Revolutionizing the Datacenter

OpenPOWER™
Resource Disaggregated Platform (RD Platform) Including Power8 and GPUs for Diverse Cloud Computing Service

Takashi Yoshikawa*
Senior Principal Researcher, NEC Corporation
(* and Guest Professor, Osaka University)

Revolutionizing the Datacenter
Resource Disaggregated Platform In-Service

- 64 servers and 70 devices in resource pool over 6 racks connected via 236 PCIe (ExpEther).
Multi-Rack-Scale Software-Defined HPC

- Cost-effective implementation of HPC cloud.
  - Scale-up performance in need by allocating PCI Express devices at resource pool.
  - Share expensive devices among multiple racks.

```
<table>
<thead>
<tr>
<th>Rack #1</th>
<th>Rack #2</th>
<th>Rack #3</th>
<th>Rack #4</th>
<th>Rack #5</th>
<th>Rack #6</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOR L2 Switch</td>
<td>TOR L2 Switch</td>
<td>TOR L2 Switch</td>
<td>TOR L2 Switch</td>
<td>TOR L2 Switch</td>
<td>TOR L2 Switch</td>
</tr>
<tr>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
</tr>
<tr>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
<td>GPU</td>
</tr>
<tr>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
</tr>
<tr>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
<td>40T SAS JBOD</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
<td>Server</td>
</tr>
</tbody>
</table>

Device Pool: 70 devices
Server Pool: 64 servers

Resource Manager
System configuration along with user requirement

236 PCIe over 10/40G
Ethernet

Interconnect

Software Define

CPU
GPU
FPGA
Flash
SD server
ExpEther: Distributed PCIe Switch Architecture

- PCIe compliant single hop switch w/ km, k-port.
  - Combination of up/down bridge and Ethernet transport
  - Ethernet is transparent even for multi-hop switches.
ExpEther: Reliable Ethernet

- Reliable transport on standard Ethernet by delay-based congestion control and retry.
- xN bandwidth, redundancy by multipath

Monitoring RTT to control packet rate

Restoration at Path Error

xN performance using N paths

ExpEther: PCI Express® over Ethernet
ExpEther: Resource Allocation by Group ID

- PCIe tree is automatically constructed among ExpEther chips having the same group ID.
  - Group ID can be prefixed or configured by management software.

*Note that this figure omits the EE chips*
OpenStack Based Resource Management

- Modify Ironic (Bare metal control) to device level.

Modify Ironic to control each device.

GUI on modified Horizon
Implementation Lineup

1G/10G ExpEther

- **ExpEther HBA**
  - x1 PCI Express
  - Dual 1000BASE-T
- **ExpEther Client**
  - 2x 1000BASE-T
  - DVIx1, HDMI x1
  - USB3.0 x1
  - USB2.0 x3
  - Headphone x1
  - Microphone x1

- **ExpEther IO Expansion Unit**
  - x16 PCIe x 1 slot
  - Dual 1000BASE-T
  - x16 PCIe2 x 2 slots (full height/full length)
  - Dual 10G SFP+ per slot

40G ExpEther (New !)

- **ExpEther HBA**
  - x8 PCIe Gen2
  - Dual 10G SFP+

Join the conversation at #OpenPOWERSummit
Power8 Server and 2x40G ExpEther

40G Ethernet switch

Power8

ExpEther expansion unit

ExpEther HBA

k80
Logical View

- ExpEther’s logical view is standard PCIe switch
  - A pair of PCI bridge
- NVIDIA k80 as a local device at local PCIe bus
CUDA Benchmark: N-Body for k80

- Comparable performance to local GPU

### Cuda N-Body (single-precision)

<table>
<thead>
<tr>
<th>Number of Particle</th>
<th>Local PCIe slot</th>
<th>ExpEther40G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,024</td>
<td>101.583</td>
<td>99.523</td>
</tr>
<tr>
<td>13,312</td>
<td>1044.422</td>
<td>1045.575</td>
</tr>
<tr>
<td>100,096</td>
<td>1483.333</td>
<td>1467.523</td>
</tr>
<tr>
<td>200,192</td>
<td>1628.356</td>
<td>1622.648</td>
</tr>
</tbody>
</table>

- Local PCIe Slot: % 97.97%
- ExpEther40G: % 100.11%
- % 98.93%
- % 99.65%
Multi GPUs in Resource Disaggregated Platform

- Easy programming, high-performance (vs. GPU Cluster)
  - All GPUs exist in local. Direct memory access (DMA).
- Cost-effective (vs. GPU datecated Machine)
  - Doesn’t need GPU-special machine, Share expensive GPU
  - Latency degrades performance
Two Application Model using GPUs

- Particle motion simulation by Runge-Kutta method
  - N-step GPU computing w/ only 2-time data transfer
  - no device communication

- Advection term calculation by Cubic Lagrange Interpolation
  - device communication in each iteration step
  - influenced by latency

Particle motion simulation in 8-GPU system

- 8 GPUs achieved speedup of 7.13 times
  - Linearly scale up proportional to the number of GPUs
    - the overhead of data transfer from/to host is dominant @ small size

Overhead of Data transfer

X 7.13 performance

Update/Second (10^9)

Problem Size

- 1x1024x1024, 100 step
- 1x1024x1024, 1000 step
- 10x1024x1024, 1000 step
- 10x1024x1024, 2000 step
- 10x1024x1024, 4000 step

# of GPUs
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8

GPGPU BOX (Proto Type)

Latency Masking of Data Communication

- Reduce frequency of communication
- Overlap communication and GPU processing
- GPU direct communication w/o host node

---

Level synchronized Breadth First Graph Search

- Comparable Scalability to GPU machine for big graph.
  - 80% of communication time is successfully masked

4-GPU machine

<table>
<thead>
<tr>
<th>TEPS (Traversed Edged Per Second)</th>
<th>Resource Disaggregated</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPUUs=1</td>
<td>GPUUs=1</td>
</tr>
<tr>
<td>GPUUs=2</td>
<td>GPUUs=2</td>
</tr>
<tr>
<td>MTEPS</td>
<td>MTEPS</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>1000</td>
<td>1000</td>
</tr>
</tbody>
</table>

| graph scale (|V|=2^sc) | graph scale (|V|=2^sc) |
|--------------|------------------|
| 15 17 19 21 23 | 15 17 19 21 23 |

TEPS(Traversed Edged Per Second)

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>root</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>current frontier</th>
<th>next frontier</th>
</tr>
</thead>
<tbody>
<tr>
<td>: current frontier</td>
<td>: next frontier</td>
</tr>
</tbody>
</table>

2016/03/07

High-Speed storage using PCI SSD

- Same IOPS for Random Read @ 4kB
  - Local: \( \text{bw}=1814.3\, \text{MB/s}, \text{iops}=464436 \)
  - ExpEther: \( \text{bw}=1810.3\, \text{MB/s}, \text{iops}=463428 \)
  - Catalog: \( \text{iops} = 450,000 \)

\[\text{Table 4: Random Read/Write Input/Output Operations Per Second (IOPS)}\]

<table>
<thead>
<tr>
<th>Specification (^1)</th>
<th>Unit</th>
<th>Intel SSD DC P3700 Series</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>400GB</td>
</tr>
<tr>
<td>Random 4KB 70/30 Read/Write (up to) (^2)</td>
<td>IOPS</td>
<td>150,000</td>
</tr>
<tr>
<td>Random 8KB 70/30 Read/Write (up to) (^3)</td>
<td>IOPS</td>
<td>75,000</td>
</tr>
<tr>
<td>Random 4KB Read (up to) (^2)</td>
<td>IOPS</td>
<td>450,000</td>
</tr>
<tr>
<td>Random 4KB Write (up to)</td>
<td>IOPS</td>
<td>75,000</td>
</tr>
<tr>
<td>Random 8KB Read (up to) (^3)</td>
<td>IOPS</td>
<td>275,000</td>
</tr>
<tr>
<td>Random 8KB Write (up to)</td>
<td>IOPS</td>
<td>32,000</td>
</tr>
</tbody>
</table>

\(^1\) Specification: Random 4KB 70/30 Read/Write (up to) and Random 8KB 70/30 Read/Write (up to) refer to the percentage of reads and writes that are sequential. Random 4KB Read (up to) and Random 8KB Read (up to) refer to the percentage of reads that are sequential.

\(^2\) Up to: Indicates the maximum performance when the specified size is accessed.

\(^3\) Up to: Indicates the maximum performance when the specified size is accessed.
Resource Disaggregated Storage (RD-Store)

- kvs over RD NVMe
  - Individually increase control CPUs / storage devices.
  - Scale up immediately when add servers independent of the amount of data.

8threads/server: with NVMe 1.1 card 302kOPS, GET 89 μsec, PUT 263μsec

YCSB(Yahoo! Cloud Service Benchmark), Read Heavy Workload (100B): Random GET 95%, PUT 5%

1-4 Control CPUs

1-5 devices

Interconnect(ExpEther)
Conclusion: Resource Disaggregated Platform

- The RD Platform expands the use of cloud data centers not only office applications but also high performance computing.
- The RD platform performs computation by allocating devices from a resource pool at the device level to scale up individual performance and functionality.
- Since the fabric is ExpEther (distributed PCIe switch over Ethernet), open standard hardware and software can be utilized to build customer’s computer systems.
- A combination of the latest Gen3x8 – 2x 40GE ExpEther and power8 server show potential for intensive computing power.
- The performance of both 8-GPU machine and KVS with shared PCIe-storage show the easy-to-scale-up feature in datacenter.
Acknowledgement

I would like to thank all the members working on developing this technology

- Hideharu Amano at Keio University
- Susumu Date and Shinji Shimojo at Osaka University
- Richard Solomon, Hiroki Iwasaki, Hironori Ando, Takashi Yagi, Hiroshi Ono, Mitsuhiro Miyata, and other Synopsys People
- Takaaki Terao, Tsuyoshi Yoshioka, Koetsu Narisawa, Nobuhiro Numata, Yosuke Yano, Takayuki Saito, Osamu Nara, Takumi Kosawa, and other ID People
- Yoich Hidaka, Junich Higuch, Jun Suzuki, Yuki Hayashi, Masahiko Takahashi, Masaki Kan, Akira Tusji, Shinya Miyakawa, Nobuharu Kami, Teruyuki Baba, Nobuyuki Enomoto, Hideyuki Shimonishi, Atsushi Iwata, Shinji Abe, Shinji Ueno, Hiroyoshi Yamane, Ryuta Niino, Kiyoshi Baba, Hiroki Yokoyama, Tatsuya Takada, Shiro Momot, Taka Koishi, Yasushi Kikkawa, Yoshimitsu Okayama, Hiroshi Baba, Tsukasa Kimura, Masayuki Terao, and other NEC people