



High Performance, Scalable and Fault-Tolerant MPI over InfiniBand: An Overview of MVAPICH/MVAPICH2 Project

Talk at Tsukuba University

by

Dhabaleswar K. (DK) Panda

The Ohio State University

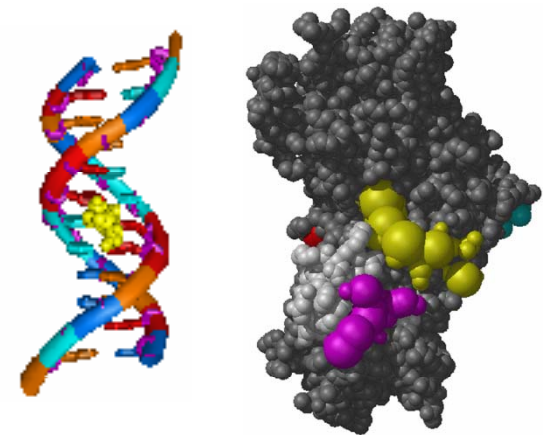
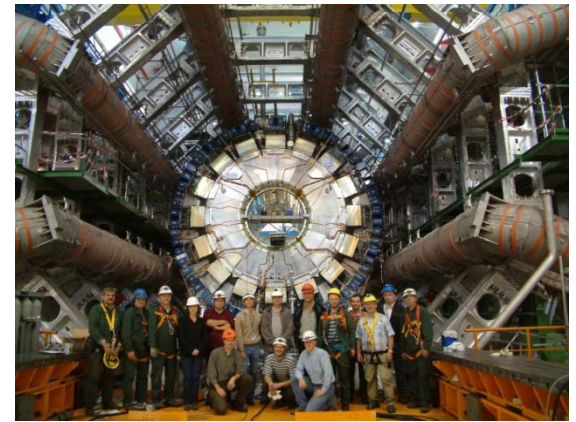
E-mail: panda@cse.ohio-state.edu

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



Current and Next Generation Applications and Computing Systems

- Big demand for
 - High Performance Computing (HPC)
 - File-systems, multimedia, database, visualization
 - Internet data-centers
- Processor performance continues to grow
 - Chip density doubling every 18 months
 - Multi-core chips are emerging
- Commodity networking also continues to grow
 - Increase in speed and features
 - Affordable pricing
- Clusters are increasingly becoming popular to design next generation computing systems
 - Scalability, Modularity and Upgradeability with compute and network technologies



Trends for Computing Clusters in the Top 500 List

- Top 500 list of Supercomputers (www.top500.org)

June 2001: 33/500 (6.6%)	June 2005: 304/500 (60.8%)
Nov 2001: 43/500 (8.6%)	Nov 2005: 360/500 (72.0%)
June 2002: 80/500 (16%)	June 2006: 364/500 (72.8%)
Nov 2002: 93/500 (18.6%)	Nov 2006: 361/500 (72.2%)
June 2003: 149/500 (29.8%)	June 2007: 373/500 (74.6%)
Nov 2003: 208/500 (41.6%)	Nov 2007: 406/500 (81.2%)
June 2004: 291/500 (58.2%)	June 2008: 400/500 (80.0%)
Nov 2004: 294/500 (58.8%)	Nov 2008: To be Announced

Growth in Commodity Network Technology

Representative commodity networks; their entries into the market

Ethernet (1979 -)	10 Mbit/sec
Fast Ethernet (1993 -)	100 Mbit/sec
Gigabit Ethernet (1995 -)	1000 Mbit /sec
ATM (1995 -)	155/622/1024 Mbit/sec
Myrinet (1993 -)	1 Gbit/sec
Fibre Channel (1994 -)	1 Gbit/sec
InfiniBand (2001 -)	2 Gbit/sec (1X SDR)
10-Gigabit Ethernet (2001 -)	10 Gbit/sec
InfiniBand (2003 -)	8 Gbit/sec (4X SDR)
InfiniBand (2005 -)	16 Gbit/sec (4X DDR)
	24 Gbit/sec (12X SDR)
InfiniBand (2007 -)	32 Gbit/sec (4X QDR)

16 times in the last 7 years

Tsukuba, Oct 2, 2008

Limitations of Traditional Host-based Protocols

- Ex: TCP/IP, UDP/IP
- Generic architecture for all network interfaces
- Host-handles almost all aspects of communication
 - Data buffering (copies on sender and receiver)
 - Data integrity (checksum)
 - Routing aspects (IP routing)
- Signaling between different layers
 - Hardware interrupt whenever a packet arrives or is sent
 - Software signals between different layers to handle protocol processing in different priority levels

Previous High Performance Network Stacks

- Virtual Interface Architecture
 - Standardized by Intel, Compaq, Microsoft
- Fast Messages (FM)
 - Developed by UIUC
- Myricom GM
 - Proprietary protocol stack from Myricom
- These network stacks set the trend for high-performance communication requirements
 - Hardware offloaded protocol stack
 - Support for fast and secure user-level access to the protocol stack

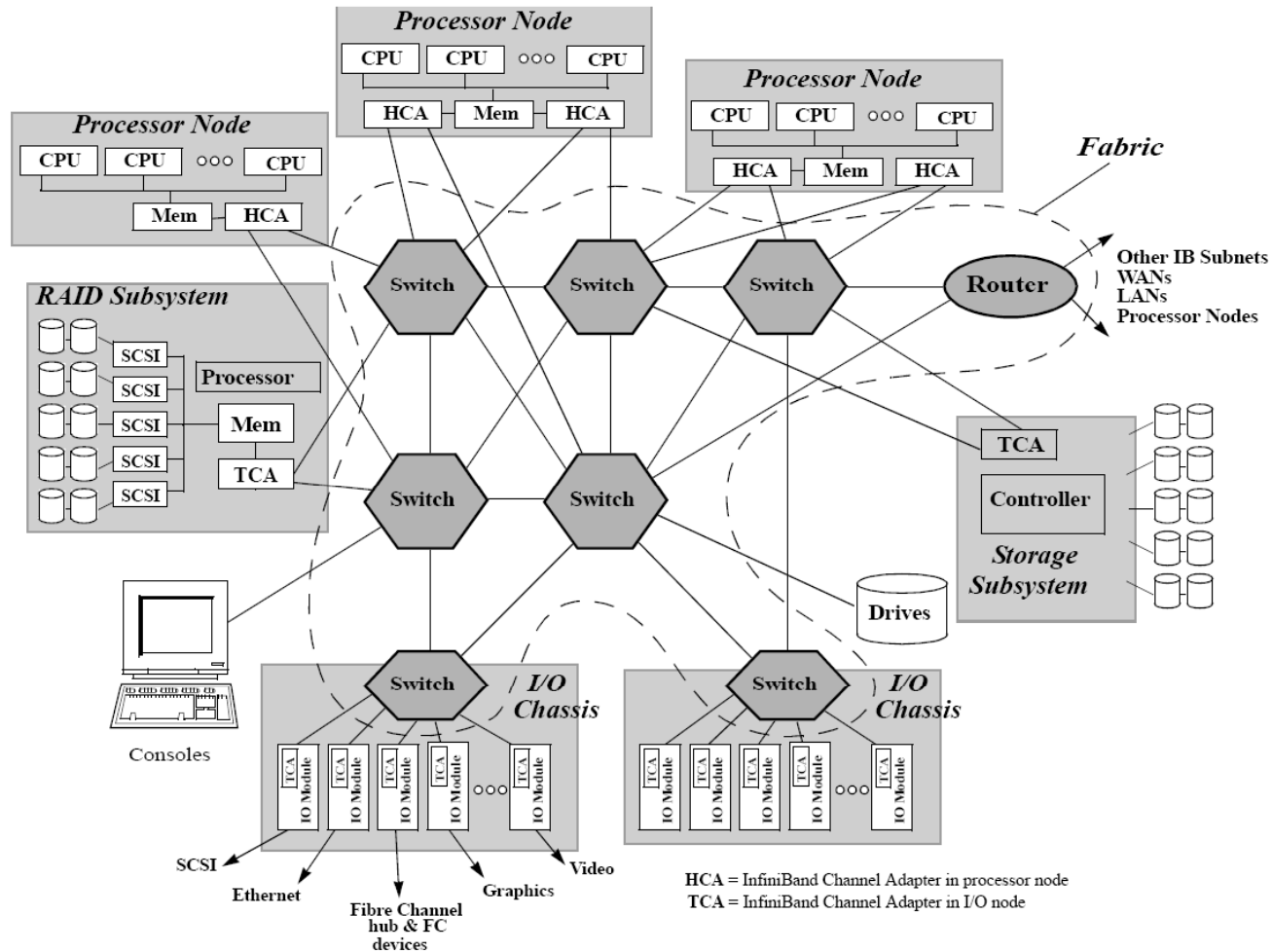
IB Trade Association

- IB Trade Association was formed with seven industry leaders (Compaq, Dell, HP, IBM, Intel, Microsoft, and Sun)
- *Goal: To design a scalable and high performance communication and I/O architecture by taking an integrated view of computing, networking, and storage technologies*
- Many other industry participated in the effort to define the IB architecture specification
- IB Architecture (Volume 1, Version 1.0) was released to public on Oct 24, 2000
 - Latest version 1.2.1 released January 2008
- <http://www.infinibandta.org>

Presentation Overview

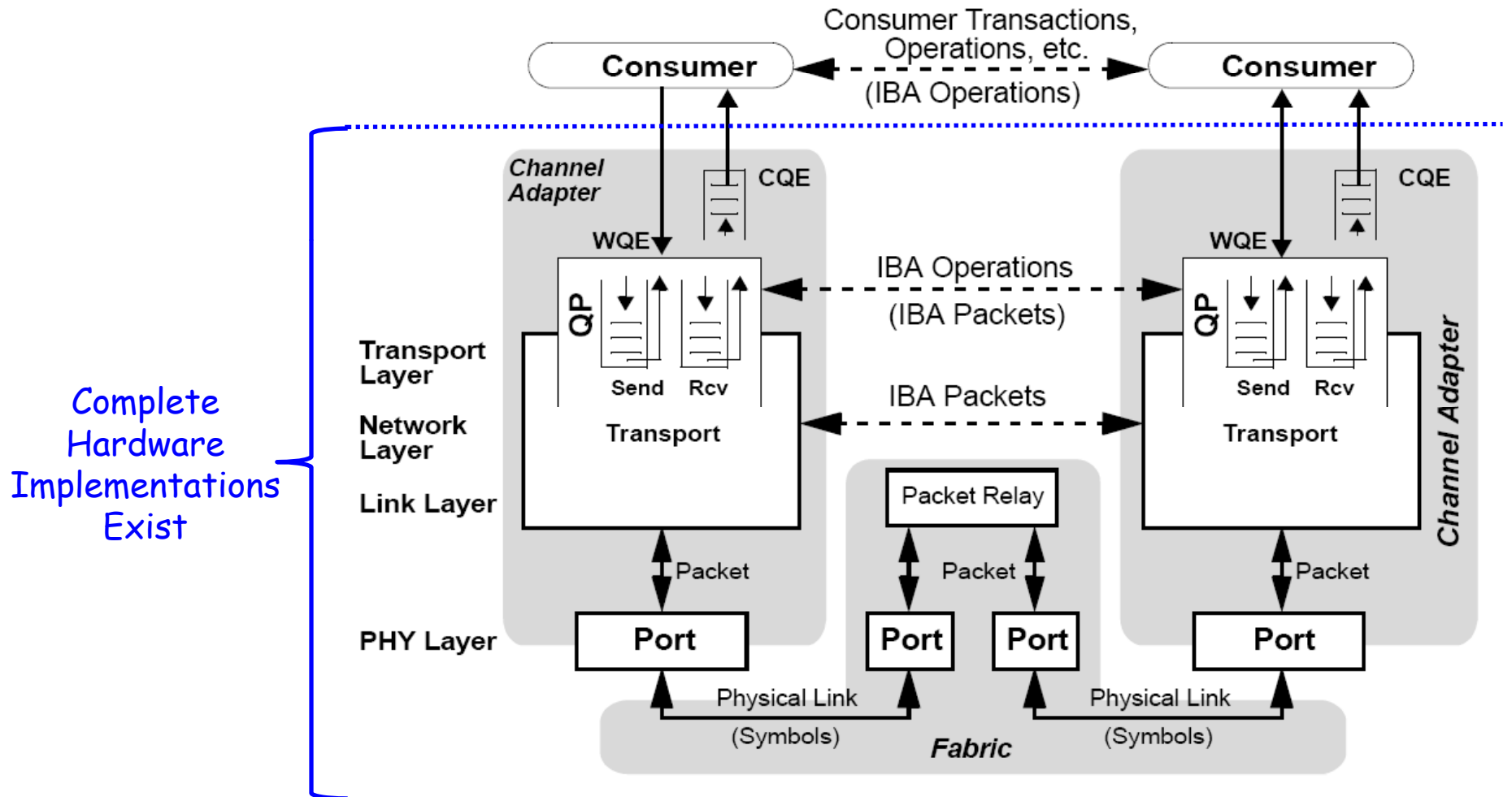
- Overview of InfiniBand
 - Features
 - Products (Hardware and Software)
 - Trends
- MVAPICH and MVAPICH2 Features
- Design Insights and Sample Performance Numbers
- Future Plans
- Conclusions and Final Q&A

A Typical IB Network



- Three primary components**
- Channel Adapters
 - Switches/Routers
 - Links and connectors

Hardware Protocol Offload



Basic IB Capabilities at Each Protocol Layer

- Link Layer
 - CRC-based data integrity, Buffering and Flow-control, Virtual Lanes, Service Levels and QoS, Switching and Multicast, WAN capabilities
- Network Layer
 - Routing and Flow Labels
- Transport Layer
 - Reliable Connection, Unreliable Datagram, Reliable Datagram and Unreliable Connection
 - Shared Receive Queued and Extended Reliable Connections (discussed in more detail later)

Communication and Management Semantics

- Two forms of communication semantics
 - Channel semantics (Send/Recv)
 - Memory semantics (RDMA, Atomic operations)
- Management model
 - A detailed management model complete with managers, agents, messages and protocols
- Verbs Interface
 - A low-level programming interface for performing communication as well as management

Communication in the Channel Semantics (Send-Receive Model)

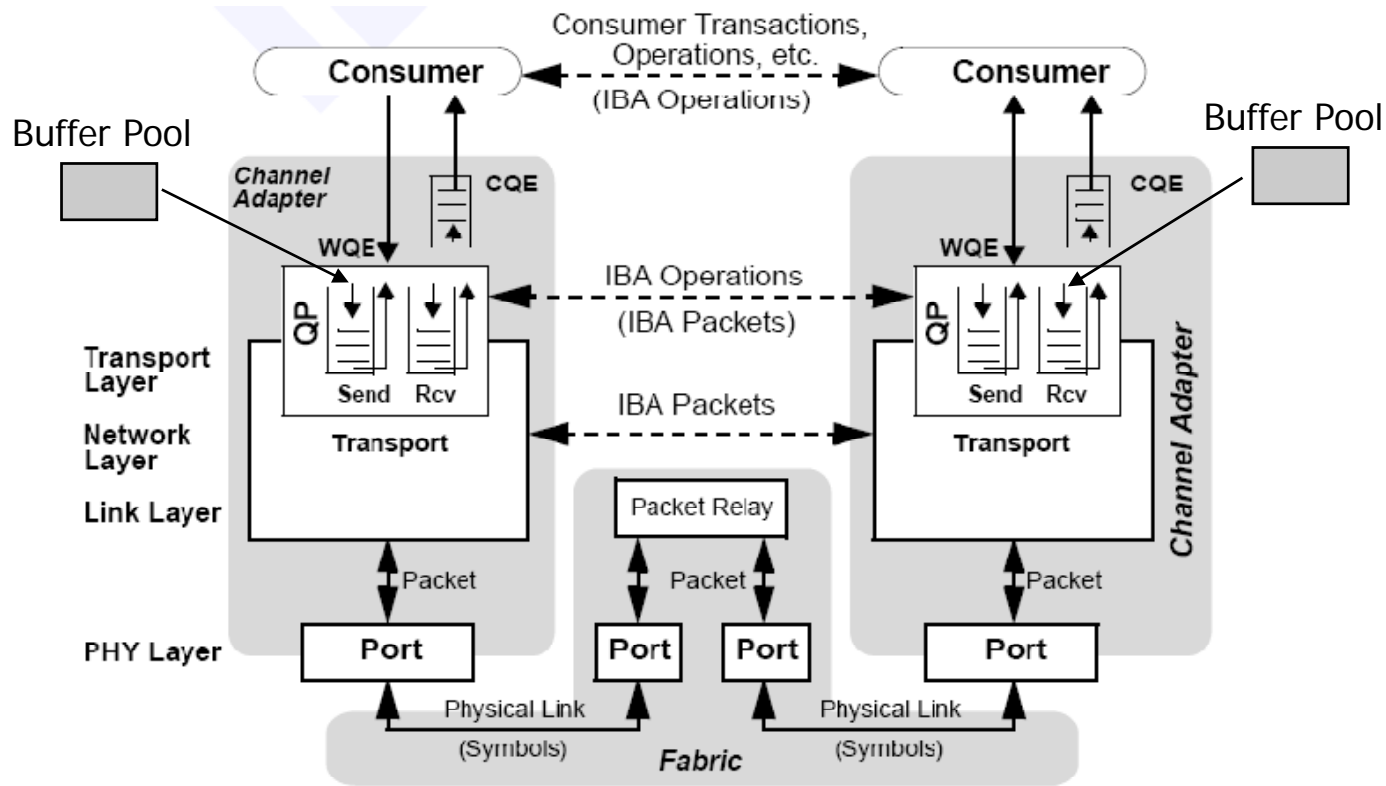
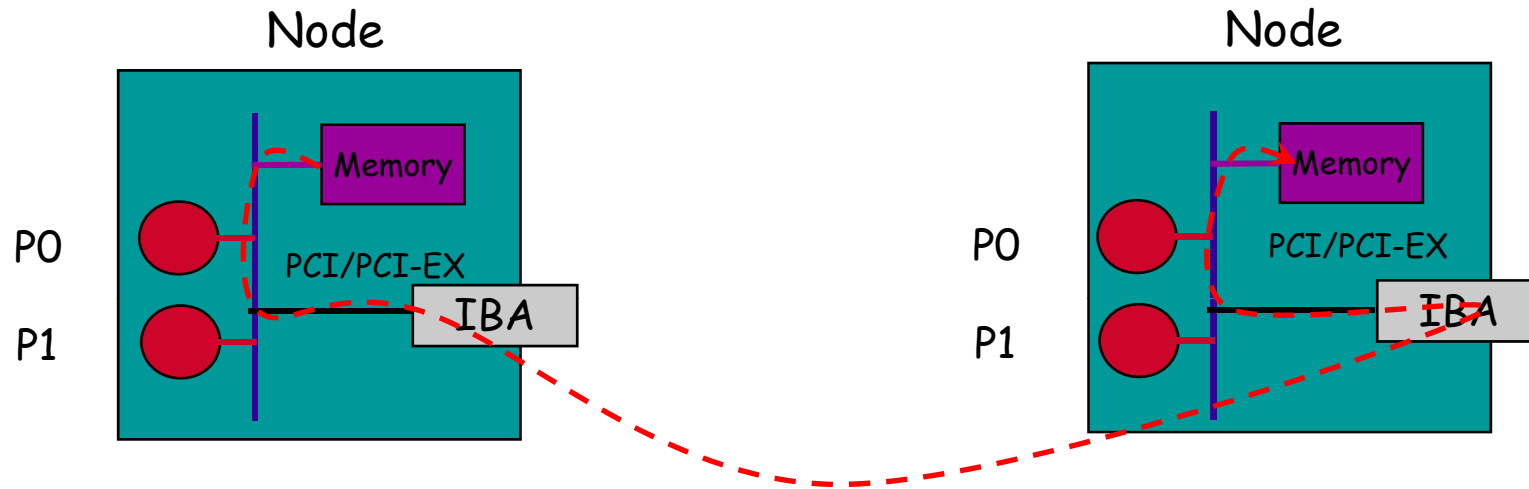


Figure 13 IBA Communication Stack

Communication in the Memory Semantics (RDMA Model)



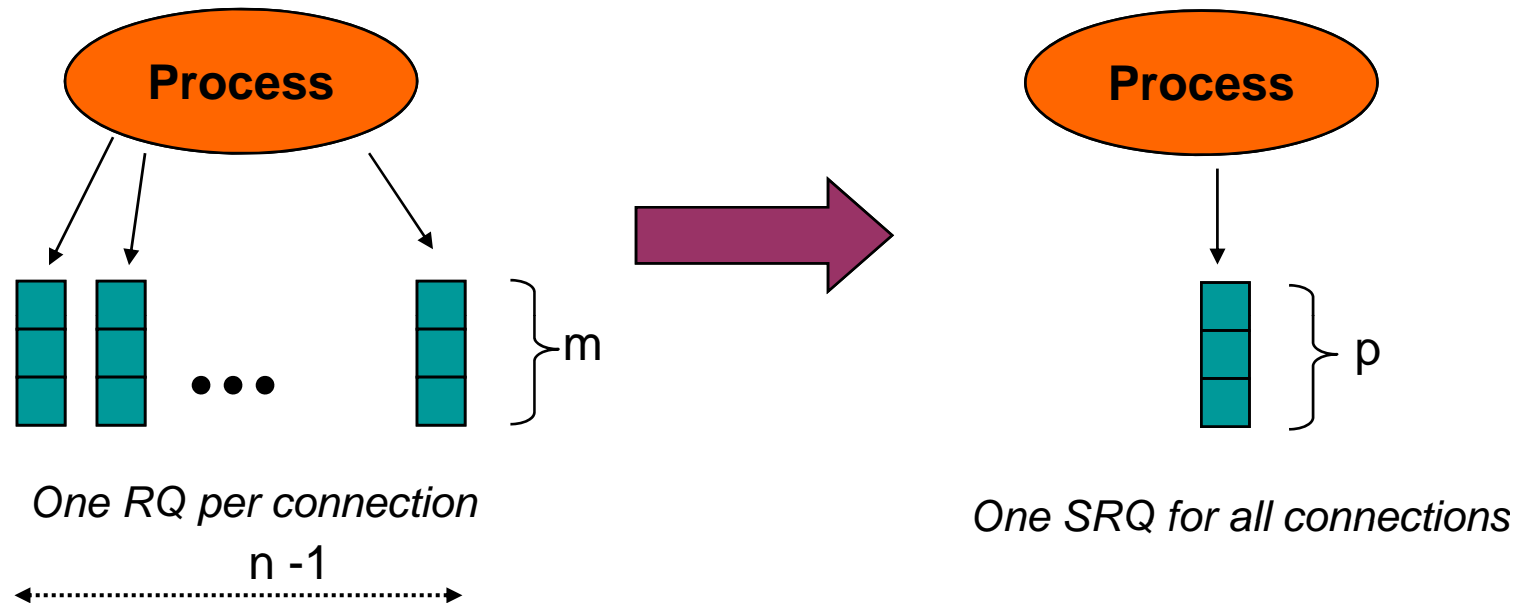
- No involvement by the CPU at the receiver (RDMA Write/Put)
- No involvement by the CPU at the sender (RDMA Read/get)
- 1-2 μ s latency (for short data)
- 1.5 - 2.6 GBps bandwidth (for large data)
- 3-5 μ s for atomic operation

IB Transport Services

Service Type	Connection Oriented	Acknowledged	Transport
Reliable Connection	yes	Yes	IBA
Unreliable Connection	yes	no	IBA
Reliable Datagram	no	Yes	IBA
Unreliable Datagram	no	no	IBA
RAW Datagram	no	no	Raw

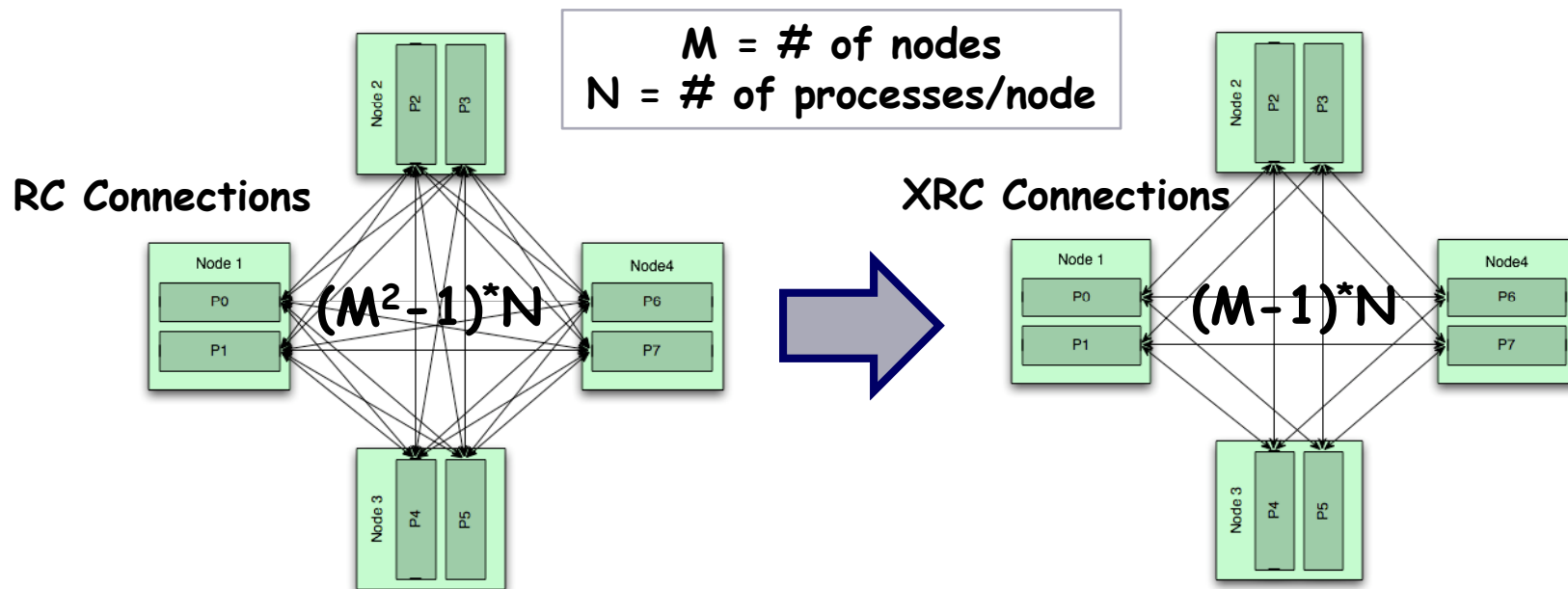
Advanced mechanisms like SRQ and new transport eXtended Reliable Connection (XRC) is introduced recently

Shared Receive Queue (SRQ)



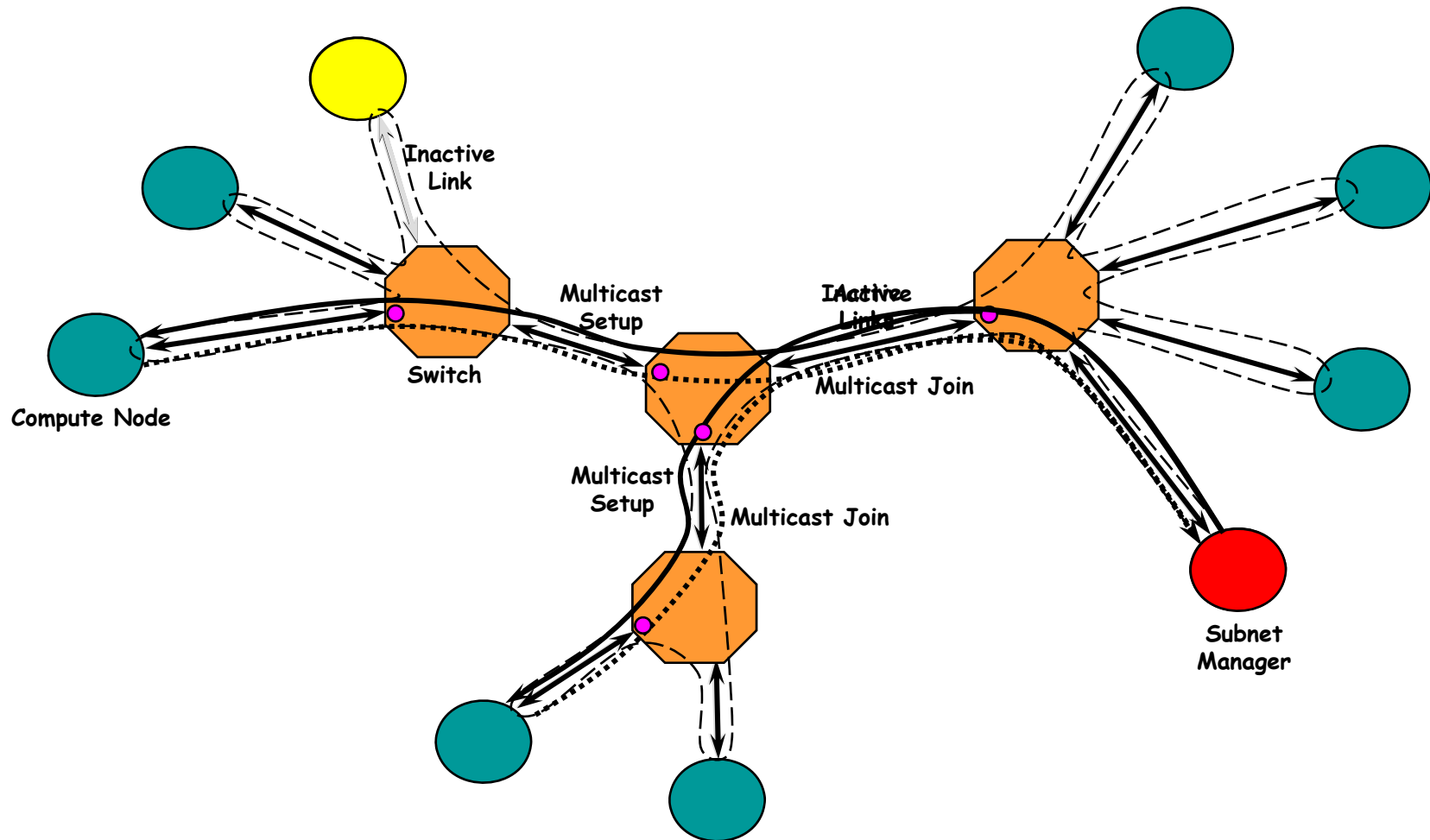
- SRQ is a hardware mechanism in IB by which a process can share receive resources (memory) across multiple connections
- A **new** feature, introduced in specification v1.2
- $0 < p \ll m \cdot (n-1)$

eXtended Reliable Connection (XRC)



- Each QP takes at least one page of memory
 - Connections between all processes is very costly for RC
- **New** IB Transport added: eXtended Reliable Connection
 - Allows connections **between nodes instead of processes**

Subnet Manager



Automatic Path Migration

- Automatically utilizes IB multipathing for network fault-tolerance
- Enables migrating connections to a different path
 - Connection recovery in the case of failures
 - Optional Feature
- Available for RC, UC, and RD
- Reliability guarantees for service type maintained during migration

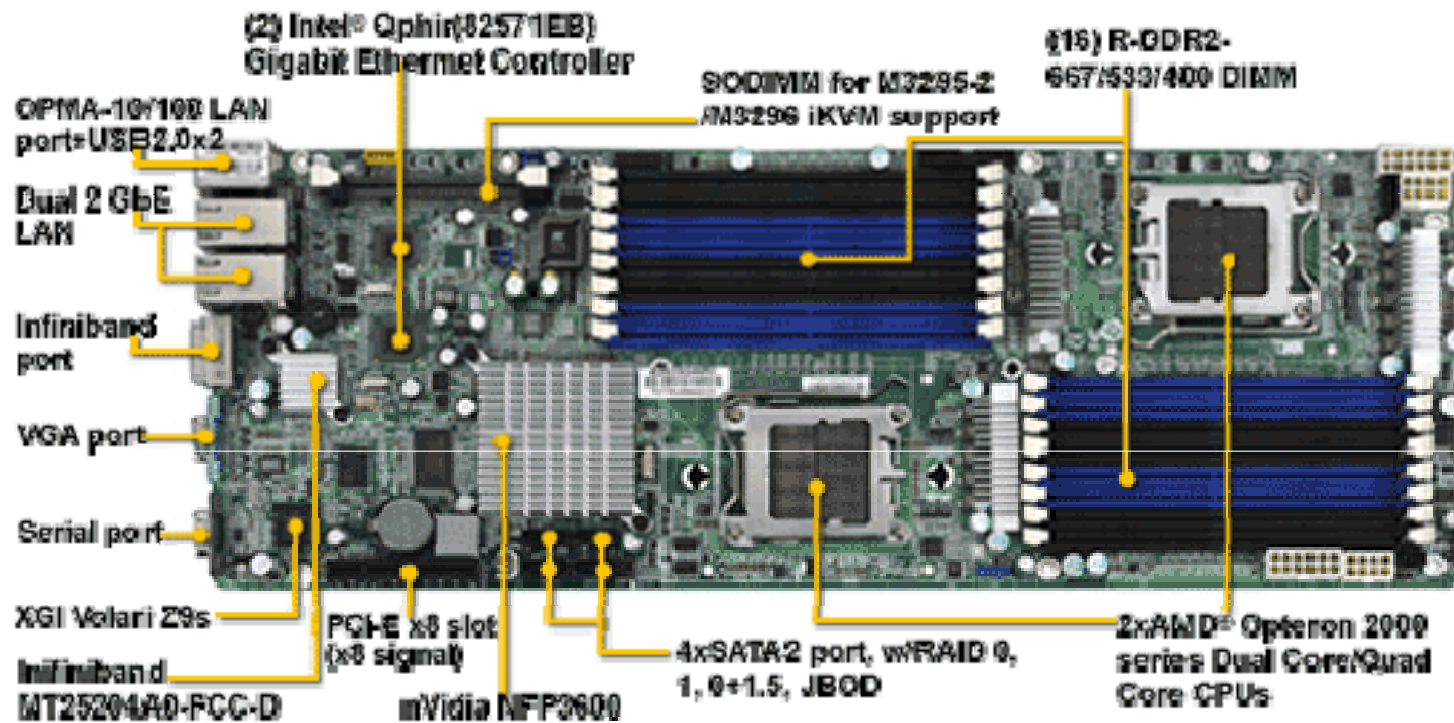
Presentation Overview

- Overview of InfiniBand
 - Features
 - Products (Hardware and Software)
 - Trends
- MVAPICH and MVAPICH2 Features
- Design Insights and Sample Performance Numbers
- Future Plans
- Conclusions and Final Q&A

IB Hardware Products

- Many IB vendors: Mellanox, Voltaire, Cisco, Qlogic
 - Aligned with many server vendors: Intel, IBM, SUN, Dell
 - And many integrators: Appro, Advanced Clustering, Microway, ...
- Broadly two kinds of adapters
 - Offloading (Mellanox) and Onloading (Qlogic)
- Adapters with different interfaces:
 - Dual port 4X with PCI-X (64 bit/133 MHz), PCIe x8, PCIe 2.0 and HT
- MemFree Adapter
 - No memory on HCA → Uses System memory (through PCIe)
 - Good for LOM designs (Tyan S2935, Supermicro 6015T-INFB)
- Different speeds
 - SDR (8 Gbps), DDR (16 Gbps) and QDR (32 Gbps)
- Some 12X SDR adapters exist as well (24 Gbps each way)

Tyan Thunder S2935 Board



(Courtesy Tyan)

IB Hardware Products (contd.)

- Customized adapters to work with IB switches
 - Cray XD1 (formerly by Octigabay), Cray CX1
- Switches:
 - 4X SDR switch (8-288 ports)
 - 12X ports available for inter-switch connectivity
 - 4X DDR switch (mainly available in 8 to 288 port models)
 - 12X switches (small sizes available)
 - 3456-port "Magnum" switch from SUN → used at TACC
 - 72-port "nano magnum" switch with DDR speed
 - New 36-port InfiniScale IV QDR switch silicon by Mellanox
 - Will allow high-density switches to be built
- Switch Routers with Gateways
 - IB-to-FC; IB-to-IP

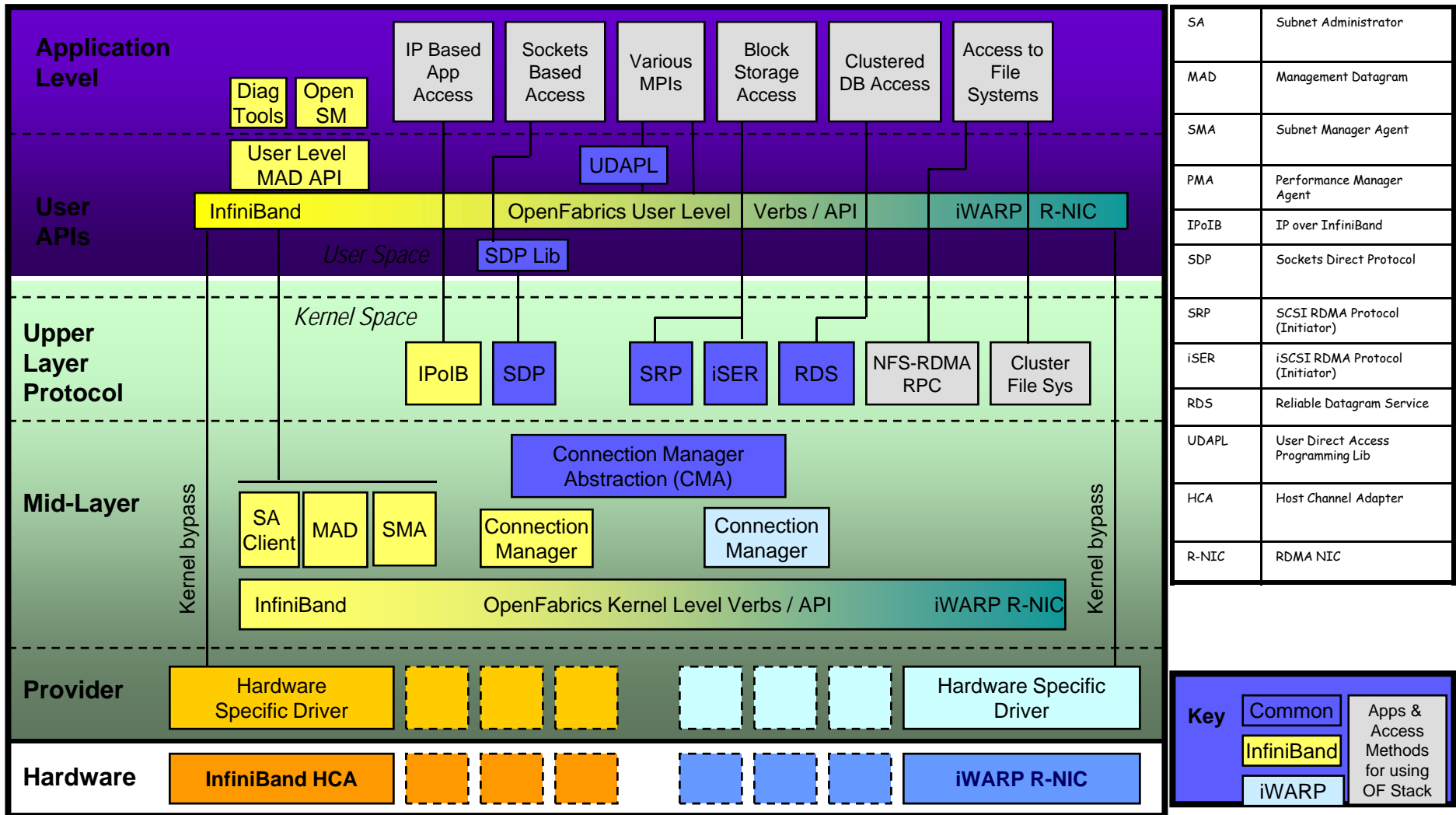
IB Software Products

- Low-level software stacks
 - VAPI (Verbs-Level API) from Mellanox
 - Modified and customized VAPI from other vendors
 - New initiative: Open Fabrics (formerly OpenIB)
 - <http://www.openfabrics.org>
 - Open-source code available with Linux distributions
 - Initially IB; later extended to incorporate iWARP
- High-level software stacks
 - MPI, SDP, IPoIB, SRP, iSER, DAPL, NFS, PVFS on various stacks (primarily VAPI and OpenFabrics)

OpenFabrics

- www.openfabrics.org
- Open source organization (formerly OpenIB)
- Incorporates both IB and iWARP in a unified manner
- Focusing on effort for Open Source IBA and iWARP support for Linux and Windows
- Design of complete software stack with 'best of breed' components
 - Gen1
 - Gen2 (current focus)
- Users can download the entire stack and run
 - Latest release is OFED 1.3.1
 - OFED 1.4 is being worked out

OpenFabrics Software Stack



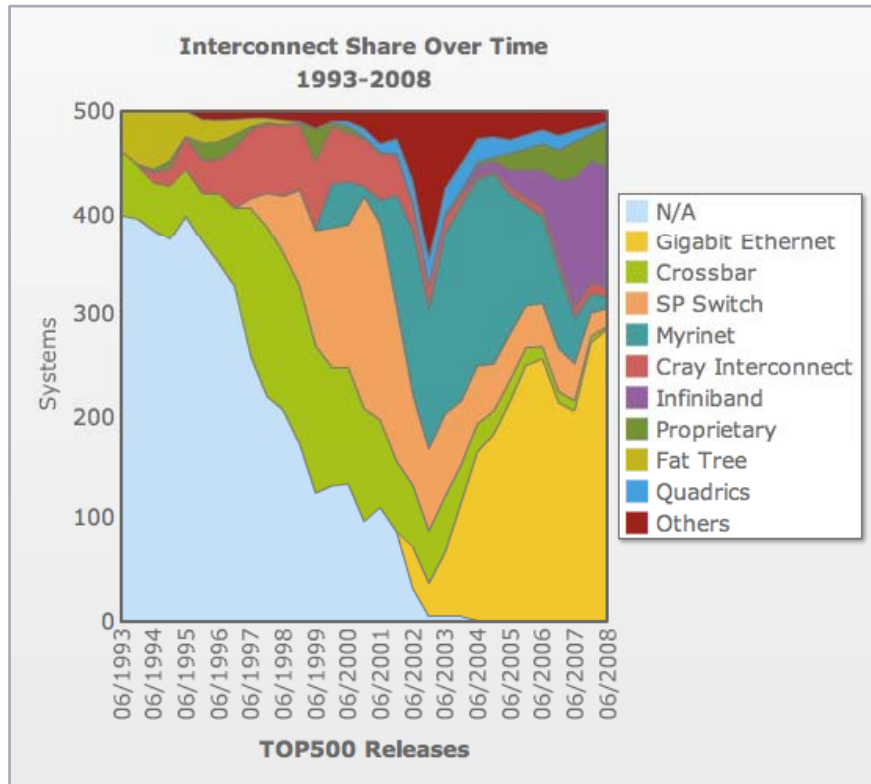
SA	Subnet Administrator
MAD	Management Datagram
SMA	Subnet Manager Agent
PMA	Performance Manager Agent
IPoIB	IP over InfiniBand
SDP	Sockets Direct Protocol
SRP	SCSI RDMA Protocol (Initiator)
iSER	iSCSI RDMA Protocol (Initiator)
RDS	Reliable Datagram Service
UDAPL	User Direct Access Programming Lib
HCA	Host Channel Adapter
R-NIC	RDMA NIC

IB Installations

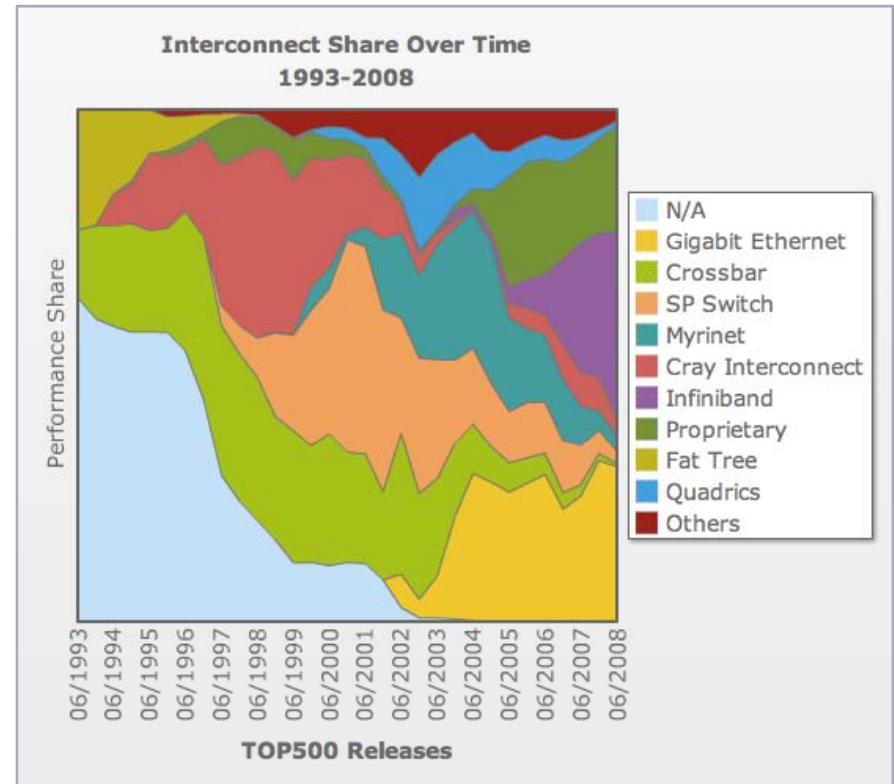
- 121 IB clusters (24.2%) in June '08 TOP500 list (www.top500.org)
- 12 IB clusters in TOP25
 - 122,400-cores (RoadRunner) at LANL (1st)
 - 62,976-cores (Ranger) at TACC (4th)
 - 14,336-cores at New Mexico (7th)
 - 14,384-cores at Tata CRL, India (8th)
 - 10,240-cores at TEP, France (10th)
 - 13,728-cores in Sweden (11th)
 - 8,320-cores in UK (18th)
 - 6,720-cores in Germany (19th)
 - 10,000-cores at CCS, Tsukuba, Japan (20th)
 - 9,600-cores at NCSA (23rd)
 - 12,344-cores at Tokyo Inst. of Technology (24th)
 - 13,824-cores at NASA/Columbia (25th)
- More are getting installed

InfiniBand in the Top500

Systems



Performance



Percentage share of InfiniBand is steadily increasing

Presentation Overview

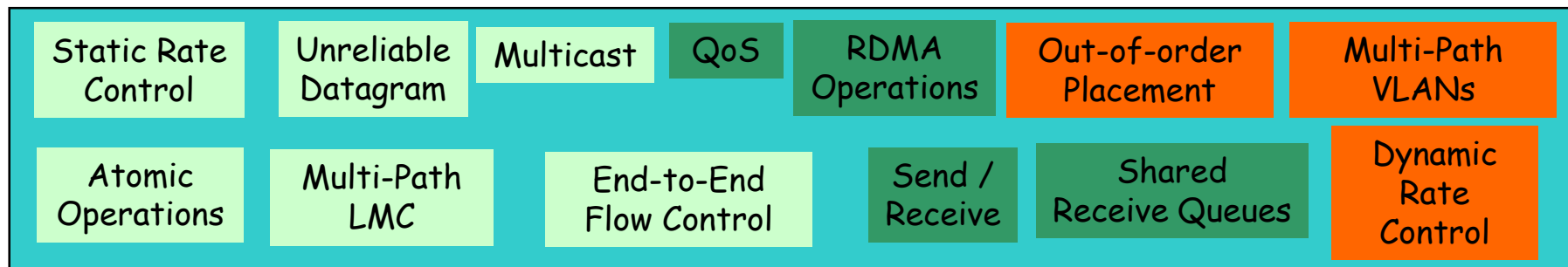
- Overview of InfiniBand
 - Features
 - Products (Hardware and Software)
 - Trends
- **MVAPICH and MVAPICH2 Features**
- Design Insights and Sample Performance Numbers
- Future Plans
- Conclusions and Final Q&A

Designing MPI Using IB/iWARP Features

MPI Design Components



Design Alternatives and Solutions

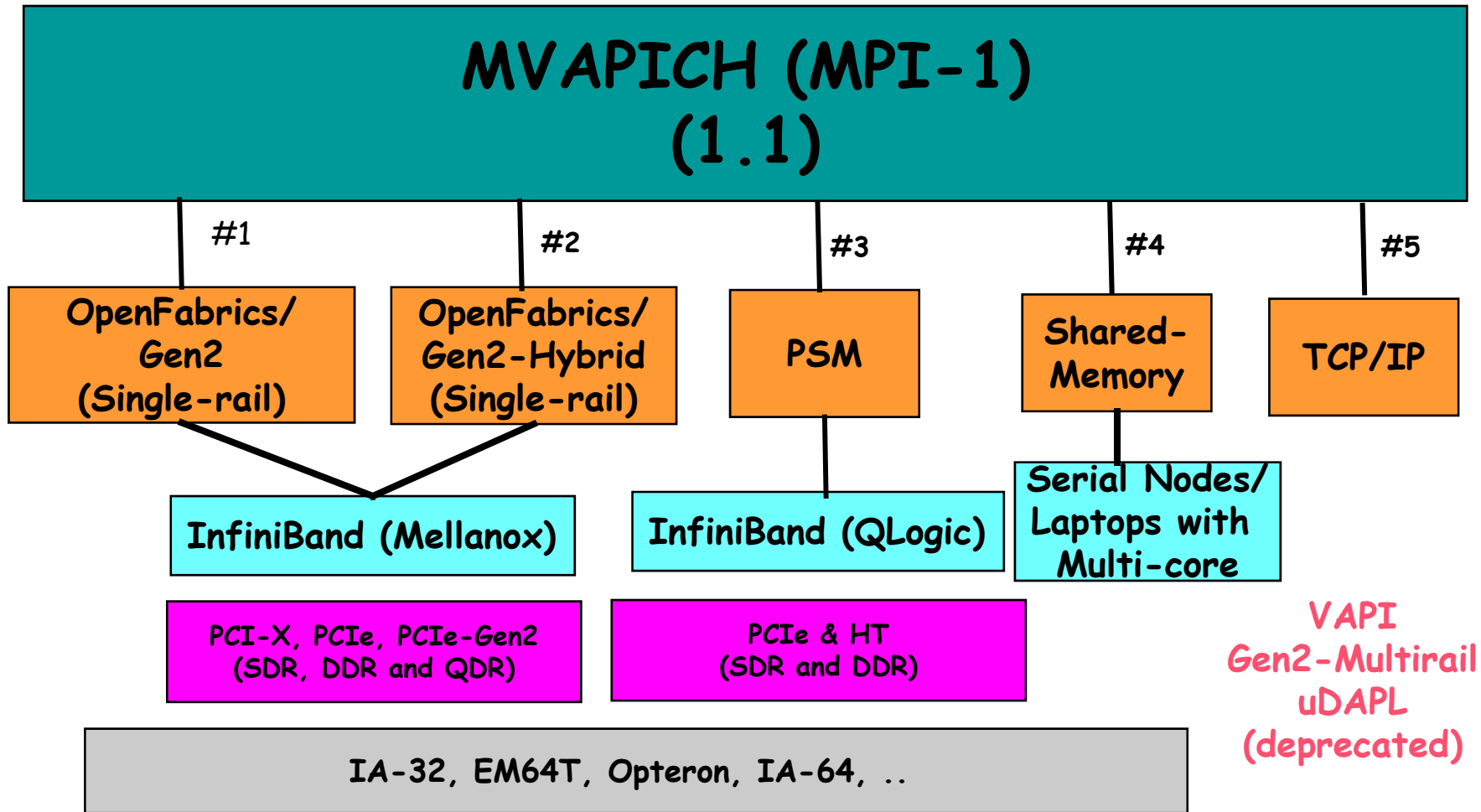


IB and iWARP/Ethernet Features

MVAPICH/MVAPICH2 Software

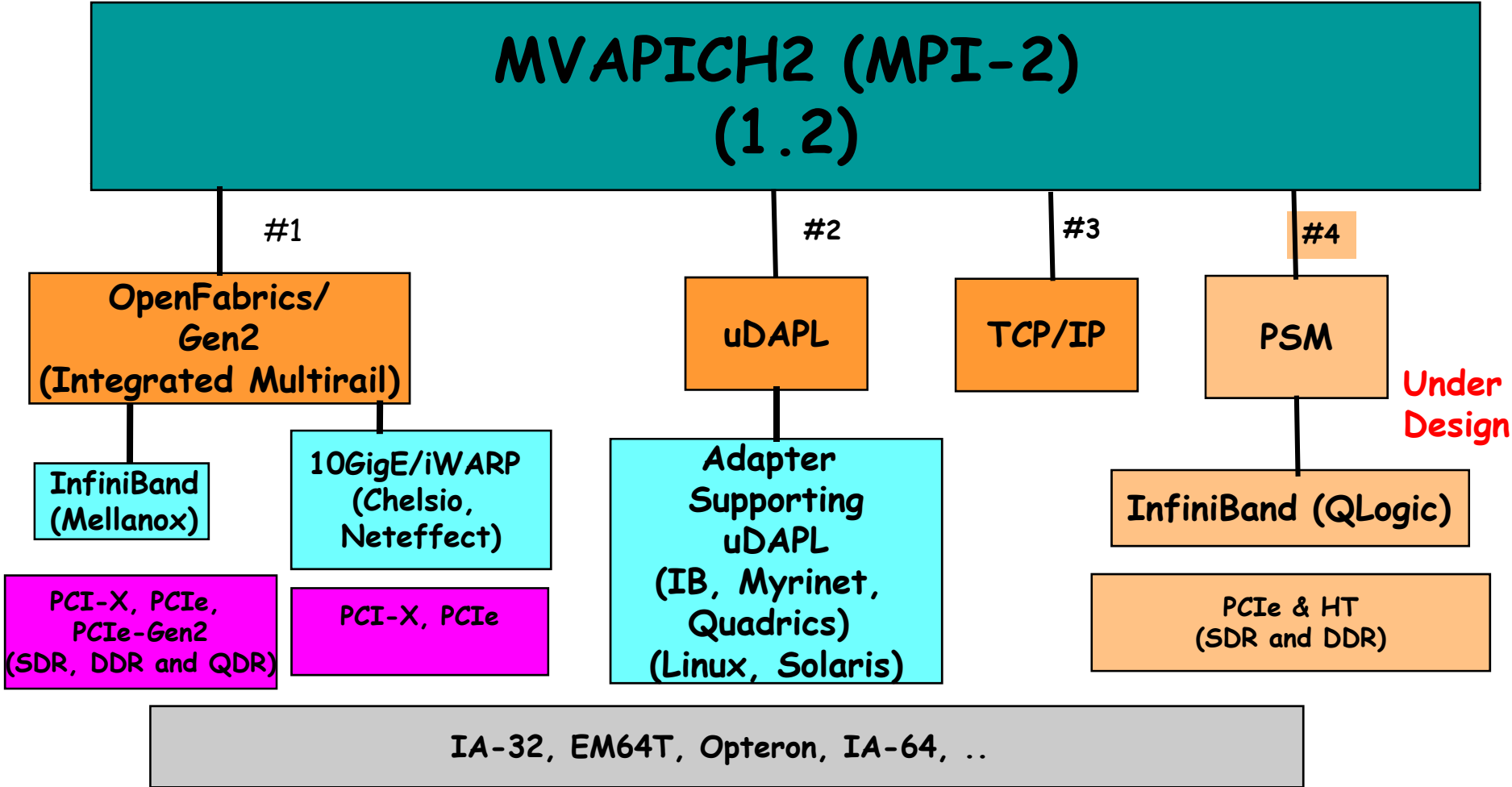
- High Performance MPI Library for IB and 10GE
 - MVAPICH (MPI-1) and MVAPICH2 (MPI-2)
 - **Latest Releases: MVAPICH 1.1RC1 and MVAPICH2 1.2RC2**
 - Used by more than 765 organizations in 42 countries
 - More than 23,000 downloads from OSU site directly
 - Empowering many TOP500 clusters
 - 4th ranked 62,976-core cluster (Ranger) at TACC
 - Available with software stacks of many IB, 10GE and server vendors including Open Fabrics Enterprise Distribution (OFED)
 - Also supports uDAPL device to work with any network supporting uDAPL
 - <http://mvapich.cse.ohio-state.edu/>

MVAPICH 1.1 Architecture



VAPI
Gen2-Multirail
uDAPL
(deprecated)



MVAPICH2 1.2 Architecture





Major Features of MVAPICH 1.1



- OpenFabrics-Gen2
 - Scalable job start-up with mpirun_rsh, support for SLURM
 - RC and XRC support
 - Flexible message coalescing
 - Multi-core-aware pt-to-pt communication
 - User-defined processor affinity for multi-core platforms
 - Multi-core-optimized collective communication
 - Asynchronous and scalable on-demand connection management
 - RDMA Write and RDMA Read-based protocols
 - Lock-free Asynchronous Progress for better overlap between computation and communication
 - Polling and blocking support for communication progress
 - Multi-pathing support leveraging LMC mechanism on large fabrics
 - Network-level fault tolerance with Automatic Path Migration (APM)
 - Mem-to-mem reliable data transfer mode (for detection of I/O error with 32-bit CRC)
- 
- 

•
•
•



Major Features of MVAPICH 1.1 (Cont'd)

- OpenFabrics-Gen2-Hybrid
 - Newly introduced interface in 1.1
 - Replaces UD interface in 1.0
 - Targeted for emerging multi-thousand-core clusters to achieve the best performance with minimal memory footprint
 - Most of the features as in Gen2
 - Adaptive selection during run-time (based on application and systems characteristics) to switch between
 - RC and UD (or between XRC and UD) transports
 - Multiple buffer organization with XRC support



Major Features of MVAPICH2 1.2



- OpenFabrics-Gen2
 - All features as in MVAPICH 1.1 (OpenFabrics-Gen2) except asynchronous progress and XRC
 - RDMA CM-based connection management (Gen2-IB and Gen2-iWARP)
 - Integrated multi-rail support for IB and 10GigE/iWARP
 - Checkpoint-Restart (currently for IB)
 - Systems-level automatic
 - Application-initiated systems-level
 - uDAPL
 - Most of the features of OpenFabrics-Gen2 except multi-rail and checkpointing
 - Flexibility for different adapters , software stacks and OS (Linux and Solaris) supporting uDAPL
- 
- 

Support for Multiple Interfaces/Adapters

- OpenFabrics/Gen2-IB and OpenFabrics/Gen2-Hybrid
 - All IB adapters supporting OpenFabrics/Gen2
- Qlogic/PSM
 - Qlogic adapters
- OpenFabrics/Gen2-iWARP
 - Chelsio
- uDAPL
 - Linux-IB
 - Solaris-IB
 - Other adapters such as Neteffect 10GigE
- TCP/IP
 - Any adapter supporting TCP/IP interface
- Shared Memory Channel (MVAPICH)
 - for running applications in a node with multi-core processors

Presentation Overview

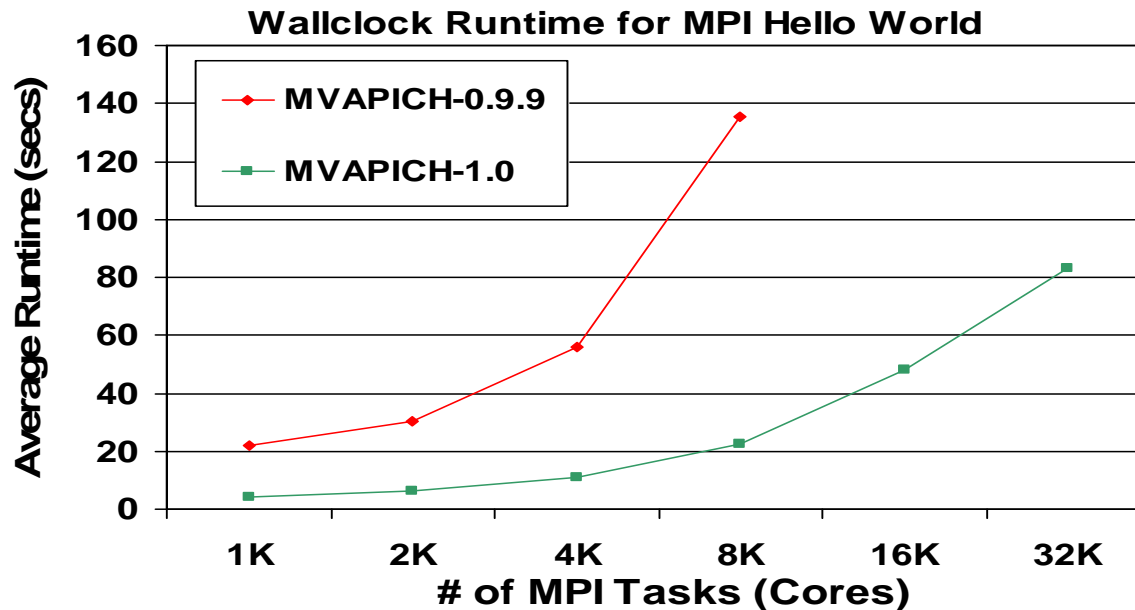
- Overview of InfiniBand
 - Features
 - Products (Hardware and Software)
 - Trends
- MVAPICH and MVAPICH2 Features
- Design Insights and Sample Performance Numbers
- Future Plans
- Conclusions and Final Q&A

Design Insights and Sample Results

- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- Integrated Multi-rail Design
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- Asynchronous Progress
- Fault Tolerance

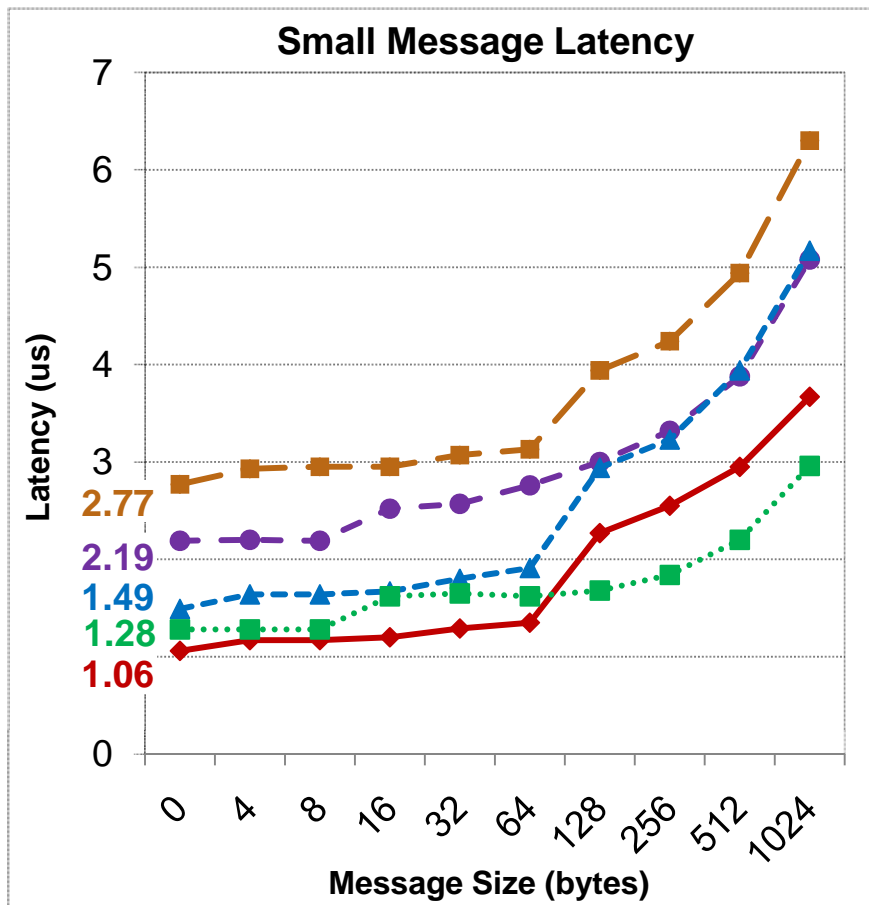
Scalable Startup

- An enhanced mpirun_rsh framework was introduced in MVAPICH 1.0 to significantly cut down job start-up on large clusters
- Is available with MVAPICH 1.1 and MVAPICH2 1.2

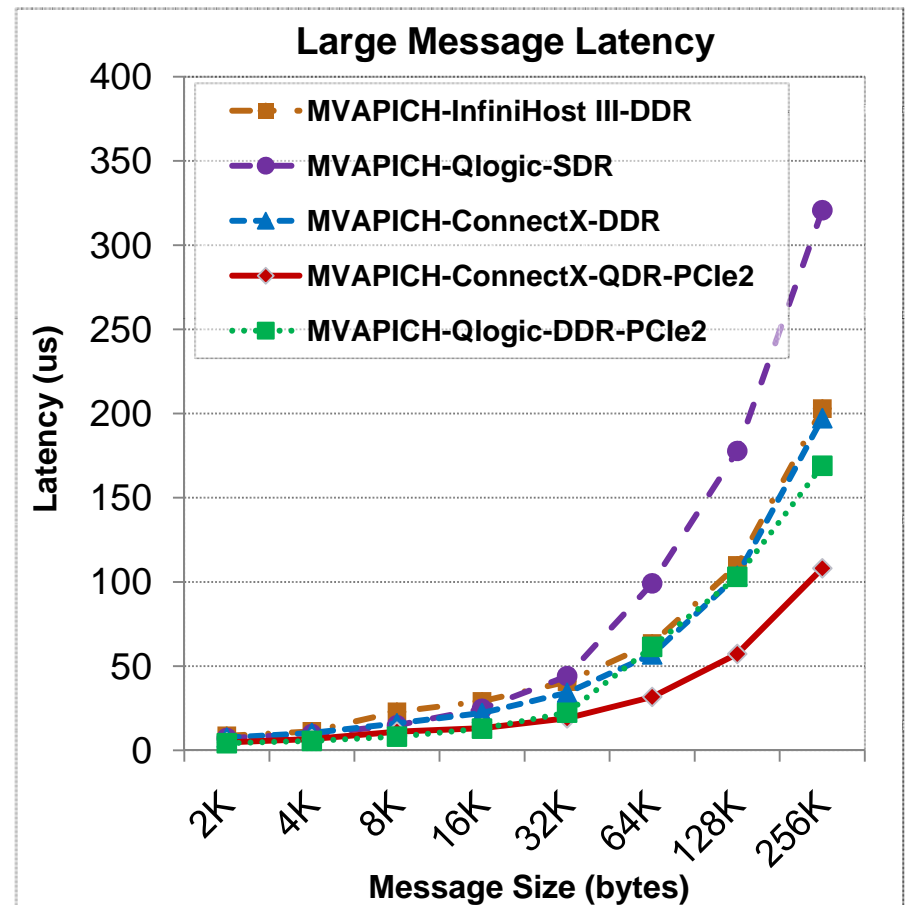


Courtesy TACC

One-way Latency: MPI over IB

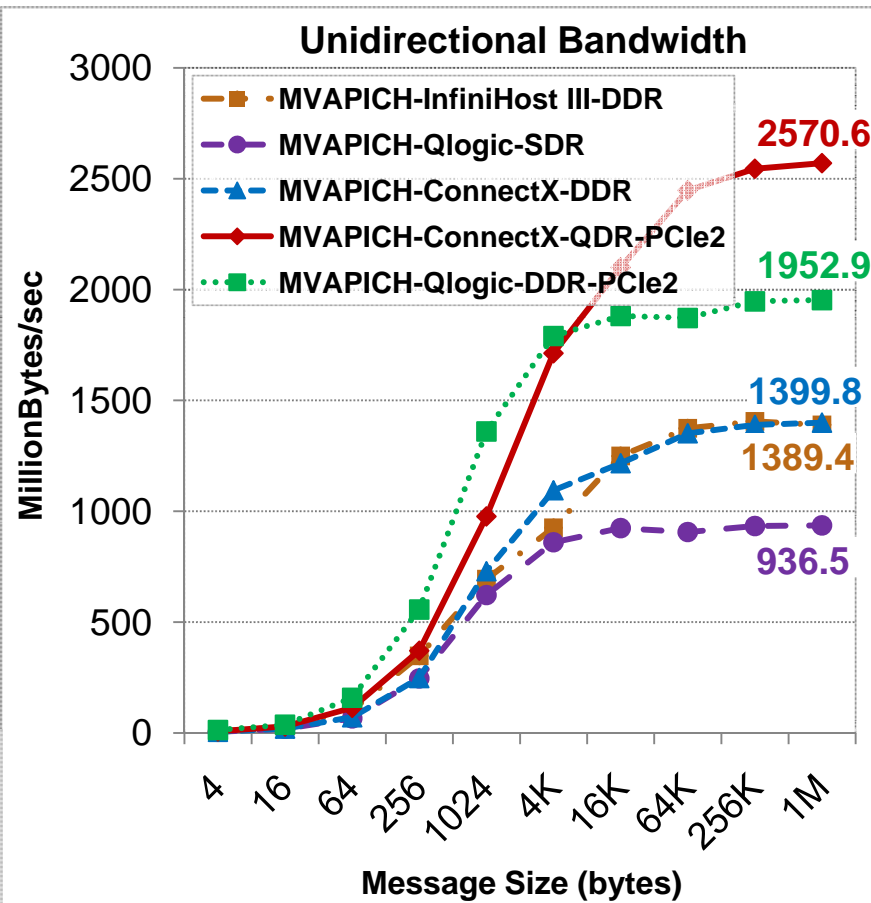


InfinitiHost III and ConnectX-DDR: 2.33 GHz Quad-core (Clovertown) Intel with IB switch

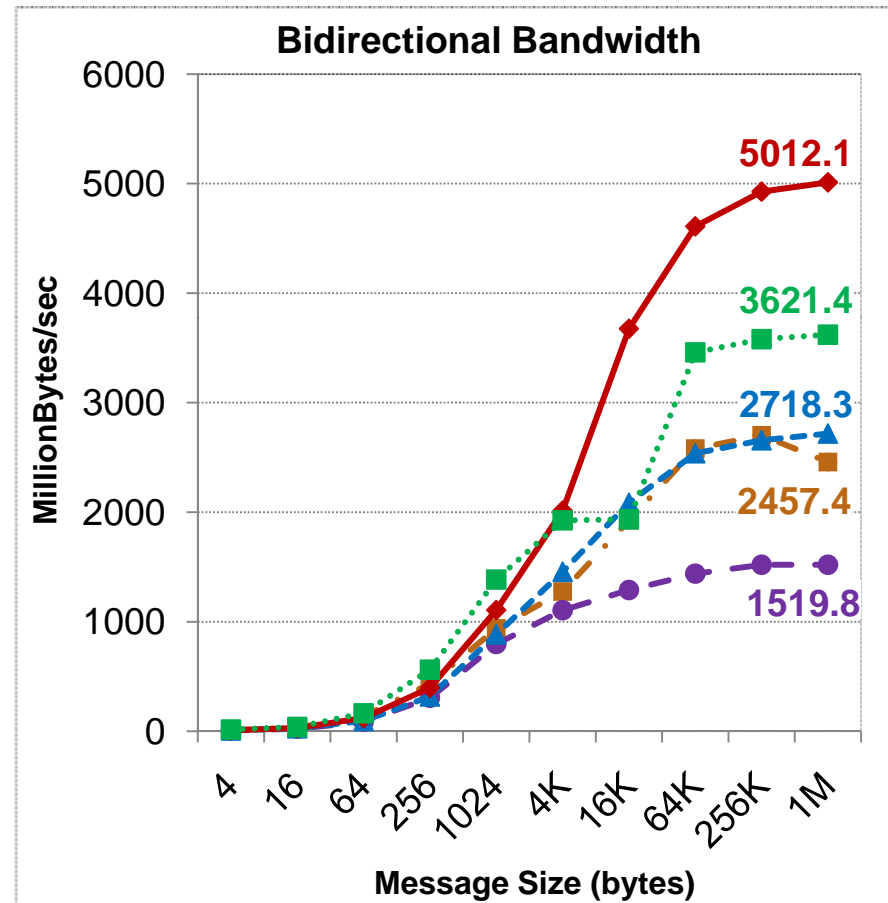


ConnectX-QDR-PCIe2: 2.83 GHz Quad-core (Harpertown) Intel with back-to-back

Bandwidth: MPI over IB



InfiniHost III and ConnectX-DDR: 2.33 GHz Quad-core (Clovertown) Intel with IB switch

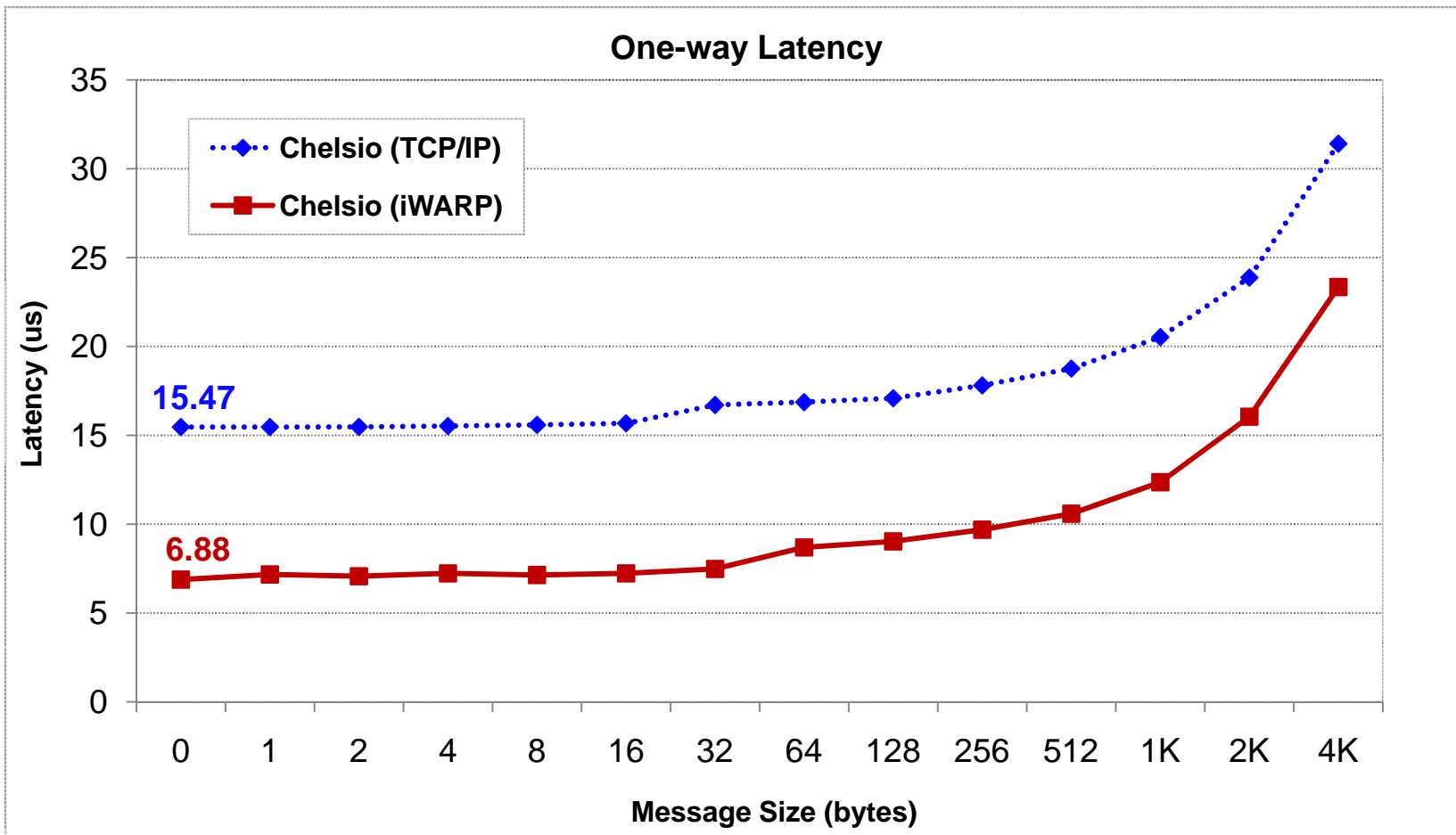


ConnectX-QDR-PCle2: 2.83 GHz Quad-core (Harpertown) Intel with back-to-back

RDMA CM and iWARP Support

- Available starting with MVAPICH2 0.9.8
- RDMA CM is supported for both
 - IB
 - 10GigE/iWARP

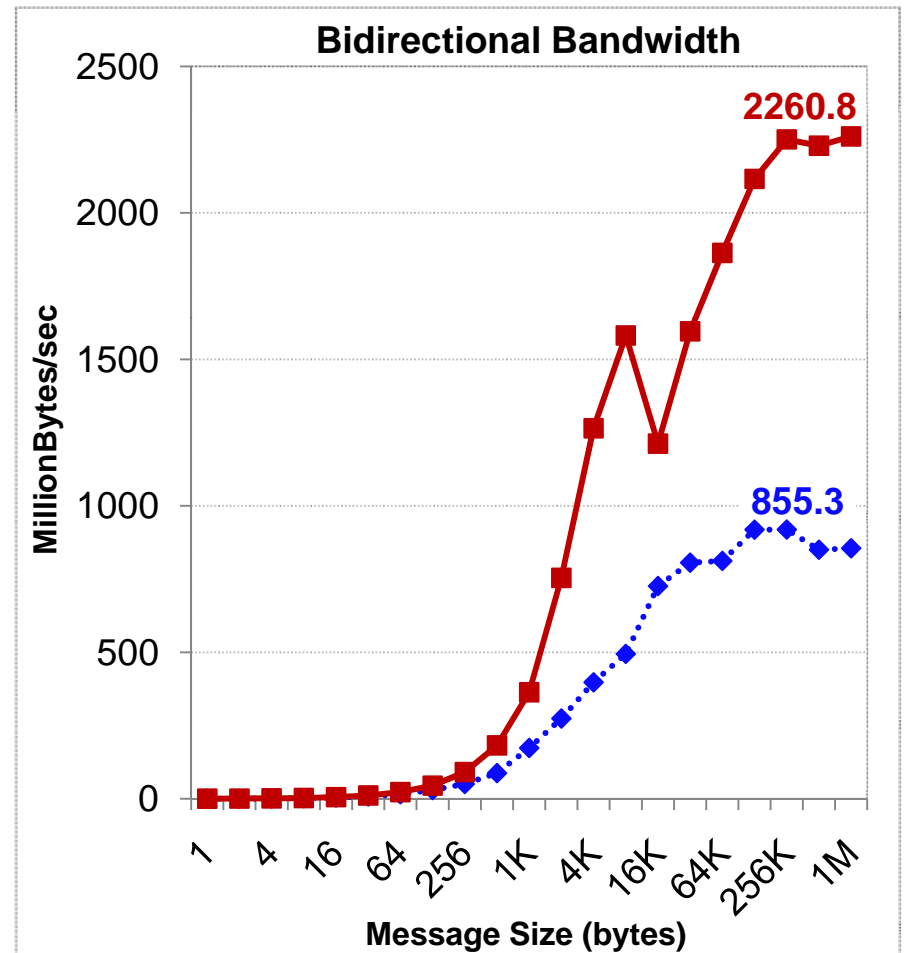
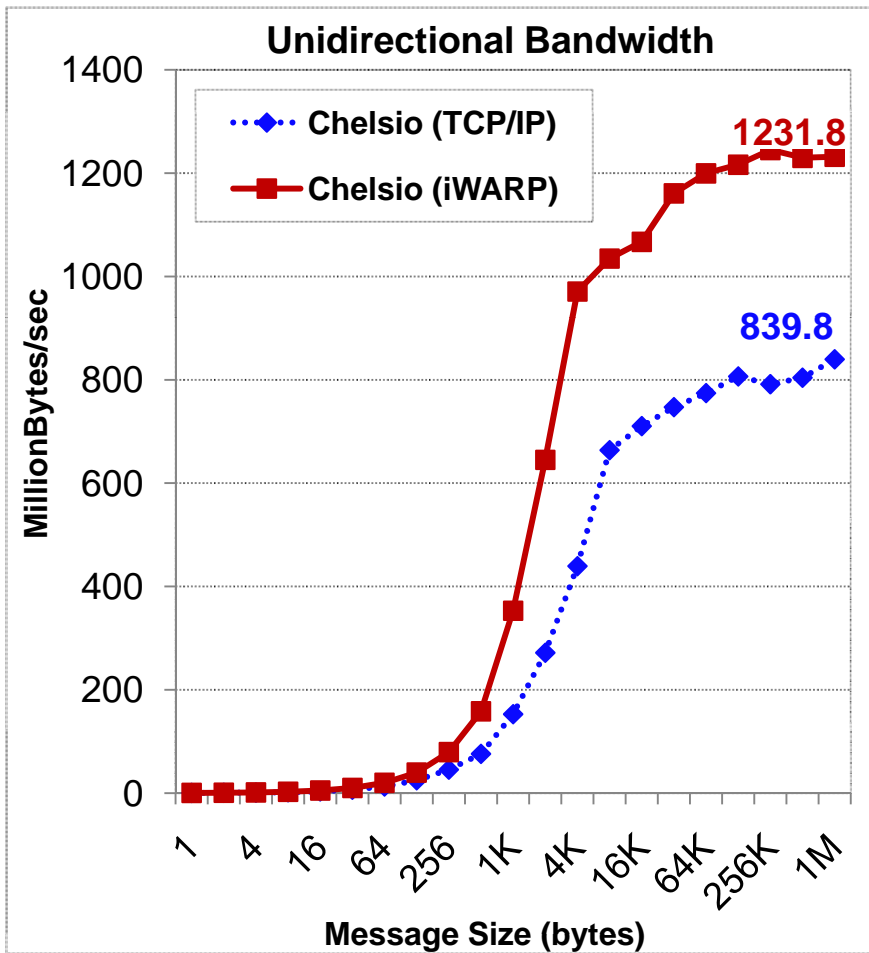
One-way Latency: MPI over iWARP



2.0 GHz Quad-core Intel with 10GE (Fulcrum) Switch

Tsukuba, Oct 2, 2008

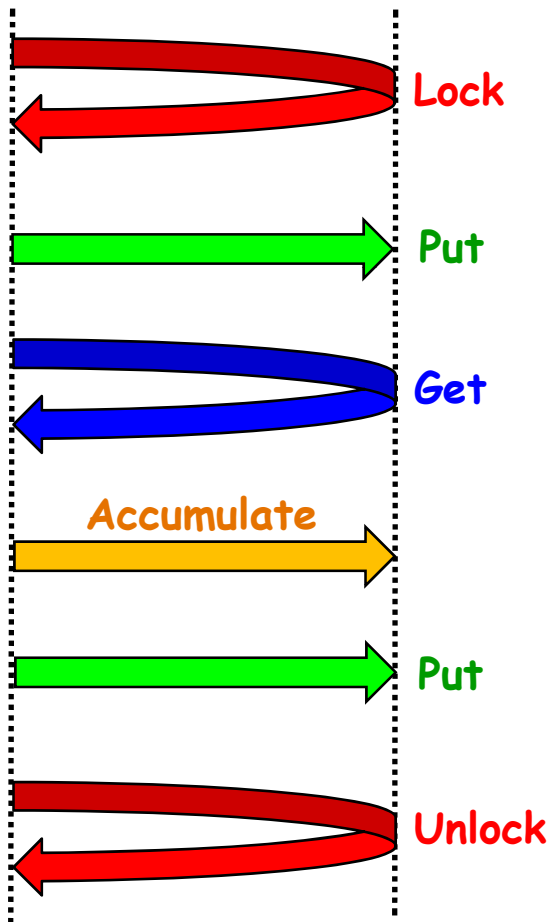
Bandwidth: MPI over iWARP



2.0 GHz Quad-core Intel with 10GE (Fulcrum) Switch



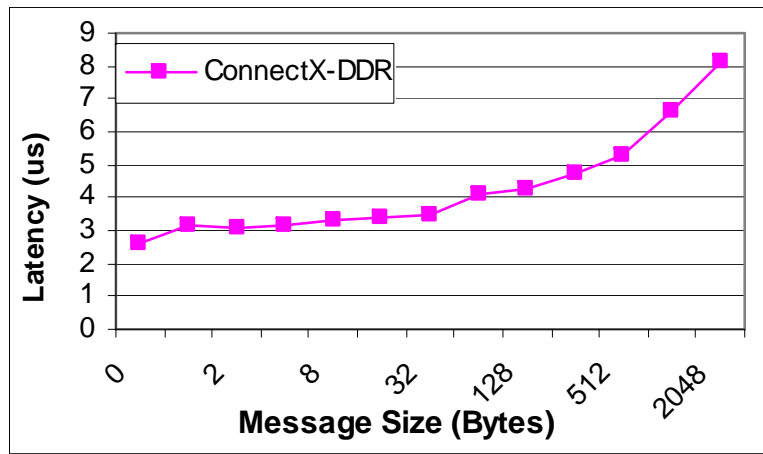
MPI One-sided Communication



- Specified by the MPI-2 standard
- Data movement operations
 - MPI_Put
 - MPI_Get
 - MPI_Accumulate
- Synchronization operations
 - MPI_Lock/MPI_Unlock
 - MPI_Win_fence
 - MPI_Win_post, MPI_Win_start, MPI_Win_complete, MPI_Win_wait

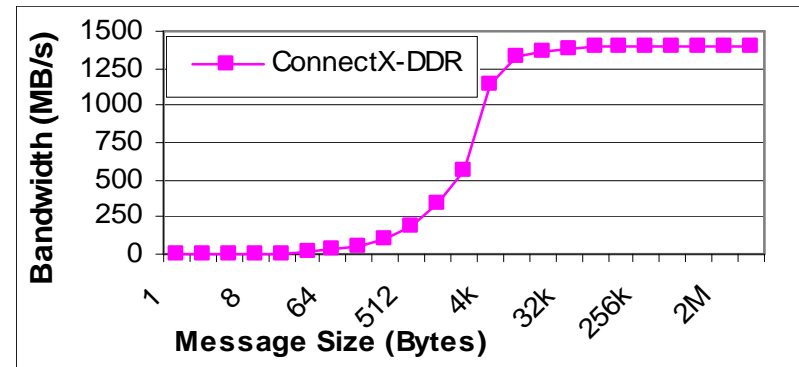
MPI_Put Performance (IB DDR)

2.57

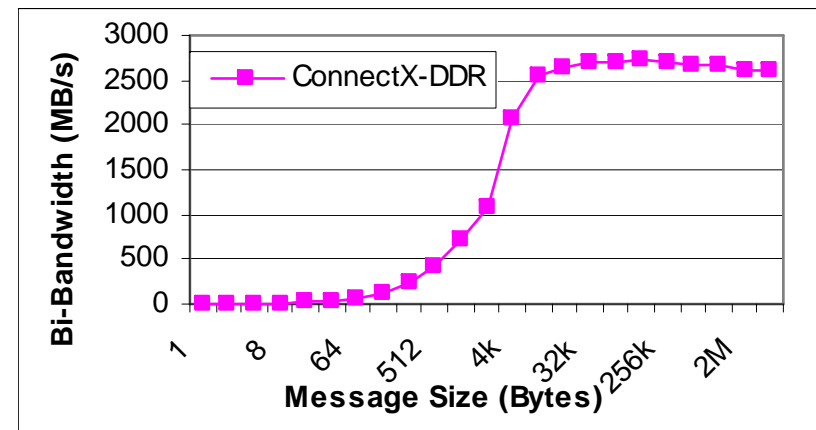


- Single port results only (EM64T, PCI-Ex)

Results for other platforms at
<http://mvapich.cse.ohio-state.edu>



1405

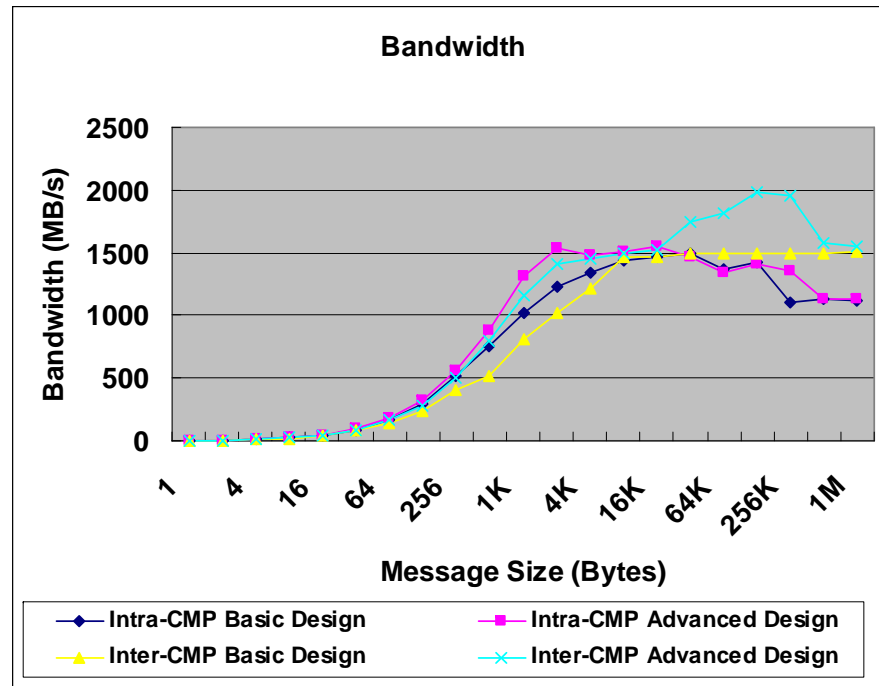
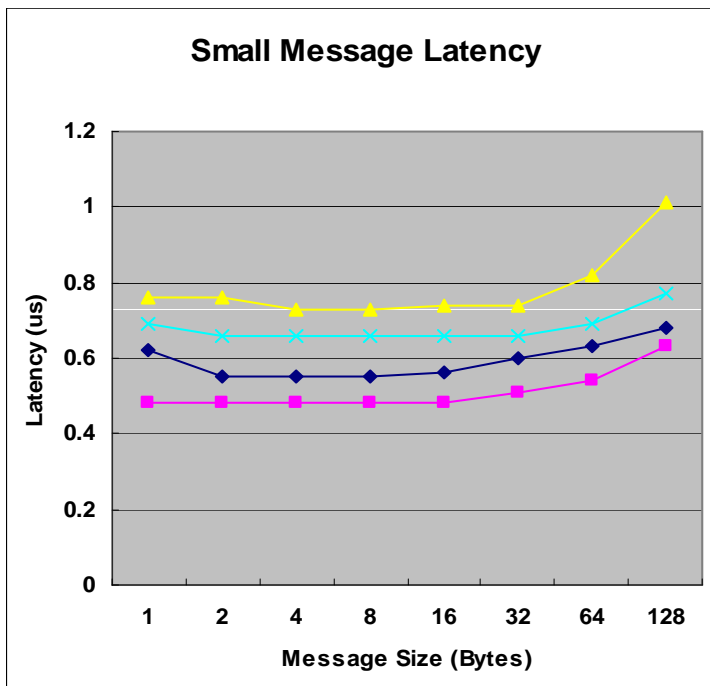


2716

Design Insights and Sample Results

- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- Integrated Multi-rail Design
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- Asynchronous Progress
- Fault Tolerance

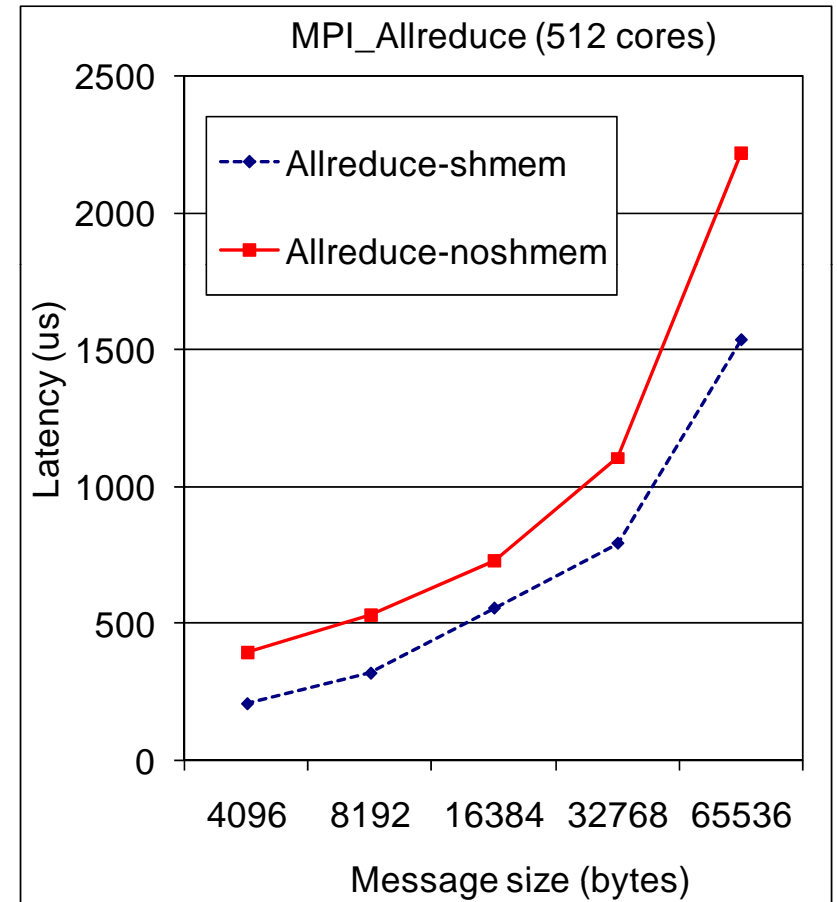
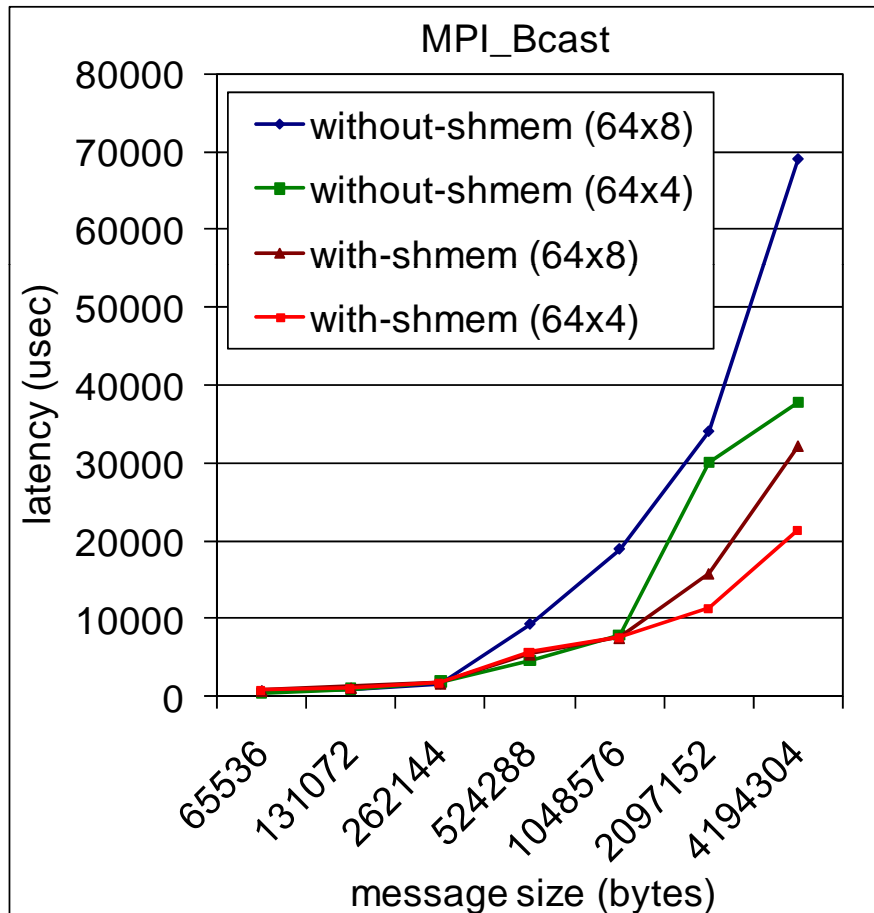
Multicore-aware Communication: Latency and Bandwidth



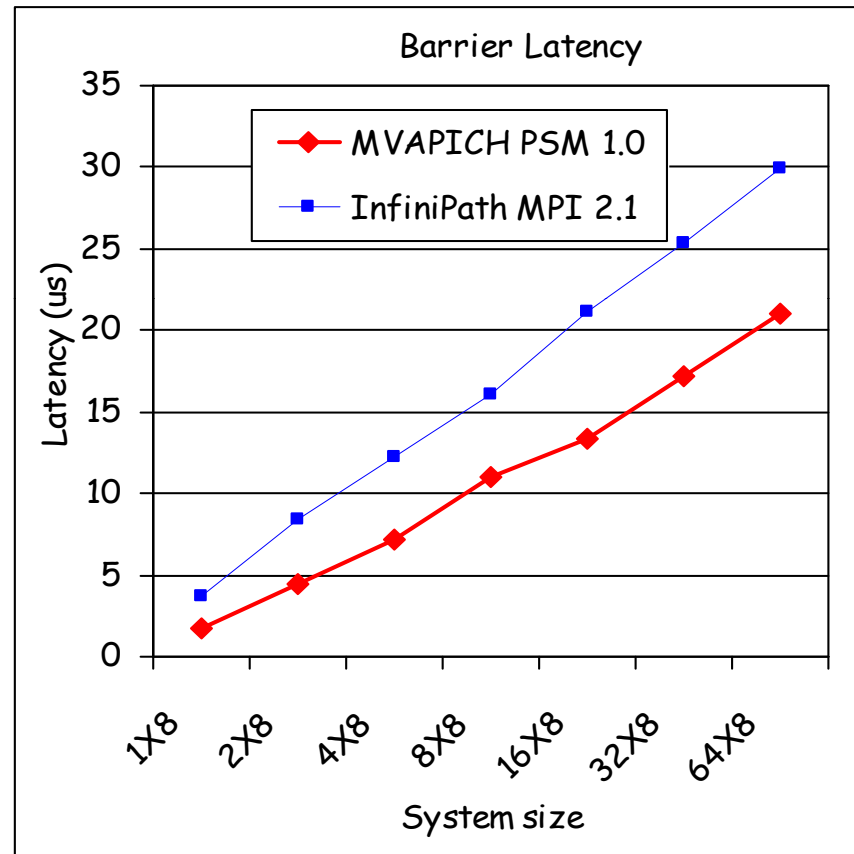
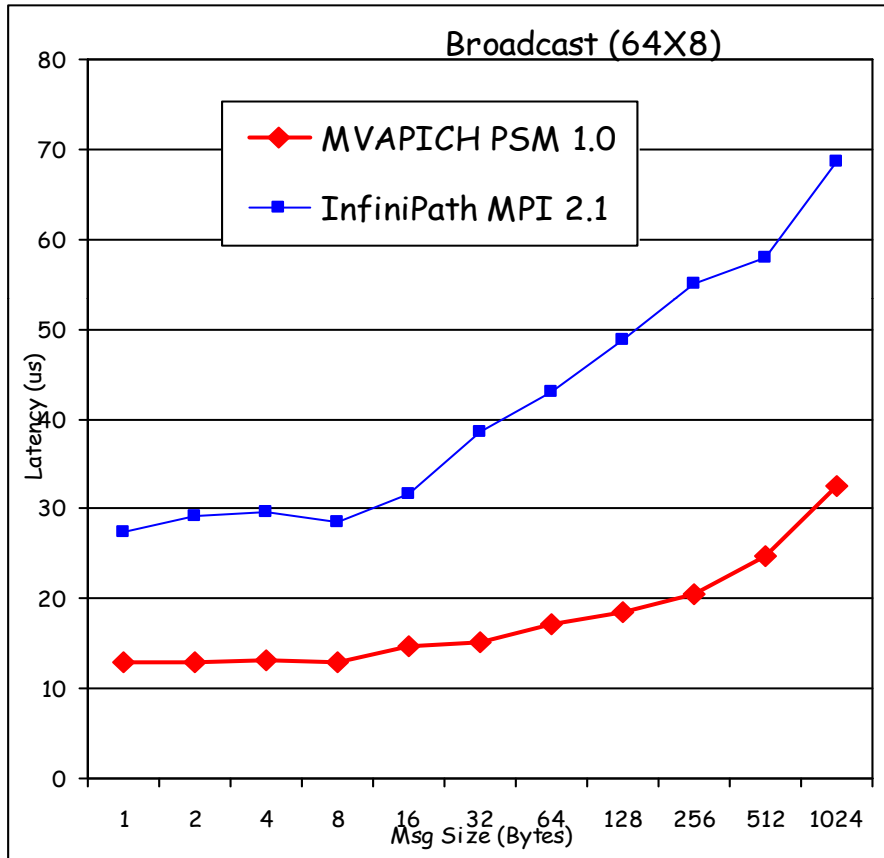
- Multicore-aware design improves both latency and bandwidth
- Available in *MVAPICH* and *MVAPICH2* stacks

L. Chai, A. Hartono and D. K. Panda, "Designing High Performance and Scalable MPI Intra-node Communication Support for Clusters", Cluster '06

Shared-memory Aware Collectives



MVAPICH-PSM Collective Performance (512 cores)

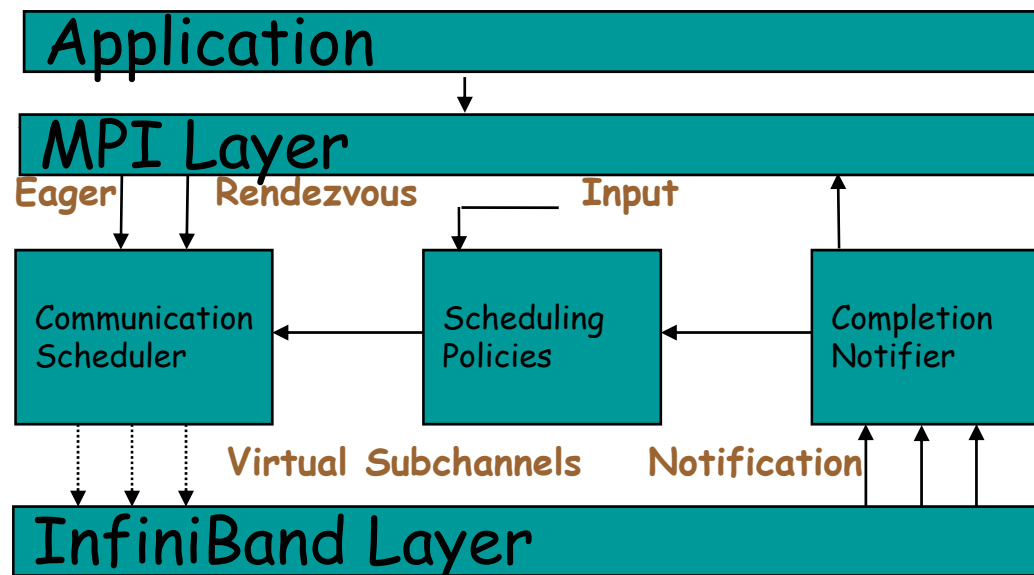


- 64 Intel Quad-core systems with dual sockets; PCIe InfiniPath Adapters
- Significant performance improvement for MPI_Bcast and MPI_Barrier

Design Insights and Sample Results

- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- **Integrated Multi-rail Design**
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- Asynchronous Progress
- Fault Tolerance

Integrated Multi-Rail Design (MVAPICH2)



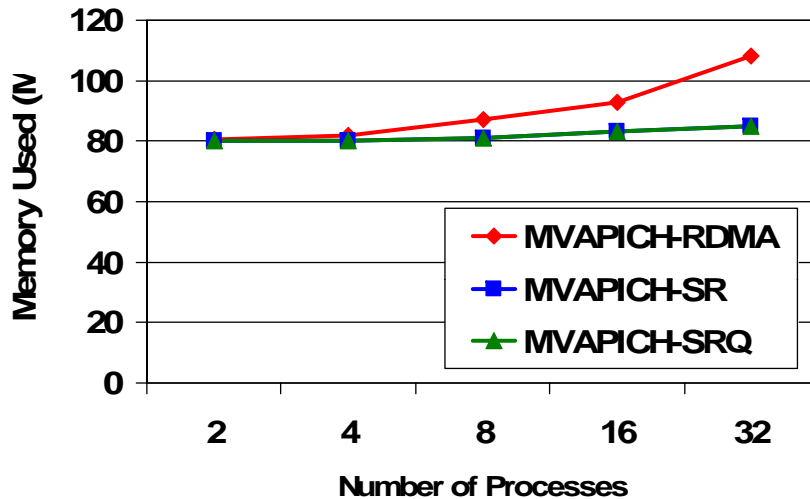
- Multiple ports/adapters
- Multiple adapters
- Multiple paths with LMCs

J. Liu, A. Vishnu and D. K. Panda. Building MultiRail InfiniBand Clusters: MPI Level Design and Performance Evaluation. Presented at Supercomputing '04, April, 2004

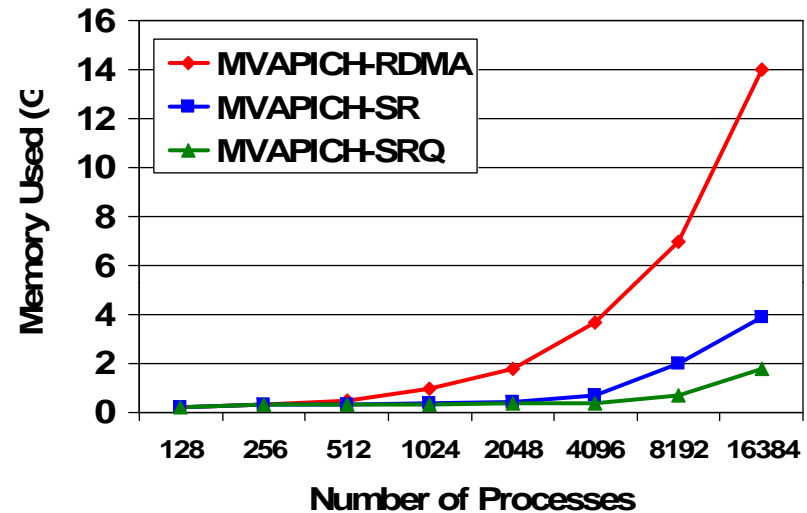
Design Insights and Sample Results

- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- Integrated Multi-rail Design
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- Asynchronous Progress
- Fault Tolerance

Memory Utilization using Shared Receive Queues



MPI_Init memory utilization

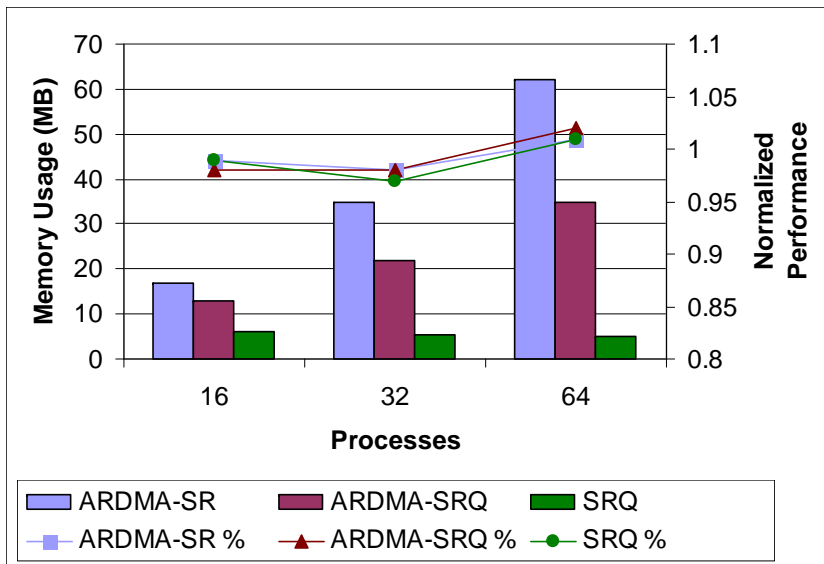


Analytical model

- SRQ consumes **only 1/10th** compared to RDMA for 16,000 processes
- Send/Recv exhausts the *Buffer Pool* after 1000 processes; consumes **2X** memory as SRQ for 16,000 processes

S. Sur, L. Chai, H. -W. Jin and D. K. Panda, "Shared Receive Queue Based Scalable MPI Design for InfiniBand Clusters", IPDPS 2006

Communication Buffer Memory Utilization with NAMD (apoa1)



Avg. RDMA channels	53.15
Avg. Low watermarks	0.03
Unexpected Msgs (%)	48.2
Total Messages	3.7e6
MPI Time (%)	23.54

- 50% messages < 128 Bytes, other 50% between 128 Bytes and 32 KB
 - 53 RDMA connections setup for 64 process experiment
- SRQ Channel takes 5-6MB of memory
 - Memory needed by SRQ decreases by 1MB going from 16 to 64

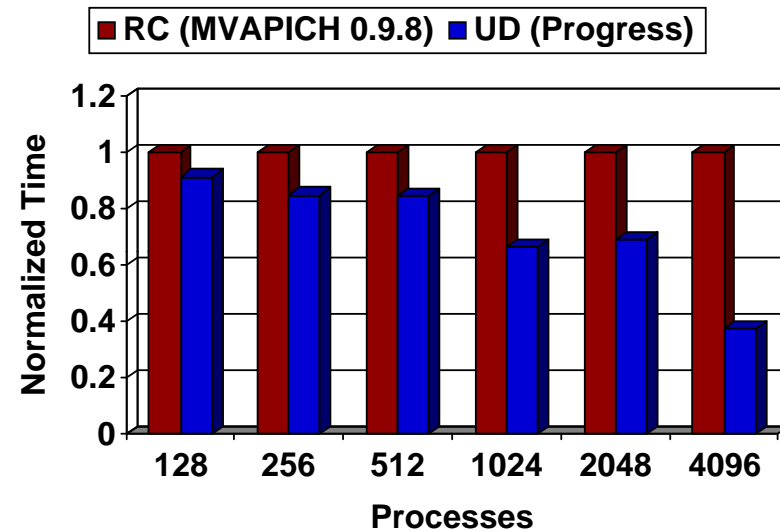
S. Sur, M. Koop and D. K. Panda, "High-Performance and Scalable MPI over InfiniBand with Reduced Memory Usage: An In-Depth Performance Analysis", SC '06

UD vs. RC: Performance and Scalability (SMG2000 Application)

Memory Usage (MB/process)

	RC (MVAPICH 0.9.8)				UD Design		
	Conn.	Buffers	Struct.	Total	Buffers	Struct	Total
512	22.9	65.0	0.3	88.2	37.0	0.2	37.2
1024	29.5	65.0	0.6	95.1	37.0	0.4	37.4
2048	42.4	65.0	1.2	107.4	37.0	0.9	37.9
4096	66.7	65.0	2.4	134.1	37.0	1.7	38.7

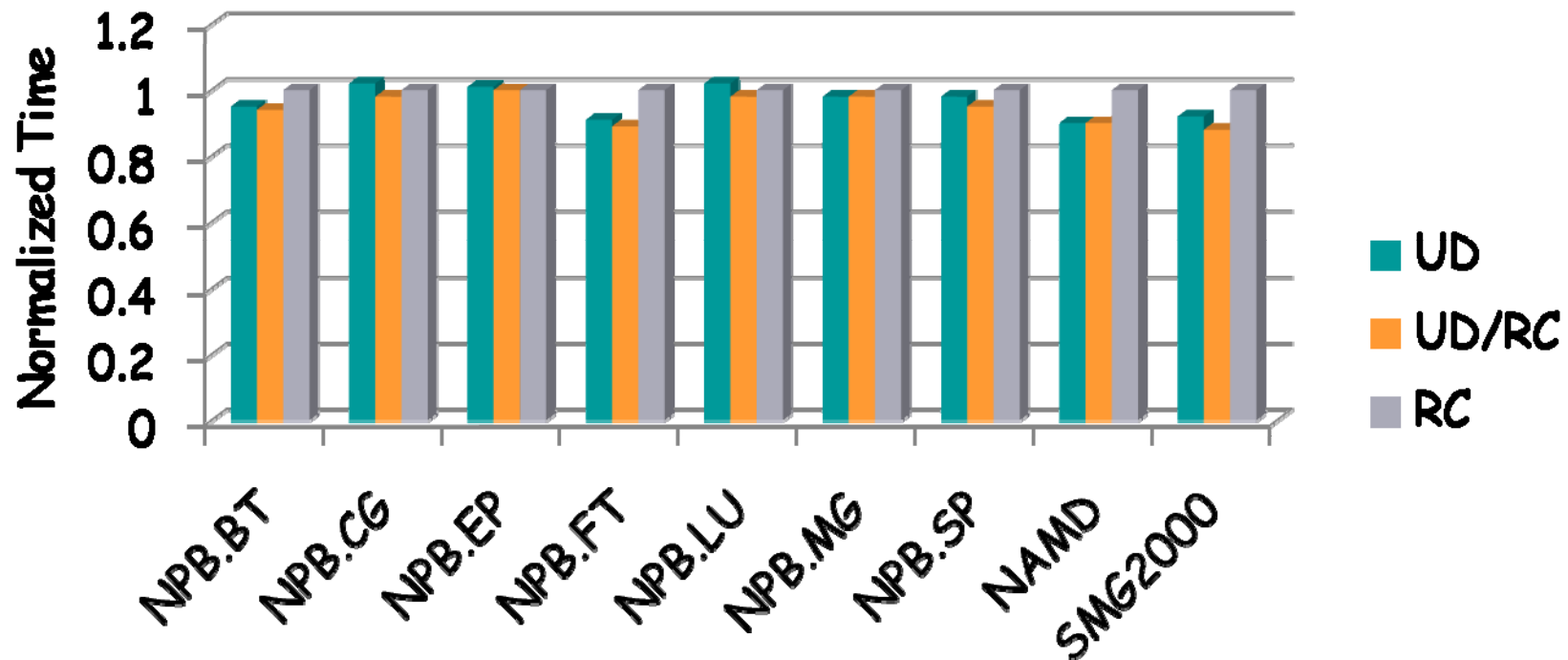
Performance



- Large number of peers per process (992 at maximum)
 - UD reduces HCA QP cache thrashing

M. Koop, S. Sur, Q. Gao and D. K. Panda, "High Performance MPI Design using Unreliable Datagram for Ultra-Scale InfiniBand Clusters," ICS '07

Impact of Hybrid RC/UD Design

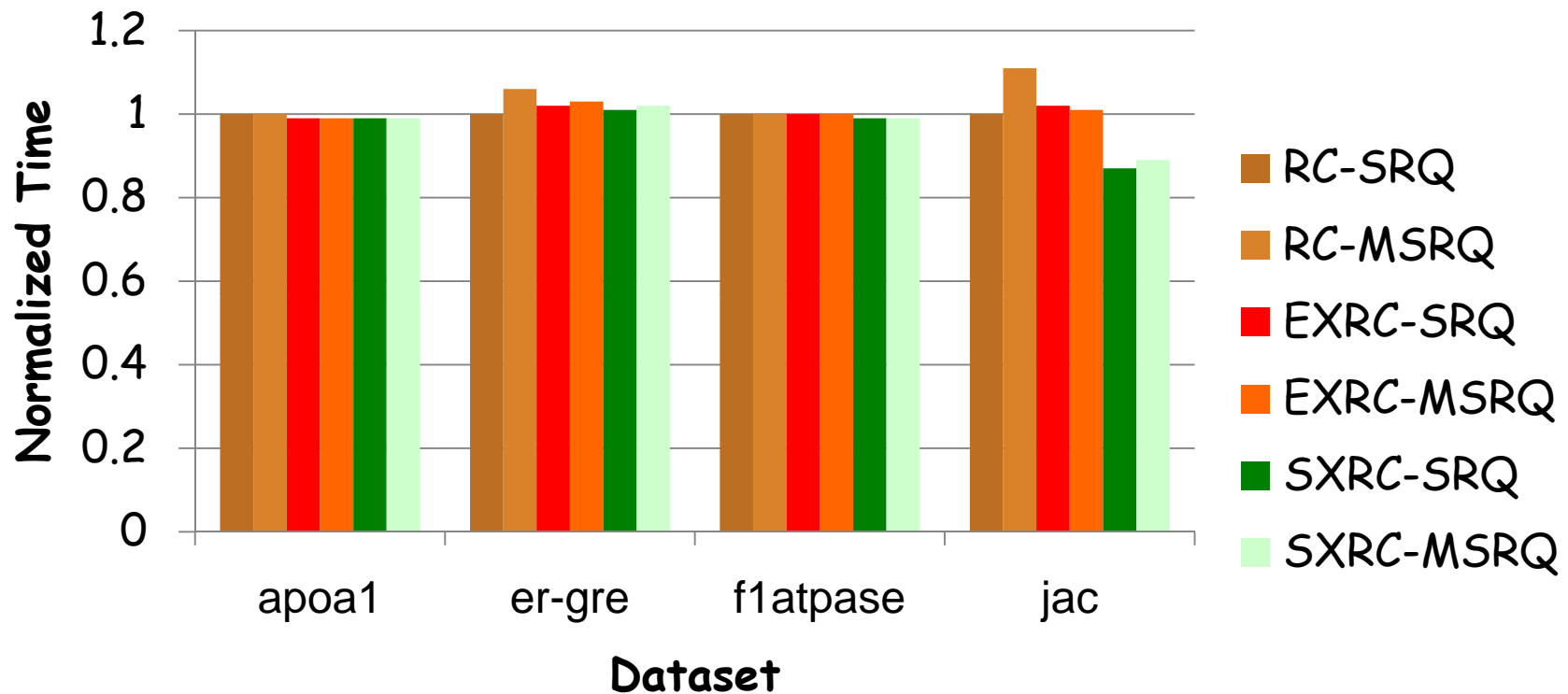


Application benchmark results on 512-core system

Combine the benefits of both RC and UD together

M. Koop, T. Jones and D. K. Panda, "MVAPICH-Aptus: Scalable High-Performance Multi-Transport MPI over InfiniBand," IPDPS '08

Impact of XRC-based Design



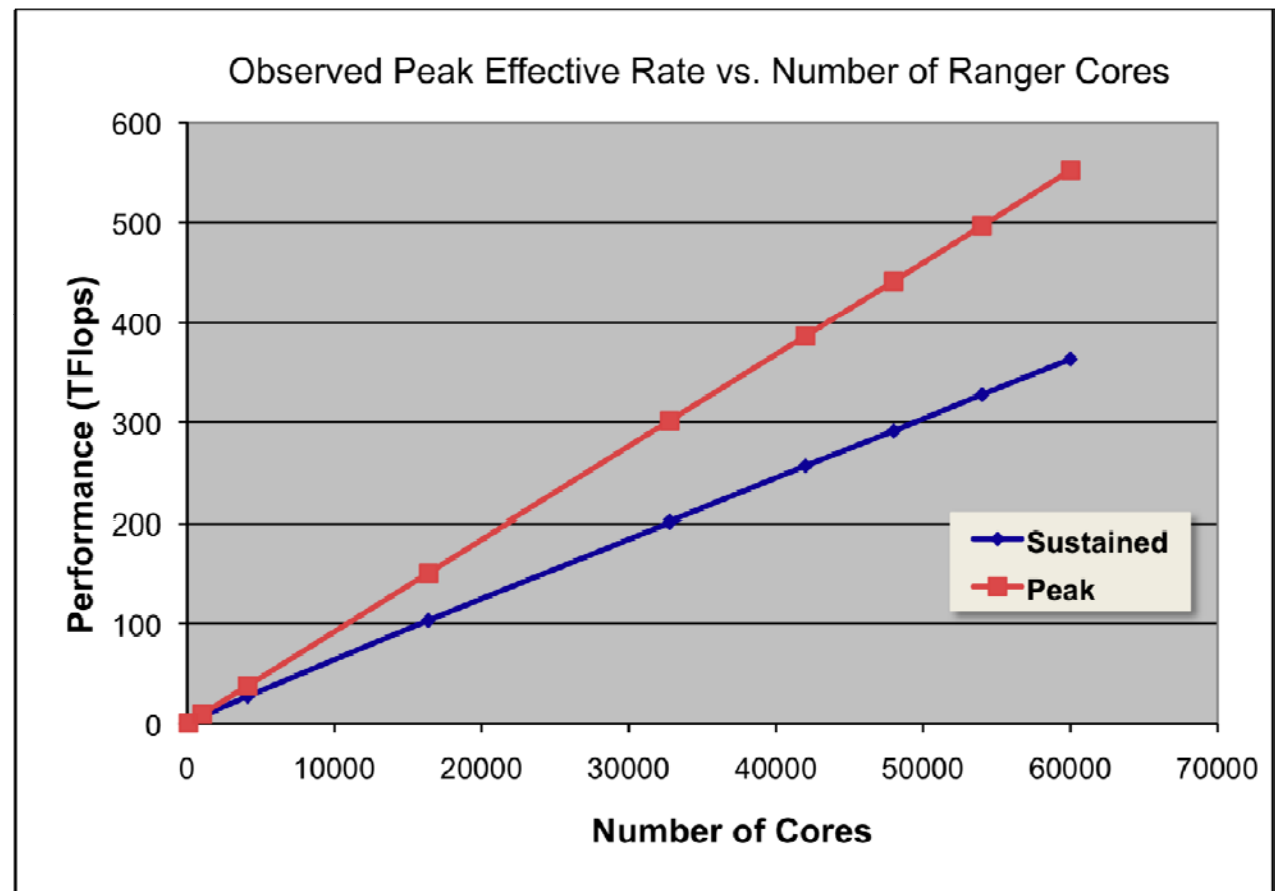
- For the *jac* dataset RC-MSRQ shows 10% worse performance
 - HCA cache is likely being thrashed
 - **SXRC modes show higher performance** since less QPs are being used (and are staying in cache)

M. Koop, J. Sridhar and D. K. Panda, "Scalable MPI Design over InfiniBand using eXtended Reliable Connection," Cluster '08

Tsukuba, Oct 2, 2008

Performance of HPC Applications on TACC Ranger using MVAPICH + IB

- Rob Farber's facial recognition application was run up to 60K cores using MVAPICH
- Ranges from 84% of peak at low end to 65% of peak at high end



http://www.tacc.utexas.edu/research/users/features/index.php?m_b_c=farber

Performance of HPC Applications on TACC Ranger: DNS/Turbulence

- 3D FFT flop count $\propto N^3 \log_2 N$

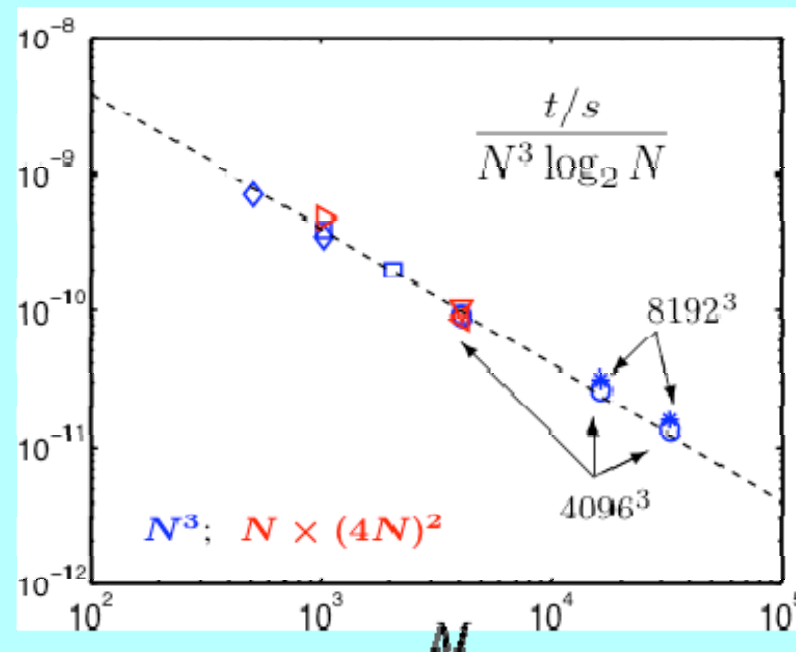
- Perfect scaling:

$$\frac{t/s}{N^3 \log_2 N} \propto M^{-1}$$

- Strong scaling:** > 98% at both 4096^3 and 8192^3 from $M = 16K$ to $32K$

- Weak scaling:** $\sim 80\%$ from $(N, M) = (2048, 2048)$ to $(8192, 32768)$

- Best timings for small M_1 : row communicator within node (16 cores) or within socket (4 cores)



Courtesy: P.K. Yeung, Diego Donzis, TG 2008

Design Insights and Sample Results

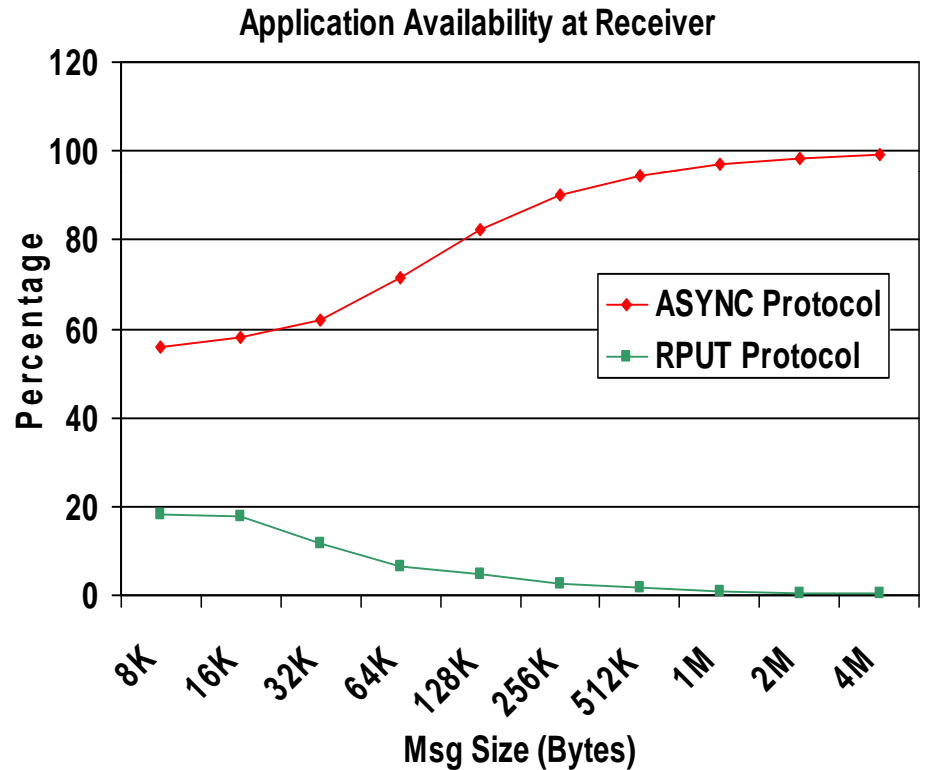
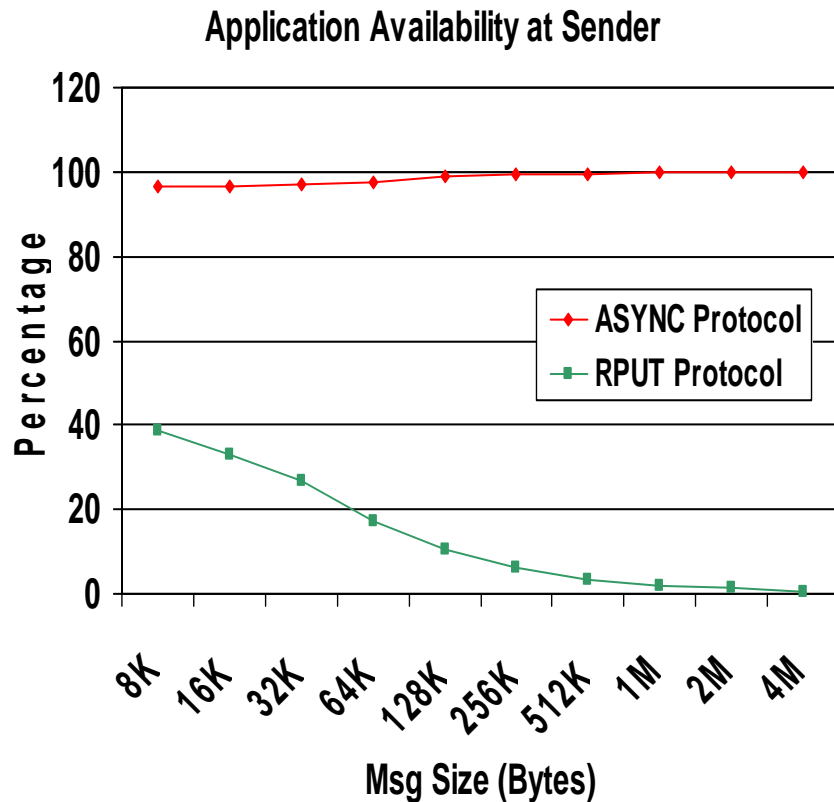
- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- Integrated Multi-rail Design
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- *Asynchronous Progress*
- Fault Tolerance

Asynchronous Progress

- Asynchronous progress (both at sender and receiver) in MVAPICH 1.0
- Design has been enhanced to a lock-free design in MVAPICH 1.1
- Potential for overlap of computation and communication

R. Kumar, A. Mamidala, M. Koop, G. Santhanaraman and D.K. Panda, [Lock-free Asynchronous Rendezvous Design for MPI Point-to-Point Communication](#), EuroPVM/MPI 2008, September 2008.

Asynchronous Progress (Mellanox DDR)



Results for SMB Benchmark



Design Insights and Sample Results

- Scalable Job Start-up
- Basic Performance
 - Two-sided Communication
 - One-sided Communication
- Multi-core-aware pt-to-pt communication
- Multi-core-aware Optimized Collective
- Integrated Multi-rail Design
- Scalability for Large-scale Systems (SRQ, UD, Hybrid & XRC)
- Applications-level Scalability
- Asynchronous Progress
- Fault Tolerance

Fault Tolerance

- Component failures are common in large-scale clusters
- Imposes need on reliability and fault tolerance
- Working along the following three angles
 - Reliable Networking with Automatic Path Migration (APM) utilizing Redundant Communication Paths (available since MVAPICH 1.0 and MVAPICH2 1.0 onward)
 - Process Fault Tolerance with Efficient Checkpoint and Restart (available since MVAPICH2 0.9.8)
 - End-to-end Reliability with memory-to-memory CRC (available since MVAPICH 0.9.9)

Network Fault-Tolerance with APM

- Network Fault Tolerance using InfiniBand Automatic Path Migration (APM)
 - Utilizes Redundant Communication Paths
 - Multiple Ports
 - LMC
- Supported in OFED 1.2

A. Vishnu, A. Mamidala, S. Narravula and D. K. Panda, "Automatic Path Migration over InfiniBand: Early Experiences", Third International Workshop on System Management Techniques, Processes, and Services, held in conjunction with IPDPS '07

Screenshots: MPI Bandwidth Test with APM

```
Shell - Konsole
Session Edit View Bookmarks Settings Help
[vishnu@d0-as4:osu_benchmarks] ../bin/mpicc osu_bw.c -o bw
[vishnu@d0-as4:osu_benchmarks] ../bin/mpirun_rsh -np 2 d0 d2 ./bw

Shell - Konsole
Session Edit View Bookmarks Settings Help
# OSU MPI Bandwidth Test (Version 2.0)
# Size      Bandwidth (MB/s)
1           0.373559
2           0.747114
4           1.490513
8           2.988996
16          5.946056
32          11.945174
64          23.590665
128         46.239120
256         93.798126
512         186.516700
1024        314.423889
2048        463.672961
4096        598.296021
8192        524.364033
16384       662.966714
32768       756.540699
65536       807.360500
131072      838.894691
myrank[0], [*] Moving to alternate path successful
myrank[1], [*] Moving to alternate path successful
262144      840.104995
524288      880.535211
1048576     885.337897
2097152     885.839118
4194304     885.855238
[vishnu@d0-as4:osu_benchmarks] █
```

```
Shell - Konsole
Session Edit View Bookmarks Settings Help
[vishnu@d0-as4:osu_benchmarks] ../bin/mpicc osu_bw.c -o bw
[vishnu@d0-as4:osu_benchmarks] ../bin/mpirun_rsh -np 2 d0 d2 ./bw
# OSU MPI Bandwidth Test (Version 2.0)
# Size      Bandwidth (MB/s)
1           0.373559
2           0.747114
4           1.490513
8           2.988996
16          5.946056
32          11.945174
64          23.590665
128         46.239120
256         93.798126
512         186.516700
1024        314.423889
2048        463.672961
4096        598.296021
8192        524.364033
16384       662.966714
32768       756.540699
65536       807.360500
131072      838.894691
myrank[0], [*] Moving to alternate path successful
myrank[1], [*] Moving to alternate path successful
262144      840.104995
█
```

Checkpoint-Restart Support in MVAPICH2

- Process-level Fault Tolerance
 - User-transparent, system-level checkpointing
 - Based on BLCR from LBNL to take coordinated checkpoints of entire program, including front end and individual processes
 - Designed novel schemes to
 - Coordinate all MPI processes to drain all in flight messages in IB connections
 - Store communication state & buffers while checkpointing
 - Restarting from the checkpoint
- Systems-level checkpoint can be initiated from the application (added in MVAPICH2 1.0)

A Running Example (Cont.)

Terminal A:
LU is running

Terminal B:
Now, Take checkpoint
Listing programs

The image shows two terminal windows side-by-side. The left window (Terminal A) shows the execution of the 'mpiexec -n 32 ./lu.C.32' command. The output displays 'NAS Parallel Benchmarks 2.2 -- LU Benchmark' with parameters: Size: 162x162x162, Iterations: 250, Number of processes: 32. It lists time steps from 1 to 120. The right window (Terminal B) shows the execution of 'mv2_checkpoint'. It displays a table of running processes:

PID	USER	TT	COMMAND	%CPU	VSZ	START	CMD
19183	gaoq	pts/0	mpiexec	0.0	18236	14:05	mpiexec -n 32 ./lu.C.32

Below the table, it prompts 'Enter PID to checkpoint or Control-C to exit: 19183', shows 'Checkpointing PID 19183', and 'Checkpoint file: context.19183'. The prompt returns to the shell.

1

2



A Running Example (Cont.)

Terminal A:

LU is not affected.
Stop it using CTRL-C

```
gaoq@cs33-gen2-~/mvapich2-0.9.8
File Edit View Terminal Tabs Help
gaoq@cs33-gen2-~/mvapich2-0.9.8
[gaoq@cs33-gen2 mvapich2-0.9.8]$ mpiexec -n 32 ./lu.C.32

NAS Parallel Benchmarks 2.2 -- LU Benchmark

Size: 162x162x162
Iterations: 250
Number of processes: 32

Time step 1
Time step 20
Time step 40
Time step 60
Time step 80
Time step 100
Time step 120
Time step 140
CTRL+C Caught... exiting
[gaoq@cs33-gen2 mvapich2-0.9.8]$
```

4

Terminal B:

Then, restart from
the checkpoint

```
gaoq@cs33-gen2-~/mvapich2-0.9.8
File Edit View Terminal Tabs Help
gaoq@cs33-gen2-~/mvapich2-0.9.8
[gaoq@cs33-gen2 mvapich2-0.9.8]$ mv2_checkpoint

PID USER      TT      COMMAND      %CPU  VSZ   START CMD
19183 gaoq      pts/0    mpiexec      0.0  18236  14:05 mpiexec -n 32 ./lu.C.32

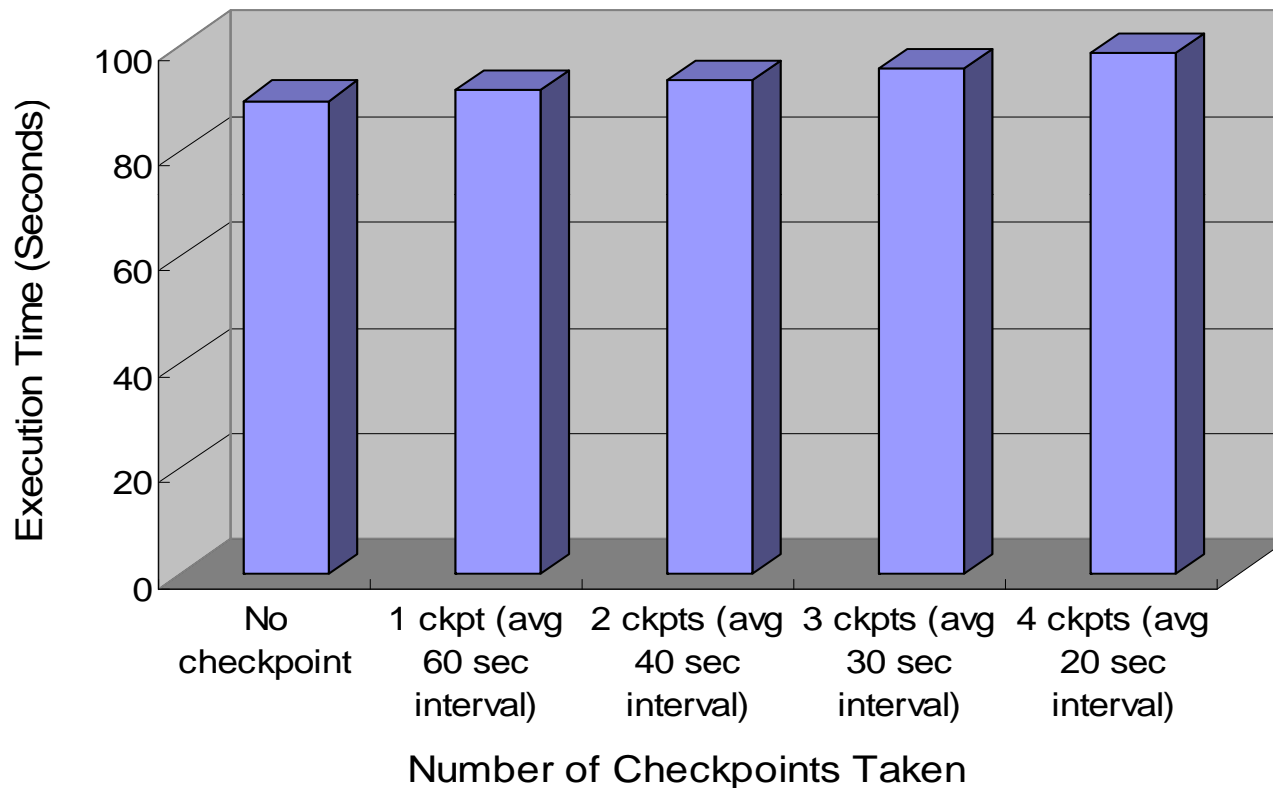
Enter PID to checkpoint or Control-C to exit: 19183
Checkpointing PID 19183
Checkpoint file: context.19183
[gaoq@cs33-gen2 mvapich2-0.9.8]$ cr_restart context.19183
mpiexec_cs33-gen2 (mpiexec 334): mpiexec: Restarting
Time step 140
Time step 160
Time step 180
Time step 200
Time step 220

```

5

Checkpoint-Restart Performance with PVFS2

NAS, LU Class C, 32x1 (Storage: 8 PVFS2 servers on IPoIB)



Q. Gao, W. Yu, W. Huang and D.K. Panda, "Application-Transparent Checkpoint/Restart for MPI over InfiniBand", ICPP '06

Presentation Overview

- Overview of InfiniBand
 - Features
 - Products (Hardware and Software)
 - Trends
- MVAPICH and MVAPICH2 Features
- Design Insights and Sample Performance Numbers
- Future Plans
- Conclusions and Final Q&A

Future Plans

- Most of the focus toward MVAPICH2
- Further enhancements to scalable job start-up
- Kernel-based (LiMIC2) shared memory pt-to-pt communication
- Optimization of collectives and one-sided communication based on new LiMIC2 shared memory communication
- Passive synchronization support for one-sided
- Flexible process binding for multi-rails
- Optimization of collectives
 - XRC
 - multi-rail
- Automatic tuning framework for pt-to-pt and collectives
- Network reliability (transparent recovery in case of adapter failure)
- Job pause-restart framework
- Performance and Memory scalability toward 100-200K cores

Conclusions

- MVAPICH and MVAPICH2 are being widely used in stable production IB clusters delivering best performance and scalability
- Also enabling clusters with 10GigE/iWARP support
- The user base stands at more than 765 organizations worldwide
- New features for scalability, high performance and fault tolerance support are aimed to deploy large-scale clusters (100-200K) nodes in the near future

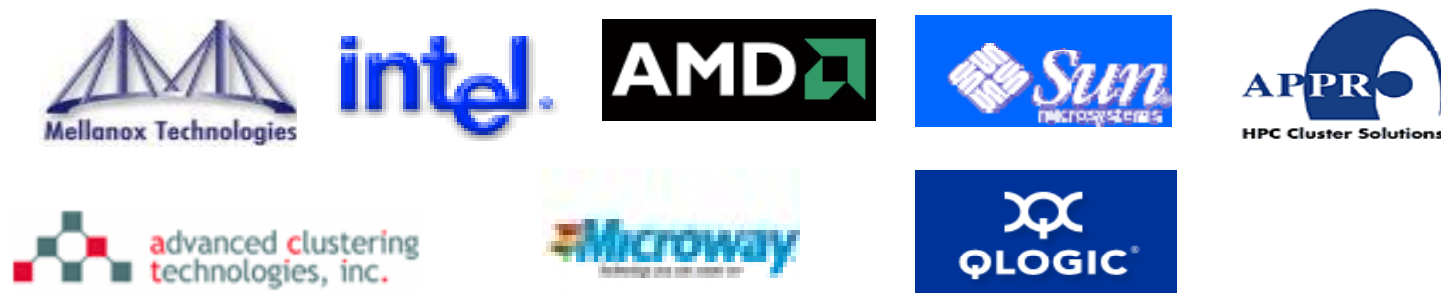
Funding Acknowledgments

Our research is supported by the following organizations

- Current Funding support by



- Current Equipment support by



Personnel Acknowledgments

Current Students

- L. Chai (Ph.D.)
- T. Gangadharappa (M. S.)
- K. Gopalakrishnan (M. S.)
- M. Koop (Ph.D.)
- P. Lai (Ph. D.)
- G. Marsh (Ph. D.)
- X. Ouyang (Ph.D.)
- G. Santhanaraman (Ph.D.)
- J. Sridhar (M. S.)
- H. Subramoni (M. S.)

Current Programmer

- J. Perkins

Past Students

- P. Balaji (Ph.D.)
- D. Buntinas (Ph.D.)
- S. Bhagvat (M.S.)
- B. Chandrasekharan (M.S.)
- W. Jiang (M.S.)
- W. Huang (Ph.D.)
- S. Kini (M.S.)
- R. Kumar (M.S.)
- S. Krishnamoorthy (M.S.)
- J. Liu (Ph.D.)
- A. Mamidala (Ph.D.)
- S. Narravula (Ph.D.)
- R. Noronha (Ph.D.)
- S. Sur (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- R. Noronha (Ph.D.)
- S. Sur (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- A. Vishnu (Ph.D.)
- J. Wu (Ph.D.)
- W. Yu (Ph.D.)