SAND2005-7986P

Open Source Software FY06 Project Review

Matt Leininger

Scalable Computing R&D Group
Sandia National Laboratories
Livermore, CA

20 December 2005

National Nuclear Security Administration

Sandia National Laboratories

Open Source Software Project

Importance to the ASC program

- Maximizes productivity of ASC InfiniBand clusters by removing current performance and scalability bottlenecks (network congestion, MPI, etc.)
- Provides higher confidence in the performance and stability of the OpenIB software stack for ASC capacity and capability platforms
- Directly supports the requirements for the OpenIB and GFS (Lustre, NFSv4) projects
- Provides clusters dedicated to testing OpenIB SW stack

FY06 Budget Profile

~$800K FY06 2.5-3.0 FTE

Matt Leininger

Michael Lee

Ida Nielsen

Dave Evensky

Joe Kenny

~$50-75K travel, purchases, misc.

Project Overview:

- Support the development of open source application middleware and InfiniBand software stack development for HPC
- Focused on driving application performance enhancements and on improving productivity
- Project management of OpenIB "PathForward" Project
- Deliver congestion control architecture support in OpenIB
- Deliver optimized and scalable MPI collectives (allreduce,broadcast, and others)
- Deliver InfiniBand multi-path routing support for load balancing and network congestion control in MPI

Project Impact to ASC Program

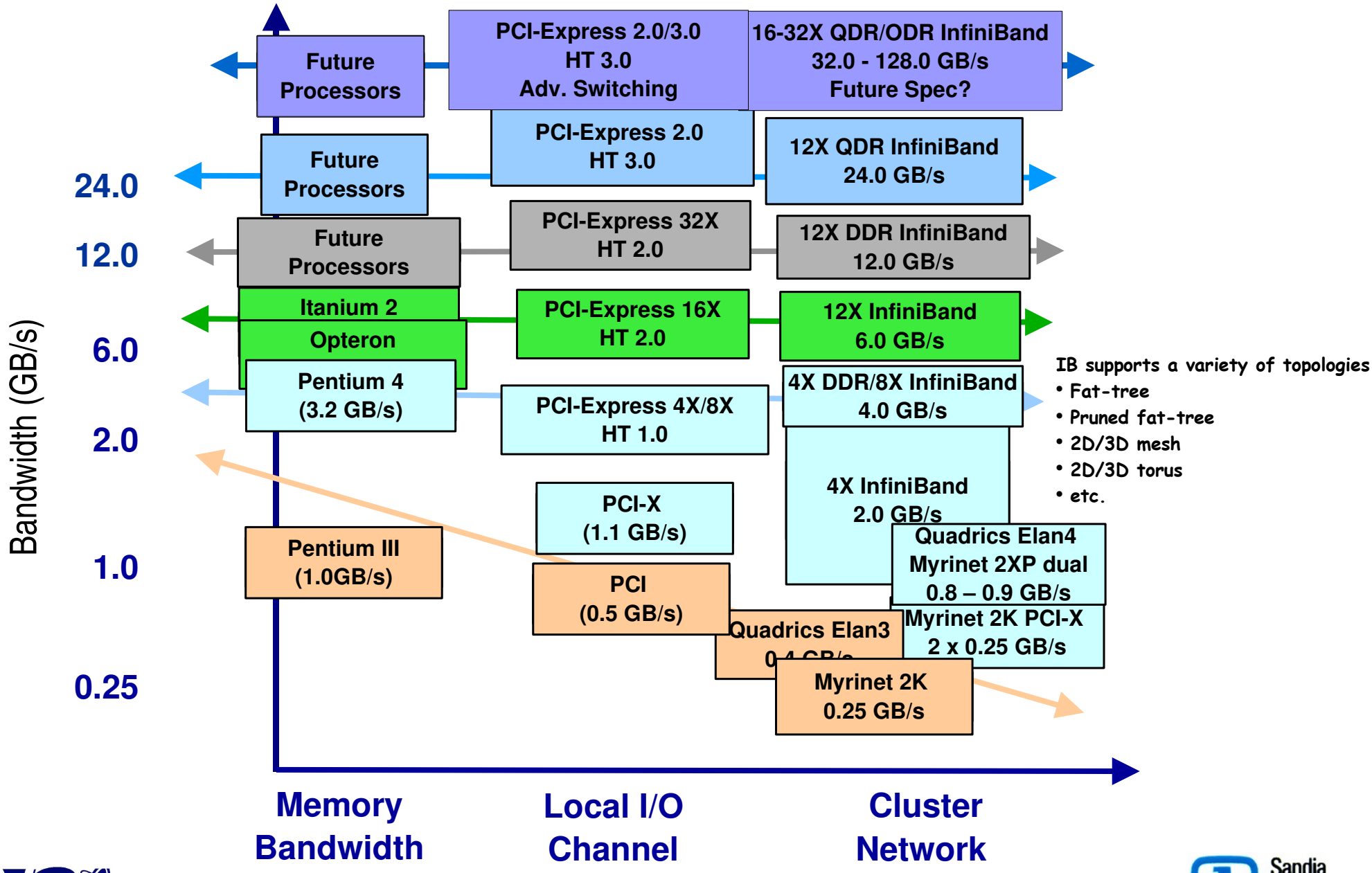
- Increases productivity of all ASC InfiniBand clusters
  - 6,500 nodes today; likely growing to 10,000+ in FY06
- Continue Sandia's leadership role in InfiniBand
  - Recognized by many outside DoE
  - JP Morgan Chase, Credit Suisse First Boston, NRL, Morgan Stanley, etc.
- Model for high funding leverage on open source open development projects with strong HPC alignment

FY06 Project Deliverables

- Network Congestion Control features for large-scale InfiniBand clusters
  - Critical to the productivity of InfiniBand clusters
- OpenIB PathForward project management, provide resources for milestone testing and code development
  - Critical for OpenIB project success; Tested quarterly
- Optimized MPI collectives (Allreduce/broadcast)
  - Critical for improving application productivity
- Source-based adaptive routing for InfiniBand in MPI
  - Critical for improving network utilization in applications
- Linux synchronized kernel scheduler
  - Useful for reducing operating system interference to improve collective performance

The Open Source Software Project is designed around maximizing the impact of InfiniBand and Linux to Sandia/DoE/ASC programs.

InfiniBand Roadmap Tracks Future Processor and I/O Performance



National Nuclear Security Administration

Sandia National Laboratories

United HPC and Industry Around IB by Forming OpenIB Alliance

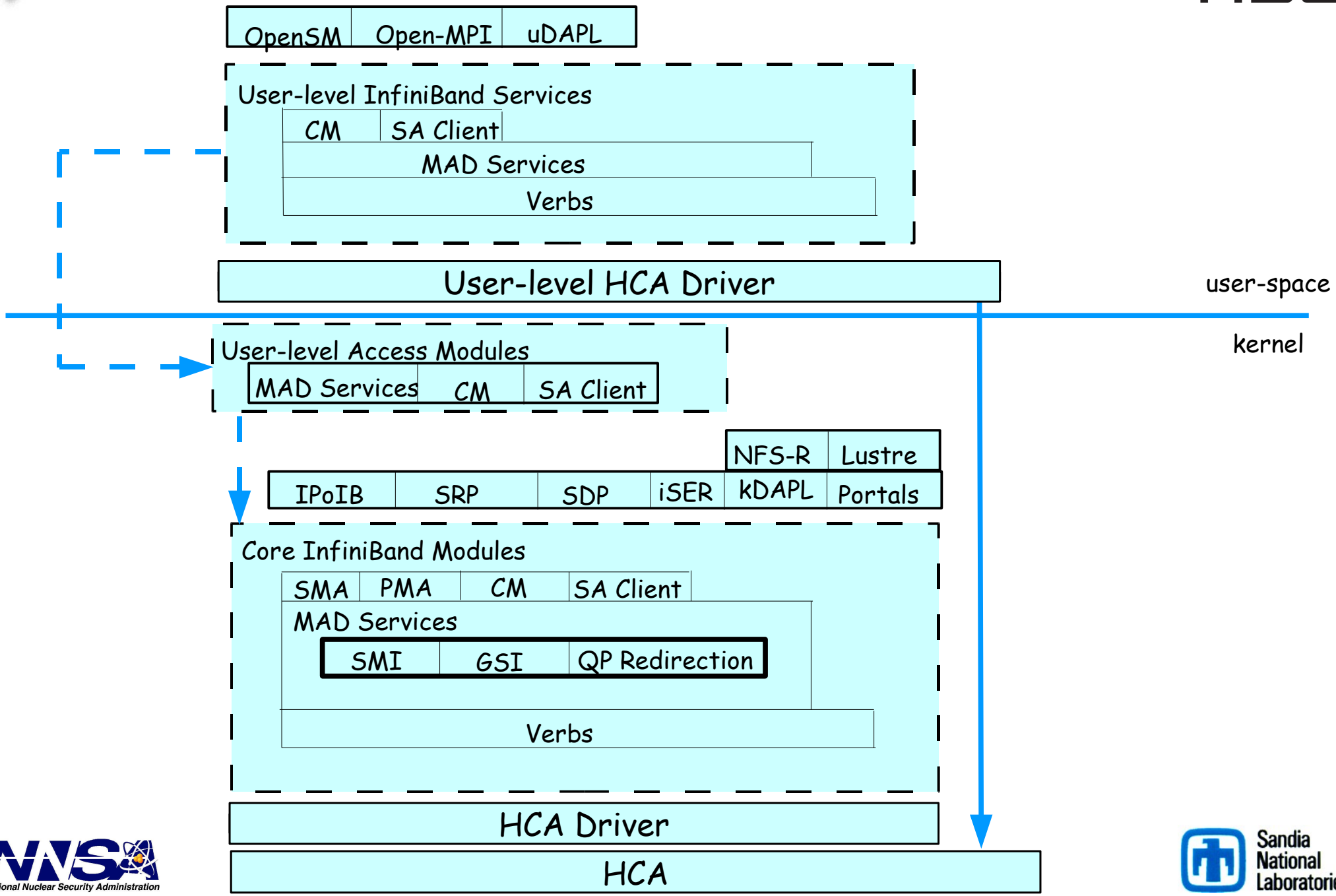
- To accelerate the development of an Linux IB "shrink-wrap" software stack for HPC
  - High performance (high bandwidth, low latency)
  - Scalability
  - Robustness and Portability
  - Reliability
  - Manageability
  - Single open source SW stack supported across multiple system vendors
  - Integrate IB SW stack into mainline Linux kernel at kernel.org
  - Supported by Linux distributions (RedHat, SuSE, etc.)
- Sandia leads DoE (ASC PF program) which funds part of the work on the OpenIB stack that is focused on HPC – Voltaire, Cisco, SilverStorm, and Intel

www.openib.org

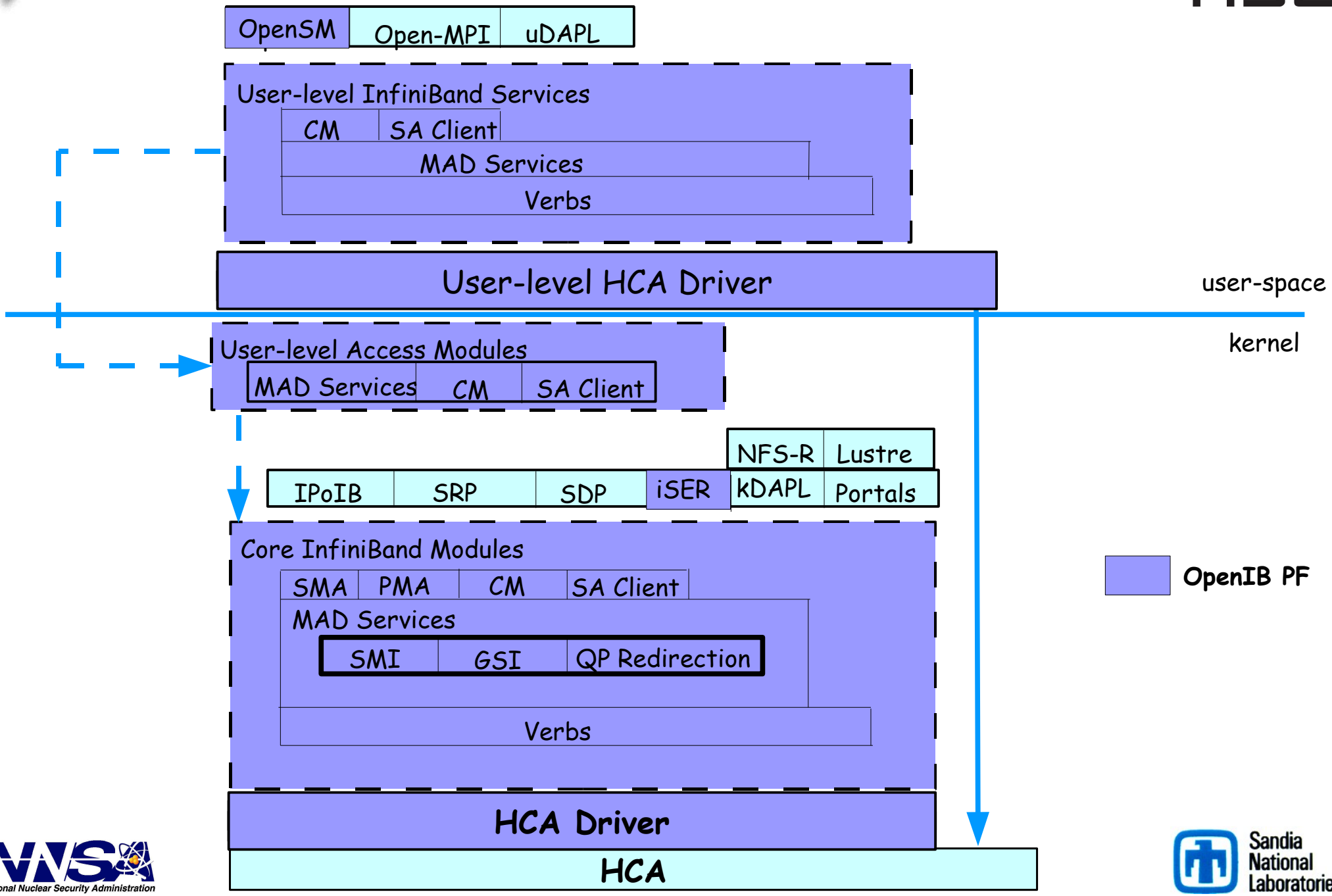
OpenIB Members

…and the open source community!

OpenIB Stack Architecture



OpenIB Stack Architecture



National Nuclear Security Administration

Sandia National Laboratories

OpenIB PathForward FY06 Milestones

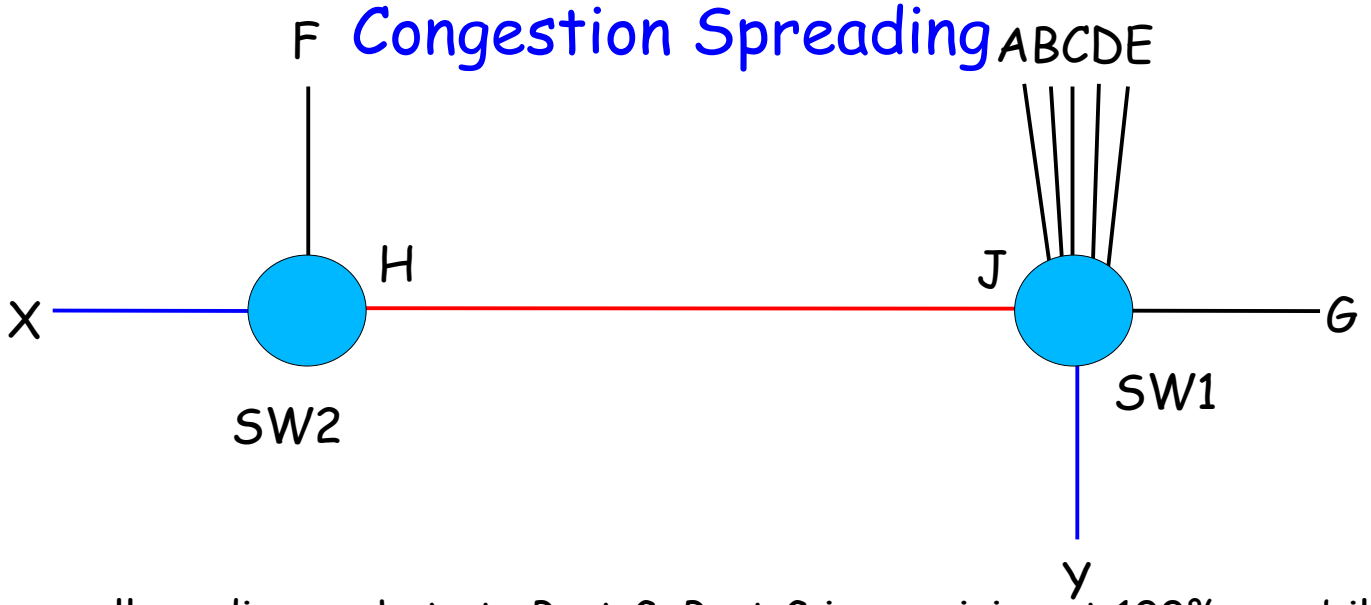
- Stage 2
  - Project Plan and Design Review
  - Alpha release of OpenIB host and network diagnostic and mgmt tools
  - Alpha release of OpenIB SW stack with HPC/PF capabilities
  - Stage 2 Release of OpenIB SW and diags with HPC/PF capabilities
- Stage 3
  - Project Plan and Design Review
  - Alpha release of OpenIB SW
  - Final release of OpenIB SW stack with HPC capabilities
- Open Source Software Project FY06 Deliverable: Manage OpenIB PF, provide resources for milestone testing, and code development

OpenIB testing on Catalyst/Talon is required for each SW milestone
Testing will use kernel.org and RedHat Linux kernels

Removing Critical HPC Application Scalability Bottlenecks

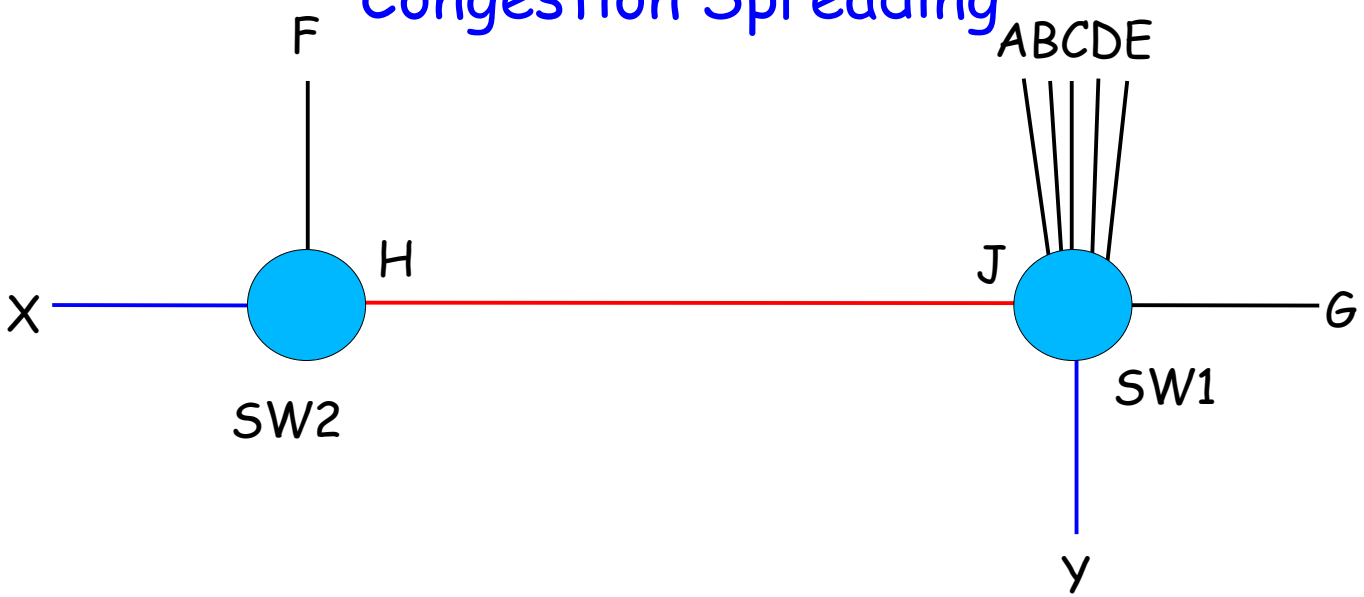
- Computer simulations are incorporating more complex physics
  - Non-local effects which lead to more stress on the network due to collective ops
  - Some applications spend > 70% of their runtime in MPI collective ops
- IB Net: efficient congestion control to prevent performance degradation
  - Congestion Control Architecture
- MPI: Improve collective algorithms and expand routing options
  - Optimized MPI Collectives
  - Source-based adaptive routing enhancement for InfiniBand
- Linux Kernel: reduce OS interference
  - Synchronized Kernel scheduler

Exhaustion of Link Credits Can Cause Congestion Spreading



- Ports A-E are all sending packets to Port G, Port G is receiving at 100% capability
- Port F also transmitting to Port G at 20% link bandwidth, will send until no more credits on inter-switch link
- Port X sending to Port Y at 20% link bandwidth
- Since data sent to Port G is consuming all the link (H to J) credits Port X cannot send to Port Y even at only 20% link bandwidth (throughput collapse)
- Root cause: Port has large queue of packets and available credits (Port G)
- Victim: Port has no credit to send packets (Port H)
- Port X is suffering from Head-of-Line blocking

Exhaustion of Link Credits Can Cause Congestion Spreading

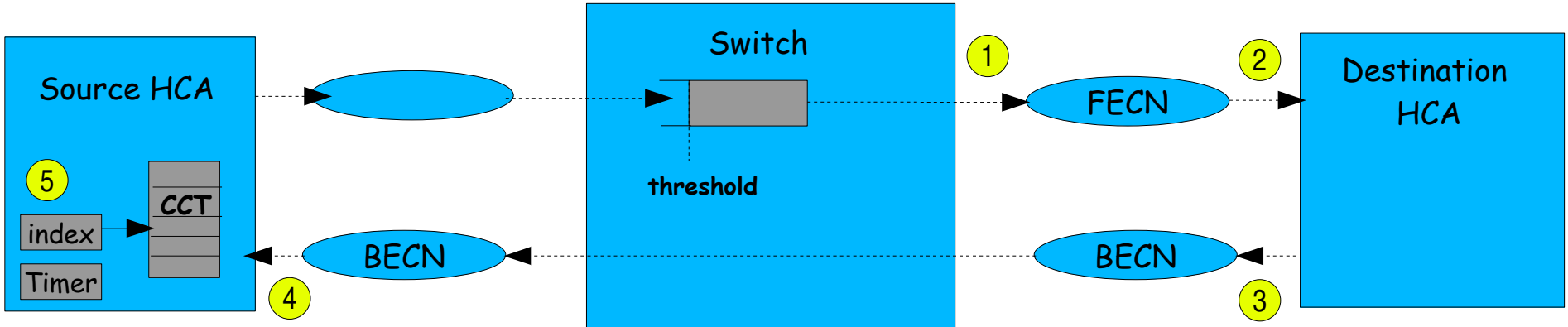


- Goal of Congestion Control is to avoid or eliminate congestion spreading
- Limit injection rates of ports that are the root cause of congestion (ports A-F) so that other ports are not effected (port X)
- Limiting injection rates of ports A-F does not cause them significant degradation (packets were waiting anyway)
- Packets from port X to port Y should be able to flow normally, since link credits will be available.

Evidence of Congestion Spreading has been seen on Thunderbird (and Virginia Tech cluster)

Developing more direct methods to diagnose congestion spreading

InfiniBand Congestion Control Architecture Avoids Congestion Spreading Problem



BECN: Backwards Explicit Congestion Notification
FECN: Forward Explicit Congestion Notification
CCT: Congestion Control Table

- (1) Congestion detection in switch. (2) Notification of congestion event to the target node. (3) Relay congestion notification back to source node(s). (4) Reduction of injection rate by source node(s). (5) Eventual recovery of initial rate after congestion has ceased.
- CCA supported in current generation of InfiniBand hardware
- Implement support in firmware (Mellanox) and software (Cisco and OpenIB)
- Optimized CCA parameters (IBM and SNL)
- Testing and evaluation at medium scale on Talon and large scale on Thunderbird (SNL)
- FY06 Deliverable: CCA features in OpenIB and specific tuning of parameters for Thunderbird

MPI Issues: Efficient collective routines

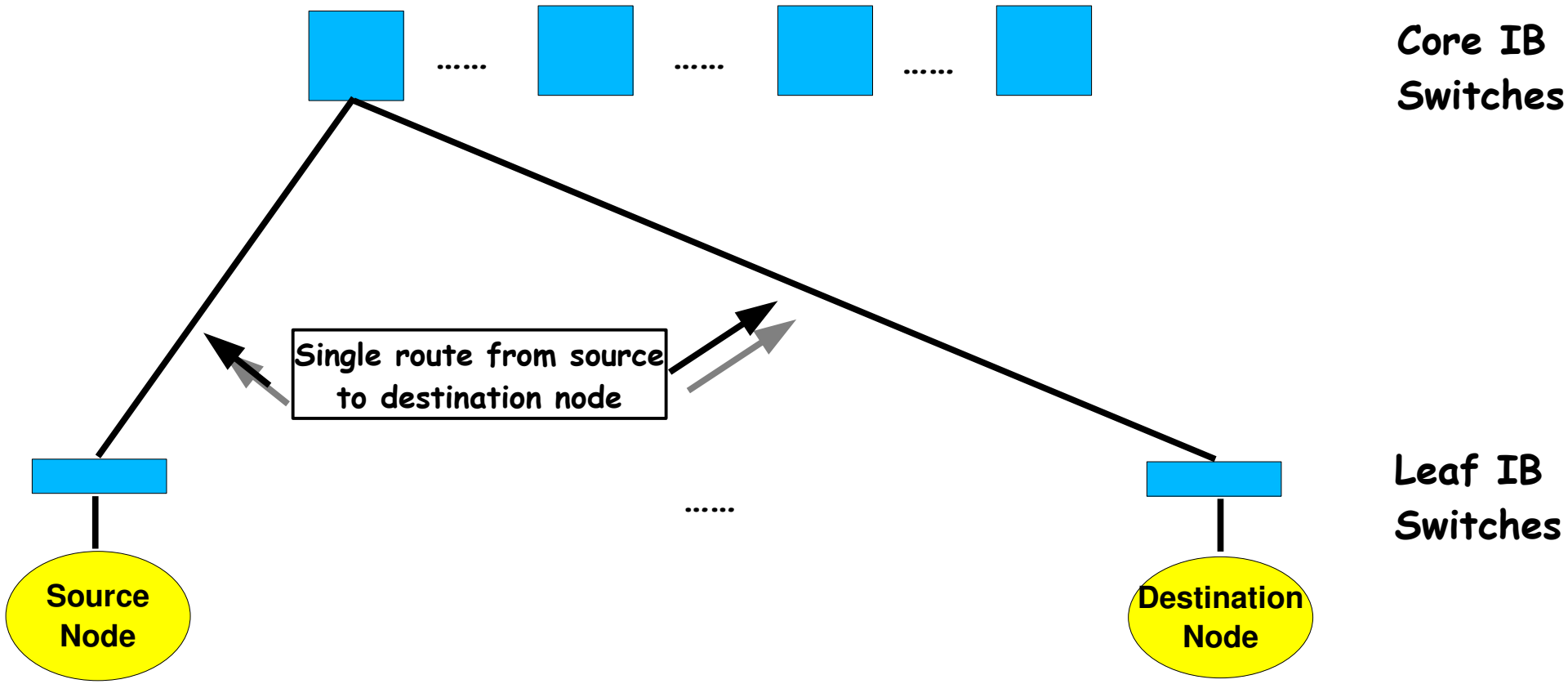
- Initially obtained very poor scaling
- MPI Allreduce and Alltoall were inefficient
- Wrote new collectives. Allreduce faster than improved MPICH2 algorithms
- Pipelined as follows:

```
Lvl Chunk
3   0    SdD Wai                                   RvD Wai
2   0    RvU WRe SdD Wai                   RvD Wai SdU Wai
1   0            RvU WRe SdD Wai RvD Wai SdU Wai
0   0                    RvU WRe SdU Wai
3   1        SdD Wai                                   RvD Wai
2   1        RvU WRe SdD Wai                   RvD Wai SdU Wai
1   1                RvU WRe SdD Wai RvD Wai SdU Wai
0   1                        RvU WRe SdU Wai
3   2            SdD Wai                                   RvD Wai
2   2            RvU WRe SdD Wai                   RvD Wai SdU Wai
1   2                    RvU WRe SdD Wai RvD Wai SdU Wai
0   2                            RvU WRe SdU Wai
```

MPI Issues: Efficient collective routines

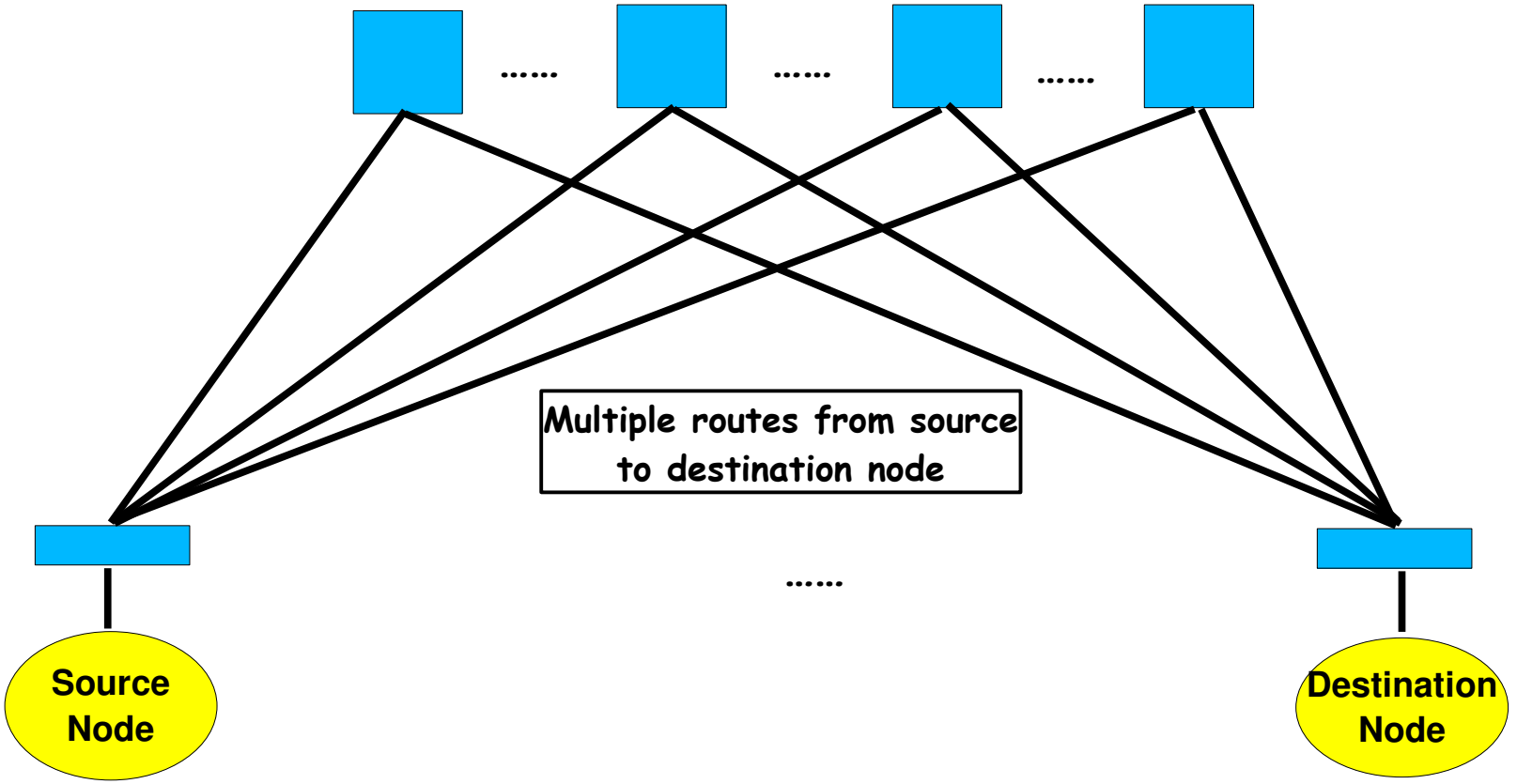
- Initial pass at optimized MPI Allreduce requires
  - Scalability testing beyond 100 nodes (target IB first)
  - Evaluation for interconnects other than InfiniBand
- Also focus on MPI broadcast
- Implementation/Integration into Open-MPI is part of a CSRF Project
- Scalability testing and evaluation are part of this project
- FY06 deliverable of optimized MPI Allreduce and broadcast into Open-MPI
- InfiniBand vendors will be supporting Open-MPI in FY06

Single Path Static Routing



- InfiniBand support static routing
- Single path deterministic static routing selects same path between src and dest nodes
- Single path static routing ignores path diversity and does a poor job of load balance
- Single path static routing is easy to implement and to make deadlock-free

Multi-Path Routing



- InfiniBand is not limited to single path routing
- InfiniBand supports 2^{LMC} multi-paths between src and dest
- InfiniBand specification limits single IB LID space to ~48,000 LIDs
- Max multi-path of 8 for Thunderbird cluster

Source-based Adaptive Routing

- Adaptive Routing Requires info on the state of the network
  - Split into space and time components
    - Local or global information
    - Current or historical
  - Global info will involve network backpressure and IB congestion control
  - Local info along with current/historical data will be evaluated in route selection
- Routing algorithms
  - InfiniBand Subnet Manager computes possible routes based on LMC
  - Set LMC > 0 (multi-path routing)
  - Evaluate 1, 2, 4, 8, and 16 paths; LMC={0, 1, 2, 3, 4}
- Multi-path route selection schemes will include:
  - Source-based adaptive using current queue lengths & history of data sent down each route
  - Random and Round-robin
- FY06 Deliverables: Multi-path adaptive routing support for IB in MPI

Linux Synchronized Kernel Scheduler

- Stretch goal for FY06 – May be able to leverage some work at LANL
- Scalable MPI collective communication has two key components:
  - MPI collective algorithms optimized for performance and scalability
  - Reduction of rogue OS behavior and OS interference
- Rogue OS behavior can be reduced/eliminated by proper kernel configuration and some OS interference can be reduced by tuning of kernel parameters (interrupt timer, etc.)
- Variance in OS process scheduling across compute nodes can have a negative impact on scalability
  - Process synchronization can alleviate this issue
  - Port Cray sync scheduler to latest kernel.org Linux kernel
  - Replace hardware network clock with btime (LANL)
  - Evaluate performance and develop recommendations for potential IB hardware network clock

Evaluate Scalability and Performance Improvements

- Evaluate performance and scalability
  - MPI benchmark suite (Sandia cbench) and several Sandia apps
  - Comparison with and without congestion control
  - Comparison to other MPI collective algorithms
  - Comparison of various routing selection schemes
  - Comparison of MPI collectives and apps with and without sync Linux kernel scheduler
- Computing Platforms
  - Use Catalyst/Talon (200 + 128 nodes) and Thunderbird (4480 nodes)
  - Currently working with LLNL on comparisons between Thunderbird (InfiniBand), Thunder (Quadrics Elan4), and Purple (IBM Federation)
  - OpenIB driver for PathScale iPath HCA (collaboration with LANL)

Other Programs Leveraging Open Source Project

- CSRF InfiniBand Architectures
- Sandia Thunderbird Cluster
- OpenIB Alliance (25 members from across industry and Tri-labs)
- GFS projects and OpenSFS (Open Scalable File System) Alliance
- Advanced Networking: iSCSI and PNFS
- SNL visualization – Red RoSE
- Performance and Debug Tools
- LANL capability machine (FY06), capacity, and visualization platforms
- LLNL visualization and capacity (FY06) platforms
- Tri-lab Advanced Architectures
- Naval Research Lab, NCSA, PSC, LBL, and Ames Lab
- Financial institutions – JP Morgan Chase, Credit Suisse First Boston, DE Shaw, etc.

Hypothetical Budget Cuts

- Budget already cut relative to FY05
- Many of the components of this project are critical to user productivity on IB clusters
- InfiniBand software development was selected as the top software priority by ASC PF program
- Additional cuts of 25%
  - Ability to make immediate impact on productivity would be greatly reduced
  - All DoE and external programs leveraging this work would suffer
  - Sandia role as leader in InfiniBand architectures would be greatly reduced
  - Reduced effectiveness in leveraging InfiniBand expertise to obtain WFO funding around InfiniBand architectures
  - Reduces impact of OpenIB to other IB platforms, current and future FY06 purchases, in Tri-labs
- Additional cuts of 50%
  - OpenIB PF would not be able to deliver on promised performance (developers need feedback from HPC users)
  - Reduced confidence on stability of OpenIB software stack
  - Reduced ability to collaborate with external parties (IBM, Cisco, etc.) to solve critical networking issues

FY07 Goals and Vision

- InfiniBand
  - QoS (Quality of Service): Service Level to Virtual Lane mapping to administor/partition IB resources
  - Evaluation of InfiniBand fully adaptive routing
  - Dynamic selection of congestion control parameters
  - Potential for OpenIB funding to include a Stage 4
  - Simulation of larger-scale IB fabrics for Petascale architectures
- Linux kernel development
  - Dynamic Co-scheduler
- MPI
  - Collectives – Alltoall, and others based on SNL how apps use collectives
  - Application specific tuning of MPI/IB multi-path and adaptive routing
- Merge of OpenIB and iWarp (RDMA over Ethernet)
  - Vendors already investing in key areas, but may need better diag tools and scalability

For more information

Matt Leininger
mlleini@sandia.gov

www.openib.org

National Nuclear Security Administration

Sandia National Laboratories