西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

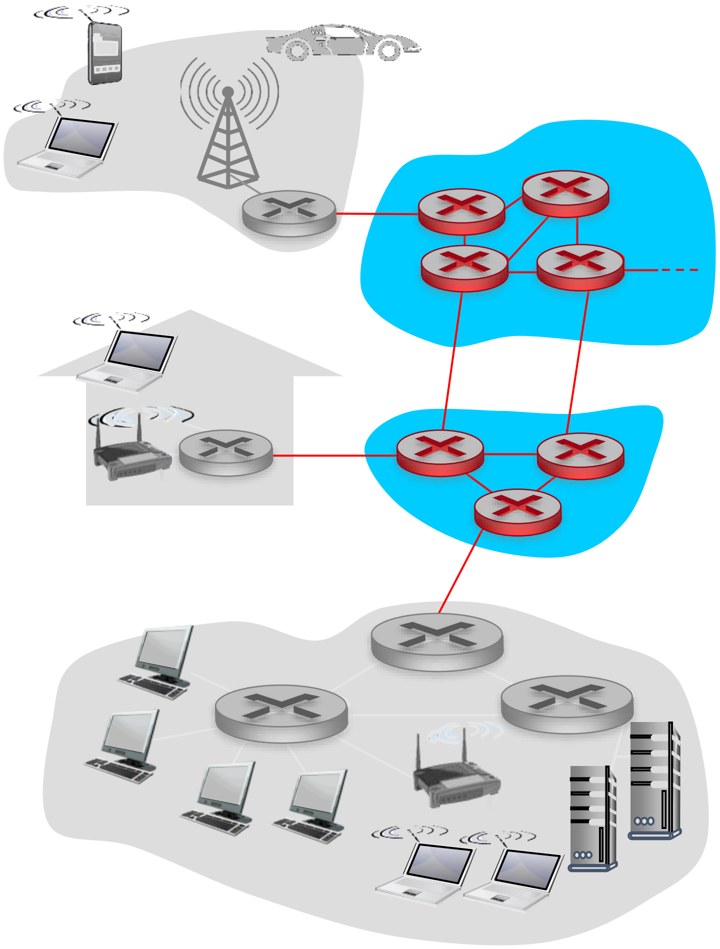
Computer Networks

Lecturer: ZHANG Ying

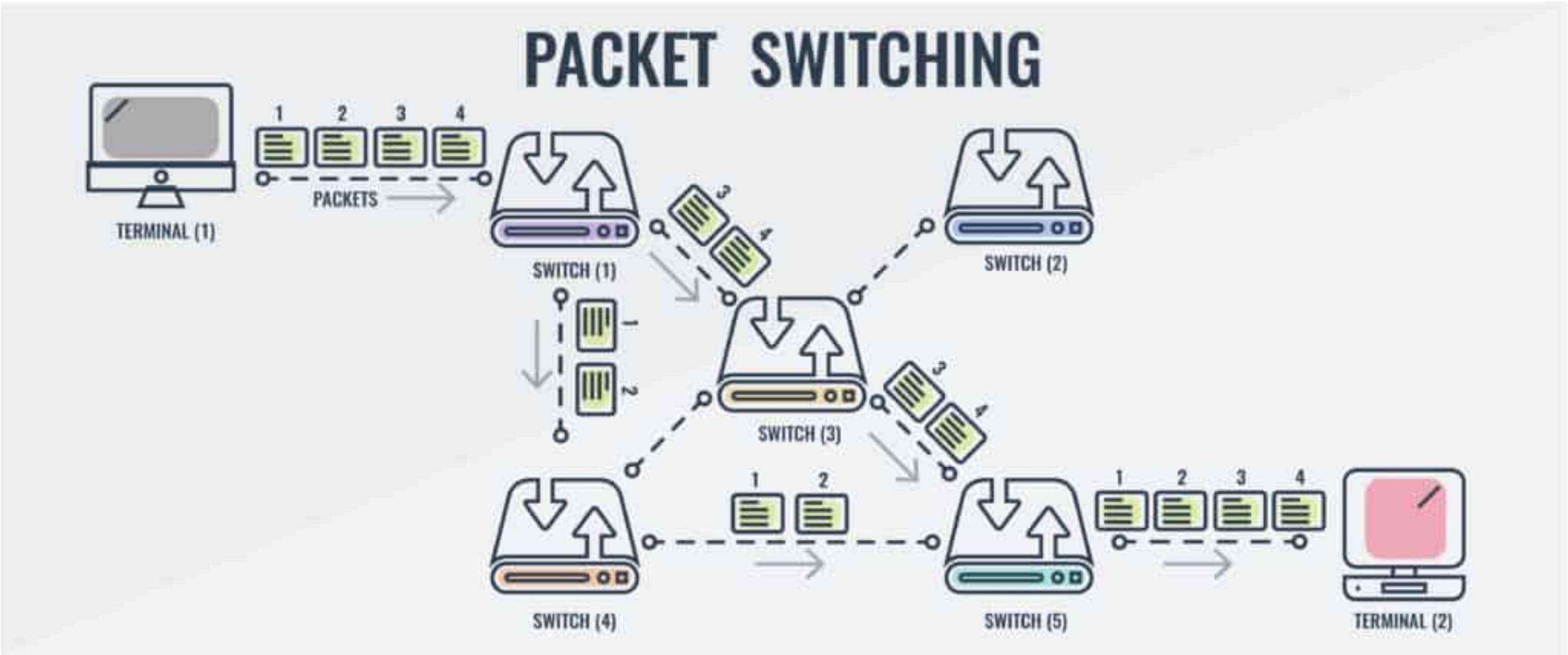
Fall semester 2022

Network core

- Network core/ backbone network is the mesh of routers that interconnect the Internet's end-systems
- Build the core via:
  - Circuit switching
  - Packet switching
  - (message switching)



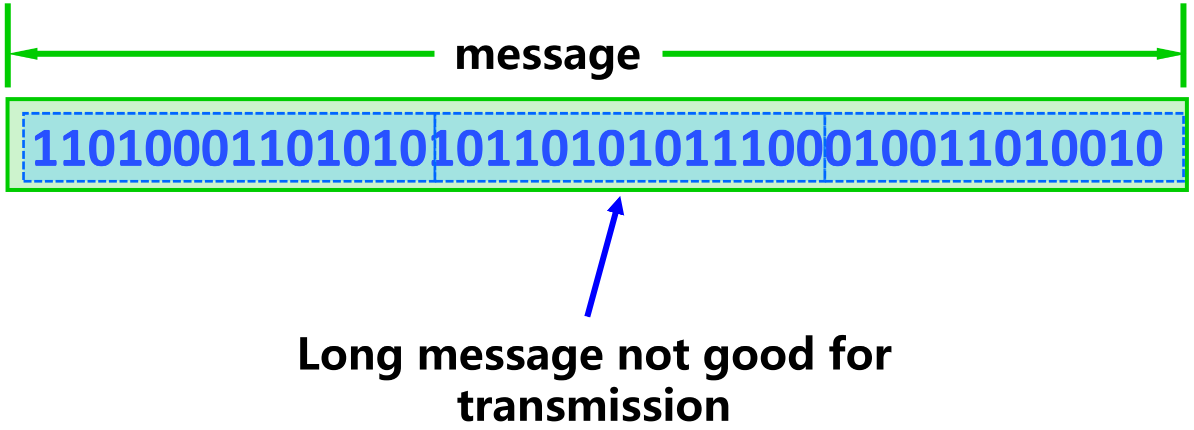
Packet switching – a quick view



- *Packet switching not requires to establish a channel. The channel is available to users throughout the data network.*
- *Long messages are broken down into packets and sent individually to the network*

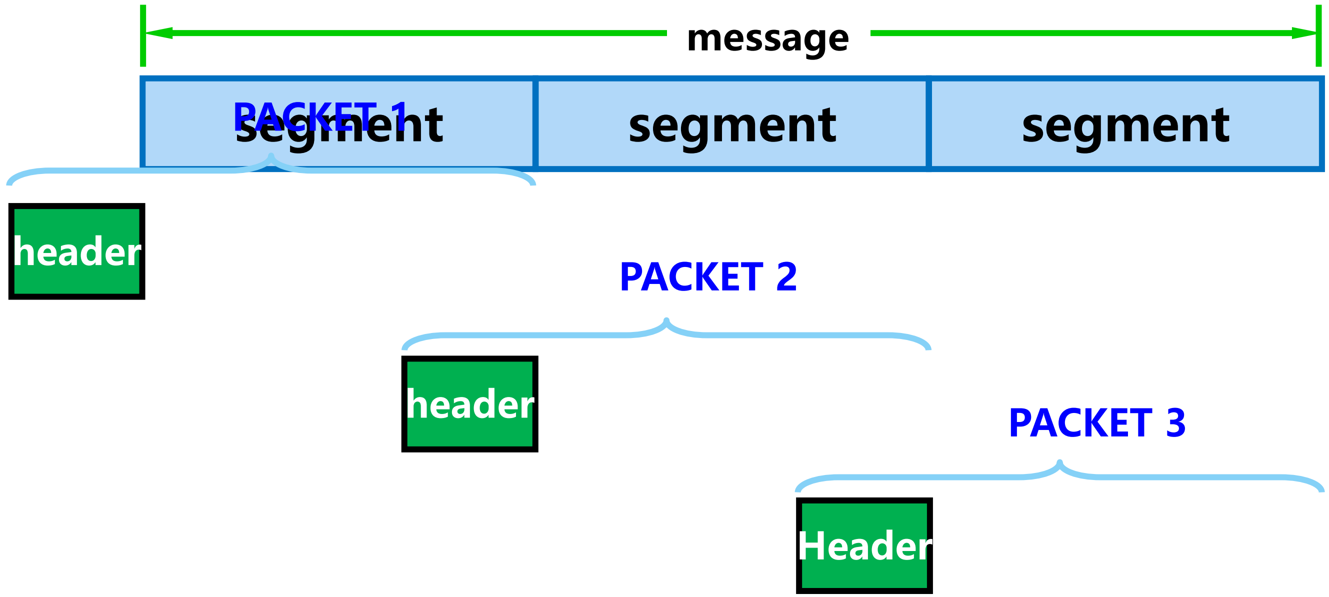
Packet – the basic unit

- Sender: First divide the longer message into shorter, fixed-length data segments



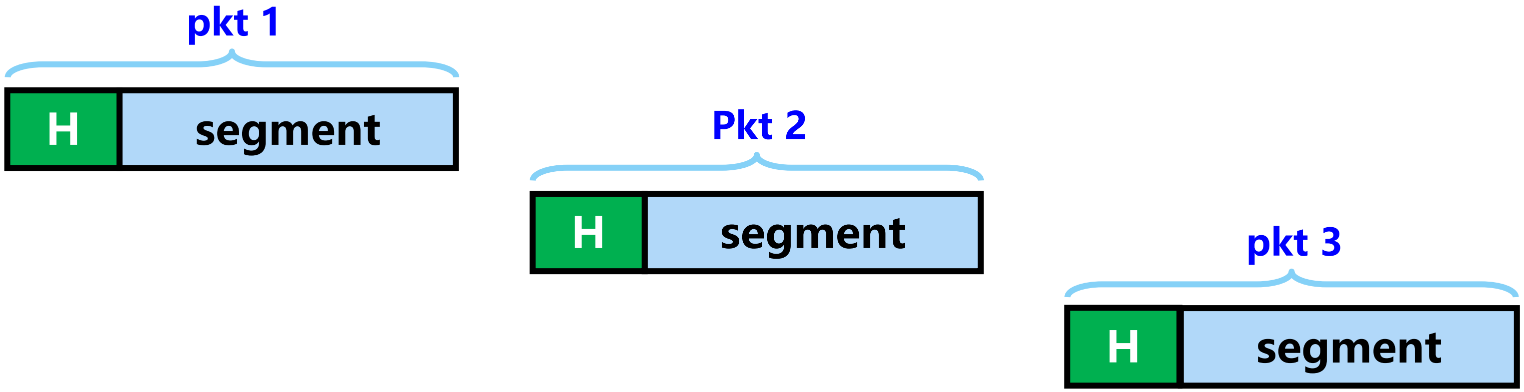
Packet – sender side

- Add a header to the front of each data segment to form a packet.



Packet – sender side

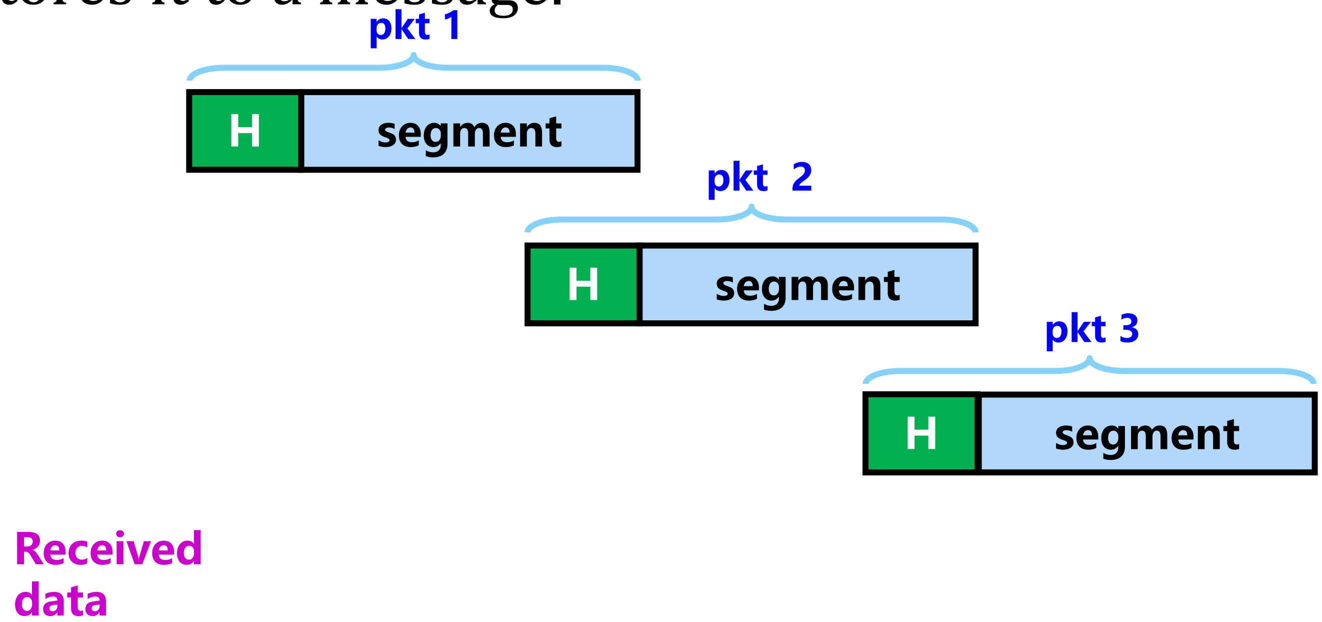
- "packets" : basic data transmission unit.
- Each packet is sent to the receiver in turn



(assuming the receiver is on the left).

Packet – receiver side

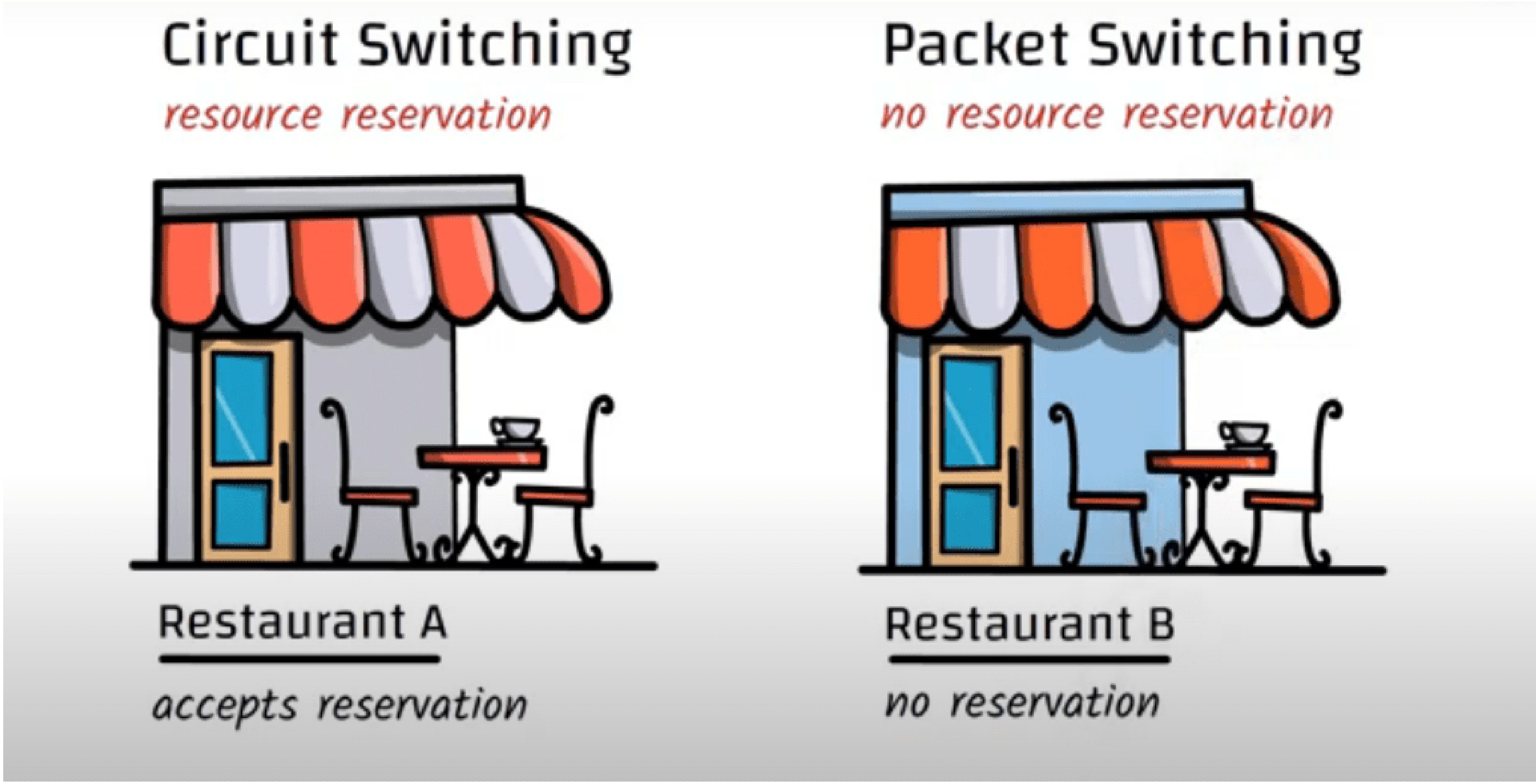
After receiving the packet, the receiving end strips the header and restores it to a message.

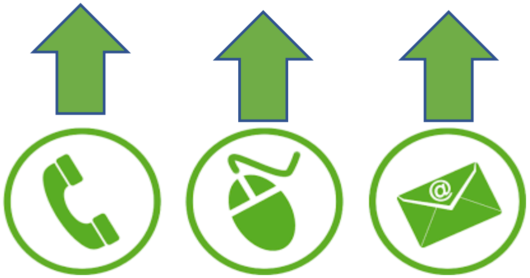


Packet – the basic unit

- At the receiving end, the received data is restored to the original message

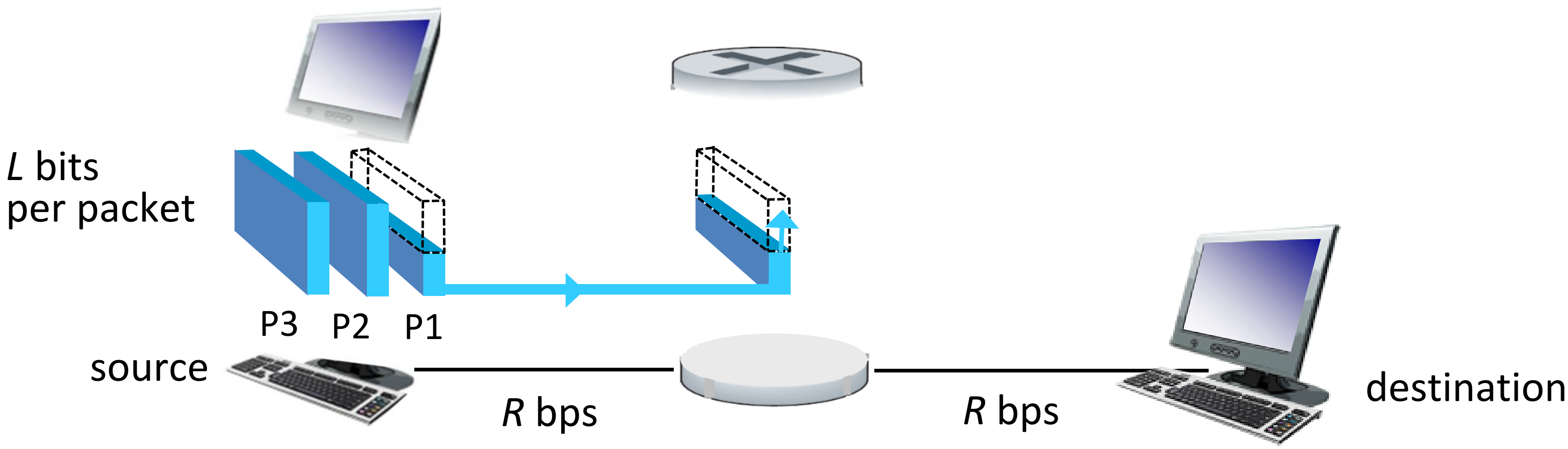
- Here we assume that the packets are transmitted without errors and are not discarded during forwarding.


Circuit Switching
resource reservation
Restaurant A
accepts reservation
Packet Switching
no resource reservation
Restaurant B
no reservation



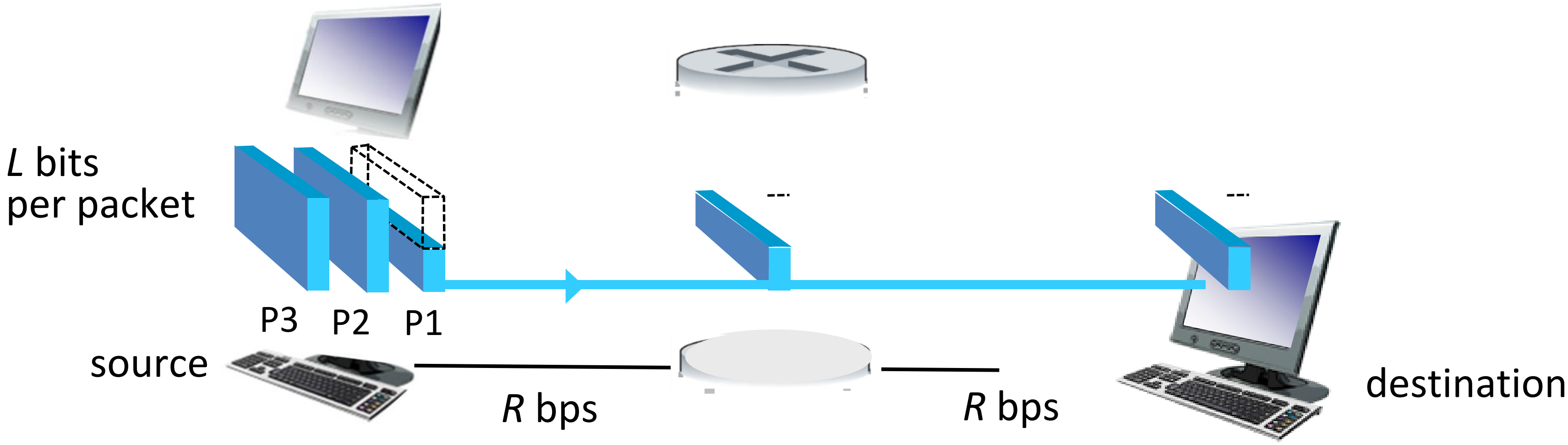

wait

Store and forward technique



- Each packet takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *store and forward:* entire packet must arrive at router before it can be transmitted on next link
- end-end delay = $2L/R$ (assume zero prop-delay)

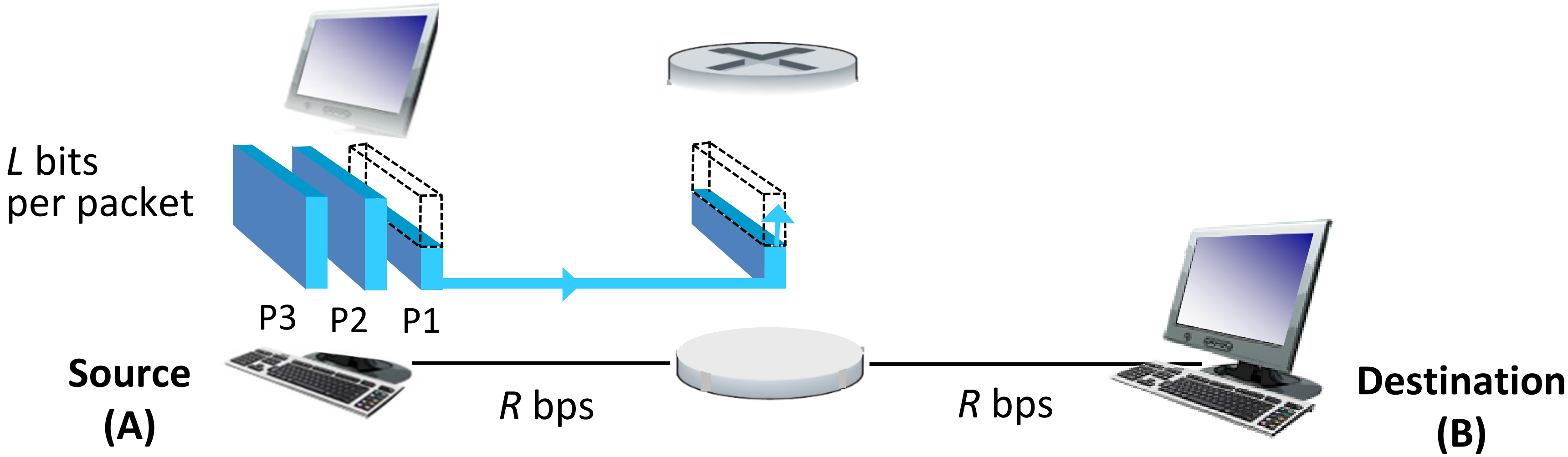
Scenario 1: What if not using store and forward?



- *The router push out each bit immediately after its arrival*
- end-end delay = L/R
- as one hop

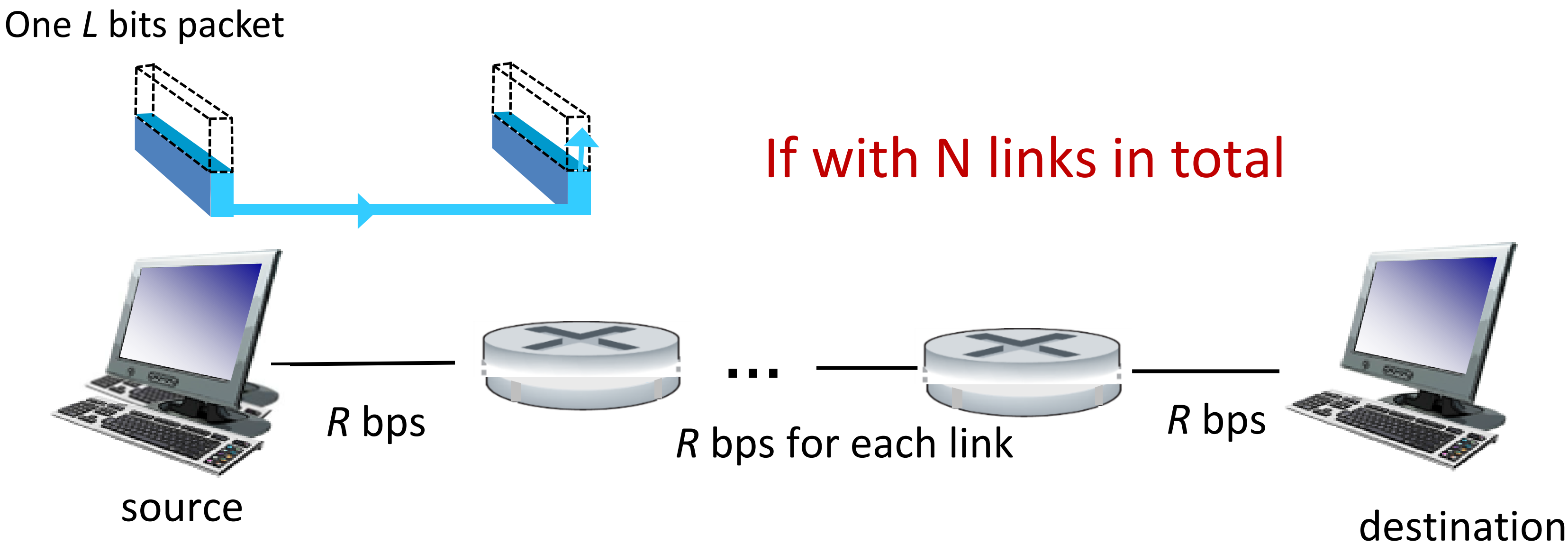
more on delay shortly …

Scenario 2: Store and forward technique with 3 pkts



- End-end delay per packet: 2L/R seconds
- 2L/R seconds: 1^{st} *pkt @ B & 2^{nd} pkt @ router*
- *3L/R seconds: 2^{nd} pkt @ B, 3^{rd} pkt @ router*
- *4L/R seconds: 3^{rd} pkt @ B*

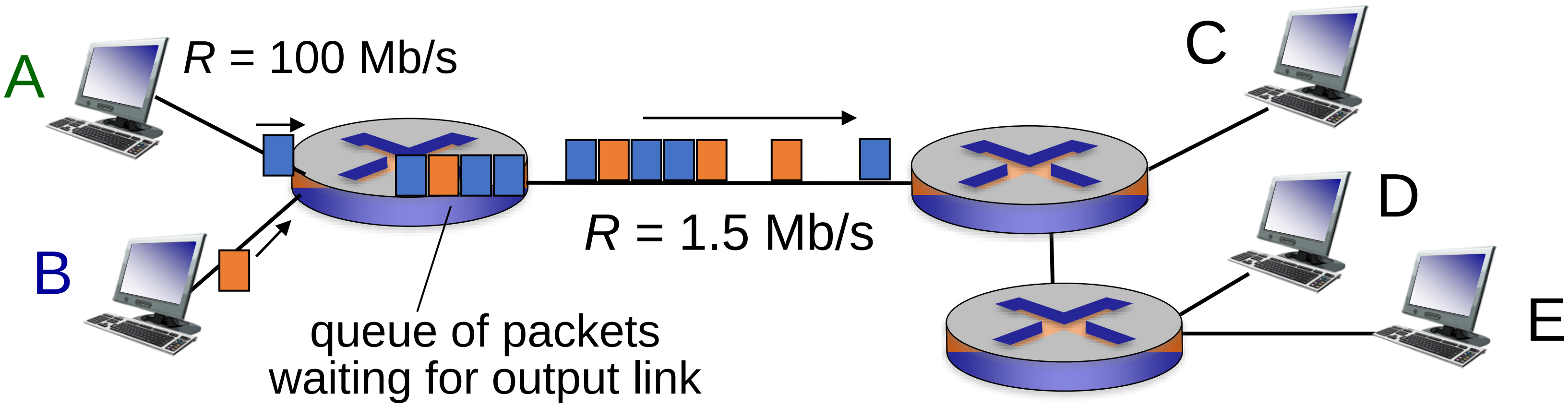
Scenario 3: A more general case with N links for 1pkt:



- By store and forwarding technique:

End-end delay = N x L / R (sec)

Packet loss



if arrival rate (in bits) to link exceeds transmission rate of link for a period of time:

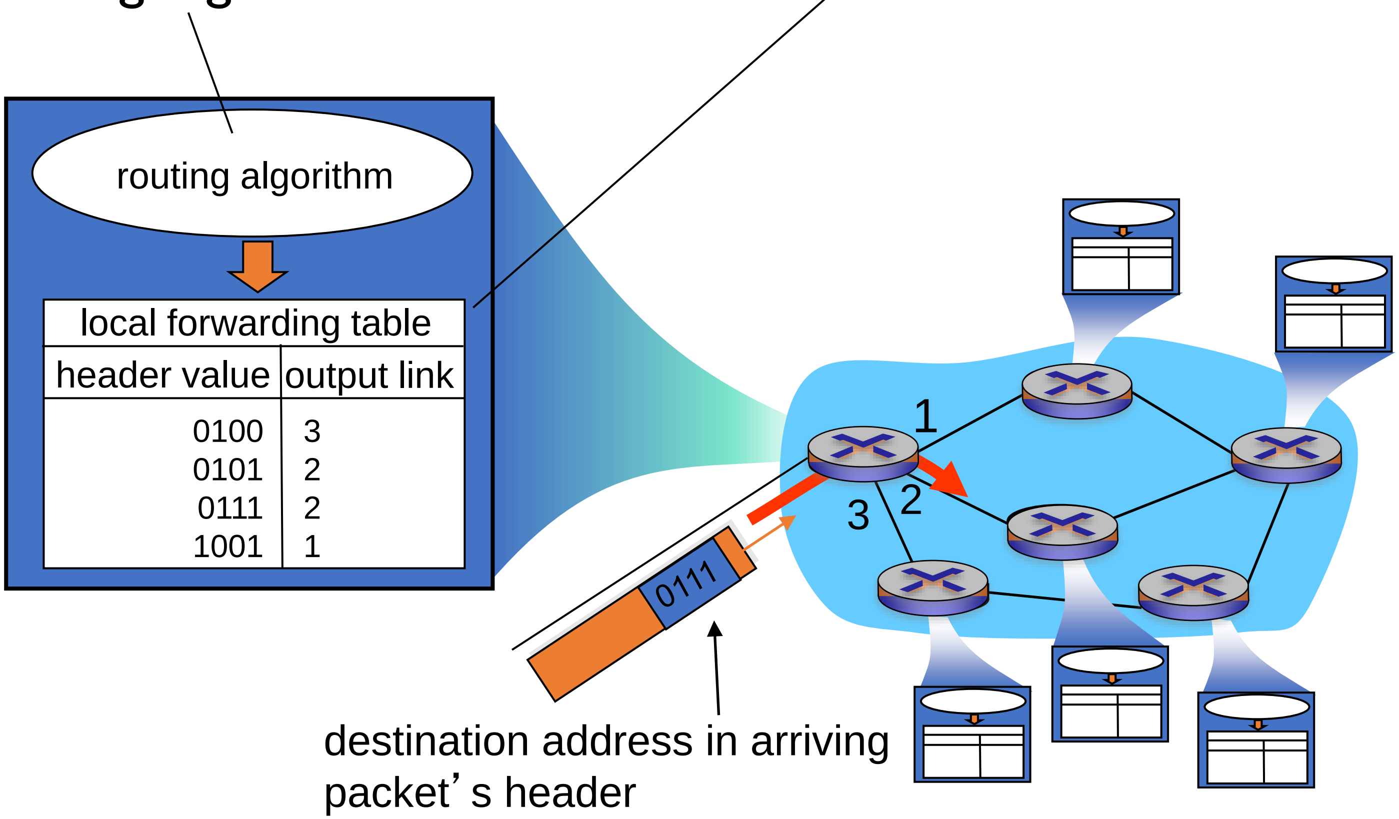
- packets will queue, wait to be transmitted on link
- packets can be dropped (lost) if memory (buffer) fills up

Two key functions

routing: determines source-destination route taken by packets

- *routing algorithms*

forwarding: move packets from router's input to appropriate router output



Router

Router processes the packet:

- Put the received packets into the cache first (temporarily store);
- Look up the forwarding table to find out which port should be forwarded to a destination address;
- Forward the packet to the appropriate port.

Router is different from Host

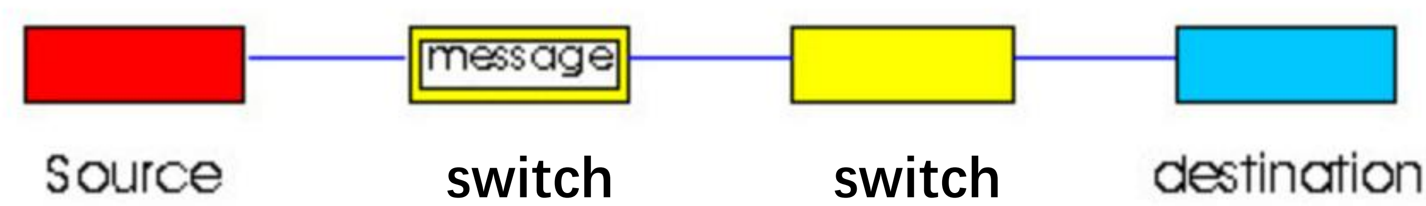
- The host computer processes information for the user and sends and receives packets to and from the network.

- The router stores and forwards the packet, and finally delivers the packet to the destination host.

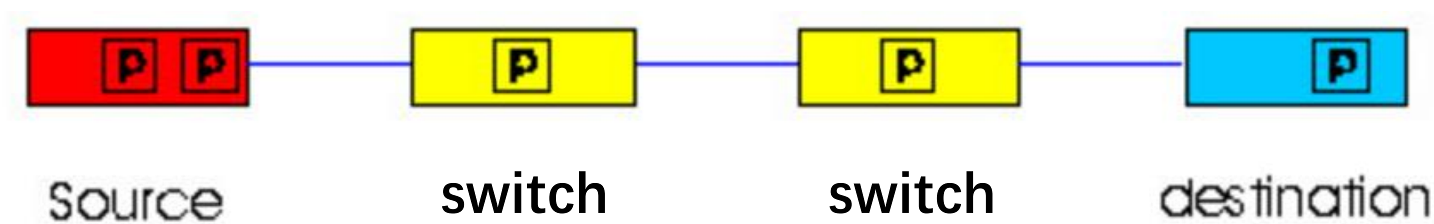
A comparison btw Circuit & Packet switching

Circuit switching	Packet switching
Dedicated transmission path	No dedicated path
Continuous transmission of data	Transmission of packets
Reserves the required bandwidth in advance	Acquires and releases bandwidth as needed.

message switching vs. packet switching

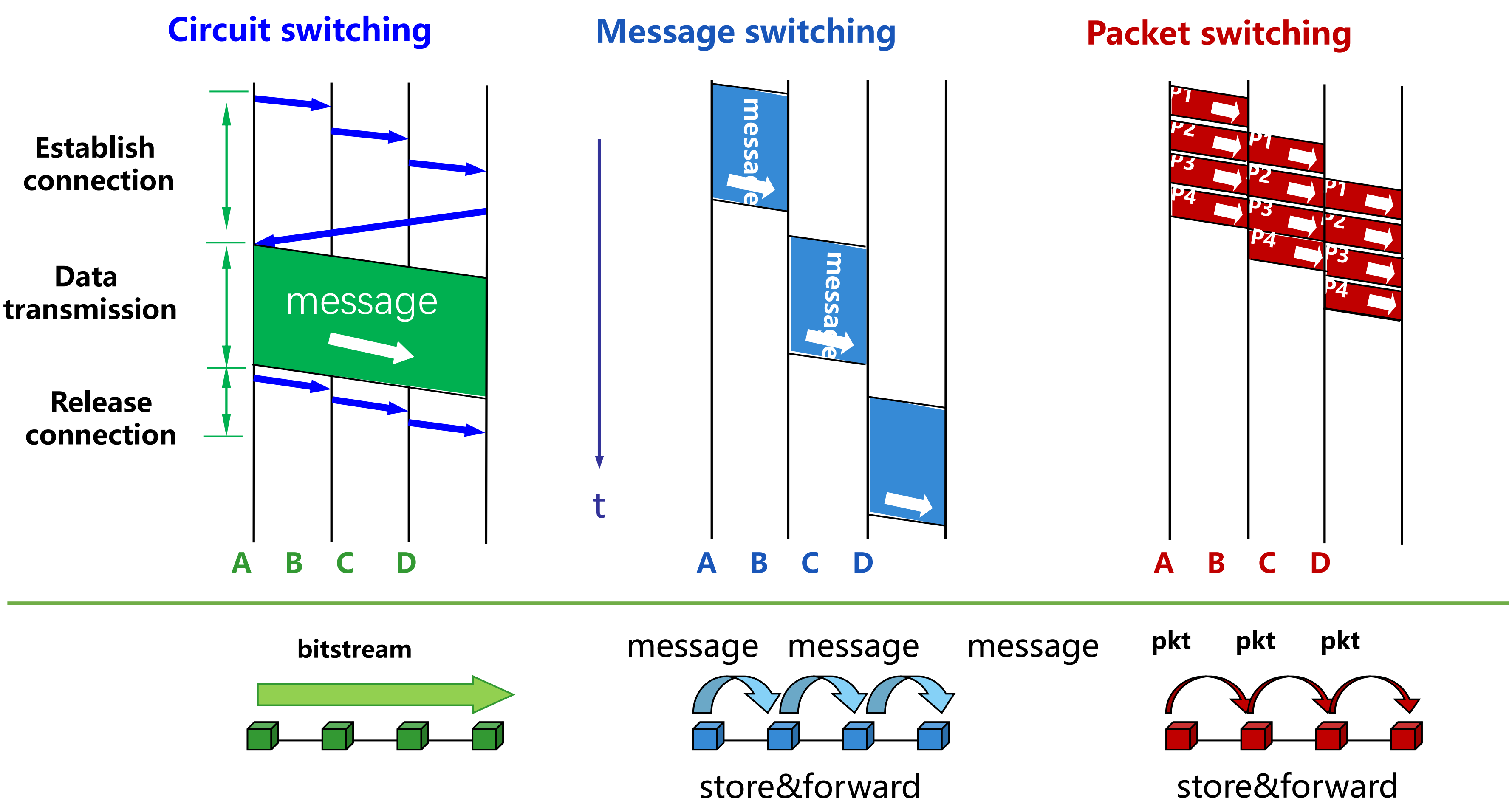


(a) message switching



(b) packet switching

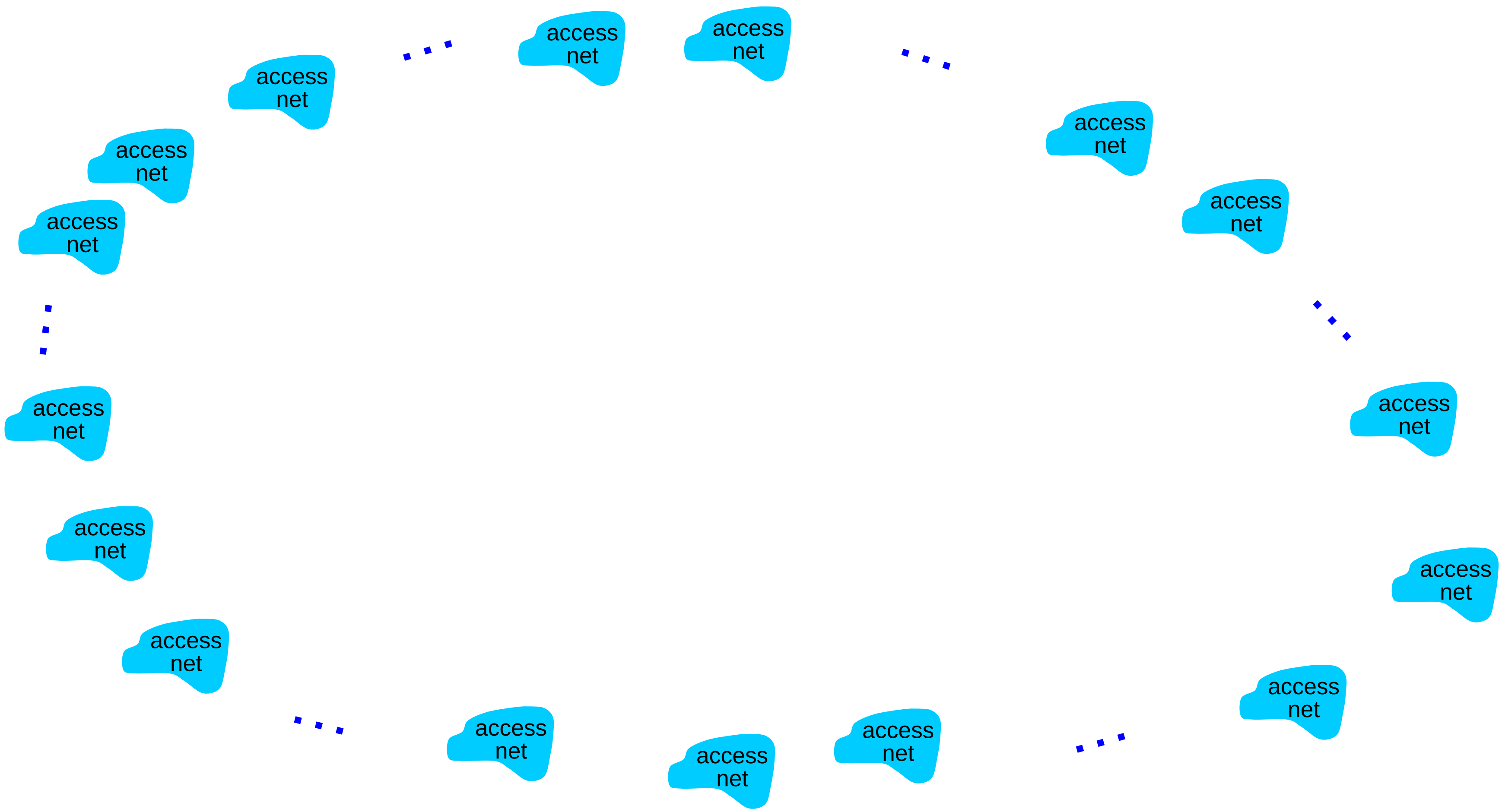
Data transmission difference



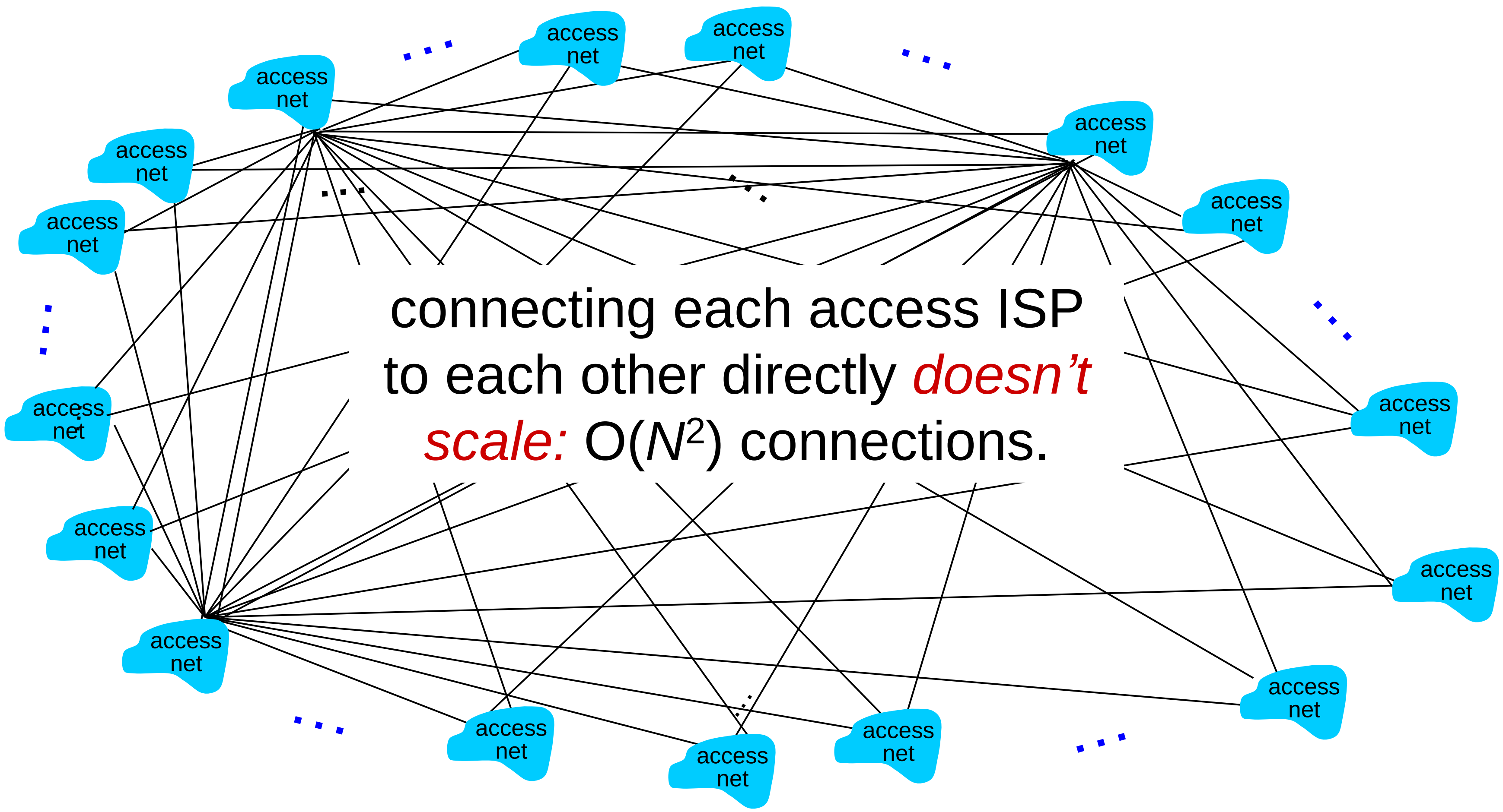
Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - residential, company and university ISPs
- Access ISPs in turn must be interconnected.
  - so that any two hosts can send packets to each other
- Resulting network of networks is very complex
  - evolution was driven by economics and national policies
- Let's take a stepwise approach to describe current Internet structure

Question: given *millions* of access ISPs, how to connect them together?



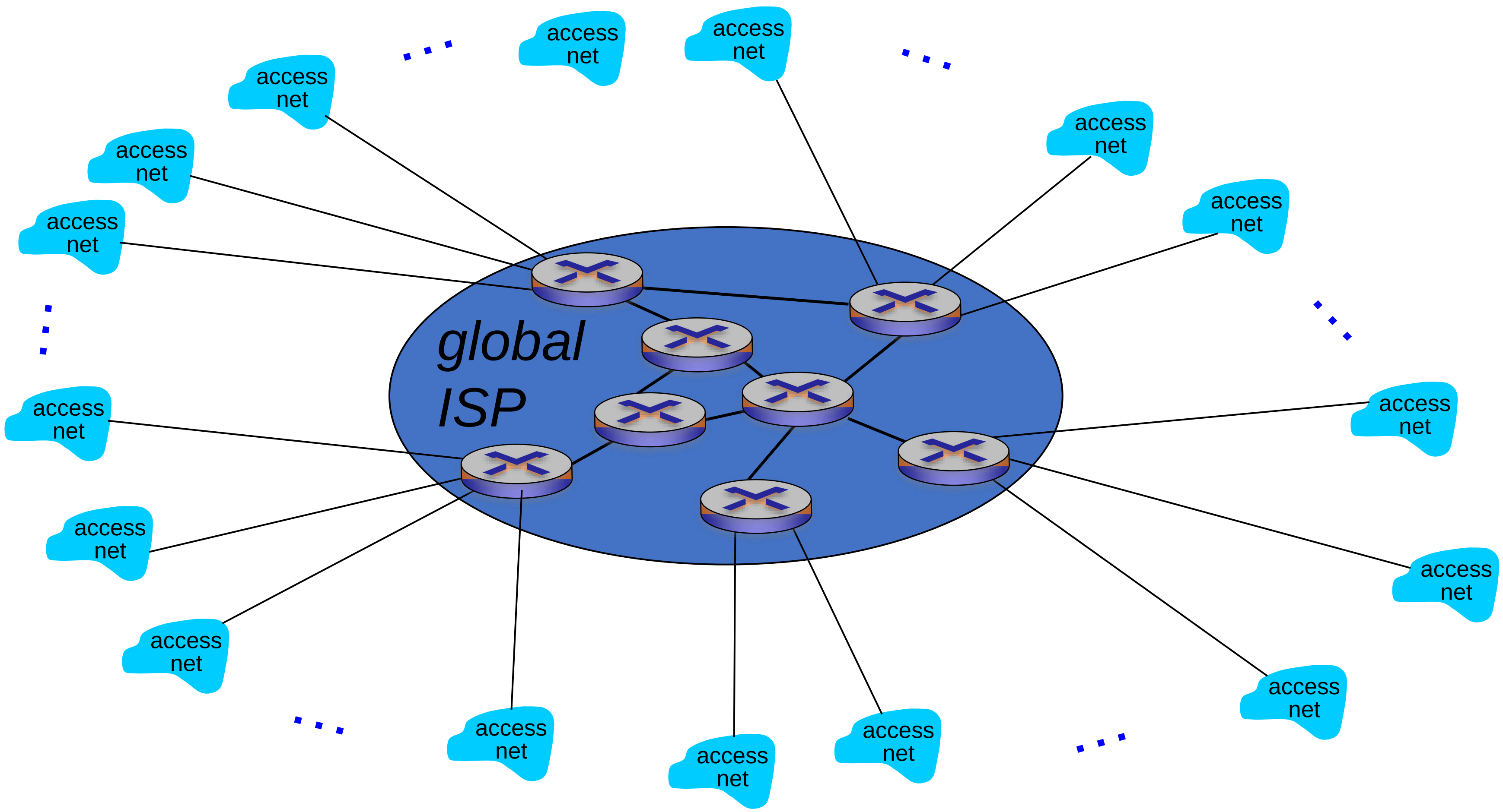
Option1: *connect each access ISP to every other access ISP?*



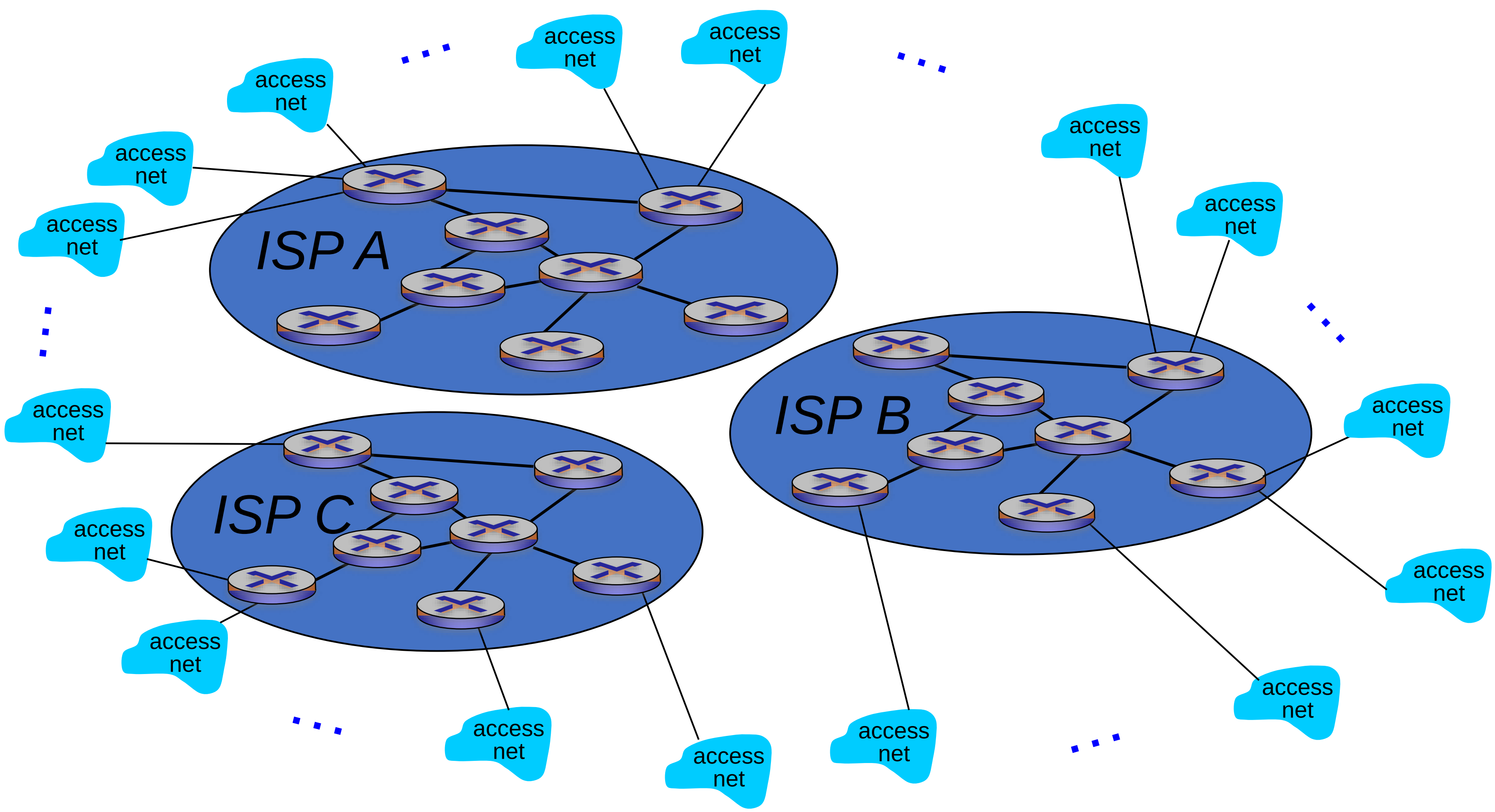
connecting each access ISP to each other directly *doesn't scale:* $O(N^2)$ connections.

Option2: connect each access ISP to one global transit ISP?

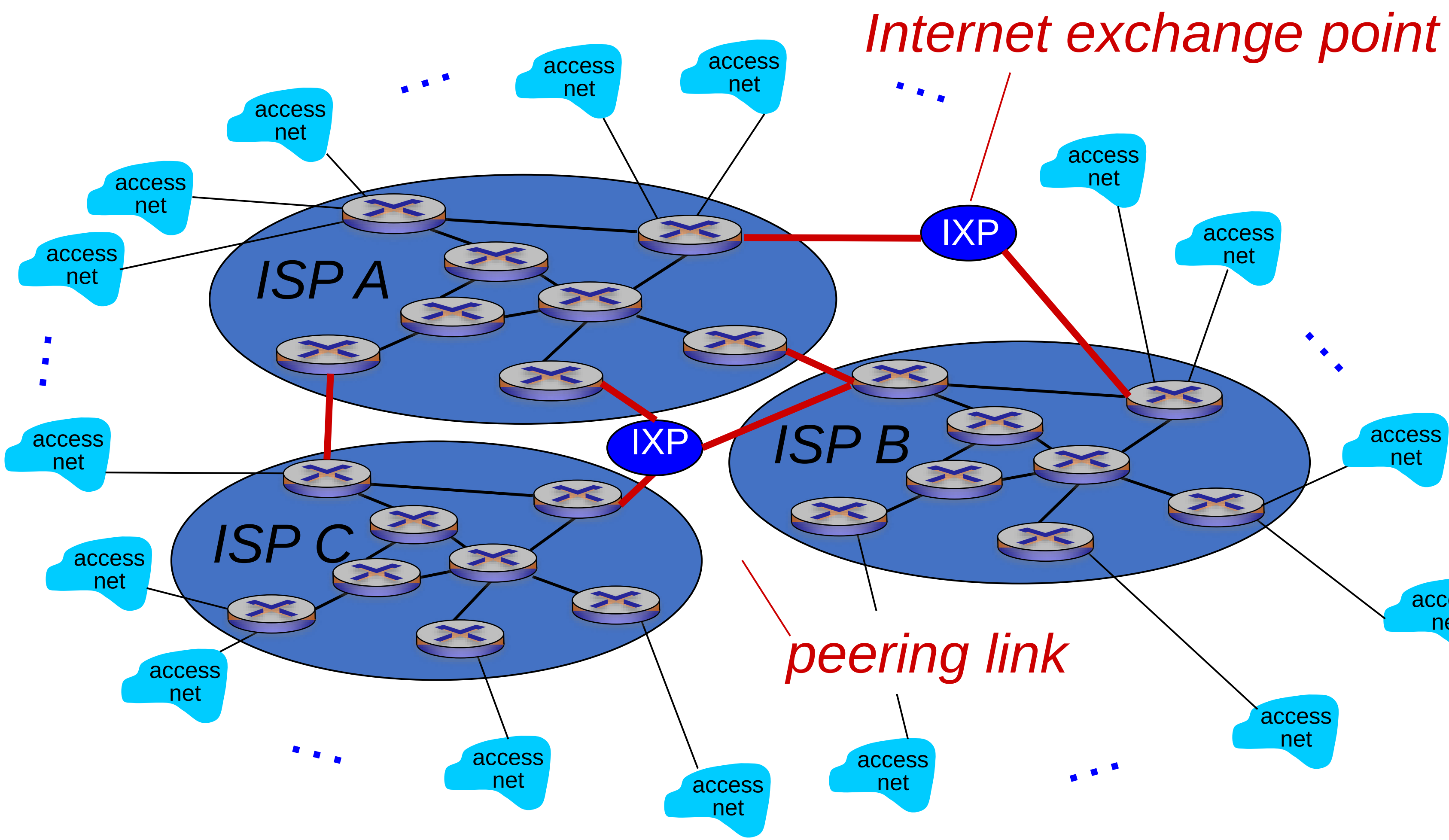
Customer and provider ISPs have economic agreement.



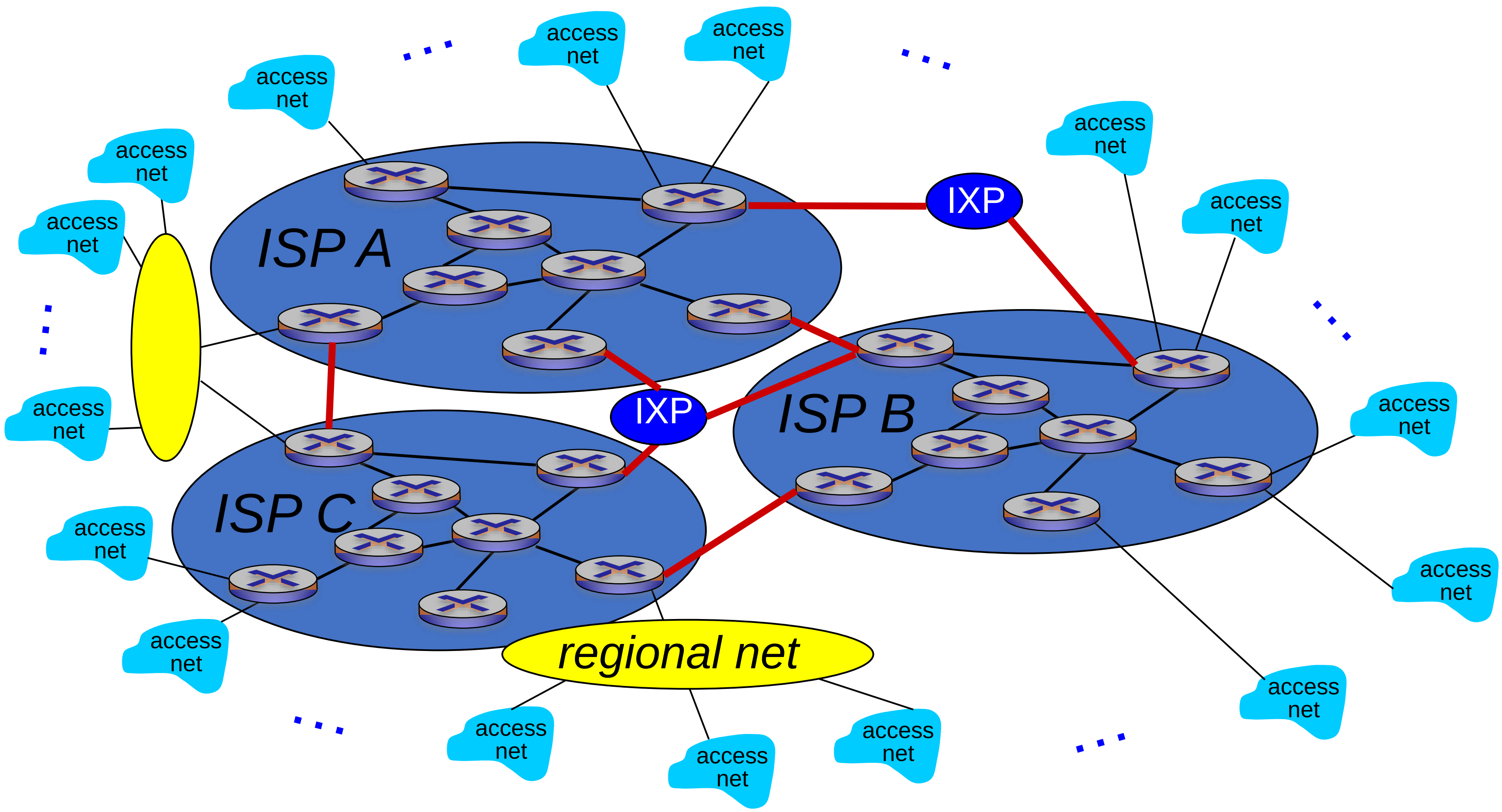
But if one global ISP is viable business, there will be competitors …. Multiple ISPs



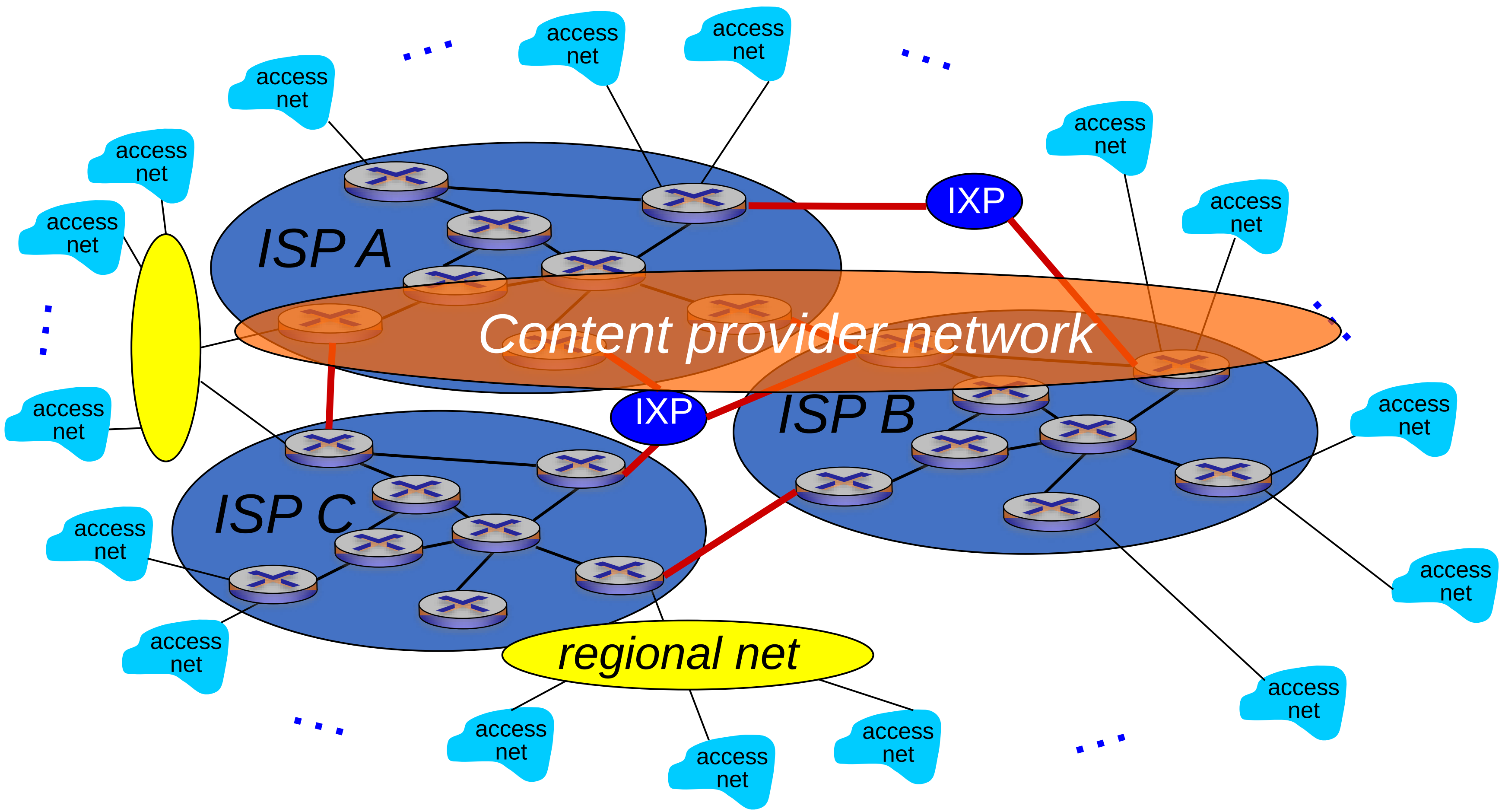
But if one global ISP is viable business, there will be competitors …. which must be interconnected



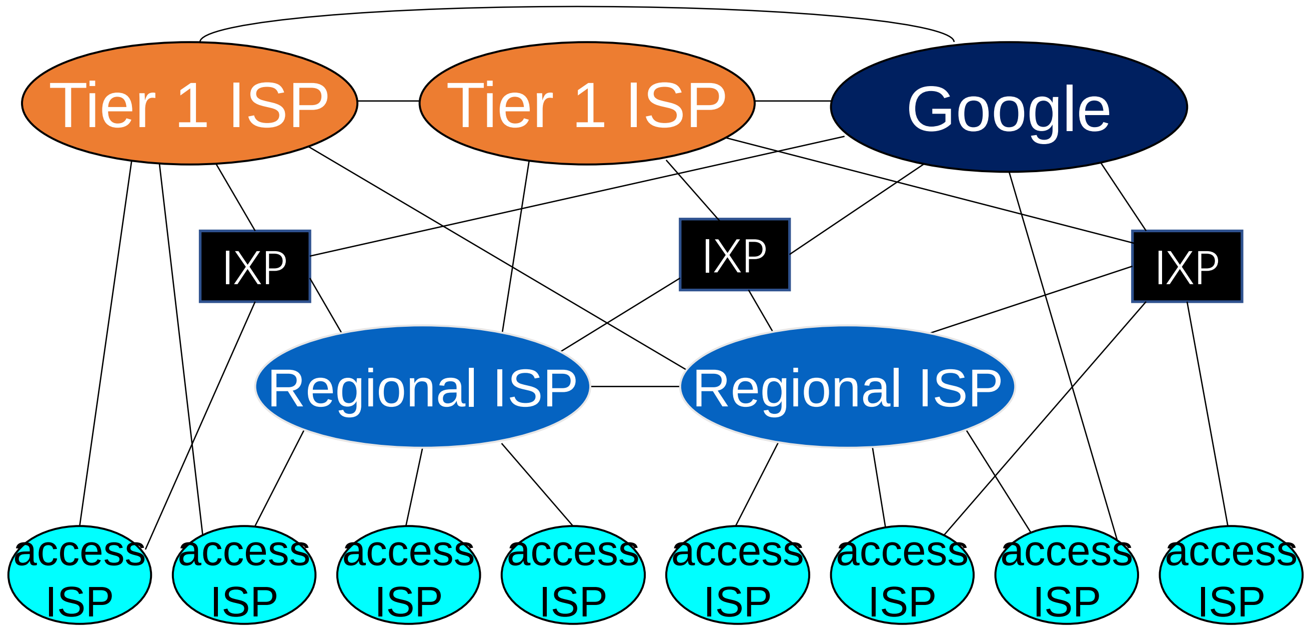
… and regional networks may arise to connect access nets to ISPs



… and content provider networks (e.g., Google, Microsoft, Akamai) may run their own network, to bring services, content close to end users



Internet structure: network of networks



At center: small # of well-connected large networks

- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider network (e.g., Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

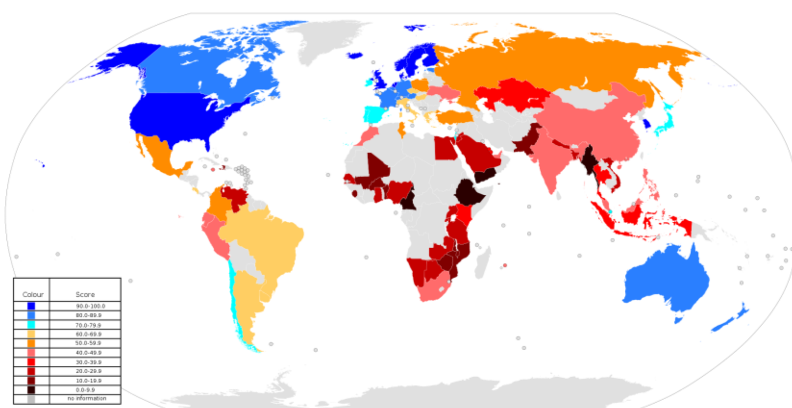
Internet or internet?

Internet	internet
Same	
Network of networks	Network of networks
Difference	
Follow TCP/IP protocals Global coverage	a network of multiple computer networks
TCP/IP	TCP/IP or others
A dedicated term	A more general term

Arbitrarily interconnecting several computer networks (no matter what protocol is used) and being able to communicate with each other, this constitutes an Internet, rather than the Internet.

World Wide Web & Internet

- The Internet has become the largest and fastest-growing computer network in the world
- The World Wide Web (World Wide Web) has become the main driving force for this exponential growth of the Internet



A global map of the Web Index for countries in 2014

Let's Share What We Know

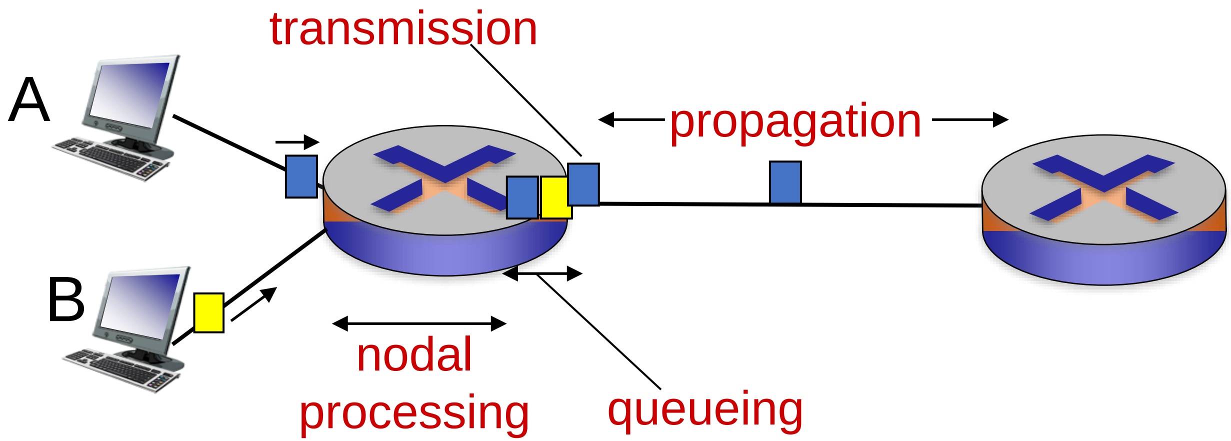
World Wide Web

historic World Wide Web logo

Outline

1	what is the Internet
2	network edge
3	network core
4	delay, loss, throughput in networks
5	protocol layers, service models

Four sources of packet delay



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

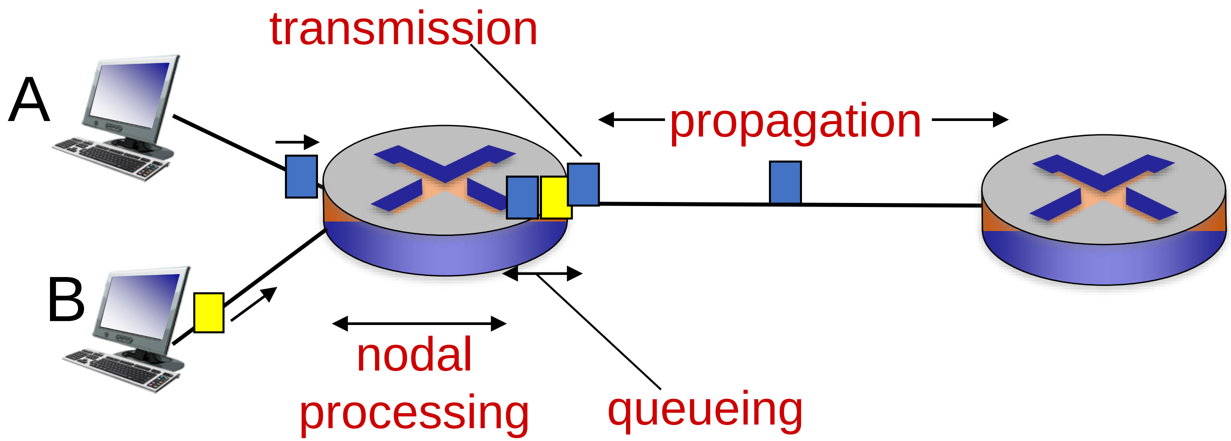
d_{proc}: nodal processing

- check bit errors
- determine output link
- typically < msec

d_{queue}: queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

Four sources of packet delay



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

d_{trans}: transmission delay:

- L: packet length (bits)
- R: link *bandwidth (bps)*
- $d_{trans} = L/R$

d_{trans} and d_{prop} *very* different

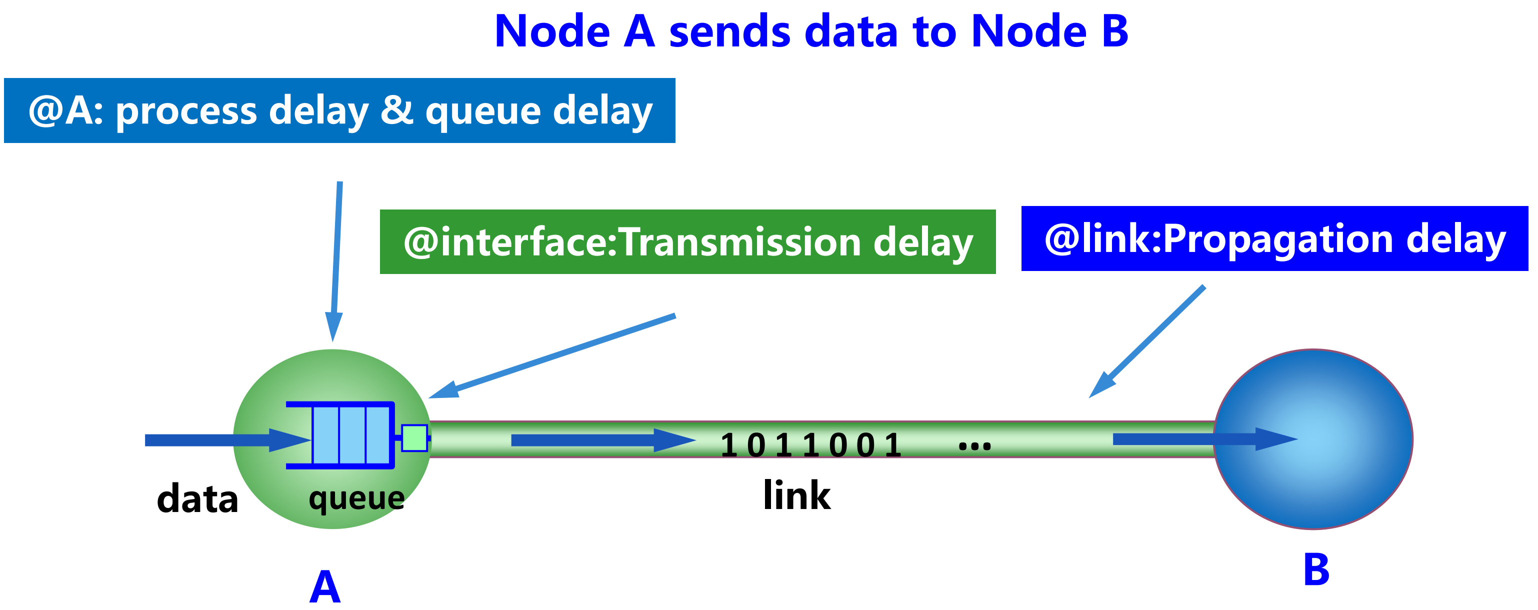
d_{prop}: propagation delay:

- d: length of physical link
- s: propagation speed (~2x10^8 m/sec)
- $d_{prop} = d/s$

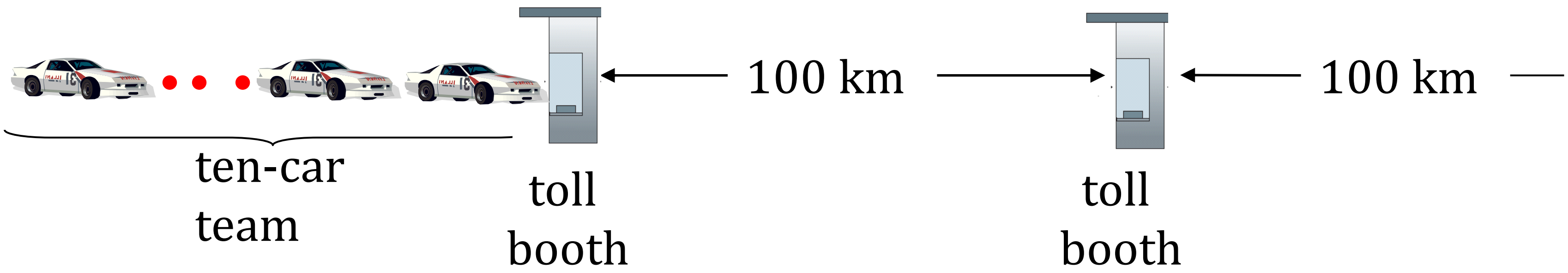
Transmission vs Propagation

- Very different
- Transmission: time to push the pkts to next link
  - NOT Related to the physical distance
- Propagation: carry the bit via physical media to reach destination.
  - Related to the physical distance

Where the delays occurs?

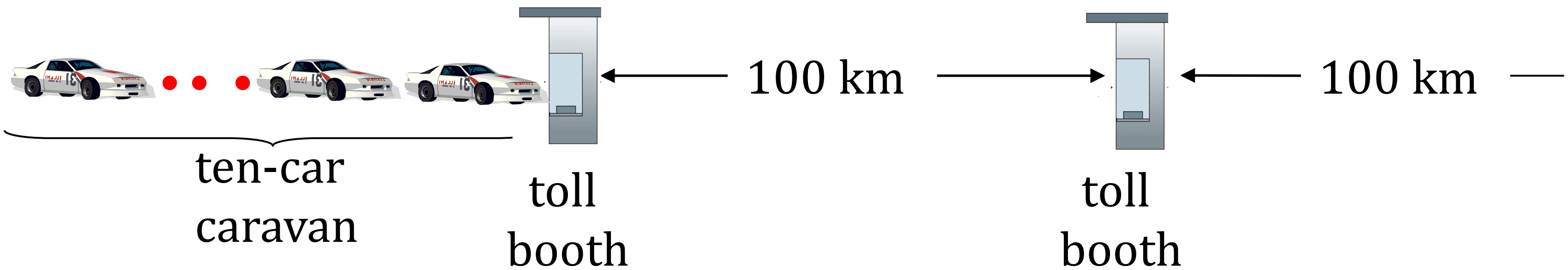


An analogy - 1



- cars "propagate" at 100 km/hr
- toll booth takes 12 sec to service car (bit transmission time)
- car ~ bit; team ~ packet
- *Q:* How long until team is lined up before 2nd toll booth?
- time to "push" entire team through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- *A:* 62 minutes

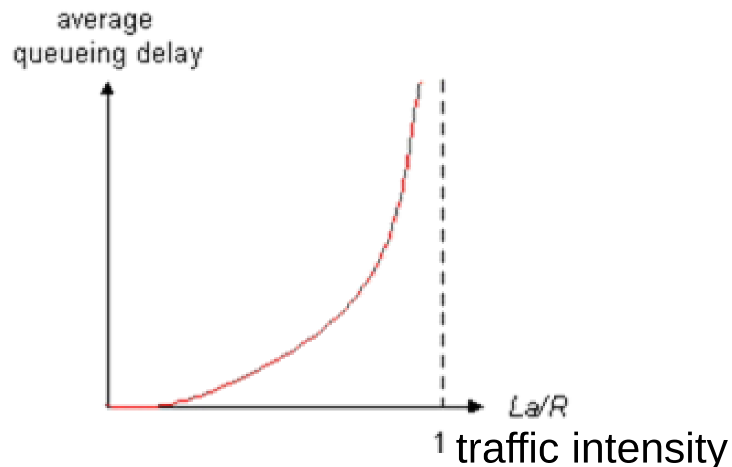
An analogy - 2



- suppose cars now "propagate" at 1000 km/hr
- suppose toll booth takes 1 min to serve a car
- *Q:* Will cars arrive to 2nd booth before all cars serviced at first booth?
- *A:* Yes! after 7 min, first car arrives at second booth; three cars still at first booth

Queueing delay and traffic intensity

- *R:* link bandwidth (bps)
- *L:* packet length (bits)
- a: average packet arrival rate

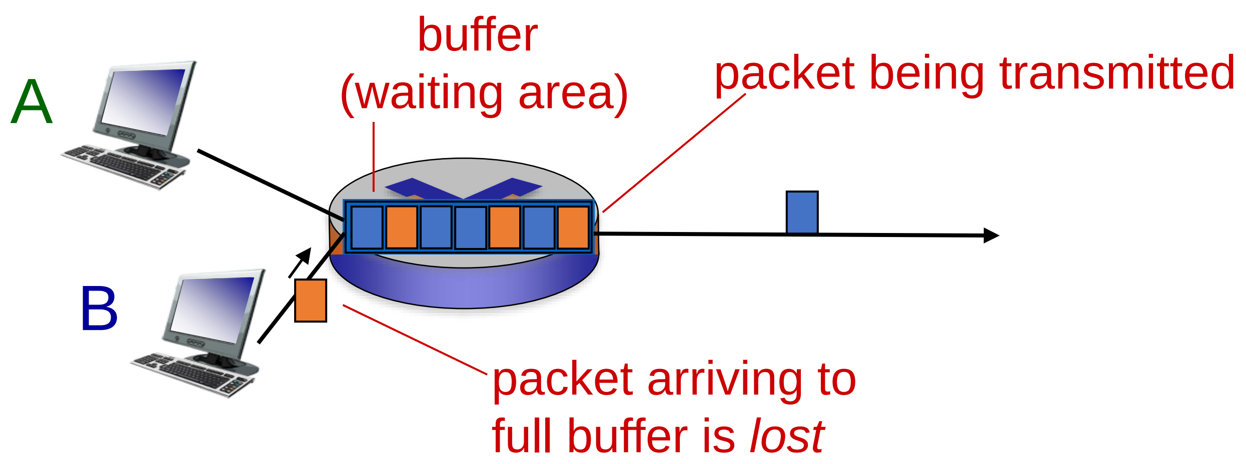


- La/R ~ 0: avg. queue delay small
- La/R ~ 1: avg. queue delay large
- La/R > 1: more “work” arriving than can be serviced
  - average delay infinite

La/R ~ 0

La/R -> 1

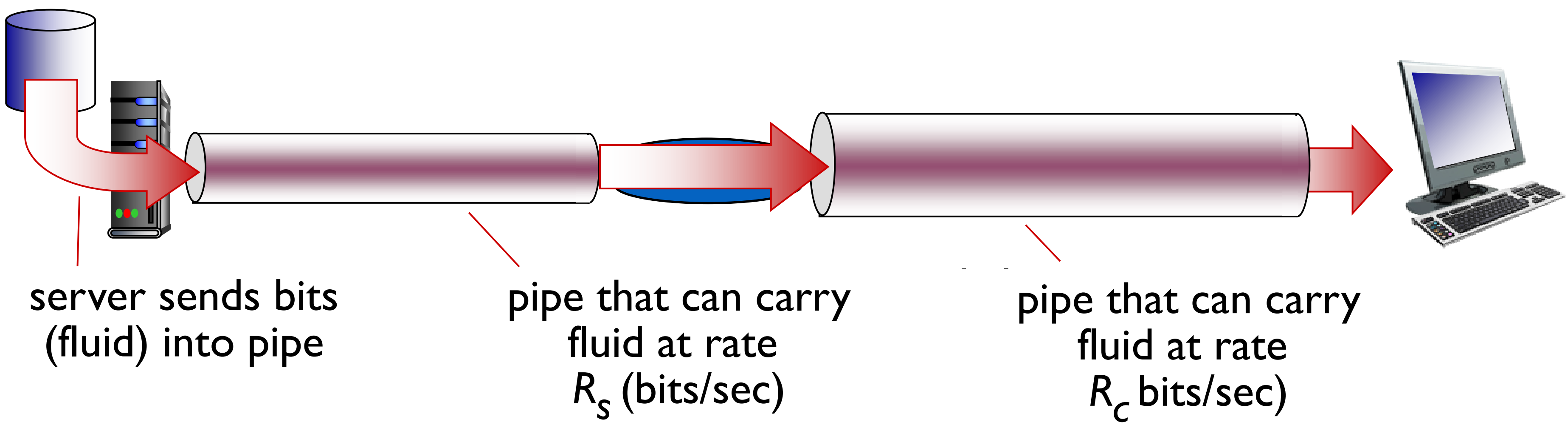
Packet loss



- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all

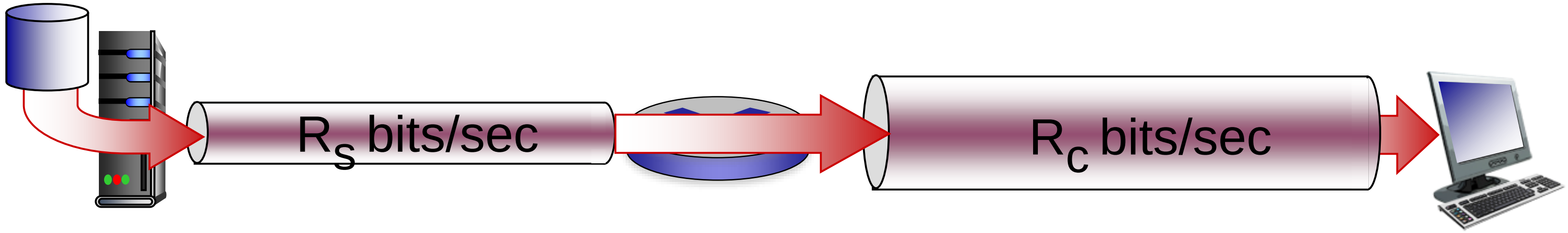
Throughput

- *throughput:* rate (bits/time unit) at which bits transferred between sender/receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time



Throughput - 1

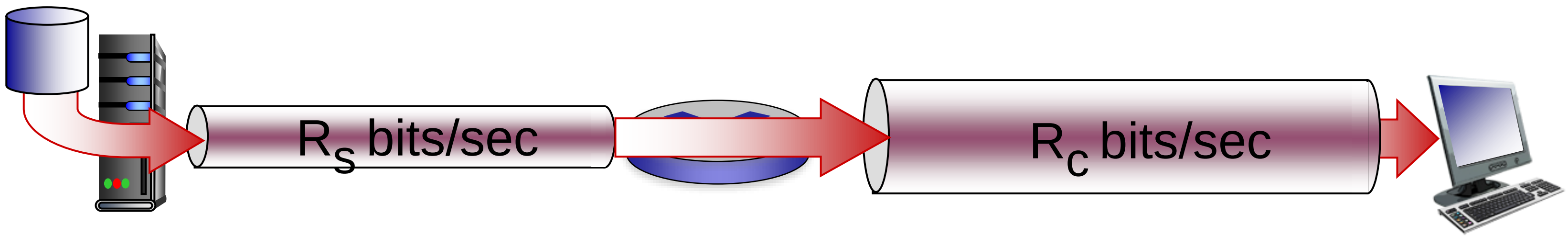
- *$R_s < R_c$* What is average end-end throughput?
- *Answer : Rs*



Throughput - 2

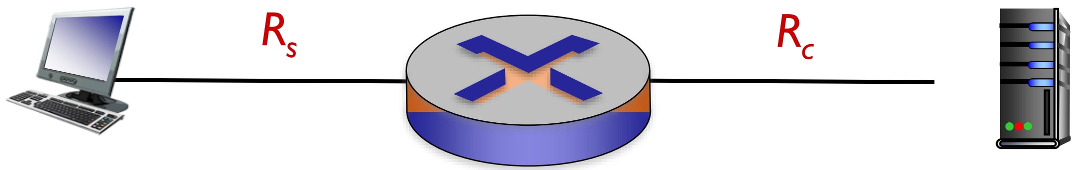
- $R_s > R_c$ What is average end-end throughput?

• *Answer : Rc*

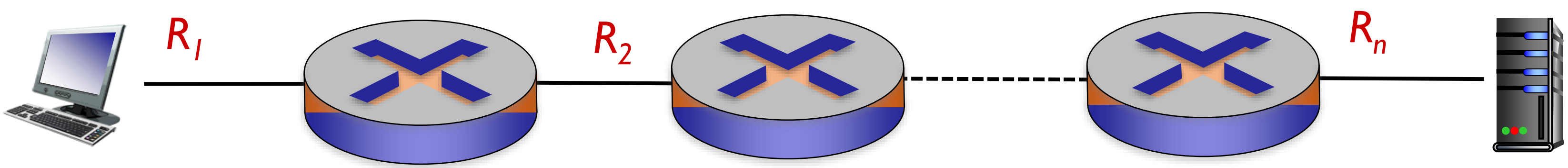


bottleneck link

link on end-end path that constrains end-end throughput



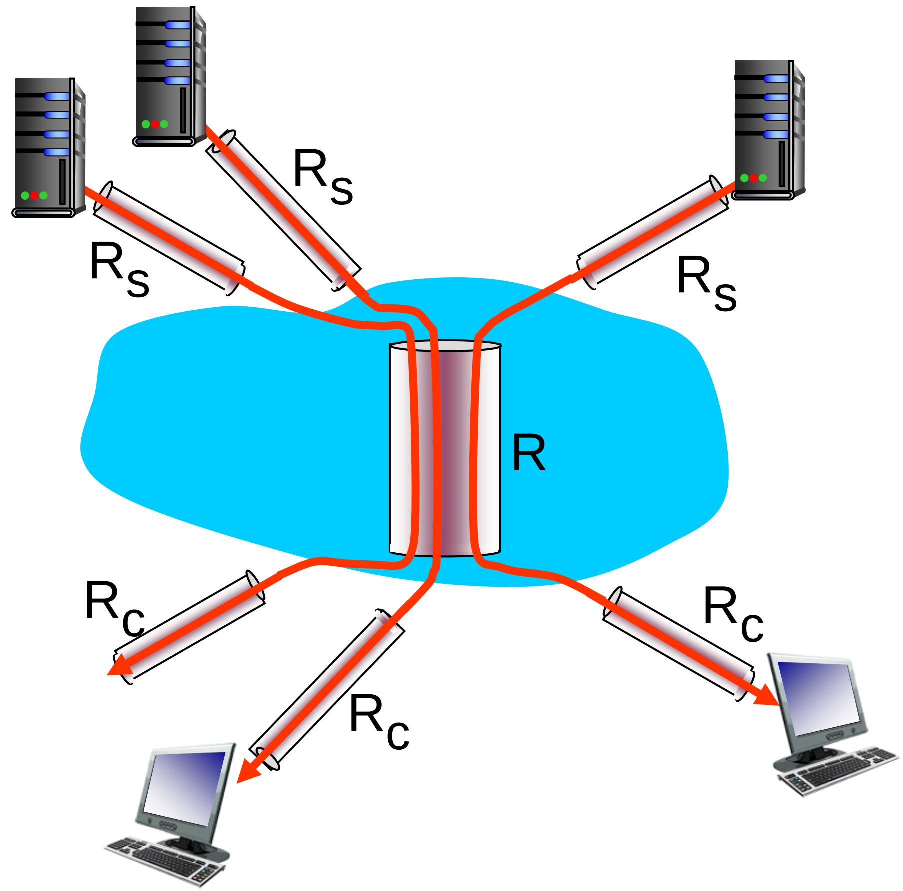
Throughput = min {Rs, Rc}



Throughput = min {R1, R2,..., Rn}

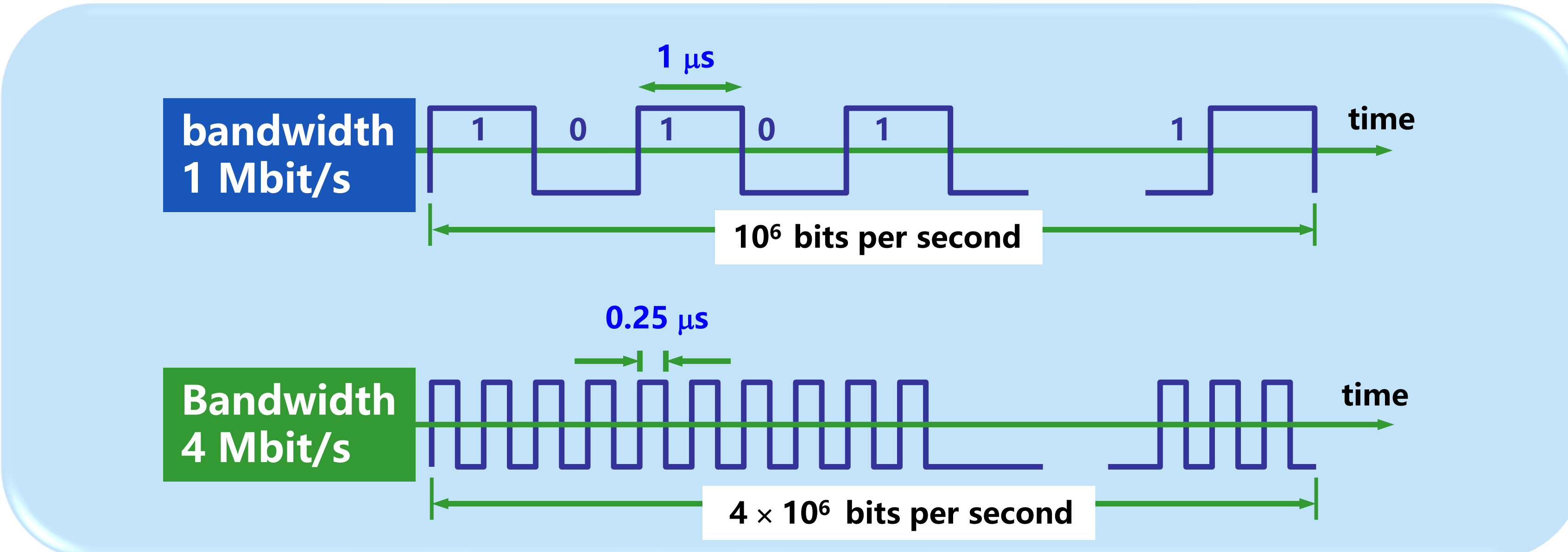
A more common example in Internet

- per-connection end-end throughput: $min(R_c, R_s, R/10)$
- in practice: R_c or R_s is often bottleneck



10 connections (fairly) share backbone bottleneck link *R* bits/sec

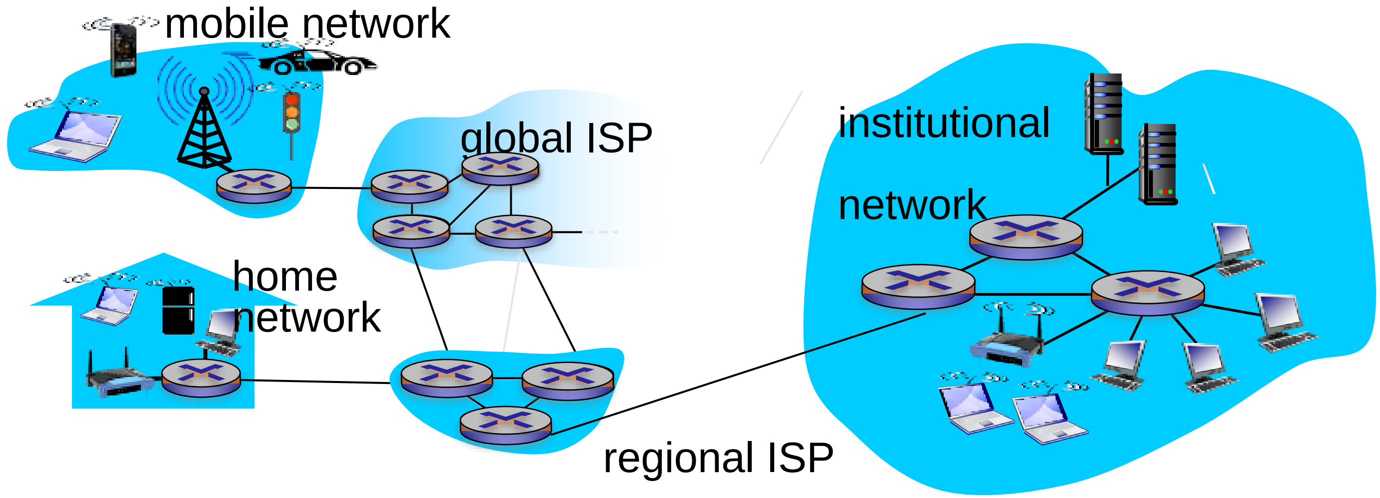
bandwidth



The width of the signal on the time axis narrows as the bandwidth increases.

Organizing the Internet

- Internet is a complex system:



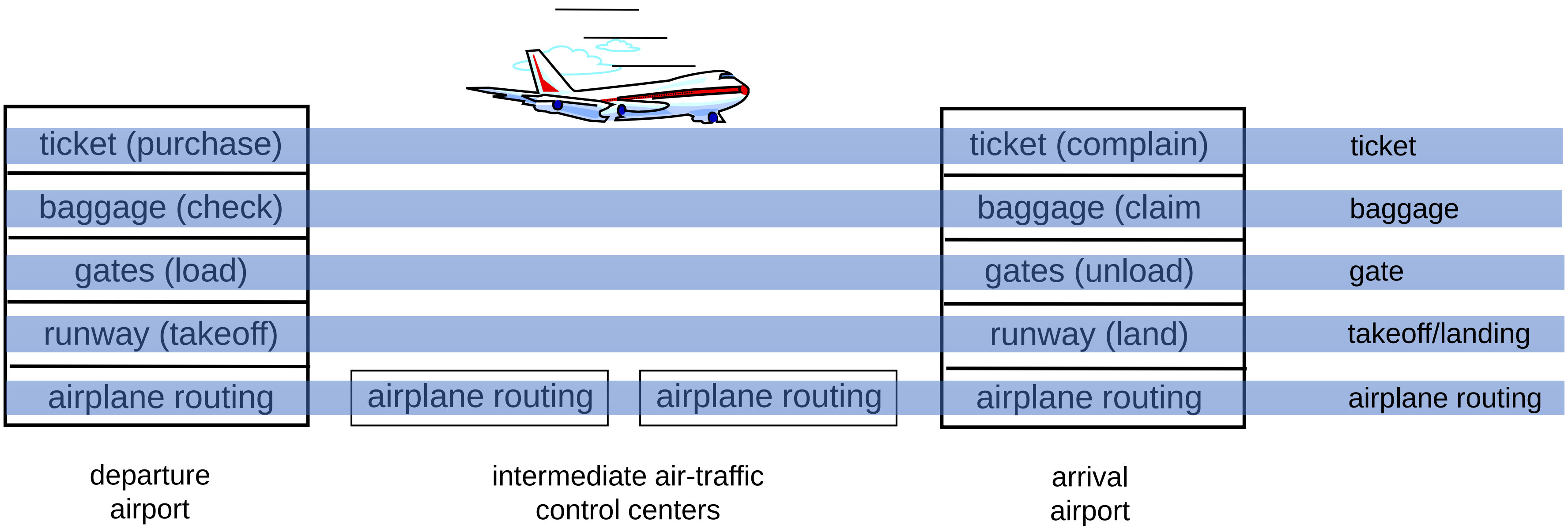
Question : is there any hope of organizing the structure of computer network?

Analogy: Organization of air travel

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

airplane routing

Layering of airline functionality



layers: each layer implements a service

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

Layered architectures

- Each layer is functionally independent
- Each layer has a defined interface to the previous & preceding layer
- Each layer builds on the previous layer
- Virtual communication takes place between layers at the same level
- Layered architectures are often called protocol stacks

Why layering?

dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer' s service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Pros & cons

pros

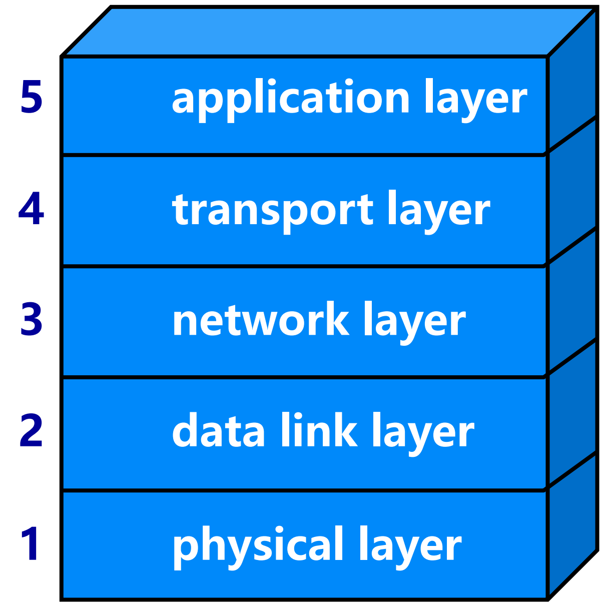
- independent
- flexible
- Separable structure
- eases maintenance
- Promote standardization

cons

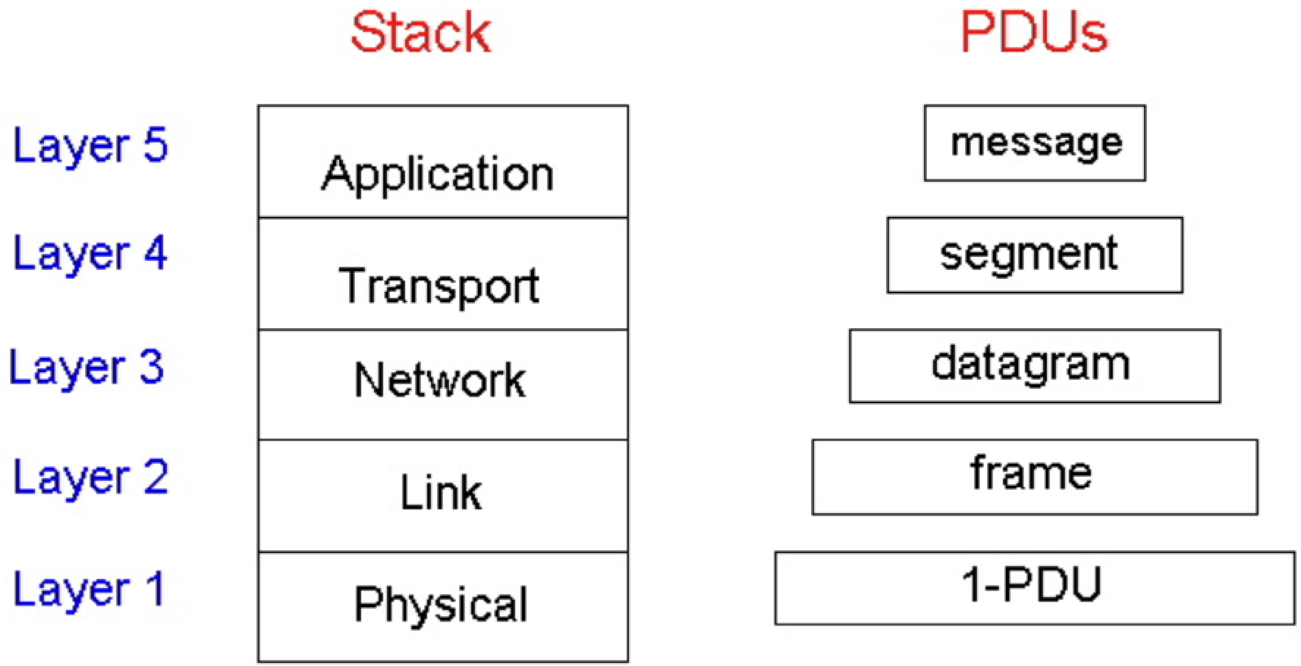
- not very efficient
- Similar function may appear in multi-layers with additional cost.

Protocol stacks of Internet

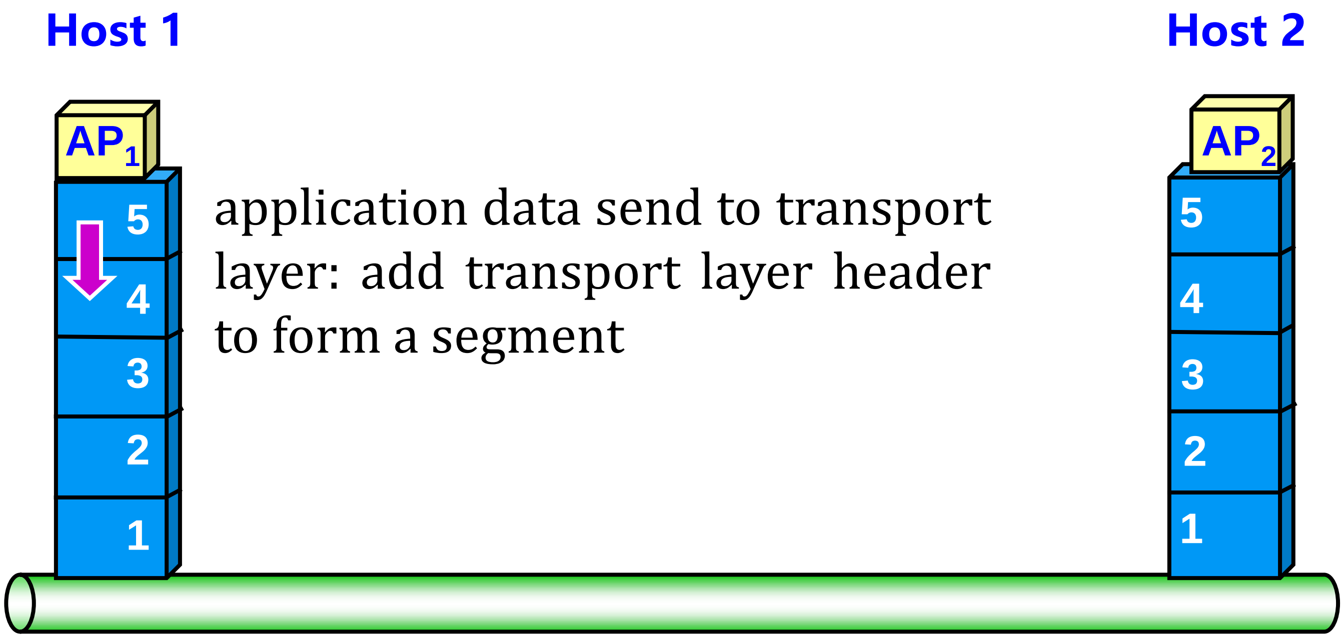
- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.111 (WiFi), PPP
- *physical:* bits “on the wire”



The protocol stack, and protocol data units

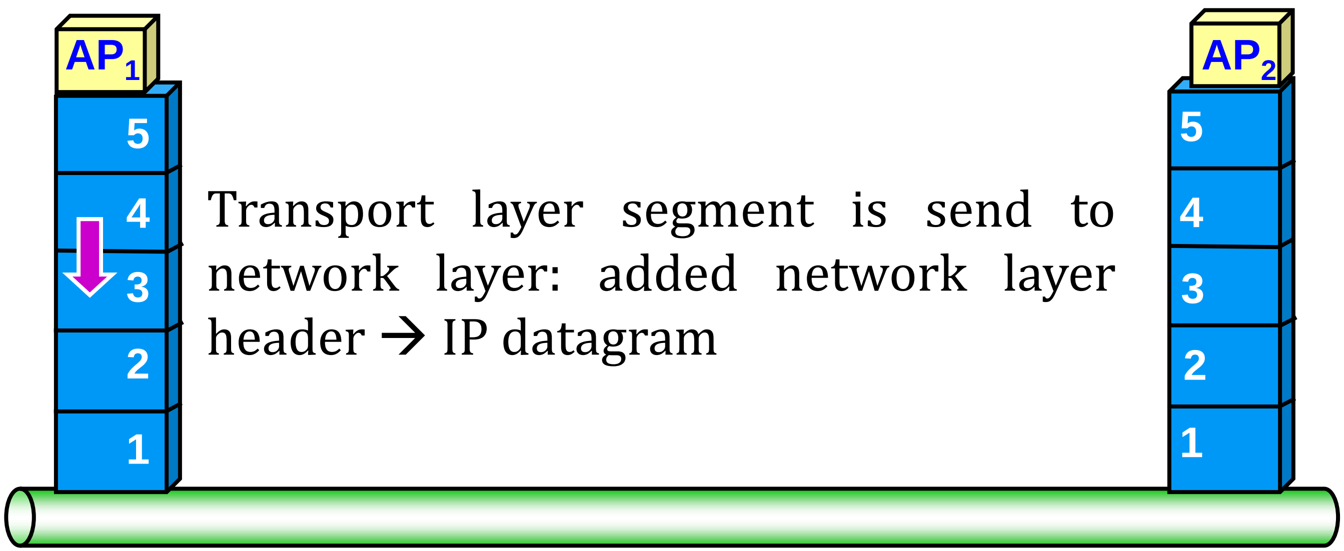


Example



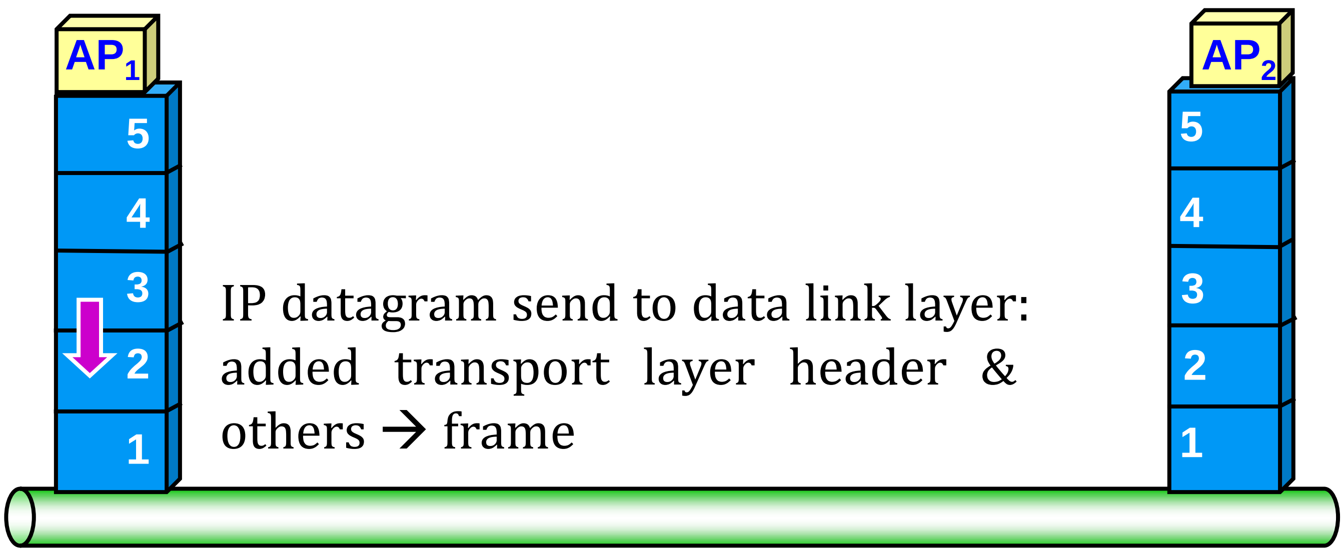
application data send to transport layer: add transport layer header to form a segment

Example



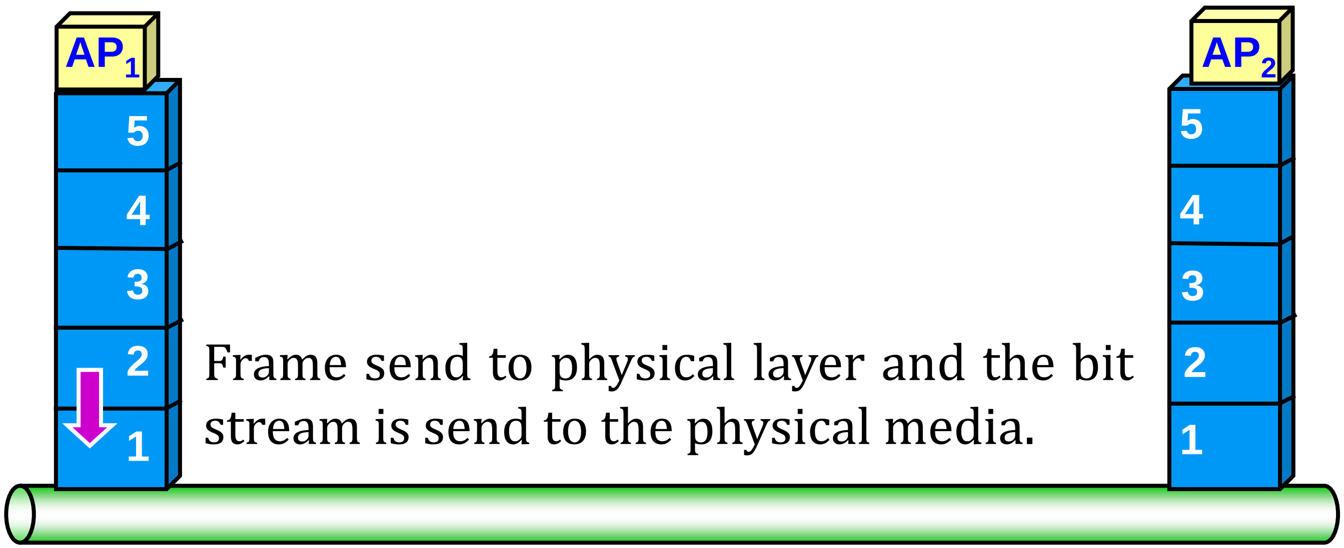
Transport layer segment is send to network layer: added network layer header → IP datagram

Example



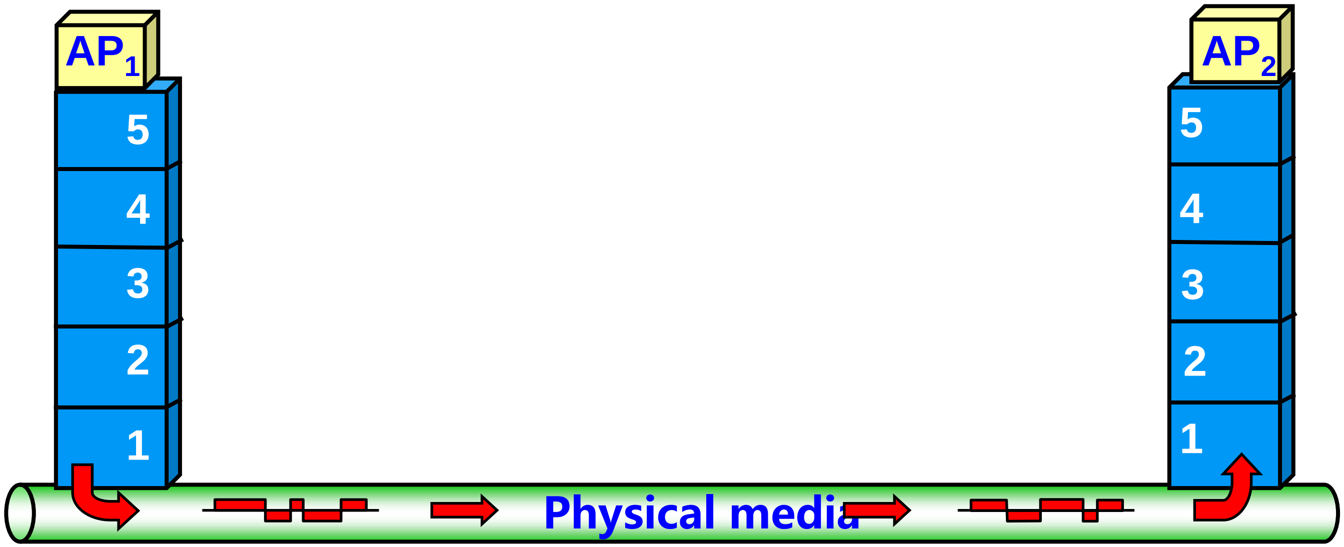
IP datagram send to data link layer: added transport layer header & others → frame

Example



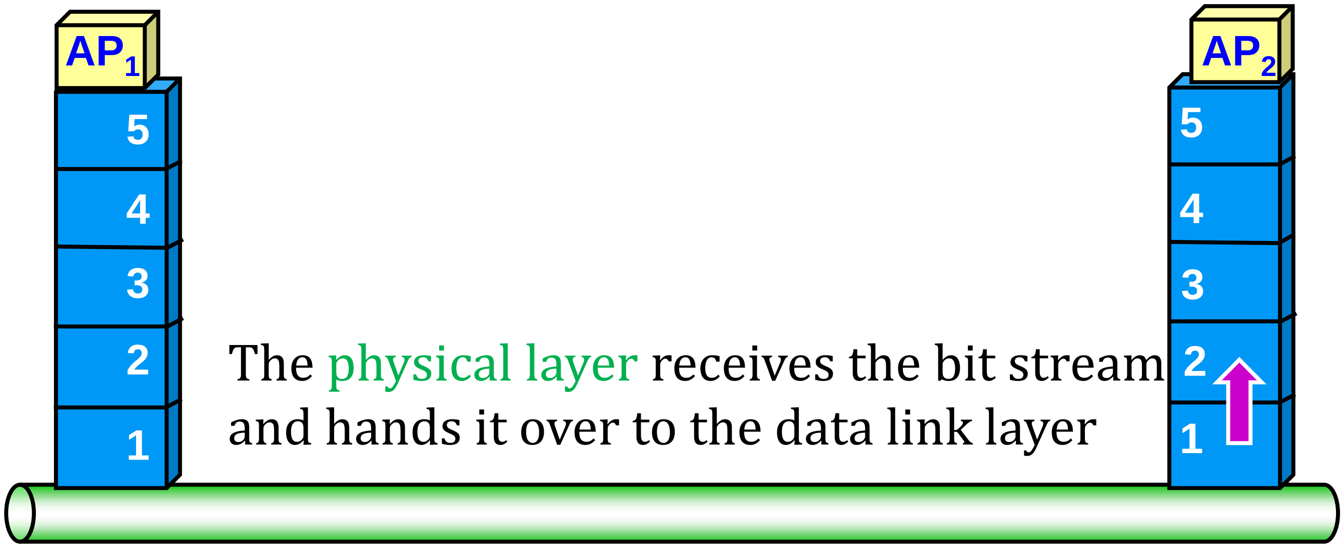
Frame send to physical layer and the bit stream is send to the physical media.

Example



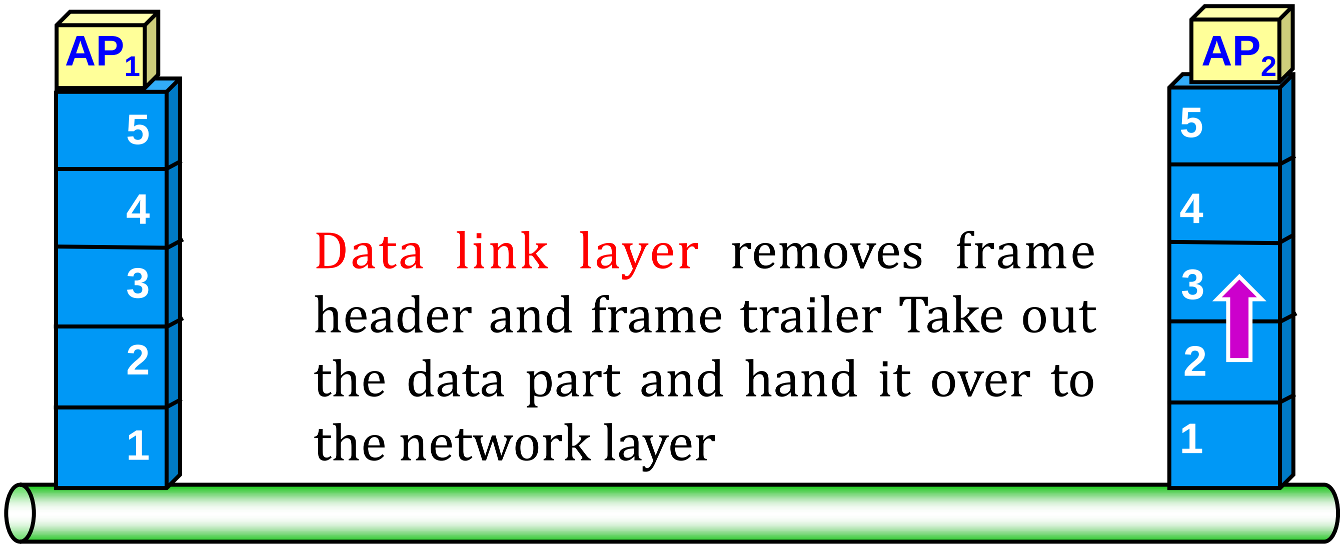
- Electrical signals (or optical signals) propagate in physical media
- From the physical layer of the sender to the physical layer of the receiver

Example



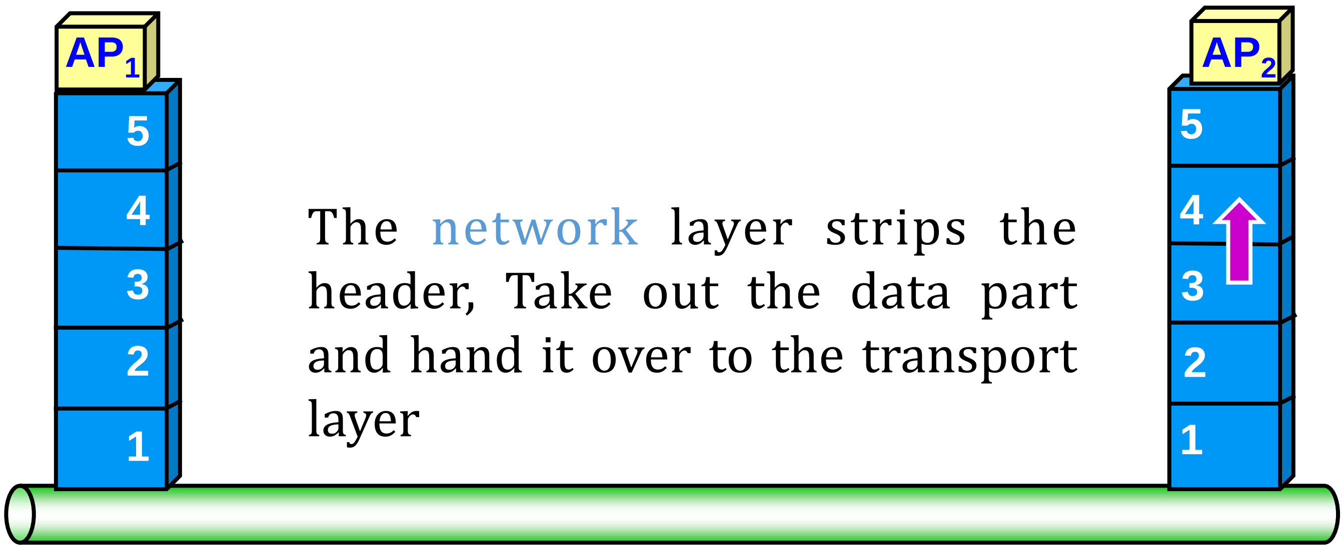
The physical layer receives the bit stream and hands it over to the data link layer

Example



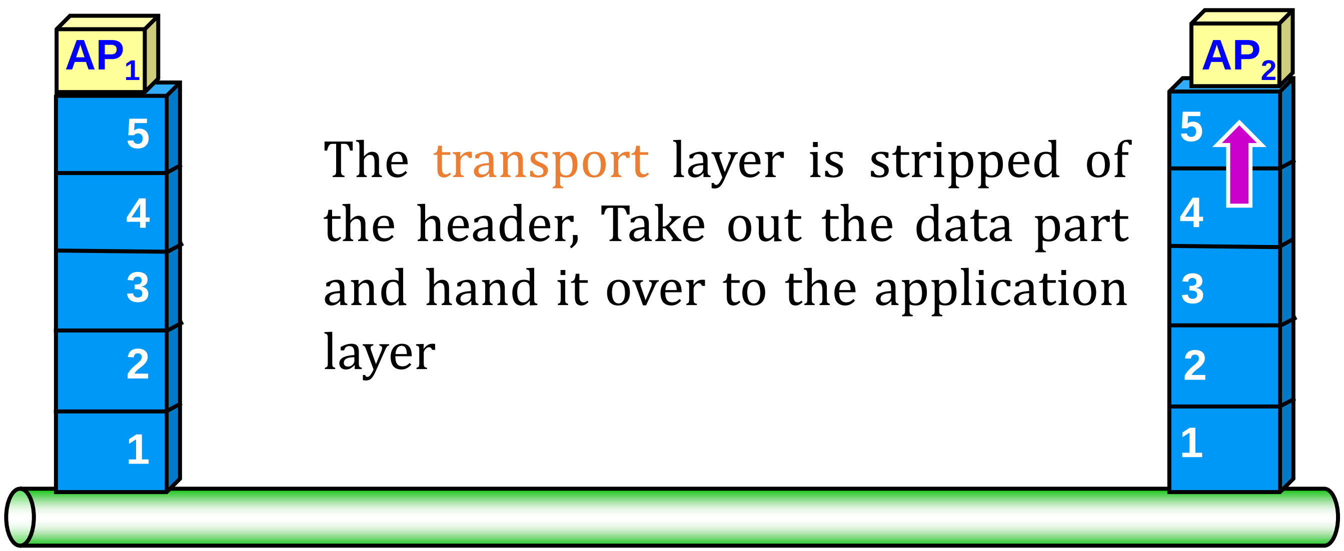
Data link layer removes frame header and frame trailer Take out the data part and hand it over to the network layer

Example



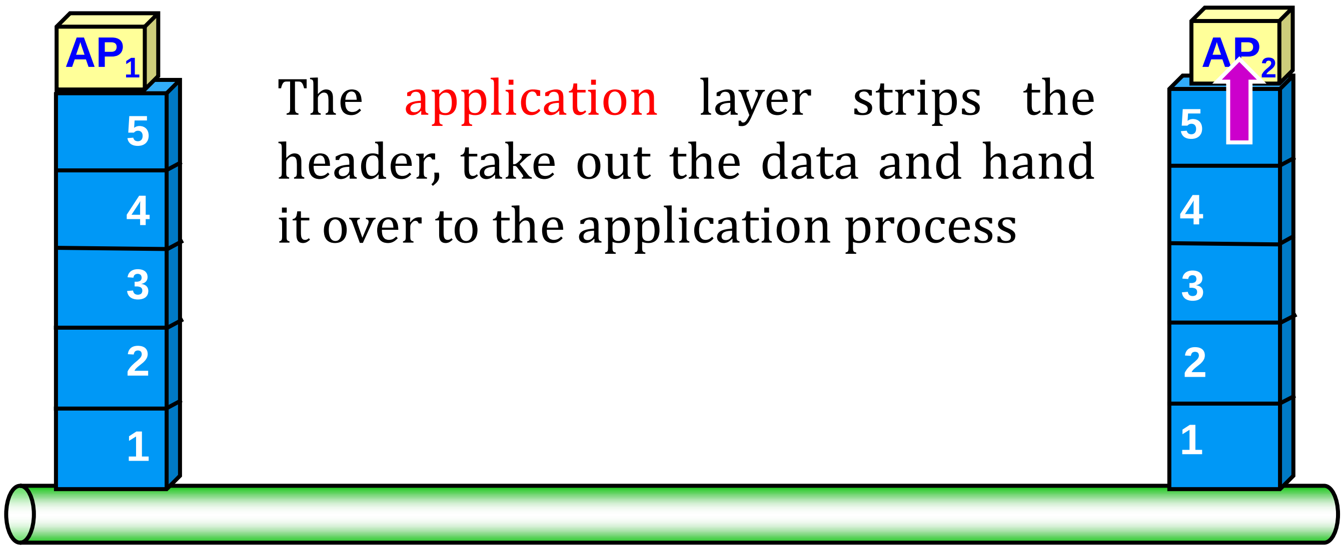
The network layer strips the header, Take out the data part and hand it over to the transport layer

Example



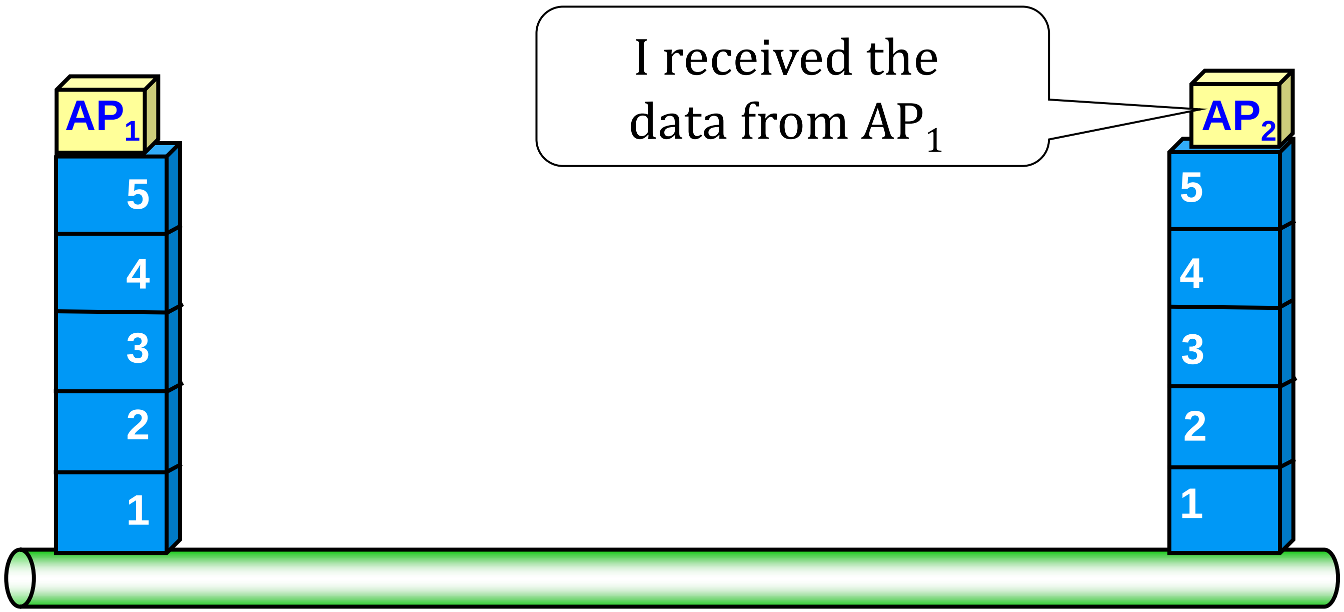
The transport layer is stripped of the header, Take out the data part and hand it over to the application layer

Example



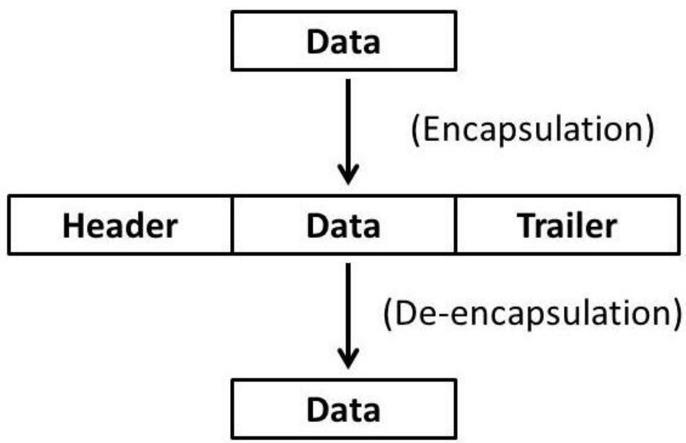
The application layer strips the header, take out the data and hand it over to the application process

Example

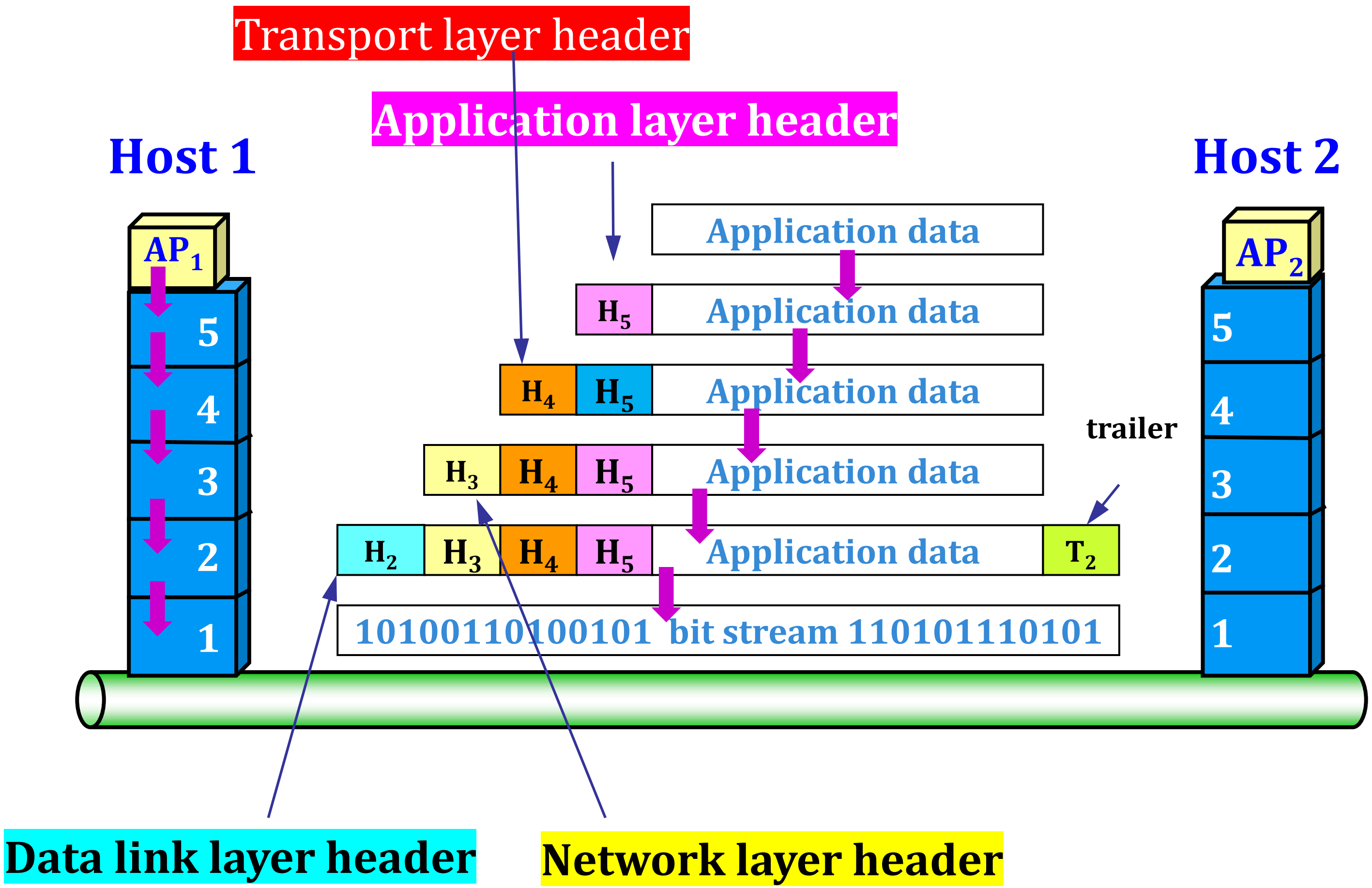


Encapsulation

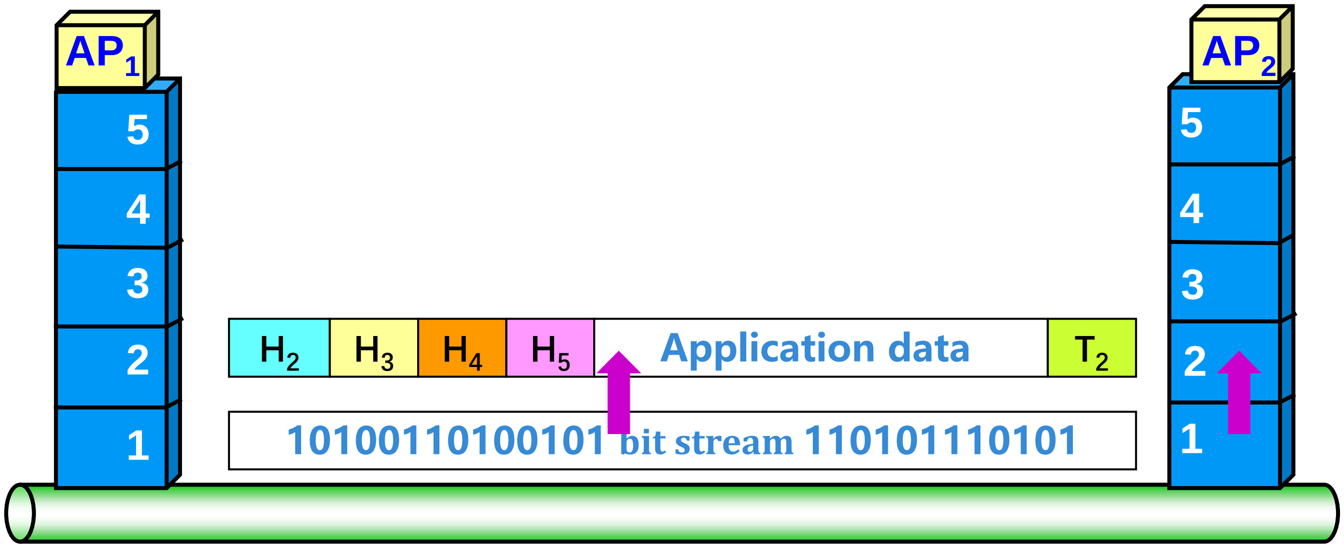
Definition : A process of adding control information as a message passes through the layered model



<u>Encapsulation and De-encapsulation</u>

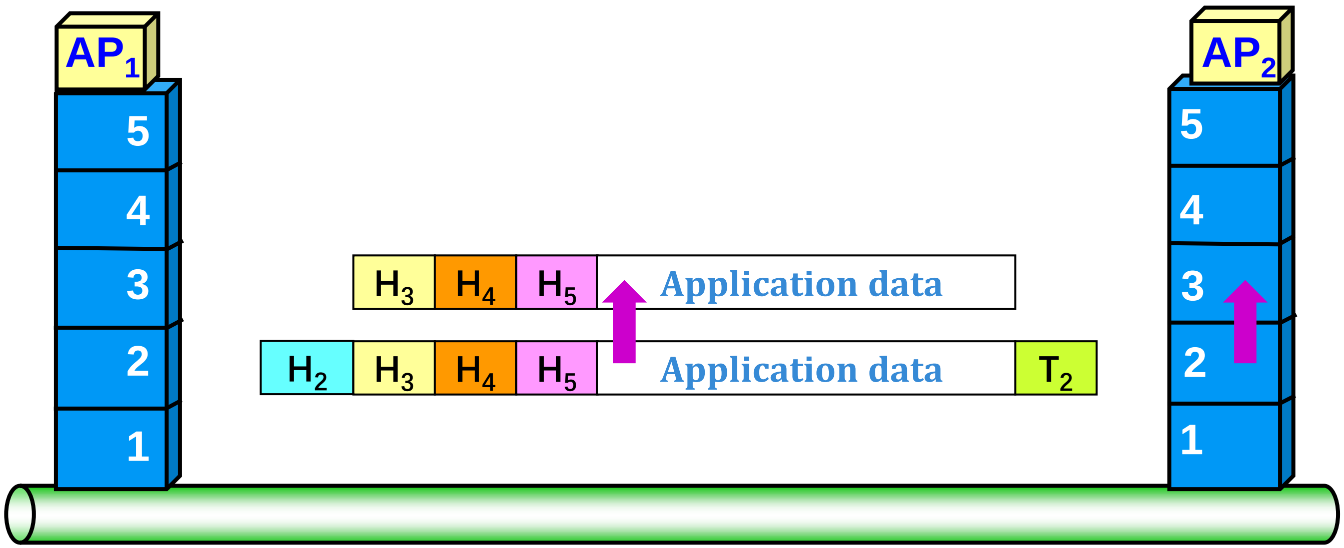


Transport layer header
Application layer header
Host 1
Host 2
AP1
AP2
5
4
3
2
1
Application data
H5
H4
H3
H2
T2
trailer
101001101001011 bit stream 110101110101
Data link layer header
Network layer header

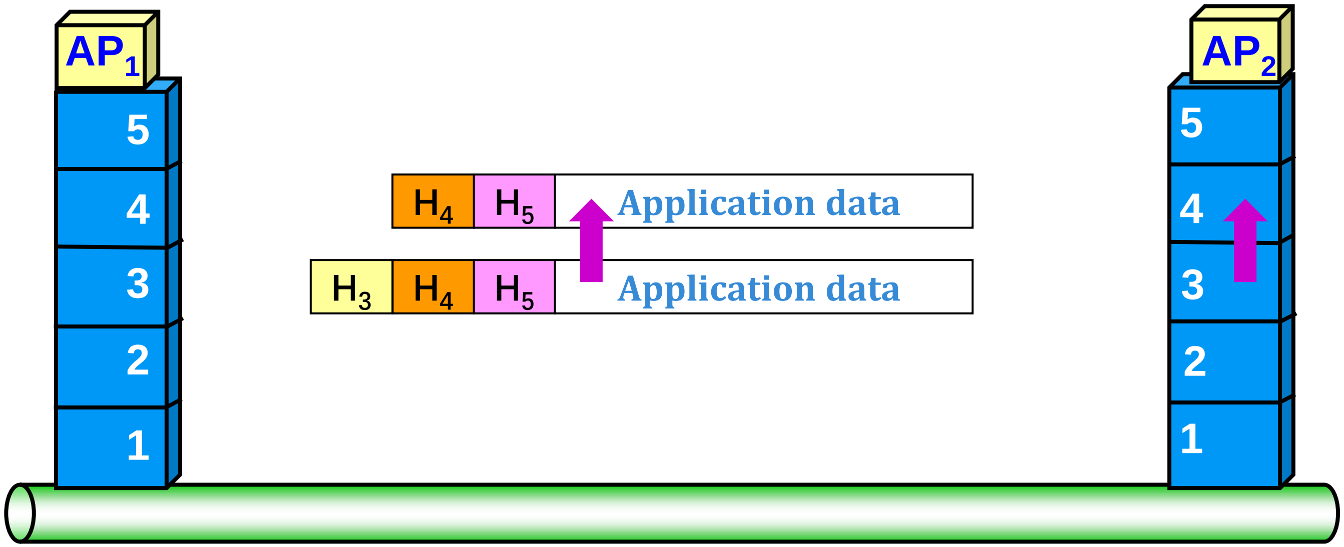


- 1. The physical layer receives the bit stream and hands it over to the data link layer

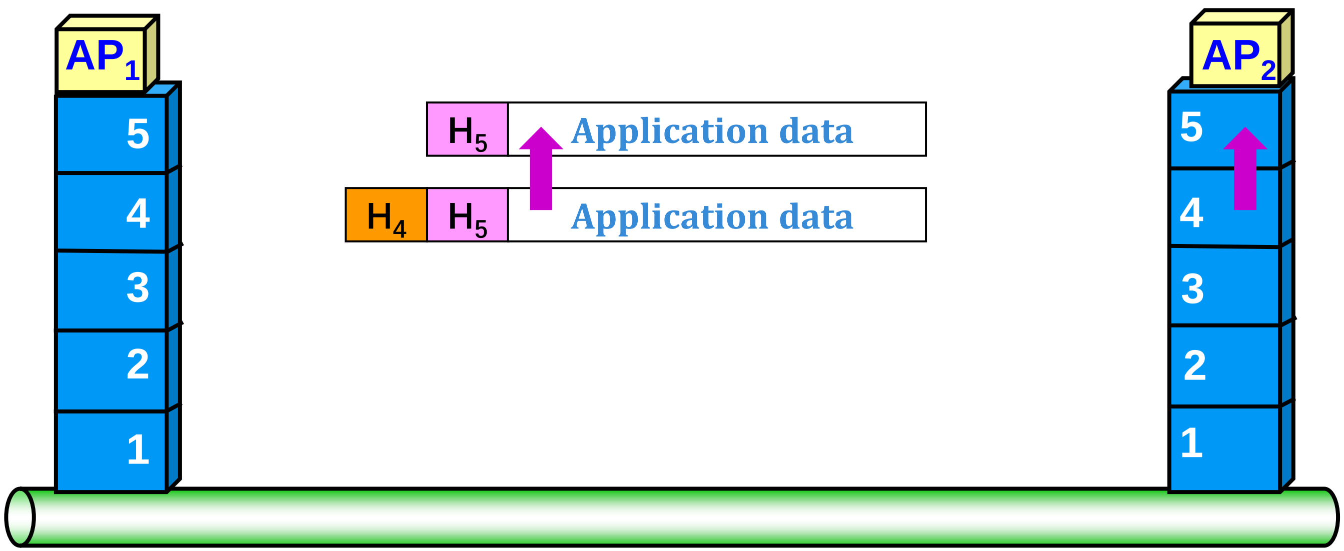
- 2， Data link layer
  - removes frame header & trailer,
  - take out the data part
  - hand it over to the network layer



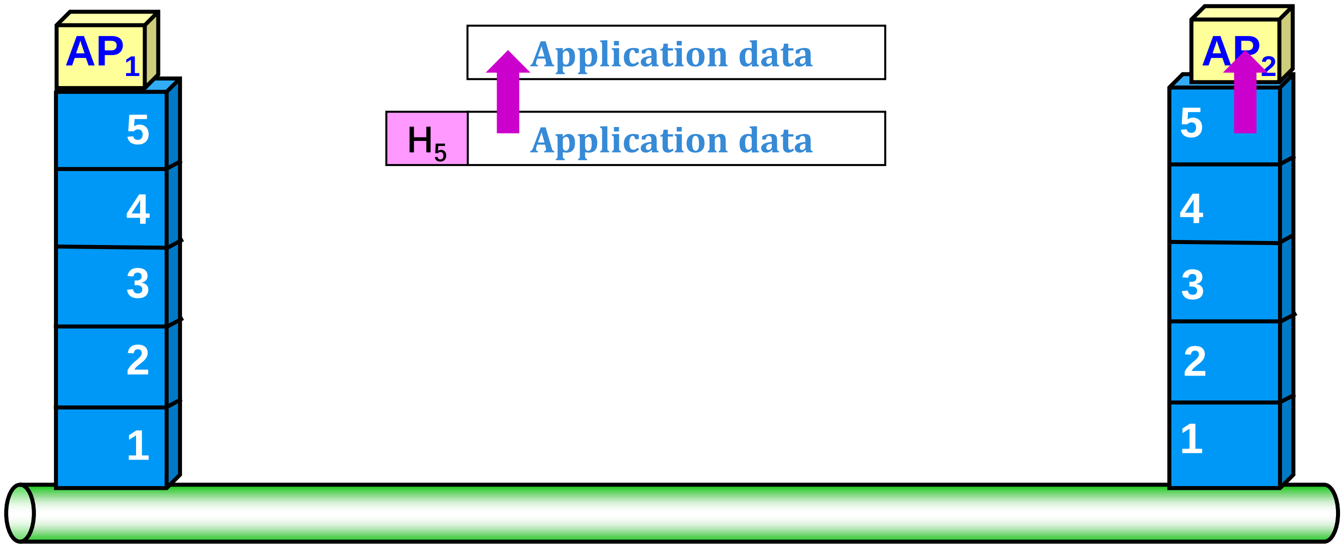
- 3， The network layer
  - strips the header,
  - take out the data part
  - hand it over to the transport layer

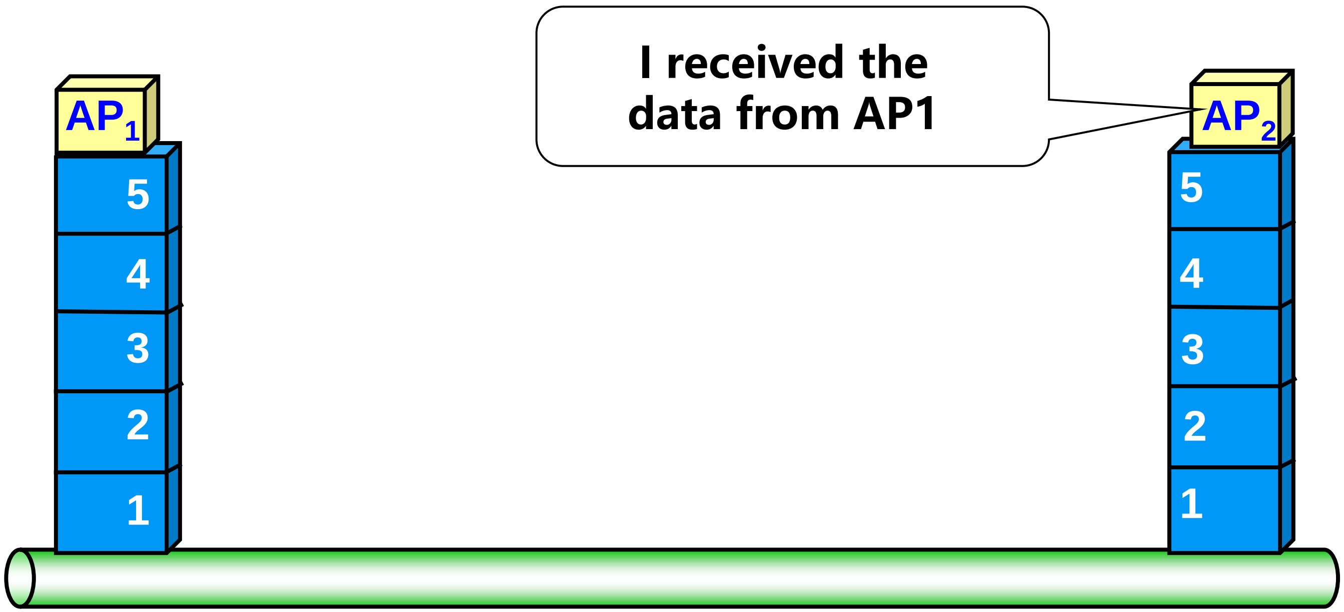


- 4， The transport layer
  - removes the header,
  - take out the data part
  - hand it over to the application layer



- 5， The application layer
  - strips the header
  - take out the data
  - hand it over to the application process

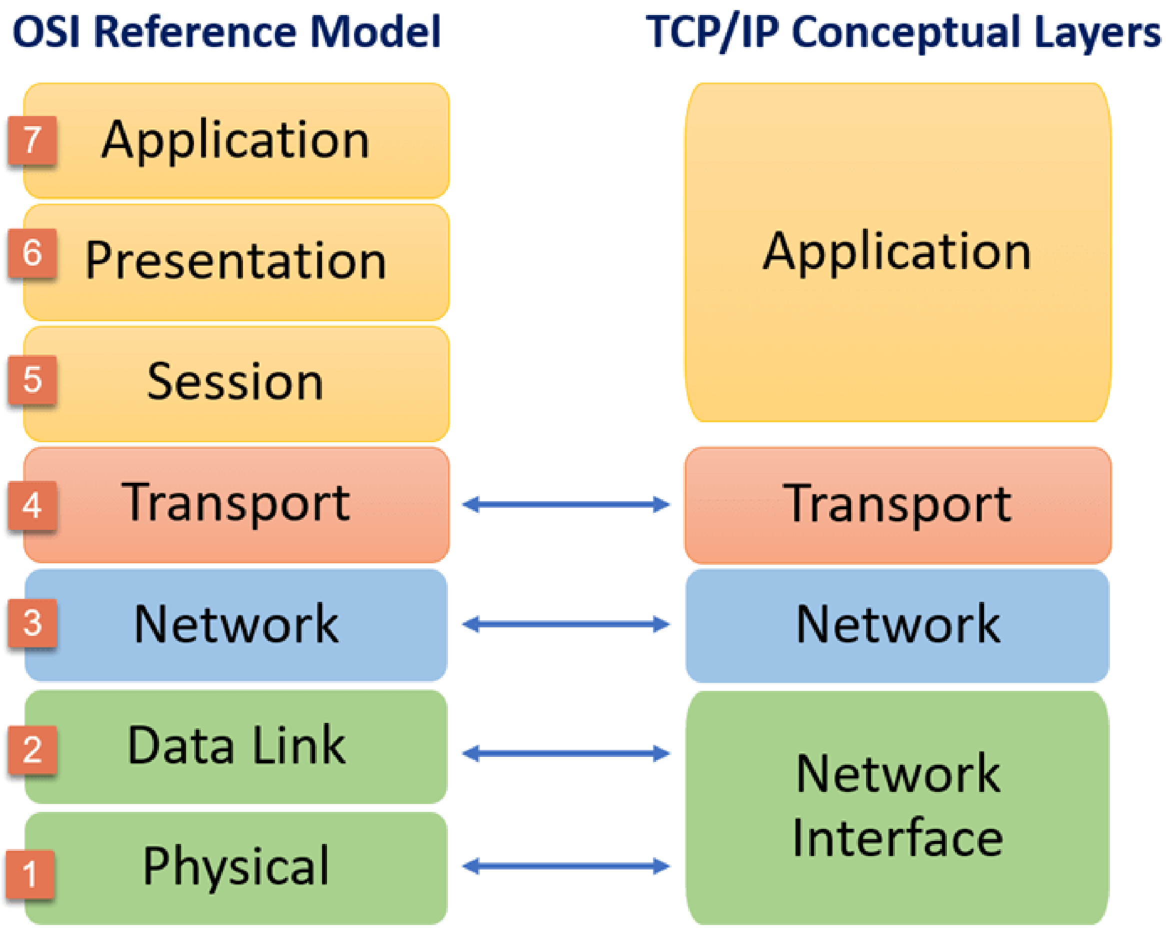



I received the data from AP1
AP1
5
4
3
2
1
AP2
5
4
3
2
1

Other protocol stack exists – OSI model

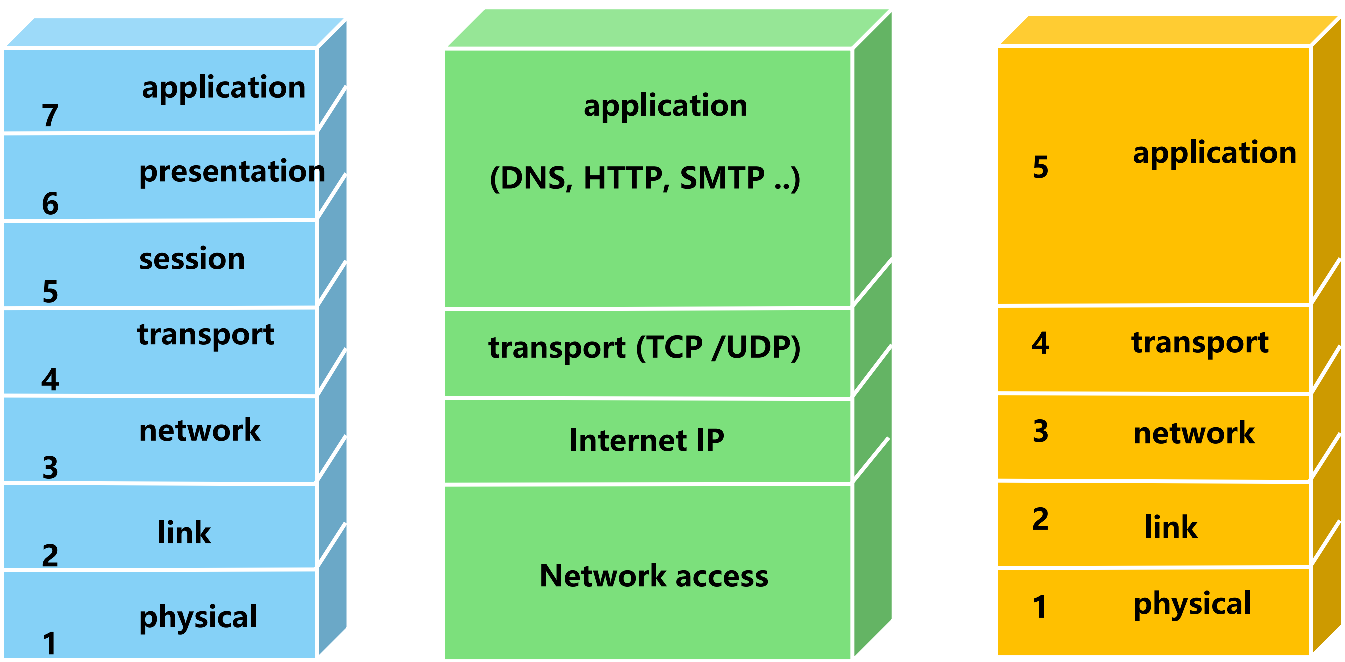
7	Application Layer	Human-computer interaction layer, where applications can access the network services
6	Presentation Layer	Ensures that data is in a usable format and is where data encryption occurs
5	Session Layer	Maintains connections and is responsible for controlling ports and sessions
4	Transport Layer	Transmits data using transmission protocols including TCP and UDP
3	Network Layer	Decides which physical path the data will take
2	Data Link Layer	Defines the format of data on the network
1	Physical Layer	Transmits raw bit stream over the physical medium

TCP/IP model



OSI Reference Model
TCP/IP Conceptual Layers
7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data Link
1 Physical
Application
Transport
Network
Network Interface

Network protocol model(s)



(a) OSI (b) TCP/IP (c) Internet

TAKEAWAYS

- Packet switching
- (Message switching)

- Network performance
  - Delay
  - Loss
  - Throughput
  - Bandwidth

- Layered protocol model
  - 5-layered model
  - Encapsulation & Decapsulation
  - OSI model & TCP/IP model

- End of Chapter 1 -