An Efficient Hardware-Software Approach to Network Fault Tolerance with InfiniBand

Abhinav Vishnu[1]

Manoj Krishnan[1] and Dhableswar K. Panda[2]

Pacific Northwest National Lab[1]

The Ohio State University[2]

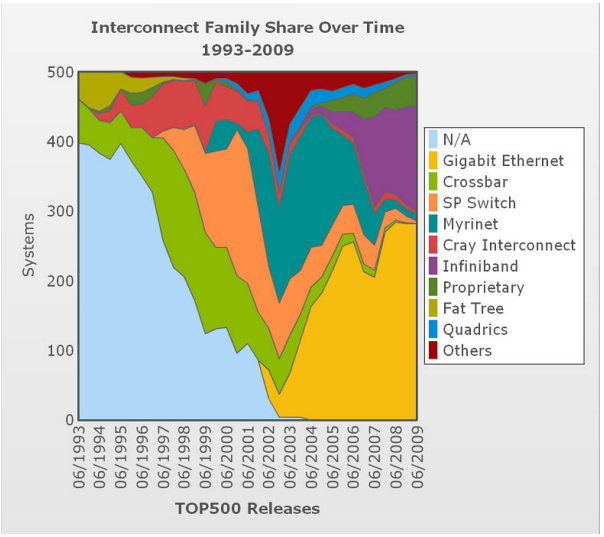
Pacific Northwest
NATIONAL LABORATORY

Outline

- Introduction
- Background and Motivation
  - InfiniBand Network Fault Tolerance Primitives
- Hybrid-IBNFT Design
  - Hardware-IBNFT and Software-IBNFT
- Performance Evaluation of Hybrid-IBNFT
  - Micro-benchmarks and NWChem
- Conclusions and Future Work

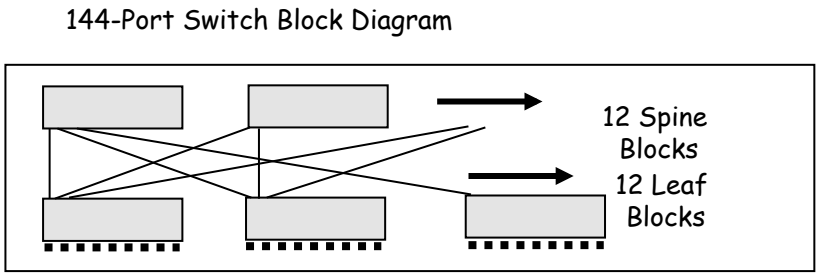
Introduction

- Clusters are observing a tremendous increase in popularity
  - Excellent price to performance ratio
  - 82% supercomputers are clusters in June 2009 TOP500 rankings
- Multiple commodity Interconnects have emerged during this trend
  - InfiniBand, Myrinet, 10GigE ….
- InfiniBand has become popular
  - Open standard and high performance
  - **~30% TOP500 systems use InfiniBand**
- Various topologies have emerged for interconnecting InfiniBand
  - Fat tree is the predominant topology
    - TACC Ranger, PNNL Chinook

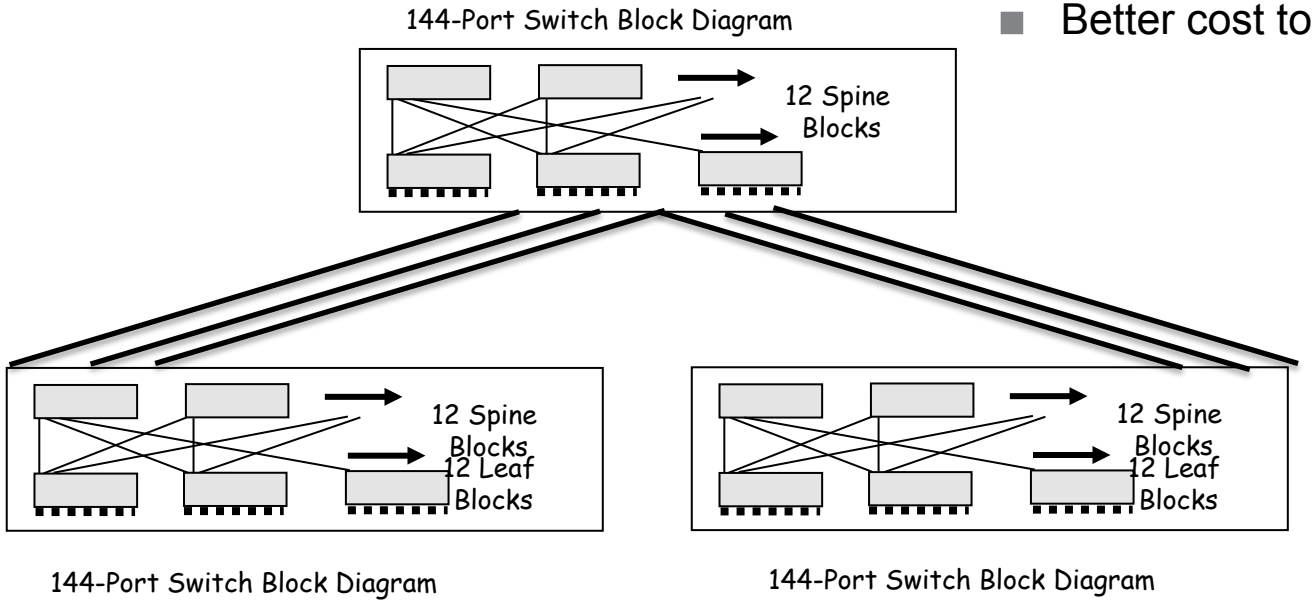


Pacific Northwest
NATIONAL LABORATORY

Typical InfiniBand Fat Tree Configurations



- Multiple leaf and spine blocks
  - Available in 144, 288 and 3456 port combinations
- Multiple paths are available between nodes present on different switch blocks
- Oversubscribed configurations are becoming popular
  - Better cost to performance ratio

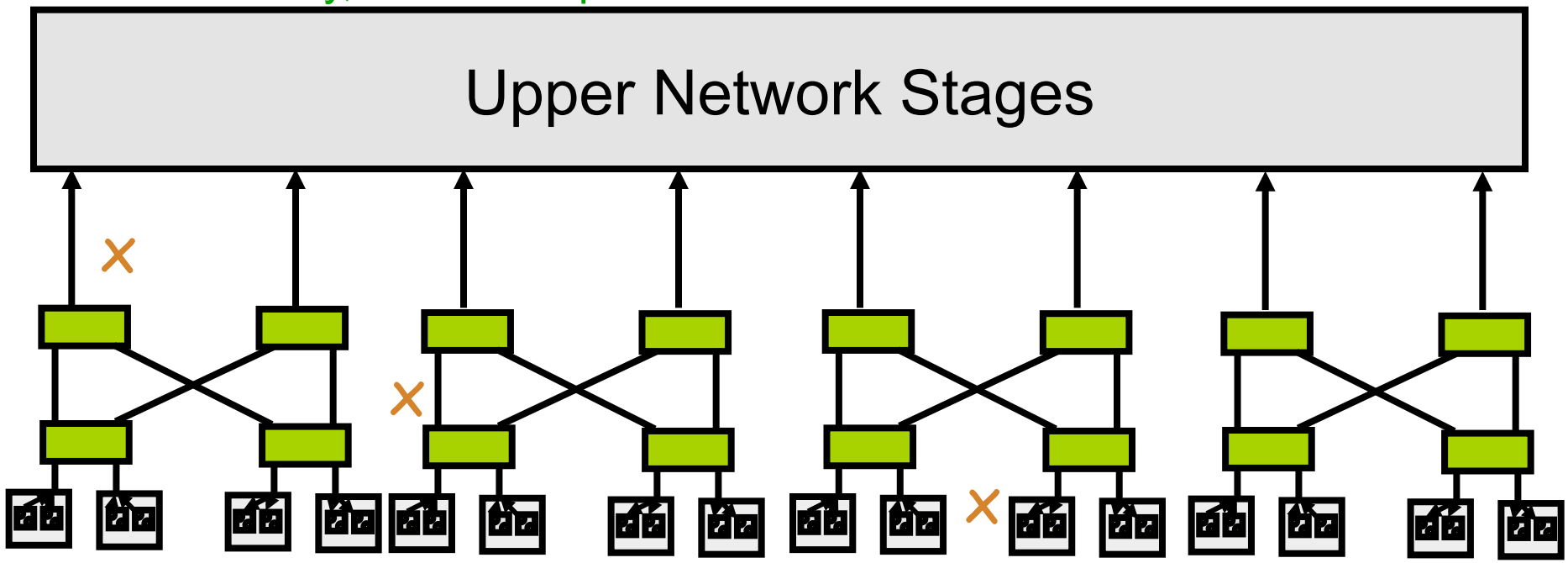


Pacific Northwest NATIONAL LABORATORY

Network Faults

Links/Switches/Adapters may fail with reduced MTBF
(Mean time between failures)

Fortunately, InfiniBand provides mechanisms to handle these issues!!

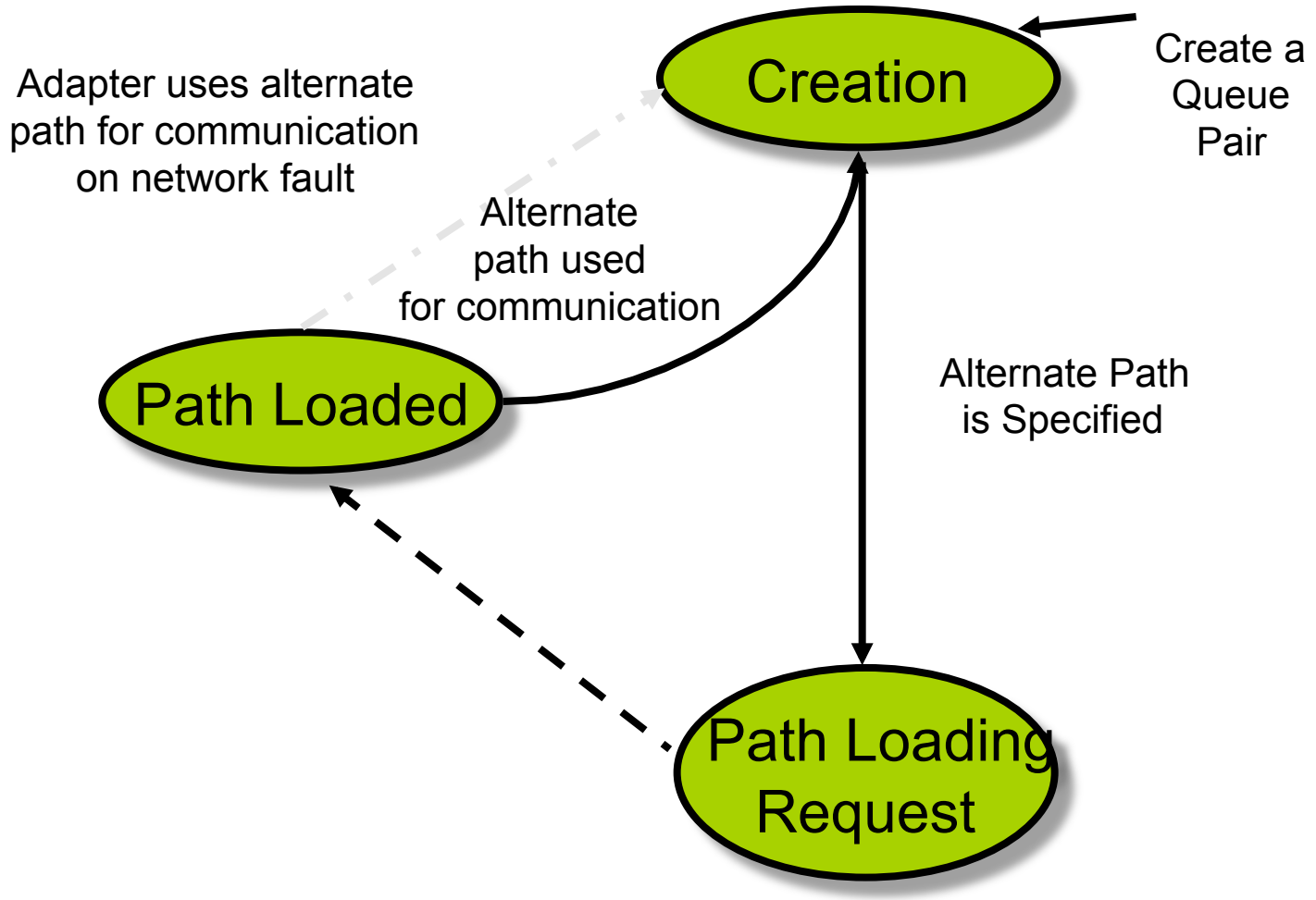


Pacific Northwest
NATIONAL LABORATORY

Outline

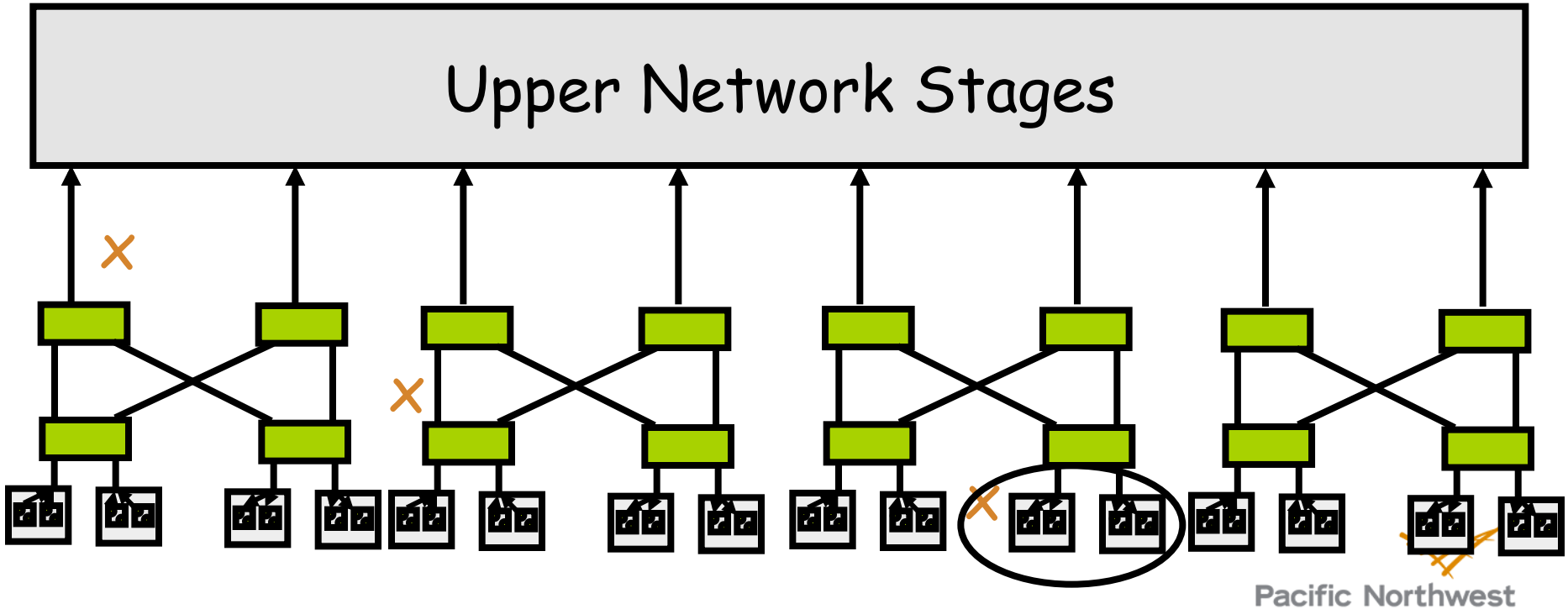
- Introduction
- Background and Motivation
  - InfiniBand Network Fault Tolerance Primitives
- Hybrid-IBNFT Design
  - Hardware-IBNFT and Software-IBNFT
- Performance Evaluation of Hybrid-IBNFT
  - Micro-benchmarks and NWChem
- Conclusions and Future Work

State Machine - Automatic Path Migration



Automatic Path Migration - Practice

- Pros
  - User transparent network fault detection and failover
  - Important for applications exhibiting long computation phases
- Cons
  - **Fails if alternate path also has a fault**
  - Increasingly common with upcoming fat tree topologies



Software Based Fault Detection and Failover

- Reliable connection transport semantics
  - In-order data delivery
  - **Exact once notification** for data transfer failure
  - **Connection cannot be used** further for data transfer without recycling the connection
- **Software based method can handle the scenarios where Automatic Path Migration fails**
  - Requires layer using software based method to manually detect faults and perform network failover
- Clearly, any of the individual methods for network fault tolerance is not sufficient ..

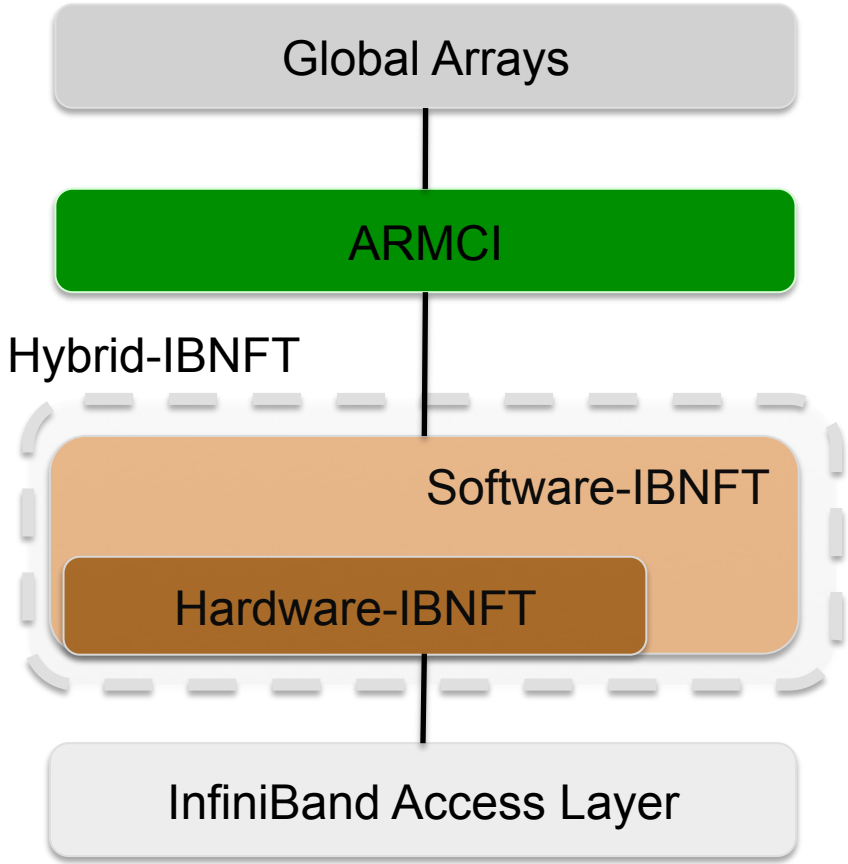
Problem Statement

- To handle different scenarios of network faults, how to efficiently design hybrid approach (Hybrid-IBNFT) for InfiniBand Network fault tolerance
  - Leverage APM based approach as much as possible (Hardware-IBNFT)
  - Fall back to the software based approach, only when necessary (Software-IBNFT)
  - Study the design challenges and performance implications

Outline

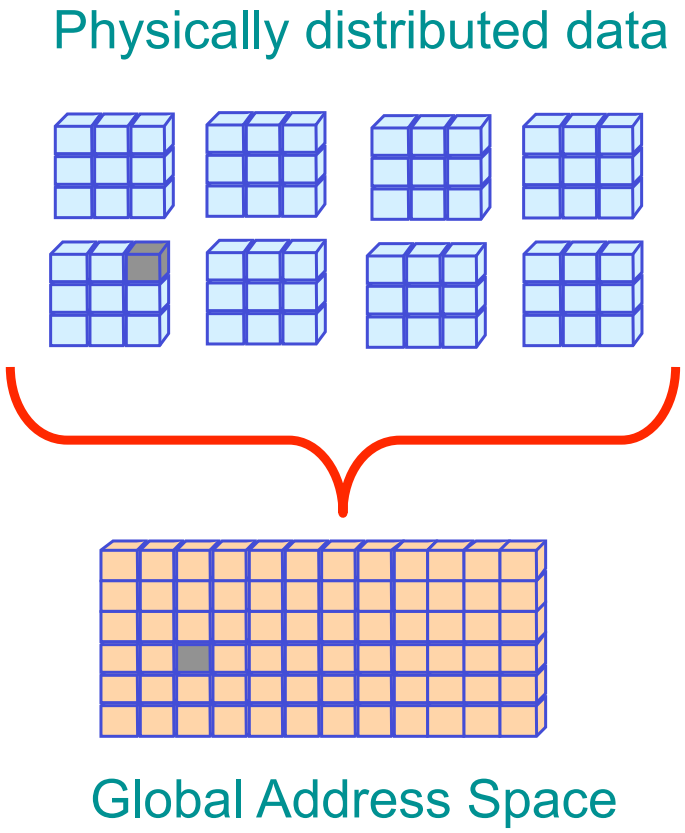
- Introduction
- Background and Motivation
  - InfiniBand Network Fault Tolerance Primitives
- Hybrid-IBNFT Design
  - Hardware-IBNFT and Software-IBNFT
- Performance Evaluation of Hybrid-IBNFT
  - Micro-benchmarks and NWChem
- Conclusions and Future Work

Overall Design of Hybrid-IBNFT



- Hybrid-IBNFT
  - Hardware-IBNFT
  - Software-IBNFT
- The approach is generic enough for runtime systems for other programming models and languages
- We use Global Arrays and ARMCI for designing Hybrid-IBNFT

Global Arrays

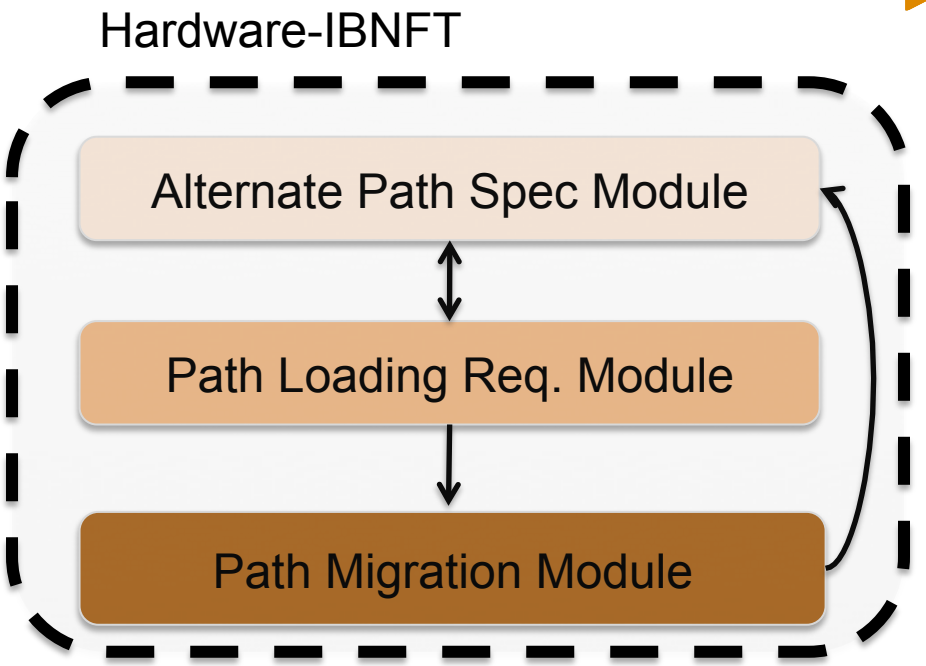


- Global Arrays is a distributed-shared object programming model
  - GA presents a shared view of physically distributed dense array objects over the nodes of a cluster
  - Provides one-sided communication model using Put/Get and Accumulate semantics
  - Used in wide variety of applications
  - Computational Chemistry
    - NWChem, molcas, molpro …
  - Bioinformatics
    - ScalaBLAST
  - Upcoming Applications
    - STOMP

Aggregate Remote Memory Copy Interface

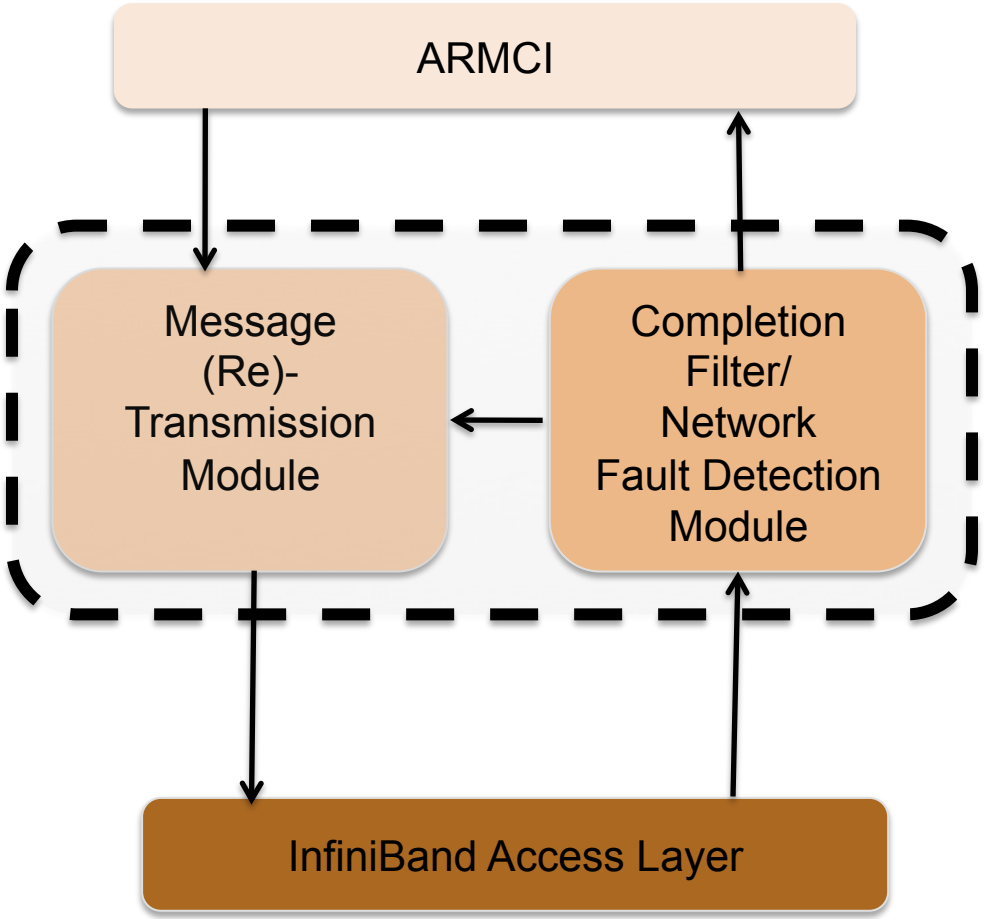
- Runtime system for Global Arrays
  - Being used with other programming models
- Provides one-sided communication primitives
  - Put, get accumulate variants
- Available on commodity interconnects
  - InfiniBand, Myrinet …
- Leadership Class machines
  - IBM BlueGene
  - Cray XTs

Hardware-IBNFT



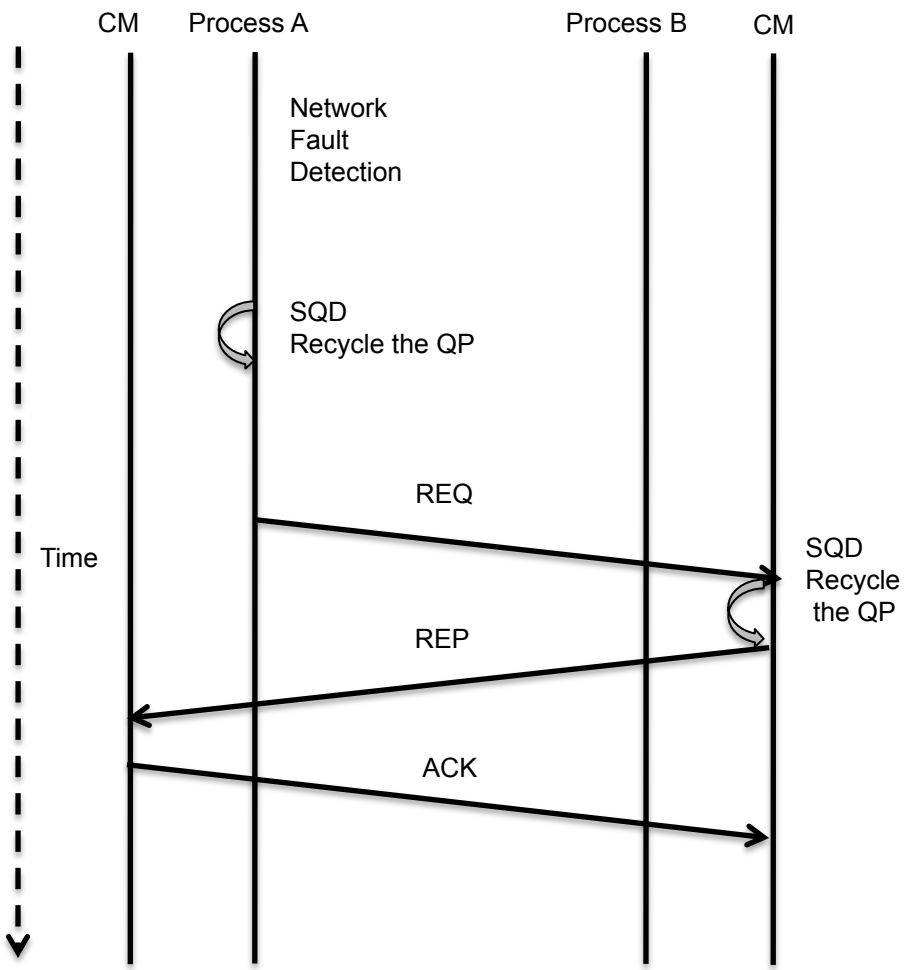
- Multiple independent modules for Automatic Path Migration
  - Alternate path specification module
    - Specification of alternate path for network failover
  - Path Loading request module
    - Request the alternate path to start loading in the APM state machine
  - Path Migration Module
    - Can be used for manual transition

Software-IBNFT



- **Network Fault detection module**
  - Filters whether a data transfer is a success/failure
  - Forwards the message to message re-transmission module
- **Message (re)-transmission module**
  - Transmits the data on
    - Request from ARMCI layer
    - Occurrence of a network failure

Communication Protocol Re-establishment

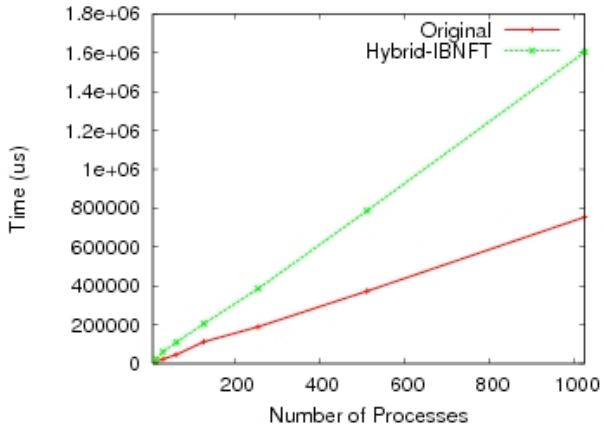


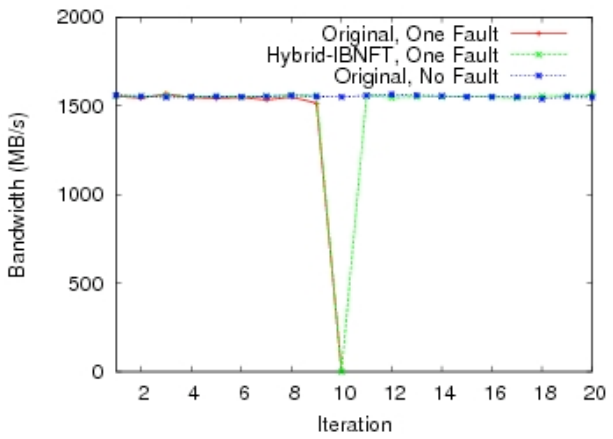
- On detection of network fault
  - Transition the QP to send queue drained
  - QP is in un-usable state
  - Recycle the QP to Ready-to-send state
  - 3-Way Acknowledgement protocol
    - REQ, REP and ACK
- Connection Manager thread is blocking
  - Active only on interrupts

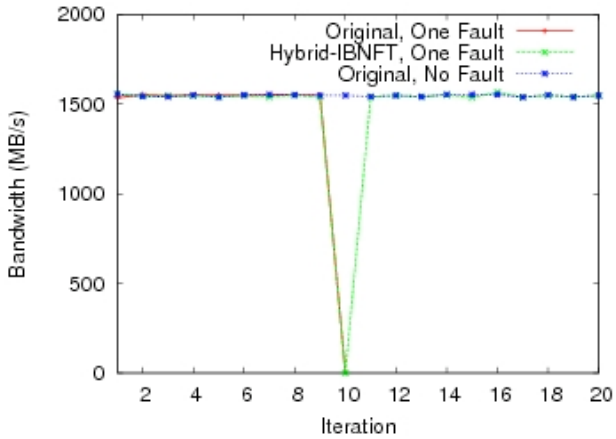
Outline

- Introduction
- Background and Motivation
  - InfiniBand Network Fault Tolerance Primitives
- Hybrid-IBNFT Design
  - Hardware-IBNFT and Software-IBNFT
- Performance Evaluation of Hybrid-IBNFT
  - Micro-benchmarks and NWChem
- Conclusions and Future Work

Performance Evaluation with Microbenchmarks

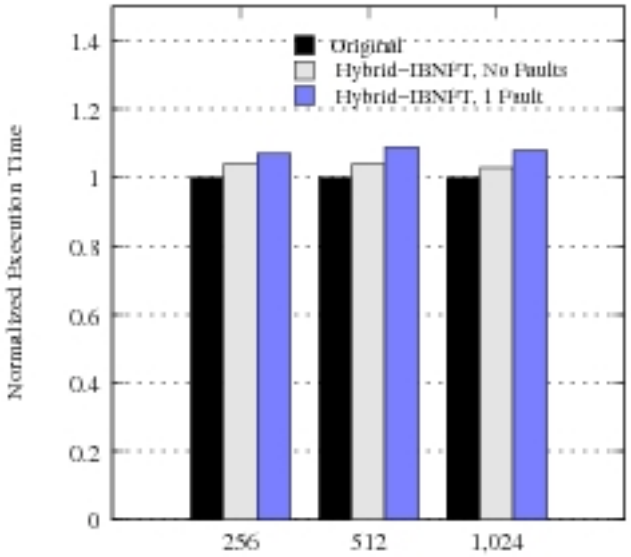




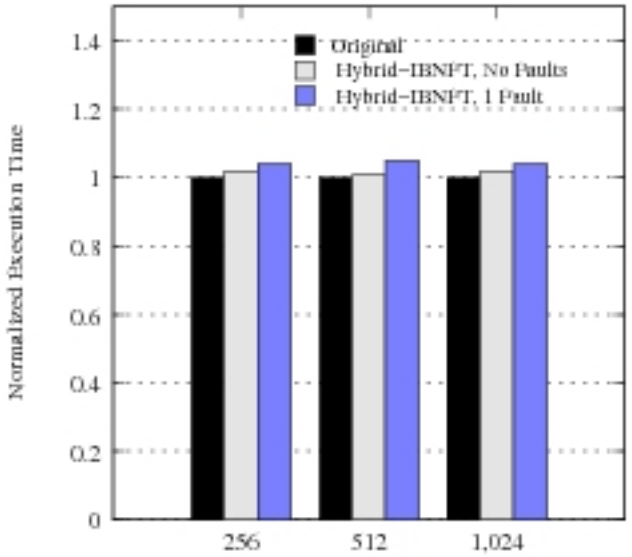


- Time for APM transitions scales linearly
- Modification of Put and Get Tests
  - Execute for one iteration
  - See the performance point at failure
- Observed bandwidth ~0 at the point of failure

Performance Evaluation with NWChem



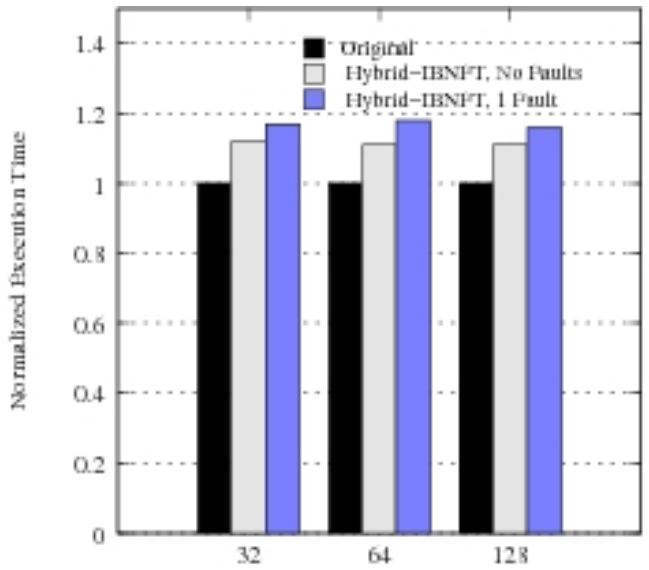
h2o7



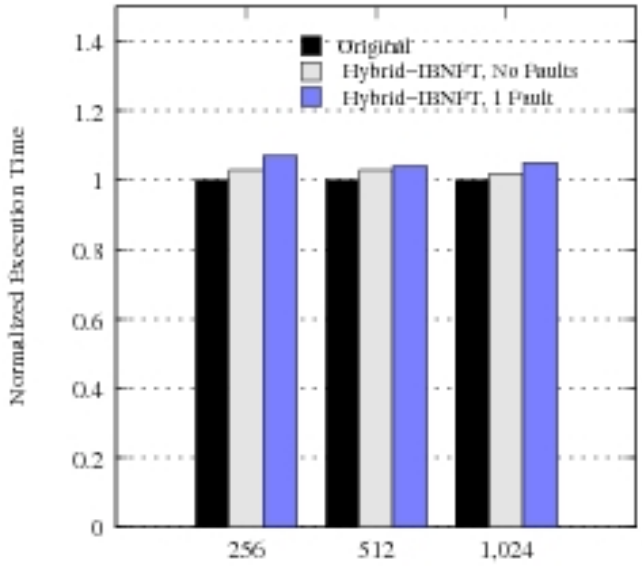
pentane

- For h2o7, negligible overhead is observed on occurrence of a fault and comparison with original schemes
- For pentane, the observed overhead is less than 5%

Performance Evaluation with NWChem (Contd).



siosi3



siosi7

- siosi3 executes in the order of seconds, and observes a significant overhead on occurrence of a fault
- siosi7 executes for multiple minutes and observes negligible overhead on fault occurrence

Outline

- Introduction
- Background and Motivation
  - InfiniBand Network Fault Tolerance Primitives
- Hybrid-IBNFT Design
  - Hardware-IBNFT and Software-IBNFT
- Performance Evaluation of Hybrid-IBNFT
  - Micro-benchmarks and NWChem
- Conclusions and Future Work

Conclusions and Future Work

- Conclusions
  - Multiple mechanisms for network fault tolerance are needed to handle different network fault scenarios
  - We presented a design for hybrid network fault tolerance using hardware and software mechanisms
  - Using micro-benchmarks and NWChem, our implementation exhibits small overhead for applications executing for a longer period of time
- Future Work
  - We plan to perform larger scale evaluations with different network fault injection methodology
  - Study the benefits of hybrid network fault tolerance with other programming models and applications

Questions

- Global Arrays
  - http://www.emsl.pnl.gov/docs/global/
- ARMCI
  - http://www.emsl.pnl.gov/docs/parsoft/armci/
- HPC-PNL
  - http://hpc.pnl.gov
- Nowlab-OSU
  - http://nowlab.cse.ohio-state.edu