Network Thinking

Web over Internet
The HTTP/TCP/IP Protocol Stack
一个常用的协议栈：**HTTP/TCP/IP**协议栈

zxu@ict.ac.cn
zhangjialin@ict.ac.cn

Outline

- What is network thinking?
- Network terms
- Connectivity
  - Naming
  - Topology
- Protocol stack
  - The Web over TCP/IP stack
  - Web programming
- Network laws
  - Performance metrics
  - Network effect
- Responsible computing

These slides acknowledge sources for additional data not cited in the textbook

4. Protocol stack 协议栈

- A network uses a protocol stack to communicate messages
  - A set of layers of protocols
  - We focus on one stack
    - Web over Internet
- Key terms
  - Message and packet
    消息 vs. 分组（包、数据包）
    - Packet is part of a message
  - Circuit switching
    versus packet switching
    线路交换 vs. 分组交换
  - The Web over Internet stack
    - HTTP
    - TCP
    - IP
    - Ethernet or WiFi
    - Wired versus wireless

The Web over Internet Stack

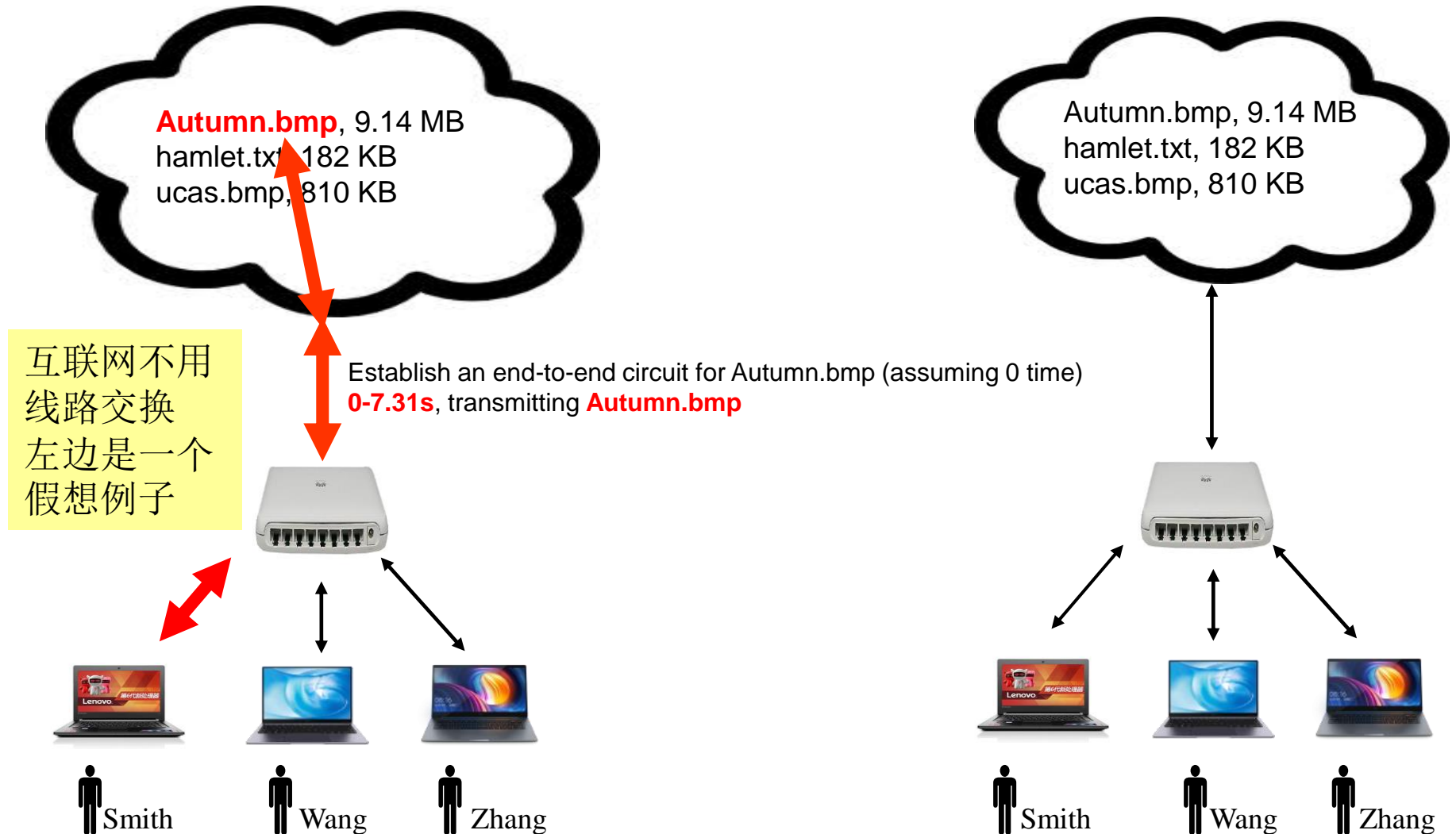| Layer	Protocol	Purpose
Application Layer Layer 5	HTTP	Access hypertext resources on a Web server from a Web client
Transport Layer Layer 4	TCP	Reliably transfer packets between two Internet hosts
Network Layer Layer 3	IP	Transfer packets between two Internet hosts in the best-effort way
Data Link Layer Layer 2	Ethernet, WiFi	Reliably transfer packets between two homogeneously connected devices
Physical Layer Layer 1	Wired or wireless, electrical or optical, cables or waveforms	Provide physical communication channels Transfer signals of individual bits

Circuit switch   vs.   packet switch

Assumptions for both systems:
(1) 10 Mbps; (2) all three tasks start at 0; (3) ignore all overheads

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

互联网不用
线路交换
左边是一个
假想例子

Establish an end-to-end circuit for Autumn.bmp (assuming 0 time)
0-7.31s, transmitting **Autumn.bmp**

Smith   Wang   Zhang

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB
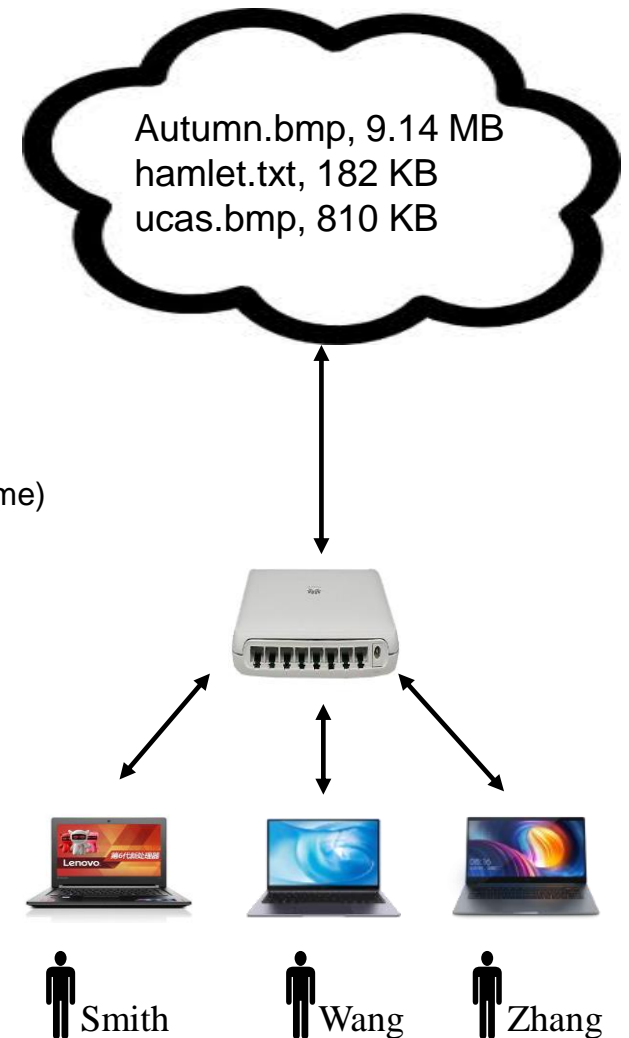
Smith   Wang   Zhang

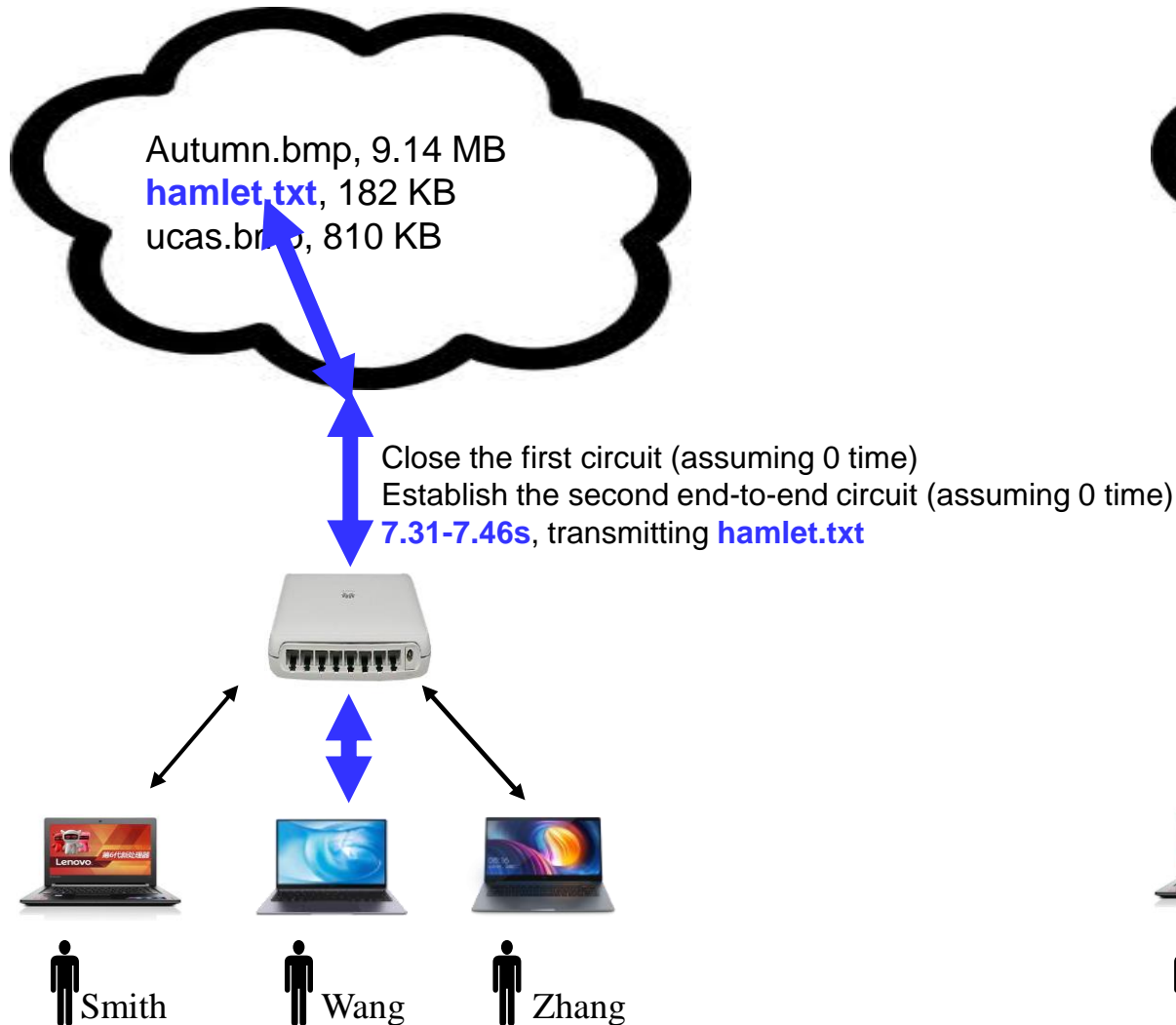# Circuit switch   vs.   packet switch

Assumptions for both systems:
(1) 10 Mbps; (2) all three tasks start at 0; (3) ignore all overheads

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

Close the first circuit (assuming 0 time)
Establish the second end-to-end circuit (assuming 0 time)
7.31-7.46s, transmitting **hamlet.txt**

Smith   Wang   Zhang

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB
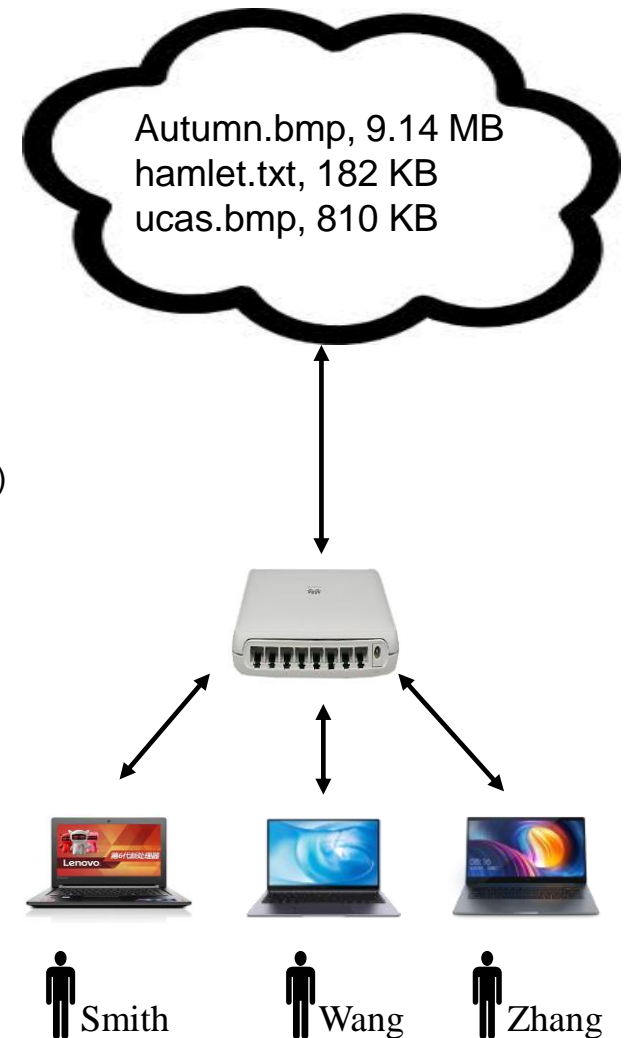
Smith   Wang   Zhang

Circuit switch   vs.   packet switch

Assumptions for both systems:
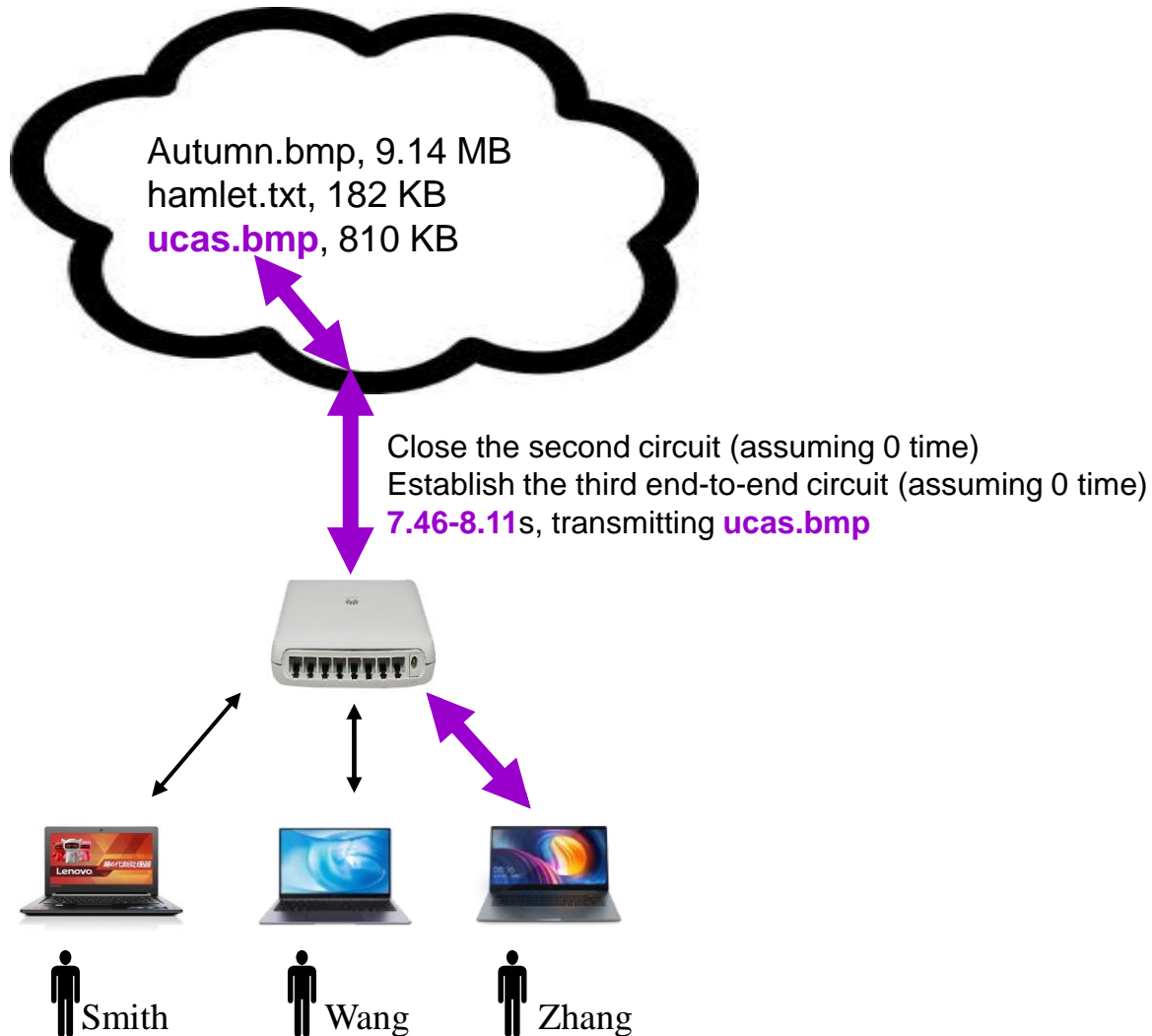(1) 10 Mbps; (2) all three tasks start at 0; (3) ignore all overheads

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

Close the second circuit (assuming 0 time)
Establish the third end-to-end circuit (assuming 0 time)
7.46-8.11s, transmitting **ucas.bmp**

Smith   Wang   Zhang

Smith   Wang   Zhang

Circuit switch   vs.   packet switch

Assumptions for both systems:
(1) 10 Mbps; (2) all three tasks start at 0; (3) ignore all overheads
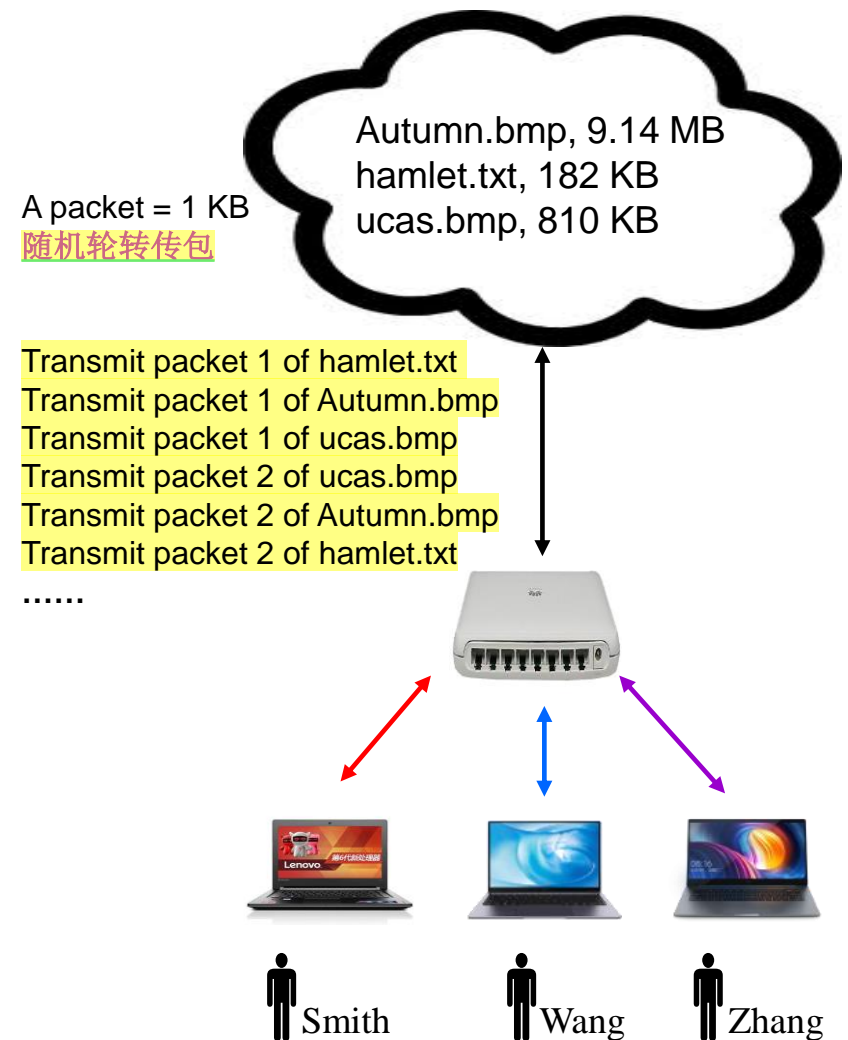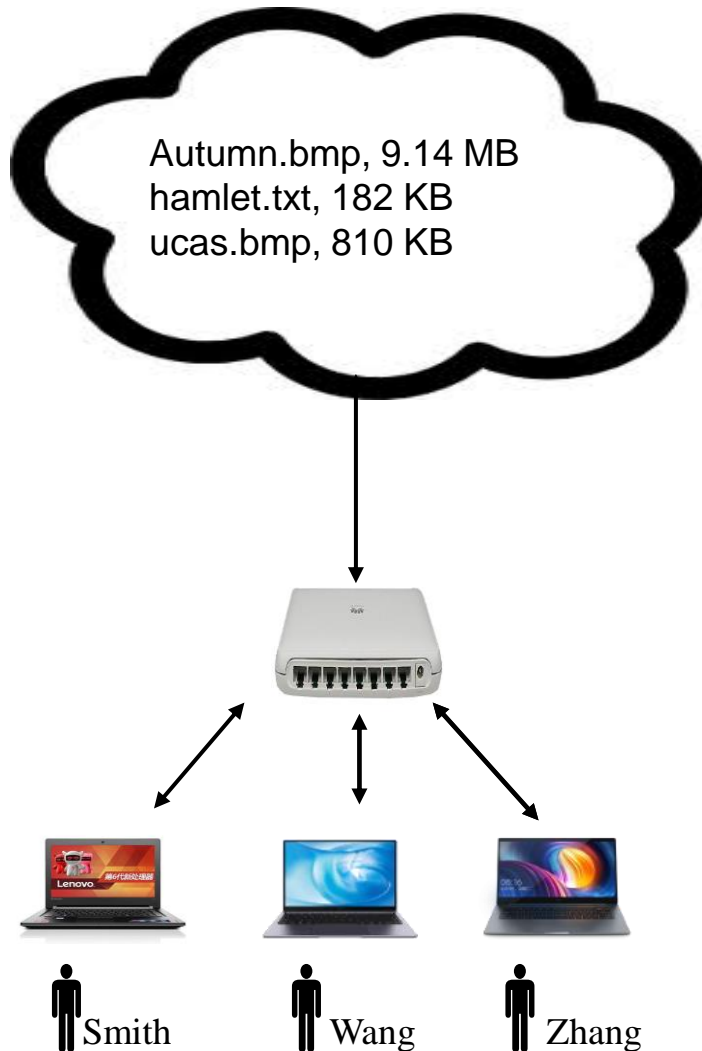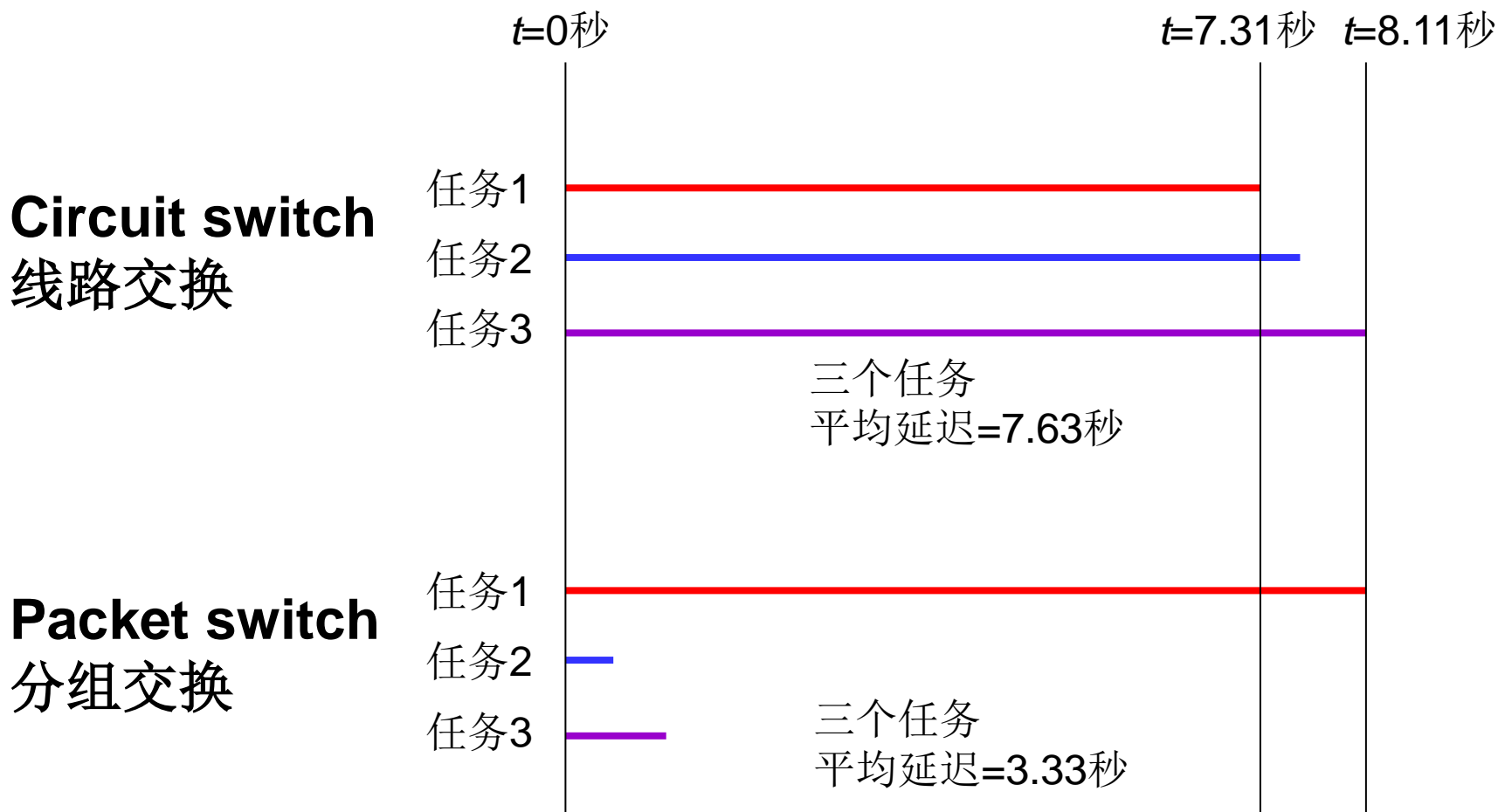

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

Smith   Wang   Zhang

A packet = 1 KB
随机轮转传包

Autumn.bmp, 9.14 MB
hamlet.txt, 182 KB
ucas.bmp, 810 KB

Transmit packet 1 of hamlet.txt
Transmit packet 1 of Autumn.bmp
Transmit packet 1 of ucas.bmp
Transmit packet 2 of ucas.bmp
Transmit packet 2 of Autumn.bmp
Transmit packet 2 of hamlet.txt
......

Smith   Wang   Zhang

假设10 Mbps带宽资源被三个下载任务平均使用

t=0秒　　　　　　　　　　　　　　　t=7.31秒　t=8.11秒

Circuit switch
线路交换

任务1
任务2
任务3

三个任务
平均延迟=7.63秒

Packet switch
分组交换

任务1
任务2
任务3

三个任务
平均延迟=3.33秒

Any packet has two parts: header and body

- Packet body is the payload data　　　　　　　　包体是载荷数据
- Packet header holds various metadata　　　　　包头是元数据
  - **Addresses** of source and destination nodes　　　　　地址
  - Error check information, e.g., Cyclic Redundancy Check (CRC)　纠错
  - Other information, e.g., control information　　　　　控制
  - Part of header may come after body
- Think of post mail
  - Body　　　　=　　　　Letter　　包体 = 信
  - Header　　　=　　　　Envelop　　包头 = 信封

Format of an Ethernet packet
How many bytes are used for packet header?

7 bytes	1 byte	6 bytes	6 bytes	2 bytes	42-1500 bytes	4 bytes
Preamble	Frame Delimiter	**Destination MAC Address**	**Source MAC Address**	Type	**Data (Payload)**	CRC

WiFi与以太网消息包（**Frame**）格式

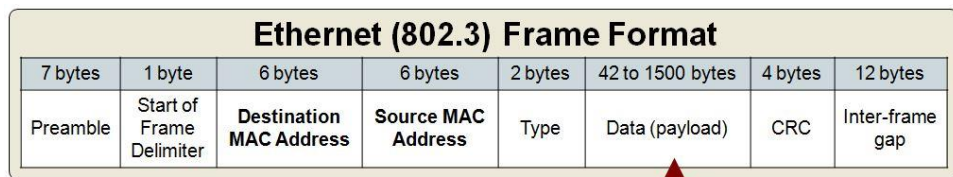- 包头+包体，部分包头也可以出现在包体后面 header + body
- 包体也称为 数据data、 负载payload
- <mark>包头也称为 元数据metadata</mark>
  - 关于数据的数据
  - 包括
    - <mark>地址</mark>
    - <mark>查错信息</mark>
    - <mark>其他控制信息</mark>

字节数		包头			包体	总计
		小计	查错	地址		
以太网	小包	38	4	12	42	80
	大包	38	4	12	1500	1538
WiFi	小包	34	4	24	0	34
	大包	34	4	24	2312	2346

Ethernet (802.3) Frame Format

7 bytes	1 byte	6 bytes	6 bytes	2 bytes	42 to 1500 bytes	4 bytes	12 bytes
Preamble	Start of Frame Delimiter	**Destination MAC Address**	**Source MAC Address**	Type	Data (payload)	CRC	Inter-frame gap

For TCP/IP communications, the payload for a frame is a packet

WiFi (802.11) Frame Format

2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	2 bytes	6 bytes	0 to 2312 bytes	4 bytes
Frame Control	Duration	MAC Address 1 (Destination)	MAC Address 2 (Source)	MAC Address 3 (Router)	Seq Control	MAC Address 4 (AP)	Data (payload)	CRC

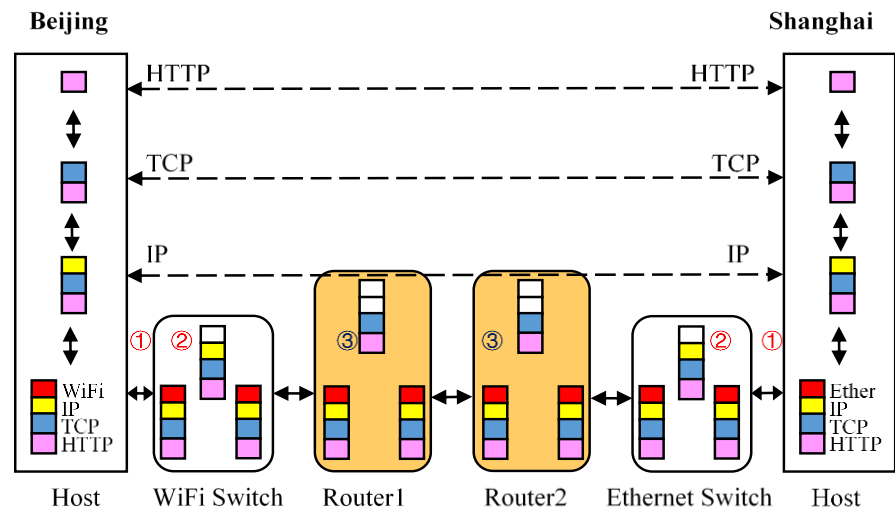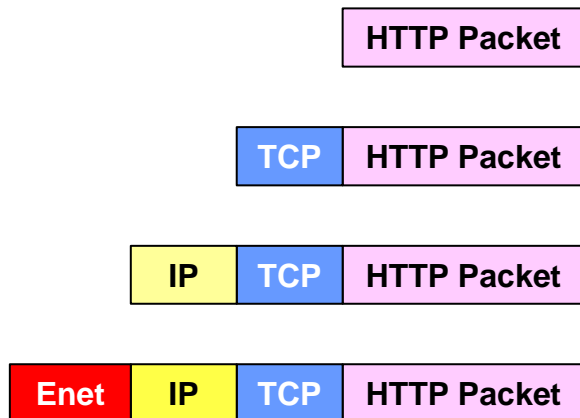# 4.1 HTTP request and response messages

- Request message: http://www.shanghaitech.edu.cn/
  - Sent to the server as a stream of packets
- Response message: the contents of the home page
  - Sent from the server as a stream of packets
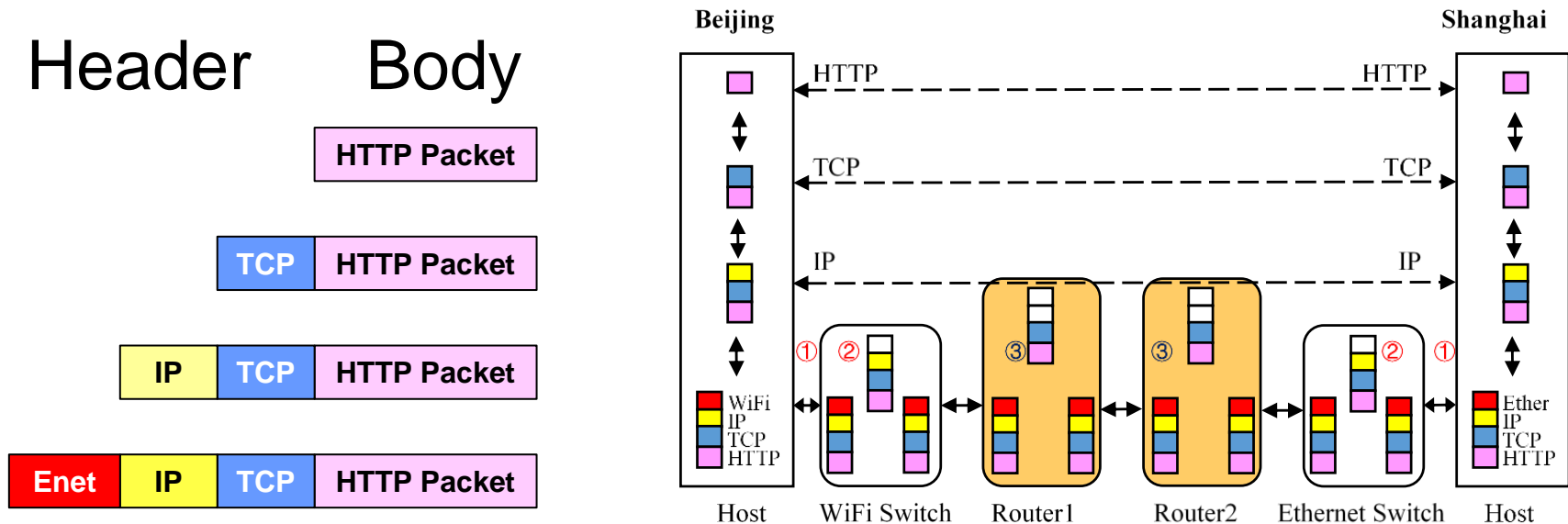
How is the response message communicated

- Response message, i.e., the contents of the home page, is divided into a number of packets, i.e., slices of the message
  - Each HTTP packet is turned into an Ethernet packet as follows
    - HTTP packet (**pink**), actually a slice of the HTTP message, is handed to the TCP layer as the body of a TCP packet
    - TCP layer adds a TCP header (**blue**) to form a TCP packet
    - The TCP packet is handed over to the IP layer as the IP packet body
    - The IP layer adds an IP header (**yellow**) to form an IP packet
    - Finally, the IP packet is handed over to the data link (Ethernet) layer as the Ethernet packet body
    - The Ethernet layer adds an Ethernet header (**red**) to form an Ethernet packet

Header   Body

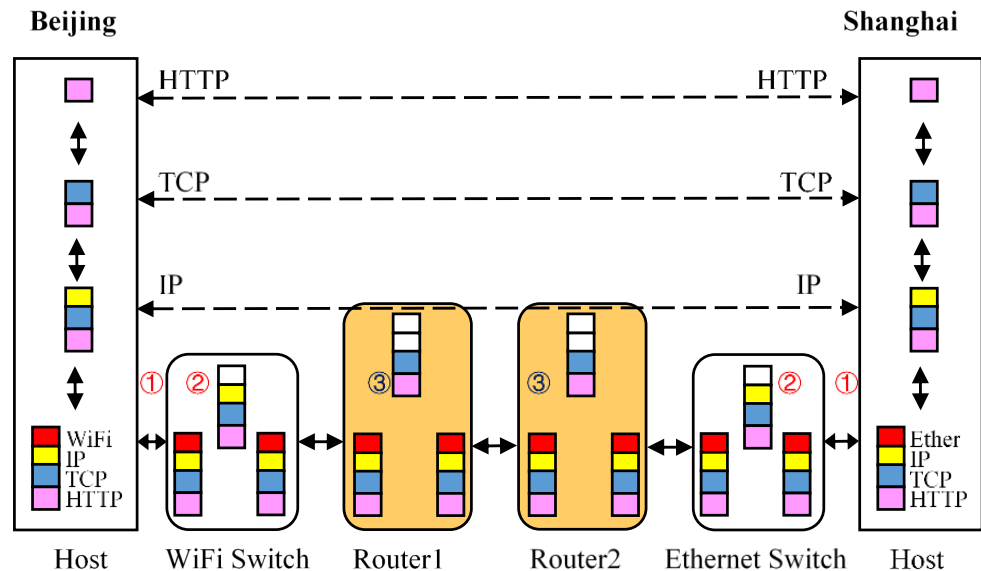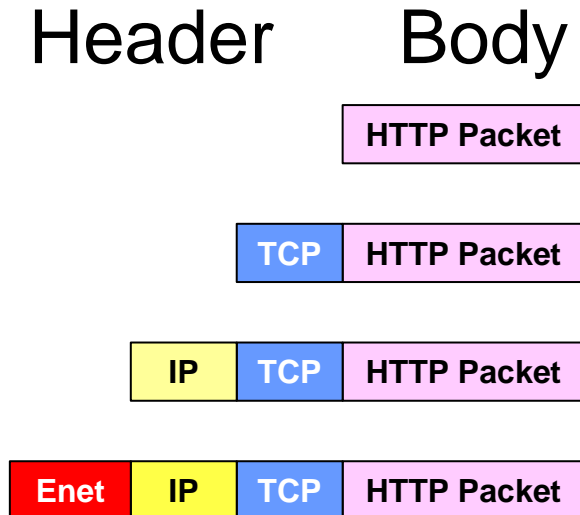|			HTTP Packet
		TCP	HTTP Packet
	IP	**TCP**	HTTP Packet
Enet	**IP**	**TCP**	HTTP Packet

How is the response message communicated

- Each HTTP packet is communicated as follows
  - ① The server host sends an HTTP packet, wrapped as an Ethernet packet, to the Ethernet switch
  - ② The switch opens the packet to reveal the Ethernet header, and then adds a new header (with a new MAC address) to form a new Ethernet packet
  - ③ When the packet arrives at Router2, the router opens the packet to reveal both the Ethernet and the IP headers and then form a new Ethernet packet by reformatting the packet and adding a new Ethernet header
  - Similar steps take place at Router1 (③) and the WiFi Switch (②), and then a WiFi packet arrives at the laptop computer host (①)
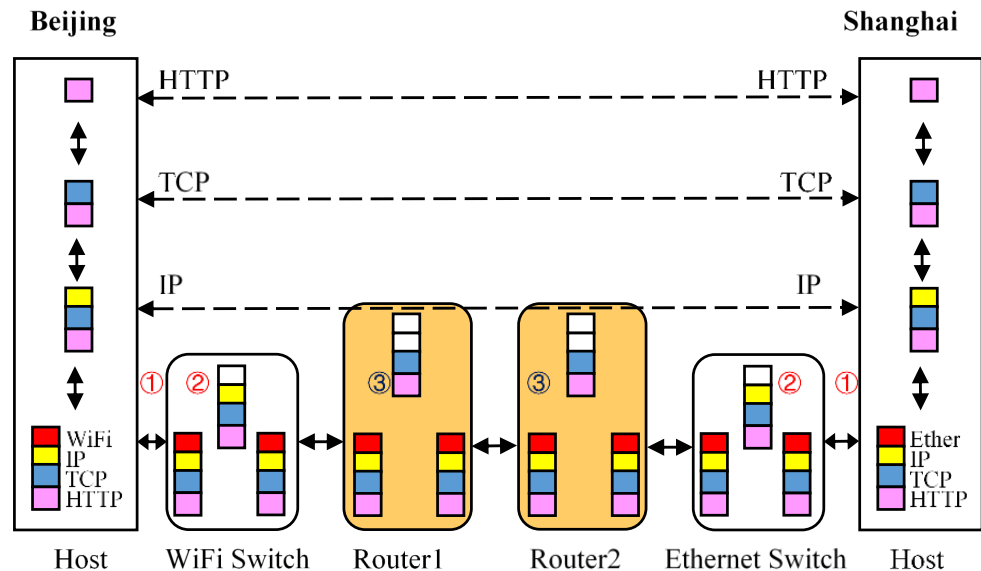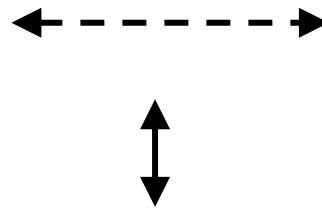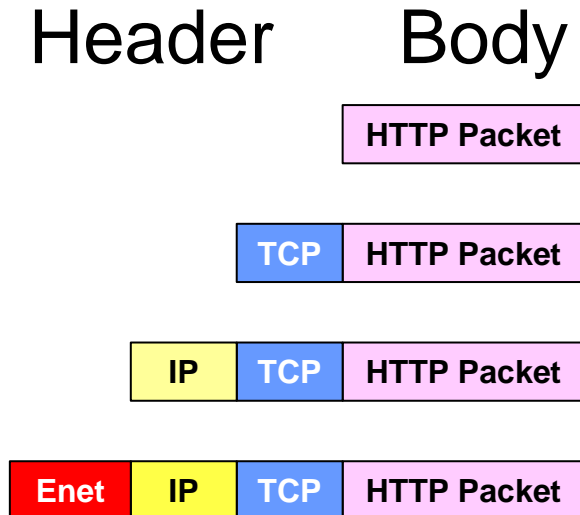
How is the response message communicated

- Each HTTP packet is communicated as follows
  - After a WiFi packet arrives at the laptop computer host in Beijing, it is unpacked by the host (the laptop computer) to reveal
  - the IP packet,
  - the TCP packet, and finally
  - the HTTP packet, i.e., a slice of the message

Header   Body

HTTP Packet

TCP	HTTP Packet

IP	TCP	HTTP Packet

Enet	IP	TCP	HTTP Packet

Beijing   Shanghai

HTTP   HTTP

TCP   TCP

IP   IP

WiFi
IP
TCP
HTTP

Host   WiFi Switch   Router1   Router2   Ethernet Switch   Host
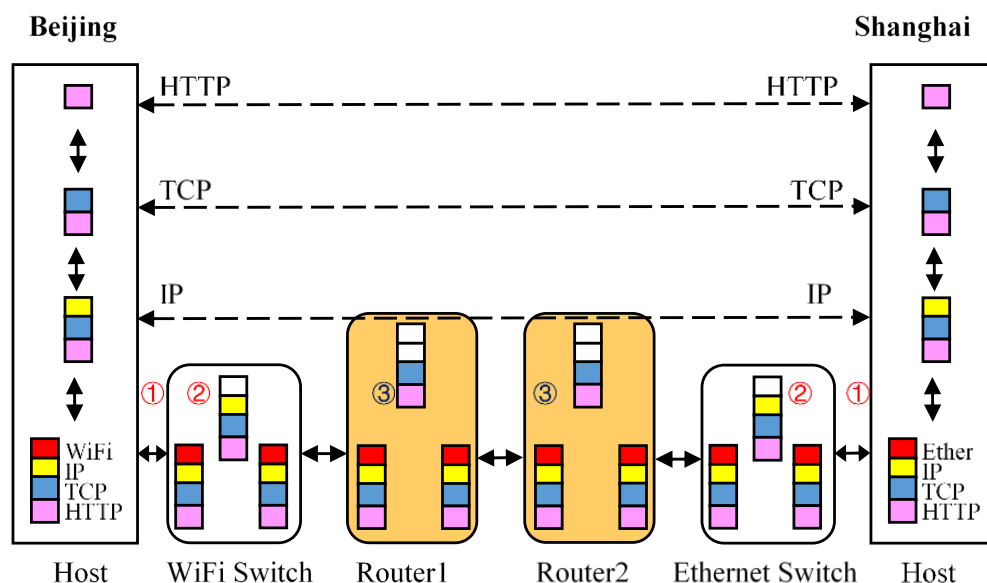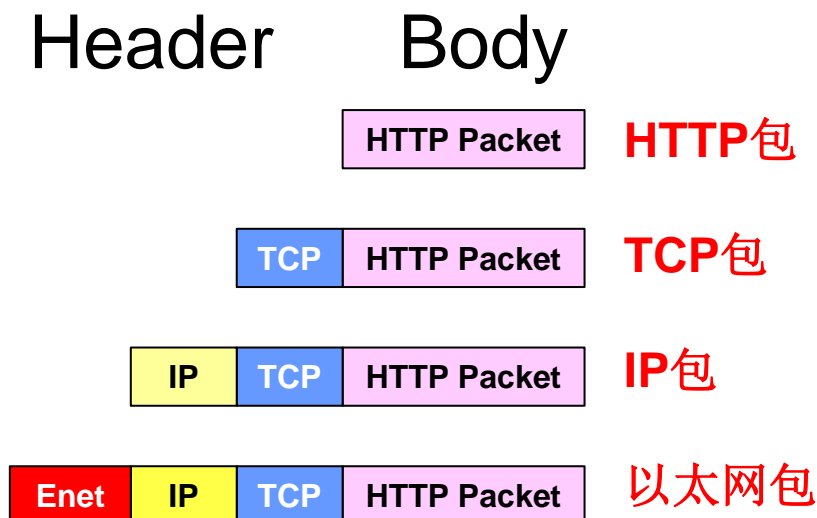
Ether
IP
TCP
HTTP

Does Zhang need to worry about TCP/IP and Ethernet when surfing the Web?

- No! A user only needs to know the peering interface HTTP
- Two types of interfaces
  - Peering interface:

对等接口

  - Service interfaces:

服务接口

Header   Body

HTTP Packet

TCP	HTTP Packet

IP	TCP	HTTP Packet

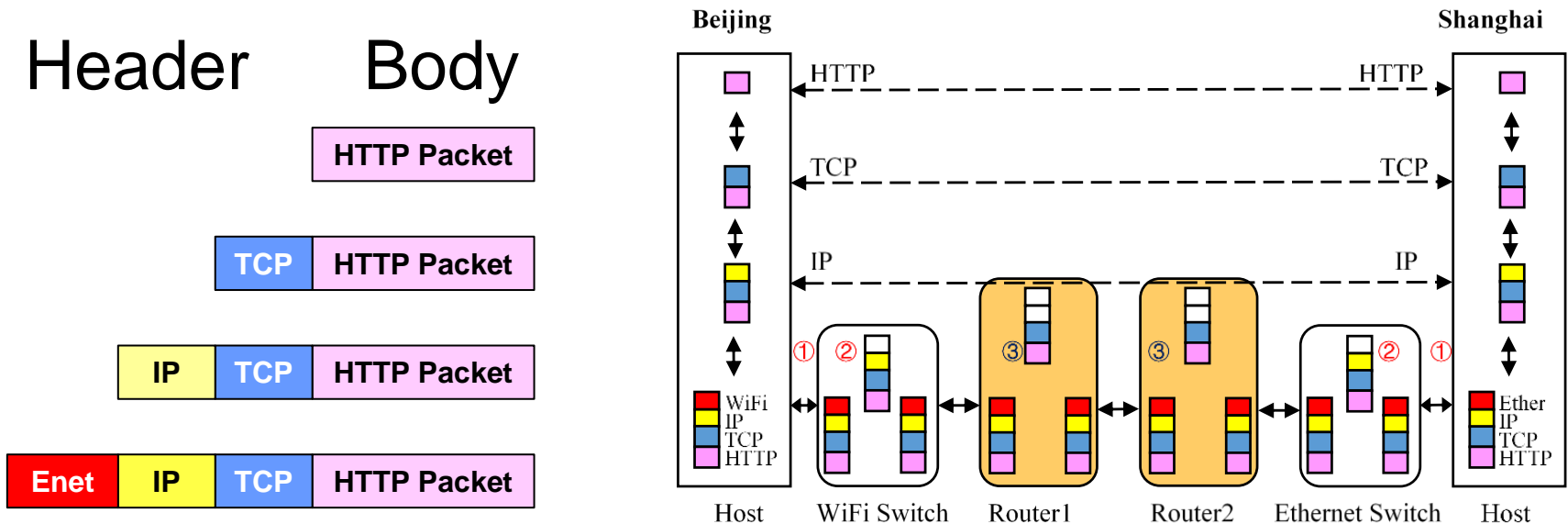| Enet	IP	TCP	HTTP Packet

Can one send an upper layer packet without also sending a lower layer packet?

- Can the Web server in Shanghai send an HTTP packet to Zhang's Web browser in Beijing, without also sending a data link layer packet, e.g., an Ethernet frame?

- No! 不能只传上层数据包(如TCP包)，而不传下层包(IP包、以太网包)

  - Any information at the HTTP layer is wrapped in a data link layer packet, and eventually wrapped in a physical layer packet
  - One cannot send a high layer packet without also sending a packet of every layer below
  - When a packet enters a network, it is in a data link layer format and travels as wired and/or wireless signals

Header   Body

HTTP Packet —— HTTP包

TCP | HTTP Packet —— TCP包

IP | TCP | HTTP Packet —— IP包

Enet | IP | TCP | HTTP Packet —— 以太网包

What is actually sent over the network hardware?

- Bit string of 0's and 1's
  任何数据包最终在物理层作为比特传递，即一串0或1信号

- Any packet is eventually encapsulated as one or more physical layer packets, which travel as wired or wireless signals

  - A physical layer packet is sent through electrical cables, electromagnetic waveforms or optical fibers, in a bit string of 0's and 1's

  - A 0 may be represented as a LOW voltage pulse or a LIGHTOFF state, while a 1 may be represented as a HIGH voltage pulse or a LIGHTON state

Do all packets travel through the same physical path?

- A message is sent from host A to host B
  - Do all packets of the message travel through the same physical path from host A to host B?
  - 从**A**到**B**的一条消息的数据包必然通过同一条通路吗？
  - Not necessarily. Internet has built-in redundancy 不一定。互联网有冗余通路
    - Possible physical paths for a 99-packet message from A to B
      - 1st packet of the message travels along the physical path **A-T-X-Y-W-B**
      - 49th packet traverses path **A-T-U-V-W-B**
        - Arriving at B before 1st packet
      - 99th packet traverses path **A-T-X-Z-Y-W-B**
    - Complete message is reassembled from the packets by their numbers