ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG

Algorithms and Methods for Distributed Storage

6 Networking

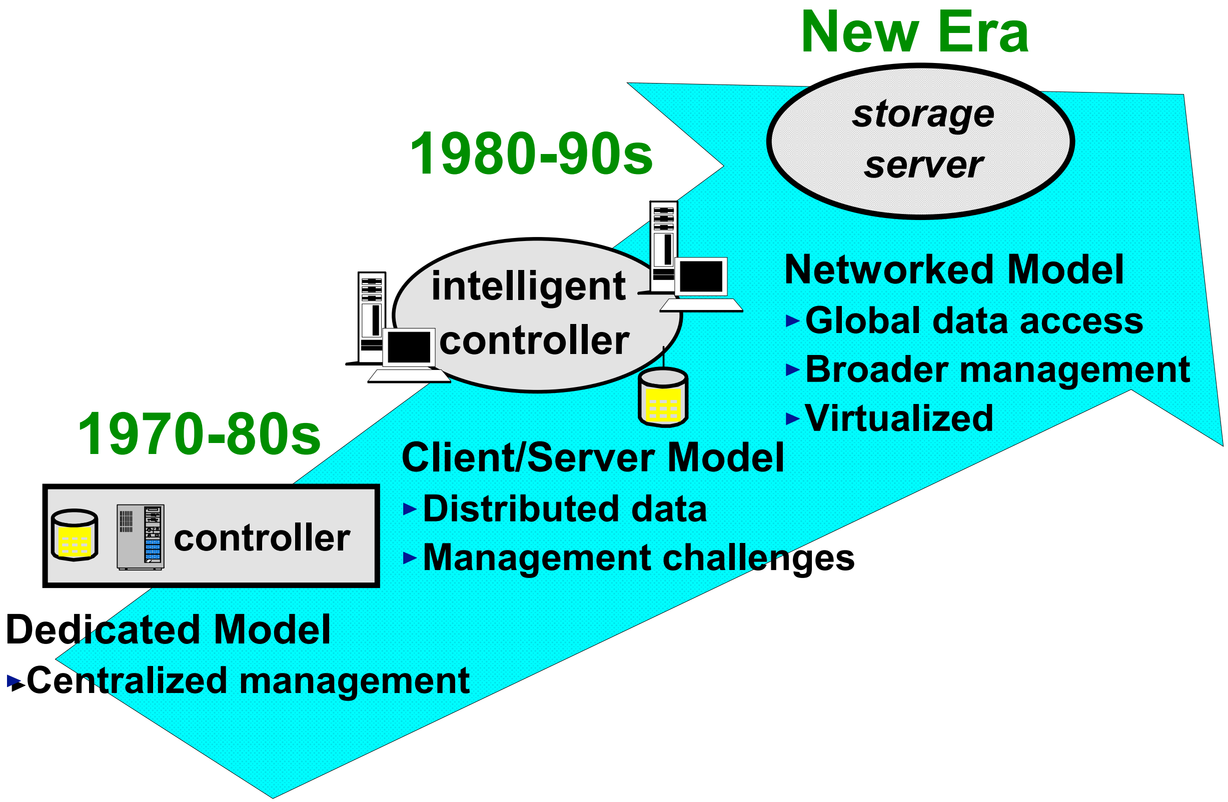
Stefan Rührup

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Wintersemester 2008/09

CoNe Freiburg

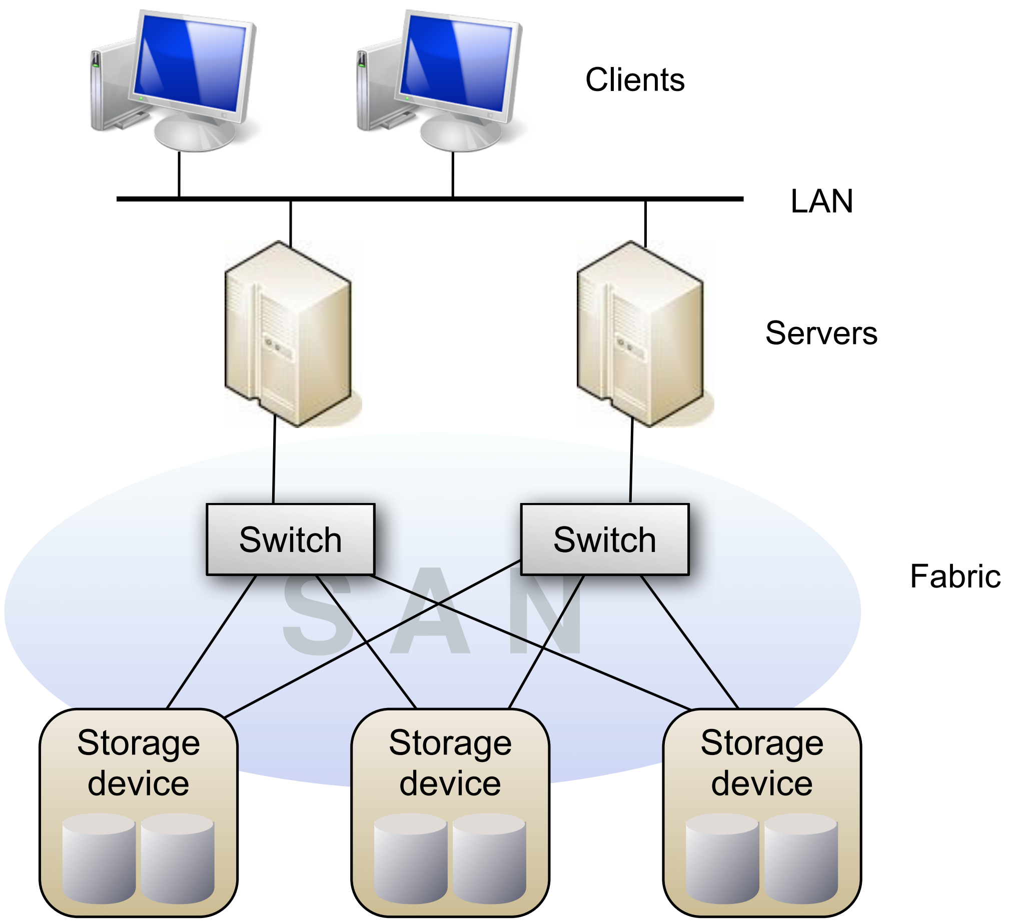
IIF INSTITUT FÜR INFORMATIK FREIBURG

Evolution of Storage



[Tate, Lucchese, Moore: Introduction to Storage Area Networks, IBM 2006]

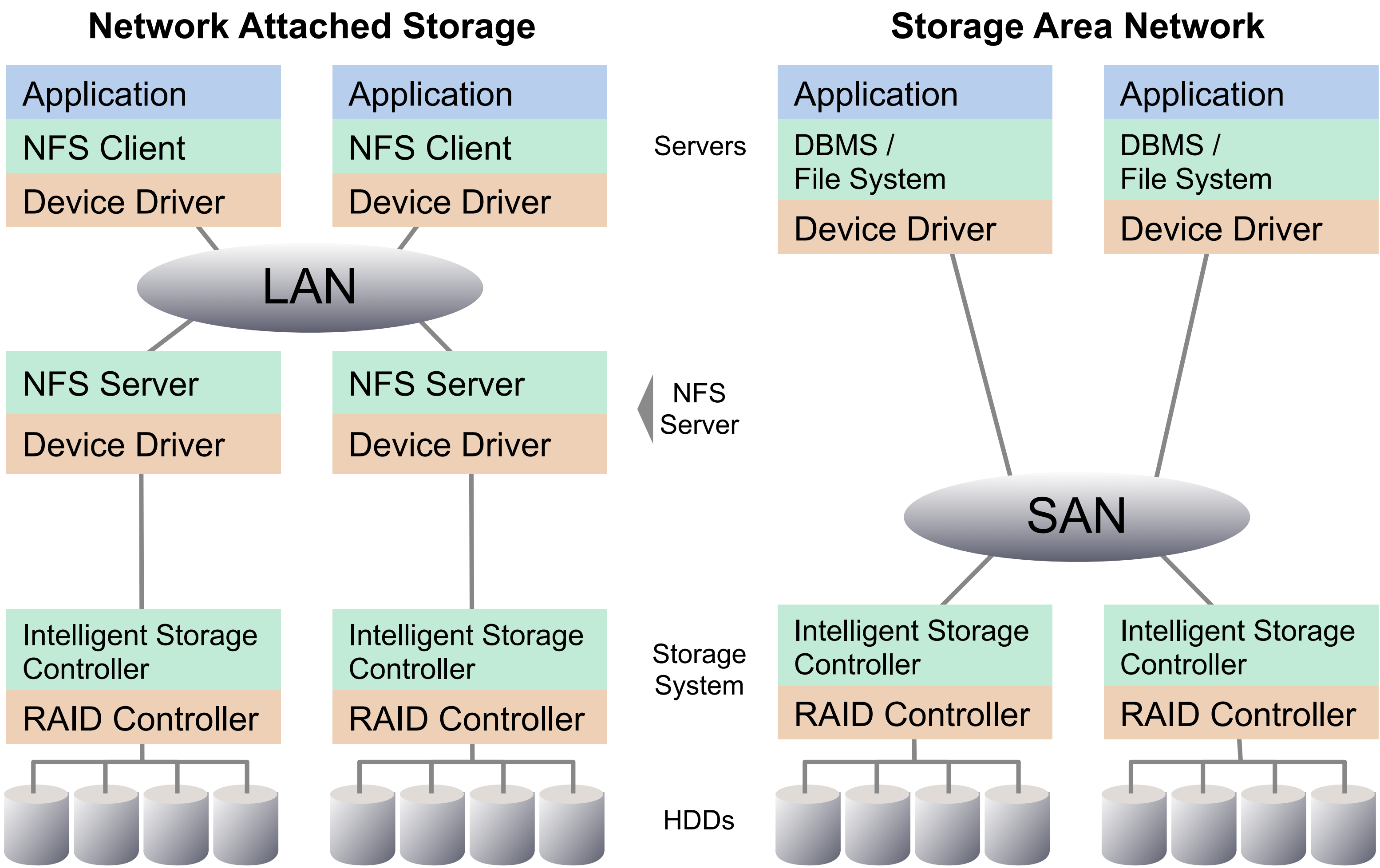
Storage Area Network



NAS and SAN

- **Network-attached Storage**
  - storage device attached to a network
  - access through NFS, AFS, SMB, etc. (file level)
- **Storage Area Network**
  - storage system of interconnected storage devices
  - access through FCP, iFCP, iSCSI (block level)

NAS and SAN



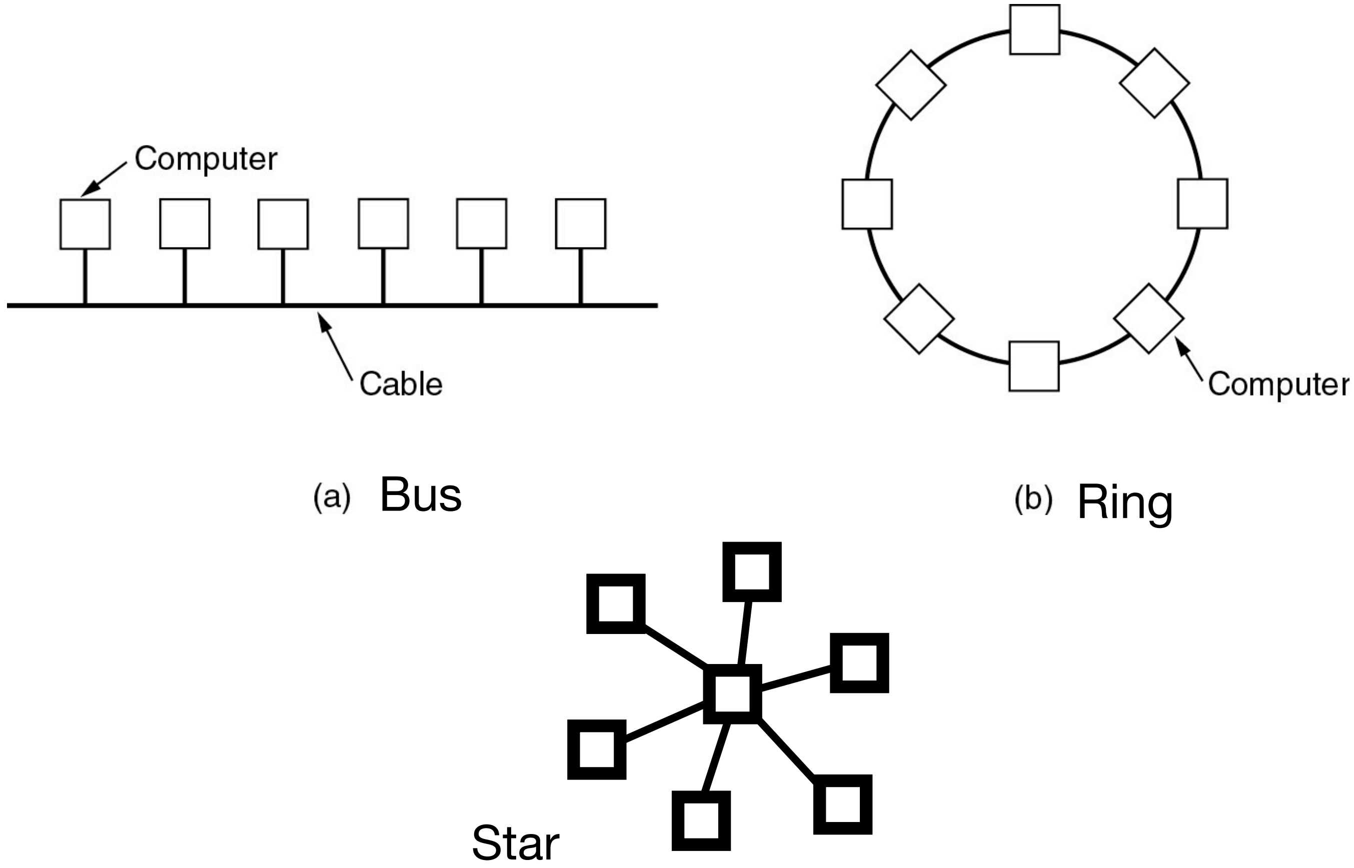
[Morris, Truskowski: The evolution of storage systems, IBM Systems Journal, 42(2), 2003]

Types of Networks

Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	Local area network
1 km	Campus	Local area network
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	Wide area network
10,000 km	Planet	The Internet

(Aus Tanenbaum)

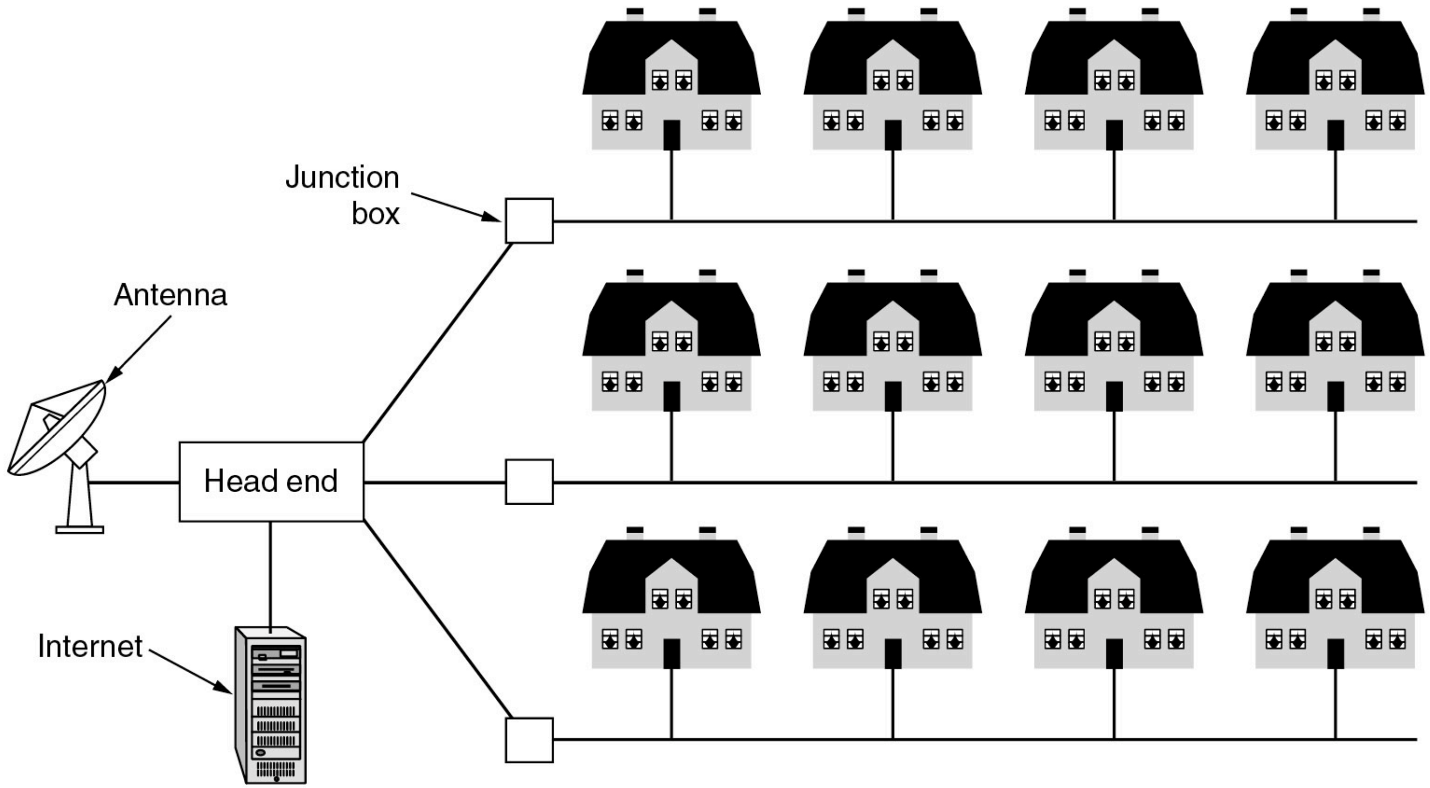
Local Area Networks (LAN)



(Aus Tanenbaum)

Metropolitan Area Networks

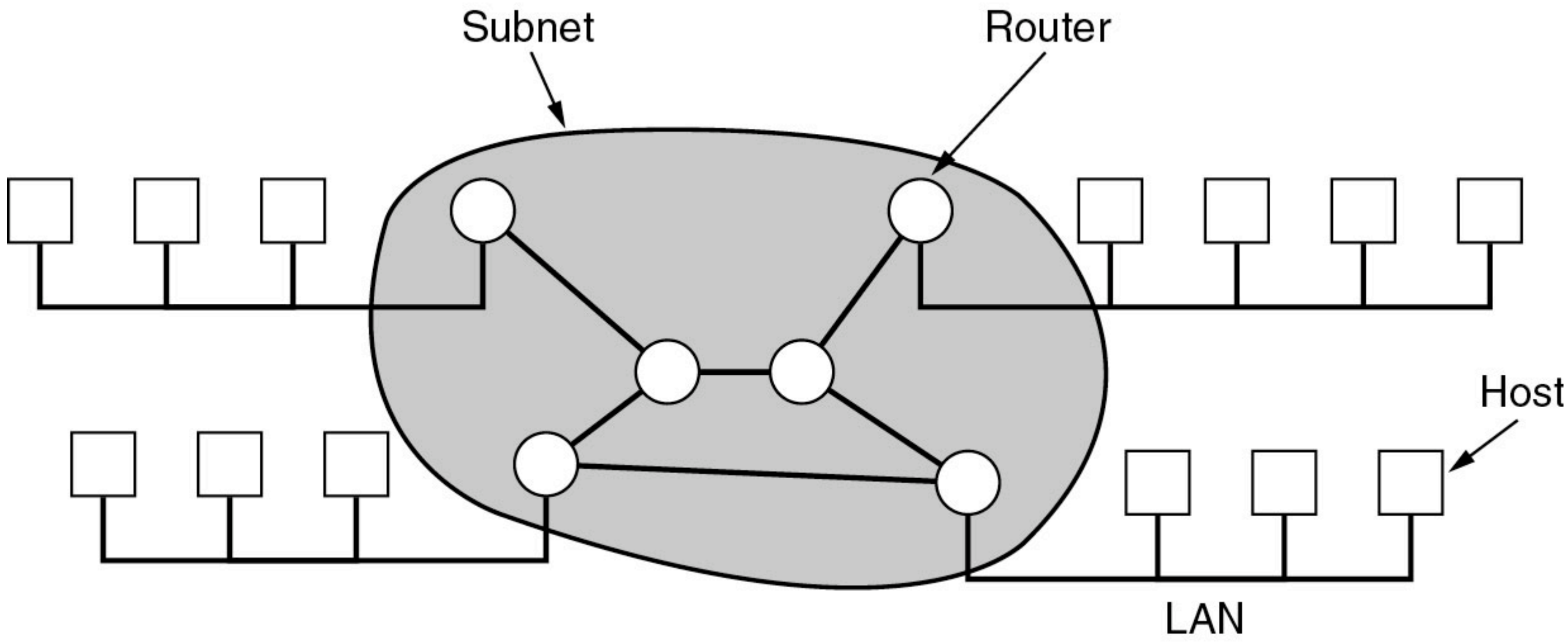
- **e.g. cable TV**



(Aus Tanenbaum)

Wide Area Networks

- **Interconnection of LANs**

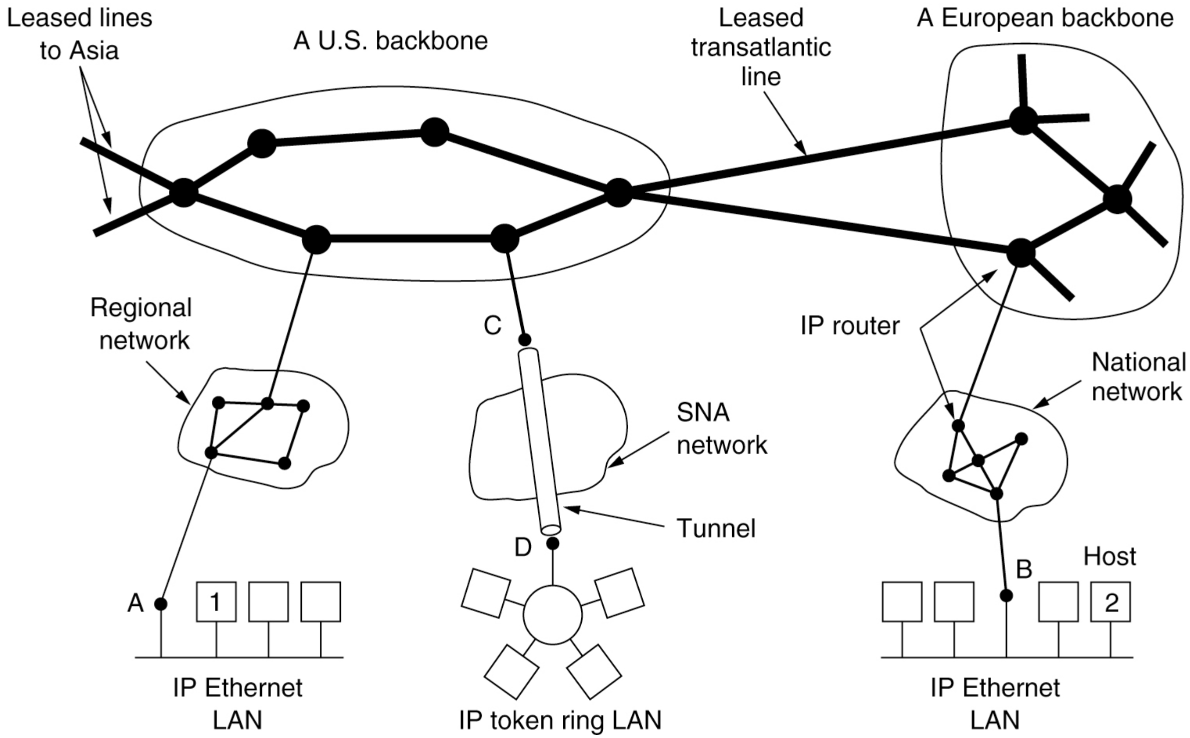


[Tanenbaum, Computer Networks]

(Aus Tanenbaum)

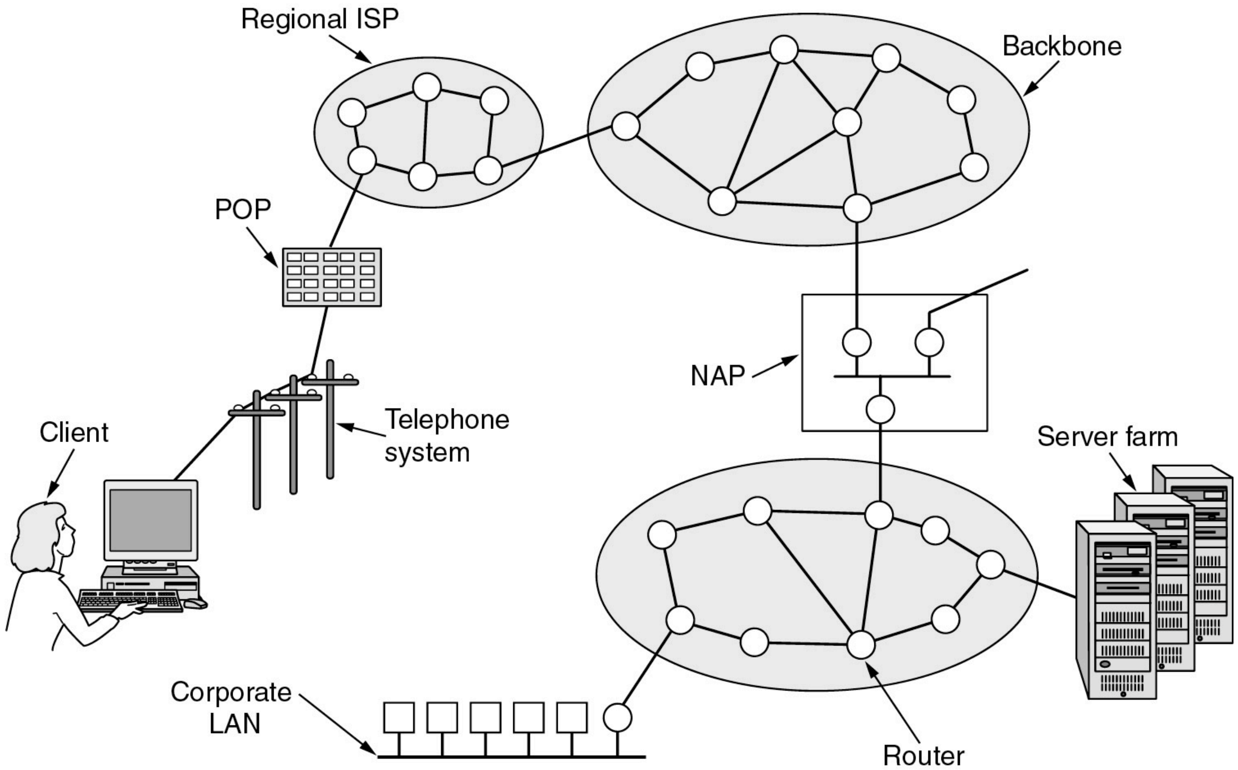
The Internet

- global system of interconnected WANs and LANs
- open, system-independent, no global control



[Tanenbaum,
Computer Networks]

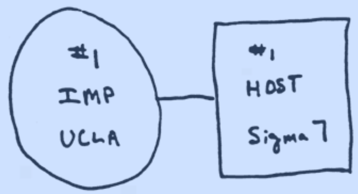
Interconnection of Subnetworks



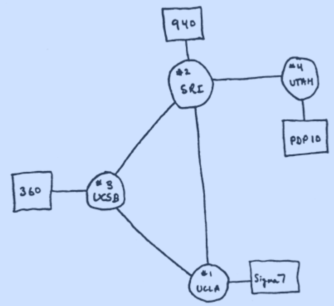
[Tanenbaum, Computer Networks]

History of the Internet

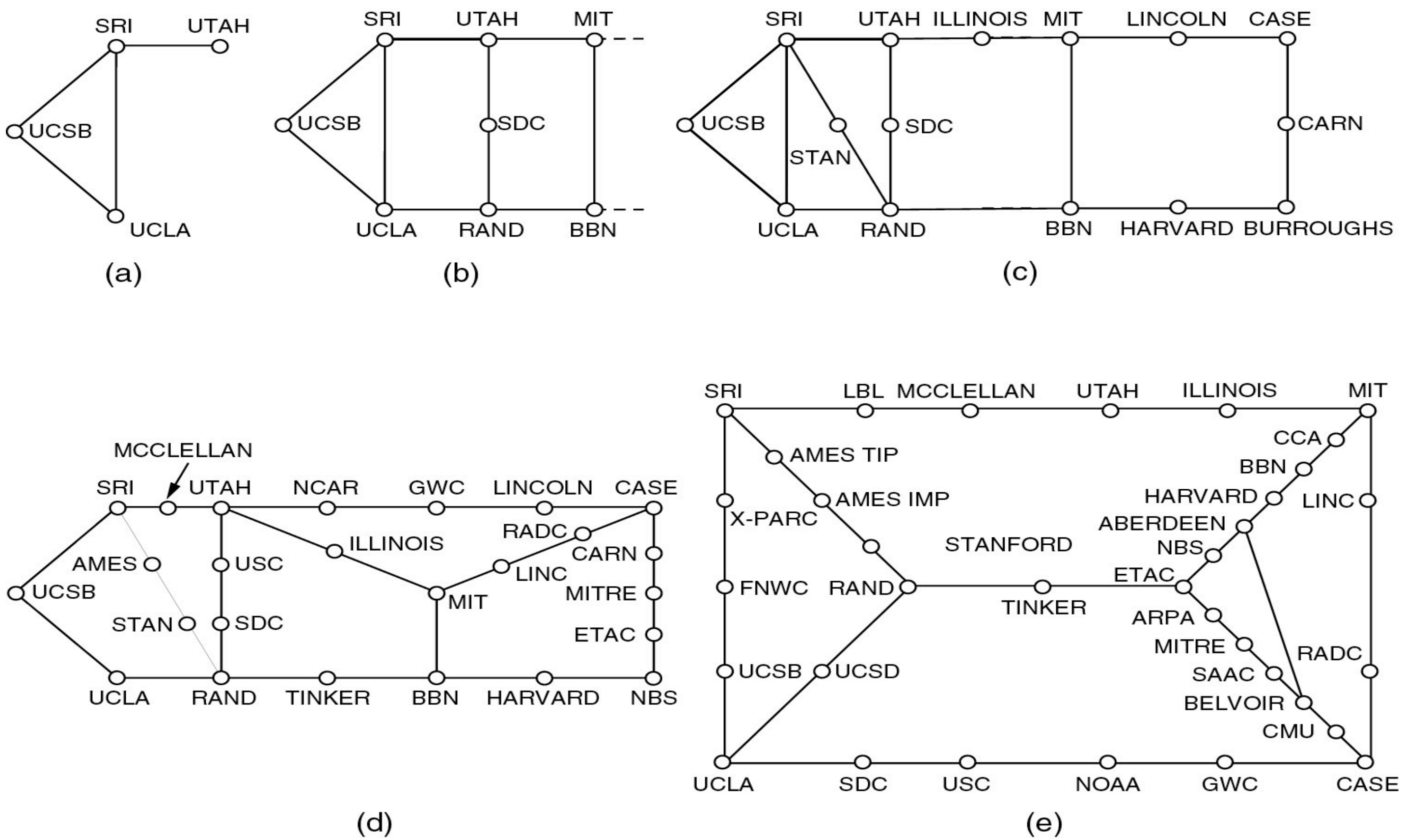
- **1961: Packet Switching Theory**
  - Leonard Kleinrock, MIT, “Information Flow in Communication Nets”
- **1962: Concept of a “Galactic Network”**
  - J.C.R. Licklider and W. Clark, MIT, “On-Line Man Computer Communication”
- **1965: Predecessor of the Internet**
  - Analog modem connection between 2 computers in the USA
- **1967: Concept of the “ARPANET”**
  - Concept of Larry Roberts
- **1969: 1st node of the “ARPANET”**
  - at UCLA (Los Angeles)
  - end 1969: 4 computers connected



conceptual sketches of the original internet



ARPANET



ARPANET (a) December 1969
(b) July 1970

(c) March 1971 (d) April 1972
(e) September 1972

An Open Network Architecture

- **Concept of Robert Kahn (DARPA 1972)**
  - Local networks are autonomous
    - independent
    - no WAN configuration
  - **packet-based** communication
  - “**best effort**” communication
    - if a packet cannot reach the destination, it will be deleted
    - the application will re-transmit
  - black-box approach to connections
    - black boxes: gateways and routers
    - packet information is not stored
    - no flow control
  - no global control
- **Basic principles of the Internet**

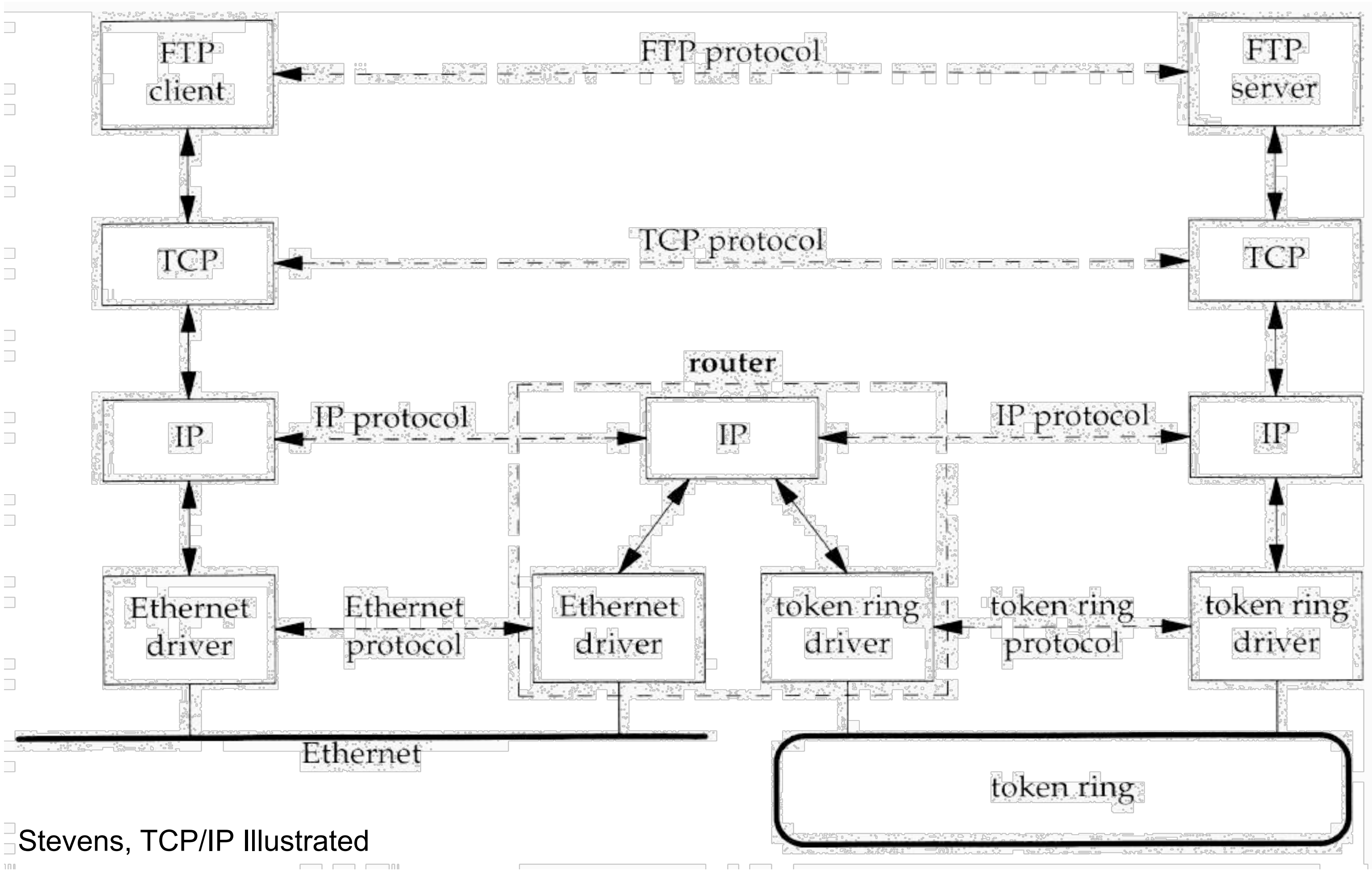
Protocols of the Internet

Application	Telnet, FTP, HTTP, SMTP (E-Mail), ...
Transport	TCP (Transmission Control Protocol) UDP (User Datagram Protocol)
Network	IP **(Internet Protocol)** + ICMP **(Internet Control Message Protocol)** + IGMP **(Internet Group Management Protoccol)**
Host-to-Network	LAN **(e.g. Ethernet, Token Ring etc.)**

TCP/IP Layers

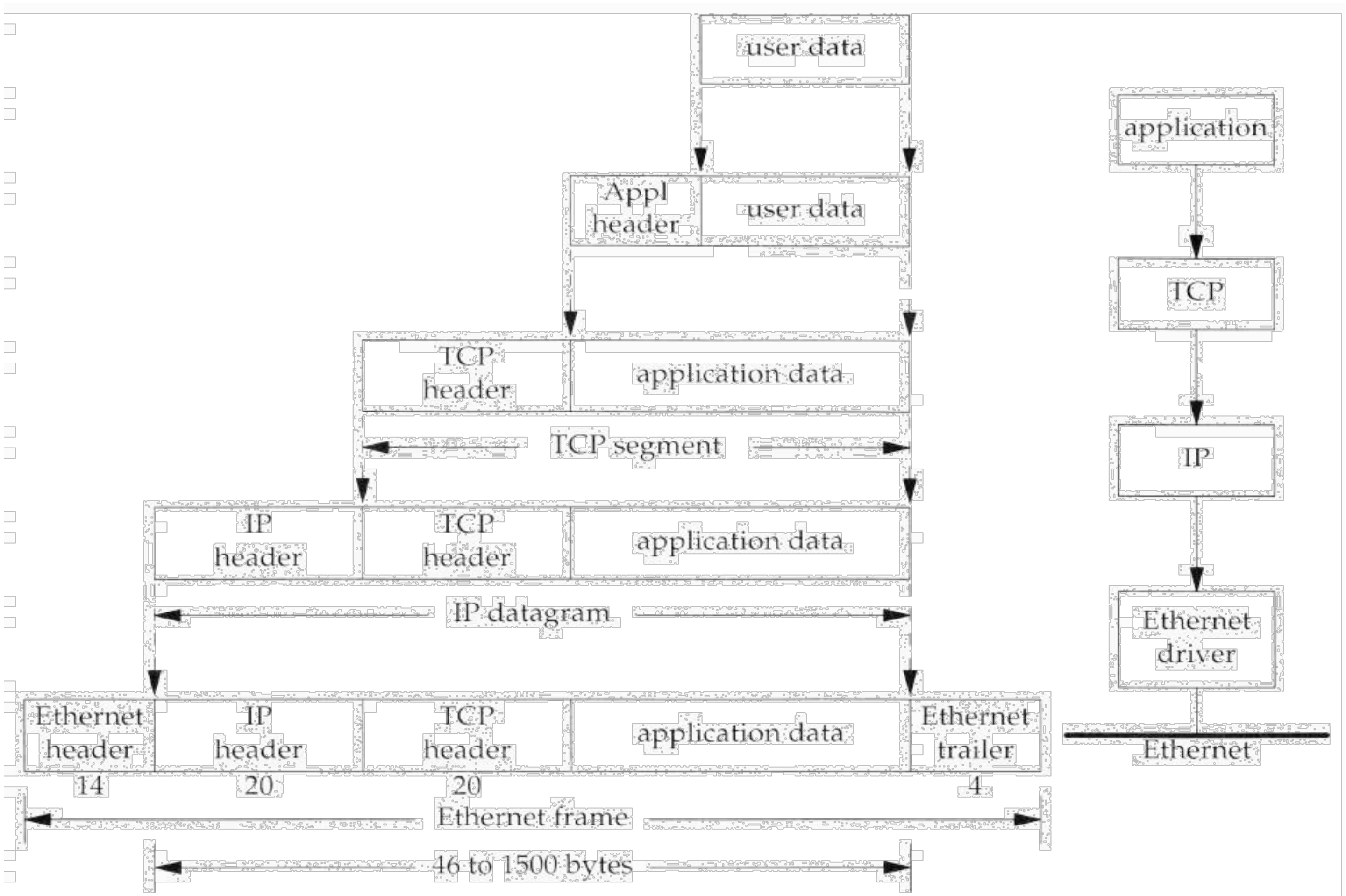
- **1. Host-to-Network**
  - Not specified, depends on the local networ,k e.g. Ethernet, WLAN 802.11, PPP, DSL
- **2. Routing Layer/Network Layer (IP - Internet Protocol)**
  - Defined packet format and protocol
  - Routing
  - Forwarding
- **3. Transport Layer**
  - TCP (Transmission Control Protocol)
    - Reliable, connection-oriented transmission
    - Fragmentation, Flow Control, Multiplexing
  - UDP (User Datagram Protocol)
    - hands packets over to IP
    - unreliable, no flow control
- **4. Application Layer**
  - Services such as TELNET, FTP, SMTP, HTTP, NNTP (for DNS), ...

Example: Routing between LANs



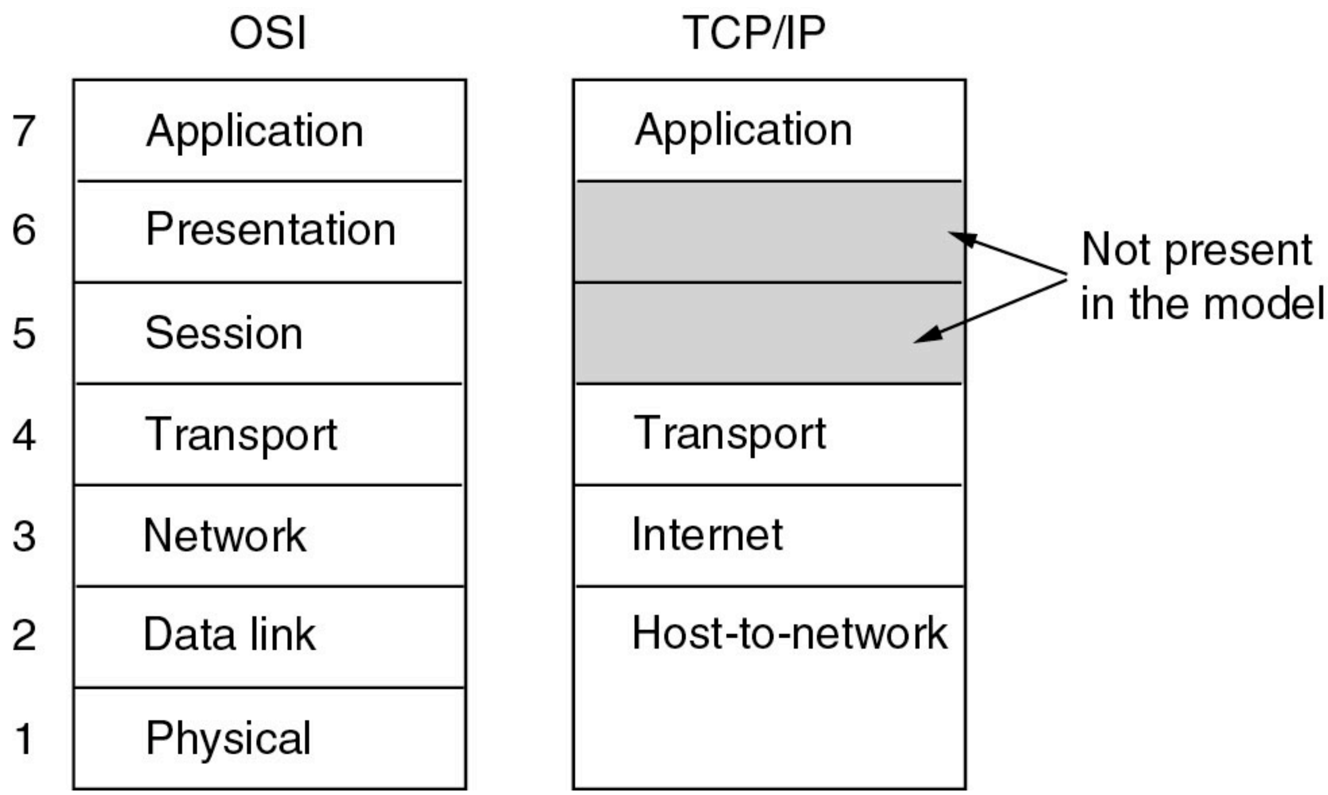
Stevens, TCP/IP Illustrated

Data/Packet Encapsulation



Stevens, TCP/IP Illustrated

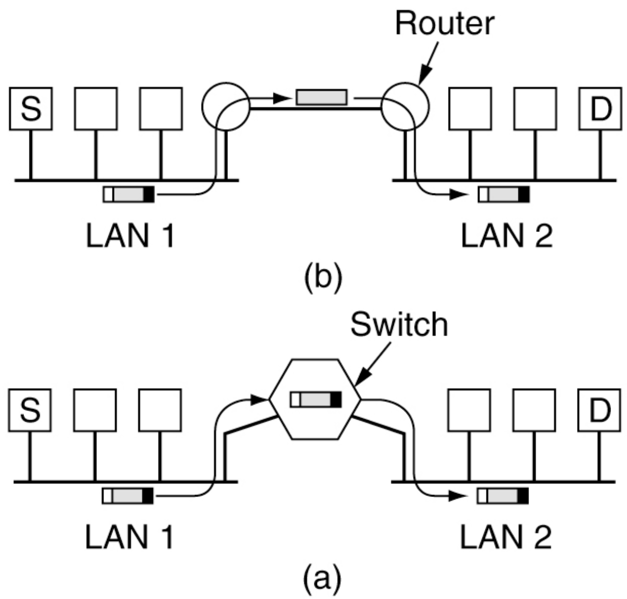
Reference Models: OSI versus TCP/IP



(Aus Tanenbaum)

Network Interconnections

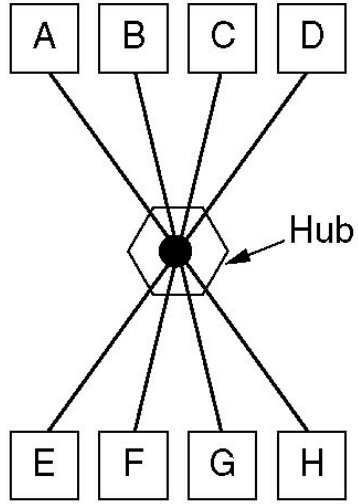
Application layer	Application gateway
Transport layer	Transport gateway
Network layer	Router
Data link layer	Bridge, switch
Physical layer	Repeater, hub

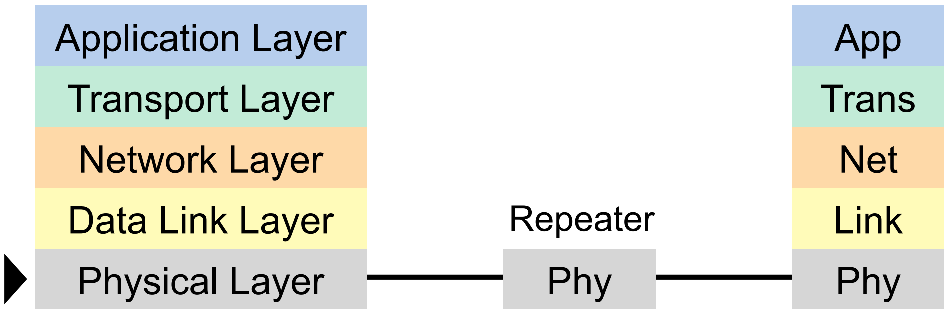


[Tanenbaum, Computer Networks]

Repeater and Hub

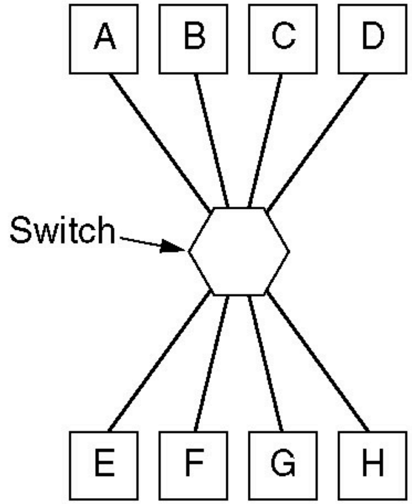
- **Receives, aplifies, re-transmits**
  - only on the signal level
  - Information remains untouched





Switch

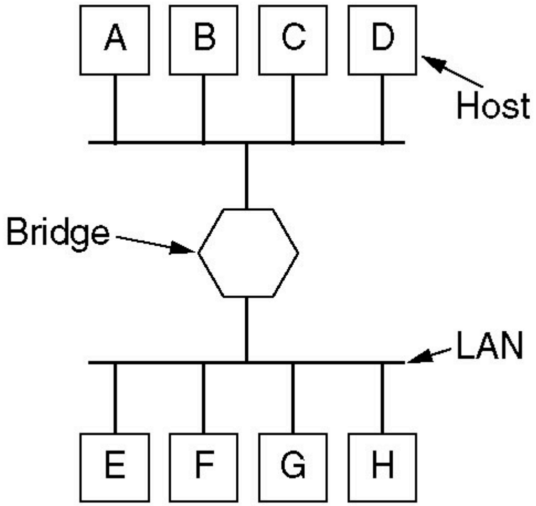
- **Connection of multiple network segments**
  - frames are forwarded only to the target segment
  - collisions are not repeated
  - store & forward (w. error correction)
  - cut through switching: forwarding starts after the header is read

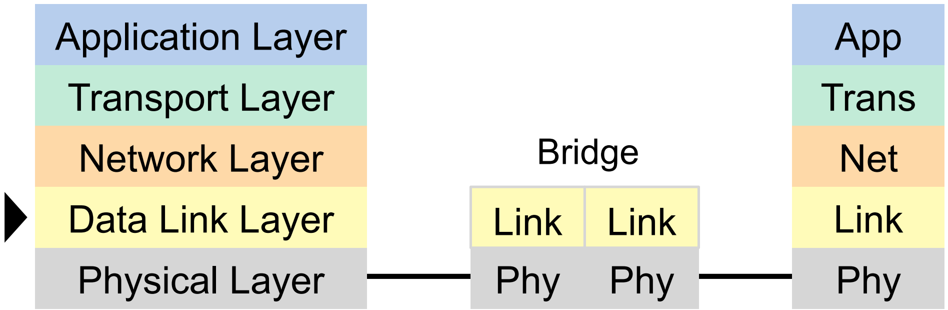


Application Layer
Transport Layer
Network Layer
▶ Data Link Layer
Physical Layer

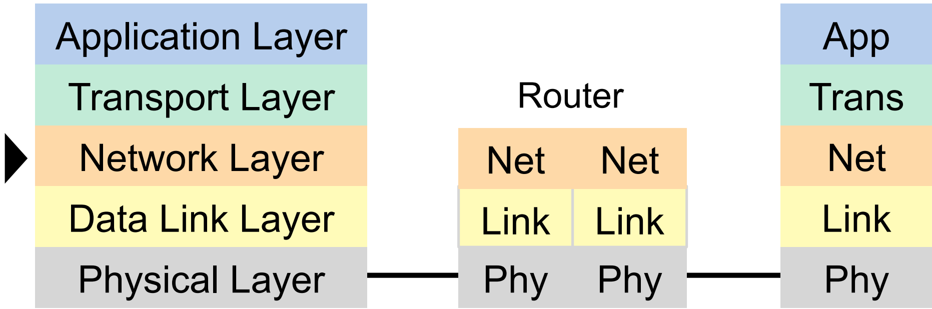
Bridge

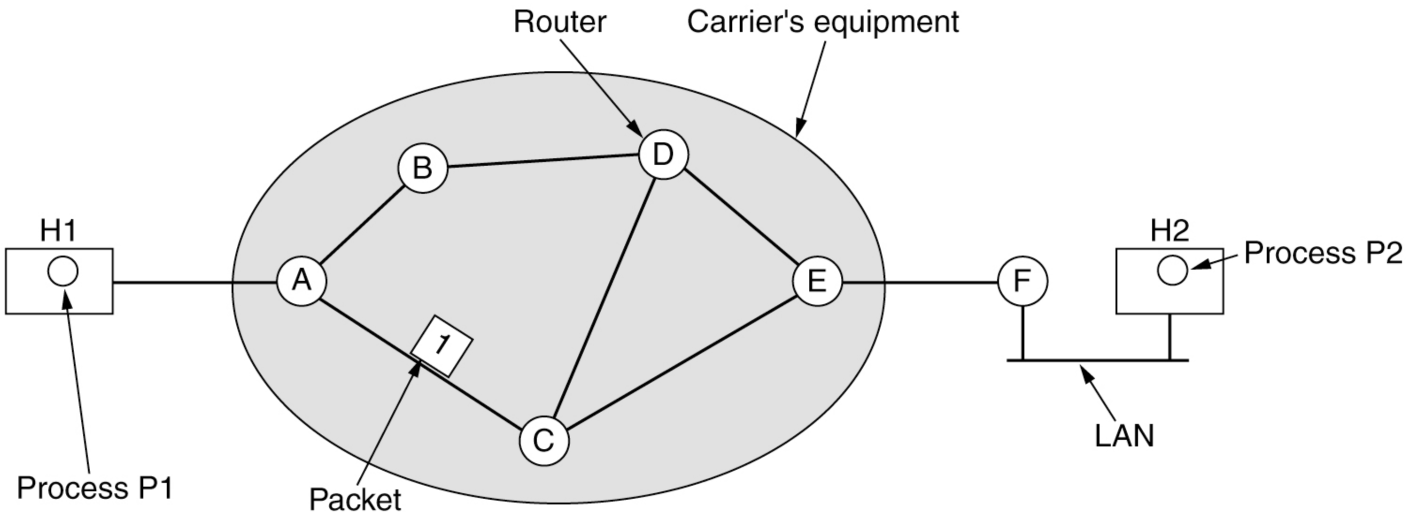
- **Connection of two network segments**
  - different access methods
  - multiport bridge similar to switch





Routing


Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer
Router
Net
Net
Link
Link
Phy
Phy
App
Trans
Net
Link
Phy


Router
Carrier's equipment
B
D
H1
A
E
F
H2
Process P2
1
C
LAN
Process P1
Packet

Why do we need a network layer?

- **Local Networks can be connected by hubs, switches, bridges**
  - Problems:
    - Hubs propagate collisions
    - Switching: Inefficient collection of routing information
    - Problem of broadcasting
    - Internet connects >> 10 Mio. local networks

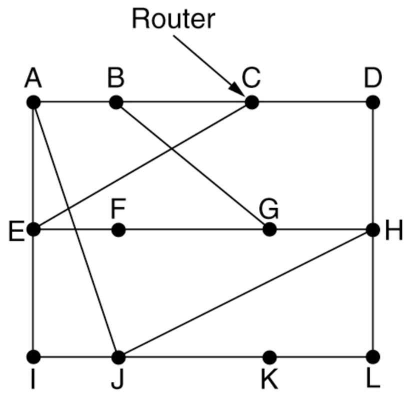
- **In large networks, routing information becomes necessary**
  - How is it collected?
  - How are packets forwarded?

Routing Tables and Packet Forwarding

- **IP Routing Table**
  - contains for each destination the address of the next gateway
  - destination: host computer or sub-network
  - default gateway

- **Packet Forwarding**
  - IP packet (datagram) contains start IP address and destination IP address
    - if destination = my address then hand over to higher layer
    - if destination in routing table then forward packet to corresponding gateway
    - if destination IP subnet in routing table then forward packet to corresponding gateway
    - otherwise, use the default gateway

Routing Table (Distance Vector)



(a)

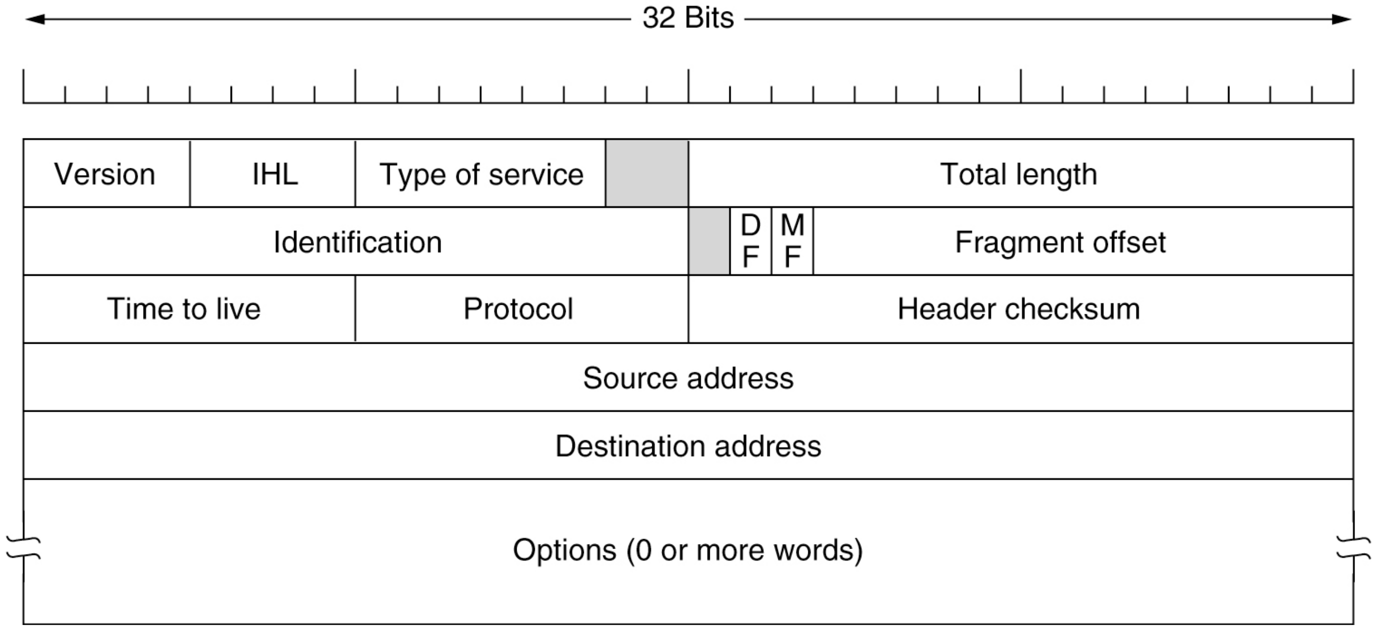
To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

[Tanenbaum, Computer Networks]

(b)

IPv4 Packet Header



IP Packet Forwarding

- **IP -Paket (datagram) contains...**
  - TTL (Time-to-Live): Hop count limit
  - Start IP Address
  - Destination IP Address
- **Packet Handling**
  - Reduce TTL (Time to Live) by 1
  - If TTL ≠ 0 then forward packet according to routing table
  - If TTL = 0 or forwarding error (buffer full etc.):
    - delete packet
    - if packet is not an ICMP Packet then
      - sende ICMP Packet with
        - start = current IP Address
        - destination = original start IP Address

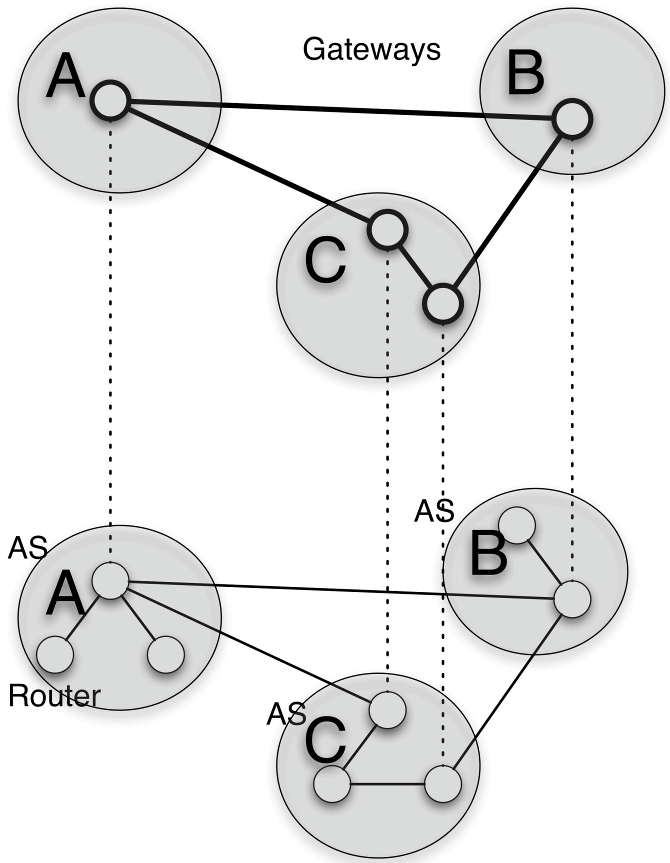
Static and Dynamic Routing

- **Static Routing**
  - Routing table created manually
  - used in small LANs

- **Dynamic Routing**
  - Routing table created by Routing Algorithm
  - **Centralized**, e.g. Link State
    - Router knows the complete network topology
  - **Decentralized**, e.g. Distance Vector
    - Router knows gateways in its local neighborhood

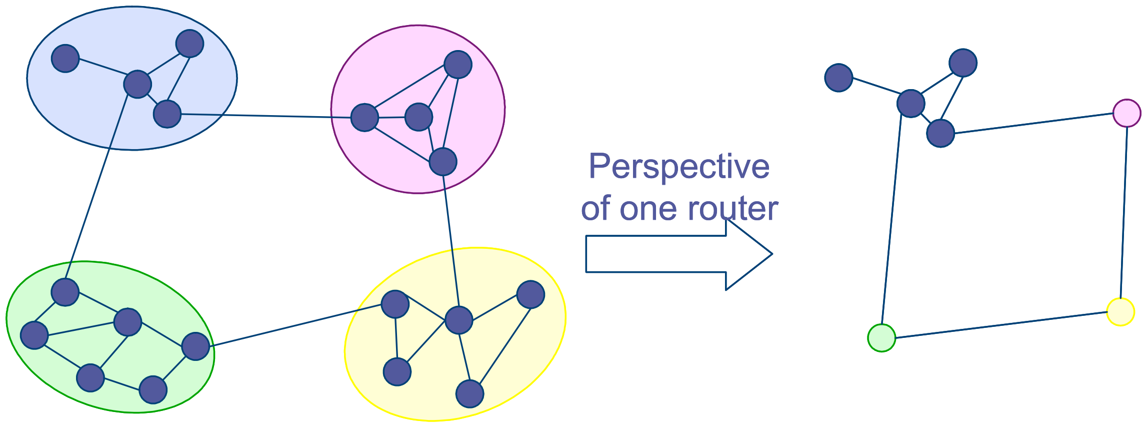
Hierarchical Routing

- **Internet consists of Autonomous Systems (AS)**
  - example: uni-freiburg.de
- **Intra-AS-Routing (Interior Gateway Protocol)**
  - z.B. RIP, OSPF, IGRP, ...
- **Inter-AS-Routing (Exterior Gateway Protocol)**
  - between Gateways
  - decentralized
  - everybody can define a metric
  - z.B. BGP



Hierarchical Addressing

- **MAC Adresses contain no structural information**



- **Hierarchical Adressing**
  - Routing simplified by using a hierarchical structure for addressing
  - Group-ID$_n$:Group-ID$_{n-1}$:…:Group-ID$_1$:Device-ID

Intra-AS Routing

- **Inter-AS**
  - Routing Information Protocol (RIP)
    - Distance Vector Algorithmus
    - Metric = hop count
    - exchange of distance vectors (by UDP)
  - Interior Gateway Routing Protocol (IGRP)
    - successor of RIP
    - different routing metrics (delay, bandwidth)
  - Open Shortest Path First (OSPF)
    - Link State Routing (every router knows the topology)
    - Route calculation by Dijkstra’s shortest path algorithm

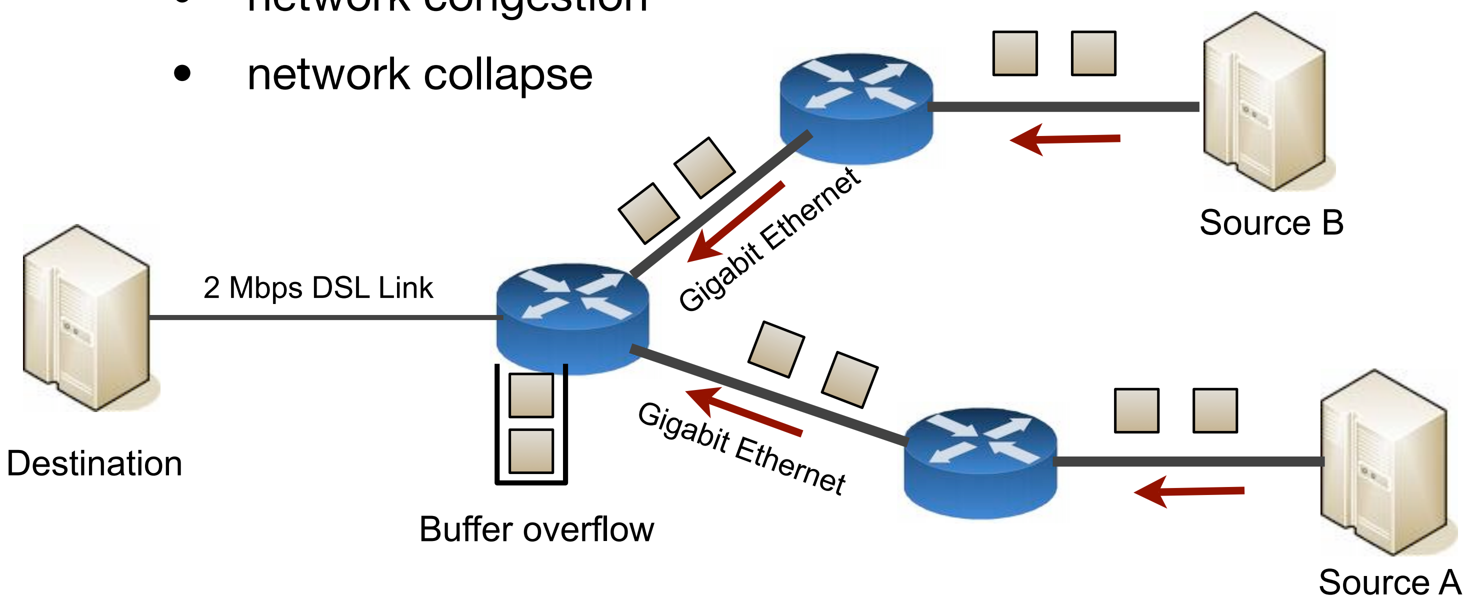
Inter-AS Routing

- **Problems of Inter-AS Routing**
  - AS may reject packets
  - Political consideration: Routing through other contries?
  - Routing metrics of different AS are not compatible
    - path optimization impossible
    - Inter-AS Routing tries to achieve reachability
  - Currently, Inter-Domain Router know more than 140.000 Networks

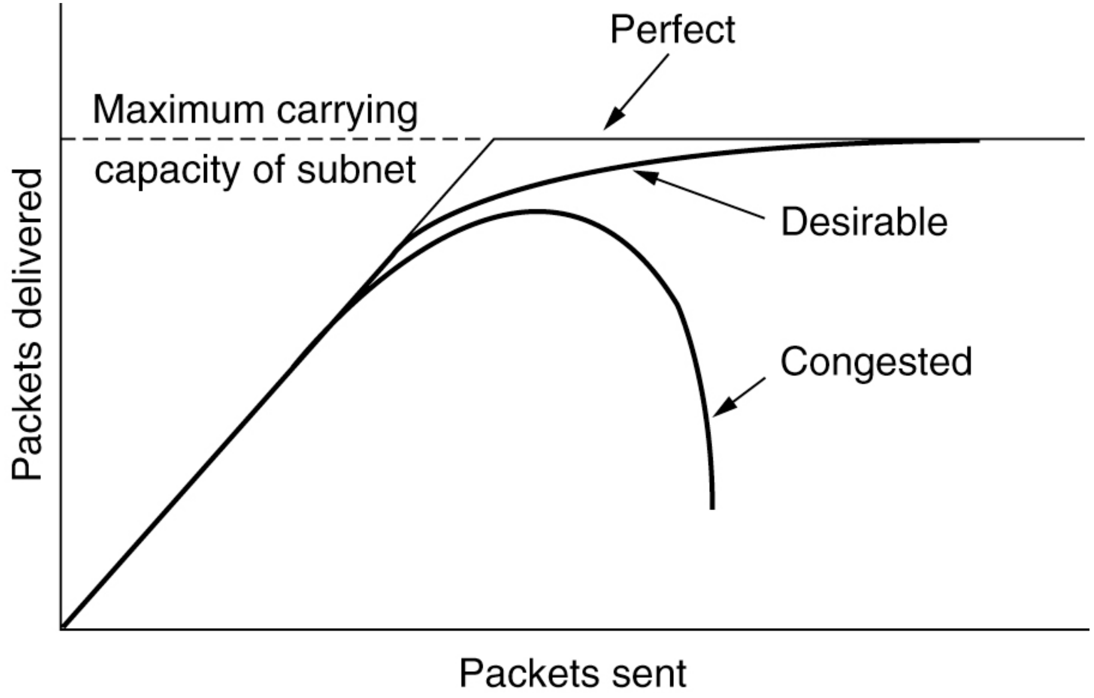
- **Border Gateway Protocol (BGP)**
  - Path-Vector Protocol

Network Congestion

- **(Sub-)Networks have limited bandwidth**
- **Injecting too many packets leads to**
  - network congestion
  - network collapse



Congestion and capacity

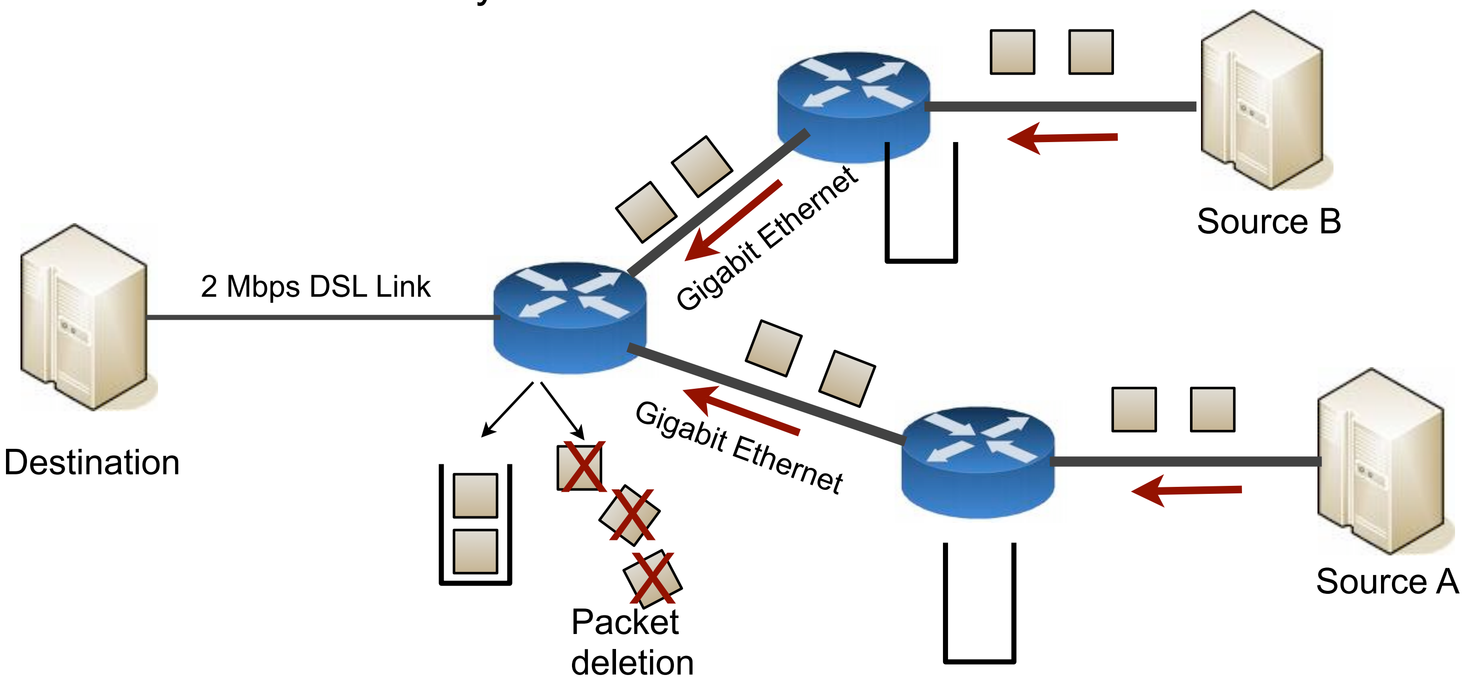


Congestion Prevention

Layer	Policies
Transport	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy • Timeout determination
Network	• Virtual circuits versus datagram inside the subnet • Packet queueing and service policy • Packet discard policy • Routing algorithm • Packet lifetime management
Data link	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy

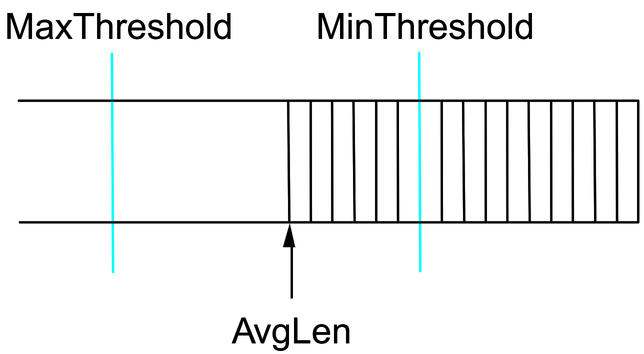
Congestion Prevention by Routers

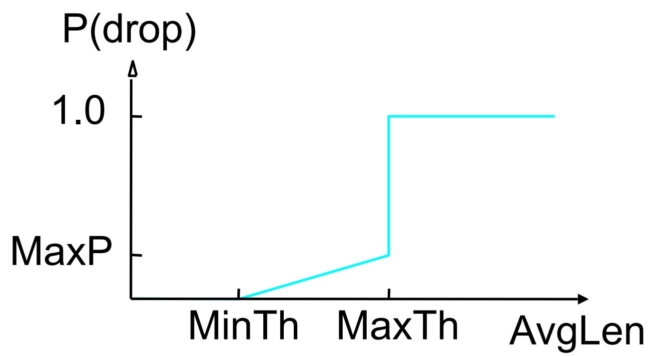
- **IP Routers drop packets**
  - Tail dropping
  - Random Early Detection



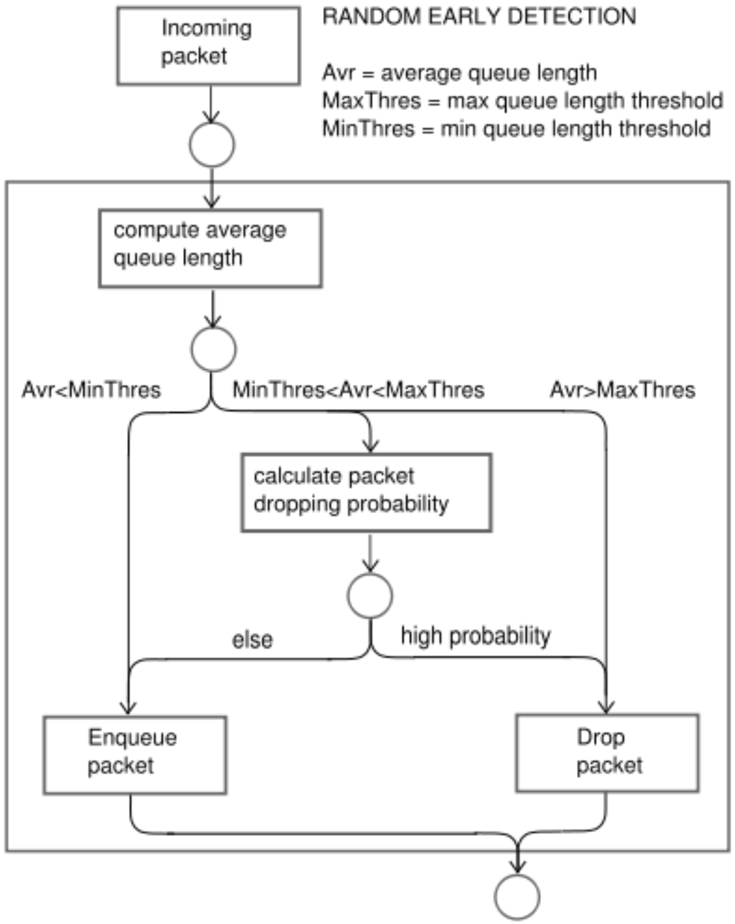
Random early detection (RED)

- Packet dropping probability grows with queue length
- Fairer than just “tail dropping”: the more a host transmits, the more likely it is that its packets are dropped



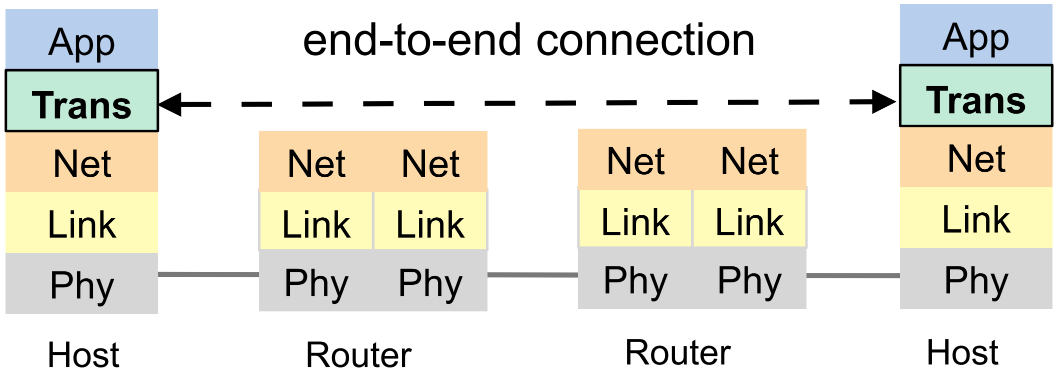


Random Early Detection (RED)



The Transport Layer

- **TCP (Transmission Control Protocol**
  - connection-oriented
  - delivers a stream of bytes
  - reliable and ordered
- **UDP (User Datagram Protocol)**
  - delivery of datagrams
  - connectionless, unreliable, unordered



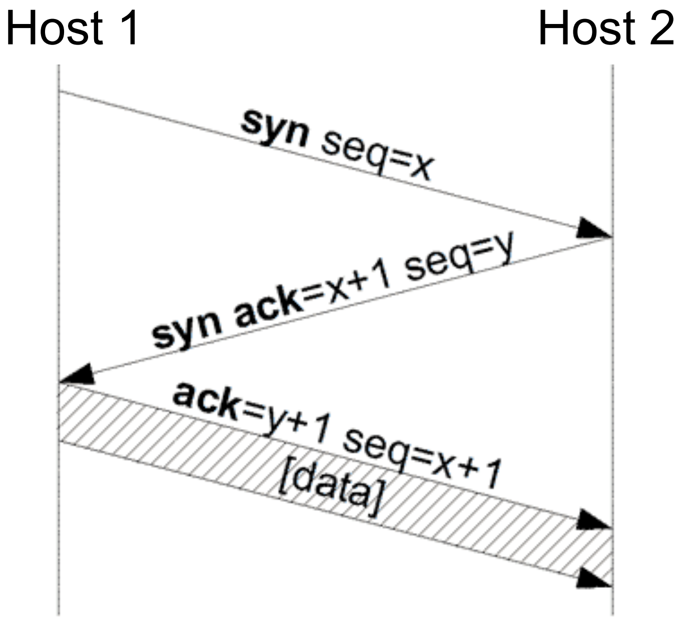
The Transmission Control Protocol (TCP)

- Connection-oriented
- Reliable delivery of a byte stream
  - fragmentation and reassembly (*TCP segments*)
  - acknowledgements and retransmission
- In-order delivery, duplicate detection
  - sequence numbers
- Flow control and congestion control
  - window-based (receiver window, congestion window)

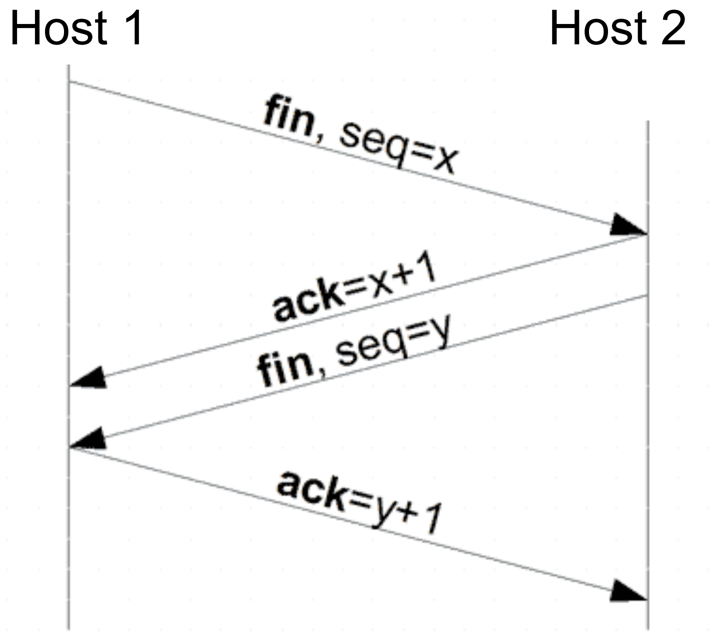
- **challenge**: IP (network layer) packets can be dropped, delayed, delivered out-of-order ...

TCP Connections

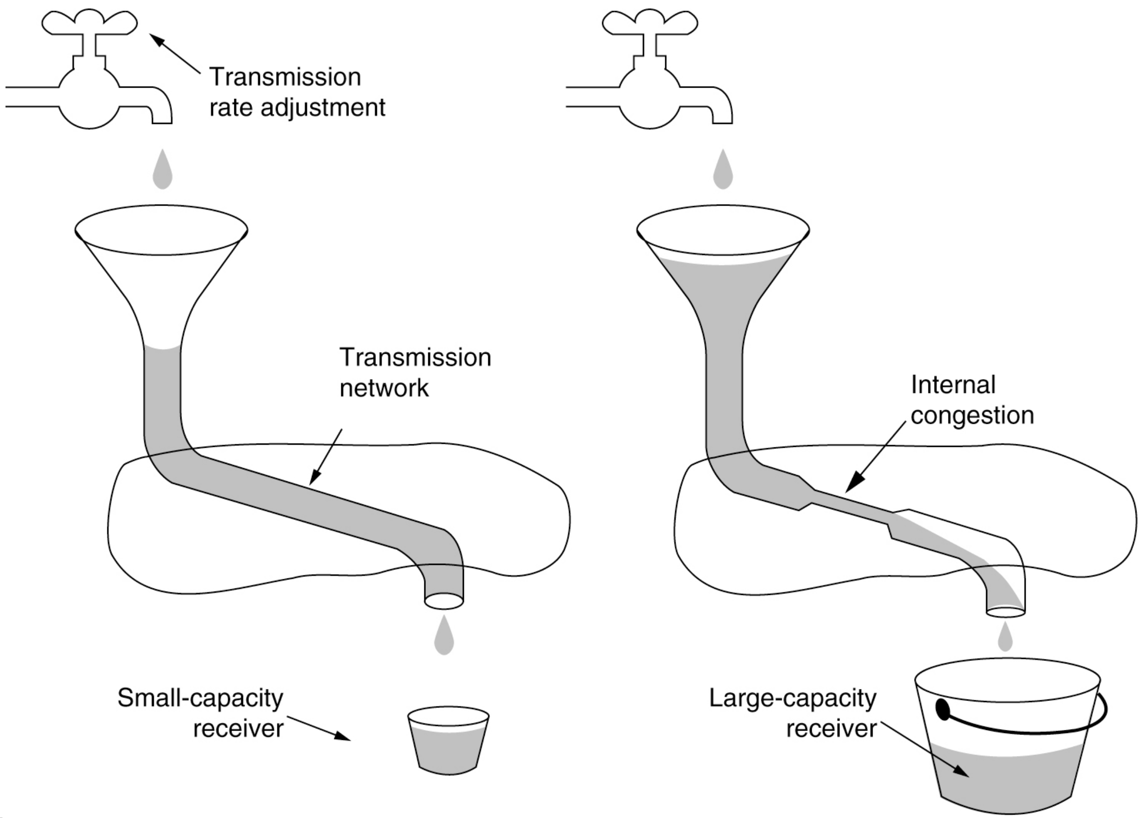
Connection establishment



Connection termination



Flow control and congestion control

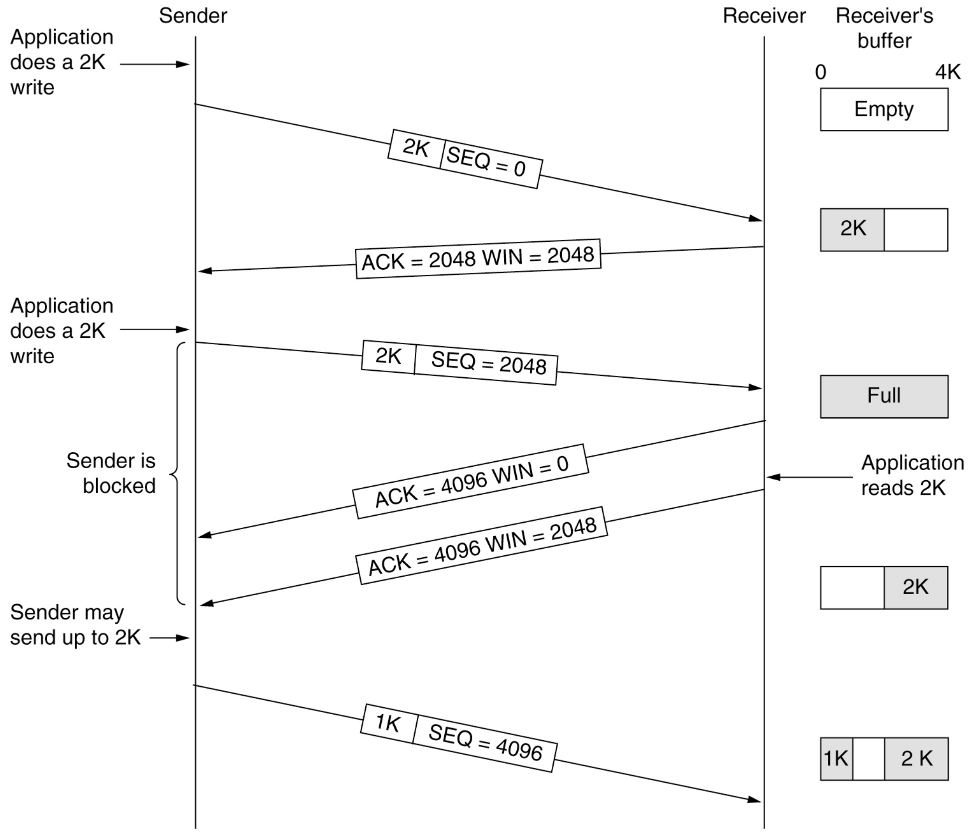


[Tanenbaum,
Computer Networks]

(a) (b)

Flow Control

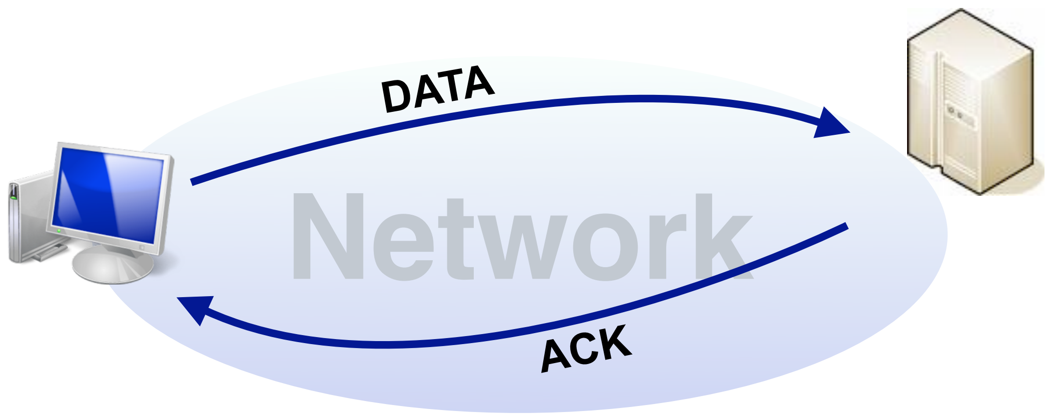
acknowledgements and window management



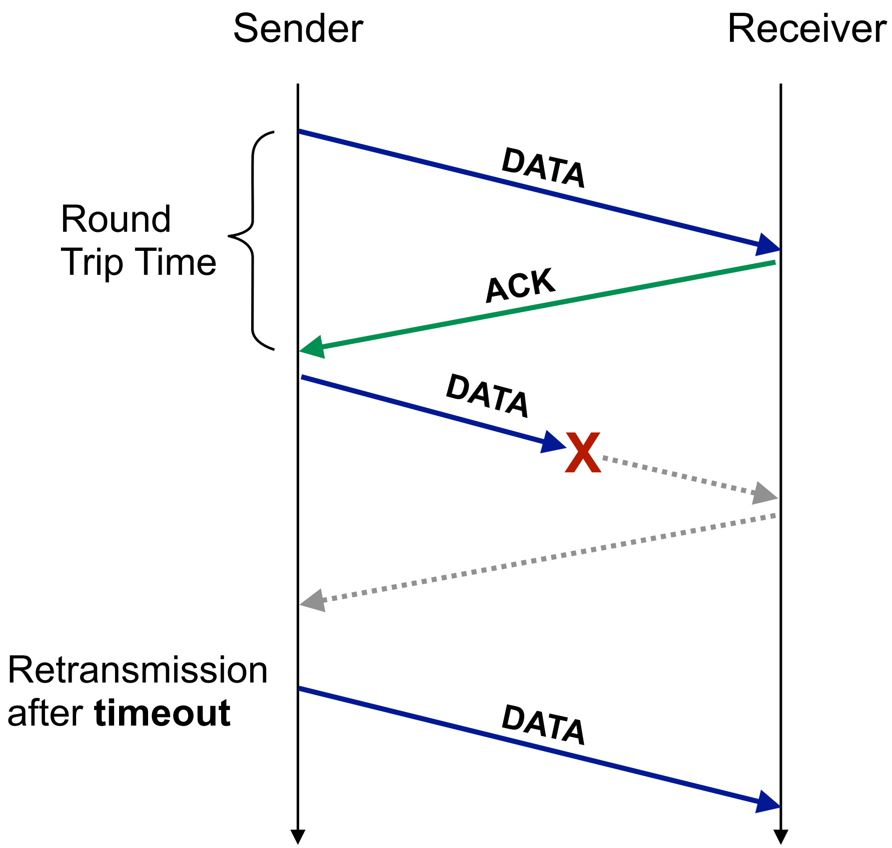
Retransmissions

- Retransmissions are triggered, if acknowledgements do not arrive
  ... but how to decide that?

- Measurement of the **round trip time (RTT)**



Retransmissions and RTT



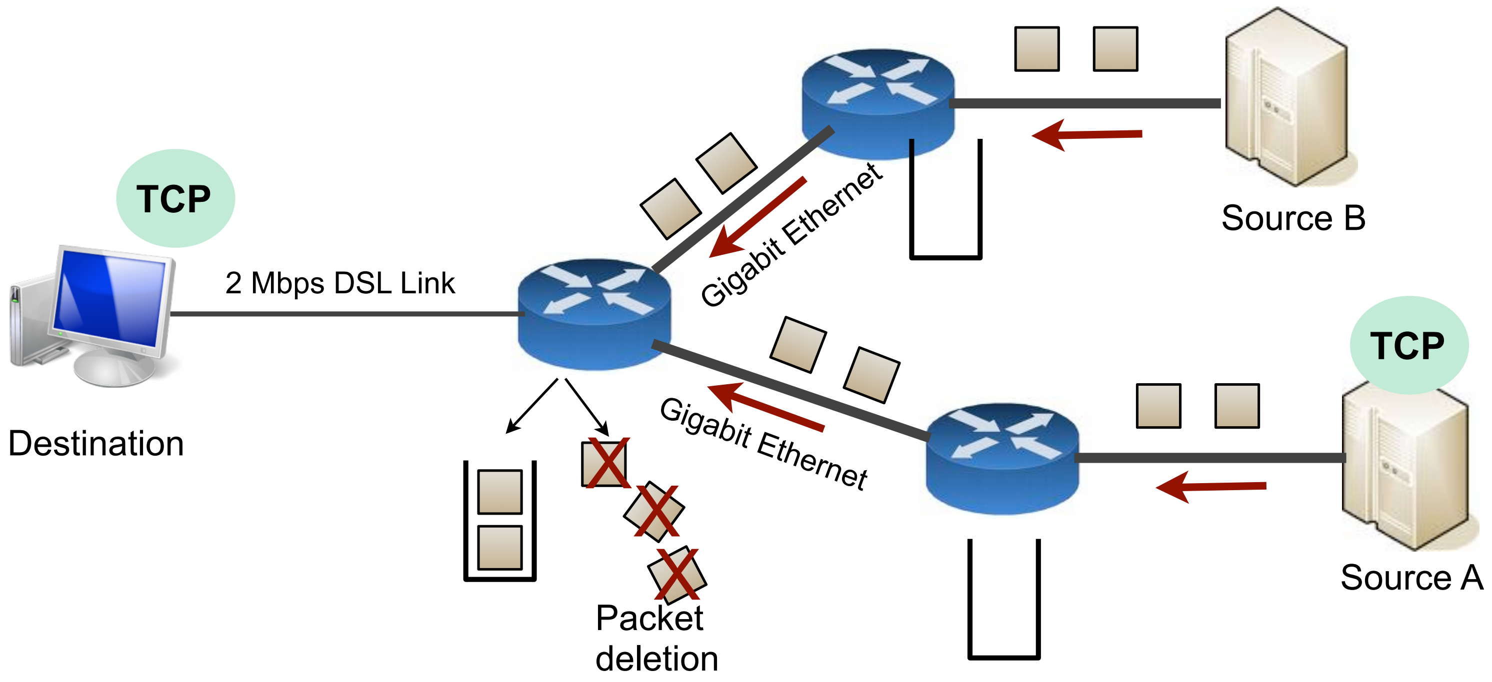
Estimation of the Round Trip Time (RTT)

- If no acknowledgement arrives before expiry of the **Retransmission Timeout (RTO)**, the packet will be retransmitted
  - RTT not predictable, fluctuating

- **RTO derived from RTT estimation:**
  - RFC 793: (M := last RTT measurement)
    - RTT $\leftarrow \alpha$ RTT + (1-α) M, where α = 0,9
    - RTO $\leftarrow \beta$ RTT, where β = 2
  - Alternative by Jacobson 88 (using the deviation D):
    - D $\leftarrow \alpha'$ D + (1-α') |RTT - M|
    - RTT $\leftarrow \alpha$ RTT + (1-α) M
    - RTO $\leftarrow$ RTT + 4D

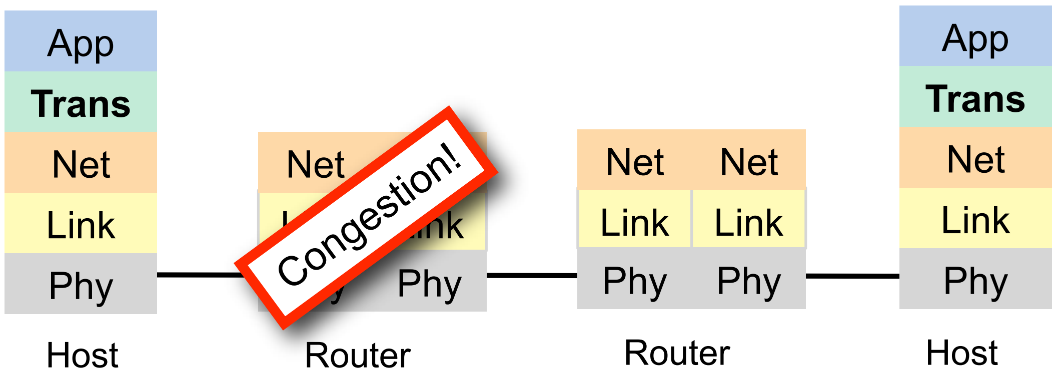
Congestion revisited

- **IP Routers drop packets**
- **TCP has to react, e.g. lower the packet injection rate**

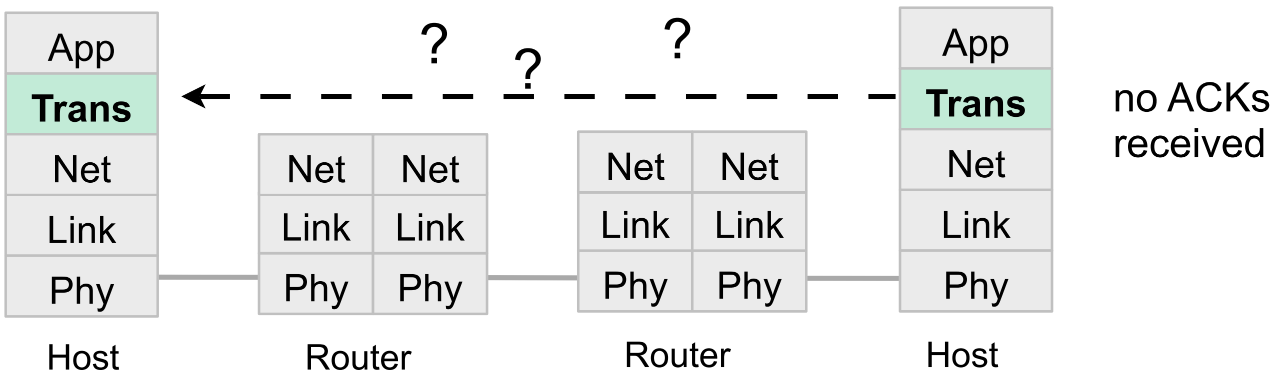


Rechnernetze und Telematik
Albert-Ludwigs-Universität Freiburg
Christian Schindelhauer

Congestion revisited

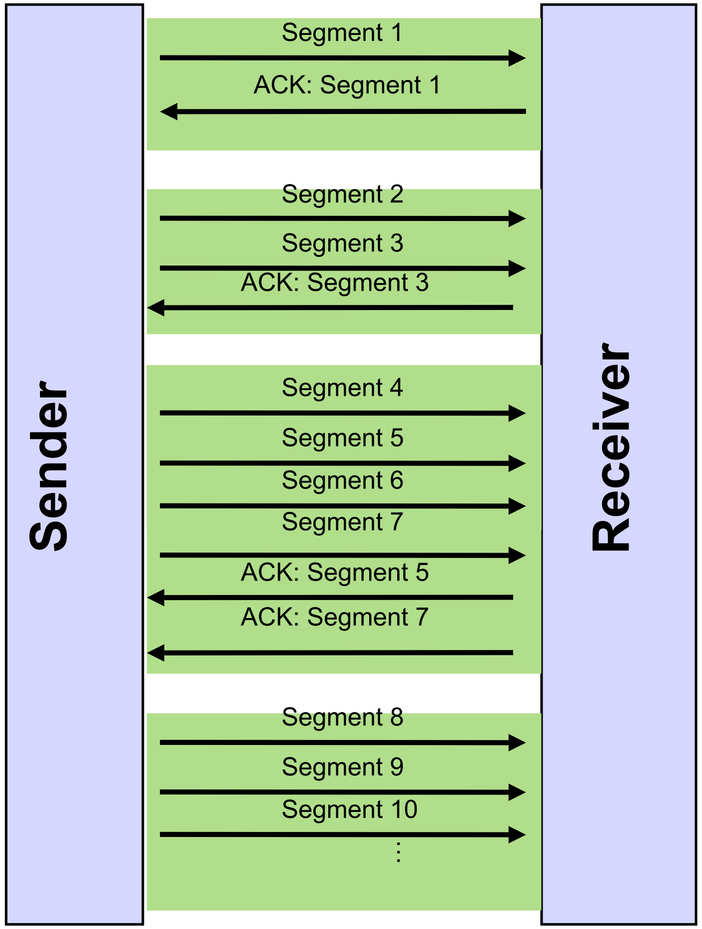


from a transport layer perspective:

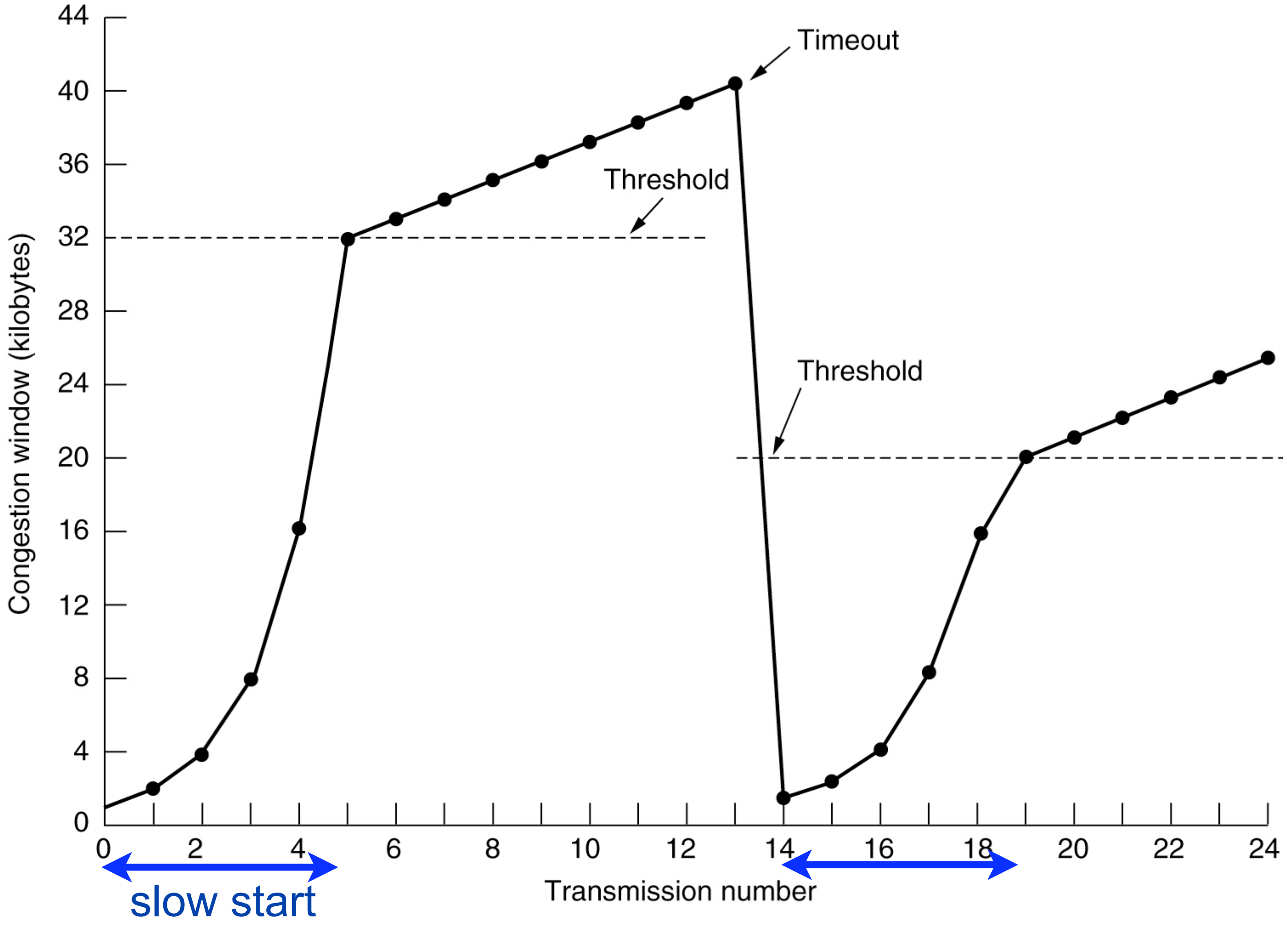


Data rate adaption and the congestion window

- **Sender does not use the maximum segment size in the beginning**
- **Congestion window (cwnd)**
  - used on the sender size
  - sending window: min {wnd,cwnd} (wnd = receiver window)
  - S: segment size
  - Initialization:
    - cwnd ← S
  - For each received acknowledgement:
    - cwnd ← cwnd + S
  - ...until a packet remains unacknowledged



Slow Start of TCP Tahoe



TCP Tahoe's slow start

- **TCP Tahoe, Jacobson 88:**
  - Congestion window (cwnd)
  - Slow Start Threshold (ssthresh)
  - S = maximum segment size
- **Initialization (after connection establishment):**
  - cwnd ← S ssthresh ← 65535
- **If a packet is lost (no acknowledgement within RTO):**
  - multiplicative decrease of ssthresh
    cwnd ← S ssthresh ← $\max\left\{2 \times S, \frac{\min\{cwnd, wnd\}}{2}\right\}$
- **If a segment is acknowledged and cwnd ≤ ssthresh then**
  - slow start: cwnd ← cwnd + S
- **If a segment is acknowledged and cwnd > ssthresh, then**

  cwnd ← cwnd + S/cwnd

x: # Packets per RTT

x ← 1	y ← max
x ← 1	y ← x/2
x ← 2·x, until x = y	
	x ← x +1

Fast Retransmit and Fast Recovery

- **TCP Tahoe [Jacobson 1988]:**
  - If only one packet is lost
    - retransmit and use the rest of the window
    - Slow Start
  - Fast Retransmit
    - after three duplicate ACKs, retransmit Packet, start with Slow Start

- **TCP Reno [Stevens 1994]**
  - After Fast Retransmit:
    - ssthresh ← min(wnd,cwnd)/2
    - cwnd ← ssthresh + 3 S
  - Fast recovery after Fast retransmit
    - Increase window size by each single acknowledgement
    - cwnd ← cwnd + S
  - Congestion avoidance: if P+x is acknowledged:
    - cwnd ← ssthresh

y ← x/2
x ← y + 3

The AIMD principle

- **TCP uses basically the following mechanism to adapt the data rate x (#packets sent per RTT):**
  - Initialization:

    $$x \leftarrow 1$$

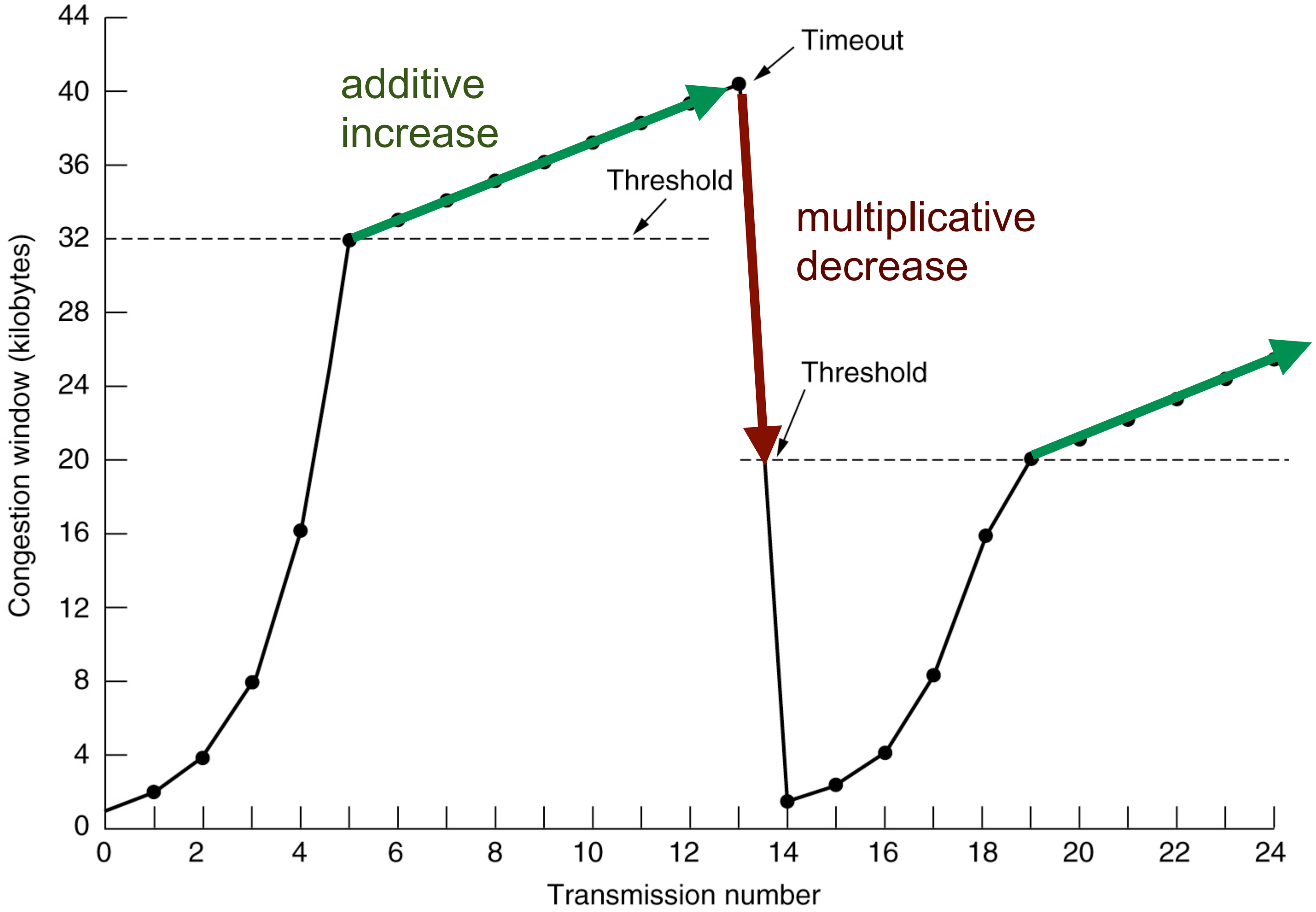
  - on packet loss: multiplicative decrease (MD)

    $$x \leftarrow x/2$$

  - if the acknowledgement for a segment arrives, perform additive increase (AI)

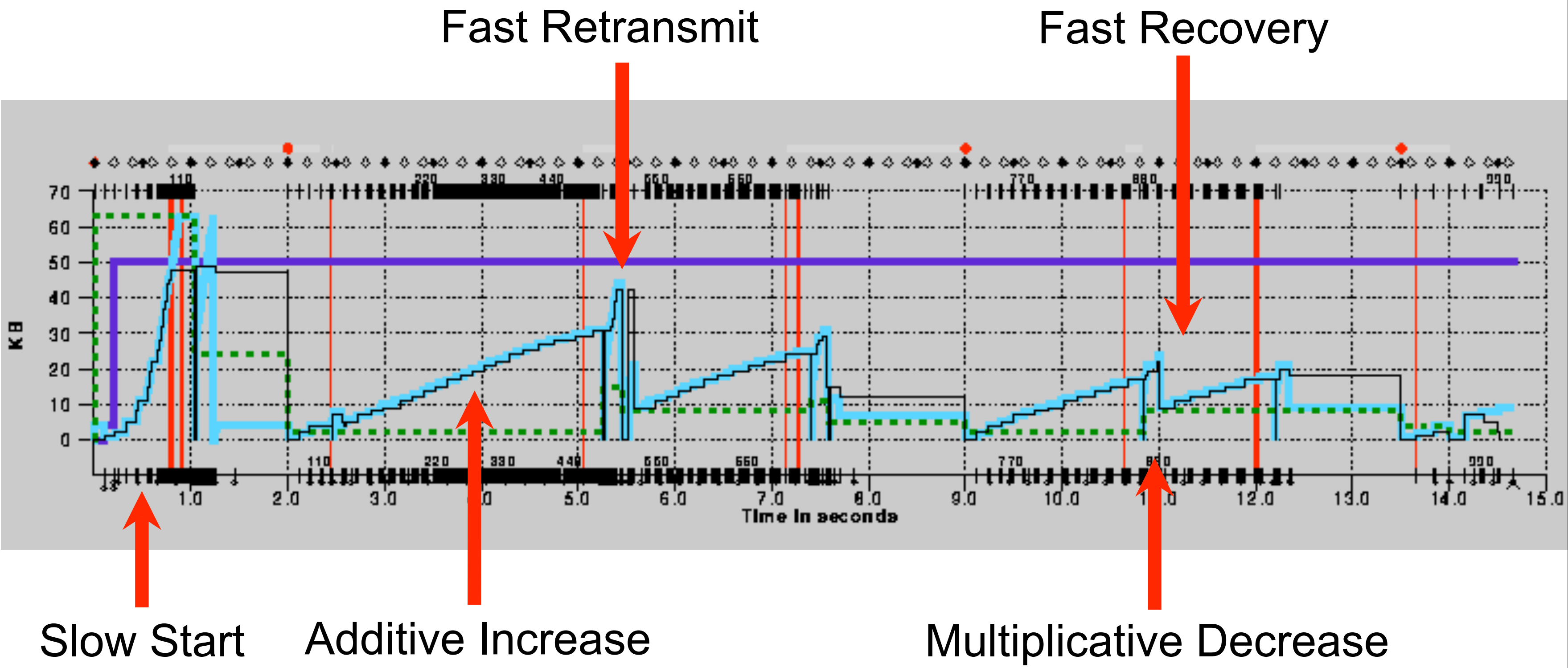
    $$x \leftarrow x + 1$$

AIMD



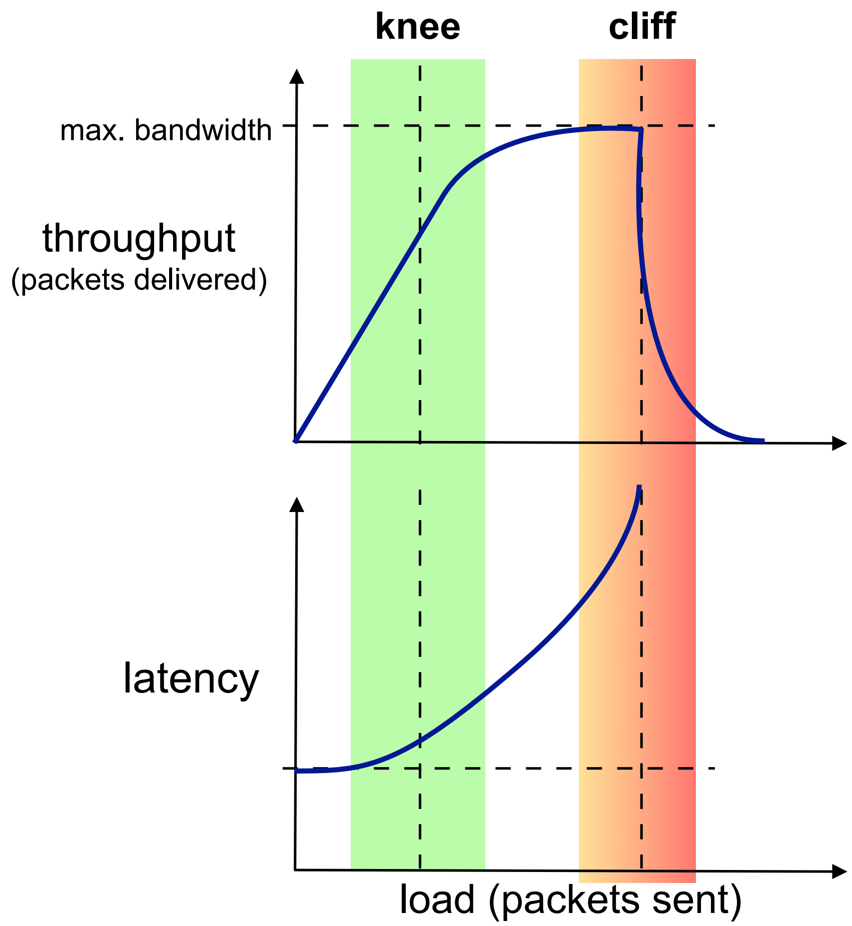
44
40
36
32
28
24
20
16
12
8
4
0
Congestion window (kilobytes)
0
2
4
6
8
10
12
14
16
18
20
22
24
Transmission number
additive
increase
Timeout
Threshold
multiplicative
decrease
Threshold

Example of TCP Reno



Throughput and Latency

- **Congested situation (cliff):**
  - high load
  - low throughput
  - all data packets are lost
- **Desired situation (knee):**
  - high load
  - high throughput
  - few data packets get lost



Simple data rate model

- **n participants, based on rounds**
  - participant i has date rate $x_i(t)$
  - initial data rate $x_1(0), \ldots, x_n(0)$
- **Feedback after round t:**
  - $y(t) = 0$, if $\sum_{i=1}^{n} x_i(t) \leq K$
  - $y(t) = 1$, if $\sum_{i=1}^{n} x_i(t) > K$
  - where K is the critical load ("knee")
- **Each participant adapts the data rate in round t+1:**
  - $x_i(t+1) = f(x_i(t), y(t))$
  - Increase strategy $f_0(x) = f(x,0)$
  - Decrease strategy $f_1(x) = f(x,1)$
- **we consider the following linear functions:**

$$f_0(x) = a_I + b_I x \quad \text{and} \quad f_1(x) = a_D + b_D x\ .$$

Variants

- **AIAD**: Additive Increase Additive Decrease

$$f_0(x) = a_I + x \quad \text{and} \quad f_1(x) = a_D + x \ ,$$

where $a_I > 0$ and $a_D < 0$.

- **MIMD**: Multiplicative Increase/Multiplicative Decrease

$$f_0(x) = b_I x \quad \text{and} \quad f_1(x) = b_D x \ ,$$

where $b_I > 1$ and $b_D < 1$.

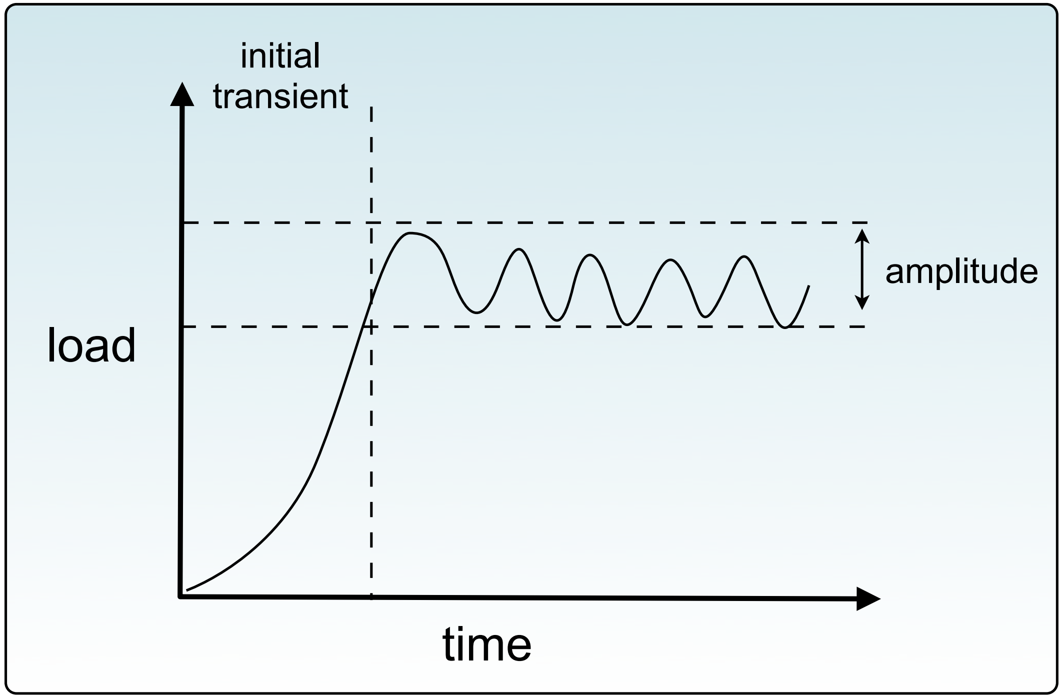
- **AIMD**: Additive Increase Multiplicative Decrease

$$f_0(x) = a_I + x \quad \text{and} \quad f_1(x) = b_D x \ ,$$

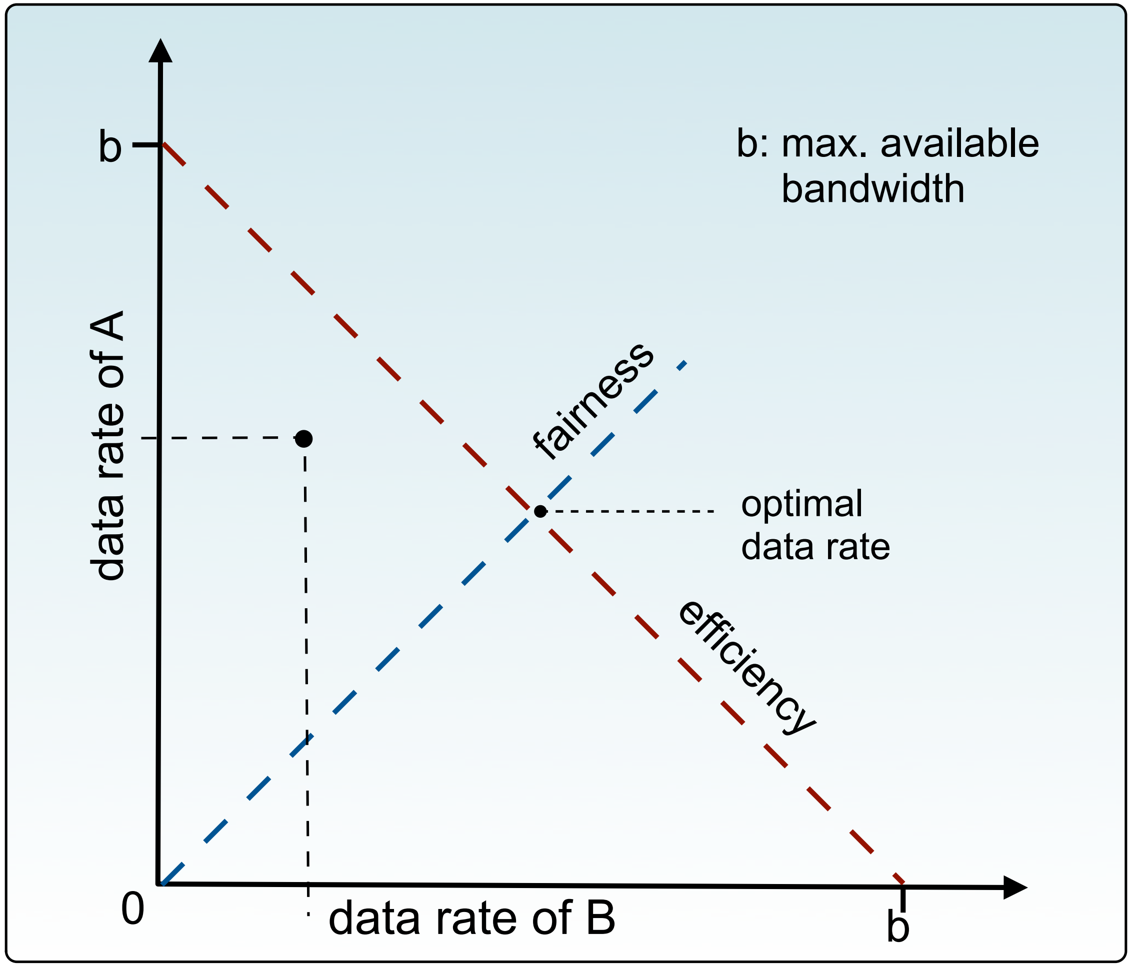
where $a_I > 0$ and $b_D < 1$.

Convergence

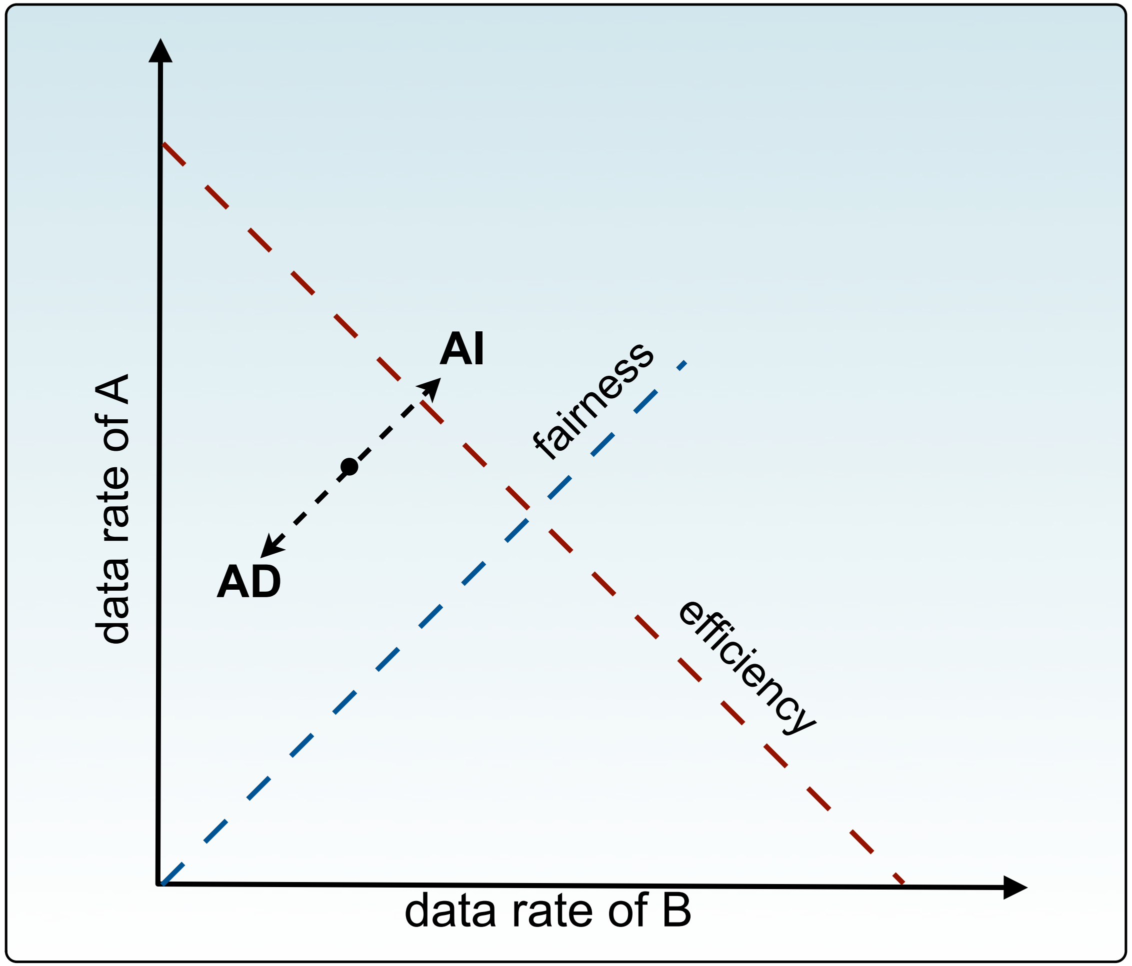
- **Convergence impossible**
- **best case: oscillation around an optimal value**



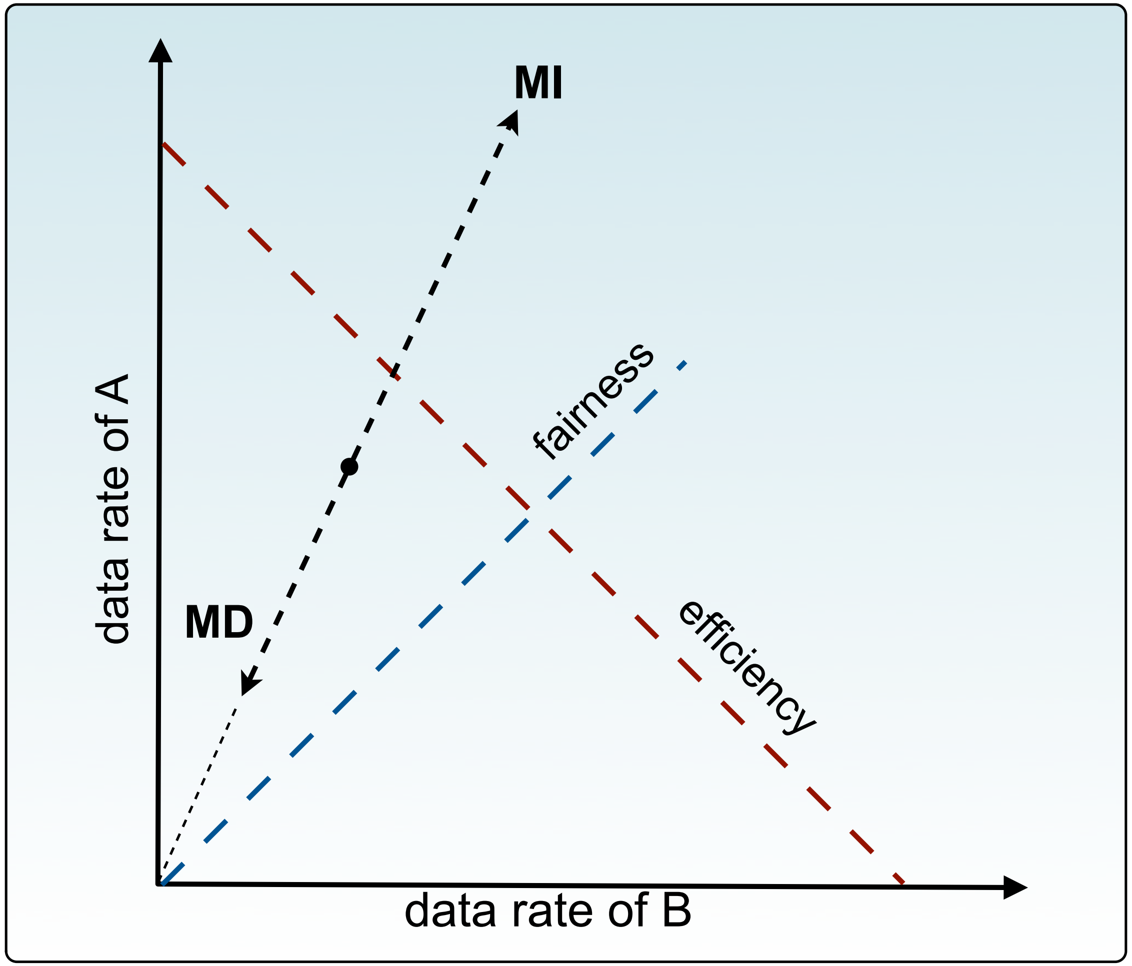
Vector diagram for 2 participants



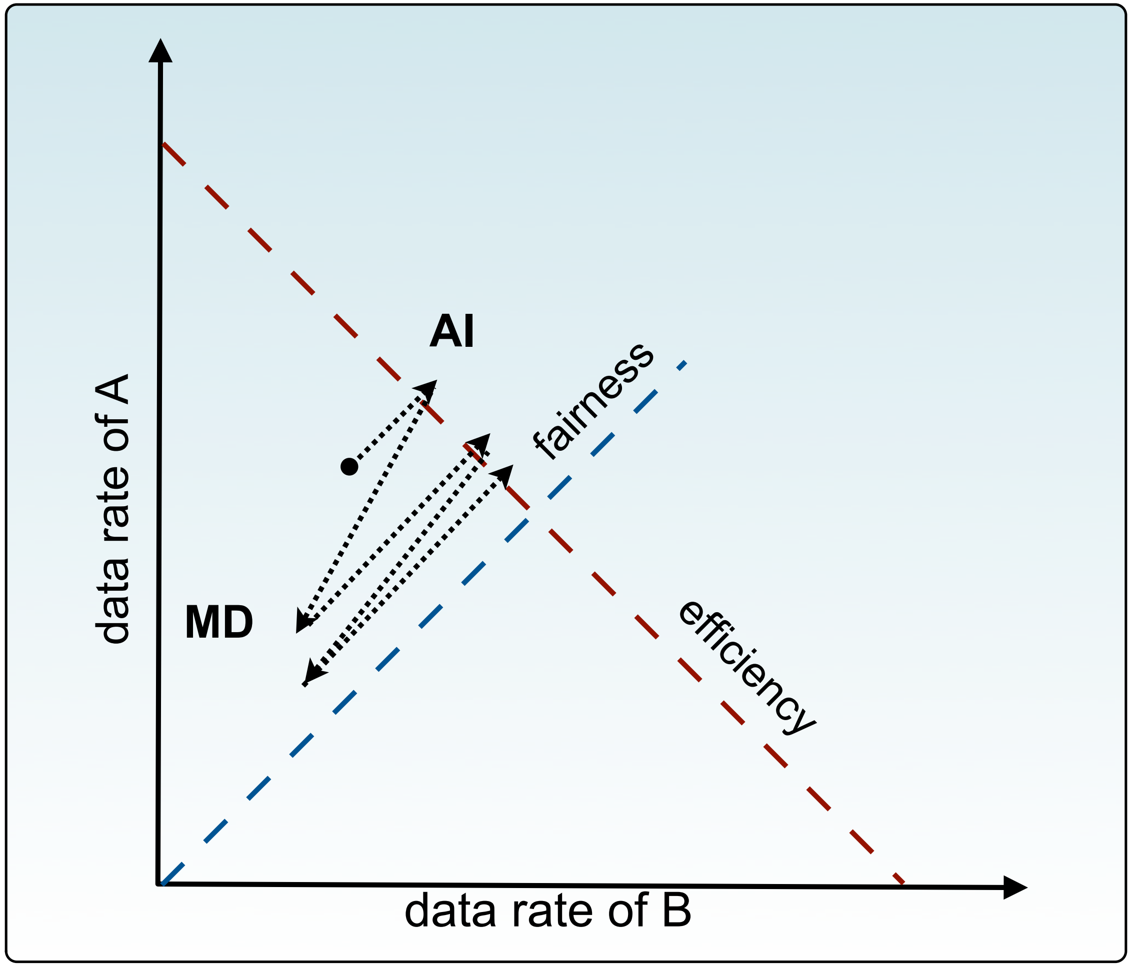
AIAD Additive Increase/ Additive Decrease



MIMD: Multiplicative Incr./ Multiplicative Decrease

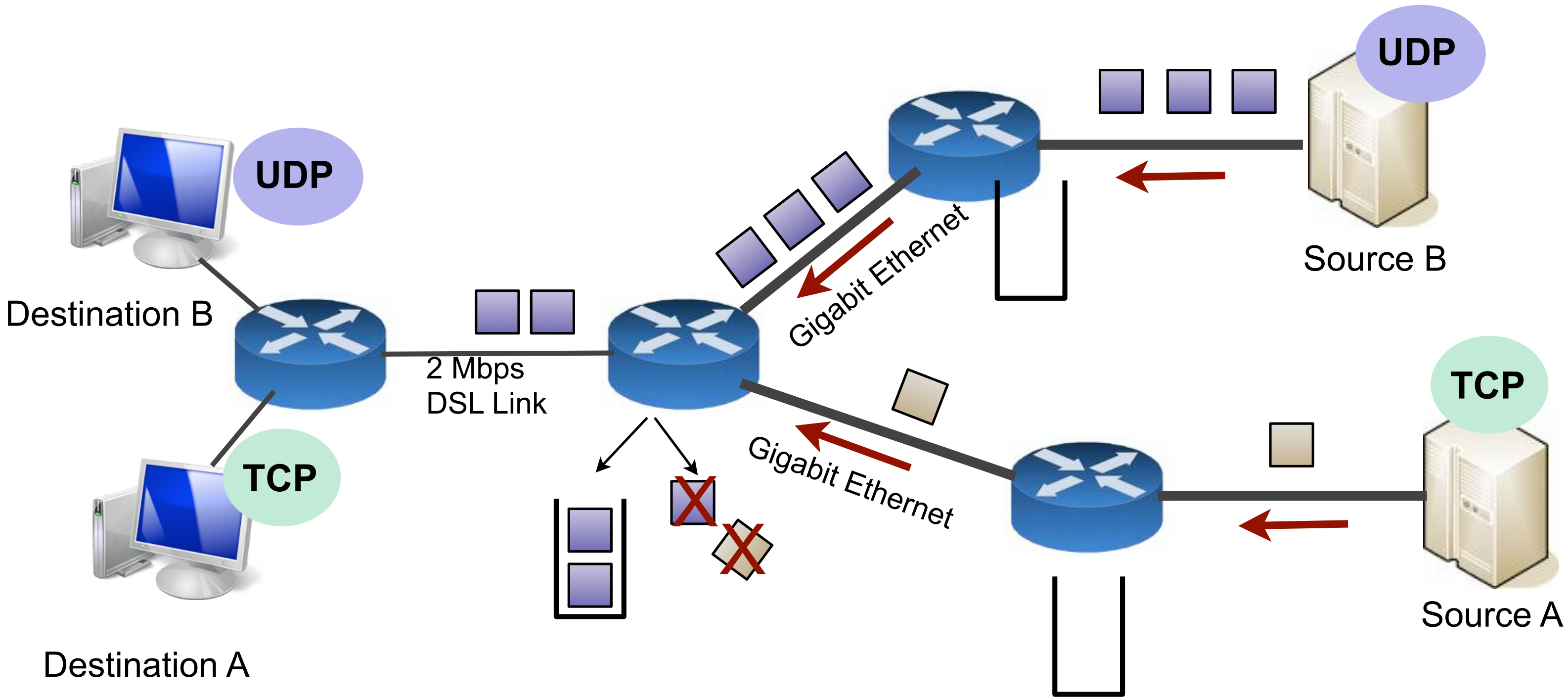


AIMD: Additively Increase/ Multiplicatively Decrease



TCP vs. UDP

- TCP reduces data rate
- UDP does not!



TCP - Conclusion

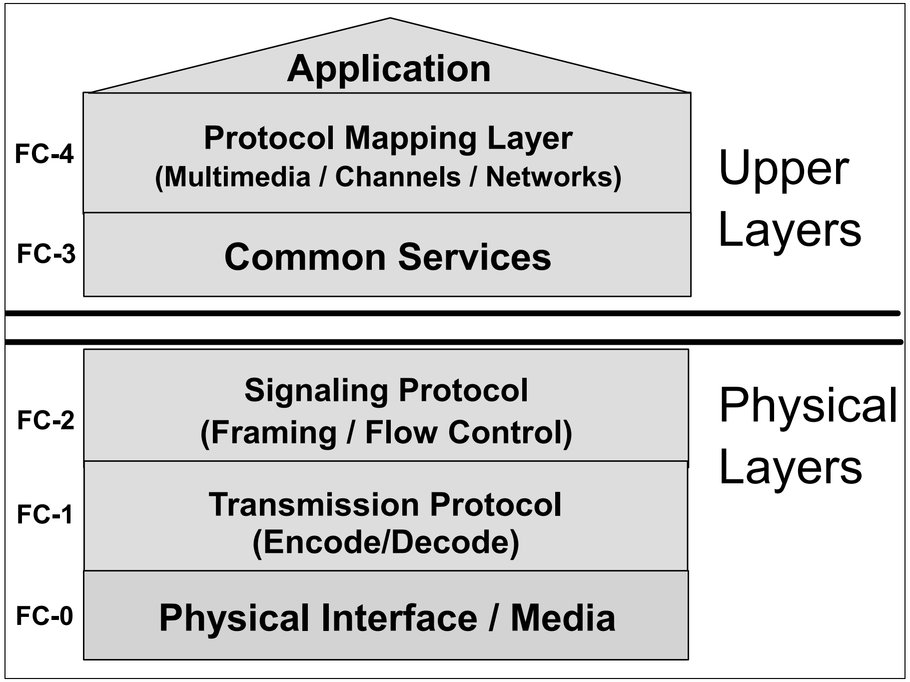
- Connection-oriented, reliable, in-order delivery of a byte stream
- Flow control and congestion control
  - Fairness among TCP streams
  - Unfair behavior of other protocols, e.g. UDP
  - Impact on latency
  - Tweaking the congestion avoidance mechanism has an impact on other applications

Storage networking

- **Fibre Channel**
  - standard connection for SANs
  - Medium: fibre-optic but also twisted pair
  - Protocol: channel-like transport of SCSI commands
  - Topologies: From point-to-point to networks
  - Advantages: flexible connectivity, networking capabilities

Fibre Channel Protocol (FCP)

- Transport protocol for SCSI commands
- Layered architecture

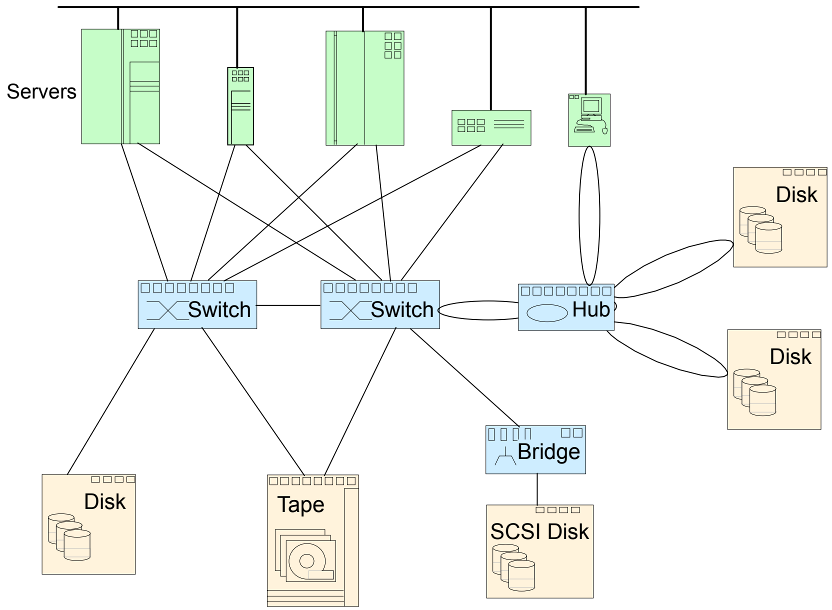


FCP Layers

FC4	**Protocol Mapping Layer**	encapsulation of other protocols
FC3	**Common Services**	encryption, striping, RAID, etc.
FC2	**Framing and Signalling**	data transport, routing
FC1	**Transmission Protocol**	8b/10b encoding and decoding
FC0	**Physical Layer**	medium

Fibre Channel Topologies

- **Point-to-Point**
  - connection of 2 nodes
- **Arbitrated Loop (FC-AL)**
  - shared bus of up to 126 nodes
- **Switched Fabric (FC-SW)**
  - interconnection network
  - routing and transport protocols



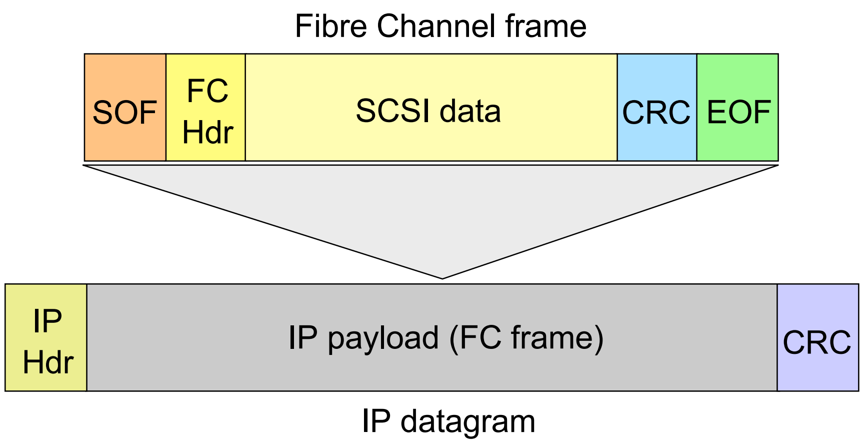
switched fabric

Network Storage Types

- **Direct attached storage (DAS)**
  - traditional storage
- **Network attached storage (NAS)**
  - storage attached to another computer accessible at file level over LAN or WAN
- **Storage area network (SAN)**
  - specialized network providing other computers with storage capacity with access on block-addressing level

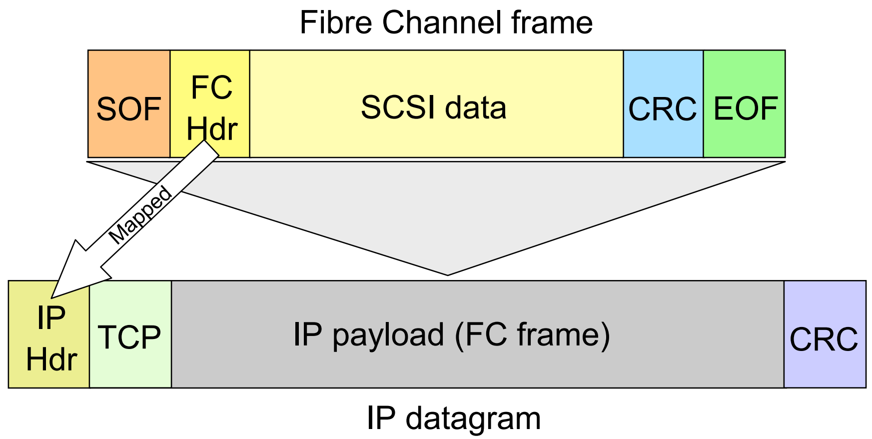
IP storage networking protocols

- **Fibre Channel over IP (FCIP)**
  - Tunneling data between SAN devices through IP networks
  - based on TCP connections
  - links SAN devices and switch fabrics over IP networks
  - Merging switch fabrics over IP links problematic
    (frequent switch reconfigurations because of link unreliability)



IP storage networking protocols

- **Internet Fibre Channel Protocol (iFCP)**
  - Fibre Channel switch fabric services over IP networks
  - based on TCP connections
  - uses IP routing and switching
  - can replace the Fibre Channel switch fabric



ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG

Algorithms and Methods for Distributed Storage

6 Networking

Stefan Rührup

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Wintersemester 2008/09

CoNe Freiburg

IIF INSTITUT FÜR INFORMATIK FREIBURG