MVAPICH
MPI, PGAS and Hybrid MPI+PGAS Library

Overview of the MVAPICH Project: Latest Status and Future Roadmap

MVAPICH User Group (MUG) Conference

by

Dhabaleswar K. (DK) Panda

The Ohio State University

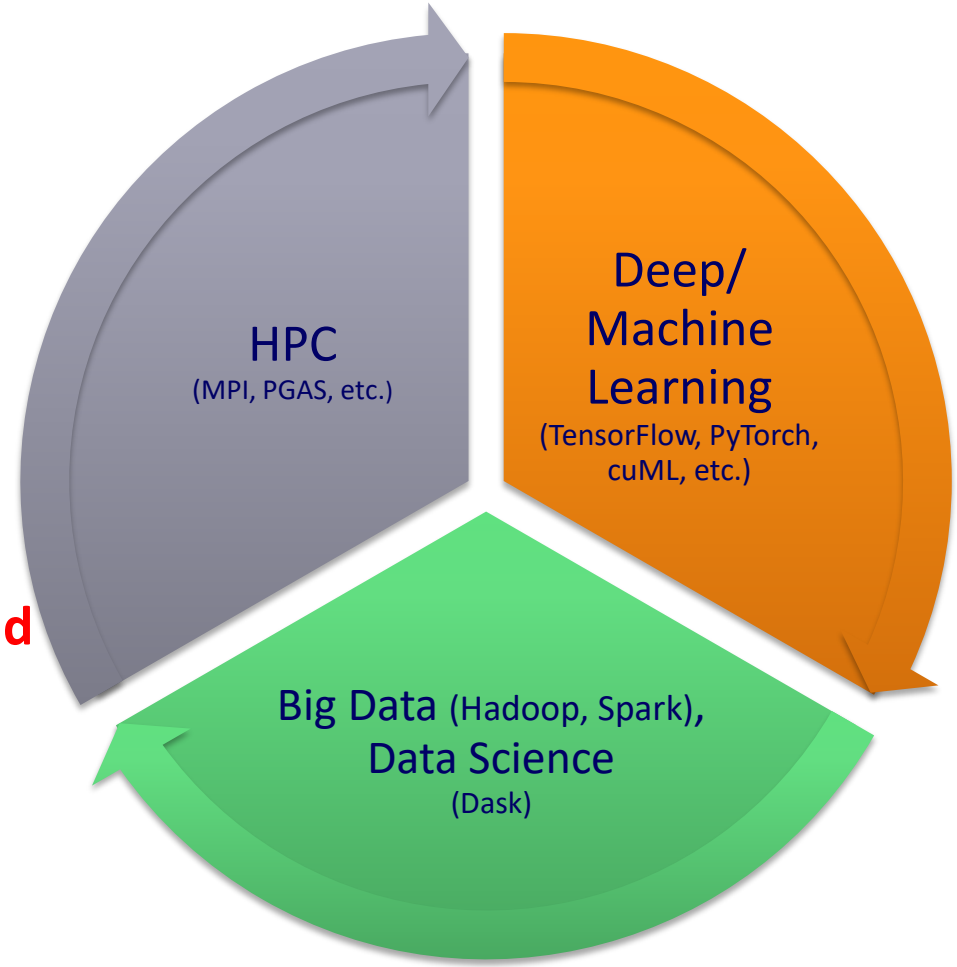
E-mail: panda@cse.ohio-state.edu

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

Computing has been evolving over the last three decades with multiple **phases:**

- **Phase 1 (1975-): Scientific Computing/HPC**
- **Phase 2 (2000-): HPC + Big Data Analytics**
- **Phase 3: (2010-): HPC + AI (Machine Learning/Deep Learning)**

Increasing Usage of HPC, AI, and Data Science in multiple Disciplines



Convergence of HPC, Deep/Machine Learning, and Data Science!

Increasing Need to Run these applications on the Cloud!!

Can MPI-Driven Converged Middleware be designed and used for all three domains?

Outline

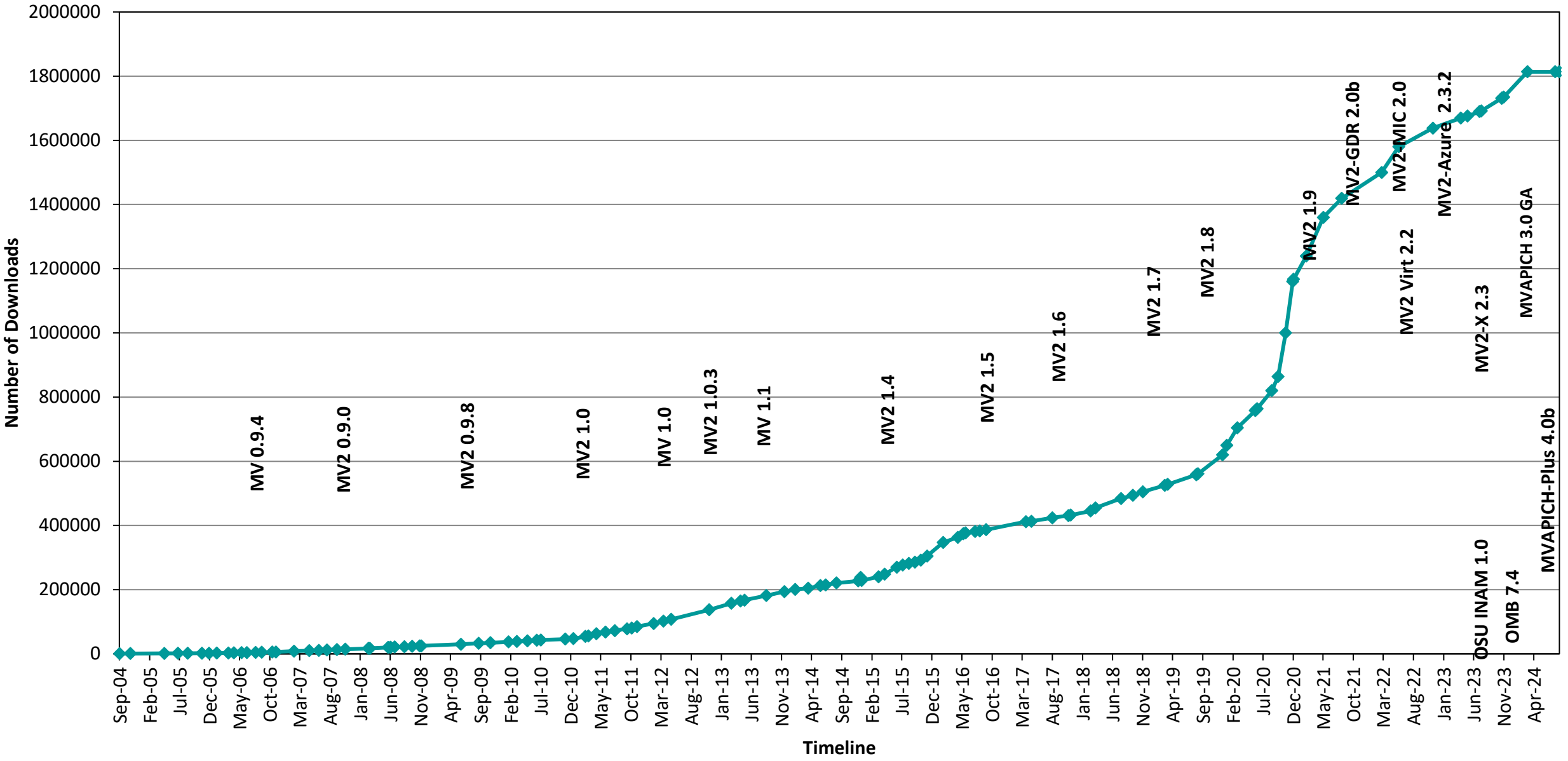
- **Brief Overview of the MVAPICH Project**
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - Optimized MVAPICH2-2.3.7+ for Broadcom RoCE
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

Overview of the MVAPICH Project

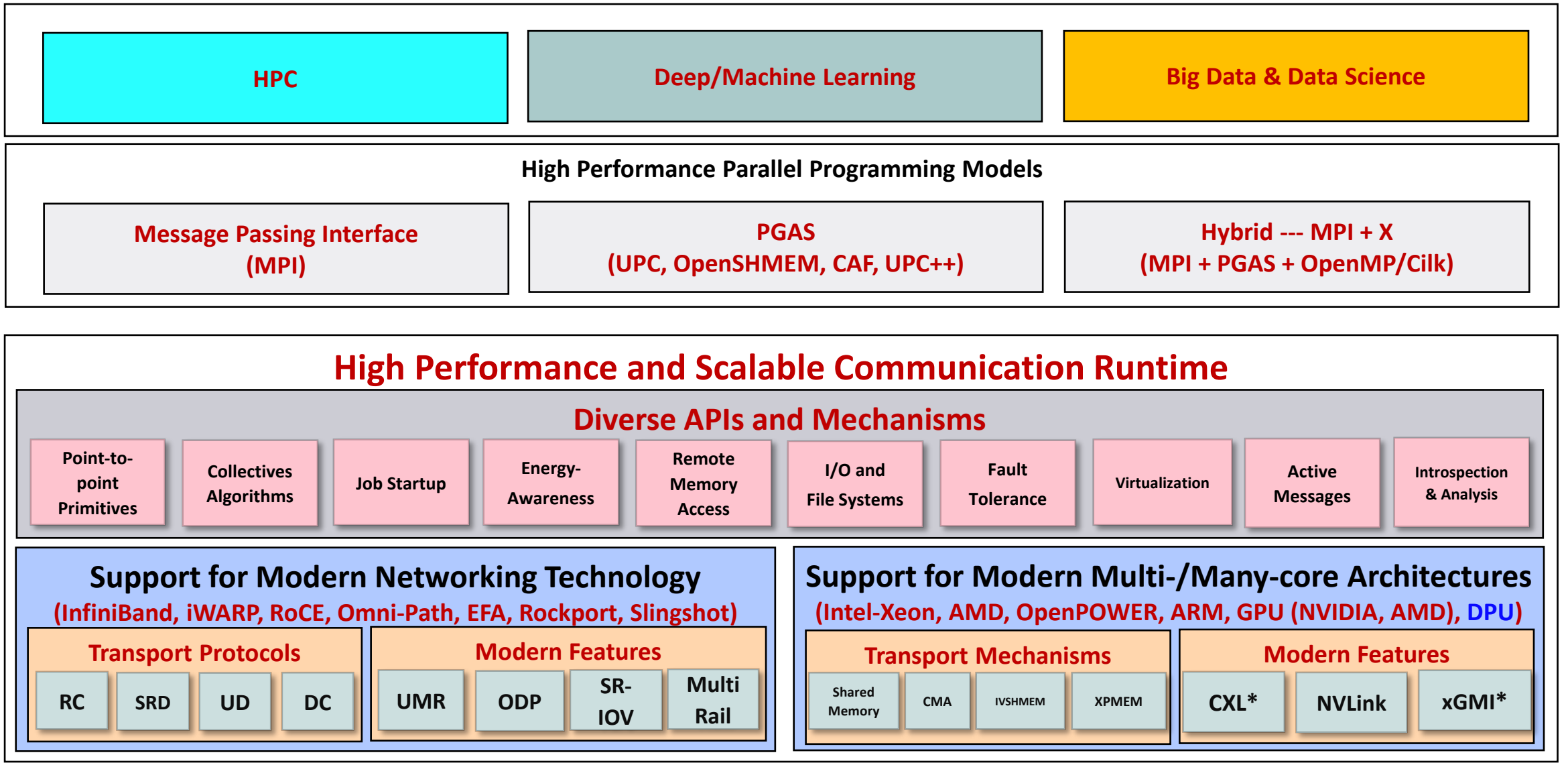
- High Performance open-source MPI Library
- Support for multiple interconnects
  - InfiniBand, Omni-Path, Ethernet/iWARP, RDMA over Converged Ethernet (RoCE), AWS EFA, OPX, Broadcom RoCE, Intel Ethernet, Rockport Networks, Slingshot 10/11
- Support for multiple platforms
  - x86, OpenPOWER, ARM, Xeon-Phi, GPGPUs (NVIDIA and AMD)
- Started in 2001, first open-source version demonstrated at SC '02
- Supports the latest MPI-3.1 standard
- http://mvapich.cse.ohio-state.edu
- Additional optimized versions for different systems/environments:
  - MVAPICH2-X (Advanced MPI + PGAS), since 2011
  - MVAPICH2-GDR with support for NVIDIA (since 2014) and AMD (since 2020) GPUs
  - MVAPICH2-MIC with support for Intel Xeon-Phi, since 2014
  - MVAPICH2-Virt with virtualization support, since 2015
  - MVAPICH2-EA with support for Energy-Awareness, since 2015
  - MVAPICH2-Azure for Azure HPC IB instances, since 2019
  - MVAPICH2-X-AWS for AWS HPC+EFA instances, since 2019
- Tools:
  - OSU MPI Micro-Benchmarks (OMB), since 2003
  - OSU InfiniBand Network Analysis and Monitoring (INAM), since 2015

- **Used by more than 3,400 organizations in 92 countries**
- **More than 1.81 Million downloads from the OSU site directly**
- Empowering many TOP500 clusters (June '24 ranking)
  - 13th , 10,649,600-core (Sunway TaihuLight) at NSC, Wuxi, China
  - 33rd , 448, 448 cores (Frontera) at TACC
  - 57th, 288,288 cores (Lassen) at LLNL
  - 75th, 570,020 cores (Nurion) in South Korea and many others
- Available with software stacks of many vendors and Linux Distros (RedHat, SuSE, OpenHPC, and Spack)
- Partner in the 33rd ranked TACC Frontera system
- **Empowering Top500 systems for more than 19 years**

MVAPICH Release Timeline and Downloads



MVAPICH Architecture (HPC, DL/ML, Big Data, & Data Science)



* Upcoming

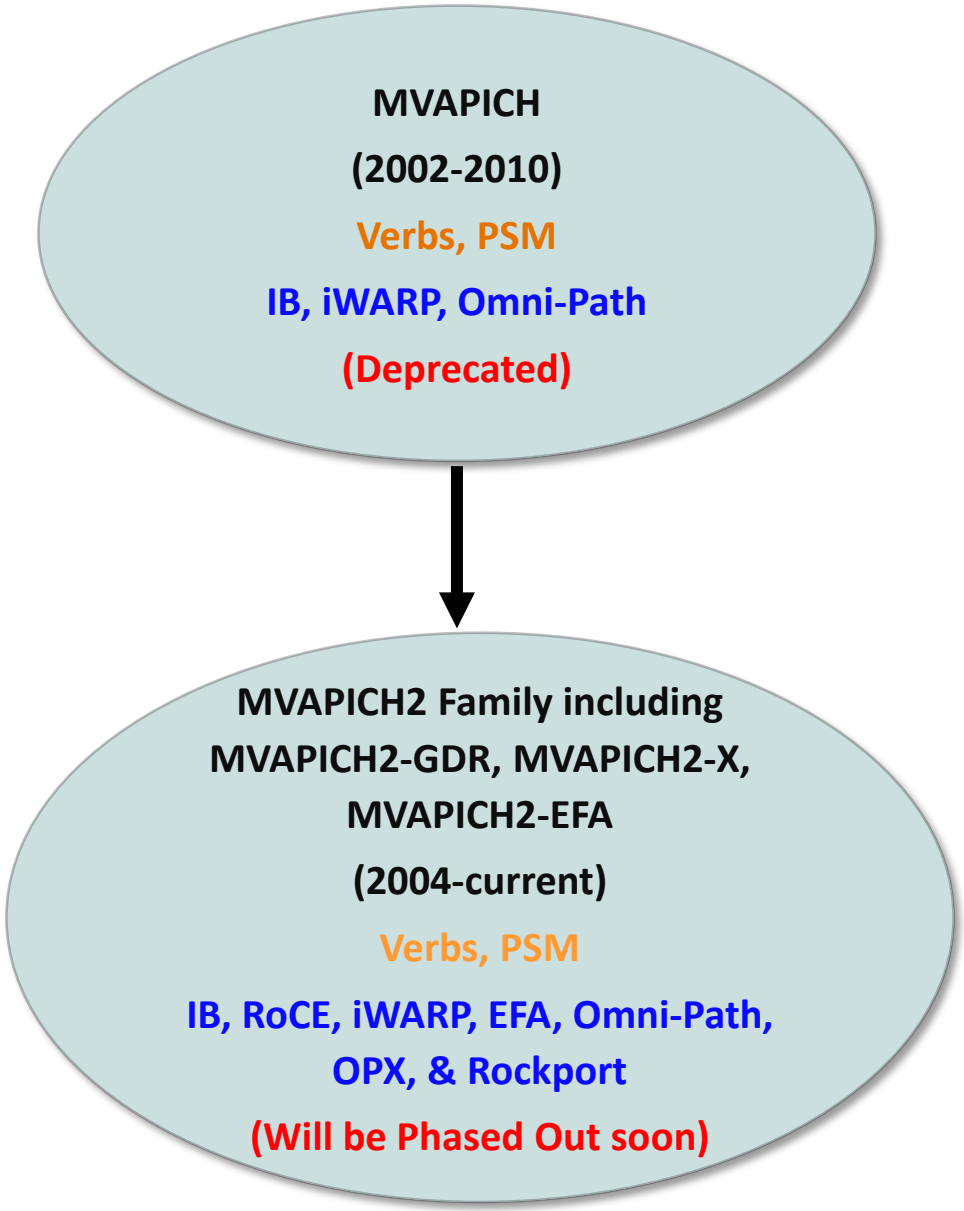
Production Quality Software Design, Development and Release

- Rigorous Q&A procedure before making a release
  - Exhaustive unit testing
  - Various test procedures on diverse range of platforms and interconnects
  - Test 19 different benchmarks and applications including, but not limited to
    - OMB, IMB, MPICH Test Suite, Intel Test Suite, NAS, Scalapak, and SPEC
  - Spend about 18,000 core hours per commit
  - Performance regression and tuning
  - Applications-based evaluation
  - Evaluation on large-scale systems
- All versions (alpha, beta, RC1 and RC2) go through the above testing

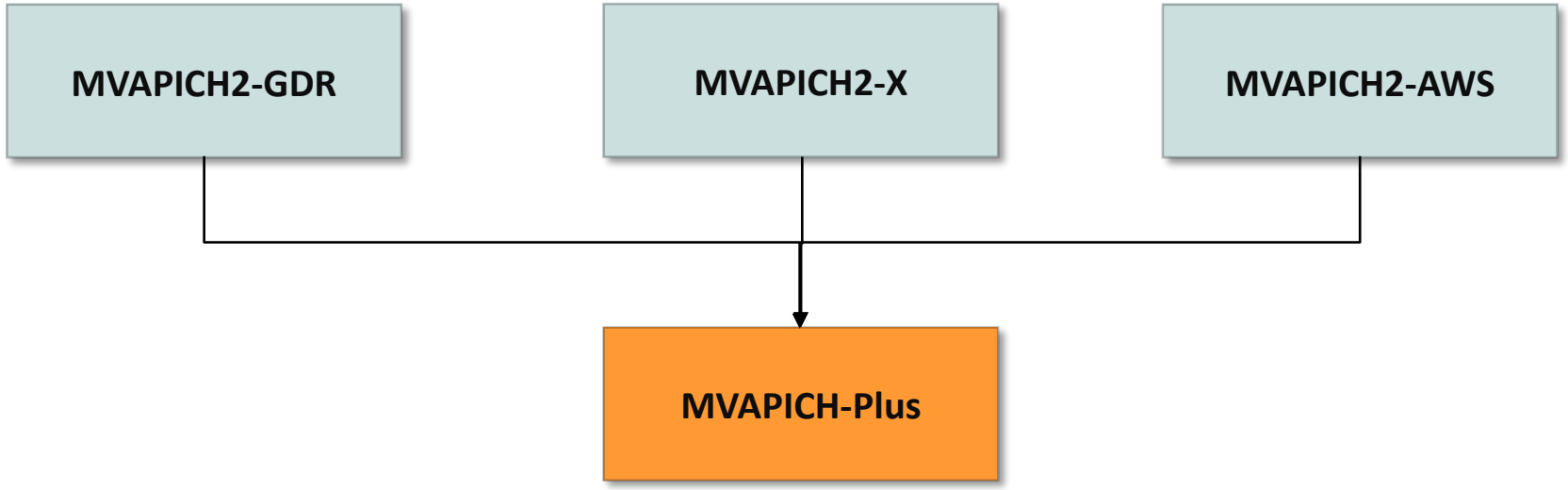
Outline

- Brief Overview of the MVAPICH Project
- **New MVAPICH-Plus Series**
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - Optimized MVAPICH2-2.3.7+ for Broadcom RoCE
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

Evolution of the MVAPICH/MVAPICH2 Project

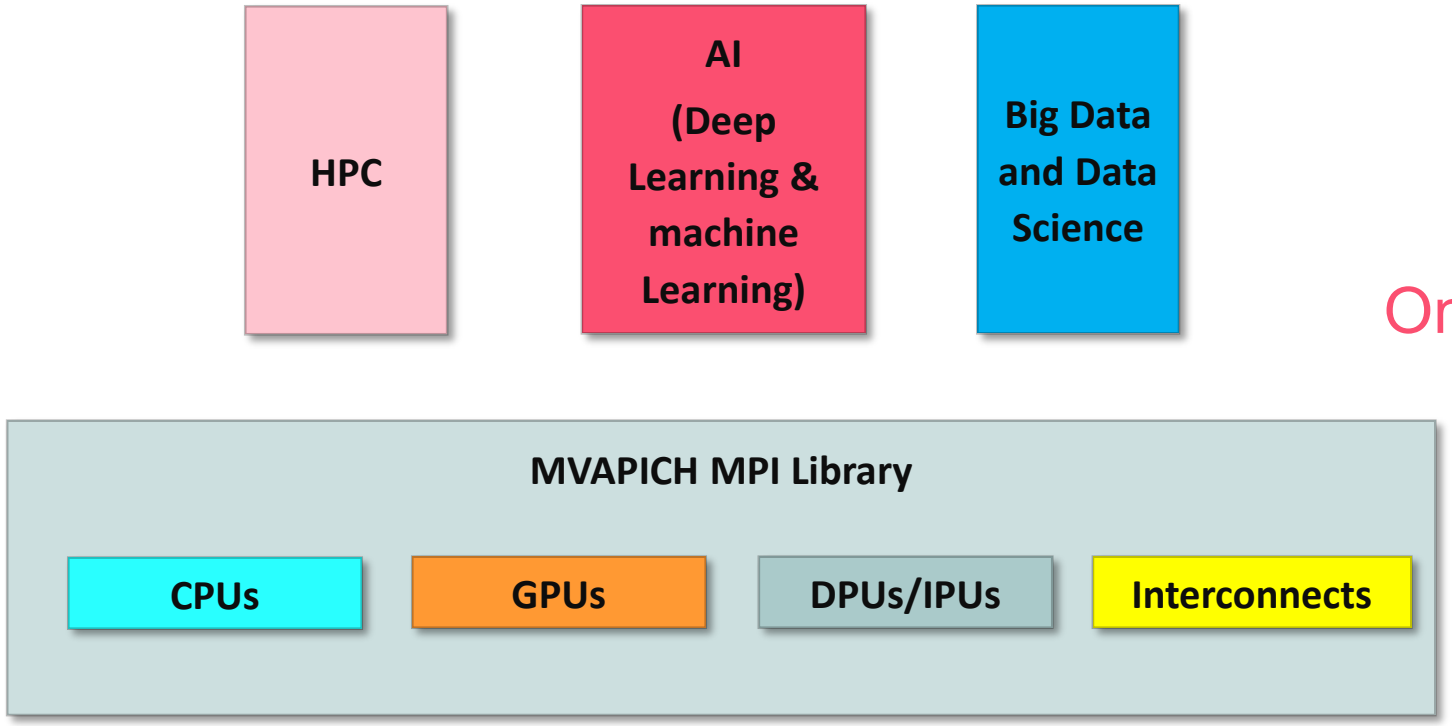


MVAPICH-Plus: One Stack for all Architectures and Interconnects



- Various libraries are being merged into one release
- Simplifies installation and deployment
- Enables utilizing the best advanced features for all architectures

MPI (MVAPICH)-driven Converged Software Stack for HPC, AI, Big Data, and Data Science

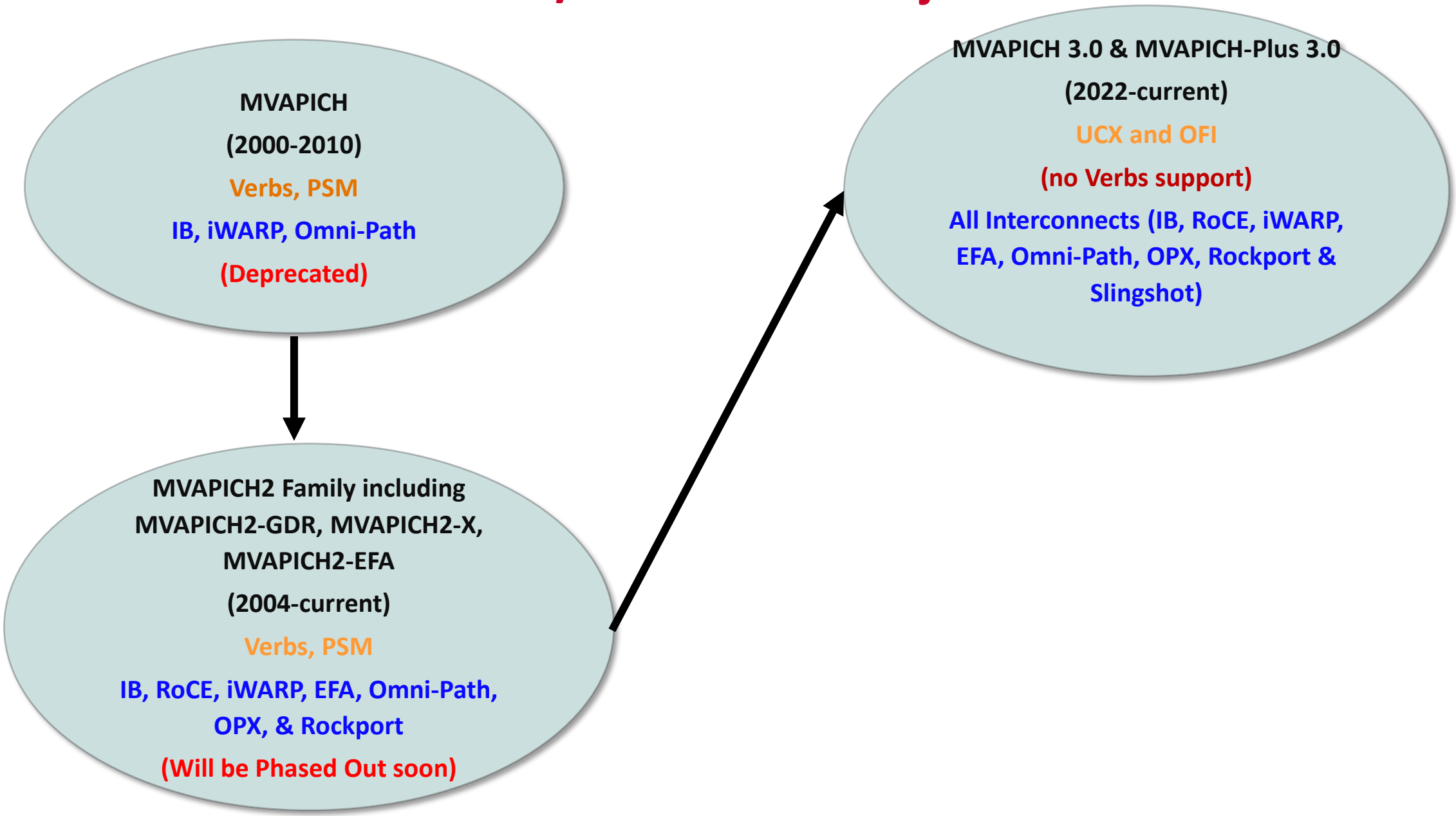


Unified Support for
On-Premise and Clouds

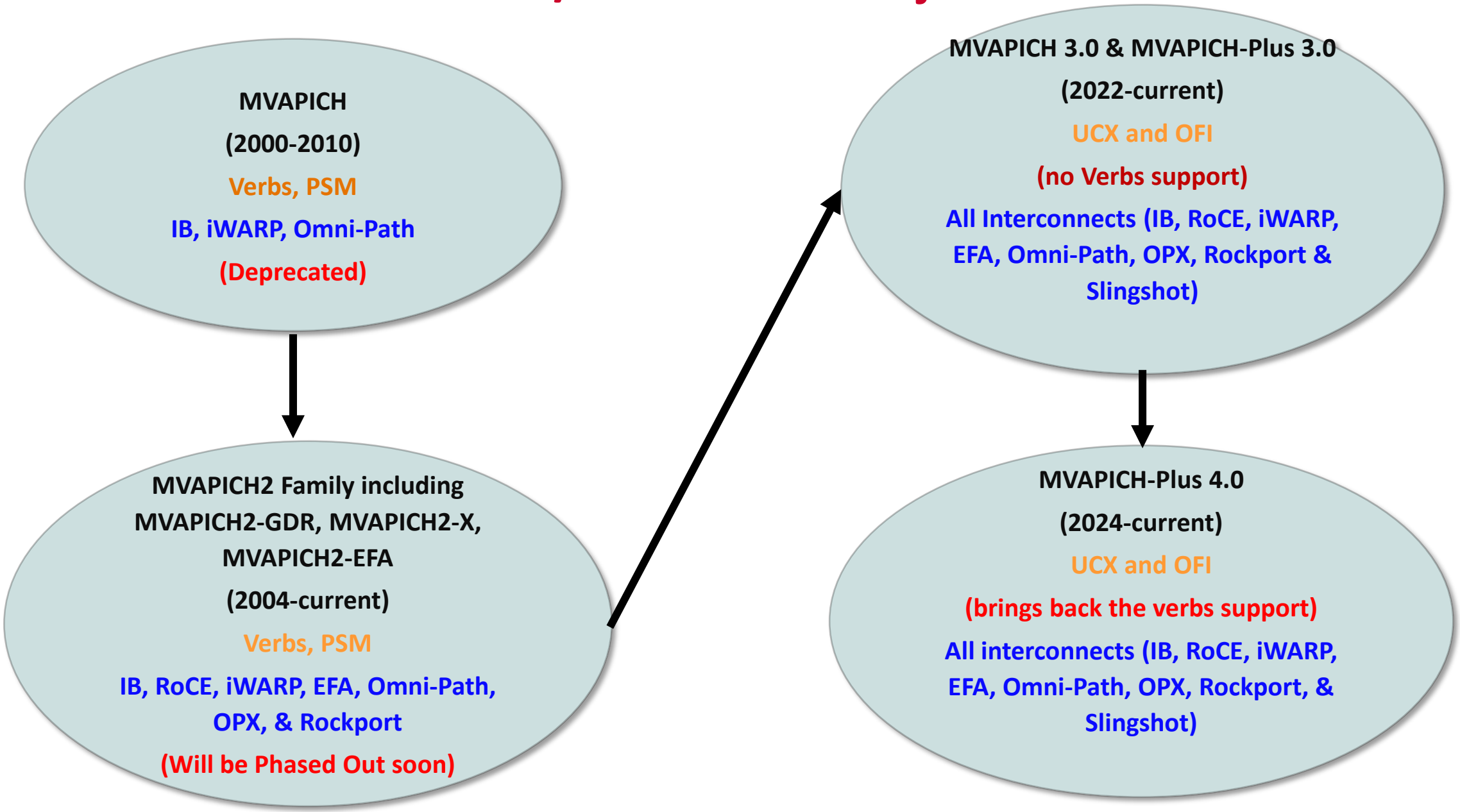
Scaling &
Performance Portability

Vendor Neutrality
CPU (3) x Interconnects (4) x GPUs (3)

Evolution of the MVAPICH/MVAPICH2 Project



Evolution of the MVAPICH/MVAPICH2 Project



Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - **MVAPICH-Plus 4.0b**
  - Optimized MVAPICH2-2.3.7+ for Broadcom RoCE
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

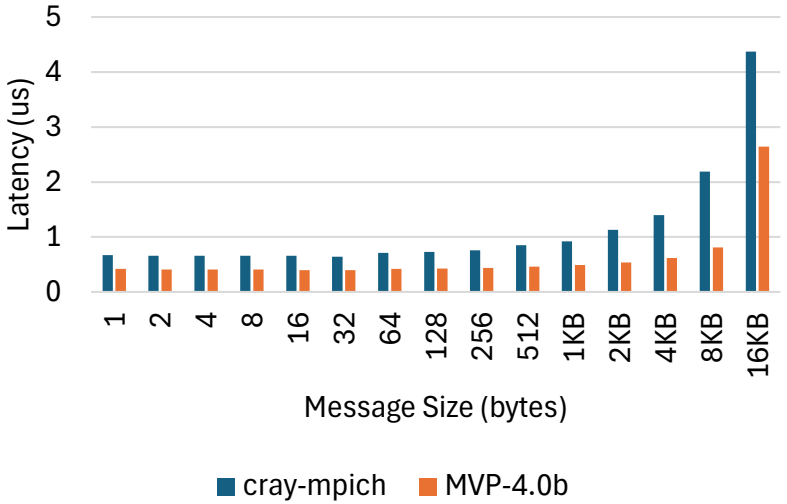
MVAPICH-Plus 4.0b

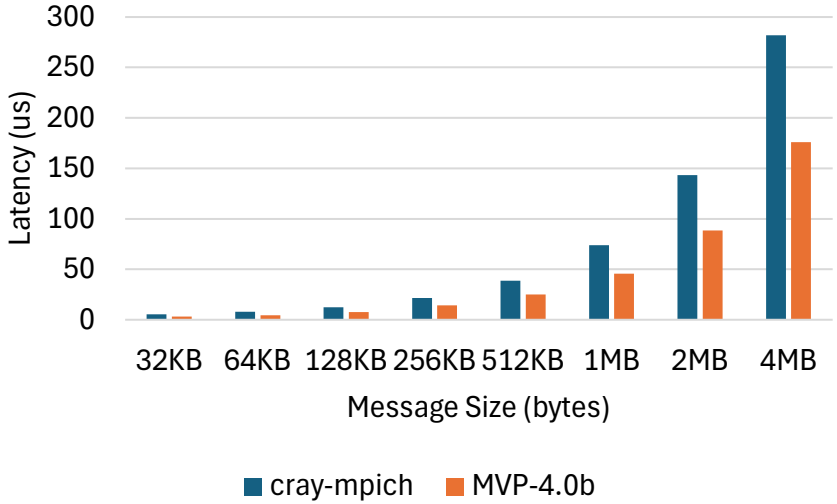
- Released on 08/26/2024
- Supports all features of MPI 4.1 standard
- Advanced MPI with unified MVAPICH2-GDR and MVAPICH2-X features
- Support for both OFI and UCX
- Support for
  - Major CPUs (x86-Intel, x86-AMD, and ARM)
  - Major GPUs (NVIDIA, AMD, and Intel)
  - Major HPC interconnects (InfiniBand, Omni-Path, ROCE, Slingshot, OPX, Ethernet/iWARP)
- Support for on-the-fly compression (Allgather, Alltoall, Allreduce, and Reduce_Scatter)
- Optimized designs for converged software stack with next-generation workflows
  - DL (MPI4DL)
    - Available from hidl.cse.ohio-state.edu
  - ML (MPI4cuML)
    - Available from hidl.cse.ohio-state.edu
  - Big Data (MPI4Spark)
    - Available from hibd.cse.ohio-state.edu
  - Data Science (MPI4Dask)
    - Available from hibd.cse.ohio-state.edu

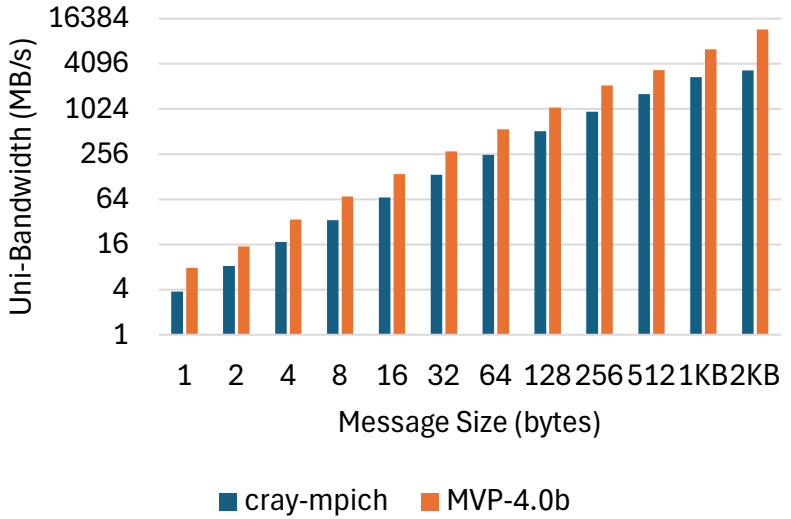
Sample Performance Numbers with MVAPICH-Plus 4.0b

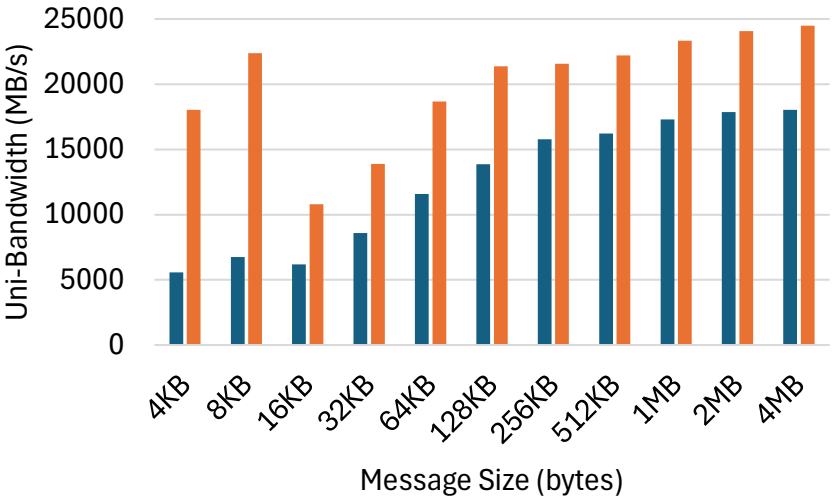
- AMD CPU + AMD GPU + Slingshot (Frontier@ORNL and Tioga@LLNL)
  - CPU-based
  - GPU-based
- Grace (ARM) CPU + Hopper (NVIDIA GPU) + Slingshot (Isamabard@Bristol)
  - GPU-based

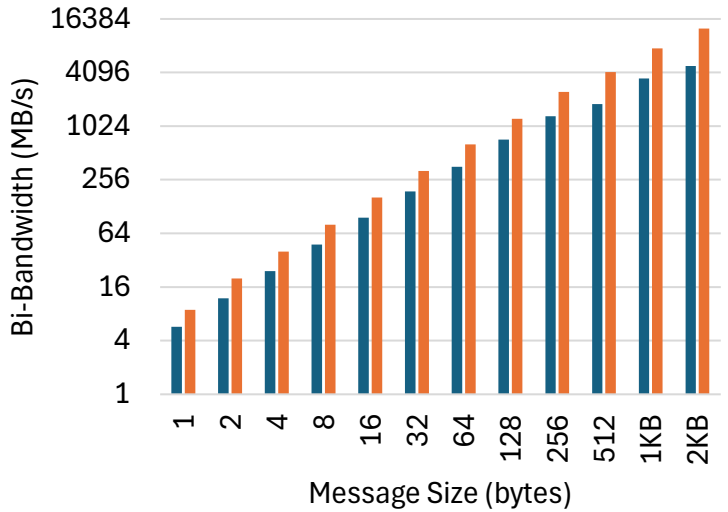
Intra-node CPU-CPU Results against Cray-MPICH on Frontier (Slingshot 11)

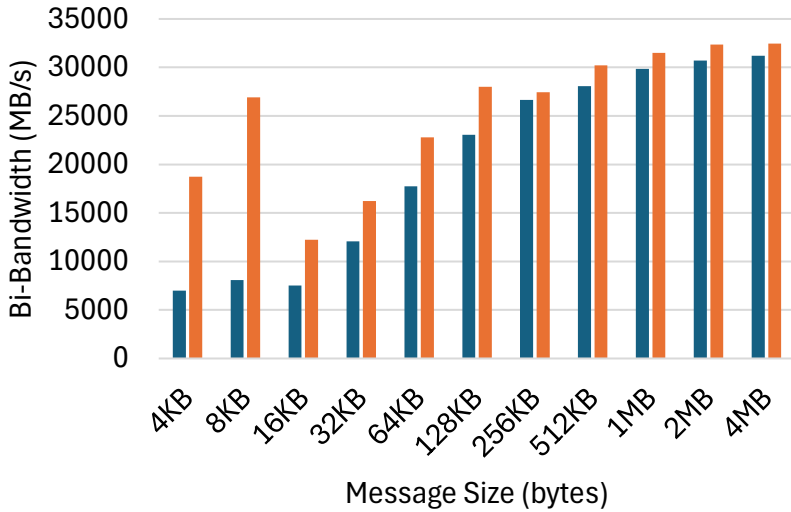




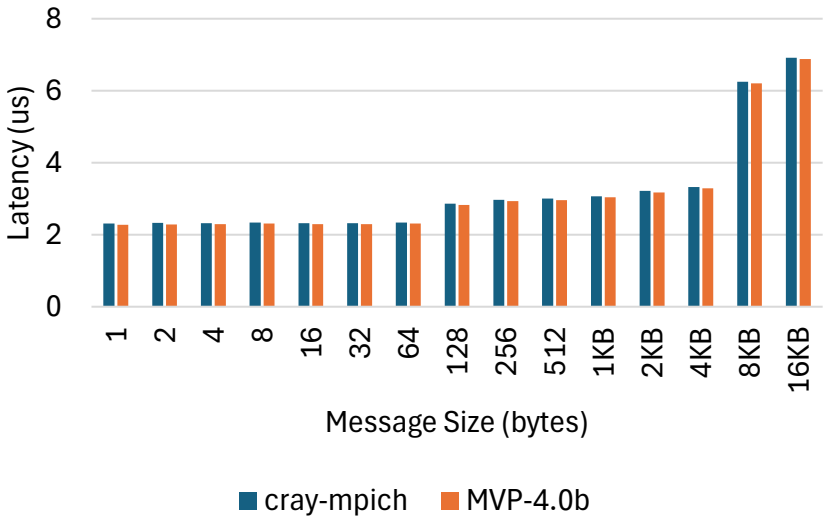


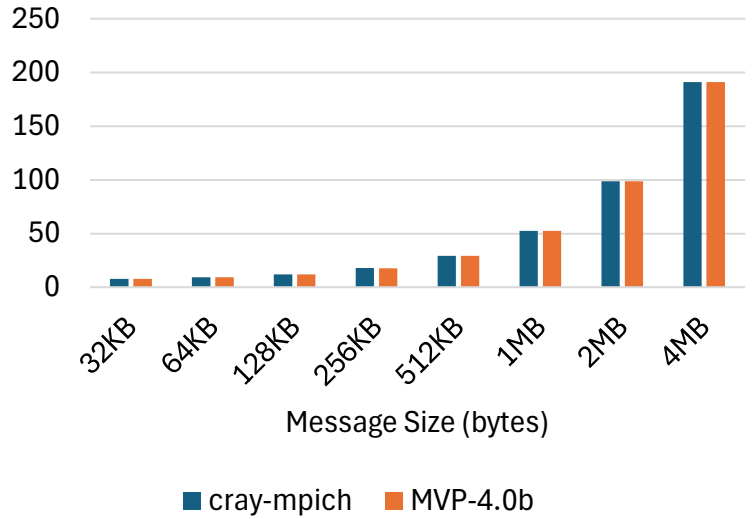


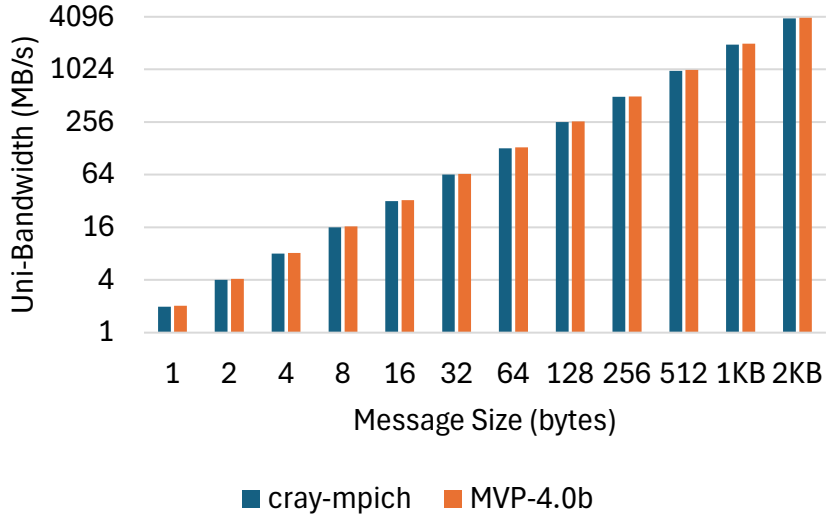


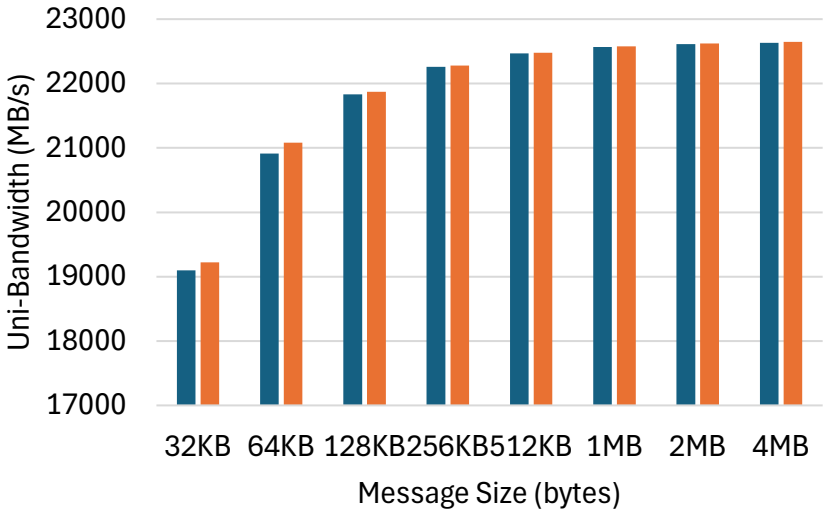


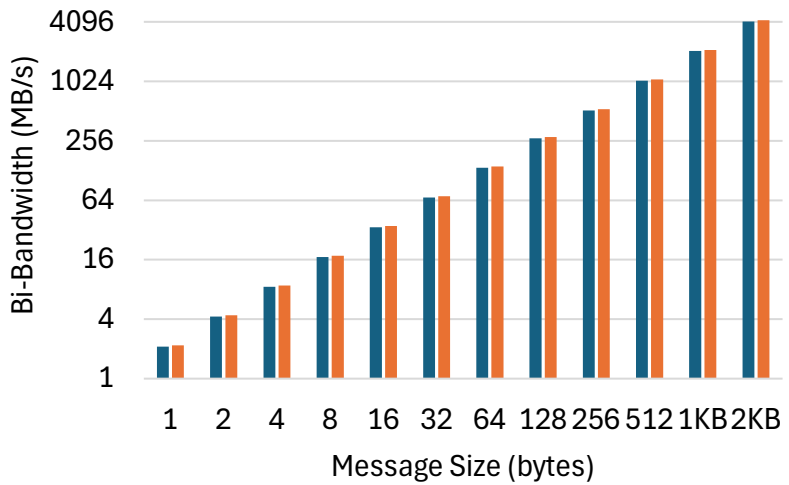
Inter-Node CPU-CPU Results against Cray-MPICH on Frontier (Slingshot 11)

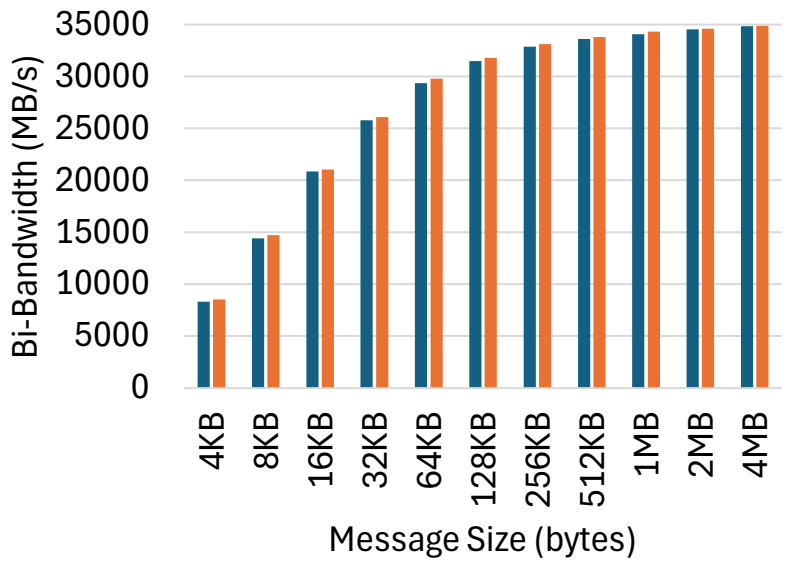




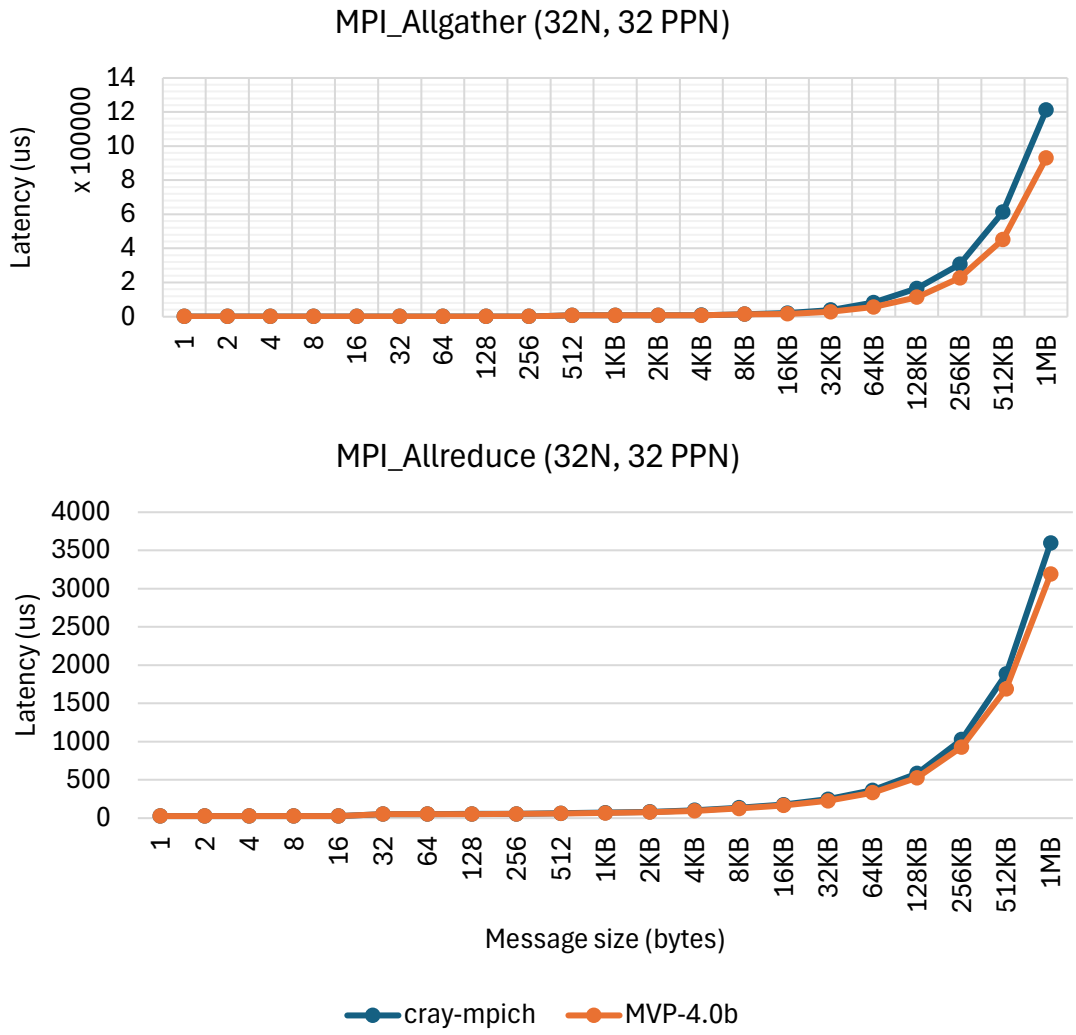


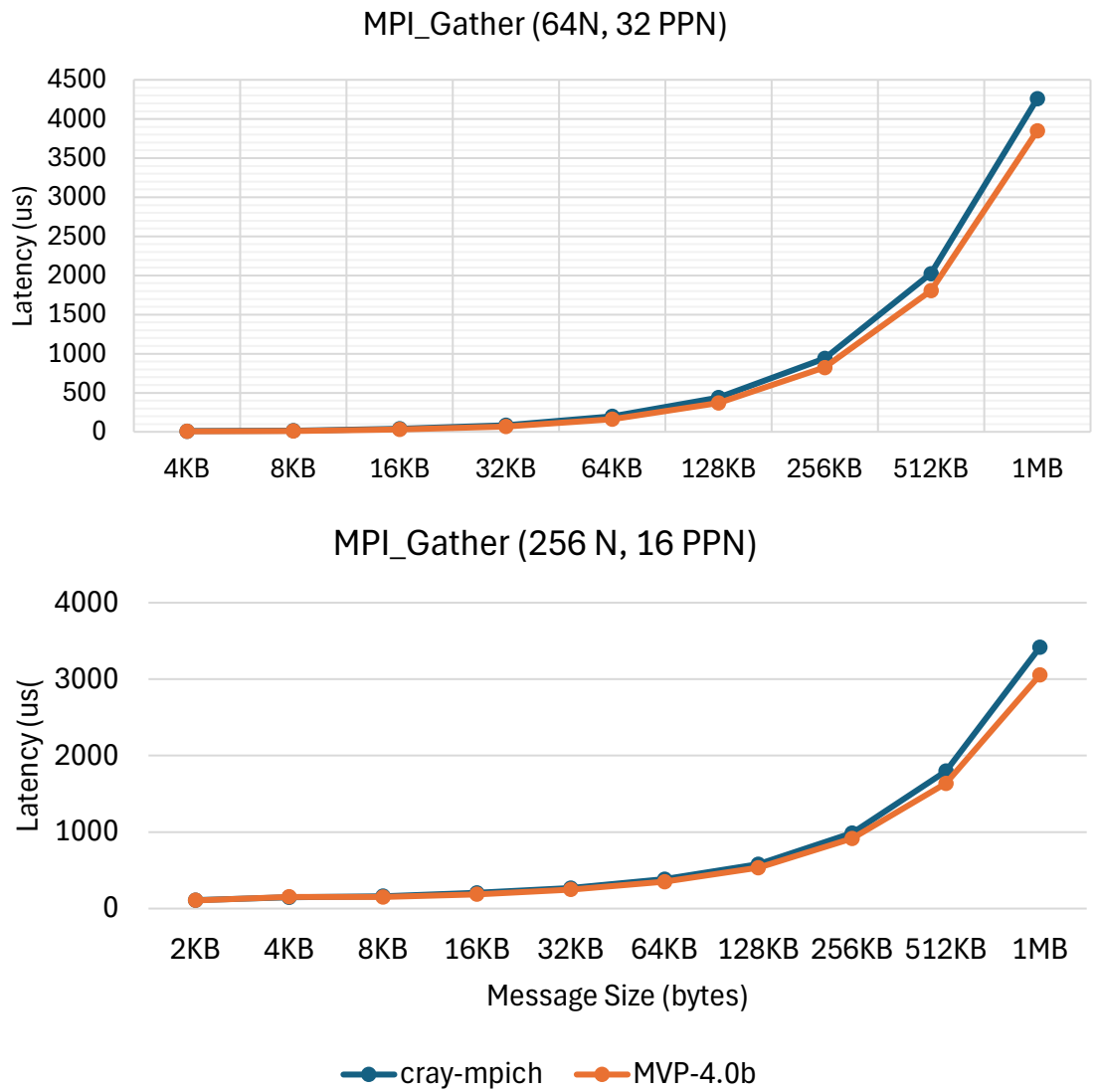






Large-Scale Collective Performance on Frontier against Cray-MPICH (Slingshot 11)

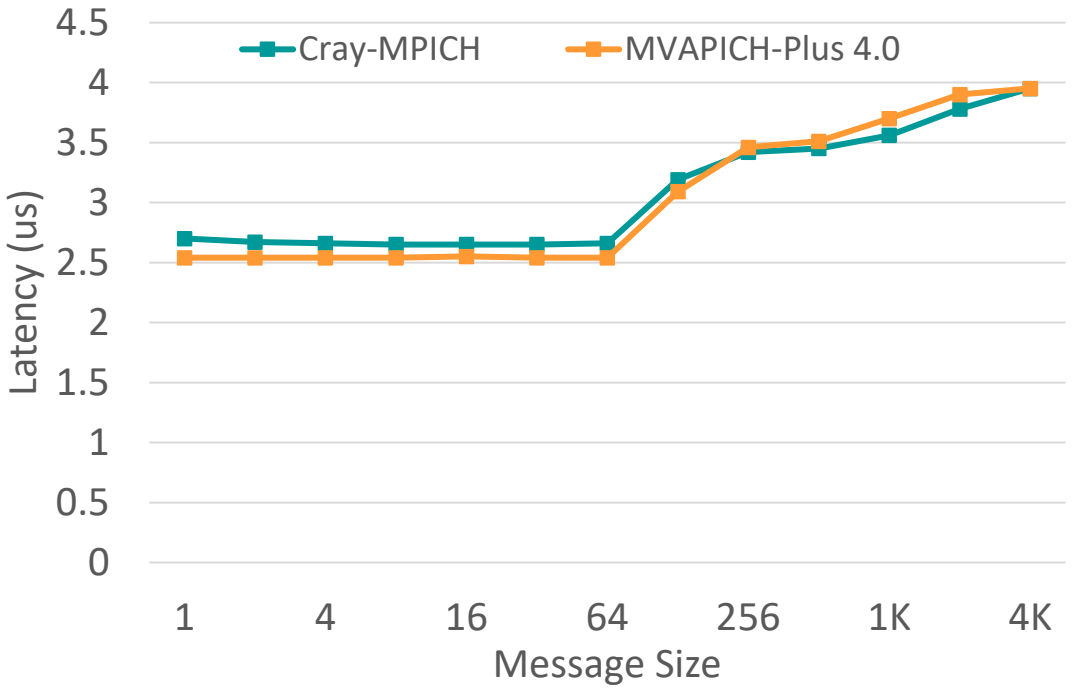




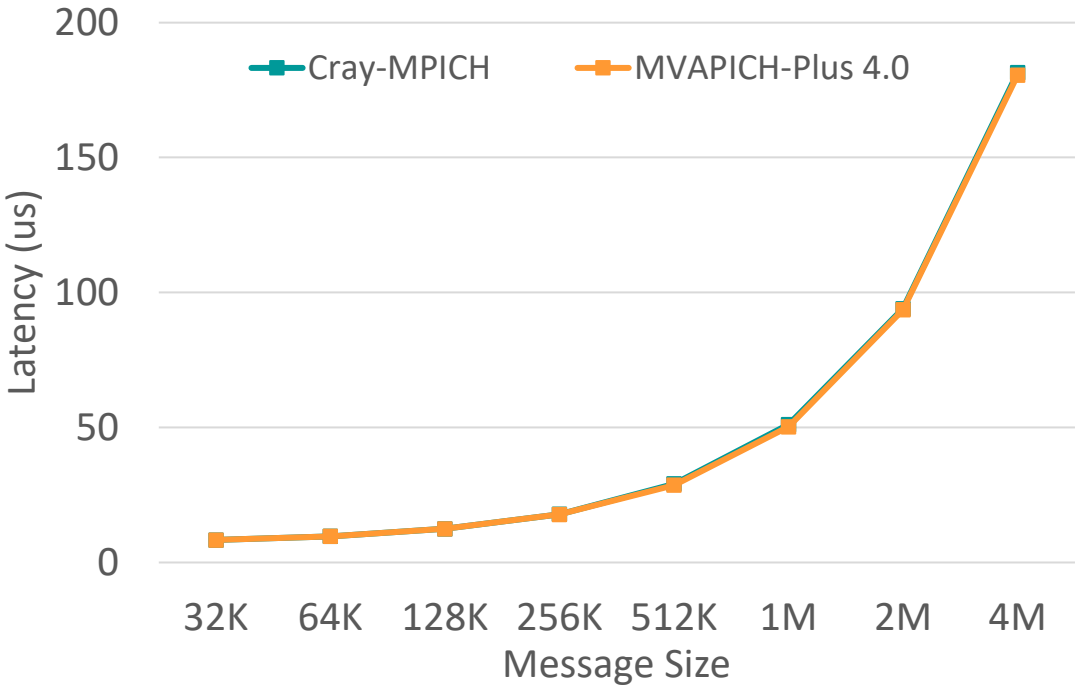
Sample Performance Numbers with MVAPICH-Plus 4.0b

- AMD CPU + AMD GPU + Slingshot (Frontier@ORNL and Tioga@LLNL)
  - CPU-based
  - GPU-based
- Grace (ARM) CPU + Hopper (NVIDIA GPU) + Slingshot (Isamabard@Bristol)
  - GPU-based

MVAPICH-PLUS GPU Optimized on Tioga (AMD MI250X GPUs) – Internode PT2PT

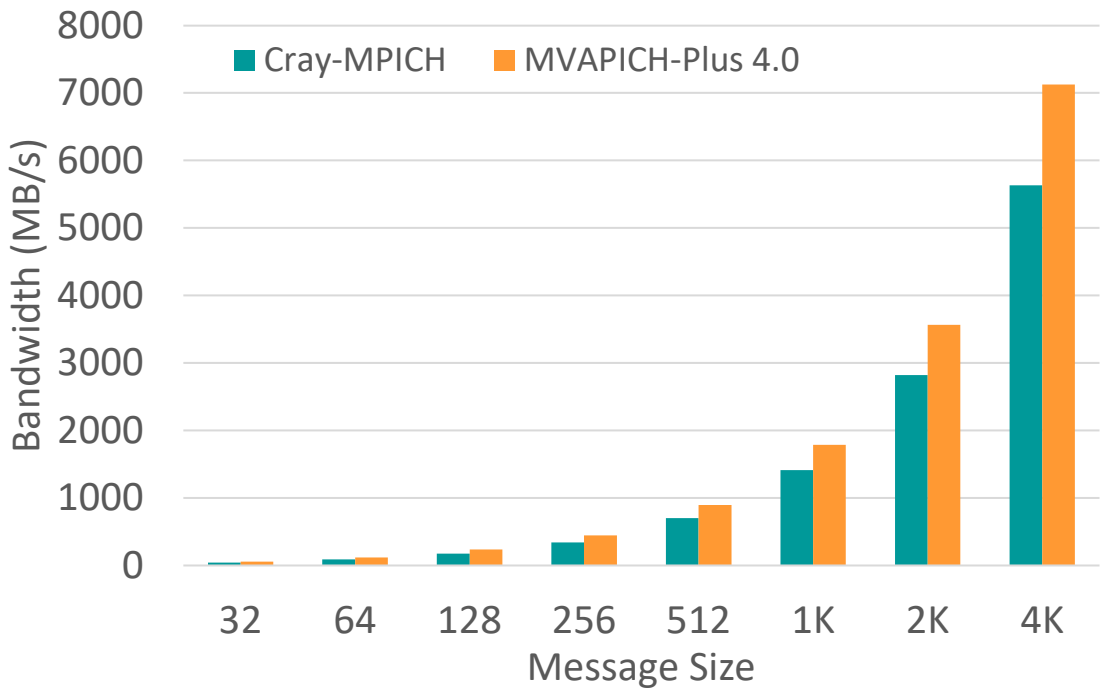


2 nodes, 1 GPN – Small Message (osu_latency)



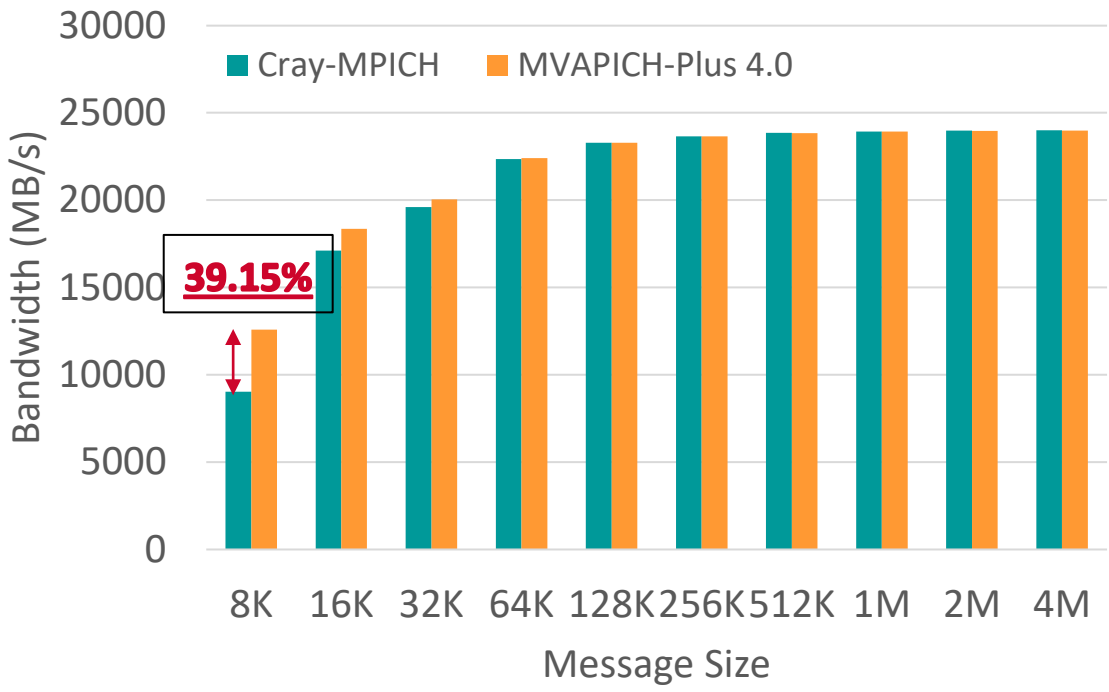
2 nodes, 1 GPN – Large Message (osu_latency)

MVAPICH-PLUS GPU Optimized on Tioga (AMD MI250X GPUs) – Internode PT2PT



2 nodes, 1 GPN – Small Message

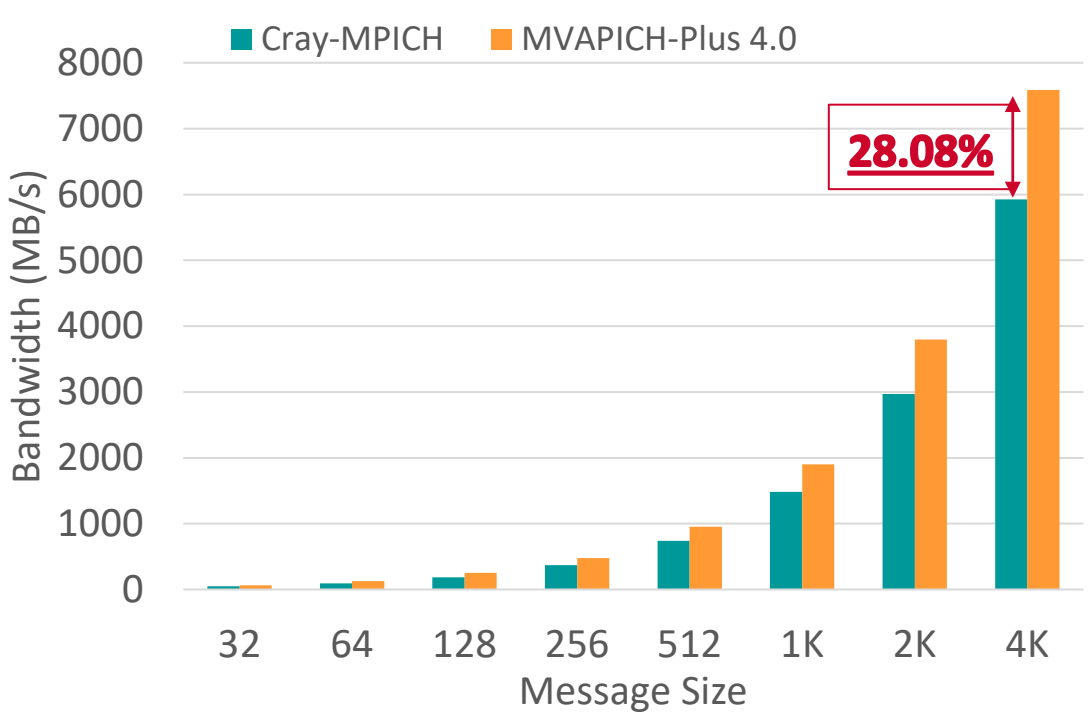
(osu_bw)



2 nodes, 1 GPN – Large Message

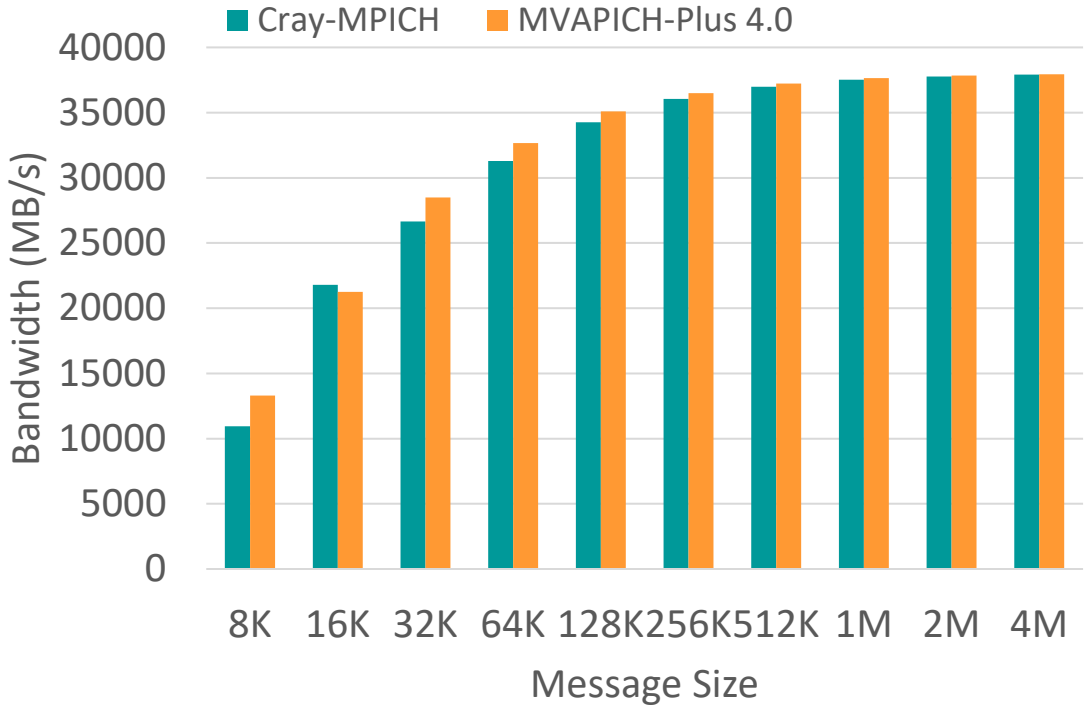
(osu_bw)

MVAPICH-PLUS GPU Optimized on Tioga (AMD MI250X GPUs) – Internode PT2PT



2 nodes, 1 GPN – Small Message

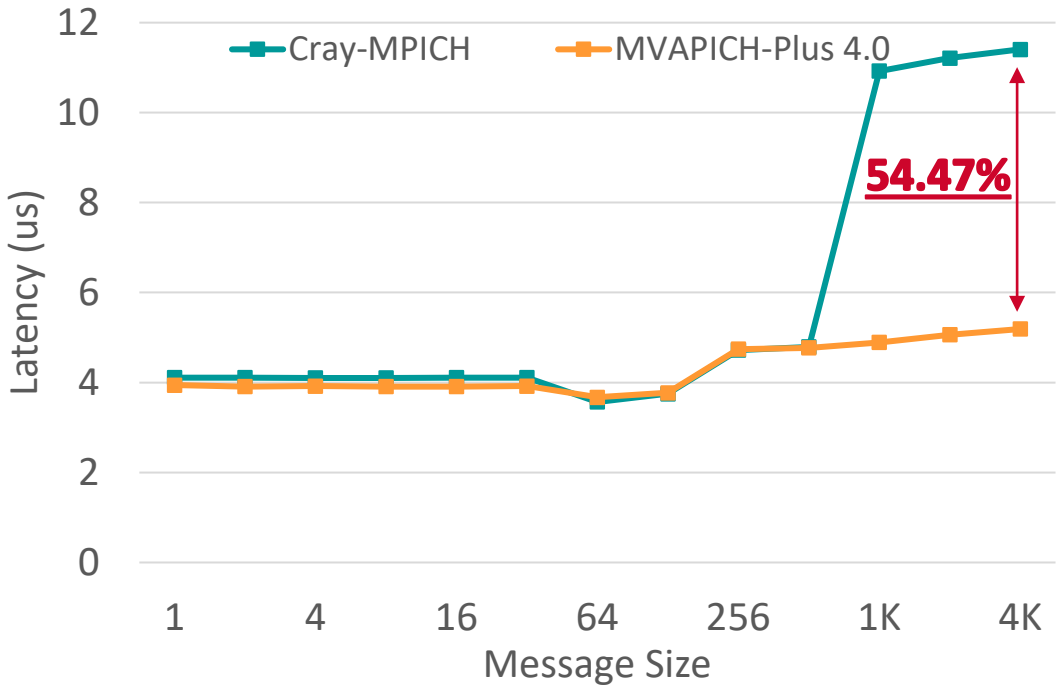
(osu_bibw)



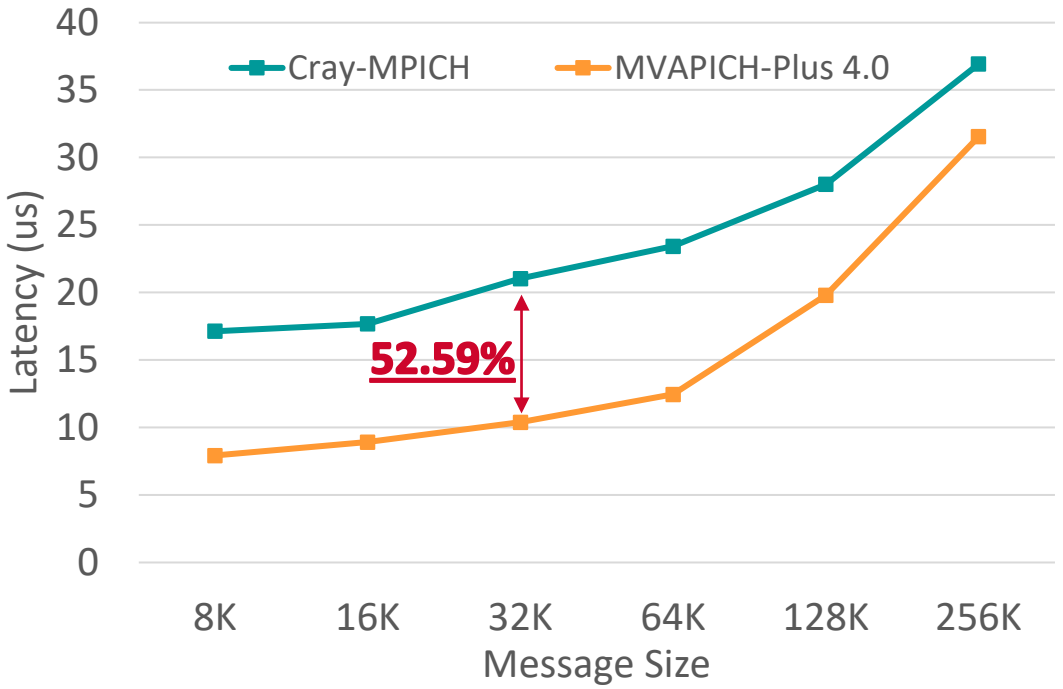
2 nodes, 1 GPN – Large Message

(osu_bibw)

MVAPICH-PLUS GPU Optimized on Tioga (AMD MI250X GPUs) – AlltoAll



2 nodes, 1 GPN – Small Message

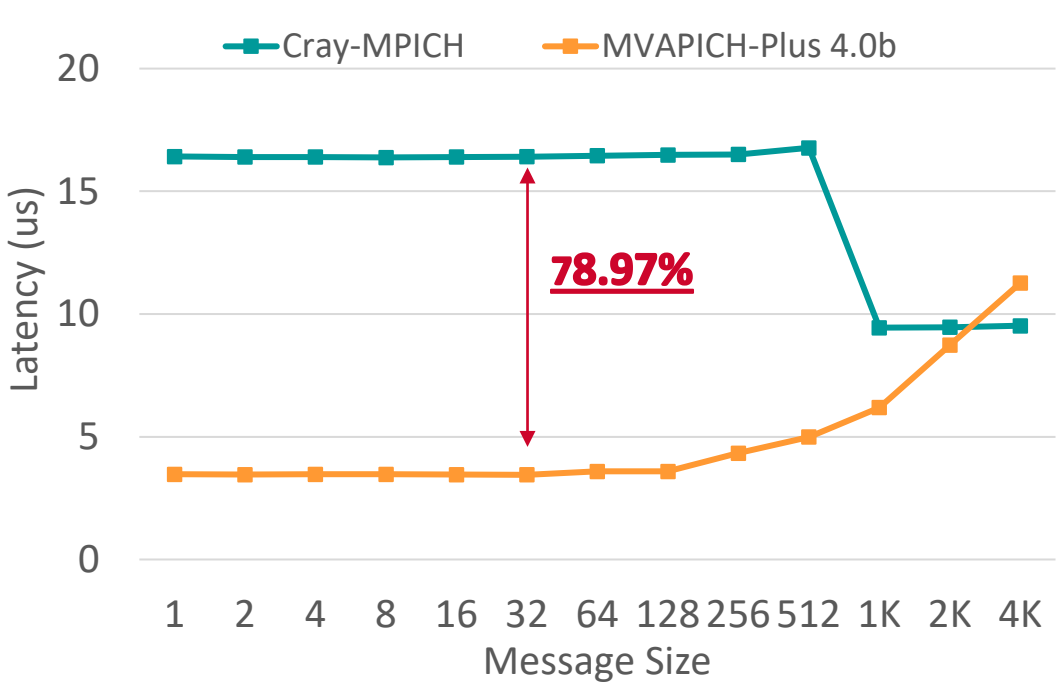


2 nodes, 1 GPN – Large Message

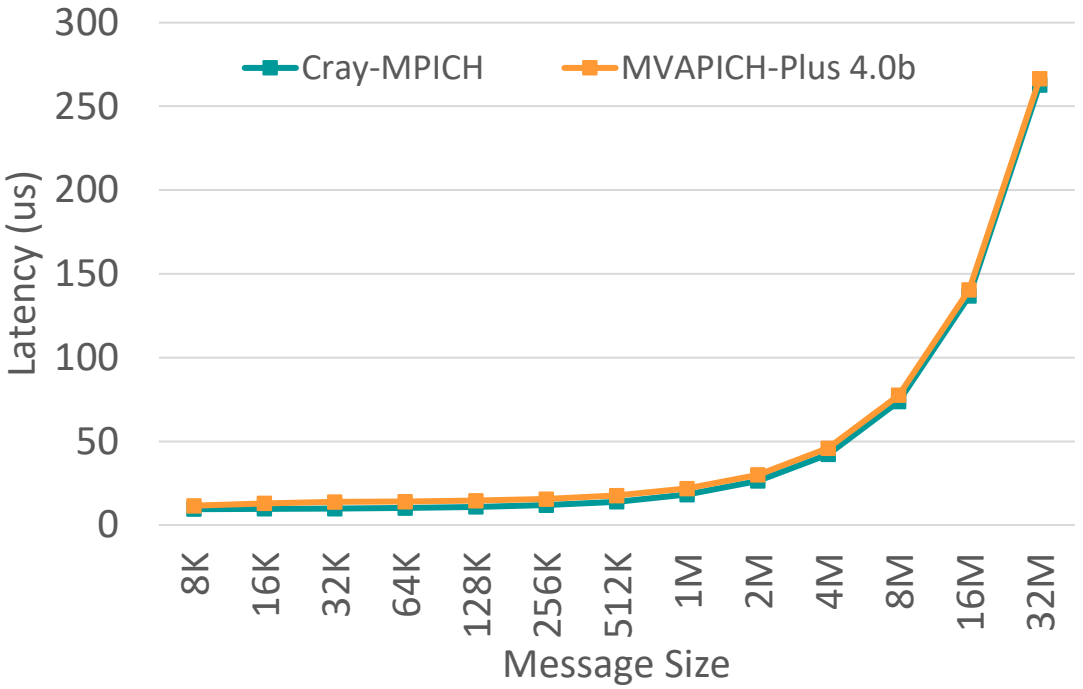
Sample Performance Numbers with MVAPICH-Plus 4.0b

- AMD CPU + AMD GPU + Slingshot (Frontier@ORNL and Tioga@LLNL)
  - CPU-based
  - GPU-based
- **Grace (ARM) CPU + Hopper (NVIDIA GPU) + Slingshot (Isamabard@Bristol)**
  - **GPU-based**

MVAPICH-PLUS GPU Optimized on Isambard – Intranode PT2PT

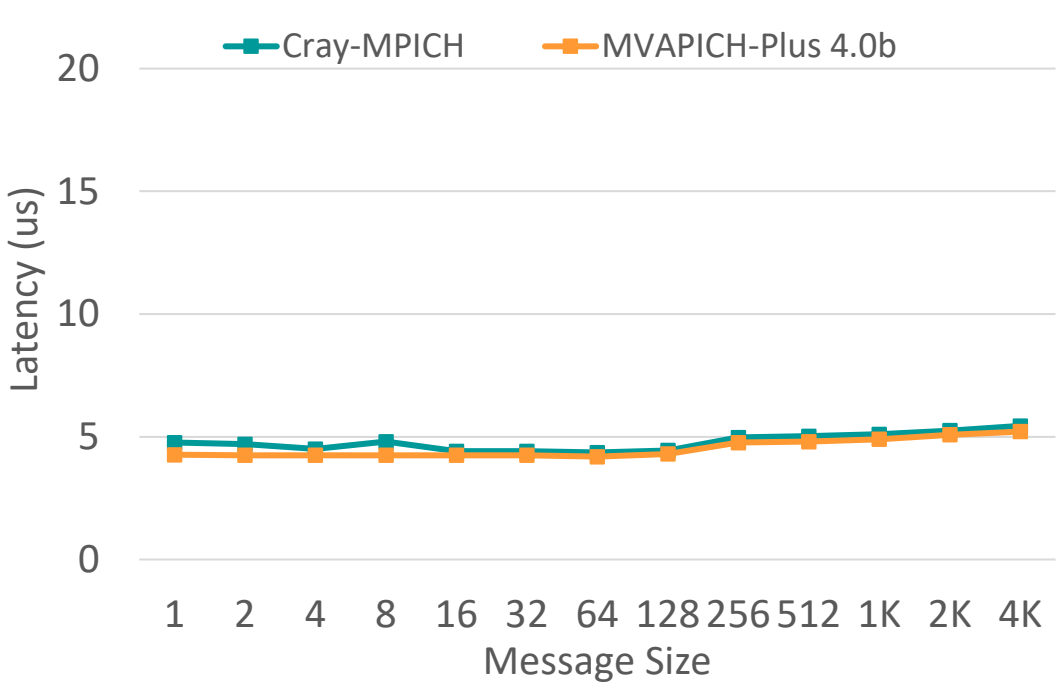


1 node, 2 GPN – Small Message (osu_latency)



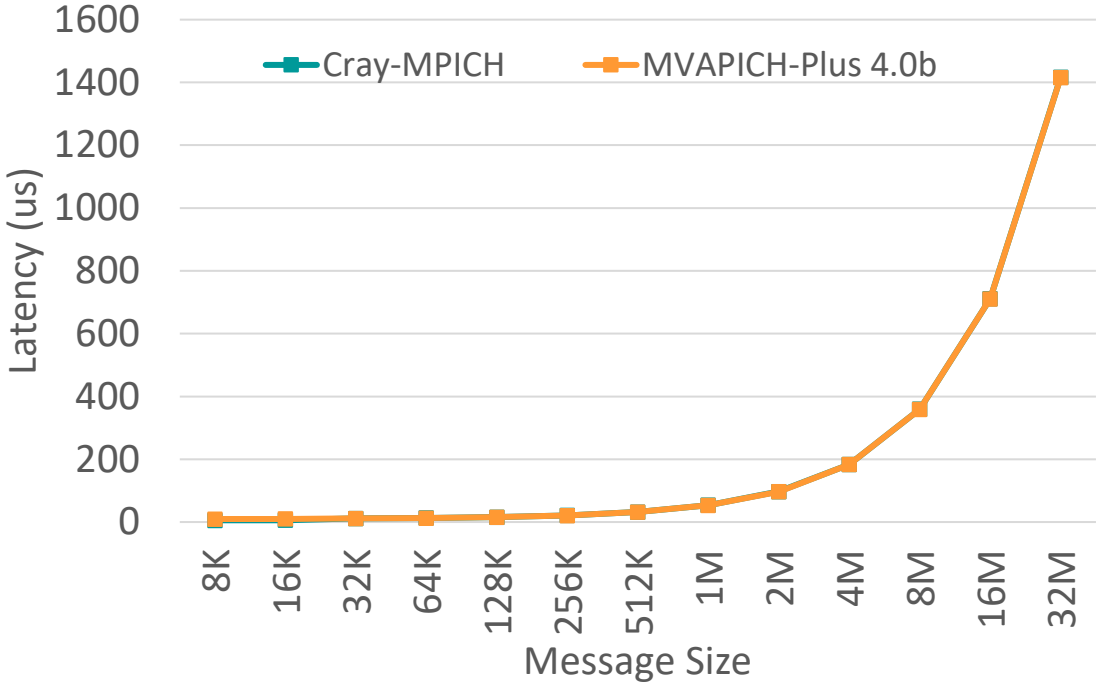
1 nodes, 2 GPN – Large Message (osu_latency)

MVAPICH-PLUS GPU Optimized on Isambard – Internode PT2PT



2 nodes, 1 GPN – Small Message

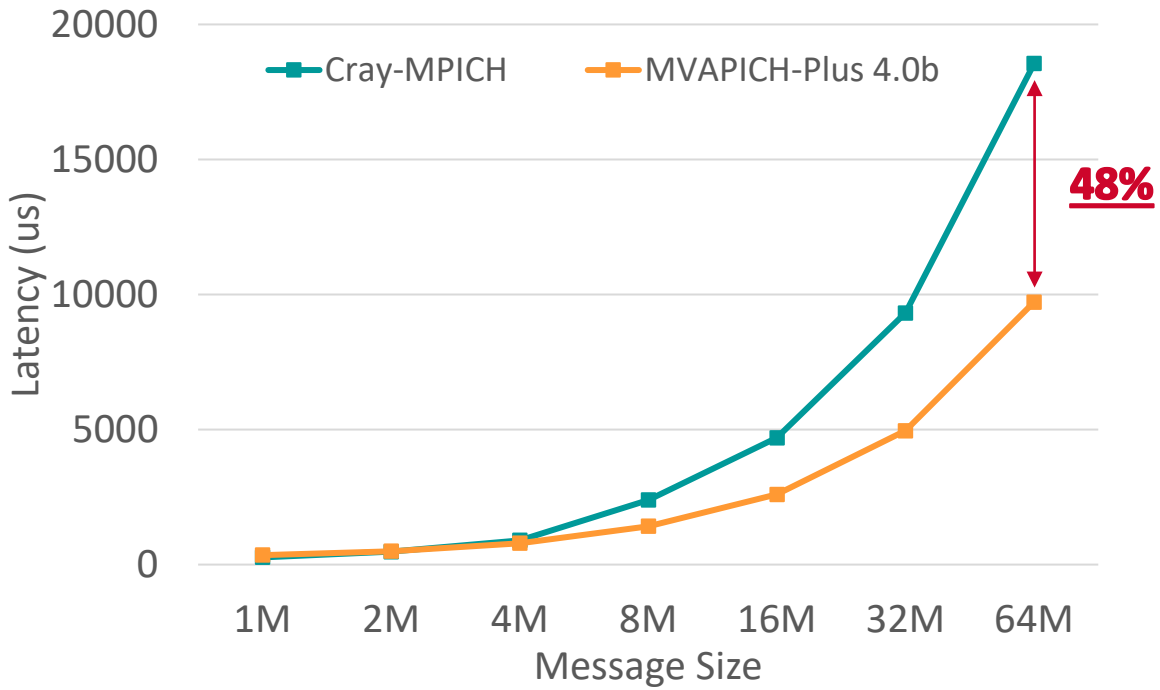
(osu_latency)



2 nodes, 1 GPN – Large Message

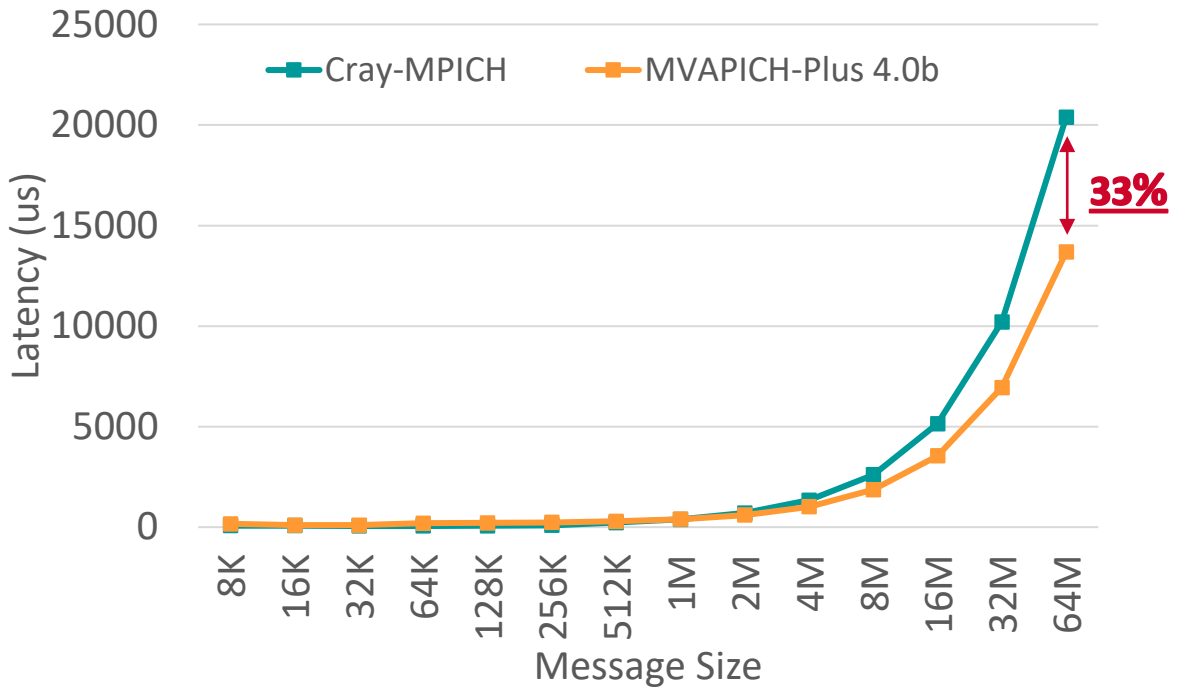
(osu_latency)

MVAPICH-PLUS GPU Optimized on Isambard – Allgather on 2 Nodes



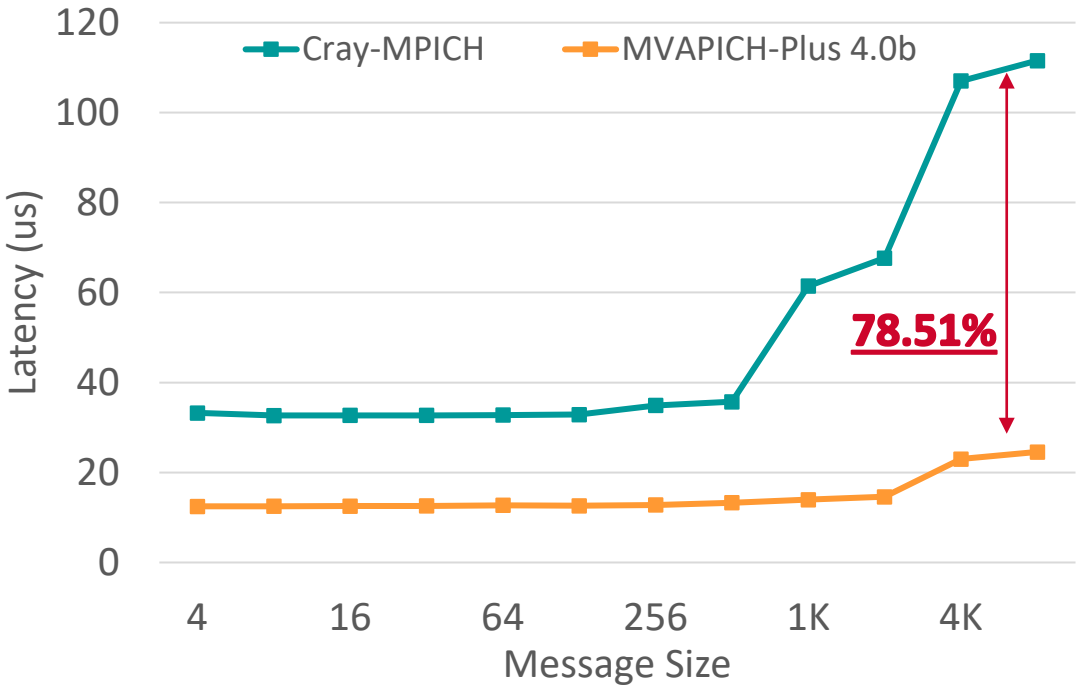
2 nodes, 4 GPN – Large Message

MVAPICH-PLUS GPU Optimized on Isambard – Alltoall on 2 Nodes

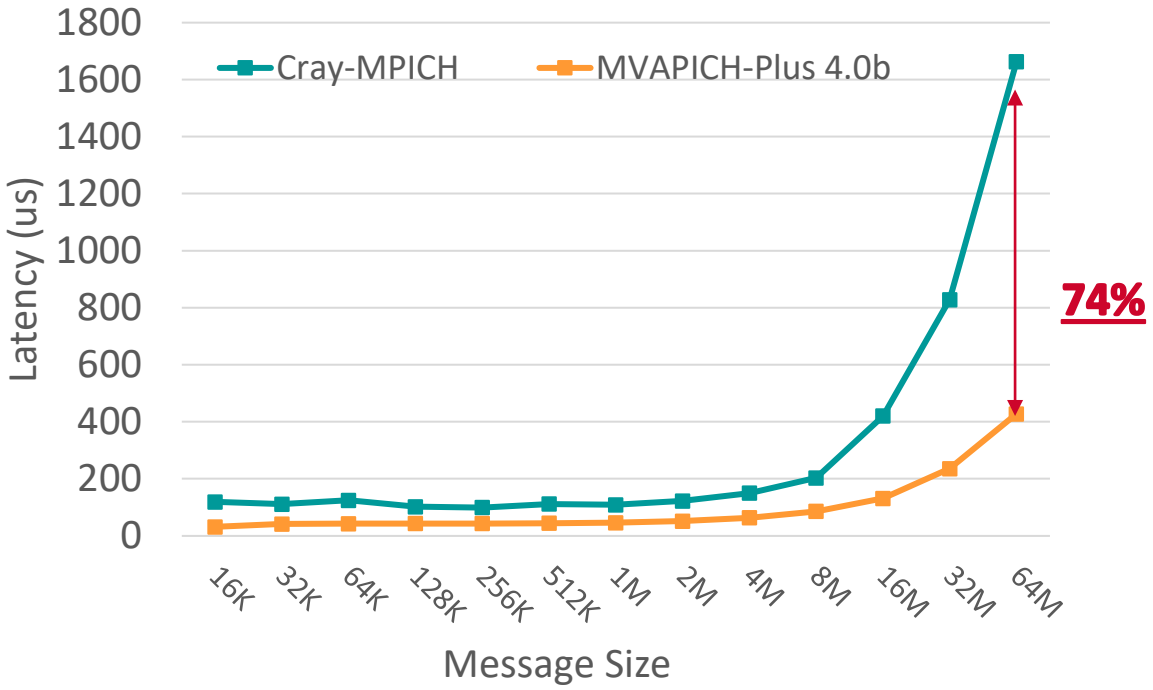


2 nodes, 4 GPN – Large Message

MVAPICH-PLUS GPU Optimized on Isambard – Allreduce on 1 Node



1 node, 4 GPN – Small Message



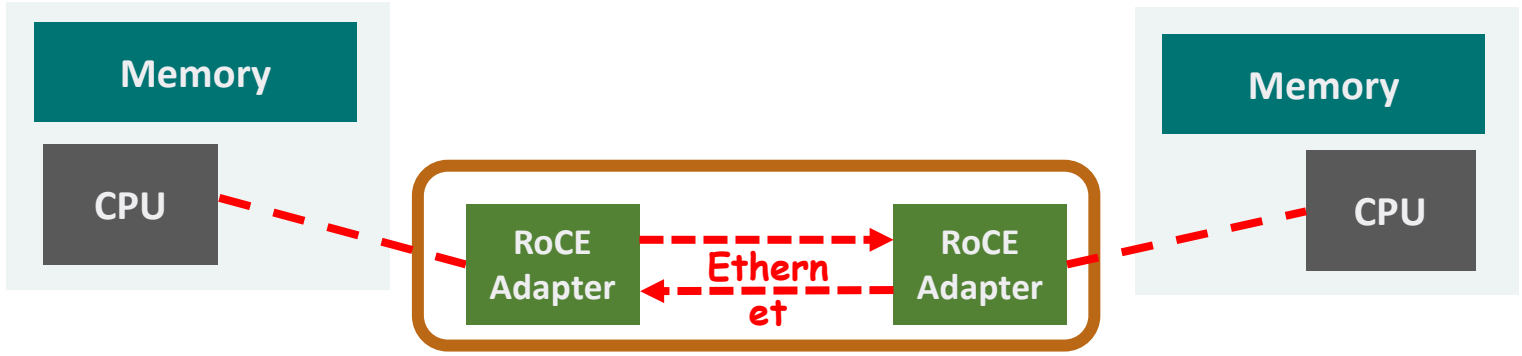
1 node, 4 GPN – Large Message

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

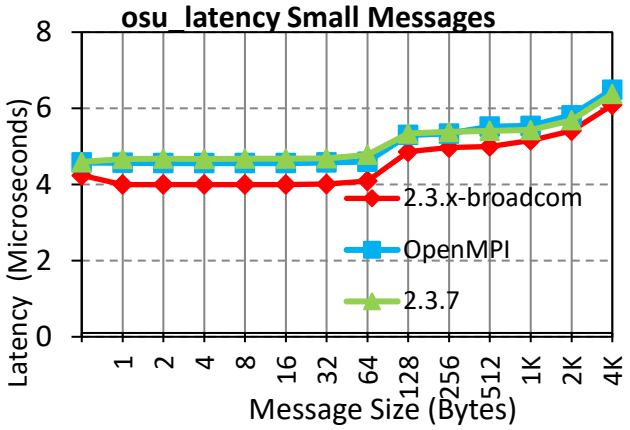
MVAPICH2 Optimization Efforts on Broadcom Thor Adapter

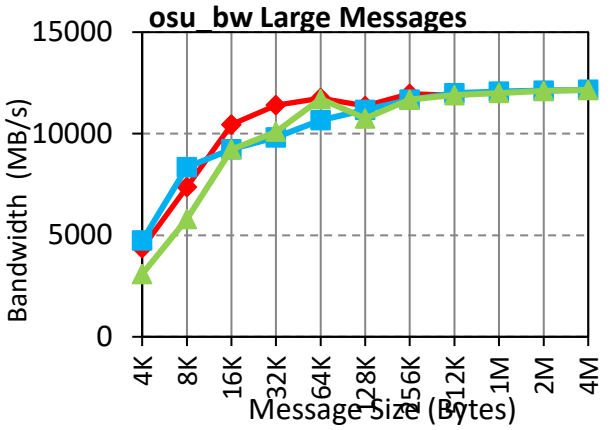
- Add corresponding point-to-point and collective optimization
- Enhance UD+RC hybrid transport protocol mode.
- Optimize the CPU mapping policy
- Support asynchronous threading progress
- Speedup MPI startup time with UD protocol
- Selective message coalescing to improve point-to-point bandwidth and message rate

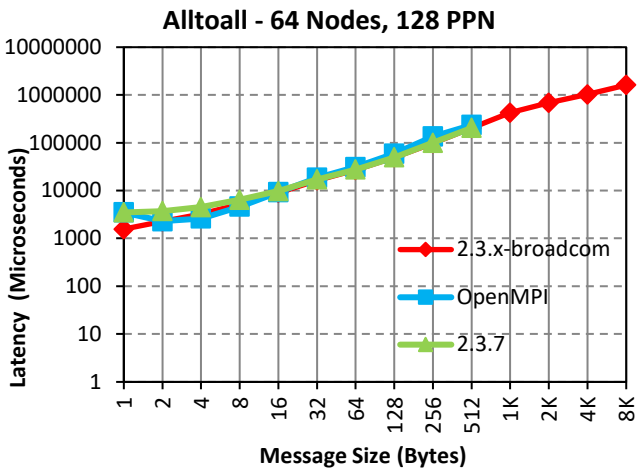


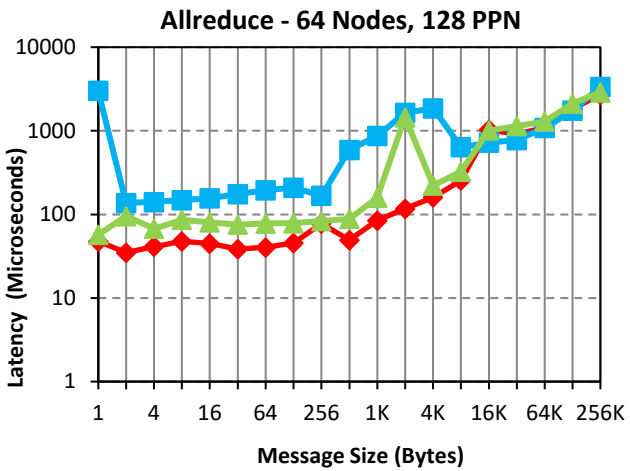
Performance Evaluation on CPU System

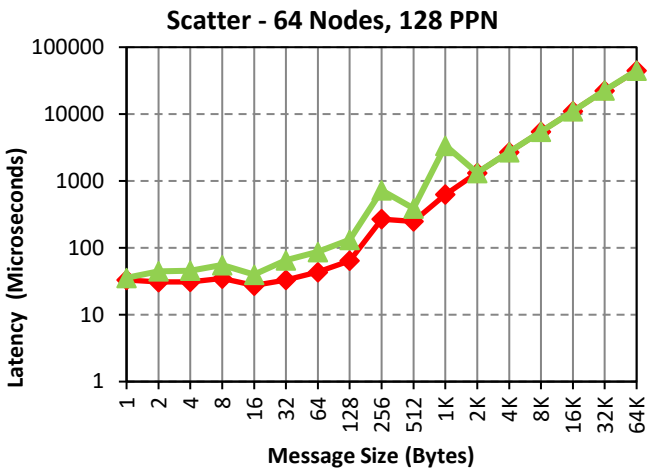
- Experiment results from Dell Bluebonnet
- Up to 20% reduction in small message point-to-point latency
- From 0.1x to 2x increase in bandwidth
- Up to 12.4x lower MPI_Allreduce latency
- Up to 5x lower MPI_Scatter latency





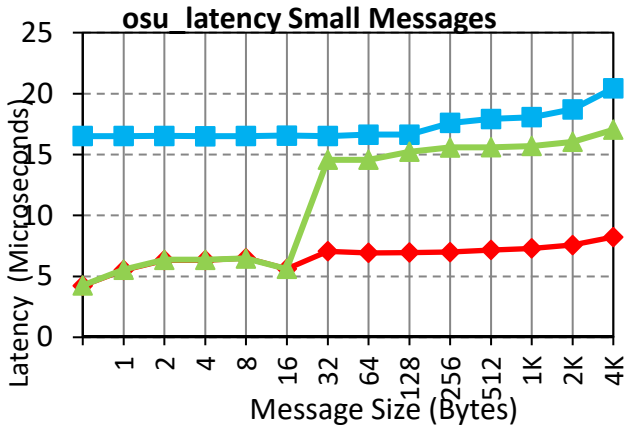


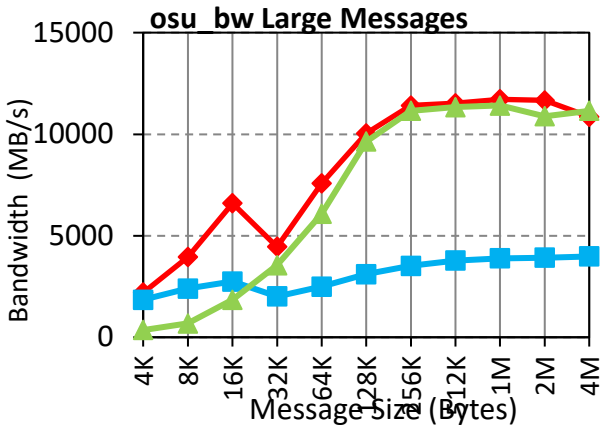


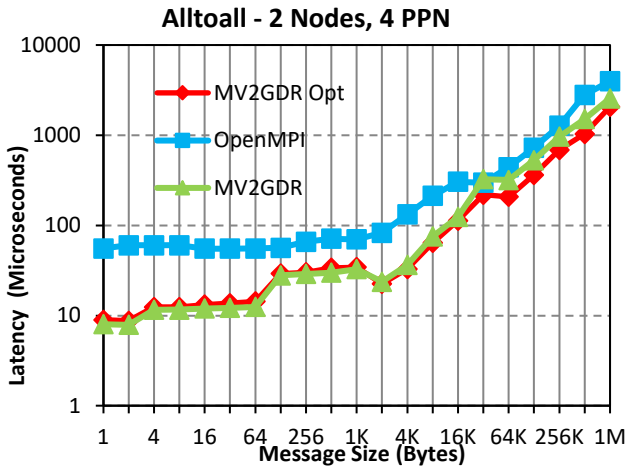


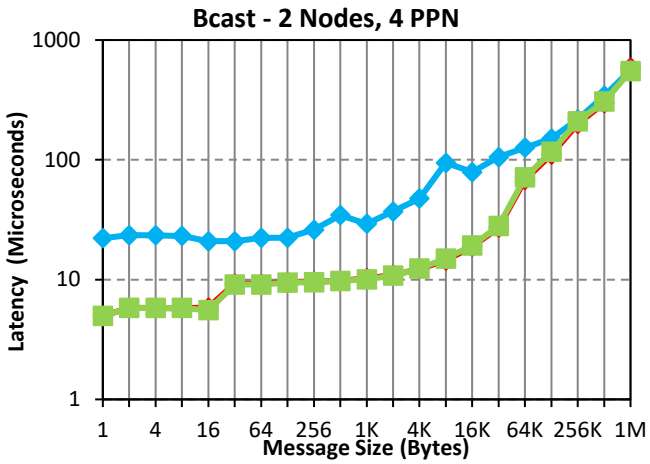
Performance Evaluation on GPU System

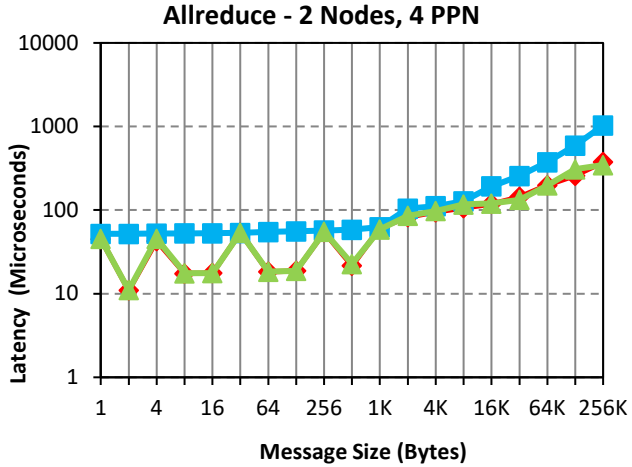
- Experiment results from Rattler2
- Up to 53% reduction in medium message point-to-point latency
- Up to 2.6x increase in bandwidth
- Up to 35% reduction in alltoall latency



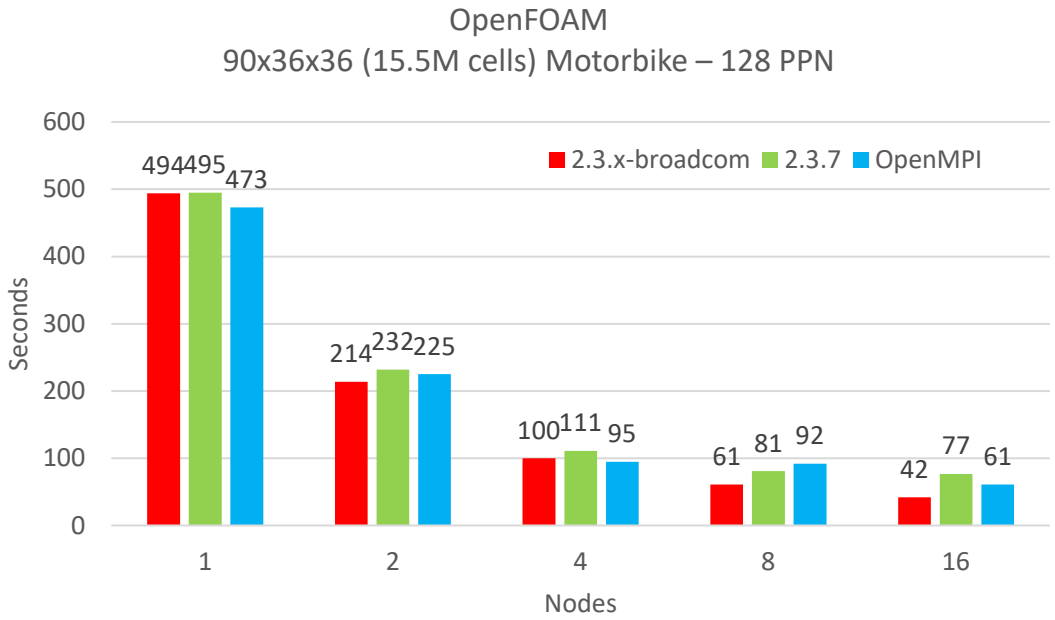


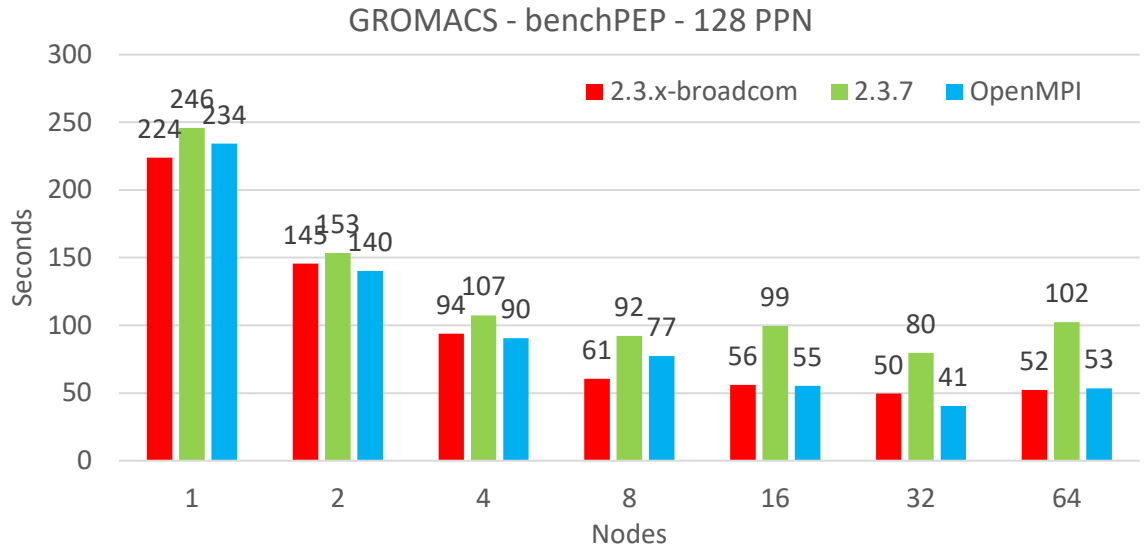


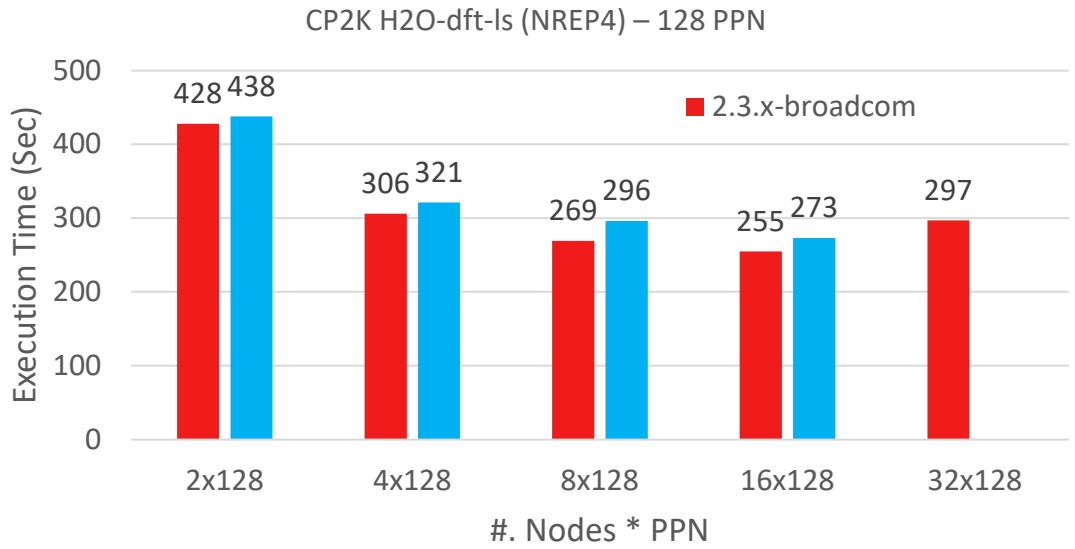




Performance Evaluation – Applications







- Reduce up to 45% execution time of OpenFOAM Motorbike on 16 nodes 128 PPN scale
- Reduce up to 51% execution time of GROMACS benchPEP on 64 nodes 128 PPN scale
- Reduce up to 9.2% execution time of CP2K H2O-dft-ls (NREP4)

Outline

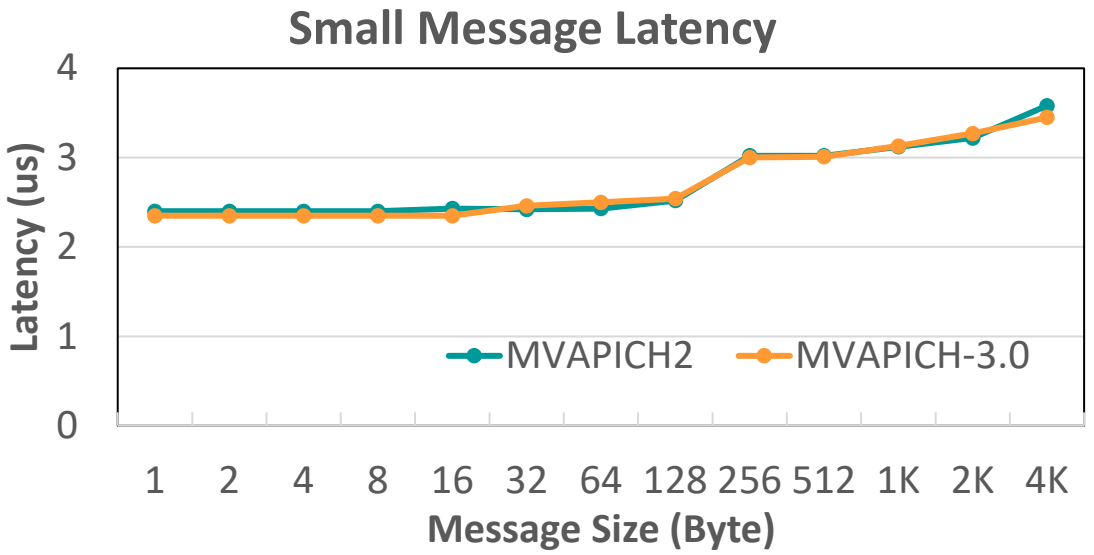
- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - **Optimized versions for Cloud (Azure and AWS)**
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

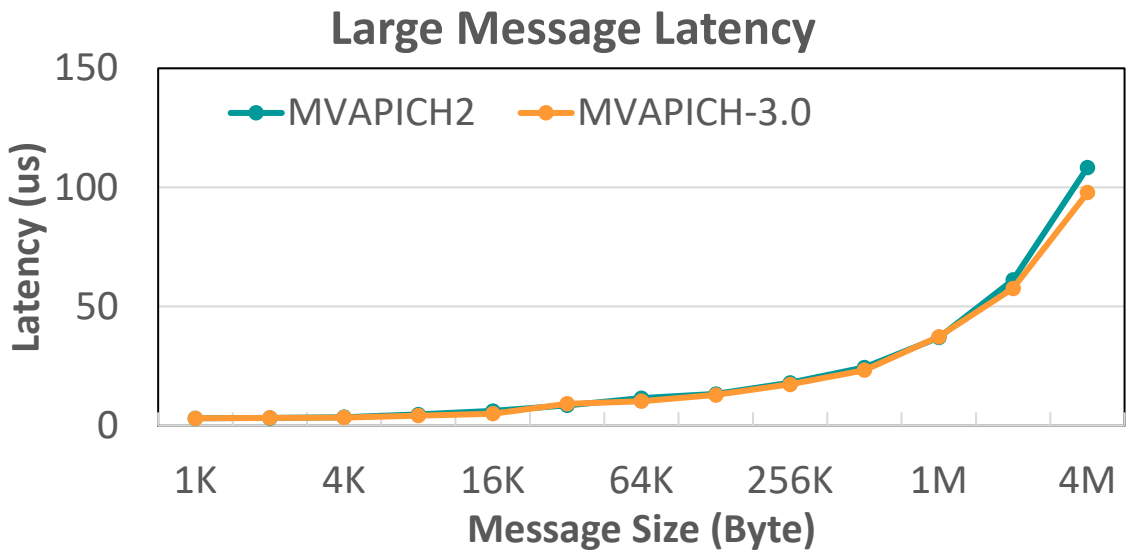
MVAPICH2-Azure Deployment

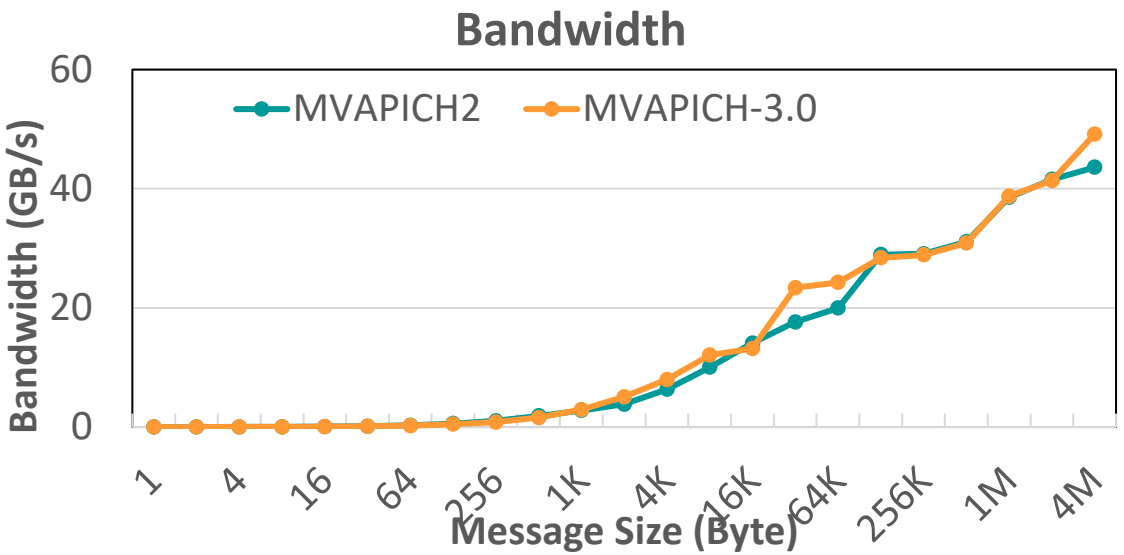
- Azure used InfiniBand for its HPC instances
- All MVAPICH2 (and other libraries like HiBD and HiDL) can work in a transparent manner
- Newly optimized with InfiniBand NDR adapter
- Integrated Azure HPC Images

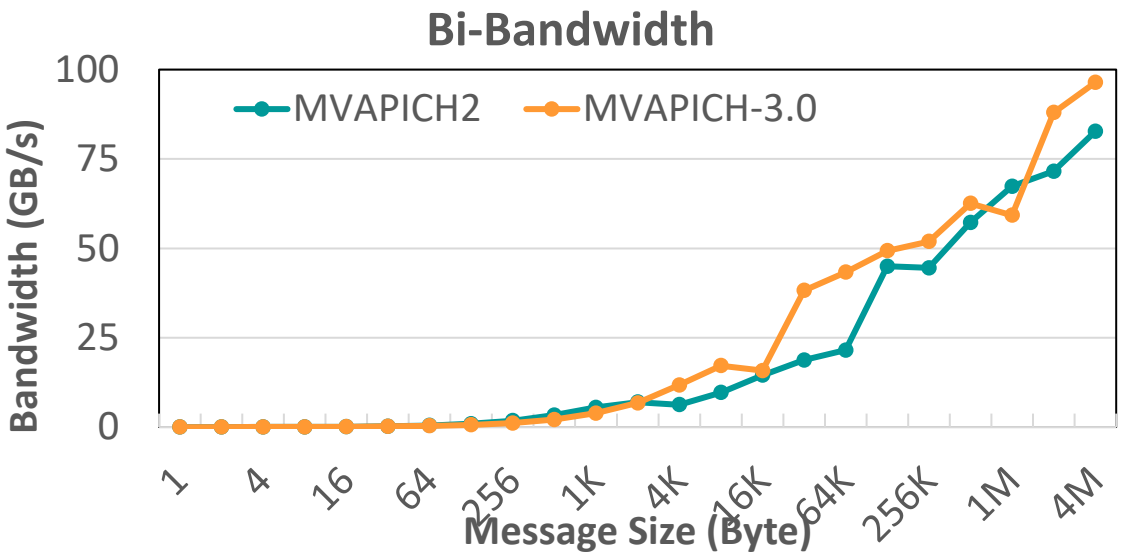
  https://github.com/Azure/azhpc-images/releases/tag/ubuntu-hpc-20240624

MVAPICH Performance on Azure VM with NDR Adapter





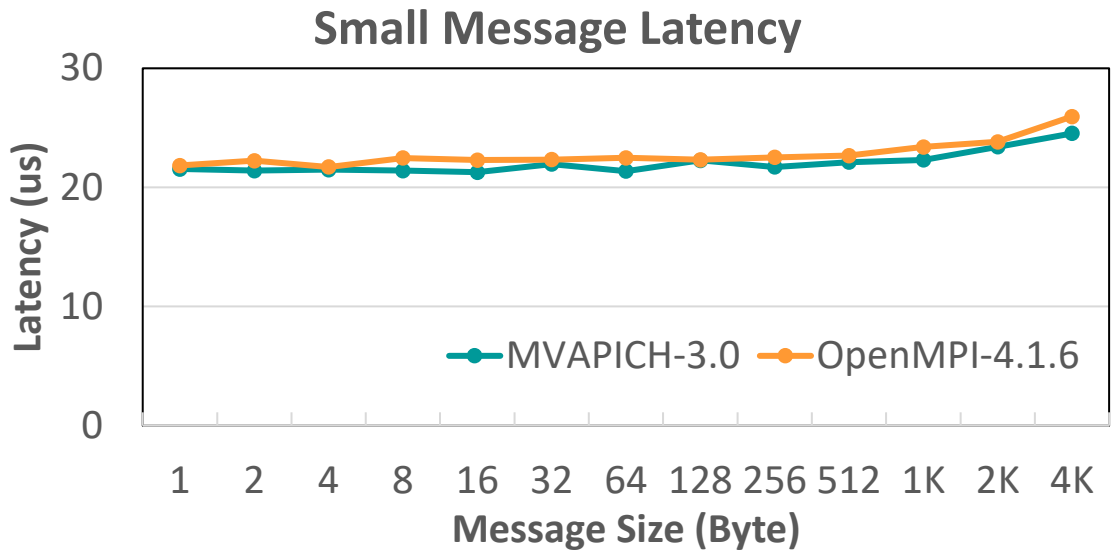


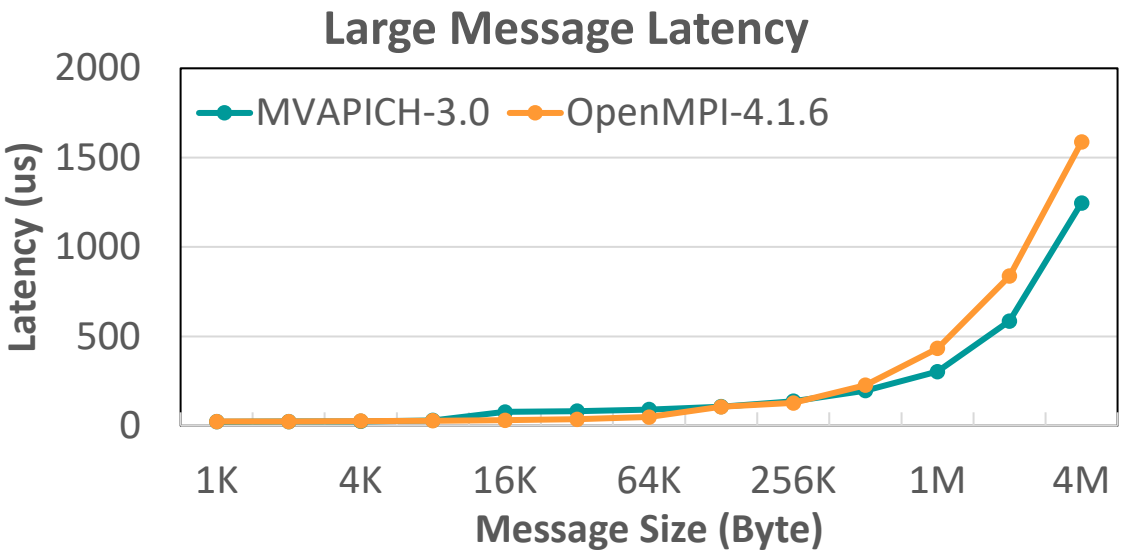


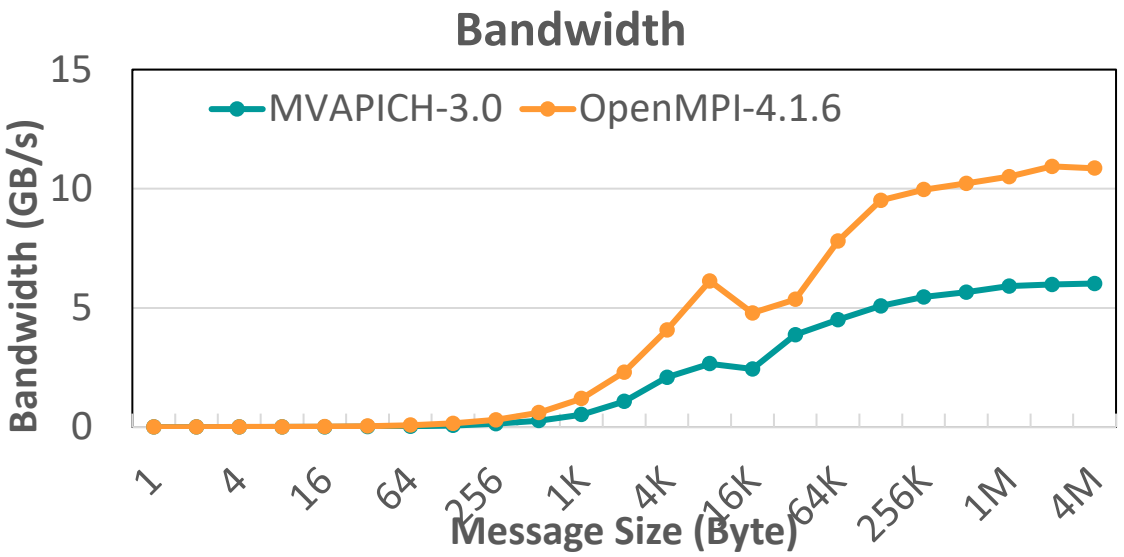
MVAPICH2-AWS Deployment

- AWS uses a proprietary Elastic Fabrics Adapter (EFA)
- Designed an optimized version of the MVAPICH2 library with EFA support
- MVAPICH2-X-AWS 2.3.7 release
- Latest MVAPICH 3.0 and MVAPICH-4.0b with Libfabric support can also work with EFA adapter

MVAPICH-3.0 Performance on AWS HPC Instance with EFA



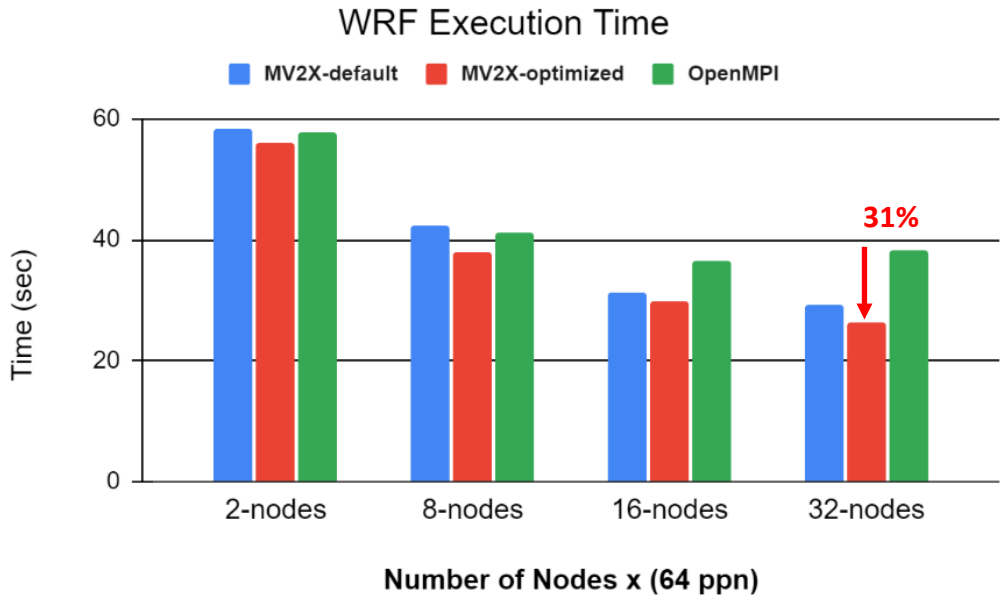


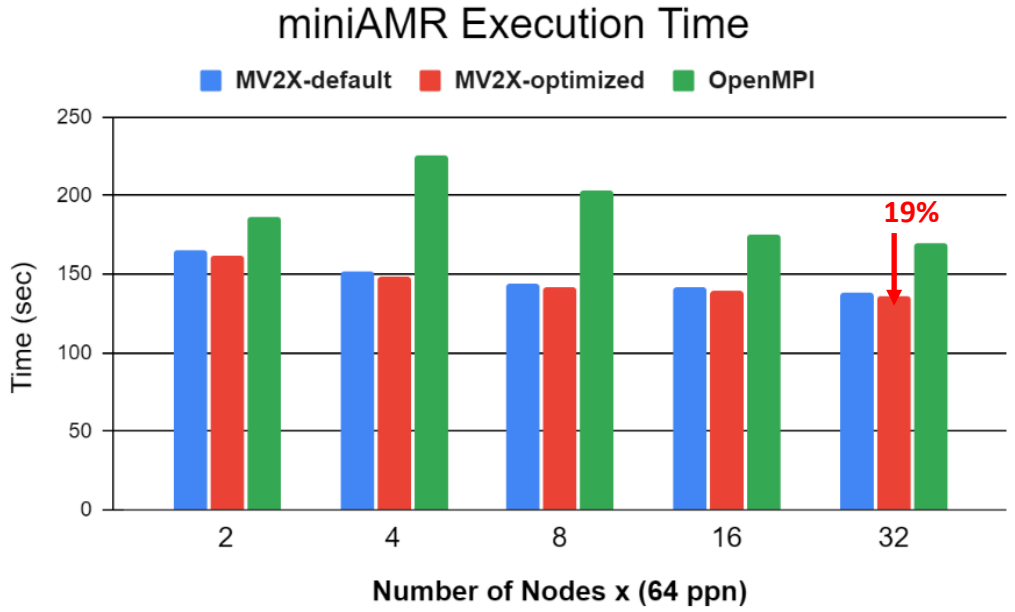


- Tested on c5n.9xlarge Instance with EFA
- Both MVAPICH-3.0 and OpenMPI-4.1.6 build with system default OFI

Application Performance on Arm (Graviton) Instances

- Application-level performance comparison:
  - Weather Research Forecasting Application (WRF) with strong scaling input dataset from 12km resolution case over the Continental U.S. domain
  - miniAMR using default benchmarking input mesh size





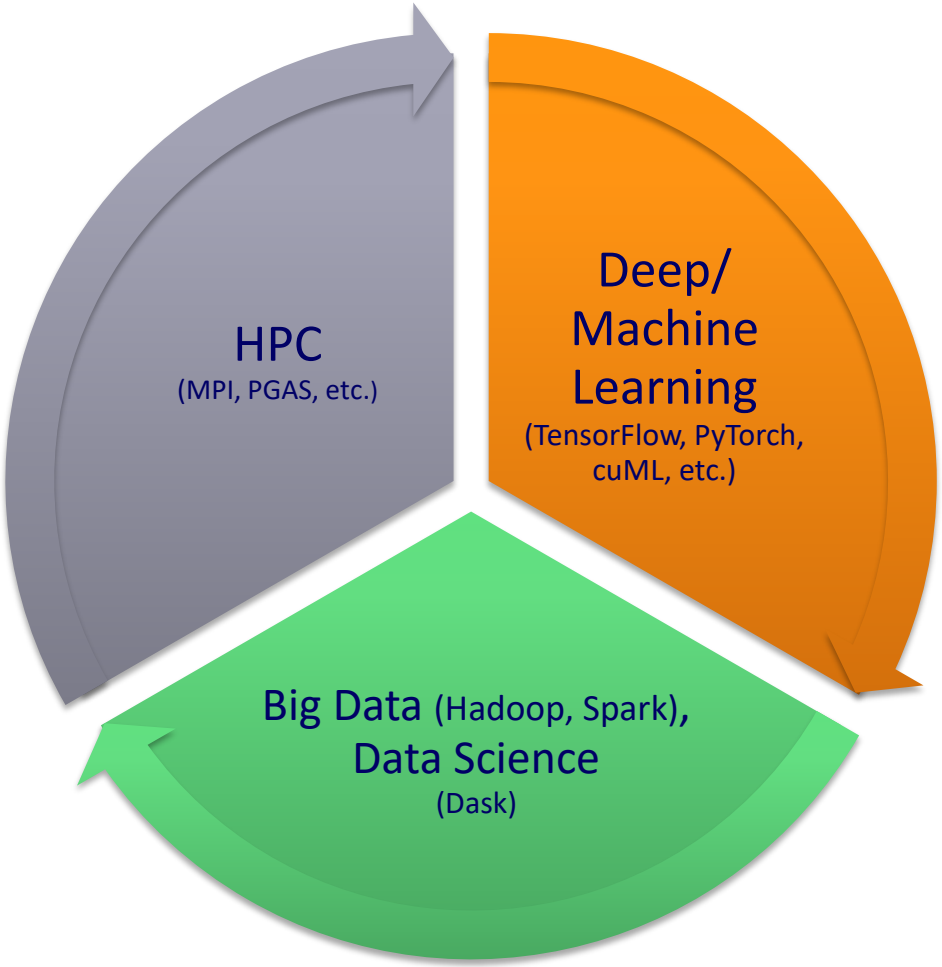
The Weather Research & Forecasting Model, https://www.mmm.ucar.edu/weather-research-and-forecasting-model
Adaptive Mesh Refinement Mini-App, https://github.com/Mantevo/miniAMR

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - **Converged software stack based on MVAPICH-Plus**
    - **Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)**
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

MVAPICH-Driven Converged Software Stack for AI, Big Data and Data Science

- MVAPICH

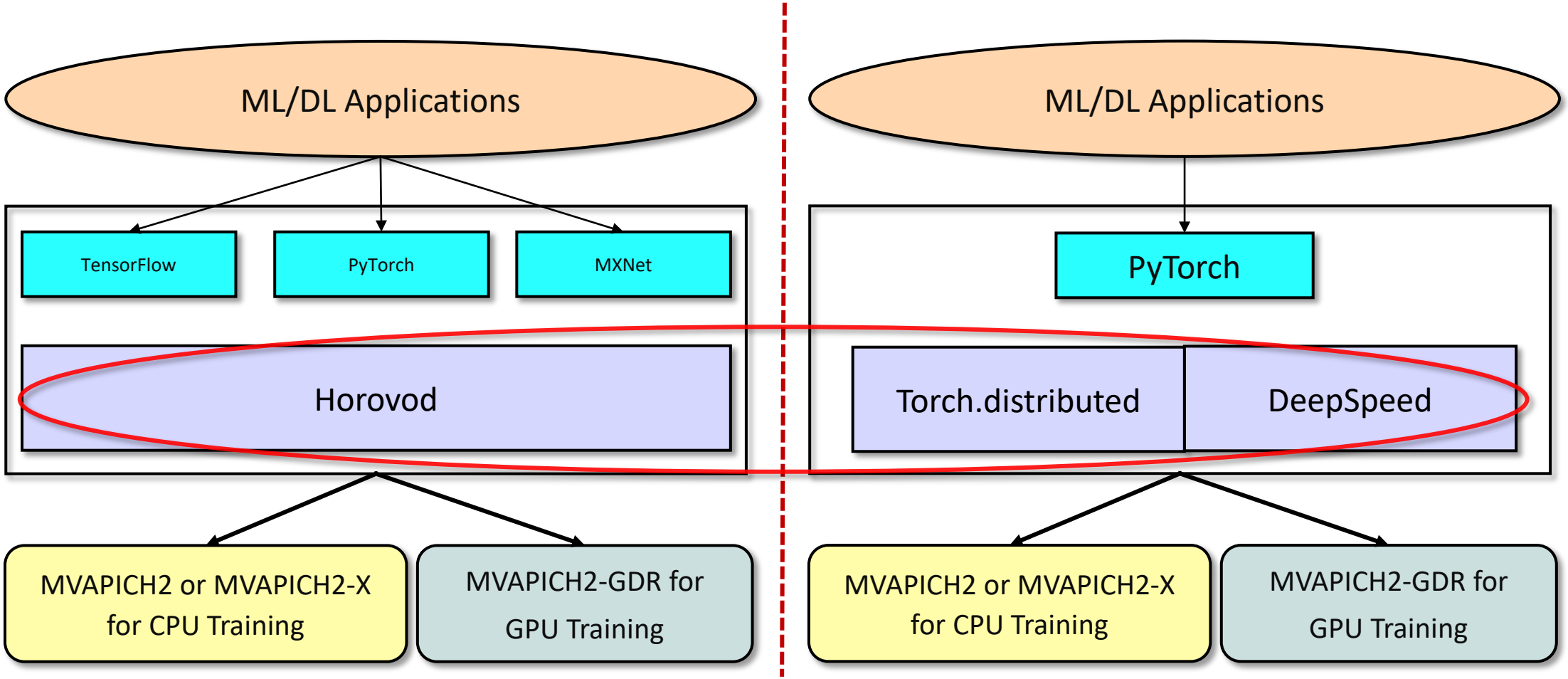


- MPI4DL
- MPI4cuML
- MCR-DL
- ParaInfer-X

Sample Designs and Solutions

- MPI-Driven High-Performance Distributed Training
  - Exploiting Hybrid (Data and Model) Parallelism for out-of-core training
  - Exploiting on-the-fly compression for LLM training
  - Mixed-and-Match Communication Runtime (MCR-DL)
- Accelerating Parallel Inference
  - In-flight Batching and MOE Models
- Accelerating CuML Applications

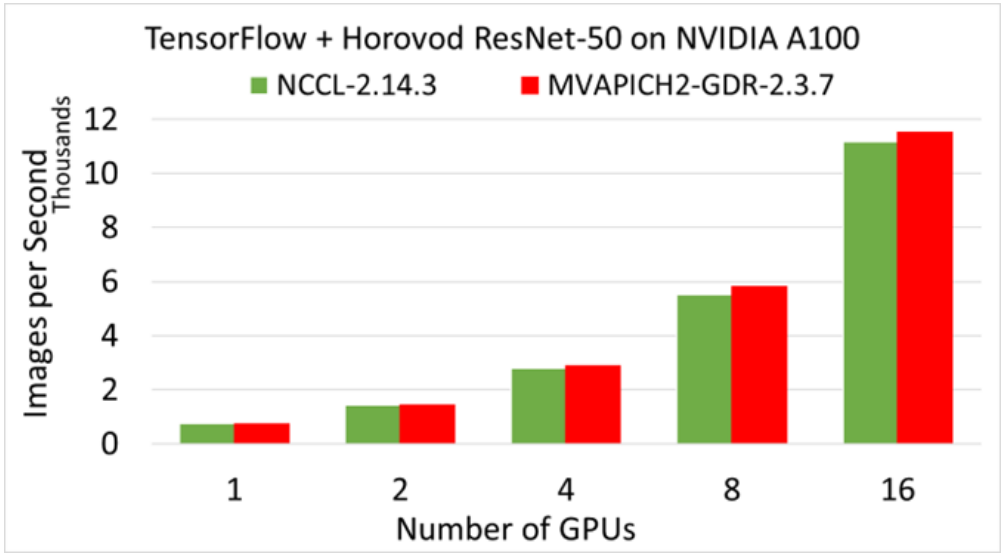
MVAPICH2 (MPI)-driven Infrastructure for ML/DL Training: MPI4DL

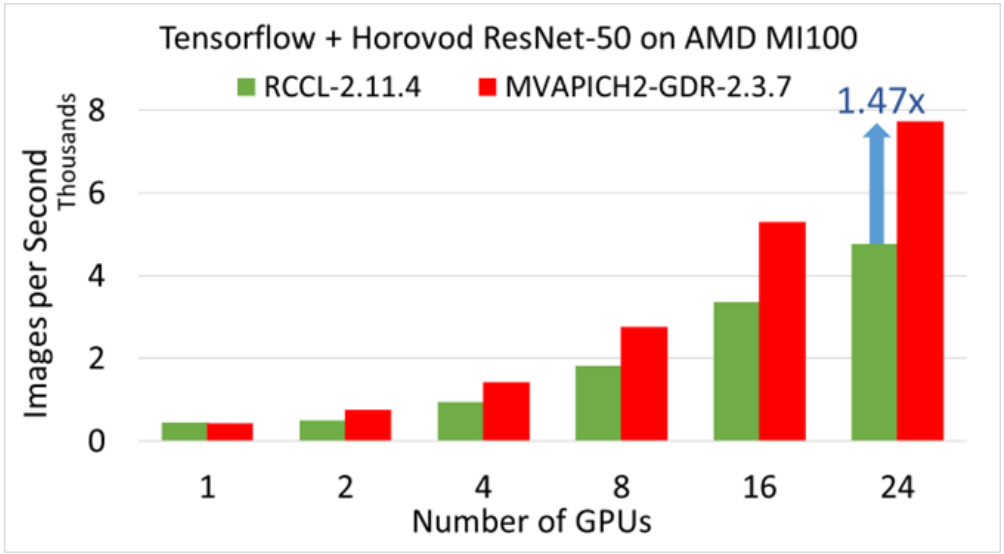


More details available from: https://github.com/OSU-Nowlab/MPI4DL and http://hidl.cse.ohio-state.edu

HiDL Software Stack Release v1.0

- Based on Horovod
- Optimized support for MPI controller in deep learning workloads
- Efficient large-message collectives (e.g. Allreduce) on various CPUs and GPUs
- GPU-Direct algorithms for collective operations (including those commonly used for data- and model-parallelism, e.g. Allgather and Alltoall)
- Support for fork safety
- Exploits efficient large message collectives
- Compatible with
  - Mellanox InfiniBand adapters (EDR, FDR, HDR)
  - Various x86-based multi-core CPUs (AMD and Intel)
  - NVIDIA A100, V100, P100, Quadro RTX 5000 GPUs
  - CUDA [9.x, 10.x, 11.x] and cuDNN [7.5.x, 7.6.x, 8.0.x, 8.2.x, 8.4.x]
  - AMD MI100 GPUs
  - ROCm [5.1.x]

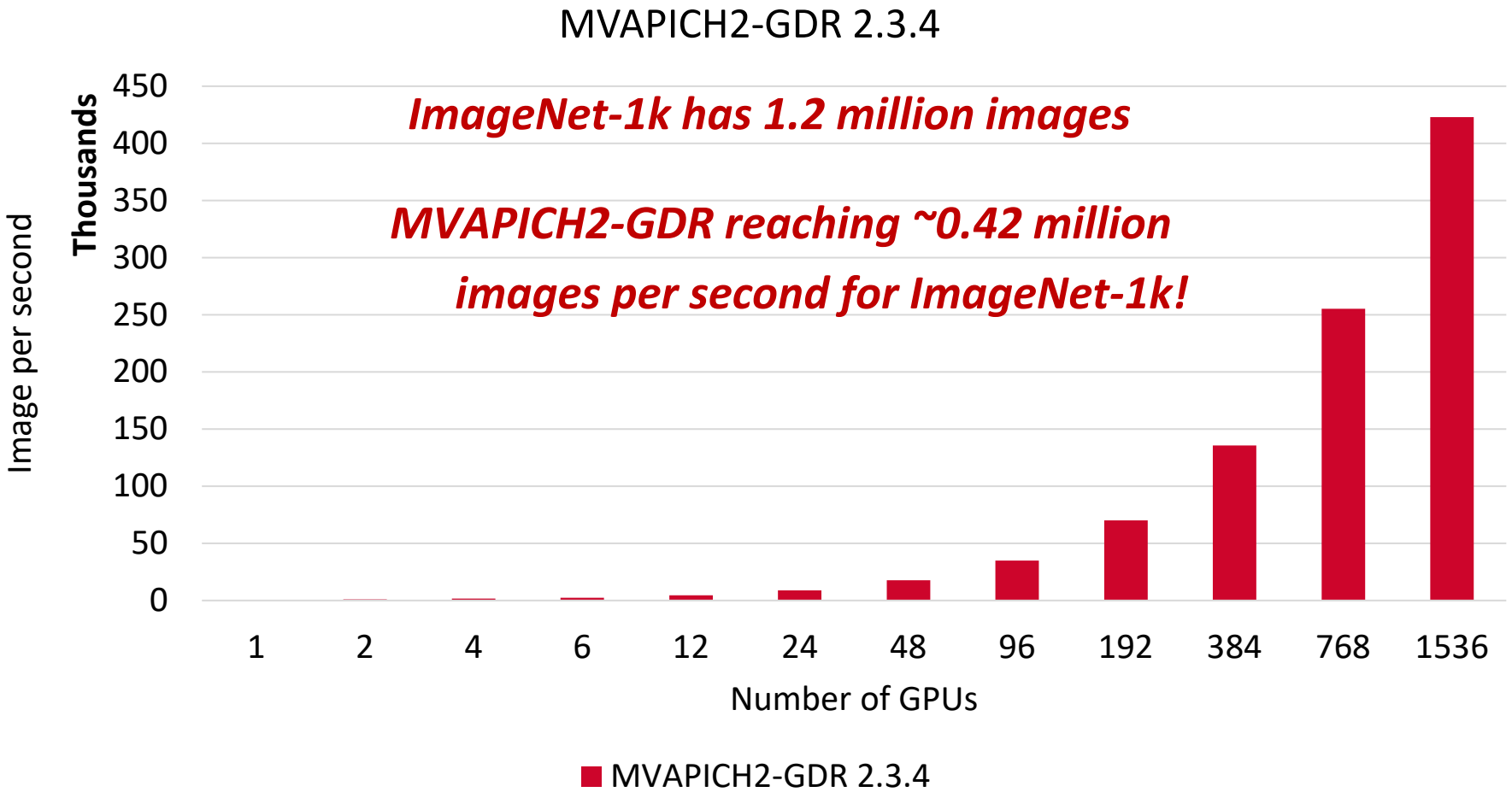




For more details: http://hidl.cse.ohio-state.edu/userguide/horovod/

Distributed TensorFlow on ORNL Summit (1,536 GPUs)

- ResNet-50 Training using TensorFlow benchmark on SUMMIT -- 1536 Volta GPUs!
- 1,281,167 (1.2 mil.) images
- Time/epoch = 3 seconds
- Total Time (90 epochs) = 3 x 90 = 270 seconds = **4.5 minutes!**

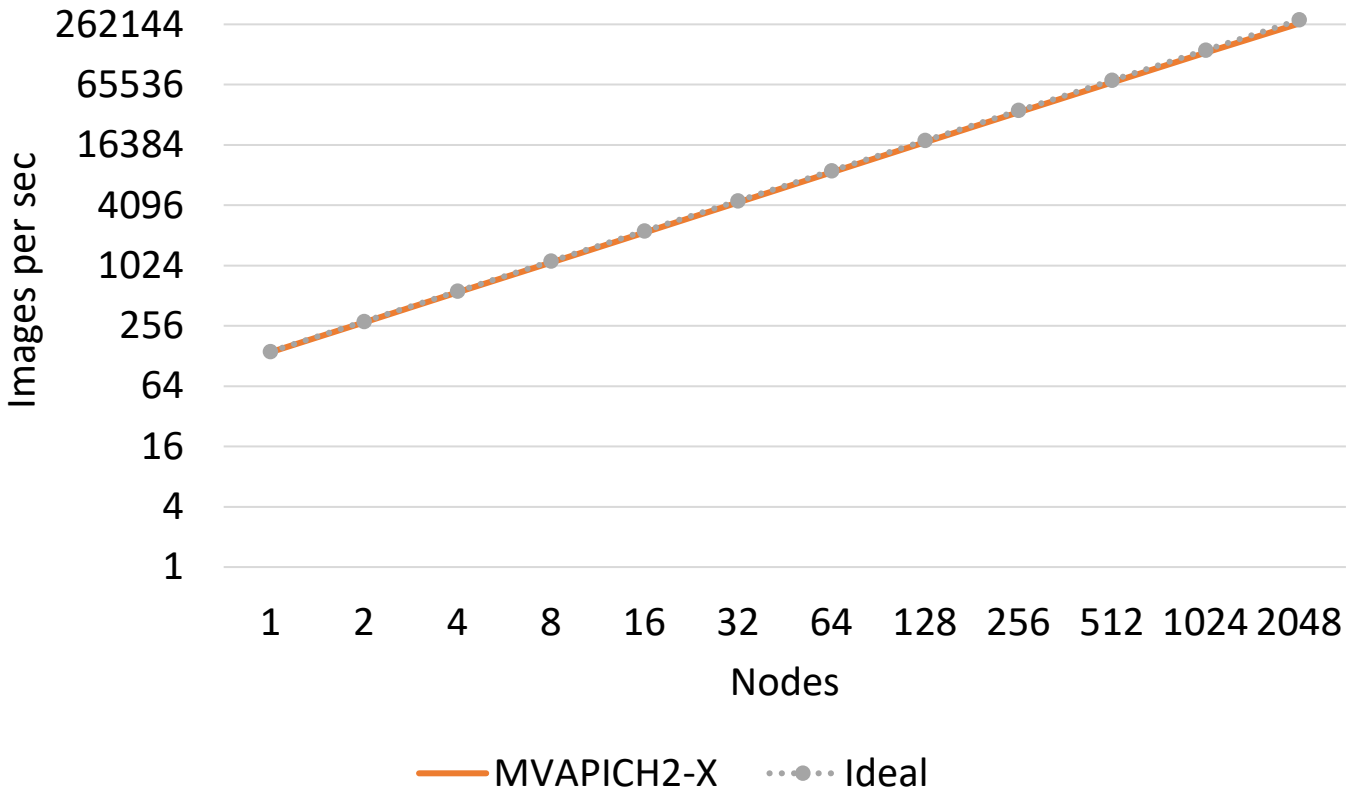


*We observed issues for NCCL2 beyond 384 GPUs

Platform: The Summit Supercomputer (#9 on Top500.org) – 6 NVIDIA Volta GPUs per node connected with NVLink, CUDA 10.1

Distributed TensorFlow on TACC Frontera (2048 CPU nodes)

- Scaled TensorFlow to 2048 nodes on Frontera using MVAPICH2
- MVAPICH2 delivers close to the ideal performance for DNN training
- Report a peak of 260,000 images/sec on 2048 nodes
- On 2048 nodes, ResNet-50 can be trained in 7 minutes!

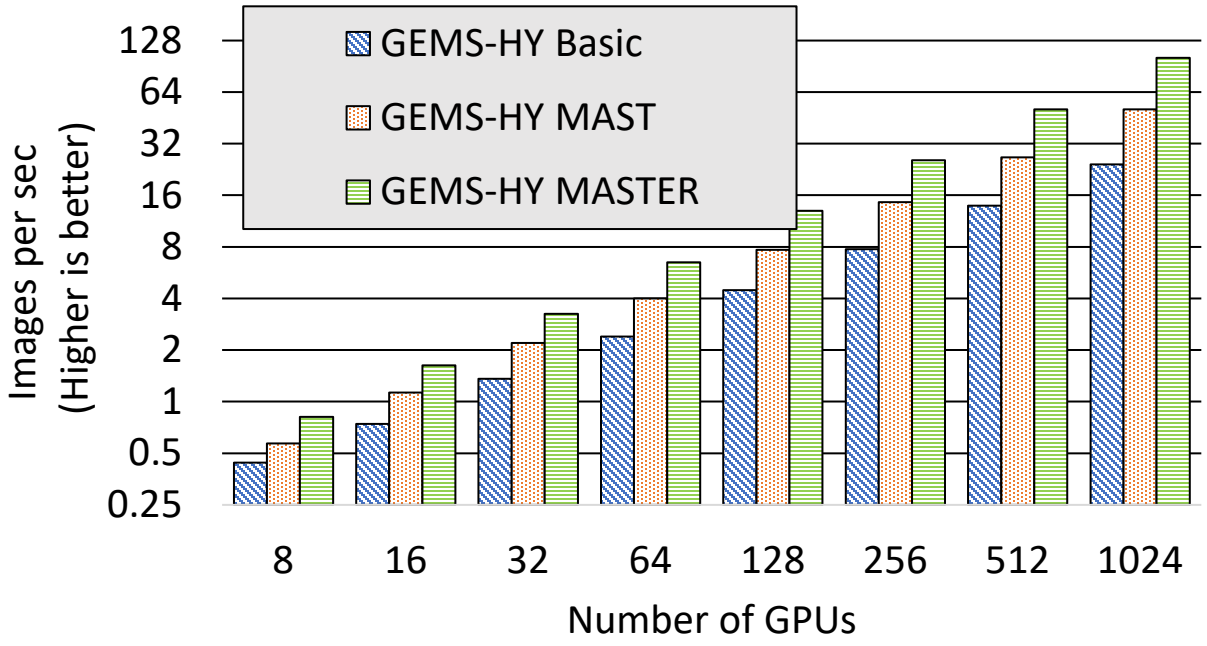


A. Jain, A. A. Awan, H. Subramoni, DK Panda, "Scaling TensorFlow, PyTorch, and MXNet using MVAPICH2 for High-Performance Deep Learning on Frontera", DLS '19 (SC '19 Workshop).

Model Parallelism (GEMS) at Scale (1,024 V100 GPUs on LLNL Lassen)

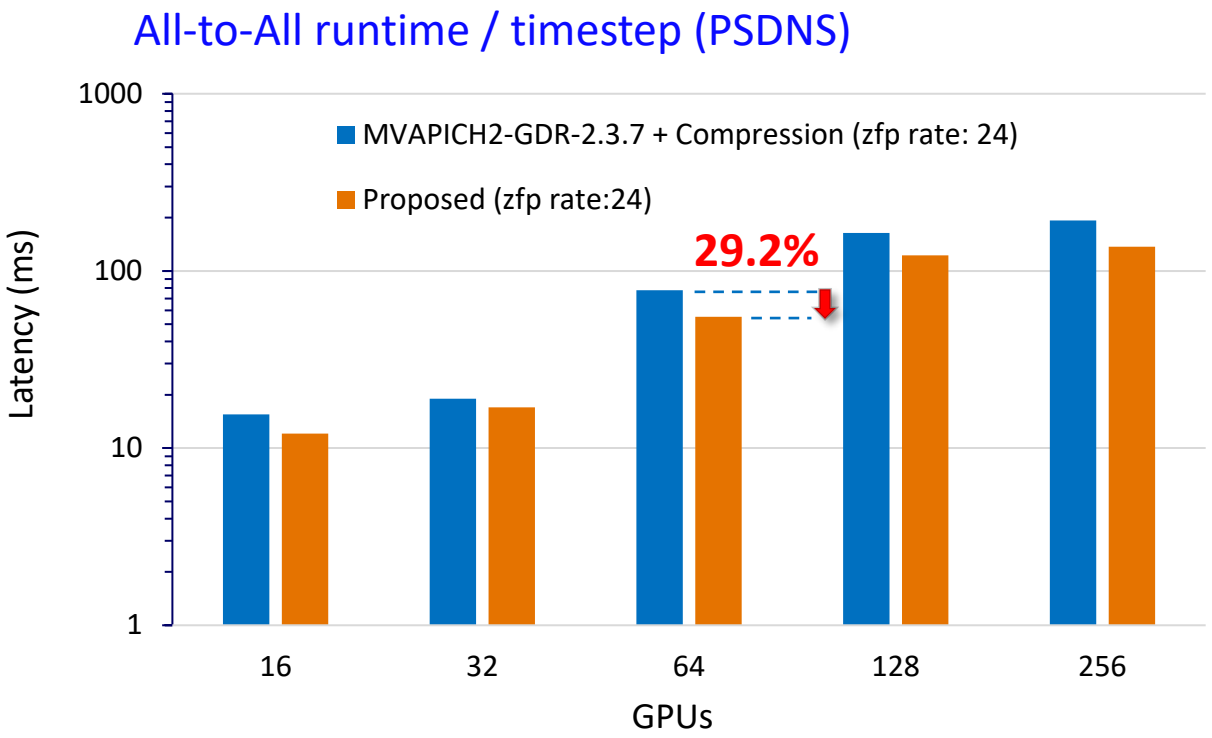
- Two Approaches:
  - Memory Aware Synchronized Training (MAST)
  - Memory Aware Synchronized Training with Enhanced Replications (MASTER)
- Setup
  - ResNet-1k on 512 X 512 images
  - 128 Replications on 1024 GPUs
- Scaling Efficiency
  - **97.32%** on 1024 nodes

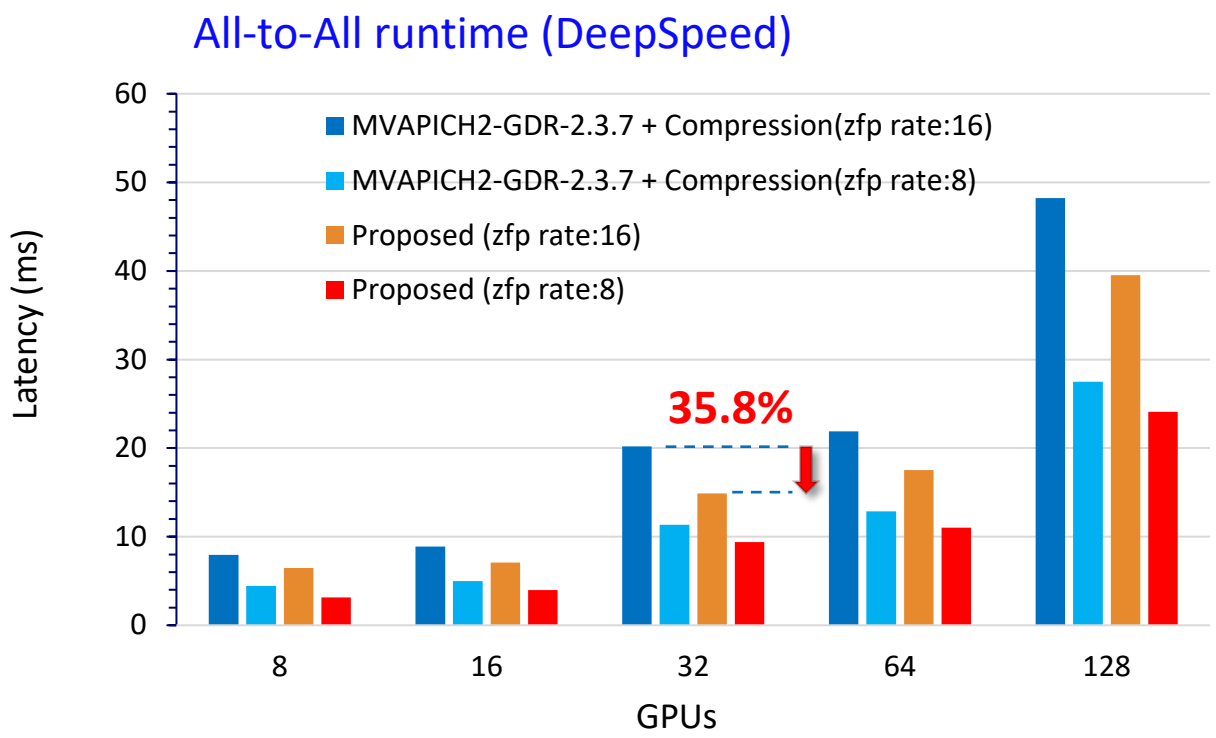
97.32% scaling efficiency on 1024 V100 GPUs (LLNL Lassen)



A. Jain, A. Awan, A. Aljuhani, J. Hashmi, Q. Anthony, H. Subramoni, D. Panda, R. Machiraju, A. Parwani, "GEMS: GPU Enabled Memory Aware Model Parallelism System for Distributed DNN", SC '20

Performance of All-to-All with Online Compression



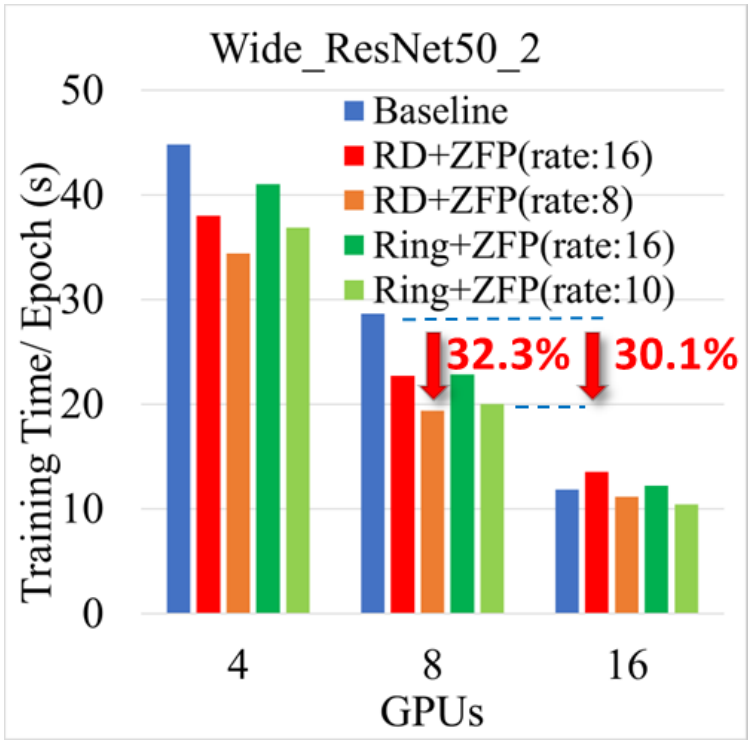


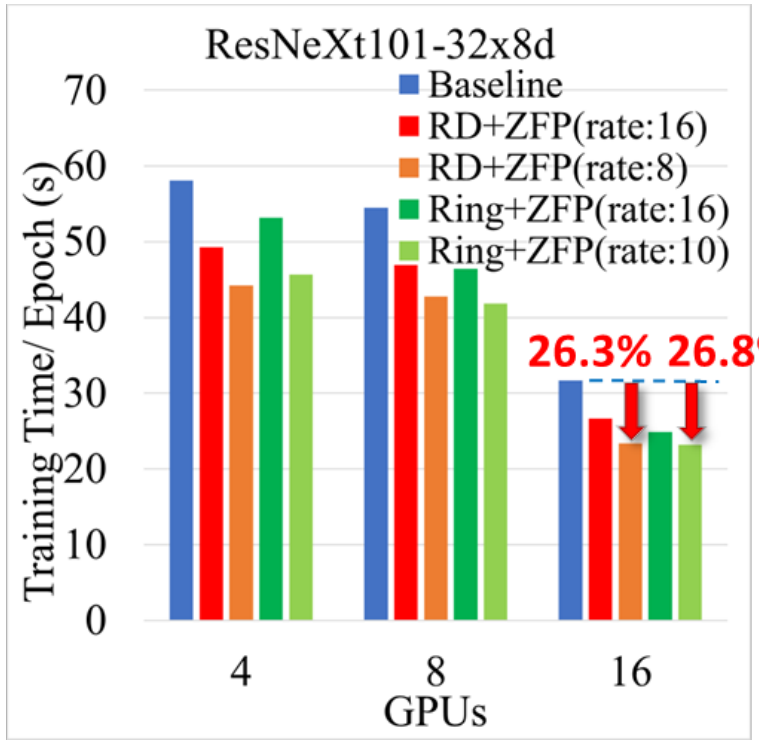
- Improvement compared to MVAPICH2-GDR-2.3.7 with Point-to-Point compression
  - 3D-FFT: Reduce All-to-All runtime by up to 29.2% with ZFP(rate: 24) on 64 GPUs
  - DeepSpeed benchmark: Reduce All-to-All runtime by up to 35.8% with ZFP(rate: 16) on 32 GPUs

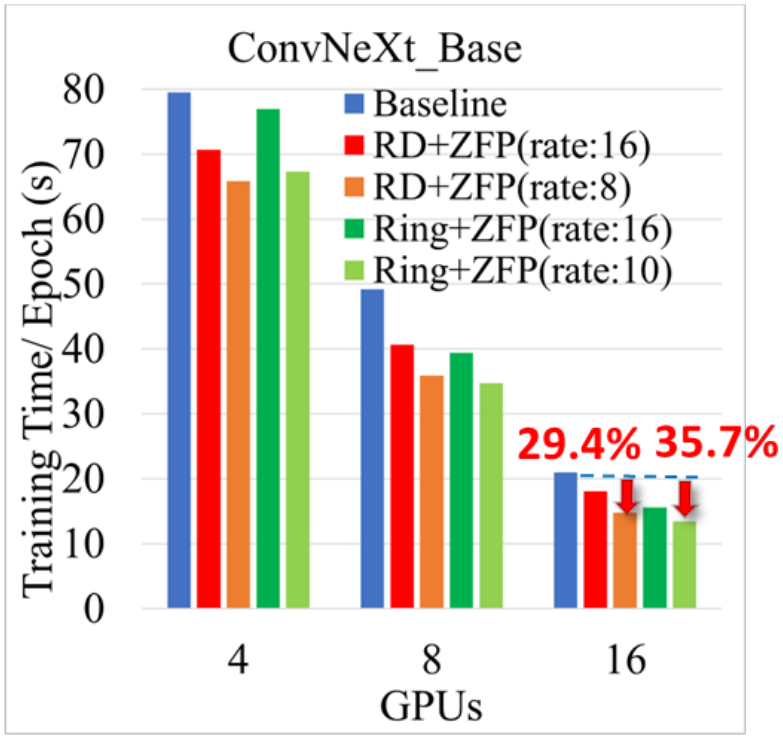
Q. Zhou, P. Kousha, Q. Anthony, K. Khorassani, A. Shafi, H. Subramoni, and D.K. Panda, "Accelerating MPI All-to-All Communication with Online Compression on Modern GPU Clusters", ISC '22.

Available in MVAPICH2-GDR 2.3.7

Performance of AllReduce with Online Compression







Cluster: Pitzer (V100 GPUs), Dataset: CIFAR10, Batch Size=128, Learning Rate=0.001

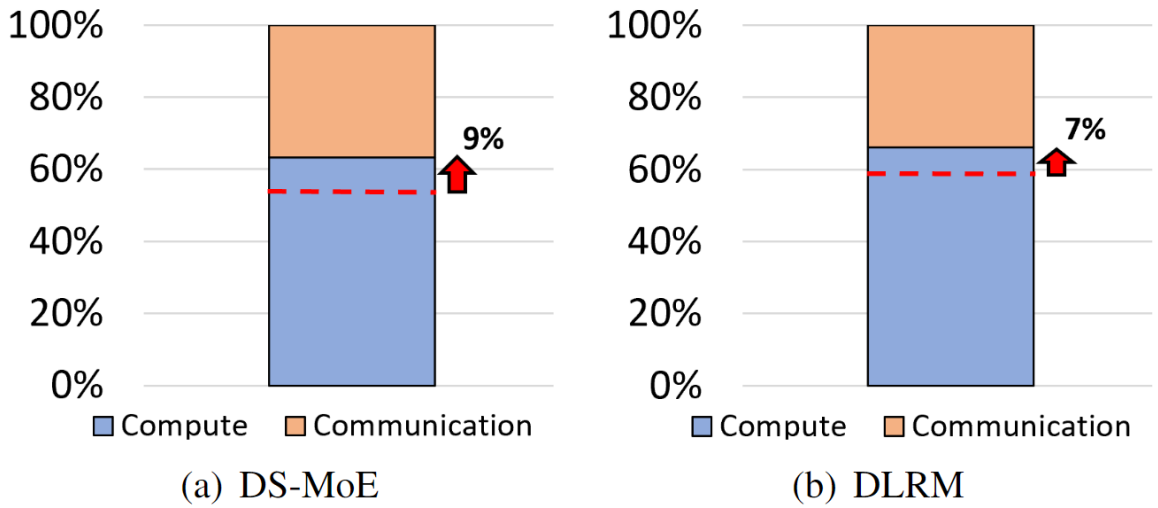
- Improvement for DDP training using PyTorch
  - Use MPI backend with proposed Ring and Recursive-Doubling(RD) AllReduce with Online compression design
  - Wide_ResNet50_2: Reduces the training time by **30.1%** (Ring+ZFP, 8 GPUs, rate: 10), **32.3%** (RD+ZFP, 8 GPUs, rate: 8)
  - ResNeXt101-32x8d: Reduce the training time by **26.8%** (Ring+ZFP, 16 GPUs, rate: 10), **26.3%** (RD+ZFP, 16 GPUs, rate: 8)
  - ConvNeXt_Base: Reduce the training time by **35.7%** (Ring+ZFP, 16 GPUs, rate: 10), **29.4%** (RD+ZFP, 16 GPUs, rate: 8)

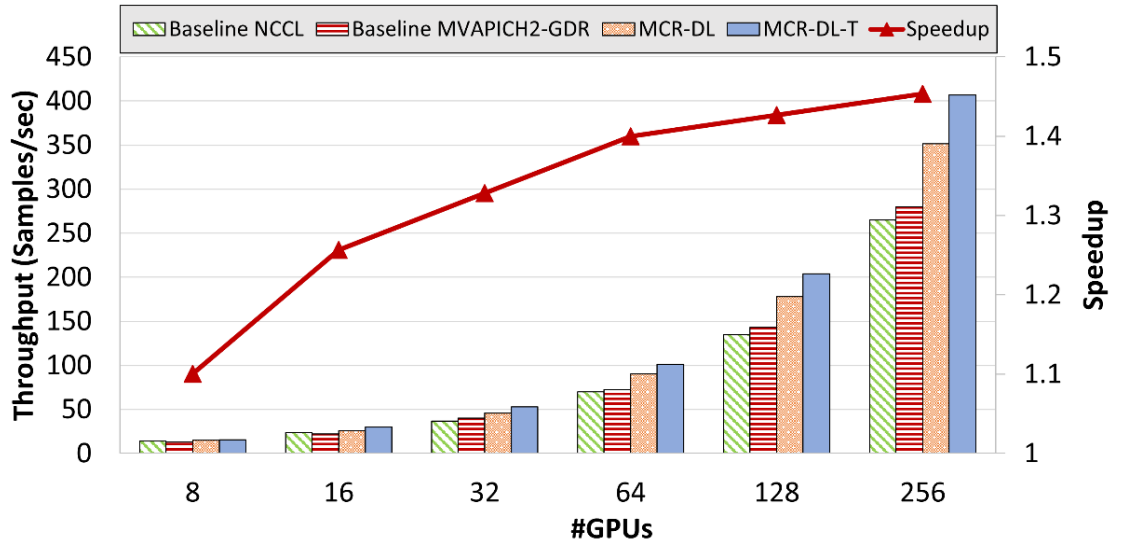
Q. Zhou, B. Ramesh, A. Shafi, M. Abduljabbar, H. Subramoni, and D.K. Panda, "Accelerating MPI AllReduce Communication with Efficient GPU-Based Compression Schemes on Modern GPU Clusters", ISC '24.

Available in MVAPICH-Plus 4.0b

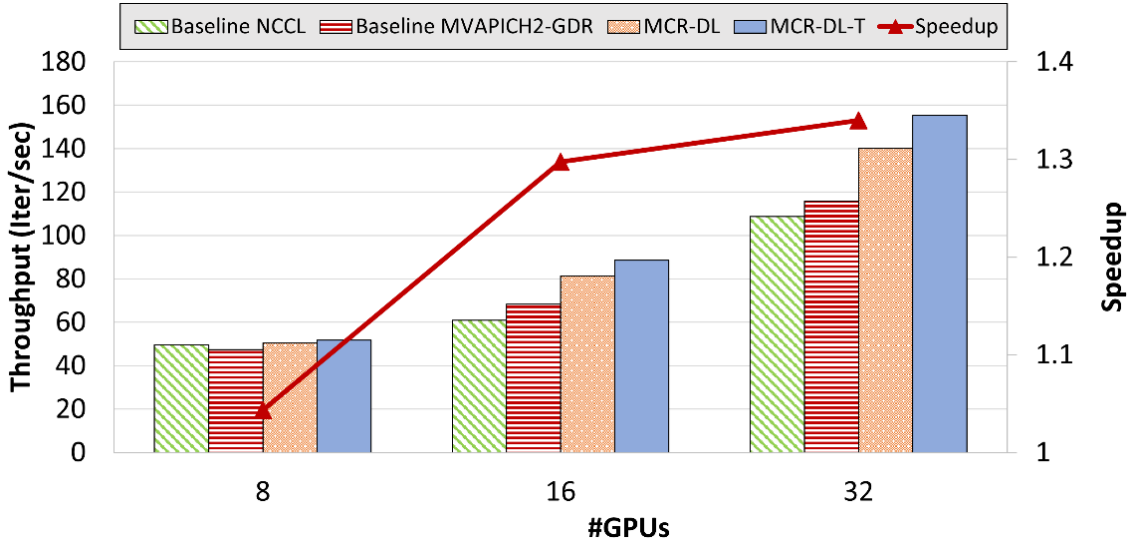
Mix-and-Match Runtime (MCR-DL)

- Benefits for DS-MoE [top] and DLRM [bottom]
- Both primarily use AllReduce and AlltoAll
- Baseline results use a single backend for all operations
- *MCR-DL* coarsely chooses the best backend for each operation based on OMB results; *MCR-DL-T* uses tuning tables to choose the best backend based on the message size
- By using the best communication backends for each individual operation, **MCR-DL-T reduces time spent in communication**





DS-MoE



DLRM

Q. Anthony, A. Awan, J. Rasley, Y. He, A. Shafi, M. Ammar Awan, J. Rasley, Y. He, A. Shafi, M. Abduljabbar, H. Subramoni, and DK Panda, "MCR-DL: Mix-and-Match Communication Runtime for Deep Learning", IPDPS '23

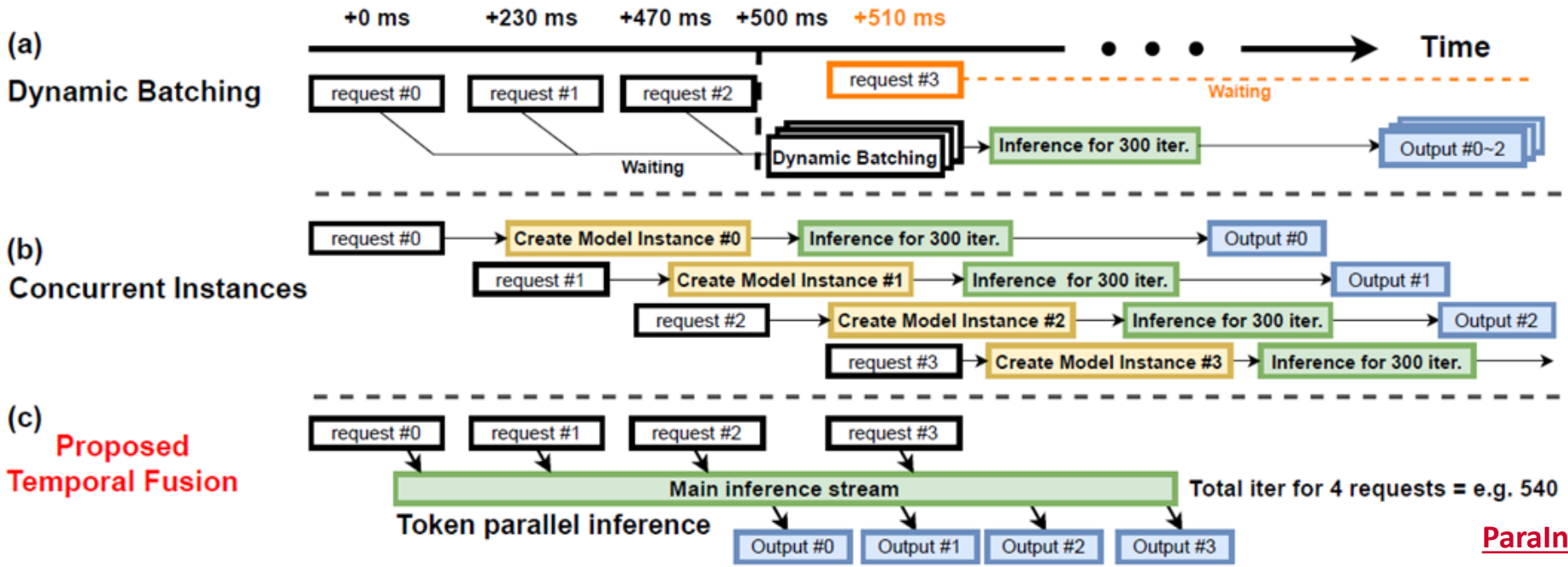
MCR-DL Release- https://github.com/OSU-Nowlab/MCR-DL

Sample Designs and Solutions

- MPI-Driven High-Performance Distributed Training
  - Exploiting Hybrid (Data and Model) Parallelism for out-of-core training
  - Exploiting on-the-fly compression for LLM training
- **Accelerating Parallel Inference**
  - **In-flight Batching and MOE Models**
- Accelerating CuML Applications

Flover: Efficient Parallel Inference on LLMs with Temporal Fusion[1]

- When serving multiple requests, how to deliver both low-latency and high-throughput?
- For generative models such as GPT, LLaMA, the generation is sequential and regulated by 'for' loop.
  - For multiple requests that arrive at different time, how do we schedule the inference?



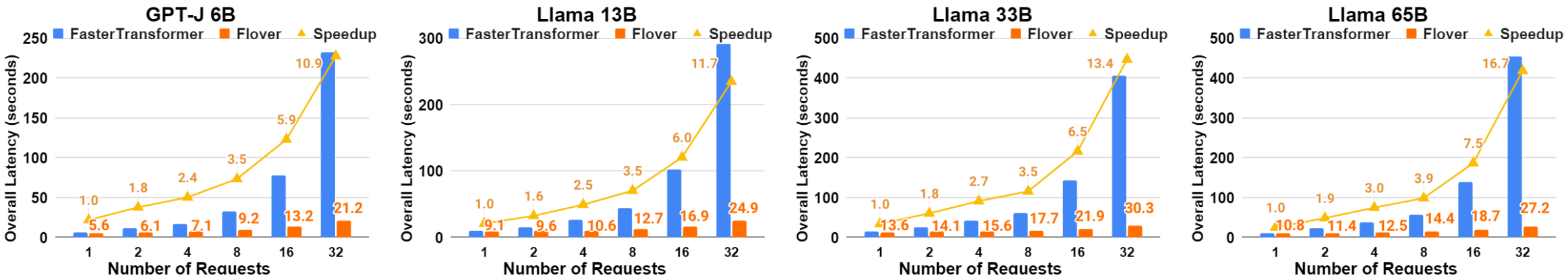
ParaInfer-X Release,
https://github.com/OSU-Nowlab/Flover

- *We leverage the temporal property in generative model to smartly batch token generation.*
  - *Only maintain one persistent inference instance for serving any incoming requests with no delay.*
  - *Efficient memory reordering strategy to assure requests' buffer continuity, avoiding internal fragments.*

[1] Yao, Jinghan, Nawras Alnaasan, Tian Chen, Aamir Shafi, and Hari Subramoni. "Flover: A Temporal Fusion Framework for Efficient Autoregressive Model Parallel Inference." *In Proceeding of HiPC 23*

Overall Latency of Inference Requests

- This experiment analyzes the performance of Flover when processing multiple requests in parallel.
  - In this part, we use a constant time interval of 500ms to study the parallel efficiency.
- Notice that for all models, the average inference latency for a single request is >> 500ms, therefore it leaves great potential for parallel acceleration.
  - For GPT-J 6B and Llama 13B, we run on 1 GPU without tensor parallelism.
  - For Llama 33B, we run on 2 GPUs with tensor parallelism of size 2.
  - For Llama 65B, we use 4 GPUs to perform degree-4 tensor parallelism.
- **Compared to FasterTransformer, our method achieves up to 16.7x speedup in latency to finish all requests.**



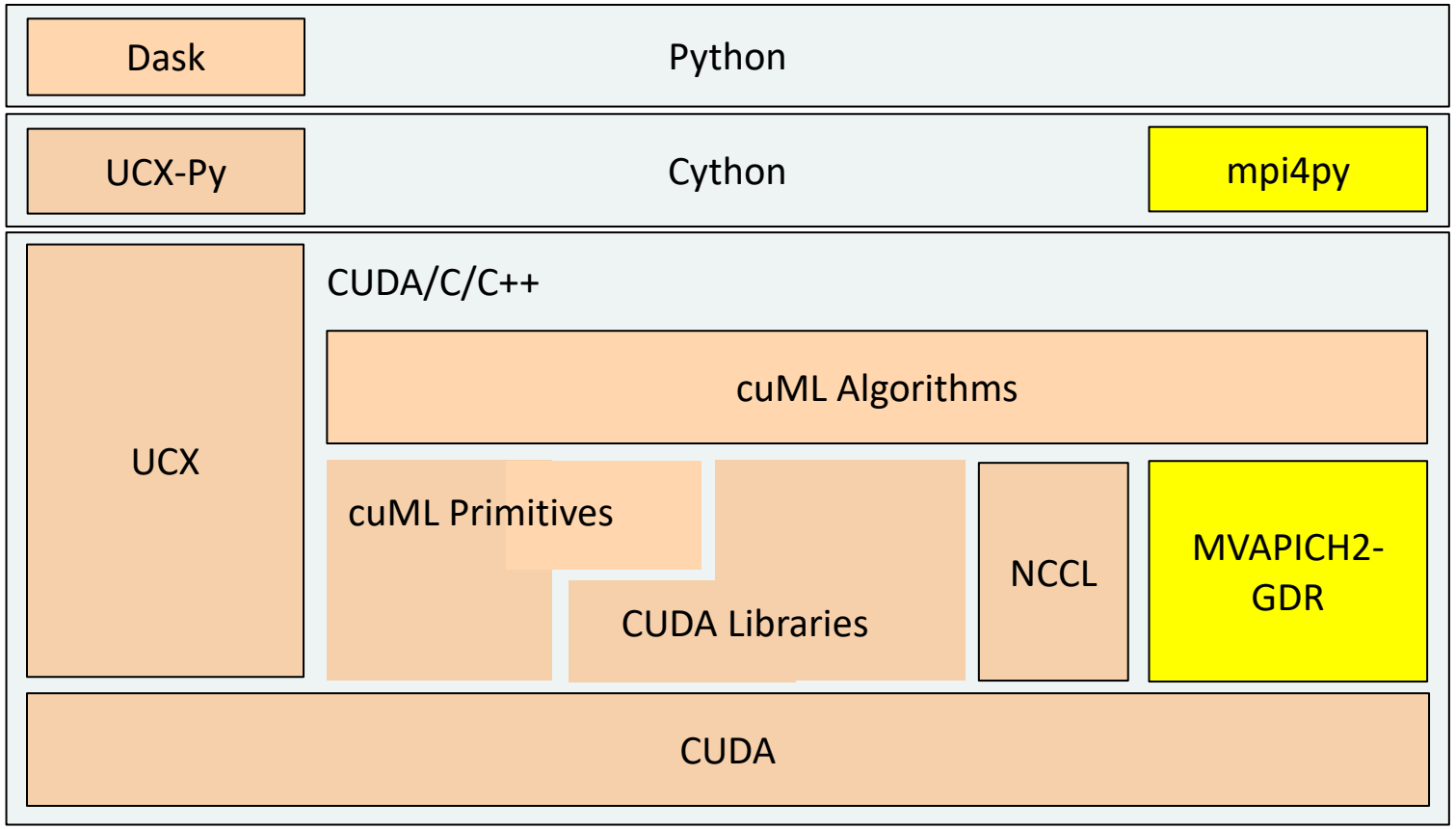
(a). Parallel inference on different models. We measure the overall time spent on parallel inference **1, 2, 4, 8, 16, 32** requests.

Sample Designs and Solutions

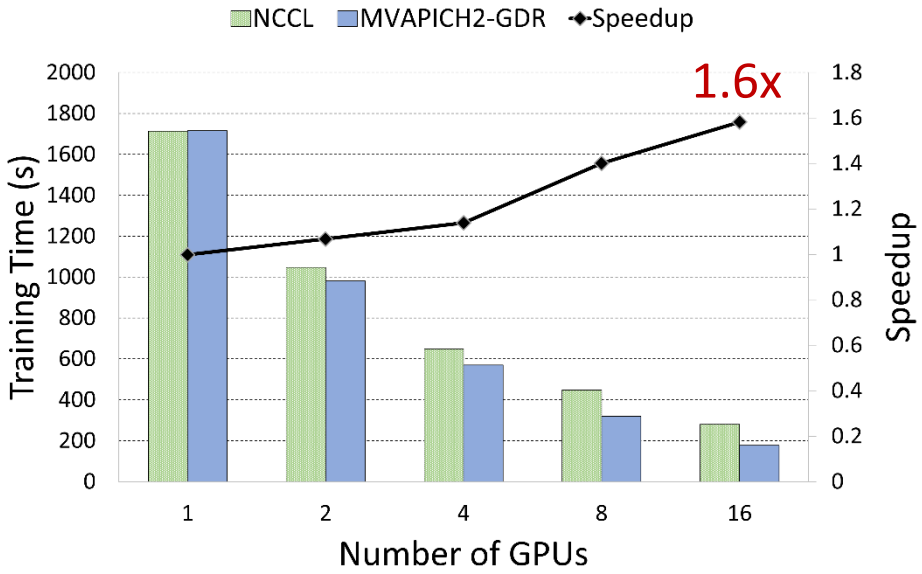
- MPI-Driven High-Performance Distributed Training
  - Exploiting Hybrid (Data and Model) Parallelism for out-of-core training
  - Exploiting on-the-fly compression for LLM training
- Accelerating Parallel Inference
  - In-flight Batching and MOE Models
- **Accelerating CuML Applications**

Accelerating cuML with MVAPICH2-GDR

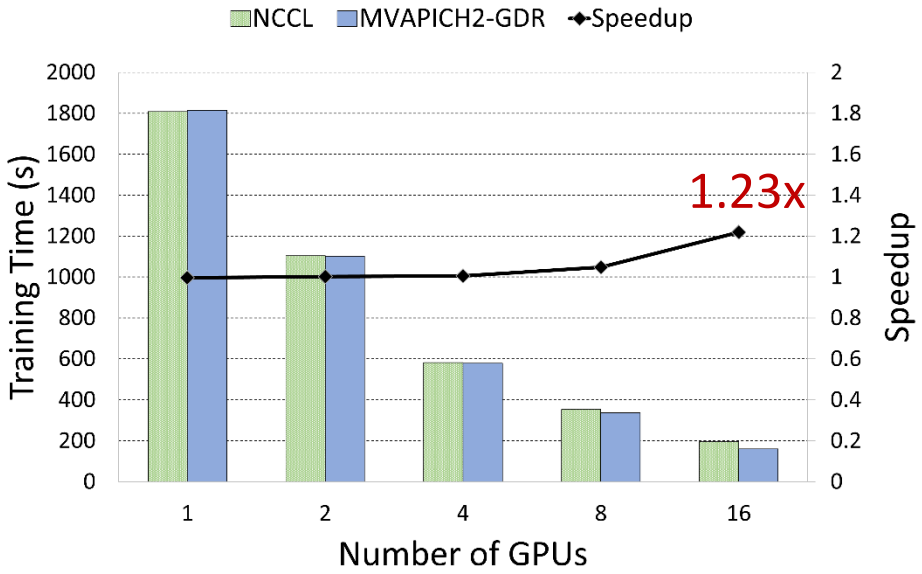
- Utilize MVAPICH2-GDR (with mpi4py) as communication backend during the training phase (the fit() function) in the Multi-node Multi-GPU (MNMG) setting over cluster of GPUs
- Communication primitives:
  - Allreduce
  - Reduce
  - Broadcast
- Exploit optimized collectives



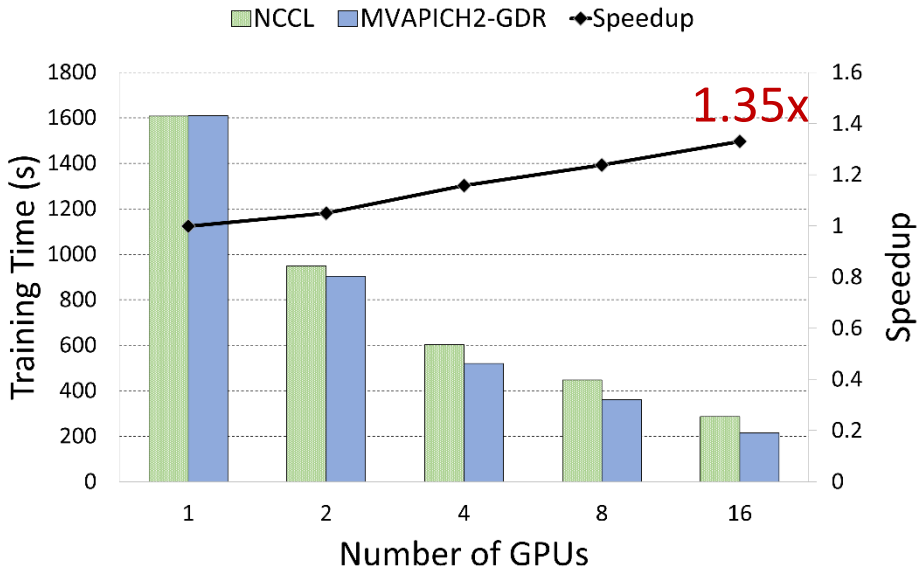
K-Means



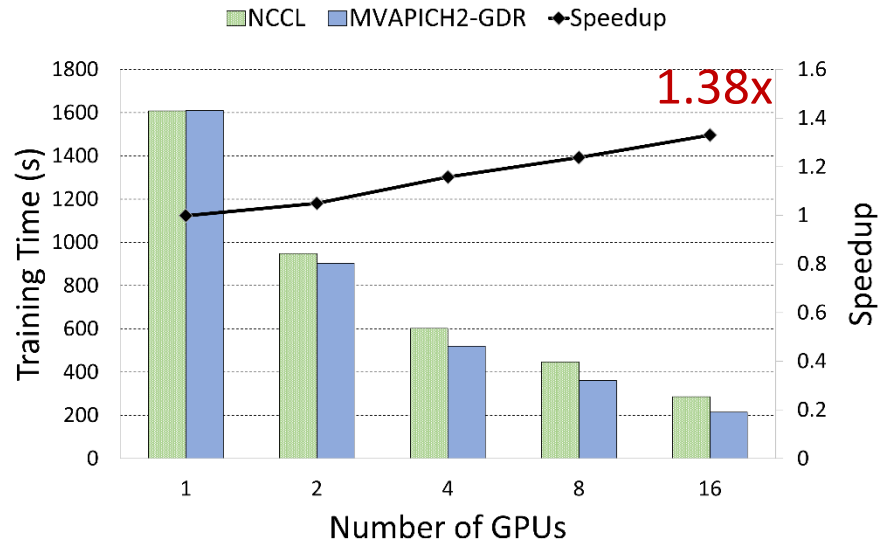
Linear Regression



Nearest Neighbors



Truncated SVD



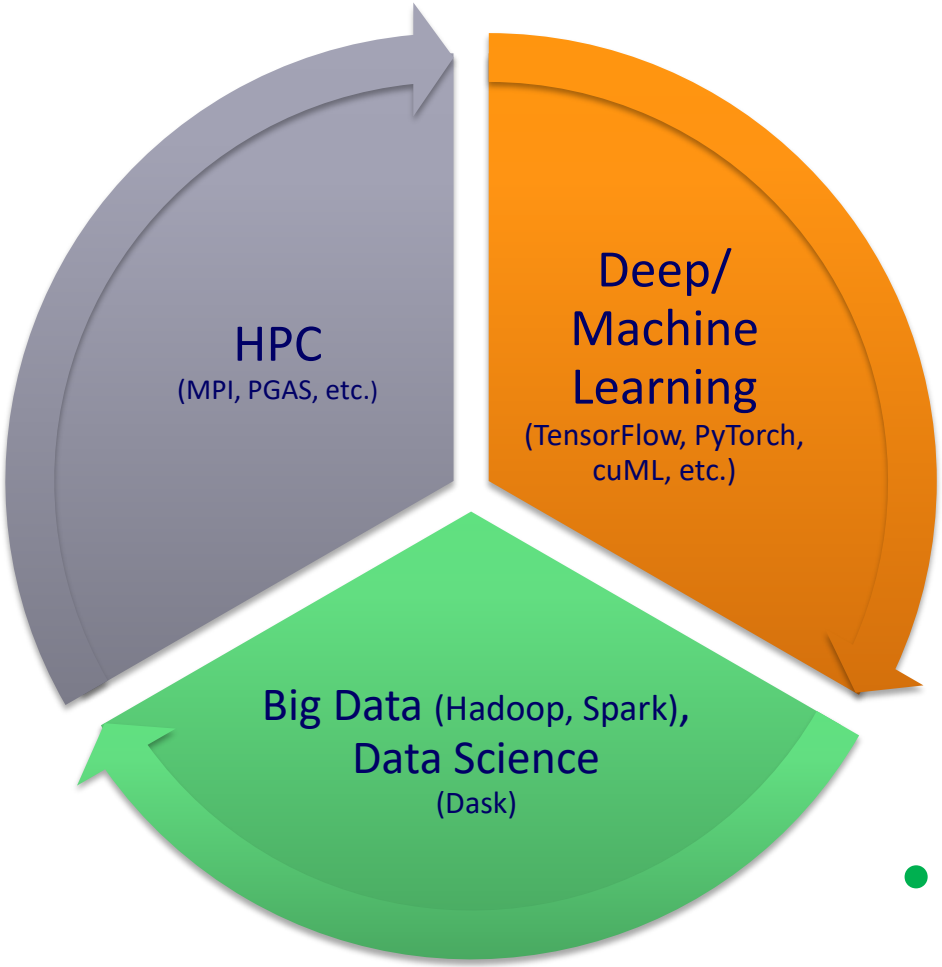
MPI4cuML 0.5

Expanse GPU System	
CPU Model	Intel Xeon Gold 6248
CPU Core Info	2x20 @ 2.5Ghz
Memory	384 GB
Interconnect	InfiniBand HDR (200 Gbps)
OS	Rocky Linux 8.5
GPU	NVIDIA V100 (4/Node)
CUDA	CUDA 11.2

M. Ghazimirsaeed , Q. Anthony , A. Shafi , H. Subramoni , and D. K. Panda, Accelerating GPU-based Machine Learning in Python using MPI Library: A Case Study with MVAPICH2-GDR, MLHPC Workshop, Nov 2020

MVAPICH-Driven Converged Software Stack for AI, Big Data and Data Science

- MVAPICH

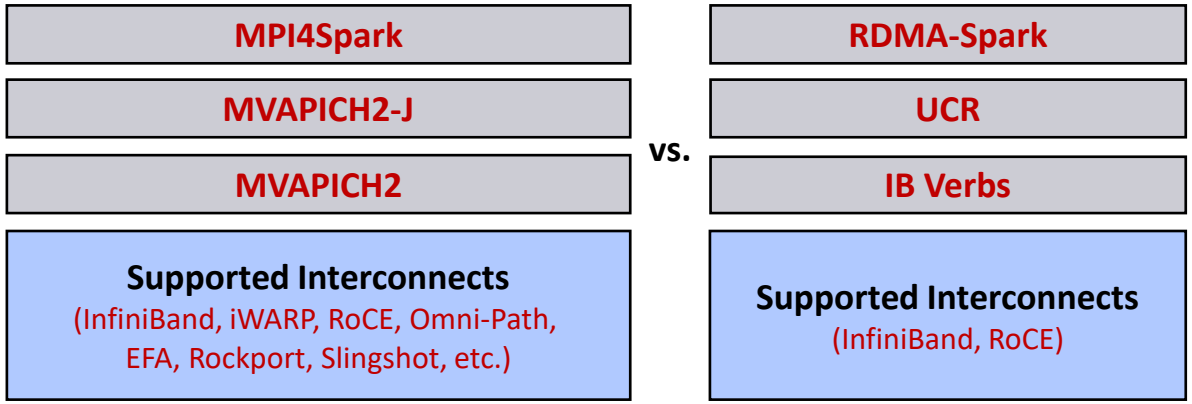


- MPI4DL
- MPI4cuML
- MCR-DL
- ParaInfer-X

- MPI4Spark
- MPI4Dask

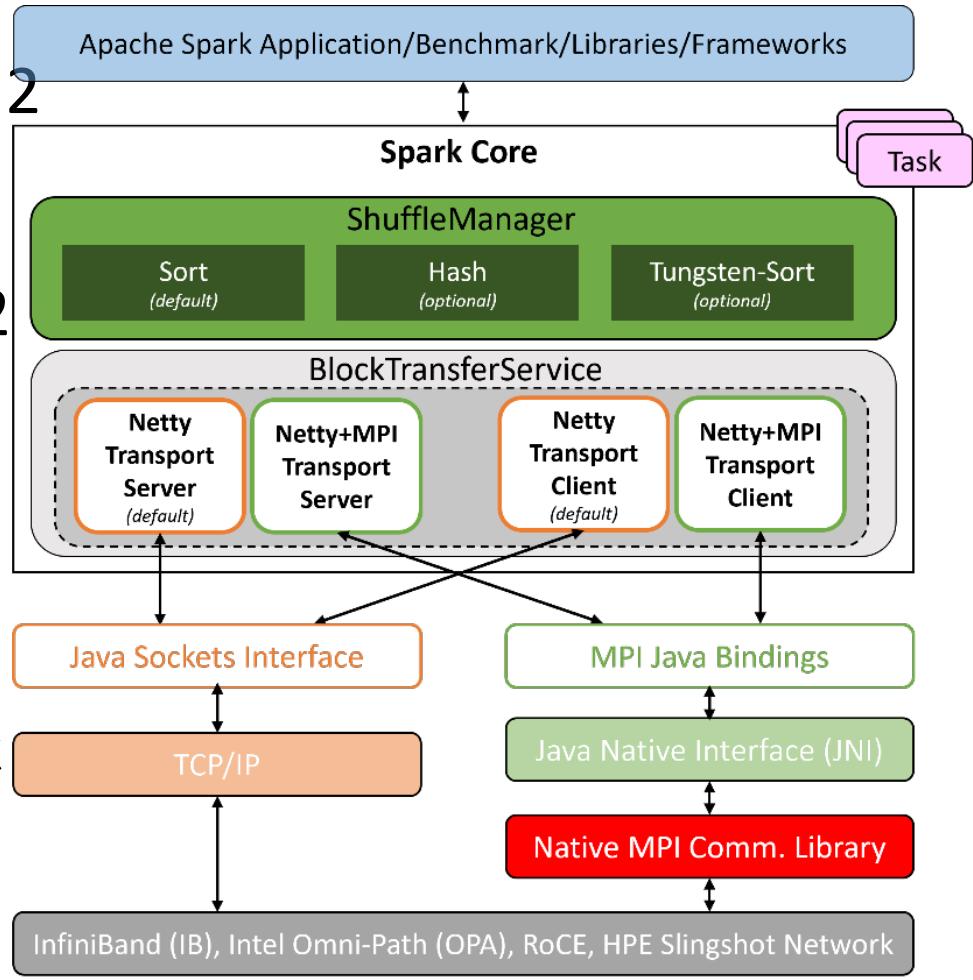
MPI4Spark: Supporting Spark over MVAPICH2

- The current approach is different from its predecessor design, RDMA-Spark (http://hibd.cse.ohio-state.edu)
  - RDMA-Spark supports only InfiniBand and RoCE
  - Requires new designs for new interconnect
- MPI4Spark supports multiple interconnects/systems through a common MPI library
  - Such as InfiniBand (IB), Intel Omni-Path (OPA), HPE Slingshot, RoCE, and others
  - No need to re-design the stack for a new interconnect as long as the MPI library supports it

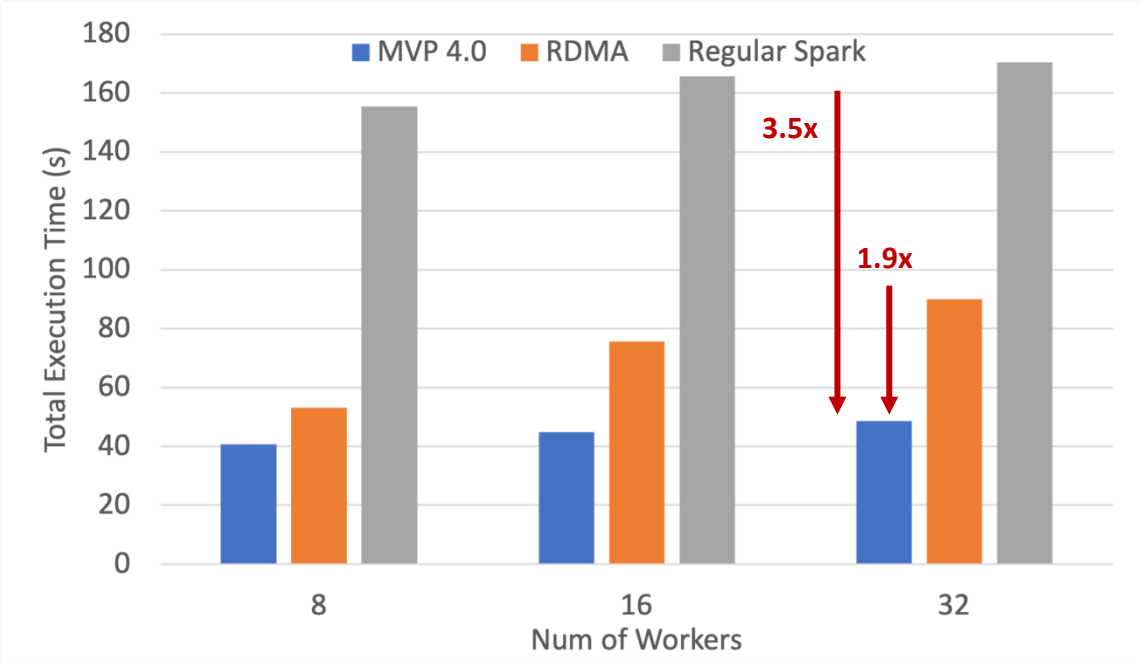


MPI4Spark: Using MVAPICH2 to Optimize Apache Spark

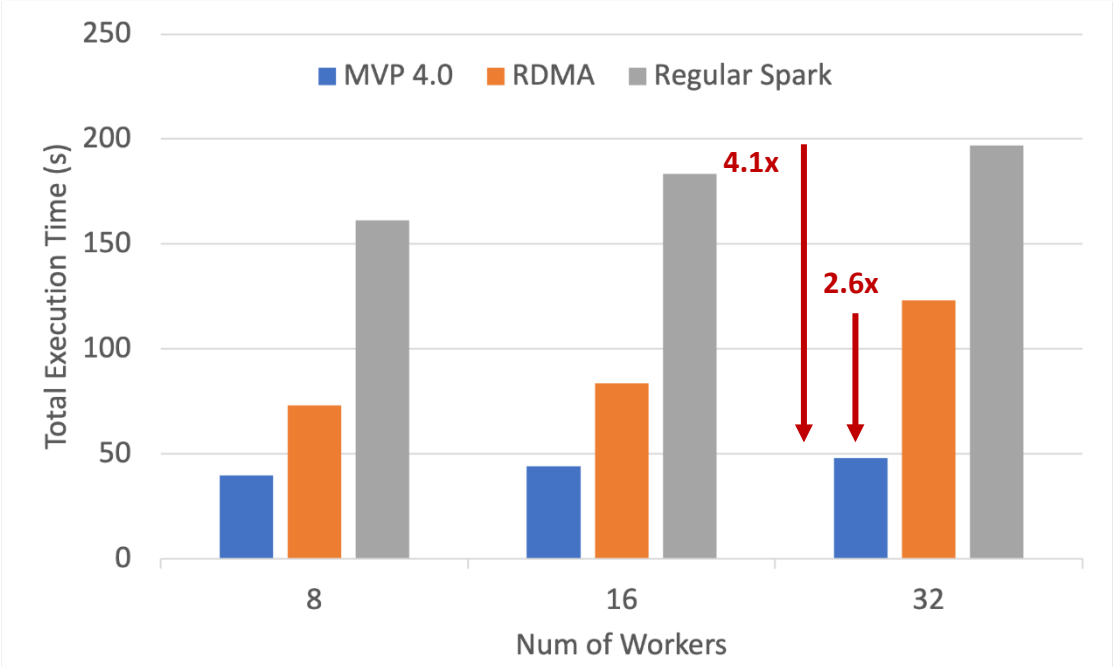
- The main motivation of this work is to utilize the communication functionality provided by MVAPICH2 in the Apache Spark framework
- MPI4Spark relies on Java bindings of the MVAPICH2 library
- Spark's default ShuffleManager relies on Netty for communication:
  - Netty is a Java New I/O (NIO) client/server framework for event-based networking applications
  - The key idea is to utilize MPI-based point-to-point communication inside Netty



Performance Evaluation with MPI4Spark + MVP 4.0



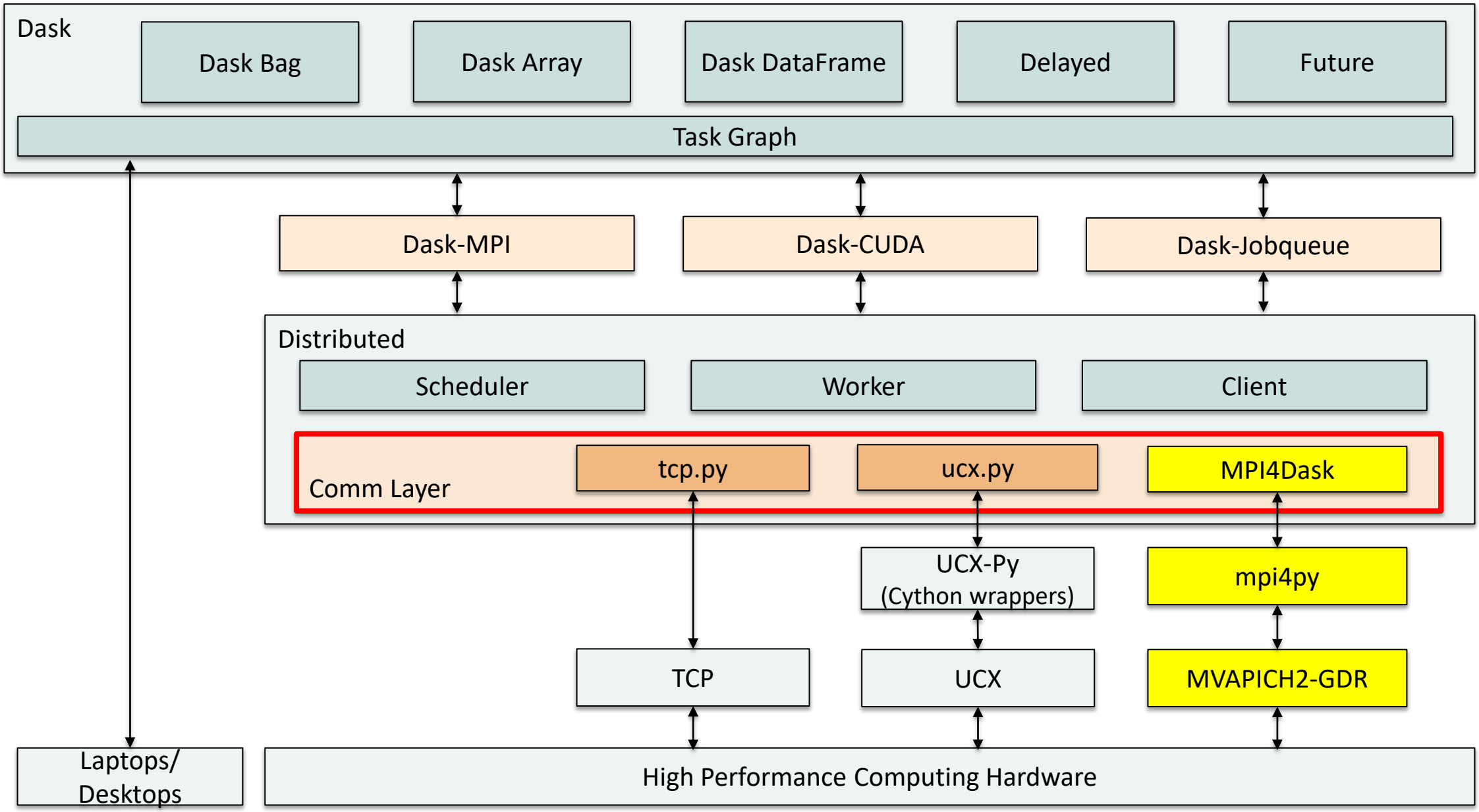
OHB-Sortby



OHB-Groupby

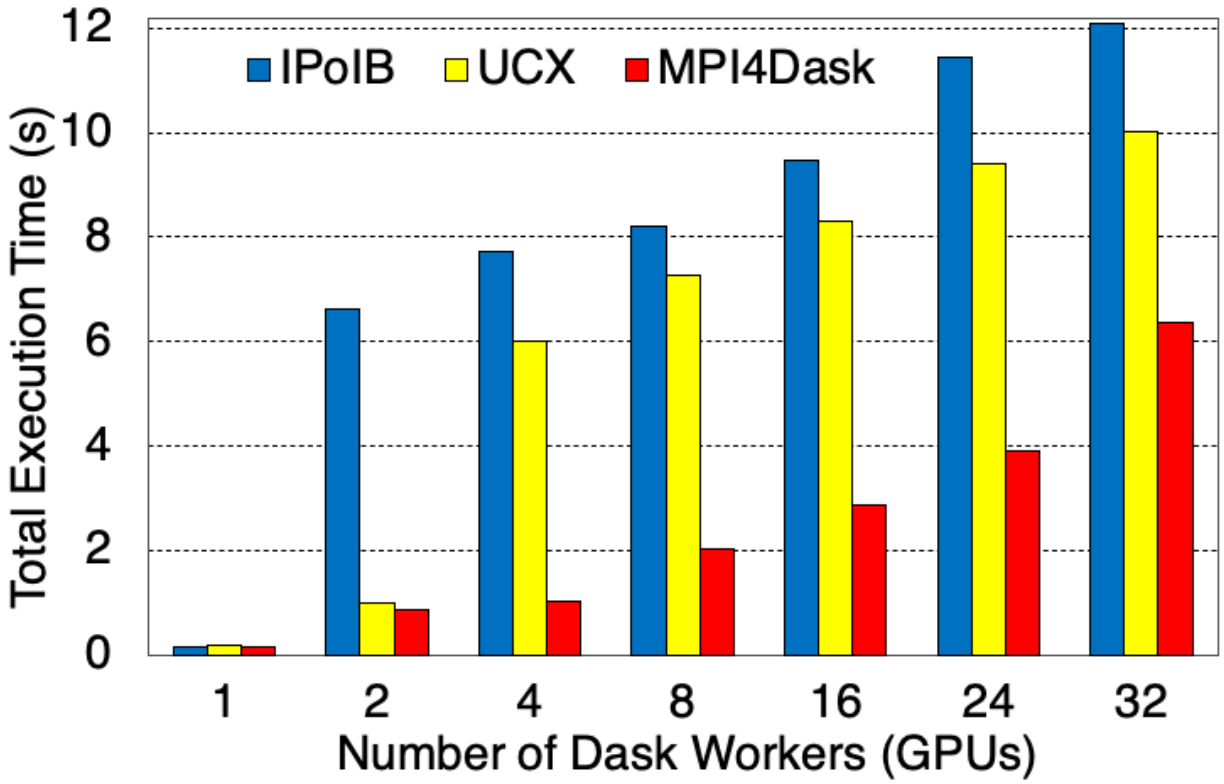
- The following are weak-scaling performance numbers of OHB benchmarks (GroupByTest and SortByTest) executed on the TACC Frontera system using MVAPICH version 4.0
- Speed-ups for the overall total execution time for 32 workers with GroupByTest is 4.1x and 2.6x compared to (regular) Spark and RDMA Spark, and for SortByTest the speed-ups are 3.5 and 1.9x, respectively.

MPI4Dask in the Dask Architecture

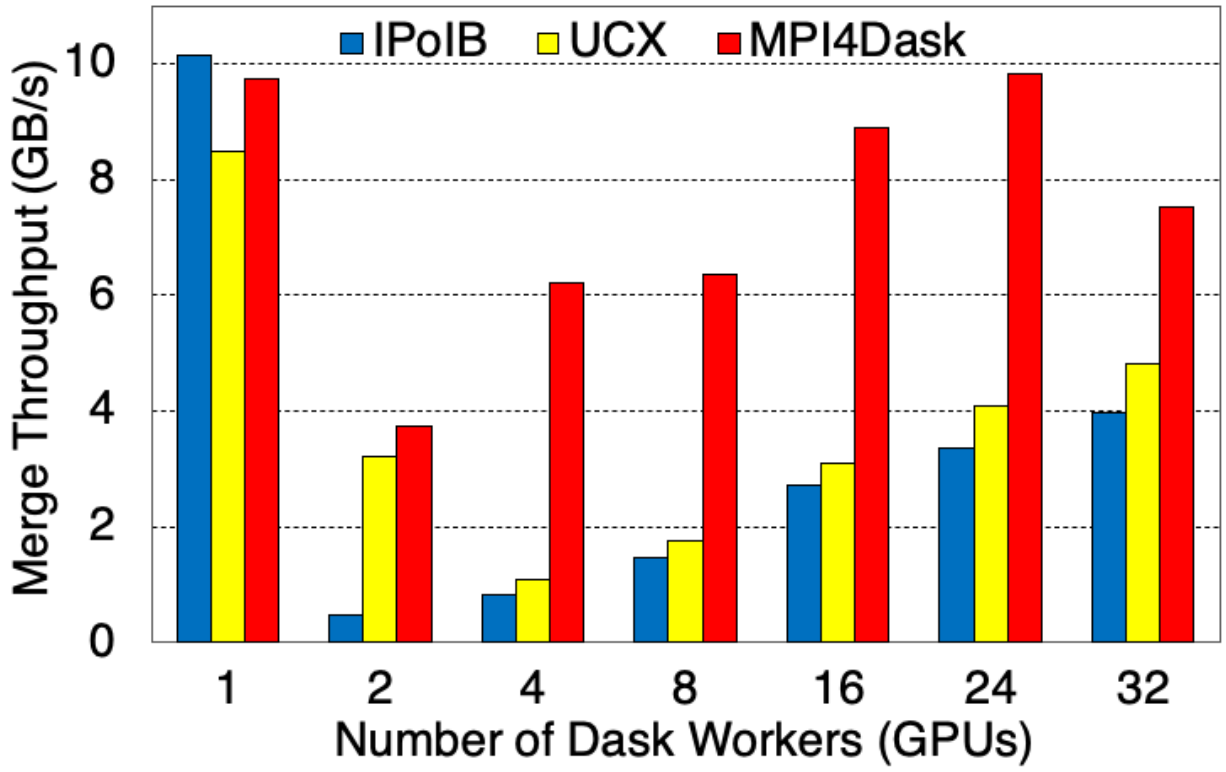


cuDF Merge (TACC Frontera GPU Subsystem)

2.91x better on average



2.90x better on average



A. Shafi , J. Hashmi , H. Subramoni , and D. K. Panda, Efficient MPI-based Communication for GPU-Accelerated Dask Applications, CCGrid '21
https://arxiv.org/abs/2101.08878

MPI4Dask 0.2 release

(http://hibd.cse.ohio-state.edu)

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - **OSU Micro-Benchmarks (OMB)**
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

OSU Microbenchmarks

- Available since 2004
- Suite of microbenchmarks to study communication performance of various programming models
- Benchmarks available for the following programming models
  - Message Passing Interface (MPI)
  - Partitioned Global Address Space (PGAS)
    - Unified Parallel C (UPC), Unified Parallel C++ (UPC++), and OpenSHMEM
- Benchmarks available for multiple accelerator-based architectures
  - Compute Unified Device Architecture (CUDA)
  - OpenACC Application Program Interface
- Part of various national resource procurement suites like NERSC-8 / Trinity Benchmarks
- Please visit the following link for more information: http://mvapich.cse.ohio-state.edu/benchmarks/

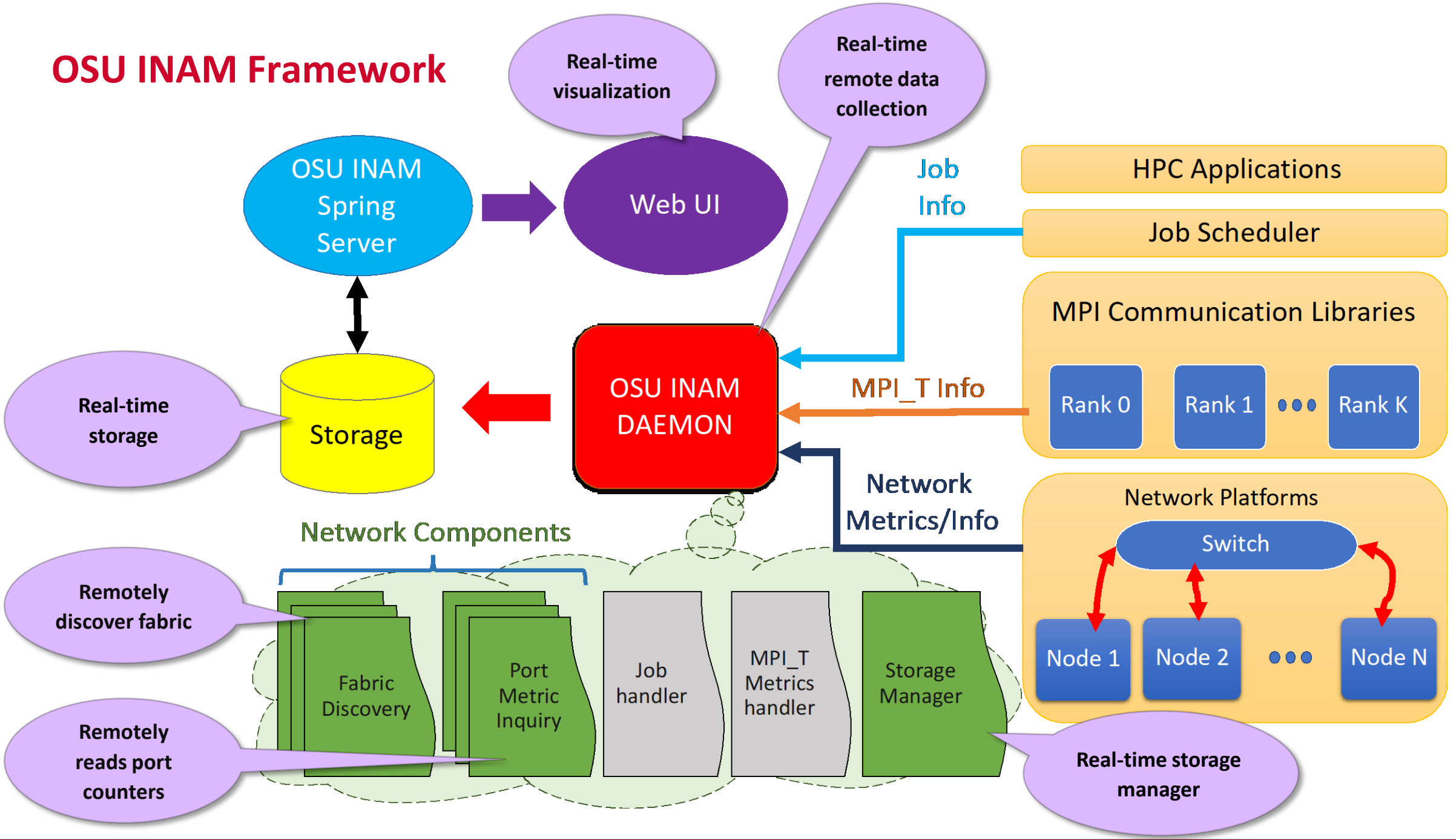
OSU Micro Benchmarks v7.4

- New features since MUG'23
  - **Add support for RCCL benchmarks.**
    - **Pt2pt, Collective**
  - **Add new benchmarks for persistent collectives.**
  - **Add new benchmarks to measure network congestion.**
    - **osu_bw_fan_in, osu_bw_fan_out**
  - **Add support for custom percentile values to evaluate benchmark performance.**
  - **Add support to log validation failures.**
  - **Add new collective benchmarks**
    - **osu_reduce_scatter_block, osu_ireduce_scatter_block**

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - **InfiniBand Network Analysis and Monitoring (INAM)**
  - Applications: Best Practices
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

OSU INAM Framework



Overview of OSU InfiniBand Network Analysis and Monitoring (INAM) Tool

- **A network monitoring and analysis tool that is capable of analyzing traffic on the InfiniBand network with inputs from the MPI runtime**
  - http://mvapich.cse.ohio-state.edu/tools/osu-inam/
- **Monitors IB clusters in real time by querying various subnet management entities and gathering input from the MPI runtimes**
- **Capability to analyze and profile node-level, job-level and process-level activities for MPI communication**
  - Point-to-Point, Collectives and RMA
- **Ability to filter data based on type of counters using "drop down" list**
- **Remotely monitor various metrics of MPI processes at user specified granularity**
- **"Job Page" to display jobs in ascending/descending order of various performance metrics in conjunction with MVAPICH2-X**
- **Visualize the data transfer happening in a "live" or "historical" fashion for entire network, job or set of nodes**
- **Sub-second port query and fabric discovery in less than 10 mins for ~2,000 nodes**

- **OSU INAM 1.0 released**
  - Enhanced the UI by making asynchronous API calls for data loading
    - Significantly improved page load performance
  - Support for continuous queries to improve visualization performance
  - Support for MySQL and InfluxDB as database backends
    - Enhanced database insertion using InfluxDB
  - Support for PBS and SLURM job scheduler as config time
    - Support for SLURM multi-cluster configuration
  - Enable emulation mode to allow users to test OSU INAM tool in a sandbox environment without actual deployment
  - Generate email notifications to alert users when user defined events occur
  - Support to display node-/job-level CPU, Virtual Memory, and Communication Buffer utilization information for historical jobs
  - Support to handle multiple job schedulers on the same fabric
  - Support to collect and visualize MPI_T based performance data
  - Support for MOFED 4.5, 4.9+, 5+
  - Support for adding user-defined labels for switches to allow better readability and usability
  - Support authentication for accessing the OSU INAM webpage
  - Optimized webpage rendering and database fetch/purge capabilities

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- **Features and Performance of Recent Releases**
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - **Applications: Best Practices**
- Upcoming Features
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

Applications-Level Tuning: Compilation of Best Practices

- MPI runtime has many parameters
- Tuning a set of parameters can help you to extract higher performance
- Compiled a list of such contributions through the MVAPICH Website
  - http://mvapich.cse.ohio-state.edu/best_practices/
- Initial list of applications
  - Amber
  - HoomDBlue
  - HPCG
  - Lulesh
  - MILC
  - Neuron
  - SMG2000
  - Cloverleaf
  - SPEC (LAMMPS, POP2, TERA_TF, WRF2)
- Soliciting additional contributions, send your results to mvapich-help at cse.ohio-state.edu.
- We will link these results with credits to you.

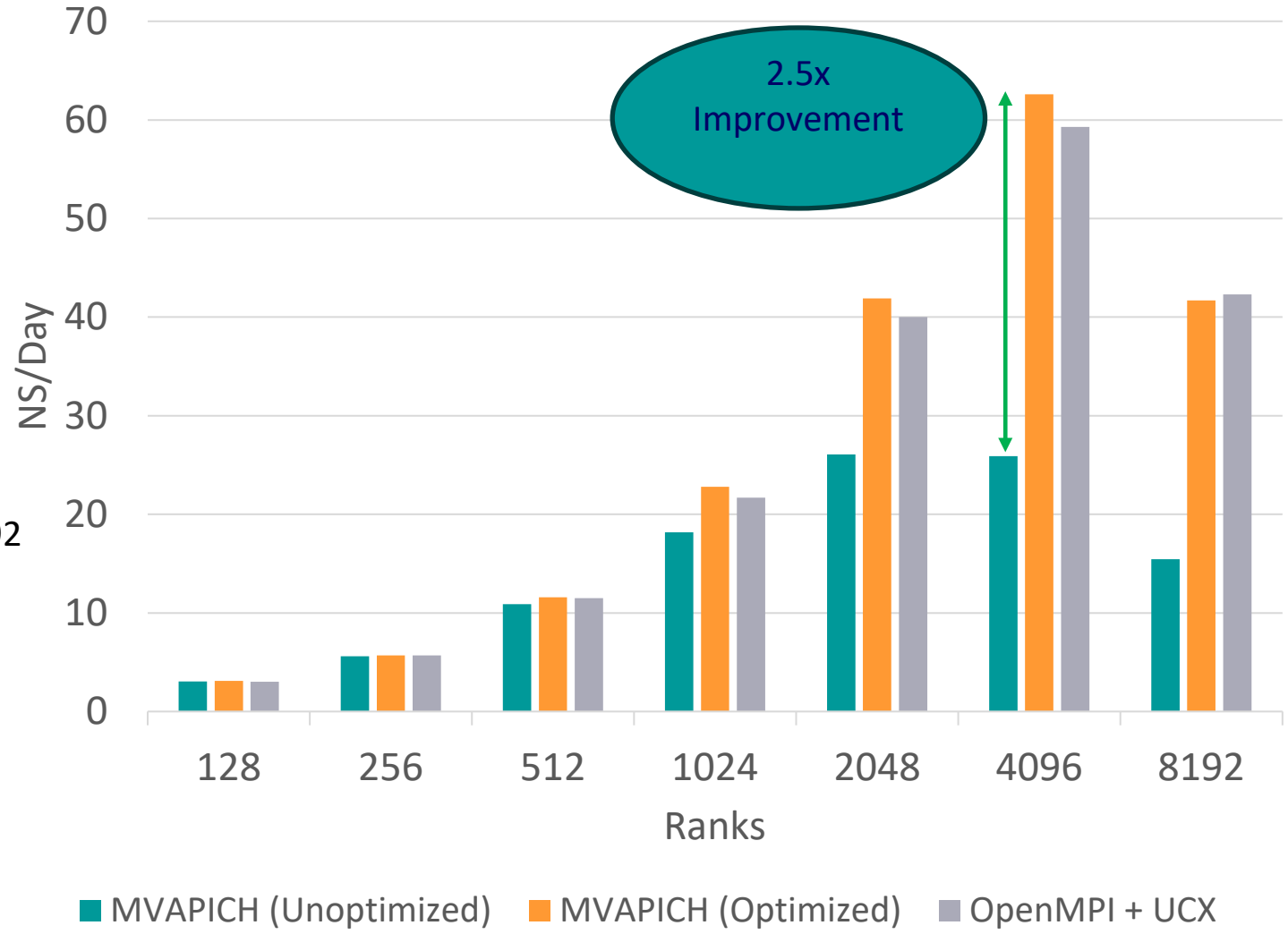
Case Study: GROMACS – Performance Engineering with TAU

Diagnosis and workaround found

- Investigate UD communication (read progress poll)
- Use RC to get the desired lead in performance
- Gains:
  - 2.5x improvement over MVAPICH baseline
  - 15% compared to OpenMPI default RC
- Update the following parameter for GROMACS runs

`MV2_HYBRID_ENABLE_THRESHOLD = 8192`

this will enable UD-hybrid communication after the 8192 threshold*

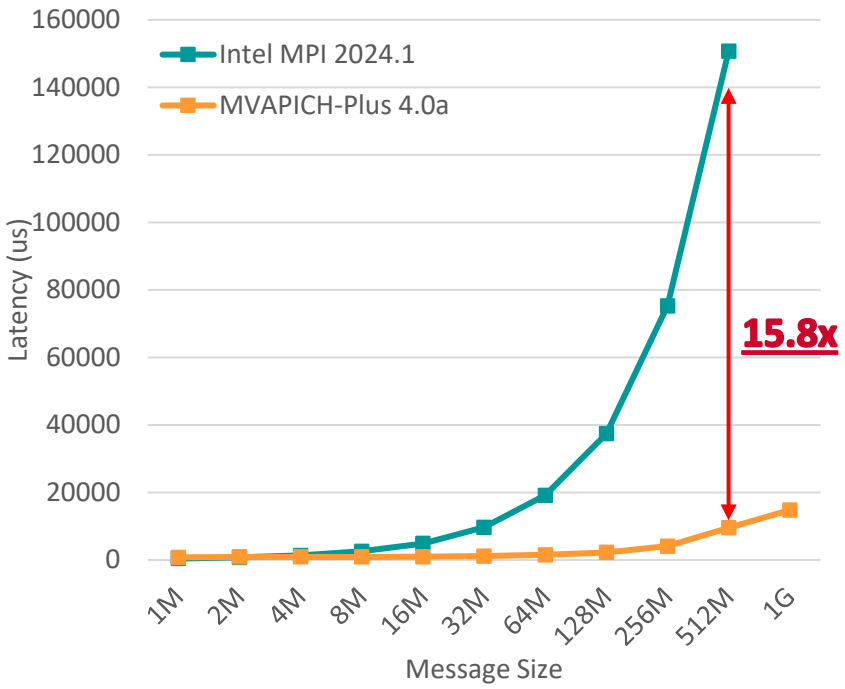


***For more details check user-guide:**
https://mvapich.cse.ohio-state.edu/static/media/mvapich/mvapich2-userguide.html#:~:text=use%20any%20HugePages.-,11.110,-MV2_HYBRID_ENABLE_THRESHOLD

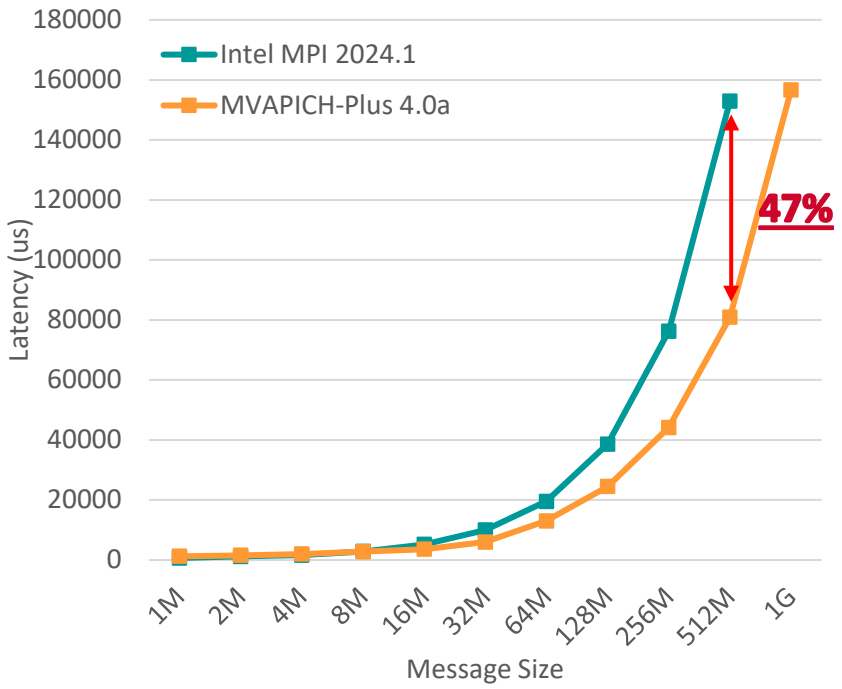
Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - **Support for AMD and Intel GPUs**
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

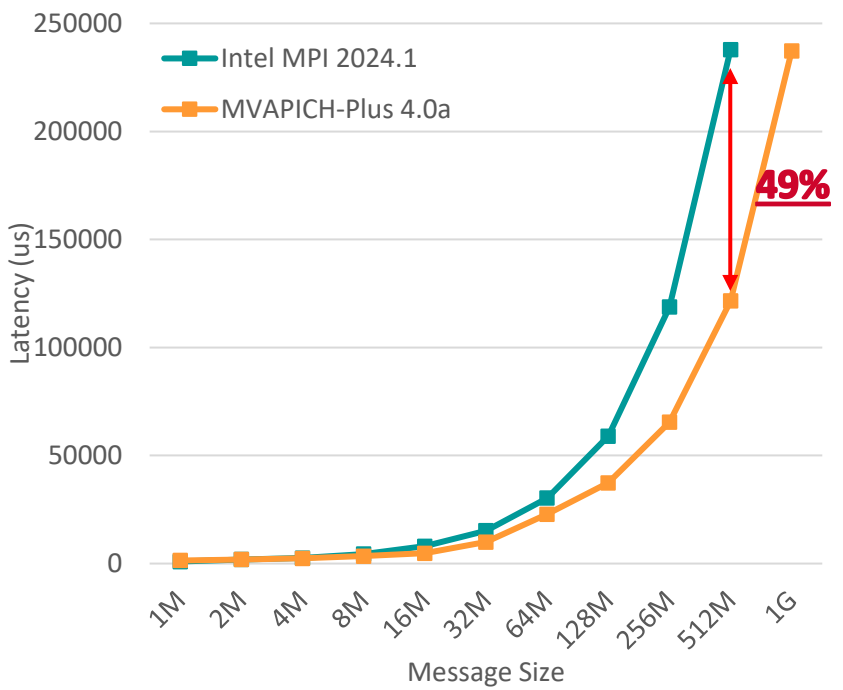
Benchmark-level Evaluations – Allreduce (Intel GPU)



1 nodes, 4 GPN – Large Message

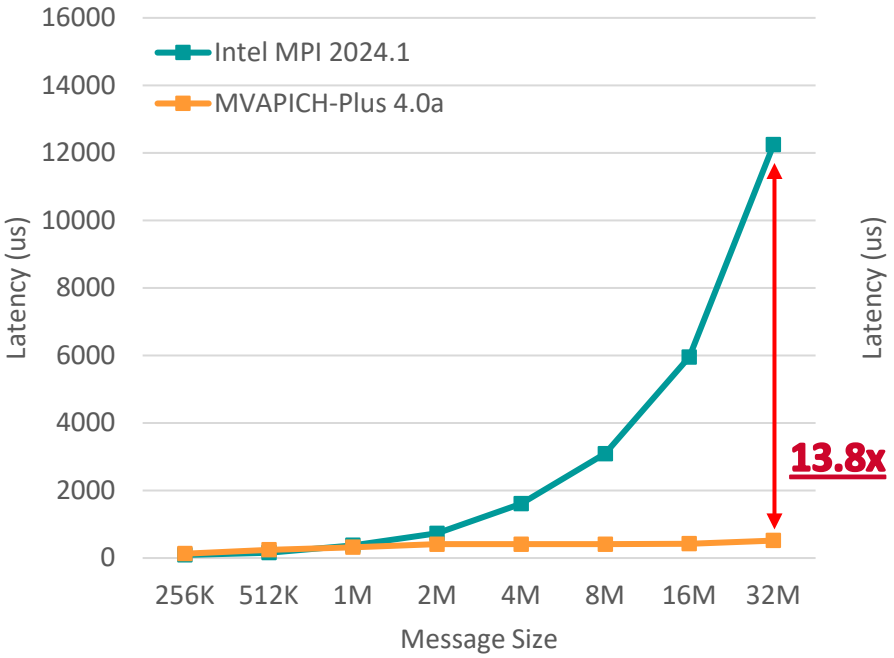


2 nodes, 4 GPN – Large Message

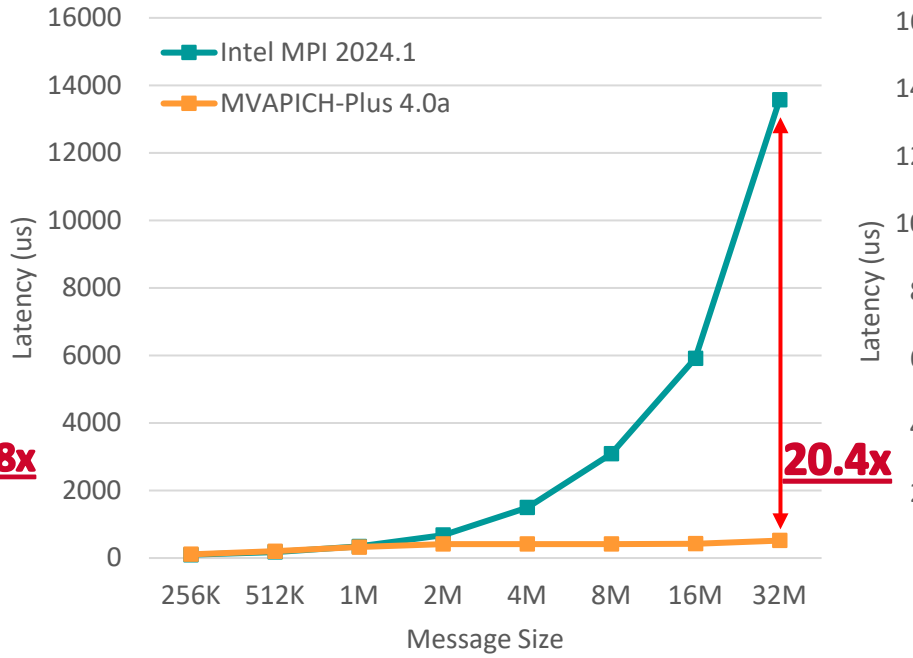


4 nodes, 4 GPN – Large Message

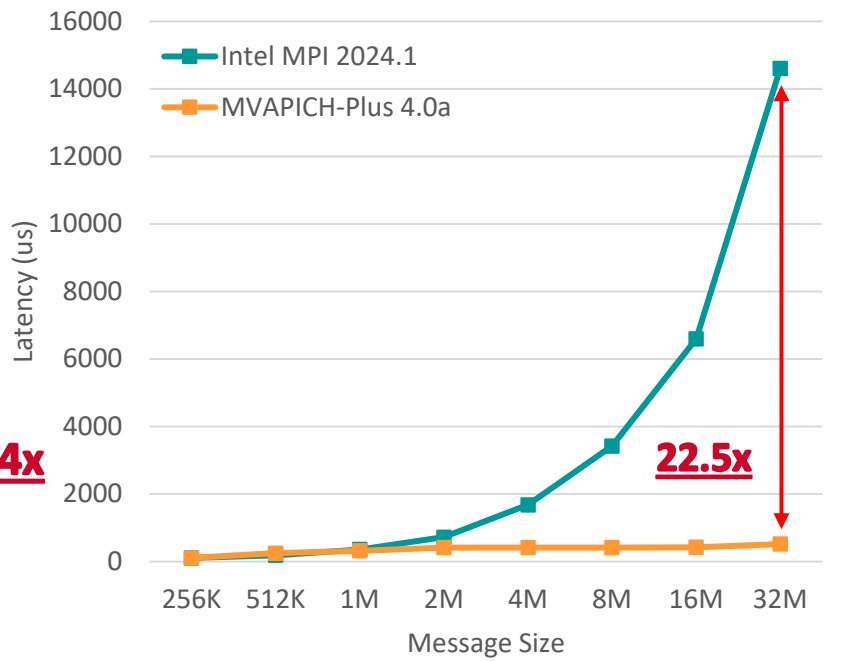
Benchmark-level Evaluations – Other Reduction-based Collectives (Intel GPU)



1 nodes, 4 GPN – Reduce

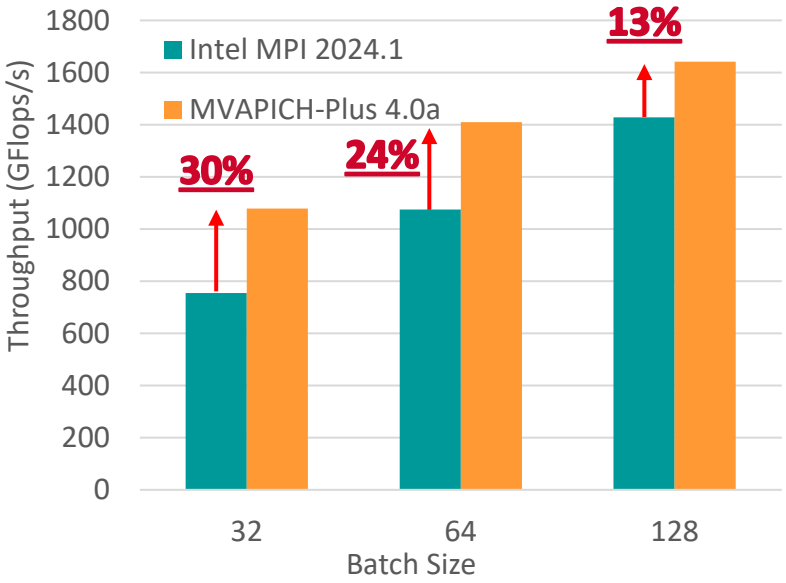


1 nodes, 4 GPN – Reduce_scatter

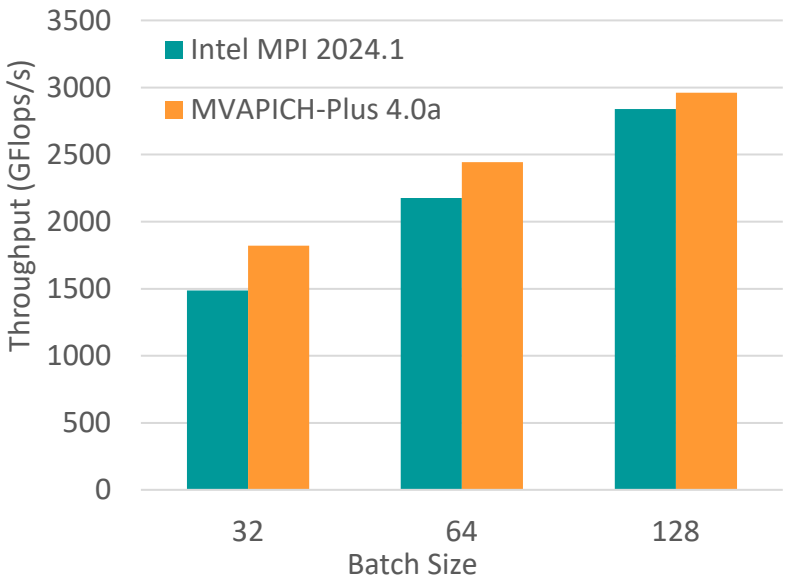


1 nodes, 4 GPN – Reduce_scatter_block

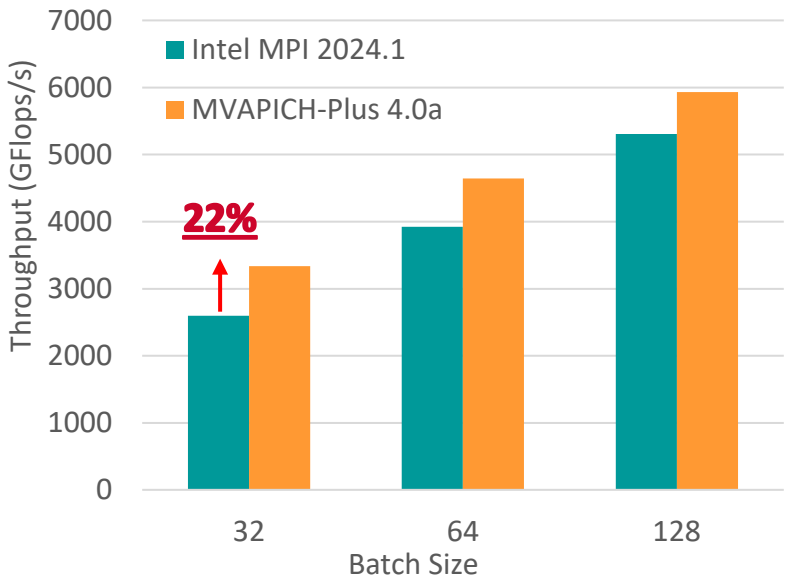
Application-level Evaluations – TensorFlow + Horovod (Intel GPU)



1 nodes, 4 GPN – Large Message

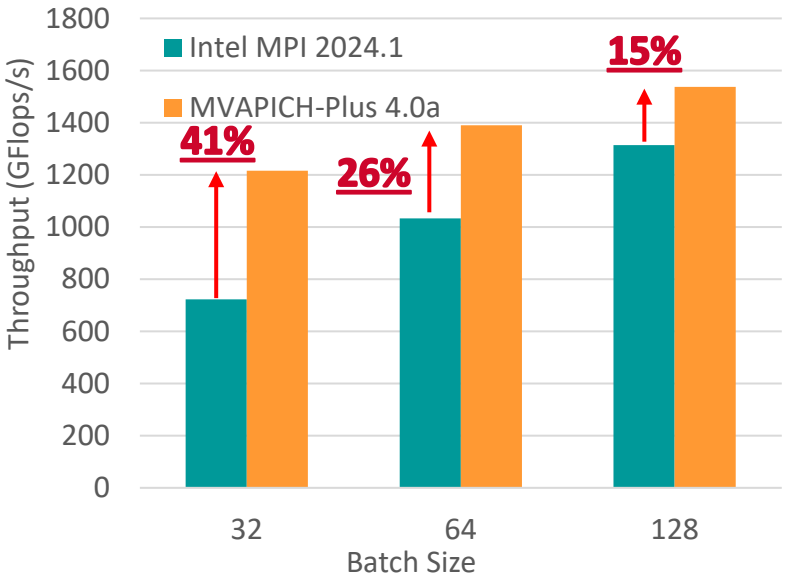


2 nodes, 4 GPN – Large Message

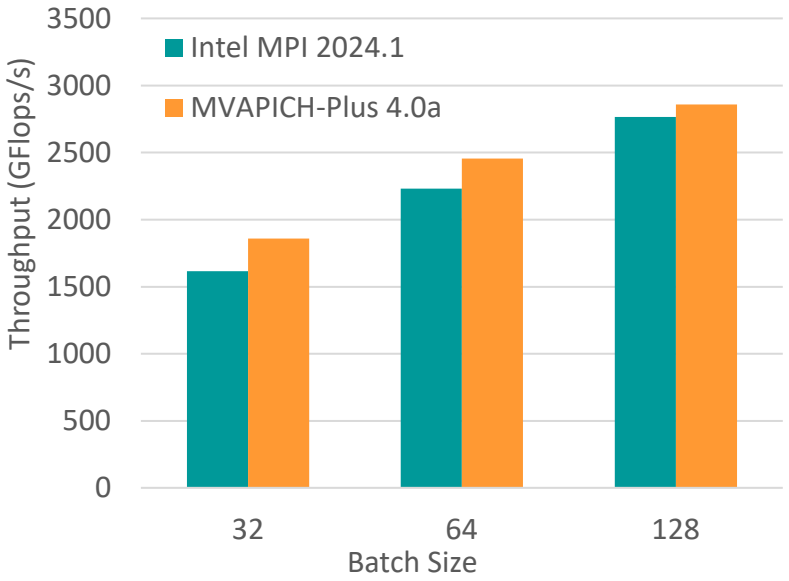


4 nodes, 4 GPN – Large Message

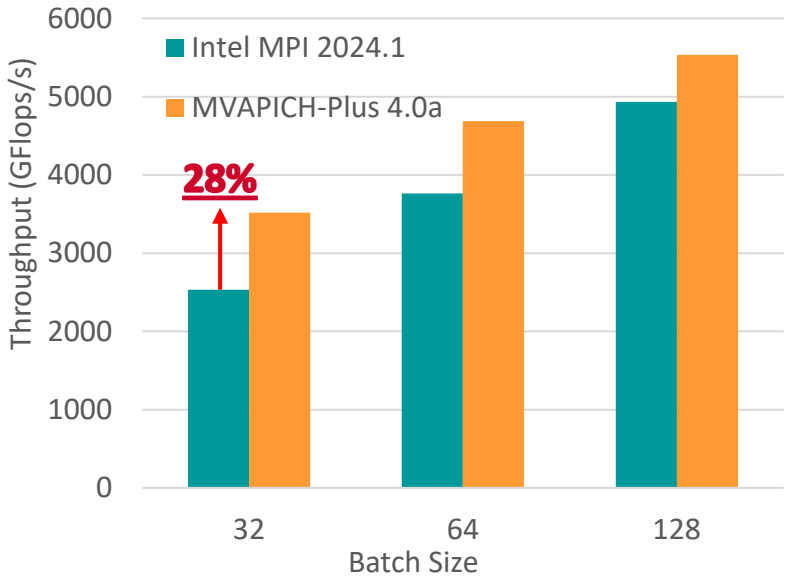
Application-level Evaluations – PyTorch + Horovod (Intel GPU)



1 nodes, 4 GPN – Large Message



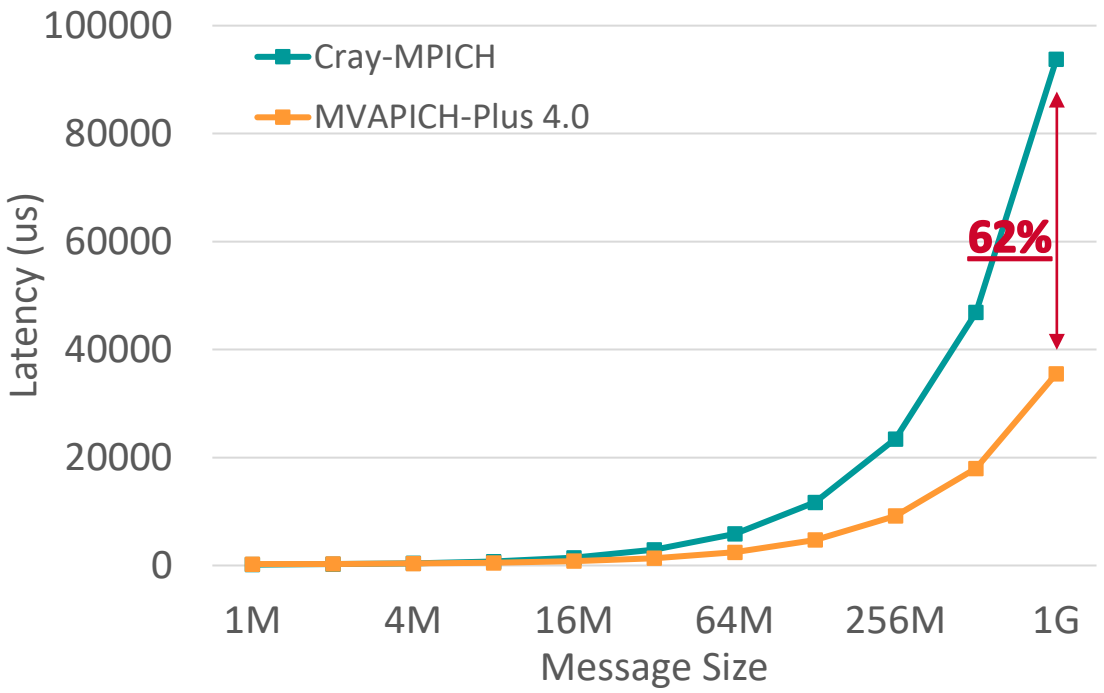
2 nodes, 4 GPN – Large Message



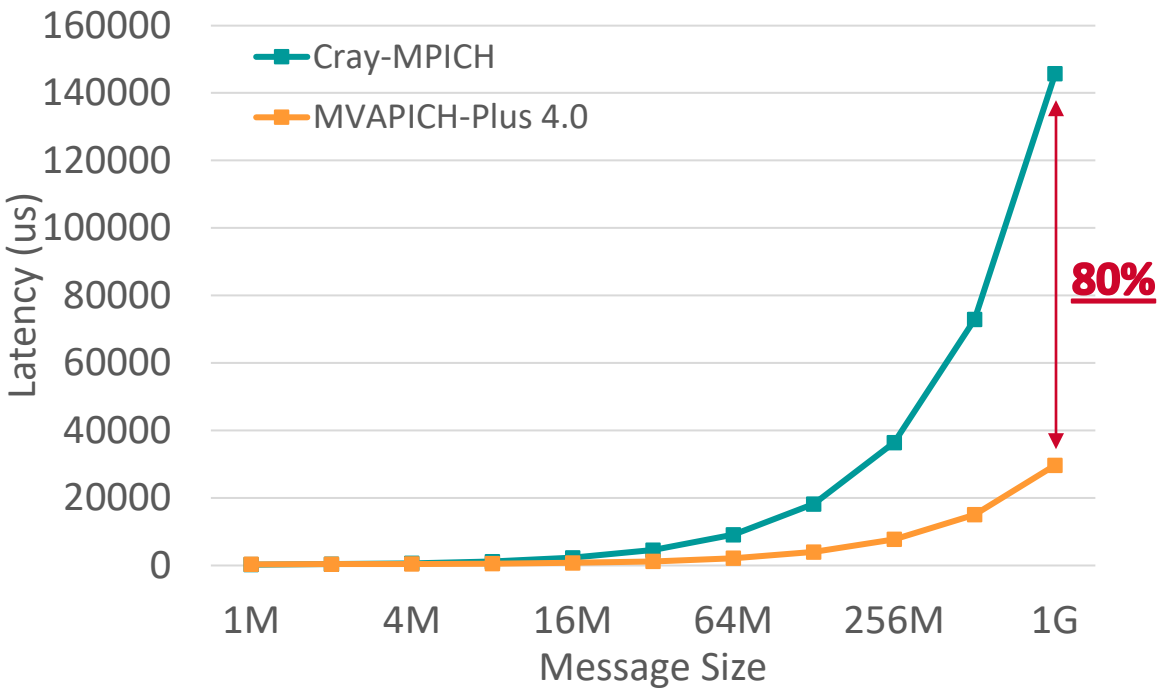
4 nodes, 4 GPN – Large Message

Short talk presented by Chen-Chun (yesterday)

MVAPICH-PLUS GPU Optimized on Frontier (AMD MI250X GPUs) – Allreduce

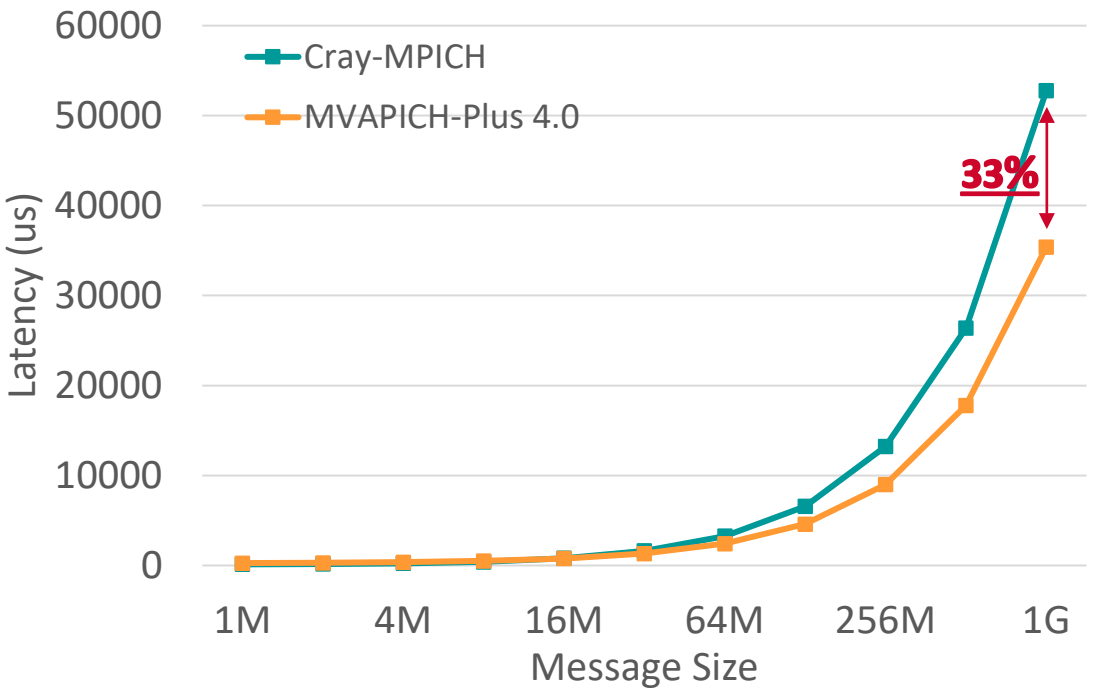


1 nodes, 4 GPN

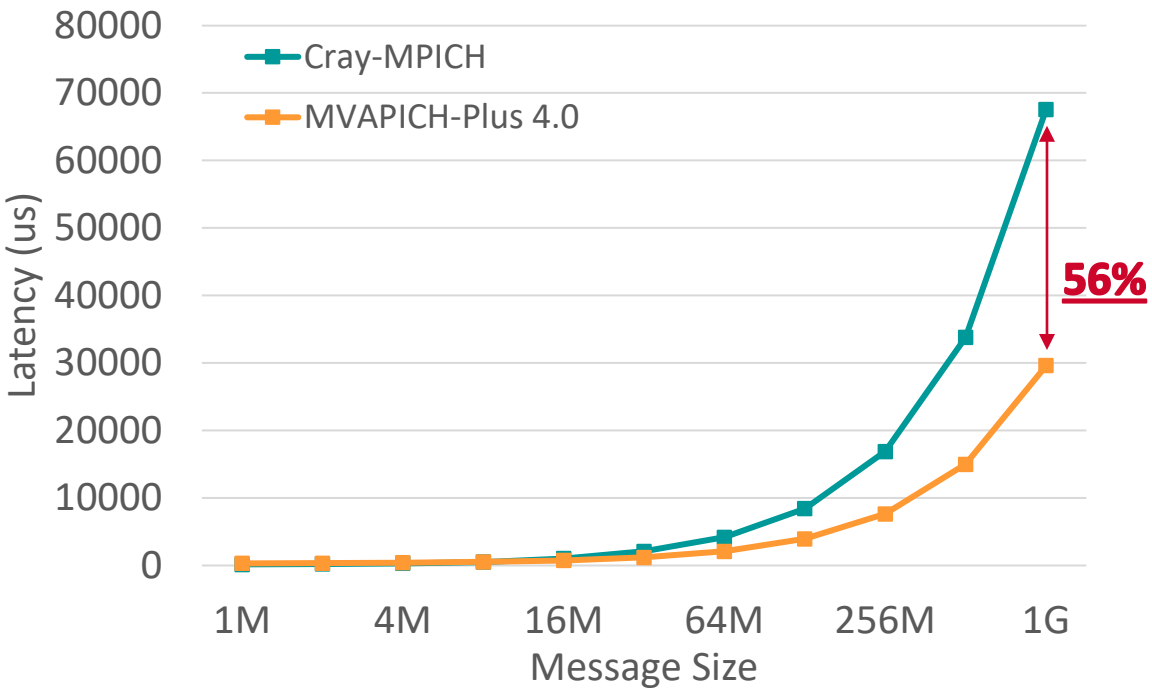


1 nodes, 8 GPN

MVAPICH-PLUS GPU Optimized on Tioga (AMD MI250X GPUs) – Allreduce



1 nodes, 4 GPN



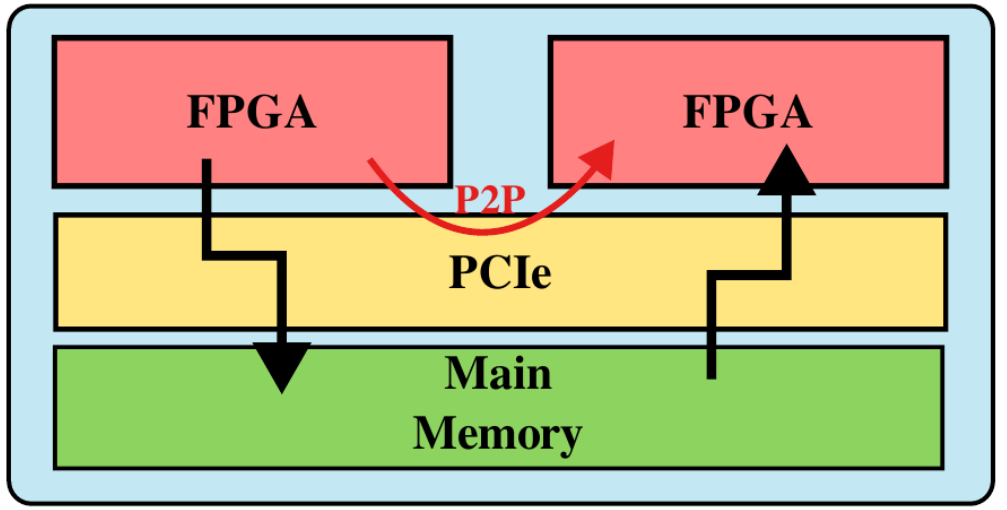
1 nodes, 8 GPN

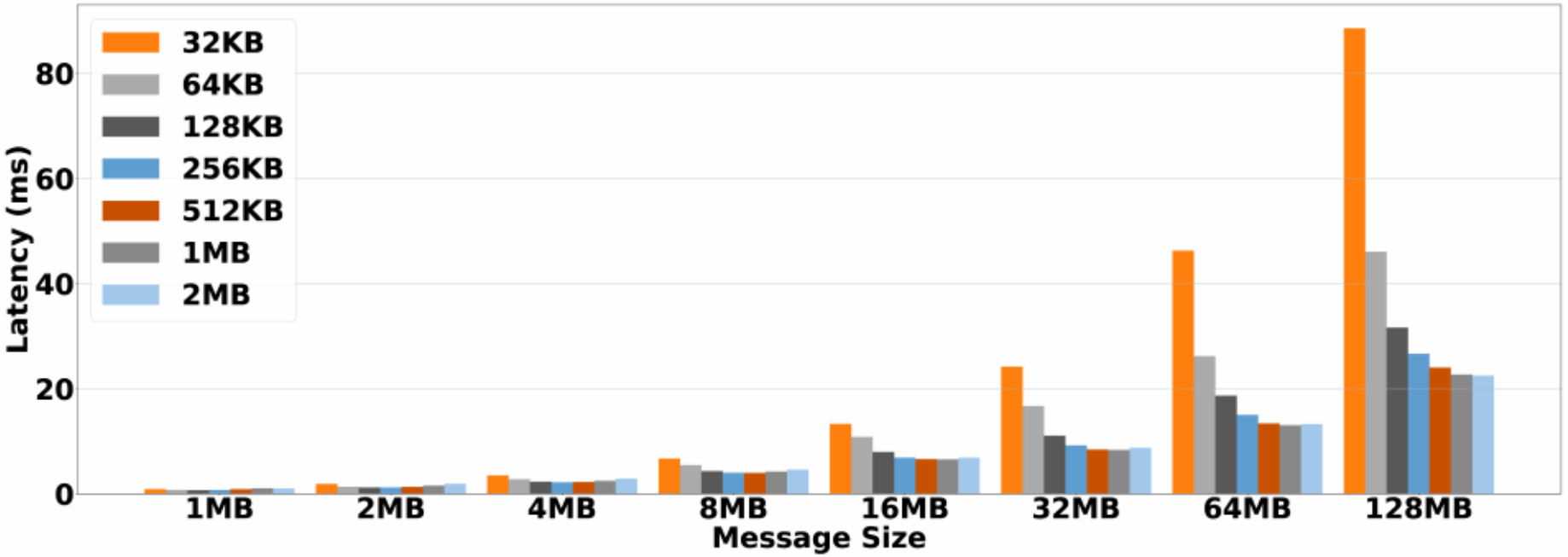
Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - Support for AMD and Intel GPUs
  - **MVAPICH and OMB for FPGA**
  - CXL Support
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

Optimized MVAPICH-FPGA Design

- P2P transfers enable data to travel between devices over PCIe without entering host memory
- Extension of OMB for FPGA-based systems





More details in the Short Talk, presented by Nick Contini (Tomorrow 3:45-5:00 pm)

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - **CXL Support**
  - Accelerating Inference
  - Conversational AI Interface (SAI)
- Conclusions

Motivation

- CXL offers an alternative approach to communicate between compute nodes through CXL devices.

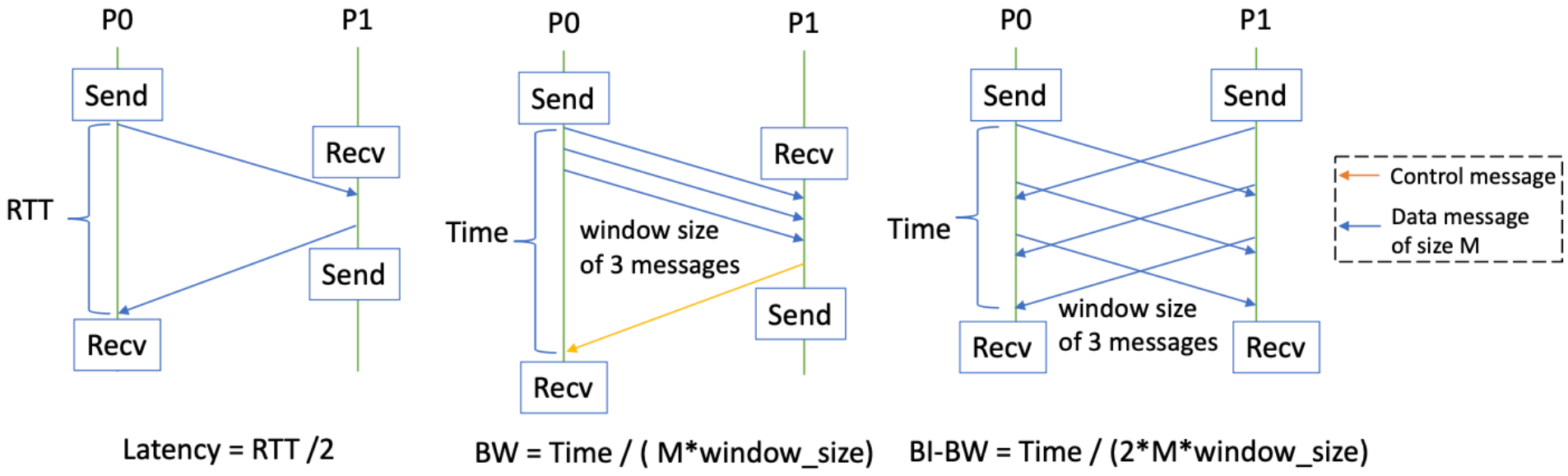
- CXL bridges the gap between local memory and remote memory

Memory Type	Latency (ns)	Connection Type	Latency Increase Factor
Main memory	80-140	CPU-attached	1x
CXL	170-250	CPU-independent	~ 2 - 3x
Disaggregated memory	2000-4000	Network-attached	~ 25 - 50x

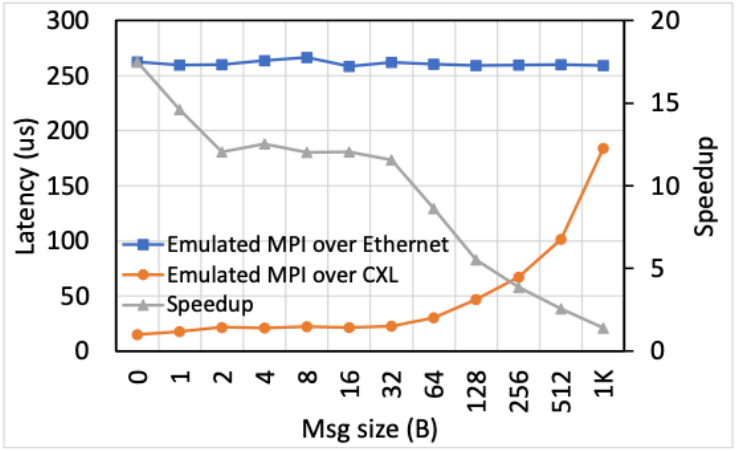
- This enables more efficient inter-node communication by using the CXL

OMB-CXL

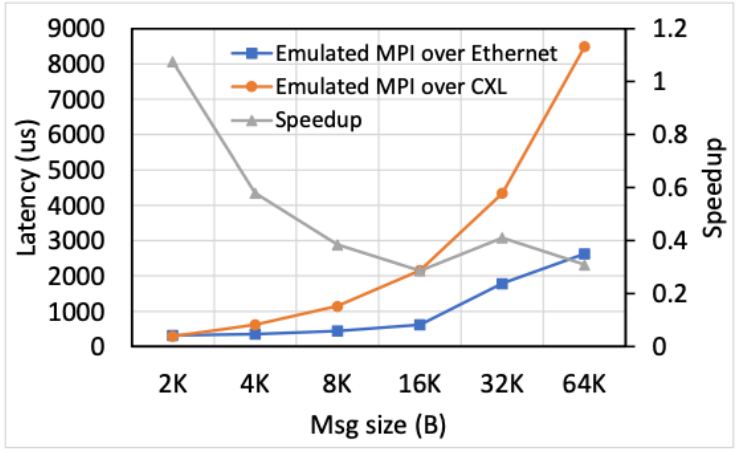
- OMB is a benchmark suite that is used evaluate MPI performance over networks
- We extend OMB and call it OMB-CXL to support P2P communication over CXL channel
- The benchmark supports different buffer configurations. The buffers can be:
  - On host memory
  - On device memory



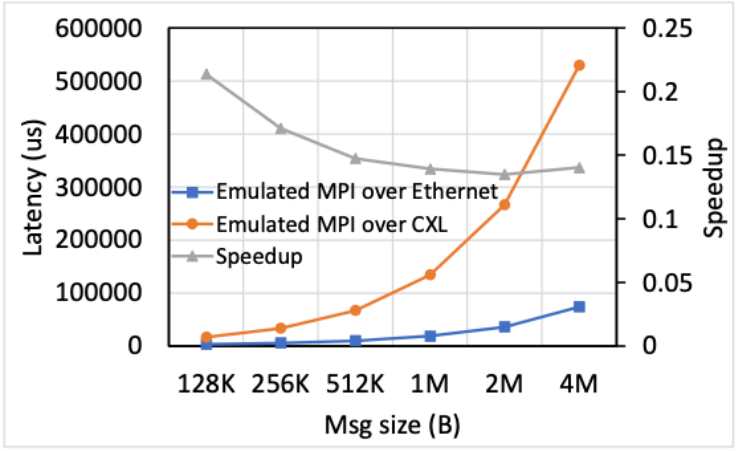
Latency Evaluation for HH and DD



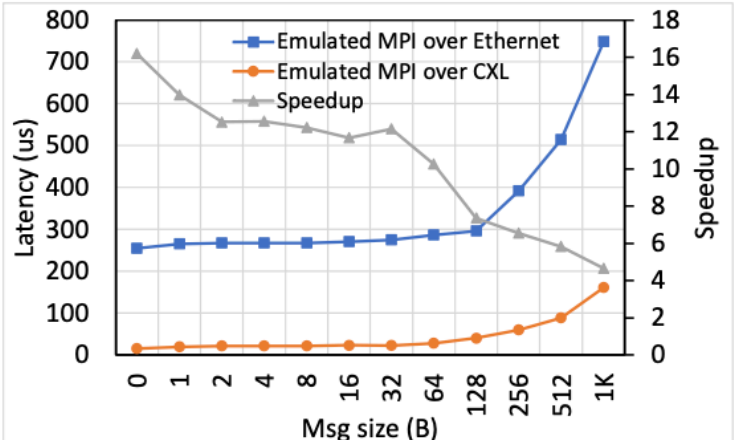
(a) HH - small message range



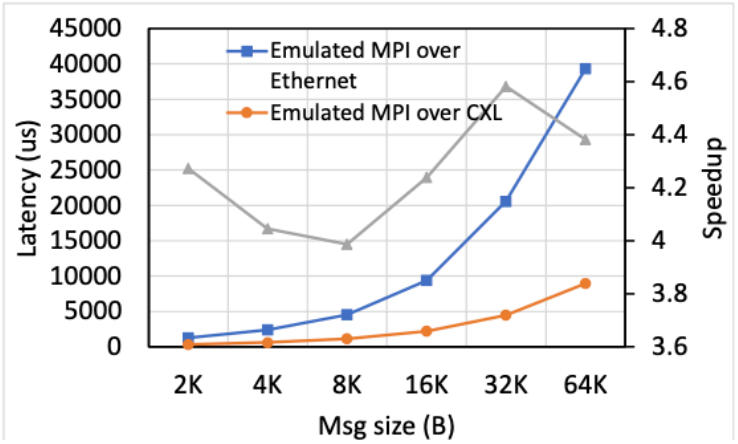
(b) HH - medium message range



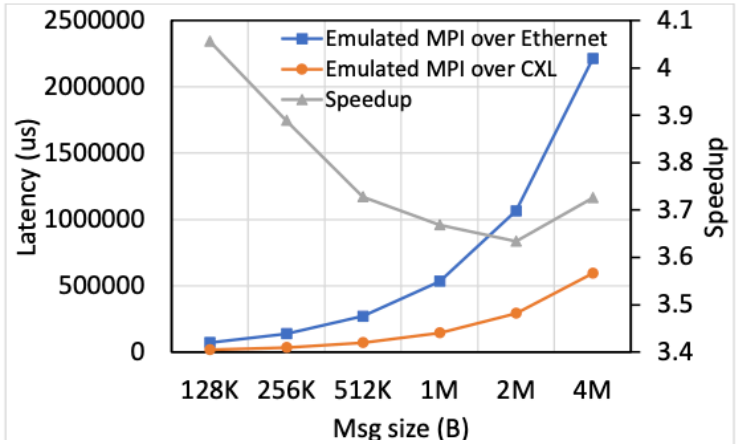
(c) HH - large message range



(d) DD - small message range



(e) DD - medium message range



(f) DD - large message range

- HH: CXL performs better for small messages. Losing in larger ones due CXL BW limitation in QEMU
- DD: CXL with shorter communication path outperforms Ethernet

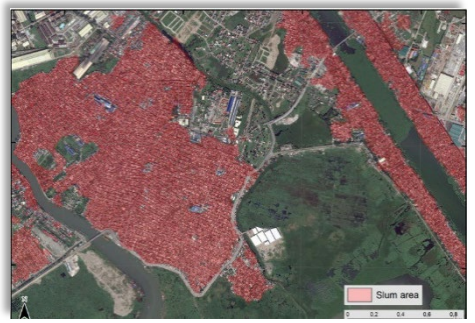
  More details in the Short Talk, presented by Tu Tran (Tomorrow 3:45-5:00 pm)

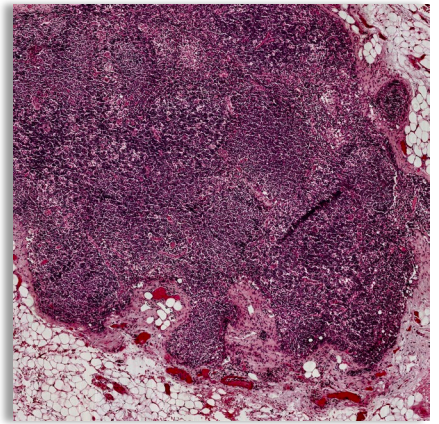
Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - **Accelerating Inference**
  - Conversational AI Interface (SAI)
- Conclusions

Introduction to High Resolution Images

Image Resolution
~100,000x100,000 pixels





- High resolution (HiRes) images typically have dimensions of 100,000×100,000 pixels (gigapixels) or higher
  - Medical Imaging, Satellite Imagery, etc
- Deep Neural Networks (DNNs) for high-resolution imaging are typically ~100M-10B parameters
  - Using a large DNN on large input HiRes images is compute- and memory-intensive.
  - For example, ResNet101 cannot scale beyond 2048×2048 or 4096×4096 image sizes on a single 40GB GPU

Problem Statement and the Proposed Solution

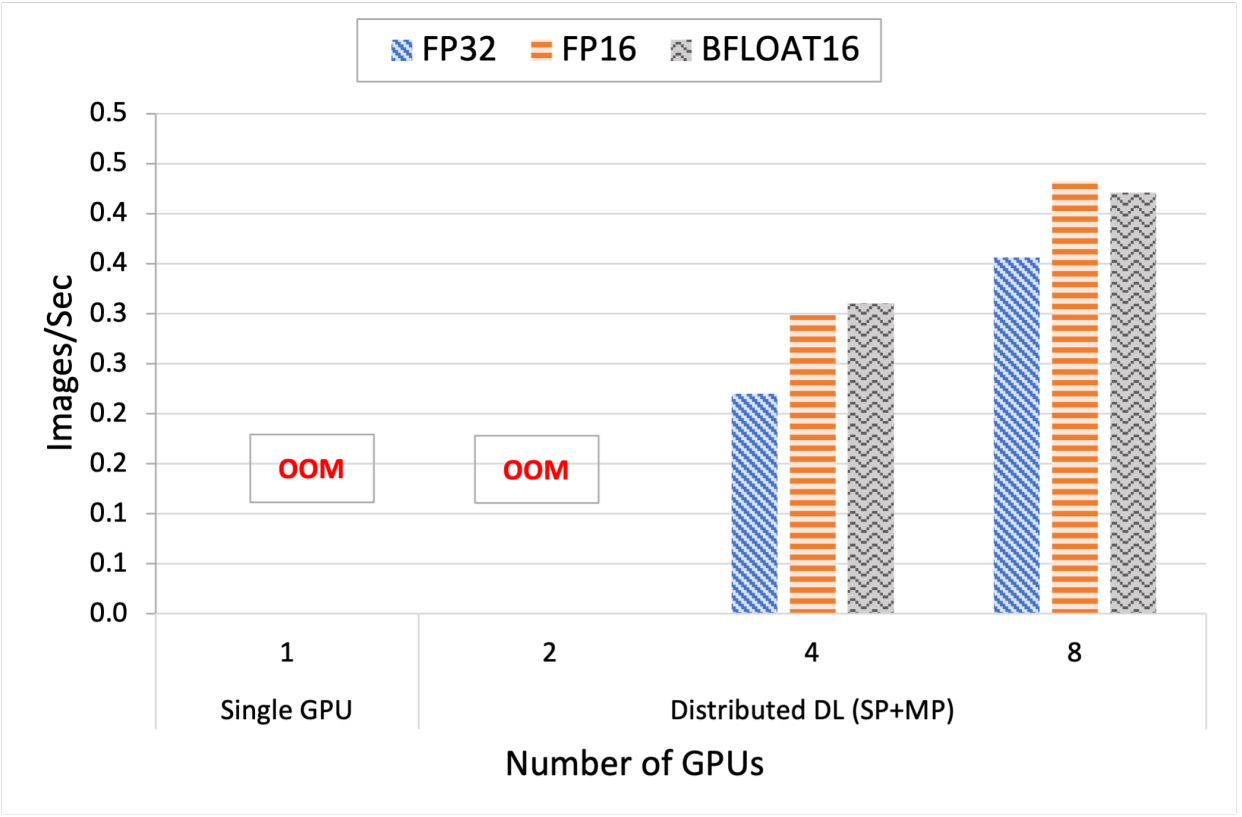
- Main issues while inferencing on HiRes images:
  - High **compute** and **memory** requirement due to large input images and large models
  - A single GPU cannot be used for inference!

- Existing literature has explored optimizing *training* on HiRes images. However, optimizing *inference* remains unexplored:
  - Most previous works focus on high-precision parallelization strategies including spatial and layer/pipeline parallelism
  - This work proposes to combine these parallelization strategies with quantization to accelerate inference for HiRes images

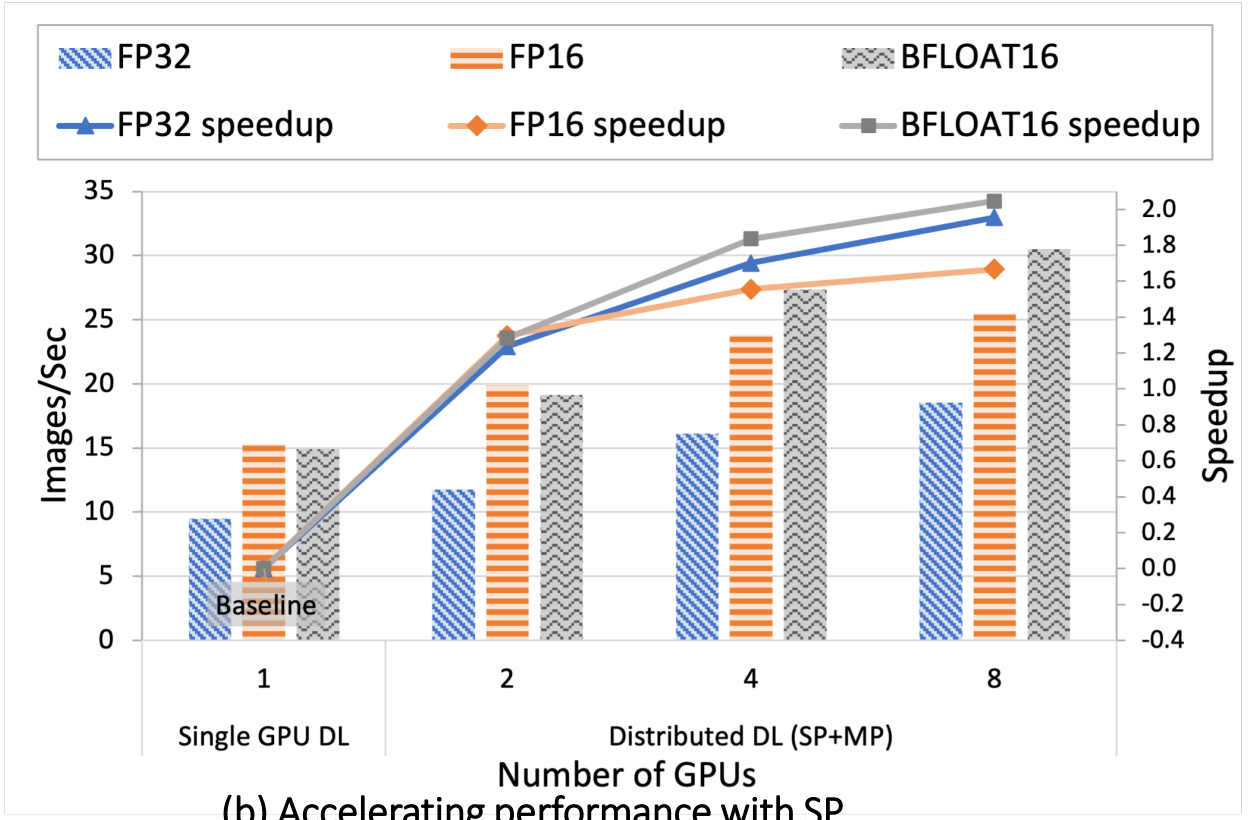
More details in the Short Talk, presented by Quentin Anthony (Yesterday 4:15-5:00 pm)

Performance Evaluation – Larger Image Sizes

- ResNet101 model for image size 8192x8192 becomes out-of-core on a 40GB GPU. SP enables inference for image size 8192x8192, further quantization improves performance
- Use of SP not only enables inference for scaled images, but also accelerate performance for smaller images when compared to single-GPU inference performance



(a) Enabling inference for 8192×8192 image with half-precision



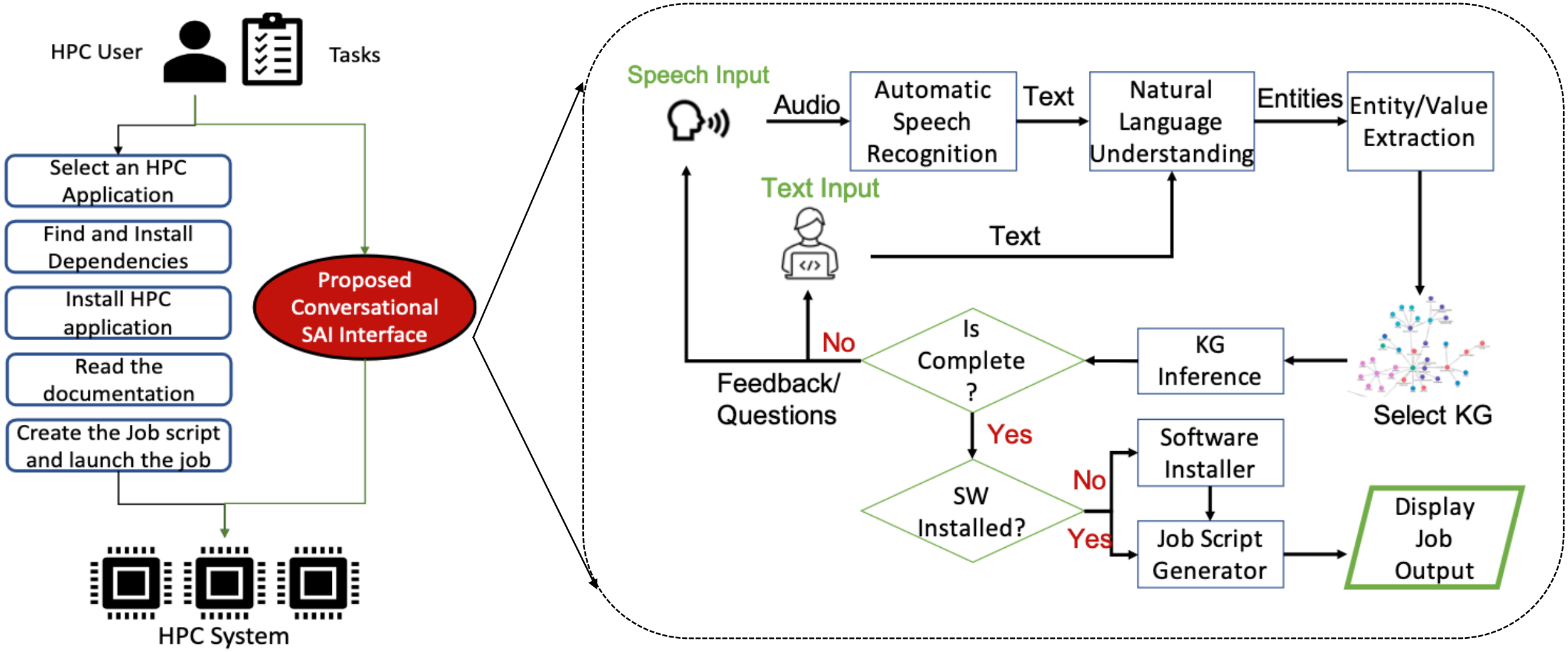
(b) Accelerating performance with SP

Throughput Evaluation for image size 2048x2048

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - **Conversational AI Interface (SAI)**
- Conclusions

Proposed Framework for Conversational AI for HPC Tasks



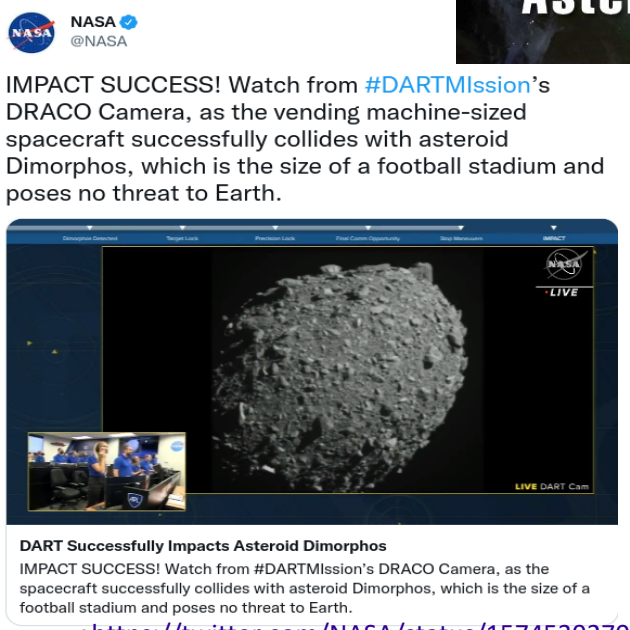
More details in the Short Talk, presented by Pouya Kousha (Yesterday 4:00-5:30 pm)

Outline

- Brief Overview of the MVAPICH Project
- New MVAPICH-Plus Series
- Features and Performance of Recent Releases
  - MVAPICH-Plus 4.0b
  - **Optimized MVAPICH2-2.3.7+ for Broadcom RoCE**
  - Optimized versions for Cloud (Azure and AWS)
  - Converged software stack based on MVAPICH-Plus
    - Support for DL (HiDL), ML (MPI4cuML), Big Data (MPI4Spark), and Data Science (MPI4Dask)
  - OSU Micro-Benchmarks (OMB)
  - InfiniBand Network Analysis and Monitoring (INAM)
  - Applications: Best Practices
- **Upcoming Features**
  - Support for AMD and Intel GPUs
  - MVAPICH and OMB for FPGA
  - CXL Support
  - Accelerating Inference
  - **Conversational AI Interface (SAI)**
- Conclusions

MVAPICH2 enabling life-changing NASA's DART mission

- Near-Earth asteroids (NEAs) have caused recent and ancient global catastrophes
  - LLNL scientists research ways to prevent NEAs using methods known as asteroid deflection
  - Joint NASA-LLNL research modelled various asteroid deflection methods (NASA's DART mission)
- MVAPICH2 lived at the core of the (NASA-DART mission) and enabled scalability
  - Underneath large-scale hydrodynamical and gravitational simulations required to compute the impact such as Spheral models

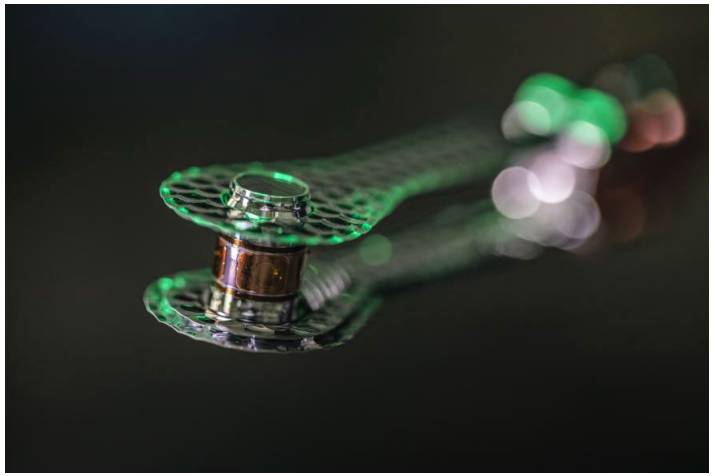


- https://twitter.com/NASA/status/1574539270987173903?s=20&t=u_4wIV9Cui2xyn9QLj286Q
- https://www.cbsnews.com/sanfrancisco/news/i-just-could-not-believe-it-livermore-team-celebrates-nasas-historic-strike-on-distant-asteroid/
- http://mug.mvapich.cse.ohio-state.edu/static/media/mug/presentations/18/moody-mug-18.pdf

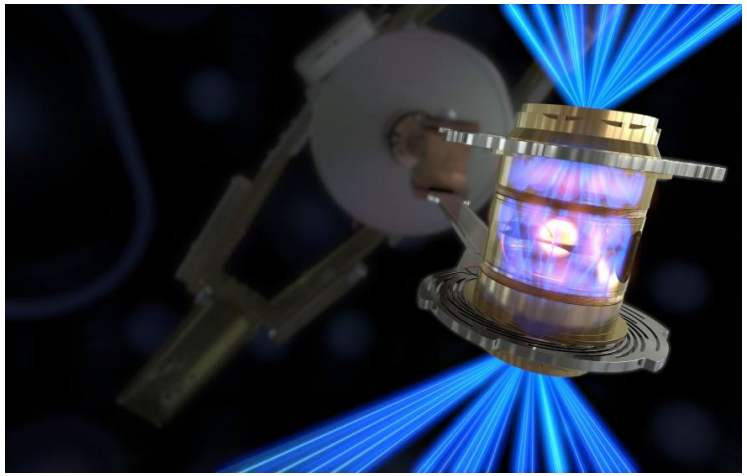
MVAPICH enabling Nuclear Fusion Research

- LLNL's National Ignition Facility (NIF) conducted the first controlled fusion experiment in history![1]
- MVAPICH, being the default MPI library on the LLNL systems, has been enabling the thousands of simulation jobs that have led to this amazing achievement!
- [1] https://www.llnl.gov/news/national-ignition-facility-achieves-fusion-ignition

The target chamber of LLNL's National Ignition Facility, where 192 laser beams delivered more than 2 million joules of ultraviolet energy to a tiny fuel pellet to create fusion ignition on Dec. 5, 2022.

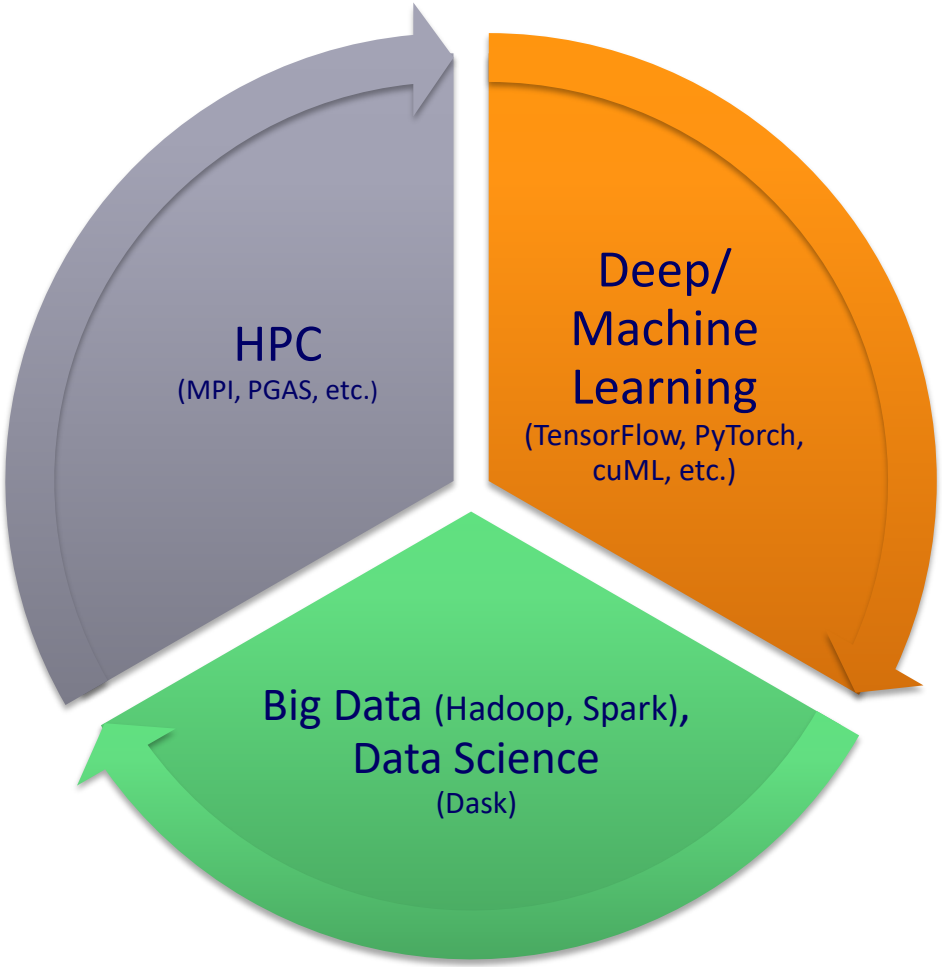


The hohlraum that houses the type of cryogenic target used to achieve ignition on Dec. 5, 2022, at LLNL's National Ignition Facility.



To create fusion ignition, the National Ignition Facility's laser energy is converted into X-rays inside the hohlraum, which then compress a fuel capsule until it implodes, creating a high temperature, high pressure plasma.

MPI-Driven Converged Software Stack for HPC, AI, Big Data and Data Science



Funding Acknowledgments

Funding Support by

Equipment Support by

Acknowledgments to all the Heroes (Past/Current Students and Staffs)

Current Students (Graduate)

- K. Al Attar (Ph.D.)
- N. Alnaasan (Ph.D.)
- Q. Anthony (Ph.D.)
- C.-C. Chun (Ph.D.)
- T. Chen
- N. Contini (Ph.D.)
- J. Hatef (Ph.D.)
- S. Lee (Ph.D.)
- B. Michalowicz (Ph.D.)
- B. Ramesh (Ph.D.)
- K. K. Suresh (Ph.D.)
- T. Tran (Ph.D.)
- A. Potlapally (Ph.D.)
- S. Xu (Ph.D.)
- L. Xu (Ph.D.)
- G. Kuncham (Ph.D.)
- J. Yao (P.hD.)
- J. Jani (M.S.)
- J. Queiser (M.S.)

Past Students

- A. Awan (Ph.D.)
- A. Augustine (M.S.)
- P. Balaji (Ph.D.)
- M. Bayatpour (Ph.D.)
- R. Biswas (M.S.)
- S. Bhagvat (M.S.)
- A. Bhat (M.S.)
- D. Buntinas (Ph.D.)
- L. Chai (Ph.D.)
- B. Chandrasekharan (M.S.)
- S. Chakraborthy (Ph.D.)
- N. Dandapanthula (M.S.)
- V. Dhanraj (M.S.)
- C.-H. Chu (Ph.D.)
- T. Gangadharappa (M.S.)
- K. Gopalakrishnan (M.S.)
- R. Gulhane (M.S.)
- J. Hashmi (Ph.D.)
- M. Han (M.S.)
- W. Huang (Ph.D.)
- A. Jain (Ph.D.)
- W. Jiang (M.S.)
- J. Jose (Ph.D.)
- M. Kedia (M.S.)
- K. S. Khorassani (Ph.D.)
- S. Kini (M.S.)
- M. Koop (Ph.D.)
- P. Kousha (Ph.D.)
- K. Kulkarni (M.S.)
- R. Kumar (M.S.)
- S. Krishnamoorthy (M.S.)
- K. Kandalla (Ph.D.)
- M. Li (Ph.D.)
- 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.)
- K. Raj (M.S.)
- R. Rajachandrasekar (Ph.D.)
- D. Shankar (Ph.D.)
- G. Santhanaraman (Ph.D.)
- N. Sarkauskas (B.S. and M.S)
- V. Sathu (M.S.)
- N. Senthil Kumar (M.S.)
- A. Singh (Ph.D.)
- J. Sridhar (M.S.)
- S. Srivastava (M.S.)
- S. Sur (Ph.D.)
- H. Subramoni (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- A. Vishnu (Ph.D.)
- J. Wu (Ph.D.)
- W. Yu (Ph.D.)
- J. Zhang (Ph.D.)
- Q. Zhou (Ph.D.)

Past Post-Docs

- D. Banerjee
- X. Besseron
- M. S. Ghazimirsaeed
- H.-W. Jin
- J. Lin
- M. Luo
- E. Mancini
- K. Manian
- S. Marcarelli
- A. Ruhela
- J. Vienne
- H. Wang

Current Research Scientists

- A. Shafi

Current Research Specialist

- R. Motlagh

Current Faculty

- H. Subramoni

Current Software Engineers

- N. Pavuk
- N. Shineman
- M. Lieber
- A. Guptha

Past Research Scientists

- K. Hamidouche
- S. Sur
- X. Lu
- M. Abduljabbar

Past Senior Research Associate

- J. Hashmi

Past Programmers

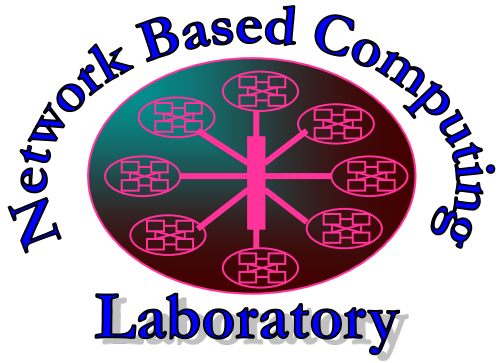
- A. Reifsteck
- D. Bureddy
- J. Perkins
- B. Seeds

Past Research Specialist

- M. Arnold
- J. Smith

Thank You!

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

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