Access HAL-DGX and OVERDRIVE servers through HAL-LOGIN3 node

- Introduction
- Rules
- Access to hal-dgx
  - 1. Interactive
  - 2. Batch script
  - 3. Access Data and/or Result with sftp
- Access to overdrive
  - 1. Interactive
  - 2. Batch script
  - 3. Access Data and/or Result with sftp

Introduction

We have prepared a hal-login3 machine as a login node so that users can request computational resources from hal-dgx and overdrive.

- hal-dgx is a NVIDIA DGX A100 machine: https://www.nvidia.com/en-us/data-center/dgx-a100/
- overdrive is a NVIDIA Arm HPC Developer Kit machine: https://developer.nvidia.com/arm-hpc-devkit

How to login **hal-login3**

```
ssh <user_id>@hal-login3.ncsa.illinois.edu
```

Type **sinfo** to check the existing partitions

```
[dmu@hal-login3 ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
arm          up 15-00:00:0      1   idle overdrive
x86*         up 15-00:00:0      1   idle hal-dgx
```

Note: hal-login3 has no shared file system. Therefore, you can not find the same layout among these three machines.

For the users' convenience we have mounted hal-dgx:/homeprojects and hal-dgx/home on hal-login3 under /dgx/home and /dgx/projects. Be advised that this is not a high speed link, so moving GB's of data is better done with an interactive job on hal-dgx and pulling the data directly there.

Rules

1. the maximum wall time for each job is 48 hours
2. the maximum GPU one user can request is 4x.

Access to hal-dgx

You need to submit an interactive job and/or batch script to request some resources to run your jobs.

1. Interactive

Request 1x GPU along with 32x CPU cores for 4 hours

```
srun --partition=x86 --time=4:00:00 --nodes=1 --ntasks-per-node=32 --sockets-per-node=1 --cores-per-socket=16 --threads-per-core=2 --mem-per-cpu=4000 --wait=0 --export=ALL --gres=gpu:a100:1 --pty /bin/bash
```

Request 2x GPU along with 64x CPU cores for 12 hours

```
srun --partition=x86 --time=12:00:00 --nodes=1 --ntasks-per-node=64 --sockets-per-node=2 --cores-per-socket=16 --threads-per-core=2 --mem-per-cpu=4000 --wait=0 --export=ALL --gres=gpu:a100:2 --pty /bin/bash
```

Request 4x GPU along with 128x CPU cores for 24 hours

```
srun --partition=x86 --time=24:00:00 --nodes=1 --ntasks-per-node=128 --sockets-per-node=4 --cores-per-socket=16
--threads-per-core=2 --mem-per-cpu=4000 --wait=0 --export=ALL --gres=gpu:a100:4 --pty /bin/bash
```

2. Batch script

```
#!/bin/bash
#SBATCH --job-name="example"
#SBATCH --output="example.%j.%N.out"
#SBATCH --partition=x86
#SBATCH --time=1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=32
#SBATCH --sockets-per-node=1
#SBATCH --cores-per-socket=16
#SBATCH --threads-per-core=2
#SBATCH --mem-per-cpu=4000
#SBATCH --gres=gpu:a100:1
#SBATCH --export=ALL

cd ~

echo STARTING `date`

srun hostname
```

3. Access Data and/or Result with sftp

1. Log on to hal-login3, start an interactive job with 1 CPU core

```
srun --pty /bin/bash
```

2. Log on to hal-dgx, start an sftp session

```
sftp hal-dgx.ncsa.illinois.edu
```

Access to overdrive

You need to submit an interactive job and/or batch script to request some resources to run your jobs.

1. Interactive

Request 1x GPU along with 40x CPU cores for 4 hours

```
srun --partition=arm --time=4:00:00 --nodes=1 --ntasks-per-node=40 --sockets-per-node=1 --cores-per-socket=40 --
threads-per-core=1 --mem-per-cpu=3200 --wait=0 --export=ALL --gres=gpu:a100:1 --pty /bin/bash
```

Request 2x GPU along with 40x CPU cores for 4 hours

```
srun --partition=arm --time=4:00:00 --nodes=1 --ntasks-per-node=80 --sockets-per-node=1 --cores-per-socket=80 --
threads-per-core=1 --mem-per-cpu=3200 --wait=0 --export=ALL --gres=gpu:a100:2 --pty /bin/bash
```

2. Batch script

```
#!/bin/bash
#SBATCH --job-name="example"
#SBATCH --output="example.%j.%N.out"
#SBATCH --partition=arm
#SBATCH --time=1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=40
#SBATCH --sockets-per-node=1
#SBATCH --cores-per-socket=40
#SBATCH --threads-per-core=1
#SBATCH --mem-per-cpu=3200
#SBATCH --gres=gpu:a100:1
#SBATCH --export=ALL

cd ~

echo STARTING `date`

srun hostname
```

3. Access Data and/or Result with sftp

1. Log on to hal-login3, start an interactive job with 1 CPU core

```
srun --partition arm --pty /bin/bash
```

2. Log on to overdrive, start an sftp session

```
sftp overdrive.ncsa.illinois.edu
```