HPC Meets Cloud: Opportunities and Challenges in Designing High-Performance MPI and Big Data Libraries on Virtualized InfiniBand Clusters

Keynote Talk at CloudCom (December 2016)

by

Dhabaleswar K. (DK) Panda

The Ohio State University

E-mail: panda@cse.ohio-state.edu

http://www.cse.ohio-state.edu/~panda
High-End Computing (HEC): ExaFlop & ExaByte

Expected to have an ExaFlop system in 2021!

Exabyte & BigData

Trends for Commodity Computing Clusters in the Top 500 List (http://www.top500.org)

Timeline

Number of Clusters

Percentage of Clusters

Number of Clusters

Timeline

Nov-96  Jan-98  Mar-00  Aug-01  Mar-03  Oct-04  May-06  Dec-07  Jul-09  Feb-11  Sep-12  Apr-14  Nov-15

Percentage of Clusters

86%
Drivers of Modern HPC Cluster Architectures

- Multi-core/many-core technologies
- Remote Direct Memory Access (RDMA)-enabled networking (InfiniBand and RoCE)
- Solid State Drives (SSDs), Non-Volatile Random-Access Memory (NVRAM), NVMe-SSD
- Accelerators (NVIDIA GPGPUs and Intel Xeon Phi)
- Available on HPC Clouds, e.g., Amazon EC2, NSF Chameleon, Microsoft Azure, etc.
InfiniBand in the Top500 (Nov 2016)

**Number of Systems**
- InfiniBand: 37%
- 10G: 36%
- Custom Interconnect: 6%
- Omnipath: 6%
- Gigabit Ethernet: 14%
- Proprietary Network: 1%
- Ethernet: 0%

**Performance**
- InfiniBand: 27%
- 10G: 48%
- Custom Interconnect: 15%
- Omnipath: 2%
- Gigabit Ethernet: 2%
- Proprietary Network: 0%
- Ethernet: 6%
Large-scale InfiniBand Installations

- 187 IB Clusters (37%) in the Nov’16 Top500 list
  - (http://www.top500.org)
- Installations in the Top 50 (15 systems):

| 241,108 cores (Pleiades) at NASA/Ames (13th) | 147,456 cores (SuperMUC) in Germany (36th) |
| 220,800 cores (Pangea) in France (16th) | 86,016 cores (SuperMUC Phase 2) in Germany (37th) |
| 462,462 cores (Stampede) at TACC (17th) | 74,520 cores (Tsubame 2.5) at Japan/GSIC (40th) |
| 144,900 cores (Cheyenne) at NCAR/USA (20th) | 194,616 cores (Cascade) at PNNL (44th) |
| 72,800 cores Cray CS-Storm in US (25th) | 76,032 cores (Makman-2) at Saudi Aramco (49th) |
| 72,800 cores Cray CS-Storm in US (26th) | 72,000 cores (Prolix) at Meteo France, France (50th) |
| 124,200 cores (Topaz) SGI ICE at ERDC DSRC in US (27th) | 73,440 cores (Beaufix2) at Meteo France, France (51st) |
| 60,512 cores (DGX SATURNV) at NVIDIA/USA (28th) | 42,688 cores (Lomonosov-2) at Russia/MSU (52nd) |
| 72,000 cores (HPC2) in Italy (29th) | 60,240 cores SGI ICE X at JAEA Japan (54th) |
| 152,692 cores (Thunder) at AFRL/USA (32nd) | and many more! |
Cloud Computing and Virtualization

- Cloud Computing focuses on maximizing the effectiveness of the shared resources
- Virtualization is the key technology for resource sharing in the Cloud
- Widely adopted in industry computing environment
HPC Cloud - Combining HPC with Cloud

• IDC expects that by 2017, HPC ecosystem revenue will jump to a record $30.2 billion. IDC foresees public clouds, and especially custom public clouds, supporting an increasing proportion of the aggregate HPC workload as these cloud facilities grow more capable and mature (Courtesy: http://www.idc.com/getdoc.jsp?containerId=247846)

• Combining HPC with Cloud is still facing challenges because of the performance overhead associated virtualization support
  – Lower performance of virtualized I/O devices

• HPC Cloud Examples
  – Amazon EC2 with Enhanced Networking
    • Using Single Root I/O Virtualization (SR-IOV)
    • Higher performance (packets per second), lower latency, and lower jitter
    • 10 GigE
  – NSF Chameleon Cloud
NSF Chameleon Cloud: A Powerful and Flexible Experimental Instrument

• Large-scale instrument
  – Targeting Big Data, Big Compute, Big Instrument research
  – ~650 nodes (~14,500 cores), 5 PB disk over two sites, 2 sites connected with 100G network

• Reconfigurable instrument
  – Bare metal reconfiguration, operated as single instrument, graduated approach for ease-of-use

• Connected instrument
  – Workload and Trace Archive
  – Partnerships with production clouds: CERN, OSDC, Rackspace, Google, and others
  – Partnerships with users

• Complementary instrument
  – Complementing GENI, Grid’5000, and other testbeds

• Sustainable instrument
  – Industry connections

http://www.chameleoncloud.org/
Chameleon Hardware

Chameleon Core Network
100Gbps uplink public network (each site)

Core Services
Front End and Data Mover Nodes

Core Services
3 PB Central File Systems, Front End and Data Movers

Heterogeneous Cloud Units
Alternate Processors and Networks

504 x86 Compute Servers
48 Dist. Storage Servers
102 Heterogeneous Servers
16 Mgt and Storage Nodes

To UTSA, GENI, Future Partners

Switch
Standard Cloud Unit
42 compute
4 storage
x2

Switch
Standard Cloud Unit
42 compute
4 storage
x10

SCUs connect to core and fully connected to each other

Chicago
Austin

42 compute
4 storage
x10

Network Based Computing Laboratory
CloudCom 2016
Capabilities and Supported Research on Chameleon

- Development of new models, algorithms, platforms, auto-scaling HA, etc., innovative application and educational uses

  *Persistent, reliable, shared clouds*

- Repeatable experiments in new models, algorithms, platforms, auto-scaling, high-availability, cloud federation, etc.

  *Isolated partition, pre-configured images reconfiguration*

- Virtualization technology (e.g., SR-IOV, accelerators), systems, networking, infrastructure-level resource management, etc.

  *Isolated partition, full bare metal reconfiguration*

- SR-IOV + InfiniBand
Single Root I/O Virtualization (SR-IOV)

- **Single Root I/O Virtualization (SR-IOV)** is providing new opportunities to design HPC cloud with very little low overhead.

- Allows a single physical device, or a Physical Function (PF), to present itself as multiple virtual devices, or Virtual Functions (VFs).

- VFs are designed based on the existing non-virtualized PFs, no need for driver change.

- Each VF can be dedicated to a single VM through PCI pass-through.

- Work with 10/40 GigE and InfiniBand.
Building HPC Cloud with SR-IOV and InfiniBand

• High-Performance Computing (HPC) has adopted advanced interconnects and protocols
  – InfiniBand
  – 10/40 Gigabit Ethernet/iWARP
  – RDMA over Converged Enhanced Ethernet (RoCE)

• Very Good Performance
  – Low latency (few micro seconds)
  – High Bandwidth (100 Gb/s with EDR InfiniBand)
  – Low CPU overhead (5-10%)

• OpenFabrics software stack with IB, iWARP and RoCE interfaces are driving HPC systems

• How to Build HPC Cloud with SR-IOV and InfiniBand for delivering optimal performance?
HPC and Big Data on Cloud Computing Systems: Challenges

Applications

HPC and Big Data Middleware

HPC (MPI, PGAS, MPI+PGAS, MPI+OpenMP, etc.)

Big Data (HDFS, MapReduce, Spark, HBase, Memcached, etc.)

Resource Management and Scheduling Systems for Cloud Computing
(OpenStack Nova, Swift, Heat; Slurm, etc.)

Communication and I/O Library

Communication Channels
(SR-IOV, IVShmem, IPC-Shm, CMA)

Locality-aware Communication

Virtualization
(Hypervisor and Container)

Data Placement & Task Scheduling

Fault-Tolerance
(Migration, Replication, etc.)

QoS-aware, etc.

Networking Technologies
(InfiniBand, Omni-Path, 1/10/40/100 GigE and Intelligent NICs)

Commodity Computing System Architectures
(Multi- and Many-core architectures and accelerators)

Storage Technologies
(HDD, SSD, NVRAM, and NVMe-SSD)
Broad Challenges in Designing Communication and I/O Middleware for HPC on Clouds

- Virtualization Support with Virtual Machines and Containers
  - KVM, Docker, Singularity, etc.
- Communication coordination among optimized communication channels on Clouds
  - SR-IOV, IVShmem, IPC-Shm, CMA, etc.
- Locality-aware communication
- Scalability for million processors
  - Support for highly-efficient inter-node and intra-node communication (both two-sided and one-sided)
- Scalable Collective communication
  - Offload
  - Non-blocking
  - Topology-aware
- Balancing intra-node and inter-node communication for next generation nodes (128-1024 cores)
  - Multiple end-points per node
- Support for efficient multi-threading
- Integrated Support for GPGPUs and Accelerators
- Fault-tolerance/resiliency
  - Migration support with virtual machines
- QoS support for communication and I/O
- Support for Hybrid MPI+PGAS programming (MPI + OpenMP, MPI + UPC, MPI + OpenSHMEM, MPI+UPC++, CAF, ...)
- Energy-Awareness
- Co-design with resource management and scheduling systems on Clouds
  - OpenStack, Slurm, etc.
Additional Challenges in Designing Communication and I/O Middleware for Big Data on Clouds

- High-Performance designs for Big Data middleware
  - RDMA-based designs to accelerate Big Data middleware on high-performance Interconnects
  - NVM-aware communication and I/O schemes for Big Data
  - SATA-/PCIe-/NVMe-SSD support
  - Parallel Filesystems support
  - Optimized overlapping among Computation, Communication, and I/O
  - Threaded Models and Synchronization

- Fault-tolerance/resiliency
  - Migration support with virtual machines
  - Data replication

- Efficient data access and placement policies

- Efficient task scheduling

- Fast deployment and automatic configurations on Clouds
Approaches to Build HPC Clouds

- MVAPICH2-Virt with SR-IOV and IVSHMEM
  - Standalone, OpenStack
- MVAPICH2 with Containers
- MVAPICH2-Virt on SLURM
  - SLURM alone, SLURM + OpenStack
- Big Data Libraries on Cloud
Overview of the MVAPICH2 Project

- High Performance open-source MPI Library for InfiniBand, Omni-Path, Ethernet/iWARP, and RDMA over Converged Ethernet (RoCE)
  - MVAPICH (MPI-1), MVAPICH2 (MPI-2.2 and MPI-3.0), Started in 2001, First version available in 2002
  - MVAPICH2-X (MPI + PGAS), Available since 2011
  - Support for GPGPUs (MVAPICH2-GDR) and MIC (MVAPICH2-MIC), Available since 2014
  - Support for Virtualization (MVAPICH2-Virt), Available since 2015
  - Support for Energy-Awareness (MVAPICH2-EA), Available since 2015
  - Support for InfiniBand Network Analysis and Monitoring (OSU INAM) since 2015
  - Used by more than 2,700 organizations in 83 countries
  - More than 404,000 (> 0.4 million) downloads from the OSU site directly
  - Empowering many TOP500 clusters (Nov ’16 ranking)
    - 1st ranked 10,649,640-core cluster (Sunway TaihuLight) at NSC, Wuxi, China
    - 13th ranked 241,108-core cluster (Pleiades) at NASA
    - 17th ranked 519,640-core cluster (Stampede) at TACC
    - 40th ranked 76,032-core cluster (Tsubame 2.5) at Tokyo Institute of Technology and many others
  - Available with software stacks of many vendors and Linux Distros (RedHat and SuSE)
- Empowering Top500 systems for over a decade
  - System-X from Virginia Tech (3rd in Nov 2003, 2,200 processors, 12.25 TFlops) -> Sunway TaihuLight at NSC, Wuxi, China (1st in Nov’16, 10,649,640 cores, 93 PFlops)

http://mvapich.cse.ohio-state.edu
MVAPICH2 Release Timeline and Downloads

Number of Downloads

Timeline

MV 0.9.4
MV2 0.9.0
MV2 0.9.8
MV2 1.0
MV2 1.0.3
MV 1.1
MV2 1.4
MV2 1.5
MV2 1.6
MV2 1.7
MV2 1.8
MV2 1.9
MV2-GDR 2.0b
MV2-MIC 2.0
MV2 2.1
MV2-GDR 2.2rc1
MV2-Virt 2.1rc2
MV2-GDR 2.2rc1
MV2-X

Network Based Computing Laboratory
CloudCom 2016
## MVAPICH2 Architecture

### High Performance Parallel Programming Models

<table>
<thead>
<tr>
<th>Message Passing Interface (MPI)</th>
<th>PGAS (UPC, OpenSHMEM, CAF, UPC++)</th>
<th>Hybrid --- MPI + X (MPI + PGAS + OpenMP/Cilk)</th>
</tr>
</thead>
</table>

### High Performance and Scalable Communication Runtime

<table>
<thead>
<tr>
<th>Diverse APIs and Mechanisms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point-to-point Primitives</td>
</tr>
</tbody>
</table>

### Support for Modern Networking Technology

*(InfiniBand, iWARP, RoCE, OmniPath)*

- Transport Protocols: RC, XRC, UD, DC
- Modern Features: UMR, ODP, SR-IOV, Multi Rail

### Support for Modern Multi-/Many-core Architectures

*(Intel-Xeon, OpenPower, Xeon-Phi (MIC, KNL), NVIDIA GPGPU)*

- Transport Mechanisms: Shared Memory, CMA, IVSHMEM
- Modern Features: MCDRAM*, NVLink*, CAPI*
## MVAPICH2 Software Family

<table>
<thead>
<tr>
<th>High-Performance Parallel Programming Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVAPICH2</td>
</tr>
<tr>
<td>MVAPICH2-X</td>
</tr>
<tr>
<td>MVAPICH2-GDR</td>
</tr>
<tr>
<td>MVAPICH2-Virt</td>
</tr>
<tr>
<td>MVAPICH2-EA</td>
</tr>
<tr>
<td>MVAPICH2-MIC</td>
</tr>
</tbody>
</table>

### Microbenchmarks

<table>
<thead>
<tr>
<th>Microbenchmarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMB</td>
</tr>
</tbody>
</table>

### Tools

<table>
<thead>
<tr>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSU INAM</td>
</tr>
<tr>
<td>OEMT</td>
</tr>
</tbody>
</table>
HPC on Cloud Computing Systems: Challenges Addressed by OSU So Far

Applications

HPC and Big Data Middleware

HPC (MPI, PGAS, MPI+PGAS, MPI+OpenMP, etc.)

Resource Management and Scheduling Systems for Cloud Computing
(OpenStack Nova, Heat; Slurm)

Communication and I/O Library

Communication Channels
(SR-IOV, IVShmem, IPC-Shm, CMA)

Locality-aware Communication

Virtualization
(Hypervisor and Container)

Fault-Tolerance & Consolidation
(Migration)

QoS-aware

Future Studies

Networking Technologies
(InfiniBand, Omni-Path, 1/10/40/100 GigE and Intelligent NICs)

Commodity Computing System Architectures
(Multi- and Many-core architectures and accelerators)

Storage Technologies
(HDD, SSD, NVRAM, and NVMe-SSD)

Network Based Computing Laboratory

CloudCom 2016
Overview of MVAPICH2-Virt with SR-IOV and IVSHMEM

- Redesign MVAPICH2 to make it virtual machine aware
  - SR-IOV shows near to native performance for inter-node point to point communication
  - IVSHMEM offers shared memory based data access across co-resident VMs
  - Locality Detector: maintains the locality information of co-resident virtual machines
  - Communication Coordinator: selects the communication channel (SR-IOV, IVSHMEM) adaptively


MVAPICH2-Virt with SR-IOV and IVSHMEM over OpenStack

- OpenStack is one of the most popular open-source solutions to build clouds and manage virtual machines
- Deployment with OpenStack
  - Supporting SR-IOV configuration
  - Supporting IVSHMEM configuration
  - Virtual Machine aware design of MVAPICH2 with SR-IOV
- An efficient approach to build HPC Clouds with MVAPICH2-Virt and OpenStack

## Performance Evaluation

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Nowlab Cloud</th>
<th>Amazon EC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance</td>
<td>4 Core/VM</td>
<td>8 Core/VM</td>
</tr>
<tr>
<td>Platform</td>
<td>RHEL 6.5 Qemu+KVM HVM SLURM 14.11.8</td>
<td>Amazon Linux (EL6) Xen HVM C3.xlarge [1] Instance</td>
</tr>
<tr>
<td>CPU</td>
<td>SandyBridge Intel(R) Xeon E5-2670 (2.6GHz)</td>
<td>IvyBridge Intel(R) Xeon E5-2680v2 (2.8GHz)</td>
</tr>
<tr>
<td>RAM</td>
<td>6 GB</td>
<td>12 GB</td>
</tr>
</tbody>
</table>

[1] Amazon EC2 C3 instances: compute-optimized instances, providing customers with the highest performing processors, good for HPC workloads

[2] Nowlab Cloud is using InfiniBand FDR (56Gbps), while Amazon EC2 C3 instances are using 10 GigE. Both have SR-IOV
Experiments Carried Out

• Point-to-point
  – Two-sided and One-sided
  – Latency and Bandwidth
  – Intra-node and Inter-node \[1\]

• Applications
  – NAS and Graph500

\[1\] Amazon EC2 does not support users to explicitly allocate VMs in one physical node so far. We allocate multiple VMs in one logical group and compare the point-to-point performance for each pair of VMs. We see the VMs who have the lowest latency as located within one physical node (Intra-node), otherwise Inter-node.
Point-to-Point Performance – Latency & Bandwidth (Intra-node)

- EC2 C3.2xlarge instances
- Compared to SR-IOV-Def, up to 84% and 158% performance improvement on Lat & BW
- Compared to Native, 3%-7% overhead for Lat, 3%-8% overhead for BW
- Compared to EC2, up to 160X and 28X performance speedup on Lat & BW
Point-to-Point Performance – Latency & Bandwidth (Inter-node)

- EC2 C3.2xlarge instances
- Similar performance with SR-IOV-Def
- Compared to Native, 2%-8% overhead on Lat & BW for 8KB+ messages
- Compared to EC2, up to 30X and 16X performance speedup on Lat & BW
Application-Level Performance (8 VM * 8 Core/VM)

- Compared to Native, 1-9% overhead for NAS
- Compared to Native, 4-9% overhead for Graph500
### Application-Level Performance on Chameleon

#### Graph500

<table>
<thead>
<tr>
<th>Problem Size (Scale, Edgefactor)</th>
<th>MV2-SR-IOV-Def</th>
<th>MV2-SR-IOV-Opt</th>
<th>MV2-Native</th>
</tr>
</thead>
<tbody>
<tr>
<td>22,20</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>24,10</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>24,16</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>24,20</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>26,10</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>26,16</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
</tbody>
</table>

#### SPEC MPI2007

<table>
<thead>
<tr>
<th>Application</th>
<th>MV2-SR-IOV-Def</th>
<th>MV2-SR-IOV-Opt</th>
<th>MV2-Native</th>
</tr>
</thead>
<tbody>
<tr>
<td>milc</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>leslie3d</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>pop2</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>GAPgeofem</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>zeusmp2</td>
<td>9.5%</td>
<td>5%</td>
<td>2%</td>
</tr>
<tr>
<td>lu</td>
<td>1%</td>
<td>5%</td>
<td>2%</td>
</tr>
</tbody>
</table>

- 32 VMs, 6 Core/VM
- Compared to Native, 2-5% overhead for Graph500 with 128 Procs
- Compared to Native, 1-9.5% overhead for SPEC MPI2007 with 128 Procs
Approaches to Build HPC Clouds

- MVAPICH2-Virt with SR-IOV and IVSHMEM
  - Standalone, OpenStack

- MVAPICH2 with Containers

- MVAPICH2-Virt on SLURM
  - SLURM alone, SLURM + OpenStack

- Big Data Libraries on Cloud
Overview of Containers-based Virtualization

- Container-based technologies (e.g., Docker) provide lightweight virtualization solutions
- Container-based virtualization – share host kernel by containers
Benefits of Containers-based Virtualization for HPC on Cloud

- Experiment on NFS Chameleon Cloud
- Container has less overhead than VM
- BFS time in Graph 500 significantly increases as the number of container increases on a host. Why?

Containers-based Design: Issues, Challenges, and Approaches

- What are the performance bottlenecks when running MPI applications on multiple containers per host in HPC cloud?
- Can we propose a new design to overcome the bottleneck on such container-based HPC cloud?
- Can optimized design deliver near-native performance for different container deployment scenarios?
- Locality-aware based design to enable CMA and Shared memory channels for MPI communication across co-resident containers

Containers Support: MVAPICH2 Intra-node Inter-Container Point-to-Point Performance on Chameleon

- Intra-Node Inter-Container
- Compared to Container-Def, up to 81% and 191% improvement on Latency and BW
- Compared to Native, minor overhead on Latency and BW
Containers Support: MVAPICH2 Collective Performance on Chameleon

- 64 Containers across 16 nodes, pinning 4 Cores per Container
- Compared to Container-Def, up to 64% and 86% improvement on Allreduce and Allgather
- Compared to Native, minor overhead on Allreduce and Allgather
Containers Support: Application-Level Performance on Chameleon

- 64 Containers across 16 nodes, pinning 4 Cores per Container
- Compared to Container-Def, up to 11% and 73% of execution time reduction for NAS and Graph 500
- Compared to Native, less than 9% and 5% overhead for NAS and Graph 500
**MVAPICH2-Virt 2.2rc1**

- Major Features and Enhancements
  - Based on MVAPICH2 2.2rc1
  - Support for efficient MPI communication over SR-IOV enabled InfiniBand networks
  - High-performance and locality-aware MPI communication with IVSHMEM for virtual machines
  - High-performance and locality-aware MPI communication with IPC-SHM and CMA for containers
  - Support for auto-detection of IVSHMEM device in virtual machines
  - Support for locality auto-detection in containers
  - Automatic communication channel selection among SR-IOV, IVSHMEM, and CMA/LiMIC2 in virtual machines
  - Automatic communication channel selection among IPC-SHM, CMA, and HCA in containers
  - Support for integration with OpenStack
  - Support for easy configuration through runtime parameters
  - Tested with
    - Docker 1.9.1 and 1.10.3
    - Mellanox InfiniBand adapters (ConnectX-3 (56Gbps))
    - OpenStack Juno
  - **Available from http://mvapich.cse.ohio-state.edu**
  - **Will be updated to the latest MVAPICH2 2.2 GA version soon**
Approaches to Build HPC Clouds

- MVAPICH2-Virt with SR-IOV and IVSHMEM
  - Standalone, OpenStack
- MVAPICH2 with Containers
- MVAPICH2-Virt on SLURM
  - SLURM alone, SLURM + OpenStack
- Big Data Libraries on Cloud
Can HPC Clouds be built with MVAPICH2-Virt on SLURM?

- SLURM is one of the most popular open-source solutions to manage huge amounts of machines in HPC clusters.
- How to build a SLURM-based HPC Cloud with near native performance for MPI applications over SR-IOV enabled InfiniBand HPC clusters?
- What are the requirements on SLURM to support SR-IOV and IVSHMEM provided in HPC Clouds?
- How much performance benefit can be achieved on MPI primitive operations and applications in “MVAPICH2-Virt on SLURM”-based HPC clouds?
Typical Usage Scenarios

- **Exclusive Allocations**: Sequential Jobs
- **Shared-host Allocations**: Concurrent Jobs
- **Exclusive Allocations**: Concurrent Jobs

Compute Nodes

Exclusive Allocations
- Sequential Jobs

Shared-host Allocations
- Concurrent Jobs

Compute Nodes

Exclusive Allocations
- Concurrent Jobs

Exclusive Allocations
- Sequential Jobs

Shared-host Allocations
- Concurrent Jobs

VM

MPI

VF1

VF2

IVSHMEM-1

IVSHMEM-2

Typical Usage Scenarios

Network Based Computing Laboratory

CloudCom 2016
Need for Supporting SR-IOV and IVSHMEM in SLURM

- Requirement of managing and isolating virtualized resources of SR-IOV and IVSHMEM
- Such kind of management and isolation is hard to be achieved by MPI library alone, but much easier with SLURM
- Efficient running MPI applications on HPC Clouds needs SLURM to support managing SR-IOV and IVSHMEM
  - Can critical HPC resources be efficiently shared among users by extending SLURM with support for SR-IOV and IVSHMEM based virtualization?
  - Can SR-IOV and IVSHMEM enabled SLURM and MPI library provide bare-metal performance for end applications on HPC Clouds?
Workflow of Running MPI Jobs with MVAPICH2-Virt on SLURM

1. SR-IOV virtual function
2. IVSHMEM device
3. Network setting
4. Image management
5. Launching VMs and check availability
6. Mount global storage, etc.
• **VM Configuration Reader** – Register all VM configuration options, set in the job control environment so that they are visible to all allocated nodes.

• **VM Launcher** – Setup VMs on each allocated nodes.
  - File based lock to detect occupied VF and exclusively allocate free VF
  - Assign a unique ID to each IVSHMEM and dynamically attach to each VM

• **VM Reclaimer** – Tear down VMs and reclaim resources
Benefits of Plugin-based Designs for SLURM

- **Coordination**
  - With global information, SLURM plugin can manage SR-IOV and IVSHMEM resources easily for concurrent jobs and multiple users

- **Performance**
  - Faster coordination, SR-IOV and IVSHMEM aware resource scheduling, etc.

- **Scalability**
  - Taking advantage of the scalable architecture of SLURM

- **Fault Tolerance**

- **Permission**

- **Security**
• VM Configuration Reader – VM options register
• VM Launcher, VM Reclaimer – Offload to underlying OpenStack infrastructure
  - PCI Whitelist to passthrough free VF to VM
  - Extend Nova to enable IVShmem when launching VM

Benefits of SLURM Plugin-based Designs with OpenStack

• Easy Management
  – Directly use underlying OpenStack infrastructure to manage authentication, image, networking, etc.

• Component Optimization
  – Directly inherit optimizations on different components of OpenStack

• Scalability
  – Taking advantage of the scalable architecture of both OpenStack and SLURM

• Performance
Comparison on Total VM Startup Time

- Task - implement and explicitly insert three components in job batch file
- SPANK - SPANK Plugin based design
- SPANK-OpenStack – offload tasks to underlying OpenStack infrastructure
32 VMs across 8 nodes, 6 Core/VM

EASJ - Compared to Native, less than 4% overhead with 128 Procs

SACJ, EACJ – Also minor overhead, when running NAS as concurrent job with 64 Procs
Approaches to Build HPC Clouds

- MVAPICH2-Virt with SR-IOV and IVSHMEM
  - Standalone, OpenStack
- MVAPICH2 with Containers
- MVAPICH2-Virt on SLURM
  - SLURM alone, SLURM + OpenStack
- Big Data Libraries on Cloud
The High-Performance Big Data (HiBD) Project

- RDMA for Apache Spark
- RDMA for Apache Hadoop 2.x (RDMA-Hadoop-2.x)
  - Plugins for Apache, Hortonworks (HDP) and Cloudera (CDH) Hadoop distributions
- RDMA for Apache HBase
- RDMA for Memcached (RDMA-Memcached)
- RDMA for Apache Hadoop 1.x (RDMA-Hadoop)
- OSU HiBD-Benchmarks (OHB)
  - HDFS, Memcached, and HBase Micro-benchmarks
- [http://hibd.cse.ohio-state.edu](http://hibd.cse.ohio-state.edu)
- Users Base: 200 organizations from 27 countries
- More than 18,900 downloads from the project site
- RDMA for Impala (upcoming)

Available for InfiniBand and RoCE
Big Data on Cloud Computing Systems: Challenges Addressed by OSU So Far

<table>
<thead>
<tr>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPC and Big Data Middleware</td>
</tr>
<tr>
<td>Big Data (HDFS, MapReduce, Spark, HBase, Memcached, etc.)</td>
</tr>
</tbody>
</table>

| Resource Management and Scheduling Systems for Cloud Computing (OpenStack Swift, Heat) |
| Communication and I/O Library |
| Communication Channels (SR-IOV) |
| Locality-aware Communication |
| Virtualization (Hypervisor) |
| Data Placement & Task Scheduling |
| Fault-Tolerance (Replication) |
| Future Studies |

| Networking Technologies (InfiniBand, Omni-Path, 1/10/40/100 GigE and Intelligent NICs) |
| Commodity Computing System Architectures (Multi- and Many-core architectures and accelerators) |
| Storage Technologies (HDD, SSD, NVRAM, and NVMe-SSD) |
High-Performance Apache Hadoop over Clouds: Challenges

- How about performance characteristics of native IB-based designs for Apache Hadoop over SR-IOV enabled cloud environment?
- To achieve locality-aware communication, how can the cluster topology be automatically detected in a scalable and efficient manner and be exposed to the Hadoop framework?
- How can we design virtualization-aware policies in Hadoop for efficiently taking advantage of the detected topology?
- Can the proposed policies improve the performance and fault tolerance of Hadoop on virtualized platforms?

“How can we design a high-performance Hadoop library for Cloud-based systems?”
Impact of HPC Cloud Networking Technologies

- Network architectures
  - IB QDR, FDR, EDR
  - 40GigE
  - 40G-RoCE

- Network protocols
  - TCP/IP, IPoIB
  - RC, UD, Others

- Cloud Technologies
  - Bare-metal, SR-IOV

Can existing designs of Hadoop components over InfiniBand need to be made “aware” of the underlying architectural trends and take advantage of the support for modern transport protocols that InfiniBand and RoCE provide?
### Design Features

- SEDA-based Thread Management
- Support RC, UD, and Hybrid transport protocols
- Architecture-aware designs for Eager, packetized, and zero-copy transfers
- JVM-bypassed buffer management
- Intelligent buffer allocation and adjustment for serialization
- InfiniBand/RoCE support for bare-metal and SR-IOV

---

**Overview of IB-based Hadoop-RPC and HBase Architecture**

- **Applications**
- **HBase**

- **Writable RPC Engine**
  - Java NIO-based RPC (Default)

- **Protobuf RPC Engine**
  - Native IB-based RPC (Plugin)

- **Java Socket Interface**
  - TCP/IP
  - 1/10/40/100 GigE
  - InfiniBand (QDR, FDR, EDR)

- **IPoIB (RC, UD)**

- **Java Native Interface (JNI)**

- **Native IB-/RoCE-based RPC Engine**
  - Multi-Channel Design (RC, UD, Hybrid)
  - RDMA Capable Networks (IB, RoCE ..)

---

Performance Benefits for Hadoop RPC and HBase

Hadoop RPC Throughput on Chameleon-Cloud

- Hadoop RPC Throughput on Chameleon-Cloud-FDR
  - up to 2.6x performance speedup over IPoIB for throughput

- HBase YCSB Workload A (read: write=50:50) on SDSC-Comet-FDR
  - Native designs (RC/UD/Hybrid) always perform better than the IPoIB-UD transport
  - up to 2.4x performance speedup over IPoIB for throughput
Overview of RDMA-Hadoop-Virt Architecture

• Virtualization-aware modules in all the four main Hadoop components:
  – **HDFS**: Virtualization-aware Block Management to improve fault-tolerance
  – **YARN**: Extensions to Container Allocation Policy to reduce network traffic
  – **MapReduce**: Extensions to Map Task Scheduling Policy to reduce network traffic
  – **Hadoop Common**: Topology Detection Module for automatic topology detection

• Communications in HDFS, MapReduce, and RPC go through RDMA-based designs over SR-IOV enabled InfiniBand

Evaluation with Applications

- 14% and 24% improvement with Default Mode for CloudBurst and Self-Join
- 30% and 55% improvement with Distributed Mode for CloudBurst and Self-Join
Available Appliances on Chameleon Cloud*

<table>
<thead>
<tr>
<th>Appliance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CentOS 7 KVM SR-IOV</td>
<td>Chameleon bare-metal image customized with the KVM hypervisor and a recompiled kernel to enable SR-IOV over InfiniBand.</td>
</tr>
<tr>
<td>MPI bare-metal cluster complex appliance (Based on Heat)</td>
<td>This appliance deploys an MPI cluster composed of bare metal instances using the MVAPICH2 library over InfiniBand.</td>
</tr>
<tr>
<td>MPI + SR-IOV KVM cluster (Based on Heat)</td>
<td>This appliance deploys an MPI cluster of KVM virtual machines using the MVAPICH2-Virt implementation and configured with SR-IOV for high-performance communication over InfiniBand.</td>
</tr>
</tbody>
</table>

- Through these available appliances, users and researchers can easily deploy HPC clouds to perform experiments and run jobs with:
  - High-Performance SR-IOV + InfiniBand
  - High-Performance MVAPICH2 Library over bare-metal InfiniBand clusters
  - High-Performance MVAPICH2 Library with Virtualization Support over SR-IOV enabled KVM clusters
  - High-Performance Hadoop with RDMA-based Enhancements Support

[*] Only include appliances contributed by OSU NowLab
MPI Complex Appliances based on MVAPICH2 on Chameleon

1. Load VM Config
2. Allocate Ports
3. Allocate FloatingIPs
4. Generate SSH Keypair
5. Launch VM
6. Attach SR-IOV Device
7. Hot plug IVShmem Device
8. Download/Install MVAPICH2-Virt
9. Populate VMs/IPs
10. Associate FloatingIPs

Cloud

VM
MPI
VM
MPI
VM
MPI
VM
MPI

... Stack ...

Cloud Com 2016

Network Based Computing Laboratory

OpenStack

HEAT
HORIZON
NOVA
NEUTRON
GLANCE
CIDER
SWIFT
......
Conclusions

- Outlined challenges and opportunities in running MPI and BigData applications in Cloud with performance
- MVAPICH2-Virt with SR-IOV and IVSHMEM is an efficient approach to build HPC Clouds
  - Standalone
  - OpenStack
- Building HPC Clouds with MVAPICH2-Virt on SLURM is possible
  - SLURM alone
  - SLURM + OpenStack
- Containers-based design for MPAPICH2-Virt
- Very little overhead with virtualization, near native performance at application level
- **MVAPICH2-Virt 2.1** is available for building HPC Clouds
  - SR-IOV, IVSHMEM, Docker support, OpenStack
- Appliances for MVAPICH2-Virt and RDMA-Hadoop are available for building HPC Clouds
- Future releases for supporting running MPI jobs in VMs/Containers with SLURM, VM migration, etc.
- SR-IOV/container support and appliances for other MVAPICH2 libraries (MVAPICH2-X, MVAPICH2-GDR, ...) and RDMA-Spark/Memcached
Funding Acknowledgments

Funding Support by

Equipment Support by
Personnel Acknowledgments

Current Students
- A. Awan (Ph.D.)
- M. Bayatpour (Ph.D.)
- S. Chakraborty (Ph.D.)
- C.-H. Chu (Ph.D.)
- S. Guganani (Ph.D.)
- M. Rahman (Ph.D.)
- J. Hashmi (Ph.D.)
- D. Shankar (Ph.D.)
- N. Islam (Ph.D.)
- A. Venkatesh (Ph.D.)
- M. Li (Ph.D.)
- J. Zhang (Ph.D.)

Past Students
- A. Augustine (M.S.)
- P. Balaji (Ph.D.)
- S. Bhagvat (M.S.)
- A. Bhat (M.S.)
- D. Buntinas (Ph.D.)
- L. Chai (Ph.D.)
- B. Chandrasekharan (M.S.)
- N. Dandapanthula (M.S.)
- V. Dhanraj (M.S.)
- T. Gangadharappa (M.S.)
- K. Gopalakrishnan (M.S.)
- W. Huang (Ph.D.)
- J. Jose (Ph.D.)
- S. Kini (M.S.)
- M. Koop (Ph.D.)
- K. Kulkarni (M.S.)
- R. Kumar (M.S.)
- S. Krishnamoorthy (M.S.)
- P. Lai (M.S.)
- J. Liu (Ph.D.)
- M. Luo (Ph.D.)
- A. Mamidala (Ph.D.)
- G. Marsh (M.S.)
- V. Meshram (M.S.)
- A. Moody (M.S.)
- S. Naravula (Ph.D.)
- R. Noronha (Ph.D.)
- X. Ouyang (Ph.D.)
- S. Pai (M.S.)
- S. Potluri (Ph.D.)
- R. Rajachandrasekar (Ph.D.)
- G. Santhanaraman (Ph.D.)
- A. Singh (Ph.D.)
- J. Sridhar (M.S.)
- S. Sur (Ph.D.)
- H. Subramoni (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- A. Vishnu (Ph.D.)
- W. Yu (Ph.D.)

Past Post-Docs
- D. Banerjee
- X. Besseron
- H.-W. Jin
- J. Lin
- M. Luo
- E. Mancini
- S. Marcarelli
- J. Vienne
- H. Wang

Current Research Scientists
- K. Hamidouche
- X. Lu
- H. Subramoni

Current Research Specialist
- J. Smith

Past Research Scientists
- S. Sur

Past Programmers
- D. Bureddy
- M. Arnold
- J. Perkins

Network Based Computing Laboratory
CloudCom 2016
Thank You!

panda@cse.ohio-state.edu

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

The MVAPICH2/MVAPICH2-X Project
http://mvapich.cse.ohio-state.edu/

The High-Performance Big Data Project
http://hibd.cse.ohio-state.edu/