Some Very Under Done Instructions for HPC 2013

These instructions are only for a user with some experience. You need proficiency in Linux and parallel programming. Some details for the HPC2013 Cluster are in the other documents. In case of problems write to hpcsupport@iitk.ac.in. You will also be added to a list hpc@lists.iitk.ac.in . Read up on the instructions for HPC2010 as well as you may find it helpful. There may be some teething problems. For people used to the older cluster please pay attention to I_MPI_FABRICS in the scripts provided.

1. You can login to the master node by ssh –X <username>@hpc2013.hpc.iitk.ac.in Your password is the same as that for CC. We have 888 regular nodes with 20 cores (128 GB RAM) and five nodes with high memory (768 GB RAM).
2. **Always, after you logon to the cluster go to the workq by saying**
   qsub –I
         After you have finished, type
   exit

   workq is an interactive queue that places you on a node where you can run commands. **You cannot access /opt/software otherwise.**

3. You can change your password on a CC machine but not on the cluster.
4. The changed password will be effective on the cluster within an hour.
5. We have created a home directory for you which will be initially empty. The path for this directory is /home/<username>.
6. We have also created a /scratch<your-name> directory that is a temporary directory. The /scratch is faster than /home so you may prefer to write out temporary results and carry out computation here. /scratch contents can be deleted at any point of time and if not in use. All software are in /opt/software
7. Please read the structure of the queues given below
8. There is no backup and you are responsible for taking regular backup of your area.
9. Please use the cluster in a sensible manner, and follow the rules of engagement, otherwise you may land up causing problems to others.

Currently you can test your programs with the Intel compiler. Here is how.

There is a file in /opt/software/intel/initpaths.
To run the intel **32 bit** compilers you have to type
    source /opt/software/intel/initpaths **ia32**
To run the intel **64 bit** compilers you have to type
    source /opt/software/intel/initpaths **intel64**

If you want to do special tuning for trace analyzer then the second argument has to be special but you will have to do your own research on this. The commands above just use the default analyzer. Please read the Intel site documentation for details. **A common mistake is using the programs compiled on one cluster directly on the another cluster. You need to recompile programs if clusters are changed.**

After you have **sourced** the files you should compile your programs using the relevant programs such as mpiicc, mpicc etc. Confusion in PATH settings is one of the main sources of error.

Use the following in a file say "test" for submitting programs. Remember to do a chmod 755 to the file. Change this file as per nodes and queue required or for the job name. You can change some variable names such as the name of the queue and job in example file given below. Number of nodes should change with the queue. You should always keep ppn 20 except for "hyperthread" queue where ppn is 40 and workq parallel job where it is 4. The hyperthread queue is an experimental queue and may give better results than normal. If this is the case then please do inform us. Make it less only in exceptional circumstances and do not make it more. Even then restrict yourself to the number of nodes limit in any queue.

Script for running job on small/medium/large queue
```
#!/bin/bash
#PBS -N <sample job name>
#PBS -q small
#PBS -l nodes=1:ppn=20
#PBS -j oe
cd $PBS_O_WORKDIR
export I_MPI_FABRICS=shm:dapl
export I_MPI_MPD_TMPDIR=/scratch/<your_name>
mpirun -machinefile $PBS_NODEFILE -n 20 ./<64bit compiled program>
```

Now use qsub test to submit the job.
     qsub test
This will give you a number for your queued job.
All output and errors will go to test.<number_you_get>. You can check the status of your jobs by using
     qstat –u <your_login_name>
You can delete your job by
     qdel – W force <your job number>
See man qsub

A short description of the queues is below. Here workq should be an interactive queue as well as a batch queue and remaining queues are only batch queues. There may be some discrepancy in this functionality as numbers of days etc. are a policy decision.

queue	walltime	Max jobs run simultaneously	Min cores	Max cores	Min nodes	Max nodes	Total Nodes
workq	24 hours 2 hrs CPU time	2 (1 login + 1 testing)	1	6	1	1	4
small	5 days	3 running 1 waiting	20	40	1	2	96
medium	4 days	3 running 1 waiting	40	120	2	6	256
large	3 days	2 running 1 waiting	120	640	6	32	482
hyperthread (each node behaves as if it has 40 cores)	5 days	1 running 1 waiting	40	80	1	2	16
highmem (for large memory jobs)	5 days	1 running 1 waiting	2	20	1	1	5
mini (for jobs of small duration)	2 hours	1 running 1 waiting	20	40	1	2	32
test	-	-	-	-	-	-	2

If you do not see /opt/software when you login, it is deliberate. **You must use qsub -I**

Script for running parallel job on workq
#!/bin/bash
#PBS -N <sample job name>
#PBS -q workq
#PBS -l ncpus=6
#PBS -j oe
cd $PBS_O_WORKDIR
export I_MPI_FABRICS=shm:dapl
cd $PBS_O_WORKDIR
export I_MPI_MPD_TMPDIR=/scratch
mpirun -machinefile $PBS_NODEFILE -n 6 ./<64bit compiled program> | tee <test.txt>

Script for running a SEQUENTIAL JOB
#! /bin/bash
#PBS -l nodes=1:ppn=1
#PBS -N <a_name_for_your_job>
#PBS -q seq
#PBS -j oe
cd $PBS_O_WORKDIR
<fully_qualified_name_of_your_executable>

Script for submitting ANSYS FLUENT JOB
Fluent-PBS Startup Script: (Fields to be edited are marked in RED)
#! /bin/bash
#PBS -l nodes=2:ppn=20
#PBS -l fluent=1
#PBS -l fluent_lic=40
#PBS -q small
#PBS -V
#EXECUTION SEQUENCE
cd $PBS_O_WORKDIR
/opt/software/ansys_inc/v150/fluent/bin/fluent 3ddp -g -cnf=$PBS_NODEFILE –t40 -i
batch_fluent.jou

Script for submitting a parallel MATLAB PCT JOB
#!/bin/bash
#PBS -N M11
#PBS -l nodes=1:ppn=20
#PBS -l matlab_user=1
#PBS -l matlab_lic=20
#PBS -q small
#PBS -S /bin/bash
#PBS -j oe
echo "I ran on:"
cd $PBS_O_WORKDIR
cat $PBS_NODEFILE
export PATH=/opt/software/matlab11b/bin:$PATH

```
matlab -nosplash -nodisplay << !!
matlabpool open local 20
your_program
matlabpool close
!!
```

For running a GAUSSIAN Job
First prepare your input file molecule.com and then type
      submitLinda
and follow the instructions

The list of all possible scripts would be long but the domain experts, which you as user are supposed to be, should figure it out. You can install software in your own directories and in no-case would you require root privileges for installing the software.

For running distributed MATLAB please meet HPCSUPPORT in CC 212

For transferring files from HPC2013 and general warnings

1. Use your favorite sftp program such as winscp sftp etc.
2. Connect to hpc2013.hpc.iitk.ac.in
3. Transfer files
4. exit
5. In no case use the node to which you login for this purpose
6. **Please do not submit jobs on the node to which you logon**