MVAPICH2-X - High-Performance MPI and PGAS Libraries for Modern Clusters

Khaled Hamidouche

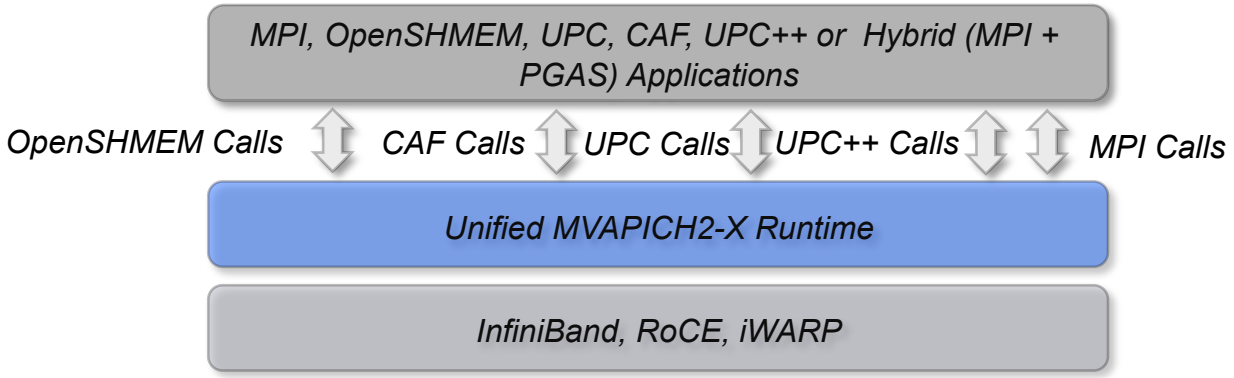
The Ohio State University

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

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

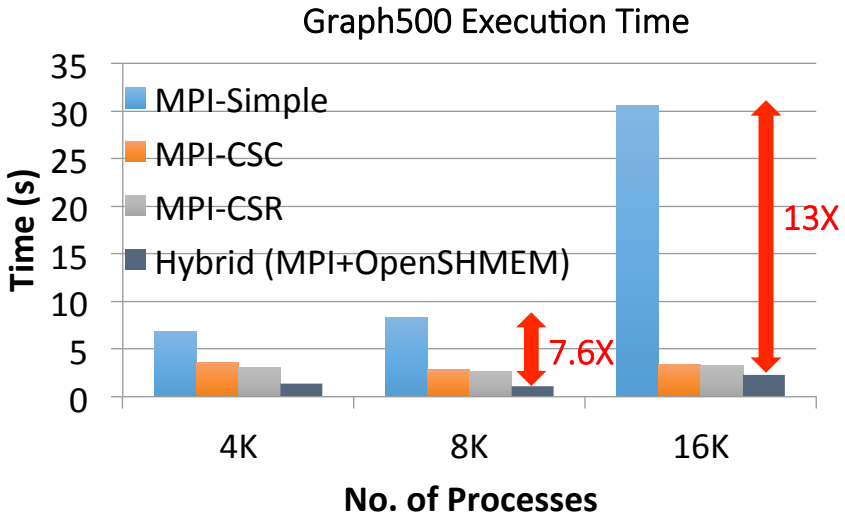
Overview of MVAPICH2 / MVAPICH2-X

- *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 2,675 organizations in 83 countries*
  - *More than 402,000 (> 0.4 million) downloads from the OSU site directly*
  - *Empowering many TOP500 clusters (Nov '16 ranking)*
    - *1st ranked 10,649,640-core cluster (Sunway TaihuLight) at NSC, Wuxi, China*
    - *13th ranked 241,108-core cluster (Pleiades) at NASA*
    - *17th ranked 519,640-core cluster (Stampede) at TACC*
    - *40th ranked 76,032-core cluster (Tsubame 2.5) at Tokyo Institute of Technology and many others*
  - *Available with software stacks of many vendors and Linux Distros (RedHat and SuSE)*
  - *http://mvapich.cse.ohio-state.edu*
- *Empowering Top500 systems for over a decade*
  - *System-X from Virginia Tech (3rd in Nov 2003, 2,200 processors, 12.25 TFlops) ->*
  - *Sunway TaihuLight at NSC, Wuxi, China (1st in Nov'16, 10,649,640 cores, 93 PFlops)*

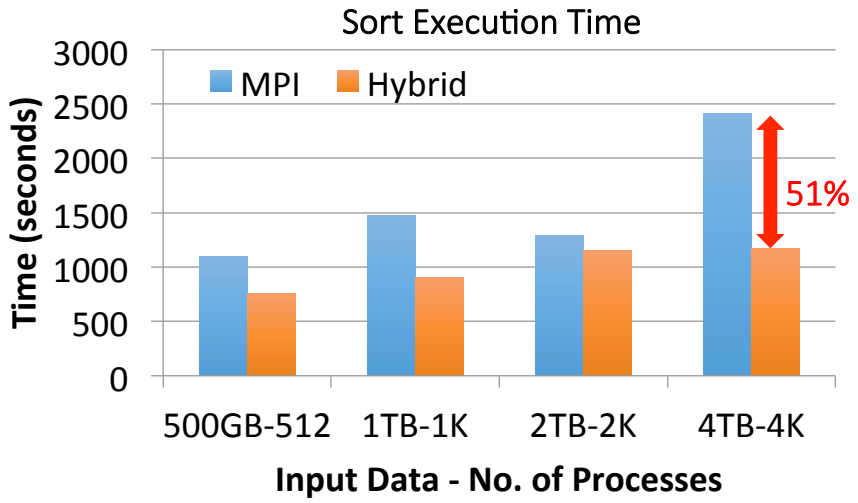


- Unified communication runtime for MPI, UPC, OpenSHMEM, CAF
- Available with MVAPICH2-X 1.9 (2012) onwards!
  - http://mvapich.cse.ohio-state.edu
- Feature Highlights
  - Supports MPI+X: OpenMP, OpenSHMEM, UPC, CAF, UPC++, MPI(+OpenMP) + OpenSHMEM, MPI(+OpenMP) + UPC + CAF
  - MPI-3 compliant, OpenSHMEM v1.0h standard compliant, UPC v1.2 standard compliant (with initial support for UPC 1.3), CAF 2008 standard (OpenUH), UPC++
  - Scalable Inter-node and intra-node communication – point-to-point and collectives

Application Level Performance with Graph500 and Sort



- *Performance of Hybrid (MPI+ OpenSHMEM) Graph500 Design*
  - *8,192 processes*
    - *2.4X improvement over MPI-CSR*
    - *7.6X improvement over MPI-Simple*
  - *16,384 processes*
    - *1.5X improvement over MPI-CSR*
    - *13X improvement over MPI-Simple*



- *Performance of Hybrid (MPI+OpenSHMEM) Sort Application*
  - *4,096 processes, 4 TB Input Size*
    - *MPI – 2408 sec; 0.16 TB/min*
    - *Hybrid – 1172 sec; 0.36 TB/min*
    - *51% improvement over MPI-design*

J. Jose, K. Kandalla, S. Potluri, J. Zhang and D. K. Panda, Optimizing Collective Communication in OpenSHMEM, Int'l Conference on Partitioned Global Address Space Programming Models (PGAS '13), October 2013.

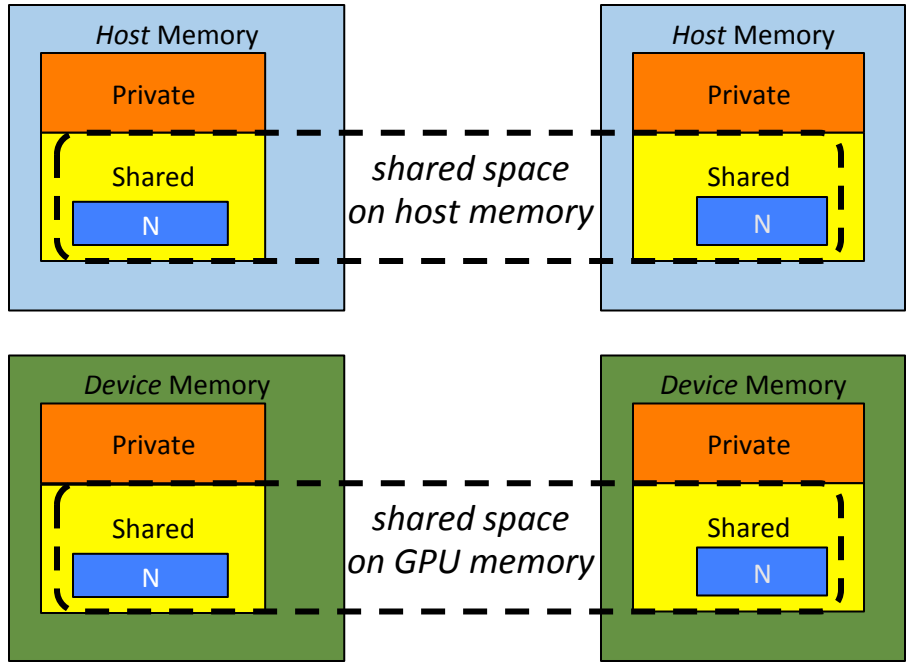
J. Jose, S. Potluri, K. Tomko and D. K. Panda, Designing Scalable Graph500 Benchmark with Hybrid MPI+OpenSHMEM Programming Models, International Supercomputing Conference (ISC'13), June 2013

J. Jose, K. Kandalla, M. Luo and D. K. Panda, Supporting Hybrid MPI and OpenSHMEM over InfiniBand: Design and Performance Evaluation, Int'l Conference on Parallel Processing (ICPP '12), September 2012

Next Step for PGAS models: Accelerator Support

Global Address Space with Host and Device Memory

- Extend Memory model for heterogeneous Memory Domains:
- heap_on_device/heap_on_host (a way to indicate location of heap)
- *host_buf = shmalloc (sizeof(int), 0); dev_buf = shmalloc (sizeof(int), 1);*



CUDA-Aware OpenSHMEM; Same extension for UPC and any other PGAS model
More extensions for efficient support for MIC systems

PE 0

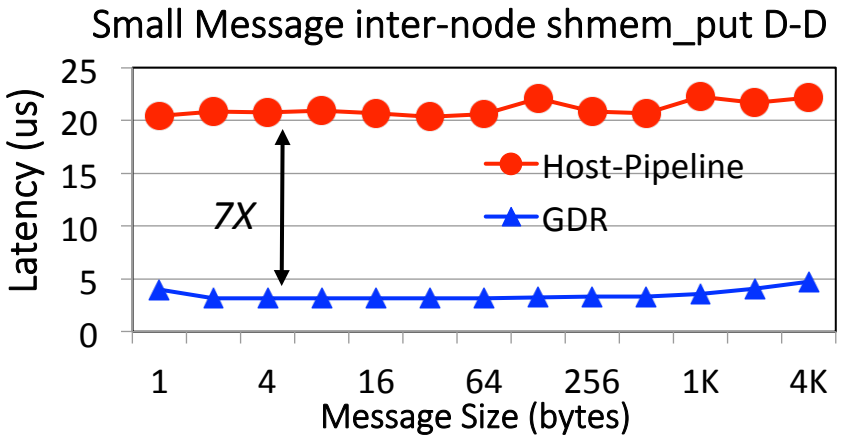
dev_buf = shmalloc (size, 1);
shmem_putmem (dev_buf, dev_buf, size, pe)

PE 1

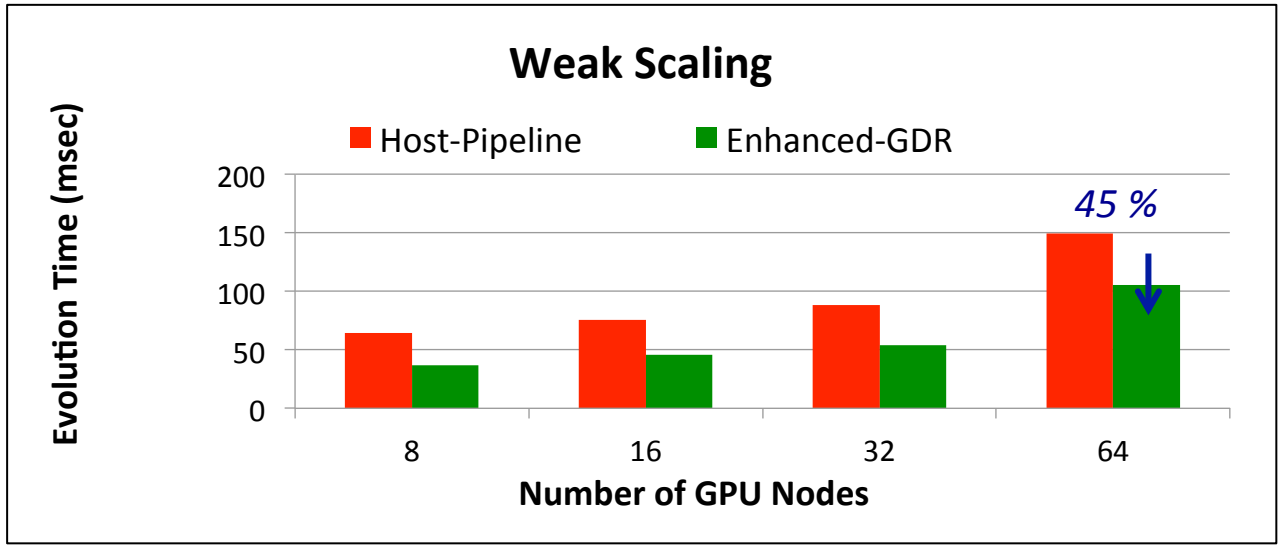
dev_buf = shmalloc (size, 1);

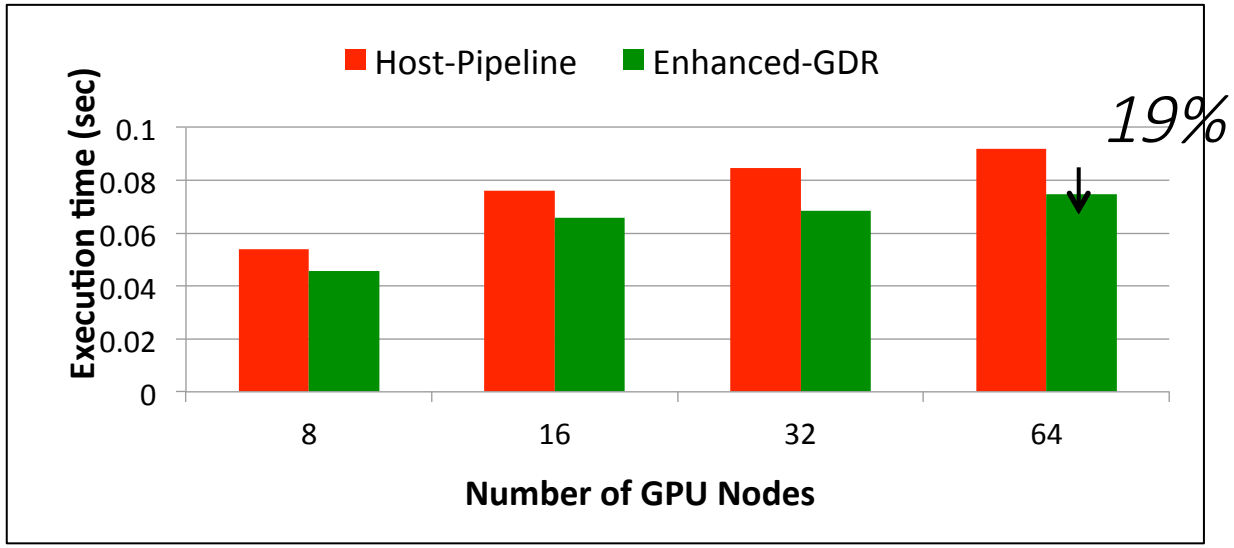
S. Potluri, D. Bureddy, H. Wang, H. Subramoni and D. K. Panda, Extending OpenSHMEM for GPU Computing, IPDPS'13

J. Jose, K. Hamidouche, X. Lu, S. Potluri, J. Zhang, K. Tomko and and D. K. Panda, High Performance OpenSHMEM for MIC Clusters: Extensions, Runtime Designs and Application Co-design IEEE CLUSTER'14 (Best Paper Nominee)



Application Evaluation: GPULBM and 2DStencil with MPI+OpenSHMEM





GPULBM: 64x64x64

2DStencil 2Kx2K

- *Redesign the application*
  - *CUDA-Aware MPI : Send/Recv=> hybrid CUDA-Aware MPI+OpenSHMEM*
  - *cudaMalloc =>shmalloc(size,1);*
  - *MPI_Send/recv => shmem_put + fence*
  - *53% and 45%*
  - *Degradation is due to small Input size*
  - *Will be available in future MVAPICH2-GDR*

- Platform: Wilkes (Intel Ivy Bridge + NVIDIA Tesla K20c + Mellanox Connect-IB)
- New designs achieve 20% and 19% improvements on 32 and 64 GPU nodes

K. Hamidouche, A. Venkatesh, A. Awan, H. Subramoni, C. Ching and D. K. Panda, Exploiting GPUDirect RDMA in Designing High Performance OpenSHMEM for GPU Clusters. IEEE Cluster 2015.

K. Hamidouche, A. Venkatesh, A. Awan, H. Subramoni, C. Ching and D. K. Panda, CUDA-Aware OpenSHMEM: Extensions and Designs for High Performance OpenSHMEM on GPU Clusters. To appear in PARCO.