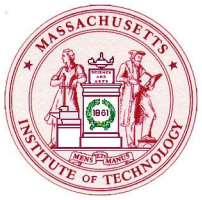
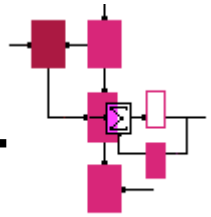


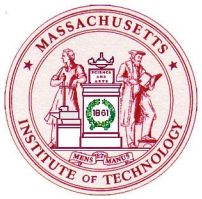


Switches, Routers and Networks

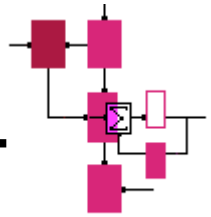
Muriel Medard

EECS

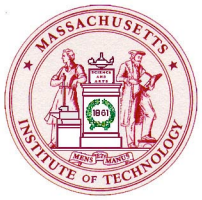
MIT



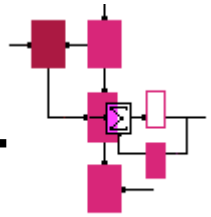
Overview



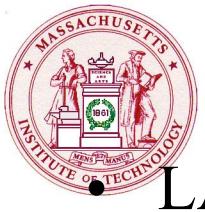
- Introduction
- Routing and switching:
  - Switch fabrics :
  - Basics of switching
  - Blocking
  - Interconnection examples
  - Complexity
  - Recursive constructions
- Interconnection routing
- Buffering - input and output
- Local area networks (LANs)
- Metropolitan area networks (MANs)
- Wide area networks (WANs)
- Trends



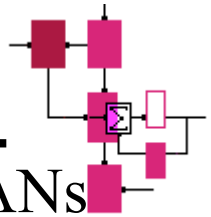
Introduction



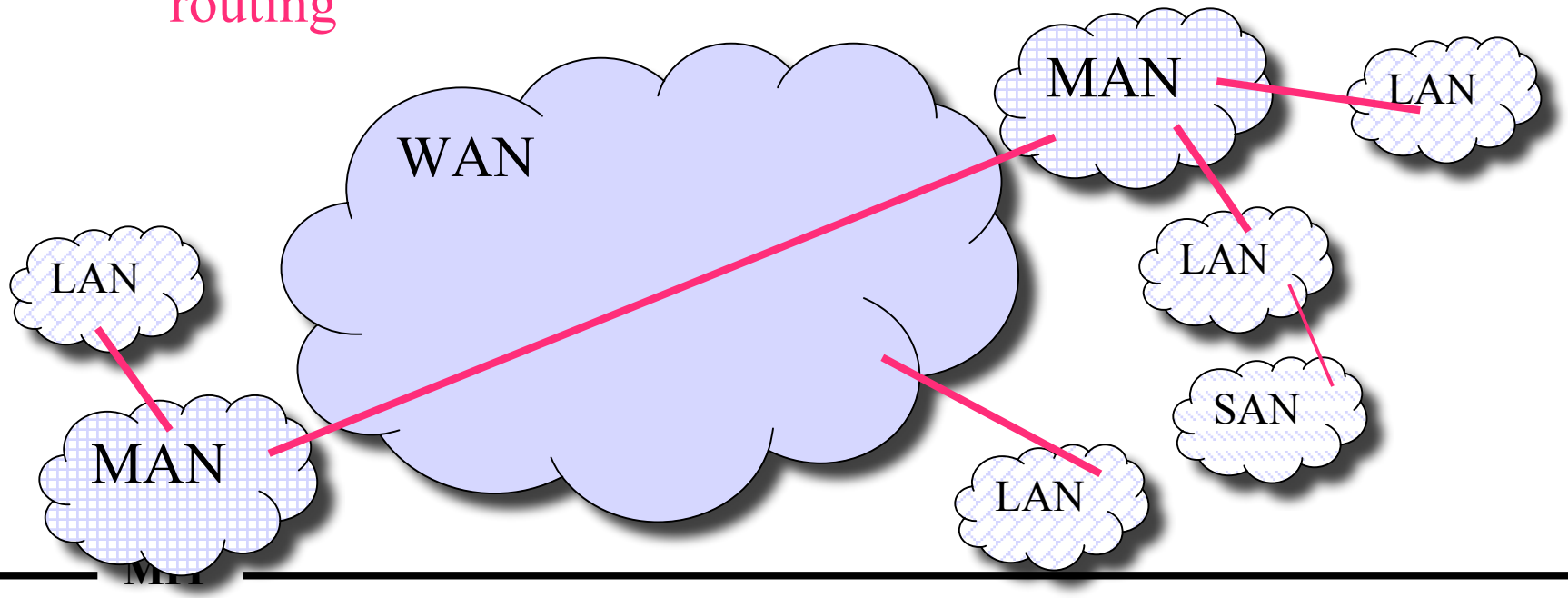
- Data networks generally evolve fairly independently for different applications and are then patched together – telephony, variety of computer applications, wireless applications
- IP is a large portion of the traffic, but it is carried by a variety of protocols throughout the network
- Voice is still the application that has determined many of the implementation issues, but its share is decreasing and voice is increasingly carried over IP (voice over IP)
- Voice-oriented networks are not very flexible, but are very robust
- IP very successful because it is very flexible, but increasingly there is a drive towards enhancing the reliability of services
- How do all of these network types and requirements fit together?

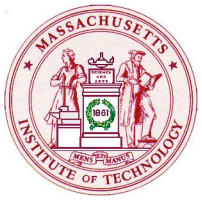


Networks

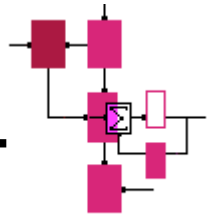


- LANs serve a wide variety of services and attach to MANs or maybe directly to WANs
- The two main purposes of a networks are:
  - Transmission across some distance: this involves amplification or regeneration (generally code-assisted)
  - The establishment of variable flows: switching and routing

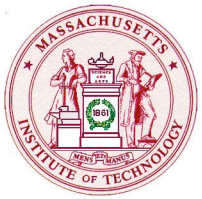




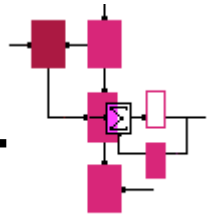
Switching and Routing

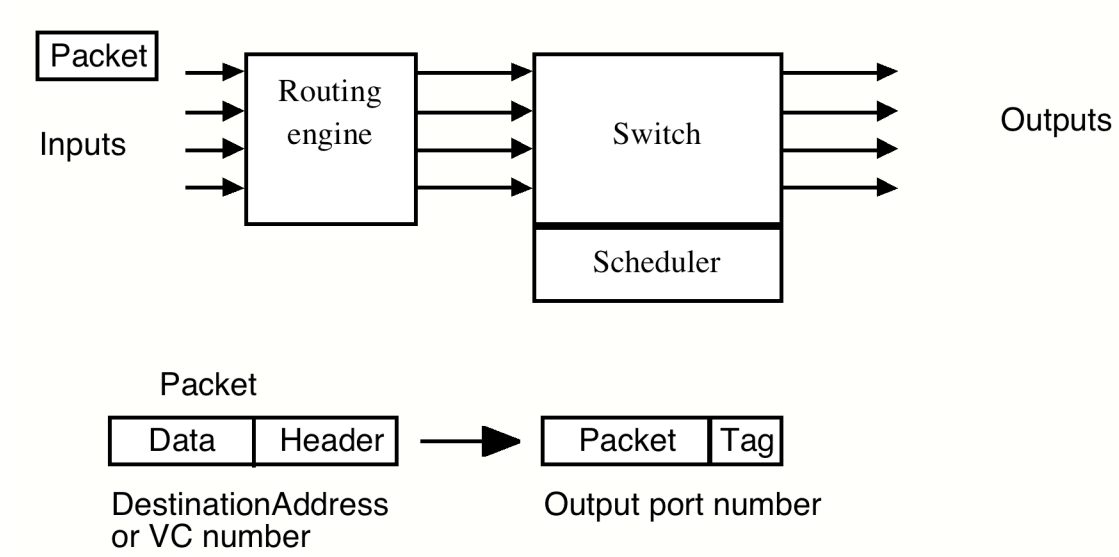


- Switching is generally the establishment of connections on a circuit basis
- Routing is generally the forwarding of traffic on a datagram basis
- Routing requires switching but not vice-versa – routing uses connections which are permanently or temporarily set up to in order to forward datagrams (those datagrams may be in circuit form, for instance VPs and VCs)

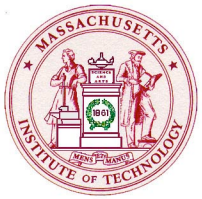


Packet routers

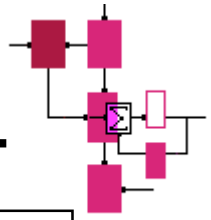


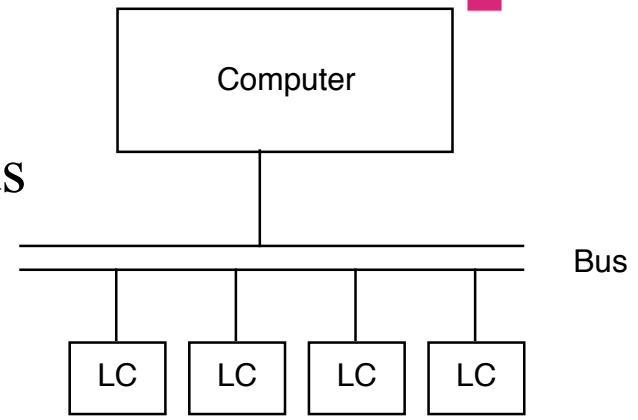


- A packet switch consists of a routing engine (table look-up), a switch scheduler, and a switch fabric.
- The routing engine looks-up the packet address in a routing table and determines which output port to send the packet.
  - Packet is tagged with port number
  - The switch uses the tag to send the packet to the proper output port

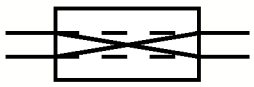


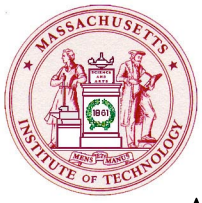
Switch fabrics



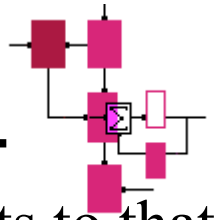


- Simplest switch fabric is simply a shared bus
  - Most of the processing is done in line cards
    - Route table look-up
    - Line cards buffer the packets
    - Line card send packets to proper output
  - Bus bandwidth must be N times LC speed (N ports)
- In general a switch fabric replaces the bus
- Switch fabrics are created from certain building blocks of smaller switches arranged in stages
- Simplest switch is a 2x2 switch, which can be either in the through or crossed position

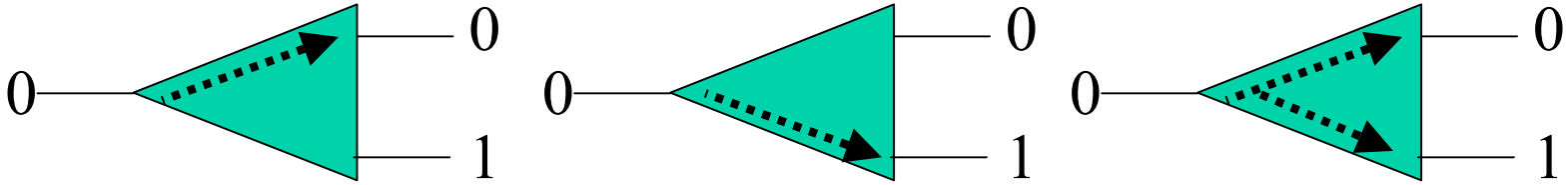




Definitions

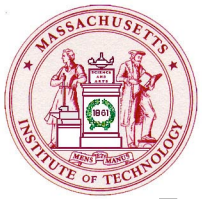


- A connection state is a mapping from the array of inputs to that of outputs; connections are either point-to-point or multicast
- Basic switch building blocks are:
  - the distributor

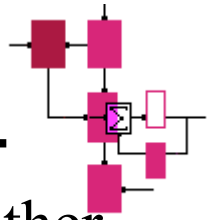


  - the concentrator

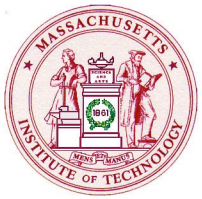
  - the 2x2 2-state point-to-point switch (switching cell)



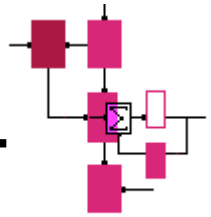
Building up



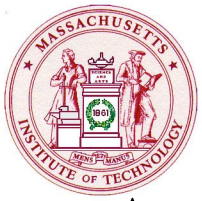
- Interconnection network: finite collection of nodes together with a set of interconnection lines such that
  - every node is an object with an array of inputs and an array of outputs
  - an interconnection line leads from an output of one node to an input of another node
  - every I/O of a node is incident with at most one interconnection line
  - an I/O is called external if it is not incident with any interconnection line
- A route from an external input to an external output is a chain of distinct $(a_0, b_0, a_1, b_1, \ldots, a_k, b_k)$ where a_0 and b_k are external, b_{j-1} is interconnected to a_j



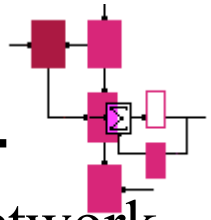
Building up



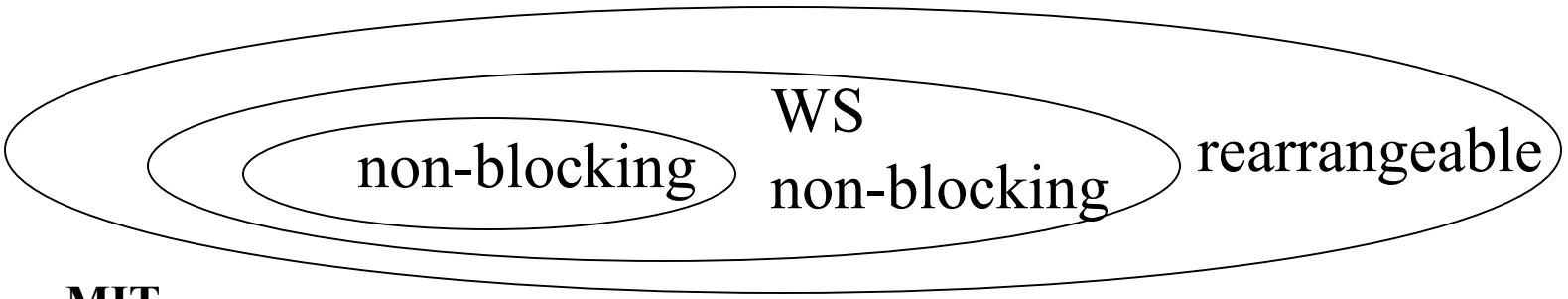
- An interconnection network is called a switching network when:
  - every node qualifies to be a switch through proper specification of connection states
  - the network is routable (there exists a route from every external input to every external output)
  - an ordering is specified on external inputs and on external outputs
- Unique routing interconnection networks: all routes from an external input to an external output are parallel, that is (a_0, b_0, a_1, b_1, …, a_k, b_k) and (a_0, b'_0, a'_1, b'_1, …, a'_k, b_k) are such that a_j, a'_j reside on the same nodes and b_j, b'_j reside on the same node
- Otherwise: alternate routing

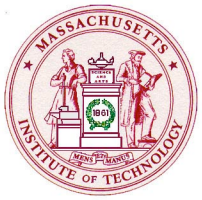


Blocking

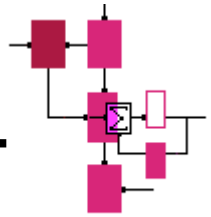


- A mxn unique routing network is called a nonblocking network if for any integer k < min(m,n)+1, any k external inputs, any k external outputs and pairing between these external I/O, there exist k disjoint routes for the matched pairs
- For a routable network, the same property is that ot a rearrangeably nonblocking, or rearrangeable network
- An interconnection network is strictly non-blocking if requests for routes are always granted under the rule of arbitrary route selection, wide-sense non-blocking if there exists an algorithm for route selection that grants all requests





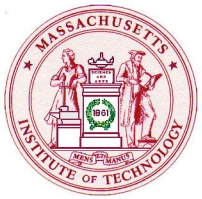
Blocking, Multi-stage networks



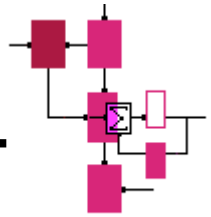
- Main connection between rearrangeability and non-blocking property is given by the following theorem:

  A switching network composed of non-blocking switches is rearrangeable iff it constructs a non-blocking switch

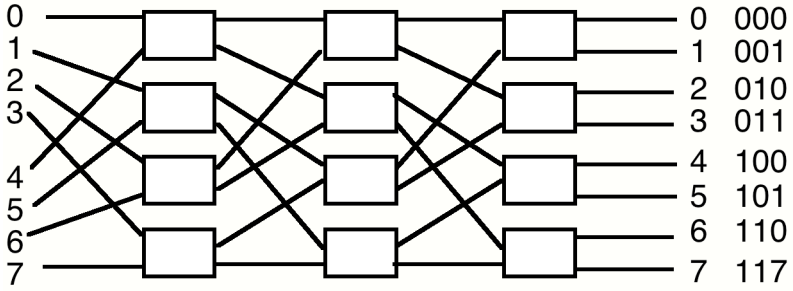
- A common means of building interconnection networks is to use a multi-stage architecture:
  - every interconnection line is between two stages
  - every external input is on a first-stage node
  - every external output is on a final-stage node
  - nodes within each stage are linearly ordered



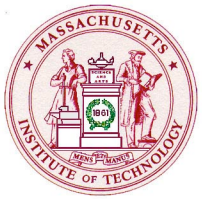
Interconnection networks



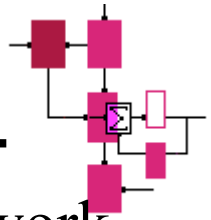
- N input, Log(N) stages with N/2 modules per stage
  Example: Omega (shuffle exchange network)



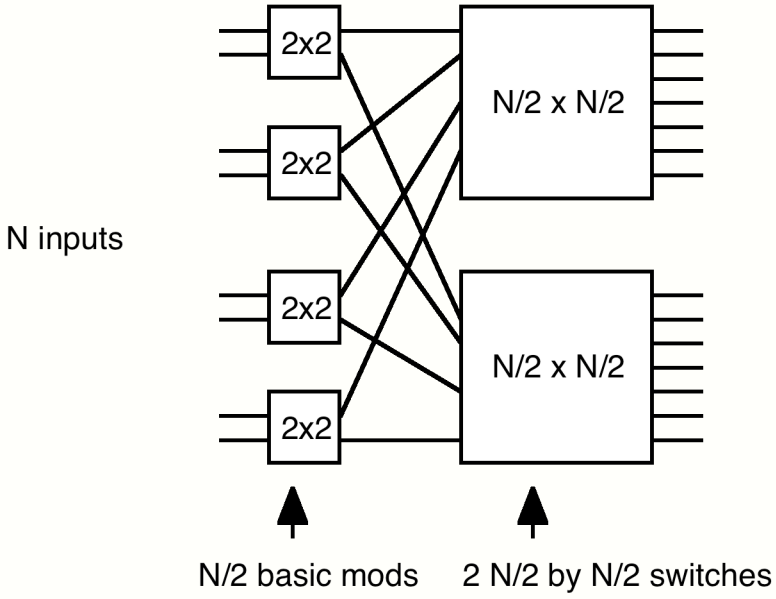
- Notice the order of inputs into a stage is a shuffle of the outputs from the previous stage: (0,4,1,5,2,6,3,7)
- Easily extended to more stages
- Any output can be reached from any input by proper switch settings
  - Not all routes can be done simultaneously
  - Exactly one route between each OD pair

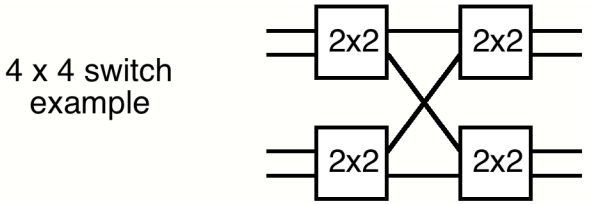


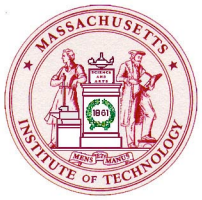
Interconnection networks



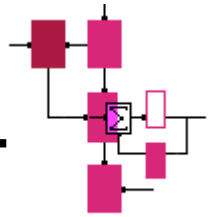
- Another example of a multi-stage interconnection network
- Built using the basic 2x2 switch module
- Recursive construction
  - Construct an N by N switch using two N/2 by N/2 switches and a new stage of N/2 basic (2x2) modules
  - N by N switch has $Log_2(N)$ stages each with N/2 basic (2x2)



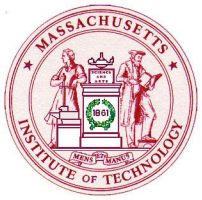




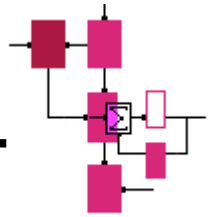
Complexity issues



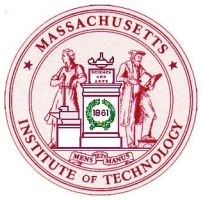
- There are many different parameters that are used to consider the complexity of an interconnection network
- Line complexity: number of interconnection lines
- Node (cell) complexity: number of small nodes (mxn where $m < 3$ and $n < 3$)
- Depth: maximum number of nodes on a route (assuming an acyclic interconnection network)
- Entropy of a switch: log of the number of connections states
- What relations exist between complexity and the capabilities of a switch?



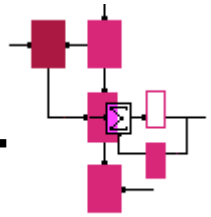
Complexity



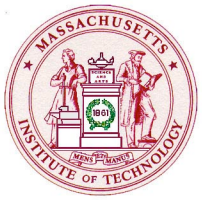
- The depth of a mxn routable interconnection network is at least max(log(m), log(n)).
- Proof: for a depth d, there are at most 2^d external outputs. Since we have routability, $n < 2^d+1$ and $m < 2^d+1$.
- When a switching network is composed of 2-state switches, the component complexity of the network is at least the entropy of the switch
- Proof: for E the number of switches, there are 2^E ways to form a combination of one connection state in every node. Each combination corresponds to at most one connection state in the node.



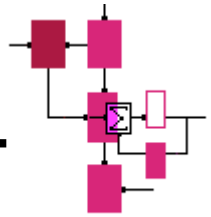
Complexity



- When a nxn rearrangeable network is composed of small nodes, its component complexity is at least log(N!)
- Proof: if we take every small node to be replaced by a 2-state point-to-point switch, then we have a non-blocking switch. Thus, there is a different connection state for everyone of the n! one-to-one mapping between the n inputs and the n outputs. We now use the relation for networks composed of 2-state switches.
- Note: using Stirling's formula, we can obtain an approximate simple bound for component complexity



Complexity



- Component complexity:

$$n! \approx \sqrt{2\pi\, n}\left(\frac{n}{e}\right)^n$$

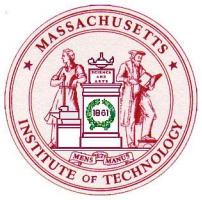
$$\Rightarrow \log(n!) = n\log(n) - 1.44n + \log\left(\frac{n}{2}\right) + \frac{\log(2\pi)}{2}$$

so component complexity is bounded from below by

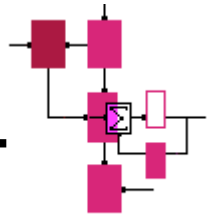
$$n\log(n) - 1.44n + \Omega(\log(n))$$

- Relation between line and component complexity:

component complexity +mn = line complexity +m + n



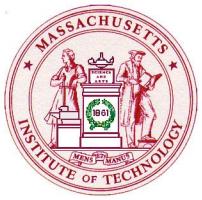
Complexity



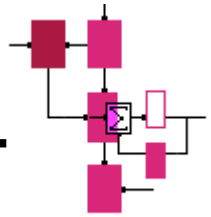
- If a mxn nonblocking network is composed of n_{12} 1x2 nodes, n_{21} 2x1 nodes, n_{22} cells, plus possibly crosspoints (edges), then

  $n_{12} + n_{21} + 4\, n_{22} = 2mn - m - n$

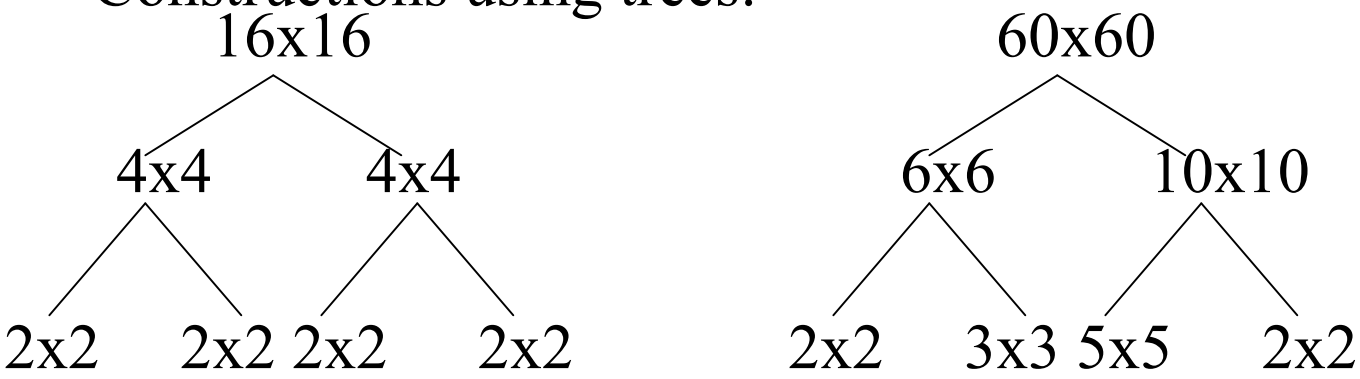
- Corollary: a nxn non-blocking network composed of small nodes has component complexity at least $0.5(n^2 - n)$
- Note: directed acyclic graphs can be seen as a special case of a network - a crosspoint network.
- We have basic complexity properties, but how do we build networks?



Recursive 2-stage construction

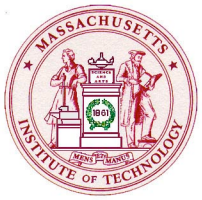


- 2-stage interconnection with parameters m and n is composed of n mxm input nodes and m nxn output nodes interconnected by a coordinate interchange (static)
- Constructions using trees:

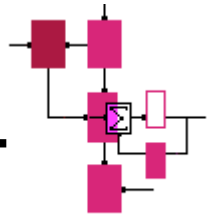


Divide and conquer

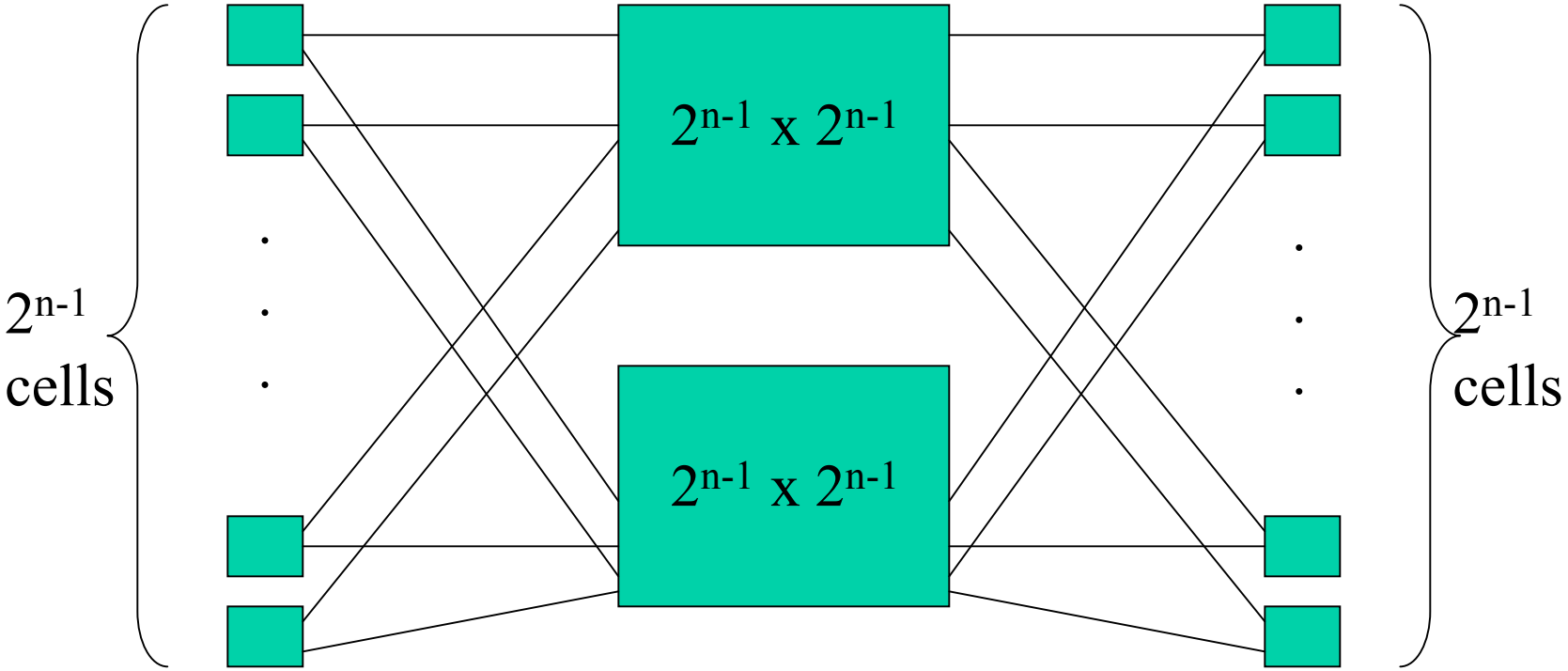
- Basic blocks need not be 2x2, trees need not be balanced

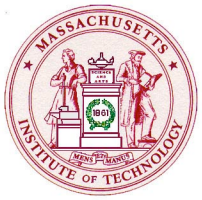


Benes approach

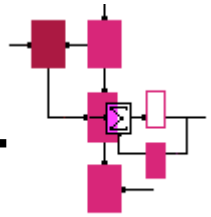


- A three stage approach in which we use as the middle stage two networks of size 2^{n-1} x 2^{n-1} to build a network of size 2^n x 2^n





Generalized 3-stage approach

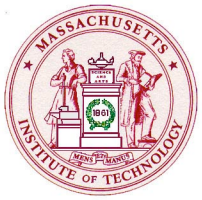


- We denote by [nxm, rxp, mxq] the 3-stage network with r nxm input nodes, m rxp middle nodes, p mxq output nodes such that
  - output y of input node x is linked to input x of of middle node y
  - output u of middle node y is linked to input y of output node u
- Rearrangeability theorem: the 3-stage network is rearrangeable iff

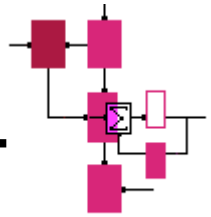
$$m \geq \min\{\max\{n, q\}, nr, pq\}$$

- It is strictly non-blocking iff

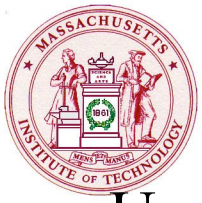
$$m \geq \min\{n + q - 1, nr, pq\}$$



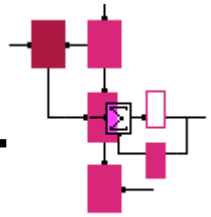
Maximum matchings



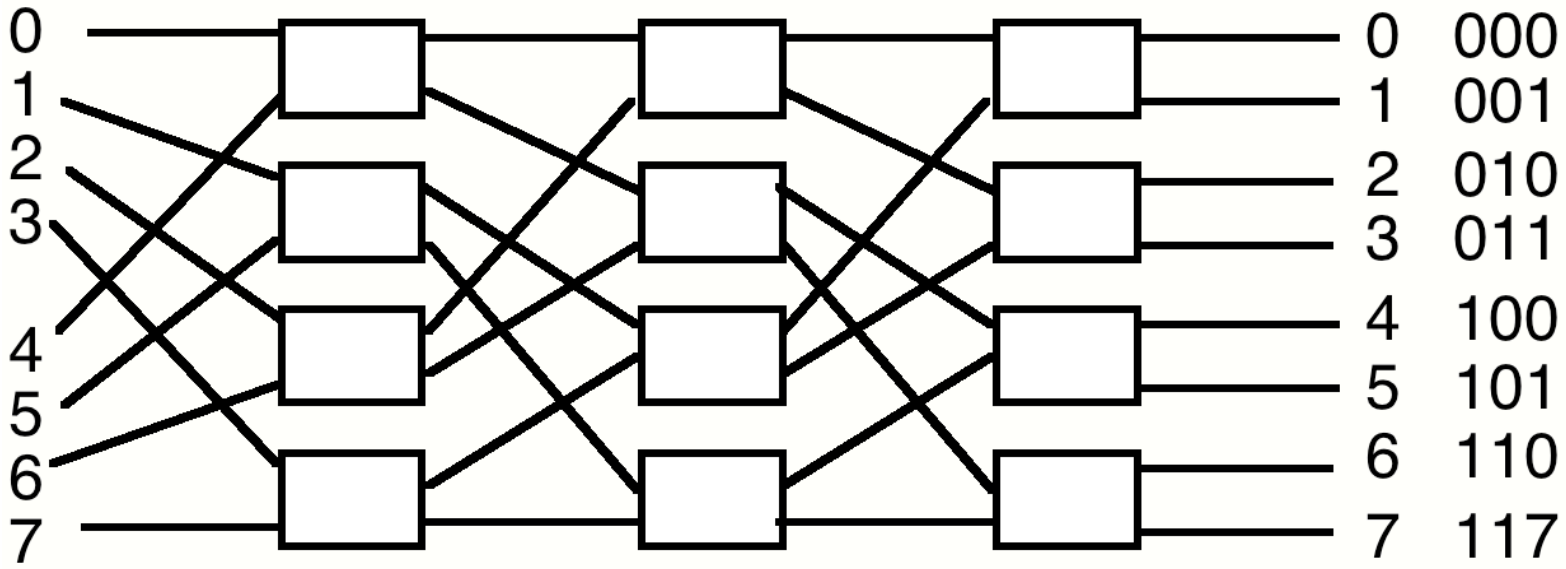
- Algorithms for finding maximum matching exist
- The best known algorithms takes $O(N^{2.5})$ operations
  - Too long for large N
- Alternatives
  - Sub-optimal solutions
  - Maximal matching: A matching that cannot be made any larger for a given backlog matrix
  - For previous example:
    (1-1,3-3) is maximal
    (2-1,1-2,3-3) is maximum
- Fact: The number of edges in a maximal matching $\geq 1/2$ the number of edges in a maximum matching

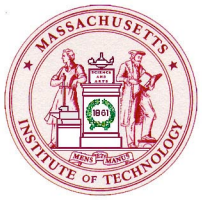


Self-routing

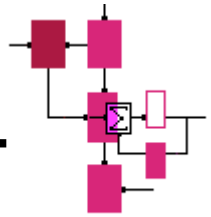


- Use the switch fabric for packet routing
- Use a tag: n bit sequence with one bit per stage of the network
  - E.g., Tag = $b_3b_2b_1$
- Module at stage i looks at bit i of the tag (b_i), and sends the packet up if b_i=0 and down if b_i=1
- In omega network, for destination port with binary address abc the tag is cba
  - Example: output 100 => tag = 001
  - Notice that regardless of input port, tag 001 will get you to output 100
- What happens when packets cannot be forwarded to the right output for the given setting of the switching fabric?

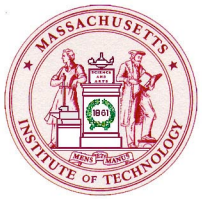

0
1
2
3
4
5
6
7
0 000
1 001
2 010
3 011
4 100
5 101
6 110
7 117



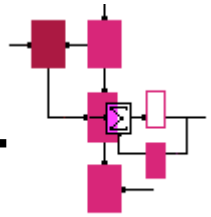
Interconnection analysis for routing



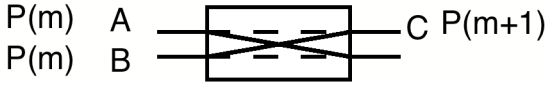
- Assume no buffering at the switches
- If two packets want to use the same port one of them is dropped
- Suppose switch has m stages
- Packet transmit time = 1 slot (between stages)
- New packet arrival at the inputs, every slot
  - Saturation analysis (for maximum throughput)
  - Uniform destination and distribution independent from packet to packet



Interconnection throughput



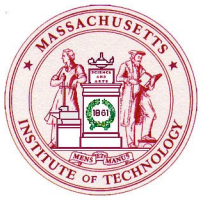
- Let P(m) be the probability that a packet is transmitted on a stage m link, P(0) = 1



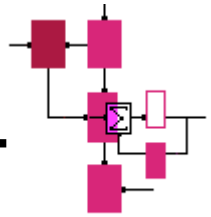
- P(m+1) = 1 – P(no packet on stage m+1 link (link c))
  = 1 – P(neither inputs to stage m+1 chooses this output)
- Each input has a packet with probability P(m) and that packet will choose the link with probability 1/2. Hence,

$$P(m+1) = 1 - (1 - \frac{1}{2}P(m))^2$$

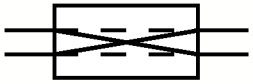
- We can now solve for P(m) recursively
- For an m stage network, throughput (per output link) is P(m), which is the probability that there is a packet at the output



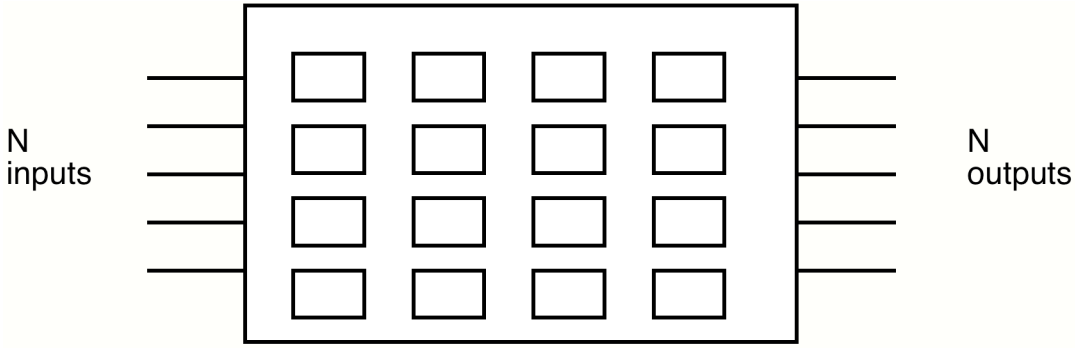
Distributed buffer

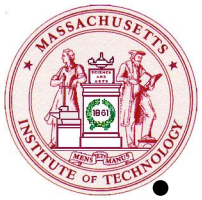


- Modular Architecture

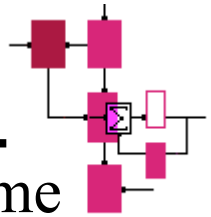


- Switch buffers: None, at input, or at output of each module
  Switch fabric consists of many 2x2 modules

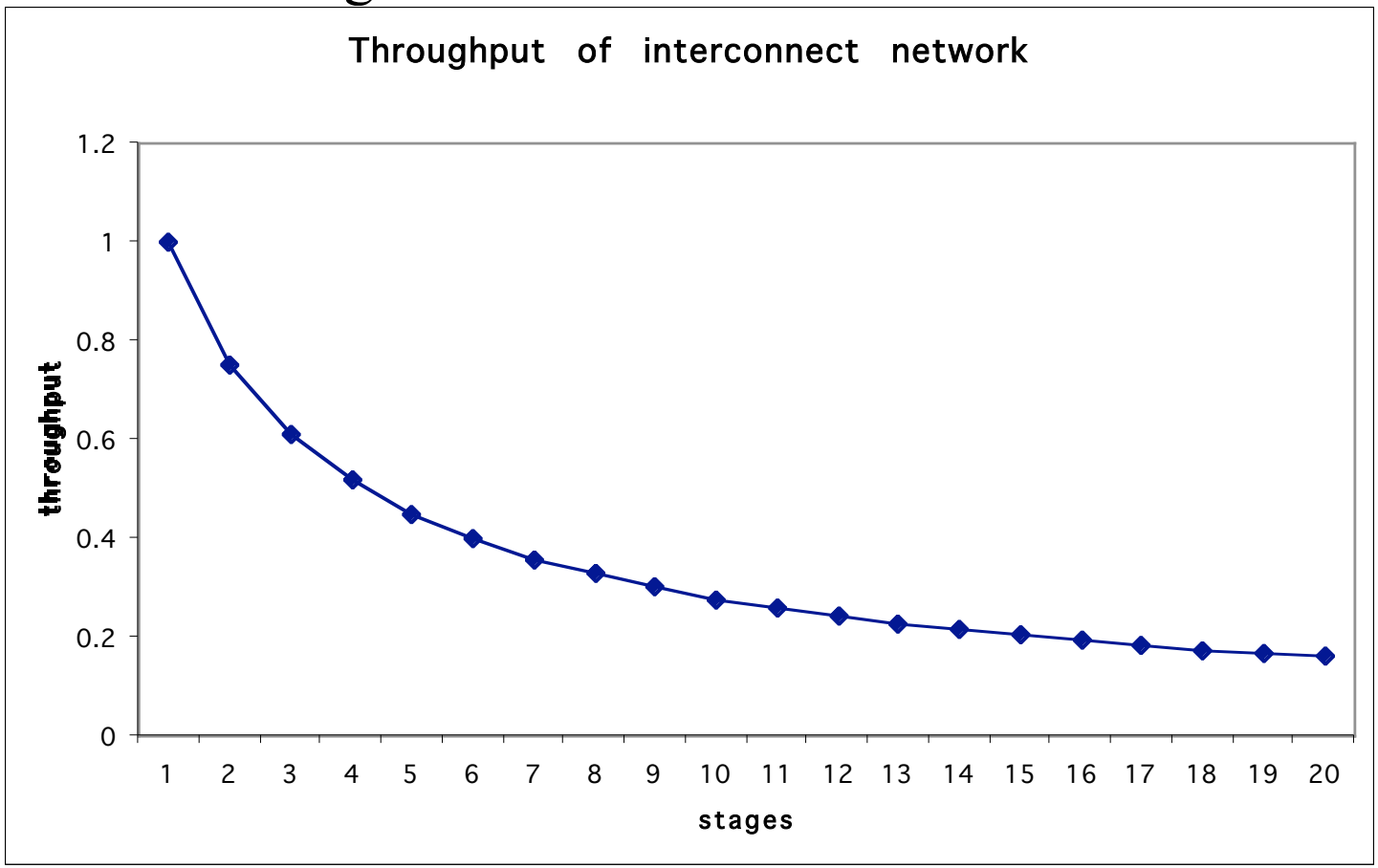


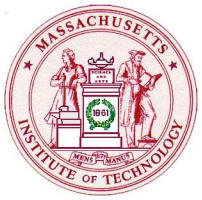


Contention and buffering

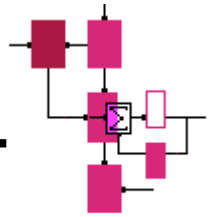


- Two packets may want to use the same link at the same time (same output port of a module): hot spot effect
- Solution: Buffering

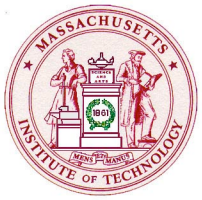




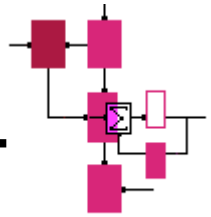
Multi-stage architecture

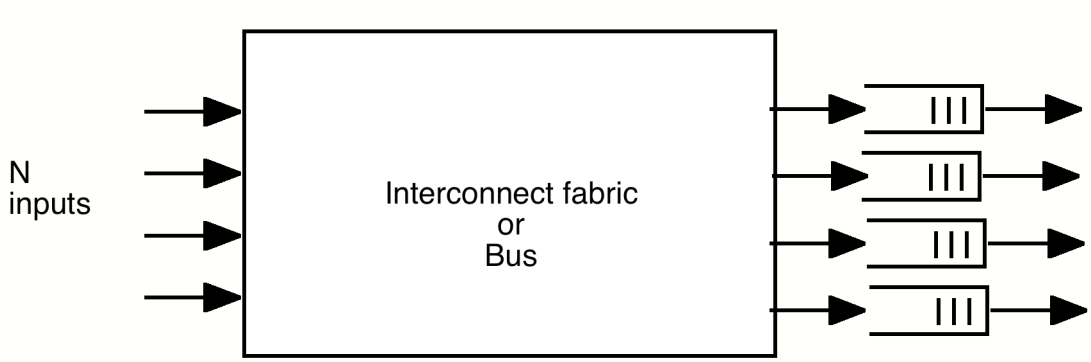


- Throughput is significantly improved by buffers at the stages
  - Buffers increase delay
  - Tradeoff between delay and throughput
- Advantages: modular, scalable, bus (links) only needs to be as fast as the line cards
- Disadvantages
  - Delays for going through the stages
    - Cut-through possible when buffers empty
  - Decreased throughput due to internal blocking
- Alternatives: Buffers that are external to the switch fabric
  - Output buffers
  - Input buffers

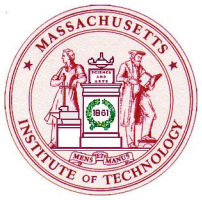


Output buffer architecture

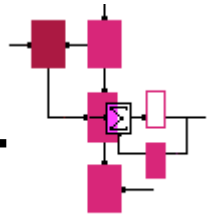




- As soon as a packet arrives, it is transferred to the appropriate output buffer
- Assume slotted system (cell switch)
- During each slot the switch fabric transfers one packet from each input (if available) to the appropriate output
  - Must be able to transfer N packets per slot
  - Bus speed must be N times the line rate
  - No queueing at the inputs
    - Buffer at most one packet at the input for one slot



Queueing Analysis



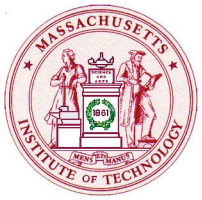
- If external arrivals to each input are Poisson (average rate $\overline{A}$), each output queue behaves as an M/D/1 queue

$$\overline{X} = \overline{X^2} = 1$$

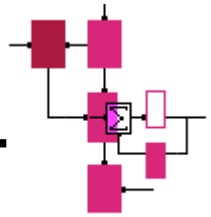
  - packet duration equaling one slot
- The average number of packets at each output is given by (M/G/1 formula):

$$N_Q = \frac{2\overline{A} - (\overline{A})^2}{2(1 - \overline{A})}$$

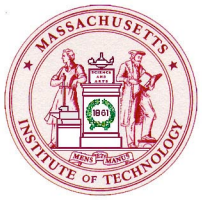
- Note that the only delay is due to the queueing at the outputs and none is due to the switch fabric



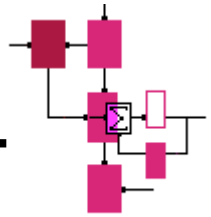
Advantages/Disadvantages of Output buffer architecture



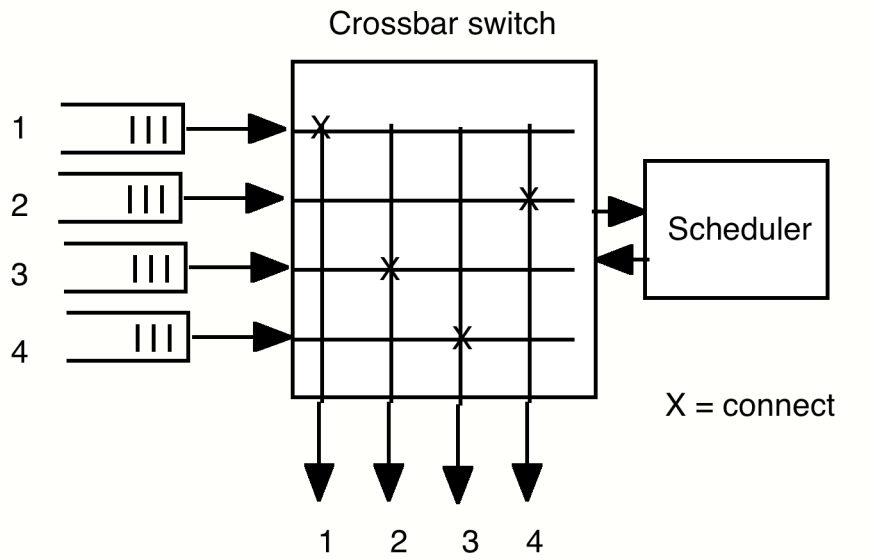
- Advantages: No delay or blocking inside switch
- Disadvantages:
  - Bus speed must be N times line speed
    - Imposes practical limit on size and capacity of switch
- Shared output buffers: output buffers are implemented in shared memory using a linked list
  - Requires less memory (due to statistical multiplexing)
  - Memory must be fast

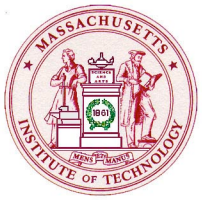


Input buffer

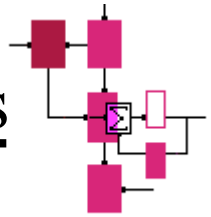


- Packets buffered at input rather than output, so switch fabric does not need to be as fast
- During each slot, the scheduler established the crossbar connections to transfer packets from the input to the outputs
  - Maximum of one packet from each input
  - Maximum of one packet to each output

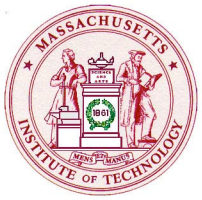




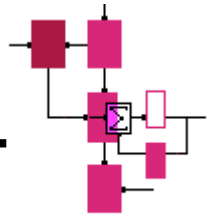
Throughput analysis of input queued switches



- Head of line (HOL) blocking – when the packets at the head of two or more input queues are destined to the same output, only one can be transferred and the others are blocked
- HOL blocking limits throughput because some inputs (consequently outputs) are kept idle during a slot even when they have other packet to send in their queue
- Consider an NxN switch and again assume that inputs are saturated (always have a packet to send)
- Uniform traffic => each packet is destined to each output with equal probability (1/N)
- Now, consider only those packets at the head of their queues (there are N of them!)



Throughput analysis, continued



- Let Q_m^i be the number of HOL packets destined to node i at the end of the m[th] slot $Q_m^i = \max(0, Q_{m-1}^i + A_m^i - 1)$

- Where

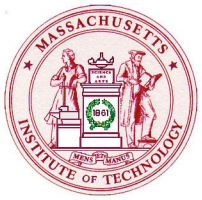
A_m^i = number of new HOL messages addressed to node i that arrive to the HOL during slot m. Now,

$$P(A_m^i = l) = \binom{C_{m-1}}{l} (1/N)^l (1-1/N)^{C_{m-1}-l}$$

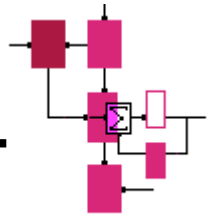
- Where

C_{m-1} = number of HOL messages that departed during the m-1 slot = number of new HOL arrivals

- As N approaches infinity, A_m^i becomes Poisson of rate C/N where C is the average number of departures per slot



Throughput analysis, continued



- In steady-state, Q^i behaves as an M/D/1 of rate $\overline{A}$ and, as before,

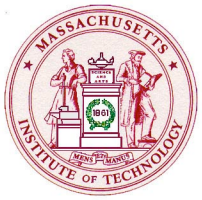
$$\overline{Q^i} = \frac{(\overline{A})^2}{2(1-\overline{A})}$$

- Notice however that the total number of packets addressed to the outputs is N (number of HOL packets). Hence,

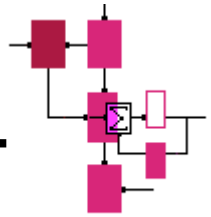
$$\sum_{i=1}^{N} Q^i = N \quad \Rightarrow \overline{Q^i} = \frac{(\overline{A})^2}{2(1-\overline{A})} = 1$$

- We can now solve, using the quadratic equation to obtain:

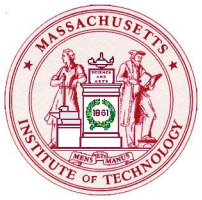
$$\overline{A} = utilization = 2 - \sqrt{2} \approx 0.58$$



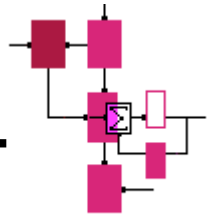
Summary of input queued switches



- The maximum throughput of an input queued switch, is limited by HOL blocking to 58% (for large N)
  - Assuming uniform traffic and FCFS service
- Advantages of input queues:
  - Simple
  - Bus rate = line rate
- Disadvantages: Throughput limitation

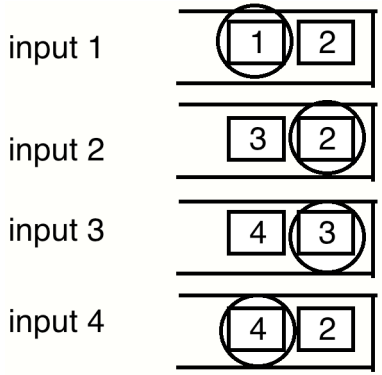


Overcoming HOL blocking

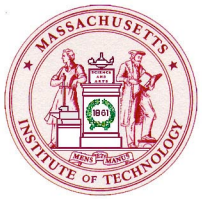


- If inputs are allowed to transfer packets that are not at the head of their queues, throughput can be substantially improved (not FCFS)

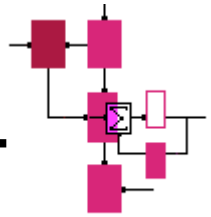
Example:



How does the scheduler decide which input to transfer to which output?

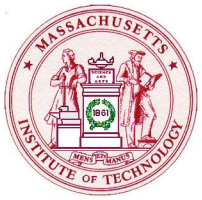


Backlog matrix

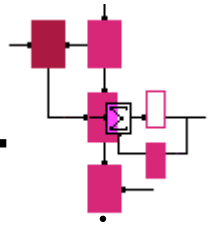


input \ output	1	2	3
1	3	3	0
2	2	0	0
3	0	0	2

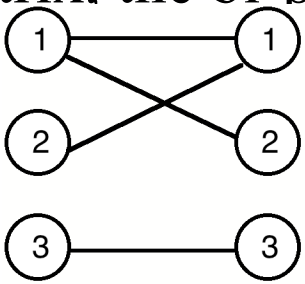
- Each entry in the backlog matrix represent the number of packets in input i's queue that are destined to output j
- During each slot the scheduler can transfer at most one packet from each input to each output
  - The scheduler must choose one packet (at most) from each row, and column of the backlog matrix
  - This can be done by solving a bi-partite graph matching algorithm
  - The bi-partite graph consists of N nodes representing the inputs and N nodes representing the outputs



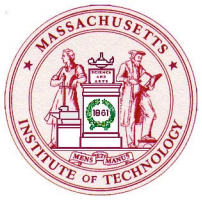
Bi-partite graph representation



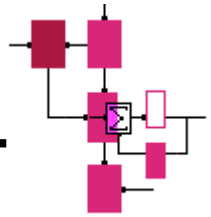
- There is an edge in the graph from an input to an output if there is a packet in the backlog matrix from that input to that output
- For previous backlog matrix, the bi-partite graph is:



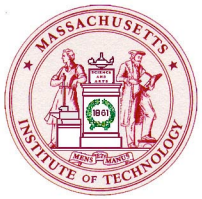
- A matching is a set of edges, such that no two edges share a node: a matching in the bi-partite graph is equivalent to a set of packets such that no two packets share a row or column in the backlog matrix
- A maximum matching is a matching with the maximum possible number of edges: a maximum matching is equivalent to the largest set of packets that can be transferred simultaneously



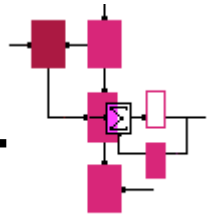
Maximum matchings



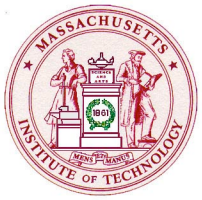
- Algorithms for finding maximum matching exist
- The best known algorithms takes $O(N^{2.5})$ operations
  - Too long for large N
- Alternatives
  - Sub-optimal solutions
  - Maximal matching: A matching that cannot be made any larger for a given backlog matrix
  - For previous example:
    (1-1,3-3) is maximal
    (2-1,1-2,3-3) is maximum
- Fact: The number of edges in a maximal matching $\geq 1/2$ the number of edges in a maximum matching



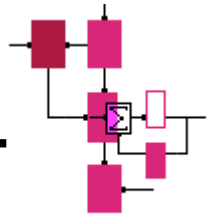
Achieving 100% throughput in an input queued switch



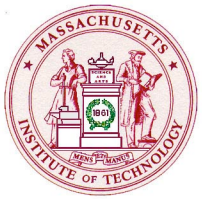
- Finding a maximum matching during each time slot does not eliminate the effects of HOL blocking
  - Must look beyond one slot at a time in making scheduling decisions
- Definition: A weighted bi-partite graph is a bi-partite graph with costs associated with the edges
- Definition: A maximum weighted matching is a matching with the maximum edge weights
- Theorem: A scheduler that chooses during each time slot the maximum weighted matching where the weight of link (i,j) is equal to the length of queue (i,j) achieves full utilization (100% throughput)
  - Proof: see "Achieving 100% throughput in an input queued switch" by N. McKeown, et. al., IEEE Transactions on Communications, Aug. 1999.



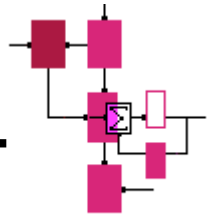
General relation with bipartite matching



- Stability of infinite input-buffered switch iff we can decompose the traffic as a convex linear combination of 0,1 sub-stochastic matrices
- Birkhoff-von Neumann principle
- This links packets and flows to circuits
- Corollary: if we know the traffic matrix well, then we can provide stable service through a TDM schedule
- Delay effects?
- Robustness to poor knowledge of the traffic?

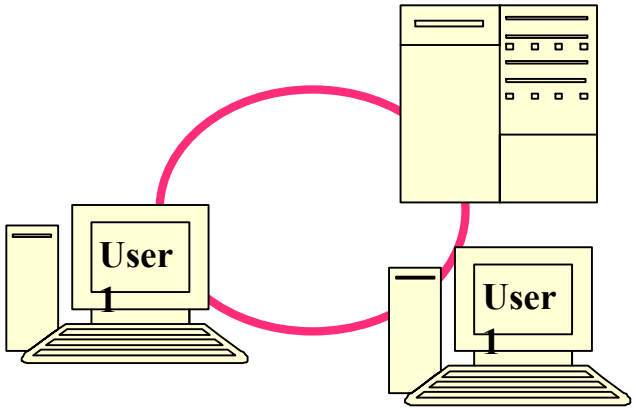


LANs



- The driver behind LANs can be roughly thought of as increasing the reach and sharing of a bus
- Traditional Ethernet: CSMA/CD, shared
- Other approach: token ring, for instance Fiber data distributed interface (FDDI)

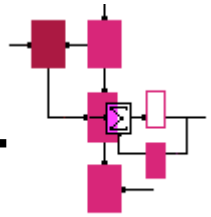
Shared ring



- Switched networks:

Lines are not shared but go through a router/switch

IEEE/ANSI 802 standards

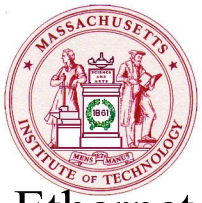


Each of the 802.3-12 have both a Medium access and a physical standard

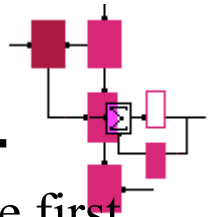
802.2 logical link control

802.1 bridging

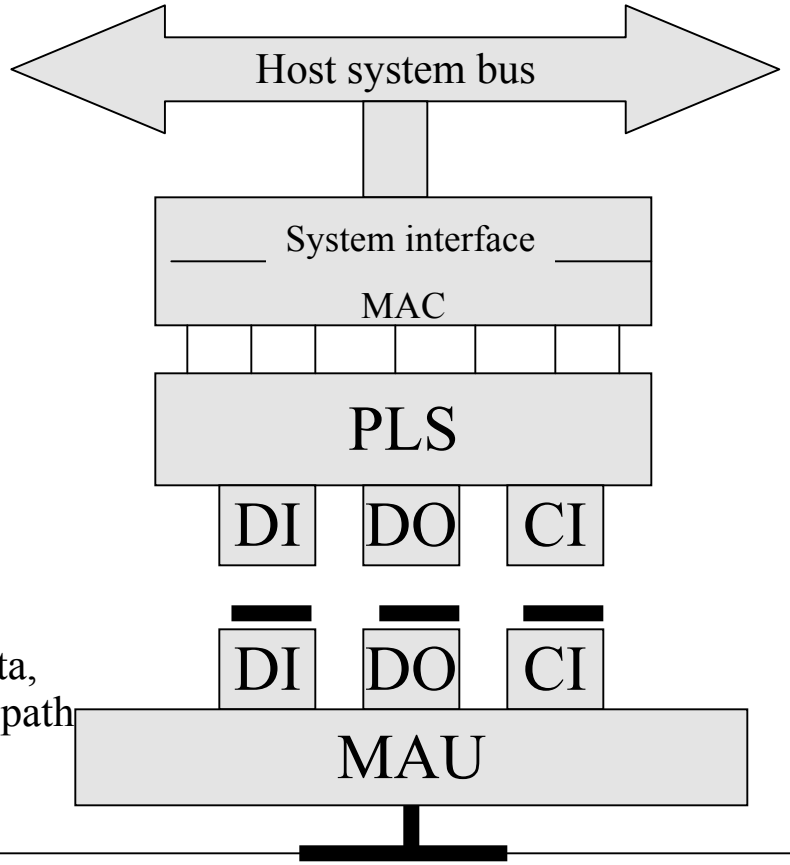
802.3: CSMA/CD Ethernet	802.4 Token bus	802.5 Token Ring	802.6 DQDB MAN	802.9 IIS LAN	802.11 Wireless LAN	802.12 DPAM
			Distributed queue dual bus	Integrated services		Demand priority Access method

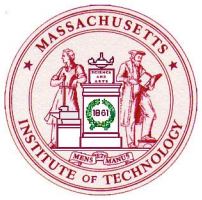


Evolution of Ethernet

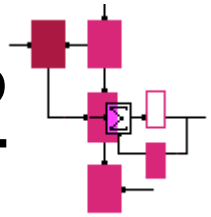


- Ethernet emerged form the ideas of shared media such as ALOHA and the first Ethernet was built at Xerox Parc in the early 1970s
- Ethernet s not completely 802.3, but a close approximation (there are some differences in the packet)
- Ethernet node:
- MAC enforces CSMA/CD and performs:
  - Transmit and receive message data encapsulation:
    - Framing
    - Addressing
    - Error detection
  - Media access management:
    - Medium allocation (collision avoidance)
    - Contention resolution (collision handling)
- PLS: physical signaling, Manchester encoding
- AUI (attachment unit interface) manages data in (DI), data out (DO) and control in (CI)
- Medium attachment unit (MAU): transmits and receives data, loops data back from DO to DI to indicate valid Tx and Rx path detects collisions, sends signal quality error signal, performs jabber function, checks link integrity

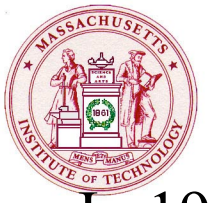




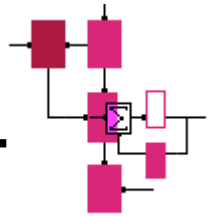
Increasing Ethernet bandwidth – the first step



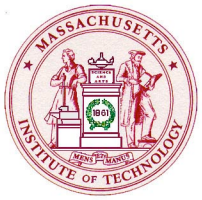
- The first Ethernet went up to 10 Mbs – 10BASE-T, over phone grade twisted pair, with a repeater in the middle of a star configuration acting as a virtual shared medium (also traditional 10Base5 and Cheapernet 10BASE2 on thick and thin coax, respectively were laid out)
- 10Base-T over fiber was developed, extending the distance between MAUs to 2 km instead of 500 m in coax
- 1990: the Etherswitch was marketed by Kalpana to boost LAN performance rather than as a bridge to interconnect different LANs and in 1993 full-duplex interconnect was also introduced by Kalpana
- Still each port could only deliver 10 Mbps, the option for higher (100 Mbps) connection was FDDI, which was expensive



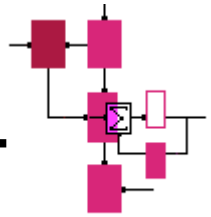
Fast Ethernet



- In 1992, Grand Junction introduced 100 Mbps Ethernet
- Standardization was done by the Fast Ethernet Alliance, while the IEEE struggled between 802.3 and a demand-priority camp, which created the 802.12 group
- Later 803.2u standardized 100BASE-T
- Main differences between 10BASE-T and 100BASE-T:
  - No more mixing segments (coax with multiple devices attached), all cabling is point to point between terminal equipment or repeaters
  - Shorter distances – 100 m for Cat 5, Cat 3 and 130 m for fiber (160 m if all fiber network)
  - Kept the MAC but changed elements below to adapt ot 100 Mbps - replaced the AUI with the media independent sublayer, added a reconciliation sublayer (going from bit-derial to nibble-serial), went from Manchester encoding to NRZ
- 10 GigE is emerging as a new standard **http://www.10gea.org/Tech-whitepapers.htm**



10 Gigabit Ethernet

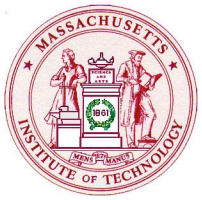


•10 GigE is emerging as a new standard

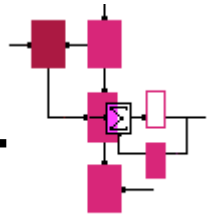
• The standard is being developed with SONET interoperability in mind with a view towards expansion in the MAN and WAN end-to-end Ethernet arena

• In particular, the load will be be matched to OC-192 loads

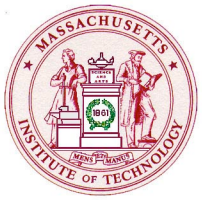
•Task force 802.3ae is in charge of developing 10 GE standard



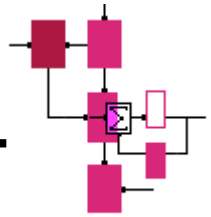
Evolution to switched LANs



- VLANS were introduced to allow for smaller broadcast group:
  - the standardization efforts have not yet yielded interoperable VLANs, they are still proprietary solutions
  - VLANs require a frame extension (802.3ac) to convey VLAN information via tagging (802.1Q) (2 tags of 16 bits each), approved in 1998
- Layer 3 switches implement some routing in hardware:
  - Routers were generally used for interconnecting LANs and for remote WAN connections
  - Switches traditionally had little intelligence but were very fast
  - Layer 3 switches still perform layer 2 switching but also some routing functionality in ASICs
  - They also implement VLANs
  - Generally support only IP



The next step in Ethernet- Gigabit Ethernet



- The Gigabit Ethernet Alliance (May 1996) started the push for Gigabit Ethernet, mostly standardized as 802.3z in 1998
- Main characteristics:
  - The MAC itself was modified so that there is 200 m network span with a single repeater
  - The MII was changed to GMII, Tx and Rx data paths widened to 8 bits
  - Adoption of 8bit/10bit fibre channel encoding
  - Carrier extension: extending or padding from 64-byte minimum to 512-byte minimum to maintain compatibility
  - Frame bursting to enhance efficiency:

    worst-case efficiency for 100 Mb/s CSMA/CD is 1000 Mb/s with CSMA/CD is

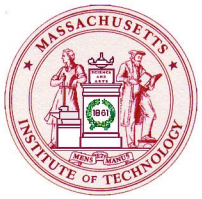
$$\frac{512}{4096+96+64} = 12\%$$

Minimum packet length

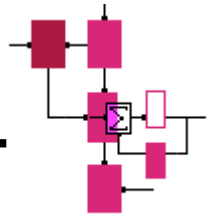
$$\frac{512}{512+96+64} = 76\%$$ for

Preamble length

Inter-frame gap



Frame-bursting for Efficiency



- Frame bursting to enhance efficiency
- Worst-case efficiency for 100 Mb/s CSMA/CD is

Minimum packet length

$$\frac{512}{512 + 96 + 64} = 76\%$$

Preamble length

Inter-frame gap

- For 1000 Mb/s with CSMA/CD is

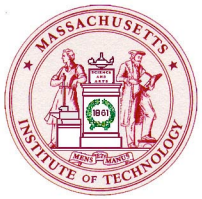
Slot time

$$\frac{512}{4096 + 96 + 64} = 12\%$$

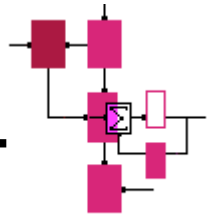
- If we allow n frames to be transmitted in a burst after the first frame then worst-case efficiency is

$$\frac{512(n+1)}{4096 + n(512 + 96 + 64)}$$

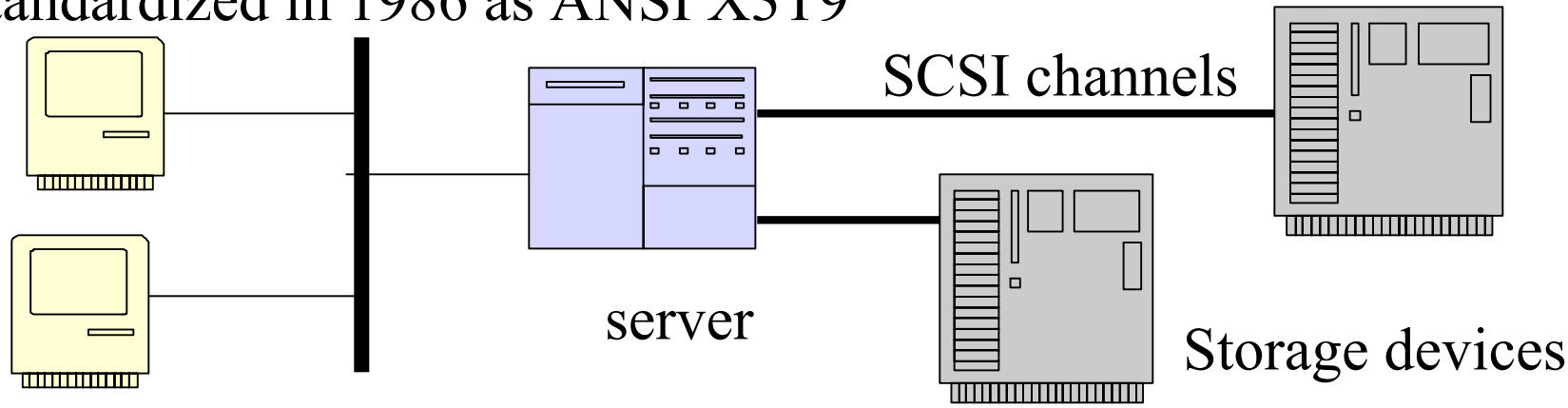
- Efficiency gains beyond 65,536 bits is minimal and is about 72% at that value



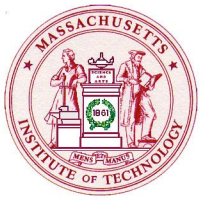
Another LAN application: storage access



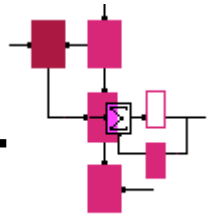
- In open systems world, dominant I/O technology is small computer system interface (SCSI), which transfers data in blocks standardized in 1986 as ANSI X3T9



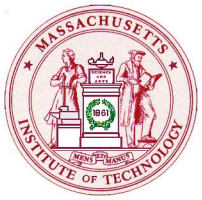
- SCSI drawbacks:
  - Two or more I/O controllers cannot easily share SCSI devices on the same I/O bus, so a single server controls connections between users and their data
  - Address on an I/O bus: 8 or 16 addresses depending on implementation
  - Distance 25 m



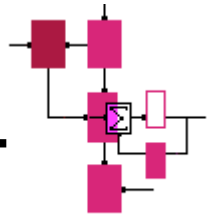
A new type of LAN – the SAN



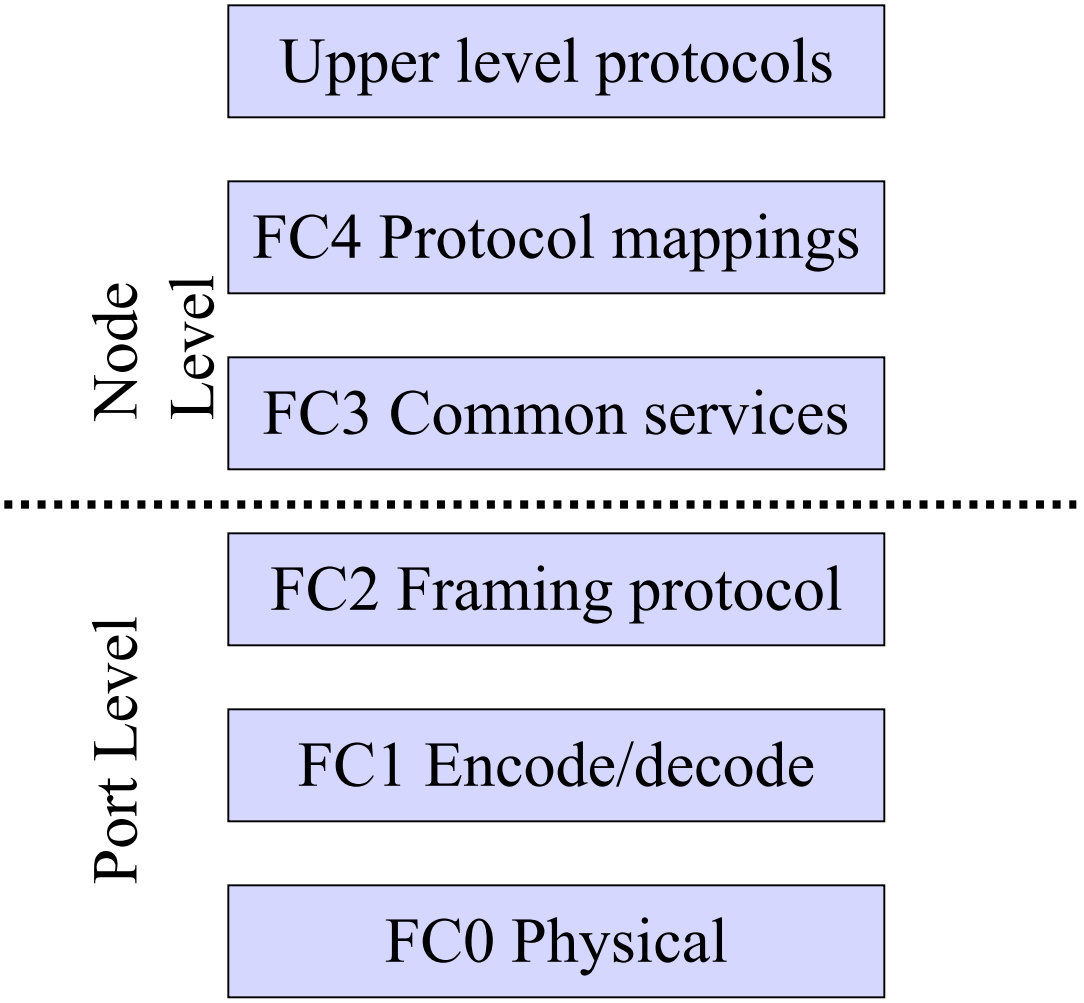
- In the same way that early LANs developed from extending the bus, the requirement for more storage has driven extending the SCSI interface to many devices and eventually replacing a single storage device with a full network, the storage area network (SAN)
- Based on Fibre Channel protocol (FC) fiber channel:
  - Gigabit per second bandwidth (1063 Mbps) and theoretically up to 4 Gbps
  - Allows SCSI in serial form rather than the parallel form usually found in SCSI (also supports HIPPI and IPI I/O protocols)
  - Distance of up to 10 km
  - 24-bit address identifier – up to 16 million ports

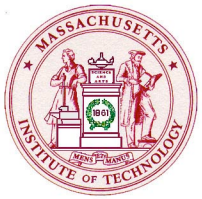


FC

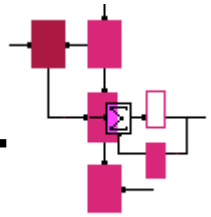


- Upper level protocols include application, device drivers, operating systems
- Common services are striping, hunt groups, multicast
- Framing: frames of up to 2112 bytes, sequences (one or more frames), exchanges (uni or bidirectional set of non-concurrent sequences, packets (one or more exchanges)

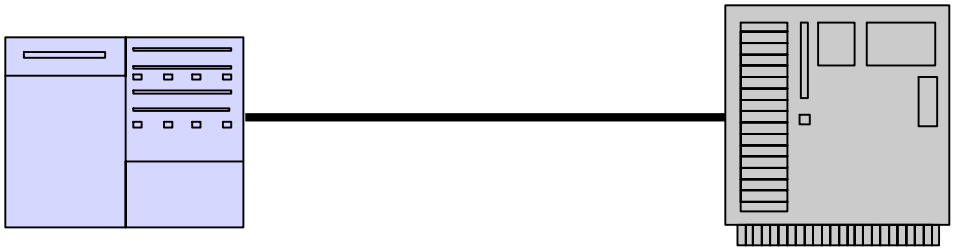




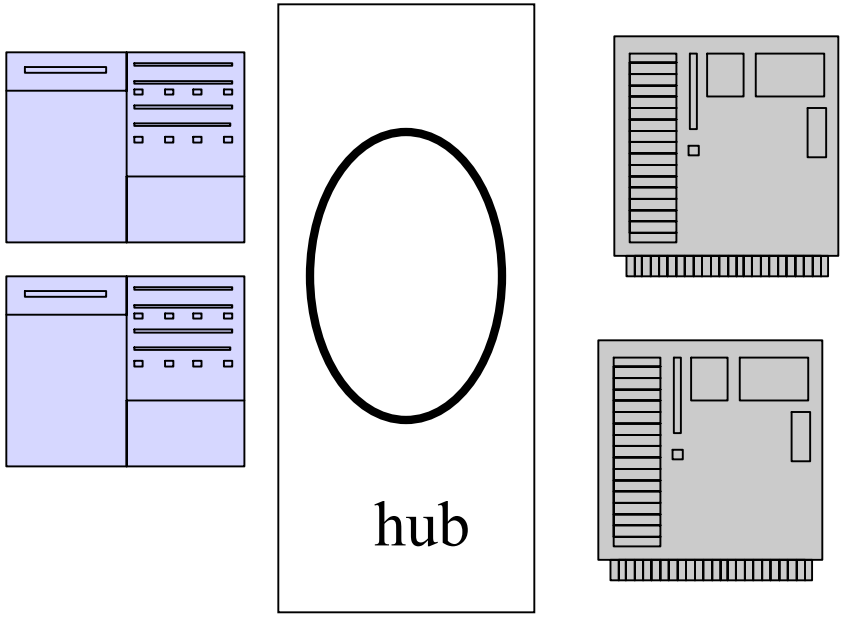
Different types of FC SAN architectures

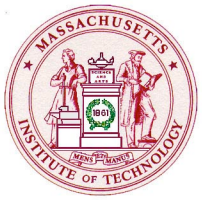


- Point-to-point

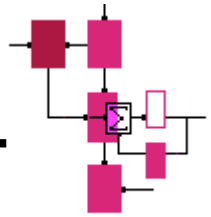


- Arbitrated loop topology:
  - up to 126 devices in a serial loop configuration
  - Each port discovers when it has been attached
  - No collisions
  - Fair access: every port wanting to initiate traffic gets to do so before another port gets a second shot

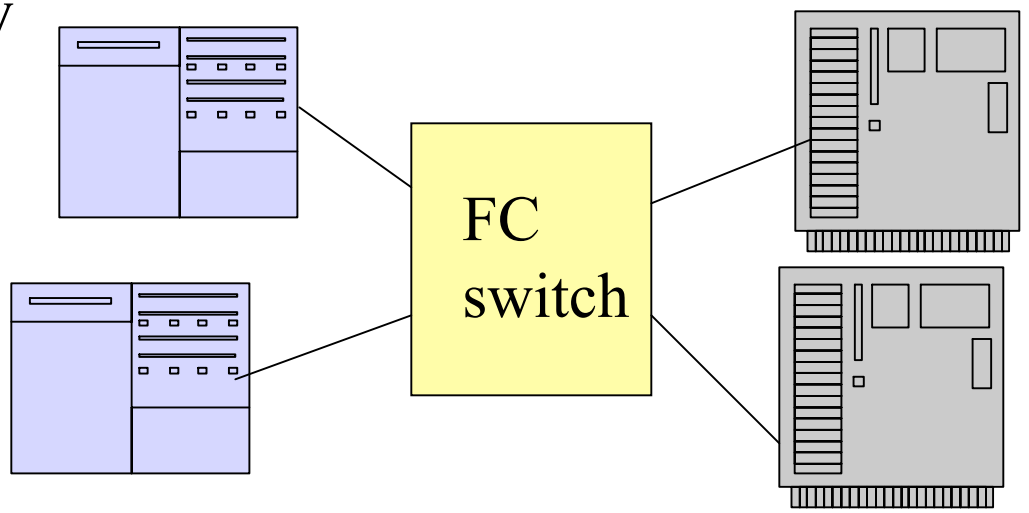




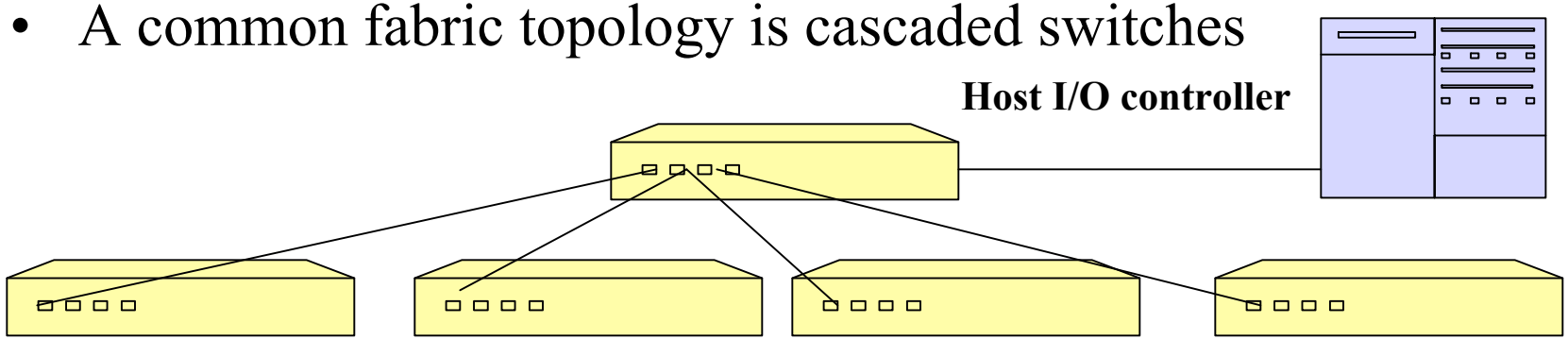
Different types of FC SAN architectures

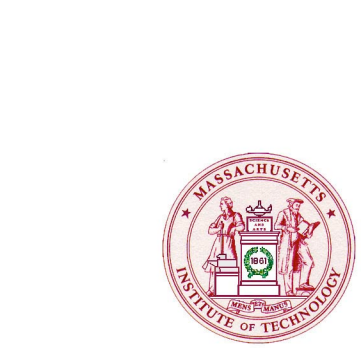


- Fabric topology

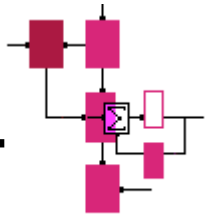


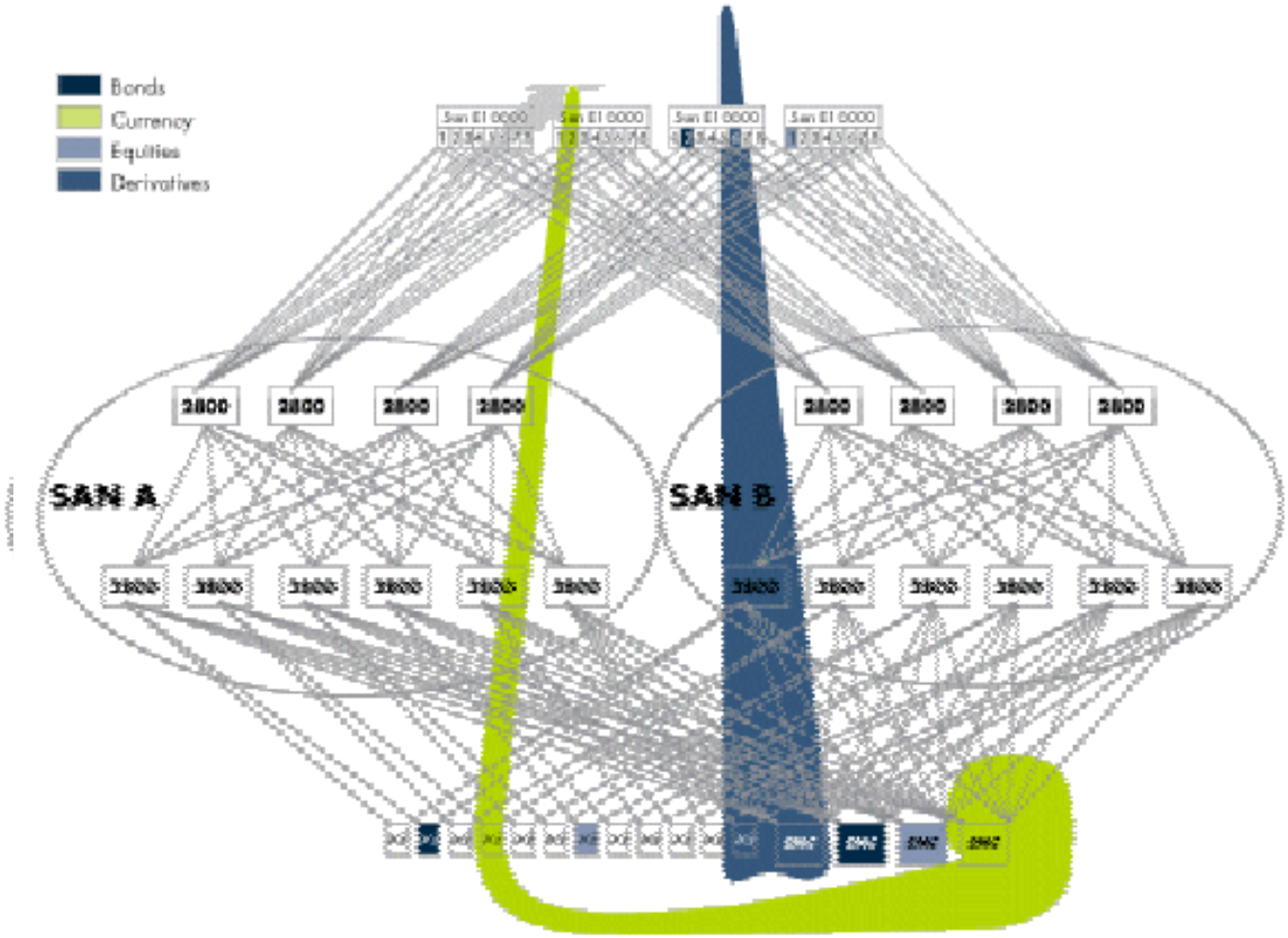
- A common fabric topology is cascaded switches



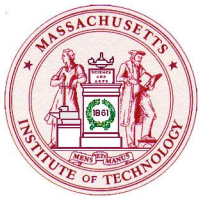


This is not a shared bus!

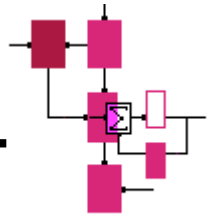




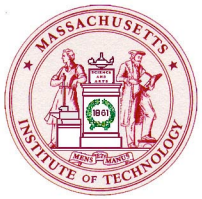
Commerzbank Brocade set-up



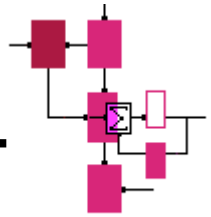
Other alternatives to SANs

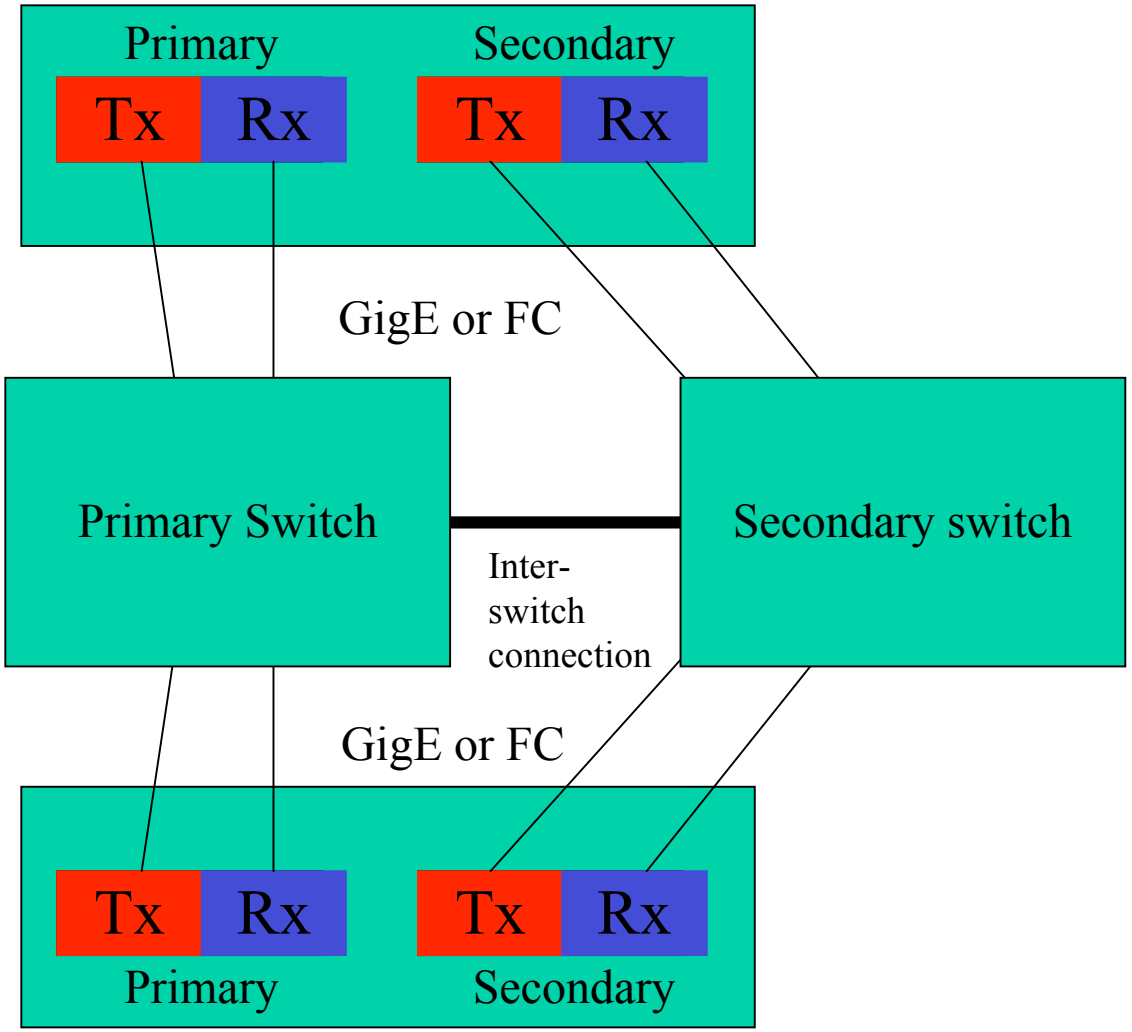


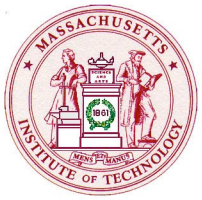
- Embedded disk drives
- Directly attached storage attached by SCSI directly, possibly shared among servers
- Network attached storage is in front of the server, directly attached to the network, rather than behind the server as a SAN
  - Protocol is generally NFS vs. FC for SAN
  - Network is Ethernet vs. FC for SAN
  - Source and target are client/server or server/server vs. server/device for SAN
  - Transfers files vs. device blocks for SAN
  - Connection is direct on network vs. I/O bus or channel on server for SAN
  - Has an embedded file system



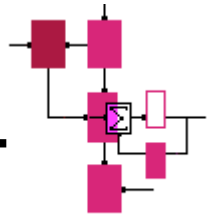
High availability in the enterprise



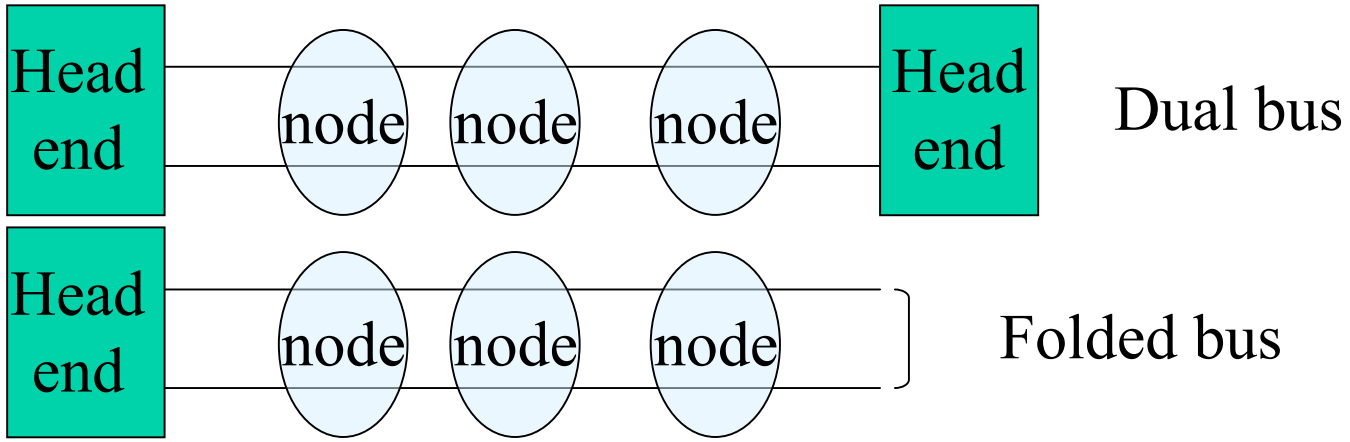


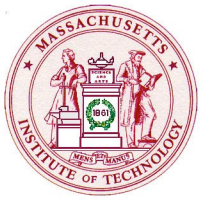


MANs

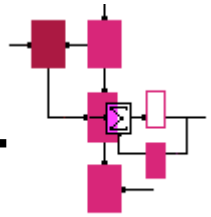


- MANs are a fuzzy area since they may operate as large LANs or simply as the last leg of a WAN
- Certain protocols are particularly oriented towards MANs, such a DQDB, dual bus either folded or not folded :
  - Exhibited certain issues with utilization fairness
  - Not very flexible in its layout architecture

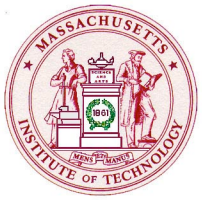




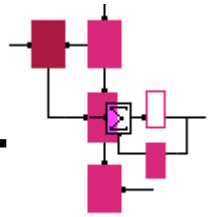
Resilient Packet Ring



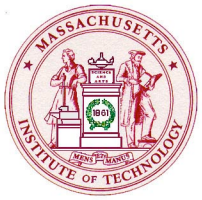
- Rings for packet access in the MAN
- Resilient packet ring alliance (RPR) and IEEE working group 802.17 (started December 2000)
- Oriented towards IP
- Recovery is done using traditional self-healing ring approach
- Maintains the same architecture as SONET rings and FDDI, but changes the MAC



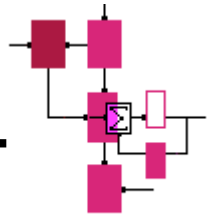
WANs



- WANs are predominantly implemented over optical networks
- The underlying protocol is SONET (synchronous optical network) or SDH in Europe and Japan (synchronous digital hierarchy)
- Synchronous, so framing is in terms of timing
- Lowest-speed SONET runs at STS-1, 51.84 Mbps
- STS frames may be concatenated with a single header, which contains pointers to the different headers of the STS frames
- SONET provides very tight requirements on reliability
- Typical implementations are UPSR or BLSR
- Recovery must occur within 50 ms, detection of a problem occurs within 2.4 microseconds



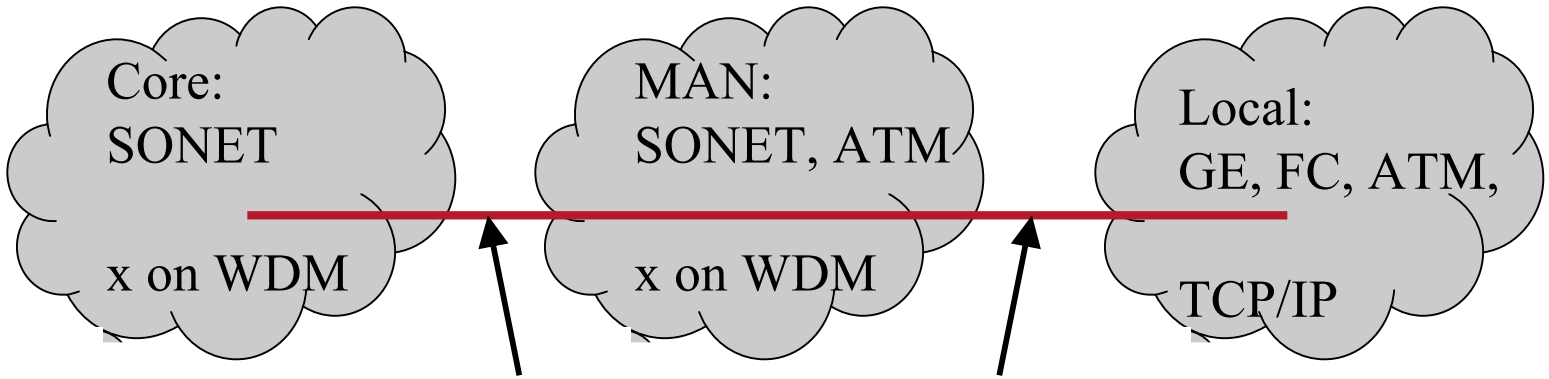
WANs



- WANs are increasingly dense and require extensive network management
- Provisioning across WANs in short time is a growing as the reselling market becomes more fluid
- WANs are increasingly called upon to perform functions heretofore reserved for LANs or MANs, so there is increasing convergence
- Speed per wavelength is now 0C-48 (2.5 Gbps), OC-192 (10 Gbps) possibly going towads 40 Gbps

Access to the Optical Infrastructure

- Two trends in optical access:
  - IP, GE being pushed closer to the core
  - streaming media pushing core-type traffic closer to the edge
- How should access be architected:
  - role of network management
  - types of nodes



Access: MPLS or other encapsulation