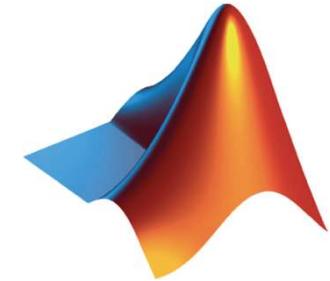


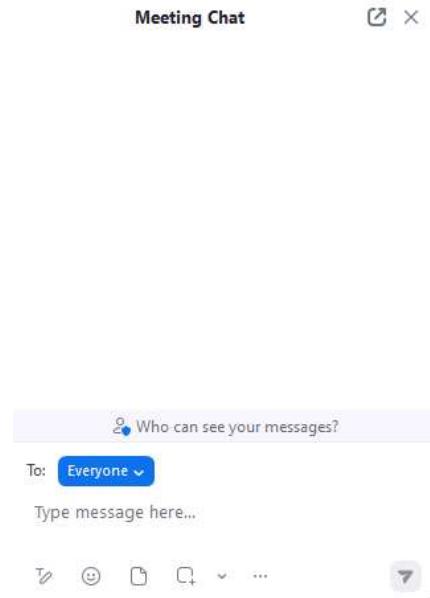
WORKSHOP: Parallel Computing with MATLAB (Part II)

Raymond Norris
Application Engineer
November 2023



Meeting Chat

- Please send chats to Everyone



Agenda

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



Agenda

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



Overview

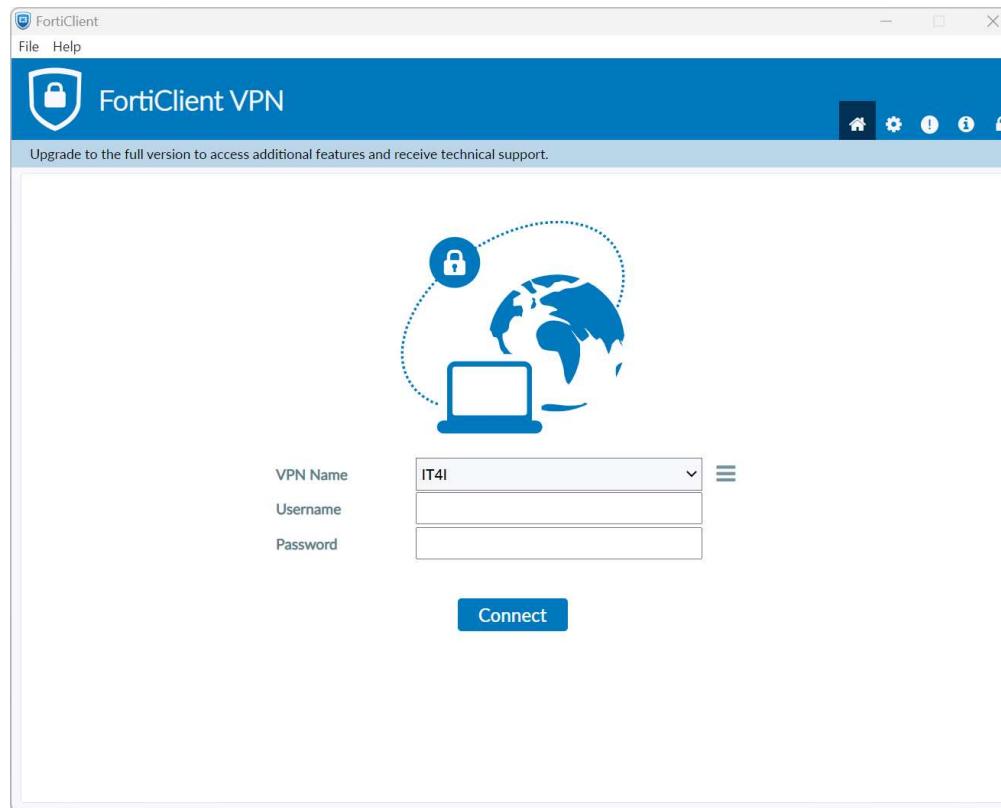
- How to configure MATLAB to submit multi-node jobs to the HPC cluster
- The job submission workflow
- Ways to tune job submissions to the HPC cluster
- How to optimize job submissions
- Troubleshooting job submission techniques
- Submitting interactive and batch jobs

A few notes about today's workshop...

- The workflow and examples are about process, not performance
- Cluster documentation
 - <https://docs.it4i.cz/karolina/introduction>
- MATLAB User Guide
 - <https://docs.it4i.cz/software/numerical-languages/matlab>
- Requirements
 - Account on Karolina cluster
 - VPN client: <https://docs.it4i.cz/general/accessing-the-clusters/vpn-access>



VPN

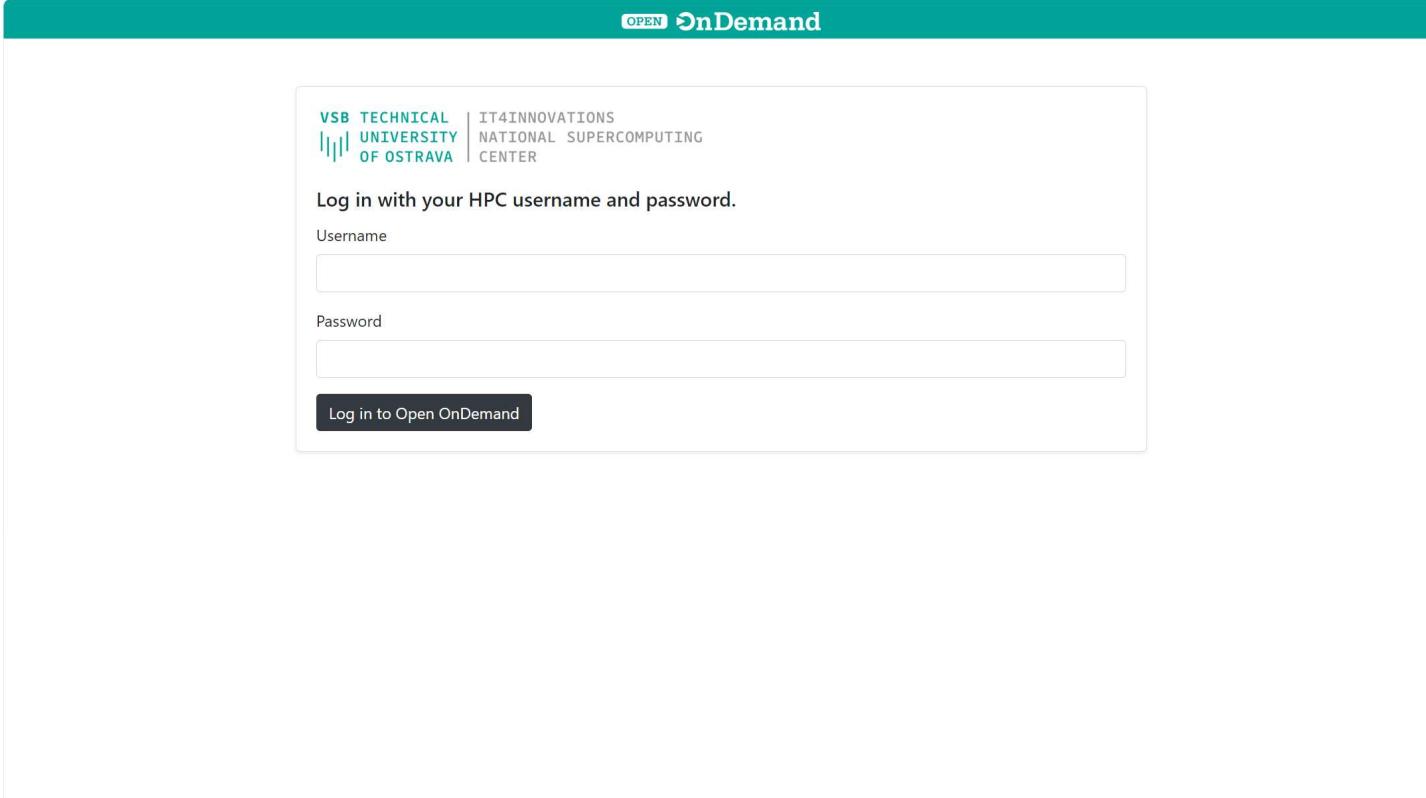


<https://docs.it4i.cz/general/accessing-the-clusters/vpn-access>

Karolina OnDemand: Getting started

- Logging in
- File system
- System prompt
- Starting MATLAB

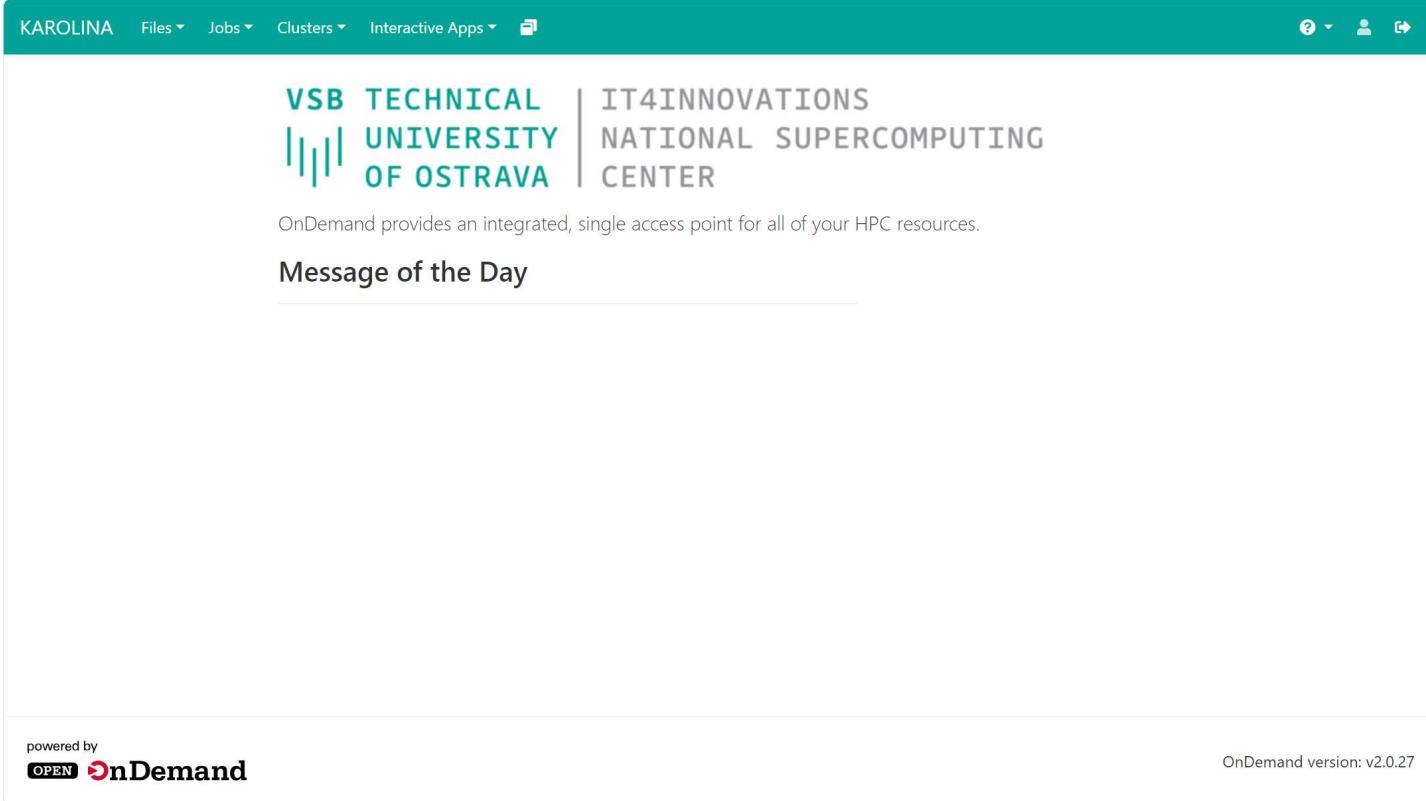
Logging into Karoline OnDemand (1)



The screenshot shows the login interface for the Karoline OnDemand system. At the top, there is a green header bar with the text "OPEN OnDemand". Below the header, the VSB Technical University of Ostrava logo is displayed, followed by the text "IT4INNOVATIONS NATIONAL SUPERCOMPUTING CENTER". A sub-header reads "Log in with your HPC username and password." There are two input fields: "Username" and "Password", each with a corresponding text input box. Below these fields is a dark blue button labeled "Log in to Open OnDemand".

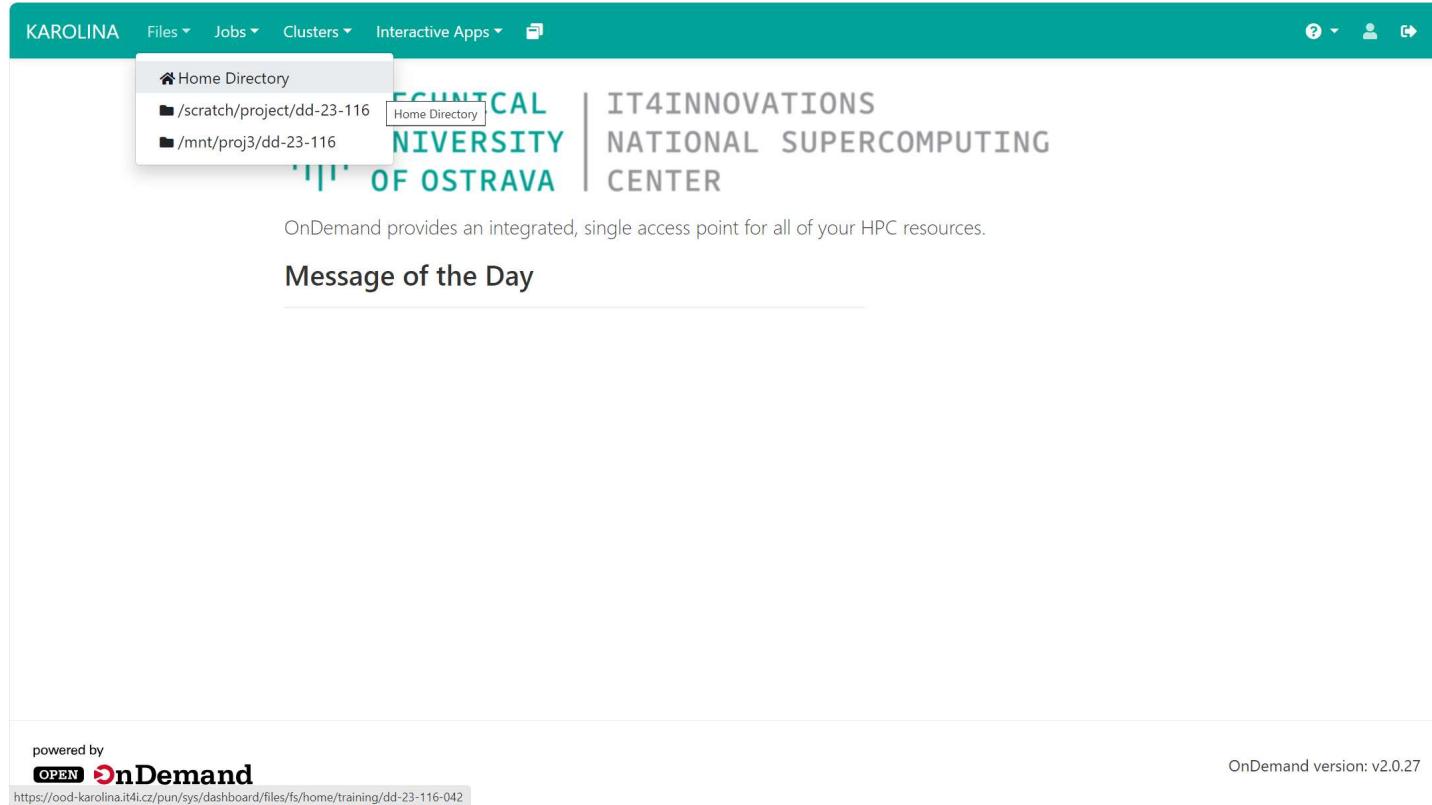
<https://ood-karolina.it4i.cz>

Logging into Karoline OnDemand (2)



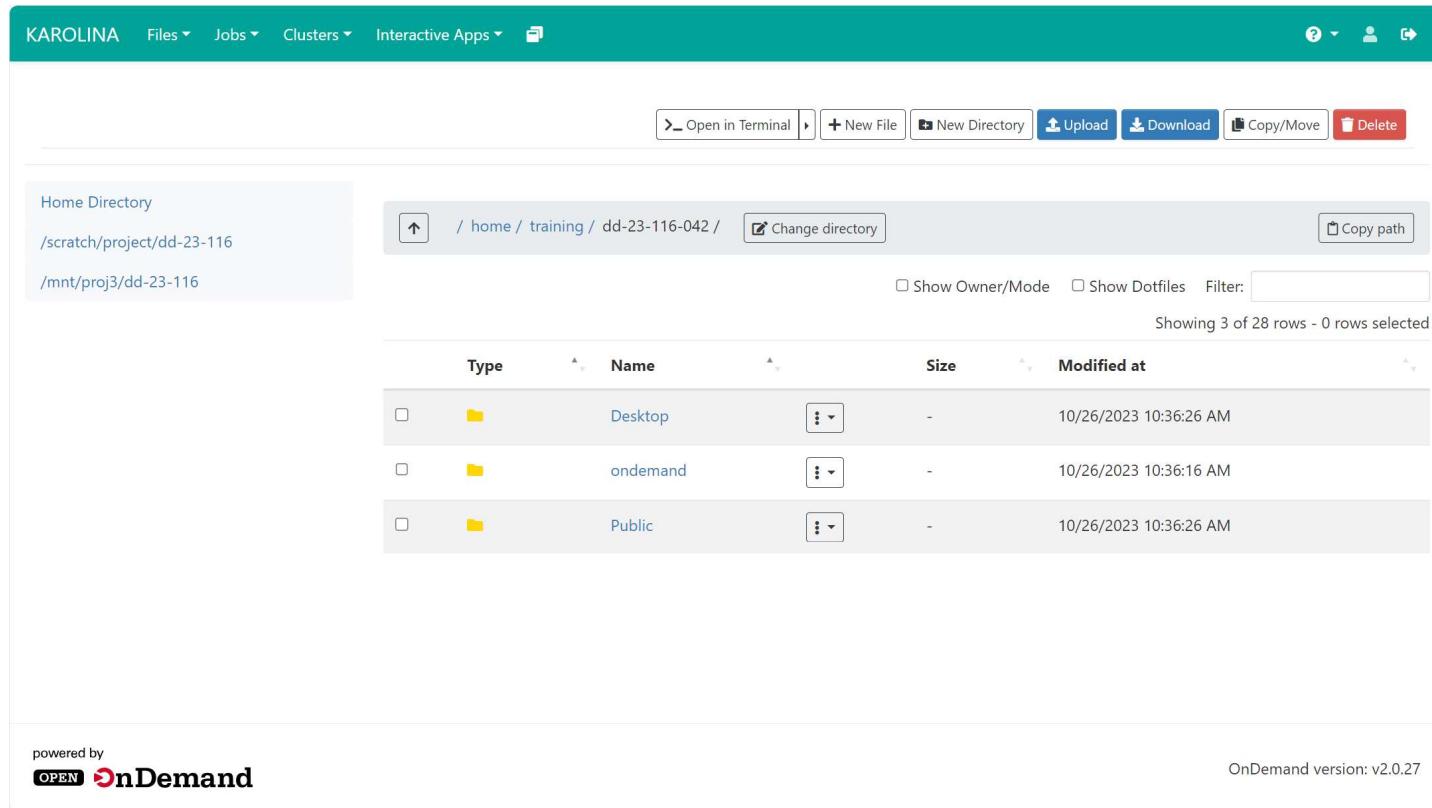
The screenshot shows the Karoline OnDemand login page. At the top, there is a navigation bar with links for KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a help icon. Below the navigation bar, the VSB Technical University of Ostrava logo is displayed next to the IT4Innovations National Supercomputing Center logo. A message states: "OnDemand provides an integrated, single access point for all of your HPC resources." Below this, a section titled "Message of the Day" is shown, which is currently empty. At the bottom left, it says "powered by OPEN OnDemand". At the bottom right, it indicates "OnDemand version: v2.0.27".

File system (1)



The screenshot shows the OnDemand HPC interface. At the top, there is a navigation bar with links for KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a help icon. Below the navigation bar, there is a sidebar with a tree view of file paths: Home Directory, /scratch/project/dd-23-116, and /mnt/proj3/dd-23-116. The path /scratch/project/dd-23-116 is currently selected. To the right of the sidebar, there is a logo for the Technical University of Ostrava and IT4INNOVATIONS National Supercomputing Center. Below the logo, a message reads: "OnDemand provides an integrated, single access point for all of your HPC resources." Underneath this message, there is a section titled "Message of the Day". At the bottom left, it says "powered by OPEN OnDemand" and provides a URL: <https://ood-karolina.it4i.cz/pun/sys/dashboard/files/fs/home/training/dd-23-116-042>. At the bottom right, it says "OnDemand version: v2.0.27".

File system (2)



The screenshot shows a web-based file system interface for the KAROLINA system. The top navigation bar includes links for Home Directory, Files, Jobs, Clusters, Interactive Apps, and a help icon. Below the navigation is a toolbar with buttons for Open in Terminal, New File, New Directory, Upload, Download, Copy/Move, and Delete.

The main area displays the current directory path: /home/training/dd-23-116-042/. It also includes a Change directory input field, a Copy path button, and checkboxes for Show Owner/Mode and Show Dotfiles, along with a Filter input field. A message indicates "Showing 3 of 28 rows - 0 rows selected".

A table lists three directories:

Type	Name	Size	Modified at
Folder	Desktop	-	10/26/2023 10:36:26 AM
Folder	ondemand	-	10/26/2023 10:36:16 AM
Folder	Public	-	10/26/2023 10:36:26 AM

At the bottom left, it says "powered by OPEN OnDemand". At the bottom right, it says "OnDemand version: v2.0.27".

Uploading files (1)

KAROLINA Files▼ Jobs▼ Clusters▼ Interactive Apps▼

Open in Terminal New File New Directory Upload Download Copy/Move Delete

Home Directory /scratch/project/dd-23-116 / Change directory Copy path

Show Owner/Mode Show Dotfiles Filter:

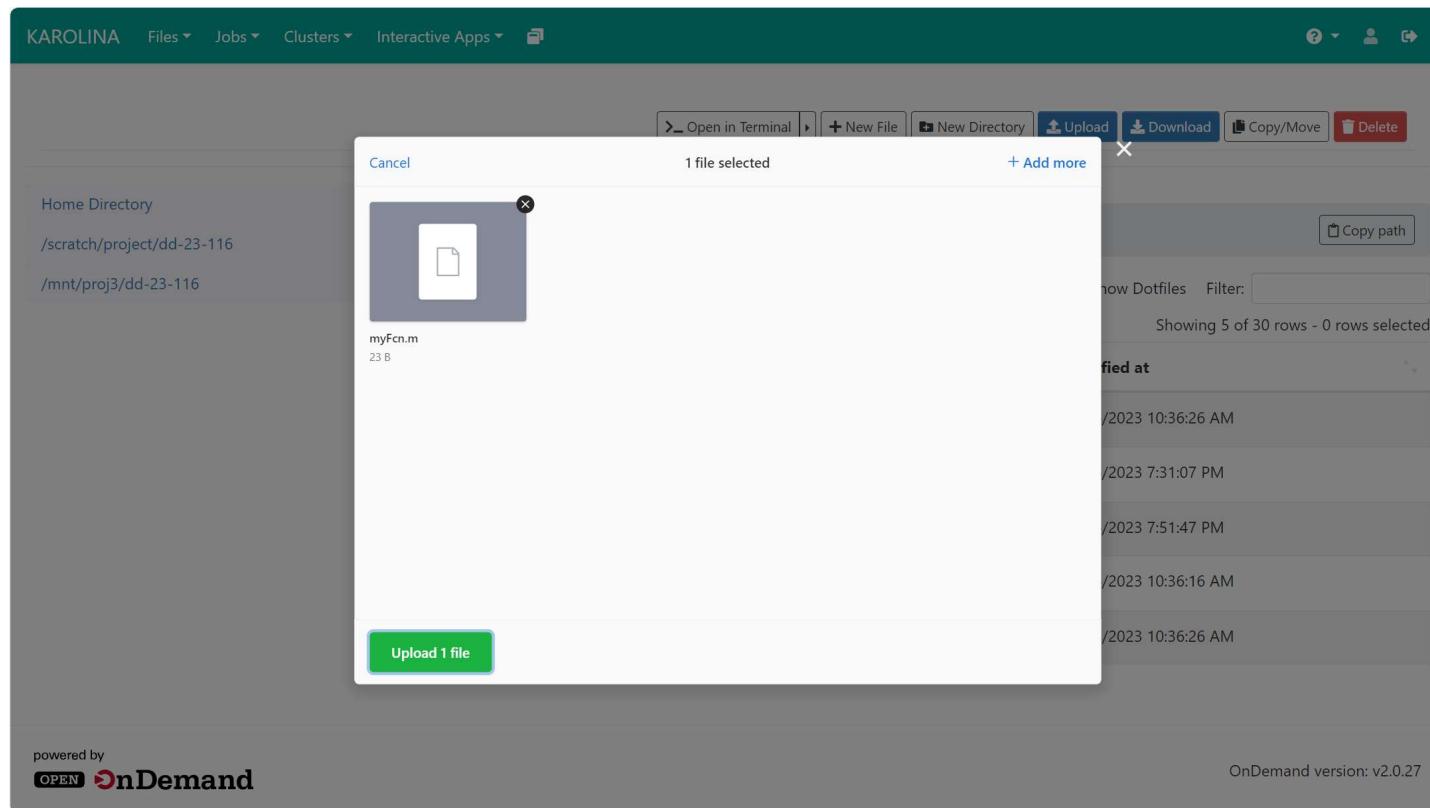
Showing 5 of 30 rows - 0 rows selected

Type	Name	Size	Modified at
Folder	Desktop	-	10/26/2023 10:36:26 AM
Folder	Documents	-	10/26/2023 7:31:07 PM
Folder	Downloads	-	10/26/2023 7:51:47 PM
Folder	ondemand	-	10/26/2023 10:36:16 AM
Folder	Public	-	10/26/2023 10:36:26 AM

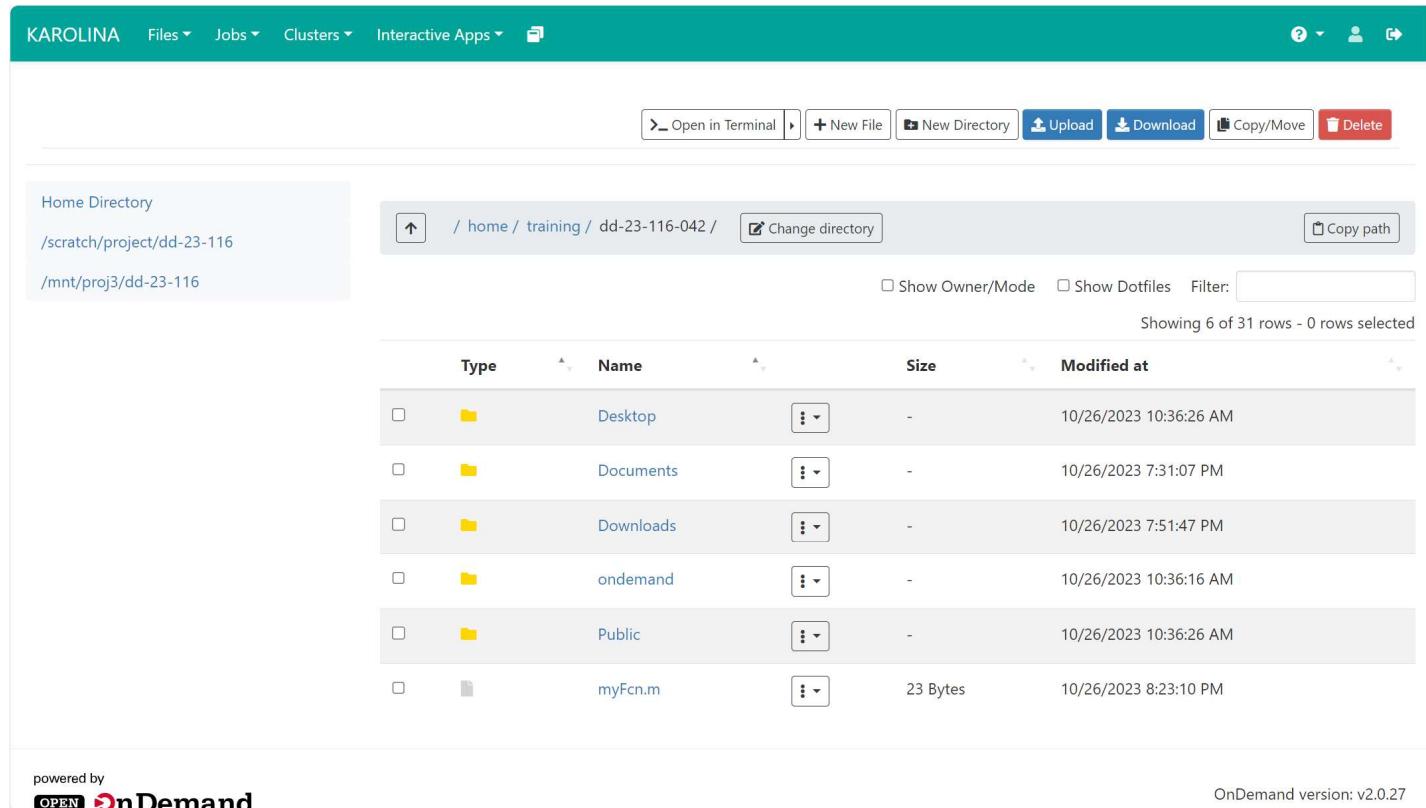
powered by
OPEN OnDemand

OnDemand version: v2.0.27

Uploading files (2)



Uploading files (3)

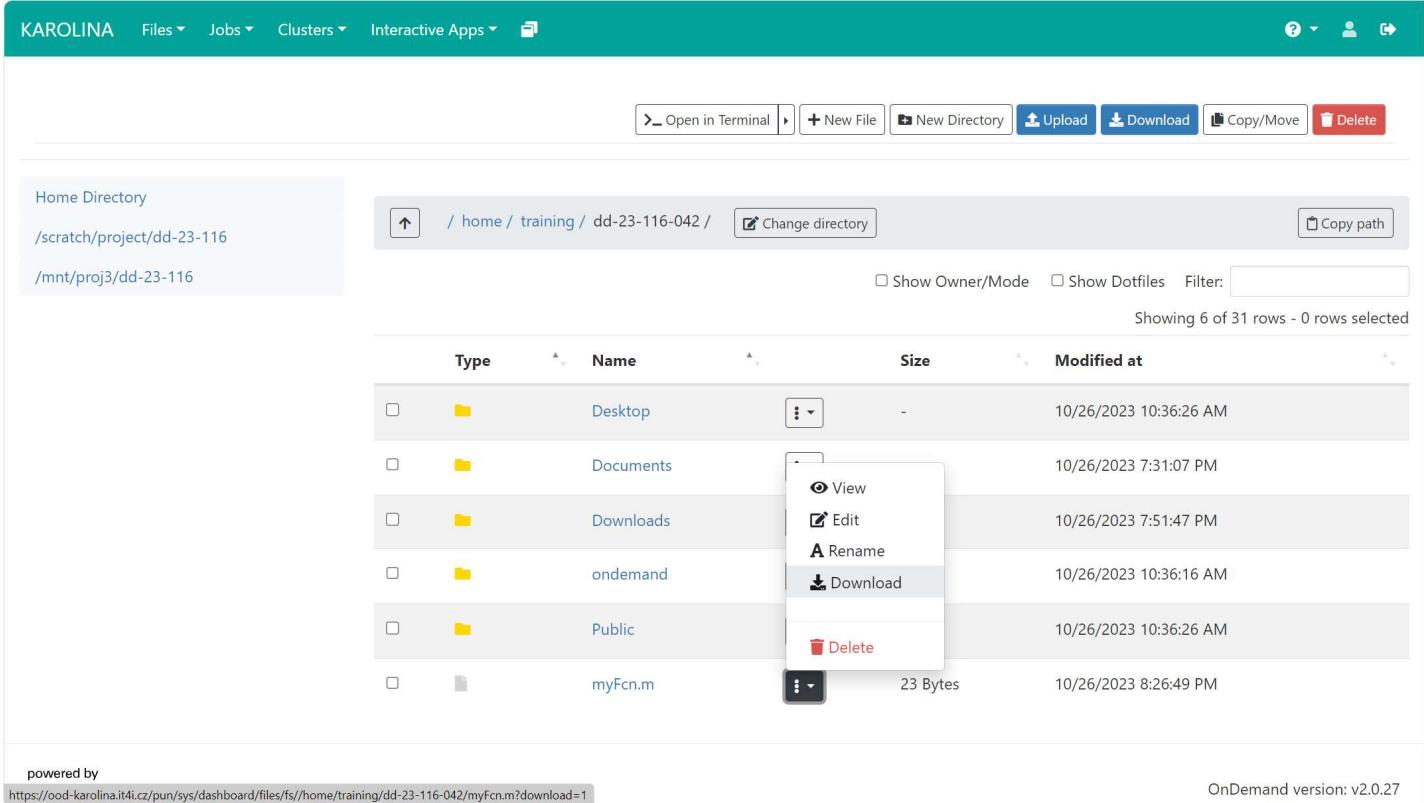


The screenshot shows a file management interface with the following details:

- Header:** KAROLINA, Files, Jobs, Clusters, Interactive Apps, a search bar, and user icons.
- Toolbar:** Open in Terminal, New File, New Directory, Upload (highlighted in blue), Download, Copy/Move, and Delete.
- Breadcrumb:** / home / training / dd-23-116-042 /
- Buttons:** Change directory, Copy path.
- Filters:** Show Owner/Mode, Show Dotfiles, Filter: [empty input].
- Message:** Showing 6 of 31 rows - 0 rows selected.
- Table:** A list of files and directories with columns: Type, Name, Size, and Modified at.

Type	Name	Size	Modified at
Folder	Desktop	-	10/26/2023 10:36:26 AM
Folder	Documents	-	10/26/2023 7:31:07 PM
Folder	Downloads	-	10/26/2023 7:51:47 PM
Folder	ondemand	-	10/26/2023 10:36:16 AM
Folder	Public	-	10/26/2023 10:36:26 AM
File	myFcn.m	23 Bytes	10/26/2023 8:23:10 PM
- Powered By:** OPEN OnDemand
- Version:** OnDemand version: v2.0.27

Downloading files

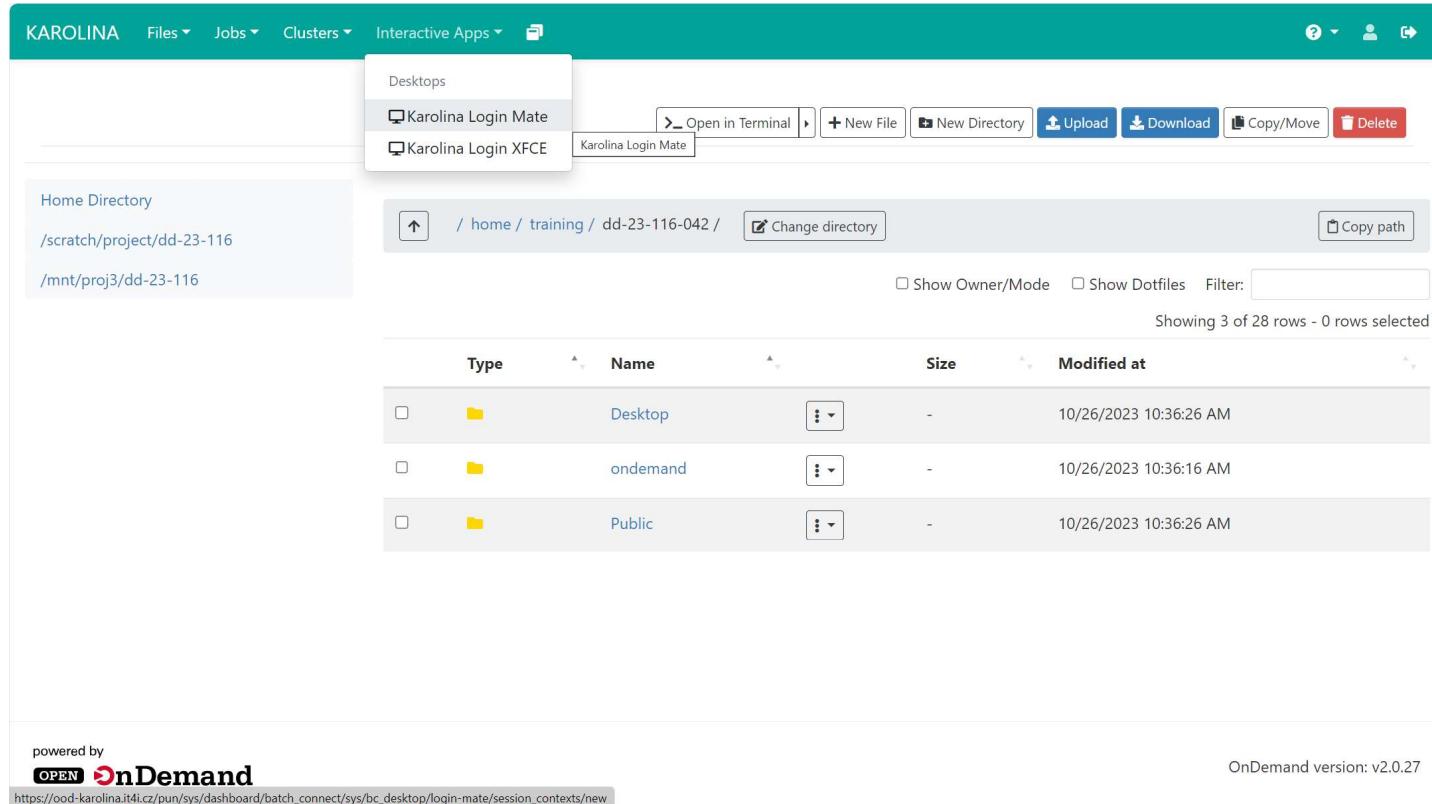


The screenshot shows a file management interface with the following details:

- Header:** KAROLINA Files ▾ Jobs ▾ Clusters ▾ Interactive Apps ▾
- Toolbar:** Open in Terminal, New File, New Directory, Upload, Download, Copy/Move, Delete
- Breadcrumb:** / home / training / dd-23-116-042 /
- Buttons:** Change directory, Copy path
- Filters:** Show Owner/Mode, Show Dotfiles, Filter: [empty]
- Message:** Showing 6 of 31 rows - 0 rows selected
- Table Headers:** Type, Name, Size, Modified at
- Table Data:**

Type	Name	Size	Modified at
Folder	Desktop	-	10/26/2023 10:36:26 AM
Folder	Documents	-	10/26/2023 7:31:07 PM
Folder	Downloads	-	10/26/2023 7:51:47 PM
Folder	ondemand	-	10/26/2023 10:36:16 AM
Folder	Public	-	10/26/2023 10:36:26 AM
File	myFcn.m	23 Bytes	10/26/2023 8:26:49 PM
- Context Menu (for 'myFcn.m'):**
 - View
 - Edit
 - Rename
 - Download** (highlighted)
 - Delete
- Footer:**
 - powered by <https://ood-karolina.it4i.cz/pun/sys/dashboard/files/fs//home/training/dd-23-116-042/myFcn.m?download=1>
 - OnDemand version: v2.0.27

Starting login session (1)

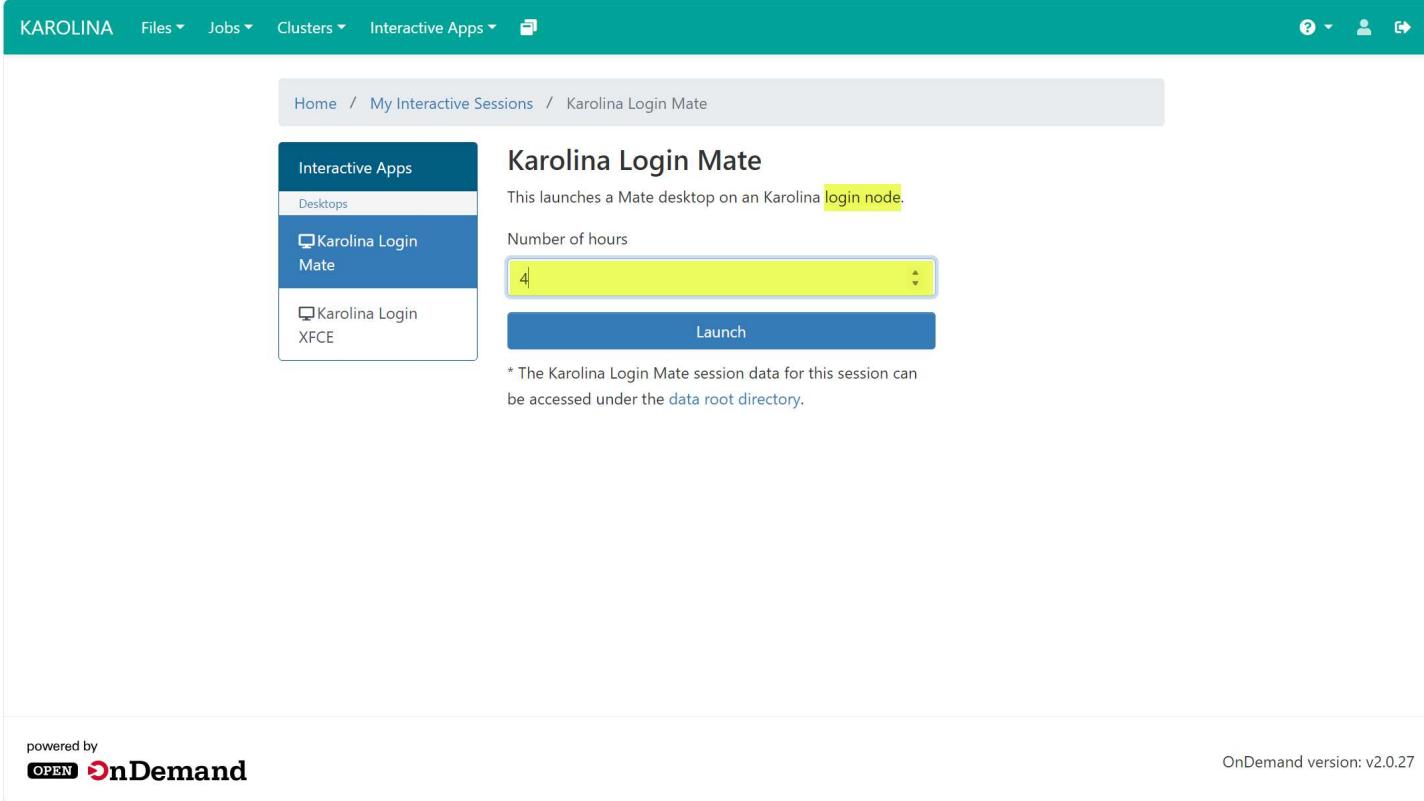


The screenshot shows the KAROLINA interface for starting a login session. At the top, there's a navigation bar with links for KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a user icon. Below the navigation bar is a dropdown menu for "Desktops" containing "Karolina Login Mate" and "Karolina Login XFCE". The main area displays the "Home Directory" path: /home/training/dd-23-116-042. It includes a breadcrumb navigation bar with an up arrow, the current directory path, a "Change directory" button, and a "Copy path" button. There are also checkboxes for "Show Owner/Mode" and "Show Dotfiles" and a "Filter" input field. A message at the bottom indicates "Showing 3 of 28 rows - 0 rows selected". Below this is a table listing three desktop entries:

Type	Name	Size	Modified at
□	Desktop	[-]	10/26/2023 10:36:26 AM
□	ondemand	[-]	10/26/2023 10:36:16 AM
□	Public	[-]	10/26/2023 10:36:26 AM

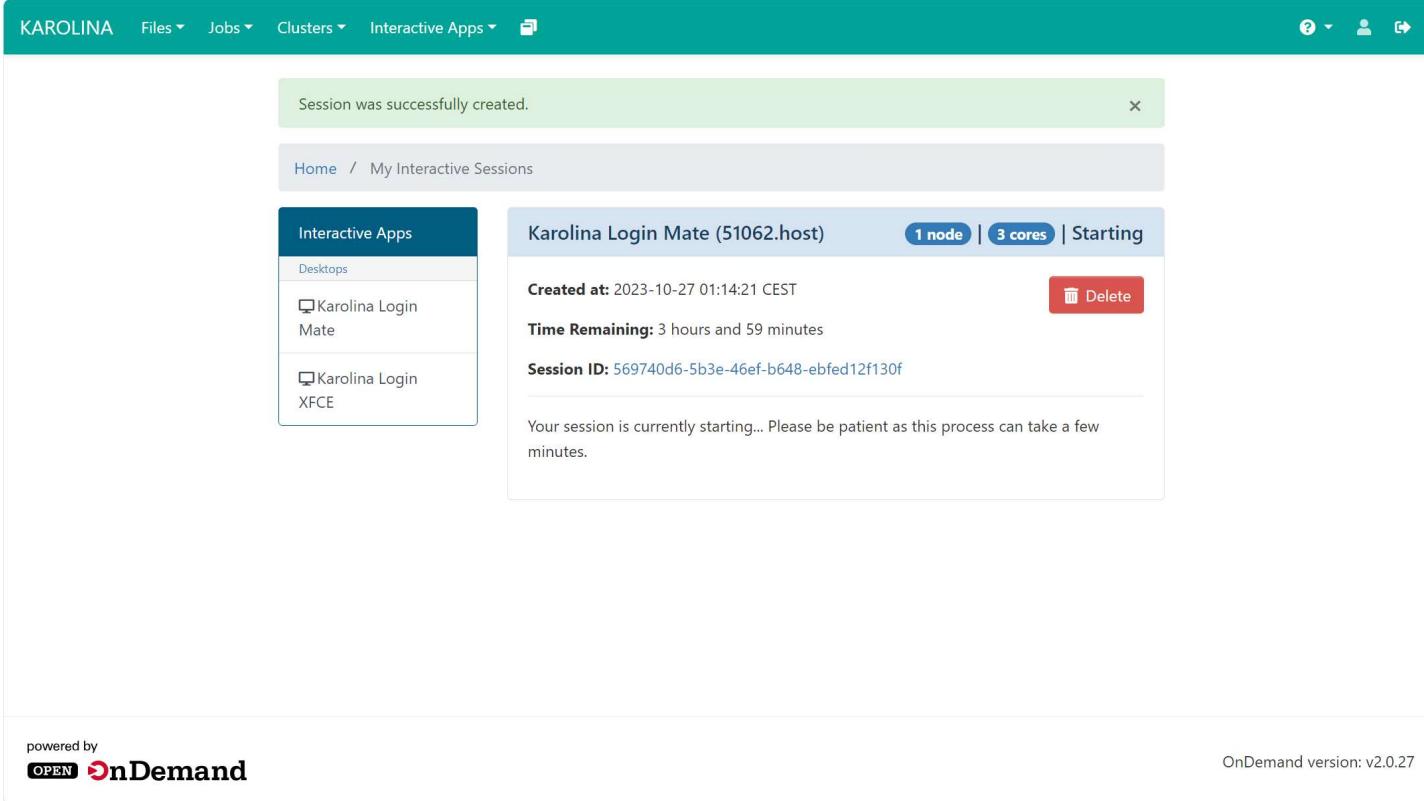
At the bottom left, it says "powered by OPEN OnDemand" with a link to https://ood-karolina.it4i.cz/pun/sys/dashboard/batch_connect/sys/bc_desktop/login-mate/session_contexts/new. At the bottom right, it says "OnDemand version: v2.0.27".

Starting login session (2)



The screenshot shows the KAROLINA web interface. The top navigation bar includes links for KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a help icon. The main content area has a breadcrumb trail: Home / My Interactive Sessions / Karolina Login Mate. On the left, a sidebar under 'Interactive Apps' lists 'Desktops', 'Karolina Login Mate' (which is selected and highlighted in blue), and 'Karolina Login Xfce'. The right side displays the 'Karolina Login Mate' configuration page. It contains a description: 'This launches a Mate desktop on an Karolina login node.', a dropdown menu for 'Number of hours' set to '4', and a large blue 'Launch' button. A note at the bottom states: '* The Karolina Login Mate session data for this session can be accessed under the data root directory.' At the bottom of the page, it says 'powered by OPEN OnDemand' and 'OnDemand version: v2.0.27'.

Starting login session (3)



The screenshot shows the KAROLINA web interface. At the top, there is a navigation bar with links for KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a help icon. Below the navigation bar, a green success message box displays the text "Session was successfully created." A close button (X) is located in the top right corner of this message box.

The main content area shows the "Interactive Apps" section. On the left, there is a sidebar titled "Interactive Apps" with a "Desktops" category containing two items: "Karolina Login Mate" and "Karolina Login XFCE".

On the right, a detailed view of a session is shown. The title of the session is "Karolina Login Mate (51062.host)". Above the session details, it says "1 node | 3 cores | Starting".

Session details include:

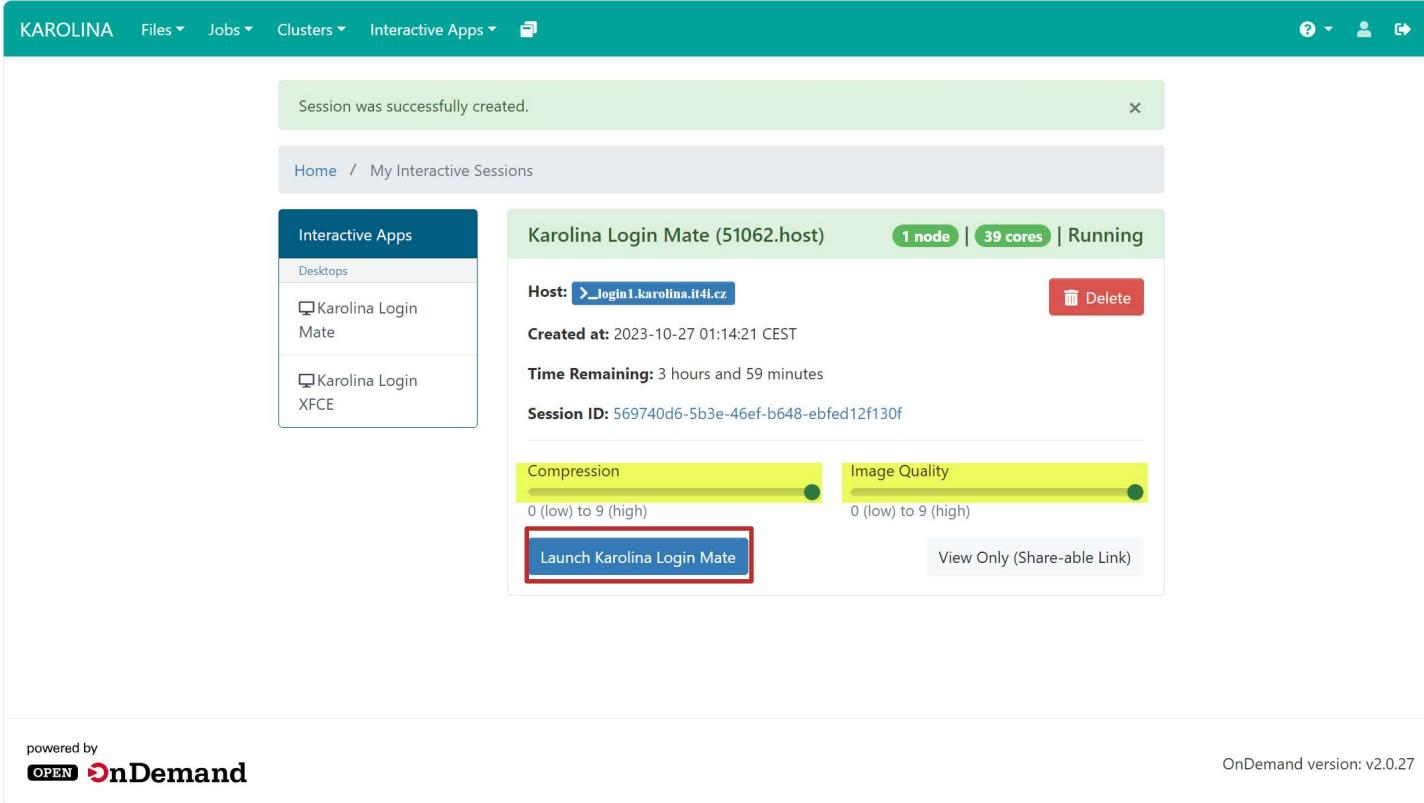
- Created at:** 2023-10-27 01:14:21 CEST
- Time Remaining:** 3 hours and 59 minutes
- Session ID:** 569740d6-5b3e-46ef-b648-ebfed12f130f

A red "Delete" button is located next to the time remaining information.

Below the session details, a message states: "Your session is currently starting... Please be patient as this process can take a few minutes."

At the bottom left, there is a "powered by" section with the "OPEN OnDemand" logo. At the bottom right, the text "OnDemand version: v2.0.27" is displayed.

Starting login session (4)



The screenshot shows the KAROLINA interface with a teal header bar containing navigation links: KAROLINA, Files, Jobs, Clusters, Interactive Apps, and a help icon. Below the header is a green success message box stating "Session was successfully created." The main content area shows a breadcrumb path: Home / My Interactive Sessions. A sidebar on the left lists "Interactive Apps" with options: Desktops, Karolina Login Mate, and Karolina Login XFCE. The central panel displays a session named "Karolina Login Mate (51062.host)" which is "Running" on "1 node | 39 cores". Key details include the host (_login1.karolina.it4i.cz), creation date (2023-10-27 01:14:21 CEST), time remaining (3 hours and 59 minutes), and session ID (569740d6-5b3e-46ef-b648-ebfed12f130f). Two sliders for "Compression" and "Image Quality" are shown, both set to 9 (high). A red box highlights the "Launch Karolina Login Mate" button, and another button below it says "View Only (Share-able Link)". At the bottom, the interface is powered by OPEN OnDemand, and the OnDemand version is v2.0.27.

Session was successfully created.

Home / My Interactive Sessions

Interactive Apps

- Desktops
- Karolina Login Mate
- Karolina Login XFCE

Karolina Login Mate (51062.host) 1 node | 39 cores | Running

Host: [_login1.karolina.it4i.cz](#) Delete

Created at: 2023-10-27 01:14:21 CEST

Time Remaining: 3 hours and 59 minutes

Session ID: 569740d6-5b3e-46ef-b648-ebfed12f130f

Compression 0 (low) to 9 (high)

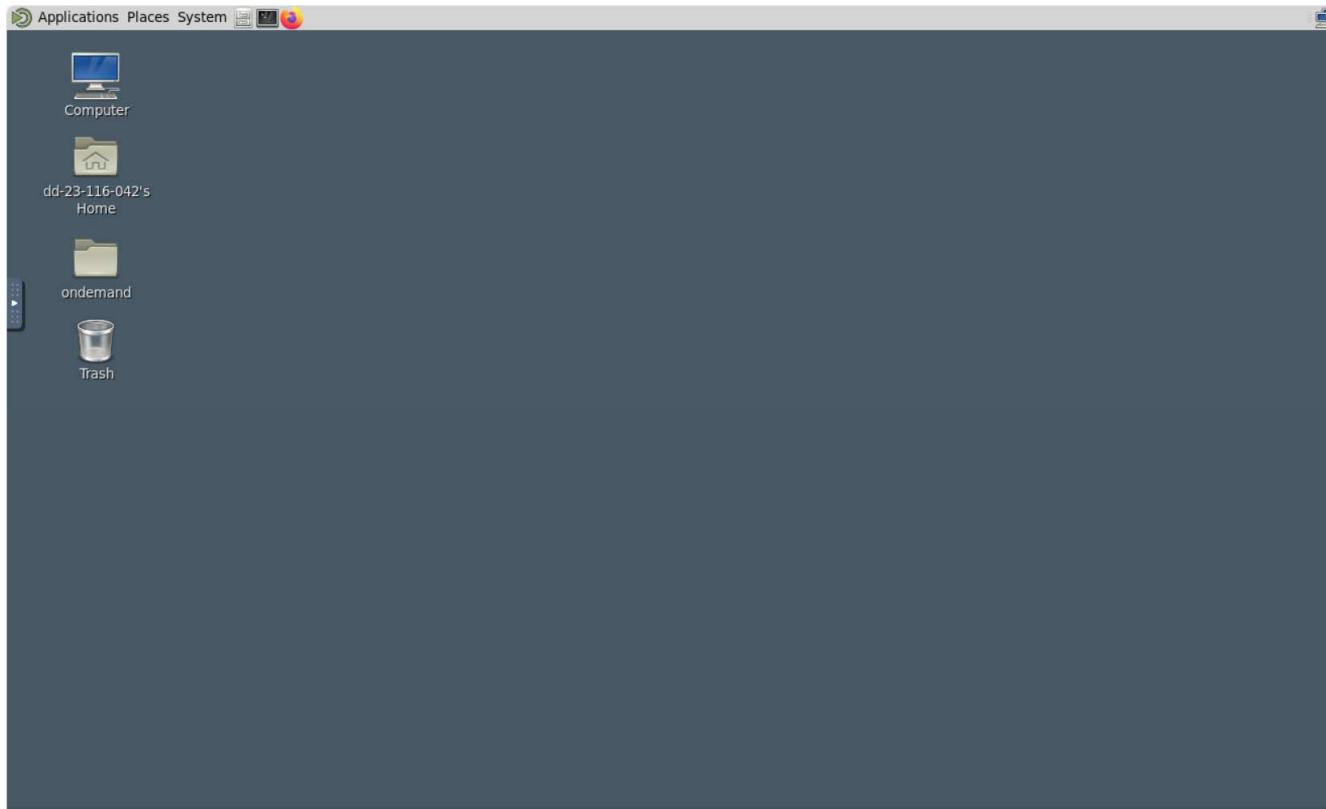
Image Quality 0 (low) to 9 (high)

Launch Karolina Login Mate View Only (Share-able Link)

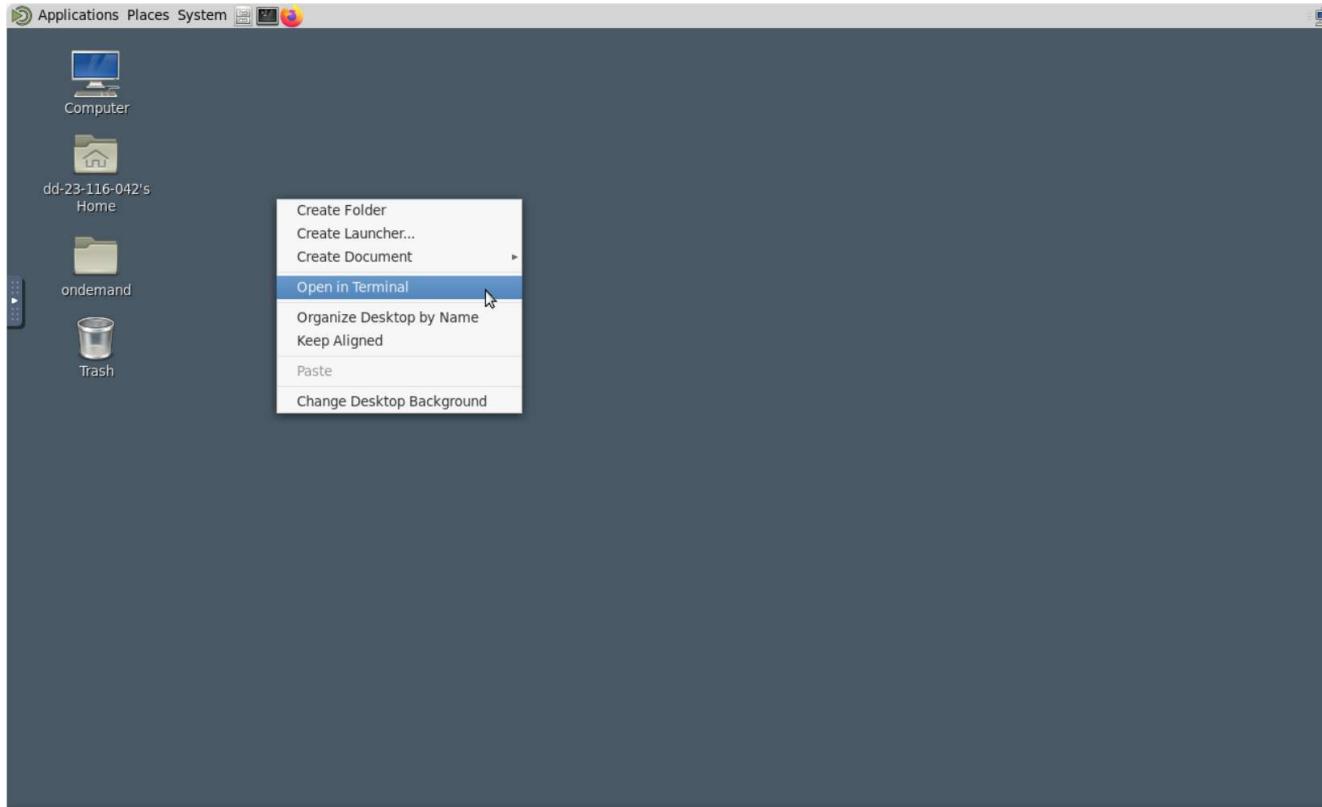
powered by **OPEN** **OnDemand**

OnDemand version: v2.0.27

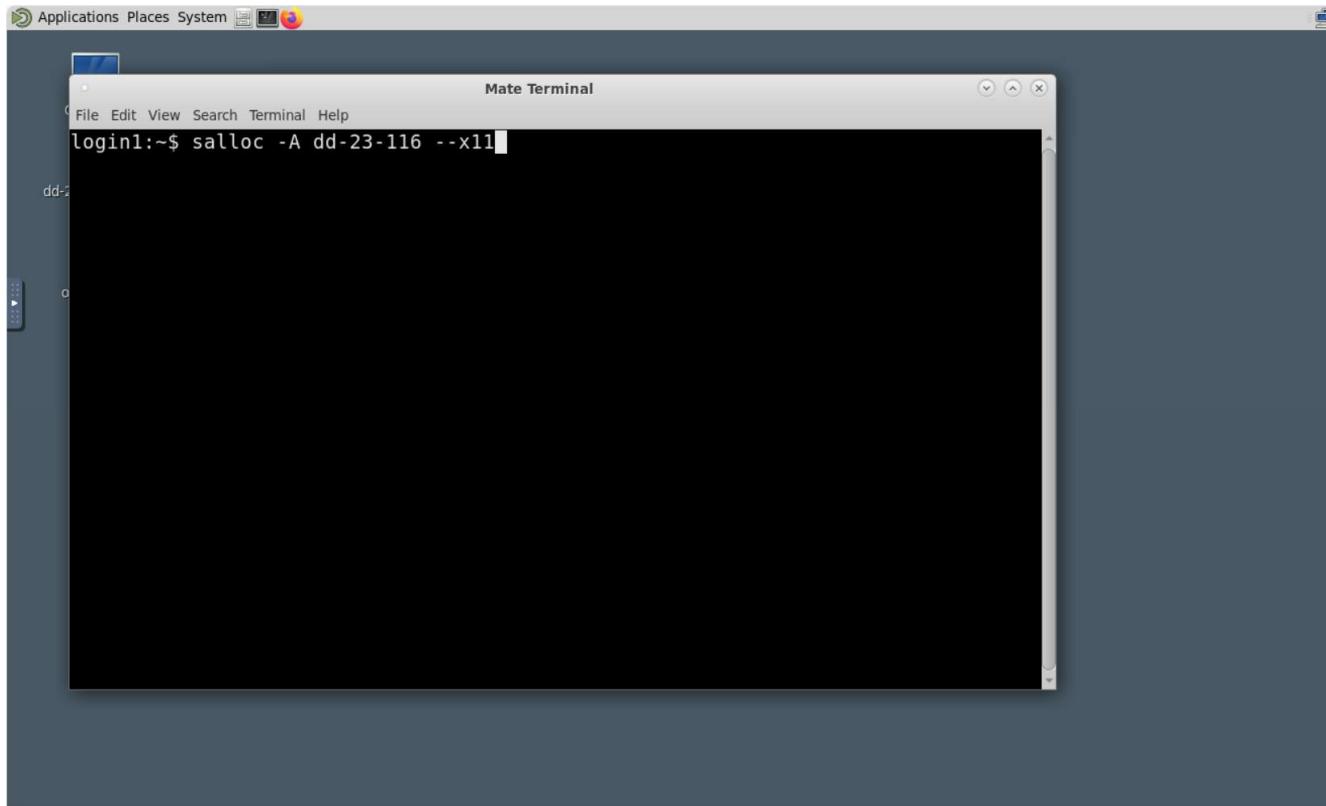
Desktop



Terminal window

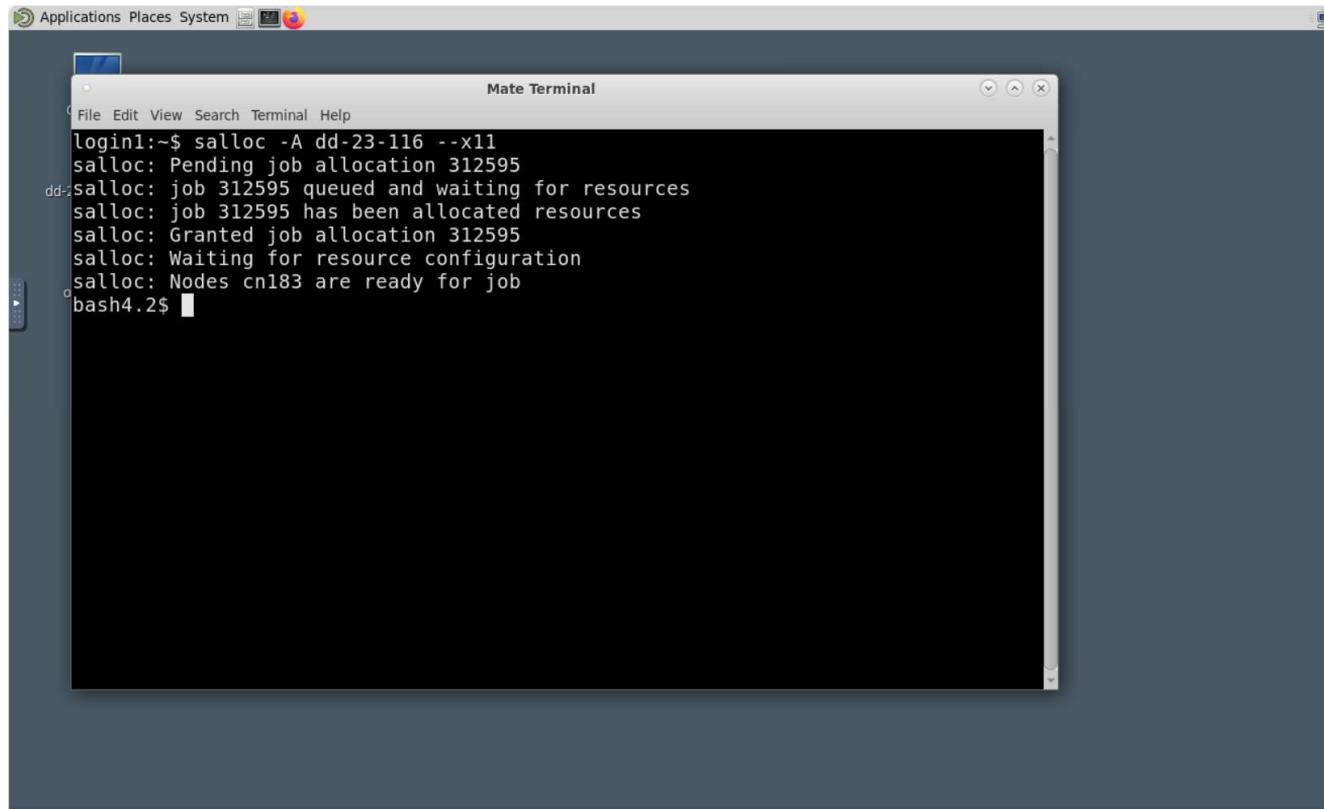


Request Compute Node

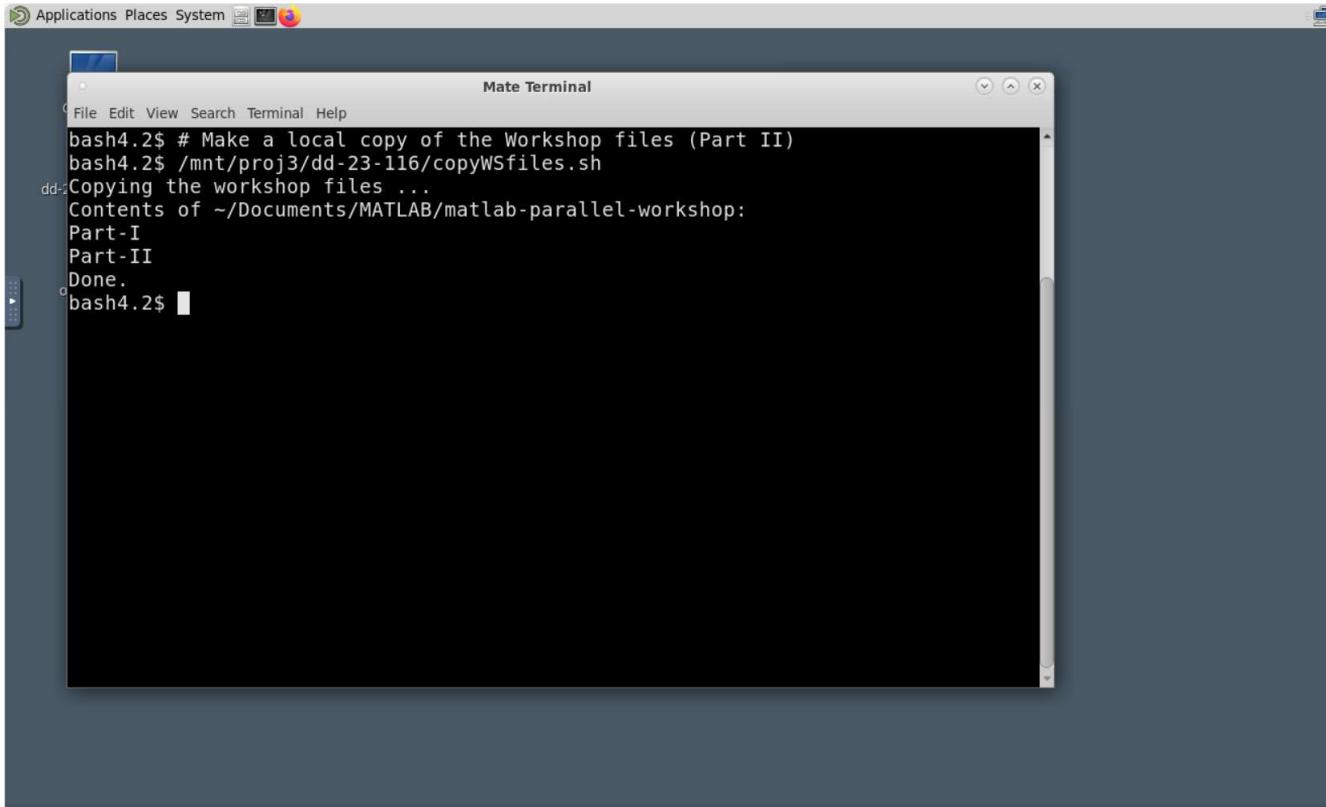


salloc -A dd-23-116 --x11

Connect to Compute Node



Copy Workshop files

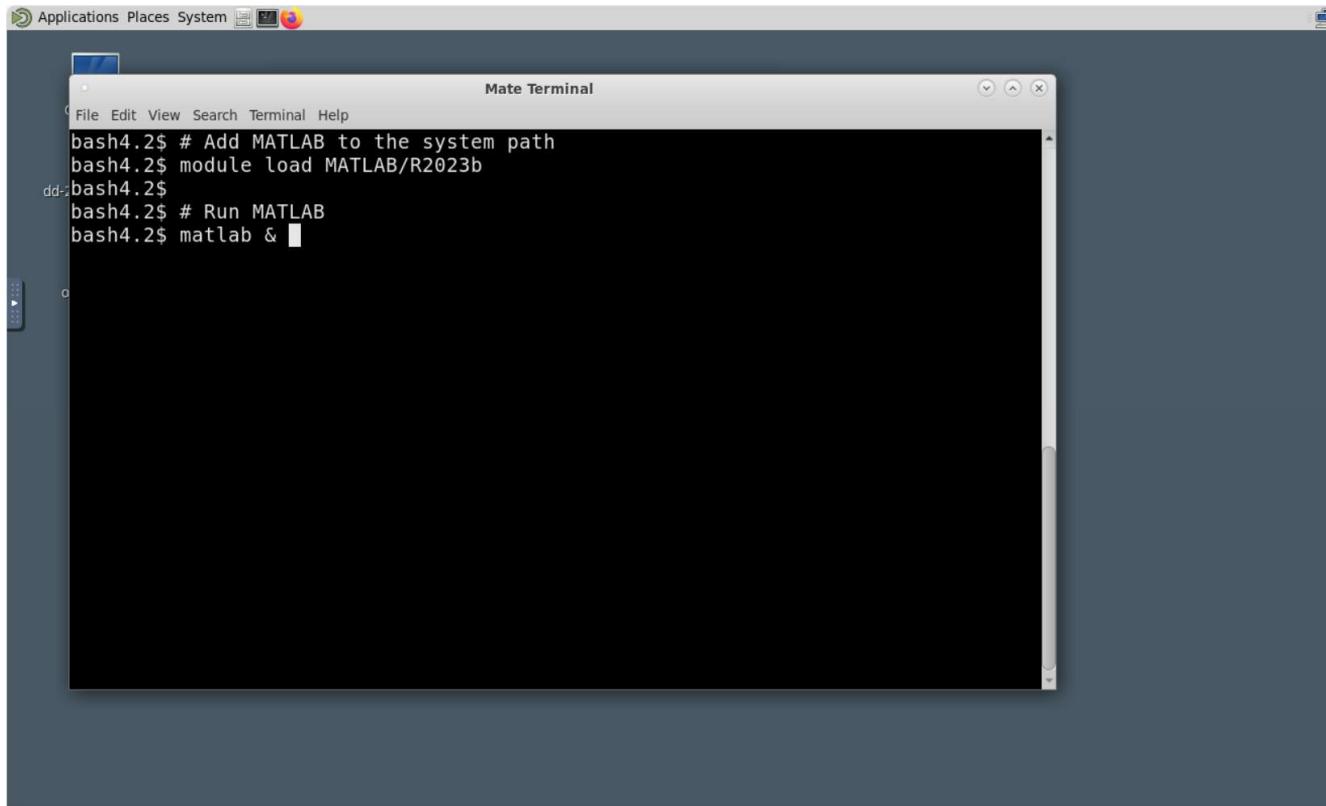


The screenshot shows a terminal window titled "Mate Terminal" running on a Linux desktop environment. The window title bar includes icons for Applications, Places, System, and a few others. The terminal menu bar includes File, Edit, View, Search, Terminal, and Help. The main window displays the following command-line session:

```
bash4.2$ # Make a local copy of the Workshop files (Part II)
bash4.2$ /mnt/proj3/dd-23-116/copyWSfiles.sh
dd: Copying the workshop files ...
Contents of ~/Documents/MATLAB/matlab-parallel-workshop:
Part-I
Part-II
Done.
bash4.2$
```

/mnt/proj3/dd-23-116/copyWSfiles.sh

Start MATLAB



```
module load MATLAB/R2023b
matlab &
```

Ways to run MATLAB

- **Interactively**
 - run serial code
 - with a parallel pool (`parpool`)
 - with batch jobs (`batch`)
- **Noninteractive**
 - write a Slurm job script (`sbatch`)

MATLAB job submitters

- parpool
 - Single session
 - Synchronous execution
 - Seamlessly runs `parfor`,
`parfeval`, and `spmd`
- batch
 - Multiple submissions
 - Non-blocking
 - Calls top-level function or script
 - Requires API to extract results

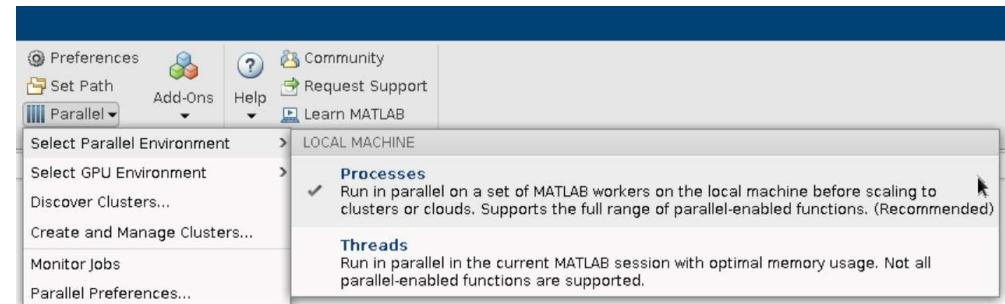
<https://www.mathworks.com/help/parallel-computing/parpool.html>

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

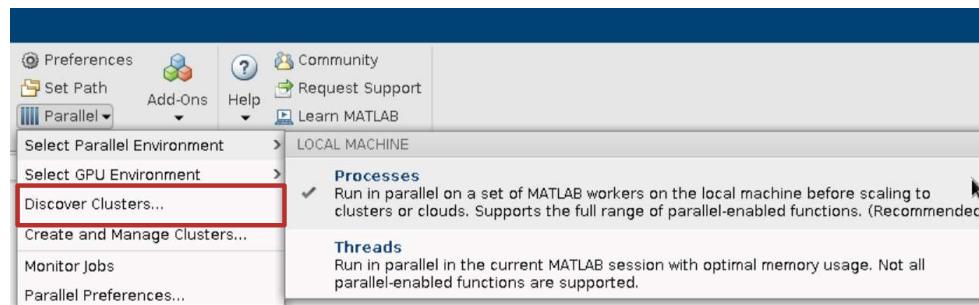
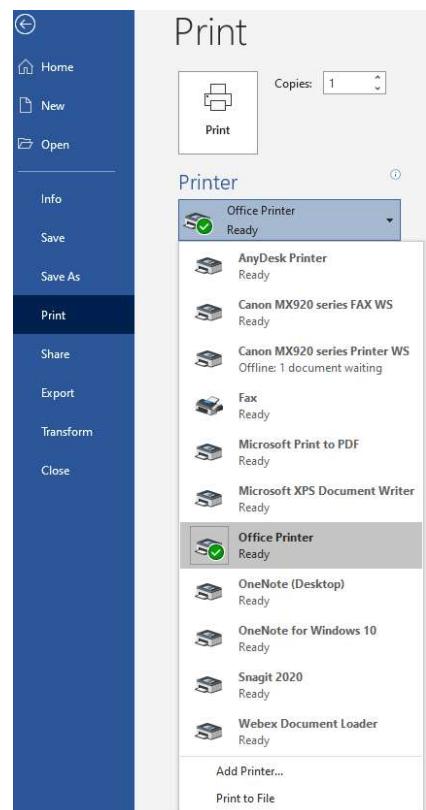
Interactively: with a parallel pool
parpool

Parallel MATLAB – Single Node

```
>> parpool("local",4);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 4 workers.
>>
>> tic, parfor idx = 1:80, pause(3), end, toc
Elapsed time is 60.218578 seconds.
```



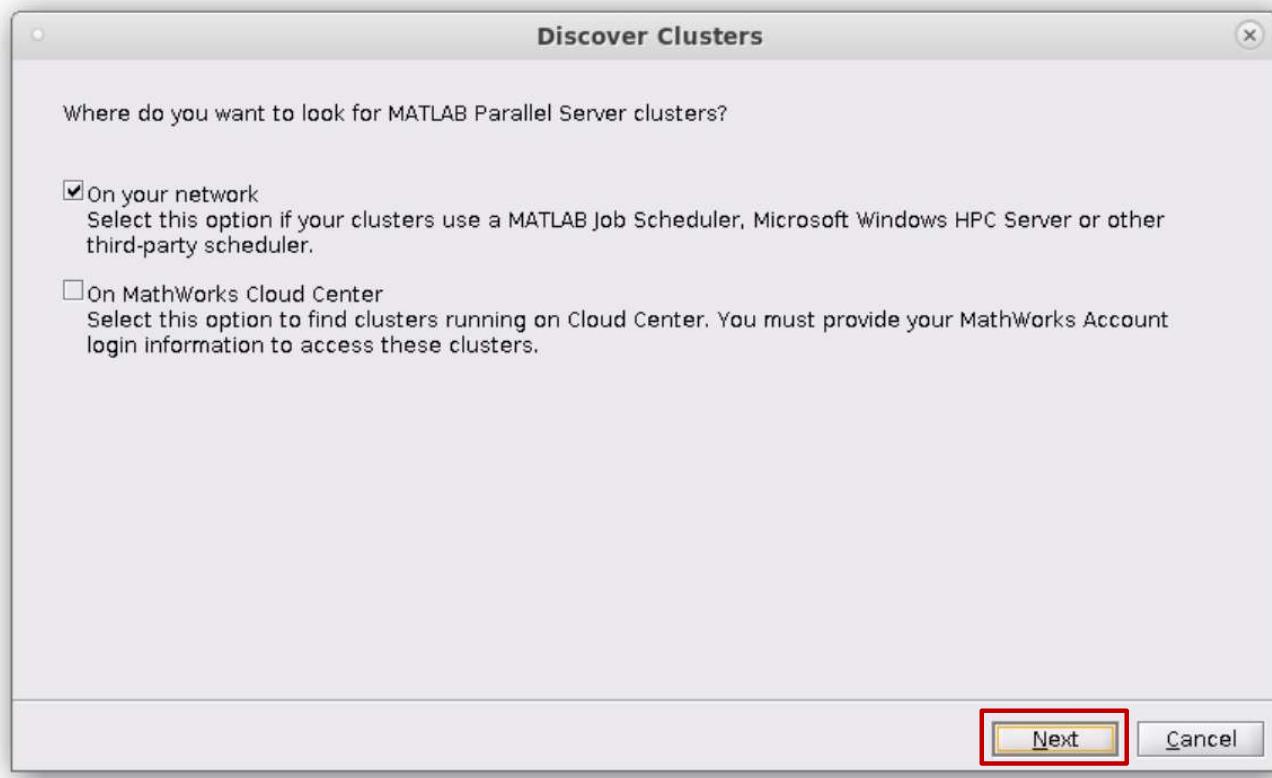
Profiles



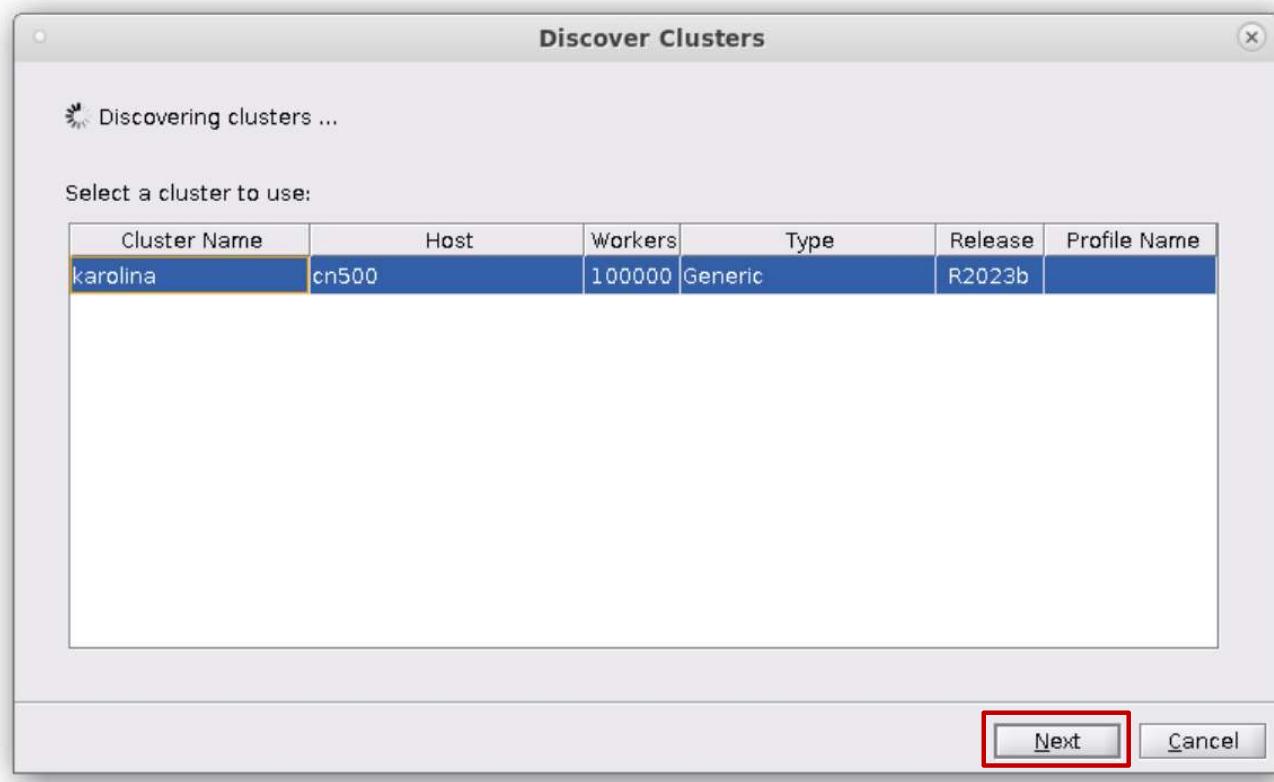
"How does MATLAB know about Karolina?"



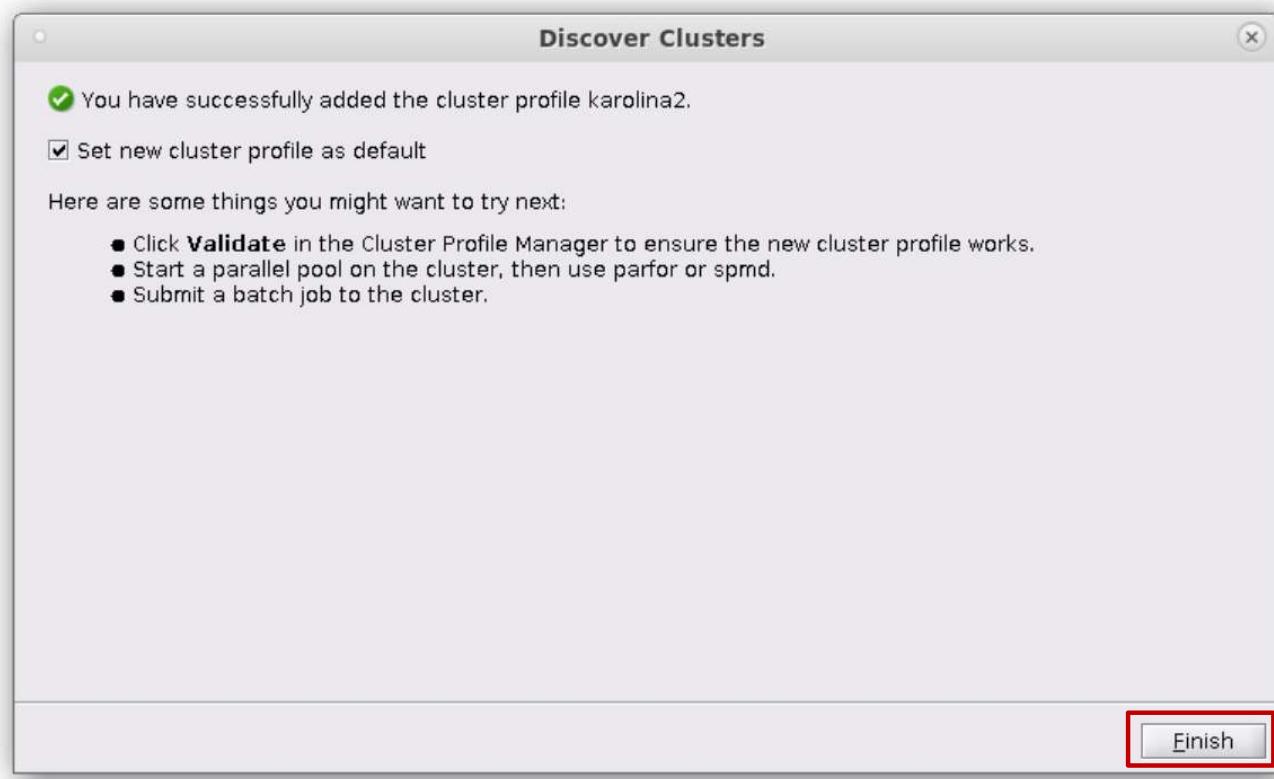
Discover Cluster (1)



Discover Cluster (2)



Discover Cluster (3)



New Karolina profile



Karolina cluster

```
>> c = parcluster("karolina");
```

Must set ProjectName before submitting jobs to KAROLINA. E.g.

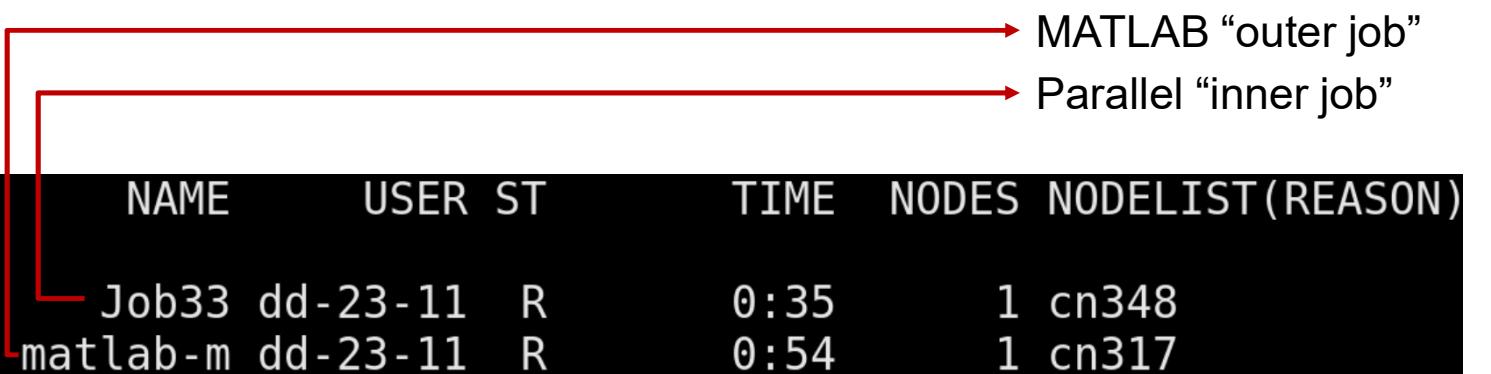
```
>> c = parcluster;  
>> c.AdditionalProperties.ProjectName = 'project-name';  
>> c.saveProfile
```

Minimum
required



Parallel MATLAB – Multi-node

- To run a multi-node MATLAB job, MATLAB will generate and submit a new Slurm job
 - Executed during any “job launcher”
 - parpool, batch, createJob
 - True regardless if we’re running MATLAB desktop or via an Slurm job script
- Need to use the new Karolina profile



JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST (REASON)
172488	qcpu	Job33	dd-23-11	R	0:35	1	cn348
172487	qcpu	matlab-m	dd-23-11	R	0:54	1	cn317

Where is the scaling (1)?

```
>> parpool("local", 4);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 4 workers.
>>
>> tic, parfor idx = 1:80, pause(3), end, toc
Elapsed time is 60.218578 seconds.
>>
>> delete(gcp)
Parallel pool using the 'Processes' profile is shutting down.
```

If there are 16x more workers than a local pool, why did it take the same amount of time?

```
>> c = parcluster("karolina");
>> pool = c.parpool(64);
Starting parallel pool (parpool) using the 'karolina' profile ...
additionalSubmitArgs =
    '--ntasks=64 --cpus-per-task=1 --ntasks-per-core=1
Connected to parallel pool with 64 workers.
>>
>> tic, parfor idx = 1:1280, pause(3), end, toc
Elapsed time is 60.500690 seconds.
```

Where is the scaling (2)?

```
>> parpool("local", 4);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 4 workers.
>>
>> tic, parfor idx = 1:80, pause(3), end, toc
Elapsed time is 60.218578 seconds.
>>
>> delete(gcp)
Parallel pool using the 'Processes' profile is shutting down.
```

How can I run 80 and
100 iteration
(separately), and
both take the same
amount of time?

```
>> c = parcluster("karolina");
>> pool = c.parpool(64);
Starting parallel pool (parpool) using the 'karolina' profile ...
additionalSubmitArgs =
    '--ntasks=64 --cpus-per-task=1 --ntasks-per-core=1
Connected to parallel pool with 64 workers.
>>
>> tic, parfor idx = 1:80, pause(3), end, toc
Elapsed time is 6.201372 seconds.
>>
>> tic, parfor idx = 1:100, pause(3), end, toc
Elapsed time is 6.079273 seconds.
```

How big of a Pool can or should we run? . . .

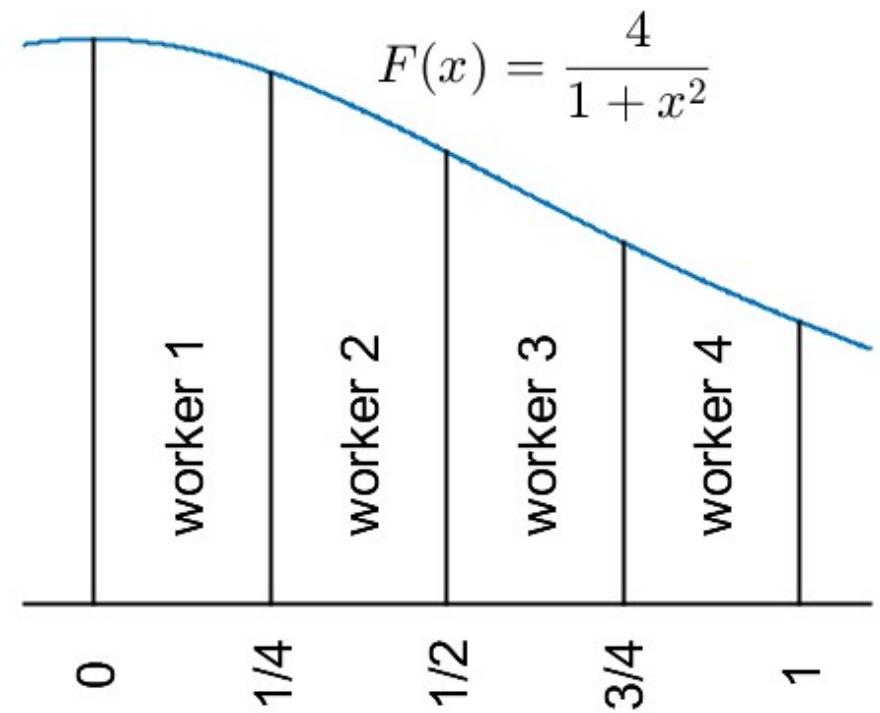
```
>> % Pool of 512 workers across 4 nodes
>> c = parcluster("karolina");
>> tic, pool = c.parpool(512); toc
Starting parallel pool (parpool) using the 'karolina' profile ..
additionalSubmitArgs =
    '--ntasks=512 --cpus-per-task=1 --ntasks-per-core=1
Connected to parallel pool with 512 workers.
Elapsed time is 205.775621 seconds.
>>
>> tic, parfor idx = 1:10240, pause(3), end, toc
Elapsed time is 60.743610 seconds.
>>
>> % Equivalent hours, if run serially
>> 10240 * 3 / 60 / 60
ans =
    8.5333
>>
IdleTimeout has been reached.
Parallel pool using the 'karolina' profile is shutting down.
>>
```

Change directories to workshop

```
>> cd(userpath)  
>> cd("matlab-parallel-workshop/Part-II")
```

Exercise: Calculate π

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



Calculate π

```
function calc_pi

p = gcp;
nsegments = p.NumWorkers;

% Range from 0 to 1, divided by number of workers
boundaries = linspace(0,1,nsegments+1);

parfor idx = 1:nsegments
    a = boundaries(idx)
    b = boundaries(idx+1);
    myIntegral(idx) = integral(@quadpi,a,b);
end

approx = sum(myIntegral);
fprintf('pi          : %.18f\n', pi)
fprintf('Approximation: %.18f\n', approx)
fprintf('Error        : %g\n', abs(pi - approx))

function y = quadpi(x)
y = 4./(1 + x.^2);
```

Calculate π

```
function calc_pi_with_spmd

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

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

    piApprox = spmdPlus(myIntegral);
end

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

function y = quadpi(x)
y = 4./(1 + x.^2);
```

Running on a local node

```
>> parpool("local", 4);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 4 workers.
>>
>> calc_pi
pi          : 3.141592653589793116
Approximation: 3.141592653589793560
Error        : 4.44089e-16
```

Shut Down Parallel Pool

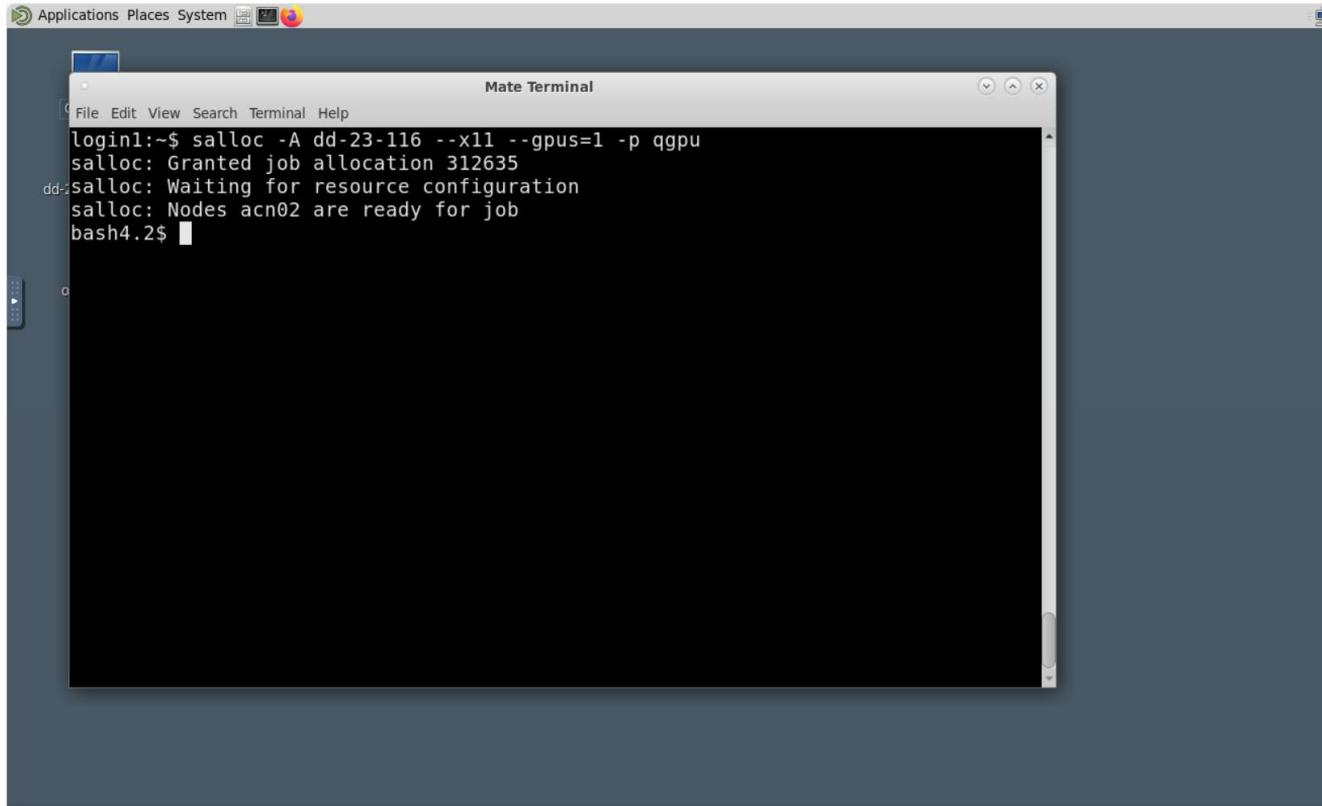
Parallel Preferences

Running across multiple nodes

```
>> parpool("karolina", 64);
Starting parallel pool (parpool) using the 'karolina' profile ..
additionalSubmitArgs =
    '--ntasks=64 --cpus-per-task=1 --ntasks-per-core=1
Connected to parallel pool with 64 workers.
>>
>> calc_pi
pi          : 3.141592653589793116
Approximation: 3.141592653589793116
Error       : 0
```

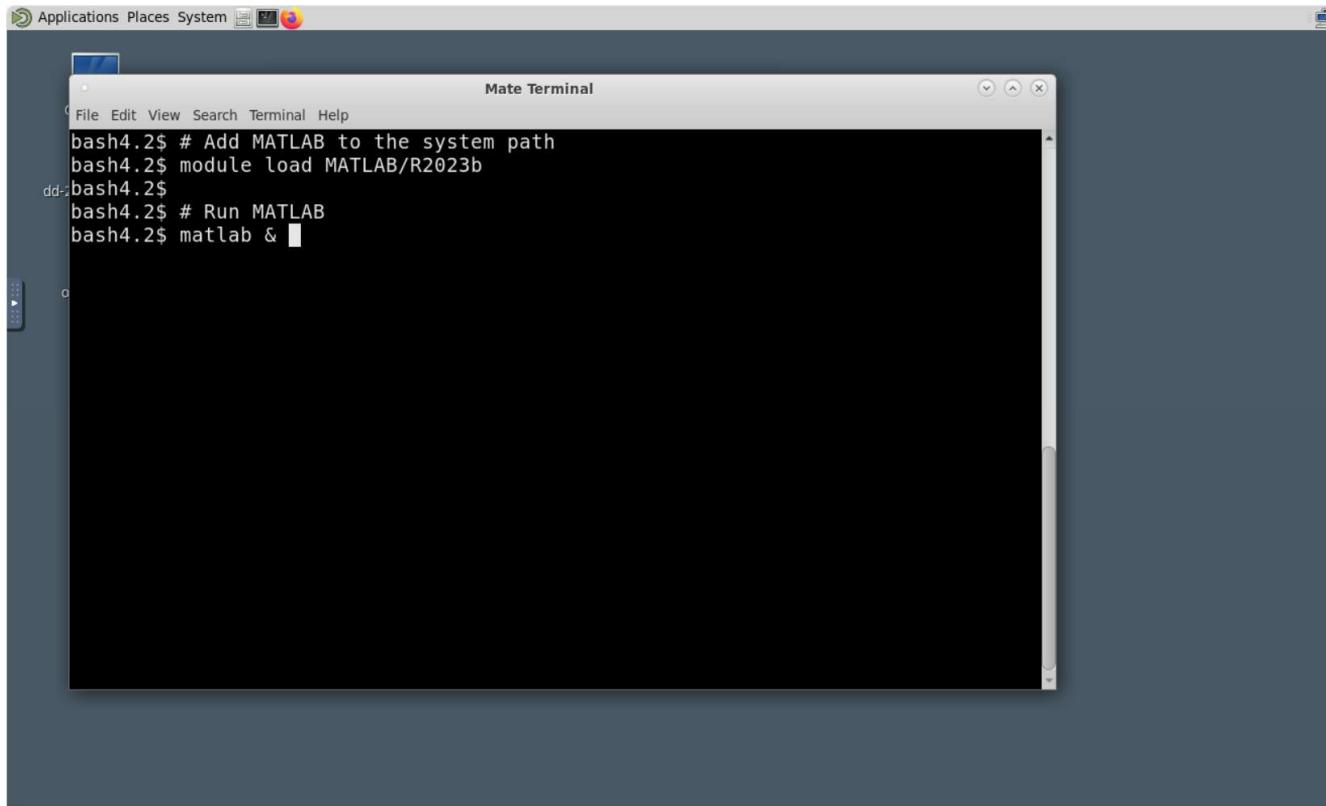
GPUs

Request GPU Compute Node



```
salloc -A dd-23-116 --x11 --gpus=1 -p qgpu
```

Start MATLAB



```
module load MATLAB/R2023b
matlab &
```

Ampere A100

```
>> gpuDevice
ans =
    CUDADevice with properties:

                    Name: 'NVIDIA A100-SXM4-40GB'
                    Index: 1
        ComputeCapability: '8.0'
      SupportsDouble: 1
 GraphicsDriverVersion: '535.104.12'
        DriverModel: 'N/A'
     ToolkitVersion: 11.8000
 MaxThreadsPerBlock: 1024
 MaxShmemPerBlock: 49152 (49.15 KB)
 MaxThreadBlockSize: [1024 1024 64]
     MaxGridSize: [2.1475e+09 65535 65535]
            SIMDWidth: 32
 TotalMemory: 42298834944 (42.30 GB)
 AvailableMemory: 41850699776 (41.85 GB)
```

Example: calc_mandelbrot (1)

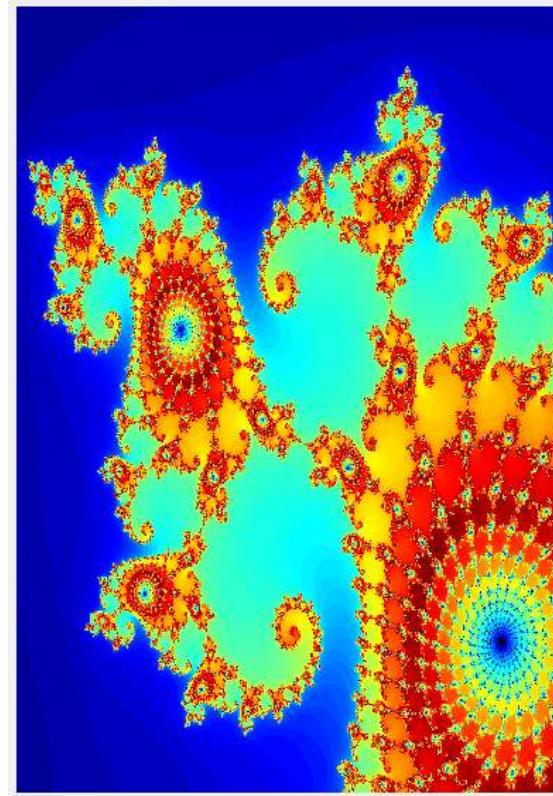
```
function [x,y,count,t] = calc_mandelbrot(type)

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

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

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

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



Example: calc_mandelbrot (2)

```
function mandelbrot_example

maxNumCompThreads(16);

% Run on CPU
[cpu_x,cpu_y,cpu_count,cpu_t] = calc_mandelbrot('CPU');

% Run on GPU
[gpu_x,gpu_y,gpu_count,gpu_t] = calc_mandelbrot('GPU');

figure
subplot(1,2,1)
imagesc(cpu_x,cpu_y,cpu_count)
colormap([jet; flipud(jet); 0 0 0]);
axis off

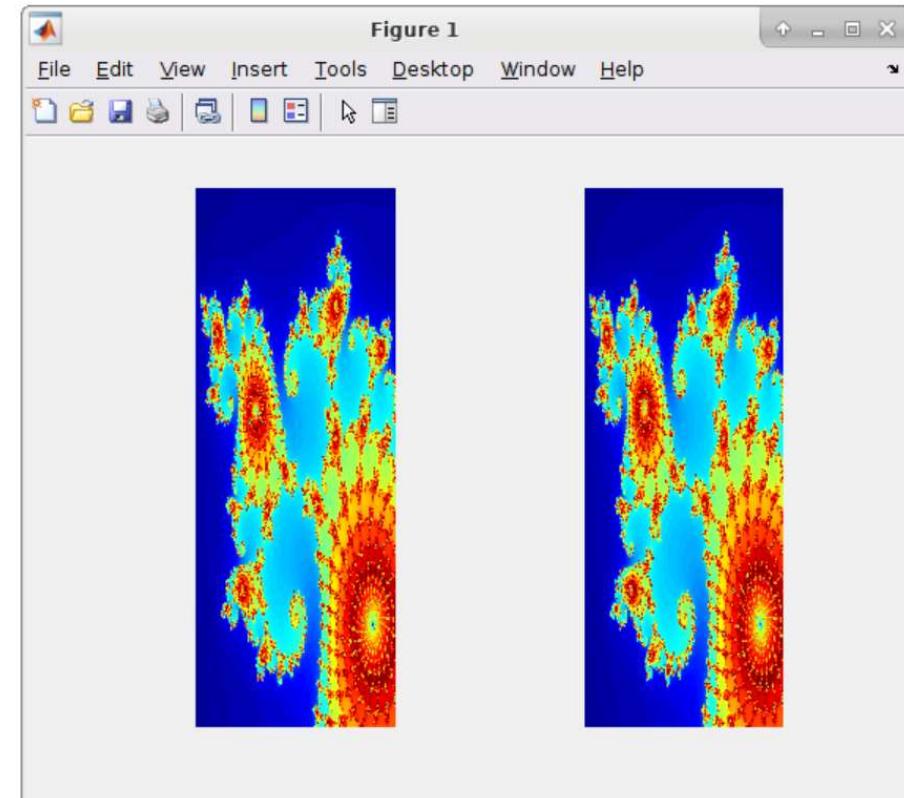
subplot(1,2,2)
imagesc(gpu_x,gpu_y,gpu_count)
colormap([jet; flipud(jet); 0 0 0]);
axis off

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

end
```

```
>> mandelbrot_example
```

Example: calc_mandelbrot (3)



```
>> mandelbrot_example  
CPU time: 35.49  
GPU time: 0.84
```

CPU time: 271.08
GPU time: 0.40

FFT benchmarking

```
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)])
```

oom: Out of memory

```
>> % Clear GPU memory
>> reset(gpuDevice)
>>
>> % GPU memory (GB)
>> d = gpuDevice;
>> round(d.AvailableMemory/1024^3)
ans =
    39
>>
>> % 32 GB matrix
>> [cpu_t, gpu_t] = calc_fft_cpu_gpu(2^16);
Total time on CPU: 32.1678
Error using gpuArray
Maximum variable size allowed on the device is exceeded.
Error in calc_fft_cpu_gpu (line 12)
matrix_gpu = gpuArray(matrix_cpu);
```

2 GB matrix

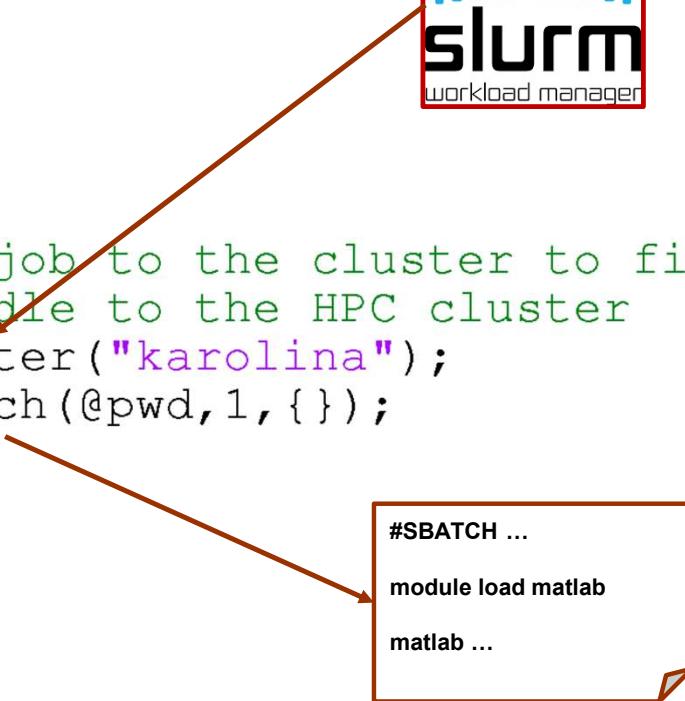
```
>> % Clear GPU memory
>> reset(gpuDevice)
>>
>> % 8 GB matrix
>> [cpu_t, gpu_t] = calc_fft_cpu_gpu(2^15);
Total time on CPU: 2.078
GPU FFT: 0.085378
Total time on GPU: 2.8222
FFT speed improvement: 24.3383 ←
Total speed improvement: 0.7363 ←
```

Interactively: with a batch job batch

Exercise: “Hello, World!”



```
>> % Submit a job to the cluster to find where MATLAB is running  
>> % Get a handle to the HPC cluster  
>> c = parcluster("karolina");  
>> job = c.batch(@pwd,1,{});
```



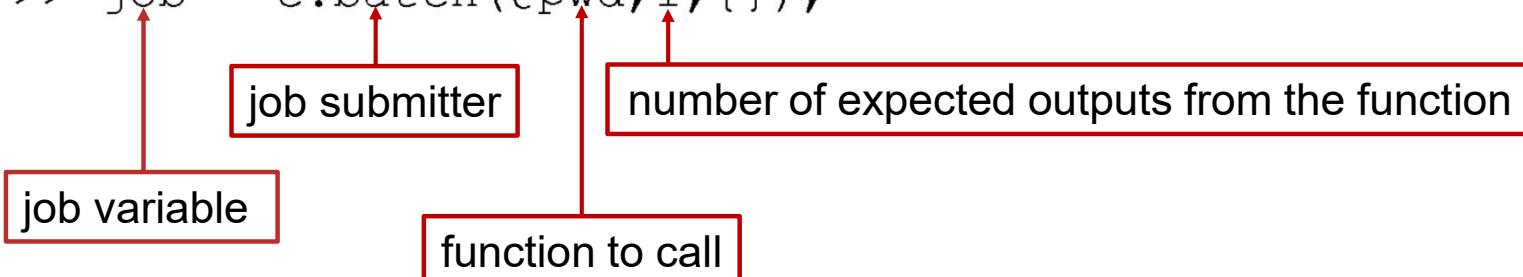
A diagram illustrating the workflow. A red arrow points from the Slurm logo to the MATLAB code. Another red arrow points from the MATLAB code to a sampleSBATCH script, which is enclosed in a red box. The sampleSBATCH script contains the following text:

```
#SBATCH ...  
module load matlab  
matlab ...
```

If no profile is supplied to `parcluster`, use the default profile

Exercise: “Hello, World!”

```
>> % Submit a job to the cluster to find where MATLAB is running  
>> % Get a handle to the HPC cluster  
>> c = parcluster("karolina");  
>> job = c.batch(@pwd,1,{});
```



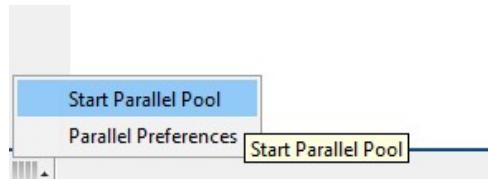
If no profile is supplied to `parcluster`, use the default profile

Fetching results

```
>> % Submit a job to the cluster to find where MATLAB is running
>> % Get a handle to the HPC cluster
>> c = parcluster("karolina");
>> job = c.batch(@pwd,1,{});
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Check the state of the job
>> job.State
ans =
    'finished'
>>
>> % Fetch the results
>> job.fetchOutputs{:}
ans =
    '/home/training/dd-23-116-042/matlab'
```

How should you start a local parallel pool?

- Call `parpool` from the Command Window
- Have MATLAB automatically start a parallel pool if it hasn't already started
- From the lower lefthand corner



Where should you start a parallel pool?

```
function parallel_example  
    %  
    parfor idx = 1:8  
        A(idx) = rand;  
    end
```

“What will happen
the next time you run
this code?”



Error using parpool

Found an interactive session. You cannot have multiple interactive sessions open simultaneously.

Then how do I tell the cluster my job needs a parallel pool?...

```
>> job = c.batch(..., 'Pool', pool_size);
```

Exercise: Submit calc_pi job

```
>> % Submit calc_pi job
>> c = parcluster("karolina");
>>
>> % Request 8 workers
>> job = c.batch(@calc_pi, 0, {}, 'Pool', 8);
additionalSubmitArgs =
    '--ntasks=9' --cpus-per-task=1 --ntasks-per-core=1
```



"If my Pool is size 8, why am I requesting 9 tasks?"

Fetch the results

```
>> % Submit calc_pi job
>> c = parcluster("karolina");
>>
>> % Request 8 workers
>> job = c.batch(@calc_pi, 0, {}, 'Pool', 8);
additionalSubmitArgs =
    '--ntasks=9 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Check the state of the job
>> job.State
ans =
    'finished'
>>
>> % Fetch the results
>> job.fetchOutputs{[:]}
```



What can get returned from running a job?

- Function output
- Diary
- Saved files

Example

```
function [time, 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
    time = toc(t0);

    disp('Finished')

    save RESULTS A
```

Job submission

```
>> job2 = c.batch(@test_fcn,1,{300}, 'Pool',10);  
additionalSubmitArgs =  
'--ntasks=11 --cpus-per-task=1 --ntasks-per-core=1
```

Fetch output

```
function [time, A] = test_fcn(sims)  
c.batch(@test_fcn, 1, {300},
```

```
>> % Fetch the results  
>> job2.fetchOutputs{ : }  
ans =  
    15.7134  
>>
```

“Where's A? ”



Diary

```
function [time, A] = test_fcn(sims)           >> % View the diary
    disp('Start sim')                         >> job2.diary
    A = nan(sims,1);                          --- Start Diary ---
    t0 = tic;                                Start sim
    parfor idx = 1:sims                      ans =
        A(idx) = idx;                         ans = 6
        pause(0.5)                           ans = 5
        idx                                ans = 4
    end                                     time = toc(t0);
    disp('Finished')                         ans = 3
    save RESULTS A                           ans = 1 4
```

Fetch the diary from calc_pi

```
>> job.diary
--- Start Diary ---
pi          : 3.141592653589793116
Approximation: 3.141592653589793116
Error       : 0

--- End Diary ---
```

Save files

“Where does **RESULTS**
get written to?”



```
function [time, 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
    time = toc(t0);

    disp('Finished')

    save RESULTS A
```

“Who needs threads?...”

```
function t = threads_example(N)

matrix = rand(N) ;

t0 = tic;
fft(matrix);
t = toc(t0);

end
```

Single threaded vs multi-threaded

```
>> % Run on a single thread (core)
>> job_01 = c.batch(@threads_example,1,{2^16});
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Time to run on a single thread
>> t_01 = job_01.fetchOutputs{::}
t_01 =
    46.0263
>>
>> % Allocate 5 threads
>> c.NumThreads = 5;
>> job_05 = c.batch(@threads_example,1,{2^16});
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=5 --ntasks-per-core=1
>>
>> % Time to run with multi-threading
>> t_05 = job_05.fetchOutputs{::}
t_05 =
    9.2995
```

Tuning jobs . . .

Failed to submit job to Slurm using command:

```
sh
'/home/training/dd-23-116-042/.matlab/generic_cluster_jobs/karolina/Job36/t
    Reason: sbatch: error: AssocMaxSubmitJobLimit
sbatch: error: Batch job submission failed: Job violates accounting/QOS policy
(job submit limit, user's size and/or time limits)
```

Other settable job properties (1)

```
>> c.AdditionalProperties  
ans =  
AdditionalProperties with properties:  
  
    AdditionalSubmitArgs: ''  
    Constraint: ''  
    EmailAddress: ''  
    EnableDebug: 0  
    GPUCard: ''  
    GPUsPerNode: 0  
    MemPerCPU: ''  
    Partition: ''  
    ProcsPerNode: 0  
    ProjectName: ''  
    QoS: ''  
    Reservation: ''  
    WallTime: ''
```

Other settable job properties (2)

- Constraint
- EmailAddress
- GPUCard
- GPUsPerNode
- MemPerCPU
- Partition
- ProcsPerNode
- **ProjectName**
- QoS
- Reservation
- WallTime

batchsim: Can I only run MATLAB? What about Simulink?

```

function simJob = batchsim_example(~)

c = parcluster;
c.AdditionalProperties.WallTime = '01:00:00';

mdl = 'ex_sldemo_househeat';

% Open and load model
openExample('simulink/OpenTheModelExample')
open_system(mdl)
load_system(mdl)

% Define temperatures
temps = 62:4:84;
tlen = length(temps);

% Initialize Simulation Inputs
in(1:tlen) = Simulink.SimulationInput(mdl);
for tidx = 1:tlen
    in(tidx) = in(tidx).setBlockParameter( ...
        [mdl '/Set Point'], 'Value', num2str(temps(tidx)));
end

% Submit job
simJob = batchsim(c, in, 'ShowProgress', 'on', ...
    'StopOnError', 'on', 'Pool', tlen);

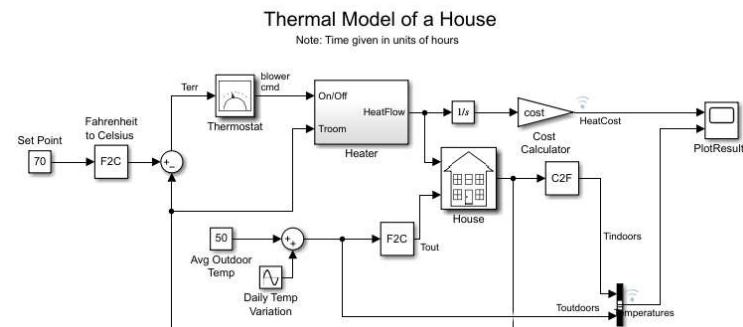
```

```

>> job = batchsim_example;
additionalSubmitArgs =
'--ntasks=7 --cpus-per-task=1 --ntasks-per-core=1
>> job.diary
--- Start Diary ---
[18-Sep-2023 22:07:25] Checking for availability of parallel pool...
[18-Sep-2023 22:07:26] Starting Simulink on parallel workers...
[18-Sep-2023 22:07:50] Configuring simulation cache folder on parallel workers
[18-Sep-2023 22:07:50] Loading model on parallel workers...
[18-Sep-2023 22:08:14] Running simulations...
[18-Sep-2023 22:08:28] Completed 1 of 6 simulation runs
[18-Sep-2023 22:08:28] Completed 2 of 6 simulation runs
[18-Sep-2023 22:08:28] Completed 3 of 6 simulation runs
[18-Sep-2023 22:08:28] Completed 4 of 6 simulation runs
[18-Sep-2023 22:08:28] Completed 5 of 6 simulation runs
[18-Sep-2023 22:08:28] Completed 6 of 6 simulation runs
[18-Sep-2023 22:08:29] Cleaning up parallel workers...

--- End Diary ---

```



Submitting scripts, instead of functions

```
>> x = 4;
>> z = rand(3);
>>
>> % Submit a script (instead of a function)
>> job = c.batch('temp = rand(10); y = x, who');
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
>> clear z
>>
>> who
```

Your variables are:

```
c      job      x

>> % Check the state of the job
>> job.State
ans =
    'finished'
```

Loading variables to local workspace

"If we cleared z, then why
does who display it?
And I didn't
need temp!"



```
>> % Load variables
>> job.load
>> who

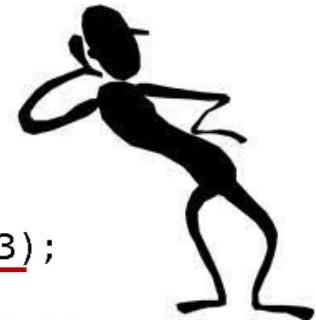
Your variables are:
ans    c    job    temp    x    y

>> job.diary
--- Start Diary ---
y =
    4

Your variables are:
temp    x    y    z

--- End Diary ---
```

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



```
>> z = rand(3);
>> % Submit a sc
>> job = c.batch
```

The cost of transferring unnecessary data

```
>> tic, job = c.batch('temp = rand(10); y = x, who'); toc  
additionalSubmitArgs =  
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1  
Elapsed time is 1.688711 seconds.  
>>  
>> z = rand(2000); % 30 MB  
>> tic, job = c.batch('temp = rand(10); y = x, who'); toc  
additionalSubmitArgs =  
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1  
Elapsed time is 3.285929 seconds.  
>>  
>> z = rand(5000); % 191 MB  
>> tic, job = c.batch('temp = rand(10); y = x, who'); toc  
additionalSubmitArgs =  
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1  
Elapsed time is 13.241239 seconds.
```

willRun: When will my job run?

```
>> % Run a job across multiple nodes
>> job = c.batch(@test_fcn,1,{1023}, 'Pool',1023);
additionalSubmitArgs =
    '--ntasks=1024 --cpus-per-task=1 --ntasks-per-core=1
>> willRun(job)
ans =
    'Resources'
```

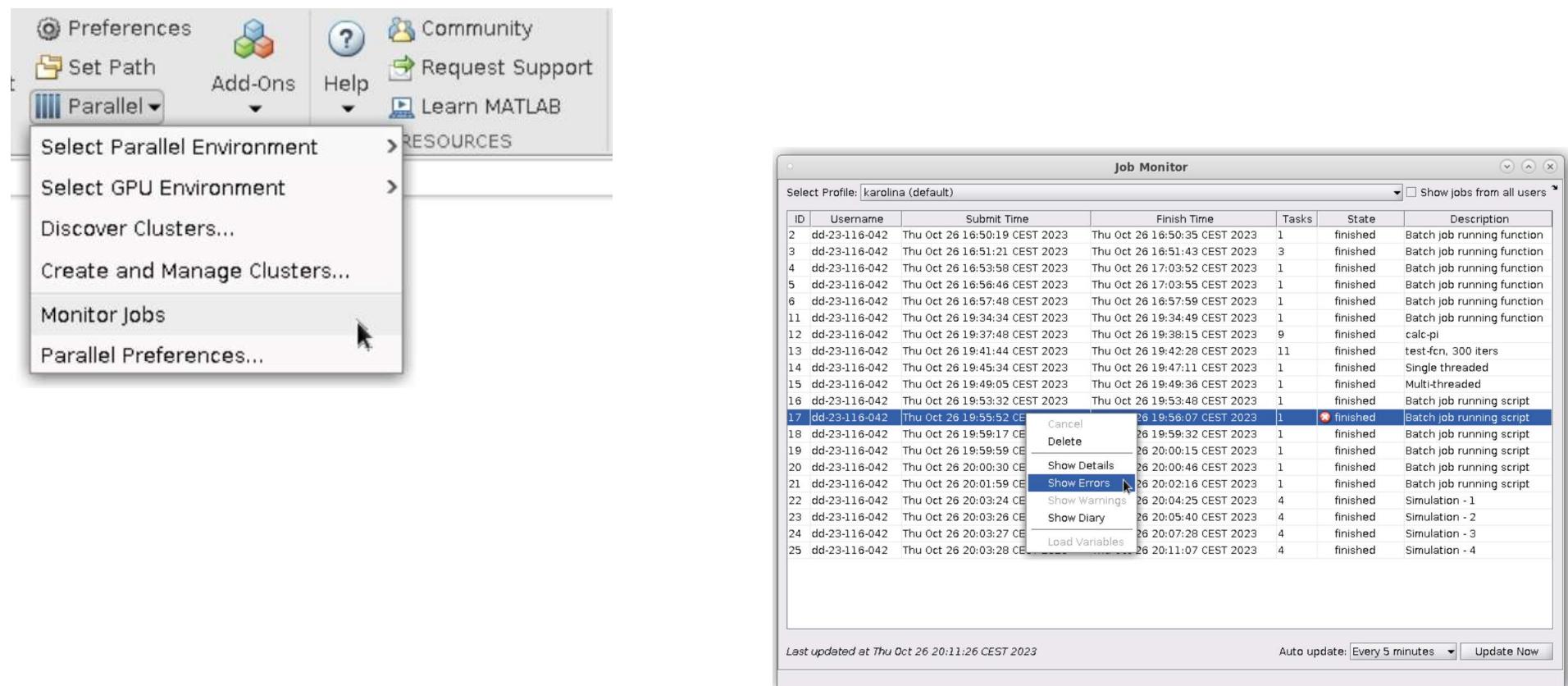


When has my job run and finished?



```
>> % Get email notification when the job has finished
>> c.AdditionalProperties.EmailAddress = 'user-id@vsb.cz';
>>
>> job = c.batch(@test_fcn,1,{300}, 'Pool',10);
additionalSubmitArgs =
'--ntasks=11 --cpus-per-task=1 --ntasks-per-core=1
                                         --mail-type=ALL --mail-user=user-id@vsb.cz'
```

Retrieving past jobs



The screenshot shows the MATLAB interface with the Parallel tab selected in the top navigation bar. A context menu is open over the "Monitor Jobs" option, listing "Select Parallel Environment", "Select GPU Environment", "Discover Clusters...", "Create and Manage Clusters...", "Monitor Jobs", and "Parallel Preferences...".

To the right, the "Job Monitor" window is displayed. It shows a table of completed jobs for the profile "karolina (default)". The table includes columns for ID, Username, Submit Time, Finish Time, Tasks, State, and Description. Job 17 is currently selected, and a context menu for this job is open, showing options: Cancel, Delete, Show Details, Show Errors (which is highlighted), Show Warnings, Show Diary, and Load Variables.

ID	Username	Submit Time	Finish Time	Tasks	State	Description	
2	dd-23-116-042	Thu Oct 26 16:50:19 CEST 2023	Thu Oct 26 16:50:35 CEST 2023	1	finished	Batch job running function	
3	dd-23-116-042	Thu Oct 26 16:51:21 CEST 2023	Thu Oct 26 16:51:43 CEST 2023	3	finished	Batch job running function	
4	dd-23-116-042	Thu Oct 26 16:53:58 CEST 2023	Thu Oct 26 17:03:52 CEST 2023	1	finished	Batch job running function	
5	dd-23-116-042	Thu Oct 26 16:56:46 CEST 2023	Thu Oct 26 17:03:55 CEST 2023	1	finished	Batch job running function	
6	dd-23-116-042	Thu Oct 26 16:57:48 CEST 2023	Thu Oct 26 16:57:59 CEST 2023	1	finished	Batch job running function	
11	dd-23-116-042	Thu Oct 26 19:34:34 CEST 2023	Thu Oct 26 19:34:49 CEST 2023	1	finished	Batch job running function	
12	dd-23-116-042	Thu Oct 26 19:37:48 CEST 2023	Thu Oct 26 19:38:15 CEST 2023	9	finished	calc-pi	
13	dd-23-116-042	Thu Oct 26 19:41:44 CEST 2023	Thu Oct 26 19:42:28 CEST 2023	11	finished	test-fcn, 300 iters	
14	dd-23-116-042	Thu Oct 26 19:45:34 CEST 2023	Thu Oct 26 19:47:11 CEST 2023	1	finished	Single threaded	
15	dd-23-116-042	Thu Oct 26 19:49:05 CEST 2023	Thu Oct 26 19:49:36 CEST 2023	1	finished	Multi-threaded	
16	dd-23-116-042	Thu Oct 26 19:53:32 CEST 2023	Thu Oct 26 19:53:48 CEST 2023	1	finished	Batch job running script	
17	dd-23-116-042	Thu Oct 26 19:55:52 CE	Cancel Delete Show Details Show Errors Show Warnings Show Diary Load Variables	26 19:56:07 CEST 2023 26 19:59:32 CEST 2023 26 20:00:15 CEST 2023 26 20:02:16 CEST 2023 26 20:04:25 CEST 2023 26 20:05:40 CEST 2023 26 20:07:28 CEST 2023 26 20:11:07 CEST 2023	1	finished	Batch job running script
18	dd-23-116-042	Thu Oct 26 19:59:17 CE					
19	dd-23-116-042	Thu Oct 26 19:59:59 CE					
20	dd-23-116-042	Thu Oct 26 20:00:30 CE	Show Details	26 20:00:48 CEST 2023	1	finished	Batch job running script
21	dd-23-116-042	Thu Oct 26 20:01:59 CE	Show Errors	26 20:02:16 CEST 2023	1	finished	Batch job running script
22	dd-23-116-042	Thu Oct 26 20:03:24 CE	Show Warnings	26 20:04:25 CEST 2023	4	finished	Simulation - 1
23	dd-23-116-042	Thu Oct 26 20:03:26 CE	Show Diary	26 20:05:40 CEST 2023	4	finished	Simulation - 2
24	dd-23-116-042	Thu Oct 26 20:03:27 CE	Load Variables	26 20:07:28 CEST 2023	4	finished	Simulation - 3
25	dd-23-116-042	Thu Oct 26 20:03:28 CE		26 20:11:07 CEST 2023	4	finished	Simulation - 4

Last updated at Thu Oct 26 20:11:26 CEST 2023 Auto update: Every 5 minutes Update Now

Keep cluster files minimal: delete jobs

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

```
>> % Delete jobs that are no longer needed  
>> job.delete
```

Debugging and Troubleshooting



Example: Errorred jobs (1)

```
>> % Undefined function
>> job = c.batch(@invalid_fcn, 0, {});
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Reference to undefined variable or function
>> job2 = c.batch('y = a');
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
```

Example: Errorred jobs (2)

```
>> % Undefined function
>> job.State
ans =
    'finished'
>>
>> job.fetchOutputs{::}
Error using parallel.Job/fetchOutputs
An error occurred during execution of Task with ID 1.
Caused by:
    Unrecognized function or variable 'invalid_fcn'.
```

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

Example: Errorred submissions (3)

```
>> % Undefined variable or script
>> job2.State
ans =
    'finished'
>>
>> job2.load
Error using parallel.Job/load
Error encountered while running the batch job. The error was:
Unrecognized function or variable 'a'.
```

If 'a' is a file that is required by 'y = a', add the full path name for 'a' to the batch job's AttachedFiles property. For more information, see [batch](#).

LogFile: Single core job

```
>> job = c.batch(@test_fcn,1,{300}, 'Pool',0);
additionalSubmitArgs =
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Retrieve log file for single worker job
>> c.getDebugLog(job.Tasks)
LOG FILE OUTPUT:
The scheduler has allocated the following node to this job: cn69
Executing: "/apps/all/MATLAB/2023b/bin/worker"

Exiting with code: 0
```

LogFile: Multi-core job

```
>> job = c.batch(@test_fcn,1,{300}, 'Pool',2);
additionalSubmitArgs =
    '--ntasks=3 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Retrieve log file for multi-worker job
>> c.getDebugLog(job)
LOG FILE OUTPUT:
The scheduler has allocated the following nodes to this job:
cn693
"/apps/all/MATLAB/2023b/bin/mw_mpiexec" -bind-to core:1 -l -n 3
[0] Parallel pool is shutting down.
Exiting with code: 0
```

Scheduler ID

```
>> job = c.batch(@test_fcn,1,{300}, 'Pool',2);
additionalSubmitArgs =
    '--ntasks=3 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Job ID vs Scheduler ID
>> job.ID
ans =
    30
>>
>> job.getTaskSchedulerIDs{::}
ans =
    '172437'
```

Designing Robust Code

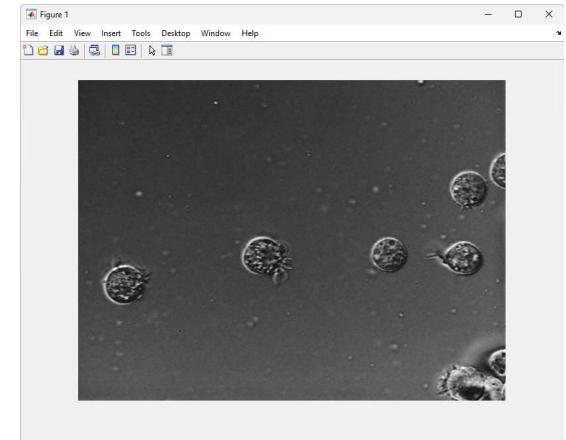


From Coding to Cluster

```
% Notes - From Coding to Cluster
% 1. Using a script, not a function
% 2. Paths are hardcoded
% 3. File separator is hard coded
% 4. Assumes TIF file exists
% 5. TIF files must be on the MATLAB path
% 6. Assumes output folder already exists where ever MATLAB is running
% 7. Results MAT-File will be overwritten next time it's run
```

```
filelist = dir('tif\*.tif');
fileNames = {filelist.name}';

segmentedCellSequence = batchProcessFiles(@detectCells,fileNames);
cd output
save SCS segmentedCellSequence
```



```
function [ofile, segmentedCellSequence] = process_files_v2(roottd,outd)
if nargin==0
    roottd = fullfile(pwd, 'tif');
    outd = fullfile(pwd, 'output');
end

filelist = dir(fullfile(roottd,'*.tif'));
if isempty(filelist)
    error('Failed to find image files: %s',roottd)
end
fileNames = {filelist.name};

addpath(roottd)
segmentedCellSequence = batchProcessFiles(@detectCells,fileNames);

% Ensure output directory exists
if exist(outd,'dir')==false
    [passed,emsg,eid] = mkdir(outd);
    if passed==false
        error(eid,emsg)
    end
end

% Add timestamp for file uniqueness
ts = strrep(strrep(datestr(now), ' ', '_'), ':', '-');
ofile = fullfile(outd,['SCS_ ' ts]);
save(ofile,'segmentedCellSequence')
```

Run it locally

```
>> parpool("local",4);
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 4 workers.
>>
>> ofile = process_files_v2
ofile =
    '/proj/mondays_with_matlab/output/SCS_19-Sep-2023_00-53-41'
```

Run it on the cluster

```
>> c = parcluster("tetralith");
>> job = c.batch(@process_files_v2,1,{ '~/Documents/MATLAB/matlab-parallel-workshop/Part-II', '~/'}, 'Pool',3);
additionalSubmitArgs =
    '--ntasks=4 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Wait for the job to finish
>> job.wait
>>
>> % Fetch the results
>> ofile = job.fetchOutputs{:}
Error using parallel.Job/fetchOutputs
An error occurred during execution of Task with ID 1.
Caused by:
    Error using process_files_v2
    Failed to find image files: ~/Documents/MATLAB/matlab-parallel-workshop/Part-II
```

Run it on the cluster

```
>> c = parcluster("tetralith");
>> job = c.batch(@process_files_v2,1,{ '~/Documents/MATLAB/matlab-parallel-workshop/Part-II/tif', '~/'}, 'Pool',3);
additionalSubmitArgs =
    '--ntasks=4 --cpus-per-task=1 --ntasks-per-core=1
>>
>> % Wait for the job to finish
>> job.wait
>>
>> % Fetch the results
>> ofile = job.fetchOutputs{;}
ofile =
    '~/SCS_19-Sep-2023_01-15-23'
```

From Coding to Cluster (2)

```
% Notes - "From Coding to Cluster"  
% 1. Using a script, not a function  
%      return status or output directory  
% 2. Paths are hardcoded  
%      pass in root directory  
% 3. File separator is hardcoded  
%      use fullfile  
% 4. Assumes TIF files exist  
%      check results when touching file system  
% 5. TIF files must be on the MATLAB path  
%      add tif folder to the MATLAB path  
% 6. Assumes output folder already exists wherever MATLAB is running  
%      supply output folder to write to.  check if folder exists;  
%      if not, create it  
% 7. Results MAT-File will be overwritten next time job is run  
%      add timestamp to filename
```



Running bulk jobs

```
function jobs = submit_jobs

c = parcluster;
c.AdditionalProperties.EmailAddress = 'my-email@work';

sims = [54 162 324 648];

for sidx = 1:length(sims)
    % Run code with different number of iterations
    jobs(sidx) = c.batch(@parallel_example,1,{sims(sidx)} , 'Pool' ,3);
    % Tag the job so that it's easier to find in the Job Monitor
    jobs(sidx).Tag = sprintf("Simulation - %d", sidx);
end

% Wait for the 2nd job to finish
jobs(2).wait

% Get the time for the 2nd job
t = jobs(2).fetchOutputs{::}
```

22	dd-23-116-042	Thu Oct 26 20:03:24 CEST 2023	4	finished	Simulation - 1
23	dd-23-116-042	Thu Oct 26 20:03:26 CEST 2023	4	finished	Simulation - 2
24	dd-23-116-042	Thu Oct 26 20:03:27 CEST 2023	4	finished	Simulation - 3
25	dd-23-116-042	Thu Oct 26 20:03:28 CEST 2023	4	finished	Simulation - 4

Noninteractively: write a Slurm job script sbatch

Submitting “locally” vs “multi-node”

```

function calc_pi_single_node
    p = gcp("nocreate");
    if isempty(p)
        % Query for available cores
        sz = str2num(getenv('SLURM_TASKS_PER_NODE'));
        if isempty(sz), sz = maxNumCompThreads; end
        p = parpool("local",sz);
    end

    nsegments = p.NumWorkers;

    % Range from 0 to 1, divided by number of workers
    boundaries = linspace(0,1,nsegments+1);

    parfor idx = 1:nsegments
        a = boundaries(idx)
        b = boundaries(idx+1);
        myIntegral(idx) = integral(@quadpi,a,b);
    end

    approx = sum(myIntegral);
    fprintf('pi           : %.18f\n', pi)
    fprintf('Approximation: %.18f\n', approx)
    fprintf('Error         : %g\n',      abs(pi - approx))

end

function y = quadpi(x)
y = 4./(1 + x.^2);

```

```

function calc_pi_multi_node
    p = gcp("nocreate");
    if isempty(p)
        % Query for available cores
        sz = str2num(getenv('SLURM_TASKS_PER_NODE'));
        if isempty(sz), sz = maxNumCompThreads; end
        p = parpool("karolina",40);
    end

    nsegments = p.NumWorkers;

    % Range from 0 to 1, divided by number of workers
    boundaries = linspace(0,1,nsegments+1);

    parfor idx = 1:nsegments
        a = boundaries(idx)
        b = boundaries(idx+1);
        myIntegral(idx) = integral(@quadpi,a,b);
    end

    approx = sum(myIntegral);
    fprintf('pi           : %.18f\n', pi)
    fprintf('Approximation: %.18f\n', approx)
    fprintf('Error         : %g\n',      abs(pi - approx))

end

function y = quadpi(x)
y = 4./(1 + x.^2);

```

Submit single-node job

```
#!/bin/sh

#SBATCH --tasks-per-node=8          # 8 local workers

# Add MATLAB to system path
module load MATLAB

# Run code
matlab -batch calc_pi_single_node
```

Remember: Add MATLAB to the system path

Single-node job (1)

Set to your
own Slurm
Account

```
bash4.2$ sbatch -A dd-23-116 matlab-single-node.slurm
Submitted batch job 172456
bash4.2$
bash4.2$ squeue -j 172456
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      172456      qcpu matlab-s dd-23-11   R      0:06      1 cn028
bash4.2$
```

sbatch -A dd-23-116 matlab-single-node.slurm

Single-node job (2)

```
bash4.2$ cat slurm-172456.out
Starting parallel pool (parpool) using the 'local' profile ...
Connected to parallel pool with 8 workers.
pi      : 3.141592653589793116
Approximation: 3.141592653589793116
Error    : 0
Parallel pool using the 'Processes' profile is shutting down.
bash4.2$
```

#!/bin/sh

#SBATCH -n 8

```
p = gcp("nocreate");
if isempty(p)
    % Query for available cores
    sz = str2num(getenv('SLURM_TASKS_PER_NODE'));
    if isempty(sz), sz = maxNumCompThreads; end
    p = parpool("local",sz);
end
```

Submit multi-node job

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

```
#!/bin/sh
```

```
#SBATCH -n 1
```

```
# 1 instance of MATLAB
```

```
# Add MATLAB to system path
module load MATLAB
```

```
# Run code
matlab -batch calc_pi_multi_node
```

Multi-node job (1)

Set to your
own Slurm
Account

```
bash4.2$ sbatch -A dd-23-116 matlab-multi-node.slurm
Submitted batch job 172487
bash4.2$
bash4.2$ squeue -j 172487
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      172487      qcpu matlab-m dd-23-11   R      0:21      1 cn317
bash4.2$
bash4.2$ squeue -u $USER -t R
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      172488      qcpu    Job33 dd-23-11   R      0:35      1 cn348
      172487      qcpu matlab-m dd-23-11   R      0:54      1 cn317
```

```
sbatch -A dd-23-116 --reservation=\
dd-23-116_2023-11-08T12:00:00_2023-11-08T17:00:00_100_qcpu\
matlab-multi-node.slurm
```

Multi-node pool of workers (2)

```
bash4.2$ cat slurm-172487.out
Starting parallel pool (parpool) using the 'karolina' profile ...
additionalSubmitArgs =
    '--ntasks=40 --cpus-per-task=1 --ntasks-per-core=1 -A dd-23-116'
Connected to parallel pool with 40 workers.
pi          : 3.141592653589793116
Approximation: 3.141592653589793560
Error       : 4.44089e-16
Parallel pool using the 'karolina' profile is shutting down.
bash4.2$
```

Summary

- Open OnDemand to Karolina
- Prototype running on interactive compute node
- Create Karolina profile
- Toggle between local (single node) and Karolina (multi-node) profiles
- Tune your job with AdditionalProperties
- Submit large scale from interactive node
- Tips, tricks and best practices for job submission and troubleshooting
- Contact Support to get started
 - support@it4i.cz
 - Who would be interested in 1-1 Coaching?



