Overview of the MVAPICH Project: Latest Status and Future Roadmap

MVAPICH2 User Group (MUG) Meeting

by

Dhabaleswar K. (DK) Panda
The Ohio State University
E-mail: panda@cse.ohio-state.edu
http://www.cse.ohio-state.edu/~panda
Drivers of Modern HPC Cluster Architectures

- Multi-core processors are ubiquitous
- InfiniBand very popular in HPC clusters
- Accelerators/Coprocessors becoming common in high-end systems
- Pushing the envelope for Exascale computing
Parallel Programming Models Overview

- Programming models provide abstract machine models
- Models can be mapped on different types of systems
  - e.g. Distributed Shared Memory (DSM), MPI within a node, etc.
- PGAS models and Hybrid MPI+PGAS models are gradually receiving importance
Supporting Programming Models for Multi-Petaflop and Exaflop Systems: Challenges

- Application Kernels/Applications
- Middleware
- Programming Models
  - MPI, PGAS (UPC, Global Arrays, OpenSHMEM), CUDA, OpenACC, Cilk, Hadoop, MapReduce, etc.

Communication Library or Runtime for Programming Models

- Point-to-point Communication (two-sided & one-sided)
- Collective Communication
- Synchronization & Locks
- I/O & File Systems
- Fault Tolerance

Networking Technologies (InfiniBand, 10/40GigE, Aries, OmniPath)

Multi/Many-core Architectures

Accelerators (NVIDIA and MIC)

Co-Design Opportunities and Challenges across Various Layers
Designing (MPI+X) at Exascale

- Scalability for million to billion processors
  - Support for highly-efficient inter-node and intra-node communication (both two-sided and one-sided)
  - Extremely minimum memory footprint
- Hybrid programming (MPI + OpenMP, MPI + UPC, MPI + OpenSHMEM, ...)
- Balancing intra-node and inter-node communication for next generation multi-core (128-1024 cores/node)
  - Multiple end-points per node
- Support for efficient multi-threading
- Scalable Collective communication
  - Offload
  - Non-blocking
  - Topology-aware
  - Power-aware
- Support for MPI-3 RMA Model
- Support for GPGPUs and Accelerators
- Fault-tolerance/resiliency
- QoS support for communication and I/O
- Virtualization Support
The MVAPICH2 Software Family

- High Performance open-source MPI Library for InfiniBand, 10Gig/iWARP, RDMA over Converged Enhanced Ethernet (RoCE), and virtualized clusters.
  - MVAPICH (MPI-1), Available since 2002
  - MVAPICH2 (MPI-2.2, MPI-3.0 and MPI-3.1), Available since 2004
  - MVAPICH2-X (Advanced MPI + PGAS), Available since 2012
  - Support for GPGPUs (MVAPICH2-GDR), Available since 2014
  - Support for MIC (MVAPICH2-MIC), Available since 2014
  - Support for Virtualization (MVAPICH2-Virt), Available since 2015

- [http://mvapich.cse.ohio-state.edu](http://mvapich.cse.ohio-state.edu)
MVAPICH Project Timeline

- MVAPICH2-Virt
- MVAPICH2-MIC
- MVAPICH2-GDR
- MVAPICH2-X
- OMB
- MVAPICH
- EOL

Timeline:
- Oct-02
- Jan-04
- Nov-04
- Jan-10
- Nov-12
- Aug-14
- Apr-15
- Jul-15

MVAPICH User Group Meeting 2015
MVAPICH/MVAPICH2 Release Timeline and Downloads

Number of Downloads

Timeline

MVAPICH User Group Meeting 2015
The MVAPICH2 Software Family (Cont.)

- Empowering many TOP500 clusters (July ‘15 ranking)
  - 8th ranked 519,640-core cluster (Stampede) at TACC
  - 11th ranked 185,344-core cluster (Pleiades) at NASA
  - 22nd ranked 76,032-core cluster (Tsubame 2.5) at Tokyo Institute of Technology and many others

- Used by more than 2,450 organizations in 76 countries

- More than 282,000 downloads from the OSU site directly

- Available with software stacks of many IB, HSE, and server vendors including Linux Distros (RedHat and SuSE)

- Empowering Top500 systems for over a decade
  - System-X from Virginia Tech (3rd in Nov 2003, 2,200 processors, 12.25 TFlops) ->
  - Stampede at TACC (8th in Jun’15, 462,462 cores, 5.168 Plops)
# Usage Guidelines for the MVAPICH2 Software Family

<table>
<thead>
<tr>
<th>Requirements</th>
<th>MVAPICH2 Library to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPI with IB, iWARP and RoCE</td>
<td>MVAPICH2</td>
</tr>
<tr>
<td>Advanced MPI, PGAS and MPI+PGAS with IB and RoCE</td>
<td>MVAPICH2-X</td>
</tr>
<tr>
<td>MPI with IB &amp; GPU</td>
<td>MVAPICH2-GDR</td>
</tr>
<tr>
<td>MPI with IB &amp; MIC</td>
<td>MVAPICH2-MIC</td>
</tr>
<tr>
<td>HPC Cloud with MPI &amp; IB</td>
<td>MVAPICH2-Virt</td>
</tr>
</tbody>
</table>
Strong Procedure for Design, Development and Release

- Research is done for exploring new designs
- Designs are first presented to conference/journal publications
- Best performing designs are incorporated into the codebase
- Rigorous Q&A procedure before making a release
  - Exhaustive unit testing
  - Various test procedures on diverse range of platforms and interconnects
  - Performance tuning
  - Applications-based evaluation
  - Evaluation on large-scale systems
- Even alpha and beta versions go through the above testing
MVAPICH2 2.2a

- Released on 08/18/2015
- Major Features and Enhancements
  - Based on MPICH-3.1.4
  - Support for backing on-demand UD CM information with shared memory for minimizing memory footprint
  - Dynamic identification of maximum read/atomic operations supported by HCA
  - Enabling support for intra-node communications in RoCE mode without shared memory
  - Updated to hwloc 1.11.0
  - Updated to sm_20 kernel optimizations for MPI Datatypes
  - Automatic detection and tuning for 24-core Haswell architecture
  - Enhanced startup performance
    - Support for PMI-2 based startup with SLURM
  - Checkpoint-Restart Support with DMTCP (Distributed MultiThreaded CheckPointing)
  - Enhanced communication performance for small/medium message sizes
  - Support for linking Intel Trace Analyzer and Collector
One-way Latency: MPI over IB with MVAPICH2

**Small Message Latency**

- **Latency (us)** vs **Message Size (bytes)**
- Key Metrics:
  - TrueScale-QDR - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - ConnectX-3-FDR - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - ConnectIB-Dual FDR - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - ConnectX-4-EDR - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 Back-to-back
Bandwidth: MPI over IB with MVAPICH2

**Unidirectional Bandwidth**

- **TrueScale-QDR**
  - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - \(12465\) MBytes/sec
  - \(11497\) MBytes/sec
  - \(6356\) MBytes/sec
  - \(3387\) MBytes/sec

- **ConnectX-3-FDR**
  - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - \(9000\) MBytes/sec

- **ConnectIB-DualFDR**
  - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 with IB switch
  - \(5000\) MBytes/sec

- **ConnectX-4-EDR**
  - 2.8 GHz Deca-core (IvyBridge) Intel PCI Gen3 Back-to-back
  - \(10000\) MBytes/sec

**Bidirectional Bandwidth**

- **TrueScale-QDR**
  - \(24353\) MBytes/sec

- **ConnectX-3-FDR**
  - \(22909\) MBytes/sec

- **ConnectIB-DualFDR**
  - \(12161\) MBytes/sec

- **ConnectX-4-EDR**
  - \(6308\) MBytes/sec

**Message Size (bytes)**

- 4, 16, 64, 256, 1024, 4K, 16K, 64K, 256K, 1M
MVAPICH2 Two-Sided Intra-Node Performance
(Shared memory and Kernel-based Zero-copy Support (LiMIC and CMA))

Latest MVAPICH2 2.2a
Intel Ivy-bridge
MVAPICH2-X for Advanced MPI, Hybrid MPI + PGAS Applications

- Current Model – Separate Runtimes for OpenSHMEM/UPC/CAF and MPI
  - Possible deadlock if both runtimes are not progressed
  - Consumes more network resource
- Unified communication runtime for MPI, UPC, OpenSHMEM, CAF available with MVAPICH2-X 1.9 onwards!
  - http://mvapich.cse.ohio-state.edu
Hybrid (MPI+PGAS) Programming

• Application sub-kernels can be re-written in MPI/PGAS based on communication characteristics

• Benefits:
  – Best of Distributed Computing Model
  – Best of Shared Memory Computing Model

• Exascale Roadmap*:
  – “Hybrid Programming is a practical way to program exascale systems”

MVAPICH2-X 2.2a

- Released on 08/18/2015
- MVAPICH2-X-2.2a Feature Highlights
  - Based on MVAPICH2 2.2a including MPI-3 features
    - Compliant with UPC 2.20.2, OpenSHMEM v1.0h and CAF 3.0.39
  - MPI (Advanced) Features
    - Support for Dynamically Connected (DC) transport protocol
      - Available for pt-to-pt, RMA and collectives - Support for Hybrid mode with RC/DC/UD/XRC
    - Support for Core-Direct based Non-blocking collectives
      - Available for Ibcast, Ibarrier, Iscatter, Igather, Ialltoall and Iallgather
  - OpenSHMEM Features
    - Support for RoCE - Support for Dynamically Connected (DC) transport protocol
  - UPC Features
    - Based on Berkeley UPC 2.20.2 (contains changes/additions in preparation for upcoming UPC 1.3 specification)
    - Support for RoCE - Support for Dynamically Connected (DC) transport protocol –
  - CAF Features
    - Support for RoCE
    - Support for Dynamically Connected (DC) transport protocol
Minimizing Memory Footprint further by DC Transport

- Constant connection cost (One QP for any peer)
- Full Feature Set (RDMA, Atomics etc)
- Separate objects for send (DC Initiator) and receive (DC Target)
  - DC Target identified by “DCT Number”
  - Messages routed with (DCT Number, LID)
  - Requires same “DC Key” to enable communication
- Available with MVAPICH2-X 2.2a

**Memory Footprint for Alltoall**

<table>
<thead>
<tr>
<th>Number of Processes</th>
<th>RC</th>
<th>DC-Pool</th>
<th>UD</th>
<th>XRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>10</td>
<td>10</td>
<td>22</td>
<td>1</td>
</tr>
<tr>
<td>160</td>
<td>10</td>
<td>10</td>
<td>47</td>
<td>1</td>
</tr>
<tr>
<td>320</td>
<td>10</td>
<td>10</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>640</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

**NAMD - Apoa1: Large data set**

<table>
<thead>
<tr>
<th>Number of Processes</th>
<th>RC</th>
<th>DC-Pool</th>
<th>UD</th>
<th>XRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>320</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>640</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>


MVAPICH User Group Meeting 2015
### Application Level Performance with Graph500 and Sort

**Graph500 Execution Time**

- **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

**Sort Execution Time**

- **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

---


MVAPICH User Group Meeting 2015
GPU-Direct RDMA (GDR) with CUDA

- OFED with support for GPUDirect RDMA is developed by NVIDIA and Mellanox
- OSU has a design of MVAPICH2 using GPUDirect RDMA
  - Hybrid design using GPU-Direct RDMA
    - GPUDirect RDMA and Host-based pipelining
    - Alleviates P2P bandwidth bottlenecks on SandyBridge and IvyBridge
  - Support for communication using multi-rail
  - Support for Mellanox Connect-IB and ConnectX VPI adapters
  - Support for RoCE with Mellanox ConnectX VPI adapters
MVAPICH2-GDR 2.1rc2

- Released on 06/24/2015

- Major Features and Enhancements
  - Based on MVAPICH2-2.1rc2
  - CUDA 7.0 compatibility
  - CUDA-Aware support for MPI_Rsend and MPI_Irsend primitives
  - Parallel intranode communication channels (shared memory for H-H and GDR for D-D)
  - Optimized H-H, H-D and D-H communication
  - Optimized intranode D-D communication
  - Optimization and tuning for point-point and collective operations
  - Update to sm_20 kernel optimization for Datatype processing
  - Optimized design for GPU based small message transfers
  - Adding R3 support for GPU based packetized transfer
  - Enhanced performance for small message host-to-device transfers
  - Support for MPI_Scan and MPI_Exscan collective operations from GPU buffers
  - Optimization of collectives with new copy designs
Performance of MVAPICH2-GDR with GPU-Direct-RDMA

MVAPICH2-GDR-2.1RC2
Intel Ivy Bridge (E5-2680 v2) node - 20 cores
NVIDIA Tesla K40c GPU
Mellanox Connect-IB Dual-FDR HCA
CUDA 7
Mellanox OFED 2.4 with GPU-Direct-RDMA
Application-Level Evaluation (HOOMD-blue)

- 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
MPI3-RMA Performance of MVAPICH2-GDR with GPU-Direct-RDMA

GPU-GPU Internode MPI Put latency (RMA put operation Device to Device)

MPI-3 RMA provides flexible synchronization and completion primitives

![Graph showing small message latency](image)

**MVAPICH2-GDR-2.1RC2**
- Intel Ivy Bridge (E5-2680 v2) node with 20 cores
- NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
- CUDA 7, Mellanox OFED 2.4 with GPU-Direct-RDMA
MPI Applications on MIC Clusters

- MPI (+X) continues to be the predominant programming model in HPC
- Flexibility in launching MPI jobs on clusters with Xeon Phi

Multi-core Centric

Host-only

Offload (/reverse Offload)

Symmetric

Copr...
MVAPICH2-MIC Design for Clusters with IB and MIC

- Offload Mode
- Intranode Communication
  - Coprocessor-only Mode
  - Symmetric Mode
- Internode Communication
  - Coprocessors-only
  - Symmetric Mode
- Multi-MIC Node Configurations
MVAPICH2-MIC 2.0

• Released on 12/02/2014

• Major Features and Enhancements
  – Based on MVAPICH2 2.0.1
  – Support for native, symmetric and offload modes of MIC usage
  – Optimized intra-MIC communication using SCIF and shared-memory channels
  – Optimized intra-Node Host-to-MIC communication using SCIF and IB channels
  – Enhanced mpirun_rsh to launch jobs in symmetric mode from the host
  – Support for proxy-based communication for inter-node transfers
    • Active-proxy, 1-hop and 2-hop designs (actively using host CPU)
    • Passive-proxy (passively using host CPU)
  – Support for MIC-aware MPI_Bcast()
  – Improved SCIF performance for pipelined communication
  – Optimized shared-memory communication performance for single-MIC jobs
  – Supports an explicit CPU-binding mechanism for MIC processes
  – Tuned pt-to-pt intra-MIC, intra-node, and inter-node transfers
  – Supports hwloc v1.9

• Running on three major systems
  – Stampede
  – Blueridge (Virginia Tech)
  – Beacon (UTK)
MIC-Remote-MIC P2P Communication with Proxy-based Communication

Latency (Large Messages)

Intra-socket P2P

Message Size (Bytes)

Bandwidth

Message Size (Bytes)

Inter-socket P2P

Message Size (Bytes)

Bandwidth

Message Size (Bytes)
Optimized MPI Collectives for MIC Clusters (Allgather & Alltoall)

32-Node-Allgather (16H + 16 M) Small Message Latency

- **MV2-MIC**
- **MV2-MIC-Opt**

32-Node-Alltoall (8H + 8 M) Large Message Latency

- **MV2-MIC**
- **MV2-MIC-Opt**

P3DFFT Performance

- **Communication**
- **Computation**


MVAPICH User Group Meeting 2015
MVAPICH2-Virt 2.1rc2

- Enables high-performance communication in virtualized environments
  - SR-IOV based communication for Inter-Node MPI communication
  - Inter-VM Shared Memory (IVSHMEM) based communication for Intra-Node-Inter-VM MPI communication
  - [http://mvapich.cse.ohio-state.edu](http://mvapich.cse.ohio-state.edu)
MVAPICH2-Virt 2.1rc2

- Released on 06/26/2015
- Major Features and Enhancements
  - Based on MVAPICH2 2.1rc2
  - Support for efficient MPI communication over SR-IOV enabled InfiniBand network
  - High-performance and locality-aware MPI communication with IVSHMEM
  - Support for IVSHMEM device auto-detection in virtual machine
  - Automatic communication channel selection among SR-IOV, IVSHMEM, and CMA/LiMIC2
  - Support for easy configuration through runtime parameters
  - Tested with - Mellanox InfiniBand adapters (ConnectX-3 (56Gbps))
Application-Level Performance (8 VM * 8 Core/VM)

- Compared to Native, 1-9% overhead for NAS
- Compared to Native, 4-9% overhead for Graph500
OSU Micro-Benchmarks (OMB)

• Started in 2004 and continuing steadily

• Allows MPI developers and users to
  – Test and evaluate MPI libraries

• Has a wide-range of benchmarks
  – Two-sided (MPI-1, MPI-2 and MPI-3)
  – One-sided (MPI-2 and MPI-3)
  – RMA (MPI-3)
  – Collectives (MPI-1, MPI-2 and MPI-3)
  – Extensions for GPU-aware communication (CUDA and OpenACC)
  – UPC (Pt-to-Pt)
  – OpenSHMEM (Pt-to-Pt and Collectives)
  – Startup

• Widely-used in the MPI community
OSU Microbenchmarks v5.0

- OSU Micro-Benchmarks 5.0 (08/18/15)
- Non-blocking collective benchmarks
  - osu_iallgather, osu_ialltoall, osu_ibARRIER, osu_ibcast, osu_igather, and osu_iscatter
  - Benchmarks can display the amount of achievable overlap
    - Overlap defined as the amount of computation that can be performed while the communication progresses in the background
    - Have the additional option: "-t" set the number of MPI_Test() calls during the dummy computation
      - set CALLS to 100, 1000, or any number > 0
- Startup benchmarks
  - osu_init
    - Measures the time taken for an MPI to complete MPI_Init
  - osu_hello
    - Measures the time taken for an MPI library to complete a simple hello world MPI program
Overview of OSU INAM

• OSU INAM monitors IB clusters in real time by querying various subnet management entities in the network

• Major features of the OSU INAM tool include:
  – Analyze and profile network-level activities with many parameters (data and errors) at user specified granularity
  – Capability to analyze and profile node-level, job-level and process-level activities for MPI communication (pt-to-pt, collectives and RMA)
  – Remotely monitor CPU utilization of MPI processes at user specified granularity
  – Visualize the data transfer happening in a "live" fashion - Live View for
    • Entire Network - Live Network Level View
    • Particular Job - Live Job Level View
    • One or multiple Nodes - Live Node Level View
  – Capability to visualize data transfer that happened in the network at a time duration in the past - Historical View for
    • Entire Network - Historical Network Level View
    • Particular Job - Historical Job Level View
    • One or multiple Nodes - Historical Node Level View
OSU INAM – Network Level View

- Show network topology of large clusters
- Visualize traffic pattern on different links
- Quickly identify congested links/links in error state
- See the history unfold – play back historical state of the network
OSU INAM – Job and Node Level Views

- **Job level view**
  - Show different network metrics (load, error, etc.) for any live job
  - Play back historical data for completed jobs to identify bottlenecks

- **Node level view provides details per process or per node**
  - CPU utilization for each rank/node
  - Bytes sent/received for MPI operations (pt-to-pt, collective, RMA)
  - Network metrics (e.g. XmitDiscard, RcvError) per rank/node
MVAPICH2-EA & OEMT

• MVAPICH2-EA (Energy-Aware)
  • A white-box approach
  • New Energy-Efficient communication protocols for pt-pt and collective operations
  • Intelligently apply the appropriate Energy saving techniques
  • Application oblivious energy saving

• OEMT
  • A library utility to measure energy consumption for MPI applications
  • Works with all MPI runtimes
  • PRELOAD option for precompiled applications
  • Does not require ROOT permission:
    • A safe kernel module to read only a subset of MSRs
MV2-EA : Application Oblivious Energy-Aware-MPI (EAM)

- An energy efficient runtime that provides energy savings without application knowledge
- Uses automatically and transparently the best energy lever
- Provides guarantees on maximum degradation with 5-41% savings at <= 5% degradation
- Pessimistic MPI applies energy reduction lever to each MPI call

MVAPICH2 – Plans for Exascale

- Performance and Memory scalability toward 500K-1M cores
  - Dynamically Connected Transport (DCT) service with Connect-IB
- Hybrid programming (MPI + OpenSHMEM, MPI + UPC, MPI + CAF ...)
- Enhanced Optimization for GPU Support and Accelerators
- Taking advantage of advanced features
  - User Mode Memory Registration (UMR)
  - On-demand Paging
- Enhanced Inter-node and Intra-node communication schemes for upcoming OmniPath enabled Knights Landing architectures
- Extended RMA support (as in MPI 3.0)
- Extended topology-aware collectives
- Power-aware point-to-point (one-sided and two-sided) and collectives
- Extended Support for MPI Tools Interface (as in MPI 3.0)
- Extended Checkpoint-Restart and migration support with SCR
Web Pointers

NOWLAB Web Page
http://nowlab.cse.ohio-state.edu

MVAPICH Web Page
http://mvapich.cse.ohio-state.edu