Getting started on subMIT: How to Interact with subMIT

subMIT Annual Workshop 2026

https://indico.mit.edu/event/1880/

Matt Heine 1/23/2026

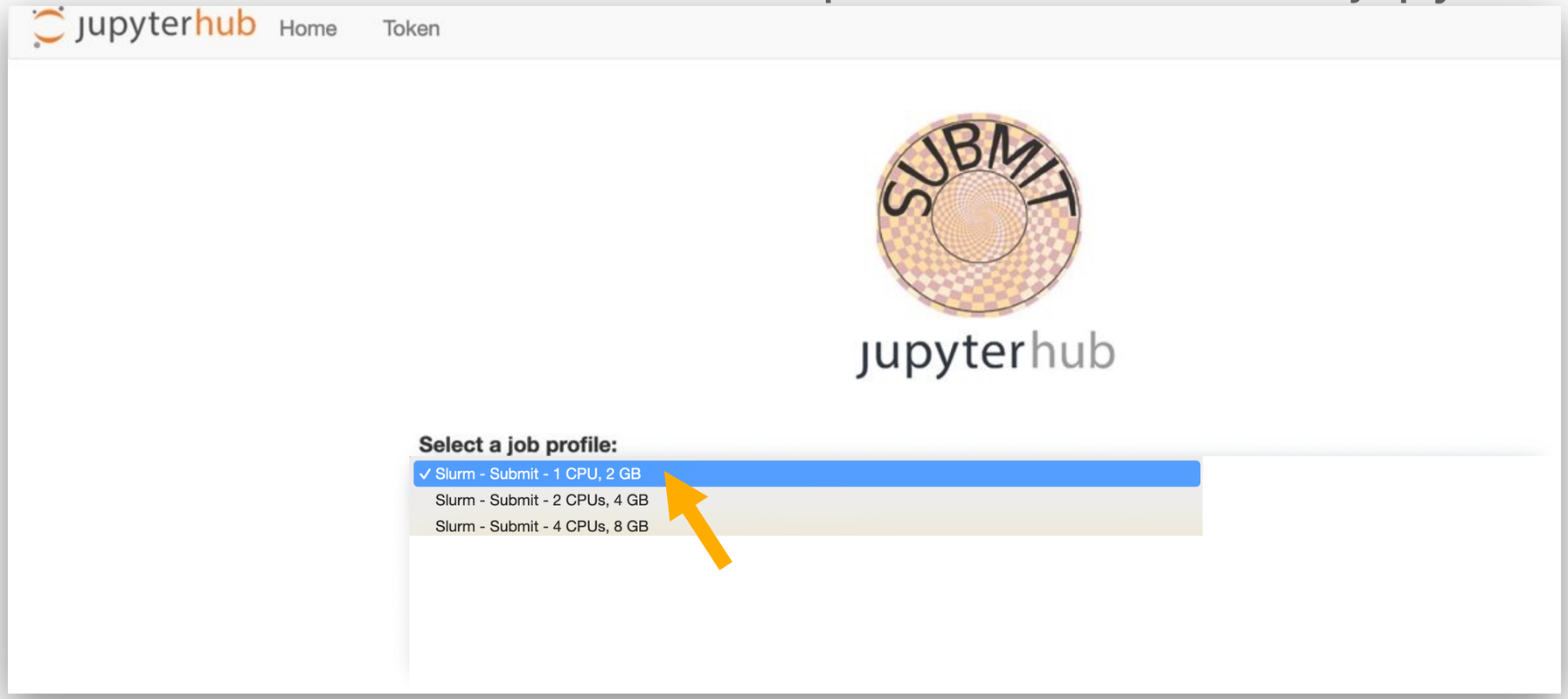
Outline

- JupyterHub
- Visual Studio Code (VSCode)
- terminal / ssh
- Login node session vs batch/interactive jobs
- Initial Account Creation

JupyterHUB

- Access subMIT from a web browser
- Create/Run Juptyer Notebooks
- Graphical Interface + built-in terminal (in web browser)
- Easy interactive access to compute node (dedicated resources)

https://submit.mit.edu/jupyter/



JupyterHUB

- File Browser

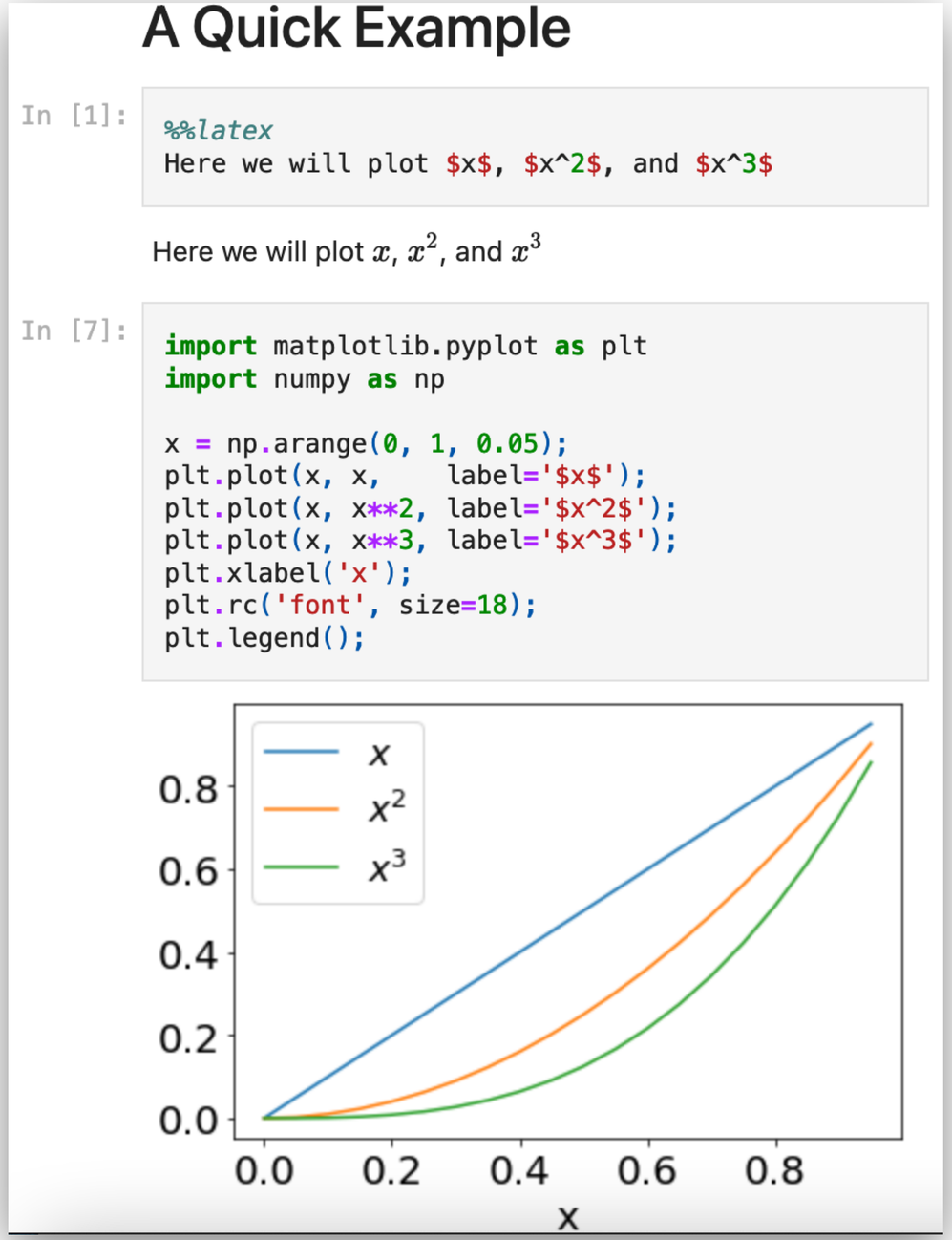
- Jupyter Notebooks: Self-Contained
  - Code
  - Results / Visualization
  - Documentation (Markdown, LaTEX)
  - Easily shared

- Kernels = sets of software / packages used to run code in your notebook
  - Use your conda environments as kernels (automatic setup)
  - Use singularity images (containers) as kernels
  - Change kernels w/ a click

- Many languages (even w/in same notebook)

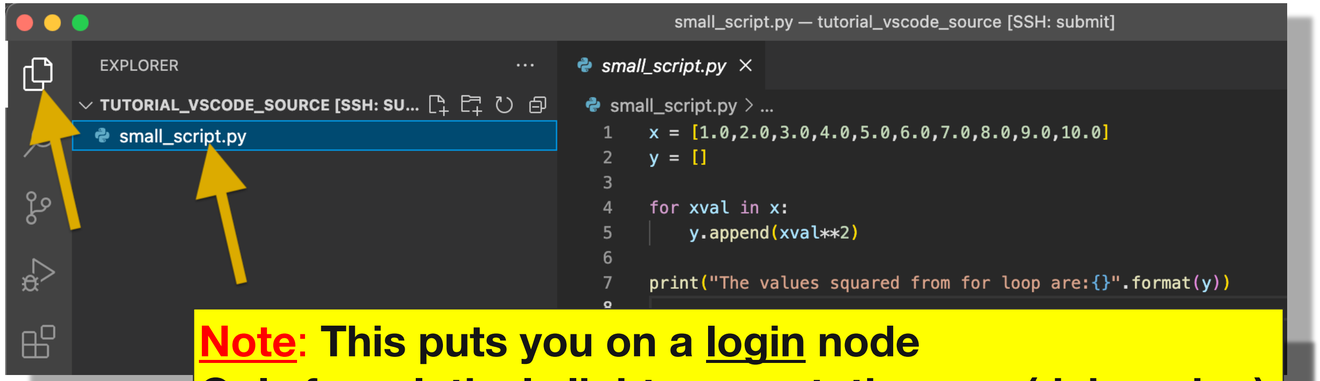
- Many extensions
  - Debugging
  - Source Control (git / GitHub)

subMIT User's Guide:
https://submit.mit.edu/submit-users-guide/access.html#jupyterhub
https://submit.mit.edu/submit-users-guide/program.html#id1

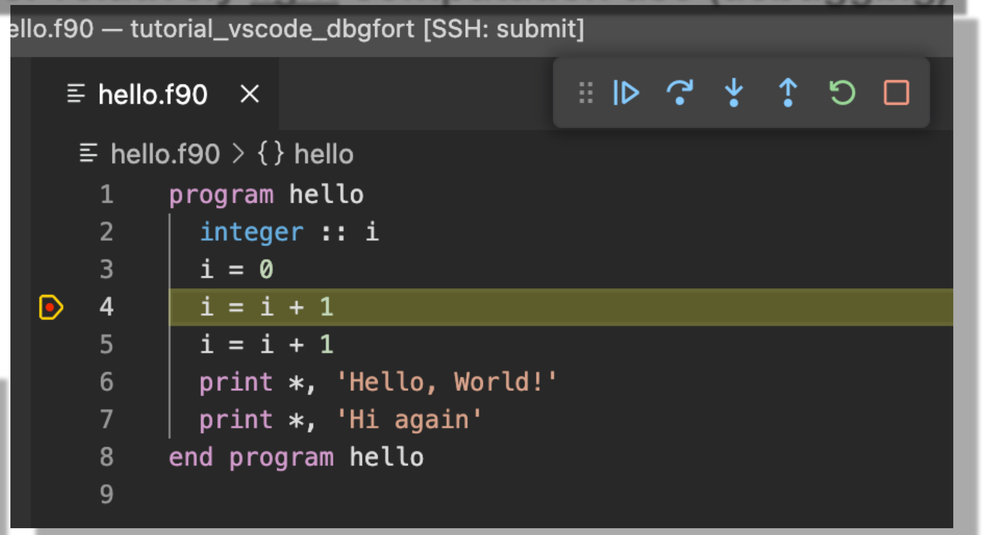


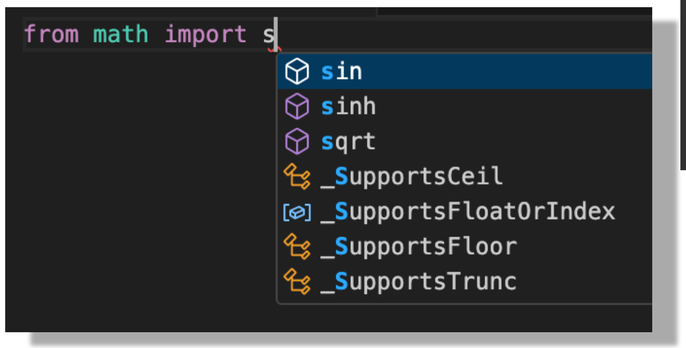
Visual Studio Code: Remote Development

- code runs on subMIT, GUI runs on your laptop
- File Browser
  - GUI to navigate/view your subMIT files/directories
- Many languages / extensions
  - Python, C/C++, Java, Julia, Fortran, …
  - LaTeX, HTML/CSS, Markdown, rst, …
- Code navigation
- Debugging (code runs on subMIT cluster)
  - breakpoints, variable inspection/watch, stack navigation
- Source Control (Integrated / GUI)
- Automatic Code Completion
  - Intellisense
  - Snippets, AI-assisted development
- subMIT User's Guide:
  - https://submit.mit.edu/submit-users-guide/access.html#vscode
  - Videos: https://submit.mit.edu/submit-users-guide/#videos
  - Tutorials: https://submit.mit.edu/submit-users-guide/#tutorials-and-examples



Note: This puts you on a login node
Only for relatively light computation use (debugging)





Change conda environments (on subMIT) for python code w/ a click

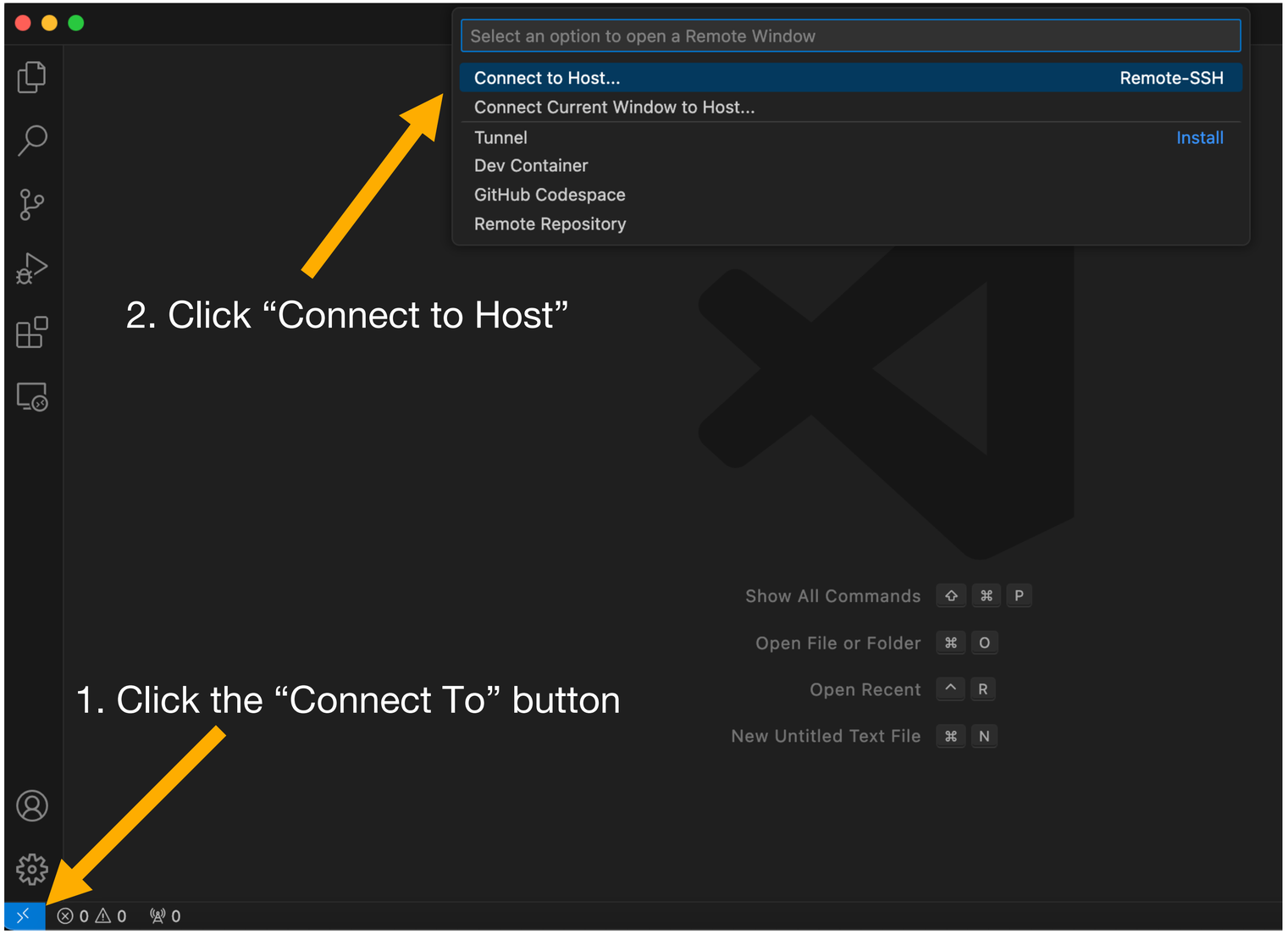
Videos

- Fast Code / File Navigation in VSCode
- Debugging on SubMIT with VSCode

Visual Studio Code: How to Connect

Note: This puts you on a login node
Only for relatively light computation use (debugging)

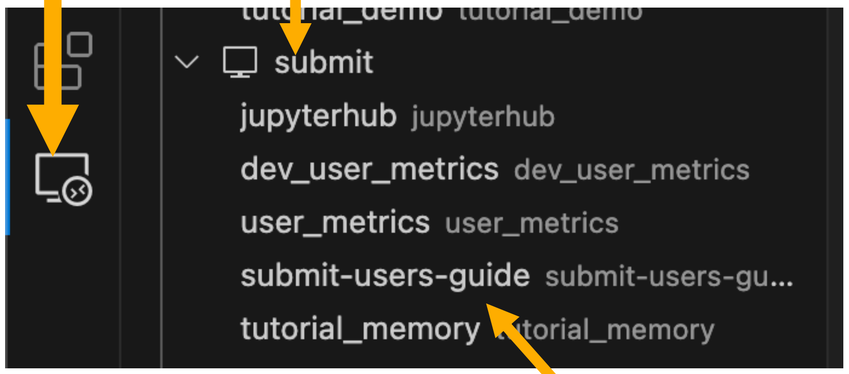
subMIT User's Guide:
https://submit.mit.edu/submit-users-guide/access.html#getting-started-with-vscode-on-submit



Click "Remote Explorer"

Then remote hosts in your ssh config file appear

OR



… along with previously-opened folders on each host

Terminal / SSH

- Classic method of interaction

```
ssh <username>@submit.mit.edu
```

Note: **Initially, this puts you on a login node**
Only for relatively light computation use

Use **srun <resource request> --pty bash** to get interactive session on compute nodes for heavy use

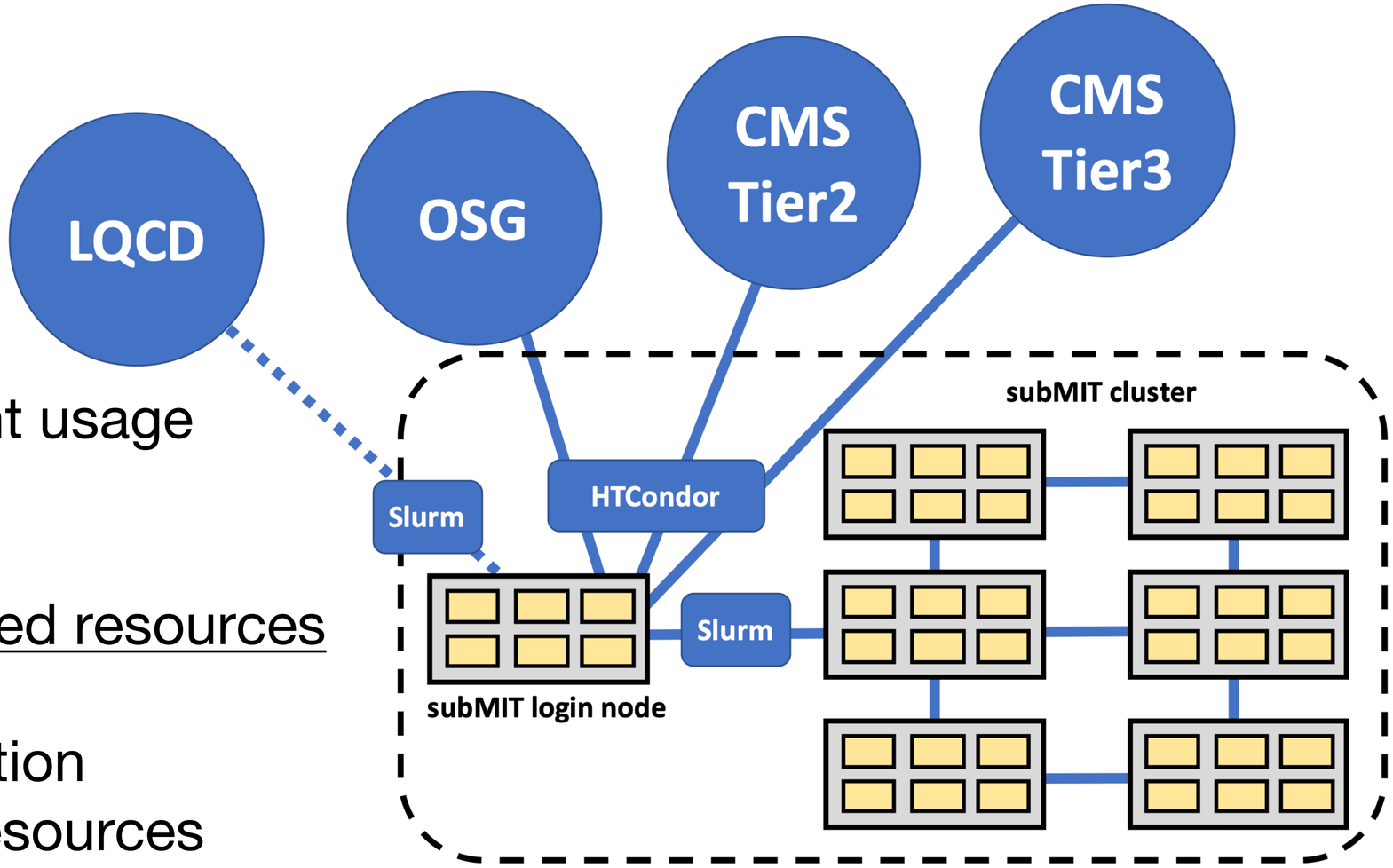
- MacOS / Linux : built-in. Windows : Windows Subsystem for Linux (WSL)

- Handy tip: set up ssh config file:
  - https://submit.mit.edu/submit-users-guide/starting.html#common-issues-with-keys

- subMIT User's Guide
  - Intro to terminal: https://submit.mit.edu/submit-users-guide/tutorials/tutorial_0.html

Batch Jobs (overview)

- See the tutorial later today for more info!



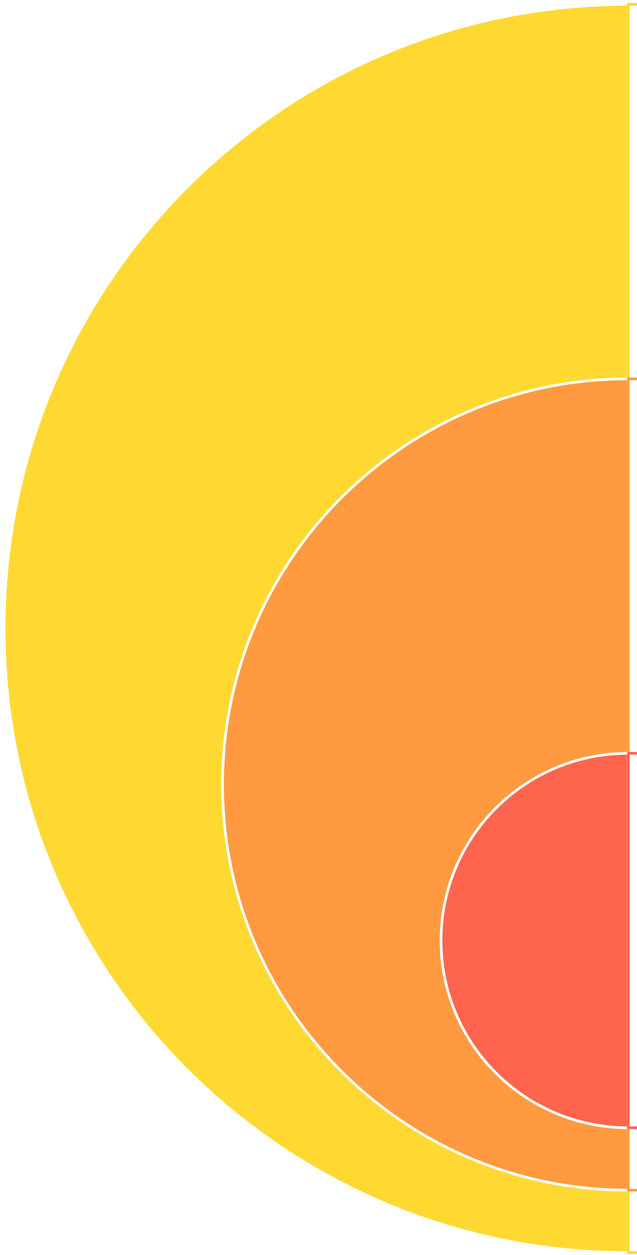
- Login nodes are only for light usage
- What are Batch jobs?
  - request/reserve dedicated resources (nodes, cores, memory)
  - run without user interaction
  - may wait in queue for resources
  - one or many submitted/run at once
- “Heavy lifting” (significant usage of resources) should be done via batch jobs or interactive SLURM jobs

Initial (one-time) Account Creation

Everyone with a Physics association is eligible, including collaborators

1. Email submit-help@mit.edu asking for an account. Include your
   - affiliation with Physics
   - Kerberos username (MIT email address)
2. Wait for a response
3. Generate ssh keys (https://submit.mit.edu/submit-users-guide/starting.html)
4. Upload your public key to the SubMIT Portal: https://submit-portal.mit.edu/

Recap



Login nodes for light usage

Heavy resource (memory, CPU) consumption should be via batch job (SLURM, HTCondor) or interactive SLURM job

Low-barrier, GUI access:

- JupyterHub (compute node)
- Visual Studio Code (login node)

Workhorse / advanced access:

- terminal / ssh (login node → compute node via SLURM salloc/srun)