choose the sources you load e.g. if you want to re-do a plot using different data files, but to show the same thing



51 155

<ロ> <同> <同> < 回> < 回>

Other save options - export database

😣 🚍 🗉 Export Database				
Dutput				
Directory name evemillmore/AMReX/output/Test				
File name visit_er_db				
Export all time states Format: _%04d				
Export to Amdv	\$			
Variables				
Delimiter 💿 Space 🔿 Comma				
2100 Variable 🔻 B_3				
I/D Options Coordinate parallel writes with groups.				
Write group size 48	× v			
Export				
Apply Post Dismis	5			

- Exporting a database allows you to save the data under a different format
- Some of the formats are also available from the save window, but in this case, you can append the entire, time-varying database to the file
- Others formats need to be exported this way doing this, it is possible to get Vislt output into gnuplot readable files (though not necessarily optimally output)

Laboratory for Scientific

Computing

52 155

• Note - not all output forms can deal with AMR, multiple materials very well



Outline

Gnuplot

2 Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

Laboratory for Scientific Computing

52 / 155 /

3 Python scripting in Vislt



Plotting - Pseudocolor

The default colour map style plot - $\textbf{Add} \rightarrow \textbf{Pseudocolor} \rightarrow <\!\! \textbf{variable} \!\!>$



user: stevernillmore Thu Jul 26 16:55:13 2018



Laboratory for Scientific Computing

<ロ> <同> <同> < 回> < 回>

53 155

э

Plotting - Pseudocolor - modifying the plot

Orszag_Tang_vortex_10.hdf:Pseudocolor - IdealMHD_p

Clicking the triangle...



- Clicking the two circles will bring up the Subset menu
- Clicking Pseudocolor will bring up the Pseudocolor plot attributes menu

Laboratory for Scientific

Computing

54 155

• Any other operators on the plot will be listed when clicking the triangle



Plotting - Pseudocolor - attributes

😠 🔿 🗊 Pseudocolor plot attributes
Data Geometry
Data
Scale 🖲 Linear 🔿 Log 🔿 Skew 1
Limits Use Original Data 💲
Minimum 0 Maximum 1
Centering 🖲 Original 🔿 Nodal 🔿 Zonal
Color
Color table Invert
Opacity Fully opaque 2
Misc
Clighting
Make default Load Save Reset
Apply Post Dismiss

- Scale If using a log scale, Vislt will complain if any values are 0 or negative
- Limits Change when making a series of plots, for consistency, or when interested in data at a particular value
- Centering see next slide
- **Color** Various colour schemes are available, some hideous
- Misc Turning off the legend can be useful, e.g. multiple plots at the same scale, or creating aesthetically pleasing, but data-light plots

Laboratory for Scientific

Computing



Plotting - Pseudocolor - attributes - centering



• Zonal on the left, Nodal on the right - the group code defaults to zonal

Laboratory for Scientific Computing



Plotting - Pseudocolor - 3D

By default, a 3D pseudocolour plot shows an opaque box and the external data plotted





Laboratory for Scientific Computing

57 155

<ロ> <問> < 回> < 回> < 回>

Plotting - Pseudocolor - attributes

800 P	eudocolor plot attributes	
Data Geo	metry	
Data		
Scale	Sinear O Log O Stew 1	
Limits	Use Original Data	
🗆 Mir	timum 0 Maximum 1	
Centering	🖲 Driginal 🔿 Nodal 🔿 Zonal	
Color		
Color tabl	e hot Invert	
Opacity	Constant 🗧	
	1 A A	36%
Misc		
🗌 Legen	d 🗹 Lighting	
Mate defau	lt Load Save	Reset
Apply	Post	Dismiss

▲御▶ ▲ 理▶ ▲ 理▶

Laboratory for Scientific Computing

э

- Altering opacity can show behaviour within the domain
- If nodal centring is used, this will, however, show patch boundaries



Plotting - Pseudocolor - attributes

😣 🌐 🐵 Pseudocolor plot attributes		
Data Geometry		
Data		
Scale 🖲 Linear 🔿 Log 🔿 Skew 1		
Limits Use Original Data *		/
minimum	1	
Centering 🖲 Driginal 🔿 Nodal 🔿 Jonal		
Color		
Color table 📄 hot 🗌 Invert		
Opacity Ramp ‡	Í	
1	1 99%	
mise		
🗌 Legend 🥃 Lighting		
		[-3]
Make default Load	Save Reset	
Apply	Post Dismiss	

< □ > < 同 >

Laboratory for Scientific Computing

э

- Zonal centring avoids this
- This is an example of a ramped opacity gradient





• Vislt output can allow for **Materials**, i.e. quantities which each have their own set of variables

Laboratory for Scientific

Computing

60 155

By default, these exist everywhere in the domain, but are 0 where they don't actually exist





• Through the plot descriptions **Subset** menu (the two circles), materials can be turned on and off イロト イポト イヨト イヨト

э

61 / 155 /

Laboratory for Scientific

Computing





- Multiple plots, limited by subset, can be plotted
- The material boundaries are interpolated automatically by Vislt, but code output may have only a single value per cell
- This will result in plot artefacts when zooming in

Laboratory for Scientific

Computing





- Subset can also be used to show only a single refinement level
- It can even show just a single patch (use Pick to identify the patch in question if there are many)

Laboratory for Scientific

Computing

63 /



Plotting - Boundary

Show only boundaries between materials - Add \rightarrow Boundary \rightarrow

 boundary_variable>





Laboratory for Scientific Computing

э

64 / 155

Plotting - Boundary - attributes

😣 🖨 🗊 Boundo	iry plot attribut	es	
Boundary colors			
 Color table 	Default		
 Single 			100%
🖲 Multiple			100%
Boundaries	0 Aluminium 1 PMMA 2 Electrode 3 Air 4 Plasma	1	
Opacity	1	1 1	100%
Options Wireframe			
Point / Line Style			
Point type Poir	nt ෫	Point size (pixels)	2
Scale point si	ze by variable	default 💌	
Line style	— solid 🛟	Line width	-1 :
Geometry Smoothing	None	○ Fast	⊖ High
Misc Legend			
Make default	Lo	ad Save	Reset
Apply		Post	Dismiss

- **Boundary colors** Each boundary can be individually chosen, or a single colour, whatever is best for visibility
- **Point/Line Style** adjusting line width is often useful, to highlight boundaries (and hide material subset artefacts!) The style of the boundary line can also be altered.
- Misc the legend is often useless, and can be removed

Laboratory for Scientific

Computing



Plotting - Contour

Show contours of a variable - $\textbf{Add} \rightarrow \textbf{Contour} \rightarrow <\!\!\textbf{variable}\!>$





Laboratory for Scientific Computing

э

66 / 155 /

<ロ> <同> <同> < 回> < 回>

Plotting - Contour - attributes

😑 😑 🙂 Contor	ır plot attributes		
Contour Levels Scale 🔘 I	.inear 🔿 Log		
Minimum	0	Maximur	n 1
Select by N le	vels 💲 10		
Contour colors			
 Color table 		Default	Invert
○ Single			
Atultiala			100%
• Multiple			
Level Color		Opacity	h
1			100%
2			100%
3	-		100%
4			100%
5	-		100%
the state		1	0
Line style	- solid	Cline width	-2 :
Misc			
👿 Legend		U Wirefram	c
Make default		Load	Save Reset
Apply			Post Dismiss

- **Contour Levels** change the contour number, spacing and selection. 10 evenly spaced contours is default. Can **Select by**:
- N levels alter number of evenly (or log) spaced contours
- Value(s) fixed parameter values for the contours must be a space-separated list (e.g. '0 1 1.4 4')
- Percent(s) Fixed intervals of the variable range - again a space separated list without '%'
- Contour colours, Line style and Misc work as for boundary plots

< <p>Image: Image: Imag



Laboratory for Scientific Computing

Contour plots in 3D produce surfaces of constant value, unfortunately opacity of a colour scheme cannot be changed, only of individual colours





Laboratory for Scientific Computing

Plotting - Contour - attributes



• If contours meet a domain boundary, **Wireframe** can be used to show them only at the boundary

Laboratory for Scientific Computing

69 155

• Operators can generalise this, however


Plotting - Curve





- Generates 1D data from both 2D and 3D plots (equivalent to the **lineout** window button in 2D)
- Add \rightarrow Curve \rightarrow operators \rightarrow Lineout \rightarrow <variable>
- A curve plot has two ways to modify the data, **Lineout** and **Curve**
- Distance is always measured from the start of the line, regardless of start coordinate



Laboratory for Scientific Computing

э

70 155

<ロ> <同> <同> < 回> < 回>

Plotting - Curve - curve attributes

🙁 😑 🐵 Curve plot attribute	
Data Geometry Extras	
Color	
Curve color 💿 Cycle	🔿 Custom
3:11	
Fill mode	No Zill 2
Color 1	100%
Color 2	100%
Misc	
🗹 Legend	🗹 Labels
Mate default	Land Same Reset
Mppin	Pismiss Dismiss

line			
Show lines			
Line style 2i	ne width	-1 :	
Boint			
Ghow points			
Symbol Point 🗘 Po	int size	5	
Symbol (Point :) Po Gtatic Po	int size int stride	5	
Crymbol (Point ;) Po Otatic Po Oynamic Po	int size int stride int density	5 1 (50 (
Oymbol (Point) Po Otatic Po Dynamic Po Mate Default (Pood	int size int stride int density Mane	5 1 (50 (

- **Curve colour** by default, Vislt cycles through curve colours, happily plotting the near-invisible yellow this can be changed
- Fill solid colour fill beneath the plotted line
- In addition to the Legend, the Labels (letters along the plot) can be turned off
- Line line style (e.g. thickness) or turn off altogether
- **Point** show the actual data points, stride takes one point for every *n* cells passed through (regardless of the amount of cell passed through for diagonal lines)





Plotting - Curve - Lineout attributes

S 🖨 🗈 Lineout operator attributes		
Point 1 0.01217277486910989 0.9634816753926702 0		
Point 2 0.3708115183246073 0.04725130890052354 0		
🗌 Interactive		
🗌 Override Global Lineout Settings		
🔲 Use Sampling		
Samples 50		
🗌 Refline Labels		
Make default Load Save Reset Apply Post Dismiss		

- The start and end points of the line can be controlled
- By default, Vislt uses 50 Samples this may not be enough to capture sharp features

Laboratory for Scientific

Computing



Plotting - Mesh

Show the mesh - $Add \rightarrow Mesh \rightarrow < mesh_variable >$





・ロト ・回ト ・ヨト ・ヨト Laboratory for Scientific Computing

æ

Plotting - Mesh - attributes

🔒 🖨 🗊 Mesh plo	t attributes	
Zone Show internal z	ones	
Color Mesh color	Foreground	O Custom
Opaque color	Background	Custom
Opaque mode	🖲 Auto 🛛 🔿 On	⊖ Off
Opacity	1 1 1	100%
Point / Line Style Point type Point Scale point size Line style	Point size (by variable	pixels) 2
Geometry Smoothing	None O Fast	⊖ High
Misc Eegend		
Make default Apply	Load	Save Reset Post Dismiss

- **Color** As seen above, the default mesh opacity of 100% obscures the plot beneath it
- Typically, values of 20% to 30% are good if you wish to see both plot features and the mesh

• • • • • • • •



Laboratory for Scientific Computing

Plotting - Mesh

This plot has mesh opacity 20%





Laboratory for Scientific Computing

э

75 / 155

<ロ> <問> < 回> < 回> < 回>

Plotting - Molecule

If code outputs particles, these can be plotted through $\textbf{Add} \rightarrow \textbf{Molecule} \rightarrow \textbf{radius}$



<ロ> <同> <同> < 同> < 同>

Laboratory for Scientific

Computing

76 / 155 /

Simulation courtesy of Tomé



Plotting - Molecule - attributes

⊗ 😑 🗊 Molecule plot attributes		
Atoms Bonds Colors		
Draw atoms as	Spheres 🛟	
Atom sphere quality	(Medium 🛟	
Radius based on	Scalar variable 🗘	
Variable for atom radius	default 👻	
Atom radius scale factor	0.2	
Fixed atom radius	0.3	
Misc		
🗌 Legend		
Mate default Load	Save Reset	
Innín	Rost Dismiss	

- The use of particles in a level set method is not the original intent of the molecule plot option
- Another use of Vislt molecular visualisation (not covered here)
- Radius based on when not dealing with atoms, options are Fixed value and Scalar variable
- Note particles are 3D spheres, even when generated by a 2D level set function - the **Project** or **Slice** operators can be used if you wish to plot with 2D images (see later)

イロト イポト イヨト イヨト

Laboratory for Scientific

Computing



Plotting - Molecule - attributes

😣 🖨 🗈 Molecule plot attr	ibutes
Atoms Bonds Colors	
Discrete colors Color table for:	
Element types	cpt_jmol
Residue types	amino_shapely
Other discrete fields	Default
Continuous colors	
Color table for scalars	plasma
Clamp minimum 0	
🗌 Clamp maximum 💷	
Misc	
🗌 Legend	
Mate default Load	Save Reset
Abblu	Post Dismiss

- Two types of particle output are possible, discrete or continuous (the group code outputs continuous)
- The **Colors** options depend on the particle output, for continuous output, options are similar to other colour selections
- When plotting level set based particles, **Bonds** does nothing

Image: Image:





Plotting - Subset

Can plot either patches or levels, - $\textbf{Add} \rightarrow \textbf{Subset} \rightarrow < \!\! \textbf{subset}_ \textbf{variable} \!\!>$





Laboratory for Scientific Computing

э

79 155

<ロ> <問> < 回> < 回> < 回>

Plotting - Subset - attributes

🔒 😑 🐵 Subset plot attributes	
Subset colors	
O Color table Default	Invert
⊖ Single	100%
Multiple	100%
0	
Subsets	
Opacity	0%
Options Wireframe	Draw internal surfaces
Point / Line Style	
Point type Point \$	Point size (pixels) 2
Scale point size by variable	default 💌
Line style - solid 🛟	Line width
Geometry Smoothing None	○ Fast ○ High
Misc	
Make default	oad Save Reset
Apply	Post Dismiss

- Most options here as seen previously
- **Options wireframe** this makes these plots more useful, instead of block of colour, the outlines of each patch/mesh are plotted

Laboratory for Scientific

Computing



Plotting - Subset

This example shows a wireframe subset of the patches





Laboratory for Scientific Computing

81 / 155 /

Plotting - Vector

Vislt can render vector variables, - Add \rightarrow Vector \rightarrow <variable>





Laboratory for Scientific Computing

3

82 / 155 /

- **▲ 🗗 ▶** - 🗲 🖹

Plotting - Vector - attributes

😣 🚍 🗊 Vector plot	attributes	
Vectors Data Glyp	bhs	
Where to place the v	ectors and how m	any of them
Vector placement	Adapted to reso	lution of mesh
	 Uniformly locat 	ed throughout mesh
Vector amount	Fixed number	400
	 Stride 	1
Only show vectors on original nodes/cells		
Make default		Load Save Reset
Apply		Post Dismiss

• Vector plots typically need work actually be useful through control over their placement and quantity

・ロト ・回ト ・ヨト ・ヨト

Laboratory for Scientific Computing

э



Plotting - Vector - attributes - placement



• Adaptive placement (default, left) prioritises placement on finer patches, uniform placement can look better as a result

Laboratory for Scientific Computing



Plotting - Vector - attributes - stride



- Controls number of vectors in an adaptive setting by changing number of cells between each vector
- This can look odd if the resolution is a multiple of the stride (25 on left, 26 on right)



Laboratory for Scientific Computing

Plotting - Vector - attributes - fixed number



• Useful for uniformly placed vectors, not as effective as stride for adaptively placed vectors

Image: A math a math

Laboratory for Scientific Computing 86 155

• 600 vectors (left) and 60000 (right)



Plotting - Vector - attributes

Vectors Data Glyphs			
Limits			
Limits Use Original Data 🔅			
Minimum 0	Maximum 1		
Color Magnitude Default Invert			
O Constant			
Misc Segend			
Make default	Load	Save	Reset
Apply		Post	Dismiss

a 💿 💿 Bector plot attributes	
Vectors Data Glyphs	
Scale	
Scale 0.5	😸 Scale by magnitude 🛛 😸 Auto scale
Style	
Clyph type Arrow	Brawhead Size 0.25
Arrow body Line : Style	-solid : Width -1 :
Vector origin 🔿 Head 🔿 Middle 🔞 Tail	
Rendering	
Geometry Quality 📵 Fast	O High
Make default	Load Save Reset
Apply	Post Dismiss

- The **Data** tab lets us control the range and colour of the vectors
- Colour is based on magnitude of the vector, though no error is given if e.g. a negative minimum is chosen
- The **Glyphs** tab controls the look of the plotted arrows
- Style is used to ensure vectors are thick enough to be legible on a plot
- Scale is a scaling factor for all the vector arrows - scaling by magnitude means large magnitude vectors appear longer

< <p>Image: Image: Imag



Laboratory for Scientific Computing

Plotting - Vector - attributes



- Scaling by magnitude turned on (left) or off (right)
- However, if most vectors are small, scaling by magnitude might not be the right option (or needs combining with data limiting)



Laboratory for Scientific Computing

Plotting - Streamlines

Technically an operator, not a plot (in recent Vislt versions) $Add \rightarrow Pseudocolor \rightarrow Operators \rightarrow IntegralCurve \rightarrow <variable>$





Laboratory for Scientific Computing

э

89 155

<ロ> <同> <同> < 回> < 回>

Plotting - Streamlines - attributes

Orszag_Tang_vorte	ex_10.hdf:operators/In	tegralCurve/IdealMHD	v
- 1ntegralCurv	e 🗙		
L O Pseudocolor			
😑 💮 🕤 IntegralEurve operato	or attributes		
Integration Appearance Advan	ced		
Source			
Source type Single Point	0		
Location 000			
Field Default			
Integration			
Integration direction	Forward		
Integrator	Dormand-Prince (Runge-K	utta)	•
Limit maximum time step			
Tolerances: max error for step	< max(abstol, reltol*velo	city i) for each componen	it i
Relative tolerance 0.0001			
Absolute tolerance 1e-06		Fraction of Bounding Boo	-
Termination			
Maximum number of steps	1000		
Limit maximum time elaps	ed for particles 10		
Limit maximum distance to	raveled by particles 10		
Make default	Loa	d Save Re	set
Apply		Post Dis	miss

- Since it is an operator, the **IntegralCurve** attributes must be selected
- **Source** how the initial points for the streamlines are seeded, e.g. along a line, circle or at a point
- Sampling how many streamlines (default is 2)
- Integration the direction and method can be chosen - direction both is often useful, unless you have an exact source - changing the method is not advised

Laboratory for Scientific

Computing



Plotting - Volume

Volume plots provide a smoothed rendering of 3D data, using opacity to show features $\mathbf{Add} \rightarrow \mathbf{Volume} \rightarrow < \mathbf{variable} >$





<ロ> <同> <同> < 回> < 回> Laboratory for Scientific Computing



э

Plotting - Volume - attributes

g 🗇 🗇 Bolume plot attributes	😢 🗇 🖓 Olume plot attributes
Renderer Options 1D transfer function	Renderer Options 2D transfer junction
Statuting Statut	Color
Oplotting 2	Color table Default + - Align Omoothing Linear ; Cqual
Optimizing Options	7 7 7 7 7
Compart support variable default +	
	Data
👿 Gample data onto regular grid Mumber of samples 50000 🕻	Ocale 💩 Linear 🔿 Leg 🔿 Otro 1
Theyels	🗆 Minimum 🖉
Gradient method 🔿 Emtered differences 😦 Gobel	
Cabina on Thereial Televisia	Opacity
Sighting	Interaction mode 💩 Zeceform 🔿 Baussian 🔿 Zeom Color Table 🛛 👹 Ohoro colors
Amdient: 0.40 ; Diffuse: 0.75 ; Openlas: 0.00 ; Ohinines: 15.0 ;	
Levy problem (optimp	
Reduction hactor Correct	Creoth Mitensitien
7%x	Opacity variable
Omorth Data	Tariable default v Minimum o Mapimum o
Mate default Dead Bave Neset	Mate dejauli Dave Neset
Apply Post Dismiss	Toph Pest Dieniss

- **Rendering Method** controls how the plot is made, with computationally cheap options looking more smeared
- 1D transfer function applies the opacity, with controls over the colour scheme (see later) and which colours are rendered opaque
- Counter-intuitive the opaque colour on the **opacity** bar is the fully transparent colour in the plot





Outline

Gnuplot

2 Vislt

Plot types

Adding operators to plots

• Altering how data is accessed and displayed

Laboratory for Scientific Computing

92 / 155 /

3 Python scripting in Vislt



Operators

- Operators act on a plot to control how the data appears
- Multiple operators (even of the same type) can be applied
- Order of operation is important!







э

93 155

<ロ> <問> < 回> < 回> < 回>

Operators - Inverse Ghost Zone

Shows the ghost zones of a patch - $\textbf{Operators} \rightarrow \textbf{Debugging} \rightarrow \textbf{Inverse Ghost Zone}$





Laboratory for Scientific Computing

94 155

<ロ> <問> < 回> < 回> < 回>

Operators - Extrude

Extends a 2D plot along a given axis to act as a 3D plot **Operators** \rightarrow **Geometry** \rightarrow **Extrude**





Laboratory for Scientific Computing

Operators - **Revolve**

Revolves a plot about a given axis **Operators** \rightarrow **Geometry** \rightarrow **Revolve**





・ロト ・回ト ・ヨト ・ヨト Laboratory for Scientific Computing

э

Operators - Revolve - attributes

V G LightningSilo_DualSubstrate_PMMATop_Z1mmNotch1mm".silo database:Plasma_p				
- Box	• ×			
- CRevolve				
- O Pseudocolor				
😣 🖨 🗊 Revolve operator attributes				
Type of Mesh?	○ Auto ○ XY ○ RZ (● ZR			
Choose axis based on mesh type?				
Axis of revolution	100			
Start angle	0			
Stop angle	270			
Number of steps	30			
Make default	Load Save Reset			
Apply	Post Dismiss			

- **Type of Mesh?** will attempt to automatically revolve a plot based on the mesh type, e.g. an *r*-*z* cylindrical plot is revolved through the **ZR** option
- Alternatively, an arbitrary axis can be chosen
- Start angle and Stop angle allow for partial revolving
- Number of steps each 'step' is effectively a triangular wedge. Low numbers have visible boundaries, but render quickly, 100-300 is suitable for a revolution that looks right



97 155

(日) (同) (三) (三)

Operators - **Box**

Only plots material within the domain of a given box $\textbf{Operators} \to \textbf{Selection} \to \textbf{Box}$





Laboratory for Scientific Computing

98 155

э

Operators - Box - attributes

😣 🖨 🗈 Box operator attributes					
Amount of cell in the range	● Some ○ All				
X-Minimum	0				
X-Maximum	1				
Y-Minimum	0				
Y-Maximum	1				
Z-Minimum	0				
Z-Maximum	1				
Inverse					
Make default Load	Save Reset				
Apply	Post Dismiss				

- Amount of cell in the range chooses how the edges of the box are defined, only really visible at low resolution
- Extents Based on physical coordinates of the data, in 2D, *z*-extents can be ignored

• Inverse - only the area *outside* the given box is plotted



Laboratory for Scientific Computing

Operators - Clip, Cylinder

Two operators that allow for a region of the plot to be 'cut away' $\textbf{Operators} \rightarrow \textbf{Selection} \rightarrow \textbf{Clip}$ **Operators** \rightarrow **Selection** \rightarrow **Cylinder**





Laboratory for Scientific Computing

э

Operators - Clip, Cylinder - attributes

😣 🔿 🗊 Elip	operator attributes	3			
Quality:	🖲 Fast	 Accurate 			
Olice type:	🖲 Plane	O Sphere			
Clip parameters					
👿 Plane 1					
Origin	0.001 0.001 0				
Normal	-0.5 -0.5 1				
🗌 Plane 2					
Origin	000				
Normal	010				
🗌 Plane 3					
Origin	000				
Normal	001				
🗌 Inverse					
Plane tool controls:					
🔿 Nothing 🔞 Plane 1 🔿 Plane 2 🔿 Plane 3					
Mate default Apply	Load	Save Reset Post Dismiss			
🙁 💿 💿 Eylinder operator attributes					
Endpoint 1 0	0 0				
Endpoint 2 1	0 0				
Radius 0	.001				
🗌 Inverse					
Mate default	Load	Save Reset			

TYOE

MBRIDGE

IINIVERSI

- Slice type up to three different planes, or a single sphere
- Clip parameters planes are defined through origin and normal, spheres through origin and radius
- Inverse as with Box, inverts what is removed
- Plane control tools The top of the plot window has a 'plane control' button, which can dynamically move the axes of a single selected plane
- Cylinders are defined through **Endpoint**s and a **Radius**

イロン イロン イヨン イヨン

Laboratory for Scientific

Computing

Operators - Isovolume

Allows a plot to be limited based on the extents of a variable Operators \rightarrow Selection \rightarrow Isovolume





Laboratory for Scientific Computing

Operators - Isovolume - attributes

😣 🖨 🗊 Isovolume operator attributes				
Lower bound min	Upper bound max			
Variable defaul	t v			
Make default	Load Save	Reset		
Apply	Post	Dismiss		

• A single Variable can be selected, default is the variable in the plot itself

Laboratory for Scientific

Computing

э

103 155

• min and max are default values (±double precision limits)



Threshold (right) is similar to isovolume (left) with different interpolation **Operators** \rightarrow **Selection** \rightarrow **Threshold**



・ロン ・回 と ・ ヨ と ・ ヨ と … э Laboratory for Scientific Computing 104


Operators - Threshold - attributes

ſ	😣 🖨 🗈 Thresh	old operator a	ttributes		I,	
	For individual threshold variables					
l	Variable	Lower bound	Upper bound	Show zone if	ł	
l	default	min	max	Part in rang 🌲	ł	
s					ł	
1.					ł.	
1					I.	
1					I.	
1r					R	
1r	Add var	iable 🔻	Delete sele	cted variable	I.	
l	For all threshold	variables			ł	
1r	Output mesh is	Zon	es from input	O Point mesh	I	
11	Make default Apply	Load	Save	Reset Dismiss	R.	

- The threshold selection is similar to isovolume, in this case, clicking 'min' or 'max' allows these numbers to be edited
- Additionally, a single threshold operator can use multiple variables to apply the limits, through Add variable
- As with the **Box** operator, the selection criteria can consider either partial or fully filled grid cells

Image: Image:



Laboratory for Scientific Computing



Operators - Threshold/Isovolume with Revolve



- Sometimes, especially when using transparency, visual artefacts appear
- The left plot shows material selected only through the Subset button, the right has an isovolume operator on the level set function at $\phi = 1 \times 10^{-4}$



Laboratory for Scientific Computing

106 155

• • • • • • • • • • • • • •

Operators - Cone, Ellipsoid, Spherical

All three produce a 2D surface from 3D data, e.g. **Operators** \rightarrow **Slicing** \rightarrow **Cone**



Laboratory for Scientific Computing

э

107 155

Cones can be projected to 2D, ellipsoids and spheres cannot



This takes a 2D slice from 3D data, and can project it to 2D $Operators \rightarrow Slicing \rightarrow Slice$





Laboratory for Scientific Computing

_ ∢ ≣ →

э

108 155

・ロト ・日下 ・ モト

Operators - Slice - attributes

😣 🖻 🗊 Slice operator attributes		
Normal		
Orthogonal 🔿 X Axis 💿 Y Axis 🔿 Z Axis 🗌 flip		
Arbitrary () 0 -1 0		
Theta-Phi 🔿 180 0		
Origin		
🔿 Point 🖲 Intercept 🔿 Percent 🔿 Zone 🔿 Node		
Intercept 0.0005		
Llp Aris		
S Project to 2D		
Direction 001		
👿 Interactive		
Make default Load Save Reset		
Apply Post Dismiss		

- Normal the three Cartesian directions are available, otherwise either a vector or a rotation can be specified
- Origin Point is a point which cuts the domain, useful for arbitrary normals, Intercept is a single point containing the slice, e.g. the origin in rotated case, or the position on the cut axis
- **Project to 2D** can be turned off, in which case the slice appears as part of a 3D plot
- **Direction** is used to project a non-orthogonal slice to 2D



Laboratory for Scientific Computing



Operators - Slice

Example of a (non-projected) slice used in conjunction with contours



110 / 155 /

Operators - ThreeSlice

Three slices, in the x-, y- and z-planes **Operators** \rightarrow **Slicing** \rightarrow **ThreeSlice**





<ロ> <同> <同> < 回> < 回> Laboratory for Scientific Computing

э

111 / 155

Operators - Elevate

$\begin{array}{l} \mbox{Creates a 3D surface of 2D data} \\ \mbox{Operators} \rightarrow \mbox{Transforms} \rightarrow \mbox{Elevate} \end{array}$





Laboratory for Scientific Computing



э

Operators - Elevate - attributes

2	😣 🖨 🗊 Elevate	operator attributes
ra	Elevation heigh	t relative to XY limits?
a	Limits Mode	Original Data Original Data
•	Scale	💿 Linear 🔘 Log 💮 Skew
	Skew factor	1
a	Use min	
a	Min	0
a	Use max	
l	Max	1
l	Elevate with ze	ro height?
l	Elevate by Variable	default 👻
l	Make default Apply	Load Save Reset Post Dismiss

- Control over the way elevation is handled is used to scale the surface heights to make features clear
- Elevate with zero height? effectively converts a 2D plane into a 3D sheet (opposite of Slice), so 2D data can be plotted with 3D data
- Elevate by Variable means the plotted variable does not have to control the surface height

< D > < A >



Laboratory for Scientific Computing

Operators - Project

Projects a 3D plot to a 2D surface **Operators** \rightarrow **Transforms** \rightarrow **Project**







-

э

114 155

-

• • • • • • • •

Operators - Project

😣 🖨 🗊 Project opera	tor attributes	
Projection type $(\mathfrak{Z}-\mathfrak{A}\mathfrak{x}\mathfrak{i}\mathfrak{s} \ \mathfrak{C}\mathfrak{a}\mathfrak{r}\mathfrak{t}\mathfrak{s}\mathfrak{s}\mathfrak{i}\mathfrak{n} \ (\mathfrak{x}'=\mathfrak{x}, \mathfrak{y}'=\mathfrak{y})$		
Vector transform method	Treat as instantaneous directions	
Mate default	Load Save Reset	
Ջենն	Post Dismiss	

- Not all 3D plots are suitable for projection - the output may contain information combined through the plot, slice may be a better option
- Projection type below is Y-Axis Cartesian (x- and z-axes remain)



Laboratory for Scientific

Computing

 $\frac{115}{155}$





Operators - Reflect

Allows the plot to be reflected along a coordinate direction $\textbf{Operators} \rightarrow \textbf{Transforms} \rightarrow \textbf{Reflect}$



Laboratory for Scientific Computing

э

116 155

<ロ> <問> < 回> < 回> < 回>



Operators - Reflect

a a Reflect ope	rator attrib	utes 🧳	
Input mode	2D	○ 3D	
Original data quadrant	+X +Y		\$
Reflection quadrants			
) +Y		
Î			
Reflection Limits:			
🖲 Use dataset min 🤇	Specify X =	0	
🖲 Use dataset min 🤇	Specify Y =	0	
● Use dataset min (Specify Z =	0	
Make default	oad :	Save Re	set
Apply		Post Disr	miss

- Input mode 2x2 grid in 2D, 2x2x2 grid in 3D, simply click the shapes corresponding to where you want to turn reflection on/off
- Original data quadrant where is the purple ball placed, allows for easy reflection through any coordinate edge
- **Reflection limits** Reflection does not have to correspond to the domain edge, and can be offset, either to remove trailing white-space, or introduce it for clarity

Laboratory for Scientific

Computing



Operators - Transform

A general set of translation, rotation and scaling operations Operators \rightarrow Transforms \rightarrow Transform

rbitrary Coordina	te Linear				
Rotate					
	Origin	0 0 0			
	Axis	0 0 1			
	Amount	0	Deg		Rad
Scale					
	Origin	000			
	Х	1	Y 1	Z 1	
Translate					
	х	0	Y 0	Z 0	
🗹 Transform vecto	s				
Make default			Load	Save	Reset

- Rotate rotates a 2D plot about a point, or 3D about an axis
- Scale multiply an axis by a given factor, e.g. can convert mm to m, or make a rectangular domain square
- **Translate** move a plot in a specified direction, quantities always based on the physical coordinate values, useful for plotting more than one image in a single window





Outline

Gnuplot

2 Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

<ロ> <問> < 回> < 回> < 回>

Laboratory for Scientific

Computing

э

118 / 155

3 Python scripting in Vislt



- Controls are accessed from the Vislt menu bar (or by shortcut keys)
- They affect the overall window display, rather than individual plots
- They also allow from manipulation of entire data files, e.g. comparisons or function creation
- We will only cover the most useful one (or, at least, the ones I find most useful...)

イロン イロン イヨン イヨン

Laboratory for Scientific

Computing

э



Controls - Annotation - General

The annotation menu pretty much controls all text in the window, almost always needs modifying to produce plots for publication or presentation

⊗ ⊜ ⊕ Annotation
General 2D 3D Array Colors Objects
Segend No annotations
S Database
Path Expansion File 2
Font name Arial 🛟 Font scale 1 🛛 Bold 🗌 Italic
Use foreground color
Time Time offset 0
Viser information
Font name Arial C Font scale 1 Bold Italic
Use foreground color
Make default Reset
Apply Post Dismiss

- Legend this can toggle all legends on or off
- Database displays the filename, cycle (often the code or output iteration) and time (if available from the output file) - useful for identifying plots, but looks a bit messy
- User Information On by default, and gives your username, e.g. crsID why?!

・ロン ・部 と ・ ヨ と ・ ヨ と …

120 155

Laboratory for Scientific

Computing



Controls - Annotation - 2D - General 2D

8⊜© Annotation			
General 2D 3D 2	ltray Colors Objects		
Show ares			
General 2D X-Ari	s Y-Aris		
👿 Auto scale label	values		
👿 Auto set ticks			
Tick mark locations	Dutside		\$
Show tick marks	Bottom-left		
Line width	-1		:
Mate default			Reset
Apply		Post	Dismiss

- The functionality of 2D and 3D is very similar
- Auto scale label values Vislt will choose the best format for displaying numbers, if deselected, each axis can be chosen independently
- Auto set ticks default is to divide the axis into 5 major regions, each of 10 minor regions, and not marking the axis extremes (e.g. (0,0)), again this can be chosen independently for each axis if necessary
- Line width this controls the width of both the bounding box and the ticks (but not the length of the ticks)
- In 3D, Show triad and Show bounding box options are also available



Laboratory for Scientific Computing



э

Controls - Annotation - 2D - X-Axis

😣 🗇 🕀 Annotation	a	
General 2D 3D	Array Colors Objects	
Show axes		
General 2D X-Axis	s Y-Avis	
Title		
Custom title	X-Axis	
Custom Units	IS	
Font name Co	purier 🗘 Font scale 1 🧭 Bold 💈	/ Italic
👿 Use foregroun	nd color	100%
🧭 Labels		
Scaling (x10"?)		v
Font name Co	ourier 🗘 Font scale 1 👿 Bold 🛚	Italic 1
🥑 Use foregrou	ind color	100%
Tick marks		
Major minimum	0	
Major maximum		
Minor spacing	0.02	
Major spacing	0.2	
Show grid		
Make default		Reset
Apply	Post	Dismiss

- All axis menus are functionally identical (2D and 3D)
- Title set the desired axis label
- Labels The numerical values on the axis, font options are always available, scaling options only if Vislt is not doing this automatically
- Tick marks If not automatically set, they are chosen here
- Show grid overlay a grid based on the major tick spacing - needs to be done for each axis, individually, only a series of lines are plotted

Laboratory for Scientific

Computing



Controls - Annotation - Objects - Legend

Text 3D Text Time slider 2D Line 3D Line	Anno Lege Lege Lege Lege	tation objects nd:Pseudocolor - Schlieren nd:Contour - IdealMHD_p nd:Mesh - AmrMesh nd:Subset - AMR Patches nd:Vector - IdealMHD_v	
Image		Hide/Show	Delete
Legend posi	manag	e legend position	V
X-scale		100% * Y-scale	100%
Orientation		Vertical, Text on Right	\$

- The objects tab can control a lot, including legend positions
- **Position** Vislt attempts to place legends sensibly (beside the plot), based on the chosen **Orientation**, which doesn't always look good for three or more
- Deselecting the checkbox gives you control, the coordinate position is the top-left corner of the legend object, as a fraction of the window
- The arrow box gives you a cross-hair location tool
- Note Vislt does not consider manually placed legends when auto-placing others

Laboratory for Scientific

Computing



Controls - Annotation - Objects - Legend

	Ceneral 2D 3D Array Colors Objects
Ta	Create new Annotation objects Text Legand Paradox - Schleren UD Text Legand Annot - Man Kesh Time sidedr Legand Markon - Man Kesh Zo Line So Line
c	Image Hide/Show Delete
01 1c 01 01	Fostion Tick Atarks Appearance Sounding box Graw title G Draw title G Draw tinhmax
01	Use foreground color Text color Font height 0.015
l	Font family Arial :
1	Make default Reset Apply Post Damins

- Appearance some control over how the legend is displayed
- Bounding box produces a coloured box behind the legend, with transparency, useful if legend needs to be on the plot
- Draw Title toggles a legend title comprising plot type and (Vislt) variable name, contents cannot be changed
- **Draw min/max** Default is to have absolute variable extents (not plot-limited) beneath the legend, not usually necessary
- The font control is similar to previous examples, size and format often need changing for readability



Laboratory for Scientific Computing

Controls - Annotation - Objects - Text

😑 😑 🔋 Annotati	on
General 2D 3D	Array Colors Objects
Creale new Text 3D Text Time slider 2D Line 3D Line Jmage	Annetation objecta Gegend-Bipmaham — Materiala Segend-Bipmaham – Materiala Segend-Biptae – 2009; Platices Segind-Bubber – 2009; Platices Segin – 2016; segind-Biptae Segind-Bibber – 2019; Platices
	Hide/Show Delete
Text	
Lower left	0.5 0.5
Steight	3% +
Text	2D text annotation
🛃 Llse foregro	und color
Text color	100%
Font family	Arial 🗘
B010	🗌 Italic 🗌 Shadow
👿 Visible	
Mate default	Reset
Apply	Post Dismiss

- New objects can be created, in addition to the legends Vislt will always ask for a name, in the GUI environment, this name is irrelevant
- Text text can be placed on plots, e.g. titles, labels, custom legend heading
- Position is a fraction of the view window, and, as the name suggests, measured from the **Lower left** of the text
- Font controls as other cases
- Certain variables can be entered as text, e.g.
 \$time will output the simulation time (avoids having to plot full database information)



Laboratory for Scientific Computing



Controls - Annotation - Objects - Line

deneral 2D 3D	Array Colors Objects	
Ereate new	Annotation objects	
Tert	Legend:Boundary — Materials	
	Legend:Bseudocolor - Air tho Segend:Swheet - 91993 Matches	
320 .7 ekt	Tert - \$time	
Time slider	2D Line	
2D Line		
3D Line		
Image		
	Hide/Show Delete	
2D Line		
Start	0.25 0.25	•
End	0.75 0.75	•
Width	-1 :	
Style	solio\$	
🗹 Llse foregro	und color	
Line color	0%	
Begin arrow	None	1
End arrow	None	•
👿 Visible		
Mate default	Res	et

- Line lines can be drawn on plot, including as arrows, useful for labelling
- Start and end, as with other objects, are fractions of the view window
- Additionally, there is **3D Text** and **3D Line**, for adding these features to 3D plots. My little experience of 3D text is that is a bit of a pain to work with

《口》《聞》《臣》《臣》

Laboratory for Scientific

Computing



Controls - Annotation - Objects - Image

- Adding additional images to a plot could actually be useful for comparing plots to those of a paper
- Applying transparency can be a graphically intensive operation





Laboratory for Scientific Computing



э

Controls - Color tables

😣 😑 💿 Colo	e tables	100 to 1 to 1
Active color tabl	e	
Continuous		hot
Discrete		levels
		🗌 Group tables by Category
Manager		
New	cpt_jmol	
Delete	cpt_rasma difference aran	
Export	hot	
Name	hot	ĭ
Oltor Number of co Color table ty Align	lors 5 pe 🖲 Ci Smoothing	ontinuous O Discrete g Einear Equal
7	7	1 4 7 1
Red 🚃		255 🗘
Green -		255 💭
Blue D		0
Alpha 📥		255 🗘
Apply		Post Dismiss

- You can make your own colour scheme, e.g. for matching another plotting program's results
- Each triangle can slide, and have its colour set, and Vislt will interpolate between them (in continuous mode)
- Modifying the existing colour tables is possible, at least within a single plot

Laboratory for Scientific

Computing



Controls - Command

😒 🍵 🕞 Commands
Commands
Record Im Pause Stop
Store commands in Active tab
Append commands to existing text
1 2 3 4 5 6 7 8 Macros
<pre>AddPost(Wester, *Diama_JOperable*, 1, 0) BetActivePiot(5) BetActivePiot(5) BetActivePiot(5) BetActivePiot(5) Berolvatts.meshType = RevolveNtis.IR # Anto, XT, EZ, ZR Berolvatts.meshType = RevolveNtis.IR # Anto, XT, EZ, ZR Berolvatts.startAngle = 0 RevolveNts.startAngle = 0 RevolveNts.startAngle = 0 BerolveNts.startAngle = 0 BerolveNts.startAngle = 0 DeardOtagi BetOperatorOptions(RevolveNts, 0) DrawTotagi BetOperatorOptions(A, 5) BetActivePiots(4) HiddwitveNist(5) DrawFotagi)</pre>
Execute Clear Make macro
Post Dismiss

- The command menu is primarily used with the python interface (see later)
- It can identify the commands which actually add, edit and draw plots
- Pressing **Record** will start recording every operation that you do through mouse clicks, and they will be printed when **Stop** is pressed
- The commands are also editable (then press **Execute**), so could offer a quick way of setting multiple desired features in a new plot

Laboratory for Scientific Computing

• There are a few commands which don't get recorded well, e.g. **Pick**



Controls - Database correlations



- Allows two or more databases to be controlled by a single time slider
- Select New, and a new window appears allowing you to name your correlation (as it will appear in the Active source menu of the main window

ew correlation	N Correlation: Correlation: Correlation	9801
Same number of states 💲	Correlation method Padded index	÷)
Padded index 🗘	Sources Lorentz HamaAluminium/PMMA_fakeCo LightningSilo_Dua/Substrate_PMMATop_ Lorentz PlasmaAluminium/PMMA_fakeCo BeamCenterPolite.xt Orszag_Tang_vortex_10.hdf	Correlated sources
Post Dismiss	Create database correlation	Cancel



Laboratory for Scientific Computing



Controls - Database correlations

Ì	110185enliteren 8 = 1 Create di	Default correlation	connetiicei 12 1: Correlation	added index *	I,
ł	Name	Correlation02			
1	Correlation method	Padded index		:	L
ł	Sources			Correlated sources	L
	LorentzPlasmaAlum LightningSilo_DualS LorentzPlasmaAlum BeamCentreProfile. Orszag_Tang_vortex	iniumPMMA_fakeCo Substrate_PMMATop_ iniumPMMA_fakeCo ixt _10.hdf	->		
	Create database co Unonsenileren	Default correlation	ton method 12 12 Correlation	Cancel added index ******	
ł	Name	Correlation02			
i	Correlation method	Padded index		:	I.
	Sources			Correlated sources	L
	LightningSilo_Dual5 BeamCentreProfile.1 Orszag_Tang_vortex	Substrate_PMMATop_ ixt _10.hdf		LorentzPlasmaAluminiumPMMA_fakeCo	
I					
_					
1	Create database co	rrelation		Cancel	1

- Click on as many databases as you wish to correlate, this highlights them in the left box
- Pressing the arrow adds them to the correlation (if you made a mistake, you can remove them again)
- Sometimes, when plotting two databases on the same window, Vislt will offer to create a correlation for you
- If the databases have different numbers of outputs, this is handled ok, on set of results will simply stop advancing whilst the other continues to

Laboratory for Scientific Computing



131 / 155 /

	Name	Schlieren		
operable				
Schlieren	Туре	Scalar Me	ih Variable	÷
		Show 1	ariable in plot menus	
	Stan	dard Editor	Python Expression Editor	
	Defin	ition		
New Delete			Inset Function. v Inset	t Variable *

- Expressions are user defined functions and variables
- () defines function evaluation, or standard mathematical use, [] an array entry and {} a vector quantity
- +, -, *, / and order of operations works as normal, other syntax is not necessarily straightforward
- Variable names, through Insert variable, are the Vislt names, and can include previously defined expressions

Image: A math a math

Laboratory for Scientific

Computing



	Nam	e Schlieren		
_operable	-	Contraction of the second		
ichlieren	Туре	Scalar Me	sh Variable	÷
		Show 🛛	ariable in plot menus	
	Sta	ndard Editor	Python Expression Editor	
	Def	inition		
New Delete			Insert Function	Variable *

- Insert function provides a list of functions, separated into approximately useful categories
- Some functions are inserted with syntax hints, for example, selecting 'lt':

lt(<var-LHS>, <var-RHS>)

- This is the less than (<) function
- Expressions can make scalars, vectors, and tensors, and can be constants, as well as functions

Image: A mathematical states and a mathem





🧧 🕤 Eppressions			
Expression List	Definitio	on	
x	Name	x	
J_operable Schlieren	Туре	Scalar Me	sh Variable 2
		Show v	ariable in plot menus
	Stand	dard Editor	Python Expression Editor
	Defin	ition	
New Delete Display expressions from database			Insert Function • Insert Variable •
Apply			Load Save Dismiss

- Perhaps surprisingly, Vislt has no inbuilt means to access coordinate information - a position expression must be created
- It does have a 'coord' function, though, this example defines the x-position
- The variable within the coord function should not matter
- The output of coord() is a vector, the *x*-component is the 0th entry

Laboratory for Scientific

Computing

134 155

Image: A mathematical states and a mathem



	Definitio	on		
	Name	J_operable		
chlieren	Type	Vector Me	sh Variable	:
		Show 1	variable in plot menus	
	Stan	dard Editor	Python Expression Editor	
	Defin	nition		
	{Plas	sma_J[0],Pla	sma_][2]}	
New Delete				
New Delete Delete			Insert Function •) Insert Varia	able v

- Defining vector expressions is just a matter of putting each component within a comma separated list enclosed by {}
- Vislt will only use vector entries up to the dimension of the plot (i.e. the first 2 or 3)

Image: A mathematical states and a mathem

Laboratory for Scientific

Computing



Another useful example - mock-Schlieren command

	Definitio	n		
X Loperable	Name	Schlieren		
Schlieren	Туре	Scalar Mesh Variable	•	
		🧭 Show variable in plot menus		
	Stand	ard Editor Python Expression Editor		
	Defini	tion		
	exp(-:	20*magnitude(gradient(IdealMHD_rho))/sqrt(IdealMHD_rho)/1000	k	
			-	
New Delete			-	
 Display expressions from database 		Insert Function v	<u> </u>	
		Load Save		

Controls - Lighting

😣 🖨 🗈 Lighting		
Mode 🖲 Edit 🔿 Preview	Active Properties	light 💡 1 💲
	Light type Direction Color Brightness	€amera ; 0 0 -1
Make default Apply		Reset Post Dismiss

- In 3D, Vislt adds lighting effects, to attempt to improve visibility
- However, this can leave some parts of the plot in shadow, fortunately lighting can be customised
- Up to 8 lights can be **Enabled** light 1 cannot be disabled
- Position can be controlled with click-and-drag the dot in the box, or through the **Direction** box
- Each light has its own colour, brightness and direction ('Camera' an external source, 'Object' - from the object, 'Ambient' - no direction)



Laboratory for Scientific Computing



Playing with lighting doesn't always produce good plots...





э

138 155

Controls - Query

	Query parameters
All	Original Data
Queries	 Actual Data
Line Scan Transform Lineout Cocalized Compactness Factor Mass Distribution Memory Usage Min MinMax Moment of Inertia Node Cocards	Do Time Query Start and End are Time steps, not cycles or times. Starting timestep Starting timestep Stride 1
Query results	Float Format: %g
-	

- The query window allows you to do some post-processing of the data
- The output is given in the **Query results** window, query will always act on the currently selected plot
- **Do Time Query** will produce a plot over time for the desired variable (not available for all queries)
- Depending on the data structure, the query results may not take account of geometric effects, e.g. cylindrical domain

Laboratory for Scientific

Computing


Controls - View - 2D View

J	😣 🖨 🐵 View					
ł	Curve view	2D view	3D view	AxisArray viev		
1	Viewport	0.2 0.95 0.15 0.95				
l	Window	533333333333333 0.025333333333333333				
s	Full Frame	Auto	🔘 On	○ Off	- 18	
,S	X Scale	Linear	r 🔿 Log	5	s	
u	Y Scale	Linear	r 🔿 Log	5	- 11	
s					s	
u					- 11	
;S					5	
s,S					s	
şS					s	
u					- 11	
l					- 11	
l					- 11	
f					- 1	
or:						
se	Commands					
l	Apply		Po	ost Dism	niss	

- The View menu customises the size of the plot relative to the size of the window, as well as the axis extents
- Viewport specifies the bottom-left and top-right corners of the plot, x first, then y
- The scaling cannot be changed by this, once one coordinate domain is full, the other will no longer be re-sized
- The **Window** corresponds to the axis ranges, allows for accuracy when zooming

Laboratory for Scientific

Computing



Controls - View - 3D View

😸 🖨 🗊 View					
Curve view 2D vie	w 3D view ArisArray vien 4 🕨				
View normal	001				
Focus	077 -4.547473508864641€-13				
Up Vector	010				
Angle of view	30				
Parallel scale	11845.8				
Near clipping	-23691.5				
Far clipping	23691.5				
Image pan	0 0				
Image 300m	1				
Shear	0 0 1				
Eye Angle (stereo)	2				
	Serspective				
Align to axis	-3 1				
🗌 Scale 3D ares	111				
Commands					
Apply Post Dismiss					

- 3D offers much more customisability, due to rotational freedom
- Personal preference get roughly the right position through click-and-drag, then tweak through View menu
- Large files can be graphically expensive to do this, so the view menu can avoid 'in-between' rendering of two different views
- Some options are fairly obvious (Image Zoom, Image pan), some fiddly (View normal) and some best adjusted through trial and error when things don't look right!



Laboratory for Scientific Computing

141

(日) (同) (三) (三)

- There are many other options in Vislt (plots, operators, controls), not covered here, mostly because I've never used them (or very rarely, a long time ago)
- For example, Controls → Data-Level Comparisons lets you do things such as subtract one database from another, and Controls → Material Options changes how boundaries are interpolated
- Other features relate to data types I've never used, or to a level of visualisation far beyond anything I've considered



(日) (同) (三) (三)

Outline

1 Gnuplot

Vislt

- Plot types
- Adding operators to plots
- Altering how data is accessed and displayed

<ロ> <問> < 回> < 回> < 回>

Laboratory for Scientific Computing

э

142 / 155

O Python scripting in Vislt



- In addition to the Vislt GUI, it can also be run remotely, through scripts
- This can speed up data processing, e.g. making similar plots from parameter study data
- It can be a safe way of generating the images for movies, especially if several minor changes are likely to be necessary, and you don't want Vislt to crash

Laboratory for Scientific

Computing

-

- It can avoid excessive graphical rendering for 3D plots
- Vislt can be operated remotely, avoiding rendering over an SSH connection
- Post processing information can be automated



- Vislt supports a command line interface, in addition to the GUI, which can be used, even without running a script (I have never tried this)
- Assuming you have a script to run, this can be called through the command line by:
 visit -cli -nowin -s <script_name>.py
- This should start generating some default output text to the terminal

Running: cli -dv -nowin -5 MeltWidthScript62_and 47.py Running: viewer -dv -nowin -noint -host 127.0.0.1 -port 5604 Running: mdserver -dv -host 127.0.0.1 -port 5604 Error opening plugin file: /home/stevemillmore/tmp/visit2.13. ZNK11xercesc_3_113XMLAttDeflist14isSerializableEv) Running: engine_ser -dv -host 127.0.0.1 -port 5604

- The error here is Vislt trying to open a plugin for file reading (GDAL here), it does not cause a problem in this case, since I don't attempt to open a GDAL file
- Note: Errors in the script itself are not necessarily handled gracefully, and will leave you in Vislt's command line interface, > exit() will get you back to the terminal



- The command line interface basically runs through python, therefore, any python program can actually be run this way
- Not recommended much slower than just running python
- However, this means that scripts can be a combination of regular, and Vislt-specific python, including from any desired libraries etc.
- We shall cover a few examples, and how to set up a file, in general, specific commands are best identified through the **Command** feature in the Controls menu



```
OpenDatabase("localhost:/data/hydra04-2/stm31/cns_amr/CNS_AMR_Multimaterial/output/oscillatingEllipseIdeal/surfaceTensionDroplet100_*.hdf database", 0)
AddPlot("Curve", "operators/Lineout/Air_LS", 1, 0)
LineoutAtts = LineoutAttributes()
LineoutAtts.point1 = (0, 1e-5, 0)
LineoutAtts.point2 = (0.5, 1e-5, 0)
LineoutAtts.interactive = 1
LineoutAtts.ignoreGlobal = 0
LineoutAtts.samplingOn = 0
LineoutAtts.numberOfSamplePoints = 10000
lineoutAtts.reflineLabels = 0
SetOperatorOptions(LineoutAtts, 0)
for state in range(TimeSliderGetNStates() ):
    SetTimeSliderState(state)
    DrawPlots()
    SaveWindowAtts = SaveWindowAttributes()
    SaveWindowAtts.outputToCurrentDirectory = 0
    SaveWindowAtts.outputDirectory = "/local/data/public/stm31/TexStuff/ValidationPaper/Figs/"
    SaveWindowAtts.fileName = "Ellipse0Levels"
    SaveWindowAtts.family = 1
    SaveWindowAtts.format = SaveWindowAtts.CURVE # BMP, CURVE, JPEG, OBJ, PNG, POSTSCRIPT, POVRAY, PPM, RGB, STL, TIFF, ULTRA, VTK, PLY
    SaveWindowAtts.width = 1024
    SaveWindowAtts.height = 1024
    SaveWindowAtts.screenCapture = 0
    SaveWindowAtts.saveTiled = 0
    SaveWindowAtts.guality = 80
    SaveWindowAtts.progressive = 0
    SaveWindowAtts.binary = 0
    SaveWindowAtts.stereo = 0
    SaveWindowAtts.compression = SaveWindowAtts.PackBits # None, PackBits, Jpeg, Deflate
    SaveWindowAtts.forceMerge = 0
    SaveWindowAtts.resConstraint = SaveWindowAtts.ScreenProportions # NoConstraint, EqualWidthHeight, ScreenProportions
    SaveWindowAtts.advancedMultiWindowSave = 0
    SetSaveWindowAttributes(SaveWindowAtts)
    SaveWindow()
```

《曰》《聞》《臣》《臣》

146

Laboratory for Scientific

Computing



Opening and closing

UNIVERSITY OF

MBRIDGE

• OpenDatabase is used to open either individual files, or entire databases

```
OpenDatabase('localhost:/data/hydra04-2/stm31/cns_em/CNS_AMB_Multinaterial/output/oscillatingEllipseIdeal/surfaceTensionDroplet100_*.hdf database", 0)
Gor i in range(1,151):
    varlableString = "localhost:"+locationString+inFileName+"/cavitationDropAluminiumVibrating"+str(i)+".hdf"
    OpenDatabase(variableString)
```

• Closing the database again is through CloseDatabase

DeleteAllPlots()

CloseDatabase(variableString)

- Vislt cannot close a database whilst it is being used for plots (same as GUI mode), DeleteAllPlots() ensures this is the case
- If plots are not closed, then they remain open until exit() is called
- In the case of looped opening, this can act as a memory leak

Laboratory for Scientific Computing





- By default, there is a focus (SetActive...) on the last file opened, plot created, window opened etc.
- This will be used for further functions, e.g. plots come from the active file, attributes are changed for the active plot
- This can be changed through: SetActiveTimeSlider(<n>) for the file, SetActiveTimePlots(<n> [,<m>]) for the plots, and SetActiveWindow(<n>) for the window
- The features are numbered in the order they are created, starting from zero
- Note multiple plots can be active at once (equivalent to highlighting multiple plot types in the GUI)
- For scripting, it is usually easiest to only modify the current plot, since commands can be placed exactly where you need them

148

Laboratory for Scientific

Computing



Adding plots

```
AddPlot("Boundary", "AMRMaterials")
bdryAtts = BoundaryAttributes()
bdryAtts.colorType = bdryAtts.colorBySingleColor
bdryAtts.singleColor = (0,0,0, 255)
bdryAtts.leendFlag = 0
setPlotOptions(bdryAtts)
```

```
AddPlot("Pseudocolor", "Aluminim_l_magnitude")
tarPsAtts = PseudocolorAttributes()
tarPsAtts.mhrlag = 1
tarPsAtts.mkn = 0
tarPsAtts.nkn = 0
tarPsAtts.colorTablesame = "hot_and_cold"
tarPsAtts.colorTablesame = 0
tarPsAtts.caling = 0
setPlotOptions(tarPsAtts)
```

```
Addoperator("Threshold")
watThAtts = ThresholdAttributes()
watThAtts.lowerBounds = 1e-5
watThAtts.listedVarNames = "Aluminium_LS"
SetOperatorOptions(watThAtts)
```

- Adding plots is a matter of AddPlot(''<Plot_name>'',''<Variable_name''
- Altering the attributes of this plot requires an attributes object to be created
- Values are then edited within this object
- They are applied to the plot through a Set...Options call
- Once all plots are set, DrawPlots() will plot them (not that you'll actually see this...)

(日)

Laboratory for Scientific

Computing



Saving plots

```
outputString = "PlasmaCarbonCurrent_VCPaper"+str(i)
saveAtts = SaveWindowAttributes()
saveAtts.outputToCurrentDirectory = 0
saveAtts.outputDirectory = "/lsc/zeushome/stm31/TexStuff,
# saveAtts.fileName = outputString
saveAtts.fileName = outputString
saveAtts.fileName = saveAtts.PNG
saveAtts.height = 1024
saveAtts.velth = 2048
saveAtts.screenCapture = 1
SetSaveWindowAttributes(saveAtts)
```

```
SaveWindow()
```

- Saving files requires the creation of SaveWindowAttributes, and setting the decided options
- Even though there is no screen, screenCapture can be used, with results at the resolution of the graphics output of the machine the script is run upon (I think)
- Once settings are saved, SaveWindow will save output to the desired location



(日)



Some examples - defining expressions

import math import sys #transparencyCutOff = 3.5e6 for i in range(0, 201): #for i in [1,10,100,150]: variableString = "localhost:/media/AnotherHardDrive/cns amr/PlasmaModelvariableString = "localhost:/home/stevemillmore/cns amr/PlasmaModel-stm OpenDatabase(variableString) schlString = "coord(AMRMesh)[0]" DefineScalarExpression("x", schlString)# schlString = "coord(AMRMesh)[1]" DefineScalarExpression("y", schlString)# schlString = "{Plasma J[0].Plasma J[2]}" DefineVectorExpression("Plasma JOperable", schlString) schlString = "{Aluminium J[0],Aluminium J[2]}" DefineVectorExpression("Aluminium JOperable", schlString) print ("Plotting " + variableString) AddPlot("Boundary", "AMRMaterials") bdrvAtts = BoundarvAttributes() bdrvAtts.colorType = bdrvAtts.ColorBySingleColor bdrvAtts.singleColor = (0.0.0, 255) bdryAtts.lineWidth = 3 bdryAtts.legendFlag = 0 SetPlotOptions(bdryAtts) AddPlot("Pseudocolor", "Aluminium_J_magnitude") tarPsAtts = PseudocolorAttributes() tarPsAtts.minFlag = 1 tarPsAtts.maxFlag = 1 tarPsAtts.min = 0 tarPsAtts.max = 3e8 tarPsAtts.colorTableName = "hot and cold" tarPsAtts.invertColorTable = 0 tarPsAtts.legendFlag = 0 tarPsAtts.scaling = 0 SetPlotOptions(tarPsAtts)



- Expressions are easy to define, simply enter the same text as would be used in the **Controls** menu
- The assigned names can then be used in place of any other variable in the plot menus

(日) (同) (三) (三)

Laboratory for Scientific



Some examples - annotation and view

```
annAtts = AnnotationAttributes()
annAtts.userInfoFlag = 0
annAtts.databaseInfoFlag = 0
annAtts.axes3D.visible = 0
annAtts.axes3D.triadFlag = 0
annAtts.axes3D.bboxFlag = 0
annAtts.axes2D.xAxis.title.font.font = annAtts.axes2D.xAxis.title.font.Arial
annAtts.axes2D.xAxis.title.font.bold = 0
annAtts.axes2D.xAxis.title.font.italic = 0
annAtts.axes2D.xAxis.title.font.scale = 2
annAtts.axes2D.yAxis.title.font.font = annAtts.axes2D.yAxis.title.font.Arial
annAtts.axes2D.yAxis.title.font.bold = 0
annAtts.axes2D.vAxis.title.font.italic = 0
annAtts.axes2D.vAxis.title.font.scale = 2
annAtts.axes2D.xAxis.label.font.font = annAtts.axes2D.xAxis.label.font.Arial
annAtts.axes2D.xAxis.label.font.bold = 0
annAtts.axes2D.xAxis.label.font.italic = 0
annAtts.axes2D.xAxis.label.font.scale = 2
annAtts.axes2D.yAxis.label.font.font = annAtts.axes2D.yAxis.label.font.Arial
annAtts.axes2D.vAxis.label.font.bold = 0
annAtts.axes2D.vAxis.label.font.italic = 0
annAtts.axes2D.yAxis.label.font.scale = 2
```

```
SetAnnotationAttributes(annAtts)
```

```
view = View2DAttributes()
view.viewportCoords = (0.15, 0.98, 0.15, 0.95)
view.windowCoords = (0, 0.05, -0.01, 0.05)
SetView2D(view)
```

```
#view = view3DAttributes()
%view.view50nral = (0.7, 0.65, 0.3)
#view.rocus = (0, 0, -0.005)
#view.rocus = (0, 0, -0.005)
#view.naprallelScale = 0.1
#view.naprallelScale = 0.2
#view.ringepane = (0, -0.12)
#view.ring
```



- Some variables can be picked by name (e.g. colorTableName when plotting pseudocolors), the fonts available are not amongst them
- Here we show both 2D and 3D (commented) view setting options



(日) (同) (三) (三)

Some examples - more selection and query

for i in range(9100, 9800):

```
zt = str(1).zftll(5)
vartablestring = 'home/steventllmore/AMReX/heat-equation-in-amrex/Exec/run3d/h
st_"szt+"/Header"
if(not os.path.isftle(vartablestring)):
```

continue

it = it + 1

OpenDatabase(variableString)

AddPlot("Pseudocolor", "phi") DrawPlots()

Query("Time")
time = GetQueryOutputValue()

outString = str(time) + " "

```
SetQueryFloatFormat("%g")
```

for dx in range(11):

```
ZonePick(coord=(0.0252, 0.0248, 0.0029))
#ZonePick(coord=(0.0252472, 0.025198, 0.0029 + float(dx) * 1e-5))
ZonePick(coord=(0.0253008, 0.0250005, 0.0029 + float(dx) * 1e-5))
pick = GetPickOutput().split()
pickVal = 0.0
# print len(pick)
for entry in range(len(pick)):
           print pick[entry]
   **
    if entry > 2 and pick[entry-2] == "<zonal>":
       pickVal = float(pick[entrv])
       # print "time", time, "pick", pickVal
       if pickVal < 290:
           print "Something has gone wrong ", pick
           exit()
outString = outString + str(pickVal) + " "
```

```
outString = outString + "\n"
historyFile.write(outString)
```

- If files are not consecutively numbered, python commands ensure missing ones are not opened
- Some queries, e.g. Time, output a single number, this is obtained as a floating point number through getQueryOutputValue()
- ZonePick can obtain information at a point, obtained through GetPickOutput()
- This is returned as a string, with text data included, hence needs manipulation before the desired output can be used

(日) (同) (三) (三)

Laboratory for Scientific

Computing



Some examples - more selection and query

for i in range(9100, 9800):

```
zl = str(1).zffl(16)
varlableString = '/how[steven[llnore/AMRek/heat-equation-in-anrek/Exec/run3d/h
et_-stri-/header'
ff("continue
tif etr + 1
OpenDatabase(varlableString))
Addflot("Pseudoccior", "pht")
DrawPlot()
()
Query("filme")
time : GetQueryOutputValue()
outString = str(time) + " =
```

```
SetQueryFloatFormat("%g")
```

for dx in range(11):

```
# ZomePick(coord=(0.0251, 0.0248, 0.002))
# ZomePick(coord=(0.02518, 0.002) + float(dx) * 1e-5))
ZomePick(coord=(0.025180, 0.002) + float(dx) * 1e-5))
Pickval = 0.0
# print len(pick)
for entry in range(len(pick)):
    # print len(pick)
    # print gick(entry)
# print inte", time, "pick", pickval
if pickval < 290:
    print "SomeThing has gone wrong ", pick
    exit()
</pre>
```

```
outString = outString + str(pickVal) + " "
outString = outString + "\n"
historyFile.write(outString)
```

```
Query("spatialExtents", use_actual_data=1)
extents = cetQueryOutputValue()
stFExtents = str(extents)
splitExtents = r(esplit(')|\(|)\), stFExtents)
# print("Extents[a1])
# print(splitExtents[a1])
meltWidth = float(splitExtents[4]) - float(splitExtents[3])
print("Melt width = ", str(meltWidth))
outString = str(1) + " + str(meltWidth) + "\n"
```

 Other queries take additional arguments to determine the range of the output

widthFile.write(outString)

 Vector output is given in a braced format, which again needs processing

(日) (同) (三) (三)

Laboratory for Scientific

Computing





- Many of Vislt's options have been covered, hopefully some of them useful!
- This certainly isn't everything Vislt can do, but once you've played around with enough of the menus, additional things become more intuitive

イロト イポト イヨト イヨト

Laboratory for Scientific

Computing

э

- This presentation will be made available as reference
- Vislt 3.0 is due hopefully this information doesn't become obsolete then...

