Lecture 7: Interconnection Networks

Prof. Fred Chong
ECS 250A Computer Architecture
Winter 1999

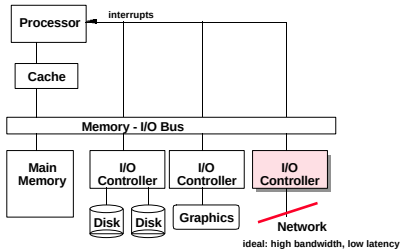
Review: Storage System Issues

- Historical Context of Storage I/O
- Secondary and Tertiary Storage Devices
- Storage I/O Performance Measures
- Processor Interface Issues
- Redundant Arrays of Inexpensive Disks (RAID)
- ABCs of UNIX File Systems
- I/O Benchmarks

Review: I/O Benchmarks

- Scaling to track technological change
- TPC: price performance as nomalizing configuration feature
- Auditing to ensure no foul play
- Throughput with restricted response time is normal measure

I/O to External Devices and Other Computers

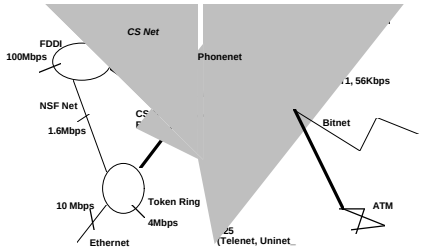


ideal: high bandwidth, low latency

Networks

- Goal: Communication between computers
- Eventual Goal: treat collection of computers as if one big computer, distributed resource sharing
- Theme: Different computers must agree on many things
  - Overriding importance of standards and protocols
  - Fault tolerance critical as well
- Warning: Terminology-rich environment

Example Major Networks

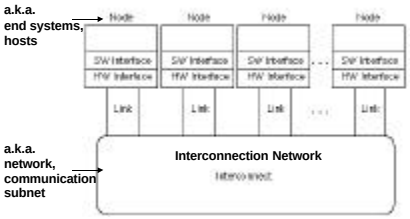


Networks

- Facets people talk a lot about:
  - direct (point-to-point) vs. indirect (multi-hop)
  - topology (e.g., bus, ring, DAG)
  - routing algorithms
  - switching (aka multiplexing)
  - wiring (e.g., choice of media, copper, coax, fiber)
- What really matters:
  - latency
  - bandwidth
  - cost
  - reliability

Interconnections (Networks)

- Examples:
  - MPP networks (SP2): 100s nodes; Š 25 meters per link
  - Local Area Networks (Ethernet): 100s nodes; Š 1000 meters
  - Wide Area Network (ATM): 1000s nodes; Š 5,000,000 meters

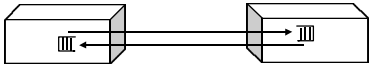


More Network Background

- Connection of 2 or more networks: Internetworking
- 3 cultures for 3 classes of networks
  - MPP: performance, latency and bandwidth
  - LAN: workstations, cost
  - WAN: telecommunications, phone call revenue
- Try for single terminology
- Motivate the interconnection complexity incrementally

ABCs of Networks

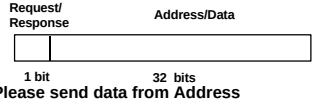
- Starting Point: Send bits between 2 computers



- Queue (FIFO) on each end
- Information sent called a "message"
- Can send both ways ("Full Duplex")
- Rules for communication? "protocol"
  - Inside a computer:
    - » Loads/Stores: Request (Address) & Response (Data)
    - » Need Request & Response signaling

A Simple Example

- What is the format of mesage?
  - Fixed? Number bytes?



0: Please send data from Address
1: Packet contains data corresponding to request

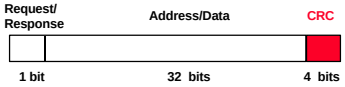
- Header/Trailer: information to deliver a message
- Payload: data in message (1 word above)

Questions About Simple Example

- What if more than 2 computers want to communicate?
  - Need computer "address field" (destination) in packet
- What if packet is garbled in transit?
  - Add "error detection field" in packet (e.g., CRC)
- What if packet is lost?
  - More "elaborate protocols" to detect loss (e.g., NAK, ARQ, time outs)
- What if multiple processes/machine?
  - Queue per process to provide protection
- Simple questions such as these lead to more complex protocols and packet formats => complexity

A Simple Example Revisted

- What is the format of packet?
  - Fixed? Number bytes?

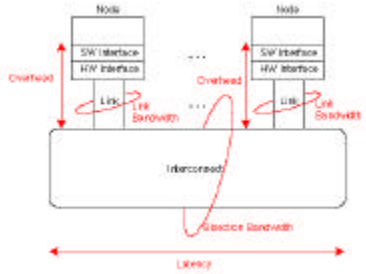


00: Request—Please send data from Address
01: Reply—Packet contains data corresponding to request
10: Acknowledge request
11: Acknowledge reply

Software to Send and Receive

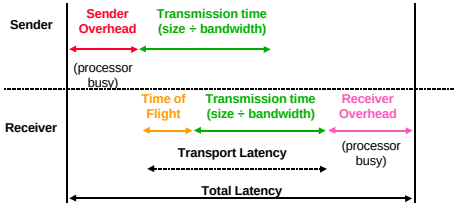
- SW Send steps
  1: Application copies data to OS buffer
  2: OS calculates checksum, starts timer
  3: OS sends data to network interface HW and says start
- SW Receive steps
  3: OS copies data from network interface HW to OS buffer
  2: OS calculates checksum, if matches send ACK; if not, *deletes message* (sender resends when timer expires)
  1: If OK, OS copies data to user address space and signals application to continue
- Sequence of steps for SW: protocol
  - Example similar to UDP/IP protocol in UNIX

Network Performance Measures



- Overhead: latency of interface vs. Latency: network

Universal Performance Metrics



Total Latency = Sender Overhead + Time of Flight + Message Size ÷ BW + Receiver Overhead

Includes header/trailer in BW calculation?

Example Performance Measures

Interconnect	*MPP*	*LAN*	*WAN*
Example	CM-5	Ethernet	ATM
Bisection BW	N x 5 MB/s	1.125 MB/s	N x 10 MB/s
Int./Link BW	20 MB/s	1.125 MB/s	10 MB/s
Transport Latency	5 µsec	15 µsec	50 to 10,000 µs
HW Overhead to/from	0.5/0.5 µs	6/6 µs	6/6 µs
SW Overhead to/from	1.6/12.4 µs	200/241 µs	207/360 µs
		(TCP/IP on LAN/WAN)	

Software overhead dominates in LAN, WAN

Total Latency Example

- 10 Mbit/sec., sending overhead of 230 µsec & receiving overhead of 270 µsec.
- a 1000 byte message (including the header), allows 1000 bytes in a single message.
- 2 situations: distance 0.1 km vs. 1000 km
- Speed of light = 299,792.5 km/sec (1/2 in media)
- $Latency_{0.1km}$ =
- $Latency_{1000km}$ =
- Long time of flight => complex WAN protocol

Total Latency Example

- 10 Mbit/sec., sending overhead of 230 µsec & receiving overhead of 270 µsec.
- a 1000 byte message (including the header), allows 1000 bytes in a single message.
- 2 situations: distance 100 m vs. 1000 km
- Speed of light = 299,792.5 km/sec
- $Latency_{0.1km}$ = 230 + 0.1km / (50% x 299,792.5) + 1000 x 8 / 10 + 270
- $Latency_{0.1km}$ = 230 + 0.67+ 800 + 270 = 1301 µsec
- $Latency_{1000km}$ = 230 + 1000 km / (50% x 299,792.5) + 1000 x 8 / 10 + 270
- $Latency_{1000km}$ = 230 + 6671+ 800 + 270 = 7971 µsec
- Long time of flight => complex WAN protocol

FTC.W99 19

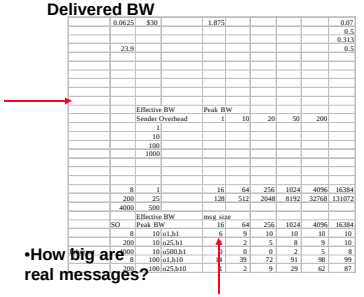
Simplified Latency Model

- Total Latency - Overhead + Message Size / BW
- Overhead = Sender Overhead + Time of Flight + Receiver Overhead
- Example: show what happens as vary
  - Overhead: 1, 25, 500 µsec
  - BW: 10,100, 1000 Mbit/sec (factors of 10)
  - Message Size: 16 Bytes to 4 MB (factors of 4)
- If overhead 500 µsec, how big a message > 10 Mb/s?

FTC.W99 20

Overhead, BW, Size

Delivered BW

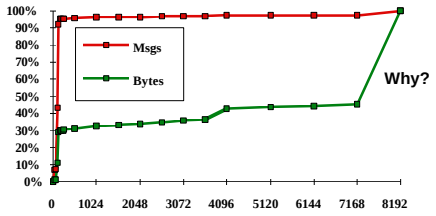


Msg Size

•How big are real messages?

FTC.W99 21

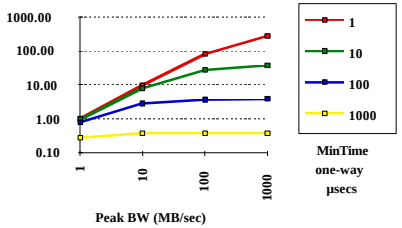
Measurement: Sizes of Message for NFS



- 95% Msgs, 30% bytes for packets Š 200 bytes
- > 50% data transfered in packets = 8KB

FTC.W99 22

Impact of Overhead on Delivered BW



- BW model: Time = overhead + msg size/peak BW
- > 50% data transfered in packets = 8KB

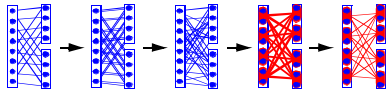
FTC.W99 23

Building a Better Butterfly: The Multiplexed Metabutterfly

Chong, Brewer, Leighton, and Knight

FTC.W99 24

Transformations



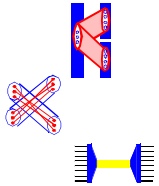
Butterfly	Dilated Butterfly	Multi-butterfly	Meta-butterfly	Multiplex
lowest cost	2X cost	high complex.	lower complex.	lower cost
poor	better	great	great	good

fault & congestion performance

FTC.W99 25

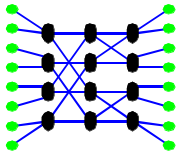
Outline

- **Expander Networks**
  - Good, but hard to build
- **Hierarchical Construction**
  - Much simpler
  - Almost as good in theory
  - Just as good in simulation
- **Multiplexing**
  - Much better grouping
  - Randomize for efficiency
- **The cost of a butterfly the performance of a multibutterfly**



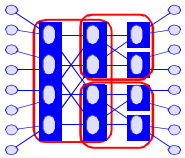
FTC.W99 26

Butterfly



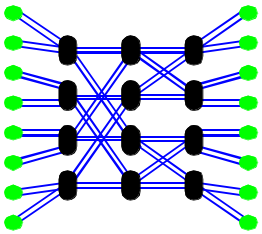
FTC.W99 27

Splitter Network



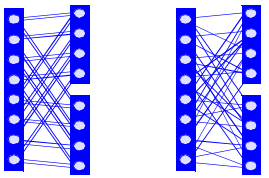
FTC.W99 28

Dilated Butterfly



FTC.W99 29

Wiring Splitters

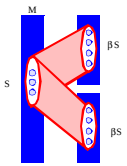


Dilated Better

FTC.W99 30

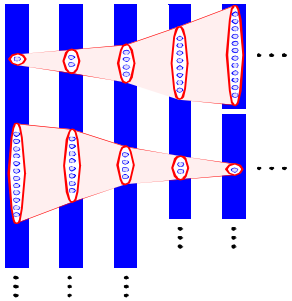
Expansion

- **Definition: A splitter has expansion if every set of $S < \alpha M$ inputs reaches at least βS outputs in each of r directions, where $\beta > 1$ and $\alpha < 1 / (\beta r)$**
- **Random wiring produces the best expansion**



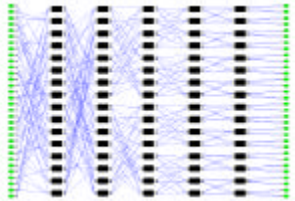
FTC.W99 31

Faults and Congestion



FTC.W99 32

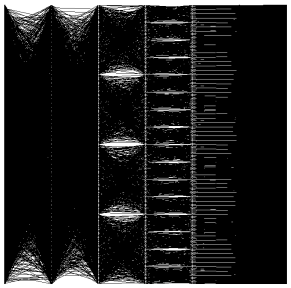
Multibutterflies



- **Randomly wired**
- **Extensively studied:**
  - **[Bassalygo & Pinsker 74] [Upfal 89]**
  - **[Leighton & Maggs 89][Arora, Leighton & Maggs 90]**
- **Tremendous fault and congestion tolerance**

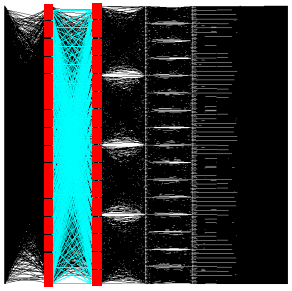
FTC.W99 33

What's Wrong?



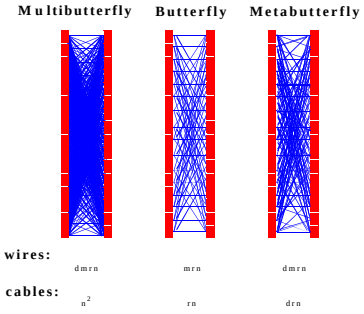
FTC.W99 34

Wiring Complexity



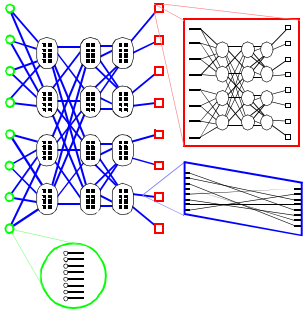
FTC.W99 35

Relative Complexity



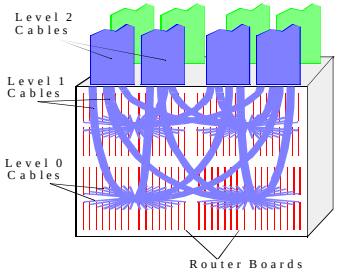
FTC.W99 36

Metabutterfly



FTC.W99 37

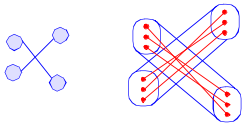
Hierarchy



- **Constant degree at each level**

FTC.W99 38

Random K-Extensions



FTC.W99 39

K-Extension Properties

- **Preserve Expansion (with high probability):**

$$\beta_{new} = \beta - 2$$

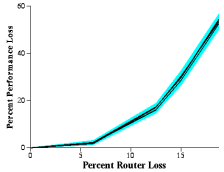
$$\alpha_{new} = \frac{\alpha^2}{\beta^2 e^4 + 4\alpha}$$

- **[Brewer, Chong, & Leighton, STOC94]**

FTC.W99 40

Empirical Results

- **Methodology of [Chong & Knight, SPAA92]**
  - **Uniformly distributed router faults**
  - **1024-processor networks with 5 stages**
  - **shared-memory traffic pattern**

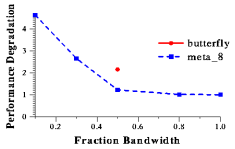


- **Metabutterflies (with metanode sizes 4, 16, 32) perform as well as the Multibutterfly**

FTC.W99 41

Multiplexing

- **Multiplex cables**
  - **Not possible with multibutterfly**
- **Random Destinations**

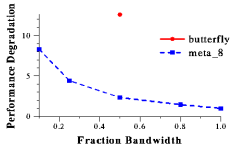


- **Can remove half the wires!**
- **2X performance of comparable butterfly**

FTC.W99 42

Multiplexing (Bit-Inverse)

- Over 5X better on bit-inverse

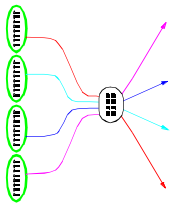


- Multiple logical paths without excess physical bandwidth

FTC.W99 43

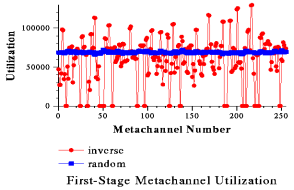
Load Balancing

- Why is bit-inverse worse than random?



FTC.W99 44

Unbalanced Loading

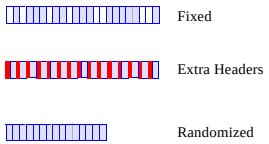


- Solutions:
  - balance bit-inverse
  - more wires in first stage
  - more bandwidth in first stages

FTC.W99 45

Randomized Multiplexing

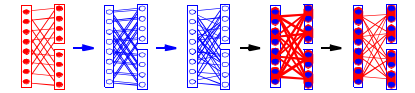
- Within cables, packet destination unimportant
  - Could be random
  - Assign each packet to any output
- Better bandwidth
  - No fixed time slots
  - No extra headers



- Dynamic randomness

FTC.W99 46

Summary



- Metabutterfly
  - best fault and congestion tolerance
- Multiplexed Metabutterfly
  - comparable cost to butterfly
  - much better fault and congestion tolerance
- K-extensions and Multiplexing
  - applicable to other networks (eg fat trees)

FTC.W99 47

Conclusions

- Other Expander-Based Networks
  - Fat-Trees
  - Deterministic Constructions
- Non-Random K-extensions
  - How many permutations?
- Other Networks with Multiplicity
- Expanders are great, but were hard to build

- K-extensions are the solution
  - Allow Fixed Cabling Degree
  - Retain Theoretical Properties
  - Equal Multibutterflies in Simulation

FTC.W99 48

Interconnect Outline

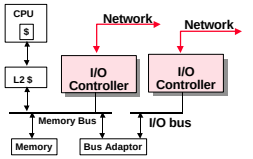
- Performance Measures
- (Metabutterfly Example)
- Interface Issues

HW Interface Issues

- Memory Bus
  - Low latency, high bandwidth
  - Limited length to maintain speed
  - Proprietary protocols
- I/O Bus
  - Standard protocols
  - Slow
  - Can be long

HW Interface Issues

- Where to connect network to computer?
  - Cache consistent to avoid flushes?
  - RAM
  - Hard Drive
  - Standard interface card?
  - MPP
  - LAN/WAN



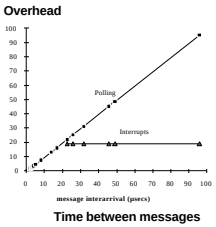
ideal: high bandwidth, low latency, standard interface

SW Interface Issues

- How to connect network to software?
  - Programmed I/O?
  - DMA?
  - Receiver interrupted or receiver polls?
- Things to avoid
  - Invoking operating system in common case
    - » Why would designers want to invoke OS for communication?
  - Operating at uncached memory speed (e.g., check status of network interface)

CM-5 Software Interface

- CM-5 example (MPP)
  - User does msgs w/out OS
  - Time per poll 1.6 µsecs; time per interrupt 19 µsecs
  - Minimum time to handle message: 0.5 µsecs
  - Enable/disable 4.9/3.8 µsecs
- As rate of messages arrving changes, use polling or interrupt?
  - Solution: Always enable interrupts, have interrupt routine poll until until no messages pending
  - Low rate => - interrupt
  - High rate => - polling



Interconnect Issues

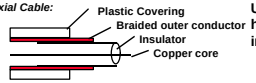
- Performance Measures
- Interface Issues
- Network Media

Network Media

Twisted Pair:

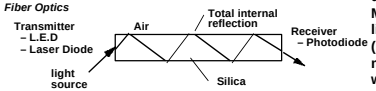
Copper, 1mm think, twisted to avoid antenna effect (telephone)

Coaxial Cable:



Used by cable companies: high BW, good noise immunity

Fiber Optics



Light: 3 parts are cable, light source, light detector. Multimode light disperse (LED), Single mode single wave (laser)

Costs of Network Media (1995)

Media	Bandwidth	Distance	Cost/meter	Cost/interface
twisted pair copper wire	1 Mb/s (20 Mb/s)	2 km (0.1 km)	$0.23	$2
coaxial cable	10 Mb/s	1 km	$1.64	$5
multimode optical fiber	600 Mb/s	2 km	$1.03	$1000
single mode optical fiber	2000 Mb/s	100 km	$1.64	$1000

Note: more elaborate signal processing allows higher BW from copper (ADSL)

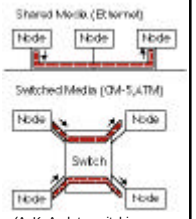
Single mode Fiber measures: BW * distance as 3X/year

Interconnect Issues

- Performance Measures
- Interface Issues
- Network Media
- Connecting Multiple Computers

Connecting Multiple Computers

- Shared Media vs. Switched: pairs communicate at same time: "point-to-point" connections
- Aggregate BW in switched network is many times shared
  - point-to-point faster since no arbitration, simpler interface
- Arbitration in Shared network?
  - Central arbiter for LAN?
  - Listen to check if being used ("Carrier Sensing")
  - Listen to check if collision ("Collision Detection")
  - Random re-send to avoid repeated collisions; not fair arbitration;
  - OK if low utilization

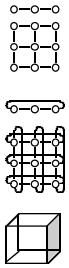


(A. K. A. data switching interchanges, multistage interconnection networks, interface message processors)

Switch Topology

- Structure of the interconnect
- Determines
  - Degree: number of links from a node
  - Diameter: max number of links crossed between nodes
  - Average distance: number of hops to random destination
  - Bisection: minimum number of links that separate the network into two halves (worst case)
- Warning: these three-dimensional drawings must be mapped onto chips and boards which are essentially two-dimensional media
  - Elegant when sketched on the blackboard may look awkward when constructed from chips, cables, boards, and boxes (largely 2D)
- Networks should not be interesting!

Important Topologies



Type	Degree	Diameter	Ave Dist	Bisection	N = 1024 Diam	Ave D
1D mesh	<=2	N-1	N/3	1		
2D mesh	<=4	$2(N^{1/2} - 1)$	$2N^{1/2} / 3$	$N^{1/2}$	63	21
3D mesh	<= 6	$3(N^{1/3} - 1)$	$3N^{1/3} / 3$	$N^{2/3}$	~30	~10
nD mesh ($N = k^n$)	<= 2n	$n(N^{1/n} - 1)$	$nN^{1/n} / 3$	$N^{(n-1)/n}$		
Ring	2	N / 2	N/4	2		
2D torus	4	$N^{1/2}$	$N^{1/2} / 2$	$2N^{1/2}$	32	16
k-ary n-cube ($N = k^n$)	2n	$n(N^{1/n})$ nk/2	$nN^{1/n}/2$ nk/4	$2k^{n-1}$	15	8 (3D)
Hypercube	n	n=LogN	n/2	N/2	10	5

Cube-Connected Cycles

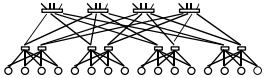
Hypercube 2^3

Topologies (cont)

N = 1024

Type	Degree	Diameter	Ave Dist	Bisection	Diam	Ave D
2D Tree	3	$2Log_3 N$	$\sim 2Log_3 N$	1	20	~20
4D Tree	5	$2Log_4 N$	$2Log_4 N$ - 2/3	1	10	9.33
kD	k+1	$Log_k N$				
2D fat tree	4	$Log_2 N$		N		
2D butterfly	4	$Log_2 N$		N/2	20	20

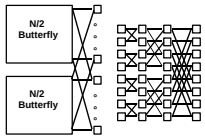
Fat Tree



CM-5 Thinned Fat Tree

Butterfly

Multistage: nodes at ends, switches in middle



- All paths equal length
- Unique path from any input to any output
- Conflicts that try to avoid
- Don't want algorithm to have to know paths

Example MPP Networks

Name	Number	Topology	Bits	Clock	Link	Bisect.	Year
nCube/ten	1-1024	10-cube	1	10 MHz	1.2	640	1987
iPSC/2	16-128	7-cube	1	16 MHz	2	345	1988
MP-1216	32-512	2D grid	1	25 MHz	3	1,300	1989
Delta	540	2D grid	16	40 MHz	40	640	1991
CM-5	32-2048	fat tree	4	40 MHz	20	10,240	1991
CS-2	32-1024	fat tree	8	70 MHz	50	50,000	1992
Paragon	4-1024	2D grid	16	100 MHz	200	6,400	1992
T3D	16-1024	3D Torus	16	150 MHz	300	19,200	1993
					MBytes/second		

No standard MPP topology!
Like everything in architecture, it's a cost/benefit trade-off.

Summary: Interconnections

- Communication between computers
- Packets for standards, protocols to cover normal and abnormal events
- Performance issues: HW & SW overhead, interconnect latency, bisection BW
- Media sets cost, distance
- Shared vs. Switched Media determines BW
- HW and SW Interface to computer affects overhead, latency, bandwidth
- Topologies: many to chose from, but (SW) overheads make them look alike; cost issues in topologies, not algorithms

Connection-Based vs. Connectionless

- Telephone: operator sets up connection between the caller and the receiver
  - Once the connection is established, conversation can continue for hours
- Share transmission lines over long distances by using switches to multiplex several conversations on the same lines
  - "Time division multiplexing" divide B/W transmission line into a fixed number of slots, with each slot assigned to a conversation
- Problem: lines busy based on number of conversations, not amount of information sent
  - Mother's Day
- Advantage: reserved bandwidth
  - Quality of Service guarantees are easy

Connection-Based vs. Connectionless

- Connectionless: every package of information must have an address => packets
  - Each package is routed to its destination by looking at its address
  - Analogy, the postal system (sending a letter)
  - also called "Statistical multiplexing"
  - Note: "Split phase buses" are sending packets

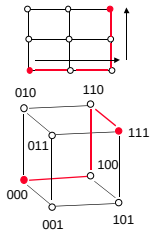
Routing Messages

- Shared Media
  - Broadcast to everyone (Ethernet)
- Switched Media needs real routing. Options:
  - Source-based routing: message specifies path to the destination (changes of direction)
  - Virtual Circuit: circuit established from source to destination, message picks the circuit to follow
  - Destination-based routing: message specifies destination, switch must pick the path
    - » deterministic: always follow same path
    - » adaptive: pick different paths to avoid congestion, failures
    - » Randomized routing: pick between several good paths to balance network load
  - Which schemes have messages arriving in order?

FTC.W99 67

Deterministic Routing Examples

- mesh: dimension-order routing
  - $(x_1, y_1) \rightarrow (x_2, y_2)$
  - first $\Delta x = x_2 - x_1$,
  - then $\Delta y = y_2 - y_1$,
- hypercube: edge-cube routing
  - $X = x_0x_1x_2 \ldots x_n \rightarrow Y = y_0y_1y_2 \ldots y_n$
  - R = X xor Y
  - Traverse dimensions of differing address in order
- tree: common ancestor
- Deadlock free?
  - Can we create a cycle from messages in-transit?



FTC.W99 68

Store and Forward vs. Cut-Through

- Store-and-forward policy: each switch waits for the full packet to arrive in switch before sending to the next switch (good for WAN)
- Cut-through routing or worm hole routing: switch examines the header, decides where to send the message, and then starts forwarding it immediately
  - In worm hole routing, when head of message is blocked, message stays strung out over the network, potentially blocking other messages (needs only buffer the piece of the packet that is sent between switches). CM-5 uses it, with each switch buffer being 4 bits per port.
  - Cut through routing lets the tail continue when head is blocked, accordioning the whole message into a single switch. (Requires a buffer large enough to hold the largest packet).

FTC.W99 69

Store and Forward vs. Cut-Through

- Advantage
  - Latency goes from-
    - » F(#hops*size) from F(#hops+size)
  - Store & Forward:
  - Cut-Through:

FTC.W99 70

Congestion Control

- Packet switched networks do not reserve bandwidth; this leads to *contention* (connection based limits input)
- Solution: prevent packets from entering until contention is reduced (e.g., freeway on-ramp metering lights)
- Options:
  - Packet discarding: If packet arrives at switch and no room in buffer, packet is discarded (e.g., UDP)
  - Flow control: between pairs of receivers and senders; use feedback to tell sender when allowed to send next packet
    - » Back-pressure: separate wires to tell to stop
    - » Window: give original sender right to send N packets before getting permission to send more; overlaps latency of interconnection with overhead to send & receive packet (e.g., TCP), adjustable window
  - Choke packets: aka "rate-based"; Each packet received by busy switch in warning state sent back to the source via choke packet. Source reduces traffic to that destination by a fixed % (e.g., ATM)

FTC.W99 71

Practical Issues for Interconnection Networks

- Standardization advantages:
  - low cost (components used repeatedly)
  - stability (many suppliers to chose from)
- Standardization disadvantages:
  - Time for committees to agree
    - » ATM packet size!!!!!
  - When to standardize?
    - » Before anything built? => Committee does design?
    - » Too early suppresses innovation
- Perfect interconnect vs. Fault Tolerant?
  - Will SW crash on single node prevent communication? (MPP typically assume perfect)
- Reliability (vs. availability) of interconnect

FTC.W99 72

Practical Issues

Interconnection	MPP	LAN	WAN
Example	CM-5	Ethernet	ATM
Standard	No	Yes	Yes
Fault Tolerance?	No	Yes	Yes
Hot Insert?	No	Yes	Yes

- Standards: required for WAN, LAN!
- Fault Tolerance: Can nodes fail and still deliver messages to other nodes? required for WAN, LAN!
- Hot Insert: If the interconnection can survive a failure, can it also continue operation while a new node is added to the interconnection? required for WAN, LAN!

FTC.W99 73

Cross-Cutting Issues for Networking

- Efficient Interface to Memory Hierarchy vs. to Network
  - SPEC ratings => fast to memory hierarchy
  - Writes go via write buffer, reads via L1 and L2 caches
- Example: 40 MHz SPARCStation(SS)-2 vs 50 MHz SS-20, no L2$ vs 50 MHz SS-20 with L2$ I/O bus latency; different generations
- SS-2: combined memory, I/O bus => 200 ns
- SS-20, no L2$: 2 busses +300ns => 500ns
- SS-20, w L2$: cache miss+500ns => 1000ns

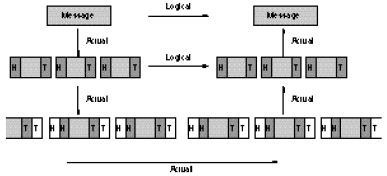
FTC.W99 74

Protocols: HW/SW Interface

- Internetworking: allows computers on independent and incompatible networks to communicate reliably and efficiently;
  - Enabling technologies: SW standards that allow reliable communications without reliable networks
  - Hierarchy of SW layers, giving each layer responsibility for portion of overall communications task, called protocol families or protocol suites
- Transmission Control Protocol/Internet Protocol (TCP/IP)
  - This protocol family is the basis of the Internet
  - IP makes best effort to deliver; TCP guarantees delivery
  - TCP/IP used even when communicating locally: NFS uses IP even though communicating across homogeneous LAN

FTC.W99 75

Protocol

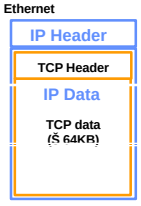


- Key to protocol families is that communication occurs logically at the same level of the protocol, called peer-to-peer, but is implemented via services at the lower level
- Danger is each level increases latency if implemented as hierarchy (e.g., multiple check sums)

FTC.W99 76

TCP/IP packet

- Application sends message
- TCP breaks into 64KB segements, adds 20B header
- IP adds 20B header, sends to network
- If Ethernet, broken into 1500B packets with headers, trailers
- Header, trailers have length field, destination, window number, version, ...



FTC.W99 77

Example Networks

- Ethernet: shared media 10 Mbit/s proposed in 1978, carrier sensing with exponential backoff on collision detection
- 15 years with no improvement; higher BW?
- Multiple Ethernets with devices to allow Ehternets to operate in parallel!
- 10 Mbit Ethernet successors?
  - FDDI: shared media (too late)
  - ATM (too late?)
  - Switched Ethernet
  - 100 Mbit Ethernet (Fast Ethernet)
  - Gigabit Ethernet

FTC.W99 78

Connecting Networks

- **Bridges: connect LANs together, passing traffic from one side to another depending on the addresses in the packet.**
  - operate at the Ethernet protocol level
  - usually simpler and cheaper than routers (no decision)
- **Routers or Gateways: these devices connect LANs to WANs or WANs to WANs and resolve incompatible addressing.**
  - Generally slower than bridges, they operate at the internetworking protocol (IP) level
  - Routers divide the interconnect into separate smaller subnets, which simplifies manageability and improves security
- **Cisco is major supplier; basically special purpose computers**

FTC.W99 79

Example Networks

	MPP	LAN	WAN
	IBM SP-2	100 Mb Ethernet	ATM
Length (meters)	10	200	100/1000
Number data lines	8	1	1
Clock Rate	40 MHz	100 MHz	155/622…
Switch?	Yes	No	Yes
Nodes (N)	Š512	Š254	-10000
Material	copper	copper	copper/fiber
Bisection BW (Mbit/s)	320xNodes	100	155xNodes
Peak Link BW (Mbits/s)	320	100	155
Measured Link BW	284	--	80

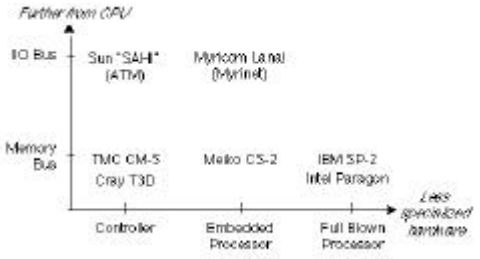
FTC.W99 80

Example Networks (cont'd)

	MPP	LAN	WAN
	IBM SP-2	100 Mb Ethernet	ATM
Latency (µsecs)	1	1.5	-50
Send+Receive Ovhd (µsecs)	39	440	630
Topology	Fat tree	Line	Star
Connectionless?	Yes	Yes	No
Store & Forward?	No	No	Yes
Congestion Control	Back-pressure	Carrier Sense	Choke packets
Standard	No	Yes	Yes
Fault Tolerance	Yes	Yes	Yes

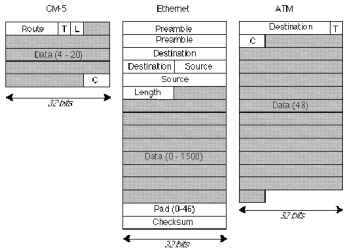
FTC.W99 81

Examples: Interface to Processor



FTC.W99 82

Packet Formats



- **Fields: Destination, Checksum(C), Length(L), Type(T)**
- **Data/Header Sizes in bytes: (4 to 20)/4, (0 to 1500)/26, 48/5**

FTC.W99 83

For your information - not going through in class - skip to summary

Example Switched LAN Performance

Network Interface	*Switch*	*Link BW*
AMD Lance Ethernet	Baynetworks EtherCell 28115	10 Mb/s
Fore SBA-200 ATM	Fore ASX-200	155 Mb/s
Myricom Myrinet	Myricom Myrinet	640 Mb/s

- **On SPARCstation-20 running Solaris 2.4 OS**
- **Myrinet is example of "System Area Network": networks for a single room or floor: 25m limit**
  - shorter => wider faster, less need for optical
  - short distance => source-based routing => simpler switches
  - Compaq-Tandem/Microsoft also sponsoring SAN, called "ServerNet"

FTC.W99 84

Example Switched LAN Performance (1995)

Switch	*Switch Latency*
Baynetworks EtherCell 28115	52.0 µsecs
Fore ASX-200 ATM	13.0 µsecs
Myricom Myrinet	0.5 µsecs

- Measurements taken from "LogP Quantyified: The Case for Low-Overhead Local Area Networks", K. Keeton, T. Anderson, D. Patterson, Hot Interconnects III, Stanford California, August 1995.

UDP/IP performance

Network	*UDP/IP roundtrip, N=8B*	*Formula*
Bay. EtherCell	1009 µsecs	+2.18*N
Fore ASX-200 ATM	1285 µsecs	+0.32*N
Myricom Myrinet	1443 µsecs	+0.36*N

- Formula from simple linear regression for tests from N = 8B to N = 8192B
- Software overhead not tuned for Fore, Myrinet; EtherCell using standard driver for Ethernet

NFS performance

Network	*Avg. NFS response*	*LinkBW/Ether*	*UDP/E.*
Bay. EtherCell	14.5 ms	1	1.00
Fore ASX-200 ATM	11.8 ms	15	1.36
Myricom Myrinet	13.3 ms	64	1.43

- Last 2 columns show ratios of link bandwidth and UDP roundtrip times for 8B message to Ethernet

Estimated Database performance (1995)

Network	*Avg. TPS*	*LinkBW/E.*	*TCP/E.*
Bay. EtherCell	77 tps	1	1.00
Fore ASX-200 ATM	67 tps	15	1.47
Myricom Myrinet	66 tps	64	1.46

- Number of Transactions per Second (TPS) for DebitCredit Benchmark; front end to server with entire database in main memory (256 MB)
  - Each transaction => 4 messages via TCP/IP
  - DebitCredit Message sizes < 200 bytes
- Last 2 columns show ratios of link bandwidth and TCP/IP roundtrip times for 8B message to Ethernet

Summary: Networking

- Protocols allow heterogeneous networking
  - Protocols allow operation in the presence of failures
  - Internetworking protocols used as LAN protocols => large overhead for LAN
- Integrated circuit revolutionizing networks as well as processors
  - Switch is a specialized computer

Parallel Computers

- Definition: "A parallel computer is a collection of processing elements that cooperate and communicate to solve large problems fast."
  Almasi and Gottlieb, *Highly Parallel Computing* ,1989
- Questions about parallel computers:
  - How large a collection?
  - How powerful are processing elements?
  - How do they cooperate and communicate?
  - How is data transmitted?
  - What type of interconnection?
  - What are HW and SW primitives for programmer?
  - Does it translate into performance?

Parallel Processors "Religion"

- The dream of computer architects since 1960: replicate processors to add performance vs. design a faster processor
- Led to innovative organization tied to particular programming models since "uniprocessors can't keep going"
  - e.g., uniprocessors must stop getting faster due to limit of speed of light: 1972, … , 1989
  - Borders religious fervor: you must believe!
  - Fervor damped some when 1990s companies went out of business: Thinking Machines, Kendall Square, ...
- Argument instead is the "pull" of opportunity of scalable performance, not the "push" of uniprocessor performance plateau

Opportunities: Scientific Computing

- Nearly Unlimited Demand (Grand Challenge):

App	*Perf (GFLOPS)*	*Memory (GB)*
48 hour weather	0.1	0.1
72 hour weather	3	1
Pharmaceutical design	100	10
Global Change, Genome	1000	1000

(Figure 1-2, page 25, of Culler, Singh, Gupta [CSG97])

- Successes in some real industries:
  - Petroleum: reservoir modeling
  - Automotive: crash simulation, drag analysis, engine
  - Aeronautics: airflow analysis, engine, structural mechanics
  - Pharmaceuticals: molecular modeling
  - Entertainment: full length movies ("A Bug's Life")
    - » You should all see A Bug's Life - it's hilarious

Opportunities: Commercial Computing

- Transaction processing & TPC-C bencmark
  - (see Chapter 1, Figure 1-4, page 28 of [CSG97])
  - small scale parallel processors to large scale
- Throughput (Transactions per minute) vs. Time (1996)

•Speedup:	1	4	8	16	32	64	112
IBM RS6000	735	1438	3119				
	1.00	*1.96*	*4.24*				
Tandem Himalaya				3043	6067	12021	20918
				1.00	*1.99*	*3.95*	*6.87*

  - IBM performance hit 1=>4, good 4=>8
  - Tandem scales: 112/16 = 7.0
- Others: File servers, electronic CAD simulation (multiple processes), WWW search engines

What level Parallelism?

- Bit level parallelism: 1970 to -1985
  - 4 bits, 8 bit, 16 bit, 32 bit microprocessors
- Instruction level parallelism (ILP): 1985 through today
  - Pipelining
  - Superscalar
  - VLIW
  - Out-of-Order execution
  - Limits to benefits of ILP?
- Process Level or Thread level parallelism; mainstream for general purpose computing?
  - Servers are parallel - spawn thread or process for each request
  - Dual-Pentium - multiprogramming

Parallel Framework Abstractions

- Layers:
  - Programming Model:
    - » Multiprogramming : lots of jobs, no communication
    - » Shared address space: communicate via memory
    - » Message passing: send and receive messages
    - » Data Parallel: several agents operate on several data sets simultaneously and then exchange information globally and simultaneously (shared or message passing)
  - Communication Abstraction:
    - » Shared address space: e.g., load, store, atomic swap
    - » Message passing: e.g., send, receive library calls
    - » Debate over this topic (ease of programming, scaling) => many hardware designs 1:1 programming model

Performance Metrics

- Memory Latency
- Memory Bandwidth
- Scalability

Architecture Below

Physical Setup of Computers

- UMA
  - Uniform Memory Access
  - Bus-based machine
- NUMA
  - Non-Uniform Memory Access
- Clumps
- New research in asymmetric processing units
  - not all processing units are created equal

Logical Addressing

- Multicomputers
  - Processors send messages to pass data
- Shared Memory
  - Each processor can access any memory

Note: Architecture may be decoupled from programming paradigm

Shared Address Model Summary

- Each processor can name every physical location in the machine
- Each process can name all data it shares with other processes
- Data transfer via load and store
- Data size: byte, word, ... or cache blocks
- Uses virtual memory to map virtual to local or remote physical
- Memory hierarchy model applies: now communication moves data to local processor cache (as load moves data from memory to cache)
  - What problem does this present?

What's is the final value of c?

Initially, c = 90

Proc A:		Proc B:	
ld	c	ld	c
add	c,c,1	add	c,c,1
st	c	st	c

Scenario 1: Scenario 2: Scenario 3: