

Container-based AI Software Deployment and Execution on AIST AI Cloud

Yusuke Tanimura, Ryousei Takano
National Institute of AIST, Japan



AIST AI Cloud (AAIC)

- Shared computing platform for AI and IoT research
 - Satisfy computing demand of AIST users
 - Provide a collaboration work environment for AIST partners
 - Try new services and tools developed by AIST
 - Try proprietary data in AIST
- Technical features:
 - NVIDIA GPU (Tesla P100) x400
 - 4PB storage (GPFS)
 - Converged infrastructure
 - Integration of AI, BigData and HPC technologies
 - IaaS (frontend) & batch execution (backend)
 - The 3rd rank in Green 500 (July, 2017)



Complicated AI Software Stack

- Numerous combination options
 - Programming language: Python, Scala, R, Java, etc.
 - GPU programming: CUDA, OpenACC, GPU-support library
 - MPI implementation: OpenMPI, MVAPICH
 - ML/DL: TensorFlow, Chainer, Caffe, Torch, MXNet, CNTK, Theano, etc.
 - Version combination of the above
- Our admin team (small group) cannot setup and test all of the combinations.
- Users actually want to build their custom environment by using Docker.
 - Reproducibility is important. (Ex. Keep the specific versions!)
 - Don't want to wait for admin to install the software.

Container-based Approach

- The admin team provides major DL/ML software tuned for AAIIC as a docker image.
- The admin team also allows users to run their Docker images, which will be customized from the above images or built from scratch.
- The approach is implemented with very lightweight container technologies.
 - Singularity (LBNL)
 - Shifter (NERSC)



Demo with Singularity

1. Docker Hub for AAIC
2. Import a docker image
3. Run the image on AAIC
 - Interactive execution
 - Batch execution

import Docker image

- \$ ssh svc.aaic.hpcc.jp
- \$ ssh gpx002.aaic.hpcc.jp # debug node
- \$ singularity create --size 8192
tensorflow.img
- \$ singularity import tensorflow.img
docker://aistairc/aaic:tensorflow

Run TensorFlow on debug node

- \$ singularity shell --nv tensorflow.img
- > source extrapath
- > python [mnist_with_summaries.py](#) --
data_dir data --log_dir logs
- > exit
- \$ exit

Run TensorFlow via qsub

- \$ qsub [test_job.sh](#)
- \$ qstat
- \$ cat pragma-demo.o*