

Sc2008

Week 1 M1-L2 network layers & physical resilience

Layered Network Architecture

Motivations for Layered Network Architecture

- Networks are complex with many pieces**

- Hosts, routers, links, applications, protocols, hardware, software

- Dealing with complex systems**

- Explicit structure allowing identification, relationship of different pieces
 - Layered reference model for discussion
- **Modularization** easing maintenance and updating
 - Change of layer's service transparent to rest of system
 - Change in network technology does not affect rest of system
- **Layering** (design vs implementation)

Layered Network Architecture

- Network organized as a stack of layers**

- Purpose of layer is to offer services to the layer above it and passes data & control information to the layer below, using a well-defined interface
- Reducing design complexity

- Protocols:** a set of rules governing communication between two peering parties/computers

- define format, order of messages sent and received among network entities, and actions taken on message transmission & receipt.

- Network Architecture:** a set of layers and protocols with specifications enabling hardware/software developers to build systems compliant with a particular architecture

Benefits of Layers

- Simplicity**

- Easy to design once layers and their interactions are defined clearly

- Flexibility**

- Easy to modify and develop networks by separate layers modifications

- Incremental Changes**

- Easy to add new layers, add new functions to a layer

OSI 7-Layer Model

- Function Decomposition**

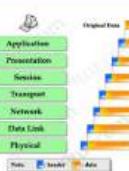
- Weakly-decoupled stack

- Encapsulation**

- Each layer adding new headers

- Peering**

- Only peer layer "communicating" with each other



OSI Reference Model: 7 Layers (More on Supplementary Materials)

OSI Model			
	Data unit	Layer	Function
Host layers	Data	7. Application	Network process to application
		6. Presentation	Data representation, encryption and decryption, convert machine dependent data to machine independent data
	Segment	5. Session	Inter-host communication, managing sessions between applications
		4. Transport	End-to-end connections, reliability and flow control
Media layers	Packet	3. Network	Path determination and logical addressing
		2. Data link	Physical addressing
	Bit	1. Physical	Media, signal and binary transmission

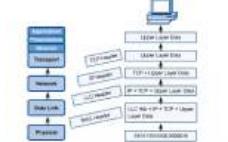
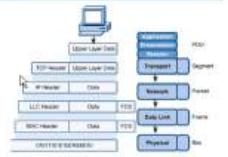
OSI in Action: Encapsulation

- A message begins at the top application layer and moves down the OSI layers to the bottom physical layer

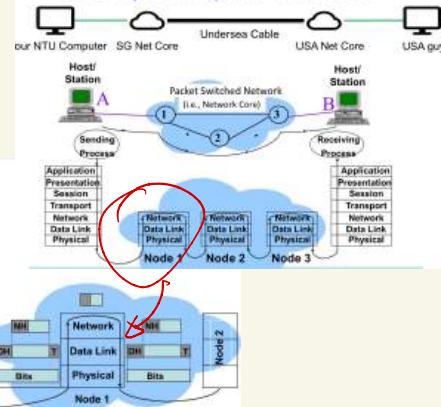
- As the message descends, each successive OSI model layers adds a header to it

- A header is **layer-specific** information that basically explains what functions the layer carries out

- Conversely, at the receiving end, headers are stripped from the message as it travels up the OSI layers.

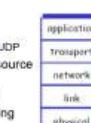


A Simple Computer Network



TCP/IP Model: 5 Layers

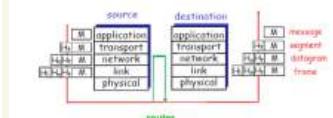
- Application:** supporting network applications
 - FTP, SMTP, HTTP
- Transport:** host-host data transfer
 - Transmission Control Protocol [TCP], UDP
- Network:** routing of datagrams from source to destination
 - Internal Protocol [IP], routing protocols
- Link:** data transfer between neighboring network elements
 - PPP, Ethernet
- Physical:** bits on the wire



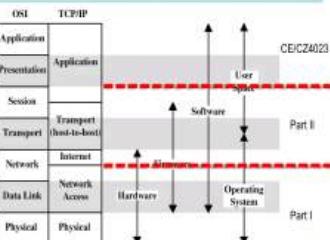
TCP/IP Internetworking

- Each layer takes data from above

- Adds header information to create new data unit
- Passes new data unit to layer below



TCP/IP vs OSI Models



Week 2

use links for abstraction

Link Failure Probability

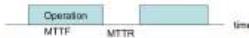
Network Reliability

- Probability that a network performs satisfactorily over a period of time

Parameters:

- Mean Time Between Failures (MTBF)
- Mean Time to Failure (MTTF)
- Mean Time to Repair (MTTR)

$$MTBF = MTTF + MTTR$$



fail after
in repair

- Link Failure Probability:** percentage of time during which the link is dysfunctional

- Link Availability:** percentage of time during which the link is functional

- b_i : Probability link "i" is broken

- r_i : Probability link "i" is available, i.e., not broken

$$r_i = 1 - b_i$$

Availability %	Downtime per year	Downtime per month*	Downtime per week
90% ("one nine")	36.5 days	72 hours	16.8 hours
95% 95%	18.25 days	36 hours	8.4 hours
97% 97%	10.50 days	21.6 hours	5.04 hours
98% 98%	7.30 days	14.4 hours	3.36 hours
99% 99%	3.65 days	7.20 hours	1.68 hours
99.5% 99.5%	1.83 days	3.60 hours	50.4 minutes
99.9% 99.9%	17.52 hours	56.25 minutes	20.16 minutes
99.99% 99.99%	8.70 hours	43.8 minutes	10.1 minutes
99.999% 99.999%	4.38 hours	21.56 minutes	5.04 minutes
99.9999% 99.9999%	0.259 minutes	4.32 minutes	1.01 minutes
99.99999% 99.99999%	31.5 seconds	25.9 seconds	6.05 seconds
99.999999% 99.999999%	3.15 seconds	2.59 seconds	0.605 seconds

Definitely in Exam Network Resilience Issues

- What's the probability of a link failure?
- Are there alternative paths?
- Is there a single point of failure?
- What is the probability for two nodes to stay connected in a network?



Failure probability b_{X-Y}

Availability probability $r_{X-Y} = 1 - b_{X-Y}$

Network Availability : Series

Parallel

$$r_{X-Y-Z} = \Pr[\text{both links survive}] = r_{X-Y} \cdot r_{Y-Z}$$

$$b_{X-Y-Z} = \Pr[\text{both links break}] = b_{X-Y} \cdot b_{Y-Z}$$

Network Resilience

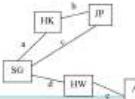
- A measure of Network Fault Tolerance

- Express in terms of probability that the network remains connected.

Assumptions

- The probability of link failures are independent of each other.

Common exam question
(Probability)
(Resilience)



Rules for Network Availability

Link in series

- Calculate that probability that all links in the series are working

Link in parallel

- Calculate the probability that all links are broken.

Combination of series and parallel

- Decompose them into paths
- Calculate network availability using path-based approach

Network Availability: Series



Given that each link has a failure probability of 0.05, Prob that SG can communicate with AU:

$$r_{SG-HW-AU} = \Pr[\text{both links survive}] = r_{SG-HW} \cdot r_{HW-AU} = (1-0.05) \cdot (1-0.05) = 0.9025$$

What is the probability that SG cannot communicate with AU?

$$b_{SG-HW-AU} = 1 - 0.9025 = 0.0975$$

Computing $1 - r_{SG-HW} \cdot r_{HW-AU}$ above is easier than summing the three products below:

$$\begin{aligned} & b_{SG-HW} \cdot b_{HW-AU} \quad b_{SG-HW} \cdot r_{HW-AU} \quad r_{SG-HW} \cdot b_{HW-AU} \\ & \boxed{\text{SG} \text{---} \text{HW} \text{---} \text{AU}} \quad \boxed{\text{SG} \text{---} \text{HW} \text{---} \text{AU}} \quad \boxed{\text{SG} \text{---} \text{HW} \text{---} \text{AU}} \end{aligned}$$

$$\begin{aligned} & 0.05 \cdot 0.05 + 0.05 \cdot 0.95 + 0.95 \cdot 0.05 \\ & 0.0025 + 0.0475 + 0.0475 \\ & = 0.0975 \end{aligned}$$

Network Availability: Parallel

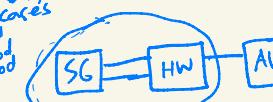


Given that each link has a failure probability of 0.05, What is the probability that SG is isolated from AU?

$$\begin{aligned} \Pr[\text{break}] &= \Pr[\text{both links break}] \\ &= b_{\text{Link B}} \cdot b_{\text{Link B}} \\ &= 0.05 \cdot 0.05 = 0.0025 \end{aligned}$$

$$b_{SG-AU} = 0.0025$$

If by availability
then node 3 cores
- a good & bad
- a bad & good
- a good & good



$$b_{SG-HW} = 0.05 \cdot 0.05 = 0.0025$$

$$\begin{aligned} r_{SG-HW-AU} &= 0.0075 \cdot (1-0.05) = 0.007125 \\ b_{SG-HW-AU} &= 1 - r_{SG-HW-AU} = 0.992875 \end{aligned}$$

Hybrid Graphs: Path-Based Approach



Given that each link has a failure probability of 0.05, Calculate the Prob that SG is isolated from AU.

$$\begin{aligned} r_{SG-HW-AU} &= r_{SG-HW} \cdot r_{HW-AU} \\ &= (1-0.05) \cdot (1-0.05) \\ &= 0.9025 \\ &= 0.0975 \end{aligned}$$

$$\begin{aligned} \text{Prob. SG Disconnected from AU} &= b_{SG-HW-AU} \cdot b_{SG-AU} \\ &= 0.0975 \cdot 0.05 \\ &= 0.004875 \end{aligned}$$

$$b = 0.05$$

$$\begin{aligned} b_{SG-AU} &= b \cdot a \cdot b \\ &= (1-0.05)(1-0.05)(0.05) \\ &= (1-0.9025)(0.05) \\ &= 0.004875 \end{aligned}$$

$$\begin{aligned} r_{SG-HW} &= 1 - 0.0025 \\ &= 0.9975 \end{aligned}$$

$$\begin{aligned} r_{SG-HW-AU} &= 0.9975 \cdot (1-0.05) \\ &= 0.992875 \end{aligned}$$

ml-L3 Data link layer (DLL) flow control

X Data Link Layer (DLL): Roles

DLL Services

- Framing:** encapsulate each network-layer datagram within a link-layer frame before transmission over the link
- Link Access:** MAC protocol specifying the rules by which a frame is transmitted onto the link
- Flow Control:** control of data flow to ensure sender not overwhelm the receiver with data
- Reliable Delivery:** move each network-layer datagram across the link without error

Functions and Mechanisms

Flow control

- Ensuring that a transmitting station does not overwhelm a receiving station with data, i.e., buffers at the receiver do not get overflowed.

No frame error



Two Flow-Control Mechanisms

- Stop-and-Wait
- Sliding Window

Stop-and-Wait Flow Control

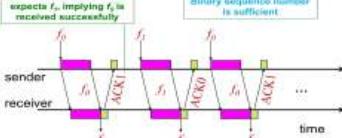


Operations:

- A packages data into a frame
- A sends the frame to B
- B waits for an ACK
- When B has received the frame, B sends an ACK
- When A has received the ACK, A resends

ACK1 means the receiver expects t_1 , implying t_1 is received successfully

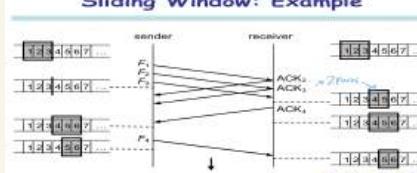
Binary sequence number is sufficient



Sliding Window Flow Control

- Allows multiple frames to be in transit.
- Sender and Receiver have buffer N long.
- Sender can send up to N frames without receiving ACKs.
- Each frame is numbered.
- ACK includes number of next expected frame.
- Sequence number bounded by field size (k bits)
 - Frames are numbered modulo 2^k
 - Sequence number $[0, 2^k - 1]$

Sliding Window: Example



Window Size Consideration

Say, window size, $N = 3$
with $k=1$ bit sequence #

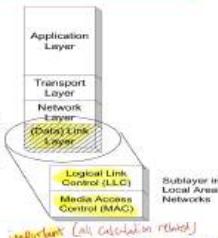
NSZ
Window Size N
bit's Sequence #

Sender: Do you ask for a retransmission of the transmitted frame or a new frame?

Receiver: Is it a retransmission or a new frame?

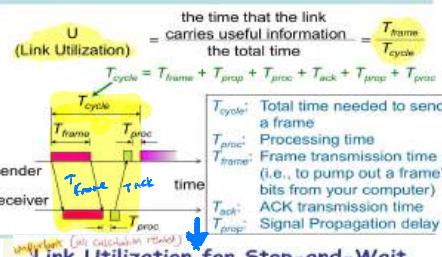
Is the second a new frame or the retransmitted frame?

Which frame is to be transmitted next after receiving ACK?



LLC performs [Call, Calculation related]

Flow-Control Link Utilization



Link Utilization for Stop-and-Wait

Assumptions

- Input is saturated
- No error
- Ignoring T_{ack} & T_{proc}

We get:

$$T_{cycle} = T_{frame} + 2T_{prop}$$

$$\text{Then: } U = T_{frame} / (T_{frame} + 2T_{prop}) = 1 / (1+2a)$$

where:
we define $a = T_{prop} / T_{frame}$

$$U = \frac{1}{1+2a}$$

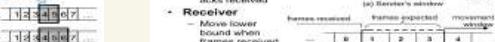
Parameter 'a' is called Normalized Propagation Delay

Sliding Window Operations

- Sender maintains a window, containing frame numbers that can be transmitted.
- Sender window shrinks from trailing edge (left side) as frames are sent.
- Receiver maintains a window as well, its window shrinks from trailing edge as frames are received.
- Receiver's window expands from the leading edge (right side) as ACKs are sent.
- Sender's window expands from the leading edge as ACKs are received.

Sliding Window Operations

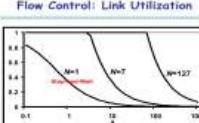
- Sender**
 - Move lower bound when frame received
 - Move upper bound when ACKs received
- Receiver**
 - Move lower bound when frame received
 - Move upper bound when ACKs sent



Sliding Window: Performance

- Performance depends upon (assume error-free operation):
 - Parameter c , and
 - Window size, N .
 - Assumption: T_{frame} and T_{prop} are negligible.
 - Frame transmission rate = 1 (normalized to itself)
 - Normalized propagation delay (one-way) = a
 - We need to consider two cases:
 - $N > 2a$: Sender can transmit continuously without exhausting its window $\Rightarrow U = 1/a$
 - $N < 2a$: Station's window is exhausted at $t = N/a$, and the station cannot send additional frame until $t = 2a + 1$. $\Rightarrow U = N/(1+2a)$

Flow Control: Link Utilization

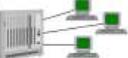


Link Configuration/Access

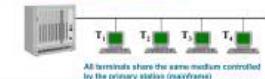
- Objective: determine who gets to transmit at when on a link

Topology

Point-to-point:



Point-to-Multipoint (Broadcast):



All terminals share the same medium controlled by the primary station (maximize)

Stop-and-Wait: Disadvantages

- If frame or ACK is lost, long waiting time is expected

To fix this, use a TIMEOUT control in the sender

- If the normalized propagation delay is long, the sender must wait a long time before it can perform the next transmission.

The link utilization $U = \frac{1}{1+2a}$ is low.

To fix this, use Buffers at the sender/receiver (sliding window operation). This will improve the numerator. Note that the denominator $1+2a$ cannot be improved.

Example

A communication link exists between two nodes A and B. The transmission rate on the link is 2.4 Mbps. The distance between A and B is 50 km and the signal velocity is 2×10^8 m/s. The frame length is 300 bytes. No frame error. Calculate the link utilization for the stop-and-wait flow control mechanism.

A 50km

B Here T_p and T_{prop} are short for T_{prop} and T_{frame} respectively.

$R = 2.4 \text{ Mbps}$, $L=300 \text{ bytes} = 2400 \text{ bits}$

$D=50\text{km}$, $v = 2 \times 10^8 \text{ m/s}$

$$U = 1/(1+2a) \rightarrow a = T_p/T_{frame} \rightarrow T_p = DV/2 = 5 \times 10^8 / 2 \times 10^6 = 250 \mu\text{s}$$

$$U = 1/(1+2(0.25)) \rightarrow a = 0.25 \rightarrow T_p = LR/2400 = 2.4 \times 10^6 / 2400 = 1000 \mu\text{s}$$

$$a = \frac{250}{1000} = 0.25$$

Learning Objectives

Data Link Layer Fundamentals

- To understand its four main functions.

Flow Control

- To understand its main purpose

Stop-and-Wait Flow-Control Mechanism

- Operational protocol
- Link utilization calculation

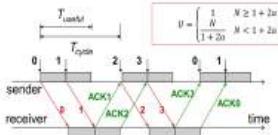
Sliding Window Flow-Control Mechanism

- Operational protocol
- Window size determination
- Link utilization calculation (two cases)

Sliding Window: Performance

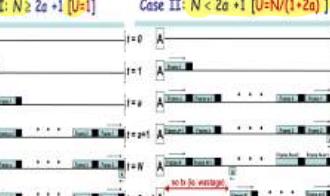
Window Size = N

$$T_{cycle} = N T_{frame} \\ T_{prop} = T_{frame} + 2 T_{prop}$$



Case I: $N > 2a + 1$ [$U=1$]

Case II: $N < 2a + 1$ [$U=N/(1+2a)$]



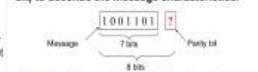
Week3: MI-LA Data Link Layer (DLL) Error Control

Error Control in Data Link Layer

- Objective**
 - To detect and correct errors that occur in frame transmission
- Frame Error in Data Link Layer (DLL)**
 - Lost Frame:** the receiver does not receive a frame (or the header was corrupted such that the frame was not recognized)
 - Damaged Frame:** the receiver receives a frame, but some of its bits are in error

X Error Detection: Parity Check

Parity Check (Odd/Even Parity): A single bit is appended to the original message (usually 7-bit) to describe the message characteristics.



Even Parity: The total number of 1s is even, i.e. 10011010. Odd Parity: The total number of 1s is odd, i.e. 10011011.

However, Parity Check can only detect odd numbers of errors!

No need to memorize

Channel Utilization: Formulas

Stop-and-Wait ARQ

$$U_{\text{Stop-and-Wait}} = \frac{1-P}{1+2aP}$$

$$N \geq 2a+1$$

$$N(1-P) \leq 2a+1$$

$$(1-P+NPK) > 2a$$

Go-back-N ARQ

$$U_{\text{Go-back-N}} = \frac{1-P}{1+2aP}$$

$$N \geq 2a+1$$

$$N(1-P) \leq 2a+1$$

$$(1-P+NPK) > 2a$$

$$N < 2a+1$$

$$(1-P+NPK) < 2a$$

$$N(1-P) < 2a$$

$$(1-P+NPK) < 2a$$

$$N < 2a+1$$

Week 4 MI-LS & MI-L MAC check for those not examinable (a lot)

Media Access Control (MAC)

- Assembly of data into frame with address and error detection fields
- Disassembly of frame
 - Address recognition
 - Error detection
- Govern access to transmission medium
 - Not found in traditional layer 2 data link control
- For the same LLC, several MAC options may be available

MAC Decision Making Options

- Where?
 - Central
 - Greeter control
 - Simple access logic at station
 - Avoids problems of co-ordination
 - Single point of failure
 - Potential bottleneck
 - Distributed
- How?
 - Synchronous (static) solutions
 - Specific capacity dedicated to connection
 - Asynchronous (dynamic) solutions
 - In response to demand

Single shared broadcast channel

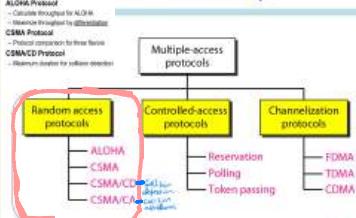
- Two or more simultaneous transmissions
 - Collision if node receives two or more signals at the same time
- MAC Protocol
 - Distributed algorithm to share the channel
 - Communication about channel sharing must use channel itself
 - No out-of-band channel for coordination
 - Shared

Ideal MAC Protocol

- Broadcast Channel of Rate R -bps
 - When one node transmits, it can send at rate R
 - When M nodes want to transmit, each can send at average rate R/M
- Full decentralized
 - No special node to coordinate transmissions
 - No synchronization of clocks, slots
- Simple
 - We call this ideal protocol as "genie-aided" MAC

Learning Objectives

MAC Taxonomy



Random Access Protocols

- When node has packet to send
 - Transmits at full channel data rate of R
 - No a-priori coordination among nodes
- Two or more transmitting nodes
 - Collision
- Design of random MAC has 3 aspects
 - Whether to sense channel status
 - How to transmit frames
 - How to detect and react to collision (What to do wif the collision)

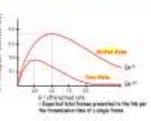
ALOHA

Assumptions

- All frames of the same size
- Time is divided into equal size slots
- Nodes are omnidirectional
- Nodes start to transmit frames only at beginning of slots
- If 2 or more nodes transmit in slot, all nodes detect collision

ALOHA Efficiency Comparison

Efficiency vs. Number of Nodes



Slotted ALOHA



Pure ALOHA

• In pure ALOHA, frames are transmitted at completely arbitrary times



Unslotted ALOHA: simpler, no synchronization

• When a frame first arrives

- Transmit immediately
- Collision probability increases

• If a node transmits, the probability that the next node transmits is α

• The probability that a slot is empty is $(1-\alpha)^n$

• An example of 6-node network

4 nodes on a single bus

• Offered load O is $\alpha \cdot N$

• Expected total number of transmitters in a slot

• Slotted ALOHA efficiency when N is large

$$\lim_{N \rightarrow \infty} P(\text{col}) = 1 - e^{-\lambda} = 1 - e^{-\alpha \cdot N}$$

$$= 1 - e^{-\alpha \cdot N} = 1 - e^{-\alpha \cdot \frac{O}{\alpha}} = 1 - e^{-O}$$

$$= O(1 - e^{-O})$$

$$= O(O - e^{-O})$$

$$= O(O^2 e^{-O})$$

<math display="block

NCS MI-L7 NY Lec 7 wired LAN: Ethernet

Learning Objectives

- Ethernet Overview**
 - Fixed frame size
 - IEEE standard MAC layer
- Ethernet MAC Previews**
 - Calculate collision rate when 80% utilization
 - Calculate collisions per second
 - Ethernet Round-trip
- Ethernet Round-trip**
 - Calculate round-trip time when 80% utilization
 - PEYOMAT

Baseband Manchester Encoding



Ethernet Frame Format

	Preamble	SFCS	DA	SA	U/L	Data / Padding	FCS
Total	10 bits	64 bits	46 bits	46 bits	1 bit	1500 bytes	4 bytes

Notes: SFCS = Sync Function Checksum; DA = Destination Address; SA = Source Address; U/L = User Load; FCS = Frame Checksum.

Fragement A-SGP: If we have the packets prioritized by type with the pattern 10101111 used for synchronizing receiver to transmitter and 11111111 for fragmentation, then Pack 2 will always arrive earlier than Pack 1 to be included in a frame to the host. Hence no retransmission.

MAC Address Examples

Description

Define the type of the following destination addresses:

a: 4E:3B:1A:00:00:00 b: 47:20:10:32:00:00

c: FF:FF:FF:FF:FF:FF

d: 00:00:00:00:00:00

e: 00:00:00:00:00:00

f: 00:00:00:00:00:00

g: 00:00:00:00:00:00

h: 00:00:00:00:00:00

i: 00:00:00:00:00:00

j: 00:00:00:00:00:00

k: 00:00:00:00:00:00

l: 00:00:00:00:00:00

m: 00:00:00:00:00:00

n: 00:00:00:00:00:00

o: 00:00:00:00:00:00

p: 00:00:00:00:00:00

q: 00:00:00:00:00:00

r: 00:00:00:00:00:00

s: 00:00:00:00:00:00

t: 00:00:00:00:00:00

u: 00:00:00:00:00:00

v: 00:00:00:00:00:00

w: 00:00:00:00:00:00

x: 00:00:00:00:00:00

y: 00:00:00:00:00:00

z: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

gg: 00:00:00:00:00:00

hh: 00:00:00:00:00:00

ii: 00:00:00:00:00:00

jj: 00:00:00:00:00:00

kk: 00:00:00:00:00:00

ll: 00:00:00:00:00:00

mm: 00:00:00:00:00:00

nn: 00:00:00:00:00:00

oo: 00:00:00:00:00:00

pp: 00:00:00:00:00:00

qq: 00:00:00:00:00:00

rr: 00:00:00:00:00:00

ss: 00:00:00:00:00:00

tt: 00:00:00:00:00:00

uu: 00:00:00:00:00:00

vv: 00:00:00:00:00:00

ww: 00:00:00:00:00:00

xx: 00:00:00:00:00:00

yy: 00:00:00:00:00:00

zz: 00:00:00:00:00:00

aa: 00:00:00:00:00:00

bb: 00:00:00:00:00:00

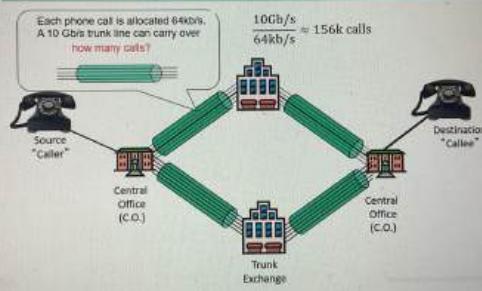
cc: 00:00:00:00:00:00

dd: 00:00:00:00:00:00

ee: 00:00:00:00:00:00

ff: 00:00:00:00:00:00

A concrete example: How many connections can be supported?



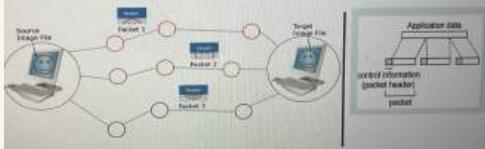
Summary of circuit switching

Advantages	Disadvantages
<ul style="list-style-type: none">Continuous data transfer without the overhead associated with packetsThe connection remains reserved and protected from competing users	<ul style="list-style-type: none">Resources occupied by particular connectionsScalability limitationFor some of the time, the connection may have no dataData rate is limited

Basic operations in packet switching

Data are transmitted in short packets

- Longer messages are split into a series of packets
- A packet is typically at the order of 1000 bytes
- Each packet contains a portion of user data plus some control info
- Control info contains at least routing (addressing) info



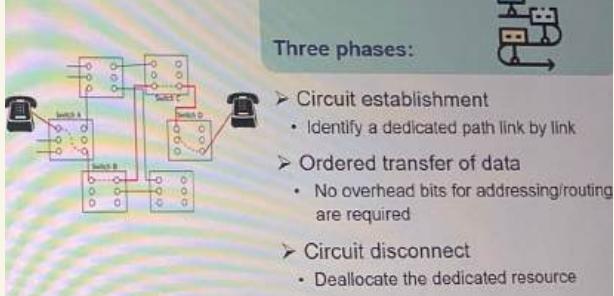
Summary of packet switching

Advantages	Disadvantages
<ul style="list-style-type: none">Efficient<ul style="list-style-type: none">Node-to-node links can be dynamically shared by many packets.Packets are queued up and transmitted as fast as possible.Even with heavy traffic, packets are still accepted in packet-switching networks.	<ul style="list-style-type: none">The queue of packets may cause a longer delayPackets may get lost and need to be retransmitted without a stable connection

Which one of the following statements is false?

Correct answer: D. Data communication on the current Internet uses circuit switching.

Basic operations in circuit switching



2 kinds

Datagram

- include seq num or dest address
- each packet treated independently
- packets arrive out of seq
- packets are called datagram

Virtual Circuit

- not reserve resources

Main points of this class

- Circuit switching needs to establish a dedicated path between two devices before they can communicate.
- Packet switching sends data as individual packets.
- In datagram packet switching, two devices do not need to pre-establish a path before data transfer.
- In virtual-circuit packet switching, two devices fix a path before data transfer, but no dedicated resources are allocated on the links of the path.

Usage of circuit switching and packet switching in the real world

- Telephone networks (e.g., landlines) that support voice calls only: circuit switching



- Data communication on the current Internet: packet switching



- Cellular networks — A topic for next lecture:

(G means generation.)

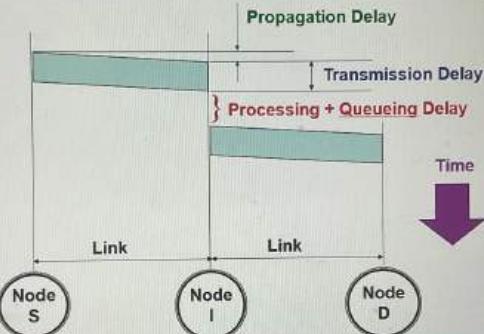
- 1G (voice only): circuit switching

- 2G and 3G (voice and limited data service): circuit switching

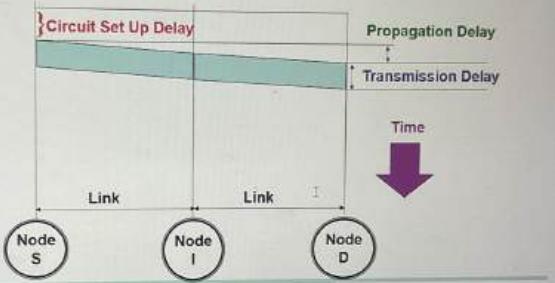
- 4G and 5G: use packet switching and transmit voice as data of higher priority over the Internet



Delay in Packet Switching Networks



Delay in Circuit Switching Networks



Learning Objectives

- **Data Transmission Technologies**
 - Understand difference between circuit and packet switched networks
 - Understand difference between datagram and virtual circuit switching
- **Delay in Packet Switched Networks**
 - Understand delay components in PSN
 - Calculate transmission delay

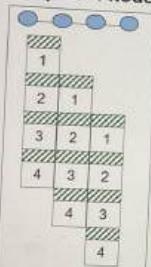
Even we assume 100 switches, switching delay is just 5% (i.e., not a major factor) of the path delay.

5. Compare the delay in sending an x -bit message over a k -hop path in a circuit-switched network and in a (lightly loaded) packet-switched network.
The circuit set up time is s sec, the propagation delay is d sec per hop, the packet size is p bits, and the data rate is b bps.
Under what conditions does the packet network have a lower delay?

Question 5's Answer — Slide 1:
With circuit switching,

Packet Transmission Delay: Overhead vs. Pipeline

Example:
3 hops & 4 nodes



$$\therefore \text{Tx delay} = 4 T_{frame} + (3-1) T_{frame}$$

In general:

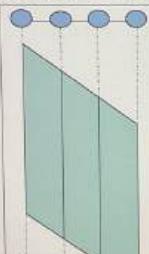
Tx Delay of Packet Switching Networks
= Tx Delay of all packets in first hop
+ (# of hops-1) * Tx Delay of 1 packet
of hops = # of nodes-1

To find the optimum packet size, other delays should also be considered:

- Processing and Queueing Delay
- Propagation Delay

Delay in Circuit Switching Networks

Example: 3 hops & 4 nodes
data size=1000 bits, link data rate=10 Kbps



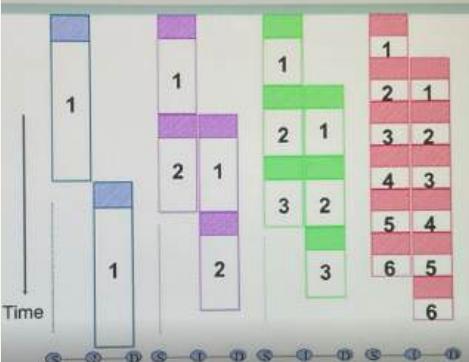
In general:

Tx Delay of Circuit Switching Networks
= all data size / data rate

To find the optimum packet size, other delays should also be considered:

- Circuit Set Up Delay
- Propagation Delay

Pipeline Effect



CE3005/CZ3006 Computer Networks Part II

Focus of Part II Lectures:

Application	Web (HTTP)	E-mail (SMTP)	Others (FTP, IM)
Transport	TCP		
Network	IP		
Data Link	LAN (Ethernet) Backbone (FDDI) Wireless LAN (802.11n, 6.9i) MAN/WAN (IPX/SPX, TCP/IP, SUSE, ATM, Frame Relay)		
Physical	Physical (Optical Fiber, Coaxial Cables)		
WAN	LAN	Internet	Physical (Optical Fiber, Coaxial Cables)

Part I Lectures:

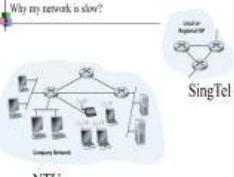
An Overview of Computer Networks and the Internet

<https://www.straitstimes.com/tech/singapore-tops-fixed-broadband-speed-rankings-but-places-4th-for-mobile>

World's fastest broadband nations

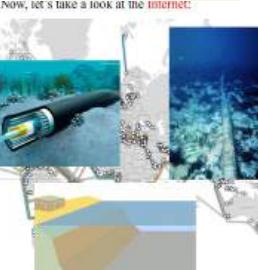
Rank	Place	Avg. download speed (Mbps)	Download a 7500 MB file (Seconds)
1	Singapore	82.10	10.34
2	Sweden	40.10	25.30
3	Taiwan	34.4	29.46
4	Denmark	33.94	30.52
5	The Netherlands	33.52	30.53
6	Latvia	30.36	33.43
7	Norway	29.13	35.09
8	Belgium	27.37	37.25
9	Hong Kong	27.16	37.42
10	Switzerland	26.95	38.01
11	Lithuania	25.75	40.45
12	Finland	24.47	40.91
13	Estonia	24.31	42.39
14	Jersey, United Kingdom	23.3	45.87
15	Hungary	23.0	44.12
16	Republic of Korea	22.9	44.43

Why my network is slow?



It is more likely that there are hardware/software issues beyond your local connection.

Now, let's take a look at the Internet:

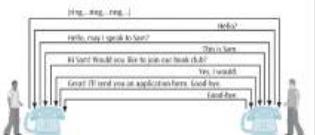


To improve performance, each station can even be connected directly to the **switch** at the access layer.



You've learnt the concept of a **communication protocol**, which is a set of **rules** defining the **format** and the **order** of messages exchanged between two parties.

When making a phone call, the recipient waits for the ring tone before answering. Then both will take turns to talk. This is a communication protocol.



An Overview of Computer Networks and the Internet

<https://www.straitstimes.com/tech/singapore-tops-fixed-broadband-speed-rankings-but-places-4th-for-mobile>

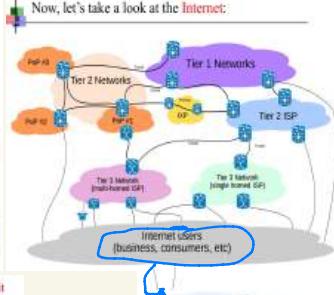
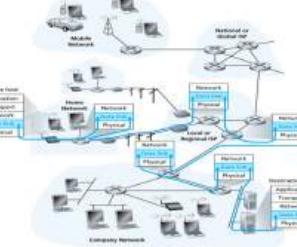
Speedtest Global Index



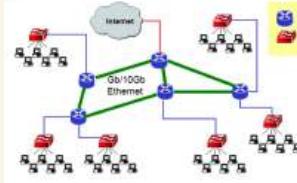
Typically, ISPs interconnect at public locations called Internet eXchange Points (IXPs) or Network Access Points (NAPs).

Point of Presence (POP) is the location where the ISP houses its network hardware (mostly routers) for subscribers to connect.

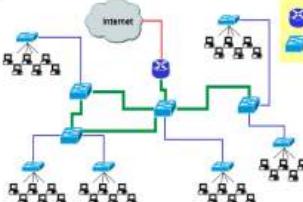
Now, let's take a look at the Internet:



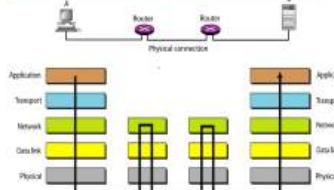
Nowadays, it's getting common to see **Gigabit/10 Gigabit high speed Ethernets** being used at the core layer to interconnect LANs.



In addition, it's getting common to see a **fully-switched network** consisting of 10/100Gb/10Gb Ethernets.



The complex task of **computer networking** is subdivided into layers called **protocol stack**. OSI is the standard but TCP/IP is in use, i.e., default standard.

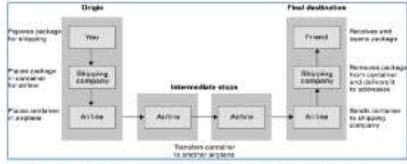


NTU IT Infrastructure

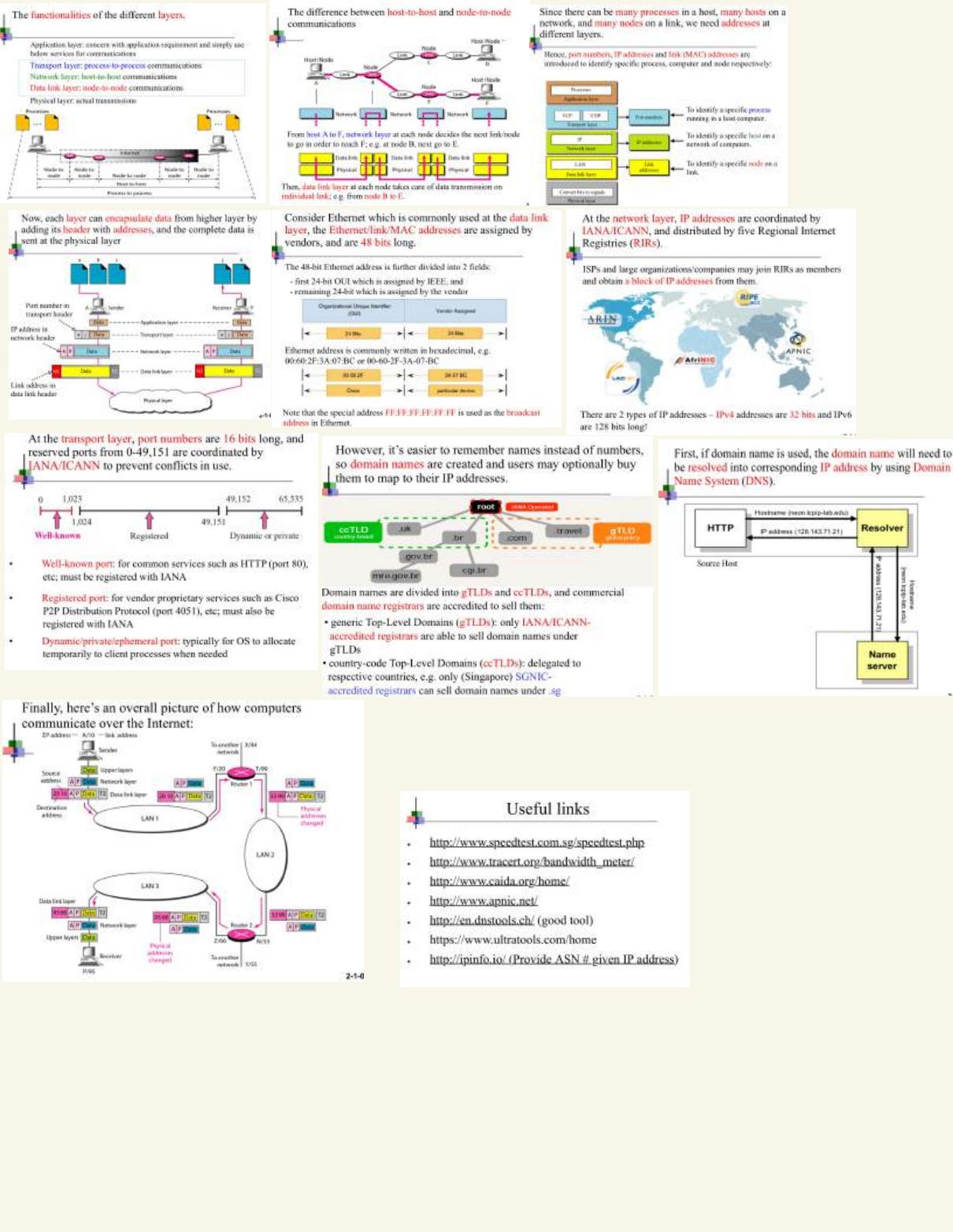


The **idea of layering** is to make a complex task manageable - each layer performs a simpler subtask and uses the services provided by lower layers.

As an analogy, to send a package to an overseas friend, we only need to take care of packing/unpacking, and use the service of a shipping company, which in turn uses the service of an airline.



ENCAPSULATION occurs as information is passed from one layer to the next.



Classless Inter-Domain Routing (RFC 1918 - 1919)

- Abandon the notion of classful addressing
- Key concept: length of network id (prefix) can be any length
- Consequence: add a network mask to IP (similar concept as subnet mask)



Router to router link (subnet)

- An IP address needs to be assigned to each active interface of a router.
- To optimise the use of IP address we usually assign a /30 to the link, eg. 172.16.31.0/30

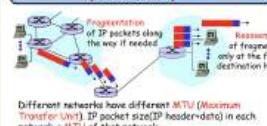


Network Address Translation (RFC 2663, 3022)

By using a NAT-enabled router, only 1 IP address is required from ISP to support the whole private network to connect to Internet.



IP Fragmentation & Reassembly (RFC 791 and 815)



IP Routing

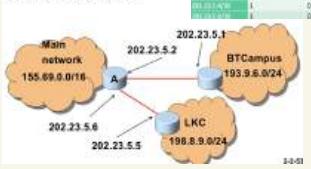
Now, we are ready to see how Internet Protocol works:



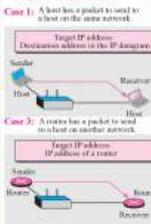
Typically, a host will not know how to send packets to destination outside its own network. Hence, it is configured with a **default gateway** (router) to assist in the forwarding.

Example

Content of Router A



ARP Packet - Target IP Address



Case 2: A host has a packet to send to a host on another network.



Case 4: A router has a packet to send to a host on the same network.



ARP packet is sent directly over Ethernet frame:



ARP Example

• ARP Request from Argon(Client):

Source hardware address: 00:02:24:71:e4:44
Source protocol address: 128.143.137.144
Target hardware address: 00:00:00:00:00:00
Target protocol address: 128.143.137.1

• ARP Reply from Router137:

Source hardware address: 00:e0:99:23:a8:20
Source protocol address: 128.143.137.1
Target hardware address: 00:02:24:71:e4:44
Target protocol address: 128.143.137.144

Since sending an ARP request/reply for each IP datagram is inefficient, hosts maintain a cache (ARP Cache) of current entries. Typically, the entries are configured to expire after 2-20 minutes.

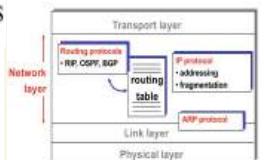
IP Address	MAC Address	Expiration Time
128.143.137.1	00:02:24:71:e4:44	2023-09-15 10:00:00
128.143.137.2	00:02:24:71:e4:45	2023-09-15 10:00:00
128.143.137.3	00:02:24:71:e4:46	2023-09-15 10:00:00
128.143.137.4	00:02:24:71:e4:47	2023-09-15 10:00:00
128.143.137.5	00:02:24:71:e4:48	2023-09-15 10:00:00
128.143.137.6	00:02:24:71:e4:49	2023-09-15 10:00:00
128.143.137.7	00:02:24:71:e4:4A	2023-09-15 10:00:00
128.143.137.8	00:02:24:71:e4:4B	2023-09-15 10:00:00
128.143.137.9	00:02:24:71:e4:4C	2023-09-15 10:00:00
128.143.137.10	00:02:24:71:e4:4D	2023-09-15 10:00:00
128.143.137.11	00:02:24:71:e4:4E	2023-09-15 10:00:00
128.143.137.12	00:02:24:71:e4:4F	2023-09-15 10:00:00
128.143.137.13	00:02:24:71:e4:50	2023-09-15 10:00:00
128.143.137.14	00:02:24:71:e4:51	2023-09-15 10:00:00
128.143.137.15	00:02:24:71:e4:52	2023-09-15 10:00:00
128.143.137.16	00:02:24:71:e4:53	2023-09-15 10:00:00
128.143.137.17	00:02:24:71:e4:54	2023-09-15 10:00:00
128.143.137.18	00:02:24:71:e4:55	2023-09-15 10:00:00
128.143.137.19	00:02:24:71:e4:56	2023-09-15 10:00:00
128.143.137.20	00:02:24:71:e4:57	2023-09-15 10:00:00
128.143.137.21	00:02:24:71:e4:58	2023-09-15 10:00:00
128.143.137.22	00:02:24:71:e4:59	2023-09-15 10:00:00
128.143.137.23	00:02:24:71:e4:5A	2023-09-15 10:00:00
128.143.137.24	00:02:24:71:e4:5B	2023-09-15 10:00:00
128.143.137.25	00:02:24:71:e4:5C	2023-09-15 10:00:00
128.143.137.26	00:02:24:71:e4:5D	2023-09-15 10:00:00
128.143.137.27	00:02:24:71:e4:5E	2023-09-15 10:00:00
128.143.137.28	00:02:24:71:e4:5F	2023-09-15 10:00:00
128.143.137.29	00:02:24:71:e4:60	2023-09-15 10:00:00
128.143.137.30	00:02:24:71:e4:61	2023-09-15 10:00:00
128.143.137.31	00:02:24:71:e4:62	2023-09-15 10:00:00
128.143.137.32	00:02:24:71:e4:63	2023-09-15 10:00:00
128.143.137.33	00:02:24:71:e4:64	2023-09-15 10:00:00
128.143.137.34	00:02:24:71:e4:65	2023-09-15 10:00:00
128.143.137.35	00:02:24:71:e4:66	2023-09-15 10:00:00
128.143.137.36	00:02:24:71:e4:67	2023-09-15 10:00:00
128.143.137.37	00:02:24:71:e4:68	2023-09-15 10:00:00
128.143.137.38	00:02:24:71:e4:69	2023-09-15 10:00:00
128.143.137.39	00:02:24:71:e4:6A	2023-09-15 10:00:00
128.143.137.40	00:02:24:71:e4:6B	2023-09-15 10:00:00
128.143.137.41	00:02:24:71:e4:6C	2023-09-15 10:00:00
128.143.137.42	00:02:24:71:e4:6D	2023-09-15 10:00:00
128.143.137.43	00:02:24:71:e4:6E	2023-09-15 10:00:00
128.143.137.44	00:02:24:71:e4:6F	2023-09-15 10:00:00
128.143.137.45	00:02:24:71:e4:70	2023-09-15 10:00:00
128.143.137.46	00:02:24:71:e4:71	2023-09-15 10:00:00
128.143.137.47	00:02:24:71:e4:72	2023-09-15 10:00:00
128.143.137.48	00:02:24:71:e4:73	2023-09-15 10:00:00
128.143.137.49	00:02:24:71:e4:74	2023-09-15 10:00:00
128.143.137.50	00:02:24:71:e4:75	2023-09-15 10:00:00
128.143.137.51	00:02:24:71:e4:76	2023-09-15 10:00:00
128.143.137.52	00:02:24:71:e4:77	2023-09-15 10:00:00
128.143.137.53	00:02:24:71:e4:78	2023-09-15 10:00:00
128.143.137.54	00:02:24:71:e4:79	2023-09-15 10:00:00
128.143.137.55	00:02:24:71:e4:7A	2023-09-15 10:00:00
128.143.137.56	00:02:24:71:e4:7B	2023-09-15 10:00:00
128.143.137.57	00:02:24:71:e4:7C	2023-09-15 10:00:00
128.143.137.58	00:02:24:71:e4:7D	2023-09-15 10:00:00
128.143.137.59	00:02:24:71:e4:7E	2023-09-15 10:00:00
128.143.137.60	00:02:24:71:e4:7F	2023-09-15 10:00:00
128.143.137.61	00:02:24:71:e4:80	2023-09-15 10:00:00
128.143.137.62	00:02:24:71:e4:81	2023-09-15 10:00:00
128.143.137.63	00:02:24:71:e4:82	2023-09-15 10:00:00
128.143.137.64	00:02:24:71:e4:83	2023-09-15 10:00:00
128.143.137.65	00:02:24:71:e4:84	2023-09-15 10:00:00
128.143.137.66	00:02:24:71:e4:85	2023-09-15 10:00:00
128.143.137.67	00:02:24:71:e4:86	2023-09-15 10:00:00
128.143.137.68	00:02:24:71:e4:87	2023-09-15 10:00:00
128.143.137.69	00:02:24:71:e4:88	2023-09-15 10:00:00
128.143.137.70	00:02:24:71:e4:89	2023-09-15 10:00:00
128.143.137.71	00:02:24:71:e4:8A	2023-09-15 10:00:00
128.143.137.72	00:02:24:71:e4:8B	2023-09-15 10:00:00
128.143.137.73	00:02:24:71:e4:8C	2023-09-15 10:00:00
128.143.137.74	00:02:24:71:e4:8D	2023-09-15 10:00:00
128.143.137.75	00:02:24:71:e4:8E	2023-09-15 10:00:00
128.143.137.76	00:02:24:71:e4:8F	2023-09-15 10:00:00
128.143.137.77	00:02:24:71:e4:90	2023-09-15 10:00:00
128.143.137.78	00:02:24:71:e4:91	2023-09-15 10:00:00
128.143.137.79	00:02:24:71:e4:92	2023-09-15 10:00:00
128.143.137.80	00:02:24:71:e4:93	2023-09-15 10:00:00
128.143.137.81	00:02:24:71:e4:94	2023-09-15 10:00:00
128.143.137.82	00:02:24:71:e4:95	2023-09-15 10:00:00
128.143.137.83	00:02:24:71:e4:96	2023-09-15 10:00:00
128.143.137.84	00:02:24:71:e4:97	2023-09-15 10:00:00
128.143.137.85	00:02:24:71:e4:98	2023-09-15 10:00:00
128.143.137.86	00:02:24:71:e4:99	2023-09-15 10:00:00
128.143.137.87	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.88	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.89	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.90	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.91	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.92	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.93	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.94	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.95	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.96	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.97	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.98	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.99	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.100	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.101	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.102	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.103	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.104	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.105	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.106	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.107	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.108	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.109	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.110	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.111	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.112	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.113	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.114	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.115	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.116	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.117	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.118	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.119	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.120	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.121	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.122	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.123	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.124	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.125	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.126	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.127	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.128	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.129	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.130	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.131	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.132	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.133	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.134	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.135	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.136	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.137	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.138	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.139	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.140	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.141	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.142	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.143	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.144	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.145	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.146	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.147	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.148	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.149	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.150	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.151	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.152	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.153	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.154	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.155	00:02:24:71:e4:9C	2023-09-15 10:00:00
128.143.137.156	00:02:24:71:e4:9D	2023-09-15 10:00:00
128.143.137.157	00:02:24:71:e4:9E	2023-09-15 10:00:00
128.143.137.158	00:02:24:71:e4:9F	2023-09-15 10:00:00
128.143.137.159	00:02:24:71:e4:9A	2023-09-15 10:00:00
128.143.137.160	00:02:24:71:e4:9B	2023-09-15 10:00:00
128.143.137.161	00:02:24:71:e4:9C	

Network Layer - IP Routing Protocols

Contents

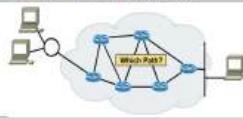
- Internet Routing
 - > Concept of Autonomous System (AS)
 - > Intra-AS and Inter-AS Routing
- Intra-AS Routing
 - > Distance Vector Routing
 - e.g. Routing Information Protocol (RIP)
 - > Link State Routing
 - e.g. Open Shortest Path First (OSPF)
- Inter-AS Routing
 - > Path Vector Routing
 - e.g. Border Gateway Protocol (BGP)

Routing: Flooding



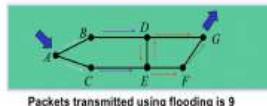
Router

A router is a device used to interconnect networks, and to forward packets by examining the destination address in the IP header of each packet.



Routing: Flooding

- A straight forward way of routing is by flooding.
- When a node receives a packet, it will forward the packet to all other links except the incoming link. The packet will be labeled with a unique identifier.
- Should the same packet return, the packet will be discarded.



Routing in the Internet

For routing purpose, Internet is divided into Autonomous Systems (AS). An AS is a group of routers under the authority of a single administration; e.g. an ISP.



Routing in the Internet

For routing purpose, Internet is divided into Autonomous Systems (AS). An AS is a group of routers under the authority of a single administration; e.g. an ISP.



Advantages:

- A packet will always get through if one or more path exists (very robust)

Disadvantages:

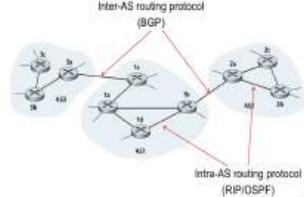
- Very wasteful of bandwidth, may cause serious congestion, hence not used in the Internet

Applications:

- Military applications (routers may be destroyed anytime)
- Ad hoc wireless networks (nodes may be turned off or moved away anytime)

Intra-AS and Inter-AS Routing

In practice, routing in Internet is done in a hierarchical manner, which includes intra-AS and inter-AS routings.



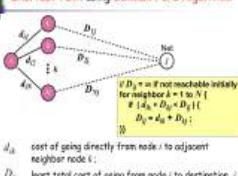
Intra-AS Routing: Distance Vector Routing

Distance Vector routing also known as "Bellman-Ford" or "old ARPANET" routing.

Essentially, consists of 3 main stages:

- Discover neighbors by multicasting request
- Exchange distance vectors (routing information) with immediate neighbors only
 - Response to request
 - Periodic updates (typically 30s interval)
 - Triggered updates due to changes
- Compute shortest-path routes (using Bellman-Ford algorithm)

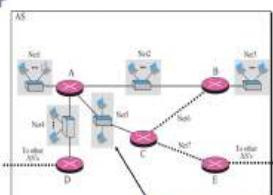
Distance Vector Routing: Computing Shortest-Path using Bellman-Ford Algorithm



d_{ik} cost of going directly from node i to adjacent neighbor node k ;

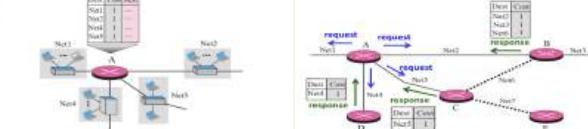
D_{ij} least total cost of going from node i to destination j

An example of Distance Vector Routing



Distance Vector Routing: Initially, a router only has its own configured routing table.

Distance Vector Routing: Discover Adjacent Neighbors and exchange distance vectors.



Quote about Distance vector

Distance Vector Routing: Count-to-Infinity problem

Before failure:

After failure:

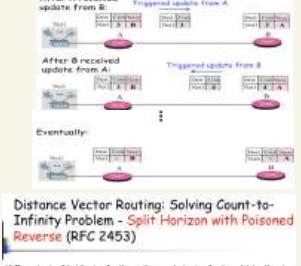
After A received update from B:

After B received update from A:

Triggered update from B:

Eventually:

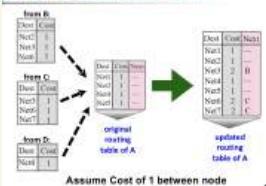
Distance Vector Routing: Solving Count-to-Infinity Problem - Split Horizon with Poisoned Reverse (RFC 2453)



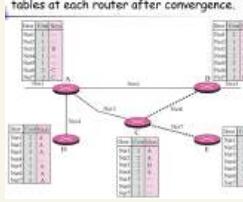
If B gets to Net1 via A, then its update to A should indicate that Net1 is unreachable.

Periodic update from B

Distance Vector Routing: Computing Shortest-Path using Bellman-Ford Algorithm



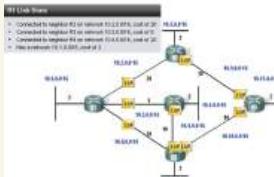
Distance Vector Routing: Resulting routing tables at each router after convergence.



Intra-AS Distance Vector Routing: Routing Information Protocol

- RIP uses Distance Vector algorithm, cost is simply based on the number of hops
- Allows maximum 15 hops, 16 indicates ∞
- Routing information exchanged every 30 sec via Response Message
- If no advertisement heard after 180 seconds \rightarrow neighbor/link declared dead
- RIP related RFC Documentations:
 - RFC 1058, 1387, 1388, 1723 (RIP version 2)

Link State Routing: Flood LSP to ALL routers

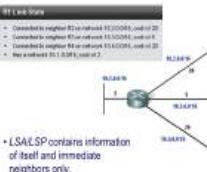
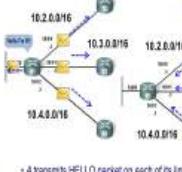


Intra-AS and Inter-AS Routing Protocols

Intra-AS Routing: Link State Routing

Link State Routing: Discover Neighbors

Link State Routing: Construct Link State Advertisement Packet (LSA/LSP)



Link State Routing: Build Link State Database

LSA (Link State Advertisement)

LSDB (Link State Database)

IP Routing Table

Dijkstra's Algorithm

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

IP Header

TCP Header

BGP Packet

Protocol design principles

LSA

LSDB

Transport Layer – UDP and TCP

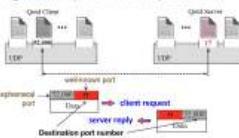
Contents

- Transport Layer
 - > Port Numbers
 - Connectionless Service
 - User Datagram Protocol (UDP)
 - Connectionless Service
 - Transmission Control Protocol (TCP)
 - > Connection Management
 - Flow Control
 - Error Control
 - Congestion Control

UDP - Datagram Service

Application layer is aware that UDP sends each message as a datagram.

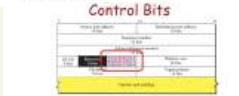
e.g. In Java, `request = new DatagramSocket();`



Transmission Control Protocol

- To support applications requiring reliable communications, TCP adds **reliability over unreliable IP**.

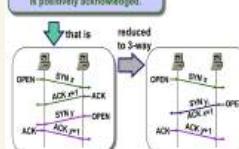
- Essentially, TCP features:**
 - **Connection Management:** A connection must be setup before data exchange can be performed.
 - **Flow Control:** Sender will not overwhelm receiver.
 - **Error Control:** Receiver detects errors, sender retransmits error packets.
 - **Congestion Control:** During transmission, sender detects network usage/congestion and adjust transmission rate.



- SYN = 1:** Synchronization sequence numbers
 - Used to establish connection
 - FIN = 1: No more data from me!
 - Used to terminate connection
 - RST = 1: Reset the connection
 - When error occurs during connection establishment
 - ACK = 1: Acknowledgment number is valid

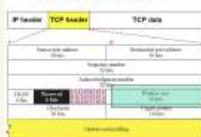
Solution: Three-way Handshake

Synchronization can be made reliable if each connection request is positively acknowledged.



- Note:**
 - A SYN and FIN segment does not carry data but consumes one sequence number
 - A SYN+ACK, FIN+ACK does not carry data but consumes one sequence number
 - ACK segment, carrying no data, does not consume any sequence number.

TCP Header Format



TCP Flow Control

So that sender won't overrun receiver's buffers by transmitting too much, too fast.

Similar to sliding window flow control in datagram layers, though some details are different.

To support bi-directional data transfer, 2 pairs of windows are used:

- A \rightarrow sender window
- A \rightarrow receiver window
- B \rightarrow sender window
- B \rightarrow receiver window

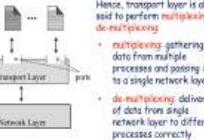
Transport Layer

- Transport layer provides end-to-end service for transferring data between processes (process-to-process communication).
- Only implemented at the end hosts.



Transport Layer - Ports

- A single transport layer is used to support multiple application processes through the use of ports.



Transport Protocols in the Internet

The Internet supports 2 main transport protocols:

UDP - User Datagram Protocol

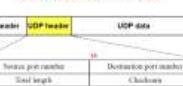
- unreliable, connectionless
- datagram oriented
- simple
- example applications:
 - routing (IP), domain name service (DNS), DHCP, real-time protocol, etc.

TCP - Transmission Control Protocol

- reliable, connection-oriented
- stream oriented
- complex
- example applications:
 - web (http), email (smtp), file transfer (ftp), video streaming, etc.



UDP Header Format



Length of UDP Datagram including header (in bytes):
Minimum = 8 bytes, no data, only header.
Maximum = 65505

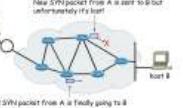
TCP Header Format



- Sequence Number (SN):
 - Each TCP connection will start with a different SN called Initial Sequence Number (ISN)
 - The position of each data byte in the byte stream is labelled by the sequence number. E.g., 1st byte (ISN=1) next 2nd byte (ISN=2) next 3rd byte (ISN=3) ...
 - SN indicates the position of the 1st byte in each segment.
- Acknowledgment Number (AN):
 - All bytes after the first byte of data expected from sender.
 - Window Size (WS):
 - Indicates the number of bytes (called *segment*) counting from the SN that the receiver is ready to accept.

Note: Other fields will be discussed in relevant slides later.

Recall that TCP is unreliable. It is possible for packets to be delayed, lost, or duplicated due to timeout mechanism.



Two-way Handshake without positive acknowledgement

Hosts A & B have a two-way connection for consecutive seqn. They exchange their seqn with each other.

Data exchange doesn't have any problems. But...

Old SYN packet from A is finally going to B after being delayed by congested router.

Host A receives SYN+ACK from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

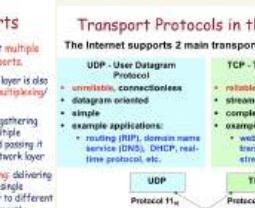
Host B sends SYN to A.

Host A receives SYN from B.

Host A sends ACK to B.

Host B receives ACK from A.

The corresponding behaviour of receiver in the sliding-window flow control is illustrated below:



Transport Layer - Ports

Transport Layer - Ports

A single transport layer is used to support multiple application processes through the use of ports.

Hence, transport layer is also used to perform multiplexing/demultiplexing.

• multiplexing gathering data from multiple processes and passing it to a single network layer

• de-multiplexing delivery of data from single network layer to correct application process.

• Hence, transport layer is also used to support connection management.

• Some applications do not need reliable communications. For example:

- broadcasting, advertising messages to users.

- sending live video streams over the internet (no reliability, rate sensitive).

Q: If UDP provides non-reliable communications, then why UDP?

A: Some applications do not need reliable communications. For example:

- broadcasting, advertising messages to users.

- sending live video streams over the internet (no reliability, rate sensitive).

Q: So, what does UDP do?

A: Provide process-to-process communication service for applications to use.

• Connection establishment: serves the following purposes

- ensure both ends are ready to communicate

- establishes initial sequence number (ISN)

- exchange window sizes (WS) in bytes

- allocate resources, e.g. buffer space, etc. to support the connection

• Connection establishment starts with a synchronization (SYN) request.

TCP Flow Control Enhancement 1 - Delayed ACK (RFC 1122)

Problem: Wasteful to send ACK only segment (40 bytes TCP+IP headers)

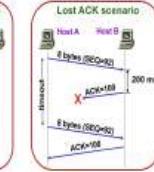
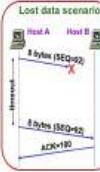
• Maximum = 100 ms, to avoid error-control timeout re-send

• ACK every alternate segment received

• Note: Piggy-backed ACK can be sent immediately



Examples of timeout retransmission scenarios



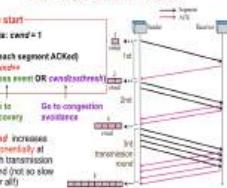
Computing Retransmission Timer RTO - Jacobson's Algorithm (RFC 6298)

- Initialization: $RTO = 1 \text{ s}$
- After 1st RTT is measured:
 - (smooth) $RTT_{SMOOTH} = RTT$
 - (RTT variation) $RTT_{VARIATION} = RTT/2$
 - $RTO = RTT_{SMOOTH} + 4 \cdot RTT_{VARIATION}$
- After each subsequent RTT is measured
 - $RTT_{VARIATION} = (1-\alpha) \cdot RTT_{VARIATION} + \beta \cdot (RTT - RTT_{SMOOTH})$, $\beta = 1/4$
 - $RTT_{SMOOTH} = (1-\alpha) \cdot RTT_{SMOOTH} + \alpha \cdot RTT$, $\alpha = 1/8$
 - $RTO = RTT_{SMOOTH} + 4 \cdot RTT_{VARIATION}$
- (minimum) $1 \leq RTO \leq \text{maximum (at least } 60\text{s)}$

Congestion

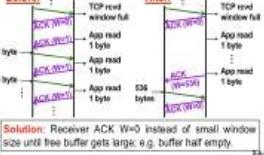


TCP Slow Start Phase



TCP Enhancement 2: avoiding Silly Window Syndrome at Receiver - Clark's solution (RFC 813, RFC 1122)

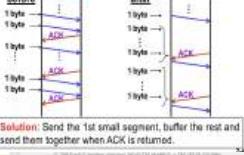
Problem: Wasteful for receiver to keep ACK with small window when sender can send more



Solution: Receiver ACKs $W/0$ instead of small window size until free buffer gets large; e.g. buffer half empty.

TCP Enhancement 3: avoiding Silly Window Syndrome at Sender - Nagle's Algorithm (RFC 896, RFC 1122)

Problem: Wasteful for sender to keep sending small segments when receiver can accept more



Solution: Send the 1st small segment, buffer the rest and send them together when ACK is returned.

TCP Error Control

WHY? So that TCP can guarantee reliable service to application layer even when IP is unreliable.

HOW? Similar to Selective-Reject in datalink layer, but the details are different.

Error types:

- Segments arriving out-of-order
 - Detected based on SN in TCP header; re-order and ACK
- Segments duplicated
 - Detected based on SN, **invalid ACK**
- Segments corrupted
 - Detected based on checksums in TCP header
 - Demand retransmission for timeout retransmission
- Segments lost
 - Wait for timeout retransmission

Retransmission Timer

Problem: How long should TCP wait for the ACK before it retransmits that segment?

- Too short (premature timeout): there will be unnecessary retransmission
- Too long (slow reaction to losses): a long period of time is required to discover a lost segment

Note: Delays in network are constantly changing in practice, so timeout must be adaptive.

Solution: Measure Round Trip Time (RTT) and compute smoothed RTT (SRTT). The Retransmission TimeOut (RTO) is then derived from SRTT.

Measuring RTT - Karn's Algorithm (RFC 1122)

- Each TCP connection measures the RTT from sending a segment to receiving its corresponding ACK.
- Typically, there is only one measurement ongoing at any time (i.e. measurements do not overlap).

Karn's Algorithm:

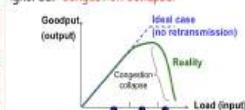
- If a segment is retransmitted due to timeout, ignore its measured RTT because it is ambiguous whether the ACK is for 1st or re-transmission.
- When retransmission occurs, set $RTO = 2 \times RTO$

Congestion Control

WHY? To prevent sending too many segments which causes every congested and useless, internally, to be a "congestion user" of the network.

HOW? Implement congestion control algorithm which controls the amount of traffic that a connection can send

What will happen if network congestion is ignored? Congestion Collapse!



- Zero retransmission, hence Output = Input
- Several retransmissions, hence output continues to increase slightly as input increases
- Transmissions are dominated by retransmissions

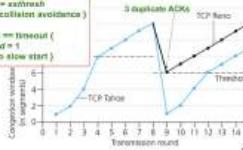
Congestion Control

TCP sender assumes network congestion when LOSS events occurred:

- Timeout or receiving duplicate ACKs
- Possibly due to queuing or buffer overflow at routers which are signs of congestion

TCP Congestion Control - Reno Algorithm: Implement Fast Recovery

Rationale: Network is too congested if other segments are getting through.



TCP Congestion Control



Assume: Maximum MSS window size is 16

Maximum MSS window size is 16

Transmission round

$$\text{Packet transmission rate} = \frac{\text{Number of packets}}{T_x + 2T_{\text{prop}}}$$

$$T_x = \frac{\text{data frame}}{\text{data rate}}$$

Contents

- **Course Logistics**
 - Teaching staff
 - Lecture
 - Tutorial
 - Lab
 - Exam
- **A Brief History of Internet**
 - How this thing gets started

Lecture



- **Time/Location**
 - Monday and Wednesday 9:30AM – 10:20AM, Lecture Theatre LT2A
 - LT2A is on the first floor of North Spine, between Block N3 and Block N4.
- **Two Parts**
 - Part I: Weeks 1-6 (Jun ZHAO)
Week 7 for E-Learning according to SCSE rules
 - Part II: Week 8-13 (Mo LI)
- **References**
 - James K. Kurose and Keith W. Ross,
Computer Networking: A Top-Down Approach (**CN**)
[https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf)
 - Douglas E. Comer, *Computer Networks and Internets* (**CNI**)
<https://bmansoori.ir/book/Computer%20Networks%20and%20Internets.pdf>

CZ3006/CE3005 - Part I

- **Focusing on Underlying Layers**
 - Physical layer resilience
 - Data link layer
 - Flow control
 - Error control
 - Local area network
 - MAC
 - Wireless LAN
 - Mobile access
 - Network architecture and performance
 - Network design patterns

Part I Syllabus – Fundamental Underlying Layers

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 23/Jan/2023 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – MAC	M1-L6-LAN-MAC.pptx
	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
Week 6:	Mobile Access Networks: From 1G to	M1-L9-Mobile.pptx

CZ3006/CE3005 – Part II

- **Covering Higher-Level Layers**
 - Applications
 - TCP protocol
 - IP protocol (main emphasis)
 - Routing process

Tutorial

- **Starting from the 3rd week**
- Try all the problems before the session
- **7 Tutorials for the whole course**
 - 6 for regular sessions
 - 1 for E-learning
- **Problems & Questions**
 - Exam questions from previous years
 - Problems asked by you



Instructors of Tutorials

Please check https://wish.wis.ntu.edu.sg/webexe/owa/aus_schedule.main for the time and location of your tutorial sessions.

Name	Tutorial groups	Email address
Anwitaman Datta	A45, A57, A58 X52	anwitaman@ntu.edu.sg
Dusit Niyato	A26, A33, A53, A54	dniyato@ntu.edu.sg
Mo LI	A21, A42, Z48	limo@ntu.edu.sg
A S Madhukumar	A34, A35, A37	ASMadhukumar@ntu.edu.sg
Lee Bu Sung	A29, A51, A52, Z59, Z60	ebslee@ntu.edu.sg
Rui TAN	A36, A41, Z21	tanrui@ntu.edu.sg
Jun ZHAO	A50	junzhao@ntu.edu.sg
Jun Luo	A49, Z47	junluo@ntu.edu.sg

Lab

- For Groups in even weeks:
Week 4 for Lab 1, Week 6 for Lab 2, Week 8 for Lab 3, Week 10 for Lab 4.
- For Groups in odd weeks:
Week 5 for Lab 1, Week 7 for Lab 2, Week 9 for Lab 3, Week 11 for Lab 4.
- You can check the time and location at:
https://wish.wis.ntu.edu.sg/webexe/owa/aus_schedule.main

Lab

The site in the last slide uses this Venue encoding:

- HPL_1 to HPL_2 = Hardware Project Lab (N4-01c-09)
- HW1_1 to HW1_2 = Hardware Lab 1 (N4-01a-03)
- HW2_1 to HW2_4 = Hardware Lab 2 (N4-01b-05)
- HW3_1 to HW3_2 = Hardware Lab 3 (N4-B1a-05)
- SPL_1 to SPL_2 = Software Project Lab (N4-B1b-11)
- SW1_1 to SW1_2 = Software Lab 1 (N4-01a-02)
- SW2_1 to SW2_2 = Software Lab 2 (N4-01c-06)
- SW3_1 to SW3_3 = Software Lab 3 (N4-B1c-14)

Lab

4 Labs

Physical attendance is compulsory for labs.

**Lab submissions are done via the NTULearn lab sites.
(Use the NTULearn lab site, not the main course site.)**

Online submission at the end of your lab session.

Marks of Labs (40% of your total score):

Lab 1: 10% of your total score

Lab 2: not graded (The purpose of Lab 2 is to practice socket programming, which will be useful for Lab 4).

Lab 3: 10% of your total score

Lab 4: 20% of your total score

Lab

- **4 lab sessions**
 - Lab 1: Understanding Networking with Internet technology
 - Lab 2: Programming network application using socket (basically writing a UDP Quote of Day client).
 - Lab 3: Understanding Network Operations and Encapsulation by Sniffing and Analysing Network Packets
 - Lab 4: Analyzing Network traffic log data using Python.

Lab 4: Please start writing your code much time before the lab session.

- **Lab 4 is an assignment.**
 - Students must prepare (write their code) before the laboratory session.
 - Students will be provided with the actual Network traffic data to analyse during the lab session.
 - The submission at the end of the lab session should include your code and also your answers to the lab questions.

Instructors of Labs

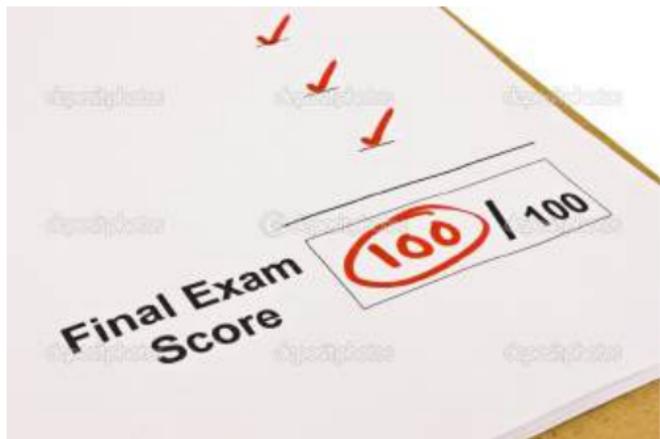
Lab groups	Instructors of Labs	TA of Labs
A37, A41, A42, A45, A52, Z21 Z47, Z48, Z59, Z60	Assoc Prof Lee Bu Sung, Francis ebslee@ntu.edu.sg	<ul style="list-style-type: none"> Premanand Rithwik (A41, A42, Z47, Z48) rithwik001@e.ntu.edu.sg Qian Liangxin (A37, A45) qian0080@e.ntu.edu.sg WhatsApp: 8790 6171 <ul style="list-style-type: none"> Si Peiyuan (A52) peiyuan001@e.ntu.edu.sg WhatsApp: 8032 4063 <ul style="list-style-type: none"> Xu Minrui (A26, Z21, Z59, Z60) minrui001@e.ntu.edu.sg
A49, A50, A51, A53, A54, A57, A58, X52	Prof Li Mo limo@ntu.edu.sg	<ul style="list-style-type: none"> Dai Gaole (X52) gaole001@e.ntu.edu.sg Si Peiyuan (A49, A50, A51) peiyuan001@e.ntu.edu.sg WhatsApp: 8032 4063 <ul style="list-style-type: none"> Yu Wenhan (A53, A54, A57, A58) wghan002@e.ntu.edu.sg WhatsApp: 92400174
A21, A26, A29, A33, A34, A35, A36	Assoc Prof Zhao Jun junzhao@ntu.edu.sg	<ul style="list-style-type: none"> Dai Gaole (A21) gaole001@e.ntu.edu.sg Lan Tianming (A33, A34, A35, A36) tianming001@e.ntu.edu.sg WhatsApp: 8039 4791 <ul style="list-style-type: none"> Qian Liangxin (A29) qian0080@e.ntu.edu.sg WhatsApp: 8790 6171

Office Hours

Role	Name	Office hours	Location	Email address	Phone/W hatsApp number	Photo
Lecturer	Jun ZHAO	Thursday 2:30pm – 5:30pm	Tutorial Room (TR) 17, 5th floor of NS4 (North Spine)	junzhao@n tu.edu.sg	8648 3534 (also WhatsApp number)	
Teaching Assistants	Wenhan YU	Monday 2:30pm – 5:30pm	Computer Network Communication Lab (CNCL), N4-B2a-01	wenhan002 @e.ntu.edu .sg	9240 0174	
	Liangxin QIAN	Tuesday 2:30pm – 5:30pm	CNCL, N4-B2a-01	qian0080@ e.ntu.edu.s g	8790 6171	
	Tianming LAN	Wednesday 2:30pm – 5:30pm	CNCL, N4-B2a-01	tianming00 1@e.ntu.ed u.sg	8039 4791	
	Peiyuan	Friday	CNCL,	peiyuan001	8032	

Exam/Grade

- **No mid-term exam**
- **One final exam (2 hours)**
- **Score= Labs (~40%) + Exam (~60%)**

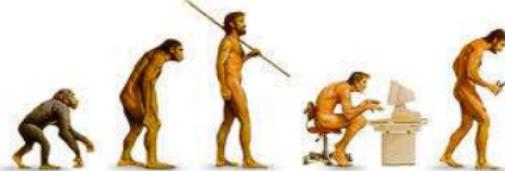


How to ACE SC2008/CZ3006/CE3005

- **Attend Tutorial**
- **Attend Lab**
- **Attend Lectures**
- **Keep your eyes open**
- **Keep your ears open**
- **Ask questions**
 - You just talked about ..., I am confused about ..., can you explain again about ...?

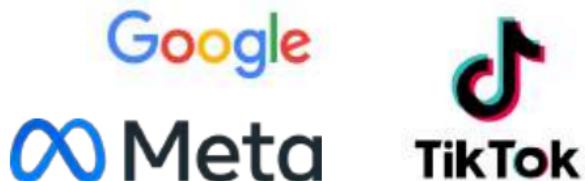


History of Internet



What is the Internet?

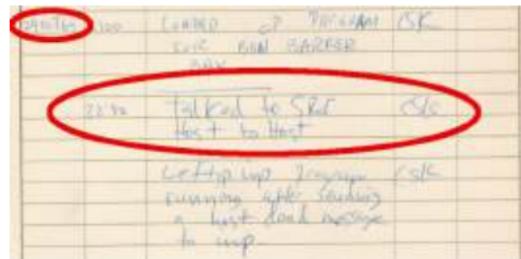
- **WWW**
- **ftp**
- **telnet**
- **Email**
- **Social networking**
- **Cryptocurrency**
- **Metaverse**



An inter-connected infrastructure for information
exchanging via standard protocols

Where Did It Come From?

- Early 1960's - DARPA (ARPA in 1960's) project headed by Licklider
- Late 1960's - ARPANET & research on packet switching by Lawrence Roberts
 - 02/09/1969 – Leonard Kleinrock's computer at UCLA became first node on the ARPANET
 - 29/10/1969 – First packets sent; Charlie Kline attempted use of remote login from UCLA to SRI; system crashed as "G" was entered
 - 05/12/1969 - Four nodes: UCLA, SRI, UCSB, University of Utah



Get more info at:
<http://www.isoc.org/internet/history/>
<http://www.packet.cc/internet.html>

History of Internet

- **1969 – First RFCs by Steve Crocker (<http://rfc.sunsite.dk/>)**
- **1971 – Email by Ray Tomlinson @ BBN**
- **1970's – Protocol development**
 - 1972-1974 TCP/IP developed by Vint Cerf & Bob Kahn
 - 1973 – Ethernet by Metcalfe @ PARC
 - 1974 TCP draft produced, split into TCP and IP in 1978
- **DNS – Distributed and scalable mechanism for resolving host names into IP addresses**
- **UC Berkeley implements TCP/IP into Unix BSD**
- **1985 – Internet used by researchers and developers**



History of Internet

- November 1988 – Internet worm affecting about 10% of the 60000 computers on the Internet (Robert Morris, Cornell)
- Tim Berners-Lee at CERN
 - Proposal for WWW in 1989
 - First web page on November 13, 1990
- Cisco(1984), Google (1998), Facebook(2004), Twitter(2006), Dropbox (2008), Instagram (2010), TikTok (2017), Trellix (2022)...
- Global average Internet speed has increased dramatically from 14.4 Kbps in 1991 to 100 Mbps in 2022. In Singapore, you can expect nearly 300 Mbps as standard.

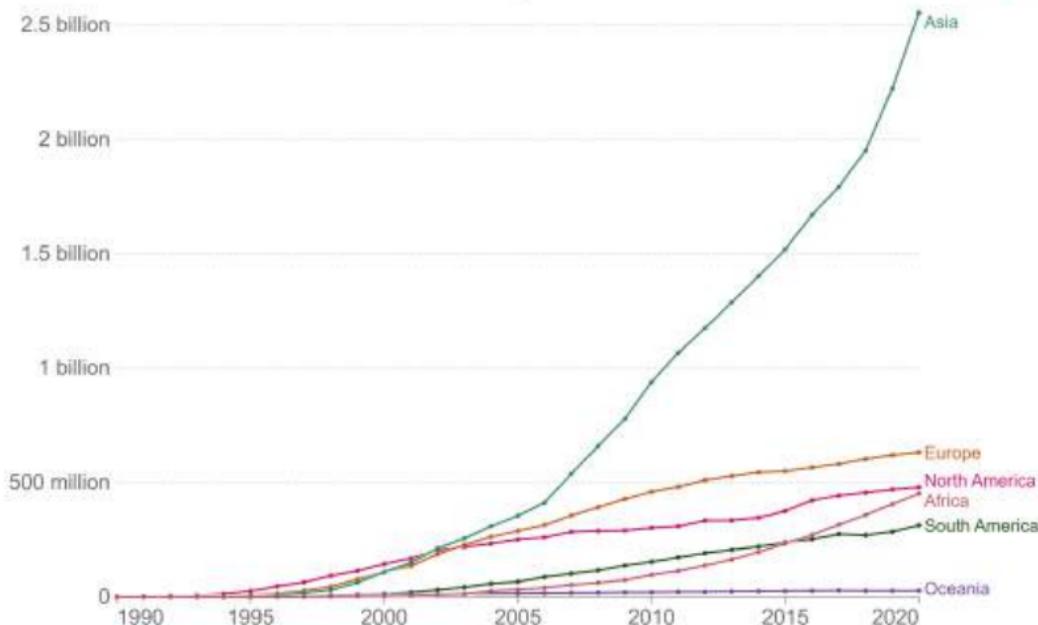


Internet Growth Trends

Number of people using the Internet

Internet users¹ are those who have used the Internet from any location in the last three months.

Our World
in Data



Source: OWID based on International Telecommunication Union (via World Bank) and UN (2022)

OurWorldInData.org/internet • CC BY

Internet Map

Father

Cerf

Kahn

Kleinrock

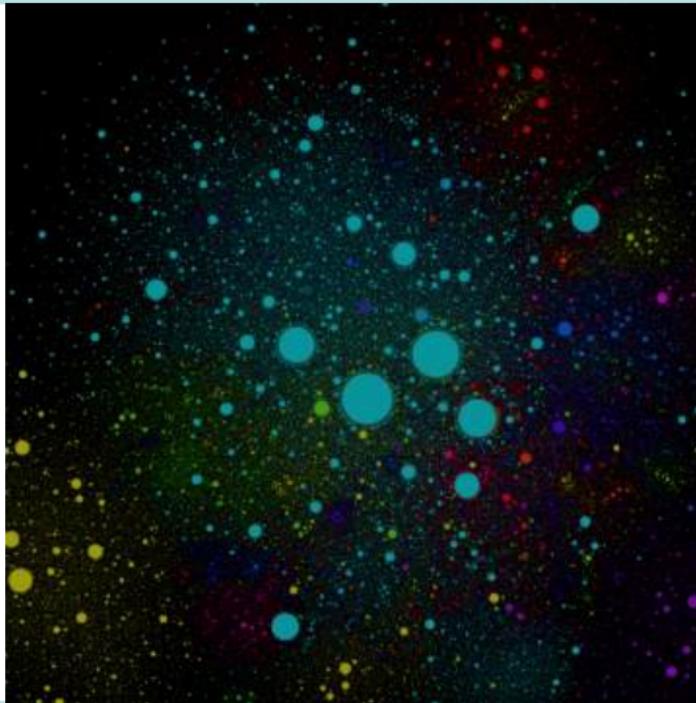
Metcalfe

Gore

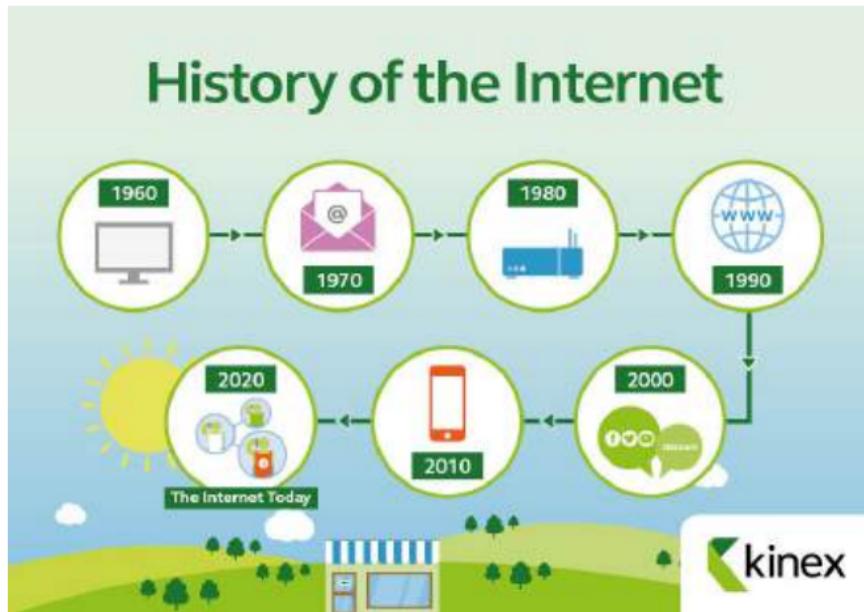
Mother

Sally

Floyd?



Brief History of the Internet



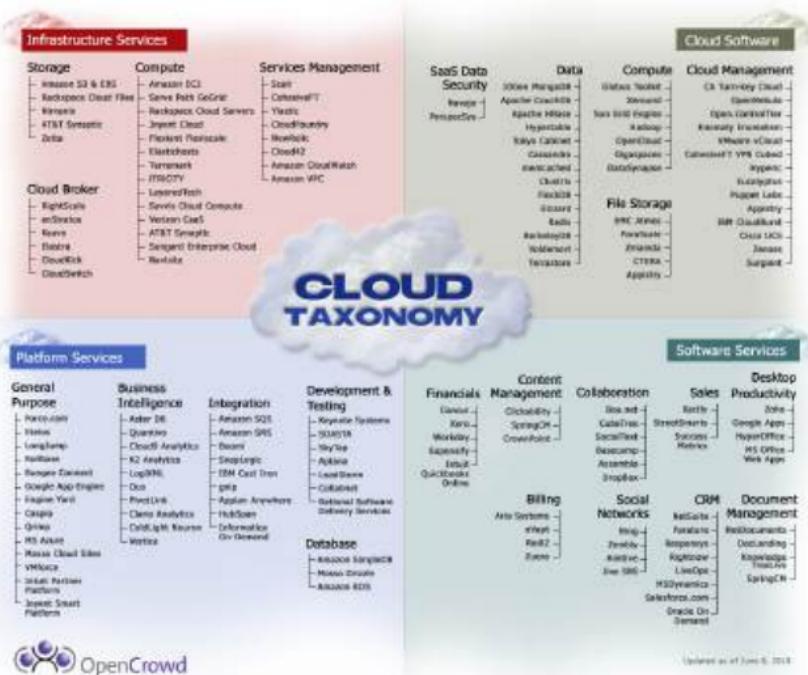
Internet Trends: Social Media

Social media landscape 2022



@FredCavazza

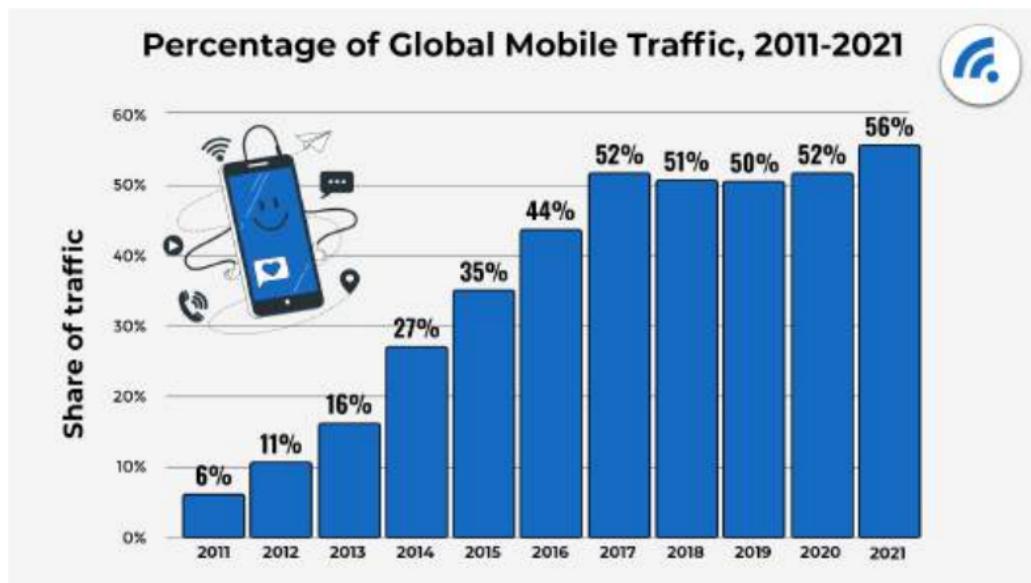
Internet Trends: Cloud Computing



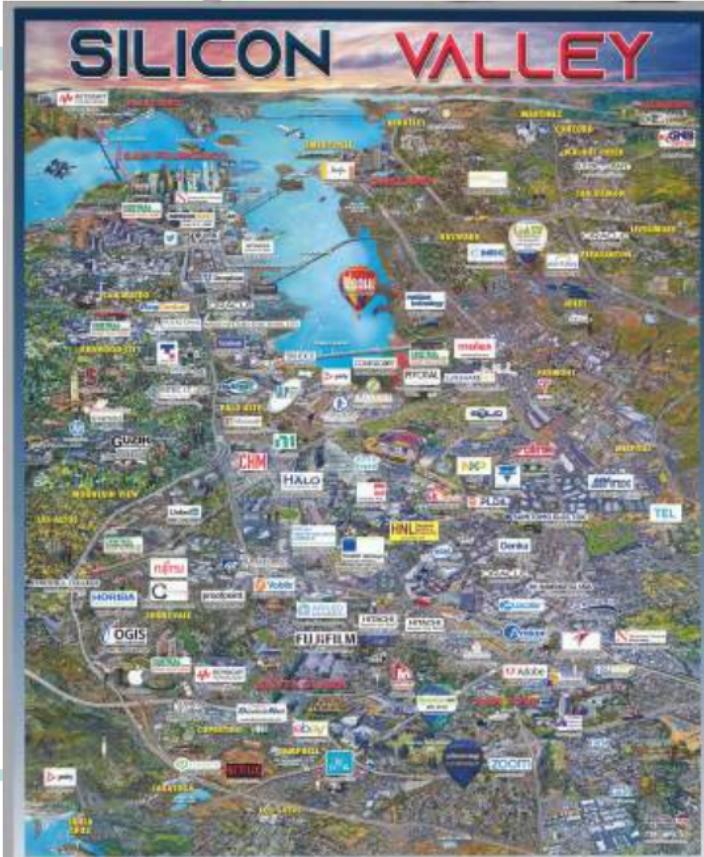
Internet Trends: Big Data



Internet Trends: Mobile Internet



Hot Spot: Silicon Valley



Lessons Learned



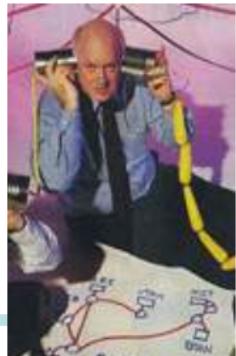
The Internet (and World Wide Web) we have today was created by some very bright, talented people who either had vision, or were inspired by other talented people's visions.

Though their ideas were not always popular, they pressed ahead.

Their perseverance and hard work brought us to where we are today.

There is a lot to be learned by studying these people, their early work and keeping in mind what they had to work with.

We, engineers, should aim to solve practical problems. Luckily, we might become rich.



Part I Syllabus

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Network paradigms	M1-L9-Paradigms.pptx

SC2008/CZ3006/CE3005

Computer Network

Lecture 2

Network Layers and Physical Resilience



Contents

- **Layered Network Architecture**
 - Motivations and Benefits
 - OSI 7-Layer Model
 - Internet 5-Layer Implementation (i.e., TCP/IP)
- **Physical Resilience**
 - Link Failure probability
 - Network resilience calculation

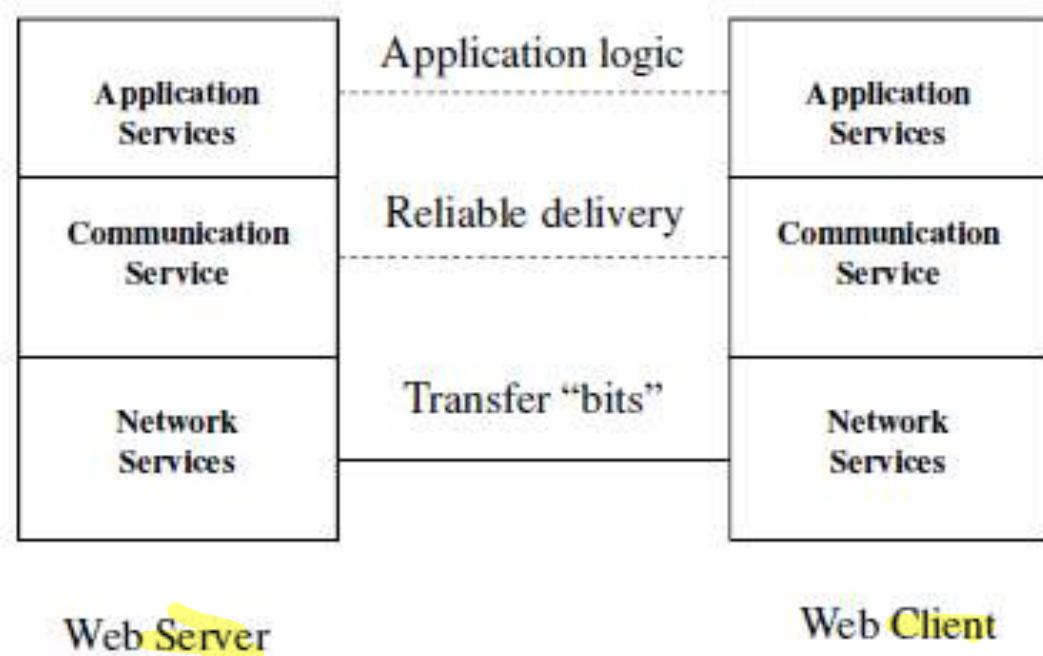
Layered Network Architecture

Motivations for Layered Network Architecture

- **Networks are complex with many pieces**
 - Hosts, routers, links, applications, protocols, hardware, software
- **Dealing with complex systems**
 - Explicit structure allowing identification, relationship of different pieces
 - Layered reference model for discussion
 - **Modularization** easing maintenance and updating
 - Change of layer's service transparent to rest of system
 - Change in network technology does not affect rest of system
 - **Layering** (design vs implementation)

A Layered Example for Web

- Browser requests web page from server
- Server determine if access is granted
- Reliable transfer page from server to client
- Physical transfer of bits from server to client



Layered Network Architecture

- **Network organized as a stack of layers**
 - Purpose of layer is to offer services to the layer above it and passes data & control information to the layer below, using a well-defined interface
 - Reducing design complexity
- **Protocols:** a set of rules governing communication between two peering parties/computers
 - define format, order of messages sent and received among network entities, and actions taken on message transmission & receipt.
- **Network Architecture:** a set of layers and protocols with specifications enabling hardware/software developers to build systems compliant with a particular architecture

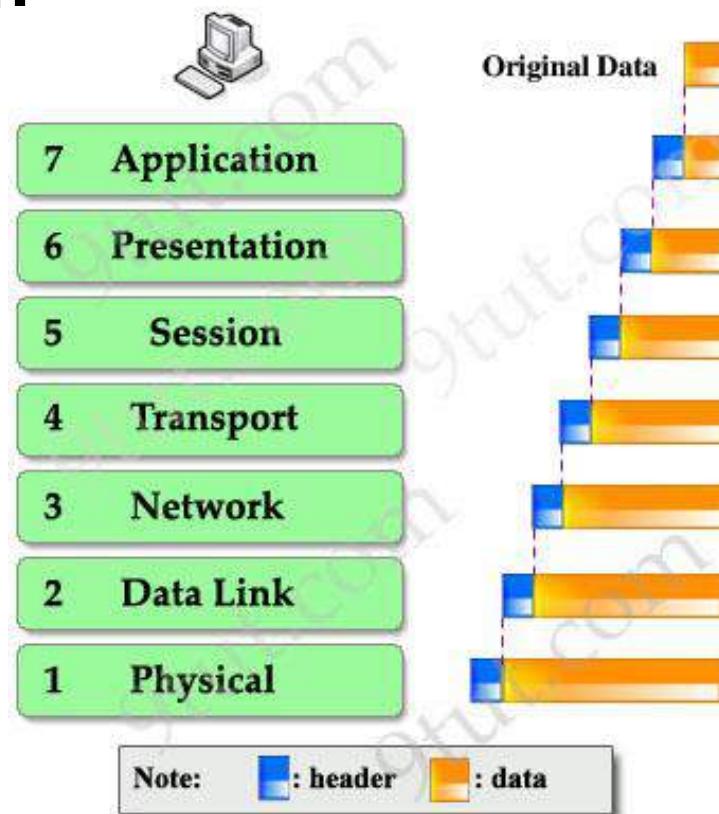


Benefits of Layers

- **Simplicity**
 - Easy to design once layers and their interactions are defined clearly
- **Flexibility**
 - Easy to modify and develop networks by separate layers modifications
- **Incremental Changes**
 - Easy to add new layers, add new functions to a layer

OSI 7-Layer Model

- **Function Decomposition**
 - Weakly-decoupled stack
- **Encapsulation**
 - Each layer adding new headers
- **Peering**
 - Only peer layer “communicating” with each other

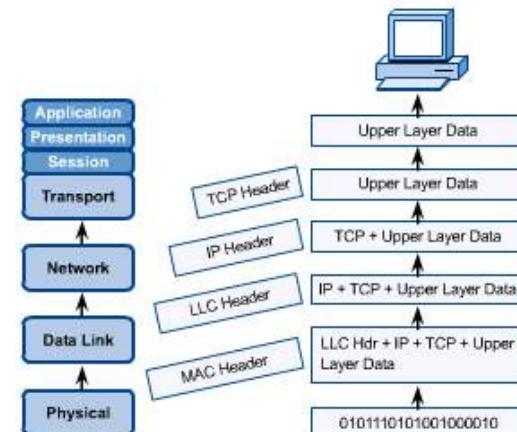
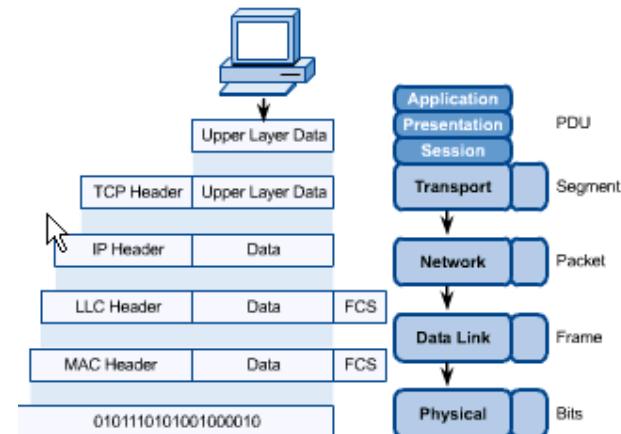


OSI Reference Model: 7 Layers (More on Supplementary Materials)

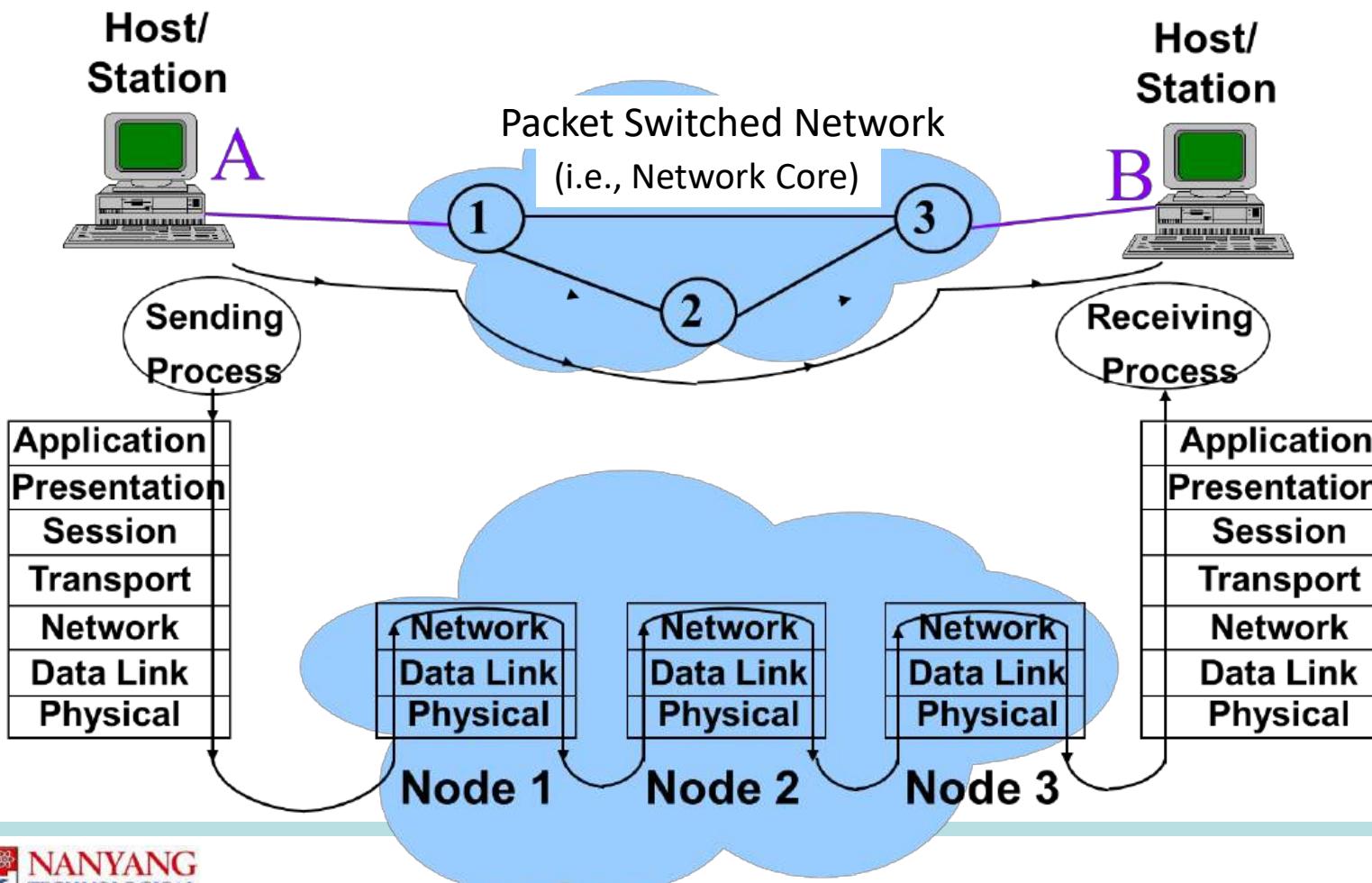
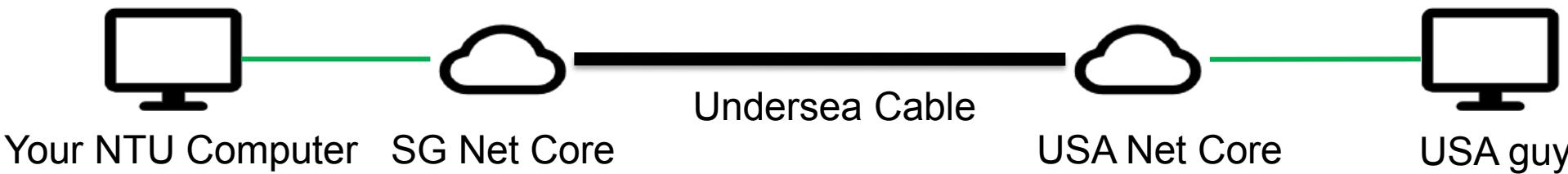
OSI Model			
	Data unit	Layer	Function
Host layers	Data	7. Application	Network process to application
		6. Presentation	Data representation, encryption and decryption, convert machine dependent data to machine independent data
		5. Session	Inter-host communication, managing sessions between applications
	Segment	4. Transport	End-to-end connections, reliability and flow control
Media layers	Packet	3. Network	Path determination and logical addressing
	Frame	2. Data link	Physical addressing
	Bit	1. Physical	Media, signal and binary transmission

OSI in Action: Encapsulation

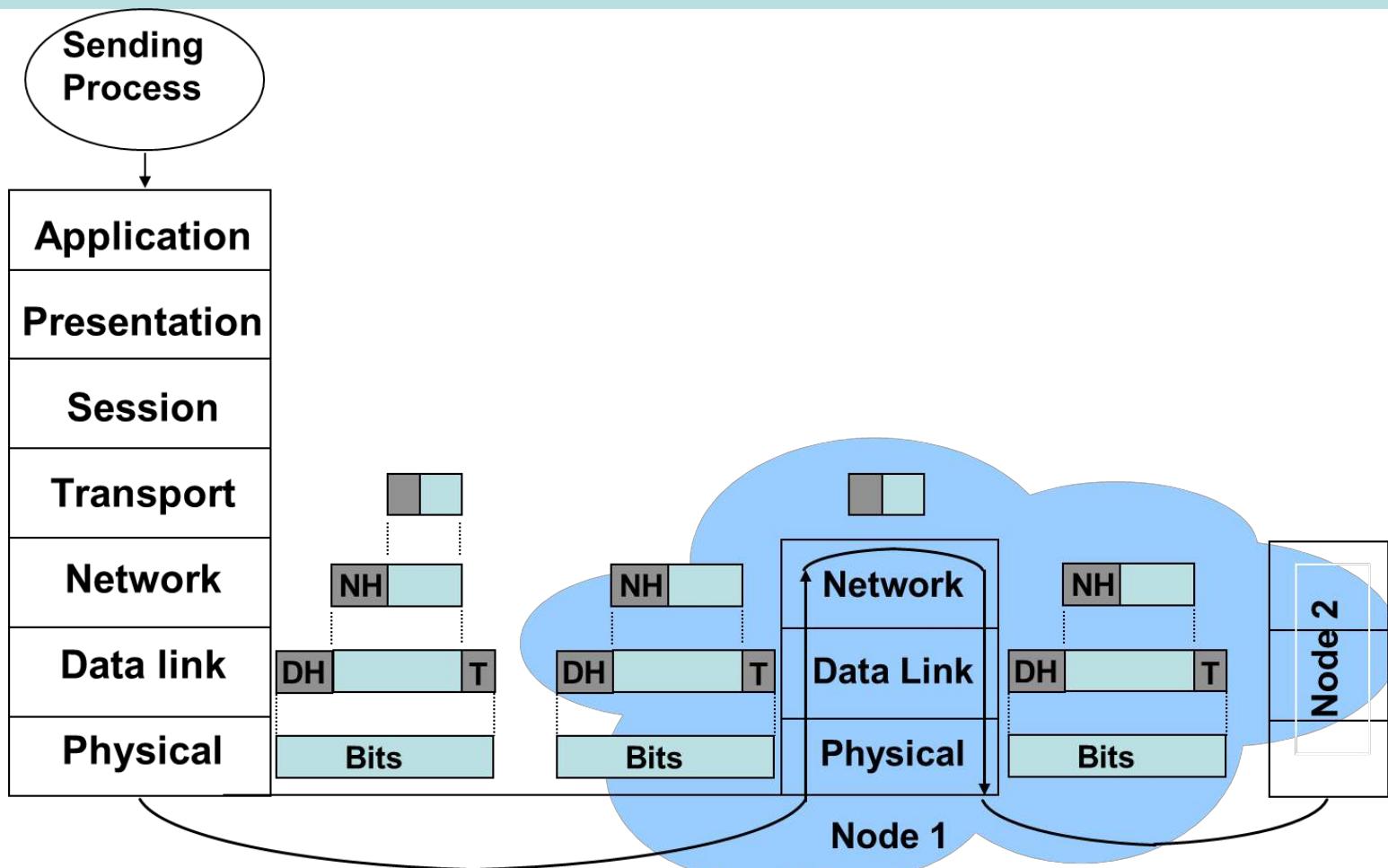
- A message begins at the top application layer and moves down the OSI layers to the bottom physical layer
- As the message descends, each successive OSI model layers adds a header to it
- A header is layer-specific information that basically explains what functions the layer carries out
- Conversely, at the receiving end, headers are stripped from the message as it travels up the OSI layers.



A Simple Computer Network

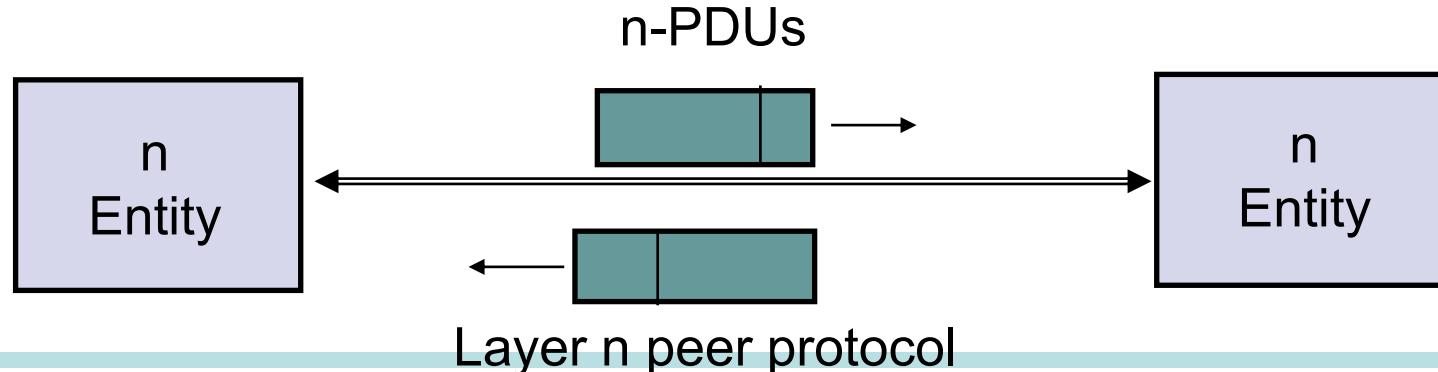


Header Processing at Switch Node



OSI Unified View: Protocols

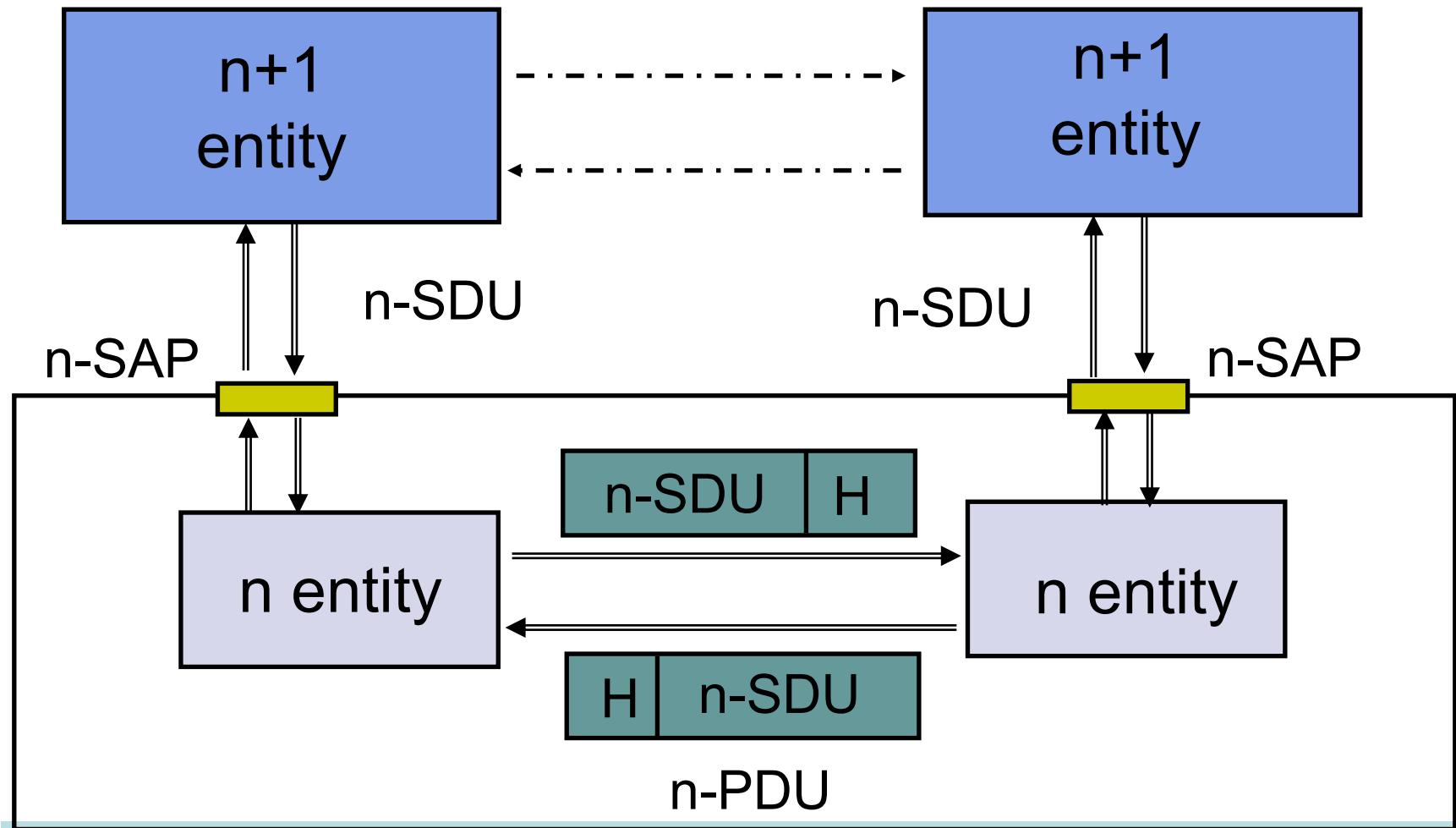
- Layer n in one machine interacts with layer n in another machine to provide a service to layer $n + 1$
- The entities comprising the corresponding layers on different machines are called *peer processes*.
- The machines use a set of rules and conventions called the *layer-n protocol*.
- Layer- n peer processes communicate by exchanging *Protocol Data Units (PDUs)*



OSI Unified View: Services

- Communication between peer processes is virtual and actually indirect
- Layer $n+1$ transfers information by invoking the services provided by layer n
- Services are available at **Service Access Points (SAP's)**
- Each layer passes data & control information to the layer below it until the physical layer is reached and transfer occurs
- The data passed to the layer below is called a **Service Data Unit (SDU)**
- SDU's are *encapsulated* in PDU's

Layers, Services & Protocols



OSI Model in a Nutshell

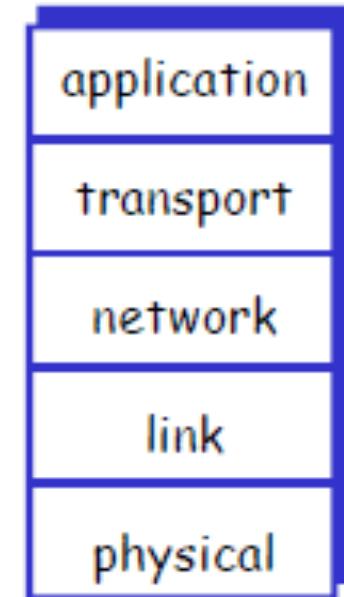
OSI (Open Source Interconnection) 7 Layer Model

Layer	Application/Example	Central Device/Protocols	DOD4 Model
Application (7) Serves as the window for users and application processes to access the network services.	End User layer Program that opens what was sent or creates what is to be sent Resource sharing • Remote file access • Remote printer access • Directory services • Network management	User Applications SMTP	
Presentation (6) Formats the data to be presented to the Application layer. It can be viewed as the "Translator" for the network.	Syntax layer encrypt & decrypt (if needed) Character code translation • Data conversion • Data compression • Data encryption • Character Set Translation	JPEG/ASCII EBDIC/TIFF/GIF PICT	Process
Session (5) Allows session establishment between processes running on different stations.	Synch & send to ports (logical ports) Session establishment, maintenance and termination • Session support - perform security, name recognition, logging, etc.	Logical Ports RPC/SQL/NFS NetBIOS names	
Transport (4) Ensures that messages are delivered error-free, in sequence, and with no losses or duplications.	TCP Host to Host, Flow Control Message segmentation • Message acknowledgement • Message traffic control • Session multiplexing	F P A C K E T R I N G TCP/SPX/UDP	Host to Host
Network (3) Controls the operations of the subnet, deciding which physical path the data takes.	Packets ("letter", contains IP address) Routing • Subnet traffic control • Frame fragmentation • Logical-physical address mapping • Subnet usage accounting	Routers IP/IPX/ICMP	Internet
Data Link (2) Provides error-free transfer of data frames from one node to another over the Physical layer.	Frames ("envelopes", contains MAC address) [NIC card — Switch — NIC card] (end to end) Establishes & terminates the logical link between nodes • Frame traffic control • Frame sequencing • Frame acknowledgment • Frame delimiting • Frame error checking • Media access control	Switch Bridge WAP PPP/SLIP Land Based Layers	Can be used on all layers
Physical (1) Concerned with the transmission and reception of the unstructured raw bit stream over the physical medium.	Physical structure Cables, hubs, etc. Data Encoding • Physical medium attachment • Transmission technique - Baseband or Broadband • Physical medium transmission Bits & Volts	Hub	Network



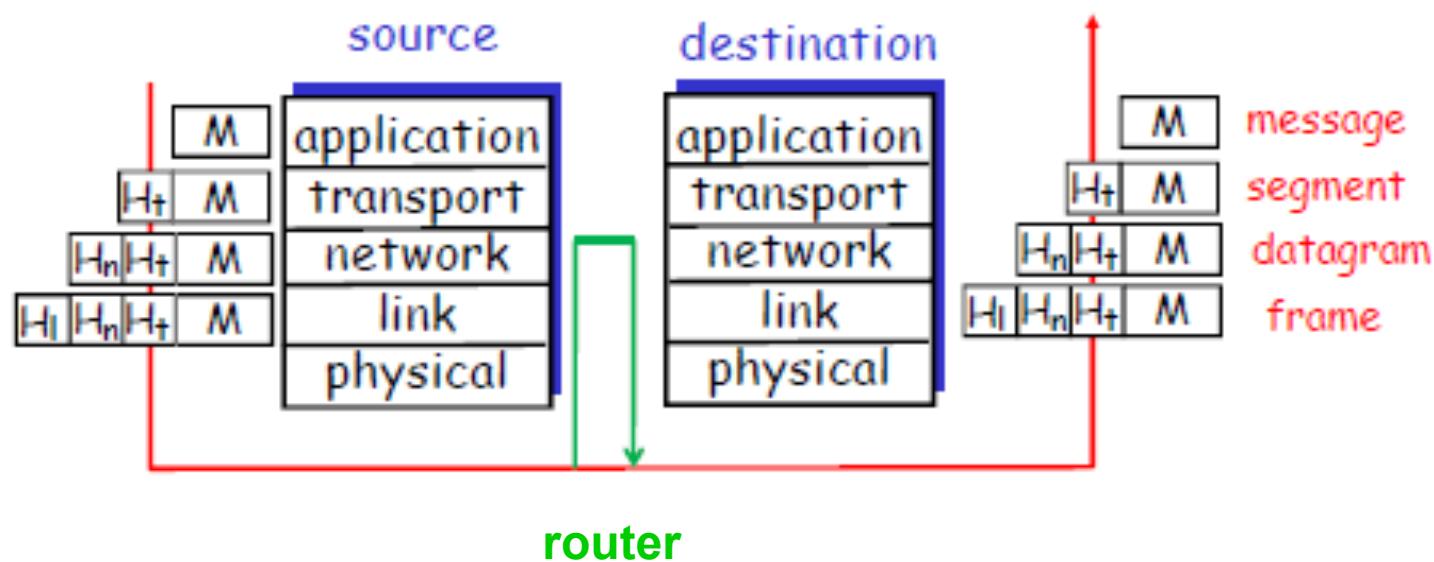
TCP/IP Model: 5 Layers

- **Application:** supporting network applications
 - FTP, SMTP, HTTP
- **Transport:** host-host data transfer
 - **Transmission Control Protocol (TCP)**, UDP
- **Network:** routing of datagrams from source to destination
 - **Internet Protocol (IP)**, routing protocols
- **Link:** data transfer between neighboring network elements
 - PPP, Ethernet
- **Physical:** bits on the wire

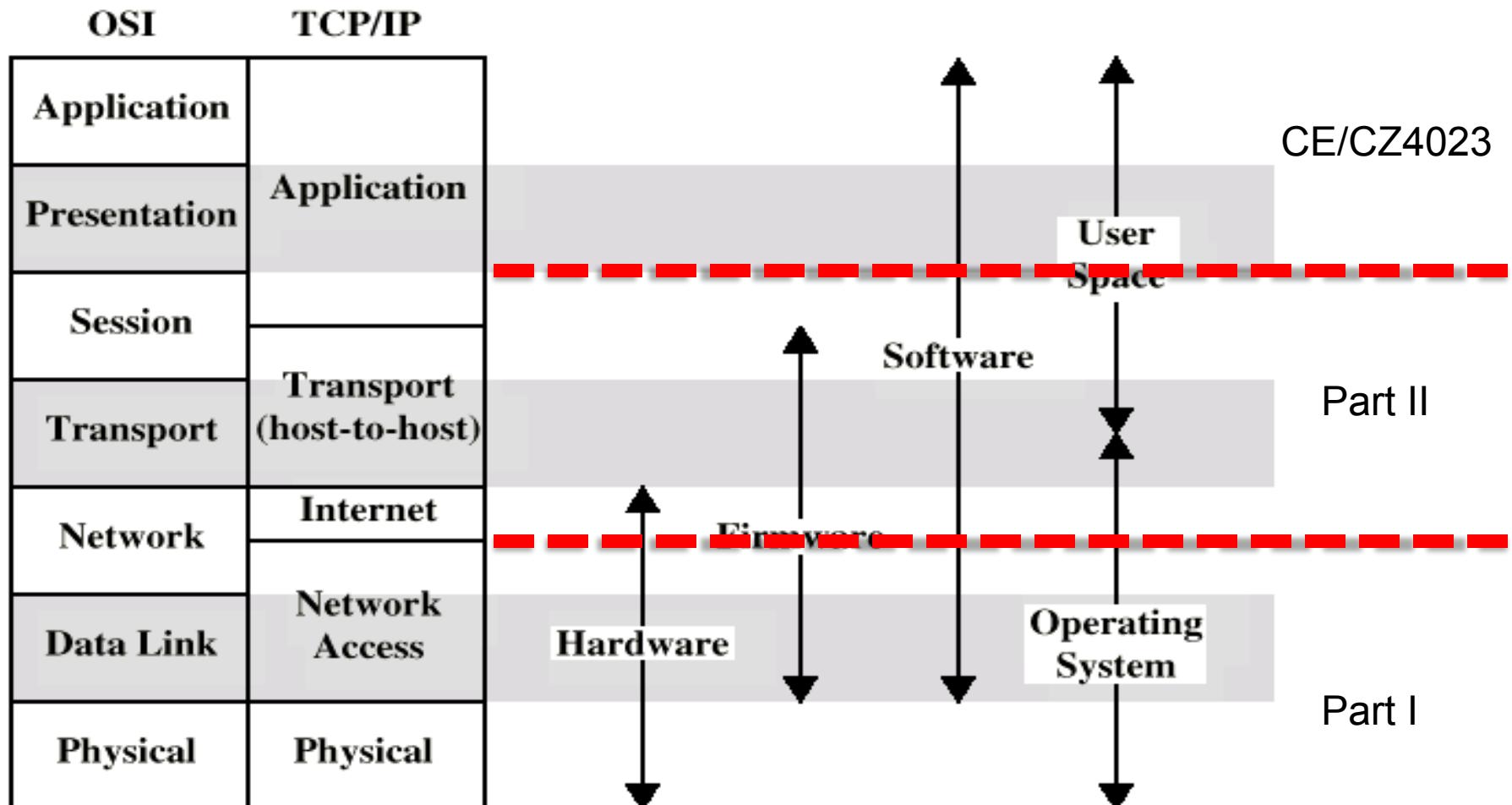


TCP/IP Internetworking

- **Each layer takes data from above**
 - Adds header information to create new data unit
 - Passes new data unit to layer below



TCP/IP vs OSI Models



Part I Syllabus

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 23/Jan/2023 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – MAC	M1-L6-LAN-MAC.pptx
	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Mobile Access Networks: From 1G to 5G	M1-L9-Mobile.pptx
	Network paradigms	M1-L10-Paradigms.pptx

Physical Layer: Network Resilience

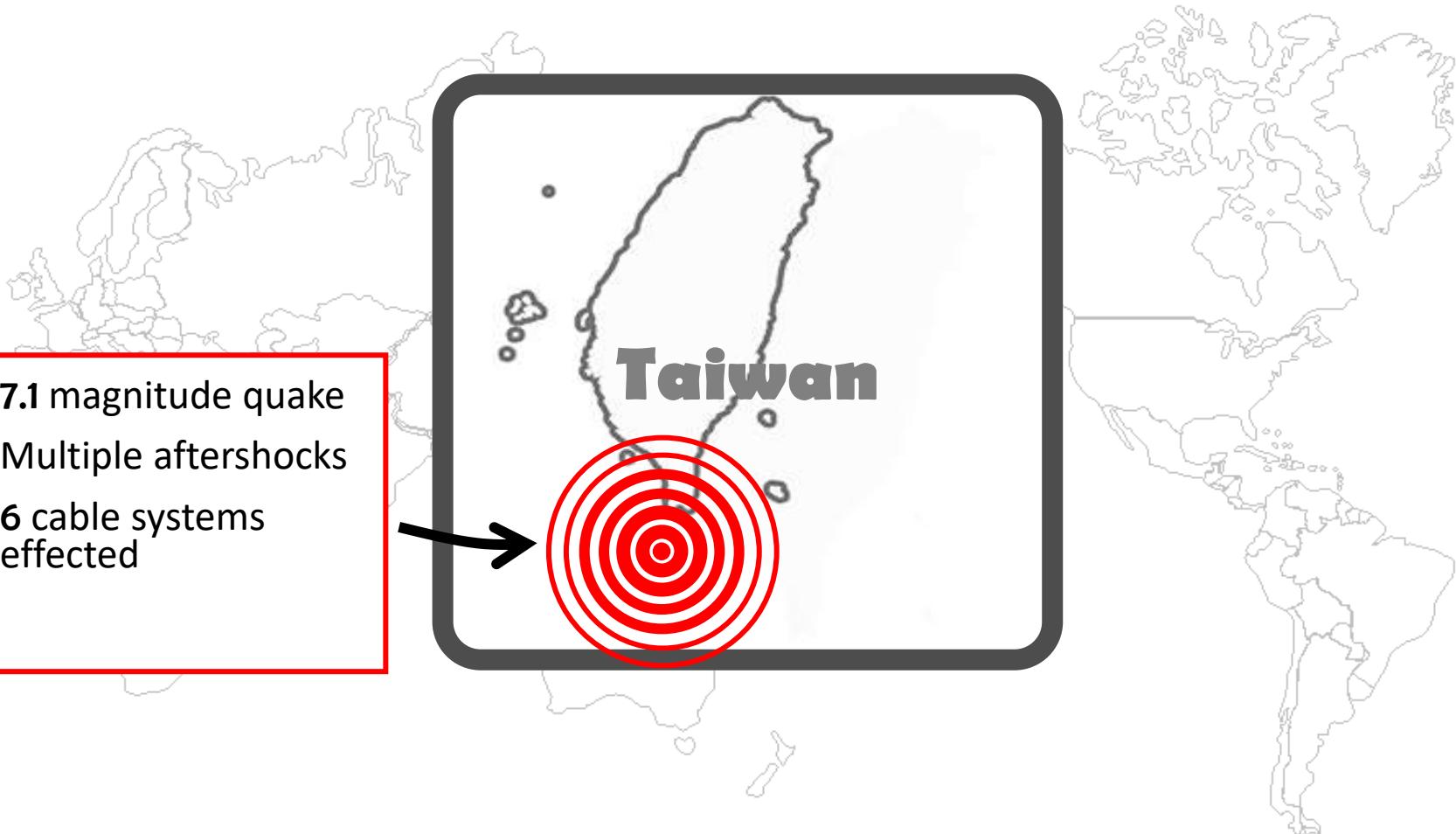
ASIA COVERAGE

Use links for abstraction

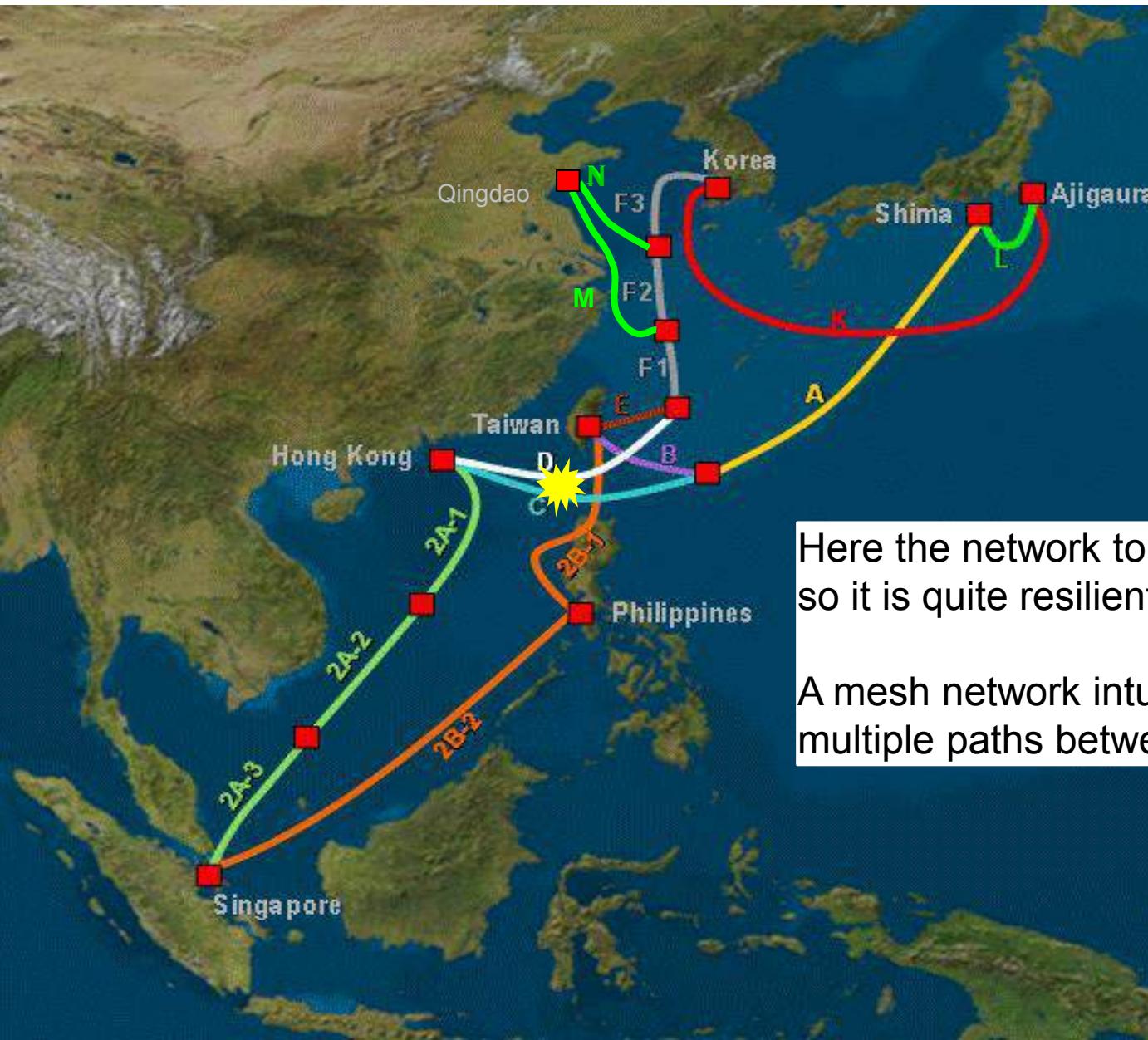


GMT 12:26:01, 26 December 2006

- 7.1 magnitude quake
- Multiple aftershocks
- 6 cable systems effected



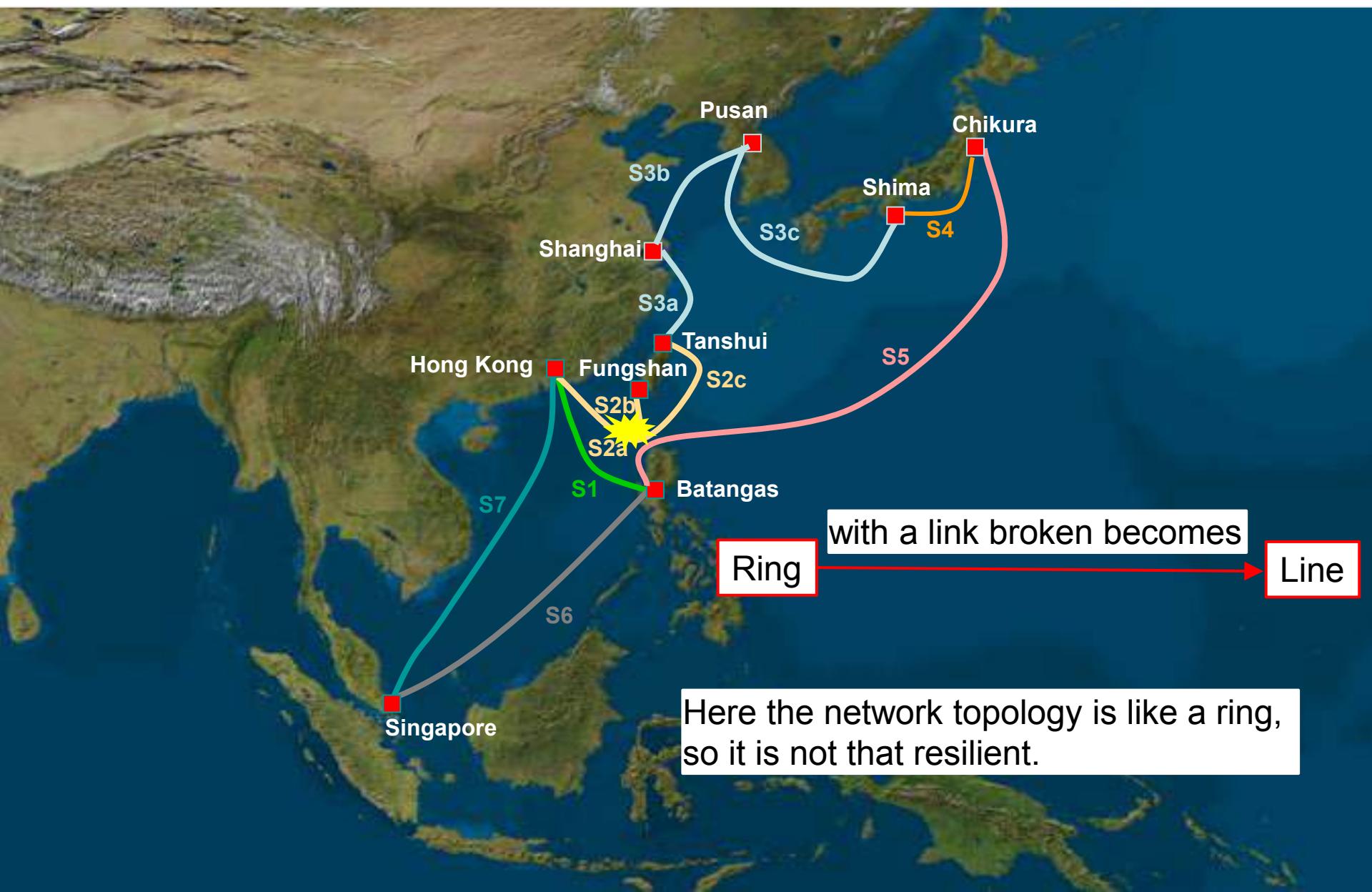
East Asia Crossing (EAC) Cable Network



Here the network topology is like a mesh, so it is quite resilient.

A mesh network intuitively means there are multiple paths between any two nodes.

City-to-City (C2C) Cable Network

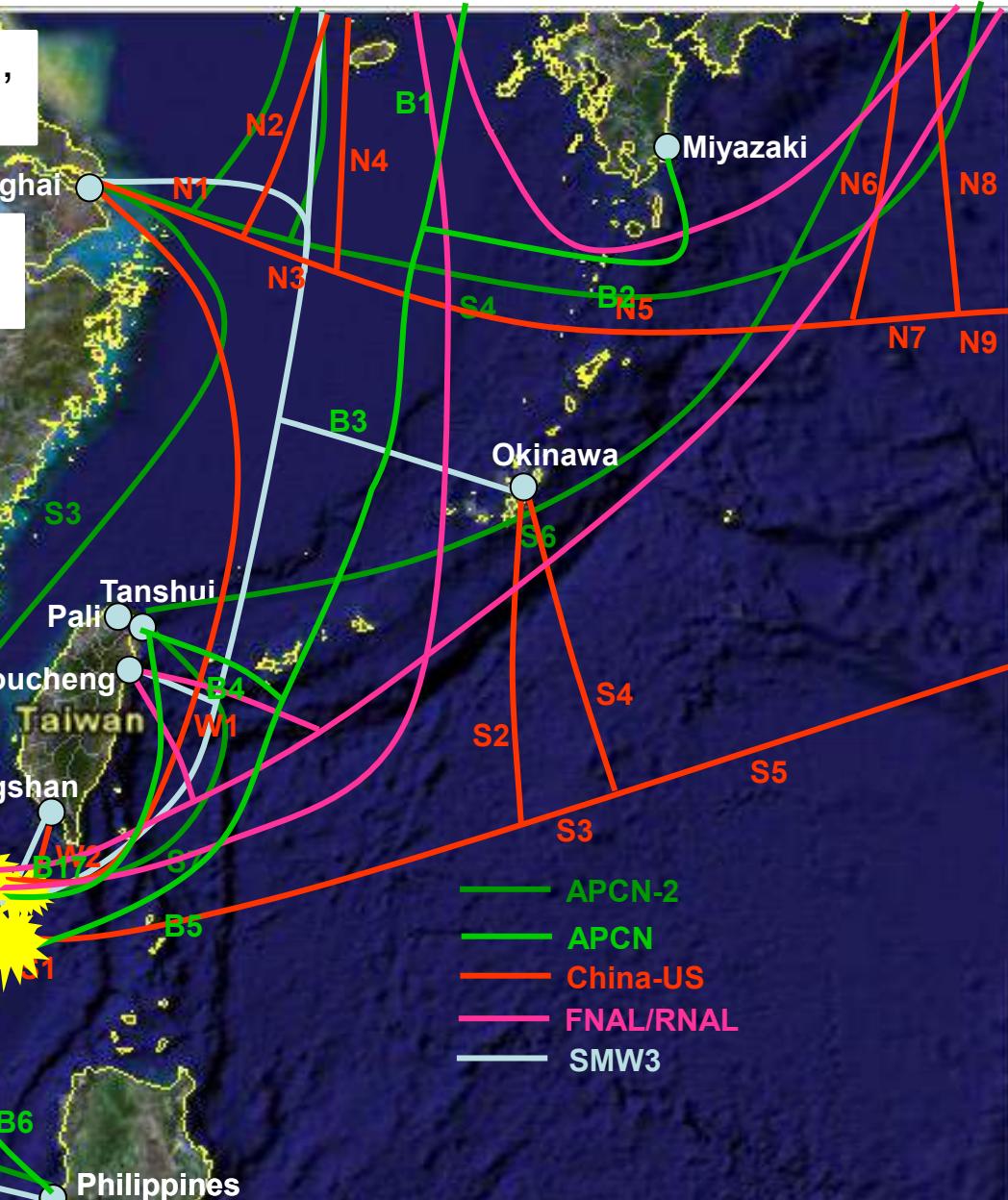


Other Cable Network

Here the network topology is like a ring,
so it is not that resilient.



Here the network topology is like a bus,
so it is not that resilient.



Ring/bus is less resilient than mesh,
but cheaper.



Network Reliability

- Probability that a network performs satisfactorily over a period of time
- Parameters:
 - Mean Time Between Failures (MTBF)
 - Mean Time to Failure (MTTF)
 - Mean Time to Repair (MTTR)

Not appear
in exam

$$\text{MTBF} = \text{MTTF} + \text{MTTR}$$



Link Failure Probability

- **Link Failure Probability**: percentage of time during which the link is dysfunctional
- **Link Availability**: percentage of time during which the link is functional
- b_i : Probability link “i” is broken
- r_i : Probability link “i” is available, i.e., not broken
- $r_i = 1 - b_i$

<i>reverted</i> R Availability %	Downtime per year	Downtime per month*	Downtime per week
90% ("one nine")	36.5 days	72 hours	16.8 hours
95%	18.25 days	36 hours	8.4 hours
97%	10.96 days	21.6 hours	5.04 hours
98%	7.30 days	14.4 hours	3.36 hours
99% ("two nines")	3.65 days	7.20 hours	1.68 hours
99.5%	1.83 days	3.60 hours	50.4 minutes
99.8%	17.52 hours	86.23 minutes	20.16 minutes
99.9% ("three nines")	8.76 hours	43.8 minutes	10.1 minutes
99.95%	4.38 hours	21.56 minutes	5.04 minutes
99.99% ("four nines")	52.56 minutes	4.32 minutes	1.01 minutes
99.999% ("five nines")	5.26 minutes	25.9 seconds	6.05 seconds
99.9999% ("six nines")	31.5 seconds	2.59 seconds	0.605 seconds
99.99999% ("seven nines")	3.15 seconds	0.259 seconds	0.0605 seconds

Definitely in exam

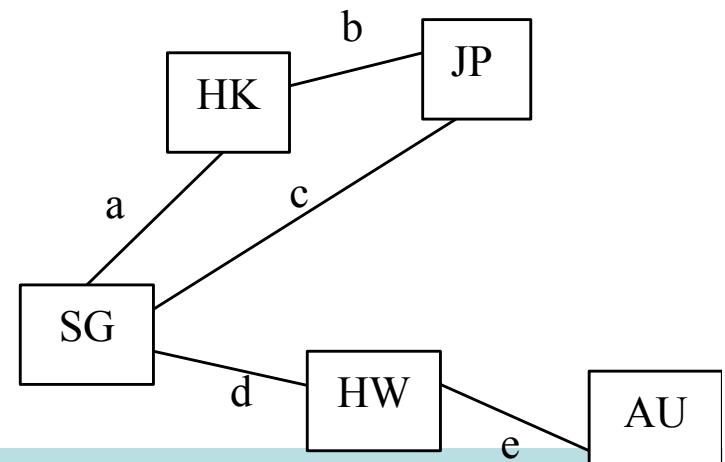
Network Resilience Issues

- What's the probability of a link failure?
- Are there alternative paths?
- Is there a single point of failure ?
- What is the probability for two nodes to stay connected in a network?

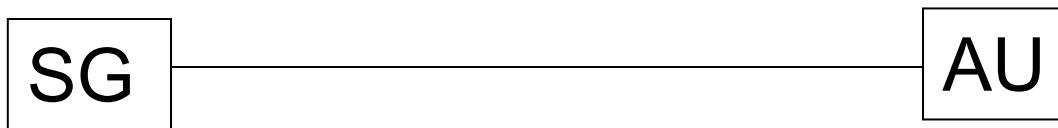


Network Resilience

- A measure of Network Fault Tolerance
- Express in terms of probability that the network remains connected.
- Assumptions
 - The probability of link failures are independent of each other.



Network Availability: Single Link



Given that the failure probability of the link is 0.05,

Failure probability:

$$b_{SG-AU} = 0.05$$

Availability probability:

$$r_{SG-AU} = 1 - 0.05 = 0.95$$

Network Availability: Series



Given that each link has a failure probability of 0.05,

Prob that SG can communicate with AU:

$$r_{SG-HW-AU} = \Pr[\text{both links survive}] = r_{SG-HW} * r_{HW-AU} = (1-0.05)*(1-0.05) \\ = 0.9025$$

! What is the probability that SG cannot communicate with AU?

$$b_{SG-HW-AU} = 1 - 0.9025 = 0.0975$$

Computing $1 - r_{SG-HW} * r_{HW-AU}$ above is easier than summing the three products below:

$$b_{SG-HW} * b_{HW-AU}$$



$$b_{SG-HW} * r_{HW-AU}$$



$$r_{SG-HW} * b_{HW-AU}$$



Network Availability: Parallel



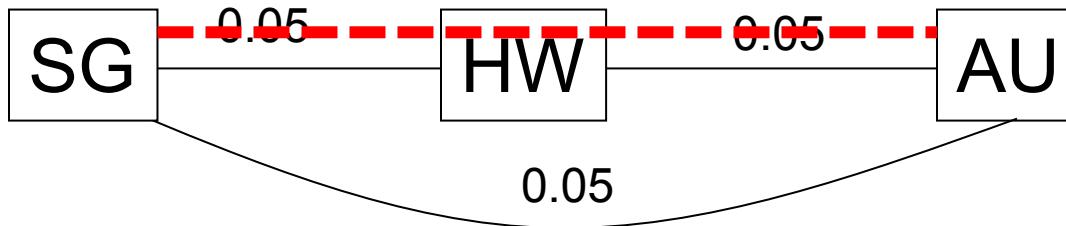
Given that each link has a failure probability of 0.05,
What is the probability that SG is isolated from AU ?

$$\begin{aligned}\text{Prob(break)} &= \Pr[\text{both links break}] \\ &= b_A * b_B \\ &= 0.05 * 0.05 = 0.0025\end{aligned}$$



$$b_{SG-AU} = 0.0025$$

Hybrid Graphs: Path-Based Approach



Given that each link has a failure probability of 0.05,
Calculate the Prob that SG is isolated from AU.

$$\begin{aligned}r_{\text{SG-HW-AU}} &= r_{\text{SG-HW}} * r_{\text{HW-AU}} \\&= (1-0.05)*(1-0.05) \\&= 0.9025\end{aligned}$$

$$\begin{aligned}b_{\text{SG-HW-AU}} &= 1 - r_{\text{SG-HW-AU}} \\&= 1 - 0.9025 \\&= 0.0975\end{aligned}$$

$$\begin{aligned}\text{Prob. SG Disconnected from AU} &= b_{\text{SG-HW-AU}} * b_{\text{SG-AU}} \\&= 0.0975 * 0.05 \\&= 0.004875\end{aligned}$$

! Rules for Network Availability

- **Link in series**
 - Calculate that probability that all links in the series are working
- **Link in parallel**
 - Calculate the probability that all links are broken.
- **Combination of series and parallel**
 - Decompose them into paths
 - Calculate network availability using path-based approach

Learning Objectives

- **Layered Network Architecture**
 - Why layering?
 - OSI model and its functions
 - TCP/IP model and its functions
 - Mapping between OSI and TCP/IP Models
- **Physical Layer Resilience**
 - Definition of link availability and calculation
 - Path-based approach

Supplementary Materials

Physical Layer

- Provides physical interface for transmission of information
 - Defines rules by which bits are passed from one system to another on a physical communication medium
 - Covers all – mechanical, electrical, functional and procedural – aspects for physical communication
 - Such characteristics as voltage levels, timing of voltage changes, physical data rates, maximum transmission distances, physical connectors, and other similar attributes are defined by physical layer specifications
-

Data Link Layer

- Data link layer attempts to provide reliable communication over the physical layer interface
- Breaks the outgoing data into frames and reassemble the received frames
- Create and detect frame boundaries
- Handle errors by implementing an acknowledgement and retransmission scheme
- Implement flow control
- Supports points-to-point as well as broadcast communication
- Supports simplex, half-duplex or full-duplex communication

Network Layer

- **Implements routing of frames (packets) through the network**
 - **Defines the most optimum path the packet should take from the source to the destination**
 - **Defines logical addressing so that any endpoint can be identified**
 - **Handles congestion in the network**
 - **Facilitates interconnection between heterogeneous networks (Internetworking)**
 - **The network layer also defines how to fragment a packet into smaller packets to accommodate different media**
-

Transport Layer

- Purpose of this layer is to provide a reliable mechanism for the exchange of data between two processes in different computers
 - Ensures that the data units are delivered error free
 - Ensures that data units are delivered in sequence
 - Ensures that there is no loss or duplication of data units
 - Provides connectionless or connection oriented service
 - Provides for the connection management
 - Multiplex multiple connection over a single channel
-

Session Layer

- Provides mechanism for controlling the dialogue between the two end systems. It defines how to start, control and end conversations (called sessions) between applications
- Requests for a logical connection to be established on an end-user's request
- Any necessary log-on or password validation is also handled by this layer
- Session layer is also responsible for terminating the connection
- This layer provides services like dialogue discipline which can be full duplex or half duplex
- Session layer can also provide check-pointing mechanism such that if a failure of some sort occurs between checkpoints, all data can be retransmitted from the last checkpoint

Presentation Layer

- Presentation layer defines the format in which the data is to be exchanged between the two communicating entities
- Also handles data compression and data encryption (cryptography)

Application Layer

- Application layer interacts with application programs and is the highest level of OSI model
- Application layer contains management functions to support distributed applications
- Examples of application layer are applications such as file transfer, electronic mail, remote login etc.

Part I Syllabus

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Network paradigms	M1-L9-Paradigms.pptx

Additional Materials

- The related content about flow control talked today in [https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer%20Networking%20A%20Top-Down%20Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer%20Networking%20A%20Top-Down%20Approach.pdf) is as follow:
 - stop-and-wait: p207-p218
 - sliding window: p218-p229
- And here is another resource for your reference <https://www.geeksforgeeks.org/flow-control-in-data-link-layer/>
- You can also find other video materials about
 - stop-and-wait [Stop-and-Wait Protocol - YouTube](#)
 - sliding window [Sliding Window Protocol - YouTube](#)

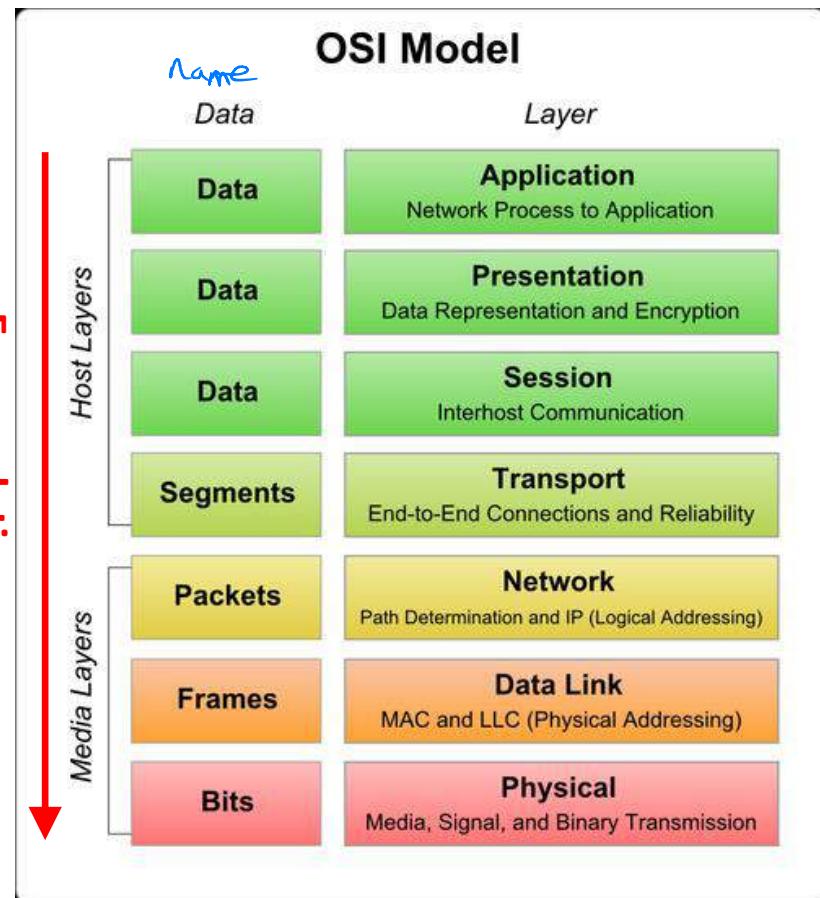
Lecture 3 Data Link Layer (DLL): Flow Control



Contents

- **Data Link Layer Fundamentals**
 - DLL Services
 - Framing mechanisms
 - Link configuration
- **Flow Control in DLL**
 - Main purpose of flow control
 - Stop-and-wait mechanism
 - Sliding window mechanism

information gets bulkier and bulkier
Encapsulation:

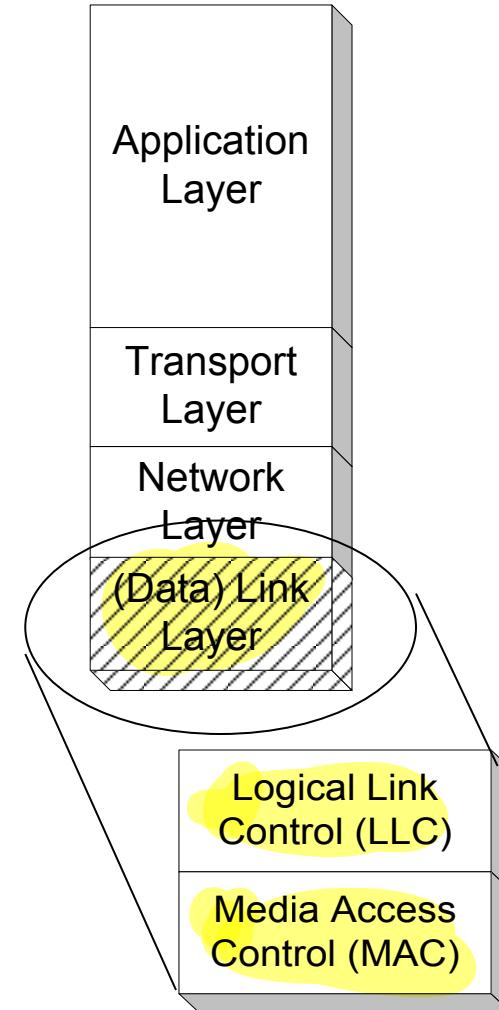


Data Link Layer Fundamentals

! Data Link Layer (DLL): Roles

- **DLL Services**

- **Framing**: encapsulate each network-layer datagram within a link-layer frame before transmission over the link
- **Link Access**: MAC protocol specifying the rules by which a frame is transmitted onto the link
- **Flow Control**: control of data flow to ensure sender not overwhelm the receiver with data
- **Reliable Delivery**: move each network-layer datagram across the link without error



Not examable

Framing

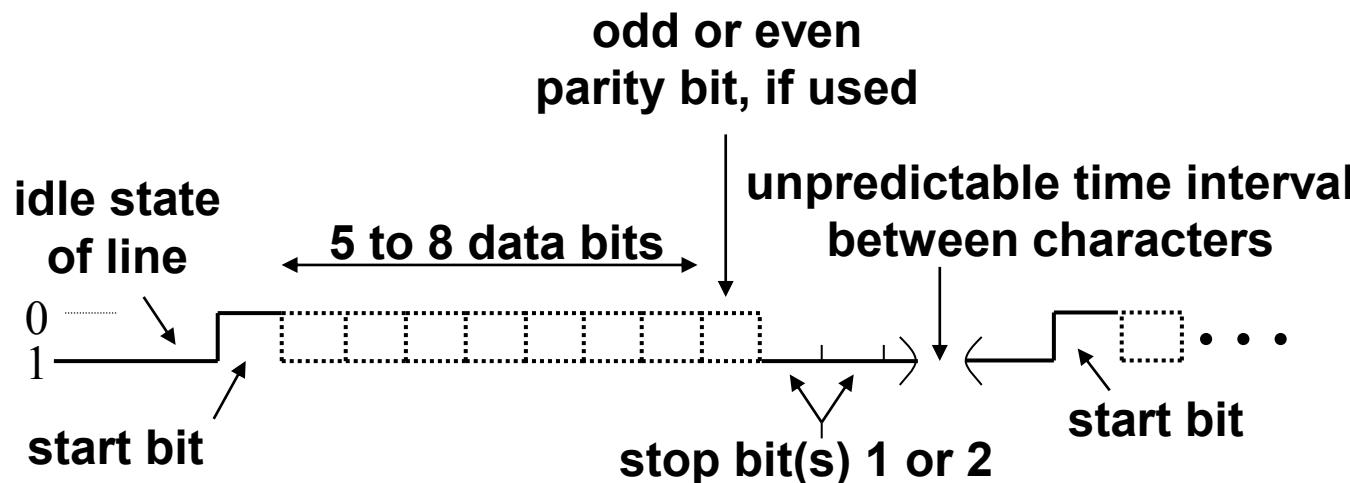
- **Byte Oriented (Character Oriented):**
 - Information is framed into a fixed 8-bit basic unit.
 - Some of these basic units are used for signaling (protocol control).
 - Good solution when digital technology was in its primitive age (late 60s).
- **Bit Oriented (HDLC)**
 - A flag is used to frame the bits sent.
 - Header/Trailer are used to describe the content of a frame. Frames may be used for control.
 - Used by all modern protocols (e.g., HDLC, PPP, Ethernet, etc).

~~Not examinable~~

Byte-Oriented Async. Transmission

- Pre-determined frame format

- Start/stop bit
- Parity check bit
- Data bits

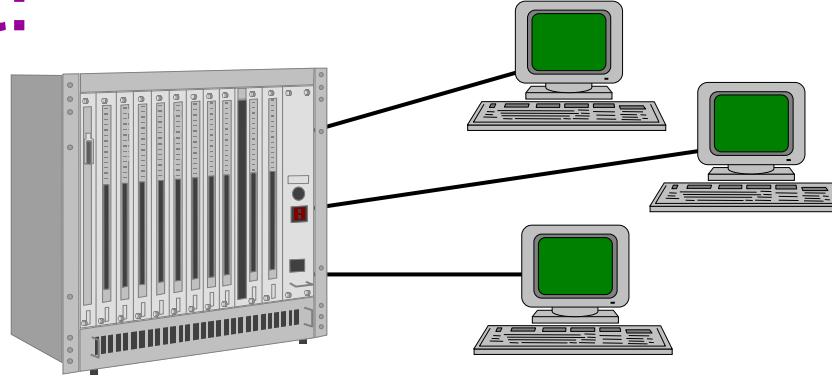


Link Configuration/Access

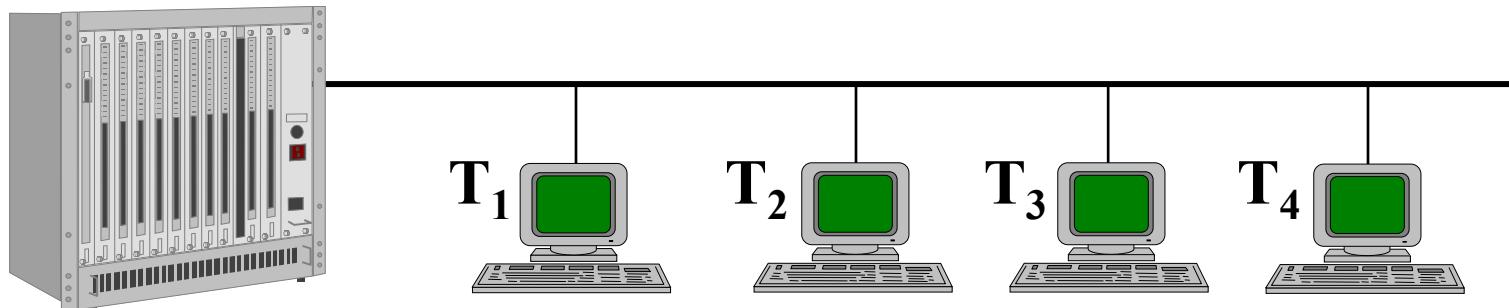
- **Objective:** determine ***who*** gets to transmit at ***when*** on a link
 - **Topology:** physical arrangement of stations
 - Point-to-Point: pairs of hosts are directly connected
 - Broadcast: all stations share a single channel
 - **Duplexity**
 - Half Duplex: Only one party may transmit at a time.
 - Full Duplex: Allows simultaneous transmission and reception between two parties (e.g., two logical half-duplex channels on a single physical channel).
- Mostly used* →

Topology

Point-to-point:



Point-to-Multipoint (Broadcast):



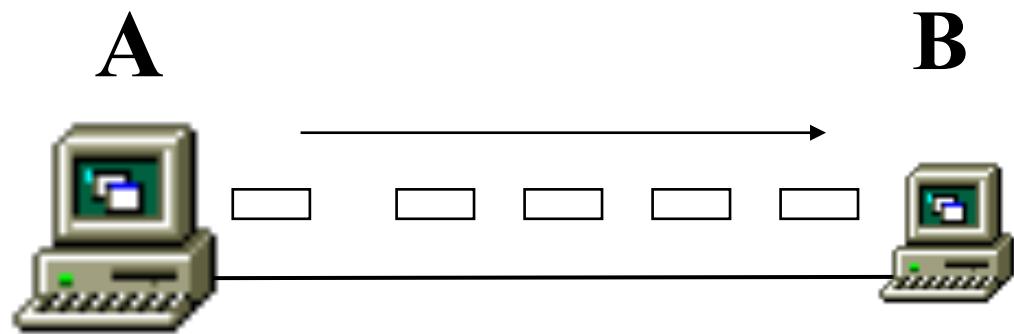
All terminals share the same medium controlled by the primary station (mainframe)

Flow Control

Functions and Mechanisms

- **Flow control**

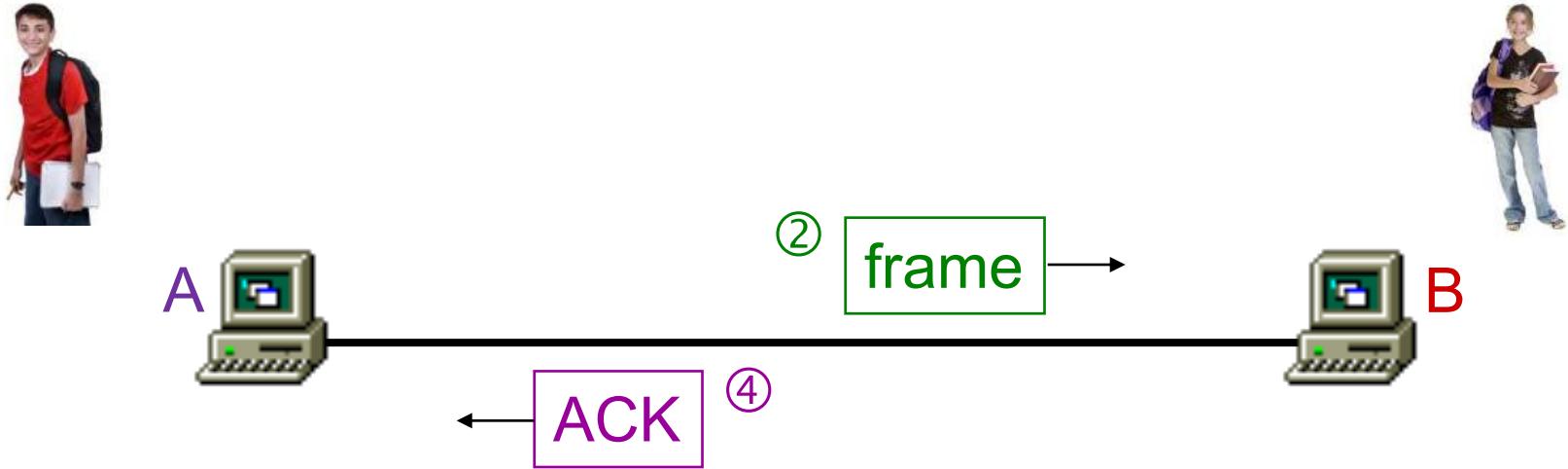
- Ensuring that a transmitting station does not overwhelm a receiving station with data, i.e., buffers at the receiver do not get overflowed.
- No frame error



- **Two Flow-Control Mechanisms**
 - Stop-and-Wait
 - Sliding Window

important

Stop-and-Wait Flow Control



Operations:

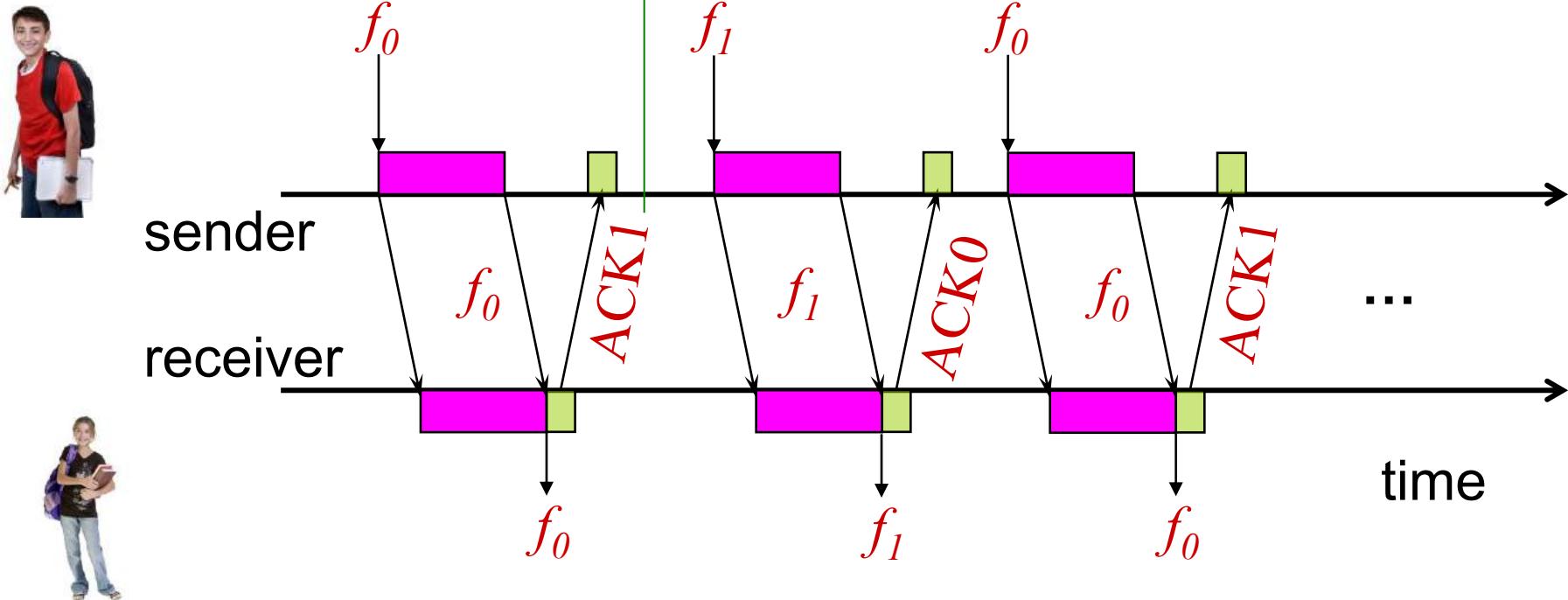
- ① A packs binary information into a frame
- ② A sends the frame to B
- ③ A waits for an ACK
- ④ When B has received the frame, B sends an ACK
- ⑤ When A has received the ACK, A repeats ①

important

Frame Flow in Stop-and-Wait

ACK1 means the receiver expects f_1 , implying f_0 is received successfully

Binary sequence number is sufficient



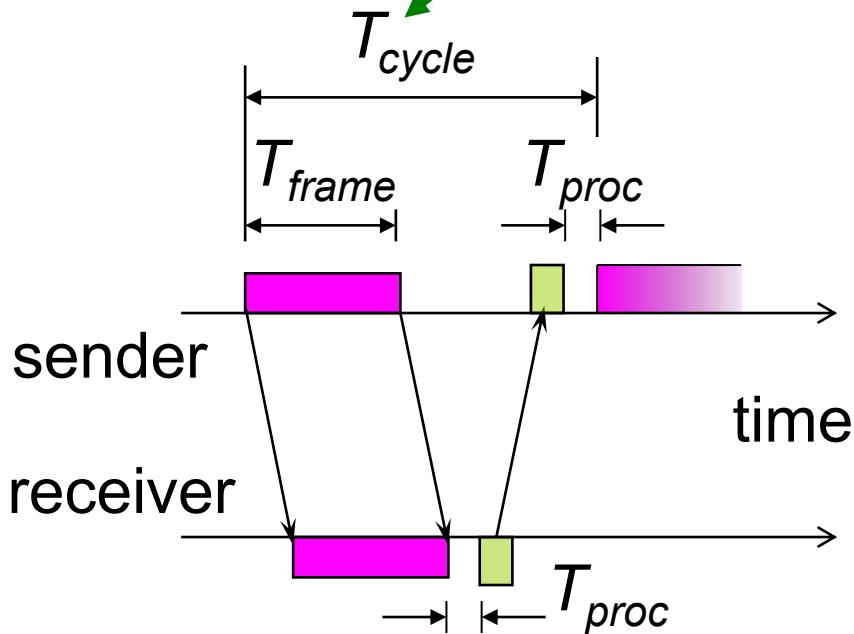
important (all calculation related)

Flow-Control Link Utilization

U
(Link Utilization)

the time that the link
= carries useful information / the total time = $\frac{T_{frame}}{T_{cycle}}$

$$T_{cycle} = T_{frame} + T_{prop} + T_{proc} + T_{ack} + T_{prop} + T_{proc}$$



- T_{cycle} : Total time needed to send a frame
- T_{proc} : Processing time
- T_{frame} : Frame transmission time (i.e., to pump out a frame's bits from your computer)
- T_{ack} : ACK transmission time
- T_{prop} : Signal Propagation delay

important (all calculation related)

Link Utilization for Stop-and-Wait

• Assumptions

- Input is saturated (assume message nonstop)
- No error
- Ignoring T_{ack} & T_{proc}

We get:

$$T_{cycle} = T_{frame} + 2 T_{prop}$$

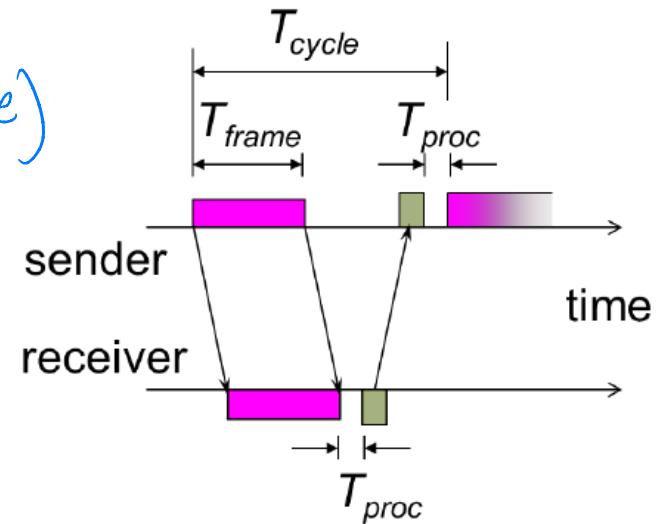
Then:

$$\begin{aligned} U &= T_{frame} / (T_{frame} + 2 T_{prop}) \\ &= 1 / (1+2a) \end{aligned}$$

where:

$$\text{we define } a = T_{prop} / T_{frame}$$

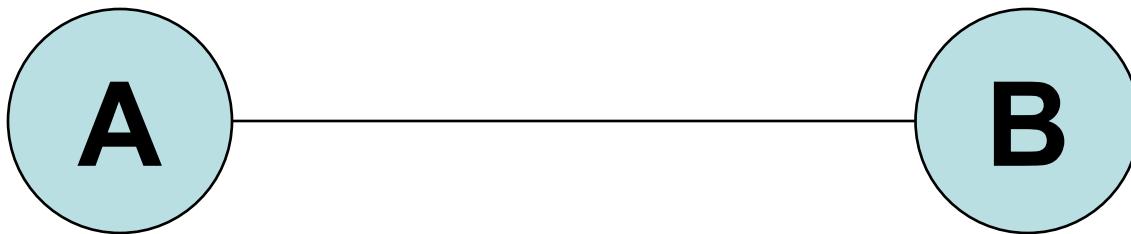
Parameter ‘a’ is called Normalized Propagation Delay



$$U = \frac{1}{1+2a}$$

Example

A communication link exists between two nodes A and B. The transmission rate on the link is 2.4 Mbps. The distance between A and B is 50 km and the signal velocity is 2×10^8 m/s. The frame length is 300 bytes. No frame error. Calculate the link utilization for the stop-&-wait flow control mechanism.



Here T_p and T_f are short for T_{prop} and T_{frame} , respectively.

$$R = 2.4 \text{ Mbps}, L = 300 \text{ bytes} = 2400 \text{ bits}$$

$$D = 50 \text{ km}, v = 2 \times 10^8 \text{ m/s}$$

$$U = 1/(1+2a) \longrightarrow a = T_p/T_f \longrightarrow T_p = D/V = 5 \times 10^4 / 2 \times 10^8 = 250 \mu\text{s}$$

$$U = 1/(1+2*0.25) \leftarrow a = 0.25 \leftarrow T_f = L/R = 2400 / 2.4 \times 10^6 = 1000 \mu\text{s}$$
$$= 2/3$$



Stop-and-Wait: Disadvantages

- If frame or ACK is lost, long waiting time is expected
 - To fix this, use a **TIMEOUT** control in the sender
- If the normalized propagation delay is long, the sender must wait a long time before it can perform the next transmission.
 - The link utilization $U = \frac{1}{1+2a}$ is low.
 - To fix this, use **Buffers** at the sender/receiver (sliding window operation). This will improve the numerator. Note that the denominator $1+2a$ cannot be improved.

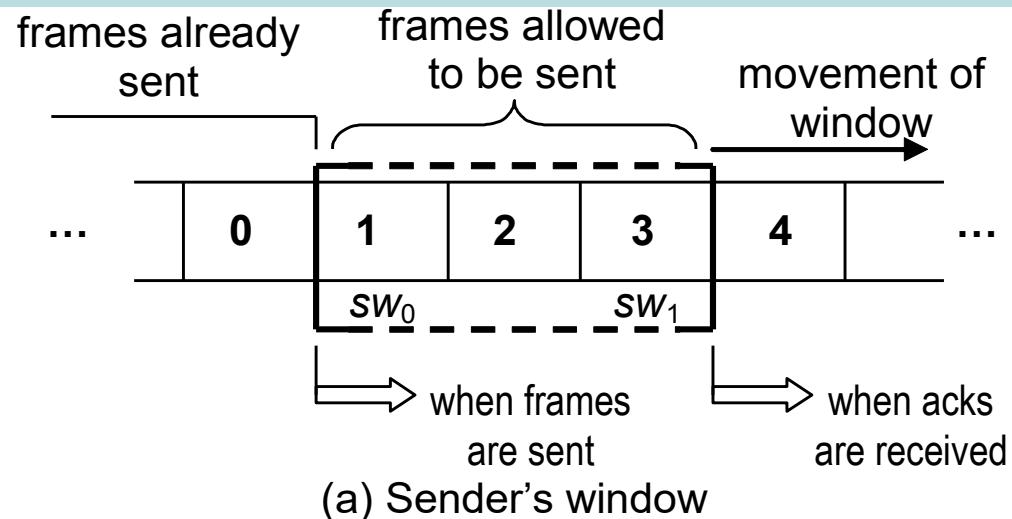
Sliding Window Flow Control

- Allows multiple frames to be in transit.
- Sender and Receiver have buffer N long.
- Sender can send up to N frames without receiving ACKs.
- Each frame is numbered.
- ACK includes number of next expected frame.
- Sequence number bounded by field size (k bits)
 - Frames are numbered modulo 2^k
 - Sequence number $[0, 2^k - 1]$

Sliding Window Operations

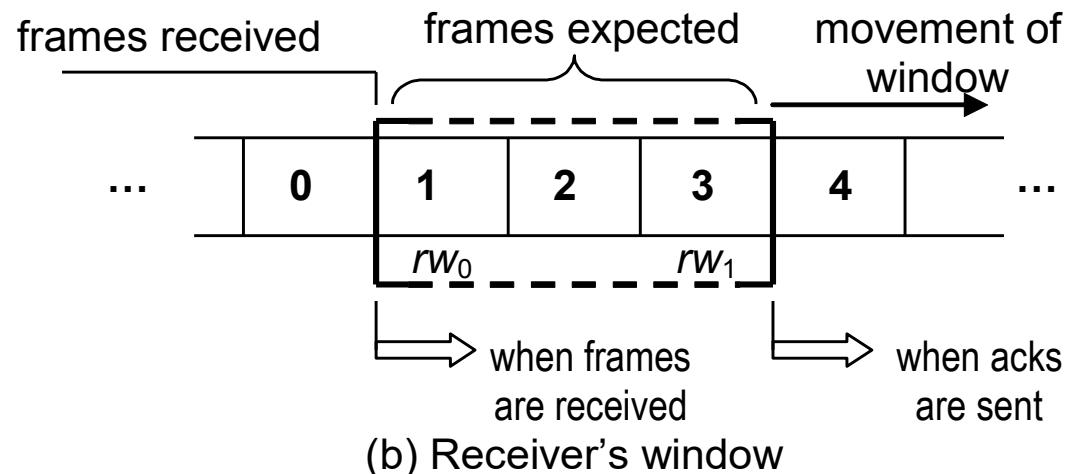
• Sender

- Move lower bound when frames sent
- Move upper bound when acks received



• Receiver

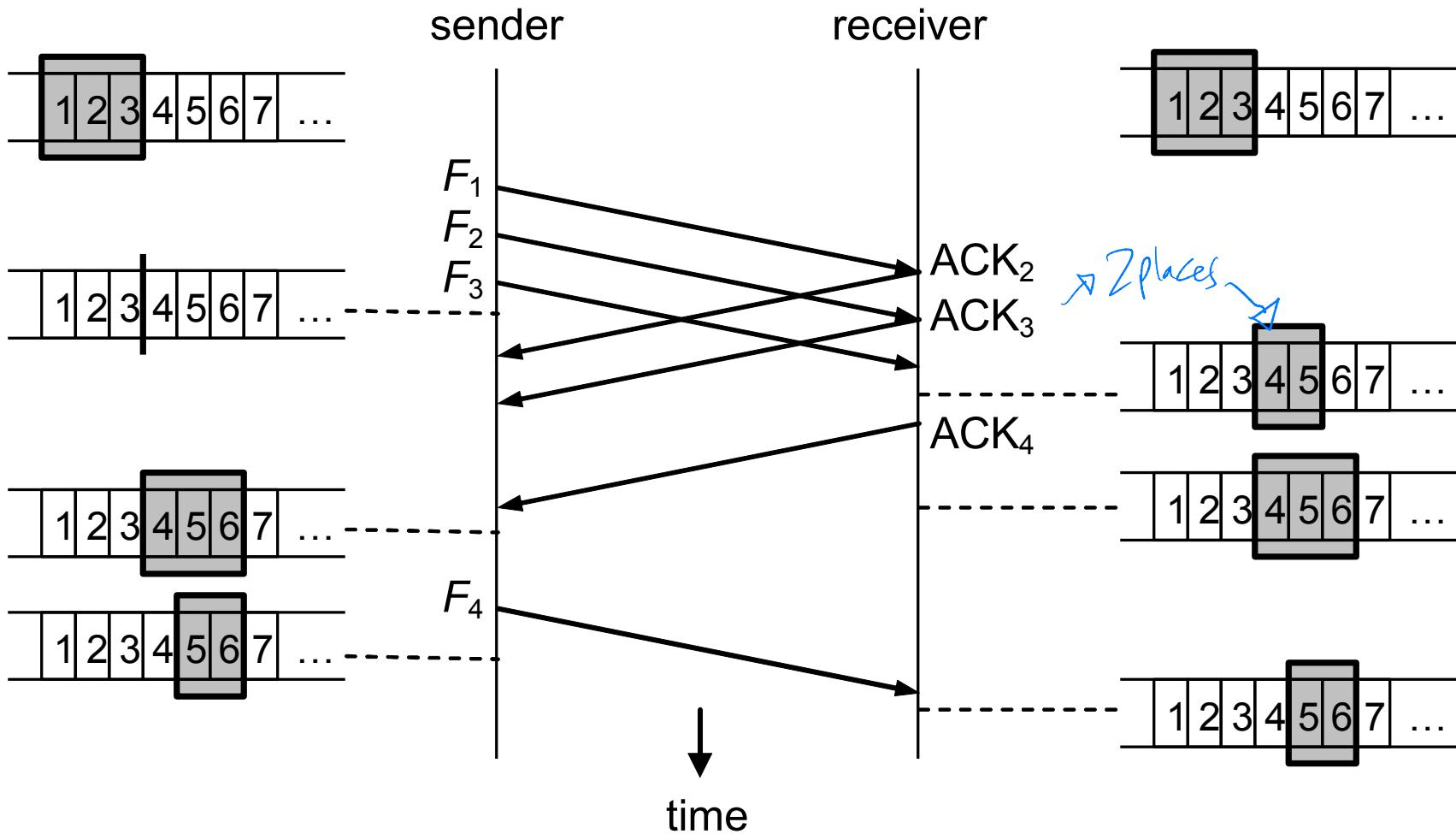
- Move lower bound when frames received
- Move upper bound when acks sent



Sliding Window Operations

- Sender maintains a window, containing frame numbers that can be transmitted.
- Sender window shrinks from trailing edge (left side) as frames are sent.
- Receiver maintains a window as well, its window shrinks from trailing edge as frames are received.
- Receiver's window expands from the leading edge (right side) as ACKs are sent.
- Sender's window expands from the leading edge as ACKs are received.

Sliding Window: Example



Sliding Window Algorithm

```
/* Protocol 4 (sliding window) is bidirectional. */

#define MAX_SEQ 1           /* must be 1 for protocol 4 */
typedef enum {frame_arrival, cksum_err, timeout} event_type;
#include "protocol.h"

void protocol4 (void)
{
    seq_nr next_frame_to_send;      /* 0 or 1 only */
    seq_nr frame_expected;         /* 0 or 1 only */
    frame r, s;                   /* scratch variables */
    packet buffer;                /* current packet being sent */

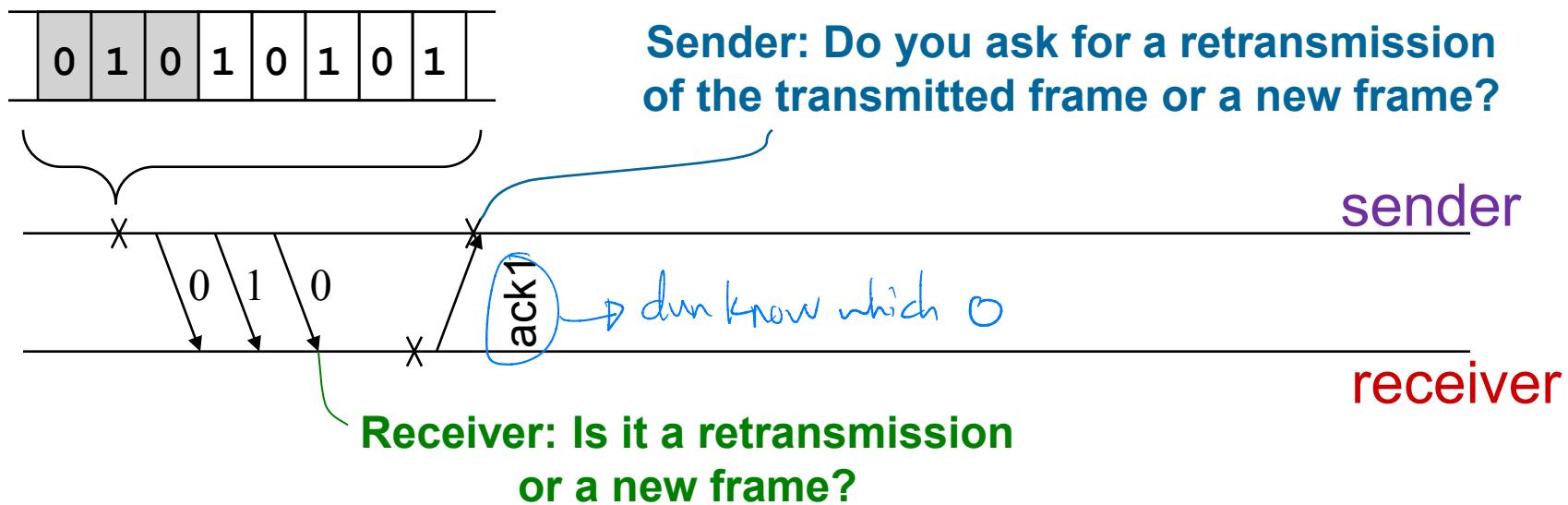
    event_type event;

    next_frame_to_send = 0;          /* next frame on the outbound stream */
    frame_expected = 0;             /* frame expected next */
    from_network_layer(&buffer);   /* fetch a packet from the network layer */
    s.info = buffer;                /* prepare to send the initial frame */
    s.seq = next_frame_to_send;     /* insert sequence number into frame */
    s.ack = 1 - frame_expected;    /* piggybacked ack */
    to_physical_layer(&s);        /* transmit the frame */
    start_timer(s.seq);            /* start the timer running */

    while (true) {
        wait_for_event(&event);      /* frame_arrival, cksum_err, or timeout */
        if (event == frame_arrival) { /* a frame has arrived undamaged. */
            from_physical_layer(&r); /* go get it */
            if (r.seq == frame_expected) { /* handle inbound frame stream. */
                to_network_layer(&r.info); /* pass packet to network layer */
                inc(frame_expected);      /* invert seq number expected next */
            }
            if (r.ack == next_frame_to_send) { /* handle outbound frame stream. */
                stop_timer(r.ack);        /* turn the timer off */
                from_network_layer(&buffer); /* fetch new pkt from network layer */
                inc(next_frame_to_send); /* invert sender's sequence number */
            }
        }
        s.info = buffer;              /* construct outbound frame */
        s.seq = next_frame_to_send;   /* insert sequence number into it */
        s.ack = 1 - frame_expected;  /* seq number of last received frame */
        to_physical_layer(&s);      /* transmit a frame */
        start_timer(s.seq);          /* start the timer running */
    }
}
```

Window Size Consideration

Say, window size, $N = 3$
with $k=1$ bit sequence number



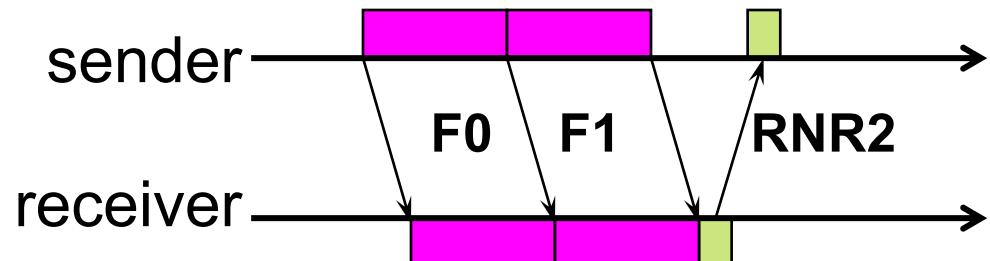
- ① Is the second **0** a new frame or the retransmitted frame?
- ② Which frame is to be transmitted next after receiving ack1?

- Based on the previous slide,
for window size N and k bits sequence number,
we need

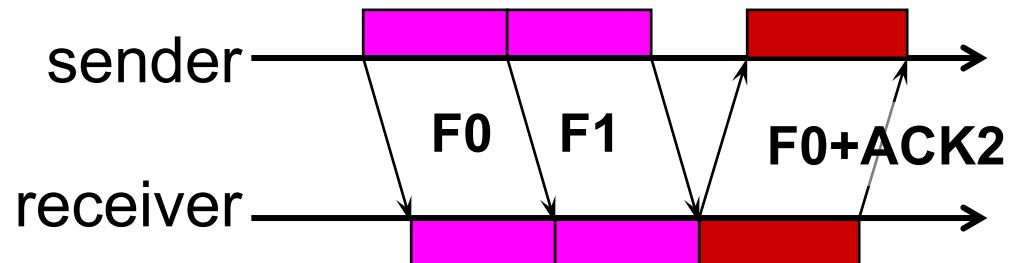
$$N \leq 2^k$$

Sliding Window: Other Features

- Receiver can acknowledge frames without permitting further transmission (by sending ‘Receive Not Ready’, RNR frame). Receiver must send a normal acknowledgement to resume.



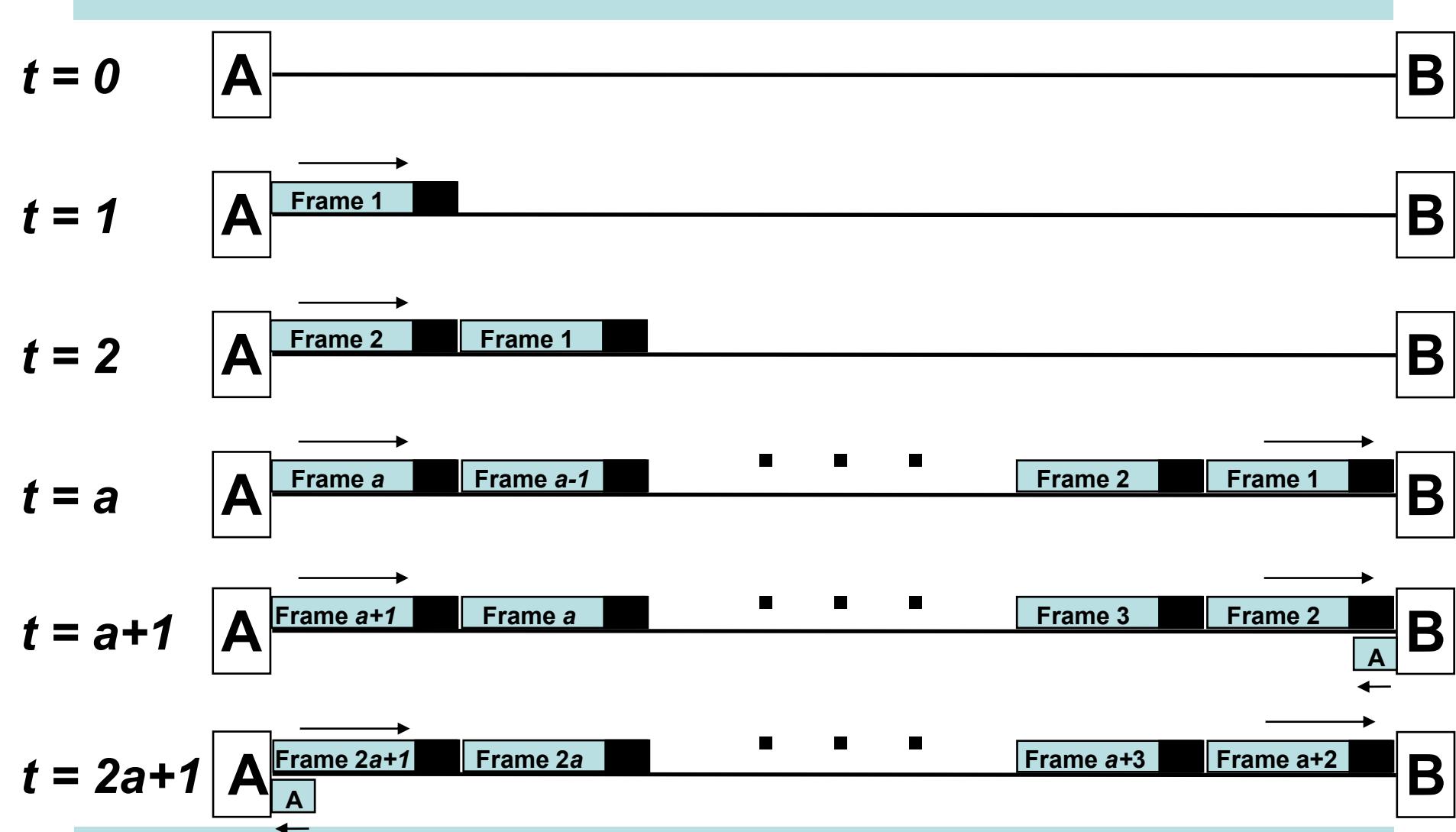
- ACK can be *piggybacked* on the data frames in the reverse direction.



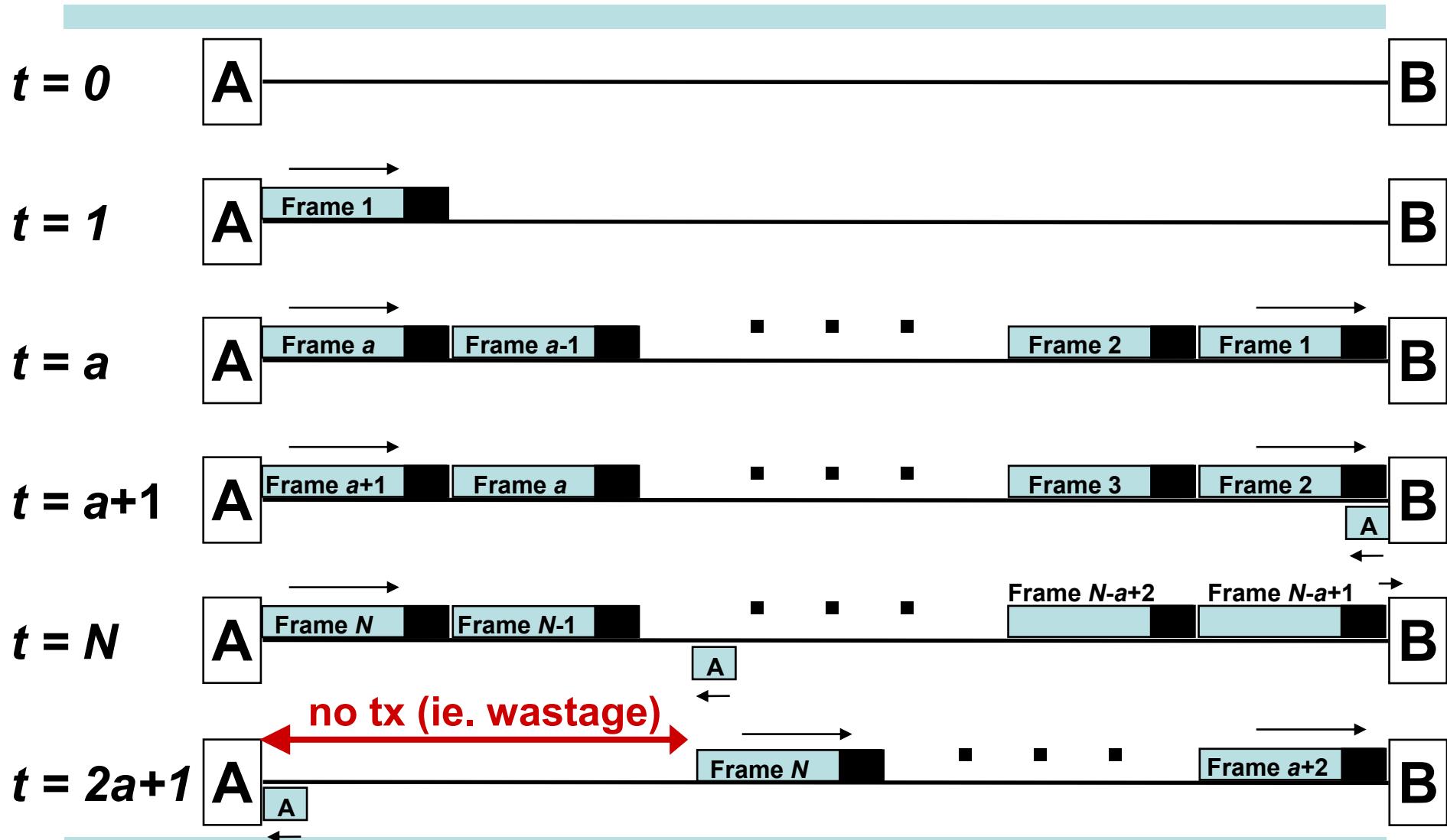
Sliding Window: Performance

- Performance depends upon (assume error-free operation):
 - Parameter a , and
 - Window size, N .
- Assumption: T_{ack} and T_{proc} are negligible.
- Frame transmission time = 1 (normalized to itself)
- Normalized propagation delay (one-way) = a
- We need to consider two cases:
 - $N \geq 2a + 1$: Station can transmit continuously without exhausting its window $\rightarrow U = 1.0$
 - $N < 2a + 1$: Station's window is exhausted at $t = N$, and the station cannot send additional frames until $t = 2a + 1$, $\rightarrow U = N/(1 + 2a)$

Case I: $N \geq 2a + 1$ [U=1]



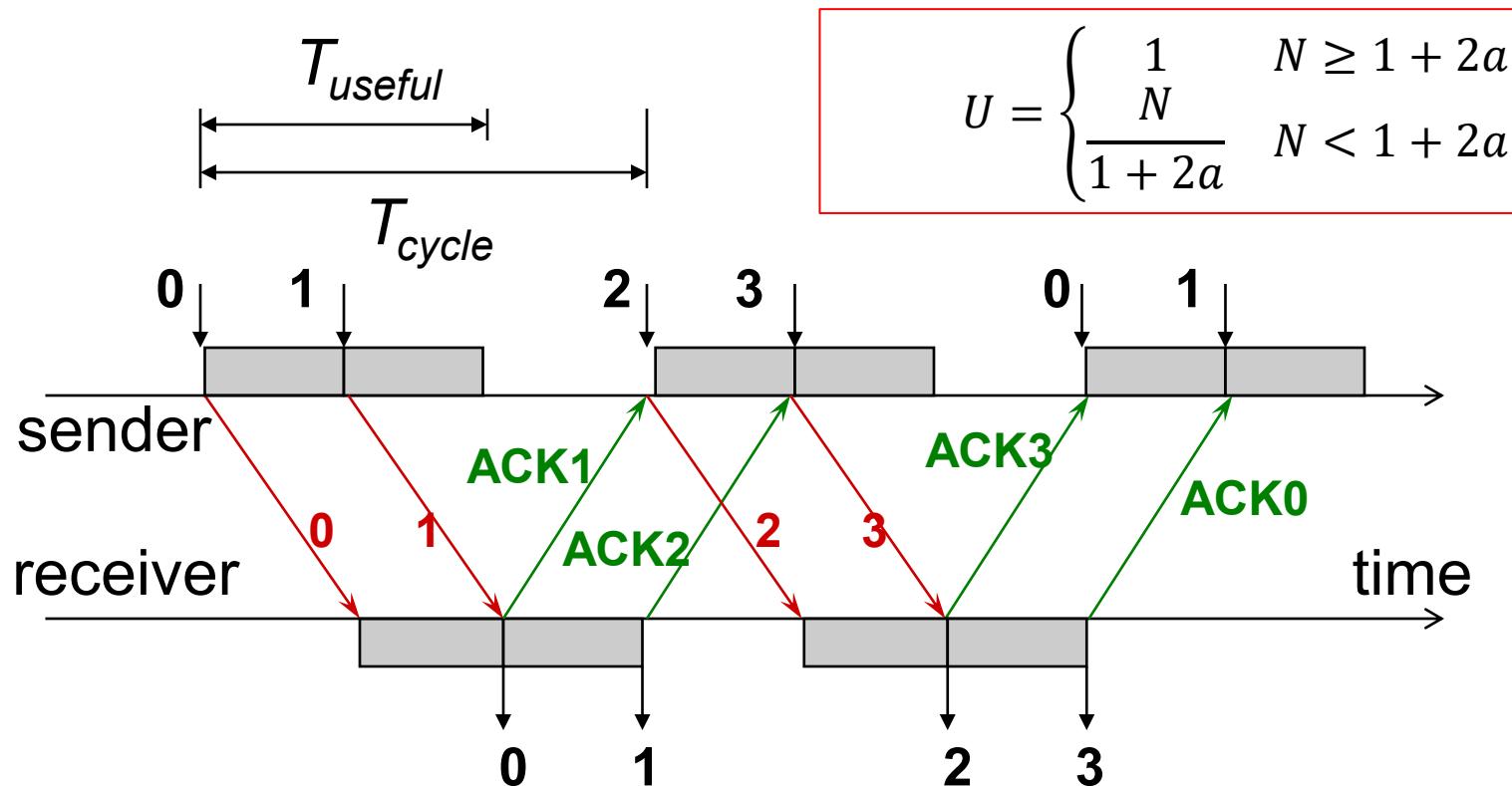
Case II: $N < 2a + 1$ [$U=N/(1+2a)$]



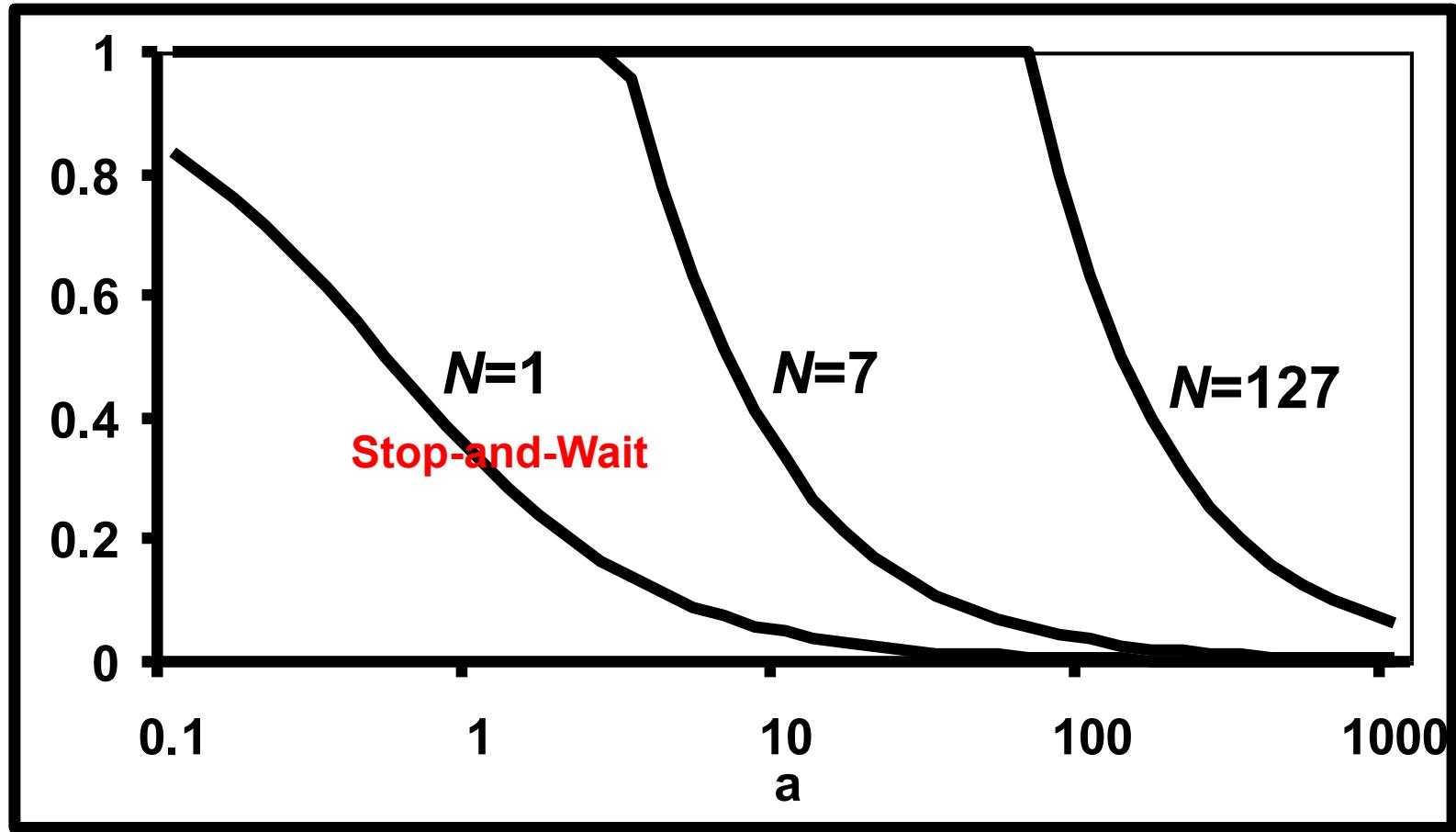
Sliding Window: Performance

Window Size = N

$$T_{useful} = N * T_{frame}$$
$$T_{cycle} = T_{frame} + 2 * T_{prop}$$



Flow Control: Link Utilization



Link Utilization versus a

Learning Objectives

- **Data Link Layer Fundamentals**
 - To understand its (four) main functions
- **Flow Control**
 - To understand its main purpose
 - Stop-and-Wait Flow-Control Mechanism
 - Operational protocol
 - Link utilization **calculation**
 - Sliding Window Flow-Control Mechanism
 - Operational protocol
 - Window size **determination**
 - Link utilization **calculation** (two cases)

Learning Objectives

- **Data Link Layer Fundamentals**
 - To understand its (four) main functions
- **Flow Control**
 - To understand its main purpose
 - Stop-and-Wait Flow-Control Mechanism
 - Operational protocol
 - Link utilization **calculation**
 - Sliding Window Flow-Control Mechanism
 - Operational protocol
 - Window size **determination**
 - Link utilization **calculation** (two cases)

Learning Objectives

- **Data Link Layer Fundamentals**
 - To understand its (four) main functions
- **Flow Control**
 - To understand its main purpose
 - Stop-and-Wait Flow-Control Mechanism
 - Operational protocol
 - Link utilization **calculation**
 - Sliding Window Flow-Control Mechanism
 - Operational protocol
 - Window size **determination**
 - Link utilization **calculation** (two cases)

Part I Syllabus

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Network paradigms	M1-L9-Paradigms.pptx

Additional Materials

- The related content talked today in [https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer Networking A Top-Down Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:
 - Error detection: P438 - P445
 - Automatic Repeat Request (ARQ): P207 – P230
- You can also find other video materials about
 - Error detection [Error Detection - YouTube](#)
 - Stop-and-Wait ARQ [Stop-and-Wait ARQ Protocol - YouTube](#)
 - Go-Back-N ARQ [Go-Back-N ARQ - YouTube](#)
 - Selective Reject ARQ [Selective Repeat ARQ - YouTube](#)

Chat over Unreliable Network

A screenshot of a mobile phone interface. At the top, there are two buttons: "Call" on the left and "Contact Info" on the right. Below them is the date and time: "2013-1-30 0:35". The main area shows a conversation between two users. The first user sends a message: "Do you want to go out for a movie tonight?". The second user replies three times with "Sure". The first user then sends a message: "Shoot!". The second user replies with "Wrong message!". The phone's status bar at the bottom shows "GAOXIAGO.COM".

Call

Contact Info

2013-1-30 0:35

Do you want to go out for a movie tonight?

Sure

Sure

Sure

Shoot!

Wrong message!

GAOXIAGO.COM



CE3005/CPE302 Computer Networks

Lecture 4 Data Link Layer (DLL): Error Control



Contents

- **Error Detection**
 - Parity Check
 - CRC
- **Automatic Repeat Request (ARQ)**
 - Stop-and-Wait ARQ
 - Go-Back-N ARQ
 - Selective Reject ARQ

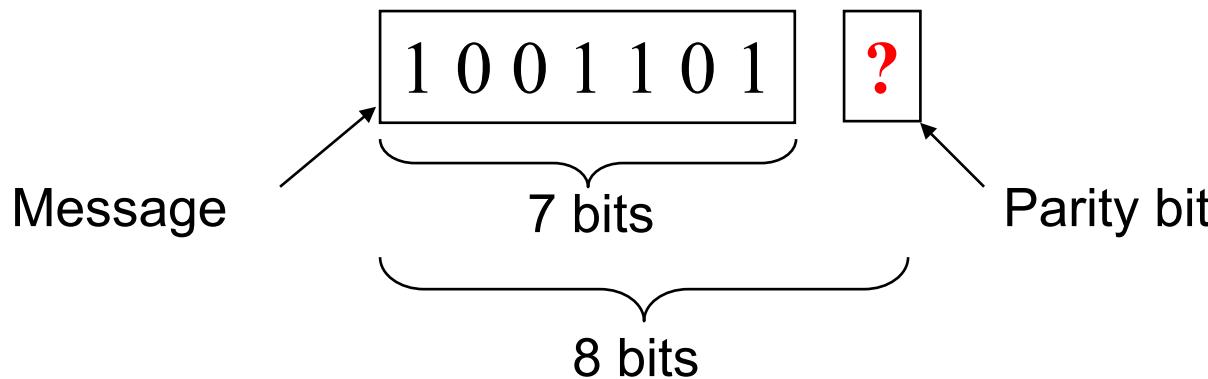
Error Control in Data Link Layer

- **Objective**
 - To detect and correct errors that occur in frame transmission
- **Frame Error in Data Link Layer (DLL)**
 - **Lost Frame**: the receiver does not receive a frame (or the header was corrupted such that the frame was not recognizable)
 - **Damaged Frame**: the receiver receives a frame, but some of its bits are in error

Error Detection Techniques

! Error Detection: Parity Check

Parity Check (Odd/Even Parity): A single bit is appended to the original message (usually 7-bit) to describe the message characteristics.



Even Parity: The total number of 1s is even, ie. 10011010

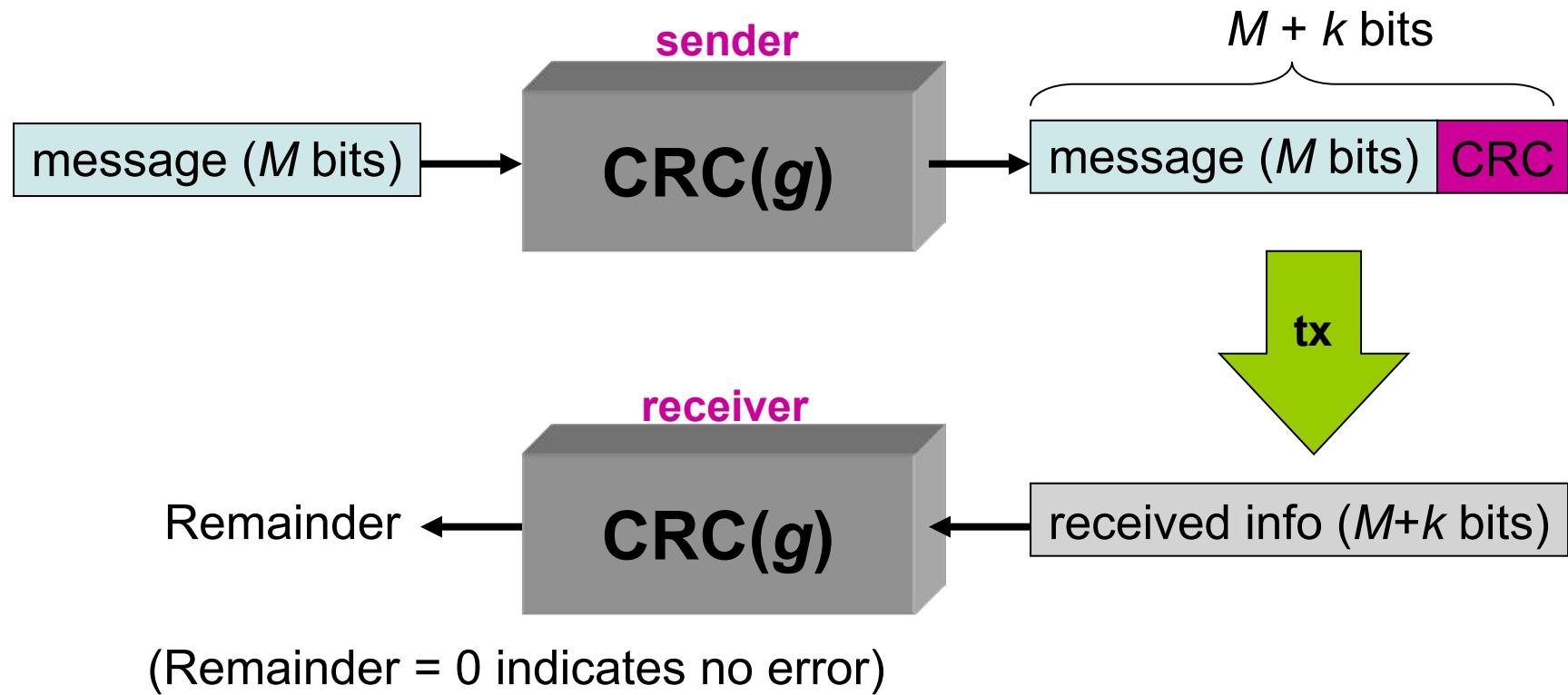
Odd Parity: The total number of 1s is odd, ie. 10011011

⌚ However, Parity Check can only detect odd numbers of errors!

Not in exam

Error Detection: CRC

Cyclic Redundancy Check (CRC): multiple parity bits are appended to the original message.



Error Correction Technique: Automatic Repeat Request (ARQ)



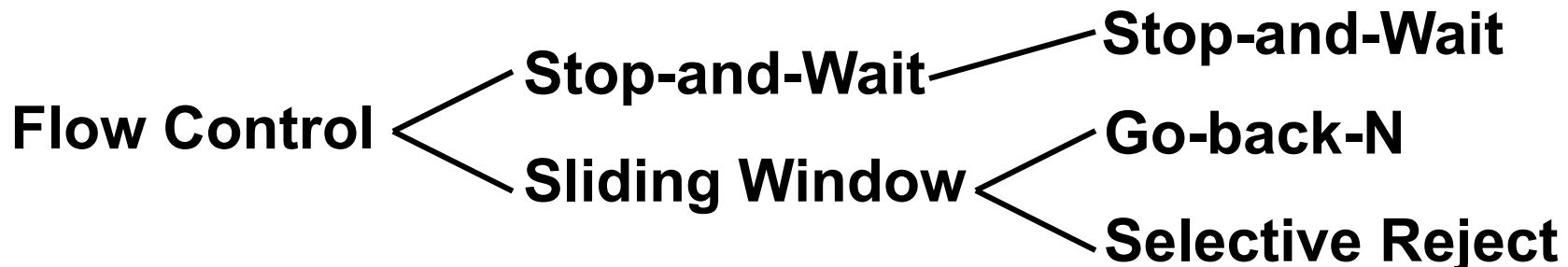
Error Correction Techniques

- **Forward Error Correction (FEC)**
 - Send more redundant bits in the message
 - Example: Hamming code, Reed-Solomon code
- **Automatic Repeat Request (ARQ)**
 - **Retransmission after timeout**: The source retransmits a frame when an expected ACK fails to return within a predetermined time duration
 - **Retransmission when requested**: The destination replies a negative ACK to inform the source about an error. The source then retransmits the corrupted frames accordingly.

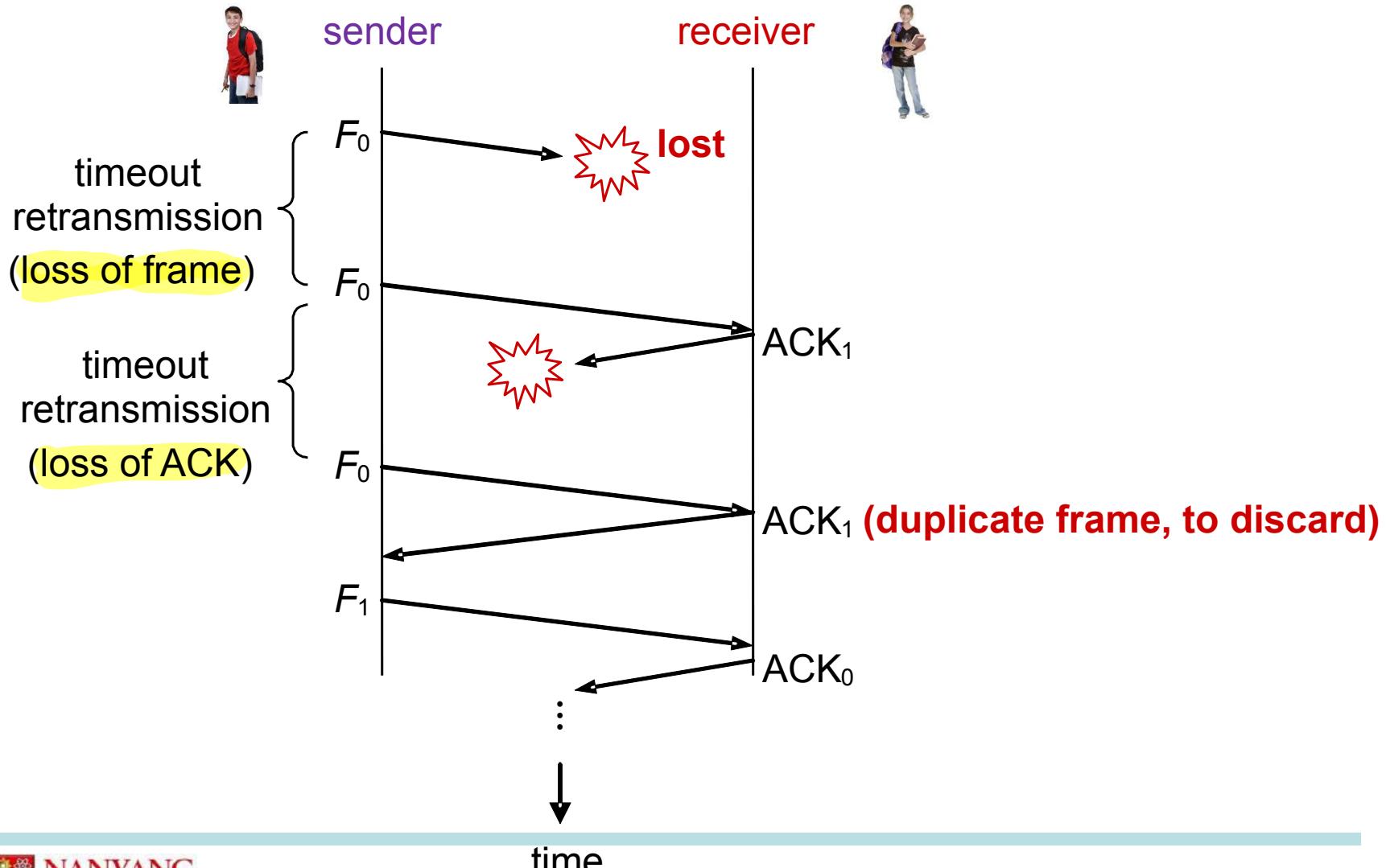


ARQ Variants

- Commonly implemented ARQ mechanisms:
 - Stop-and-Wait ARQ
 - Sliding Window - Go-back-N ARQ:
Frames are accepted strictly in the sequence.
 - Sliding Window - Selective-Reject ARQ:
Sometimes called “Selective Repeat ARQ”. Frames which arrive out of sequence (but are within the open window at the receiver) are accepted.



Stop-and-Wait ARQ: Illustration



Stop-and-Wait ARQ (Protocol)

- **Source:** transmits a single frame and waits for ACK.
- **Destination:**
 - Frame received correctly - send an ACK.
 - Damaged frame received - There are two variations:
 - Discard it, and do nothing else.
 - Send a **NAK** (negative acknowledgement).
- **Source:**
 - If ACK is received properly, transmit next frame.
 - If NAK is received, retransmit the same frame.
 - If no ACK is received within timeout, transmitter timeouts, and retransmits the same frame.
 - If ACK is damaged, transmitter will not recognize it, transmitter will timeout and retransmit the same frame. Receiver gets two copies of the same frame, discard one.

Stop-and-Wait ARQ: Performance

Throughput (U)
(Link Utilization) = $\frac{\text{The time that the link carries useful information}}{\text{The total time}} = \frac{T_{frame}}{T_{cycle}}$

$$U_{SaW}^{ARQ} = \frac{1}{1+2a} \Pr\{\text{no error}\} + 0 \cdot \Pr\{\text{frame error}\}$$

$$= \frac{1}{1+2a} (1-P) + 0 \cdot P$$

$$= \frac{1-P}{1+2a}$$

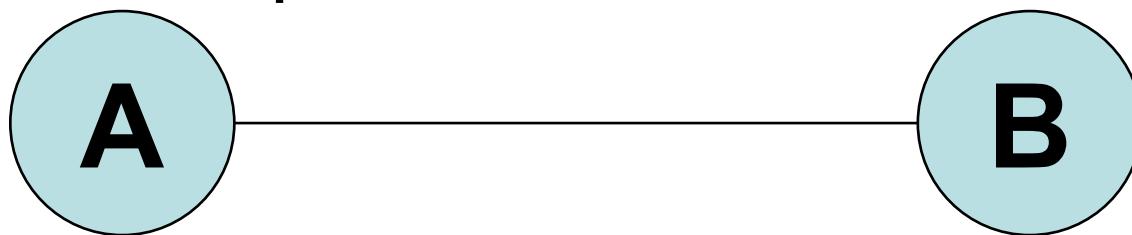
how much time travel
Come from transmission delay

$$\alpha = \frac{T_{\text{Propagation}}}{T_{\text{frame}}}$$

P: Frame loss probability
a: normalized prop. delay

Example

A communication link exists between two nodes A and B. The transmission rate on the link is 2.4 Mbps. The distance between A and B is 50 km and the signal velocity is 2×10^8 m/s. The frame length is 300 bytes. Frame loss probability is 0.1. Calculate the link utilization for the stop-&-wait ARQ mechanism.



$$R = 2.4 \text{ Mbps}, L = 300 \text{ bytes} = 2400 \text{ bits}$$

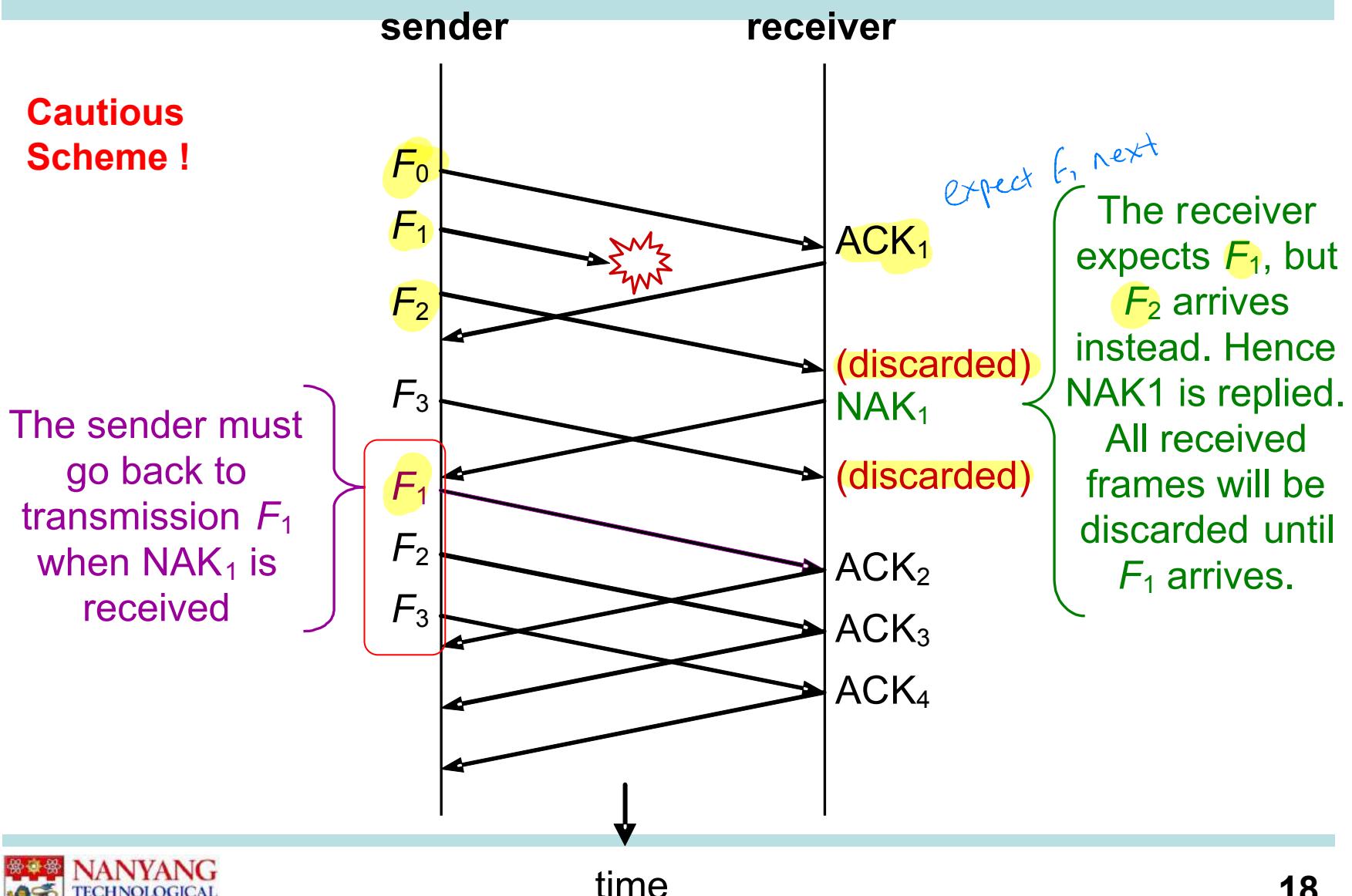
$$H = 50 \text{ km}, v = 2 \times 10^8 \text{ m/s}$$

$$P = 0.1$$

$$U = (1 - P) / (1 + 2a) \longrightarrow a = T_p / T_f \longrightarrow T_p = H/V = 5 \times 10^4 / 2 \times 10^8 = 250 \mu\text{s}$$

$$U = (1 - 0.1) / (1 + 2 * 0.25) \leftarrow a = 0.25 \leftarrow T_f = L/R = 2400 / 2.4 \times 10^6 = 1000 \mu\text{s}$$
$$= 0.6$$

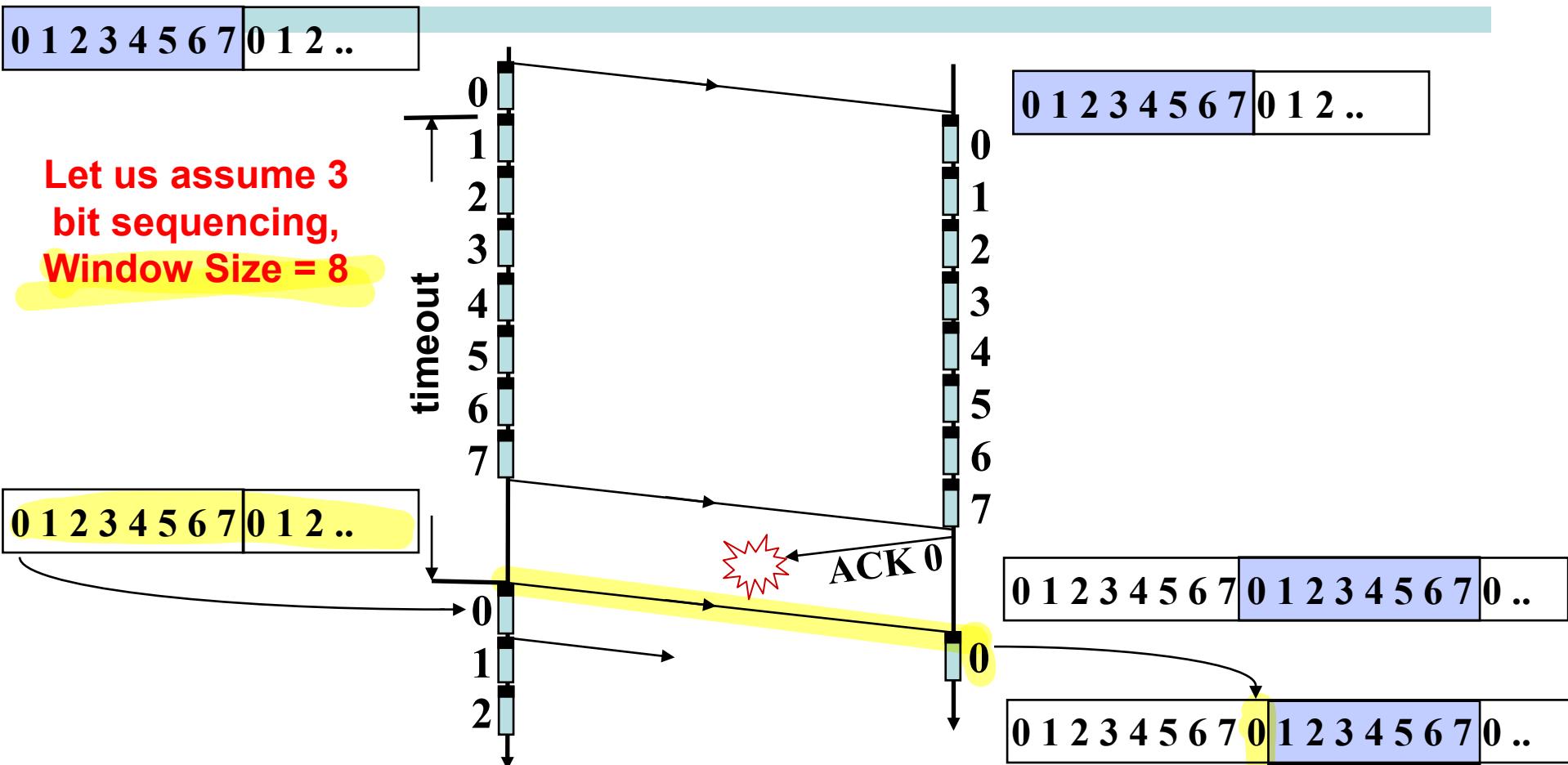
Go-Back-N ARQ: Illustration



Go-Back-N ARQ: Protocol

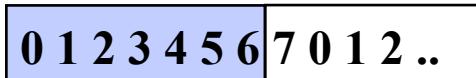
- **Source:** transmits frames sequentially based on sliding window.
- **Destination:**
 - For error-free frames, ACKs are sent as usual. ACK is usually called ‘Receive Ready’ (RR)
 - Can use ‘Receiver Not Ready’ (RNR) for controlling the flow.
 - **If a damaged frame is received, NAK is sent.** NAK is usually called ‘Reject’ (REJ). The destination discards that frame, and all subsequent frames until erroneous frame is received correctly.
- **Source:**
 - **If NAK is received, retransmit that frame and all subsequent frames.**

Go-Back-N: Max Window Size

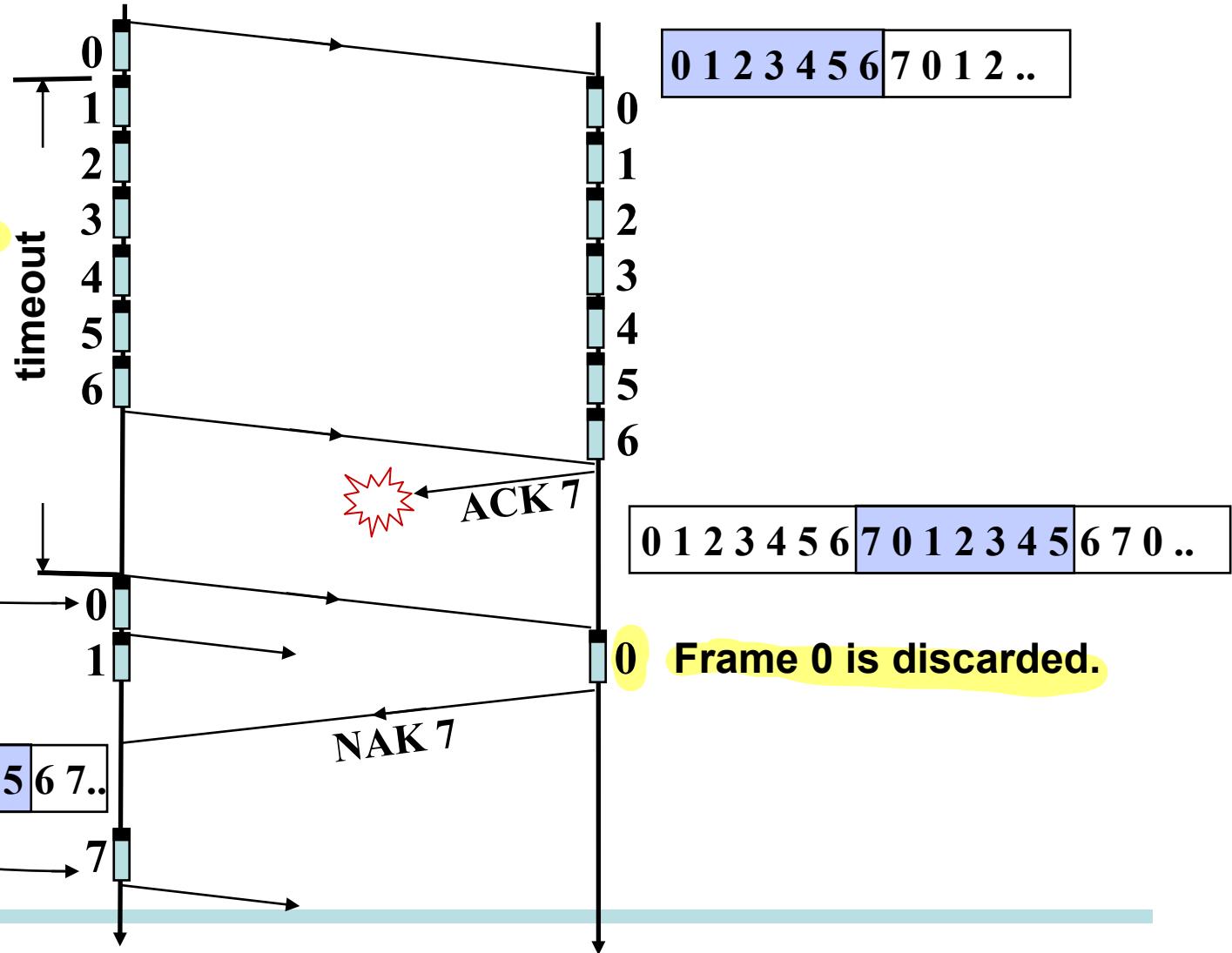


Frame 0 is inserted at a wrong place. For this reason, maximum window size allowed is one less than that permitted by the sequence number. With k bit sequencing, max. window size is $2^k - 1$.

Go-Back-N: Max Window Size



Let us assume 3 bit sequencing,
Window Size = 7

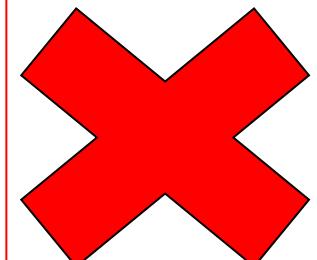


Go-Back-N: Performance

Assumptions:

1. T_{ack} and T_{proc} are negligible.
2. Frames are never completely lost on the medium.
3. ACKs and NAKs are never in error.
4. Each frame is (individually) acknowledged immediately.
5. Sender always has frames to send.

$$U_{GBN}^{ARQ} = \begin{cases} \frac{1-P}{1+2aP} & N \geq 2a+1 \\ \frac{N(1-P)}{(1-P+NP)(1+2a)} & N < 2a+1 \end{cases}$$



Go-Back-N: Performance

Assumptions:

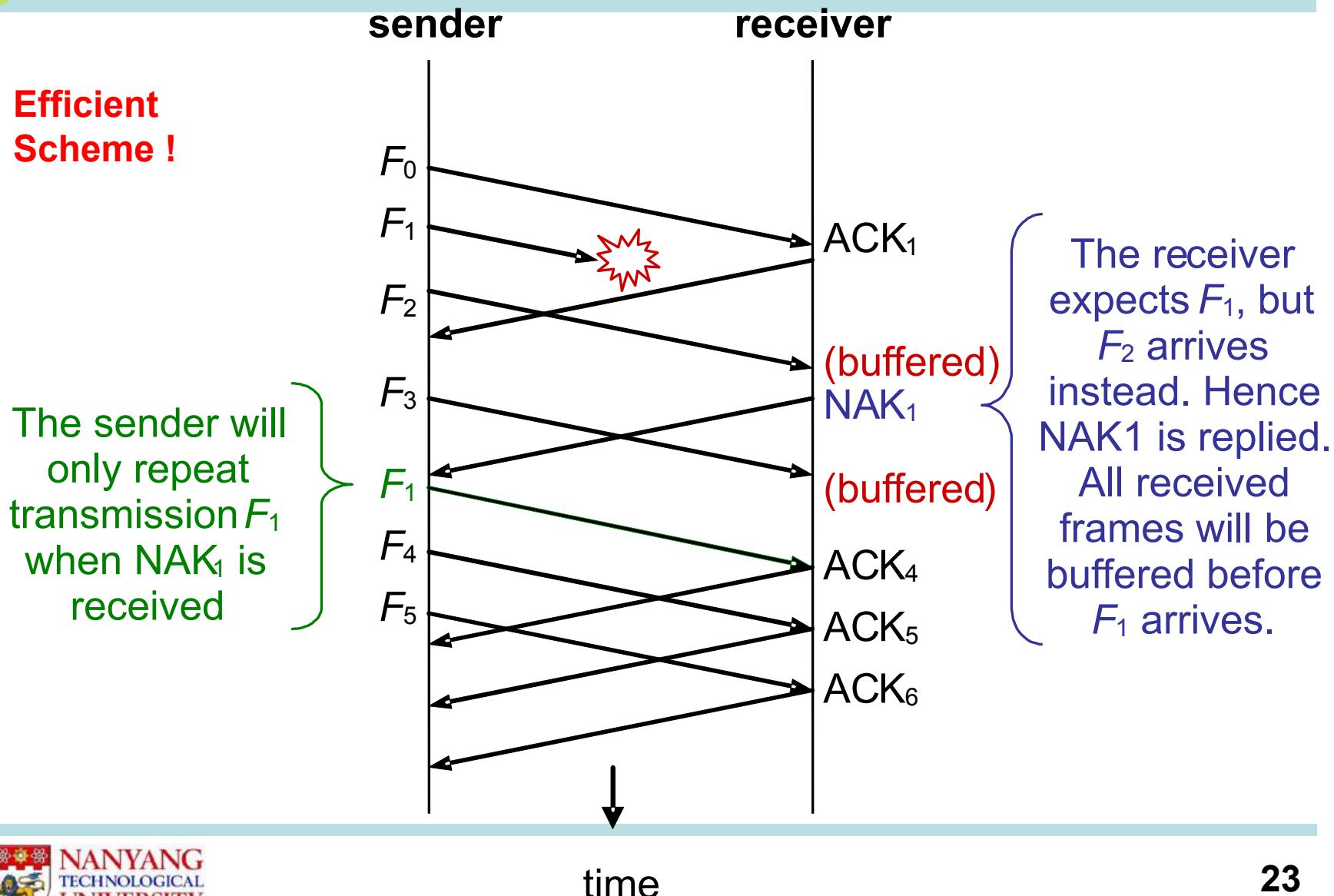
1. T_{ack} and T_{proc} are negligible.
2. Frames are never completely lost on the medium.
3. ACKs and NAKs are never in error.
4. Each frame is (individually) acknowledged immediately.

$$U_{\text{ARQ}}^{\text{Go-Back-N}} = \begin{cases} \frac{(1-P)}{1+2aP} & N \geq 2a+1 \\ \frac{V(1-P)}{(1-P+NP)(1+2a)} & N \leq 2a+1 \end{cases}$$

This link utilization formula for Go-Back-N is complex, so we do not require you to remember it. When you need to compute the link utilization for Go-Back-N, you just use the link utilization formula for Selective Reject ARQ of Slide 27. In other words, to simplify your computation, we consider link utilization for Go-Back-N is the same as link utilization for Selective Reject ARQ.

Selective Reject ARQ: Illustration

Efficient Scheme !

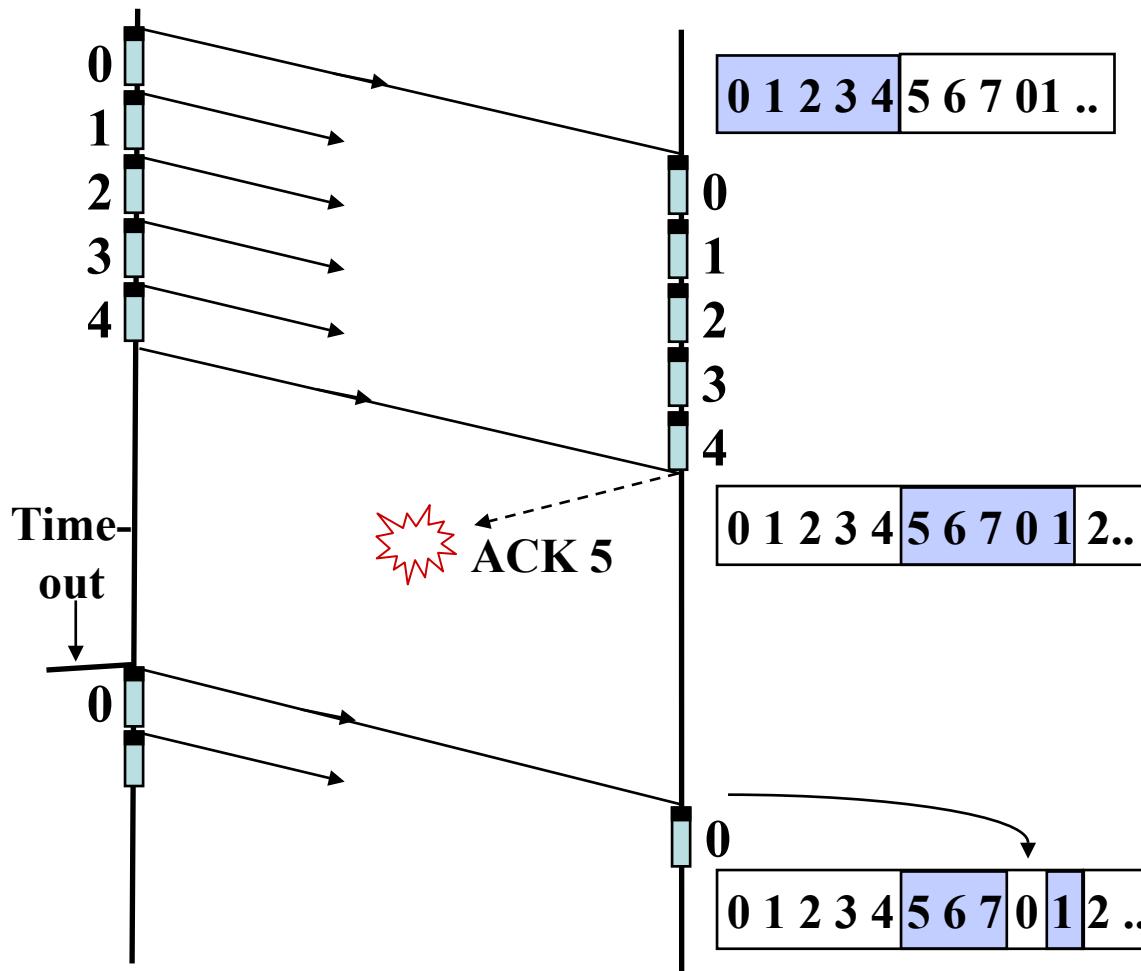


Selective Reject ARQ

- Only rejected frames are retransmitted, (and of course those that time out).
- Receiver informs transmitter of rejected frame n by sending ‘NAK n ’ (‘Selective Reject n ’ or simply ‘SREJ n ’ in HDLC implementation)
- After receiving an erroneous frame, subsequent frames are accepted by the receiver and buffered.
- After receiving the valid copy of the error frame, frames are put in proper order and passed to the higher layer.
- Minimizes retransmission, and thus more efficient than Go-back-N.
- Receiver requires more complex buffer management.

Selective Reject ARQ: Max Window Size

2^{k-1}

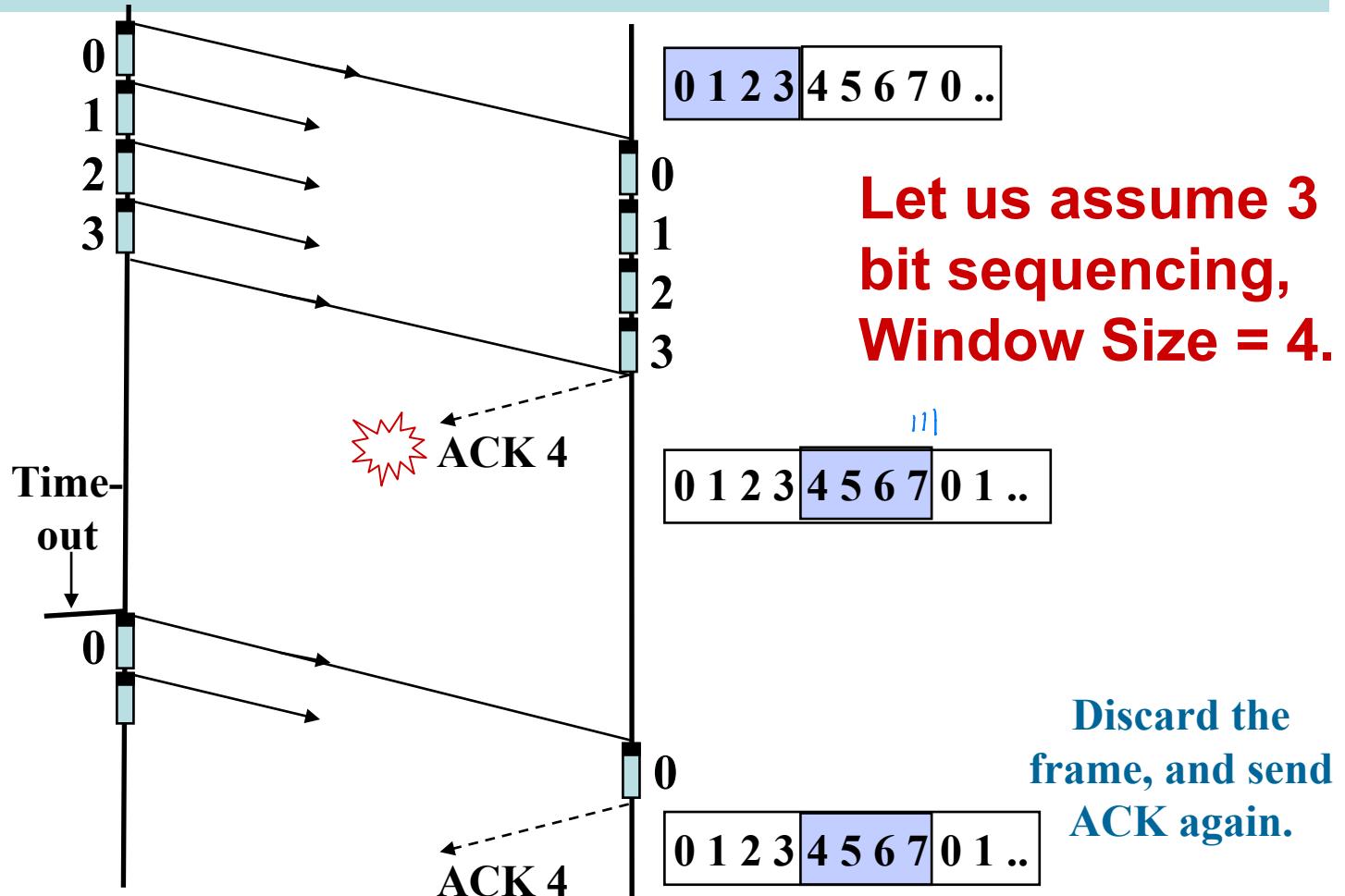


Let us assume 3 bit sequencing. Window Size = 5

The station assumes that frames 5, 6, & 7 have been lost, and it will accept frame 0, (and 1).

Conclusion: Window size of 5 cannot be permitted with 3 bit sequencing

Selective Reject ARQ: Max Window Size



Conclusion: With k bit sequencing, max window size is 2^{k-1} .

Selective Reject ARQ: Performance

P : Frame loss probability

a : normalized prop. Delay

$$U = \frac{N\bar{F}}{1+2a}, N < 2a+1$$

Since frame loss prob for each tx is independent, in $1+2a$ cycle, we expect N transmissions, each with prob P of failure due to errors.

where $\Pr\{F = n\} = \begin{cases} P, & n = 0 \\ 1 - P, & n = 1 \end{cases}$

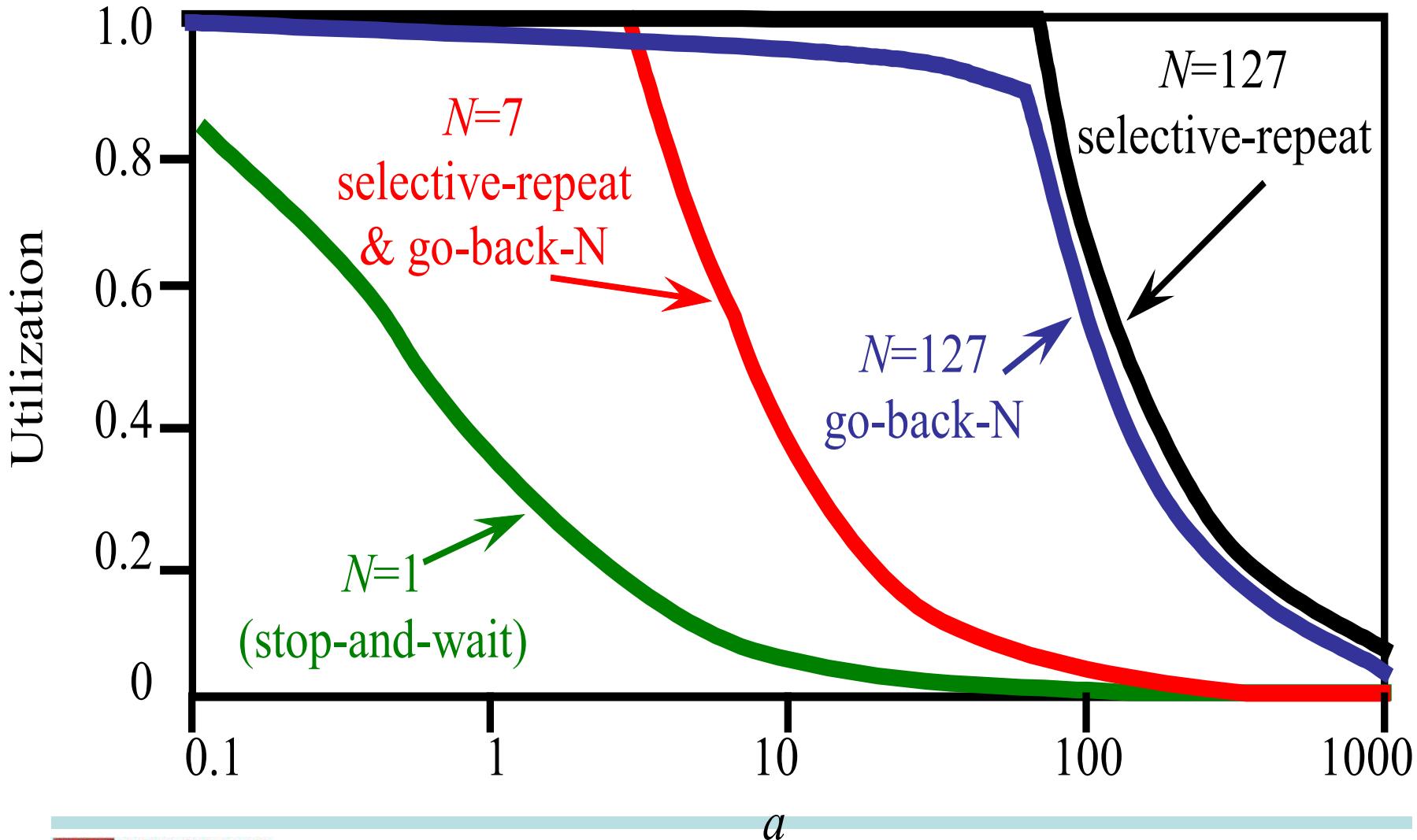
and $\bar{F} = 1 - P$.

$$\text{Hence } U_{\text{Selective reject}} = \frac{N(1-P)}{1+2a}$$

$$U_{SR}^{ARQ} = \begin{cases} 1 - P & N \geq 2a + 1 \\ \frac{N(1 - P)}{1 + 2a} & N < 2a + 1 \end{cases}$$

Setting $P=0$ reduces the above to that of Sliding Window.

ARQ Performance

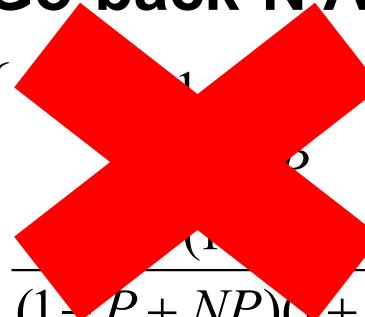


! Channel Utilization: Formulas

Stop-and-Wait ARQ

$$U_{\text{Stop-and-Wait}} = \frac{1-P}{1+2a}$$

Go-back-N ARQ

$$U_{\text{Go-back-}N} = \begin{cases} \frac{1}{(1-P+NP)(1+2a)} & N \geq 2a+1 \\ \frac{1}{(1-P)(1+2a)} & N < 2a+1 \end{cases}$$


Sliding Window (no errors)

$$U_{\text{Sliding Window}} = \begin{cases} 1 & N \geq 2a+1 \\ \frac{N}{1+2a} & N < 2a+1 \end{cases}$$

P: frame error probability

a: normalized propagation delay

N: window size

U: Channel Utilization (between 0 and 1)

Selective Reject ARQ

$$U_{\text{Selective reject}} = \begin{cases} 1-P & N \geq 2a+1 \\ \frac{N(1-P)}{1+2a} & N < 2a+1 \end{cases}$$

Learning Objectives

- **Stop-and-Wait ARQ**
 - To label frame flow
 - Channel Utilization Calculation
- **Go-Back-N ARQ (GBN)**
 - To label frame flow
 - To determine Max Window Size
- **Selective Reject ARQ (SR)**
 - To label frame flow
 - To determine Max Window Size
 - Link utilization calculation
 - Comparison between GBN and SR

Learning Objectives

- **Stop-and-Wait ARQ**
 - To label frame flow
 - Channel Utilization Calculation
- **Go-Back-N ARQ (GBN)**
 - To label frame flow
 - To determine Max Window Size
- **Selective Reject ARQ (SR)**
 - To label frame flow
 - To determine Max Window Size
 - Link utilization calculation
 - Comparison between GBN and SR

Part I Syllabus - Fundamental Underlying Layers

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Mobile Access Networks	M1-L9-Mobile.pptx
Week 7: 20/Feb/2023 22/Feb/2023	E-learning for Network paradigms	M1-L10-Paradigms.pptx
	Network paradigms	M1-L10-Paradigms.pptx

Additional Materials

- The related content talked today in
[https://eclasse.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclasse.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:
 - Chapter 5
- <https://www.techtarget.com/searchnetworking/definition/local-area-network-LAN>

Mingling Among a Cocktail party



SC2008/CZ3006/CE3005

Computer Network

most of topic not in exam

Lecture 5

Local Area Network (LAN): Introduction



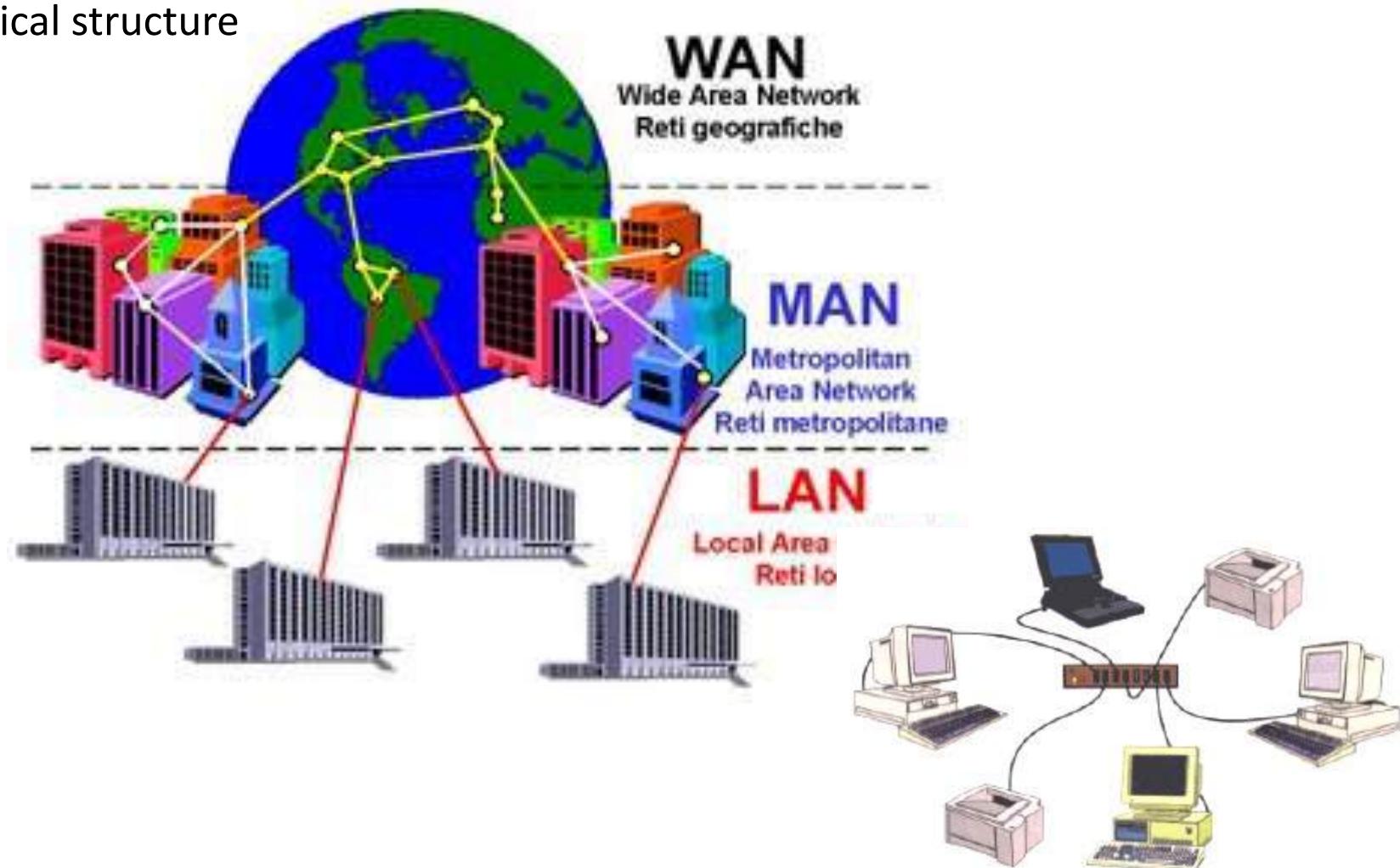
Contents

- **Local Area Network**
 - Definition and Taxonomy
 - Protocol Architecture
- **LAN Topologies**
 - Bus, Tree, Ring and Star
 - Choice of topology
- **Transmission Media**
- **Medium Access Control**
 - Functions and Features
 - Static Channel Allocation
 - Dynamic Channel Allocation

Local Area Network (LAN)

WAN/MAN/LAN

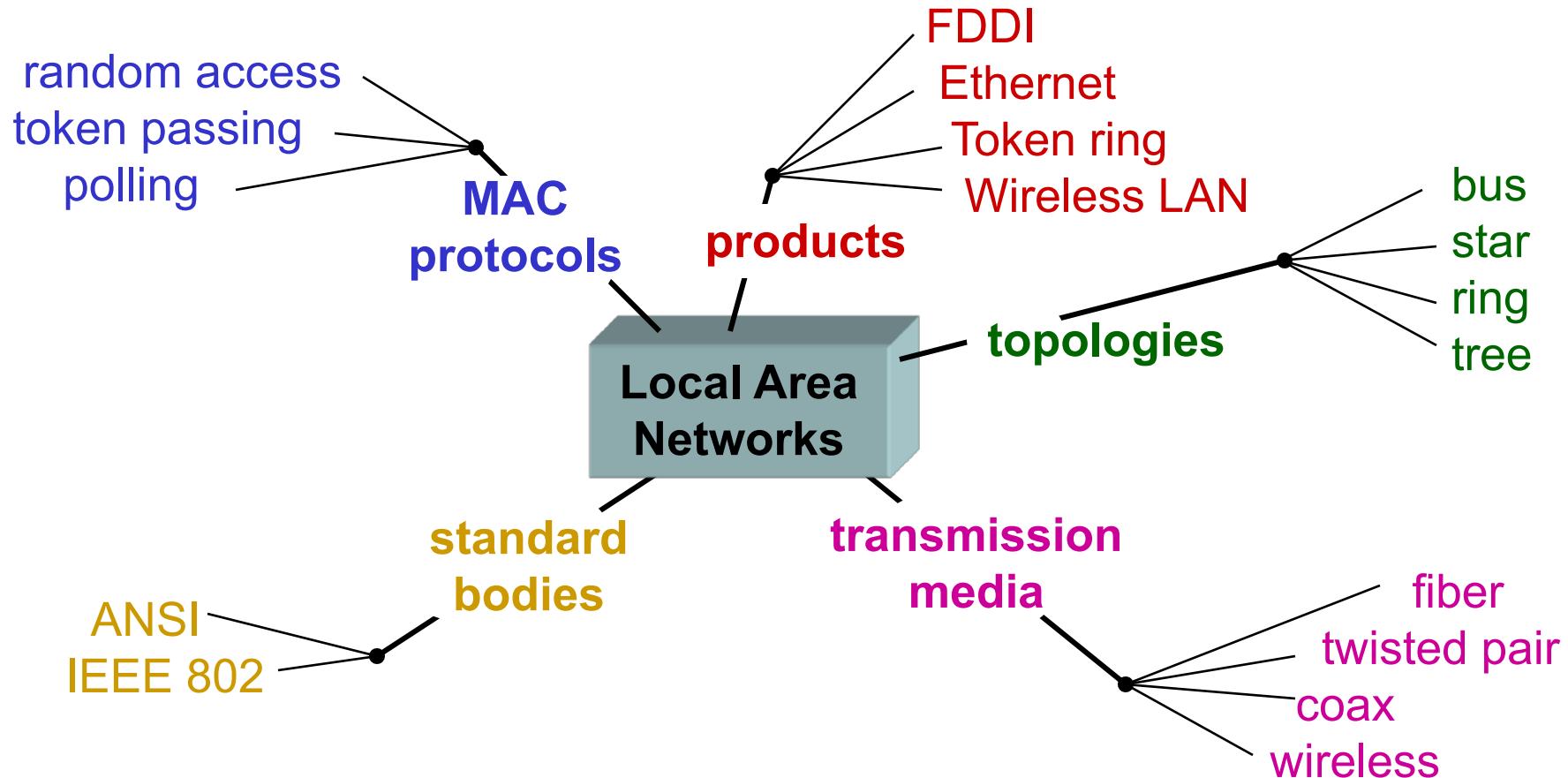
hierarchical structure



LAN (Local Area Networks)

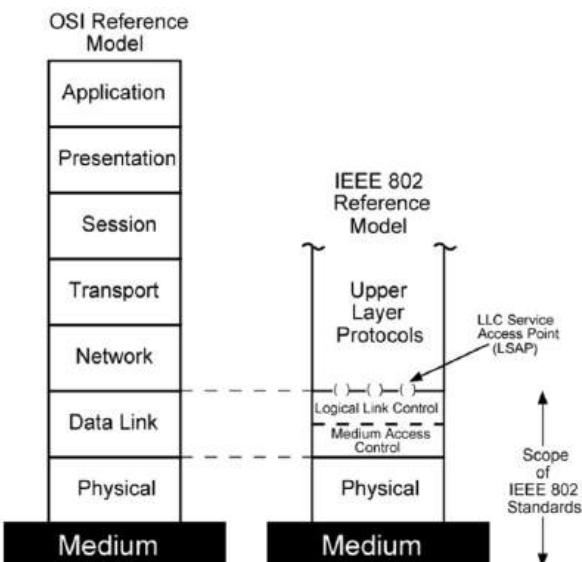
- **LAN is a computer network that covers a small area (home, office, building, campus)**
 - a few kilometers
- **LANs (usually) do not involve leased lines; cabling and equipments belong to the LAN owner.**
- **LAN consists of**
 - Shared transmission medium
 - not so valid today due to switched LANs
 - regulations for orderly access to the medium
 - set of hardware and software for the interfacing devices

LAN Taxonomy



LAN Protocol Architecture

- Corresponds to lower two layers of OSI model
 - But mostly LANs do not follow OSI model
- Current LANs are most likely to be based on Ethernet protocols developed by IEEE 802 committee
- IEEE 802 reference model
 - Logical link control (LLC)
 - Media access control (MAC)
 - Physical



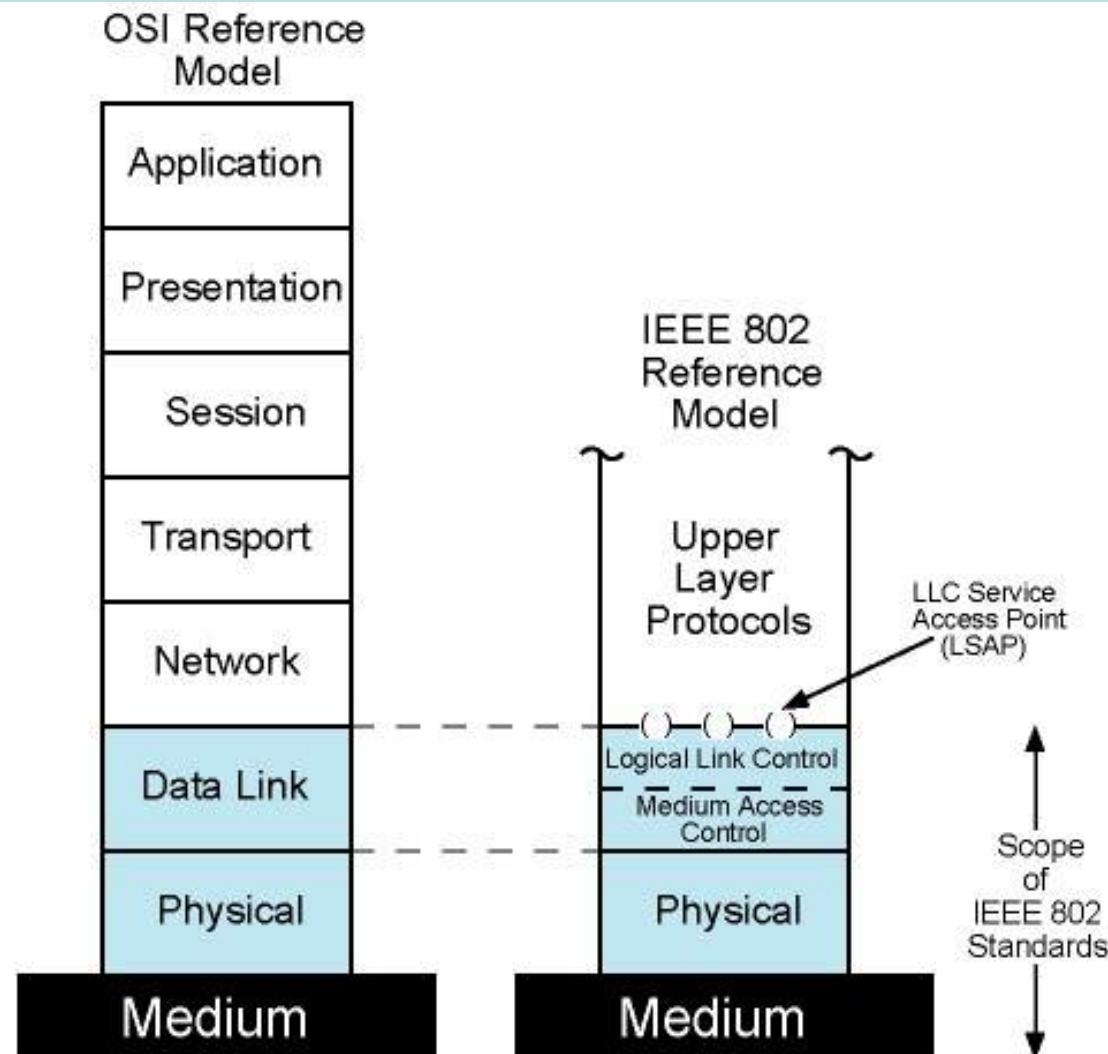
IEEE 802 Layers - Physical

- **Signal encoding/decoding**
- **Preamble generation/removal**
 - for synchronization
- **Bit transmission/reception**
- **Specification for topology and transmission medium**
- **WiFi (Wireless Fidelity) vs. LiFi (Light Fidelity)**

IEEE 802 Layers - DLL

- **OSI layer 2 (Data Link) is divided into two in IEEE 802**
 - Logical Link Control (LLC) layer
 - Medium Access Control (MAC) layer
- **LLC layer**
 - Interface to higher levels
 - flow control, error control
 - Based on classical Data Link Control Protocols (so we have already covered it earlier)
- **MAC layer**
 - Prepare data for transmission
 - Error detection
 - Address recognition
 - Govern access to transmission medium
 - Not found in traditional layer 2 data link control

IEEE 802 Protocols vs OSI Model



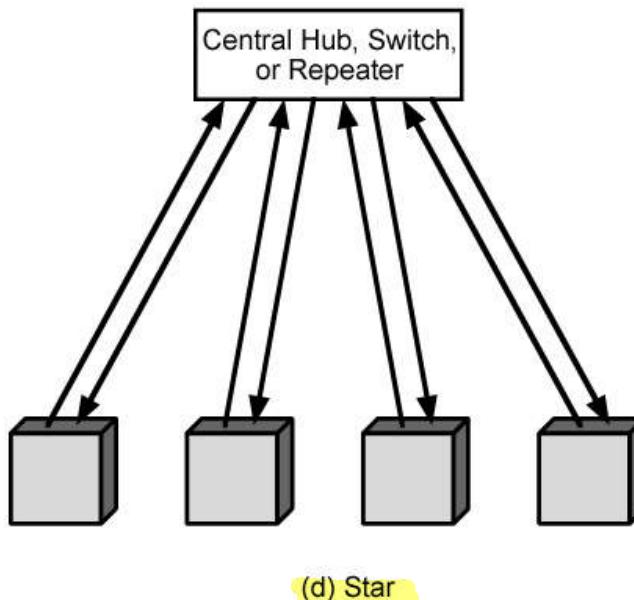
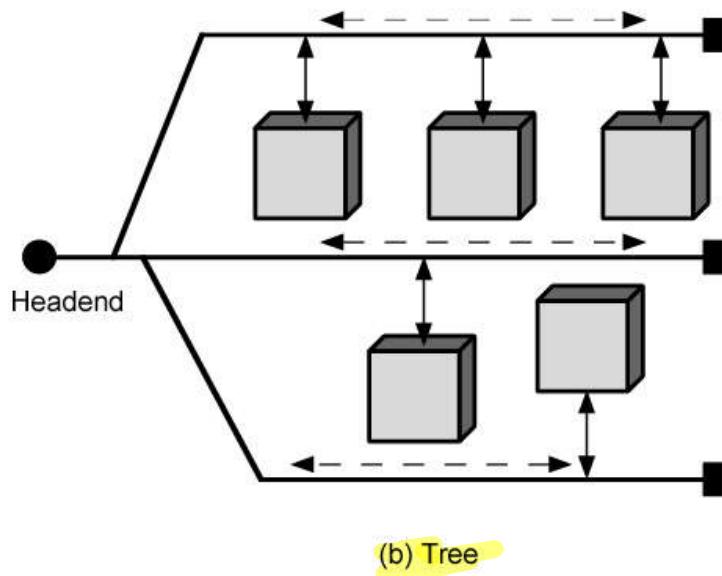
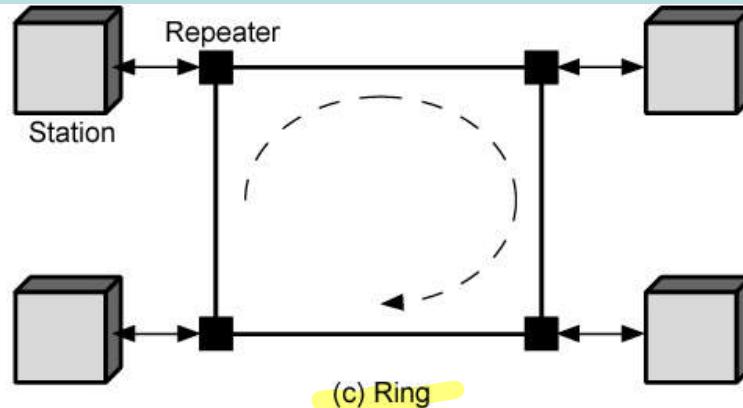
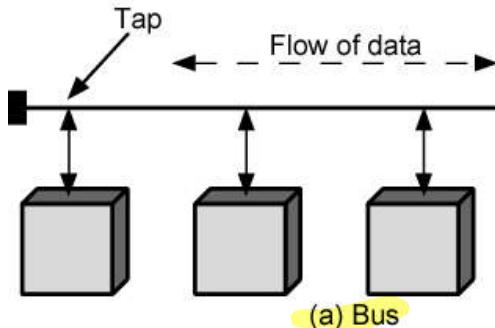
LAN in a Nutshell

		IEEE 802.2 Logical Link Control Protocol						
Data Link	LLC	802.3	802.4	802.5	802.6	802.11	802.12	802.14
MAC	CSMA /CD used by Ethernet	Token Bus	Token Ring	DQDB	CSMA /CA used by WiFi	Round Robin		HFC
Physical	Coax UTP STP Fiber B,T,S	Coax Fiber B,T,S	UTP STP Fiber R	Fiber DB	Radio Infrared ---	UTP S, T	Coax T	

Topologies (see next slide): Bus, Tree, Star, Ring, DualBus

LAN Topologies

LAN Topologies: Bus, Tree, Ring and Star

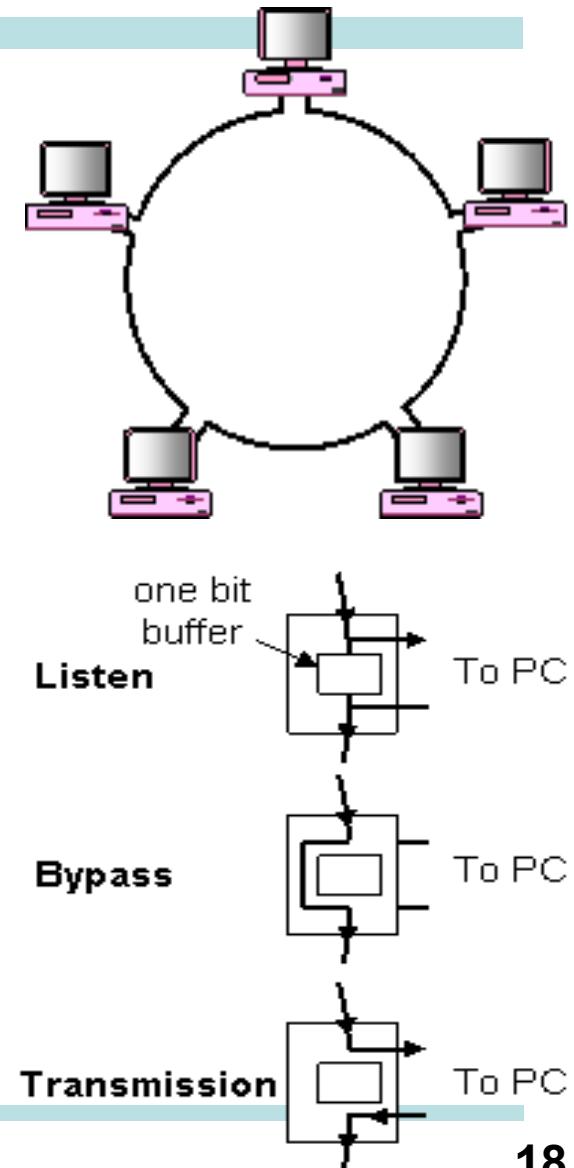


Bus and Tree

- **Multipoint medium**
- **Transmission propagates throughout medium**
- **Heard by all stations**
 - Need to identify target station
 - Each station has unique address
- **Full duplex connection between station and tap**
 - Allows for transmission and reception
- **Need to regulate transmission**
 - To avoid collisions
 - If two stations transmit at same time, signals overlap
 - To avoid continuous transmission from a single station.
 - Solution: Transmit Data in small blocks – frames
- **Terminator absorbs frames at end of medium**

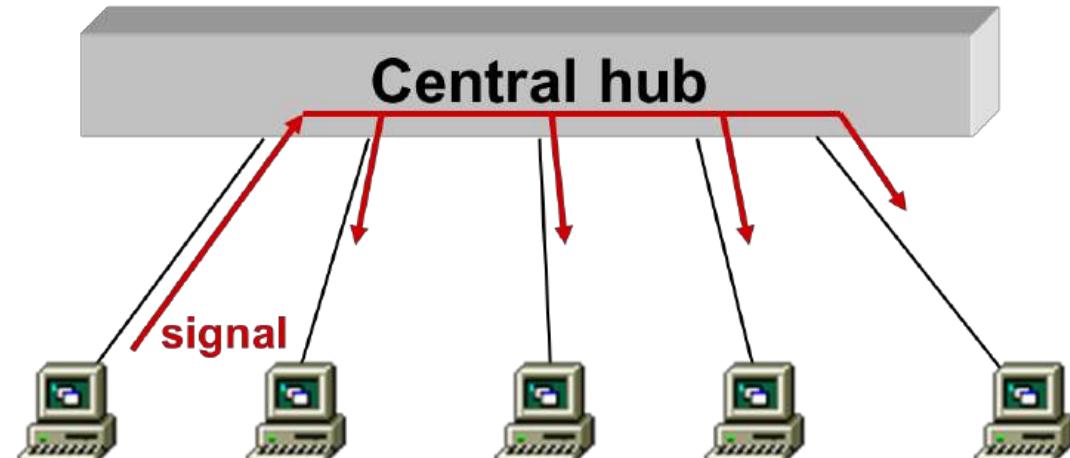
Ring Topology

- **Repeaters are joined by point to point links in closed loop**
 - Receive data on one link and retransmit on another
 - Links are unidirectional
 - Stations attach to repeaters
- **Data Frames**
 - Circulate past all stations
 - Destination recognizes address and copies frame
 - Frame circulates back to source where it is removed
- **Medium access control determines when station can insert frame**



Star Topology

- **Each station connected directly to central node**
 - using a full-duplex (bi-directional) link
- **Central node can broadcast (hub)**
 - Physical star, but logically like bus since broadcast
 - Only one station can transmit at a time; otherwise, collision occurs
- **Central node can act as frame switch**
 - retransmits only to destination
 - today's technology



Choice of Topology

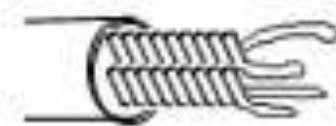
- **Reliability**
- **Expandability**
- **Performance**
- **Needs considering in context of:**
 - Medium
 - Wiring layout
 - Access control

Transmission Medium

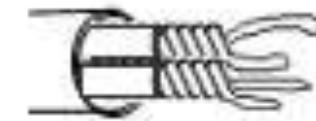
Medium Available (1)

- **Voice grade unshielded twisted pair (UTP)**
 - Cat 3/ Cheap
 - Well understood
 - Use existing telephone wiring in office building
 - Low data rates
- **Shielded twisted pair (STP) and baseband coaxial**
 - More expensive than UTP but higher data rates
- **Broadband cable**
 - Still more expensive and higher data rate

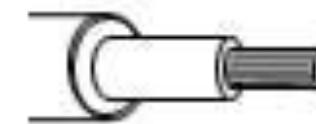
Networking Cables



Unshielded twisted-pair cable



Shielded twisted-pair cable



Coaxial cable

<http://www.computerhope.com>

Media Available (2)

- **High performance unshielded twisted pair (UTP)**
 - Cat 5 and above (5e and 6)
 - High data rate for small number of devices
 - Switched star topology for large installations
- **Optical fiber**
 - Electromagnetic isolation
 - High capacity
 - Small size
 - High cost of components
 - High skill needed to install and maintain
- **Wireless Channel**
 - Fading channel



Media Access Control (MAC)

Media Access Control

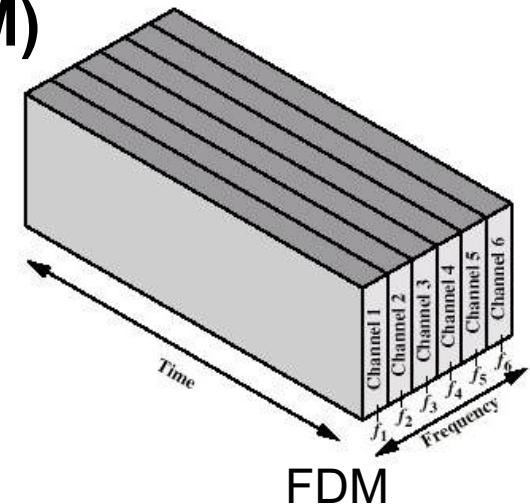
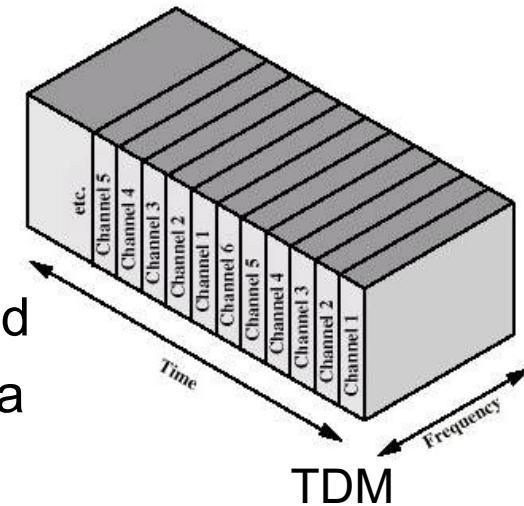
- **Assembly of data into frame with address and error detection fields**
- **Disassembly of frame**
 - Address recognition
 - Error detection
- **Govern access to transmission medium**
 - Not found in traditional layer 2 data link control
- **For the same LLC, several MAC options may be available**

MAC Decision Making Options

- **Where?**
 - Central
 - Greater control
 - Simple access logic at station
 - Avoids problems of co-ordination
 - Single point of failure
 - Potential bottleneck
 - Distributed
- **How?**
 - Synchronous (static) solutions
 - Specific capacity dedicated to connection
 - Asynchronous (dynamic) solutions
 - In response to demand

Static Channel Allocation

- **Time Division Multiplexing (TDM)**
 - Each user is statically allocated one time slot
 - if a particular user does not have anything to send, that period is wasted
 - User may not utilize the whole channel for a time slot
- **Frequency Division Multiplexing (FDM)**
 - Channel is divided to carry different signals at different frequencies
 - Efficient if there is a constant (one for each slot) amount of users with continuous traffic
- **Code Division Multiplexing (CDM)**



Dynamic Channel Allocation (1)

- **Round robin**
 - Each station has a turn to transmit
 - declines or transmits up to a certain data limit
 - overhead of passing the turn in either case
 - Performs well if many stations have data to transmit for most of the time
 - otherwise passing the turn would cause inefficiency
- **Reservation**
 - It is used for stream traffic, where time on the medium is divided into slots, much as with TDM.
 - Reservation can be made in centralized or distributed fashion.

Dynamic Channel Allocation (2)

- **Contention**
 - All stations contend to transmit
 - No control to determine whose turn is it
 - Stations send data by taking risk of collision (with others' packets)
 - however they understand collisions by listening to the channel, so that they can retransmit
 - **Several implementation methods: Aloha, CSMA, etc**
 - In general, good for bursty traffic
 - Typical traffic types for most networks
 - Efficient under light or moderate load
 - Performance is bad under heavy load

Learning Objectives

- **Local Area Network**
 - Functions of each layer: physical, LLC and MAC
 - 802 Protocol family
- **LAN Topologies**
 - Frame transmission over Bus, Tree, Ring and Star
- **Transmission Media**
- **Medium Access Control**
 - Pros and Cons of Static Channel Allocation
 - Comparison among Dynamic Channel Allocation

Part I Syllabus - Fundamental Underlying Layers

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Mobile Access Networks	M1-L9-Mobile.pptx
Week 7: 20/Feb/2023 22/Feb/2023	E-learning for Network paradigms	M1-L10-Paradigms.pptx
	Network paradigms	M1-L10-Paradigms.pptx

How to mingle among cocktail



- 1) When to start speaking?**
- 2) What to speak?**
- 3) Whether/How to react to interruption?**

SC2008/CZ3006/CE3005

Computer Network

Lecture 6

Medium Access Control (MAC) Protocols



Contents

- **Medium Access Control Protocol**
 - Ideal MAC Protocol
 - MAC Taxonomy
- **ALOHA Protocols**
 - Slotted ALOHA
 - Pure ALOHA
- **CSMA Protocol**
 - Vulnerable time in CSMA
 - CSMA Variants
- **CSMA/CD Protocol**
 - Collision Detection

Medium Access Control Protocols

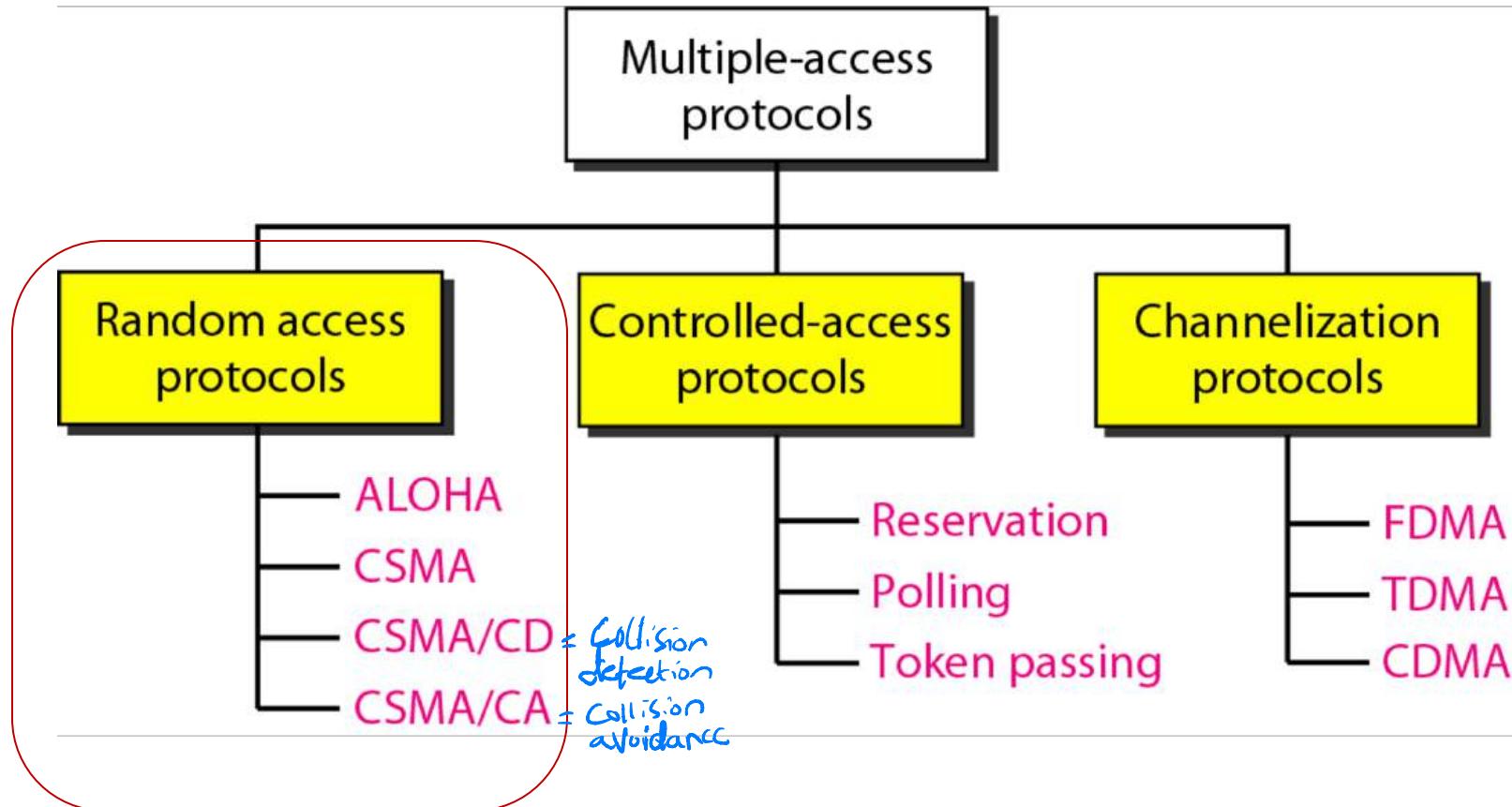
- **Single shared broadcast channel**
- **Two or more simultaneous transmissions**
 - **Collision** if node receives two or more signals at the same time
- **MAC Protocol**
 - Distributed algorithm to share the channel
 - Communication about channel sharing must use channel itself
 - No out-of-band channel for coordination
 - ↳ *inband*

Ideal MAC Protocol

- **Broadcast Channel of Rate R -bps**
 - When one node transmits, it can send at rate R
 - When M nodes want to transmit, each can send at average rate R/M
 - Full decentralized
 - No special node to coordinate transmissions
 - No synchronization of clocks, slots
 - Simple
 - We call this ideal protocol as “*genie-aided*” MAC



MAC Taxonomy



Random Access Protocols

- **When node has packet to send**
 - Transmits at full channel data rate of R
 - No a-priori coordination among nodes
- **Two or more transmitting nodes**
 - Collision
- **Design of random MAC has 3 aspects**
 - Whether to sense channel status
 - How to transmit frames
 - How to detect and react to collision (What to do with the collision)

ALOHA Protocols

aloha

/ə'laʊhə/

exclamation & noun

a Hawaiian word used when greeting or parting from someone.

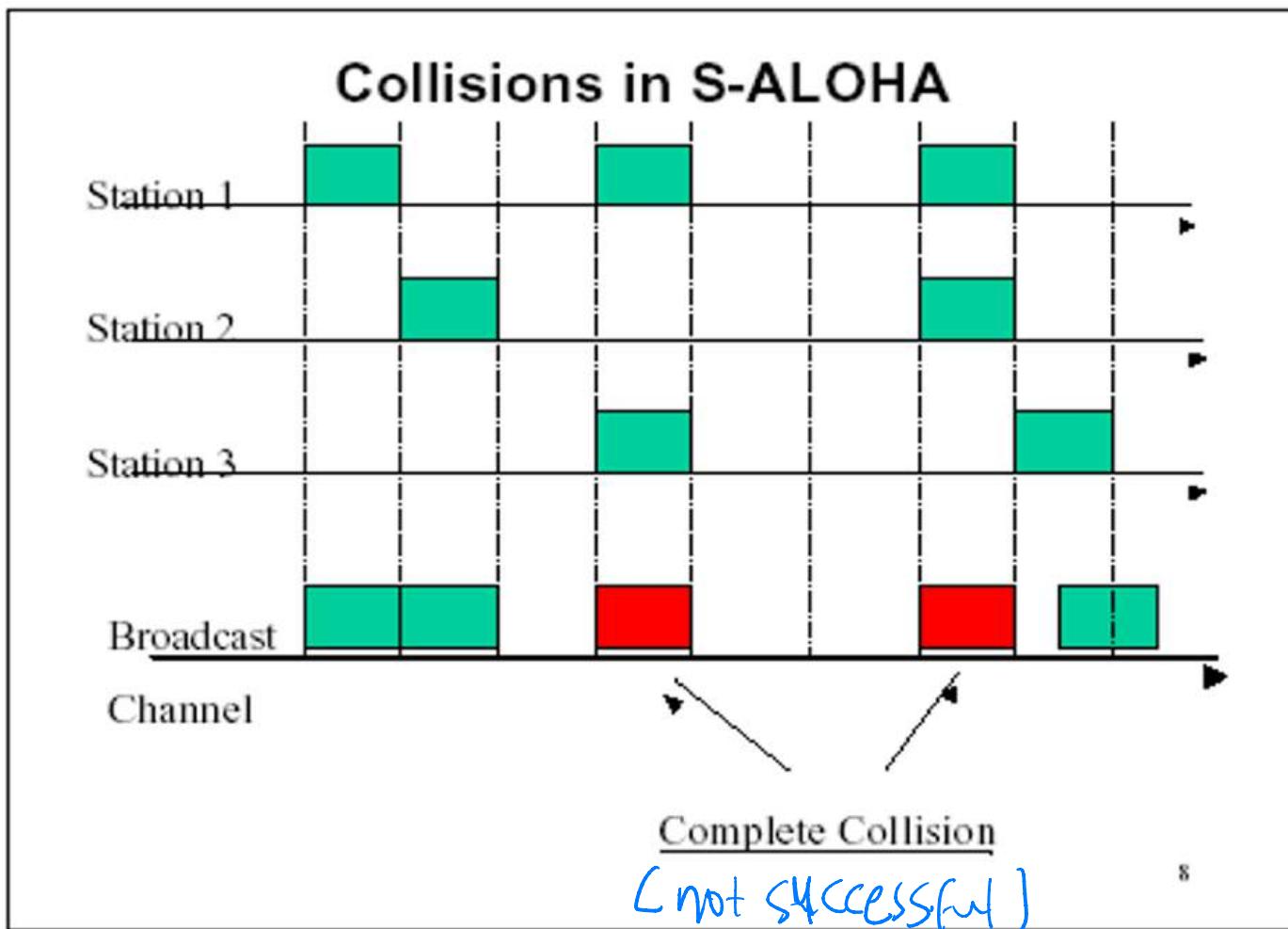
ALOHA

- **Inventor**
 - Norm Abramson
- **Assumptions**
 - All frames of the same size
 - Time is divided into equal size slots, time to transmit 1 frame
 - Nodes are **synchronized**
 - Nodes start to transmit frames only at beginning of slots
 - If 2 or more nodes transmit in slot, all nodes detect collision



Born	April 1, 1932 (age 87) Boston, Massachusetts
Nationality	American
Alma mater	Stanford University Harvard University
Awards	IEEE Alexander Graham Bell Medal (2007) Scientific career
Fields	Electrical Engineering and Computer Sciences
Institutions	University of Hawaii
Doctoral advisor	Willis Harman
Doctoral students	Thomas M. Cover Robert A. Scholtz

Slotted ALOHA

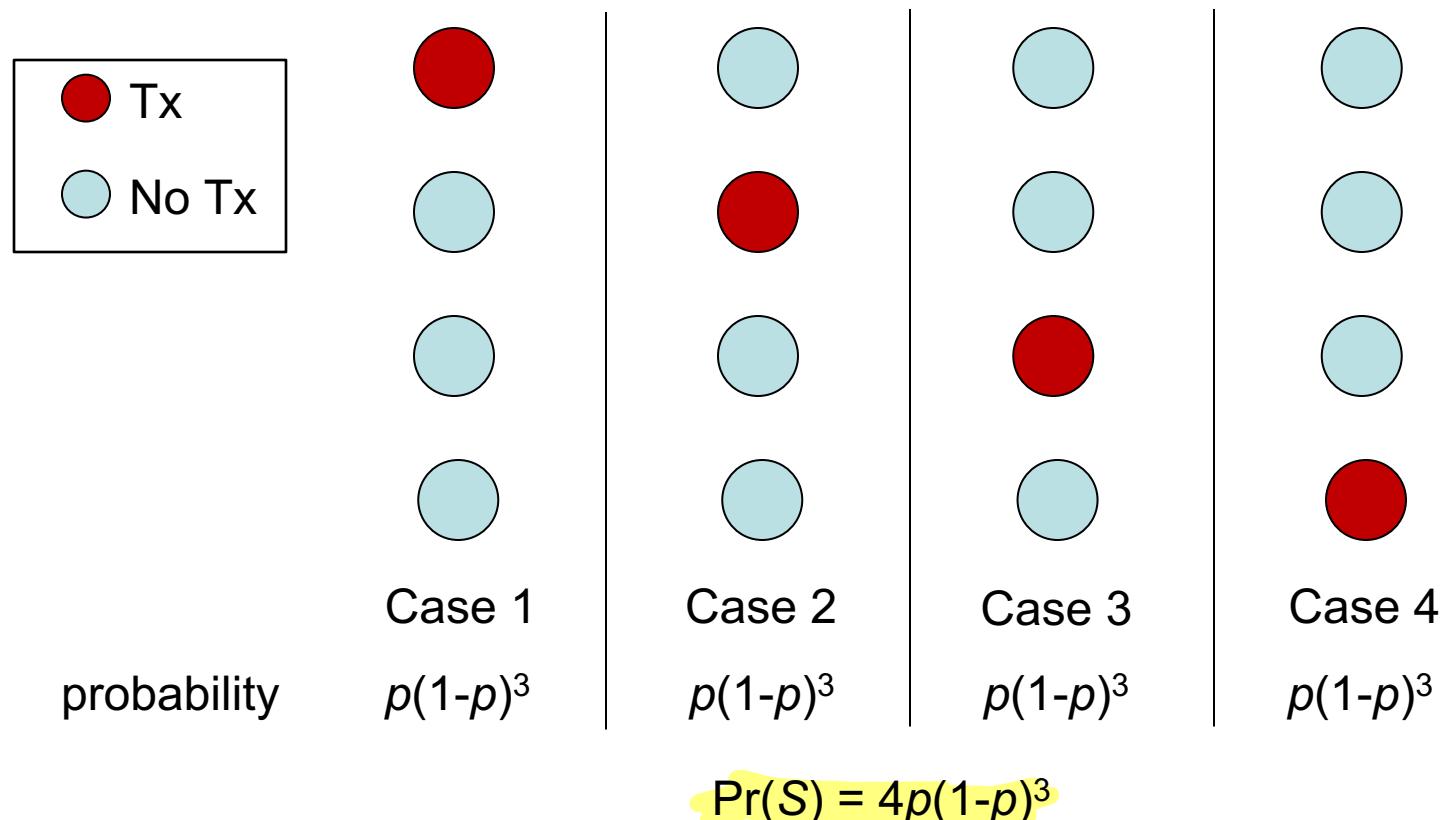


Slotted ALOHA Efficiency

- **Result of a slot**
 - Successful (S): only one node transmits
 - Collision (C): 2^+ nodes transmits
 - Empty (E): no transmission
- **If, there are N nodes and in each slot, each node transmits with probability p**
 - If a node i transmits, the probability that the transmission is successful is $\Pr(S_i) = p (1-p)^{(N-1)}$
 - The probability that a slot is successful is
 $\Pr(S) = N p (1-p)^{(N-1)}$

Slotted ALOHA Efficiency

- An example of 4-node network
 - 4 cases for a successful slot



Slotted ALOHA Efficiency

- **Offered load $G = Np$**

- Expected total number of transmissions in a slot
- **Slotted ALOHA efficiency when N is large**

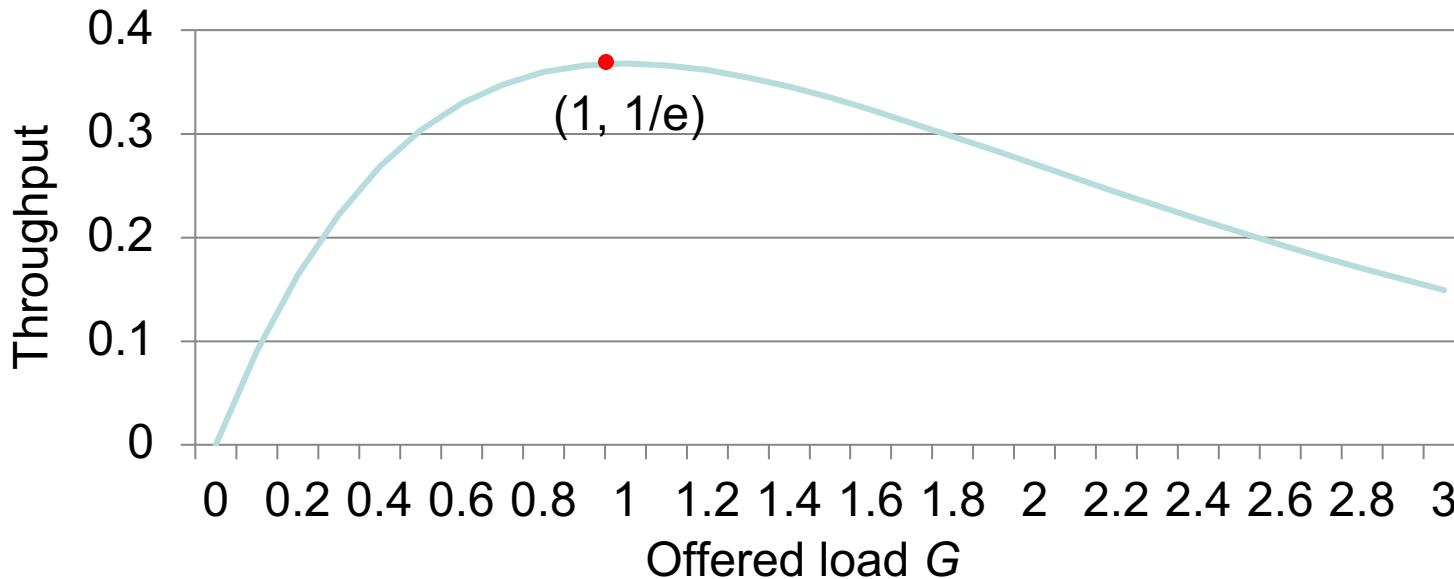
$$\begin{aligned}\lim_{N \rightarrow \infty} \Pr(S) &= \lim_{N \rightarrow \infty} Np(1-p)^{N-1} \\ &\stackrel{*}{=} \lim_{p \rightarrow 0} G(1-p)^{\frac{G}{p}-1} \\ &= G \cdot \left(\lim_{p \rightarrow 0} (1-p)^{1/p} \right)^G \cdot \left(\lim_{p \rightarrow 0} (1-p)^{-1} \right) \\ &\stackrel{**}{=} Ge^{-G}\end{aligned}$$

* When $N \rightarrow \infty$, $p \rightarrow 0$ as G is bounded

** $\lim_{p \rightarrow 0} (1-p)^{\frac{1}{p}} \rightarrow \frac{1}{e}$ by the definition of e

$$e = \lim_{x \rightarrow \infty} \left(1 + \frac{1}{x}\right)^x$$

Slotted ALOHA Efficiency



- $\Pr(S)$ is throughput in frames per frame time
- $\Pr(S) \leq 1/e (\approx 0.37)$ **
 - $1/e$ achieved when $G = 1$
 - At the same time, $\Pr(E) \approx 0.37$, $\Pr(C) \approx 0.26$

** Tutorial 3.4

Pros and Cons of Slotted ALOHA

- **Pros**

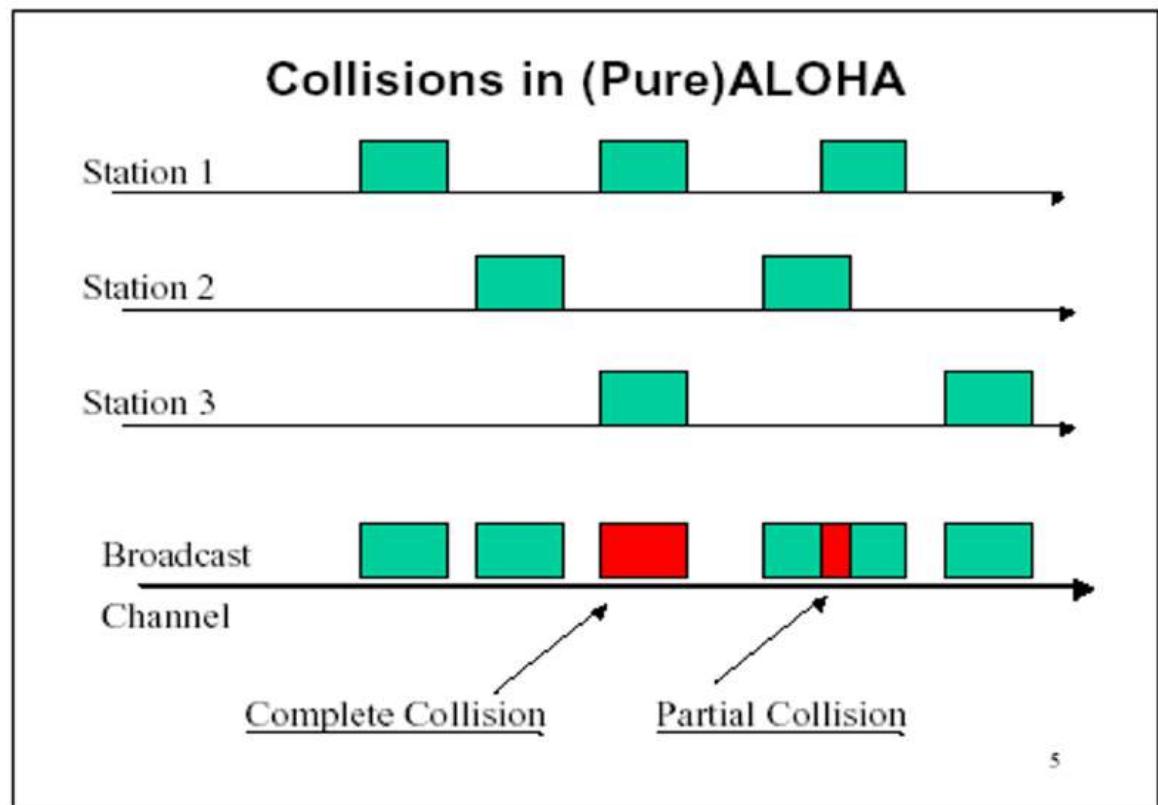
- Single active node can continuously transmit at full rate of channel
- Highly decentralized: only slots need to be sync
- Simple

- **Cons**

- Collisions, wasting slots
- Empty slots
- Clock synchronization

Pure ALOHA

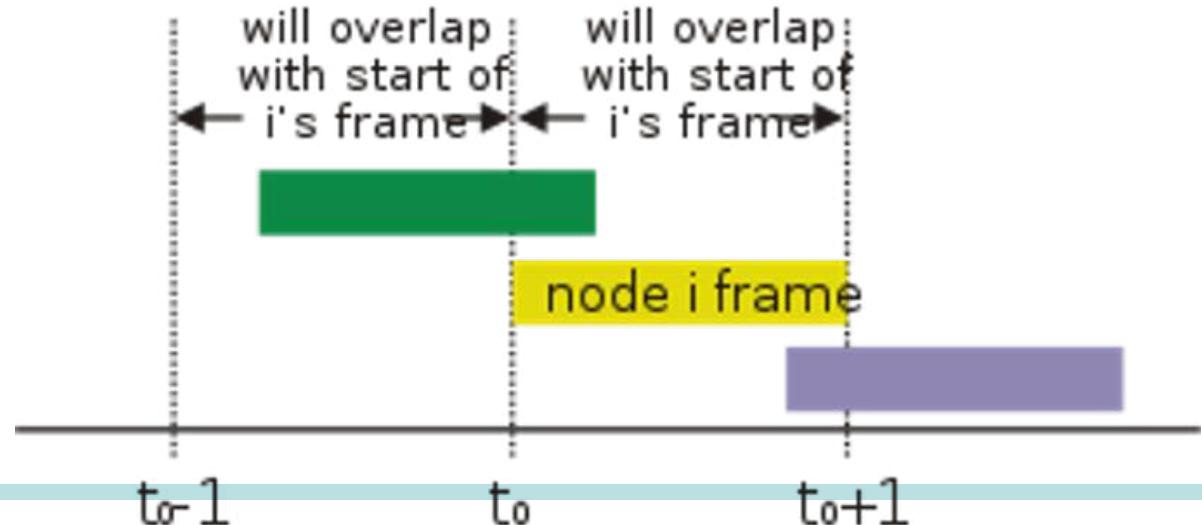
- In pure ALOHA, frames are transmitted at completely arbitrary times



5

Pure ALOHA

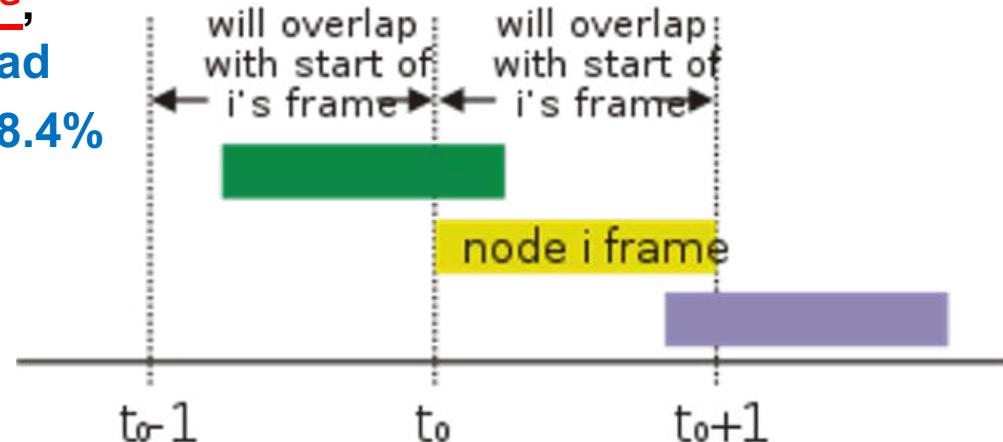
- **Unslotted Aloha: simpler, no synchronization**
- **When frame first arrives**
 - Transmit immediately
- **Collision probability increases:**
 - Frame sent at t_0 collides with other frames sent in $[t_0-1, t_0+1]$



Aloha Efficiency: Pure ALOHA

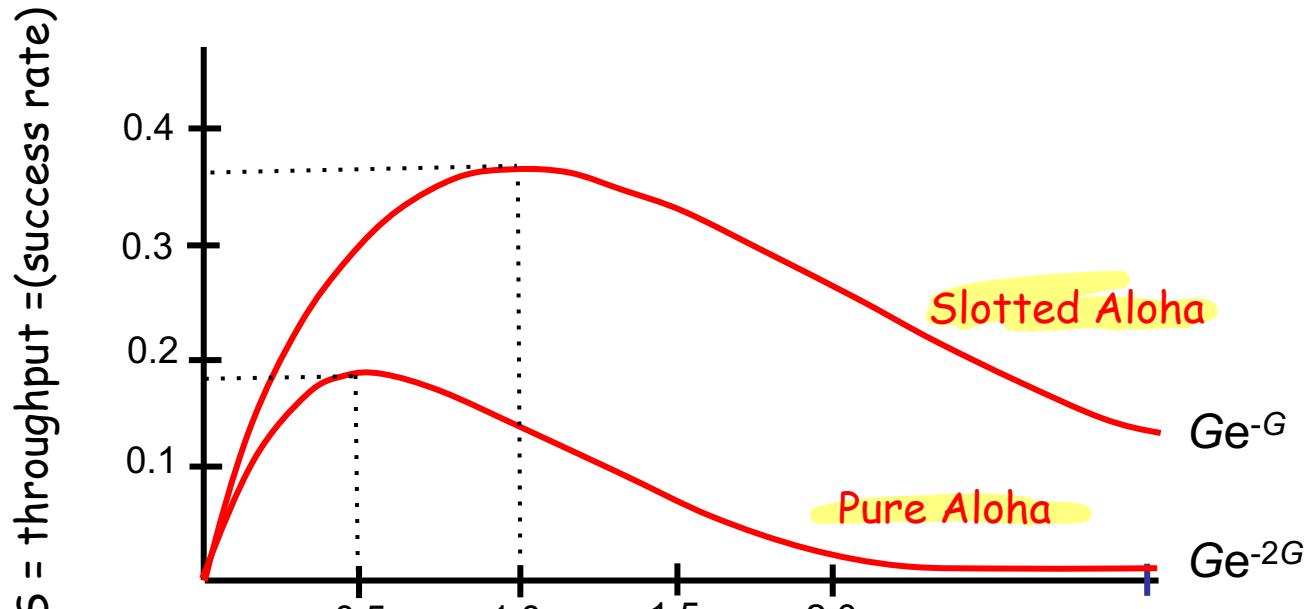
Pure Aloha: Partial transmission collision can occur (i.e., my 1st half of the transmission collides with your 2nd half)

- $\Pr(\text{success by given node}) = P(\text{node transmit}) * P(\text{no other node transmits in } [t_0-1, t_0]) * P(\text{no other node transmit in } [t_0, t_0+1])$
 $= p \times (1-p)^{(N-1)} \times (1-p)^{(N-1)}$
 $= p (1-p)^{(2N-2)}$
- So for the network, $\Pr(S) = N p (1-p)^{(2N-2)}$
- For very large N , $\Pr(S) = Ge^{-2G}$, where $G=Np$ is the offered load
- Therefore, $\Pr(S) \leq 1/(2e) = 18.4\%$



ALOHA Efficiency Comparison

Slotted Aloha vs Pure Aloha



G = offered load rate
= Expected total frames presented to the link per
the transmission time of a single frame

Carrier-Sense Multiple-Access (CSMA)

Carrier-Sense Multiple-Access

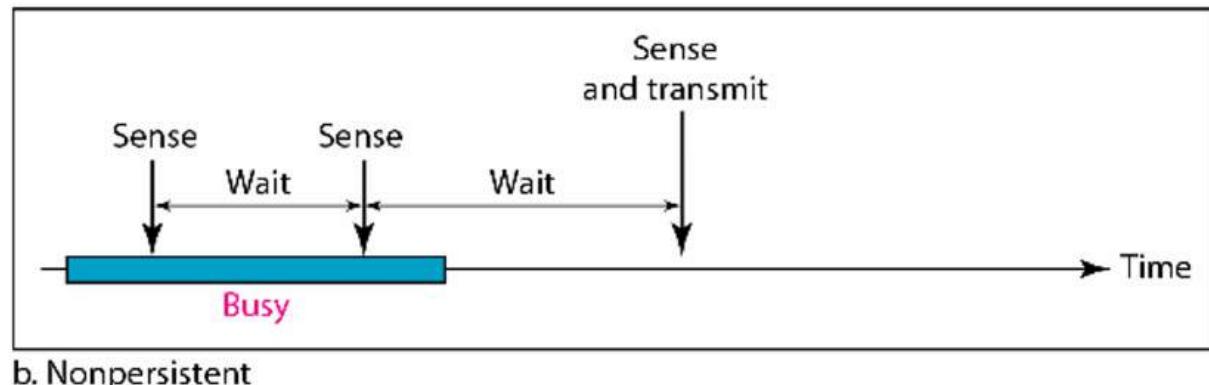
- To improve performance, avoid transmissions that are certain to cause collisions
- Based on the fact that in LAN propagation time is very small
 - If a frame was sent by a station, all stations know immediately so they can wait before start sending
 - A station with frames to be sent, should sense the medium for the presence of another transmission (carrier) before it starts its own transmission
- This can reduce the possibility of collision but it cannot eliminate it.
 - Collision can only happen when more than one station begin transmitting within a short time (the **propagation time** period)

CSMA Variants

- **Different CSMA protocols that determine:**
 - What a station should do when the medium is idle?
 - What a station should do when the medium is busy?
- **Three Types of CSMA Protocols**
 - Non-persistent CSMA
 - 1-Persistent CSMA
 - P-Persistent CSMA

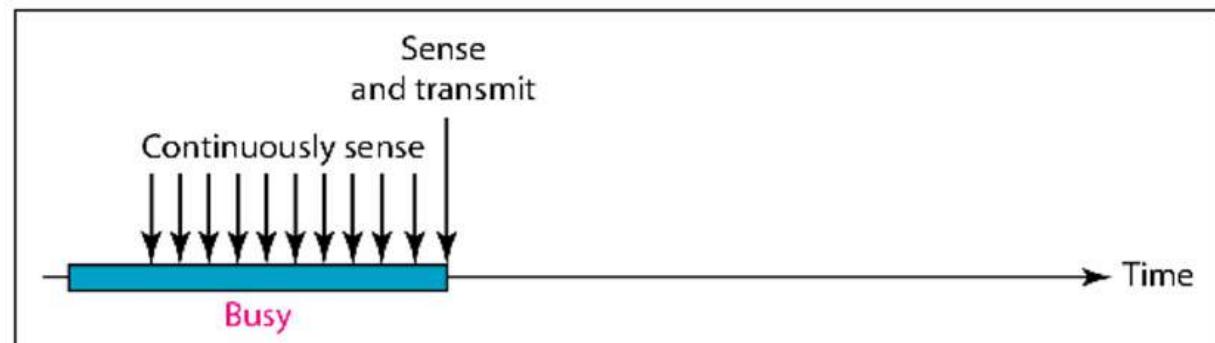
Non-persistent CSMA

- A station with frames to be sent, should sense the medium
 1. If medium is idle, transmit; otherwise, go to 2
 2. If medium is busy, (backoff) wait a *random amount of time* and repeat 1
- Non-persistent Stations are **deferential** (respect others)
- Performance:
 - Random delays reduces probability of collisions because two stations with data to be transmitted would wait for different amount of times.
 - Bandwidth is **wasted** if waiting time (backoff) is large because medium will remain idle following end of transmission even if one or more stations have frames to send



1-Persistent CSMA

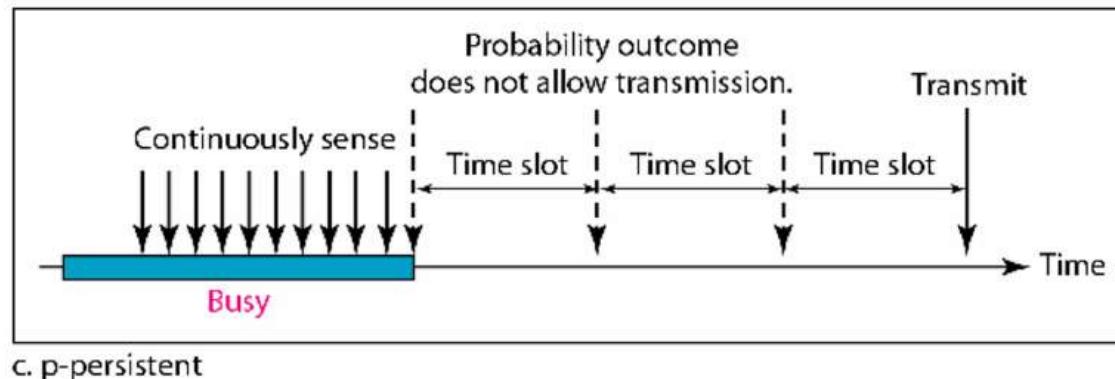
- To avoid idle channel time, 1-persistent protocol used
- Station wishing to transmit listens to the medium:
 - If medium idle, transmit immediately;
 - If medium busy, continuously listen until medium becomes idle; then transmit immediately with probability 1
- 1-persistent stations are **selfish**
- Performance
 - If two or more stations becomes ready at the same time, **collision guaranteed**



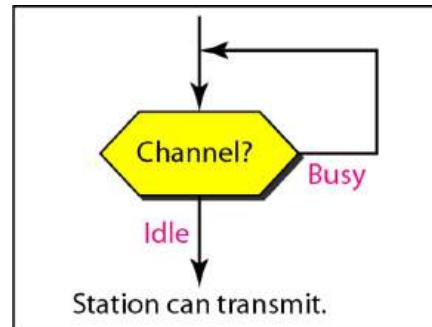
a. 1-persistent

P-Persistent CSMA

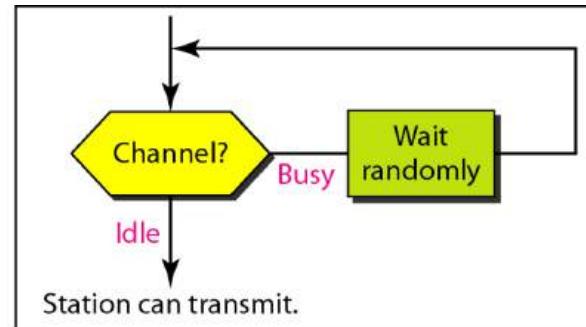
- Time is divided to slots where each time unit (slot) typically equals maximum propagation delay
- Station wishing to transmit listens to the medium:
 1. If medium idle,
 - transmit with probability (p), OR
 - wait one time unit (slot) with probability ($1 - p$), then repeat 1.
 2. If medium busy, continuously listen until idle and repeat step 1
- Performance (wise guy)
 - Reduces the possibility of collisions like non-persistent
 - Reduces channel idle time like 1-persistent



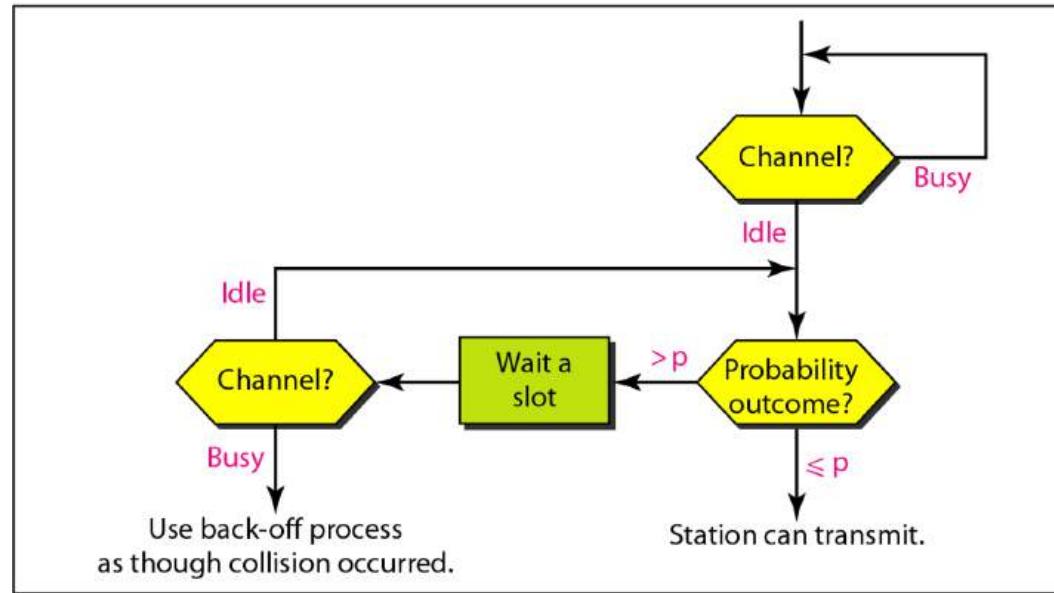
Flow Diagrams for CSMA



a. 1-persistent

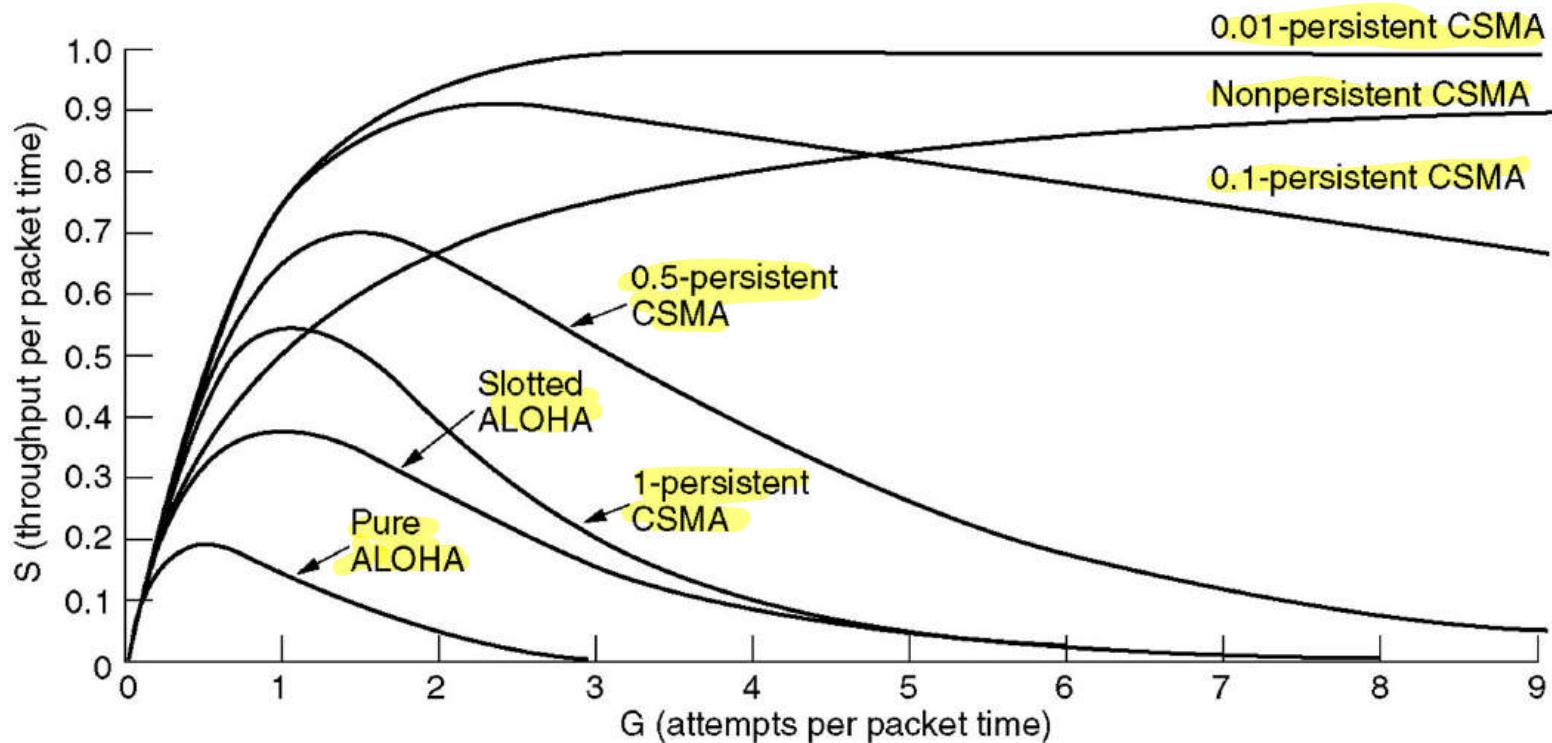


b. Nonpersistent



c. p-persistent

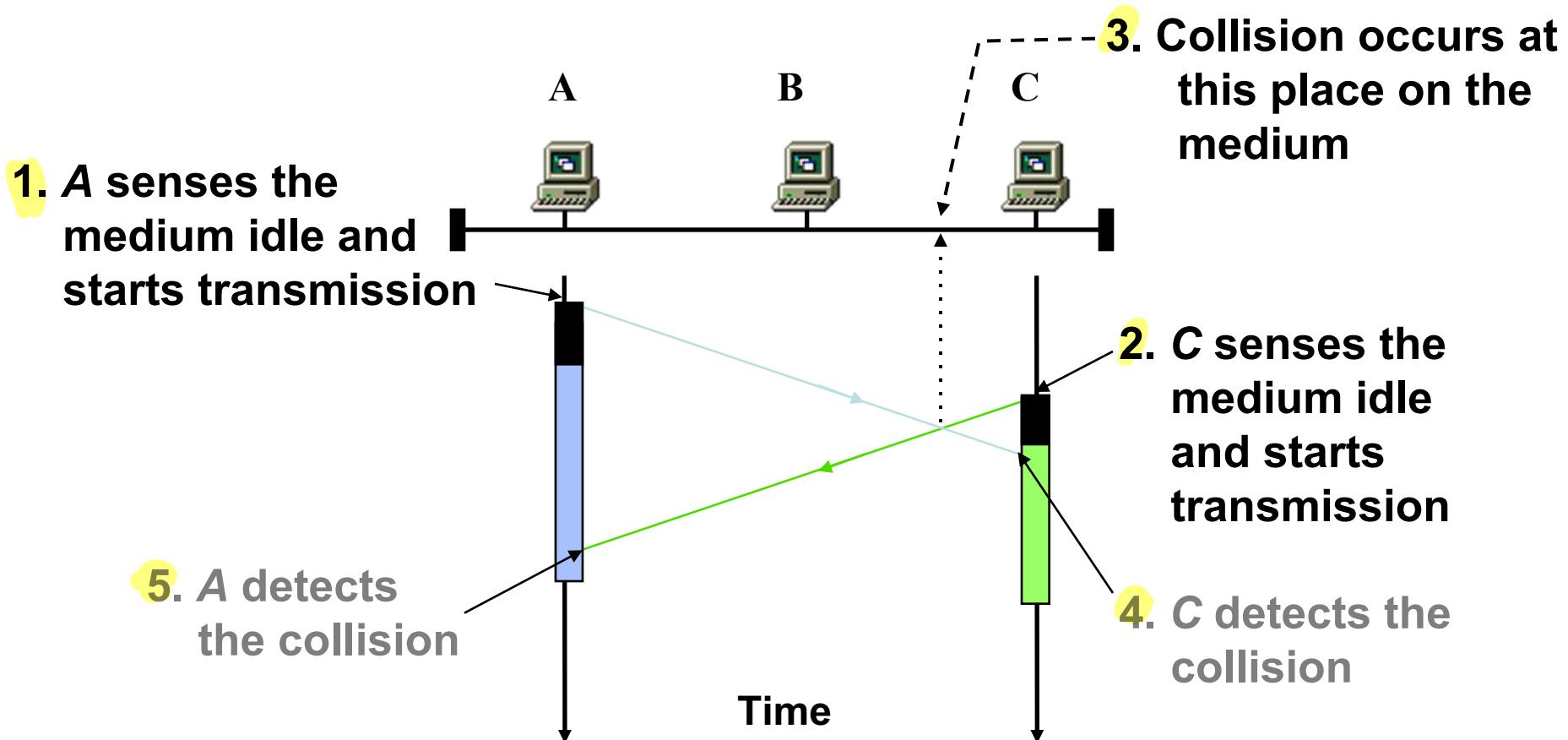
CSMA Efficiency



Comparison of the channel utilization versus load for various random access protocols.

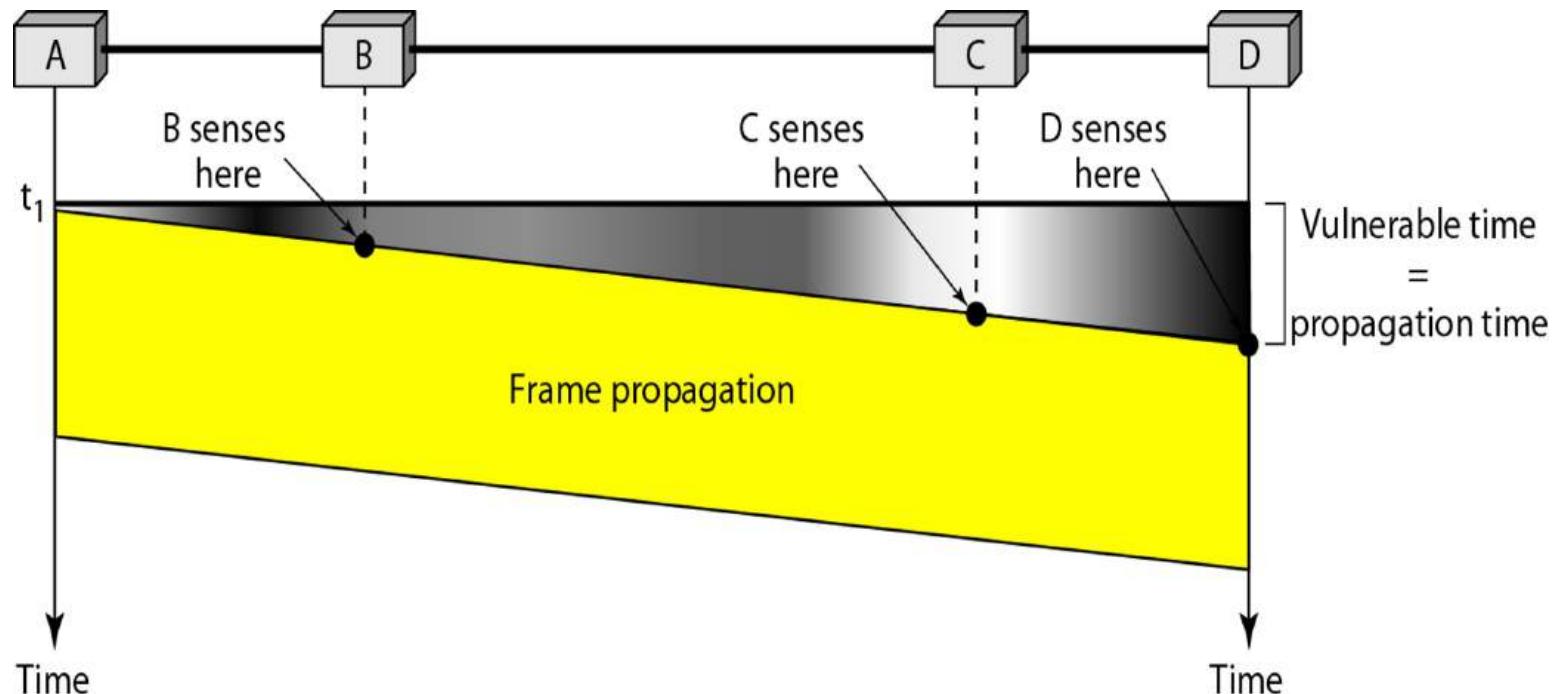
CSMA/CD Protocol

Collision in CSMA



Vulnerable Time in CSMA

- Vulnerable time for CSMA is the maximum propagation time
- The longer the propagation delay, the worse the performance of the protocol.



CSMA/CD (Collision Detection)

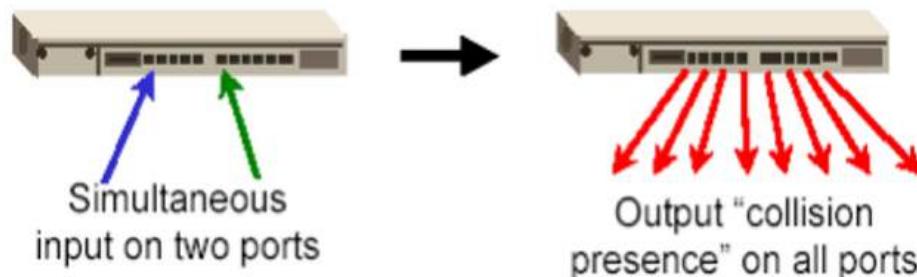
- **CSMA has channel wastage**
 - If a collision has occurred, colliding packets are still to be fully transmitted.
- **CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*) overcomes this:**
 - While transmitting, the sender is listening to medium for collisions.
 - Sender stops transmission if collision has occurred, reducing channel wastage.
- **CSMA/CD is widely used for bus topology LANs (IEEE 802.3, Ethernet)**

How to detect a Collision?

- **Transceiver**
 - A node monitors the media while transmitting. If the observed power is higher than the transmitted power of its own signal, it means collision occurred.



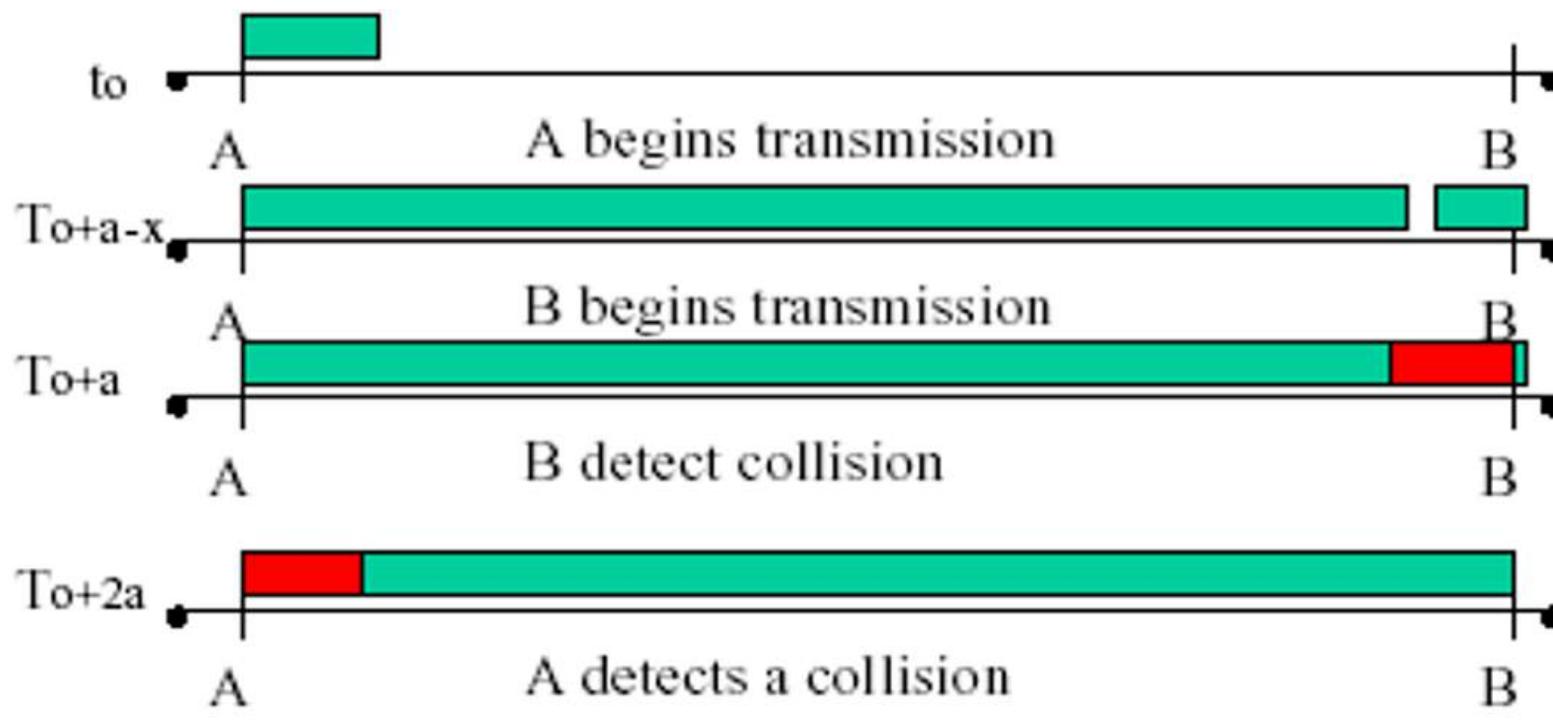
- **Hub**
 - If input occurs simultaneously on two ports, it indicates a collision. Hub send a collision presence signal on all ports.



Collision Detection

- **Question:** How long does it take to detect a collision?
- **Answer:** In the worst case, twice the maximum propagation delay of the medium

Note: a = maximum propagation delay



CSMA/CD Protocol

- **Transmission protocol**
 - Use one of the CSMA persistent algorithms
- **If a collision is detected by a station during its transmission, it should do the following**
 - Abort transmission, and
 - Transmit a *jam signal* (48 bits) to notify other stations of collision so that they will **discard the transmitted frame** also to make sure that the collision signal will stay until detected by the furthest station
 - After sending the *jam signal*, **backoff (wait) for a random amount of time**, then
 - Transmit the frame again

Learning Objectives

- **ALOHA Protocol**
 - Calculate throughput for ALOHA
 - Maximize throughput by differentiation
- **CSMA Protocol**
 - Protocol comparison for three flavors
- **CSMA/CD Protocol**
 - Maximum duration for collision detection

CSMA/CD (Collision Detection)

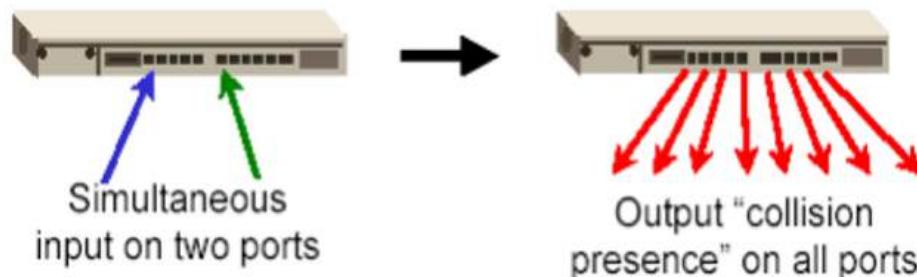
- **CSMA has channel wastage**
 - If a collision has occurred, colliding packets are still to be fully transmitted.
- **CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*) overcomes this:**
 - While transmitting, the sender is listening to medium for collisions.
 - Sender stops transmission if collision has occurred, reducing channel wastage.
- **CSMA/CD is widely used for bus topology LANs (IEEE 802.3, Ethernet)**

How to detect a Collision?

- **Transceiver**
 - A node monitors the media while transmitting. If the observed power is higher than the transmitted power of its own signal, it means collision occurred.



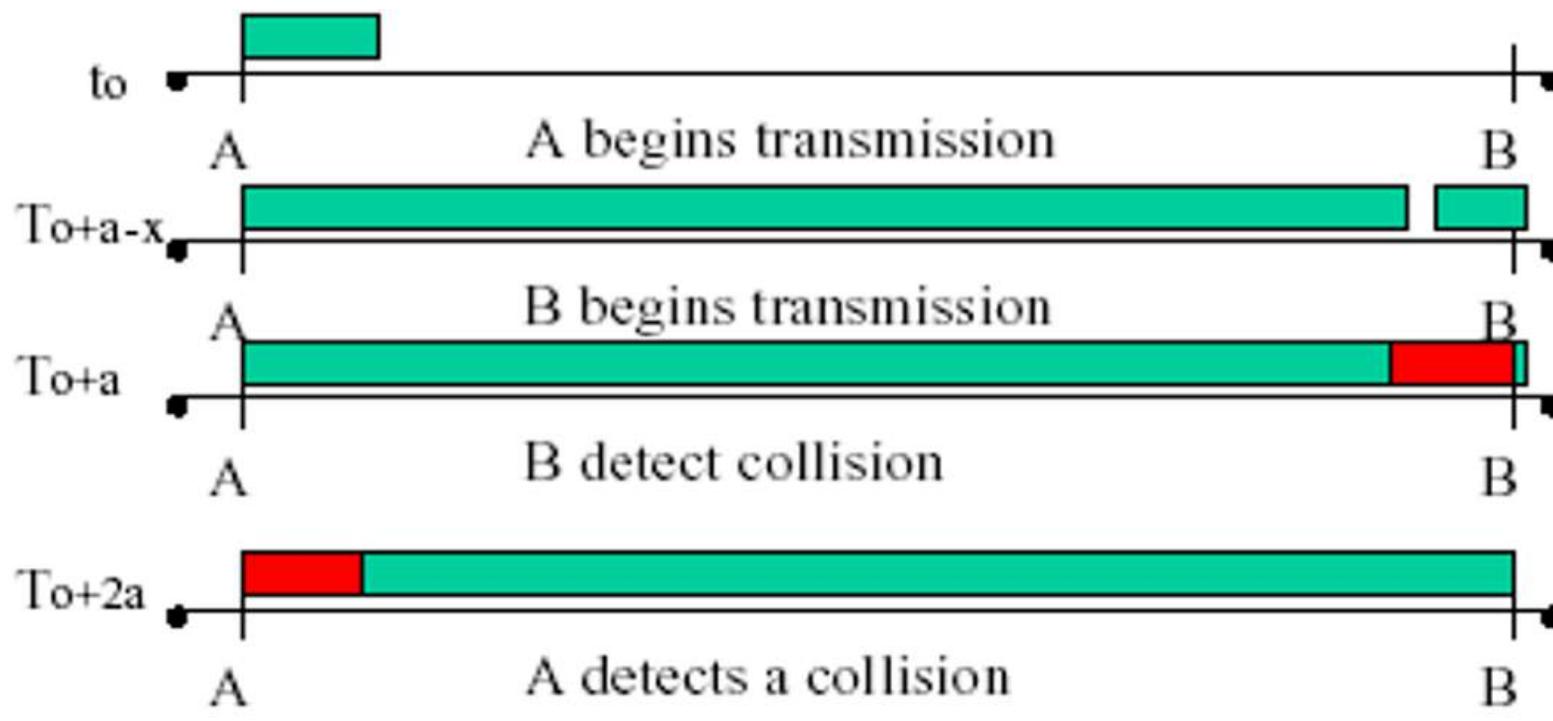
- **Hub**
 - If input occurs simultaneously on two ports, it indicates a collision. Hub send a collision presence signal on all ports.



Collision Detection

- **Question:** How long does it take to detect a collision?
- **Answer:** In the worst case, twice the maximum propagation delay of the medium

Note: a = maximum propagation delay



CSMA/CD Protocol

- **Transmission protocol**
 - Use one of the CSMA persistent algorithms
- **If a collision is detected by a station during its transmission, it should do the following**
 - Abort transmission, and
 - Transmit a *jam signal* (48 bits) to notify other stations of collision so that they will **discard the transmitted frame** also to make sure that the collision signal will stay until detected by the furthest station
 - After sending the *jam signal*, **backoff (wait) for a random amount of time**, then
 - Transmit the frame again

Learning Objectives

- **ALOHA Protocol**
 - Calculate throughput for ALOHA
 - Maximize throughput by differentiation
- **CSMA Protocol**
 - Protocol comparison for three flavors
- **CSMA/CD Protocol**
 - Maximum duration for collision detection

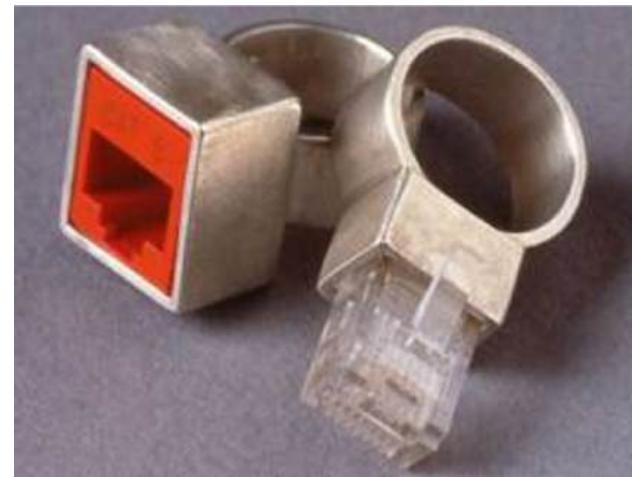
Additional Materials

- The related content talked today in
[https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:
 - Ethernet: Page 469 – Page 476
 - Frame: Page 537 – Page 541
- You can also find other video materials about
 - Ethernet <https://www.youtube.com/watch?v=MzhiVE6OuQA>
 - MAC Addresses <https://www.youtube.com/watch?v=FkiTOMn-XGw>
 - Frame Format <https://www.youtube.com/watch?v=jy4kBazJCKM>

Network Engineer's World



When we design our house



When we design our wedding rings

SC2008/CZ3006/CE3005

Computer Network

Lecture 7

Wired LAN: Ethernet



Contents

- **Ethernet Overview**
 - Ethernet Standard
 - Transmission Media
 - Physical Layer
 - Ethernet versions
- **Ethernet Frame Format**
 - Frame Format
 - MAC Address
- **Ethernet MAC Protocol**
 - Minimum Frame Size
 - Binary Exponential Backoff
- **Ethernet Evolution**
 - Bridged and Switched Ethernet
 - Fast, Gigabit and Ten-Gigabit Ethernet
 - Ethernet Pros and Cons

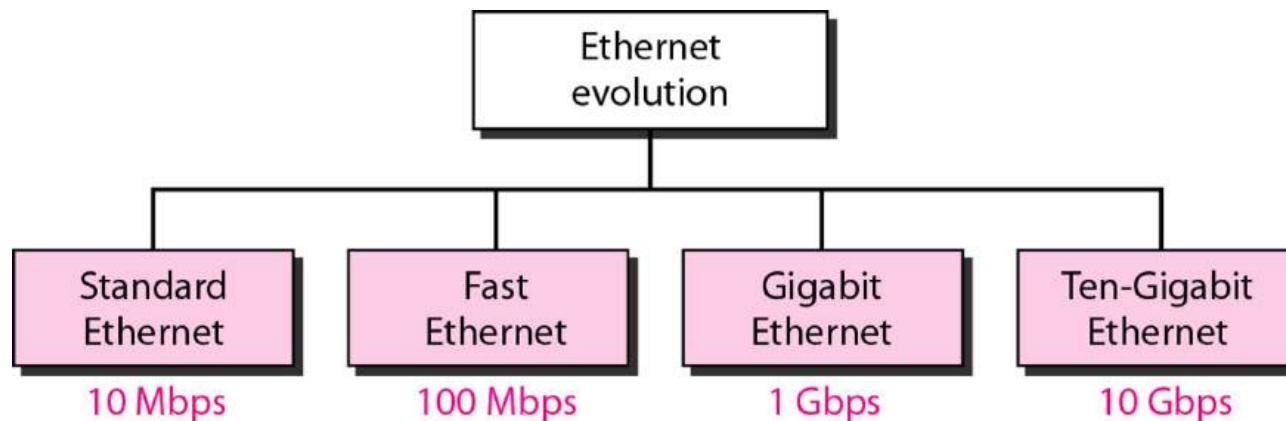
Ethernet Overview

Ethernet: A Brief History

not examinable

- **History**

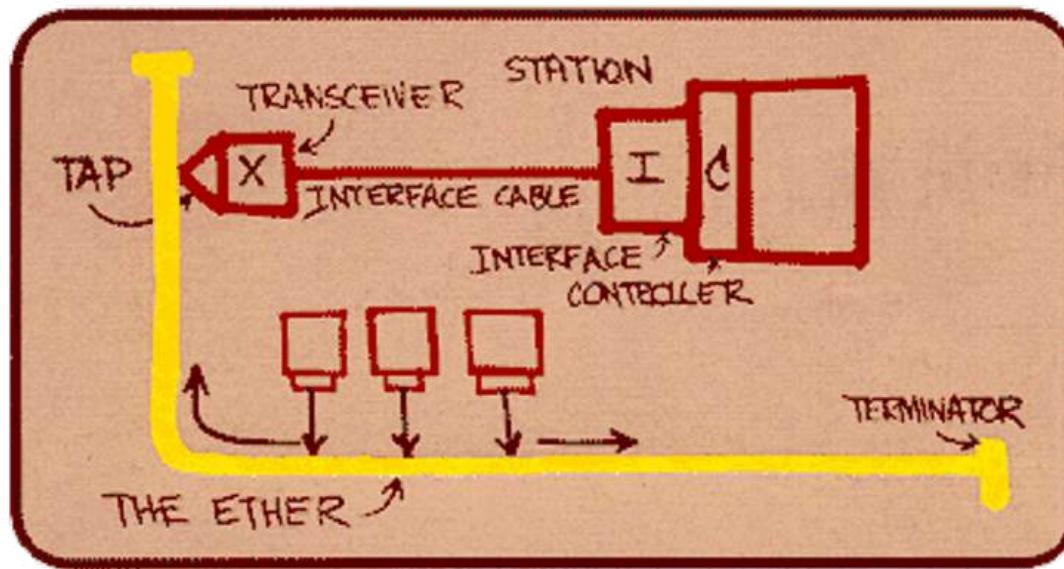
- Invented by *Bob Metcalfe* and others at Xerox PARC in mid-1970s
- Roots in Aloha packet-radio network
- Standardized by Xerox, DEC, and Intel in 1978
- LAN standards define physical and MAC layers
 - IEEE 802.3 (CSMA/CD - Ethernet) standard – originally 2Mbps
 - IEEE 802.3u standard for 100Mbps Ethernet
 - IEEE 802.3z standard for 1,000Mbps Ethernet



Ethernet Standard: Physical Layer

not examinable

- 802.3 standard defines both physical and MAC layer details
 - MAC Protocol: CSMA/CD
 - Physical layer:



Metcalfe's original Ethernet Sketch: The idea was first documented in a memo that Metcalfe wrote on May 22, 1973, where he named it after the disproven luminiferous ether as an "omnipresent, completely-passive medium for the propagation of electromagnetic waves".

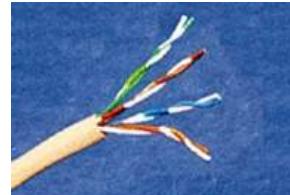
Ethernet Transmission Media

not examinable

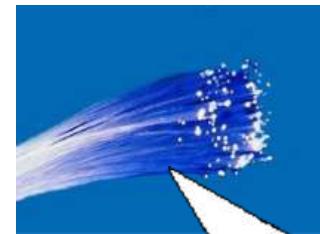
- **Coaxial cable**
 - Thick coax: Used by original Ethernet (bus topology)
 - Thin coax: More flexible, but shorter network span
- **Twisted pair:** For star topology, used with a hub, most commonly used
- **Optical fiber:** Expensive, Fragile, Difficult to handle, but high data rate. Used in backbone.



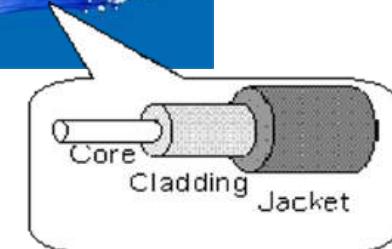
coax



twisted pair



optical
fiber



802.3 Physical Layer Configurations

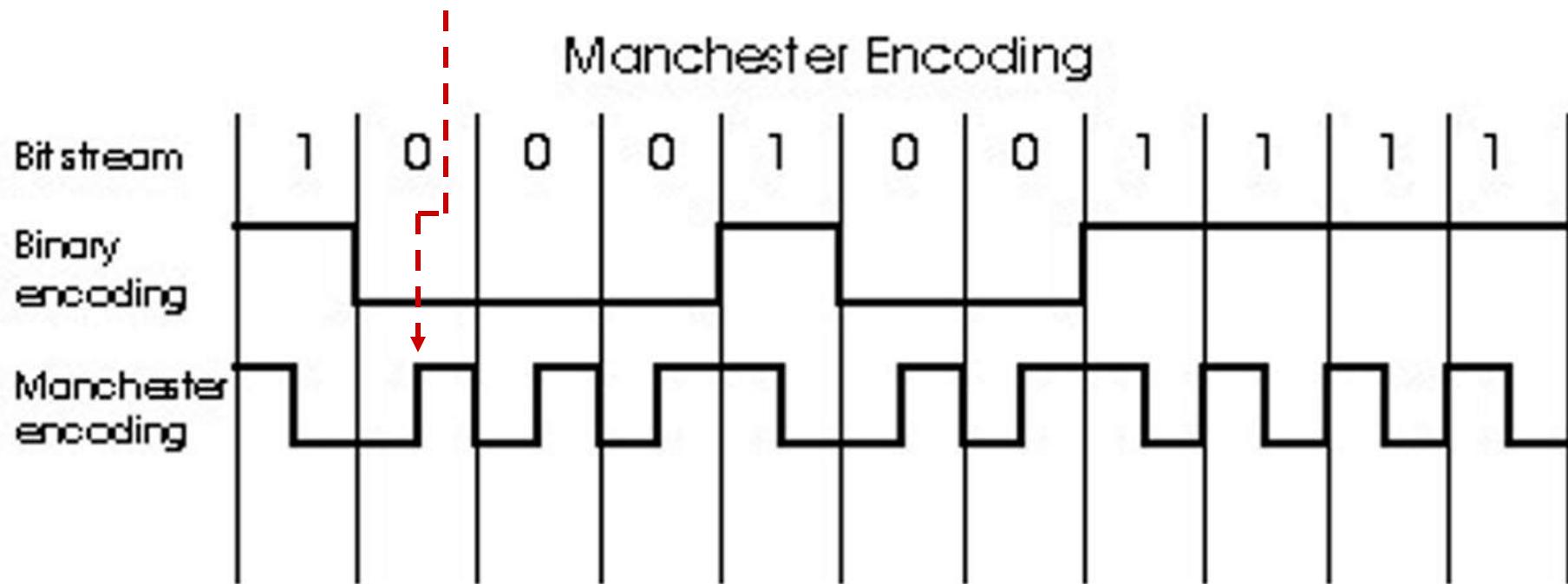
not examinable

- **Physical layer configurations are specified in three parts**
- **Data rate (10, 100, 1,000)**
 - 10, 100, 1,000Mbps
- **Signaling method (base, broad)**
 - Baseband: digital signaling
 - Broadband: analog signaling
- **Cabling (2, 5, T, F, S, L)**
 - 2 – 200m thin coax
 - 5 – 500m thick coax (original Ethernet cabling)
 - T – Twisted pair
 - F – Optical fiber
 - S – Short wave laser over multimode fiber
 - L – Long wave laser over single mode fiber

Baseband Manchester Encoding

not examinable

- Baseband here means that no carrier is modulated; instead bits are encoded using Manchester encoding and transmitted directly by modifying voltage of a DC signal.
- Manchester encoding ensures that a voltage transition occurs in each bit time which helps receiver in clock synchronization.



Ethernet Versions

not examinable

- Ethernet comes with several versions based on its network configurations. Important ones are:
 - 10BASE-5 (Original Ethernet)
 - 10BASE-2 (Cheapernet)
 - 10BASE-T (Star topology using a hub)
 - Others (eg 10BASE-FL, 10BASE-FP, etc)

Notation:

10 BASE 5

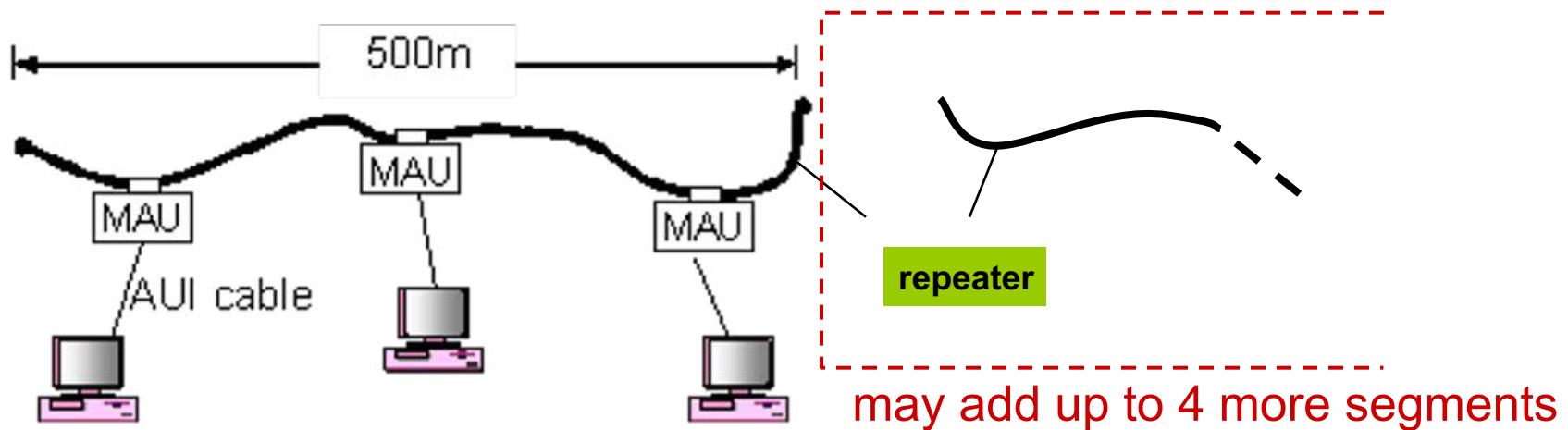
10 Mbps

Signal mode,
Base or Broad

Medium segment length (in 100m unit) or type. Possible options are: 5,2,T,F,etc

10BASE-5

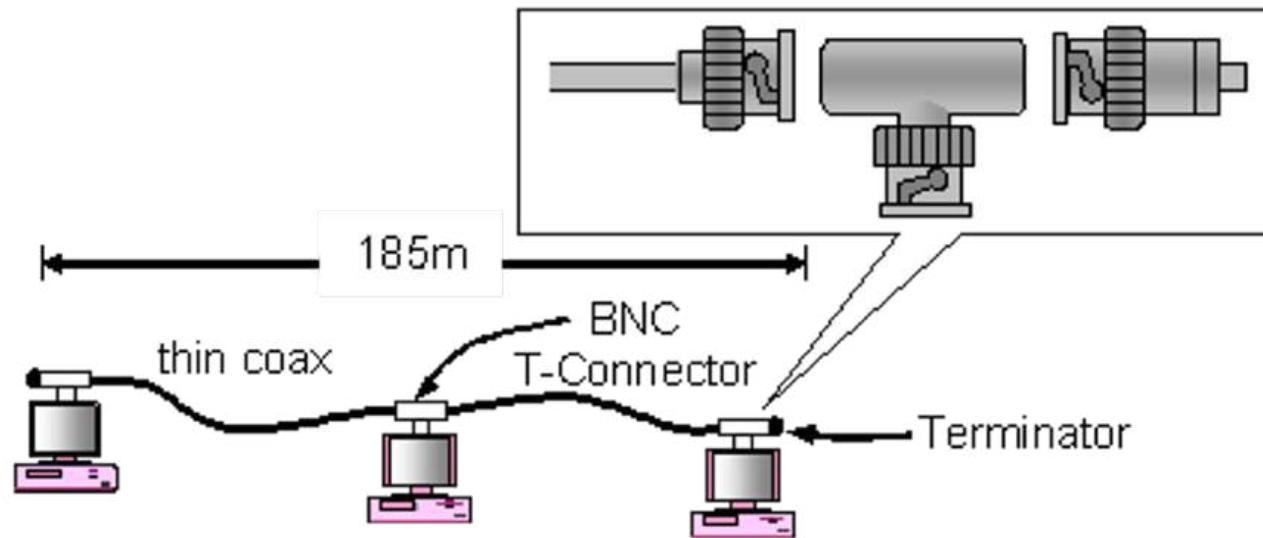
not examinable



- Original Ethernet design, thick coax (50ohm) is used
- Each segment is of 500m long (max)
- Four repeaters (max) can be used to connect up to 5 segments
- AUI cable connecting the PC and the thick coax cannot be longer than 50m

10BASE-2

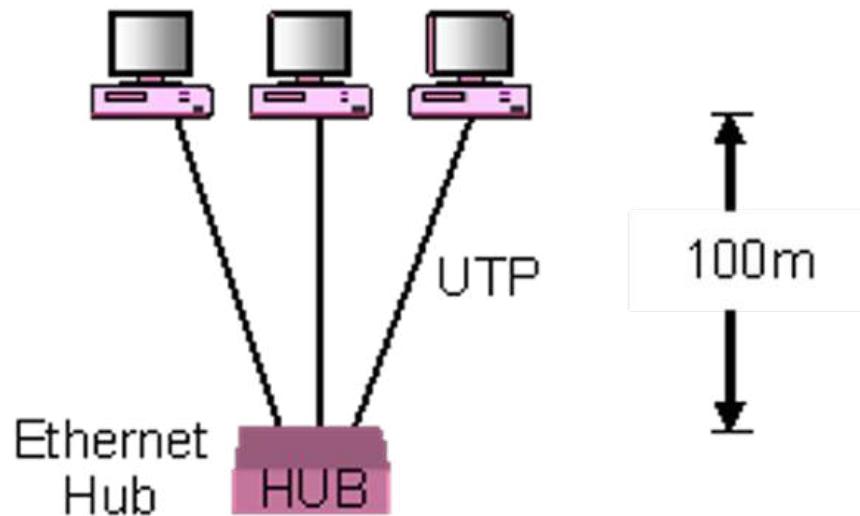
not examinable



- Called cheapernet because it is cheap to deploy. Thin coax is used
- Each segment is of 185m long (max), min cable length between two computers is 0.5m, max 30 nodes
- Up to 4 repeaters, (so entire network cable=925m)

10BASE-T

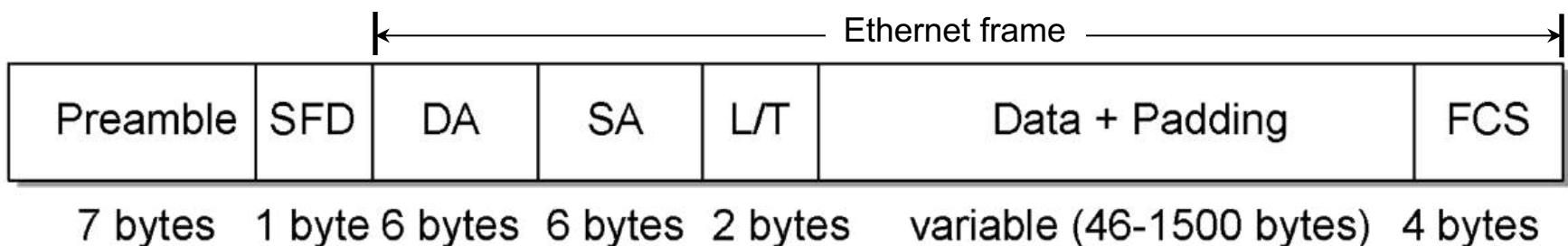
not examinable



- 'T' in 10BASE-T stands for Twisted pairs (UTP: unshielded twisted pairs). At least two pairs of wires.
- Most popular Ethernet option
- **Cable length between a hub & a computer = 100m**

Ethernet Frame Format

Ethernet Frame Format



SFD = Start Frame Delimiter

L/T = Length/Type

SA = Source Address

FCS = Frame Check Sequence

DA = Destination Address

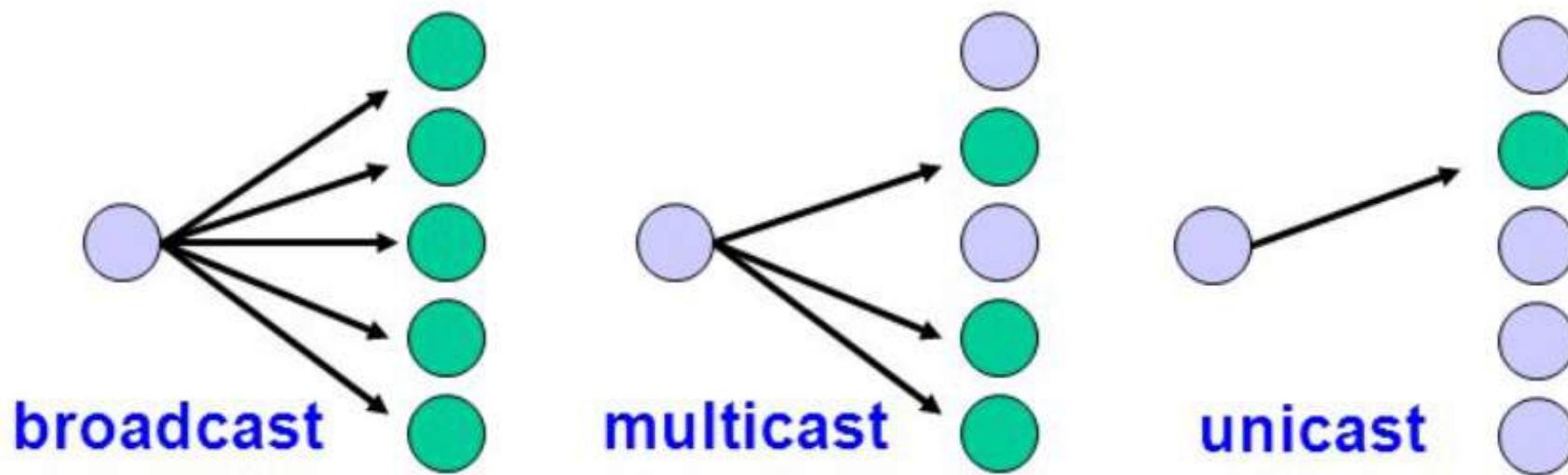
- **Preamble & SFD:** 7 bytes with the pattern 10101010 followed by one byte with the pattern 10101011; used for synchronizing receiver to sender clock (clocks are never exact)
- **Pad field** allows additional dummy data to be included to a frame for min. frame size requirement.
- **Frame Check Sequence** field enables error detection using CRC. It covers between Source Address and Pad fields.

Ethernet Frames

- **Preamble is a sequence of 7 bytes, each set to “10101010”**
 - Used to synchronize receiver before actual data is sent
- **Addresses**
 - unique, 48-bit unicast address assigned to each adapter
 - example: 08 : 00 : e4 : b1 : 02 : a2
 - Each manufacturer gets their own address range
 - broadcast: all 1s
 - multicast: first bit (from right) of the first byte is 1
 - unicast: first bit (from right) of the first byte is 0
- **Type field is a demultiplexing key used to determine which higher level protocol the frame should be delivered to**
- **Body can contain up to 1500 bytes of data**

Broadcast, Multicast, Unicast

- **Broadcast: One to All**
- **Multicast: One to Many**
- **Unicast: One to One**



MAC Address Examples

Question

Define the type of the following destination addresses:

- a. $4A:30:10:21:10:1A$
- b. $47:20:1B:2E:08:EE$
- c. $FF:FF:FF:FF:FF:FF$

Solution

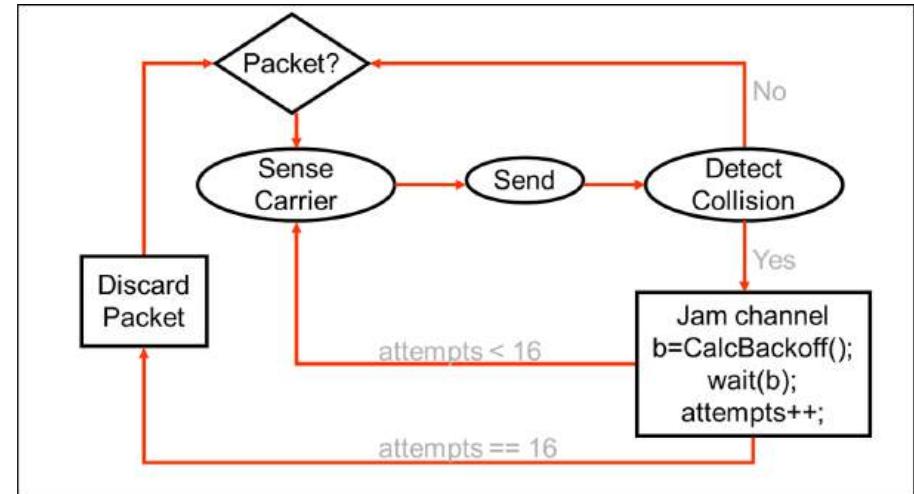
To find the type of the address, we need to look at the **second hexadecimal digit from the left**. If it is even, the address is unicast. If it is odd, the address is multicast. If all digits are F's, the address is broadcast. Therefore, we have the following:

- a. This is a unicast address because A in binary is 1010.
- b. This is a multicast address because 7 in binary is 0111.
- c. This is a broadcast address because all digits are F's.

Ethernet's MAC

Ethernet's MAC Algorithm

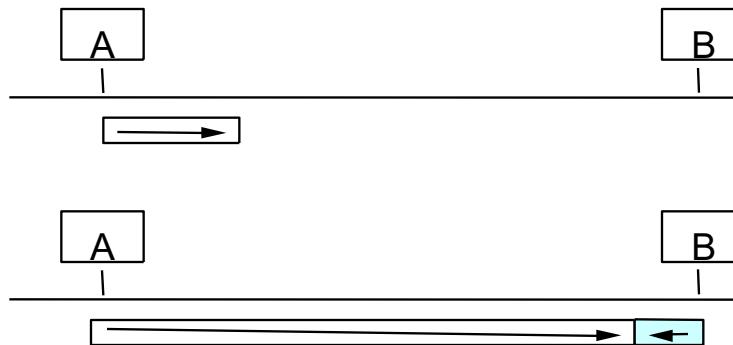
- **Ethernet uses CSMA/CD**
 - Listens to line before/during sending
- **If line is idle (no carrier sensed)**
 - Send packet immediately
 - Upper bound message size of 1500 bytes
 - Must wait 9.6us between back-to-back frames
- **If line is busy (carrier sensed)**
 - Wait until idle and transmit packet immediately
 - called *1-persistent* sending
- **If collision detected**
 - Stop sending and jam signal
 - Try again later



Frame Collisions

- **Collisions are caused when two adaptors transmit at the same time (adaptors sense collision based on voltage differences)**
 - Both found line to be idle
 - Both had been waiting for a busy line to become idle

A starts
at
time 0



Message almost
there at time T
when
B starts –
collision!

How can we be sure A knows about the collision?

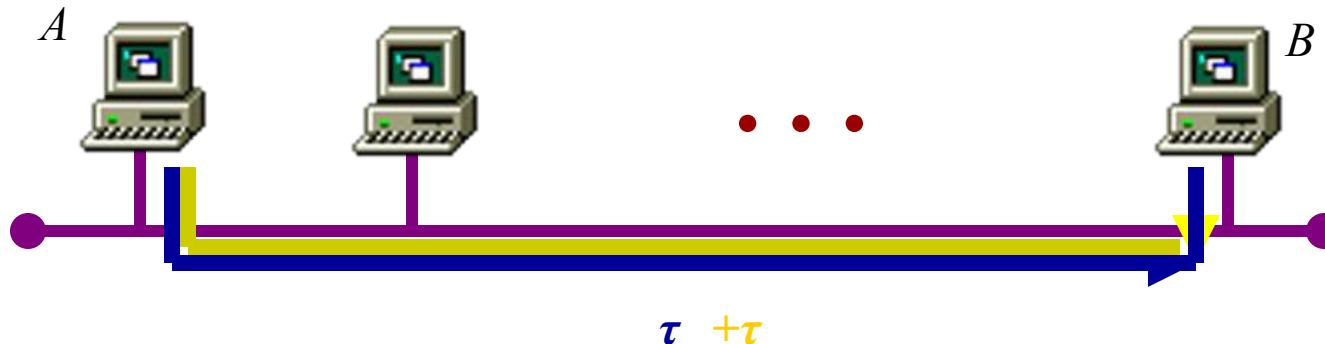
Ethernet Minimum Frame Size

So, a sender must keep its transmission until it is sure that there are no other transmissions on the medium it is not aware of before it ends its transmission.

But then **how long to keep its transmission?**

A transmission requires an end-to-end signal propagation time (τ) to reach all stations (the longest), then a potential collision requires τ unit of time to return to the sender.

So the frame transmission $\geq 2\tau$; i.e., signal round trip time.

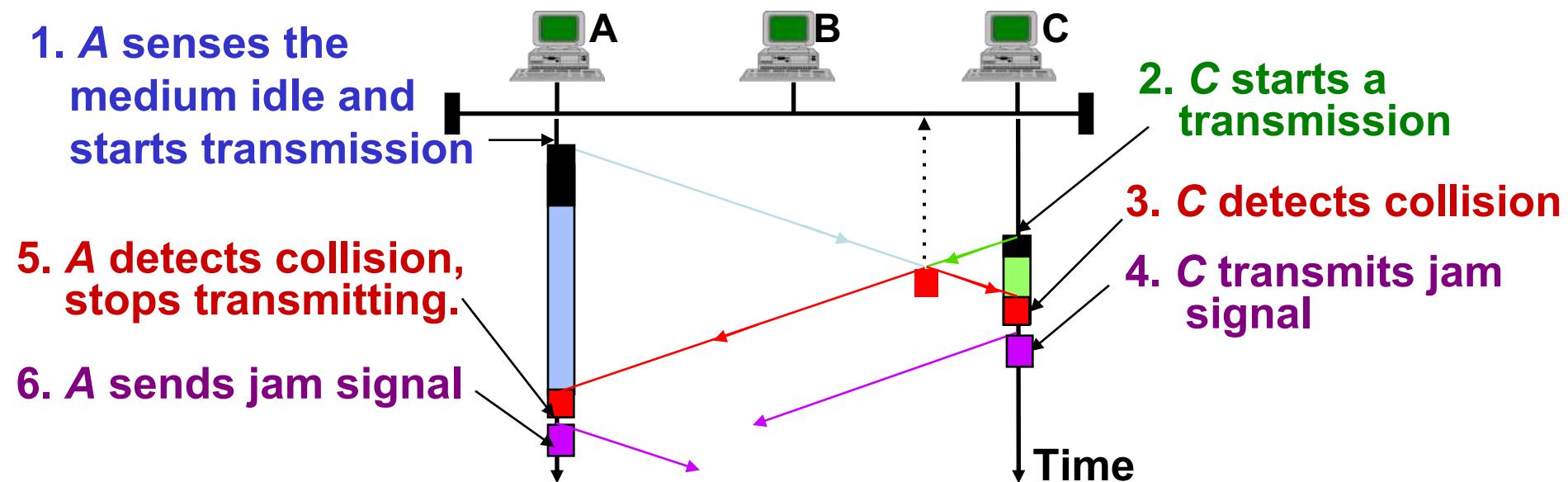


Collision Detection

- **How can A know that a collision has taken place?**
 - Must have a mechanism to ensure retransmission on collision
 - A's message reaches B at time T
 - B's message reaches A at time 2T
 - So, A must still be transmitting at 2T
- **IEEE 802.3 specifies max value of 2T to be 51.2us**
 - This relates to maximum distance of 2500m between hosts
 - At 10Mbps it takes 0.1us to transmit one bit so 512 bits (64B) take 51.2us to send
 - So, Ethernet frames must be at least 64B long
 - 14B header, 46B data, 4B CRC
 - Padding is used if data is less than 46B
- **Send jamming signal after collision is detected to ensure all hosts see collision**
 - 48 bit signal

Ethernet: Jam Signals

- In the previous slide, B's transmission cannot be too brief, otherwise A might not see the collision
- To avoid this, a station is required to transmit a jam sequence (32 to 48 bits long) after it has detected a collision. This will make the collision more obvious.



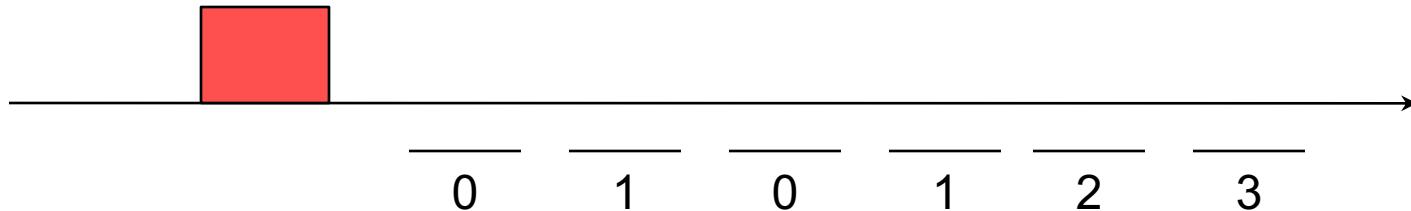
Binary Exponential Backoff (BEB)

- What should stations do when a collision is detected?
 - Discard the packet and let upper layer protocols do the retransmission as in Aloha, slotted Aloha, and CSMA ( NO!!!!). But how?
 - They can choose a future slot randomly to retransmit based on *Binary Exponential Backoff* (BEB).
- Delay time is selected using binary exponential backoff
 - 1st time: choose K from {0,1} then delay = K * 51.2us
 - 2nd time: choose K from {0,1,2,3} then delay = K * 51.2us
 - nth time: delay = K x 51.2us, for K from {0,1,... 2ⁿ – 1}
 - Note max value for K = 1023
 - give up after several tries (usually 16)
 - Report transmit error to host
- If delay were not random, then there is a chance that sources would retransmit in lock step
- Why not just choose from small set for K
 - This works fine for a small number of hosts
 - Large number of nodes would result in more collisions

probing

BEB: Example

Channel



A



B



1st Retrial

$$P(\text{Collision}) = P(A \& B=0) + P(A \& B=1)$$

$$\begin{aligned} P(A \& B=0) &= P(A=0) * P(B=0) \\ &= 0.5 * 0.5 = 0.25 \end{aligned}$$

$$P(\text{Collision}) = 0.25 + 0.25 = 0.5$$

2nd Retrial

$$\begin{aligned} P(\text{Collision}) &= P(A \& B=0) + P(A \& B=1) \\ &\quad + P(A \& B=2) + P(A \& B=3) \end{aligned}$$

$$\begin{aligned} P(A \& B=0) &= P(A=0) * P(B=0) \\ &= 0.25 * 0.25 = 0.0625 \end{aligned}$$

$$P(\text{Collision}) = 4 * 0.0625 = 0.25$$

MAC Algorithm from Receiver

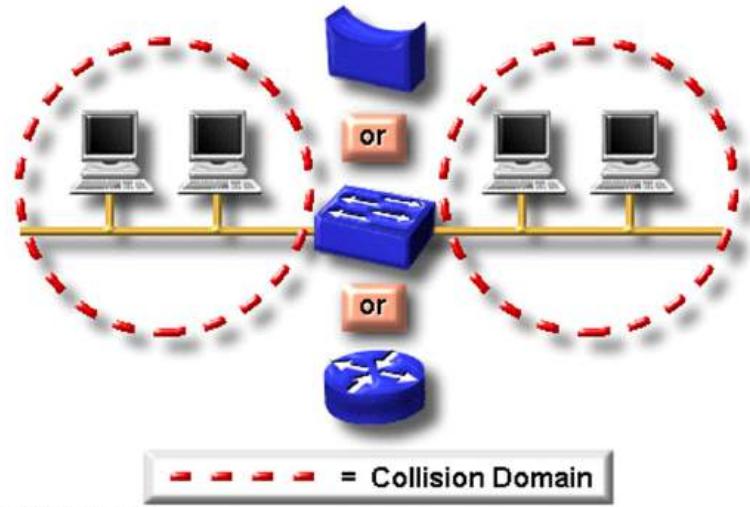
- **Senders handle all access control**
- **Receivers simply read frames with acceptable address**
 - Address to host
 - Address to broadcast
 - Address to multicast to which host belongs
 - All frames if host is in **promiscuous** mode

Ethernet Evolutions

Collision Domain

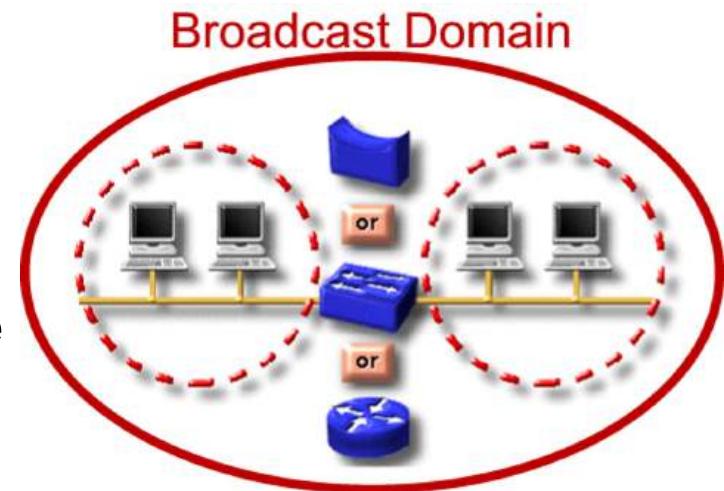
- **Network region in which collisions are propagated.**
 - Repeaters and hubs propagate collisions.
 - Bridges, switches and routers do not.
- **Collision frequency can be kept low by breaking the network into segments bounded by:**
 - bridges
 - switches
 - routers

Separating Collision Domains



Broadcast Domain

- **Network region in which broadcast frames are propagated.**
 - Repeaters, hubs, bridges, & switches propagate broadcasts. (**Layer 2**)
 - Routers either do or don't, depending on their configuration. (**Layer 3**)
- **Broadcasts are necessary for network function.**
- **Some devices and protocols produce lots of broadcasts; avoid them.**
- **Broadcast frequency can be kept manageable by limiting the LAN size.**
- **LANs can then be cross-connected by routers to make a larger internetwork.**



Bridged Ethernet

Domain



a. Without bridging

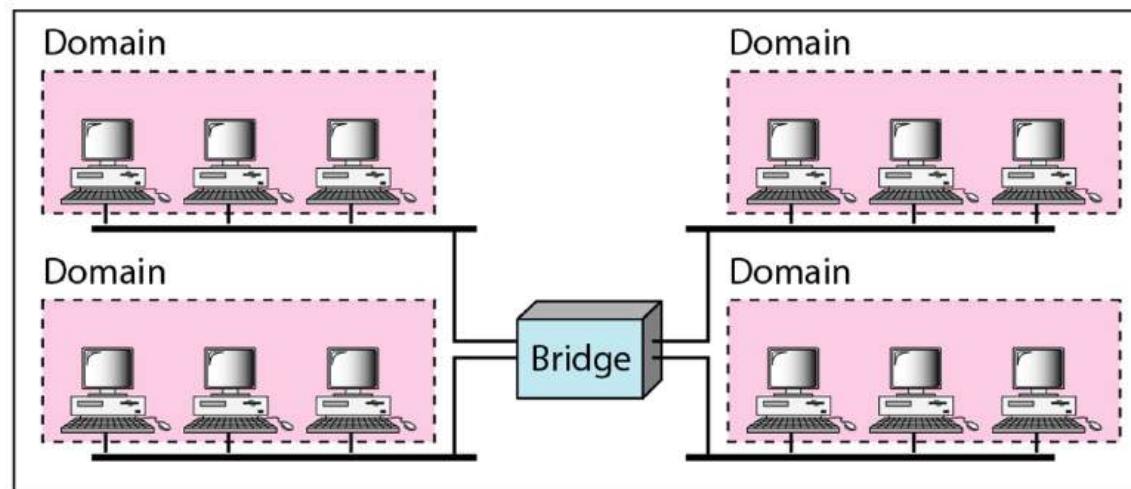
Domain

Domain

Domain

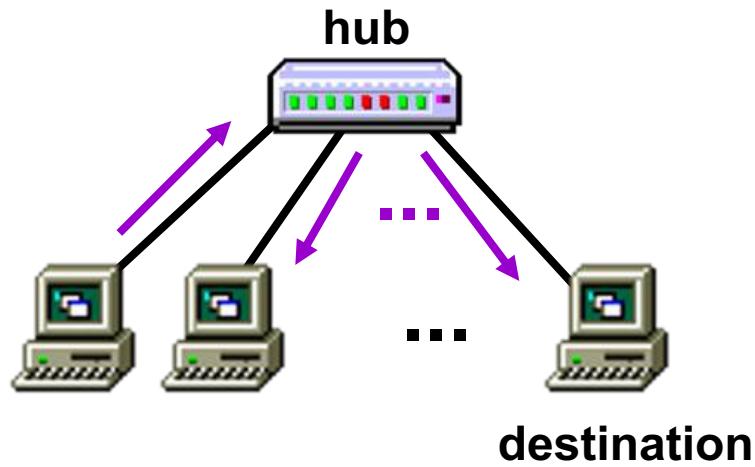
Domain

Bridge



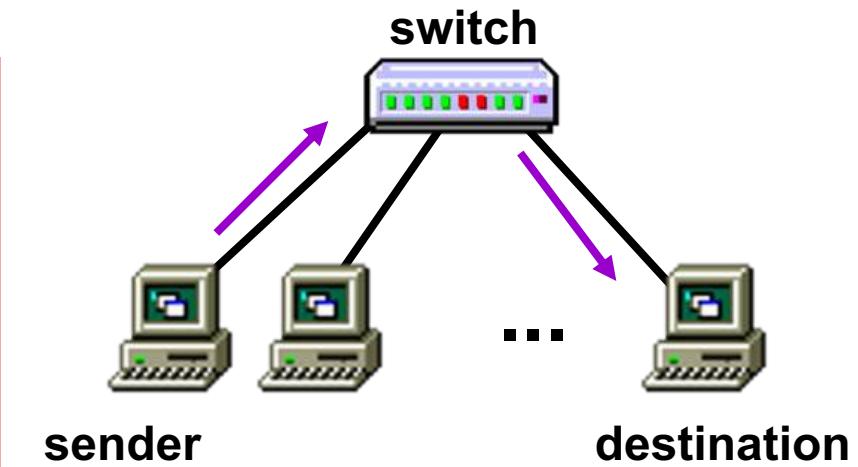
b. With bridging

Shared versus Switched



A repeater (or a hub) forwards the received signals to all output ports except the incoming port.

A collision occurs when two computers transmit at the same time. In this case, the channel carries no useful information.



A switch (or a switching hub) forwards the received signals only to the destination.

When two transmissions arrive at the switch at the same time, they will be stored in different buffers so that their frames can be forwarded later. No collision is resulted.

Switched Ethernet

not examinable

- **Switches forward and filter frames based on LAN addresses**
 - It's not a bus or a router (simple forwarding tables are maintained)
- **Very scalable**
 - Options for many interfaces
 - Full duplex operation (send/receive frames simultaneously)
- **Connect two or more “segments” by copying data frames between them**
 - Switches only copy data when needed
 - key difference from repeaters
- **Higher link bandwidth**
 - Collisions are completely avoided
- **Much greater aggregate bandwidth**
 - Separate segments can send at once

Fast Ethernet (FE)

not examinable

- **Fast Ethernet (100Mbps) has technology very similar to 10Mbps Ethernet**
 - Uses different physical layer encoding (4B5B)
 - Many NIC's are 10/100 capable
 - Can be used at either speed
- **Summary of Fast Ethernet Implementation**

<i>Characteristics</i>	<i>100Base-TX</i>	<i>100Base-FX</i>	<i>100Base-T4</i>
Media	Cat 5 UTP or STP	Fiber	Cat 4 UTP
Number of wires	2	2	4
Maximum length	100 m	100 m	100 m
Block encoding	4B/5B	4B/5B	
Line encoding	MLT-3	NRZ-I	8B/6T

Gigabit Ethernet (GbE)

not examinable

- **Gigabit Ethernet (1,000Mbps)**
 - Compatible with lower speeds
 - Uses standard framing and CSMA/CD algorithm
 - Distances are severely limited
 - Used for backbones and inter-router connectivity
 - Cost competitive

<i>Characteristics</i>	<i>1000Base-SX</i>	<i>1000Base-LX</i>	<i>1000Base-CX</i>	<i>1000Base-T</i>
Media	Fiber short-wave	Fiber long-wave	STP	Cat 5 UTP
Number of wires	2	2	2	4
Maximum length	550 m	5000 m	25 m	100 m
Block encoding	8B/10B	8B/10B	8B/10B	
Line encoding	NRZ	NRZ	NRZ	4D-PAM5

Ten-Gigabit Ethernet (10GbE)

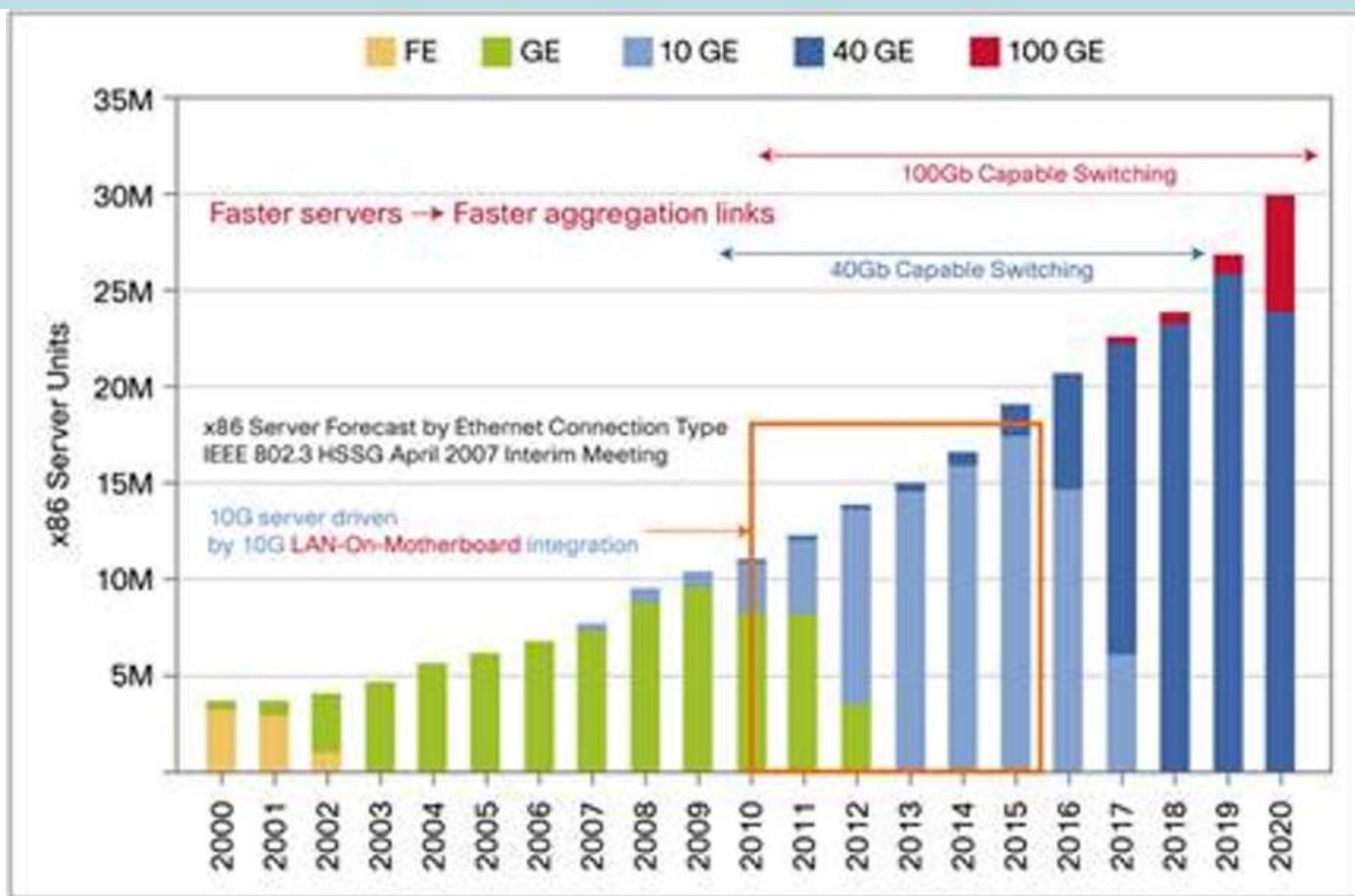
not examinable

- **Ten-Gigabit Ethernet (10Gbps)**
 - Defined by IEEE 802.3ae-2002
 - Higher-grade copper cables required: cat 6a or Class F/Cat 7 cables for links up to 100m

Characteristics	10GBase-S	10GBase-L	10GBase-E
Media	Short-wave 850-nm multimode	Long-wave 1310-nm single mode	Extended 1550-mm single mode
Maximum length	300 m	10 km	40 km

Ethernet Adoption Trends

not examinable



http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_paper_c11-696667.html

Experiences with Ethernet

not examinable

- **Ethernets work best under light loads**
 - Utilization over 30% is considered heavy
 - Network capacity is wasted by collisions
- **Most networks are limited to about 200 hosts**
 - Specification allows for up to 1024
- **Most networks are much shorter**
 - 5 to 10 microsecond RTT
- **Transport level flow control helps reduce load (number of back to back packets)**
- **Ethernet is inexpensive, fast and easy to administer!**

Ethernet Problems

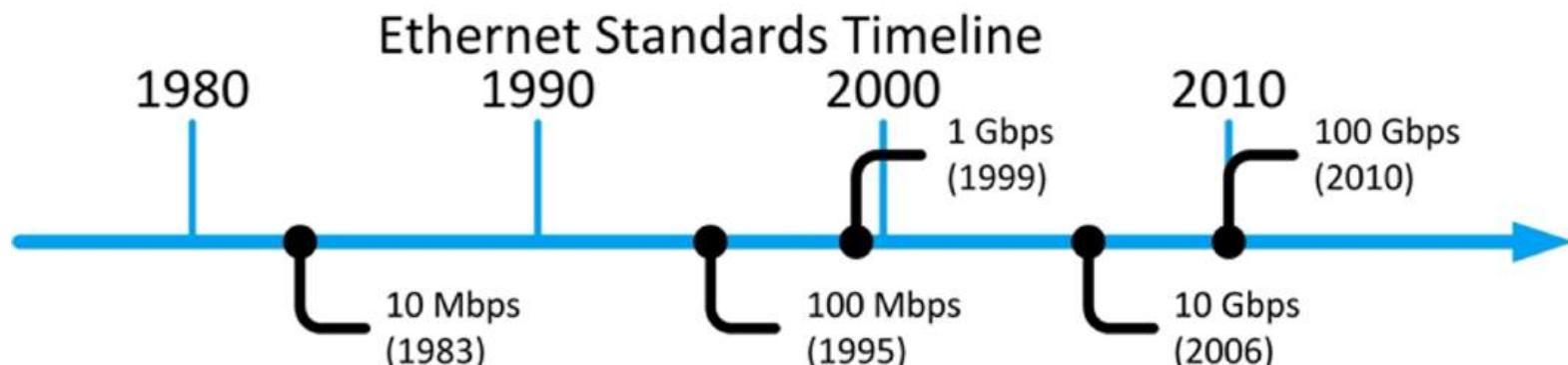
not examinable

- **Ethernet's peak utilization is pretty low (like Aloha)**
- **Peak throughput worsens with**
 - More hosts
 - More collisions
 - Longer links
 - Collisions take longer to observe, more wasted bandwidth
- **Efficiency can be improved by avoiding the above conditions**

Why did Ethernet win?

not examinable

- There are many LAN protocols (ARCNET, Token ring, AppleTalk, etc)
 - Price
 - Performance
 - Availability
 - Ease of use
 - Scalability



Learning Objectives

- **Ethernet Overview**
 - Read Ethernet versions **not examinable**
- **Ethernet Frame Format**
 - Understand MAC address: unicast/broadcast **examinable**
- **Ethernet MAC Protocols**
 - Calculate minimum frame size **examinable**
the frame transmission $\geq 2T$; i.e., signal round trip time.
 - Calculate collision rate under BEB scheme **examinable**
- **Ethernet Evolutions**
 - Count collision/broadcast domains **examinable**
 - FE/GbE/10GbE **not examinable**

Additional Materials

- The related content talked today in [https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:
 - Physical Media: Page 18 - Page 22
 - MAC Protocols: Page 445 - Page 449
 - ALOHA Protocols: Page 450 - Page 453
 - CSMA and CSMA/CD Protocols: Page 453- Page 459
- You can also find other video materials about
 - Physical Layer <https://www.youtube.com/watch?v=rKzDbdGhcdY>
 - CSMA <https://www.youtube.com/watch?v=MAZi6VoekYw>
 - Controlled Access Protocol-Reservation <https://www.youtube.com/watch?v=baaPXiQ44vs>

What is the problem with the guy?



SC2008/CZ3006/CE3005

Computer Network

Lecture 8

Wireless LAN: IEEE 802.11



Contents

- **WLAN Overview**
 - WLAN Standard
 - WLAN Architecture
 - WLAN Protocol Stack
- **802.11 Physical Layer**
- **802.11 MAC Layer**
 - Hidden and Exposed Terminal Problems
 - CSMA/CA Protocol
 - MAC Management
- **Multi-Access Reservation Protocol**
 - Scheme
 - Throughput Calculation

WLAN Overview

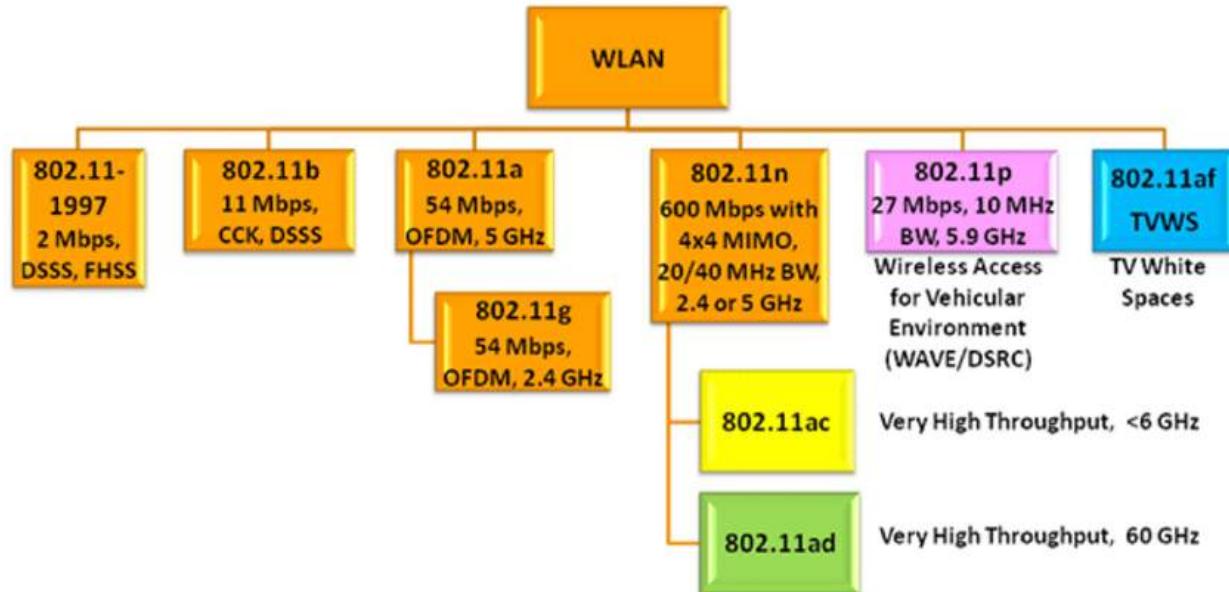
LAN/WLAN World

- **LANs provide connectivity for interconnecting computing resources at local levels of an organization**
- **Wired LANs**
 - Limitations because of physical, hard-wired infrastructure
- **Wireless LANs**
 - Flexibility
 - Portability
 - Mobility
 - Ease of Installation

IEEE 802.11 WLAN Standard

not examinable

- In response to lacking standards, IEEE developed the first internationally recognized wireless LAN standard – IEEE 802.11
- IEEE published 802.11 in 1997, after seven years of work
- Most prominent specification for WLANs
- Scope of IEEE 802.11 is limited to Physical and Data Link Layers

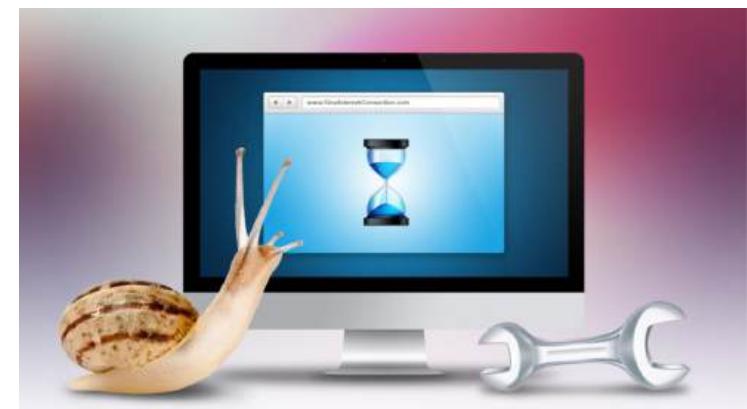
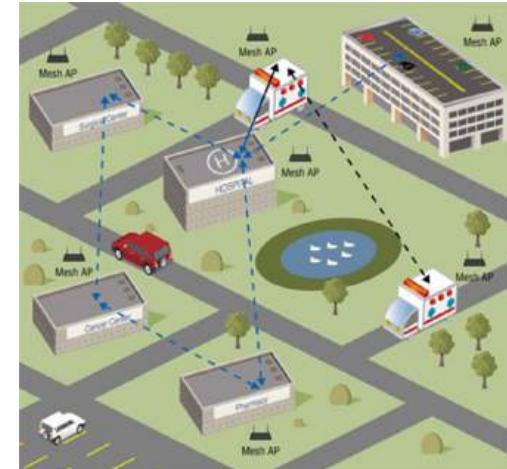


DSRC = Dedicated Short-Range Communications

Wireless LANs: Characteristics

not examinable

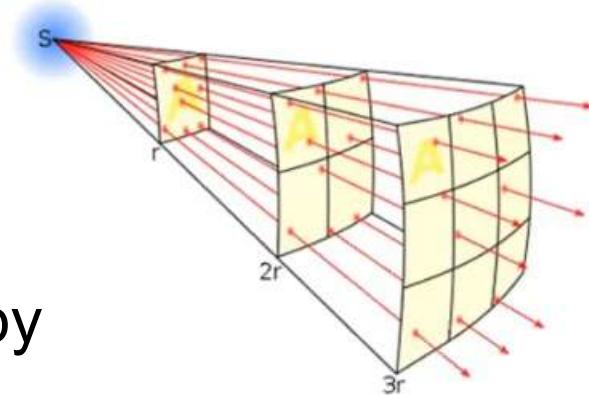
- **Advantages**
 - Flexible deployment
 - Minimal wiring difficulties
 - More robust against disasters (earthquake, etc)
 - Historic buildings, conferences, trade shows,...
- **Disadvantages**
 - Low bandwidth (1-10 Mbit/s) compared to wired networks
 - Proprietary solutions
 - Need to follow wireless spectrum regulations



Wireless Link Characteristics

not examinable

- Different from wired link ...
 - Decreased signal strength: radio signal attenuates as it propagates through matter (path loss)
 - Interference from other sources: standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., phone); devices interfere as well
 - Multipath propagation: radio signal reflects off objects ground, arriving at destination at slightly different times
- ... make communication over wireless link much more “difficult”



WLAN Architecture

not examinable

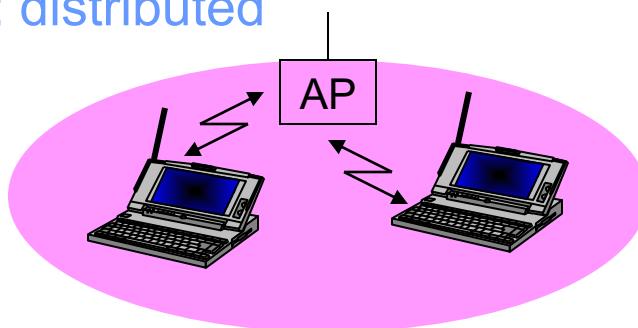
- **Building Modules**

- Station (STA)
 - Mobile node
 - Smartphone, pad, laptop
- Access Point (AP)
 - Stations are connected to access points.

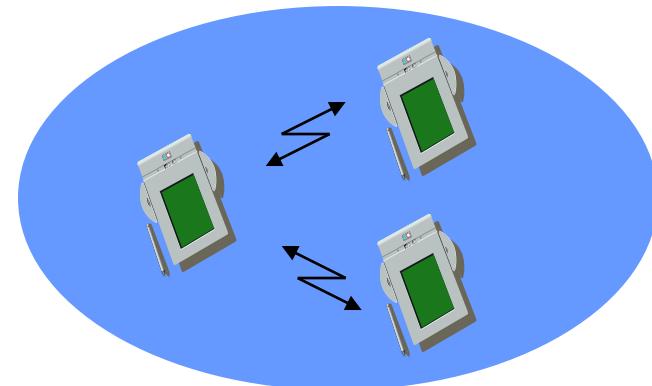


- **Two Architectural Modes**

- Infrastructure: centralized
- Ad Hoc: distributed



Infrastructure



Ad Hoc

(Extended) Service Set

not examinable

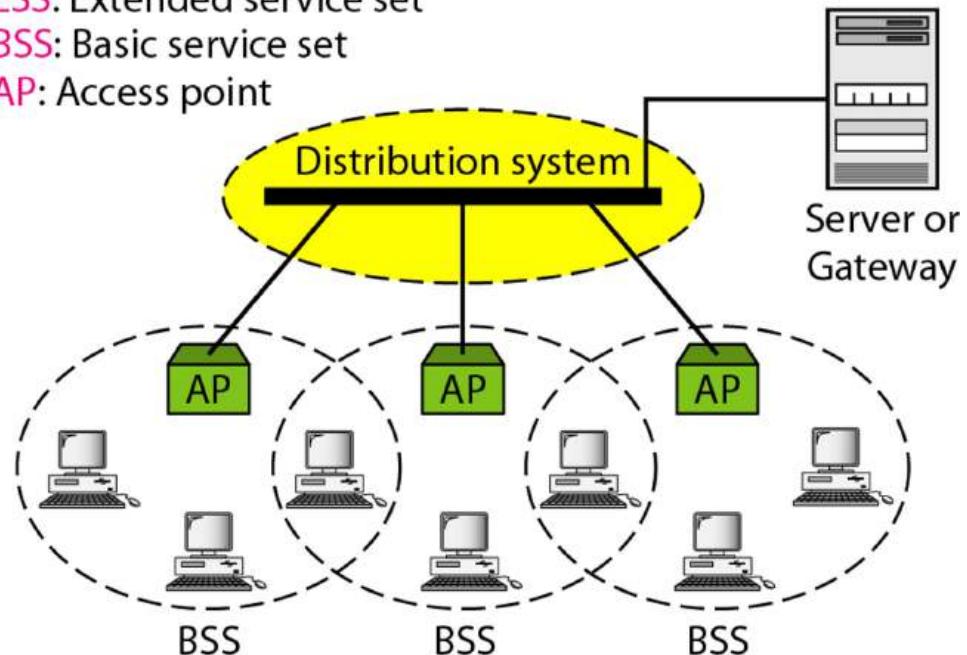
- **Basic Service Set (BSS)**
 - Stations and the AP within the same radio coverage form a BSS.
- **Extended Service Set (ESS)**
 - Several BSSs connected through APs form an ESS.



ESS: Extended service set

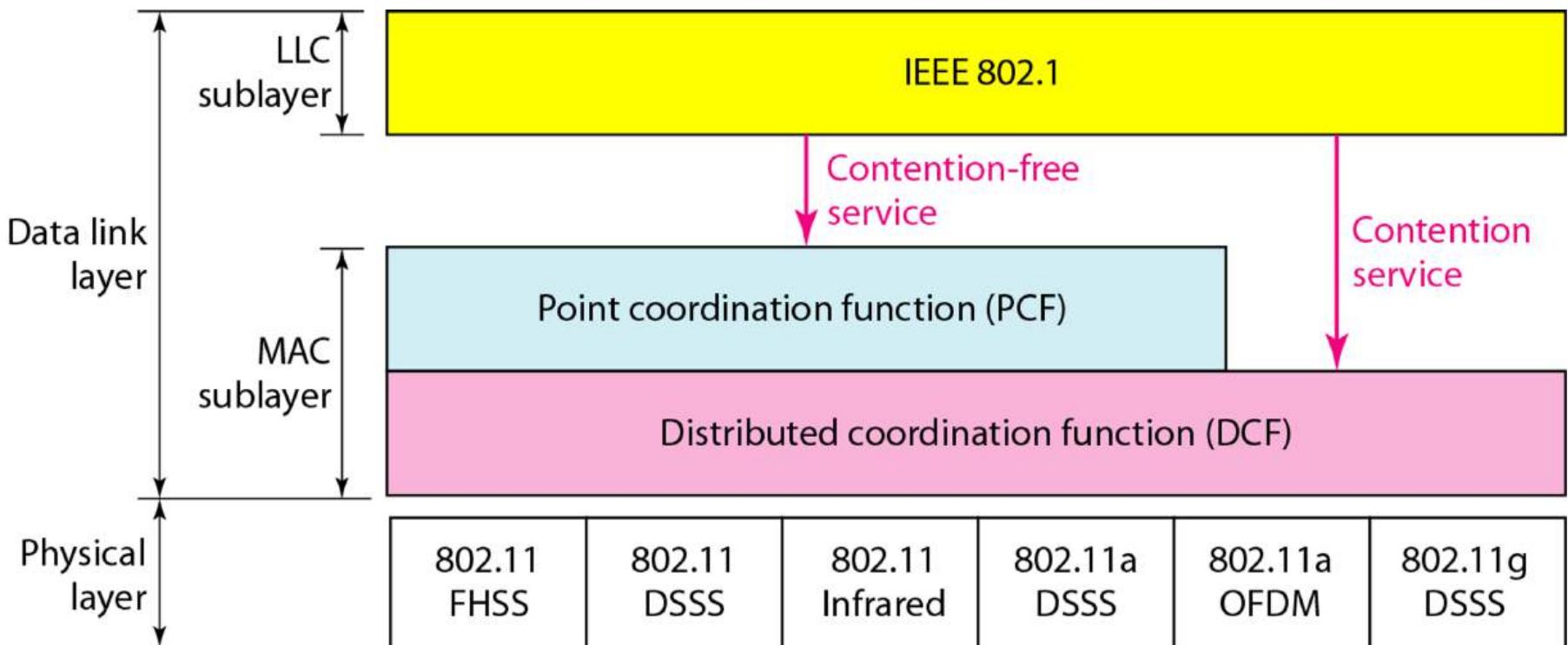
BSS: Basic service set

AP: Access point



802.11 Protocol Stack

not examinable



not examinable

Wireless Physical Layer

Radio Spectrum

not examinable

- **Radio Frequency bands are allocated to different applications**
 - The use of most frequency bands needs licenses
 - IEEE 802.11 uses industrial, scientific and medical (ISM) bands that don't require licenses if the radio transmissions follow the national/global regulations

Sub-THz Radio Spectrum

not examinable

**UNITED
STATES
FREQUENCY
ALLOCATIONS**

THE RADIO SPECTRUM



IEEE 802.11 Physical Layer

not examinable

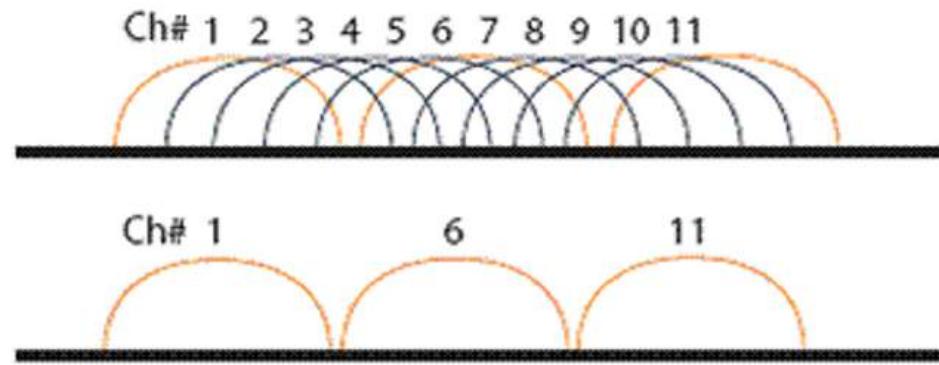
	802.11b	802.11g	802.11a	802.11n	
Frequency Band	2.4GHz	5GHz	2.4GHz	2.4	5
Non-overlapping Channels	3	3	12	3	12
Baseline BW Per Channel	11Mbps	54Mbps	54Mbps	65	65
Max BW Per Channel	11Mbps	54Mbps	54Mbps	130	270
MIMO	1	1	1	4	4
Modulation	DSSS	DSSS/OFDM	OFDM	OFDM	

IEEE 802.11 Channels, Association

not examinable

- **802.11b:** 2.4GHz-2.485GHz spectrum divided into **11 channels** at different frequencies
 - AP admin chooses frequency for AP
 - Interference possible: channel can be same as that chosen by neighboring AP!

802.11b/g Operating Channels



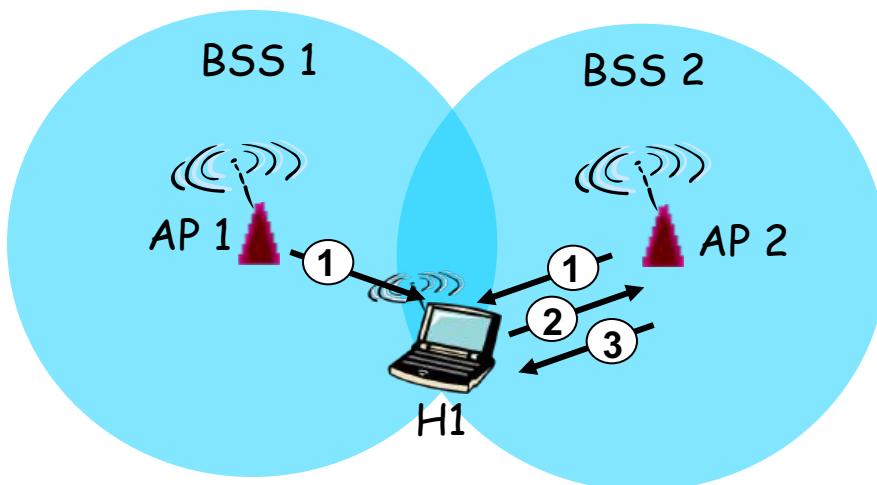
IEEE 802.11 Channels, Association

not examinable

- **802.11b: 2.4GHz-2.485GHz spectrum divided into 11 channels at different frequencies**
 - AP admin chooses frequency for AP
 - Interference possible: channel can be same as that chosen by neighboring AP!
- **Host: must associate with an AP**
 - Scans channels, listening for beacon frames containing AP's name (SSID) and MAC address
 - Selects AP to associate with
 - May perform authentication (security purpose)
 - Will run DHCP to get IP address in AP's subnet

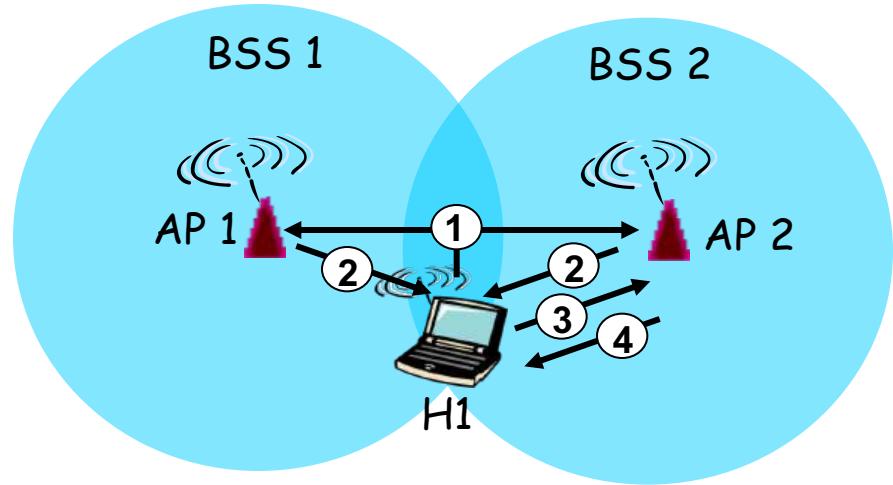
802.11 Passive/Active Scanning

not examinable



Passive Scanning:

- (1) beacon frames sent from APs
- (2) association Request frame sent:
H1 to selected AP
- (3) association Response frame sent:
Selected AP to H1



Active Scanning:

- (1) Probe Request frame broadcast
from H1
- (2) Probes response frame sent from
APs
- (3) Association Request frame sent:
H1 to selected AP
- (4) Association Response frame
sent: selected AP to H1

not examinable

802.11 MAC

802.11 MAC Sublayer

not examinable

- **New challenges caused by the nature of wireless communications**
 - Broadcast
 - Signal attenuation
 - Pervasive electromagnetic noise
- **Three functional areas**
 - Access control (random access vs controlled access)
 - Reliable data delivery (against noises and collisions)
 - Security (authentication, packet injection, ...)
- **Two additional problems:**
 - Hidden Terminal Problem
 - Exposed Terminal Problem

Access Control

not examinable

- **Distributed Coordination Function (DCF)**
 - Distributed access protocol
 - Contention-based
 - Makes use of CSMA/CA
 - Suited for ad-hoc network and asynchronous traffic
- **Point Coordination Function (PCF)**
 - Alternative access method on top of DCF
 - Centralized access protocol
 - Contention-free, and works like polling
 - Suited for time-bound services like voice and multimedia

Reliable Data Delivery

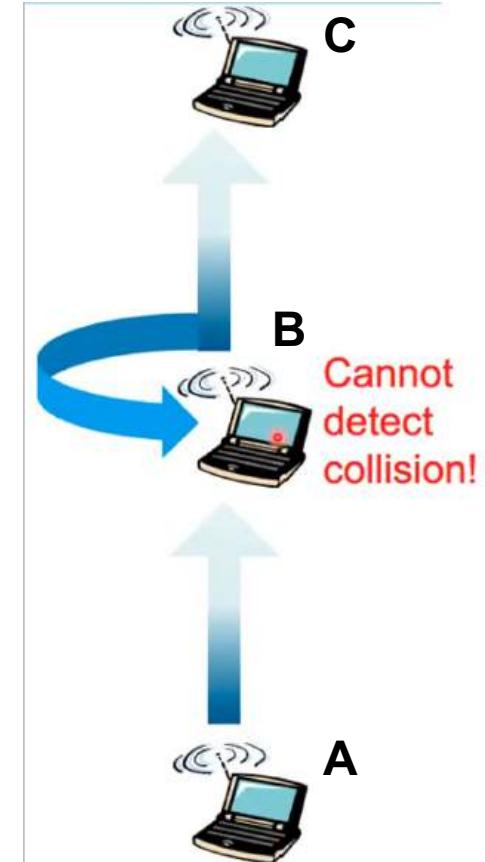
not examinable

- **Loss of frames due to noise, interference and propagation effects**
- **Frame exchange protocol**
 - Sender broadcasts data
 - Receiver responds with acknowledgement (ACK)
 - If sender does not receive ACK, it retransmits frame
- **Four frame exchange for enhanced reliability**
 - Sender issues request-to-send (RTS)
 - Receiver responds with clear-to-send (CTS)
 - Sender transmits data
 - Receiver responds with ACK

802.11 Multi-Access

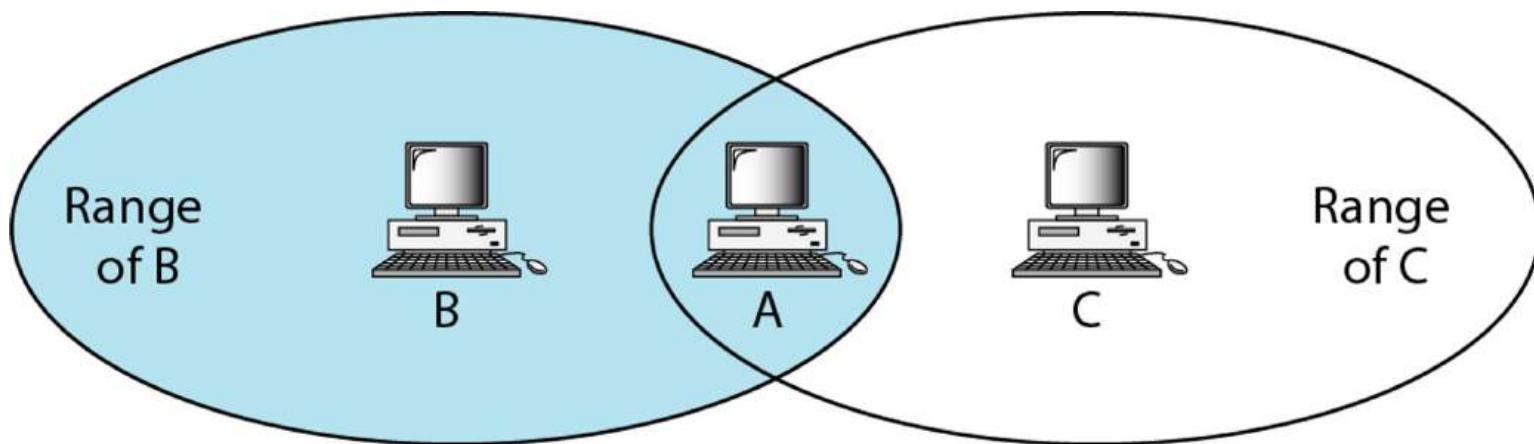
not examinable

- **Collision**
 - A receiver hears transmissions from 2⁺ nodes at the same time
- **802.11: CSMA - sense before transmitting**
 - Don't collide with ongoing transmission by other node
- **802.11: no collision detection!**
 - Difficult to receive (sense collisions) when transmitting due to weak received signals (fading)
 - Can't sense all carriers & collisions in any case: hidden terminal problem
- **802.11: avoid collisions**
 - CSMA/C(ollision)A(voidance)



Hidden Terminal Problem

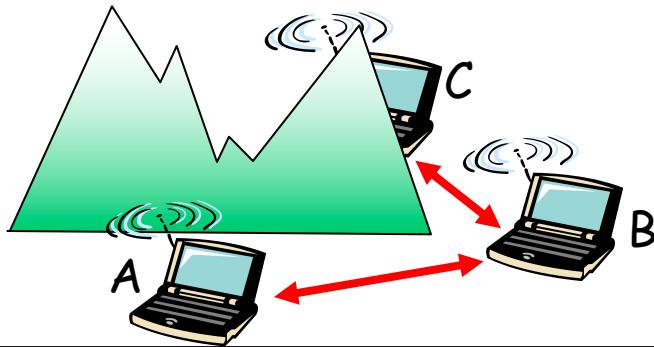
not examinable



B and C are hidden from each other with respect to A.

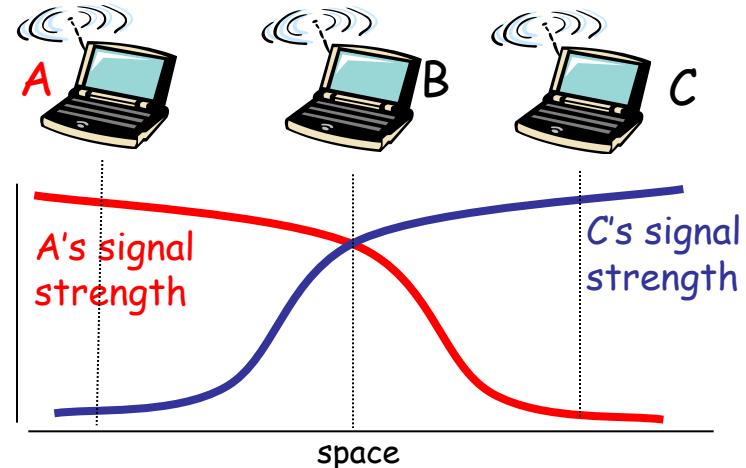
Hidden Terminals in Wireless LAN

not examinable



Case 1: Caused by barrier

- B, A hear each other
- B, C hear each other
- A, C can not hear each other
 - A, C unaware of their interference at B
 - A is a hidden terminal to C, vice versa



Case 2: Caused by signal attenuation

- B, A hear each other
- B, C hear each other
- A, C can not hear each other
 - A, C unaware of their interference at B

Question: Does Ethernet have hidden terminal problem?

Answer: No. Ethernet does not have Case 1 and Case 2 above.



Breakout session to discuss

Collision Avoidance

not examinable

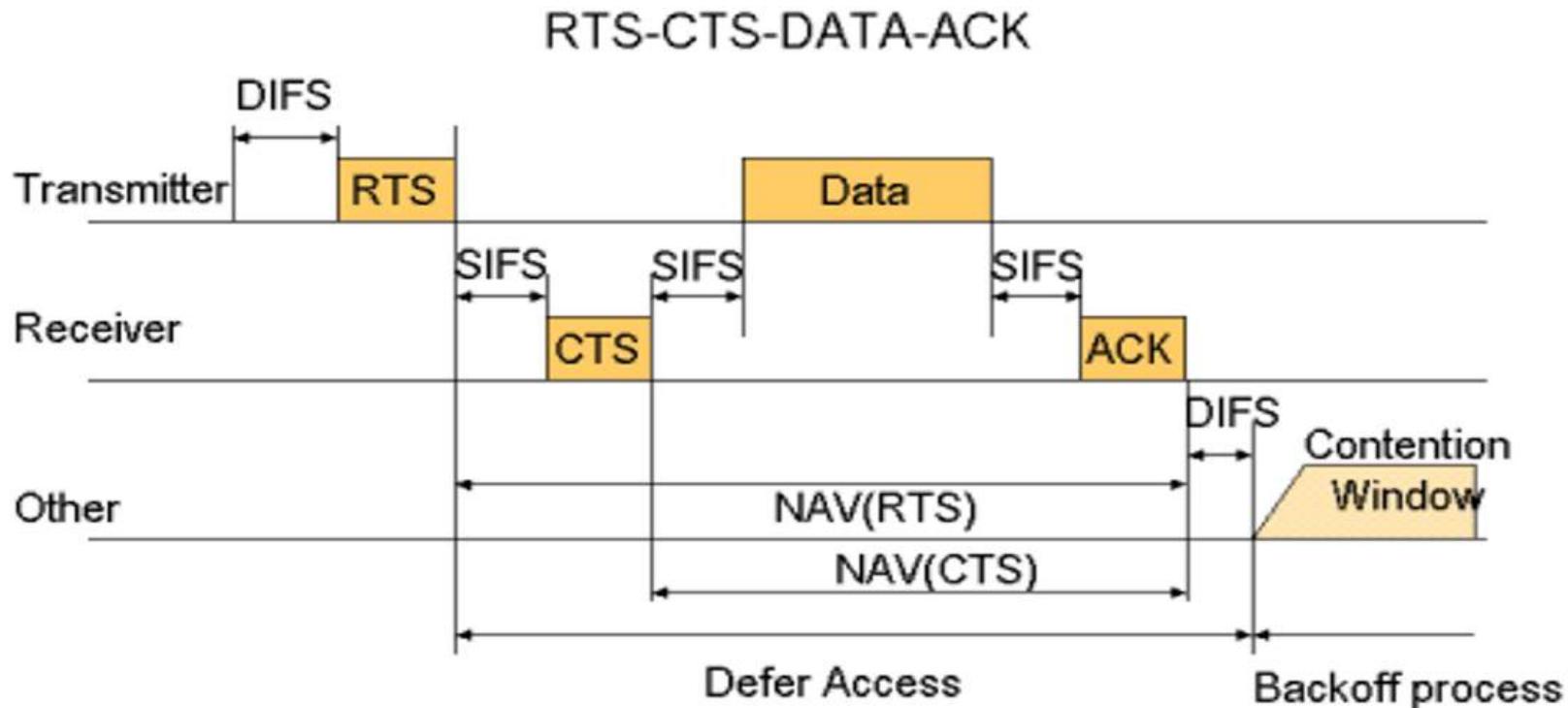
idea: Sender to “reserve” channel for a long data frame

- Sender first transmits a *small* **request-to-send (RTS)** packet to receiver using CSMA
 - RTSs may still collide with each other, or an RTS may collide with an ongoing data frame
 - but they’re short
- Receiver broadcasts **clear-to-send (CTS)** in response to RTS
- CTS heard by all nodes
 - Sender transmits data frame
 - Other stations defer transmissions

Avoid data frame collisions completely
using small reservation packets!

RTS-CTS-DATA-ACK

not examinable



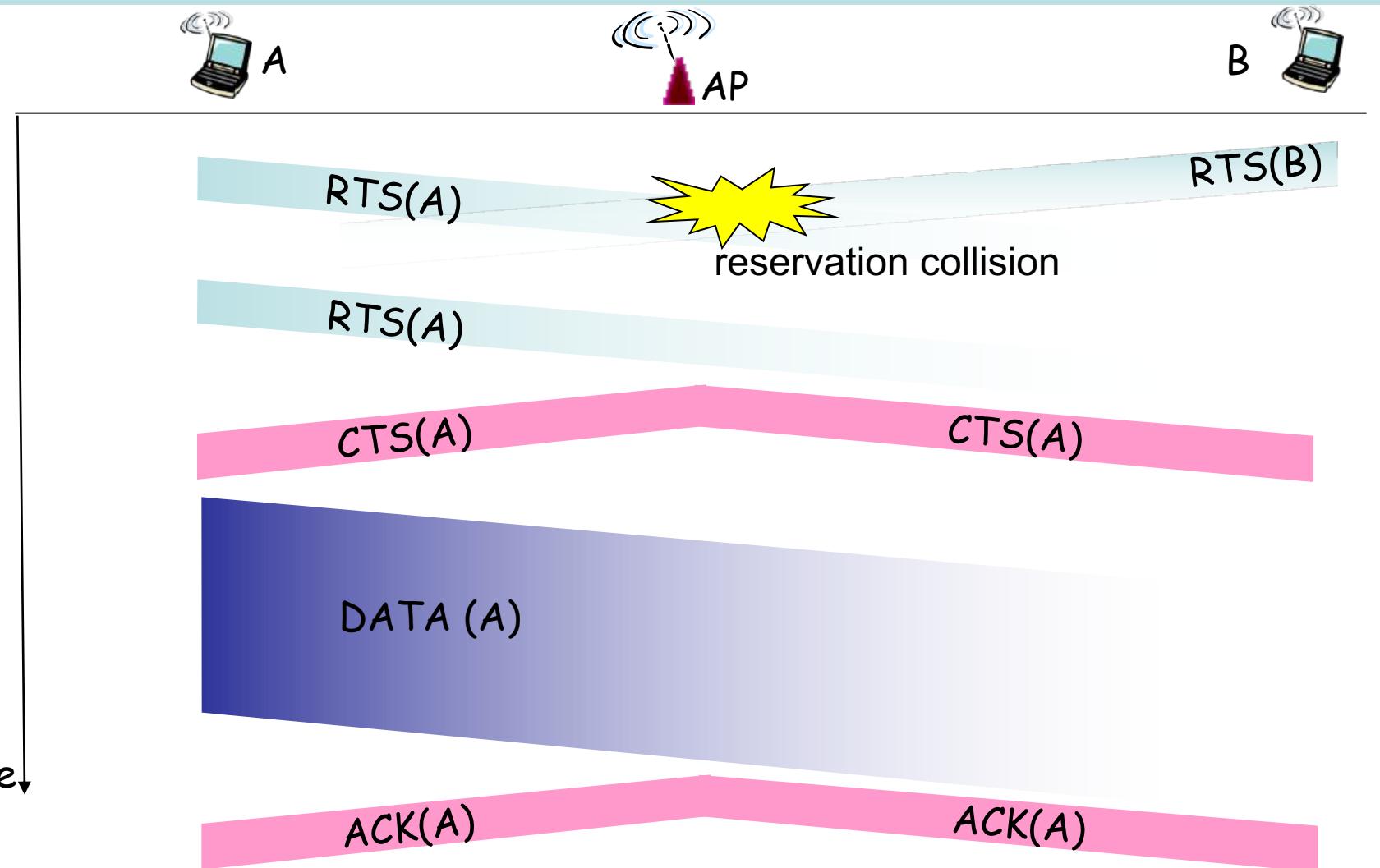
DIFS: Distributed IFS (Inter-frame Space)
for carrier sense

RTS: Request-To-Send
SIFS: Short IFS

CTS: Clear-To-Send
ACK: Acknowledgement
NAV: Network Allocation Vector

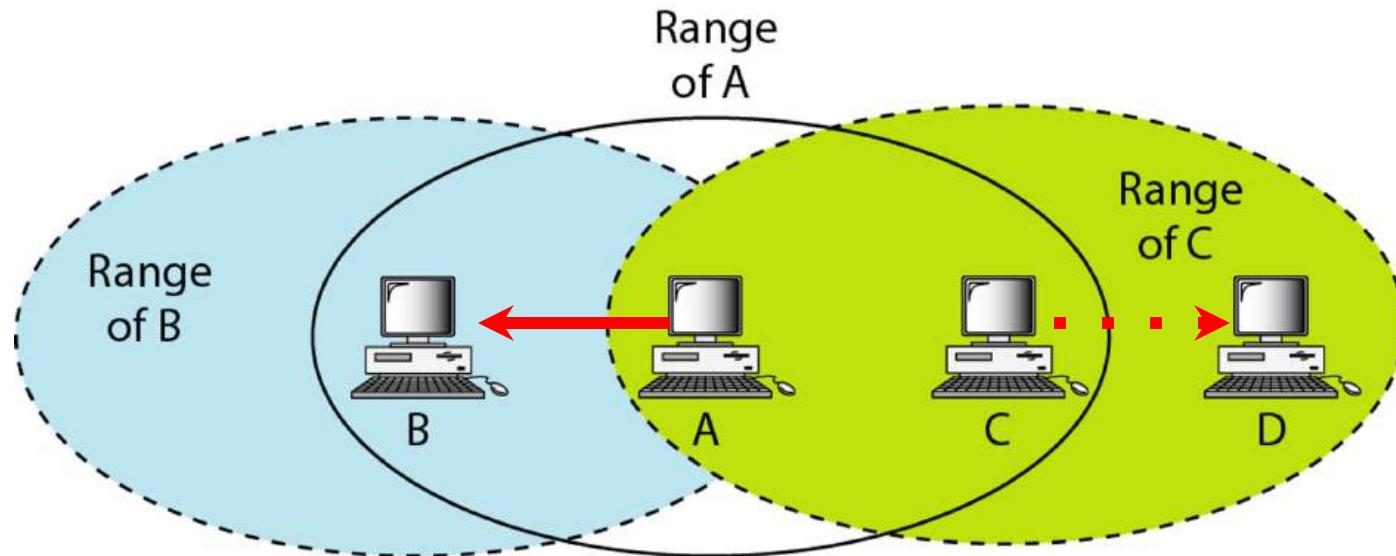
Handshaking in Hidden Terminal Problem

not examinable



Exposed Terminal Problem

not examinable

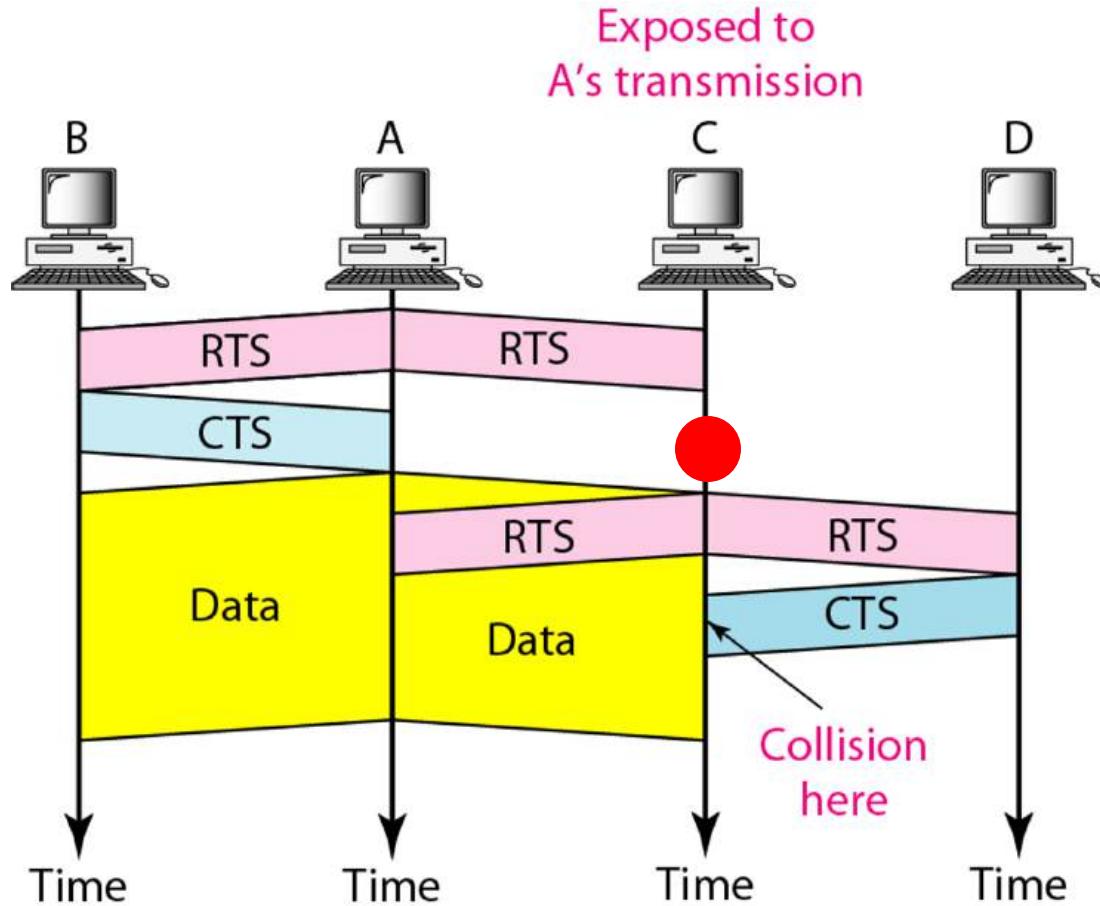


C is exposed to transmission from A to B.

Handshaking in Exposed Terminal Problem

not examinable

- RTS-CTS ensures no collision
- but doesn't solve the opportunity wasting problem



802.11 Frame

not examinable

2	2	6	6	6	2	6	0 - 2312	4
frame control	duration	address 1	address 2	address 3	seq control	address 4	payload	CRC

Address 1: MAC address of wireless host or AP to receive this frame

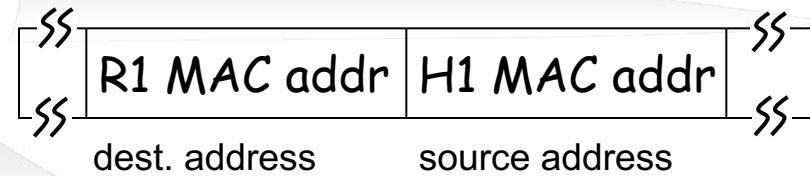
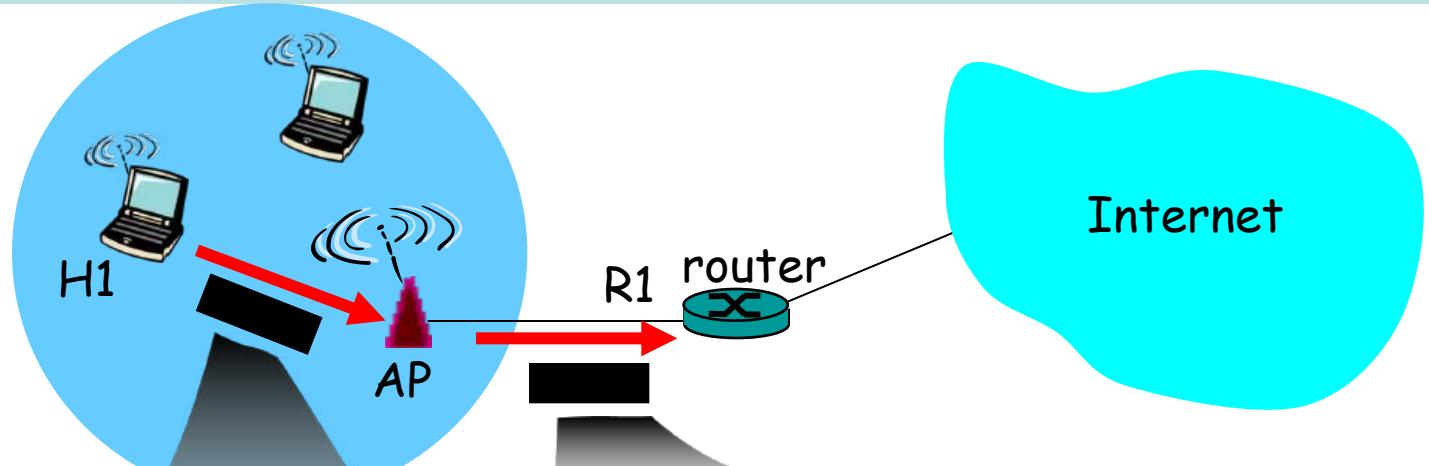
Address 2: MAC address of wireless host or AP transmitting this frame

Address 3: MAC address of router interface to which AP is attached

Address 4: used only in ad hoc mode

802.11 Addressing

not examinable



Ethernet frame



802.11 frame

802.11 Advanced Capabilities

not examinable

- **Synchronization**
 - finding and staying with a WLAN
 - synchronization functions
- **Power Management**
 - sleeping without missing any messages
 - power management functions
- **Roaming**
 - functions for joining a network
 - changing access points
 - scanning for access points
- **Management information base**

Examinable

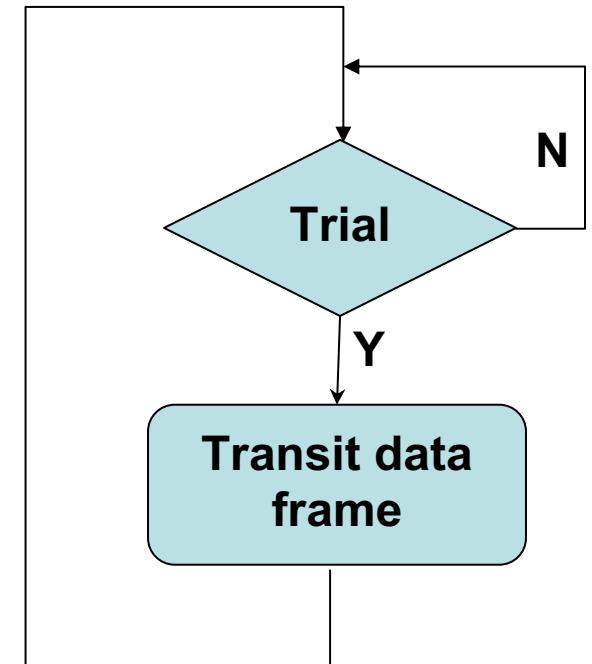
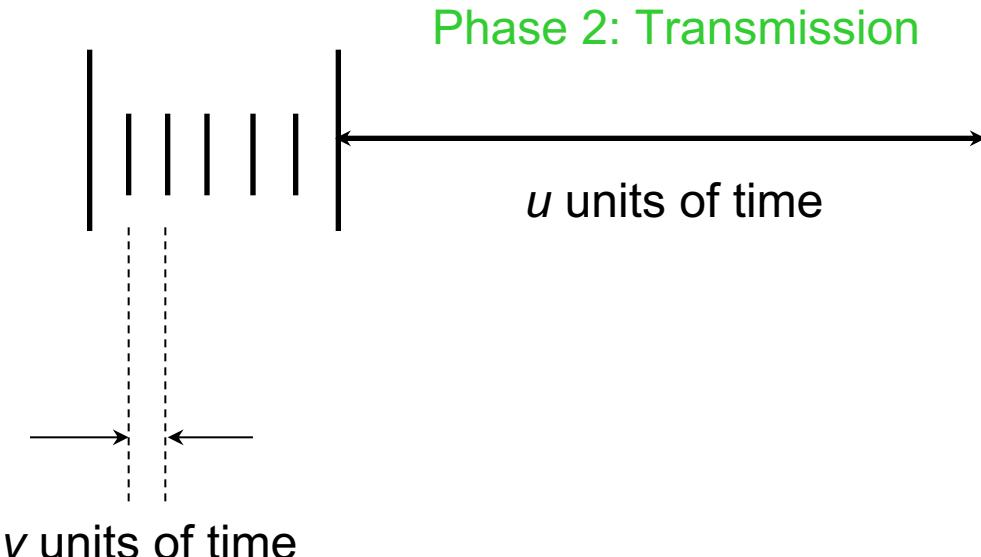
Multi-Access Reservation Protocol (MARP)

Multi-Access Reservation Protocol

- **Two-Phase Protocol**
 - Phase 1: Channel Reservation
 - Phase 2: Data Transmission

Examinable

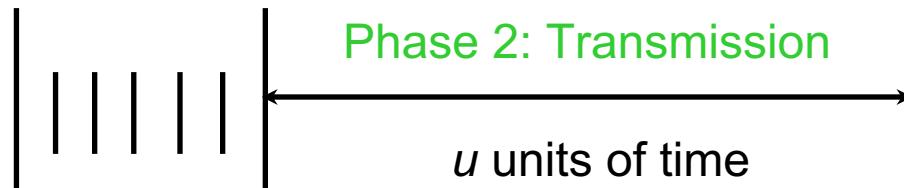
Phase 1: Reservation



MARP Transmission Window

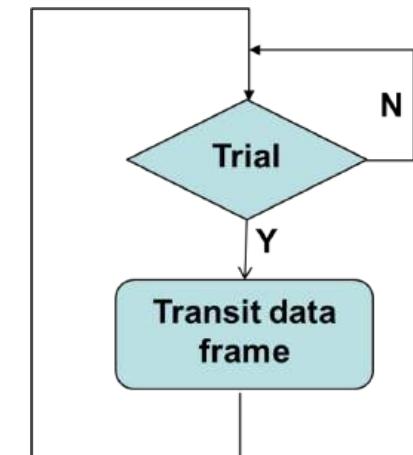
Phase 1: Reservation

Examinable



How many reservation trial frames?

- Assume that the channel utilization in reservation phase: S_r
- Number of reservation trial frames to reserve the channel: X
 - $X = 1$ (the first trial succeeds) with probability of S_r
 - $X = k$ (the first $k-1$ trials fail, the k^{th} trial succeeds) with probability of $S_r(1-S_r)^{k-1}$
 - This is a geometric distribution, so $E[X] = 1/S_r$
- The average transmission window is $u + v/S_r$ units of time



MARP Throughput

Examinable

$$\text{Throughput } S = \frac{\text{Time for message transmission}}{\text{Transmission window}}$$

Case	Message Length	Reservation Phase Length	Throughput
Reservation frame not used for message data bits	u	v/S_r	$S = \frac{u}{u + v/S_r}$
Reservation frame used for message data bits	u	v/S_r	$S = \frac{u}{(u - v) + v/S_r}$

MARP Example

Examinable

Consider an experimental LAN using an MARP for data transmission. The protocol consists of two phases. In phase 1, it adopts some MAC protocol for transmission stations to reserve the channel. In phase 2, when one station reserves the channel, it transmits one frame. The length of reservation frame is 5ms, and the length of the data frame is 1s. No information bit is carried in the reservation frame. If the MAC protocol used in phase 1 has a utilization of 0.5, what is the throughput of the multi-access reservation protocol?

CRACK Framework:

Context: MARP with no data bits in reservation

fRamwork: the throughput of MARP is $S=1/(1+v/S_r)$

Apply: $v=5\text{ms}$, $S_r=0.5$

Calculation: $S = 1/(1+0.005/0.5)=1/1.01=0.99$

checK: $S <= 1$

Local Area Network Summary

MAC Protocols		Transmission Protocol			Throughput/ Utilization	Note									
		Carrier Sensing	Frame Transmission	Collision Detection											
Aloha	Slotted	<ul style="list-style-type: none"> None 	<ul style="list-style-type: none"> Each transmits in a slot immediately with probability p 	<ul style="list-style-type: none"> When a collision is detected, the colliding frames are transmitted up to their last bits. 	$S = Np(1 - p)^{(N-1)} = Ge^{-G}$	Number of Stations: N Probability of Attempt: p Attempt Rate: $G = Np$									
	Pure		<ul style="list-style-type: none"> Each transmits immediately with probability p 		$S = Np(1 - p)^{2(N-1)} = Ge^{-2G}$										
CSMA	Non-Persistent	<ul style="list-style-type: none"> Must sense channel before transmission 	<ul style="list-style-type: none"> When a busy channel is sensed, a station defers for a random period of time before next sense 	<ul style="list-style-type: none"> When a busy channel is sensed, a station continues to sense until the channel turns idle. Then, with probability p, it transmits, and with probability $1 - p$, it defers to next time slot. 											
	P-Persistent														
	1-Persistent		<ul style="list-style-type: none"> A special case of P-Persistent where $p = 1$ 												
CSMA/CD (Ethernet)		<ul style="list-style-type: none"> Must sense channel before transmission 	<ul style="list-style-type: none"> The same as CSMA 	<ul style="list-style-type: none"> When a collision is detected, transmissions are aborted to reduce the channel wastage. 		Minimum Frame Size <ul style="list-style-type: none"> $T_{frame} \geq 2\tau$ Binary Exponential Backoff In i-th retransmission, the slot is chosen from a uniformly distributed random variable R_i in the range of $[0, 2^K - 1]$, where $K = \min(i, 10)$. 									
CSMA/CA (802.11)		<ul style="list-style-type: none"> Must sense channel before transmission 	<p>Sender:</p> <ul style="list-style-type: none"> If sense channel idle for DIFS, then transmit entire frame (no CD). If sense channel busy, then start random backoff time. Transmits when timer expires. If no ACK, increase random backoff interval <p>Receiver:</p> <ul style="list-style-type: none"> If frame received OK, return ACK after SIFS 	<ul style="list-style-type: none"> No collision detection due to hidden terminal 	Multi-Access Reservation <ul style="list-style-type: none"> Use random-access with mini-frame (v unit of time) to reserve the channel If reservation successful, transmit u unit of data frame 										
					<table border="1"> <tr> <td></td><td>$\frac{u}{u + v/S_r}$</td><td>$\frac{u}{(u - v) + \frac{v}{S_r}}$</td></tr> <tr> <td>Total data length</td><td>u</td><td>u</td></tr> <tr> <td>Data bit in mini-frame</td><td>No</td><td>Yes</td></tr> </table>		$\frac{u}{u + v/S_r}$	$\frac{u}{(u - v) + \frac{v}{S_r}}$	Total data length	u	u	Data bit in mini-frame	No	Yes	
	$\frac{u}{u + v/S_r}$	$\frac{u}{(u - v) + \frac{v}{S_r}}$													
Total data length	u	u													
Data bit in mini-frame	No	Yes													

Learning Objectives

- **WLAN Overview**
 - Understand two alternative WLAN architectures
- **802.11 Physical Layer**
 - Understand different transmission schemes
- **802.11 MAC Layer**
 - Understand hidden and exposed terminal problems
 - Understand CSMA/CA protocol
- **Multi-Access Reservation Protocol (MARP)**
 - Understand the scheme of MARP
 - Calculate and maximize throughput for MARP

not examinable

Reading Material

Wireless Physical Layer (I)

not examinable

- **Physical layer conforms to OSI (five options)**
 - 1997: **802.11** infrared, FHSS, DHSS
 - 1999: **802.11a** OFDM and **802.11b** HR-DSSS
 - 2001: **802.11g** OFDM
- **802.11 Infrared**
 - Two capacities 1 Mbps or 2 Mbps.
 - Range is 10 to 20 meters and cannot penetrate walls.
 - Does not work outdoors.
- **802.11 FHSS (Frequency Hopping Spread Spectrum)**
 - The main issue is multipath fading.
 - 79 non-overlapping channels, each 1 MHz wide at low end of 2.4 GHz ISM band.
 - Same pseudo-random number generator used by all stations.
 - Dwell time: min. time on channel before hopping (400msec).

Wireless Physical Layer (II)

not examinable

- **802.11 DSSS (*Direct Sequence Spread Spectrum*)**
 - Spreads signal over entire spectrum using pseudo-random sequence (similar to CDMA see Tanenbaum sec. 2.6.2).
 - Each bit transmitted using an 11 chips Barker sequence, PSK at 1Mbaud.
 - 1 or 2 Mbps.
- **802.11a OFDM (*Orthogonal Frequency Divisional Multiplexing*)**
 - Compatible with European HiperLan2.
 - 54Mbps in wider 5.5 GHz band □ transmission range is limited.
 - Uses 52 FDM channels (48 for data; 4 for synchronization).
 - Encoding is complex (PSM up to 18 Mbps and QAM above this capacity).
 - E.g., at 54Mbps 216 data bits encoded into 288-bit symbols.
 - More difficulty penetrating walls.

Wireless Physical Layer (III)

not examinable

- **802.11b HR-DSSS (*High Rate Direct Sequence Spread Spectrum*)**
 - 11a and 11b shows a split in the standards committee.
 - 11b approved and hit the market before 11a.
 - Up to 11 Mbps in 2.4 GHz band using 11 million chips/sec.
 - Note in this bandwidth all these protocols have to deal with interference from microwave ovens, cordless phones and garage door openers.
 - Range is 7 times greater than 11a.
 - **11b and 11a are incompatible!!**
- **802.11g OFDM(*Orthogonal Frequency Division Multiplexing*)**
 - An attempt to combine the best of both 802.11a and 802.11b.
 - Supports bandwidths up to 54 Mbps.
 - Uses 2.4 GHz frequency for greater range.
 - Is backward compatible with 802.11b.

Part I Syllabus - Fundamental Underlying Layers

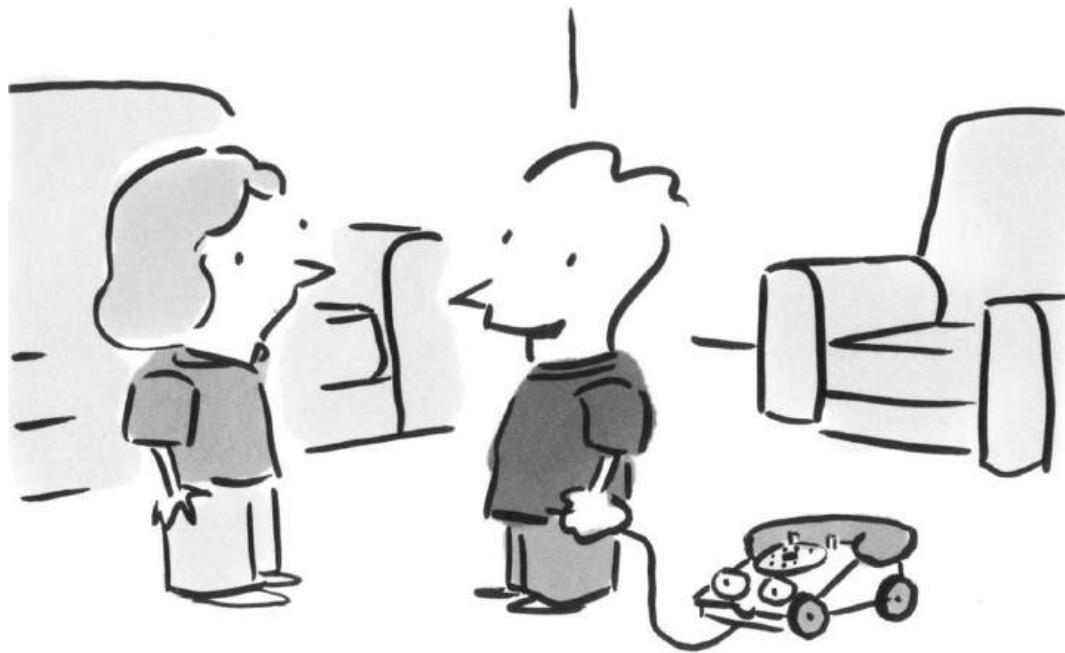
Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Mobile Access Networks	M1-L9-Mobile.pptx
Week 7: 20/Feb/2023 22/Feb/2023	E-learning for Network paradigms	M1-L10-Paradigms.pptx
	Network paradigms	M1-L10-Paradigms.pptx

Additional Materials

- The related content talked today in
[https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclass.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:
 - WiFi: 802.11 Wireless LANs: Page 526 – Page 546
 - Cellular Internet Access: Page 547 – Page 555
 - Mobility Management: Principles: Page 555 – Page 564
- You can also find other video materials about
 - 802.11 WiFi <https://www.youtube.com/watch?v=t3FVP5wuG4g>
 - Wireless Communication
<https://www.youtube.com/watch?v=8T7orRAQgic&list=PLCyR4nKNLRkFTER9ohRBnbRFK0pWe0Qtf&index=1&t=61s>

Mobile Access

© MARK ANDERSON, WWW.ANDERZTOONS.COM



ANDERSON

"The best part is there's no
roaming charges!"

SC2008/CZ3006/CE3005

Computer Network

Lecture 9

Mobile Access Networks:

From 1G to 5G

(Not Examinable)



Contents

- **Wireless Link Standards**

- 802.11 (Wi-Fi)
- Cellular (1G to 5G)
- 802.15 (e.g., ZigBee)
- 802.16 (WiMax)

- **Cellular Networks**

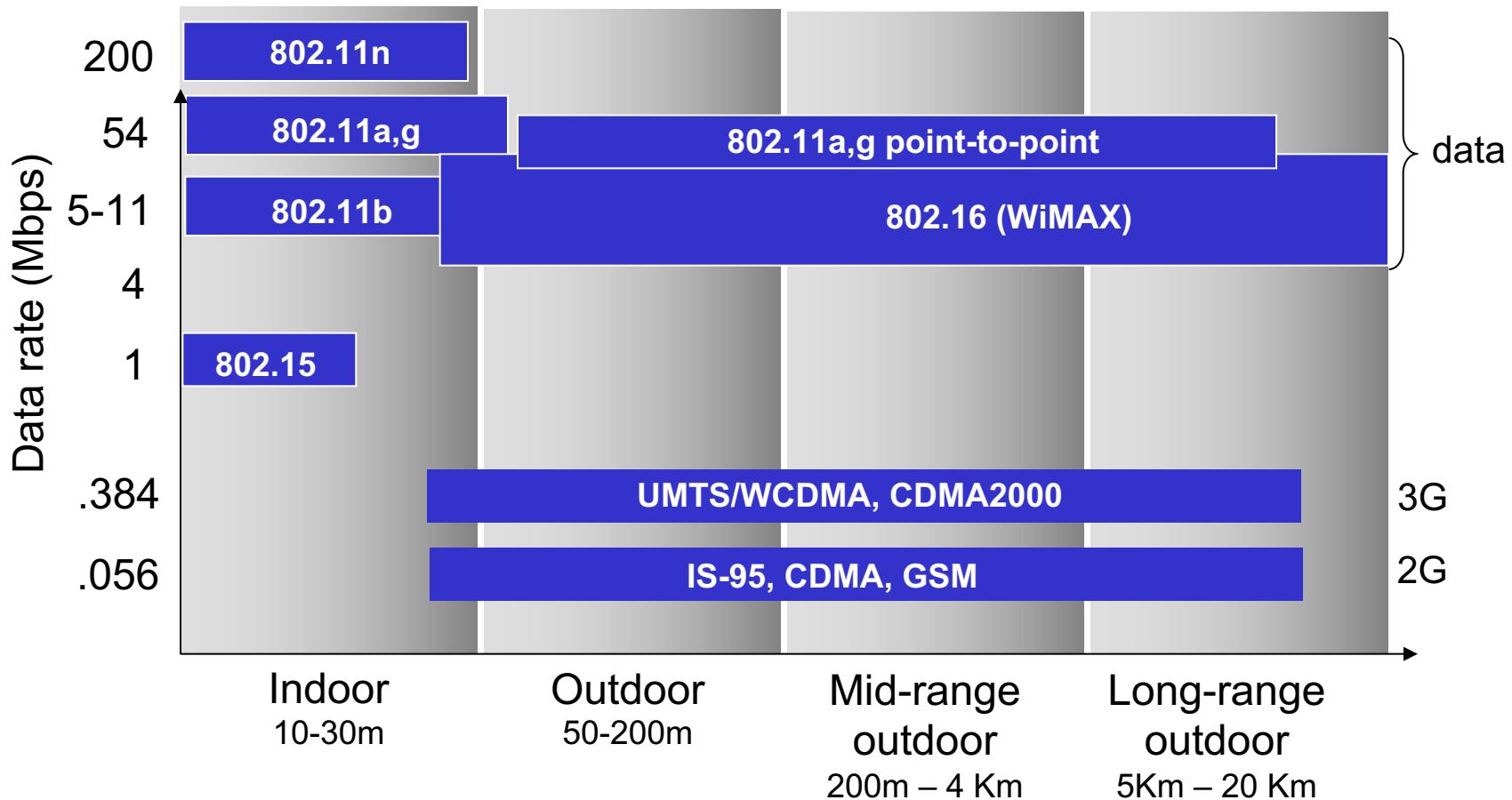
- Network architecture
- MAC protocols
- 1G to 5G

- **Mobility handling**

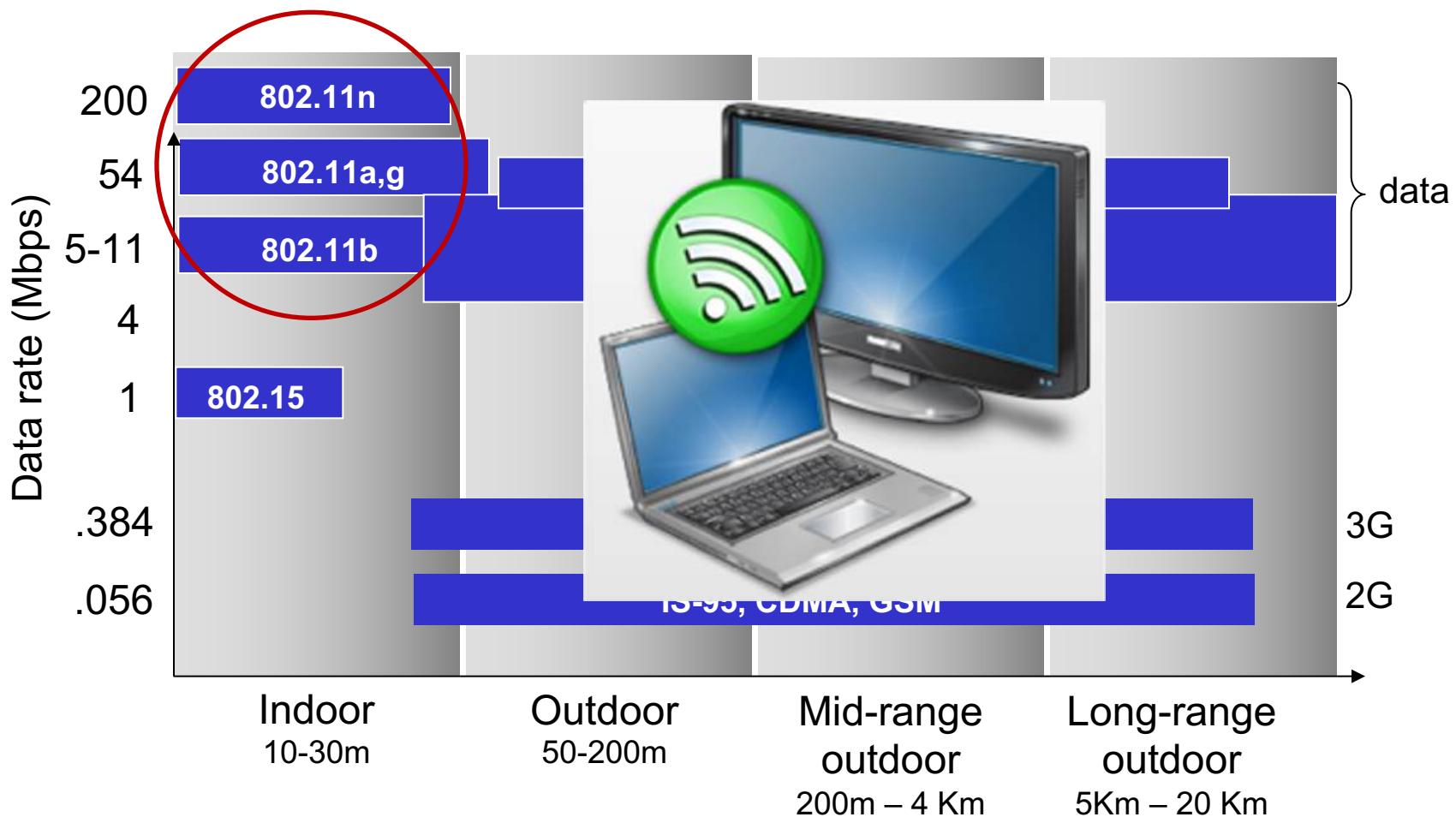
- Mobility vocabulary
- Indirect routing
- Direct routing

Wireless Link Standards

Characteristics of Selected Wireless Link Standards



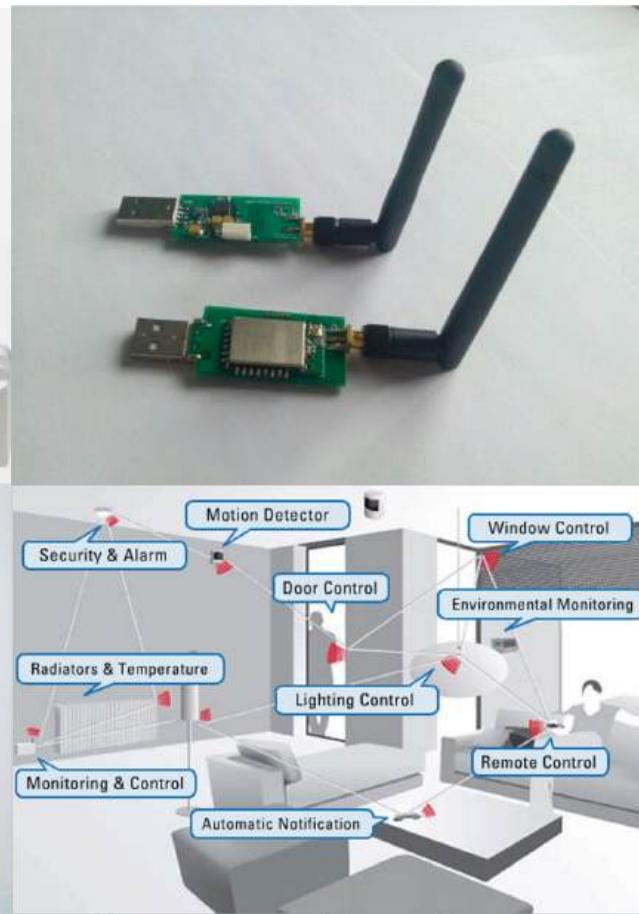
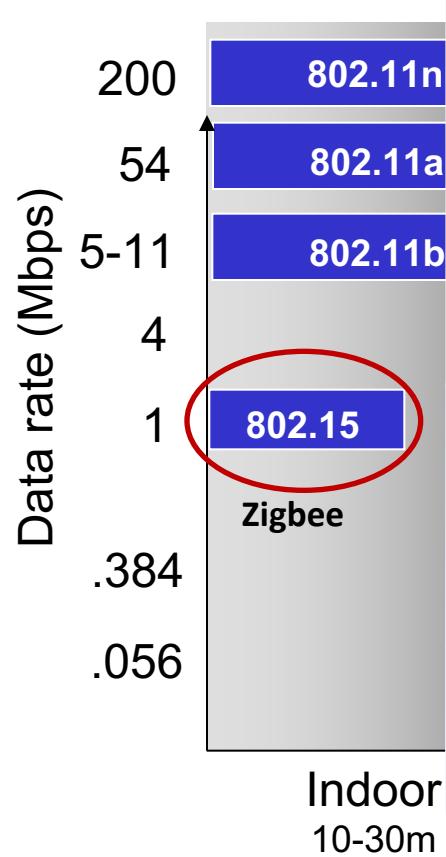
Characteristics of Selected Wireless Link Standards



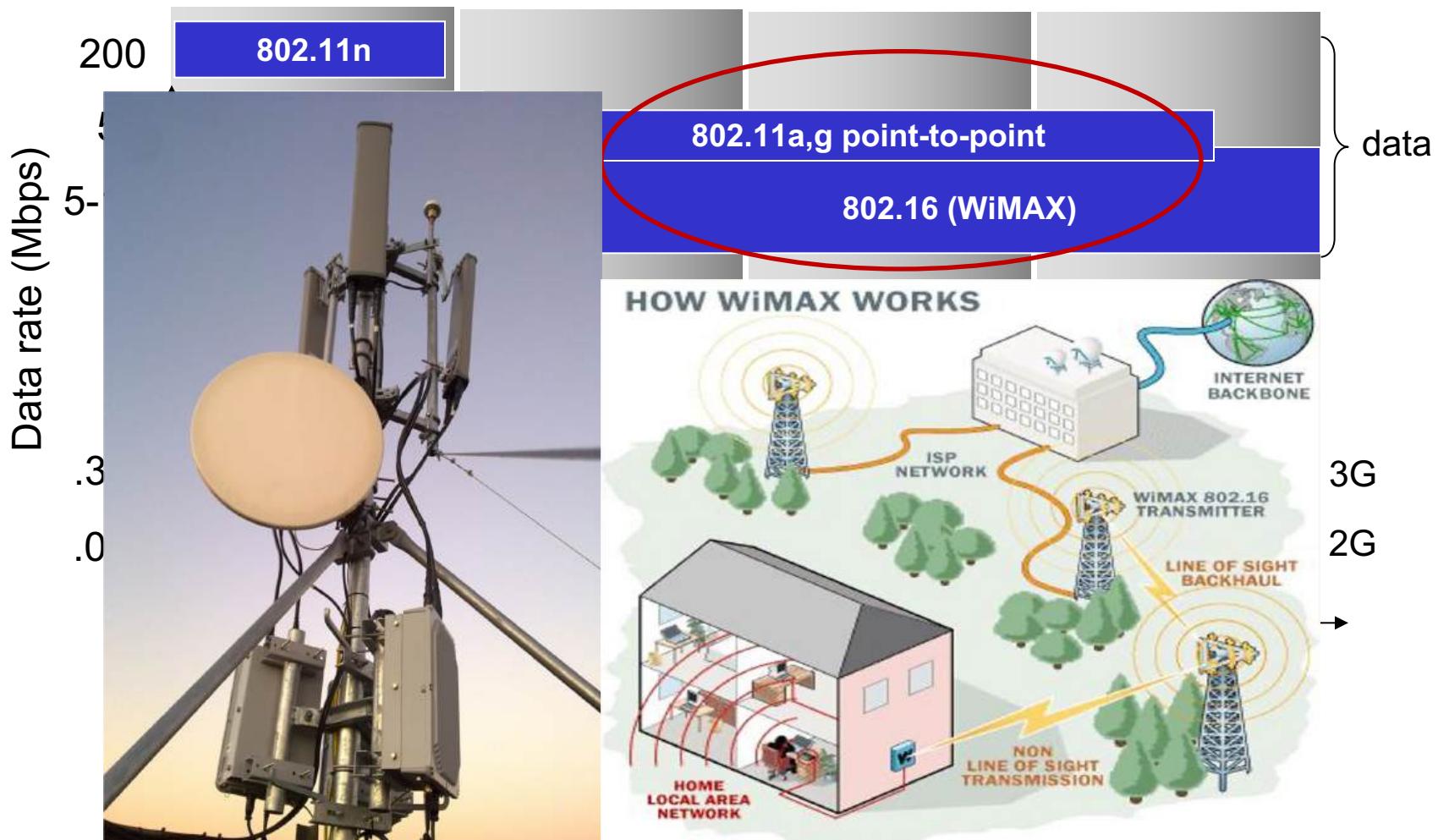
Characteristics of Selected Wireless Link Standards



Characteristics of Selected Wireless Link Standards



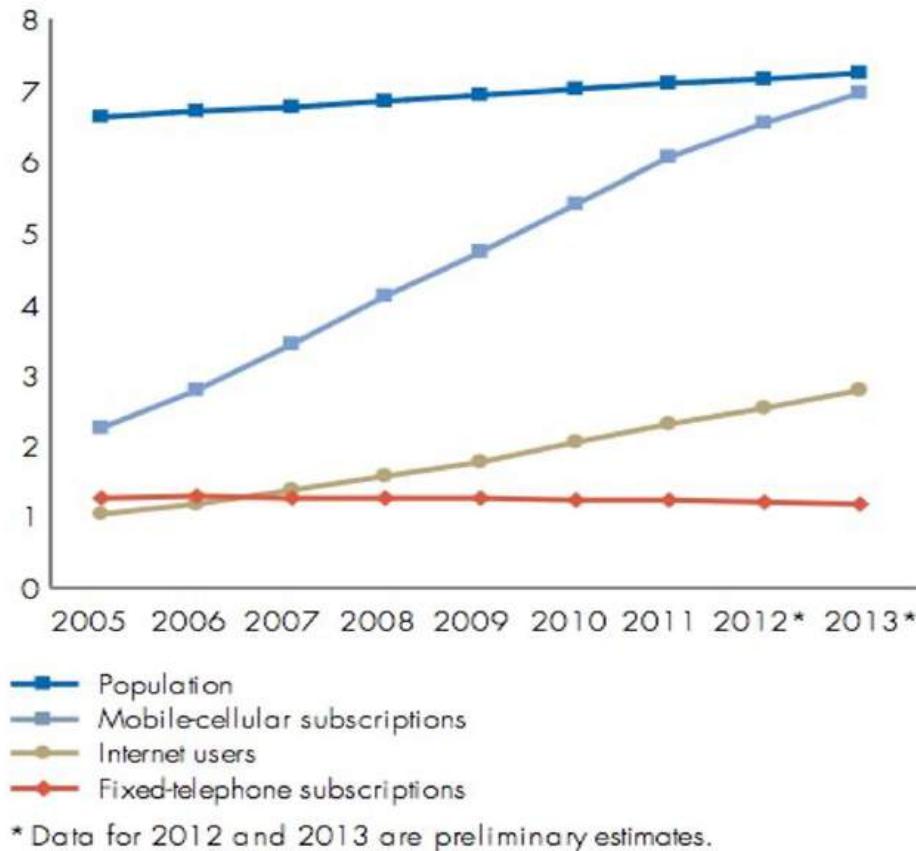
Characteristics of Selected Wireless Link Standards



Cellular Networks: 1G to 5G

Boost of Mobiles

**Estimated number of mobile-cellular subscriptions,
Internet users and fixed-telephone subscriptions,
2005-2013 (Billions)**



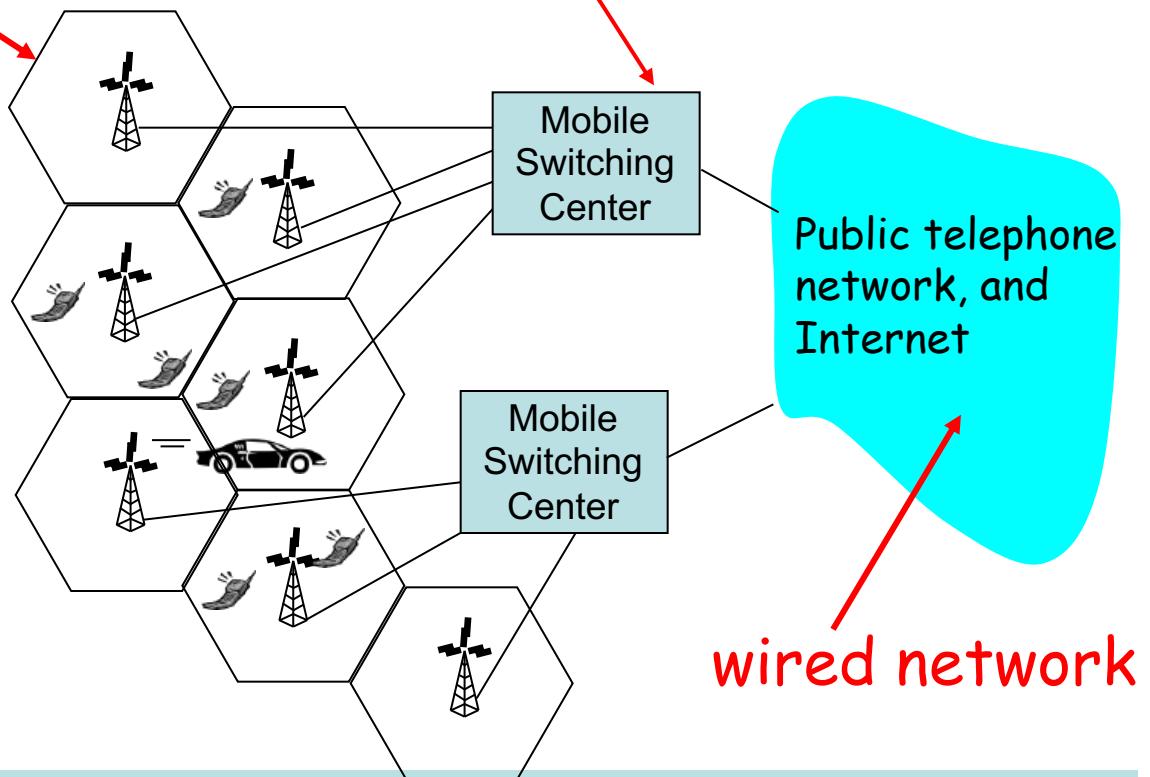
Cellular Network Architecture

cell

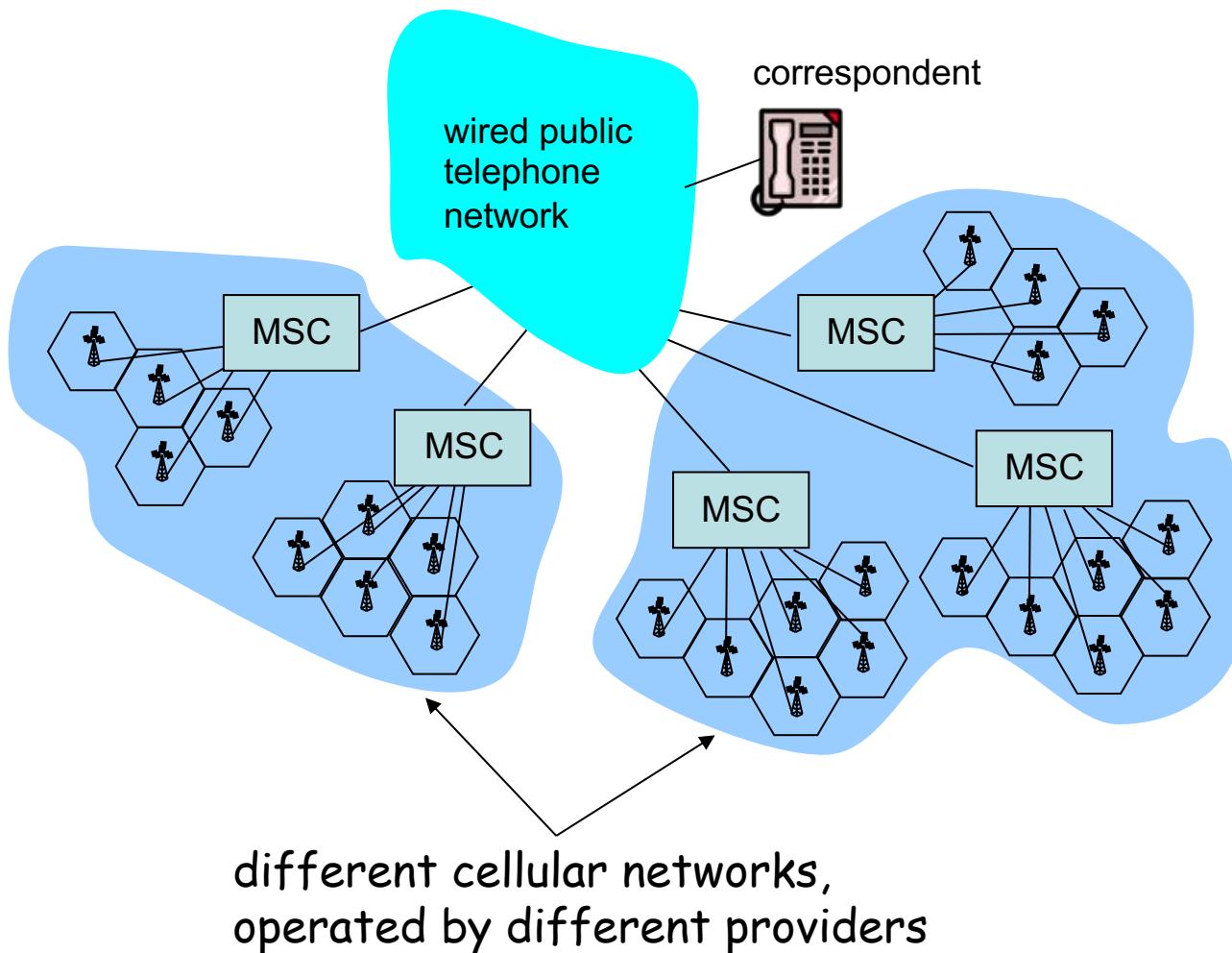
- ❑ covers geographical region
- ❑ **base station (BS)** similar to 802.11 AP
- ❑ **mobile users** attach to network through BS
- ❑ **air-interface:** physical and link layer protocol between mobile and BS

MSC

- ❑ connects cells to wide area net
- ❑ manages call setup
- ❑ handles mobility



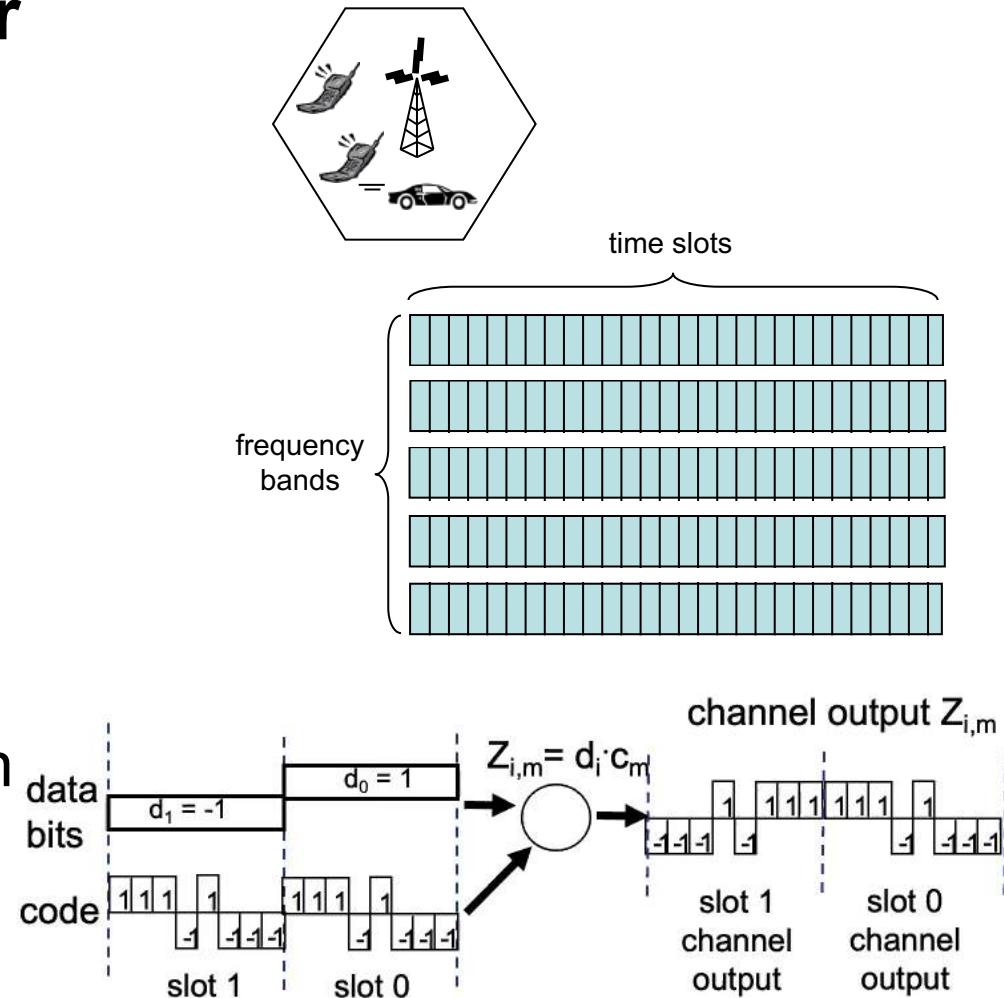
Interconnection of Cellular Networks



Cellular Network: The First Hop

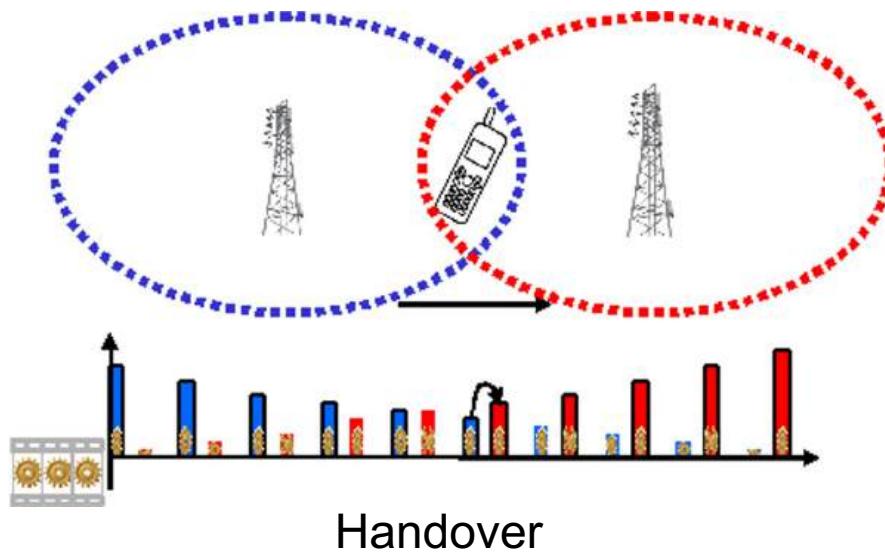
- Two techniques for sharing mobile-to-BS radio spectrum

- Combined **FDMA/TDMA**: divide spectrum in frequency channels, divide each channel into time slots
- **CDMA**: code division multiple access



Handover and Roaming

- **Handover:** transfer a call or data session from one cell to another within the same cellular network
 - Handled by Mobile Switching Center (MSC), no charge
- **Roaming:** the mobile moves from its home cellular network to a foreign network



Cellular Standards: Brief Survey



1G: analog, 3 decades ago



2G/2.5G: 50-384kbps, 2 decades ago



3G: up to 14Mbps

4G: up to 300Mbps

Last decade



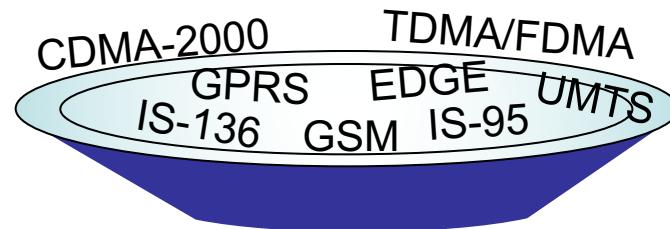
5G: up to 430Mbps using sub-6GHz

up to 10Gbps using >24GHz

From 2019

Cellular Standards: Brief Survey

- **2G systems: voice channels**
 - IS-136 TMDA: combined FDMA/TDMA (North America)
 - GSM (global system for mobile communications): combined FDMA/TDMA
 - Most widely deployed
 - IS-95 CDMA: code division multiple access

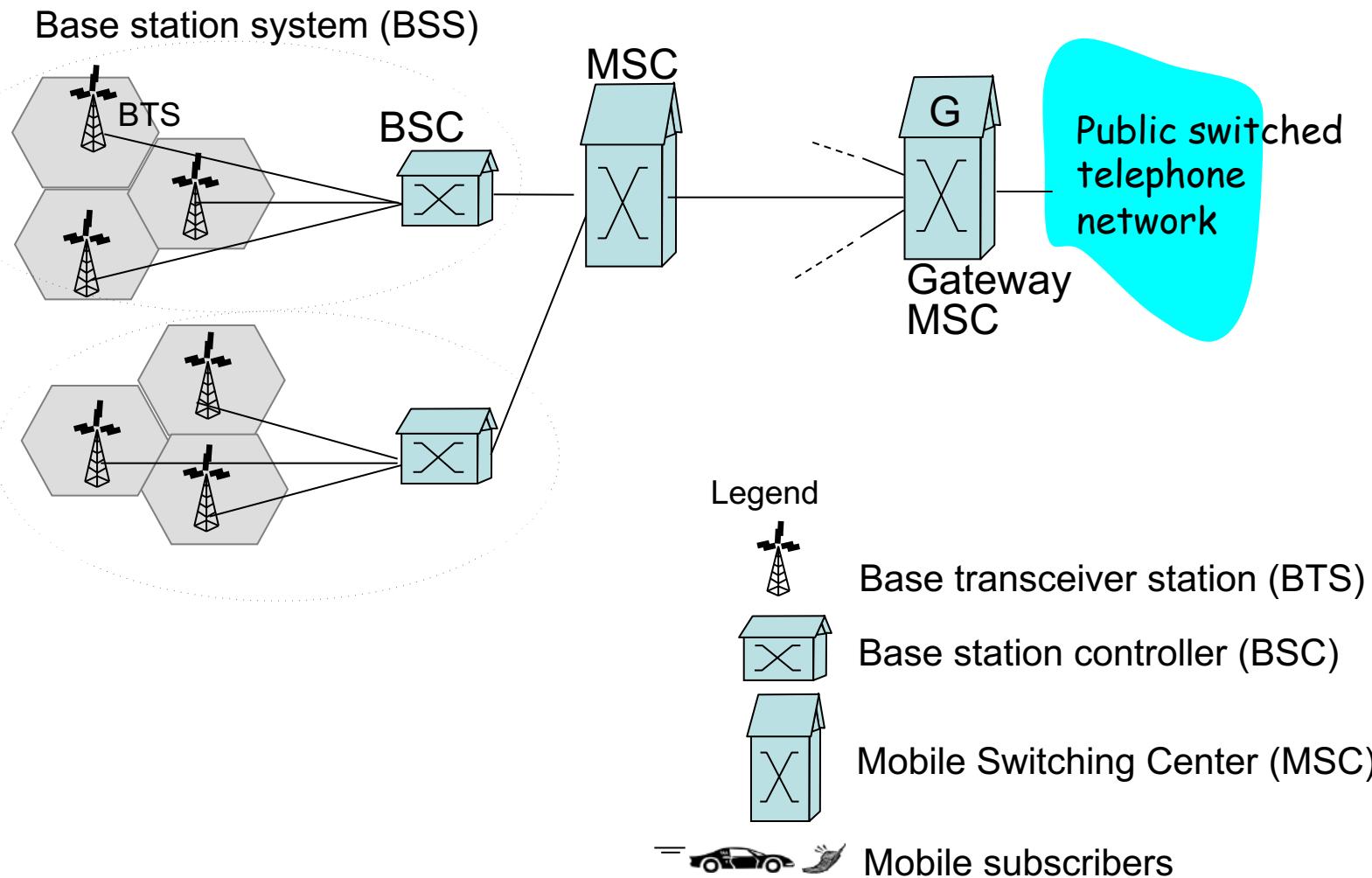


Don't drown in a bowl of alphabet soup: use this for reference only

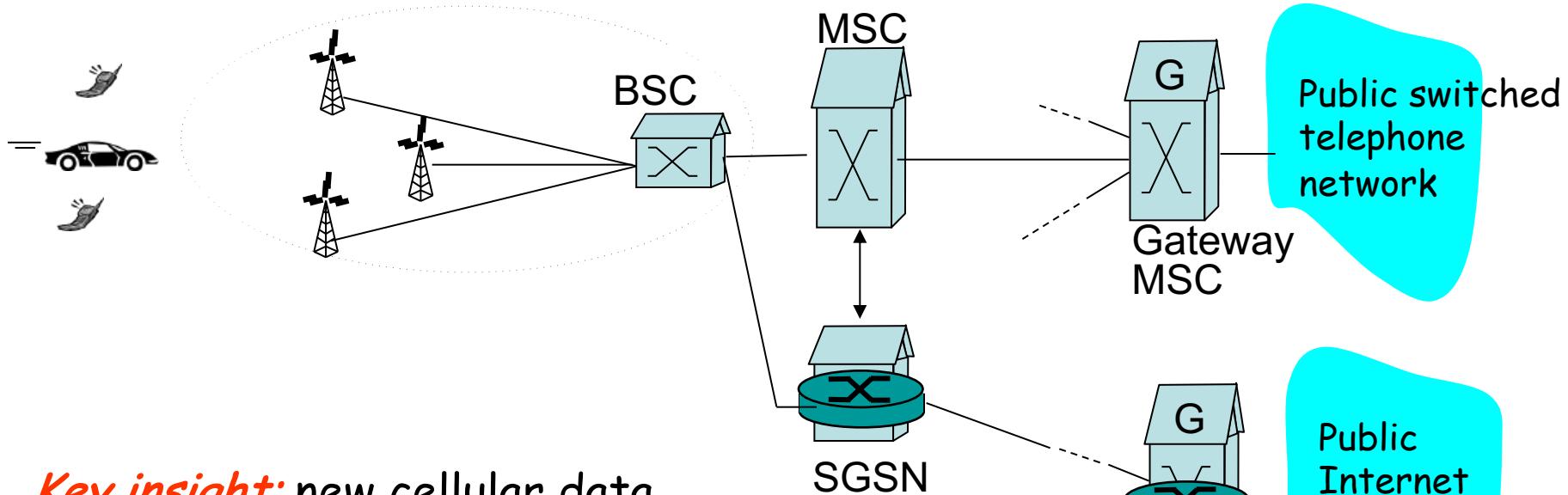
Cellular Standards: Brief Survey

- **2.5G systems: voice and data channels**
 - 2G extensions: for those who can't wait for 3G
 - General packet radio service (GPRS)
 - Evolved from GSM
 - Data sent on multiple channels (if available)
 - Enhanced data rates for global evolution (EDGE)
 - Also evolved from GSM, using enhanced modulation
 - Data rates up to 384kbps
 - CDMA-2000 (phase 1)
 - Data rates up to 144kbps
 - Evolved from IS-95

2G (voice) Network Architecture

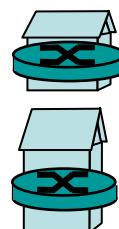


2.5G (Voice+Data) Network Architecture



Key insight: new cellular data network operates *in parallel* with existing cellular voice network

- ❑ voice network unchanged in core
- ❑ data network operates in parallel



Serving GPRS Support Node (SGSN)



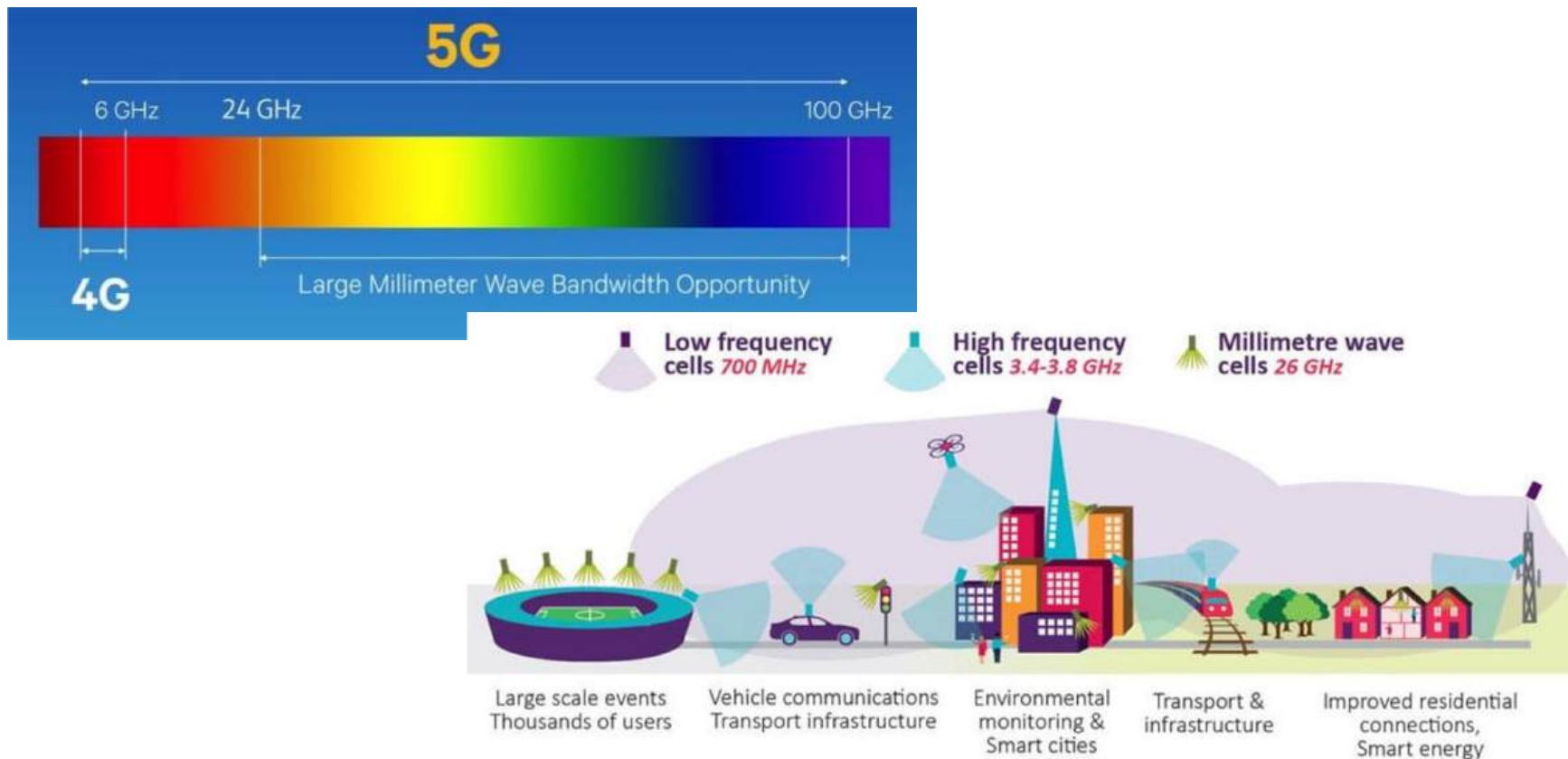
Gateway GPRS Support Node (GGSN)

Cellular Standards: Brief Survey

- **3G systems: voice + data**
 - Universal Mobile Telecommunications Service (UMTS)
 - CDMA in TDMA slots
 - Data service: up to 14 Mbps
- **4G systems: data**
 - All-IP network: voice in data packets
 - New wireless access technologies: OFDM, MIMO, etc
 - Data rate: up to 300 Mbps

Cellular Standards: Brief Survey

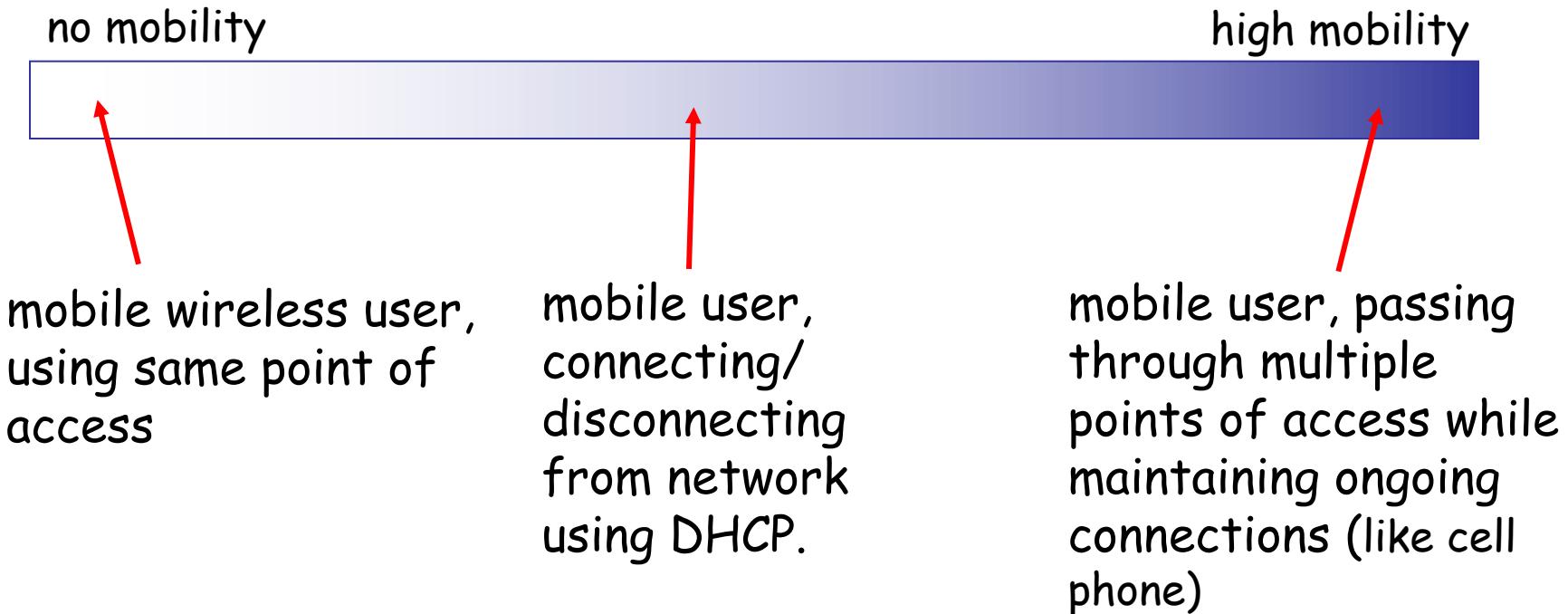
- **5G systems: massive data**
 - Microwaves (4G) + millimeter waves (high bandwidth)



Mobility (Roaming) Handling

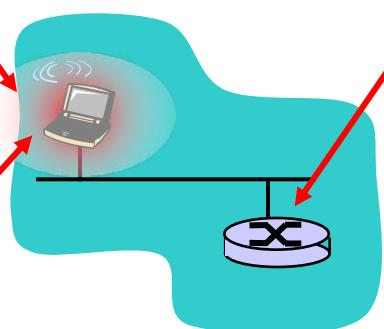
What Is Mobility?

- **Spectrum of mobility, from the network perspective:**

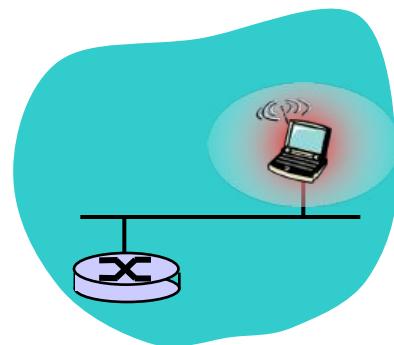


Mobility: Vocabulary

home network: permanent
"home" of mobile
(e.g., 128.119.40/24)



home agent: entity that will perform mobility functions on behalf of mobile, when mobile is remote



Permanent address:
address in home
network, can always be
used to reach mobile
e.g., 128.119.40.186

wide area
network

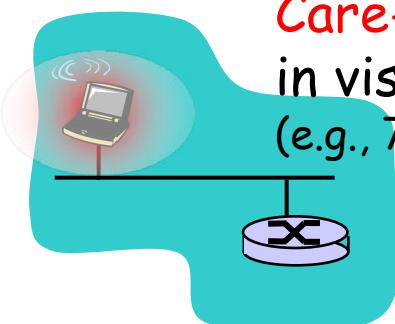

correspondent

Mobility: Vocabulary

Permanent address: remains constant (e.g., 128.119.40.186)

visited network: network in which mobile currently resides (e.g., 79.129.13/24)

Care-of-address: address in visited network.
(e.g., 79.129.13.2)



correspondent: wants to communicate with mobile



foreign agent: entity in visited network that performs mobility functions on behalf of mobile.

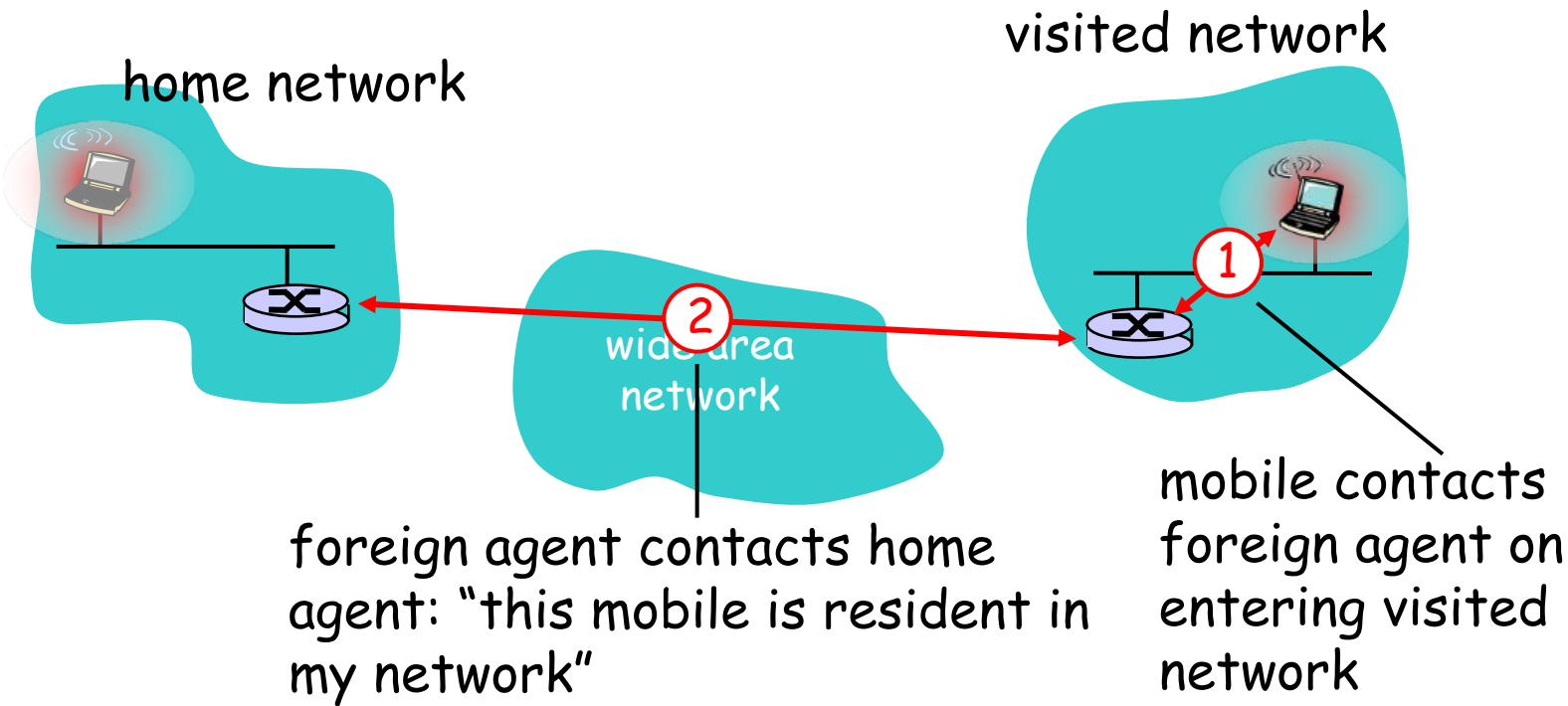
Mobility: Approaches

- **Let routing handle it:** routers advertise permanent address of mobile-nodes-in-residence via usual routing table exchange
 - Routing tables indicate where each mobile located
 - No changes to end-systems
- **Let end-systems handle it:**
 - **Indirect routing:** communication from correspondent to mobile goes through home agent, then forwarded to remote
 - **Direct routing:** correspondent gets foreign address of mobile, sends directly to mobile

Mobility: Approaches

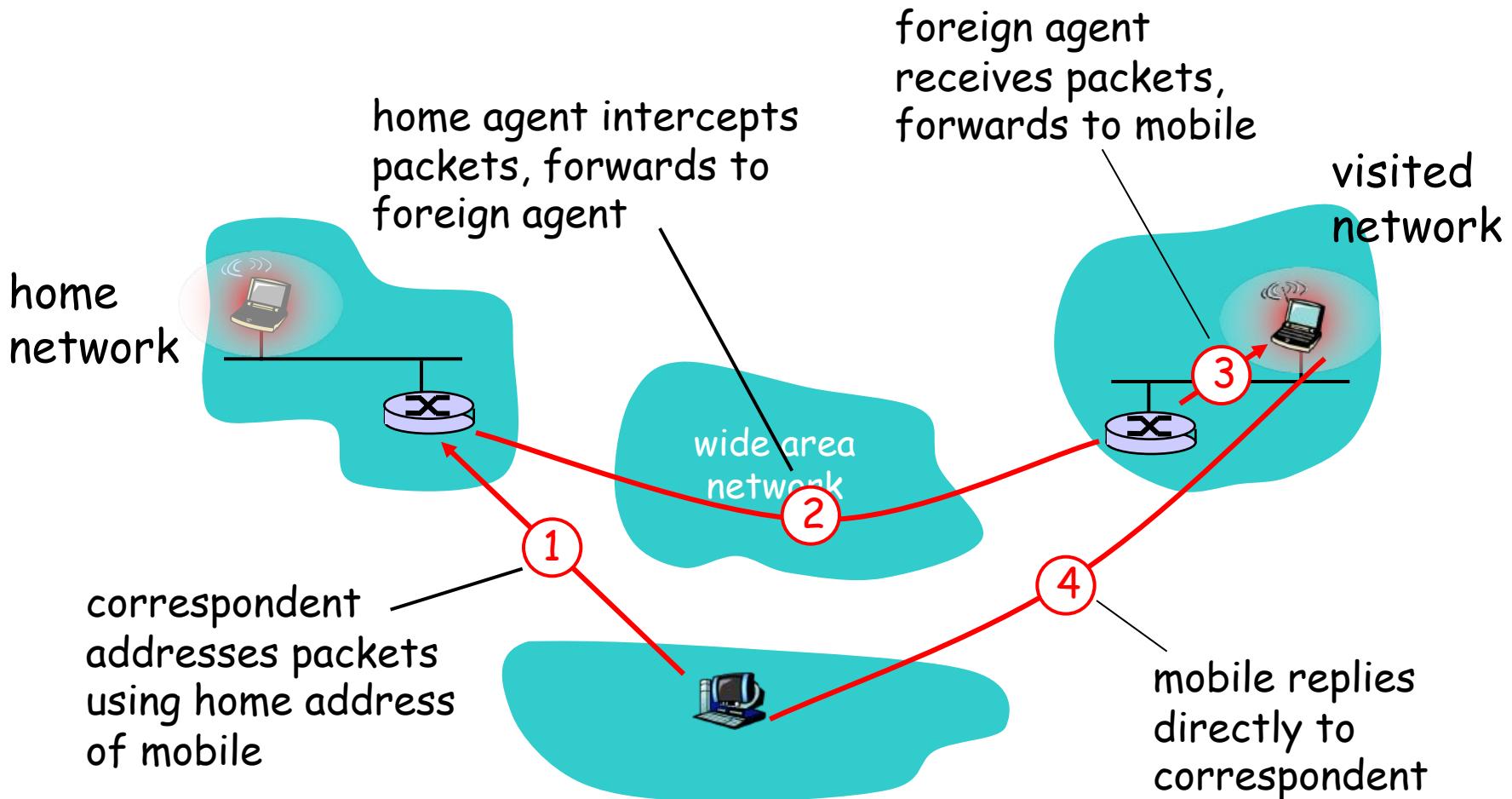
- Let routing handle it: routers advertise permanent addresses of mobile-nodes-in-residence via routing table exchange
 - Routing tables must store each mobile located
 - No changes to end-systems
- **Let end-systems handle it:**
 - **Indirect routing:** communication from correspondent to mobile goes through home agent, then forwarded to remote
 - **Direct routing:** correspondent gets foreign address of mobile, sends directly to mobile

Mobility: Registration



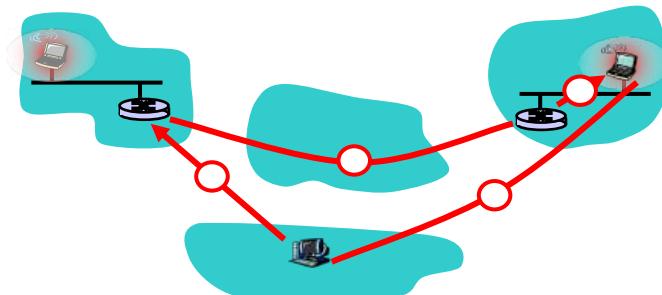
- **End result:**
 - Foreign agent knows about mobile
 - Home agent knows location of mobile

Mobility via Indirect Routing



Indirect Routing: Comments

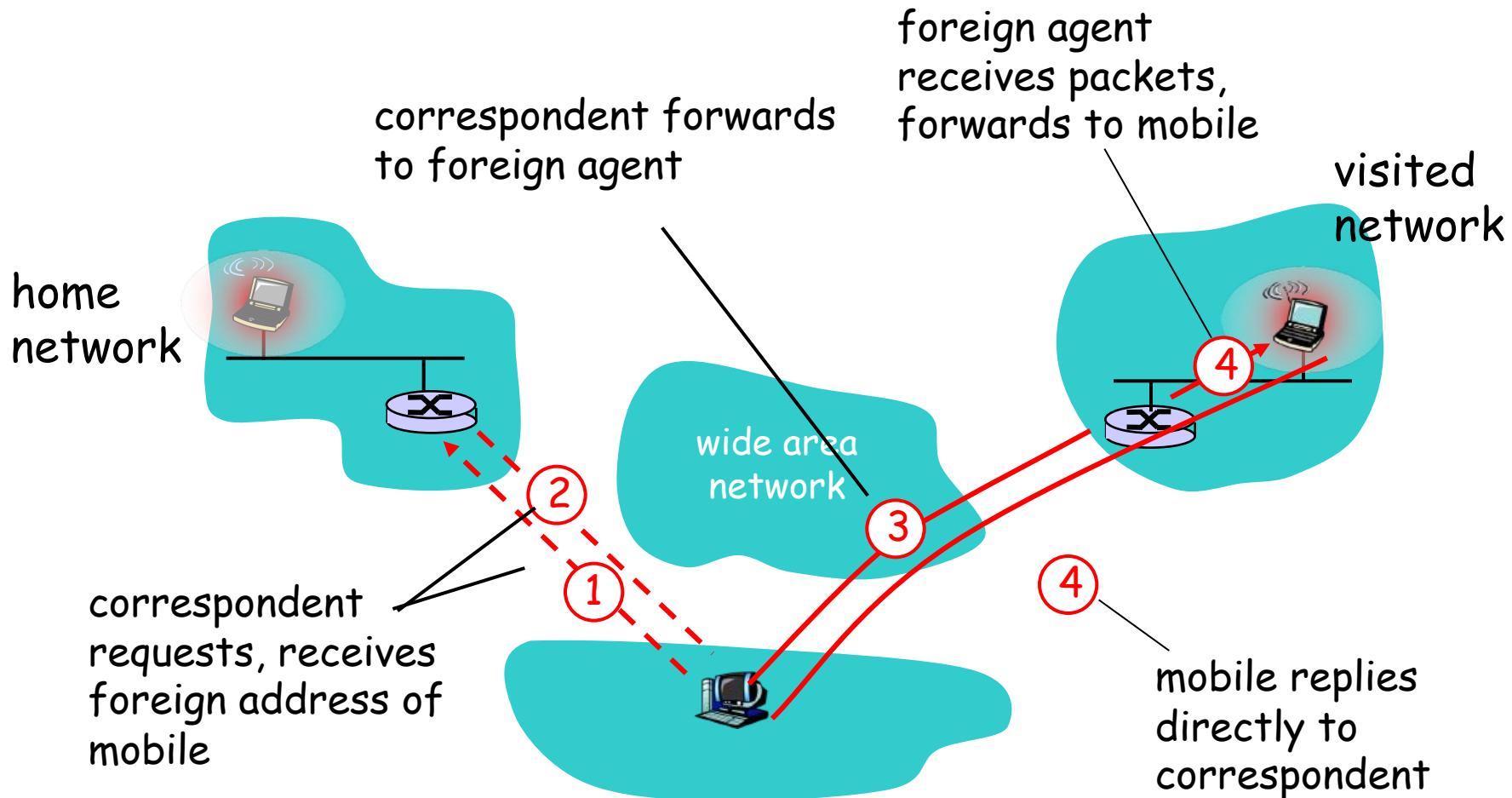
- **Mobile uses two addresses**
 - Permanent address: used by correspondent (hence mobile location is **transparent** to correspondent)
 - Care-of-address (**Foreign address**): used by home agent to forward datagrams to mobile
- **Triangle routing: correspondent-home network-mobile**
 - Inefficient when correspondent & mobile are in same network



Indirect Routing: Moving Between Networks

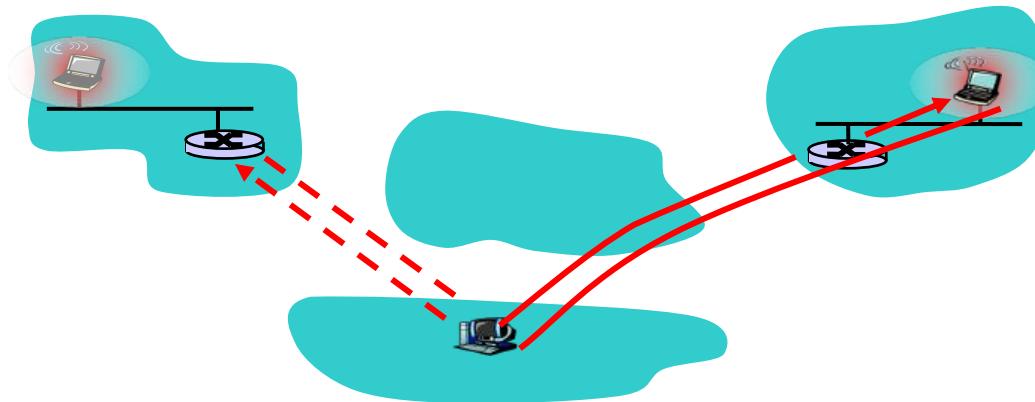
- **Suppose mobile user moves to another network**
 - Register with new foreign agent
 - New foreign agent registers with home agent
 - Home agent update care-of-address for mobile
 - Packets continue to be forwarded to mobile (but with new care-of-address)
- **Mobility, changing foreign networks**
 - Transparent: ongoing connections can be maintained!

Mobility via Direct Routing



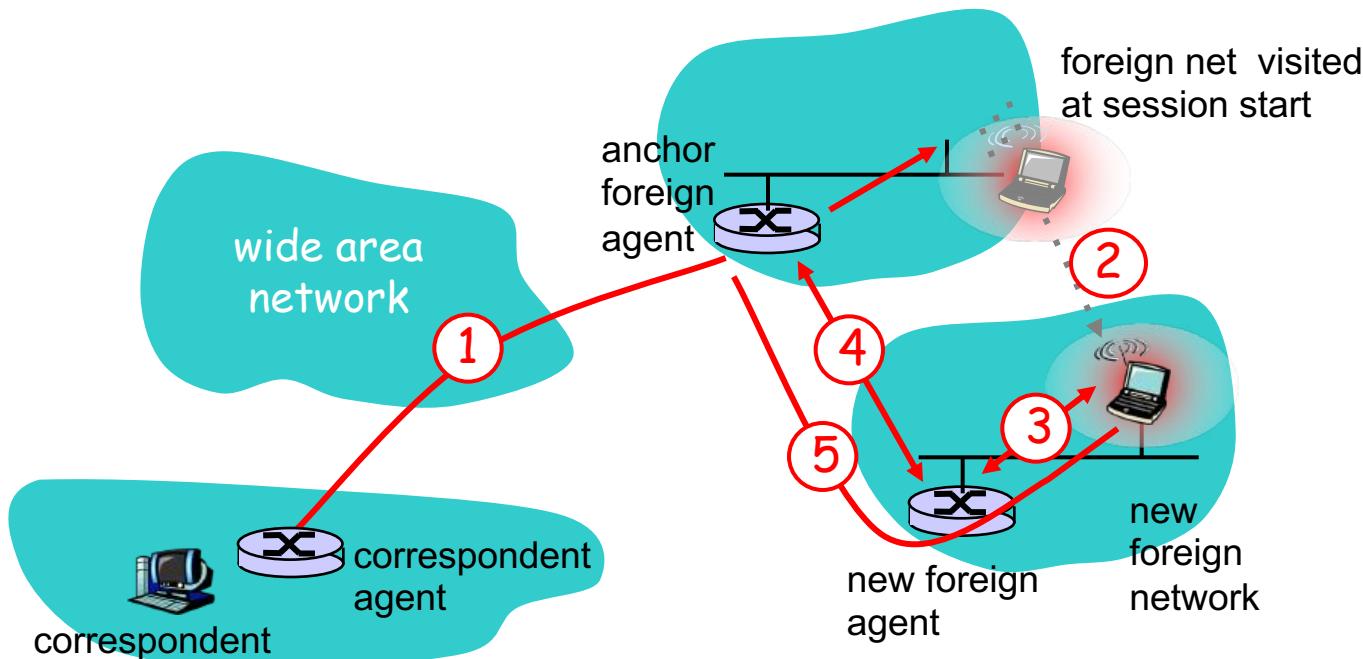
Mobility via Direct Routing: Comments

- Overcome triangle routing problem
- **Non-transparent to correspondent:** correspondent must get care-of-address from home agent
 - What if mobile changes visited network?



Accommodating Mobility with Direct Routing

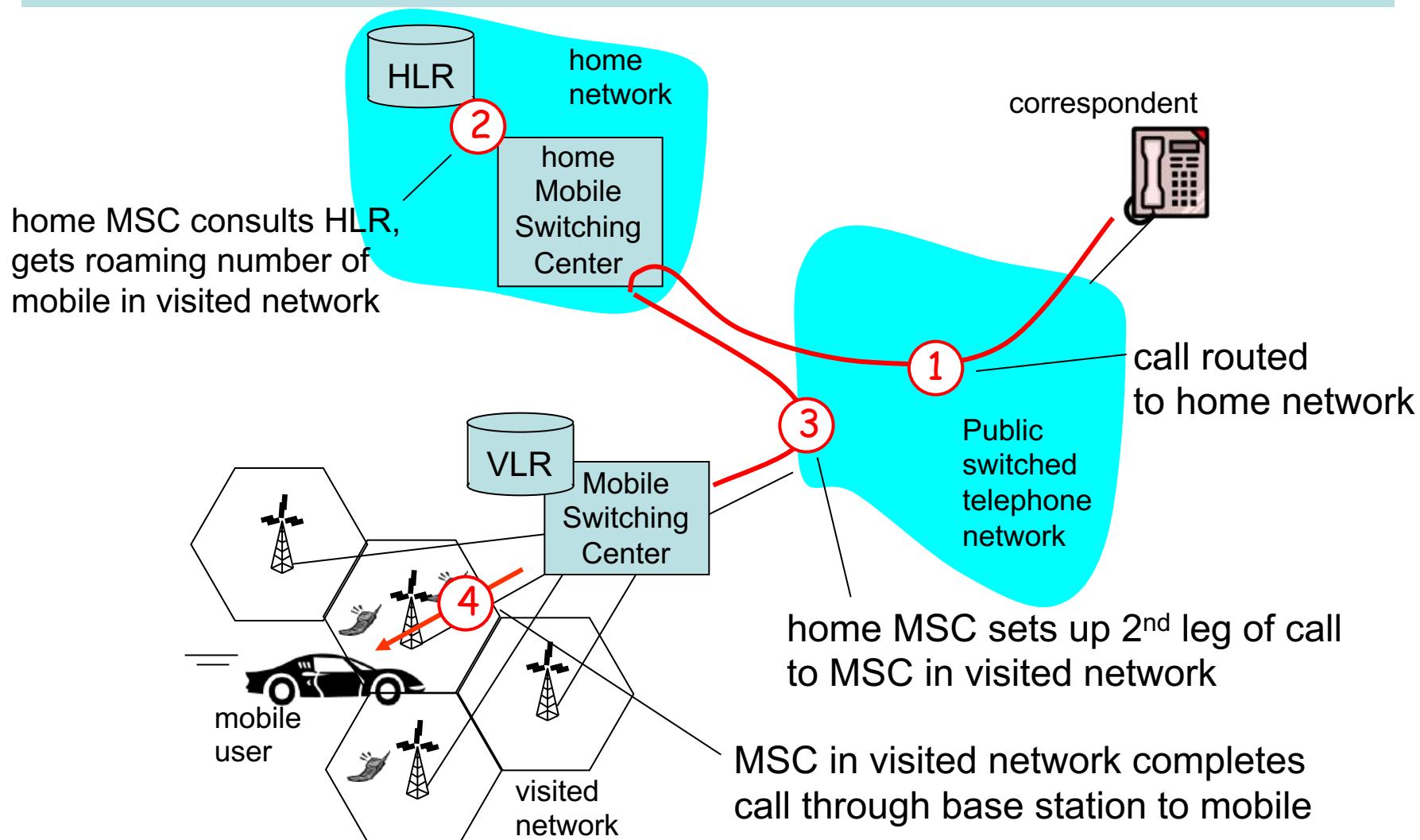
- Anchor foreign agent: FA in first visited network
- Data always routed first to anchor FA
- When mobile moves: new FA arranges to have data forwarded from old FA (chaining)



Handling Mobility in Cellular Networks

- **Home network:** network of cellular provider you subscribe to (e.g., SingTel, M1)
 - Home location register (HLR): database in home network containing permanent cell phone #, profile information (services, preferences, billing), information about current location (could be in another network)
- **Visited network:** network in which mobile currently resides
 - Visitor location register (VLR): database with entry for each user currently in network
 - Could be home network

GSM: Indirect Routing to Mobile



Wireless, Mobility: Impact on Higher Layer Protocols

- **Logically, impact should be minimal ...**
 - Best effort service model remains unchanged
 - Higher layers can (and do) run over wireless, mobile
- **... but performance-wise**
 - Packet loss/delay due to bit-errors (discarded packets, delays for link-layer retransmissions), and handoff
 - Delay impairments for real-time traffic
 - Limited bandwidth of wireless links

Learning Objectives

- **Wireless Link Standards**
 - Read the characteristics of the standards
- **Cellular Networks**
 - Understand the network architecture
 - Understand the key features of 1G to 5G
- **Mobility (roaming) handling**
 - Read mobility vocabulary
 - Understand indirect and direct routing

Part I Syllabus - Fundamental Underlying Layers

Date	Subject	File
Week 1: 9/Jan/2023 11/Jan/2023	Introduction: course logistics and Internet history	M1-L1-Introduction.pptx
	Layered Network Architecture	First part of M1-L2-Network Layer & Physical Resilience.pptx
Week 2: 16/Jan/2023 18/Jan/2023	Physical Layer: Network Resilience	Second part of M1-L2-Network Layer & Physical Resilience.pptx
	Data link layer – Flow control	M1-L3-DLL-Flow Control.pptx
Week 3: 25/Jan/2023	Data link layer – Error control	M1-L4-DLL-Error Control.pptx
Week 4: 30/Jan/2023 01/Feb/2023	Local area network – Introduction	M1-L5-LAN-Introduction.pptx
	Local area network – MAC	M1-L6-LAN-MAC.pptx
Week 5: 06/Feb/2023 08/Feb/2023	Local area network – Ethernet	First part of M1-L7-LAN-Ethernet.pptx
	Local area network – Ethernet Evolutions	Second part of M1-L7-LAN-Ethernet.pptx
Week 6: 13/Feb/2023 15/Feb/2023	Local area network – WLAN	M1-L8-LAN-WLAN.pptx
	Mobile Access Networks	M1-L9-Mobile.pptx
Week 7: 20/Feb/2023 22/Feb/2023	E-learning for Network paradigms	M1-L10-Paradigms.pptx
	Network paradigms	M1-L10-Paradigms.pptx

Additional Materials

- **The related content talked today in [https://eclasse.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20\(Lectures\)/Computer_Networking_A_Top-Down_Approach.pdf](https://eclasse.teicrete.gr/modules/document/file.php/TP326/%CE%98%CE%B5%CF%89%CF%81%CE%AF%CE%B1%20(Lectures)/Computer_Networking_A_Top-Down_Approach.pdf) is as follow:**
 - Circuit Switching: Page 27 – Page 32
 - Packet Switching: Page 22 – Page 27
 - Delay, Loss and Throughput: Page 35 - Page 47
- **You can also find other video materials about**
 - Switching Techniques <https://www.youtube.com/watch?v=-HIJ4psu5aU>
 - Delay <https://www.youtube.com/watch?v=wo3M5G9ZHo0>

How do you get your water?



VS



SC2008/CZ3006/CE3005

Computer Network

Lecture 10

Network Paradigms

(Not Examinable)



Contents

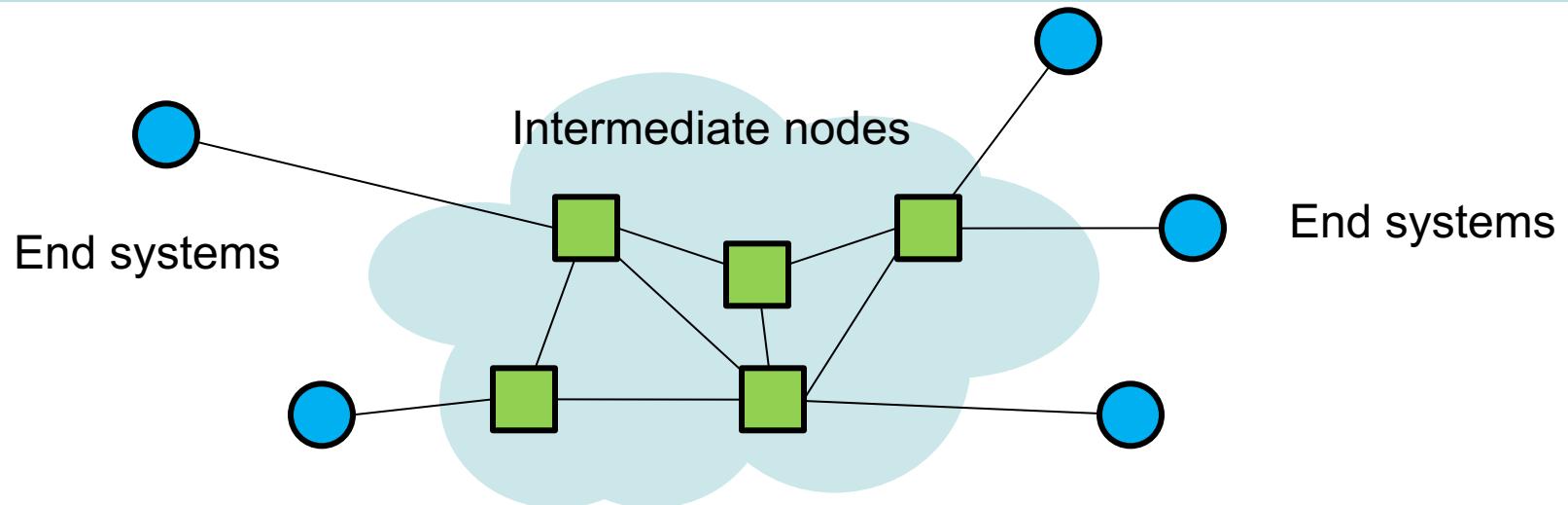
- **Data Transmission Technologies**
 - Network Design Approaches
 - Circuited Switched Network
 - Packet Switched Network
 - Connection-Oriented vs Connectionless
- **Delay in Packet Switched Network**
 - Delay components in packet switched network
 - Transmission delay calculation
 - Pipeline

Data Transmission Paradigms

Overview

- Networks are used to interconnect many devices: LAN/WAN
- Now, wide area networks
 - Since the invention of the telephone, **circuit switching** has been the dominant technology for voice communications.
 - Since 1970, **packet switching** has evolved substantially for digital data communications. It was designed to provide a more efficient facility than circuit switching for bursty data traffic.
 - Two types of packet switching:
 - Datagram (such as today's Internet)
 - Virtual circuit (such as Frame Relay, Asynchronous Transfer Mode (ATM))

Network Design



Approach 1: Smart intermediate nodes & dumb end systems

- E.g., telecommunication networks
- Achieve very complex traffic management but difficult to upgrade

Approach 2: Dumb intermediate nodes & smart end systems

- E.g., the Internet
- Attract innovation at applications but difficult to manage traffic

(Intermediate) Switching Nodes

- **Nodes may connect to other nodes, or to some stations.**
- **Network is usually partially connected**
 - However, some redundant connections are desirable for reliability
- **Two different switching technologies**
 - Circuit switching
 - Packet switching

Communication Networks Taxonomy

Communication Networks

**Circuit Switched
(Telephone Calls)**

**Packet Switched
(Data Communications)**

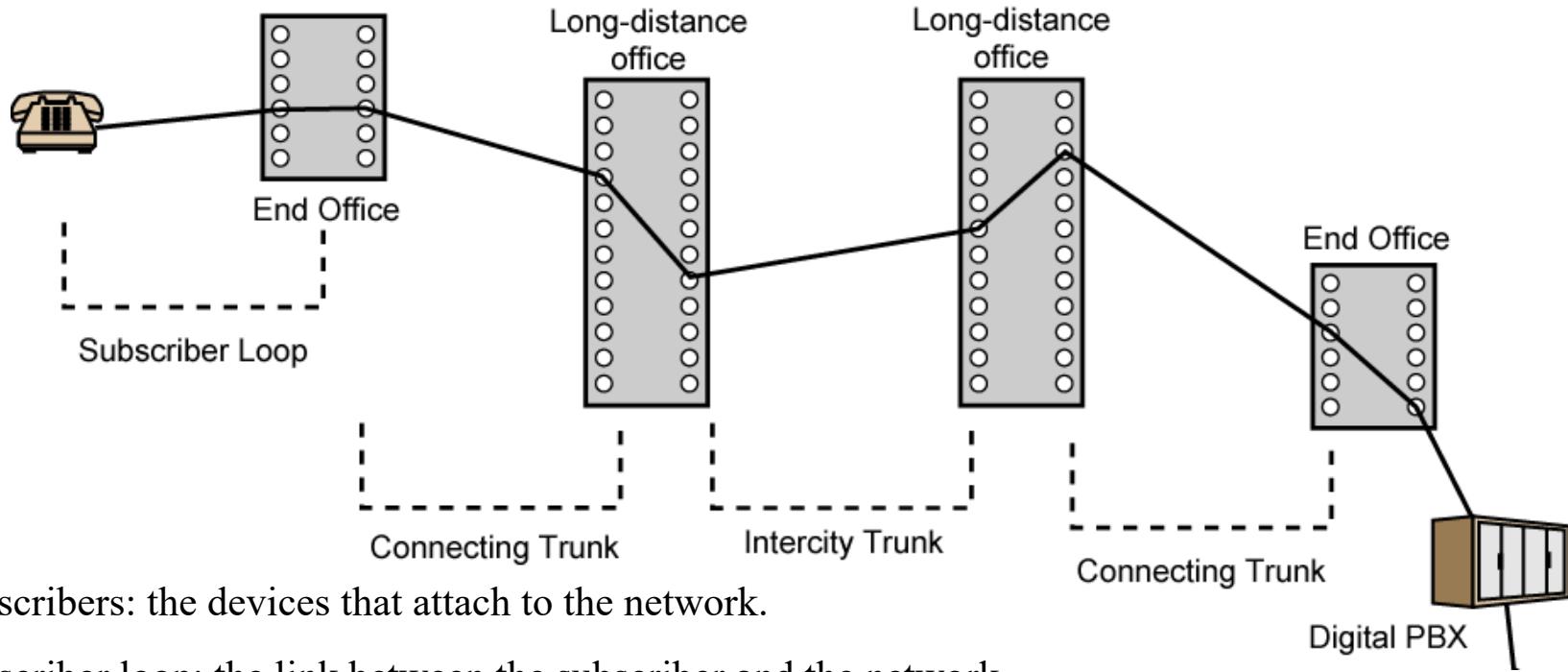
**Virtual Circuit Switching
(Connection Oriented)
eg. X.25, ATM, etc**

**Datagram Switching
(Connectionless)
eg. Internet**

Circuit Switching

- **Circuit switching:**
 - There is a dedicated communication path between two stations (end-to-end)
 - The path is a connected sequence of links between network nodes. On each physical link, a logical channel is dedicated to the connection.
- **Communication via circuit switching has three phases:**
 - Circuit establishment (link by link)
 - Routing & resource allocation (FDM or TDM)
 - Data transfer
 - Circuit disconnect
 - Deallocate the dedicated resources
- **The switches must know how to find the route to the destination and how to allocate bandwidth (channel) to establish a connection.**

Public Switched Telephone Network (PSTN)



Subscribers: the devices that attach to the network.

Subscriber loop: the link between the subscriber and the network.

Exchanges: the switching centers in the network.

End office: the switching center that directly supports subscribers.

Trunks: the branches between exchanges. They carry multiple voice-frequency circuits using either FDM or synchronous TDM.

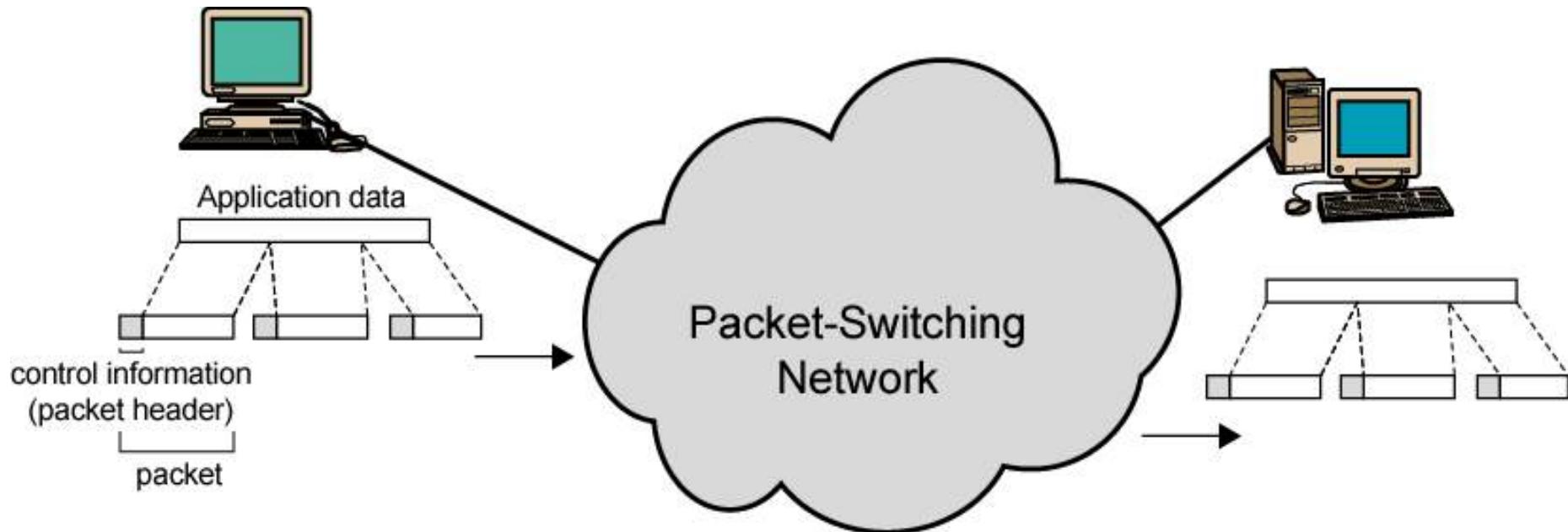
Packet Switching Principles

- **Problem of circuit switching**
 - Designed for voice service
 - Resources dedicated to a particular call
 - For data transmission, much of the time the connection is idle (say, web browsing)
 - Data rate is fixed
 - Both ends must operate at the same rate during the entire period of connection
- **Packet switching is designed to address these problems.**

Basic Operation

- **Data are transmitted in short packets**
 - Typically at the order of 1000 bytes
 - Longer messages are split into series of packets
 - Each packet contains a portion of user data plus some control info
- **Control info contains at least**
 - Routing (addressing) info, so as to be routed to the intended destination
- **Store and Forward**
 - On each switching node, packets are received, stored briefly (buffered) and passed on to the next node.

Use of Packets



Advantages of Packet Switching

- **Line efficiency**
 - Single node-to-node link can be dynamically shared by many packets over time
 - Packets are queued up and transmitted as fast as possible
- **Data rate conversion**
 - Each station connects to the local node at its own speed
- **In circuit-switching, a connection could be blocked if there lacks free resources. On a packet-switching network, even with heavy traffic, packets are still accepted, yet causing an increased delay.**
- **Priorities can be used**
 - On each node, packets with higher priority can be forwarded first. They will experience less delay than lower-priority packets.

Packet Switching Technique

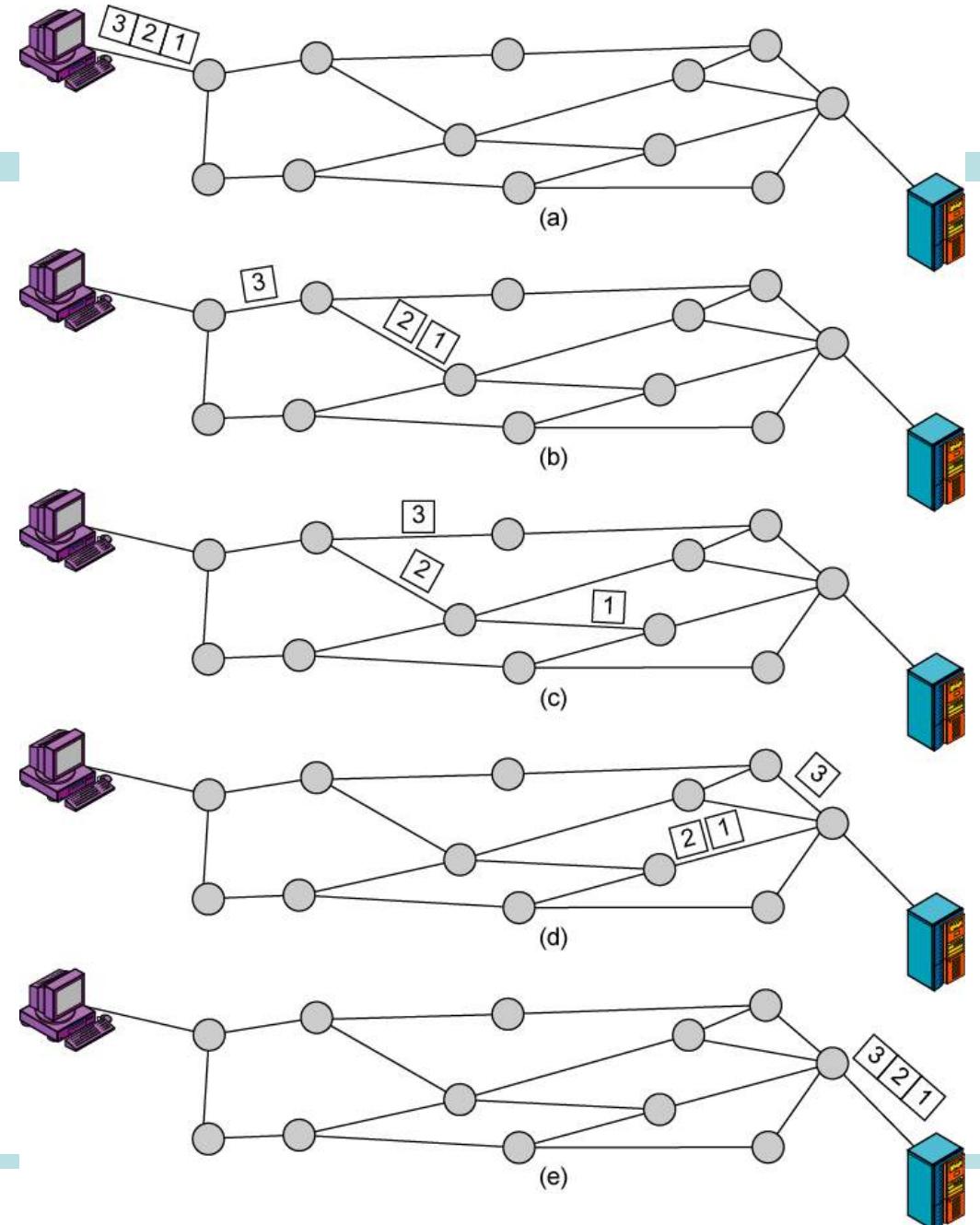
- A station breaks long message into packets
- Packets are sent out to the network sequentially, one at a time
- How will the network handle this stream of packets as it attempts to route them through the network and deliver them to the intended destination?
 - Two alternative approaches
 - Datagram approach
 - Virtual circuit approach

Datagram

- Each packet is treated independently, with no reference to packets that have gone before.
 - Each node chooses the next node on a packet's path.
- Packets can take any possible route.
- Packets may arrive at the receiver out of order.
- Packets may go missing.
- It is up to the receiver to re-order packets and recover from missing packets.
- Example: **Internet**

Datagram

- Each packet carries the full destination address.
- Each packet is treated independently.
- Packets may arrive out of sequence.
- Packets are called datagrams.



Virtual Circuit

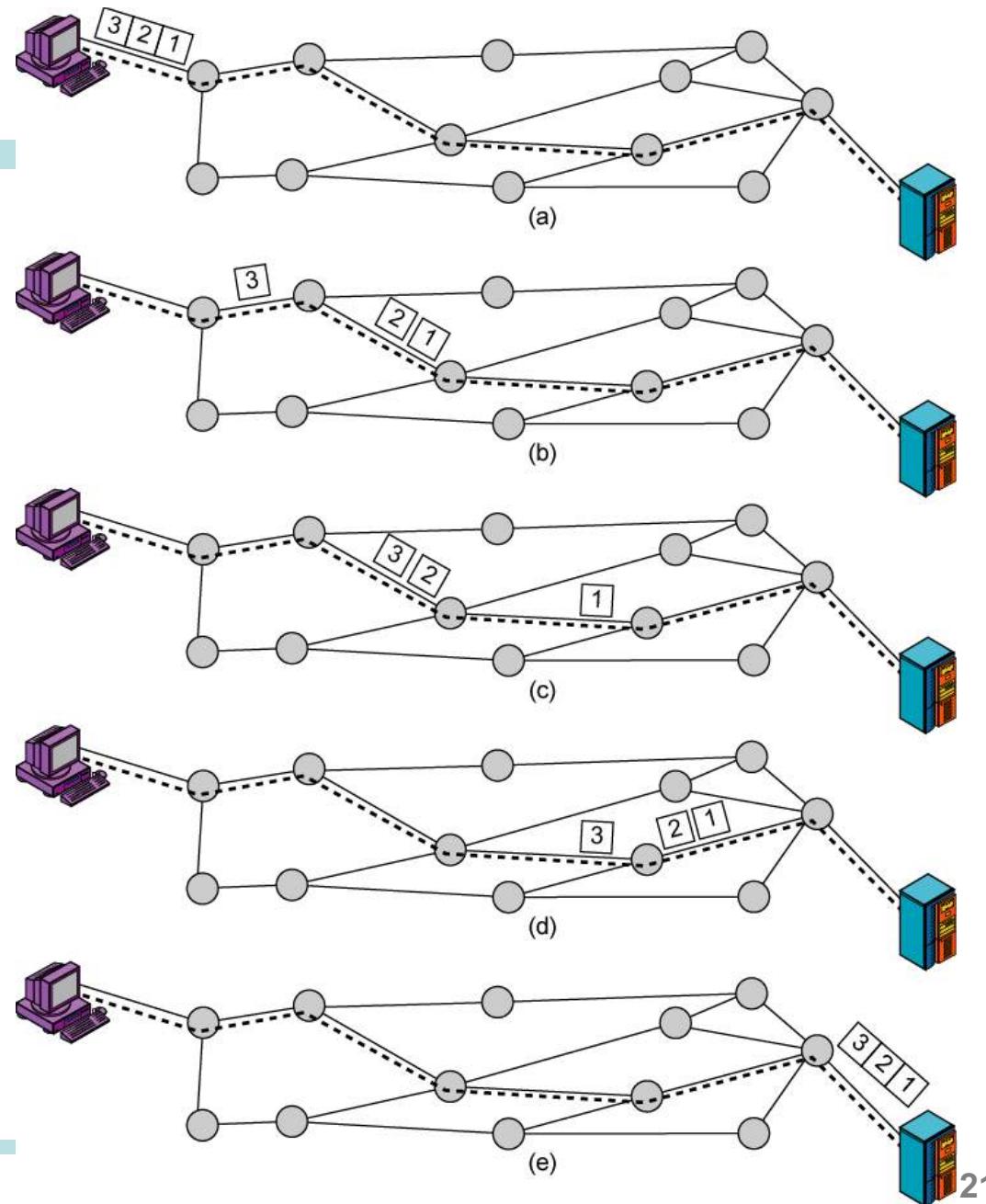
- In virtual circuit, a pre-planned route is established before any packets are sent, then all packets follow the same route.
- Each packet contains a **virtual circuit identifier** instead of destination address, and each node on the pre-established route knows where to forward such packets.
 - The node need not make a routing decision for each packet.
- Example: X.25, Frame Relay, ATM

Virtual Circuit

A route between stations is set up prior to data transfer.

All the data packets then follow the same route.

But there is no dedicated resources reserved for the virtual circuit! Packets need to be stored-and-forwarded.



Virtual Circuits v Datagram

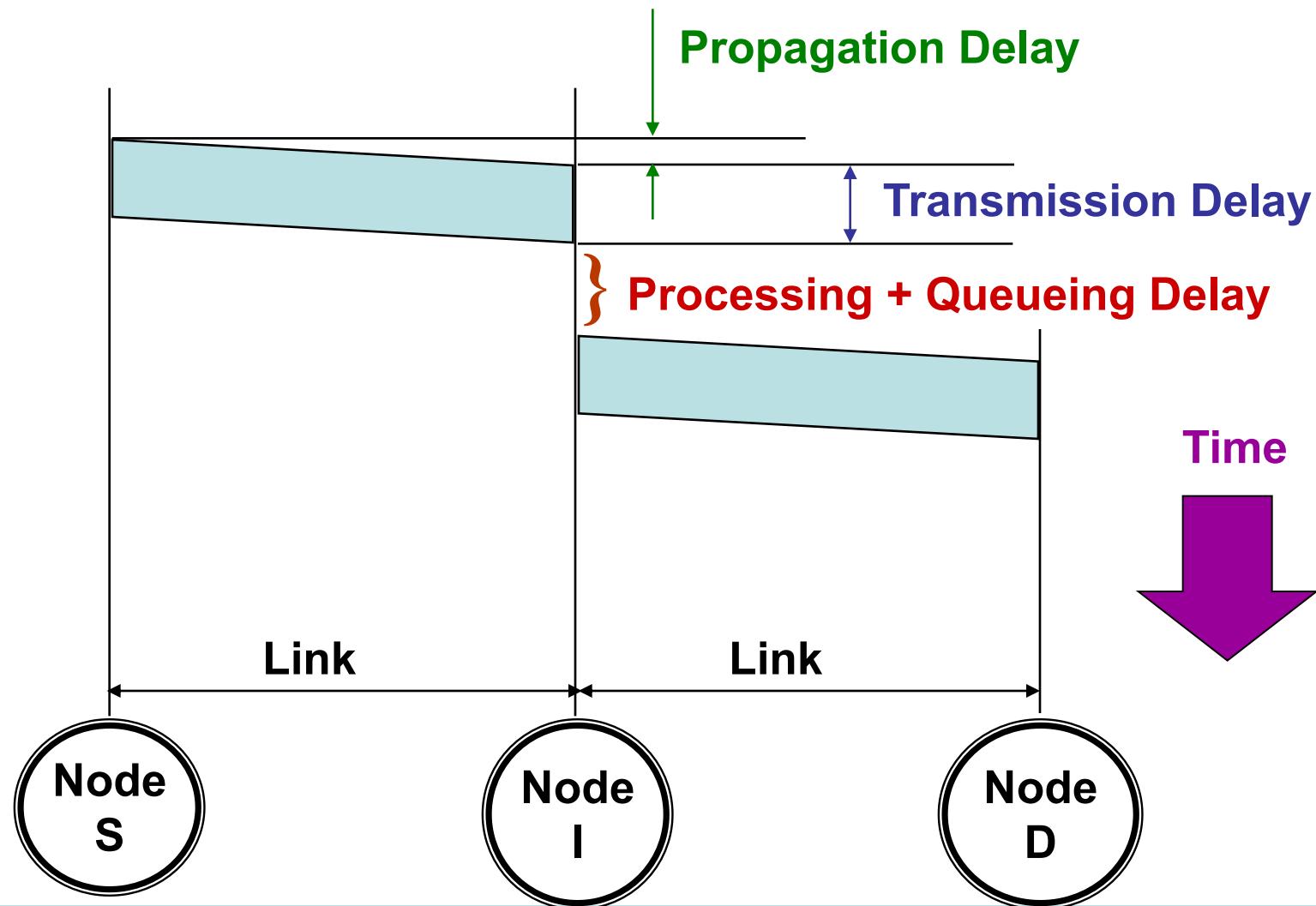
- **Virtual circuits**
 - Network can provide sequencing (packets arrive at the same order) and error control (retransmission between two nodes).
 - Packets are forwarded more quickly
 - Based on the virtual circuit identifier
 - No routing decisions to make
 - Less reliable
 - If a node fails, all virtual circuits that pass through that node fail.
- **Datagram**
 - No call setup phase
 - Good for bursty data, such as Web applications
 - More flexible
 - If a node fails, packets may find an alternate route
 - Routing can be used to avoid congested parts of the network

Comparison of communication switching techniques

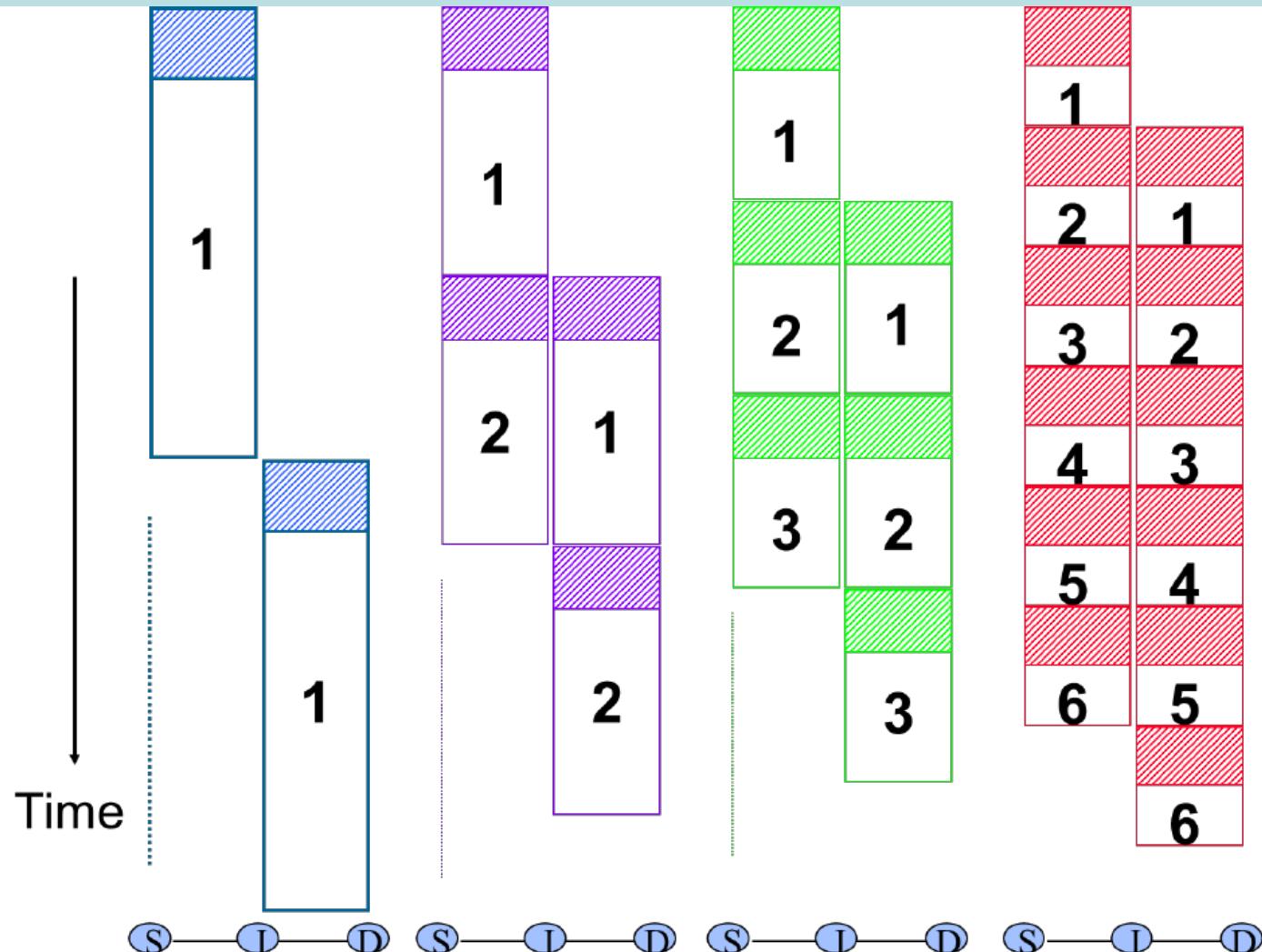
Circuit Switching	Datagram Packet Switching	Virtual Circuit Packet Switching
Dedicated transmission path	No dedicated path	No dedicated path
Continuous transmission of data	Transmission of packets	Transmission of packets
Fast enough for interactive	Fast enough for interactive	Fast enough for interactive
Messages are not stored	Packets may be stored until delivered	Packets stored until delivered
The path is established for entire conversation	Route established for each packet	Route established for entire conversation
Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay; packet transmission delay
Busy signal if called party busy	Sender may be notified if packet not delivered	Sender notified of connection denial
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Electromechanical or computerized switching nodes	Small switching nodes	Small switching nodes
User responsible for message loss protection	Network may be responsible for individual packets	Network may be responsible for packet sequences
Usually no speed or code conversion	Speed and code conversion	Speed and code conversion
Fixed bandwidth	Dynamic use of bandwidth	Dynamic use of bandwidth
No overhead bits after call setup	Overhead bits in each packet	Overhead bits in each packet

Packet Transmission Time

Delay in Packet Switched Networks



Pipeline Effect

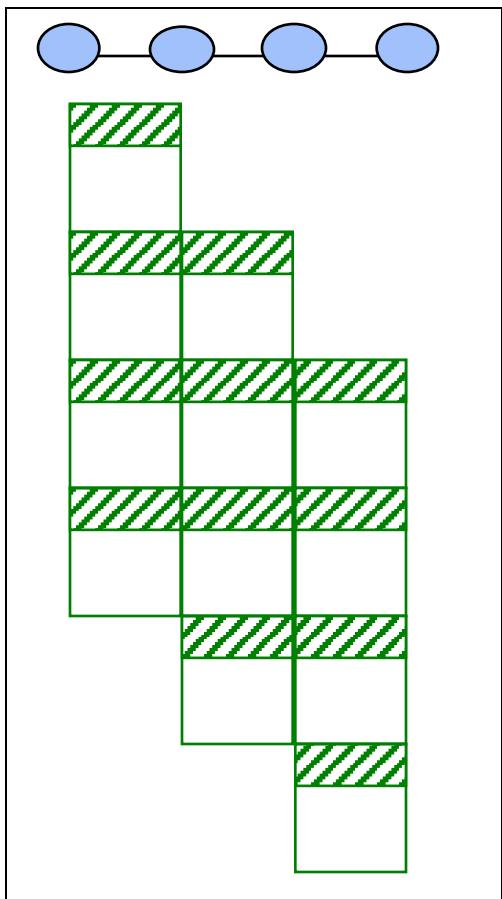


Packet Transmission Time: Overhead vs. Pipeline

Example:

3 hops & 4 fragments

$$\therefore \text{Tx time} = 4 T_{frame} + (3-1) T_{frame}$$



In general:

Tx Delay

$$\begin{aligned} &= \text{Tx Delay of all packets in first hop} \\ &+ (\# \text{ of hops}-1) * \text{Tx Delay of 1 packet} \end{aligned}$$

To find the optimum packet size, other delays should also be considered:

- Processing Delay
- Queueing Delay
- Signal Propagation Delay

Packet Size (Transmission Time Consideration)

- **Example 1:** A packet of size 1000 bits would have transmission delay of 100 msec on a 10 Kbps link.
- **Example 2:** Consider a VC from node S to D through an intermediate node I (two hops). Link rate = 8 bps. Message is of 30 bytes. Header of a packet is 3 bytes.
 - Case-1: Message is transmitted as a single packet.
Tx. Delay = 33 bytes * 2 packet Tx / link rate = 66 sec
 - Case-2: Message is transmitted in 2 packets.
Tx. Delay = 18 bytes * 3 packet Tx / link rate = 54 sec
 - Case-3: Message is transmitted in 3 packets.
Tx. Delay = 13 bytes * 4 packet Tx / link rate = 52 sec
 - Case-4: Message is transmitted in 6 packets.
Tx. Delay = 8 bytes * 7 packet Tx / link rate = 56 sec

Transmission Time



Transmission time
 $= 13*8 \text{ bits} / 8 \text{ bps} = 13 \text{ s}$

Total Transmission time
 $= 4*13 \text{ s} = 52 \text{ s}$

Learning Objectives

- **Data Transmission Technologies**
 - Understand difference between circuit and packet switched networks
 - Understand difference between datagram and virtual circuit switching
- **Delay in Packet Switched Networks**
 - Understand delay components in PSN
 - Calculate packet transmission delay

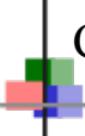


NANYANG
TECHNOLOGICAL
UNIVERSITY

CE3005: Computer Networks
CZ3006: Netcentric Computing
Part 2

Prof. Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

School of Computer Science and Engineering



CE3005/CZ3006 Computer Networks Part II

Focus of Part II Lectures:	Application	Web (HTTP)	E-mail (SMTP)	Others (FTP, IM)
	Transport	TCP		
	Network	Internet IP		
Part I Lectures:	Data Link	LAN (Ethernet)	Backbone (Ethernet, ATM, FDDI)	Wireless LAN (802.11b, a, g)
	Physical	MAN/WAN (POTS, ISDN, T1, SONET, ATM, Frame Relay)		Internet (DSL, Cable Modem)
LAN		WAN		



NANYANG
TECHNOLOGICAL
UNIVERSITY

CE3005: Computer Networks
CZ3006: Netcentric Computing

An Overview of Computer Networks
and the Internet

Prof.Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

School of Computer Science and Engineering

<https://www.straitstimes.com/tech/singapore-tops-fixed-broadband-speed-rankings-but-places-4th-for-mobile>

World's fastest broadband nations

Rank	Place	Average download speed (Mbps)	Download a 7.5GB HD film (Minutes:Seconds)
1	Singapore	55.13	18:34
2	Sweden	40.16	25:30
3	Taiwan	34.4	29:46
4	Denmark	33.54	30:32
5	The Netherlands	33.52	30:33
6	Latvia	30.36	33:43
7	Norway	29.13	35:09
8	Belgium	27.37	37:25
9	Hong Kong	27.16	37:42
10	Switzerland	26.93	38:01
11	Lithuania	25.12	40:46
12	Japan	24.47	41:51
13	Estonia	24.11	42:28
14	Jersey, United Kingdom	23.3	43:57
15	Hungary	23.16	44:12
16	Republic of Korea	22.9	44:43

Speedtest Global Index

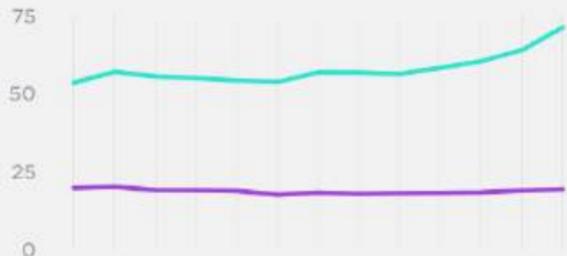
Ranking mobile and fixed broadband speeds from around the world on a monthly basis.

For an in-depth analysis of how COVID-19 has affected global internet performance, read [this article](#).

← Singapore November 2020

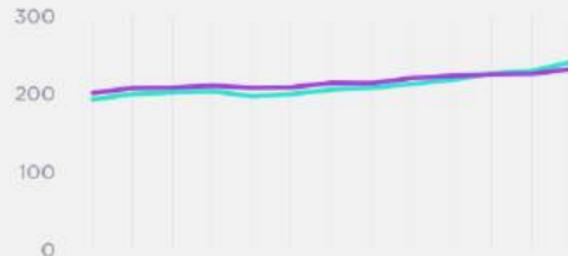
Mobile

Rank	Download	Upload	Latency
16	71.32 Mbps	19.55 Mbps	20 ms



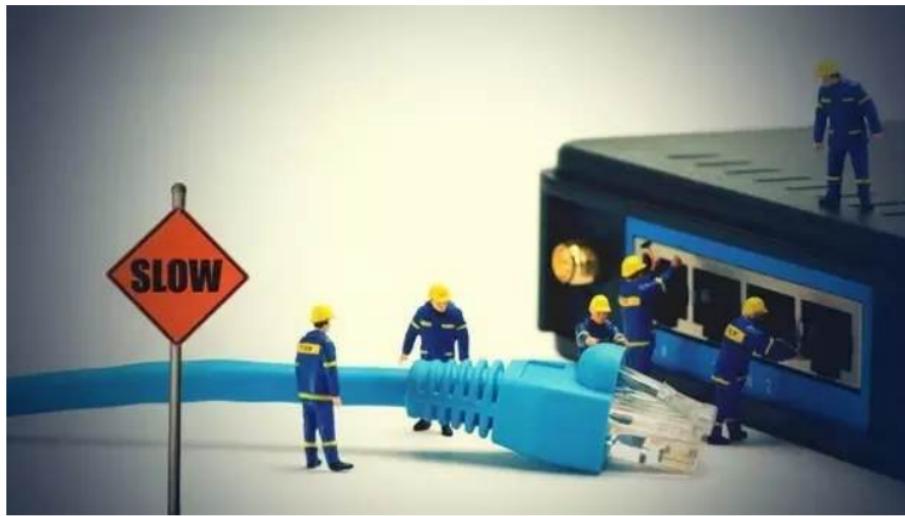
Fixed Broadband

Rank	Download	Upload	Latency
1	241.10 Mbps	232.20 Mbps	13 ms



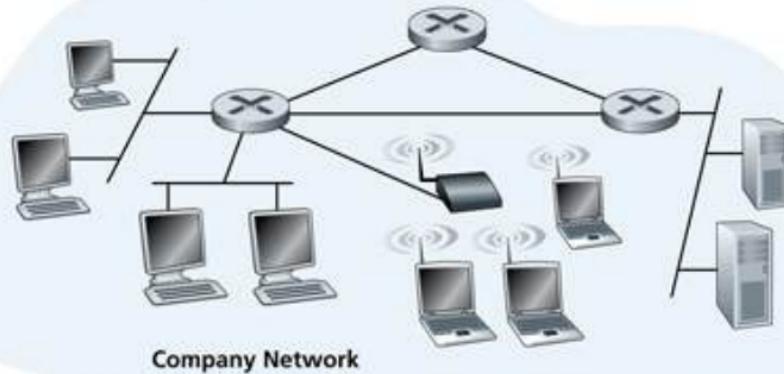


Why my network is slow?



It is more likely that there are hardware/software issues beyond your local connection.

Why my network is slow?

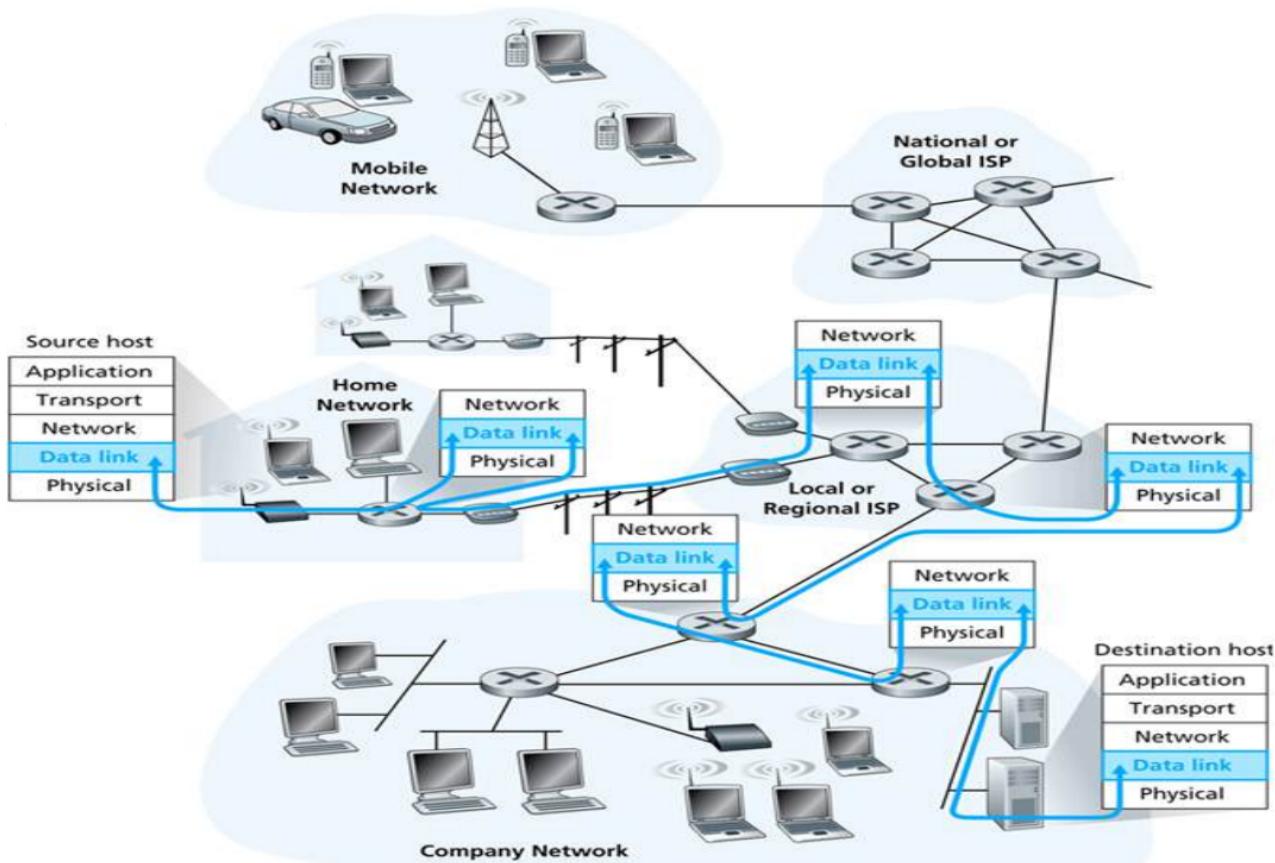


NTU

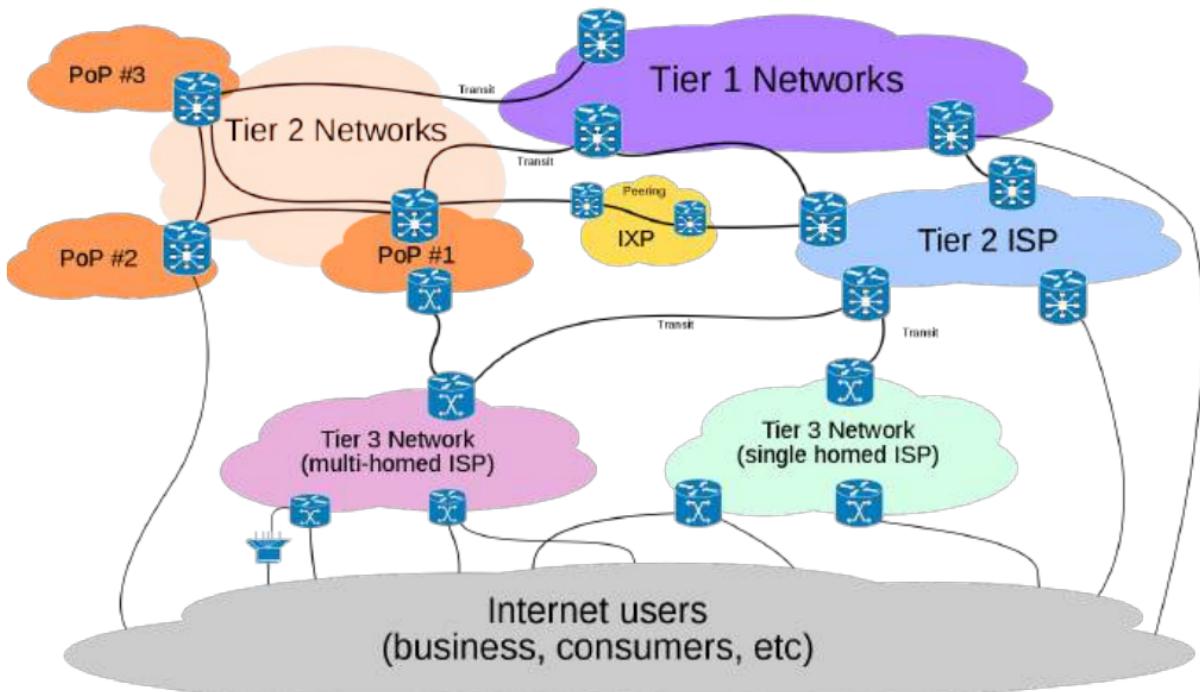


SingTel

Now, let's take a look at the Internet:



Now, let's take a look at the **Internet**:

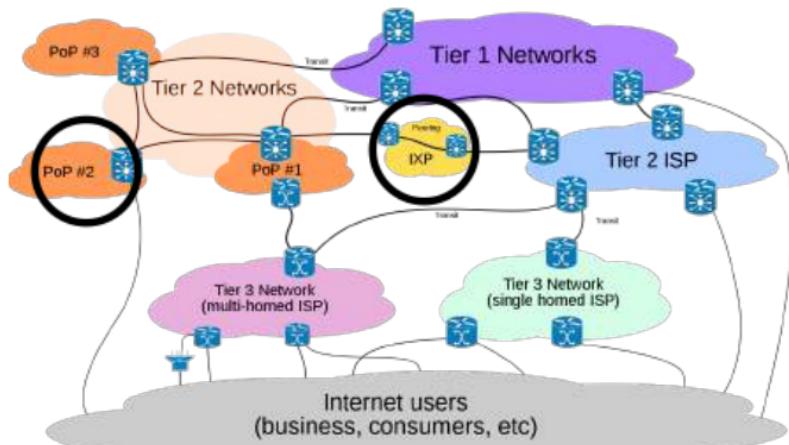




Now, let's take a look at the **Internet**:

Typically, ISPs interconnect at public locations called **Internet eXchange Points (IXPs)** or **Network Access Points (NAPs)**.

Point of Presence (POP) is the location where the ISP houses its network hardware (mostly routers) for subscribers to connect.

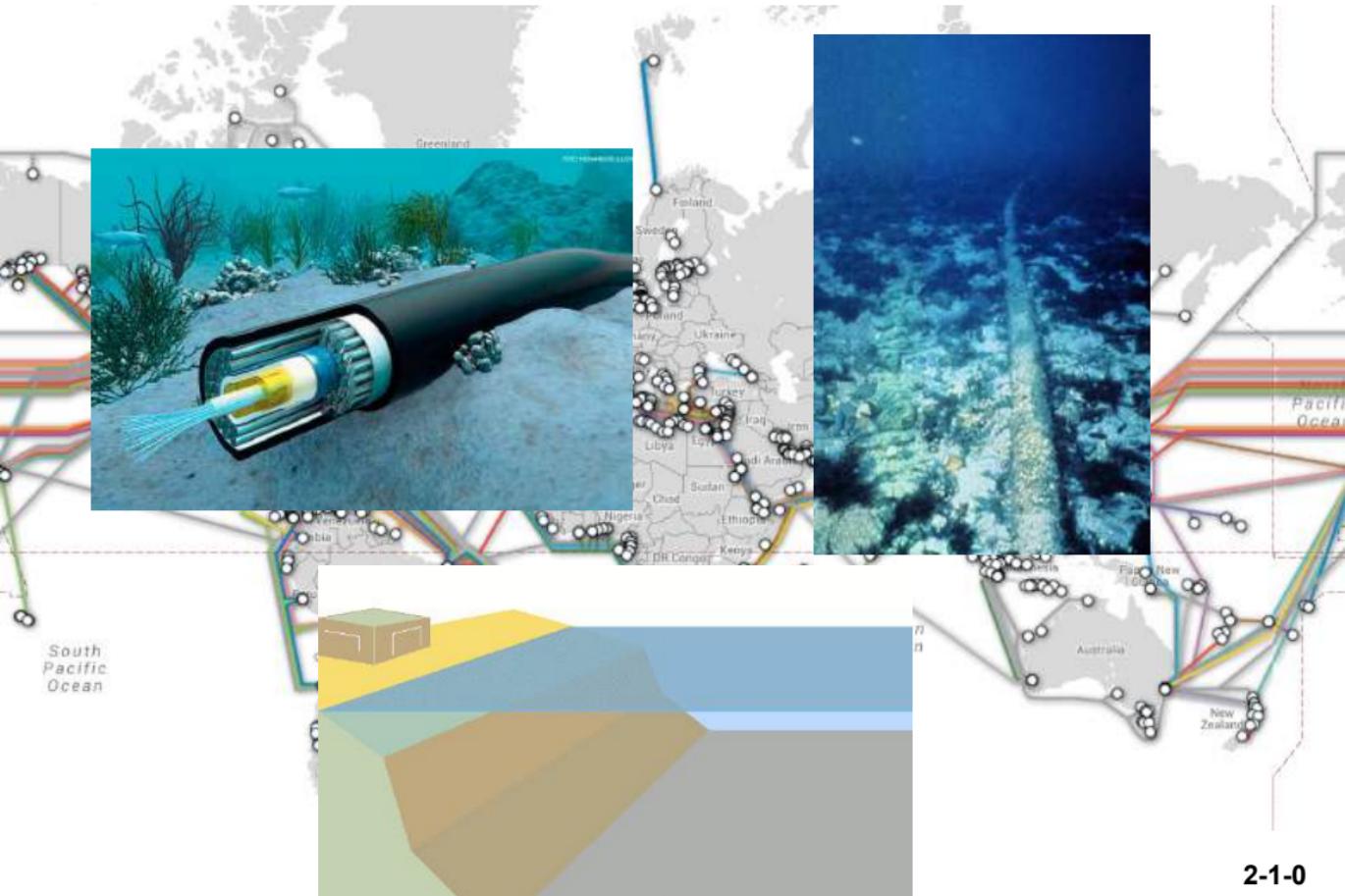




Now, let's take a look at the **Internet**:

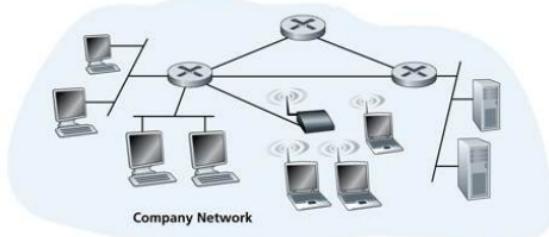
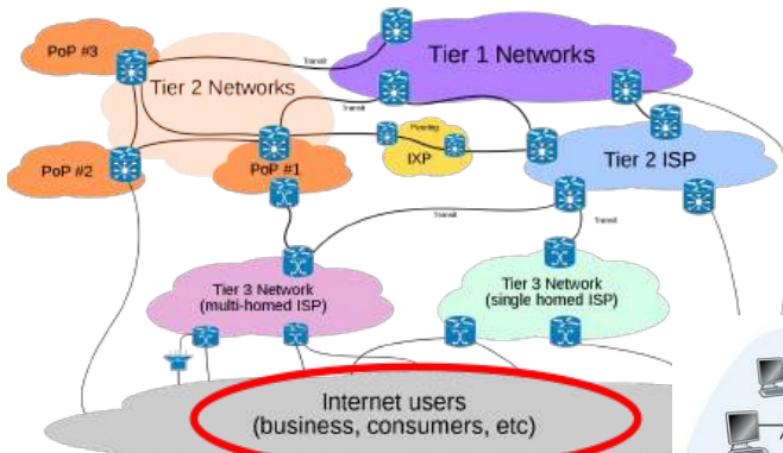


Now, let's take a look at the **Internet**:



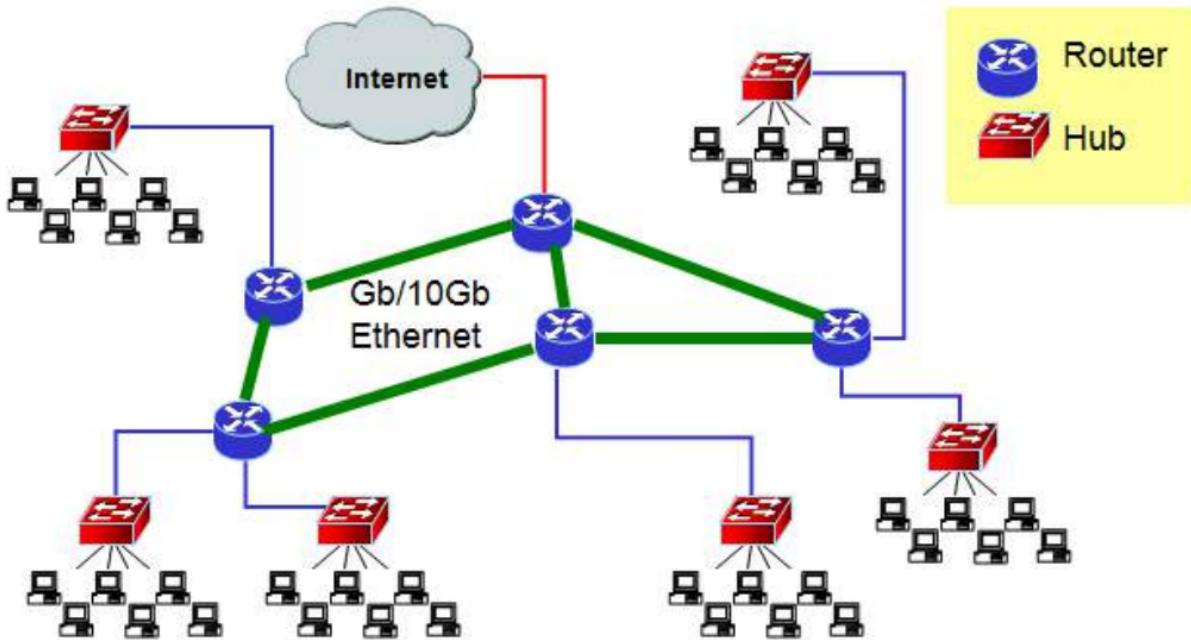


Now, let's take a look at the **Internet**:



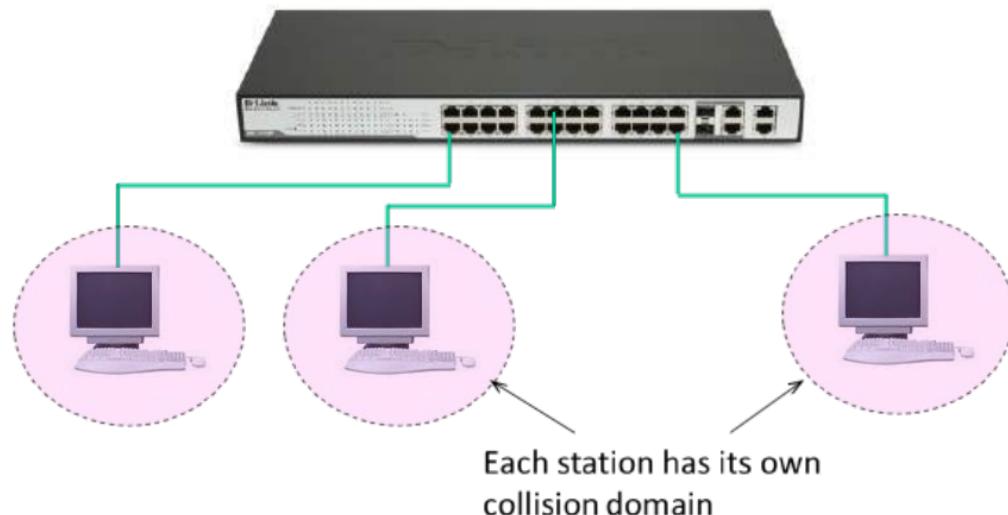
NTU

Nowadays, it's getting common to see **Gigabit/10 Gigabit** high speed **Ethernets** being used at the core layer to interconnect LANs.

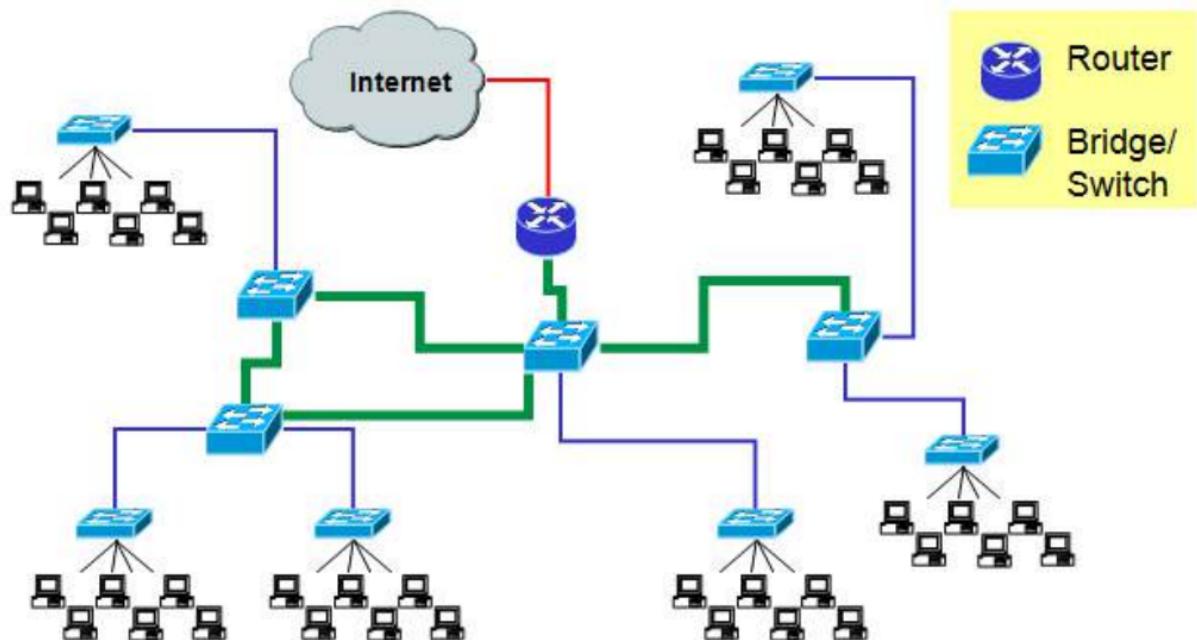


To improve performance, each station can even be connected directly to the **switch** at the access layer.

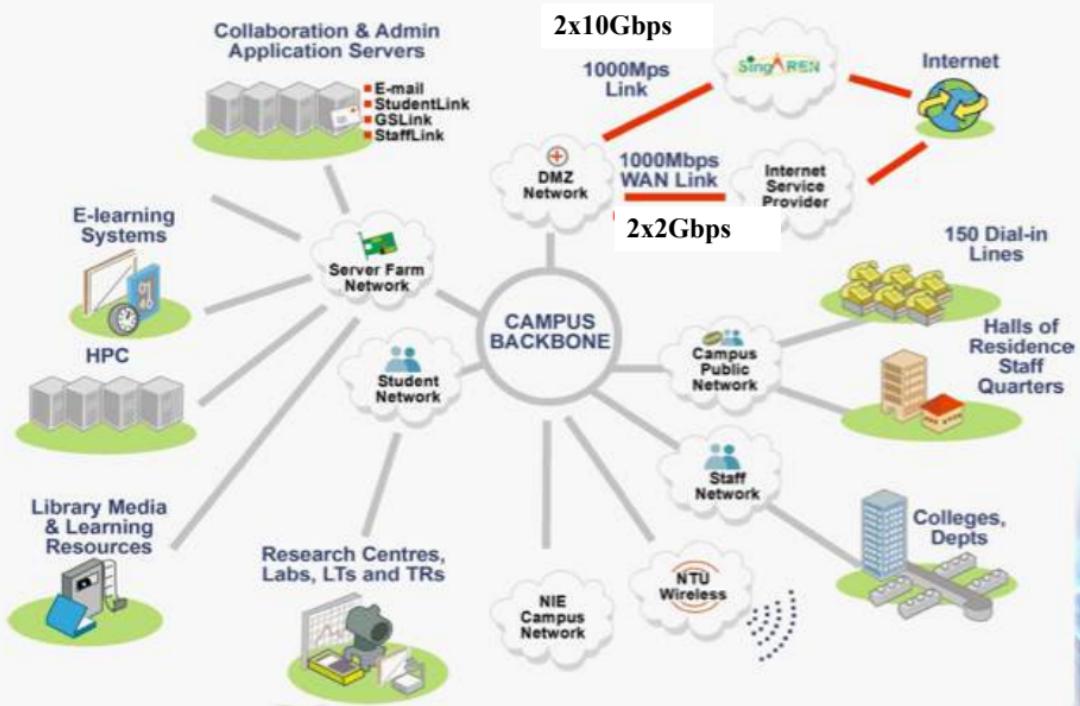
In this case, there is effectively only 1 station per collision domain, which means **collision is non-existence**. CSMA/CD, although still implemented, is now considered not used.



In addition, it's getting common to see a **fully-switched network** consisting of 10/100/Gb/10Gb Ethernets.

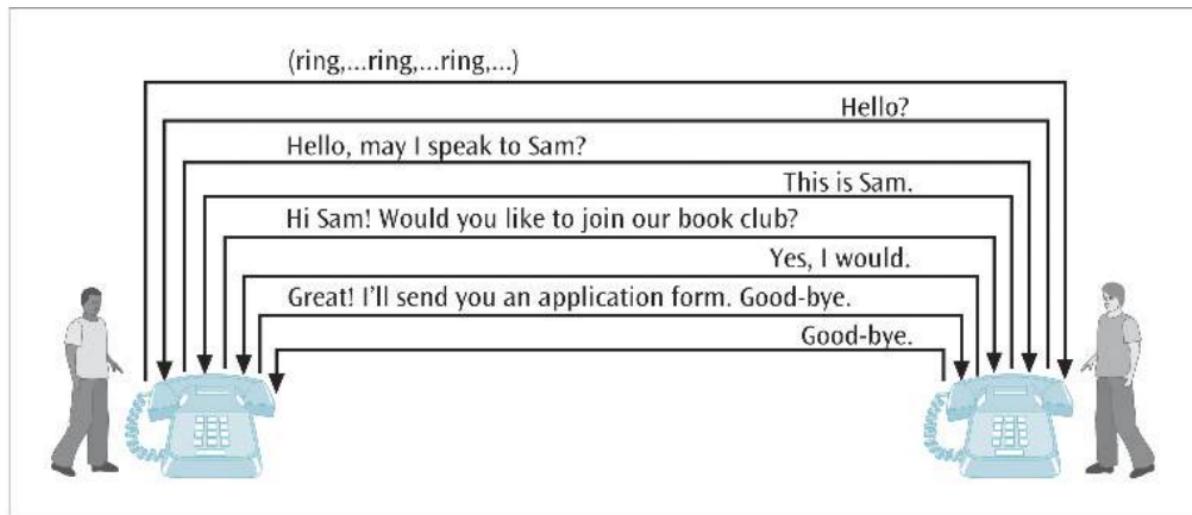


NTU IT Infrastructure

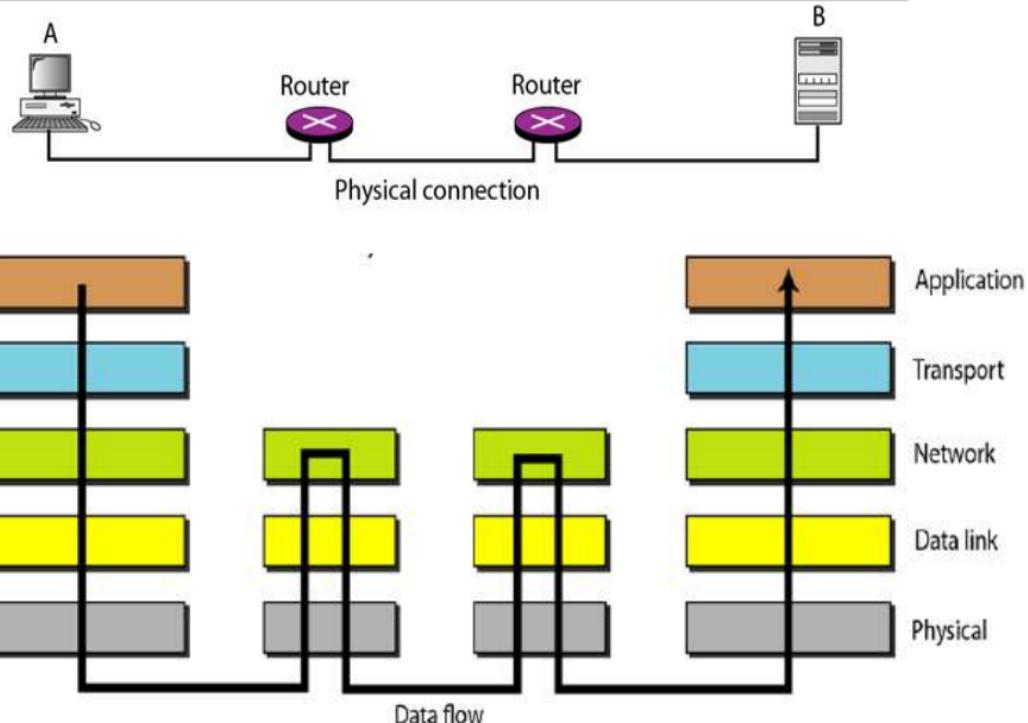


You've learnt the concept of a **communication protocol**, which is a set of **rules** defining the **format** and the **order** of messages exchanged between two parties.

When making a phone call, the recipient waits for the ring tone before answering. Then both will take turns to talk. This is a communication protocol.

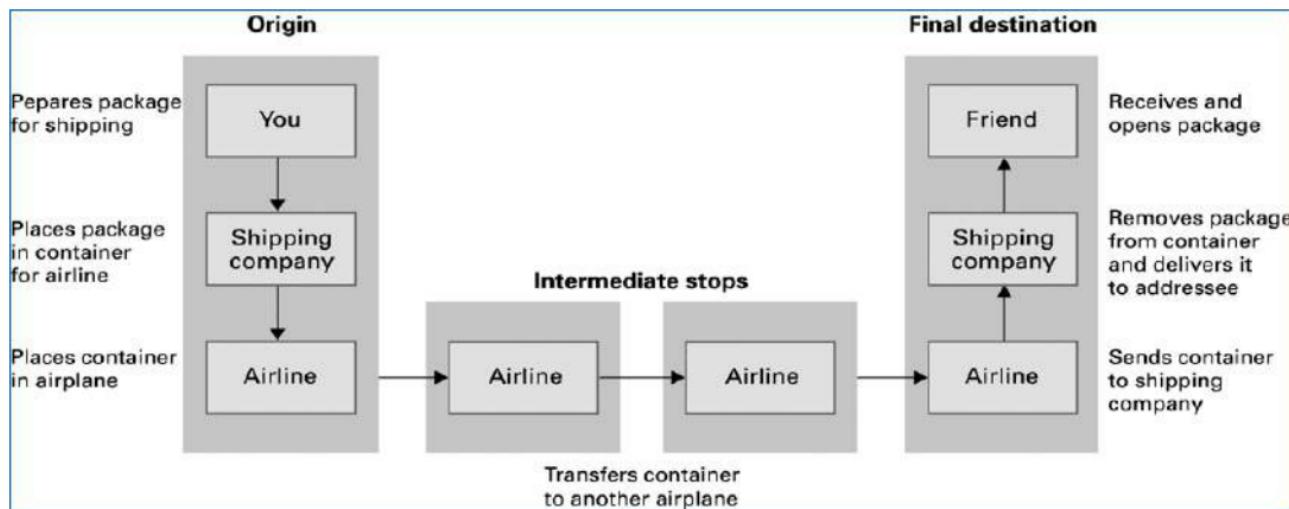


The complex task of **computer networking** is subdivided into **layers** called **protocol stack**. OSI is the standard but TCP/IP is in use, i.e., default standard.



The **idea of layering** is to make a complex task manageable - each layer performs a simpler subtask and uses the services provided by lower layers.

As an analogy, to send a package to an oversea friend, we only need to take care of packing/unpacking, and use the service of a shipping company, which in turn uses the service of an airline.



ENCAPSULATION occurs as information is passed from one layer to the next

The functionalities of the different layers.

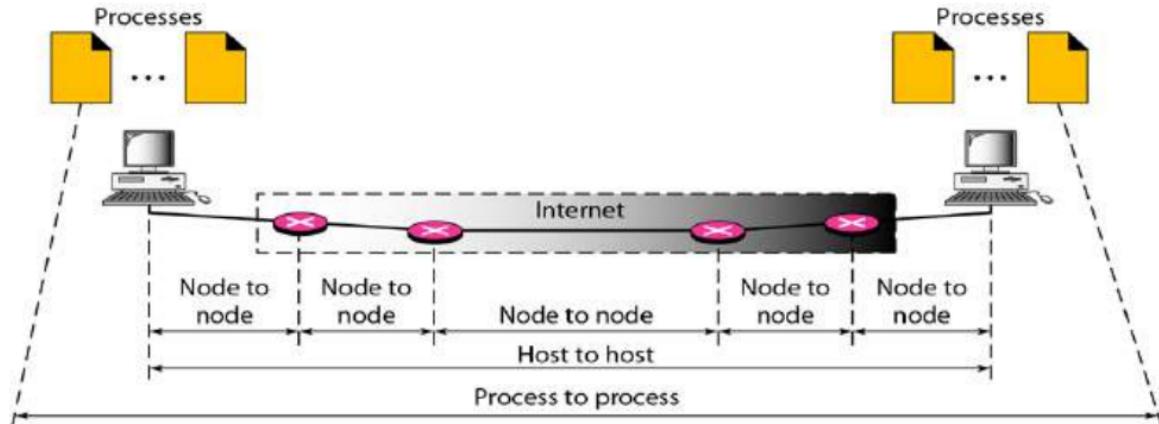
Application layer: concern with application requirement and simply use below services for communications

Transport layer: process-to-process communications

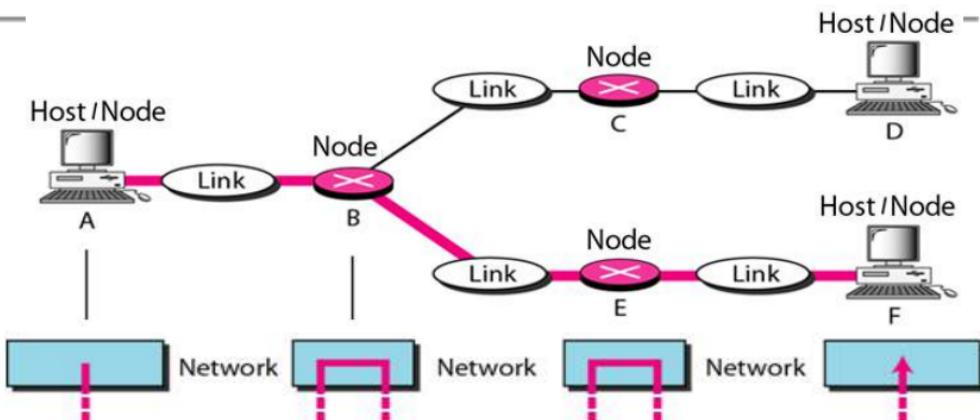
Network layer: host-to-host communications

Data link layer: node-to-node communications

Physical layer: actual transmissions



The difference between host-to-host and node-to-node communications



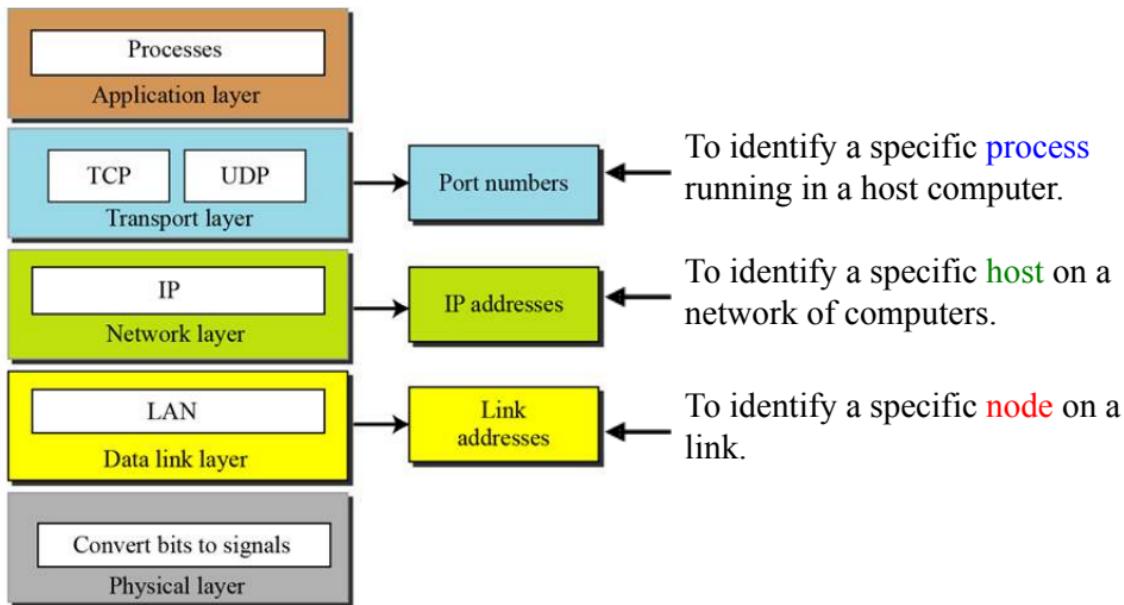
From host A to F, network layer at each node decides the next link/node to go in order to reach F; e.g. at node B, next go to E.



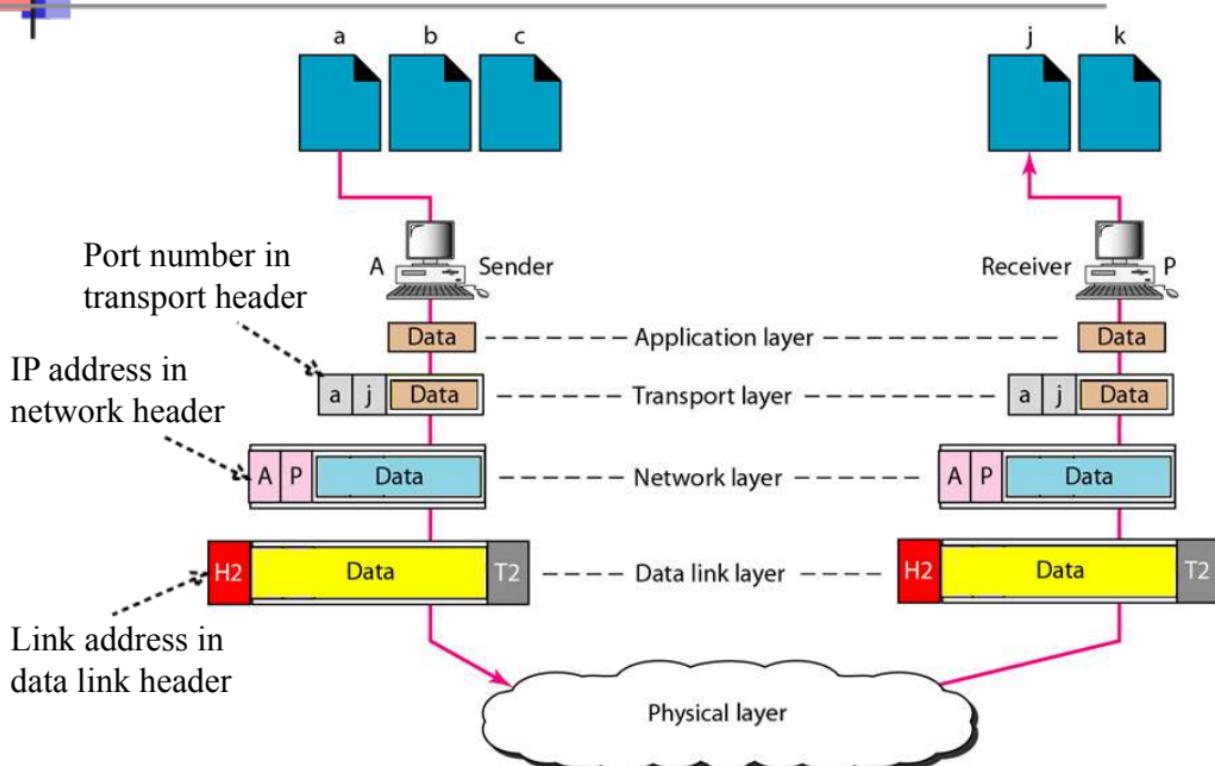
Then, data link layer at each node takes care of data transmission on individual link; e.g. from node B to E.

Since there can be **many processes** in a host, **many hosts** on a network, and **many nodes** on a link, we need **addresses** at different layers.

Hence, **port numbers**, **IP addresses** and **link (MAC) addresses** are introduced to identify specific process, computer and node respectively:



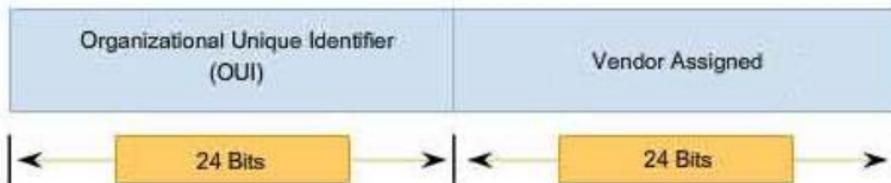
Now, each layer can **encapsulate data** from higher layer by adding its **header** with **addresses**, and the complete data is sent at the physical layer



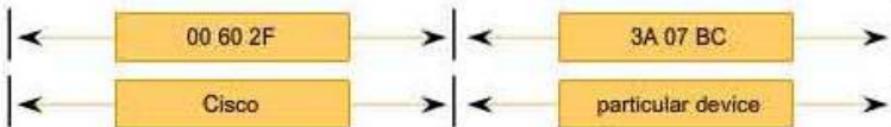
Consider Ethernet which is commonly used at the **data link layer**, the **Ethernet/link/MAC addresses** are assigned by vendors, and are **48 bits** long.

The 48-bit Ethernet address is further divided into 2 fields:

- first 24-bit OUI which is assigned by IEEE, and
- remaining 24-bit which is assigned by the vendor



Ethernet address is commonly written in hexadecimal, e.g.
00:60:2F:3A:07:BC or 00-60-2F-3A-07-BC



Note that the special address FF:FF:FF:FF:FF:FF is used as the **broadcast address** in Ethernet.

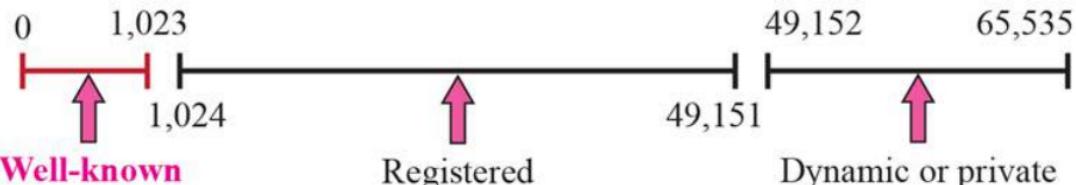
At the **network layer**, IP addresses are coordinated by **IANA/ICANN**, and distributed by five Regional Internet Registries (**RIRs**).

ISPs and large organizations/companies may join RIRs as members and obtain **a block of IP addresses** from them.



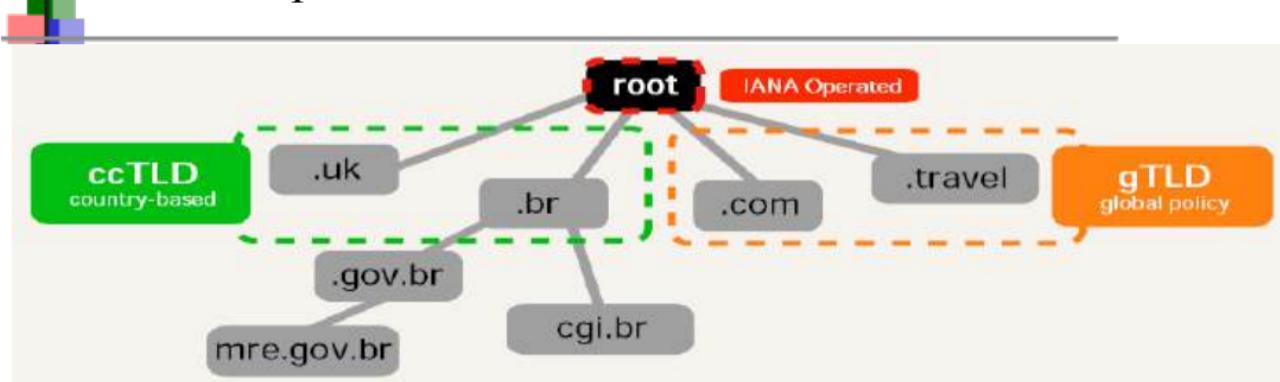
There are 2 types of IP addresses – IPv4 addresses are **32 bits** and IPv6 are 128 bits long!

At the **transport layer**, port numbers are 16 bits long, and reserved ports from 0-49,151 are coordinated by **IANA/ICANN** to prevent conflicts in use.



- **Well-known port:** for common services such as HTTP (port 80), etc; must be registered with IANA
- **Registered port:** for vendor proprietary services such as Cisco P2P Distribution Protocol (port 4051), etc; must also be registered with IANA
- **Dynamic/private/ephemeral port:** typically for OS to allocate temporarily to client processes when needed

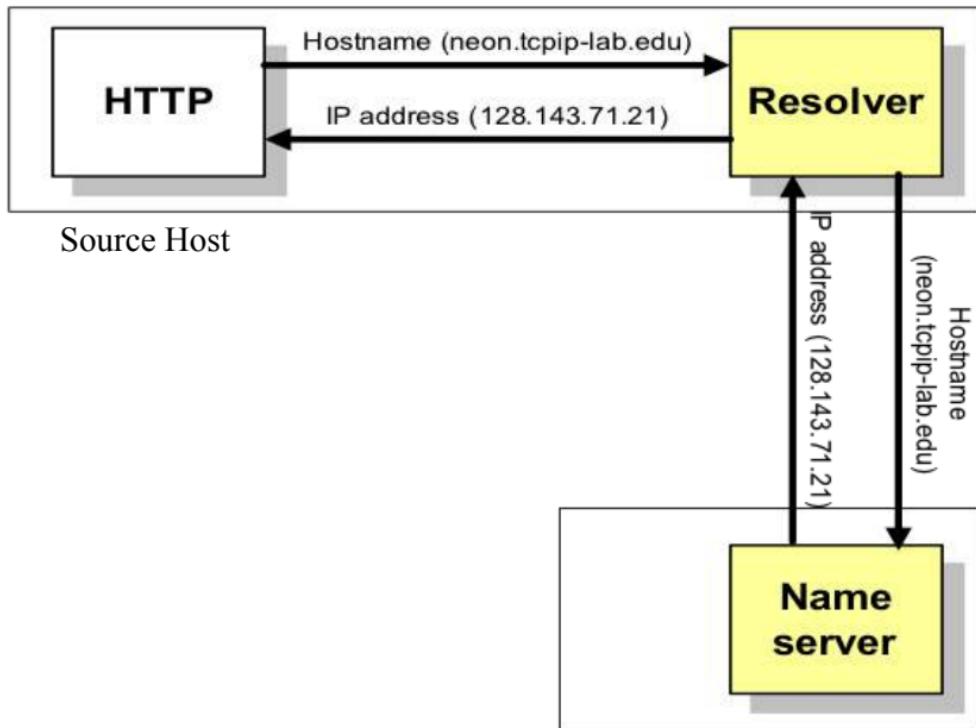
However, it's easier to remember names instead of numbers, so **domain names** are created and users may optionally buy them to map to their IP addresses.



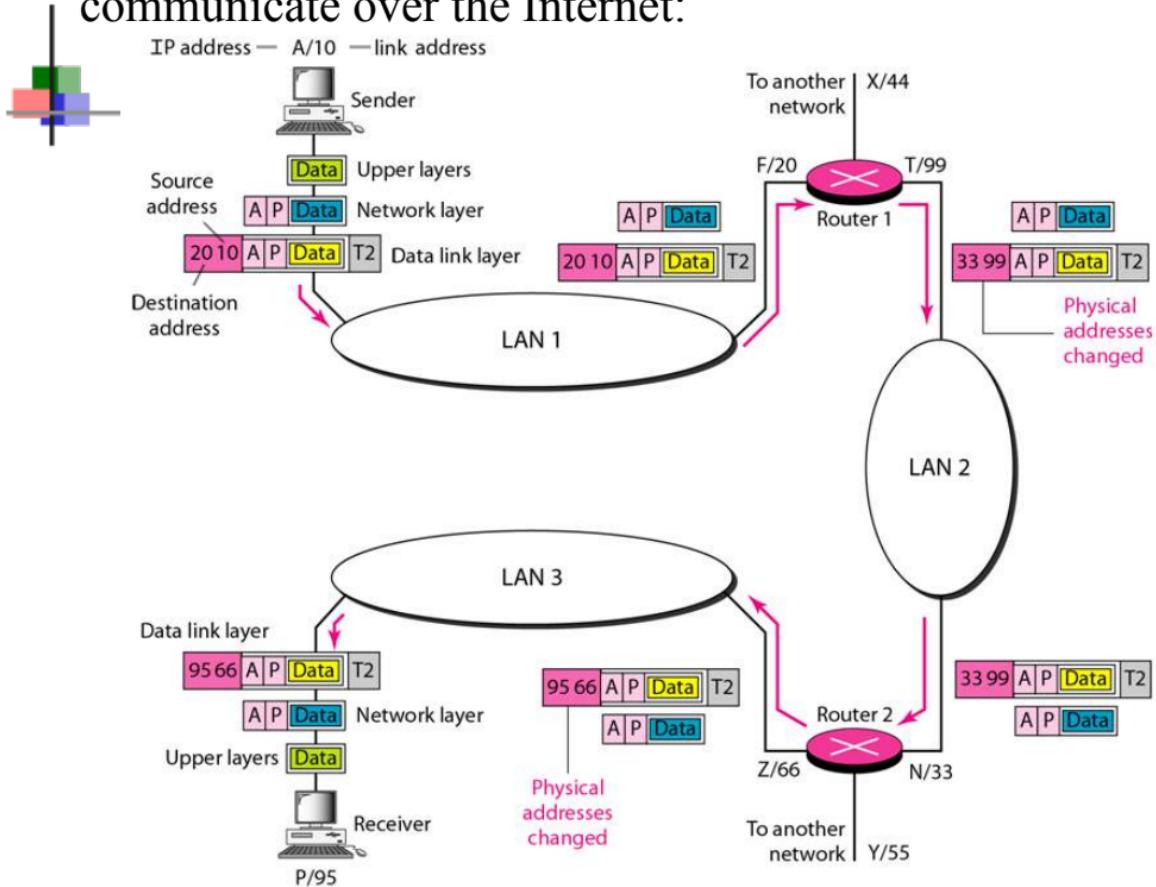
Domain names are divided into **gTLDs** and **ccTLDs**, and commercial **domain name registrars** are accredited to sell them:

- generic Top-Level Domains (**gTLDs**): only **IANA/ICANN-accredited registrars** are able to sell domain names under gTLDs
- country-code Top-Level Domains (**ccTLDs**): delegated to respective countries, e.g. only (Singapore) **SGNIC-accredited registrars** can sell domain names under **.sg**

First, if domain name is used, the **domain name** will need to be **resolved** into corresponding **IP address** by using **Domain Name System (DNS)**.



Finally, here's an overall picture of how computers communicate over the Internet:





Useful links

- <http://www.speedtest.com.sg/speedtest.php>
- http://www.tracert.org/bandwidth_meter/
- <http://www.caida.org/home/>
- <http://www.apnic.net/>
- <http://en.dnstools.ch/> (good tool)
- <https://www.ultratools.com/home>
- <http://ipinfo.io/> (Provide ASN # given IP address)



NANYANG
TECHNOLOGICAL
UNIVERSITY

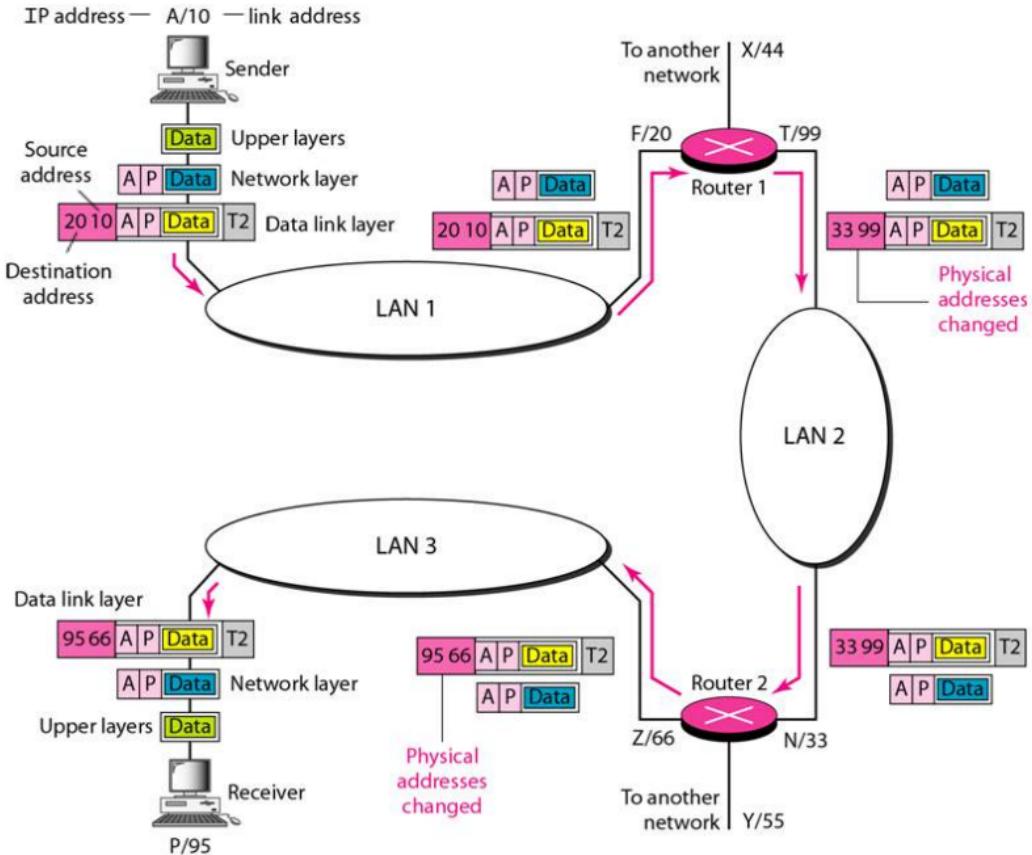
CE3005: Computer Networks
CZ3006: Netcentric Computing

Network Layer - Internet Protocol (IP)

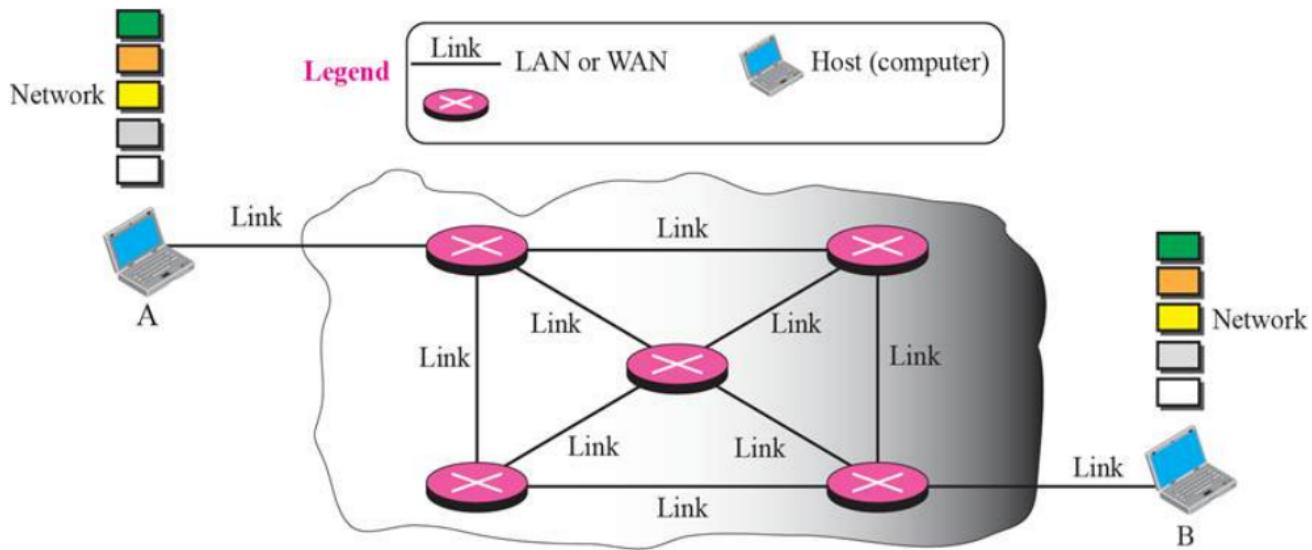
Prof.Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

School of Computer Science and Engineering

How computers communicate over the Internet:

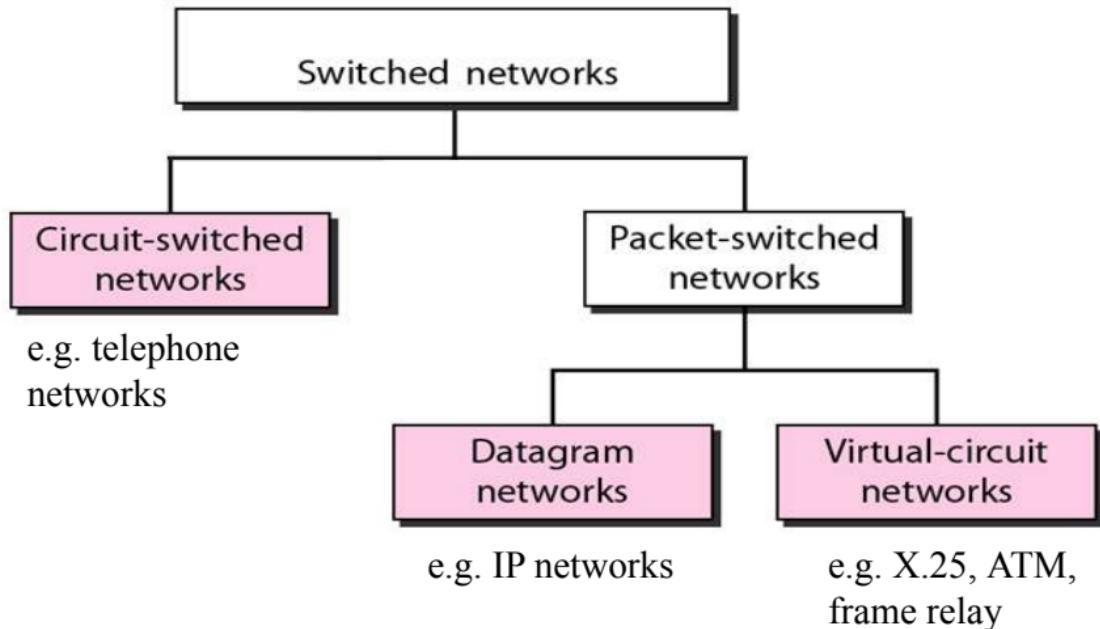


Network Layer



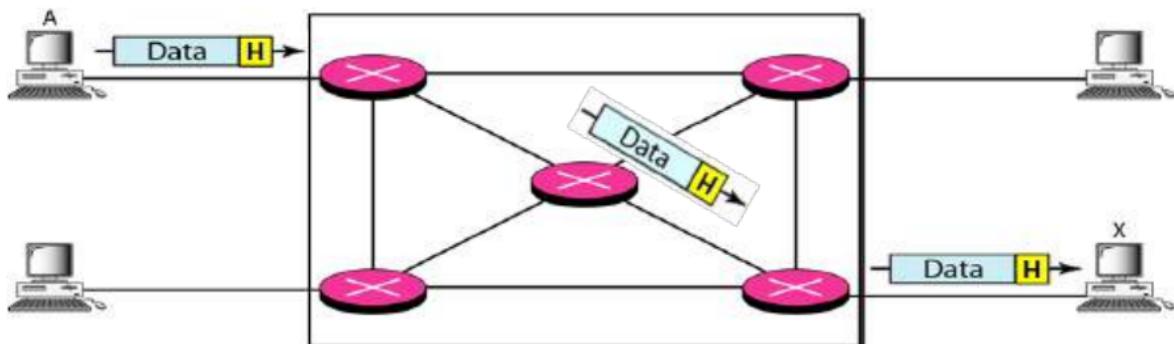


Network Taxonomy



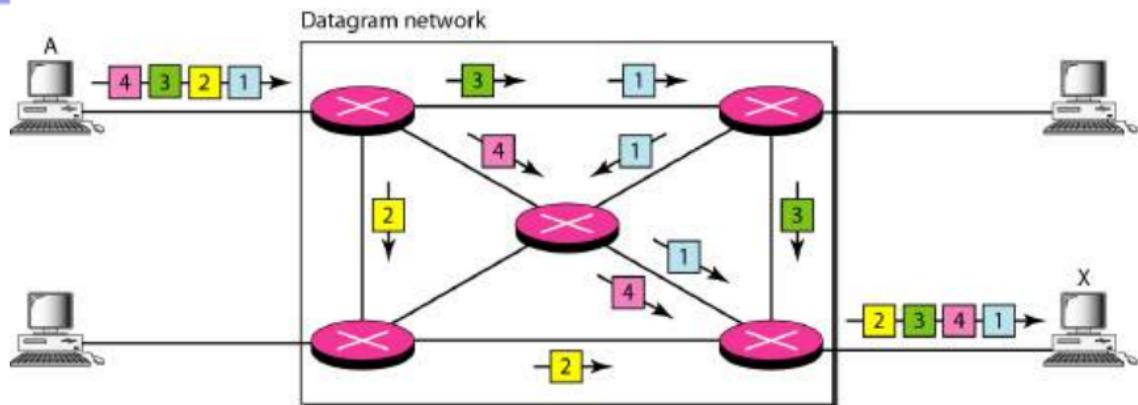
In **packet-switched datagram** networks, transmission begins without establishing a communication path – **Addressing** is needed in each packet.

For example, to transmit data from A to X:



1. A can transmit immediately;
2. However, A needs to append a **header H** containing the destination **address** so that intermediate packet-switched nodes know how to send it to X;
3. At the same time, packet-switched **nodes** are **shared** with others for transmissions (**efficient use of resources**).

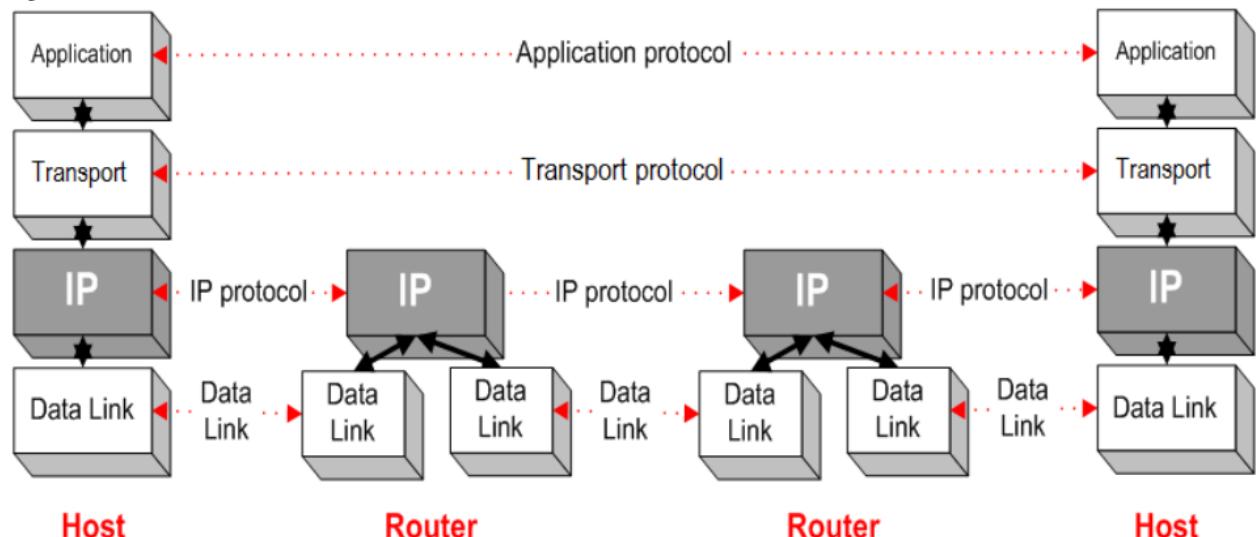
In addition, data is divided into **suitable size packets** depending on **MTU** (Maximum Transfer Unit) of individual network - **Fragmentation**.



Note that since there is no reservation of communication path, **different packets** may take **different path** depending on the load of packet switches.

As a result, packets may arrive out of sequence, or even corrupted and lost! (It will be up to upper layer to handle this, e.g. TCP at transport layer – to be discussed in later module.)

Internet Protocol (IP)



A packet will traverse through intermediate routers **hop-by-hop** from source to destination.

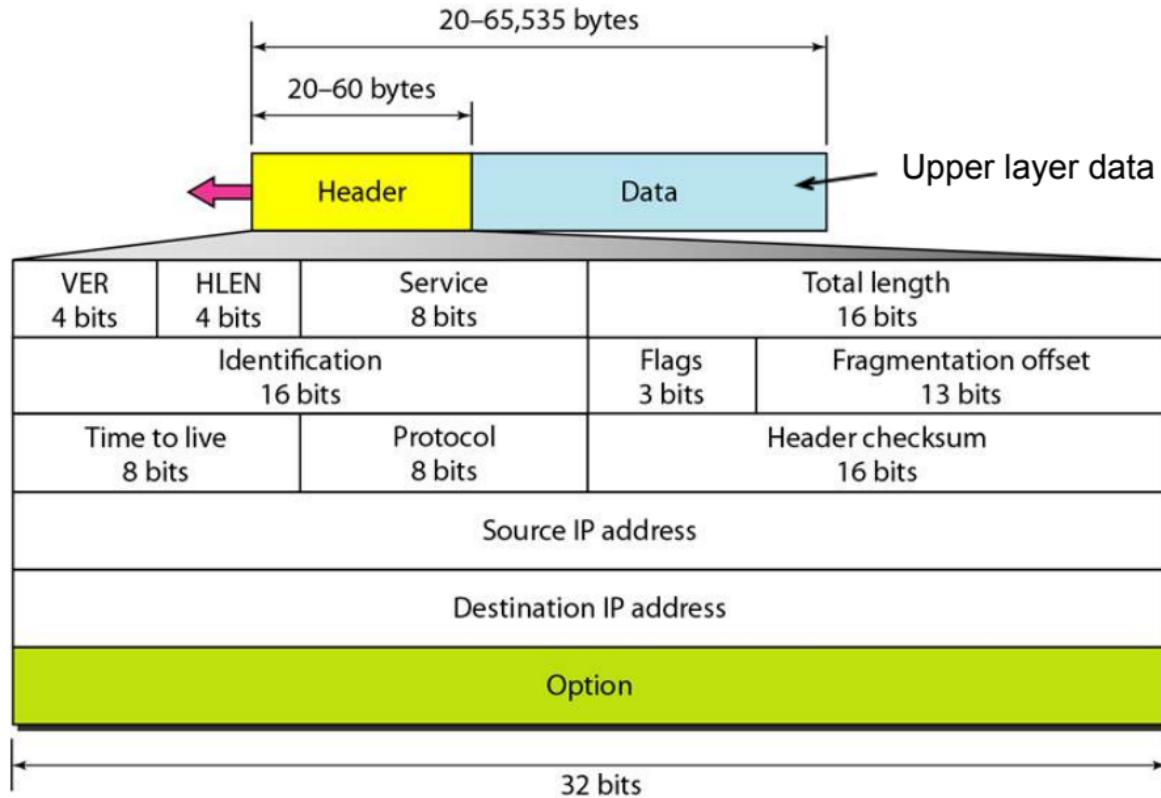


Internet Protocol

Some characteristics:

- It has two basic protocol functions:
 - Addressing
 - Fragmentation
- It provides a connectionless unreliable best-effort (datagram) service:
 - Connectionless: each packet is handled independently, no flow control
 - Unreliable: no error control
 - Best-effort: no throughput guarantee, no delay guarantee, no Quality of Service (QoS) guarantee

The **IP header**, specifically **IPv4** (version 4), is designed as follows: (RFC 791)



IPv4 Header

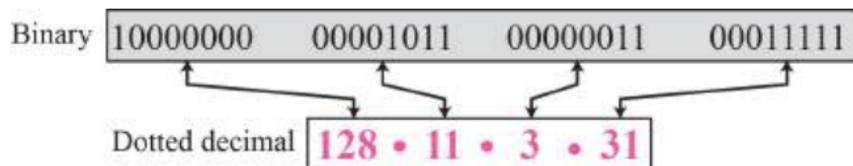
- **Version (VER):** Version number of IP. Current widely-used version is 4.
- **Internet Header Length (HLEN):** Length of header (in multiples of 4 bytes). Typically 5, representing header length of $5 \times 4 = 20$ bytes.
- **Type of Service:** ignore
- **Total Length:** Length of datagram including header (in bytes). Max. is 65,535.
 - Max typical size - 1,500
- **Option Fields (variable length):** ignore

IPv4 Header

- **Protocol:** Indicates the protocol that IP is carrying. (0116 for ICMP, 0616 for TCP, 1116 for UDP)
- **Header Checksum:** For verifying the header is free from error.
- **Source and Destination IP Address:** Indicates the IP addresses of source and destination.

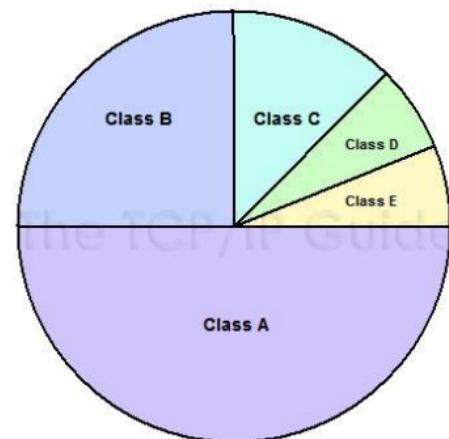
NOTE: Remaining fields will be discussed in relevant slides later.

IP address notation:

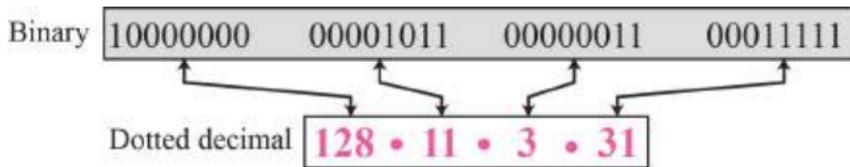


Class

	0	8	16	24	32
A	0	Network ID (bits 2 to 8)		Host ID (24 bits)	
B	1 0	Network ID (bits 3 to 16)		Host ID (16 bits)	
C	1 1 0	Network ID (bits 4 to 24)		Host ID (8 bits)	
D	1 1 1 0	Multicast Group Address (28 bits)			
E	1 1 1 1	Experimental Address ID (bits 5 to 32)			



IP address notation:



Class

	0	8	16	24	32	
A	0	Network ID (bits 2 to 8)		Host ID (24 bits)		1.0.0.0 to 126.255.255.255
B	1 0	Network ID (bits 3 to 16)		Host ID (16 bits)		128.0.0.0 to 191.255.255.255
C	1 1 0	Network ID (bits 4 to 24)		Host ID (8 bits)		192.0.0.0 to 223.255.255.255

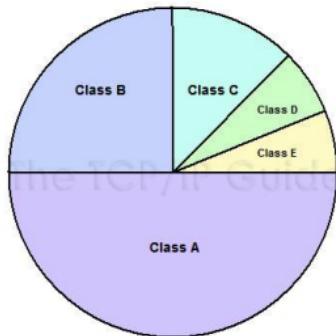
Historical classful network architecture

Class	Leading bits	Size of network number bit field	Size of rest bit field	Number of networks	Addresses per network	Start address	End address
A	0	8	24	128 (2^7)	16,777,216 (2^{24})	0.0.0.0	127.255.255.255
B	10	16	16	16,384 (2^{14})	65,536 (2^{16})	128.0.0.0	191.255.255.255
C	110	24	8	2,097,152 (2^{21})	256 (2^8)	192.0.0.0	223.255.255.255

Number Resources

Overview

Abuse Issues

[Overview](#)[Questions and Answers](#)

Number Resources

We are responsible for global coordination of the Internet Protocol addressing systems, as well as the Autonomous System Numbers used for routing Internet traffic.

Currently there are two types of Internet Protocol (IP) addresses in active use: IP version 4 (IPv4) and IP version 6 (IPv6). IPv4 was initially deployed on 1 January 1983 and is still the most commonly used version. IPv4 addresses are 32-bit numbers often expressed as 4 octets in "dotted decimal" notation (for example, 192.0.2.53). Deployment of the IPv6 protocol began in 1999. IPv6 addresses are 128-bit numbers and are conventionally expressed using hexadecimal strings (for example, 2001:0db8:582:ae33::29).

Both IPv4 and IPv6 addresses are generally assigned in a hierarchical manner. Users are assigned IP addresses by Internet service providers (ISPs). ISPs obtain allocations of IP addresses from a local Internet registry (LIR) or National Internet Registry (NIR), or from their appropriate Regional Internet Registry (RIR):



REGISTRY	AREA COVERED
AFRINIC	Africa Region
APNIC	Asia/Pacific Region
ARIN	Canada, USA, and some Caribbean Islands
LACNIC	Latin America and some Caribbean Islands
RIPE NCC	Europe, the Middle East, and Central Asia

Block	Organization	IANA date	RIR date
4.0.0.0/8	CenturyLink, Inc.	1992-12	1992-12-01
12.0.0.0/8	AT&T Services	1995-06	1983-08-23
17.0.0.0/8	Apple Inc.	1992-07	1990-04-16
19.0.0.0/8	Ford Motor Company	1995-05	1988-06-15
38.0.0.0/8	PSINet, Inc.	1994-09	1991-04-16
44.0.0.0/8	Amateur Radio Digital Communications	1992-07	1992-07-01
48.0.0.0/8	Prudential Securities Inc.	1995-05	1990-12-07
56.0.0.0/8	US Postal Service	1994-06	1992-11-02
73.0.0.0/8	Comcast Corporation	N/A	2005-04-19

List of assigned /8 blocks to the United States Department of Defense [\[edit\]](#)

Block	Organization	IANA date	RIR date	Notes
6.0.0.0/8	Army Information Systems Center	1994-02	1994-02-01	Headquarters, USAISC.
7.0.0.0/8	DoD Network Information Center	1995-04	1997-11-24	Formerly IANA - Reserved 1995-04. Entirely assigned to DoD Network Information Center (DNIC) 1997-11-24. Updated to Administered by ARIN not before 2007.
11.0.0.0/8	DoD Intel Information Systems	1993-05	1984-01-19	
21.0.0.0/8	DDN-RVN	1991-07	1991-07-01	DoD Network Information Center (DNIC).
22.0.0.0/8	Defense Information Systems Agency	1993-05	1989-06-26	DoD Network Information Center (DNIC).
26.0.0.0/8	Defense Information Systems Agency	1995-05	1995-05-01	DoD Network Information Center (DNIC).
28.0.0.0/8	DSI-North	1992-07		DoD Network Information Center (DNIC).
29.0.0.0/8	Defense Information Systems Agency	1991-07	1991-07-01	DoD Network Information Center (DNIC).
30.0.0.0/8	Defense Information Systems Agency	1991-07	1991-07-01	DoD Network Information Center (DNIC).
33.0.0.0/8	DLA Systems Automation Center	1991-01	1991-01-01	DoD Network Information Center (DNIC).
55.0.0.0/8	DoD Network Information Center	1995-04	1996-10-26	Headquarters, USAISC. Formerly Boeing Computer Services 1995-04. Updated to DoD Network Information Center in 2007-02.
214.0.0.0/8	US-DOD	1998-03	1998-03-27	DoD Network Information Center (DNIC).
215.0.0.0/8	US-DOD	1998-03	1998-06-05	DoD Network Information Center (DNIC).

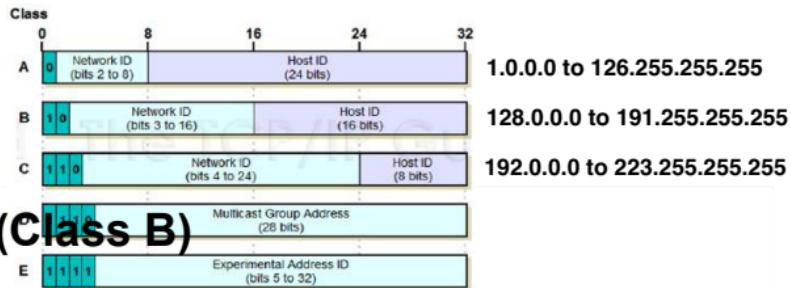
IPv4 Address: Special Use (RFC 5735)

- Network and/or Host id – all ‘0’s: (can only be used as **source address**; e.g. during startup to get own IP in DHCP)
 - 0.0.0.0 means this host on this network.
 - 155.69.0.0 means the network/subnet ID.
- Network and/or Host id – all ‘1’s: (can only be used as **destination address**)
 - 255.255.255.255 limited broadcast within this network (ARP).
 - 155.69.255.255 directed broadcast on 155.69.x.x network.
- Loopback Address (127.x.y.z):
 - Internal loopback to same host. Useful for self-testing of network software. “x.y.z” can be any valid value, eg, 127.0.0.1.

Exercise 1

- What is the Network address (assume classful addressing) of the following IP address

155.69.8.9



194.155.69.0.0 (Class B)

120.194.3.8.0 (Class C)

160.120.0.0.0 (Class A)

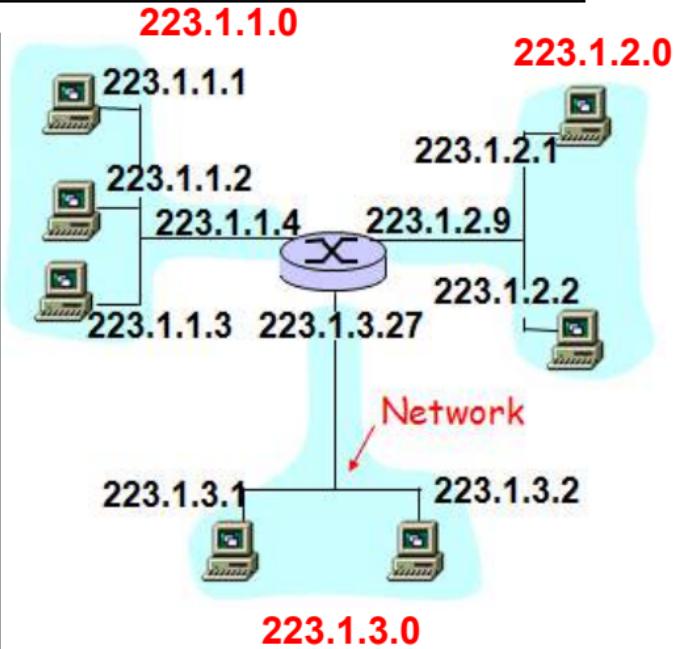
NO CLASS!! You cannot have values larger than 255

Ethernet adapter Ethernet:

```
Connection-specific DNS Suffix . : ntu.edu.sg
Description . . . . . : Realtek PCIe GBE Family Controller
Physical Address. . . . . : F4-4D-30-F9-B8-56
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::8cd8:55f3:71ff:6ad7%7(PREFERRED)
IPv4 Address. . . . . : 155.69.142.98(Preferred)
Subnet Mask . . . . . : 255.255.252.0
Lease Obtained. . . . . : Monday, 7 October 2019 9:41:53 AM
Lease Expires . . . . . : Wednesday, 9 October 2019 9:56:54 AM
Default Gateway . . . . . : 155.69.143.254
DHCP Server . . . . . : 155.69.3.8
DHCPv6 IAID . . . . . : 116673840
DHCPv6 Client DUID. . . . . : 00-01-00-01-23-84-08-53-F4-4D-30-F9-B8-56
DNS Servers . . . . . : 155.69.3.7
                           155.69.3.8
                           155.69.3.9
Primary WINS Server . . . . . : 155.69.5.54
Secondary WINS Server . . . . . : 155.69.4.83
NetBIOS over Tcpip. . . . . : Enabled
```

IPv4 Address

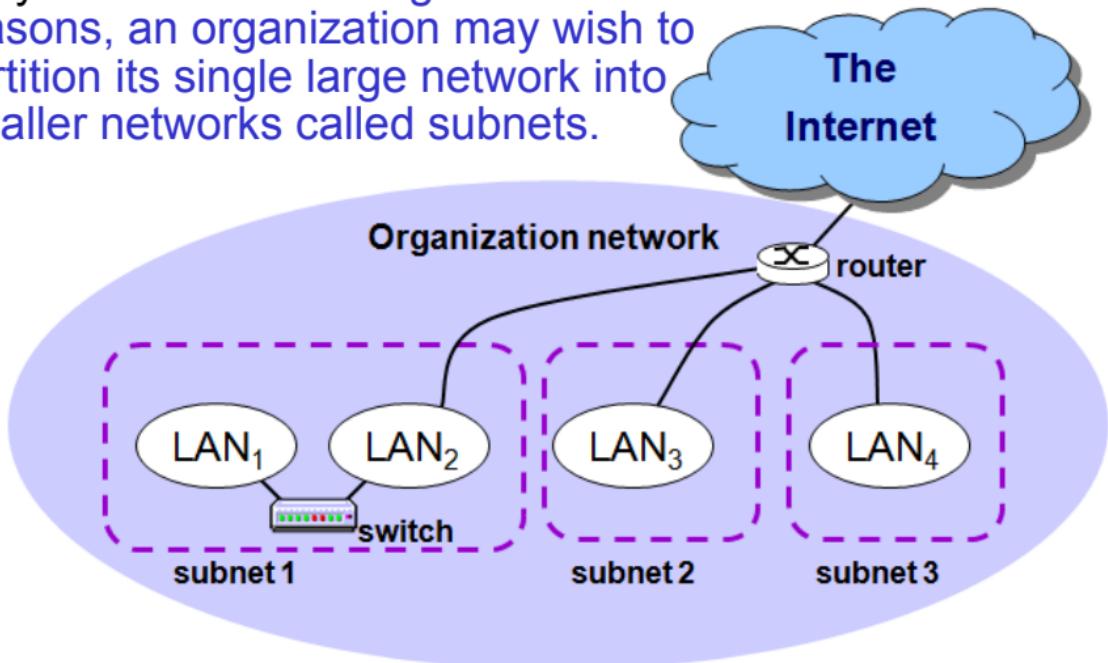
- **IP address:** assigned to host/router's *interface*
- **Interface:** connection between host/router and physical link
- **What's a network?** (from IP address perspective)
 - device interfaces with same network id of IP address
 - can physically reach each other without intermediate router



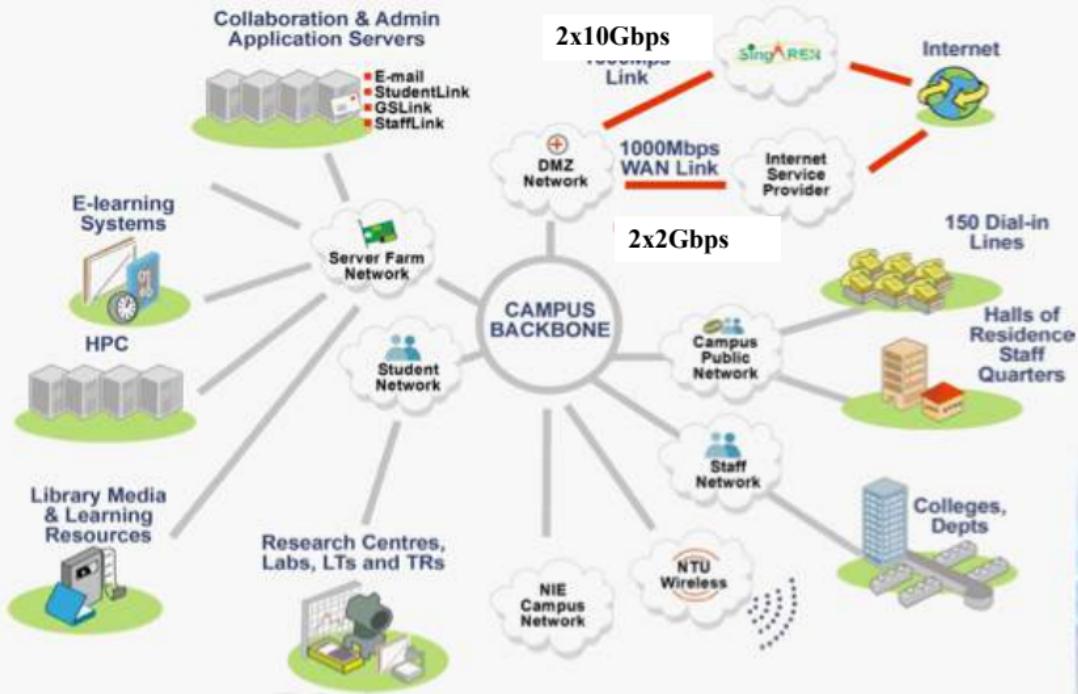
Subnetting

(RFC 950)

- Why? For easier management or other reasons, an organization may wish to partition its single large network into smaller networks called subnets.



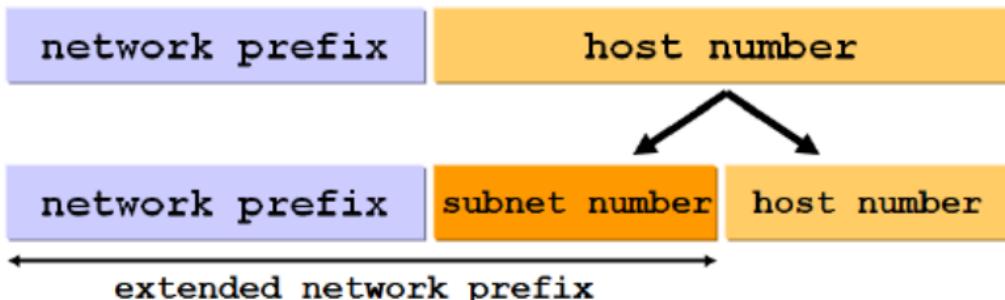
NTU IT Infrastructure



Subnetting

(RFC 950)

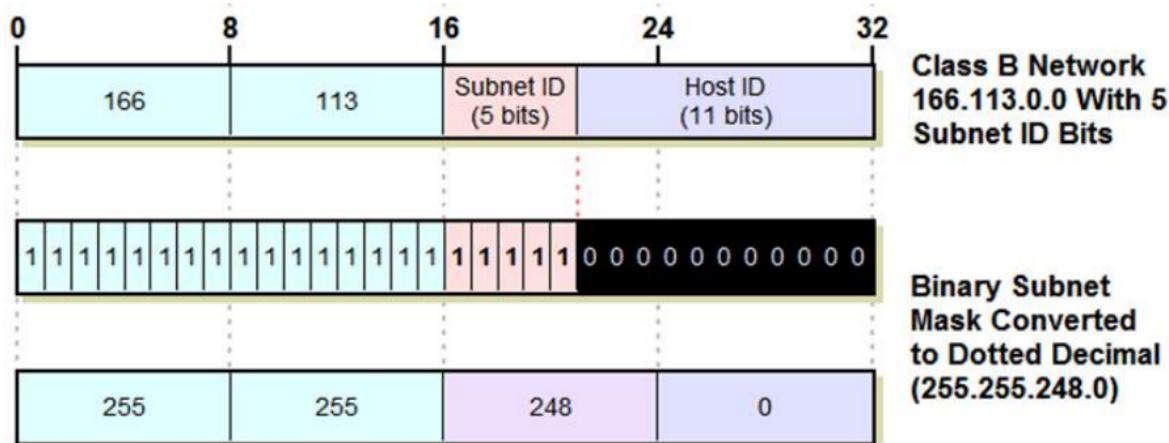
- How? Add another level of hierarchy to the IP address structure.



- Hence, organization is free to decide the number of bits for subnet and host numbers.
- Externally, the organization network is still viewed as a single large network.

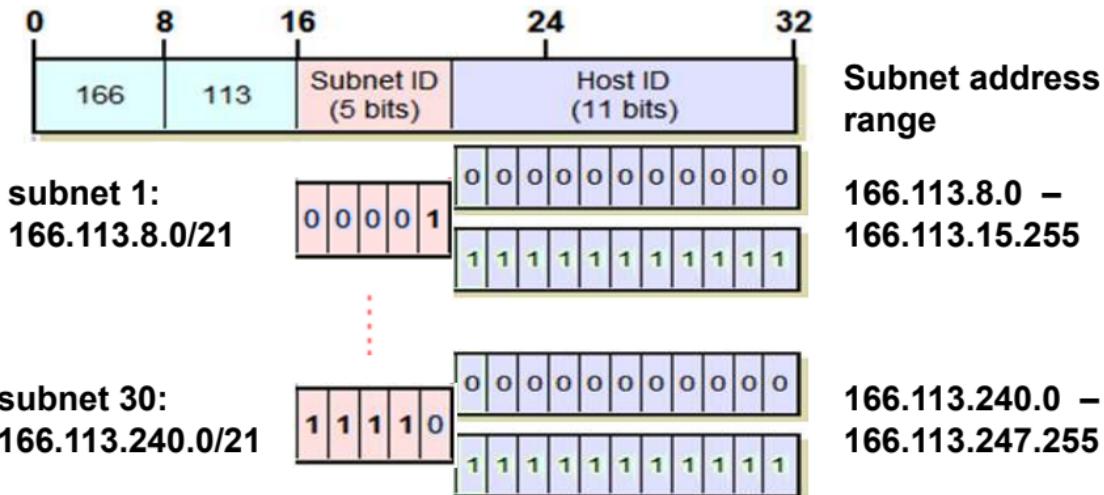
Subnet Masks

- To indicate the length of extended network prefix, use a **subnet mask w.x.y.z** (bits corresponding to extended network prefix are set to '1's, and '0's otherwise.)



Subnet Address Calculation

- 'slash' notation: **a.b.c.d/x** where x indicates # bits for extended network prefix



Maximum # hosts in each subnet = $2^{11} - 2 = 2046$ because

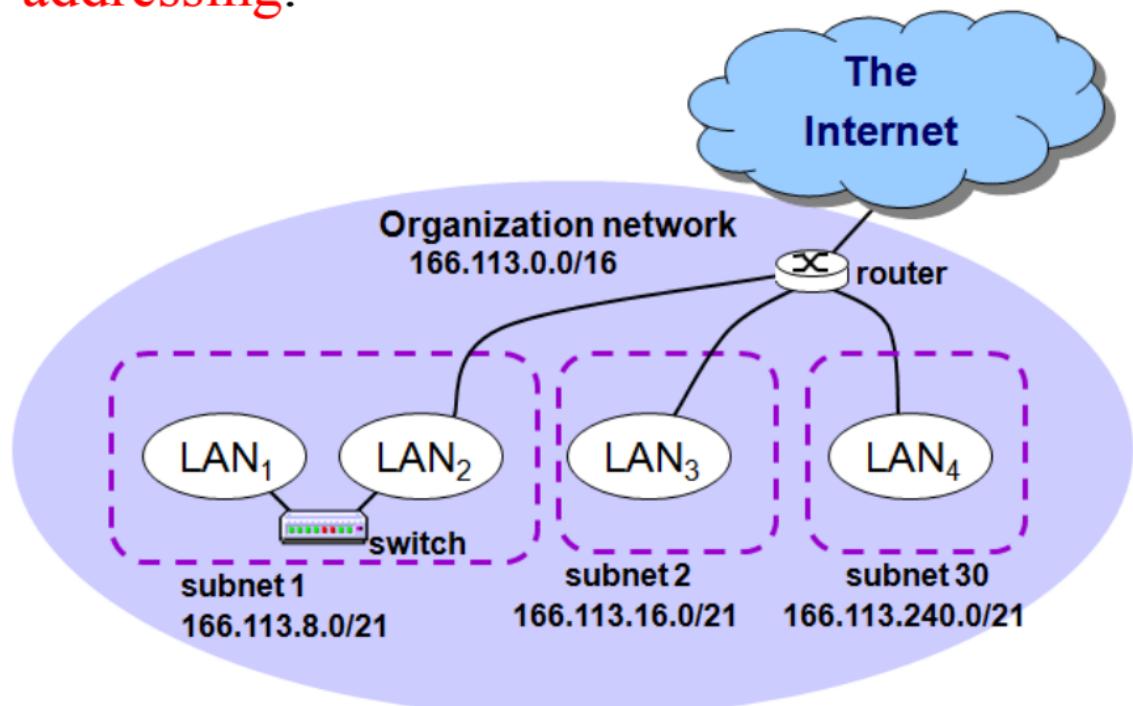
- host ID = all '0's indicates network/subnet ID number
- host ID = all '1's indicates broadcast address

```
Configuration for interface "Ethernet"
  DHCP enabled: Yes
  IP Address: 155.69.142.98
  Subnet Prefix: 155.69.140.0/22 (mask 255.255.252.0)
  Default Gateway: 155.69.143.254
  Gateway Metric: 0
  InterfaceMetric: 25
  DNS servers configured through DHCP: 155.69.3.7
                                         155.69.3.8
                                         155.69.3.9
  Register with which suffix: Primary only
  WINS servers configured through DHCP: 155.69.4.83
                                         155.69.5.54

Configuration for interface "Loopback Pseudo-Interface 1"
  DHCP enabled: No
  IP Address: 127.0.0.1
  Subnet Prefix: 127.0.0.0/8 (mask 255.0.0.0)
  InterfaceMetric: 75
  Statically Configured DNS Servers: None
  Register with which suffix: Primary only
  Statically Configured WINS Servers: None
```

**255.255.11111100.0
155.69.10001110.98**

An example subnetting with classful addressing.

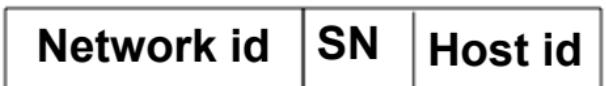


Subnet Broadcasting

- **Subnet broadcast (set host address to all 1s):**
 - Say Class B address: 166.113.0.0/21
 - Subnet mask: 255.255.248.0
 - Then 166.113.15.255 means broadcast to the subnet
166.113.8.0/21 (166.113.00001111.1111111)
- **All subnets broadcast (set subnet & host addresses to all 1s):**
 - **166.113.255.255 means broadcast to all hosts in all subnets**

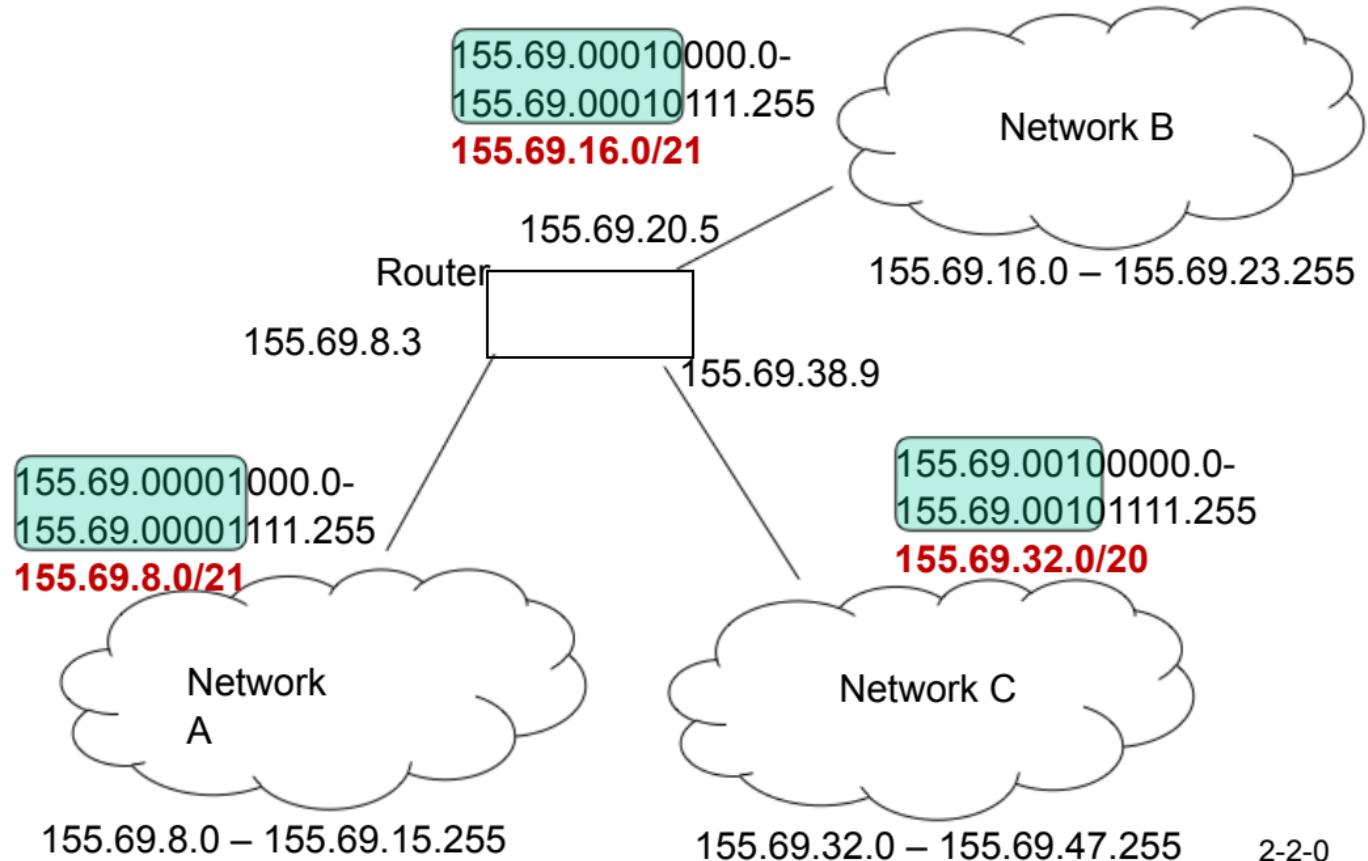
Summary of classful addressing

- IP address – 32 bits, eg. 155.69.8.3
- Classful addressing(value of first 8 bits of the address)
 - A = 1 -126
 - B = 128 – 191
 - C = 192 – 223
- Subnetting – Breaking up the network into smaller network, called subnet. Uses Subnet Mask (eg. 255.255.252.0).



- Subnet mask is usually used during the set-up communication, to determine if the source and destination are in the same subnet.

Example network



IPv4 Address Exhaustion!!!

IP classful addressing problems:

- Inefficient use of address space

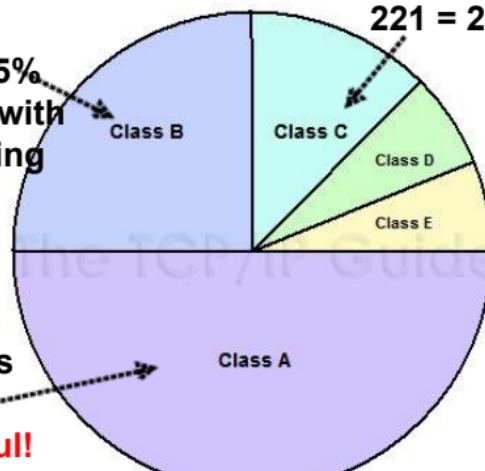
$$2^{14} = 16,384$$

organizations owns 25% of total IP addresses, with each Class B supporting up to 65,534 hosts. Do they need that many?

$$128 - 2 = 126$$
 organizations

owns 50% of total IP

- addresses = very wasteful!



$2^{21} = 2,097,152$ organizations owns 12.5% of total IP addresses. But each Class C supporting only 254 hosts is too small for many.

'today's
global network and future growth.'

IPv4 address exhaustion

- "Exhaustion" is defined here as the time when the pool of available addresses in each RIR reaches the "last /8 threshold" of 16,777,216 addresses.

<http://www.potaroo.net/tools/ipv4/index.html>

RIR	Projected exhaustion date	Remaining addresses in RIR pool (/8)
APNIC	19 April 2011(actual)	0.8394
RIPE NCC	7 Sept 2012(actual)	0.9281
ARIN	19 Feb 2015	0.8340
LAARNIC	10 June 2014(actual)	0.2338
AFRINIC	17 July 2019	3.0749



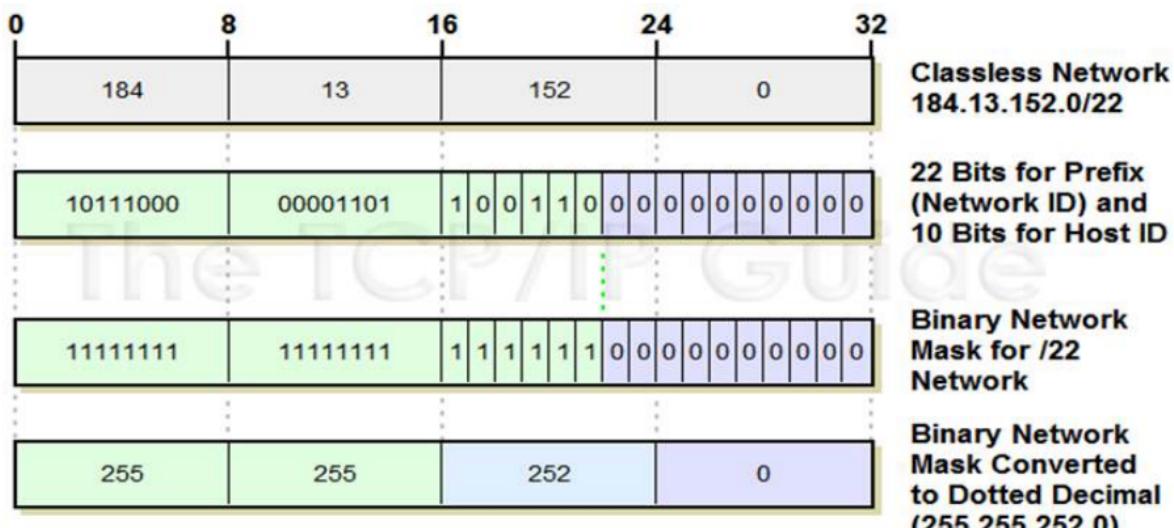
IPv4 Address Exhaustion

- **Solutions to IPv4 Address Exhaustion:**
- **Classless InterDomain Routing (short-term solution)**
 - This reduces the wastage in address allocation.
 - Organizations will be given adequate but not excessive address space.
- **Network Address Translation (NAT) using Private IP Addresses (will ease but not solve the problem)**
 - A single machine (usually a router) with an IP address representing many computers behind it. IP addresses require translation.
- **IP version 6, 128-bit space (long-term solution)**
 - It will be large enough to install several billion computers on every square meter of the Earth's surface!
 - Problem: People has no motivation to upgrade, so how?

Classless Inter-Domain Routing

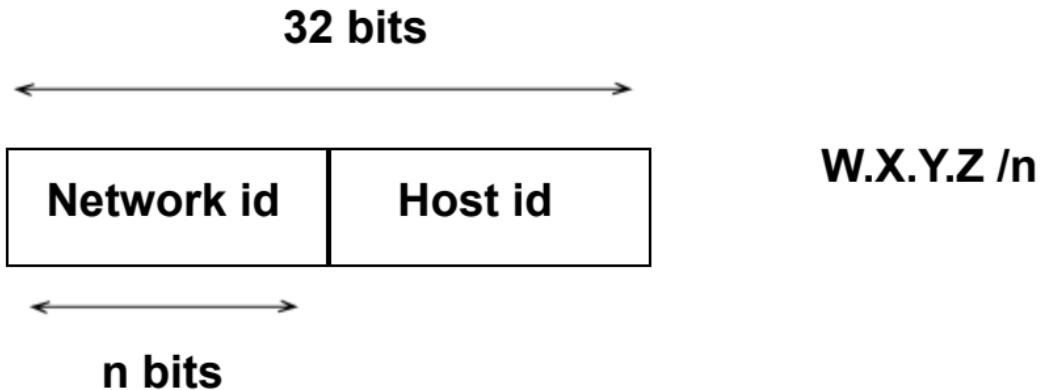
(RFC 1517 – 1519)

- Abandon the notion of classful addressing
- Key concept: length of network id (prefix) can be any length
- Consequence: add a network mask w.x.y.z (similar concept as subnet mask)

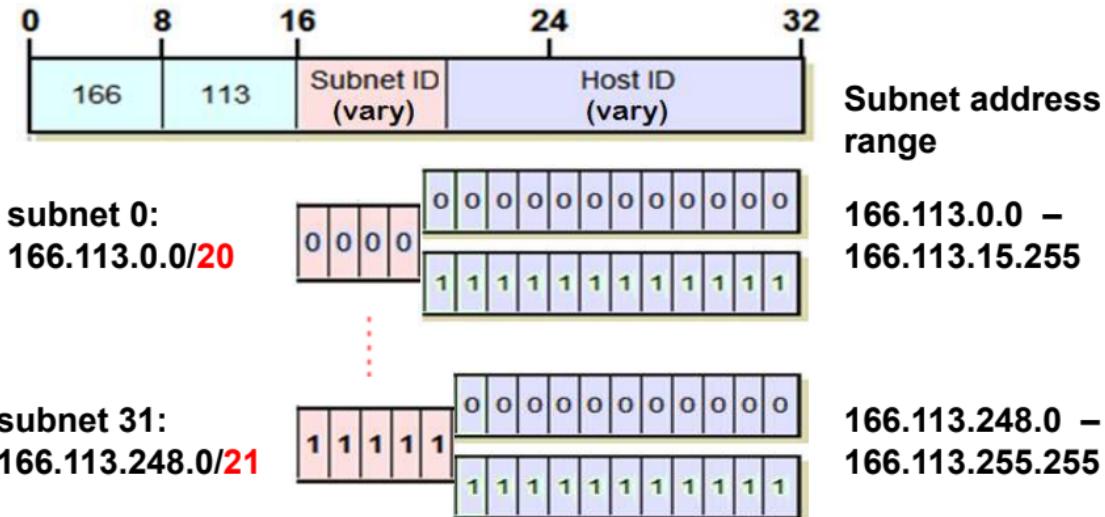


CIDR

- The IP address consists of the Network id, and Host id.
- “/n” where “n” is the number of bits allocated to Network id.

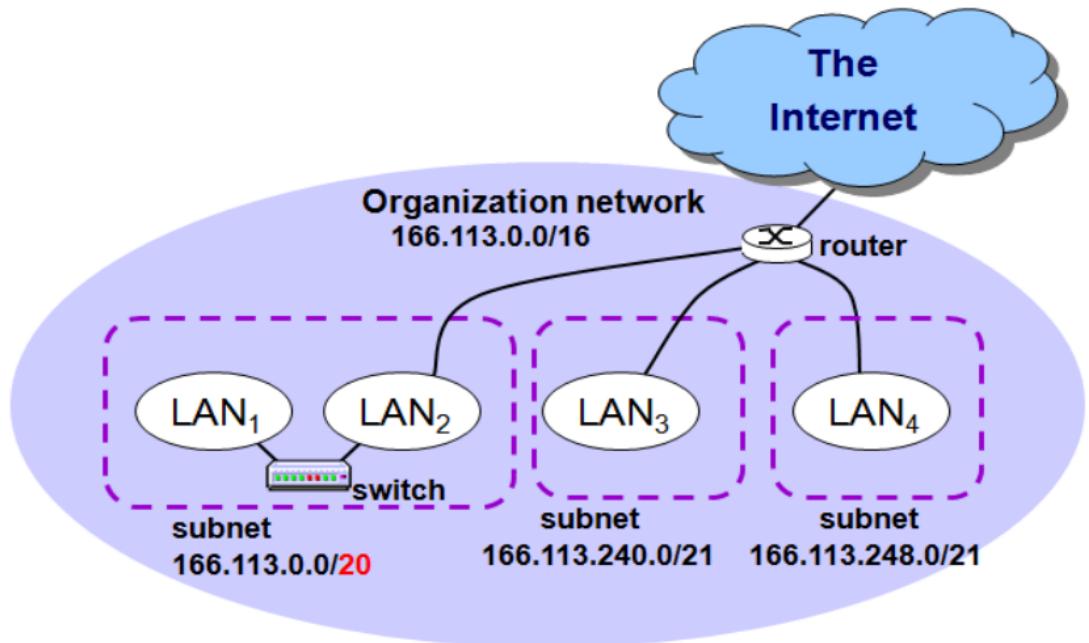


In fact, **CIDR** can be extended to **subnetting**. Hence, it's now possible to have **variable-length** subnet masks.



Note: The idea of all subnets broadcast is made obsolete. Therefore, subnet numbers including all '0's and all '1's can now be used.

An example subnetting with CIDR.

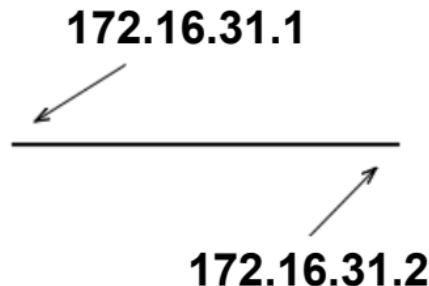


Router to router link (subnet)

- An IP address needs to be assigned to each active interface of a router.
- To optimised the use of IP address we usually assigned a /30 to the link, eg. 172.16.31.0/30

Valid IP addresses	Comments
172.16.31.00000000	Network address
172.16.31.00000001	
172.16.31.00000010	
172.16.31.00000011	Broadcast

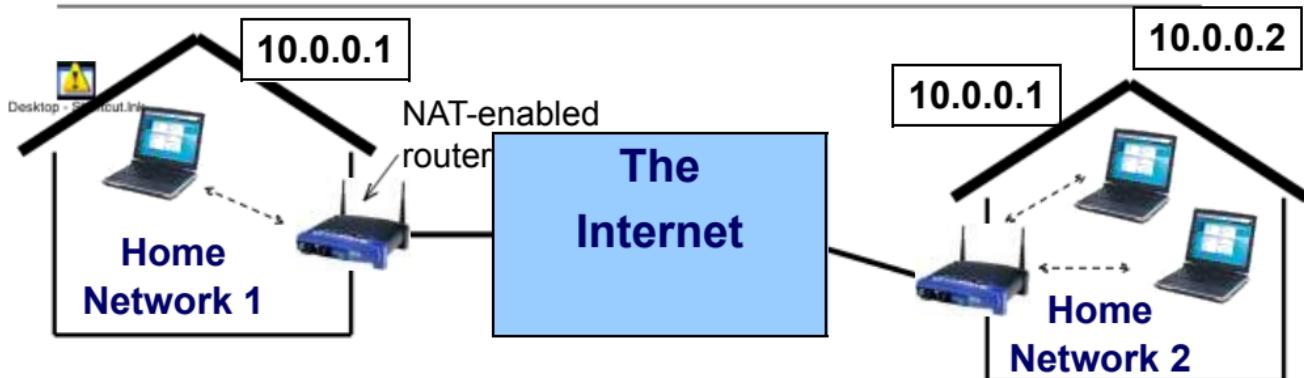
↔ Network id



Network Address Translation (NAT)

(RFC 1918) **Private IP address** for Private Internet:

- 10.0.0.0/8 (10.0.0.0 – 10.255.255.255)
- 172.16.0.0/12 (172.16.0.0 – 172.31.255.255)
- 192.168.0.0/16 (192.168.0.0 – 192.168.255.255)



- Private IP addresses will not be forwarded into the Internet.
- Hence, different private networks can re-use the same private IP addresses.

```

DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes

Ethernet adapter Local Area Connection:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :
Description . . . . . : Realtek PCIe GBE Family Controller
Physical Address. . . . . : 9C-8E-99-3E-EF-68
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes

Wireless LAN adapter Wireless Network Connection 3:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :
Description . . . . . : Microsoft Virtual WiFi Miniport Adapter
Physical Address. . . . . : AC-81-12-9E-89-51
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes

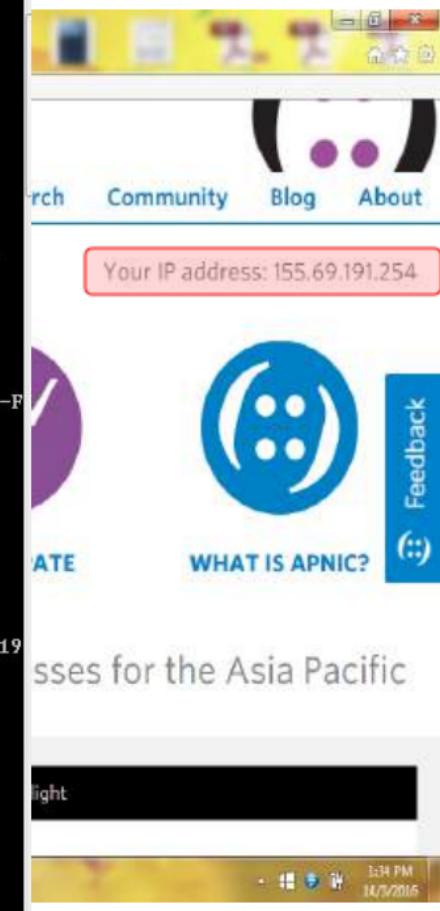
Wireless LAN adapter Wireless Network Connection 2:

Connection-specific DNS Suffix . . . . . : ntu.edu.sg
Description . . . . . : Broadcom 43224AG 802.11a/b/g/draft-n Wi-Fi Adapter
Physical Address. . . . . : AC-81-12-9E-89-51
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::941:875c:e1e9:b4c9%16(PREFERRED)
IPv4 Address. . . . . : 10.25.153.209(PREFERRED)
Subnet Mask . . . . . : 255.255.0.0
Lease Obtained. . . . . : Monday, 14 March, 2016 12:47:22 PM
Lease Expires . . . . . : Monday, 14 March, 2016 8:35:06 PM
Default Gateway . . . . . : 10.25.0.1
DHCP Server . . . . . : 155.69.3.8
DHCPv6 IAID . . . . . : 464290066
DHCPv6 Client DUID. . . . . : 00-01-00-01-15-BA-1D-3C-90-00-4E-82-29-19
DNS Servers . . . . . : 155.69.3.9
                      155.69.3.7
                      155.69.3.8
Primary WINS Server . . . . . : 155.69.5.54
Secondary WINS Server . . . . . : 155.69.4.83
NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter VirtualBox Host-Only Network:

Connection-specific DNS Suffix . . . . . :
Description . . . . . : VirtualBox Host-Only Ethernet Adapter
Physical Address. . . . . : 08-00-27-00-84-8C
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::7115:62ee:f191:137f%26(PREFERRED)
IPv4 Address. . . . . : 192.168.56.1(PREFERRED)

```



Network Address Translation

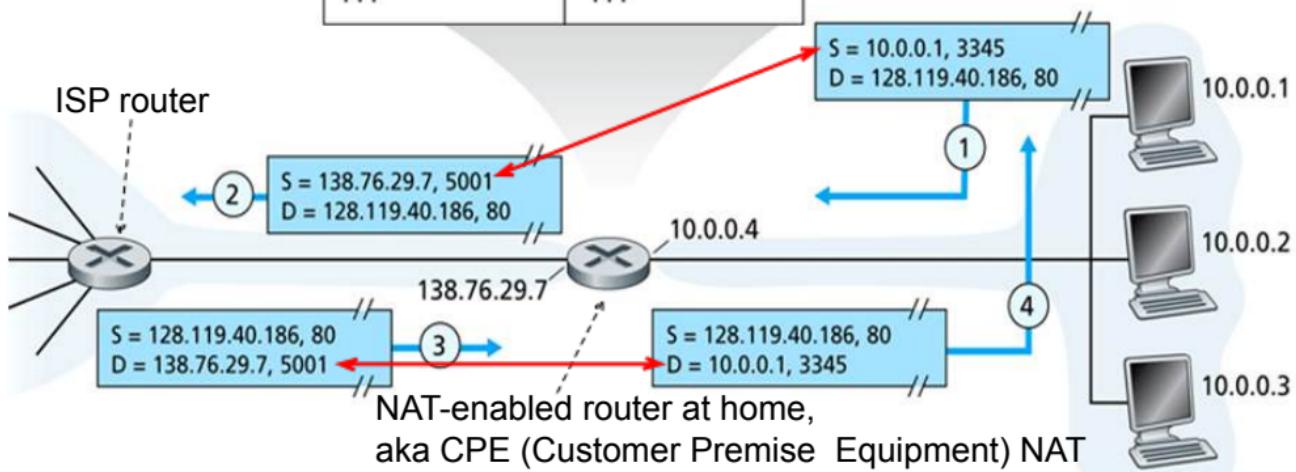
(RFC 2661, 3022)

By using a **NAT-enabled router**, only 1 IP address is required from ISP to support the whole private network to connect to Internet.

NAT translation table	
Public IP/port	Private IP/port
138.76.29.7, 5001	10.0.0.1, 3345
...	...

aka:

NAPT (Network Address and Port Translation) or simply
PAT (Port Address Translation)

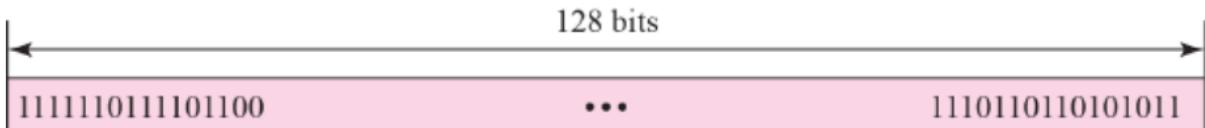


IPv6 (version 6)

Standardised since 1995 (RFC 1883, 2460)

Main enhancement in IPv6:

- **Expanded Address Space: 128-bit IPv6 address**



- **Colon hexadecimal notation:**

FDEC : BA98 : 7654 : 3210 : ADBF : BBFF : 2922 : FFFF

- **Abbreviated notation:**

- **within each 16-bit value, 0000 can be written as 0**
- **consecutive groups of 0s can be replaced by ::**





IPv6 Header

- Simplification of Header: faster processing.

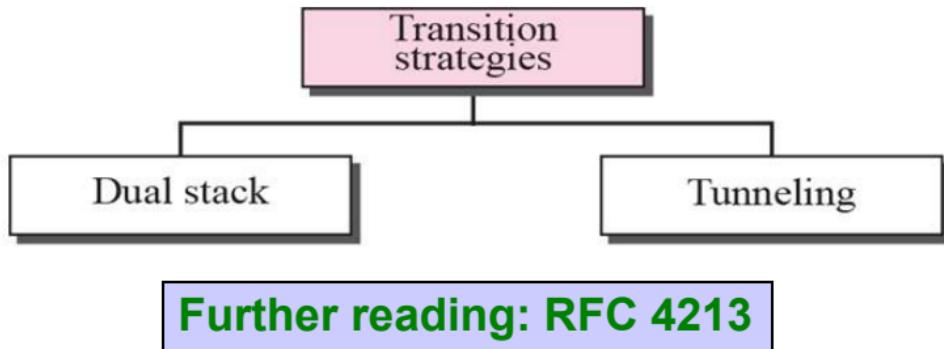
32 bits		
Ver	Traffic Class	Flow Label
Payload Length	Next Header	Hop Limit
Source Address (128 bits)		
Destination Address (128 bits)		
Data		

Note: IPv4 and IPv6 are NOT compatible. Only the first 4-bit (ver field) in IPv4 and IPv6 headers are the same to distinguish them.



Transition from IPv4 to IPv6

Not all routers can be upgraded simultaneously. So, how will the network operate with mixed IPv4 and IPv6 routers?



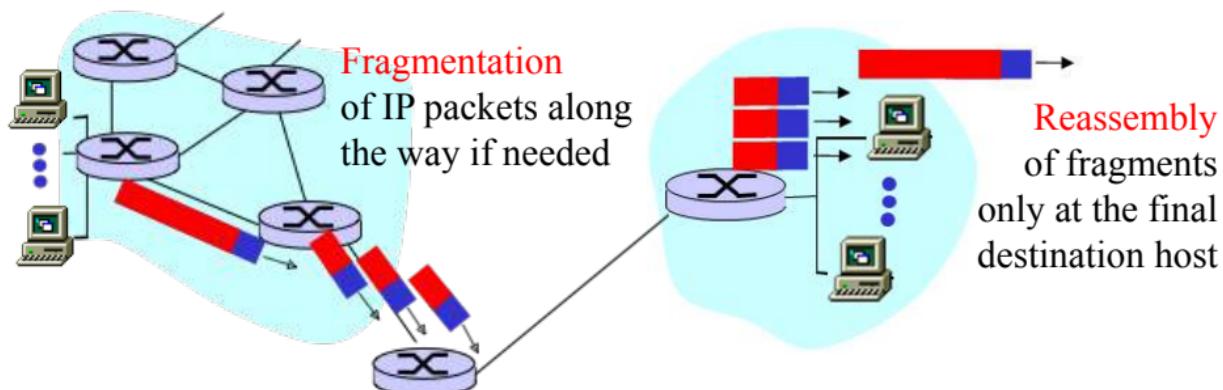
Note on terminology:

Encapsulate – lower layer carrying upper layer data

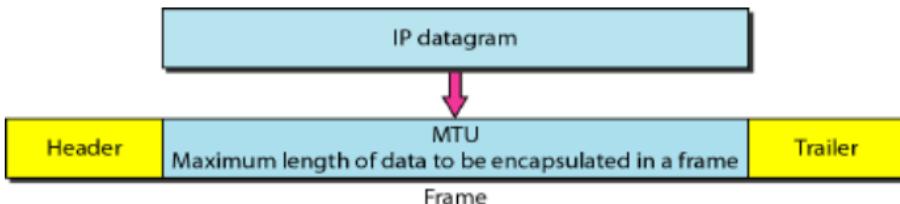
Tunnel – upper layer carrying same or lower layer data

IP Fragmentation & Reassembly

(RFC 791 and 815)

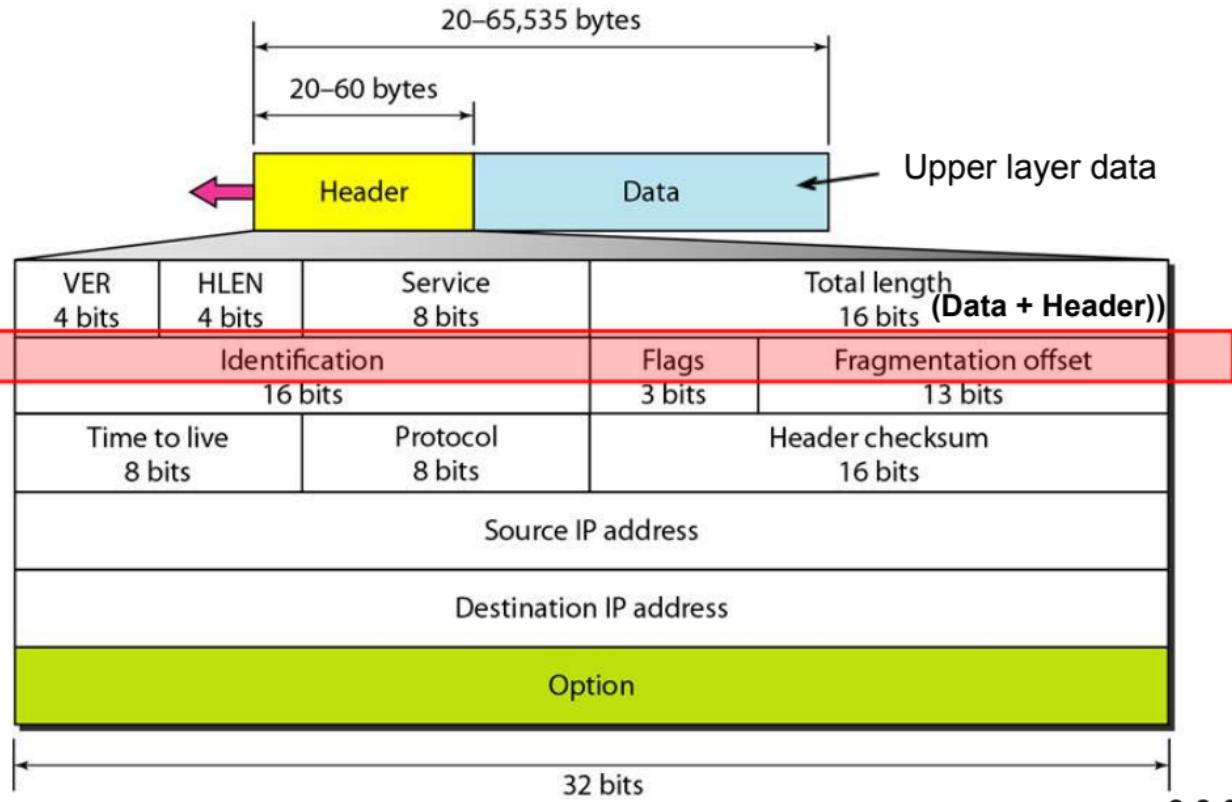


Different networks have different **MTU (Maximum Transfer Unit)**.
IP packet size(IP header+data) in each network \leq MTU of that network.

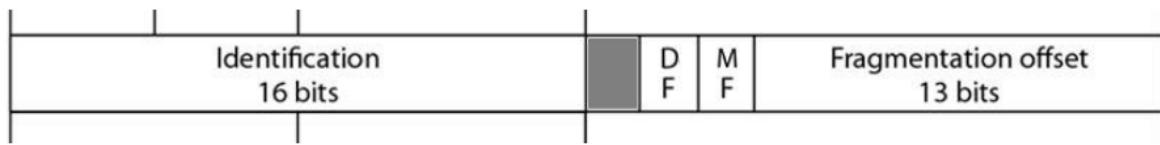


IPv4 header

(RFC 791)

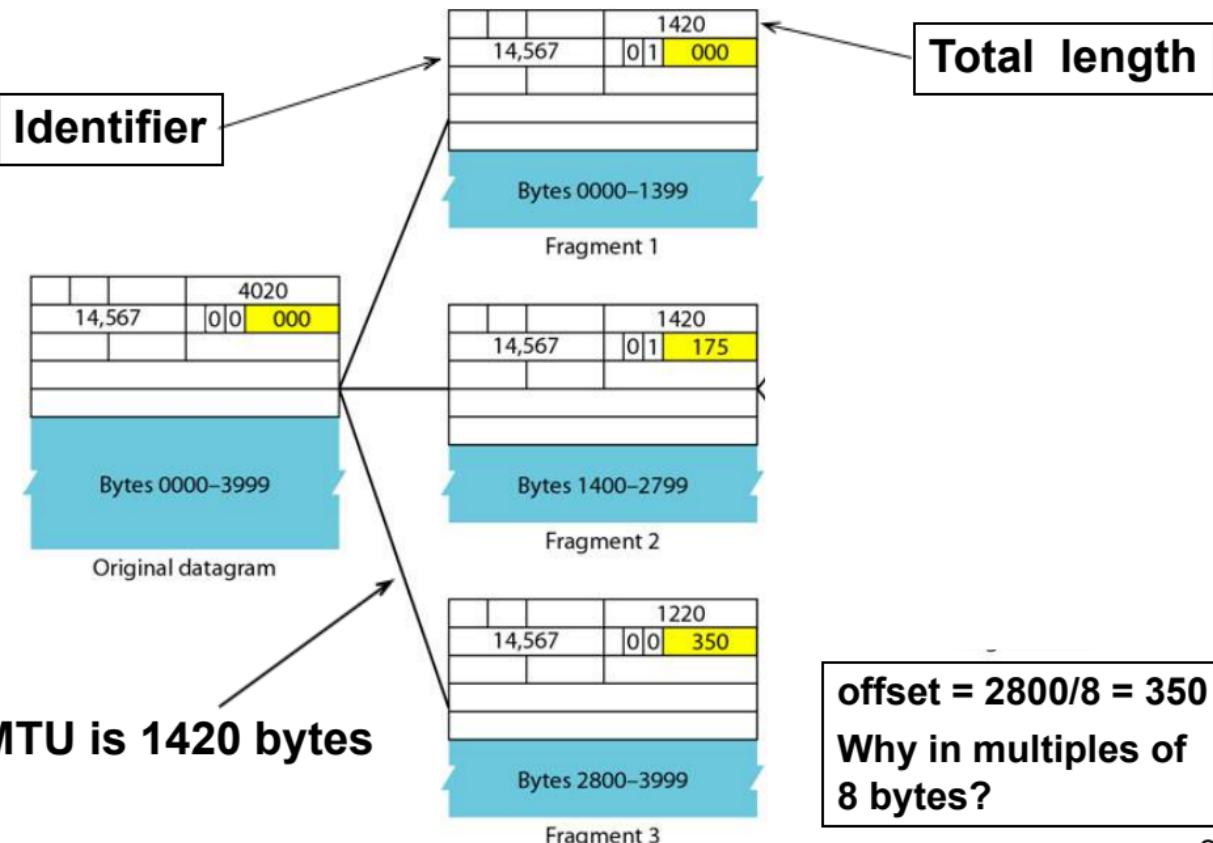


IPv4 Header – Fragmentation Fields



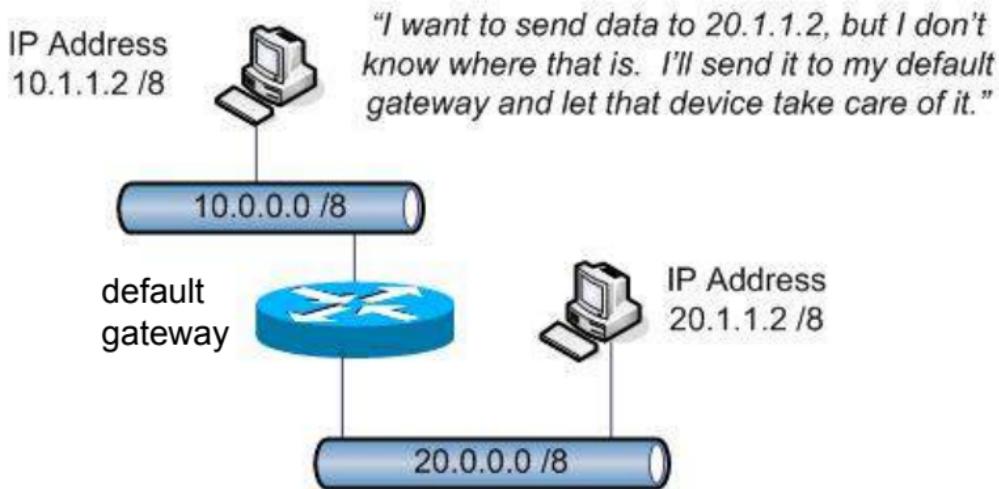
- **Identification:** For reassembly purpose, all fragments of a datagram contain the same identification value.
- **Don't Fragment (DF):** If the flag is set, the datagram is not fragmented.
- **More Fragments (MF):** The flag is set to 1 for all fragments except the last one.
- **Fragment Offset:** Tells where in the current datagram this fragment belongs. All fragments except the last one must be a **multiple of 8 bytes**, the basic fragment unit.

Example of IP Fragmentation



IP Routing

Now, we are ready to see how Internet Protocol works:



Typically, a host will not know how to send packets to destination outside its own network. Hence, it is configured with a **default gateway** (**router**) to assist in the forwarding.

Routing process

- The routing table consist of the following
 - Network address: Destination Network address
 - Cost: Arbitrary cost, number of hops
 - Next hop : Who to pass to next

Network address	Cost	Next hop
155.69.0.0	1	Directly connected
122.0.0.0	1	Directly connected
194.8.9.0	7	122.5.6.1

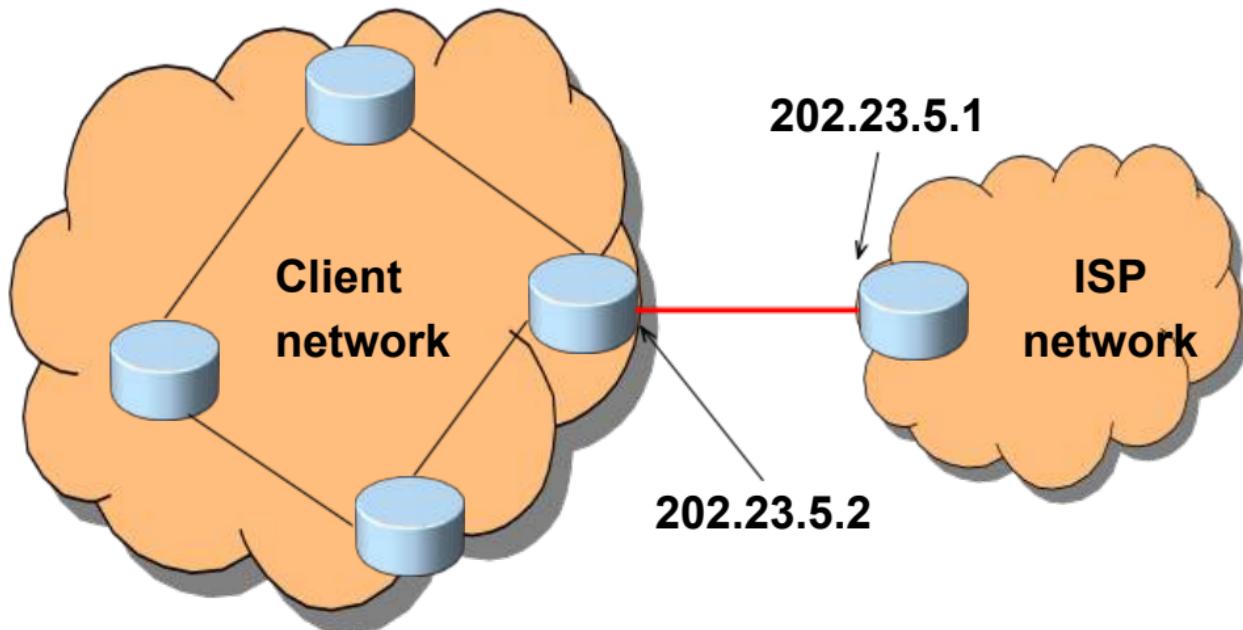


Routing process

- Extract Destination IP address and compute destination network address
- Look-up next hop
 - IF
 - Destination IP network = Directly connected THEN send on the specified interface
 - ELSE IF
 - Destination IP address appear as host specific route THEN route as specified
 - ELSE IF
 - Destination Network address is in the routing table THEN send to the specified IP address through the specified interface
 - ELSE
 - Send packet through DEFAULT route

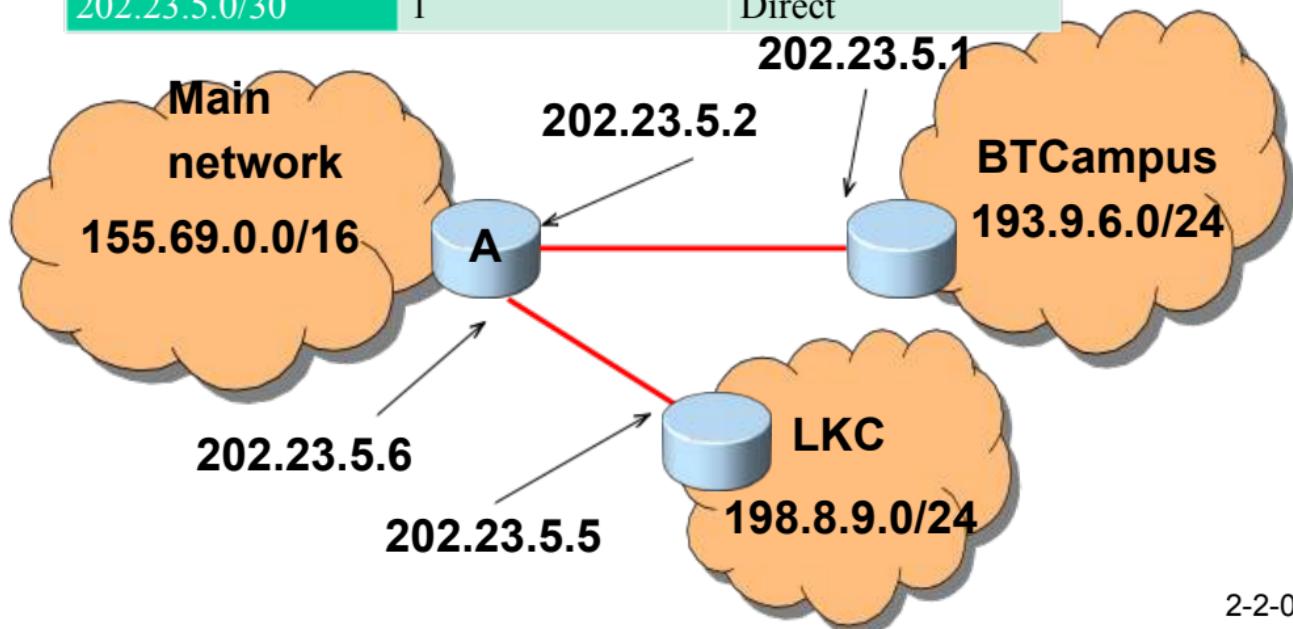
Default route

- A default route is usually provided that points to ISP router (202.23.5.1 in the example)



Network address	Cost	Next hop
155.69.0.0/16	1	Direct
193.9.6.0/24	2	202.23.5.1
198.8.9.0/24	2	202.23.5.5
202.23.5.4/30	1	Direct
202.23.5.0/30	1	Direct

-



Microsoft Windows [Version 10.0.17134.1550]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\limo>route print

Interface List

7...f4 4d 30 f9 b8 56Realtek PCIe GBE Family Controller
9...00 ff 7c e4 22 a3Zscaler Network Adapter 1.0.2.0
1.....Software Loopback Interface 1

IPv4 Route Table

Active Routes:

Network Destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	155.69.143.254	155.69.142.169	25
127.0.0.0	255.0.0.0	On-link	127.0.0.1	331
127.0.0.1	255.255.255.255	On-link	127.0.0.1	331
127.255.255.255	255.255.255.255	On-link	127.0.0.1	331
155.69.140.0	255.255.252.0	On-link	155.69.142.169	281
155.69.142.169	255.255.255.255	On-link	155.69.142.169	281
155.69.143.255	255.255.255.255	On-link	155.69.142.169	281
224.0.0.0	240.0.0.0	On-link	127.0.0.1	331
224.0.0.0	240.0.0.0	On-link	155.69.142.169	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	331
255.255.255.255	255.255.255.255	On-link	155.69.142.169	281

Persistent Routes:

None

IPv6 Route Table

Active Routes:

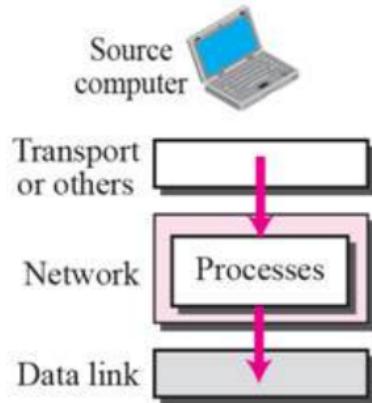
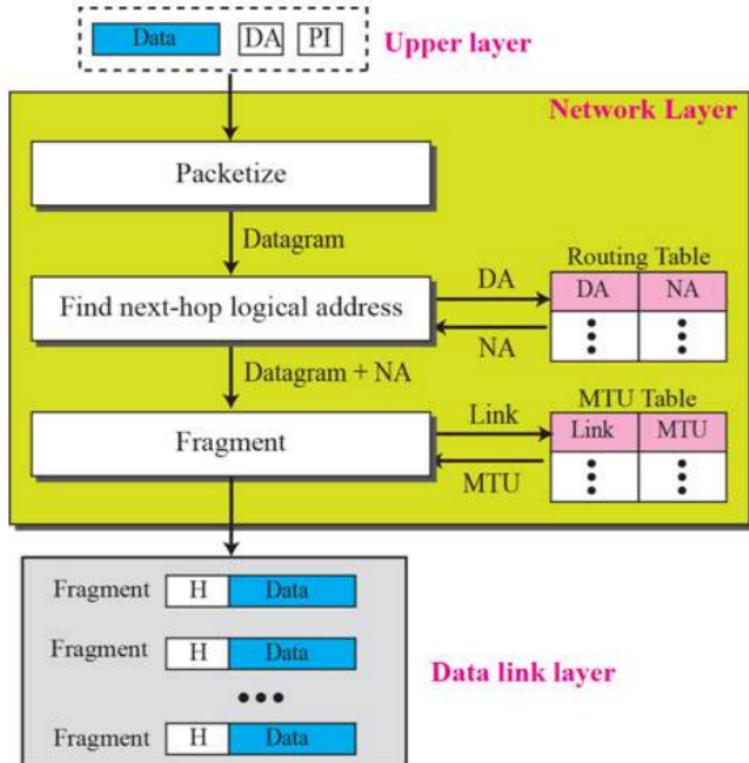
If Metric	Network Destination	Gateway
1 331	::/128	On-link
7 281	fe00::/64	On-link
7 281	fe80::8cd8:55f3:71ff:6ad7/128	On-link
1 331	ff00::/8	On-link
7 281	ff00::/8	On-link

Persistent Routes:

None

C:\Users\limo>

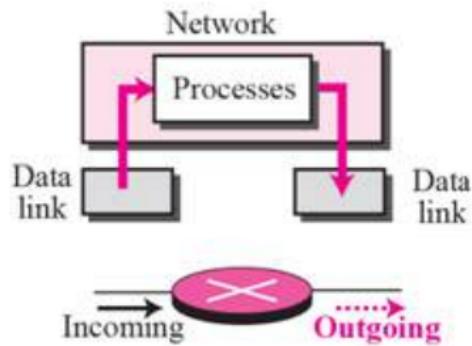
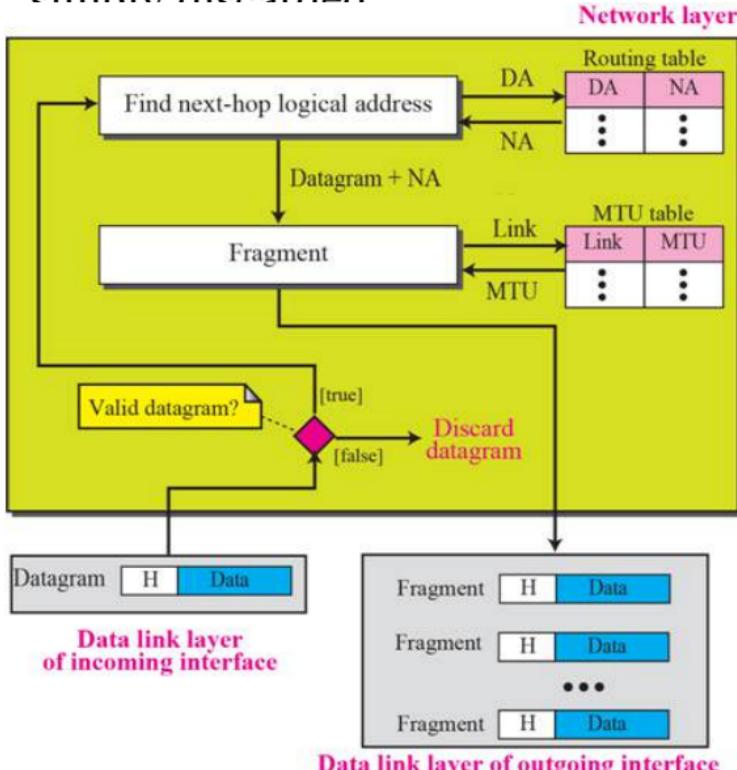
Combining Routing and Fragmentation – At source host, IP encapsulates upper layer data into packet. Then determine route and MTU, and fragment if needed.



Legend

Data	Upper layer data
DA	Destination IP address
PI	Protocol ID
NA	Next-hop IP address
MTU	Maximum Transfer Unit
H	Datagram header

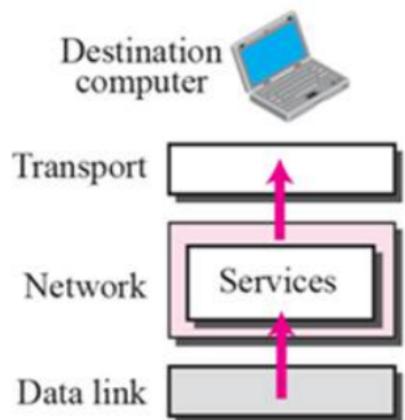
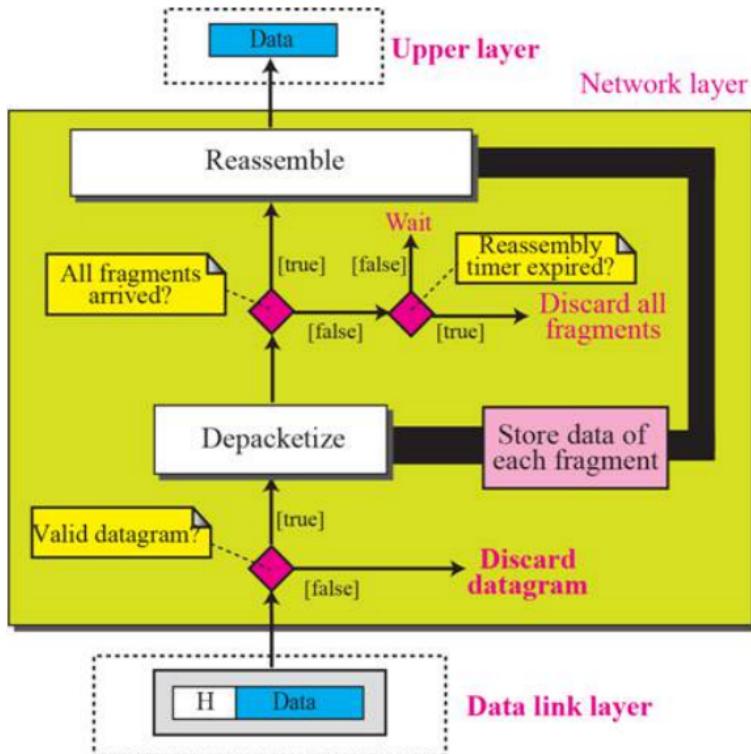
At each router, IP determines **next route** and **MTU**, and further **fragment** packets if necessary. But if DF flag is set, then it is **simply discarded**



Legend

Data	Upper layer data
DA	Destination IP address
NA	Next-hop IP address
MTU	Maximum Transfer Unit
H	Datagram header

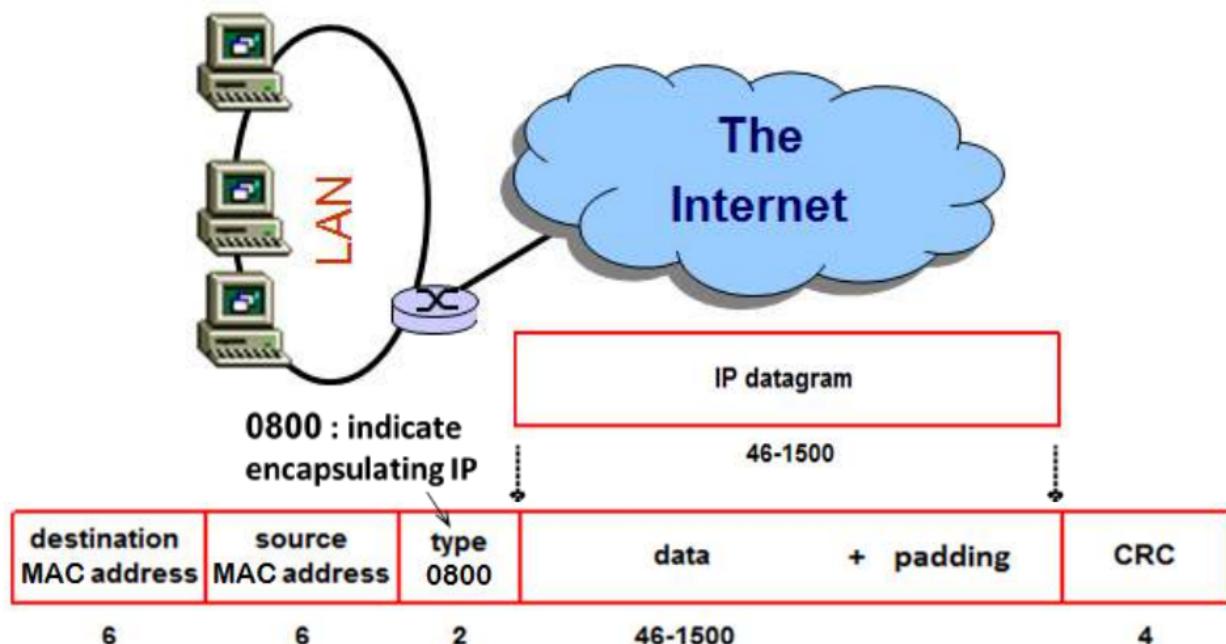
Finally, at **destination host**, IP **reassembles** packets (if fragmented) before returning data to upper layer.



Legend

Data	Data of upper layer
H	Datagram header

To complete the picture, let's now consider IP over Ethernet (data link layer protocol).

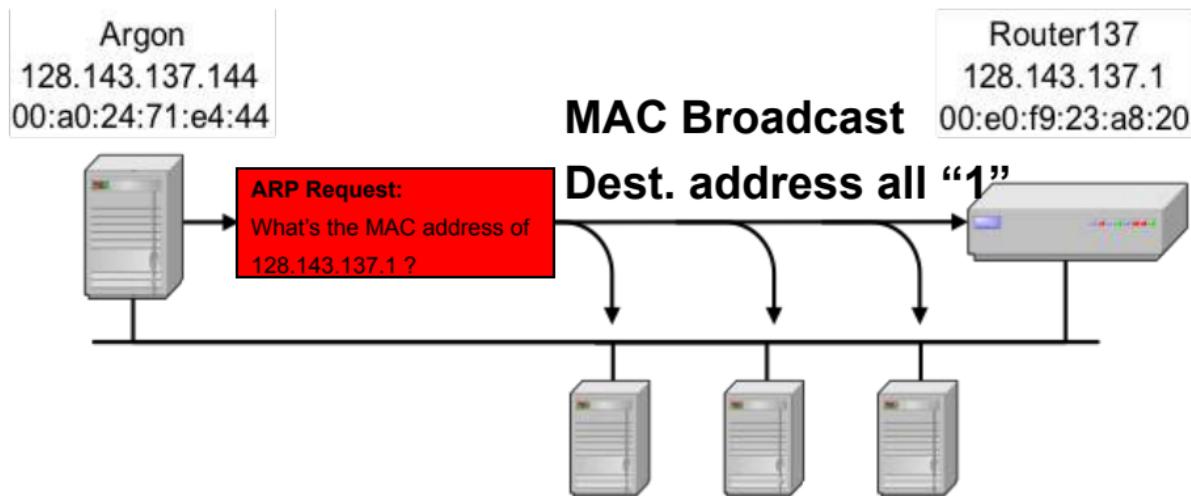


- How to go from source to destination when only IP address is known but Ethernet requires MAC address?

Address Resolution Protocol (RFC 826)

ARP Request:

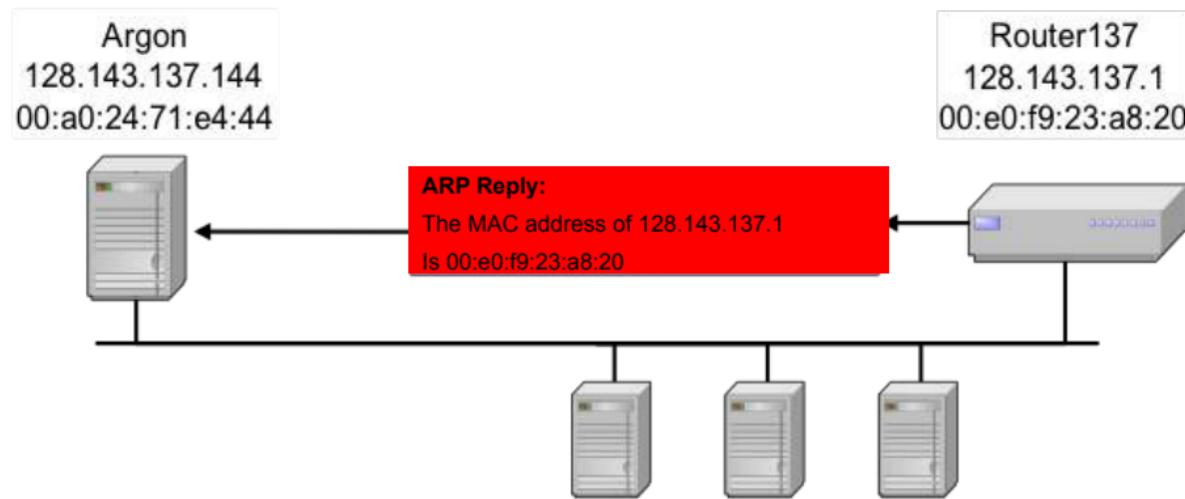
Argon broadcasts an ARP request to all stations on the network: “What is the MAC address of Router137?”



Address Resolution Protocol (RFC 826)

ARP Reply:

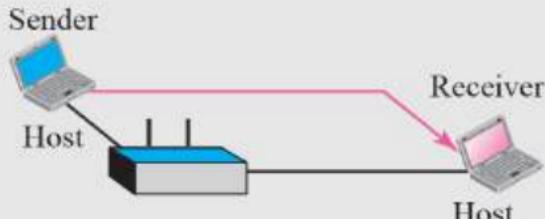
Router 137 responds with an ARP Reply which contains its MAC address



ARP Packet – Target IP Address

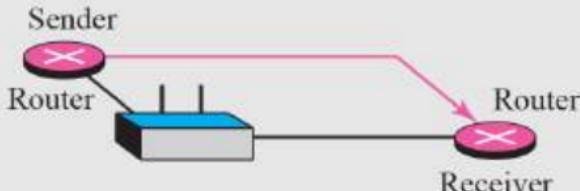
Case 1: A host has a packet to send to a host on the same network.

Target IP address:
Destination address in the IP datagram



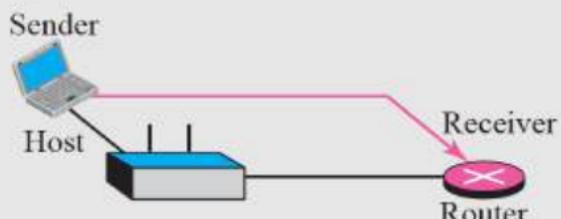
Case 3: A router has a packet to send to a host on another network.

Target IP address:
IP address of a router



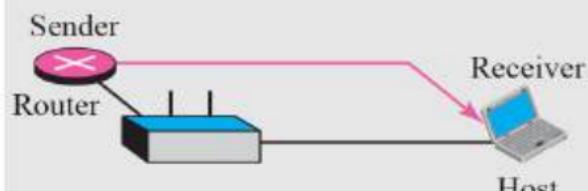
Case 2: A host has a packet to send to a host on another network.

Target IP address:
IP address of a router

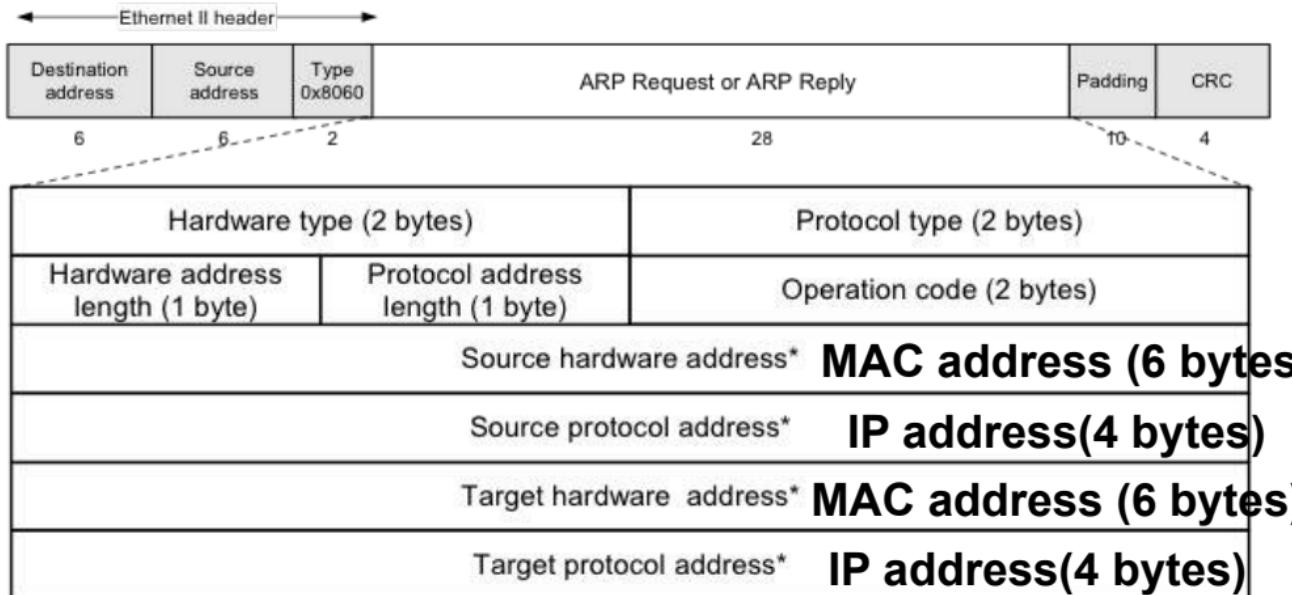


Case 4: A router has a packet to send to a host on the same network.

Target IP address:
Destination address in the IP datagram



ARP packet is sent directly over Ethernet frame:



* Note: The length of the address fields is determined by the corresponding address length fields

ARP Example

- *ARP Request from Argon(Client):*

Source hardware address: 00:a0:24:71:e4:44



Source protocol address: 128.143.137.144

Target hardware address: 00:00:00:00:00:00

Target protocol address: 128.143.137.1

- *ARP Reply from Router137:*

Source hardware address: 00:e0:f9:23:a8:20



Source protocol address: 128.143.137.1

Target hardware address: 00:a0:24:71:e4:44

Target protocol address: 128.143.137.144

ARP Cache

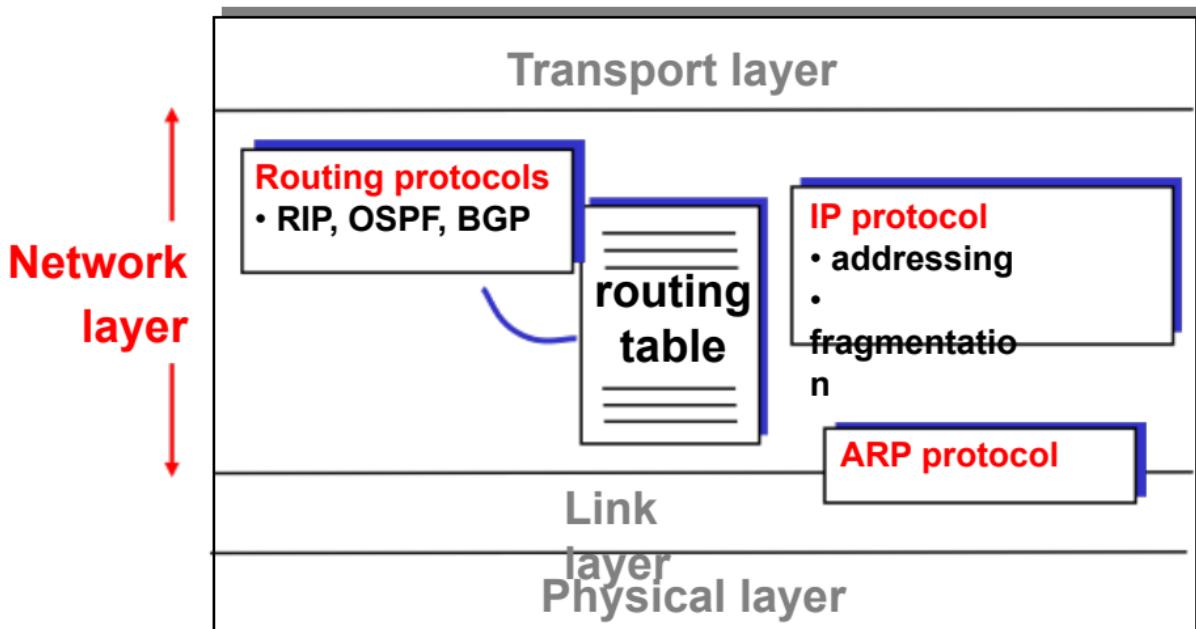
- Since sending an ARP request/reply for each IP datagram is inefficient, hosts maintain a cache (ARP Cache) of current entries. Typically, the entries are configured to expire after 2-20 minutes.

```
Microsoft Windows [Version 10.0.17134.1550]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\limo>arp -a

Interface: 155.69.142.169 --- 0x7
  Internet Address      Physical Address      Type
  155.69.141.31          90-b1-1c-9f-75-52    dynamic
  155.69.142.71          94-c6-91-91-31-3d    dynamic
  155.69.142.7           78-7b-8a-d0-18-e4    dynamic
  155.69.142.19          c0-3f-d5-fa-08-5a    dynamic
  155.69.142.22          a4-bb-6d-43-37-a3    dynamic
  155.69.142.23          78-7b-8a-d5-82-f1    dynamic
  155.69.142.30          78-7b-8a-cd-bd-fa    dynamic
  155.69.142.58          c8-f7-50-9c-8c-6d    dynamic
  155.69.142.67          80-4a-14-67-27-de    dynamic
  155.69.142.69          ec-08-6b-43-14-02    dynamic
  155.69.142.85          00-1f-b5-33-fc-64    dynamic
  155.69.142.101         78-7b-8a-ca-f3-12    dynamic
  155.69.142.114         f0-18-98-f1-0e-cf    dynamic
```

Summary of IP



IP Related Protocols

ICMP, PING, TRACERT

Internet Control Message Protocol (RFC792 & 1122)

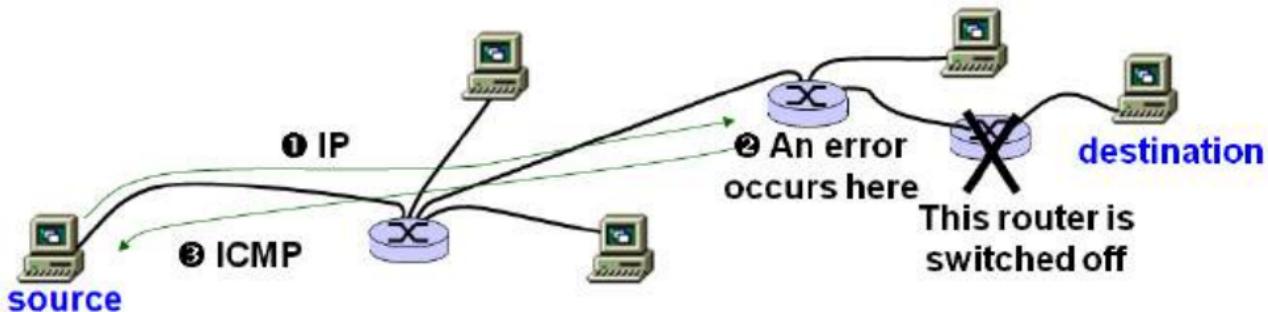
Wikipedia: The Internet Control Message Protocol (ICMP) is a supporting protocol in the Internet protocol suite. It is used by network devices, including routers, to send error messages and operational information indicating, for example, that a requested service is not available or that a host or router could not be reached.

Internet Control Message Protocol

(RFC 792 & 1122)

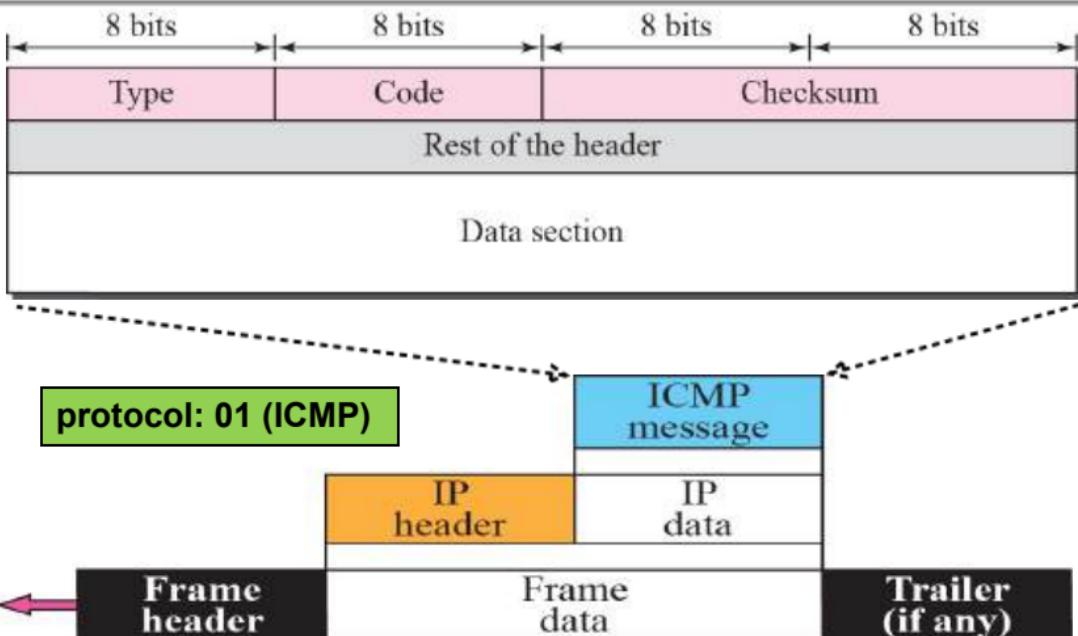
What if a source IP packet cannot be delivered to destination?

Although IP does not perform error control, it must still report errors. Otherwise, the source TCP (transport layer) may retransmit all data causing IP to send same packets again that can never reach the destination.

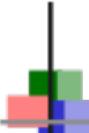


Hence, ICMP is created for routers/hosts to report errors to the source. It is the responsibility of the source to handle the reported problems.

ICMP packet is sent over IP packet, which in turn is sent over data link layer protocol, e.g. Ethernet.

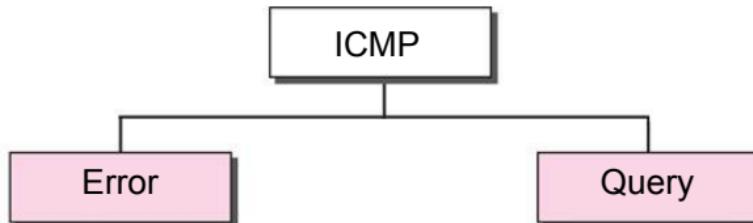


Note: Although ICMP is over IP, it is considered a layer-3 protocol.



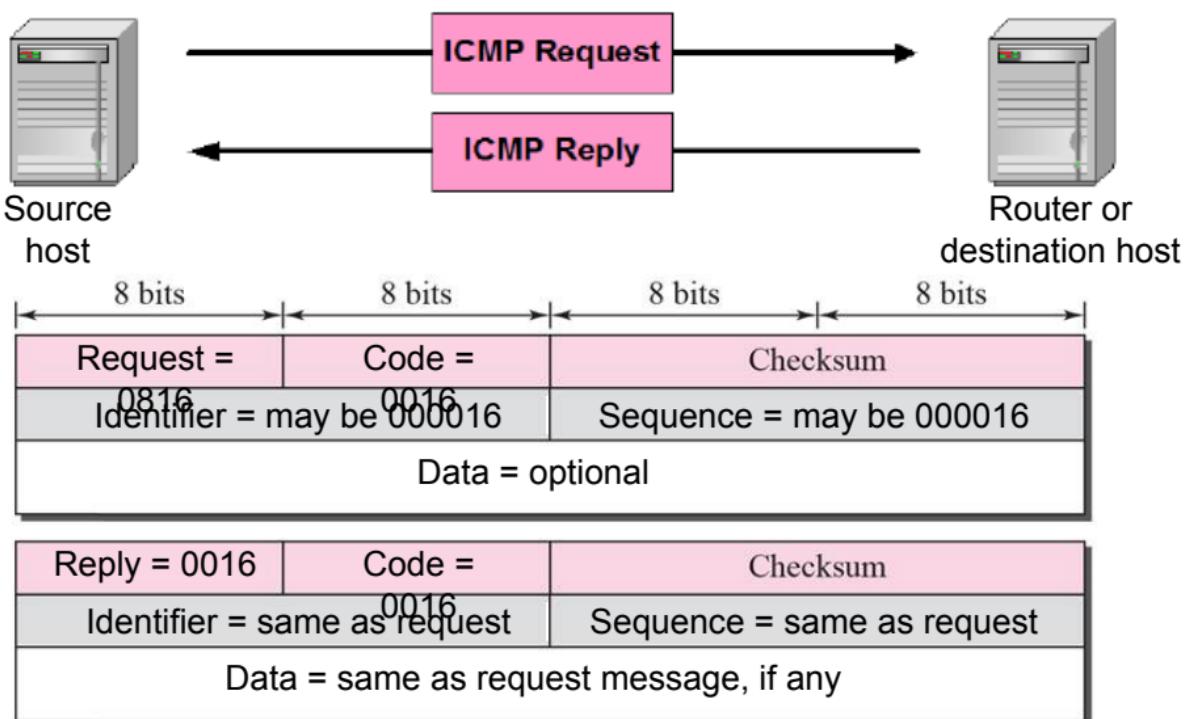
ICMP

Generally, there are 2 types of ICMP messages:



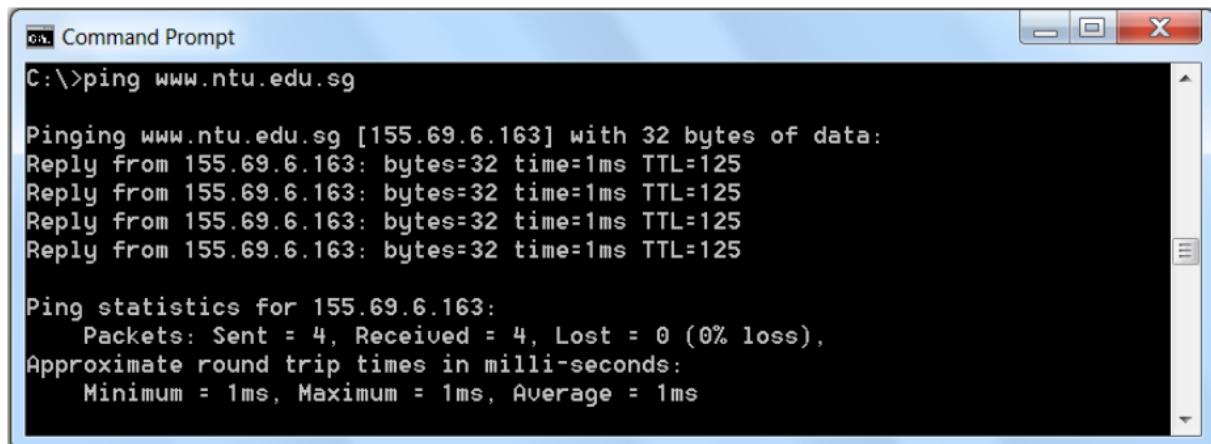
<i>Category</i>	<i>Type</i>	<i>Message</i>
Error-reporting messages	3	Destination unreachable
	4	Source quench
	11	Time exceeded
	12	Parameter problem
	5	Redirection
Query messages	8 or 0	Echo request or reply
	13 or 14	Timestamp request or reply

ICMP echo request message is sent by a source host to query a router or destination host, which will respond with an **ICMP echo reply** message.



ping network tool

ping (Packet InterNet Grouper) is a useful network debugging tool for testing the reachability of a host/router. Basically, it operates by sending/receiving ICMP echo messages.

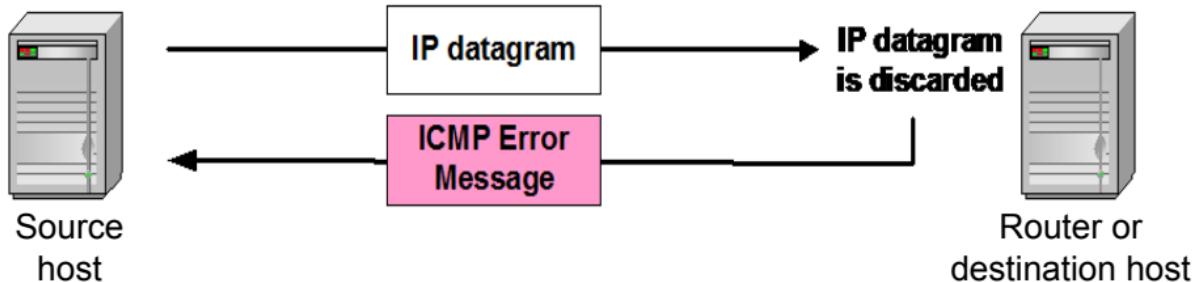


```
C:\>ping www.ntu.edu.sg

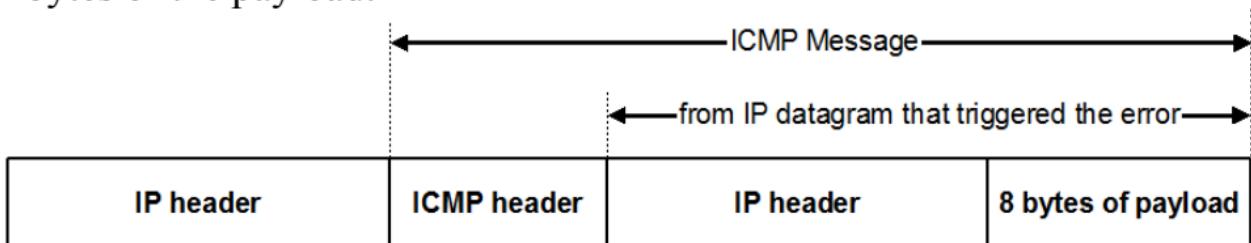
Pinging www.ntu.edu.sg [155.69.6.163] with 32 bytes of data:
Reply from 155.69.6.163: bytes=32 time=1ms TTL=125

Ping statistics for 155.69.6.163:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

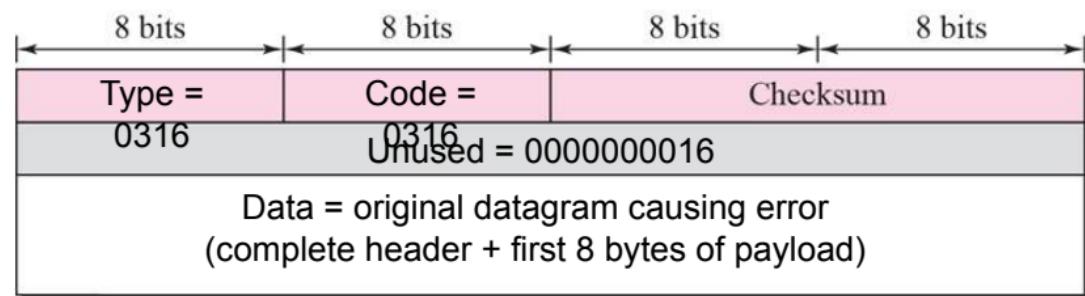
ICMP error message is sent by a router or destination host to inform source host that its datagram has been received in error and discarded.



The data section of ICMP error message will contain part of the original IP datagram in error - the complete IP header and first 8 bytes of the payload:

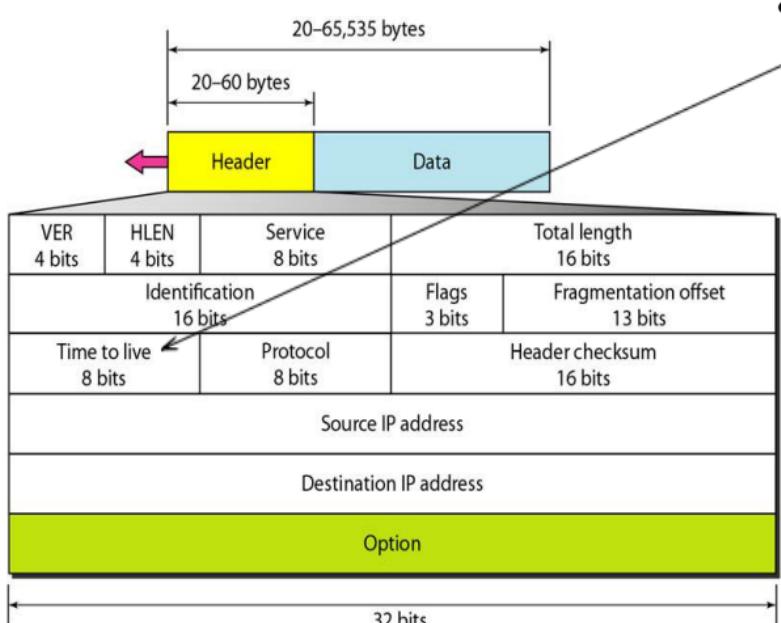


Example: ICMP Port Unreachable



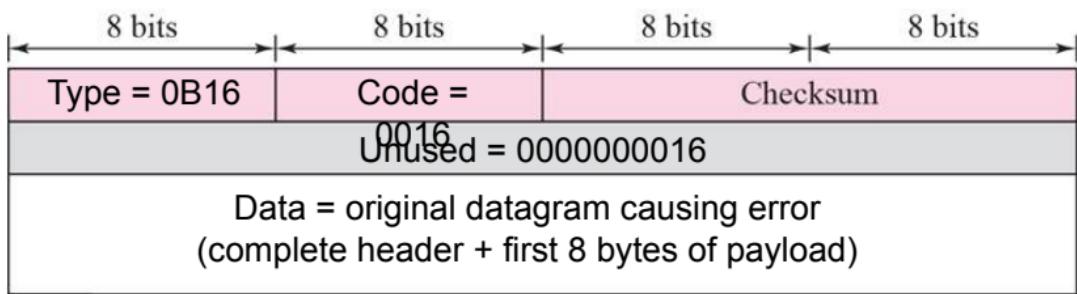
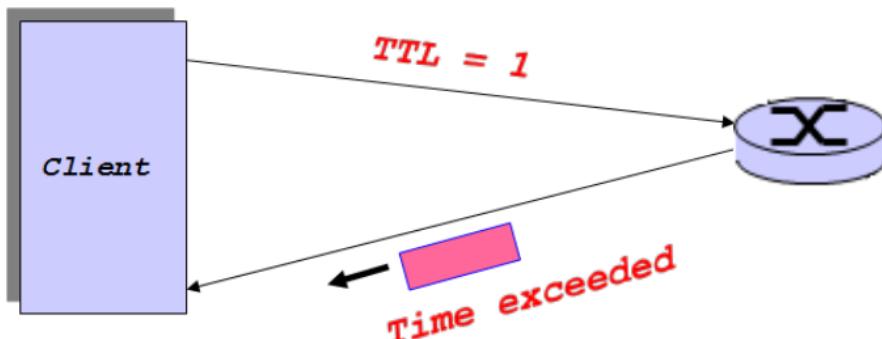
Example: ICMP Time Exceeded

Finally, last field in **IP header** that we've yet to discussed :)



- **Time to Live:** It is a counter used to limit the IP datagram lifetime, number of hops. The counter is initialized with an integer value up to 255, and when it reaches **zero**, the IP datagram is discarded.

Example: ICMP Time Exceeded





Starting at \$5.97

Discover how ▾

Track a Shipment

Help

Enter up to 30 FedEx tracking, door tag or FedEx Office order numbers (one per line).

Track

My Shipments

Track and save tracking results
for your next visit to fedex.com

Ship date:

Wed 10/18/2017

SANTA CLARA, CA US



In transit

On FedEx vehicle for delivery

SINGAPORE, SG

Confirm Delivery

Request Notifications

More actions ▾

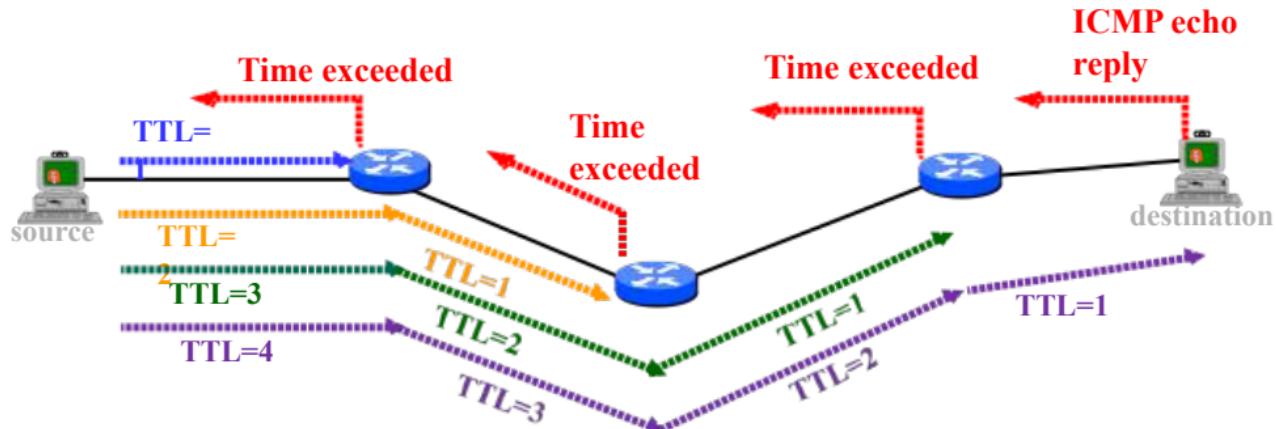
Travel History

Help Hide

Date/Time	Activity	Location
- 10/23/2017 - Monday		
7:45 am	On FedEx vehicle for delivery	SINGAPORE SG
- 10/21/2017 - Saturday		
8:25 am	At local FedEx facility	SINGAPORE SG
8:25 am	At local FedEx facility	SINGAPORE SG
7:37 am	In transit	SINGAPORE SG
	Package available for clearance	SINGAPORE SG
7:37 am	International shipment release - Import	SINGAPORE SG
7:37 am	At local FedEx facility	SINGAPORE SG
- 10/20/2017 - Friday		
7:13 pm	At destination sort facility	SINGAPORE SG
- 10/19/2017 - Thursday		
3:55 am	Departed FedEx location	OAKLAND, CA
- 10/18/2017 - Wednesday		
10:56 pm	In transit	OAKLAND, CA
10:10 pm	Arrived at FedEx location	OAKLAND, CA
9:05 pm	Left FedEx origin facility	SUNNYVALE, CA
4:34 pm	Picked up	SUNNYVALE, CA
3:48 pm	Shipment information sent to FedEx	SUNNYVALE, CA

tracert network tool

tracert (trace route) is another useful network debugging tool for tracing a path from source to destination host.



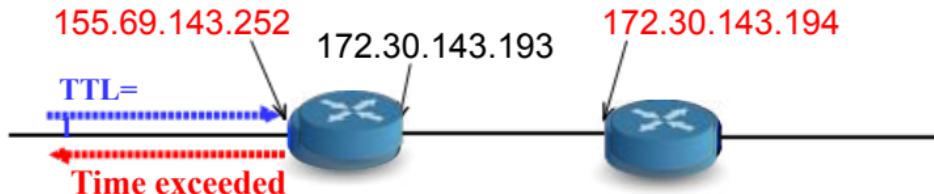
It operates by sending a sequence of ICMP echo request over IP with TTL set to 1, 2, ... until the destination is reached.

Note: tracert can also be implemented using UDP (layer-4) with unused port number (e.g. 33534) over IP.

tracert network tool

```
Command Prompt  
C:\>tracert www.ntu.edu.sg  
  
Tracing route to www.ntu.edu.sg [155.69.6.163]  
over a maximum of 30 hops:  
  
 1  <1 ms    <1 ms    <1 ms  155.69.143.252  
 2  1 ms     1 ms     1 ms  172.30.143.194  
 3  1 ms     1 ms     1 ms  172.30.2.193  
 4  1 ms     1 ms     1 ms  www.ntu.edu.sg [155.69.6.163]  
  
Trace complete.
```

Recall IP address is associated with an interface, not a device. Typically, the source address field of the IP that carries the ICMP message is the address of the interface that sends it.



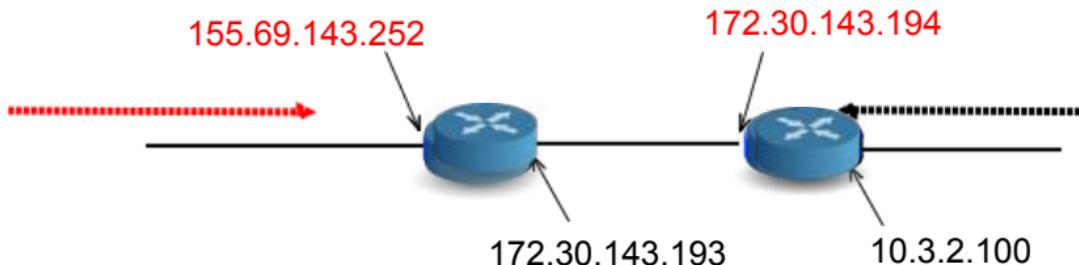
Tracert example

Tracert to 10.3.2.5

>155.69.143.252
>172.30.143.194
>10.3.2.5

Tracert to 155.69.143.100

>10.3.2.100
>172.30.143.193
>155.69.143.100



Singapore to Switzerland

traceroute to www.isg.hest.ethz.ch (129.132.19.217), 30 hops max, 60 byte packets

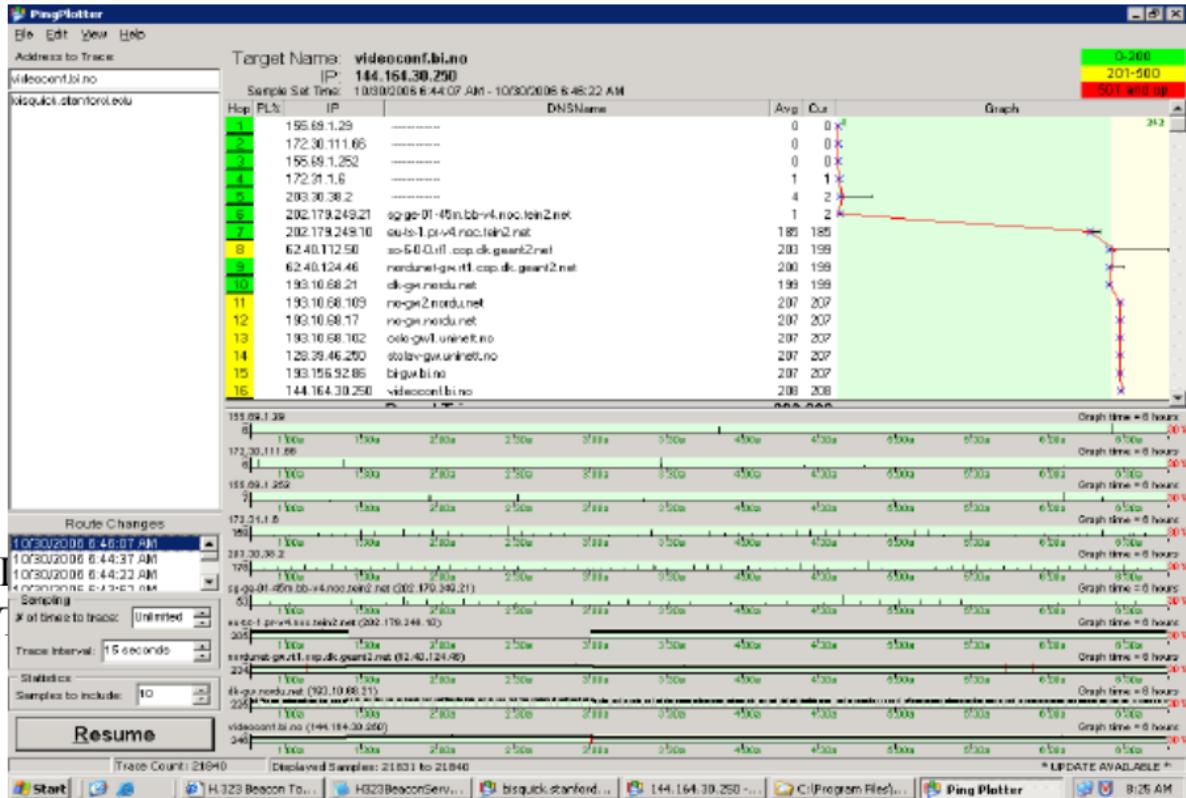
1 203.30.39.254 (203.30.39.254) 0.843 ms 1.023 ms 0.804 ms
2 sg-ge-01-v4.bb.tein3.net (202.179.249.57) 0.958 ms 0.957 ms 0.859 ms
3 mb-so-01-v4.bb.tein3.net (202.179.249.82) 58.638 ms 58.545 ms 58.563 ms
4 eu-mad-pr-v4.bb.tein3.net (202.179.249.86) 171.371 ms 171.378 ms 171.353 ms
5 as2.rt1.gen.ch.geant2.net (62.40.112.25) 193.451 ms 193.471 ms 193.444 ms
6 switch-lb2-gw.rt1.gen.ch.geant.net (62.40.124.106) 193.580 ms 193.627 ms
193.425 ms
7 swiLS2-10GE-1-3.switch.ch (130.59.37.2) 194.444 ms 194.408 ms 194.485 ms
8 swiEZ2-10GE-1-1.switch.ch (130.59.36.206) 197.782 ms 197.859 ms 197.806 ms
9 rou-gw-rz-tengig-to-switch.ethz.ch (192.33.92.1) 197.818 ms 197.834 ms
197.811 ms
10 rou-fw-rz-rz-gw.ethz.ch (192.33.92.169) 197.883 ms 198.359 ms 200.217 ms
11 * * *

Singapore to Brazil

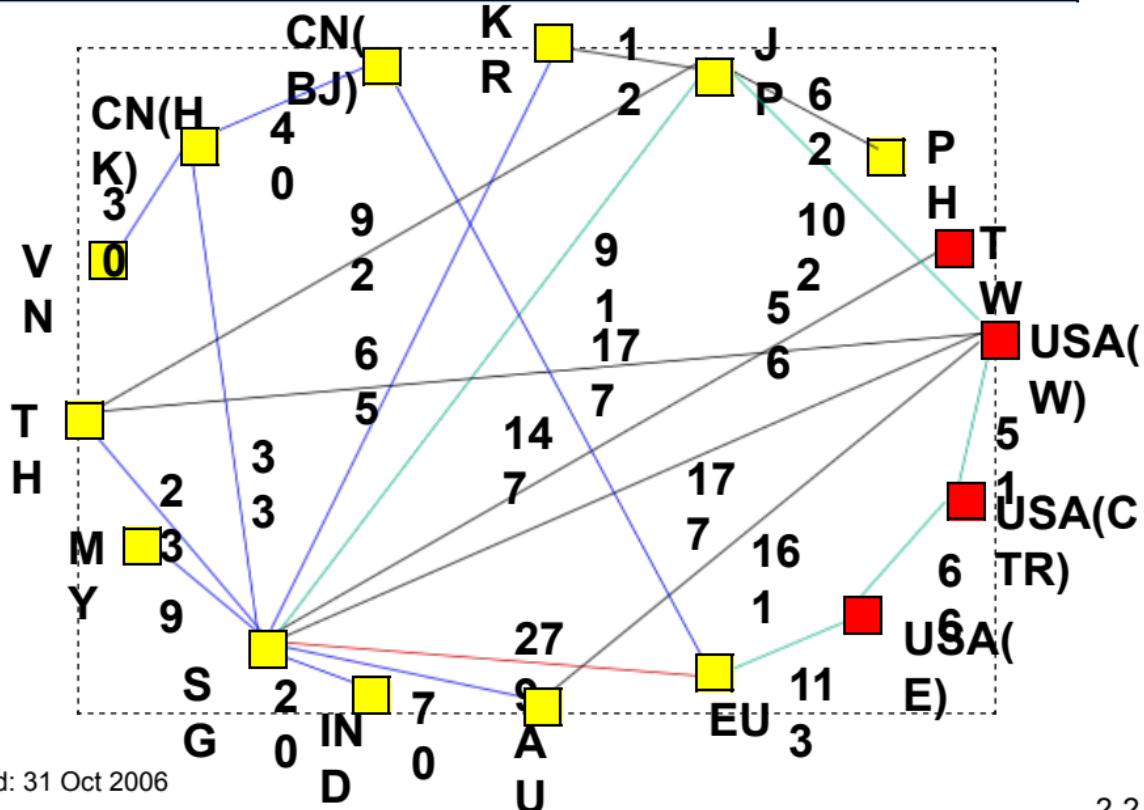
traceroute to www.rnp.br (200.143.193.5), 64 hops max

- 1 203.30.39.254 (203.30.39.254) 0.457ms 0.340ms 0.409ms
- 2 192.31.99.85 (192.31.99.85) 0.742ms 0.345ms 0.430ms
- 3 192.31.99.249 (192.31.99.249) 179.086ms 179.436ms 180.342ms
- 4 192.31.99.161 (192.31.99.161) 229.212ms 232.749ms 234.068ms
- 5 192.31.99.134 (192.31.99.134) 237.848ms 223.930ms 223.785ms
- 6 **64.57.28.51 (64.57.28.51) 241.727ms 64.57.28.201 (64.57.28.201)**
- 757.368ms 749.757ms**
- 7 198.32.11.106 (198.32.11.106) 385.874ms 385.820ms 385.839ms
- 8 200.0.204.130 (200.0.204.130) 386.440ms 386.427ms 386.377ms
- 9 200.143.252.70 (200.143.252.70) 394.697ms 393.998ms 394.001ms
- 10 200.143.255.45 (200.143.255.45) 394.753ms 394.309ms 394.286ms
- 11 200.143.193.129 (200.143.193.129) 394.068ms 394.102ms 394.063ms

tracert network tool



TEIN tracert network map



Created: 31 Oct 2006



CE3005: Computer Networks
CZ3006: Netcentric Computing

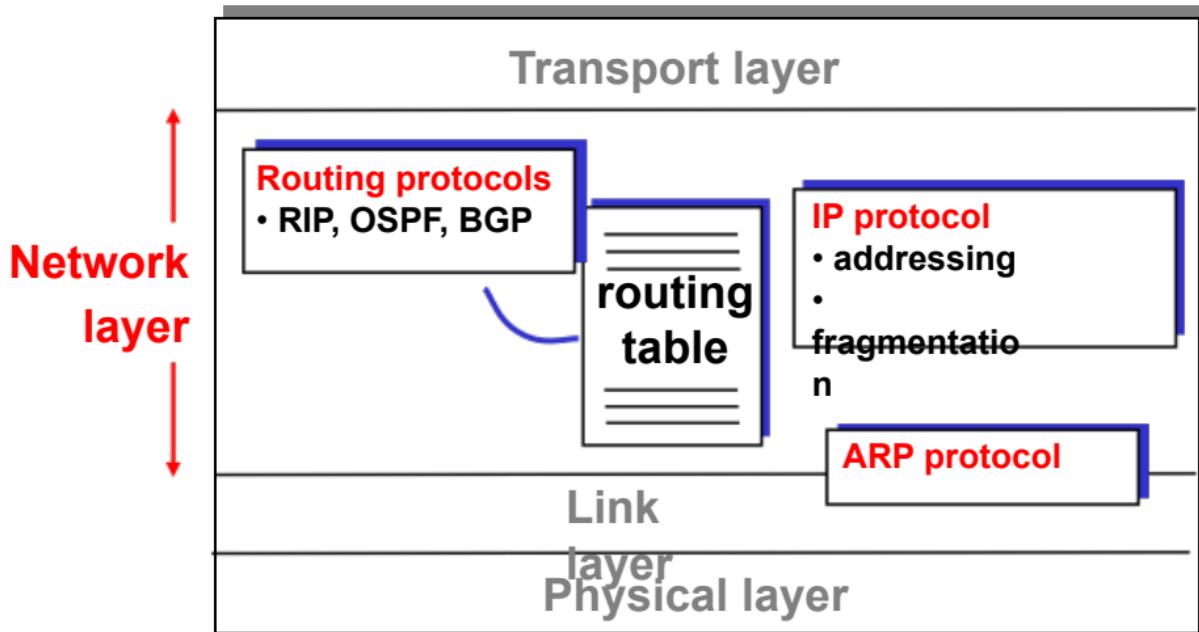
Network Layer - IP Routing Protocols

Prof.Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

School of Computer Science and Engineering

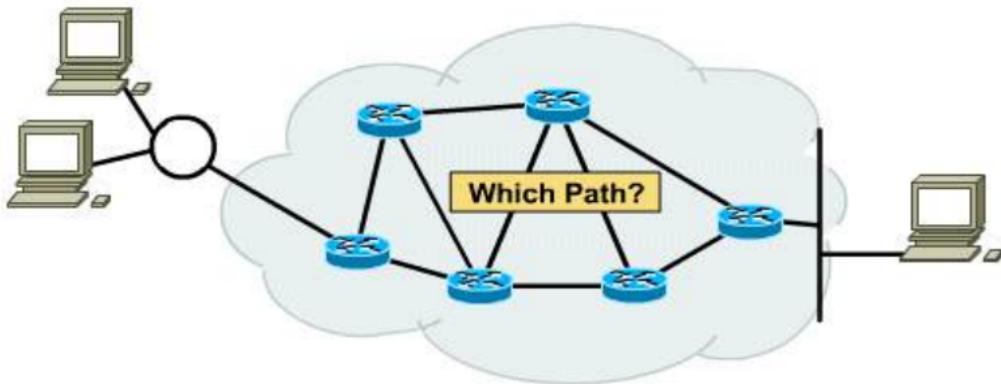
Contents

- Internet Routing
 - Concept of Autonomous System (AS)
 - Intra-AS and Inter-AS Routing
- Intra-AS Routing
 - Distance Vector Routing
 - e.g. Routing Information Protocol (RIP)
 - Link State Routing
 - e.g. Open Shortest Path First (OSPF)
- Inter-AS Routing
 - Path Vector Routing
 - e.g. Border Gateway Protocol (BGP)



Router

A **router** is a device used to interconnect networks, and to forward packets by **examining the destination address** in the **IP header** of each packet.

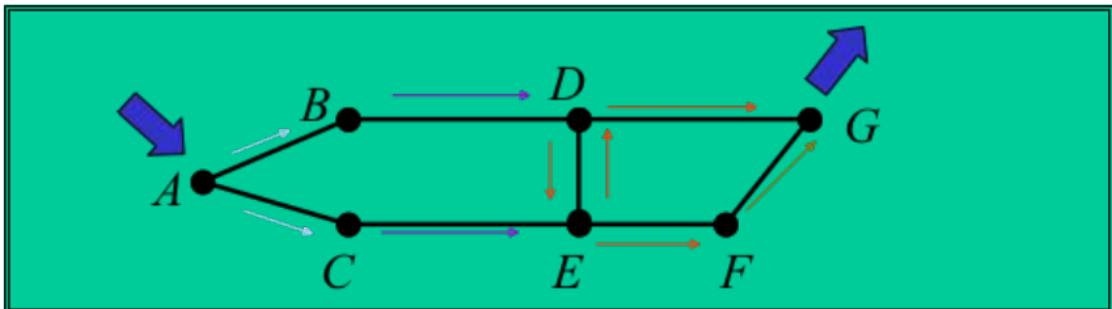


How does a router decide which path to forward?

>> **Routing Protocol**, used to initialize/update **routing table** so that a route/path can be determined.

Routing: Flooding

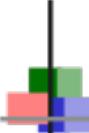
- A straight forward way of routing is by **flooding**.
- When a node receives a packet, it will forward the packet to all other links except the incoming link. The packet will be labeled with a unique identifier
- Should the same packet return, the packet will be discarded



Packets transmitted using flooding is 9

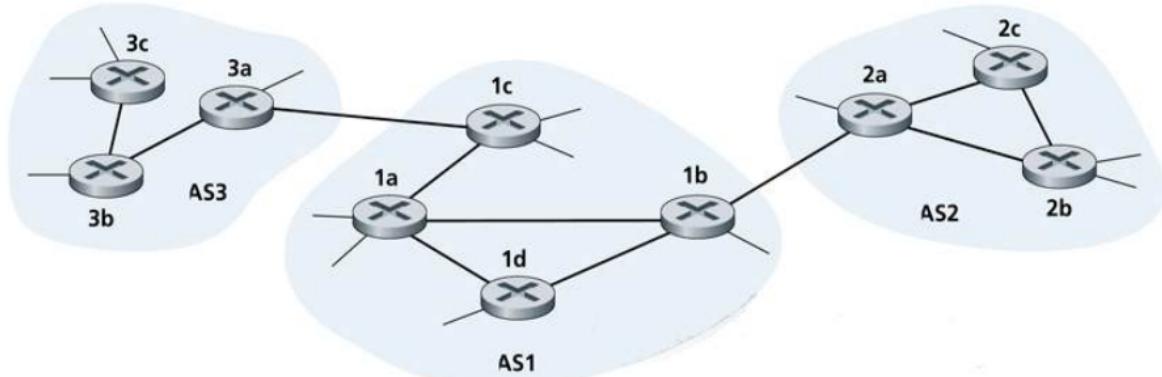
Routing: Flooding

- **Advantages:**
 - A packet will always get through if one or more path exists (very robust)
- **Disadvantages:**
 - Very wasteful of bandwidth, may cause serious congestion, hence not used in the Internet
- **Applications:**
 - Military applications (routers may be destroyed anytime)
 - Ad hoc wireless networks (nodes may be turned off or moved away anytime)



Routing in the Internet

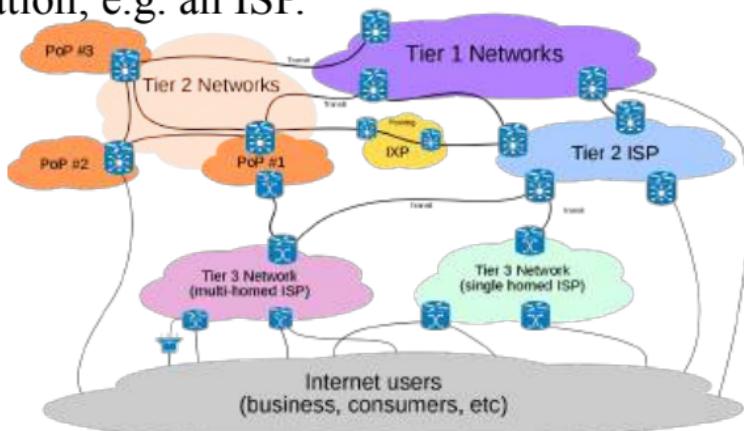
For routing purpose, **Internet** is divided into **Autonomous Systems (AS)**. An AS is a **group of routers** under the authority of a single administration; e.g. an ISP.



Each AS is uniquely identified by a 2-byte (0 - 65,535) or 4-byte (65,536 - 4,294,967,295) **ASN (AS number)**, which is assigned by IANA; e.g. ASN of NTU: AS9419.

Routing in the Internet

For routing purpose, **Internet** is divided into **Autonomous Systems (AS)**. An AS is a **group of routers** under the authority of a single administration; e.g. an ISP.



Each AS is uniquely identified by a 2-byte (0 - 65,535) or 4-byte (65,536 - 4,294,967,295) **ASN (AS number)**, which is assigned by IANA; e.g. ASN of NTU: AS9419.

IP Tools[Decimal IP Calculator](#)**ASN Information**[CIDR/Netmask](#)[What's your IP](#)[IP Geo-location Lookup](#)[IPWHOIS Lookup](#)

ASN Lookup & Information

[Email](#) [Share](#)

The ASN Information tool provides complete autonomous system (AS) information.

Autonomous Systems are routable networks within the public Internet, administered by the local RIRs and assigned to owners of networks. The ASN Information tool displays information about an IP address's Autonomous System Number (ASN) such as: IP owner, registration date, issuing registrar and the max range of the AS with total IPs.

Enter an AS number, IP address, or a Company name.

Related Tools: [CIDR/Netmask](#) [What's your IP](#) [Decimal IP Calculator](#)

AS9419

Country: SG

Registration Date: 1998-10-01

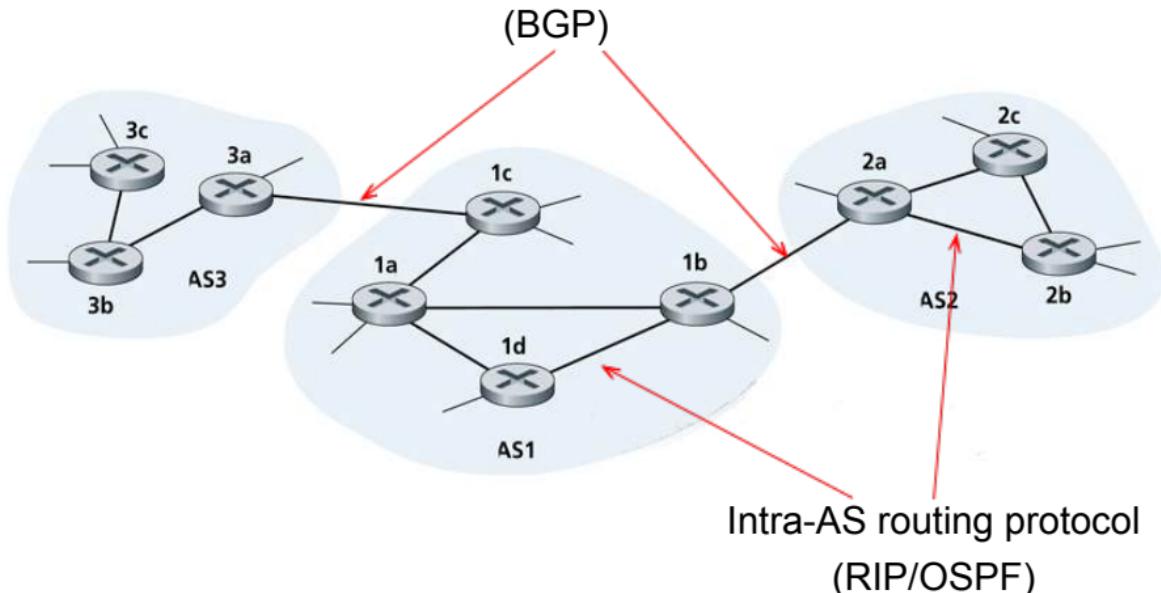
Registrar: apnic

Owner: NTU-AS-AP Nanyang Technological University, SG

Intra-AS and Inter-AS Routing

In practice, routing in Internet is done in a hierarchical manner, which includes **intra-AS** and **inter-AS** routings.

Inter-AS routing protocol



Intra-AS routing protocol
(RIP/OSPF)



Intra-AS and Inter-AS Routing

In practice, routing in Internet is done in a hierarchical manner, which includes **intra-AS** and **inter-AS** routings.

Intra-AS Routing:

- Routing within an AS
- Protocols for Intra-AS routing are also called **Interior Gateway Protocols (IGPs)**
- Different AS can choose to run their preferred protocols
- e.g. intra-AS routers: 1a, 1b, 1c, 1d in AS1

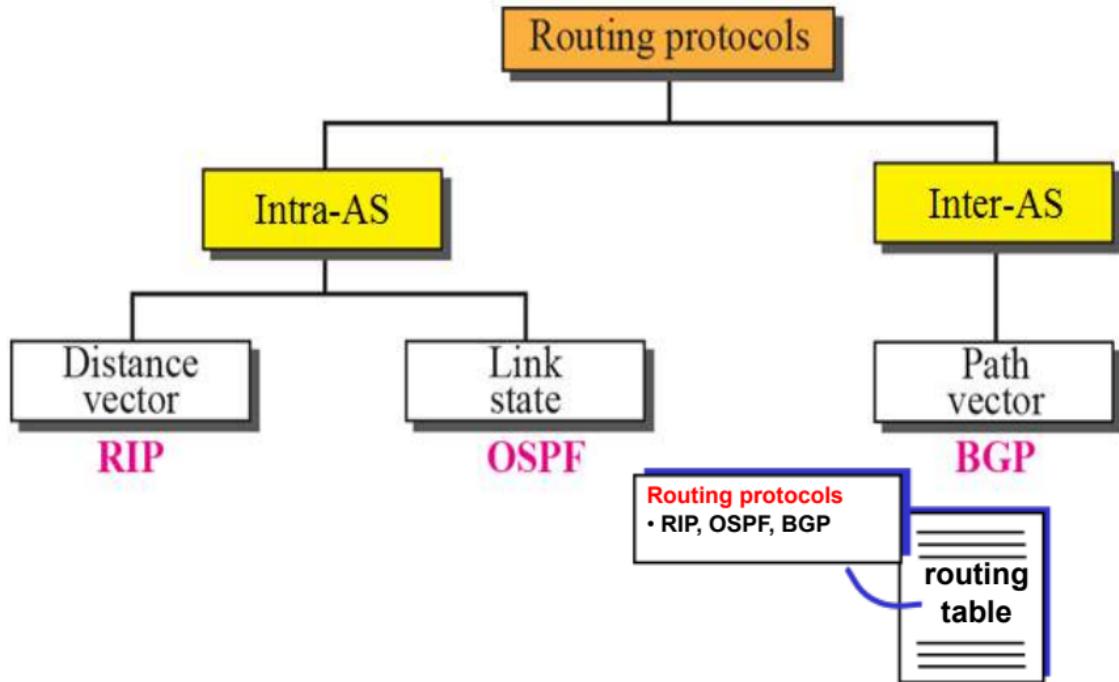
Inter-AS Routing:

- Routing between AS
- Protocols for Inter-AS routing are also called **Exterior Gateway Protocols (EGPs)**
- All AS must run the same standard protocol
- e.g. inter-AS routers: 1b, 2a, 1c, 3a

Why different Intra- and Inter-AS Routing?

- **Policy:**
 - Inter-AS: different admin wants control over how its traffic are forwarded, who routes through its network
 - Intra-AS: single admin, so no policy decision needed
- **Performance:**
 - Intra-AS: can focus on performance
 - Inter-AS: policy may dominate over performance
- **Scale:**
 - Internet is too large to be treated as a single routing domain

Intra-AS and Inter-AS Routing Protocols



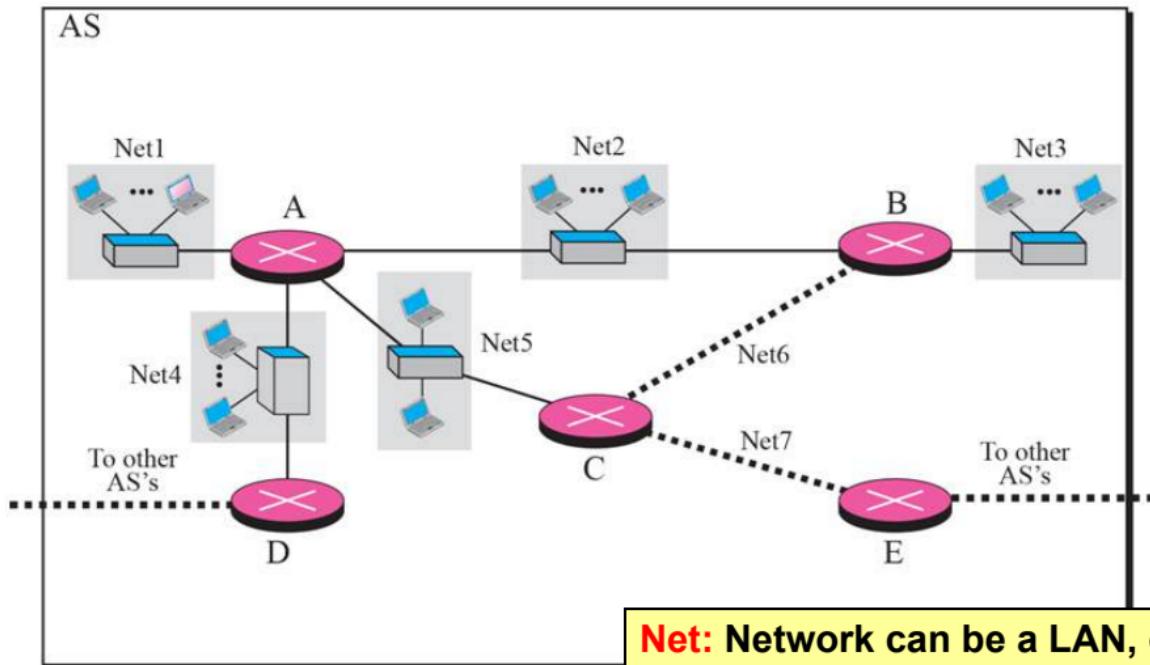
Intra-AS Routing: Distance Vector Routing

Distance Vector routing also known as
“Bellman-Ford” or “old ARPANET” routing.

Essentially, consists of 3 main stages:

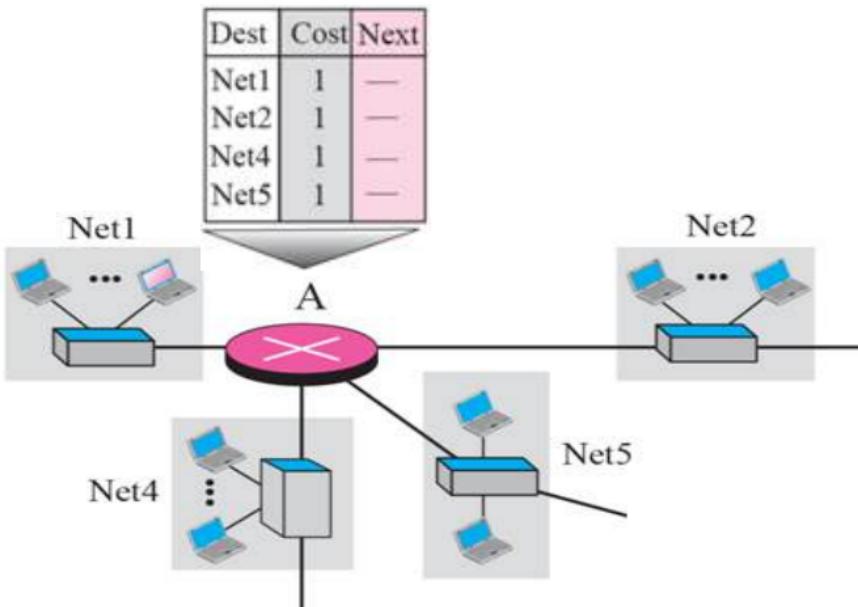
- Discover neighbors by multicasting request
- Exchange distance vectors (routing information) with immediate neighbors only
 - Response to request
 - Periodic updates (typically 30s interval)
 - Triggered updates due to changes
- Compute shortest-path routes (using Bellman-Ford algorithm)

An example of Distance Vector Routing



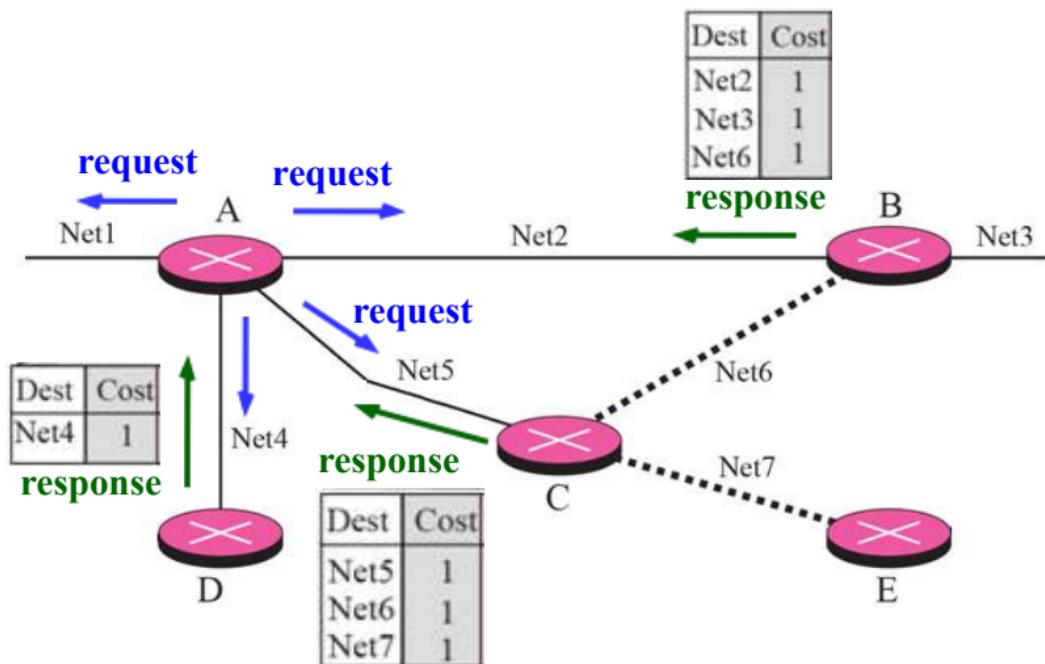
Net: Network can be a LAN, or
a point-to-point link

Distance Vector Routing: Initially, a router only has its own configured routing table.

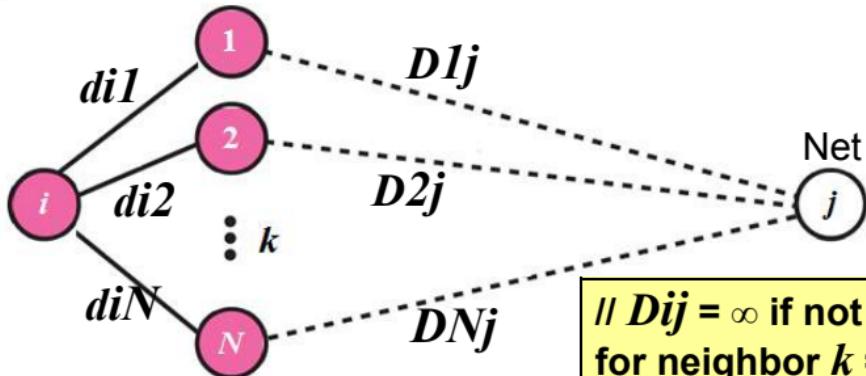


For simplicity but without loss of generality, assume each link has a cost of 1.

Distance Vector Routing: Discover Adjacent Neighbors and exchange distance vectors.



Distance Vector Routing: Computing Shortest-Path using Bellman-Ford Algorithm



```
//  $Dij = \infty$  if not reachable initially  
for neighbor  $k = 1$  to  $N$  {  
    if ( $dik + Dkj < Dij$ ) {  
         $Dij = dik + Dkj$  ;  
    }  
}
```

d_{ik} = cost of going directly from node i to adjacent neighbor node k ;

D_{ij} = least total cost of going from node i to destination j

Distance Vector Routing: Computing Shortest-Path using Bellman-Ford Algorithm

from B:

Dest	Cost
Net2	1
Net3	1
Net6	1

from C:

Dest	Cost
Net5	1
Net6	1
Net7	1

from D:

Dest	Cost
Net4	1

Dest	Cost	Next
Net1	1	—
Net2	1	—
Net3	2	B
Net4	1	—
Net5	1	—
Net6	2	C
Net7	2	C

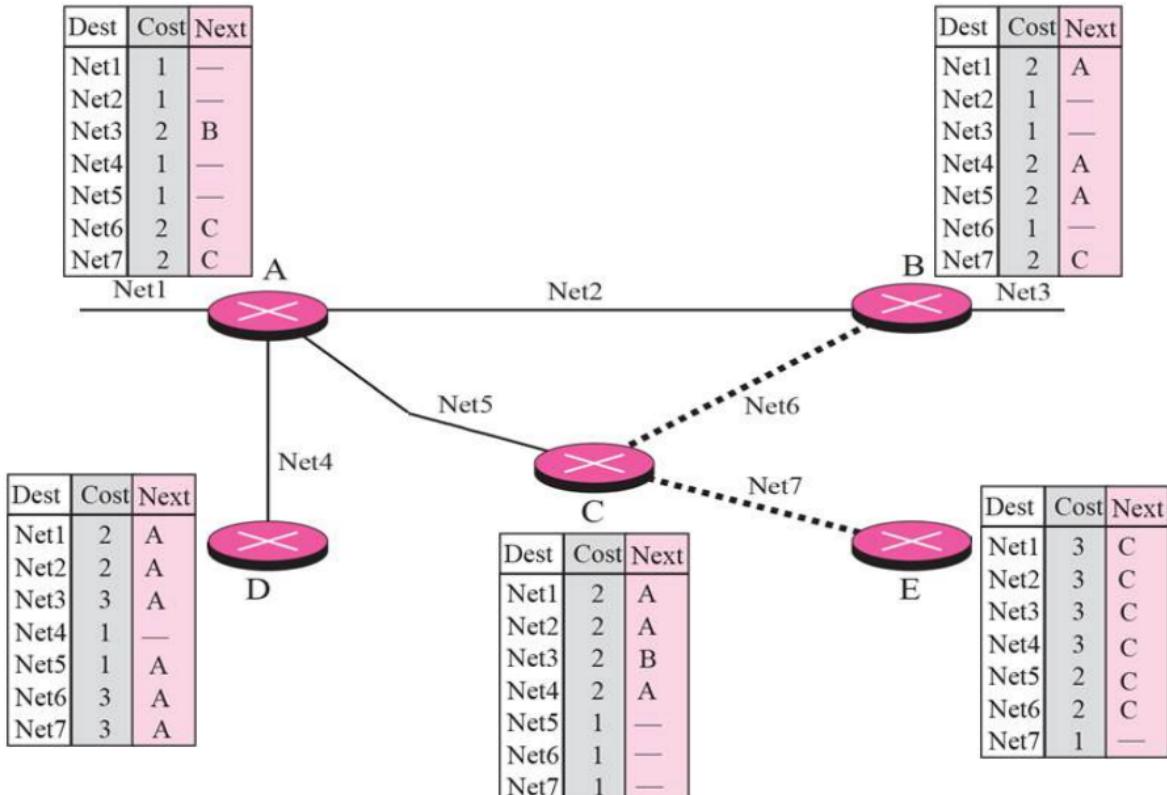
original
routing
table of A

Dest	Cost	Next
Net1	1	—
Net2	1	—
Net3	2	B
Net4	1	—
Net5	1	—
Net6	2	C
Net7	2	C

updated
routing
table of A

Assume Cost of 1 between node

Distance Vector Routing: Resulting routing tables at each router after convergence.



Quote about Distance vector

Not as fast as
Link state

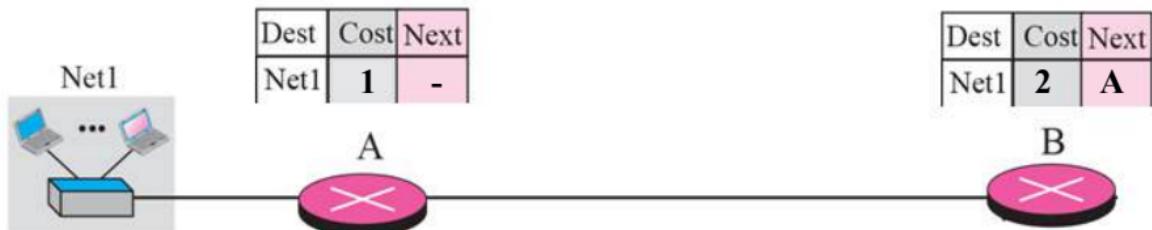
- **GOOD news travel fast,**

Caused by
count to Infinity

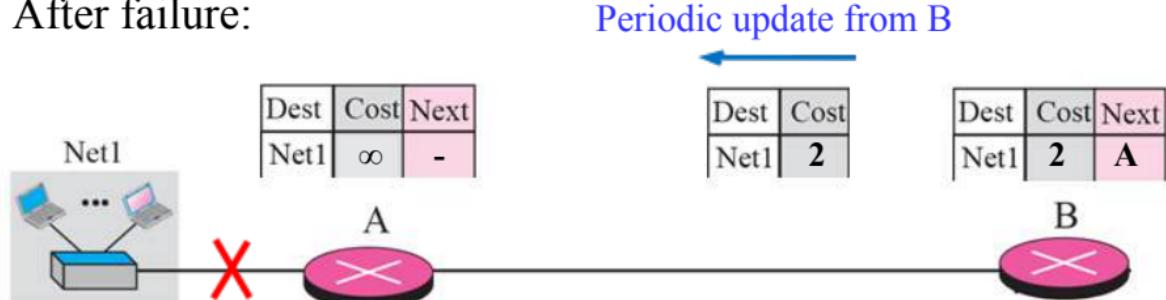
- **BAD news travel slowly**

Distance Vector Routing: Count-to-Infinity problem

Before failure:

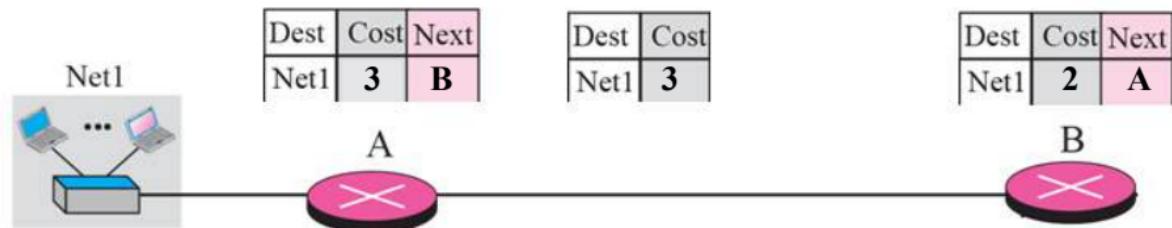


After failure:

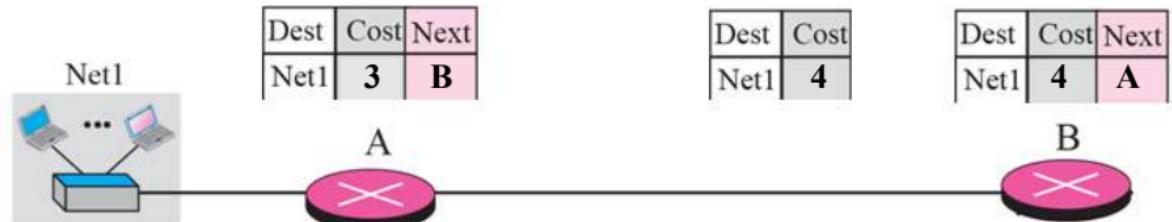


Distance Vector Routing: Count-to-Infinity problem

After A received update from B:



After B received update from A:



Distance Vector Routing: Count-to-Infinity problem

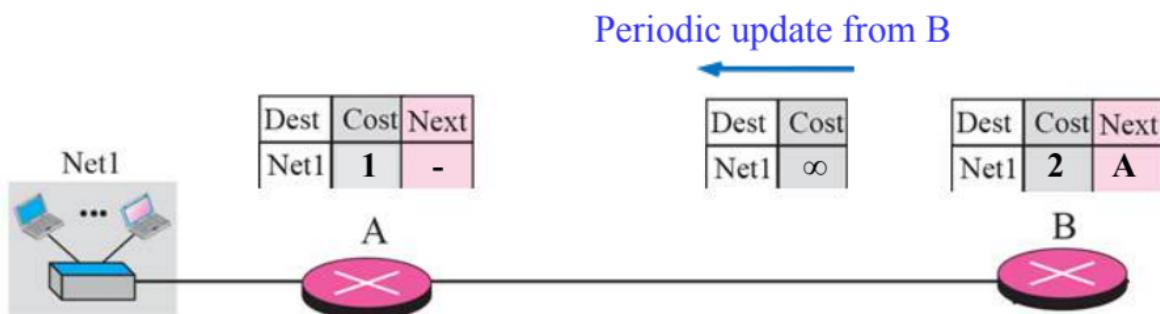
⋮

Eventually:



Distance Vector Routing: Solving Count-to-Infinity Problem - Split Horizon with Poisoned Reverse (RFC 2453)

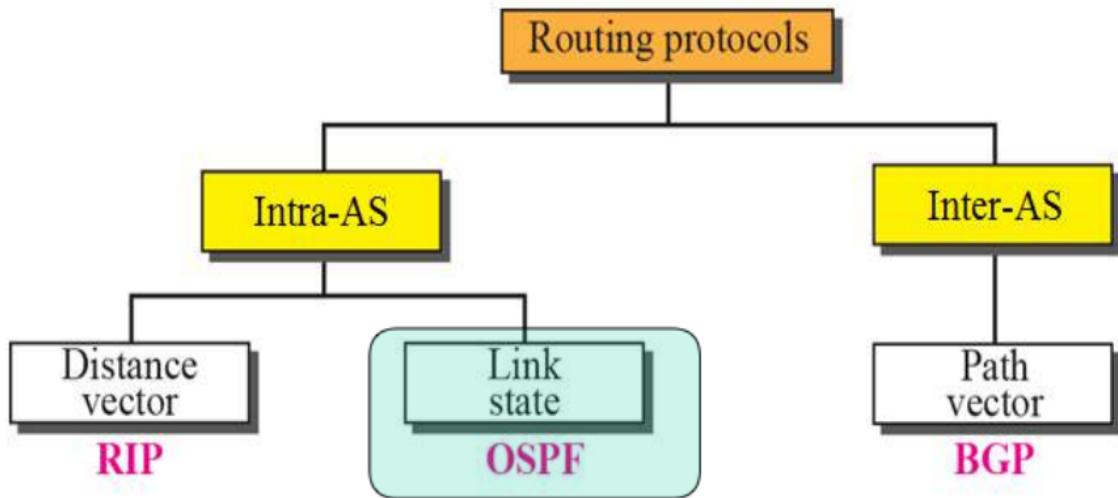
If B gets to Net1 via A, then its update to A should indicate that Net1 is unreachable.



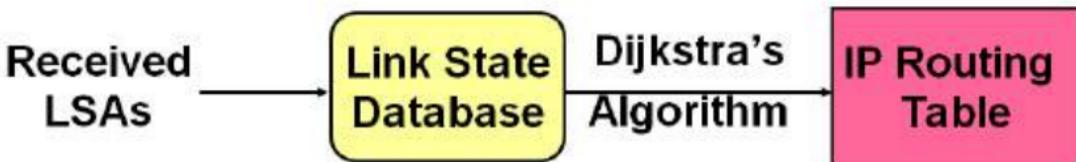
Intra-AS Distance Vector Routing: Routing Information Protocol

- RIP uses **Distance Vector** algorithm, cost is simply based on the number of hops
- Allows maximum 15 hops, 16 indicates ∞
- Routing information exchanged every 30 sec via Response Message
- If no advertisement heard after 180 seconds neighbor/link declared dead
- RIP related RFC Documentations:
 - [RFC 1058, 1387, 1388, 1723 \(RIP version 2\)](#)

Intra-AS and Inter-AS Routing Protocols



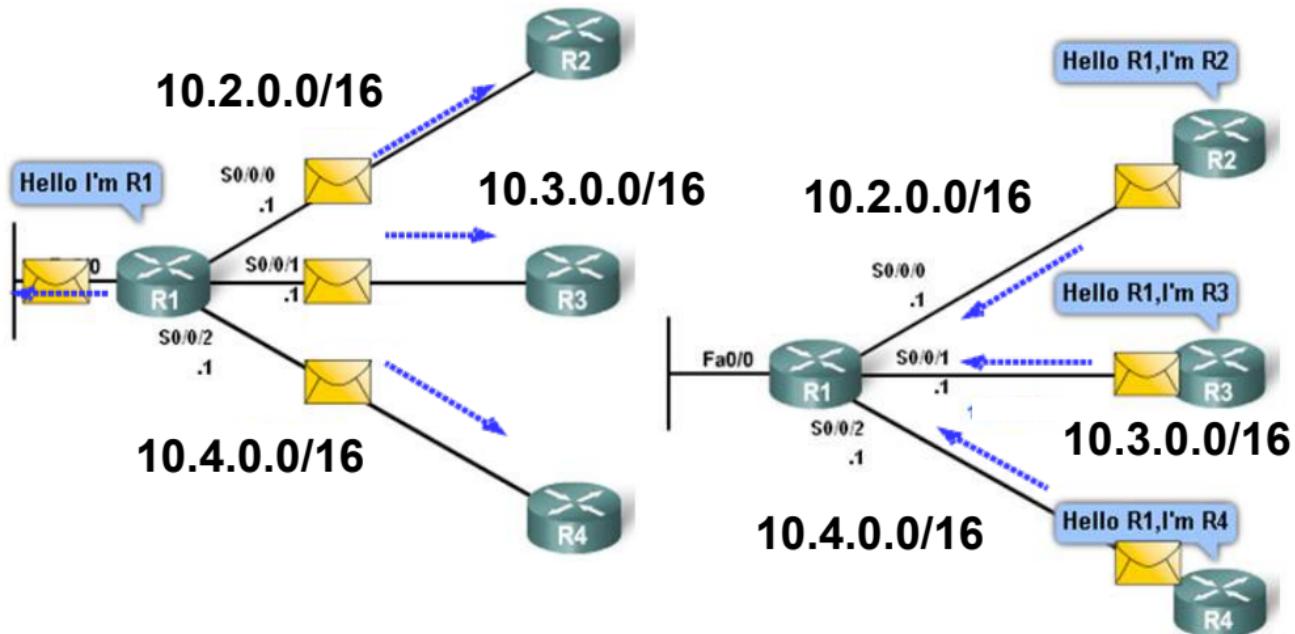
Intra-AS Routing: Link State Routing



Consists of Five stages:

- Discover Neighbors by multicasting Hello
- Construct Link State Advertisement Packet (LSA/LSP)
- Flood LSA/LSP to ALL Routers
 - During initial start-up
 - When there is a change in topology
- Construct Link State Database
- Compute Shortest-Path Routes (using Dijkstra's algorithm)

Link State Routing: Discover Neighbors

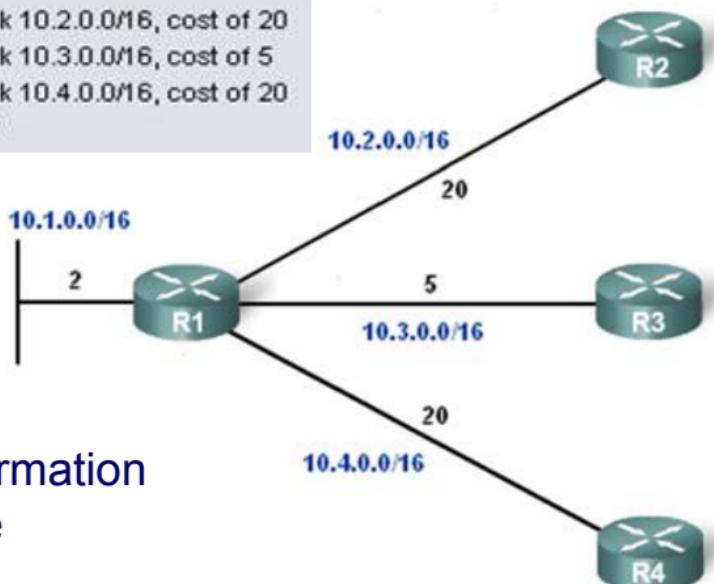


- A transmits HELLO packet on each of its links
- A's neighbors identify themselves to A

Link State Routing: Construct Link State Advertisement Packet (LSA/LSP)

R1 Link-State

- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

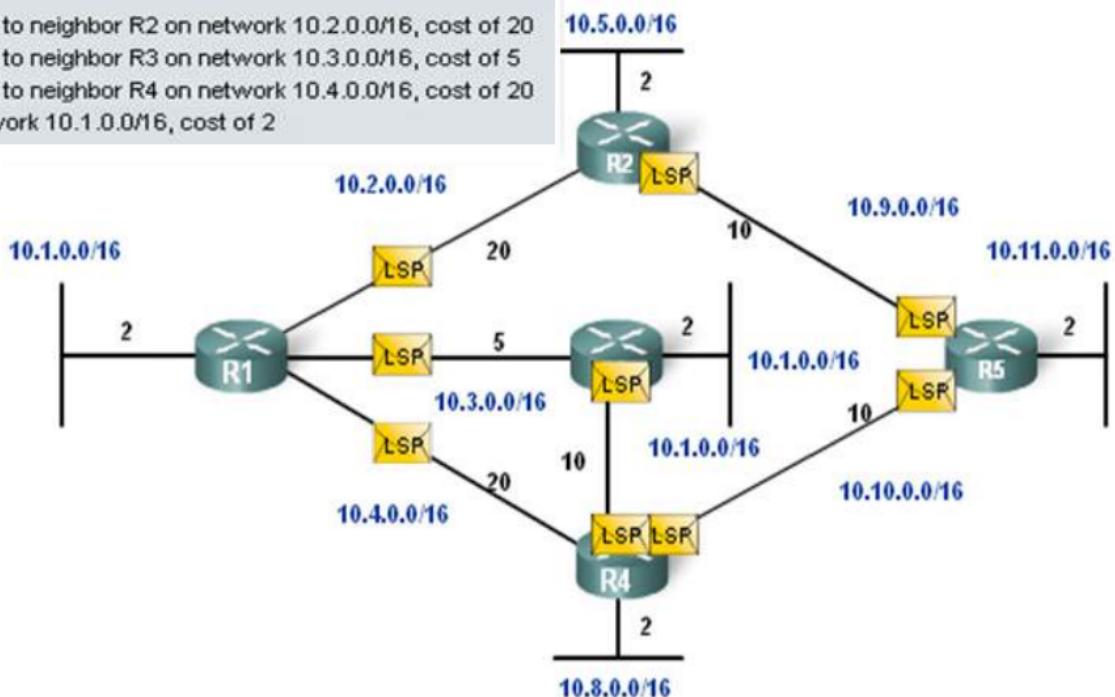


- LSA/LSP contains information of itself and immediate neighbors only.

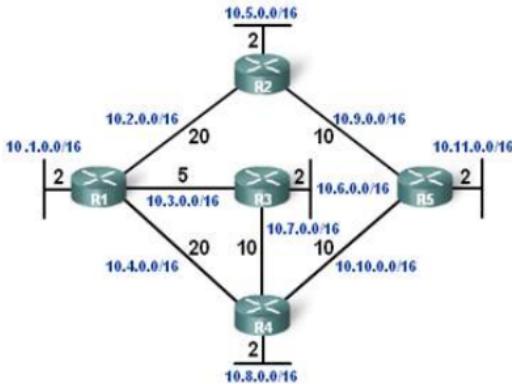
Link State Routing: Flood LSP to ALL routers

R1 Link-State

- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2



Link State Routing: Build Link State Database



Each router builds its own **link state database** to have a complete topology of the whole network.

R1 Link-State Database

R1 Link-states:

- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

LSPs from R2:

- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2

LSPs from R3:

- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2

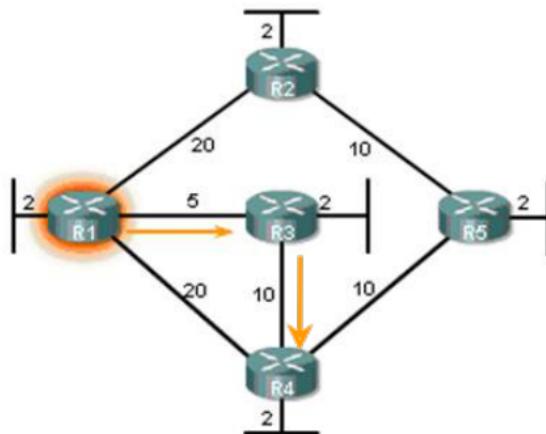
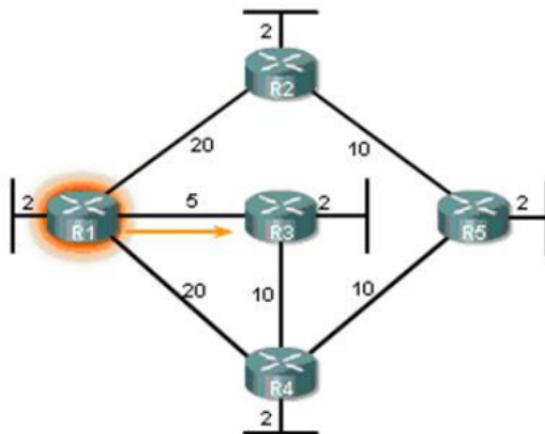
LSPs from R4:

- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2

LSPs from R5:

- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2

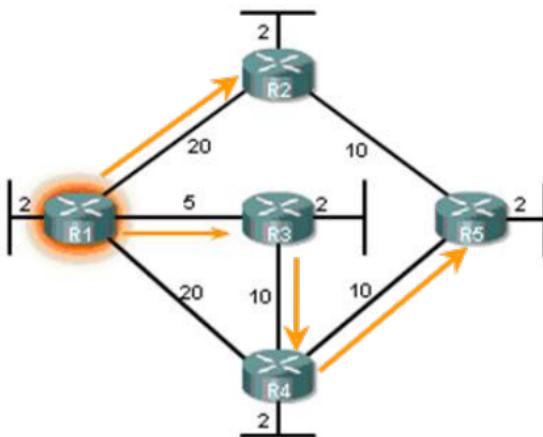
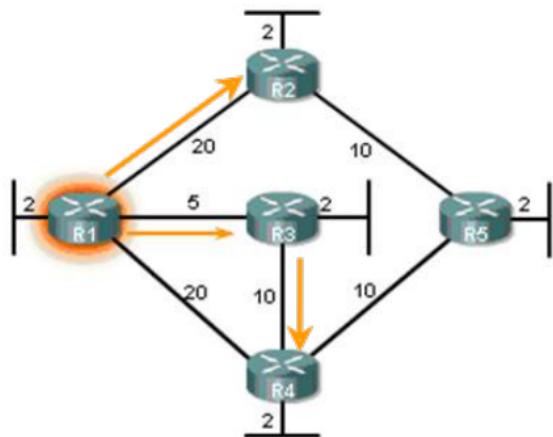
Link State Routing: Compute **shortest-path** routes using Dijkstra's Algorithm



Iteration	D_{12}	D_{13}	D_{14}	D_{15}
{1}	20	5✓	20	∞

Iteration	D_{12}	D_{13}	D_{14}	D_{15}
{1}	20	5✓	20	∞
{1,3}	20	5	15✓	∞

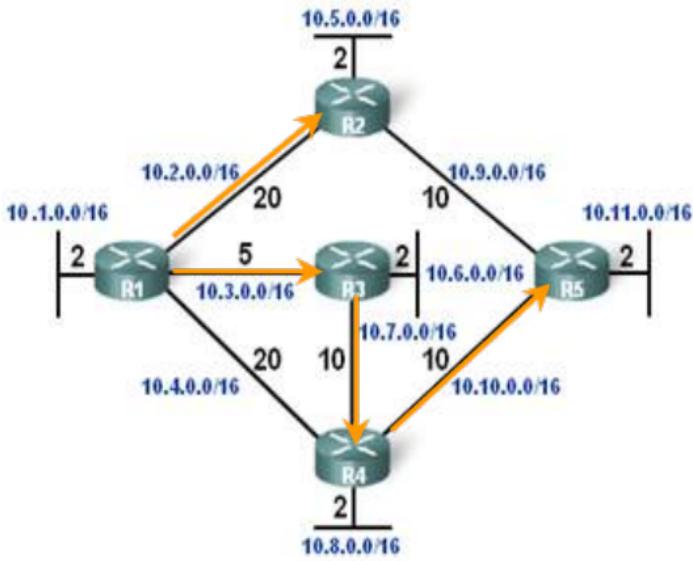
Link State Routing: Compute **shortest-path** routes using Dijkstra's Algorithm



Iteration	D_{12}	D_{13}	D_{14}	D_{15}
{1}	20	5✓	20	∞
{1,3}	20	5	15✓	∞
{1,3,4}	20✓	5	15	25

Iteration	D_{12}	D_{13}	D_{14}	D_{15}
{1}	20	5✓	20	∞
{1,3}	20	5	15✓	∞
{1,3,4}	20✓	5	15	25
{1,3,4,2}	20	5	15	25✓

Link State Routing: Update routing table



R1 Routing Table

Directly Connected Networks

- 10.1.0.0/16 Directly Connected Network
- 10.2.0.0/16 Directly Connected Network
- 10.3.0.0/16 Directly Connected Network
- 10.4.0.0/16 Directly Connected Network

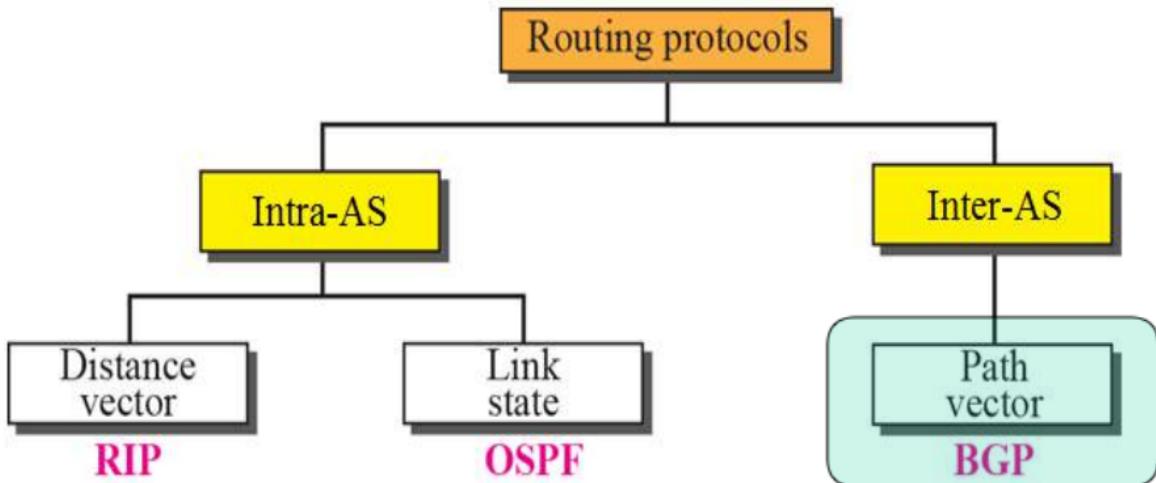
Remote Networks

- 10.5.0.0/16 via R2, cost = 22
- 10.6.0.0/16 via R3, cost = 7
- 10.7.0.0/16 via R3, cost = 15
- 10.8.0.0/16 via R3, cost = 17
- 10.9.0.0/16 via R2, cost = 30
- 10.10.0.0/16 via R3, cost = 25
- 10.11.0.0/16 via R3, cost = 27

Intra-AS Link State Routing: Open Shortest Path First

- “Open”: publicly available
- Uses Link State algorithm
 - LSA packet dissemination
 - Topology map at each node
 - Route computation using Dijkstra’s algorithm
- Advertisements disseminated to entire AS (via flooding)
- OSPF related RFC documentations:
RFC 1131, 1247, 1583 (OSPF version 2)

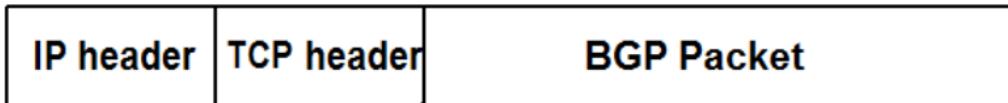
Intra-AS and Inter-AS Routing Protocols



Inter-AS Path Vector Routing: Border Gateway Protocol (BGP)

Consists of 3 main stages:

- **Configure border router to know its neighbors**
- **Exchange path vectors (routing information) with neighbors**
- **Select path based on policy**
- BGP: the *de facto* standard, current ver 4 (RFC 4271)



port:
179

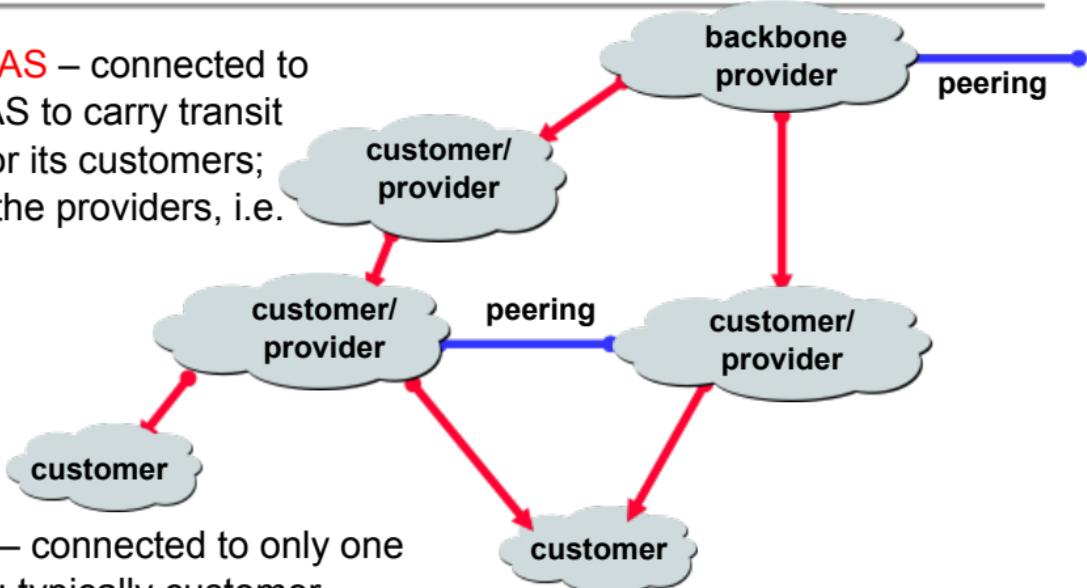
After configuration, BGP routers establish TCP connections with their neighbors to exchange routing information.

Protocol design principles

- **Scalable.** Backbone AS must be able to find the destination.
- **Loop free.**
- **Autonomy of AS routing policy**

Broadly, AS can be classified into **stub AS**, **multi-homed AS** or **transit AS**.

Transit AS – connected to other AS to carry transit traffic for its customers; mainly the providers, i.e. ISPs

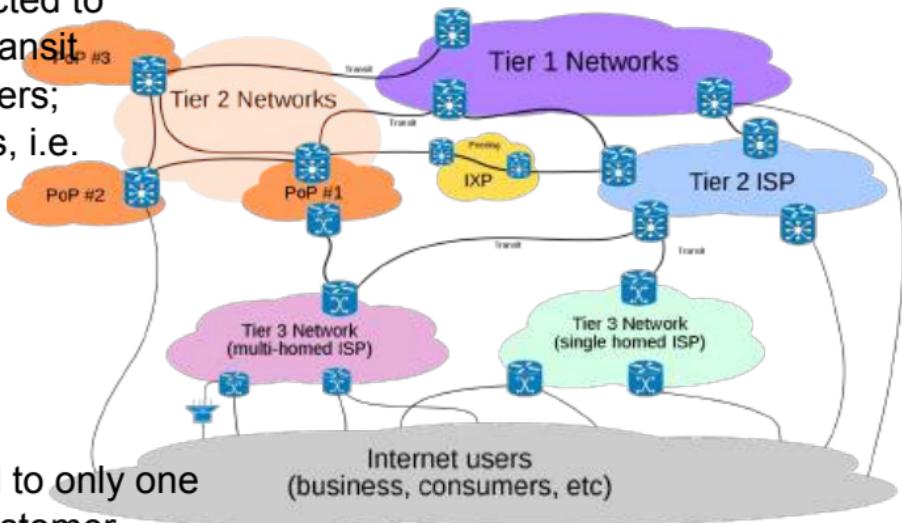


Stub AS – connected to only one other AS; typically customer connected to its provider. In fact, it's **not necessary for stub AS to run BGP** since it has only 1 path to its ISP.

Multi-homed AS – connected to more than one AS, but does not carry transit traffic; typically for customer requiring reliability

Broadly, AS can be classified into **stub AS**, **multi-homed AS** or **transit AS**.

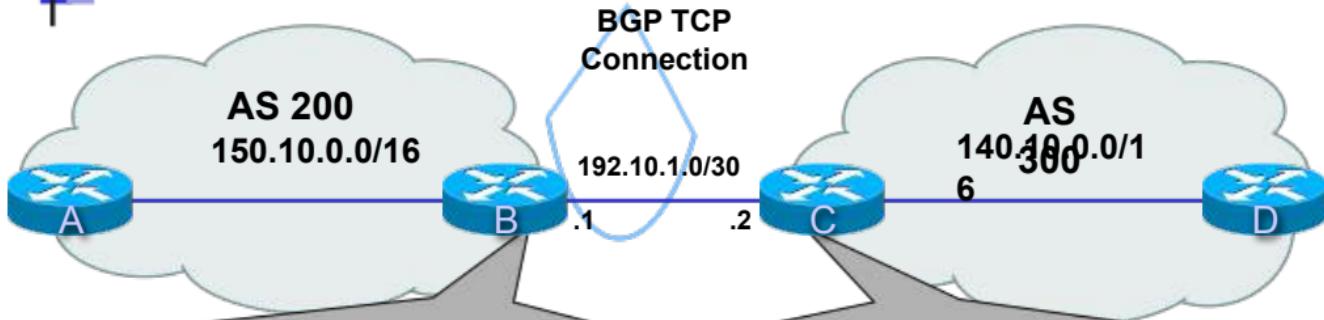
Transit AS – connected to other AS to carry transit traffic for its customers; mainly the providers, i.e. ISPs



Stub AS – connected to only one other AS; typically customer connected to its provider. In fact, it's **not necessary for stub AS to run BGP** since it has only 1 path to its ISP.

Multi-homed AS – connected to more than one AS, but does not carry transit traffic; typically for customer requiring reliability

Configuring BGP routers to know their neighbors (peers) to establish TCP connections.



```
interface Serial 0
ip address 192.10.1.1 255.255.255.252

router bgp 200
network 150.10.0.0 mask 255.255.0.0
neighbor 192.10.1.2 remote-as 300
```

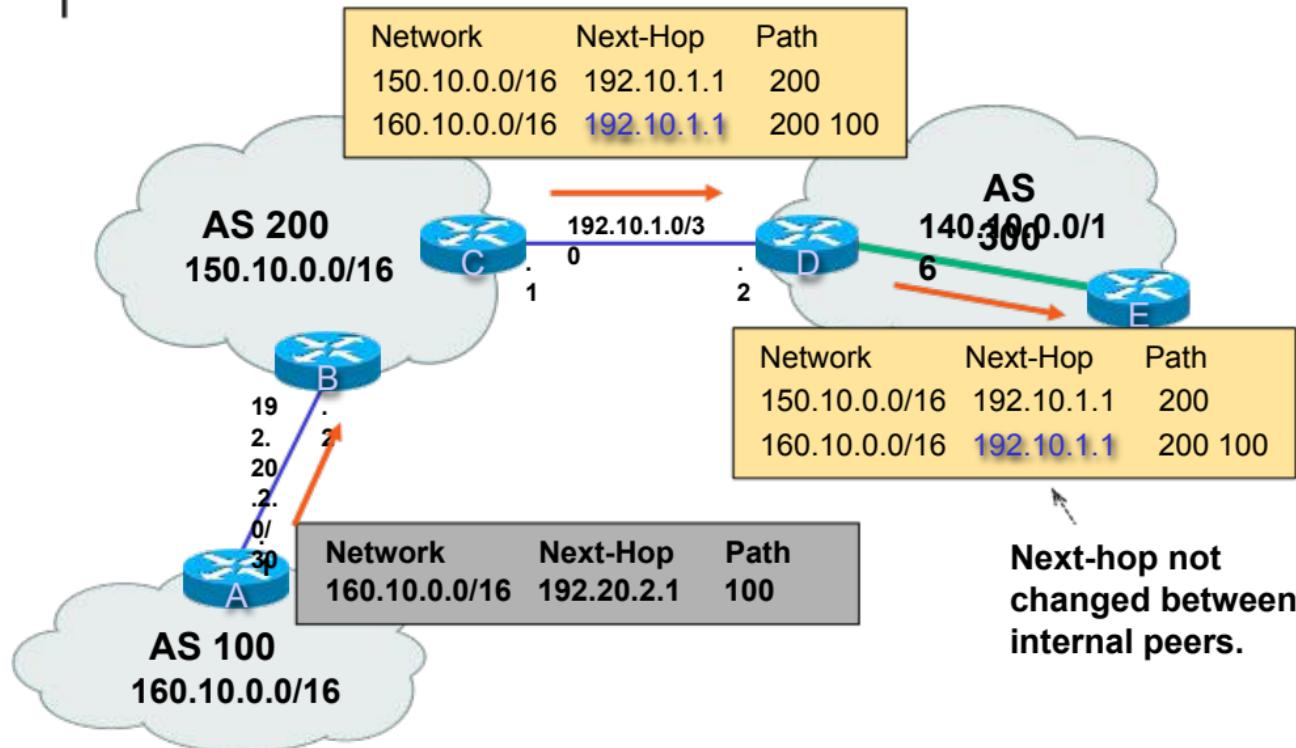
```
interface Serial 0
ip address 192.10.1.2 255.255.255.252

router bgp 300
network 140.10.0.0 mask 255.255.0.0
neighbor 192.10.1.1 remote-as 200
```

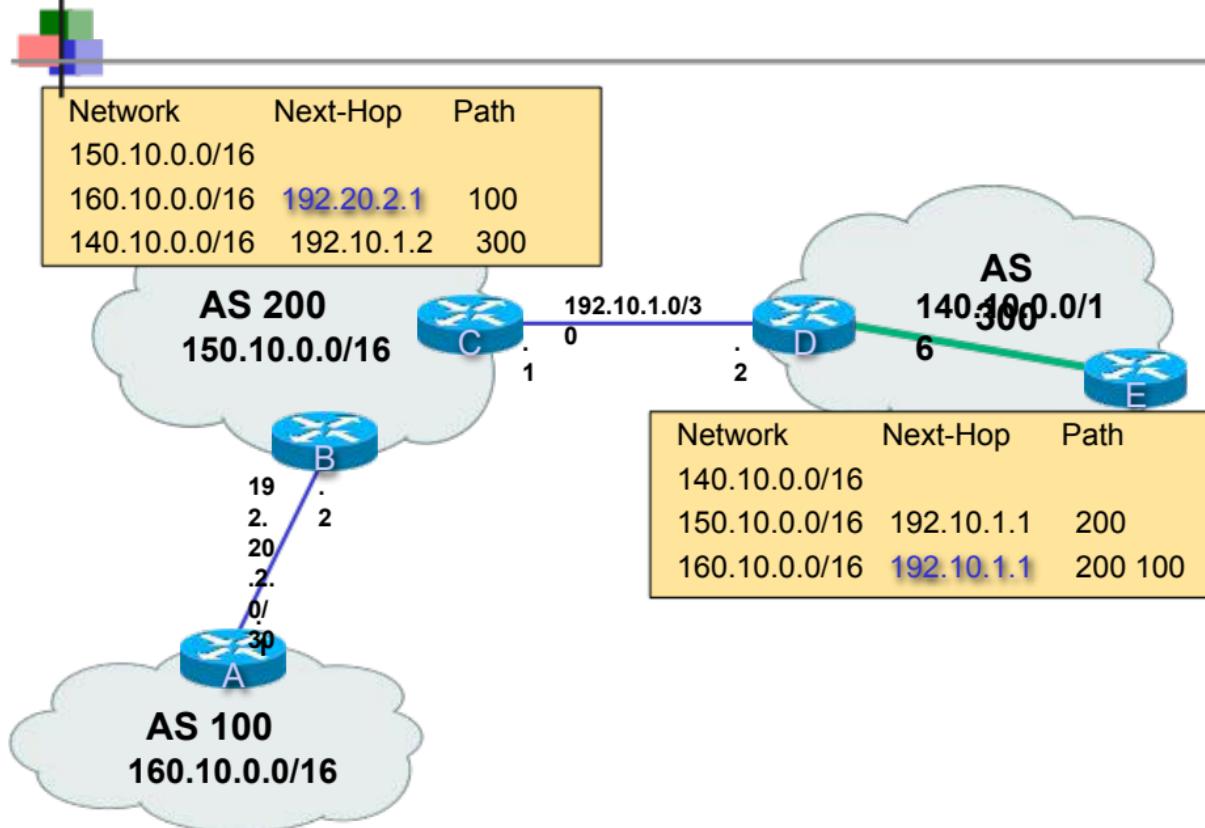
External peers (between different AS) are normally adjacent to each other and share a subnet.

Internal peers may be in any subnet within the same AS.

Peers exchange routing information containing complete AS Path (to avoid loop problem)



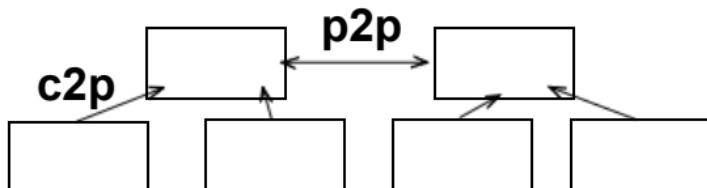
AS Table content

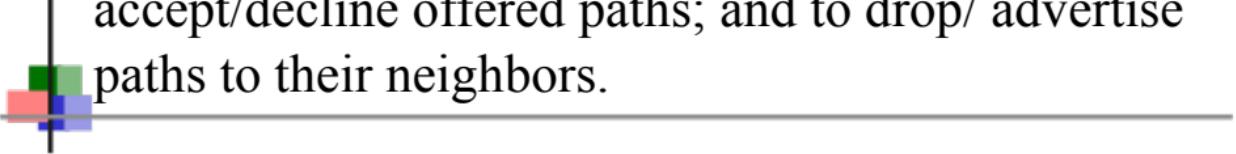


There is also the **commercial relationship** between neighbouring ASes.

ASes relationship:

- **Customer-Provider:** Customer pays the provider to send and receive traffic
 - e.g. NTU and SingTel
- **Peer-to-peer:** Two Ases agree to transit each other traffic.
 - eg. SingAREN and AARnet





Based on **policy**, BGP routers can decide to accept/decline offered paths; and to drop/ advertise paths to their neighbors.

Path selection (policy-based):

- **Import policy:** may or may not select path offered
 - e.g. cost, business relationships, don't route through competitors, loop prevention reasons.
- **Export policy:** can filter routes you don't want to tell neighbors
 - eg. don't want to route traffic to Z □ don't advertise any route to Z

ISP interconnection fees will not be regulated, says IDA

Fees smaller ISPs pay StarHub and Singtel not hurting consumers, says watchdog

Irene Tham

Tech Editor

itham@sph.com.sg

The local authorities will not interfere with interconnection fees that smaller Internet service providers (ISPs) pay to the two major telcos here, saying that consumers have not been harmed.

The Infocomm Development Authority (IDA) issued its decision earlier this week after concluding its two-month public consultation held last year.

Smaller ISPs like M1, ViewQwest, MyRepublic and SuperInternet pay Singtel and StarHub fees for the shortest path - and fastest access - for their subscribers and hosted websites. Some of them said it was two to three times the price of international links charged by international telcos.

Date: 27 August 2016

Source: The Straits Times
2-3-0



Industry players say interconnection fees have gone up rapidly, consumers are not pleased by the smaller savings from broadband from calling interconnection services. ST PHOTO: ISRAEL

ISP interconnection fees will not be regulated, says IDA

Fees smaller ISPs pay StarHub and Singtel not hurting consumers, says watchdog

Ameniethu
Twe & Falmer

StarHub's justification will not be accepted because it does not offer interconnecting at the same speeds as provider. Telecoms service providers (TSPs) pay for the no fee interconnection services, while smaller ISPs pay for the interconnection fees.

The Communications Development Authority (CDA) has decided to accept StarHub's fees this month after evaluating its new market position, said Ameniethu.

StarHub CEO Saseen, Naveen Oberoi, Shyamprakash, and Rajesh Chakrabarty, managing director and executive director for the division, were at the CDA yesterday.

They are the first two telecommunications operators to receive the go-ahead to increase their interconnection fees.

Forced interconnection fees of US\$10 million a year were to have been phased out under the Telecommunications Act of 1996. The CDA wanted the EMA to maintain "peering" – or the free exchange of traffic – among local ISPs without a premium in charging fees.

But the CDA has given StarHub "full" peering allowances, as stipulated in the telecommunications act.

The Singaporean telecoms commission also says the dual-bandwidth scheme will not affect smaller ISPs' revenue.

In an earlier decision, the CDA said to have been based on evidence of the cost savings from the new fees, the smaller ISPs' costs had risen by 10 per cent.

"Smaller ISPs are not prevented from negotiating content on the terms and conditions of interconnection fees, even if they have multiple ISP partners," said Ameniethu, adding that the CDA can take such measures.

StarHub's proposal and that from Singtel will be reviewed by the end of next month before the CDA makes its final decision.

For a full explanation, let me introduce StarHub again. Two weeks ago, I wrote about how StarHub and Singtel are likely going to merge like Google and Intel did, which have a different story with them.

I'm afraid that the value of StarHub's data is not as good as the value of Singtel's data, so it's an acquisition.

Singtel's business of fibres today is more concentrated in the north, while StarHub's is more concentrated in the south. So, if StarHub's fibre network is merged with Singtel's, it will be more concentrated in the south. This gives Singtel a better position in the south.

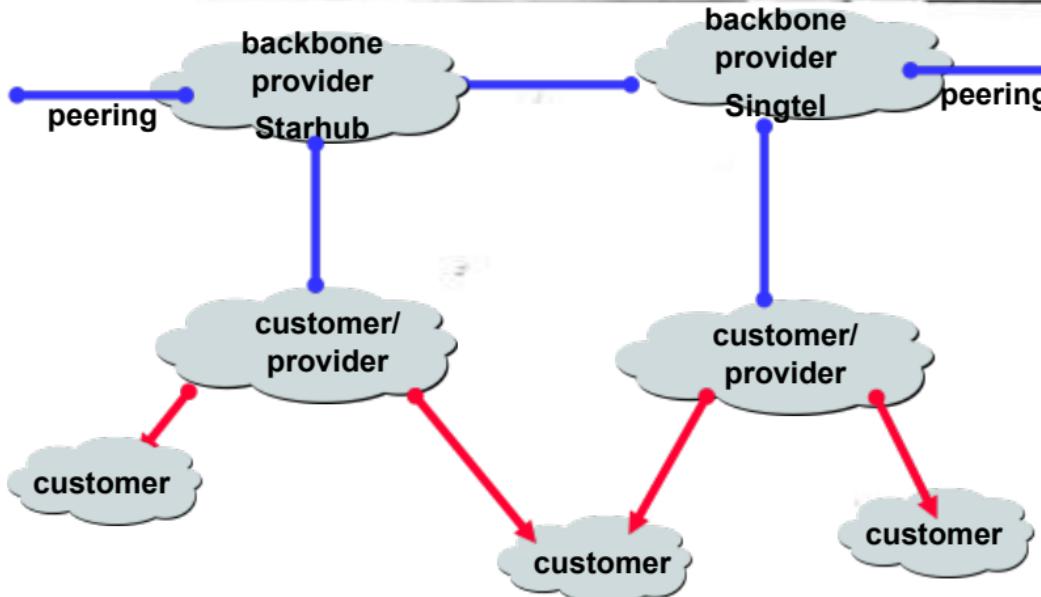
The CDA also stated that the companies are not being regulated by the government. On the contrary, they have been freed from having to follow regulations.

An initial high-speed fibre optic transmission plan was given for 2009, which included 100,000 km of fibre and many segments of the 200,000-kilometre network planned – 100,000 kilometres – could be built by 2010.

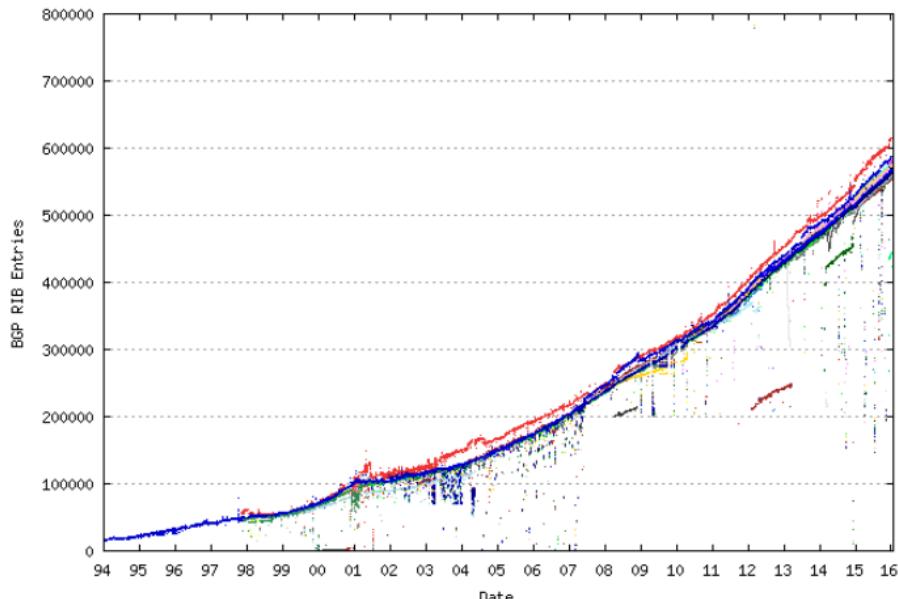
Singtel and StarHub supported the CDA's decision.

StarHub's fibre network is one of the largest in the world, and it is already competitive and can deliver higher speed connection.

It's a good news for consumers.

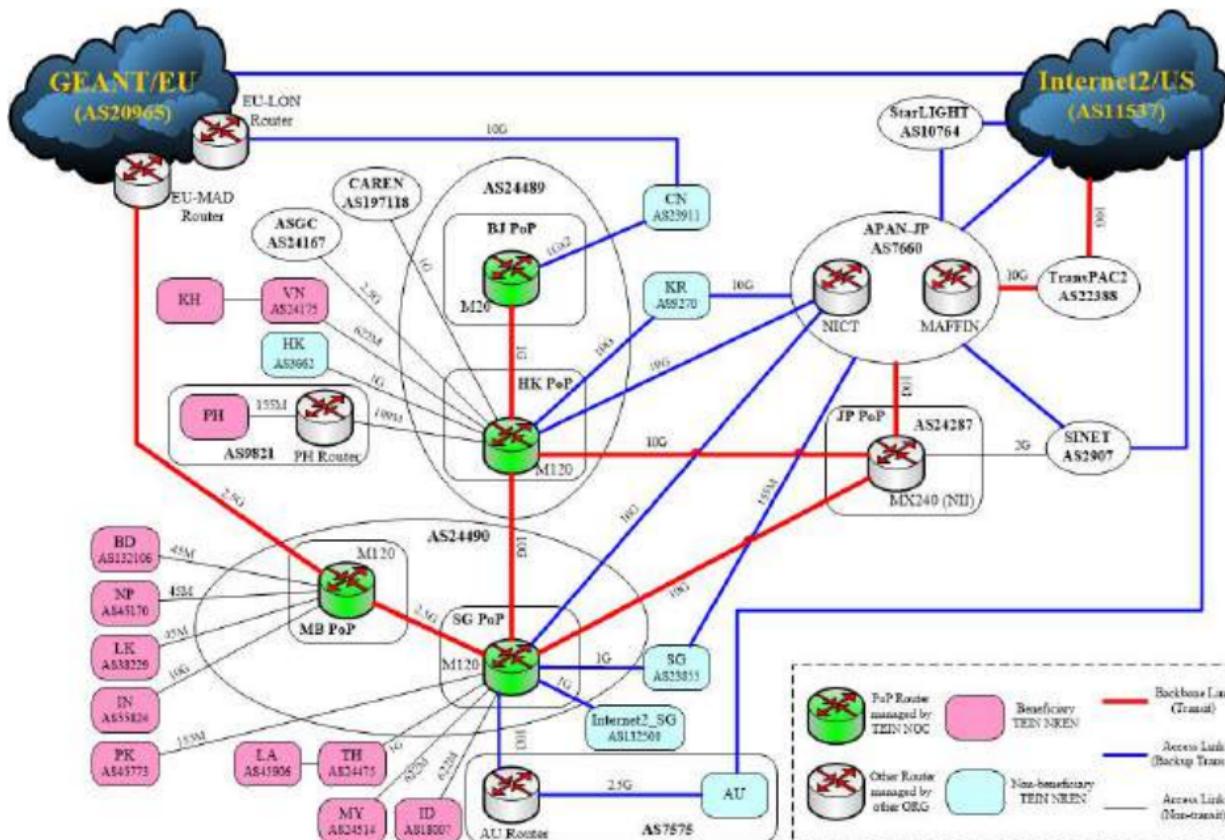


BGP routing table from 1994 - present

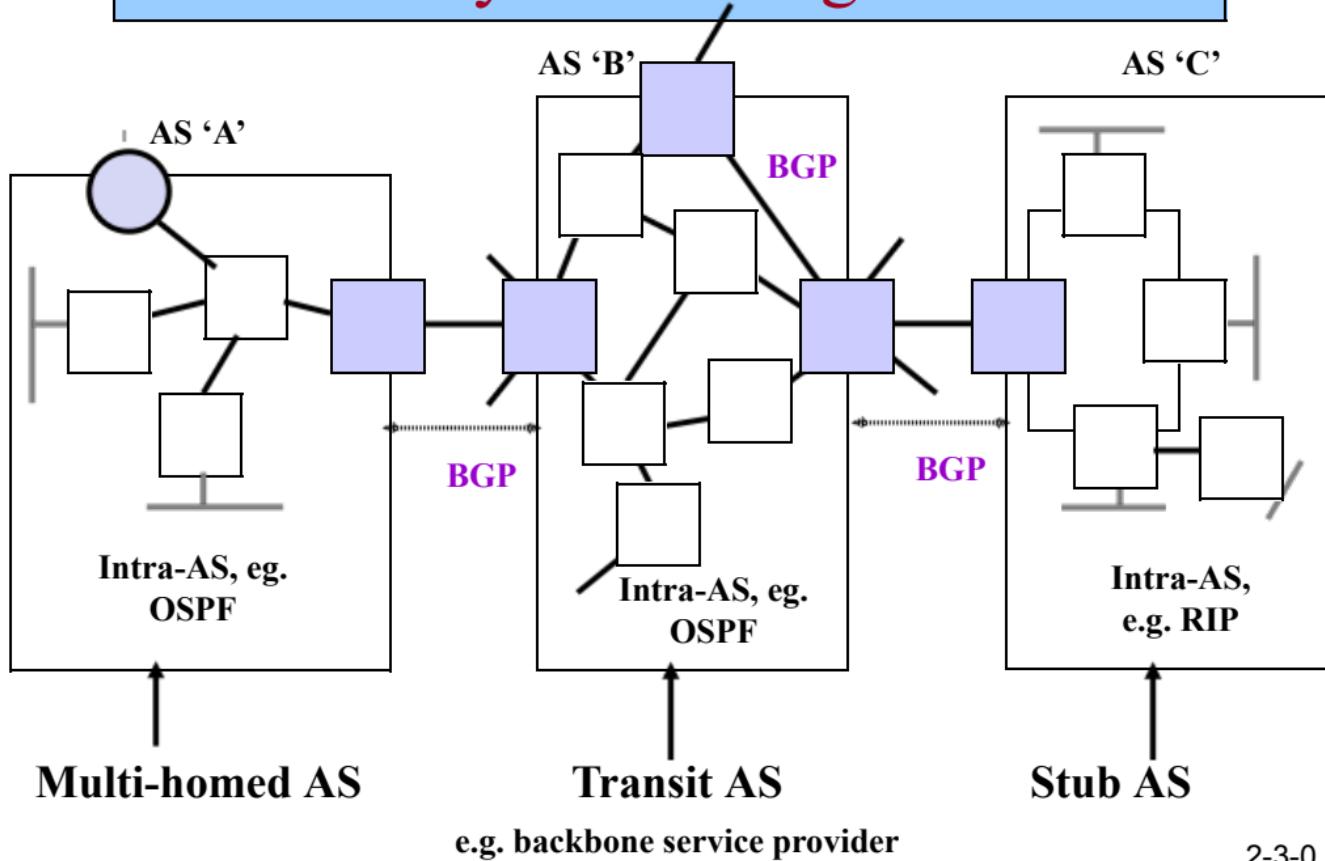


Source : <http://bgp.potaroo.net/>

TEIN Topology (~ 31 Jan 2015)



Summary of Routing Protocols





CE3005: Computer Networks
CZ3006: Netcentric Computing

Transport Layer – UDP and TCP

Prof.Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

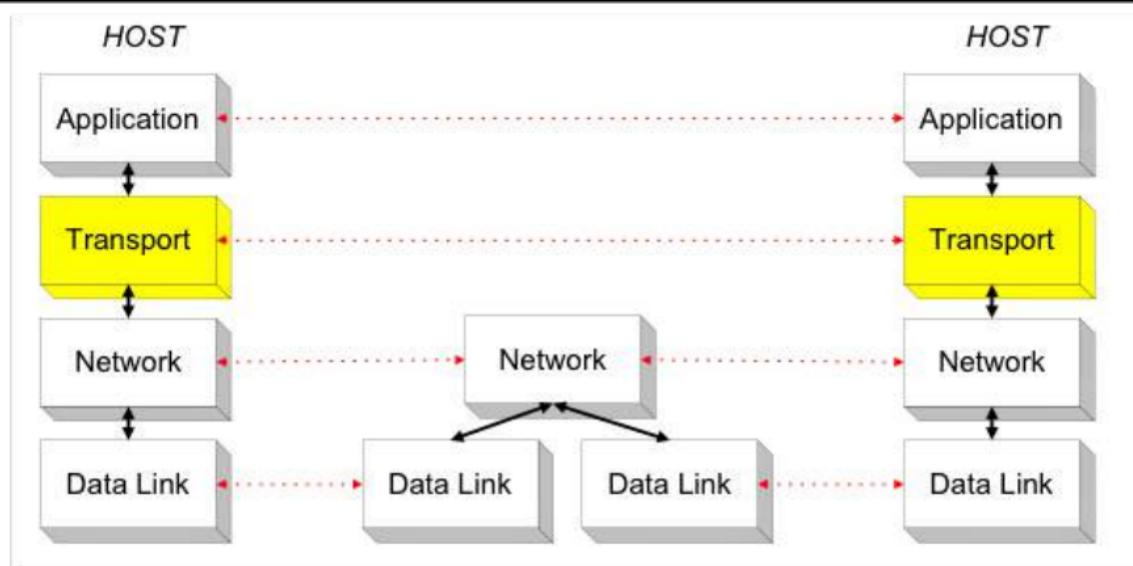
School of Computer Science and Engineering

Contents

- Transport Layer
 - Port Numbers
 - Connectionless Service
 - User Datagram Protocol (UDP)
 - Connection-Oriented Service
 - Transmission Control Protocol (TCP)
 - Transmission Control Protocol (TCP)
 - Connection Management
 - Flow Control
 - Error Control
 - Congestion Control

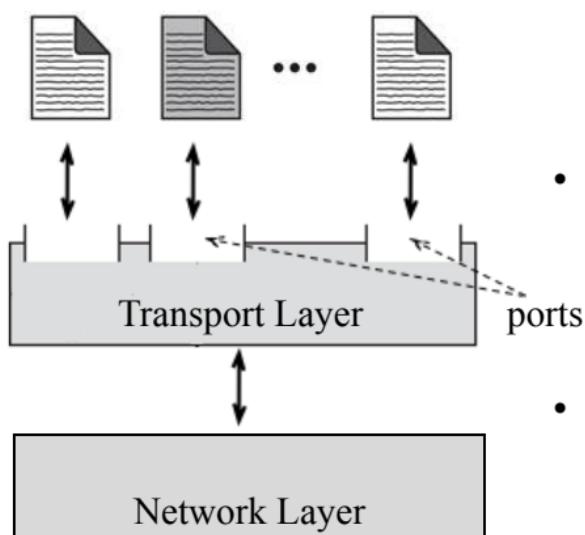
Transport Layer

- Transport layer provides end-to-end service for transferring data between processes (**process-to-process communication**).
- Only implemented at the end hosts.



Transport Layer - Ports

A single transport layer is used to support multiple application processes through the use of ports.



Hence, transport layer is also said to perform **multiplexing/ de-multiplexing**:

- **multiplexing:** gathering data from multiple processes and passing it to a single network layer
- **de-multiplexing:** delivering of data from single network layer to different processes correctly

Transport Protocols in the Internet

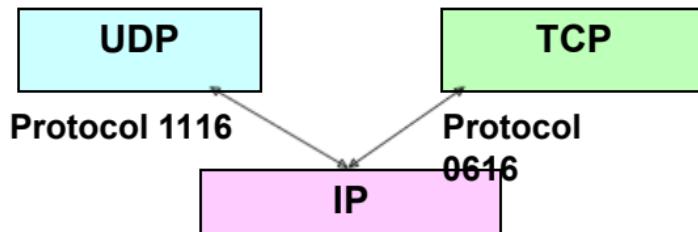
The Internet supports 2 main transport protocols:

UDP - User Datagram Protocol

- **unreliable**, connectionless
- datagram oriented
- simple
- example applications:
 - routing (RIP), domain name service (DNS), DHCP, real-time protocol, etc.

TCP - Transmission Control Protocol

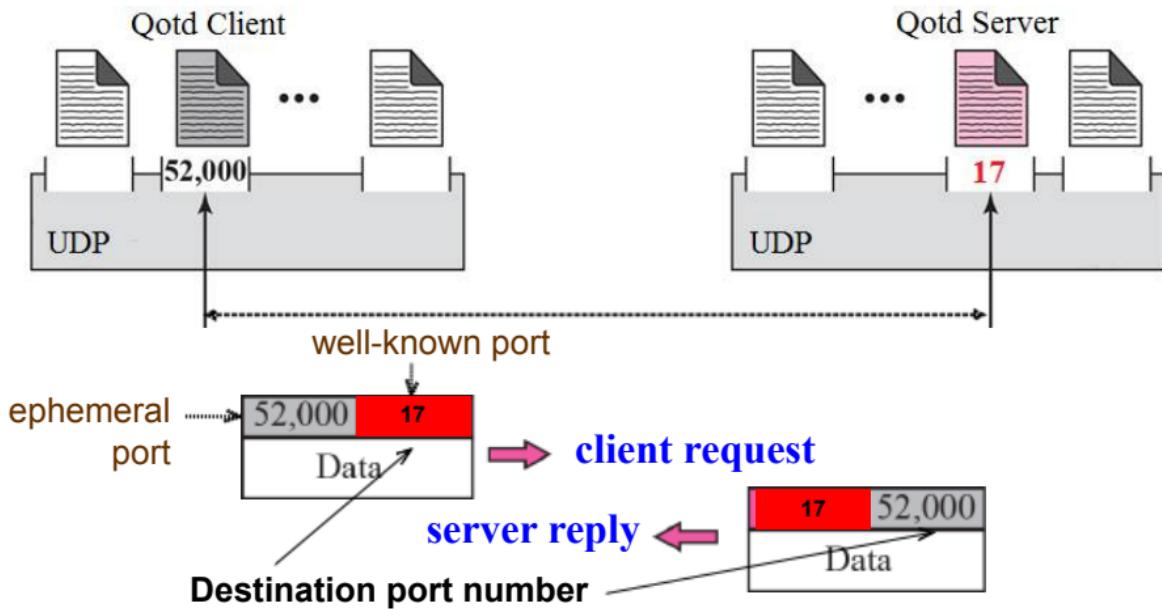
- **reliable**, connection-oriented
- stream oriented
- complex
- example applications:
 - web (http), email (smtp), file transfer (ftp), video streaming, etc.



UDP – Datagram Service

Application layer is aware that UDP sends each message as a datagram.

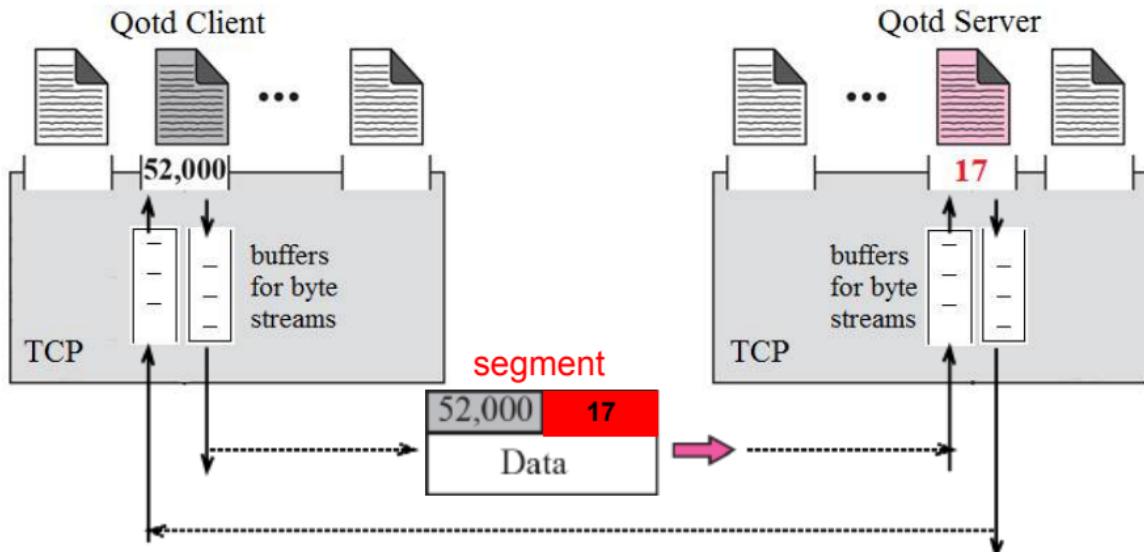
e.g. in Java, `request = new DatagramPacket();`



TCP – Byte Stream Service

Application layer views TCP as a channel for sending stream of bytes, and is NOT aware that bytes are sent in blocks called segments.

e.g. in Java, `outStream = socket.getOutputStream();
request = outStream.write(message);`



User Datagram Protocol (UDP)

Q: If UDP provides non-reliable communications, then why UDP?

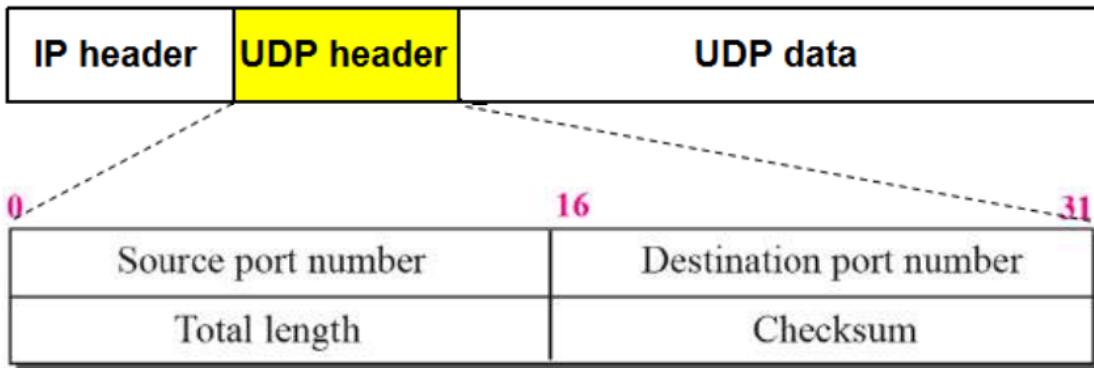
A: Some applications do not need reliable communications. For example:

- Broadcasting, advertising messages to users.
- Sending live video streams over the Internet (loss tolerant, rate sensitive).

Q: So, what does UDP do?

A: Provide **process-to-process** communication service for applications to use.

UDP Header Format



Length of UDP datagram including header

(in bytes):

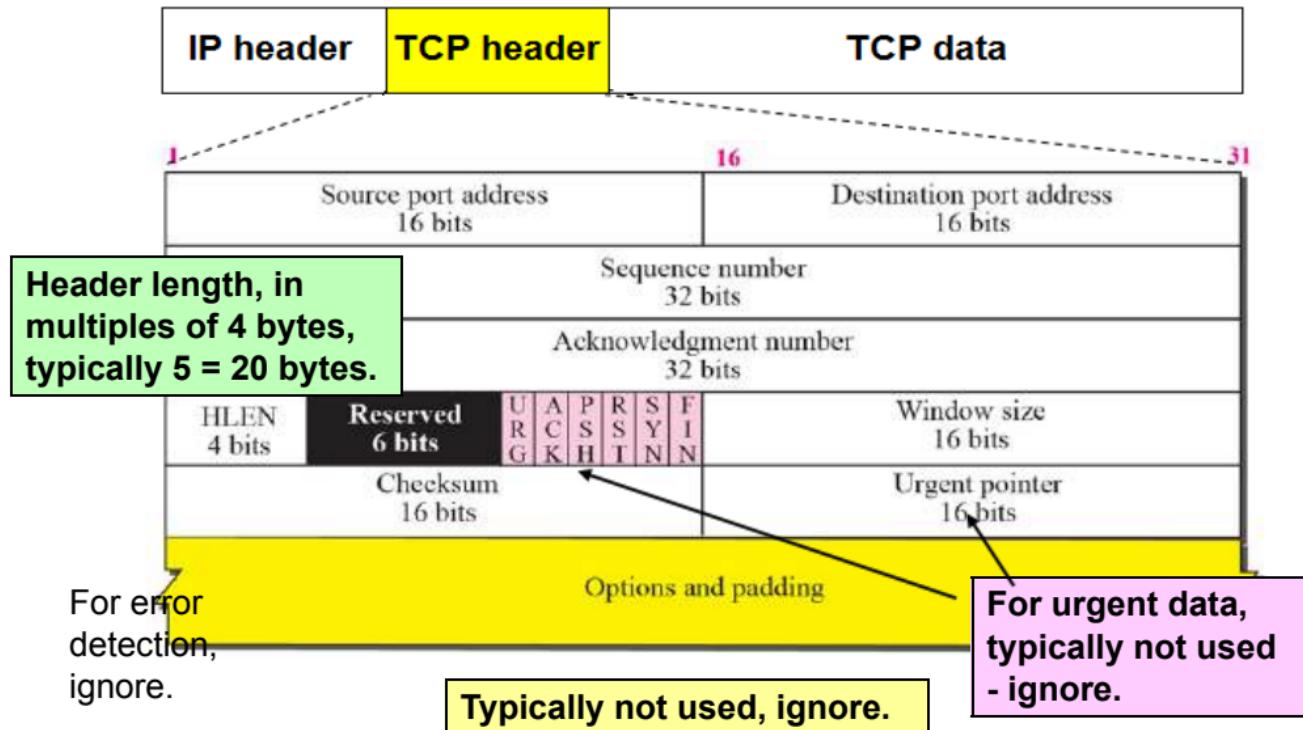
Minimum – 8 (i.e. no data, only header)

Maximum – 65535

Transmission Control Protocol

- To support applications requiring reliable communications, TCP adds **reliability** over **unreliable IP**.
- Essentially, TCP features:
 - **Connection Management:** A connection must be setup before data exchange can be performed.
 - **Flow Control:** Sender will not overwhelm receiver.
 - **Error Control:** Receiver detects errors, sender retransmits error packets.
 - **Congestion Control:** During transmission, sender detects network usage(congestion) and adjust transmission rate.

TCP Header Format



TCP Header Format

- Port addresses

Port address	Type	Description of use
0-1023	Well Known port	Used by system processes to provide network services
1024-49151	Registered port	Assigned by IANA upon request by entities. Can also be used by user.
49152-65535	Ephemeral port	Dynamic or private ports that cannot be registered by IANA

TCP Header Format

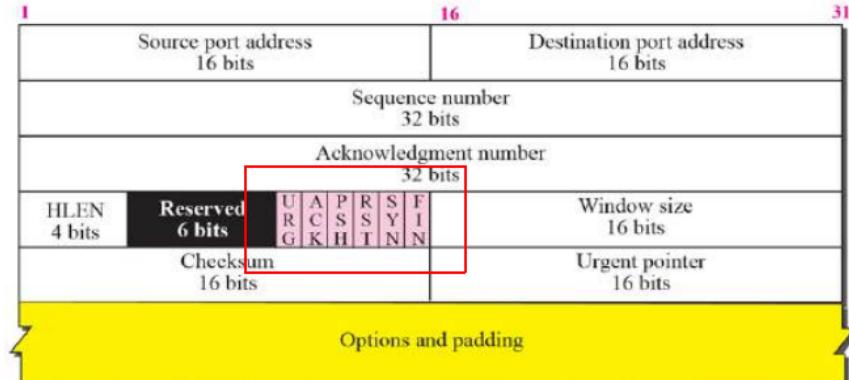
- Sequence Number (SN):
 - Each TCP connection will start with a different SN called Initial Sequence Number (ISN)
 - The position of each data byte in the byte stream is labelled from ISN+1, and cycle back to 0 once reaching $2^{32} - 1$; i.e. 1st byte $(ISN+1) \bmod 2^{32}$, 2nd byte $(ISN+2) \bmod 2^{32}$, ...
 - SN indicates the position of the 1st byte in each segment
- Acknowledgement Number (AN):
 - AN of the next data byte expected from sender
 - Also imply all bytes up to AN-1 have been received correctly
- Window Size (W):
 - Indicate the number of bytes (also called credits) counting from AN that the receiver is ready to accept

Note: Other fields will be discussed in relevant slides later.

TCP Connection Management

- **Connection establishment: serves the following purposes**
 - ensure both ends are ready to communicate
 - establish **initial sequence number (ISN)**
 - exchange parameter, e.g. **window size (in bytes)**
 - allocate resources, eg. buffer space, etc. to support the connection
- **Connection establishment starts with a synchronization (SYN) request.**

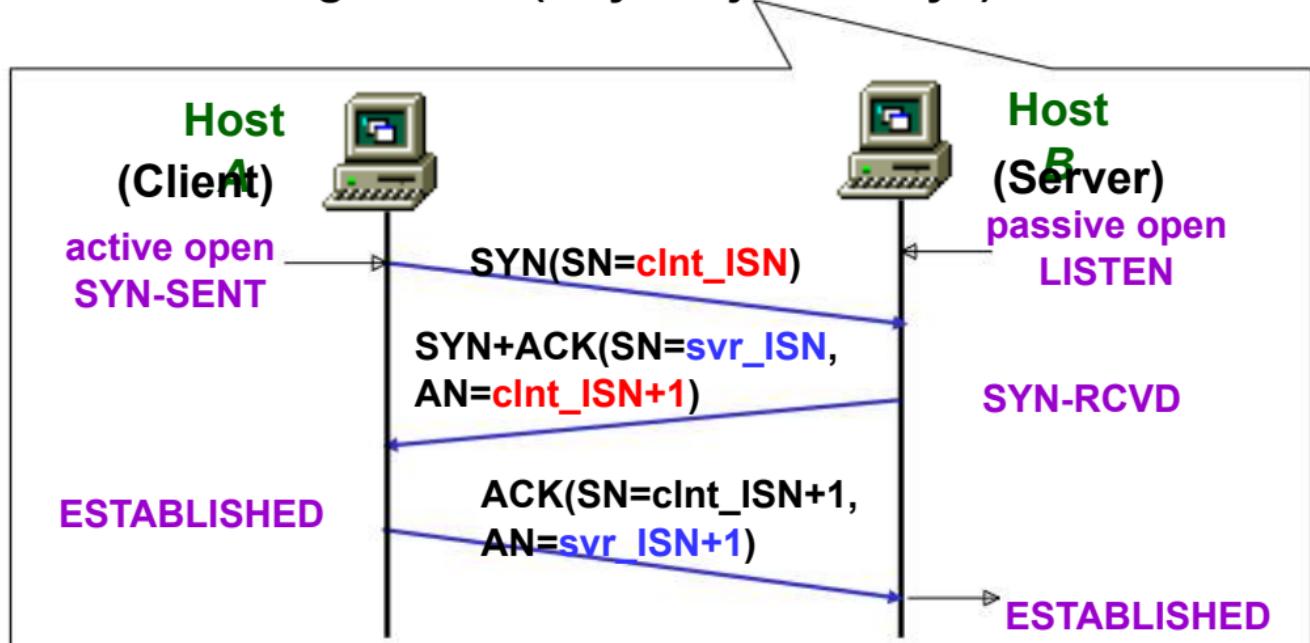
Control Bits



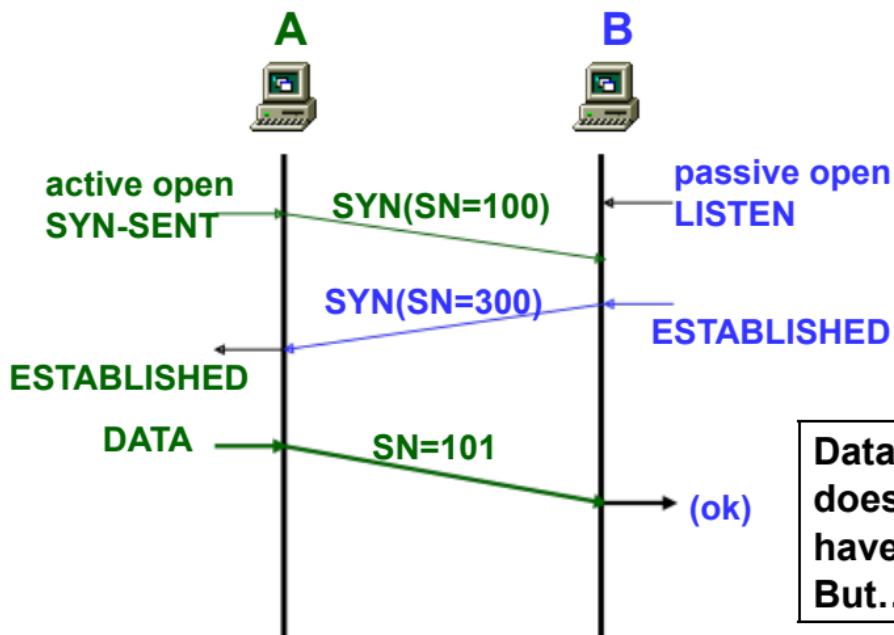
- **SYN = 1: Synchronize sequence numbers**
 - Used to establish connection
- **FIN = 1: No more data from me!**
 - Used to terminate connection
- **RST = 1: Reset the connection**
 - Used when error occurs during connection establishment
- **ACK = 1: Acknowledgment number is valid**

TCP: Connection Establishment

- TCP uses 3-way handshake approach with positive acknowledgements (why not just 2-way?)



Two-way Handshake without positive acknowledgement

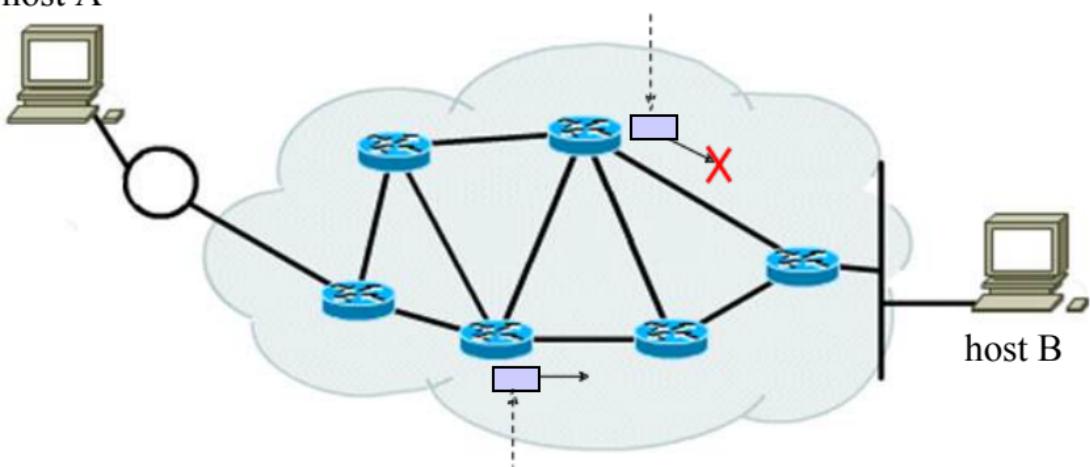


Hosts A & B use a two-way handshake for connection setup. They exchange their ISN with each other.

Data exchange doesn't seem to have any problem. But...

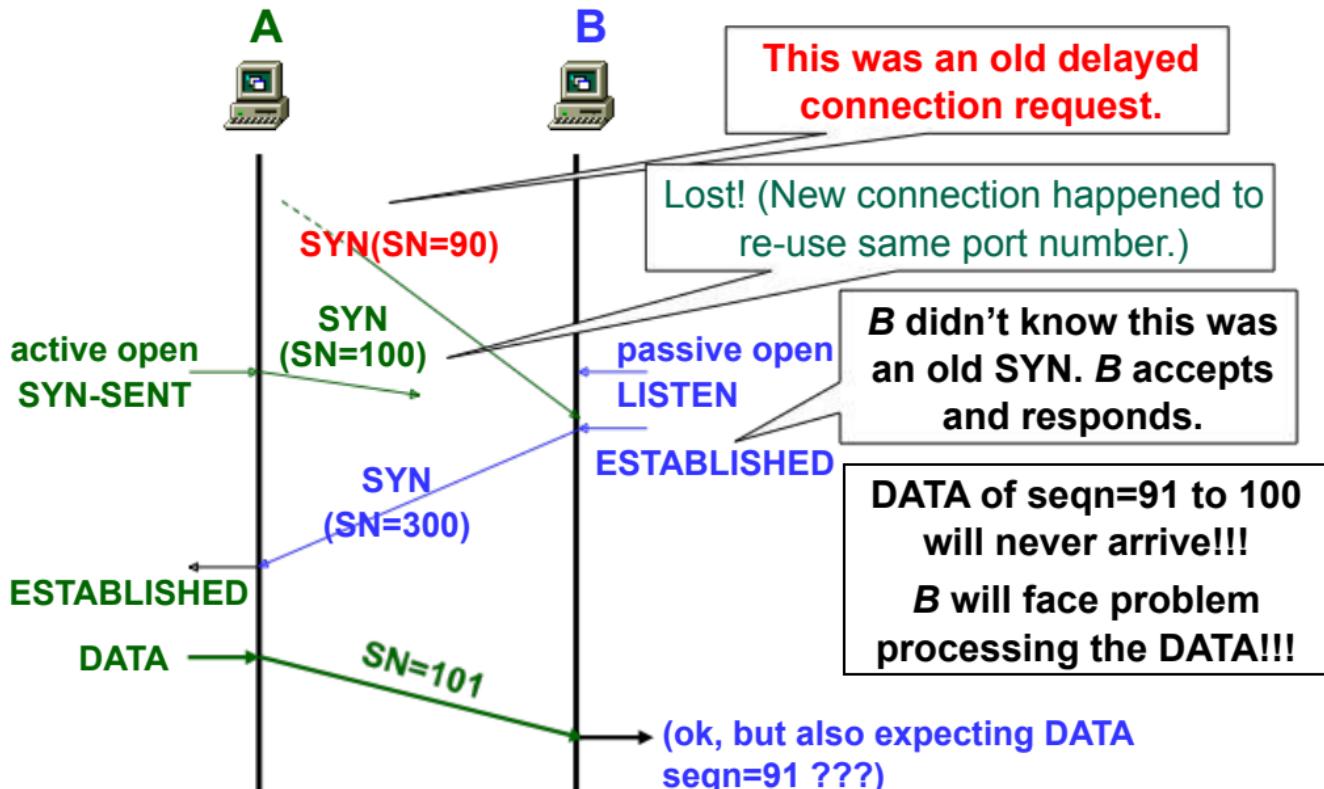
Recall that IP is unreliable. It is possible for packets to be delayed, lost, or duplicated due to timeout re-sent by TCP.

New SYN packet from A is sent to B but unfortunately it's lost!



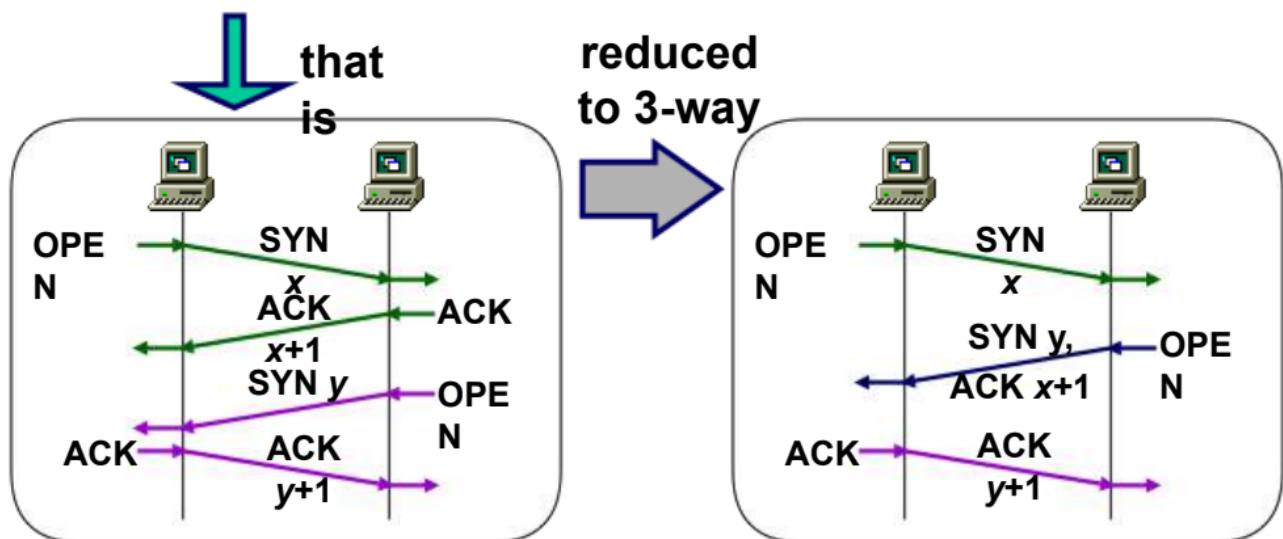
Old SYN packet from A is finally going to B after being delayed by congested router.

Two-way Handshake Problem

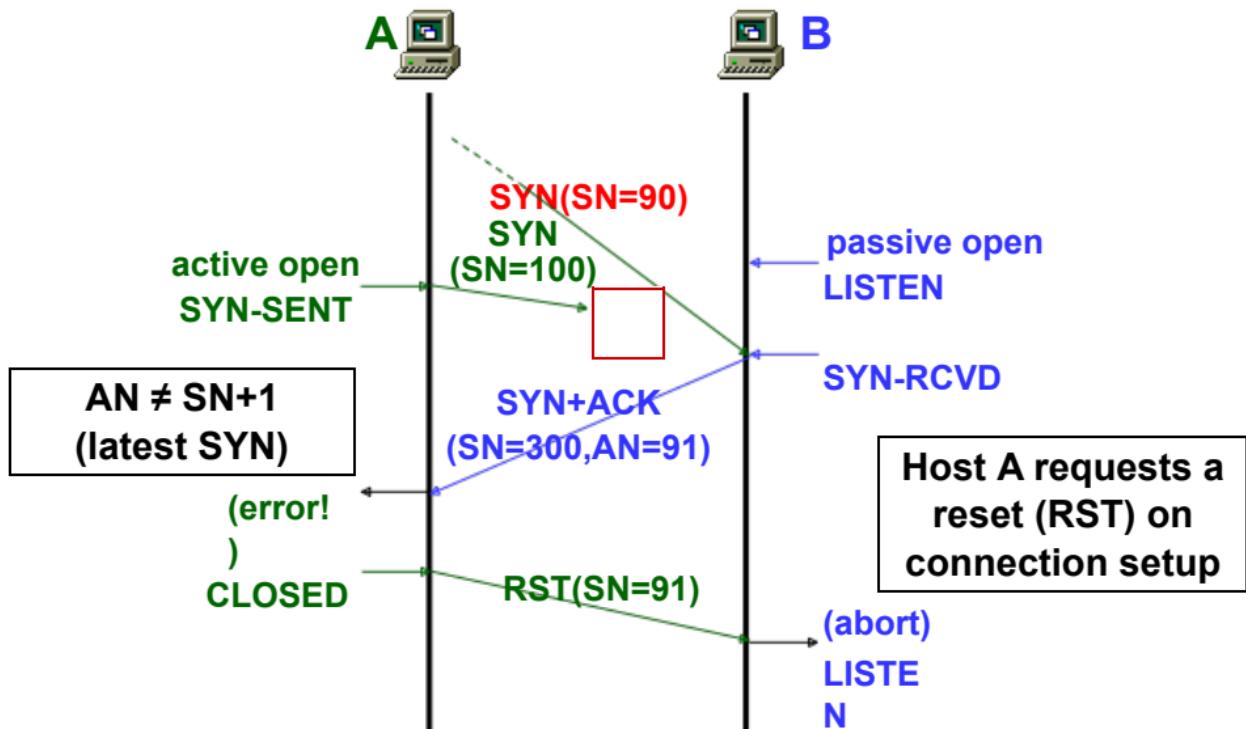


Solution: Three-way Handshake

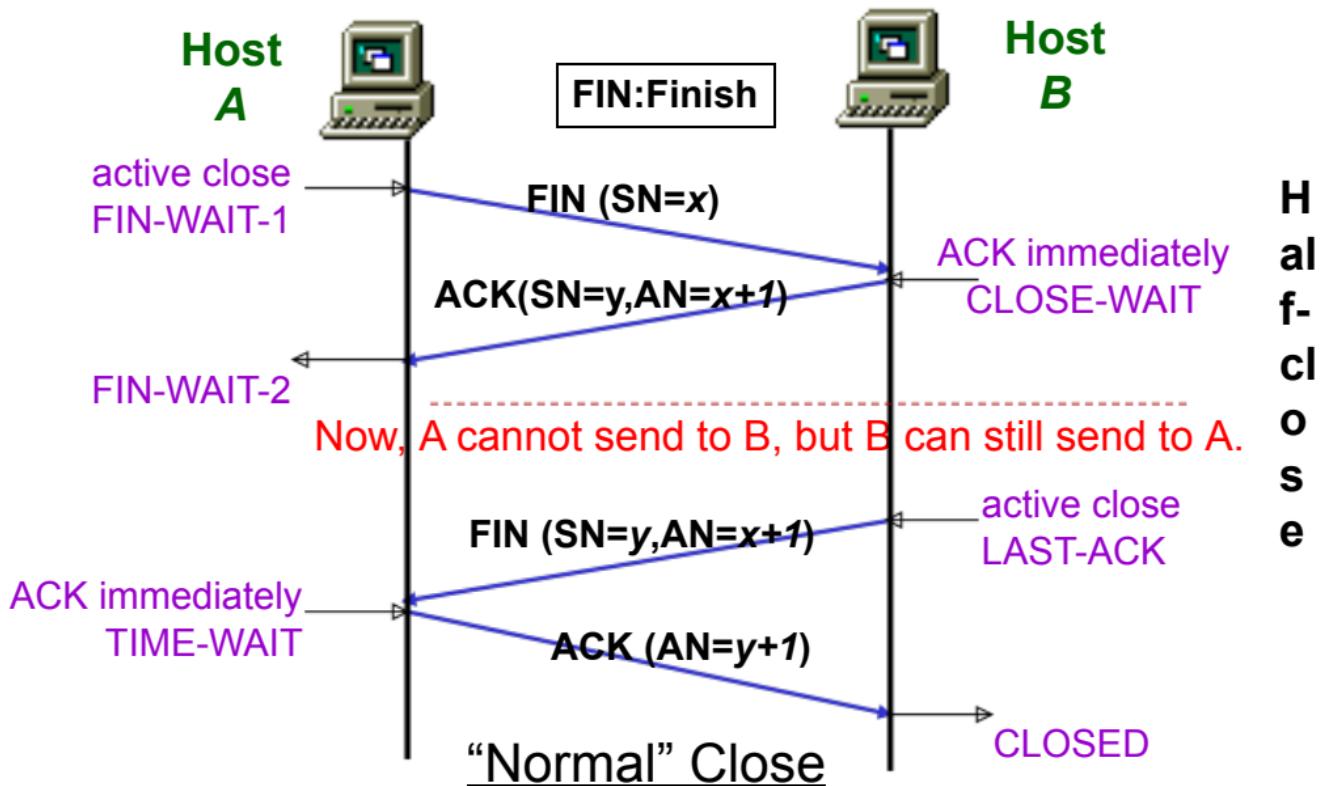
Synchronization can be made reliable if each connection request is positively acknowledged.



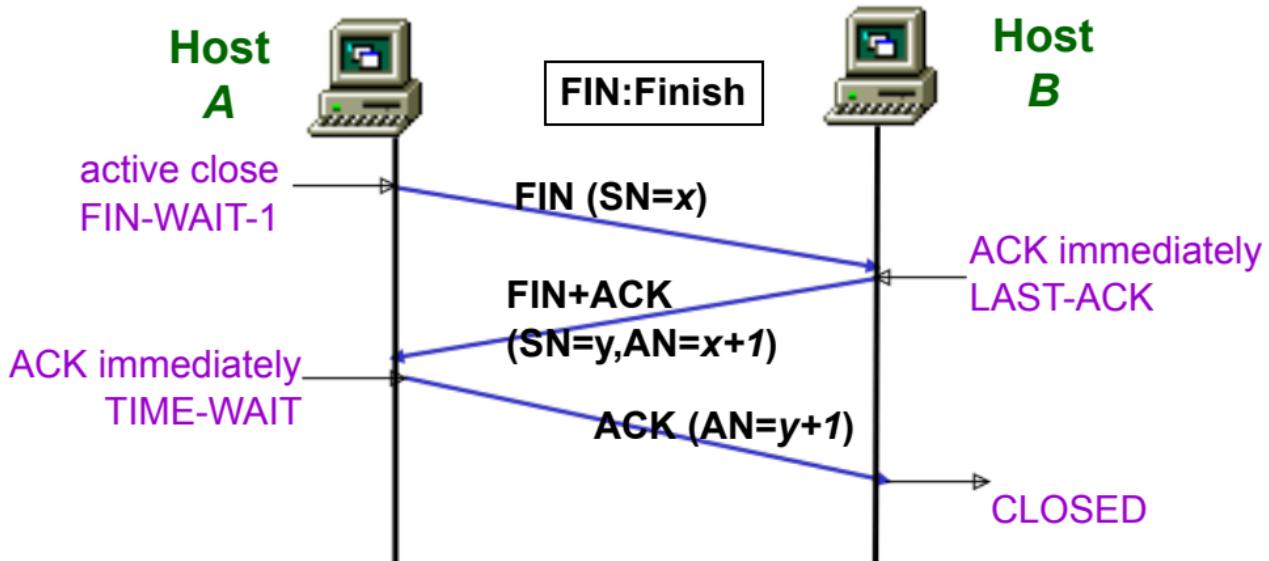
Three-way Handshake (Put to test)



TCP: Connection Termination

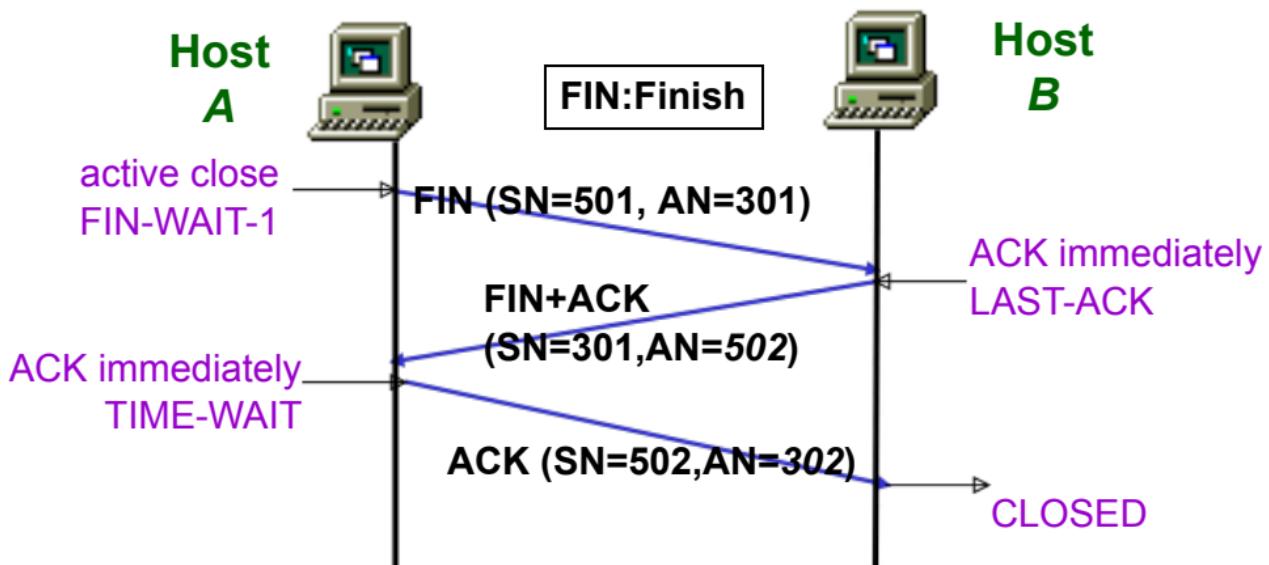


TCP: Connection Termination (another scenario)



When B does not have outstanding data to send to A when receiving FIN from A.

TCP: Connection Termination (Numerical example)

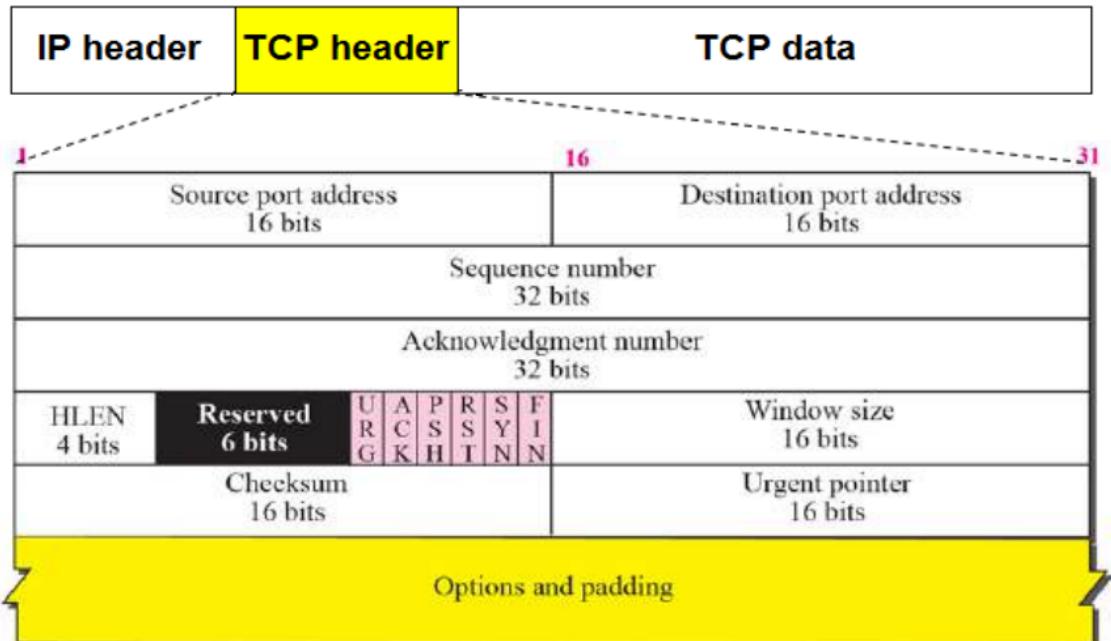


SN

- Note:

- A SYN and FIN segment does not carry data but consumes one sequence number
- A SYN +ACK, FIN+ ACK does not carry data but consumes one sequence number
- ACK segment, carrying no data, does not consume any sequence number.

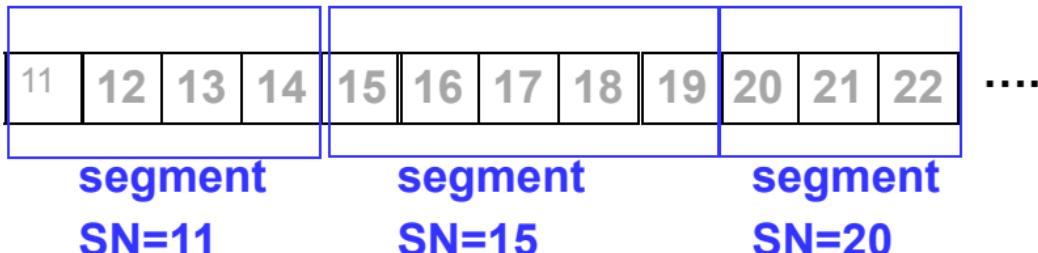
TCP Header Format



Sequence Number (SN)

- 32 bit, range: 0 ~ 2³²-1
- Each connection starts with a different SN called **Initial Sequence Number (ISN)**
- The position of each **data byte** in the byte stream is labelled from ISN+1, and cycle back to 0 once reaching 2³² -1
- SN of a TCP segment indicates the position of its 1st **data byte** in a data stream

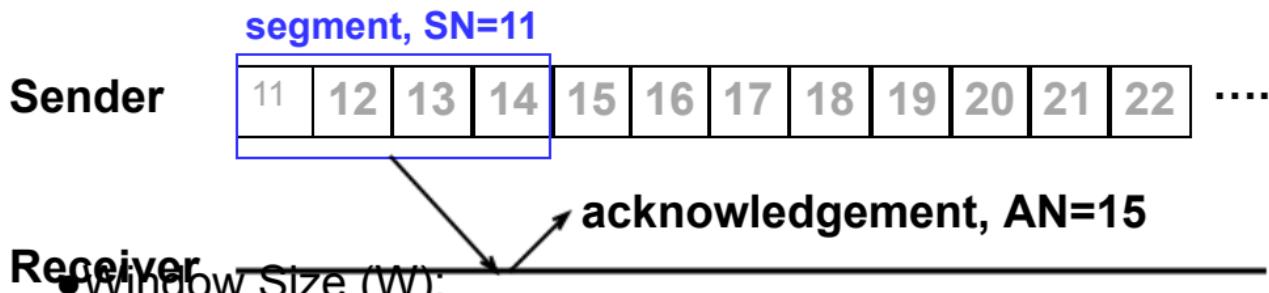
Byte stream to be sent, ISN=11



Acknowledgement Number & Window Size

- Acknowledgement Number (AN):

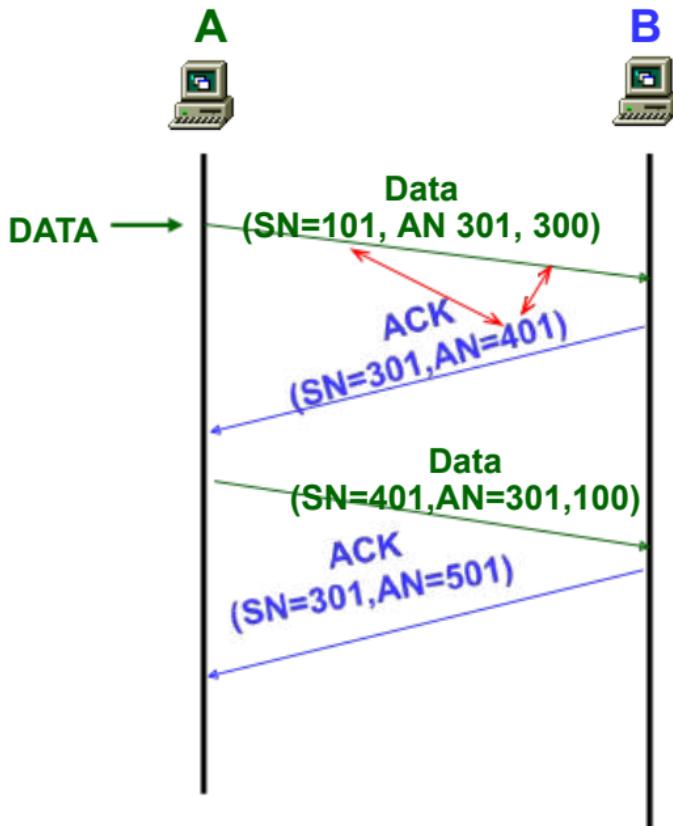
- SN of the next data byte expected from sender
- Imply all bytes up to AN-1 have been received correctly



- Window Size (W):

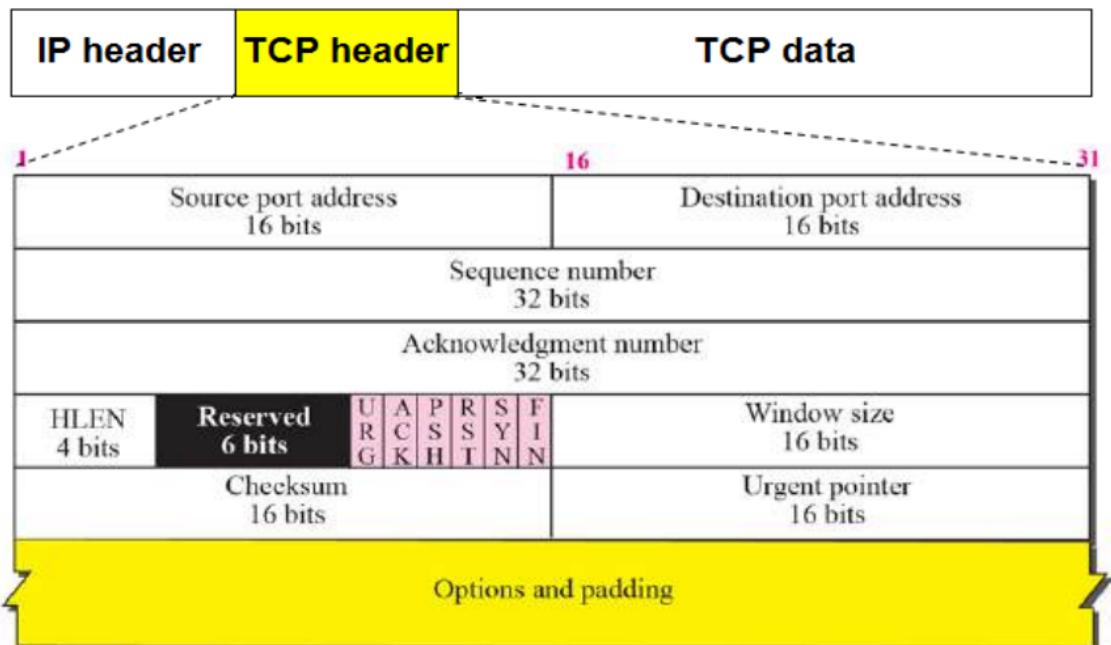
- The number of bytes counting from AN that the receiver is ready to accept

Data Transfer... (A -□ B)



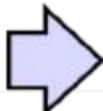
Segment format
(SN, AN, Data size)

TCP Header Format



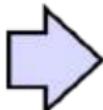
TCP Flow Control

WHY
?



So that sender won't **overrun receiver's buffers** by transmitting too much, too fast.

HOW
?



Similar to **sliding-window flow control** in **datalink layer**, although some details are different.

To support bi-directional data transfer, 2 pairs of windows are used:

A \rightarrow B: sender window



segment

A \rightarrow B: receiver window



B \rightarrow A: receiver window



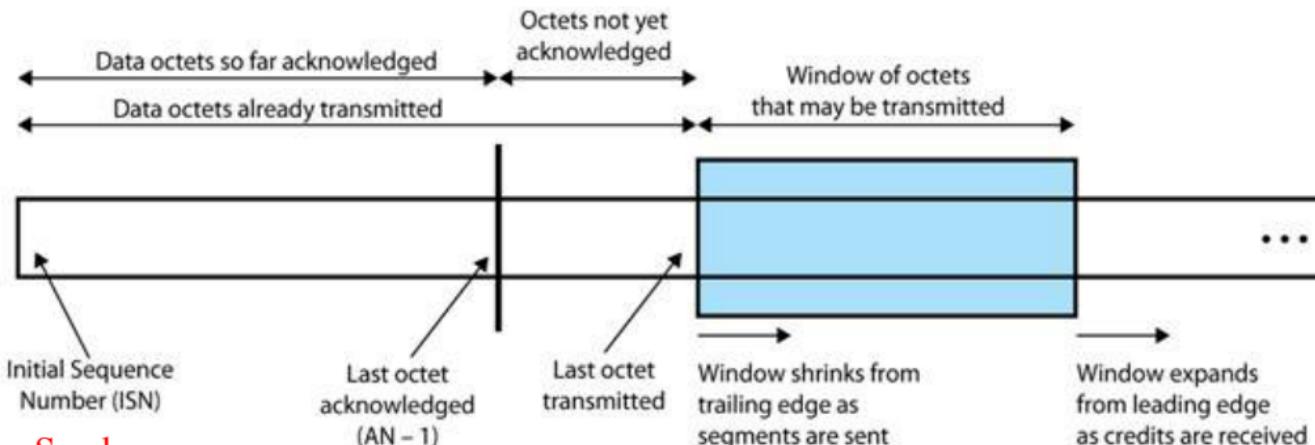
Host A

Host B

B \rightarrow A: sender window



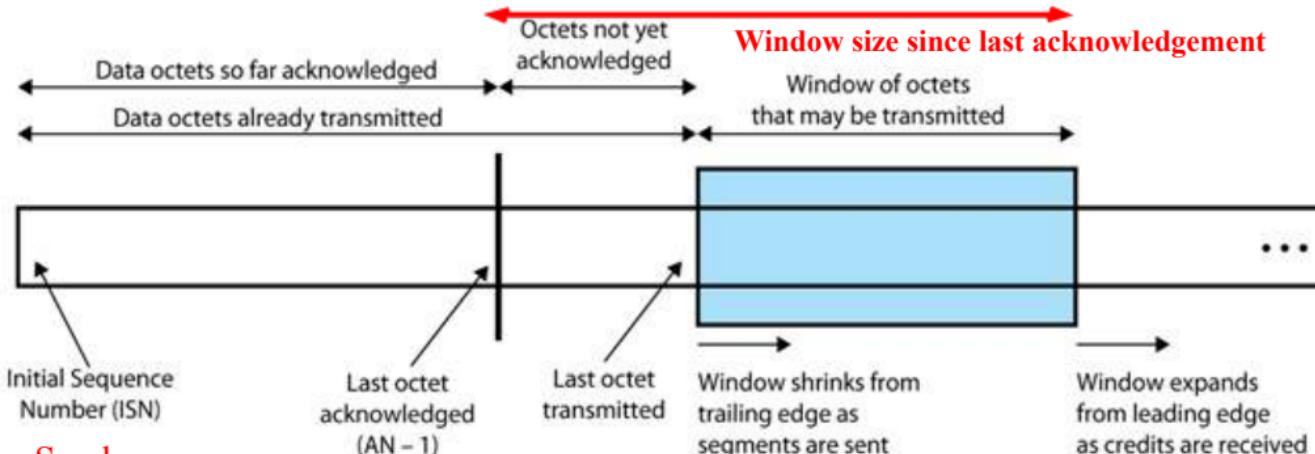
In **sliding-window flow control**, multiple segments are allowed to be in transit at the same time. The behaviour of **sender** is illustrated below:



Sender:

- Maintain a **blue window** representing bytes that can be transmitted without ACK
- When segment is **sent**, **shrink blue window** from trailing edge
- Stop sending when **blue window size = 0**
- When **ACK** is **received**, new **blue window size = W bytes** starting from AN.

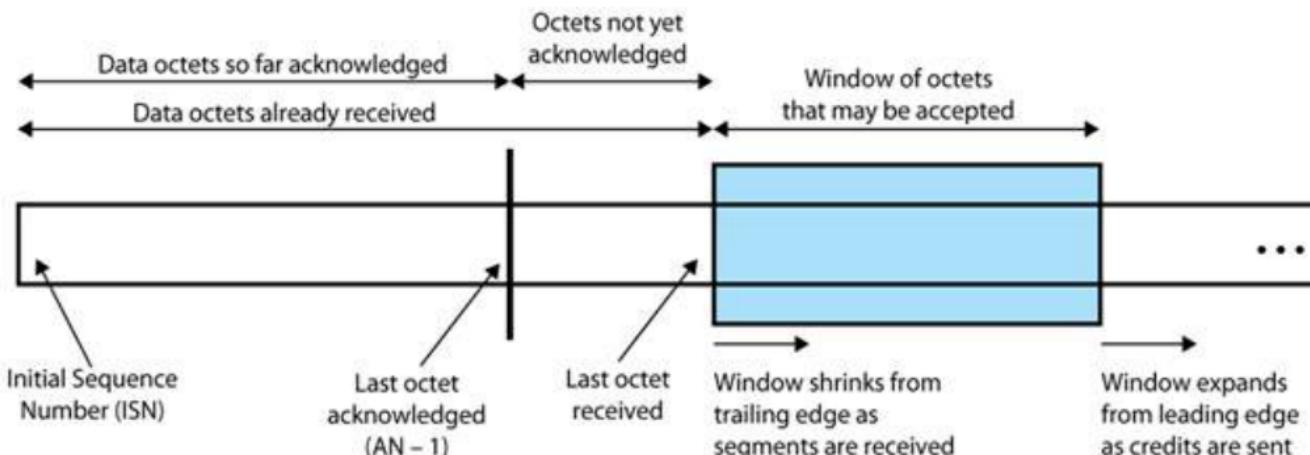
In **sliding-window flow control**, multiple segments are allowed to be in transit at the same time. The behaviour of **sender** is illustrated below:



Sender:

- Maintain a **blue window** representing bytes that can be transmitted without ACK
- When segment is **sent**, shrink **blue window** from trailing edge
- Stop sending when **blue window size = 0**
- When **ACK** is **received**, new **blue window size = W bytes** starting from AN.

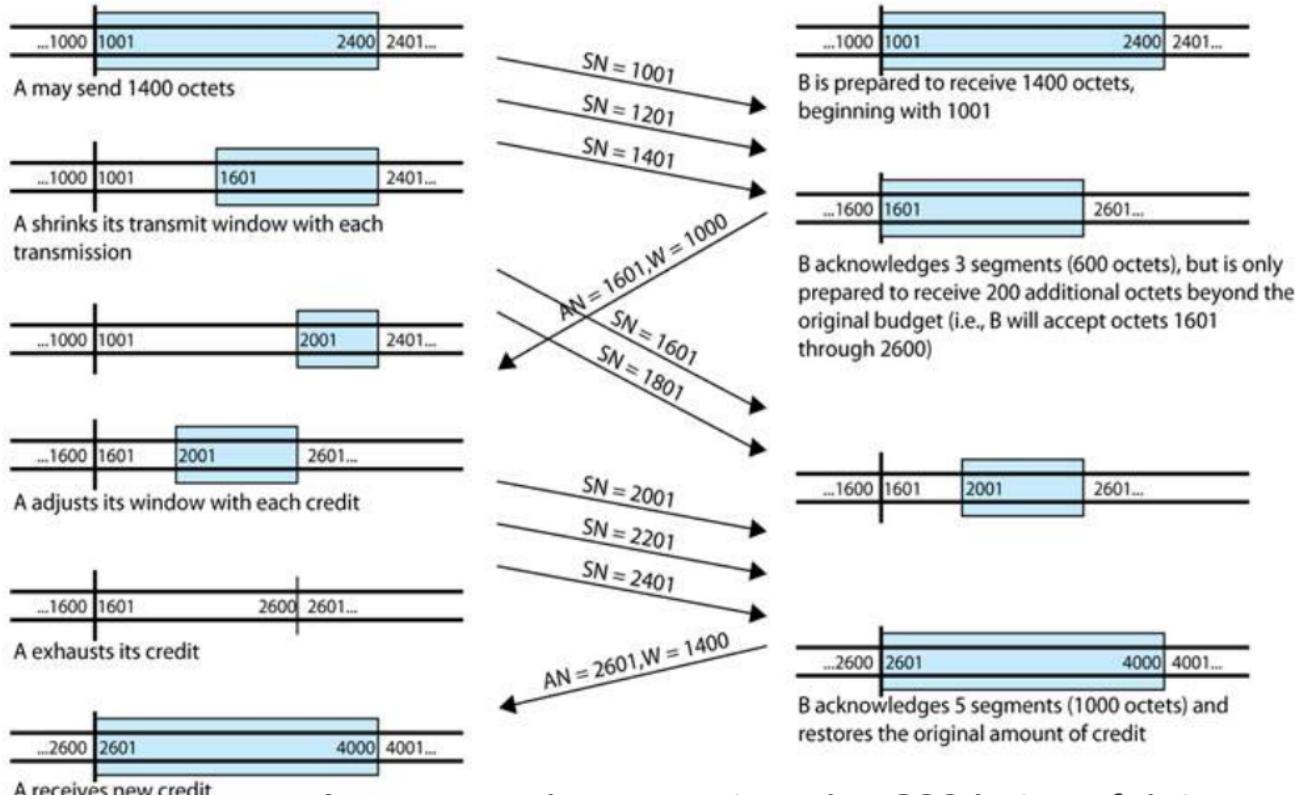
The corresponding behaviour of **receiver** in the **sliding-window flow control** is illustrated below:



Receiver:

- Maintain a **blue window** representing bytes ready to accept
- When segment is received, **shrink blue window** from trailing edge
- If **NOT** ready to accept more segments, **send ACK with credit $W =$ remaining blue window size**
- If **ready** to accept more segments, **send ACK with $W >$ remaining blue window size**, and **expand blue window** from leading edge

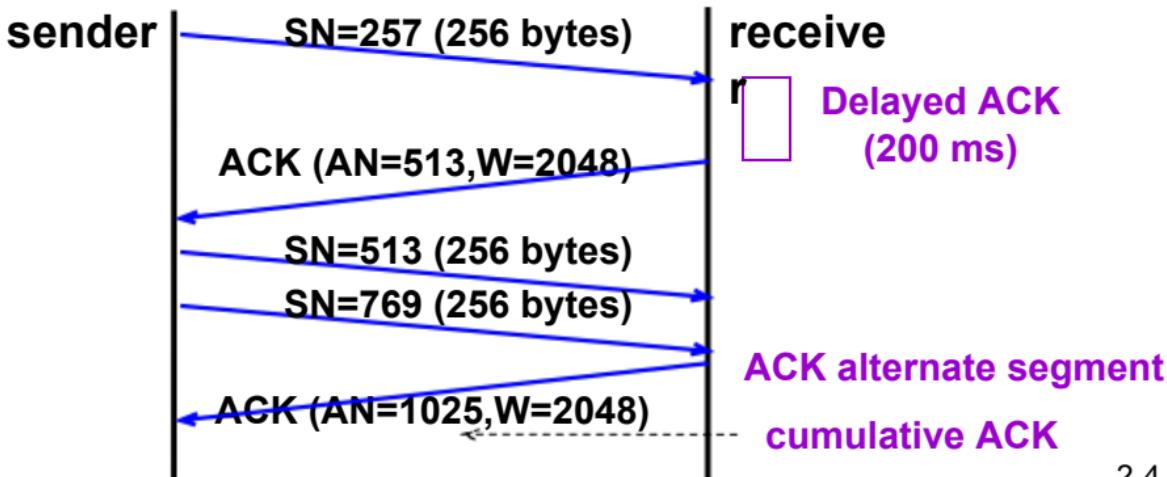
Here is an overall picture of how **sliding-window** flow control works:



TCP Flow Control Enhancement 1 – Delayed ACK (RFC 1122)

Problem: Wasteful to send ACK only segment (40 bytes TCP+IP headers)

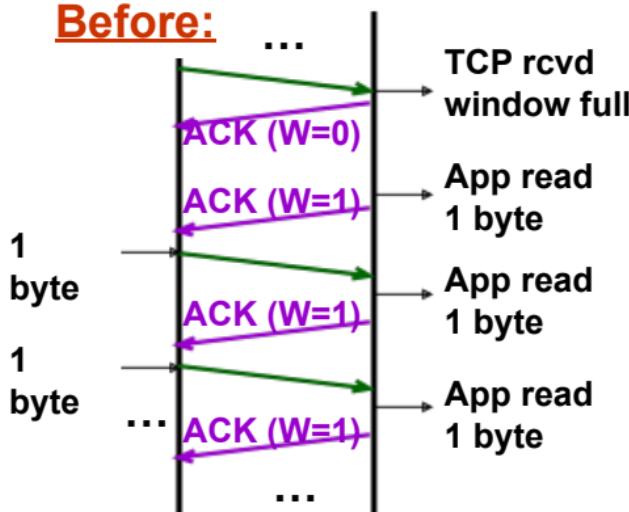
- Maximum < 500 ms, to avoid error-control timeout re-sent
- ACK every alternate segment received
- (Note: Piggy-backed ACK can be sent immediately)



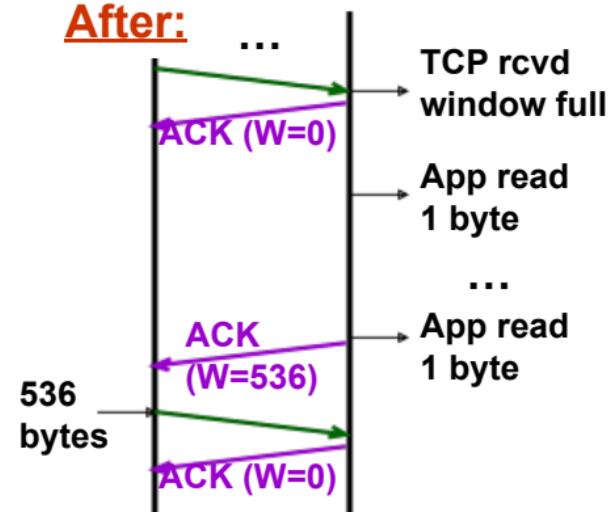
TCP Enhancement 2: avoiding Silly Window Syndrome at Receiver – Clark's solution

Problem: Wasteful for receiver to keep ACK with small window when sender can send more

Before:



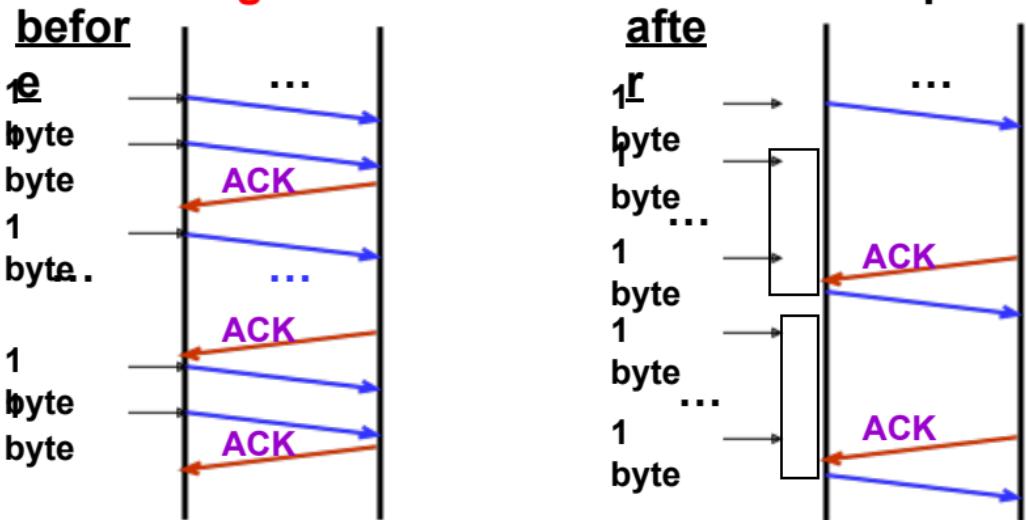
After:



Solution: Receiver ACK W=0 instead of small window size until free buffer gets large; e.g. buffer half empty.

TCP Enhancement 3: avoiding Silly Window Syndrome at Sender - Nagle's Algorithm (RFC 896, RFC 1122)

Problem: Wasteful for **sender** to keep **sending small segments** when receiver can accept more



Solution: Send the 1st small segment, buffer the rest and send them together when ACK is returned.

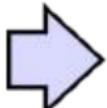
TCP Error Control

WHY
?



So that **TCP** can guarantee **reliable** service to application layer even when **IP** is **unreliable**.

HOW
?



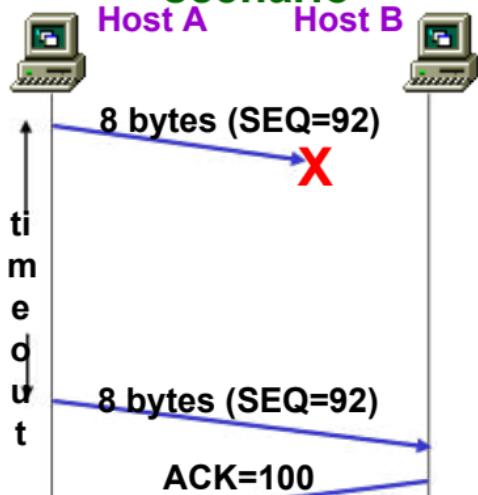
Similar to Selective-Reject in **datalink layer**, but the details are different.

Error types:

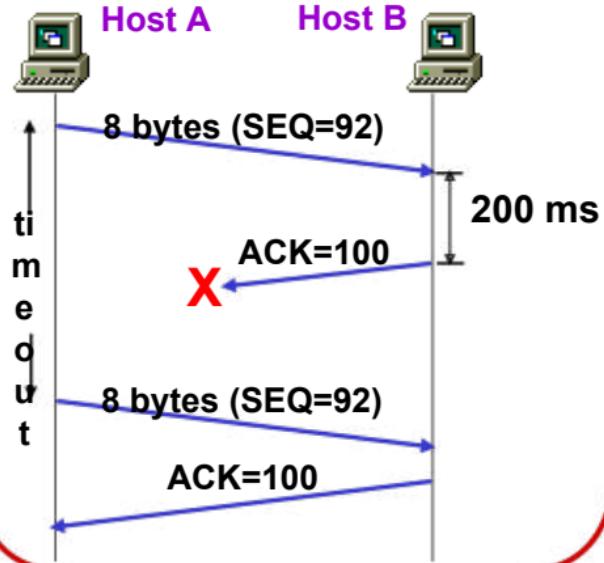
- Segments arriving out-of-order
 - Detected based on SN in TCP header; re-order and ACK
- Segments duplicated
 - Detected based on SN; discard and ACK
- Segments corrupted
 - Detected based on checksum in TCP header
 - Discard and wait for timeout retransmission
- Segments loss
 - Wait for timeout retransmission

Examples of timeout retransmission scenarios

Lost data scenario



Lost ACK scenario



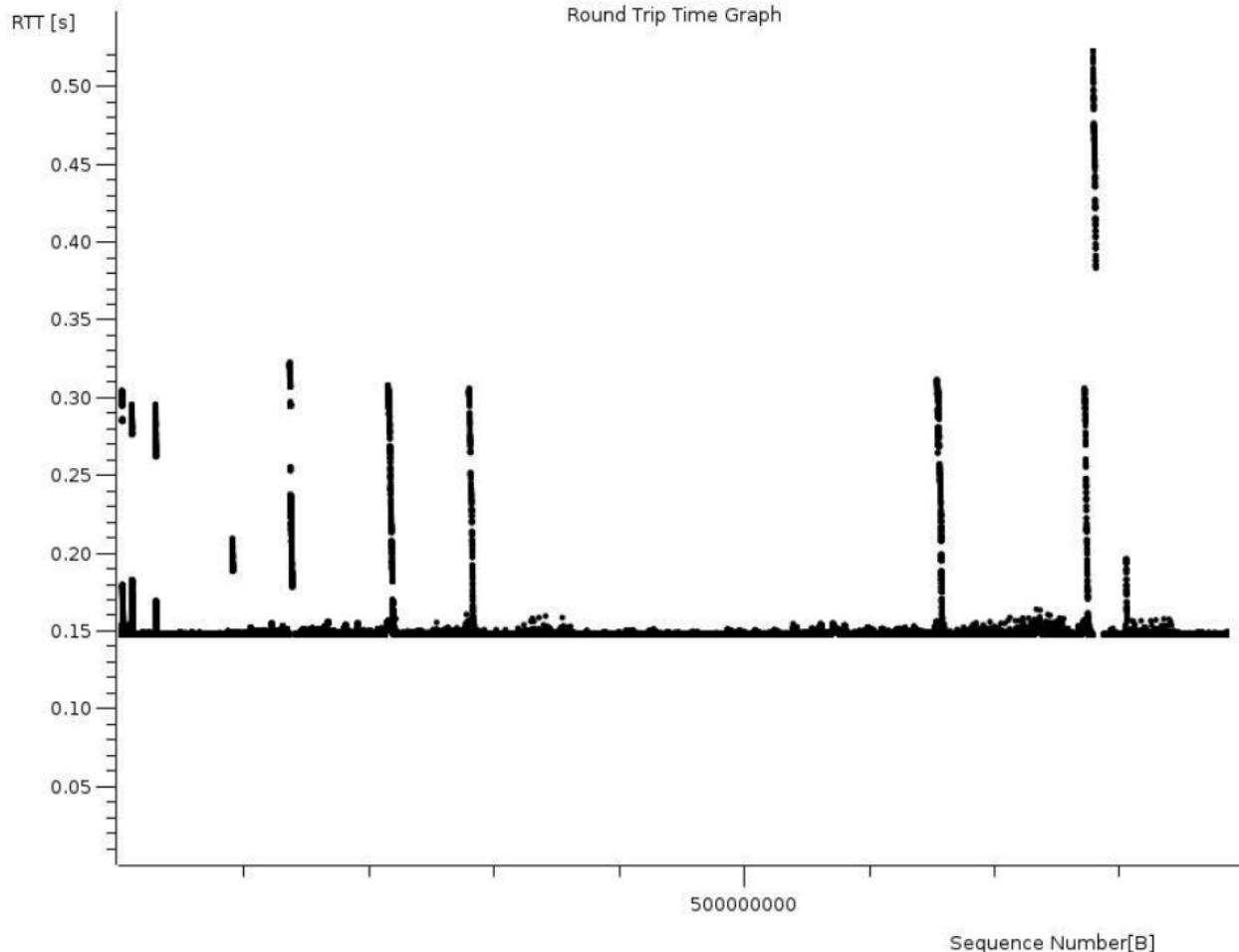
Retransmission Timer

Problem: How long should TCP wait for the ACK before it retransmits that segment?

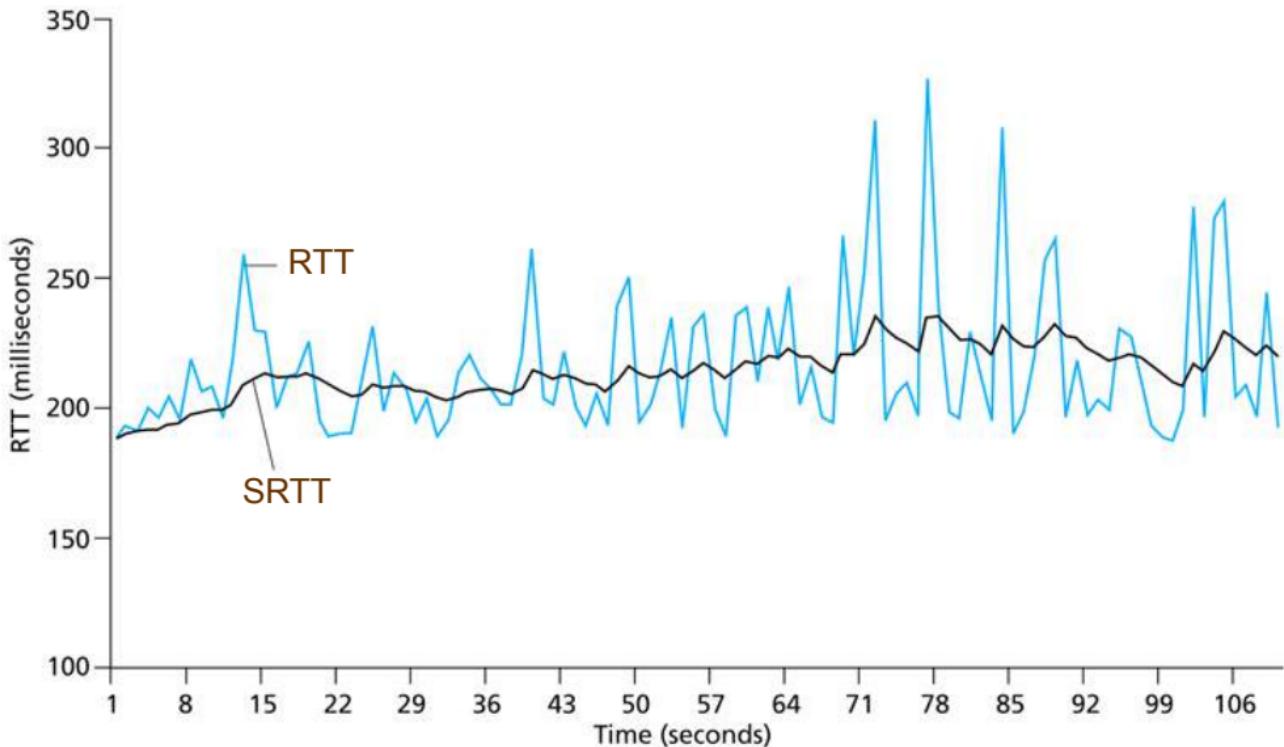
- Too short** (premature timeout): there will be unnecessary retransmission
- Too long** (slow reaction to losses): a long period of time is required to discover a lost segment

Note: Delays in network are constantly changing in practice, so timeout must be adaptive.

Solution: Measure Round Trip Time (RTT) and compute smoothed RTT (SRTT). The Retransmission TimeOut (RTO) is then derived from SRTT.



Example of RTT and SRTT



Computing Retransmission Timer RTO – Jacobson's Algorithm (RFC 6298)

- Initialization:
- After 1st RTT is measured:

(smoothed RTT) $SRTT = RTT$
(RTT variation) $RTTVar = RTT/2$
 $RTO = SRTT + 4 \cdot RTTVar$

- After each subsequent RTT is measured:

$RTTVar = (1 - \alpha) \cdot RTTVar + \alpha \cdot |SRTT - RTT|, \alpha = 1/4$
 $SRTT = (1 - \alpha) \cdot SRTT + \alpha \cdot RTT, \alpha = 1/8$
 $RTO = SRTT + 4 \cdot RTTVar$

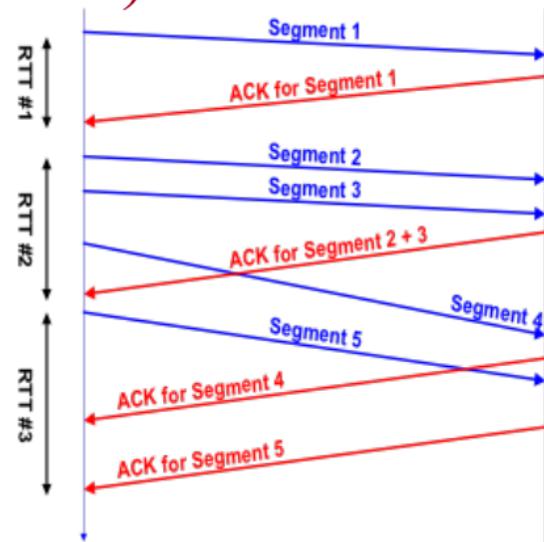
- (minimum) 1s \leq RTO \leq maximum (at least 60s)

Round	10	11
SRTT	20	$((7/8) \cdot 20) + ((1/8) \cdot 16) = 19.5$
RTTVar	10	$((3/4) \cdot 10) + ((1/4) \cdot 4) = 8.5$
RTT	16	
RTO	60	53.5

Measuring RTT – Karn's Algorithm (RFC 1122)

Each TCP connection **measures** the **RTT** from sending a segment to receiving its corresponding ACK.

Typically, there is only one measurement ongoing at any time (i.e. measurements do not overlap).



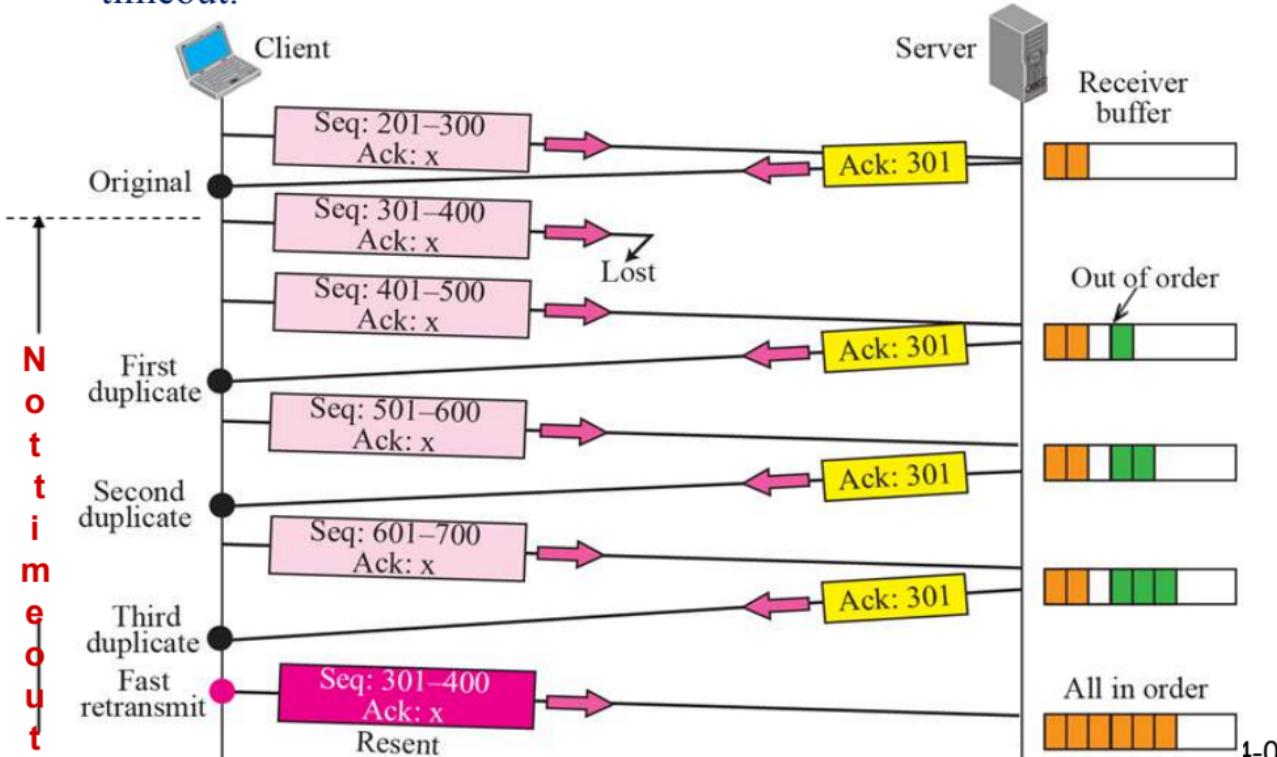
Karn's Algorithm:

- If a segment is **retransmitted** due to timeout, **ignore** its **measured RTT** because it is ambiguous whether the ACK is for 1st or re-transmission.
- When retransmission occurs, set **RTO = 2 \square RTO**

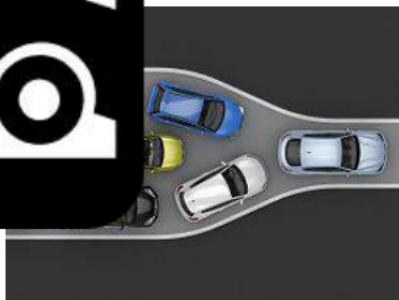
TCP Error Control Enhancement – Fast Retransmit

Fast

- Fast retransmit if receiving 3 duplicate ACKs instead of waiting for timeout.

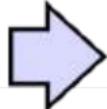


Congestion



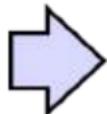
Congestion Control

WHY
?



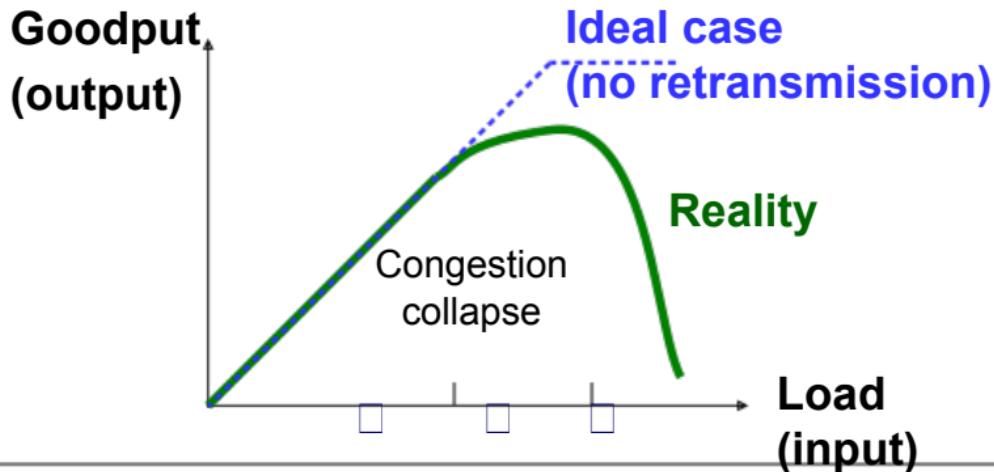
To prevent **senders** from **sending too much** traffic to the network such that it becomes overly congested and useless; informally, to be a “**considerate user**” of the network

HOW
?



Implement **congestion control algorithm** with a **congestion window** which controls the amount of traffic that a connection can send

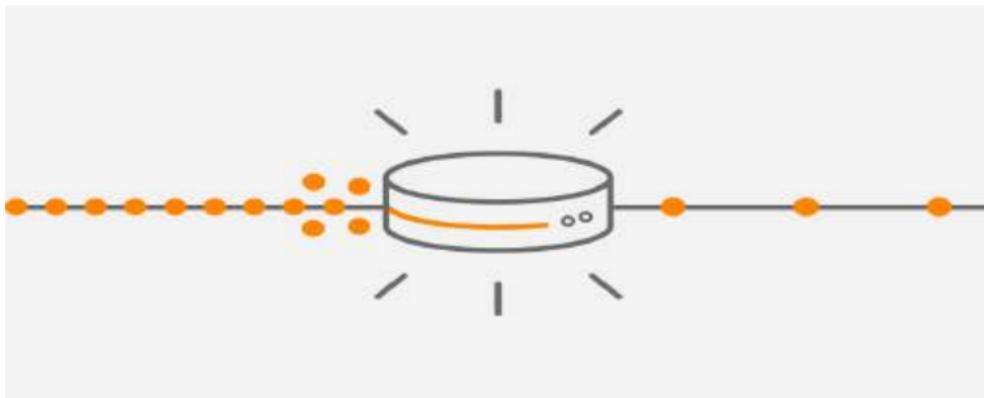
What will happen if network congestion is ignored? **Congestion Collapse!**



$$\text{Output} = \text{Input} + \text{Retransmission}$$

- Zero retransmission, hence $\text{Output} = \text{Input}$
- Several retransmissions, hence output continues to increase slightly as input increases
- Transmissions are dominated by **retransmissions**

Congestion Control



TCP sender assumes network congestion when **LOSS events** occurred:

- **Timeout or receiving duplicate ACKs**
- **Possibly due to queueing or buffer overflow at routers which are signs of congestion**

TCP Congestion Control

- Two phases
 - Slow Start
 - Congestion Avoidance
- Two parameters:
 - **cwnd**: Congestion Window, measured in number of **MSS** (maximum segment size, typically **536 bytes**, but can be changed by using TCP options field).
 - **ssthresh**: Slow Start Threshold defines the point to transit from slow start to congestion avoidance phase; in practice, typically set large value for initial **ssthresh** (**half- maximum number of MSS**)

Note: With congestion control, maximum data bytes that can be sent without ACK = $\min \{ W, cwnd \times MSS \}$

Window size
in bytes

Convert to bytes

TCP Slow Start Phase

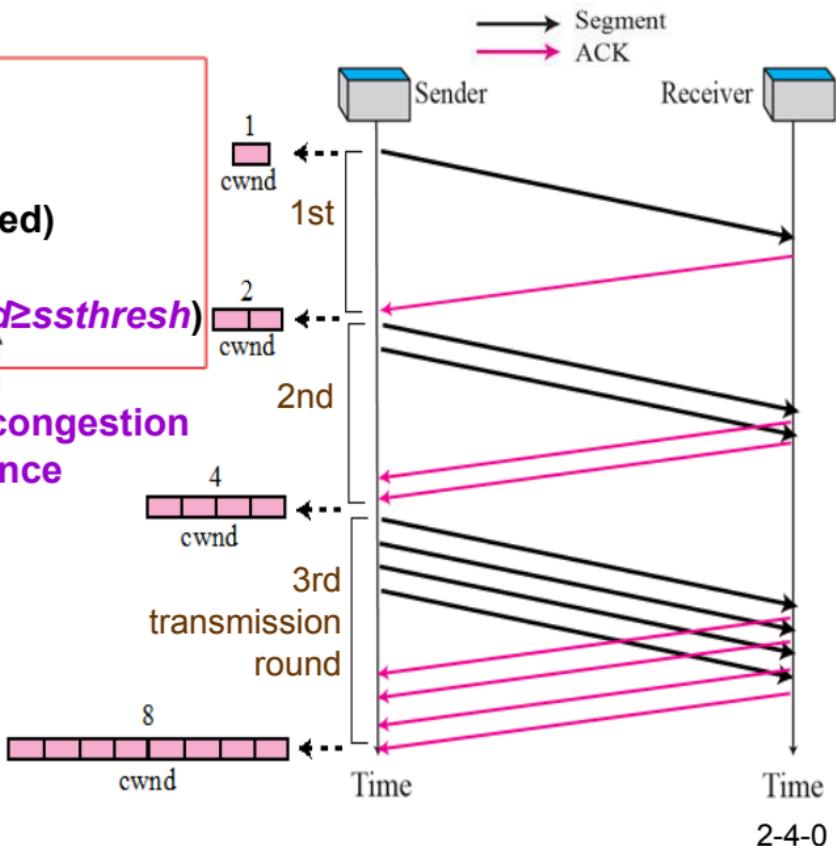
Slow start

```
initialize: cwnd = 1
do
    for (each segment ACKed)
        cwnd++
    until (loss event OR cwnd ≥ ssthresh)
```

Go to recovery

Go to congestion avoidance

- **cwnd** increases exponentially at each transmission round (not so slow after all!)



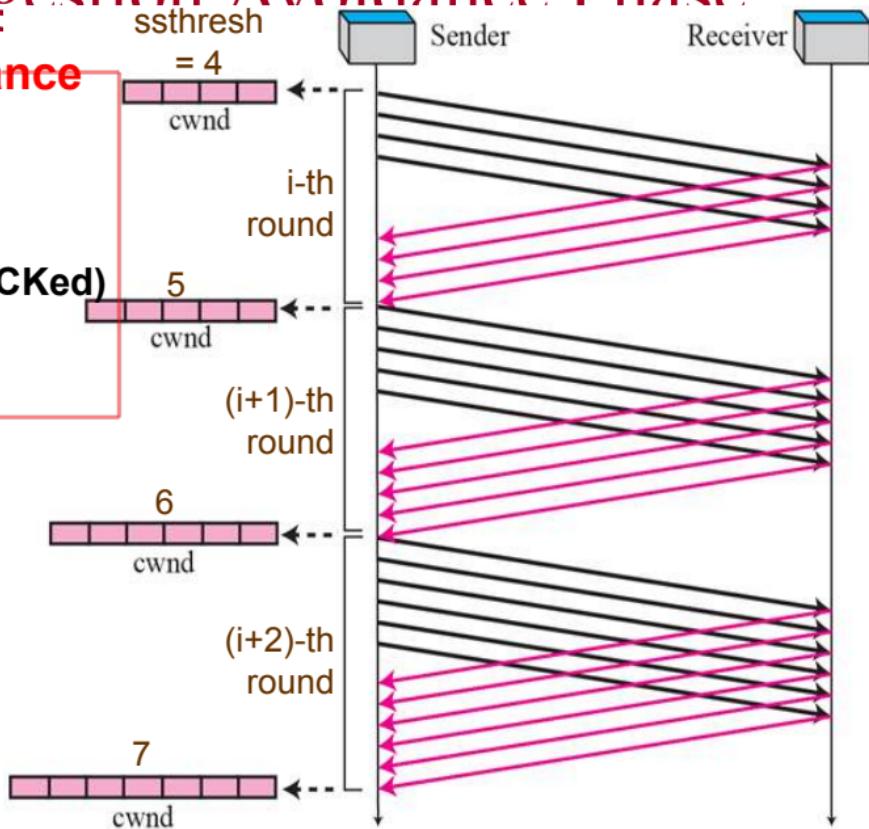
TCP Congestion Avoidance Phase

Congestion avoidance

```
/* slow start is over */  
/* cwnd ≥ ssthresh */  
do  
    for (every segment ACKed)  
        cwnd += 1 / |cwnd|  
    until (loss event)
```

Go to recovery

- In contrast, **cwnd** only increases **linearly** at each transmission round.



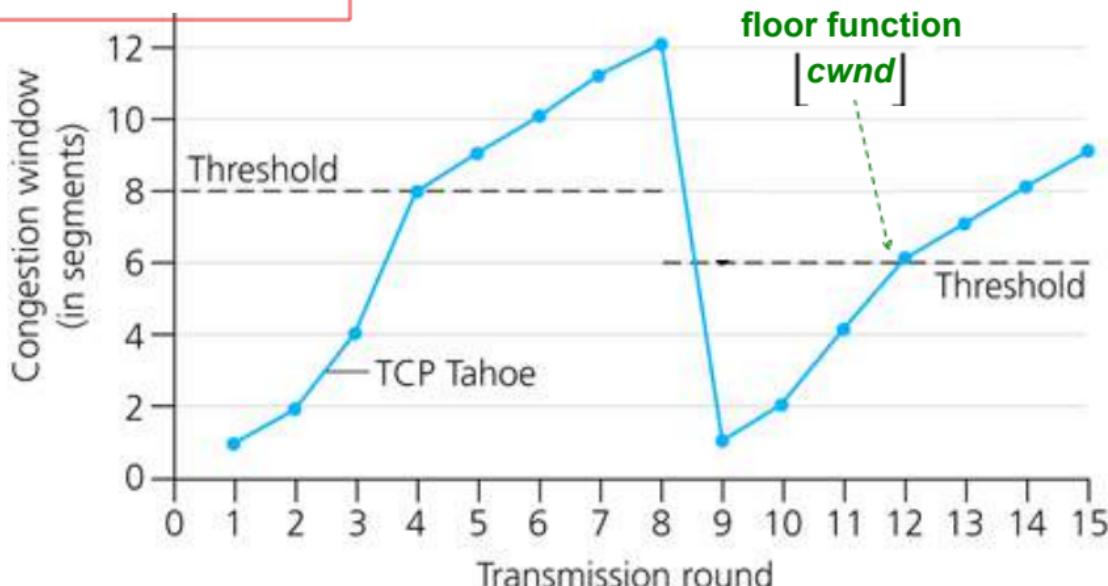
TCP Congestion Control – Tahoe Algorithm

Recovery

$$ssthresh = \lfloor cwnd/2 \rfloor$$

$$cwnd = 1$$

go back to slow start

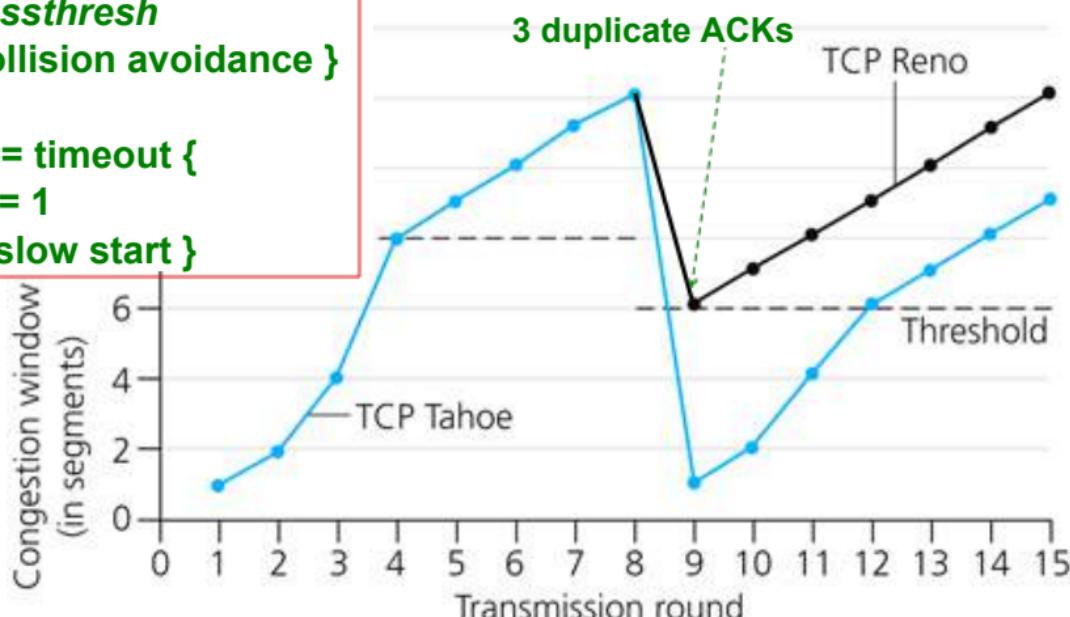


TCP Congestion Control – Reno Algorithm: Implement Fast Recovery

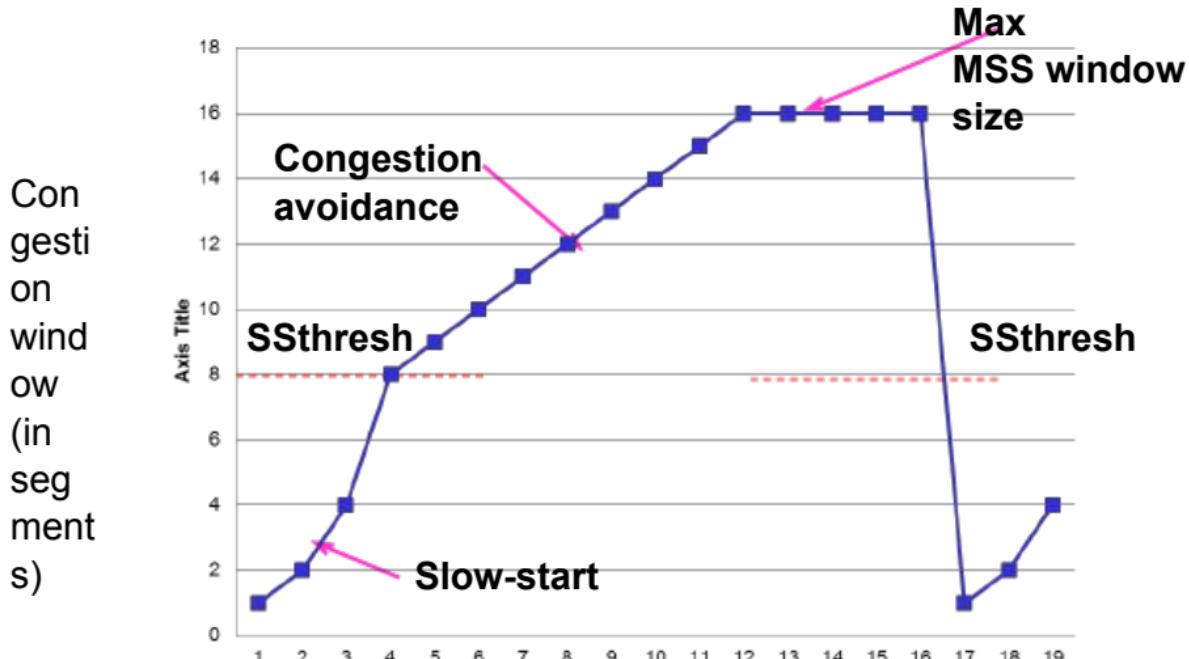
Recovery

```
ssthresh = |cwnd/2|
if loss == 3 duplicate ACKs {
    cwnd = ssthresh
    go to collision avoidance }
else
    if loss == timeout {
        cwnd = 1
        go to slow start }
```

Rationale: Network is not too congested if other segments are getting through.



TCP Congestion Control



Assume:

Transmission round

Maximum MSS window size is 16

Calculation of TCP throughput

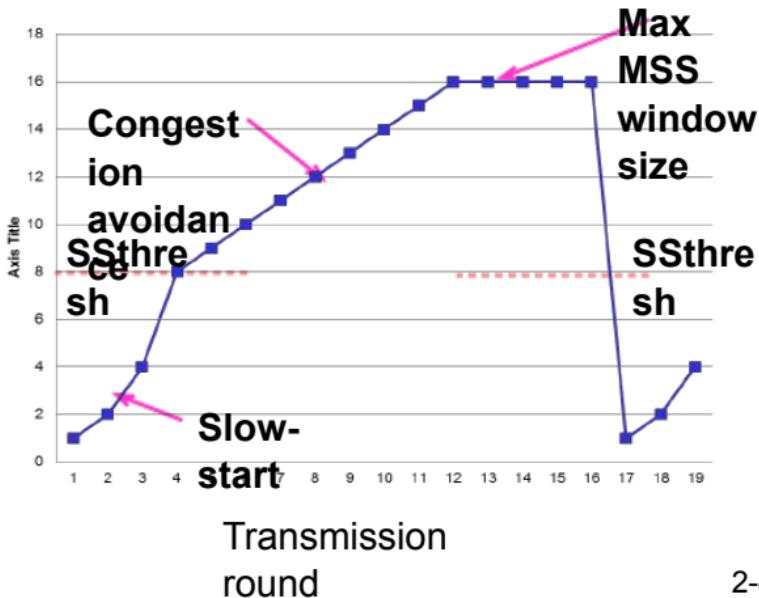
- **TCP Throughput**
- $(\text{cwnd} * \text{MSS}) / \text{RTT} = \text{throughput}$
- Assume $((\text{cwnd} * \text{MSS} * 8) / \text{Transmission rate}) \ll \text{RTT}$
- Assume no buffer constraint.

MSS = 1000 B

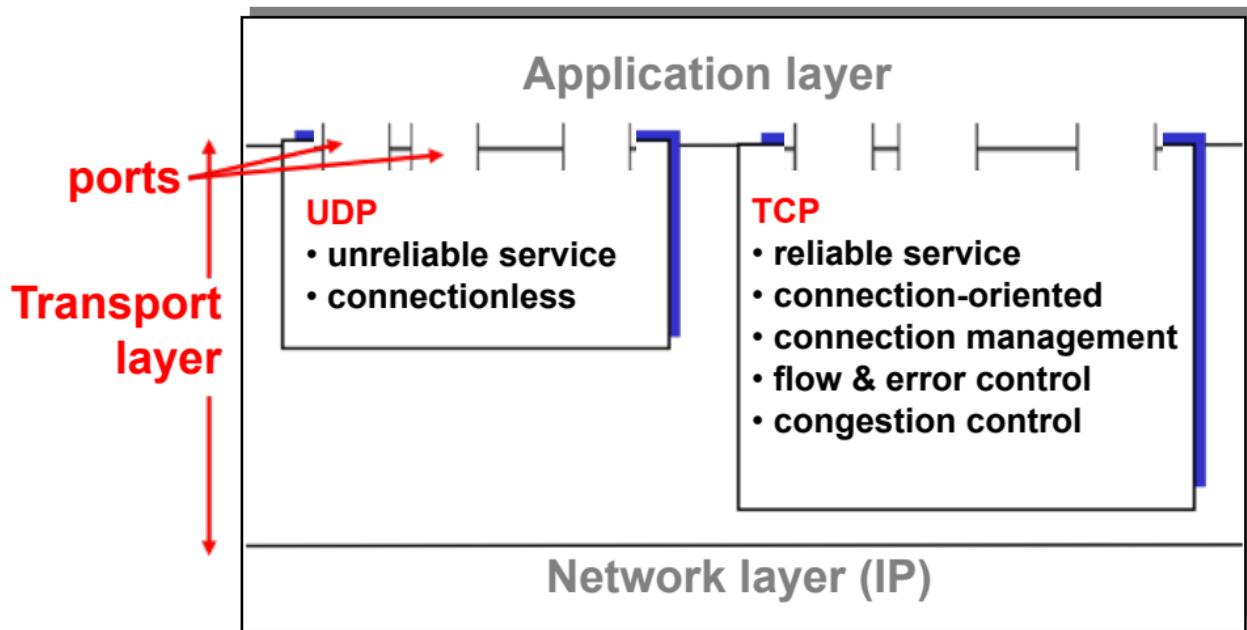
RTT = 1 second

Max throughput
 $= 16 \times \text{MSS}/1$
 $= 16\text{KB per sec.}$
 $= 128 \text{ Kbps}$

Congestion estimation window (in segments)



Summary of Transport Layer





CE3005: Computer Networks
CZ3006: Netcentric Computing

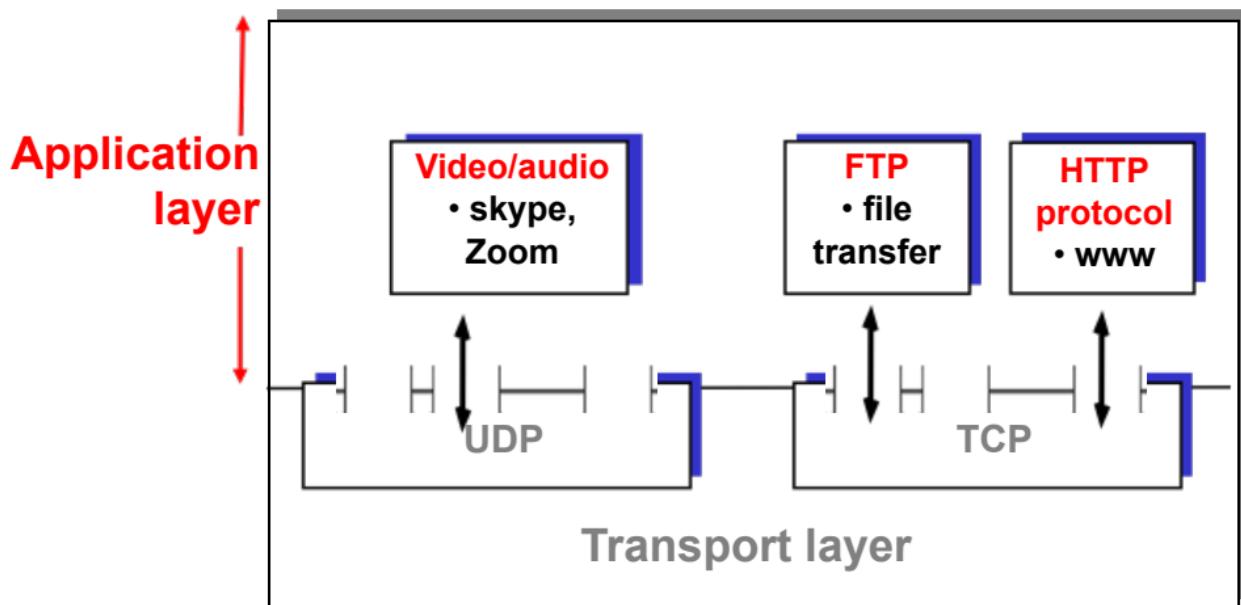
Application Layer – WWW and HTTP

Prof.Mo Li
e-mail: limo@ntu.edu.sg
Location: N4-2b-61

School of Computer Science and Engineering

Application Layer

There are many protocols/applications at the application layer.



WWW and HTTP

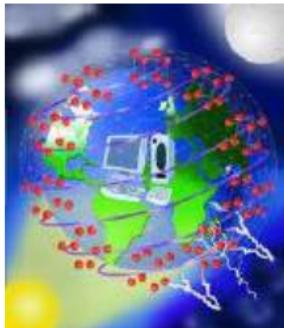
- **World Wide Web (WWW)** is simply a network application which allows a client to access hypertext file from a server – more than just text – images, sound, links, etc.
- **Origins:** Tim Berners-Lee at CERN proposed the Web in 1989
- Purpose: to allow scientists to have access to many databases of documents describing scientific work through their own computers

WWW and HTTP

- **Web or Internet?**

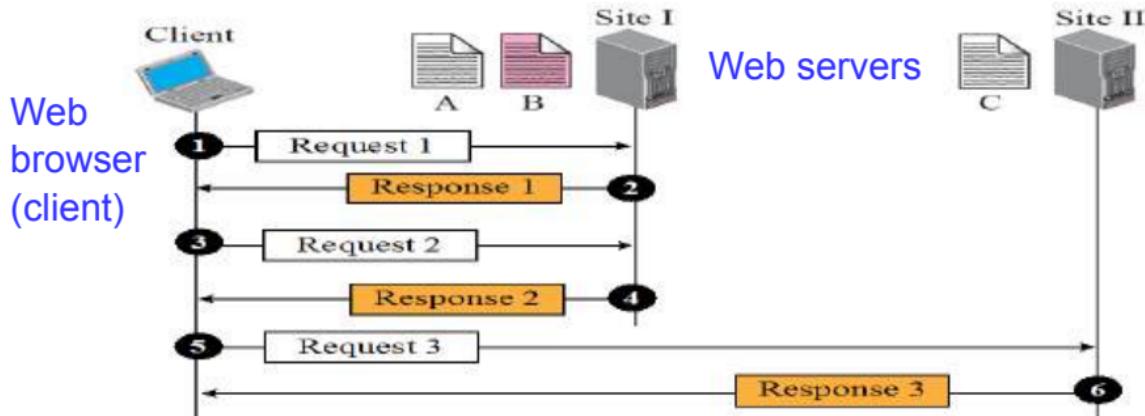
The **Internet** is a collection of **computers and other devices** connected by equipment that allows them **to communicate** with each other

The **Web** is a collection of **software and protocols** that run as **applications of Internet** (e.g., HyperText Transfer Protocol (HTTP), telnet, ftp)



WWW and HTTP

- World Wide Web (WWW) allows a client to access a file (hypertext) from a server.



- HyperText Transfer Protocol (HTTP) is the application layer protocol used by WWW. It is designed to run over TCP with server listening at well-known port 80.
- Basically, HTTP consists of request/response messages.

HTTP (Hyper Text Transfer Protocol)

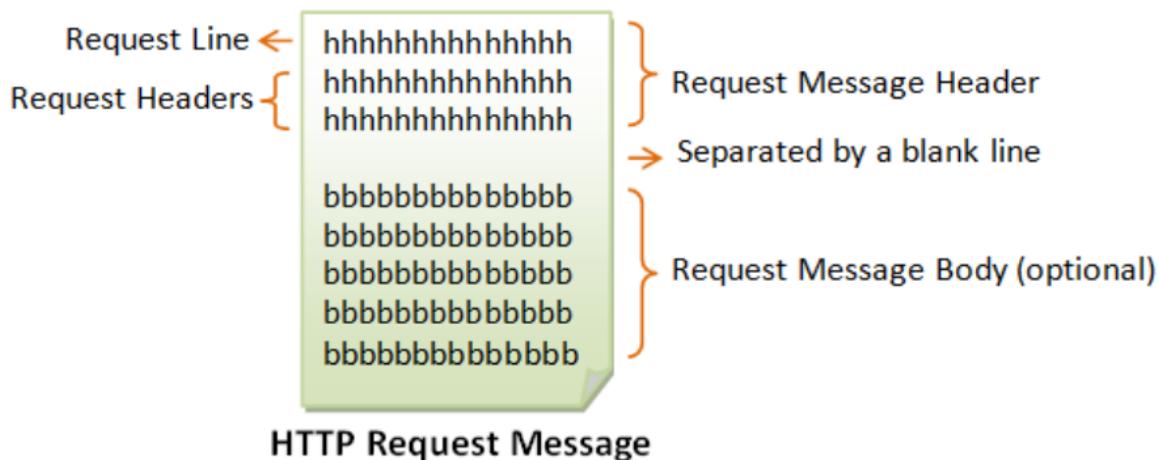
- **HTTP Request**

- Form: **HTTP-method doc-path-URL HTTP-version**

- Header fields**

- blank line**

- Message body (optional)**



HTTP (Hyper Text Transfer Protocol)

- **HTTP Request**
- Form: **HTTP-method doc-path-URL HTTP-version**
 - Header fields**
 - blank line**
 - Message body (optional)**
- An example of the first line of a request:
GET /pub/WWW/TheProject.html HTTP/1.1
- **Most commonly used methods:**
 - GET – Fetch a document**
 - POST – Execute the document, using the data in body**
(now mainly used for form data processing)
 - HEAD – Fetch just the header of the document**
 - PUT – Store a new document on the server**
 - DELETE – Remove a document from the server**

HTTP (Hyper Text Transfer Protocol)

- **Common request header fields:**
 - Accept: text/plain** (specify browser's preference for the type of the requested document)
 - Accept: text/***
 - If-Modified-Since: date** (send the requested document only if it has been modified since the given date)
 - Content-Length: number of bytes** (for POST request)
 - Content-Type: text/html** (for POST request)
 -
- **Type specifications: type/subtype**
 - E.g. **text/plain, text/html, image/gif, image/jpeg**

HTTP (Hyper Text Transfer Protocol)

```
GET /doc/test.html HTTP/1.1 → Request Line
Host: www.test101.com
Accept: image/gif, image/jpeg, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0
Content-Length: 35
→ A blank line separates header & body
bookId=12345&author=Tan+Ah+Teck → Request Message Body
```

The diagram illustrates the structure of an HTTP request message. It is divided into four main sections: Request Line, Request Headers, Request Message Header, and Request Message Body. The Request Line consists of the method (GET), the URL (/doc/test.html), and the protocol version (HTTP/1.1). The Request Headers include Host, Accept, Accept-Language, Accept-Encoding, User-Agent, and Content-Length. A blank line separates the headers from the body. The Request Message Body contains the parameters bookId and author.

HTTP (Hyper Text Transfer Protocol)

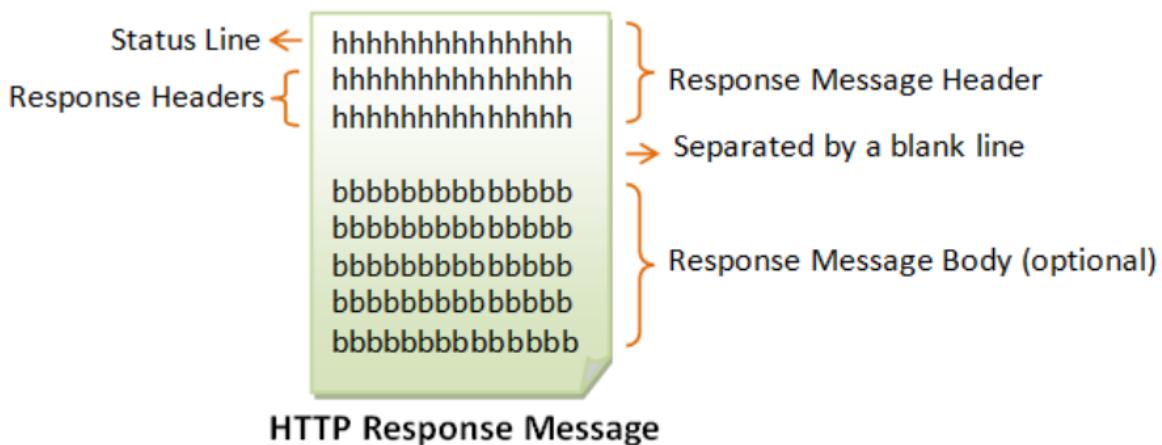
- **HTTP Response**

- **Form:** Status line

- Response header fields**

- blank line**

- Response body (optional)**



HTTP (Hyper Text Transfer Protocol)

- **HTTP Response**
- Form: Status line
 - Response header fields
 - blank line
 - Response body (optional)
- Status line format:
 - HTTP-version status-code explanation
 - Example: HTTP/1.1 200 OK
 - Status code is a three-digit number

100 Continue: The server received the request and in the process of giving the response.

200 OK: The request is fulfilled.

301 Move Permanently: The resource requested for has been permanently moved to a new location. The URL of the new location is given in the response header called **Location**. The client should issue a new request to the new location.

302 Found & Redirect (or Move Temporarily): Same as 301, but the new location is temporarily in nature. The client should issue a new request, but applications need not update the references.

304 Not Modified: In response to the **If-Modified-Since** conditional GET request, the server notifies that the resource requested has not been modified.

400 Bad Request: Server could not interpret or understand the request, probably syntax error in the request message.

401 Authentication Required: The requested resource is protected, and require client's credential (username/password).

403 Forbidden: Server refuses to supply the resource, regardless of identity of client.

404 Not Found: The requested resource cannot be found in the server.

405 Method Not Allowed: The request method used, e.g., POST, PUT, DELETE, is a valid method. However, the server does not allow that method for the resource requested.

408 Request Timeout:

414 Request URI too Large:

500 Internal Server Error: Server is confused, often caused by an error in the server-side program responding to the request.

501 Method Not Implemented: The request method used is invalid (could be caused by a typing error, e.g., "GET" misspell as "Get").

502 Bad Gateway: Gateway indicates that it receives a bad response from the upstream server.

503 Service Unavailable: Server cannot response due to overloading or maintenance. The client can try again later.

504 Gateway Timeout: Gateway indicates that it receives a timeout from an upstream server.

HTTP (Hyper Text Transfer Protocol)

HTTP/1.1 200 OK

Date: Sun, 08 Feb xxxx 01:11:12 GMT
Server: Apache/1.3.29 (Win32)
Last-Modified: Sat, 07 Feb xxxx
ETag: "0-23-4024c3a5"
Accept-Ranges: bytes
Content-Length: 35
Connection: close
Content-Type: text/html

<h1>My Home page</h1>

Status Line

Response Headers

Response Message Header

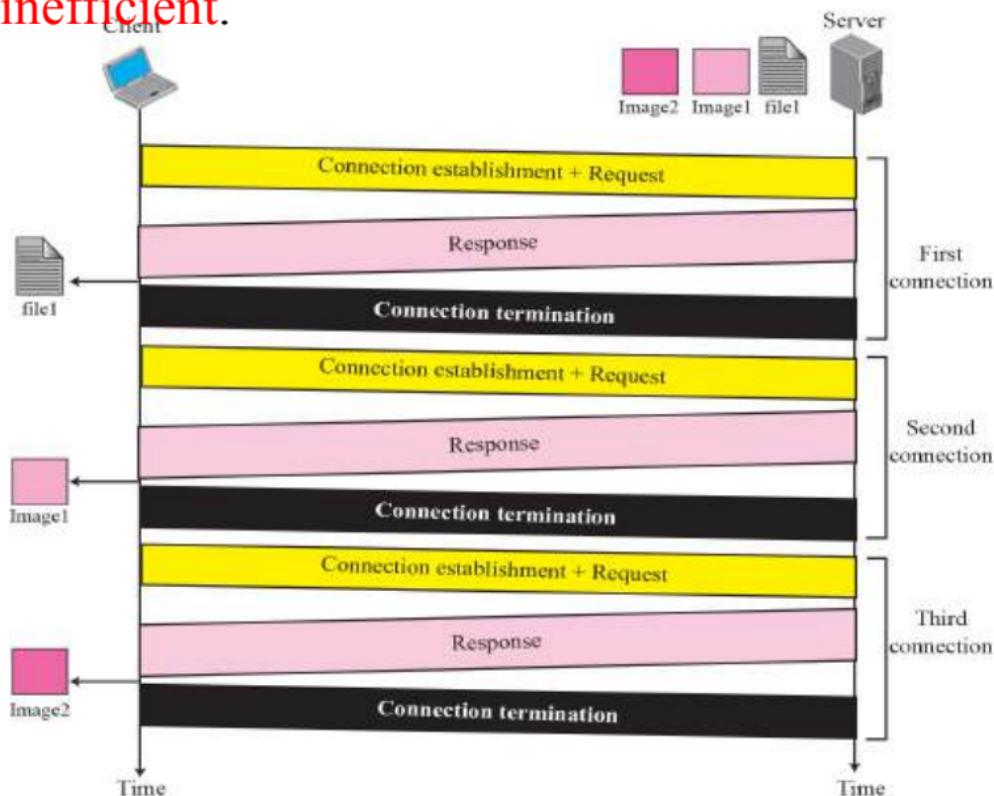
A blank line separates header & body

Response Message Body

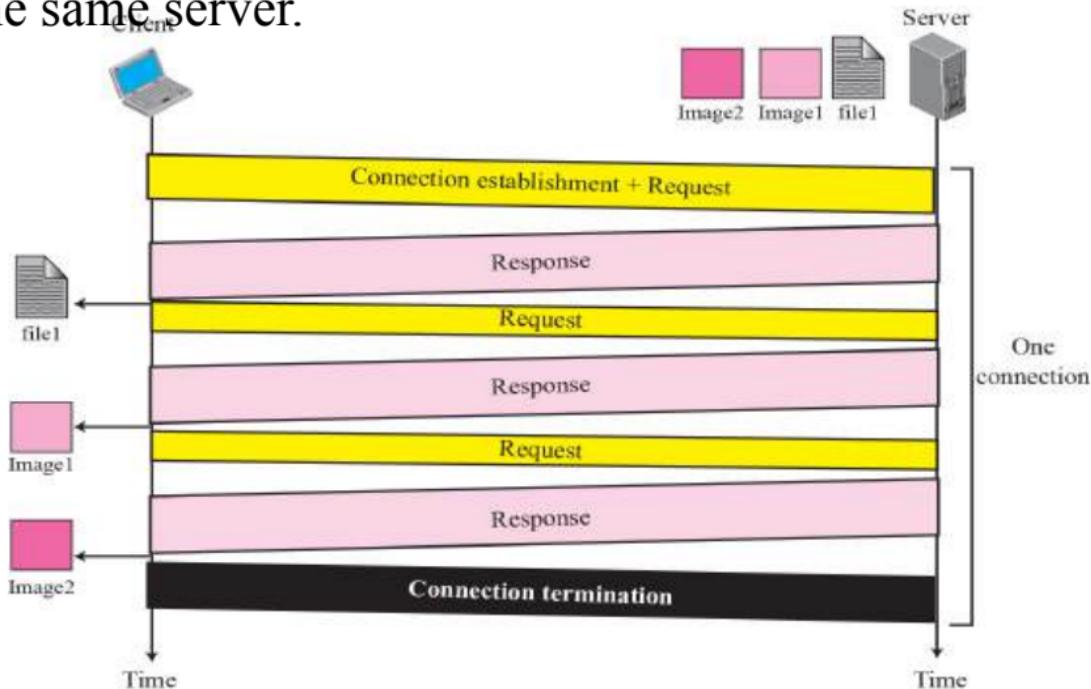
```
</div>  
  
<div id="wrapper" class="clearfix" style="height:auto;">  
    <div id="footer">  
        <div>  
  
            <ul id="fooSecond">  
                <li>50 Nanyang Avenue</li>  
                <li>Singapore 639798</li>  
                <li>Tel: (65) 67911744</li>  
            </ul>  
            <ul id="fooThird">  
                <li><a href="/AboutNTU/contactntu/Pages/university.aspx">University</a></li>  
                <li><a href="/Transportation/Pages/GettingHere.aspx">Getting Here</a></li>  
                <li><a href="/Transportation/Pages/GettingThere.aspx">Getting There</a></li>  
                <li><a href="http://blogs.ntu.edu.sg/">Blogs</a></li>  
            </ul>  
            <ul id="fooFourth">  
                <li><a href="https://wits.ntu.edu.sg/webex81">WebEx</a></li>  
            </ul>  
        </div>  
    </div>  
</div>
```



Non-persistent HTTP: individual TCP connection/termination for each pair of request/response to access one file – **inefficient**.



Persistent HTTP: multiple request/response messages within one TCP connection – **efficient** for accessing multiple files in the same server.



Implementing a simple Web Client:

```
import java.net.*;
import java.io.*;
import javax.swing.*;

class WebClient
{
    public static void main(String args[])
    {
        String webServer = "www.ntu.edu.sg";
        int httpPort = 80;
        Socket sock;

        try
        {
            // Opening a connection to the web server
            sock = new Socket(webServer, httpPort);

            // Constructing HTTP request header
            StringBuffer strToSend = new StringBuffer("");
            strToSend.append("GET / HTTP/1.1\r\n");
            strToSend.append("Host: "+webServer+"\r\n");
            strToSend.append("Connection: close\r\n");
            strToSend.append("\r\n");

            // Sending HTTP request
            System.out.println("Sending a GET request ... \n");
            byte [] outByte = strToSend.toString().getBytes();
            sock.getOutputStream().write(outByte);
        }
    }
}
```

JAVA Socket
Programming

Constructing
HTTP header

Sending HTTP
request

Implementing a simple Web Client (cont)

```
// Receiving HTTP response
byte inByte[] = new byte[1024];
StringBuffer inBuffer = new StringBuffer("");
while(true)
{
    int numBytes = sock.getInputStream().read(inByte,0,1024);
    if (numBytes<0) break;

    // Storing receiving texts into a buffer
    String str = new String(inByte,0,numBytes);
    inBuffer.append(str);
}

// Removing HTTP response header
int idx = inBuffer.indexOf("\r\n\r\n");
String htmlStr = inBuffer.substring(idx+4);

// Display HTML page received
Jframe frame = new Jframe("SimpleWebClient");
frame.setDefaultCloseOperation(WindowConstants.DISPOSE_ON_CLOSE);
JEditorPane pane = new JEditorPane("text/html", htmlStr);
frame.getContentPane().add(new ScrollPane(pane));
frame.setSize(600,400);
frame.setVisible(true);
}
catch(Exception e) {};
}
```

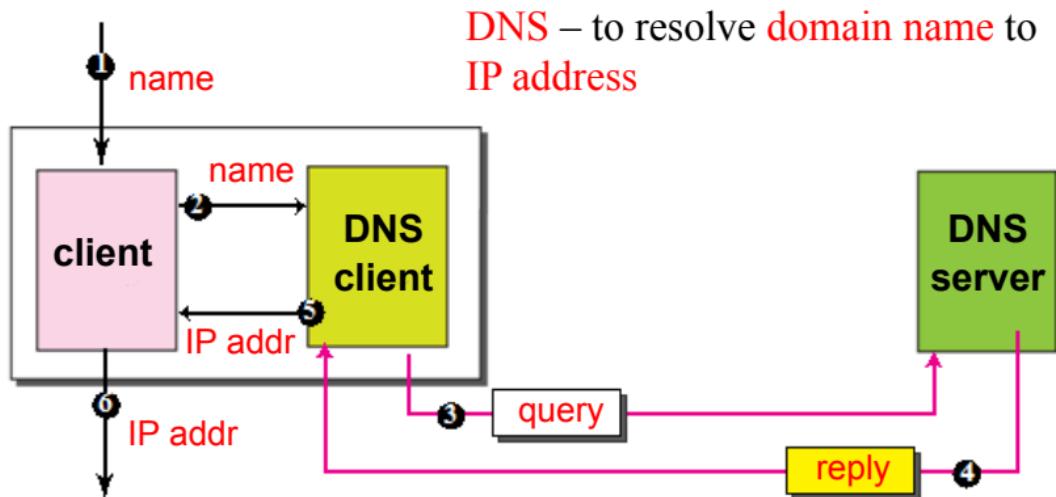
Receiving HTTP response

Removing HTTP header (should interpret accordingly)

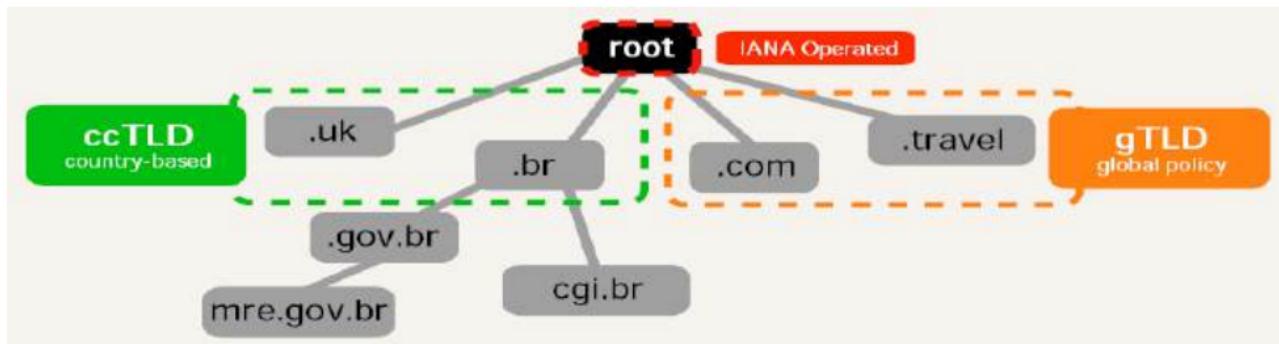
Domain Name System (DNS)

Given only the domain name of a server, how does a client know the IP address to send to destination?

e.g, www.ntu.edu.sg



- DNS protocol is designed to run over UDP with server listening at well-known port 53.



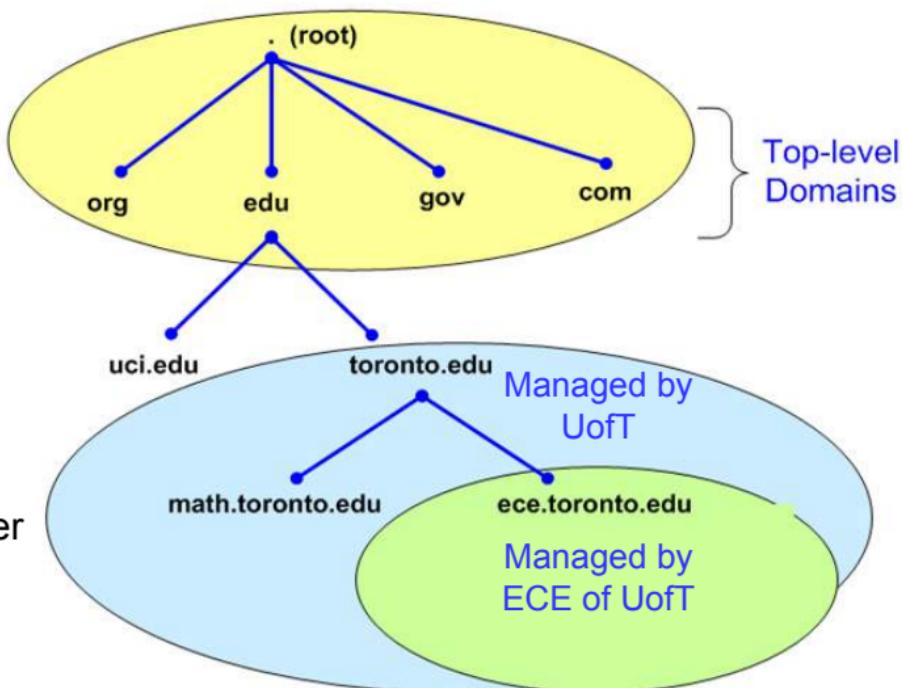
Domain names are divided into **gTLDs** and **ccTLDs**, and commercial domain name registrars are accredited to sell them:

- generic Top-Level Domains (**gTLDs**): only **IANA/ICANN-accredited registrars** are able to sell domain names under **gTLDs**
- country-code Top-Level Domains (**ccTLDs**): delegated to respective countries, e.g. only (Singapore) **SGNIC-accredited registrars** can sell domain names under **.sg**

For scalability, **domain names** are designed to be **hierarchical**; e.g. ece.toronto.edu

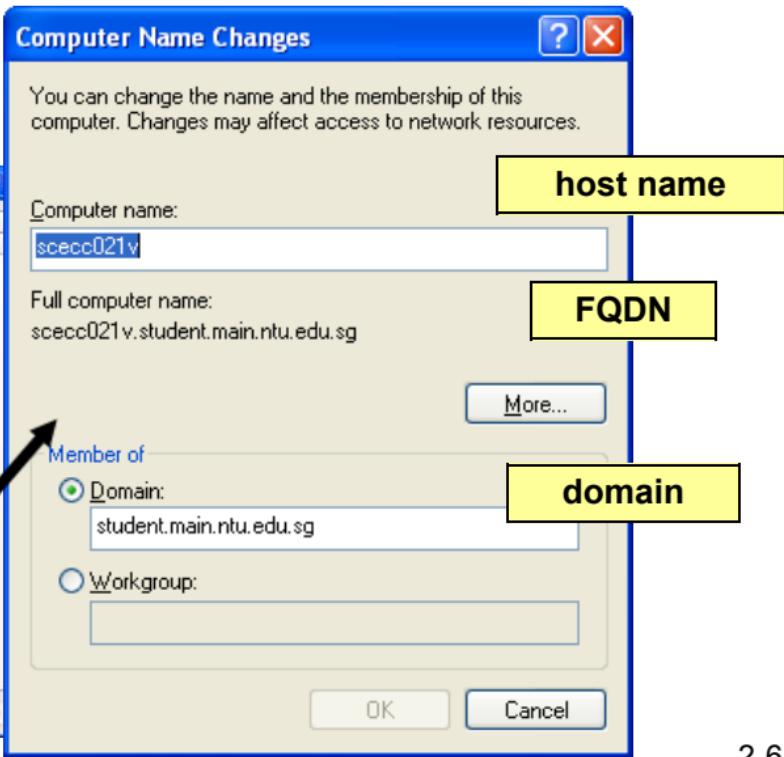
3rd level 2nd level top level

- Top-level domains are managed by IANA
- Below top-level domains, management of name space is delegated to respective organizations
- Each organization can delegate further



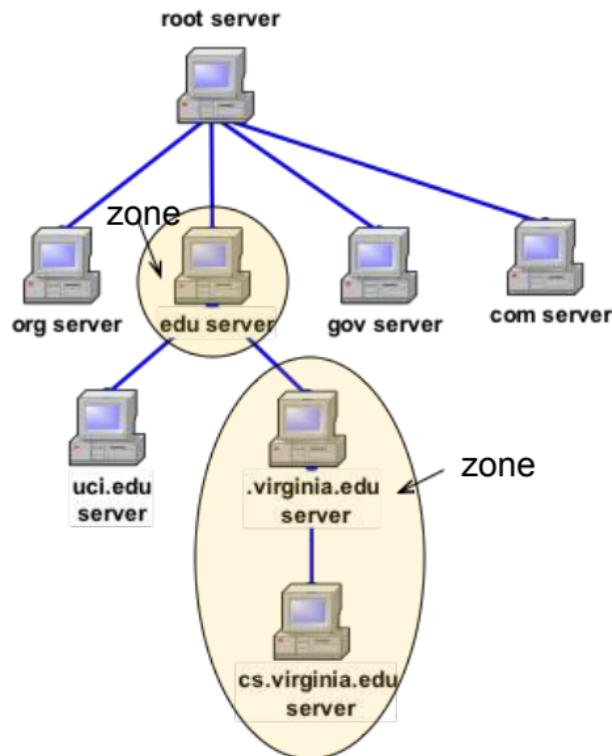
A **fully qualified domain name (FQDN)** is a completely specified domain name consisting of a host name and a domain.

Eg. configuring domain name in Windows:



Following the hierarchy of domain names, a **hierarchy** of **name servers** are set up to provide DNS services.

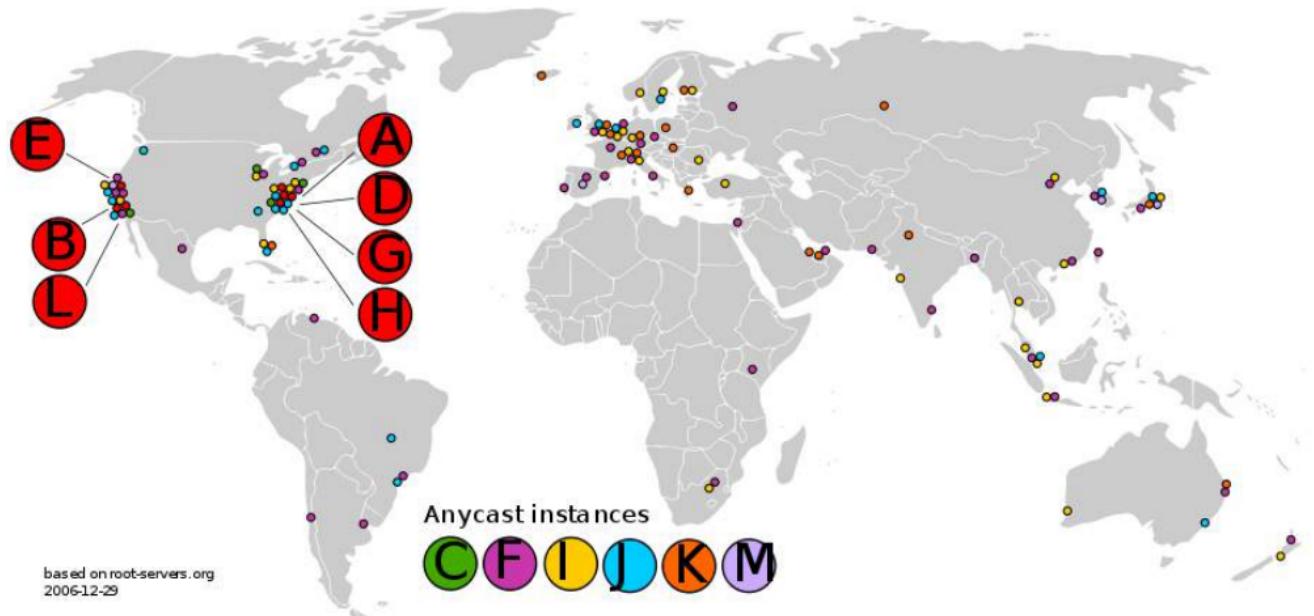
- Each server is responsible (**authoritative**) for a **zone** of the DNS namespace.
- A **zone** can be **a node**; e.g. **edu server** is authoritative for ***xxx.edu***
- A **zone** can also consist of **multiple nodes**; e.g. **virginia.edu server** is authoritative for ***xxx.virginia.edu***, including ***xxx.cs.virginia.edu***



To be fault tolerant, there are **13 root name servers** which are configured to know the authoritative servers for TLDs.

<u>Domain Name</u>	<u>Operator</u>	<u>IP Address</u>
a.root-servers.net	VeriSign	198.41.0.4
b.root-servers.net	USC-ISI	192.228.79.201
c.root-servers.net	Cogent Communications	192.33.4.12
d.root-servers.net	University of Maryland	128.8.10.90
e.root-servers.net	NASA	192.203.230.10
f.root-servers.net	Internet Systems Consortium	192.5.5.241
g.root-servers.net	US DoD	192.112.36.4
h.root-servers.net	US Army Research Lab	128.63.2.53
i.root-servers.net	Autonomica Stockholm	192.36.148.17
j.root-servers.net	VeriSign	192.58.128.30
k.root-servers.net	RIPE London	193.0.14.129
l.root-servers.net	ICANN Los Angeles	199.7.83.42
m.root-servers.net	WIDE Tokyo	202.12.27.33

In reality, there are more than 13 physical root name servers through the use of **anycast**.



Anycast – a group of servers are identified by the same IP address, and packets are routed to the nearest servers

Now, we are ready to understand how it works for internet applications:

To reach an Internet resource, we need to specify:

1. Method/Protocol used (**application layer**)
2. Host using IP address or domain name (**network layer**)
3. Port number (or none if using default well-known port) (**transport layer**)
4. Path and document name (application layer)

In practice, all above are concatenated into a single string called Uniform Resource Locator (URL)



For example:

<http://www.prenhall.com/reed/index.html>

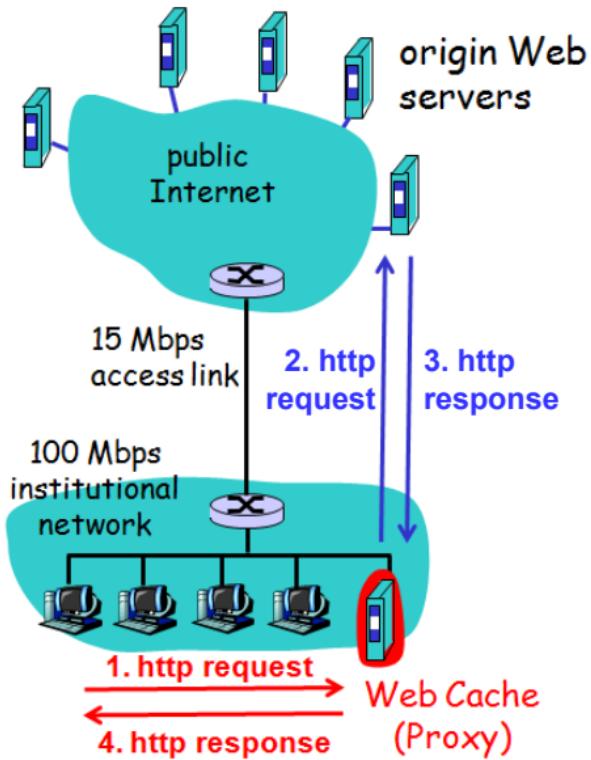
Web Proxy (Cache)

Why Web Proxy?

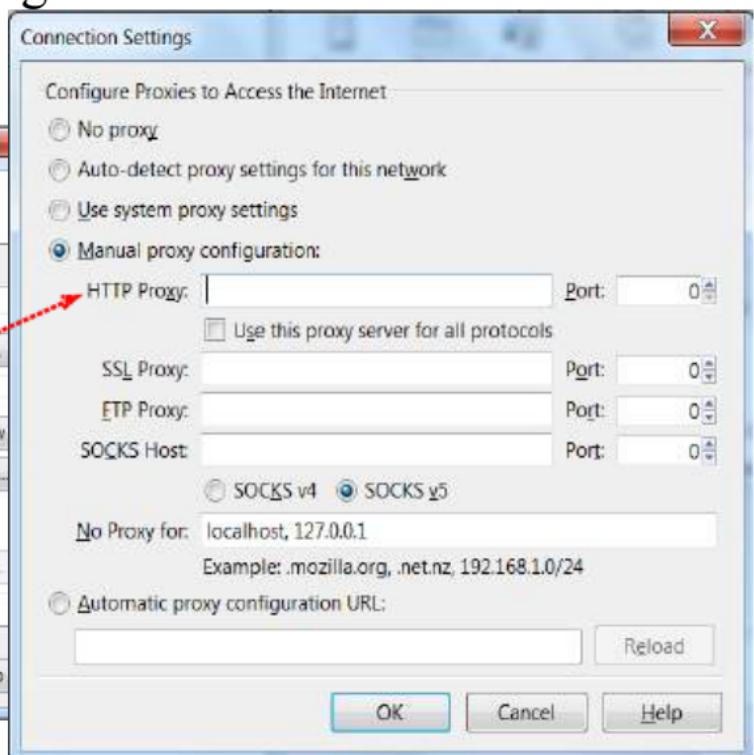
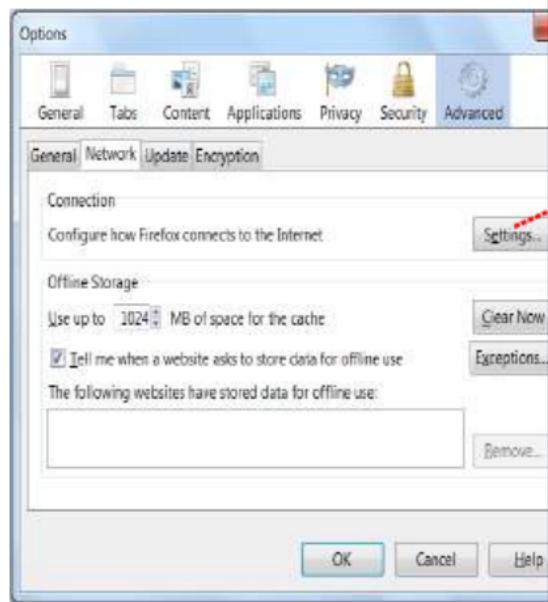
- Improve performance by caching
- Reduce traffic load on costly access link
- Monitor/Filter contents

How Web Proxy Works?

1. Client requests to proxy
2. (if content not available) proxy requests to origin server
3. Origin server responds to proxy
4. Proxy responds to client

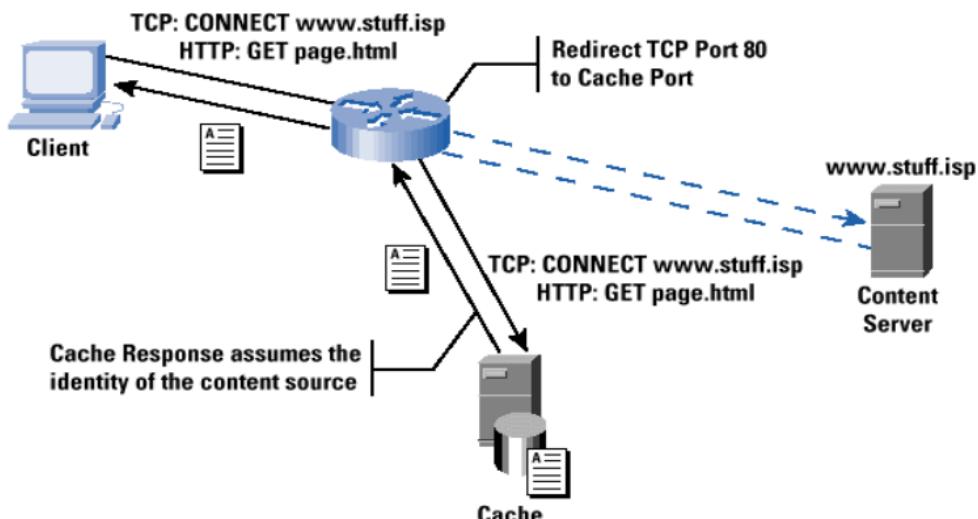


Traditionally, **Web proxy** is implemented by requiring users to **explicitly** configure their browsers.

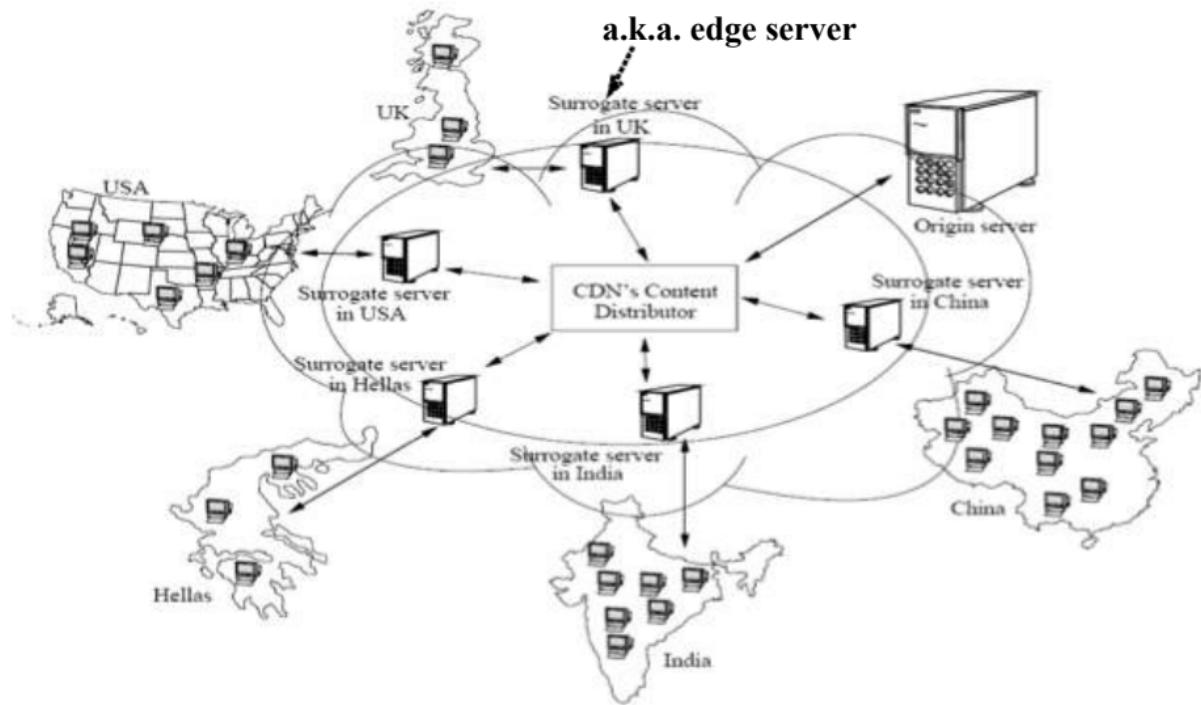


However, note that **Web proxy** can be implemented **transparently** without the knowledge of users/servers!

Basically, the organization/ISP configures its routers to **intercept** all Web traffic and **re-direct** (**mis-direct**) them to its Web proxy, which **masquerades** as the destination server!



Alternatively, if performance is important, an option for content provider is to use the service of **Content Delivery/Distribution Network (CDN)**.



For example, one of the world's largest CDN is Akamai network.

The Akamai CDN (2011):

85,000+
Servers

950+
Networks

660+
Cities

70+
Countries

Resulting in traffic of:

5.4 petabytes / day

790+ billion hits / day

436+ million unique clients IPs / day



Finally, a review of what we have covered in this course:

