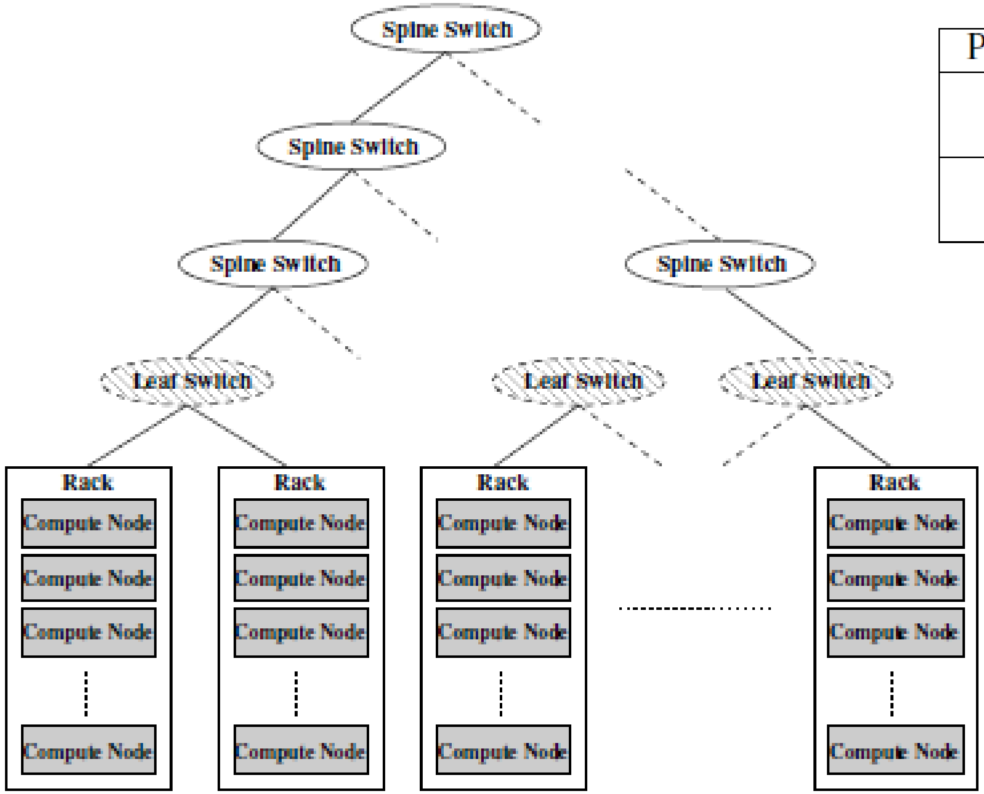
Topology-aware Collectives

Mar 12, 2021

Designing Topology-Aware Collective Communication Algorithms for Large Scale InfiniBand Clusters: Case Studies with Scatter and Gather

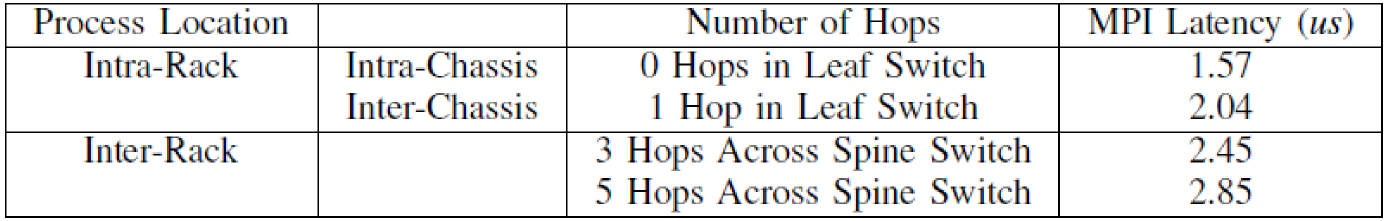
Krishna Kandalla, Hari Subramoni, Abhinav Vishnu and Dhabaleswar K. (DK) Panda

Effect of Topology on Latency



A typical topology of large-scale systems

(TACC Ranger system)



Process Location		Number of Hops	MPI Latency (us)
Intra-Rack	Intra-Chassis	0 Hops in Leaf Switch	1.57
	Inter-Chassis	1 Hop in Leaf Switch	2.04
Inter-Rack		3 Hops Across Spine Switch	2.45
		5 Hops Across Spine Switch	2.85

Topology-aware Collective algorithms

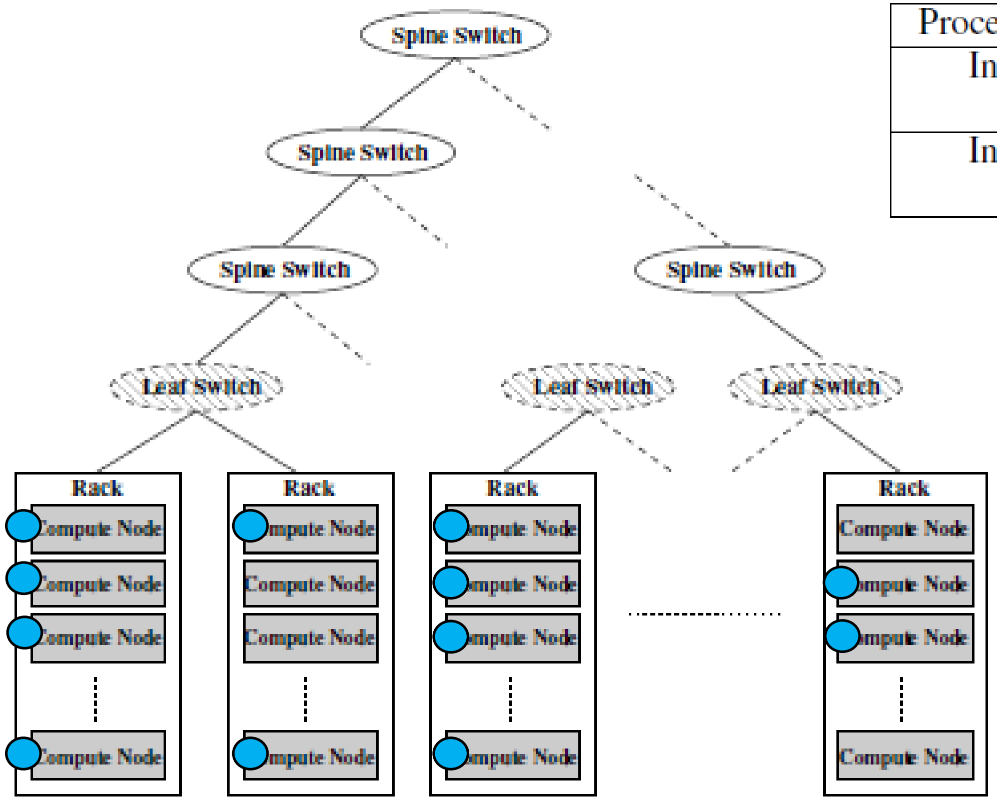
- Detect the topology of large-scale Infiniband clusters
- Topology-aware Gather and Scatter
- Modified communication model
- 54% improvement on micro-benchmarks

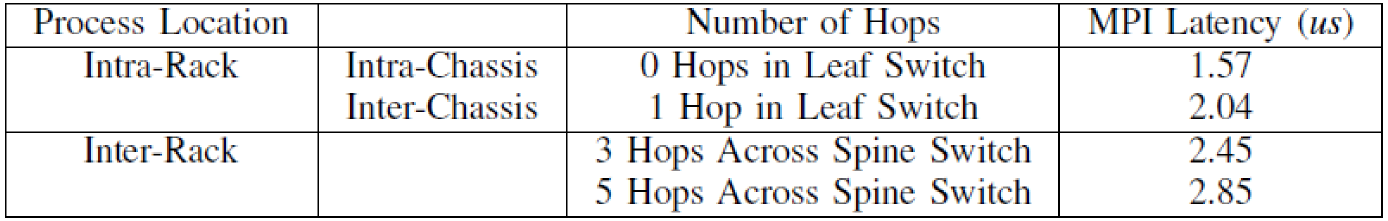
Discover Topology

- Infiniband tools
  - ibnetdiscover – outputs the switch connections / identifiers
  - One-time discovery (in general)

- MPI_Init
  - Create intra-chassis communicators – all nodes in the same chassis
  - Create intra-switch communicators – all nodes in the same leaf switch
  - Assign one chassis-leader and one switch-leader
  - Create switch-leader and chassis-leader communicators

Example of Sub-communicators

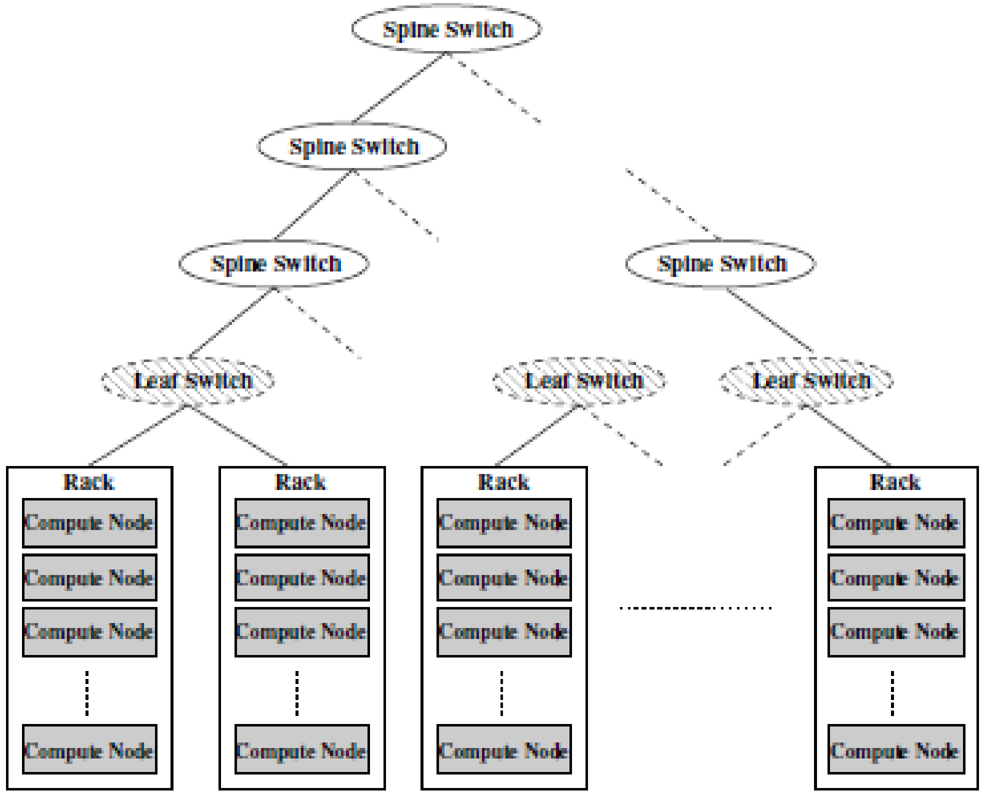




Process Location		Number of Hops	MPI Latency (*us*)
Intra-Rack	Intra-Chassis	0 Hops in Leaf Switch	1.57
	Inter-Chassis	1 Hop in Leaf Switch	2.04
Inter-Rack		3 Hops Across Spine Switch	2.45
		5 Hops Across Spine Switch	2.85

- intra-chassis communicators
- intra-switch communicators
- chassis-leader communicators
- switch-leader communicators

Cost of Communication



Cost involved for communication within the same node
L: t_s-intra-node
B: t_w-intra-node

Cost of communication within the same leaf switch
L: t_s-intra-switch
B: t_w-intra-switch

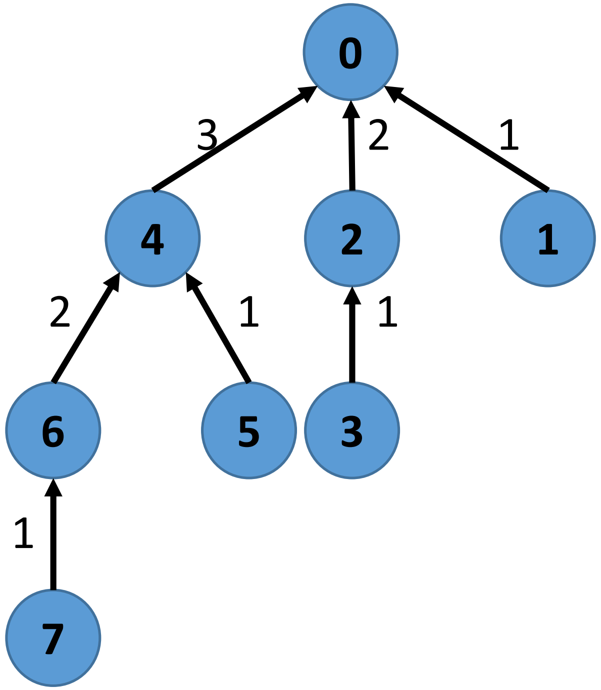
Cost involved for an inter-switch communication
L: t_s-inter-switch
B: t_w-inter-switch

t_s-intra-node < t_s-intra-switch < t_s-inter-switch
t_w-intra-node < t_w-intra-switch < t_w-inter-switch

1. Actual cost depends on the #hops based on the actual placement of processes
2. Contention for intra-node/switch << inter-switch

Cost Model (Gather)

Number of racks = R
Number of processes = P
Message size = N



Number of exchanges at i^{th} level: C_i
C_1 = Number of intra-node transfers
C_2 = Number of intra-switch transfers
C_3 = Number of inter-switch transfers

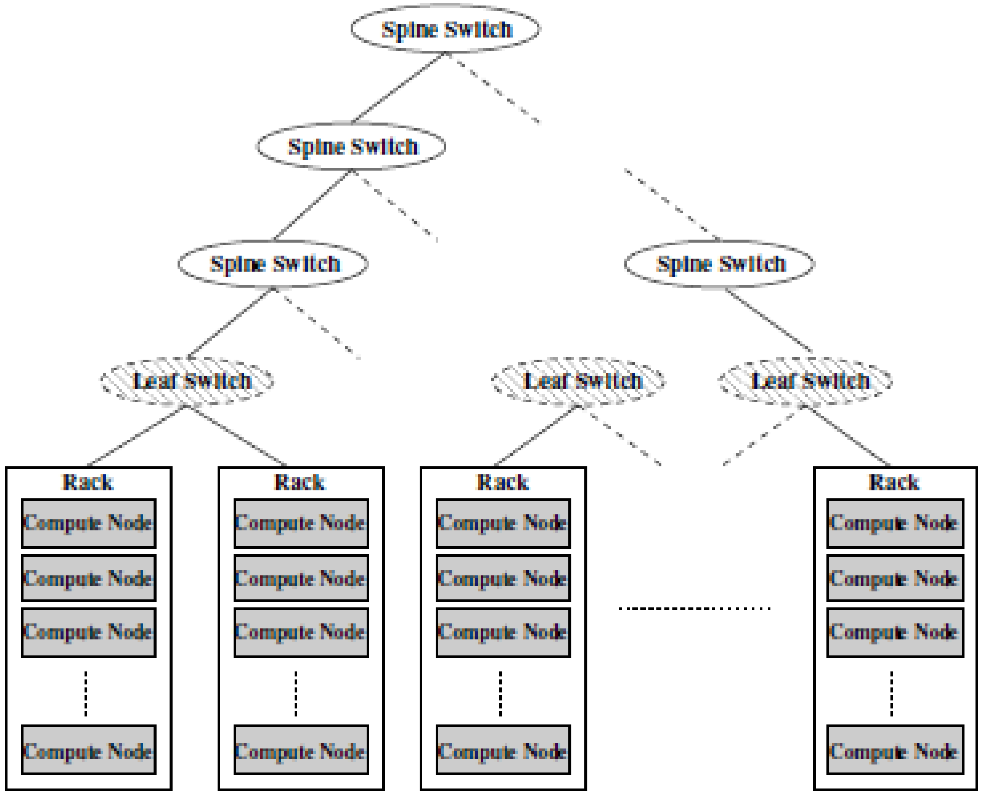
Cost of data transfer at each level: γ, β, δ
Switch-level contention: α

$$\begin{aligned} T_{binomial} =& (t_s\text{-}inter\text{-}node * C_1 + t_s\text{-}intra\text{-}switch * C_2 \\ & + \alpha * t_s\text{-}inter\text{-}switch * C_3) + t_w\text{-}intra\text{-}node \\ & * (C_1) * (N * \gamma) + t_w\text{-}intra\text{-}switch \\ & * (C_2) * (N * \beta) + \alpha * t_w\text{-}inter\text{-}switch \\ & * (C_3) * (N * \delta) \end{aligned}$$

Communication Cost for Gather (Binomial)

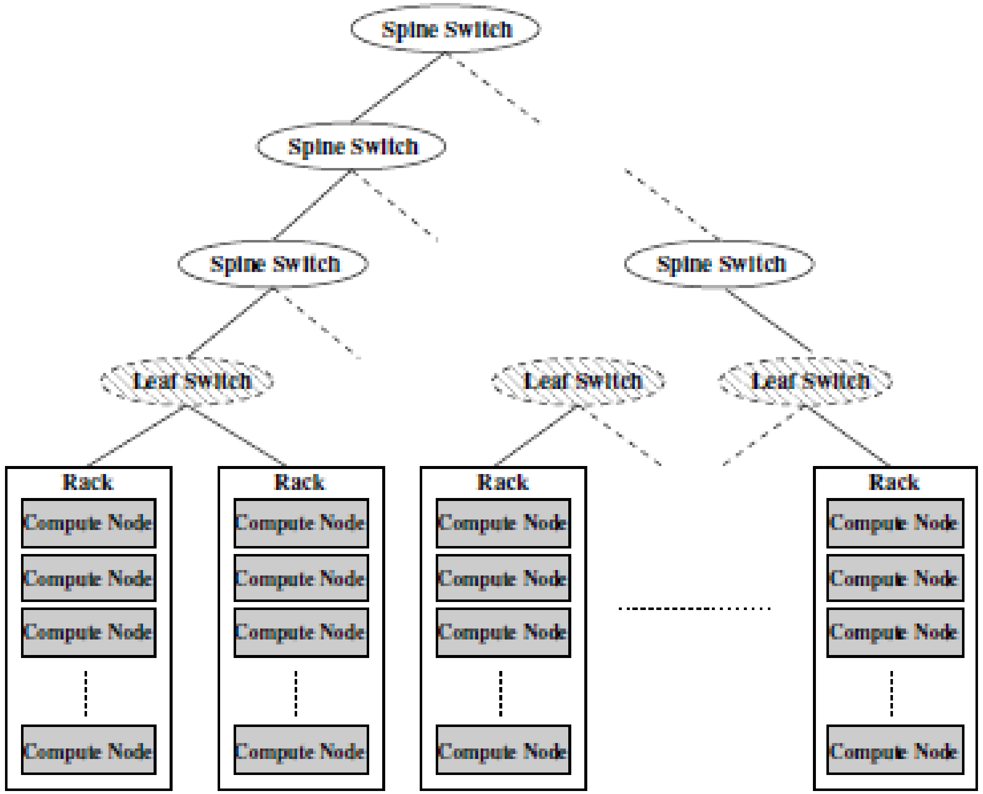
- Cost of data transfer at each level: γ, β, δ
- The bandwidth term is obtained by adding costs at each level
- $C_1 * \gamma + C_2 * \beta + C_3 * \delta = (p-1)/p$

$$T_{binomial} = (t_s\text{-}inter\text{-}node * C_1 + t_s\text{-}intra\text{-}switch * C_2 + \alpha * t_s\text{-}inter\text{-}switch * C_3) + t_w\text{-}intra\text{-}node * (C_1) * (N * \gamma) + t_w\text{-}intra\text{-}switch * (C_2) * (N * \beta) + \alpha * t_w\text{-}inter\text{-}switch * (C_3) * (N * \delta)$$



Topology-aware Gather

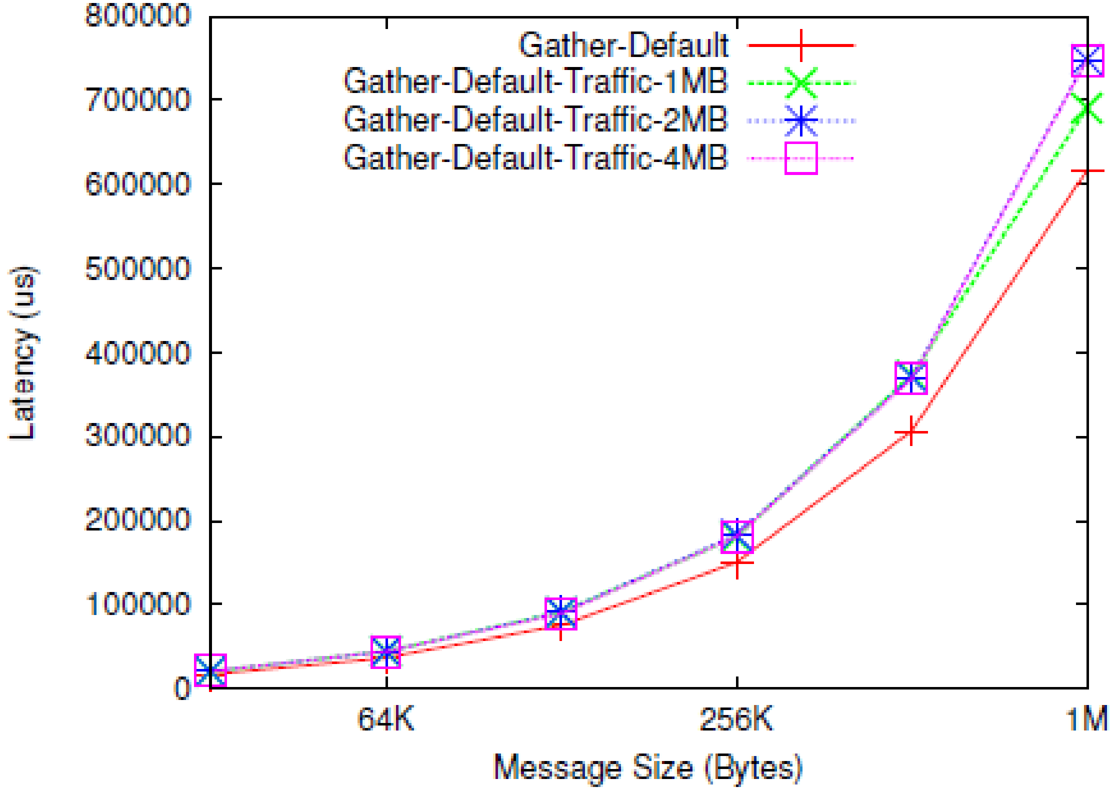
- Rack-leader processes independently perform intra-switch gather
- R rack leaders perform inter-switch gather
- Reduced L and B terms (due to reduction in inter-switch exchanges)



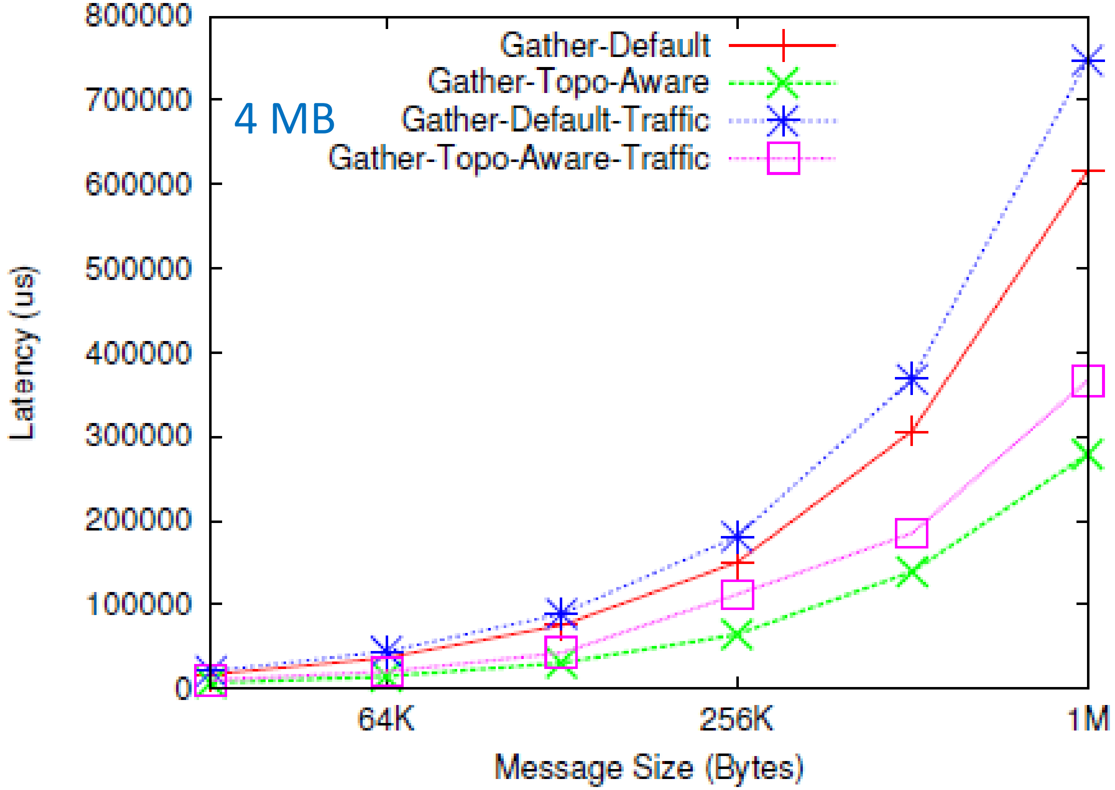
Experimental Setup

- A simple benchmark code that iterates through various message sizes (0 – 1 MB) and invokes a collective call several times in a loop.
- AlltoAll is used to create background traffic

Gather Results (With and Without Traffic)

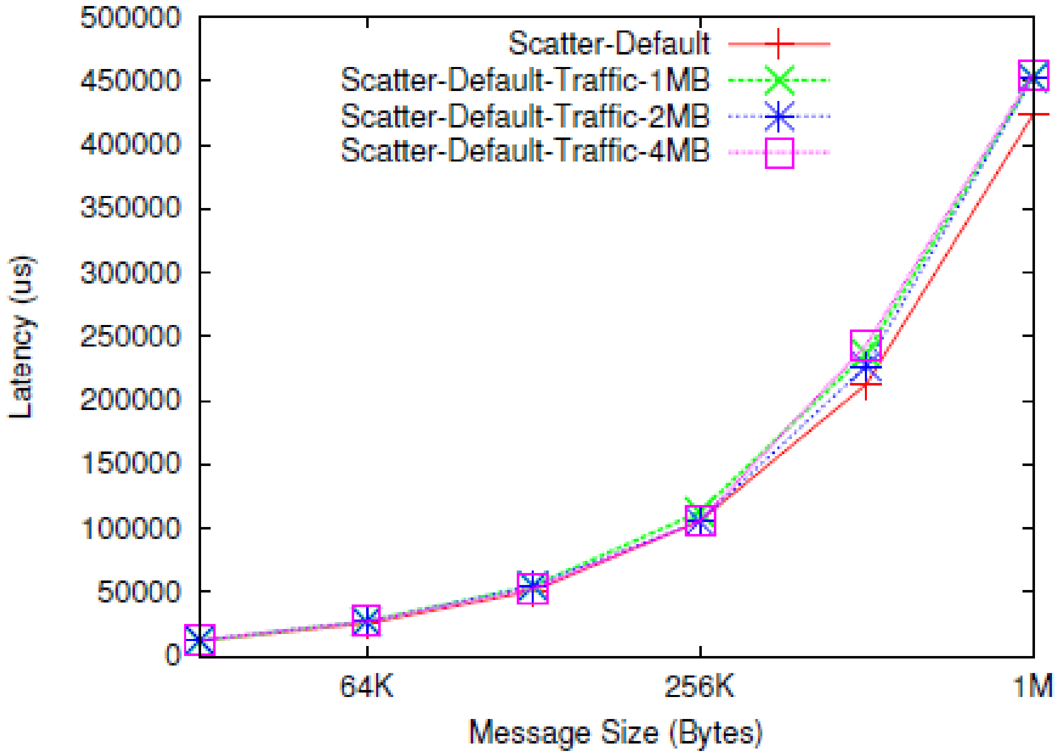


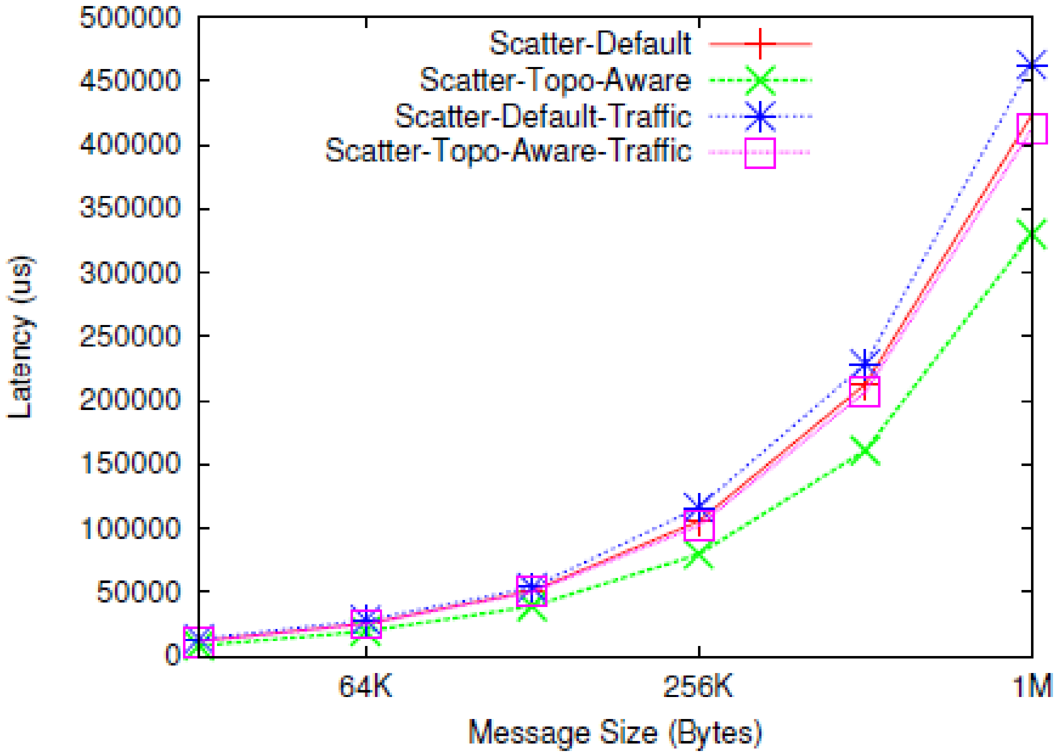
21% drop



4 MB

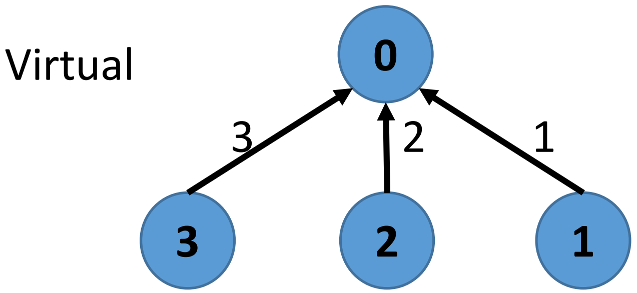
Scatter Results

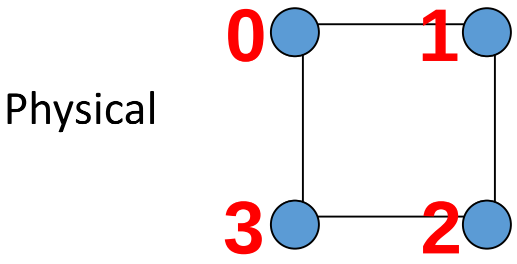


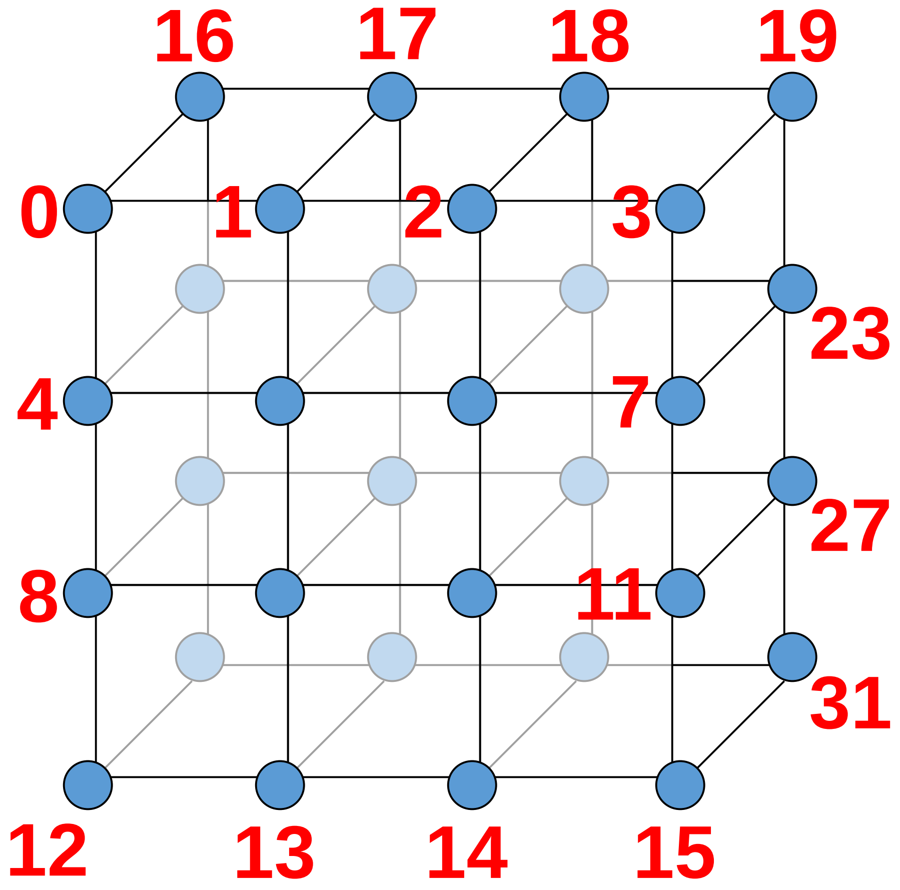


Process Mapping

Virtual-to-physical Mapping

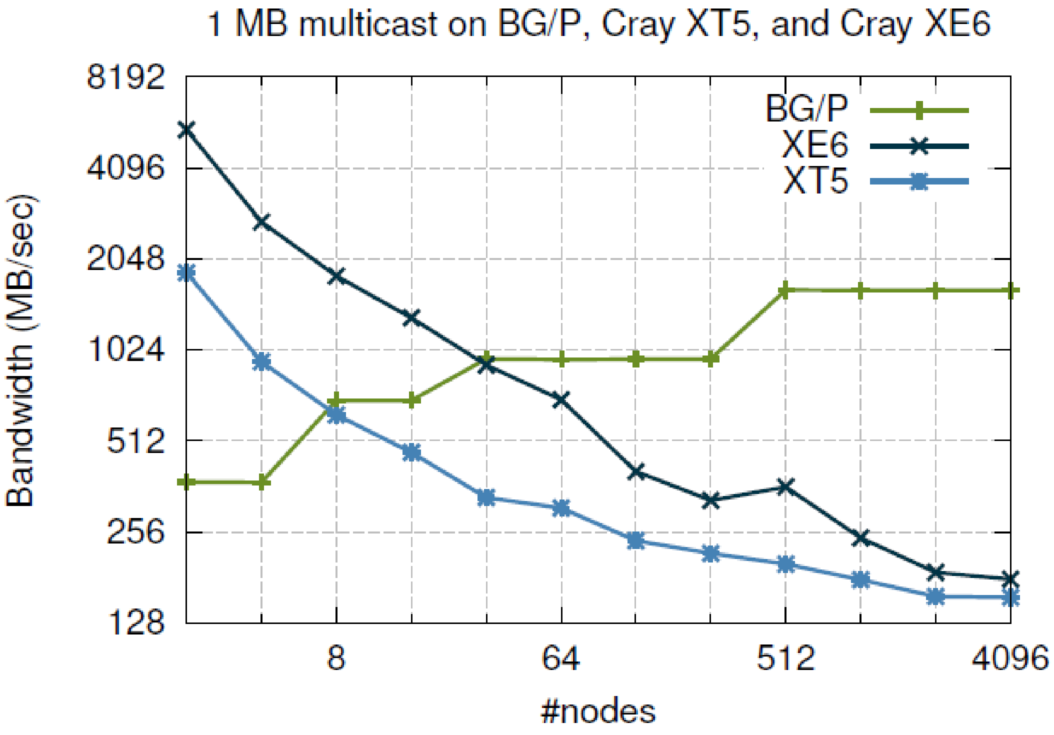






4 x 4 x 2 3D torus

Performance of Multicasts



Improving communication performance in dense linear algebra via topology aware collectives, SC11

- Multicast: Broadcast to a subset of nodes
- Bandwidth of a 1 MB multicast drops by 30x on Cray XE6 (Hopper)
- Bandwidth grows by a factor of 4.3x on the Intrepid Blue Gene/P (BG/P)
- How does the bandwidth improve?
- BG/P has 3D torus, proprietary rectangular algorithms to saturate links simultaneously
- Cray uses the binomial algorithm (Hopper also has 3D torus)