First steps on the clusters

This page explains what to do after [having successfully connected](#) to one of the clusters.

Step-by-step guide

1. **What is installed on the clusters**

   To see the list of installed software (modules), load the compiler and the MPI implementation you intend to use and do

   ```
   module spider
   ```

   > ⓘ **Open Source or proprietary?**
   >
   > On our systems, we have compiled and installed software modules either with **Intel Compiler and Intel MPI** or **GCC and MVAPICH2** and those are the only supported compiler/MPI combinations.

2. **Getting the examples**

   Once you have logged in to the machine, we suggest you download the examples with the command:

   > git clone https://c4science.ch/diffusion/SCEXAMPLES/scitas-examples.git

   Here is a list of our examples:

```
Advanced
        - FakeJobArray
        - JobArray
        - JobArray2
        - OccupyOneNode


Basic
        - hello.run
        - MPI
        - one_GPU.run
        - Pi_integral
        - Pi_mc


Modules
        - Abaqus
        - adf
         - Ansys
        - Comsol
        - cp2k
        - cpmd
        - fluent
        - gaussian
        - GPU_amber
         - GPU_gromacs
        - maple
        - Mathematica
        - Matlab
        - molpro
        - oommf
        - ParaView
        - R
        - spark
        - tensorflow
        - vasp
```

3. **Running the examples**

   Enter the directory scitas-examples and choose the example to run by navigating the folders. We have three categories of examples: Basic (examples to get you started), Advanced (including hybrid jobs and job arrays) and Modules (specific examples of installed software).

   To run an example, e.g. HPL-mpi of the Advanced category, do:

   ```
   sbatch --partition=debug hpl.run
   ```

   or, if you do not wish to run on the debug partition,

   ```
   sbatch hpl.run
   ```

4. **Running interactive jobs**

   An interactive job allows you to connect directly to a compute node and type commands that run on the compute node. Simply type the command Sinteract from the login node to start an interactive session with 1 core and 4GB of memory for 30 minutes.

   You can use the parameters to Sinteract (for help type: interact -h) to request more resources or more time.

```
usage: Sinteract [-c cores] [-n tasks] [-t time] [-m memory] [-p partition] [-a account] [-q qos] [-g
resource] [-r reservation]

options:
 -c cores cores per task (default: 1)
 -n tasks number of tasks (default: 1)
 -t time as hh:mm:ss (default: 00:30:00)
 -m memory as #[K|M|G] (default: 4G)
 -p partition (default: parallel)
 -a account (default: phpc2017)
 -q qos as [normal|gpu|gpu_free|mic|...] (default: )
 -g resource as [gpu|mic][:count] (default is empty)
 -r reservation reservation name (default is empty
```

e.g. to run an mpi job with 16 processes for one hour using 32 GB of memory on a debug node:

```
Sinteract -n 16 -t 01:00:00 -m 32G -p debug
```

> ⊘ On the Izar cluster, the -g option is necessary to request the desired number of GPUs. For example:
> Sinteract -g gpu:1

Related articles

- Connecting to the clusters
- First steps on the clusters
- Ask for help
- Using software on the SCITAS clusters