ATLAS

Cloud Native Deployments of Bare-Metal High-Performance AI Workflows

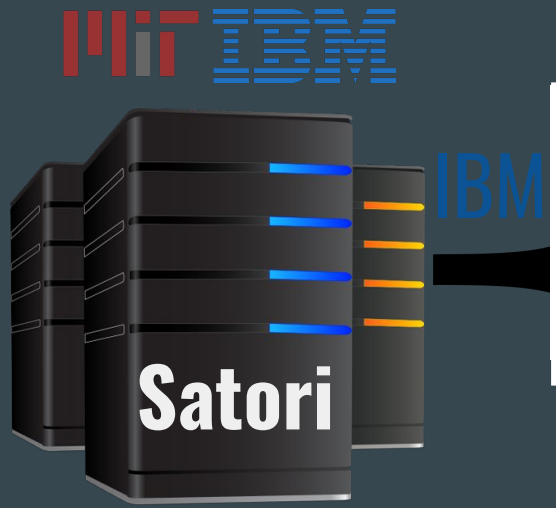
George | Hao | Jing | Shawn | Shubham

MENTOR Chris | Mike | Ravi

Outline

- Quick Recap
- Architecture
- Last Sprint Progress
- Current Sprint & Burndown
- Performance Metrics
- OpenShift - BuildConfig
- Deploy CPU-only Image on OpenShift
- Reflection *(What we learned and what we didn't do well so far)*

Quick Recap



Students

Brainstorming energy-saving hacks on Satori, MIT's new supercomputer

Three-day hackathon explores methods for making artificial intelligence faster and more sustainable.

Kim Martineau | MIT Quest for Intelligence
February 11, 2020

Press Inquiries

RELATED

Admins

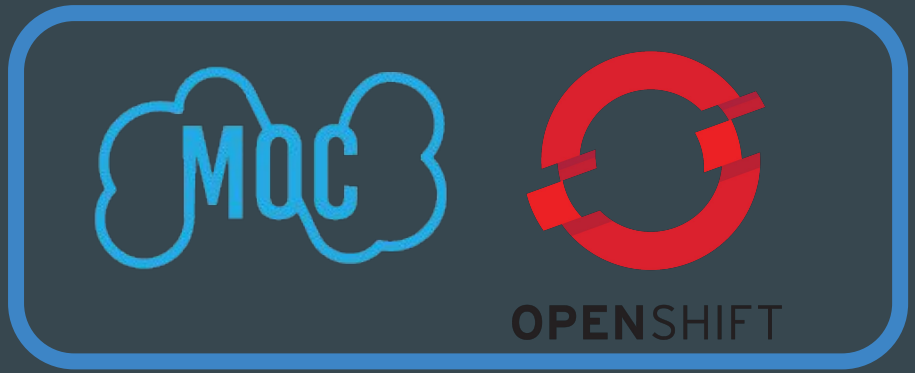
Power9 cpu

NVidia V100 32GB

High-speed network

Admins & Researchers

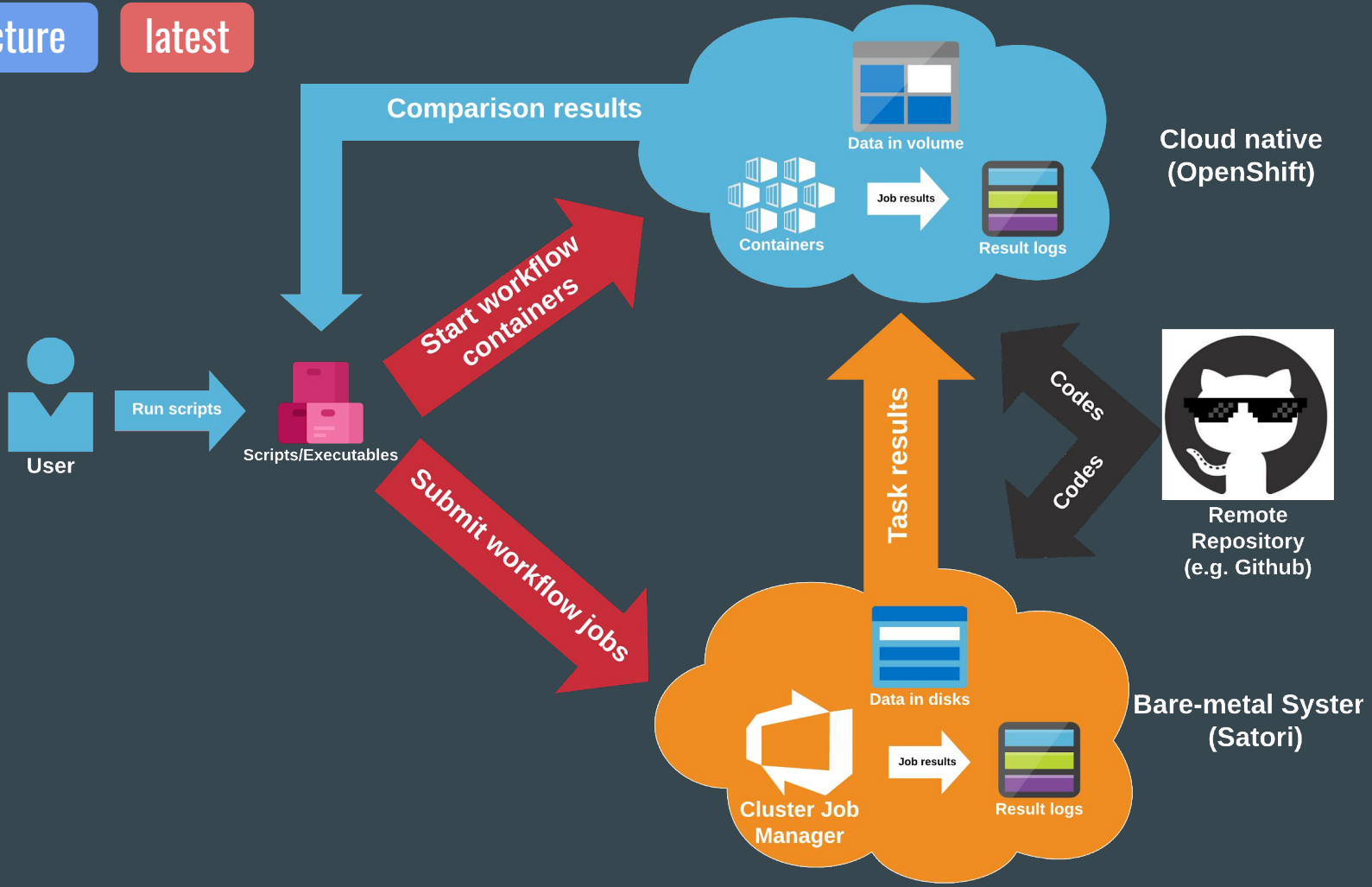
Bare-metal env.



Cloud env.

Architecture latest

Architecture latest



Last Sprint Progress

Last Sprint Progress

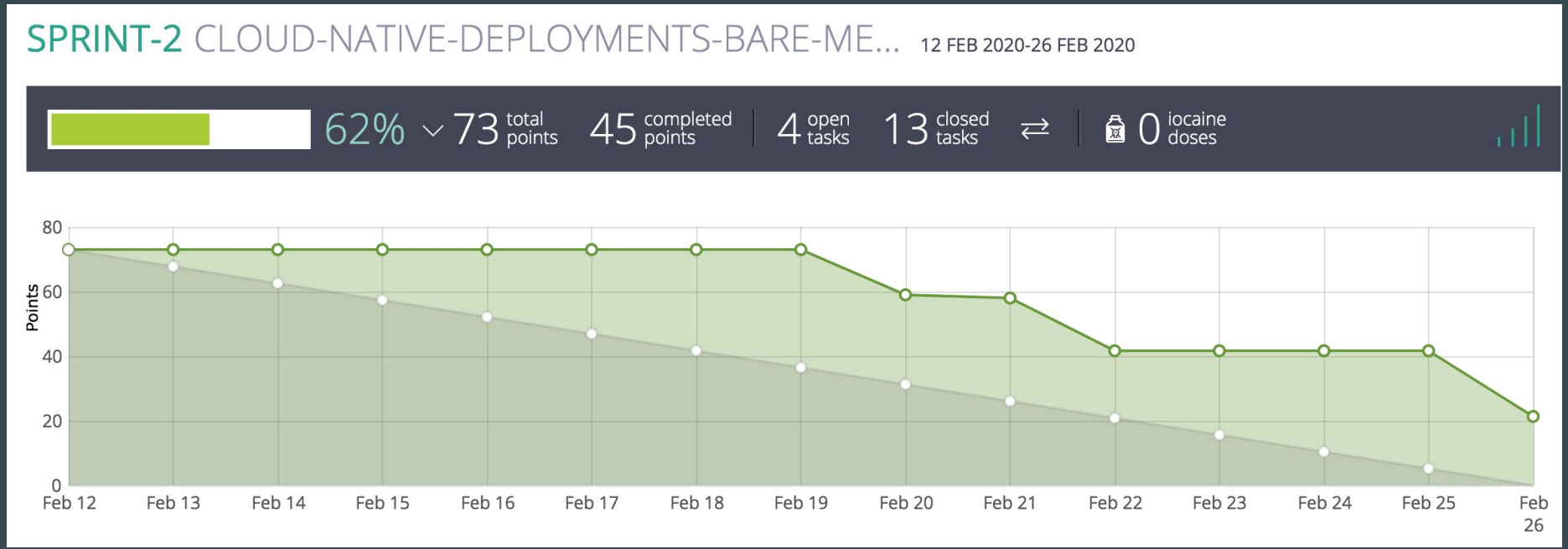
- Environment setup and logistics
  - X86-based OpenShift cluster
  - Blocked PowerPC-based OpenShift cluster
- Successfully ran the workflow in bare-metal env (Satori).
- Successfully containerized the Big-GAN workflow in local env.
  - Generated Dockerfile
  - Cpu-only, non-gpu

Current Sprint & Burndown

Sprint (2/12 ~ 2/26)

- Survey Performance metrics
- Implement script to measure Performance on Satori (In progress)
- Get familiar with OpenShift
- Run BigGAN with CPU on OpenShift
- Run BigGAN with GPU on OpenShift Were blocked until a week ago

Burndown Chart



Performance Metric

Performance Metrics

- CPU usage
- GPU usage
- Cost
- Time to completion

Fig: Comparison between bare metal and cloud

Performance Metrics: How do we measure?

- Linux utilities like ‘top’ give real time CPU stats, ‘PBench’ on OpenShift
- nvidia-smi - NVIDIA System Management Interface program (GPU stats)
- Script these utilities, to check usage periodically

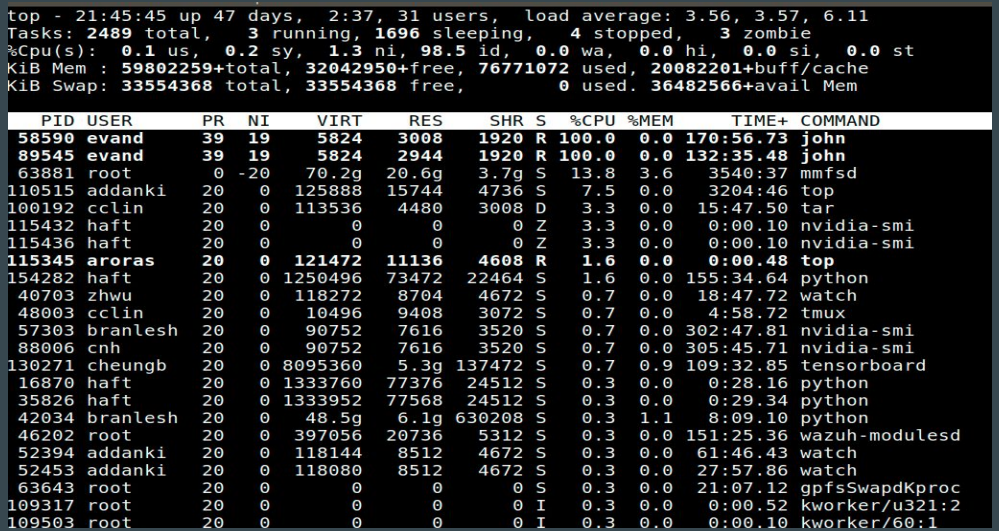


Fig: ‘top’ command to show process usage of computers resources

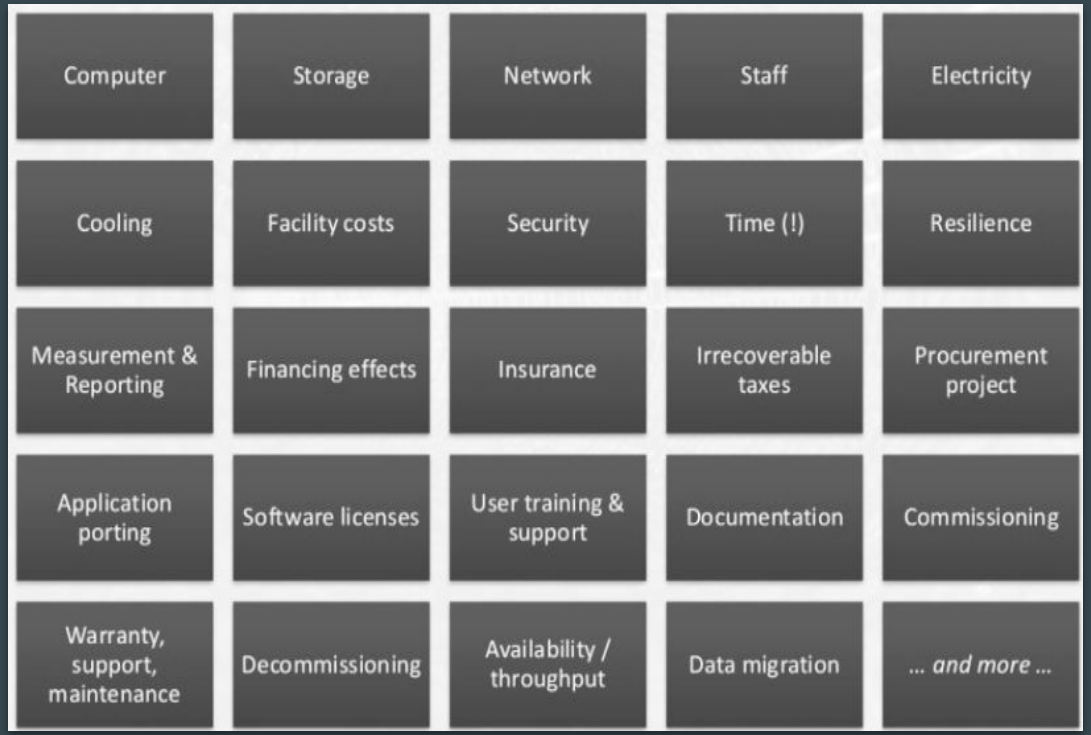


Fig: Common Costs to Measure in an HPC

- Total Cost of Ownership (TCO):
  - Costs owning/operating HPC
  - Important to HPC owners/managers
  - Always costs *something*
- Total Cost of Solution (TCS):
  - Costs incurred using HPC service for project
  - Important to HPC users
  - May cost *nothing*

```
(wmlce-1.6.2) [haozuo@service0001 ~]$ nvidia-smi
Tue Feb 25 21:39:57 2020
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 418.87.00    Driver Version: 418.87.00    CUDA Version: 10.1     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  Tesla V100-SXM2...  On   | 00000004:04:00.0 Off |                    0 |
| N/A   40C    P0    40W / 300W |     10MiB / 32480MiB |      0%   E. Process |
+-------------------------------+----------------------+----------------------+
|   1  Tesla V100-SXM2...  On   | 00000035:03:00.0 Off |                    0 |
| N/A   42C    P0    53W / 300W |  31311MiB / 32480MiB |      0%   E. Process |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|    1     42034      C   python                                   31301MiB |
+-----------------------------------------------------------------------------+
```

Requesting exclusive node

Nodes are not fully utilized

Universal baseline (MIPS, CPU, GPU)

Deploy AI workflows with BuildConfig

ON

OpenShift

WHAT IS BuildConfig

BuildConfig

AUTOMATIC

IMAGE BUILD DEPLOY UPDATE

YAML File

BuildConfig Similar Docker Compose

ONLY for OpenShift

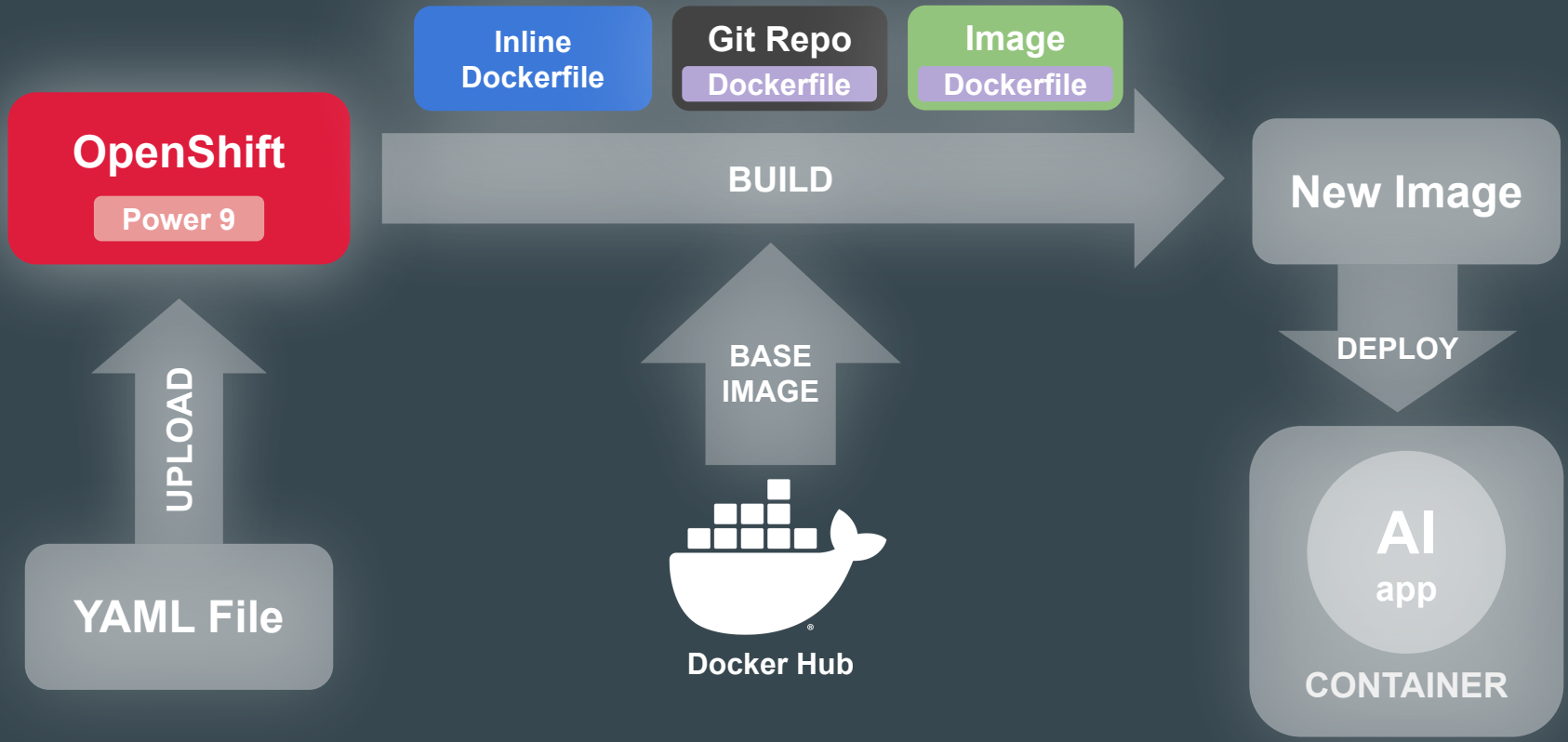
CAN DO MORE

BuildConfig

VARIOUS SOURCE SUPPORT

Normally want sources contain a **Dockerfile**

BuildConfig Workflow



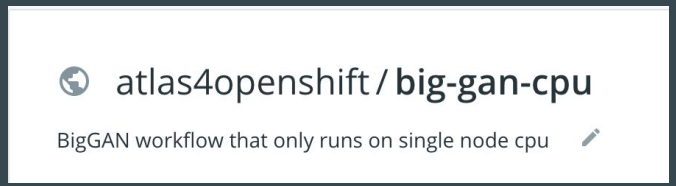
Deploy CPU-only Image on OpenShift

Deploy CPU-only Image on OpenShift

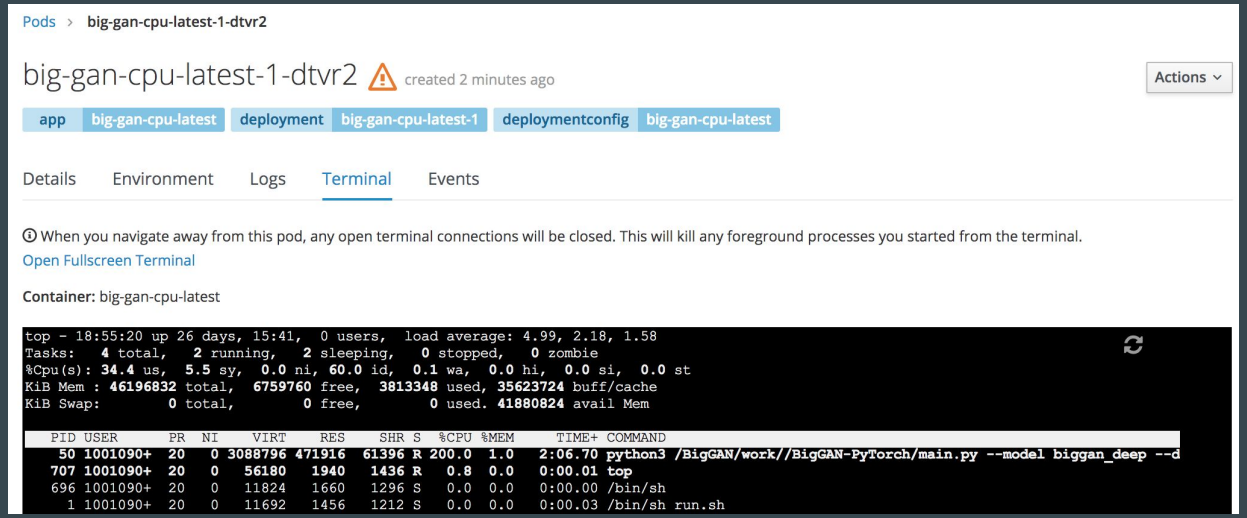
- Wrote Dockerfile that builds the image

```
FROM centos:centos7

# Install yum dependencies
RUN yum -y update; yum clean all
RUN yum -y install epel-release gcc openssl-devel bzip2-devel git python3

# Add python requirements & install
# Utilize docker cache to prevent pip installing every time
ADD ./requirements.txt /python/requirements.txt
RUN pip3 install --upgrade pip && pip install -r /python/requirements.txt

# Add working directory
ADD ./ /BigGAN
WORKDIR /BigGAN

CMD cd /BigGAN
CMD /bin/sh run.sh
```

- Pushed image to team DockerHub



Deploy CPU-only Image on OpenShift



^ Workflow runs successfully on k-openshift (x86) platform

```
sh-4.2$ cat /proc/cpuinfo | grep name
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
model name      : Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz
```

^ Cpu info for k-openshift platform

Reflection

Reflection

- What we've learned in this sprint:
  - More about OpenShift (BuildConfig, Containerization ..etc)
  - Resource profiling tools on Linux system (pbench, top, nvidia-smi ..etc)
  - Backfill scheduling concept
- What can be improved:
  - Estimation/Overstretching (Assigned too much tasks)
  - Practice to break down task into smaller granularity
  - General cloud infrastructure knowledge
  - OpenShift documentation.... :((((

Demo (if time allows)