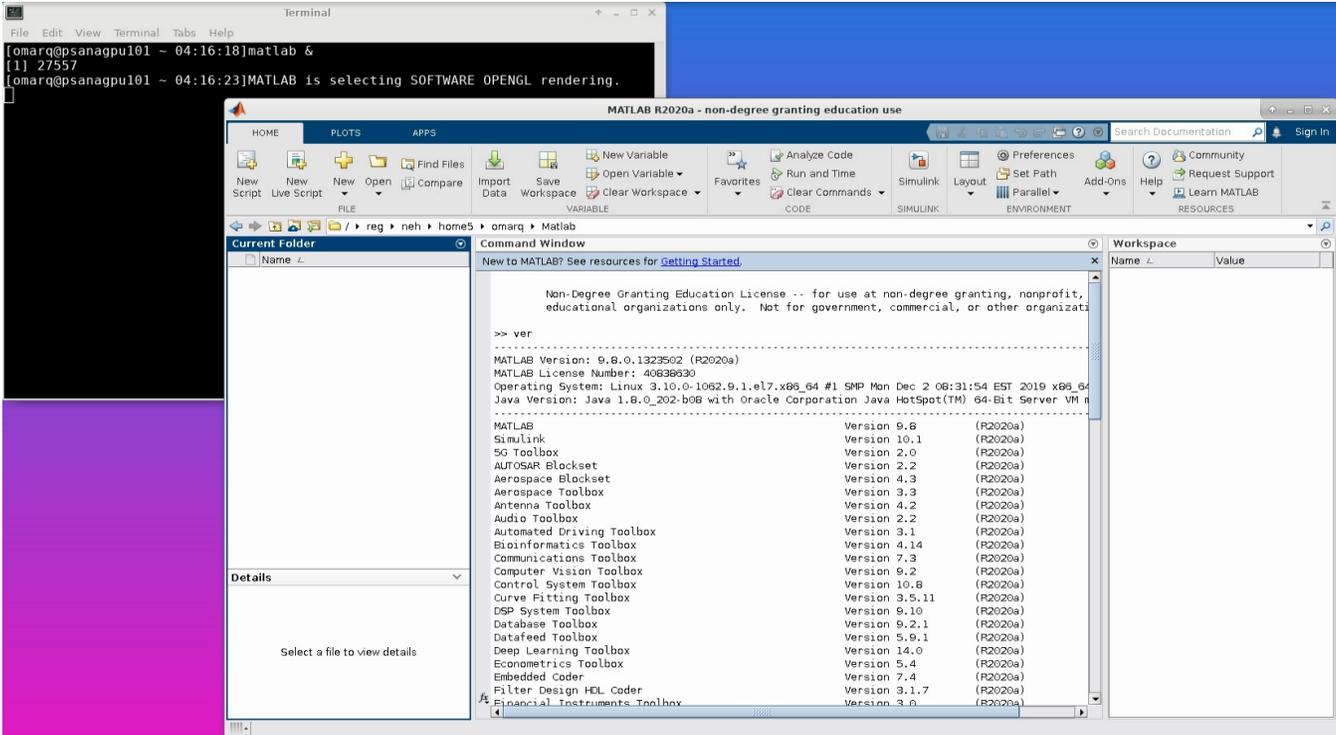


Matlab

You can run matlab from any node in the psana pool:

```
ssh psana
matlab &
```



Current Version

Current Release is **R2020a**.

If the *matlab* command is not accessible, please do the following:

```
export PATH=/usr/local/bin:$PATH
```

Parallel Toolbox

The parallel toolbox allows you to make use of multiple cores on your local node (for example the `parfor` command will launch matlab workers on the multiple cores that are available).

MathWorks has a nice introduction to use of the `parfor` command: <http://www.mathworks.com/help/distcomp/getting-started-with-parfor.html>

The following is meant to serve as a warning, and is an example of the misuse of `parfor` (i.e. it will not correctly calculate Fibonacci numbers, as one might expect):

```
f = zeros(1,50);
f(1) = 1;
f(2) = 2;
parfor n = 3:50
    f(n) = f(n-1) + f(n-2);
end
```

Launching Matlab in Background

If you have a matlab script named `myscript.m` in your current directory, a good way to launch matlab in the background is

```
nohup matlab -nodesktop -nosplash -r "try; myscript; quit; catch; end" > myoutput.log 2>&1 &
```

By using the unix utility `nohup`, (short for no hang up) you will be able to close the terminal window you are using and logout without interrupting the matlab process. Note that we do not add the `.m` extension to `myscript`. By adding the command `quit` if the script is successful, and the `try catch` clause if there is a failure, we ensure that the matlab process will close properly when finished. This is important for managing the matlab licenses. You can check on the output by looking at the file `myoutput.log` that you are capturing output to, as well as errors to (with the `2>&1` command).

Reattaching to a Running Session

Users doing long computations in interactive sessions will typically lock their machine overnight and return to the Matlab session the next day. If you are experiencing network connection problems, the matlab session will typically be killed when the connection is lost. In addition to using `nohup` and batch mode as above, one can use the unix utility `tmux` to reconnect to a matlab session that is run in a terminal window (as opposed to in its typical graphical window). For instance,

```
tmux  
matlab --nosplash --nodesktop
```

Let's suppose this was done on the host `psanaxyz`. If you lose the connection to `psanaxyz`, you can go back to that node and reattach:

```
ssh psanaxyz  
tmux attach
```