How to use the Training system.

Dr. Dimitris Dellis

GRNET

Athens, 25-28 Nov. 2014

System Details

- 6 8-core Xeon X5650 based compute nodes
- 2 NVIDIA Tesla GPUs per node
- 16 GB memory per node
- 4x QDR Infiniband network for MPI and for I/O to the global filesystem
- 35TB of FhGFS global storage

- You received via e-mail, your username, something like `pr14u##` and private key as attachment
- Save your private key (file `id_rsa`) in attachments somewhere in your laptop
- Connect to EUCLID using your username and private key
- If you are using openSSH :

```
ssh -i PATHTO_id_rsa USERNAME@euclid.cyi.ac.cy
```

- If you are using putty for SSH connections, convert the private key to putty format :
  `puttygen id_rsa -o private_key_putty`
- Create new session in putty with :
  `Username : Your Username`
  `Hostname : euclid.cyi.ac.cy`
- In SSH → Auth section of putty session configuration select the converted to putty format private key :
  `private_key_putty`
- If you are using something different ask support staff before HandOn sessions

- Software is accessible via environment modules
- Environment modules set for you mainly the $PATH and $LD_LIBRARY_PATH variables
- Basic modules examples
  - `module avail`. List available modules. You get a list of available modules. If a package is available in more than one version, one is marked as default (D).
  - Example
    - If you look for OpenMPI items, you'll see :
      OpenMPI/1.6.4-GCC-4.6.4
      OpenMPI/1.6.4-GCC-4.7.2
      OpenMPI/1.6.5-GCC-4.8.1
      OpenMPI/1.7.3-GCC-4.8.2
      OpenMPI/1.7.3-gcccuda-2.6.10 (D) ← Default

- module load : Load a module
- Example :
  - module load OpenMPI :
    Loads the environment described in the default OpenMPI module configuration. Equivalent to
    module load OpenMPI/1.7.3-gcccuda-2.6.10.

- module list : List already loaded modules.
- Example :
  - module list

```
[pr14u01@euclid ~]$ module list

Currently Loaded Modules:
1) GCC/4.8.2
2) CUDA/5.5.22-GCC-4.8.2
3) gcccuda/2.6.10
4) hwloc/1.8-gcccuda-2.6.10
5) OpenMPI/1.7.3-gcccuda-2.6.10
```

- Note that a module load or module switch loads/switch/unload the nesessary module dependencies. You may see different version of previously loaded module.

- module switch

  Switch versions environment for a package.
- Example
  - module switch OpenMPI OpenMPI/1.6.5-GCC-4.8.1

    Switch from whatever version is loaded for OpenMPI to 1.6.5-GCC-4.8.1 version
- module unload

  Unload whatever was set by module
- Example
  - module unload OpenMPI
  - Unsets (forget) whatever was set by the last loaded OpenMPI module

- Batch system/WorkLoad Manager in EUCLID is SLURM.
- Other, Commonly used Batch systems/Workload managers are :
  - LSF
  - Torque(PBS/Maui)
  - SGE
  - LoadLeveler

- Basic use of SLURM
- List running Jobs
- squeue

```
[pr14u01@euclid ~]$ squeue
JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
  171       cpu   runjob  pr14u01  R    1:58:18      1 e01
```

Says that Job 171 of user pr14u01 with name runjob is running on partition cpu, using 1 nodes, on node e01

- A more detailed view of running jobs

```
squeue  -o "%.6i %.9P %.6j %.7u %.2t %.7M %.5D %.4C %.9B %.19e"
 JOBID PARTITION   NAME    USER ST    TIME NODES CPUS EXEC_HOST           END_TIME
   171       cpu runjob pr14u01  R 3:33:24     1    8       e01 2014-11-22T23:04:08
```

Says that Job 171 of user pr14u01 with name runjob is running on partition cpu, using 1 nodes, 8 cores, on node e01, consumed Wall time is 3:33:24, requested wall time expires at 2014-11-22 23:04:08

- Cancel a Job

  scancel 171

- Submitting a Job
  - Prepare the job script : Describe what you request
    1. Which Queue (cpu or gpu on EUCLID)
    2. Number of Nodes
    3. Number of (MPI) Tasks
    4. Number of Threads per task
    5. Memory per Task(optional)
    6. Job Wall Time (optional)
    7. STDERR and STDOUT FILES (optional)
    8. more details, see man sbatch.

- Minimal Job Script

```
#!/bin/bash
#SBATCH --nodes=1    <==== Request 1 node
                           i.e. 8 cores on EUCLID
module load iccifort <==== Load the necessary modules
Do What you usually do to run. For example.
mpirun my.exe
```

- A more detailed script

```
#!/bin/bash
#SBATCH -N 2  # Number of Tasks : 2 MPI Tasks
#SBATCH -c 8  # Threads per Task : 8 Threads/MPI Task
#SBATCH -n 1  # MPI Task per node
# ==> 16 cores on 2 nodes with 8 threads/MPI task
export OMP_NUM_THREADS=8
Do What you usually do to run. For example.
mpirun my.exe
```

- Submit a Job
  - `sbatch JobScript`

- Use of Batch System Environment variables
  - SLURM_NNODES
    : Number of Nodes
  - SLURM_NTASKS
    : Number of MPI Tasks
  - SLURM_TASKS_PER_NODE
    : Number of Tasks per Node
  - SLURM_CPUS_PER_TASK
    : Threads per Task
  - SLURM_NPROCS
    : Number of MPI Tasks: Equal to SLURM_NTASKS
  - SLURM_CPUS_ON_NODE
    : Number of CPUs per node
  - SLURM_JOB_CPUS_PER_NODE
    : Number of CPUs per node
  - SLURM_JOB_NUM_NODES
    : Number of Nodes

Questions ?

- Plz connect to training machine
- Work with modules
- Work with Batch system - Please do not send normal applications but simple commands like

  `hostname, date`

Problems ? Questions?