What is Jupyter?

- At NERSC, we say “Jupyter” in reference to a collection of many things
  - Access shareable Jupyter “notebooks” via JupyterHub
- What can I put in a Jupyter notebook?
  - Live code
  - Equations
  - Visualizations
  - Narrative text
  - Interactive widgets
- What applications would I use a notebook for?
  - Data cleaning and data transformation
  - Numerical simulation
  - Statistical modeling
  - Data visualization
  - Machine learning
  - Workflows and analytics frameworks
How Do I Use Jupyter at NERSC?

- [https://jupyter.nersc.gov](https://jupyter.nersc.gov)

Authenticate  Choose  Go!
# How Do I Choose a Notebook Server to Spawn?

**Shared CPU:**
- Notebook on one of 40 login nodes
- Same Python env as SSH login
- Can submit jobs via `sbatch`

**Exclusive CPU/GPU:**
- Notebook in job allocation
- CPU node or GPU node
- Uses NERSC hours

**Configurable Job:**
- Notebook in job allocation
- CPU node(s) or GPU node(s)
- Uses NERSC hours
- Can be used in reservations

<table>
<thead>
<tr>
<th></th>
<th>Shared CPU Node</th>
<th>Exclusive CPU Node</th>
<th>Exclusive GPU Node</th>
<th>Configurable Job</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Perlmutter</strong></td>
<td>start</td>
<td>start</td>
<td>start</td>
<td>start</td>
</tr>
<tr>
<td><strong>Resources</strong></td>
<td>Use a node shared with other users’ notebooks but outside the batch queues.</td>
<td>Use your own node within a job allocation using defaults.</td>
<td>Use multiple compute nodes with specialized settings.</td>
<td></td>
</tr>
<tr>
<td><strong>Use Cases</strong></td>
<td>Visualization and analytics that are not memory intensive and can run on just a few cores.</td>
<td>Visualization, analytics, machine learning that is compute or memory intensive but can be done on a single node.</td>
<td>Multi-node analytics jobs, jobs in reservations, custom project charging, and more.</td>
<td></td>
</tr>
</tbody>
</table>

**Shared = other users and processes on the same node**

**Exclusive and configurable = compute nodes just for your notebook and processes**

---

[NERSC logo]

**NERSC**

[BERKELEY LAB logo]

**BERKELEY LAB**

[DOE logo]

**U.S. DEPARTMENT OF ENERGY**

[Office of Science logo]

**Office of Science**
# Configurable Job Settings

## Server Options

<table>
<thead>
<tr>
<th>Account (&quot;_g&quot; suffix will be added as needed):</th>
</tr>
</thead>
<tbody>
<tr>
<td>nstaff</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constraint:</th>
</tr>
</thead>
<tbody>
<tr>
<td>gpu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>QOS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>jupyter</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cpus-per-task (node has 128 cpus):</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>gpus-per-task (node has 4 GPUs):</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>nodes (maximum of 4 for jupyter QOS):</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ntasks-per-node:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reservation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>(None)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>time (time limit in minutes):</th>
</tr>
</thead>
<tbody>
<tr>
<td>360</td>
</tr>
</tbody>
</table>

---

**Configurable Job**

- Use multiple compute nodes with specialized settings.
- Multi-node analytics jobs, jobs in reservations, custom project charging, and more.

**Start**
JupyterLab Interface: NERSC Add-ons

- Favorites
- Bookmark your favorite places on the file systems
- Pre-populated with $HOME and $PSCRATCH
- Add the current directory by clicking the ★ icon
JupyterLab Interface: NERSC Add-ons

- **Open from Path...**
- **Jump to anywhere in the file system**

- **Recents**
- **Recent locations you’ve visited on the file system**
Kernels: How You Compute with Jupyter

- The kernel is what actually runs your code
- Default kernel is NERSC Python
  - From Python module
- Other kernels also provided
  - Julia, R
  - ML packages
- Bring your own kernel

https://docs.jupyter.org/en/latest/projects/architecture/content-architecture.html
Your Own Jupyter Kernel

● A common Jupyter question:
  ○ “How do I take a conda environment and use it from Jupyter?”

● Several ways to accomplish this; we recommend:

$ module load python
$ conda create -n myenv python=3.9
$ source activate myenv
(myenv) $ conda install ipykernel <other-packages> ...
(myenv) $ python -m ipykernel install --user --name myenv-jupyter

● Point your browser to jupyter.nersc.gov
  ○ May need to restart notebook server via control panel
● Kernel “myenv-jupyter” should be present in the kernel list

This creates a “kernelspec” file
The kernelspec File

```
(myenv) user@login01:~$ cat \\
   $HOME/.local/share/jupyter/kernels/myenv-jupyter/kernel.json
{
  "argv": [
    "/global/homes/u/user/.conda/envs/myenv/bin/python",
    "-m",
    "ipykernel_launcher",
    "-f",
    "{connection_file}"
  ],
  "display_name": "myenv-jupyter",
  "language": "python"
}
```
Additional Customization

{
    "argv": [
        "/global/homes/u/user/.conda/envs/myenv/bin/python",
        "-m",
        "ipykernel_launcher",
        "-f",
        "{connection_file}"
    ],
    "display_name": "myenv-jupyter",
    "language": "python",
    "env": {
        "PATH": ...
    }
}
The kernel helper script is the most flexible approach for NERSC users since it easily enables use of modules, environment variables, etc.

```bash
#!/bin/bash
export SOMETHING=123
module load foo
exec python -m ipykernel "$@
```
A Shifter kernelspec File

```json
{
    "argv": [
        "shifter",
        "--image=continuumio/anaconda3:latest",
        "/opt/conda/bin/python",
        "-m",
        "ipykernel_launcher",
        "-f",
        "{connection_file}"
    ],
    "display_name": "my-shifter-kernel",
    "language": "python"
}
```
Debugging Jupyter Issues

(myenv) user@login01:~$ cat ~/.jupyter-perlmutter.log

[IPKernelApp] ERROR | No such comm target registered: jupyter.widget.control
[IPKernelApp] WARNING | No such comm: aa07e0e8-5f78-4899-ab3f-8af339f1318e
[I 2023-06-12 14:20:17.036 SingleUserLabApp kernelmanager:321] Starting buffering for fcb31e09-6a2a-427e-aaf8-f15d1a443bda:fbe5d17f-91a2-49d7-bf22-1da23dc8ef4b
[I 2023-06-12 14:20:17.111 SingleUserLabApp kernelmanager:321] Starting buffering for fac60c02-f294-4a49-b711-89501fedefc8:006691d0-c3c5-480c-aacb-ffde01ab6169
[W 2023-06-12 14:20:17.291 SingleUserLabApp kernelmanager:321] Starting buffering for b9cb4f21-1f8c-4917-b7a5-4653b158d87b:230a9755-8454-4f84-a097-041c7e885bb
[IPKernelApp] ERROR | No such comm target registered: jupyter.widget.control
[IPKernelApp] WARNING | No such comm: 8844d734-bdf7-4159-b1ab-4534db8105b6
Jupyter Usage at NERSC

For comparison, about 3000 users per month connect via ssh.
Go to https://jupyter.nersc.gov to use Jupyter at NERSC
Use a kernelspec to use a conda environment in your notebook
You can customize those kernelspec files in many ways
We work on making Jupyter work and work better for you

Always looking for:
- New ways to empower Jupyter users
- Feedback, advice, and even help: https://help.nersc.gov/

Thank you!