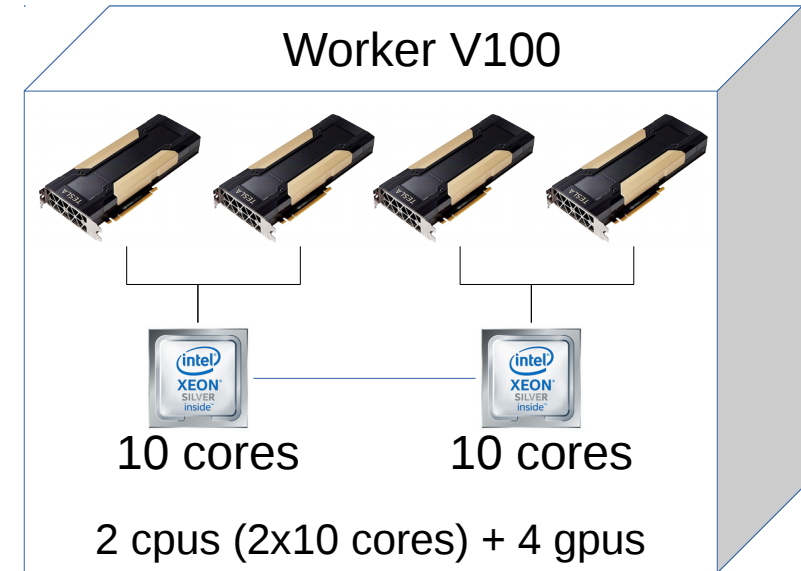
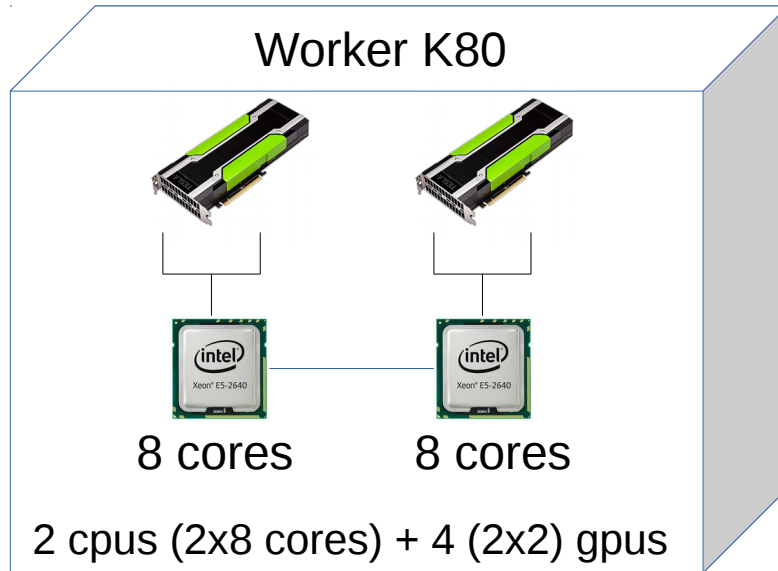


GPU Status at CC-IN2P3

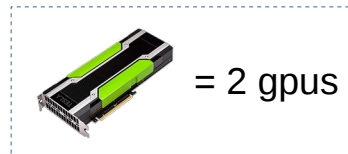
September 21, 2020

- ▶ **Hardware**
- ▶ **Softwares environment**
- ▶ **Usage**
- ▶ **Conclusion & perspectives**



K80 farm

- ▶ **10 workers:**
 - 2 Intel(R) Xeon(R) CPU E5-2640 (8 cores)
 - 128GB RAM
 - SSD disk
- ▶ - 2 Nvidia **Tesla K80** cards
(4 GPU Nvidia GK210, 12 Go DDR5 each)
- ▶ **40 GPU au total**
- ▶ Network
 - Infiniband interconnection

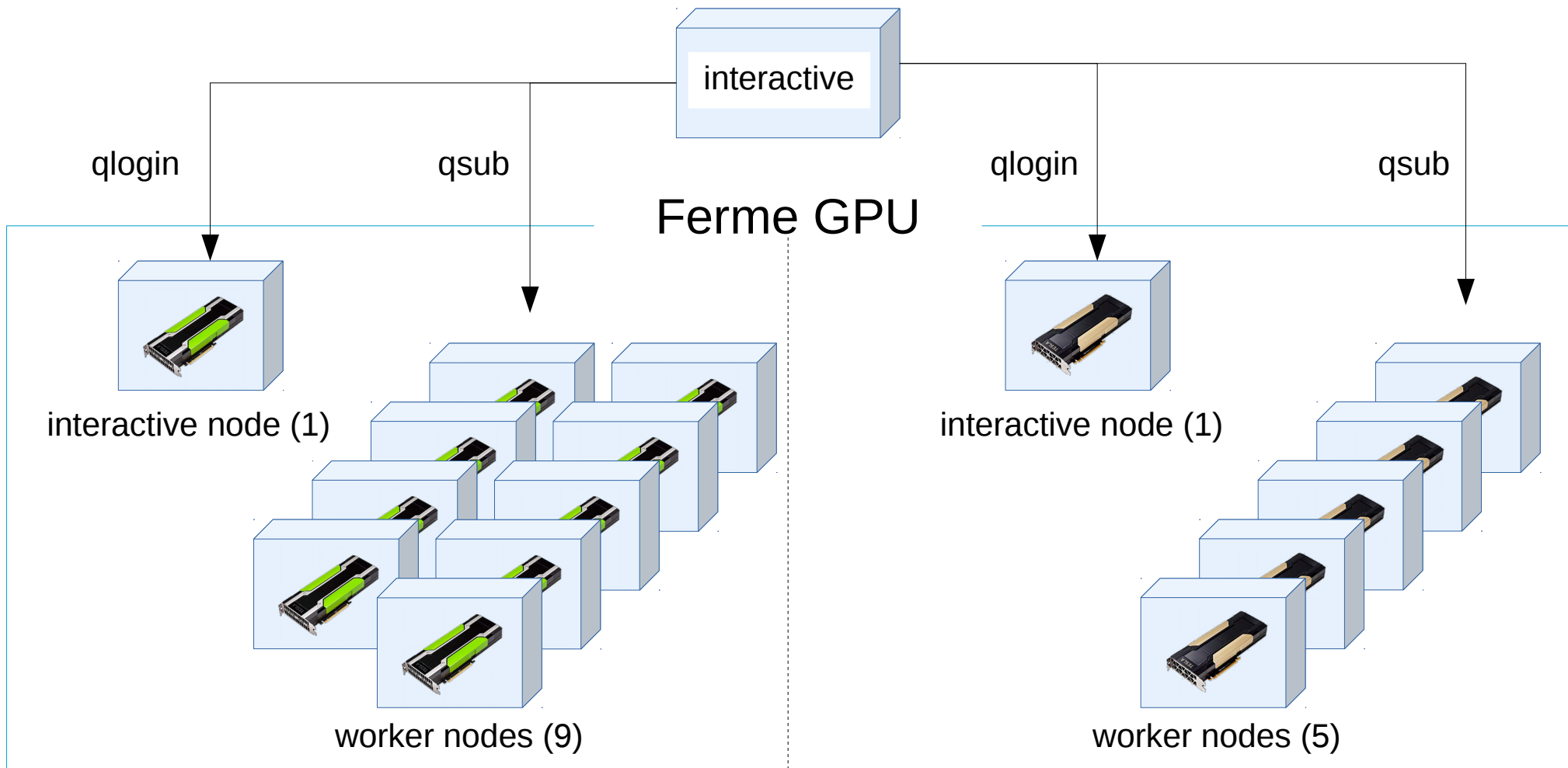


V100 farm

- ▶ **6 workers:**
 - 2 Intel(R) Xeon(R) Silver 4114 (10 cores)
 - 192GB RAM
 - SSD M2 disk
- ▶ - 4 Nvidia **Tesla V100** 32GB PCIe cards
- ▶ **24 GPU (total)**
- ▶ Network
 - NO Infiniband interconnection!



- ▶ First, request an access (authorisation required)
- ▶ Classical submission on Grid Engine (qsub) in multicore or parallel mode (https://doc.cc.in2p3.fr/jobs_gpu)
- ▶ Accounting rules still to be defined (K80 vs V100)



Current gpu driver version

440.64

Available libraries



OpenCL



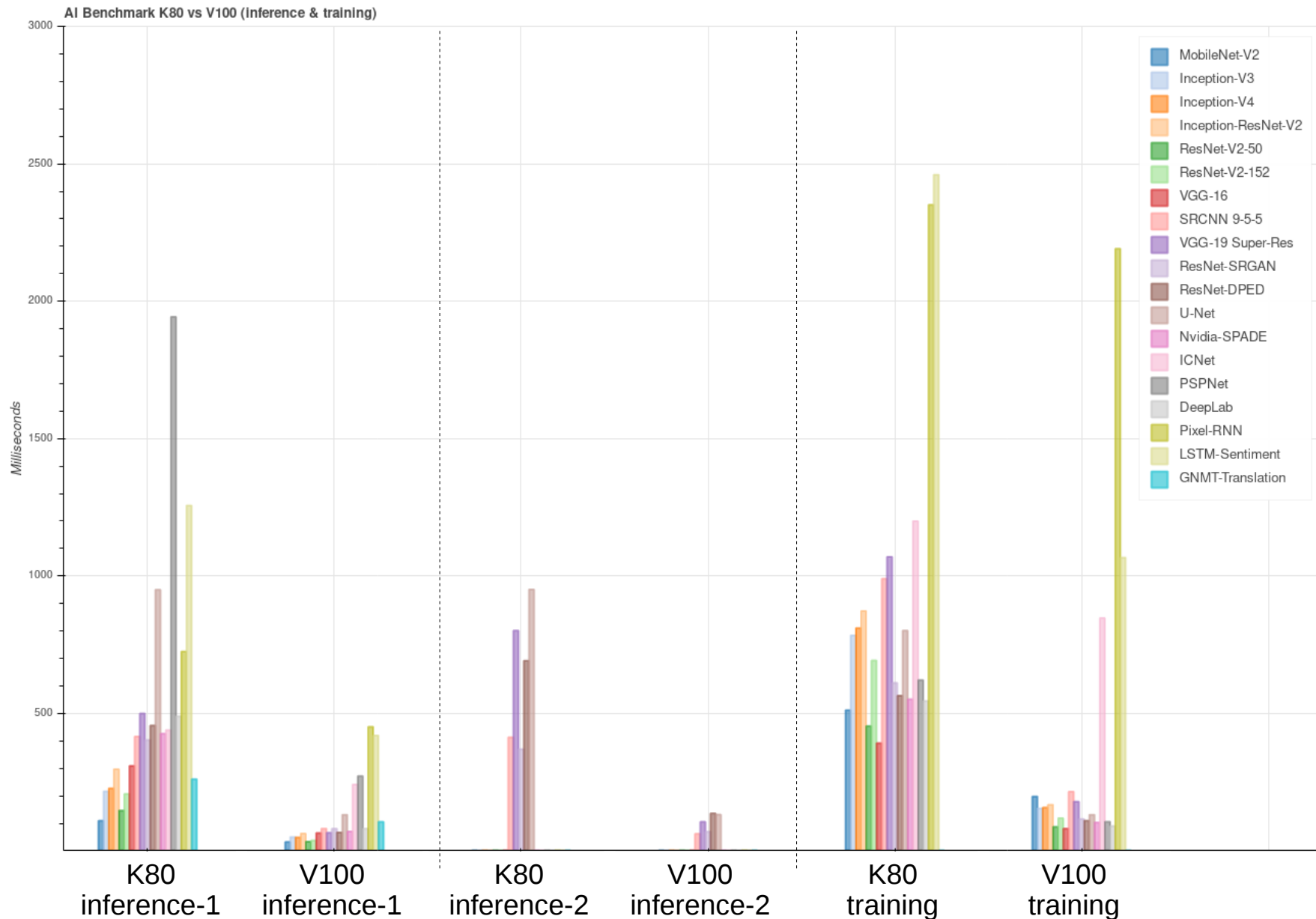
OPEN MPI

Customized softwares

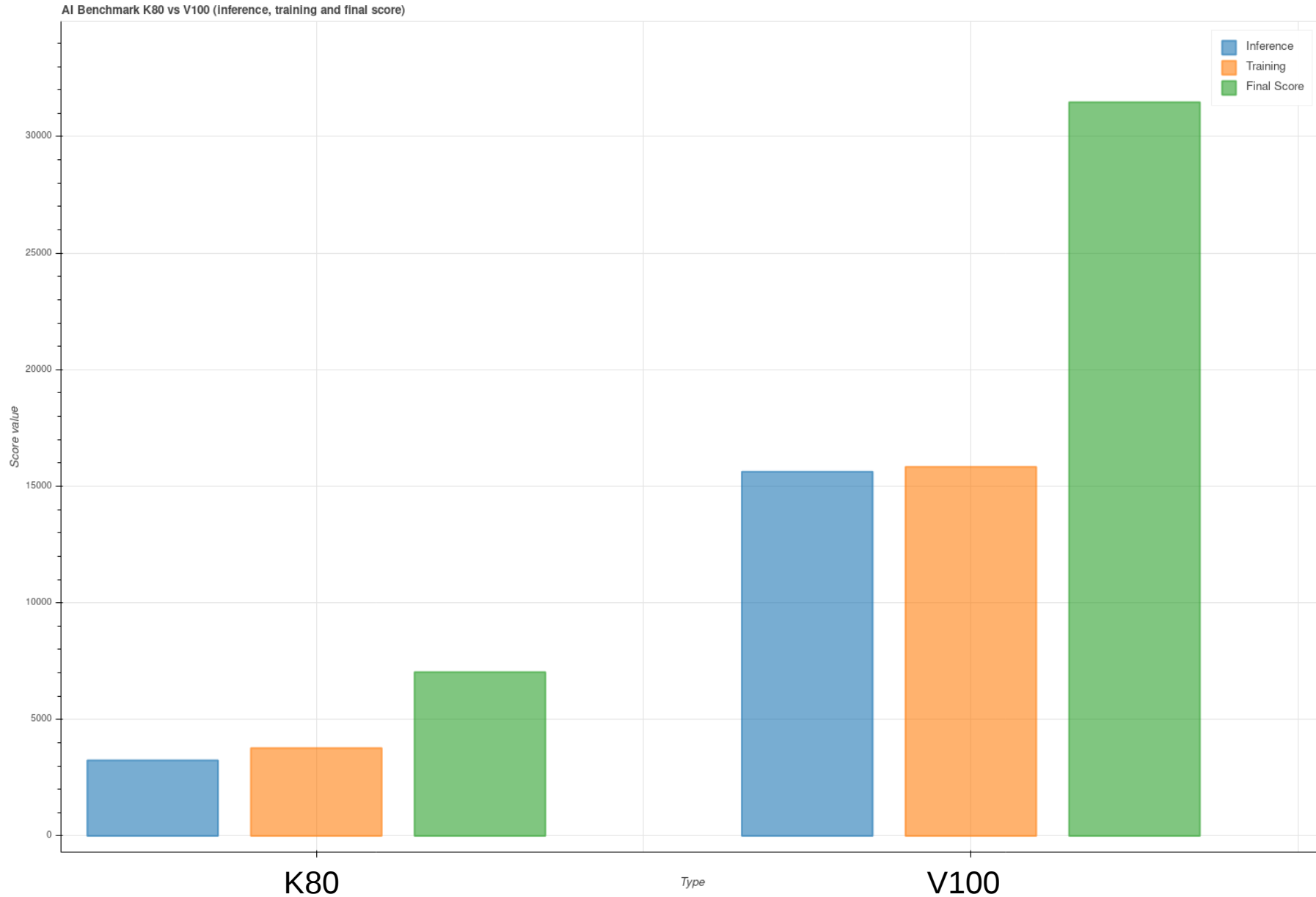
- ▶ Customized softwares provided thanks to **Singularity** containers



- AI Benchmark (<https://pypi.org/project/ai-benchmark/#modal-close>)

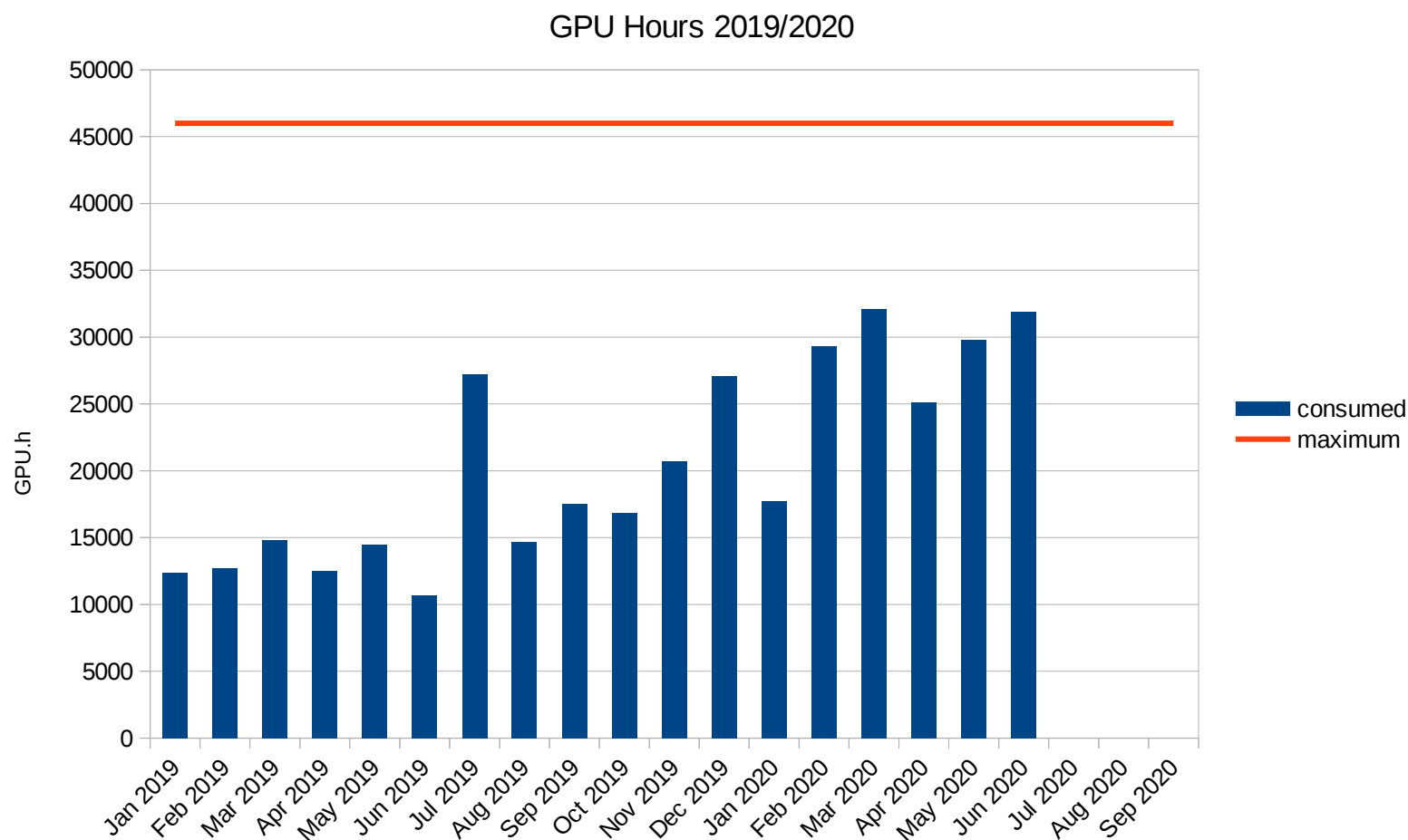


- AI Benchmark (<https://pypi.org/project/ai-benchmark/#modal-close>)



- GPU requests for 2020 in GPU-hours
 - K80 ~351k hours & V100 ~210k hours → 561k hours available
- Resources requests (some are not yet validated, rough numbers)
 - IN2P3 experiments asks for ~140k hours
 - ATLAS (35k) and LSST (87k) being the top 2 requestors
 - Other experiments (no-IN2P3) asks for more than 1500k hours
- Need to extend the current farm if we want to cope with all the requests → internal discussion on-going

- ~x2 between 2019 and 2020
- Not that far from limits



- Current GPU farms overall well used
 - Better usage: more and more multi-GPU and multi-nodes jobs
 - Still « holes » though, especially in the K80 farm
- Some basic software environments provided
 - Conda & Python + TensorFlow & PyTorch + Singularity images
 - Do you need something else ? Please let us know
- Current discussion about extending the GPU farm
 - Come back to us if you would like to use some specific one

- ▶ Interactive Worker and Batch Worker nodes are the same in terms of architecture (same cpus, gpus, memory)
- ▶ Batch scheduler provides access to Interactive Worker nodes in shell mode



Interactive Worker access (qlogin)



Specifying the use of sps resources

Nb of gpus <1 - 4>

Specifying K80 or V100 farm

```
qlogin -l GPU=1,sps=1,GPUtype=

|      |
|------|
| K80  |
| V100 |

 -q mc_gpu_interactive -pe multicores_gpu 4
```

Custom parameters

Queue

Nb of cpu cores

- ▶ Batch scheduler provides access on Batch Worker nodes to run a program



Batch submission (qsub)



Specifying the use of sps resources

Nb of gpus <1 - 4>

Specifying K80 or V100 farm

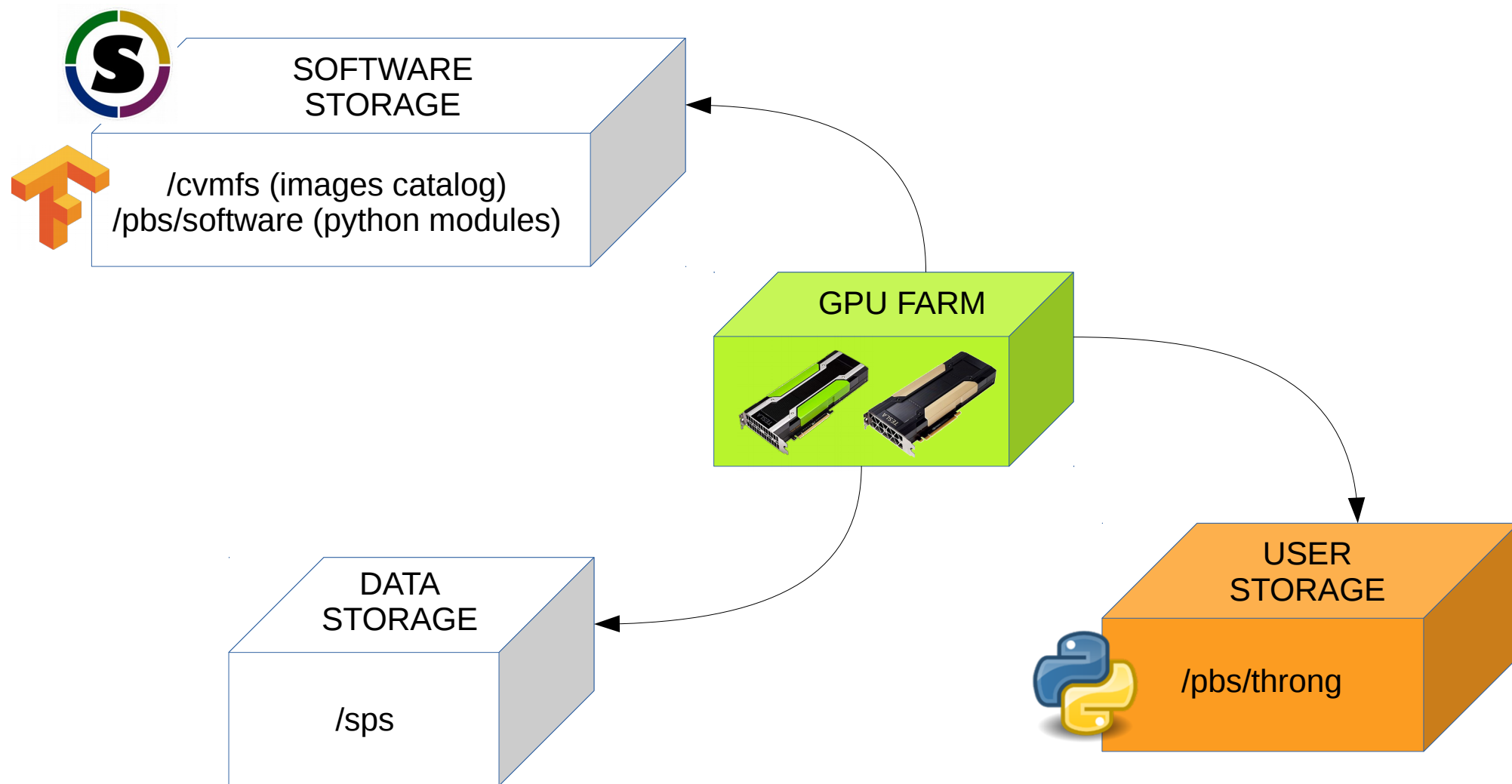
qsub -l GPU=1,sps=1,GPUtype=

K80
V100

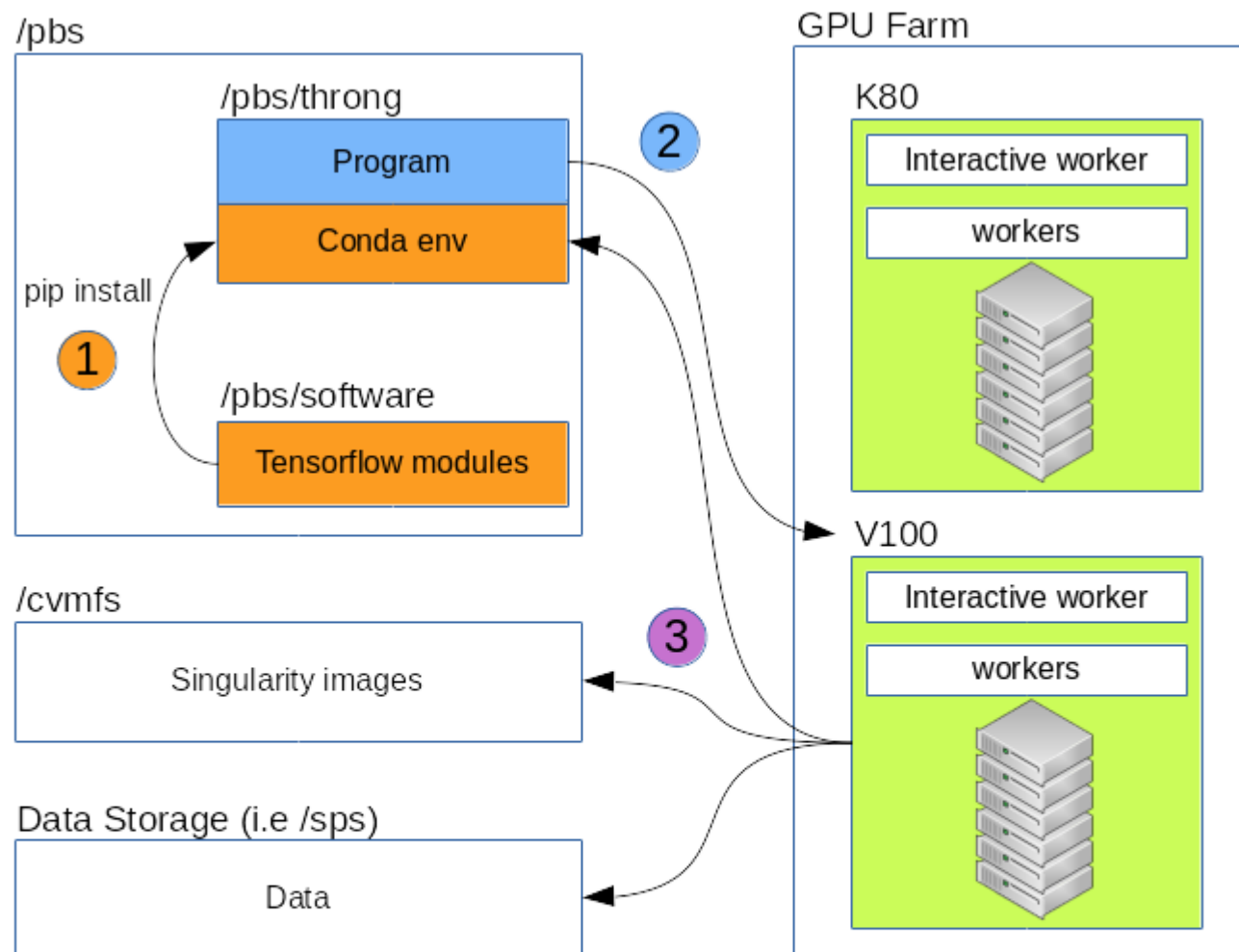
[options] <file_to_execute>

Custom parameters

- Worker nodes can access different types of storages



Workflow example



- 1** Install the tensorflow module you need depending on your python environment (python 2.7 or python 3.6) and the GPU type you want (K80 or V100)
- 2** Submit your code to the GPU farm, specifying which image you want to run it, and your python environment
- 3** The GPU farm computes your code through the specified environment

Questions ?

Bertrand.rigaud@cc.in2p3.fr

More questions? Ask us :
[OTRS ticketing System](#)

Thank you for your attention.