InfiniBand and High-speed Ethernet for Dummies

A Tutorial at SC ’10

by

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

Pavan Balaji
Argonne National Laboratory
E-mail: balaji@mcs.anl.gov
http://www.mcs.anl.gov/~balaji

Sayantan Sur
The Ohio State University
E-mail: surs@cse.ohio-state.edu
http://www.cse.ohio-state.edu/~surs
Presentation Overview

• Introduction
• Why InfiniBand and High-speed Ethernet?
• Overview of IB, HSE, their Convergence and Features
• IB and HSE HW/SW Products and Installations
• Sample Case Studies and Performance Numbers
• Conclusions and Final Q&A
Current and Next Generation Applications and Computing Systems

• Growth of High Performance Computing
  – Growth in processor performance
    • Chip density doubles every 18 months
  – Growth in commodity networking
    • Increase in speed/features + reducing cost

• Clusters: popular choice for HPC
  – Scalability, Modularity and Upgradeability
Trends for Computing Clusters in the Top 500 List (http://www.top500.org)

<table>
<thead>
<tr>
<th>Month</th>
<th>Nov. 1996: 0/500 (0%)</th>
<th>Nov. 2001: 43/500 (8.6%)</th>
<th>Nov. 2006: 361/500 (72.2%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov. 1997: 1/500 (0.2%)</td>
<td>Jun. 2002: 80/500 (16%)</td>
<td>Jun. 2007: 373/500 (74.6%)</td>
<td></td>
</tr>
<tr>
<td>Nov. 1997: 1/500 (0.2%)</td>
<td>Nov. 2002: 93/500 (18.6%)</td>
<td>Nov. 2007: 406/500 (81.2%)</td>
<td></td>
</tr>
<tr>
<td>Jun. 1998: 1/500 (0.2%)</td>
<td>Jun. 2003: 149/500 (29.8%)</td>
<td>Jun. 2008: 400/500 (80.0%)</td>
<td></td>
</tr>
<tr>
<td>Nov. 1998: 2/500 (0.4%)</td>
<td>Nov. 2003: 208/500 (41.6%)</td>
<td>Nov. 2008: 410/500 (82.0%)</td>
<td></td>
</tr>
<tr>
<td>Jun. 1999: 6/500 (1.2%)</td>
<td>Jun. 2004: 291/500 (58.2%)</td>
<td>Jun. 2009: 410/500 (82.0%)</td>
<td></td>
</tr>
<tr>
<td>Nov. 1999: 7/500 (1.4%)</td>
<td>Nov. 2004: 294/500 (58.8%)</td>
<td>Nov. 2009: 417/500 (83.4%)</td>
<td></td>
</tr>
<tr>
<td>Jun. 2000: 11/500 (2.2%)</td>
<td>Jun. 2005: 304/500 (60.8%)</td>
<td>Jun. 2010: 424/500 (84.8%)</td>
<td></td>
</tr>
<tr>
<td>Nov. 2000: 28/500 (5.6%)</td>
<td>Nov. 2005: 360/500 (72.0%)</td>
<td>Nov. 2010: To be announced</td>
<td></td>
</tr>
<tr>
<td>Jun. 2001: 33/500 (6.6%)</td>
<td>Jun. 2006: 364/500 (72.8%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Integrated High-End Computing Environments

Enterprise Multi-tier Datacenter for Visualization and Mining

- Tier 1: Routers/Servers
- Tier 2: Application Server, Switch
- Tier 3: Database Server, Switch

Storage cluster

- Meta-Data Manager
- I/O Server Node
- Database Server

LAN/WAN

Compute cluster

- Frontend
- LAN
- I/O Server Node

Database Server

Application Server

Switch
Cloud Computing Environments

LAN / WAN

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Virtual Machine
Virtual Machine
Physical Machine

Physical Meta-Data Manager

Meta Data

Physical I/O Server Node

Data

Physical I/O Server Node

Data

Physical I/O Server Node

Data

Physical I/O Server Node

Data

Virtual Network File System

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data

I/O Server Node

Data
Networking and I/O Requirements

- Good Systems Area Network with excellent performance (low latency, high bandwidth and low CPU utilization) for inter-processor communication (IPC) and I/O
- Good Storage Area Networks high performance I/O
- Good WAN connectivity in addition to intra-cluster SAN/LAN connectivity
- Quality of Service (QoS) for interactive applications
- RAS (Reliability, Availability, and Serviceability)
- With low cost
Major Components in Computing Systems

- Hardware components
  - Processing cores and memory subsystem
  - I/O bus or links
  - Network adapters/switches
- Software components
  - Communication stack
- **Bottlenecks can artificially limit the network performance the user perceives**
Processing Bottlenecks in Traditional Protocols

- Ex: TCP/IP, UDP/IP
- Generic architecture for all networks
- Host processor 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 on packet arrival or transmission
  - Software signals between different layers to handle protocol processing in different priority levels
Bottlenecks in Traditional I/O Interfaces and Networks

- Traditionally relied on bus-based technologies (last mile bottleneck)
  - E.g., PCI, PCI-X
  - One bit per wire
  - Performance increase through:
    - Increasing clock speed
    - Increasing bus width
  - Not scalable:
    - Cross talk between bits
    - Skew between wires
    - Signal integrity makes it difficult to increase bus width significantly, especially for high clock speeds

<table>
<thead>
<tr>
<th></th>
<th>Year</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI</td>
<td>1990</td>
<td>33MHz/32bit: 1.05Gbps</td>
</tr>
<tr>
<td>PCI-X</td>
<td>1998</td>
<td>133MHz/64bit: 8.5Gbps</td>
</tr>
<tr>
<td>PCI-X</td>
<td>2003</td>
<td>266-533MHz/64bit: 17Gbps</td>
</tr>
</tbody>
</table>
Bottlenecks on Traditional Networks

- Network speeds saturated at around 1Gbps
  - Features provided were limited
  - Commodity networks were not considered scalable enough for very large-scale systems

<table>
<thead>
<tr>
<th>Network Type</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet (1979 - )</td>
<td>10 Mbit/sec</td>
</tr>
<tr>
<td>Fast Ethernet (1993 - )</td>
<td>100 Mbit/sec</td>
</tr>
<tr>
<td>Gigabit Ethernet (1995 - )</td>
<td>1000 Mbit/sec</td>
</tr>
<tr>
<td>ATM (1995 - )</td>
<td>155/622/1024 Mbit/sec</td>
</tr>
<tr>
<td>Myrinet (1993 - )</td>
<td>1 Gbit/sec</td>
</tr>
<tr>
<td>Fibre Channel (1994 - )</td>
<td>1 Gbit/sec</td>
</tr>
</tbody>
</table>
Motivation for InfiniBand and High-speed Ethernet

- Industry Networking Standards
- InfiniBand and High-speed Ethernet were introduced into the market to address these bottlenecks
- InfiniBand aimed at all three bottlenecks (protocol processing, I/O bus, and network speed)
- Ethernet aimed at directly handling the network speed bottleneck and relying on complementary technologies to alleviate the protocol processing and I/O bus bottlenecks
Presentation Overview

• Introduction

• Why InfiniBand and High-speed Ethernet?

• Overview of IB, HSE, their Convergence and Features

• IB and HSE HW/SW Products and Installations

• Sample Case Studies and Performance Numbers

• Conclusions and Final Q&A
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](http://www.infinibandta.org)
High-speed Ethernet Consortium (10GE/40GE/100GE)

- 10GE Alliance formed by several industry leaders to take the Ethernet family to the next speed step
- Goal: To achieve a scalable and high performance communication architecture while maintaining backward compatibility with Ethernet
- http://www.ethernetalliance.org
- 40-Gbps (Servers) and 100-Gbps Ethernet (Backbones, Switches, Routers): IEEE 802.3 WG
- Energy-efficient and power-conscious protocols
  - On-the-fly link speed reduction for under-utilized links
Tackling Communication Bottlenecks with IB and HSE

- Network speed bottlenecks
- Protocol processing bottlenecks
- I/O interface bottlenecks
Network Bottleneck Alleviation: InfiniBand ("Infinite Bandwidth") and High-speed Ethernet (10/40/100 GE)

- Bit serial differential signaling
  - Independent pairs of wires to transmit independent data (called a lane)
  - Scalable to any number of lanes
  - Easy to increase clock speed of lanes (since each lane consists only of a pair of wires)

- Theoretically, no perceived limit on the bandwidth
## Network Speed Acceleration with IB and HSE

<table>
<thead>
<tr>
<th>Network Type</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet (1979 - )</td>
<td>10 Mbit/sec</td>
</tr>
<tr>
<td>Fast Ethernet (1993 -)</td>
<td>100 Mbit/sec</td>
</tr>
<tr>
<td>Gigabit Ethernet (1995 -)</td>
<td>1000 Mbit /sec</td>
</tr>
<tr>
<td>ATM (1995 -)</td>
<td>155/622/1024 Mbit/sec</td>
</tr>
<tr>
<td>Myrinet (1993 -)</td>
<td>1 Gbit/sec</td>
</tr>
<tr>
<td>Fibre Channel (1994 -)</td>
<td>1 Gbit/sec</td>
</tr>
<tr>
<td>InfiniBand (2001 -)</td>
<td>2 Gbit/sec (1X SDR)</td>
</tr>
<tr>
<td>10-Gigabit Ethernet (2001 -)</td>
<td>10 Gbit/sec</td>
</tr>
<tr>
<td>InfiniBand (2003 -)</td>
<td>8 Gbit/sec (4X SDR)</td>
</tr>
<tr>
<td>InfiniBand (2005 -)</td>
<td>16 Gbit/sec (4X DDR)</td>
</tr>
<tr>
<td>InfiniBand (2007 -)</td>
<td>24 Gbit/sec (12X SDR)</td>
</tr>
<tr>
<td>40-Gigabit Ethernet (2010 -)</td>
<td>32 Gbit/sec (4X QDR)</td>
</tr>
<tr>
<td>InfiniBand (2011 -)</td>
<td>54.4 Gbit/sec (4X FDR)</td>
</tr>
<tr>
<td>InfiniBand (2012 -)</td>
<td>100 Gbit/sec (4X EDR)</td>
</tr>
</tbody>
</table>

20 times in the last 9 years
InfiniBand Link Speed Standardization Roadmap

<table>
<thead>
<tr>
<th># of Lanes per direction</th>
<th>Per Lane &amp; Rounded Per Link Bandwidth (Gb/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Per Lane &amp; Rounded Per Link Bandwidth (Gb/s)</td>
</tr>
<tr>
<td>12</td>
<td>48+48</td>
</tr>
<tr>
<td>8</td>
<td>32+32</td>
</tr>
<tr>
<td>4</td>
<td>16+16</td>
</tr>
<tr>
<td>1</td>
<td>4+4</td>
</tr>
</tbody>
</table>

NDR = Next Data Rate
HDR = High Data Rate
EDR = Enhanced Data Rate
FDR = Fourteen Data Rate
QDR = Quad Data Rate
DDR = Double Data Rate
SDR = Single Data Rate (not shown)
A Note on Data Encoding

• All communication channels utilize data encoding
  – Imbalance in the number of 1’s and 0’s being transmitted
    • To achieve DC-balancing and allowing for clock recovery
      – DC-balancing: bit errors can occur when a (relatively) long series of 1’s (charges the capacitor of the high-pass filter)
      – Clock recovery: Data receiver phase-aligns its clock to the transitions in the data stream. This only works when there are enough 0-1 or 1-0 transitions.
    – Converts data into a format with more uniform 1’s and 0’s
• Many networks so far (1GE, Myrinet, Quadrics) used 8b/10b encoding
• New networks (IB (post-QDR), HSE (&gt;= 10GE) use 64b/66b encoding

• The eternal IB confusion:
  – All networks other than IB specify data rate (1-Gigabit Ethernet == 1Gbps data rate)
  – IB initially broke this convention -- when IB (up to QDR) is reported as 10/20/40Gbps, that’s actually the signaling rate: 8/16/32Gbps data rate
  – IB EDR and later standards fixed this “error” and started reporting the data rate (IB EDR reported as 100Gbps is truly data rate: 103.125Gbps signaling rate)
  – All through the tutorial, we only specify the data rate for all networks
Tackling Communication Bottlenecks with IB and HSE

• Network speed bottlenecks

• Protocol processing bottlenecks

• I/O interface bottlenecks
Capabilities of High-Performance Networks

- Intelligent Network Interface Cards
- Support entire protocol processing completely in hardware (hardware protocol offload engines)
- Provide a rich communication interface to applications
  - User-level communication capability
  - Gets rid of intermediate data buffering requirements
- No software signaling between communication layers
  - All layers are implemented on a dedicated hardware unit, and not on a shared host CPU
Previous High-Performance Network Stacks

- **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

- **Virtual Interface Architecture (VIA)**
  - Standardized by Intel, Compaq, Microsoft
  - Precursor to IB
IB Hardware Acceleration

• Some IB models have multiple hardware accelerators
  – E.g., Mellanox IB adapters

• Protocol Offload Engines
  – Completely implement ISO/OSI layers 2-4 (link layer, network layer
    and transport layer) in hardware

• Additional hardware supported features also present
  – RDMA, Multicast, QoS, Fault Tolerance, and many more
Ethernet Hardware Acceleration

• Interrupt Coalescing
  – Improves throughput, but degrades latency

• Jumbo Frames
  – No latency impact; Incompatible with existing switches

• Hardware Checksum Engines
  – Checksum performed in hardware → significantly faster
  – Shown to have minimal benefit independently

• Segmentation Offload Engines (a.k.a. Virtual MTU)
  – Host processor “thinks” that the adapter supports large Jumbo frames, but the adapter splits it into regular sized (1500-byte) frames
  – Supported by most HSE products because of its backward compatibility → considered “regular” Ethernet
  – Heavily used in the “server-on-steroids” model
    • High performance servers connected to regular clients
TOE and iWARP Accelerators

• TCP Offload Engines (TOE)
  – Hardware Acceleration for the entire TCP/IP stack
  – Initially patented by Tehuti Networks
  – Actually refers to the IC on the network adapter that implements TCP/IP
  – In practice, usually referred to as the entire network adapter

• Internet Wide-Area RDMA Protocol (iWARP)
  – Standardized by IETF and the RDMA Consortium
  – Support acceleration features (like IB) for Ethernet

Converged (Enhanced) Ethernet (CEE or CE)

• Also known as “Datacenter Ethernet” or “Lossless Ethernet”
  – Combines a number of optional Ethernet standards into one umbrella as mandatory requirements

• Sample enhancements include:
  – Priority-based flow-control: Link-level flow control for each Class of Service (CoS)
  – Enhanced Transmission Selection (ETS): Bandwidth assignment to each CoS
  – Datacenter Bridging Exchange Protocols (DBX): Congestion notification, Priority classes
  – End-to-end Congestion notification: Per flow congestion control to supplement per link flow control
Tackling Communication Bottlenecks with IB and HSE

- Network speed bottlenecks
- Protocol processing bottlenecks
- I/O interface bottlenecks
Interplay with I/O Technologies

• InfiniBand initially intended to replace I/O bus technologies with networking-like technology
  – That is, bit serial differential signaling
  – With enhancements in I/O technologies that use a similar architecture (HyperTransport, PCI Express), this has become mostly irrelevant now

• Both IB and HSE today come as network adapters that plug into existing I/O technologies
Trends in I/O Interfaces with Servers

- Recent trends in I/O interfaces show that they are nearly matching head-to-head with network speeds (though they still lag a little bit)

<table>
<thead>
<tr>
<th></th>
<th>Year</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI</td>
<td>1990</td>
<td>33MHz/32bit: 1.05Gbps (shared bidirectional)</td>
</tr>
<tr>
<td>PCI-X</td>
<td>1998 (v1.0), 2003 (v2.0)</td>
<td>133MHz/64bit: 8.5Gbps (shared bidirectional)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>266-533MHz/64bit: 17Gbps (shared bidirectional)</td>
</tr>
<tr>
<td>AMD HyperTransport (HT)</td>
<td>2001 (v1.0), 2004 (v2.0), 2006 (v3.0), 2008 (v3.1)</td>
<td>102.4Gbps (v1.0), 179.2Gbps (v2.0), 332.8Gbps (v3.0), 409.6Gbps (v3.1) (32 lanes)</td>
</tr>
<tr>
<td>PCI-Express (PCIe) by Intel</td>
<td>2003 (Gen1), 2007 (Gen2), 2009 (Gen3 standard)</td>
<td>Gen1: 4X (8Gbps), 8X (16Gbps), 16X (32Gbps), Gen2: 4X (16Gbps), 8X (32Gbps), 16X (64Gbps), Gen3: 4X (~32Gbps), 8X (~64Gbps), 16X (~128Gbps)</td>
</tr>
<tr>
<td>Intel QuickPath Interconnect (QPI)</td>
<td>2009</td>
<td>153.6-204.8Gbps (20 lanes)</td>
</tr>
</tbody>
</table>
Presentation Overview

• Introduction

• Why InfiniBand and High-speed Ethernet?

• Overview of IB, HSE, their Convergence and Features

• IB and HSE HW/SW Products and Installations

• Sample Case Studies and Performance Numbers

• Conclusions and Final Q&A
IB, HSE and their Convergence

• InfiniBand
  – Architecture and Basic Hardware Components
  – Communication Model and Semantics
  – Novel Features
  – Subnet Management and Services

• High-speed Ethernet Family
  – Internet Wide Area RDMA Protocol (iWARP)
  – Alternate vendor-specific protocol stacks

• InfiniBand/Ethernet Convergence Technologies
  – Virtual Protocol Interconnect (VPI)
  – InfiniBand over Ethernet (IBoE)
  – RDMA over Converged Enhanced Ethernet (RoCE)

• Interfacing with GPUs
Comparing InfiniBand with Traditional Networking Stack

**Application Layer**
- HTTP, FTP, MPI, File Systems
- MPI, PGAS, File Systems

**Transport Layer**
- TCP, UDP
- OpenFabrics Verbs
- RC (reliable), UD (unreliable)

**Network Layer**
- Routing
- OpenSM (management tool)
- Flow-control and Error Detection
- Copper, Optical or Wireless

**Link Layer**
- Flow-control and Error Detection
- Copper or Optical

**Physical Layer**
- Copper, Optical or Wireless

Traditional Ethernet

InfiniBand
IB Overview

• InfiniBand
  – Architecture and Basic Hardware Components
  – Communication Model and Semantics
    • Communication Model
    • Memory registration and protection
    • Channel and memory semantics
  – Novel Features
    • Hardware Protocol Offload
      – Link, network and transport layer features
  – Subnet Management and Services
Components: Channel Adapters

- Used by processing and I/O units to connect to fabric
- Consume & generate IB packets
- Programmable DMA engines with protection features
- May have multiple ports
  - Independent buffering channeled through Virtual Lanes
- Host Channel Adapters (HCAs)
Components: Switches and Routers

- Relay packets from a link to another
- Switches: intra-subnet
- Routers: inter-subnet
- May support multicast
Components: Links & Repeaters

• Network Links
  – Copper, Optical, Printed Circuit wiring on Back Plane
  – Not directly addressable

• Traditional adapters built for copper cabling
  – Restricted by cable length (signal integrity)
  – For example, QDR copper cables are restricted to 7m

• Intel Connects: Optical cables with Copper-to-optical conversion hubs (acquired by Emcore)
  – Up to 100m length
  – 550 picoseconds copper-to-optical conversion latency

• Available from other vendors (Luxtera)

• Repeaters (Vol. 2 of InfiniBand specification)
IB Overview

• InfiniBand
  – Architecture and Basic Hardware Components
  – Communication Model and Semantics
    • Communication Model
    • Memory registration and protection
    • Channel and memory semantics
  – Novel Features
    • Hardware Protocol Offload
      – Link, network and transport layer features
  – Subnet Management and Services
IB Communication Model

Basic InfiniBand Communication Semantics
Queue Pair Model

- Each QP has two queues
  - Send Queue (SQ)
  - Receive Queue (RQ)
  - Work requests are queued to the QP (WQEs: “Wookies”)
- QP to be linked to a Complete Queue (CQ)
  - Gives notification of operation completion from QPs
  - Completed WQEs are placed in the CQ with additional information (CQEs: “Cookies”)

Diagram: Queue Pair Model with InfiniBand Device
Memory Registration

Before we do any communication:
All memory used for communication must be registered

1. Registration Request
   - Send virtual address and length

2. Kernel handles virtual->physical mapping and pins region into physical memory
   - Process cannot map memory that it does not own (security !)

3. HCA caches the virtual to physical mapping and issues a handle
   - Includes an l_key and r_key

4. Handle is returned to application
Memory Protection

For security, keys are required for all operations that touch buffers

- To send or receive data the l_key must be provided to the HCA
  - HCA verifies access to local memory
- For RDMA, initiator must have the r_key for the remote virtual address
  - Possibly exchanged with a send/recv
  - r_key is not encrypted in IB

$l_key$ is needed for RDMA operations
Communication in the Channel Semantics
(Send/Receive Model)

Processor is involved only to:

1. Post receive WQE
2. Post send WQE
3. Pull out completed CQEs from the CQ

Send WQE contains information about the send buffer (multiple non-contiguous segments)

Receive WQE contains information on the receive buffer (multiple non-contiguous segments); Incoming messages have to be matched to a receive WQE to know where to place
Communication in the Memory Semantics (RDMA Model)

Initiator processor is involved only to:
1. Post send WQE
2. Pull out completed CQE from the send CQ

No involvement from the target processor

Send WQE contains information about the send buffer (multiple segments) and the receive buffer (single segment)
Communication in the Memory Semantics (Atomics)

Initiator processor is involved only to:

1. Post send WQE
2. Pull out completed CQE from the send CQ

No involvement from the target processor

Send WQE contains information about the send buffer (single 64-bit segment) and the receive buffer (single 64-bit segment)

IB supports compare-and-swap and fetch-and-add atomic operations
IB Overview

- **InfiniBand**
  - Architecture and Basic Hardware Components
  - Communication Model and Semantics
    - Communication Model
    - Memory registration and protection
    - Channel and memory semantics
  - Novel Features
    - Hardware Protocol Offload
      - Link, network and transport layer features
    - Subnet Management and Services
Hardware Protocol Offload

Complete Hardware Implementations Exist

% Hardware Implementations Exist

Consumer Transactions, Operations, etc. (IBA Operations)

Consumer

Product

Network Layer

Transport Layer

Link Layer

PHY Layer

Fabric

Channel Adapter

QP

Send

Rcv

WQE

CQE

Packet

Packet

Packet

Packet

IBA Operations (IBA Packets)

IBA Packets

Packet Relay

Channel Adapter

QP

Send

Rcv

WQE

CQE

Packet

Packet

Packet

Packet

IBA Operations (IBA Packets)

IBA Packets

Packet Relay
Link/Network Layer Capabilities

- CRC-based Data Integrity
- Buffering and Flow Control
- Congestion Control
- Virtual Lanes, Service Levels and QoS
- Switching and Multicast
- Network Fault Tolerance
- IB WAN Capability
CRC-based Data Integrity

- Two forms of CRC to achieve both early error detection and end-to-end reliability
  - Invariant CRC (ICRC) covers fields that do not change per link (per network hop)
    - E.g., routing headers (if there are no routers), transport headers, data payload
    - 32-bit CRC (compatible with Ethernet CRC)
    - End-to-end reliability (does not include I/O bus)
  - Variant CRC (VCRC) covers everything
    - 16-bit CRC
    - Erroneous packets do not have to reach the destination
    - Early error detection
Buffering and Flow Control

• IB provides three-levels of communication throttling/control mechanisms
  – Link-level flow control (link layer feature)
  – Message-level flow control (transport layer feature): discussed later
  – Congestion control (part of the link layer features)
• IB provides an absolute credit-based flow-control
  – Receiver guarantees that enough space is allotted for N blocks of data
  – Occasional update of available credits by the receiver
• Has no relation to the number of messages, but only to the total amount of data being sent
  – One 1MB message is equivalent to 1024 1KB messages (except for rounding off at message boundaries)
Congestion Control

• Why do we need congestion control even when flow control is available?

• Flow control does not know anything about who is using up the available bandwidth
  – Consider Node A sending data to Node B and Node C sending data to Node D: both communications might use a common intermediate link
  – If A is sending lot of data to B, the link-level flow control of the intermediate link will throttle all flows (including communication from C to D)
  – Idea of congestion control is to only throttle A → B communication and allow C → D communication to proceed
Congestion Control Working

• Switch detects congestion on a link
  – Detects whether it is the root or victim of congestion

• IB follows a three-step protocol
  – Forward Explicit Congestion Notification
    • Used to communicate congested port status
    • Switch sets FECN bit; marks packets leaving the congested state
  – Backward Explicit Congestion Notification
    • Destination sends BECN to sender informing about congestion
  – Injection Rate Control (Throttling)
    • Source throttles its send rate temporarily (timer based)
    • Original injection rate reduces over time
    • Congestion control may be performed per QP or SL

• Pro-active → does not wait for packet drops to occur
Link/Network Layer Capabilities

- CRC-based Data Integrity
- Buffering and Flow Control
- Congestion Control
- Virtual Lanes, Service Levels and QoS
- Switching and Multicast
- Network Fault Tolerance
- IB WAN Capability
Virtual Lanes

- Multiple virtual links within same physical link
  - Between 2 and 16

- Separate buffers and flow control
  - Avoids Head-of-Line Blocking

- VL15: reserved for management

- Each port supports one or more data VL
Service Levels and QoS

• Service Level (SL):
  – Packets may operate at one of 16 different SLs
  – Meaning not defined by IB

• SL to VL mapping:
  – SL determines which VL on the next link is to be used
  – Each port (switches, routers, end nodes) has a SL to VL mapping table configured by the subnet management

• Partitions:
  – Fabric administration (through Subnet Manager) may assign specific SLs to different partitions to isolate traffic flows
Traffic Segregation Benefits

- InfiniBand Virtual Lanes allow the multiplexing of multiple independent logical traffic flows on the same physical link.
- Providing the benefits of independent, separate networks while eliminating the cost and difficulties associated with maintaining two or more networks.

(Courtesy: Mellanox Technologies)
Switching (Layer-2 Routing) and Multicast

- Each port has one or more associated LIDs (Local Identifiers)
  - Switches look up which port to forward a packet to based on its destination LID (DLID)
  - This information is maintained at the switch

- For multicast packets, the switch needs to maintain multiple output ports to forward the packet to
  - Packet is replicated to each appropriate output port
  - Ensures at-most once delivery & loop-free forwarding
  - There is an interface for a group management protocol
    - Create, join/leave, prune, delete group
Switch Complex

- Basic unit of switching is a crossbar
  - Current InfiniBand products use either 24-port (DDR) or 36-port (QDR) crossbars

- Switches available in the market are typically collections of crossbars within a single cabinet

- Do not confuse “non-blocking switches” with “crossbars”
  - Crossbars provide all-to-all connectivity to all connected nodes
    - For any random node pair selection, all communication is non-blocking
  - Non-blocking switches provide a fat-tree of many crossbars
    - For any random node pair selection, there exists a switch configuration such that communication is non-blocking
    - If the communication pattern changes, the same switch configuration might no longer provide fully non-blocking communication
IB Switching/Routing: An Example

Switching: IB supports Virtual Cut Through (VCT)

Routing: Unspecified by IB SPEC
Up*/Down*, Shift are popular routing engines supported by OFED

- Fat-Tree is a popular topology for IB Cluster
  - Different over-subscription ratio may be used
- Other topologies are also being used
  - 3D Torus (Sandia Red Sky) and SGI Altix (Hypercube)

- Someone has to setup the forwarding tables and give every port an LID
  - “Subnet Manager” does this work
- Different routing algorithms give different paths

**An Example IB Switch Block Diagram (Mellanox 144-Port)**

- **Spine Blocks**
- **Leaf Blocks**

<table>
<thead>
<tr>
<th>DLID</th>
<th>Out-Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

LID: 2
LID: 4

P1
P2
More on Multipathing

• Similar to basic switching, except...
  – ... sender can utilize multiple LIDs associated to the same destination port
    • Packets sent to one DLID take a fixed path
    • Different packets can be sent using different DLIDs
    • Each DLID can have a different path (switch can be configured differently for each DLID)

• Can cause out-of-order arrival of packets
  – IB uses a simplistic approach:
    • If packets in one connection arrive out-of-order, they are dropped
  – Easier to use different DLIDs for different connections
    • This is what most high-level libraries using IB do!
IB Multicast Example

Switch decodes inbound packet header (LRH) DLID to determine target output ports.

Router decodes inbound packet header (GRH) GID multicast address to determine target output ports.
Network Level Fault Tolerance: Automatic Path Migration

- Automatically utilizes multipathing for network fault-tolerance (optional feature)
- Idea is that the high-level library (or application) using IB will have one primary path, and one fall-back path
  - Enables migrating connections to a different path
    - Connection recovery in the case of failures
- Available for RC, UC, and RD
- Reliability guarantees for service type maintained during migration
- Issue is that there is only one fall-back path (in hardware). If there is more than one failure (or a failure that affects both paths), the application will have to handle this in software
IB WAN Capability

• Getting increased attention for:
  – Remote Storage, Remote Visualization
  – Cluster Aggregation (Cluster-of-clusters)

• IB-Optical switches by multiple vendors
  – Obsidian Research Corporation: www.obsidianresearch.com
  – Bay Microsystems: www.baymicrosystems.com
  – Layer-1 changes from copper to optical; everything else stays the same
    • Low-latency copper-optical-copper conversion

• Large link-level buffers for flow-control
  – Data messages do not have to wait for round-trip hops
  – Important in the wide-area network
Hardware Protocol Offload

Complete Hardware Implementations Exist

Consumer Transactions, Operations, etc. (IBA Operations)
### IB Transport Services

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Connection Oriented</th>
<th>Acknowledged</th>
<th>Transport</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reliable Connection</td>
<td>Yes</td>
<td>Yes</td>
<td>IBA</td>
</tr>
<tr>
<td>Unreliable Connection</td>
<td>Yes</td>
<td>No</td>
<td>IBA</td>
</tr>
<tr>
<td>Reliable Datagram</td>
<td>No</td>
<td>Yes</td>
<td>IBA</td>
</tr>
<tr>
<td>Unreliable Datagram</td>
<td>No</td>
<td>No</td>
<td>IBA</td>
</tr>
<tr>
<td>RAW Datagram</td>
<td>No</td>
<td>No</td>
<td>Raw</td>
</tr>
</tbody>
</table>

- Each transport service can have zero or more QPs associated with it
  - E.g., you can have four QPs based on RC and one QP based on UD
# Trade-offs in Different Transport Types

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Reliable Connection</th>
<th>Reliable Datagram</th>
<th>Unreliable Datagram</th>
<th>Unreliable Connection</th>
<th>Raw Datagram (both IPv6 &amp; ethtype)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalability (M processes on N Processor nodes communicating with all processes on all nodes)</td>
<td>M^2N QPs required on each processor node, per CA</td>
<td>M QPs required on each processor node, per CA</td>
<td>M QPs required on each processor node, per CA</td>
<td>M^2N QPs required on each processor node, per CA</td>
<td>1 QP required on each end node, per CA</td>
</tr>
<tr>
<td>Corrupt data detected</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data delivery guarantee</td>
<td>Data delivered exactly once</td>
<td></td>
<td>No guarantees</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data order guaranteed</td>
<td>Yes, per connection</td>
<td>Yes, packets from any one source QP are ordered to multiple destination QPs</td>
<td>No</td>
<td>Unordered and duplicate packets are detected.</td>
<td>No</td>
</tr>
<tr>
<td>Data loss detected</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Error recovery</td>
<td>Reliable. Errors are detected at both the requestor and the responder. The requestor can transparently recover from errors (retransmission, alternate path, etc.) without any involvement of the client application. QP processing is halted only if the destination is inoperable or all fabric paths between the channel adapters have failed.</td>
<td>Unreliable. Packets with some types of errors may not be delivered. Neither source nor destination QPs are informed of dropped packets.</td>
<td>Unreliable. Packets with errors, including sequence errors, are detected and may be logged by the responder. The requestor is not informed.</td>
<td>Unreliable. Packets with errors are not delivered. The requestor and responder are not informed of dropped packets.</td>
<td></td>
</tr>
</tbody>
</table>
Transport Layer Capabilities

• Data Segmentation
• Transaction Ordering
• Message-level Flow Control
• Static Rate Control and Auto-negotiation
Data Segmentation

- IB transport layer provides a message-level communication granularity, not byte-level (unlike TCP)
- Application can hand over a large message
  - Network adapter segments it to MTU sized packets
  - Single notification when the entire message is transmitted or received (not per packet)
- Reduced host overhead to send/receive messages
  - Depends on the number of messages, not the number of bytes
Transaction Ordering

• IB follows a strong transaction ordering for RC
• Sender network adapter transmits messages in the order in which WQEs were posted
• Each QP utilizes a single LID
  – All WQEs posted on same QP take the same path
  – All packets are received by the receiver in the same order
  – All receive WQEs are completed in the order in which they were posted
Message-level Flow-Control

• Also called as End-to-end Flow-control
  – Does not depend on the number of network hops

• Separate from Link-level Flow-Control
  – Link-level flow-control only relies on the number of bytes being transmitted, not the number of messages
  – Message-level flow-control only relies on the number of messages transferred, not the number of bytes

• If 5 receive WQEs are posted, the sender can send 5 messages (can post 5 send WQEs)
  – If the sent messages are larger than what the receive buffers are posted, flow-control cannot handle it
Static Rate Control and Auto-Negotiation

• IB allows link rates to be statically changed
  – On a 4X link, we can set data to be sent at 1X
  – For heterogeneous links, rate can be set to the lowest link rate
  – Useful for low-priority traffic

• Auto-negotiation also available
  – E.g., if you connect a 4X adapter to a 1X switch, data is automatically sent at 1X rate

• Only fixed settings available
  – Cannot set rate requirement to 3.16 Gbps, for example
IB Overview

• InfiniBand
  – Architecture and Basic Hardware Components
  – Communication Model and Semantics
    • Communication Model
    • Memory registration and protection
    • Channel and memory semantics
  – Novel Features
    • Hardware Protocol Offload
      – Link, network and transport layer features
  – Subnet Management and Services
Concepts in IB Management

- **Agents**
  - Processes or hardware units running on each adapter, switch, router (everything on the network)
  - Provide capability to query and set parameters

- **Managers**
  - Make high-level decisions and implement it on the network fabric using the agents

- **Messaging schemes**
  - Used for interactions between the manager and agents (or between agents)

- **Messages**
Subnet Manager
IB, HSE and their Convergence

- InfiniBand
  - Architecture and Basic Hardware Components
  - Communication Model and Semantics
  - Novel Features
  - Subnet Management and Services
- High-speed Ethernet Family
  - Internet Wide Area RDMA Protocol (iWARP)
  - Alternate vendor-specific protocol stacks
- InfiniBand/Ethernet Convergence Technologies
  - Virtual Protocol Interconnect (VPI)
  - InfiniBand over Ethernet (IBoE)
  - RDMA over Converged Enhanced Ethernet (RoCE)
- Interfacing with GPUs
HSE Overview

• High-speed Ethernet Family
  – Internet Wide-Area RDMA Protocol (iWARP)
    • Architecture and Components
    • Features
      – Out-of-order data placement
      – Dynamic and Fine-grained Data Rate control
      – Multipathing using VLANs
      – Link Aggregation
      – Connection Management
    • Existing Implementations of HSE/iWARP
  – Alternate Vendor-specific Stacks
    • MX over Ethernet (for Myricom 10GE adapters)
    • Datagram Bypass Layer (for Myricom 10GE adapters)
    • Solarflare OpenOnload (for Solarflare 10GE adapters)
## IB and HSE RDMA Models: Commonalities and Differences

<table>
<thead>
<tr>
<th>Feature</th>
<th>IB</th>
<th>iWARP/HSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Acceleration</td>
<td>Supported</td>
<td>Supported (for TOE and iWARP)</td>
</tr>
<tr>
<td>RDMA</td>
<td>Supported</td>
<td>Supported (for iWARP)</td>
</tr>
<tr>
<td>Atomic Operations</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td>Multicast</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>Data Placement</td>
<td>Ordered</td>
<td>Out-of-order (for iWARP)</td>
</tr>
<tr>
<td>Data Rate-control</td>
<td>Static and Coarse-grained</td>
<td>Dynamic and Fine-grained (for TOE and iWARP)</td>
</tr>
<tr>
<td>QoS</td>
<td>Prioritization</td>
<td>Prioritization and Fixed Bandwidth QoS</td>
</tr>
</tbody>
</table>
iWARP Architecture and Components

- **RDMA Protocol (RDMAP)**
  - Feature-rich interface
  - Security Management

- **Remote Direct Data Placement (RDDP)**
  - Data Placement and Delivery
  - Multi Stream Semantics
  - Connection Management

- **Marker PDU Aligned (MPA)**
  - Middle Box Fragmentation
  - Data Integrity (CRC)

(Courtesy iWARP Specification)
HSE Overview

• High-speed Ethernet Family
  – Internet Wide-Area RDMA Protocol (iWARP)
    • Architecture and Components
    • Features
      – Out-of-order data placement
      – Dynamic and Fine-grained Data Rate control
      – Multipathing using VLANs
      – Link Aggregation
      – Connection Management
    • Existing Implementations of HSE/iWARP
  – Alternate Vendor-specific Stacks
    • MX over Ethernet (for Myricom 10GE adapters)
    • Datagram Bypass Layer (for Myricom 10GE adapters)
    • Solarflare OpenOnload (for Solarflare 10GE adapters)
Decoupled Data Placement and Data Delivery

• Place data as it arrives, whether in or out-of-order
• If data is out-of-order, place it at the appropriate offset
• Issues from the application’s perspective:
  – Second half of the message has been placed does not mean that the first half of the message has arrived as well
  – If one message has been placed, it does not mean that that the previous messages have been placed
• Issues from protocol stack’s perspective
  – The receiver network stack has to understand each frame of data
    • If the frame is unchanged during transmission, this is easy!
  – The MPA protocol layer adds appropriate information at regular intervals to allow the receiver to identify fragmented frames
Dynamic and Fine-grained Rate Control

• Part of the Ethernet standard, not iWARP
  – Network vendors use a separate interface to support it

• Dynamic bandwidth allocation to flows based on interval between two packets in a flow
  – E.g., one stall for every packet sent on a 10 Gbps network refers to a bandwidth allocation of 5 Gbps
  – Complicated because of TCP windowing behavior

• Important for high-latency/high-bandwidth networks
  – Large windows exposed on the receiver side
  – Receiver overflow controlled through rate control
Prioritization and Fixed Bandwidth QoS

• Can allow for simple prioritization:
  – E.g., connection 1 performs better than connection 2
  – 8 classes provided (a connection can be in any class)
    • Similar to SLs in InfiniBand
  – Two priority classes for high-priority traffic
    • E.g., management traffic or your favorite application

• Or can allow for specific bandwidth requests:
  – E.g., can request for 3.62 Gbps bandwidth
  – Packet pacing and stalls used to achieve this

• Query functionality to find out “remaining bandwidth”
HSE Overview

- **High-speed Ethernet Family**
  - Internet Wide-Area RDMA Protocol (iWARP)
    - Architecture and Components
    - Features
      - Out-of-order data placement
      - Dynamic and Fine-grained Data Rate control
      - Multipathing using VLANs
      - Link Aggregation
      - Connection Management
    - Existing Implementations of HSE/iWARP
  - Alternate Vendor-specific Stacks
    - MX over Ethernet (for Myricom 10GE adapters)
    - Datagram Bypass Layer (for Myricom 10GE adapters)
    - Solarflare OpenOnload (for Solarflare 10GE adapters)
VLAN based Multipathing

• Ethernet basic switching
  – Network is broken down to a tree by disabling links
  – Pros: No live-locks and simple switching
  – Cons: Single path between nodes and wastage of links

• VLAN based multipathing
  – Overlay many logical networks on one physical network
    • Each overlay network will break down into a unique tree
    • Depending on which overlay network you send on, you get a different path
    • Adding nodes/links is simple; you just add a new overlay
    • Older overlays will continue to work as earlier
Example VLAN Configuration

• Basic Ethernet converts the topology to a tree
  – Wastes four of the links

• Can be considered as two different VLANs
  – All the links in the network are utilized

• Can be used for:
  – High Performance
  – Security (if someone has to get access only to a part of the network)
  – Fault tolerance

• Supported by several switch vendors
  • Woven Systems, Cisco
Link Aggregation

• Link aggregation allows for multiple links to logically look like a single faster link
  − Done at a hardware level
  − Several multi-port network adapters allow for packets sequencing to avoid out-of-order packets
Connection Management in iWARP

- The iWARP standard is defined on a reliable IP-based protocol
  - Primarily provides an InfiniBand RC transport like behavior
  - The standard also specifies SRQ semantics, but current implementations do not provide them (but is being worked on)
  - There is no InfiniBand UD like capability

- iWARP transport binding standards exist for TCP/IP and SCTP/IP
  - Both bindings require a connection between all communicating pairs
HSE Overview

• High-speed Ethernet Family
  – Internet Wide-Area RDMA Protocol (iWARP)
    • Architecture and Components
    • Features
      – Out-of-order data placement
      – Dynamic and Fine-grained Data Rate control
      – Multipathing using VLANs
      – Link Aggregation
      – Connection Management
    • Existing Implementations of HSE/iWARP
      – Alternate Vendor-specific Stacks
        • MX over Ethernet (for Myricom 10GE adapters)
        • Datagram Bypass Layer (for Myricom 10GE adapters)
        • Solarflare OpenOnload (for Solarflare 10GE adapters)
Current Usage of Ethernet

- Regular Ethernet
- TOE
- System Area Network or Cluster Environment

Wide Area Network

- Regular Ethernet Cluster
- iWARP Cluster
- Distributed Cluster Environment
Software iWARP based Compatibility

- Regular Ethernet adapters and TOEs are fully compatible
- Compatibility with iWARP required
- Software iWARP emulates the functionality of iWARP on the host
  - Fully compatible with hardware iWARP
  - Internally utilizes host TCP/IP stack
Different iWARP Implementations

OSU, OSC, IBM

- Application
  - User-level iWARP
- Sockets
- TCP
- IP
- Device Driver
- Network Adapter

OSU, ANL

- Application
  - High Performance Sockets
- Sockets
  - TCP
  - IP
  - Device Driver
- Software iWARP
- Offloaded TCP
- Offloaded IP
- Network Adapter

Chelsio, NetEffect (Intel)

- Application
  - High Performance Sockets
- Sockets
  - TCP
  - IP
  - Device Driver
- Offloaded iWARP
- Offloaded TCP
- Offloaded IP
- Network Adapter

Regular Ethernet Adapters
TCP Offload Engines
iWARP compliant Adapters
HSE Overview

• **High-speed Ethernet Family**
  - Internet Wide-Area RDMA Protocol (iWARP)
    • Architecture and Components
    • Features
      - Out-of-order data placement
      - Dynamic and Fine-grained Data Rate control
      - Multipathing using VLANs
      - Link Aggregation
      - Connection Management
    • Existing Implementations of HSE/iWARP
  - **Alternate Vendor-specific Stacks**
    • MX over Ethernet (for Myricom 10GE adapters)
    • Datagram Bypass Layer (for Myricom 10GE adapters)
    • Solarflare OpenOnload (for Solarflare 10GE adapters)
Myrinet Express (MX)

• Proprietary communication layer developed by Myricom for their Myrinet adapters
  – Third generation communication layer (after FM and GM)
  – Supports Myrinet-2000 and the newer Myri-10G adapters

• Low-level “MPI-like” messaging layer
  – Almost one-to-one match with MPI semantics (including connection-less model, implicit memory registration and tag matching)
  – Later versions added some more advanced communication methods such as RDMA to support other programming models such as ARMCI (low-level runtime for the Global Arrays PGAS library)

• Open-MX
  – New open-source implementation of the MX interface for non-Myrinet adapters from INRIA, France
Datagram Bypass Layer (DBL)

• Another proprietary communication layer developed by Myricom
  – Compatible with regular UDP sockets (embraces and extends)
  – Idea is to bypass the kernel stack and give UDP applications direct access to the network adapter
    • High performance and low-jitter

• Primary motivation: Financial market applications (e.g., stock market)
  – Applications prefer unreliable communication
  – Timeliness is more important than reliability

• This stack is covered by NDA; more details can be requested from Myricom
Solarflare Communications: OpenOnload Stack

- HPC Networking Stack provides many performance benefits, but has limitations for certain types of scenarios, especially where applications tend to fork(), exec() and need asynchronous advancement (per application)

- Solarflare approach:
  - Network hardware provides user-safe interface to route packets directly to apps based on flow information in headers
  - Protocol processing can happen in both kernel and user space
  - Protocol state shared between app and kernel using shared memory

*Courtesy Solarflare communications (www.openonload.org/openonload-google-talk.pdf)*
IB, HSE and their Convergence

• InfiniBand
  – Architecture and Basic Hardware Components
  – Communication Model and Semantics
  – Novel Features
  – Subnet Management and Services

• High-speed Ethernet Family
  – Internet Wide Area RDMA Protocol (iWARP)
  – Alternate vendor-specific protocol stacks

• InfiniBand/Ethernet Convergence Technologies
  – Virtual Protocol Interconnect (VPI)
  – InfiniBand over Ethernet (IBoE)
  – RDMA over Converged Enhanced Ethernet (RoCE)

• Interfacing with GPUs
Virtual Protocol Interconnect (VPI)

- Single network firmware to support both IB and Ethernet
- Autosensing of layer-2 protocol
  - Can be configured to automatically work with either IB or Ethernet networks
- Multi-port adapters can use one port on IB and another on Ethernet
- Multiple use modes:
  - Datacenters with IB inside the cluster and Ethernet outside
  - Clusters with IB network and Ethernet management
(InfiniBand) RDMA over Ethernet (IBoE or RDMAoE)

- Native convergence of IB network and transport layers with Ethernet link layer
- IB packets encapsulated in Ethernet frames
- IB network layer already uses IPv6 frames
- Pros:
  - Works natively in Ethernet environments (entire Ethernet management ecosystem is available)
  - Has all the benefits of IB verbs
- Cons:
  - Network bandwidth might be limited to Ethernet switches: 10GE switches available; 40GE yet to arrive; 32 Gbps IB available
  - Some IB native link-layer features are optional in (regular) Ethernet
- Approved by OFA board to be included into OFED
(InfiniBand) RDMA over Converged Enhanced Ethernet (RoCE)

- Very similar to IB over Ethernet
  - Often used interchangeably with IBoE
  - Can be used to explicitly specify link layer is Converged Enhanced Ethernet (CE)

- Pros:
  - Works natively in Ethernet environments (entire Ethernet management ecosystem is available)
  - Has all the benefits of IB verbs
  - CEE is very similar to the link layer of native IB, so there are no missing features

- Cons:
  - Network bandwidth might be limited to Ethernet switches: 10GE switches available; 40GE yet to arrive; 32 Gbps IB available
## IB and HSE: Feature Comparison

<table>
<thead>
<tr>
<th></th>
<th>IB</th>
<th>iWARP/HSE</th>
<th>RDMAoE</th>
<th>RoCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Acceleration</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>RDMA</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Atomic Operations</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Multicast</td>
<td>Optional</td>
<td>No</td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>Data Placement</td>
<td>Ordered</td>
<td>Out-of-order</td>
<td>Ordered</td>
<td>Ordered</td>
</tr>
<tr>
<td>Prioritization</td>
<td>Optional</td>
<td>Optional</td>
<td>Optional</td>
<td>Yes</td>
</tr>
<tr>
<td>Fixed BW QoS (ETS)</td>
<td>No</td>
<td>Optional</td>
<td>Optional</td>
<td>Yes</td>
</tr>
<tr>
<td>Ethernet Compatibility</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>TCP/IP Compatibility</td>
<td>Yes (using IPoIB)</td>
<td>Yes (using IPoIB)</td>
<td>Yes (using IPoIB)</td>
<td>Yes (using IPoIB)</td>
</tr>
</tbody>
</table>
IB, HSE and their Convergence

- InfiniBand
  - Architecture and Basic Hardware Components
  - Communication Model and Semantics
  - Novel Features
  - Subnet Management and Services
- High-speed Ethernet Family
  - Internet Wide Area RDMA Protocol (iWARP)
  - Alternate vendor-specific protocol stacks
- InfiniBand/Ethernet Convergence Technologies
  - Virtual Protocol Interconnect (VPI)
  - InfiniBand over Ethernet (IBoE)
  - RDMA over Converged Enhanced Ethernet (RoCE)
- Interfacing with GPUs
InfiniBand + GPU systems

- Many systems today want to use systems that have both GPUs and high-speed networks such as InfiniBand
- Problem: Lack of a common memory registration mechanism
  - Each device has to pin the host memory it will use
  - Many operating systems do not allow multiple devices to register the same memory pages
- Previous solution:
  - Use different buffer for each device and copy data
GPU-Direct

• Collaboration between Mellanox and NVIDIA to converge on one memory registration technique

• Both devices register a common host buffer
  – GPU copies data to this buffer, and the network adapter can directly read from this buffer (or vice-versa)

• *Note that GPU-Direct does not allow you to bypass host memory*
  – *In that sense it is not true “direct” communication between the network and GPU*
Presentation Overview

• Introduction

• Why InfiniBand and High-speed Ethernet?

• Overview of IB, HSE, their Convergence and Features

• **IB and HSE HW/SW Products and Installations**

• Sample Case Studies and Performance Numbers

• Conclusions and Final Q&A
IB Hardware Products

- Many IB vendors: Mellanox, Voltaire and 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)

- New ConnectX-2 adapter from Mellanox supports offload for collectives (Barrier, Broadcast, etc.)
Tyan Thunder S2935 Board

(Courtesy Tyan)

Similar boards from Supermicro with LOM features are also available
IB Hardware Products (contd.)

- Customized adapters to work with IB switches
  - Cray XD1 (formerly by Octigabay), Cray CX1
- Switches:
  - 4X SDR and DDR (8-288 ports); 12X SDR (small sizes)
  - 3456-port “Magnum” switch from SUN → used at TACC
    - 72-port “nano magnum”
  - 36-port Mellanox InfiniScale IV QDR switch silicon in 2008
    - Up to 648-port QDR switch by Mellanox and SUN
    - Some internal ports are 96 Gbps (12X QDR)
  - New IB switch silicon from Qlogic introduced at SC ’08
    - Up to 846-port QDR switch by Qlogic
- Switch Routers with Gateways
  - IB-to-FC; IB-to-IP
10G, 40G and 100G Ethernet Products

- 10GE adapters: Intel, Myricom, Mellanox (ConnectX)
- 10GE/iWARP adapters: Chelsio, NetEffect (now owned by Intel)
- 40GE adapters: Mellanox ConnectX2-EN 40G
- 10GE switches
  - Fulcrum Microsystems
    - Low latency switch based on 24-port silicon
    - FM4000 switch with IP routing, and TCP/UDP support
  - Fujitsu, Myricom (512 ports), Force10, Cisco, Arista (formerly Arastra)
- 40GE and 100GE switches
  - Nortel Networks
    - 10GE downlinks with 40GE and 100GE uplinks
  - Broadcom has announced 40GE switch in early 2010
Products Providing IB and HSE Convergence

- Mellanox ConnectX Adapter
- Supports IB and HSE convergence
- Ports can be configured to support IB or HSE
- Support for VPI and RoCE
  - 8 Gbps (SDR), 16Gbps (DDR) and 32Gbps (QDR) rates available for IB
  - 10GE rate available for RoCE
  - 40GE rate for RoCE is expected to be available in near future
Software Convergence with OpenFabrics

• Open source organization (formerly OpenIB)
  – www.openfabrics.org

• Incorporates both IB and iWARP in a unified manner
  – Support for Linux and Windows
  – Design of complete stack with `best of breed` components
    • Gen1
    • Gen2 (current focus)

• Users can download the entire stack and run
  – Latest release is OFED 1.5.1
  – OFED 1.5.2 and 1.6 are underway
OpenFabrics Stack with Unified Verbs Interface

User Level
- Mellanox (libmthca)
- QLogic (libipathverbs)
- IBM (libehca)
- Chelsio (libcxgb3)

Kernel Level
- Mellanox (ib_mthca)
- QLogic (ib_ipath)
- IBM (ib_ehca)
- Chelsio (ib_cxgb3)

Verbs Interface (libibverbs)
OpenFabrics on Convergent IB/HSE

- For IBoE and RoCE, the upper-level stacks remain completely unchanged
- Within the hardware:
  - Transport and network layers remain completely unchanged
  - Both IB and Ethernet (or CEE) link layers are supported on the network adapter
- Note: The OpenFabrics stack is not valid for the Ethernet path in VPI
  - That still uses sockets and TCP/IP
OpenFabrics Software Stack

**Application Level**
- Diag Tools
- Open SM
- IP Based App Access
- Sockets Based Access
- Various MPIs
- Block Storage Access
- Clustered DB Access
- Access to File Systems

**User APIs**
- User Level MAD API
- User Space
- SDP Lib
- InfiniBand
- OpenFabrics User Level
- Verbs / API
- iWARP
- R-NIC

**Upper Layer Protocol**
- IPoIB
- SDP
- SRP
- iSER
- RDS
- NFS-RDMA
- RPC
- Cluster File Sys

**Mid-Layer**
- Connection Manager Abstraction (CMA)
- Connection Manager

**Kernel Space**
- IPoIB
- SDP
- SRP
- iSER
- RDS
- NFS-RDMA
- RPC
- Cluster File Sys

**Provider**
- Hardware Specific Driver
- InfiniBand HCA

**Hardware**
- InfiniBand HCA
- iWARP R-NIC

**Key**
- Common
- InfiniBand
- Apps & Access Methods for using OF Stack

**Common**
- 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
- iSER iSCSI RDMA Protocol (Initiator)
- RDS Reliable Datagram Service
- UDAPL User Direct Access Programming Lib
- HCA Host Channel Adapter
- R-NIC RDMA NIC

**Upper Layer Protocol Common**
- InfiniBand
- Key Hardware Specific Driver
- iWARP R-NIC

**Mid-Layer Common**
- Connection Manager
- Key Hardware Specific Driver
- iWARP R-NIC

**Provider Common**
- Hardware Specific Driver
- InfiniBand HCA
- iWARP R-NIC

**Application Level Common**
- Diag Tools
- Open SM
- IP Based App Access
- Sockets Based Access
- Various MPIs
- Block Storage Access
- Clustered DB Access
- Access to File Systems

**User APIs Common**
- User Level MAD API
- User Space
- SDP Lib
- InfiniBand
- OpenFabrics User Level
- Verbs / API
- iWARP
- R-NIC

**Upper Layer Protocol Common**
- IPoIB
- SDP
- SRP
- iSER
- RDS
- NFS-RDMA
- RPC
- Cluster File Sys

**Mid-Layer Common**
- Connection Manager Abstraction (CMA)
- Connection Manager

**Kernel Space Common**
- IPoIB
- SDP
- SRP
- iSER
- RDS
- NFS-RDMA
- RPC
- Cluster File Sys

**Provider Common**
- Hardware Specific Driver
- InfiniBand HCA

**Hardware Common**
- InfiniBand HCA
- iWARP R-NIC

**OpenFabrics Alliance**

**SC '10**
InfiniBand in the Top500

Percentage share of InfiniBand is steadily increasing
InfiniBand in the Top500 (Jun. 2010)

Number of Systems

- Gigabit Ethernet: 41.4%
- InfiniBand: 48.4%
- Proprietary: 6%
- Myrinet: 0%
- Quadrics: 0%
- NUMAlink: 0%
- Fat Tree: 1%
- SP Switch: 0%
- Cray Interconnect: 2%

Performance

- Gigabit Ethernet: 49%
- InfiniBand: 42%
- Proprietary: 24%
- Myrinet: 23%
- Quadrics: 0%
- NUMAlink: 0%
- Fat Tree: 0%
- SP Switch: 0%
- Cray Interconnect: 2%
Large-scale InfiniBand Installations

- 207 IB Clusters (41.4%) in the Jun ‘10 Top500 list ([http://www.top500.org](http://www.top500.org))
- Installations in the Top 30 (15 systems):

<table>
<thead>
<tr>
<th>System Description</th>
<th>Location</th>
<th>Cores</th>
</tr>
</thead>
<tbody>
<tr>
<td>120,640 cores (Nebulae) in China</td>
<td>(2nd)</td>
<td>120,640</td>
</tr>
<tr>
<td>122,400 cores (RoadRunner) at LANL</td>
<td>(3rd)</td>
<td>122,400</td>
</tr>
<tr>
<td>81,920 cores (Pleiades) at NASA Ames</td>
<td>(6th)</td>
<td>81,920</td>
</tr>
<tr>
<td>71,680 cores (Tianhe-1) in China</td>
<td>(7th)</td>
<td>71,680</td>
</tr>
<tr>
<td>42,440 cores (Red Sky) at Sandia</td>
<td>(10th)</td>
<td>42,440</td>
</tr>
<tr>
<td>62,976 cores (Ranger) at TACC</td>
<td>(11th)</td>
<td>62,976</td>
</tr>
<tr>
<td>35,360 cores (Lomonosov) in Russia</td>
<td>(13th)</td>
<td>35,360</td>
</tr>
<tr>
<td>26,304 cores (Europa) in Germany</td>
<td>(14th)</td>
<td>26,304</td>
</tr>
<tr>
<td>26,304 cores (Europa) in Germany</td>
<td>(14th)</td>
<td>26,304</td>
</tr>
<tr>
<td>23,040 cores (Jade) at GENCI</td>
<td>(18th)</td>
<td>23,040</td>
</tr>
<tr>
<td>33,120 cores (Mole-8.5) in China</td>
<td>(19th)</td>
<td>33,120</td>
</tr>
<tr>
<td>17,072 cores at JAEA in Japan</td>
<td>(22nd)</td>
<td>17,072</td>
</tr>
<tr>
<td>30,720 cores (Dawning) at Shanghai</td>
<td>(19th)</td>
<td>30,720</td>
</tr>
<tr>
<td>24,704 cores by HP</td>
<td>(25th)</td>
<td>24,704</td>
</tr>
<tr>
<td>15,360 cores at ERDC</td>
<td>(30th)</td>
<td>15,360</td>
</tr>
</tbody>
</table>

More are getting installed!
HSE Scientific Computing Installations

• HSE compute systems
  – 5,600-core installation in Purdue with 10GE Chelsio/iWARP
    • #102 in June 2010 TOP500 list
  – 640-core installation in University of Heidelberg, Germany
  – 512-core installation at Sandia National Laboratory (SNL) with Chelsio/iWARP and Woven Systems switch
  – 256-core installation at Argonne National Lab with Myri-10G

• Integrated Systems
  – BG/P uses 10GE for I/O (ranks 4, 8, 11, and 18 in the Top 25)
Other HSE Installations

- HSE has most of its popularity in enterprise computing and other non-scientific markets including Wide-area networking.

- Example Enterprise Computing Domains
  - Enterprise Datacenters (HP, Intel)
  - Animation firms (e.g., Universal Studios (“The Hulk”), 20th Century Fox (“Avatar”), and many new movies using 10GE)
  - Amazon’s HPC cloud offering uses 10GE internally
  - Heavily used in financial markets (users are typically undisclosed)

- ESnet to install $62M 100GE infrastructure for US DOE
Presentation Overview

- Introduction
- Why InfiniBand and High-speed Ethernet?
- Overview of IB, HSE, their Convergence and Features
- IB and HSE HW/SW Products and Installations
- Sample Case Studies and Performance Numbers
- Conclusions and Final Q&A
Case Studies

- Low-level Performance
- Message Passing Interface (MPI)
Low-level Latency Measurements

**Small Messages**

- VPI-IB
- Native IB
- VPI-Eth
- RoCE

**Large Messages**

ConnectX-DDR: 2.4 GHz Quad-core (Nehalem) Intel with IB and 10GE switches

RoCE has a slight overhead compared to native IB because it operates at a slower clock rate (required to support only a 10Gbps link for Ethernet, as compared to a 32Gbps link for IB)
Low-level Uni-directional Bandwidth Measurements

ConnectX-DDR: 2.4 GHz Quad-core (Nehalem) Intel with IB and 10GE switches
Case Studies

- Low-level Performance
- Message Passing Interface (MPI)
MVAPICH/MVAPICH2 Software

- High Performance MPI Library for IB and HSE
  - MVAPICH (MPI-1) and MVAPICH2 (MPI-2.2)
  - Used by more than 1,185 organizations in 59 countries
  - More than 44,000 downloads from OSU site directly
  - Empowering many TOP500 clusters
    - 6th ranked 81,920-core cluster (Pleiades) at NASA
    - 7th ranked 71,680-core cluster (Tianhe-1) in China
    - 11th ranked 62,976-core cluster (Ranger) at TACC
  - Available with software stacks of many IB, HSE and server vendors including Open Fabrics Enterprise Distribution (OFED)
    - http://mvapich.cse.ohio-state.edu
MPICH2 Software Stack

• High-performance and Widely Portable MPI
  – Supports MPI-1, MPI-2, MPI-2.1 and MPI-2.2
  – Supports multiple networks (TCP, IB, iWARP, Myrinet)
  – Commercial support by many vendors
    • IBM (integrated stack distributed by Argonne)
    • Microsoft, Intel (in process of integrating their stack)
  – Used by many derivative implementations
    • E.g., MVAPICH2, IBM, Intel, Microsoft, Cray, Myricom
    • MPICH2 and its derivatives support many Top500 systems (estimated at more than 90%)
  – Available with many software distributions
  – Integrated with the ROMIO MPI-IO implementation and the MPE profiling library
One-way Latency: MPI over IB

Small Message Latency

Large Message Latency

All numbers taken on 2.4 GHz Quad-core (Nehalem) Intel with IB switch
Bandwidth: MPI over IB

Unidirectional Bandwidth

Bidirectional Bandwidth

All numbers taken on 2.4 GHz Quad-core (Nehalem) Intel with IB switch
One-way Latency: MPI over iWARP

The graph shows the one-way latency for different message sizes using Chelsio (TCP/IP), Chelsio (iWARP), Intel-NetEffect (TCP/IP), and Intel-NetEffect (iWARP). The latency is measured in microseconds (us) and the message size is in bytes.

Key features:
- **Chelsio (TCP/IP)**: Demonstrates a steady increase in latency with increasing message size.
- **Chelsio (iWARP)**: Shows a consistent performance across various message sizes.
- **Intel-NetEffect (TCP/IP)**: Exhibits a slight increase in latency with increasing message size.
- **Intel-NetEffect (iWARP)**: Displays a notable increase in latency as the message size exceeds 1K bytes.

The graph is based on a **2.4 GHz Quad-core Intel (Clovertown) with 10GE (Fulcrum) Switch** configuration.
Bandwidth: MPI over iWARP

2.33 GHz Quad-core Intel (Clovertown) with 10GE (Fulcrum) Switch
Convergent Technologies: MPI Latency

ConnectX-DDR: 2.4 GHz Quad-core (Nehalem) Intel with IB and 10GE switches
Convergent Technologies: MPI Uni- and Bi-directional Bandwidth

ConnectX-DDR: 2.4 GHz Quad-core (Nehalem) Intel with IB and 10GE switches
Presentation Overview

- Introduction
- Why InfiniBand and High-speed Ethernet?
- Overview of IB, HSE, their Convergence and Features
- IB and HSE HW/SW Products and Installations
- Sample Case Studies and Performance Numbers

- Conclusions and Final Q&A
Concluding Remarks

- Presented network architectures & trends in Clusters
- Presented background and details of IB and HSE
  - Highlighted the main features of IB and HSE and their convergence
  - Gave an overview of IB and HSE hardware/software products
  - Discussed sample performance numbers in designing various high-end systems with IB and HSE
- IB and HSE are emerging as new architectures leading to a new generation of networked computing systems, opening many research issues needing novel solutions
Funding Acknowledgments

Funding Support by

Equipment Support by
# Personnel Acknowledgments

## Current Students
- N. Dandapanthula (M.S.)
- J. Jose (Ph.D.)
- K. Kandalla (M.S.)
- P. Lai (Ph.D.)
- M. Luo (Ph.D.)
- S. Pai (M.S.)
- V. Meshram (M.S.)
- X. Ouyang (Ph.D.)
- S. Potluri (Ph.D.)
- R. Rajachandrasekhar (Ph.D.)
- H. Subramoni (Ph.D.)

## Past Students
- P. Balaji (Ph.D.)
- D. Buntinas (Ph.D.)
- S. Bhagvat (M.S.)
- L. Chai (Ph.D.)
- B. Chandrasekharan (M.S.)
- T. Gangadharappa (M.S.)
- K. Gopalakrishnan (M.S.)
- W. Huang (Ph.D.)
- W. Jiang (M.S.)
- S. Kini (M.S.)
- M. Koop (Ph.D.)
- R. Kumar (M.S.)
- S. Krishnamoorthy (M.S.)

- P. Lai (Ph.D.)
- J. Liu (Ph.D.)
- A. Mamidala (Ph.D.)
- G. Marsh (M.S.)
- S. Naravula (Ph.D.)
- R. Noronha (Ph.D.)
- G. Santhanaraman (Ph.D.)
- J. Sridhar (M.S.)
- S. Sur (Ph.D.)
- K. Vaidyanathan (Ph.D.)
- A. Vishnu (Ph.D.)
- J. Wu (Ph.D.)
- W. Yu (Ph.D.)

## Current Research Scientist
- S. Sur

## Current Post-Docs
- H. Wang
- J. Vienne

## Past Post-Docs
- E. Mancini
- S. Marcarelli
- H.-W. Jin

## Current Programmers
- M. Arnold
- J. Perkins
Web Pointers

http://www.cse.ohio-state.edu/~panda
http://www.cse.ohio-state.edu/~surs
http://www.mcs.anl.gov/~balaji
http://nowlab.cse.ohio-state.edu

MVAPICH Web Page

http://mvapich.cse.ohio-state.edu

panda@cse.ohio-state.edu
surs@cse.ohio-state.edu
balaji@mcs.anl.gov