Managing the software environment for a classroom deployment of OOD

Karen Tomko

ktomko@osc.edu

Contributions from: Trey Dockendorf, Jeff Ohrstrom, Shameema Oottikkal, Summer Wang, Zhi Qiang You and many others

Agenda

- Introduction to OSC Classroom Support
- How we do it
  - Process and documentation
  - Classroom OOD Instance
  - Software Environment Management
  - Homework Submission

An endorsement

Phil Thomas, Kent State University (KSU), recently presented these slides as part of a talk at and Education Elevated (e2) held at KSU.

I Don’t GET paid by OSC, No Free Trips To Vegas for me. (Not even a OSC Tee-shirt)

But I am an OSC Campus Champion

FOR

Kent State University

Phil

KENT STATE UNIVERSITY

What OSC is not!

- Scary
- Hard for students/faculty to use
- Hard to Develop Classroom Environment
- Expensive
- Hard to get help
- Hard to Administer
- Not a Windows Environment

KENT STATE UNIVERSITY

Why considering using OSC is in the Classroom Space.

- Easy to connect to from anywhere
- Easy To use On-Demand access for students and faculty
- Scalable to accommodate 100’s of students
- Large Software Selection - Most Free
- Other Software Can Be installed.
- Remote and Local License Server options
- Highly Customizable classroom Environment
- Great technical Support and Consulting
- Gets students familiar with high end computing
- Removes costs of huge expensive Lab Environments
- *Free*

KENT STATE UNIVERSITY

Background on OSC Classroom Support

OSC has supported classroom use of our systems for more than 20 years

Some Examples:

- Traditional command line access for a dual level HPC course
- Computational Chemistry for graduate students
- Webmo for an introductory chemistry course
- An introductory Statistics course for life science majors (Fall 2019)
- In 2020 – virtual labs for a wide range of courses to support remote learning
- In 2023 – GenAI (Stable Diffusion) for Digital Art and Technology students at Ohio U.

OHIO HIGHER EDUCATION COURSES

30 Organizations | 79 Departments | 209 Courses | 6,750 Enrollees

Organizations	Dept	Courses	Enrollees
Air Force Institute of Technology	1	2	4
Bluffton University	1	1	6
Bowling Green State University	1	7	84
Case Western Reserve University	1	2	32
Cedarville University	1	1	2
Cleveland State University	3	10	106
College of Wooster	1	1	8
Denison University	2	3	51
Heidelberg University	1	1	1
Kent State University	5	17	389
Kenyon College	1	2	32
Marietta College	1	5	31
Miami University	4	7	402
Mount Vernon Nazarene University	2	2	22
Muskingum University	1	1	7

Organizations	Dept	Courses	Enrollees
Ohio Dominican University	1	3	24
Ohio Northern University	1	1	8
Ohio State University	29	105	4,982
Ohio University	2	2	13
Otterbein University	1	1	13
University of Akron	1	1	2
University of Cincinnati	5	10	163
University of Dayton	1	2	38
University of Mount Union	2	3	13
University of Toledo	2	3	70
Wayne State University	1	1	3
Wittenberg University	1	1	11
Wright State University	2	3	36
Xavier University	2	2	18
Youngstown State University	2	9	179

Characteristics of classroom Use

- Most of the students are inexperienced in HPC and Linux
- Generally, lightweight/small computing requirements
- May have synchronous use during classroom/lab times
- Varying class sizes, but some are large
- Specific software requirements for the class
- Instructors may want to share files with the class
- Homework grading with consistent software environment

Characteristics of classroom Use

- Most of the students are inexperienced in HPC and Linux -> Simple interface
- Generally, lightweight/small computing requirements
- May have synchronous use during classroom/lab times

} Responsiveness

- Varying class sizes, but some are large -> Scalability
- Specific software requirements for the class -> Software support is critical
- Instructors may want to share files with the class
- Homework grading with consistent software environment

} Customized project setup

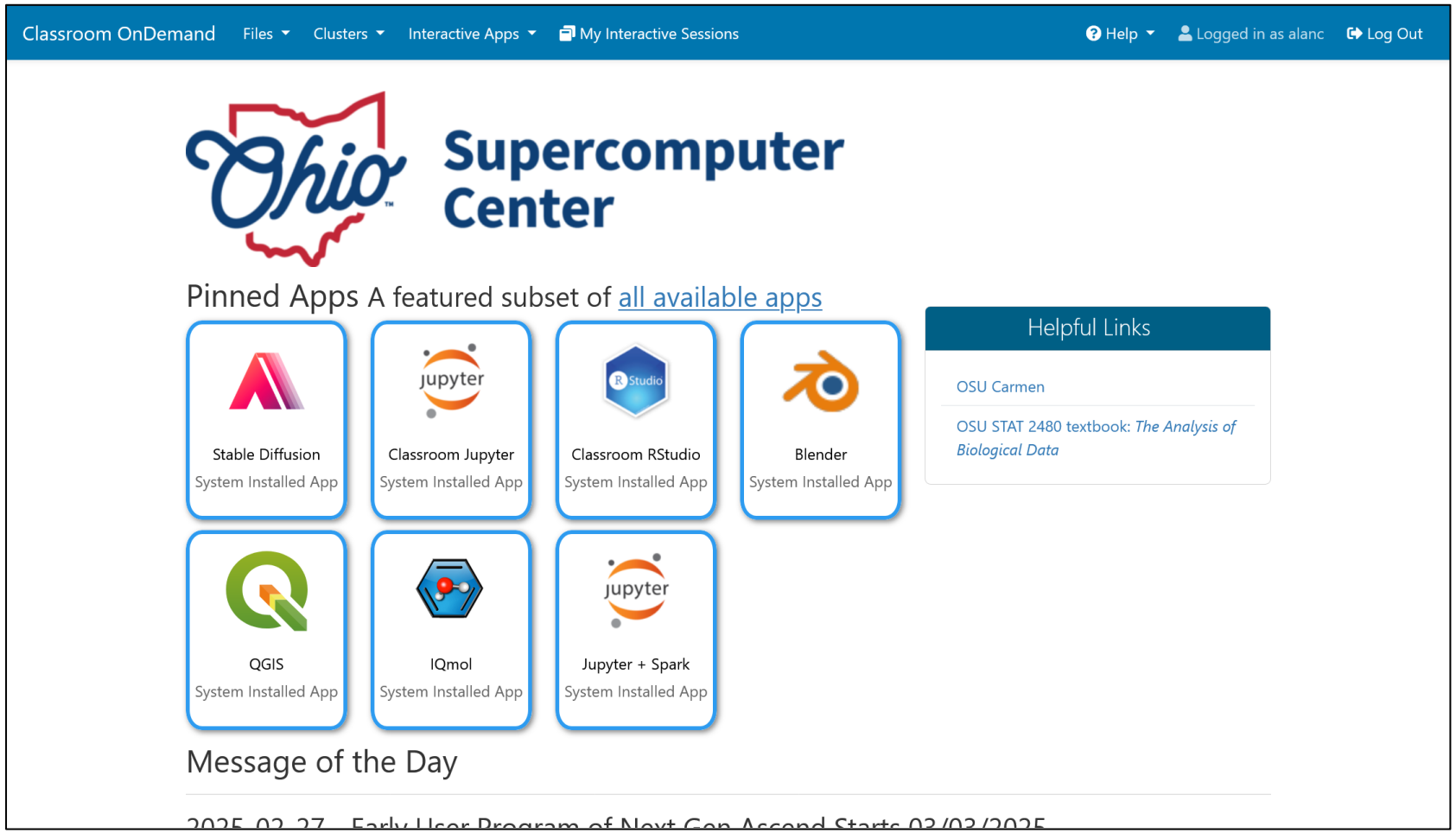
How it’s done – not just OOD

- Documentation and Process
- OOD Customization + Kubernetes
- Software Environment Management
- Homework submission support

Documentation and Process

- Client facing documentation on OSC's website
  - Classroom Project Resources Guide
  - Classroom Guide for Students
  - Using Jupyter for Classroom
  - Using Rstudio for Classroom
- Customized allocation and account setup for classrooms
  - Shared storage with specific permission settings
  - Support for student self service to add themselves to classroom
- Workflow in ServiceNow (our client support platform)

OOD Customizations



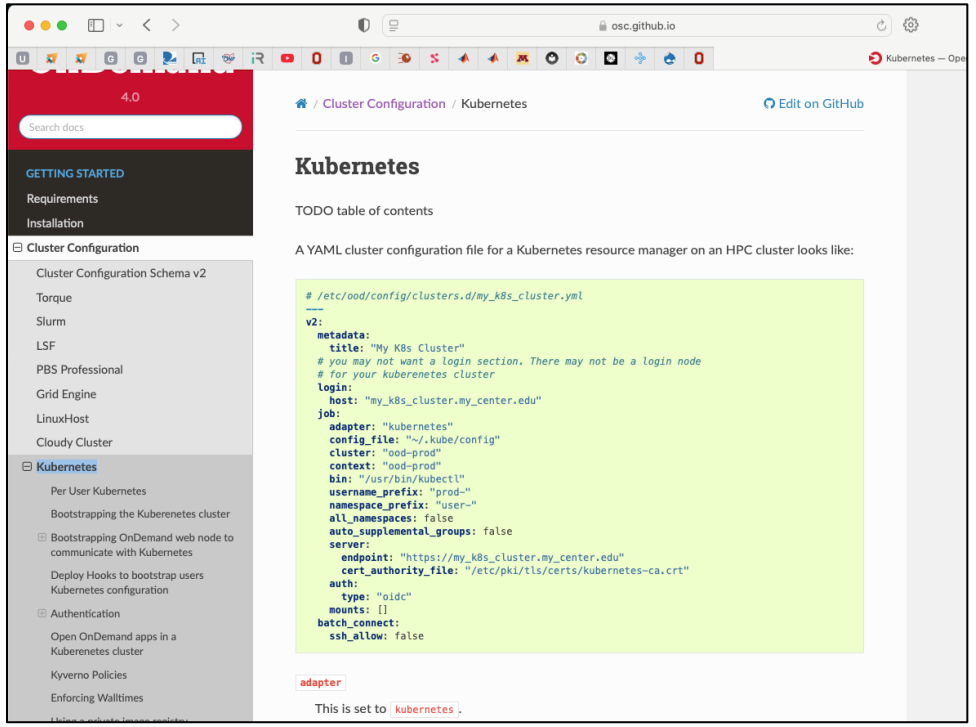
- Separate instance for classroom use
- Small set of apps
- Kubernetes scheduled compute resources
- Custom apps
- Class specific app launch

Kubernetes and OOD at OSC

Tight integration with our HPC environment

- Access to OSC filesystems from containers including home directories, share project space, /apps volume
- Account and group information from LDAP is mapped to K8 environment to prevent privilege escalation
- Internal container registry for trusted containers
- For more information see

  T. Dockendorf, T. Baer, and D. Johnson. 2022. Early Experiences with Tight Integration of Kubernetes in an HPC Environment, PEARC '22. https://doi.org/10.1145/3491418.3535150



Information on OOD Cluster Configuration for Kubernetes: https://osc.github.io/ood-documentation/latest/installation/resource-manager/kubernetes.html#

OOD Customizations

- Custom Batch Connect Apps
  - Classroom Jupyter
  - Classroom Rstudio
- Entry for each class in cluster.d/kubernetes.yml
- Classroom app generates the app job launch form based on the contents of the .yml file
- Instructor provides inputs that get represented in the .yml, for example
  - Class project name
  - Resource requirements (i.e. Small, medium, ...) for k8 resources
  - Time requirements for lab sessions

Jupyter classroom App Example
https://github.com/OSC/bc_classroom_jupyter

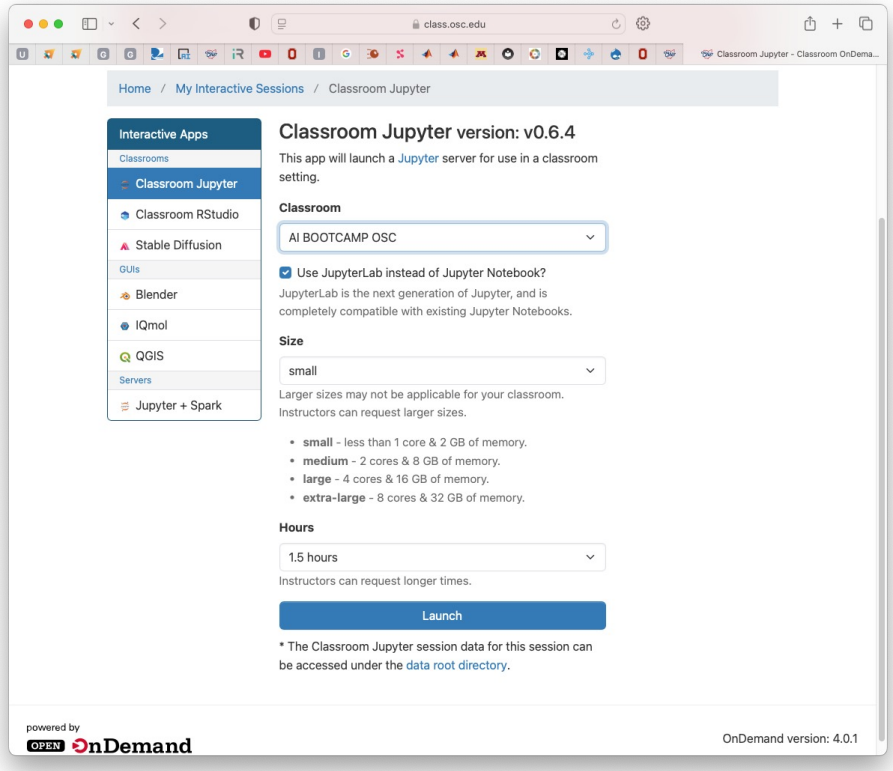
cluster.d/kubernetes.yml

https://ondemand-test.osc.edu/pun/sys/dashboard/files/fs/etc/ood/config/clusters.d/kubernetes-test.yml

docs slacks github osc discourse asana oh-tech/osu misc a11y rails tmp ACCESS demo links OOD (1) Discord | #rails-...

```
---
v2:
  metadata:
    title: Kubernetes
    hidden: true
  job:
    adapter: kubernetes
    cluster: ood-test
    bin: "/usr/local/bin/kubectl"
    username_prefix: test-
    namespace_prefix: user-
    all_namespaces: false
    auto_supplemental_groups: true
    server:
      endpoint: https://kubecontroller-test.infra.osc.edu:6443
      cert_authority_file: "/opt/osc/etc/kubernetes-ca-test.crt"
    auth:
      type: oidc
  custom:
    classrooms:
      jupyter:
        AI_BOOTCAMP_OSC:
          hours: 3
          project: PZS1124
        ASTRONOMY_8824_OSU:
          project: PAS2055
        BA_64061_KSU:
          size: large
          project: PGS0333
        BA_64061_KSU_SEC1:
          size: xlarge
          hours: 3
          project: PGS0341
        BIOL_356_DENISON:
          size: xlarge
```

Classroom Jupyter Launch form



Python Environment Challenge

Scenario: We have a class of 50 students with HPC accounts and the instructor asks them to start a Jupyter session and load a set of packages, so that they can step through some lab exercises.

What could go wrong?

Python Environment Requirements

Leads to some requirements

1. Provide a working, class-specific environment for the students
2. Isolated from other python environments that the student or instructor might be using
3. That is reproducible (session to session, student work to homework grading)

Custom Jupyter environment

Instructor View

1. Login with instructor account
2. From the command line run a single command with arguments project ID and course ID to create the folder

   <OSC_base>/project_ID/course_ID/jupyter
3. Login into class OOD instance, launch Jupyter and Install packages
4. Put shared files in materials directory
5. <OSC_base>/project_ID/course_ID/materials

Student View

1. Create account and get added to class allocation
2. Login to class.osc.edu
3. launch classroom jupyter, environment loaded based on <OSC_base>/project_ID/course_ID/jupyter
4. Jupyter workspace set to $HOME/osc_classes/course_ID
5. Files from class materials directory copied to workspace on notebook launch

Behind the scenes

- A set of scripts for creating and managing the paths that Jupyter will use as it's home instead of the default user home directory
- A module file with some extra logic to build the custom path names, set the Jupyter home/workspace, and activate the instructor created environment

Homework Submission and Grading

Requirements:

- Consistent software environment for grading
- Easy to use directory and file management
- Access to popular tools

Solutions:

- Custom OSC scripts and user docs
  - Instructor: create submit directory
  - Student: copy files to the submit directory
- Instructions for installing Nbgrader for Jupyter

Other Classroom Apps at OSC

- Rstudio and custom R environment handled similar to Jupyter and Python
- Stable Diffusion deployed for a digital arts class
  - Evan Jaffe, et. al. Stable Diffusion in the Classroom: Deploying interactive GPU-enabled ML workloads with Open OnDemand and Kubernetes. PEARC '24. https://doi.org/10.1145/3626203.3670526
- Maya and Vray applications for architecture / digital arts

THANK YOU

OSC.EDU

Department of Higher Education

Ohio Supercomputer Center

Ohio Supercomputer Center