ParaView

Tyson Whitehead

January 13, 2026

COURSE

CHAPTER

ONE

WEB ACCESS

Open OnDemand (OOD) web based interface is available for some clusters

- https://ondemand.sharcnet.ca
- https://ondemand.scinet.utoronto.ca

while others have JupyterHub (JH)

- https://jupyterhub.rorqual.alliancecan.ca
- https://jupyterhub.fir.alliancecan.ca

Either of these lets you start a desktop session

1.1 Open OnDemand

Sign in with your alliance username and password. While the functionality is the same, menu layouts differ between clusters. The nibi menus are as follows

files
: transfer files to/from cluster,

compute node
: stuff that runs on compute nodes (*i.e.*, in a job),

login node
: stuff that runs on login nodes (*i.e.*, immediately), and

my interactive sessions
: connect to/cancel interactive compute stuff.

The desktop session is persistent. Closing the tab just closes the tab. It does not end the session. To reattach go to *My Interactive Sessions* and click *Launch Desktop* again. Pick *Log out* under the *File* menu in the menu bar along the top of the actual desktop to shut it down cleanly. Click *Cancel* on *My Interactive Sessions* to kill it.

1.1.1 Login Node

Login is stuff that happens on a login node. It starts instantly and runs infinitely, but login nodes are shared and each user's total usage per login node is limited to

- total memory per-login: 16 GiB, and
- total cpu usage per-login: 6.

1.1.2 Compute Node

Compute is stuff that happens on compute nodes. It runs in the context of a job and has whatever resources and time the job has been allocated. Here are general guidelines for specifying your allocation

account
: ending in *gpu* if using a GPU (including hardware OpenGL) and *cpu* otherwise,

time
: 8hrs or less to run in the interactive queue for faster start,

computers
: one unless explicitly starting a multinode MPI process with `srun` or `mpirun`,

cores
: enough cores so memory/core is 12G or less (regular memory queue) for faster start,

memory per computer
: at least 8GB for a desktop session (see cores note about keeping memory/core 12G or less), and

gpu
: if a GPU is required, the type the cluster has most of for faster start.

1.2 JupyterHub

Sign in with your alliance username and password. Enter your allocation details and click start. Here are general guidelines for specifying what your allocation

reservation
: *None* unless taking part in a course (not this one) with a reservations,

account
: ending in *gpu* if using a GPU (including hardware OpenGL) and *cpu* otherwise,

time
: 3hrs or less to run in the interactive queue for faster start,

cores
: enough cores so memory/core is 12G or less (regular memory queue) for faster start,

memory
: at least 8GB for a desktop session (see cores note about keeping memory/core 12G or less),

oversubscription
: select if possible (shares CPUs with other users) for faster start,

gpu configuration
: if a GPU is required, the type the cluster has most of for faster start.

user interface
JupyterLab (*Jupter Notebook* is the older version before the renaming).

Once JupyterLab starts, click the *Mate Desktop* icon in the launcher window. This will open a desktop session in now browser tab.

The desktop session is persistent. Closing the tab just closes the tab. It does not end the session. Clicking the *Desktop* icon again will just reopen the existing one. To close it pick *Log out* in the menu bar along the top of the Desktop. The same is true of the entire JupyterLab session. Pick *Log out* under the *File* menu in the menu bar along the top to end it

1.3 Running Stuff

To get a terminal pick *Mate terminal* under *Applications* followed by *System Tools* in the menu bar along the top. The Compute Canada (the old name for the alliance) stack is loaded by default.

1.3.1 Clipboard

Browser limits remote access to clipboard for security reasons. With OOD, most modern browsers, appart from Firefox, will prompt for permission after which cutting and pasting between the desktop session and you desktop will just work.

For JupyterHub or OOD with Firefox, you have to first paste in or copy out of an intermediate cliboard. Under JupyterHub the intemediate clipboard is accessed at the top of the browser window (click on *Remote Clipboard*) and for OOD it is accessed on the right-hand side of the browser window (click on the pull out tab and then the clipboard icon).

1.3.2 Accelerated OpenGL

It is possible to use hardware accelerated OpenGL via VirtualGL if a whole GPU (i.e., not a multi-instance GPU) has been allocated. This can help applications showing poor interactivity when processing large scene graphs.

VirtualGL is enabled by default on OOD on nibi. For other clusters or JupyterHub it is necessary to start the program as follows

```
vglrun -d egl PROGRAM [ARGS ...]
```

1.3.3 JupyterLab Modules

JupyterLab is a python program, so it reloads some python modules that would not normally be loaded by default. If these conflict with other modules you need, just unload them or do a

```
module reset
```

before loading your desired modules.

CHAPTER

TWO

PARAVIEW

2.1 Personal computer

Available for Windows, OS X, and Linux

2.1.1 Windows and OS X

Download latest version from ParaView website

- `MPI` means in can use multiple CPUs (non-MPI is available on Windows so you can use it and not have to install the separate MS-MPI package unless you want to)
- `AMD64` and `x86_64` are synonyms that both means 64bit version for AMD or Intel chips (AMD is just a reference to the fact that AMD developed the x86_64 instruction set)
- `arm64` means 64bit version for ARM chips (currently just the Mac M1)
- older version may work if newer fails on older OS or graphics card

2.1.2 Linux

Install with your package manager (can also download as above)

- Debian and Ubuntu: `sudo apt-get install paraview`
- Fedora: `sudo yum install paraview`

2.2 Clusters

Available in the Compute Canada `StdEnv` (`StdEnv/2020` also requires loading the `gcc` module)

```
$ module load paraview
```

Then you can just run it

```
$ paraview
```

2.2.1 VirtualGL

VirtualGL (GPU accelerated OpenGL) is enabled by default on nibi if an appropriate GPU was allocated. Recent versions of ParaView will give the error *ISPCRT Error (1): Fail to load libispcrt_device_cpu.so library*. Working around this requires disabling the VirtualGL *libdlfaker.so* (dynamic linker faker) `LD_PRELOAD` when starting it

```
$ LD_PRELOAD=${LD_PRELOAD/libdlfaker.so/} paraview
```

JupyterLab desktop does not enable VirtualGL by default. If you have allocated an appropriate GPU you can manually enable it by running it as

```
$ vglrun -d egl paraview
```

If you get the aforementioned error *ISPCRT Error (1): Fail to load libispcrt_device_cpu.so library*, then you need to tell `vglrun` to disable the VirtualGL *libdlfaker.so*

```
$ vglrun -nodl -d egl paraview
```

CHAPTER

THREE

TUTORIAL

Working through the ParaView tutorial is one of the quickest and easiest ways to get up-to-speed on ParaView.

- basic usage (what we will be going over)
- batch python scripting
- visualizing large models

The data files are supposed to come with recent ParaView versions. They can also be downloaded from the ParaView files data directory

```
$ wget -d https://www.paraview.org/files/data/ParaViewTutorialData.zip
$ unzip ParaViewTutorialData.zip
```

or, on graham, copied directly from my account to yours

```
$ cp -r /home/tyson/ParaViewTutorialData .
```

3.1 Basis of visualization

- map raw data to visual data
- spacial and temporal data
- topology and types of grids

3.2 User interface

- menu bar
- tool bars
- pipeline browser
- properties panel
- view

3.3 Basic interface

- creating a source
- interacting with a 3d view
- modifying visualization parameters (filter, display, view)
- undo and redo (regular vs camera)

3.4 Loading data

- opening file (selecting which variables to load)
- representation and field coloring
- scaling

3.5 Filters

- selecting filters (toolbar, menu, and search)
- applying a filter (contours, slices)

3.6 Multiview

- creating a multiple views
- linking cameras
- re-arranging the views

3.7 Vector visualization

- streamlines
- tubes and glyphs
- surface LIC

3.8 Volume rendering

- Enabling
- Transfer function

3.9 Time

- Loading temporal data
- Temporary interpolation

3.10 Output

- Annotations
- Screenshots
- Video

3.11 Plotting

- histogram plot
- plot over a line in space
- plot over a curve
- plot over time

3.12 Selections

- Query and view based selections
- Data vs spatial selections
- Selection labels
- Extract selection and spreadsheet

3.13 Animation

- Properties
- Keyframes
- Tracks

CHAPTER

FOUR

EXERCISES

4.1 Plotting Exercise

The purpose of this exercise is to practice using paraview by duplicating this visualization of the temperature above disk, and this plot of pressure and temperature above disk.. The steps required to do this are as follows:

- Load the `disk_out_ref.ex2` file.
- Clip it with a plane through the origin in the +X direction.
- Display the temperature and the grid on it.
- Change the color map to black-body radiation to better represent temperature and label the scale `Temperature`.
- Plot pressure and temperature one grid point above the center point (note directions in filter `display` settings regarding use of `ctrl` to snap plot line to grid points) with proper labels (remember you have to set different chart axis to separate scales).
- Add a text source to add `Submission: <Your Name>` (note that you can work around an export scene centering bug by switching to coordinates after positioning it using the window location).
- Try saving both a screenshot and exporting a pdf scene and then zoom in on them both to see what the difference is (on gra-vdi you can use the browser to view them by going to `file:///home/<username>/` and using `ctrl-+` to zoom in and `ctrl+0` to reset).
- Export a pdf scene of the clipped `disk_out_ref.ex2` as `disk.pdf` and submit it.
- Export a pdf scene of the plot as `plot.pdf` and submit it.

4.2 Time Exercise

The purpose of this exercise is to practice using paraview by duplicating this interpolated movie showing the can being crushed. The steps required to do this are as follows:

- Skim the section in the paraview tutorial on time (pg. 51-56) and load the `can.ex2` file.
- Color it by the equivalent plastic strain `eqps` scaled across all steps, adjust the view to see the inside of the can, and use the VCR toolbar to step through/play the data series.
- Enable the animation view, set the mode to `sequence` with `100` frames, and then add a `temporal interpolator` filter to interpolate the 100 from the original 42.
- Add a `annotate time` source to put `<Your Name>: <Time>` in the corner.
- Save an 5 second (5s x 20fps = 100frames) 1024x512 animation video called `crush.avi` (or `crush.mp4`, `crush.ogv`, etc.) and submit it.

4.3 Selection Exercise

The purpose of this exercise is to practice using paraview by duplicating this distribution plot of equivalent plastic strain over time in top 20% of cells at end. The steps required to do this are as follows:

- Skim the section in the paraview tutorial on selection (pg. 63-69) and load the `can.ex2` file.
- Color it by the equivalent plastic strain `eqps` across all time steps, adjust the view to see the inside of the can, and advance to the final frame.
- Enable the `find data` view, use it to find the maximum 'eqps' value from the `can.ex2 data producer`, calculate 80% of this value by hand, and change in to find all cells with `eqps` greater or equal to this value.
- Plot the distribution of `eqps` values for this selection over time by adding a `plot selection over time` (the `plot selection over time` is a shortcut for manually adding the filter), and think about why this only shows a distribution for the final six frames.
- Select the `freeze` option to lock the selection to set of cells selected on the final step (instead of the set of cell matching the criteria on each frame, answering the prior question), and update the `plot selection over time` fliter with this selection via the `copy active selection` filter `properties` button.
- Cleanup the plot label, use the view properties to label the chart `Submission: <Your Name>`, export a scene pdf `selection.pdf`, and upload it as your submission.

Note that not all pdf viewers properly show the quartile region (darker grey) in the generated pdf file for this exercise, so do not be concerned if appears to be missing in your pdf scene export.

4.4 Animation Exercise

The purpose of this exercise is to practice using paraview by duplicating this animated movie of MRI head slices. The steps required to do this are as follows:

- Skim the section in the paraview tutorial on animation (pg. 70-76) and load the `headsq.vti` file.
- Add a clip filter in the -Z direction, display the `Scalars_` property, and orient the display with Y up and X left.
- We want an animation showing all the Z slices of the data in this file, so we look at the `information` tab (beside the `properties` tab) for the `headsq.vti` source to see that the Z bound goes from 0 to 186 and the extent goes from 0 to 93 (i.e., there are 94 X-Y planes of data, with a new plane every 2 units of Z).
- Adding a 2s delay, at 20fps, the 0 to 93 extent would go from frame 40/20fps = 2s to frame (40+93)/20fps = 6.65s, so open the `animation` view, pick the `sequence` mode and set the end time to `6.65`s and the number of frames to 40+94 = `134` (i.e, giving a mapping of frames 0-39 and 40-133 to to times 0-1.95s and 2-6.65s).
- Select the `clip type - origin (2)` property in the clip filter as the parameter to vary below the timeline and press the blue + button to add it.
- Double click on the default ramp mapping of 0 to 255, add an intermediate point with the `new` button and set it to be `185.99` from `0`s to `2`s and `185.99` to `-0.01` from `2`s to `6.65`s.
- View the `headsq.vti` source properties while using the green VCR control arrows it the toolbar to single step through a few timesteps to verify the clipping plane steps through the data extent as expected (i.e., starting at 2s, it should go 183.99, 181.99, 179.99, . . . , 0.99, -0.01).
- Add an annotated time filter, set the `scale` to `20`, and the `shift` to `-40` (confusingly the scale is applied before the shift despite their order) then, after verifying it is making integer steps, set the format string to `<Your name>: {time:.0f}` (note the `:.0f` to specify zero decimal places) in order to show the data slice.
- Save an 20fps 512x512`animation video called`slices.avi(orslices.mp4, slices.ogv`, etc.) and submit it.

CHAPTER

FIVE

SEARCH

- search