OpenPOWER™

Can we further boost HPC Performance?

Integrate IBM Power System to OpenStack Environment

(Part 1)

Ankit Purohit, Research Engineer
NTT Communications

Join the Conversation #OpenPOWERSummit

Agenda

1. Our Background
2. Providing GPU Resources: P100
3. Benchmarking with different tool : DGX-1 vs S822LC for HPC(Minsky)
4. I/O advantages of IBM Power System
5. Performance improvement with Memory Interleave: nbody
6. Summary

Agenda

1. Our Background
2. Providing GPU Resources: P100
3. Benchmarking with different tool : DGX-1 vs S822LC for HPC(Minsky)
4. I/O advantages of IBM Power System
5. Performance improvement with Memory Interleave: nbody
6. Summary

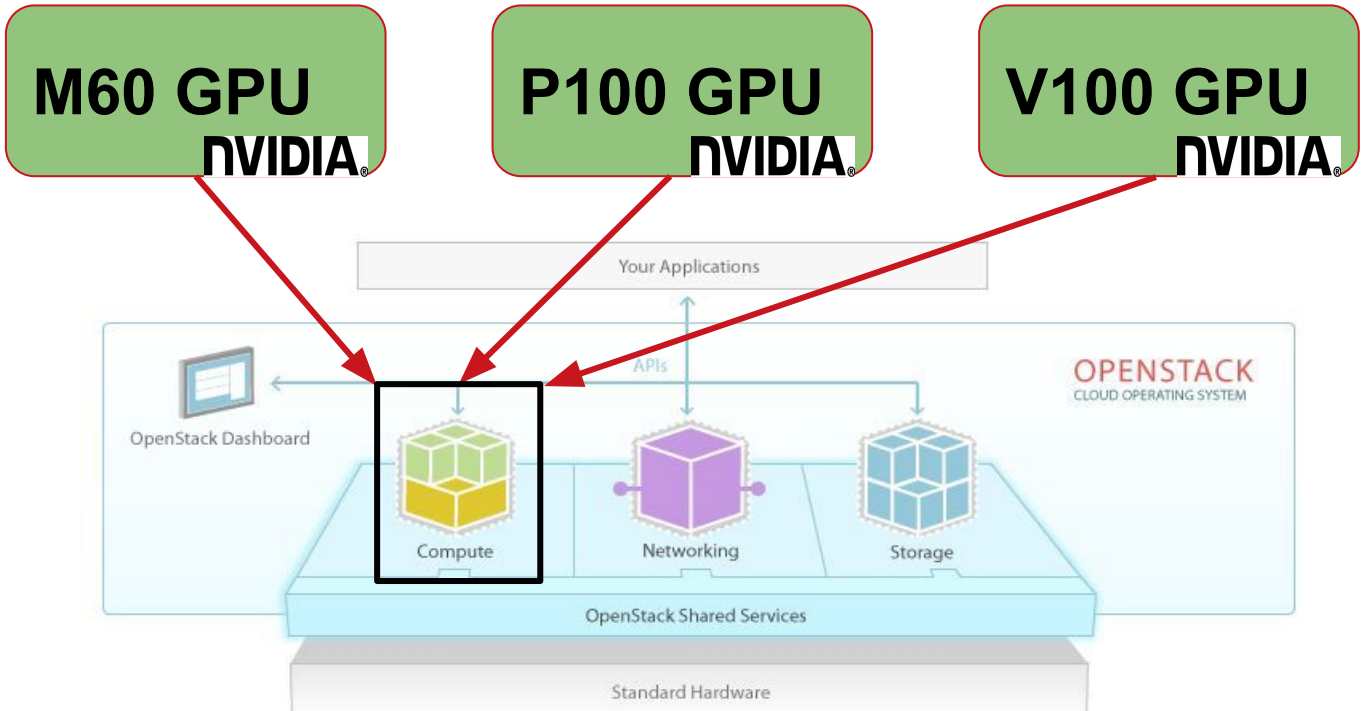
Background

- NTT Communications is the largest Telecommunications company in Japan and has subsidiaries and offices in over 110 cities worldwide.
- Part of a Fortune Global 100 company.
- NTTCom offers Artificial Intelligence communication engine like COTOHA.
  http://www.ntt.com/en/services/application/cotoha.html
- Currently, we provide GPU(M60,P100 and V100 [x86]) cloud using OpenStack.

Our current situation

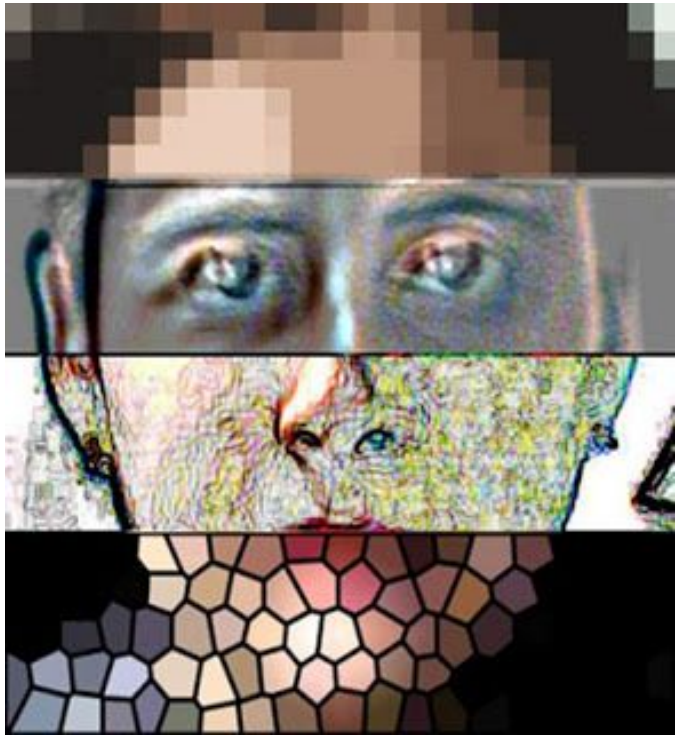
OpenPOWER

- We provide instances to users using following GPU cards with various flavors on x86 servers.



courtesy:https://www.google.co.jp/search?q=openstack+architecture&tbm=isch&source=lnms&sa=X&ved=0ahUKEwiEnsCR8sTZAhWMTLwKHYFWDV0Q_AUICigB&biw=1278&bih=636&dpr=2#imgrc=mKRInZyK9yS6IM:

We need more computing for “Deep Learning”



How about IBM POWER8?

IBM Power System S822LC (Minsky)

- POWER8 has more power than x86
- The difference between POWER8 and DGX-1 (x86) will be discussed soon in part 2.

Agenda

OpenPOWER

1. Our Background
2. Providing GPU Resources: P100
3. Benchmarking with different tool : DGX-1 vs S822LC for HPC(Minsky)
4. I/O advantages of IBM Power System
5. Performance improvement with Memory Interleave: nbody
6. Summary

Providing GPU Resources with P100 GPU

OpenPOWER

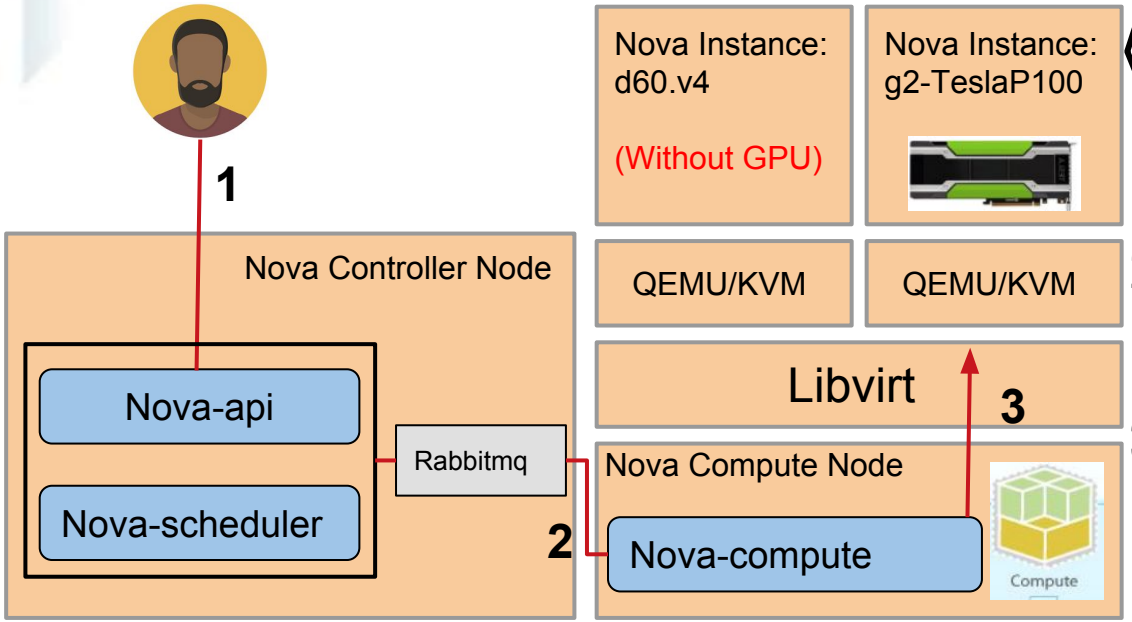
- We provide instances to in-house users with and without GPU on our private cloud environment with following.

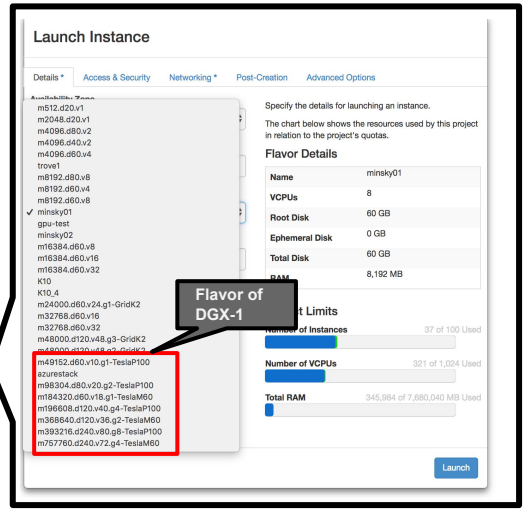
OpenStack using pci Passthrough with KVM

openstack™ CLOUD SOFTWARE

Private Cloud with OpenStack

1. Nova-api receives GPU-VM request from user





2. Nova-scheduler determines which compute node to allocate
3. Nova-compute launches GPU-VM using Libvirt with KVM PCI-passthrough on nova compute node

Fig. Nova architecture

Our aim

OpenPOWER

OpenStack uses KVM → Can POWER8 too uses KVM? → Can we integrate POWER8 to OpenStack?

PCI Passthrough with KVM

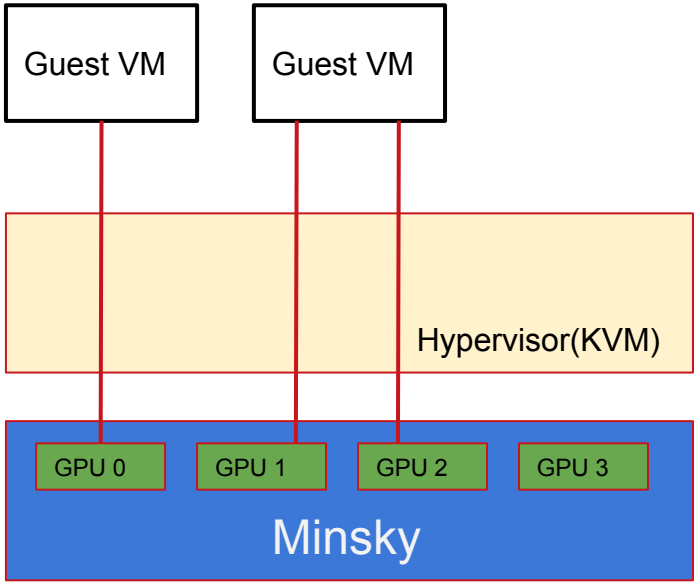


Fig. Passthrough with KVM

Verification Environment :

Server	IBM Power System S822LC(Minsky)
OS	Ubuntu 17.10
KVM Version	1:2.10+dfsg-0ubuntu3.1
GPU	NVIDIA P100
Firmare	Upgrade to Beta Version

PCI Passthrough with KVM(Sample Output)

OpenPOWER

This gives you a Non-GPU virtual machine.

It is a script to create virtual machine
Virsh XML specification to run on Host

```
openstack@openstack:~/LABSVC/tools$ ./create_vm.sh -s P -n NON_GPU_VM -i 152 -c 4 -m 4 -d 5
creating NON_GPU_VM with ip 152
NON_GPU_VM-P.qcow2 NON_GPU_VM-cldinit.img NON_GPU_VM-cldinit.u NON_GPU_VM-cldinit.m f4e03c01-faa1-43b3-8b30-51f35e86a279
/home/openstack/LABSVC/NON_GPU_VM/NON_GPU_VM-P.qcow2 /home/openstack/LABSVC/NON_GPU_VM/NON_GPU_VM-cldinit.img 85:98
creating the image now .. this might take a bit
Image resized.
openstack@openstack:~/LABSVC/tools$ cd ..
openstack@openstack:~/LABSVC$ cd NON_GPU_VM/
openstack@openstack:~/LABSVC/NON_GPU_VM$ sudo virsh create NON_GPU_VM.xml
setlocale: No such file or directory
Domain NON_GPU_VM created from NON_GPU_VM.xml

openstack@openstack:~/LABSVC/NON_GPU_VM$ sudo virsh console NON_GPU_VM
setlocale: No such file or directory
Connected to domain NON_GPU_VM
Escape character is ^]

Ubuntu 16.04.3 LTS ubuntu hvc0

ubuntu login:
```

Create a domain from XML file

Connect to guest console

```
openstack@openstack:~/LABSVC/4_GPU_VM$ sudo virsh list
setlocale: No such file or directory
 Id    Name                           State
----------------------------------------------------
 3     NON_GPU_VM                     running
```

Connect to guest console

PCI Passthrough with KVM(Sample Output)

- Reusable image is created and CUDA has installed and tested.
- *.qcow2 which is under LABSVC/CUDA_VM/CUDA_VM-G.qcow2 has copied into image directory so that it can be reused.
- Now a virtual image can be repeatedly created from the “snapshot”

Users don’t need to install appropriate Nvidia-driver every time while creating VM.

```
openstack@openstack:~/LABSVC/CUDA_VM$ sudo virsh list
setlocale: No such file or directory
 Id    Name                           State
----------------------------------------------------
 3     NON_GPU_VM                     running
 12    CUDA_VM                        running
```

VM with CUDA is running

This gives you GPU virtual machine.

This Command scans PCI bus for appropriate P100 GPUs and presents them to users to choose from.

More space will be needed

```
openstack@openstack:~/LABSVC/tools$ ./create_vm.sh -s G -n 4_GPU_VM -i 152 -c 4 -m 128 -d 30
creating 4_GPU_VM with ip 152
4_GPU_VM-G.qcow2 4_GPU_VM-cldinit.img 4_GPU_VM-cldinit.u 4_GPU_VM-cldinit.m 447f7e47-38b3-458c-a36f-eae4cb82b2f1
/home/openstack/LABSVC/4_GPU_VM/4_GPU_VM-G.qcow2 /home/openstack/LABSVC/4_GPU_VM/4_GPU_VM-cldinit.img 85:98
gpuspec is D gpuspec.spec
0: 0002:01:00.0
1: 0003:01:00.0
2: 000a:01:00.0
3: 000b:01:00.0
select all gpus you want to add (separated by space): 0 1 2 3
```

1, 2, 3 or all GPU cards can be selected

PCI Passthrough with KVM(Sample Output)

OpenPOWER

- 4 GPUs is assigned to 4_GPU_VM and it is currently running.

```
openstack@openstack:~/LABSVC/CUDA_VM$ sudo virsh list
setlocale: No such file or directory
 Id    Name                           State
----------------------------------------------------
 3     NON_GPU_VM                     running
 11    4_GPU_VM                       running
 12    CUDA_VM                        running
```

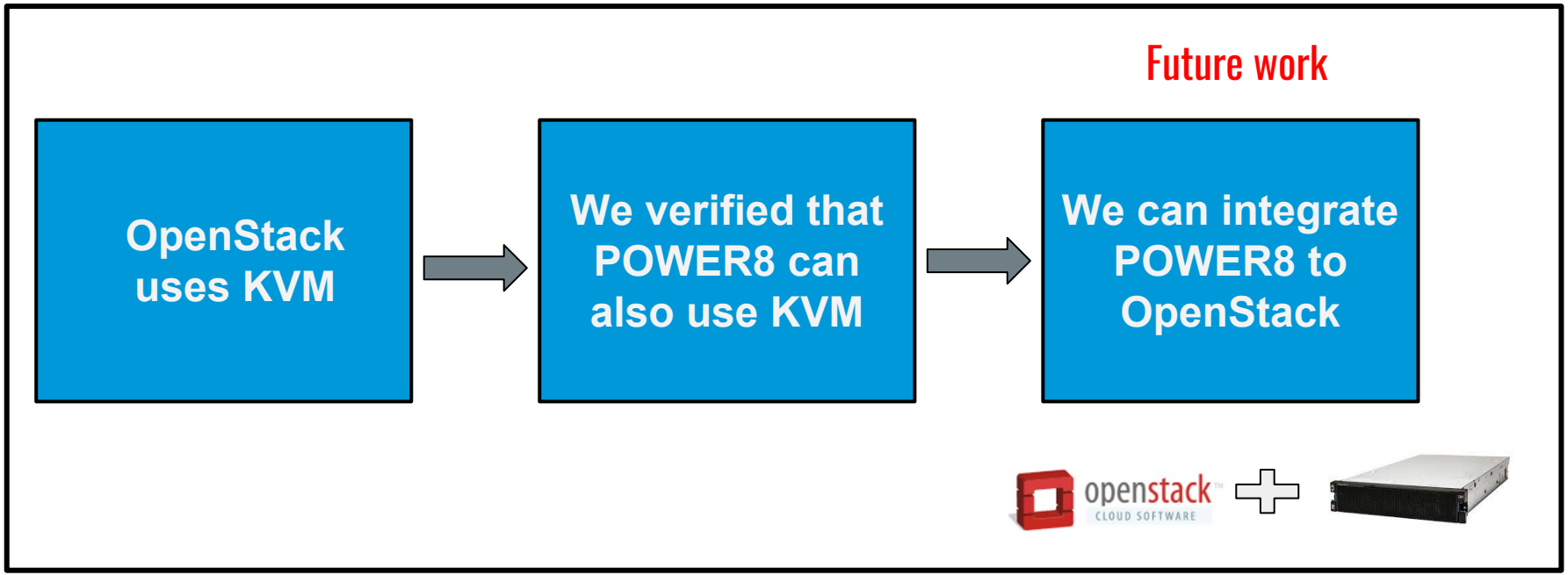
4_GPU_VM is in running state

- And after login to 4_GPU_VM, we can see 4 NVIDIA GPU cards

4*P100 GPU is accessed by 4_GPU_VM

```
ubuntu@4_GPU_VM:~$ lspci -nn | grep -i nvidia
00:08.0 3D controller [0302]: NVIDIA Corporation Device [10de:15f9] (rev a1)
00:0b.0 3D controller [0302]: NVIDIA Corporation Device [10de:15f9] (rev a1)
00:0e.0 3D controller [0302]: NVIDIA Corporation Device [10de:15f9] (rev a1)
00:11.0 3D controller [0302]: NVIDIA Corporation Device [10de:15f9] (rev a1)
```

Future work



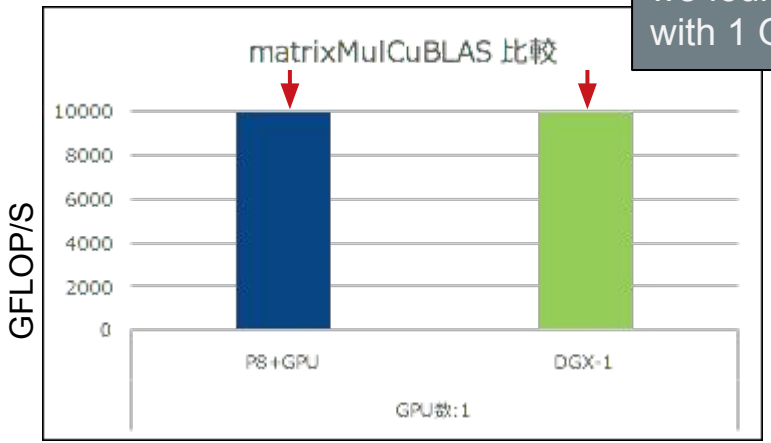
Agenda

1. Our Background
2. Providing GPU Resources: P100
3. Benchmarking with different tool : DGX-1 vs S822LC for HPC(Minsky)
4. I/O advantages of IBM Power System
5. Performance improvement with Memory Interleave: nbody
6. Summary

Benchmarking with MatrixMulCuBLAS

Overload: MatrixA(12800,9600), MatrixB(9600,6400), MatrixC(12800,6400)

↑ Higher is Better!!



For both POWER8 and DGX-1, we found the same performance with 1 GPU

- Since this program compares the performance of only one GPU, the performance is equivalent in both models equipped with the same GPU (Tesla P100)
- It contains simple matrix calculations so memory amount and bus width do not affect the results.

Benchmarking with GROMACS

OpenPOWER

How to build GROMACS

Build CMAKE

Building FFTW

Building GROMACS

Sample data for the simulation

Running Jobs

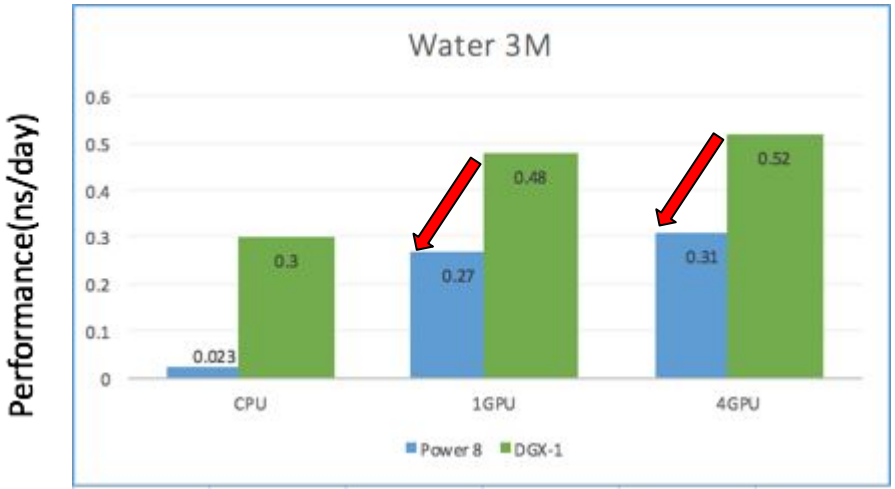
```
$ cat gromacs-all-in-one.sh
#!/bin/bash -x

wget http://cmake.org/files/v3.5/cmake-3.5.2.tar.gz
tar zxvf cmake-3.5.2.tar.gz
cd cmake-3.5.2
./configure
make
cd ..

wget ftp://ftp.fftw.org/pub/fftw/fftw-3.3.5.tar.gz
tar zxvf fftw-3.3.5.tar.gz
cd fftw-3.3.5
./configure --enable-float
make
make install
cd ..

wget ftp://ftp.gromacs.org/pub/gromacs/gromacs-5.1.4.tar.gz
tar zxvf gromacs-5.1.4.tar.gz
mkdir build
cd build
cmake /downloads/gromacs-5.1.4 -DGMX_BUILD_OWN_FFTW=ON -DGMX_GPU=on -
    DCUDA_TOOLKIT_ROOT_DIR=/usr/local/cuda
make
make install
cd ..

wget ftp://ftp.gromacs.org/pub/benchmarks/rnase_bench_systems.tar.gz
tar zxvf rnase_bench_systems.tar.gz

cd rnase_dodec
source ${CURDIR}/gromacs-5.1.4_binary/bin/GMXRC
gmx grompp -f pme_verlet.mdp -c conf.gro
gmx mdrun -s topol.tpr -v -ntmpi # -ntomp # -pin on
cd ..
```

-It's simulation software for benchmarking and measuring performance using different workloads such as proteins, lipids and nucleic acids

Benchmarking with Gromacs v5.1

Workload: Comparison by sample simulation "3M Water Size" (water molecules)

↑ Higher is Better!!



Graph. GPU performance comparison with Gromacs v5.1

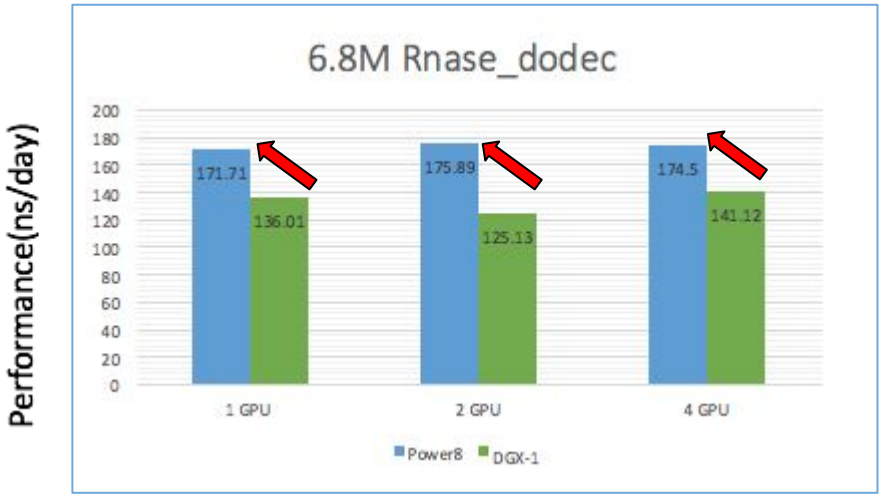
- When using this program, DGX - 1 generally got high speed results as compare to POWER8 system for 1 GPU as well as for 4 GPUs.

- It is considered to be caused by the code of GROMACS v5.1 which is not optimized for Power system and also not optimized for softwares such as compiler, library, etc.

Benchmarking with Gromacs v2016.3

Workload: Comparison by sample simulation " 6.8M Rnase_dodec data"

Higher is Better!!



Graph. GPU performance comparison with Gromacs v2016.3

- POWER8 has more performance in case of 1 GPU, 2 GPUs and 4 GPUs than DGX-1 for GROMACS v2016.3
- GROMACS v2016.3 is optimized for POWER8 system.
- Fully SIMD CPU code for Power is available from this version

Benchmarking with nbody

- Nbody is kind of cuda sample program.
- This program can calculate single precision and double precision by using GPU and the results are displayed in GFLOPS.
- It can be also calculated by CPU only.

How to run nbody

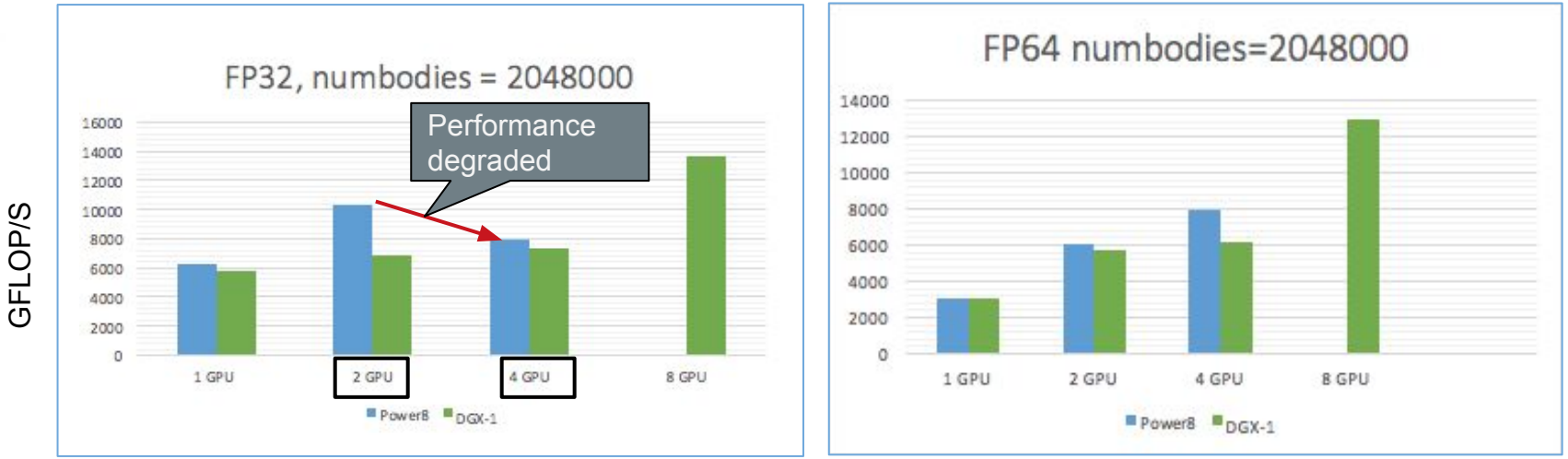
```
$ ./nbody -benchmark -numbodies=2048000 -numdevices=1
```

-benchmark : (run benchmark to measure performance)
-numbodies : (number of bodies (>= 1) to run in simulation)
(for GPU benchmark:2048000, for CPU benchmark:20480)
-numdevice : (where i=(number of CUDA devices > 0) to use for simulation)
-cpu : (run n-body simulation on the CPU)]
-fp64 : (use double precision floating point values for simulation)

Benchmarking with nbody

Workload: Comparison by numbodies=2048000, FP32

↑ Higher is Better!!



Graph. GPU performance comparison with nbody for single and double precision.

- Although, performance of Power 8 increases up to 2 GPUs, but performance drops at single precision with 4 GPUs

Why such performance degradation and how to tackle it??

IBM lab Services has solved this problem , Please listen Part 2..

Thank you so much for listening.

Contact me at : a.purohit@ntt.com