CSCI 466: Networks

Lecture 3: OSI Model, Packet Forwarding, Network Performance

Reese Pearsall

Fall 2022

Announcements

- TA: **Justin Mau**
  - Office Hours: TR 12:00 – 1:00 in Barnard Hall 259
  - Email: justindmau@gmail.com

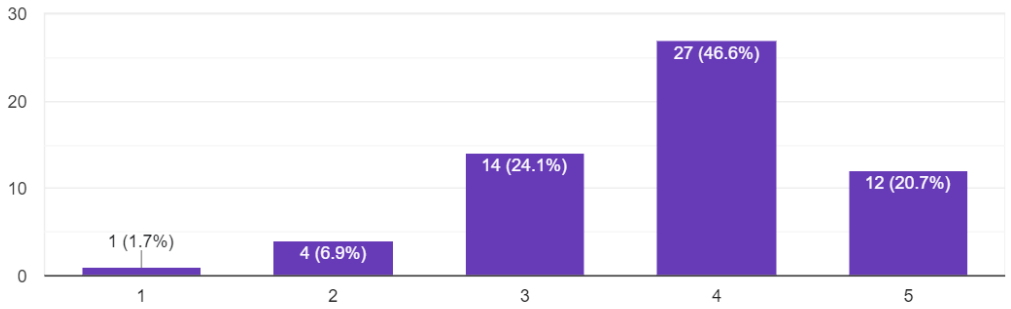
- I am working on improving the audio quality for recordings

Questionnaire Stuff

How comfortable are you with Python?

58 responses

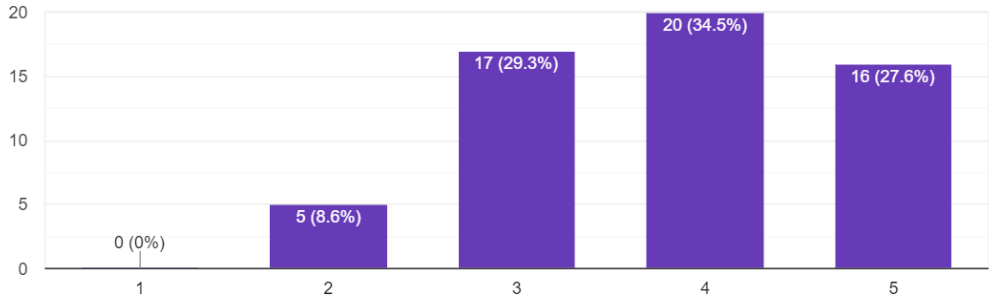
Copy



How comfortable are you with Git/Github

58 responses

Copy

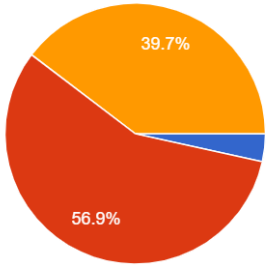


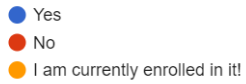
It makes my life much less stressful knowing people have used Git ☺

Have you taken CSCI 476- Computer Security?

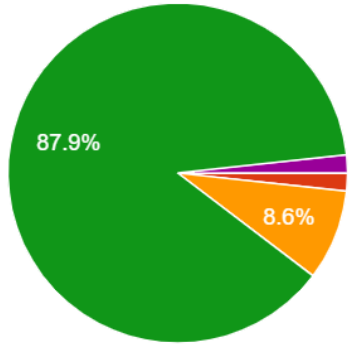
58 responses

Copy





To the 39.7%-- There will be some overlap



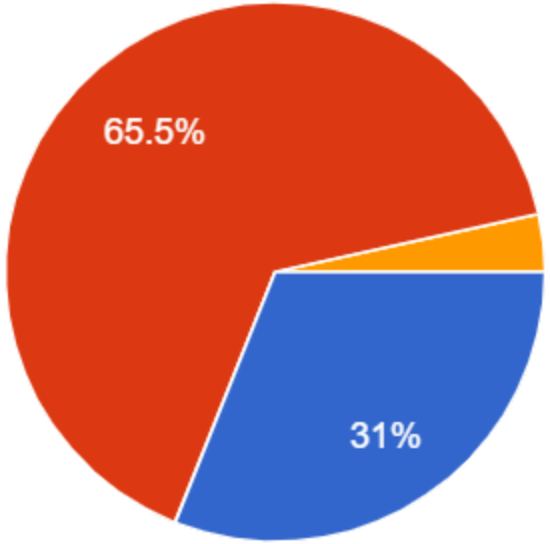
"I like group projects",
"I hate group projects",
"I am a hardcore procrastinator"

Questionnaire Stuff

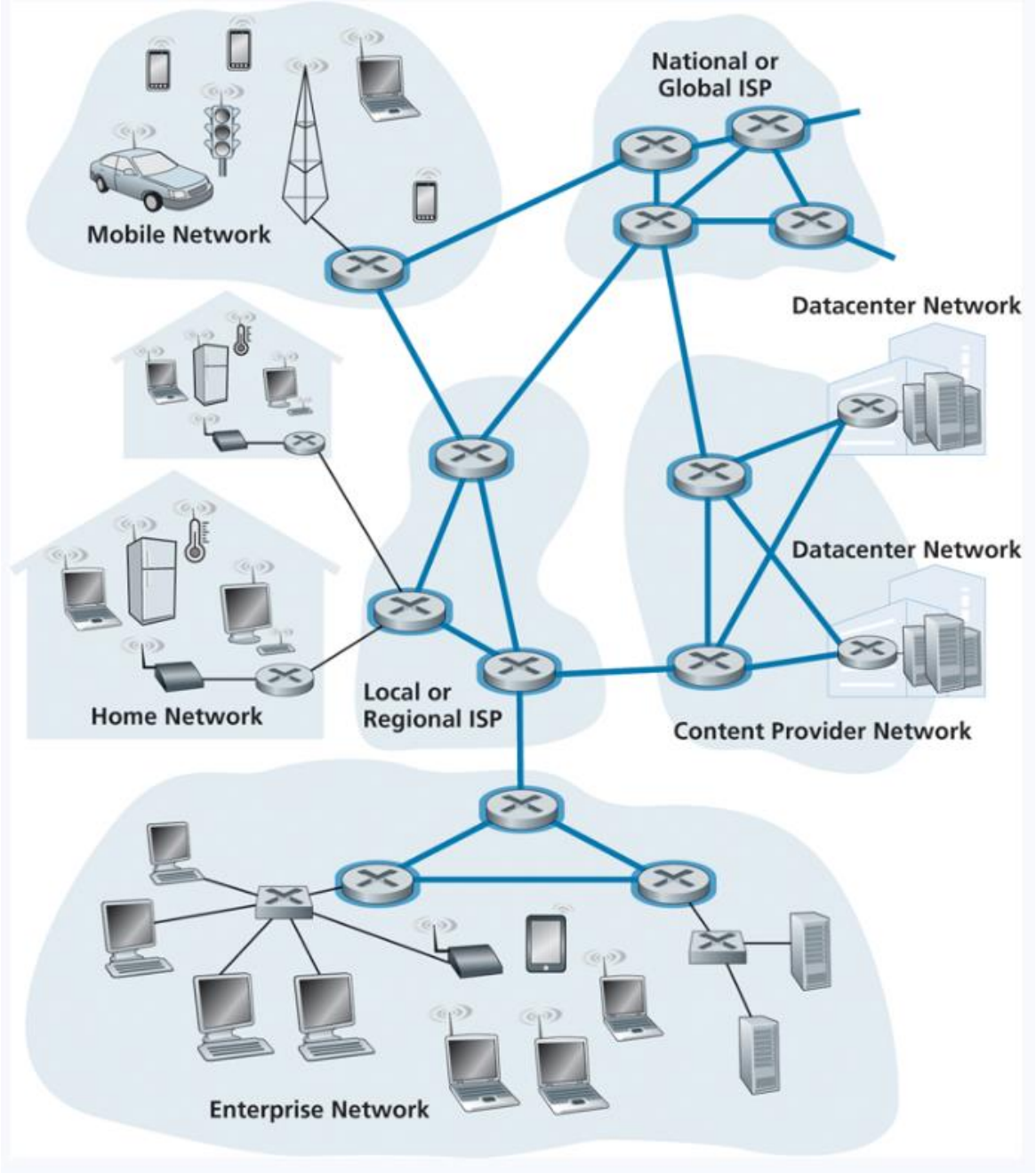
Which variable naming convention do you prefer?

58 responses

Copy



- Snake Case
- Camel Case
- I am a crazy person that uses a different convention



The internet is a *network of networks*, connected by **packet switches** and **communication links**

Messages going from A to B are split into **packets**

"Good morning, I hope you are having a good day!"

To: Host A
John Paxton
192.42.98.11

From: Host B
Reese Pearsall
192.5.223.42

```
Good morning, I hope you
are having a good day!
```

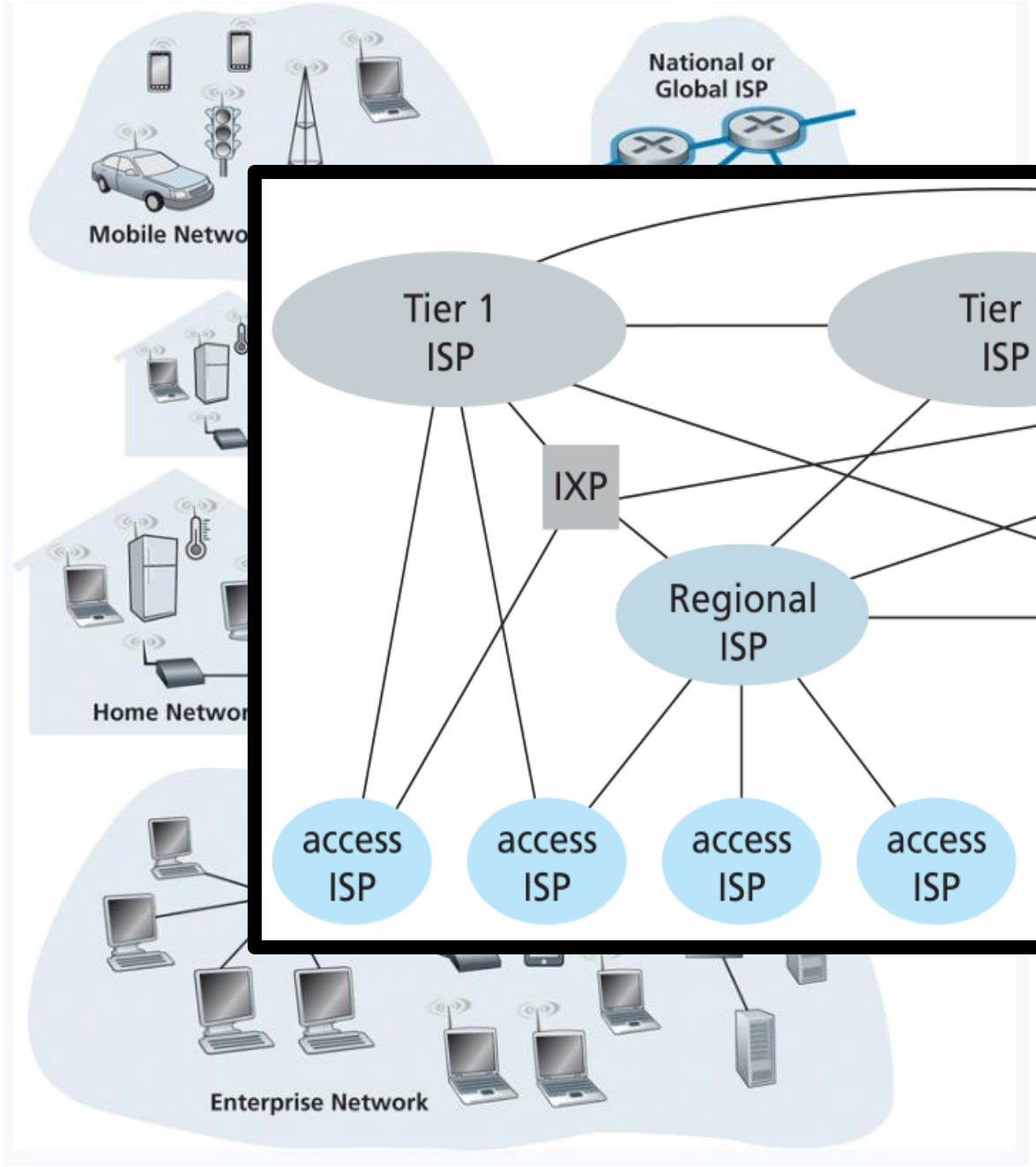
Generated Packet

Communication links have different transmission rates

10 Mbps

500 kbps

100 kbps



The internet is a *network of networks*, connected by
on links

into **packets**

y!"

Generated Packet

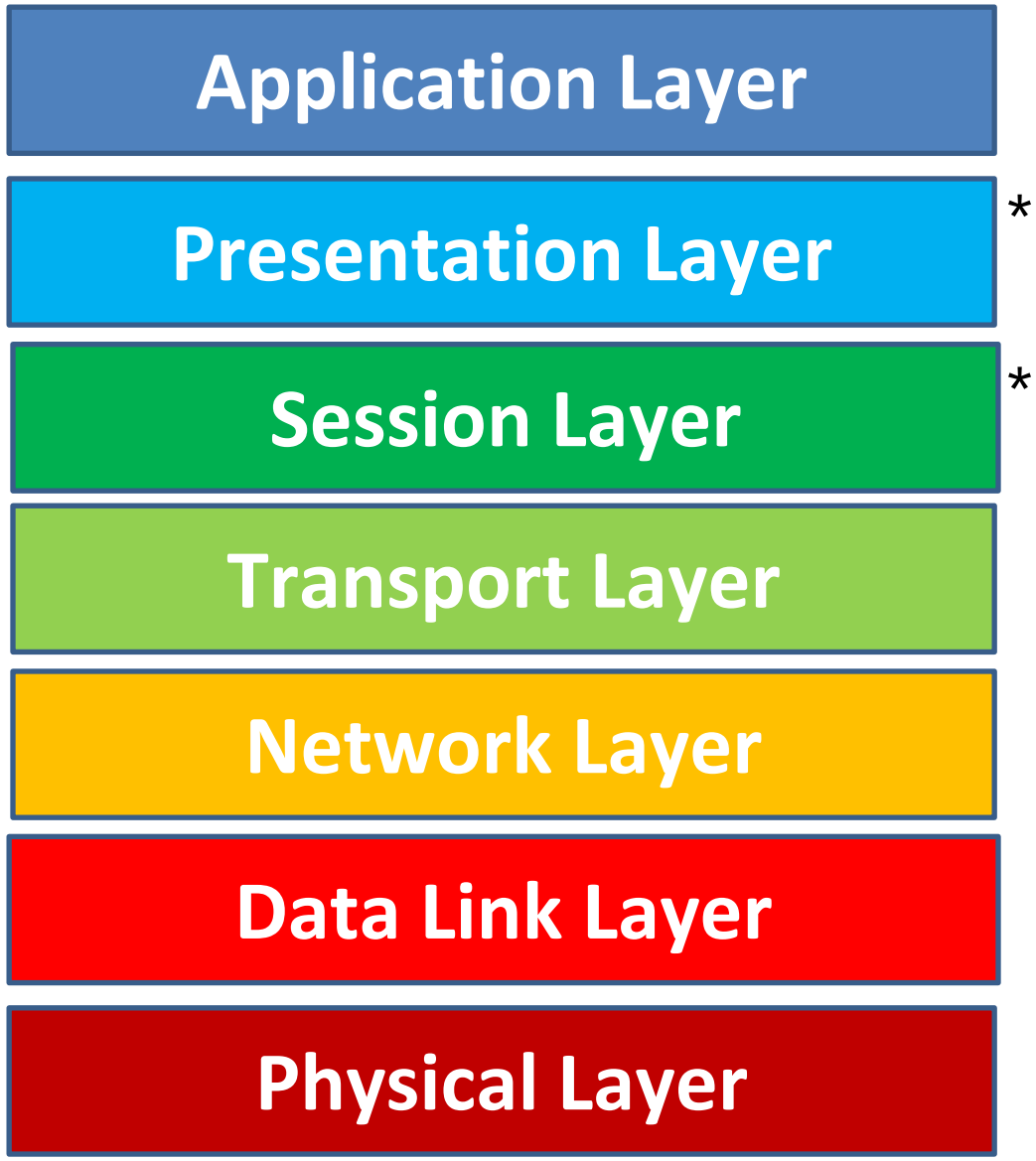
smission rates

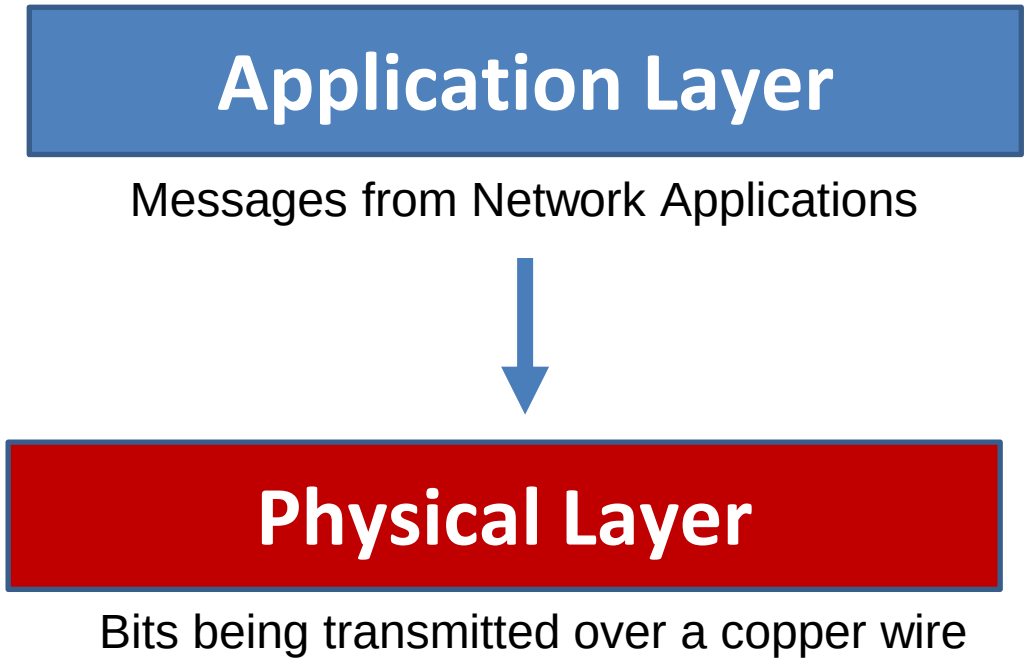
10 Mbps

500 kbps

100 kbps

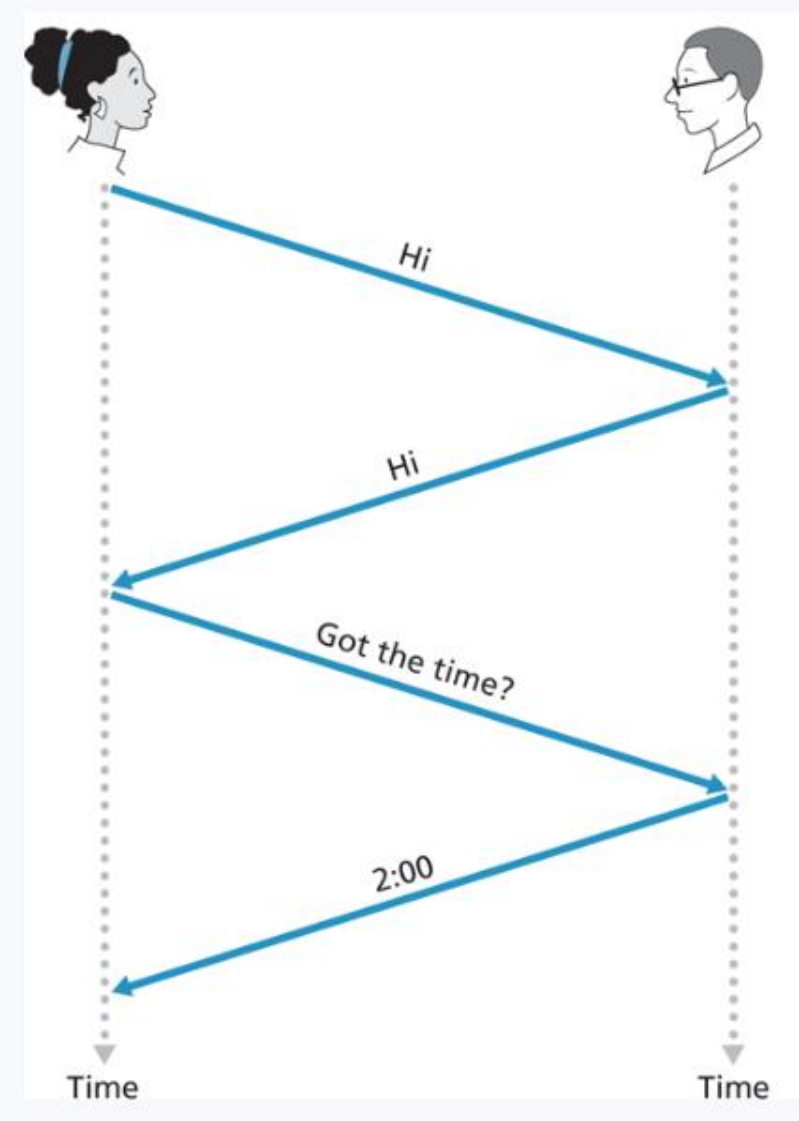
OSI Model



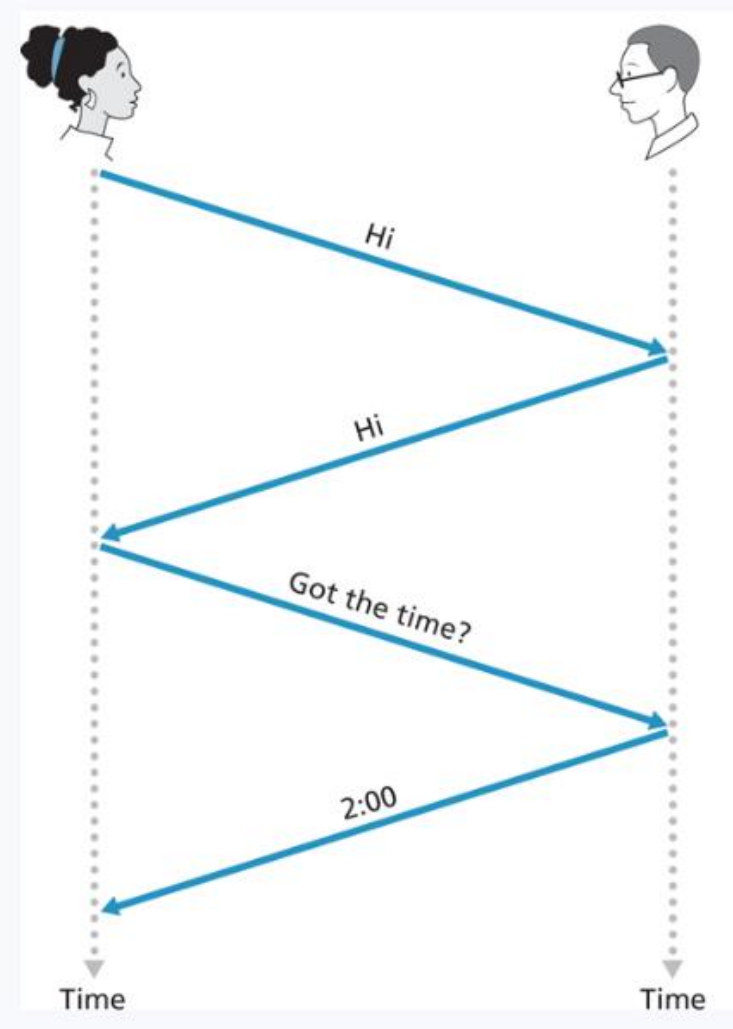


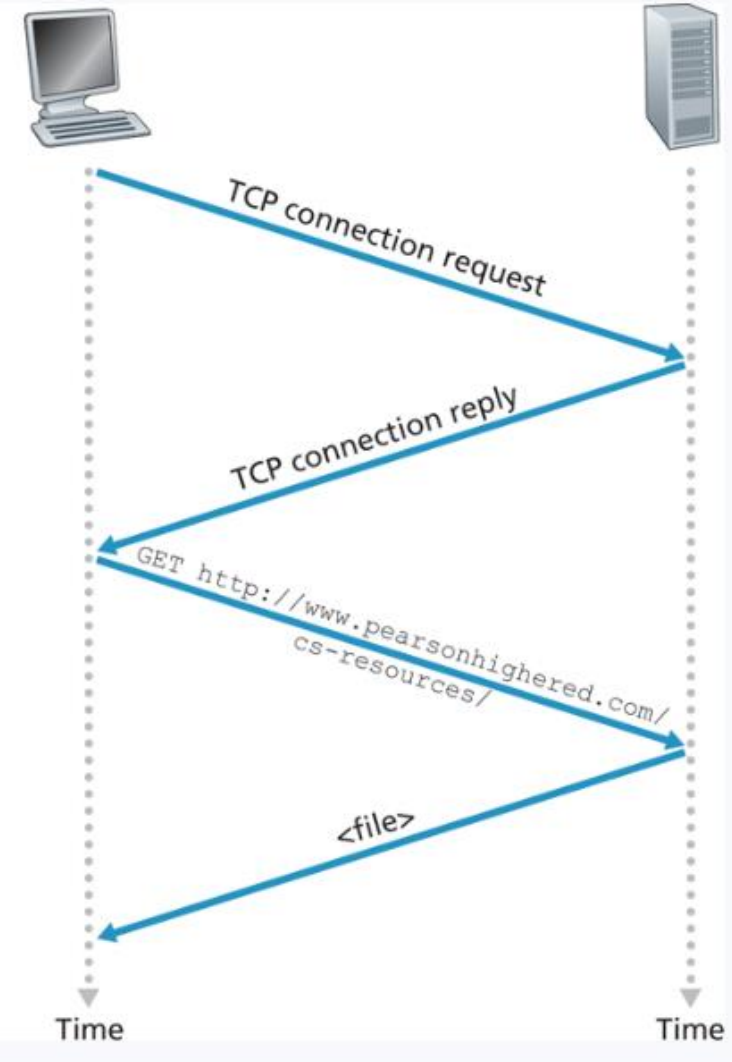
In the textbook, they condense it to a 5-layer model, but 7 layers is what is most used

What is a protocol?

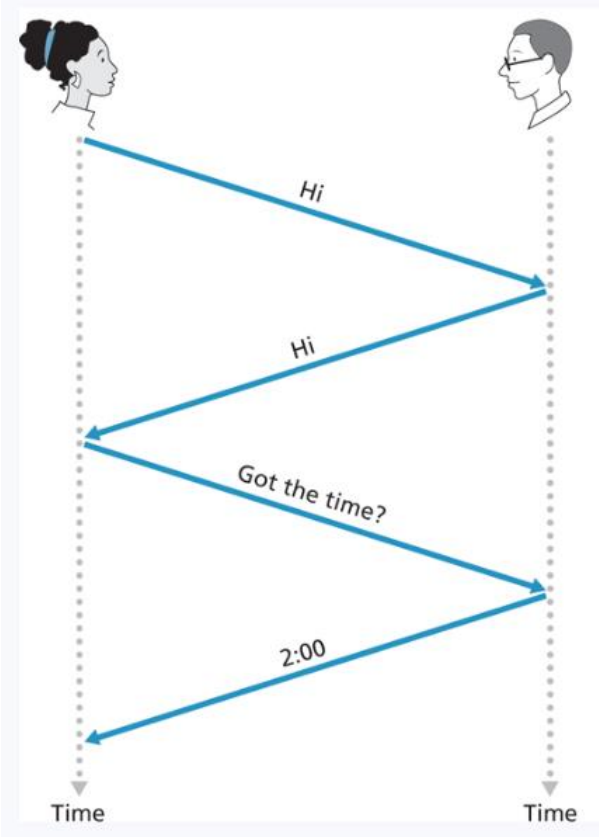


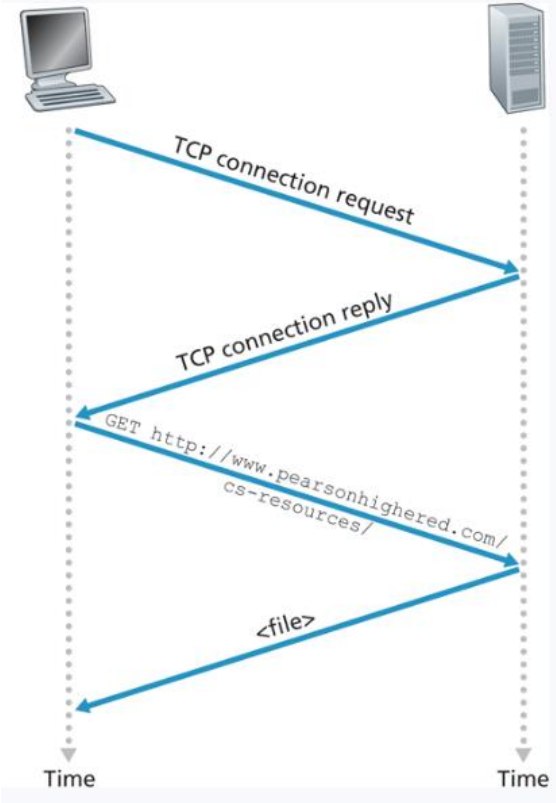
What is a protocol?





What is a protocol?





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

Application Layer

The layer which interacts directly with applications and provides necessary protocols and services for web applications

Humans interact with this layer

Data from user → Application Layer → Sent to next layer down

Application Layer

The layer which interacts directly with applications and provides necessary protocols and services for web applications

Humans interact with this layer

Data from user → Application Layer → Sent to next layer down

Search query on website →

```
GET /index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
Accept: text/html
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

→ Sent to presentation layer

Application Layer

The layer which interacts directly with applications and provides necessary protocols and services for web applications

Humans interact with this layer

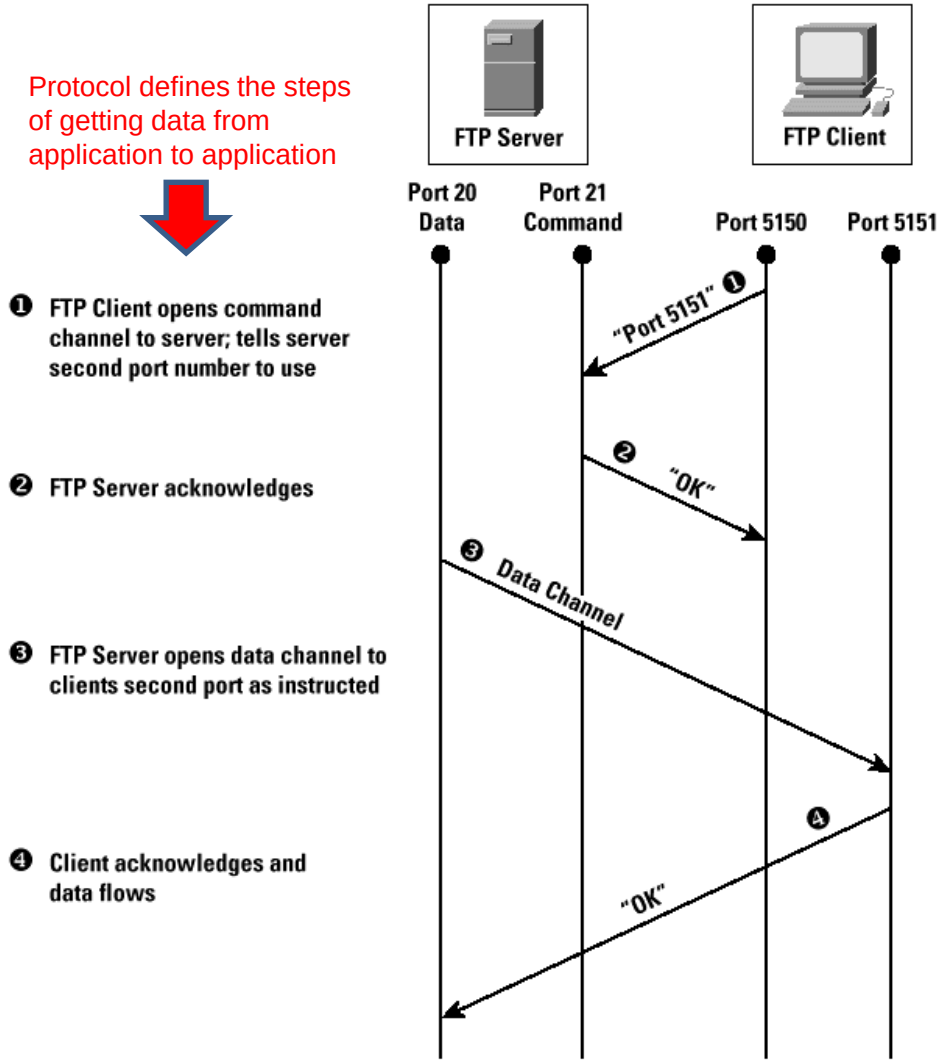
Data from user → Application Layer → Sent to next layer down

Search query on website →

```
GET /index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
Accept: text/html
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

→ Sent to presentation layer

Protocol defines the steps of getting data from application to application



Application Layer

The layer which interacts directly with applications and provides necessary protocols and services for web applications

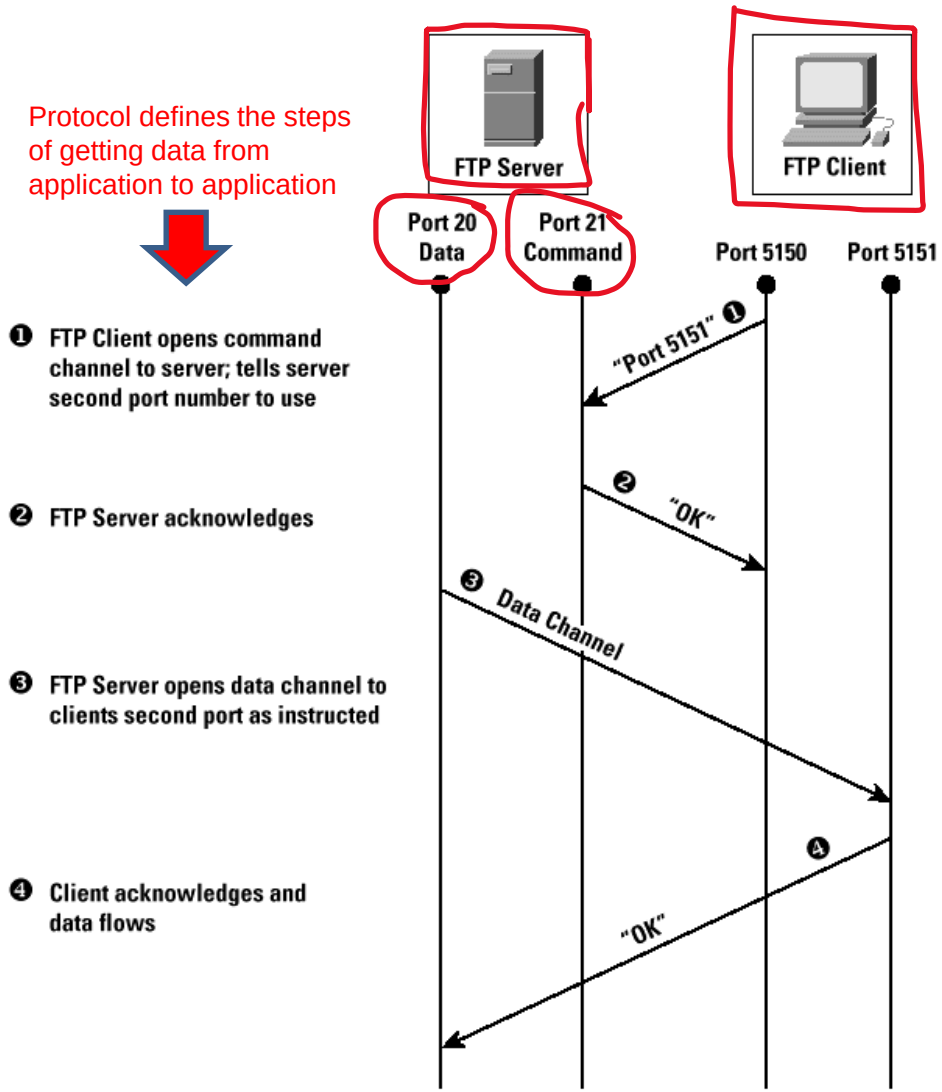
Humans interact with this layer

Data from user → Application Layer → Sent to next layer down

Search query on website →

```
GET /index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
Accept: text/html
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

→ Sent to presentation layer



Presentation Layer

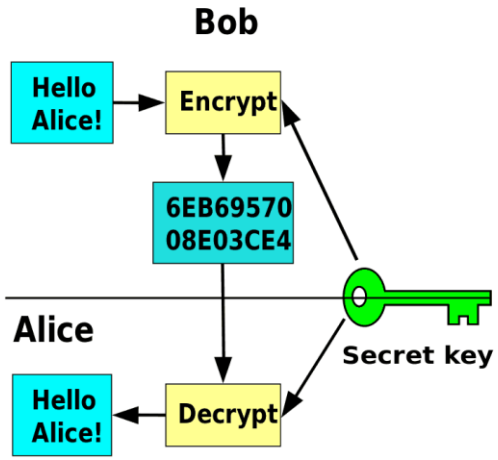
The layer which allows applications to interpret meaning of data

Translation

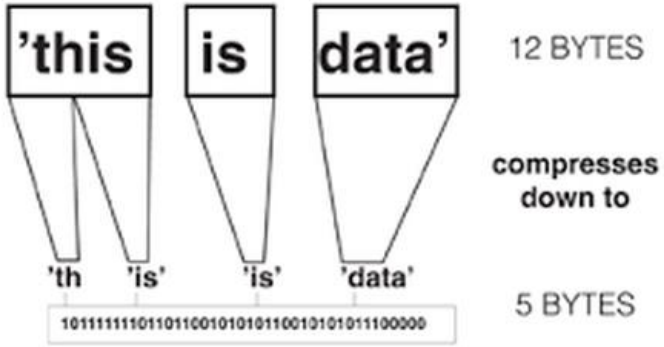
Text encoding → Encoding, Ascii

Bit/Byte order
File Syntax

Encryption



Compression

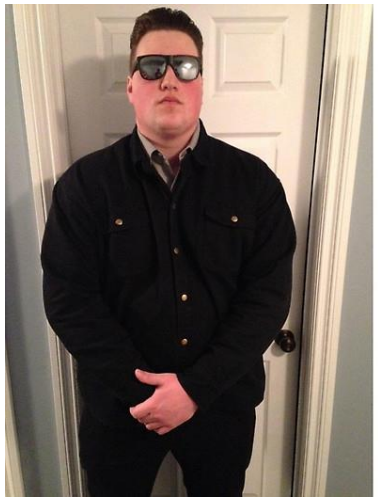


Session Layer

Manages, monitors, and synchronizing “sessions” between endpoints

Implements checkpoints while data is flowing

Authentication
Authorization



Application Layer

Presentation Layer

Session Layer

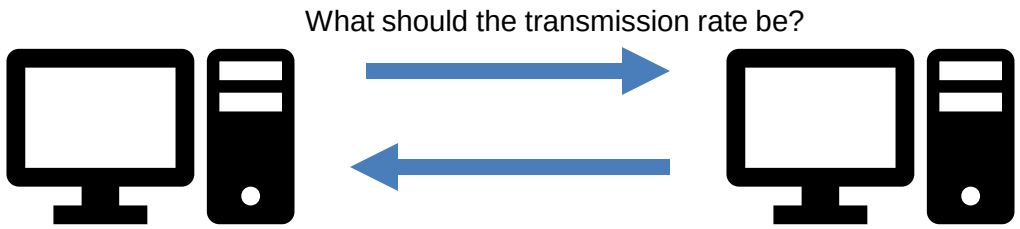
Most of this functionality is handled by our web browsers

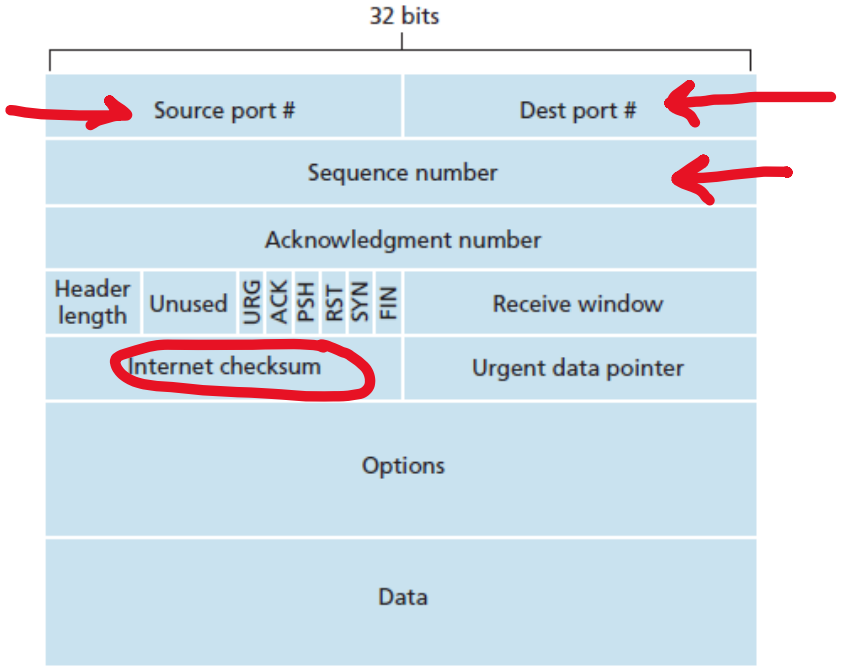
Transport Layer

Manges end to end communication and *method* of how data will be transferred

Ensures that the data received at host will be in the same order in which it was transmitted

Splits up packets into smaller **segments**





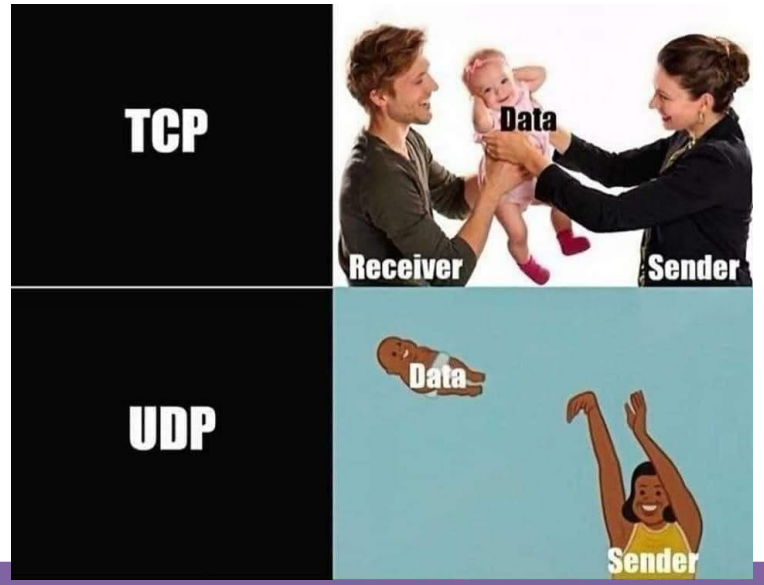
Transport Layer

Transmission Control Protocol (**TCP**)

- Requires an established connection to transmit data.
- Guarantees delivery of data in order
- Extensive error checking and acknowledgement of data

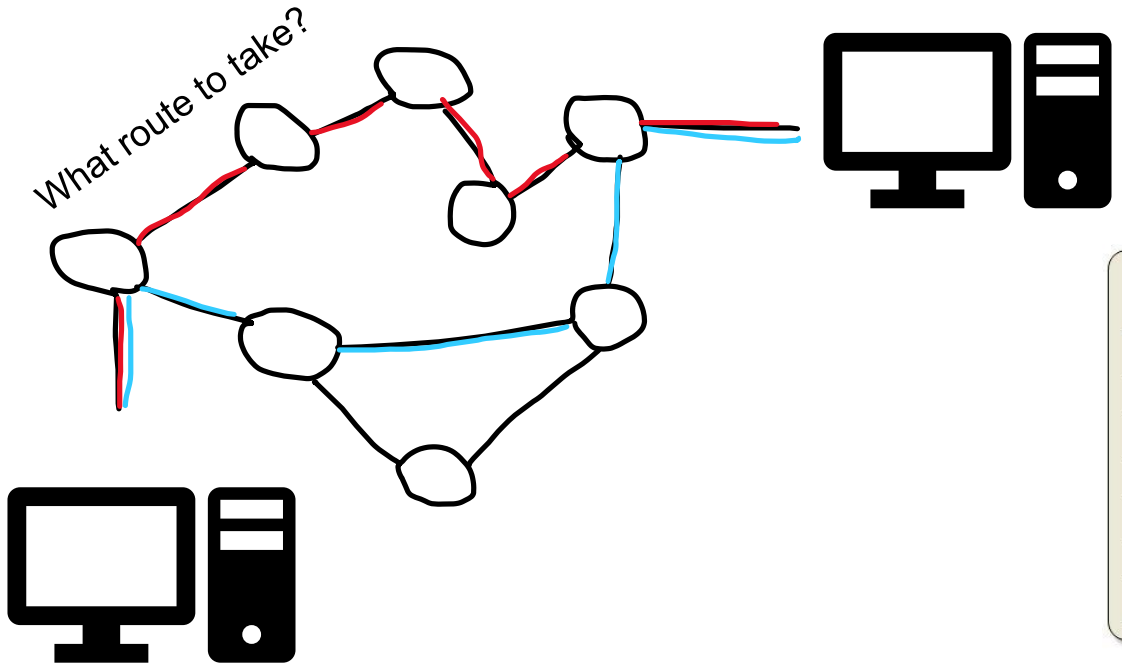
User Datagram Protocol (**UDP**)

- Connectionless protocol
- Faster, Simpler
- Not reliable
- No acknowledgement of data, basic error checking



Network Layer

Primary purpose is to move **datagrams** from one host to another, and to determine physical path to destination



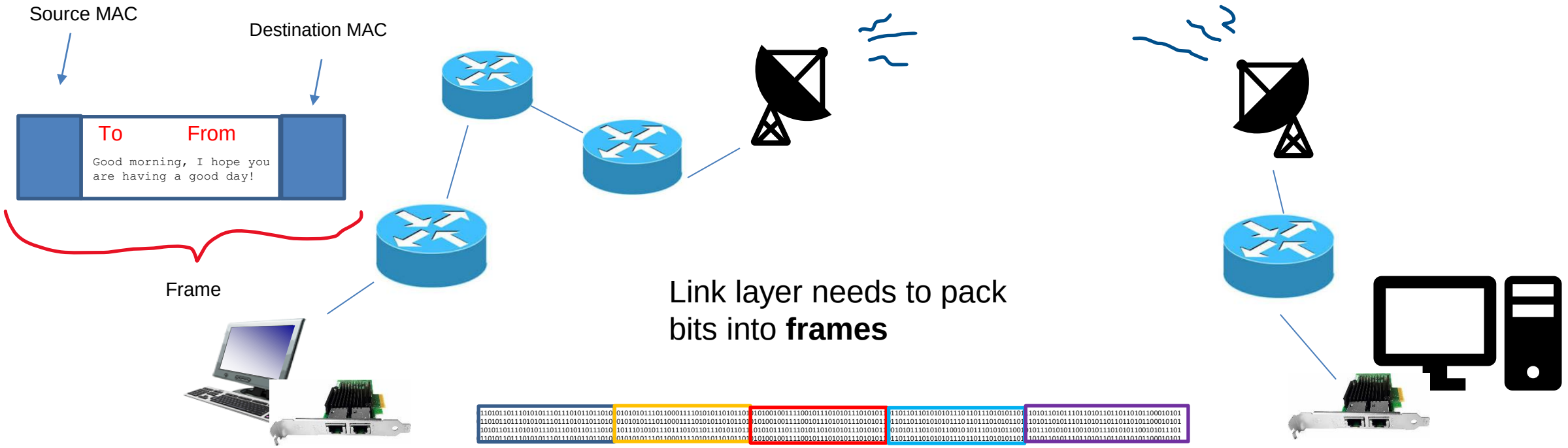
We also assign **IP addresses** to our packet

IPv4 Packet Header Format

Bit #	0	7	8	15	16	23	24	31
0	Version	IHL	DSCP	ECN	Total Length			
32	Identification				Flags	Fragment Offset		
64	Time to Live		Protocol		Header Checksum			
96	Source IP Address							
128	Destination IP Address							
160	Options (if IHL > 5)							

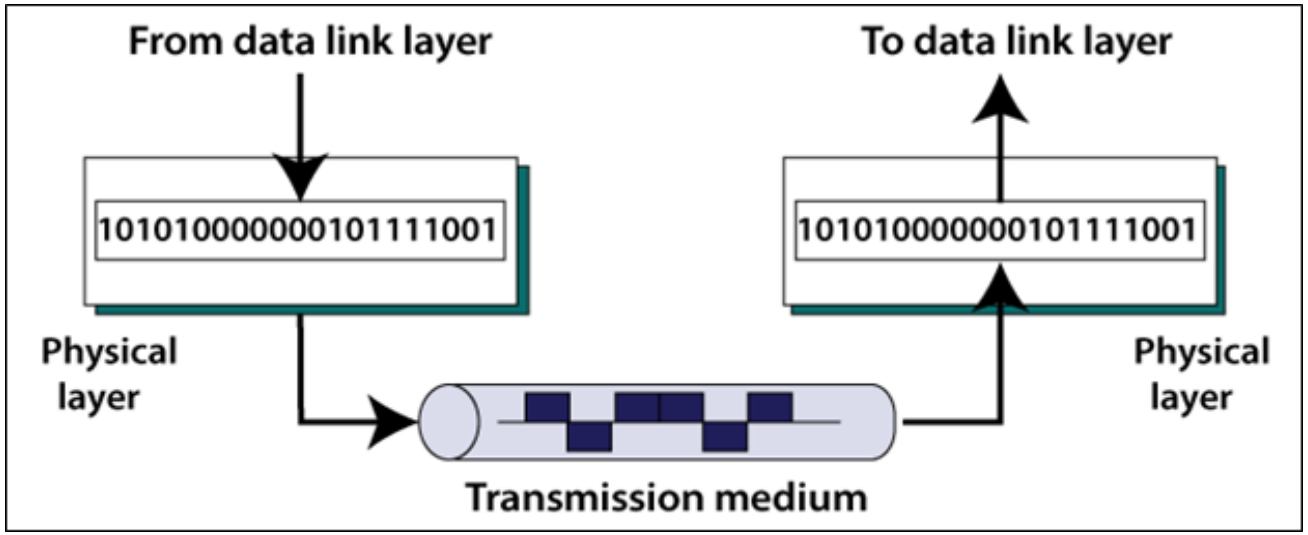
Data Link Layer

Handles the formatting and physical addressing of the “packet” before reaching the destination

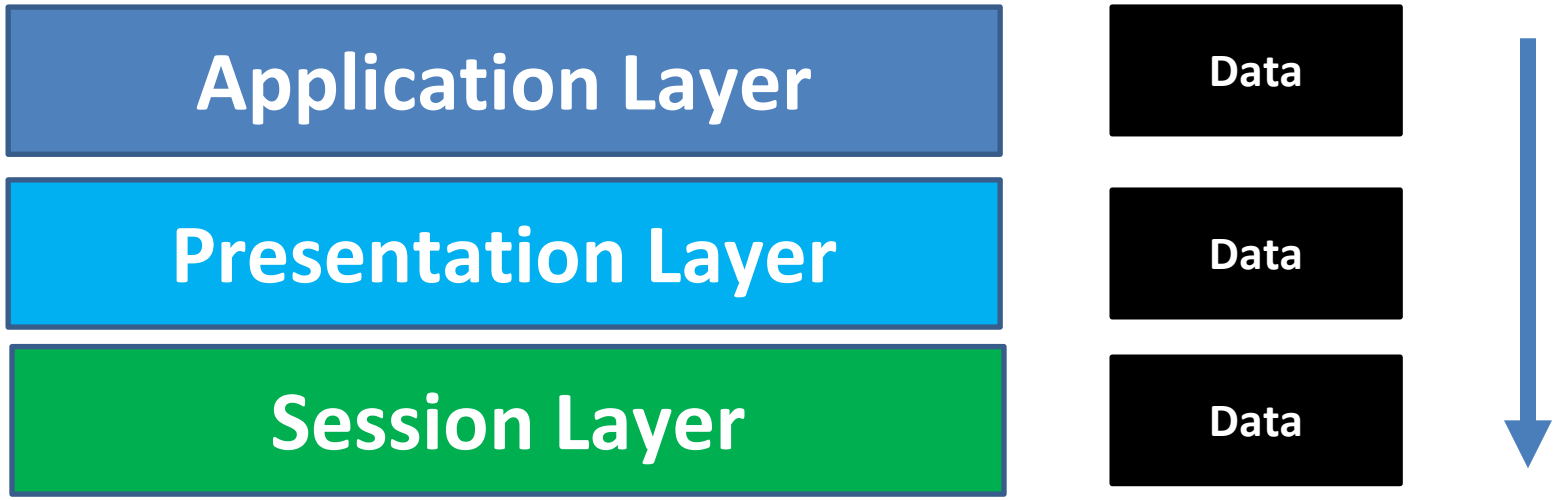


Physical Layer

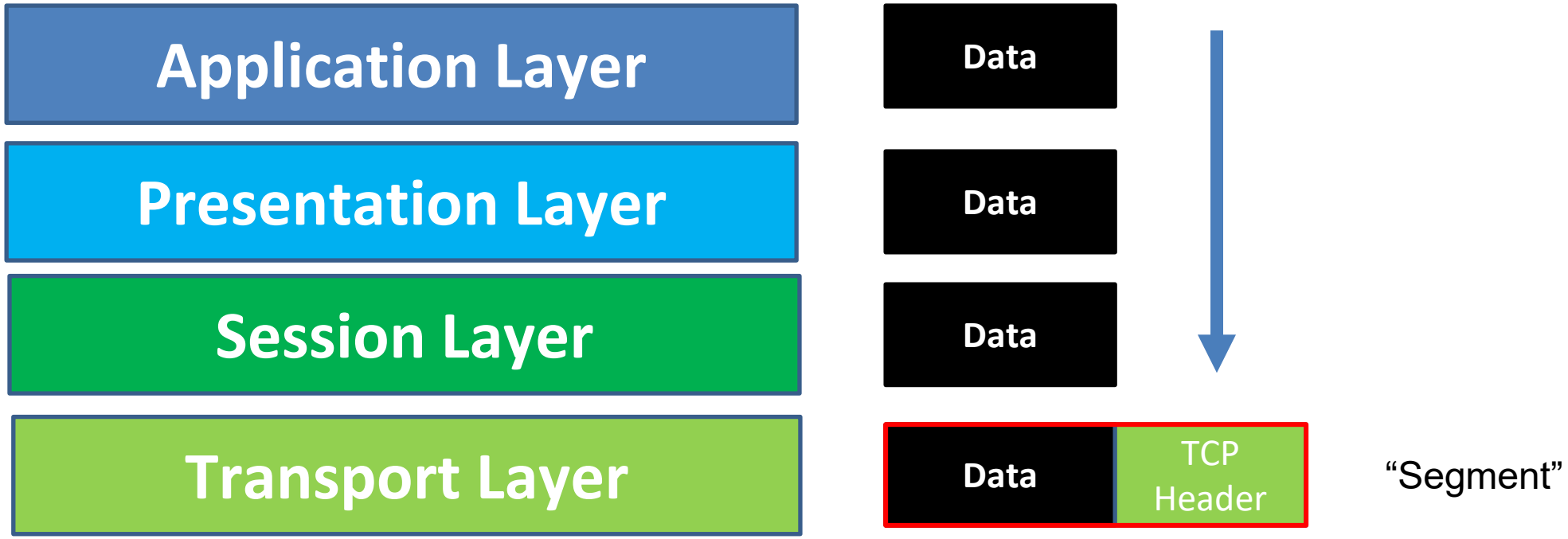
Transmits bits into physical signals over some medium



Encapsulation

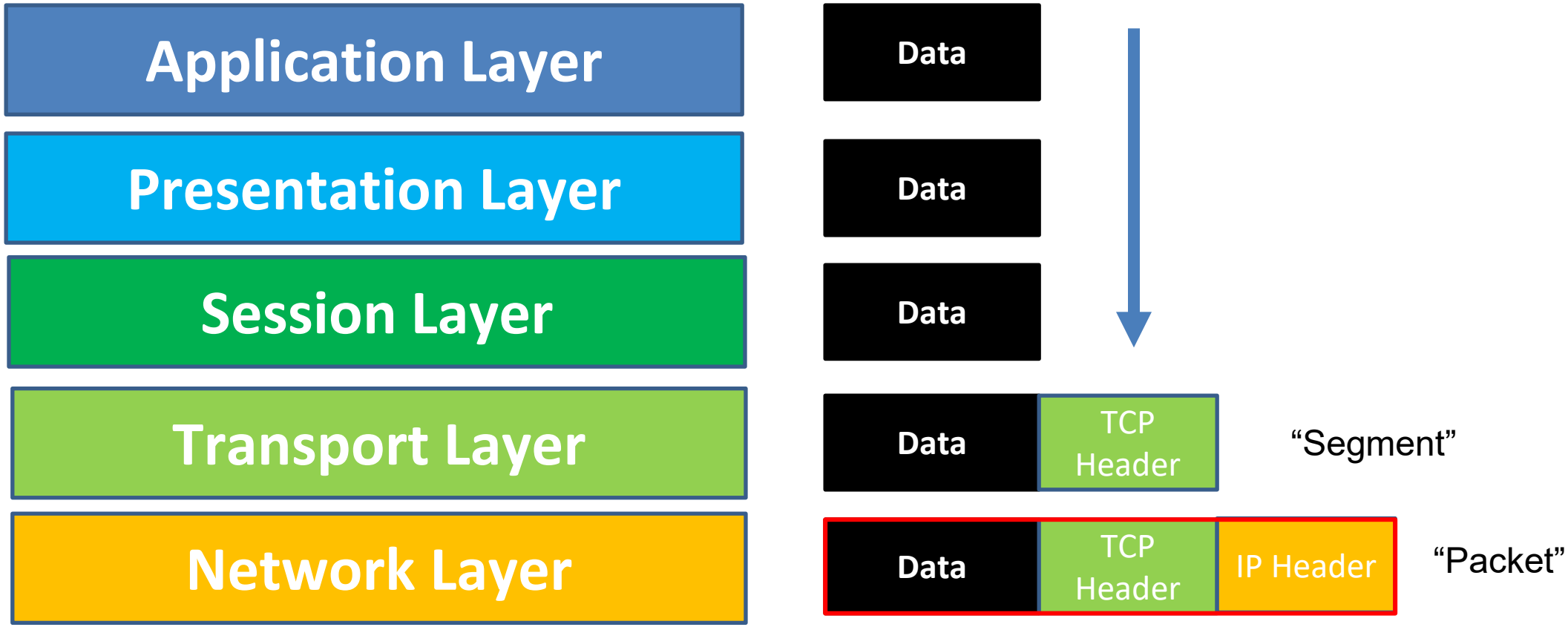


Encapsulation

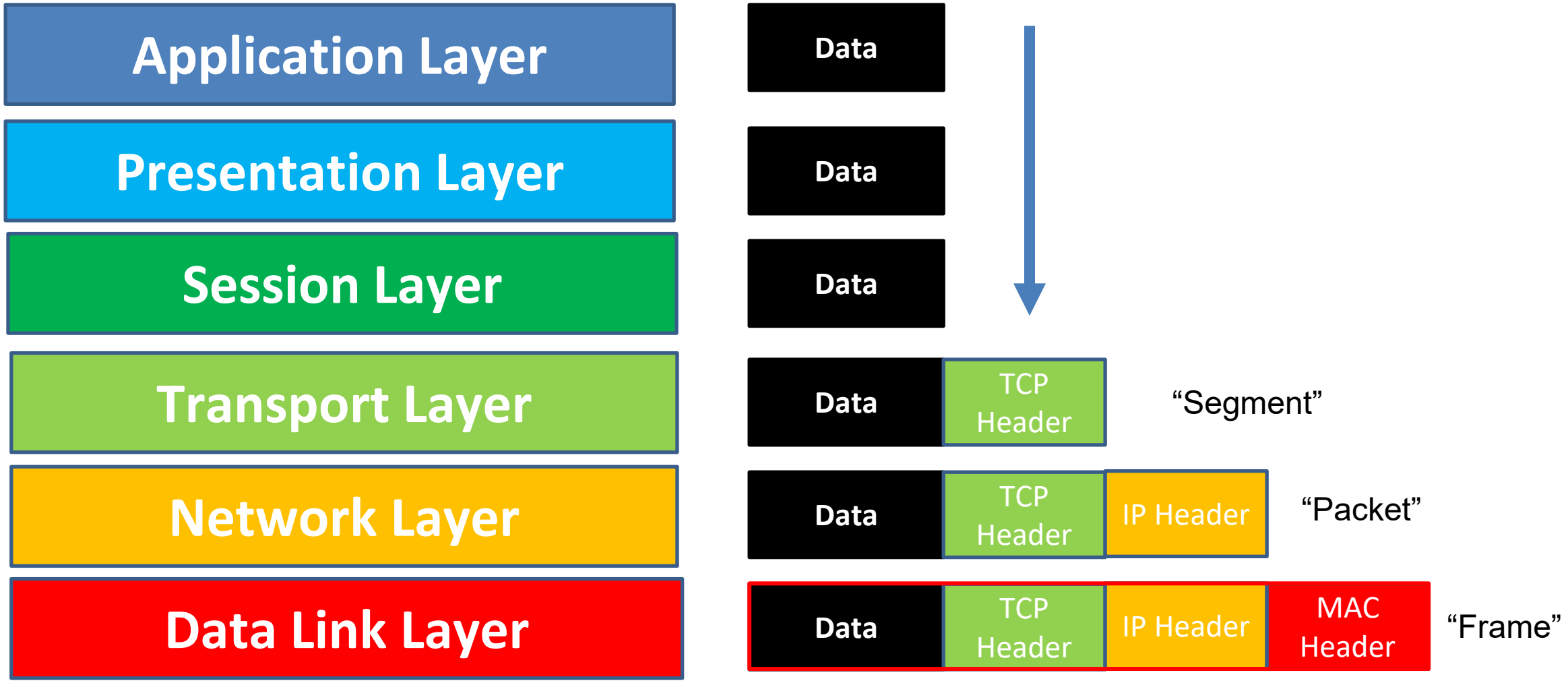


"Segment"

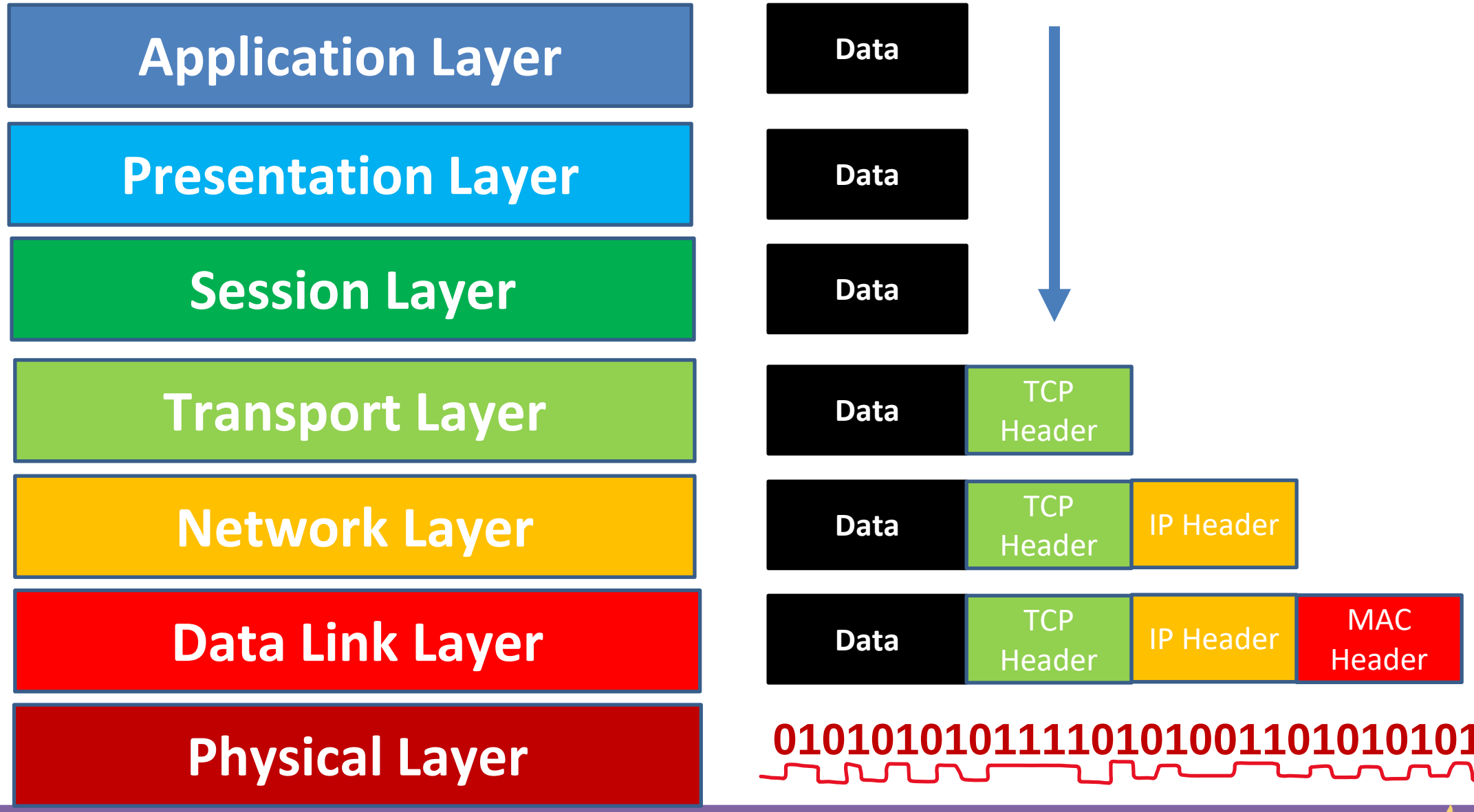
Encapsulation



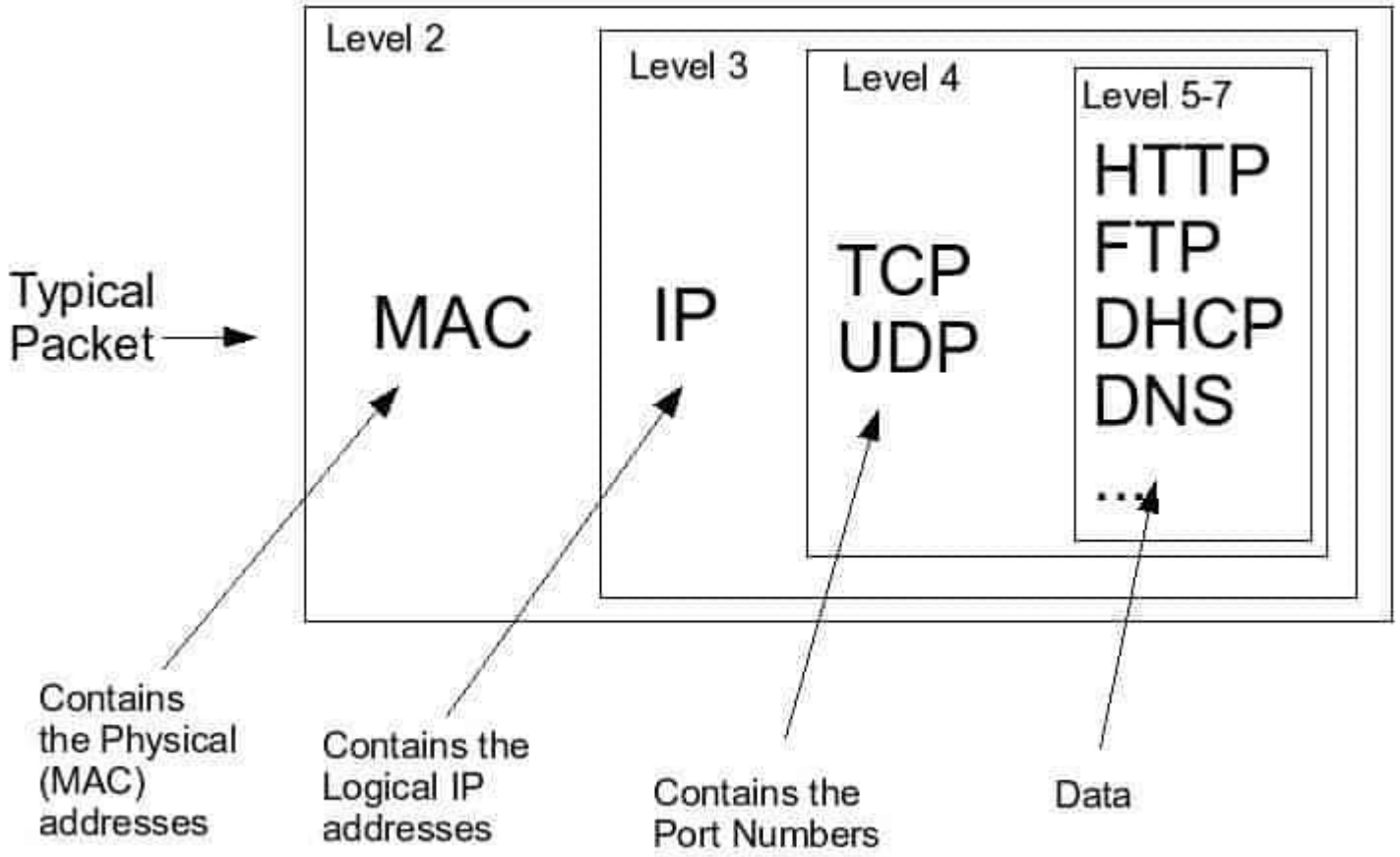
Encapsulation



Encapsulation



Encapsulation



Ticket (purchase)

Ticket (purchase)

Baggage (Check)

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

Airplane Routing

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

Airplane Routing

Airplane Routing

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

Airplane Routing

Runway landing

Airplane Routing

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

Airplane Routing

Gates (unload)

Runway landing

Airplane Routing

OSI Model

Ticket (purchase)

Baggage (Check)

Gates (load)

Runway Takeoff

Airplane Routing

Baggage (claim)

Gates (unload)

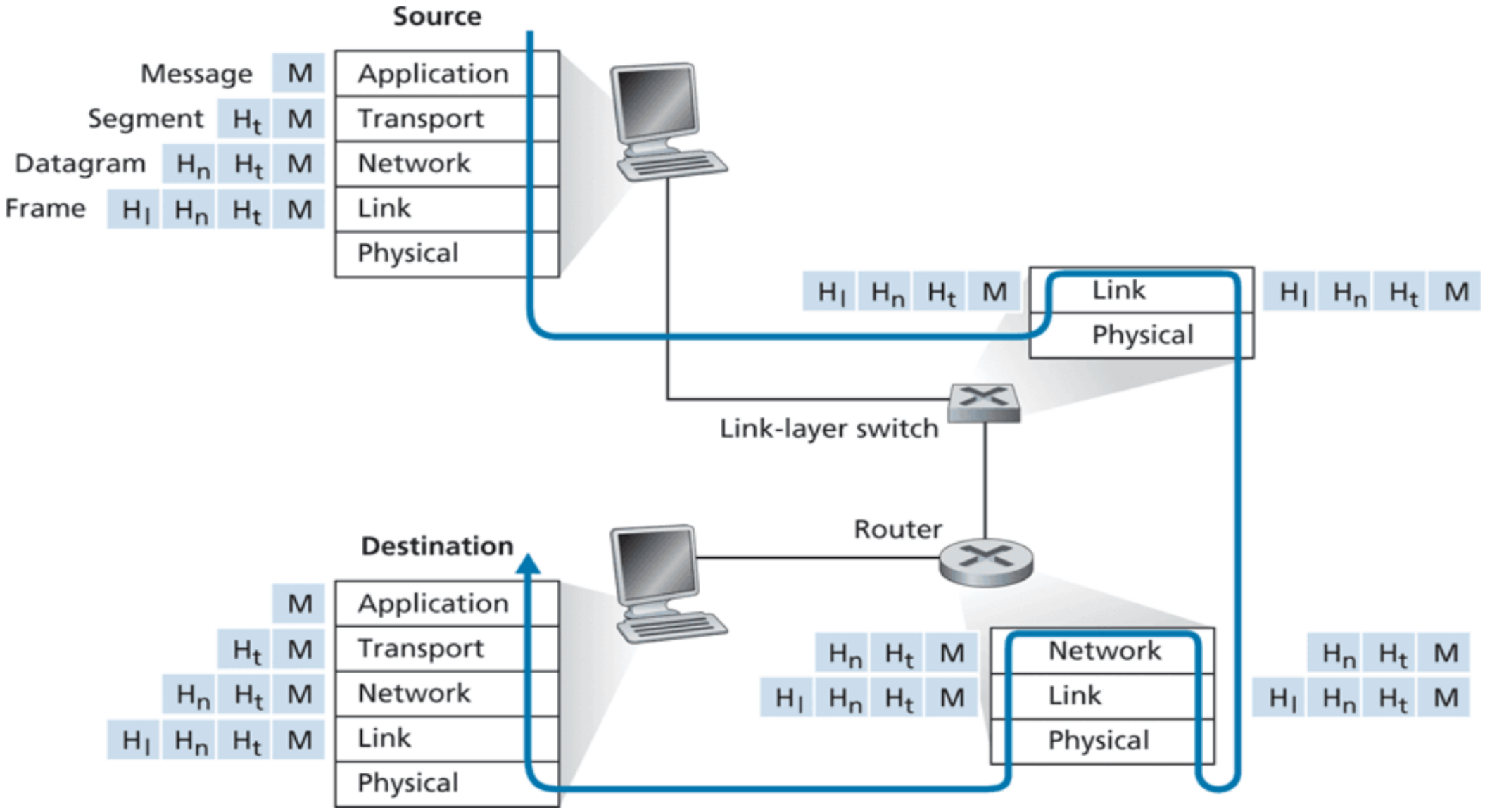
Runway landing

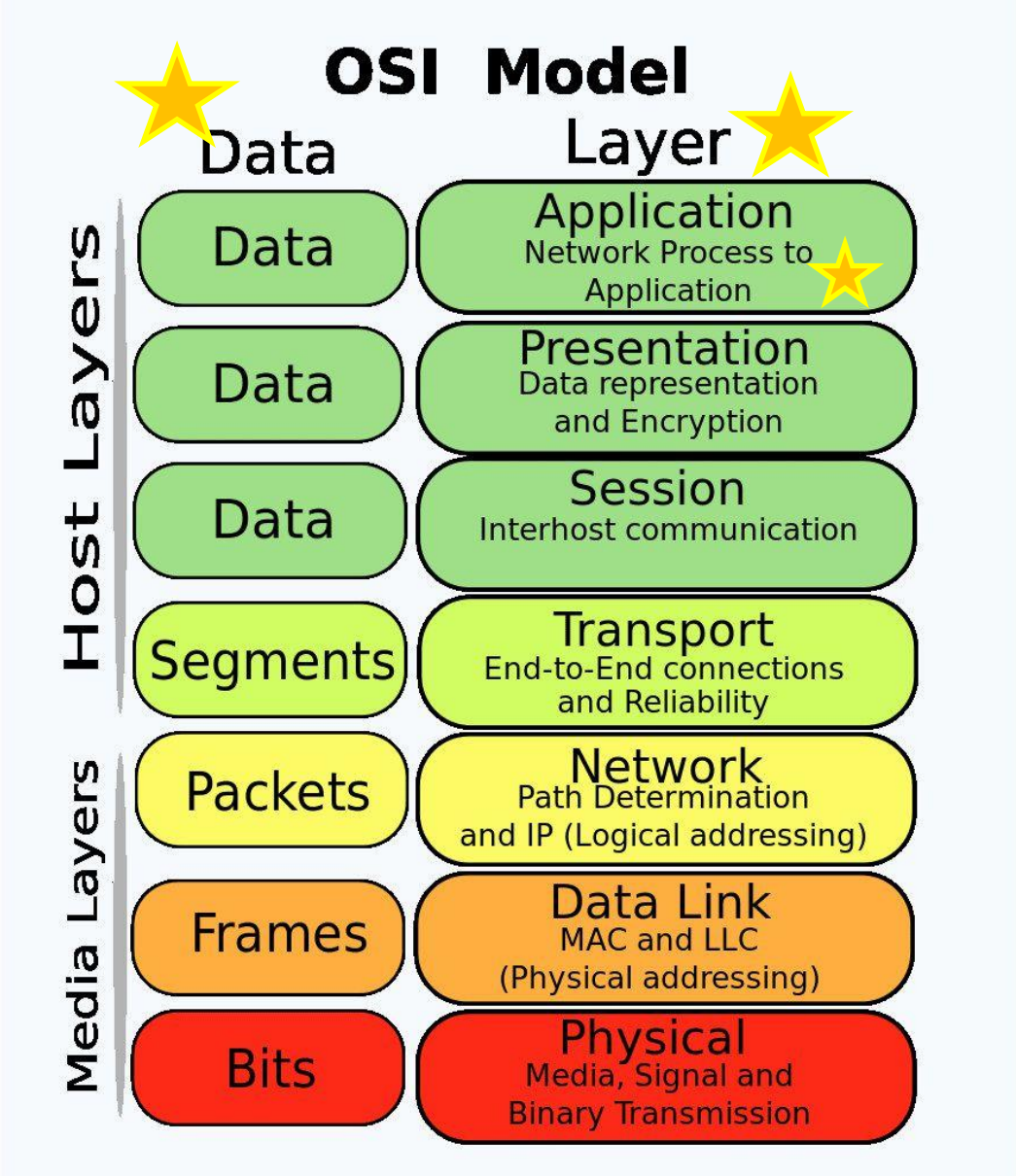
Airplane Routing

OSI Model

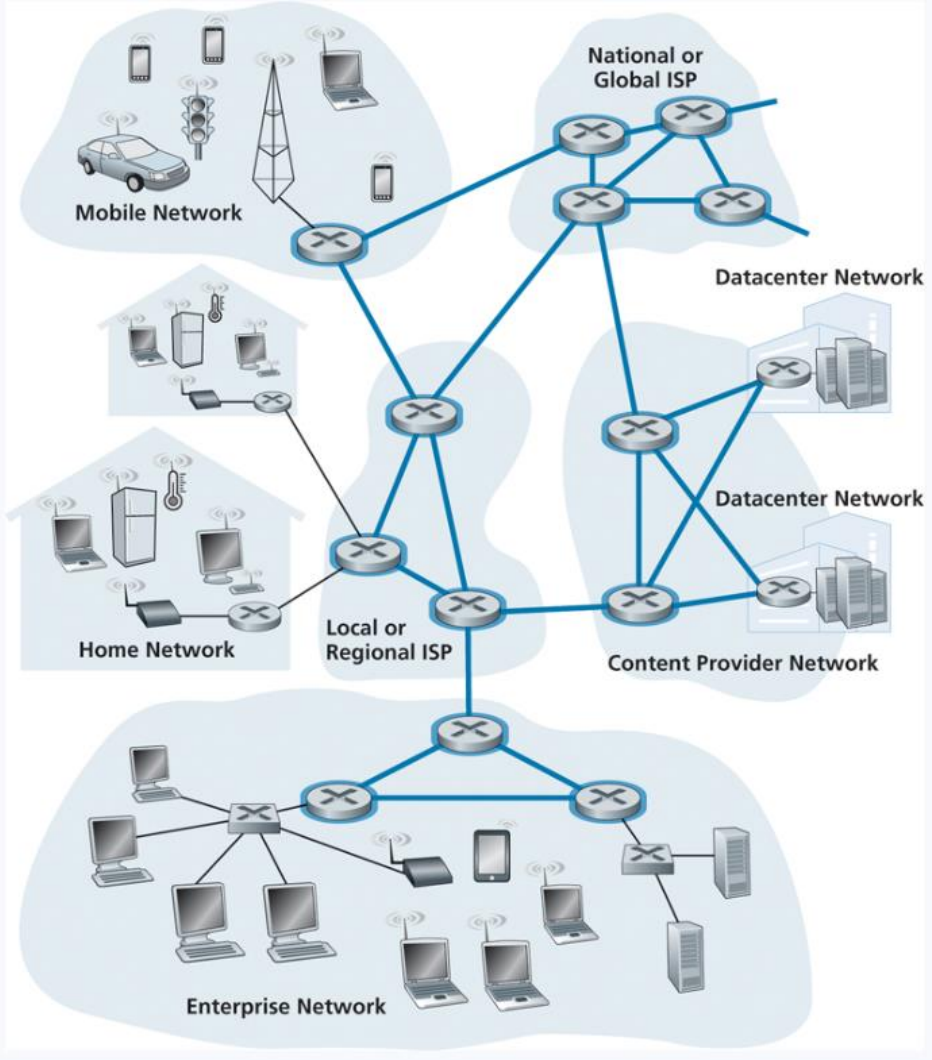
Ticket (purchase)	Tickets (complain)
Baggage (Check)	Baggage (claim)
Gates (load)	Gates (unload)
Runway Takeoff	Runway landing
Airplane Routing	Airplane Routing

OSI Model


Source
Message
Segment
Datagram
Frame
M
H_t
H_n
H_l
Application
Transport
Network
Link
Physical
Link-layer switch
Router
Destination
Application
Transport
Network
Link
Physical

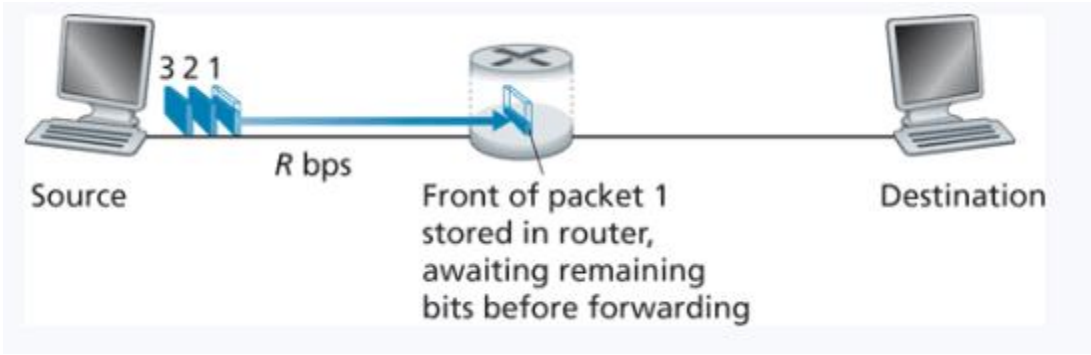

OSI Model
Data
Layer
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data representation and Encryption
Data
Session
Interhost communication
Segments
Transport
End-to-End connections and Reliability
Packets
Network
Path Determination and IP (Logical addressing)
Frames
Data Link
MAC and LLC (Physical addressing)
Bits
Physical
Media, Signal and Binary Transmission

Data Forwarding



Packet Switching

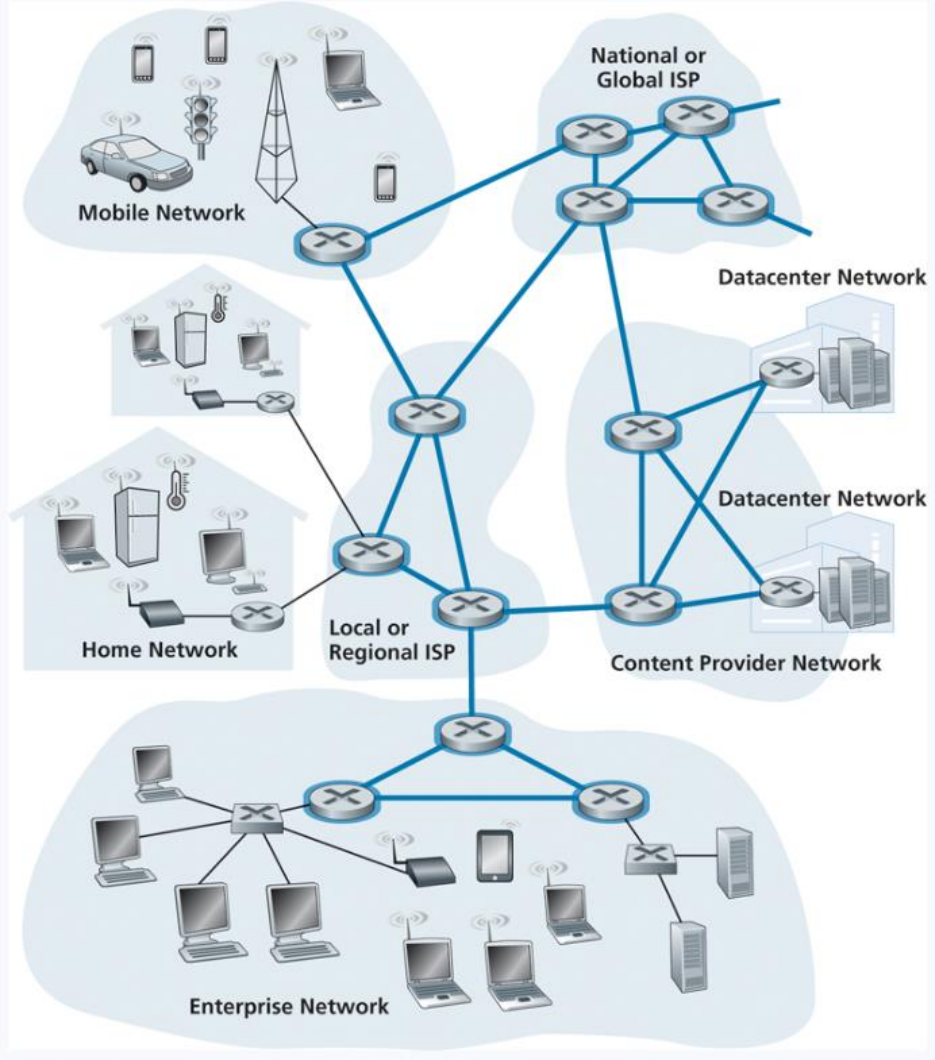
Uses **store-and-forward** transmission



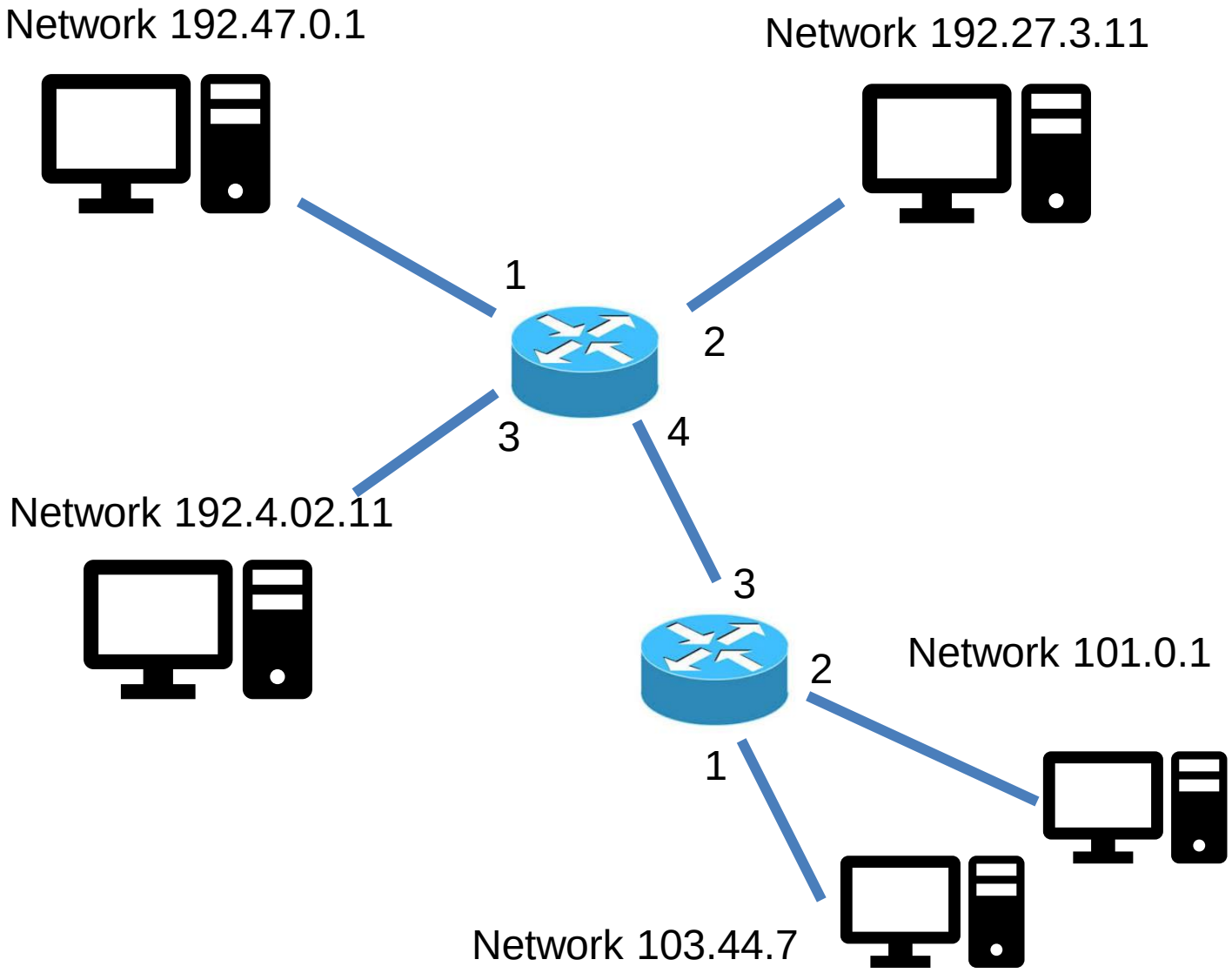
Time to transmit *L* bits over a link with transmission rate of *R*:

$$L / R$$

Data Forwarding



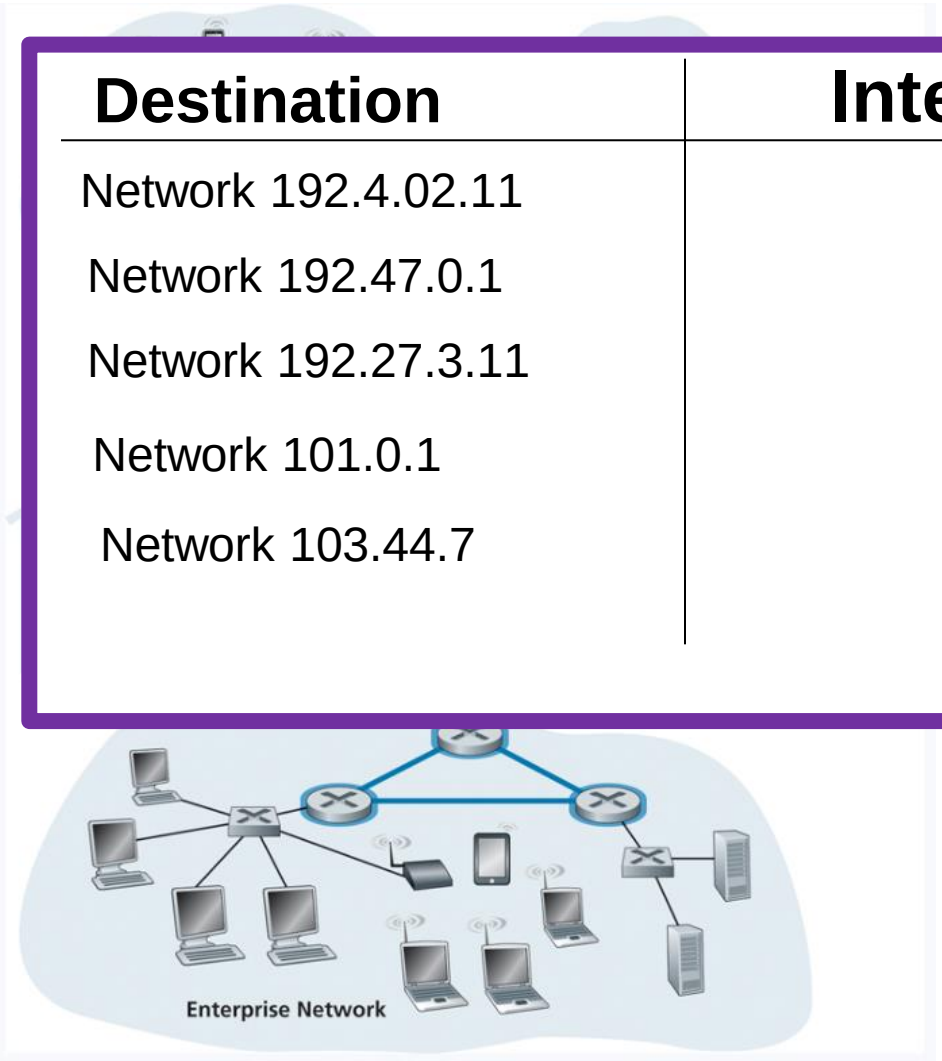
Routing Table

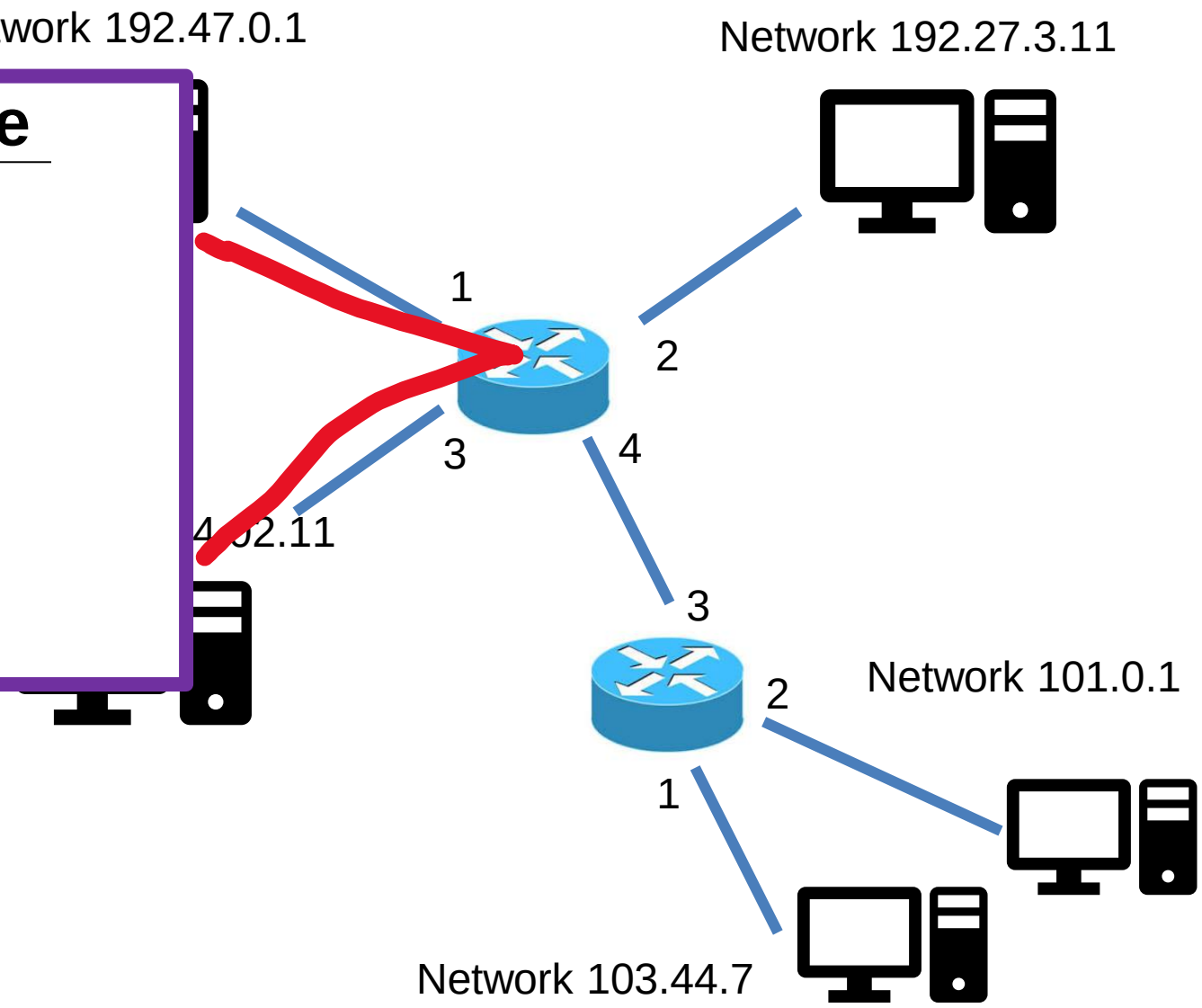


Data Forwarding

Routing Table

Destination	Interface
Network 192.4.02.11	3
Network 192.47.0.1	1
Network 192.27.3.11	2
Network 101.0.1	4
Network 103.44.7	4



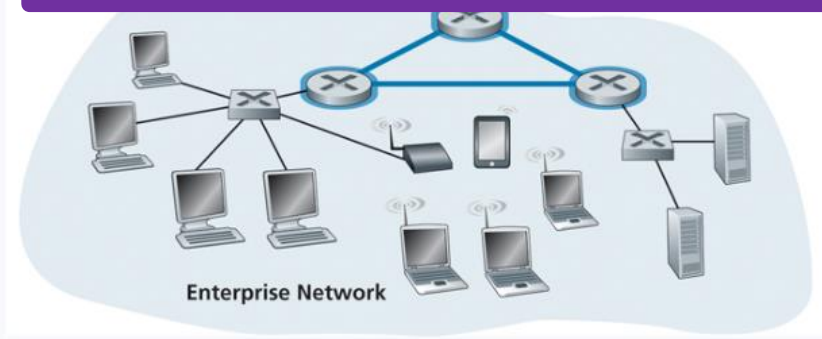


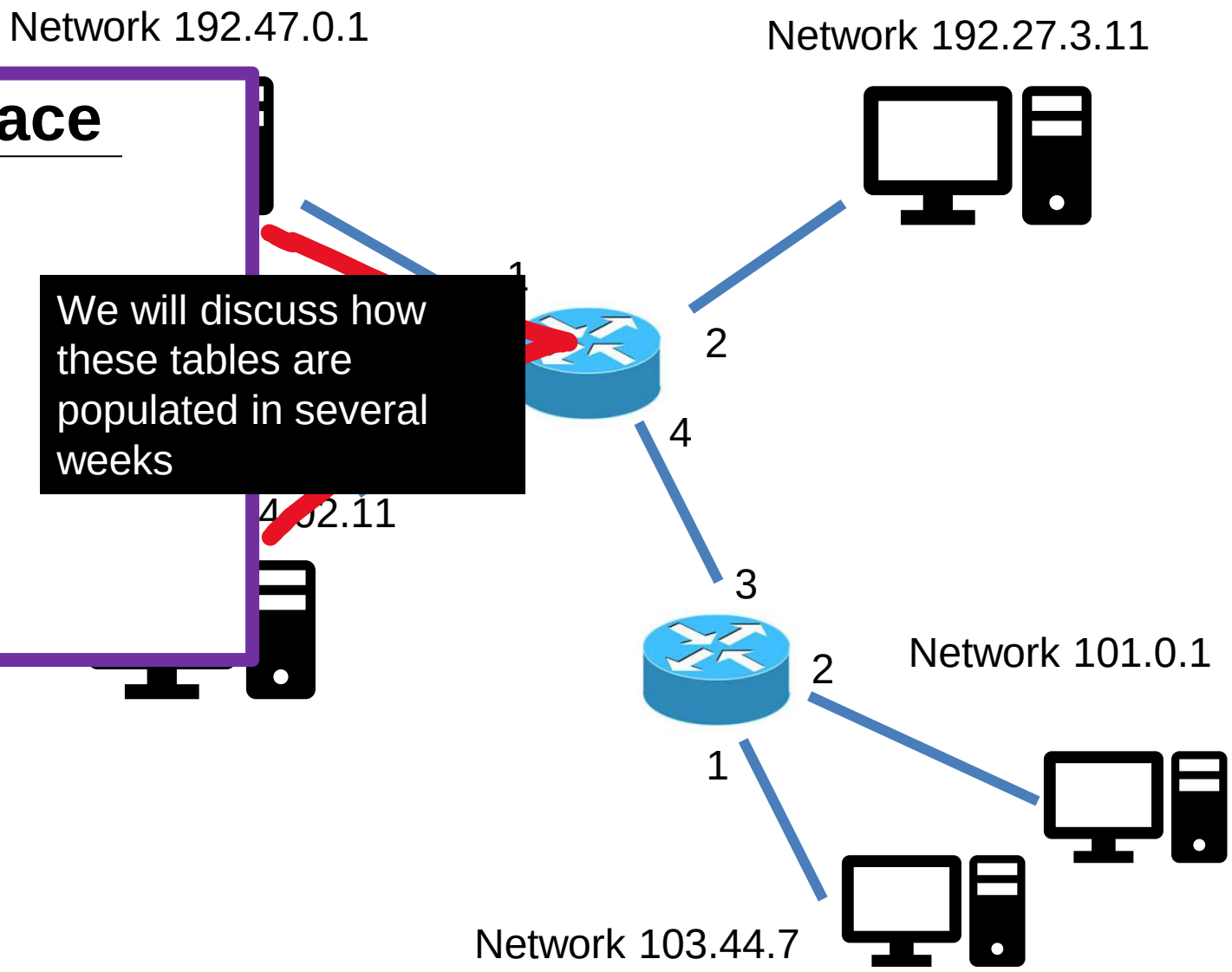
Data Forwarding

Routing Table

Destination	Interface
Network 192.4.02.11	3
Network 192.47.0.1	1
Network 192.27.3.11	2
Network 101.0.1	4
Network 103.44.7	4

We will discuss how these tables are populated in several weeks





Announcements

First homework will be released either tomorrow or Friday. Due at the end of next week.

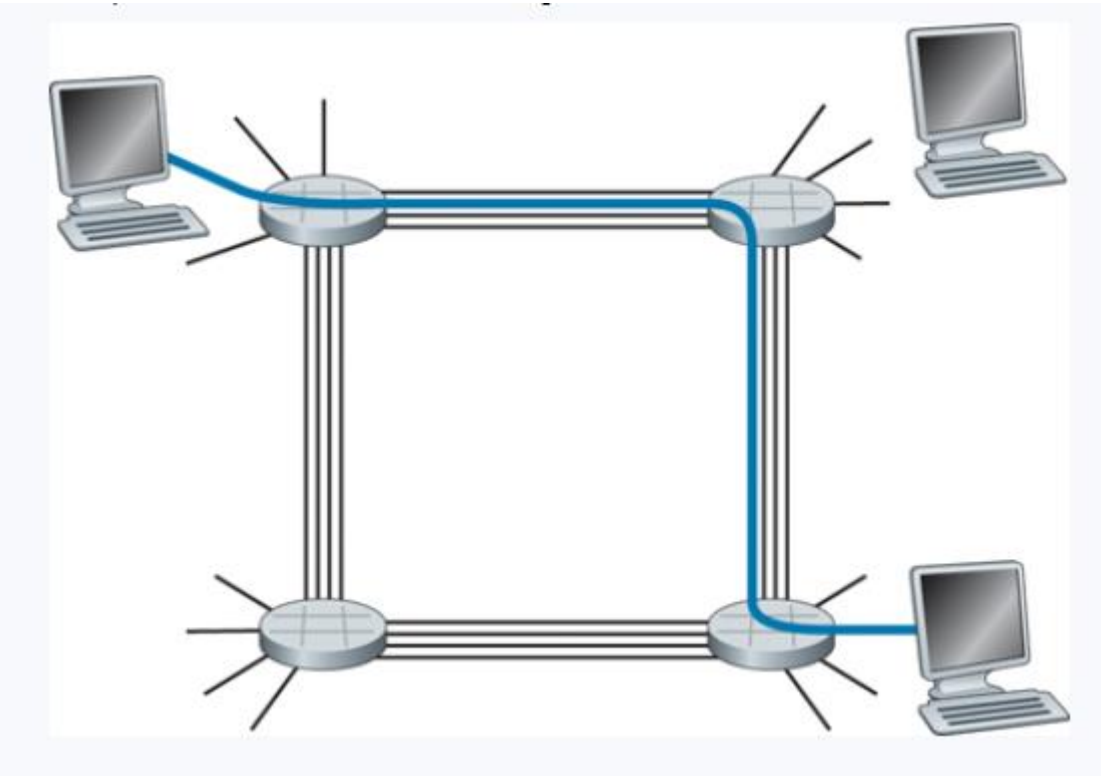
I am currently working out the details of Programming Assignment 1

No class on Monday

Data Forwarding

In circuit switching, the path and resources for transmitting from A to B is **reserved**

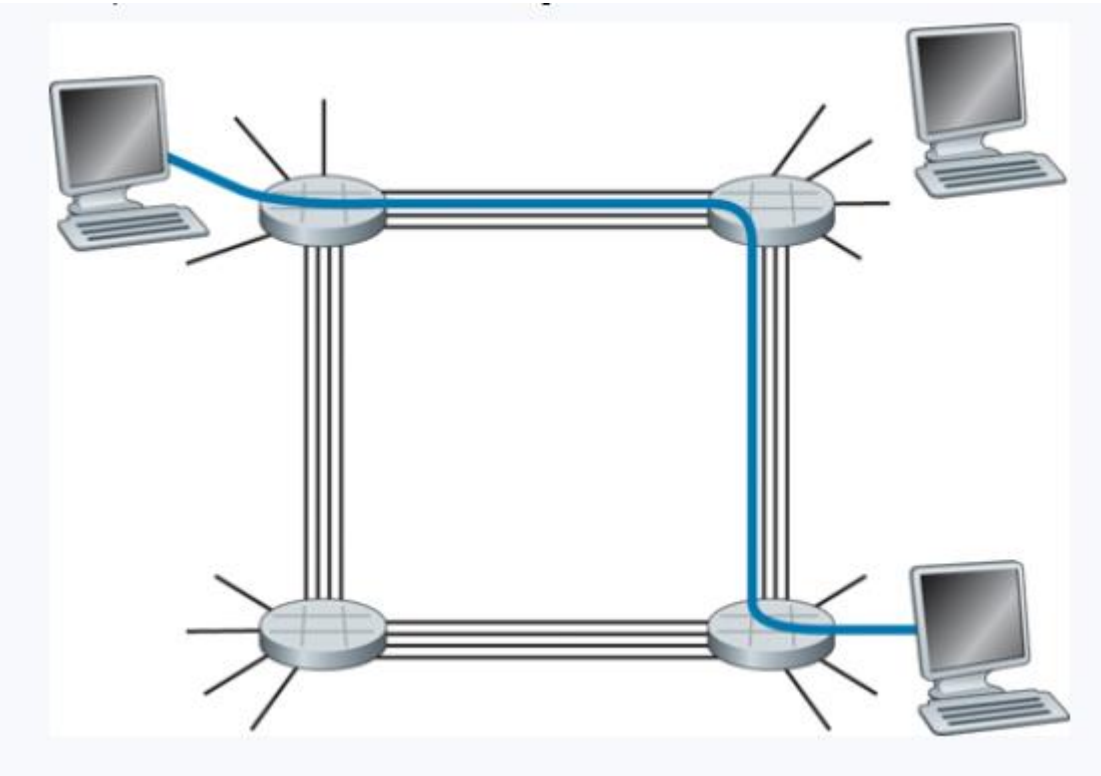
Communication links are divided into **circuits**, which allow for concurrent usage of the link



Data Forwarding

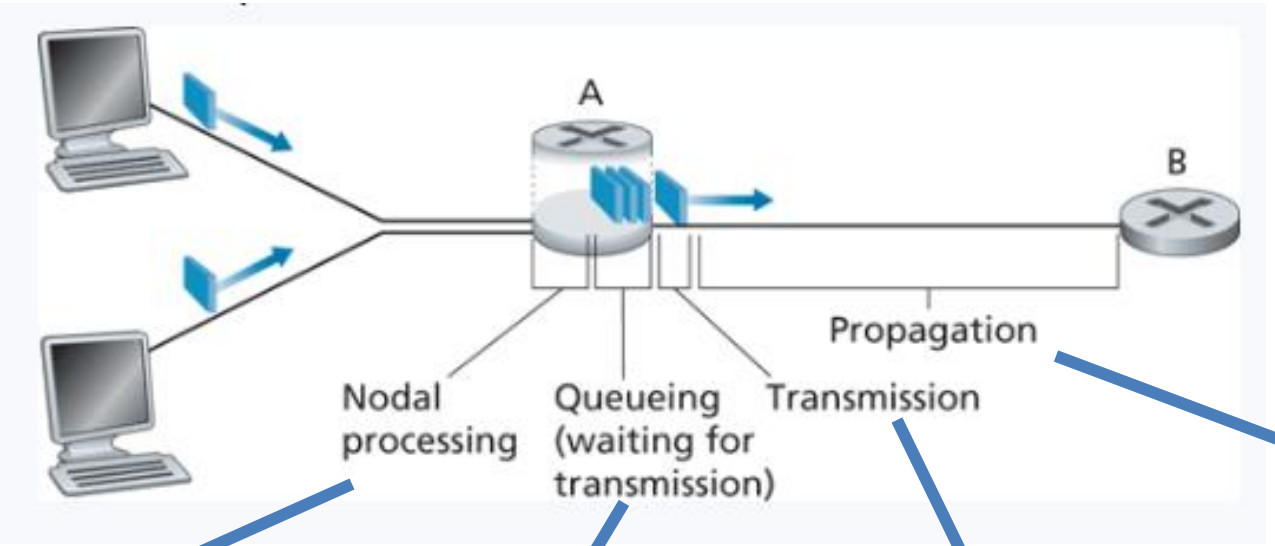
In circuit switching, the path and resources for transmitting from A to B is **reserved**

Communication links are divided into **circuits**, which allow for concurrent usage of the link



Reserved spaces that are not in use result in **silent periods**

Network Delay



Propagation Delay- time needed to propagate to destination
(dependent on the medium)

Transmission delay- Amount of time required to push all the packet's bits into the link

Queueing Delay- Time the packet sits in the queue

Processing delay- time required to examine the packet's header and determine where to direct the packet

Nodal Delay = Processing delay + Queueing delay + Transmission delay + Propagation delay

Network Delay

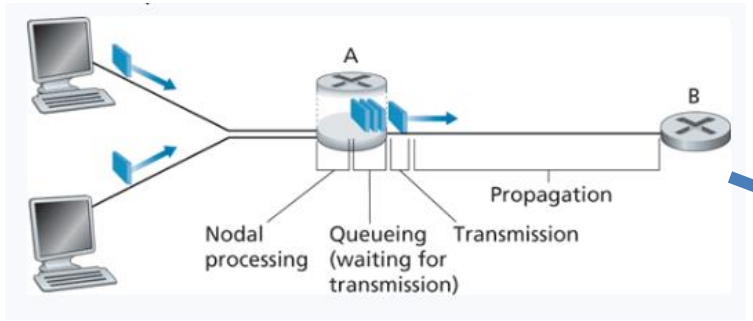
Processing delay- time required to examine the packet's header and determine where to direct the packet

Queueing Delay- Time the packet sits in the queue

pagation Delay- time
eded to propagate to
tination
ependent on the medium)

Nodal Delay = Processing delay + Queueing delay + Transmission delay + Propagation delay

Network Delay



Propagation Delay- time needed to propagate to destination

(dependent on the medium)

Queueing Delay- Time the packet sits in the queue

Transmission delay- Amount of time required to push all the packet's bits into the link

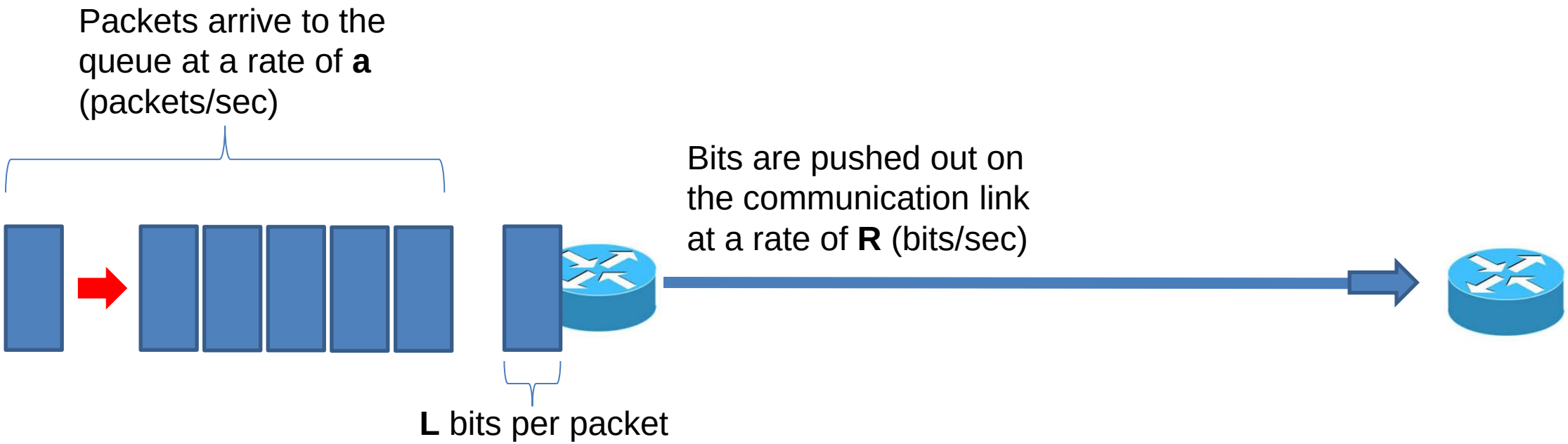
Processing delay- time required to examine the packet's header and determine where to direct the packet

Nodal Delay = Processing delay + Queueing delay + Transmission delay + Propagation delay

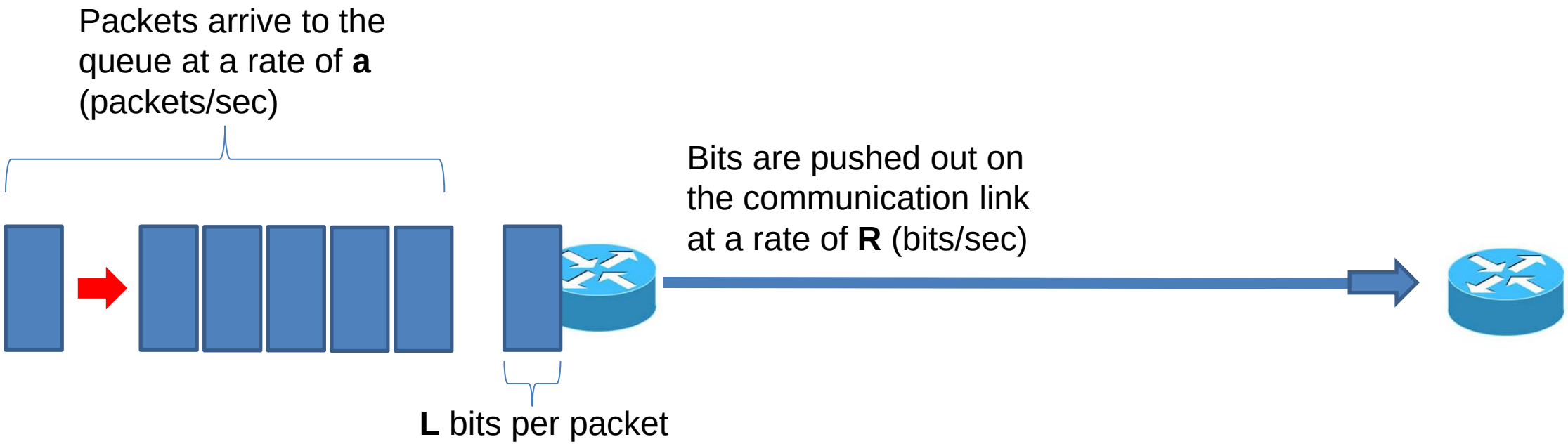
In an <u>uncongested</u> network with **N** links between source and destination

End-to-end Delay= N(Processing delay + Transmission delay + Propagation delay)

Queueing Delay

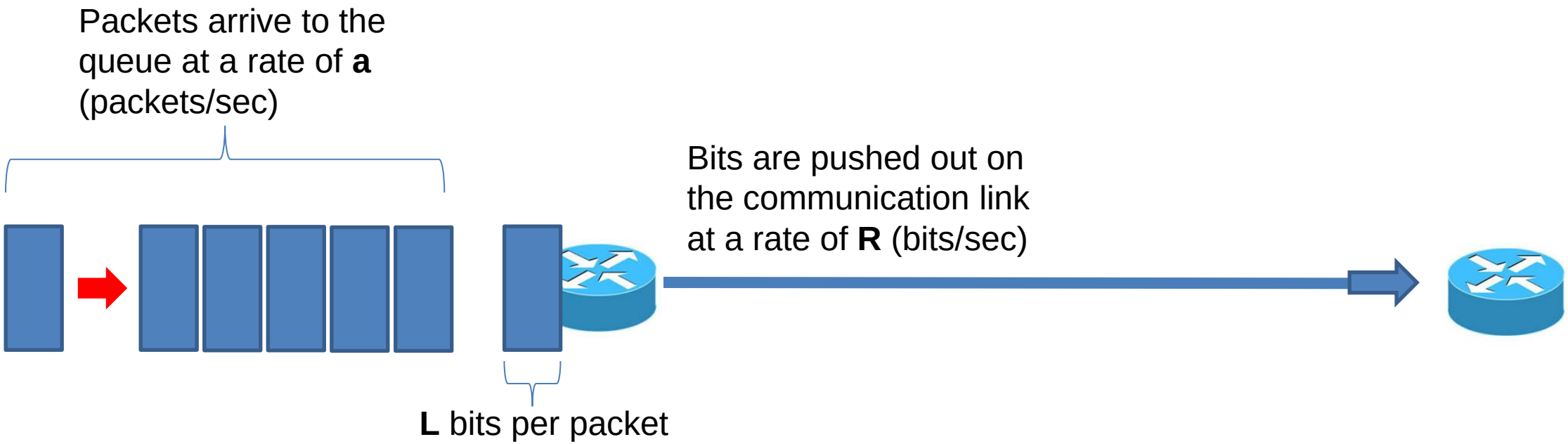


Queueing Delay



$$\textbf{Traffic Intensity} = \frac{L * a}{R}$$

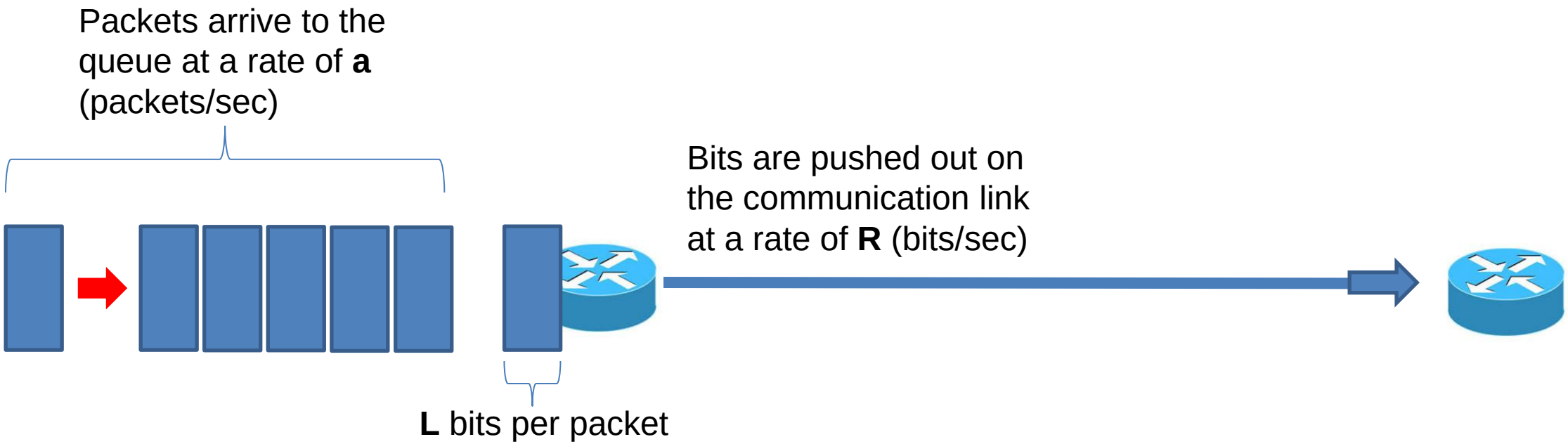
Queueing Delay



Traffic Intensity $= \frac{L * a}{R}$

Ratio of **average bits that arrive at queue** to **how quick we can process one bit**

Queueing Delay

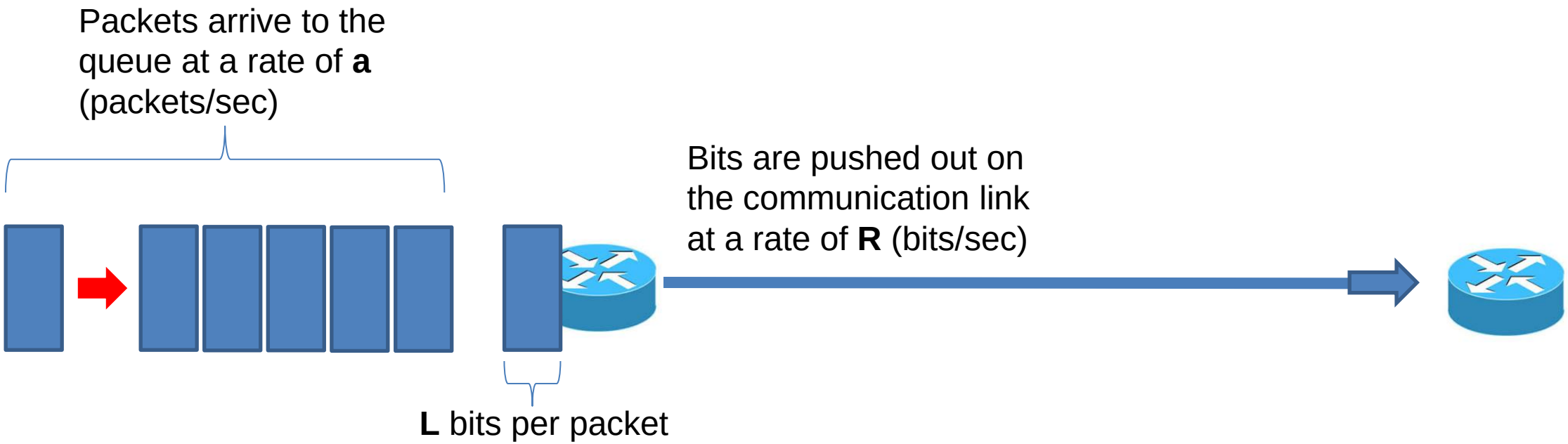


Traffic Intensity $= \frac{L * a}{R}$

Ratio of **average bits that arrive at queue** to **how quick we can process one bit**

If traffic intensity > 1 ?

Queueing Delay



Traffic Intensity $= \frac{L * a}{R}$

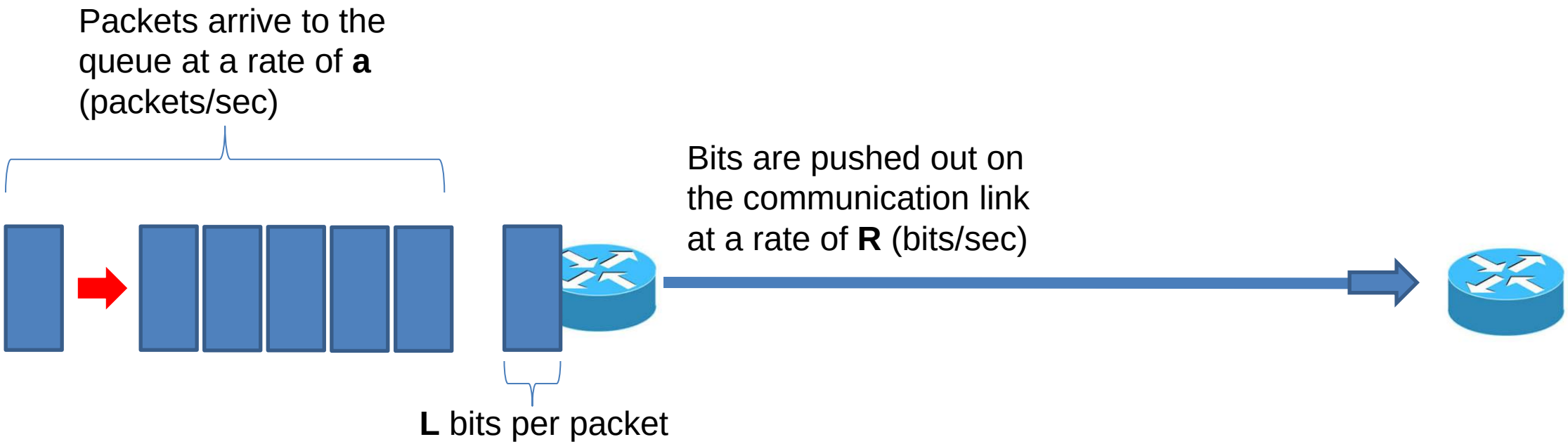
Ratio of **average bits that arrive at queue** to **how quick we can process one bit**

If traffic intensity > 1 ?

Bits arrive to the queue faster than we can process them

Bad!

Queueing Delay

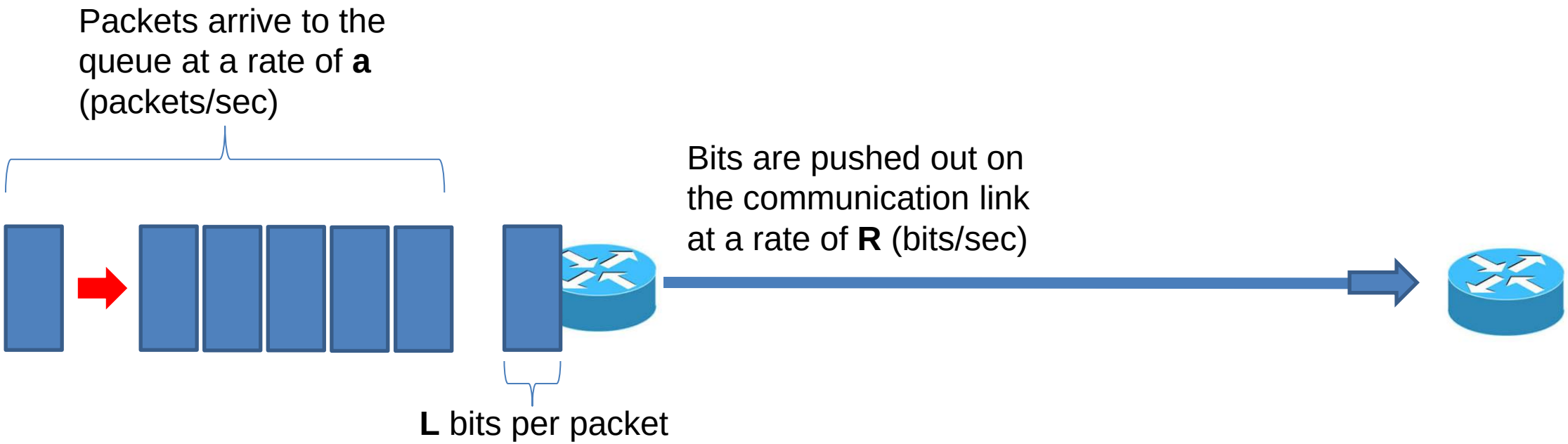


Traffic Intensity $= \frac{L * a}{R}$

Ratio of **average bits that arrive at queue** to **how quick we can process one bit**

If traffic intensity ≤ 1 ?

Queueing Delay



Traffic Intensity $= \frac{L * a}{R}$

Ratio of **average bits that arrive at queue** to **how quick we can process one bit**

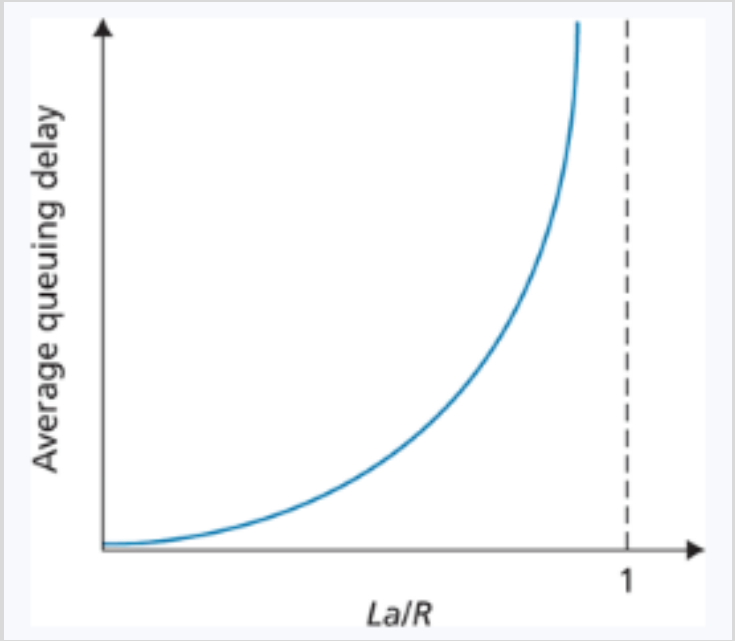
If traffic intensity ≤ 1 ?

We can process bits faster than they arrive**

Good!

Queueing Delay

Pa
qu
(pa



As **traffic intensity** grows, the **average queueing delay** will increase rapidly

The **arrival rate** of packets is typically random, so Queueing Delay is difficult to predict.

Traffi

If traffic intensity ≤ 1 ?

We can process bits faster than they arrive**

Good!

Traceroute

Network diagnostic tool that displays route taken to destination and RTT for each hop

```
C:\Users\Reese Pearsall>tracert google.com

Tracing route to google.com [172.217.14.238]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  gateway119.254.msu.montana.edu [153.90.119.254]
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4    <1 ms    <1 ms    <1 ms  153.90.125.254
  5    <1 ms    <1 ms    <1 ms  10.196.6.10
  6     1 ms     1 ms    <1 ms  rnedge-prodfw.msu.montana.edu [192.105.205.131]
  7    15 ms    15 ms    16 ms  ae13--538.icar-sttl1-2.infra.pnw-gigapop.net [209.124.190.212]
  8    15 ms    15 ms    15 ms  209.124.190.202
  9    17 ms    17 ms    17 ms  142.251.70.99
 10    16 ms    16 ms    16 ms  209.85.254.247
 11    15 ms    15 ms    15 ms  sea30s02-in-f14.1e100.net [172.217.14.238]

Trace complete.
```

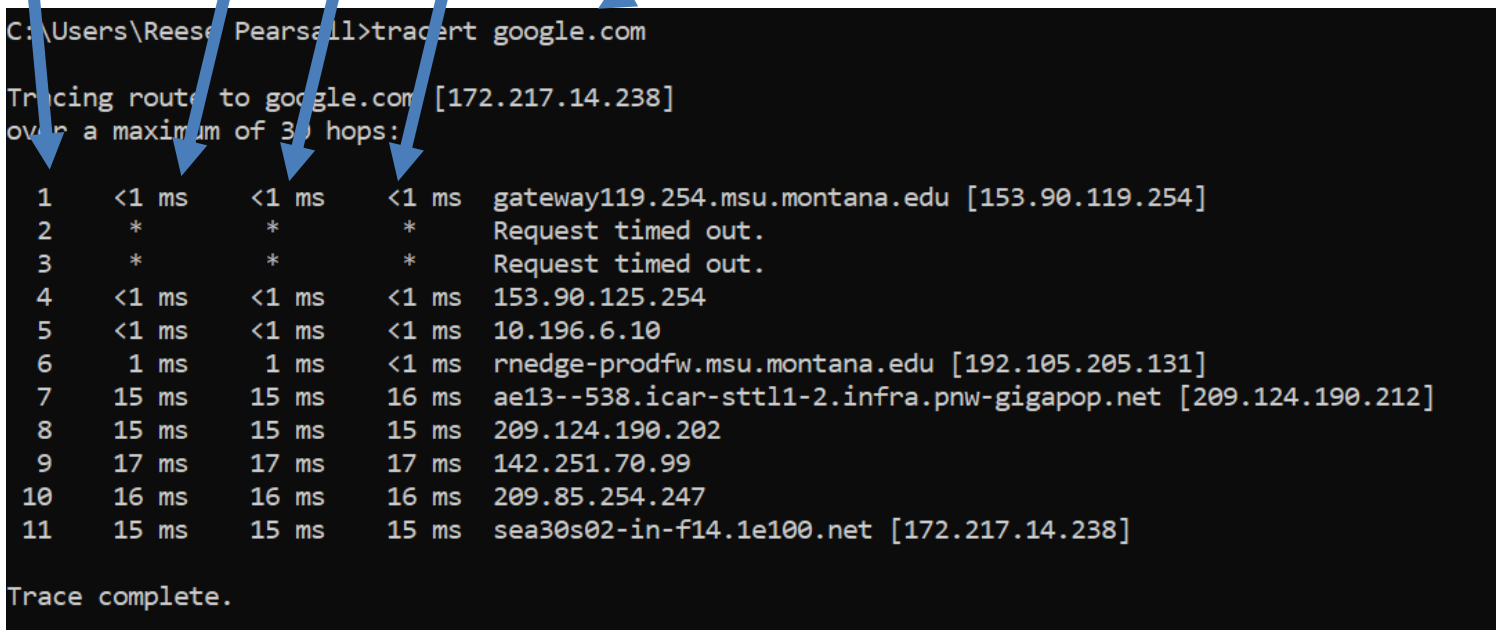
Traceroute

Network diagnostic tool that displays route taken to destination and RTT for each hop

Hop #

RTT time for each packet

Destination



Traceroute

Network diagnostic tool that displays route taken to destination and RTT for each hop

Hop #

RTT time for each packet

Destination

```
C:\Users\Reese Pearsall>tracert google.com

Tracing route to google.com [172.217.14.238]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  gateway119.254.msu.montana.edu [153.90.119.254]
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4    <1 ms    <1 ms    <1 ms  153.90.125.254
  5    <1 ms    <1 ms    <1 ms  10.196.6.10
  6     1 ms     1 ms    <1 ms  rnedge-prodfw.msu.montana.edu [192.105.205.131]
  7    15 ms    15 ms    16 ms  ae13--538.icar-sttl1-2.infra.pnw-gigapop.net [209.124.190.212]
  8    15 ms    15 ms    15 ms  209.124.190.202
  9    17 ms    17 ms    17 ms  142.251.70.99
 10    16 ms    16 ms    16 ms  209.85.254.247
 11    15 ms    15 ms    15 ms  sea30s02-in-f14.1e100.net [172.217.14.238]

Trace complete.
```

whois- provides registration data of a domain or IP address

172.217.14.238 address profil

Whois | Diagnostics

IP Whois

```
NetRange:       172.217.0.0 - 172.217.255.255
CIDR:           172.217.0.0/16
NetName:        GOOGLE
NetHandle:      NET-172-217-0-0-1
Parent:         NET172 (NET-172-0-0-0-0)
NetType:        Direct Allocation
OriginAS:       AS15169
Organization:   Google LLC (GOGL)
RegDate:        2012-04-16
Updated:        2012-04-16
Ref:            https://rdap.arin.net/registry/ip/172.217.0.0


OrgName:        Google LLC
OrgId:          GOGL
Address:        1600 Amphitheatre Parkway
```

Traceroute

Network diagnostic tool that displays route taken to destination and RTT for each hop

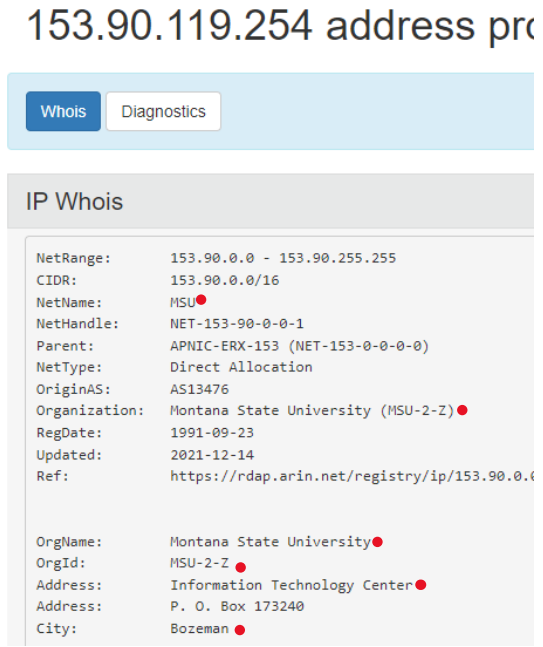
Hop #

RTT time for each packet

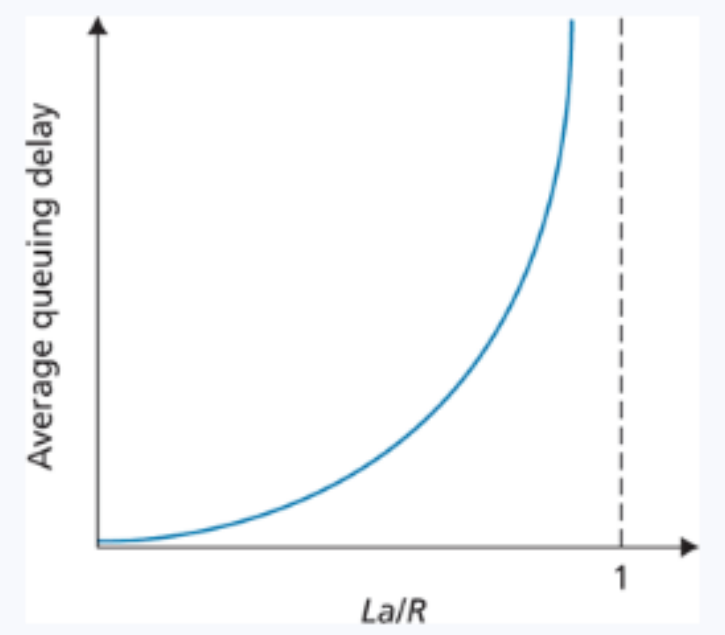
Destination

```
C:\Users\Reese Pearsall>tracert google.com

Tracing route to google.com [172.217.14.238]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  gateway119.254.msu.montana.edu [153.90.119.254]
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4    <1 ms    <1 ms    <1 ms  153.90.125.254
  5    <1 ms    <1 ms    <1 ms  10.196.6.10
  6     1 ms     1 ms    <1 ms  rnedge-prodfw.msu.montana.edu [192.105.205.131]
  7    15 ms    15 ms    16 ms  ae13--538.icar-sttl1-2.infra.pnw-gigapop.net [209.124.190.212]
  8    15 ms    15 ms    15 ms  209.124.190.202
  9    17 ms    17 ms    17 ms  142.251.70.99
 10    16 ms    16 ms    16 ms  209.85.254.247
 11    15 ms    15 ms    15 ms  sea30s02-in-f14.1e100.net [172.217.14.238]

Trace complete.
```

whois- provides registration data of a domain or IP address



Packet Loss



Queues have finite space.

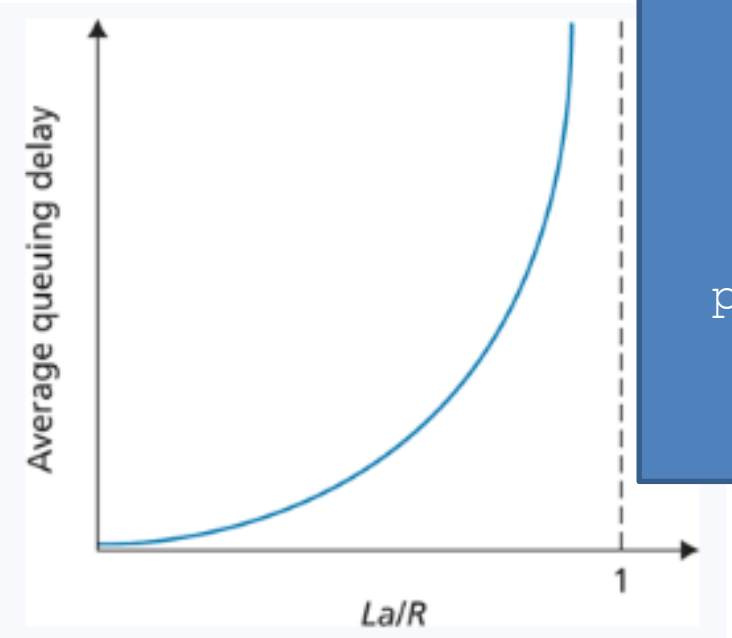
If a router has no place to store the packet, the packet will be **dropped** or **lost**

The number of lost packets will increase as **traffic intensity** increases

In addition to delay, the **probability of packet loss** is helpful to calculate

Packet Loss

Queues have finite space.



Ping test

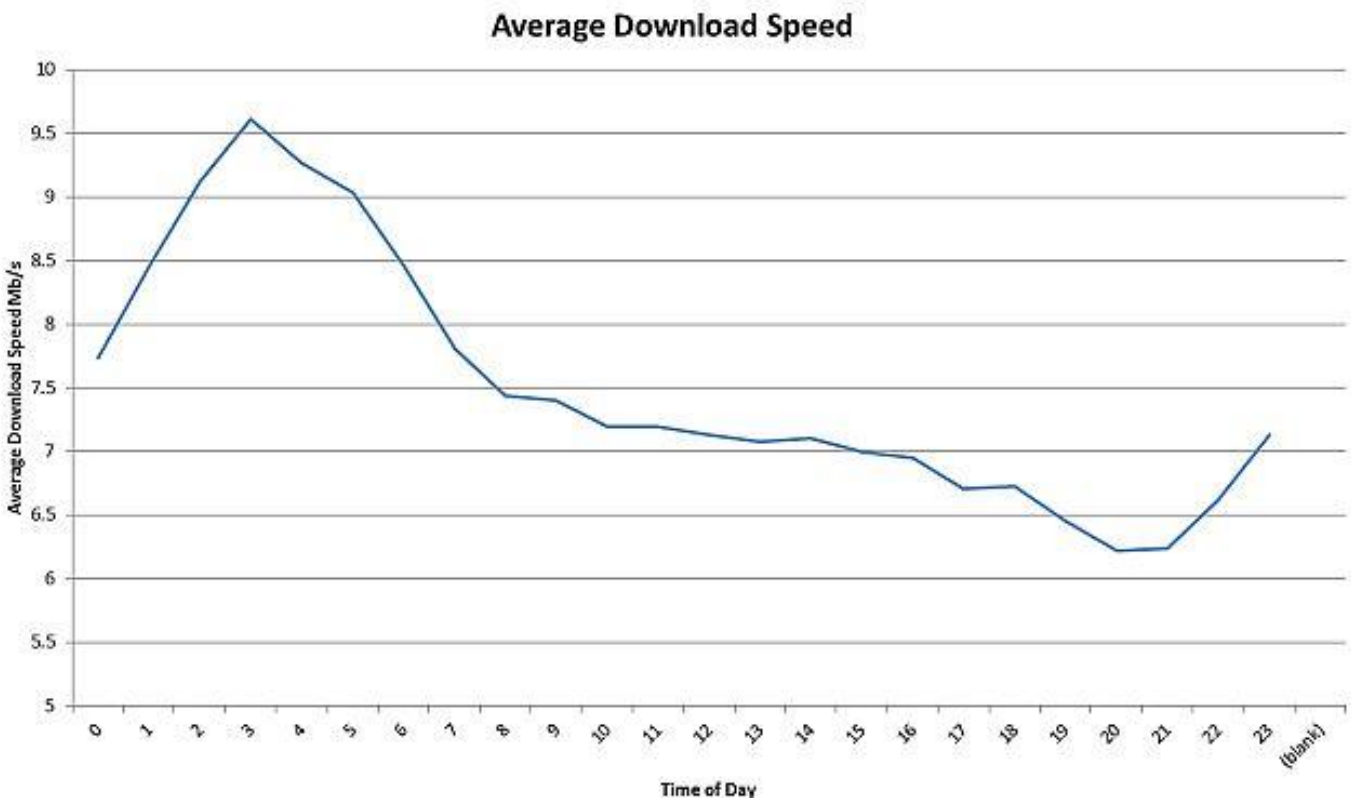
```
ping -n 100 8.8.8.8
ping -i 20 -n 10 131.255.7.26
```

e to store
will be

ckets will
ensity increases

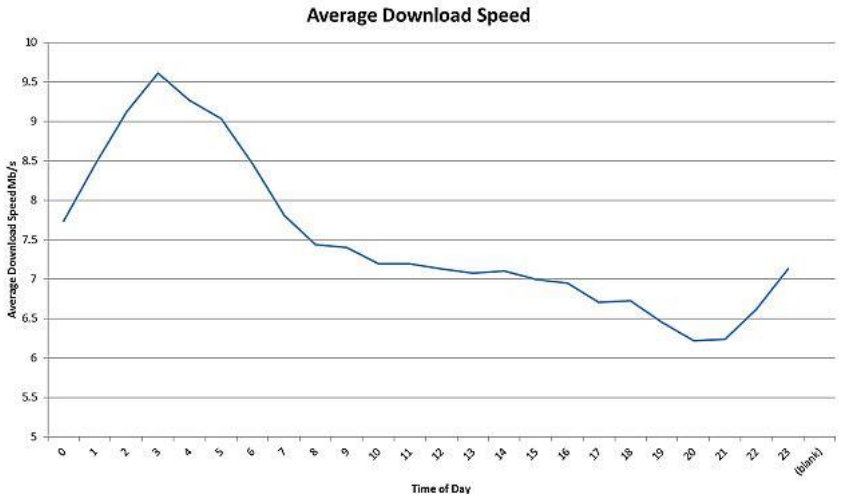
In addition to delay, the **probability of packet loss** is helpful to calculate

Throughput

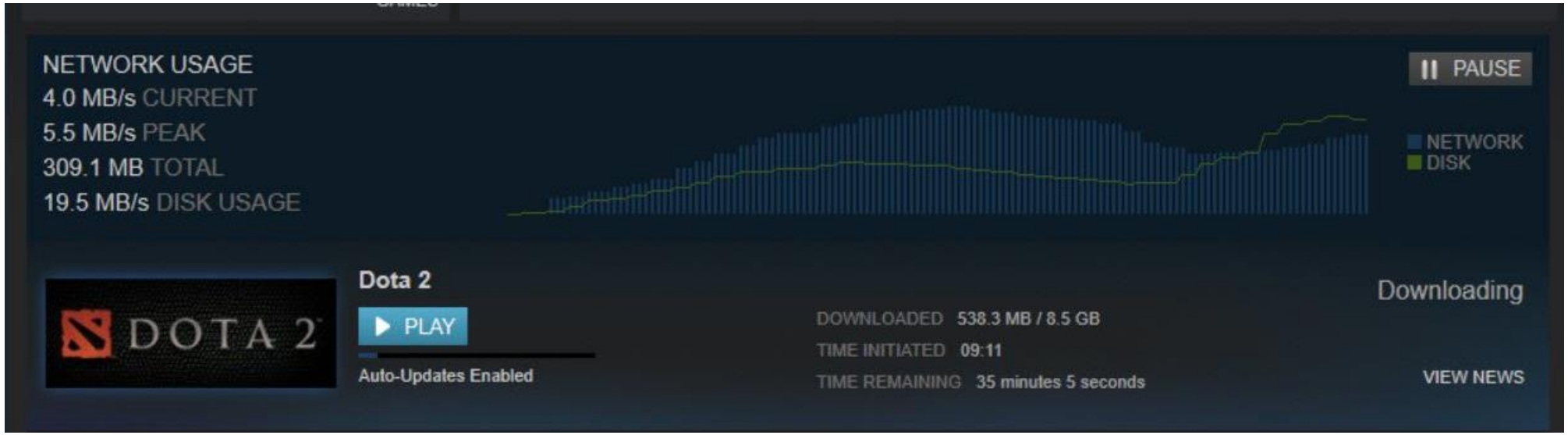


Throughput is the amount of data transferred from one place to another within a given time period

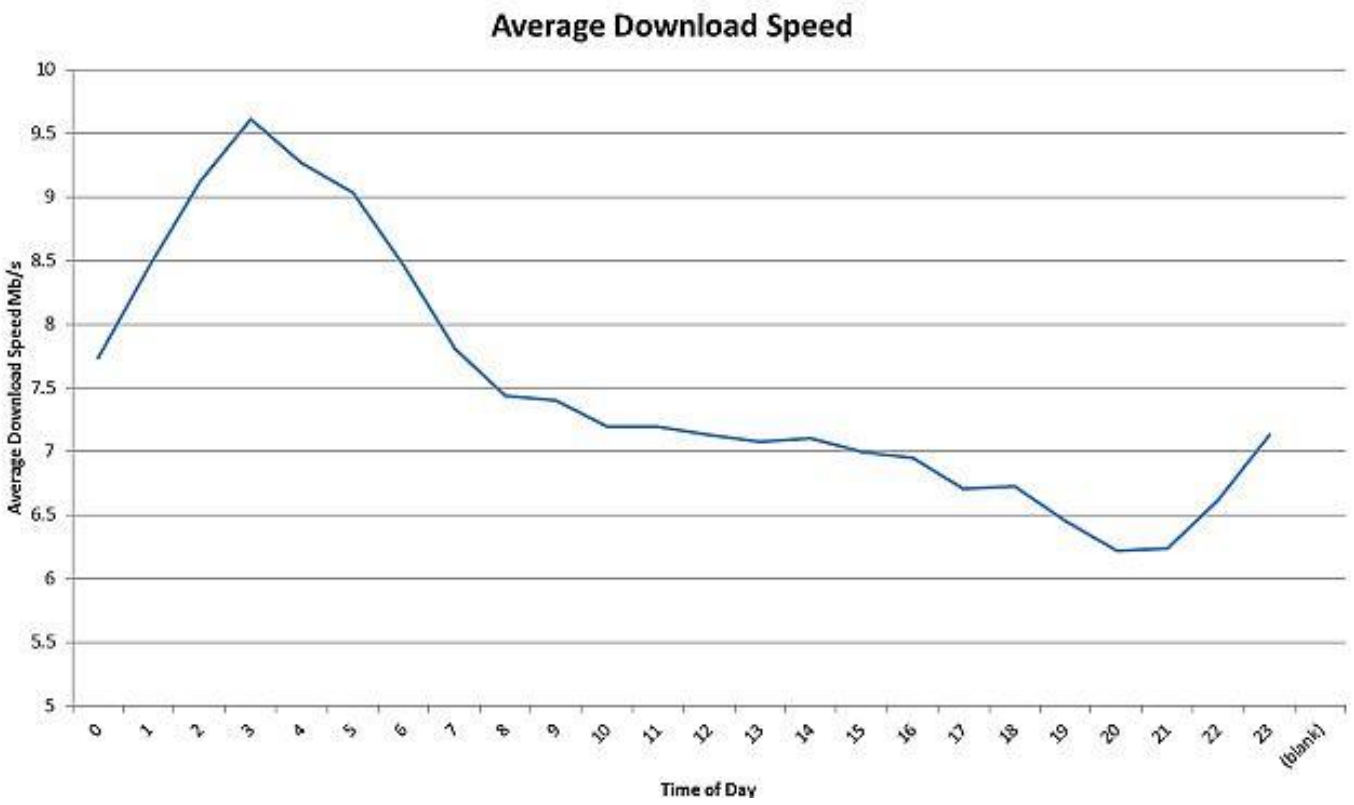
Throughput



Throughput is the amount of data transferred from one place to another within a given time period



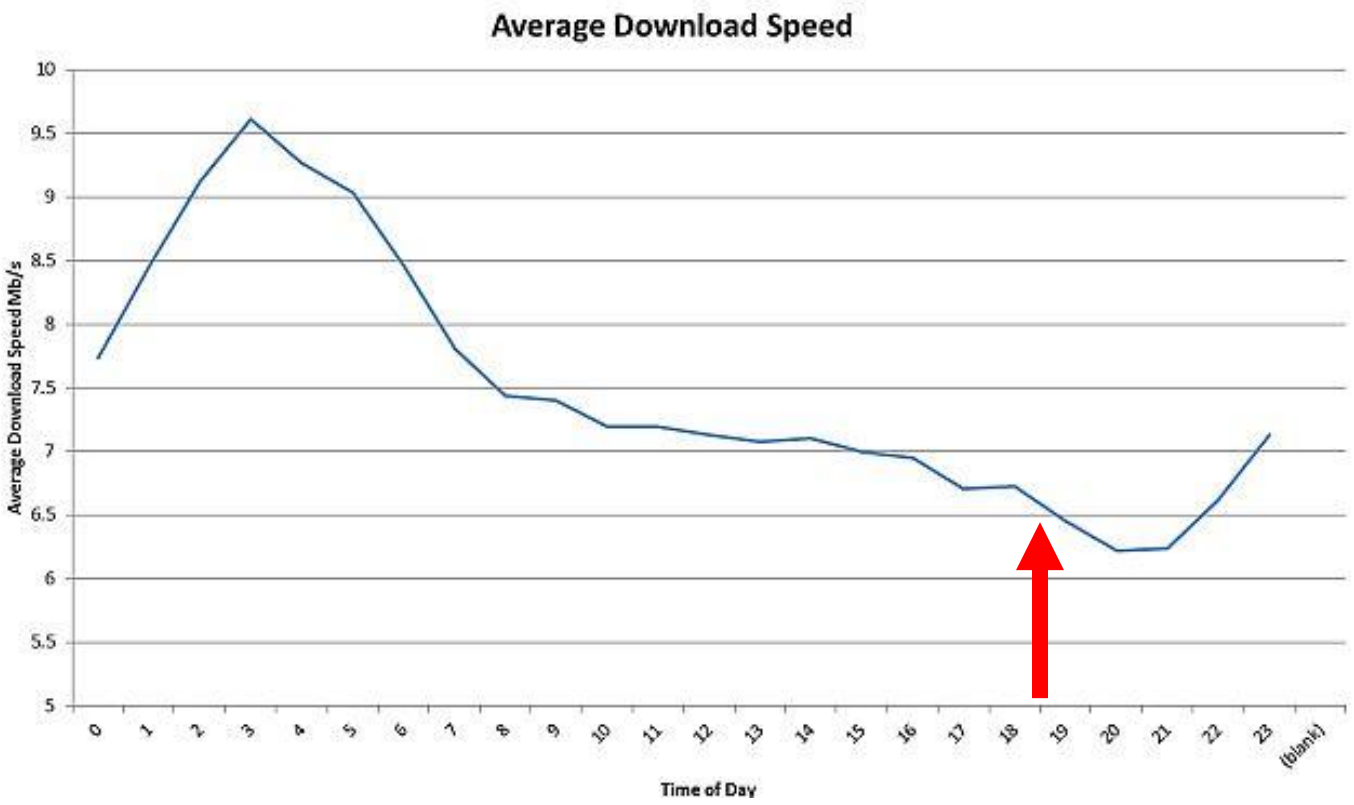
Throughput



Throughput is the amount of data transferred from one place to another within a given time period

Instantaneous throughput is the throughput for a given point in time

Throughput

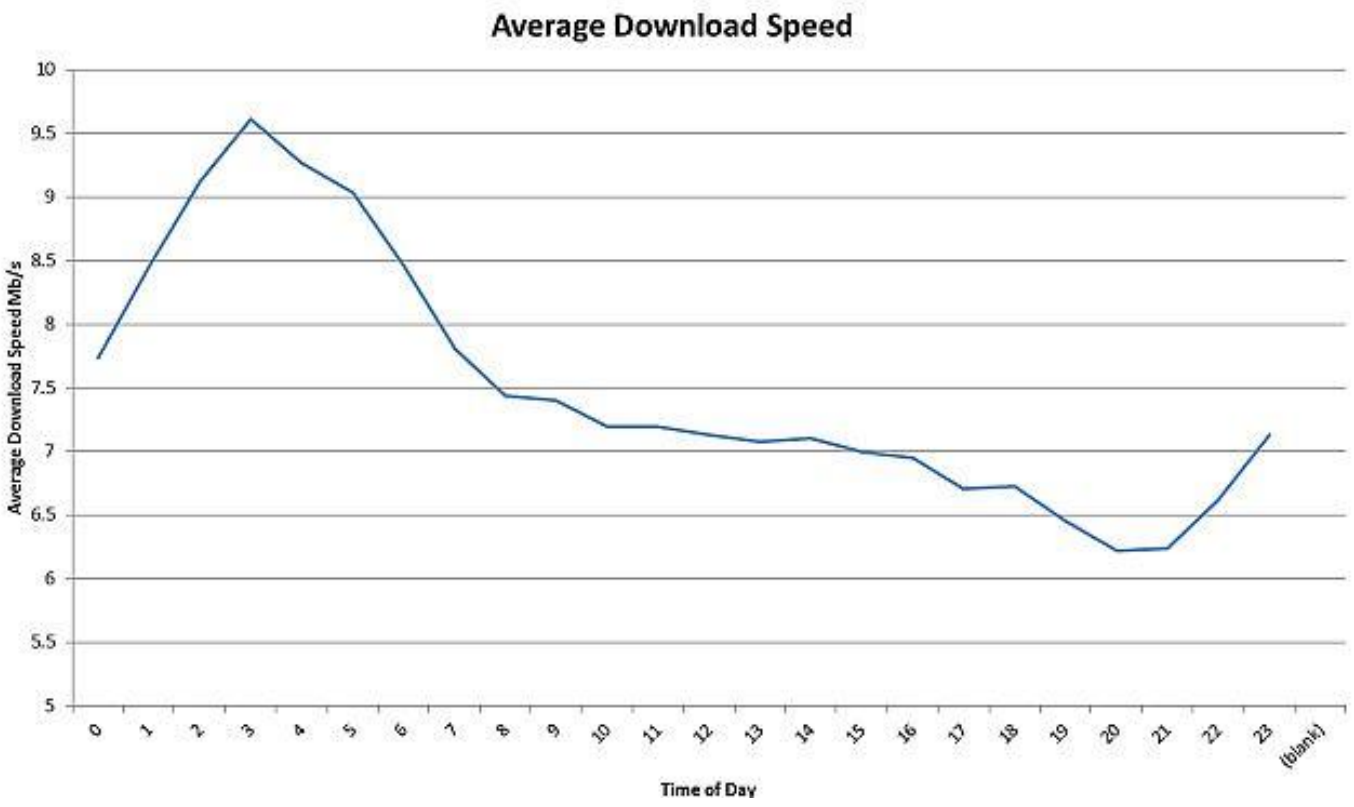


Throughput is the amount of data transferred from one place to another within a given time period

Instantaneous throughput is the throughput for a given point in time

The instantaneous throughput at 7:00 PM is roughly 6.5 Mbps

Throughput



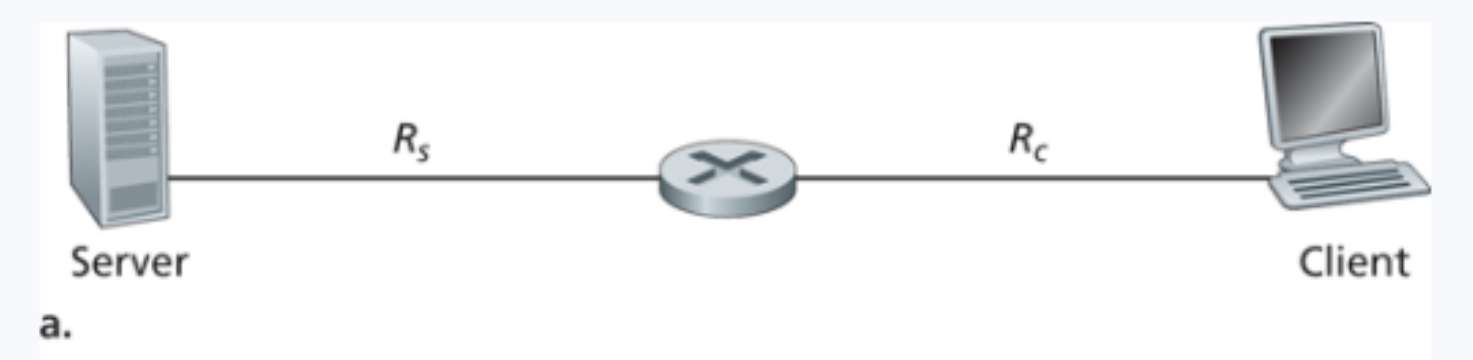
Throughput is the amount of data transferred from one place to another within a given time period

Instantaneous throughput is the throughput for a given point in time

In general, if F bits takes T seconds to transfer from one endpoint to another, the **average throughput** is:

$$\frac{F}{T}$$

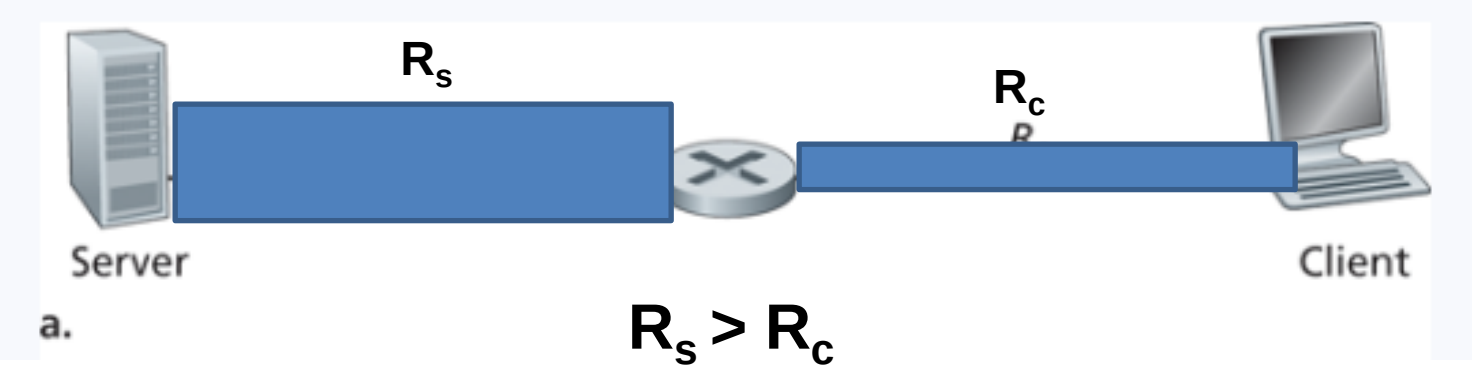
Throughput



R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

Throughput

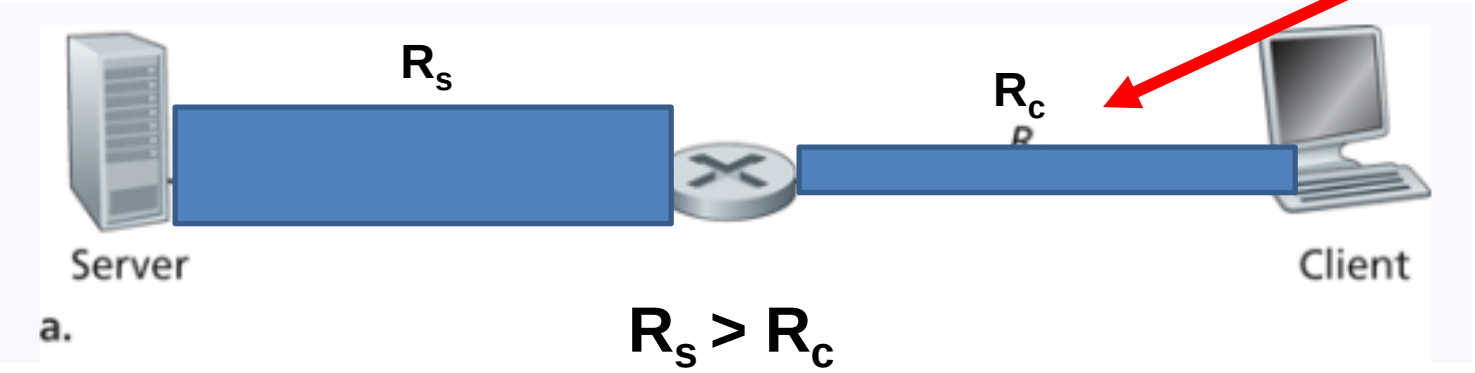


R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

Throughput

Point in network where flow of data is impaired or stopped



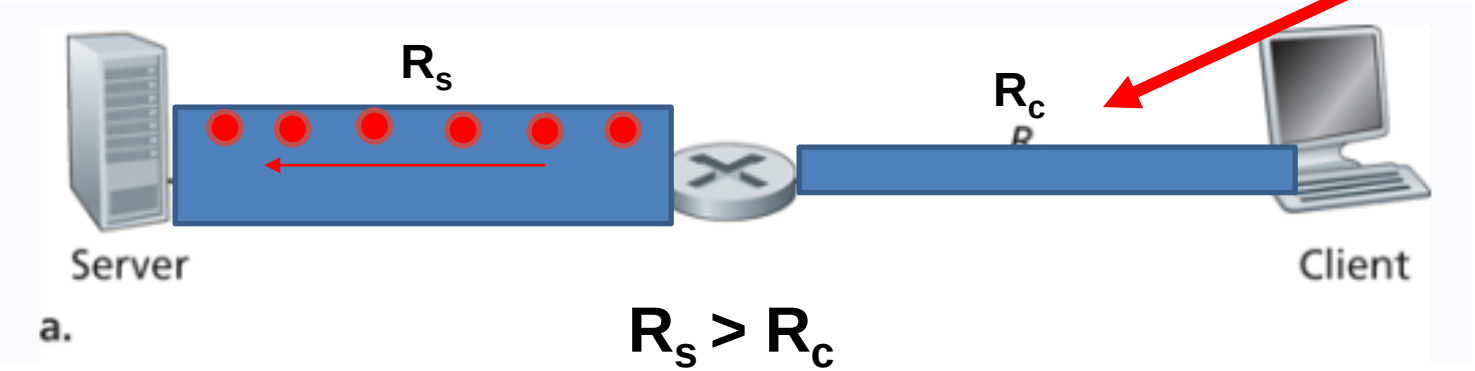
R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

Throughput

Point in network where flow of data is impaired or stopped

Bottleneck



$R_s > R_c$

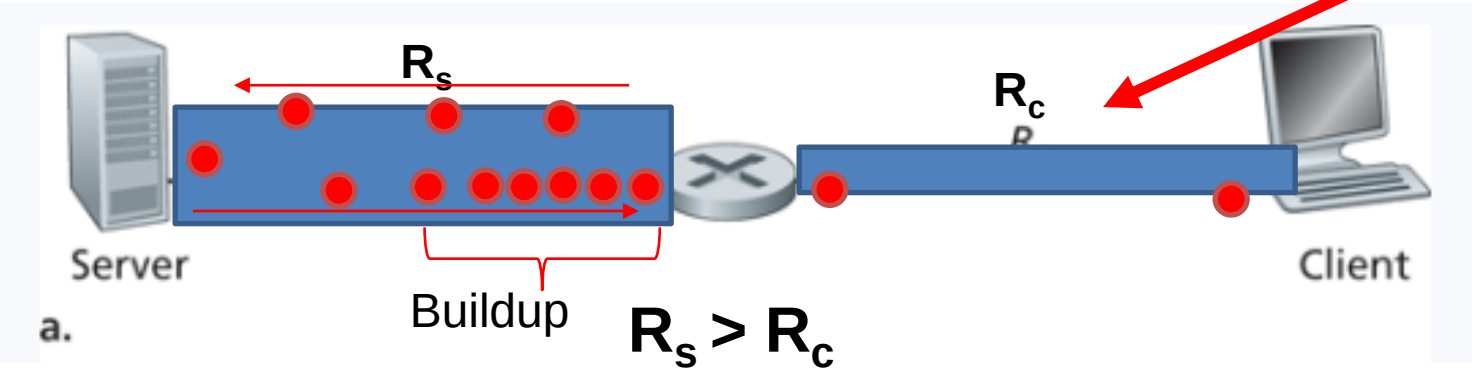
R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

Throughput

Point in network where flow of data is impaired or stopped

Bottleneck



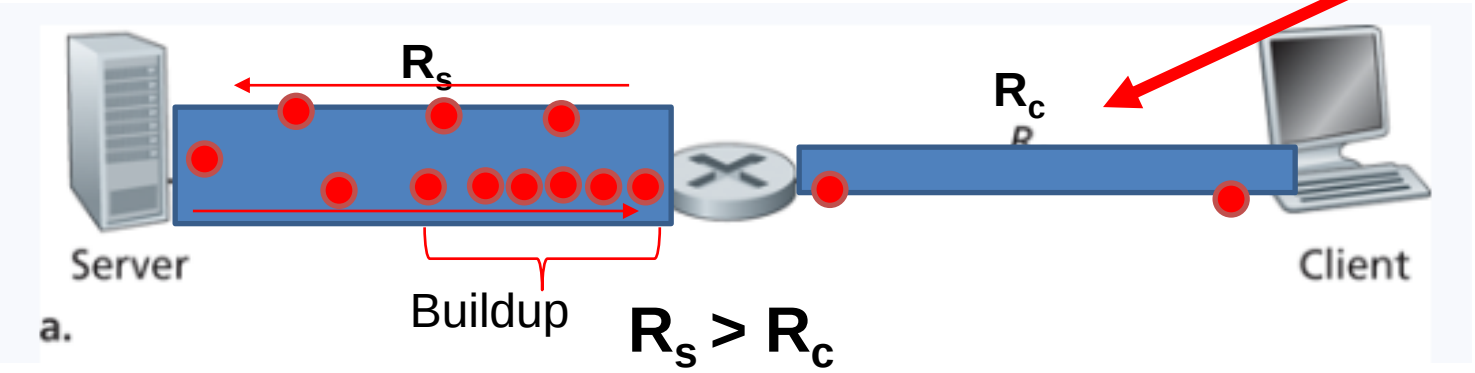
R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

Throughput

Point in network where flow of data is impaired or stopped

Bottleneck



R_s = transmission rate for server to router R_c = transmission rate for client to router

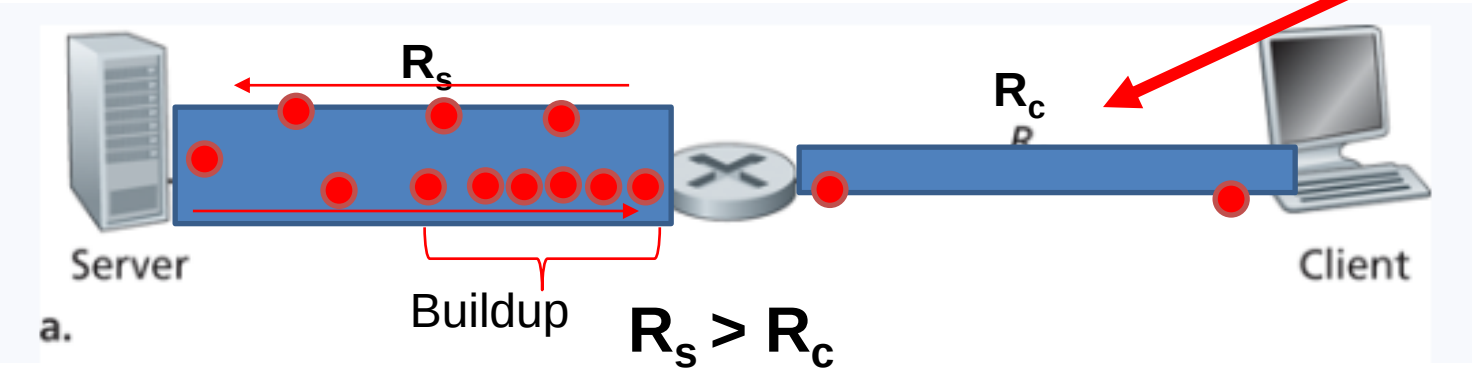
Throughput for server to client?

Dependent on the bottleneck link!

Throughput

Point in network where flow of data is impaired or stopped

Bottleneck



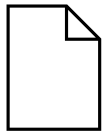
R_s = transmission rate for server to router R_c = transmission rate for client to router

Throughput for server to client?

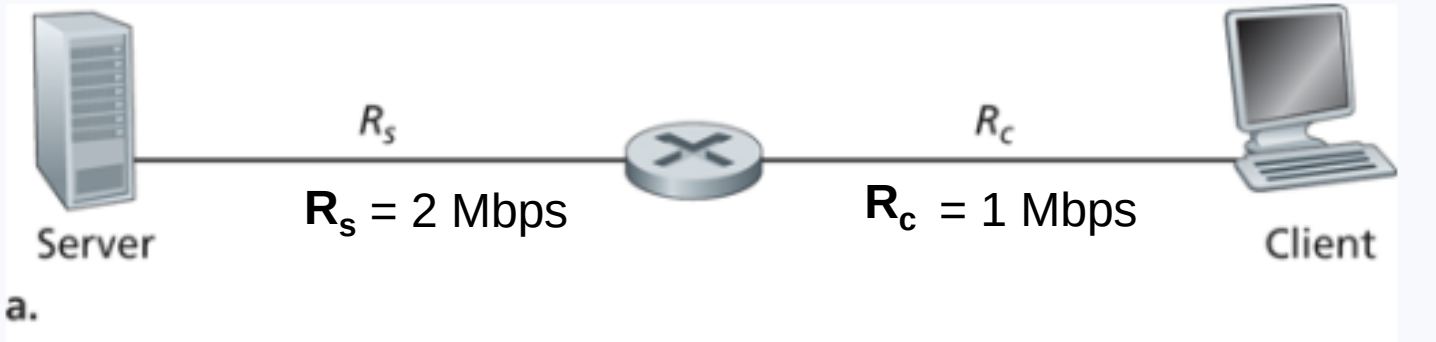
$$\frac{F}{\min\{R_s, R_c\}}$$

Dependent on the bottleneck link!

Throughput



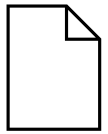
F = 32,000,000 bits



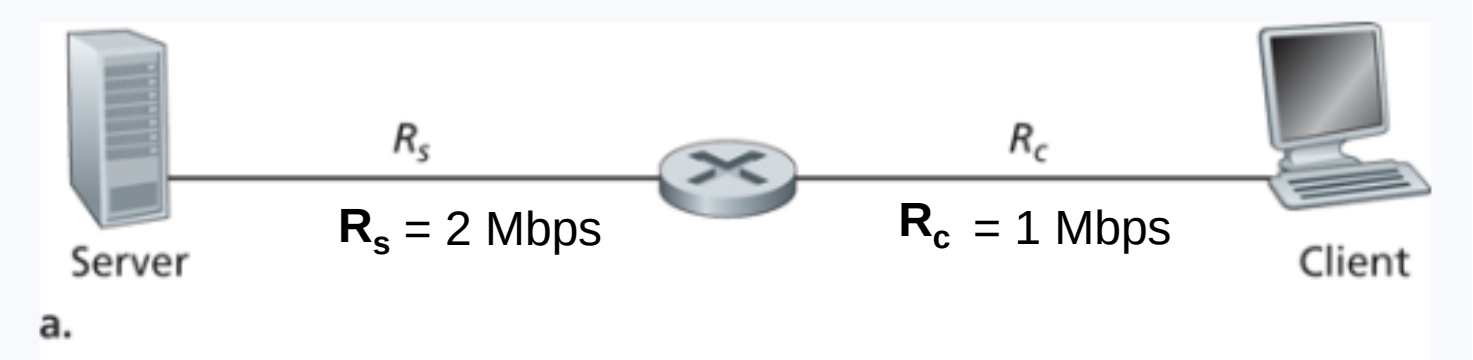
R_s = transmission rate for server to router R_c = transmission rate for client to router

Time needed to transfer file?

Throughput



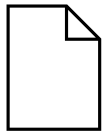
F = 32,000,000 bits = 32 Mb



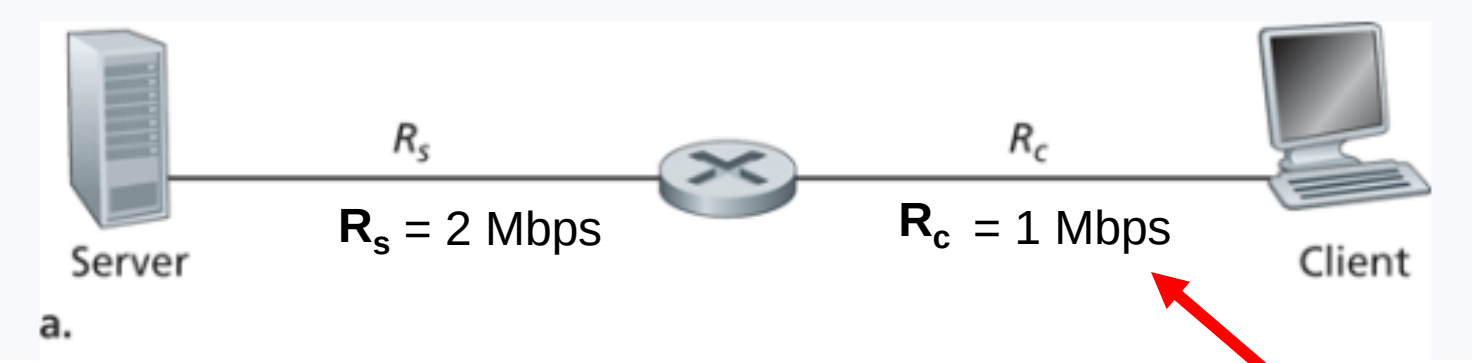
R_s = transmission rate for server to router R_c = transmission rate for client to router

Time needed to transfer file?

Throughput



F = 32,000,000 bits = 32 Mb



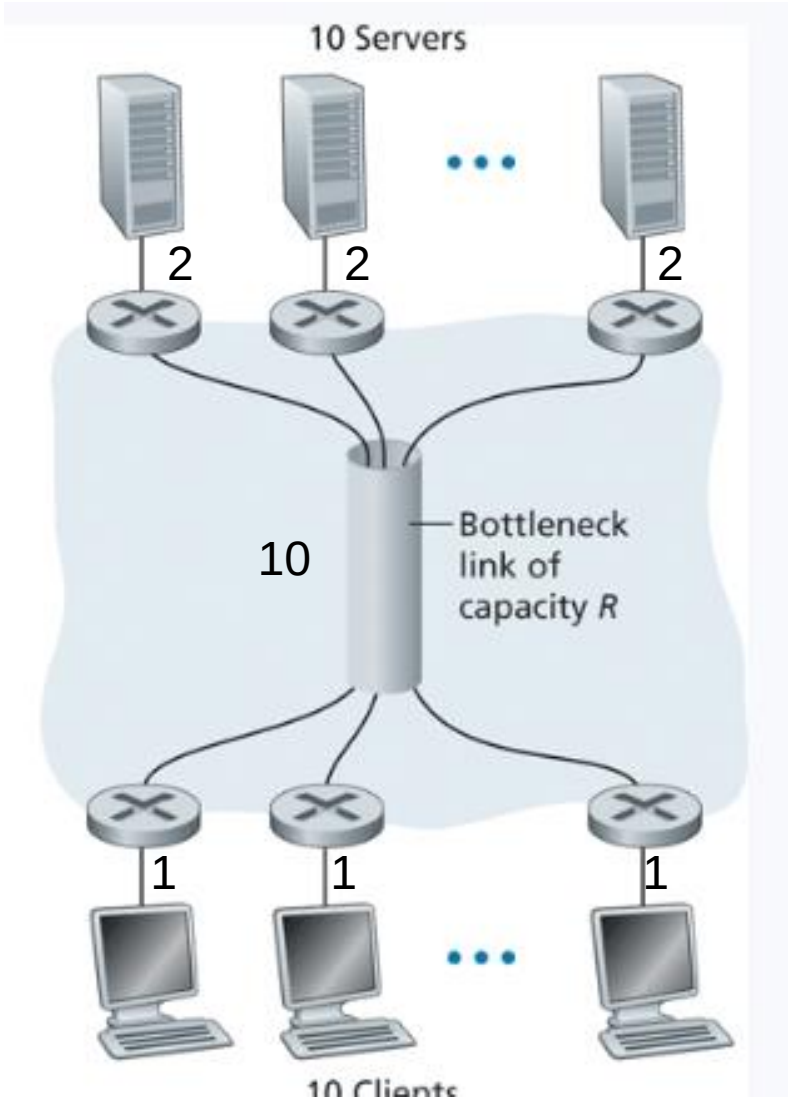
Dependent on the bottleneck

R_s = transmission rate for server to router R_c = transmission rate for client to router

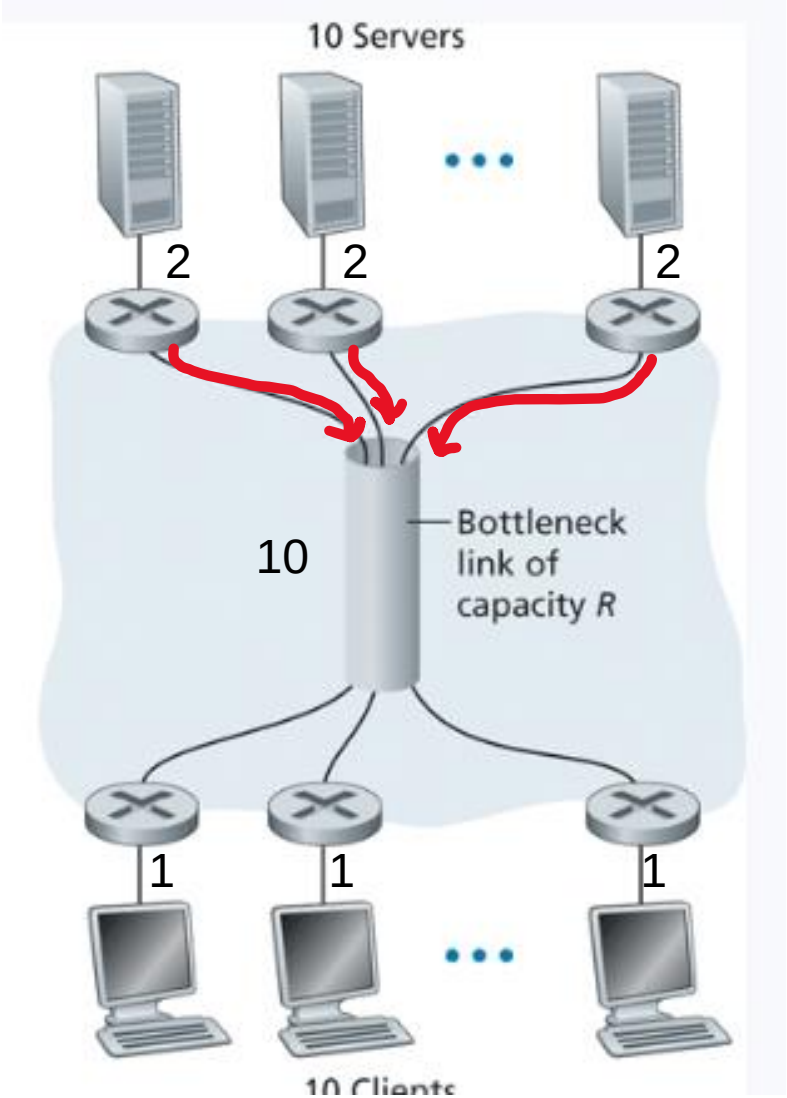
Time needed to transfer file?

32 seconds

Throughput


10 Servers
2
2
2
10
Bottleneck
link of
capacity R
1
1
1
10 Clients

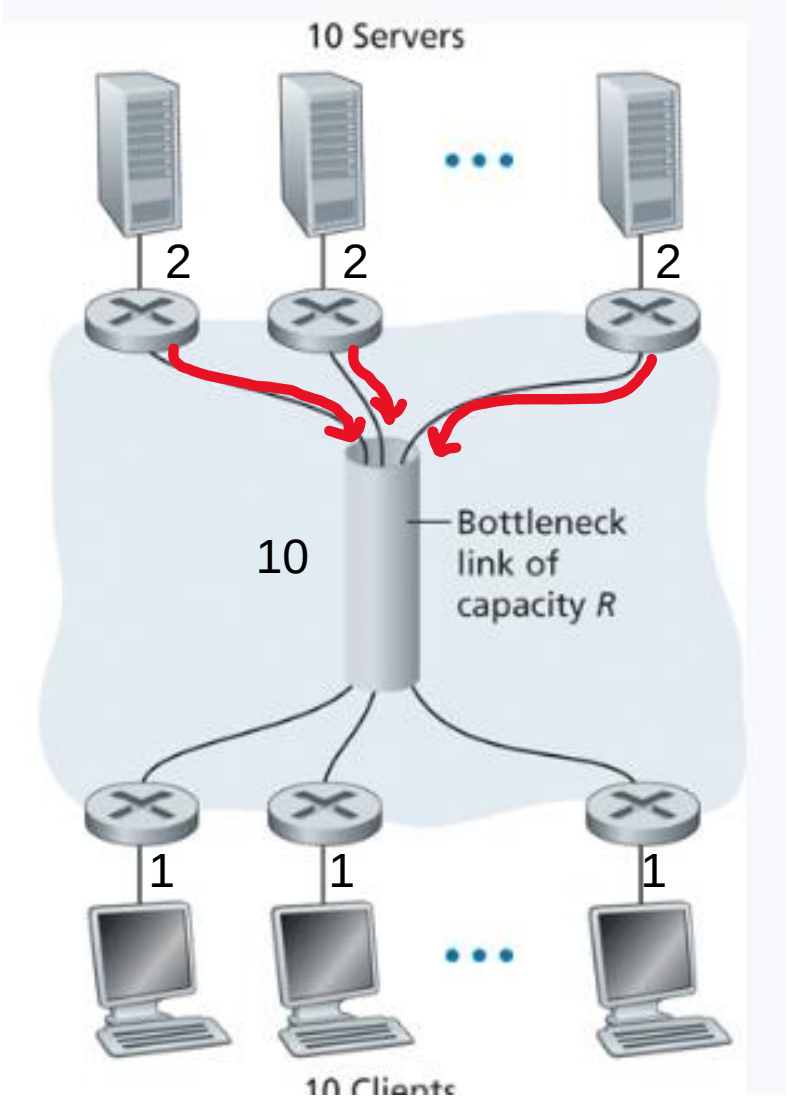
Throughput



Link is shared across 10 servers

So each link supports 0.5 Mbps

Throughput



Link is shared across 10 servers

So each link supports 0.5 Mbps

Sometimes, the bottleneck won't be the link with the slowest transmission rate