HPC InfiniBand Requirements:

Lessons Learned from Five Years of Building InfiniBand Clusters

Matt Leininger, Steve Poole, Kim Yates

Sandia National Laboratories
Los Alamos National Laboratory
Lawrence Livermore National Laboratory

6 February 2006

DoE has tracked InfiniBand for several generations

2001-2002: Nitro I & II: IB blade reference designs (SNL) 2.2 GHz Xeon processors, small clusters, funded early MPI/IB work, and Cadillac (LANL) 128 node cluster

2003: Catalyst: 128 nodes 4X PCI-X IB (SNL), Blue Steel: 256 dual nodes 4X PCI-X (LANL), 96 nodes 4X PCI-X Viz Red RoSE (SNL)

2004: Catalyst: Added 85 nodes 4X PCIe IB, 288 port IB switch(SNL), ~300 nodes 4X PCIe Viz Red Rose (SNL)

2005: Thunderbird and Talon: 4,480 and 128 dual 3.6 Ghz nodes, 4X PCIe IB (SNL)

Lustre/IB production @ SNL Red RoSE

2006: 2,000 nodes PCIe IB (LANL), and more to come;
Estimate ~9k-10k nodes by end of year (SNL+LANL+LLNL)

Goals of InfiniBand Software PathForward

- To accelerate the development of an Linux IB software stack for HPC
  - High performance (high bandwidth, low latency, low CPU overhead)
  - Scalability
  - Robustness
  - Portability
  - Reliability
  - Manageability
  - Single open source SW stack and diagnostic tool set supported across multiple (i.e. all) system vendors
  - Integrate IB SW stack into mainline Linux kernel at kernel.org
  - Get stack into Linux distributions (RedHat, SuSE, etc.)

OpenIB was formed around these goals

DoE ASC PathForward program has been funding OpenIB development since early 2005

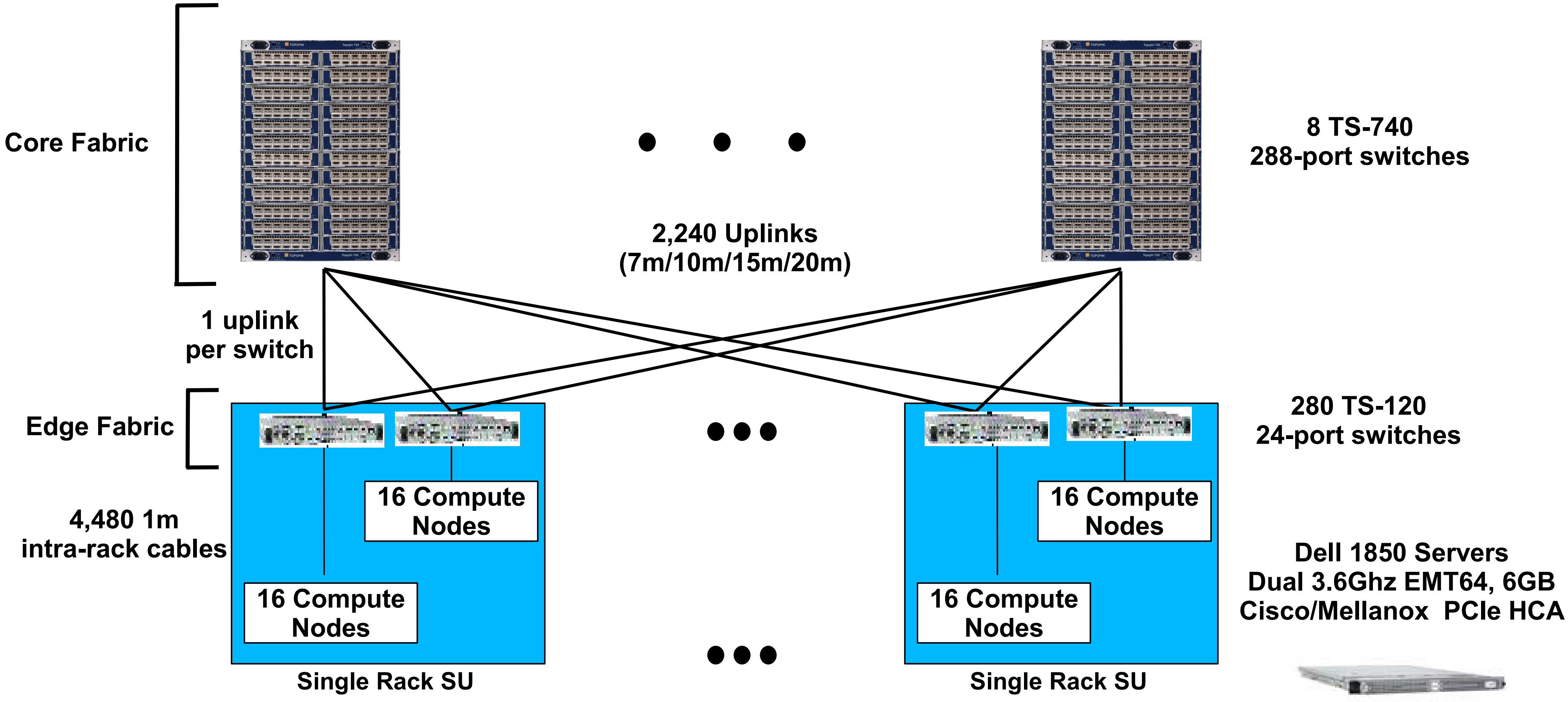
Experiences on 128 node Catalyst InfiniBand Cluster 2003

- Linpack 1.076 Tflops (1.567 theoretical)
- 111th on Top500 Nov. 2003
- 69% overall efficiency, 95% scalable
- Debug tools
  - Check all HCAs/nodes (vstat); All ok? then...
  - Run Linpack
  - If fails then run nnode/2 Linpack
  - Repeat until Linpack works
  - Continue "bi-section" debugging until bad cable/switch port found
  - Very painful for 128 node cluster, simple problems could take hours
  - LANL had to do this for 256 node Blue Steel
  - Better MPI debug information helped a little

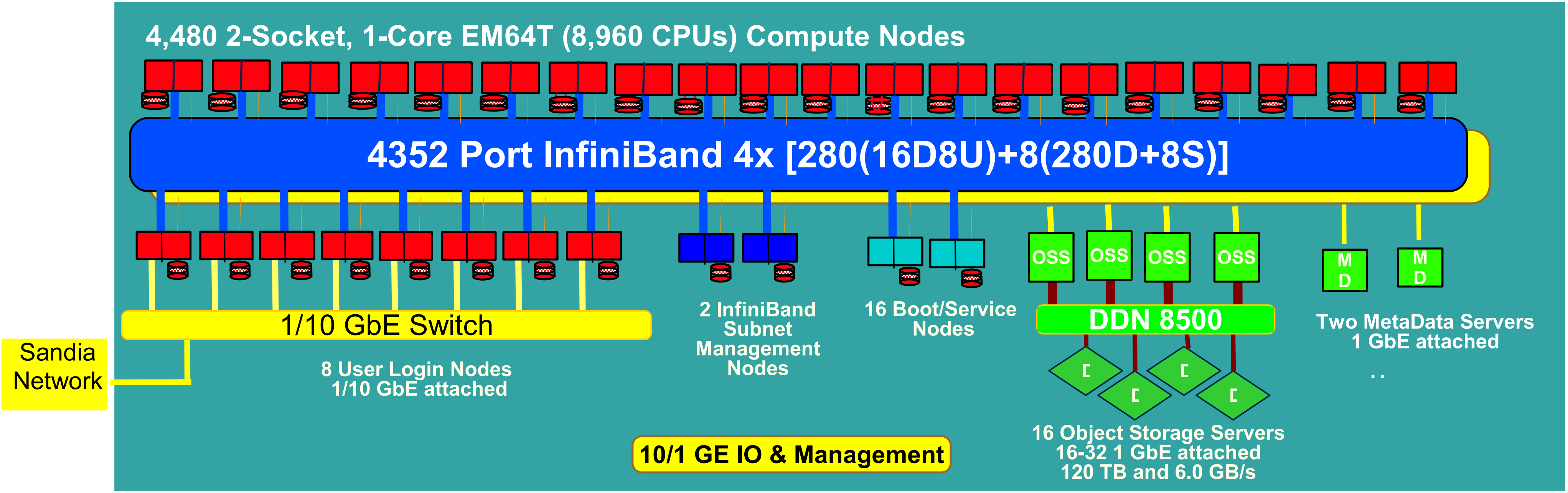
But that was 2003 .... What about today?

Sandia Thunderbird Cluster

8,960 Processor, 65TF/s



Thunderbird Architecture



System Parameters

- 14.4 GF/s dual socket 3.6 GHz single core Intel SMP nodes DDR-2 400 SDRAM
- 50% blocking (2:1 oversubscription of InfiniBand fabric)
- ~300 InfiniBand switches to manage
- ~9,000 InfiniBand ports
- ~33,600 meters (or 21 miles) of 4X InfiniBand copper cables
- ~10,000 meters (or 6 miles) of copper Ethernet cables
- 26,880 1 GB DDR-2 400 SDRAM modules
- 1.8 MW of power, 400 tons of cooling
- Up to 2000 nodes Linpack efficiency was ~82%

#5 in Top500
38.2 Tflops on 3721 nodes
71% efficiency

National Nuclear Security Administration

Thunderbird Software

- Currently using proprietary InfiniBand software stacks
  - Upgrading to OpenIB later this year
- Host-based SM can initialize ~4,000 nodes in 58 seconds
- Ability to monitor and track most errors very quickly
- Network congestion information is still difficult to extract from fabric
  - Congestion is a bottleneck to scalability
- MPI memory scalability remains an issue (currently testing Open MPI)
  - Reducing MPI resources/buffer lead to lock-ups and/or poor performance
- Other requirements for OpenIB are based on experiences on Thunderbird

Host Side Diagnostic and Management Tools

- HCA “burn-in” diagnostics (memtest, resource exhaustion, stress tests)
- Number of HCAs in node, state of IB drivers, number of network planes the node is attached to, speed of IB links, implemented services, route configurations, and performance
- Scalable network flash of HCA firmware
- Network performance and traffic counters at the node level
- Version info for drivers, HCA, and other services
  - Consistency/Compatibility checks
- Diagnostic and Management tools accessible via API and CLI

Fabric and Subnet Manager Requirements

- Ability to obtain network topology, congestion, and traffic information through CLI and API
- Sweep and fully configure fabric of 8,192 ports in less than 1 minute
- Pluggable modules for fabric route computations
- Support for fat-tree and 3D Mesh/Torus network topologies
- Fabric debug tools (ping, dump, trace, walkpath)
- Automated OS multi-vendor health monitoring of IB network
  - Monitor historical data on fabric for more subtle problems
- Open source tools to obtain all information from multi-vendor environments

MPI and OpenIB Verbs Requirements

- High performance (near line rate) and scalable to 1,000's of nodes
- Memory footprint scalability to 1,000's of nodes
- Latency through MPI and Verbs layer less than 1us end-to-end
- High performance UD, RC, RD, and RDMA
- Increased performance for small and medium sized messages
- Support for low latency interrupt mode
- QoS and multi-path support
- Support for multiple HCAs per node
- Use fabric topology data for performance enhancements
- Fast path to HCA and QP data for use in source-based adaptive routing
- Thread safety

HPC InfiniBand Requirements

- Improved UD performance and support for RD
- Improved BW/lat. for small-medium sized messages (critical for perf scalability)
- ~1 us latency (from user program mem on node A to user program mem on node B)
- Full support for congestion control architecture in HW
- Fix flow control in SRQ (Tim Woodall and Jeff Squyres)
- Reliable hardware multicast/broadcast
- Improve performance of and/or eliminate memory registration
- Support for queued DMA's
- MPI collectives or primitives in HW via collective offload engine (reduce,allreduce,reduce scatter, gather+scatter)

HPC InfiniBand Requirements

- Multi-path/dispersive routing (LMC>0 and MPI support)
- Fully adaptive routing
- IB to IB routers
- Low bit error rate (<< 10^{-15})
- 24X/36X QDR/ODR InfiniBand
- Affordable fiber options for 12X SDR/DDR/QDR (same cost as copper)
- Expand LID space and number of service levels (BG/L sized platforms 64k nodes, 12k nodes)
- HW support for data transfer ops. (MPI, UPC, Portals, CAF)

Booting Over InfiniBand

- Currently booting a Bproc cluster over IB ONLY
- No Ethernet needed
- How?
  - Use LinuxBIOS
  - Payload in flash that is full SMP 2.6.14
  - Does insmod of the appropriate modules, then ipconfig, then rarp
  - No scripts needed
  - Pulls down new kernel and does a kexec
  - Since the first kernel is full SMP you don't always need to exec a phase 2 so boot times can be REALLY fast
- Done by Ron Minnich (LANL) and Hal Rosenstock (Voltaire)

OpenIB 1.0 Feature Requirements

- Many features we need are already in OpenIB as part of DoE PF
- IB software and MPI must scale well to thousands of processors
- Full support for congestion control architecture
- Single diagnostics and management tool set that support multiple vendor hardware
- SM scalable to 1000's of nodes, config fabric in < 60 s
- Software testing/hardening and Q&A
  - We need automated regression testing framework
  - Set up multiple sites for automated nightly testing of OpenIB stack
- Booting over IB
- Support from OpenIB and vendor community for one version of OpenIB

Strengthen commitment to Open source collaboration

- Ubiquitous OpenIB stack (+ iWarp) will *expand market*
- *Ubiquity requires quality, stability, and support*
- *"Free" software is not cost-free*
  - *Put your highest-quality SW in OpenIB stack*
  - *Create a more robust development and collaborative infrastructure (rely on annual dues, ... ?)*
  - *Customers willing to pay good money for maintenance and support – need commercial support/maintenance services*
- *Multi-vendor OpenIB stacks won't fly*
  - *Companies need to support the same SW stack version and work as a community to support and harden the stack*
- *Vibrant multi-vendor ecosystem*

National Nuclear Security Administration

OpenIB community issues

- Improve and control the quality of the software stack
  - Performance
  - Compliance
  - Diagnostic tool set
  - Industrial-strength support for collaborative devel. and rigorous regression testing
- Gain momentum
  - Visibility, products in market, membership, active participation
- What should be in the OpenIB distributions?
  - How shall the community decide this?
  - How do we make it happen?
  - Must be resolved soon
  - We have a list ... need to take the next step
- OpenIB has successfully created a collaborative development environment
  - Now need to create a collaborative environment for Q&A and support

For more information

Matt Leininger mlleini@sandia.gov

Steve Poole spoole@lanl.gov

Kim Yates yates2@llnl.gov