WORKSHOP: Parallel Computing with MATLAB (Part II)

WESTGRID

Raymond Norris
Application Engineer, MathWorks

July 21, 2021

Agenda

- Part I – Parallel Computing with MATLAB on the Desktop
  - Parallel Computing Toolbox
  - MATLAB Online
- Part II – Scaling MATLAB to Compute Canada clusters
  - MATLAB Parallel Server
  - VNC

https://docs.computecanada.ca/wiki/MATLAB

Agenda

- Part I – Parallel Computing with MATLAB on the Desktop
  - Parallel Computing Toolbox
  - MATLAB Online
- Part II – Scaling MATLAB to Compute Canada clusters
  - MATLAB Parallel Server
  - VNC

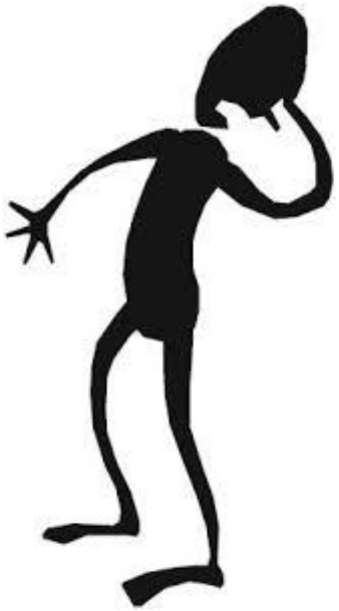
https://docs.computecanada.ca/wiki/MATLAB

Scaling MATLAB to Compute Canada clusters

- Accessing and running MATLAB on local HPC clusters
- Running parallel and multi-node MATLAB jobs

A note about today's workshop…

- The workflow and examples are about process, not performance



Accessing and running MATLAB on local HPC clusters (1)

- Two options
  - ssh
    - Command line interface
    - Useful for either low-bandwidth or automation
  - VNC
    - Graphical interface
    - https://docs.computecanada.ca/wiki/VNC

Connect to Compute Canada resources

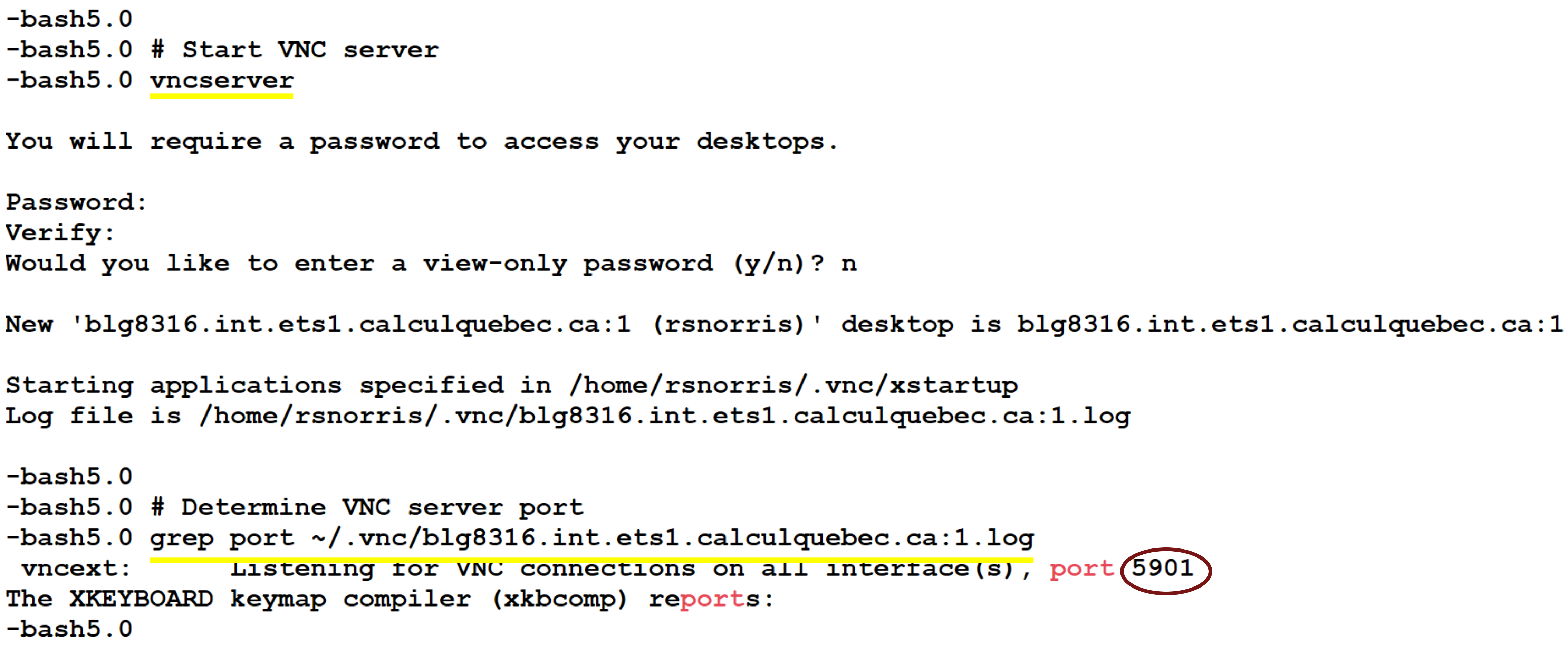
```
-bash5.0
-bash5.0 # Connect to login node
-bash5.0 ssh rsnorris@beluga.computecanada.ca
rsnorris@beluga.computecanada.ca's password:
Last login: Tue Jul 20 01:56:59 2021 from c-24-62-20-69.hsd1.ma.comcast.net
################################################################################
  _       __ _
 | |__   /_/| |_   _  __ _  __ _   Bienvenue sur Béluga / Welcome to Béluga
 | '_ \ / _ \ | | | |/ _` |/ _` |
 | |_) |  __/ | |_| | (_| | (_| |  Aide/Support:    support@calculcanada.ca
 |_.__/ \___|_|\__,_|\__, |\__,_|  Globus endpoint: computecanada#beluga-dtn
                     |___/         Documentation:   docs.calculcanada.ca

################################################################################
-bash4.2
-bash4.2 # Allocate compute node
-bash4.2 salloc --time=3:00:00 --cpus-per-task=8 --mem=32gb
salloc: Pending job allocation 22820987
salloc: job 22820987 queued and waiting for resources
salloc: job 22820987 has been allocated resources
salloc: Granted job allocation 22820987
salloc: Waiting for resource configuration
salloc: Nodes blg8316 are ready for job
-bash5.0
-bash5.0 export XDG_RUNTIME_DIR=${SLURM_TMPDIR}
-bash5.0
```

Beluga

Cedar

Graham

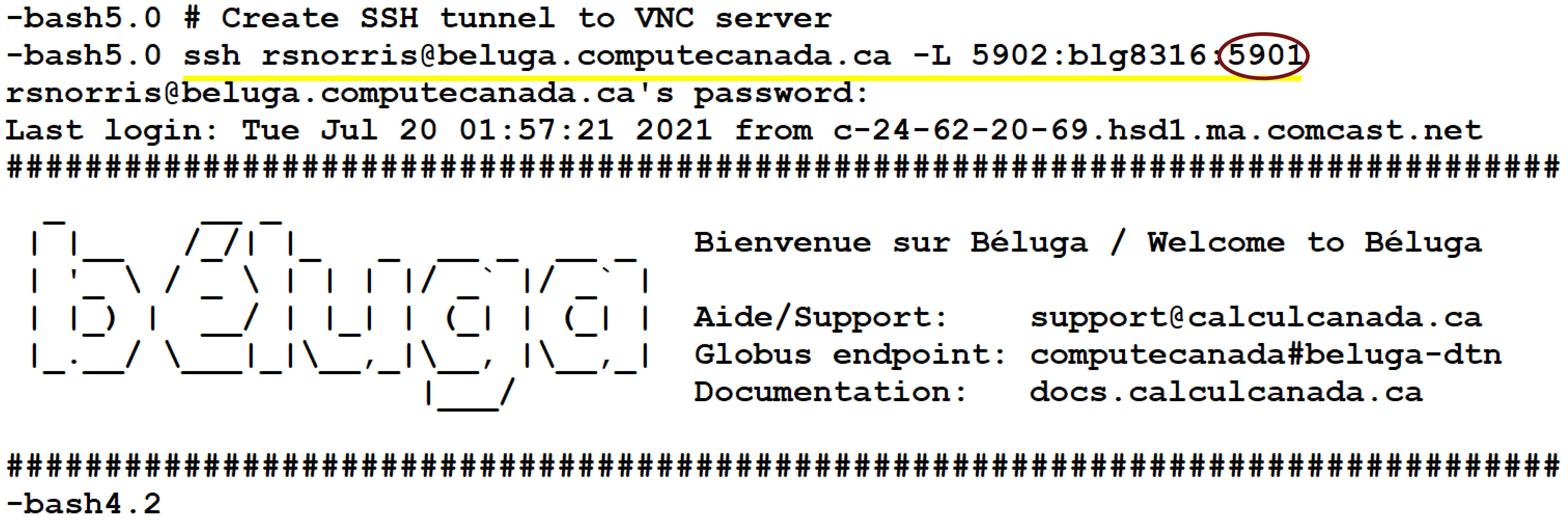
Start VNC session



```
-bash5.0
-bash5.0 # Start VNC server
-bash5.0 vncserver

You will require a password to access your desktops.

Password:
Verify:
Would you like to enter a view-only password (y/n)? n

New 'blg8316.int.ets1.calculquebec.ca:1 (rsnorris)' desktop is blg8316.int.ets1.calculquebec.ca:1

Starting applications specified in /home/rsnorris/.vnc/xstartup
Log file is /home/rsnorris/.vnc/blg8316.int.ets1.calculquebec.ca:1.log

-bash5.0
-bash5.0 # Determine VNC server port
-bash5.0 grep port ~/.vnc/blg8316.int.ets1.calculquebec.ca:1.log
 vncext:      Listening for VNC connections on all interface(s), port 5901
The XKEYBOARD keymap compiler (xkbcomp) reports:
-bash5.0
```

Create SSH tunnel

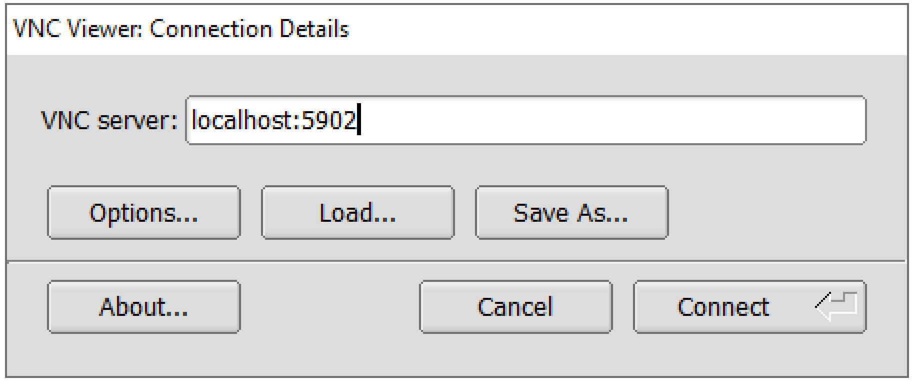
MUST KEEP THIS SSH SESSION CONNECTED



```
-bash5.0 # Create SSH tunnel to VNC server
-bash5.0 ssh rsnorris@beluga.computecanada.ca -L 5902:blg8316:5901
rsnorris@beluga.computecanada.ca's password:
Last login: Tue Jul 20 01:57:21 2021 from c-24-62-20-69.hsd1.ma.comcast.net
################################################################################

  _       __ _
 | |__   /_/| |_   _  __ _  __ _   Bienvenue sur Béluga / Welcome to Béluga
 | '_ \ / _ \ | | | |/ _` |/ _` |
 | |_) |  __/ | |_| | (_| | (_| |  Aide/Support:    support@calculcanada.ca
 |_.__/ \___|_|\__,_|\__, |\__,_|  Globus endpoint: computecanada#beluga-dtn
                     |___/         Documentation:    docs.calculcanada.ca

################################################################################
-bash4.2
```

Open VNC session

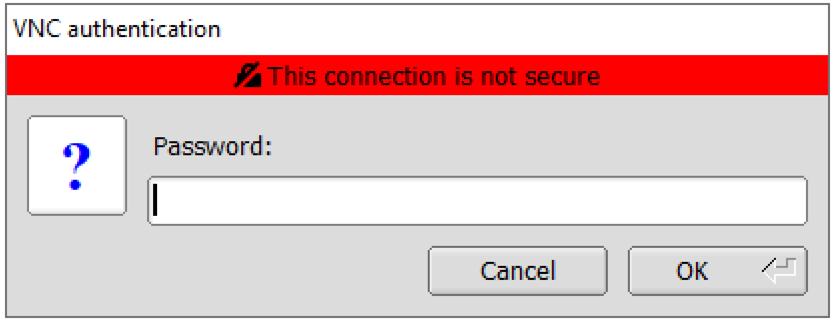
Port from SSH tunnel



VNC Viewer: Connection Details

VNC server: localhost:5902

VNC password,
not cluster password

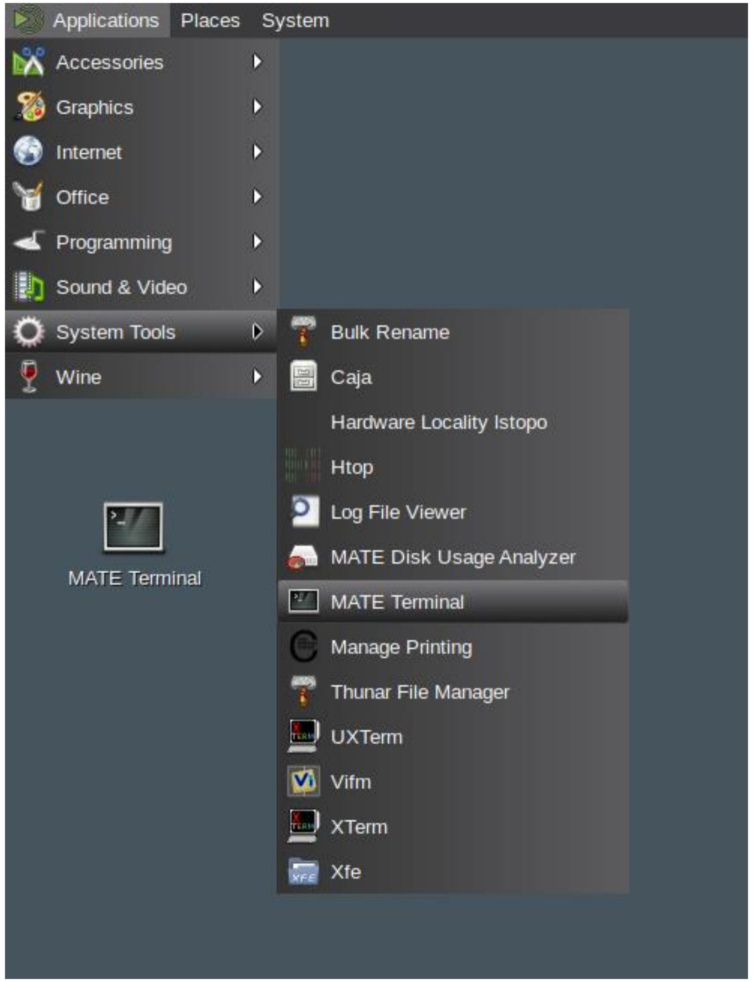


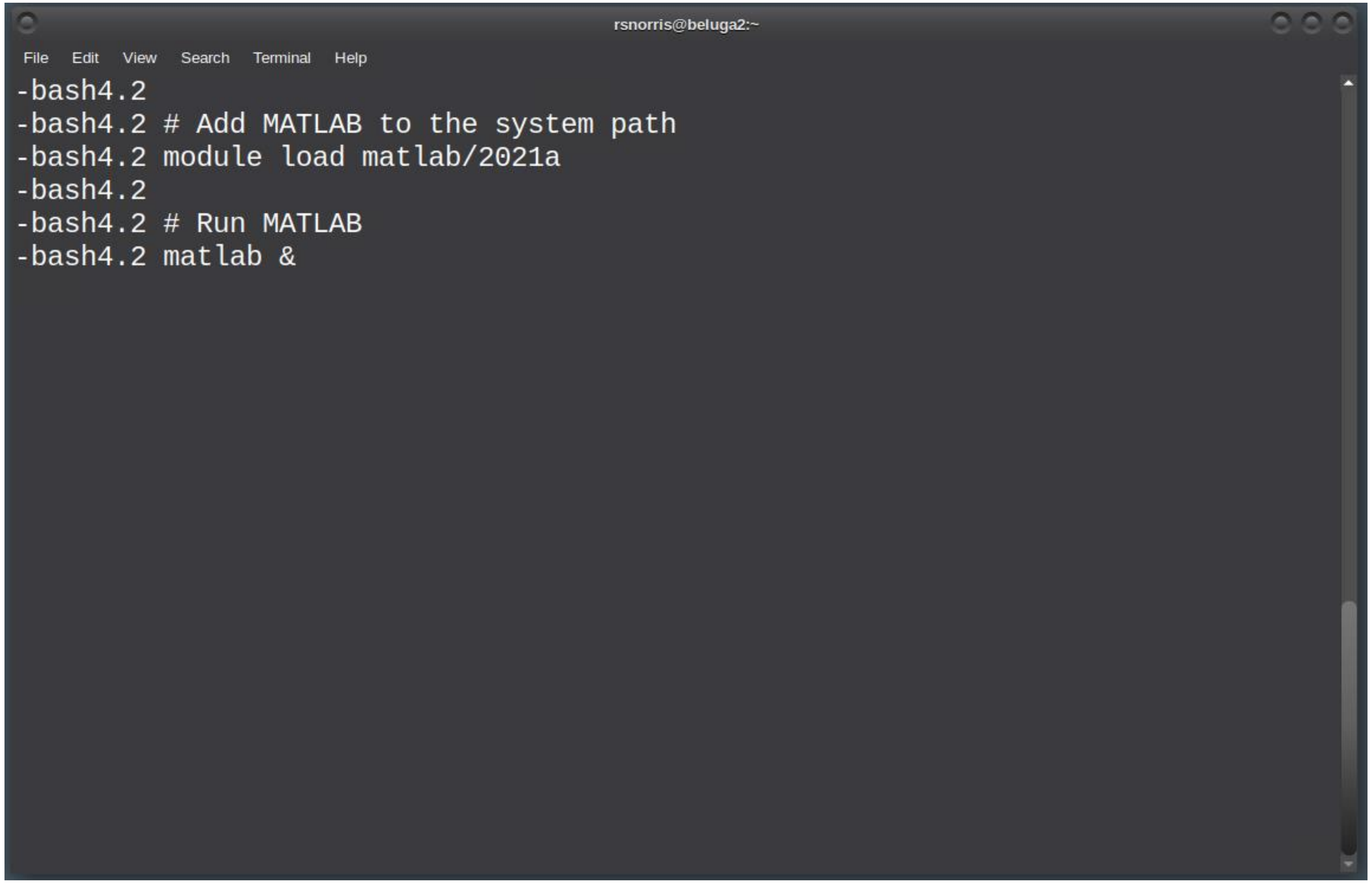
VNC authentication

This connection is not secure

Password:

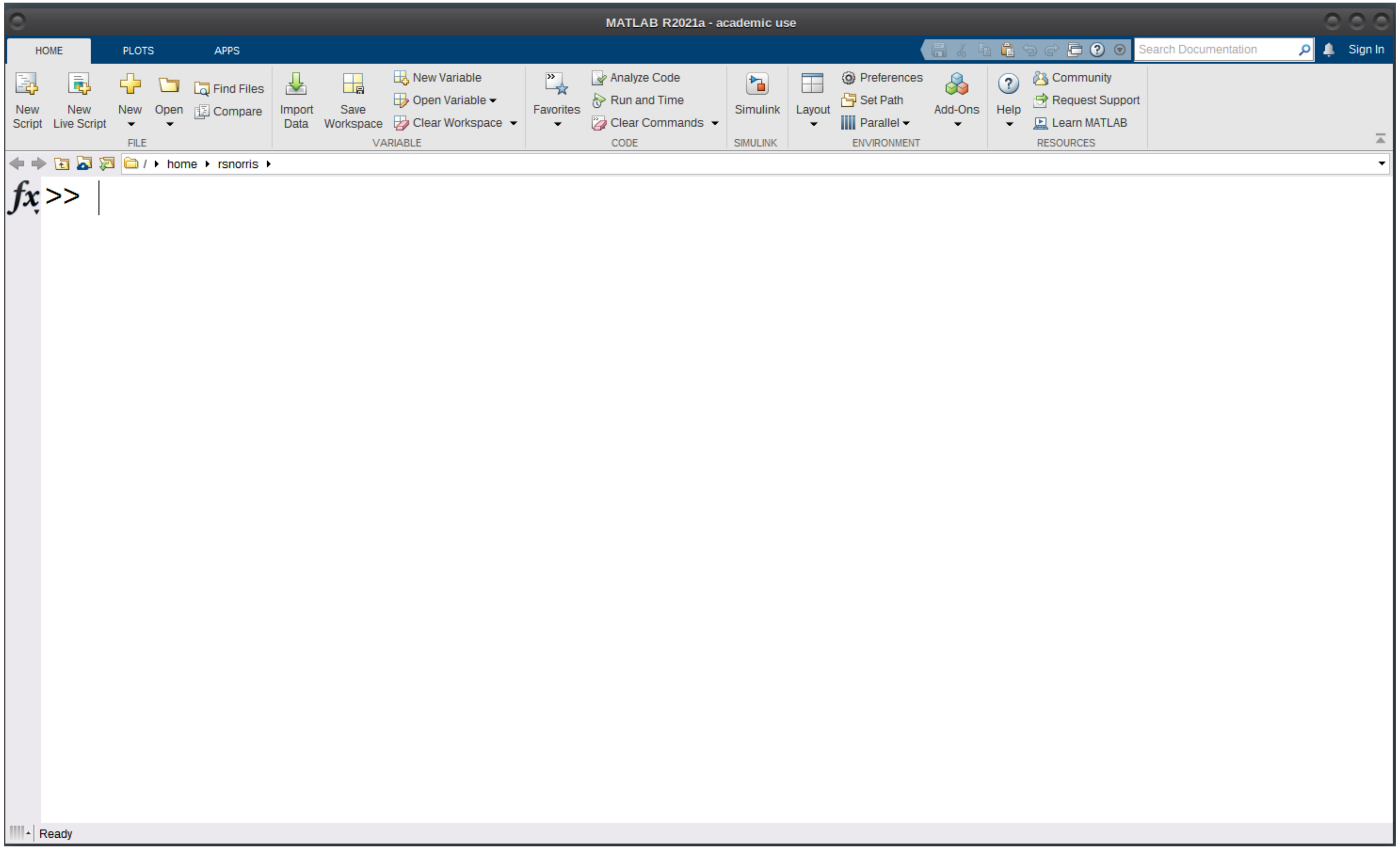
Start MATLAB



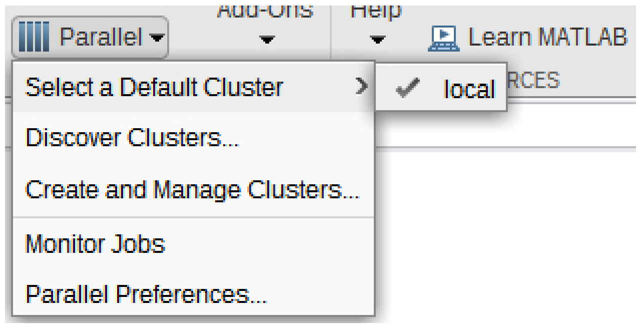


```
-bash4.2
-bash4.2 # Add MATLAB to the system path
-bash4.2 module load matlab/2021a
-bash4.2
-bash4.2 # Run MATLAB
-bash4.2 matlab &
```

Start MATLAB



Parallel MATLAB – Single Node



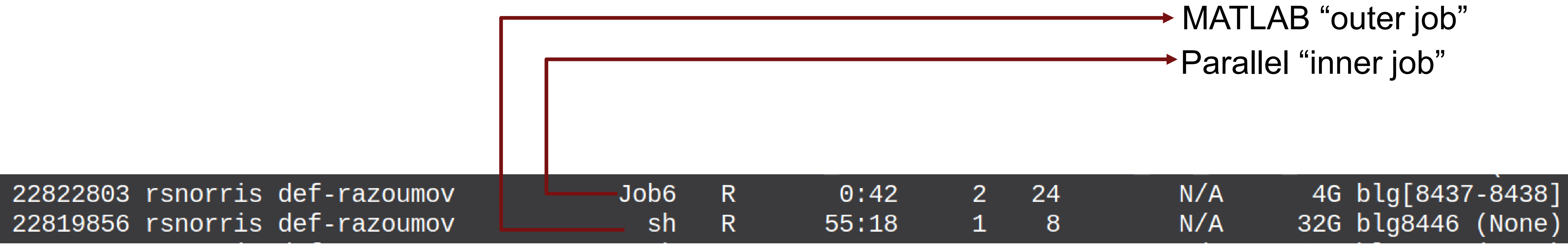
```
>> maxNumCompThreads
ans =
     8
>>
>> parpool('local');
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 8).
>>
>> tic, parfor idx = 1:160, pause(3), end, toc
Elapsed time is 61.334370 seconds.
>>
```

```
-bash4.2 # Allocate compute node
-bash4.2 salloc --time=3:00:00 --cpus-per-task=8 --mem=32gb
salloc: Pending job allocation 22820987
```

Parallel MATLAB – Multi-node (1)

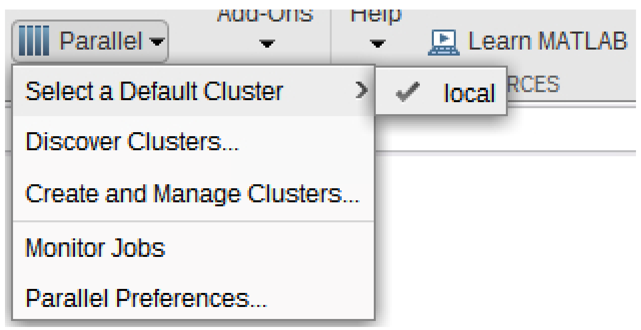
- In order to run a multi-node MATLAB job, MATLAB will generate and submit a new Slurm job
  - Executed during any “job launcher”
    - `parpool`, `batch`, `createJob`
  - Run asynchronously while MATLAB session is running, except `parpool`
  - True regardless if we’re running MATLAB desktop or a Slurm job script

- Need to generate a new profile for Compute Canada cluster



```
22822803 rsnorris def-razoumov        Job6    R       0:42     2    24         N/A      4G blg[8437-8438]
22819856 rsnorris def-razoumov          sh    R      55:18     1     8         N/A     32G blg8446 (None)
```

local profile



“How does MATLAB know about Compute Canada clusters?”

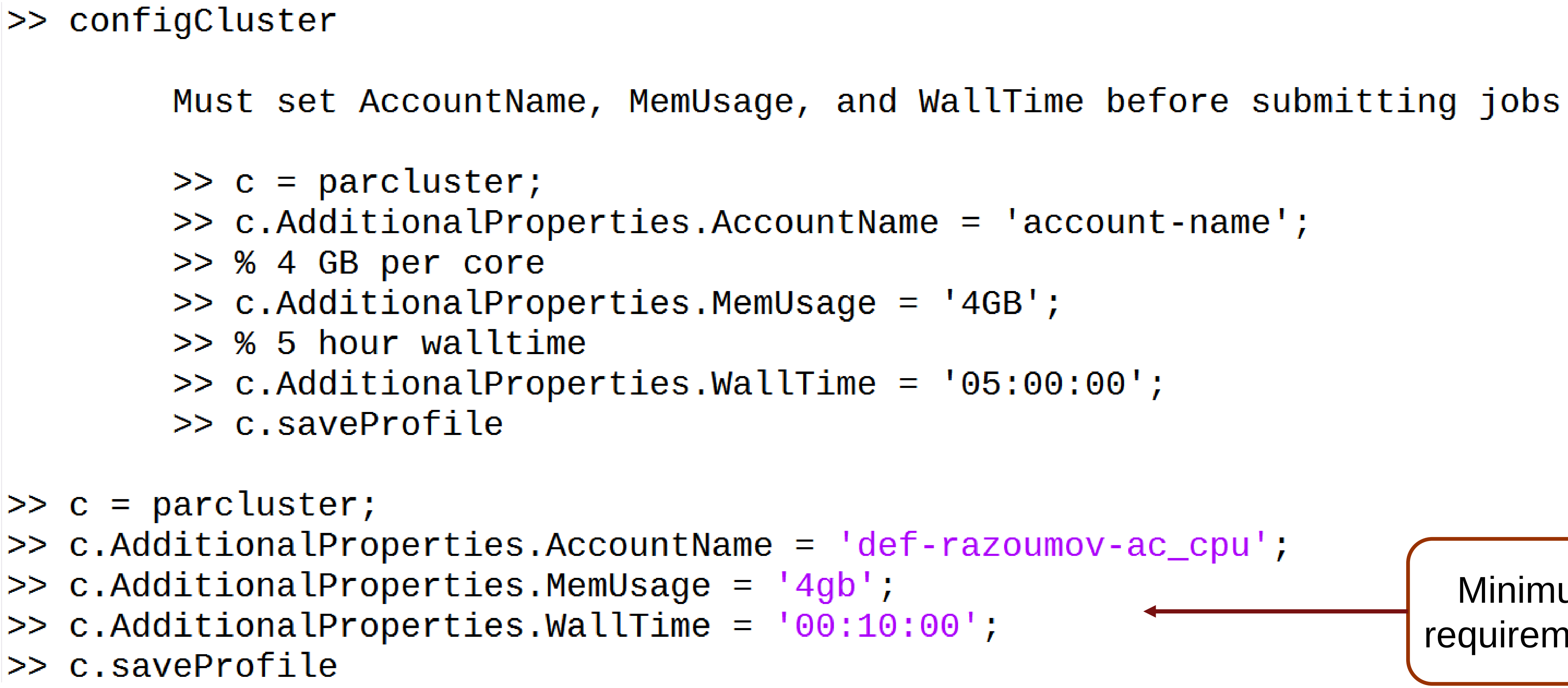
MATLAB Support Package for Compute Canada

- The MATLAB Support Package for Slurm can be found only all three clusters in

```
/home/rsnorris/Public/matlab
% cp -frp /home/rsnorris/Public/matlab ~
```

- Plan is to eventually move it to $matlabroot and update Compute Canada MATLAB wiki page

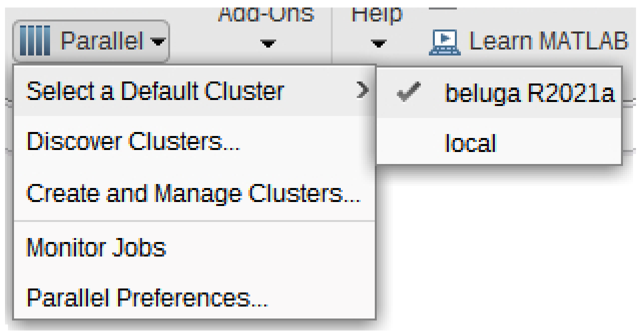
Configure MATLAB to create a Compute Canada profile



```
>> configCluster

        Must set AccountName, MemUsage, and WallTime before submitting jobs

        >> c = parcluster;
        >> c.AdditionalProperties.AccountName = 'account-name';
        >> % 4 GB per core
        >> c.AdditionalProperties.MemUsage = '4GB';
        >> % 5 hour walltime
        >> c.AdditionalProperties.WallTime = '05:00:00';
        >> c.saveProfile

>> c = parcluster;
>> c.AdditionalProperties.AccountName = 'def-razoumov-ac_cpu';
>> c.AdditionalProperties.MemUsage = '4gb';
>> c.AdditionalProperties.WallTime = '00:10:00';
>> c.saveProfile
```

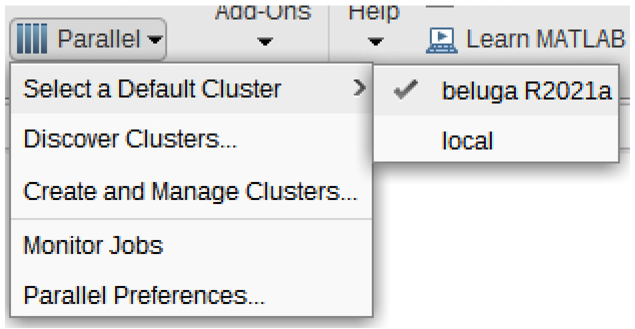
Minimum requirements

New Compute Canada profile



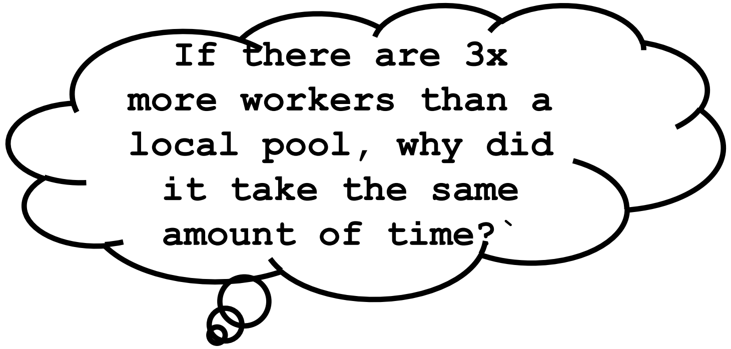
Only call `configCluster` once

Parallel MATLAB – Multi-node (2)

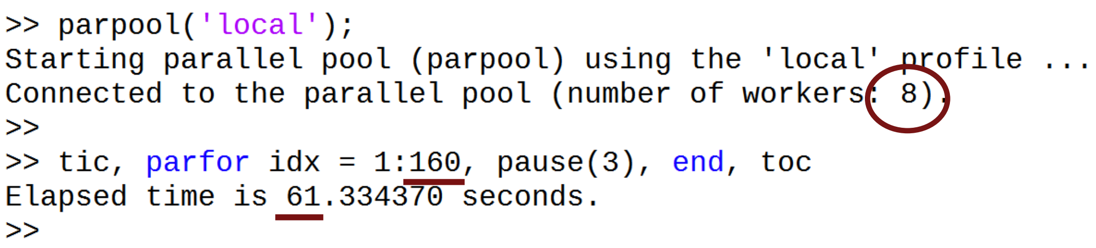


```
>> % Get a handle to the cluster
>> c = parcluster;
>>
>> % Start parallel pool
>> pool = c.parpool(24);
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...
additionalSubmitArgs =
    '--ntasks=24 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov
Connected to the parallel pool (number of workers: 24).
>>
>> tic, parfor idx = 1:480, pause(3), end, toc
Elapsed time is 60.535884 seconds.
>>
```

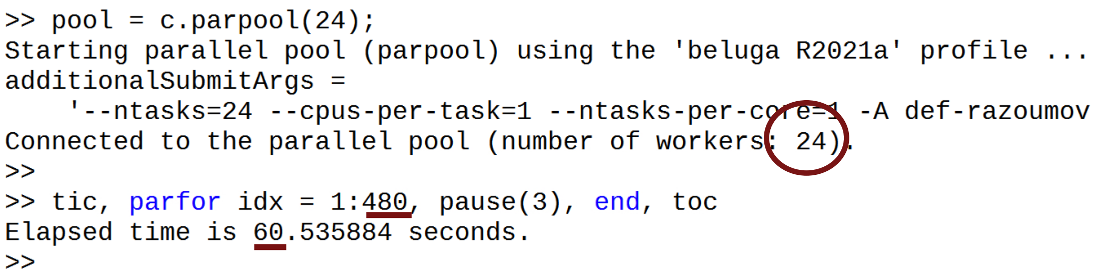
Where is the scaling?



If there are 3x more workers than a local pool, why did it take the same amount of time?`

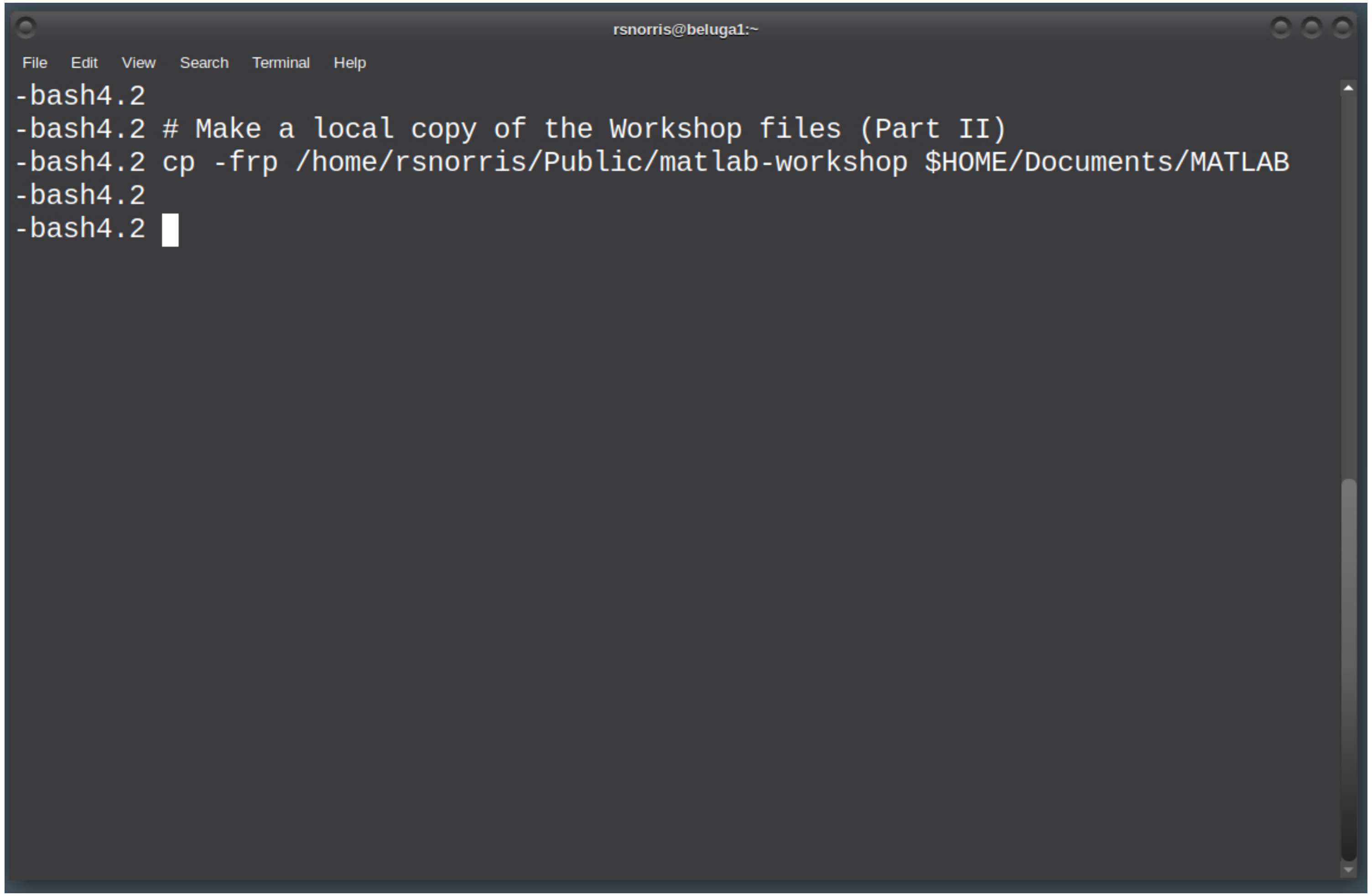


```
>> parpool('local');
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 8).
>>
>> tic, parfor idx = 1:160, pause(3), end, toc
Elapsed time is 61.334370 seconds.
>>
```



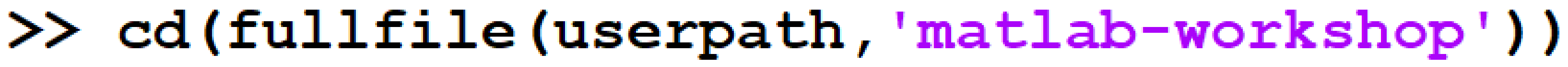
```
>> pool = c.parpool(24);
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...
additionalSubmitArgs =
    '--ntasks=24 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov
Connected to the parallel pool (number of workers: 24).
>>
>> tic, parfor idx = 1:480, pause(3), end, toc
Elapsed time is 60.535884 seconds.
>>
```

Download workshop files



```
-bash4.2
-bash4.2 # Make a local copy of the Workshop files (Part II)
-bash4.2 cp -frp /home/rsnorris/Public/matlab-workshop $HOME/Documents/MATLAB
-bash4.2
-bash4.2 
```

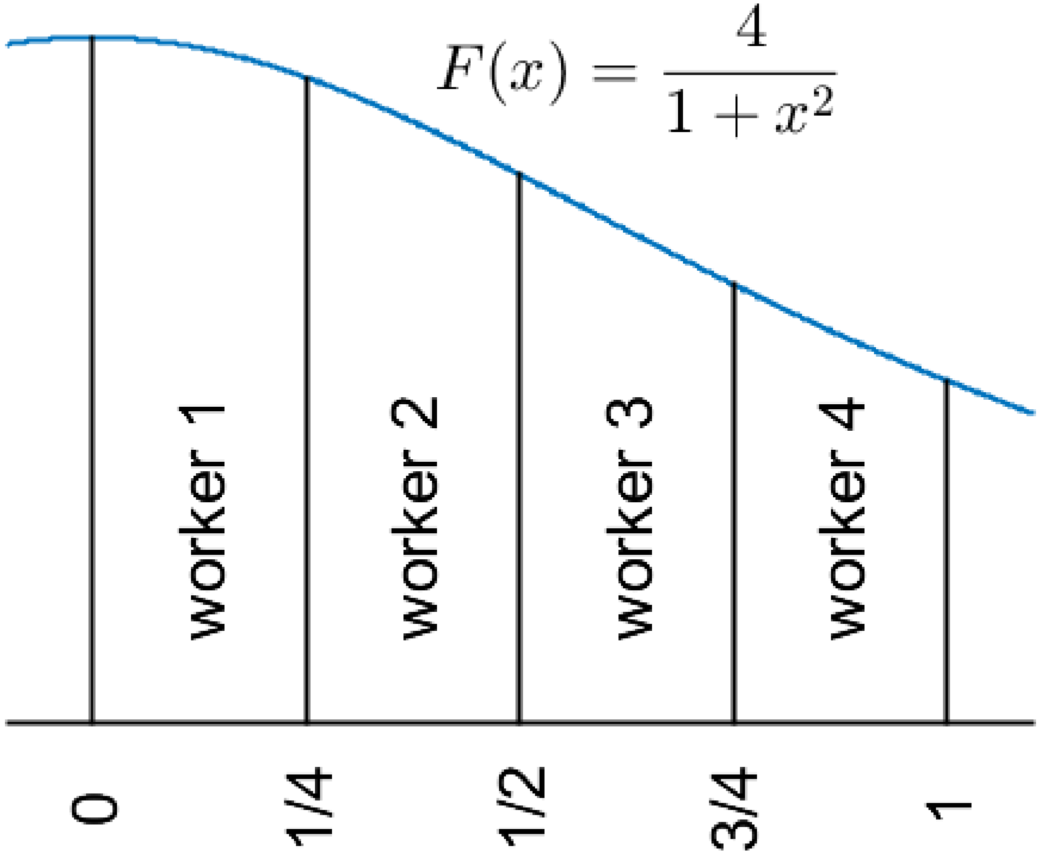
Change directories to workshop



```
>> cd(fullfile(userpath,'matlab-workshop'))
```

Exercise: Calculate π

$$\int_0^1 \frac{4}{1+x^2}\,dx = 4(atan(1) - atan(0)) = \pi$$

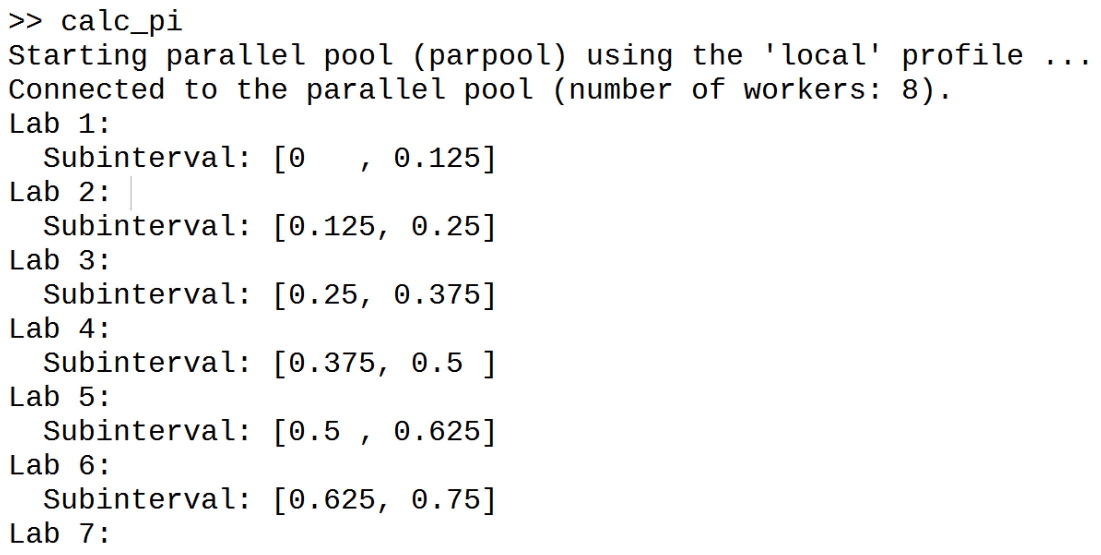


Calculate π

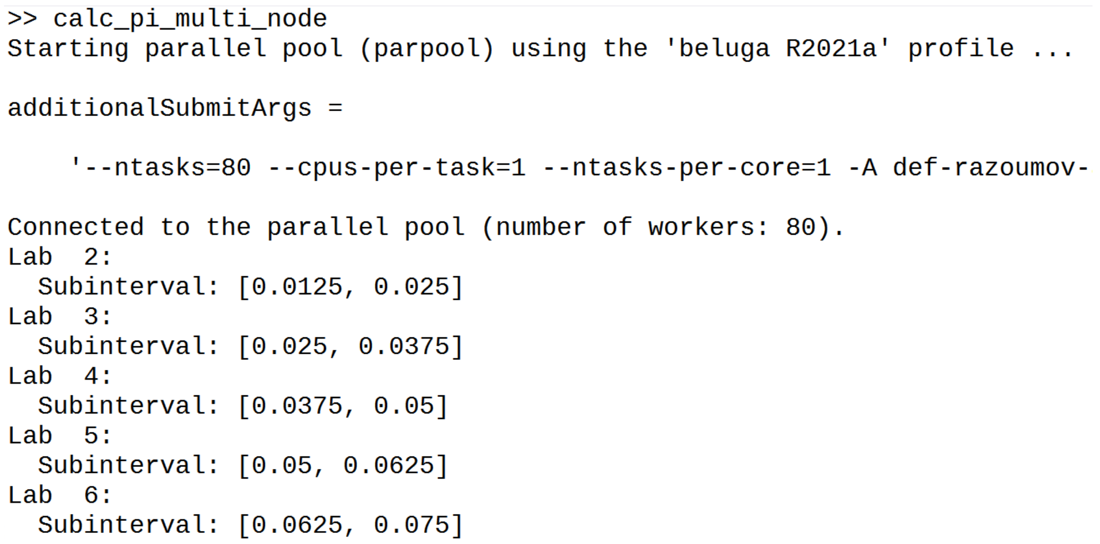
```
function calc_pi

c = parcluster('local');

% Query for available cores (assume either Slurm or PBS)
sz = str2num([getenv('SLURM_CPUS_PER_TASK') getenv('PBS_NP')]); %#ok<ST2NM>
if isempty(sz), sz = maxNumCompThreads; end

c.parpool(sz);

spmd
    a = (labindex - 1)/numlabs;
    b = labindex/numlabs;
    fprintf('Subinterval: [%-4g, %-4g]\n', a, b)

    myIntegral = integral(@quadpi, a, b);
    fprintf('Subinterval: [%-4g, %-4g]   Integral: %4g\n', a, b, myIntegral)

    piApprox = gplus(myIntegral);
end

approx1 = piApprox{1};  % 1st element holds value on worker 1
fprintf('pi           : %.18f\n', pi)
fprintf('Approximation: %.18f\n', approx1)
fprintf('Error        : %g\n',    abs(pi - approx1))


function y = quadpi(x)
%QUADPI Return data to approximate pi.

% Derivative of 4*atan(x)
y = 4./(1 + x.^2);
```

```
function calc_pi_multi_node

c = parcluster;
c.AdditionalProperties.MemUsage = '4gb';
c.AdditionalProperties.WallTime = '00:10:00';


c.parpool(80);

spmd
    a = (labindex - 1)/numlabs;
    b = labindex/numlabs;
    fprintf('Subinterval: [%-4g, %-4g]\n', a, b)

    myIntegral = integral(@quadpi, a, b);
    fprintf('Subinterval: [%-4g, %-4g]   Integral: %4g\n', a, b, myIntegral)

    piApprox = gplus(myIntegral);
end

approx1 = piApprox{1};  % 1st element holds value on worker 1
fprintf('pi           : %.18f\n', pi)
fprintf('Approximation: %.18f\n', approx1)
fprintf('Error        : %g\n',    abs(pi - approx1))


function y = quadpi(x)
%QUADPI Return data to approximate pi.

% Derivative of 4*atan(x)
y = 4./(1 + x.^2);
```

Results

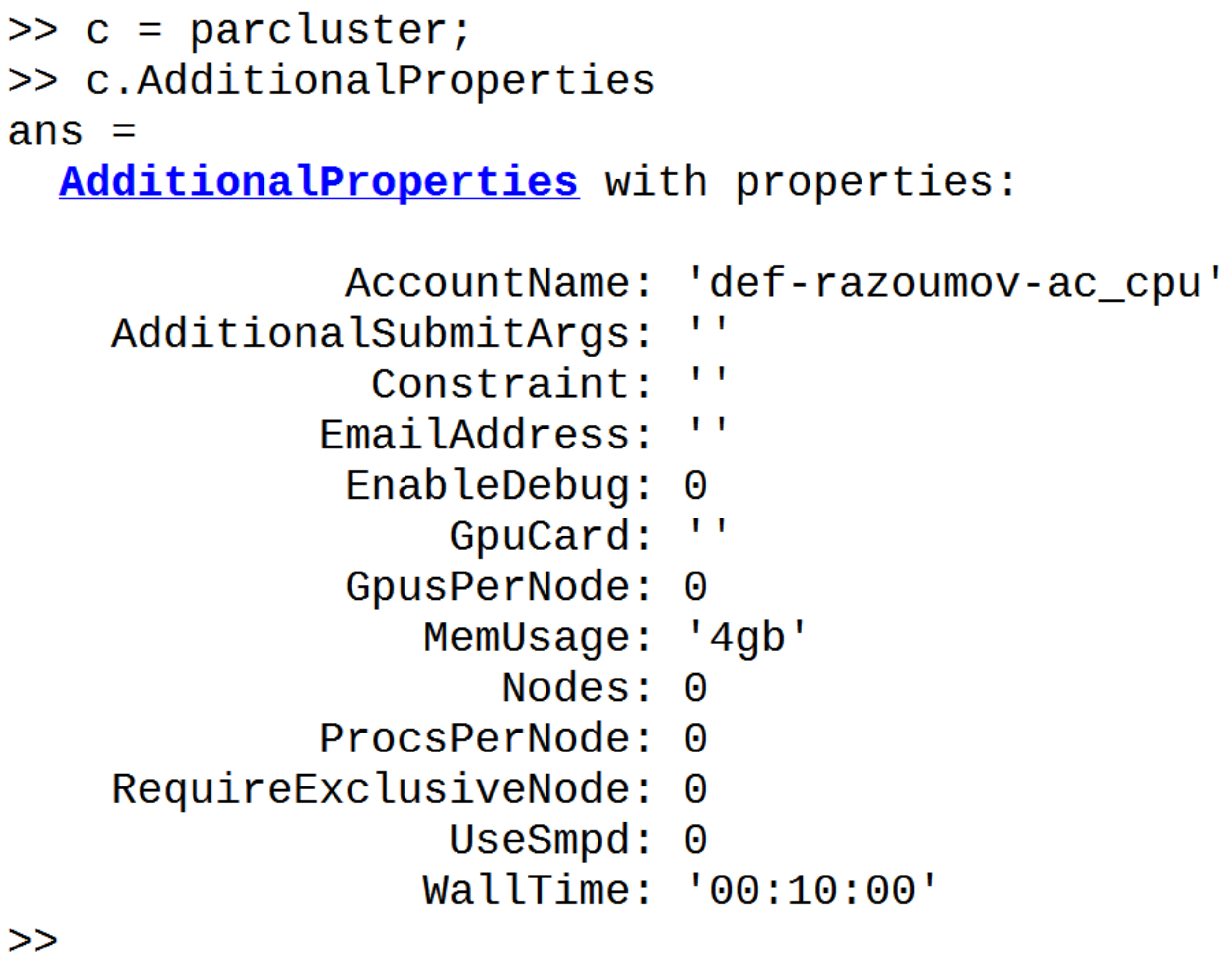


```
>> calc_pi
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 8).
Lab 1:
  Subinterval: [0    , 0.125]
Lab 2:
  Subinterval: [0.125, 0.25]
Lab 3:
  Subinterval: [0.25, 0.375]
Lab 4:
  Subinterval: [0.375, 0.5 ]
Lab 5:
  Subinterval: [0.5 , 0.625]
Lab 6:
  Subinterval: [0.625, 0.75]
Lab 7:
```



```
>> calc_pi_multi_node
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...

additionalSubmitArgs =

    '--ntasks=80 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov-

Connected to the parallel pool (number of workers: 80).
Lab  2:
  Subinterval: [0.0125, 0.025]
Lab  3:
  Subinterval: [0.025, 0.0375]
Lab  4:
  Subinterval: [0.0375, 0.05]
Lab  5:
  Subinterval: [0.05, 0.0625]
Lab  6:
  Subinterval: [0.0625, 0.075]
```

Other settable job properties (1)



```
>> c = parcluster;
>> c.AdditionalProperties
ans =
  AdditionalProperties with properties:

             AccountName: 'def-razoumov-ac_cpu'
    AdditionalSubmitArgs: ''
              Constraint: ''
            EmailAddress: ''
             EnableDebug: 0
                 GpuCard: ''
             GpusPerNode: 0
                MemUsage: '4gb'
                   Nodes: 0
             ProcsPerNode: 0
    RequireExclusiveNode: 0
                 UseSmpd: 0
                WallTime: '00:10:00'
>>
```

Other settable job properties (2)

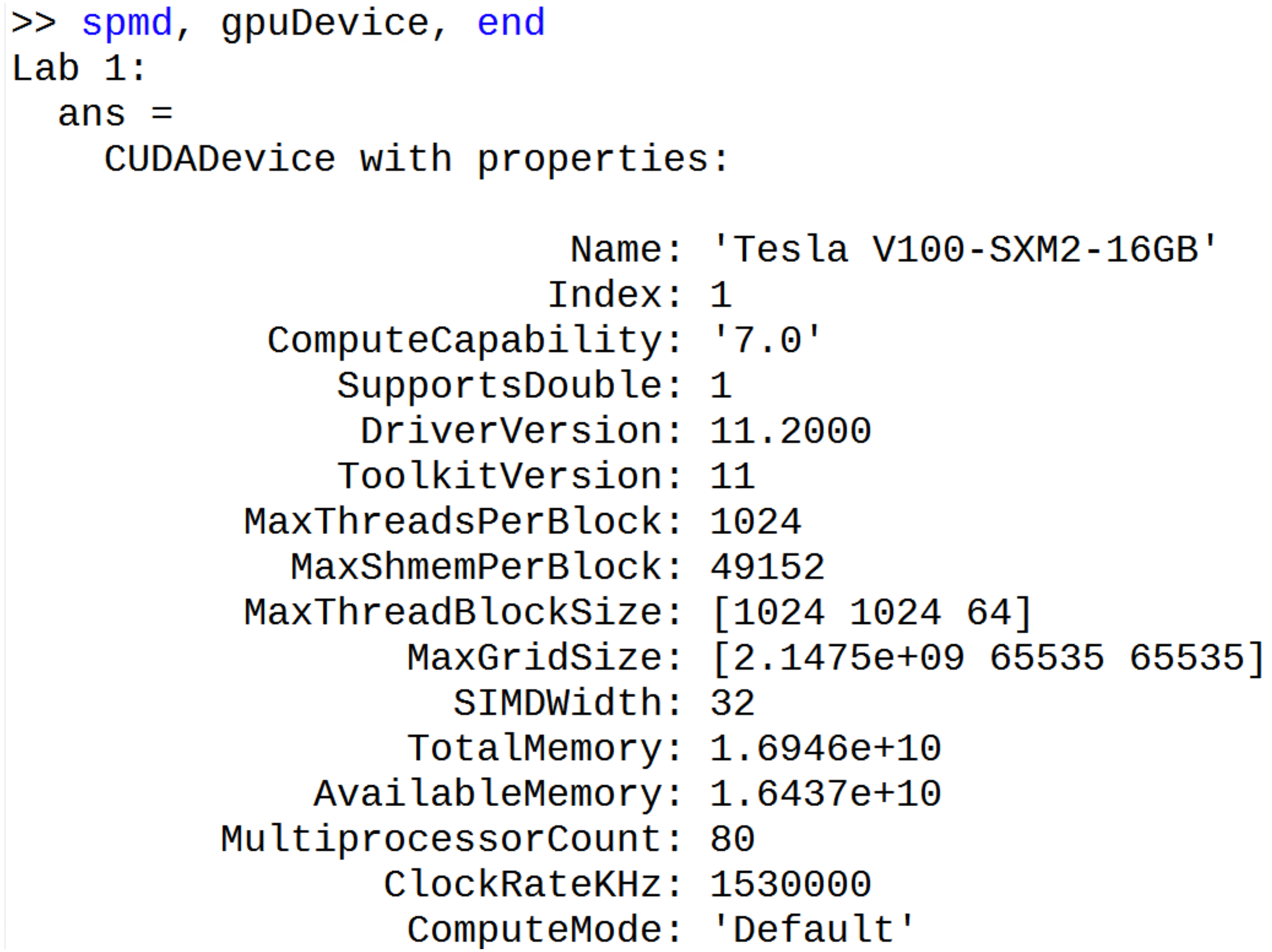
- AccountName
- Constraint
- EmailAddress
- GpuCard
- GpusPerNode
- MemUsage
- Nodes
- ProcsPerNode
- RequireExclusiveNode
- WallTime

GPUs

Start pool with GPU node

```
>> % Start a parallel pool with a GPU
>> c = parcluster;
>> c.AdditionalProperties.GpusPerNode = 1;
>>
>> pool = c.parpool(1);
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov-
Connected to the parallel pool (number of workers: 1).
>>
```

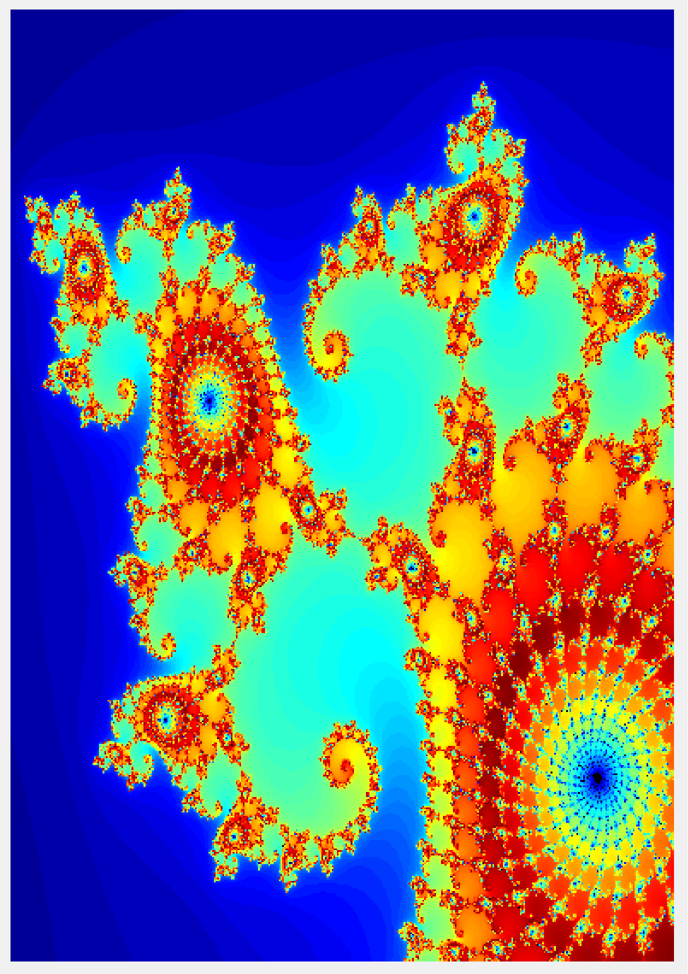
Tesla V100



```
>> spmd, gpuDevice, end
Lab 1:
  ans =
    CUDADevice with properties:

                      Name: 'Tesla V100-SXM2-16GB'
                     Index: 1
         ComputeCapability: '7.0'
            SupportsDouble: 1
             DriverVersion: 11.2000
            ToolkitVersion: 11
        MaxThreadsPerBlock: 1024
          MaxShmemPerBlock: 49152
        MaxThreadBlockSize: [1024 1024 64]
               MaxGridSize: [2.1475e+09 65535 65535]
                 SIMDWidth: 32
               TotalMemory: 1.6946e+10
           AvailableMemory: 1.6437e+10
       MultiprocessorCount: 80
              ClockRateKHz: 1530000
               ComputeMode: 'Default'
```

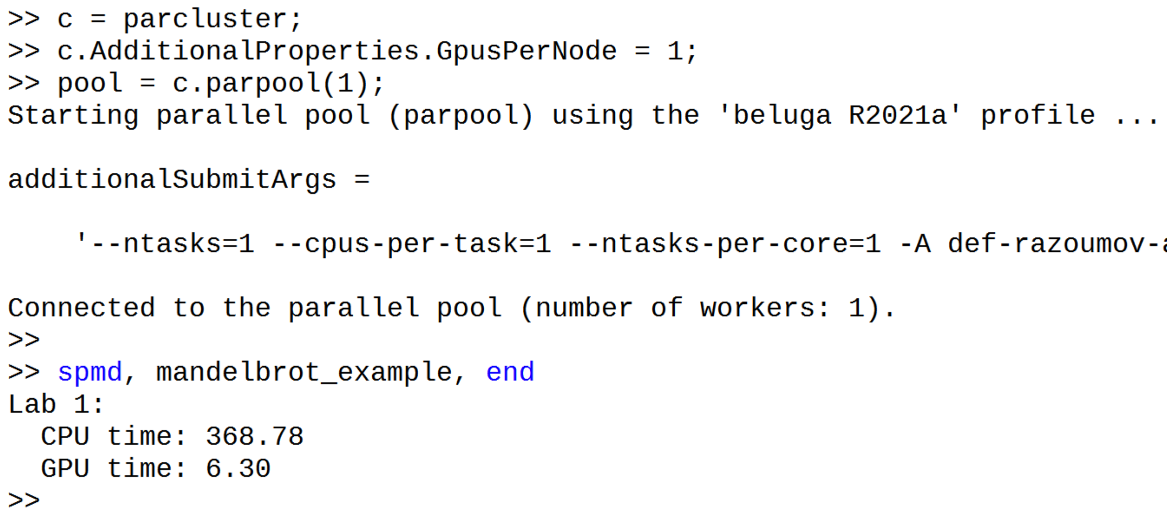
calc_mandelbrot

```
function [x,y,count,t] = calc_mandelbrot(type)

maxIterations = 1000;
gridSize = 4000;
xlim = [-0.748766713922161, -0.748766707771757];
ylim = [ 0.123640844894862,  0.123640851045266];

t0 = tic;
if strcmp(type,'gpuArray')
    x = gpuArray.linspace(xlim(1),xlim(2),gridSize);
    y = gpuArray.linspace(ylim(1),ylim(2),gridSize);
else
    x = linspace(xlim(1),xlim(2),gridSize);
    y = linspace(ylim(1),ylim(2),gridSize);
end

[xGrid,yGrid] = meshgrid(x,y);
z0 = complex(xGrid,yGrid);
count = ones(size(z0),type);

z = z0;
for n = 0:maxIterations
    z = z.*z + z0;
    inside = abs(z) <= 2;
    count = count + inside;
end
count = log(count);
t = toc(t0);

end
```



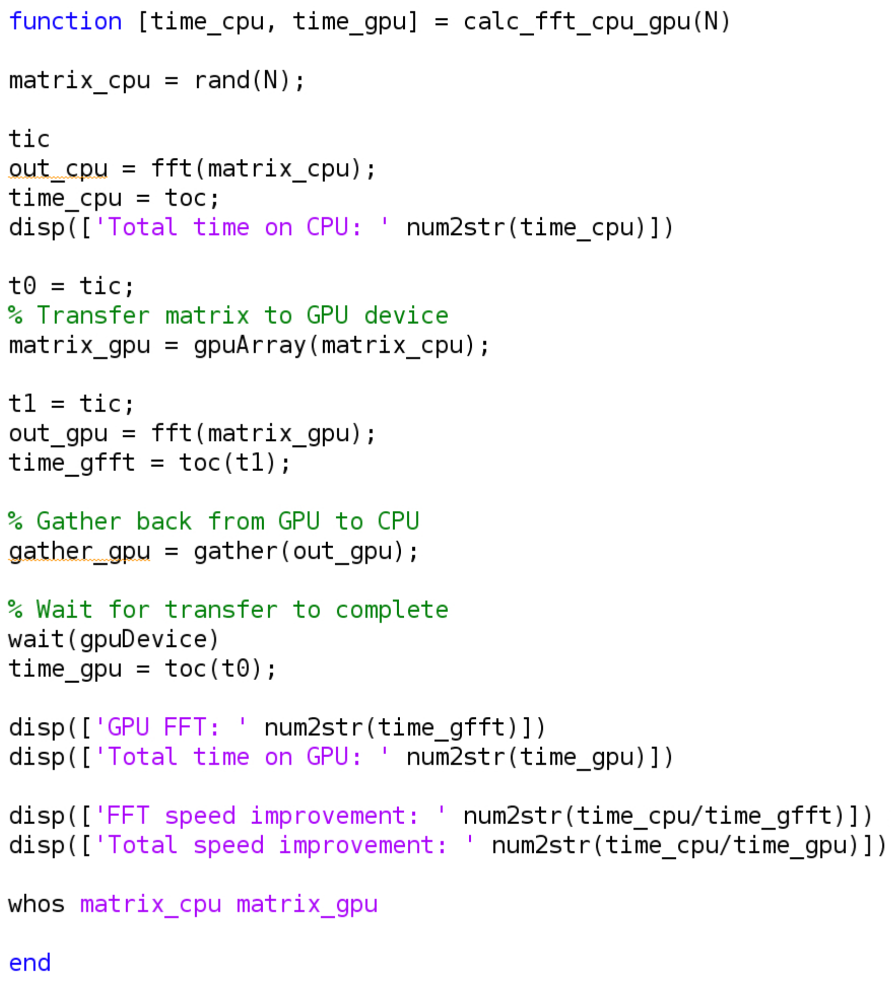
mandelbrot_example

```
function mandelbrot_example

% Run on CPU
[~, ~, ~, cpu_t] = calc_mandelbrot('double');

% Run on GPU
[~, ~, ~, gpu_t] = calc_mandelbrot('gpuArray');

fprintf('CPU time: %0.2f\n',cpu_t)
fprintf('GPU time: %0.2f\n',gpu_t)

end
```



```
>> c = parcluster;
>> c.AdditionalProperties.GpusPerNode = 1;
>> pool = c.parpool(1);
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov-a

Connected to the parallel pool (number of workers: 1).
>>
>> spmd, mandelbrot_example, end
Lab 1:
  CPU time: 368.78
  GPU time: 6.30
>>
```

GPU example – FFT (1)



```
function [time_cpu, time_gpu] = calc_fft_cpu_gpu(N)

matrix_cpu = rand(N);

tic
out_cpu = fft(matrix_cpu);
time_cpu = toc;
disp(['Total time on CPU: ' num2str(time_cpu)])

t0 = tic;
% Transfer matrix to GPU device
matrix_gpu = gpuArray(matrix_cpu);

t1 = tic;
out_gpu = fft(matrix_gpu);
time_gfft = toc(t1);

% Gather back from GPU to CPU
gather_gpu = gather(out_gpu);

% Wait for transfer to complete
wait(gpuDevice)
time_gpu = toc(t0);

disp(['GPU FFT: ' num2str(time_gfft)])
disp(['Total time on GPU: ' num2str(time_gpu)])

disp(['FFT speed improvement: ' num2str(time_cpu/time_gfft)])
disp(['Total speed improvement: ' num2str(time_cpu/time_gpu)])

whos matrix_cpu matrix_gpu

end
```

GPU example – FFT (2)

Why did the GPU code run faster the 2nd time?

```
>> % Matrix: 128 mb
>> sz = 2^12;
>>
>> % Use smpd block
>> spmd, [cpu_t, gpu_t] = calc_fft_cpu_gpu(sz); end
Lab 1:
  Total time on CPU: 0.32701
  GPU FFT: 0.53272
  Total time on GPU: 5.1045
  FFT speed improvement: 0.61386
  Total speed improvement: 0.064064
    Name                  Size                     Bytes  Class

    matrix_cpu         4096x4096               134217728  double
    matrix_gpu         4096x4096                       4  gpuArray

>>
```

```
>> % Why will the GPU run faster the second time?
>> spmd, [cpu_t, gpu_t] = calc_fft_cpu_gpu(sz); end
Lab 1:
  Total time on CPU: 0.20585
  GPU FFT: 0.001909
  Total time on GPU: 0.22969
  FFT speed improvement: 107.8287
  Total speed improvement: 0.89619
    Name                  Size                     Bytes  Class

    matrix_cpu         4096x4096               134217728  double
    matrix_gpu         4096x4096                       4  gpuArray

>>
```

Submit single-node job & multi-node job

If we're running a multi-node job, why did we only request 1 core?

Why are we asking for more walltime for a multi-node job?

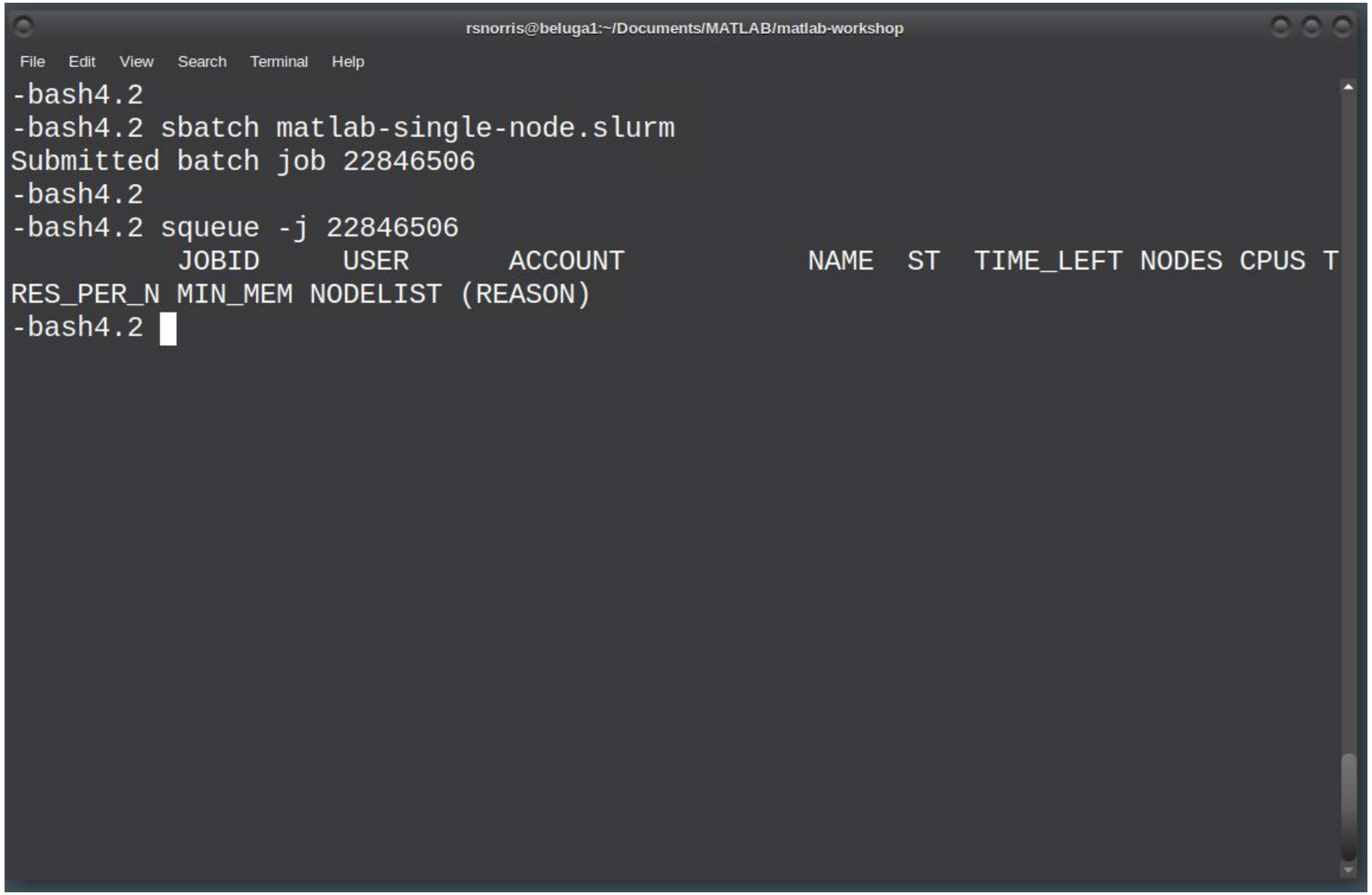
```
#!/bin/sh

#SBATCH -n 1                          # 1 instance of MATLAB
#SBATCH --cpus-per-task=20            # 20 cores per instance
#SBATCH --mem-per-cpu=4gb             # 4 GB RAM per core
#SBATCH --time=00:10:00               # 10 minutes

# Add MATLAB to system path
module load matlab

# Run code
matlab -batch calc_pi
```

matlab-single-node.slurm

```
#!/bin/sh

#SBATCH -n 1                          # 1 instance of MATLAB
#SBATCH --cpus-per-task=1             # 1 core per instance
#SBATCH --mem-per-cpu=4gb             # 4 GB RAM per core
#SBATCH --time=00:20:00               # 20 minutes

# Add MATLAB to system path
module load matlab

# Run code
matlab -batch calc_pi_multi_node
```

matlab-multi-node.slurm

Single-node job (1)



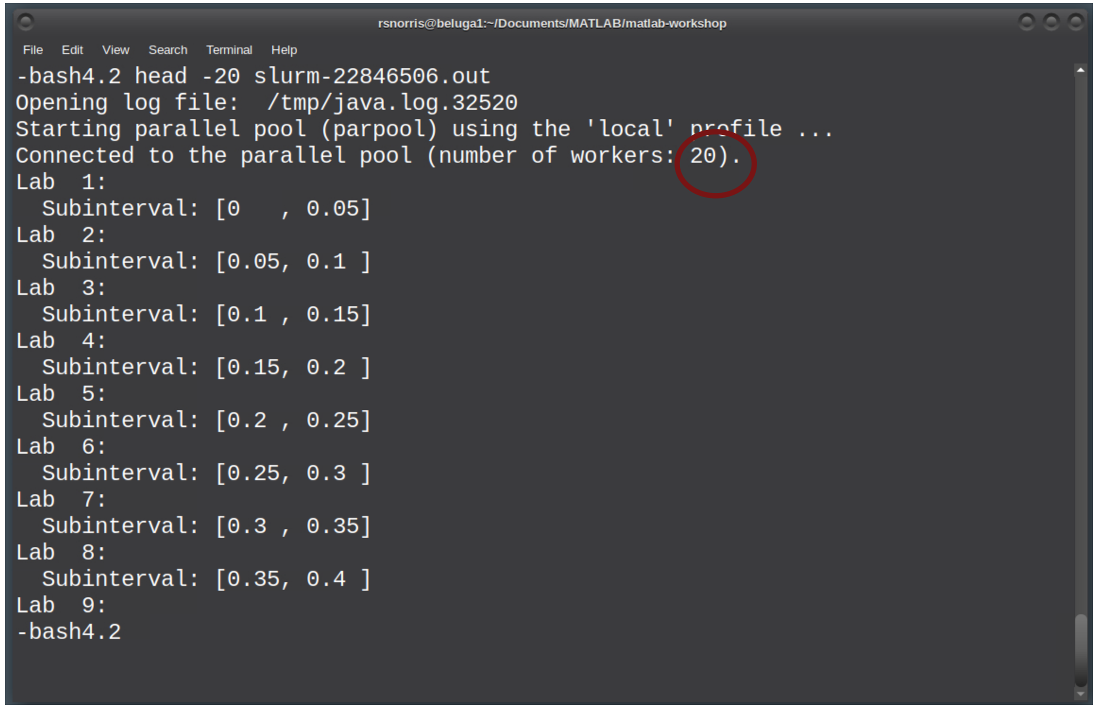
rsnorris@beluga1:~/Documents/MATLAB/matlab-workshop
File Edit View Search Terminal Help

```
-bash4.2
-bash4.2 sbatch matlab-single-node.slurm
Submitted batch job 22846506
-bash4.2
-bash4.2 squeue -j 22846506
          JOBID     USER      ACCOUNT           NAME  ST  TIME_LEFT NODES CPUS T
RES_PER_N MIN_MEM NODELIST (REASON)
-bash4.2
```

Single-node job (2)

How many MATLAB processes are running? 19? 20? 21?

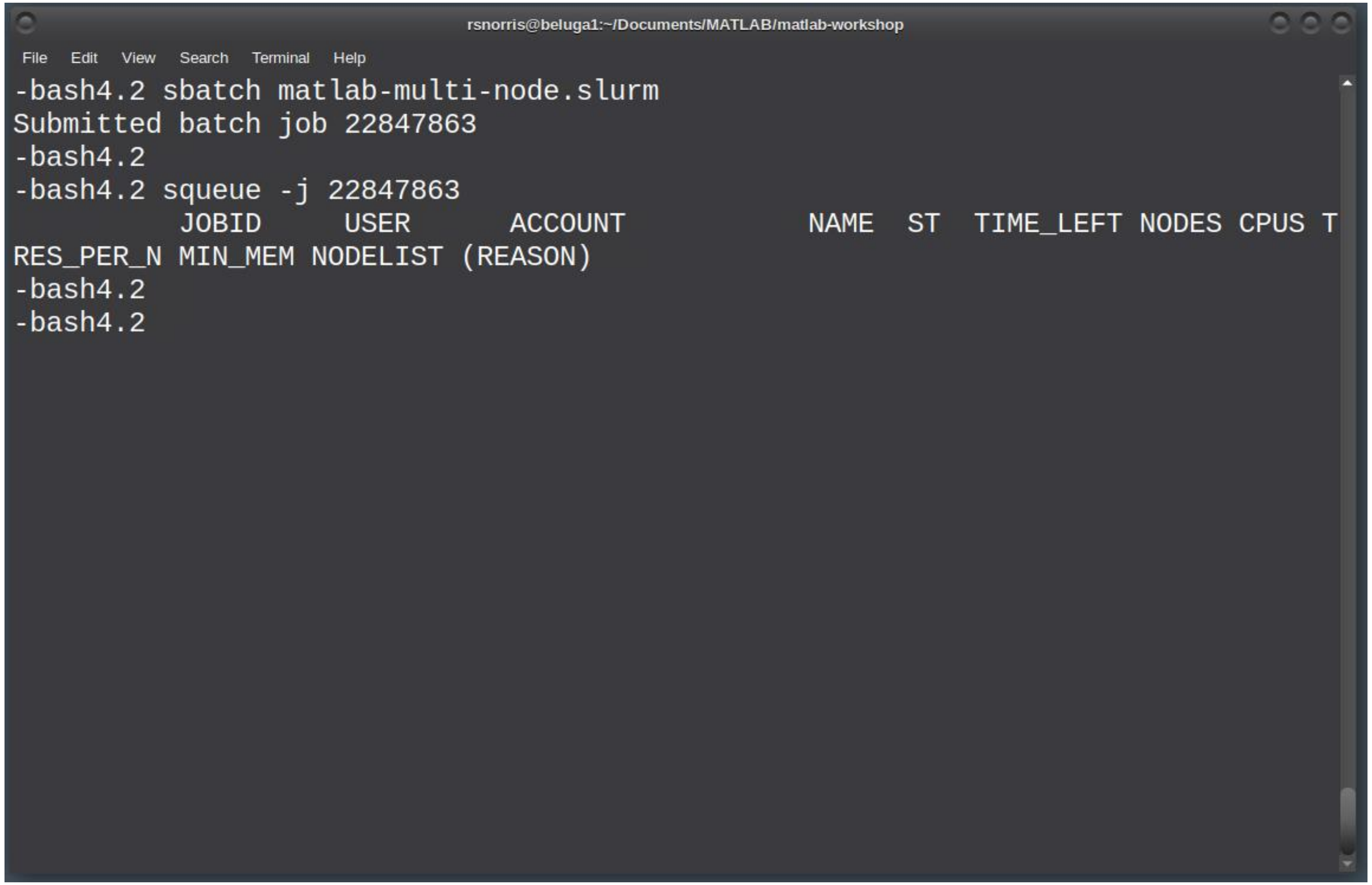
```
function calc_pi

c = parcluster('local');

% Query for available cores (assume either Slurm or PBS)
sz = str2num([getenv('SLURM_CPUS_PER_TASK') getenv('PBS_NP')]); %#ok<ST2NM>
if isempty(sz), sz = maxNumCompThreads; end

c.parpool(sz);
```


rsnorris@beluga1:~/Documents/MATLAB/matlab-workshop
File Edit View Search Terminal Help

```
-bash4.2 head -20 slurm-22846506.out
Opening log file:  /tmp/java.log.32520
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 20).
Lab  1:
  Subinterval: [0   , 0.05]
Lab  2:
  Subinterval: [0.05, 0.1 ]
Lab  3:
  Subinterval: [0.1 , 0.15]
Lab  4:
  Subinterval: [0.15, 0.2 ]
Lab  5:
  Subinterval: [0.2 , 0.25]
Lab  6:
  Subinterval: [0.25, 0.3 ]
Lab  7:
  Subinterval: [0.3 , 0.35]
Lab  8:
  Subinterval: [0.35, 0.4 ]
Lab  9:
-bash4.2
```

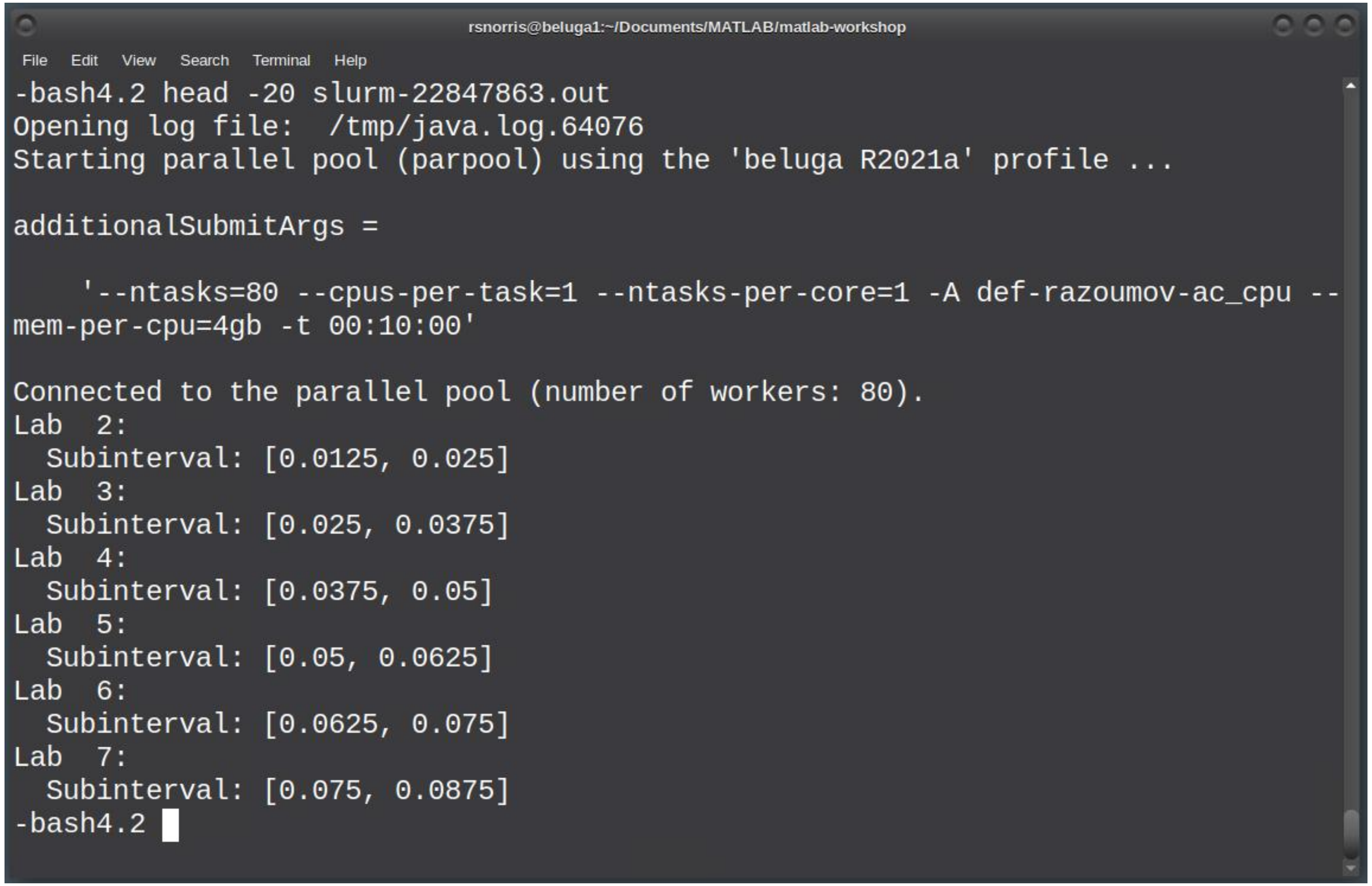
```
#!/bin/sh

#SBATCH -n 1                          # 1 instance of MATLAB
#SBATCH --cpus-per-task=20            # 20 cores per instance
#SBATCH --mem-per-cpu=4gb             # 4 GB RAM per core
#SBATCH --time=00:10:00               # 10 minutes

# Add MATLAB to system path
module load matlab

# Run code
matlab -batch calc_pi
```

Multi-node job (1)


rsnorris@beluga1:~/Documents/MATLAB/matlab-workshop
File Edit View Search Terminal Help

```
-bash4.2 sbatch matlab-multi-node.slurm
Submitted batch job 22847863
-bash4.2
-bash4.2 squeue -j 22847863
          JOBID     USER       ACCOUNT                NAME  ST   TIME_LEFT NODES CPUS T
RES_PER_N MIN_MEM NODELIST (REASON)
-bash4.2
-bash4.2
```

Multi-node pool of workers (2)



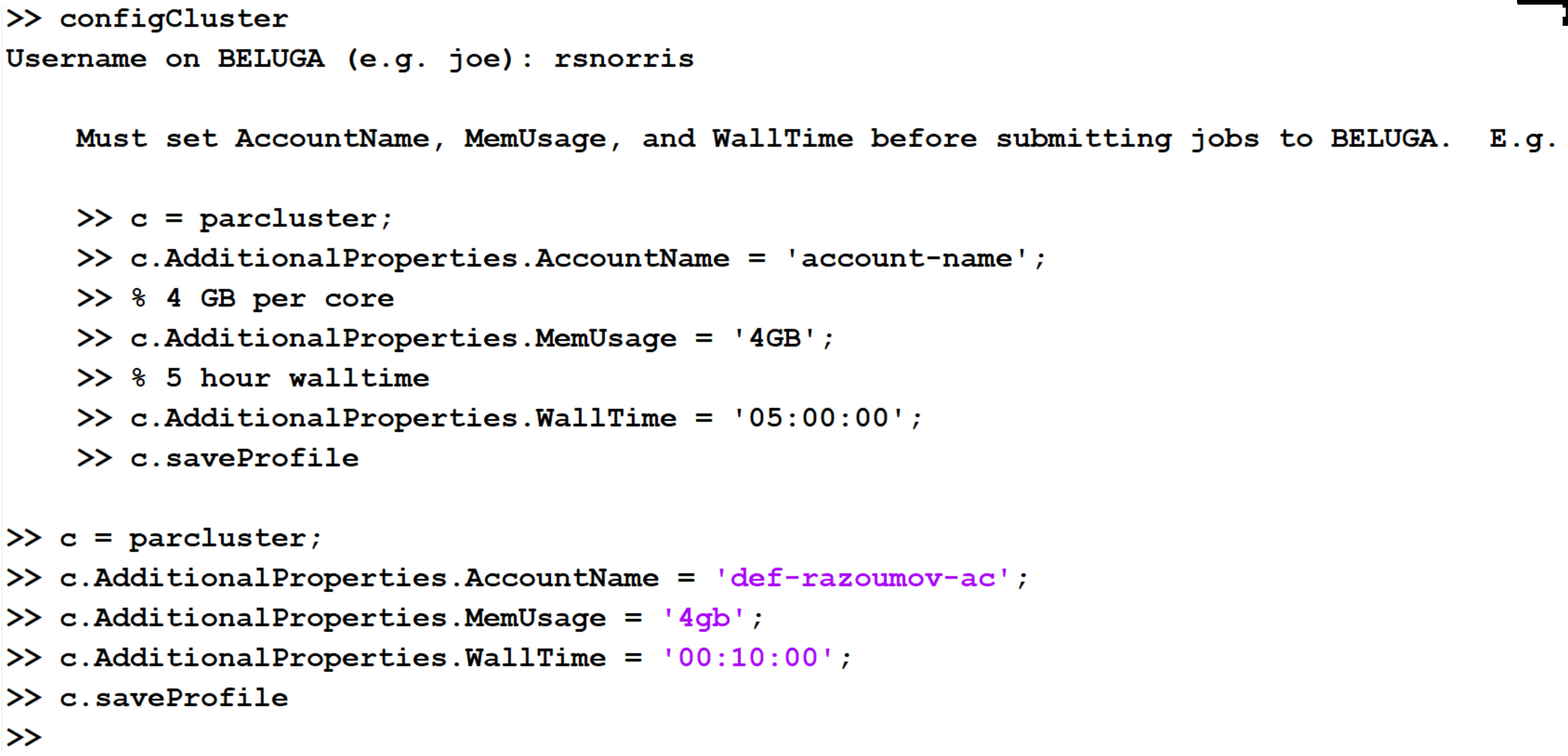
rsnorris@beluga1:~/Documents/MATLAB/matlab-workshop
File Edit View Search Terminal Help

```
-bash4.2 head -20 slurm-22847863.out
Opening log file:  /tmp/java.log.64076
Starting parallel pool (parpool) using the 'beluga R2021a' profile ...

additionalSubmitArgs =

    '--ntasks=80 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov-ac_cpu --
mem-per-cpu=4gb -t 00:10:00'

Connected to the parallel pool (number of workers: 80).
Lab  2:
  Subinterval: [0.0125, 0.025]
Lab  3:
  Subinterval: [0.025, 0.0375]
Lab  4:
  Subinterval: [0.0375, 0.05]
Lab  5:
  Subinterval: [0.05, 0.0625]
Lab  6:
  Subinterval: [0.0625, 0.075]
Lab  7:
  Subinterval: [0.075, 0.0875]
-bash4.2
```

Wait. What about Cedar and Graham? . . .

- Cedar and Graham* operate the same way (i.e., calling `configCluster`)

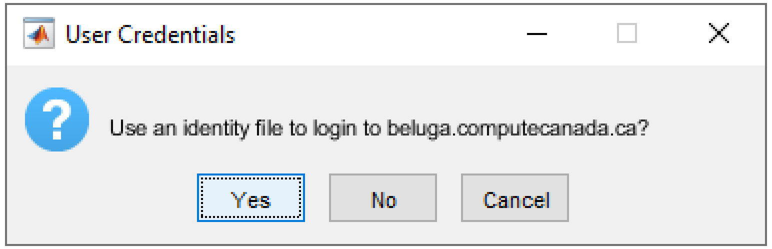
Wait, Wait. What about running MATLAB on my local desktop?



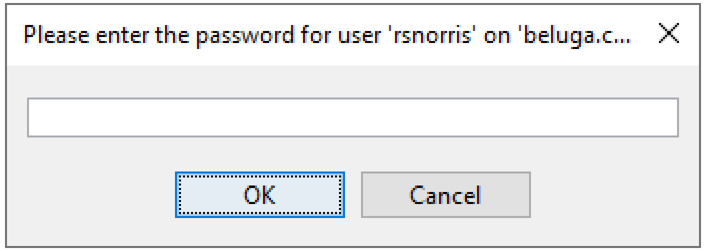
```
>> configCluster
Username on BELUGA (e.g. joe): rsnorris

    Must set AccountName, MemUsage, and WallTime before submitting jobs to BELUGA.  E.g.

    >> c = parcluster;
    >> c.AdditionalProperties.AccountName = 'account-name';
    >> % 4 GB per core
    >> c.AdditionalProperties.MemUsage = '4GB';
    >> % 5 hour walltime
    >> c.AdditionalProperties.WallTime = '05:00:00';
    >> c.saveProfile

>> c = parcluster;
>> c.AdditionalProperties.AccountName = 'def-razoumov-ac';
>> c.AdditionalProperties.MemUsage = '4gb';
>> c.AdditionalProperties.WallTime = '00:10:00';
>> c.saveProfile
>>
```

Remote Submission

```
>> % Submit calc_pi job
>> c = parcluster;
>> job = c.batch(@calc_pi,0,{},'CurrentFolder','.','AutoAddClientPath',false,'Pool',79);
additionalSubmitArgs =
    '--ntasks=80 --cpus-per-task=1 --ntasks-per-core=1 -A def-razoumov-ac --mem-per-cpu=4gb -t 00:10:00'
>>
>> job.State
ans =
    'finished'
>>
```

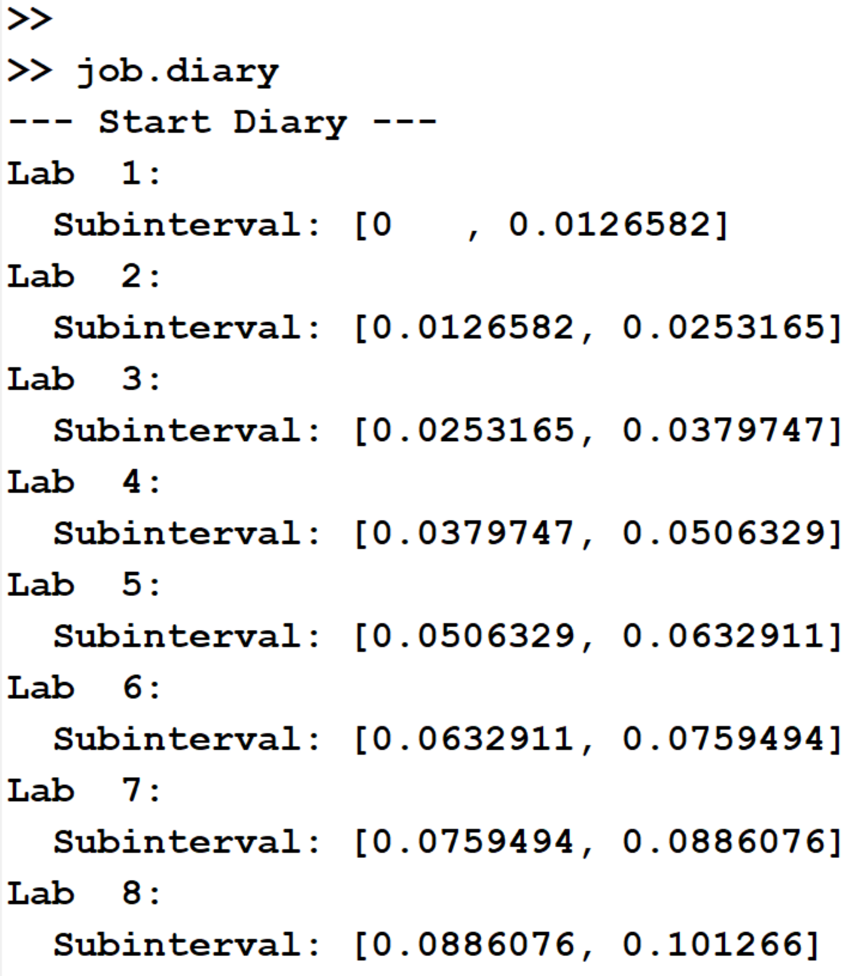


Use an identity file to login to beluga.computecanada.ca?



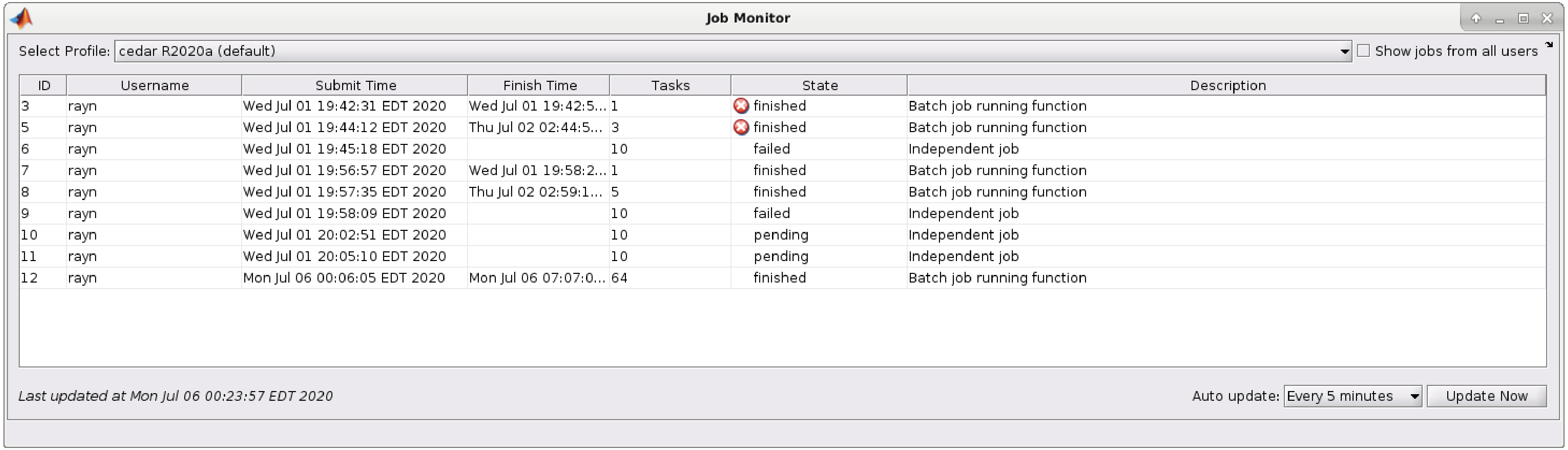
Please enter the password for user 'rsnorris' on 'beluga.c...

Fetch results / diary



```
>>
>> job.diary
--- Start Diary ---
Lab  1:
  Subinterval: [0   , 0.0126582]
Lab  2:
  Subinterval: [0.0126582, 0.0253165]
Lab  3:
  Subinterval: [0.0253165, 0.0379747]
Lab  4:
  Subinterval: [0.0379747, 0.0506329]
Lab  5:
  Subinterval: [0.0506329, 0.0632911]
Lab  6:
  Subinterval: [0.0632911, 0.0759494]
Lab  7:
  Subinterval: [0.0759494, 0.0886076]
Lab  8:
  Subinterval: [0.0886076, 0.101266]
```

Getting Prior Results – Job Monitor



Job Monitor

Select Profile: cedar R2020a (default) □ Show jobs from all users

ID	Username	Submit Time	Finish Time	Tasks	State	Description
3	rayn	Wed Jul 01 19:42:31 EDT 2020	Wed Jul 01 19:42:5...	1	⊗ finished	Batch job running function
5	rayn	Wed Jul 01 19:44:12 EDT 2020	Thu Jul 02 02:44:5...	3	⊗ finished	Batch job running function
6	rayn	Wed Jul 01 19:45:18 EDT 2020		10	failed	Independent job
7	rayn	Wed Jul 01 19:56:57 EDT 2020	Wed Jul 01 19:58:2...	1	finished	Batch job running function
8	rayn	Wed Jul 01 19:57:35 EDT 2020	Thu Jul 02 02:59:1...	5	finished	Batch job running function
9	rayn	Wed Jul 01 19:58:09 EDT 2020		10	failed	Independent job
10	rayn	Wed Jul 01 20:02:51 EDT 2020		10	pending	Independent job
11	rayn	Wed Jul 01 20:05:10 EDT 2020		10	pending	Independent job
12	rayn	Mon Jul 06 00:06:05 EDT 2020	Mon Jul 06 07:07:0...	64	finished	Batch job running function

Last updated at Mon Jul 06 00:23:57 EDT 2020

Auto update: Every 5 minutes