UNIVERSITAT JAUME·I

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

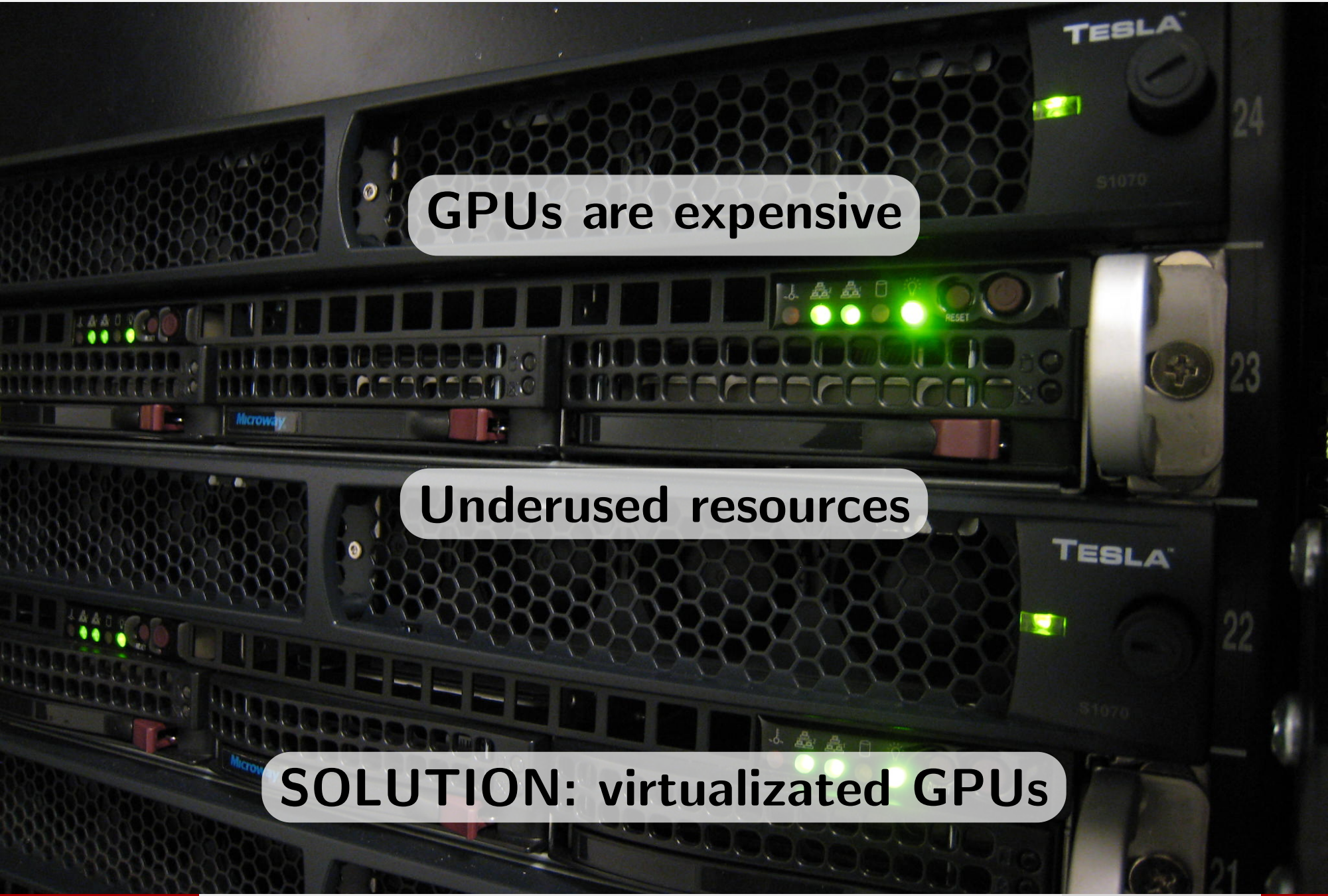
Extending Slurm with Support for Remote GPU Virtualization

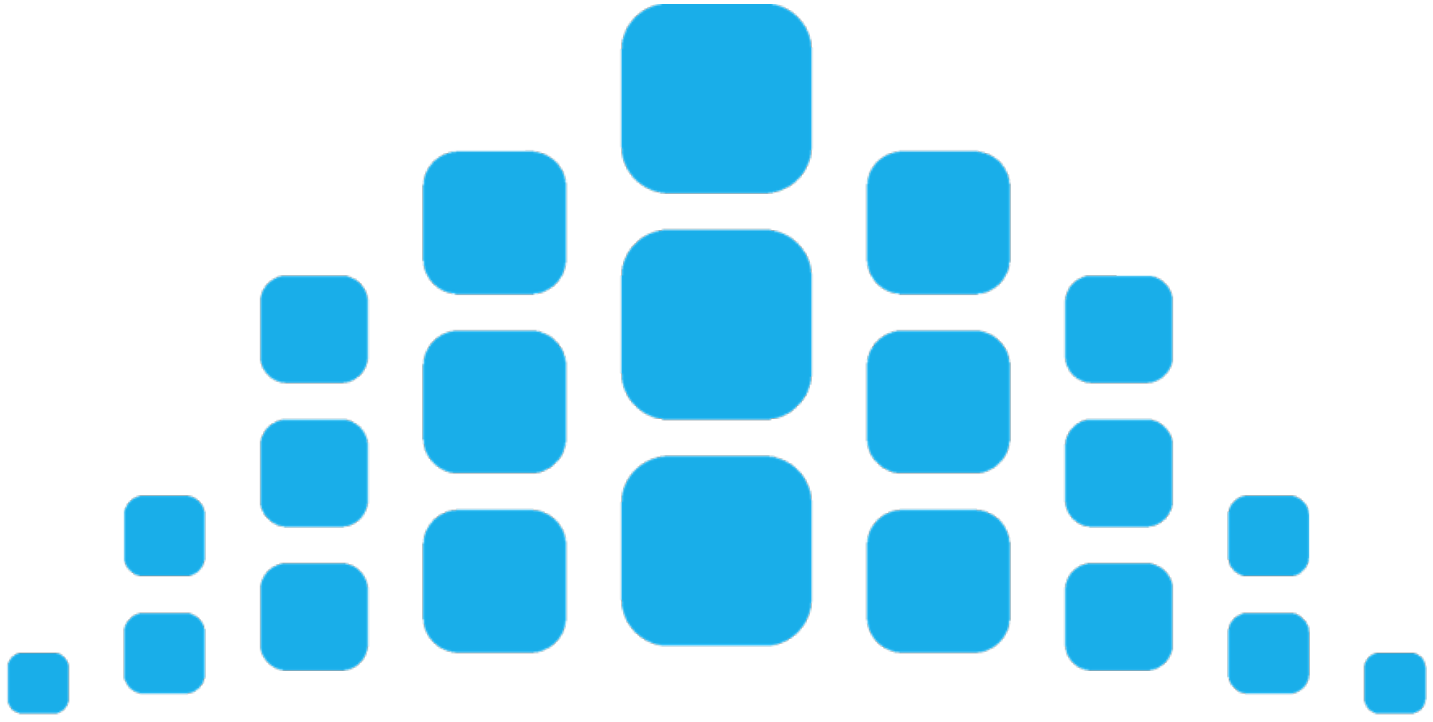
Sergio Iserte, Adrián Castelló, Rafael Mayo,
Enrique S. Quintana-Ortí, Federico Silla, Jose Duato

Slurm User Group Meeting 2014

23 September 2014, Lugano (Switzerland)

Introduction

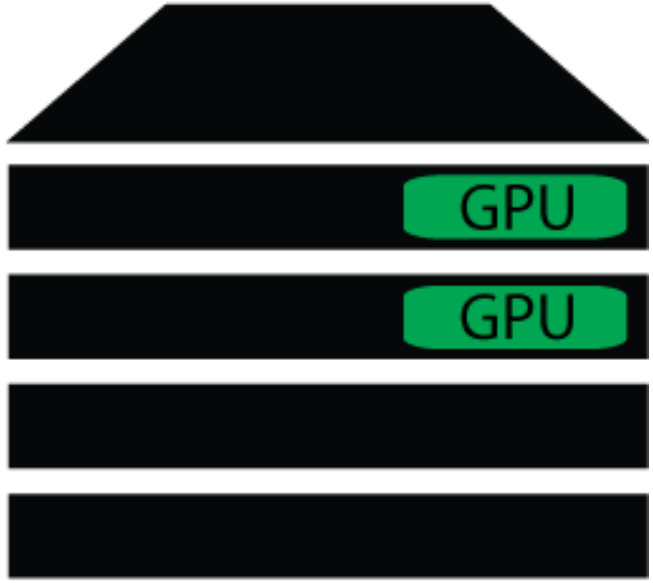




slurm

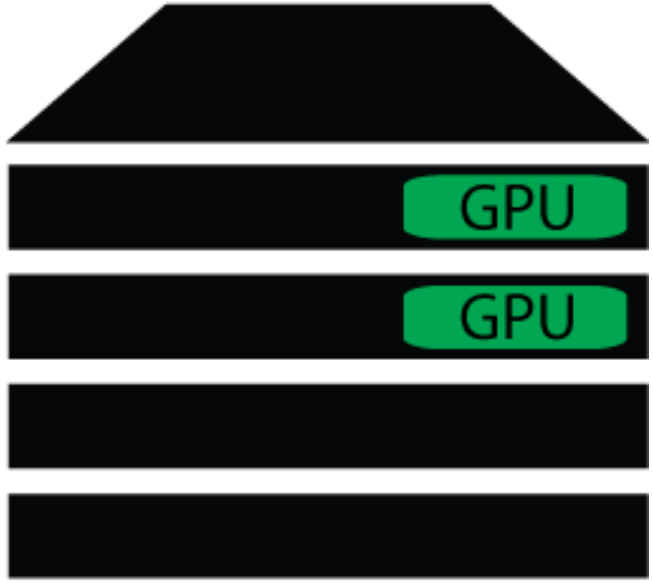
workload manager

Resources Management

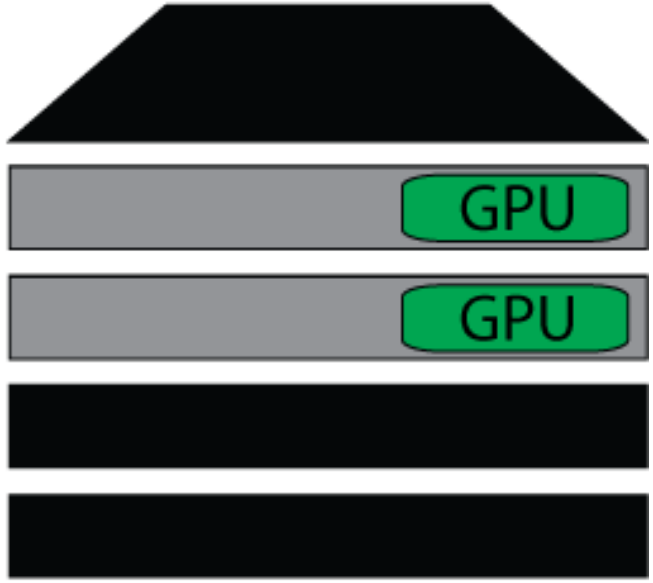


Resources Management

2 nodes

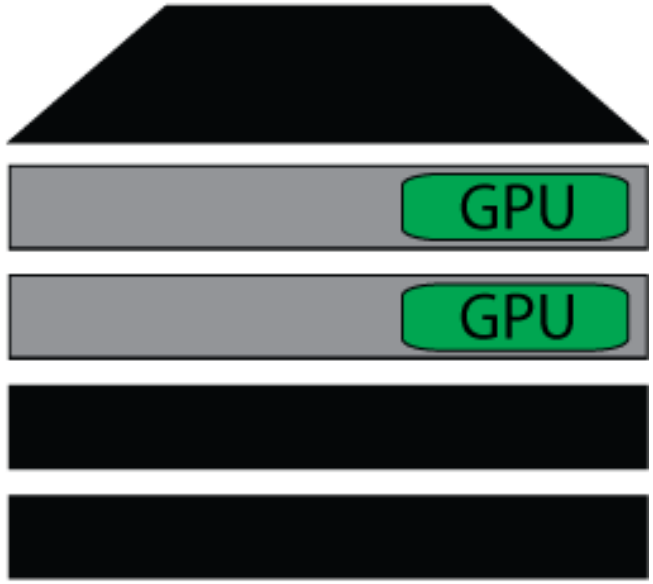


Resources Management



Resources Management

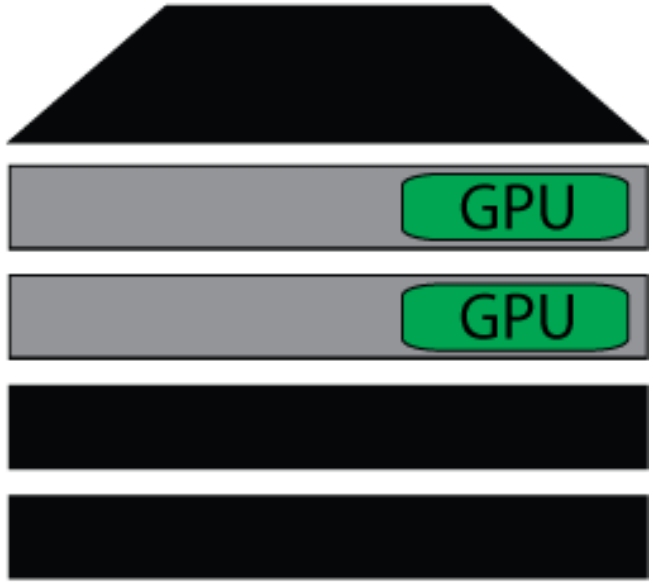
2 nodes
2 GPUs



Resources Management

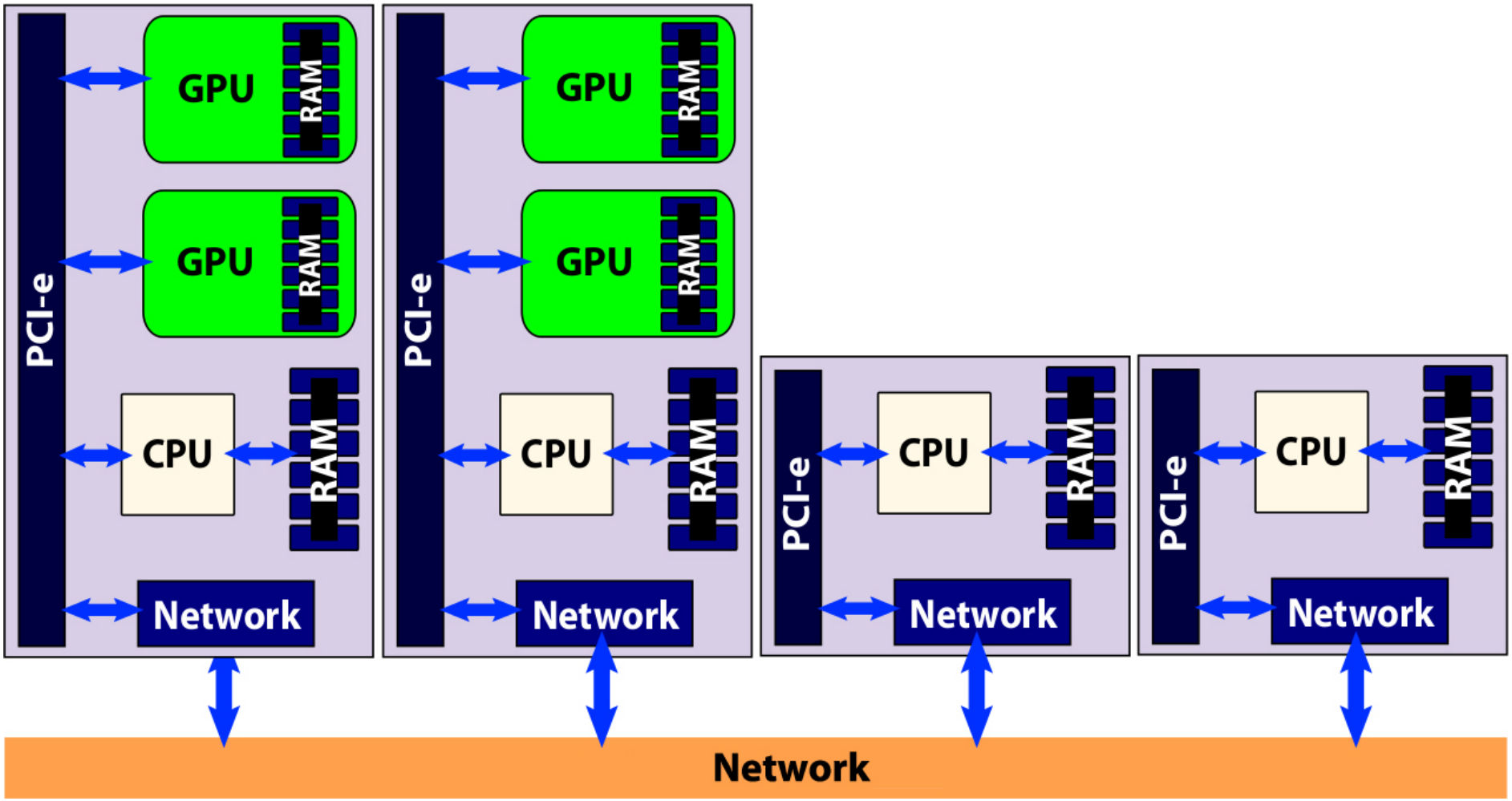
2 nodes
2 GPUs

???

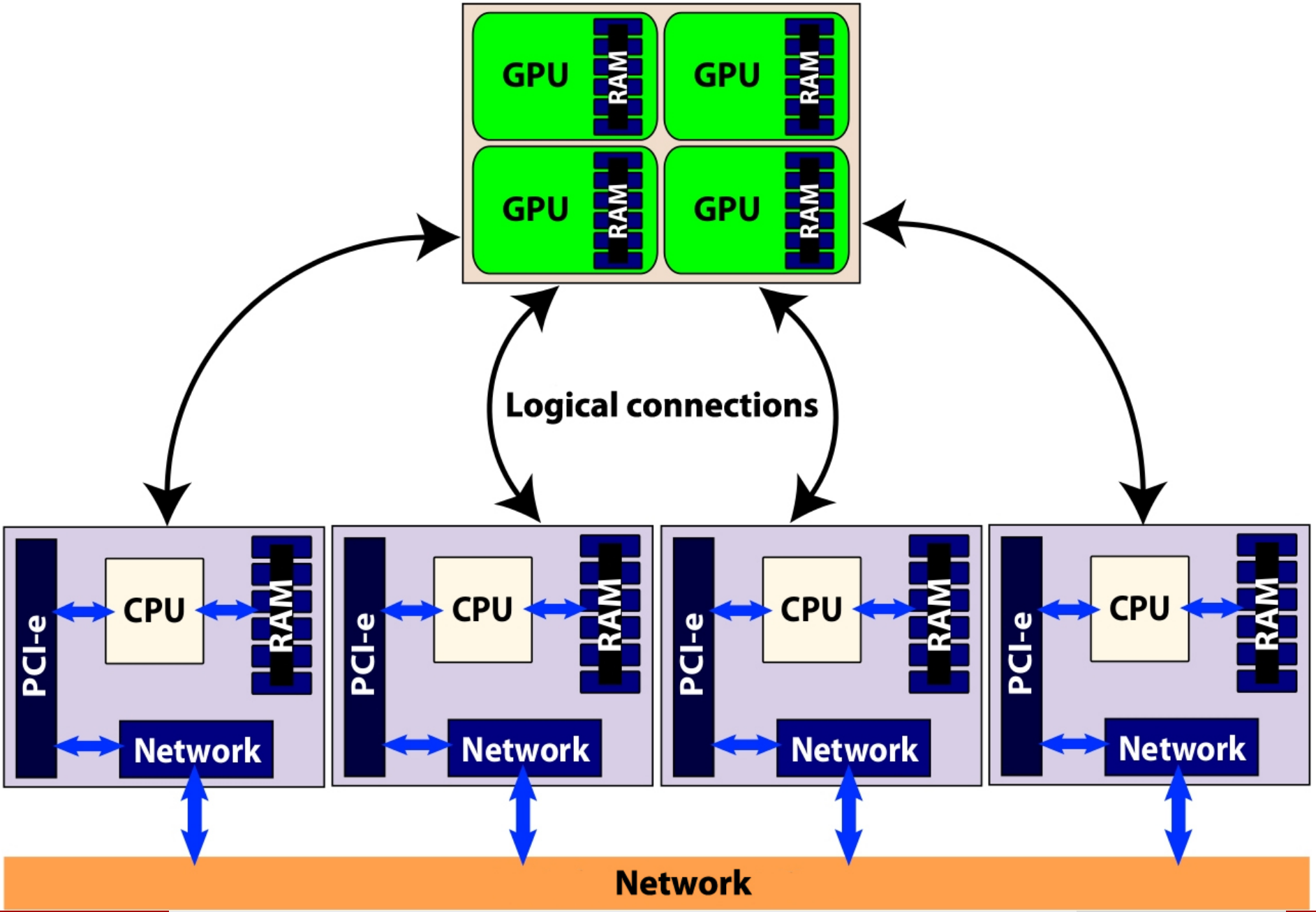



rCUDA
remote CUDA

Cluster without GPU Virtualization

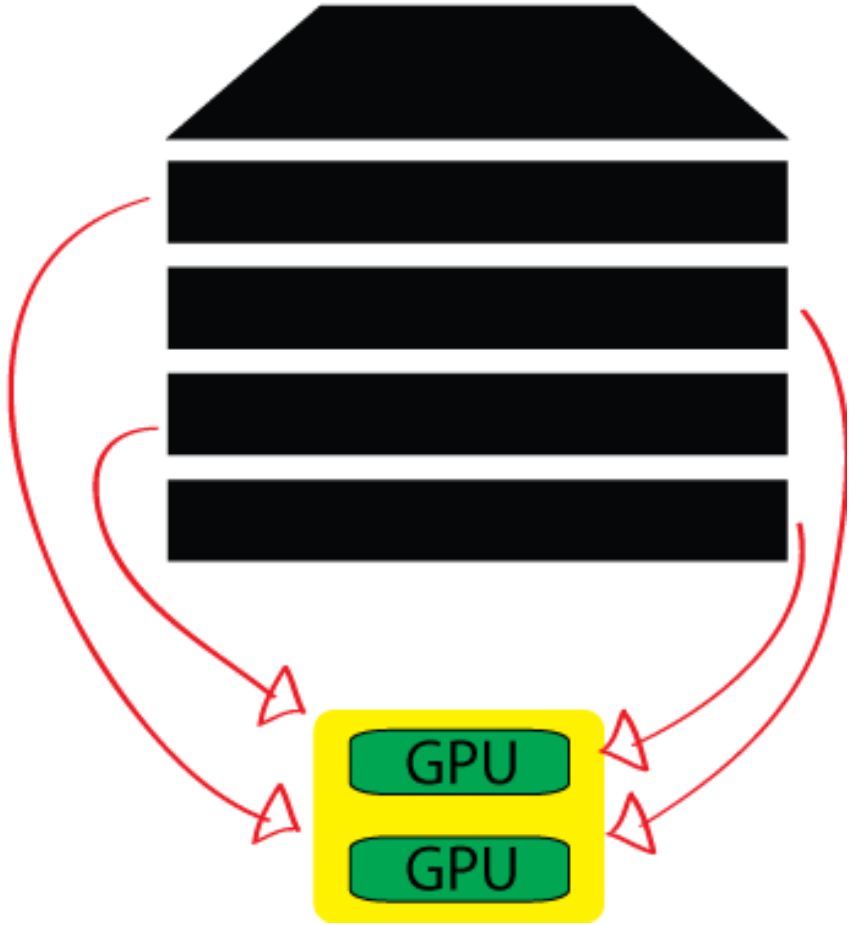


Cluster with rCUDA

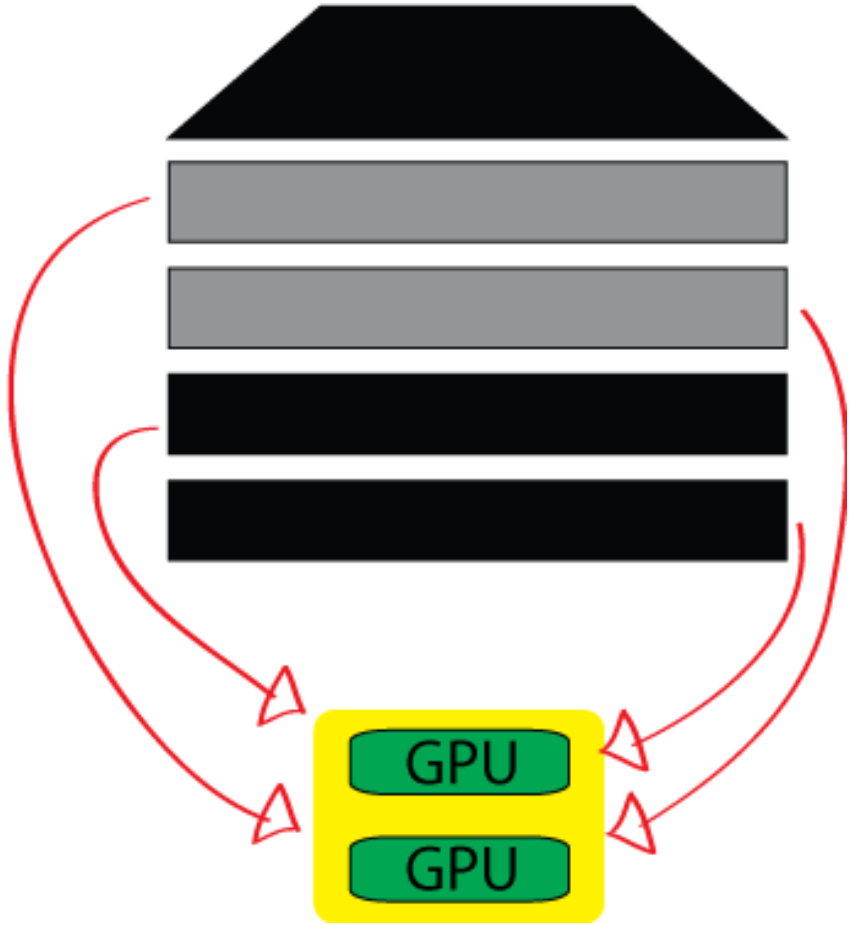


Virtualized Resources Management

2 nodes

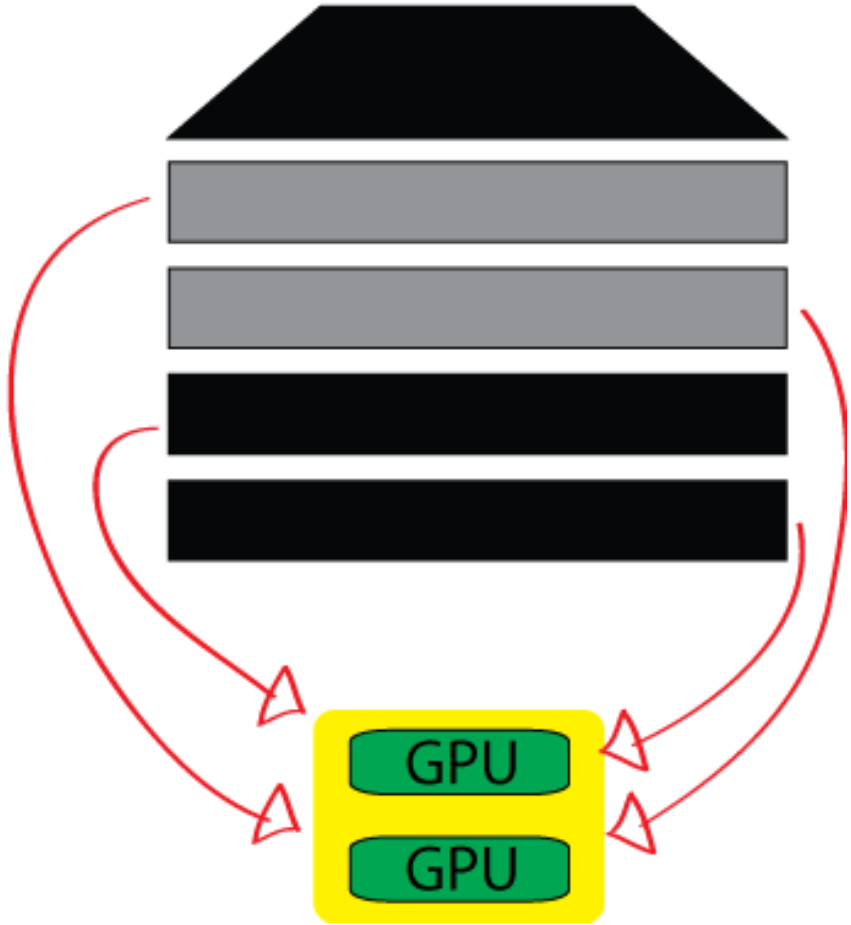


Virtualized Resources Management

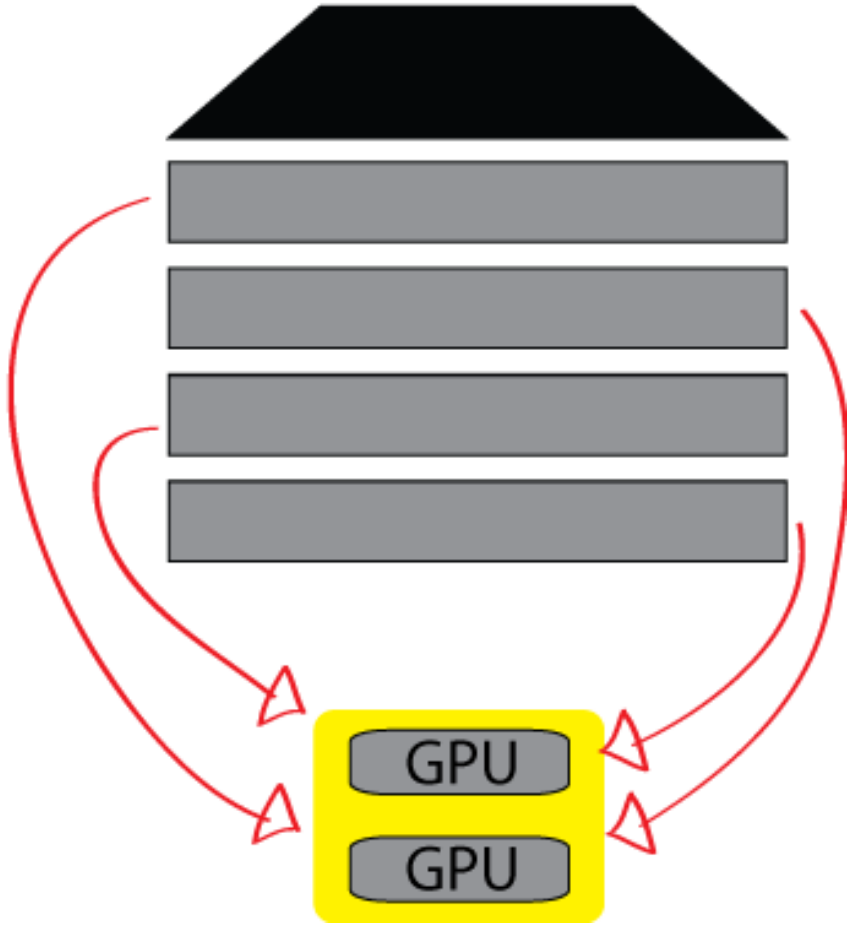


Virtualized Resources Management

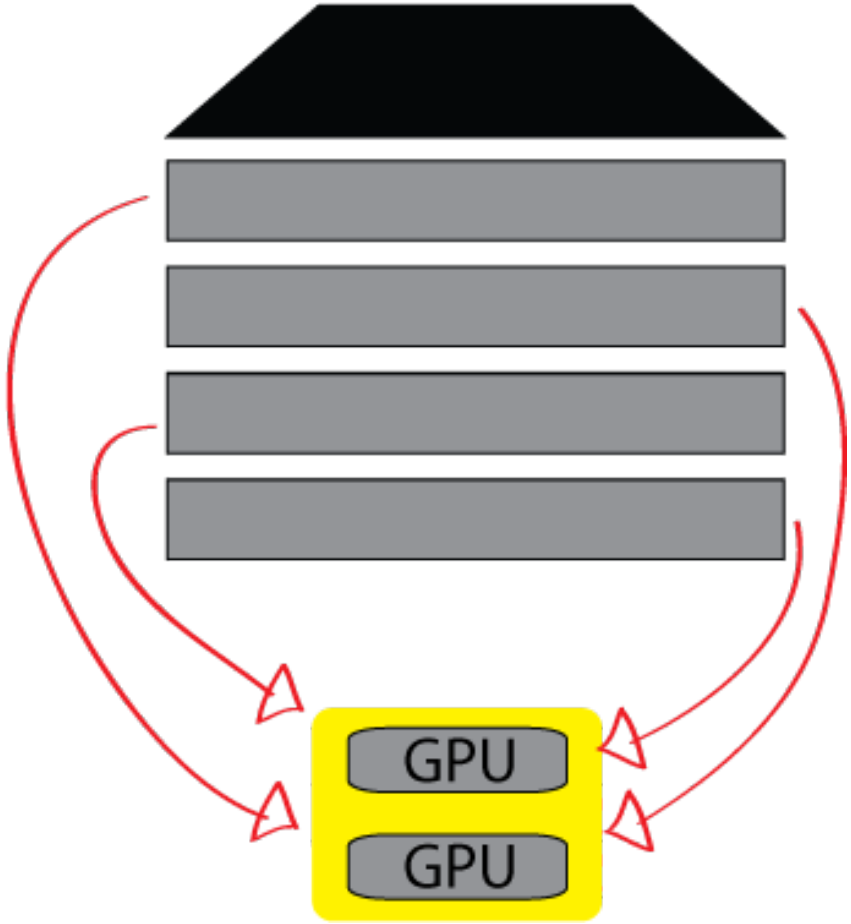
2 nodes
2 GPUs

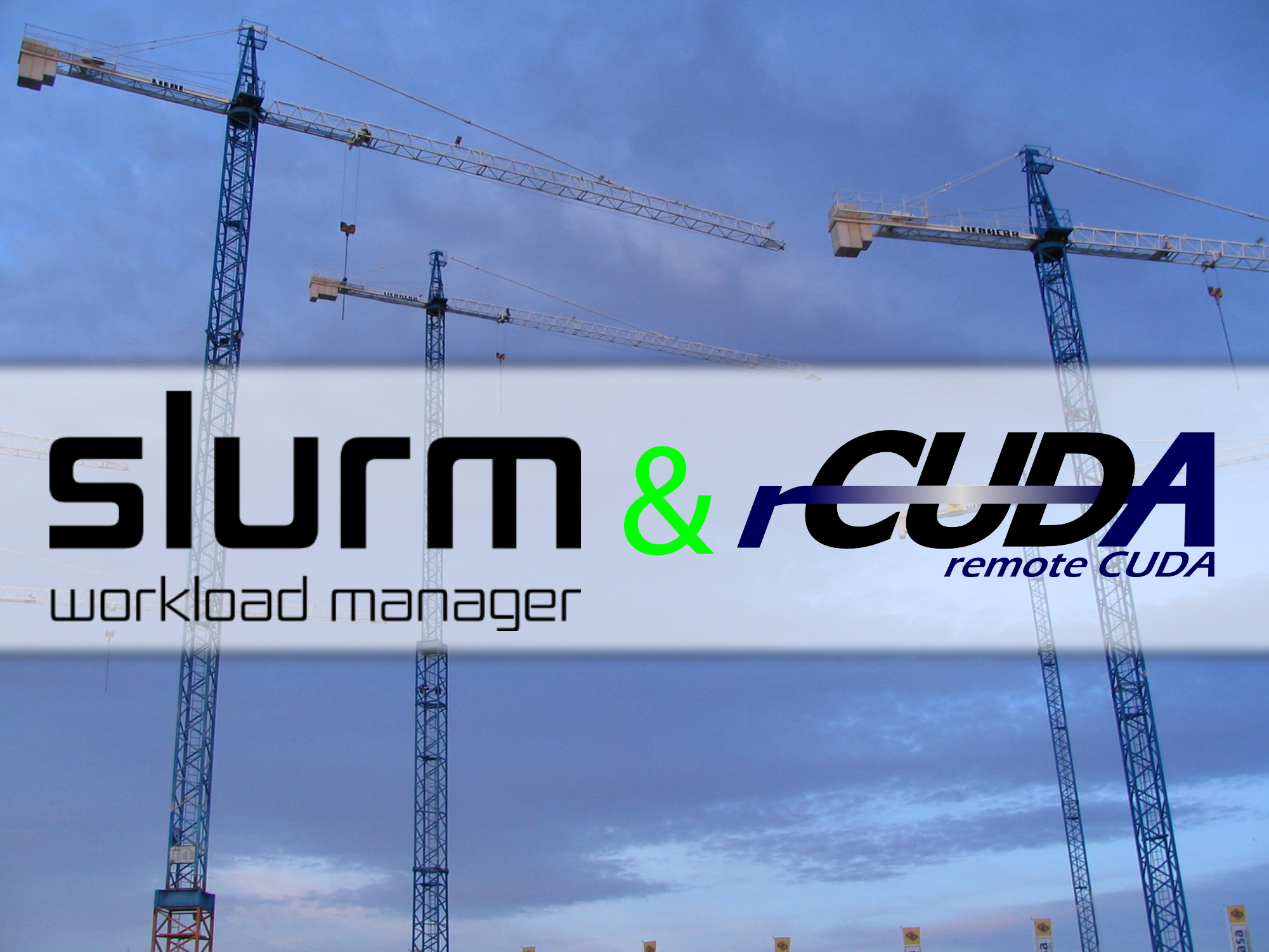


Virtualized Resources Management

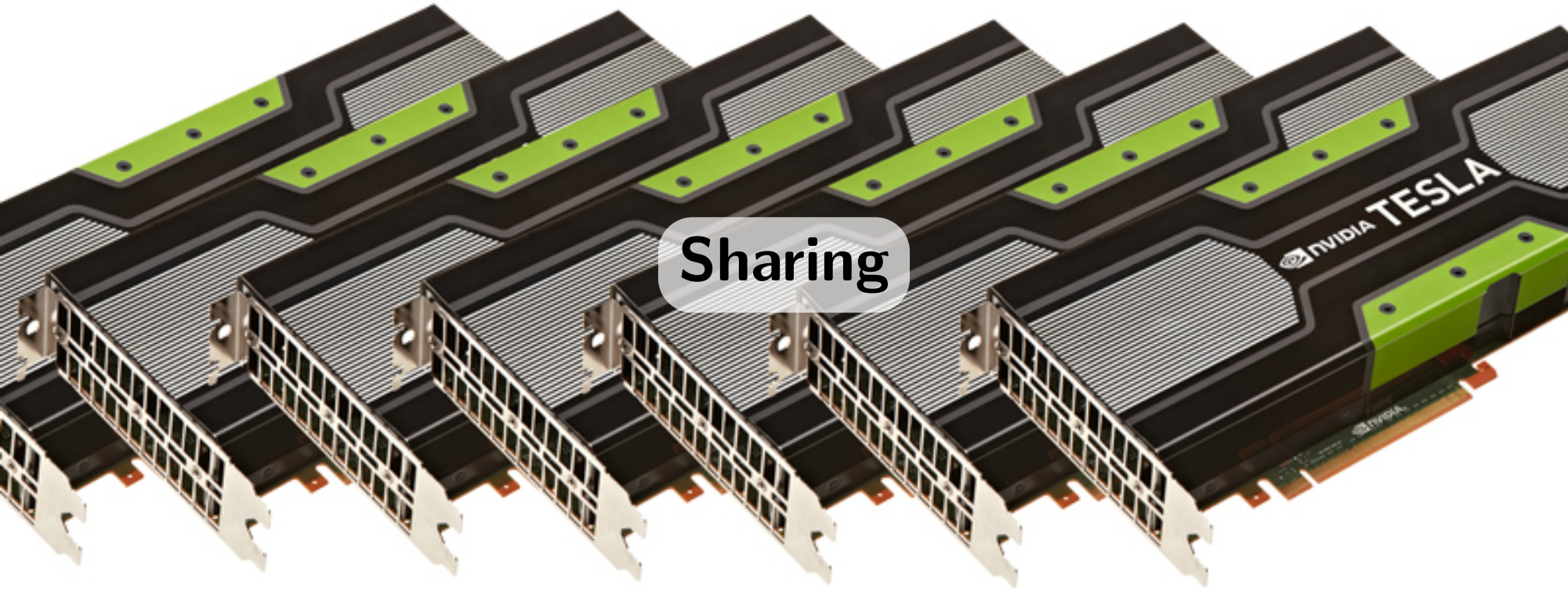


Virtualized Resources Management



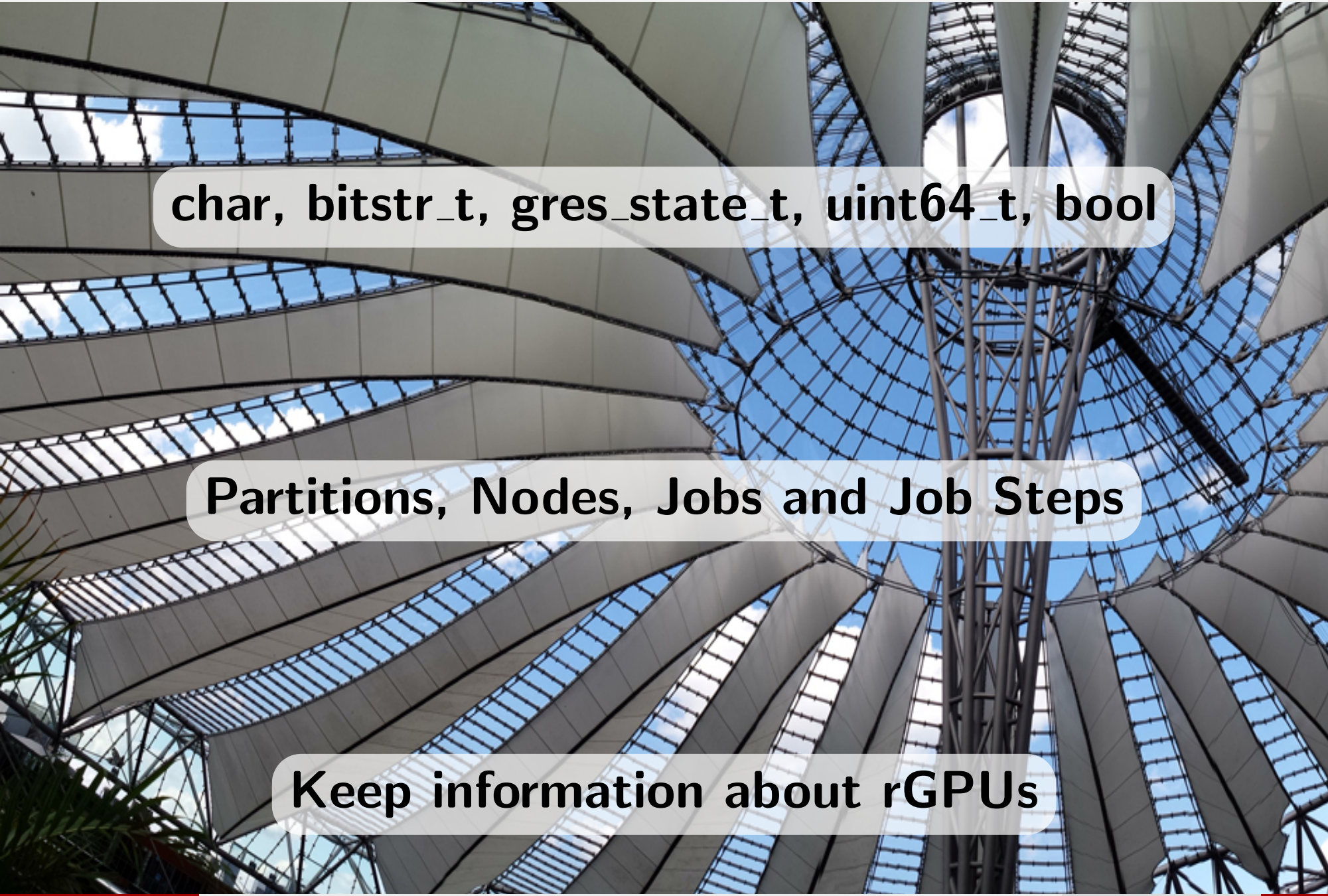

slurm
workload manager
&
rCUDA
remote CUDA

Special resource: rGPU



Remote Usage

Data structures



RPC Protocol

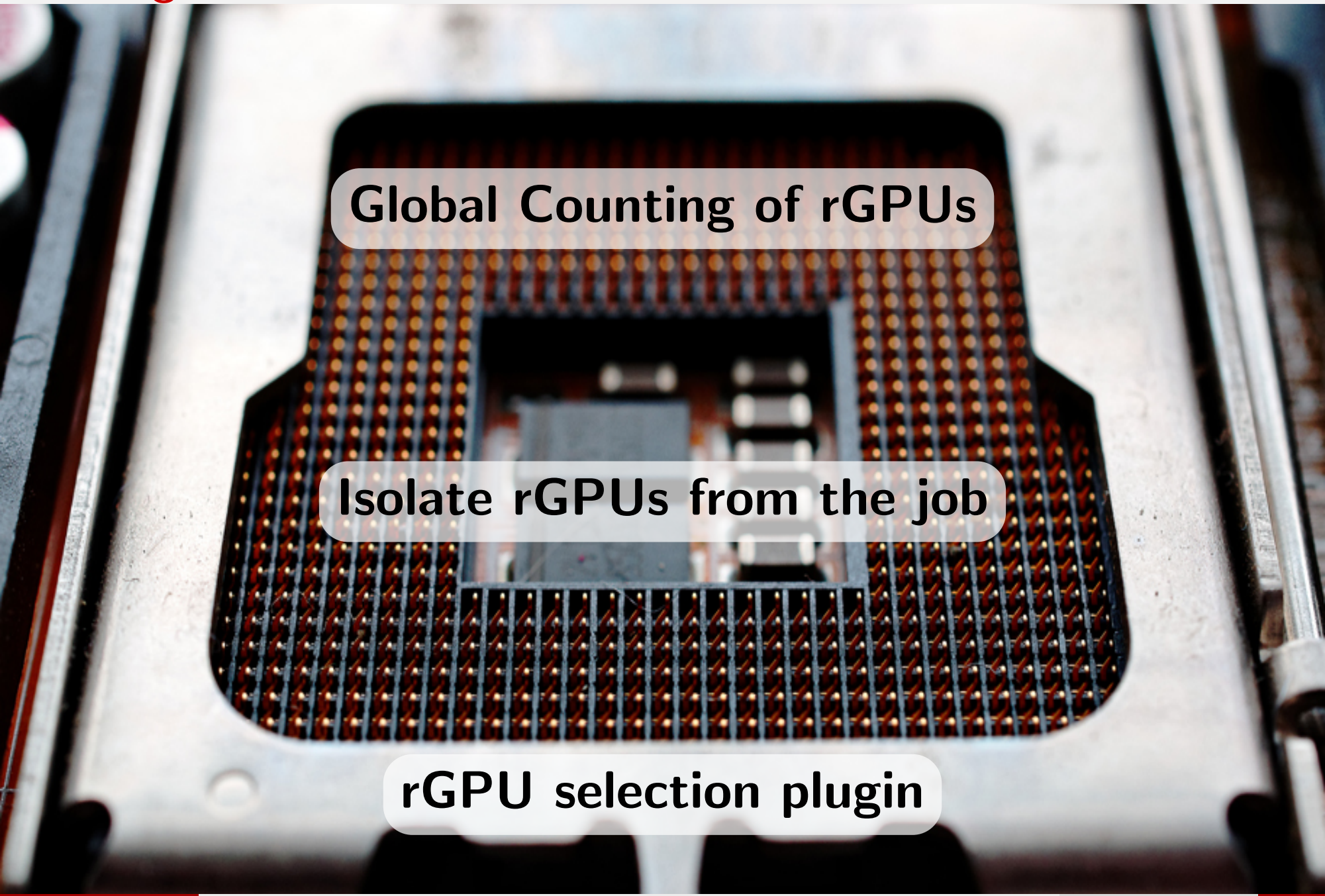
```
char *rgpulist;
```

```
packstr(msg->rgpulist, buffer);
```

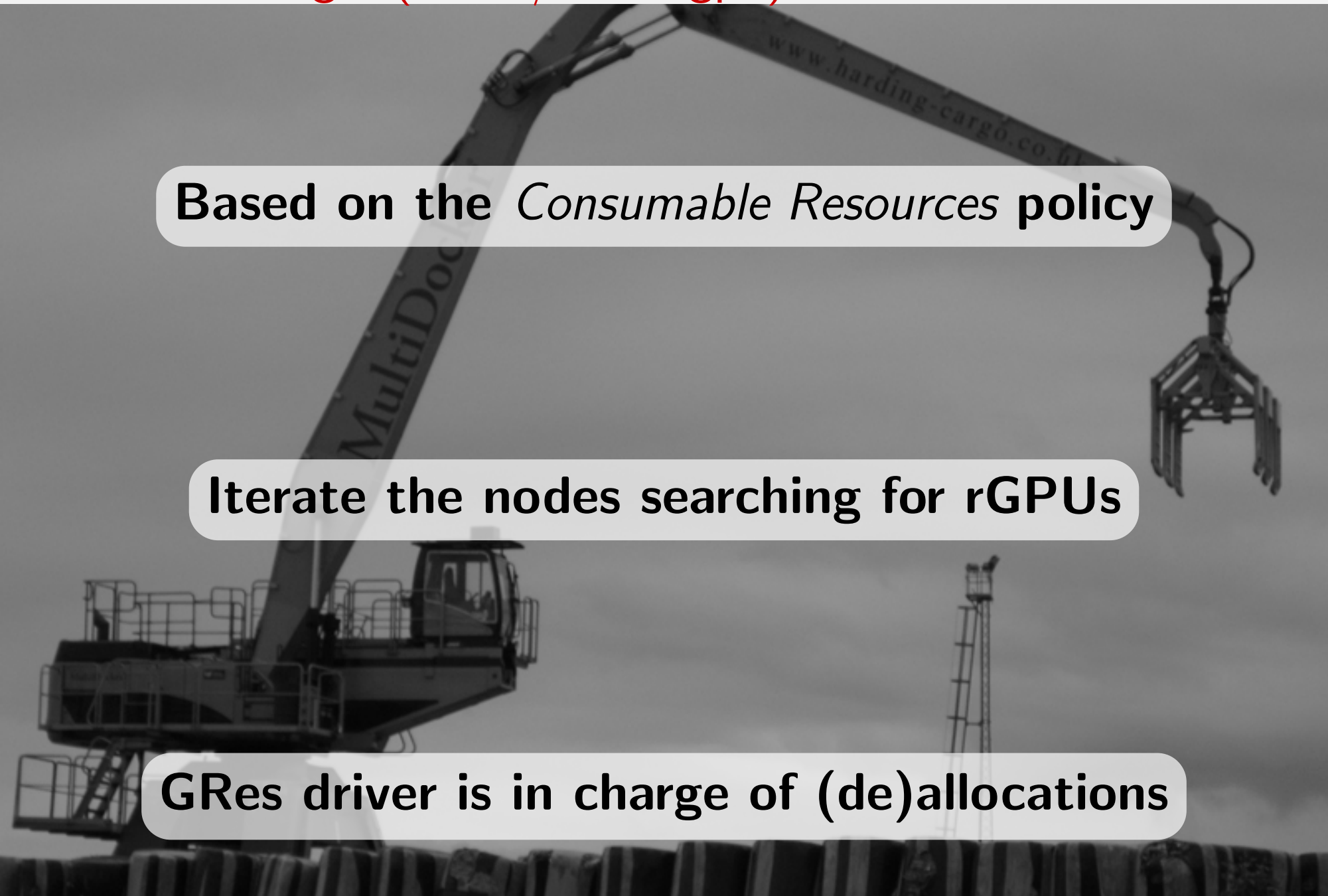
```
safe_unpackstr_xmalloc(msg->rgpulist, &uint32_size, buffer);
```

Environment Variables

Management daemon *slurmctld*



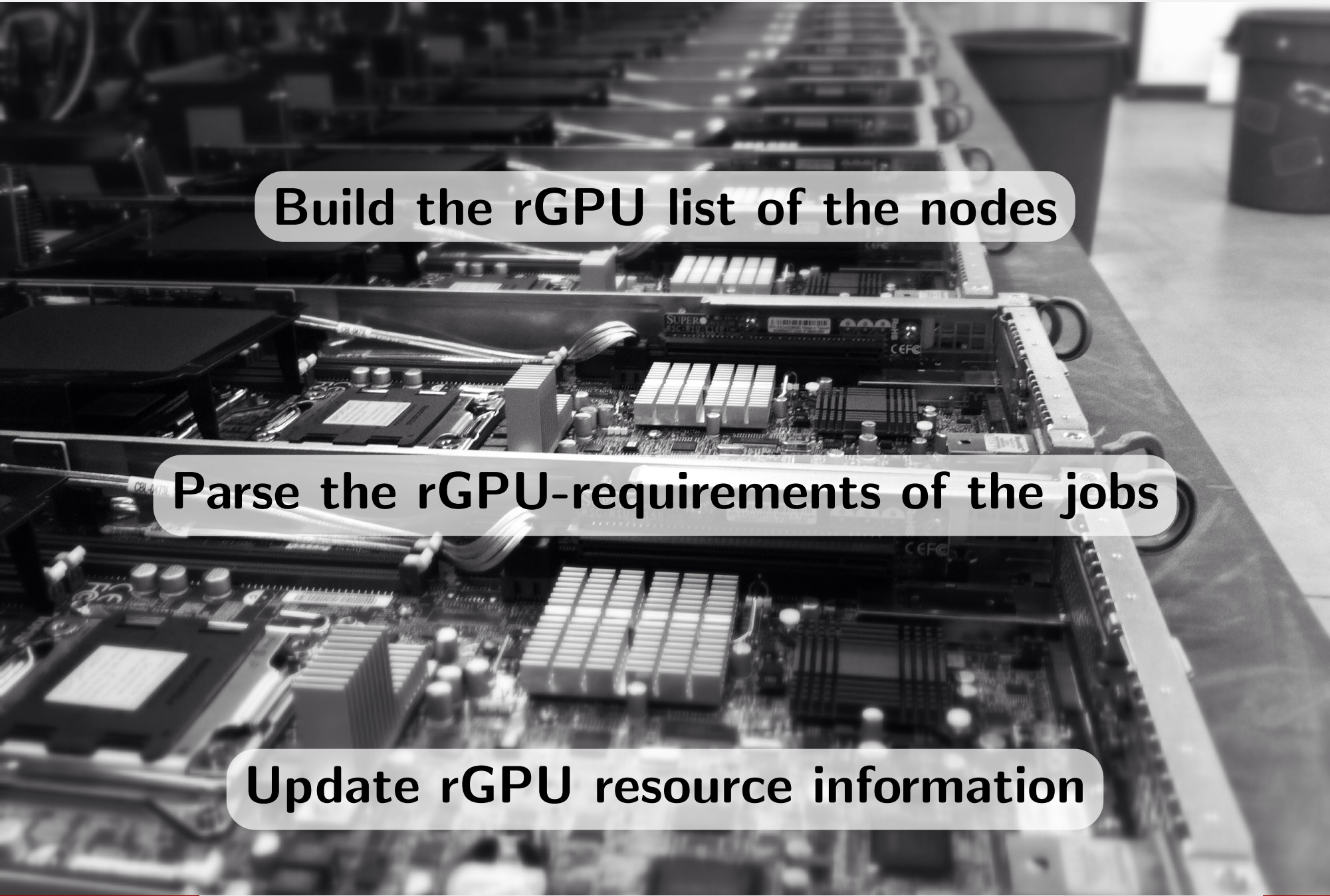
Selection Plugin (select/cons_rgpu)



Based on the *Consumable Resources* policy

Iterate the nodes searching for rGPUs

GRes driver is in charge of (de)allocations


Build the rGPU list of the nodes
Parse the rGPU-requirements of the jobs
Update rGPU resource information

slurm.conf

```
SelectType = select/cons_rgpu
SelectTypeParameters = CR_CORE
GresTypes = rgpu[,gpu]

NodeName=node1 NodeHostname=node1
   CPUs=12 Sockets=2 CoresPerSocket=6
   ThreadsPerCore=1 RealMemory=32072
   Gres=rgpu:1[,gpu:1]
```

gres.conf

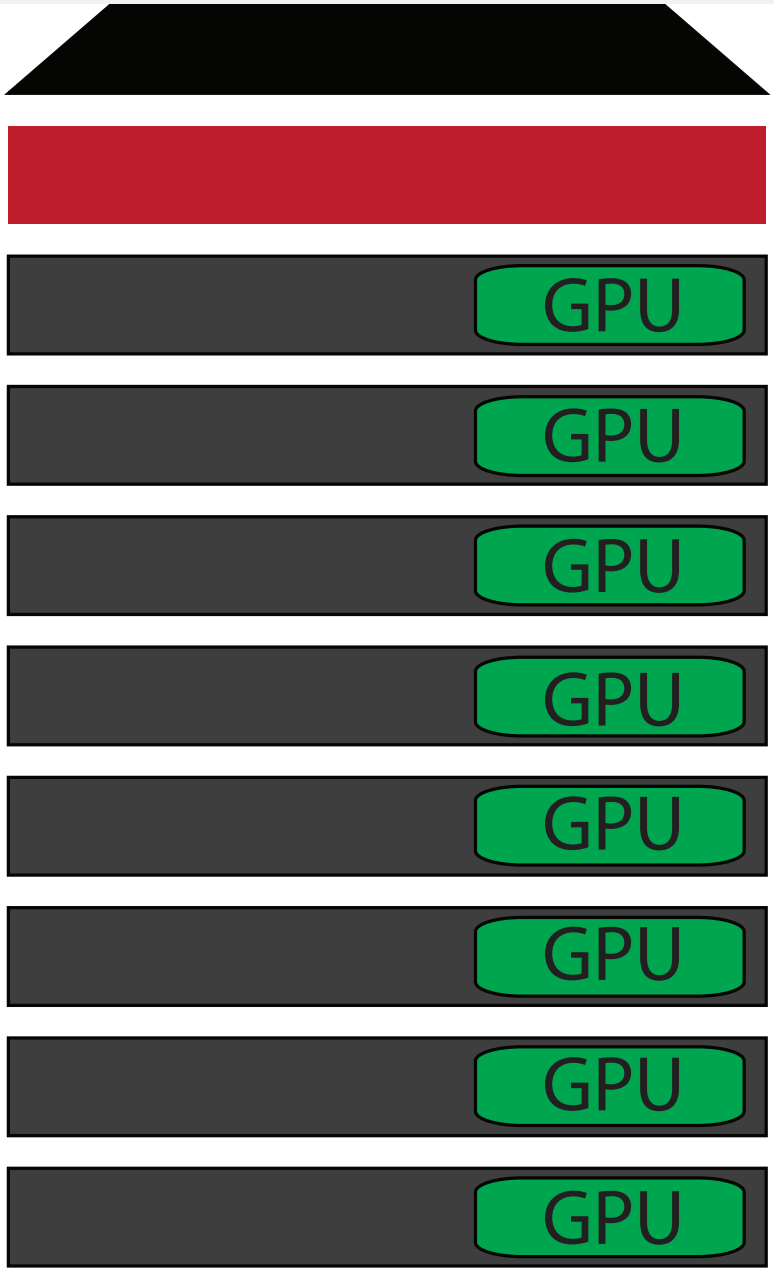
```
Name=rgpu File=/dev/nvidia0 Cuda=3.5 Mem=4726M
[Name=gpu File=/dev/nvidia0]
```

srun, sbatch, salloc

```
--rcuda-mode=(shared|excl)
--gres=rgpu(:X(:Y)?(:Z)?)?
            X = [1-9]+[0-9]*
            Y = [1-9]+[0-9]*[ kKmMgG]
            Z = [1-9]\.[0-9](cc|CC)
```


experimentation

Platform



CentOS 6.4

2 x Intel Xeon E5-2620 Hexacore

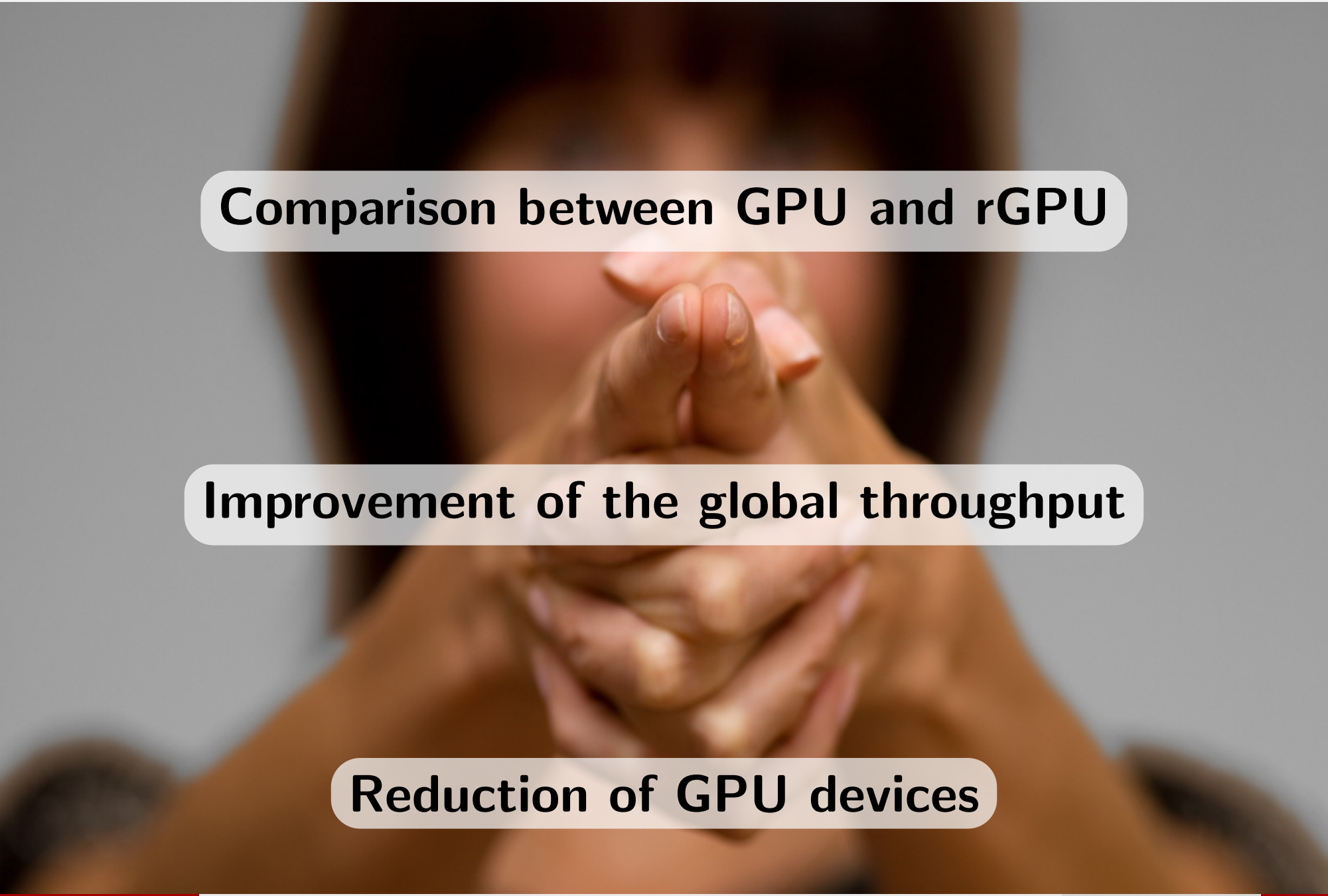
NVIDIA Tesla K20 GPU

Mellanox SX6025 (FDR)

Applications

Application	Multi-process	Multi-thread	GPU Computational Load
GPU-Blast	-	X	Medium
LAMMPS	X	-	High
MCUDA-MEME	X	X	Medium
GROMACS	X	X	None, only CPU

Workloads



Comparison between GPU and rGPU

Improvement of the global throughput

Reduction of GPU devices

Maximum Performance

Application	CUDA submission	rCUDA submission
GPU-Blast	-c6 –gres=gpu:1	-c6 –gres=rgpu:1:1686M
LAMMPS	-N5 -n5 –gres=gpu:1	-N5 -n5 –gres=rgpu:5:3275M
MCUDA-MEME	-N4 -n4 –gres=gpu:1	-N4 -n4 –gres=rgpu:4:163M
GROMACS	-N2 -n2 -c12	-N2 -n2 -c12

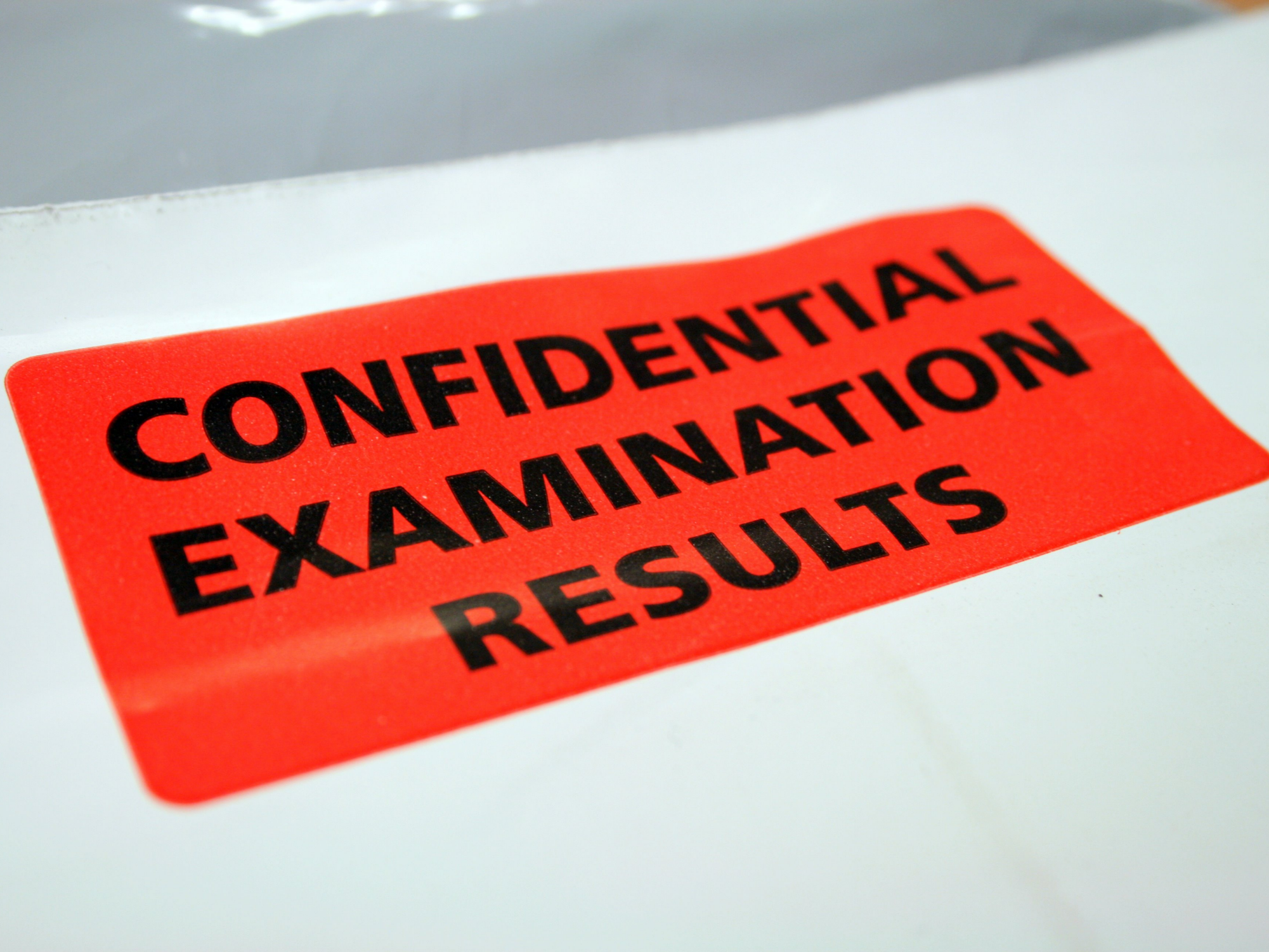
High **Performance** *Computing*

HPC $\rightarrow$ **HTC**

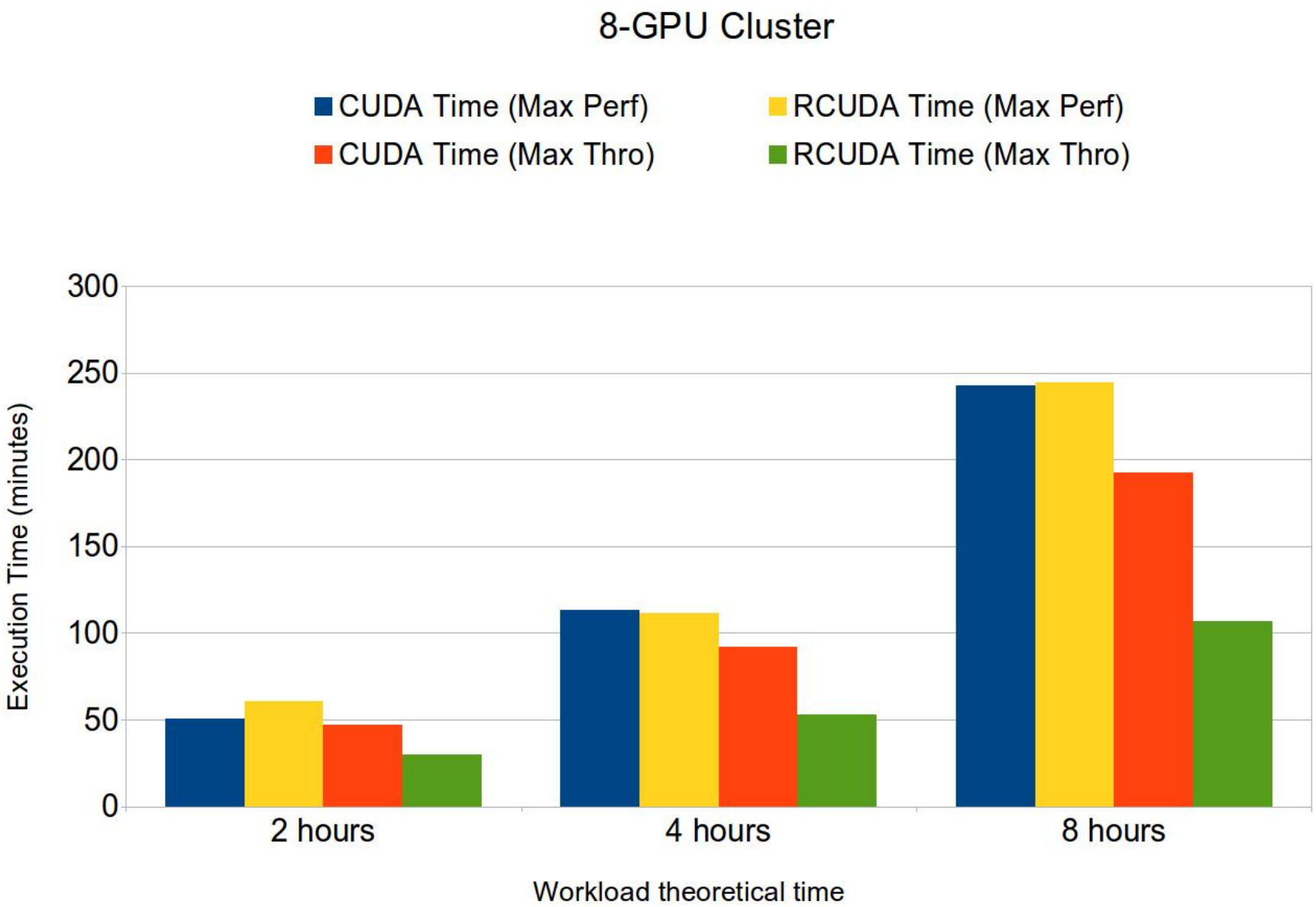
High **Throughput** *Computing*

Maximum Throughput

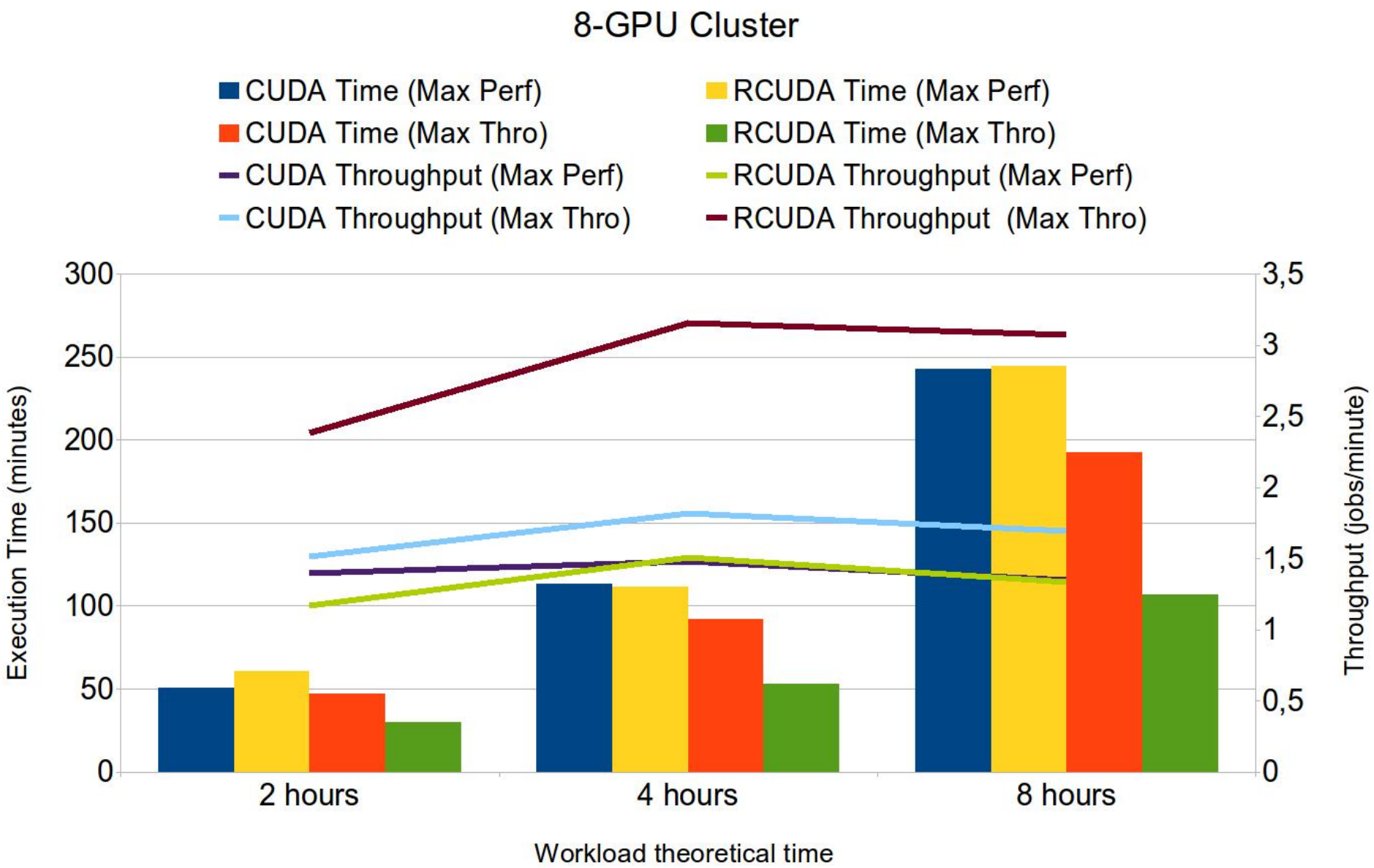
Application	CUDA submission	rCUDA submission
GPU-Blast	-c6 –gres=gpu:1	-c6 –gres=rgpu:1:1686M
LAMMPS	~~-N5~~ -n5 –gres=gpu:1	~~-N5~~ -n5 –gres=rgpu:5:3275M
MCUDA-MEME	-N4 -n4 –gres=gpu:1	~~-N4~~ -n4 –gres=rgpu:4:163M
GROMACS	-N2 -n2 -c12	-N2 -n2 -c12


CONFIDENTIAL
EXAMINATION
RESULTS

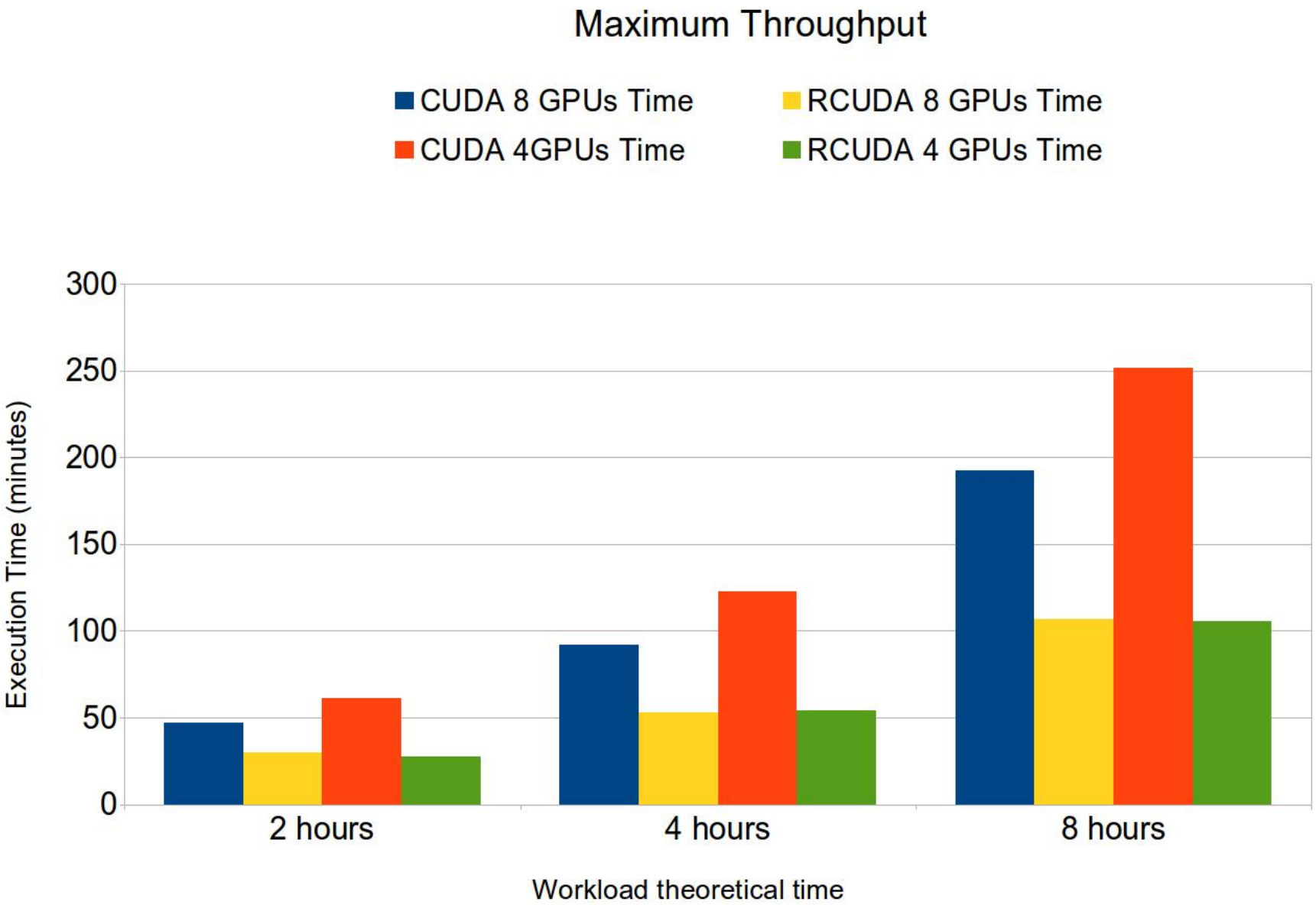
Max. Performance VS Max. Throughput



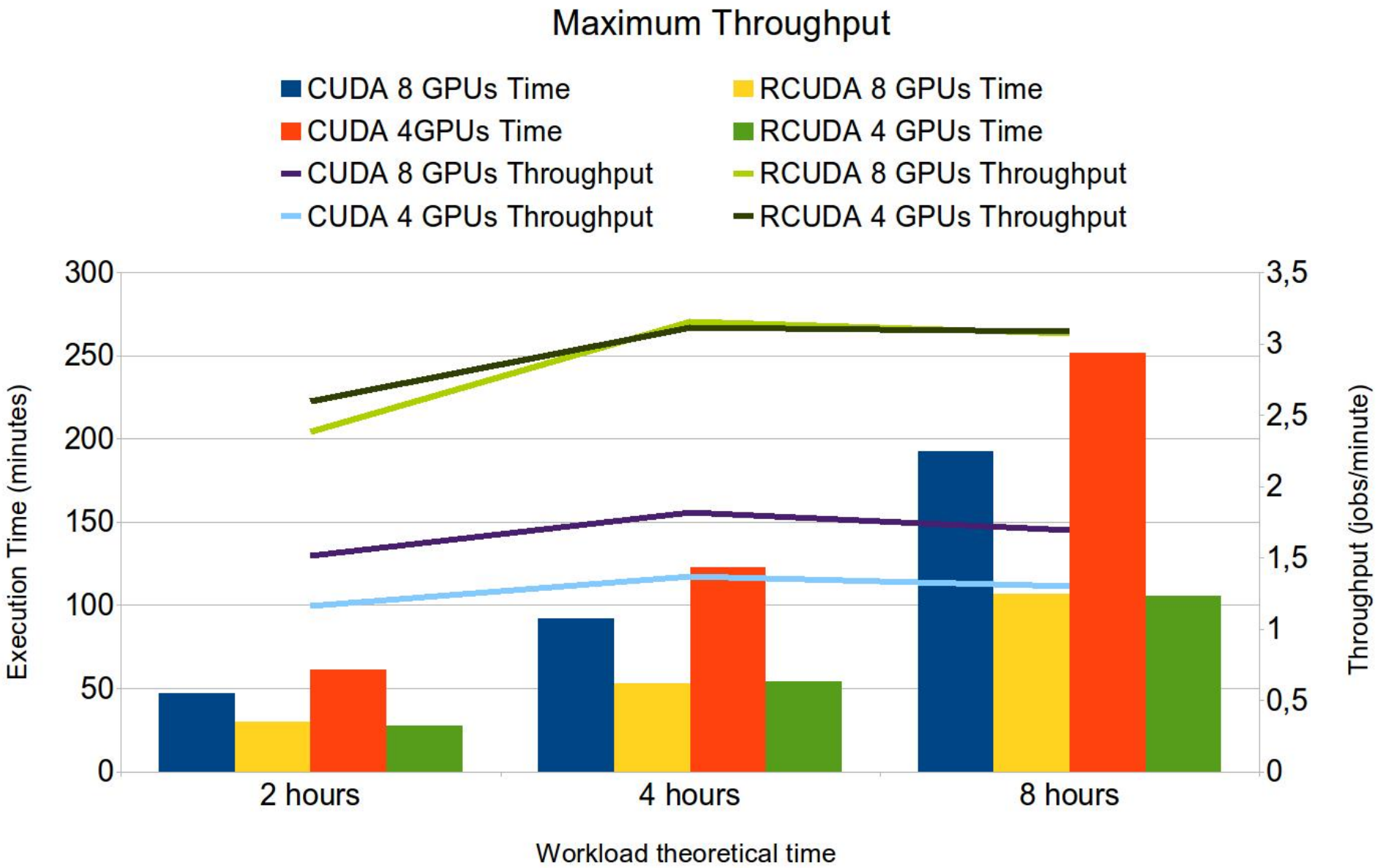
Max. Performance VS Max. Throughput



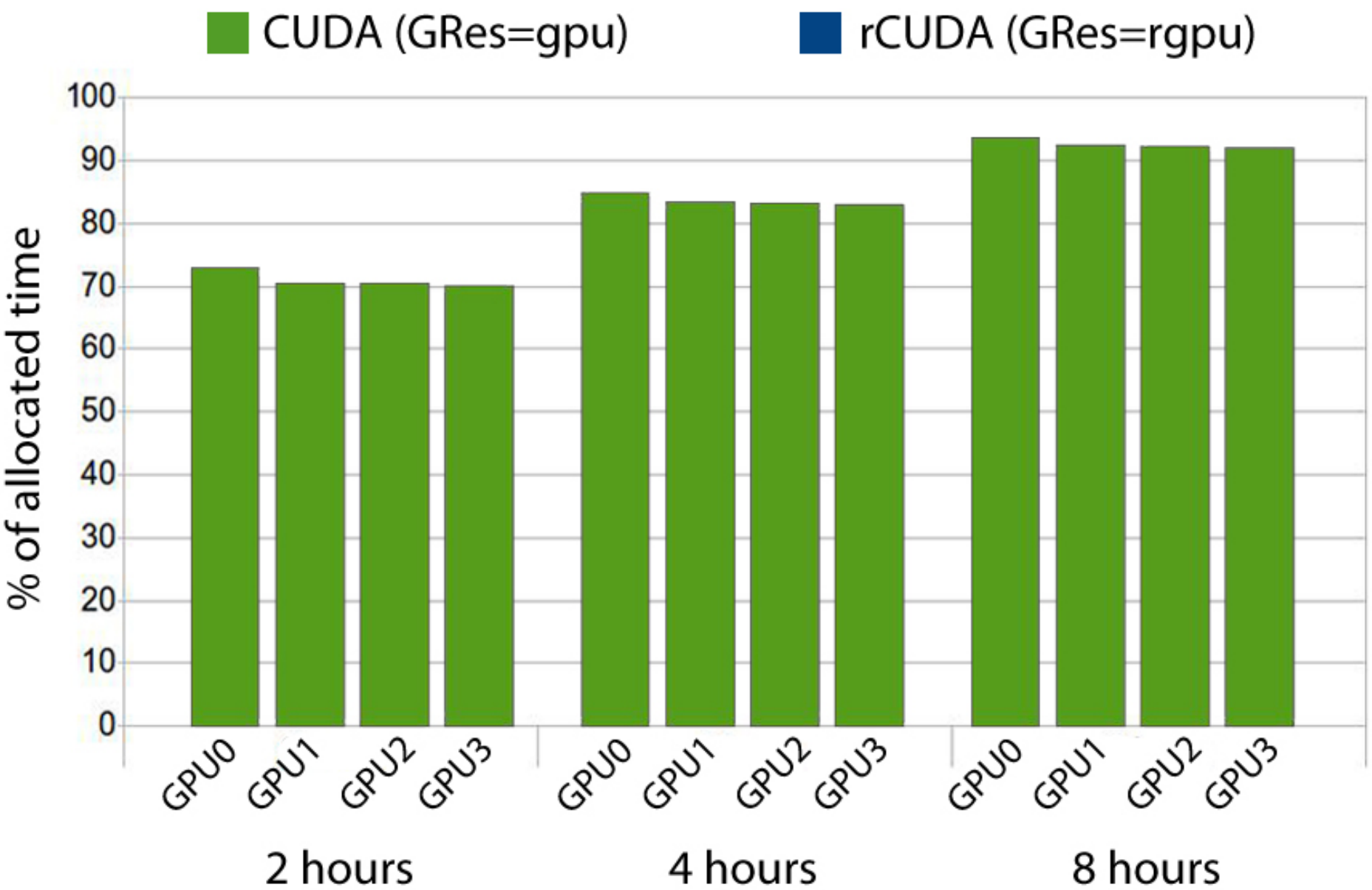
Max. Throughput with 8 and 4 GPUs



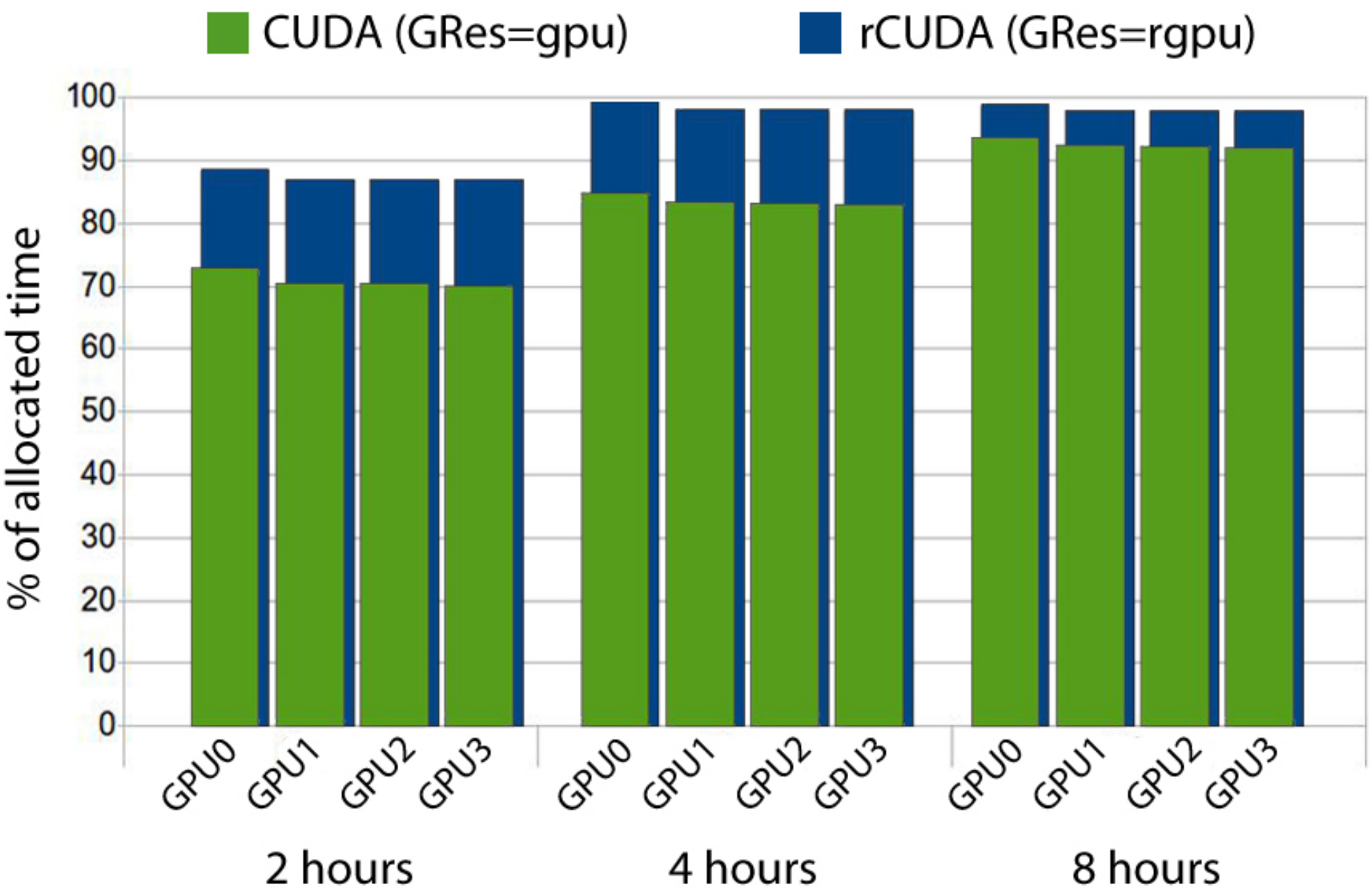
Max. Throughput with 8 and 4 GPUs



GPU Usage

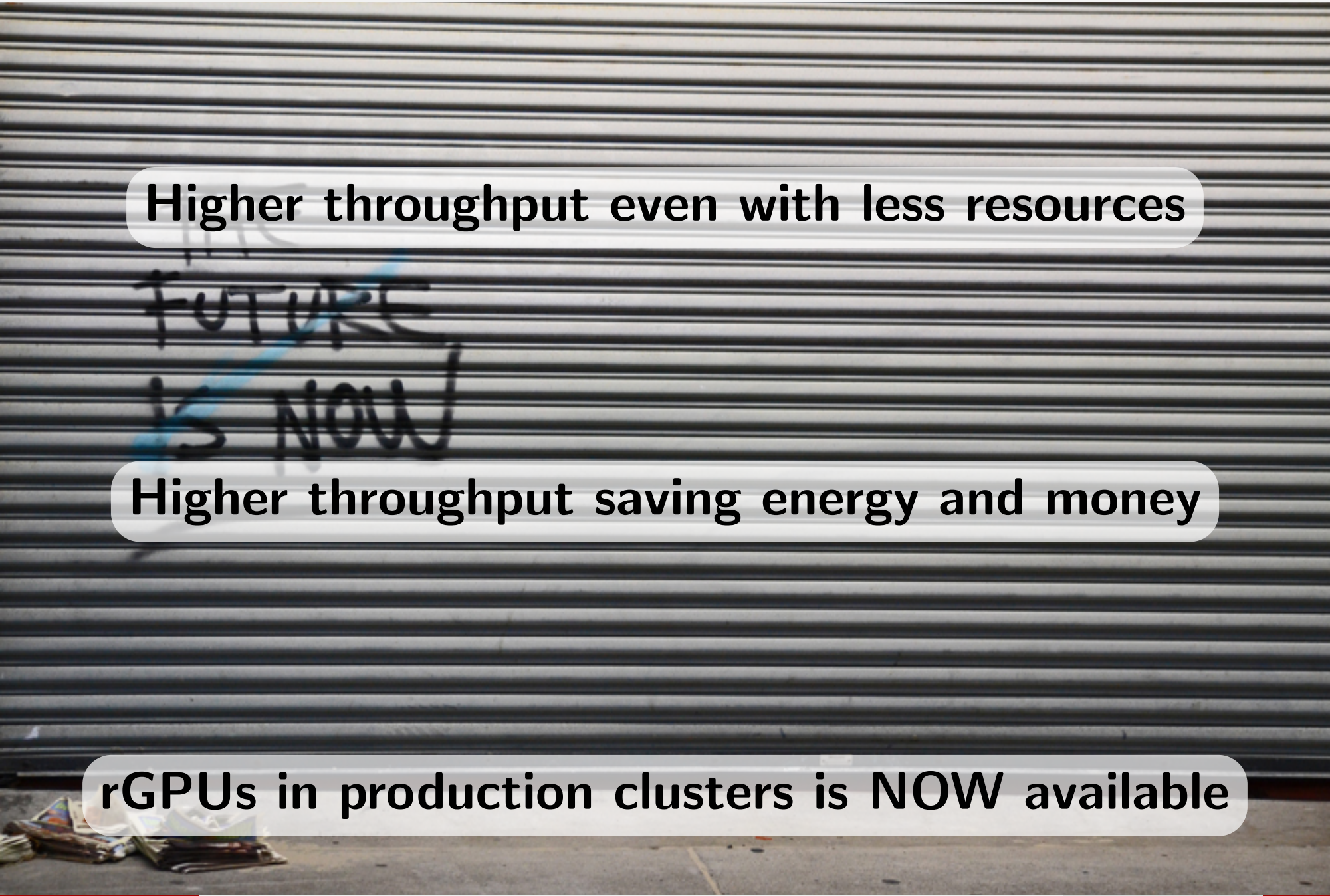

CUDA (GRes=gpu)
rCUDA (GRes=rgpu)
% of allocated time
100
90
80
70
60
50
40
30
20
10
0
GPU0
GPU1
GPU2
GPU3
GPU0
GPU1
GPU2
GPU3
GPU0
GPU1
GPU2
GPU3
2 hours
4 hours
8 hours

GPU Usage


CUDA (GRes=gpu)
rCUDA (GRes=rgpu)
% of allocated time
100
90
80
70
60
50
40
30
20
10
0
GPU0
GPU1
GPU2
GPU3
GPU0
GPU1
GPU2
GPU3
GPU0
GPU1
GPU2
GPU3
2 hours
4 hours
8 hours

Conclusions

Conclusions



Higher throughput even with less resources

Higher throughput saving energy and money

rGPUs in production clusters is NOW available

Make the most of your GPUs with

REFERENCE:

S. Iserte, A. Castelló, R. Mayo, E. S. Quintana-Ortí, F. Silla, J. Duato, C. Reaño, J. Prades. C. Reaño, J. Prades.
SLURM Support for Remote GPU Virtualization: Implementation and Performance Study, in SBAC-PAD, 2014 (accepted).