

HIGH PERFORMANCE RESEARCH COMPUTING

ACES: Advanced Python for High Performance Computing Workflows

April 14, 2026

Richard Lawrence



High Performance
Research Computing

DIVISION OF RESEARCH

Outline

- ACES Login
- Modules
- Jupyter
- Virtual Environments
- Conda Environments
- Containers
- Parallelism with Joblib and Dask
- Checkpointing with Pickle

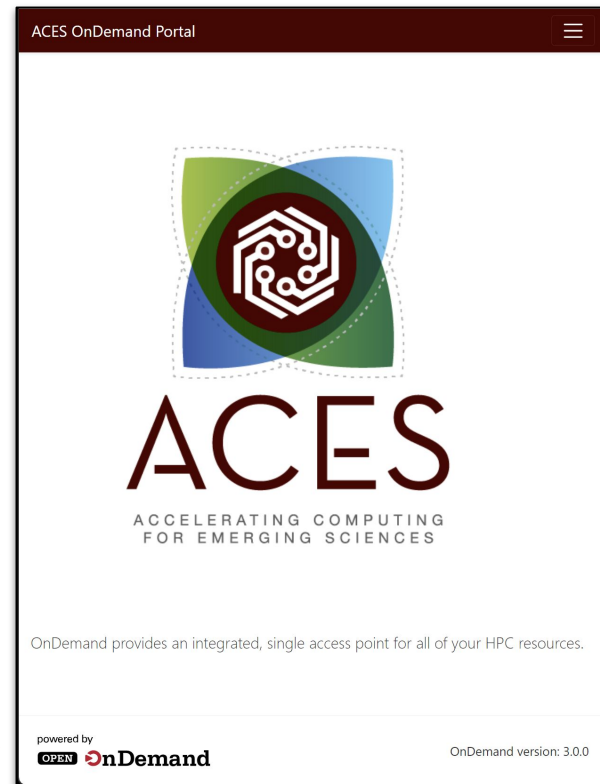
ACES Login

ACES Portal



ACES Portal portal-aces.hprc.tamu.edu
is the web-based user interface for the ACES cluster

Open OnDemand (OOD) is an advanced web-based
graphical interface framework for HPC users



Accessing ACES via the Portal (ACCESS)






The screenshot shows the ACCESS portal interface. At the top, there is a navigation bar with links for ALLOCATIONS, SUPPORT, OPERATIONS, and METRICS, along with a search icon and a Login link. Below the navigation bar is the ACCESS logo. A section titled "Consent to Attribute Release" contains a message from TAMU ACES ACCESS OIDC requesting access to user information, including CILogon user identifier, name, email address, and affiliation. Below this is a "Select an Identity Provider" dropdown menu with "ACCESS CI (XSEDE)" selected. A red box highlights this dropdown menu. Below the dropdown is a "Remember this selection" checkbox and a yellow "LOG ON" button.


Select the Identity Provider appropriate for your account.

The screenshot shows the ACCESS portal login form. At the top is the ACCESS logo. Below it is a message: "If you had an XSEDE account, please enter your XSEDE username and password for ACCESS login." There are two input fields: "ACCESS ID" and "ACCESS Password". Below these fields is a yellow "LOGIN" button. At the bottom, there are links for "Register for an ACCESS ID", "Forgot your password?", and "Need Help?".

Log-in using your ACCESS or institutional credentials.

The ACES Portal

ACES OnDemand Portal Files ▾ Jobs ▾ Clusters ▾ Interactive Apps ▾ Affinity Groups ▾ Dashboard ▾ Utilities ▾     


ACES
ACCELERATING COMPUTING
FOR EMERGING SCIENCES

Today we'll mostly use these two tabs

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

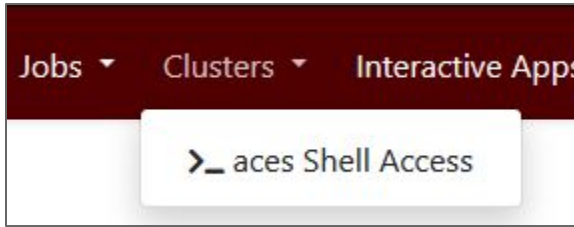
Message of the Day

IMPORTANT POLICY INFORMATION

- **Unauthorized use of HPRC resources is prohibited and subject to criminal prosecution.**
- **Use of HPRC resources in violation of United States export control laws and regulations is prohibited.**
- **Sharing HPRC account and password information is in violation of State Law. Any shared accounts will be DISABLED.**
- **Authorized users must also adhere to ALL policies at: <https://hprc.tamu.edu/policies>**

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

ACES Shell



Select:
Clusters
-> _aces Shell Access

Welcome to the ACES
login node.

```
portal-aces.hprc.tamu.edu/pun/sys/shell/ssh/login.aces
Host: login.aces Themes: Default

=====
| Texas A&M University High Performance Research Computing |
| Website: https://hprc.tamu.edu |
| Consulting: help@hprc.tamu.edu (preferred) or (979) 845-0219 |
| ACES Documentation: https://hprc.tamu.edu/kb/User-Guides/ACES |
| FASTER Documentation: https://hprc.tamu.edu/kb/User-Guides/FASTER |
| Grace Documentation: https://hprc.tamu.edu/kb/User-Guides/Grace |
| Terra Documentation: https://hprc.tamu.edu/kb/User-Guides/Terra |
| YouTube Channel: https://www.youtube.com/texasamhprc |
|=====

*****
* == IMPORTANT POLICY INFORMATION == *
* - Unauthorized use of HPRC resources is prohibited and subject to *
* criminal prosecution. *
* - Use of HPRC resources in violation of United States export control *
* laws and regulations is prohibited. Current HPRC staff members are *
* US citizens and legal residents. *
* - Sharing HPRC account and password information is in violation of *
* Texas State Law. Any shared accounts will be DISABLED. *
* - Authorized users must also adhere to ALL policies at: *
* https://hprc.tamu.edu/policies/ *
*****

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

Please restrict usage to 8 CORES across ALL login nodes.
Users found in violation of this policy will be SUSPENDED.

To see these messages again, run the motd command.
[u.rl117197@aces-login2 ~]$
```

Modules

Module Usage Basics

module avail

- Lists all available modules (may be slow).
- Navigation:
 - spacebar, arrows, or j and k
 - quit with q
- Case-sensitive search: /
- Use **mla** instead to save results to a file as well (will be named `module.avail.aces` or similar)

module spider <word>

- Case-sensitive search for modules with “word” in name.
- Provide an exact name to see dependencies.

```
[u.jw123527@aces-login2 ~]$ module spider Python

Python:
-----
Description:
  Python is a programming language that lets you work more quickly and integrate your systems more effectively.

Versions:
  Python/2.7.18-bare
  Python/2.7.18
  Python/3.8.6
  Python/3.9.5-bare
  Python/3.9.5
  Python/3.9.6-bare
  Python/3.9.6
  Python/3.10.4-bare
  Python/3.10.4
  Python/3.10.8-bare
  Python/3.10.8
  Python/3.11.3
  Python/3.11.5

Other possible modules matches:
  Biopython Boost.Python Brotli-python IPython LASSO-Python Python-bundle-PyPI flatbuffers-python graphvi

To find other possible module matches execute:

$ module -r spider '.*Python.*'
```

Module Usage Basics

`module load <module>`

- add <module> paths to the current environment variables

`module list`

- see what modules are loaded in your current session

`module purge`

- unload all modules

```
[u.jw123527@aces-login2 ~]$ module load GCCcore/13.2.0
[u.jw123527@aces-login2 ~]$ module list
```

```
Currently Loaded Modules:
 1) GCCcore/13.2.0
```

```
[u.jw123527@aces-login2 ~]$ module purge
[u.jw123527@aces-login2 ~]$ module list
No modules loaded
[u.jw123527@aces-login2 ~]$
```

Hands-on Exercise: Module Loading

1. `m1a Python` -See which versions of Python are available.
2. `m1 Python/3.11.3` -error! You cannot do that yet.
3. `module spider Python/3.11.3` -Learn how to load this module.
4. `m1 Python/3.11.3` -Fill in the blank (with the correct toolchains) to load this module.
5. `m1 list` -List all loaded modules.
6. `m1 GCC/13.2.0` -Change version of a loaded Toolchain module (GCC); notice the message about reloaded modules.
7. `m1 list` -List all loaded modules.
8. `m1 purge` -Remove all loaded modules.

Jupyter

Jupyter Notebook for Python Training

The screenshot shows the ACES OnDemand Portal interface. The top navigation bar includes 'Files', 'Jobs', 'Clusters', 'Interactive Apps', 'Affinity Groups', and 'Chatt'. The 'Interactive Apps' dropdown menu is open, listing various applications. The 'Jupyter Notebook for Python Training' option is highlighted with a red box. The main content area displays the ACES logo, a message of the day, and status updates.

ACES OnDemand Portal Files Jobs Clusters **Interactive Apps** Affinity Groups Chatt

GUI

- VNC
- NextSilicon VNC

Imaging

- CryoSPARC
- ImageJ
- Jmol
- Paraview
- cisTEM

Servers

- Jupyter Notebook
- Jupyter Notebook for Python Training**
- JupyterLab
- RStudio
- TensorBoard
- Tutorials OnDemand

ACES
ACCELERATING COMPUTING
FOR EMERGING SCIENCES

OnDemand provides an integrated, single access point

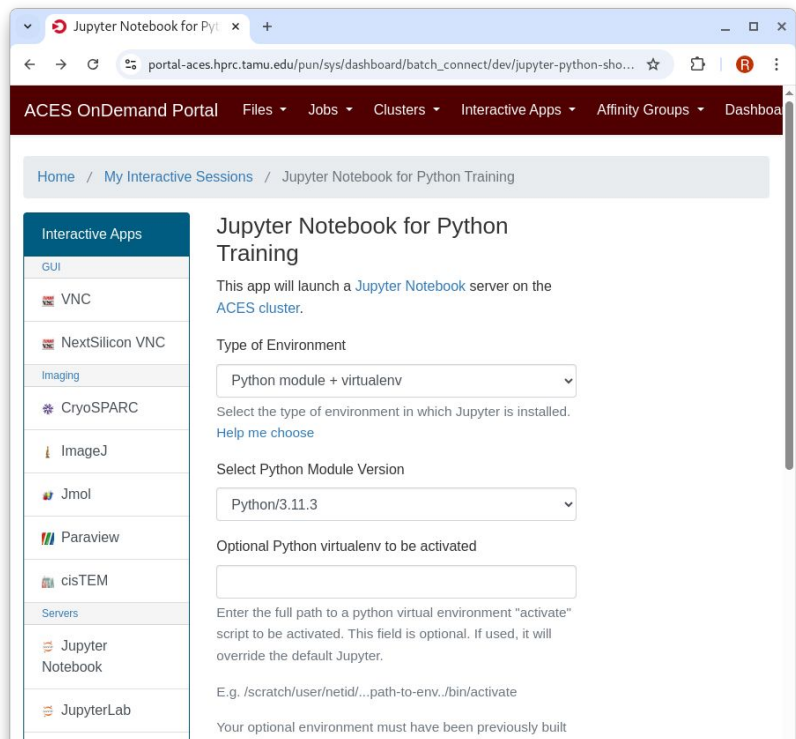
Message of the Day

ACES Status, January 27
The pvc queue is now available with many nodes with PVCs composed of fabrics.

ACES Accounting, January 13
NOTICE: Accounting on ACES will be activated on Monday, January 13, 2025.

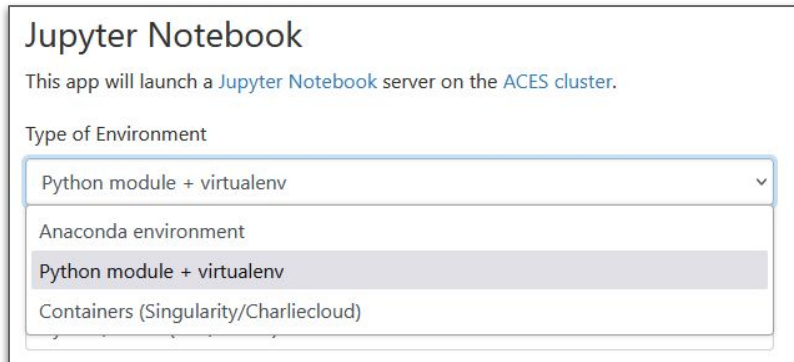
Jupyter Notebook Interactive App

We'll come back to this form several times this class. Note the "Type of Environment" menu.



The screenshot shows a web browser window with the URL `portal-aces.hprc.tamu.edu/pun/sys/dashboard/batch_connect/dev/jupyter-python-sho...`. The page title is "Jupyter Notebook for Python Training". On the left is a sidebar menu with categories: Interactive Apps, GUI, VNC, NextSilicon VNC, Imaging, CryoSPARC, ImageJ, Jmol, Paraview, cisTEM, Servers, Jupyter Notebook, and JupyterLab. The main content area is titled "Jupyter Notebook for Python Training" and contains the following text and form elements:

- Description: "This app will launch a [Jupyter Notebook](#) server on the ACES cluster."
- Form: "Type of Environment" dropdown menu with "Python module + virtualenv" selected.
- Text: "Select the type of environment in which Jupyter is installed. [Help me choose](#)"
- Form: "Select Python Module Version" dropdown menu with "Python/3.11.3" selected.
- Form: "Optional Python virtualenv to be activated" text input field.
- Text: "Enter the full path to a python virtual environment 'activate' script to be activated. This field is optional. If used, it will override the default Jupyter."
- Text: "E.g. `/scratch/user/netid/...path-to-env./bin/activate`"
- Text: "Your optional environment must have been previously built"



This image is a close-up of the "Type of Environment" dropdown menu from the screenshot above. The menu is open, showing the following options:

- Python module + virtualenv (selected)
- Anaconda environment
- Python module + virtualenv
- Containers (Singularity/Charliecloud)

Jupyter Notebook for Python Training

Number of cores (4)

Launch

* The Jupyter Notebook for Python Training session data for this session can be accessed under the [data root directory](#).

The screenshot displays a list of Jupyter Notebook sessions. The top session is 'Queued', the middle one is 'Starting', and the bottom one is 'Running'. The 'Running' session details include: Host: >_ ac014, Created at: 2025-04-24 13:28:18 CDT, Time Remaining: 28 hours, and Session ID: 3d1d7b71-7ded-4e3a-b670-c451ddaadbc6. A 'Delete' button is visible next to the host information. A 'Connect to Jupyter' button is highlighted with a red box at the bottom of the running session details.

Session Name	Status
Jupyter Notebook for Python Training (1053157)	Queued
Jupyter Notebook for Python Training (1053157)	Starting (1 node 4 cores)
Jupyter Notebook for Python Training (1053157)	Running (1 node 4 cores)

Host: >_ ac014 Delete

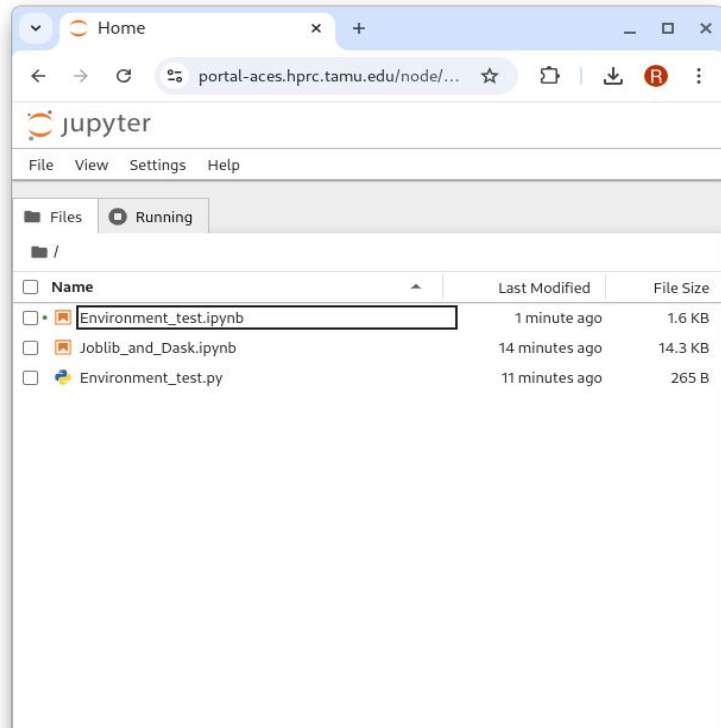
Created at: 2025-04-24 13:28:18 CDT

Time Remaining: 28 hours

Session ID: 3d1d7b71-7ded-4e3a-b670-c451ddaadbc6

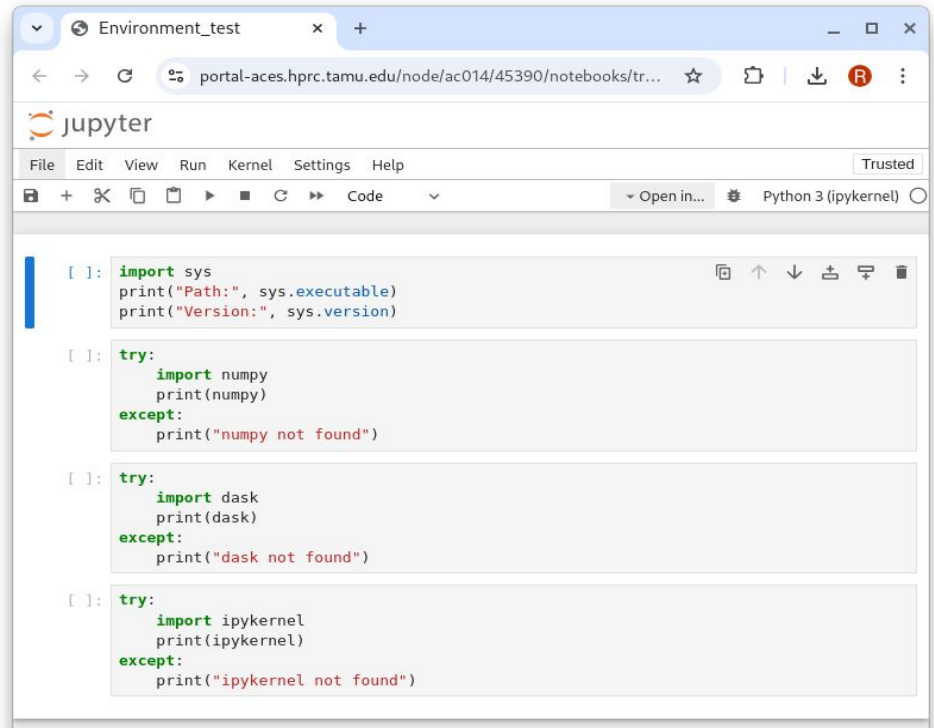
Connect to Jupyter

Environment Test Notebook



The screenshot shows the JupyterLab interface with the file browser on the left. The file 'Environment_test.ipynb' is selected and highlighted. The table below shows the details of the files in the current directory.

Name	Last Modified	File Size
Environment_test.ipynb	1 minute ago	1.6 KB
Joblib_and_Dask.ipynb	14 minutes ago	14.3 KB
Environment_test.py	11 minutes ago	265 B



The screenshot shows the JupyterLab code editor with the following Python code:

```
[ ]: import sys
print("Path:", sys.executable)
print("Version:", sys.version)

[ ]: try:
import numpy
print(numpy)
except:
print("numpy not found")

[ ]: try:
import dask
print(dask)
except:
print("dask not found")

[ ]: try:
import ipykernel
print(ipykernel)
except:
print("ipykernel not found")
```

Environment Test Script

```
cd $SCRATCH/training-2026-04-14
```

```
./Environment_test.py
```

```
Path: /usr/bin/python3
```

```
Version: 3.6.8 (default, Nov 15 2024, 08:11:39)
```

```
[GCC 8.5.0 20210514 (Red Hat 8.5.0-22)]
```

```
numpy not found
```

```
dask not found
```

```
ipykernel not found
```

```
module load foss/2022b SciPy-bundle/2023.02
```

```
./Environment_test.py
```

```
Path: /sw/eb/sw/Python/3.10.8-GCCcore-12.2.0/bin/python3
```

```
Version: 3.10.8 (main, Apr 4 2023, 12:24:51) [GCC 12.2.0]
```

```
<module 'numpy' from '/sw/eb/sw/SciPy-bundle/2023.02-gfbf-2022b/lib/python3.10/... '>
```

```
dask not found
```

```
ipykernel not found
```

Virtual Environments

Python and Virtual Environments

Python is a language which supports many external libraries in the form of extensions (called Python Packages).

Some commonly used packages:

- SciPy & NumPy
- Jupyter notebook
- Scikit-learn

You will not have these in your environment by default, but there are several ways to get them yourself.

We'll start with *virtual environments*.

See also: <https://hprc.tamu.edu/kb/Software/Python/#hprc-venv-management-tools>

Virtual Environments

- What is a virtual environment?
 - a file structure to hold different Python versions and package dependencies
- Why do I need one?
 - Without one, all packages are installed into your base Python Installation, which may cause conflicts across different projects.
- A good practice is to name your environment so that you can identify which Python version is in your virtualenv so that you know which module to load.
- Typically you have a virtual environment for each of your projects.

Hands-on Exercise: Python Software Install

```
1. ml purge  
ml GCCcore/12.3.0 Python/3.11.3
```

```
2. create_venv myEnv [-d <description>]
```

```
3. source activate_venv myEnv
```

```
4. cd $SCRATCH/training-2026-04-14  
./Environment_test.py
```

```
5. pip install dask
```

```
6. ./Environment_test.py
```

```
7. deactivate
```

-Set up the underlying Python module

-Use HPRC's "create_venv" tool.
Creates a virtual environment in
\$SCRATCH/virtual_envs

-Activate virtual environment.

-Check if dask is installed (it is not).

-Install dask.

-Where is dask installed?

-Close virtual environment.

Also Recommended

```
source activate_venv myEnv
```

```
pip install notebook numpy joblib dask ipycytoscape
```

needed for parallel exercises later

```
python3 -m ipykernel install --user --name=myEnv
```

ipykernel module from the notebook package has an additional (optional) installation step

```
deactivate
```

```
list_venvs
```

Virtual Environments in Jupyter

To use your virtualenv in a Jupyter notebook:

- Select the Jupyter Notebook Interactive App in the portal.
- Choose “Python module + virtualenv”.
- Select the module used to build your venv.

Jupyter Notebook

This app will launch a [Jupyter Notebook](#) server on the [ACES cluster](#).

Type of Environment

Python module + virtualenv

Select the type of environment in which Jupyter is installed. [Help me choose](#)

Select Python Module Version

Python/3.11.3 (foss/2023a)

Python/3.11.3 (foss/2023a)

Python/3.10.8 (foss/2022b)

Python/3.9.6 (foss/2021b)

Enter the full path to a python virtual environment "activate" script to be activated. This field is optional. If used, it will override the default Jupyter.

E.g. /scratch/user/netid/...path-to-env../bin/activate

Your optional environment must have been previously built with the Python module selected in the Module option above. It is expected to have a Jupyter package installed. Please see [instructions](#).

Select Path

Virtual Environments in Jupyter

After selecting the module, provide the path to the 'activate' script of your virtual environment.

Can type it in directly... →

yours (likely) at
`$SCRATCH/virtual_envs/myEnv/bin/activate`

backup copy at
`$SCRATCH/training-2026-04-14/training-env/bin/activate`

...Or select it with a mini file browser here →

Jupyter Notebook

This app will launch a [Jupyter Notebook](#) server on the [ACES](#) cluster.

Type of Environment

Select the type of environment in which Jupyter is installed. [Help me choose](#)

Select Python Module Version

Optional Python virtualenv to be activated

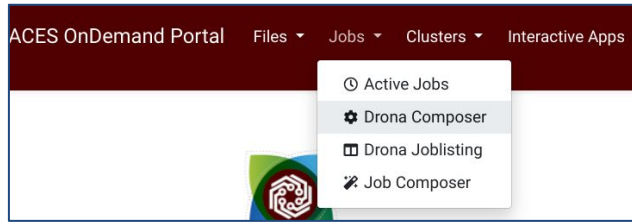
Enter the full path to a python virtual environment "activate" script to be activated. This field is optional. If used, it will override the default Jupyter.

E.g. `/scratch/user/netid/...path-to-env./bin/activate`

Your optional environment must have been previously built with the Python module selected in the Module option above. It is expected to have a Jupyter package installed. Please see [instructions](#).

Running Python Codes Using Drona

- Drona assists researchers in generating and submitting (Python) jobs
- Select Python environment
- Choose your virtual env from dropdown or create one on the fly (uses create_venv)
- Select main python script and other required scripts/data.
- Drona generate and submit batch job on user's behalf

A screenshot of the Drona job configuration form. The form includes fields for 'Job Name', 'Location' (with a 'Change' button and the path '/scratch/user/u.mp108705/drona_composer/runs'), and 'Environments' (set to 'Python'). The 'Python Environment' dropdown is open, showing options: 'Private virtual env', 'PyTorch', 'Private virtual env', 'Create new private virtual env', 'Shared virtual env', and 'Module (latest)'. The 'My environments' dropdown is set to '- Choose an option -'. There is an 'Upload files/data' section with a 'Select an option' dropdown and an 'Add' button. A 'Hide History' button is located at the bottom right of the form.

Conda Environments

[Ana]Conda

Conda is a package manager that may handle dependencies better than plain virtual environments and pip... but has more clutter.

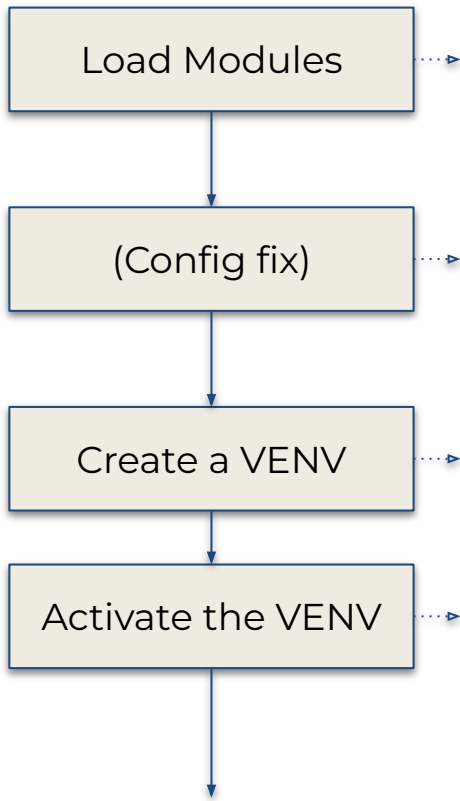
Anaconda is a collection of packages including Conda and many other tools useful to Python programmers.

So loading an Anaconda module will give you some useful packages to start and you'll use Conda to add more.

See our KB page for more information:

<https://hprc.tamu.edu/kb/Software/ANACONDA/>

Conda Environments



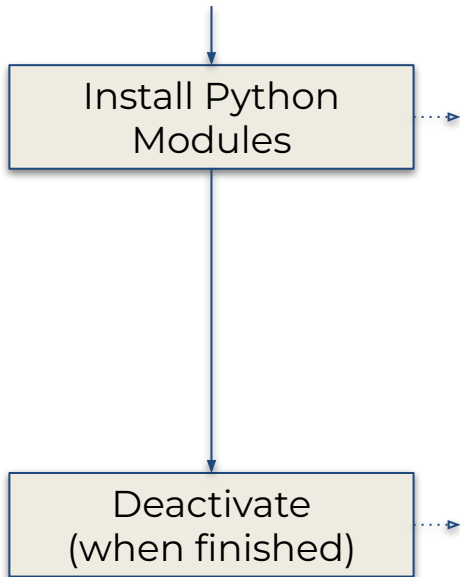
```
# clean up and load Anaconda  
module purge  
module load Anaconda3/2024.02-1
```

```
#1-time only: stop some Anaconda auto-loading  
conda config --set auto_activate_base False
```

```
# create a Python virtual environment  
conda create -n myEnv
```

```
# activate the virtual environment  
source activate myEnv
```

Conda Environments



```
# install required packages for use in the portal  
conda install dask  
conda install -c conda-forge notebook  
conda install -c conda-forge ipycytoscape
```

```
# check what we have installed  
cd $SCRATCH/training-2026-04-14  
./Environment_test.py
```

```
# deactivate the virtual environment  
# conda deactivate
```

Anaconda Environments

To use your Anaconda environment in a Jupyter notebook:

- Select the Jupyter Notebook Interactive App in the portal.
- Choose “Anaconda environment”.
- Select the module used to build your environment.
- Select your environment.

Jupyter Notebook

This app will launch a [Jupyter Notebook](#) server on the [ACES cluster](#).

Type of Environment

Anaconda environment

Select the type of environment in which Jupyter is installed. [Help me choose](#)

Select Anaconda Version

Anaconda3/2024.02-1

Select a Conda module to load. All modules listed include Python and Jupyter.

Optional Conda Environment to be Activated

myEnv

Enter the name of environment to be activated. This field is optional.

Your optional environment must have been previously built with the Anaconda module selected in the Module option above. Please see [instructions](#).

[Select Path](#)

Containers

Set Up Your Singularity Environment

Get to a compute node from the login node.

```
srun --time=120 --ntasks=1 --mem=4G --pty bash -i
```

following along live? add:
--reservation=training_python

Return to your tutorial directory (if necessary).

```
cd $SCRATCH/training-2026-04-14
```

Set your singularity cache directory for temporary files.

```
export SINGULARITY_CACHEDIR=$TMPDIR
```

Connect to the internet for fetching images.

```
module load WebProxy
```

Singularity Pull

Singularity can fetch images from repositories and also convert them to the singularity file format at the same time.

```
singularity pull [target-filename] <source>
```

Where <source> refers to something on the internet. The syntax depends on where the source is located.

and [target-filename] includes the file extension.

Singularity Pull Example

The `<source>` argument for Docker images looks like

```
docker://<group>/<name>[:<tag>]
```

Therefore, the pull command for the Jupyter example is

```
singularity pull docker://jupyter/scipy-notebook:latest
```

The default filename will be `scipy-notebook_latest.sif`

(Download now, or just use the backup copy linked to
`/scratch/training/singularity/scipy-notebook_2023.sif`)

Singularity Environment Test

```
cd $SCRATCH/training-2026-04-14
./Environment_test.py
```

```
Path: /usr/bin/python3
```

```
Version: 3.6.8 (default, Nov 15 2024, 08:11:39)
```

```
[GCC 8.5.0 20210514 (Red Hat 8.5.0-22)]
```

```
numpy not found
```

```
dask not found
```

```
ipykernel not found
```

```
singularity exec -B /scratch scipy-notebook_2023.sif ./Environment_test.py
singularity exec -B /scratch scipy-notebook_latest.sif ./Environment_test.py
```

```
Path: /opt/conda/bin/python3
```

```
Version: 3.11.6 | packaged by conda-forge | (main, Oct 3 2023, 10:40:35) [GCC 12.3.0]
```

```
<module 'numpy' from '/opt/conda/lib/python3.11/site-packages/numpy/__init__.py'>
```

```
<module 'dask' from '/opt/conda/lib/python3.11/site-packages/dask/__init__.py'>
```

```
<module 'ipykernel' from '/opt/conda/lib/python3.11/site-packages/ipykernel/__init__.py'>
```

Interactive Graphical Computing

The image shows a screenshot of the ACES OnDemand Portal interface. The top navigation bar is dark red and contains the following items: "ACES OnDemand Portal", "Files", "Jobs", "Clusters", "Interactive Apps", and "Dashboard". The "Interactive Apps" menu is open, showing a list of application categories and their respective icons and names:

- GUI
 - VNC
 - Nextsilicon VNC
- Imaging
 - CryoSPARC
 - ImageJ
 - cisTEM
- Servers
 - Jupyter Notebook
 - JupyterLab
 - RStudio

The "Jupyter Notebook" option in the Servers category is highlighted with a yellow box. A red arrow points from a box containing the text "click click" to the "Interactive Apps" menu, and another red arrow points from the same box to the "Jupyter Notebook" option. To the left of the menu is a large, stylized logo consisting of overlapping green, blue, and red shapes with a white circuit-like pattern in the center.

Containerized Jupyter Notebook

Home / My Interactive Sessions / Jupyter Notebook

Interactive Apps

- GUI
- Matlab
- VNC**
- NextSilicon VNC
- Imaging
- CryoSPARC
- ImageJ
- Jmol
- Paraview
- cisTEM
- Servers
- Jupyter Notebook

Jupyter Notebook

This app will launch a *Jupyter Notebook* server on the ACES cluster.

Type of Environment

Containers (Singularity/Charliecloud)

Select the type of environment in which Jupyter is installed.
[Help me choose](#)

Path to container image file

`$SCRATCH/pyHPC/scipy-notebook_latest.sif`

Enter the full path to an image file. Recommended that this be located in your \$SCRATCH directory.

[Singularity](#) images and [Charliecloud](#) images are supported. Images should contain the Jupyter app. Charliecloud images should additionally have a /scratch directory.

Select Path

Choose *Containers*

Enter
`$SCRATCH/training-2026-04-14/scipy-notebook_latest.sif`
or wherever your file actually is (see Slide 34–35)

Backup copy at
`/scratch/training/singularity/scipy-notebook_2023.sif`

...Continued

click
...wait
click
...wait
click

Launch

Jupyter Notebook (5488) 1 node | 1 core | Starting

Jupyter Notebook (5489) 1 node | 1 core | Running

Host: >_ac110 Delete

Created at: 2023-09-21 15:39:52 CDT

Time Remaining: 56 minutes

Session ID: a5f41dfd-7c0d-44e3-aea7-7331c66a4d24

Connect to Jupyter

New Upload

- Notebook
- Terminal
- Console
- New File
- New Folder

last month

File Edit View Run Kernel Settings Help Trusted

+ ✂ 📄 📁 ▶ ⏪ ⏩ Code ...

```
[1]:  
import numpy  
print(numpy)
```

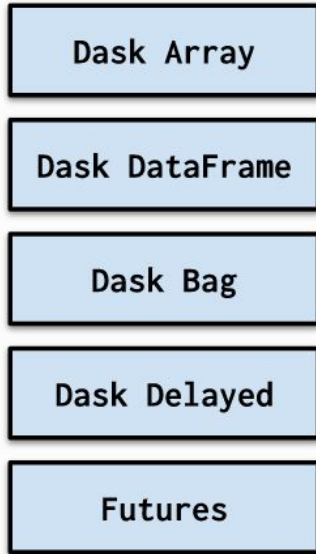
<module 'numpy' from '/opt/conda/lib/python3.11/site-packages/numpy/_init_.py'>

WOW

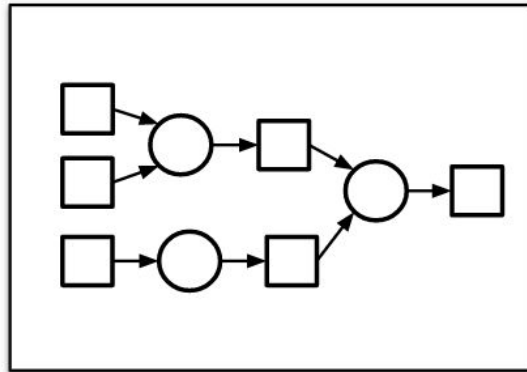
Parallelism with Joblib and Dask

Dask Concepts

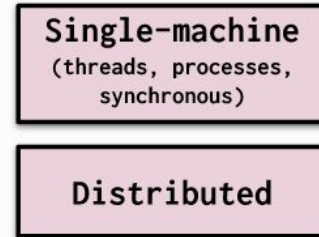
Collections
(create task graphs)



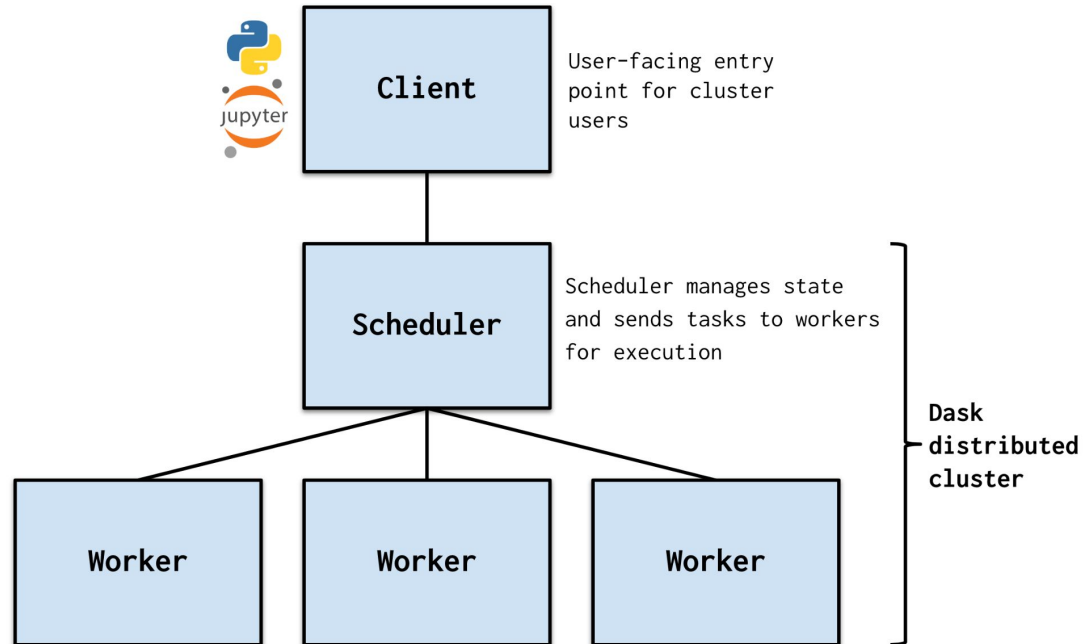
Task Graph



Schedulers
(execute task graphs)



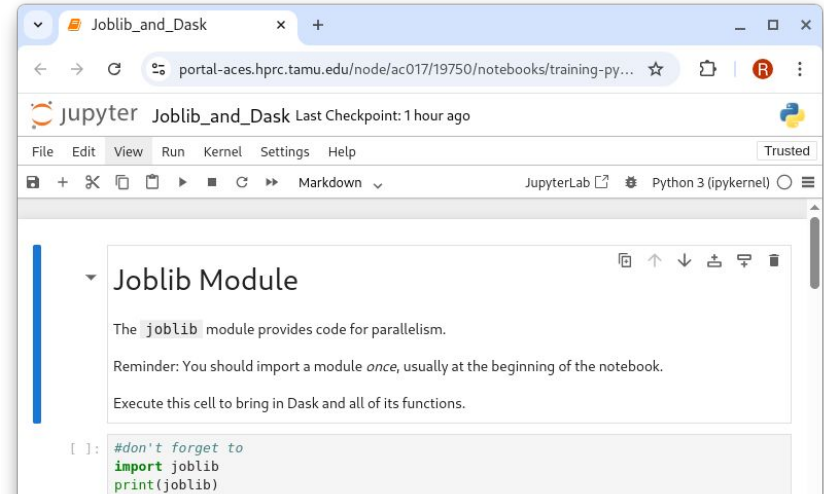
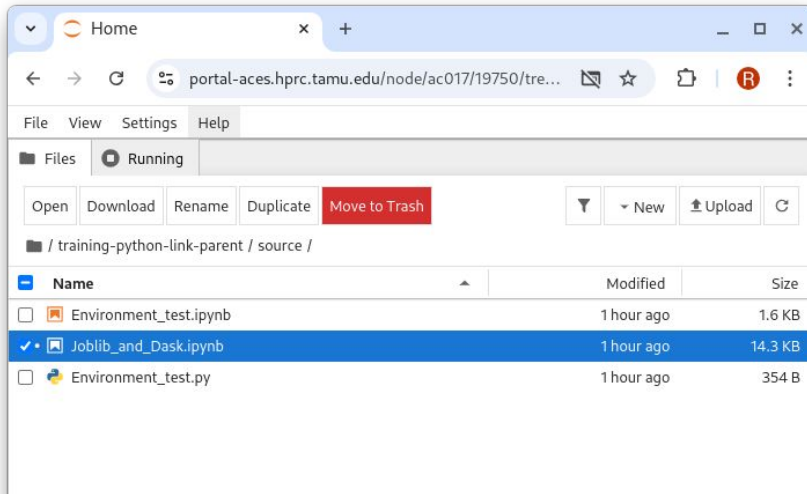
Dask Flowchart



Workers compute tasks / store and serve computed results to other workers or clients (Source: <https://docs.dask.org/en/latest/>)

Parallelism Exercises

- Use an environment that has the recommended modules:
 - myEnv
 - scipy-notebook_2023.sif



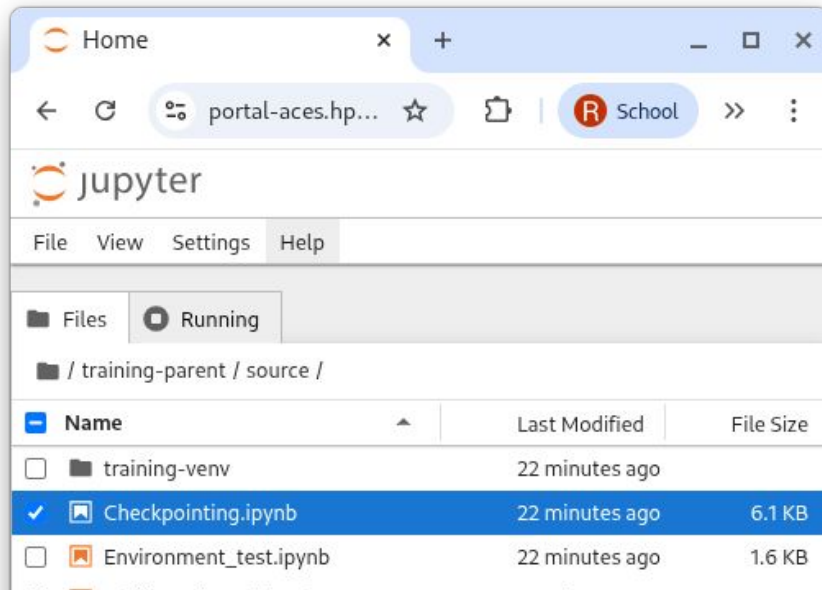
Checkpointing with Pickle

Checkpointing Concepts

- Serialization
 - Objects in memory → Binary file
- Deserialization
 - Binary file → Objects in memory

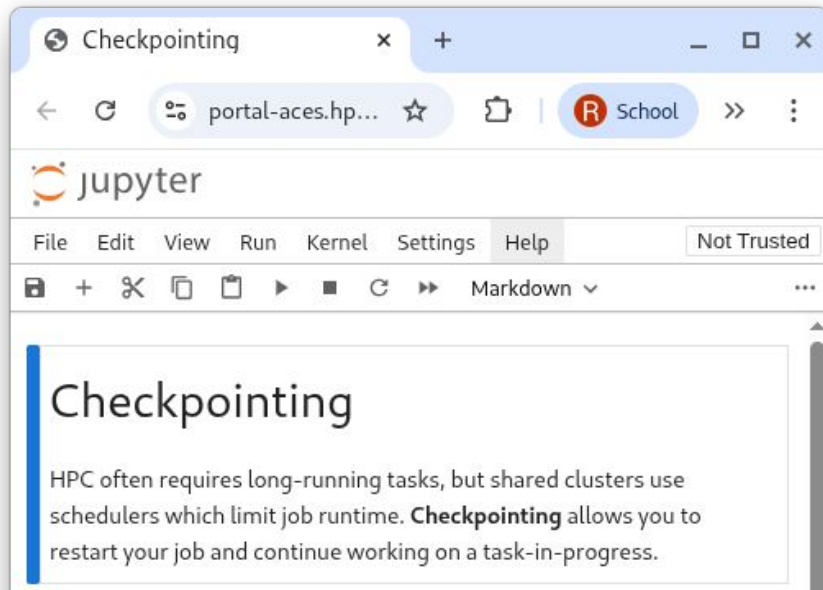
Checkpointing Exercise

- Use any environment



A screenshot of a JupyterLab interface. The browser tab is titled "Home" and the address bar shows "portal-aces.hp...". The JupyterLab logo and a menu bar (File, View, Settings, Help) are visible. Below the menu bar, there are tabs for "Files" and "Running". The "Files" tab is active, showing a file browser for the path "/ training-parent / source /". A table lists the files in the directory:

Name	Last Modified	File Size
training-venv	22 minutes ago	
✓ Checkpointing.ipynb	22 minutes ago	6.1 KB
Environment_test.ipynb	22 minutes ago	1.6 KB



A screenshot of a JupyterLab interface showing a notebook titled "Checkpointing". The browser tab is titled "Checkpointing" and the address bar shows "portal-aces.hp...". The JupyterLab logo and a menu bar (File, Edit, View, Run, Kernel, Settings, Help) are visible. A "Not Trusted" warning is present in the top right corner. Below the menu bar, there are icons for file operations and a "Markdown" dropdown menu. The main content area displays the title "Checkpointing" and the following text:

HPC often requires long-running tasks, but shared clusters use schedulers which limit job runtime. **Checkpointing** allows you to restart your job and continue working on a task-in-progress.

Acknowledgements

This work was supported by

- the National Science Foundation (NSF), award numbers:
 - 2112356 - ACES - Accelerating Computing for Emerging Sciences
 - 1925764 - SWEETER - SouthWest Expertise in Expanding, Training, Education and Research
 - 2019129 - FASTER - Fostering Accelerated Scientific Transformations, Education, and Research
- Staff and students at Texas A&M High-Performance Research Computing
- ACCESS CCEP pilot program, Tier-II

Need Help?

First check the [FAQ](#)

- [Knowledge Base](#)
- Send us a ticket using the dashboard tab in the ACES portal
- Email further questions to help@hprc.tamu.edu

Help us help you -- when you contact us, tell us:

- Which cluster you're using
- Your username
- Job id(s) if any
- Location of your jobfile, input/output files
- Application used, if any
- Module(s) loaded, if any
- Error messages
- Steps you have taken, so we can reproduce the problem



High Performance
Research Computing
DIVISION OF RESEARCH

<https://hprc.tamu.edu>

HPRC Helpdesk:

help@hprc.tamu.edu

Phone: 979-845-0219

Take our short course survey!



HPRC Survey

https://u.tamu.edu/hprc_shortcourse_survey

Thank you! Any Questions?



Lunch until 1:30 pm (central)

