A Scalable InfiniBand Network Topology-Aware Performance Analysis Tool for MPI

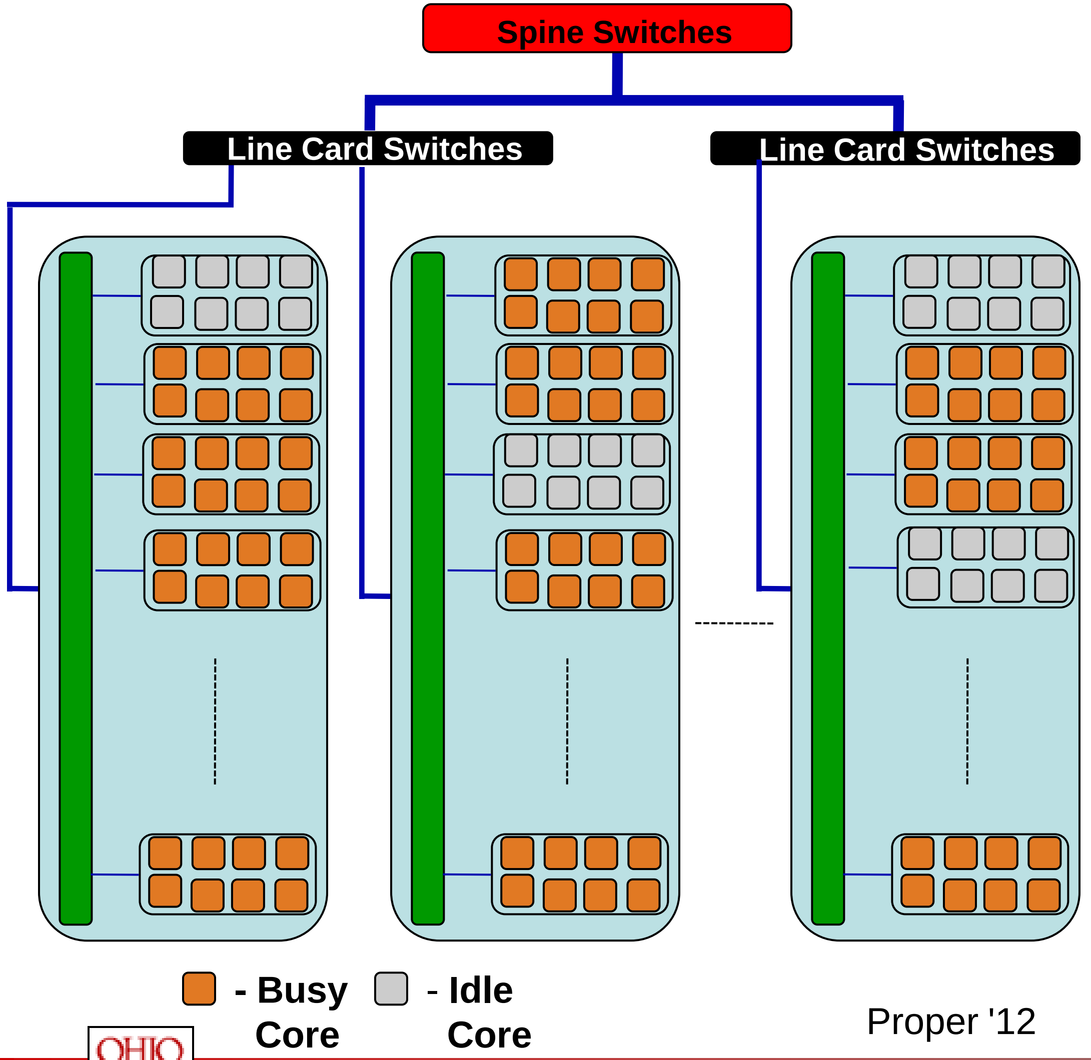
Hari Subramoni, Jerome Vienne, and Dhabaleswar. K. Panda

Department of Computer Science and Engineering
The Ohio State University

Outline

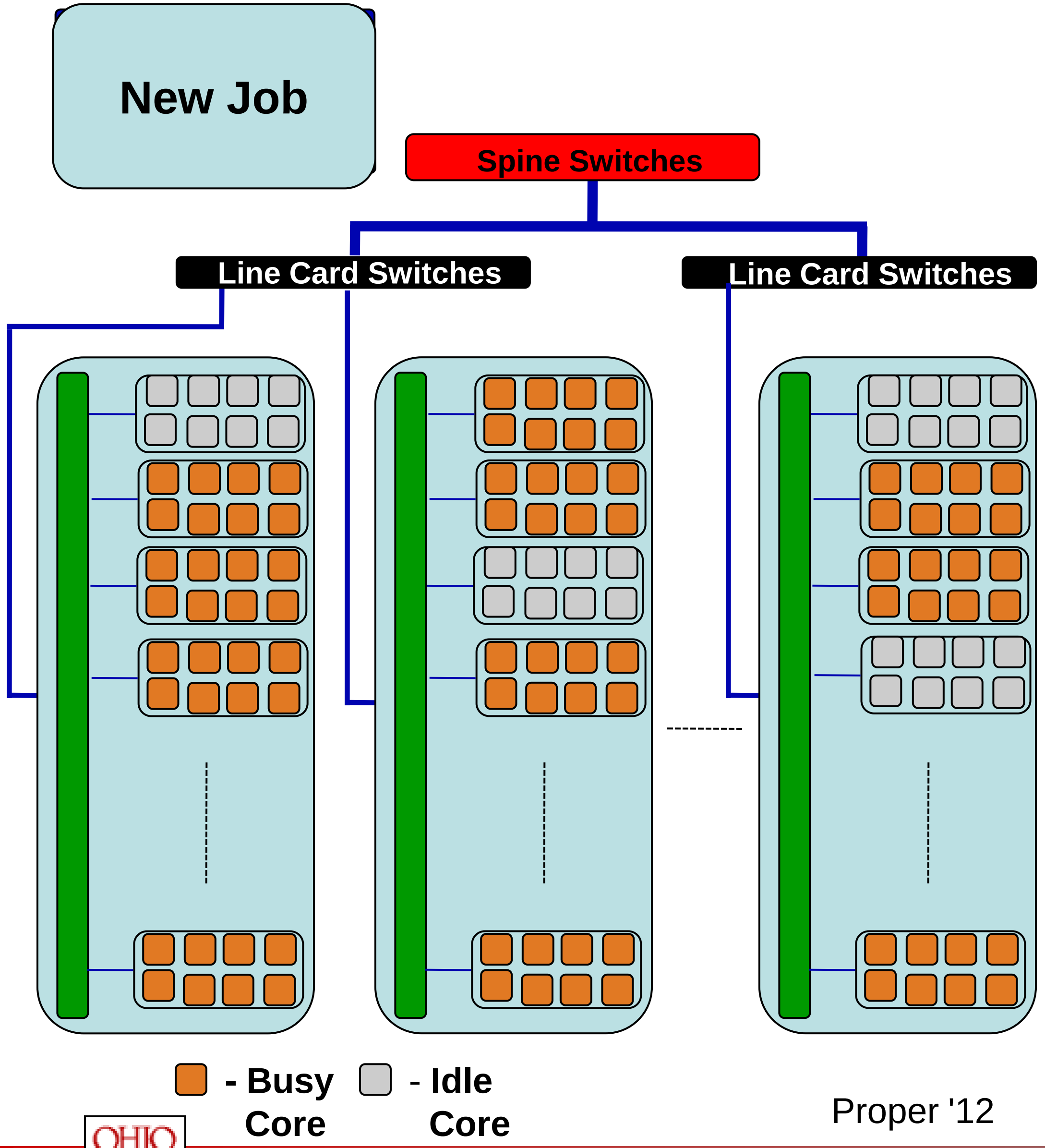
- Introduction
- Problem Statement
- Design of Network Topology-Aware Performance Analysis Tool for MPI
- Performance Evaluation
- Conclusions and Future Work

Introduction



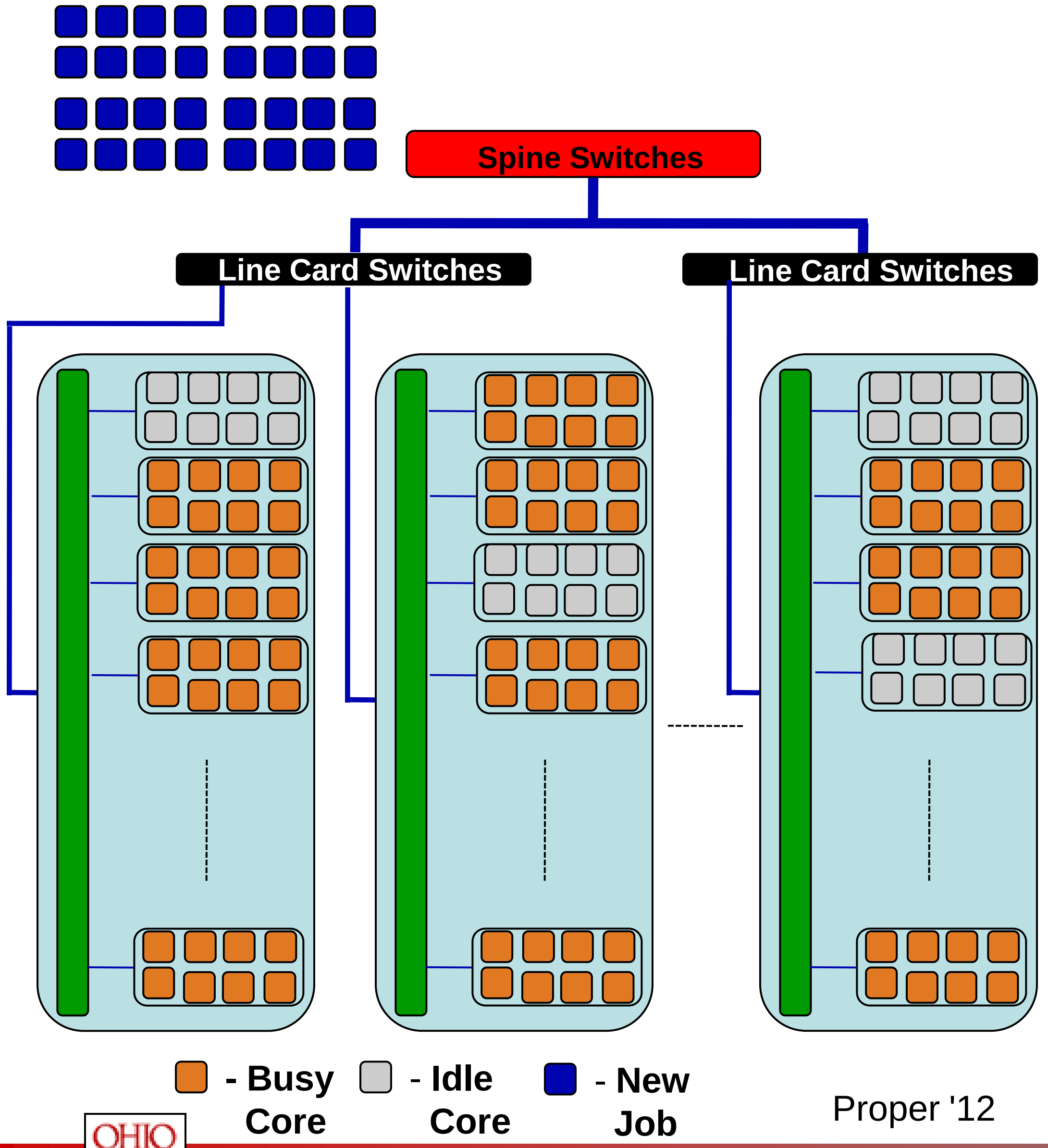
- Supercomputing systems organized as racks of nodes interconnected using complex network architectures

Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs

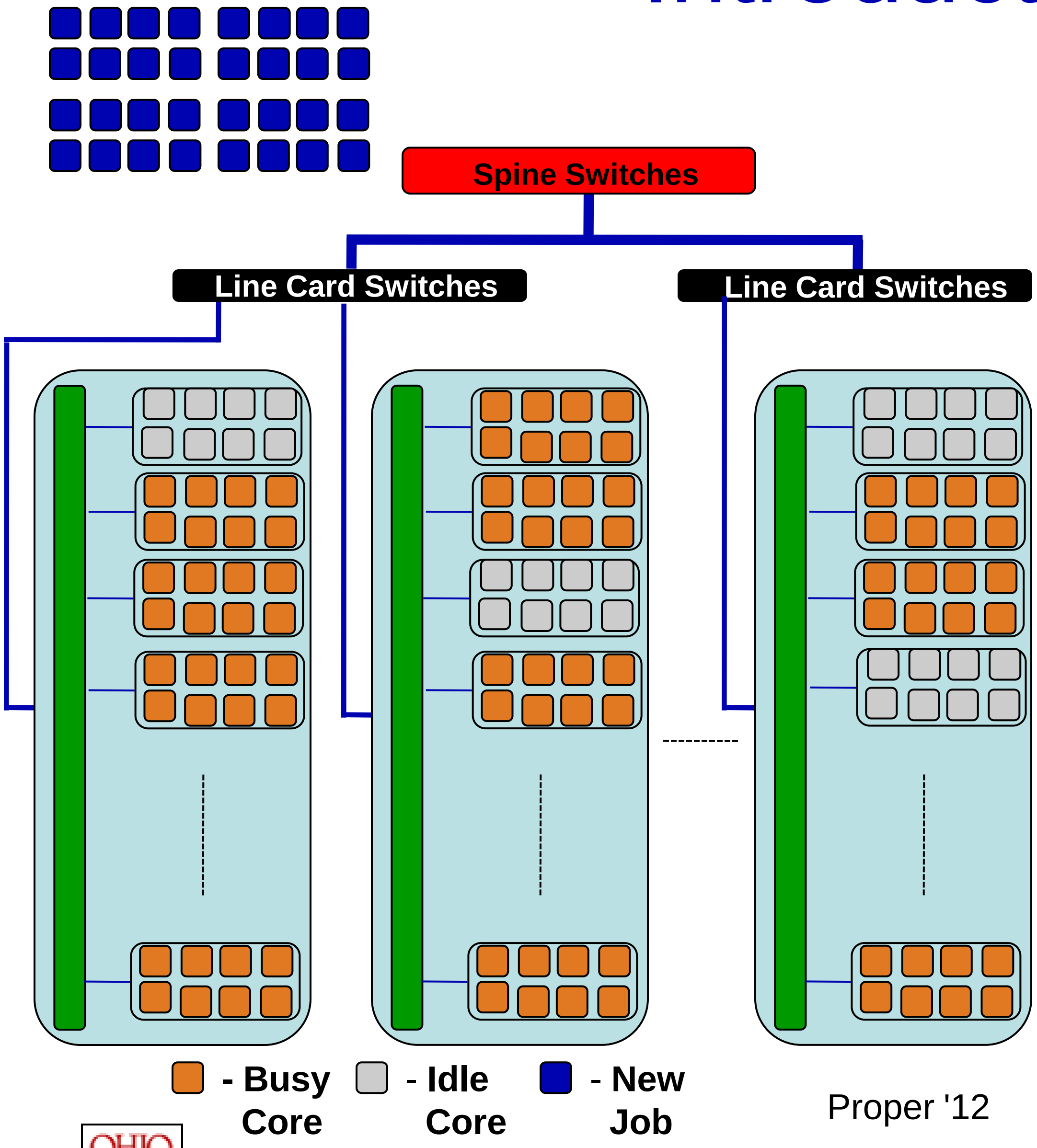
Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs

Proper '12

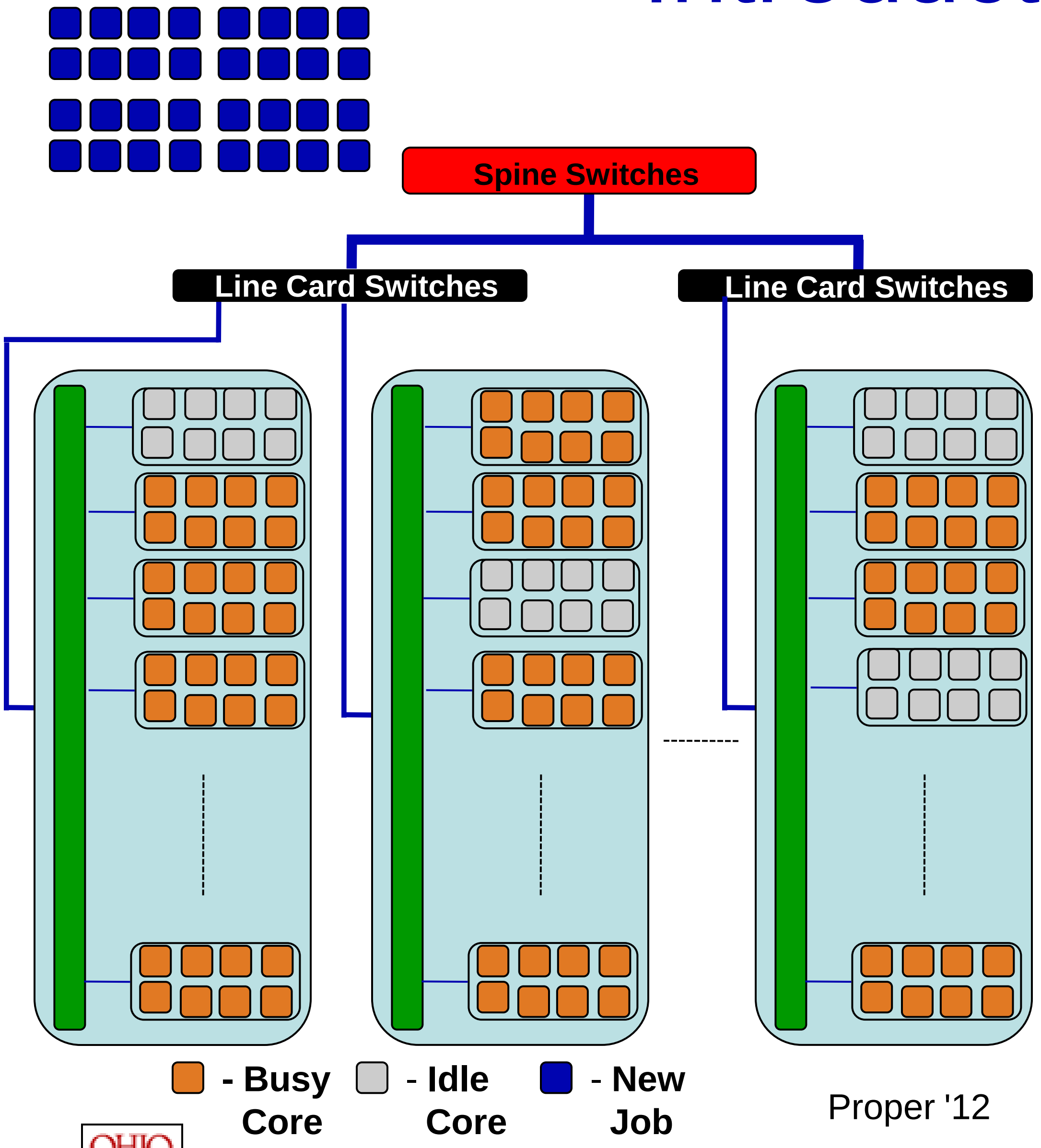
Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high

Proper '12

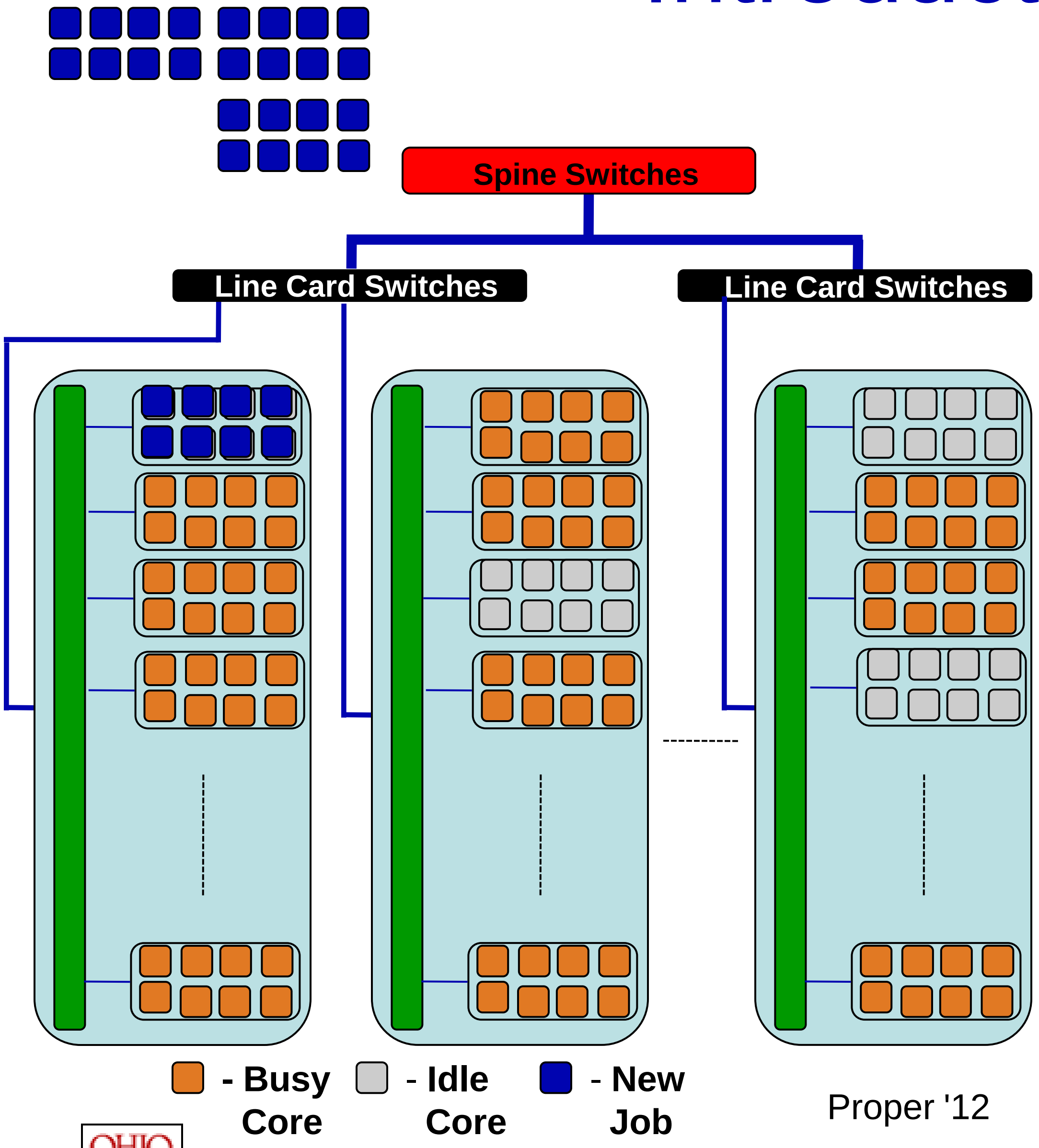
Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high
- Individual processes belonging to one job can get scattered

Proper '12

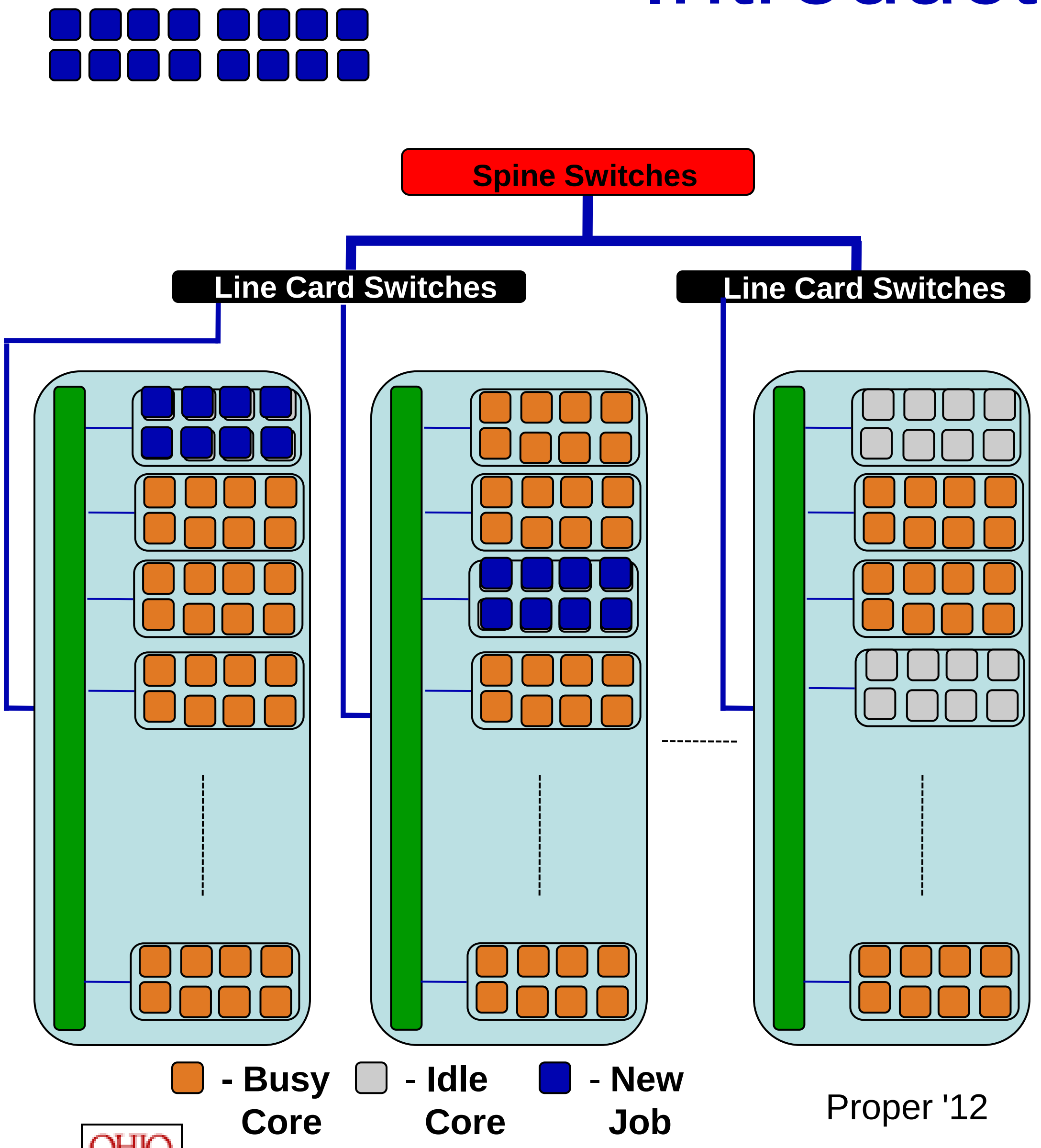
Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high
- Individual processes belonging to one job can get scattered

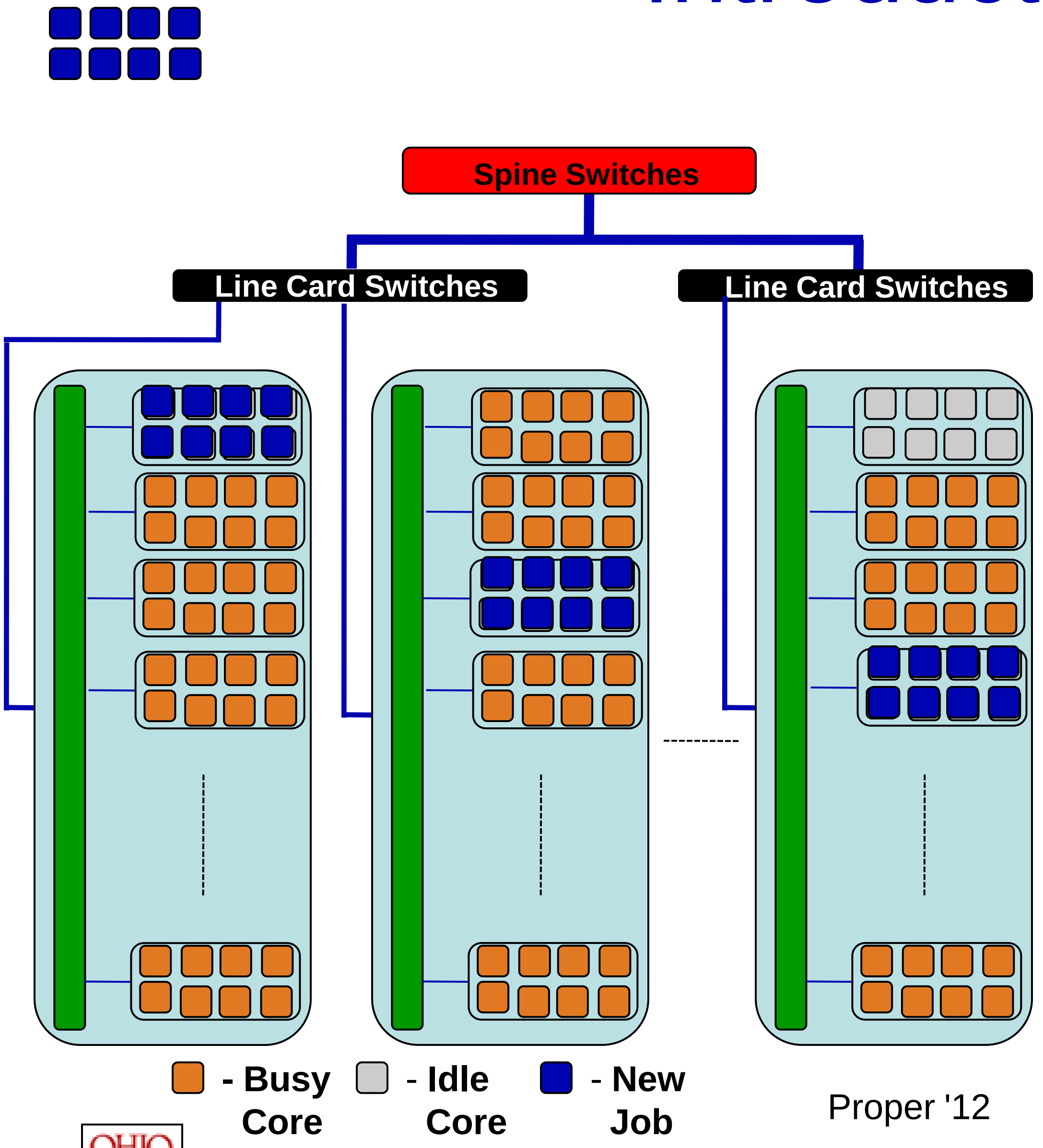
Proper '12

Introduction



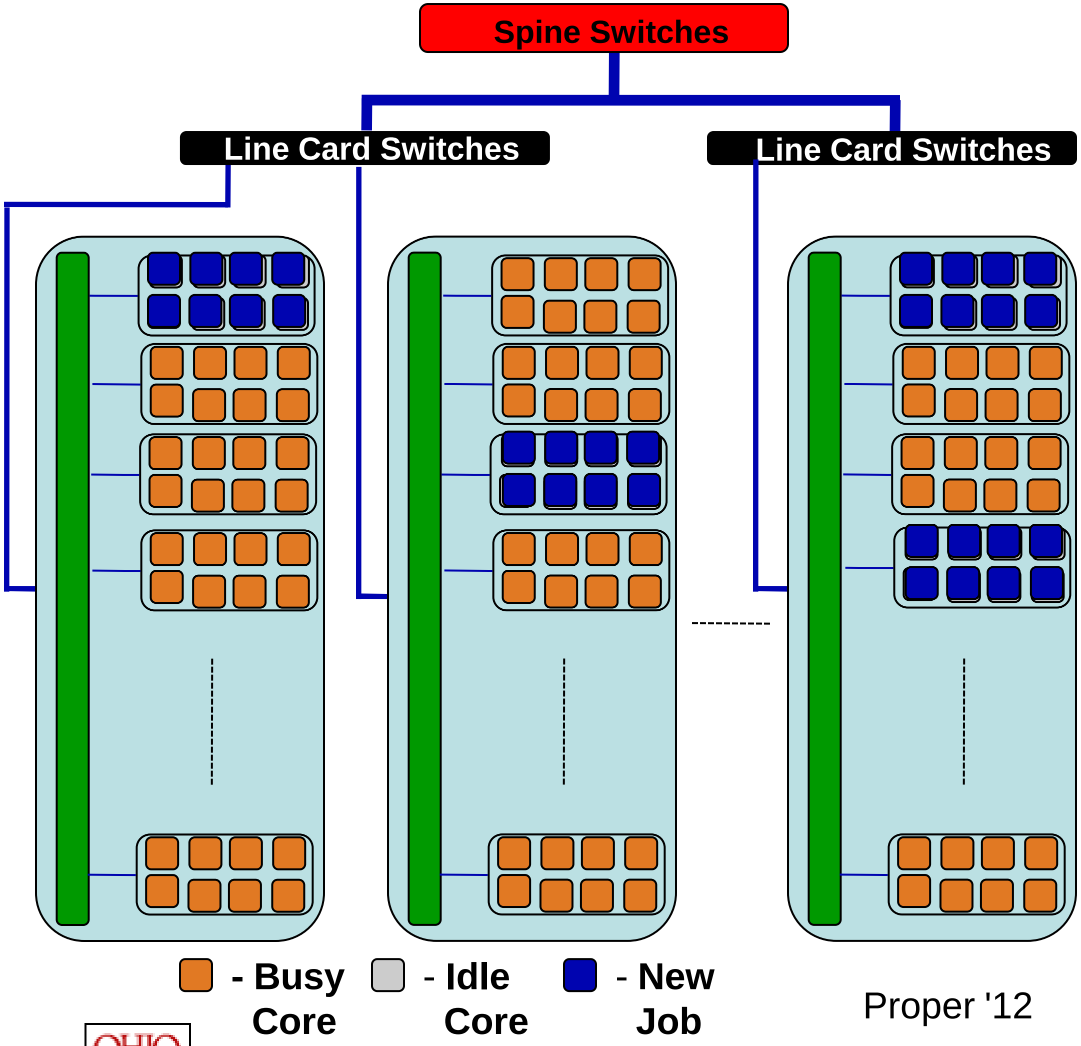
- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high
- Individual processes belonging to one job can get scattered

Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high
- Individual processes belonging to one job can get scattered

Introduction



- Supercomputing systems organized as racks of nodes interconnected using complex network architectures
- Job schedulers used to allocate compute nodes to various jobs
- Primary responsibility of scheduler is to keep system throughput high
- Individual processes belonging to one job can get scattered

Proper '12

Introduction

MPI communication performance across varying levels of switch topology on TACC Ranger

Process Location	Number of Hops	MPI Latency (us)
Intra-Rack	1 Hops in Leaf Switch	1.57
Inter-Rack	3 Hops Across Spine Switch	2.45
	5 Hops Across Spine Switch	2.85

Introduction

MPI communication performance across varying levels of switch topology on TACC Ranger

Process Location	Number of Hops	MPI Latency (us)
Intra-Rack	1 Hops in Leaf Switch	1.57
Inter-Rack	3 Hops Across Spine Switch	2.45
	5 Hops Across Spine Switch	2.85

81% Worse

- Performance degrades as number of hops increases

Introduction

MPI communication performance across varying levels of switch topology on TACC Ranger

Process Location	Number of Hops	MPI Latency (us)
Intra-Rack	1 Hops in Leaf Switch	1.57
Inter-Rack	3 Hops Across Spine Switch	2.45
	5 Hops Across Spine Switch	2.85

81% Worse

- Performance degrades as number of hops increases
- Critical to understand the communication overheads caused due to network topology

Introduction

MPI communication performance across varying levels of switch topology on TACC Ranger

Process Location	Number of Hops	MPI Latency (us)
Intra-Rack	1 Hops in Leaf Switch	1.57
Inter-Rack	3 Hops Across Spine Switch	2.45
	5 Hops Across Spine Switch	2.85

81% Worse

- Performance degrades as number of hops increases
- Critical to understand the communication overheads caused due to network topology
- Need a tool to analyze and visualize the communication pattern in a network-topology-aware manner

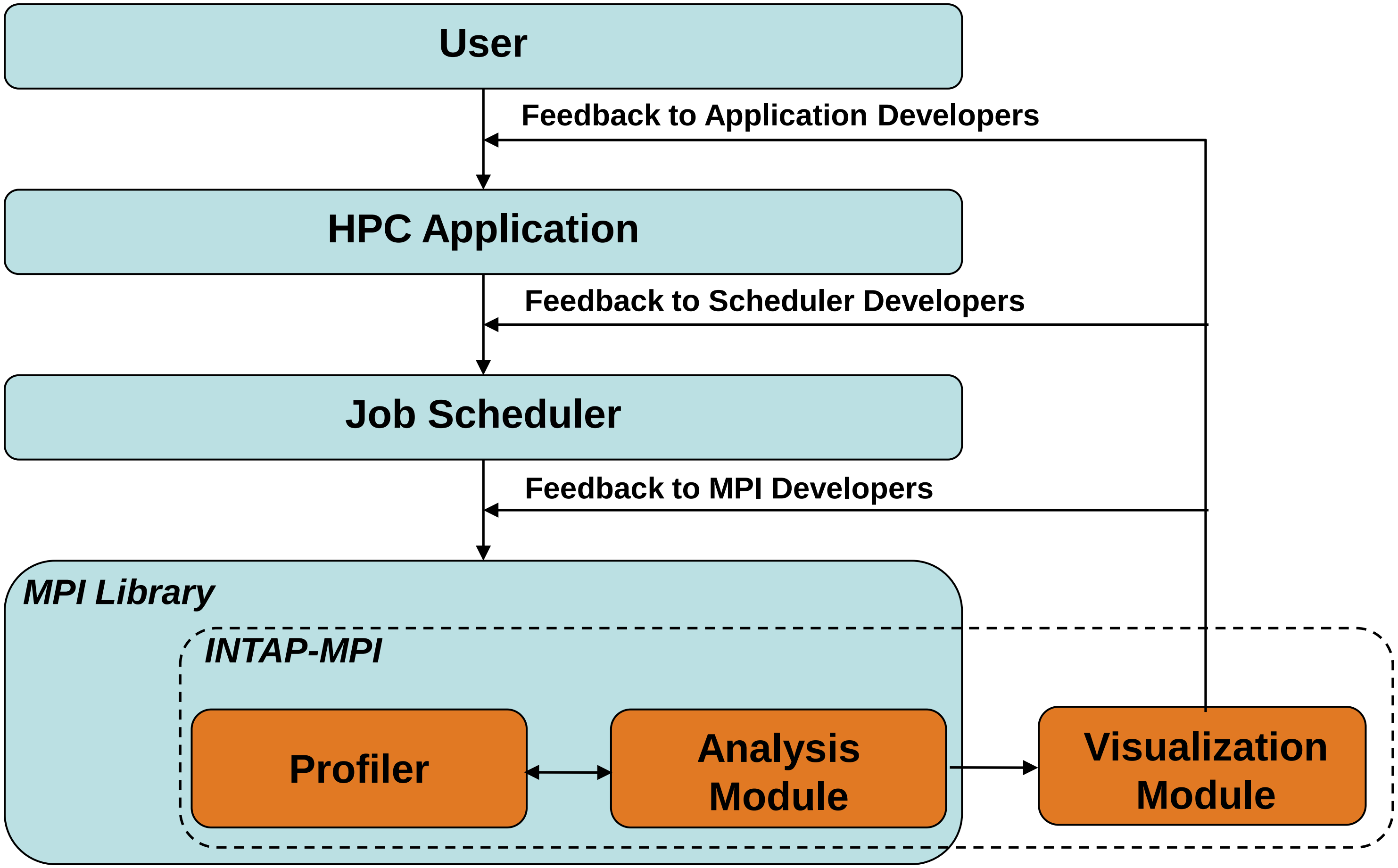
Outline

- Introduction
- Problem Statement
- Design of Network Topology-Aware Performance Analysis Tool for MPI
- Performance Evaluation
- Conclusions and Future Work

Problem Statement

Can a scalable, low-overhead, network topology-aware profiler be designed for IB clusters that is capable of depicting the communication pattern of high performance MPI applications?

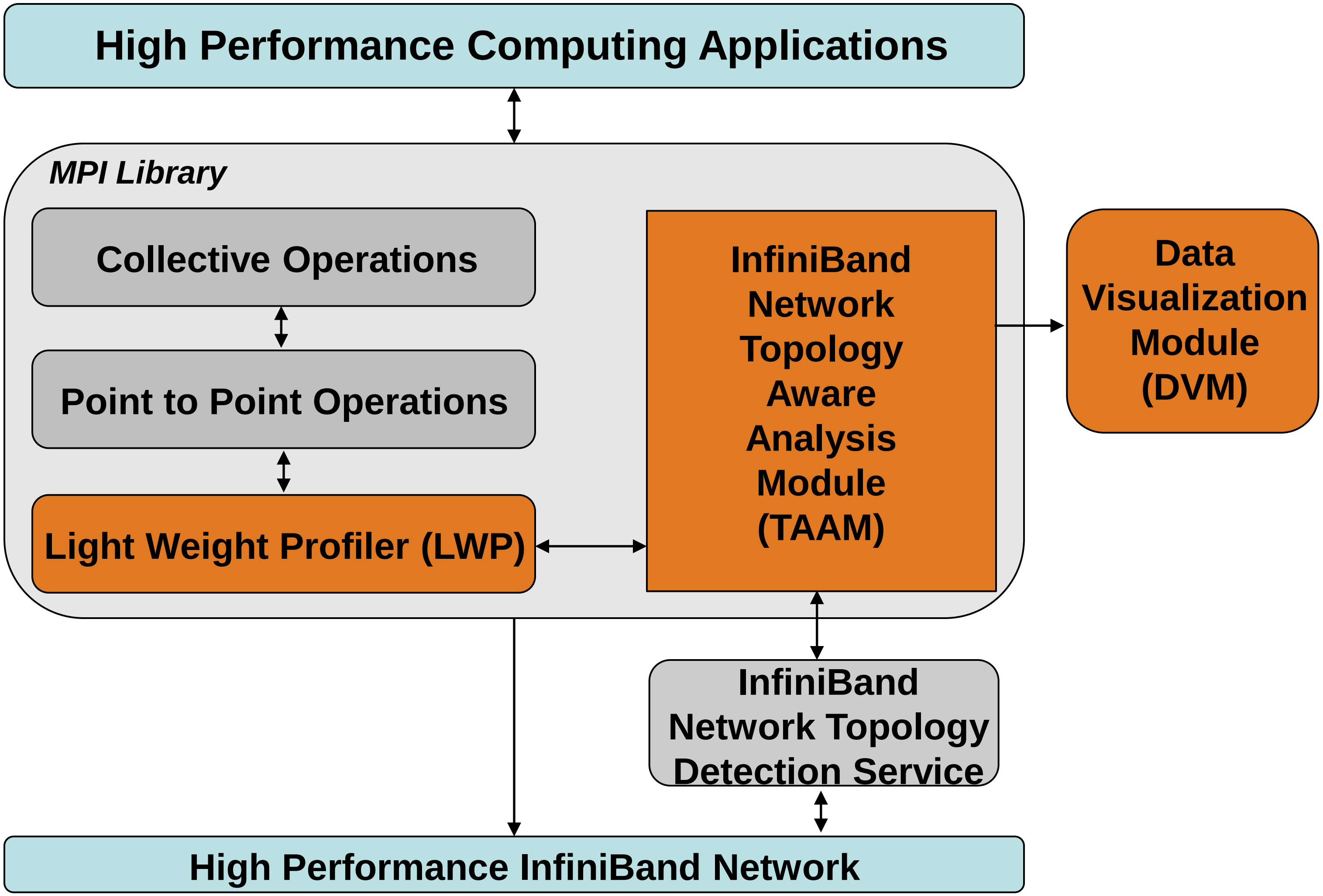
Envisioned Use Cases



Outline

- Introduction
- Problem Statement
- Design of Network Topology-Aware Performance Analysis Tool for MPI
- Performance Evaluation
- Conclusions and Future Work

Overall Framework of INTAP-MPI


High Performance Computing Applications
MPI Library
Collective Operations
Point to Point Operations
Light Weight Profiler (LWP)
InfiniBand Network Topology Aware Analysis Module (TAAM)
Data Visualization Module (DVM)
InfiniBand Network Topology Detection Service
High Performance InfiniBand Network

Topology Aware Analysis Module

- TAAM initiates and coordinates all profiling and analysis based on user input
  - Can be done for entire application through environment variables or
  - Designated parts of application by means of Unix signals
- On receiving user input, TAAM
  - Informs LWP to start logging intra/inter node communication in MPI library
    - Can request LWP to log either number or volume of messages
  - Queries the IB Network Topology Detection service and identifies process layout

Topology Aware Analysis Module

- Remains idle until application terminates or receives a signal from the user
- On application termination / receiving user signal
  - TAAM informs the LWP to stop logging and receives logged communication profile
  - TAAM in rank '0' gathers communication profile and classifies it based on the topology information
  - Classification possible at various granularities
    - Process, Compute node, Switch blade etc
- Passes classified communication profile to DVM

Data Visualization Module

- Visualizes the network topology-aware communication profile generated by TAAM
- Generates two kinds of graphs based on number of network hops
  - Stacked histogram showing the split up of physical communication
  - Heatmap depicting the relative volume of various types of message transfers
- Graphs can represent various granularities
  - Depends on user specified granularity passed on by TAAM

Light Weight Profiler

- Logs all communication in MPI library
- Integrated into lowest communication layers
  - Allows to capture actual communication behavior
    - Includes any fragmentation done by MPI for load balancing
- On receiving signal to start from TAAM, LWP
  - Allocates array dynamically – “On-Demand”
    - Initially allocates small number of locations
    - Size increases dynamically based on communication pattern
    - Worst case – O($N_{\textit{\textbf{processes}}}$) bytes to profile communication pattern between each pair of processes
- On receiving signal to stop, LWP transfers logged information to TAAM on local process
- Frees allocated memory

Outline

- Introduction
- Problem Statement
- Design of Network Topology-Aware Performance Analysis Tool for MPI
- Performance Evaluation
- Conclusions and Future Work

Experimental Testbed

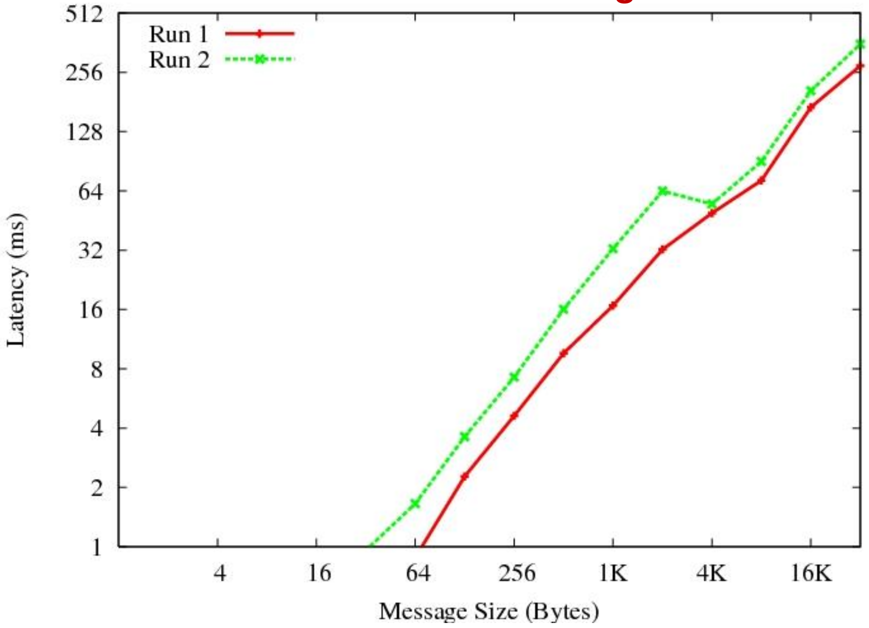
- Compute platforms
  - Ranger
    - 3,936 16-way SMP compute nodes (62,976 cores)
    - 2.3 GHz Opteron cores with 32 GB per node
    - Two 3,456 port SDR Sun IB Datacenter switches
      - 7-stage, full-CLOS FAT tree
  - Hyperion
    - 1,400 Intel Xeon 5640 cores
    - 2.53 GHz Nehalem cores with 8GB per node, 12 MB L3 cache
    - 171-port Mellanox QDR switch
      - 11 leafs, each having 16 ports, partial FAT tree
- MPI Library – MVAPICH2-1.8

MVAPICH/MVAPICH2 Software

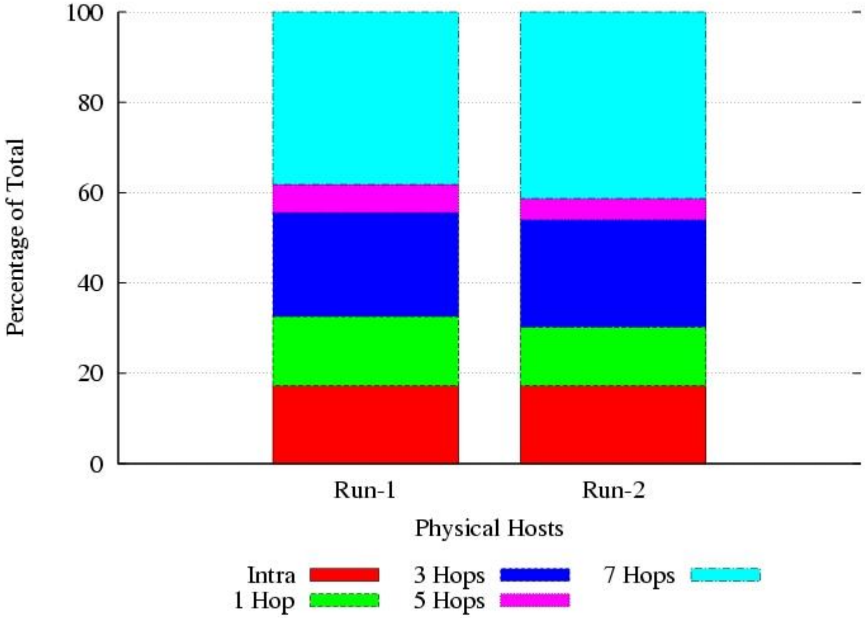
- High Performance open-source MPI Library for InfiniBand, 10Gig/iWARP and RDMA over Converged Enhanced Ethernet (RoCE)
  - MVAPICH (MPI-1) and MVAPICH2 (MPI-2.2), Available since 2002
  - Used by more than 1,930 organizations (HPC Centers, Industry and Universities) in 68 countries
  - More than 124,000 downloads from OSU site directly
  - Empowering many TOP500 clusters
    - 11th ranked 125,980-core cluster (Pleiades) at NASA
    - 14th ranked 73,278-core cluster (Tsubame 2.0) at Tokyo Institute of Technology
    - 40h ranked 62,976-core cluster (Ranger) at TACC
    - and many others
  - Available with software stacks of many IB, HSE and server vendors including Linux Distros (RedHat and SuSE)
  - http://mvapich.cse.ohio-state.edu
- Partner in the upcoming U.S. NSF-TACC Stampede (10-15 PFlop) System

Visualizing Network Characteristics of Collectives (MPI_Alltoall)

Performance comparison of 256 process MPI_Alltoall on Ranger



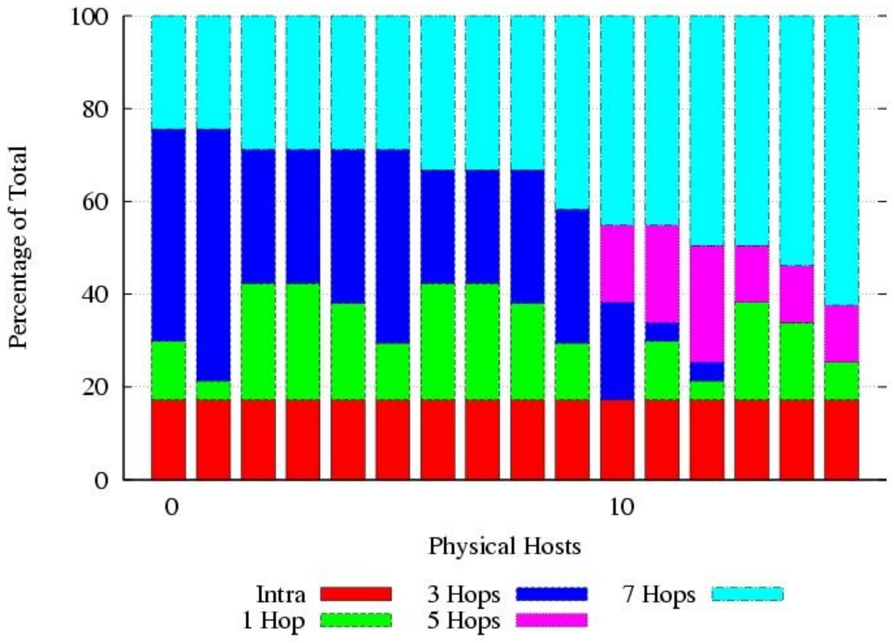
Summary of network level communication



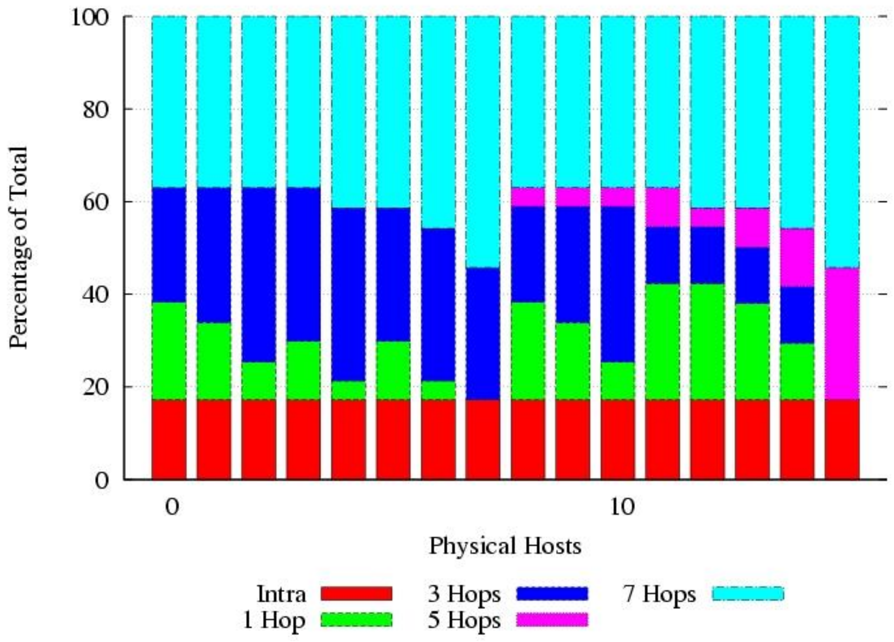
- Run #1 performs better than Run #2
- Run #2 has more 7 hop and less 5, 3, 1 hop communication compared to Run#1

Visualizing Network Characteristics of Collectives (MPI_Alltoall)

Bar graph depicting communication characteristics of 256 process MPI_Alltoall at node level



Run #1

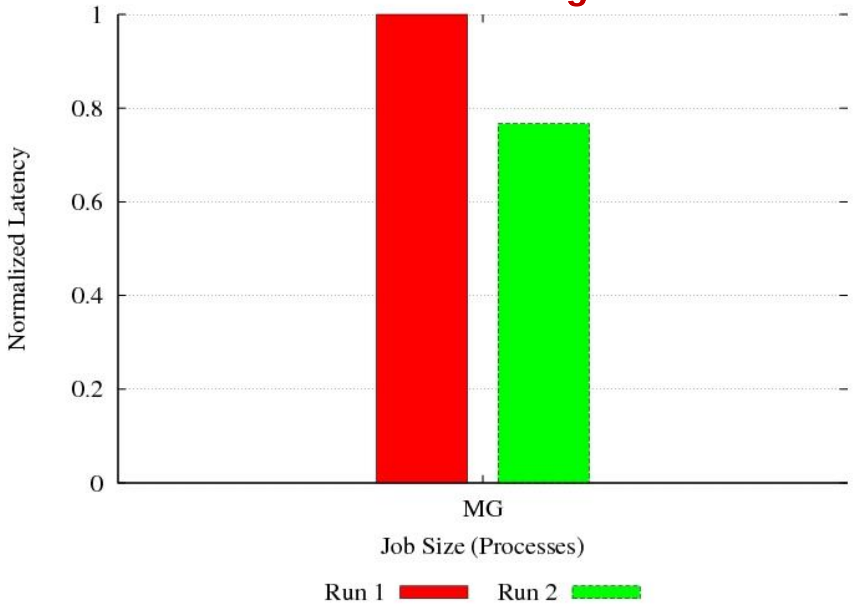


Run #2

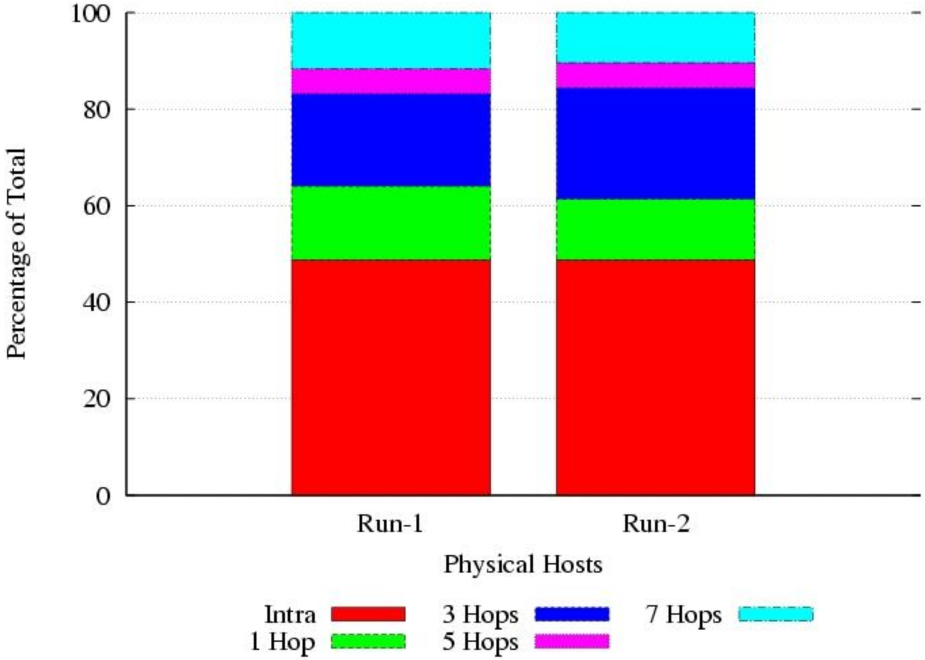
- Capable of depicting communication pattern at finer granularity
- Enable scheduler developers to create better allocation policies

Visualizing Network Characteristics of Point-to-point Communication

Performance comparison of 256 process NAS MG on Ranger



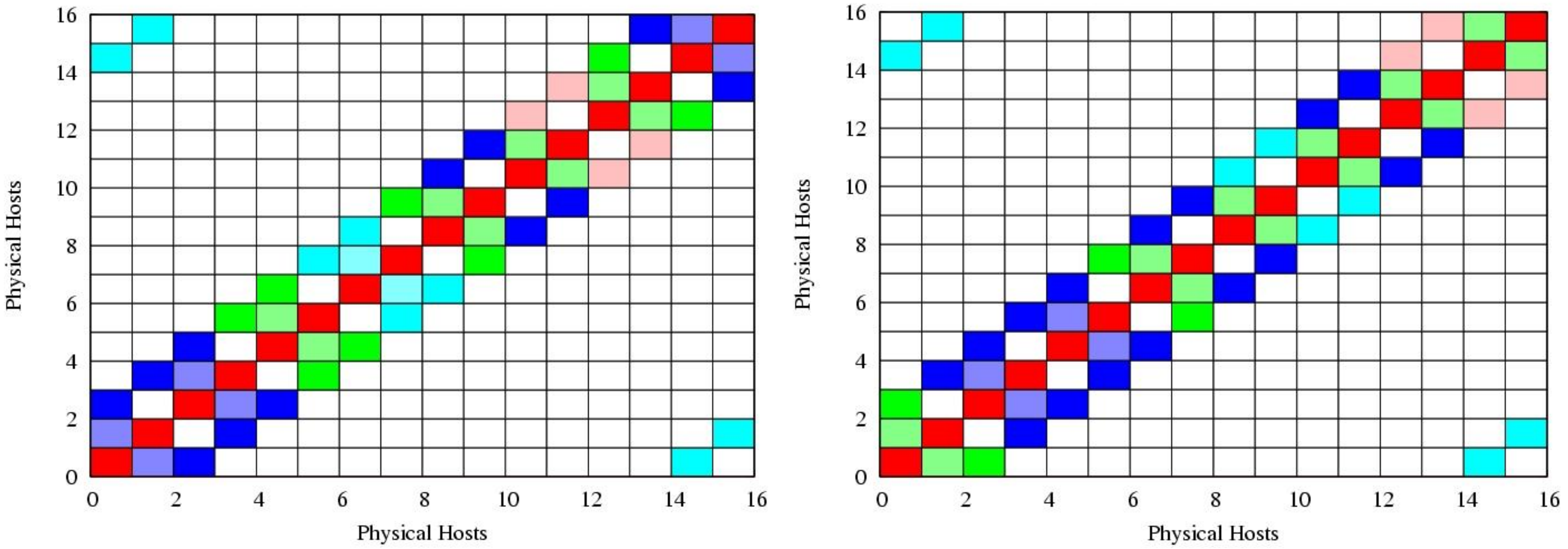
Summary of network level communication



- NAS MG does mostly Point-to-point communication
- Run #2 performs better than Run #1
- Run #1 has more 7-hop communication compared to Run#2

Visualizing Network Characteristics of Point-to-point Communication

Heatmap depicting communication characteristics of 256 process NAS MG at node level



Run #1

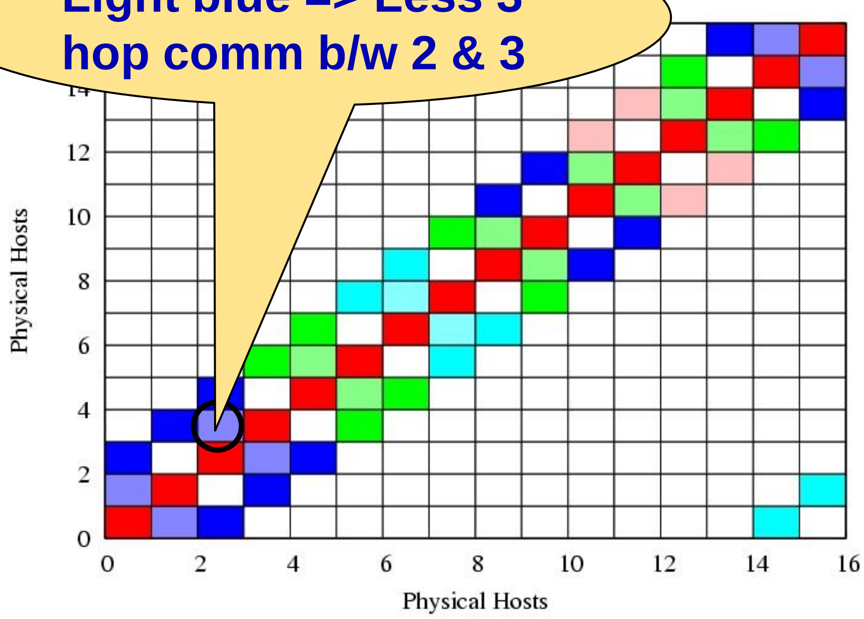
Run #2

- Intensity of colors represent relative communication volumes
  - Greater intensity = More communication

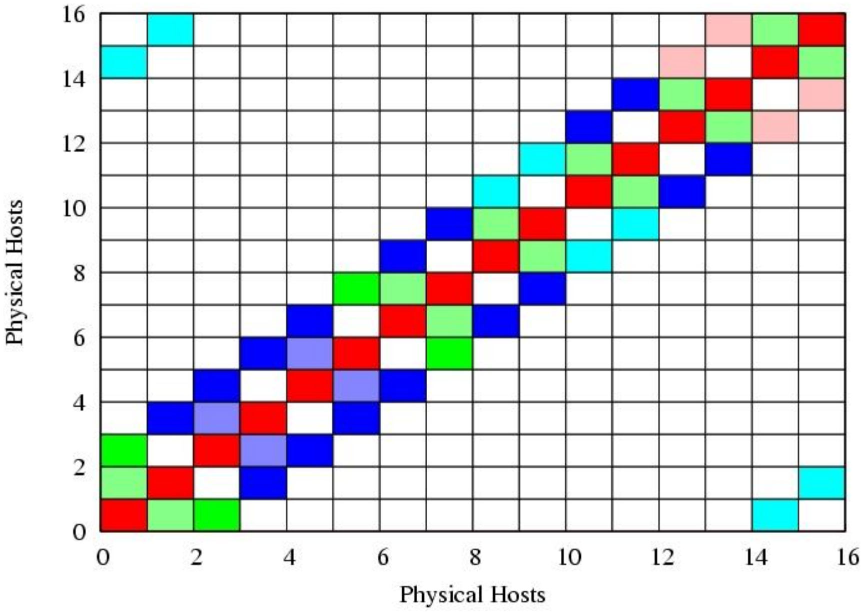
Visualizing Network Characteristics of Point-to-point Communication

unication characteristics of 256 process NAS MG at node level

Light blue => Less 3 hop comm b/w 2 & 3



Run #1

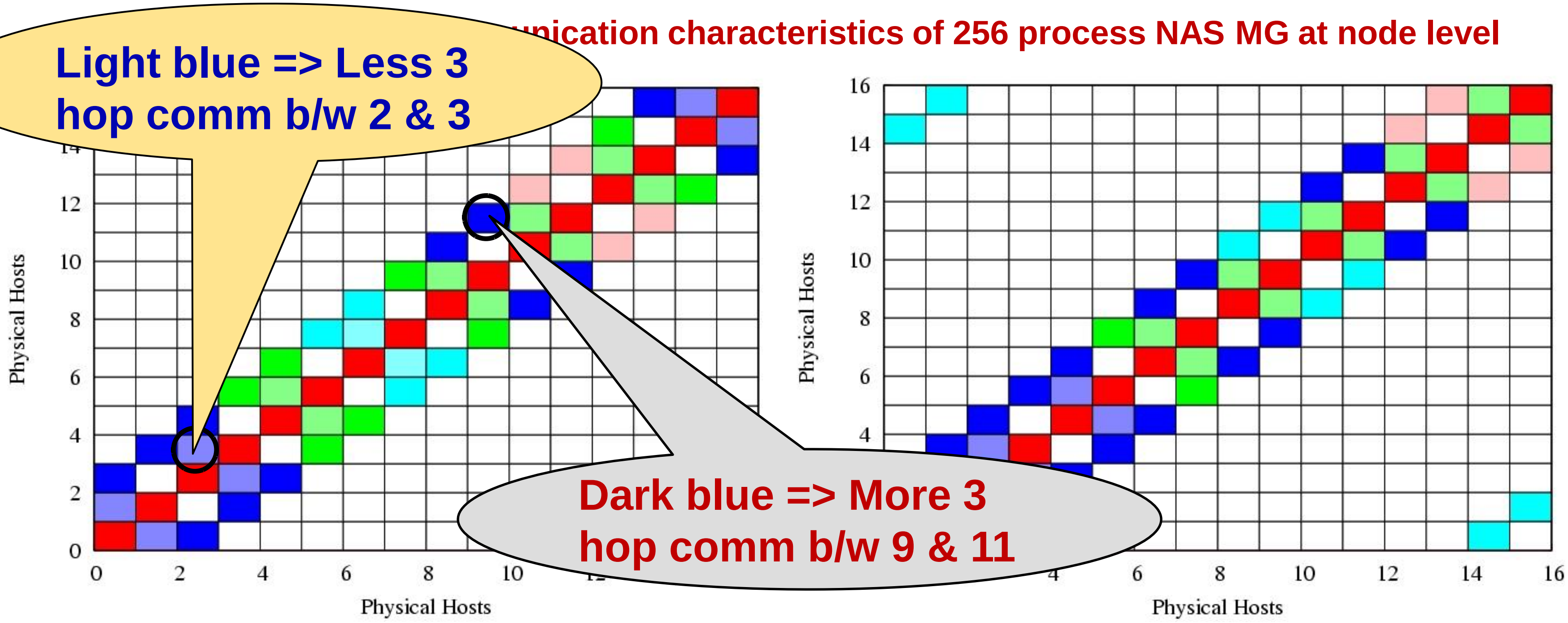


Run #2

- Intensity of colors represent relative communication volumes
  - Greater intensity = More communication

Visualizing Network Characteristics of Point-to-point Communication

unication characteristics of 256 process NAS MG at node level



Light blue => Less 3 hop comm b/w 2 & 3

Dark blue => More 3 hop comm b/w 9 & 11

Run #1

Run #2

- Intensity of colors represent relative communication volumes
  - Greater intensity = More communication
- Can be used by MPI developers for topo-aware communication

Impact of INTAP-MPI on Memory Consumption

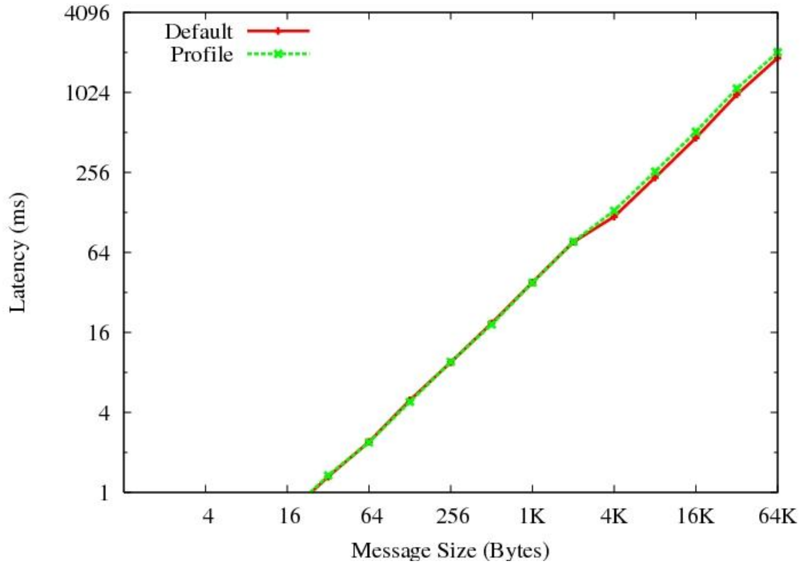
Maximum memory consumption for profiling MPI_Alltoall at process level granularity

Job Size (# Processes)	**64**	**128**	**256**	**512**	**1,024**
Memory Overhead (MB)	0.04	0.16	0.58	2.19	8.61

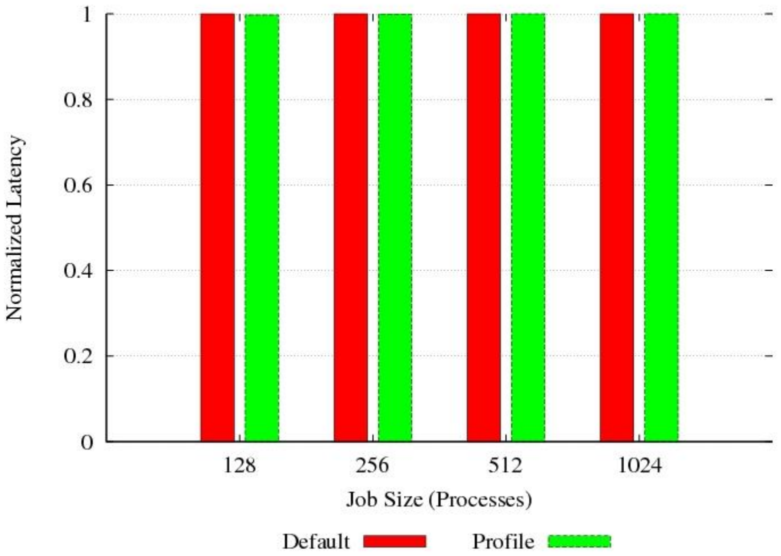
- Memory at each rank allocated dynamically based on need
  - Worst case – O($N_{\textit{processes}}$) bytes
  - Only rank ‘0’ needs to allocate large memory data structures
- These values hold for any application irrespective of the amount of memory consumed by the application

Impact of INTAP-MPI on Performance

Impact of INTAP-MPI on performance of MPI_Alltoall on Hyperion



Performance for 1,024 processes



Performance for 64 KB message size

- INTAP-MPI has very little impact on performance of profiled application

Outline

- Introduction
- Problem Statement
- Design of Network Topology-Aware Performance Analysis Tool for MPI
- Performance Evaluation
- Conclusions and Future Work

Conclusions & Future Work

- Designed and developed INTAP-MPI
  - Network topology-aware, scalable, low-overhead MPI profiler
  - Gives the flexibility to profile entire MPI applications or specific sections of the application
  - Integrated into the MVAPICH2 MPI library
- Able to profile and visualize the communication pattern of applications with very low memory and performance overhead at scale
- In future, we would like to extend this work to make it easily usable for other MPI libraries and software stacks

Thank you!

{subramon, viennej, panda}@cse.ohio-state.edu

Network-Based Computing Laboratory

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

Network Based Computing Laboratory

MVAPICH

Thank you!