CLUSTER USER GUIDE

Contributors: Gaël Guédon, Riccardo Mereu, Luigi Urbinati
Last update: ***April, 2019***

1 INTRODUCTION

This document is aimed at all members collaborating with the research groups that are part of the CFDHub (including Master and PhD students, Post-docs and all other members).

The goal is to provide information in order to get started using the available computational resources autonomously and share the rules for a common use of the whole HPC system.

1.1 WHAT IS THE "CLUSTER"?

The "cluster" is a HPC system located in Milan, Bovisa campus, which is at disposal for intensive computations. It is shared by various research groups of the Politecnico di Milano from various departments such as Bio, Chemical, Energy, Environmental and Mechanical Engineering. The management responsible is Dr. Riccardo Mereu and the technical responsible is Luigi Urbinati.

1.2 HARDWARE

It is composed of several homogeneous groups of computing units (also called "nodes"):

- Main group 1: 16 nodes DELL M620 (blade version) with Intel(R) Xeon(R) CPU E5-2650 V2 @ 2.6GHz processors for a total of 256 cores and 1024 Gb RAM
  node-0-1 up to node-0-16
- Main group 2: 2 nodes DELL R720 with Intel(R) Xeon(R) CPU E5-2650 V2 @ 2.6GHz processors and GPU NVIDIA K40 unit for a total of 32 cores, 2 GPUs and 128 Gb RAM
  nodevg-0-1 and nodevg-0-2
- Energy group 1: 6 nodes DELL R410 with Intel(R) Xeon(R) CPU E5670 @ 2.93GHz processors for a total of 72 cores and 192 Gb RAM
  nodo10 up to nodo15
- Energy group 2: 6 nodes
  4 nodes DELL R620 with Intel(R) Xeon(R) CPU E5-2680 0 @ 2.70GHz processors for a total of 64 cores and 256 Gb RAM
  nodo16 up to nodo19
  2 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 40 cores and 256 Gb RAM
  node-1-12 up to node-1-13

- Energy group 3: 8 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2640 v3 @ 2.60GHz processors for a total of 128 cores and 512 Gb RAM
  node-1-1 up to node-1-8
- Energy group 4: 16 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 256 cores and 1024 Gb RAM
  node-3-1 up to node-3-16
- Energy group 5: 12 nodes
  6 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 96 cores and 384 Gb RAM
  node-4-1 up to node-4-6
  6 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz processors for a total of 168 cores and 768 Gb RAM
  node-4-7 up to node-4-12
- Energy group 6
  2 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 40 cores and 256 Gb RAM
  node-1-14 up to node-1-15
- Chemical group 1: 16 nodes DELL C6100 with Intel(R) Xeon(R) CPU X5675 @ 3.07GHz processors for a total of 192 cores and 512 Gb RAM
  nodo-c01 up to nodo-c16
- Chemical group 2: 6 nodes
  3 nodes DELL R410 with Intel(R) Xeon(R) CPU E5540 @ 2.53GHz processors for a total of 24 cores and 72 Gb RAM
  nodo-n01 up to nodo-n03
  3 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 60 cores and 384 Gb RAM
  node-1-9 up to node-1-11
- Chemical group 3: 3 nodes DELL R410 with Intel(R) Xeon(R) CPU E5670 @ 2.93GHz processors for a total of 36 cores and 96 Gb RAM
  nodo-n04 up to nodo-n06
- Chemical group 4: 16 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 256 cores and 1024 Gb RAM
  node-2-1 up to node-2-16
- Chemical group 5: 18 nodes
  16 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz processors for a total of 256 cores and 1024 Gb RAM
  node-5-1 up to node-5-16
  2 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz processors for a total of 40 cores and 256 Gb RAM
  node-6-1 and node-6-2
  2 nodes DELL R730 with Intel(R) Xeon(R) CPU E5-2630 V3 @ 2.40GHz processors and GPU NVIDIA K80 unit for a total of 32 cores, 2 GPUs and 128 Gb RAM
  nodevg-5-1 and nodevg-5-2
- Bioengineering group 1:
  2 nodes SUN SF4450 with Intel(R) Xeon(R) CPU X7350 @ 2.93GHz processors for a total of 16 cores and 64 Gb RAM

nodo-b01 up to nodo-b02

2 nodes DELL R410 with Intel(R) Xeon(R) CPU E5670 @ 2.93GHz processors for a total of 24 cores and 64 Gb RAM

nodo-b03 up to nodo-b04

- Bioengineering group 2: 2 nodes Fujitsu RX300S7 with Intel(R) Xeon(R) CPU X5670 @ 2.93GHz processors for a total of 32 cores and 192 Gb RAM

  nodo-b05 and nodo-b06
- Mechanical group 1: 3 nodes

  1 node DELL R410 with Intel(R) Xeon(R) CPU E5670 @ 2.93GHz processors for a total of 12 cores and 24 Gb RAM

  nodo-m01

  2 nodes DELL M630 with Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz processors for a total of 56 cores and 256 Gb RAM

  node-6-14 and 6-15

1.3 THE "MASTER" AND ACCESS NODES (NODEVG-0-X)

The **master** is the machine that manages all the communications between the cluster resources and the login nodes. **Login nodes (nodevg-0-x)** are the nodes used to manage the interaction between users and graphical parts and the computing nodes. When you log into the cluster you actually log into the *master* and then to the *login nodes*.

In order to use the computing resources you will have to log into *nodevg-0-1* or *nodevg-0-2* to successively connect to the selected computing nodes.

Never launch a computational run from the master! Always check to be connected to the selected computing nodes before to launch the run!

All the processes running on the master or the login nodes will be killed without any advice if affecting the functionality of the whole system.

1.4 ADDITIONAL NOTES

In the following sections we will often refer to some keywords that are user specific, some of them are given below:

- <username> this is your username and you have to substitute it in the command or field;
- <password> this is your password and you have to substitute it in the command or field;
- <vncport> this is the port number of your VNC connection;
- <node> this is the name of the computing node you would like to use.

Vocabulary:

- alias this is a command line shortcut for an expression;
- bashrc this is a file that contain instructions that are executed each time a terminal is opened or each time you log into a machine through ssh.

2 ACCESS TO THE CFDHUB HPC

This section describes how to install and set-up the programs needed to connect your terminal (workstation or laptop) to the cluster graphical interface (VNC).

Three ways exist to access the CFDHub HPC:

(i) directly from PoliMi network;
(ii) from VPN service;
(iii) from tunnelling service.

The latter two methods are used when you are "outside" PoliMi. Specifically, the access through tunnelling machine (iii) has to be request and it is activated only for users without the opportunity to get a VPN access from his/her own Department.

2.1 VPN SERVICE ACTIVATION (ACCESS OUTSIDE POLIMI)

As first step to access the CFDHub HPC machines from outside PoliMi, you are required to ask your own Department the activation of the VPN service for your PoliMi account. Some indications about what it is and how it works are reported at https://www.asict.polimi.it/en/network-services/vpn.html

For whom is not able to get the VPN access (e.g., master thesis students) the access will be provided by using a tunnelling approach. This kind of access needs to be activated by the technical manager of the CFDHub and should be indicated when the access for new users is required.

2.2 STEPS FOR HPC ACCESS

Next sessions describe the steps to access the HPC machines (through both VPN service and tunnelling machine):

- installation of the software for remote control of the HPC machines;

- setting of SSH session to access and control the HPC machines;

- creation of the graphical port for remote control of the HPC machine and use of VNC tool;

- setting of VNC session to graphically access and control the HPC machines;

- setting to upload and download files to/from HPC machines to the local one.

2.3 SOFTWARE INSTALLATION

1) Download and install the software Mobaxterm: https://mobaxterm.mobatek.net/download.html

2.4 SETTING OF SSH SESSION

2) Open the software and go to 'Sessions' Tab → 'New Session' → choose 'SSH' and open the 'Session setting' panel

3) Set up the SSH session as described in Figure 1 for connection through VPN. Use your username instead of 'mereu' and 'nodevg-0-1' instead of 'nodevg-0-2' based on your group category[1]
4) Set up the SSH session as described in Figure 2 for connection through tunnelling machine. Use your username instead of 'mereu' and '10.0.0.121' instead of '10.0.0.122' based on your group category[2]
5) Click OK to save the session

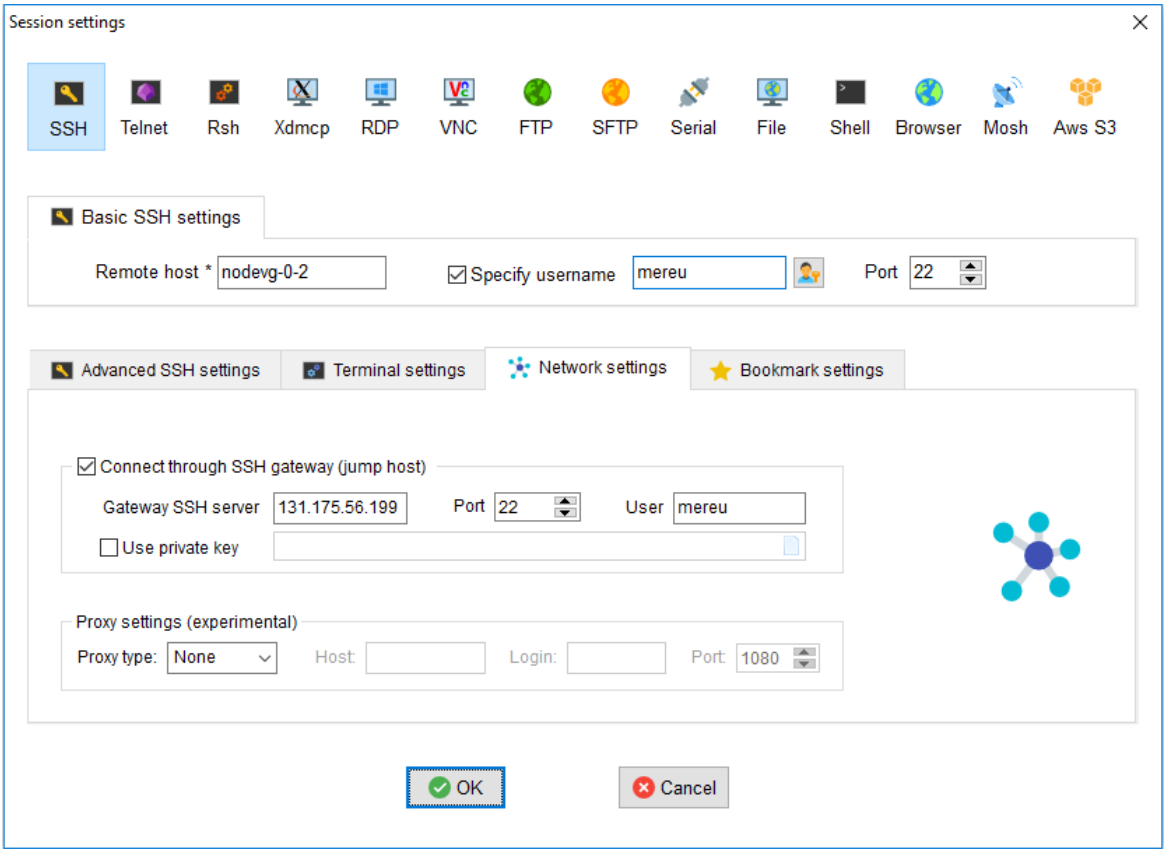


Figure 1 SSH setting for access through VPN service

[1] nodevg-0-1: research group of Chemical, Mechanical and Mathematical Departments
nodevg-0-2: research group of Energy, Electronic-BioEngineering and not mentioned Departments

[2] 10.0.0.121: research group of Chemical, Mechanical and Mathematical Departments
10.0.0.122: research group of Energy, Electronic-BioEngineering and not mentioned Departments

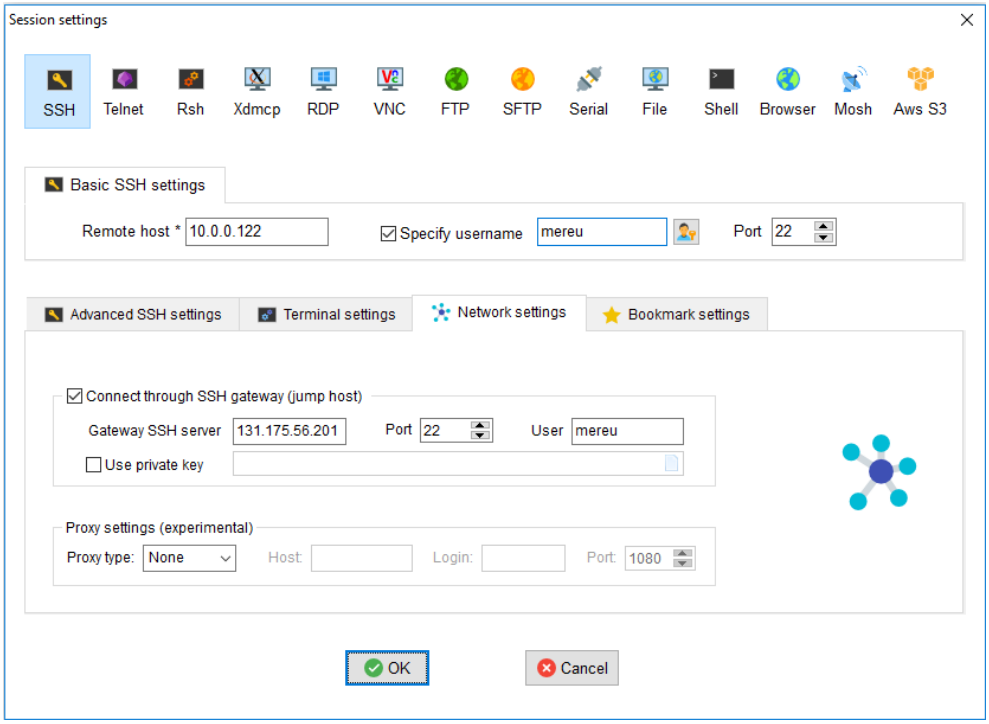


Figure 2 SSH setting for access through tunneling machine

2.5 CREATION OF THE GRAPHICAL PORT

6) Open the SSH session created in the previous steps
7) In the SSH terminal digit 'vncserver -name *username* -geometry *0000X0000* -depth 24' where *username* is your user and *0000x0000* the resolution of the monitor you use to work (e.g. vncserver -name mereu -geometry 1920x1080 -depth 24)
8) The graphical port assigned to you is indicated by the system (e.g. Desktop 'TurboVNC: nodevg-0-2.local:01 (mereu)' started on display nodevg-0-2.local:01)

2.6 SETTING OF VNC SESSION

9) Open the software and go to 'Sessions' Tab → 'New Session' → choose 'VNC' and open the 'Session setting' panel
10) Set up the VNC session as described in Figure 3 for connection through VPN. Use your username instead of 'mereu', 'nodevg-0-1' instead of 'nodevg-0-2' based on your group category[3] and the port number provided by the system instead of 5901 of step 8 and Figure 3 (e.g. if the assigned port is 35 put 5935)
11) Set up the VNC session as described in Figure 4 for connection through tunnelling machine. Use your username instead of 'mereu', '10.0.0.121' instead of '10.0.0.122' based on your group category[4] and the port number provided by the system instead of 5901 of step 8 and Figure 2 (e.g. if the assigned port is 35 put 5935)
12) Click OK to save the session

[3] nodevg-0-1: research group of Chemical, Mechanical and Mathematical Departments
nodevg-0-2: research group of Energy, Electronic-BioEngineering and not mentioned Departments

[4] 10.0.0.121: research group of Chemical, Mechanical and Mathematical Departments
10.0.0.122: research group of Energy, Electronic-BioEngineering and not mentioned Departments

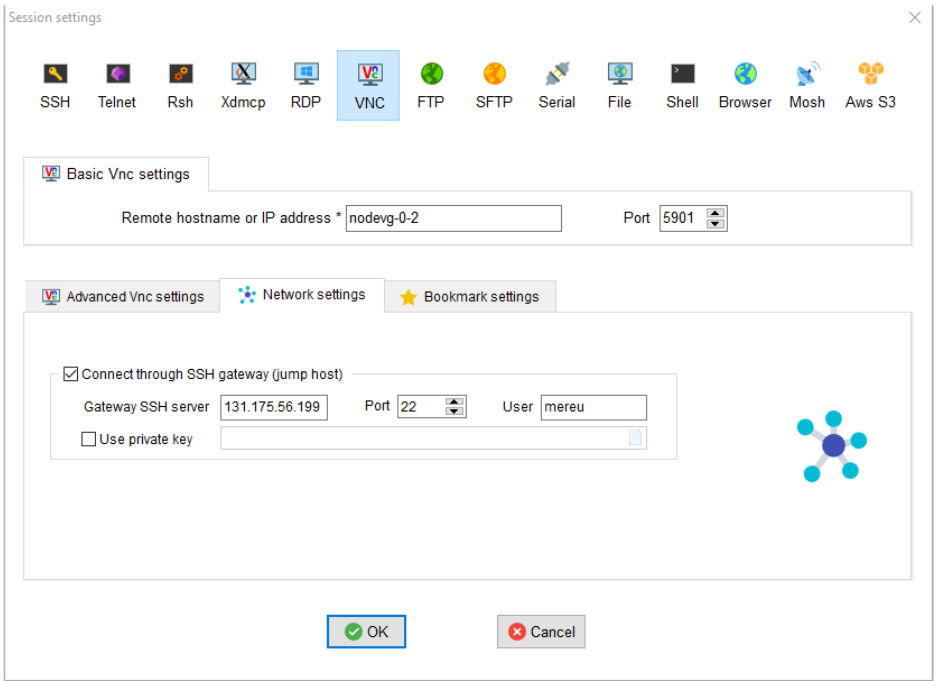


Figure 3 VNC setting for access through VPN service

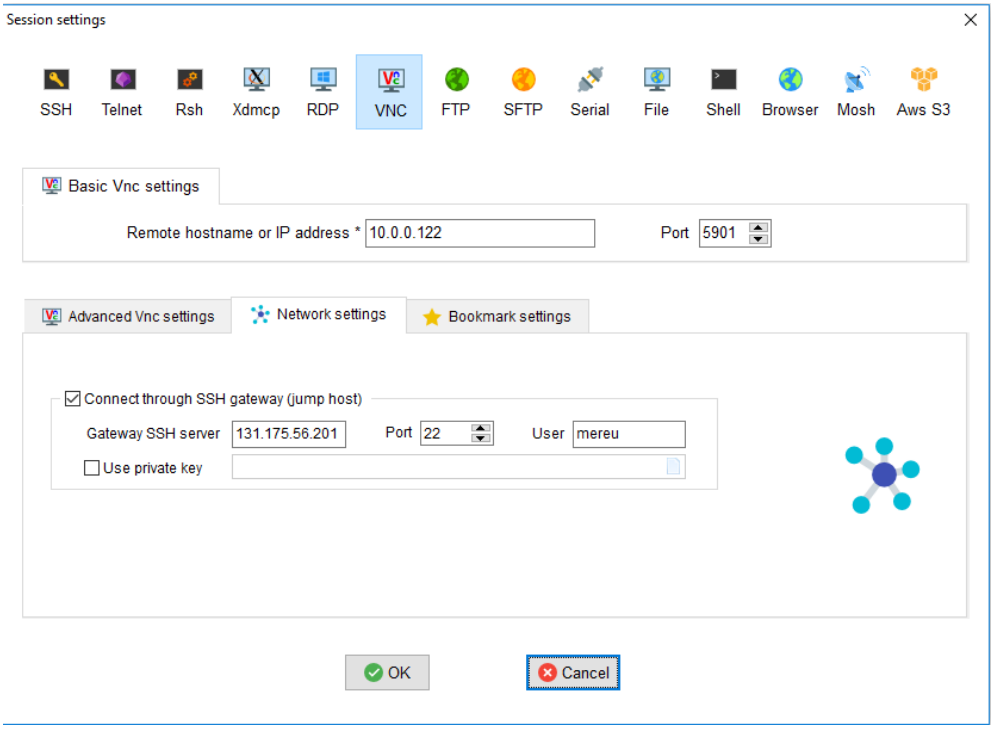


Figure 4 VNC setting for access through tunneling machine

3 TRANSFER FILES

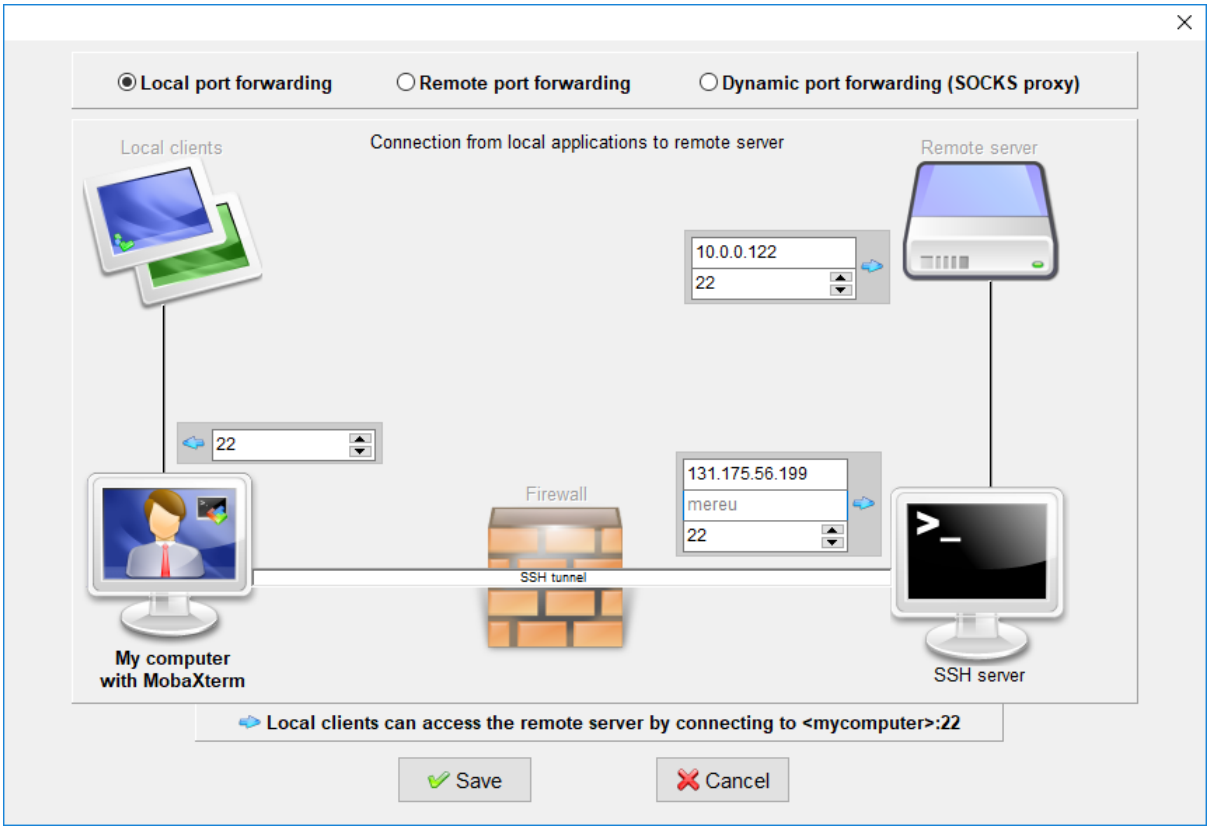
In order to transfer files from your terminal to the cluster and vice versa, the FTP the software Filezilla can be used.

3.1 SOFTWARE INSTALLATION

1) Download and install the software Filezilla: https://filezilla-project.org/download.php?type=client

3.2 TUNNELLING SETUP

2) In MobaxTerm open 'tools' and then 'MobaSSHTunnel (port forwarding)'
3) Create a local port forwarding (New SSH Tunnel) with the set-up indicated in the image below. Use your username instead of 'mereu', '10.0.0.121' instead of '10.0.0.122' based on your group category[5]
4) Open tab 'Tunnelling' and run the symbol play in 'MobaSSHTunnel (port forwarding)'



5) Open Filezilla or session 'SFTP' in mobaxterm and insert host 127.0.0.1, your username, your password and port 22

[5] 10.0.0.121: research group of Chemical, Mechanical and Mathematical Departments
10.0.0.122: research group of Energy, Electronic-BioEngineering and not mentioned Departments

Once you inserted your user data and accessed to the cluster, you will see in the left side your terminal and in the right side the cluster folders (/home/energia/mereu in the example above).

To transfer (copy) data just drag files from one side to the other.

4 WORKING AREAS

The system is divided in different working areas in order to permit a more sustainable and efficient use of the available resources. The main working areas are reported and described here, please check the rules for using each working area in order to avoid affecting the whole system functionality:

/home

Purpose: to save personal data such as libraries, sources, compiled code, documents etc. In general, this area is reserved to files that you think should be backed up.

Capacity: a quota for each group is assigned, this limitation permits to avoid the filling up of the /home area affecting other groups or users. To know the total quota and the actual occupancy of the available space type "repquota –augs"

Access: all nodes

Backup: YES

/fast-scratch & /big-scratch

Purpose: launch runs and put data actually on use. In order to preserve the purpose of this area and avoid a filling up of the area **all data older than 50 days will be deleted from this area.** Please be careful and move your data to /ARCHIVIO area when they are not on use anymore

Capacity: approx. 6Tb to 30Tb on SSD (high speed) cache disk interfaces (normal) NLSAS disks to speed up data exchange processes.

Access: all nodes

Backup: NO

/ARCHIVIO

Purpose: save the results and data you want to keep for long term. This area permits to store data without affecting the running processes in other working areas.

Capacity: related to the amount of storage purchased as a group, divided into blocks of 8Tb.

Access: all nodes

Backup: NO, however considered reliable being residing on enterprise band hard-drives with multi-disk data redundancy

5 ACCESS TO THE NODES

5.1 NODE ACCESS

First of all, **you will always have to log into a computing node before running any program**. Running a program by mistake on the "master" or "login node" will slow down, if not block, every user connection… So be careful!

After having logged to the cluster through the VNC client, you will have the possibility to open a terminal (see section 10 Additional resources).

In order to log into a node you will have to run this command on the terminal:

```
[<username>@nodevg-0-X ~]$ ssh -Y <node>
```

5.2 CHECK NODE USAGE

You can briefly look at node usage opening Firefox or any web browser on the cluster side and looking at the web page http://master/ganglia.

From there you can have an overview of the CPU usage of all the nodes. The main drawback is that it still does not allow you to check if a node is "completely" free, i.e., someone may still be using the node to set-up or post-process a case… You will have to follow the next instructions to check completely the availability of a computing node.

After having accessed the node (see previous section), you will have to run this command to check the active processes of all users:

```
[<username>@<node> ~]$ ps -aux
```

It will list all the processes and you will have to look for the ones that may use CPU (3rd column) and/or RAM memory (4th column). You should not occupy a computing node if you see processes that use CPU and/or RAM. It means it is not free. If you have doubts you can **ask your supervisor**.

6 ACCESS TO THE QUEUE (INTERACTIVE JOBS)

In order to access common nodes (node-0-5 to node-0-16) the use of queue is required. To access the node the following steps are suggested.

6.1 AVAILABLE QUEUE AND NODES

- node-0-5 to node-0-8 will be accessible from 'hub72.q' -> the queue will stop all process after 72 hours (3 days) and the re-submission of the job will be required

- node-0-9 to node-0-16 will be accessible from 'hub.q' -> the queue will stop all process after 168 hours (1 week) and the re-submission of the job will be required

6.2 CHECK OF THE QUEUE

qstat -f *'queue name'* (available from nodevg-0-1 and nodevg-0-2)

Please, look at the nodes that are assigned to you and the related number of processor for each node. In this way you know the available nodes can be assigned to you.

6.3 ACCESS TO NODES

Using the interactive approach (qlogin) it is possible to access the available nodes indicating the number of processors, parallel system and node to be used, through the command:

qlogin -pe mpi ## -q "*@node-0-X"

where *##* is the number of processors and *node-0-X* the node selected in the previous step

Example:

```
[luigi@nodevg-0-1 ~]$ qstat -f -q hub.q
queuename                      qtype resv/used/tot. load_avg arch          states
---------------------------------------------------------------------------------
hub.q@node-0-10.local          BIP   0/16/16        1.00     lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-11.local          BIP   0/16/16        0.11     lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-12.local          BIP   0/16/16        13.15    lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-13.local          BIP   0/16/16        16.12    lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-14.local          BIP   0/16/16        16.35    lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-15.local          BIP   0/16/16        14.05    lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-16.local          BIP   0/0/16         0.14     lx-amd64
---------------------------------------------------------------------------------
hub.q@node-0-9.local           BIP   0/16/16        15.41    lx-amd64
[luigi@nodevg-0-1 ~]$qlogin -pe mpi 16 -q "*@node-0-16"
[luigi@nodevg-0-1 ~]$ qstat -u luigi
job-ID  prior   name       user         state submit/start at     queue                          slots ja-task-ID
-----------------------------------------------------------------------------------------------------------------
  18090 0.53227 QLOGIN     luigi        r     03/05/2019 17:56:57 hub.q@node-0-16.local          16
```

7 USE OF THE SOFTWARE

7.1 LIST OF SOFTWARE

- ABAQUS
- ANSYS PACKAGE
- COMSOL
- FDS
- MATLAB
- Python
- OpenFOAM
- foam-extend

Check the directory /software/ for software versions and other software.

7.2 ANSYS FLUENT USAGE

This section presents how to get started with the ANSYS Fluent CFD code. It does not cover how to use the code but only how to access it and run simulations on a single or multiple computing nodes. It is expected that the user have some experience with the code.

7.2.1 LAUNCH FLUENT FROM TERMINAL

Before to run a case it is necessary to enter in the node you want to use, as explained in the section 4.1 and 4.2.

After you opened a terminal (as explained in section 10 Additional resources) and entered in the node, you should go into the folder where you have the case to be run, <mydir>, and type:

```
[<username>@<node> <mydir>]$
/software/ansys162/ansys_inc/v170/fluent/bin/fluent
```

to launch Fluent version 17.0. If you want to run release 16.2, type:

```
[<username>@<node> <mydir>]$
/software/ansys162/ansys_inc/v162/fluent/bin/fluent
```

In case you want to run the release 14, 14.5 or 15 type:

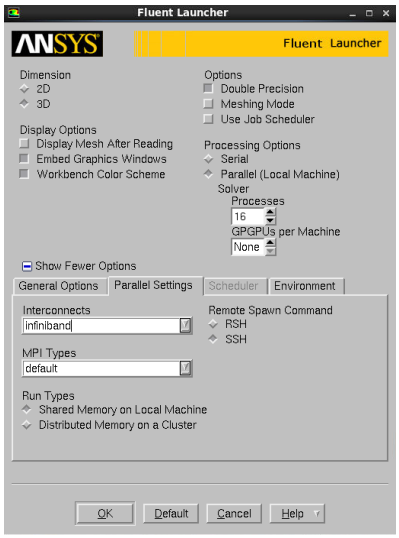
```
[<username>@<node> <mydir>]$ /software/ansys_inc/<version>/fluent/bin/fluent
```

where <version> is v140, v145 or v150 according to the version wanted.

You can also create an alias for each version, but it is up to you to understand how to do it.

7.2.2 SINGLE NODE PARALLEL SET-UP

To run Fluent in parallel on a single node, the following options should be set-up on the launcher:



First log into the node you want to use!

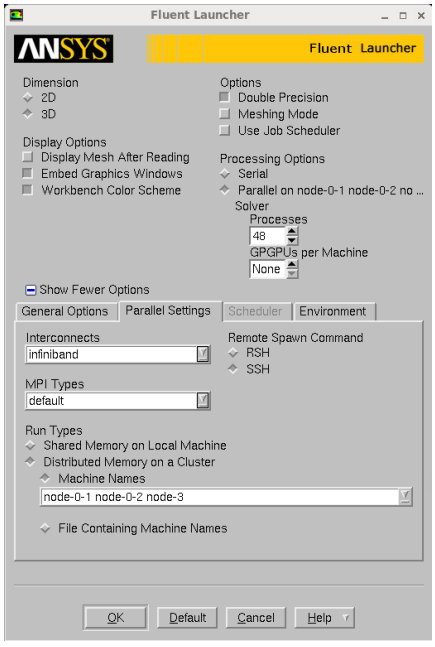
Processing Options: Parallel

Processes: the number of cores to use

Run Types: Shared Memory on Local Machine

7.2.3 MULTIPLE NODES PARALLEL SET-UP

To run Fluent in parallel on multiple nodes, the following options should be set-up on the launcher:



First log into one of the nodes you want to use!

Processing Options: Parallel

Processes: the **total** number of cores to use

Run Types: Distributed Memory on a Cluster

Machine Names: the list of nodes to use

7.3 MATLAB USAGE

Before to run a case it is necessary to enter in the node you want to use, as explained in the section 4.1 and 4.2.

After you opened a terminal (as explained in section 10 Additional resources) and entered in the node, you should go into the folder where you have the case to be run, <mydir>, and type:

```
[<username>@<node> <mydir>]$ /software/MATLAB/R2015a/bin/matlab
```

to run the release 2015a of MATLAB. Other versions are run similarly.

7.4 OPENFOAM USAGE

This section presents how to get started with the OpenFOAM CFD code. It does not cover how to use the code but only how to access it and run simulations on a single or multiple computing nodes. It is expected that the user have some experience with the code.

7.4.1 PRELIMINARY SET-UP

First of all, you have to check if an “alias” has been created for the sourcing of the code “bashrc”. To perform this operation you will have to open your user “bashrc” in a text editor, e.g. running this command:

```
[<user>@master ~]$ gedit .bashrc
```

The “bashrc” file should contain the following string for the OpenFOAM “bashrc” alias

```
alias OF231='module load openmpi-x86_64; source /software/OpenFOAM/OpenFOAM-
2.3.1/etc/bashrc WM_NCOMPPROCS=4 foamCompiler=ThirdParty WM_COMPILER=Gcc48
WM_MPLIB=SYSTEMOPENMPI'
```

It also should contain this line to load the system OpenMPI libraries used for parallel computing

```
module load openmpi-x86_64 || export PATH=$PATH:/usr/lib64/openmpi/bin
```

These operations should be performed only once.

7.4.2 START USING OPENFOAM

Each time you would like to use the OpenFOAM code, you will have to load its environment variables. This is done running the alias defined in your "bashrc", i.e., if you are on the master, running the command

```
[<user>@master ~]$ OF231
```

Otherwise if you are on a computing node

```
[<user>@<node> ~]$ OF231
```

7.4.3 RUNNING IN PARALLEL ON A SINGLE NODE

To run an application in parallel on a single node you will generally need to run the following commands after having logged into the node and moved to your working directory <mydir>

```
[<user>@<node> <mydir>]$ decomposePar
[<user>@<node> <mydir>]$ mpirun –np <NP> <application> -parallel
[<user>@<node> <mydir>]$ reconstructPar
```

Where <NP> is the number of cores to use and <application> is the name of the solver to be used.

7.4.4 RUNNING IN PARALLEL ON MULTIPLE NODES

To run an application in parallel on multiple nodes you will generally need to run the following commands after having logged into one of the nodes

```
[<user>@<node1> <mydir>]$ decomposePar
[<user>@<node1> <mydir>]$ mpirun –np <NP> -host <node1>,<node2>,<node3>
                          <application> -parallel
[<user>@<node1> <mydir>]$ reconstructPar
```

Where <NP> is the total number of cores to use, i.e., the number of processes that will run on each node is the total number of cores divided by the number of nodes.

8 AVAILABLE LIBRARIES

To check system libraries and version, type

```
[<user>@nodevg-0-1 <mydir>]$ rpm -qa
```

To filter the results, type

```
[<user>@nodevg-0-1 <mydir>]$ rpm -qa | grep <library_to_check>
```

For additional compiled libraries (local), check the folders in /software/chimica2/

9 ADVANCED USAGE

Here is an incomplete list of tips and tricks for advanced usage.

9.1 KEEP CURRENT DIRECTORY AFTER SSH

Add a function in your bashrc:

```
function sshh() {
  ssh -t $@ "cd $PWD; bash -l"
}
```

Then in order to log into a node from your working directory <mydir> do:

```
[<user>@nodevg-0-1 <mydir>]$ sshh –Y <node>
```

10 ADDITIONAL RESOURCES

If you are not familiar with the Linux environment you can use the following links to open a terminal and basic Linux commands:

Open a terminal

Basic Linux Commands