Lecture 13 (part 1)
Thread Level Parallelism (6)

EEC 171 Parallel Architectures

John Owens
UC Davis

Credits

- © John Owens / UC Davis 2007–8.
- Thanks to many sources for slide material: Computer Organization and Design (Patterson & Hennessy) © 2005, Computer Architecture (Hennessy & Patterson) © 2007, Inside the Machine (Jon Stokes) © 2007, © Dan Connors / University of Colorado 2007, © Kathy Yelick / UCB 2007, © Wen-Mei Hwu/David Kirk, University of Illinois 2007, © David Patterson / UCB 2003–7, © John Lazzaro / UCB 2006, © Mary Jane Irwin / Penn State 2005, © John Kubiatowicz / UCB 2002, © Krste Asinovic/Arvind / MIT 2002, © Morgan Kaufmann Publishers 1998.

Outline

- Interconnection Networks
- Grab bag:
  - Amdahl's Law
  - Novices & Parallel Programming
  - Interconnect Technologies

Network Topology

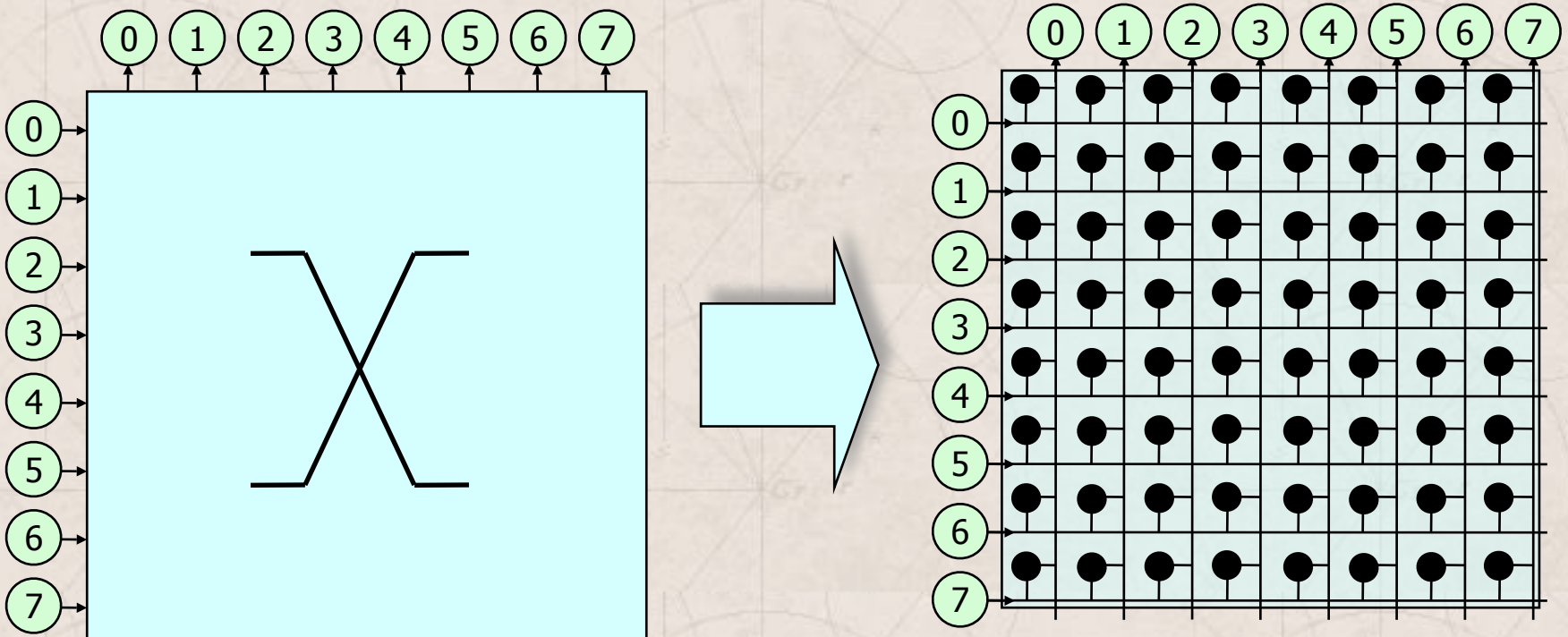
Preliminaries and Evolution

- One switch suffices to connect a small number of devices
  - Number of switch ports limited by VLSI technology, power consumption, packaging, and other such *cost* constraints
- A *fabric* of interconnected switches (i.e., *switch fabric* or *network fabric*) is needed when the number of devices is much larger
  - The *topology* must make a path(s) available for every pair of devices—property of *connectedness* or *full access* (*What paths?)*
- Topology defines the connection structure across all components
  - *Bisection bandwidth*: the *minimum* bandwidth of all links crossing a network split into two roughly equal halves
  - *Full bisection bandwidth*:
    - Network $BW_{Bisection}$ = Injection (or Reception) $BW_{Bisection} = N/2$
  - Bisection bandwidth mainly affects *performance*
- Topology is constrained primarily by local chip/board *pin-outs*; secondarily, (if at all) by global bisection bandwidth

Network Topology

Centralized Switched (Indirect) Networks

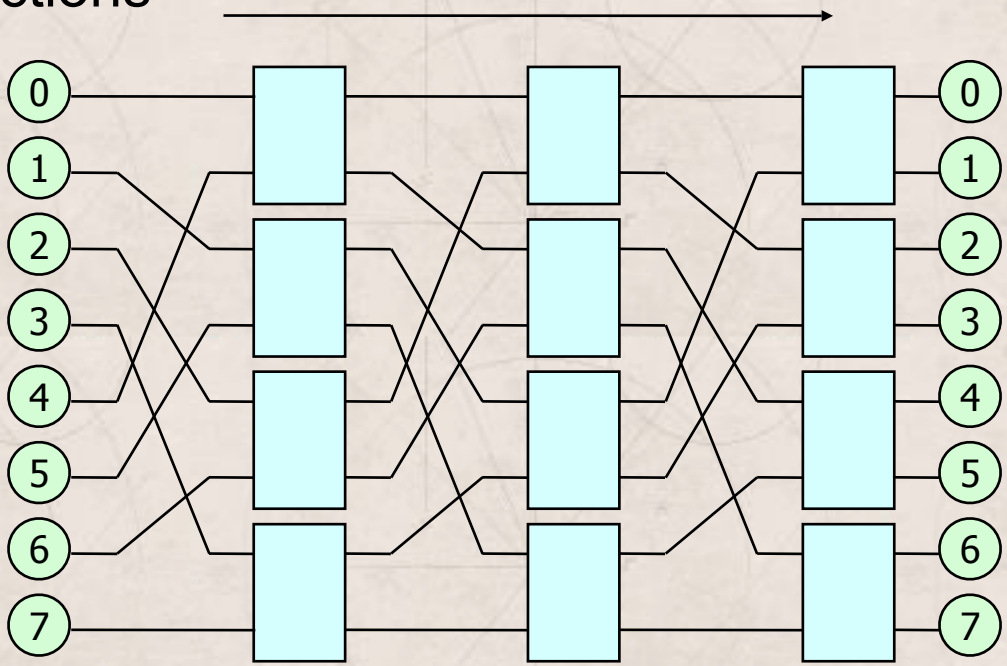
- ***Crossbar network***
  - Crosspoint switch complexity increases quadratically with the number of crossbar input/output ports, *N*, i.e., grows as $O(N^2)$
  - Has the property of being *non-blocking*



Network Topology

Centralized Switched (Indirect) Networks

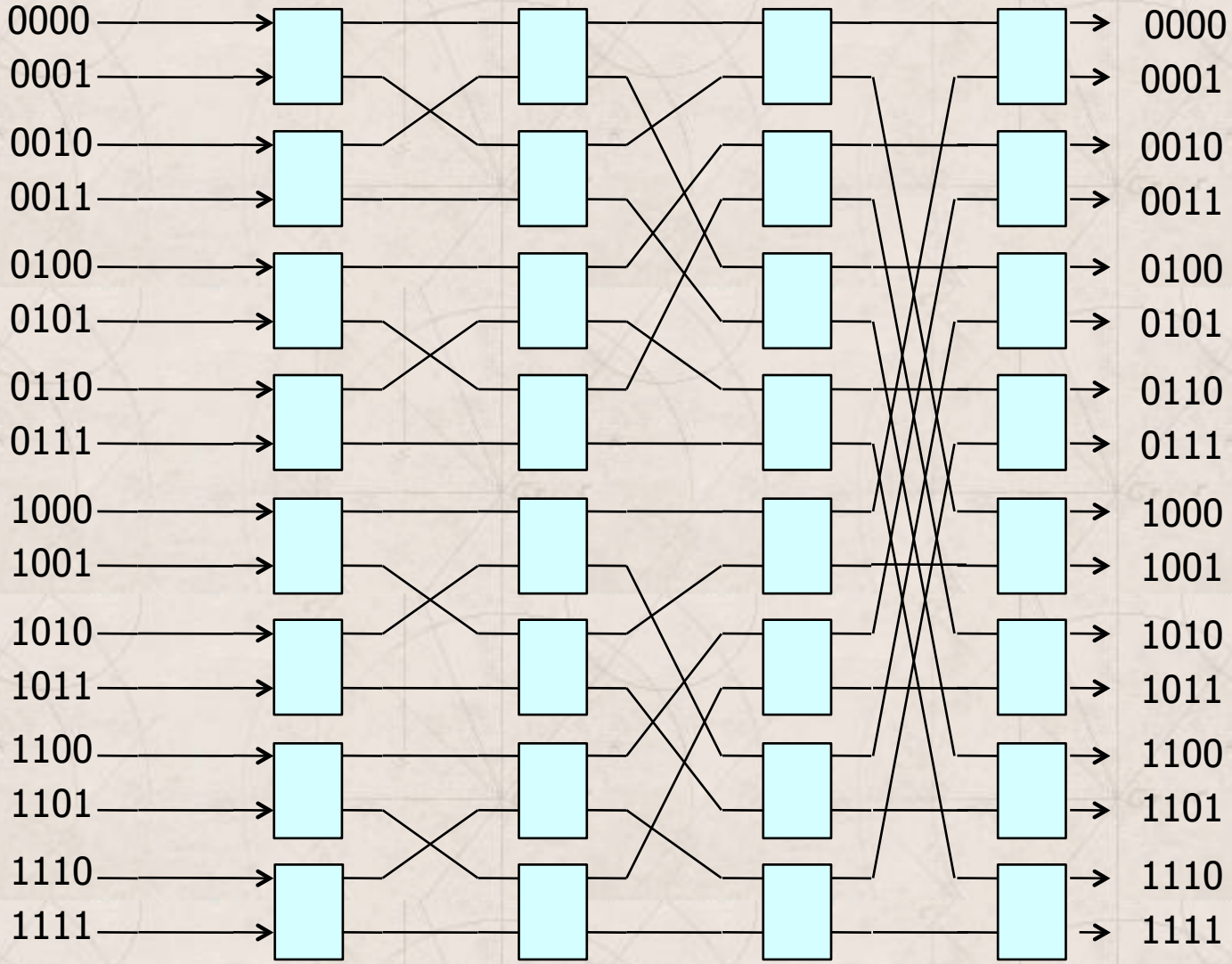
- ***Multistage interconnection networks (MINs)***
  - Crossbar split into several stages consisting of smaller crossbars
  - Complexity grows as $O(N \times \log N)$, where N is # of end nodes
  - Inter-stage connections represented by a set of permutation functions



Omega topology, perfect-shuffle exchange

Network Topology

Centralized Switched (Indirect) Networks

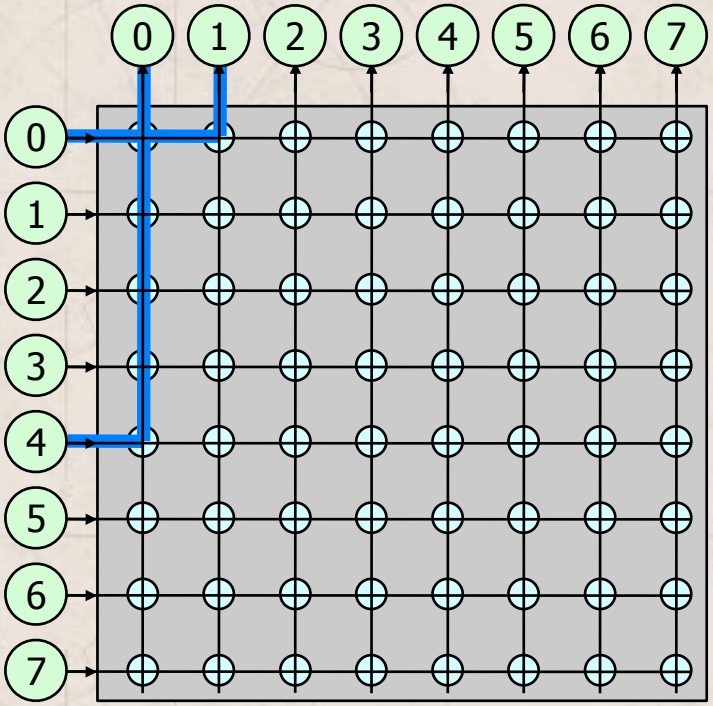


16 port, *4 stage **Butterfly** network*

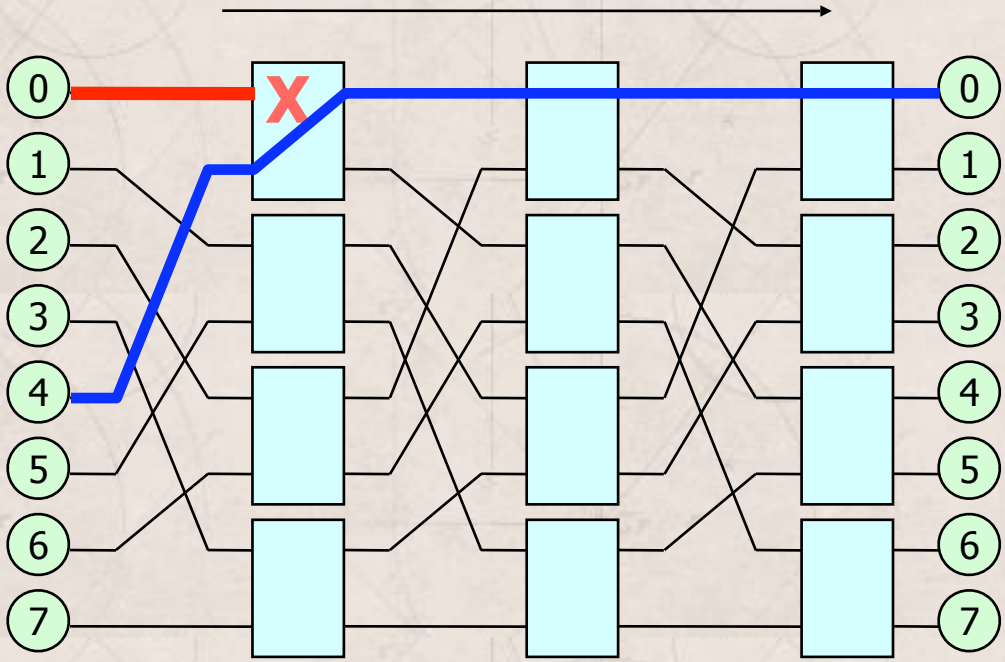
Network Topology

Centralized Switched (Indirect) Networks

- Reduction in MIN switch *cost* comes at the price of *performance*
  - Network has the property of being *blocking*
  - *Contention* is more likely to occur on network links
    - Paths from different sources to different destinations share one or more links



non-blocking topology



blocking topology

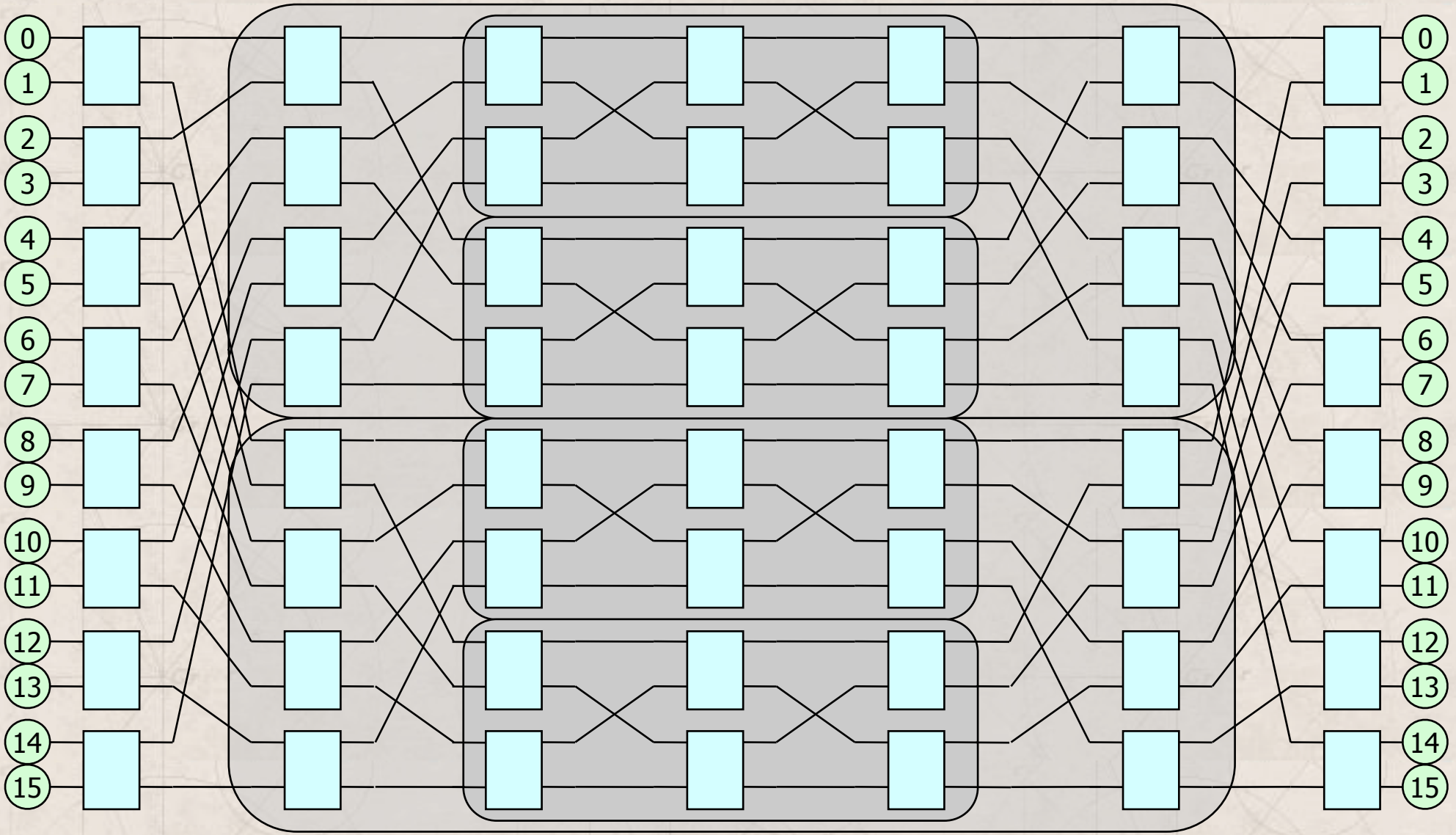
Network Topology

Centralized Switched (Indirect) Networks

- How to reduce blocking in MINs? *Provide alternative paths!*
  - Use larger switches (can equate to using more switches)
    - ***Clos network***: minimally three stages (non-blocking)
      - A larger switch in the middle of two other switch stages provides enough alternative paths to avoid all conflicts
  - Use more switches
    - Add $\log_k N$ - 1 stages, mirroring the original topology
      - *Rearrangeably non-blocking*
      - Allows for non-conflicting paths
      - Doubles network *hop count (distance), d*
      - Centralized control can rearrange established paths
    - ***Benes topology***: $2(\log_2 N)$ - 1 stages (rearrangeably non-blocking)
      - Recursively applies the three-stage Clos network concept to the middle-stage set of switches to reduce all switches to 2 x 2

Network Topology

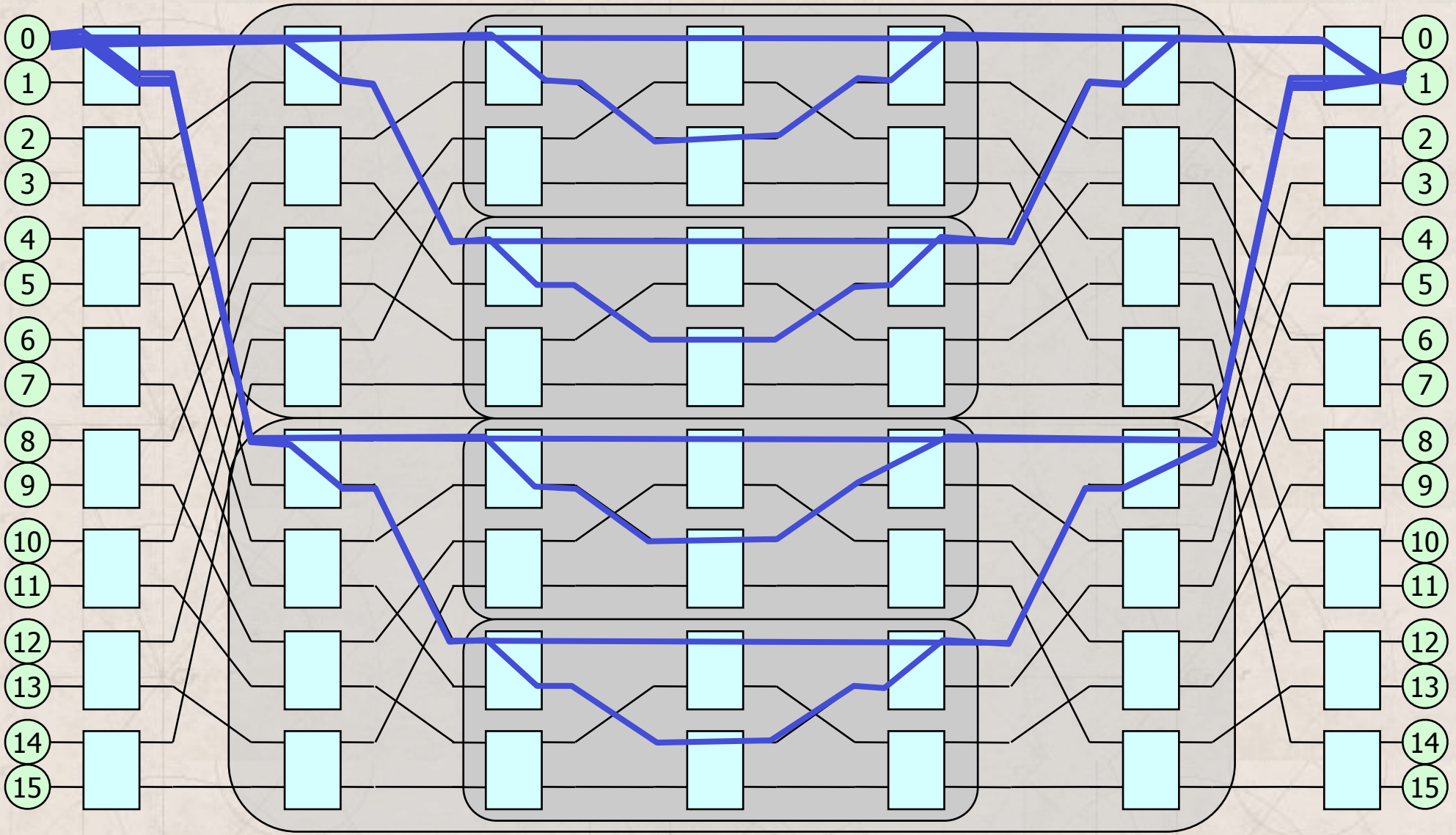
Centralized Switched (Indirect) Networks



16 port, ***7 stage Clos*** network = ***Benes topology***

Network Topology

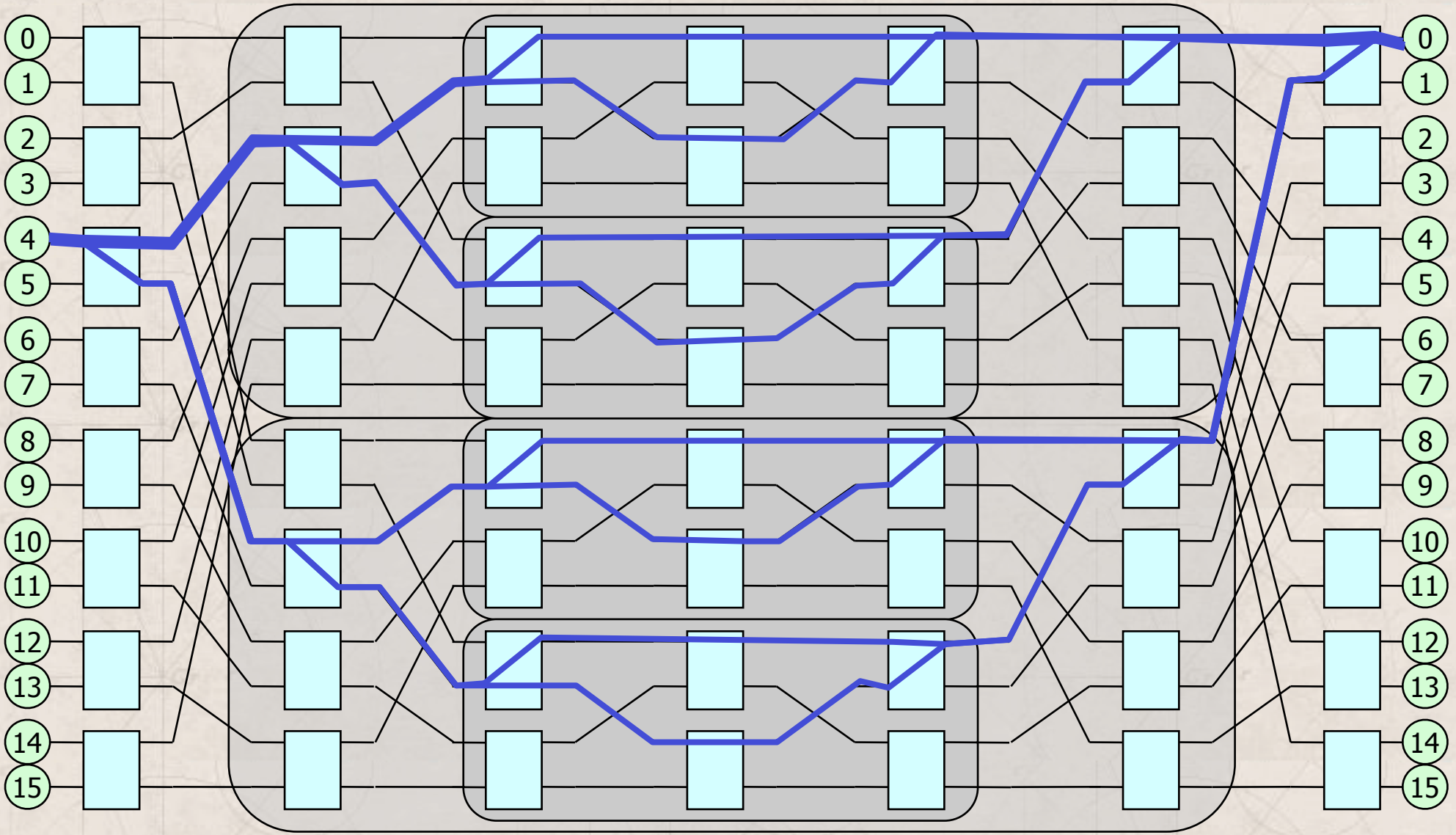
Centralized Switched (Indirect) Networks



Alternative paths from 0 to 1. 16 port, *7 stage Clos* network = *Benes topology*

Network Topology

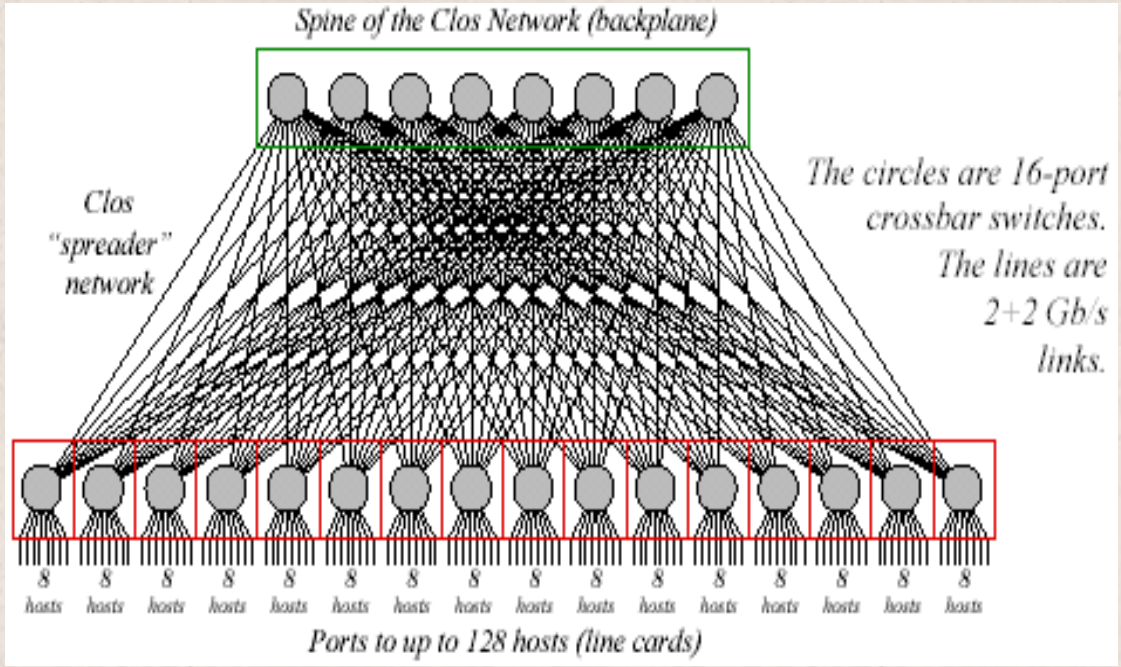
Centralized Switched (Indirect) Networks

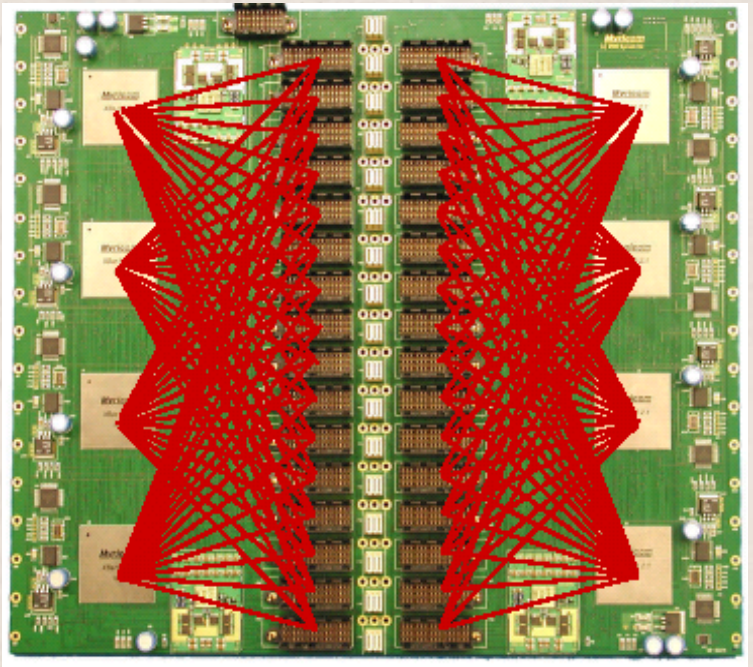


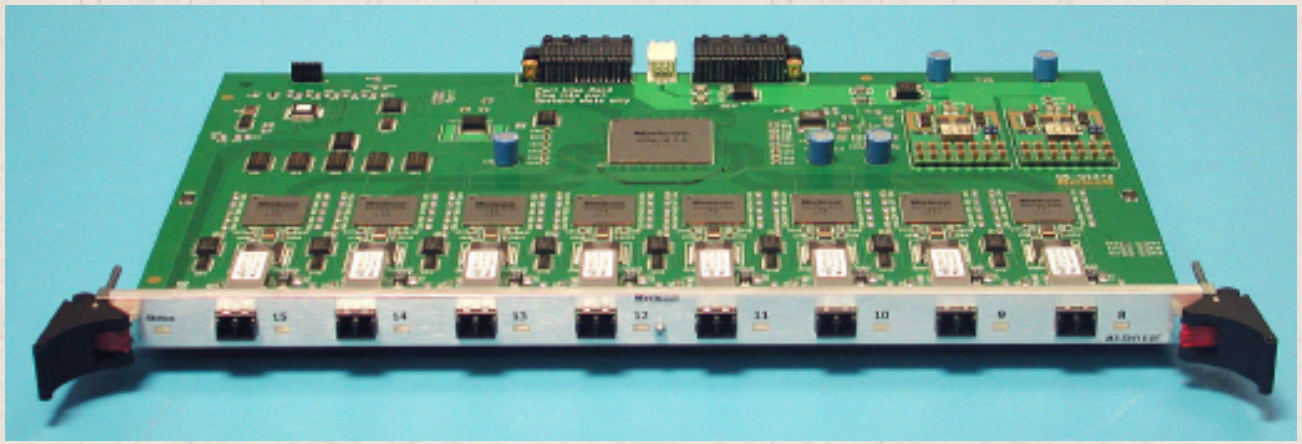
Alternative paths from 4 to 0. 16 port, *7 stage Clos* network = *Benes topology*

Network Topology

Myrinet-2000 Clos Network for 128 Hosts







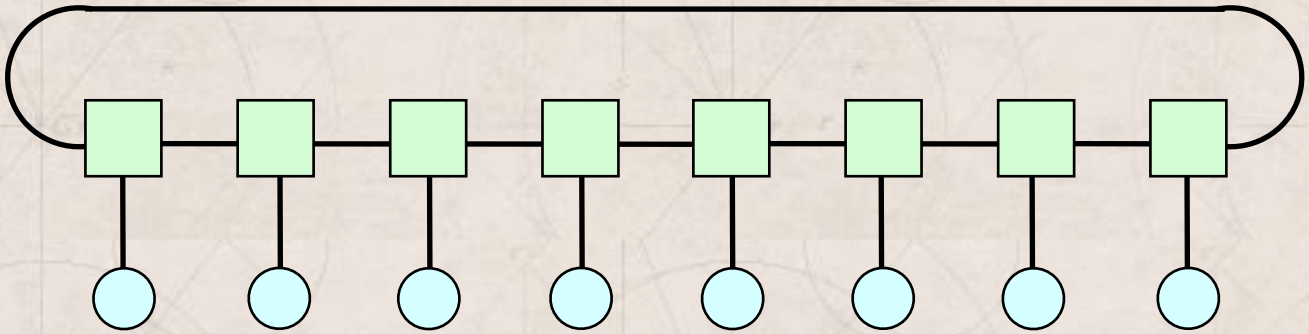
- **Backplane of the M3-E128 Switch**
- **M3-SW16-8F fiber line card (8 ports)**

http://myri.com

Network Topology

Distributed Switched (Direct) Networks

- ***Bidirectional Ring networks***
  - *N* switches (3 × 3) and *N* bidirectional network links
  - Simultaneous packet transport over disjoint paths
  - Packets must hop across intermediate nodes
  - Shortest direction usually selected (*N*/4 hops, on average)



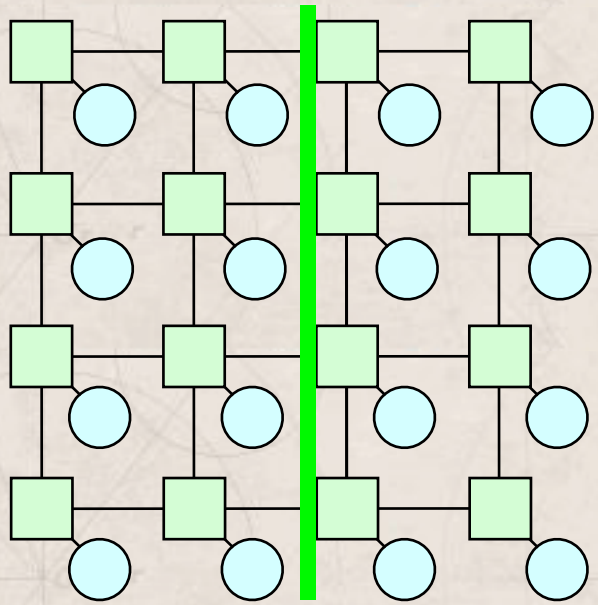
Network Topology

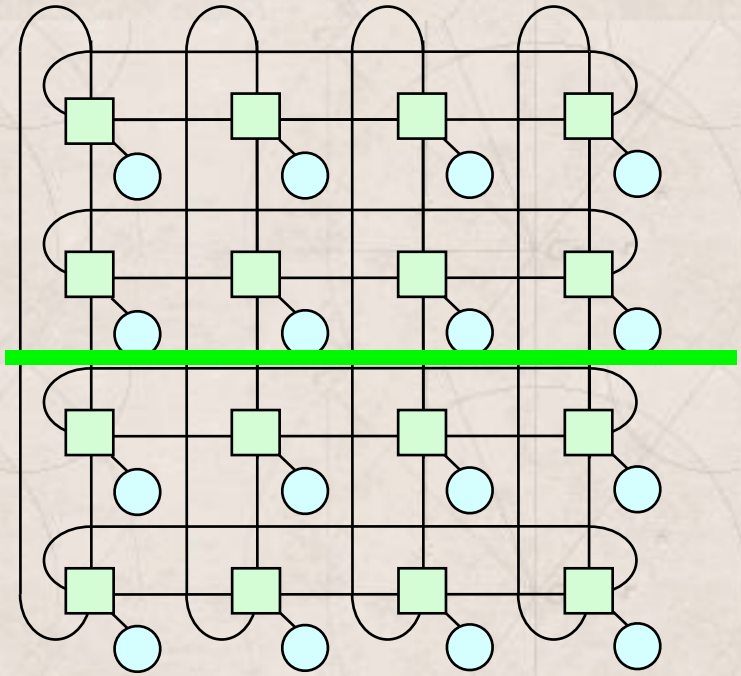
Distributed Switched (Direct) Networks:

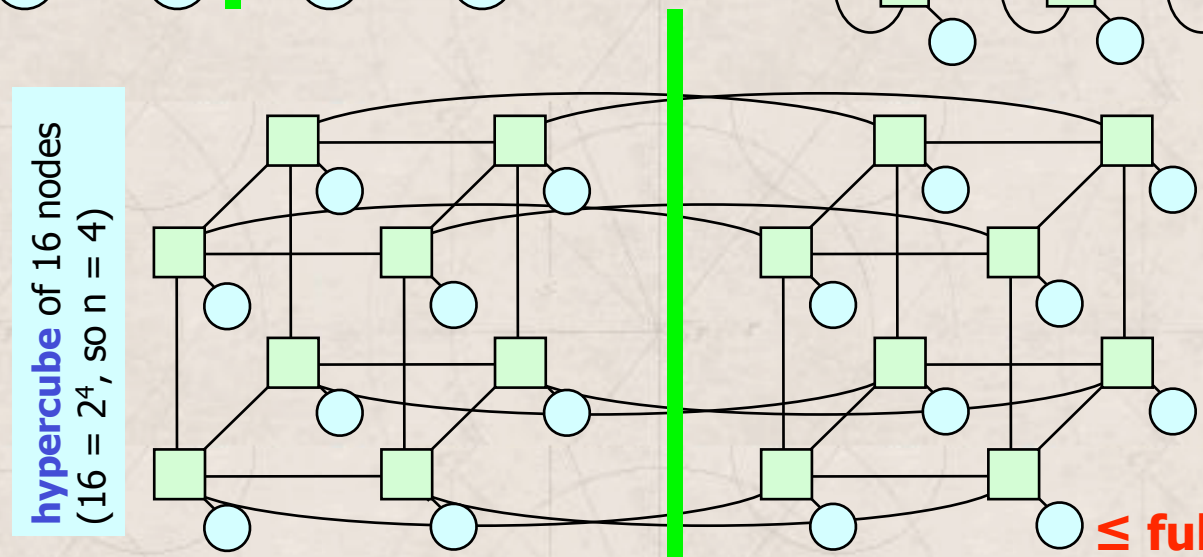
- *Fully connected and ring topologies delimit the two extremes*
- ***The ideal topology:***
  - Cost approaching a ring
  - Performance approaching a fully connected (crossbar) topology
- More practical topologies:
  - ***k-ary n-cubes*** (*meshes*, *tori*, *hypercubes*)
    - ***k*** nodes connected in each dimension, with ***n*** total dimensions
    - *Symmetry* and *regularity*
      - network implementation is simplified
      - routing is simplified

Network Topology

Distributed Switched (Direct) Networks







Network Bisection

≤ full bisection bandwidth!

"Performance Analysis of k-ary n-cube Interconnection Networks," W. J. Dally, IEEE Trans. on Computers, Vol. 39, No. 6, pp. 775–785, June, 1990.

Network Topology

Topological Characteristics of Commercial Machines

Company	System [Network] Name	Max. number of nodes [x # CPUs]	Basic network topology	Injection [Recept'n] node BW in MBytes/s	# of data bits per link per direction	Raw network link BW per direction in Mbytes/sec	Raw network bisection BW (bidir) in Gbytes/s
Intel	ASCI Red Paragon	4,510 [x 2]	2-D mesh 64 x 64	400 [400]	16 bits	400	51.2
IBM	ASCI White SP Power3 [Colony]	512 [x 16]	BMIN w/8-port bidirect. switches (fat-tree or Omega)	500 [500]	8 bits (+1 bit of control)	500	256
Intel	Thunter Itanium2 Tiger4 [QsNet^{II}]	1,024 [x 4]	fat tree w/8-port bidirectional switches	928 [928]	8 bits (+2 control for 4b/5b enc)	1,333	1,365
Cray	XT3 [SeaStar]	30,508 [x 1]	3-D torus 40 x 32 x 24	3,200 [3,200]	12 bits	3,800	5,836.8
Cray	X1E	1,024 [x 1]	4-way bristled 2-D torus (~ 23 x 11) with express links	1,600 [1,600]	16 bits	1,600	51.2
IBM	ASC Purple pSeries 575 [Federation]	>1,280 [x 8]	BMIN w/8-port bidirect. switches (fat-tree or Omega)	2,000 [2,000]	8 bits (+2 bits of control)	2,000	2,560
IBM	Blue Gene/L eServer Sol. [Torus Net]	65,536 [x 2]	3-D torus 32 x 32 x 64	612,5 [1,050]	1 bit (bit serial)	175	358.4

Routing, Arbitration, and Switching

Routing

- Performed at each switch, regardless of topology
- Defines the "allowed" path(s) for each packet (*Which paths?*)
- Needed to *direct packets through network* to intended destinations
- ***Ideally:***
  - *Supply as many routing options to packets as there are paths provided by the topology, and evenly distribute network traffic among network links using those paths, minimizing contention*
- *Problems:* situations that cause packets never to reach their dest.
  - ***Livelock***
    - Arises from an unbounded number of allowed non-minimal hops
    - *Solution:* restrict the number of non-minimal (mis)hops allowed
  - ***Deadlock***
    - Arises from a set of packets being blocked waiting only for network resources (i.e., links, buffers) held by other packets in the set
    - Probability increases with increased traffic & decreased availability

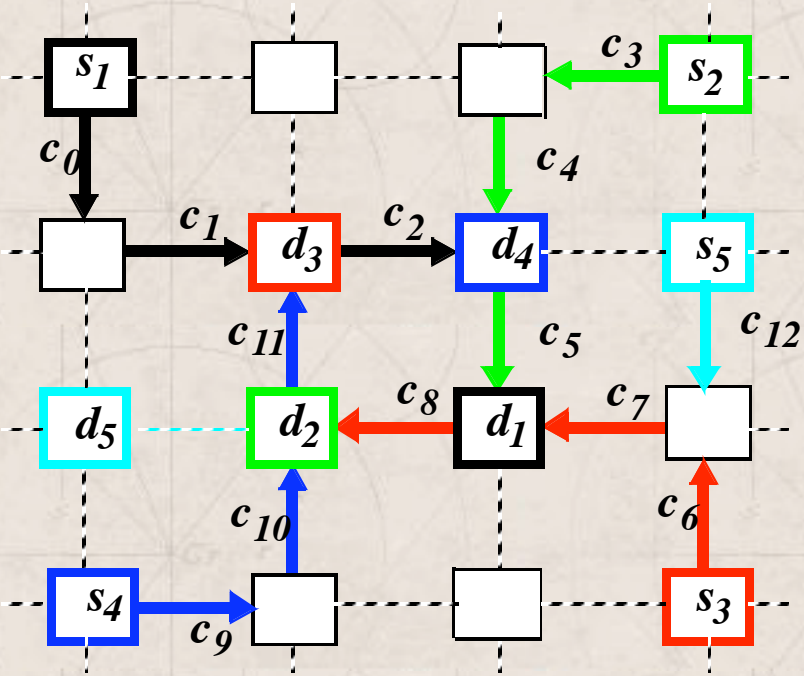
Routing, Arbitration, and Switching

Routing

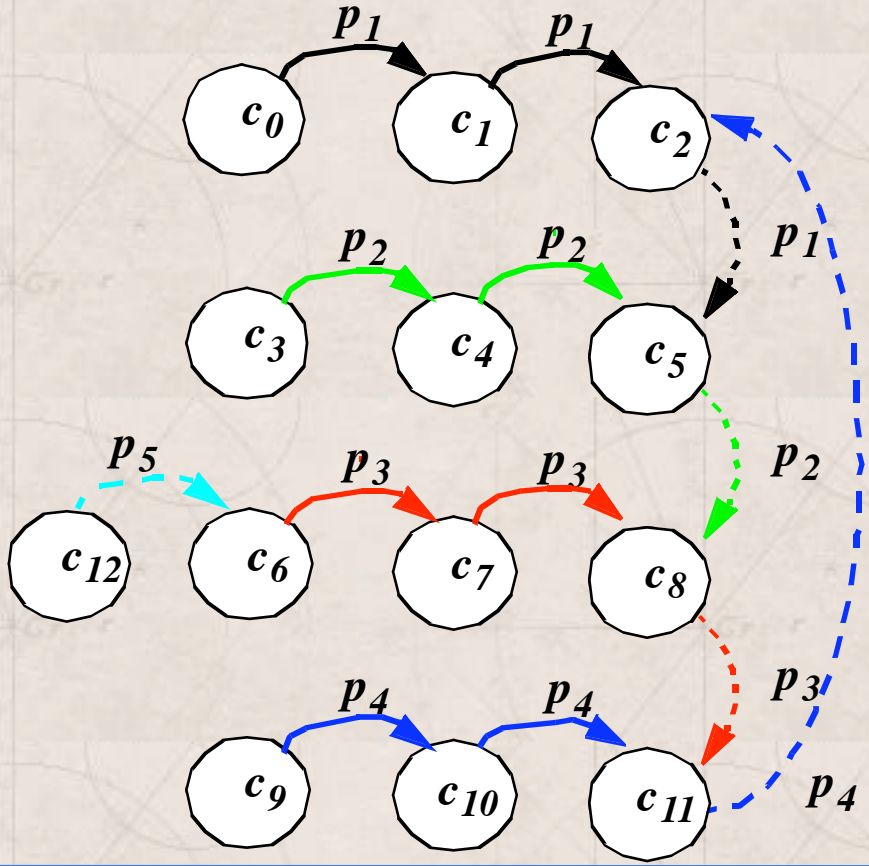
- Common forms of deadlock:
  - ***Routing-induced deadlock***

c_i = channel ***i***
s_i = source node ***i***
d_i = destination node ***i***
p_i = packet ***i***

Routing of packets in a 2D mesh



Channel dependency graph



"A Formal Model of Message Blocking and Deadlock Resolution in Interconnection Networks," S. Warnakulasuriya and T. Pinkston, IEEE Trans. on Parallel and Distributed Systems , Vol. 11, No. 3, pp. 212–229, March, 2000.

Examples of Interconnection Networks

On-Chip Networks (OCNs)

Institution & Processor [Network] name	Year built	Number of network ports [cores or tiles + other ports]	Basic network topology	# of data bits per link per direction	Link bandwidth [link clock speed]	Routing; Arbitration; Switching	# of chip metal layers; flow control; # VCs
MIT Raw [General Dynamic Network]	2002	16 port [16 tiles]	2-D mesh 4 x 4	32 bits	0.9 GBps [225 MHz, clocked at proc speed]	XY DOR w/ request-reply deadlock recovery; RR arbitration; wormhole	6 layers; credit-based; no VCs
IBM POWER5	2004	7 ports [2 PE cores + 5 other ports]	Crossbar	256 b Inst fetch; 64 b for stores; 256 b LDs	[1.9 GHz, clocked at proc speed]	Shortest-path; non-blocking; circuit switch	7 layers; handshaking; no virtual channels
U.T. Austin TRIPS EDGE [Operand Network]	2005	25 ports [25 execution unit tiles]	2-D mesh 5 x 5	110 bits	5.86 GBps [533 MHz clk scaled by 80%]	XY DOR; distributed RR arbitration; wormhole	7 layers; on/off flow control; no VCs
U.T. Austin TRIPS EDGE [On-Chip Network]	2005	40 ports [16 L2 tiles + 24 network interface tile]	2-D mesh 10 x 4	128 bits	6.8 GBps [533 MHz clk scaled by 80%]	XY DOR; distributed RR arbitration; VCT switched	7 layers; credit-based flow control; 4 VCs
Sony, IBM, Toshiba Cell BE [Element Interconnect Bus]	2005	12 ports [1 PPE and 8 SPEs + 3 other ports for memory, I/&O interface]	Ring 4 total, 2 in each direction	128 bits data (+16 bits tag)	25.6 GBps [1.6 GHz, clocked at half the proc speed]	Shortest-path; tree-based RR arb. (centralized); pipelined circuit switch	8 layers; credit-based flow control; no VCs
Sun UltraSPARC T1 processor	2005	Up to 13 ports [8 PE cores + 4 L2 banks + 1 shared I/O]	Crossbar	128 b both for the 8 cores and the 4 L2 banks	19.2 GBps [1.2 GHz, clocked at proc speed]	Shortest-path; age-based arbitration; VCT switched	9 layers; handshaking; no VCs

Examples of Interconnection Networks

Cell Broadband Engine Element Interconnect Bus

- Cell BE is successor to PlayStation 2's Emotion Engine
  - 300 MHz MIPS-based
  - Uses two vector elements
  - 6.2 GFLOPS (Single Precision)
  - 72KB Cache + 16KB Scratch Pad RAM
  - 240mm^2 on 0.25-micron process
- PlayStation 3 uses the Cell BE*
  - 3.2 GHz POWER-based
  - Eight SIMD (Vector) Processor Elements
  - >200 GFLOPS (Single Precision)
  - 544KB cache + 2MB Local Store RAM
  - 235mm^2 on 90-nanometer SOI process

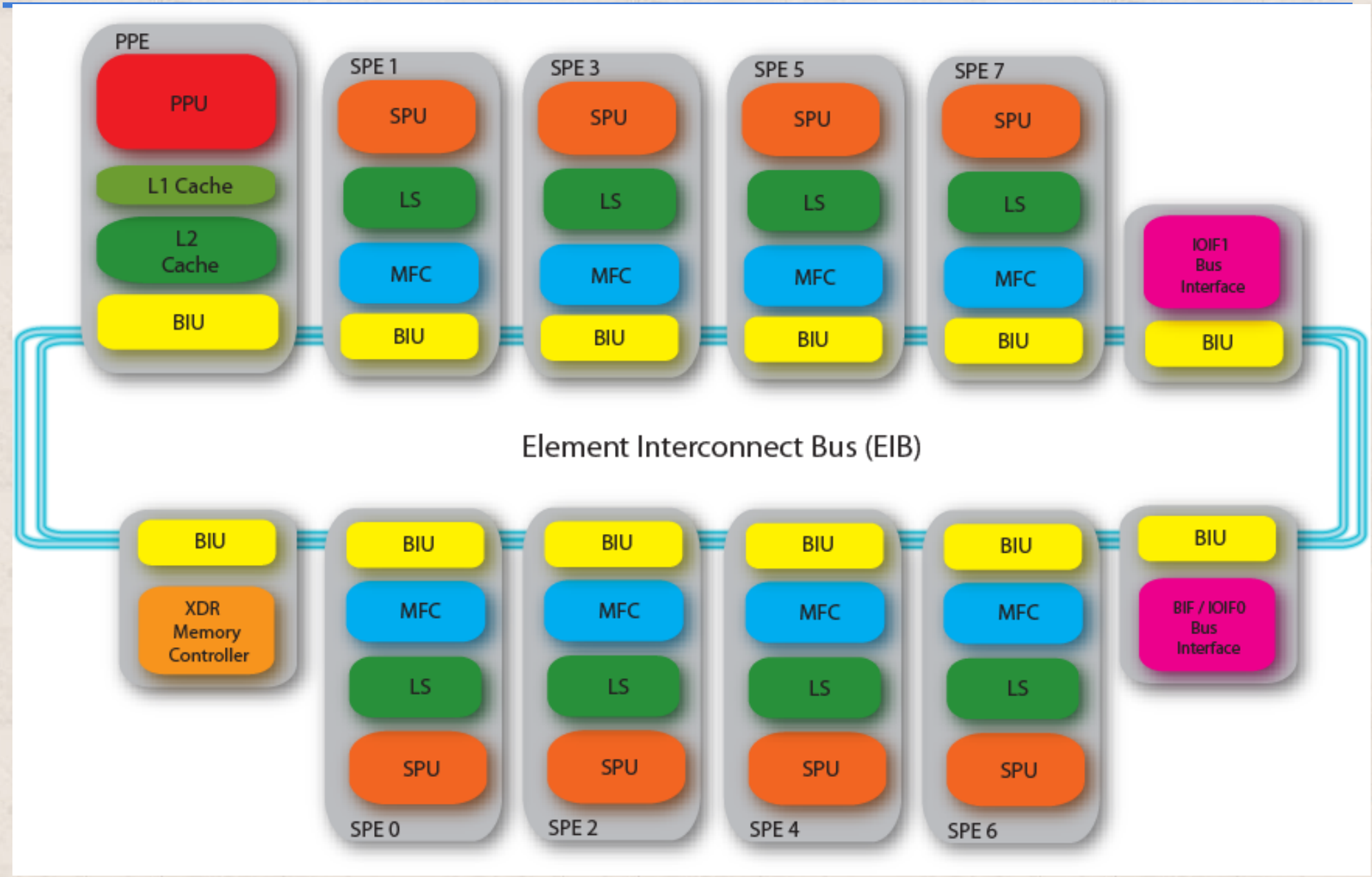
*Sony has decided to use only 7 SPEs for the PlayStation 3 to improve yield. Eight SPEs will be assumed for the purposes of this discussion.

Examples of Interconnection Networks

Cell Broadband Engine Element Interconnect Bus

- Cell Broadband Engine (Cell BE): 200 GFLOPS
  - 12 Elements (devices) interconnected by EIB:
    - *One* 64-bit Power processor element (PPE) with aggregate bandwidth of 51.2 GB/s
    - *Eight* 128-bit SIMD synergistic processor elements (SPE) with local store, each with a bandwidth of 51.2 GB/s
    - *One* memory interface controller (MIC) element with memory bandwidth of 25.6 GB/s
    - *Two* configurable I/O interface elements: 35 GB/s (out) and 25GB/s (in) of I/O bandwidth
  - Element Interconnect Bus (EIB):
    - *Four* unidirectional *rings* (*two in each direction*) each connect the heterogeneous 12 elements (end node devices)
    - Data links: 128 bits wide @ 1.6 GHz; data bandwidth: 25.6 GB/s
    - Provides coherent and non-coherent data transfer
    - Should optimize network traffic flow (throughput) and utilization while minimizing network latency and overhead

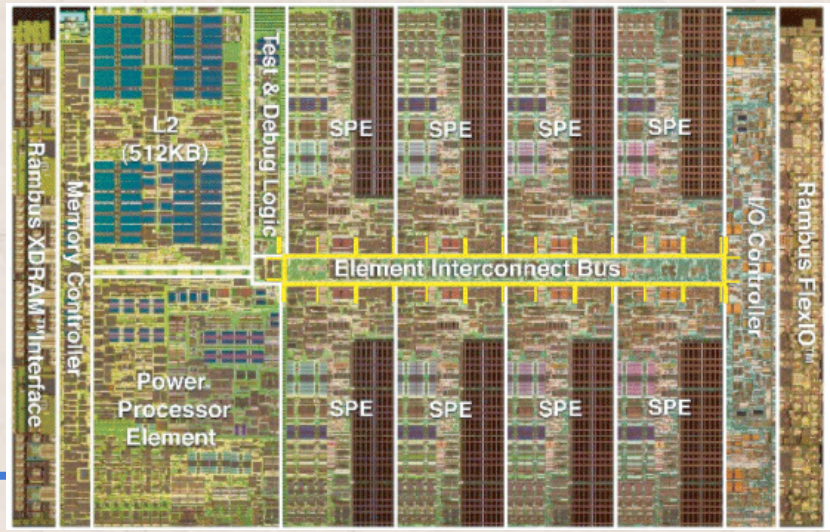
Examples of Interconnection Networks

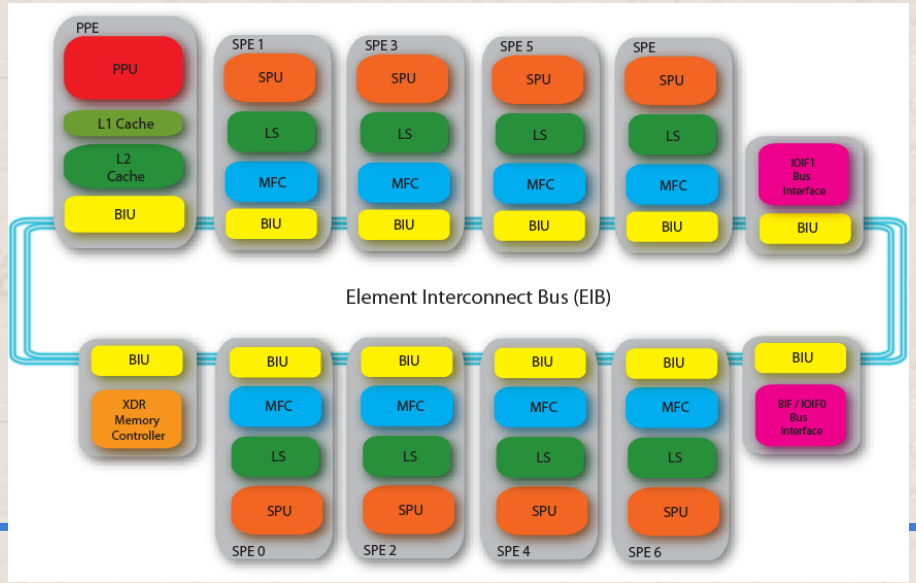


Examples of Interconnection Networks

Cell Broadband Engine Element Interconnect Bus

- *Element Interconnect Bus* (*EIB*)
  - Packet size: 16B – 128B (no headers); pipelined circuit switching
  - Credit-based flow control (command bus central token manager)
  - Two-stage, dual round-robin centralized network arbiter
  - Allows up to 64 outstanding requests (DMA)
    - 64 Request Buffers in the MIC; 16 Request Buffers per SPE
  - Latency: 1 cycle/hop, transmission time (largest packet) 8 cycles
  - Effective bandwidth: peak 307.2 GB/s, max. sustainable 204.8 GB/s





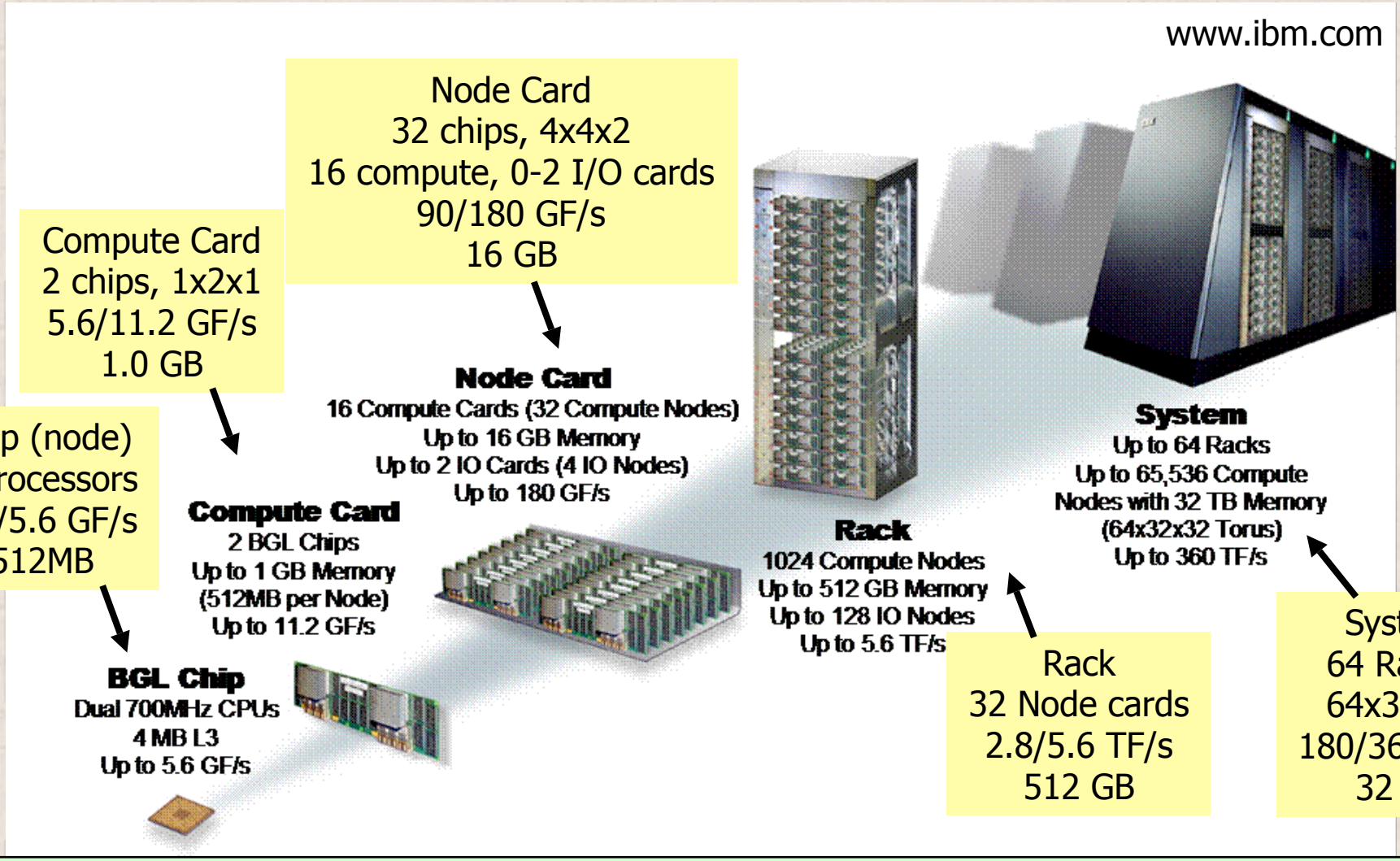
Examples of Interconnection Networks

Blue Gene/L 3D Torus Network

- 360 TFLOPS (peak)
- 2,500 square feet
- Connects 65,536 dual-processor nodes and 1,024 I/O nodes
  - One processor for computation; other meant for communication

Examples of Interconnection Networks

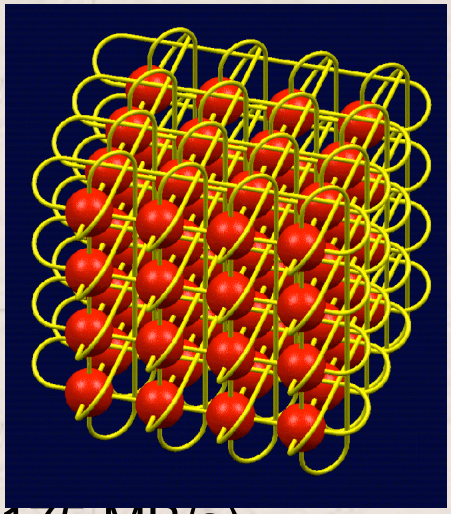
Blue Gene/L 3D Torus Network



Node distribution: Two nodes on a 2 x 1 x 1 compute card, 16 compute cards + 2 I/O cards on a 4 x 4 x 2 node board, 16 node boards on an 8 x 8 x 8 midplane, 2 midplanes on a 1,024 node rack, 8.6 meters maximum physical link length

Examples of Interconnection Networks

Blue Gene/L 3D Torus Network



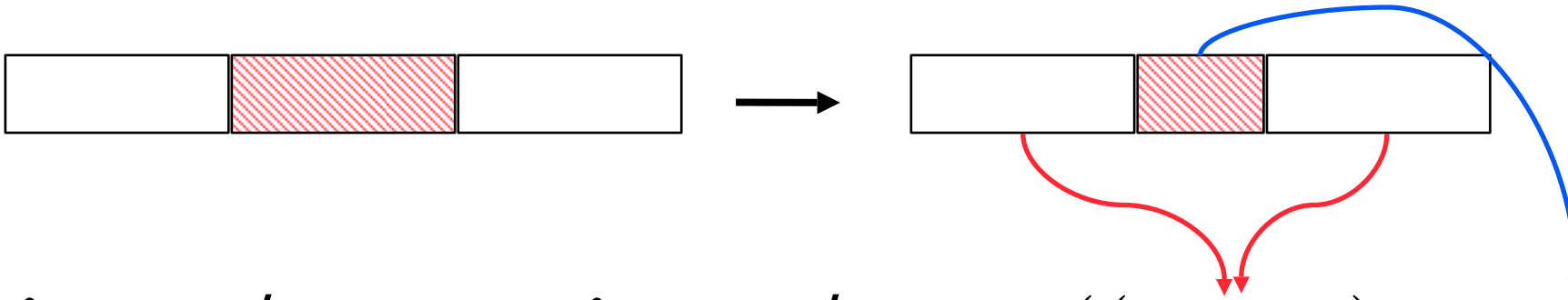
- Main network: 32 x 32 x 64 3-D torus
  - Each node connects to six other nodes
  - Full routing in hardware
- Links and Bandwidth
  - 12 bit-serial links per node (6 in, 6 out)
  - Torus clock speed runs at 1/4th of processor rate
  - Each link is 1.4 Gb/s at target 700-MHz clock rate (175 MB/s)
  - High internal switch connectivity to keep all links busy
    - › External switch input links: 6 at 175 MB/s each (1,050 MB/s aggregate)
    - › External switch output links: 6 at 175 MB/s each (1,050 MB/s aggregate)
    - › Internal datapath crossbar input links: 12 at 175 MB/s each
    - › Internal datapath crossbar output links: 6 at 175 MB/s each
    - › Switch injection links: 7 at 175 MBps each (2 cores, each with 4 FIFOs)
    - › Switch reception links: 12 at 175 MBps each (2 cores, each with 7 FIFOs)

Encountering Amdahl's Law

- Speedup due to enhancement E is

$$\text{Speedup with E} = \frac{\text{Exec time w/o E}}{\text{Exec time with E}}$$

- Suppose that enhancement E accelerates a fraction F (F <1) of the task by a factor S (S>1) and the remainder of the task is unaffected



$$\text{ExTime w/ E} = \text{ExTime w/o E} \times ((1 - F) + F/S)$$

$$\text{Speedup w/ E} = \frac{1}{(1 - F) + F/S}$$

Challenges of Parallel Processing

- Application parallelism ⇒ primarily via new algorithms that have better parallel performance
- Long remote latency impact ⇒ both by architect and by the programmer
- For example, reduce frequency of remote accesses either by
  - Caching shared data (HW)
  - Restructuring the data layout to make more accesses local (SW)

Examples: Amdahl's Law

$$\text{Speedup w/ E} = 1 / ((1\text{-}F) + F/S)$$

- Consider an enhancement which runs 20 times faster but which is only usable 25% of the time.
  - Speedup w/ E =
- What if it's usable only 15% of the time?
  - Speedup w/ E =
- Amdahl's Law tells us that to achieve linear speedup with 100 processors, none of the original computation can be scalar!
- To get a speedup of 99 from 100 processors, the percentage of the original program that could be scalar would have to be 0.01% or less

Challenges of Parallel Processing

- Second challenge is long latency to remote memory
- Suppose 32 CPU MP, 2 GHz, 200 ns remote memory, all local accesses hit memory hierarchy and base CPI is 0.5. (Remote access = 200/0.5 = 400 clock cycles.)
- What is performance impact if 0.2% instructions involve remote access?
  - 1.5X
  - 2.0X
  - 2.5X

Challenges of Parallel Processing

- Application parallelism ⇒ primarily via new algorithms that have better parallel performance
- Long remote latency impact ⇒ both by architect and by the programmer
- For example, reduce frequency of remote accesses either by
  - Caching shared data (HW)
  - Restructuring the data layout to make more accesses local (SW)

How hard is parallel programming anyway?

- Parallel Programmer Productivity: A Case Study of Novice Parallel Programmers
  - Lorin Hochstein, Jeff Carver, Forrest Shull, Sima Asgari, Victor Basili, Jeffrey K. Hollingsworth, Marvin V. Zelkowitz
  - Supercomputing 2005

Why use students for testing?

- First, multiple students are routinely given the same assignment to perform, and thus we are able to conduct experiments in a way to control for the skills of specific programmers.
- Second, graduate students in a HPC class are fairly typical of a large class of novice HPC programmers who may have years of experience in their application domain but very little in HPC-style programming.
- Finally, due to the relatively low costs, student studies are an excellent environment to debug protocols that might be later used on practicing HPC programmers.

Tests run

	Serial	MPI	OpenMP	Co-Array Fortran	StarP	XMT
Nearest-Neighbor Type Problems						
Game of Life	C3A3	C3A3 C0A1 C1A1	C3A3			
Grid of Resistors	C2A2	C2A2	C2A2		C2A2	
Sharks & Fishes		C6A2	C6A2	C6A2		
Laplace's Eq.		C2A3			P2A3	
SWIM			C0A2			
Broadcast Type Problems						
LU Decomposition			C4A1			
Parallel Mat-vec					C3A4	
Quantum Dynamics		C7A1				
Embarrassingly Parallel Type Problems						
Buffon-Laplace Nee-dle		C2A1 C3A1	C2A1 C3A1		C2A1 C3A1	
(Miscellaneous Problem Types)						
Parallel Sorting		C3A2	C3A2		C3A2	
Array Compaction						C5A1
Randomized Selection						C5A2

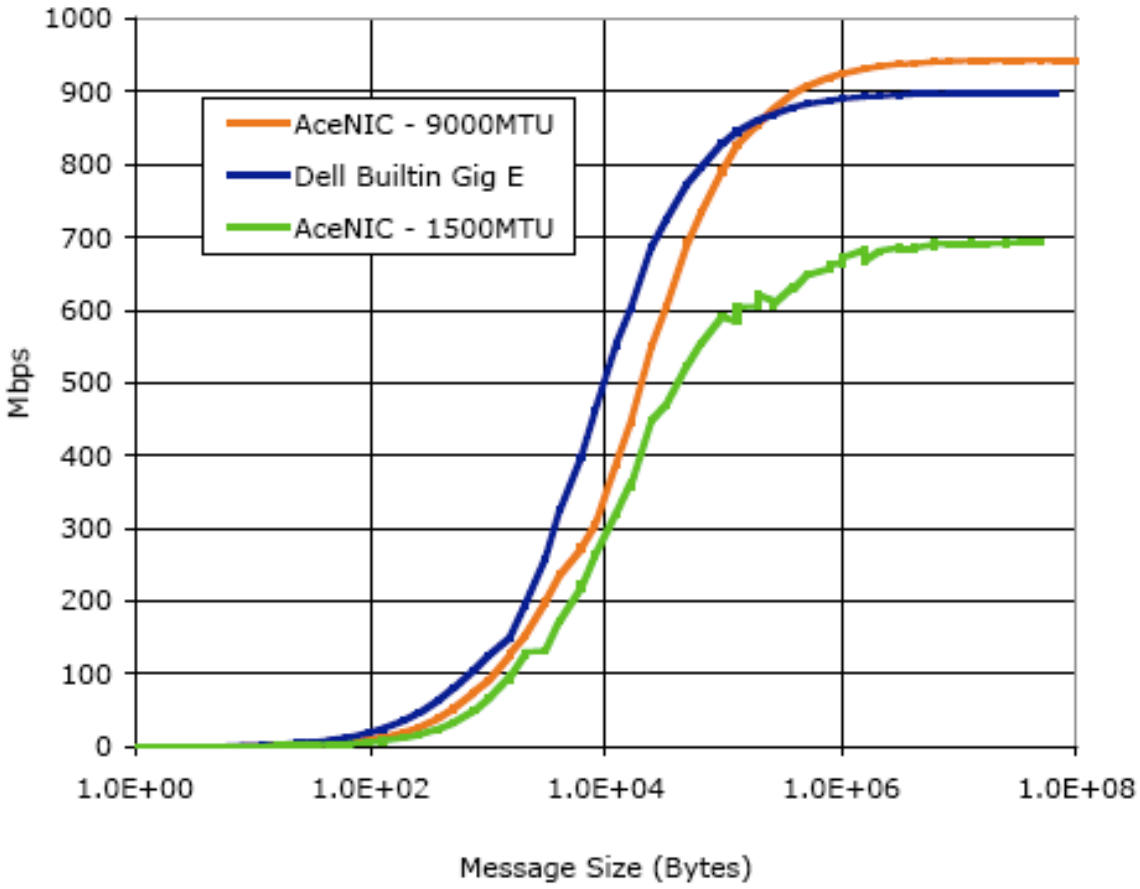
What They Learned

- Novices are able to achieve speedup on a parallel machine.
- MPI and OpenMP both require more {code, cost per line, effort} than serial implementations
  - MPI takes more effort than OpenMP

Data set	**Programming Model**	**Speedup on 8 processors**
Speedup w.r.t. serial version		
C1A1	MPI	mean 4.74, sd 1.97, n=2
C3A3	MPI	mean 2.8, sd 1.9, n=3
C3A3	OpenMP	mean 6.7, sd 9.1, n=2
Speedup w.r.t. parallel version run on 1 processor		
C0A1	MPI	mean 5.0, sd 2.1, n=13
C1A1	MPI	mean 4.8, sd 2.0, n=3
C3A3	MPI	mean 5.6, sd 2.5, n=5
C3A3	OpenMP	mean 5.7, sd 3.0, n=4

Ethernet Performance

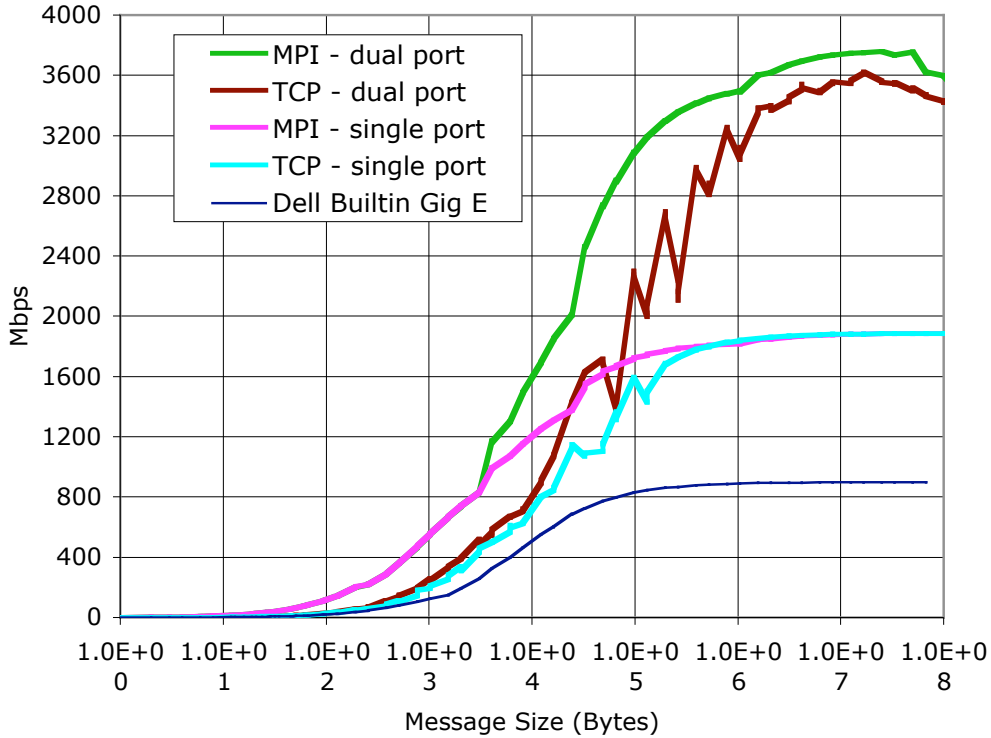
- Achieves close to theoretical bw even with default 1500 B/ message
- Broadcom part: 31 µs latency



Performance Characteristics of Dual-Processor HPC Cluster Nodes Based on 64-bit Commodity Processors, Purkayastha et al.

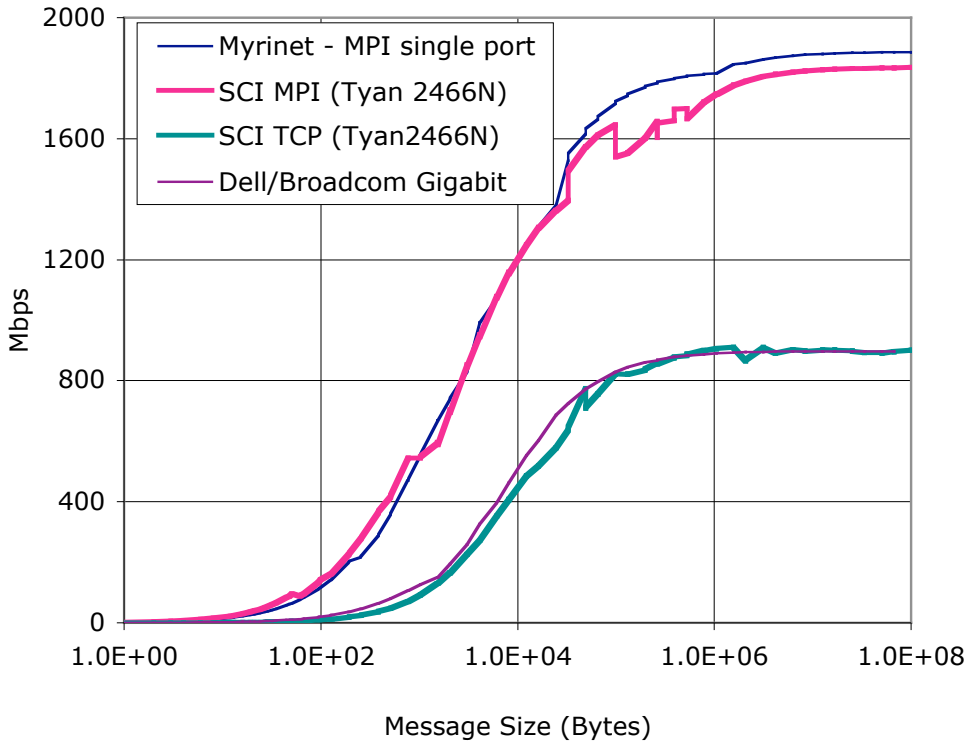
Myrinet

- Interconnect designed with clustering in mind
- 2 Gbps links, possibly 2 physical links (so 4 Gb/s)
- $850/node up to 128 nodes, $1737/node up to 1024 nodes
- MPI latency 6–7 µs
- TCP/IP latency 27–30 µs
  - Strong support!



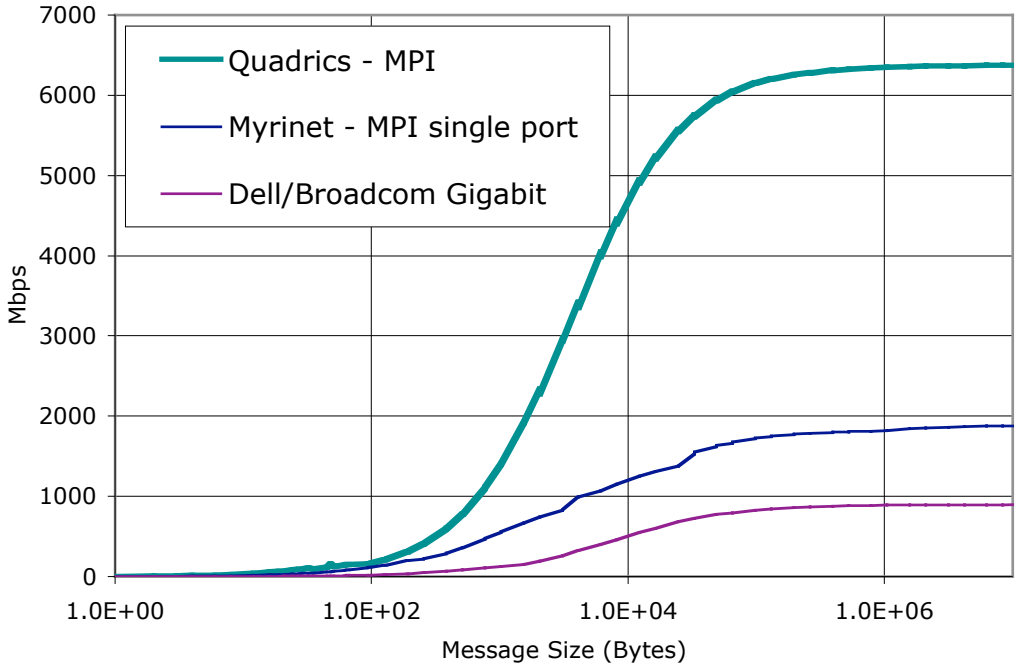
Scalable Coherent Interface

- Not a “switched” interconnect
- Max scalability: 64–100 nodes for 2D torus ($1095/node), 640–1000 for 3D torus ($1595/node)
- MPI: 4 µs latency, 1830 Mbps
- TCP/IP: 900 Mbps



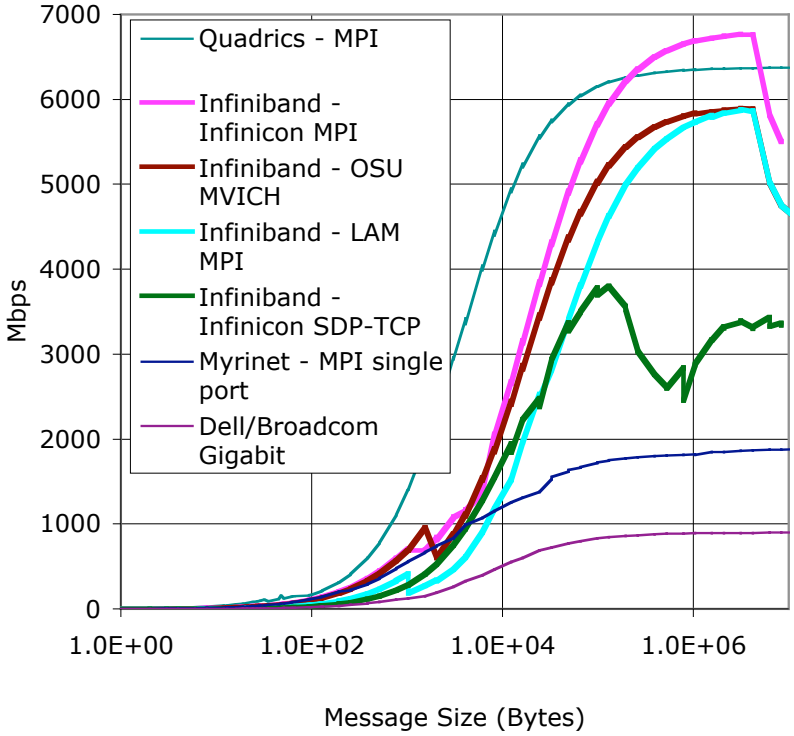
Quadrics

- “a premium interconnect choice on high-end systems such as the Compaq AlphaServer SC”
- Max 4096 ports
- $2400/port for small system up to $4078/port for 1024 node system
- MPI: 2–3 µs latency, 6370 Mpbs bw



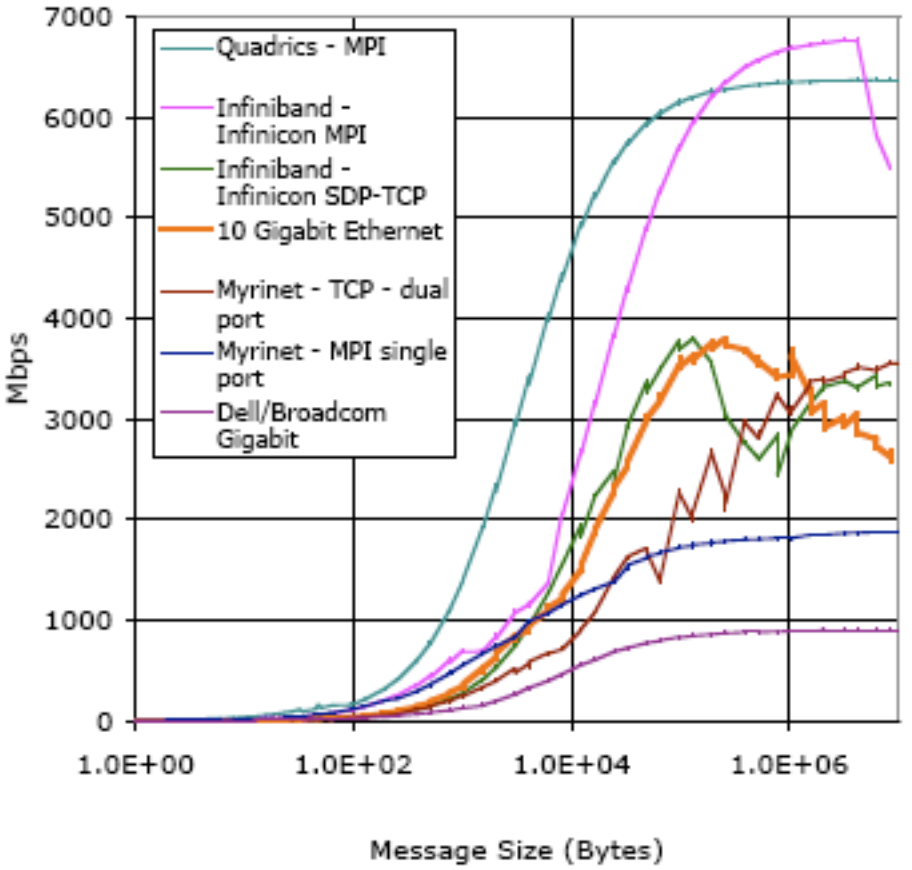
Infiniband

- Defined by industry consortium
  - Scalable—base is 2.5 Gb/s link, scales to 30 Gb/s
  - Current parts are 4x (10 Gb/s)
- $1200–1600/node
- Also used as system interconnect
- 6750 Mb/s, 6–7 µs



10 Gb Ethernet

- Similar tradeoffs to previous Ethernets
- $10k per switch port
- Only 3700 Mb/s



Application Results

#node, 2Procs per	1	2	4
[illegible]	–		
[illegible]	–		
[illegible]	–		
[illegible]	–		
[illegible]			

# node, 1 proc per	1	2	4
[illegible]	–		
[illegible]	–		
[illegible]	–		
[illegible]	–		
[illegible]			