

MSI CUDA Tutorial

Operational Aspects of GPU
Computing

Jeff McDonald HPC Operations

GPU Computing Requirements

- For computing on a GPU system, one must be provided a GPU processor through a graphics processing card. Special software (Cuda) allows users to directly access the GPU processors for computing...for this you must have a Cuda-Enabled GPU card.
- Cuda is relatively new and there are several versions. The latest supporting double precision arithmetic is version 2.0.

Resources

- MSI has been provided (by NVIDIA) twenty units of the GTX 480 GPU nodes. Sixteen of the are available in four nodes: gput0[1-4]. Each node has four GPU nodes.
- Users need to be members of the vglusers group in order to access the GPUs on the nodes.

Node Properties

- CPU: 2 x E5620 @ 2.40 GHz
- 16 total cores
- 24 GB of memory
- 4 x PCI x16 bus for GPUs
- 1 TB local storage
- Connected to all MSI standard home/authentication systems
- Setup with Virtual GL configurations

GPU Properties

- Cuda Driver Version: 4.10
- Runtime Version: 4.0
- CUDA Capability Version 2.0
- Cuda Cores: 480
- Global Memory: 1.5 GB
- GPU Clock speed 1.4 GHz
- Max Threads per block 1024
- 1 Concurrent Copy + Execution
- No memory ECC support

System Tools

- Nvidia tool: nvidia-smi
 - Not very useful for the Quadro cards
 - Only have access to memory, temperature
- deviceQuery
 - Lists the device properties (but not ownership or load)
- nvidia-settings
 - Useful for tuning x applications mostly

Cuda Toolkits

- In order to run codes on the GPU, you need to use special compiler directives which are provided by NVIDIA:

```
$ module load cuda  
$ which nvcc  
/usr/local/cuda/bin/nvcc
```

- Must be loaded in order to compile cuda software and run it.

New Acquisitions

- GPGPU system 8 nodes, 32 GPGPUs, IB connectivity
- 2 x Dell C410x
- 32 Tesla C2050s
- Available Late November



Hands-On Examples

- Login to the gpu nodes
- Modules available
- `deviceQuery`
- Running virtual GL