



# MVAPICH

MPI, PGAS and Hybrid MPI+PGAS Library

## MVAPICH2-GDR: Pushing the Frontier of HPC and Deep Learning

Talk at NVIDIA booth (SC 2017)

by

**Dhabaleswar K. (DK) Panda**

The Ohio State University

E-mail: [panda@cse.ohio-state.edu](mailto:panda@cse.ohio-state.edu)

<http://www.cse.ohio-state.edu/~panda>

# Outline

- Overview of the MVAPICH2 Project
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
- What's new with MVAPICH2-GDR
  - Multi-stream Communication for IPC
  - CMA-based Intra-node Communication Support
  - Maximal overlap in MPI Datatype Processing
  - Initial support for GPUDirect Async feature
- Streaming Support with IB Multicast and GDR
- High-Performance Deep Learning (HiDL) with MVAPICH2-GDR
- Conclusions

# 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 2,825 organizations in 85 countries**
  - **More than 433,000 (> 0.4 million) downloads from the OSU site directly**
  - Empowering many TOP500 clusters (June '17 ranking)
    - **1st, 10,649,600-core (Sunway TaihuLight) at National Supercomputing Center in Wuxi, China**
    - 15th, 241,108-core (Pleiades) at NASA
    - 20th, 462,462-core (Stampede) at TACC
    - 44th, 74,520-core (Tsubame 2.5) at Tokyo Institute of Technology
  - Available with software stacks of many vendors and Linux Distros (RedHat and SuSE)
  - <http://mvapich.cse.ohio-state.edu>
- Empowering Top500 systems for over a decade
  - System-X from Virginia Tech (3<sup>rd</sup> in Nov 2003, 2,200 processors, 12.25 TFlops) ->
  - Sunway TaihuLight (1<sup>st</sup> in Jun'17, 10M cores, 100 PFlops)



# MVAPICH2 Architecture

## High Performance Parallel Programming Models

**Message Passing Interface  
(MPI)**

**PGAS  
(UPC, OpenSHMEM, CAF, UPC++)**

**Hybrid --- MPI + X  
(MPI + PGAS + OpenMP/Cilk)**

## High Performance and Scalable Communication Runtime

### Diverse APIs and Mechanisms

Point-to-point  
Primitives

Collectives  
Algorithms

Job Startup

Energy-  
Awareness

Remote  
Memory  
Access

I/O and  
File Systems

Fault  
Tolerance

Virtualization

Active  
Messages

Introspection  
& Analysis

### 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\*

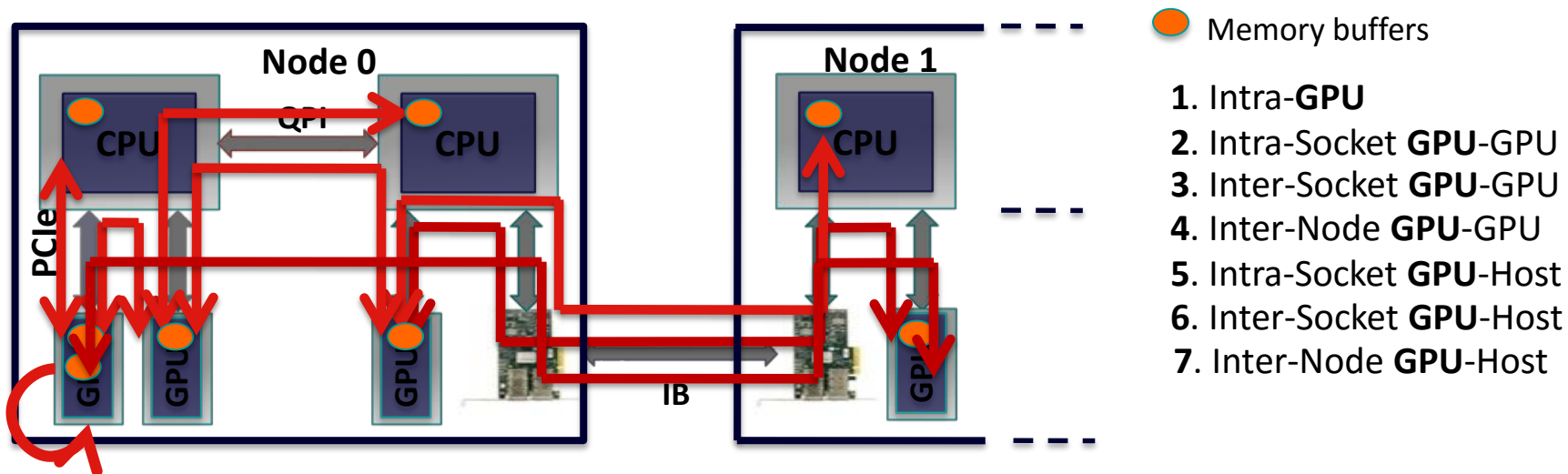
\* Upcoming

# MVAPICH2 Software Family

High-Performance Parallel Programming Libraries	
MVAPICH2	Support for InfiniBand, Omni-Path, Ethernet/iWARP, and RoCE
MVAPICH2-X	Advanced MPI features, OSU INAM, PGAS (OpenSHMEM, UPC, UPC++, and CAF), and MPI+PGAS programming models with unified communication runtime
MVAPICH2-GDR	Optimized MPI for clusters with NVIDIA GPUs
MVAPICH2-Virt	High-performance and scalable MPI for hypervisor and container based HPC cloud
MVAPICH2-EA	Energy aware and High-performance MPI
MVAPICH2-MIC	Optimized MPI for clusters with Intel KNC
Microbenchmarks	
OMB	Microbenchmarks suite to evaluate MPI and PGAS (OpenSHMEM, UPC, and UPC++) libraries for CPUs and GPUs
Tools	
OSU INAM	Network monitoring, profiling, and analysis for clusters with MPI and scheduler integration
OEMT	Utility to measure the energy consumption of MPI applications

# MVAPICH2-GDR: Optimizing MPI Data Movement on GPU Clusters

- Connected as PCIe devices – Flexibility but Complexity



8. Inter-Node GPU-GPU with IB adapter on remote socket  
and more . . .

- For each path different schemes: Shared\_mem, IPC, GPUDirect RDMA, pipeline ...
- Critical for runtimes to optimize data movement while hiding the complexity

# GPU-Aware (CUDA-Aware) MPI Library: MVAPICH2-GPU

- Standard MPI interfaces used for unified data movement
- Takes advantage of Unified Virtual Addressing ( $\geq$  CUDA 4.0)
- Overlaps data movement from GPU with RDMA transfers

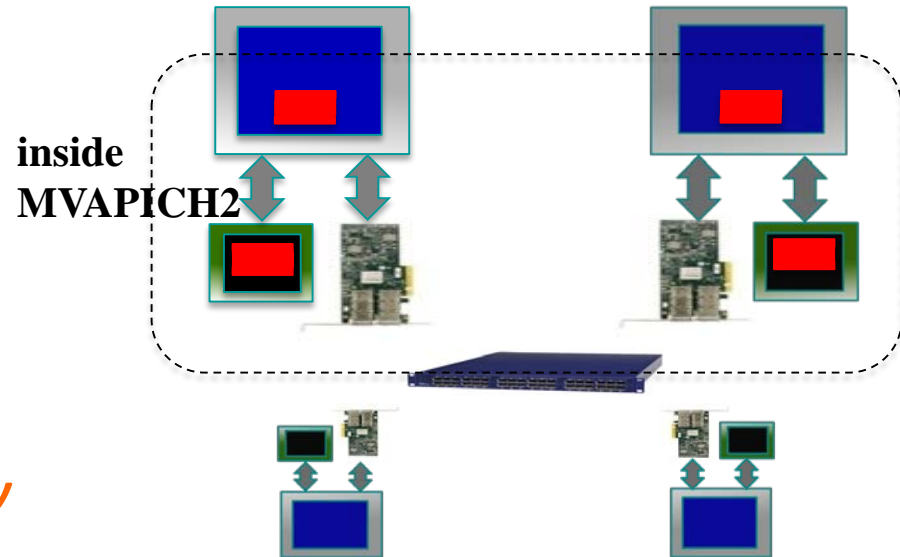
## At Sender:

```
MPI_Send(s_devbuf, size, ...);
```

## At Receiver:

```
MPI_Recv(r_devbuf, size, ...);
```

*High Performance and High Productivity*



## CUDA-Aware MPI: MVAPICH2-GDR 1.8-2.3 Releases

- Support for MPI communication from NVIDIA GPU device memory
- High performance RDMA-based inter-node point-to-point communication (GPU-GPU, GPU-Host and Host-GPU)
- High performance intra-node point-to-point communication for multi-GPU adapters/node (GPU-GPU, GPU-Host and Host-GPU)
- Taking advantage of CUDA IPC (available since CUDA 4.1) in intra-node communication for multiple GPU adapters/node
- Optimized and tuned collectives for GPU device buffers
- MPI datatype support for point-to-point and collective communication from GPU device buffers
- Unified memory



## Using MVAPICH2-GPUDirect Version

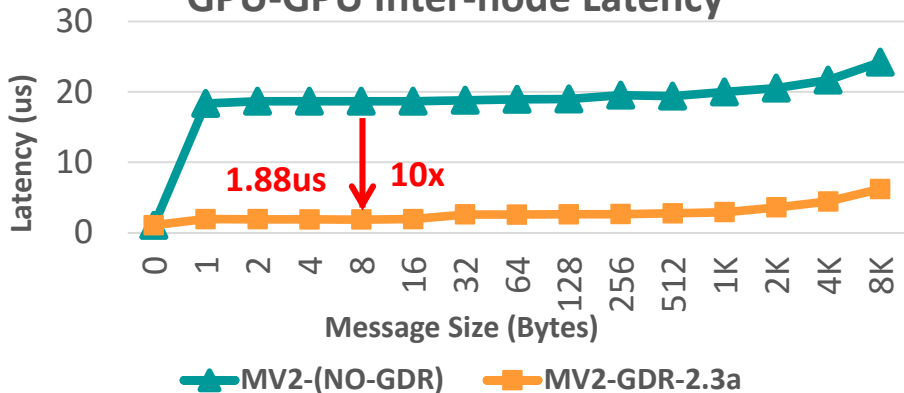
- MVAPICH2-2.3 with GDR support can be downloaded from <https://mvapich.cse.ohio-state.edu/download/mvapich2gdr/>
- System software requirements
  - Mellanox OFED 3.2 or later
  - NVIDIA Driver 367.48 or later
  - NVIDIA CUDA Toolkit 7.5/8.0/9.0 or later
  - Plugin for GPUDirect RDMA  
[http://www.mellanox.com/page/products\\_dyn?product\\_family=116](http://www.mellanox.com/page/products_dyn?product_family=116)
  - Strongly recommended
  - GDRCOPY module from NVIDIA  
<https://github.com/NVIDIA/gdrcopy>
- Contact MVAPICH help list with any questions related to the package  
[mvapich-help@cse.ohio-state.edu](mailto:mvapich-help@cse.ohio-state.edu)

# MVAPICH2-GDR 2.3a

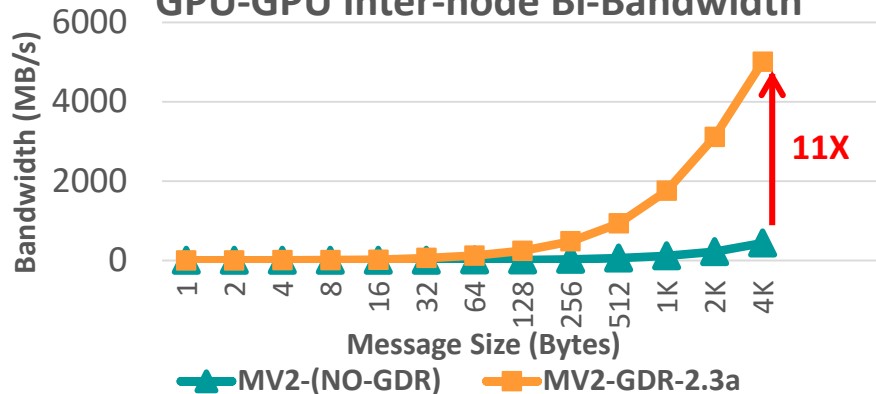
- Released on 11/09/2017
- Major Features and Enhancements
  - **Based on MVAPICH2 2.2**
  - **Support for CUDA 9.0**
  - **Add support for Volta (V100) GPU**
  - **Support for OpenPOWER with NVLink**
  - **Efficient Multiple CUDA stream-based IPC communication for multi-GPU systems with and without NVLink**
  - **Enhanced performance of GPU-based point-to-point communication**
  - **Leverage Linux Cross Memory Attach (CMA) feature for enhanced host-based communication**
  - **Enhanced performance of MPI\_Allreduce for GPU-resident data**
  - **InfiniBand Multicast (IB-MCAST) based designs for GPU-based broadcast and streaming applications**
    - **Basic support for IB-MCAST designs with GPUDirect RDMA**
    - **Advanced support for zero-copy IB-MCAST designs with GPUDirect RDMA**
    - **Advanced reliability support for IB-MCAST designs**
  - **Efficient broadcast designs for Deep Learning applications**
  - **Enhanced collective tuning on Xeon, OpenPOWER, and NVIDIA DGX-1 systems**

# Optimized MVAPICH2-GDR Design

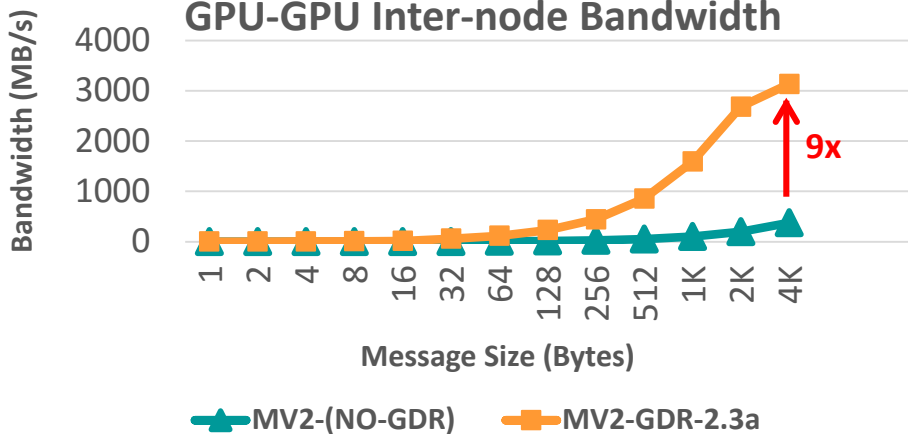
## GPU-GPU Inter-node Latency



## GPU-GPU Inter-node Bi-Bandwidth



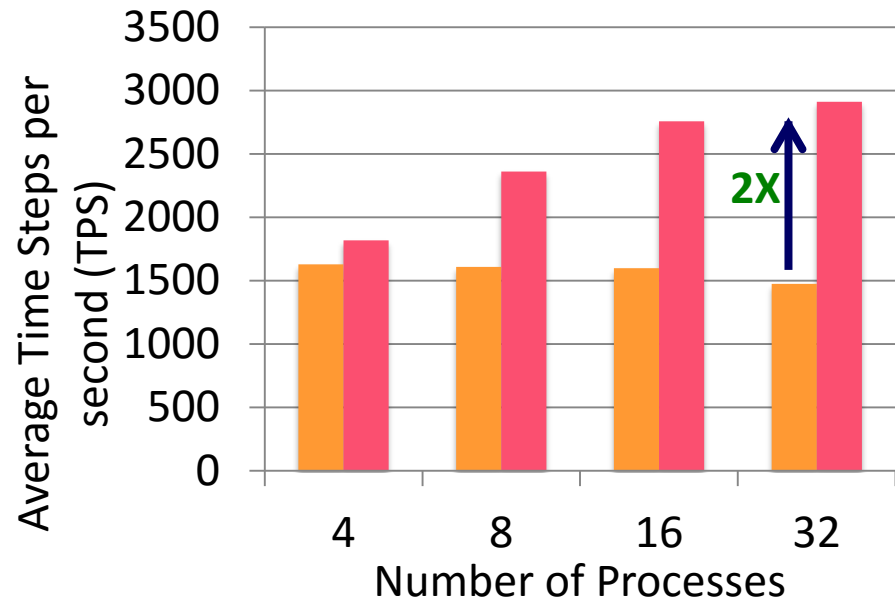
## GPU-GPU Inter-node Bandwidth



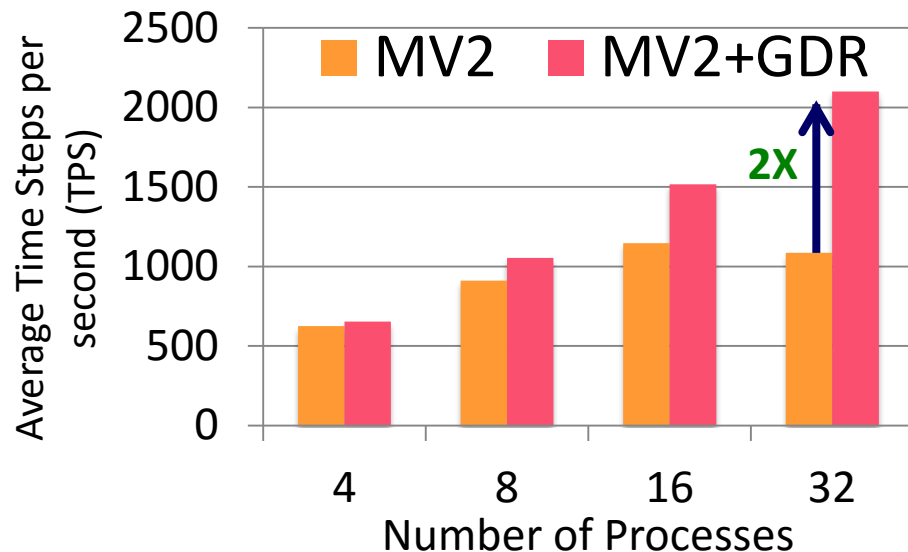
**MVAPICH2-GDR-2.3a**  
**Intel Haswell (E5-2687W @ 3.10 GHz) node - 20 cores**  
**NVIDIA Volta V100 GPU**  
**Mellanox Connect-X4 EDR HCA**  
**CUDA 9.0**  
**Mellanox OFED 4.0 with GPU-Direct-RDMA**

# Application-Level Evaluation (HOOMD-blue)

## 64K Particles



## 256K Particles



- Platform: Wilkes (Intel Ivy Bridge + NVIDIA Tesla K20c + Mellanox Connect-IB)
- **HoomdBlue Version 1.0.5**
  - GDRCOPY enabled: MV2\_USE\_CUDA=1 MV2\_IBA\_HCA=mlx5\_0 MV2\_IBA\_EAGER\_THRESHOLD=32768 MV2\_VBUF\_TOTAL\_SIZE=32768 MV2\_USE\_GPUDIRECT\_LOOPBACK\_LIMIT=32768 MV2\_USE\_GPUDIRECT\_GDRCOPY=1 MV2\_USE\_GPUDIRECT\_GDRCOPY\_LIMIT=16384

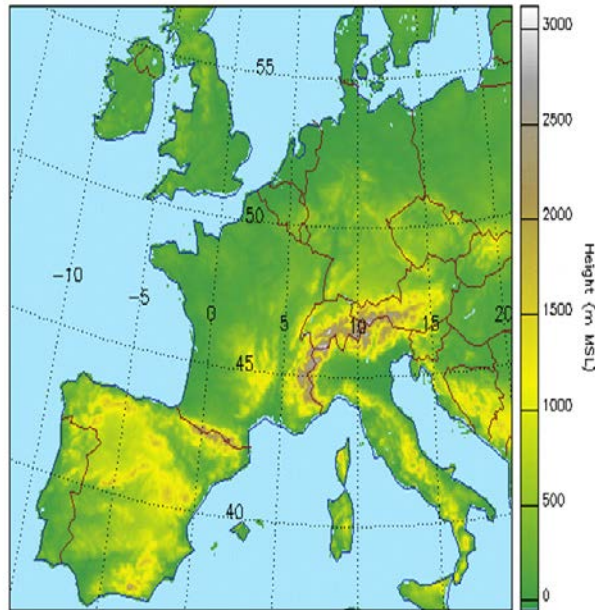
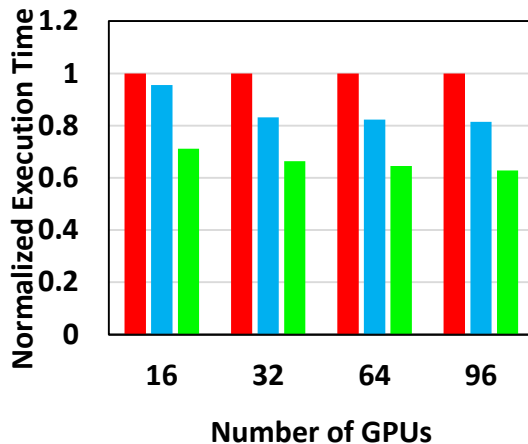
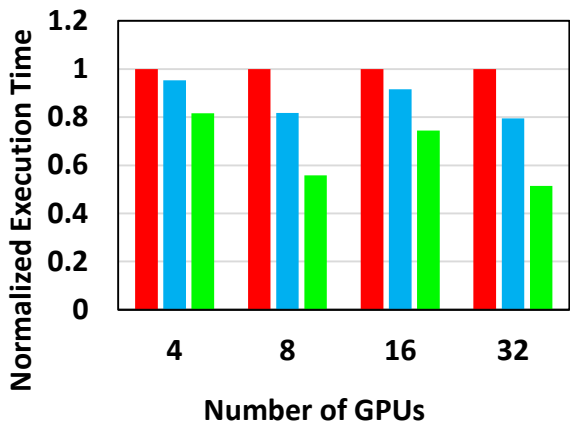
# Application-Level Evaluation (Cosmo) and Weather Forecasting in Switzerland

## Wilkes GPU Cluster

## CSCS GPU cluster

■ Default ■ Callback-based ■ Event-based

■ Default ■ Callback-based ■ Event-based



- 2X improvement on 32 GPUs nodes
- 30% improvement on 96 GPU nodes (8 GPUs/node)

Cosmo model: <http://www2.cosmo-model.org/content/tasks/operational/meteoSwiss/>

**On-going collaboration with CSCS and MeteoSwiss (Switzerland) in co-designing MV2-GDR and Cosmo Application**

C. Chu, K. Hamidouche, A. Venkatesh, D. Banerjee, H. Subramoni, and D. K. Panda, Exploiting Maximal Overlap for Non-Contiguous Data Movement Processing on Modern GPU-enabled Systems, IPDPS'16

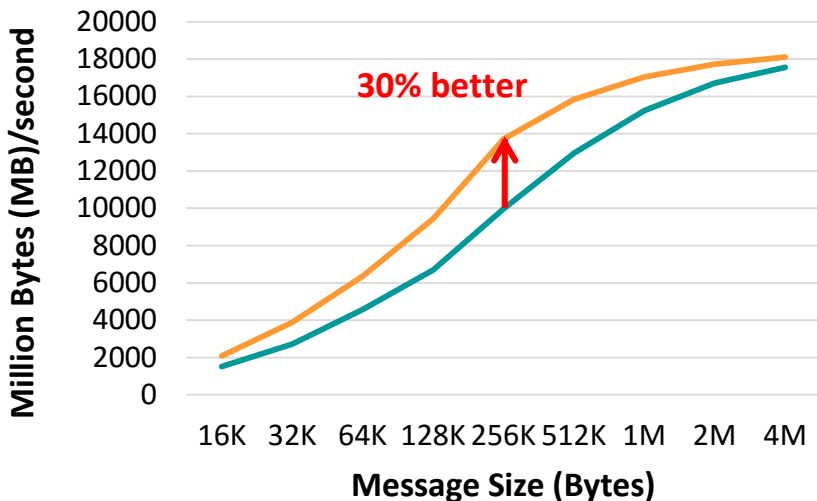
# Outline

- Overview of the MVAPICH2 Project
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
- **What's new with MVAPICH2-GDR**
  - Multi-stream Communication for IPC
  - CMA-based Intra-node Communication Support
  - Support for OpenPower and NVLink
  - Initial support for GPUDirect Async feature
- Streaming Support with IB Multicast and GDR
- High-Performance Deep Learning (HiDL) with MVAPICH2-GDR
- Conclusions

# Multi-stream Communication using CUDA IPC on OpenPOWER and DGX-1

- Up to **16% higher** Device to Device (D2D) bandwidth on OpenPOWER + NVLink inter-connect
- Up to **30% higher** D2D bandwidth on DGX-1 with NVLink
- 

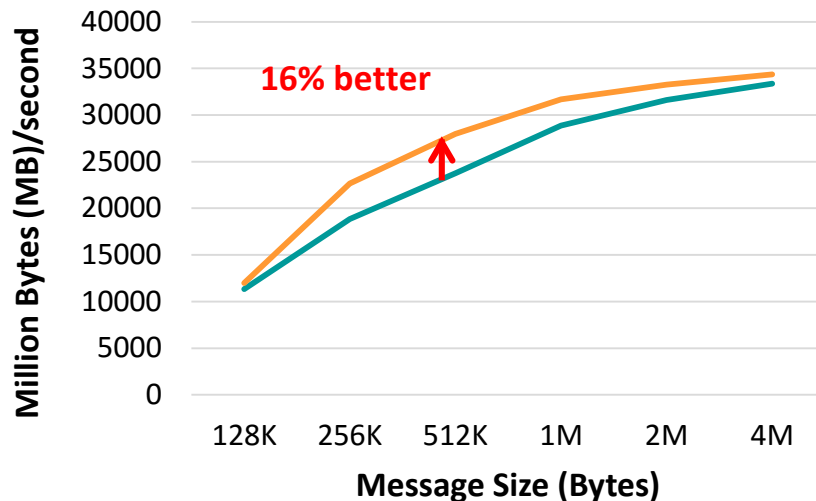
Pt-to-pt (D-D) Bandwidth:  
Benefits of Multi-stream CUDA IPC Design



— 1-stream — 4-streams

Available with **MVAPICH2-GDR-2.3a**

Pt-to-pt (D-D) Bandwidth:  
Benefits of Multi-stream CUDA IPC Design

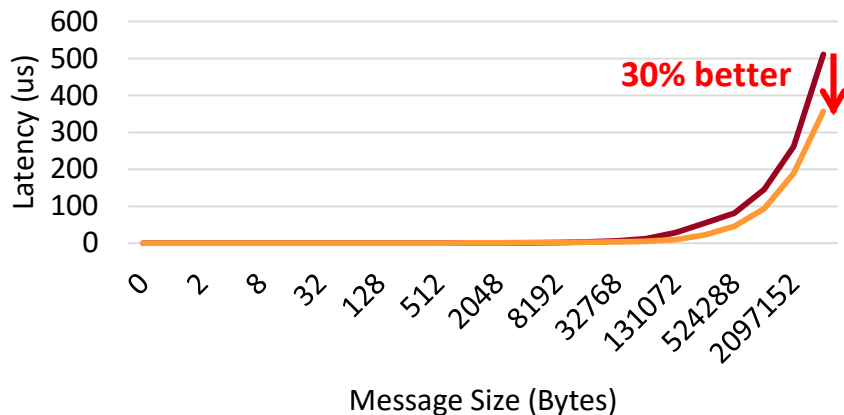


— 1-stream — 4-streams

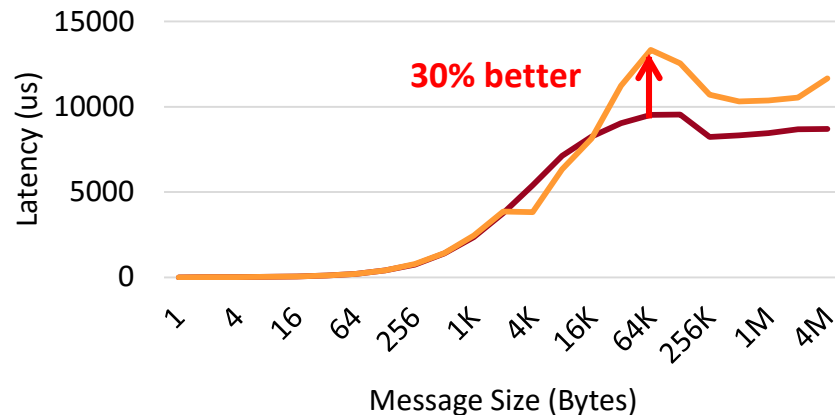
# CMA-based Intra-node Communication Support

- Up to **30% lower** Host-to-Host (H2H) latency and **30% higher** H2H Bandwidth

### INTRA-NODE Pt-to-Pt (H2H) LATENCY



### INTRA-NODE Pt-to-Pt (H2H) BANDWIDTH



— MV2-GDR (w/out CMA)

— MV2-GDR (w/ CMA)

— MV2-GDR (w/out CMA)

— MV2-GDR (w/ CMA)

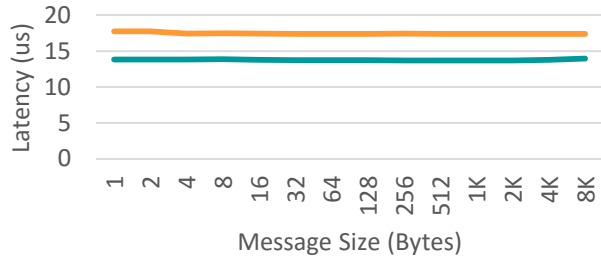
**MVAPICH2-GDR-2.3a**

**Intel Broadwell (E5-2680 v4 @ 3240 GHz) node – 28 cores  
NVIDIA Tesla K-80 GPU, and Mellanox Connect-X4 EDR HCA  
CUDA 8.0, Mellanox OFED 4.0 with GPU-Direct-RDMA**



# MVAPICH2-GDR: Performance on OpenPOWER (NVLink + Pascal)

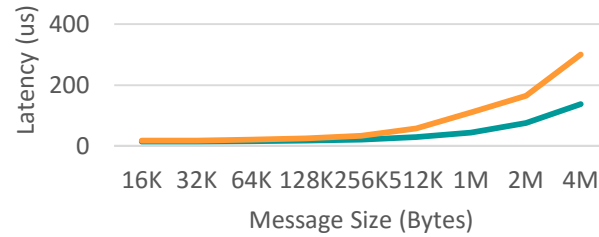
## INTRA-NODE LATENCY (SMALL)



— INTRA-SOCKET(NVLINK) — INTER-SOCKET

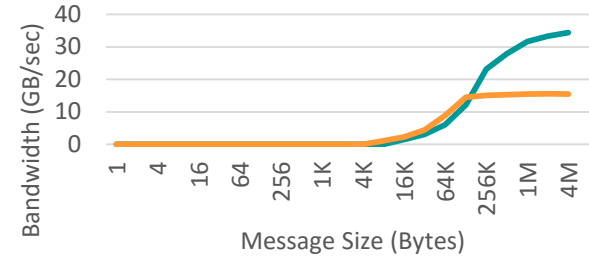
**Intra-node Latency: 13.8 us (without GPUDirectRDMA)**

## INTRA-NODE LATENCY (LARGE)



— INTRA-SOCKET(NVLINK) — INTER-SOCKET

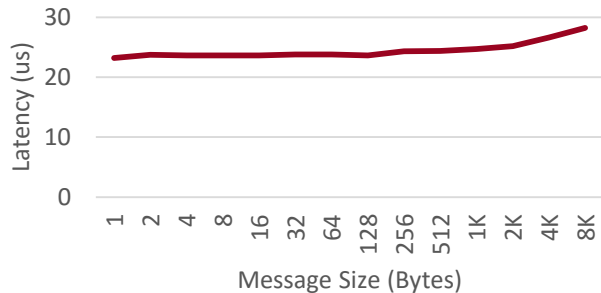
## INTRA-NODE BANDWIDTH



— INTRA-SOCKET(NVLINK) — INTER-SOCKET

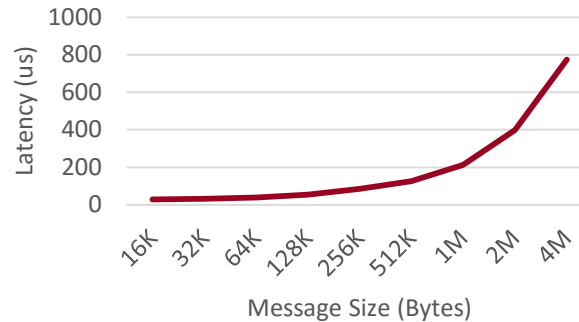
**Intra-node Bandwidth: 33.2 GB/sec (NVLINK)**

## INTER-NODE LATENCY (SMALL)



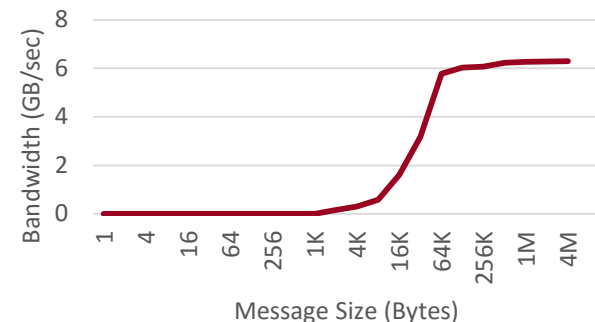
**Inter-node Latency: 23 us (without GPUDirectRDMA)**

## INTER-NODE LATENCY (LARGE)



**Available in MVAPICH2-GDR 2.3a**

## INTER-NODE BANDWIDTH



**Inter-node Bandwidth: 6 GB/sec (FDR)**

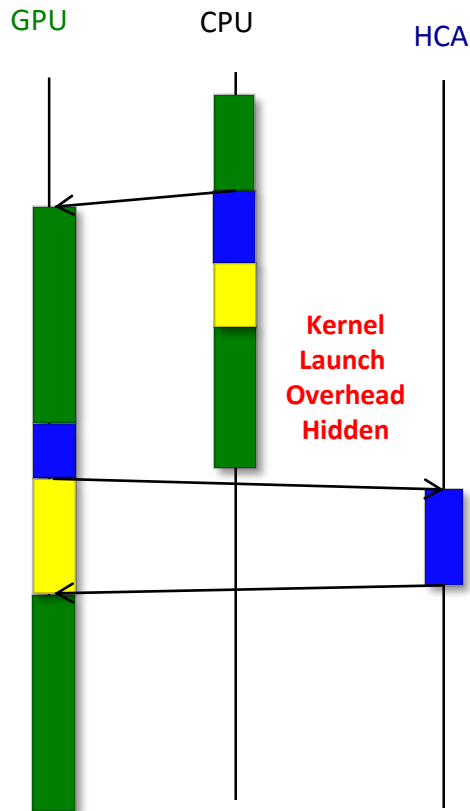
*Platform: OpenPOWER (ppc64le) nodes equipped with a dual-socket CPU, 4 Pascal P100-SXM GPUs, and 4X-FDR InfiniBand Inter-connect*

# Control Flow Decoupling through GPUDirect Async

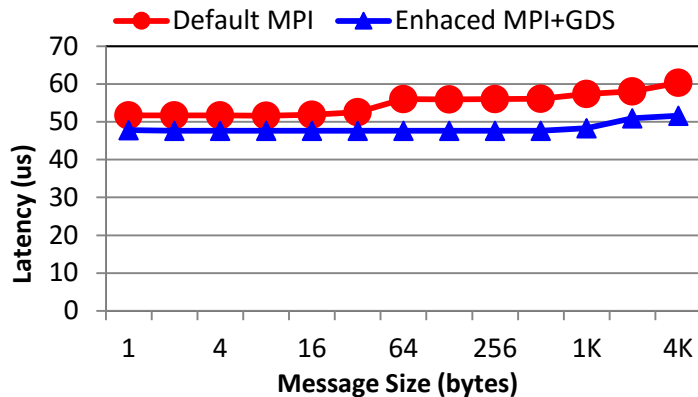
```

CUDA_Kernel_a<<<>>(A..., stream1)
MPI_Isend (A, ..., req1, stream1)
MPI_Wait (req1, stream1) (non-blocking from CPU)
CUDA_Kernel_b<<<>>(B..., stream1)
    
```

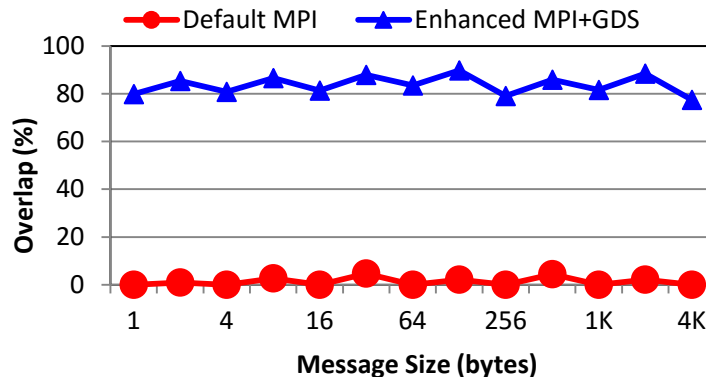
- CPU offloads the compute, communication and synchronization tasks to GPU
  - All operations **asynchronous from CPU**
    - **Hide the overhead of kernel launch**
- Needs **stream-based extensions** to MPI semantics
- Latency Oriented: Able to hide the kernel launch overhead - **25% improvement** at 256 Bytes
- Throughput Oriented: Asynchronously to offload queue the Communication and computation tasks - **14% improvement** at 1KB message size
- Intel Sandy Bridge, NVIDIA K20 and Mellanox FDR HCA
- Will be available in a public release soon



Latency oriented: Kernel+Send and Recv+Kernel



Overlap with host computation/communication

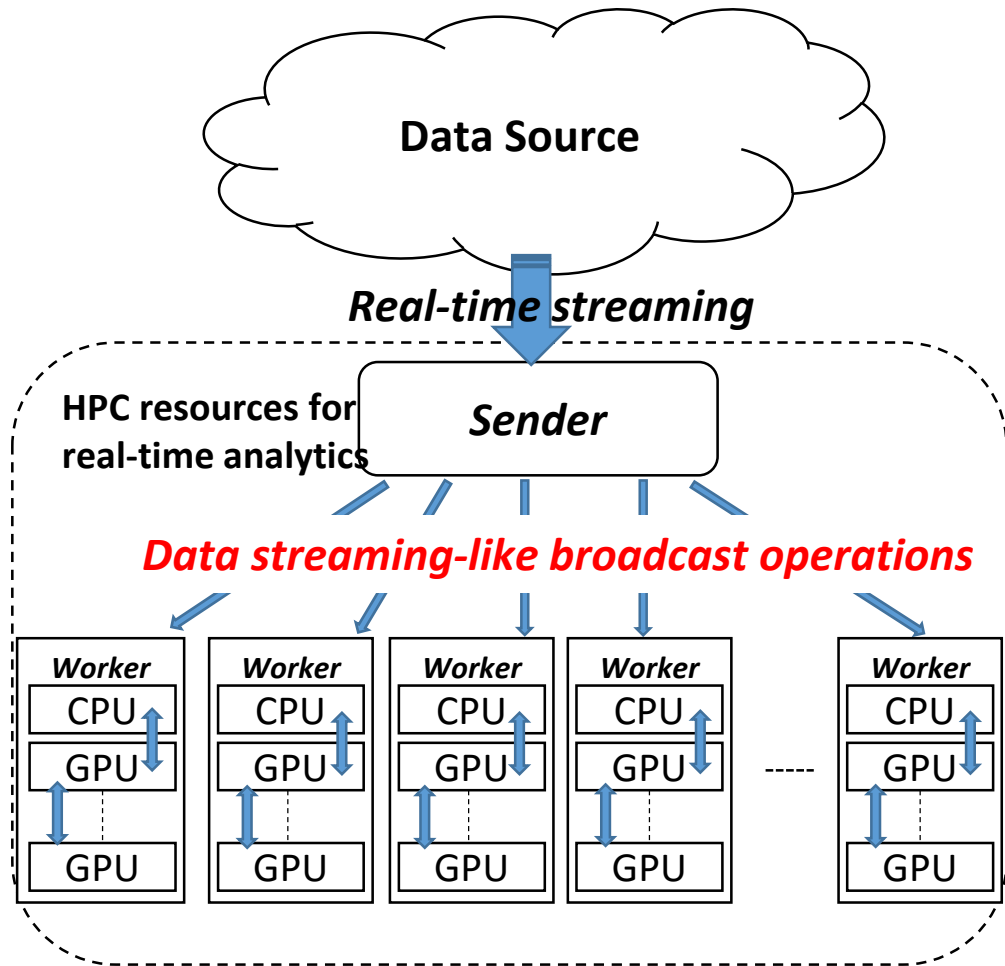


# Outline

- Overview of the MVAPICH2 Project
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
- What's new with MVAPICH2-GDR
  - Multi-stream Communication for IPC
  - CMA-based Intra-node Communication Support
  - Support for OpenPower and NVLink
  - Initial support for GPUDirect Async feature
- **Streaming Support with IB Multicast and GDR**
- High-Performance Deep Learning (HiDL) with MVAPICH2-GDR
- Conclusions

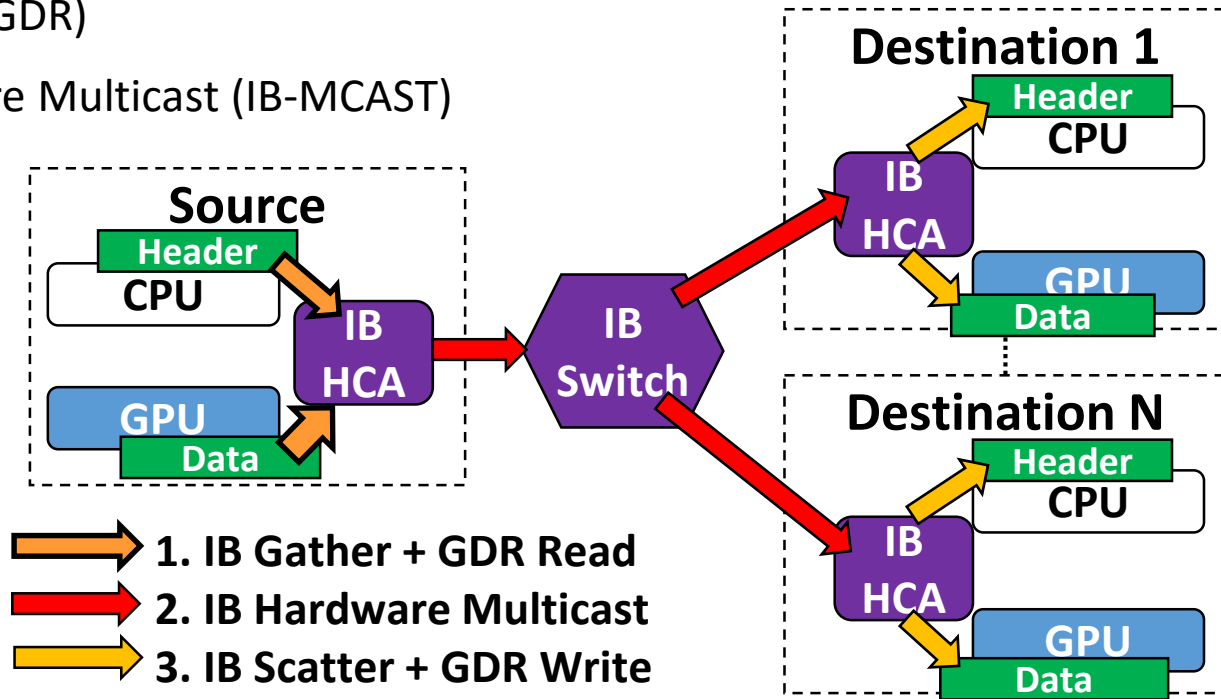
# Streaming Applications

- Streaming applications on HPC systems
  1. Communication (**MPI**)
    - Broadcast-type operations
  2. Computation (**CUDA**)
    - Multiple GPU nodes as workers



# Hardware Multicast-based Broadcast

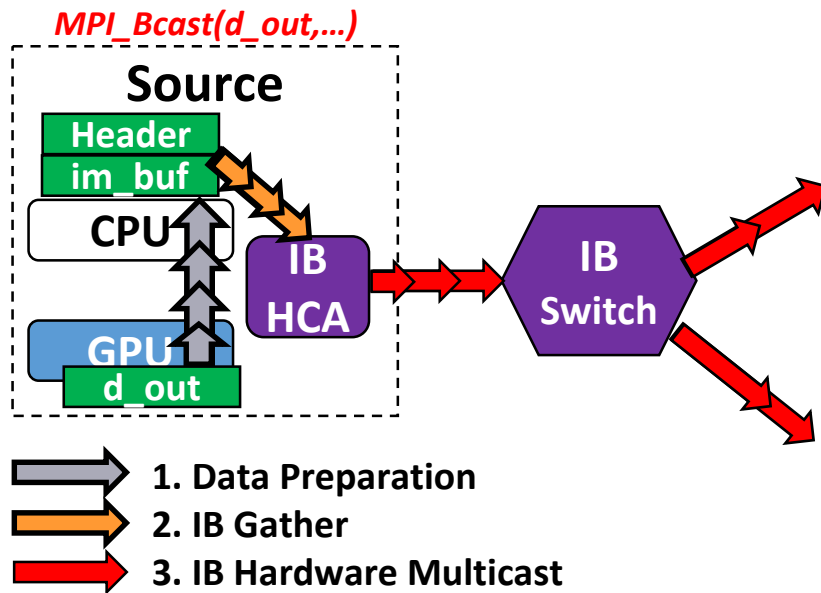
- For GPU-resident data, using
  - GPUDirect RDMA (GDR)
  - InfiniBand Hardware Multicast (IB-MCAST)
- Overhead
  - IB UD limit
  - GDR limit



A. Venkatesh, H. Subramoni, K. Hamidouche, and D. K. Panda, "A High Performance Broadcast Design with Hardware Multicast and GPUDirect RDMA for Streaming Applications on InfiniBand Clusters," in *HiPC 2014*, Dec 2014.

# Optimized Broadcast Send

- Preparing Intermediate buffer (*im\_buf*)
  - Page-locked (pinned) host buffer
    - Fast Device-Host data movement
  - Allocated at initialization phase
    - Low overhead
- Streaming data through host
  - Fine-tuned chunked data
  - Asynchronous copy operations
    - Three-stage pipeline

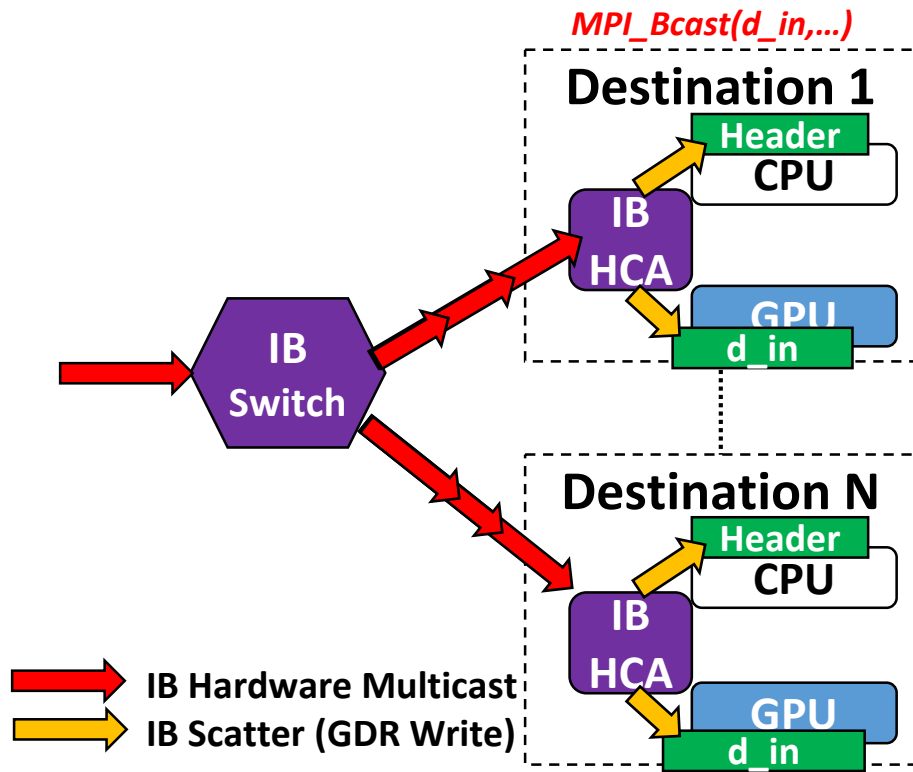


C.-H. Chu, X. Lu, A. A. Awan, H. Subramoni, J. Hashmi, B. Elton and D. K. Panda., "Efficient and Scalable Multi-Source Streaming Broadcast on GPU Clusters for Deep Learning," ICPP 2017, Aug 14-17, 2017.

# Optimized Broadcast Receive

- Zero-copy broadcast receive
  - Pre-posted user buffer ( $d_{in}$ )
  - Avoids additional data movement
  - Leverages IB Scatter and GDR features
  - Low-latency
  - Free-up PCIe resources for applications

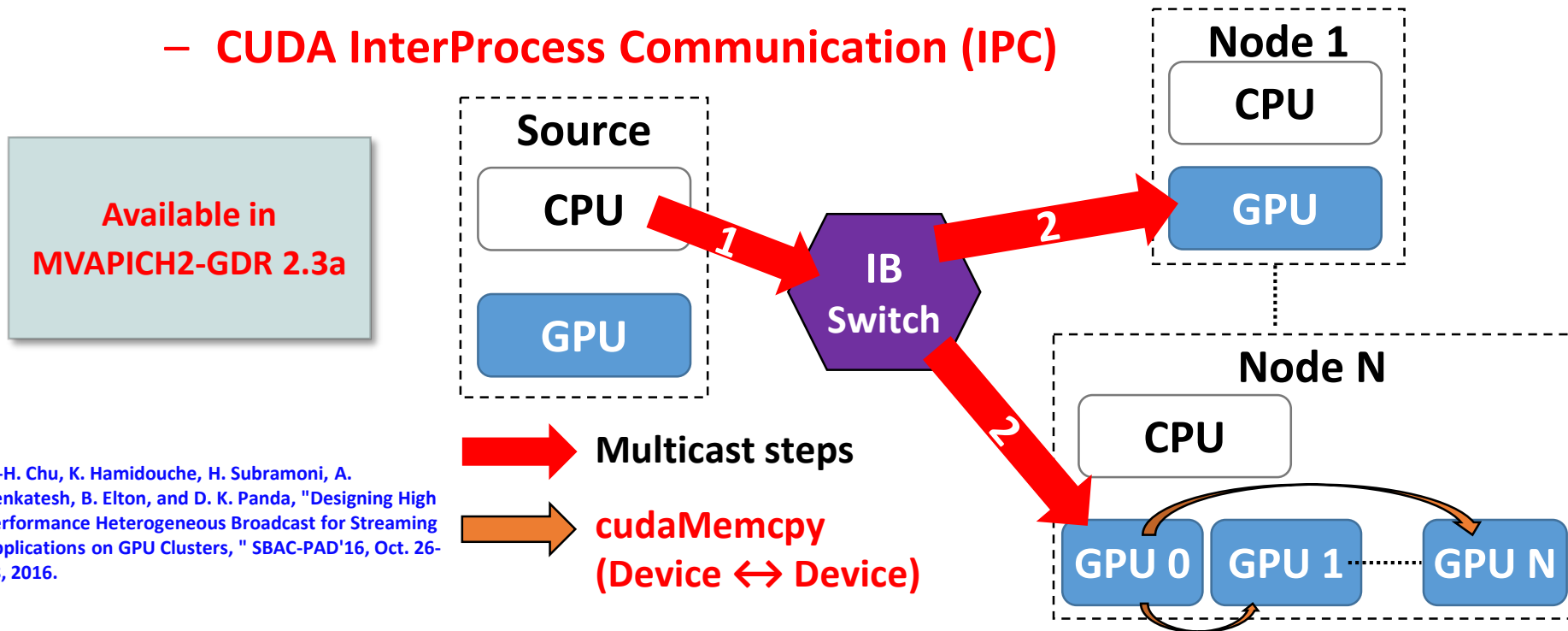
C.-H. Chu, X. Lu, A. A. Awan, H. Subramoni, J. Hashmi, B. Elton and D. K. Panda., "Efficient and Scalable Multi-Source Streaming Broadcast on GPU Clusters for Deep Learning," ICPP 2017, Aug 14-17, 2017.



# Broadcast on Multi-GPU systems

- Proposed Intra-node Topology-Aware Broadcast

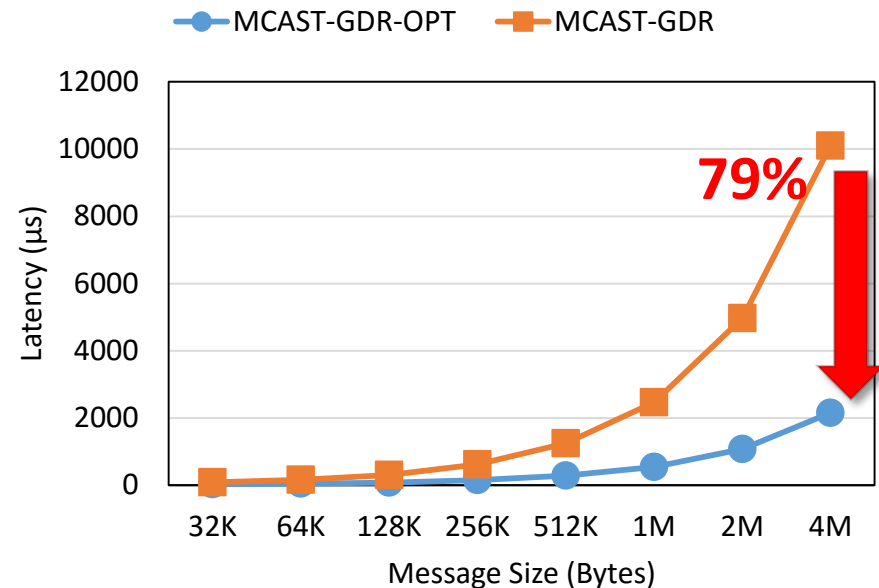
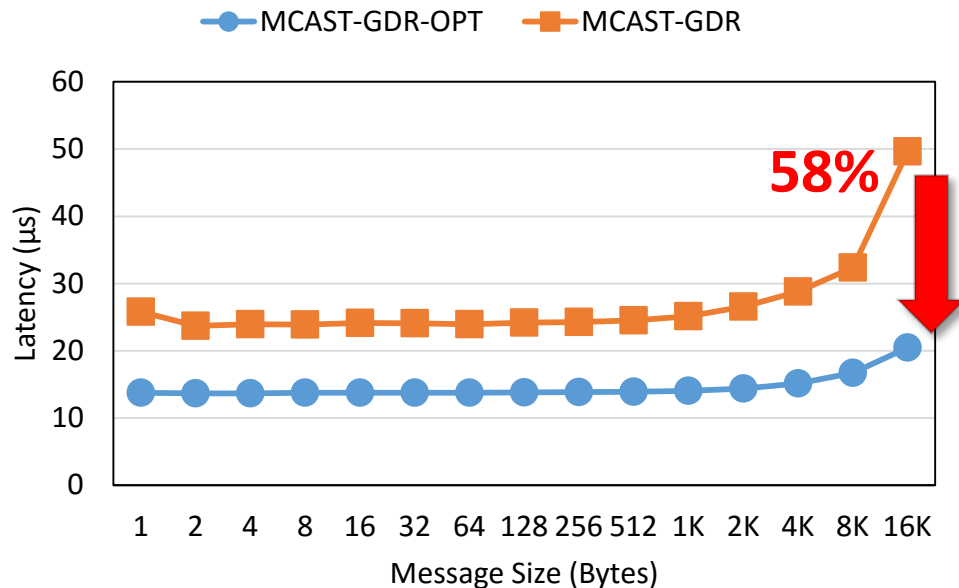
- CUDA InterProcess Communication (IPC)



C.-H. Chu, K. Hamidouche, H. Subramoni, A. Venkatesh, B. Elton, and D. K. Panda, "Designing High Performance Heterogeneous Broadcast for Streaming Applications on GPU Clusters," SBAC-PAD'16, Oct. 26-28, 2016.



# Streaming Benchmark @ CSCS (88 GPUs)



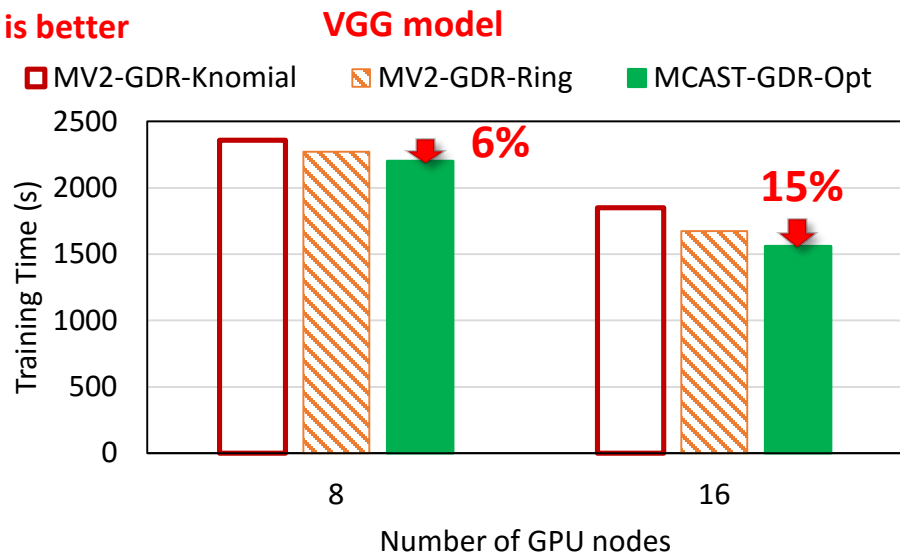
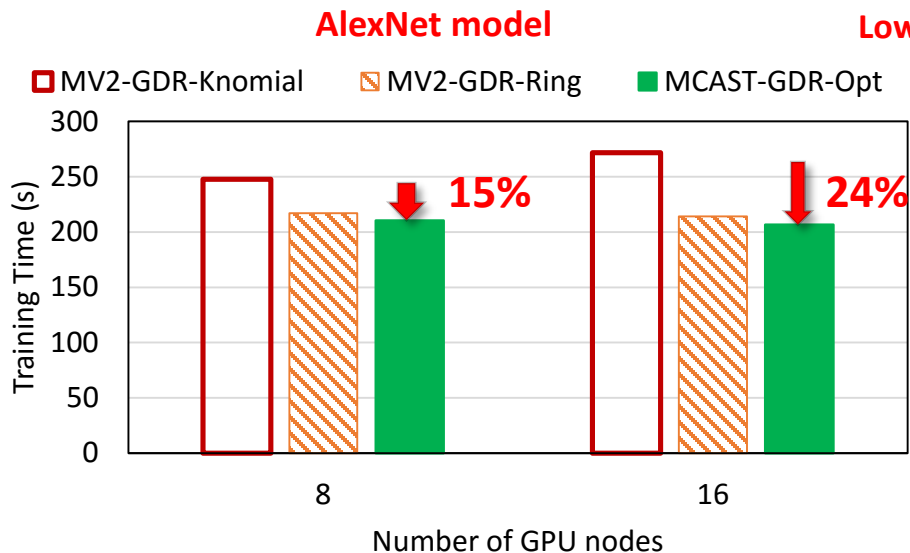
- **IB-MCAST + GDR + Topology-aware IPC-based schemes**

- Up to **58% and 79% reduction** for small and large messages

C.-H. Chu, K. Hamidouche, H. Subramoni, A. Venkatesh, B. Elton, and D. K. Panda, "Designing High Performance Heterogeneous Broadcast for Streaming Applications on GPU Clusters," SBAC-PAD'16, Oct. 26-28, 2016.

# Application Evaluation: Deep Learning Frameworks

- @ RI2 cluster, 16 GPUs, 1 GPU/node
  - Microsoft Cognitive Toolkit (CNTK) [<https://github.com/Microsoft/CNTK>]



- Reduces up to 24% and 15% of latency for AlexNet and VGG models
- Higher improvement can be observed for larger system sizes

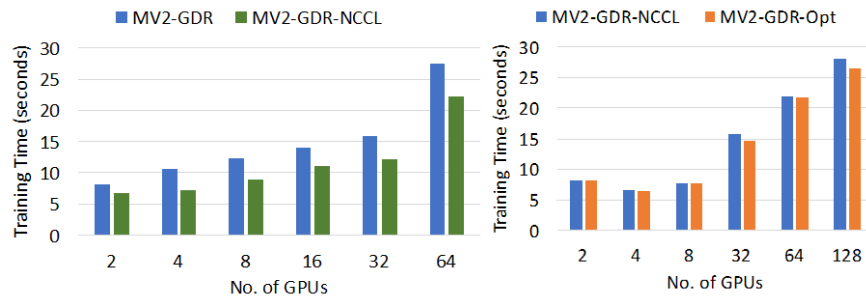
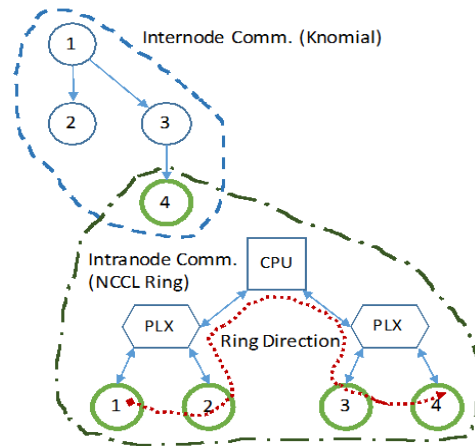
C.-H. Chu, X. Lu, A. A. Awan, H. Subramoni, J. Hashmi, B. Elton, and D. K. Panda, Efficient and Scalable Multi-Source Streaming Broadcast on GPU Clusters for Deep Learning, ICPP'17.

# Outline

- Overview of the MVAPICH2 Project
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
- What's new with MVAPICH2-GDR
  - Multi-stream Communication for IPC
  - CMA-based Intra-node Communication Support
  - Support for OpenPower and NVLink
  - Initial support for GPUDirect Async feature
- Streaming Support with IB Multicast and GDR
- **High-Performance Deep Learning (HiDL) with MVAPICH2-GDR**
- **Conclusions**

# Efficient Broadcast: MVAPICH2-GDR and NCCL

- NCCL 1.x had some limitations
  - Only worked for a single node; no scale-out on multiple nodes
  - Degradation across IOH (socket) for scale-up (within a node)
- We propose optimized MPI\_Bcast design that exploits NCCL [1]
  - Communication of very large GPU buffers
  - Scale-out on large number of dense multi-GPU nodes
- Hierarchical Communication that efficiently exploits:
  - CUDA-Aware MPI\_Bcast in MV2-GDR
  - NCCL Broadcast for intra-node transfers
- Can pure MPI-level designs be done that achieve similar or better performance than NCCL-based approach? [2]



## VGG Training with CNTK

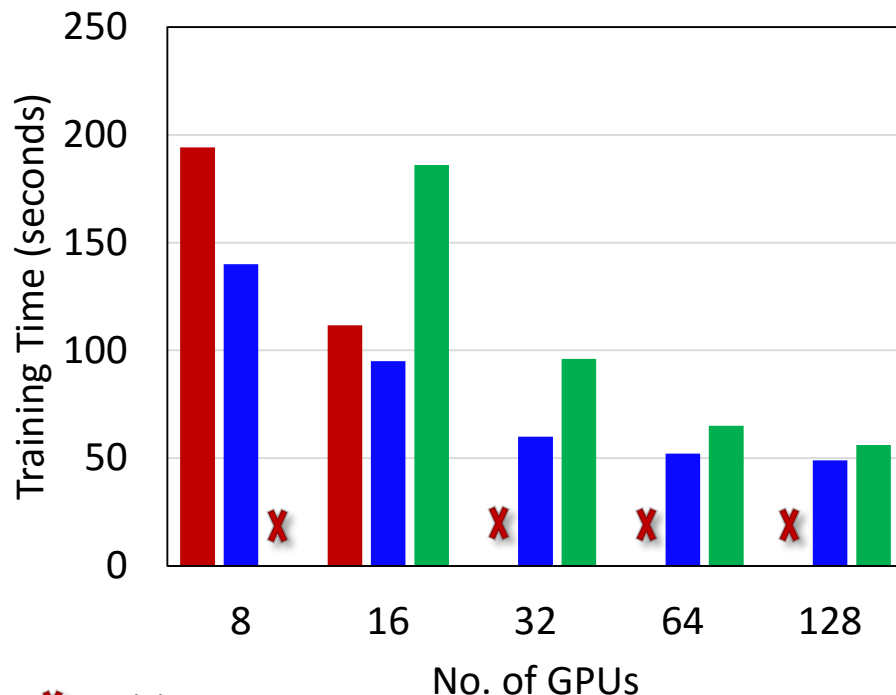
1. A. A. Awan, K. Hamidouche, A. Venkatesh, and D. K. Panda, Efficient Large Message Broadcast using NCCL and CUDA-Aware MPI for Deep Learning. In *Proceedings of the 23rd European MPI Users' Group Meeting (EuroMPI 2016)*. [Best Paper Nominee]

2. A. A. Awan, C-H. Chu, H. Subramoni, and D. K. Panda. Optimized Broadcast for Deep Learning Workloads on Dense-GPU InfiniBand Clusters: MPI or NCCL?, arXiv '17 (<https://arxiv.org/abs/1707.09414>)

# OSU-Caffe 0.9: Scalable Deep Learning on GPU Clusters

- Caffe : A flexible and layered Deep Learning framework.
- Benefits and Weaknesses
  - Multi-GPU Training within a single node
  - Performance degradation for GPUs across different sockets
  - Limited Scale-out
- OSU-Caffe: MPI-based Parallel Training
  - Enable Scale-up (within a node) and Scale-out (across multi-GPU nodes)
  - Scale-out on 64 GPUs for training CIFAR-10 network on CIFAR-10 dataset
  - Scale-out on 128 GPUs for training GoogLeNet network on ImageNet dataset

GoogLeNet (ImageNet) on 128 GPUs



X Invalid use case

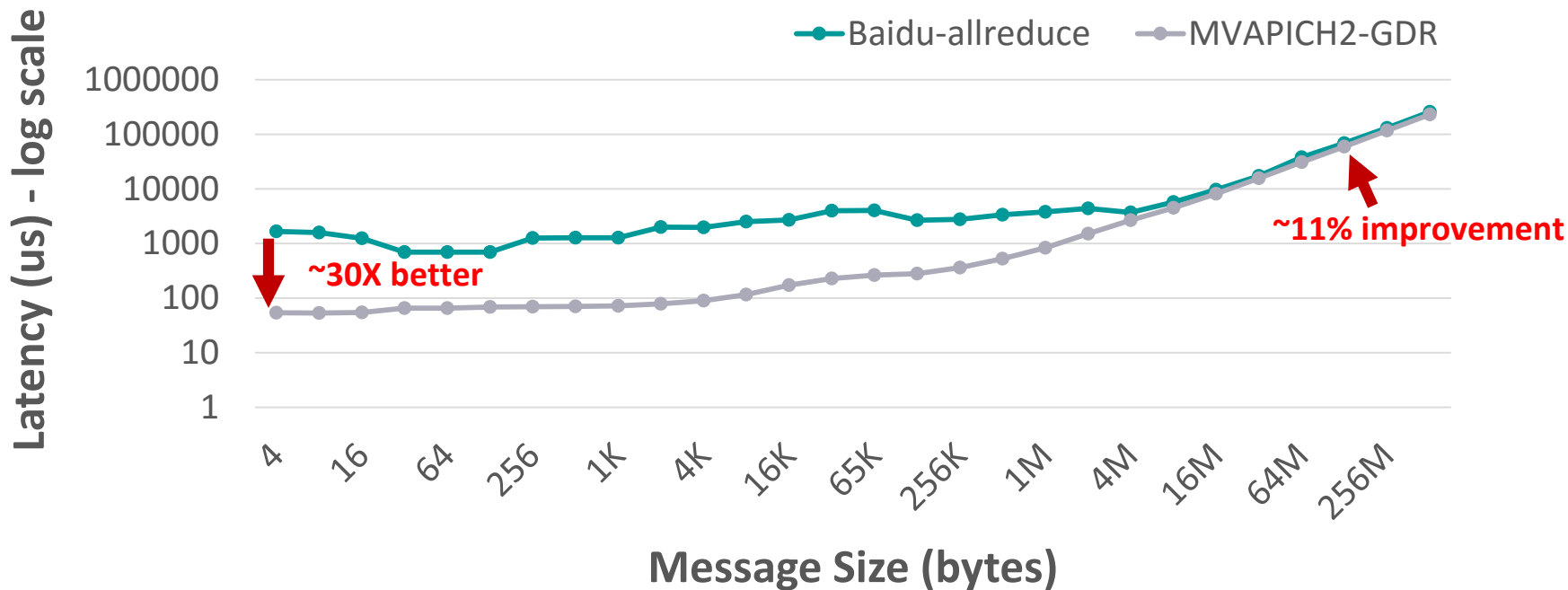
■ Caffe ■ OSU-Caffe (1024) ■ OSU-Caffe (2048)

OSU-Caffe 0.9 available from HiDL site

## Large Message Allreduce: MVAPICH2-GDR vs. Baidu-allreduce

- Performance gains for MVAPICH2-GDR 2.3a\* compared to Baidu-allreduce

8 GPUs (4 nodes log scale-allreduce vs MVAPICH2-GDR)

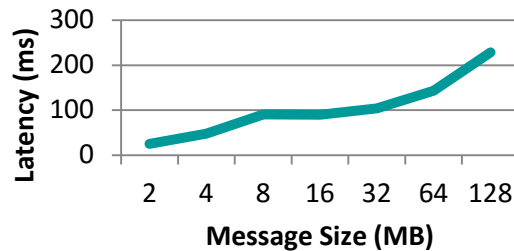


\*Available with MVAPICH2-GDR 2.3a

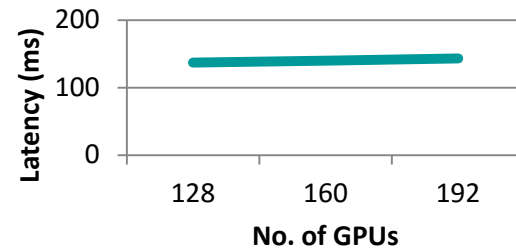
# Large Message Optimized Collectives for Deep Learning

- MVAPICH2-GDR provides optimized collectives for **large message sizes**
- Optimized Reduce, Allreduce, and Bcast
- **Good scaling with large number of GPUs**
- **Available in MVAPICH2-GDR 2.2 and higher**

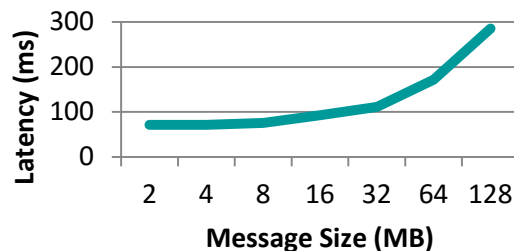
Reduce – 192 GPUs



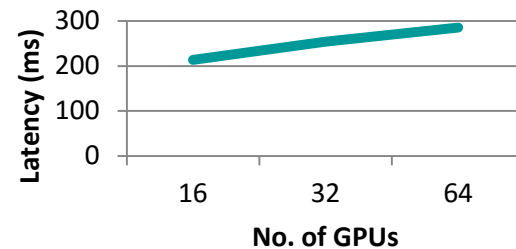
Reduce – 64 MB



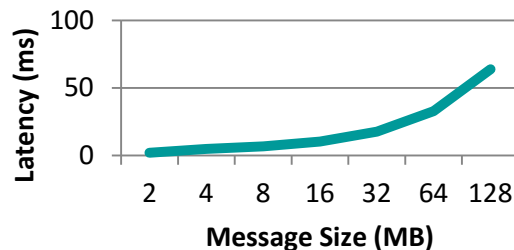
Allreduce – 64 GPUs



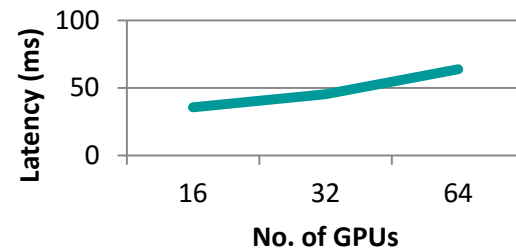
Allreduce - 128 MB



Bcast – 64 GPUs



Bcast 128 MB



# Outline

- Overview of the MVAPICH2 Project
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
- What's new with MVAPICH2-GDR
  - Multi-stream Communication for IPC
  - CMA-based Intra-node Communication Support
  - Support for OpenPower and NVLink
  - Initial support for GPUDirect Async feature
- Streaming Support with IB Multicast and GDR
- High-Performance Deep Learning (HiDL) with MVAPICH2-GDR
- **Conclusions**

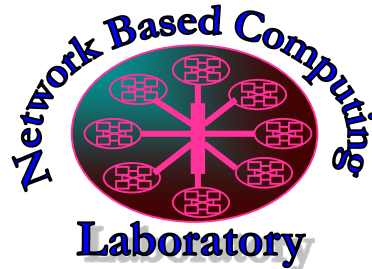


## Conclusions

- MVAPICH2 optimizes MPI communication on InfiniBand clusters with GPUs
- Provides optimized designs for point-to-point two-sided and one-sided communication, datatype processing and collective operations
- Takes advantage of CUDA features like IPC and GPUDirect RDMA families
- Allows flexible solutions for streaming applications with GPUs
- **HiDL: Accelerating your Deep Learning framework on HPC systems**
  - Tight interaction with MVAPICH2-GDR to boost the performance on GPU cluster
  - Scale-out to multi-GPU nodes

# Thank You!

[panda@cse.ohio-state.edu](mailto:panda@cse.ohio-state.edu)



Network-Based Computing Laboratory

<http://nowlab.cse.ohio-state.edu/>



The High-Performance MPI/PGAS Project  
<http://mvapich.cse.ohio-state.edu/>



The High-Performance Deep Learning Project  
<http://hidl.cse.ohio-state.edu/>