University of BRISTOL

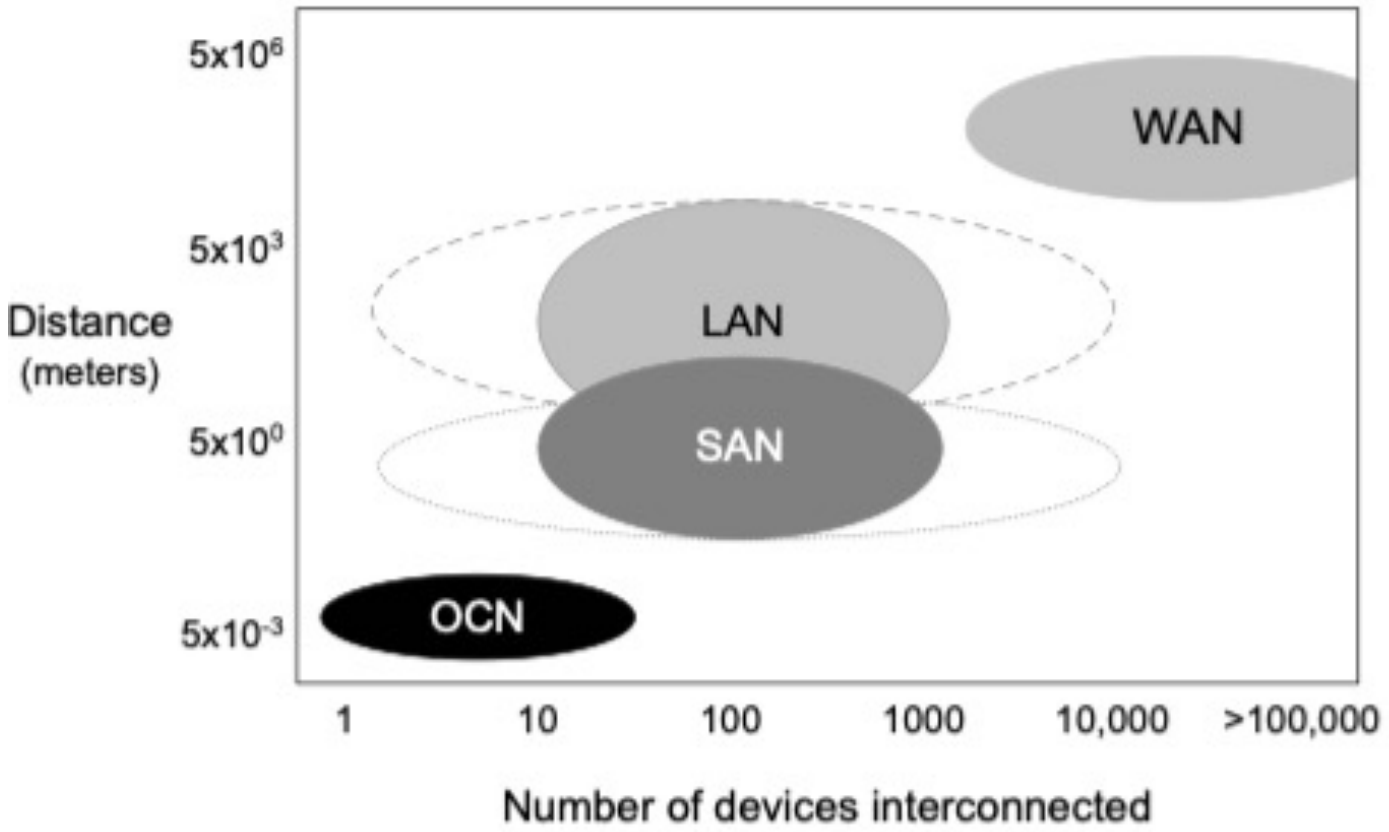
Lecture 4
Interconnection Networks
4.1 Network Topologies

Introduction of Interconnection Networks

University of BRISTOL

- Interconnection network is the communication substrate for components in a computer system
  - Many-core Chip Multiprocessors employ an on-chip network for low-latency, high-bandwidth load/store operations between processing cores and memory, and among processing cores within a chip package.
- Processor, memory, and its associated IO devices are often packaged together and referred to as a ***processing node***. The system-level interconnection network connects all the processing nodes according to the ***network topology***.

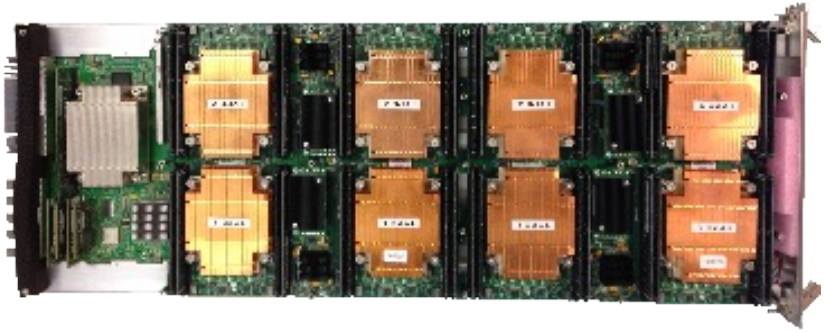
Interconnection Networks



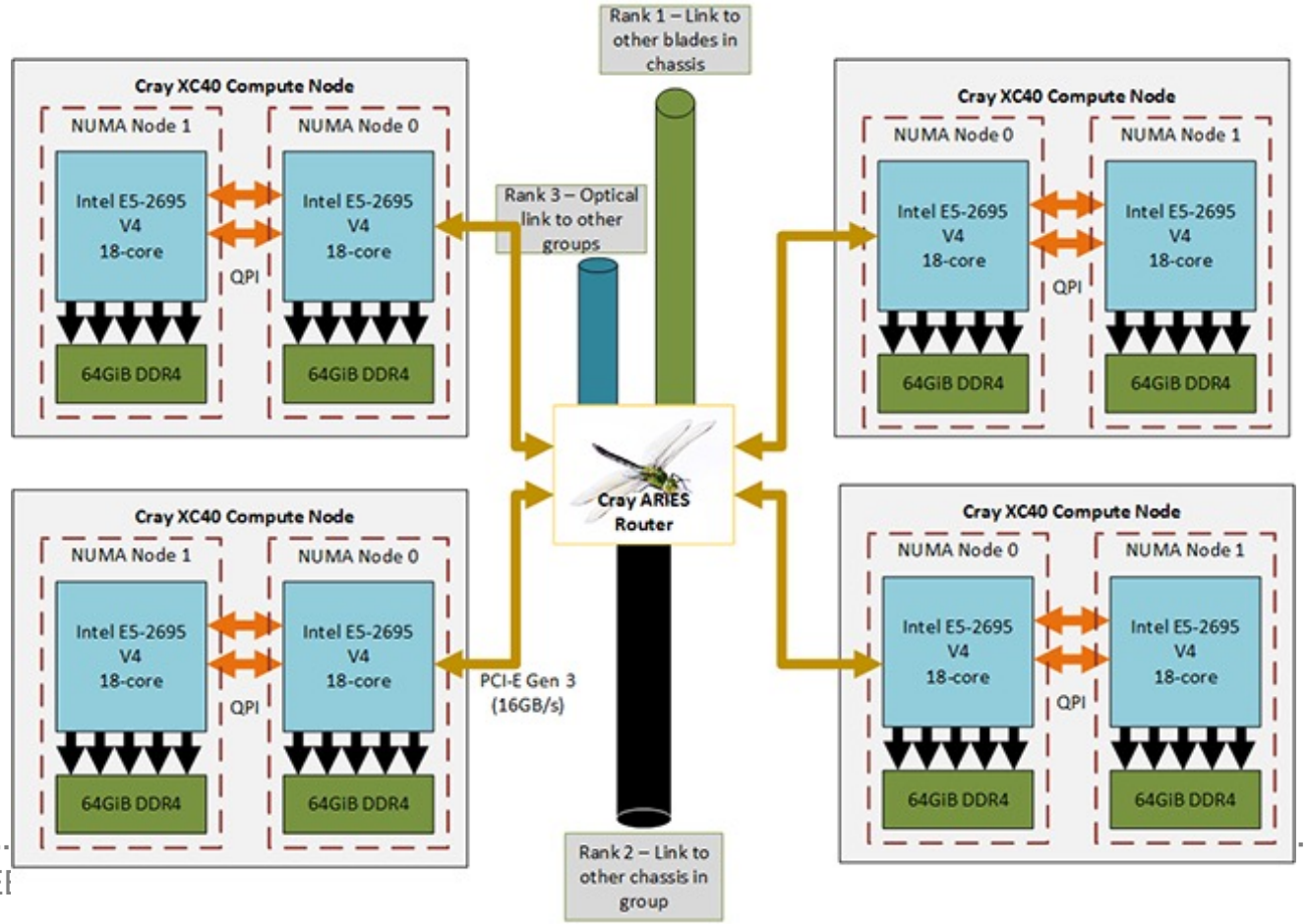
- On-chip networks (OCNs)
- System/storage area networks
- Local area networks (LANs)
- Wide area networks (WANs)

Example of Supercomputing and Interconnection networks

University of BRISTOL



An XC40 compute blade. In the main part of the blade, you can see the heat sinks for the eight CPU chips of the four nodes. At the back of the blade is the Aries router.



A diagram of an XC40 compute blade. Each blade has four dual-socket nodes and an Aries router chip.

To build a network

University of BRISTOL

- Topology (how things are connected):
  - Crossbar, ring, 2-D and 3-D meshes or torus, hypercube, tree, butterfly, perfect shuffle ....
- Routing algorithm (path used):
  - Example in 2D torus: all east-west then all north-south (avoids deadlock).
- Switching strategy:
  - Circuit switching: full path reserved for entire message, like the telephone.
  - Packet switching: message broken into separately-routed packets, like the post office.
- Flow control (what to do if there is congestion):
  - Store data temporarily in buffers, re-route data to other nodes, tell source node to temporarily halt, discard, etc.

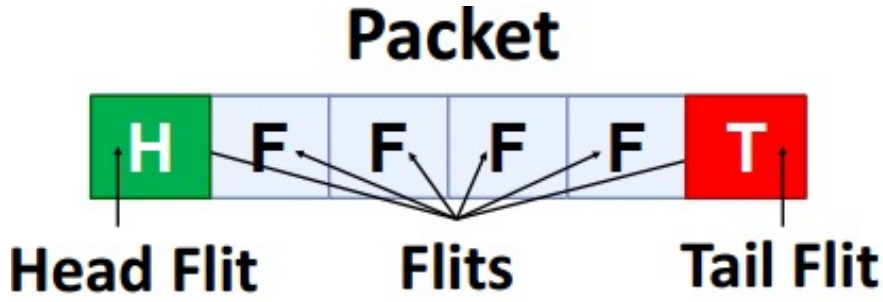
Terminology

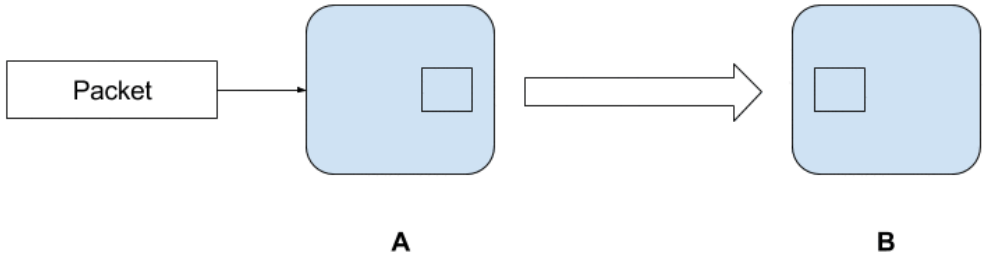
University of BRISTOL

- Network interface
  - Connects endpoints (e.g. cores) to network.
  - Decouples computation/communication
- Links
  - Bundle of wires that carries a signal
- Switch/router
  - Connects fixed number of input channels to fixed number of output channels
- Channel/path/route
  - A single logical connection between routers/switches
  - Average hop count over all sources and destinations:

More Terminology

- Node
  - A network endpoint connected to a router/switch
- Message
  - Unit of transfer for network clients (e.g. cores, memory)
- Packet
  - Unit of transfer for network
- Flit
  - Flow control digit
  - Unit of flow control within network





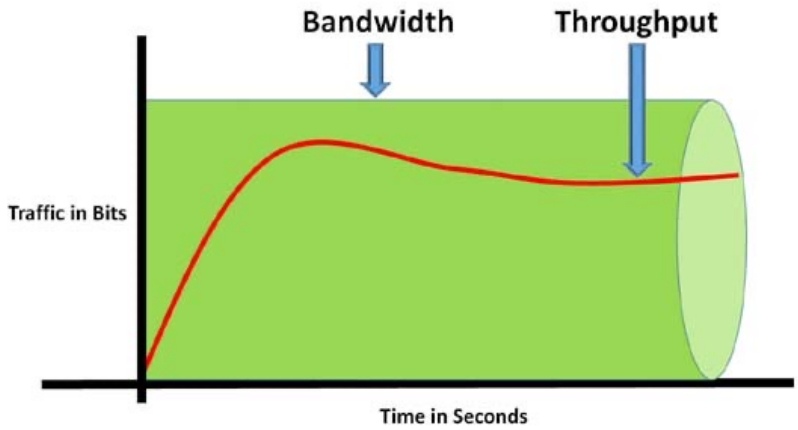
Bandwidth and Throughput

University of BRISTOL

- Bandwidth:
  - The bandwidth of a link = $w \times 1/t$
    - w is the number of channels (i.e. wires, wavelengths)
    - t is the time per bit (i.e. 100 ps/bit)
  - Bandwidth in b/s, i.e., $10^* 10^{10}$ bits/sec = 100 Gb/ s
  - Effective bandwidth is usually lower than physical link bandwidth due to packet overhead.

Routing and control header
Data payload
Error code
Trailer

- Throughput: the quantity of data being sent and received within a unit of time
  - how much data was transferred from a source at any given time



Throughput and Maximum Channel Load

University of BRISTOL

- The throughput of a network is the data rate in bits per second that the network accepts per input port.
  - A property of the entire network
  - Depend on routing and flow control
- Maximum channel load: $\gamma_{max} = max_{c \in C}\, \gamma_c$

  The ratio of the bandwidth demanded from channel c to the bandwidth of the input ports: γ_c

$$\gamma_{max} = max_{c \in C}\ \gamma_c \geq\ \gamma_b = \frac{N}{2B_c}$$

- Ideal throughput of a topology: the input bandwidth that saturate the bottleneck channel $\Theta_{ideal} = \frac{b}{\gamma_{max}} \leq \frac{2bB_c}{N}$

Performance Properties of a Network: Latency

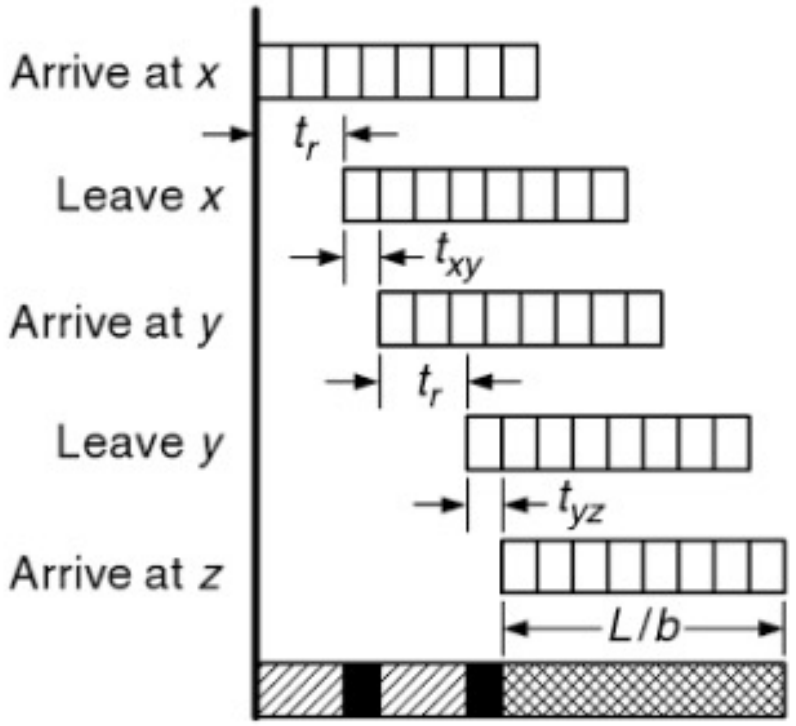
University of BRISTOL

- Latency: delay between send and receive times
  - Latency tends to vary widely across architectures
  - Vendors often report hardware latencies (wire time)
  - Application programmers care about software latencies
  - Observations:
    - Hardware/software latencies often differ by 1-2 orders of magnitude
    - Maximum hardware latency varies with diameter, but the variation in software latency is usually negligible
    - Latency is important for programs with many small messages

Zero-load latency (no contention)

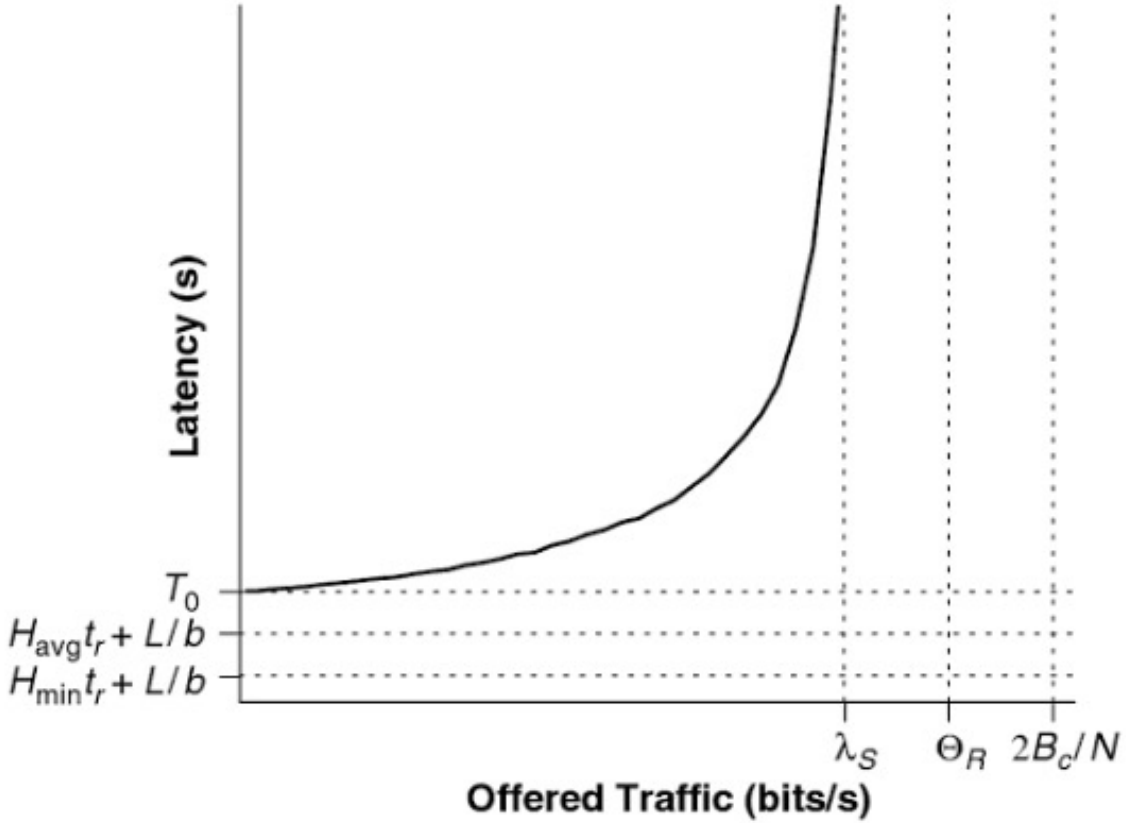
$$T_0 = H_{min} t_r + \frac{D_{min}}{v} + \frac{L}{b}$$

- $T_r = H_{min} t_r$. Router delay, H_{min}: average minimum hop count. t_r: single router delay.
- Time of flight: $T_w = D_{min}/v$. D_{min} average distance. v: propagation velocity.



$$H_{min} = \frac{1}{N^2} \sum_{x,y \in N} H(x,y)$$

Latency vs. offered traffic curve



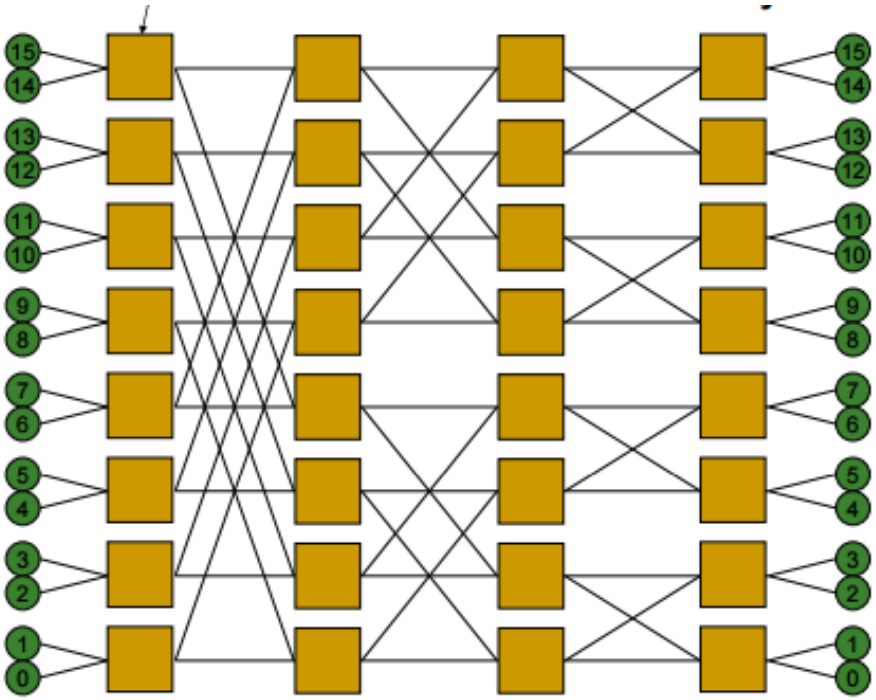
L: packet length
b: bandwidth
Hmin: average minimum hop count
Havg: average hop count

T_0: Zero-load latency
λ_s: Saturation throughput
B_c: Bisection bandwidth
N: node number
Θ_R: routing algorithm related bounding

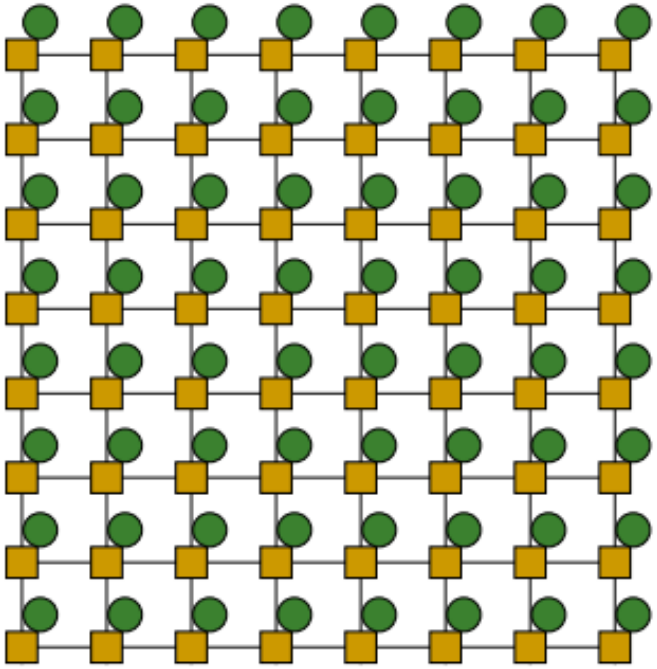
Direct & Indirect Networks

University of BRISTOL

- Direct: Every switch also is network end point
  - Ex: mesh, torus, and hypercubes
- Indirect: Not all switches are end points
  - Ex: Butterfly, Fat Tree, Clos



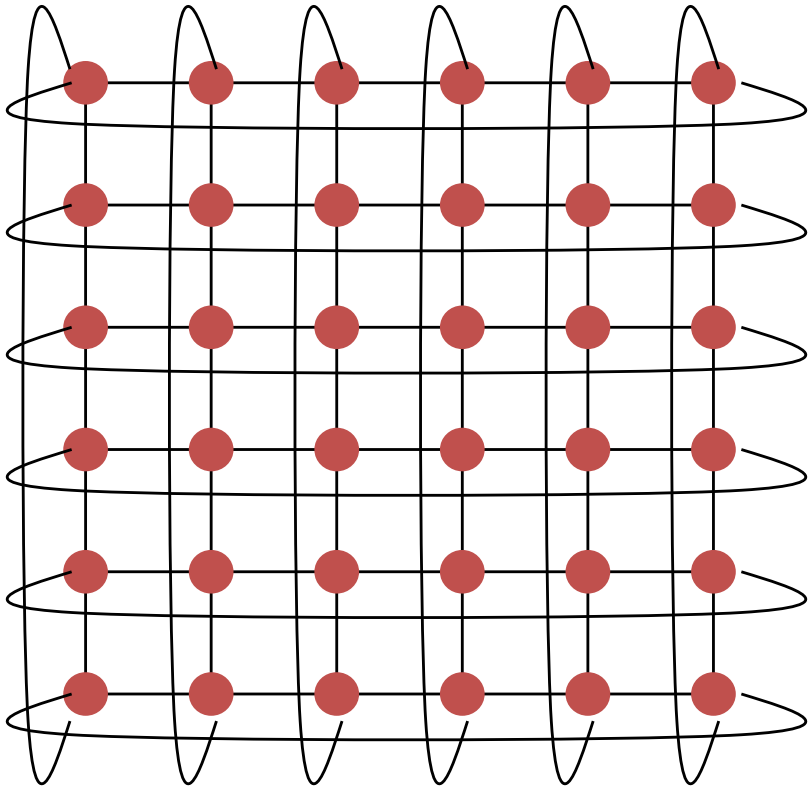
Indirect



Direct

Topology

- Definition: determines arrangement of channels and nodes in network
- First step in network design
- Routing and flow control build on properties of topology



A 6-ary 2-cube network
2D torus network

Network Topology

University of BRISTOL

- In the past, there was considerable research in network topology and in mapping algorithms to topology.
  - Key cost to be minimized: number of "hops" between nodes (e.g. "store and forward")
  - Modern networks hide hop cost (i.e., "cut-through" or "wormhole " flow control), so topology is no longer a major factor in algorithm performance.

Key metrics for network topology

University of BRISTOL

- Diameter

  the maximum (over all pairs of nodes) of the shortest path between a given pair of nodes.

- Average routing distance

  average number of hops across all valid routes

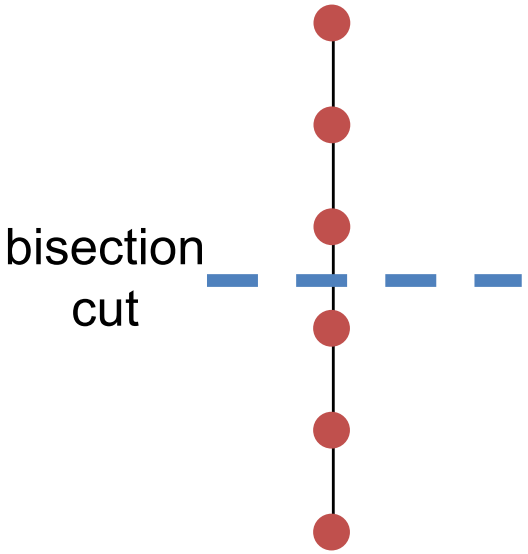
- Bisection bandwidth

Bisection bandwidth

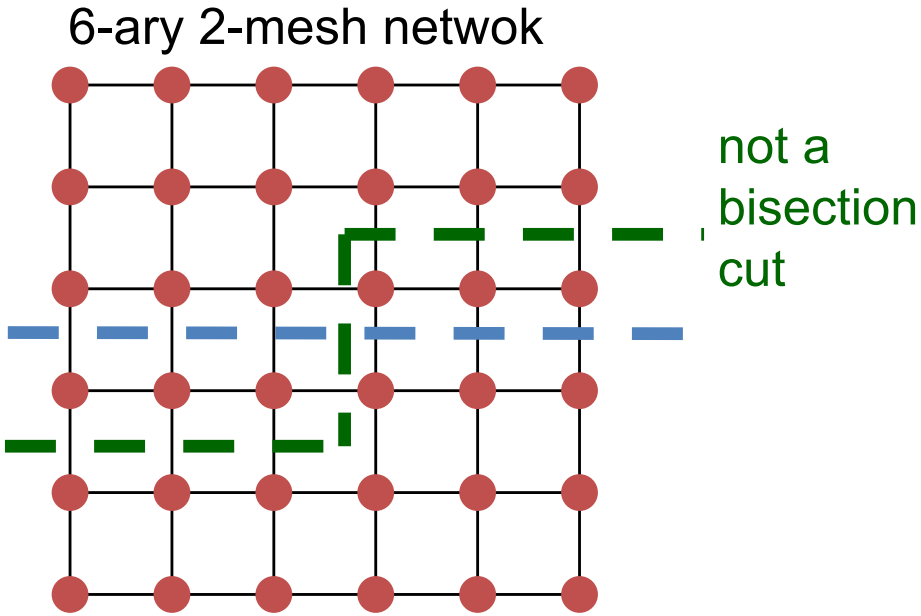
University of BRISTOL

- A bisection bandwidth of a network is the available bandwidth between two partitions that cut the entire network or the terminal nodes nearly in half.
  - The bisection bandwidth of a network, B_b, is the minimum bandwidth over all bisections of the network.
  - The channel bisection of a network, B_c, is the minimum channel count over all bisections of the network.
- In network cost model, the bisection bandwidth of a network indicates the amount of global wiring required to implement it.

Performance Properties of a Network: Bisection Bandwidth



bisection bw= link bw

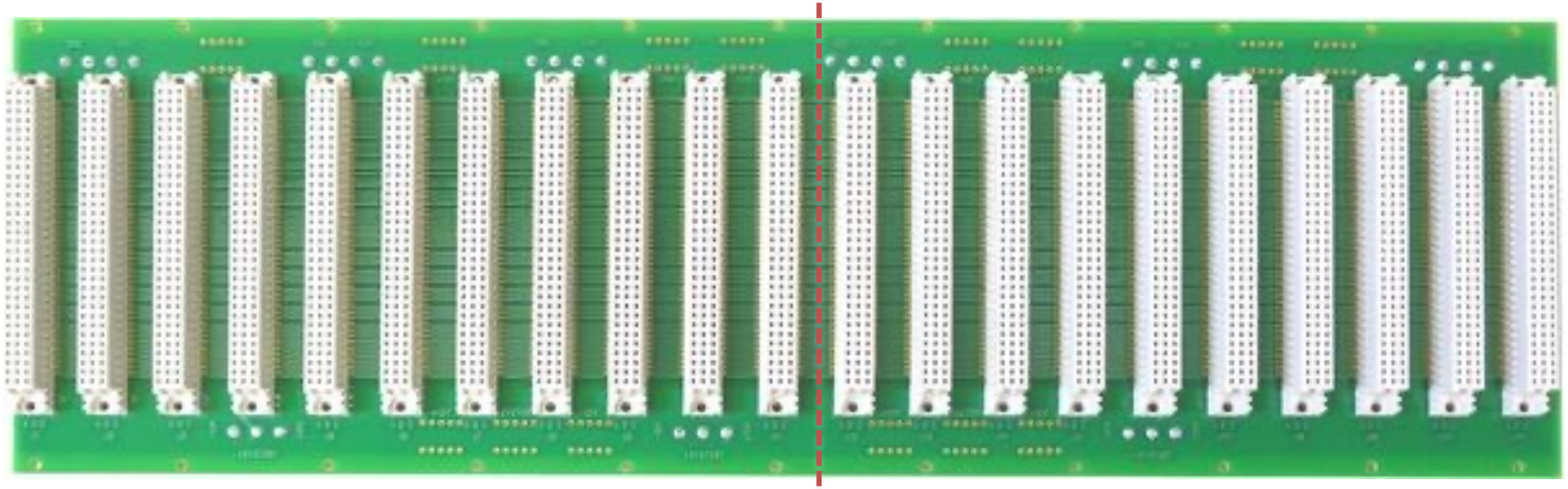


*bisection bw = sqrt(n) * link bw , n = No. of nodes*

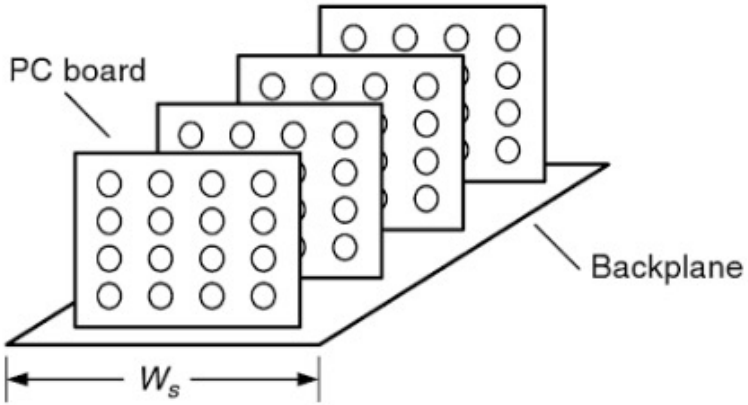
- **Bisection Bandwidth** is important for algorithms in which all processors need to communicate with all others

- **Why is it relevant:** if traffic is completely random, the probability of message going across the two halves is ½.
  - if all nodes send a message, the bisection bandwidth will have to be N/2

Backplane

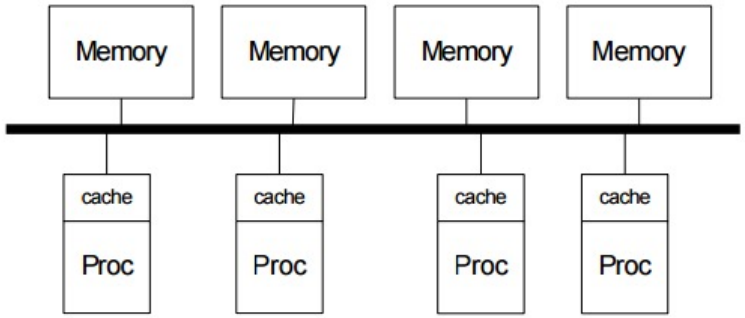


- Bisection bandwidth gives the true bandwidth available in the entire system. Bisection bandwidth accounts for the bottleneck bandwidth of entire network.
- Need to take account physical limitations

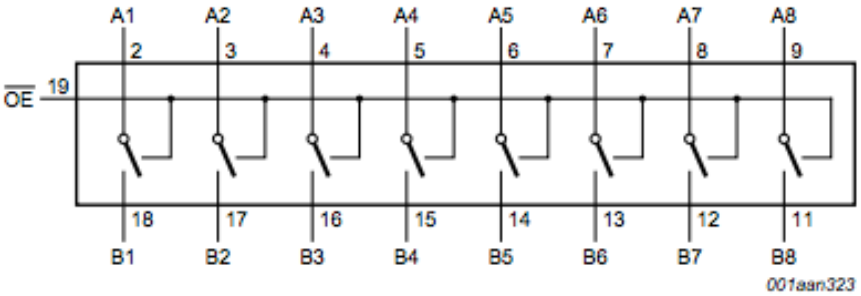


Bus Switch

- Pros:
  - Simple
  - Cost effective for a small number of nodes
  - Easy to implement coherence



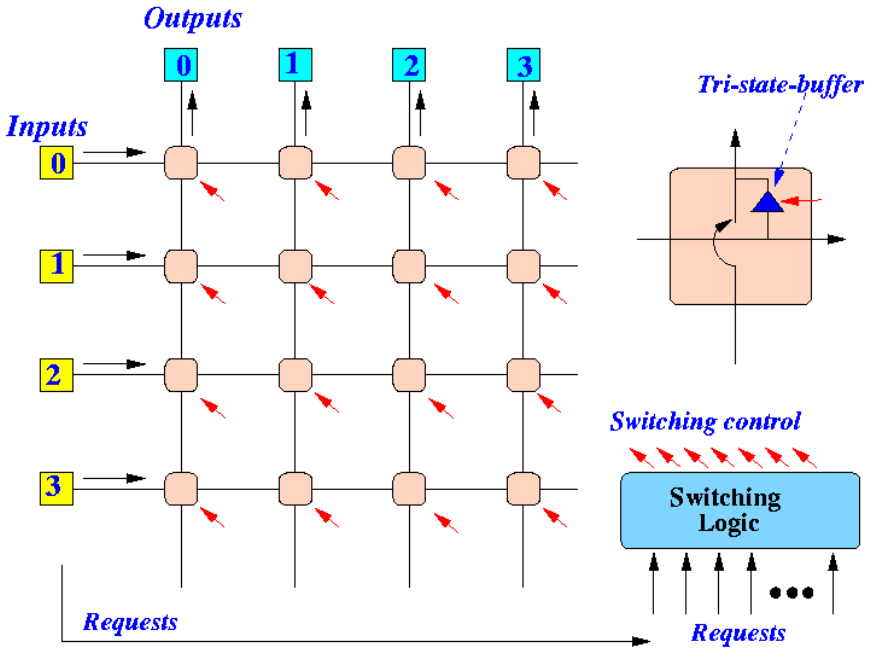
- Cons:
  - Not scalable to large number of nodes (limited bandwidth, electrical loading)
  - High contention Memory

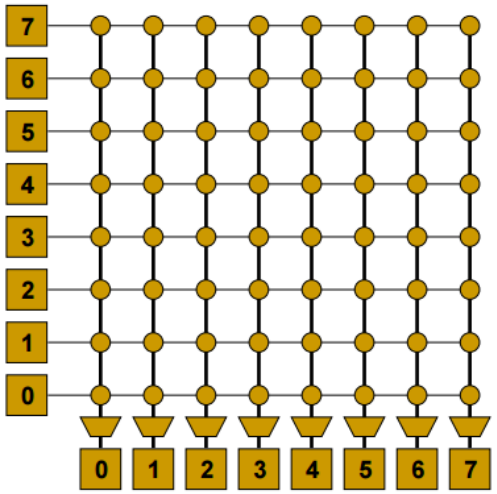


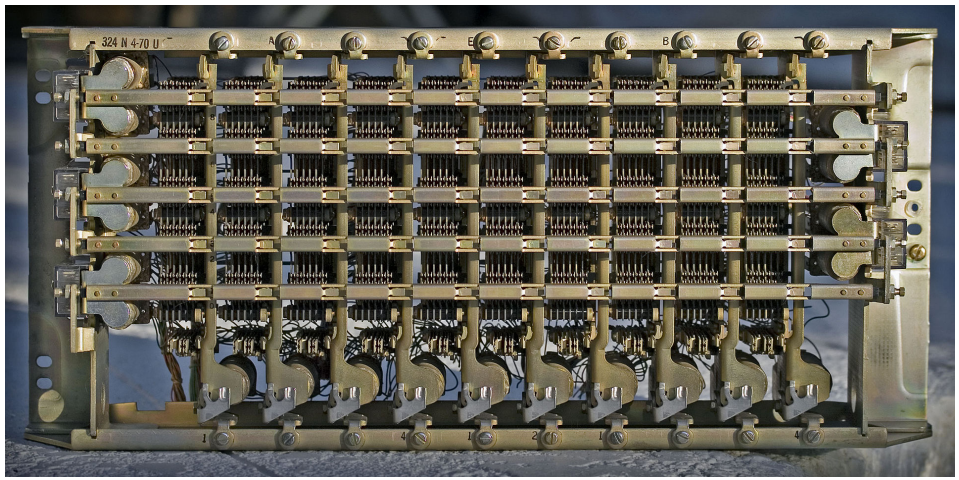
8-bit Bus Switch

Crossbar Switch

- Every node connected to all others (non-blocking)
- Good for small number of nodes
- Pros:
  - Low latency and high throughput
- Cons:
  - Expensive
  - Not scalable ⇨ $O(N^2)$ cost





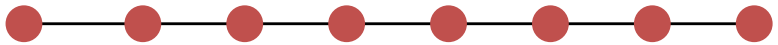


Western Electric 100-point six-wire Type B crossbar switch

Linear and Ring Topologies

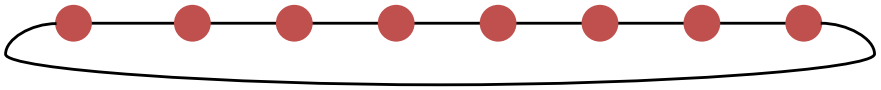
University of BRISTOL

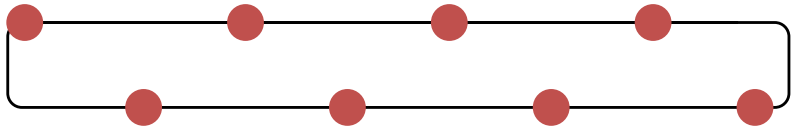
- Linear array (1D mesh)



  - # of nodes = **N**
  - Diameter = **N** -1; average distance ~ **N** /3.
  - Bisection bandwidth = 1 (in units of link bandwidth).

- Torus or Ring





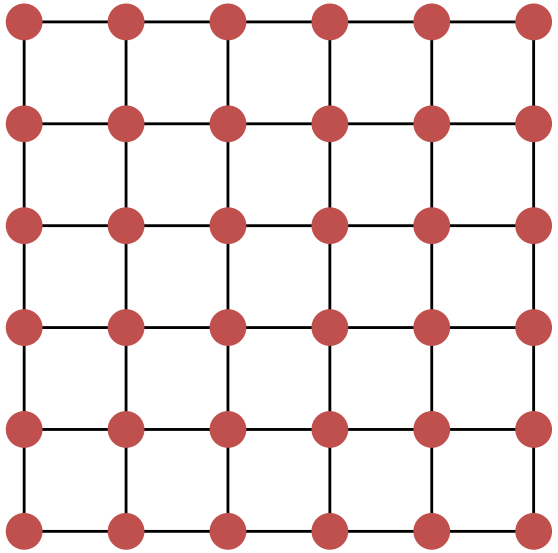
  - Effectively a repeated bus (linear array): multiple messages in transit.
  - Natural for algorithms that work with 1D arrays.
  - Bisection bandwidth = 2. Diameter = **N** /2; average distance ~ **N** /4.

Mesh and Torus (k-ary n-cube)

University of BRISTOL

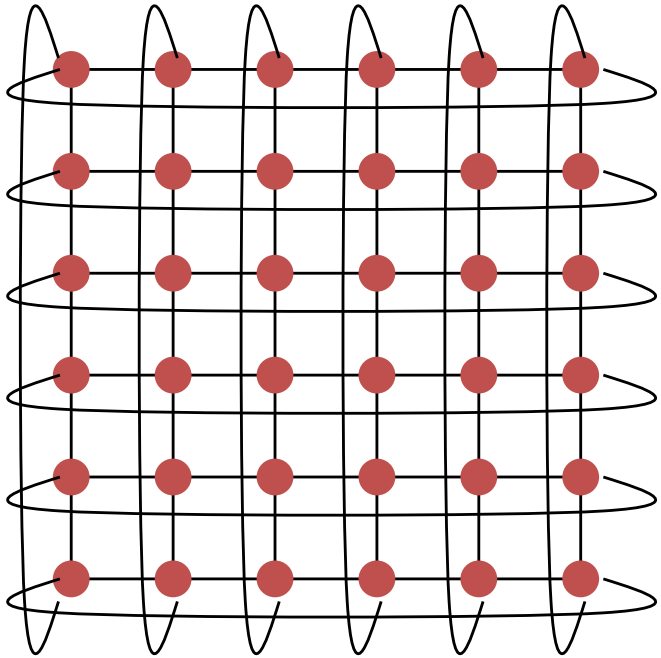
Two-dimensional mesh (k-ary 2-cube)

- # Nodes =N = k^2
- Diameter = 2 * (sqrt(N) – 1)
- Bisection bandwidth = sqrt(N)



2D Torus (k-ary 2-cube torus)

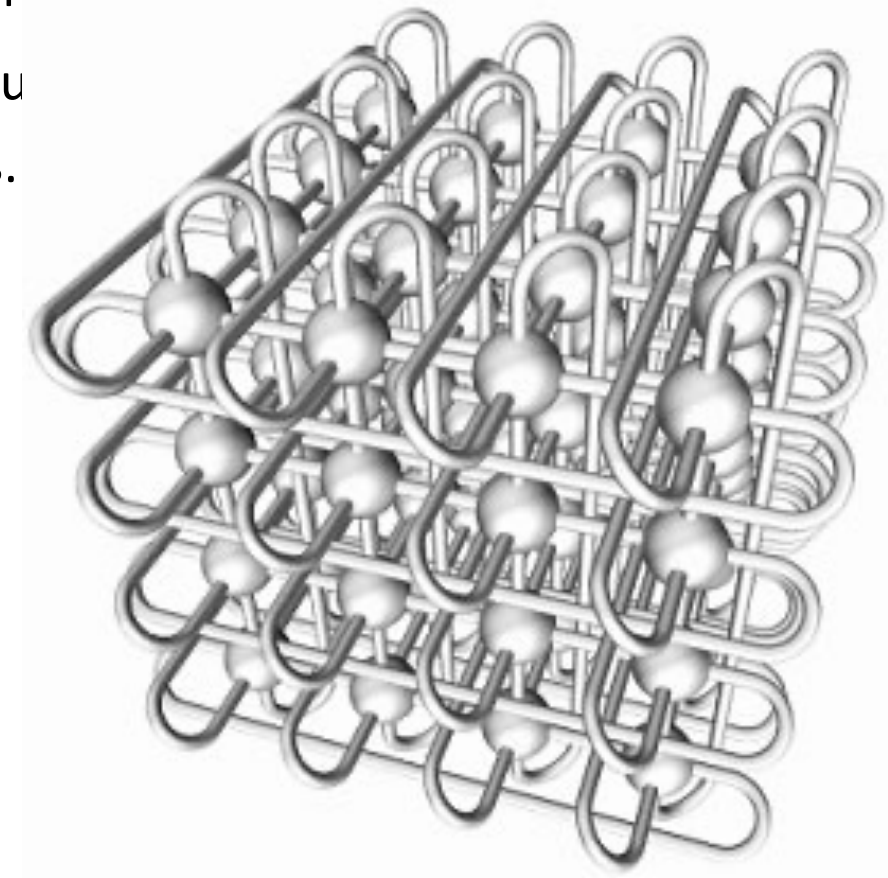
- # Nodes =N= k^2
- where k =nodes/dimension)
- Diameter = sqrt(N)
- Bisection bandwidth = 2* sqrt(N)



- Generalizes to higher dimensions (Cray T3D used 3D Torus).
- Natural for algorithms that work with 2D and/or 3D arrays.

3D Torus (4-ary 3-cube)

- In a 3D torus every compute node is connected to six neighbors
- Nodes located on the "edge" of the toru
  simply they use long wrap-around links.
  - # of nodes = **4^3=64**
  - Diameter = **6**;
  - Bisection bandwidth = 2*4^2=16



3D torus rendered by Fujitsu. (c) Fujitsu and RIKEN, 2009

K-ary *n*-cube Mesh/Torus

- Radix: describe both the number of input and output ports on the router, and the size or number of nodes in each dimension of the network.

$$r = 2n + 1$$

Dimension *n* is limited by practical packaging constraints. n=2 or n=3.

Torus vs. Mesh (k-ary n-cube)

University of BRISTOL

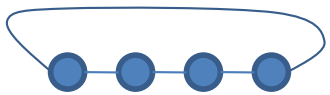
- Number of nodes $N = k^n$

	Mesh	Torus
Switch degree	2n	2n
Diameter	n(k-1)	nk/2
Bisection bandwidth	k^{n-1}	$2k^{n-1}$
Average routing distance	nk/3, k even n(k/3-1/3k), k odd	nk/4, k even n(k/4-1/4k), k odd

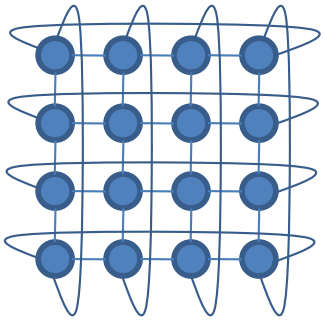
Question: Why mesh network still exists?

K-ary n-cube mesh/torus

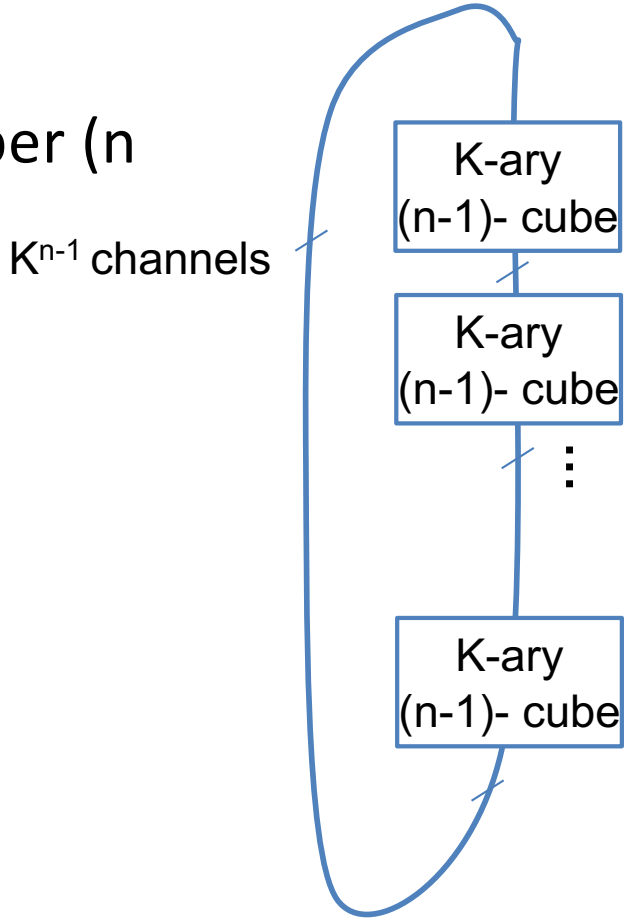
- Extended from binary (hypercube) to k-ary
- Each dimension has k elements
- Each node is identified by a k-based number (n digits).



4-ary 1-cube



4-ary 2-cube



Constructing a k-ary n-cube from k k-ary (n-1)-cubes

Radix and Dimension

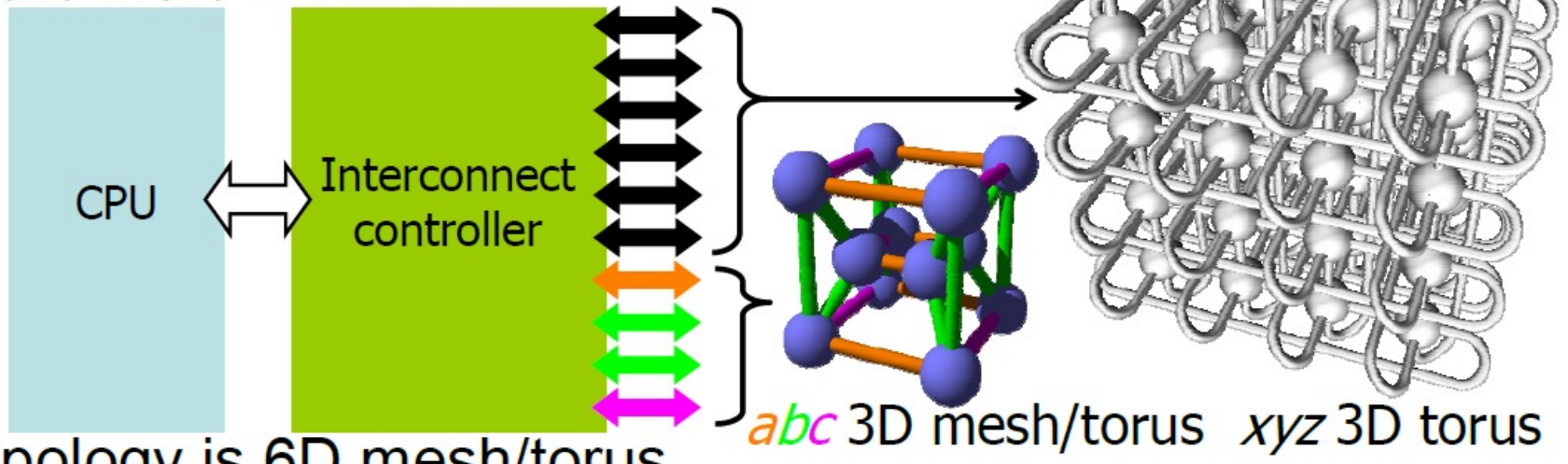
University of BRISTOL

- Direct networks
  - Radix: refers to the number of nodes within a dimension,
  - Network size can be further increased by adding multiple dimensions.
- Indirect networks
  - Radix: refers to the number of ports of a switch
  - Dimension: the number of stages in the network.

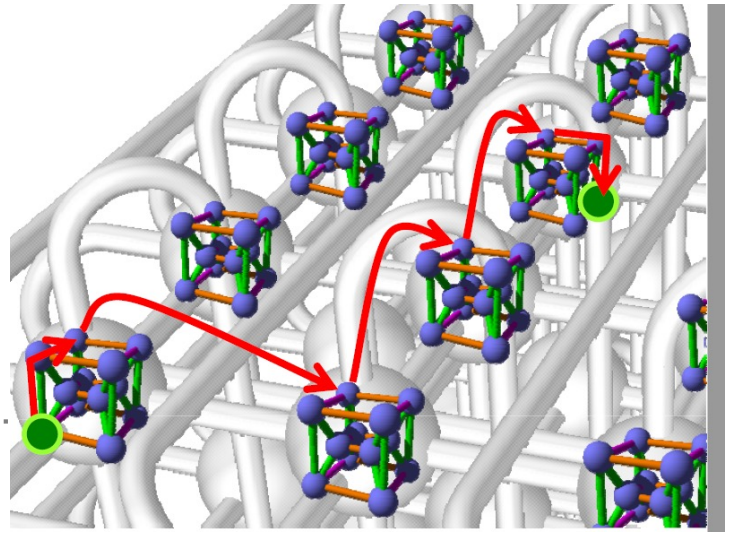
E.g. 6D Mesh/torus by Fujitsu

University of BRISTOL

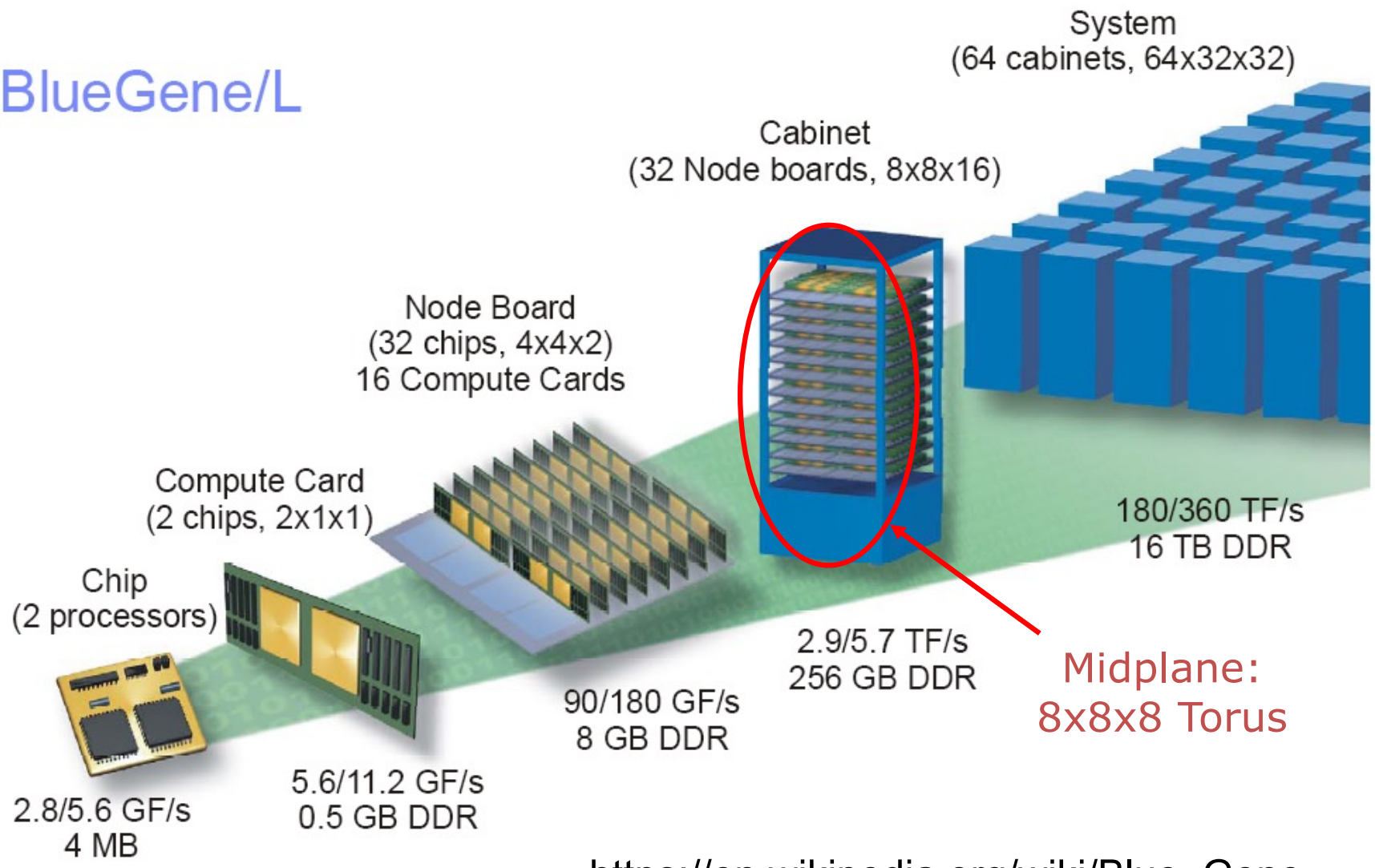
- 6 links ⇨ Scalable *xyz* 3D torus
- 4 links ⇨ Fixed size *abc* 3D mesh/torus
  - |*a*|=2, |*b*|=3, |*c*|=2 ⇨ 12 nodes



- Total topology is 6D mesh/torus
  - Cartesian product of *xyz* and *abc* mesh/torus



Torus application

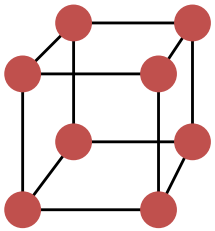


https://en.wikipedia.org/wiki/Blue_Gene

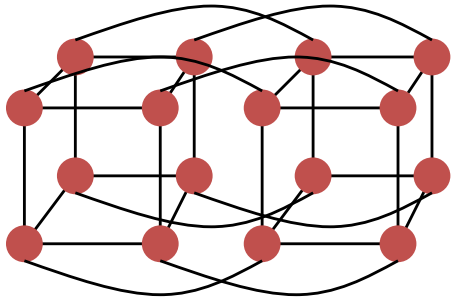
K-ARY n-CUBE Hypercubes

University of BRISTOL

- Number of nodes N = 2^d for dimension d (d= $\log_2 N$).
  - Diameter = d.
  - Bisection bandwidth = 2^{d-1}. (good)
  - Degree: n = $\log_2$ N=d
  - Cost=No of links: [P x log2(P)]/2 = [2^d x d]/2

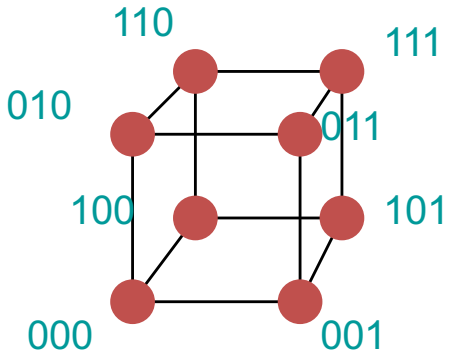


3d

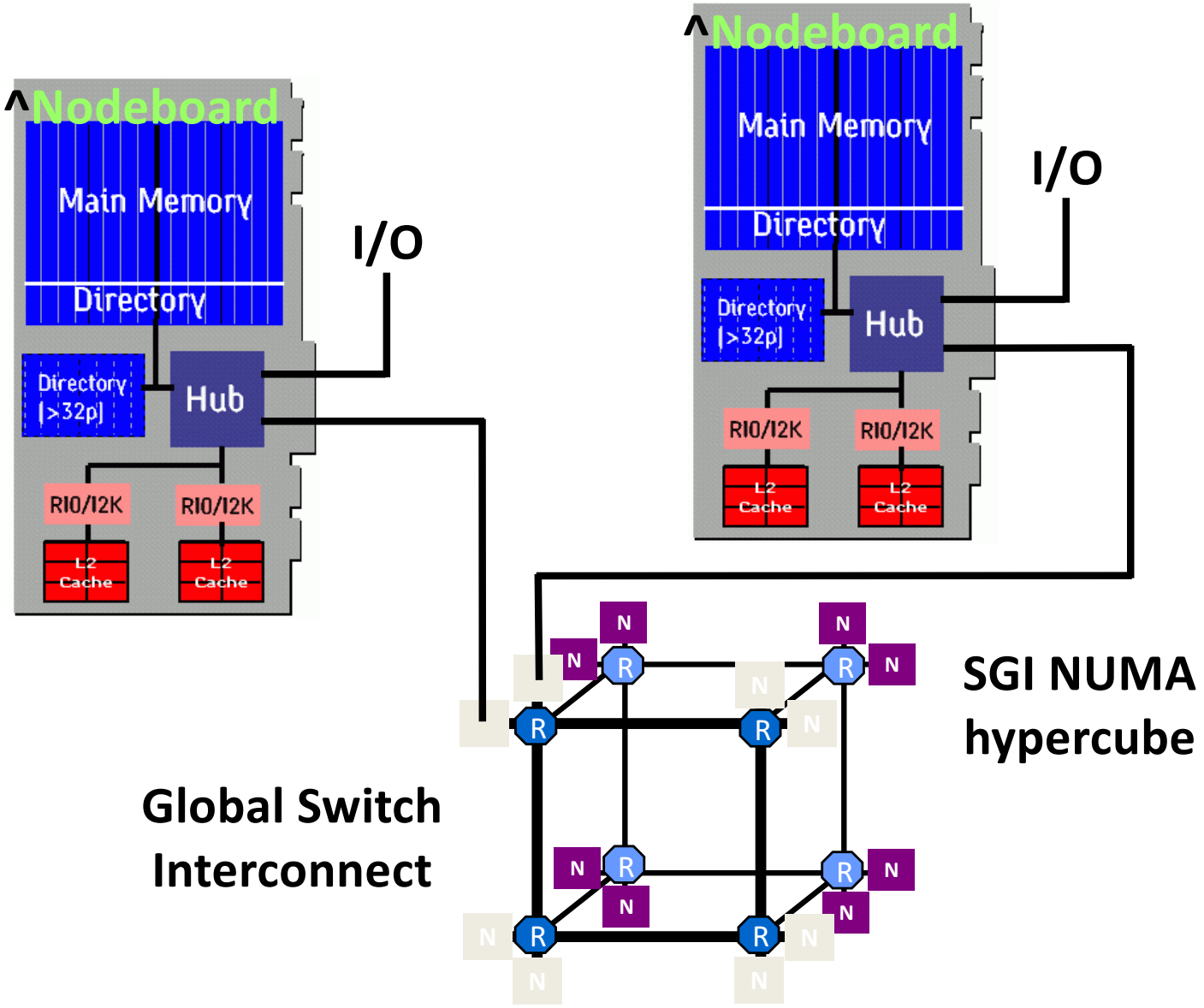


4d

- Popular in early machines (Intel iPSC, NCUBE).
  - Lots of clever algorithms.
- Grey code addressing:
  - Nodes labeled as n-bit binary
  - Hamming distance:
    - $D_H(x, y)$ = # of positions in which *x* & *y* differ
    - *x* & *y* are binary vectors
    - $D_H(1100, 0111) = 3$



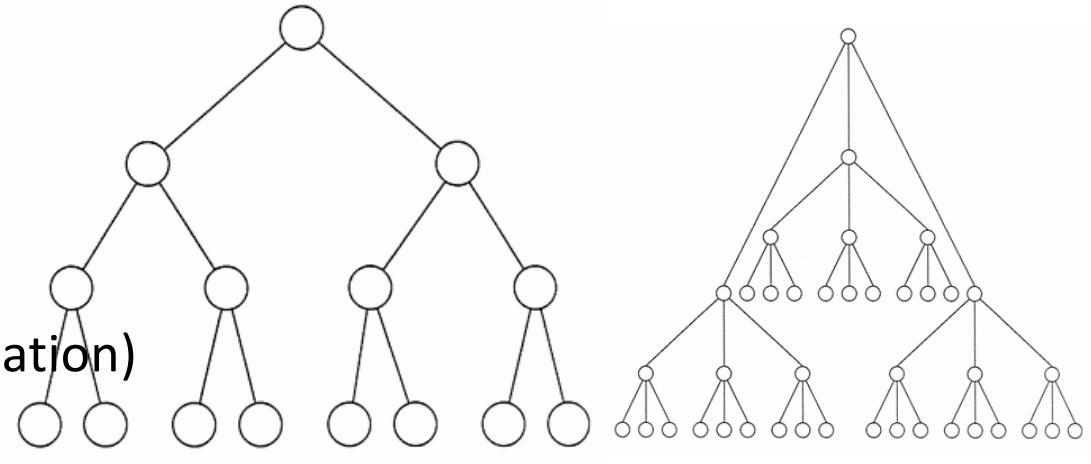
Origin SGI NUMA Architecture



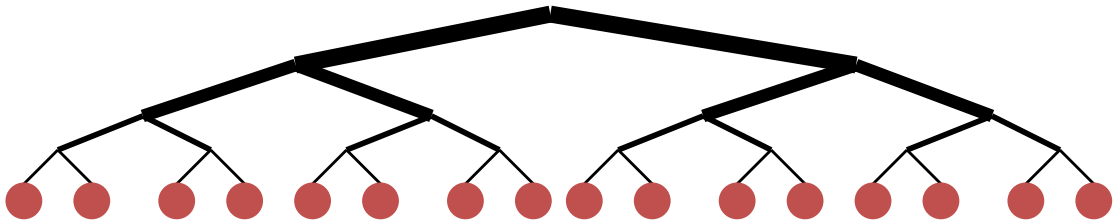
Trees

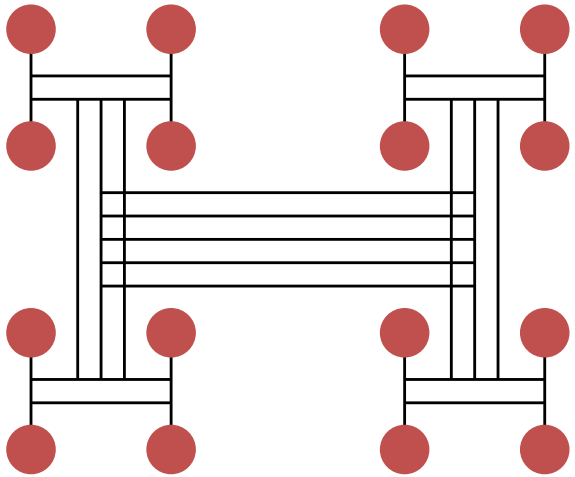
University of BRISTOL

- Diameter = $\log_2 N$ for binary trees.
- Bisection bandwidth = 1
- Easy layout as planar graph
- Many tree algorithms (e.g., summation)



- **Fat trees** avoid bisection bandwidth problem:
  - More (or wider) links near top
  - Example: Thinking Machines CM-5

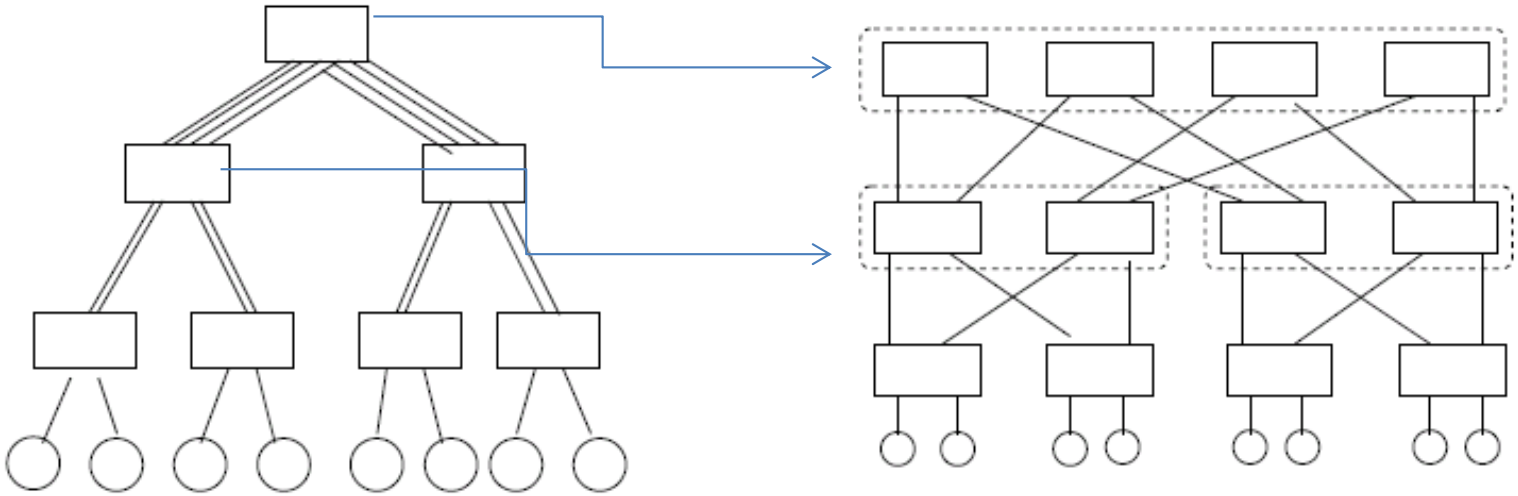




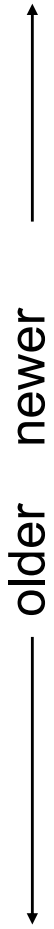
Practical Fat-trees

University of BRISTOL

- Use smaller switches to approximate large switches.
  - Connectivity is reduced, but the topology is not implementable
  - Most commodity large clusters use this topology. Also call constant bisection bandwidth network (CBB)



Topologies in Real Machines



Red Storm (Opteron + Cray network, future)	**3D Mesh**
Blue Gene/L	**3D Torus**
SGI Altix	**Fat tree**
Cray X1	**4D Hypercube***
Myricom (Millennium)	**Arbitrary**
Quadrics (in HP Alpha server clusters)	**Fat tree**
IBM SP	**Fat tree (approx)**
SGI Origin	**Hypercube**
Intel Paragon (old)	**2D Mesh**
BBN Butterfly (really old)	**Butterfly**

Many of these are approximations:
E.g., the X1 is really a "quad bristled hypercube" and some of the fat trees are not as fat as they should be at the top

Simulation Software

University of BRISTOL

- Book:Principles and Practices of Interconnection Networks
- Simulation software: Booksim2

  https://github.com/booksim/booksim2

Summary

University of BRISTOL

- Introduce Network Terminology
- Understanding network topologies