Ohio Supercomputer Center

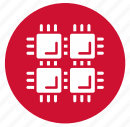
An OH·TECH Consortium Member

Deploying and Managing an OnDemand Instance

Doug Johnson, Trey Dockendorf

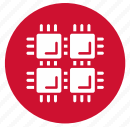
Ohio Supercomputer Center

This work is supported by the National Science Foundation of the United States under the award NSF SI2-SSE-1534949.



Outline

- OnDemand introduction and architectural overview
- Installation and configuration
- Considerations for production
- Build environment
- Questions/discussion and resources

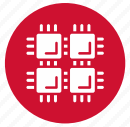


Open OnDemand Overview

Provides an easy to install and use, web-based access to supercomputers, resulting in intuitive, innovative support for interactive supercomputing.

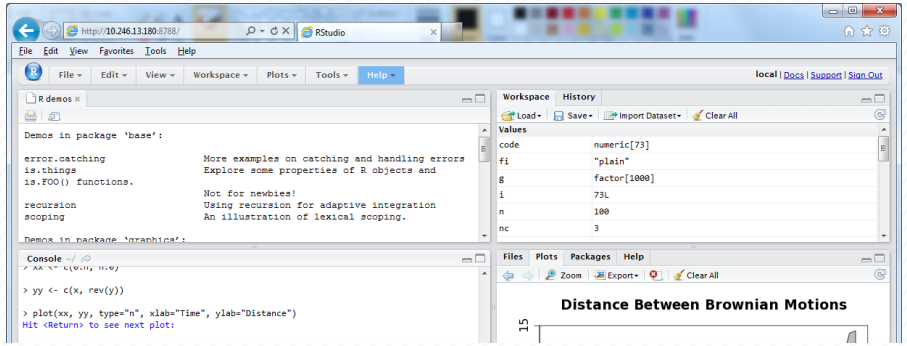
Features include:

- Plugin-free web experience for access to HPC resources
- Easy file management
- Command-line shell access
- Job management and monitoring
- Graphical desktop environments and applications

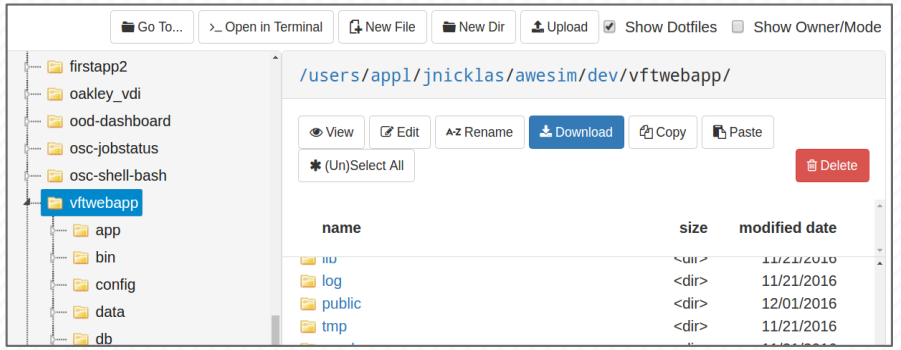


Interactive Apps and GUIs & Cluster Access

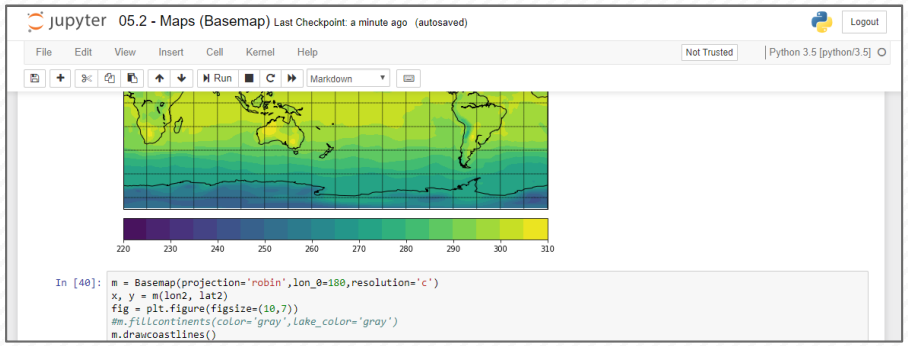
RStudio Server – R IDE



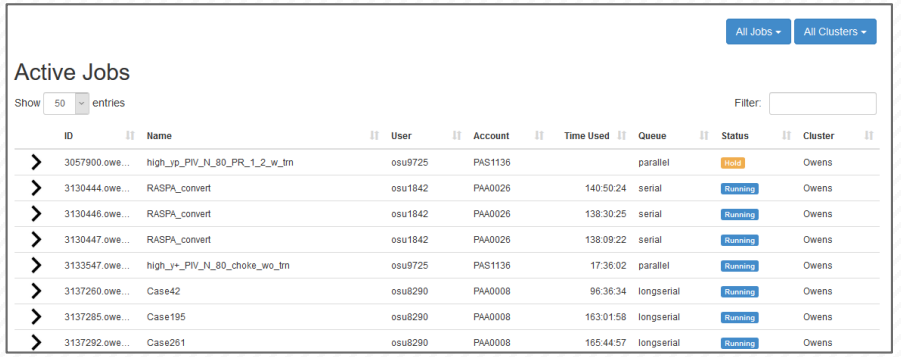
File Access (browse, edit, etc)



Jupyter Notebook – Python IDE



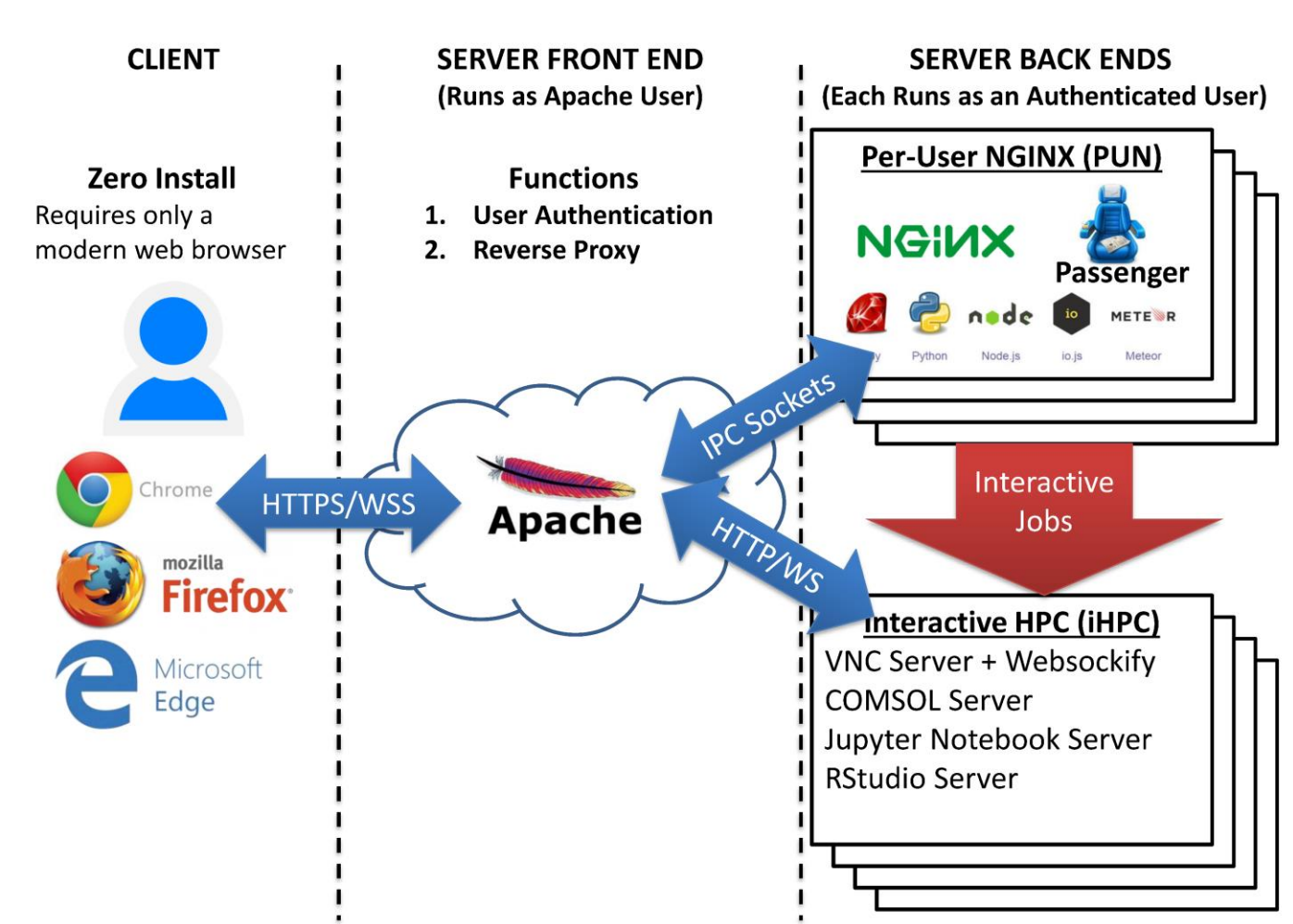
Manage Jobs (view, submit, etc)



And many more, such as ANSYS Workbench, Abaqus/CAE, MATLAB, Paraview, COMSOL Multiphysics, VNC desktops and GUIs

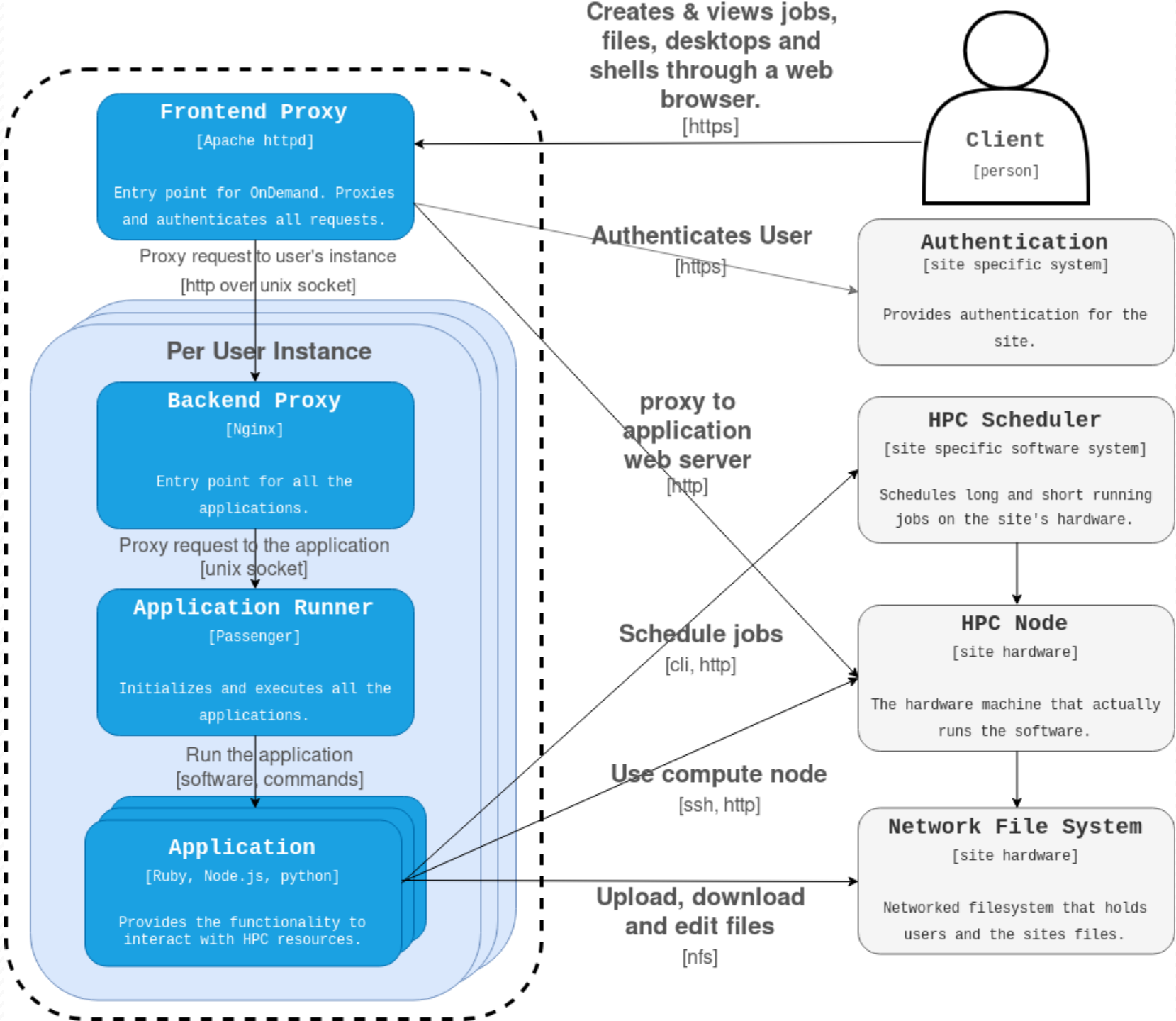
And many more, such as in-browser SSH terminal, job constructors

Architecture at a High Level



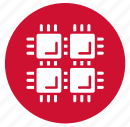
Visualizing the software architecture

- PUN is running as the authenticated user, performs operations as that user
  - Submit jobs
  - Operate on files
  - Phusion Passenger for web application execution
  - Can support user developed web applications securely
- PUNs are started via scripts executed through `sudo`
- Reverse proxy completes secure HTTP communication over SSL to HPC compute resources
  - Replaces the need for user managed SSH tunnels



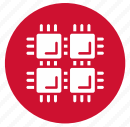
Installation and configuration: System Requirements

- **Hardware**
  - OSC – VMware VM w/ 16 vCPU & 64GB RAM (~120 concurrent users)
- **Operating System**
  - Red Hat based OS – version 6 or 7
- **Batch client software**
  - Needs to be able to submit and query jobs (sbatch, squeue, qsub, qstat, etc)
- **Shared filesystems**
  - Home directories and other cluster filesystems
  - OSC uses NFSv4 home and GPFS over NFS using CES
- **An OnDemand server is more similar to a login node than a standard web server**



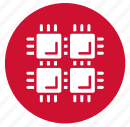
Installation and configuration:

- **YUM repos and RPMs**
  - Enable EPEL and Software Collections (SCL) repos
  - Install OnDemand YUM repo then RPM package
    - We provide Passenger and NGINX as SCL versions were deprecated
- **Configuration files**
  - YAML file per cluster – used to define login and batch environment
  - YAML file to generate Apache configuration
    - All configuration files under `/etc/ood`
- **Services**
  - Manage PUNs via nginx_stage
  - httpd24-httpd
- **Puppet module**
  - Yumrepo -> Package -> File -> Service



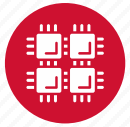
Installation and configuration: Compute Environment

- **Dependencies for Interactive Apps (ie. Desktop)**
  - TurboVNC, websockify, nmap-ncat
    - Modules or installed to system – RPMs available
  - Desktop environment (XFCE, MATE, etc)
- **Modules environment**
  - Lmod or TCL, something to load modules
- **Singularity**
  - Mostly for RStudio Server
  - More apps using Singularity in the future
- **Firewall**
  - Reverse proxy for OnDemand needs to connect to HPC resources



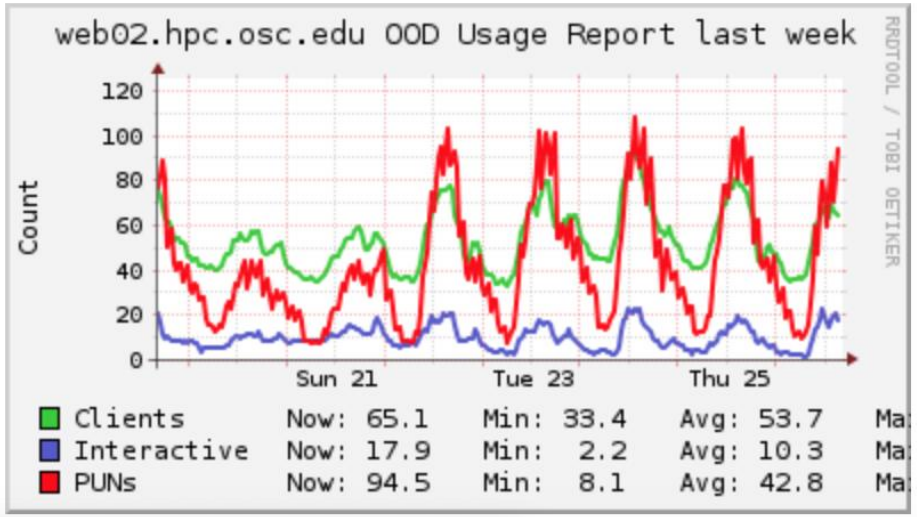
Installation and configuration: Interactive application considerations

- **Ensure short wait times for Interactive apps**
  - OSC uses “quick” batch environment with dedicated nodes
    - Moab tuned for 3s scheduling interval
  - SLURM sites could use dedicated partition or high priority QoS
  - Another possible solution is over subscription
  - Management of interactive access to HPC resources is not a solved problem
- **Must be enabled in Apache**
  - Interactive reverse proxy Apache configs not enabled by default



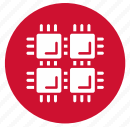
Considerations for Production:

- **Upgrades – “yum update ondemand”**
  - OnDemand Apache changes outside YAML will be lost but old config backed up
  - PUN cleanup and restarts can be slow if lots of logged in users
  - Any steps needed outside RPM update will be in Release Notes
  - In-service upgrades possible
- **Separate environments: dev -> test -> production**
  - dev: Developers have elevated privileges, soon will pull RPMs from CI/CD
  - test: A mirror of production but receives all updates first
- **Backups**
  - Configuration files - /etc/ood
  - Custom apps - /var/www/ood
  - Apache configs - /opt/rh/httpd24/root/etc/httpd/conf.d
- **Monitoring**
  - https://github.com/OSC/ondemand-metrics
  - Ganglia – nginx_stage + procfs + Apache status
- **App Development**
  - Enable per-user or for everyone
  - Code run on OnDemand web node as user, becomes login node



Build Environment

- **ondemand-packaging**
  - https://github.com/OSC/ondemand-packaging
  - Builds RPMs for ondemand and apps using docker + mock
  - Intended to facilitate custom app packaging
- **CI/CD**
  - Build RPM for every tag (includes pre-release tags)
  - Github to Gitlab CI/CD utilizing ondemand-packaging
- **Images for testing**
  - https://github.com/OSC/ood-images
  - Docker, Vagrant, Vagrant w/ SLURM, VMware images



Questions/discussion and Resources

- Open OnDemand User Group at PEARC19
  - **July 30, 5:15-6:15pm, room “Crystal C”**
- Visit our website
  - http://openondemand.org
- Use our Discourse
  - https://discourse.osc.edu/c/open-ondemand
- Join our mailing list
  - https://lists.osu.edu/mailman/listinfo/ood-users
- Our webinars are planned roughly quarterly
  - Let us know what you’d like to learn about next