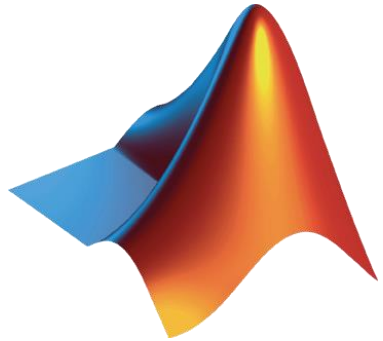
WORKSHOP: Parallel Computing With MATLAB (Part II)

Damian Pietrus
Application Engineer, MathWorks
April 8, 2022



Agenda

- Part I – Parallel Computing with MATLAB on the Desktop
  - Parallel Computing Toolbox
  - MATLAB Online
- Part II – Scaling MATLAB to Teton
  - MATLAB Parallel Server
  - Open OnDemand

Teton Documentation:

https://arccwiki.atlassian.net/wiki/spaces/DOCUMENTAT/pages/1553301507/Teton

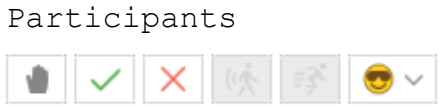
Agenda

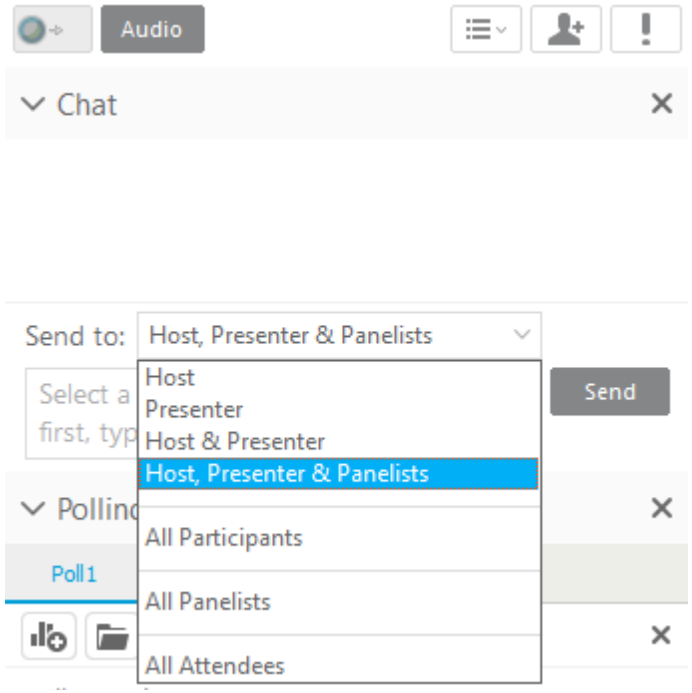
- Part I – Parallel Computing with MATLAB on the Desktop
  - Parallel Computing Toolbox
  - MATLAB Online
- Part II – Scaling MATLAB to Teton
  - MATLAB Parallel Server
  - Open OnDemand

Teton Documentation:
https://arccwiki.atlassian.net/wiki/spaces/DOCUMENTAT/pages/1553301507/Teton

Chatting

- Send to at least the *Host, Presenter & Panelists*
- Ideally, send to *All Attendees*



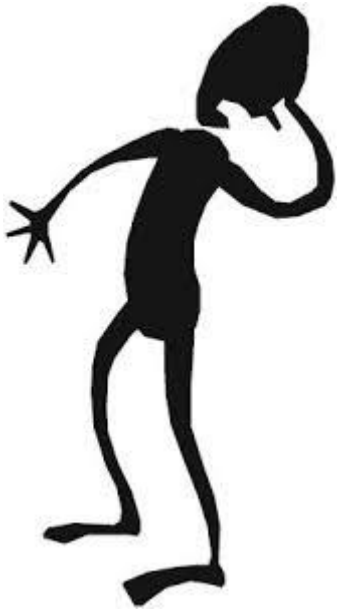


Scaling MATLAB to Teton

- Accessing and running MATLAB on local HPC clusters
- Running single- 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
    - Might be useful for either low-bandwidth or automation
  - Open OnDemand
    - Graphical interface
    - Also supports xterm and copying files

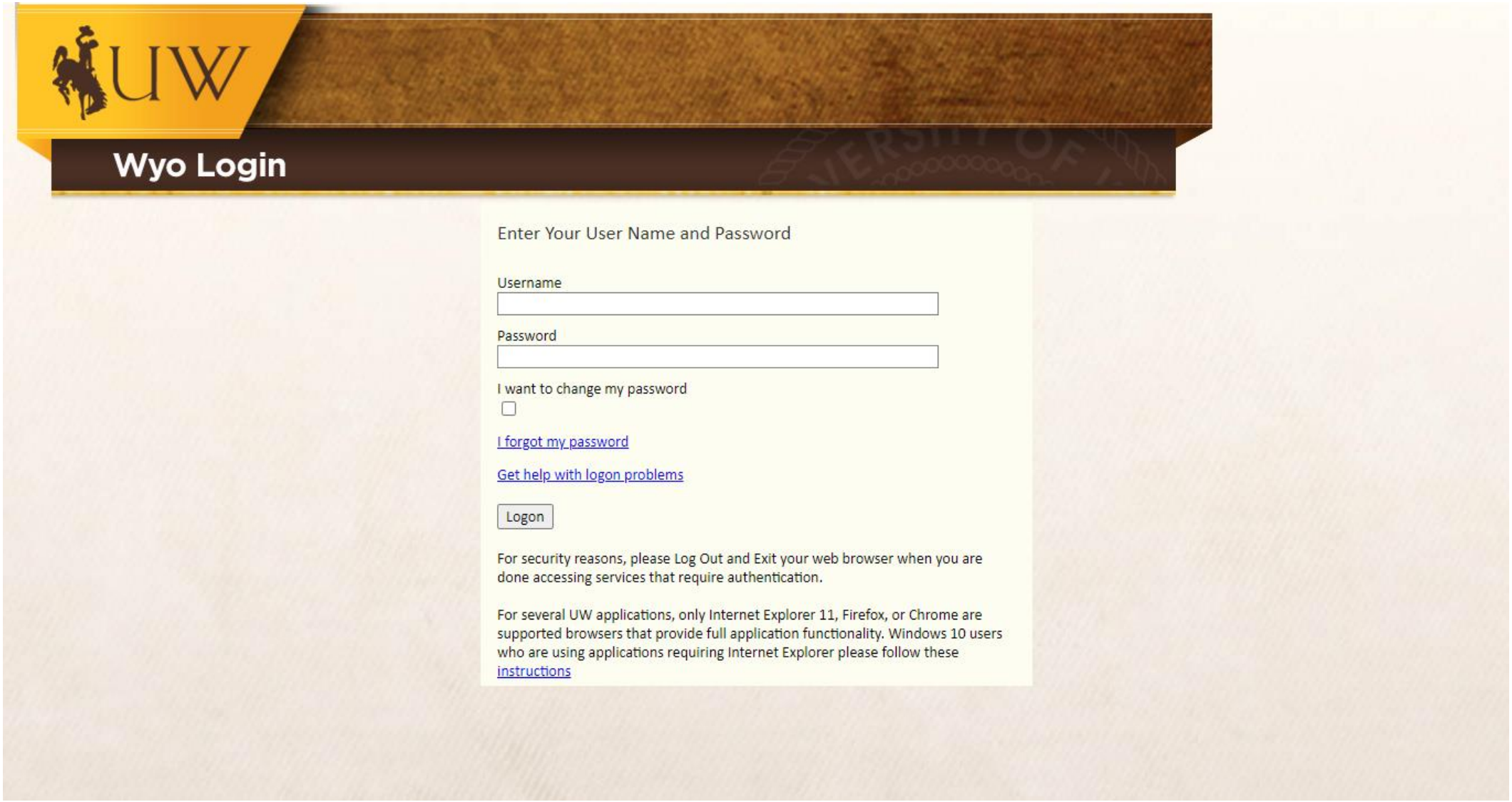
Accessing and running MATLAB on local HPC clusters (2)

- Open OnDemand
  - Web-based, no client software needs to be installed on your local machine (beside a modern browser)

- Access
  - https://southpass.arcc.uwyo.edu/
  - Use login ID and Password

- Local resources for OOD
  - Overview: https://arccwiki.atlassian.net/wiki/spaces/DOCUMENTAT/pages/1298071553/Southpass

Log into OOD: https://southpass.arcc.uwyo.edu/



ARCC | Apps | Files | Jobs | Clusters | Interactive Apps | My Interactive Sessions | Help | Logged in as dpietrus | Log Out

OnDemand provides an integrated, single access point for all of your HPC resources.

Pinned Apps A featured subset of all available apps

- Jupyter — System Installed App
- Active Jobs — System Installed App
- Teton Mate Desktop — System Installed App
- Teton Xfce Desktop — System Installed App
- Home Directory — System Installed App
- Job Composer — System Installed App
- Teton Shell Access — System Installed App
- Teton System Status — System Installed App

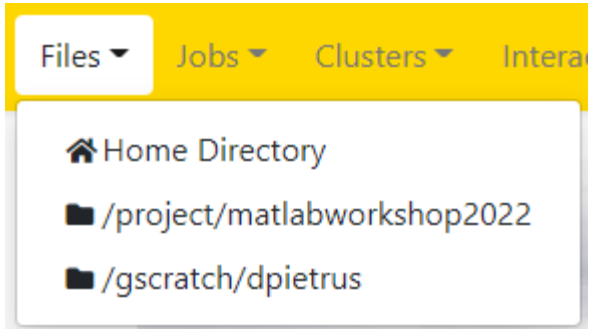
Message of the Day

```
**************************************************
                    _______      ____
                    __  __/____ __  /______
_______
                    __  /  _  _ \_  __/_  __
\__  __ \
                    _  /   /  __// /_  / /_/
/_  / / /
                    /_/    \___/ \__/  \____/
/_/ /_/
**************************************************
Welcome to the University of Wyoming Teton
compute cluster hosted by the
Advanced Research Computing Center (ARCC). This
system is for authorized
users only.

- Accounts are for single users only. Do NOT
share your account with others.
- Users do not have "sudo" privileges and never
will.
- Please avoid building and installing your own
MPI as you will miss key
  features of the job scheduler and interconnect
```

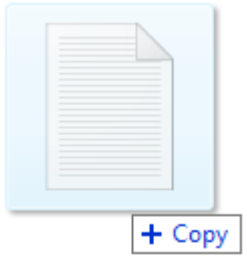
OOD: Getting started

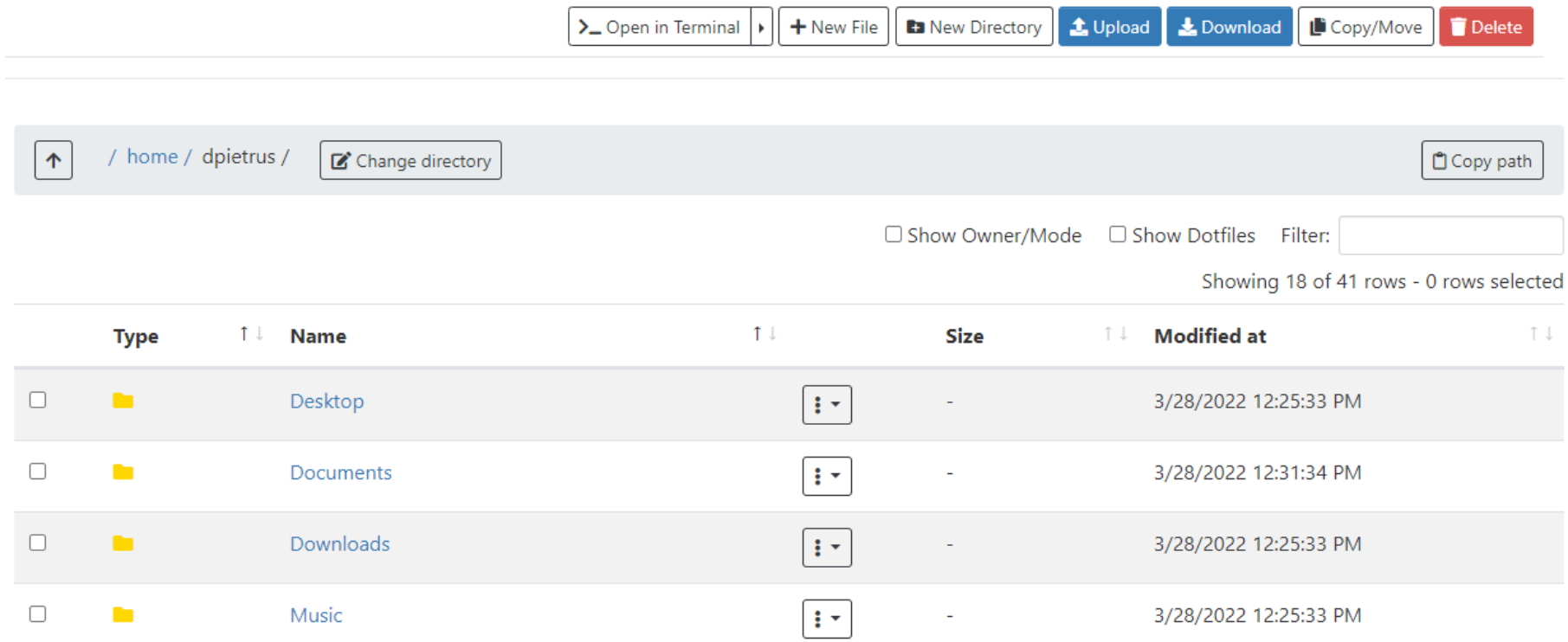
- File system
- Desktop App + MATLAB
- Command prompt

OOD: File system



Click and Drag to upload





Download workshop files (1)


```
********************************************************************************
Check out the ARCC wiki for help and information

  https://arccwiki.atlassian.net/wiki/spaces/DOCUMENTAT/overview

********************************************************************************
Modules:

  module spider - This command will show all packages currently installed

********************************************************************************

Maintenance Scheduled:   Jan 5th, 2022 from 8am to 8pm
+-------------------------------------------------------------------------------+
|      *arccquota tool*      |          Block          |          File          |
+-------------------------------------------------------------------------------+
|            Path            |  Used     Limit     %  |   Used     Limit     %  |
+-------------------------------------------------------------------------------+
| /home/dpietrus             |   0.18g   25.00g   0.72 |     1.0K      0.0   0.00 |
| /gscratch/dpietrus         |   0.00k    5.00t   0.00 |      1.0      0.0   0.00 |
+-------------------------------------------------------------------------------+
| /project/matlabworkshop2022 |   0.00k    1.00t   0.00 |     15.0      0.0   0.00 |
|  `- dpietrus               |   0.00k    0.00k   0.00 |      2.0      0.0   0.00 |
+-------------------------------------------------------------------------------+
Resetting modules to system default
[dpietrus@tlog1 ~]$
```

Download workshop files (2)

```
[dpietrus@tlog1 ~]$
[dpietrus@tlog1 ~]$ # Make a local copy of the Workshop files (Part II)
[dpietrus@tlog1 ~]$ mkdir -p ~/Documents/MATLAB
[dpietrus@tlog1 ~]$ cp -frp /project/matlabworkshop2022/matlab-workshop-files/ ~/Documents/MATLAB/
[dpietrus@tlog1 ~]$
[dpietrus@tlog1 ~]$ ls -l ~/Documents/MATLAB/matlab-workshop-files/
total 6
-rw-r--r-- 1 dpietrus dpietrus 685 Mar 28 14:41 calc_fft_cpu_gpu.m
-rw-r--r-- 1 dpietrus dpietrus 662 Mar 28 14:41 calc_mandelbrot.m
-rw-r--r-- 1 dpietrus dpietrus 849 Mar 28 14:41 calc_pi.m
-rw-r--r-- 1 dpietrus dpietrus 861 Mar 28 14:41 calc_pi_multi_node.m
-rw-r--r-- 1 dpietrus dpietrus 226 Mar 28 14:41 mandelbrot_example.m
-rw-r--r-- 1 dpietrus dpietrus 522 Mar 28 14:41 matlab-gpu.slurm
-rw-r--r-- 1 dpietrus dpietrus 402 Mar 28 14:41 matlab-multi-node.slurm
-rw-r--r-- 1 dpietrus dpietrus 447 Mar 28 14:41 matlab-single-node.slurm
-rw-r--r-- 1 dpietrus dpietrus 418 Mar 28 14:41 matlab-sle.slurm
-rw-r--r-- 1 dpietrus dpietrus 429 Mar 28 14:41 solve_sys_linear_eqns.m
-rw-r--r-- 1 dpietrus dpietrus 194 Mar 28 14:41 test_fcn.m
[dpietrus@tlog1 ~]$
```

Ways to run MATLAB

- Interactively
  - with a parallel pool (`parpool`)
  - with batch jobs (`batch`)
- Noninteractive
  - in a SLURM job script (`sbatch`)

MATLAB job submitters

- `parpool`
  - Single session
  - Synchronous execution
  - Seamlessly runs `parfor`, `parfeval`, and `spmd`

https://www.mathworks.com/help/parallel-computing/parpool.html

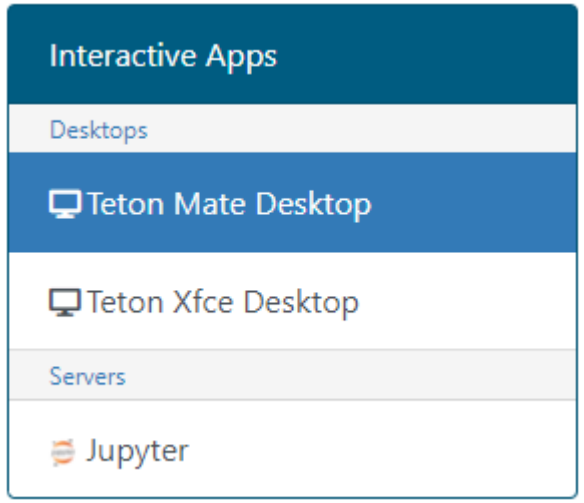
- `batch`
  - Multiple submissions
  - Non-blocking
  - Calls top-level function or script
  - Requires API to extract results

https://www.mathworks.com/help/parallel-computing/batch.html

Interactively: with a parallel pool

`parpool`

OOD: MATLAB via Desktop App (1)



Teton Mate Desktop

This app will launch an interactive desktop on one or more compute nodes. You will have full access to the resources these nodes provide. This is analogous to an interactive batch job.

Project/Account

matlabworkshop2022

Specify the name of your project

Number of hours

4

The maximum number of hours your desktop session will run

Desktop Configuration

2 CPU, 8GB memory

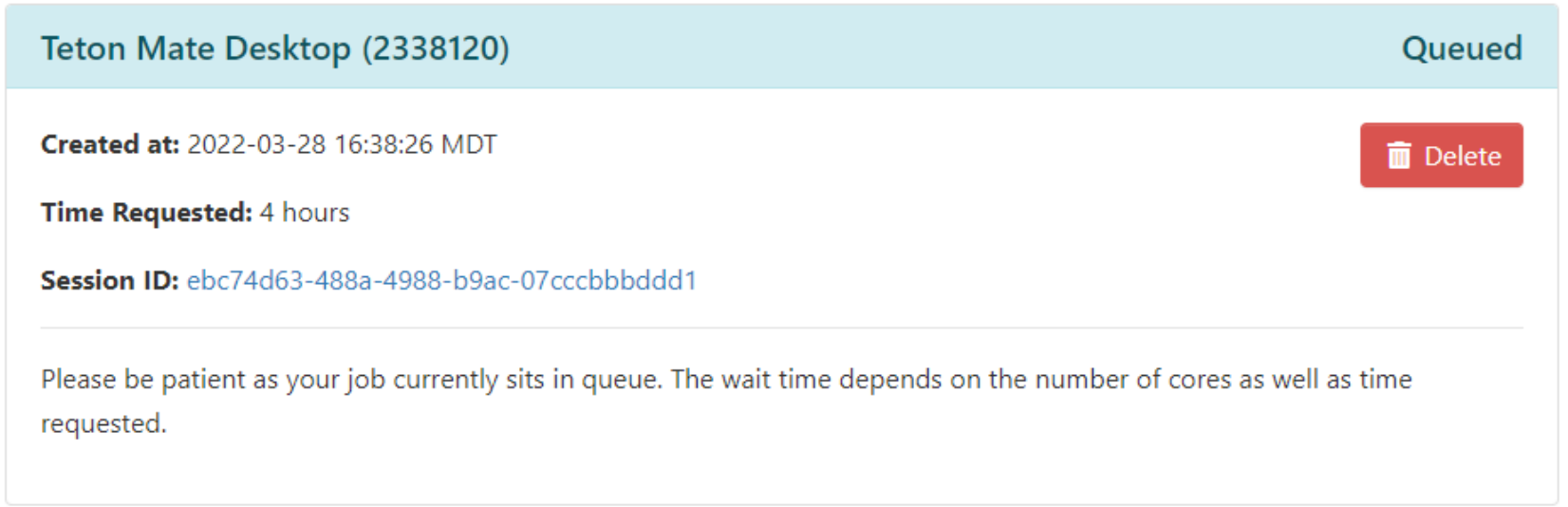
Select the CPU and memory configuration for the desktop

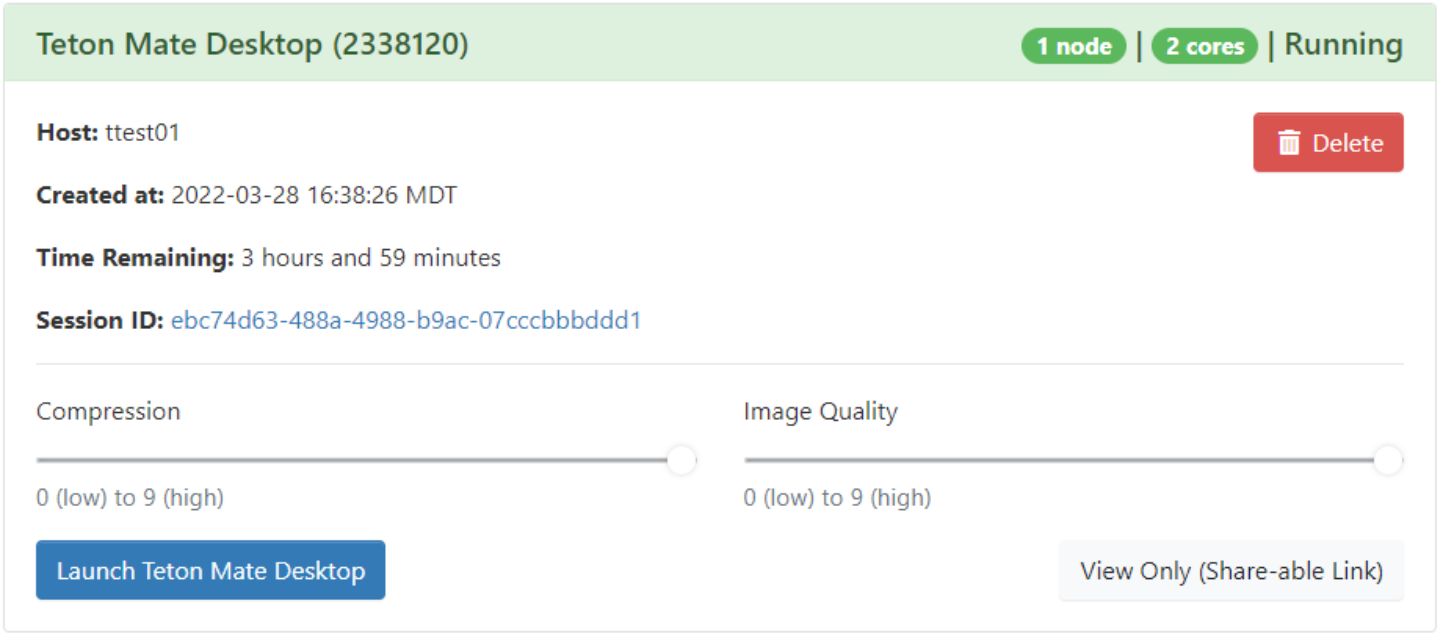
☐ I would like to receive an email when the session starts

Launch

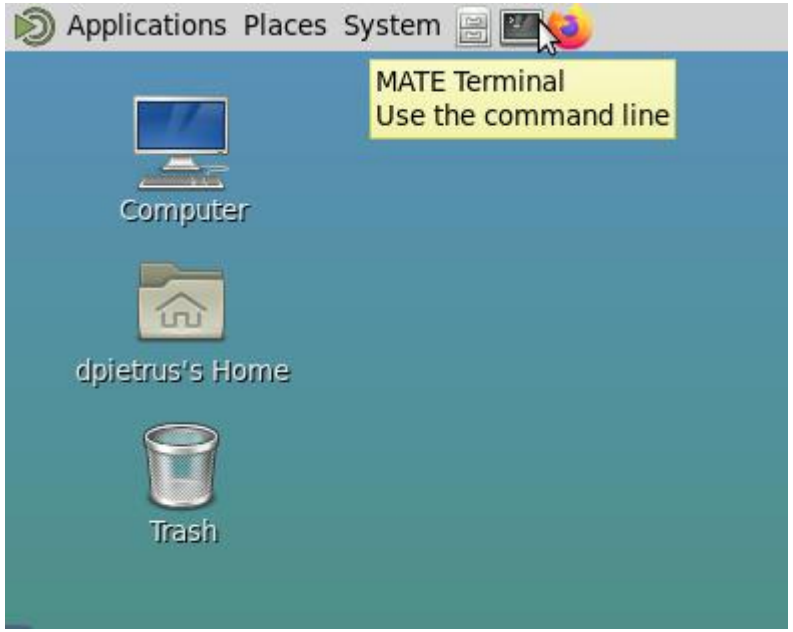
* The Teton Mate Desktop session data for this session can be accessed under the data root directory.

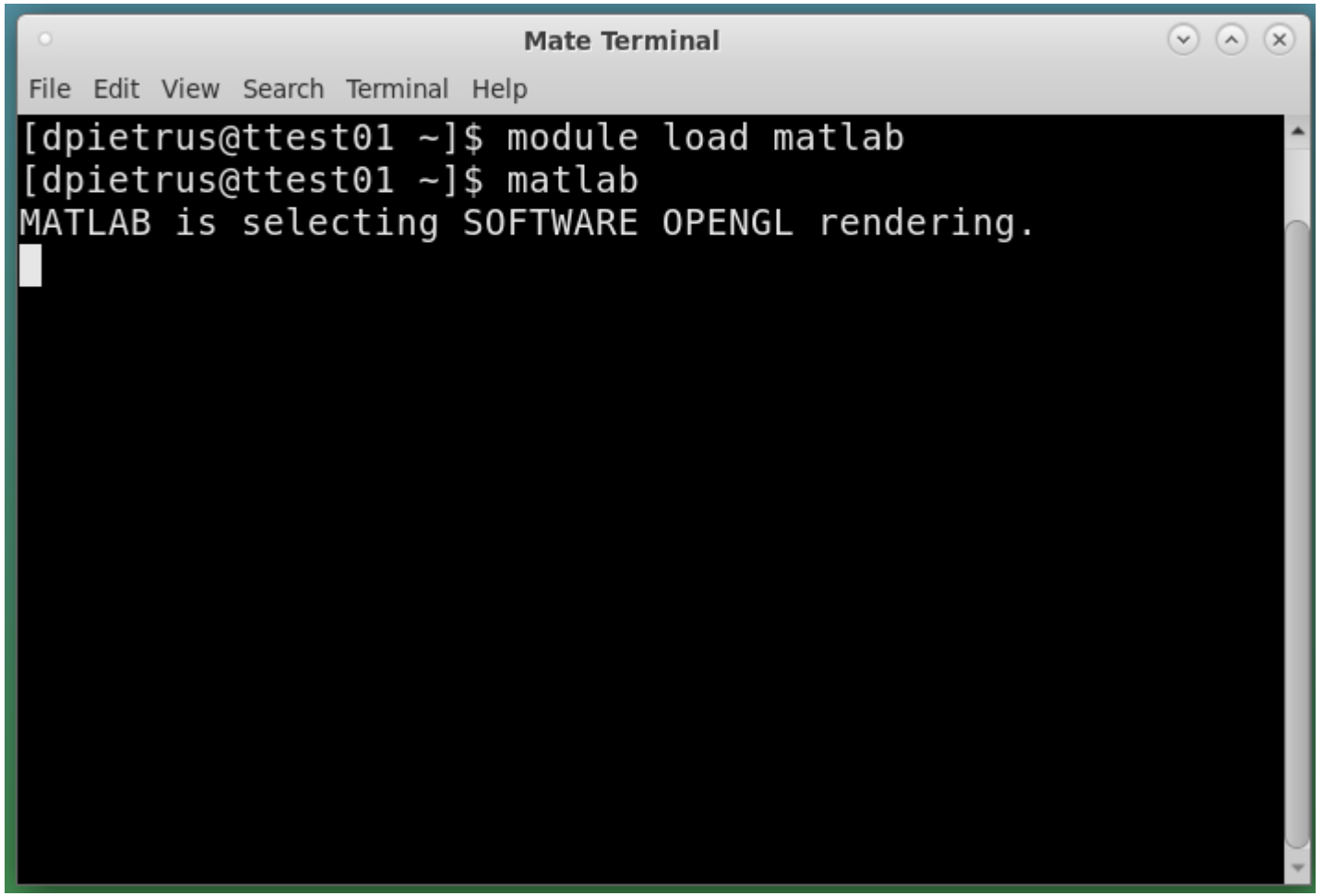
OOD: MATLAB via Desktop App (2)



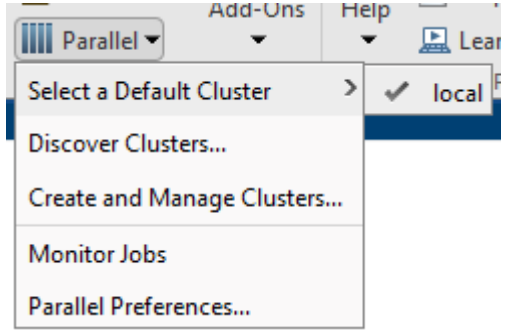


OOD: MATLAB App (3)





Parallel MATLAB – Single Node




```
>> maxNumCompThreads

ans =

     2
```

Teton Mate Desktop

This app will launch an interactive desktop on one or more compute nodes. You will have full access to the resources these nodes provide. This is analogous to an interactive batch job.

Project/Account

matlabworkshop2022

Specify the name of your project

Number of hours

4

The maximum number of hours your desktop session will run

Desktop Configuration

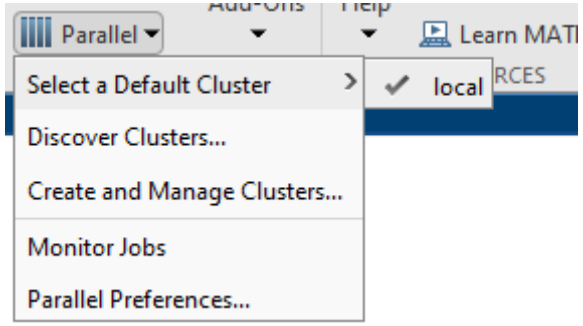
2 CPU, 8GB memory

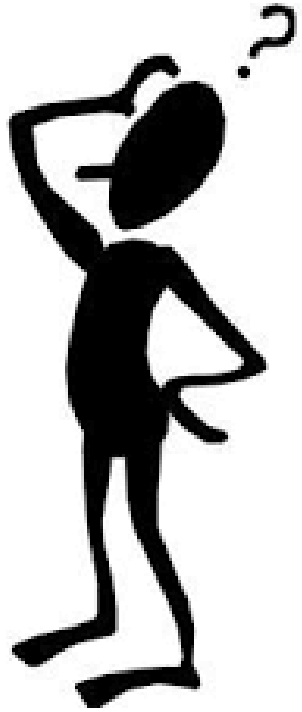
Select the CPU and memory configuration for the desktop

```
>> p = parpool('local',2);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 2).
>>
>> tic, parfor idx = 1:40, pause(3), end, toc
Elapsed time is 60.272708 seconds.
>>
```

local profile

"How does MATLAB know about Teton?"



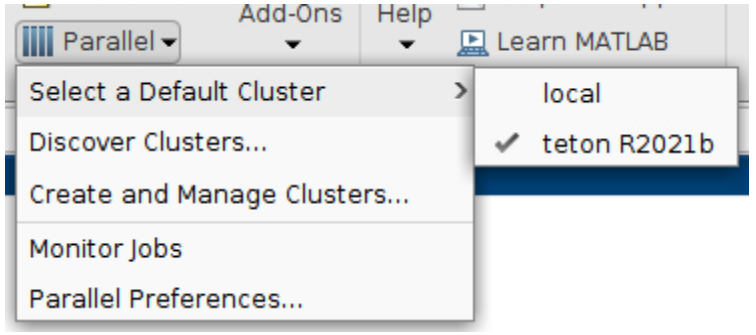


Configure MATLAB to create Teton profile

```
>> % Create a new profile for Teton
>> configCluster

        Must set AccountName and WallTime before submitting jobs to TETON.  E.g.

        >> c = parcluster;
        >> c.AdditionalProperties.AccountName = 'account-name';
        >> c.AdditionalProperties.WallTime = '5:00:00';
        >> c.saveProfile

>>
```

Note the message – we will address this in a few slides

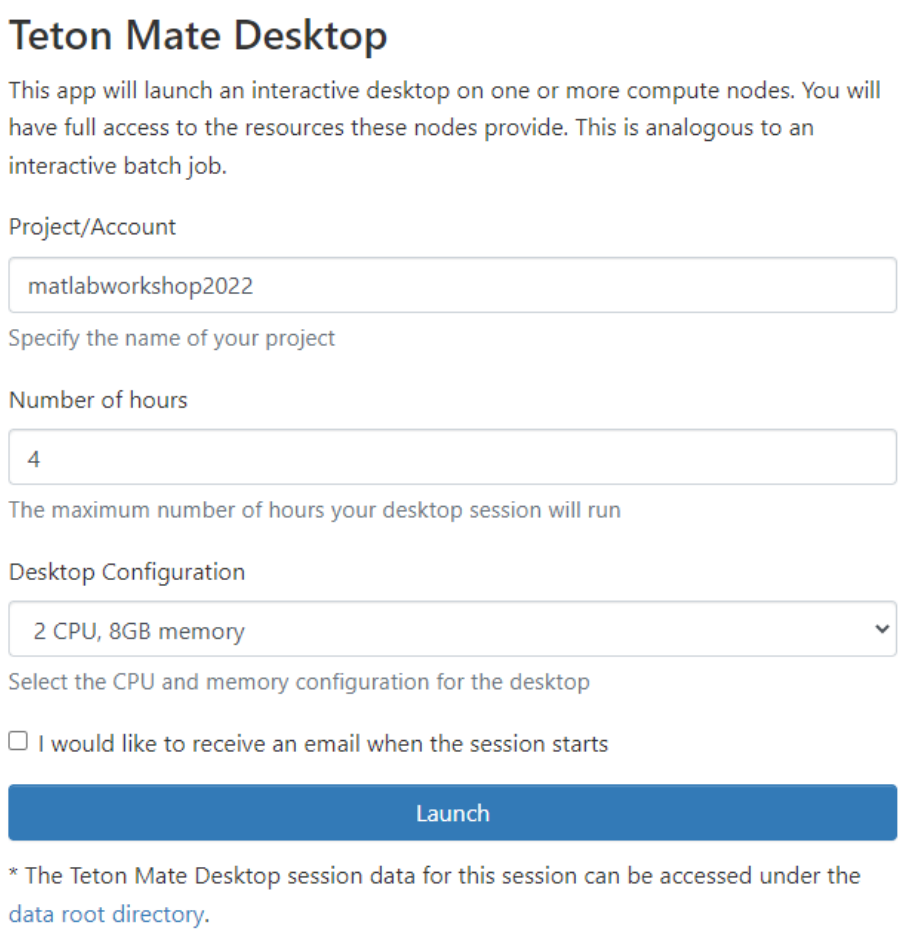
New Teton profile



Only call `configCluster` once

Job submission flags

```
>> c = parcluster;
>> c.AdditionalProperties

ans =

  AdditionalProperties with properties:

              AccountName: ''
    AdditionalSubmitArgs: ''
             EmailAddress: ''
              EnableDebug: 0
                  GpuCard: ''
              GpusPerNode: 0
                 MemUsage: ''
                    Nodes: 0
             ProcsPerNode: 0
                QueueName: ''
              Reservation: ''
                  UseSmpd: 0
                 WallTime: ''

>>
```

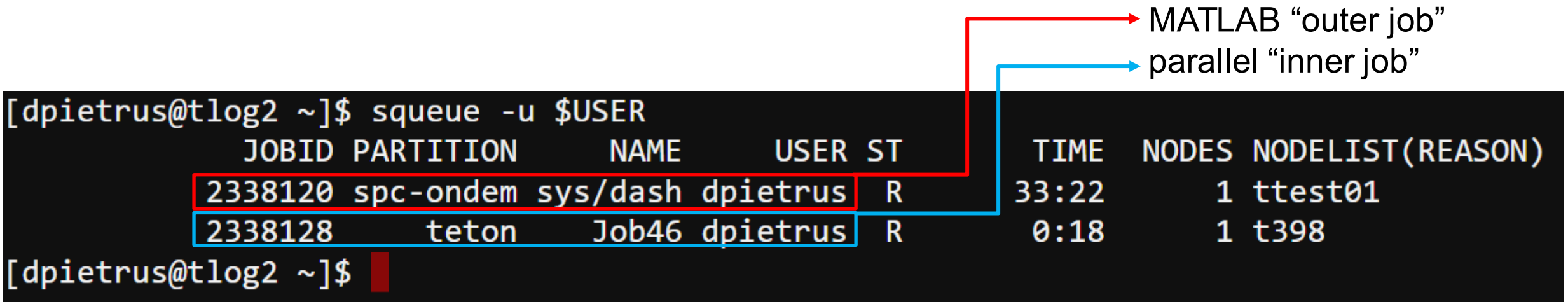


Job submission flags

```
>> % Required fields
>> c.AdditionalProperties.AccountName = 'matlabworkshop2022';
>> c.AdditionalProperties.WallTime = '1:00:00';
>>
>> % Required for this workshop only
>> c.AdditionalProperties.Reservation = 'MATLAB_WS';
>>
>> % Saves changes to profile for future use
>> c.saveProfile
>>
```

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 `pool`
  - True regardless if we're running MATLAB via OOD or a SLURM job script



Parallel MATLAB – Multi-node (2)

```
>> p = parpool('local',2);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 2).
>>
>> tic, parfor idx = 1:40, pause(3), end, toc
Elapsed time is 60.272708 seconds.
>>
```

```
>> % Start multi-node parallel pool
>> p = c.parpool(40);
Starting parallel pool (parpool) using the 'teton R2021b' profile ...

additionalSubmitArgs =

    '--ntasks=40 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00'

Connected to the parallel pool (number of workers: 40).
>> tic, parfor idx = 1:800, pause(3), end, toc
Elapsed time is 60.379086 seconds.
>>
```

“20x more workers,
but the same time?”

How big of a Pool? . . .

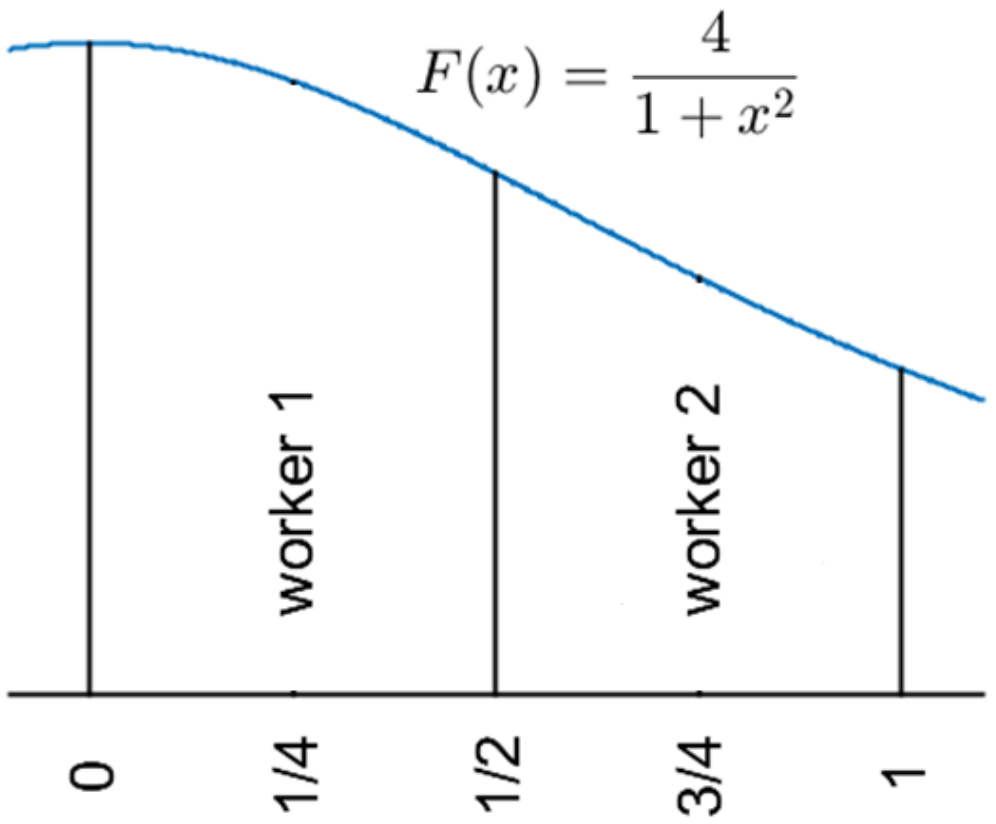
```
>> % Pool of 200 workers
>> tic, p = c.parpool(200); toc
Starting parallel pool (parpool) using the 'teton R2021b' profile ...

additionalSubmitArgs =

    '--ntasks=200 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00'

Connected to the parallel pool (number of workers: 200).
Elapsed time is 47.164235 seconds.
>>
>> tic, parfor idx = 1:4000, pause(3), end, toc
Elapsed time is 60.507177 seconds.
>> % Equivalent hours if run serially
>> 4000 * 3 / 60 / 60

ans =

    3.3333

>>
```

Change directories to workshop

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

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

if isempty(gcp('nocreate')), c.parpool(sz); end

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;
% Required properties
c.AdditionalProperties.WallTime = '1:00:00';
c.AdditionalProperties.AccountName = 'matlabworkshop2022';
% Required for this workshop only
c.AdditionalProperties.Reservation = 'MATLAB_WS';
% Optional properties
c.AdditionalProperties.MemUsage = '5G';

if isempty(gcp('nocreate')), c.parpool(20); end

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: 2).
Worker 1:
  Subinterval: [0   , 0.5 ]
Worker 2:
  Subinterval: [0.5 , 1   ]
Worker 1:
  Subinterval: [0   , 0.5 ]   Integral: 1.85459
Worker 2:
  Subinterval: [0.5 , 1   ]   Integral: 1.287
pi           : 3.141592653589793116
Approximation: 3.141592653589793116
Error        : 0
```

```
>> calc_pi_multi_node
Starting parallel pool (parpool) using the 'teton R2021b' profile ...

additionalSubmitArgs =

    '--ntasks=20 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworks

Connected to the parallel pool (number of workers: 20).
Worker  1:
  Subinterval: [0   , 0.025]
Worker  2:
  Subinterval: [0.025, 0.05]
Worker  3:
  Subinterval: [0.05, 0.075]

. . .

Worker 18:
  Subinterval: [0.425, 0.45]   Integral: 0.0839331
Worker 20:
  Subinterval: [0.525, 0.55]   Integral: 0.0775848
pi           : 3.141592653589793116
Approximation: 3.141592653589793116
Error        : 0
```

GPUs

GPU Availability

Partition	GPU Type	Device ID	Nodes	CUDA Cores	GPU Memory Size (GB)	Compute Capability
moran	GeForce GTX Titan	[1-2]	mdbg01	2688	6	3.5
moran	GeForce GTX Titan X	0 [2-3]	mdbg01 mdbg02	3072	12	5.2
moran	Tesla K20m	[0-1] 1	m[025-32], m[075-82], m086, m268	2496	4.7	3.5
moran	Tesla K20Xm	[0-1] 0	m219/20/27/28, m235/36, m243/4, m251/2/9, m260/7, m268	2688	5.7	3.5
moran	Tesla K40c	[0-1]	mdbg02	2880	11.4	3.5
moran-bigmem-gpu	Tesla K80	[0-7]	mbm[01-02]	2496	11.4	3.7
teton-gpu	Tesla P100	[0-1]	tbm[03-10]	3584	16	6.0

The following two GPU nodes are reserved for AI use. These are special nodes running Ubuntu and CUDA 11.0.

Partition	GPU	Devices	Nodes	CUDA Cores	Tensor Cores	GPU Memory Size (GB)	Compute Capability
dgx	Tesla V100	[0-7]	mdgx01	5120	640	16	7.0
dgx	Tesla V100	[0-7]	tdgx01	5120	640	32	7.0

Start pool with GPU node

```
>> % Start a parallel pool with a GPU
>> c = parcluster;
>> c.AdditionalProperties.GpusPerNode = 1;
>> c.AdditionalProperties.AccountName = 'matlabworkshop2022';
>> c.AdditionalProperties.Reservation = '';
>>
>> p = c.parpool(1);
Starting parallel pool (parpool) using the 'teton R2021b' profile ...

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00

Connected to the parallel pool (number of workers: 1).
>>
```

Tesla K80 – Card we were assigned

```
>> spmd, gpuDevice, end
Worker 1:
  Warning: Support for GPU devices with Compute Capability 3.7 will be removed in a future MATLAB release.
  CUDADevice with properties:

                      Name: 'Tesla K80'
                     Index: 1
         ComputeCapability: '3.7'
            SupportsDouble: 1
             DriverVersion: 11.2000
            ToolkitVersion: 11
        MaxThreadsPerBlock: 1024
          MaxShmemPerBlock: 49152
        MaxThreadBlockSize: [1024 1024 64]
               MaxGridSize: [2.1475e+09 65535 65535]
                 SIMDWidth: 32
               TotalMemory: 1.1997e+10
           AvailableMemory: 1.1852e+10
       MultiprocessorCount: 13
              ClockRateKHz: 823500
               ComputeMode: 'Default'
      GPUOverlapsTransfers: 1
    KernelExecutionTimeout: 0
          CanMapHostMemory: 1
           DeviceSupported: 1
           DeviceAvailable: 1
            DeviceSelected: 1
```

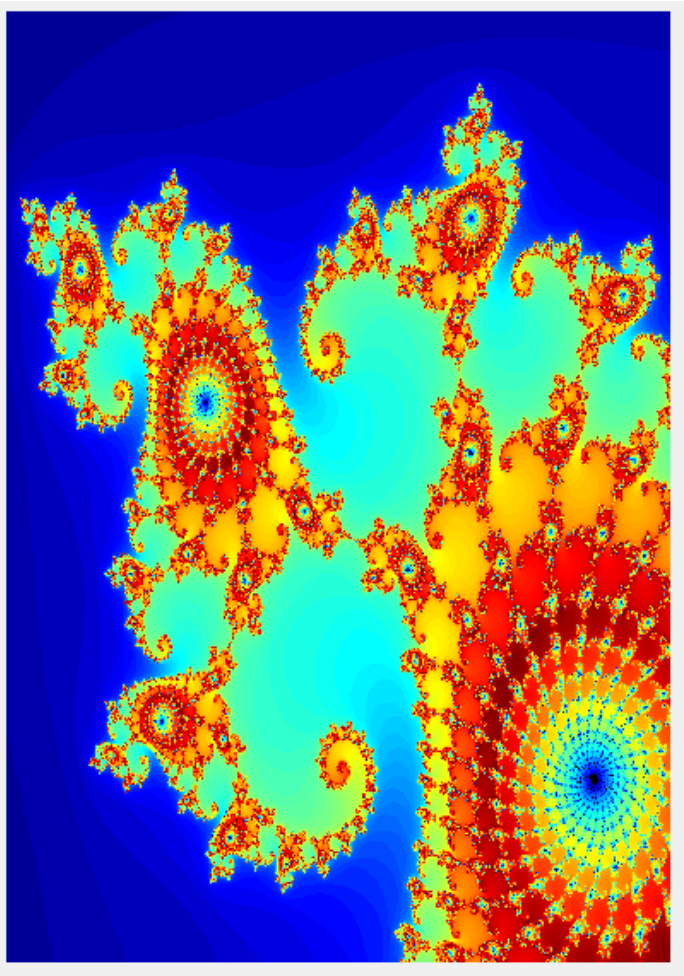
Note

GPU computing requirements:

- MATLAB® supports NVIDIA® GPU architectures with compute capability 3.5 to 8.x.

Example: mandelbrot (1)

```
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
```



Example: mandelbrot (2)

```
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
```

```
>> spmd, mandelbrot_example, end
Lab 1:
  CPU time: 467.88
  GPU time: 3.32
>>
```

Example: FFT (1)

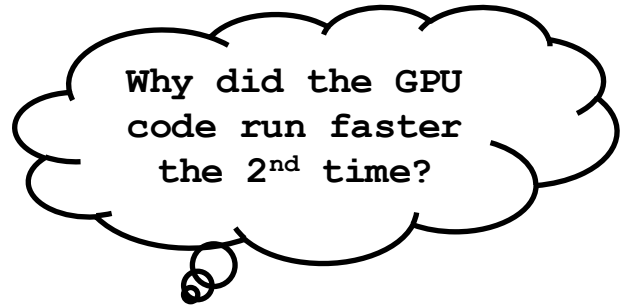
```
function [time_cpu, time_gpu] = calc_fft_cpu_gpu(N)

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
```

Example: FFT (2)

```
>> % 0.5 GB Array
>> sz = 2^13 * 2^13 * 8 / 1024^3

sz =

    0.5000

>>
>> % GPU memory (GB)
>> spmd, d = gpuDevice; d.AvailableMemory/1024^3, end
Worker 1:

  ans =

      5.4119

>>
```

Example: FFT (3)

```
>> spmd, [cpu_t, gpu_t] = calc_fft_cpu_gpu(2^13); end
Worker 1:
  Total time on CPU: 1.0573
  GPU FFT: 0.14827
  Total time on GPU: 3.4917
  FFT speed improvement: 7.1309
  Total speed improvement: 0.30281
    Name                Size                 Bytes  Class       Attributes

    matrix_cpu       8192x8192            536870912  double
    matrix_gpu       8192x8192            536870912  gpuArray
>>
>> % Why will the GPU run faster the second time?
>> spmd, [cpu_t, gpu_t] = calc_fft_cpu_gpu(2^13); end
Worker 1:
  Total time on CPU: 0.87341
  GPU FFT: 0.004067
  Total time on GPU: 1.0257
  FFT speed improvement: 214.7558
  Total speed improvement: 0.8515
    Name                Size                 Bytes  Class       Attributes

    matrix_cpu       8192x8192            536870912  double
    matrix_gpu       8192x8192            536870912  gpuArray
```



Turnoff GPU requests when you don’t need them anymore

```
>> c.AdditionalProperties.GpusPerNode = 0;
>> c.AdditionalProperties.Reservation = 'MATLAB_WS';
```

Interactively: with batch jobs

`batch`

Exercise: “Hello, World!”

```
>> % Submit job to find out where MATlAB is running
>> c = parcluster;
>> j = c.batch(@pwd,1,{});

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00

>> % Check the state of the job
>> j.State

ans =

    'finished'

>> % Fetch the results
>> j.fetchOutputs{:}

ans =

    '/pfs/tc1/home/dpietrus/Documents/MATLAB/matlab-workshop-files'
```

Set the `batch` *CurrentFolder* argument to change default location

Exercise: Calculate π

```
>> % Submit calc_pi job
>> c = parcluster;
>>
>> % Request 19 workers
>> j = c.batch(@calc_pi,0,{},'Pool',19);

additionalSubmitArgs =

    '--ntasks=20 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>>
```

"If my Pool is size 19, why am I requesting 20 cores?"

Fetch the results

```
>> % Request 19 workers
>> j = c.batch(@calc_pi,0,{},'Pool',19);

additionalSubmitArgs =

    '--ntasks=20 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>> % Check the state of the job
>> j.State

ans =

    'finished'

>> % Fetch the results
>> j.fetchOutputs{:}
>>
```

“Where’s the output?”

Fetch the diary

```
>> j.diary
--- Start Diary ---
Worker  1:
  Subinterval: [0   , 0.0526316]
Worker  2:
  Subinterval: [0.0526316, 0.105263]
Worker  3:
  Subinterval: [0.105263, 0.157895]
Worker  4:
  Subinterval: [0.157895, 0.210526]
Worker  5:
  Subinterval: [0.210526, 0.263158]
Worker  6:
  Subinterval: [0.263158, 0.315789]
Worker  7:
  Subinterval: [0.315789, 0.368421]
Worker  8:
  Subinterval: [0.368421, 0.421053]
Worker 14:
  Subinterval: [0.684211, 0.736842]   Integral: 0.139906
Worker 15:
  Subinterval: [0.736842, 0.789474]   Integral: 0.133051
Worker 16:
  Subinterval: [0.789474, 0.842105]   Integral: 0.126414
Worker 17:
  Subinterval: [0.842105, 0.894737]   Integral: 0.120027
Worker 18:
  Subinterval: [0.894737, 0.947368]   Integral: 0.113912
Worker 19:
  Subinterval: [0.947368, 1   ]   Integral: 0.108082
pi           : 3.141592653589793116
Approximation: 3.141592653589793560
Error        : 4.44089e-16

--- End Diary ---
>>
```

What gets “returned”?

- Function output
- Diary
- Saved files

Example

"What size Pool am I running?"

```
function [t, A] = test_fcn(sims)

disp('Start sim')

A = nan(sims,1);
t0 = tic;
parfor idx = 1:sims
    A(idx) = idx;
    pause(0.5)
    idx
end
t = toc(t0);

disp('Finished')

save RESULTS A
```

Job submission

```
>> j = c.batch(@test_fcn,1,{100},'Pool',10);

additionalSubmitArgs =

    '--ntasks=11 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>>
```

Fetch the results

```
function [t, A] = test_fcn(sims)
```

```
c.batch(@test_fcn,1,{100},
```

```
>> % Fetch the results
>> j.fetchOutputs{:}
ans =
    7.0650
>>
```

"Where's *A*?"

Fetch the diary

```
% View the diary
j.diary
--- Start Diary ---
Start sim

ans =

     2


ans =

     4

  ...

ans =

   100


ans =

    98

Finished

--- End Diary ---
```

```
function [t, A] = test_fcn(sims)

disp('Start sim')

A = nan(sims,1);
t0 = tic;
parfor idx = 1:sims
    A(idx) = idx;
    pause(0.5)
    idx
end
t = toc(t0);

disp('Finished')

save RESULTS A
```

Save files

"Where does **RESULTS** get written to?"

```
function [t, A] = test_fcn(sims)

disp('Start sim')

A = nan(sims,1);
t0 = tic;
parfor idx = 1:sims
    A(idx) = idx;
    pause(0.5)
    idx
end
t = toc(t0);

disp('Finished')

save RESULTS A
```

Submitting scripts, instead of functions

```
>> z = 10;
>>
>> % Submit a script (instead of a function)
>> j = c.batch('x = 3; y = 4, z');

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1

>>
```

Loading variables to local workspace

```
>> z = 10;
>>
>> % Submit a script (instead of a function)
>> j = c.batch('x = 3; y = 4, z');

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
```

"If we cleared `z`, then why does `who` display it?

```
>> clear z
>> who

Your variables are:

c  j

>> % Check the state of the job
>> j.State

ans =

    'finished'

>> % Load variables from the job
>> j.load
>> who

Your variables are:

ans  c    j    x    y    z

>>
```

"I'll pass all of the variables in my local workspace to all of the workers. Then I'll receive everything the workers generate and load it back to my local workspace."

```
>> z = 10;
>>
>> % Submit a
>> j = c.batch
```

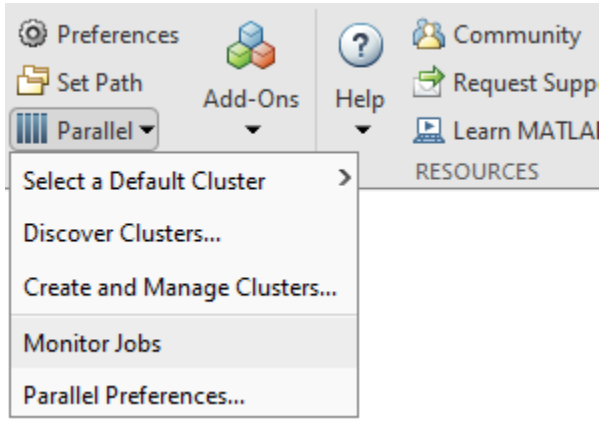
Getting the diary of scripts

```
>> j.diary
--- Start Diary ---

y =

     4


z =

    10


--- End Diary ---
>>
```

When has my job run and finished?

```
>> % Get email notification when job has finished
>> c.AdditionalProperties.EmailAddress = 'user-id@uwyo.edu';
>>
>> j = c.batch(@test_fcn,1,{100},'Pool',10);

additionalSubmitArgs =

    '--ntasks=11 --cpus-per-task=1 --ntasks-per-core=1 --mail-user=user-id@uwyo.edu

>>
```

Retrieving past jobs



Job Monitor

Select Profile: teton R2021b (default) — Show jobs from all users

ID	Username	Submit Time	Finish Time	Tasks	State	Description
61	dpietrus	Tue Mar 29 15:15:25 MDT 2022	Tue Mar 29 15:15:53 MDT 2022	1	finished	Batch job running function
62	dpietrus	Tue Mar 29 15:16:54 MDT 2022	Tue Mar 29 15:17:27 MDT 2022	1	finished	Batch job running function
63	dpietrus	Tue Mar 29 15:19:56 MDT 2022	Tue Mar 29 15:20:44 MDT 2022	20	finished	Batch job running function
64	dpietrus	Tue Mar 29 15:42:39 MDT 2022	Tue Mar 29 15:43:40 MDT 2022	11	finished	Batch job running function
65	dpietrus	Tue Mar 29 15:46:28 MDT 2022	Tue Mar 29 15:46:50 MDT 2022	1	finished	Batch job running script
66	dpietrus	Tue Mar 29 15:49:41 MDT 2022	Tue Mar 29 15:50:28 MDT 2022	1	finished	Batch job running script
67	dpietrus	Tue Mar 29 15:54:06 MDT 2022	Tue Mar 29 15:54:36 MDT 2022	4	finished	Batch job running function
68	dpietrus	Tue Mar 29 15:57:41 MDT 2022	Tue Mar 29 15:58:15 MDT 2022	11	finished	Batch job running function

Cancel
Delete
Show Details
Show Errors
Show Warnings
Show Diary
Fetch Outputs

Last updated at Tue Mar 29 16:02:22 MDT 2022

Auto update: Every 5 minutes — Update Now

Keep cluster files minimal: delete jobs

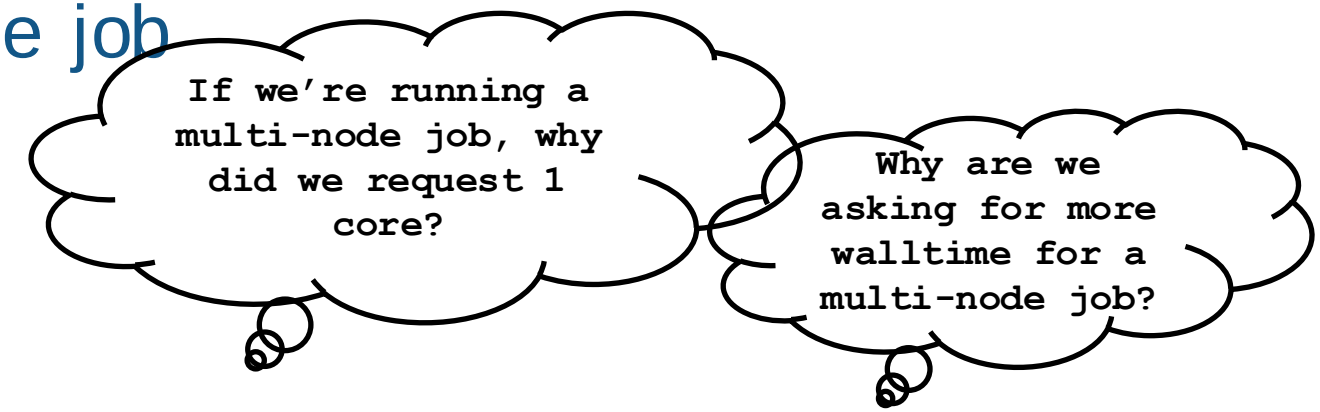
- As a good practice, delete jobs you no longer need

```
>> % Finished with the job, delete it to cleanup list of jobs
>> j.delete
>>
```

Noninteractively: in a SLURM job script

`sbatch`

Submit single-node job/multi-node job




```
#!/bin/sh

#SBATCH -n 1                          # 1 instance of MATLAB
#SBATCH --cpus-per-task=8             # 8 cores per instance
#SBATCH --mem-per-cpu=4gb             # 4 GB RAM per core
#SBATCH --time=00:10:00               # 10 minutes
#SBATCH -A matlabworkshop2022         # Account Name
#SBATCH --reservation=MATLAB_WS       # Just for this Workshop

# Add MATLAB to system path
module load matlab/R2021b

# 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
#SBATCH -A matlabworkshop2022         # Account Name
#SBATCH --reservation=MATLAB_WS       # Just for this Workshop

# Add MATLAB to system path
module load matlab/R2021b

# Run code
matlab -batch calc_pi_multi_node
```

matlab-multi-node.slurm

Job submission

```
[dpietrus@tlog2 matlab-workshop-files]$ sbatch matlab-single-node.slurm
Submitted batch job 2338906
[dpietrus@tlog2 matlab-workshop-files]$
[dpietrus@tlog2 matlab-workshop-files]$ sbatch matlab-multi-node.slurm
Submitted batch job 2338907
[dpietrus@tlog2 matlab-workshop-files]$
```

Local pool of workers

```
[dpietrus@tlog2 matlab-workshop-files]$ head slurm-2338906.out

        Trial License -- for use to evaluate programs for possible

Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 8).
Worker 1:
  Subinterval: [0   , 0.125]
Worker 2:
  Subinterval: [0.125, 0.25]
Worker 3:
[dpietrus@tlog2 matlab-workshop-files]$ tail slurm-2338906.out
  Subinterval: [0.5 , 0.625]   Integral: 0.379807
Worker 6:
  Subinterval: [0.625, 0.75]   Integral: 0.339607
Worker 7:
  Subinterval: [0.75, 0.875]   Integral: 0.301316
Worker 8:
  Subinterval: [0.875, 1   ]   Integral: 0.266273
pi            : 3.1415926535589793116
Approximation: 3.1415926535589792672
Error         : 4.44089e-16
[dpietrus@tlog2 matlab-workshop-files]$
```

```
c = parcluster('local');

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

if isempty(gcp('nocreate')), c.parpool(sz), end
```

```
#!/bin/sh

#SBATCH -n 1                          # 1 instance of MATLAB
#SBATCH --cpus-per-task=8             # 8 cores per instance
#SBATCH --mem-per-cpu=4gb             # 4 GB RAM per core
#SBATCH --time=00:10:00               # 10 minutes
#SBATCH -A matlabworkshop2022         # Account Name
#SBATCH --reservation=MATLAB_WS       # Just for this Workshop

# Add MATLAB to system path
module load matlab/R2021b

# Run code
matlab -batch calc_pi
```

Multi-node pool of workers

```
[dpietrus@tlog2 matlab-workshop-files]$ head slurm-2338907.out

        Trial License -- for use to evaluate programs for possible purchase as an end-user only.

Starting parallel pool (parpool) using the 'teton R2021b' profile ...

additionalSubmitArgs =

    '--ntasks=20 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00 --mem-per-cpu=5G'

Connected to the parallel pool (number of workers: 20).
[dpietrus@tlog2 matlab-workshop-files]$ tail slurm-2338907.out
  Subinterval: [0.8 , 0.85]   Integral: 0.119012
Worker 18:
  Subinterval: [0.85, 0.9 ]   Integral: 0.113284
Worker 19:
  Subinterval: [0.9 , 0.95]   Integral: 0.107791
Worker 20:
  Subinterval: [0.95, 1   ]   Integral: 0.102542
pi            : 3.1415926535897931
Approximation: 3.1415926535897931
Error         : 0
[dpietrus@tlog2 matlab-workshop-files]$
```

Debugging and Troubleshooting

Scheduler ID

```
>> j = c.batch(@pwd,1,{});

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>> % Job ID vs Scheduler ID
>> j.ID

ans =

    70

>> j.getTaskSchedulerIDs{1}

ans =

    '2338912'

>>
```

Example: Errored submissions (1)

```
>> % Undefined function
>> j = c.batch(@invalid_fcn,0,{});
Warning: Unable to calculate the dependencies of the files:
  invalid_fcn

because:
File, function or class "invalid_fcn" may not exist. Neither WHICH nor EXIST could find an
exact, case-sensitive match. Please check the spelling of the name, and that any required
directories are on the MATLAB path.
> In parallel.internal.apishared/AttachedFiles/calculateAttachedFiles (line 315)
In parallel.internal.cluster.AbstractBatchHelper>iWrapCalculateAttachedFiles (line 819)
In parallel.internal.cluster.AbstractBatchHelper>@(f)iWrapCalculateAttachedFiles(f,function
In parallel.internal.cluster.AbstractBatchHelper>iCalculateTaskDependencies (line 790)
In parallel.internal.cluster/AbstractBatchHelper/getCreateTaskInputs (line 646)
In parallel.internal.cluster/BatchHelper2/doBatch (line 37)
In parallel.Cluster/batch (line 155)

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022 -t 1:00:00 --me

>>
```

Example: Errored submissions (2)

```
>> % Undefined function
>> j.State

ans =

    'finished'

>> j.fetchOutputs{:}
Error using parallel.Job/fetchOutputs (line 1300)
An error occurred during execution of Task with ID 1.

Caused by:
    Unrecognized function or variable 'invalid_fcn'.

>>
```

Fetching outputs, even with no outputs returned, to get the error message

Logfile: Single core job

```
>> j = c.batch(@pwd,1,{});

additionalSubmitArgs =

    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>> % Retreive log file for single core job
>> c.getDebugLog(j.Tasks(1))
LOG FILE OUTPUT:
Executing: /pfs/tc1/apps/el7-x86_64/u/opt/matlab/R2021b/bin/worker
Exiting with code: 0

>>
```

Logfile: Multi-core job

```
>> j = c.batch(@pwd,1,{},'Pool',2);

additionalSubmitArgs =

    '--ntasks=3 --cpus-per-task=1 --ntasks-per-core=1 -A matlabworkshop2022

>> % Retreive log file for multi-core job
>> c.getDebugLog(j)
LOG FILE OUTPUT:
The scheduler has allocated the following nodes to this job:
m003
"/pfs/tc1/apps/el7-x86_64/u/opt/matlab/R2021b/bin/mw_mpiexec" -l -n 3 "/pfs/
[0] Sending a stop signal to all the labs...
[0] Parallel pool is shutting down.[0]
Exiting with code: 0

>>
```