#### NERSC is the Production HPC & Data Facility for DOE Office of Science Research





Office of Science

Largest funder of physical science research in U.S.



Bio Energy, Environment



Computing



Materials, Chemistry, Geophysics



Particle Physics, Astrophysics



**Nuclear Physics** 



Fusion Energy, Plasma Physics

# **The Cori System**

- Cori will transition HPC and datacentric workloads to energy efficient architectures
- Includes "data partition" with Haswell nodes, data friendly queues, pool of large memory interactive nodes
- Also has NVRAM layer, Software Defined Network



Image source: Wikipedia

System named after Gerty Cori, Biochemist and first American woman to receive the Nobel prize in science.







- Combine an annotated notebook with live code execution and results
  - "literate computing" environment, where narrative and computation go hand in hand
- Jupyter emerged from iPython and is becoming a very popular web based framework for interactive notebook computing
- Not just python has over 60 language kernels
   Julia, R, Bash, Root, Ruby ...
- c/f Fernando Perez keynote

### **Notebook examples**



#### https://github.com/materialsproject/mapidoc/blob/master/example\_note books/pymatgen\_examples/Analyze%20and%20plot%20band%20structures .ipynb

In [9]: #is the material a metal (i.e., the fermi level cross a band) print bs.is metal() #print information on the band gap print bs.get\_band\_gap() #print the energy of the 20th band and 10th kpoint print bs.bands[Spin.up][20][10] #print energy of direct band gap print bs.get\_direct\_band\_gap() #print information on the vbm print bs.get\_vbm() False {'energy': 1.797800000000005, 'transition': u'(0.591,0.409,0.000)-\\Gamma', 'direct': False} 18.0201 2.6904 {u'kpoint': <pymatgen.electronic structure.bandstructure.Kpoint object at 0x5192610>, u'kpoint ind ex': [123], u'energy': 6.1023, u'band\_index': {1: [15]}, u'projections': {}} Here, we plot the bs object. By default for an insulator we have en energy limit of cbm+4eV and vbm-4 eV In [7]: %matplotlib inline from pymatgen.electronic\_structure.plotter import BSPlotter plotter=BSPlotter(bs) plotter.get\_plot().show() 4  $E_{f} \left( eV \right)$ 0 Ē  $B_1|\mathbf{B}$  $\mathbf{F}P_1$ ZILP Z XIQ Wave Vector





# HPC workflows at scale for data-intensive science => asynchronous, batch execution.

But ...

# Scientific insight => interactive, iterative exploration and analysis

#### How can Jupyter bridge this gap?

### **Motivation**



- Python is the most popular language/tool at NERSC
- Users want to be able to use Jupyter notebooks to drive their jobs, run post-job analyses etc.
- Access to NERSC resources through these interfaces
  - Filesystems
  - Batch Queue
  - Network, DBs
- Support for custom libraries and kernels
- Centralized service to deploy notebooks in a standard authenticated manner

#### Jupyterhub

- Service to deploy notebooks in a multi-user environment
- Manages user authentication, notebook deployment and web proxies
- Node.js + Python Tornado





### **Jupyterhub Example**



✓ Jupyter Hub
 ✓ Jupyter Hub
 ✓ C
 Attps://ipython.nersc.gov/hub/login
 Apps
 M Gmail: Email from G...
 C + Add to Delicious

¶☆ 🜔 🦷 💆 🕃 😨 🝵

Sign in							
Username:							
shreyas							
Password:							
••••••							
Sign							

Files Running

Clusters

#### Select items to perform actions on them.

#### **But Wait .. There's More**

- Jupyter also includes •
  - File Browser
  - Text Editor (with full syntax highlighting)
  - Terminal

ĊJ	upyter potcar_setup.py - 09/05/2014
File	e Edit View Language
1	#!/alobal/ul/s/shrevas/dev/testcif/bin/pvthon2.7
2	······································
3	import os
4	import glob
5	import shutil
6	import subprocess
7	
8	pspdir = ""
9	
10	count = 0
11	while not os.path.exists(pspdir):
12	if count i= 0:
14	princ( invalid vasp dif: )
15	"etc. subdirs are present. If you obtained the DSPs "
16	"directly from VASP, this should typically be the "
17	"directory that you untar the files to : ")
18	print
19	•
20	targetdir = raw_input("Please enter the fullpath of the where you want to "
21	"create your pymatgen resources directory: ")
22	print
23	
24	os.makedirs(targetdir)
25	<pre>print("Generating pymatgen resources directory")</pre>
26	
27	for (parent, subdirs, files) in os.walk(pspdir):
28	for subdir in subdirs:
29	filen(filenerges) > 0:
31	If $fen(fiftehames) > 0$ : bacadir = os path join(targetdir os path bacaname(parent))
32	if not os nath evis(hasodir).
33	os.makedirs(basedir)
34	fname = filenames[0]
35	<pre>dest = os.path.join(basedir, os.path.basename(fname))</pre>

A / global / homes / s / shreyas / dev
۵.
bdcclient-2.2.1
bdctest
C Composetest
data_docker
□ □ db_scripts
G gridmap-verify
🗆 🗀 Idap
□ □ osg
csg-pki-tools.orig
C testcif
C tmpvenv
Untitled.ipynb
Signal Stress St
osg-pki-tools.tar.gz

#### 💭 jupyter

				_	_				
bash-4.1\$ ls -l tmp									
total 16416									
drwxrwxr-x 2	shreyas	shreyas	512	Jan	29	2016	build		
drwxr-xr-x 2	shreyas	shreyas	512	Oct	7	19:13	conf		
drwxr-xr-x 3	shreyas	shreyas	4096	Oct	7	19:14	conf.d		
drwxrwxr-x 4	shreyas	shreyas	4096	May	9	2013	dist.eugridpma.info		
drwxrwxr-x 5	shreyas	shreyas	512	Oct	21	16:51	foo		
drwxrwxr-x 5	shreyas	shreyas	512	Oct	21	23:01	foobar		
-rw 1	shreyas	shreyas	16384	Apr	30	2015	globus-user-env.sh.swp		
drwxrwxr-x 5	shreyas	shreyas	512	Oct	14	22:30	msq		
drwxrwxr-x 5	shreyas	shreyas	512	Jun	3	22:25	bwa		
drwxrwxr-x 6	shreyas	shreyas	512	Jan	29	2016	pydap-3.1.1		
drwxrwxr-x 5	shreyas	shreyas	512	Jan	29	2016	pydap-3.2		
drwxr-x 18	shreyas	shreyas	4096	May	26	2015	Python-2.7.10		
-rw-r 1	shreyas	shreyas	16768806	May	23	2015	Python-2.7.10.tgz		
drwxr-x 5	shreyas	shreyas	512	May	26	2015	req		
bash-4.1\$									



- Jupyterhub on a single node
- Service runs as root and spins up notebooks locally
- Jupyter has access to global filesystems and lives within the NERSC network

# **Jupyterhub in a Single Container**







- Users would like access to big compute jobs!!!
- Access to Cori Scratch Filesystem
- Existing Software on Cori (including kernels)
- Take advantage of large memory interactive nodes
- Security considerations
  - Don't want to run as root
  - Don't want to run public facing web service inside a Cori node
- New service uses a distributed model where Jupyterhub runs on a science gateway node but spins up notebooks remotely

# **New Jupyterhub Implementation For Cori**



Nersc

YEARS

at the OREFRONT

- 14 -



Jupyter architecture is completely pluggable

So we developed

- Custom Authenticator that allows user to login and get a GSI X509 certificate
- Custom Spawner that creates a remote notebook via SSH (using public key or GSI auth)
- Ipython Magic Commands to launch and query jobs via SLURM



- User logs in with username and password. Authenticator uses myproxy to login to NERSC CA server with username/password and retrieves credentials (X509 certificate)
- Jupyterhub runs as a standalone service and doesn't need root access. In fact, no root access needed across this architecture.
- <u>https://github.com/NERSC/gsiauthenticator</u>



- We wrote an SSH Spawner that will will SSH into the Cori node with users credential
  - Supports GSISSH (use with certificates from GSI authenticator)
  - Supports SSH key based auth
- SSH Spawner starts up notebook server process and goes away; Notebook server communicates directly with hub
  - No tunnels or persistent connections needed
- Keep track of the PID for poll and shutdown functions (also via SSH)
- Inspired by Andrea Zonca's RemoteSpawner (SDSC)
- <u>https://github.com/NERSC/SSHSpawner</u>

## **Ipython Magic for SLURM**



- Created some simple magics to facilitate interaction with SLURM batch queuing system
- Implemented %%sbatch and %squeue
- Pandas mode to slurp the results
- https://github.com/NERSC/slurm-magic

```
In [2]: %%sbatch -p debug -t 10 -N 1
#!/bin/bash
srun -n 32 hostname
. . . :
Out[2]: u'Submitted batch job 2754280\n'
                                                      In [3]: %squeue -u rthomas
                                                      Out[3]:
In [3]: !cat slurm-2754280.out
                                                           JOBID
                                                                      USER ACCOUNT NAME PARTITION
                                                                                                          NODES TIME LIMIT TIME ST \
                                                                                                      QOS
nid00044
                                                                                                               1
                                                      0 2764292 rthomas
                                                                             mpccc
                                                                                     sh
                                                                                            debug
                                                                                                   debug
                                                                                                                      10:00 0:11 R
nid00044
                                                         PRIORITY
                                                                            SUBMIT TIME
                                                                                                  START TIME
. . .
                                                            69060 2016-07-21T21:12:34 2016-07-21T21:13:21
                                                      0
```

#### **Other Features**

- Allow users to browse other filesystems while defaulting to homedir
- Users can drop in custom kernels in their .ipython dir





#### Use Case: OpenMSI + Jupytherhub

#### openmsi.nersc.gov https://jupyter.nersc.gov/user/oruebel/notel Untitled11 Last Checkpoint: Last Monday at 11:07 PM (autosaved) Control Panel Logout Flax\_Pod\_12\_day\_old\_CS.h5, Entry#0, Dataset# 0 Store Settings: ON 🛓 View Insert Cell Kernel Help Python 2 O Launch, run, Raw Peak Cube 100 999.98999 💠 🖌 😽 🍽 Baw Peak Cube Cell Toolbar: None B + ≫ (2) B + ↓ ▶ ■ C Code . ON Channel 2: Channel 3: CN Point #1: 97, 101 and integrate / 459.9959 ا د 639.9939 sx[: 2.000 plt.imshow(d[:,:,sx[1]]) 1,800 # put a red dot, size 40, at 2 locations. 1,600 plt.scatter(c[:100], r[:100], c='r', s=40) 1,400 notebooks 1,200 plt.show() 1,000 -800 -600 -400 200 -0 100 400 500 Point #2: 194, 202 900 800 -700 -600 -500 -120 400-300 -200 In [4]: #Execute the analysis #cockaw comsi\_cx(nameKey='CX on Raw Data') # cockaw comsi\_cx(nameKey='CX on Raw Data') # cockaw.execute( msidata=d , rank=6, objectiveDim=omsi\_cx.dimension\_index['imageDim'] ) In [ ]: #Save the analysis to file # e.create\_analysis( ocx ) # f.close\_file() © 2012-2014 Lawrence Berkeley National Lab - Privacy & Terms - Contact Manage data, analyses, Share analysis users and resources results Customize templates Template analysis notebook repository User notebook repository GitHub This repository Search Explore Features Enterprise Blog GitHub This repository Search Sign in benbowen / OpenMSI\_Factorization\_Examples ⊕ Watch 1 ★ Star 0 ¥ Fork 0 benbowen / OpenMSI\_Factorization\_Examples @ Watch 1 ★ Star 0 ¥ Fork 0 ¥ branch: master ŷ branch: master -:= 除 II (A 0 $\diamond$ OpenMSI Factorization Examples / MSI Factorizations Using CUR and CX.ipvnb OpenMSI Factorization Examples / MSI Factorizations Using CUR and CX.ipvnb Contribute new benbowen on Sep 24, 2014 added not ven on Sep 24, 2014 added no 7459 lines (7459 sloc) 939.163 kB analysis templates 7459 lines (7459 sloc) 939.163 kB History 🗃 🧪 🍵 Raw Blame History 🖉 In [1]: Spylab inline In [1]: spylab inline %config InlineBackend.figure\_format = 'svg' onfig InlineBackend.figure\_format = 'svg' Populating the interactive namespace from numpy and matplotlik Populating the interactive namespace from numpy and matplotlib import necessary packages import necessary packages In [2]: from pylab import \* import matplotlib.pyplot as plt from mpl\_toolkis.axes\_grid1 import ImageGrid import time sys.path.append('/Users/bpb/Data/programming/C to your openmsi-th Accetion In [2]: from pylab import \* import matplotlib.pyplot as plt from mpl\_toolits.ascs\_gridl import ImagoGrid import time sys.path.append('/Umers/bpb/Data/programming/s sys.path.append('/Umers/bpb/Data/programming/s) ramming/OpenMSI/CodeForgeRepo/opennsi-tk/') #add path nsi-tk locatio from omsi.dataformat.omsi\_file import \* from omsi.analysis.multivariate\_stats.omsi cx import from omsi.dataformat.omsi\_file import \* from omsi.analysis.multivariate\_stats.omsi\_cx import We'll use the openmsi file api to set a pointer to the peak cube. Currently We'll use the openmsi file api to set a pointer to the peak cube. Currently it is stored in "d" it is stored in "d' In [3]: #Load the input data inputFile = '/Osers/bpb/20120711 Brain (2).h5' In [3]: #Load the input data inputFile = '/Users/bpb/20120711 Brain (2).h5'

#### jupyter.nersc.gov



- Driving a Spark/Dask job from a notebook on Cori
- Spawning notebook kernels on Cori compute nodes
- Tighter interaction with MPI jobs
- Interactive HPC workflows (LDRD)



# "I'll never have to leave a notebook again

that's like the ultimate dream"

### We're Hiring



- Postdoc to work on Jupyter for human-in-the-loop interactive HPC workflows
- Various CSE positions
- http://jobs.lbl.gov

**Contact: scholia@lbl.gov** 

#### - 24 -

#### More info

#### **Contact us:**

- <u>scholia@lbl.gov</u>
- scanon@lbl.gov
- <u>rcthomas@lbl.gov</u>

All our code is on github https://github.com/NERSC/gsiauthenticator https://github.com/NERSC/SSHSpawner https://github.com/NERSC/slurm-magic

