

Data Communication I

Lecture 2

presented by

LONG Teklim

IIC, Phnom Penh

Mobile: 012 29 79 28

Contact: Longteklm@gmail.com

Literature (excerpt)

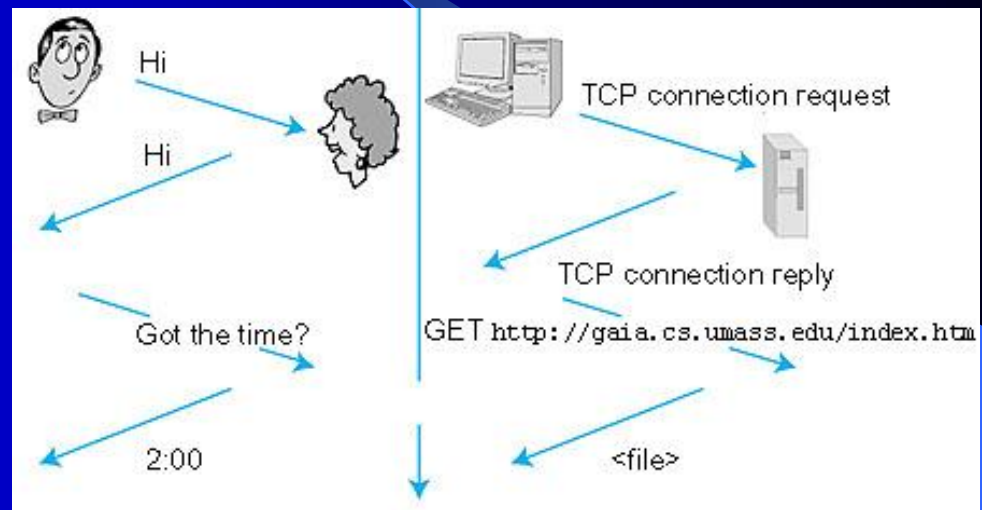
- William Stallings: Data and Computer Communications
- William J. Beyda: Data Communications – From Basics to Broadband
- Craig Hunt: TCP/IP Network Administration (O'Reilly)
- Monson-Haefel: Java Messaging Service
- Aaron Skonnard: Essential XML Quick Reference
- David A. Chappel: Java Web Services (O'Reilly)
- Kurose: Computer Networking: A Top-Down Approach Featuring the Internet

Today's Lecture Plan

- Complete “feel” of a network
- Protocols
- The Protocol Stack (A set of network protocol layers that work together. The seven protocol layers is often called a stack, as is the set of TCP/IP protocols that define communication over the internet.
- The term *stack* also refers to the actual software that processes the protocols. So, for example, programmers sometimes talk about *loading a stack*, which means to load the software required to use a specific set of protocols. Another common phrase is *binding a stack*, which refers to linking a set of network protocols to a network interface card (NIC). Every NIC must have at least one stack bound to it.)
- Key Performance Measures

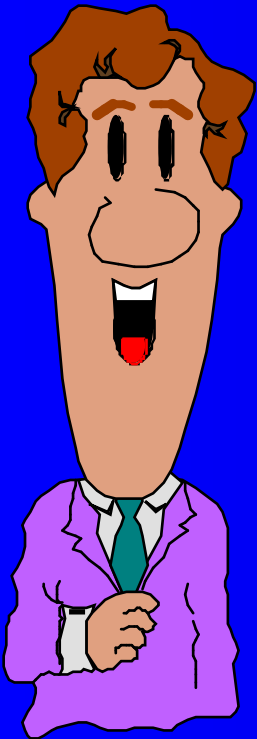
Protocol Review

- Defines acceptable behaviour:
- Agreed data format
 - Handshake
 - Message
 - Response



- *A **protocol** defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event.*

Protocols - A “Human” Example



Do you have 2 seats for “Blue Men”?

yes

In the front?

yes

How much?

\$100 each

Here is my Visa Card no.

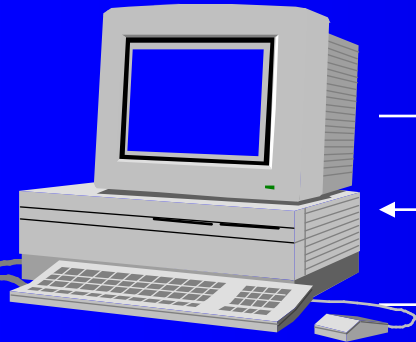
Ok, your tickets can be picked up.

Thank you.

Goodbye.



Protocols - Email



HELLO "oetzi.unibz.it"

250 OK

MAIL FROM:<rmessner@unibz.it>

250 OK

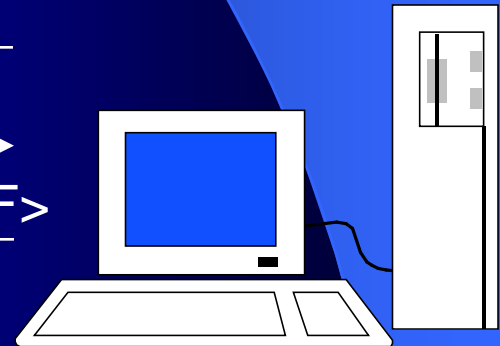
RCPT TO:<bgates@microsoft.com>

250 OK

DATA

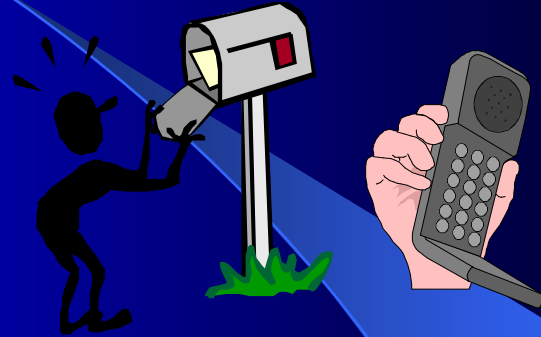
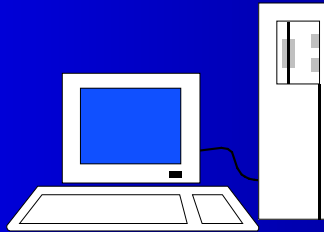
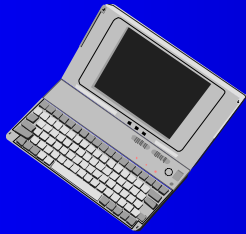
354 Start mail input; end with <CRLF>.<CRLF>

Blah, blah, blah



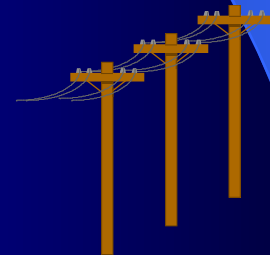
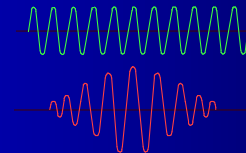
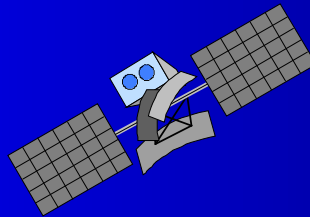
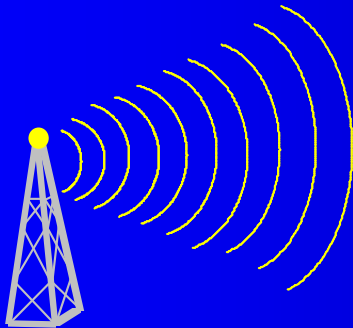
Protocol Layers

applications



transport protocol

technology



Why Protocol Layers?

- To deliver the email
 - Application messages
 - Connection to remote host/socket
 - Reliability* of delivery of messages
 - Routing of connection/messages
 - Detection of errors in frames
- Functions assigned to appropriate layer

* In general, **reliability** (systemic def.) is the ability of a person or system to perform and maintain its functions in routine circumstances, as well as hostile or unexpected circumstances.

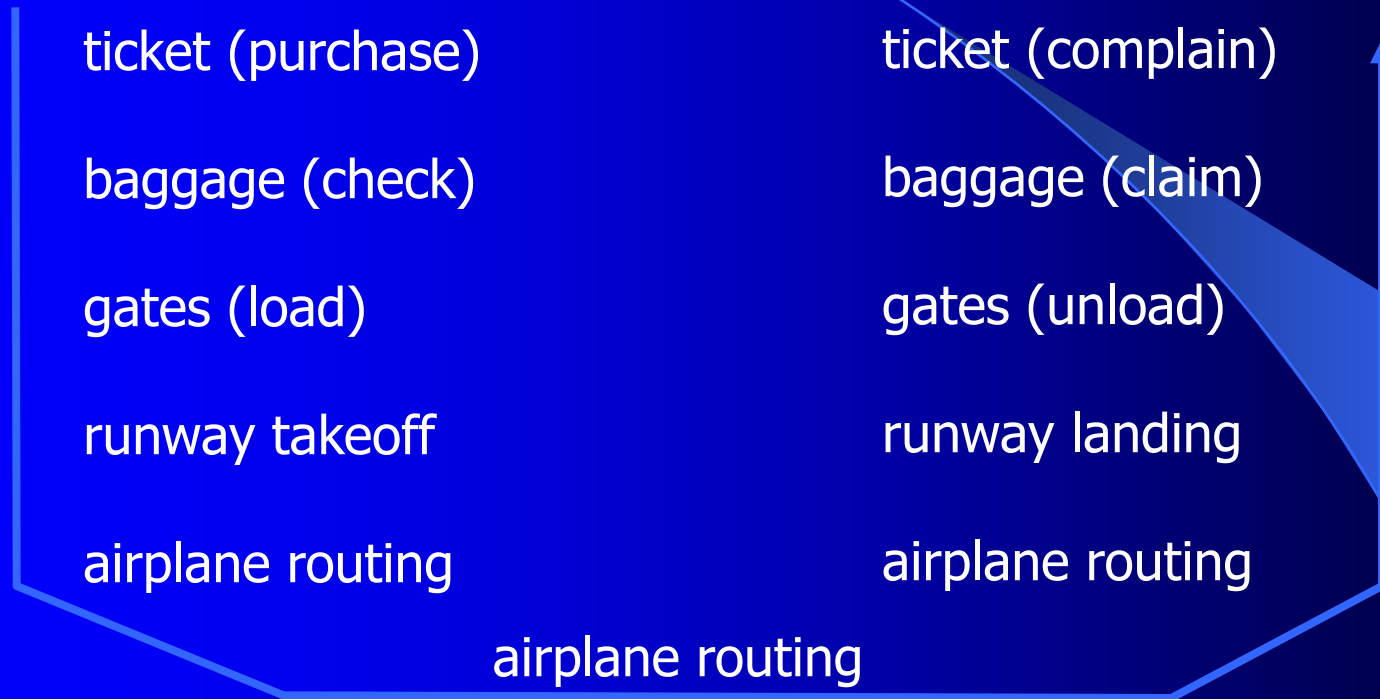
Layers = Abstraction

Networks are complex with many “pieces”:

- hosts
- Routers*
- links of various media
- applications
- protocols
- hardware, software

* A **router** distributes Digital computer information that is contained within a data packet. Each data packet contains address information that a router can use to determine if the source and destination are on the same network

Organisation of air travel



- a series of steps

Organisation of air travel II

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing
airplane routing	

Layers: each layer implements a service

- via its own internal-layer actions
- relies on services provided by layer below

By Student C.Sokna

Layered air travel: services

Counter-to-counter delivery of person+bags

baggage-claim-to-baggage-claim delivery

people transfer: loading gate to arrival gate

runway-to-runway delivery of plane

airplane routing from source to destination

Distributed implementation of layer functionality

Departing airport

ticket (purchase)

baggage (check)

gates (load)

runway takeoff

airplane routing

ticket (complain)

baggage (claim)

gates (unload)

runway landing

airplane routing

arriving airport

intermediate air traffic sites

airplane routing

airplane routing

airplane routing

By Student C.Sokna

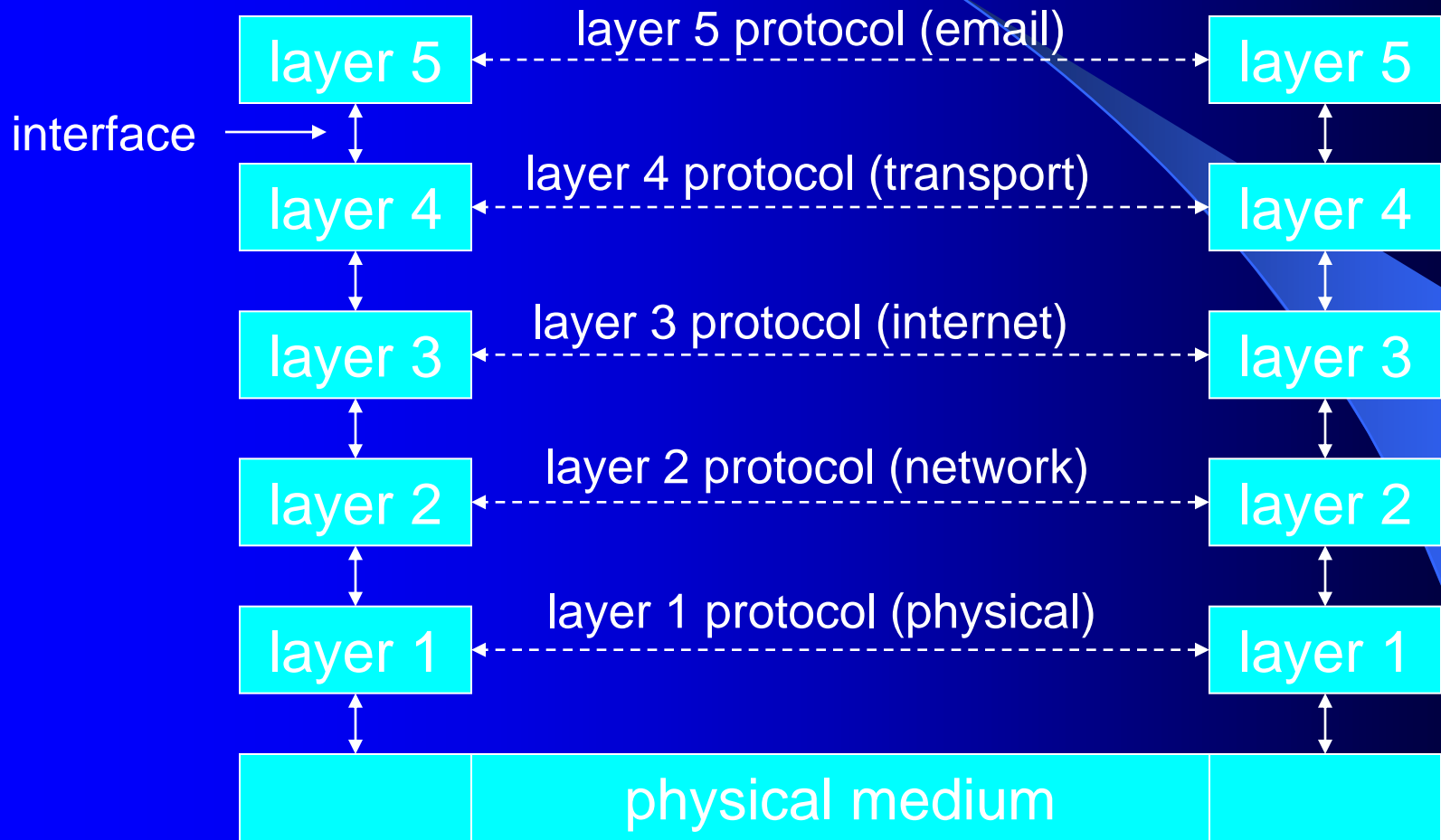
Why layering?

Dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
- modularity eases maintenance, updating of system
- change of implementation of a layer is transparent to rest of system
 - e.g., change in gate procedure doesn't affect rest of system

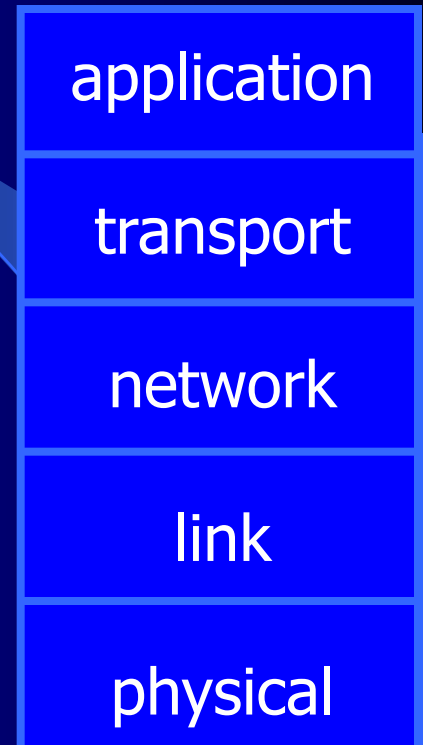
Is layering considered harmful?

Example Reference Model

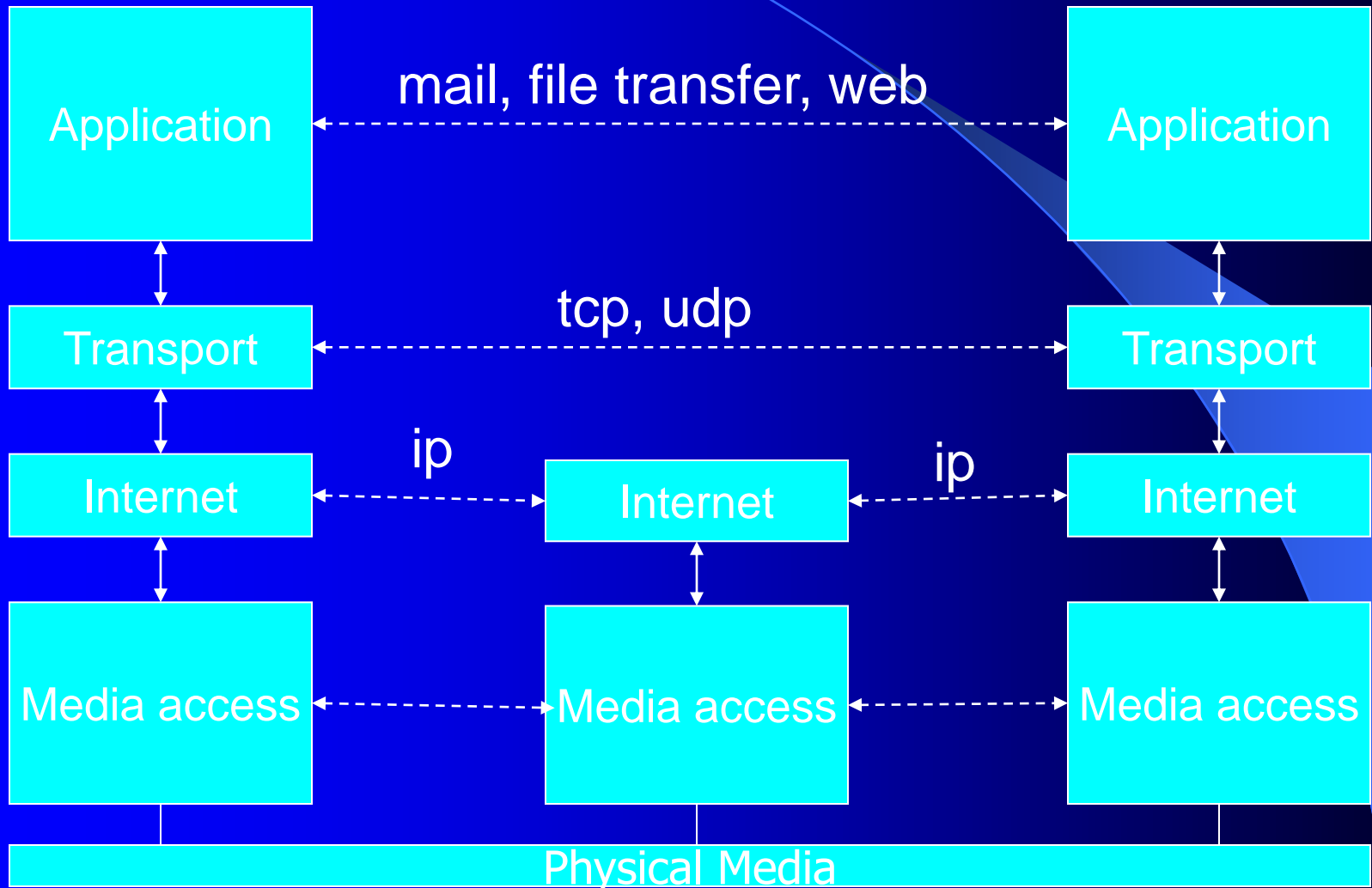


Internet protocol stack

- **application:** supporting netapps
 - ftp, smtp*, http
 - * **Simple Mail Transfer Protocol (SMTP)** is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks.
- **transport:** end host- end host
 - tcp, udp
- **network:** source to destination
 - ip, routing protocols
- **link:** neighboring network elements
 - ppp, ethernet
- **physical:** bits “on the wire”



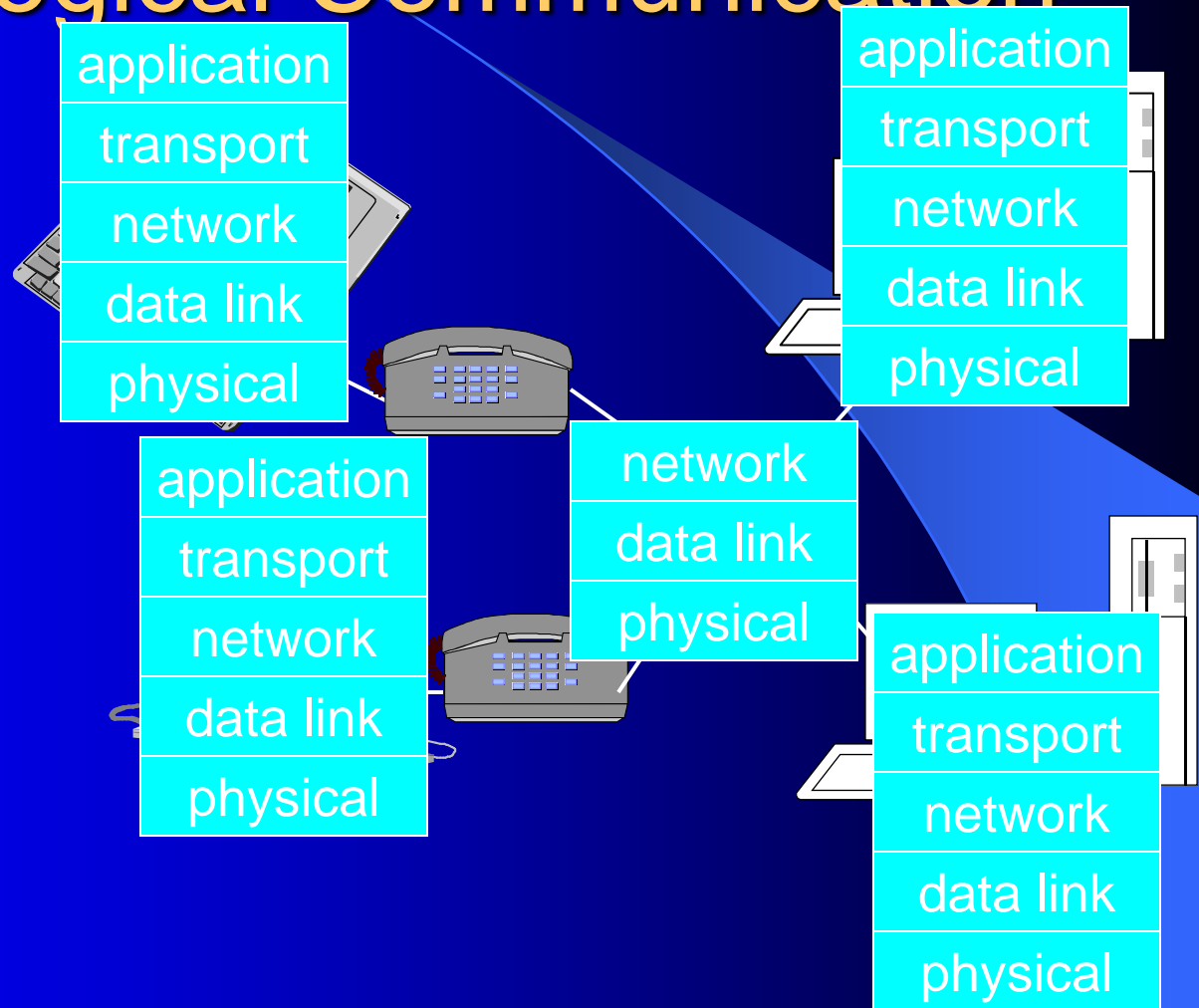
The TCP/IP Layers



Layers: Logical Communication

Each Layer

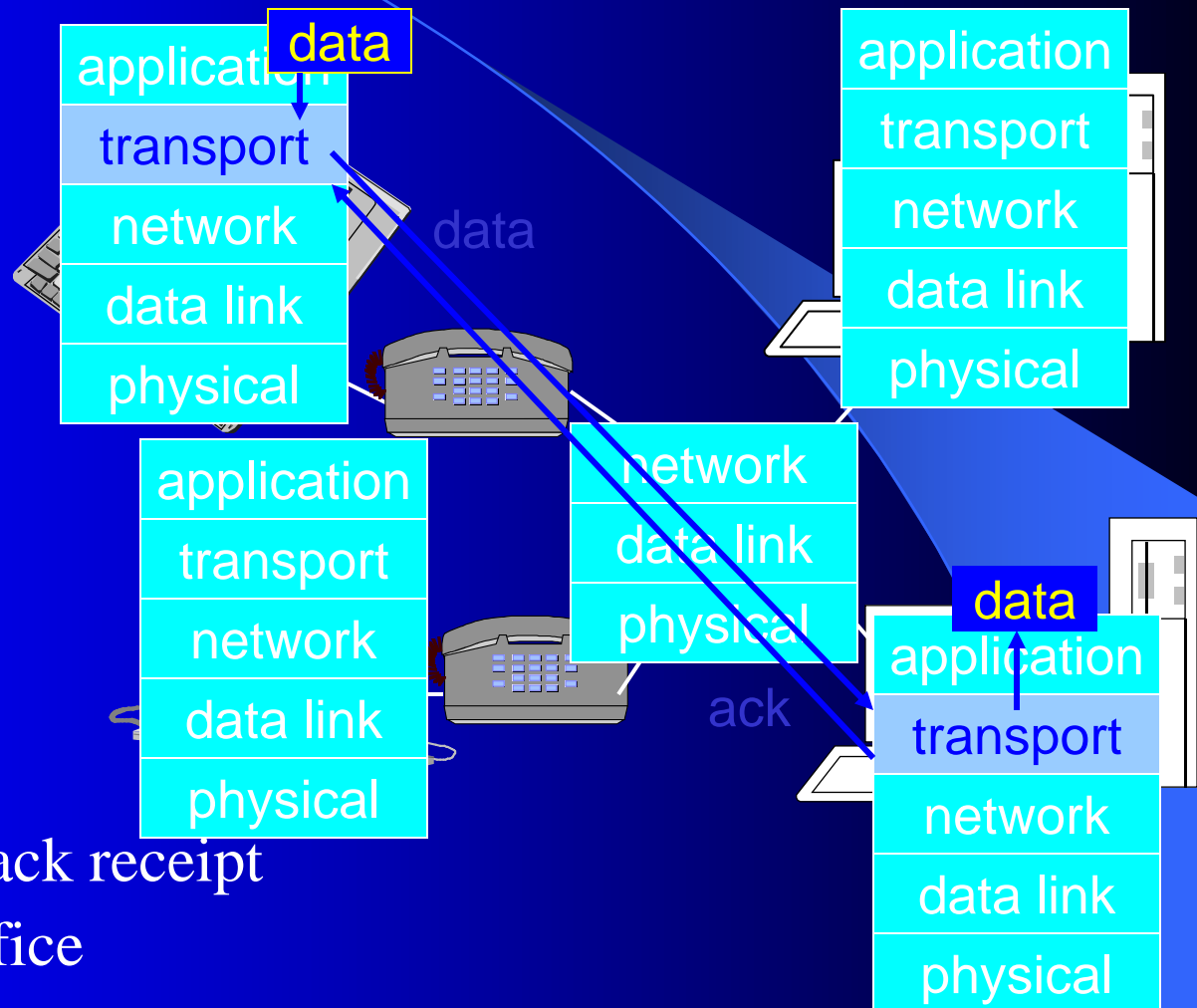
- distributed
- “entities” implement layer functions at each node
- entities perform actions, exchange messages with peers



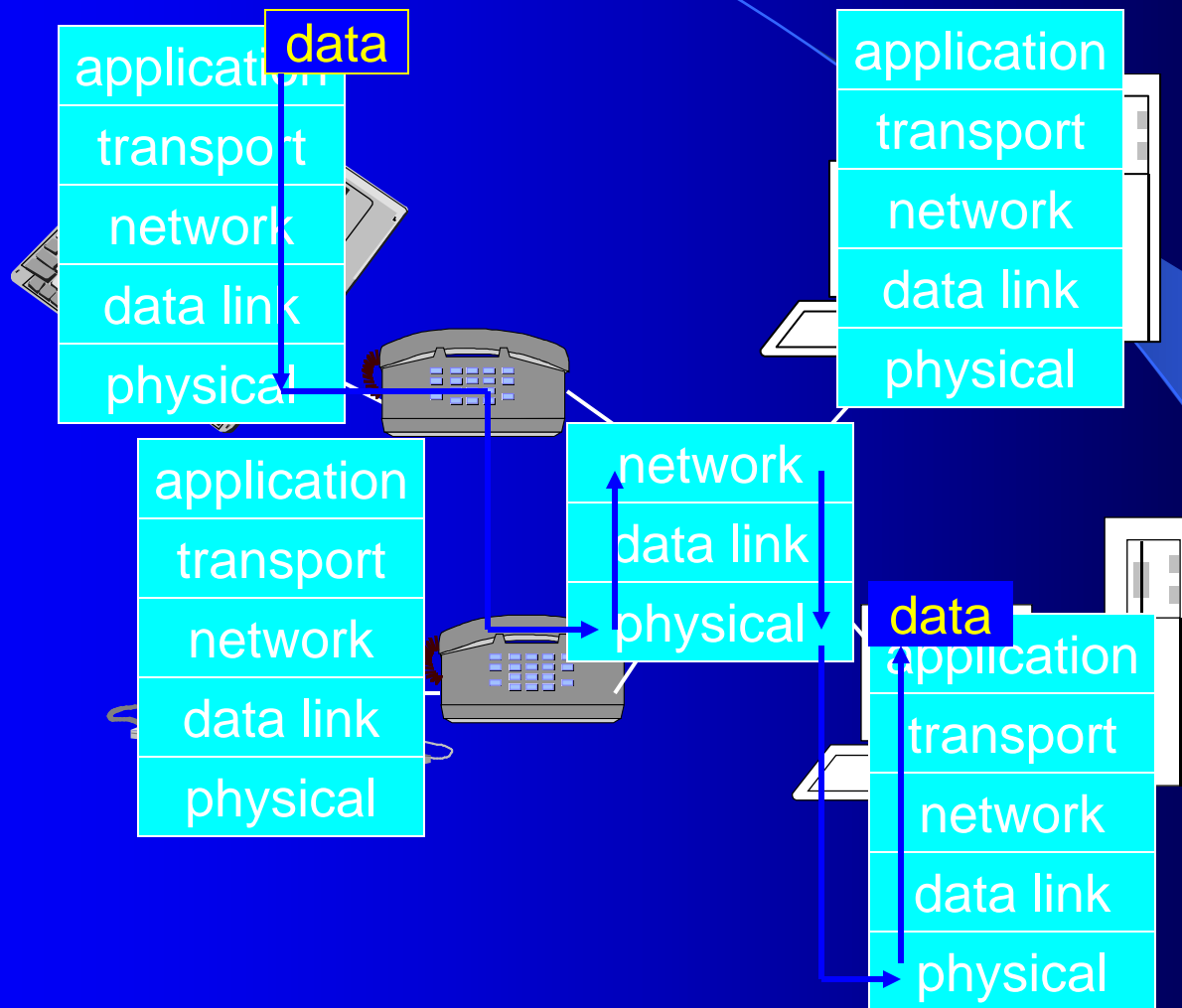
Layers: Logical Communication

E.g. transport

- take data from app
- add addressing, reliability check info to form “datagram”
- send datagram to peer
- wait for peer to ack receipt
- analogy: post office



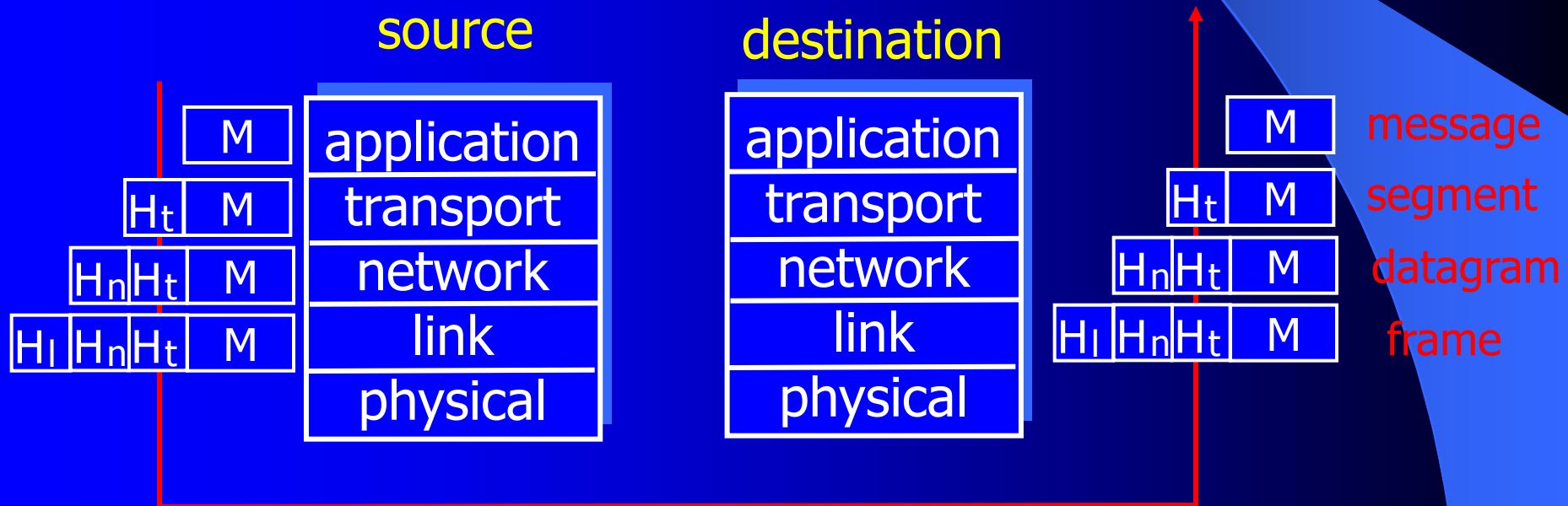
Layers: Physical Communication



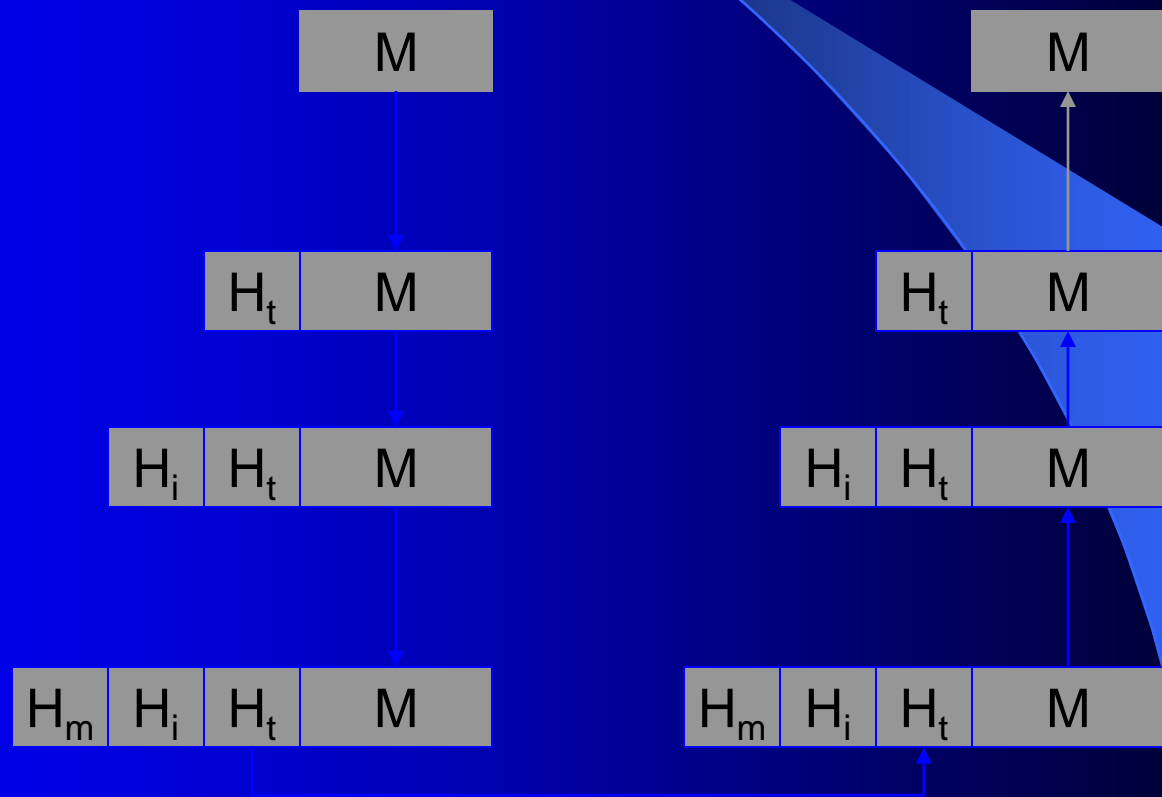
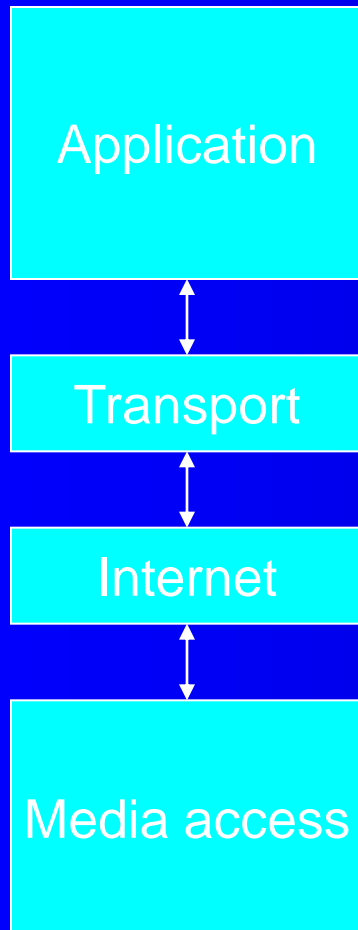
Protocol layering and data

Each layer takes data from above

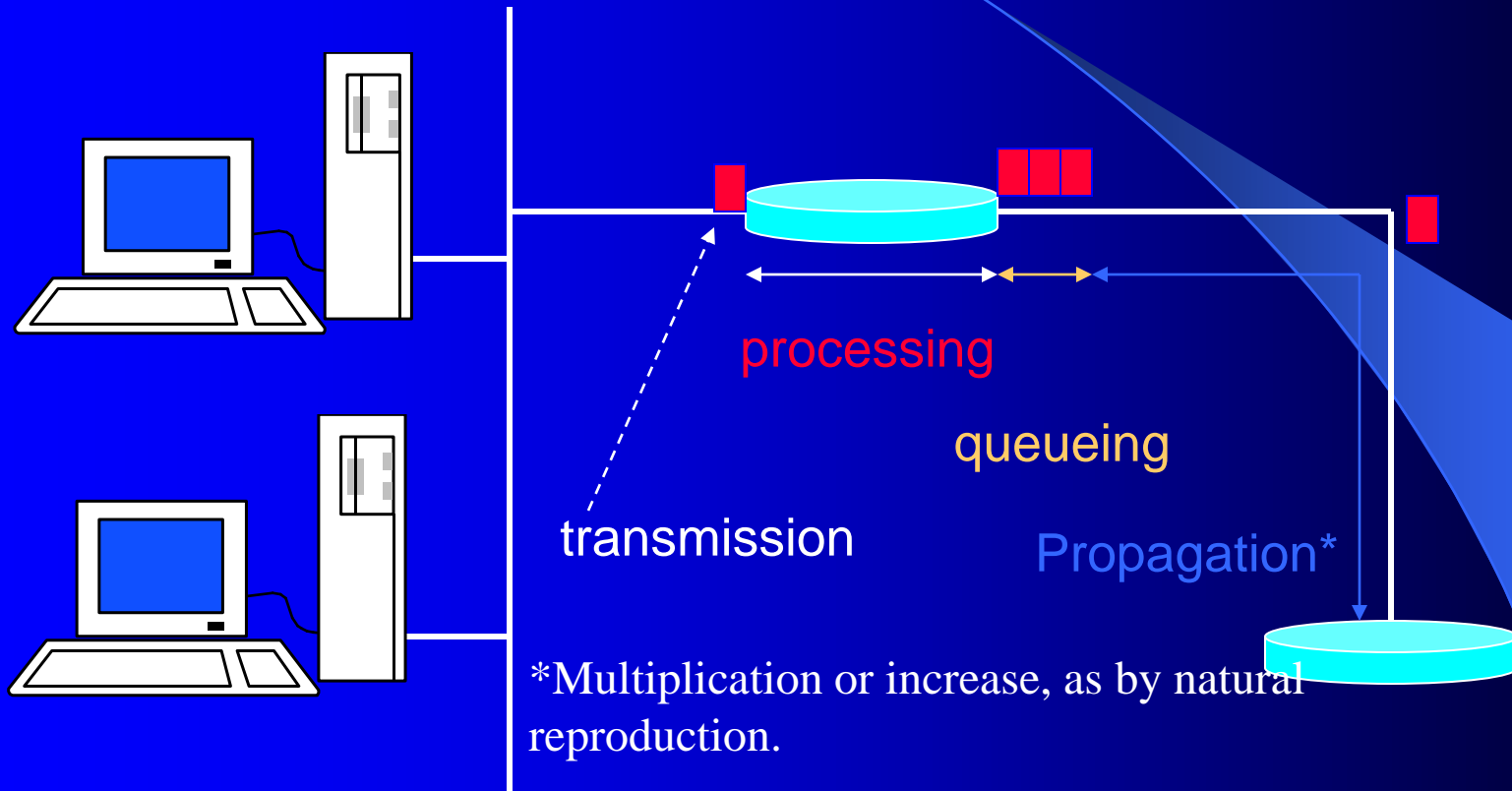
- adds header information to create new data unit
- passes new data unit to layer below



Layers in Action



Delay and Loss in Networks



$$d_{\text{node}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}$$

Queuing Delay

Packets may
be lost!

- Varies from packet to packet
- Depends on traffic
- Consider a router:

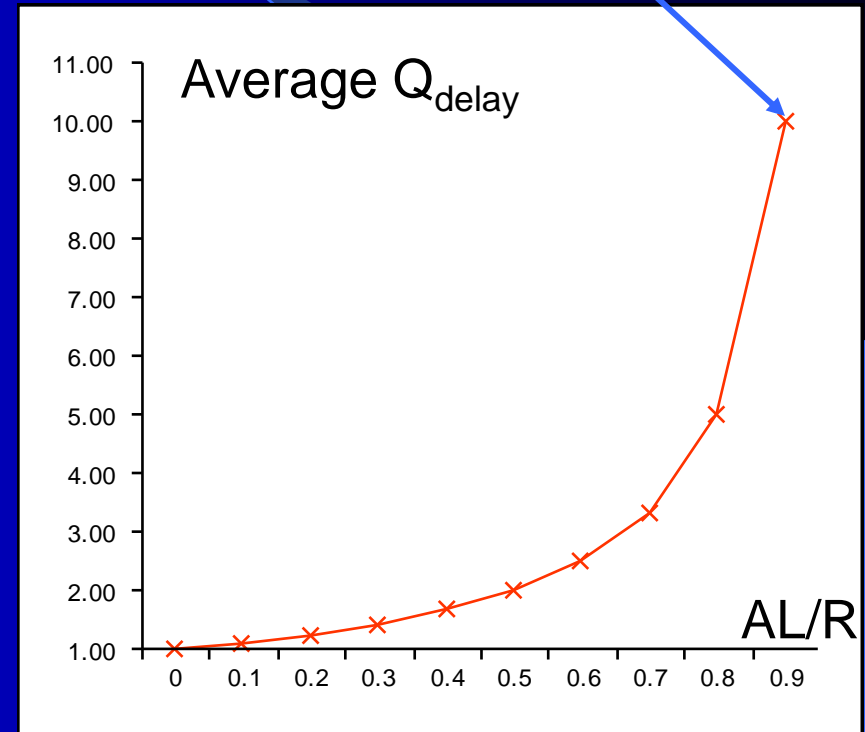
A = arrival rate (n/sec)

R = transmission rate (bits/sec)

L = average packet size (bits)

R/L = max rate of packet
transmission

AL/R = traffic intensity



Q & A, Feedback for this Lecture

- Don't be shy !
- Ask questions ! Get answers (hopefully 😊)
- Help further improvement !

The End

- Thank you for your attention !