HANDOUT

Hardware & running efficiently

Layout / overview

MPI / threads

Memory management

HANDOUT

Computational efficiency

You should run as fast as possible.

BUT:
When it runs as fast as you need it to – stop.

HANDOUT

Aim:

- Avoid crashes (run stable)
- Allow tuning and effcient use

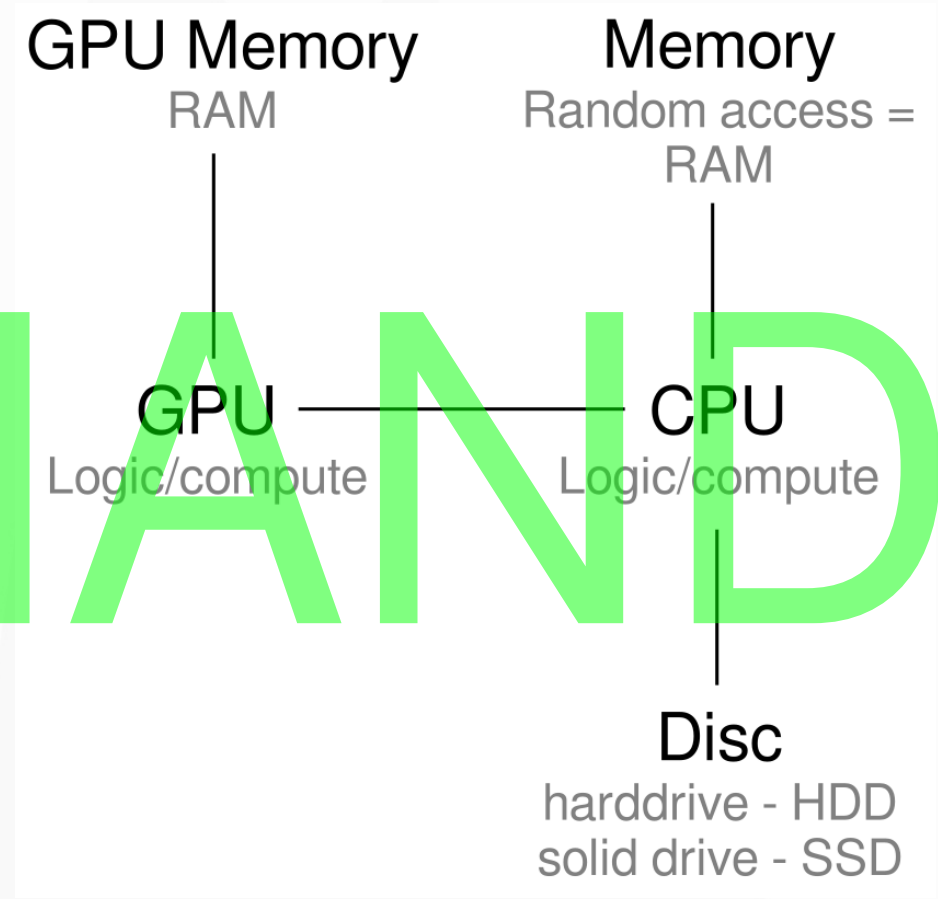
Ideally, RELION should take care of all these things for you, (tune run-settings and memory stability)

WE ARE WORKING ON IT. Right now, you need to be a bit clever. (because relion is not (yet))

HANDOUT

Layout / overview

A Computer.



Side-note:

In many contexts, and technical literature, a GPU can also be called

“ a device ”
“ an accelerator ”

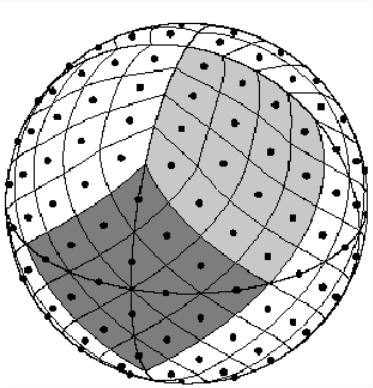
Whereas the CPU may in contrast be termed by

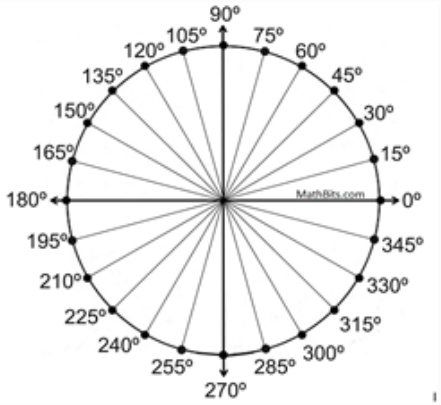
“ the host ”

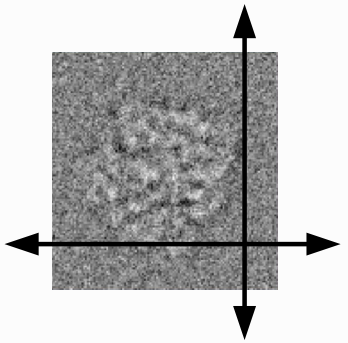
HANDOUT

3D Alignment

10k partcl.

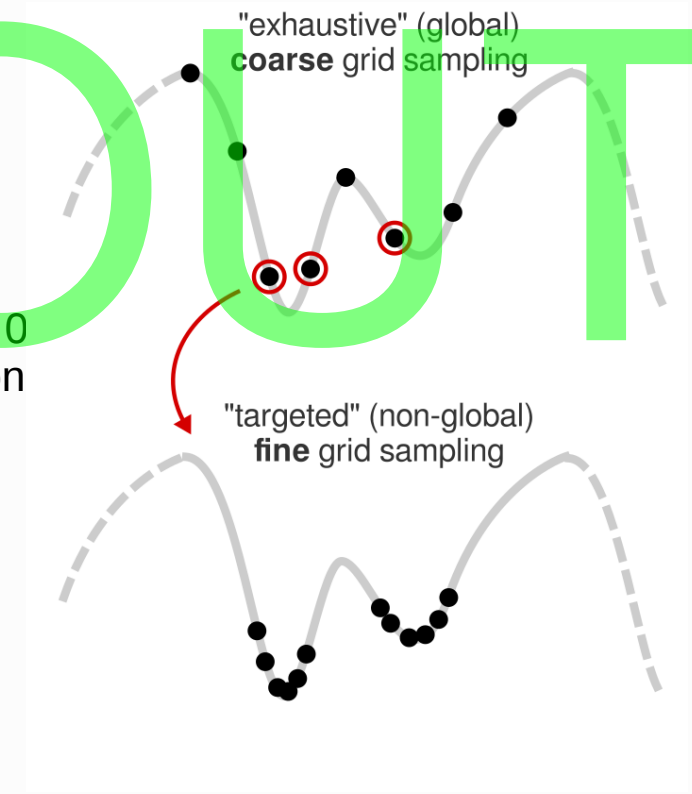




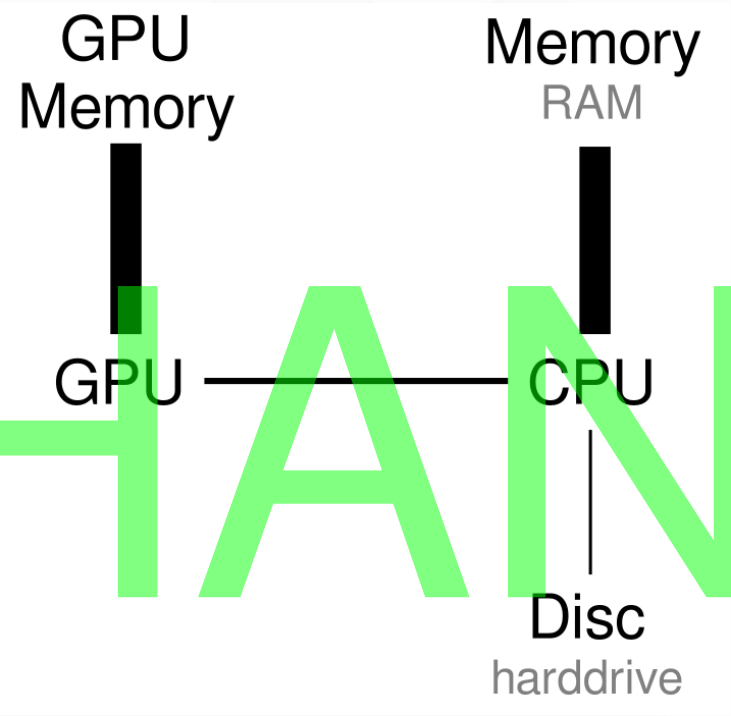


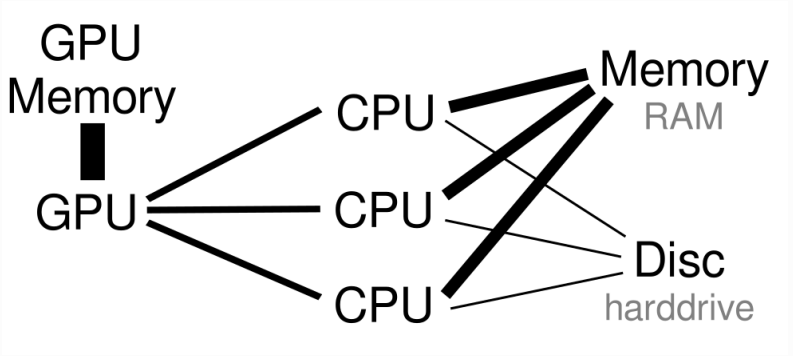
HANDOUT

Expectation iteration 18 of 25
13/ **13 sec** .....~~(,_,">
Maximization ...
3/ 3 sec .....~~(,_,">
Estimating accuracies in the orientational assignment ...
7/ 7 sec .....~~(,_,">
Auto-refine: Estimated accuracy angles= 2.629 degrees; offsets= 0
CurrentResolution= **15.3913** Angstroms, which requires orientation degrees for a particle of diameter 200 Angstroms
Oversampling= 0 NrHiddenVariableSamplingPoints= **387,072**
OrientationalSampling= 15 NrOrientations= **4608**
TranslationalSampling= 2 NrTranslations= **21**
==========
Oversampling= 1 NrHiddenVariableSamplingPoints= **12,386,304**
OrientationalSampling= 7.5 NrOrientations= 36864
TranslationalSampling= 1 NrTranslations= 84

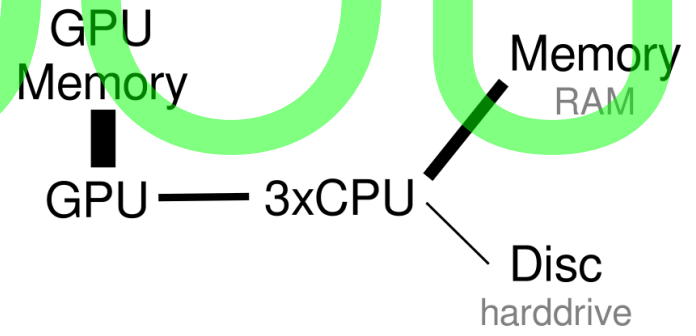


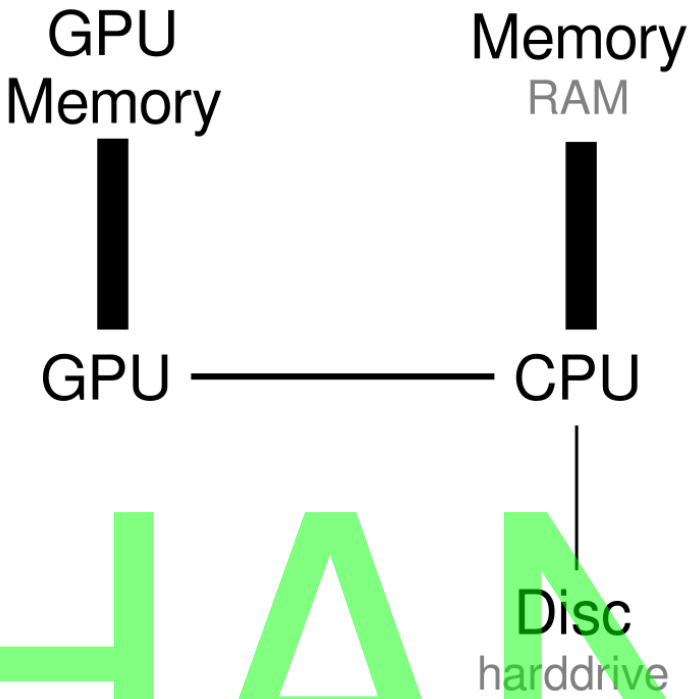
A Computer.

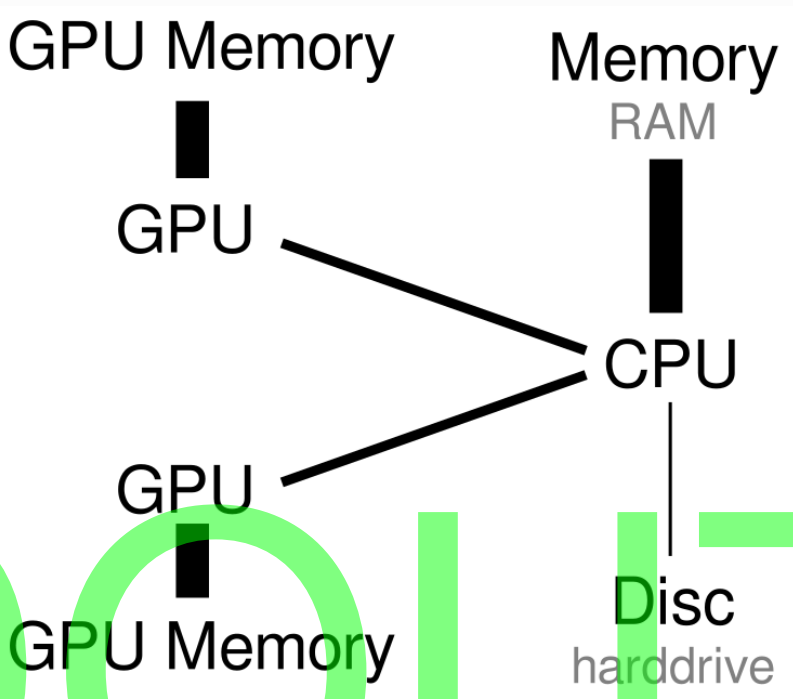




HANDOUT







HANDOUT

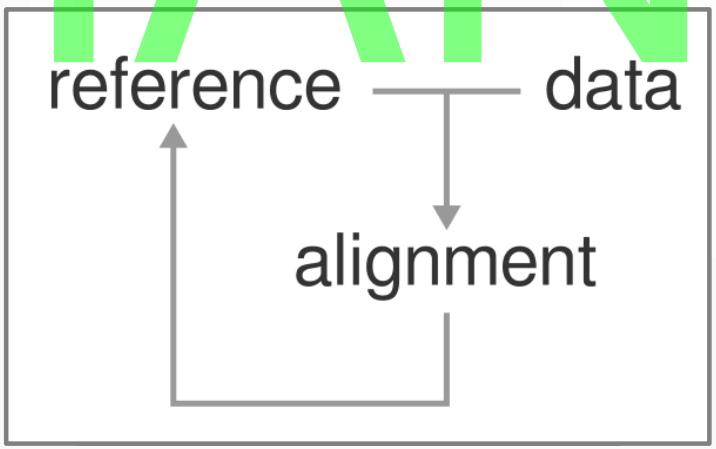
How many CPUs per GPU?

"How much **hardware** connected to RAM do I need to make tasks for the GPU, to keep the GPU busy?"

Dep. on CPU/GPU, but 2-3, typically. This concerns what to **buy**, not what to **run**.

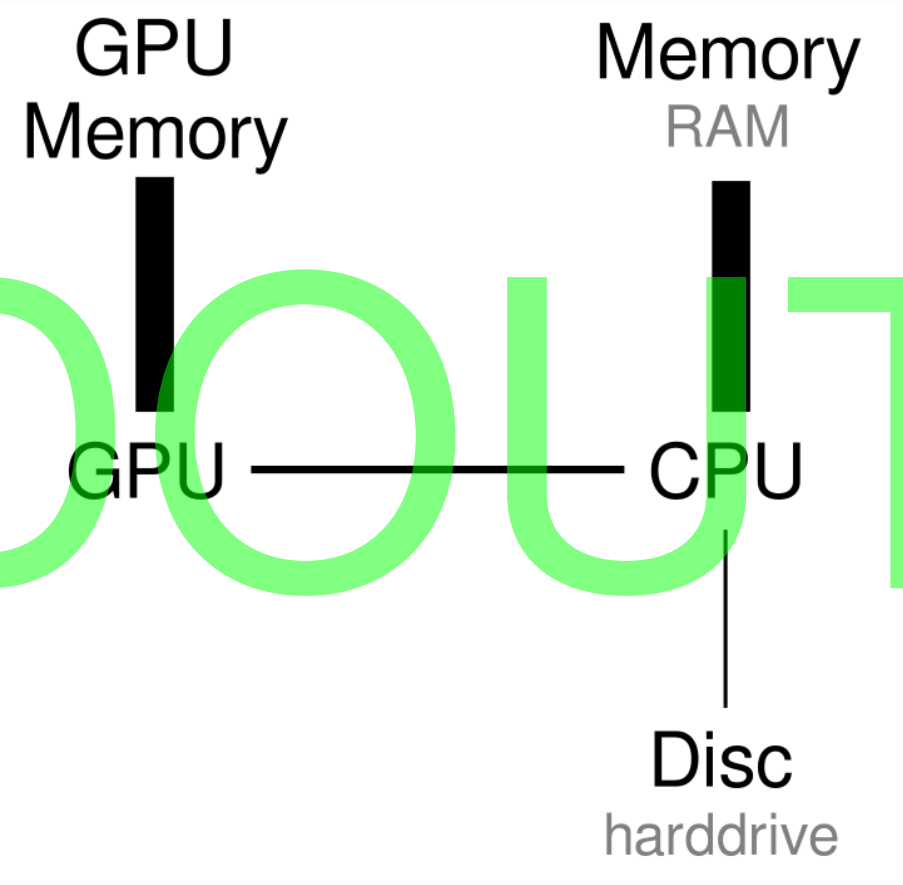
Ideally, we would have the references on the GPU and keep all the data as accessible as possible.

We also want to run as many CPUs as possible to keep the GPU busy.



Class reference(s)

Dataset



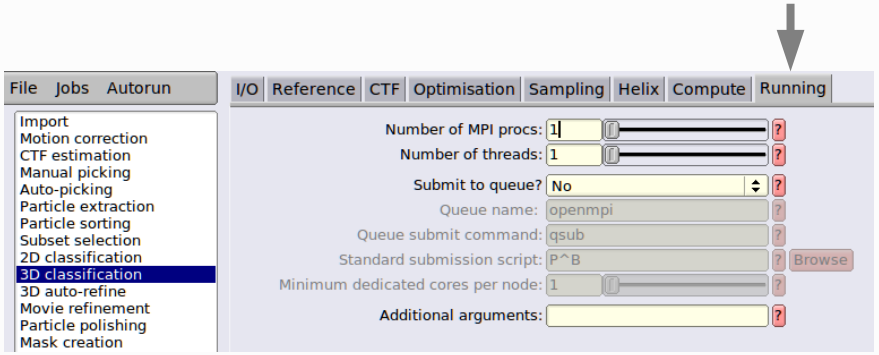
(Dataset)

HANDOUT

HANDOUT

Running parallel software

MPI processes



MPI = Message Passing Interface

Designed to supply communication of parallel processes which do not share memory.

As a consequence, they CANNOT share memory (objects).

In Relion, the first MPI *rank* (0), distributes work to the others.

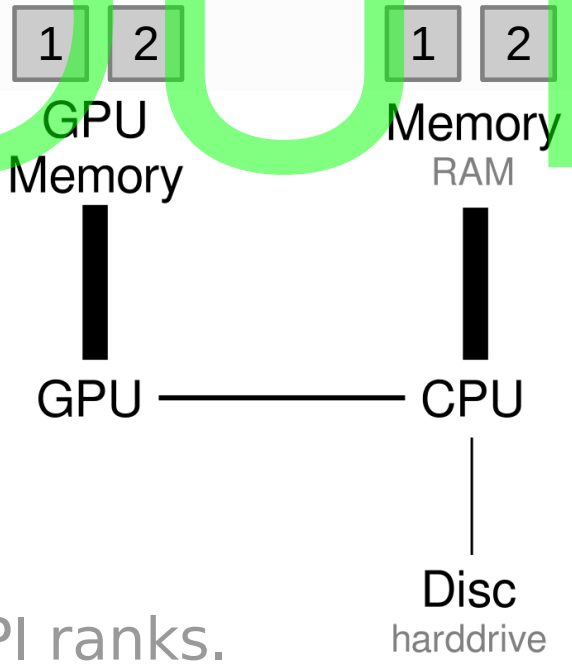
Master (0) distributes work to

slave (1)

slave (2)

...

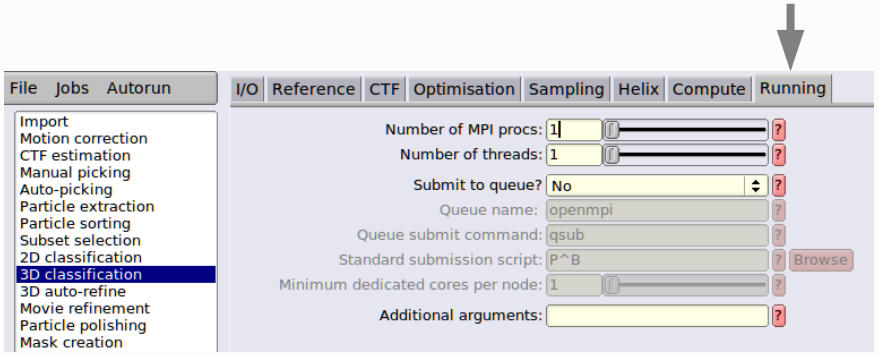
slave (N)



NOTE; to get N slaves working, start N+1 MPI ranks.

HANDOUT

Threads



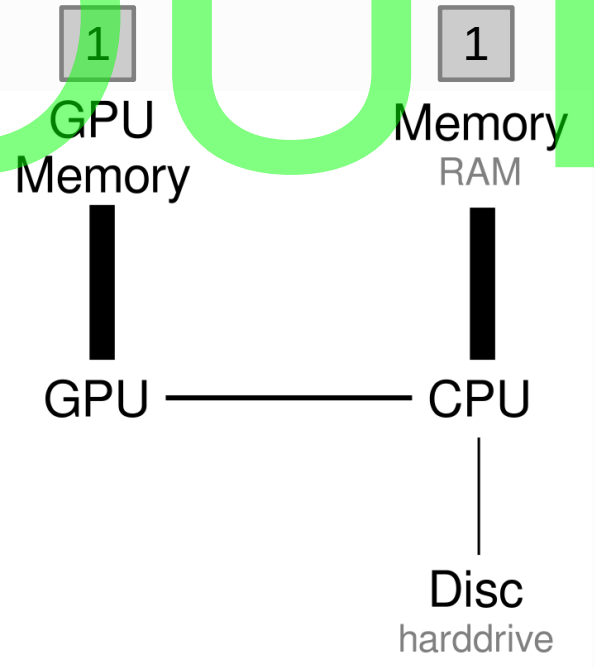
"workers"

Each MPI-slave will start a number of sub-processes, which are called threads.

Threads (within an MPI-rank) can work on the same memory object!

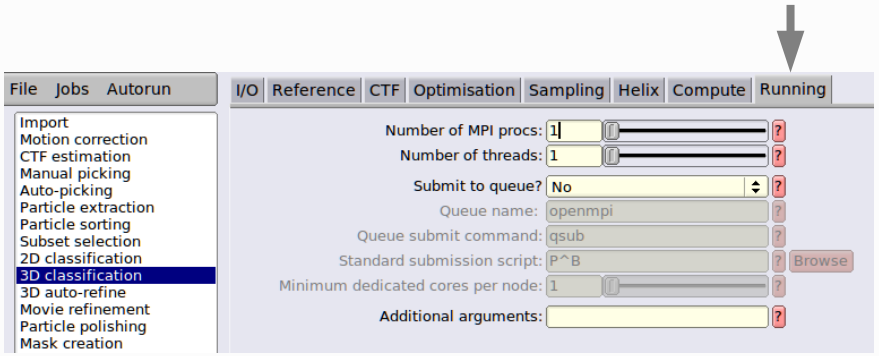
Threads are **all slaves** (there is no master)

If we specify 5 MPIs and 3 threads, how many working threads do we get?



HANDOUT

Threads



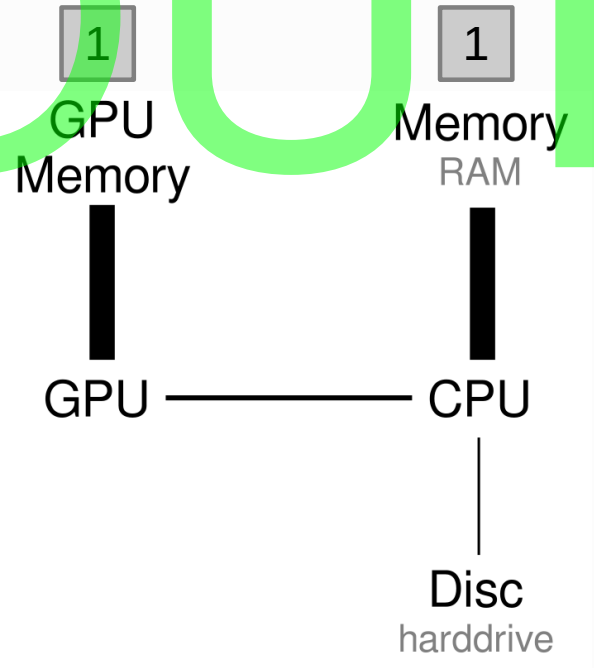
"workers"

Each MPI-slave will start a number of sub-processes, which are called threads.

Threads (within an MPI-rank) can work on the same memory object!

Threads are **all slaves** (there is no master)

HANDOUT



If we specify 5 MPIs and 3 threads, how many working threads do we get?

12 (4 slaves * 3 threads)

Recommendations

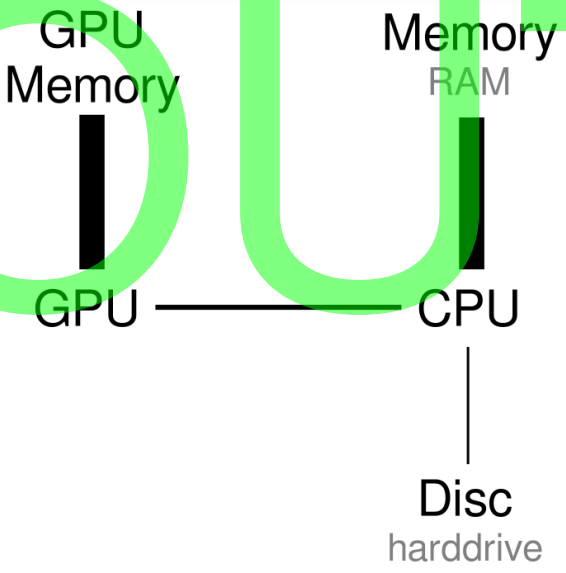
A number of MPI slaves will be faster than the same number of threads. How much faster? Depends, but 20%.

Will change in future versions

HANDOUT

4 MPI slaves will take 4 times the memory, both on CPU and GPU.

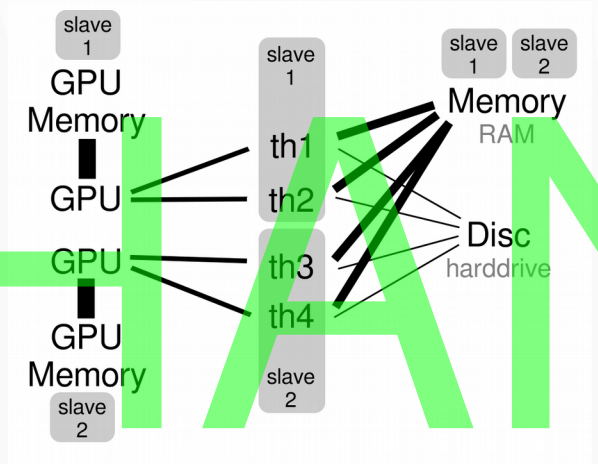
The general recommendation is 1 MPI / GPU, **if you can pay the CPU-side RAM cost.**



CPU-RAM:	**60 GB**
Dataset:	**35 GB**
GPUs:	**2**
MPIs:	**3?** (2 slaves + 1 master)

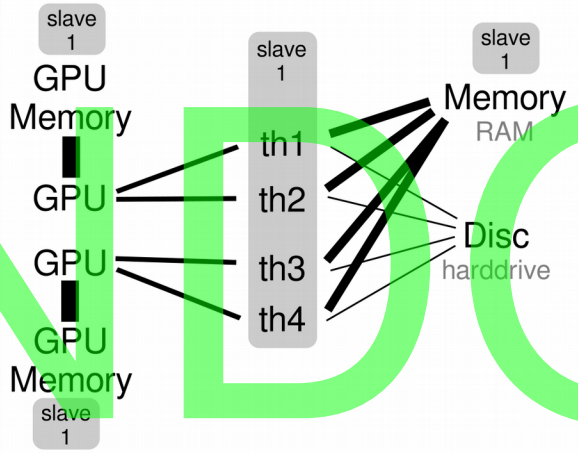
Recommendations

3 MPIs (2 slaves)
&
2 threads



We will need to go to disc for at least some of our data each iteration

no MPI (1 slave)
&
4 threads

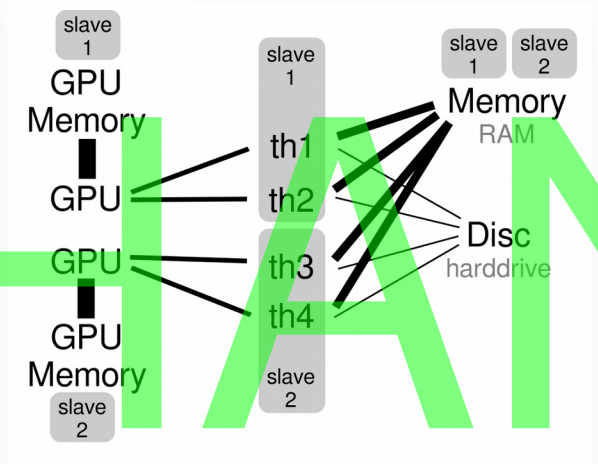


All data can fit into RAM and we may completely avoid **slow** disc access

HANDOUT

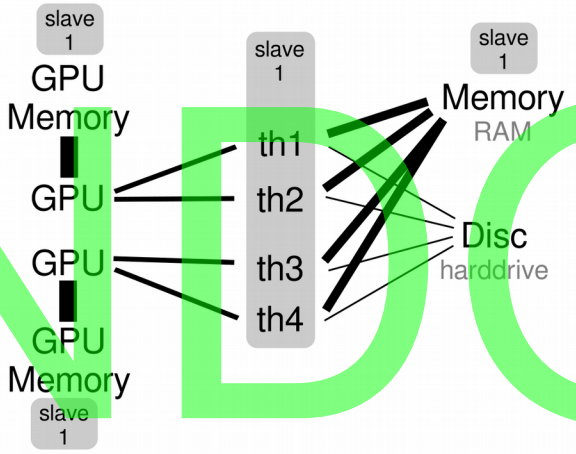
Recommendations

3 MPIs (2 slaves)
&
2 threads



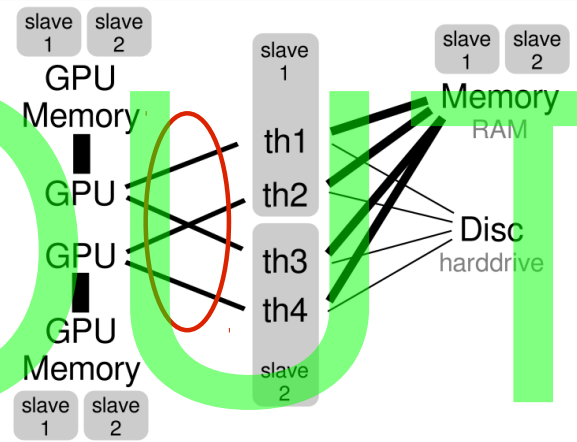
We will need to go to disc for at least some of our data each iteration

no MPI (1 slave)
&
4 threads



All data can fit into RAM and we may completely avoid **slow** disc access

3 MPI (2 slave)
&
2 threads

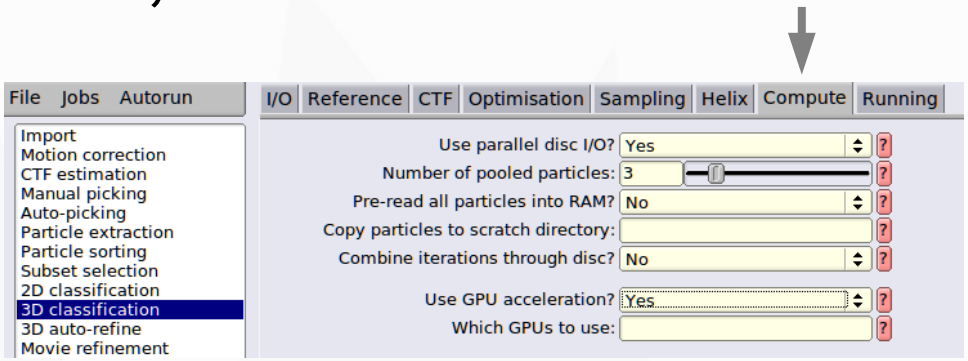


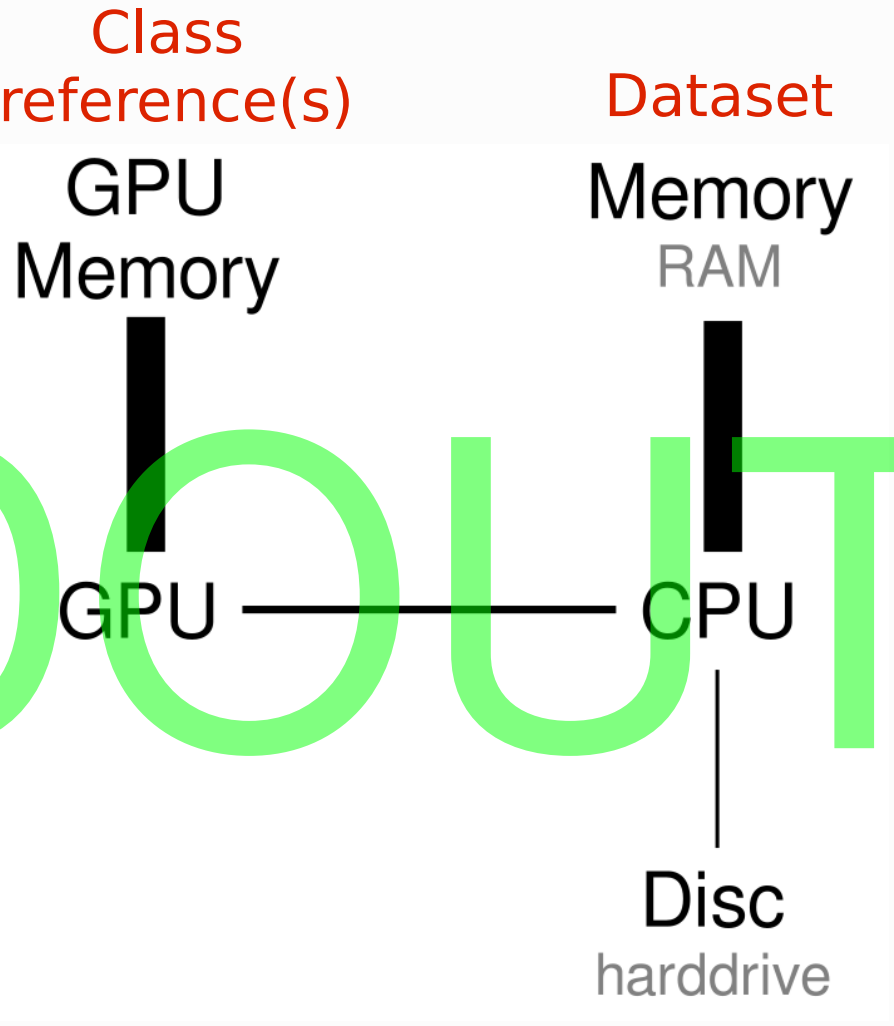
We will need to go to disc for at least some of our data each iteration,

And GPU is **split**

HANDOUT

So, choices.





HANDOUT

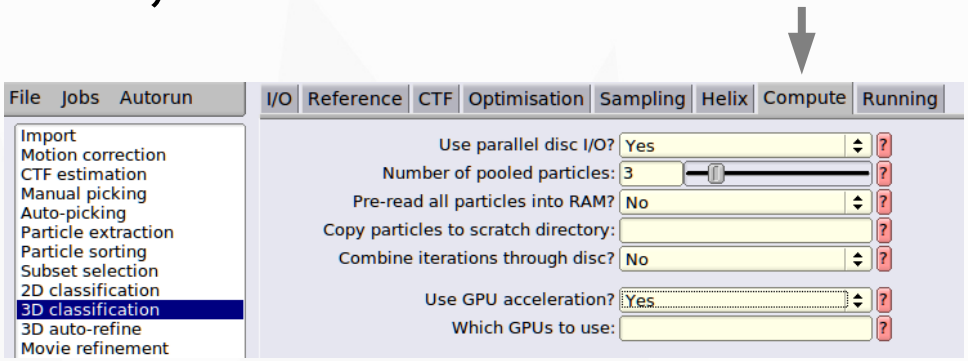
Parallel disc I/O?

"Should the slaves get data themselves?"

If they don't, they ask the master rank to send it to them.

Sending over MPI is slowish, but data might fit into RAM if you are using many ranks.

So, choices.

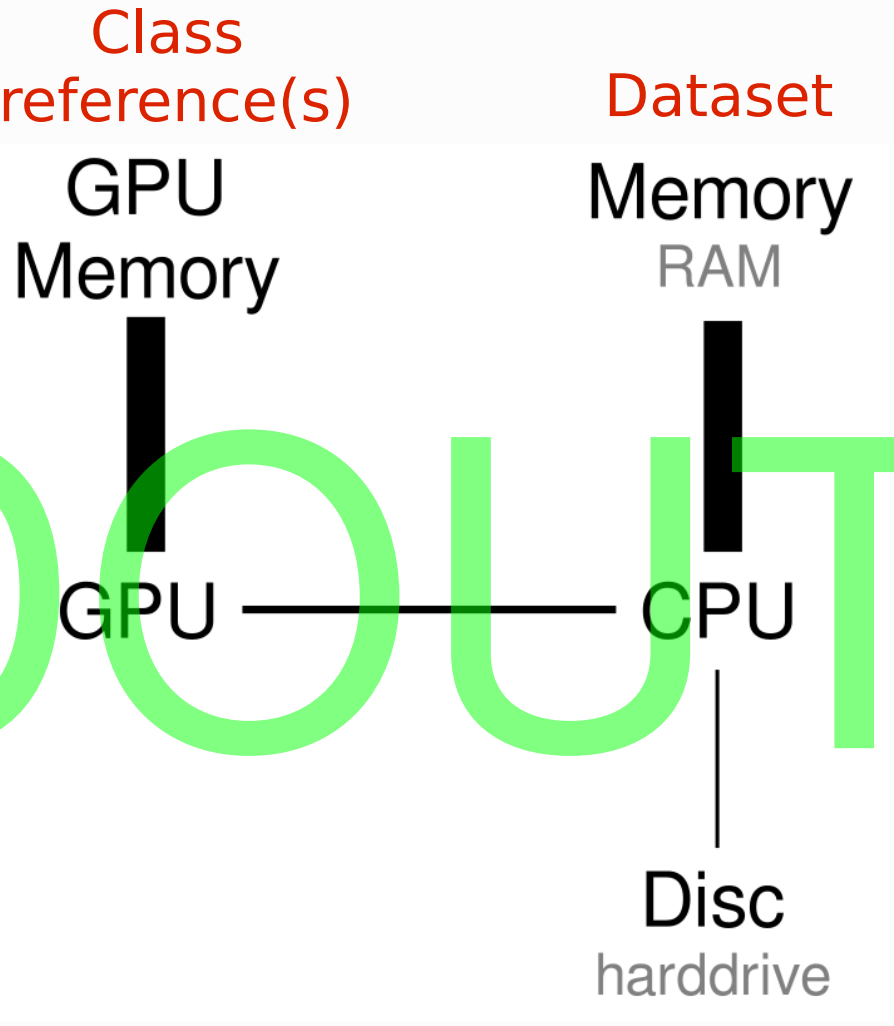


HANDOUT

Pooled particles

"How many images should each Thread process at a time"

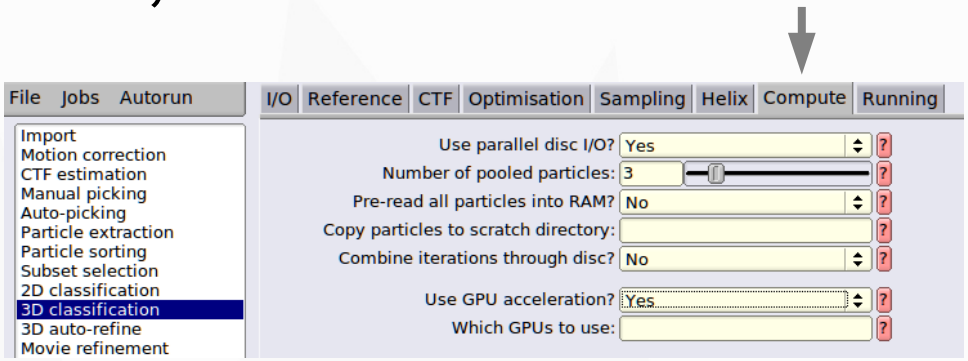
1 is tiny
10 is ok
30 is good

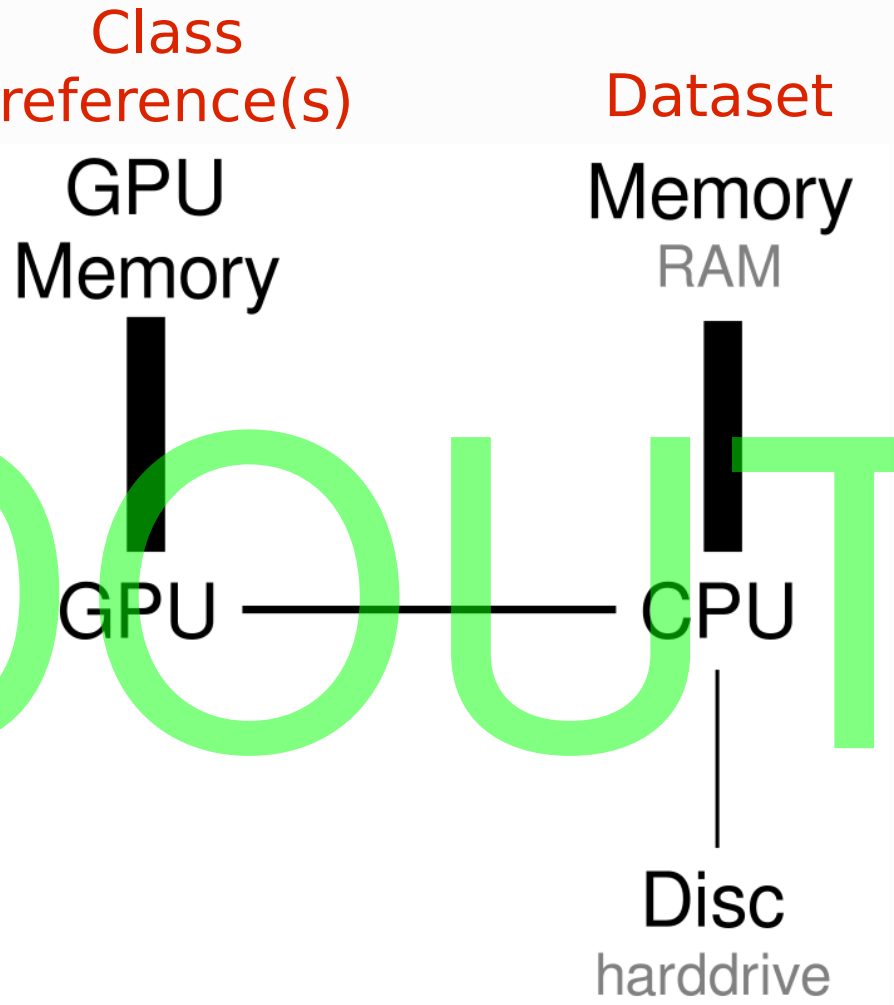


Set too big on very large resources, performance will start to drop.

16 MPIs running 8 threads each, data 105k prtcls.

So, choices.





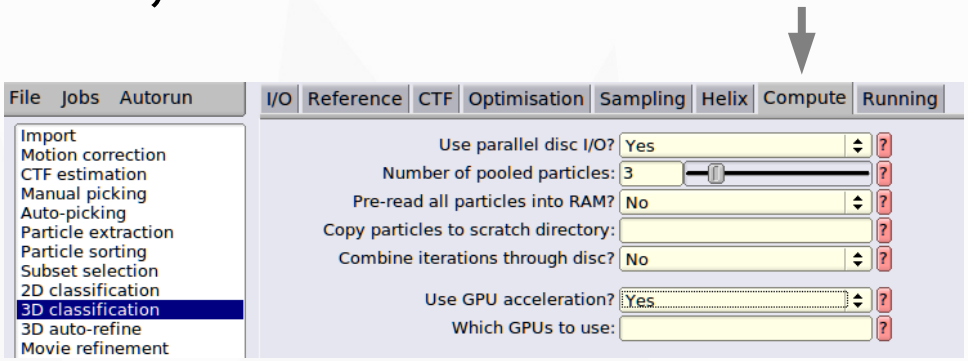
HANDOUT

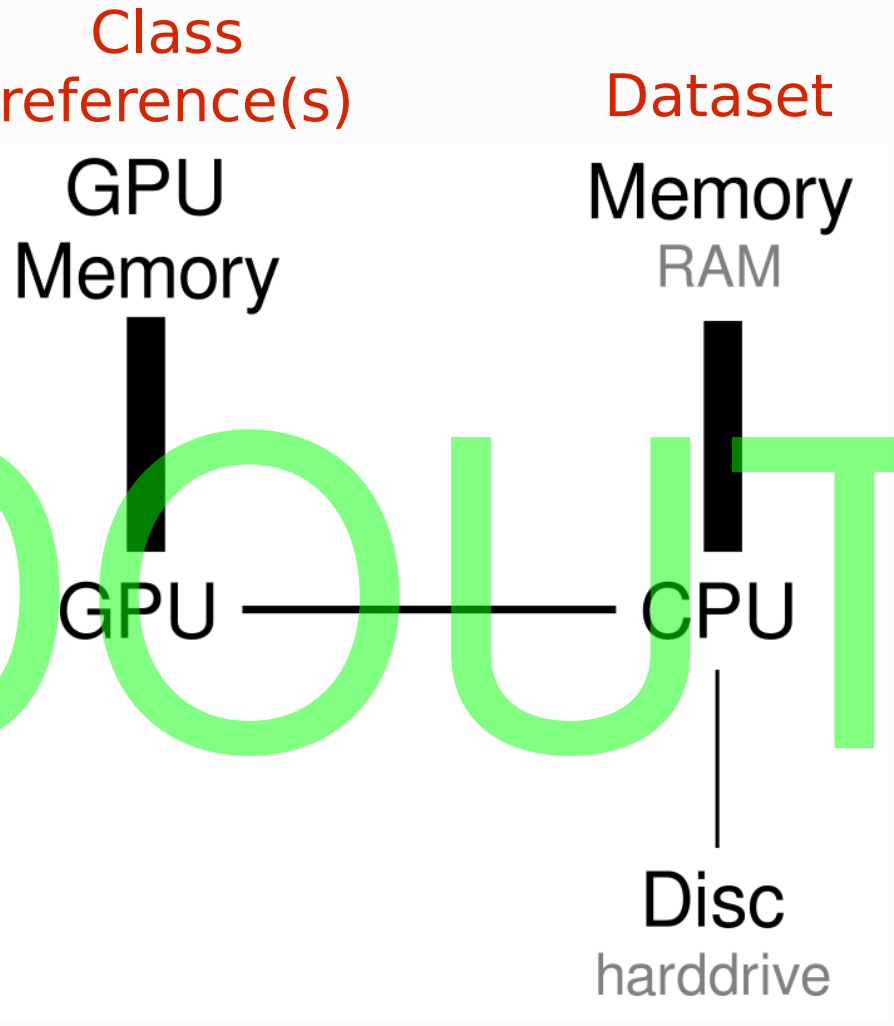
Pre-read into RAM?

Most of the time the operating system will keep data cached if it fits into RAM anyway.

Use in combination
Use parallel disc I/O? = no

So, choices.





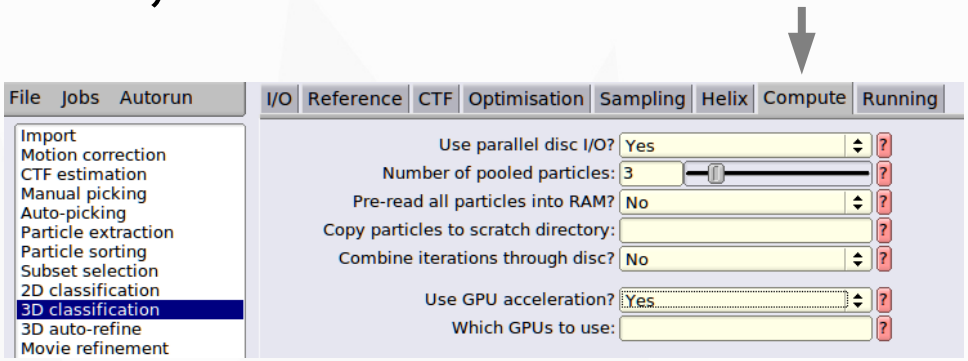
HANDOUT

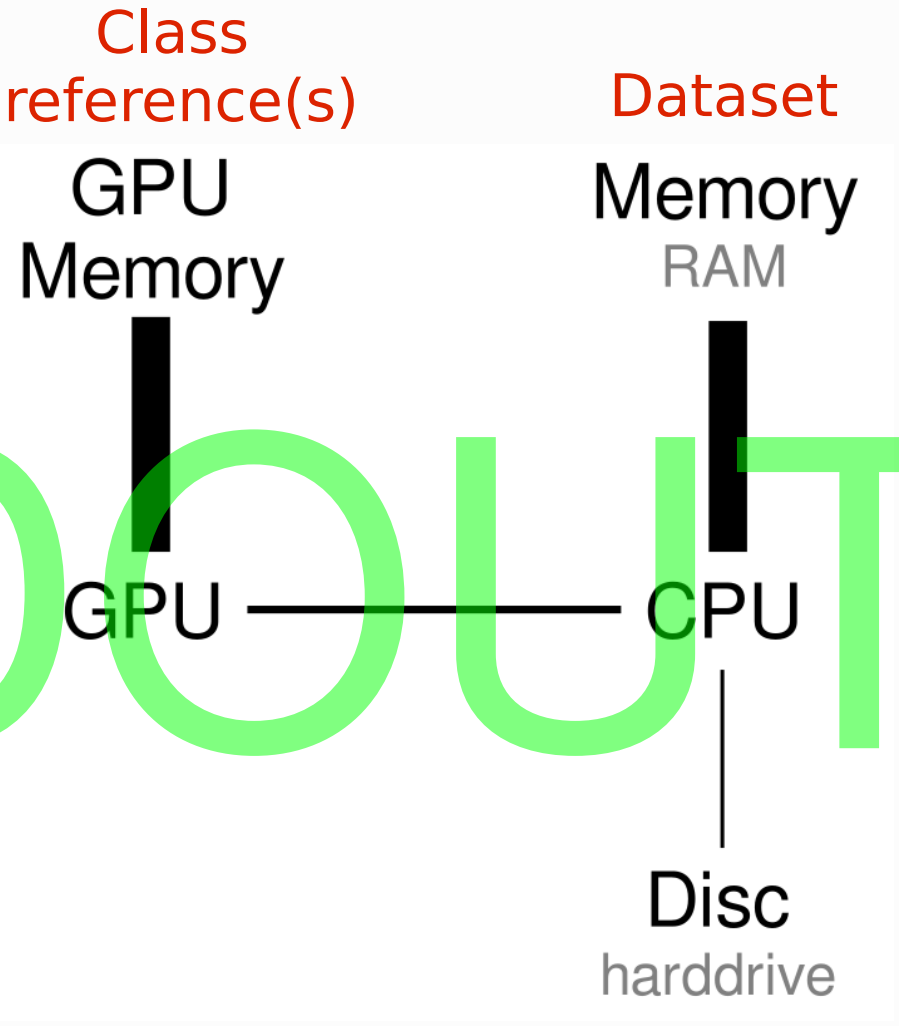
Copy to scratch?

Sometimes we have another, faster access-point, like an SSD. It is good to copy the data there, but it takes a bit of time. Worth it?

--reuse_scratch

So, choices.





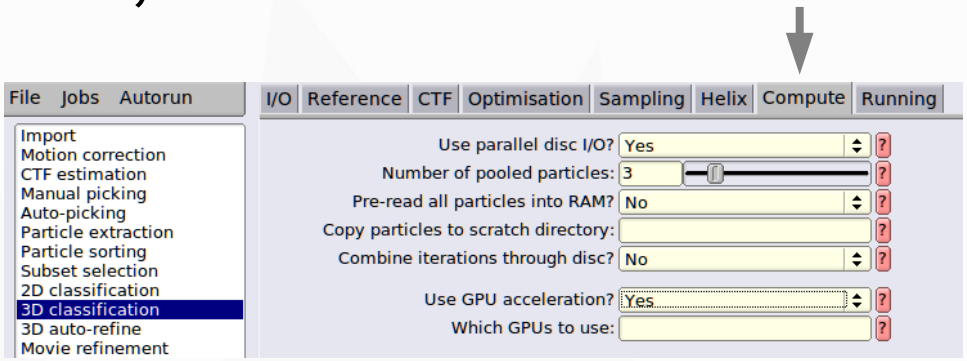
HANDOUT

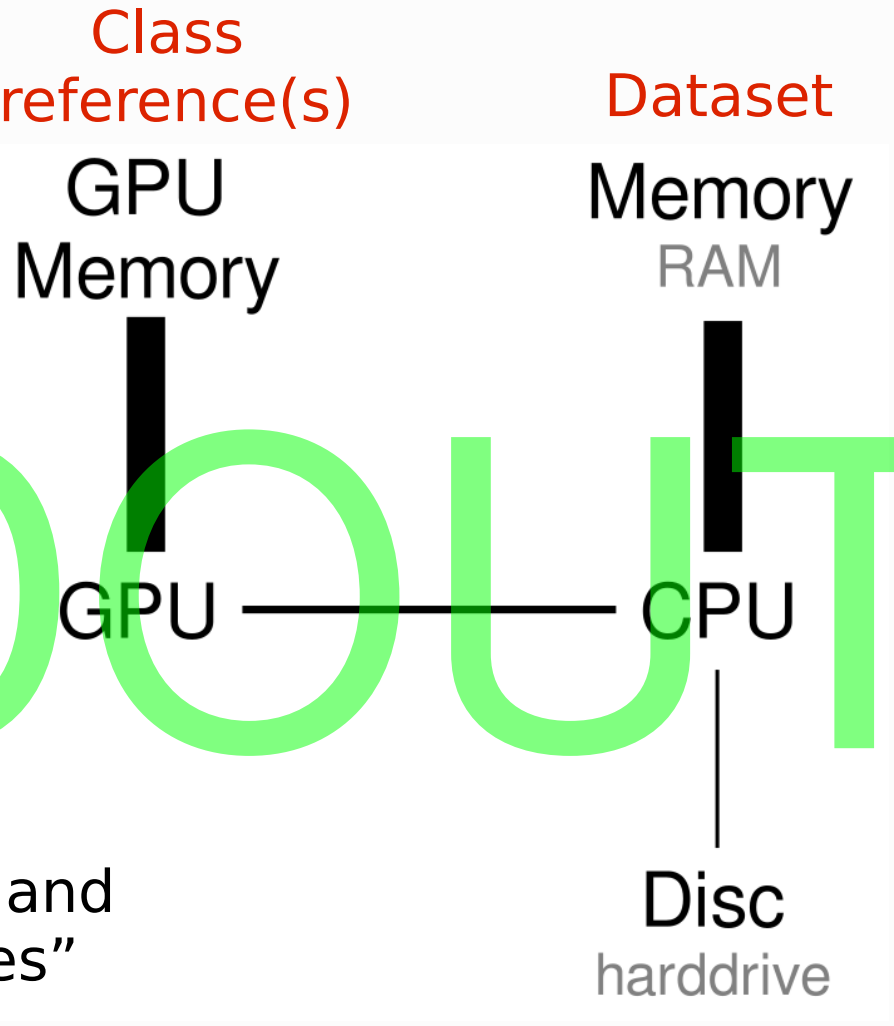
Comine through disc?

Set to “no”.

If you have network issues, try “yes”

So, choices.





HANDOUT

Which GPUs to use?

"I don't know" → **leave it blank**

"I share this system with somebody and they told me not to hog the resources"

"I want to be able to run chimera, coot, and some other stuff at the same time"

Auto-refinement **must** use MPI

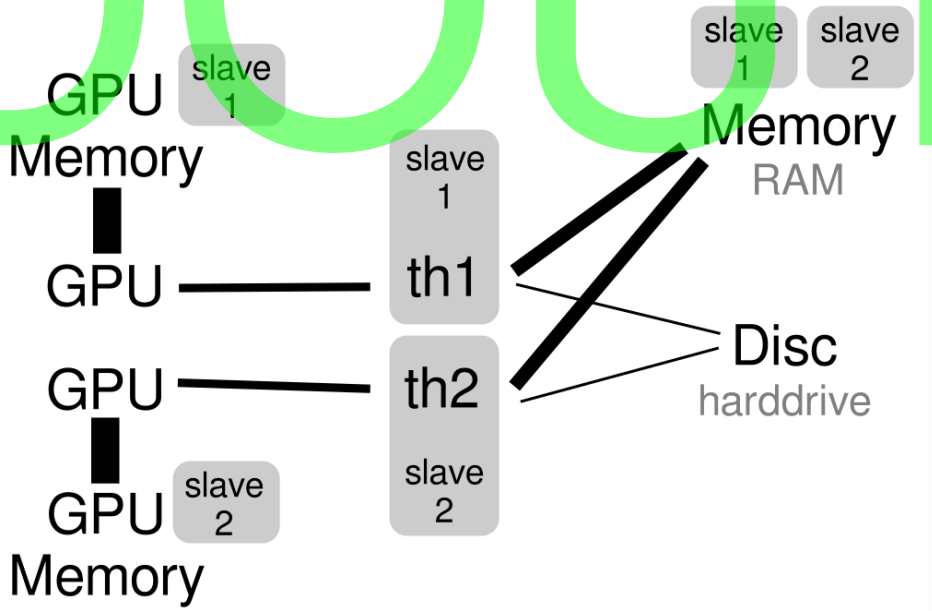
Relion splits the data in halfs, currently by dividing the data across an even number of MPI **slaves**.

Our machine has 2 GPUs

3 ranks (1 master + 2 slaves)

1 thread for simplicity

We are trying to get something like this →



HANDOUT

Warnings about split/GPU-mem during auto-refine

```
=== RELION MPI setup ===
+ Number of MPI processes             = 3
+ Master  (0) runs on host            = tcbl20
+ Slave     1 runs on host            = tcbl20
+ Slave     2 runs on host            = tcbl20
=================
uniqueHost tcbl20 has 2 ranks.
Using explicit indexing on slave 1 to assign devices  0
Thread 0 on slave 1 mapped to device 0
Using explicit indexing on slave 2 to assign devices  0
Thread 0 on slave 2 mapped to device 0
Device 0 on tcbl20 is split between 2 slaves


                         ***WARNING***

With the current settings and hardware, you will be able to
use an estimated image-size of 351 pixels during the last iteration...

...but your input box-size (image_size) is however 360. This means that
you will likely run out of memory on the GPU(s), and will have to then re-start
from the last completed iteration (i.e. continue from it) without the use of GPUs.

You are also splitting at least one GPU between two or more mpi-slaves, which
might be the limiting factor, since each mpi-slave that shares a GPU increases the
use of memory. In this case we recommend running a single mpi-slave per GPU, which
will still yield good performance and possibly a more stable execution.


 Running CPU instructions in double precision.
 Estimating initial noise spectra
   1/   1 sec ..........................................................~~(,_,">
```

Machine has 2 GPUs
(GTX 1070 8GB)

--gpu 0

You may (or may not) see this

HANDOUT

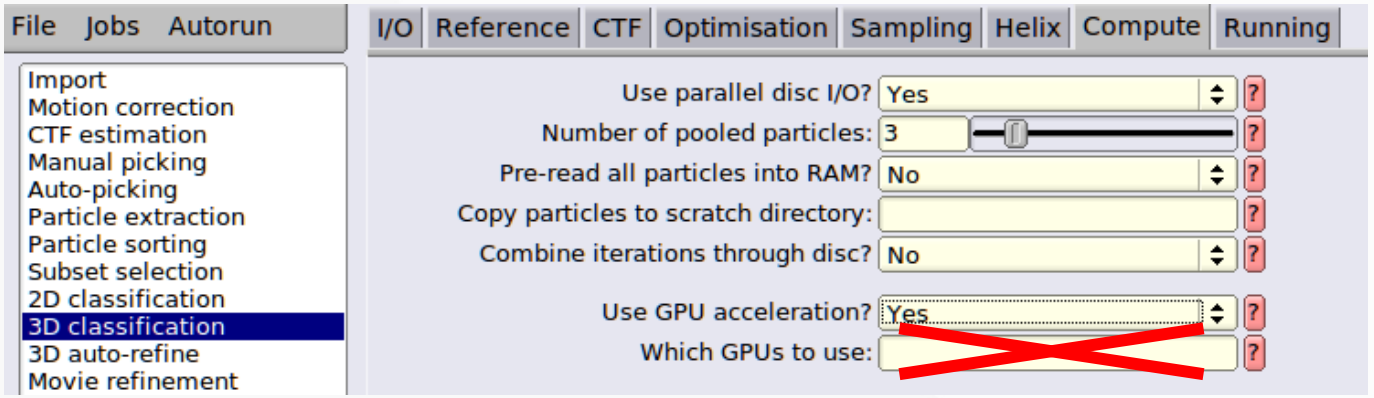
Warnings about GPU-mem during auto-refine

Machine has 2 GPUs
(GTX 1070 8GB)

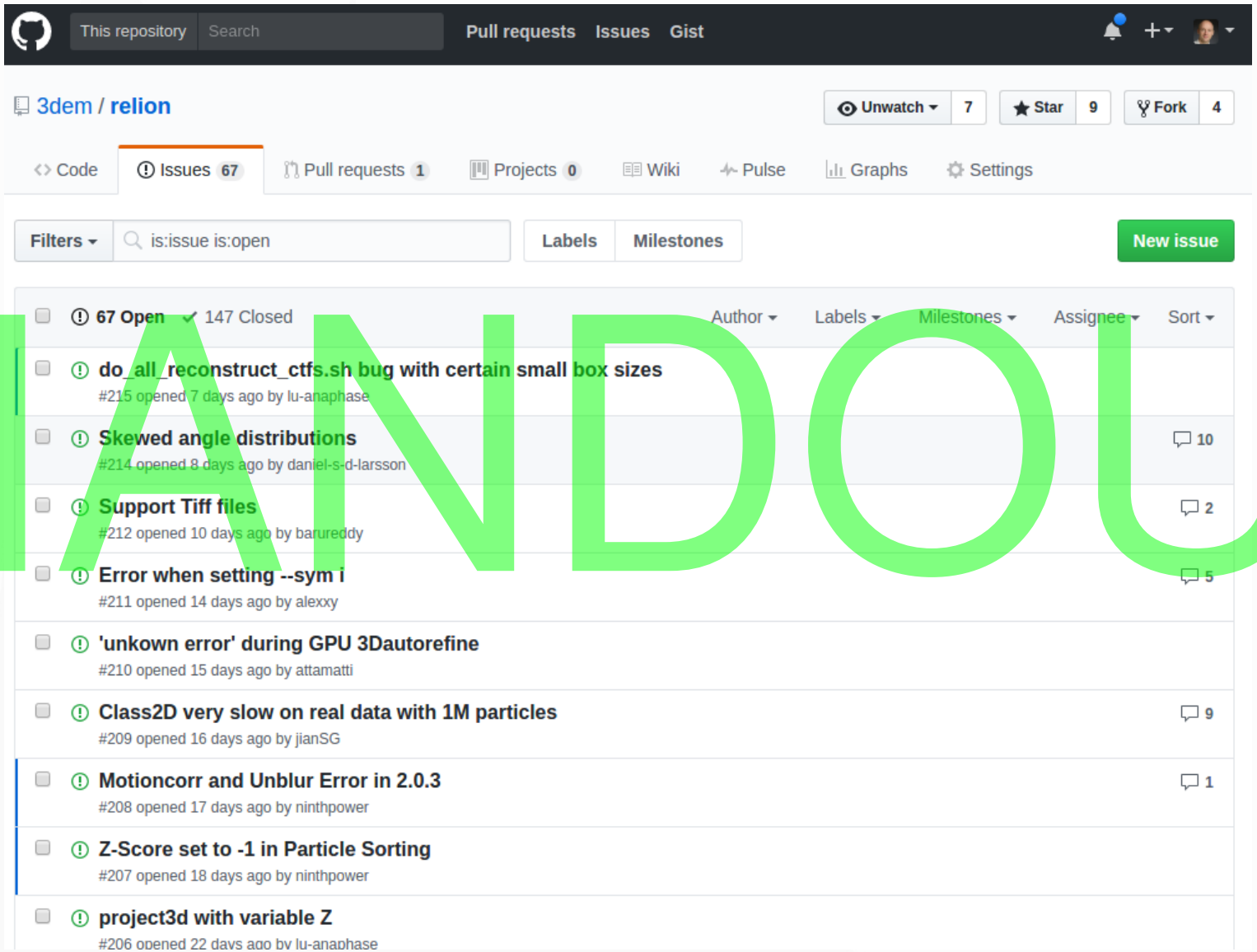
--gpu

```
 === RELION MPI setup ===
 + Number of MPI processes             = 3
 + Master  (0) runs on host            = tcbl20
 + Slave     1 runs on host            = tcbl20
 =================
 + Slave     2 runs on host            = tcbl20
 uniqueHost tcbl20 has 2 ranks.
GPU-ids not specified for this rank, threads will automatically be mapped to available devices.
 Thread 0 on slave 1 mapped to device 0
GPU-ids not specified for this rank, threads will automatically be mapped to available devices.
 Thread 0 on slave 2 mapped to device 1
 Running CPU instructions in double precision.
 Estimating initial noise spectra
   1/   1 sec ..........................................................~~(,_,">
```

HANDOUT



Github is for errors/bugs/questions/suggestions



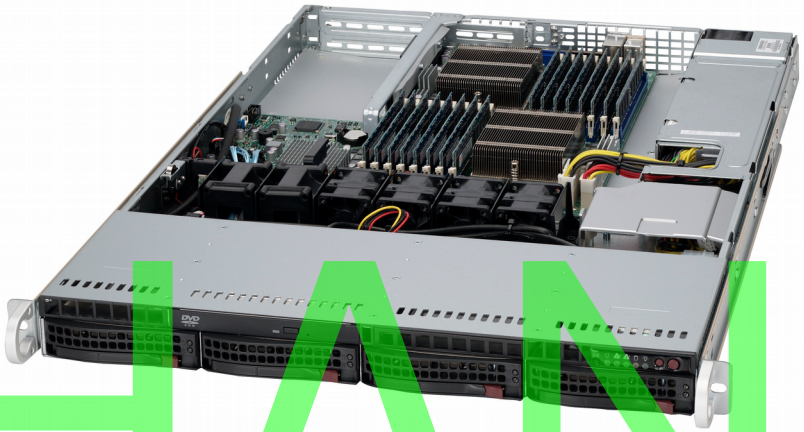
HANDOUT

HANDOUT

Clusters and Desktops (running Relion)

Clusters are collection of compute units (nodes)

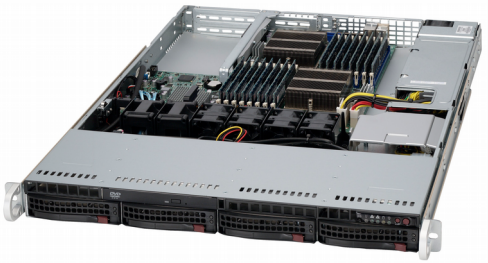
slim node (blade) with 56 CPU cores



HANDOUT

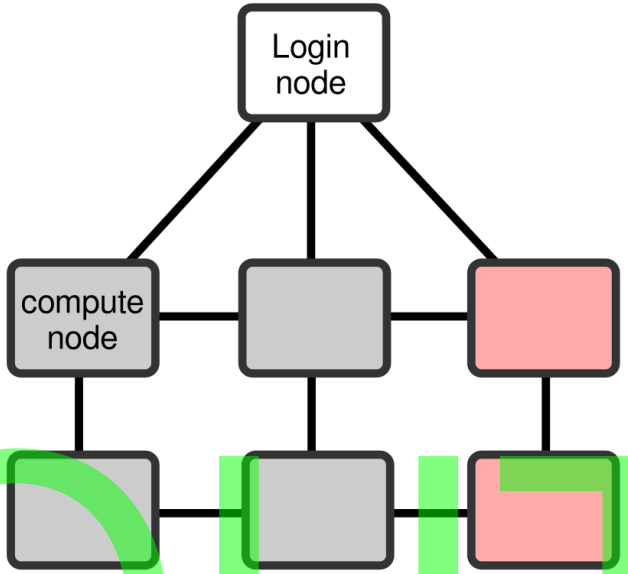
Clusters use queues

node



cluster

cluster

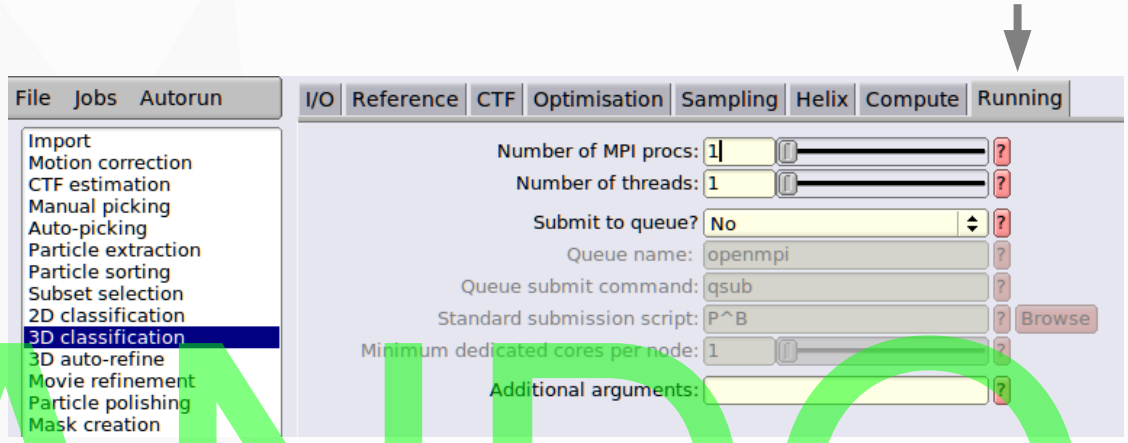


HANDOUT

On the login-node you submit jobs to a queue, which schedules and runs your job(s).

```
JOBID     PARTITION     NAME       USER      STATUS    TIME  NODES NODELIST(REASON)
3739      tcb           cpc_work   magnusl   PD        0:00      1 (Dependency)
3748      tcb           cpc_work   magnusl   PD        0:00      1 (Dependency)
3749      tcb           cpc_work   magnusl   PD        0:00      1 (Dependency)
3750      tcb           cpc_work   magnusl   PD        0:00      1 (Dependency)
3712      tcb           cpc_work   magnusl   R      9:30:07      1 gpu04
3713      tcb           cpc_work   magnusl   R      9:29:59      1 gpu06
3714      tcb           cpc_work   magnusl   R     23:18:55      1 gpu08
3737      tcb           M11run8_   almaan    R     10:17:41      1 gpu02
3738      tcb           M11run9_   almaan    R     10:17:35      1 gpu07
3740      tcb           cpc_work   magnusl   R      6:32:39      1 gpu09
3747      tcb           cpc_work   magnusl   R      2:33:15      1 gpu03
3752      tcb           bash       wennberg  R      2:09:36      1 gpu10
```

Through the Relion GUI on the login node



HANDOUT

The queue submit command depends on the queue-system the cluster uses.

sbatch (**slurm**), qsub (**PBS**), bsub (**LSF**)

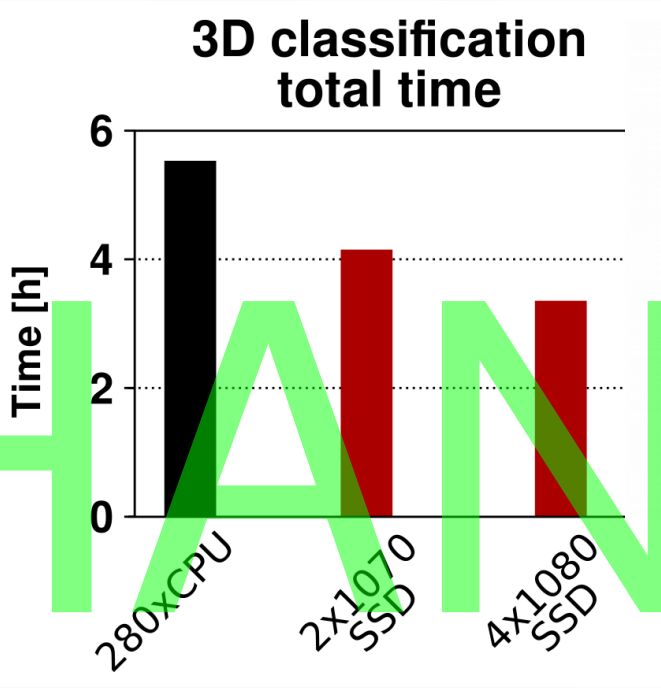
So does the form and content of the job-script. **Relion creates a script and submits it to the queue,** but you can define a template job-script that relion augments with the setting in the GUI.

Desktops can run Relion as well.

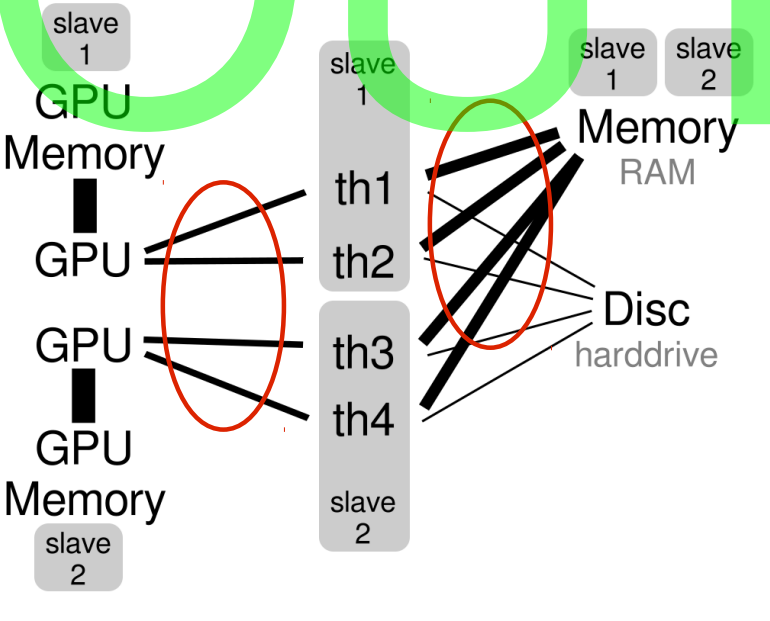
HANDOUT

High-end (really fast, not quiet, and very warm)
4 x GTX1080

Desktops are difficult to make powerful



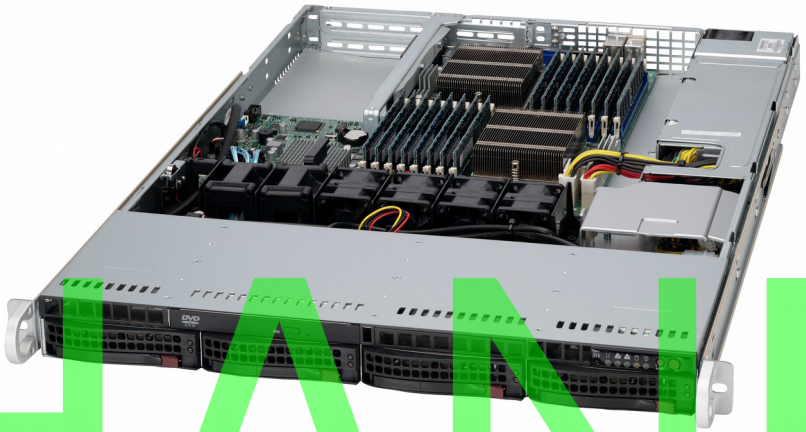
HANDOUT

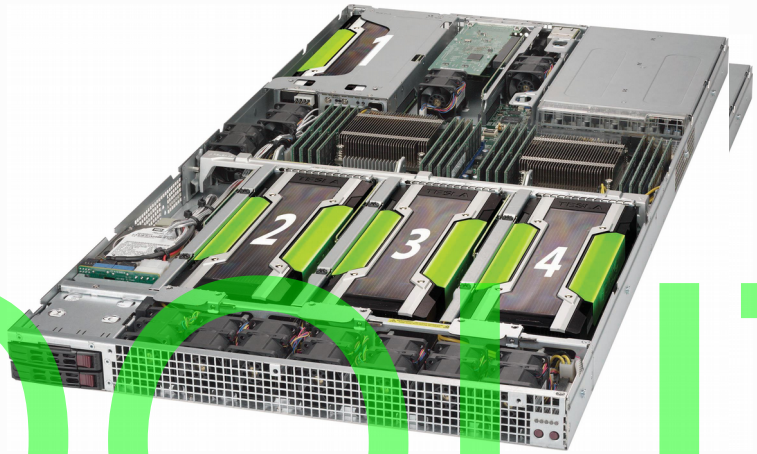


The GPUs are simply starting to complete work faster than the **CPU can create it,** and **Memory/bandwidth deliver it.**

Hardware

Slim nodes (blades) with 4 x 1080





HANDOUT

Thanks!

HANDOUT

Qs?