Share and aggregate GPUs in your cluster

F. Silla
Technical University of Valencia
Spain
Remote GPU virtualization
Accelerating applications

- Many applications require a lot of computing resources
- Execution time is usually increased
- Applications are accelerated to get their execution time reduced
GPUs as accelerators

The basic building block is a node with one or more GPUs
GPU-accelerated clusters

From the programming point of view, just a set of nodes:
• one or more CPUs per node (probably with several cores per CPU)
• one or more GPUs per node (typically between 1 and 4)
• an interconnection network
Getting benefit from accelerated clusters

GPUs only bring benefits for the right kind of code:

- There must be data parallelism in the code: this is the only way of taking benefit from the hundreds of processors in a GPU
  - Code with **low** amount of data parallelism
  - Code with **high** amount of data parallelism
  - Code with **moderate** amount of data parallelism
Getting benefit from accelerated clusters

GPUs only bring benefits for the right kind of code:

- There must be **data parallelism in the code**: this is the only way of taking benefit from the hundreds of processors in a GPU

- Code with **low** amount of data parallelism
- Code with **high** amount of data parallelism
- Code with **moderate** amount of data parallelism

**Do you really need a GPU-accelerated cluster?**
Getting benefit from accelerated clusters

GPUs only bring benefits for the **right kind of code**: 

- There must be **data parallelism in the code**: this is the only way of taking benefit from the hundreds of processors in a GPU
  - Code with **low** amount of data parallelism
  - Code with **high** amount of data parallelism
  - Code with **moderate** amount of data parallelism

You can take a lot of benefit from your GPU-accelerated cluster
Getting benefit from accelerated clusters

GPUs only bring benefits for the right kind of code:
• There must be data parallelism in the code: this is the only way of taking benefit from the hundreds of processors in a GPU
  • Code with low amount of data parallelism
  • Code with high amount of data parallelism
  • Code with moderate amount of data parallelism

Applications will make a moderate use of the GPUs in your cluster
Money leakage in current clusters?

For applications with moderate levels of data parallelism, GPUs in a cluster may be idle for long periods of time:

- Initial acquisition costs not amortized
- Space: GPUs reduce CPU density
Further concerns in accelerated clusters

FACTS:
• Applications can only use the GPUs within their node

CONCERNS:
• For applications with moderate levels of data parallelism, many GPUs in the cluster may be idle for long periods of time
• Multi-GPU applications running on a subset of nodes cannot make use of the tremendous GPU resources available at other cluster nodes (even if they are idle)
What is missing is ...

... some flexibility for using the GPUs in the cluster
Addressing current concerns

- A way of addressing the first concern is by removing the GPUs that are not really needed and sharing the remaining ones among all the nodes.

- For the second concern, all the GPUs in the cluster could be granted to a single application.
Remote GPU virtualization

• This new configuration requires:
  • A way of seamlessly sharing GPUs across nodes in the cluster (remote GPU virtualization)
  • Enhanced job schedulers that take into account the new GPU configuration
How GPU virtualization adds value

GPU virtualization allows a new vision of a GPU deployment, moving from the traditional cluster configuration:

to the following one ....
How GPU virtualization adds value

Physical configuration

Logical connections

Interconnection Network
Benefits from remote GPU virtualization

Remote GPU virtualization increases GPU utilization, also lowering power consumption, at the same time that initial acquisition costs are reduced.
GPU virtualization and power

- Remote GPU virtualization can **further reduce power consumption**:
  - enhancing the scheduling process so that GPU servers are **put into low-power sleeping modes** when not needed
The rCUDA GPU virtualization framework

A framework enabling a CUDA-based application running in one (or some) node(s) to access GPUs in other nodes

It is useful for:

Applications with moderate level of data parallelism
Applications for multi-GPU computing
About rCUDA

Available at developers’ website under request:

- High-performance InfiniBand communications library
- TCP/IP compatible for other networks
- Binary compatible with support for CUDA 5
- ARM support
- SLURM support

http://www.rcuda.net