Interconnect Your Future

Paving the Road to Exascale

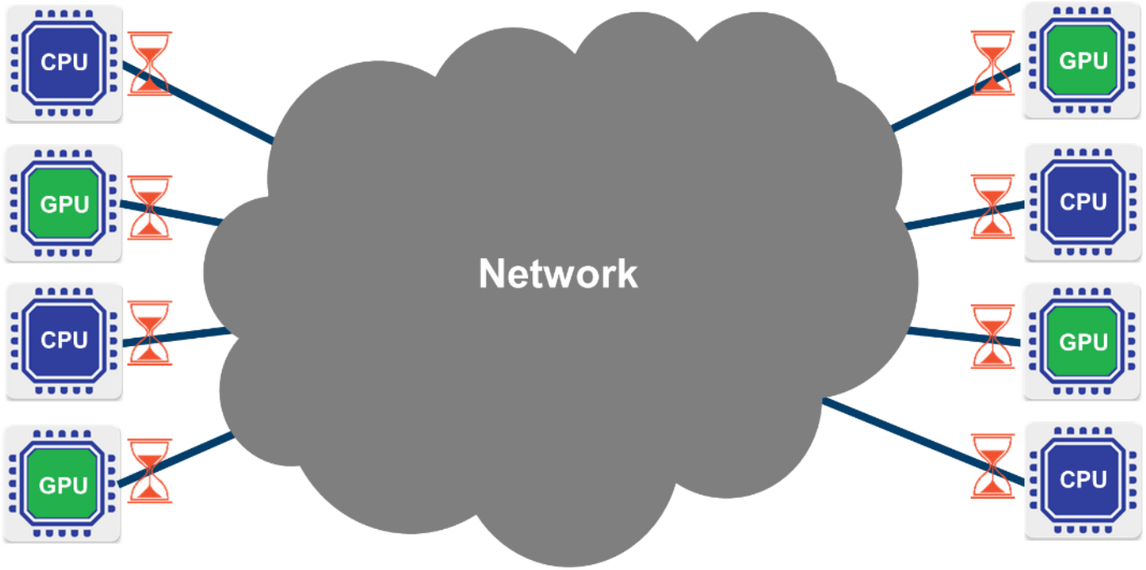
OpenSHMEM Workshop 2017

Mellanox TECHNOLOGIES

Connect. Accelerate. Outperform.™

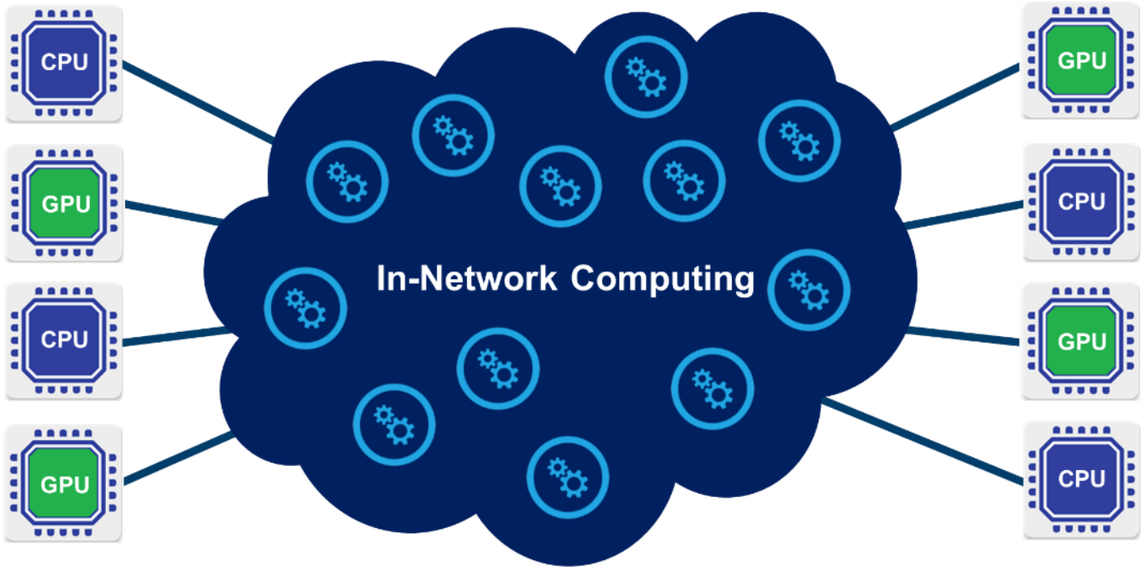
Exponential Data Growth – The Need for Intelligent and Faster Interconnect

CPU-Centric (Onload)



Must Wait for the Data
Creates Performance Bottlenecks

Data-Centric (Offload)



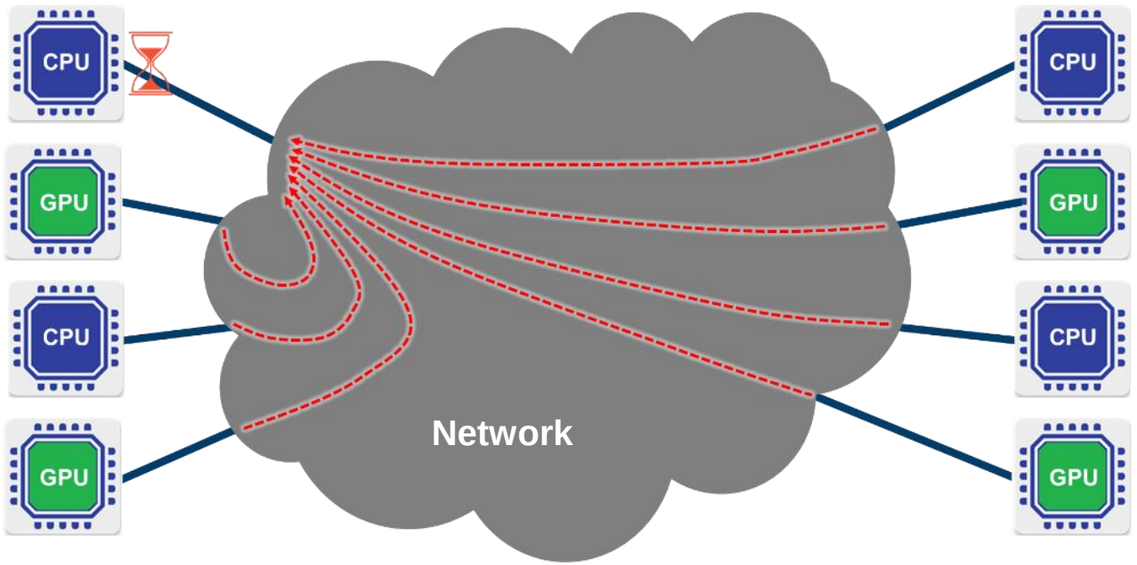
Process Data as it Moves!

Faster Data Speeds and In-Network Computing Enable Higher Performance and Scale

© 2014 Mellanox Technologies

Data Centric Architecture to Overcome Latency Bottlenecks

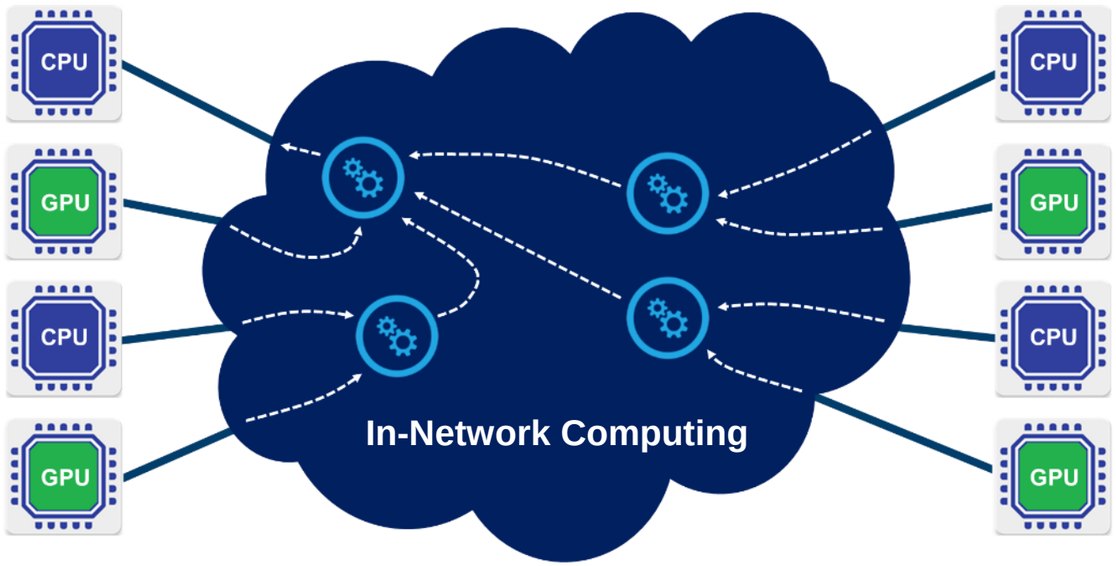
CPU-Centric (Onload)



HPC / Machine Learning

Communications Latencies of 30-40us

Data-Centric (Offload)



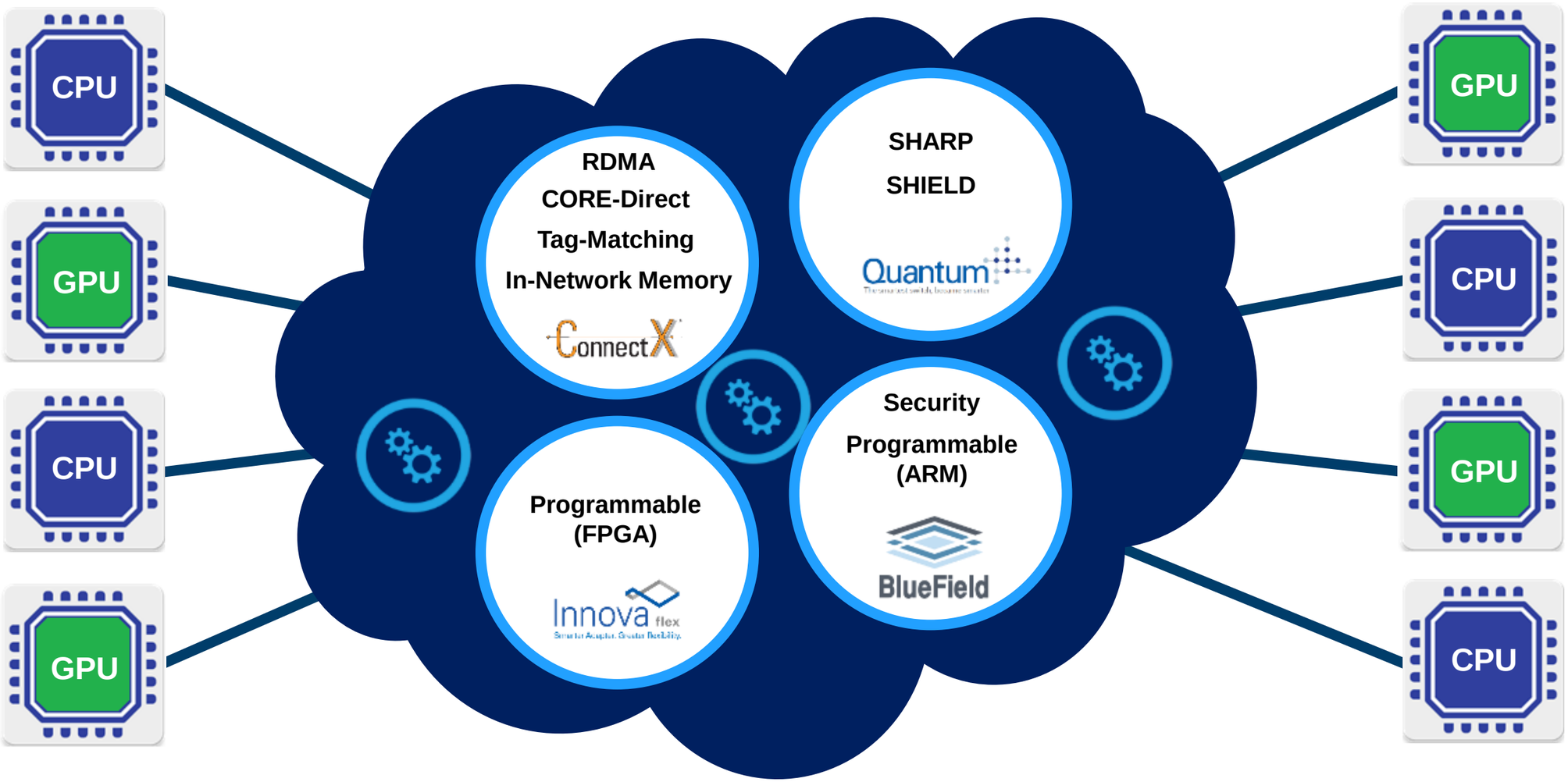
HPC / Machine Learning

Communications Latencies of 3-4us

Intelligent Interconnect Paves the Road to Exascale Performance

© 2014 Mellanox Technologies

In-Network Computing to Enable Data-Centric Data Center



In-Network Computing Key for Highest Return on Investment

© 2014 Mellanox Technologies

Highest-Performance 100/200Gb/s Interconnect Solutions

Adapters — ConnectX®-6

200Gb/s Adapter, 0.6us latency
200 million messages per second
(10 / 25 / 40 / 50 / 56 / 100 / 200Gb/s)

Switch — Quantum (The smartest switch, became smarter)

40 HDR (200Gb/s) InfiniBand Ports
80 HDR100 InfiniBand Ports
Throughput of 16Tb/s, <90ns Latency

Switch — Spectrum™

32 100GbE Ports, 64 25/50GbE Ports
(10 / 25 / 40 / 50 / 100GbE)
Throughput of 3.2Tb/s

Interconnect — LinkX™

Transceivers
Active Optical and Copper Cables
(10 / 25 / 40 / 50 / 56 / 100 / 200Gb/s)

VCSELs, Silicon Photonics and Copper

Software — HPC-X™

MPI, SHMEM/PGAS, UPC
For Commercial and Open Source Applications
Leverages Hardware Accelerations

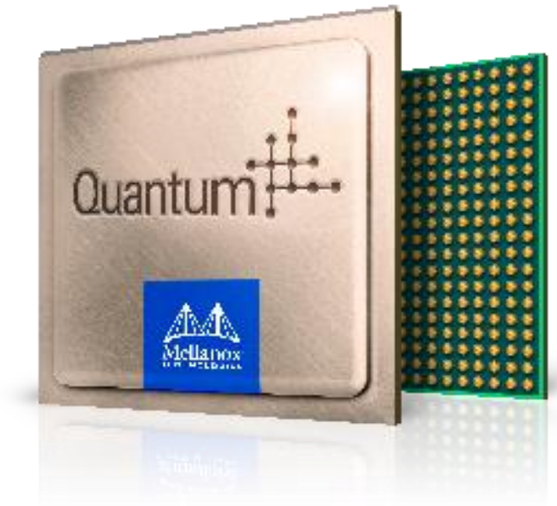
© 2014 Mellanox Technologies

Quantum 200G HDR InfiniBand Smart Switch

40 Ports of 200G HDR InfiniBand
80 Ports of 100G HDR100 InfiniBand
Switch System 800 Ports 200G, 1600 Ports 100G

16Tb/s Switch Capacity
Extremely Low Latency of 90ns
15.6 Billion Messages per Second

In-Network Computing (SHARP Technology)
Flexible Topologies (Fat-Tree, Torus, Dragonfly, etc.)
Advanced Adaptive Routing



Quantum
The smartest switch, became smarter

© 2014 Mellanox Technologies

ConnectX-6 200G HDR InfiniBand and Ethernet Smart Adapter

100/200Gb/s Throughput
0.6usec End-to-End Latency
175/200M Messages per Second

PCIe Gen3 and Gen4
Integrated PCIe Switch and Multi-Host Technology
Advanced Adaptive Routing

In-Network Computing (Collectives, Tag Matching)
In-Network Memory
Storage (NVMe), Security and Network Offloads

© 2014 Mellanox Technologies

Mellanox’s OpenSHMEM Support

Mellanox's OpenSHMEM Support: Status Update

- Major focus: Utilize Mellanox hardware capabilities to accelerate the OSHMEM implementation
  - Offload Network management
  - RDMA
  - Hardware atomic support
  - Dynamically Connected Transport
  - SHARP
  - DC/RC adaptive routing support
  - On Demand Paging
  - SHIELD
- V1.3 OpenSHMEM SPEC fully support
- SHMEM big job start improvements with PMIx
- OpenSHMEM collectives
  - SHARP supported (HCOLL)
  - OpenSHMEM shared memory collectives
- UCX support in spml UCX
- Data path performance improvements with direct verbs in UCX
  - Includes atomic support
- Added support for transparent huge page allocation
- Added support for shmem pointer
- Active participants in the OpenSHMEM community
  - Working groups of greatest interest
    - Collectives
    - Threading support

© 2014 Mellanox Technologies

SHIELD

SHIELD – Self Healing Interconnect Technology

© 2014 Mellanox Technologies

The SHIELD Self Healing Interconnect Technology

- Software-based solutions for network failures create long delays: 5-30 seconds for 1K to 10K node clusters
- During software-based recovery time, data can be lost, applications can fail
- Adaptive Routing creates further issues (failing links may act as “black holes”)
- Mellanox SHIELD technology is an innovative hardware-based solution
- SHIELD technology enables the generation of Self-Healing Interconnect
- The ability to overcome network failures by the network intelligent devices
- Accelerates network recovery time by 5000X
- Highly scalable and available for EDR and HDR solutions and beyond

SHIELD
SELF-HEALING INTERCONNECT TECHNOLOGY

Self-Healing Network Enables Unbreakable Data Centers

© 2014 Mellanox Technologies

On Demand Paging and OSHMEM Startup Optimization

The SHMEM startup problem

- **Goal**: Reduce SHMEM startup time
- Since RDMA is used, the global address space has to be registered
  - memheap + global data
- Heap registration is the dominant factor of shmem_init() time
  - Pinning and clearing the pages
  - Writing translations to the HCA
- Can take few seconds or more for large heaps
- Need to reduce memory registration time

© 2014 Mellanox Technologies

Memory Pinning: Additional Costs

- No canonical memory optimizations

Demand paging	Over commitment	Page migration
Delayed allocation	Swapping	NUMA migration
Mmap-ed files	Deduplication	Compaction
Calloc with zero page	Copy on write	Transparent huge pages

- Complicates administration and deployment
  - Unprivileged applications must be granted explicit rights to lock memory
  - Worst-case pinning in the absence of a good alternative to estimate pinning requirements
  - IO buffers limited to size of physical memory

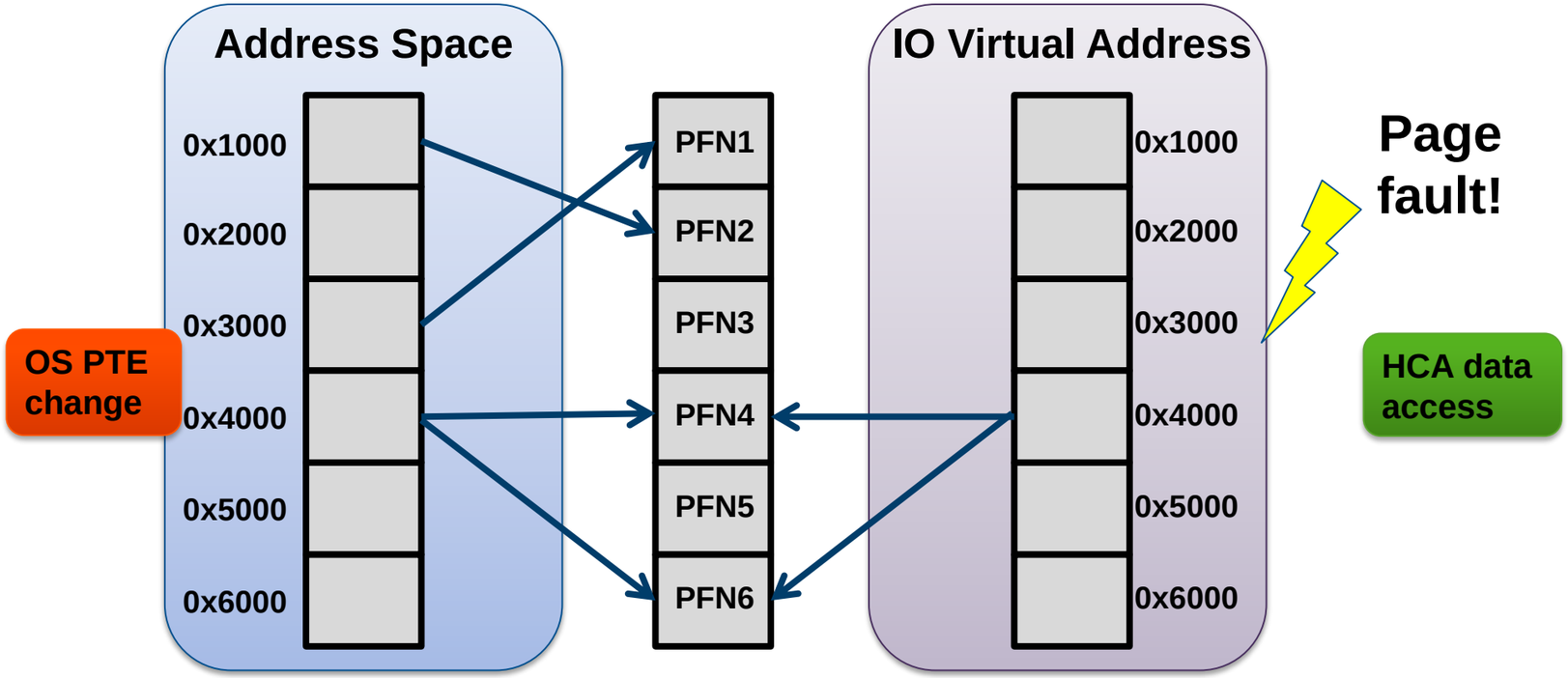
© 2014 Mellanox Technologies

On-Demand Paging (ODP)

- HCA translation tables may contain non-present pages
  - Initially, a new MR is created with non-present pages
  - Virtual memory mappings don't necessarily exist
- MR pages are ***never*** pinned by the OS
  - Paged in when HCA needs them
  - Paged out when reclaimed by the OS
- Eliminates the price of pinning
  - Unlimited MR sizes
    - No need for special privileges
  - Physical memory optimized to hold current working set
    - For both CPU and IO access
  - Application pages may be migrated at will

© 2014 Mellanox Technologies

ODP Operation

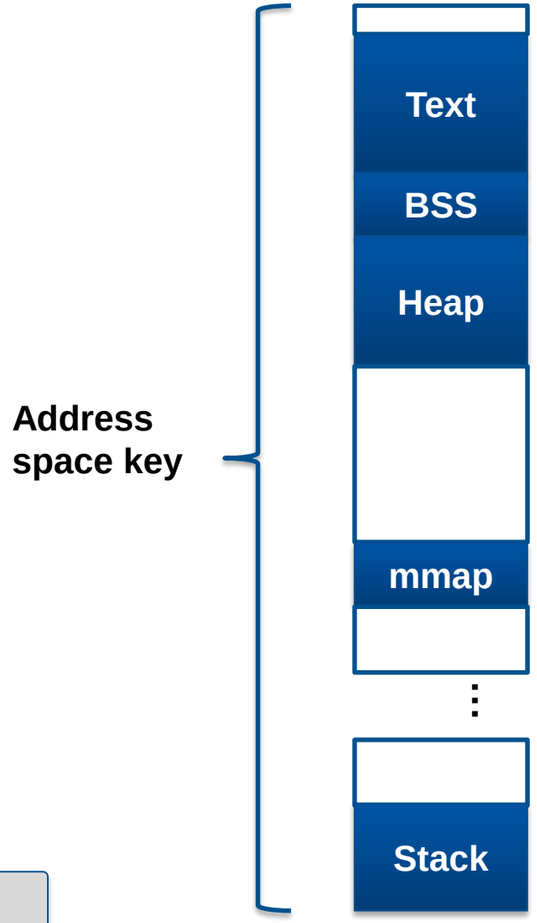


The ODP promise:
IO virtual address mapping == Process virtual address mapping

© 2014 Mellanox Technologies

ODP: The Global Address Space Key

- Register the whole process address space with a ***single*** key
  - MR covers existing and future memory mappings
- MR covers unmapped address ranges
  - Permissions checked at access (page fault) time
    - VMA permissions
    - MR access rights
  - RDMA access rights revoked upon invalidation or permission changes
- Granular remote permissions via Memory Windows
  - User-space equivalent for Fast Registration Work Requests…
- Eliminates the price of memory management
  - All data transfer done based on the address space key
  - No need to register and track any other MRs (!)




```
mr = ibv_reg_mr(pd, NULL, -1, IBV_ACCESS_LOCAL_WRITE | IBV_ACCESS_ON_DEMAND);
```

© 2014 Mellanox Technologies

ODP: Memory Prefetching

- Pre-faults (populates) ODP MRs

- Best effort hint
  - Not necessarily all pages are pre-fetched
  - No guarantees that pages remain resident
  - Asynchronous
    - Can be invoked opportunistically in parallel to IO

- Use cases
  - Avoid multiple page faults by small transactions
  - Pre-fault a large region about to be accessed by IO

- EFAULT returned when
  - Range exceeds the MR
  - Requested range not mapped to address space

```
struct ibv_prefetch_attr {
        uint32_t comp_mask;
        int flags; /* IBV_ACCESS_LOCAL_WRITE */
        void *addr;
        size_t length;
};

int ibv_prefetch_mr(struct ibv_mr *mr,
                    struct ibv_prefetch_attr *attr,
                    size_t attr_size);
```

© 2014 Mellanox Technologies

SHMEM and ODP

- Register the heap with ODP instead of pinning
  - Completes much faster than traditional registration
  - shmem_init() time is reduced significantly

- Optimization: background ODP-prefetch
  - High rate of ODP faults may affect runtime performance
  - Driver maps pages in the background while application continues to run
  - A page would mostly likely be mapped before accessed by the application
  - Small amount of page faults in the common case

- Application Flow
  - Application allocates memory with shmalloc()
  - SHMEM asks UCX to prefetch the allocated memory
    - Calling ucp_mem_advise(UCP_MADV_WILLNEED)
  - UCX issues ibv_exp_prefetch_mr() verb
  - Driver schedules background work to:
    - get_user_pages() on the allocated memory
    - Issue UMR to update HW translation tables

© 2014 Mellanox Technologies

Performance results

- Measuring shmem_init() time for an applications which uses 180G per node
  - Skipping few warmup iterations
- Setup details:
  - 64 nodes, 32 processes per node
  - Intel(R) Xeon(R) CPU E5-2697A v4 @ 2.60GHz
  - DDR4 2400 MHz
  - ConnectX-4 EDR, Fat tree

Iteration #	Default startup time (seconds)	ODP-optimized startup time (seconds)
1	5.308	2.209
2	5.287	2.205
3	5.244	2.228
4	5.235	2.359
5	5.507	2.287
6	5.321	2.086
7	5.282	2.164
8	5.263	2.266
9	5.273	2.283
10	5.222	2.242

© 2014 Mellanox Technologies

Alltoall – Small data Aggregation

Algorithm Overview

- Newly developed algorithm – aggregation based (Patent pending)
- Use hardware scatter/gather to reduce memory copies

© 2014 Mellanox Technologies

All-to-All latencies – 8 bytes

- 30 nodes, 20 processes per node

Implementation	Latency (usec)
Open MPI - Pairwise	202.35
Open MPI - Bruck	175.11
Open MPI – Bruck + Shared memory	105.44
IntelMPI + Shared Memory	117.39
New Algorithm	95.98

© 2014 Mellanox Technologies

Broadcast Collective Operation Optimization

Key Features

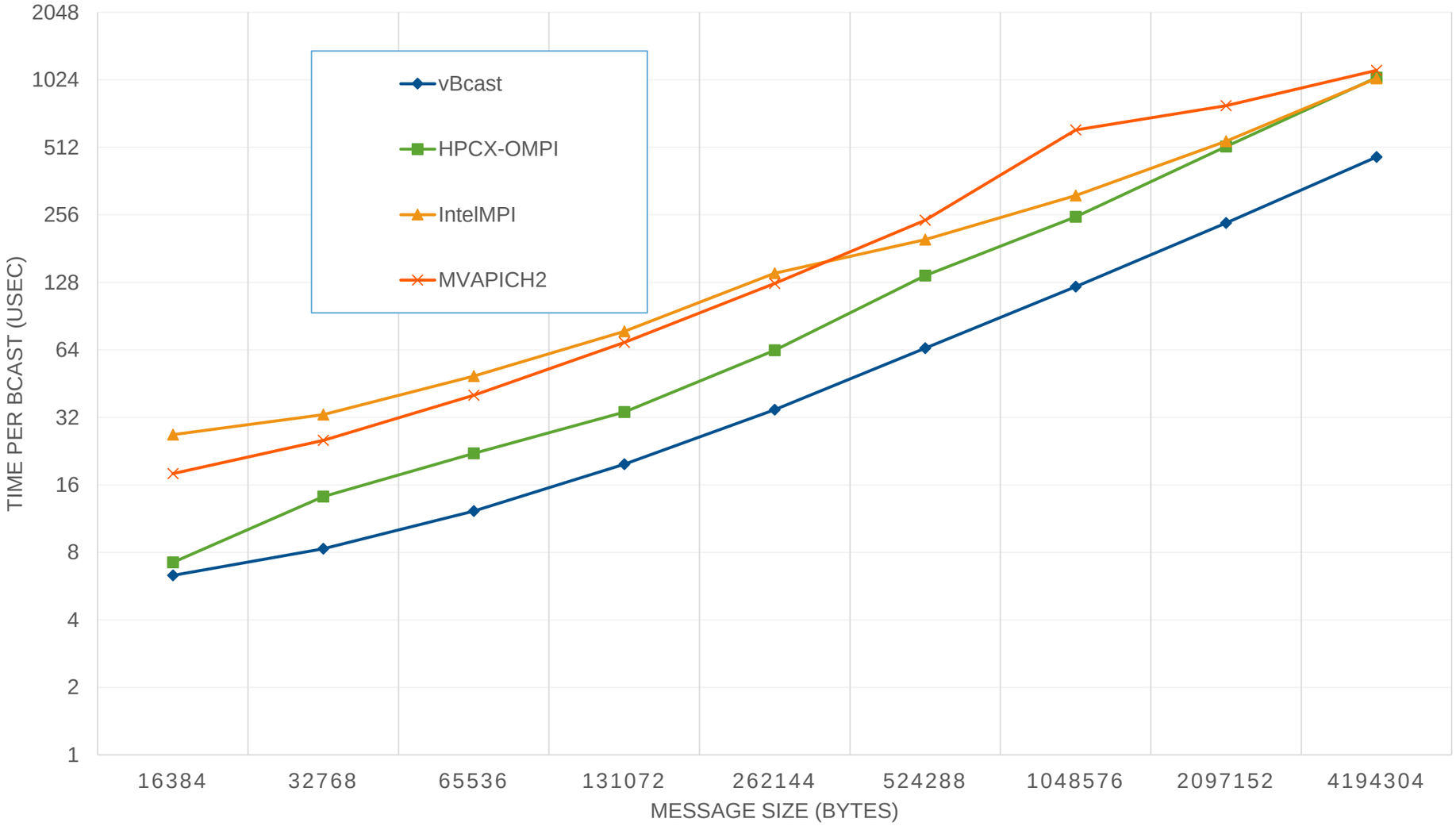
- Hardware multicast: capability to replicate packets at the switch – Order (1) cost
- However “unreliable”, needs reliability protocol.
- Attempted before: always duplicating data through reliable channel – either before [1], or after [2] the multicast.
- Target: No duplication, O(1) good case, O(logF) for F failures.

[1] Liu, Jiuxing, Amith R. Mamidala, and Dhabaleswar K. Panda. "Fast and scalable MPI-level broadcast using InfiniBand's hardware multicast support." *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*. IEEE, 2004.

[2] Hoefler, Torsten, Christian Siebert, and Wolfgang Rehm. "A practically constant-time MPI Broadcast Algorithm for large-scale InfiniBand Clusters with Multicast." *Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International*. IEEE, 2007.

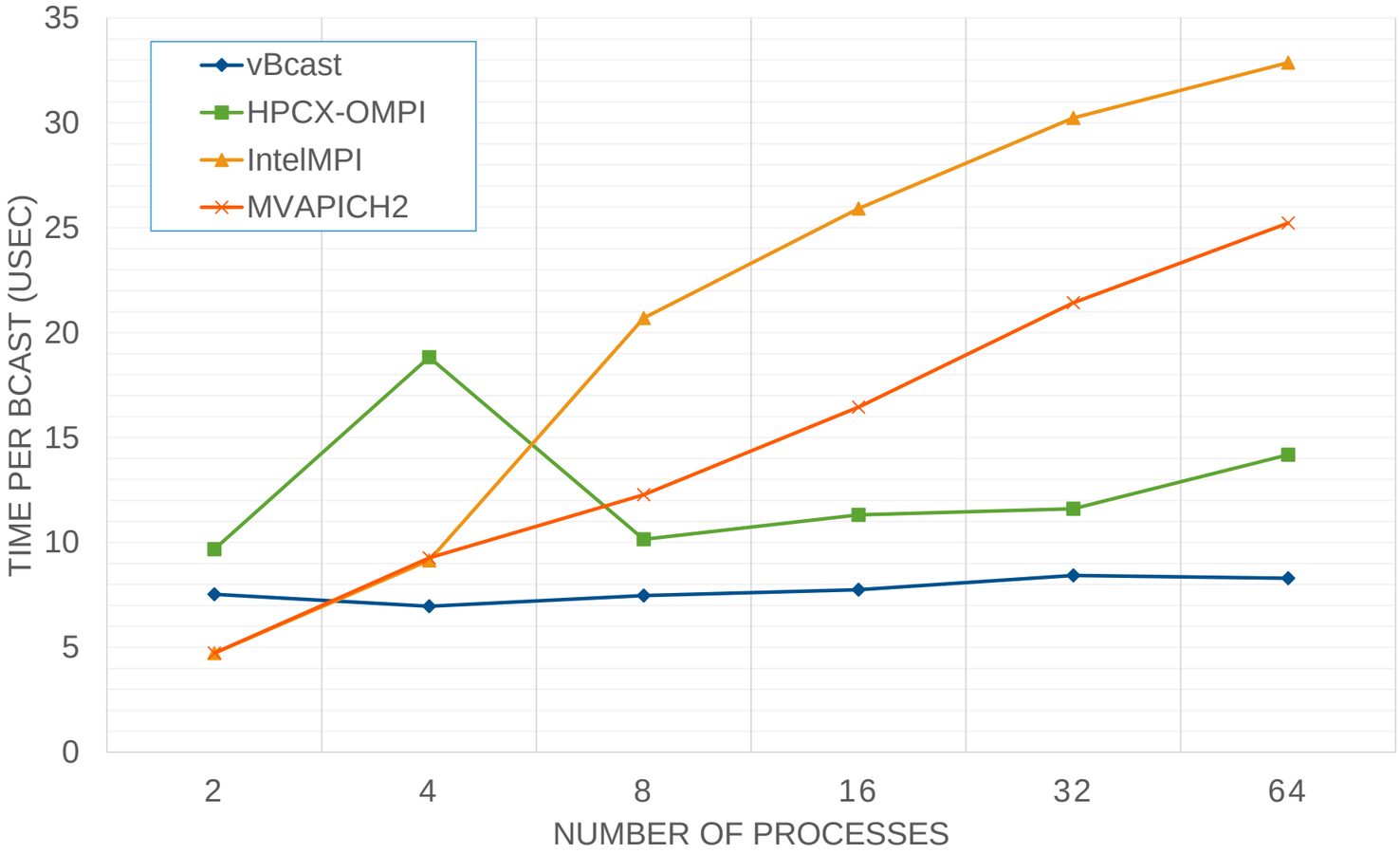
© 2014 Mellanox Technologies

64 Processes, Varying Data Size



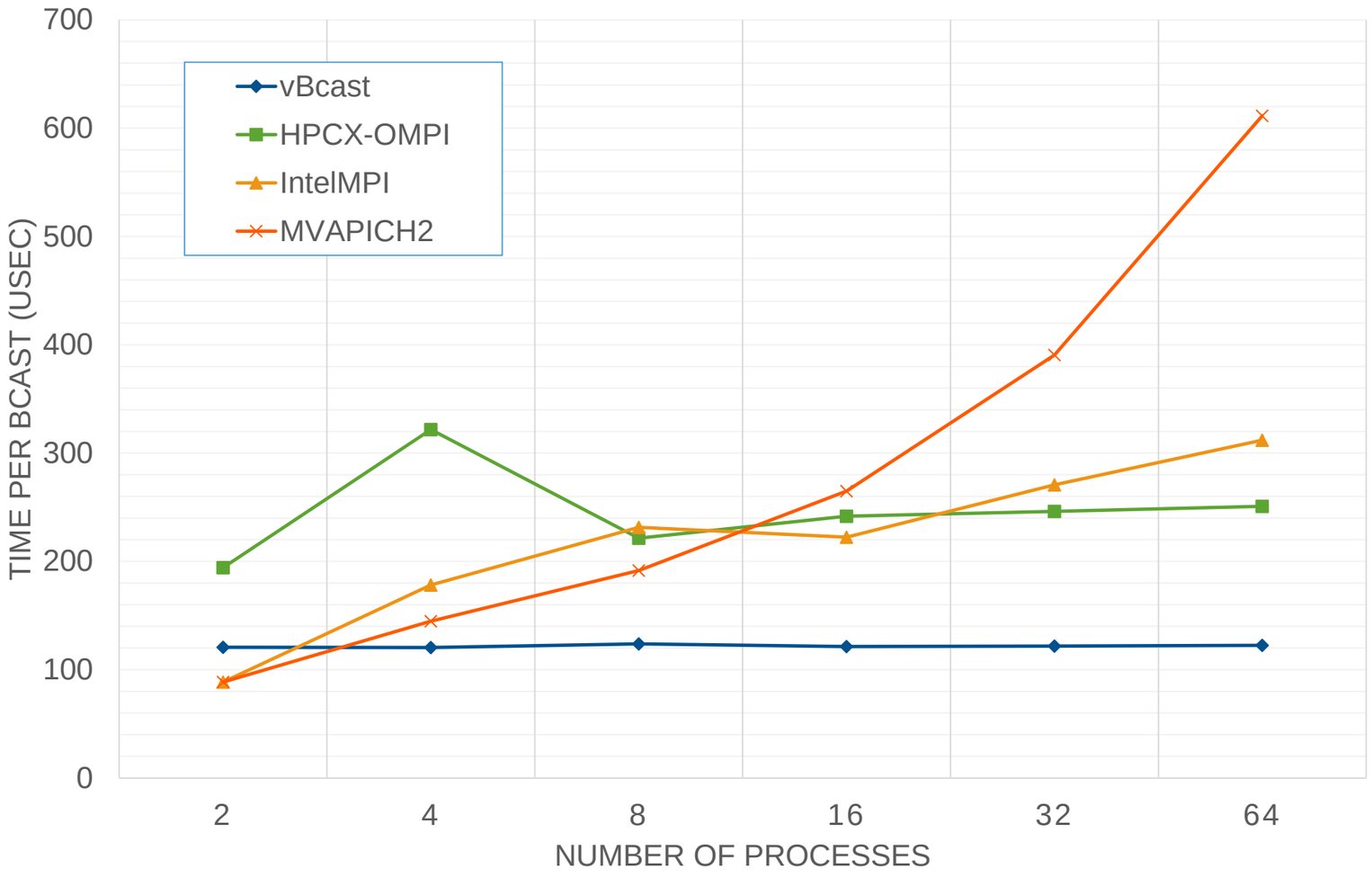
© 2014 Mellanox Technologies

32KB Varying Number of Processes



© 2014 Mellanox Technologies

1MB Varying Number of Processes



© 2014 Mellanox Technologies

Scalable Hierarchical Aggregation and Reduction Protocol (SHARP)

Compute in the Interior of the network

Feature Description

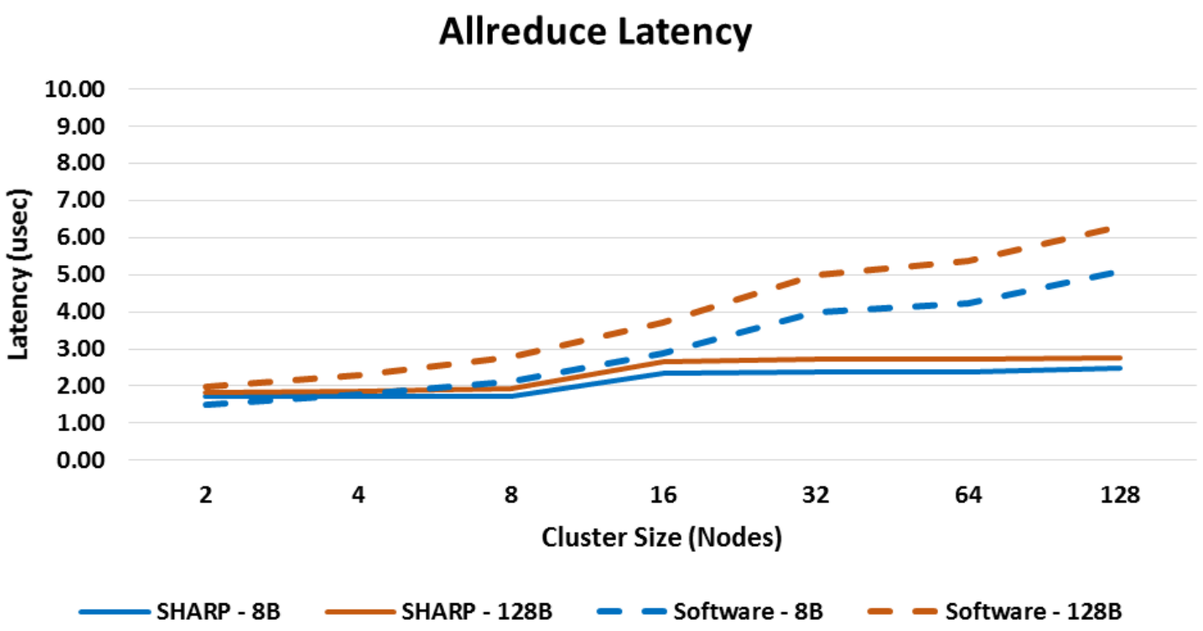
- Reliable Scalable General Purpose Primitive, Applicable to Multiple Use-cases
  - In-network Tree based aggregation mechanism
  - Large number of groups
  - Many simultaneous outstanding operations in flight

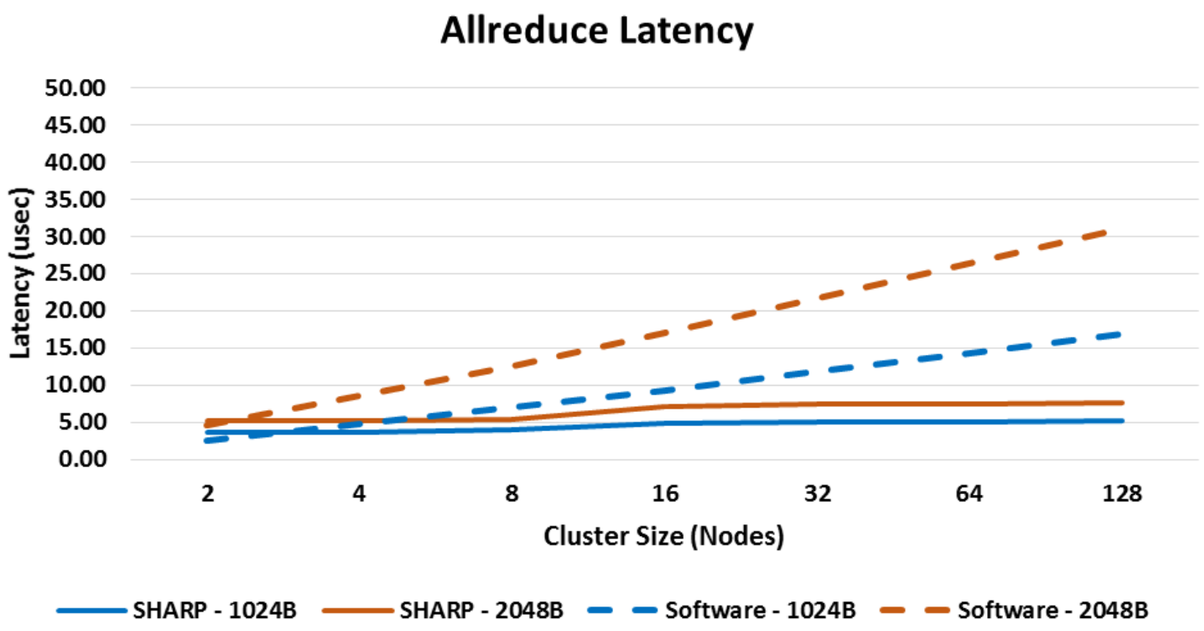
Accelerating HPC applications

- Scalable High Performance Collective Offload
  - Barrier, Reduce, All-Reduce
  - Sum, Min, Max, Min-loc, max-loc, OR, XOR, AND
  - Integer and Floating-Point
  - Repeatable results
- Significantly reduce MPI collective runtime
- Increase CPU availability and efficiency
- Enable communication and computation overlap

© 2014 Mellanox Technologies

SHARP Allreduce Performance Advantages





SHARP enables 75% Reduction in Latency
Providing Scalable Flat Latency

© 2014 Mellanox Technologies

Thank You

Mellanox TECHNOLOGIES

Connect. Accelerate. Outperform.™