

A photograph of a modern, multi-story building with a large glass facade and a brick section on the left. The building is identified by a vertical sign as the Massachusetts Green High Performance Computing Center (MGHPCC). The glass reflects the sky and surrounding environment. In the foreground, there is a paved area and some landscaping.

MGHPCC

MIT Kavli Institute Research Computing Cluster at Massachusetts Green High Performance Computing Center

Paul Hsi, January 2022

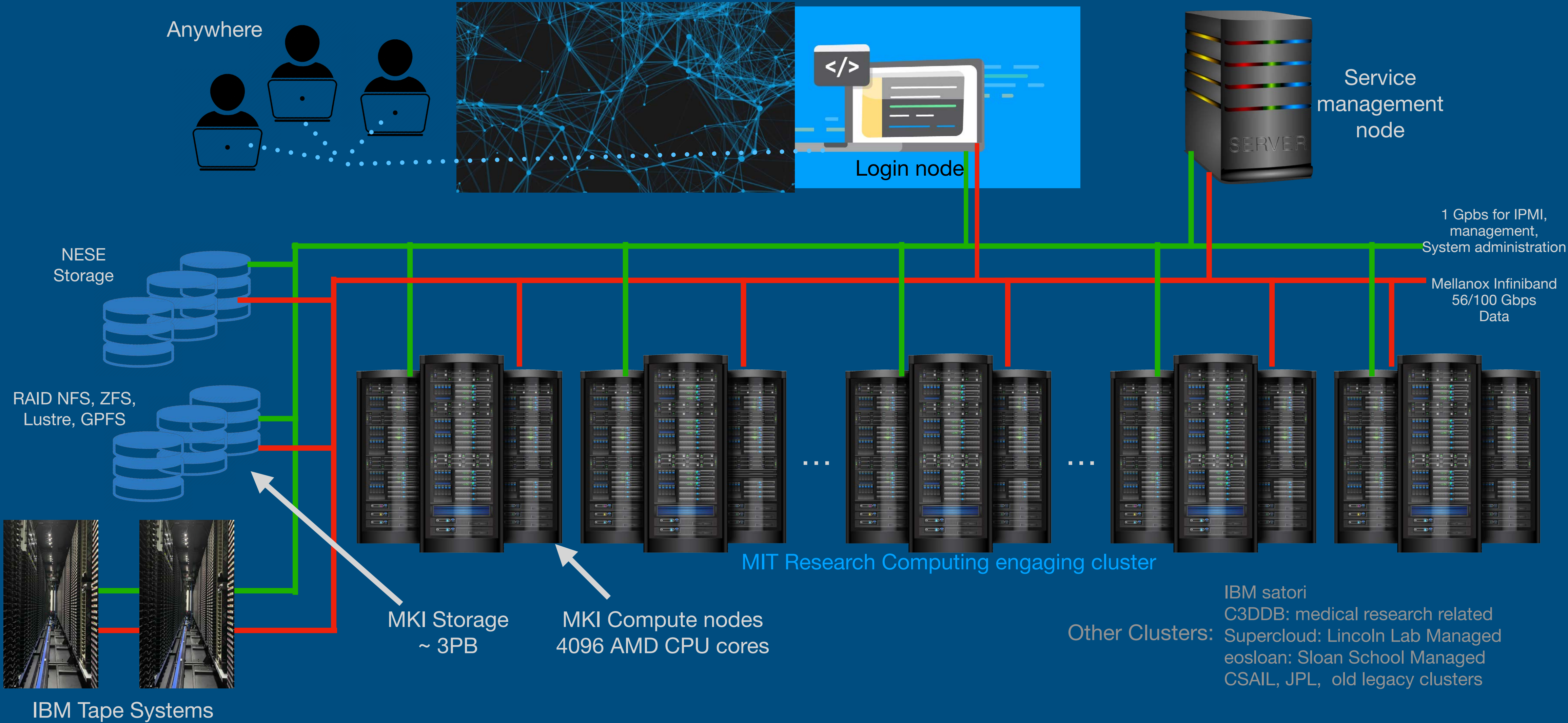






Massachusetts Green High Performance Computing Center (MGHPCC)

UMass, Northeastern, Harvard, BU, MIT, and New England Regional HPC

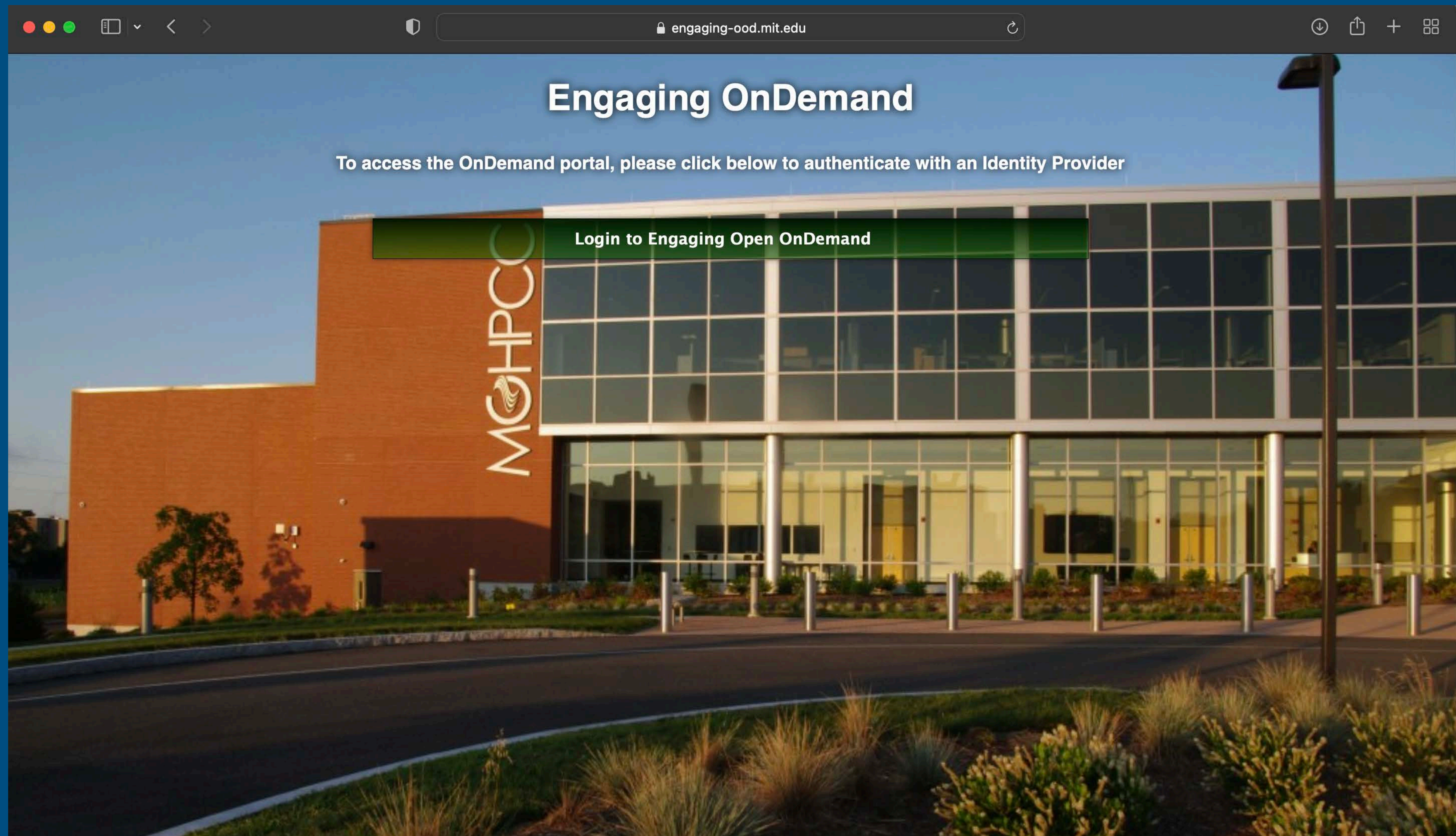


MIT Kavli Institute HPC at MGHPCC

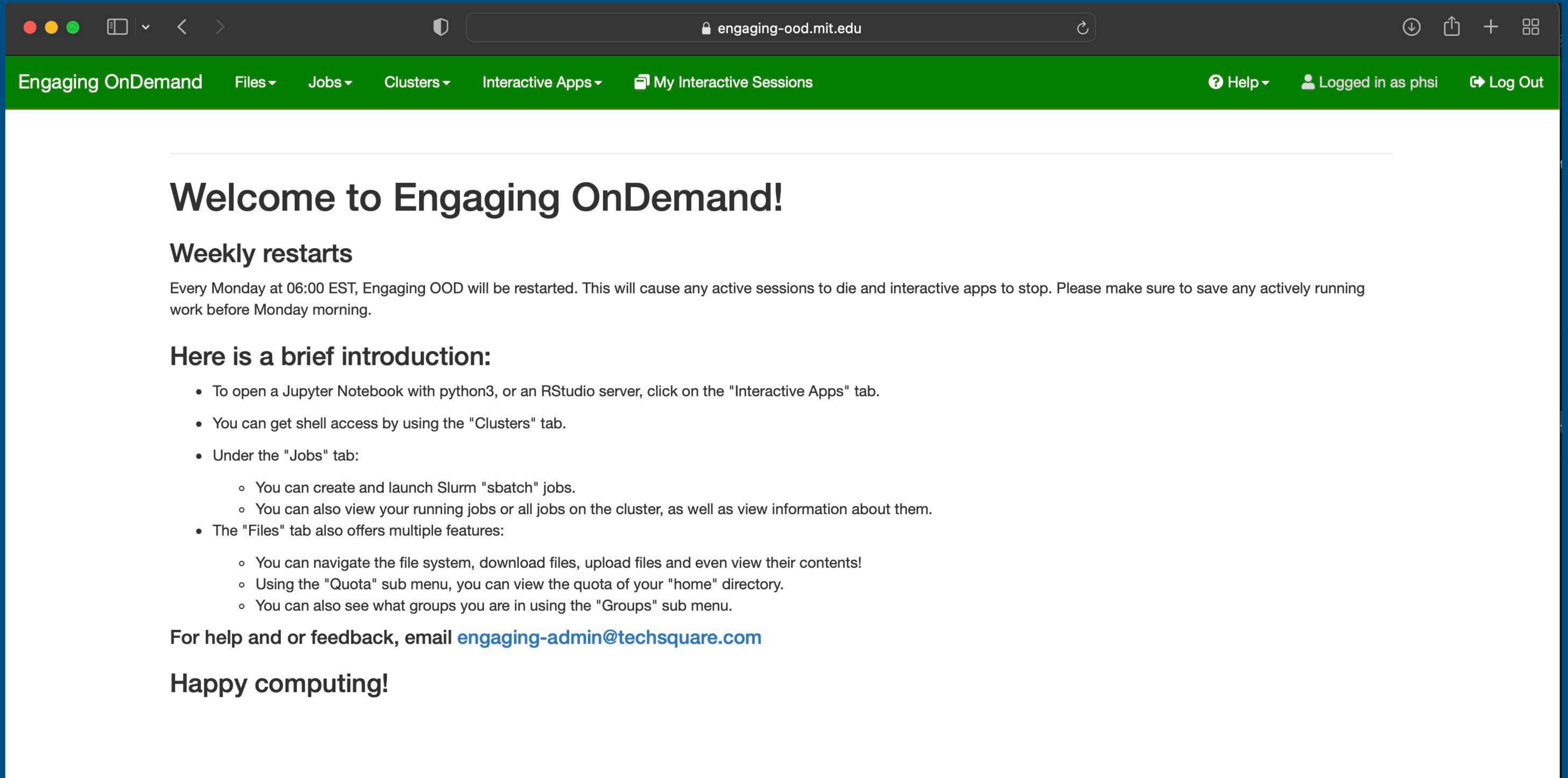
Hardware Specification

- Compute nodes (64 nodes in 16 Chassis, Total 4096 CPU cores)
DELL PowerEdge C6400-R/C6525
AMD7542 2.90GHz, 32C/64T 128M Cache, 225W
384 GB 320MT/s RDIMMs – 6GB/core
On board management port and ethernet port for administration
Mellanox ConnectX-6 single port HDR100 QSFP56 Infiniband adapter
- Storage
Mixed servers from DELL, Seagate and Thinkmate
Hardware RAID-6
NFS mount
Current total storage is about 3PB (include a new a bit over 1PB storage on the way)
- Additional storage and GPU Servers are in the plan
Another over 1PB storage and several GPU nodes may come later this year
- MIT's new rack space may come in the Summer.

How to sign up an account



After you sign in



The screenshot shows a web browser window with the URL `engaging-ood.mit.edu`. The browser's address bar and tabs are visible at the top. Below the browser window, a green navigation bar contains the following items: "Engaging OnDemand", "Files", "Jobs", "Clusters", "Interactive Apps", "My Interactive Sessions", "Help", "Logged in as phsi", and "Log Out". The main content area of the page has a white background and features a large heading "Welcome to Engaging OnDemand!". Below this heading is a section titled "Weekly restarts" with a paragraph explaining that the system restarts every Monday at 06:00 EST, affecting active sessions and interactive apps. This is followed by a section titled "Here is a brief introduction:" which contains a bulleted list of instructions for using the platform's features. At the bottom of the page, there is a line of text providing contact information for help and feedback, and a final line saying "Happy computing!".

Engaging OnDemand Files Jobs Clusters Interactive Apps My Interactive Sessions Help Logged in as phsi Log Out

Welcome to Engaging OnDemand!

Weekly restarts

Every Monday at 06:00 EST, Engaging OOD will be restarted. This will cause any active sessions to die and interactive apps to stop. Please make sure to save any actively running work before Monday morning.

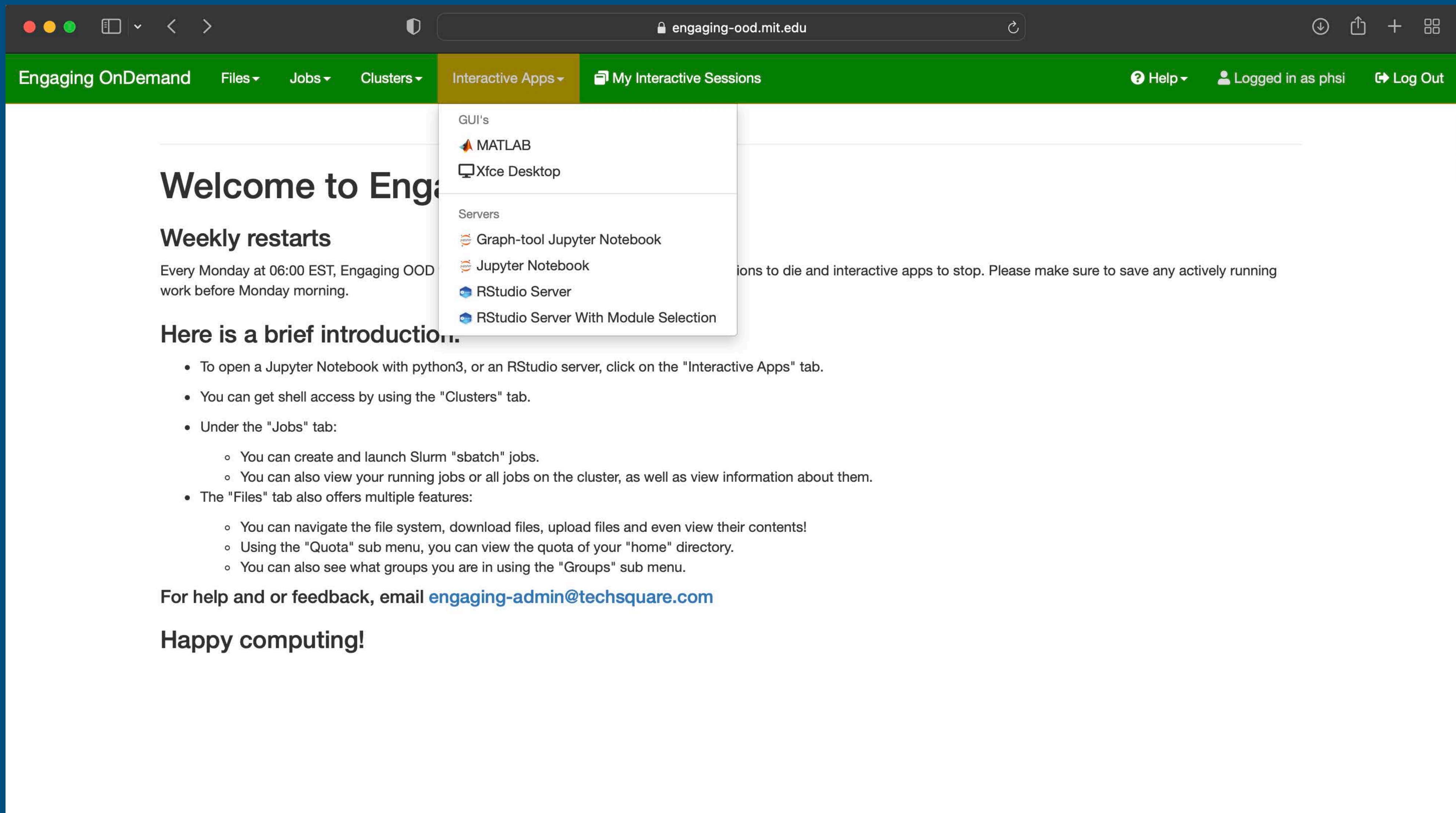
Here is a brief introduction:

- To open a Jupyter Notebook with python3, or an RStudio server, click on the "Interactive Apps" tab.
- You can get shell access by using the "Clusters" tab.
- Under the "Jobs" tab:
 - You can create and launch Slurm "sbatch" jobs.
 - You can also view your running jobs or all jobs on the cluster, as well as view information about them.
- The "Files" tab also offers multiple features:
 - You can navigate the file system, download files, upload files and even view their contents!
 - Using the "Quota" sub menu, you can view the quota of your "home" directory.
 - You can also see what groups you are in using the "Groups" sub menu.

For help and or feedback, email engaging-admin@techsquare.com

Happy computing!

Some interactive apps available



The screenshot shows a web browser window at engaging-ood.mit.edu. The navigation bar includes links for Engaging OnDemand, Files, Jobs, Clusters, Interactive Apps (selected), and My Interactive Sessions. The user is logged in as 'phsi' and can log out. The main content area displays a welcome message, a weekly restart schedule (Every Monday at 06:00 EST), and a brief introduction to the platform's features. A dropdown menu for 'Interactive Apps' is open, listing GUIs (MATLAB, Xfce Desktop) and Servers (Graph-tool Jupyter Notebook, Jupyter Notebook, RStudio Server, RStudio Server With Module Selection).

Welcome to Engaging OnDemand

Weekly restarts
Every Monday at 06:00 EST, Engaging OOD restarts all sessions to die and interactive apps to stop. Please make sure to save any actively running work before Monday morning.

Here is a brief introduction...

- To open a Jupyter Notebook with python3, or an RStudio server, click on the "Interactive Apps" tab.
- You can get shell access by using the "Clusters" tab.
- Under the "Jobs" tab:
 - You can create and launch Slurm "sbatch" jobs.
 - You can also view your running jobs or all jobs on the cluster, as well as view information about them.
- The "Files" tab also offers multiple features:
 - You can navigate the file system, download files, upload files and even view their contents!
 - Using the "Quota" sub menu, you can view the quota of your "home" directory.
 - You can also see what groups you are in using the "Groups" sub menu.

For help and or feedback, email engaging-admin@techsquare.com

Happy computing!

How to find out what modules loaded and available to you

engaging-ood.mit.edu

Dashboard - Engaging OnDemand

phsi@eofe7:~

```
[phsi@eofe7 ~]$ module list
Currently Loaded Modulefiles:
  1) gcc/6.2.0(default)  2) slurm/14.11
[phsi@eofe7 ~]$ module avail
```

/home/software/gcc/6.2.0/pkg/modulefiles

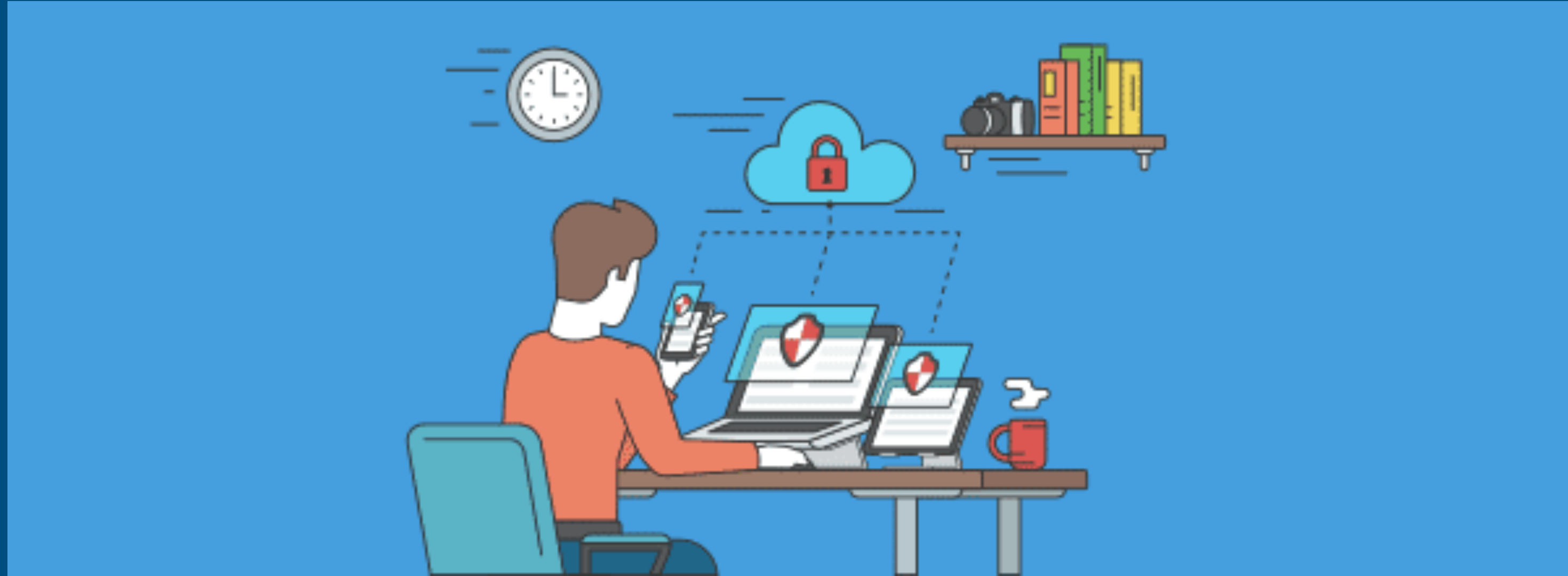
boost/1.69.0	fftw/3.3.8	hdf5/1.10.5	netcdf/4.6.3	openblas/0.3.6	openmpi/3.0.0	openmpi/4.0.4
campari/v3_09052017	foo/1.0	lapack/3.8.0	openblas/0.3.3	openmpi/2.1.1	openmpi/3.0.4	scalapack/2.0.2

/home/software/modulefiles

abaqus/2017	cudnn/6.0	gsl/2.5	mosek/8	python/3.6.3	soci/4.0.0
abaqus/2018	cudnn/7	guile/3.0.4	mosek/9	python/3.8.3	spades/3.14.1
ambertools/20	cudnn/7.5.1	gurobi/8.0.1	mosek/9.2	python/3.9.4	spark/2.4.5-hadoop2.7
anaconda2/4.5.12	cudnn/8.2.1_cuda11.3	gurobi/8.1.1	mpfr/3.1.4	qe/6.5	specfem3d/1.0
anaconda3/2019.10	cudnn/8.2.2_cuda10.2	hdf5/1.10.5-cxx	mpich/3.0.4	QuantumEspresso/6.5	spherical/02192c
anaconda3/2020.11	distro/20170205	hdf5/1.10.5-parallel	multinest/3.10	R/3.3.1	sqlite3/3.22.0
anaconda3/4.4.0	dmtcp/2.5.2	help2man/1.46.9	muscle/3.8.425	R/3.3.3	star/2.7.3a
ant/1.10.8	eigen3/3.3.7	hh-suite/20180314	namd/2.13b2	R/3.4.2	SuiteSparse/5.4.0_gcc6.2.0
argtable/2.13	fastqc/0.11.9	intel/2017-01	namd/2.13-cuda	R/3.5.3	tclx/8.4.0
arrow/0.16.0	ffmpeg/3.3.3	intel/2018-01	namd/2.13-mpi	R/3.6.2	tecplot/360ex_2019r1
automake/1.16.2	firefox/93	intel/2020-04	namd/2.14	R/3.6.3	tecplot/chorus_2010r1
bdw-gc/8.0.4	fltk/1.3.5	interpretableai/license	netcdf/4.6.3-parallel	R/4.0.5	texlive/2019
beagle-lib/20171106	foo/1.0	iqtree/1.6.3	nwchem/6.5	R/4.1.0	tmux/2.8
beast/1.8.4	freetype/2.10.2	isl/0.15	nwchem/6.8	raplcap/0.1.1	udunits2/2.2.26
bedtools/2.29.2	gaussian/09.b01	jdk/1.8.0_121	openbabel/2.4.1	rclone/1.50.2	unistring/0.9.10
blast/2.12.0	gaussian/16.c01	jre/1.8.0_301	OpenBLAS/0.2.0	rsoft/2019_09	valgrind/3.16.1
boost/1.58.0	gaussian/16.c01_avx2	jsoncpp/0.10.7	opencascade/7.4.0	rsoft/2020_09	vasp/5.4.4
boost/1.70.0	gcc/11.2.0	jsoncpp/1.9.3	openFOAM/5.x	rsoft/2021_09	vasp/5.4.4.gpu
cairo/1.15.12	gcc/4.8.5	julia/0.6.0	orca/4.1.1	rstudio/1.2.5033	vasp/ase/5.4.4
cbiomes/addXML/1.0	gcc/4.9.4	julia/0.6.2	orca/4.1.2-stack	rstudio/1.3.959	vasp/ase/5.4.4_test
CentOS6-compat/1.0	gcc/5.4.0	julia/1.1.0	parallel/20180722	ruby/2.5.3	vasp/chihang/5.4.4
clustal_omega/1.2.4	gcc/6.2.0(default)	julia/1.2.0	paraview/5.7	schrodinger/2016-4	vasp/chihang/5.4.4-w90
cmake/3.17.3	gcc/6.3.0	julia/1.3.1	paraview/5.9	scotch/6.0.9	vasp/liju/5.4.4
cmake/3.9.6	gcc/8.3.0	julia/1.4.0	paraview/5.9_headless_server	scr/1.1	vasp/shaohorn/5.4.4
comsol/52	gcc/9.3.0	julia/1.4.2	pathway-tools/24.5	silvaco/tcad/2017.00	vim/8.2
comsol/52a	gdal/2.4.2	julia/1.5.2	pcr2/10.36	silvaco/tcad/2019.00	visit/2.12.13
comsol/53	genesis/1.5	julia/1.6.5	pdsh/2.33	simmobility/20191126	wannier90/1.2
comsol/53-psfc	geos/3.5.2	kcachegrind/0.7.4	polymake/3.1	singularity/2.3.1	wannier90/3.0.0
comsol/55-ase	gerris/20170609	knitro/12.0.0	poppler/0.81.0	singularity/2.3.2	wannier-tools/2019-09-05
comsol/55-fink	git-lfs/2.5.2	lammps/12Dec18	postgresql/12.3	singularity/2.4.2	wget/1.19.1
comsol/56-fink	glpk/4.65	lammps/22Aug18	postgresql/9.5.22	singularity/2.5.2	wxwidgets/3.1.2
cp2k/4.1	gmesh/4.6.0	lammps/29oct20	powercap/0.1.1	singularity/3.0.2	xerces-c++/3.2.3
cplex/128	go/1.11.4	libcrococo/0.6.13	PowerFlow/2021_R2	singularity/3.2.1	xsd/4.0.0
cuda/10.0	go/1.14.4	libctl/4.5.0	PowerTherm/2021_R2	singularity/3.3.0	xtb/6.2.2
cuda/10.1	gperftools/11-6-2018	libmsr/0.3.1	proj/4.8.0	singularity/3.4.2	zlib/1.2.11
cuda/10.2	graph-tool/2.29	libsvg2-devel/2.40.20	python/2.7.13	singularity/3.5.3	
cuda/11.3	graphviz/2.47.0	libtool/2.4.6	python/2.7.14	singularity/3.6.1	
cuda/8.0	gromacs/2018.4	lua/5.3.5	python/2.7.14u4	singularity/3.7.0	
cuda/9.0	gromacs/2018.4-gpu	lyx/2.3.6.1	python/3.6.0	slicer/4.10.2	
cudnn/5.1	gromacs/2018.4-opa	molecularGSM/1.0	python/3.6.1	socat/1.7.3.4	

How to check nodes availability and jobs status

```
[phsi@eofe7 ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
sched_any_quicktest* up       15:00      1 drain* node142
sched_any_quicktest* up       15:00      3 mix node[028,101,279]
sched_any_quicktest* up       15:00      2 idle node[029,145]
sched_any up       12:00:00      5 drain* node[023-024,087,106,123]
sched_any up       12:00:00      1 mix node235
sched_any up       12:00:00     11 alloc node[019-022,025-026,107-108,122,161,170]
sched_mit_hill up       12:00:00     43 drain* node[001-006,017-018,023-024,033-034,036,045,055-056,059,062,064,081,087,089-090,095,099,106,109-112,118,121,123-124,128,138,141,163,167-168,186,195,385]
sched_mit_hill up       12:00:00     22 drain node[027,030-031,035,057-058,060-061,063,120,125-126,131-134,136-137,139-140,144,166]
sched_mit_hill up       12:00:00     48 mix node[032,054,069-072,100,184-185,187-192,235,360-384,386-389,394,967-968]
sched_mit_hill up       12:00:00     82 alloc node[007-016,019-022,025-026,037-044,046-053,065-068,073,082-086,088,091-094,096-098,102-105,107-108,113-117,122,130,135,143,146,154,157,159,161-162,164-165,170,181-183,193-194,196-198]
sched_mit_hill up       12:00:00     30 idle node[147-153,155-156,158,160,169,171-180,331-338]
newnodes up       12:00:00      5 drain* node[280,283,424,426-427]
newnodes up       12:00:00     11 mix node[119,129,275,282,285-286,339-342,425]
newnodes up       12:00:00      4 alloc node[262,281,284,395]
newnodes up       12:00:00     24 idle node[254-257,265-266,271-274,276-278,287-289,317-320,390-393]
sched_mit_mvogelsb up      14-00:00:0      4 alloc node[1435-1438]
sched_mit_mvogelsb up      14-00:00:0     19 idle node[1259-1260,1262-1266,1431-1434,1439-1446]
sched_mit_mki up      14-00:00:0      1 resv node1400
sched_mit_mki up      14-00:00:0     13 alloc node[1399,1404-1411,1435-1438]
sched_mit_mki up      14-00:00:0     34 idle node[1401-1403,1412-1434,1439-1446]
sched_mit_lncib up      14-00:00:0     16 idle node[1415-1430]
[phsi@eofe7 ~]$ sinfo -p sched_mit_mki
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
sched_mit_mki up      14-00:00:0      1 resv node1400
sched_mit_mki up      14-00:00:0     13 alloc node[1399,1404-1411,1435-1438]
sched_mit_mki up      14-00:00:0     34 idle node[1401-1403,1412-1434,1439-1446]
[phsi@eofe7 ~]$ squeue -a
JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON)
36793670 sched_sys node024. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node024)
36793705 sched_sys node095. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node095)
36793709 sched_sys node099. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node099)
36793779 sched_sys node163. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node163)
36793842 sched_sys node930. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node930)
36793844 sched_sys node932. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node932)
36793848 sched_sys node934. root PD 0:00 1 (ReqNodeNotAvail, UnavailableNodes:node934)
36687963 sched_any monodept chenandy PD 0:00 1 (PartitionTimeLimit)
38831576 sched_any submit_b dstk PD 0:00 4 (PartitionNodeLimit)
38831585 sched_any submit_b dstk PD 0:00 4 (PartitionNodeLimit)
38831590 sched_any submit_b dstk PD 0:00 8 (PartitionNodeLimit)
38831596 sched_any submit_b dstk PD 0:00 4 (PartitionNodeLimit)
38831597 sched_any submit_b dstk PD 0:00 8 (PartitionNodeLimit)
39309234 newnodes, run.slur dgoldebr PD 0:00 4 (PartitionTimeLimit)
39405015 sched_any testbatc lindegrd PD 0:00 1 (PartitionTimeLimit)
39600661 sched_any hessian_ congwang PD 0:00 1 (PartitionTimeLimit)
39600662 sched_any hessian_ congwang PD 0:00 1 (PartitionTimeLimit)
39600663 sched_any pca_back congwang PD 0:00 1 (PartitionTimeLimit)
40350348 sched_eng sys/dash jml1 PD 0:00 1 (Resources)
40350280 sched_eng sys/dash ifrah PD 0:00 1 (Priority)
```

Happy Computing!