Handout

Local Machine

Juptyer Lab

Start Jupyter with the following command, ideally in the `$HOME` directory.

```
jupyter lab
```

- Evaluate a cell: `Ctrl+Enter`
- Evaluate a cell and move to next: `Shift+Enter`
- Create a new cell below: `Esc B`
- Delete a cell: `Esc X`

Visual Studio Code

- Open a regular Terminal: `Ctrl+~`
- Open integrated Julia REPL: `Alt-J Alt-O`
- Kill integrated Julia REPL: `Alt-J Alt-K`
- Restart integrated Julia REPL: `Alt-J Alt-R`
- Execute a line/block of code: `Shift+Enter` and `Ctrl+Enter` (similar to Jupyter)

Julia

- `]` to get into package manager (Pkg) mode
- `?` to get into help mode
- `;` to get into shell mode

Using MPI

It's recommended to run the MPI parts on the cluster. But if you want to you can also use MPI on the local machine. In any case, you should use `~/.julia/bin/mpiexecjl` instead of just `mpirun` or `mpiexec`. For example, to run a MPI program with 4 ranks use

```
mpiexecjl --project -n 4 julia myprogram.jl
```

(or use the full path `~/.julia/bin/mpiexecjl` if necessary)

Hawk Cluster

Note: There is no proper internet connection on Hawk.

Logging in

Note: You should/can not use your private laptop to acces Hawk!

```
ssh hlrskXY@hawk.hww.hlrs.de
```

Julia on Hawk

To make Julia available on Hawk simply type

```
ml julia
```

We've already instantiated the course environment for you such that all Julia packages are available if you run `julia --project` inside of the course folder (`~/JuliaHLRS23`).

Interactive compute-node sessions

To get an interactive session on a Hawk compute node run e.g.

```
qsub -I -l select=1:node_type=rome -l walltime=01:00:00
```

or the script `get-cpu-node-interactive.sh` inside your HOME directory. Here, `-I` indicates interactive mode and the walltime is set to one hour. If you plan to use **MPI**, use the following to get an interactive session or run `get-cpu-node-interactive-MPI.sh` in your HOME directory.

```
qsub -I -l select=1:node_type=rome:mpiprocs=128 -l walltime=01:00:00
```

For Thursday and Friday (Days 3 and 4) we have reserved a few Hawk nodes for the course. To use them add `-q R_julia` to the qsub commands above.

Job submission

If you want to submit a non-interactive job, you first need to create a job file (see example below or `hawk_job.qbs` in your HOME directory).

```
#!/bin/bash
#PBS -N myjob # Change to whatever you like
#PBS -l select=1:node_type=rome
##PBS -q R_julia # uncomment to use the course reservation
#PBS -l walltime=00:30:00 # 30 minutes - change to whatever necessary.
#PBS -j oe
#PBS -o hawk_job.output

# change to the directory that the job was submitted from
cd "$PBS_O_WORKDIR"

# load necessary modules
ml r
ml julia

# run program
julia --project yourfile.jl # Change filename
```

To submit this job to the scheduler use `qsub`, e.g. `qsub hawk_job.qbs`. With `qstat` (or `qstat -rnw`) you can get a list of your scheduled/running jobs.

VSCode remote on Hawk

Connecting

- `CTRL + SHIFT + P` (opens the popup menu) → `Remote-SSH: Connect to Host...`
- Input `hlrskXY@hawk.hww.hlrs.de` for the hostname.
- You need to enter your password.

Julia Extension

Installing the extension

- Open the extension tab in the sider bar on the left (`CTRL + SHIFT + X`), click on the three dots at the top and select `Install from VSIX...`.

- Enter the following path and press Enter:

```
/lustre/hpe/ws10/ws10.1/ws/vtraincb-juliahlrs/vscode/language-julia-1.47.2.vsix
```

Julia wrapper script To use the Julia extension on Hawk you must point the extension to a Julia wrapper script that first loads the Julia module (i.e. `ml julia`) and then runs Julia. The path to the script is:

```
/lustre/hpe/ws10/ws10.1/ws/vtraincb-juliahlrs/vscode/julia_vscode_wrapper.sh
```

To set the relevant setting, press `CTRL + ,` (comma), select the tab (at the top) that says "Hawk" and then search for "julia executable". Finally, copy paste the path above into the text field of the setting.

Note: You should only have to do this **once**, as it should remember the setting for the rest of the course.