中国科学技术大学

The University of Science and Technology of China

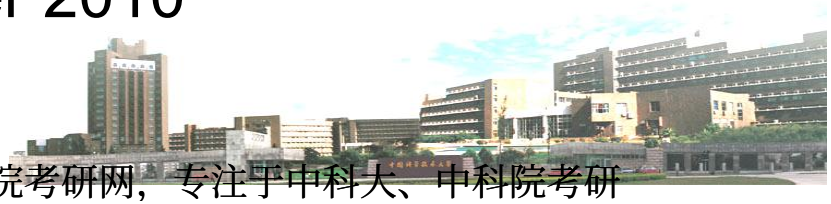
COMPUTER NETWORKS

Chapter 01

Introduction (1)

Shoubao Yang
School of Computer Science and Technology
E-mail: syang@ustc.edu.cn
http://202.38.64.11/~syang

September 2010



About the course (1)

- Purpose of this course
  - Let students learn and get familiar with computer networks basic concepts and aspects; thoroughly understand network architectures, primary ideas of network engineering, as well as network applications; build up ground work for further study and R&D in computer networks field.

About the course (2)

- Contents of the course

Will systematically study computer networks principles, ISO/OSI reference model (seven layers), especially the hybrid model of five layers, and TCP/IP model.

ISO/OSI-rm	*Hybrid*
Application	application
Presentation	
Session	
Transport	transport
Network	network
Data link	data link
Physical	physical

About the course (3)

- **Textbook**

  Andrew Tanenbaum, Computer Networks, 4thed, Prentice Hall, 2003; 潘爱民译，清华，2004

- **References**
  - Larry Peterson, Computer Networks: A Systems Approach, 2nd ed, Morgan Kaufmann, 2000
  - Computer Networking: A Top-Down Approach Featuring the Internet, 3/E, James F. Kurose, Keith W. Ross, Addison Wesley, 2004.5
  - A lots of other network books in Chinese or English

About the course (4)

- **Credits**

  60/20 teaching hours with 3.5 credits

- **Assignments**

  Homework is assigned un-periodically, maybe once a week

- **Quiz and lab works**

  un-periodically assigned in or after class

  visit network center and super computing center

- **Grading policy**

  40% for assignments, lab works and quiz;

  60% for final.

Contact Information

- **Instructor**

  Shoubao Yang 杨寿保

  3601540 syang@ustc.edu.cn

  http://202.38.64.11/~syang

- **TAs**

  王喜妹, 周文煜, 方君 3603029

  wxm11008, hndazwy, lonely58 @mail.ustc.edu.cn

- **Course Website:** 202.38.64.10/~wxm11008

Chapter 01 Introduction

Contents of This Book

Historic Review

- ENIAC: 1946, the very first computer
- ARPANET: 1969, the very first computer network for advanced research project agent of defense department, USA.
- Internet: 1986
- National Information Infrastructure: 1993
- Commercialized Internet: 1995
- NGI: 1995, Next Generation Internet
- I2: 1996, Internet2
- CNGI: Cernet2 + …

Early Stage of ENIAC

计算机网络的产生背景

- 是 20 世纪 60 年代美苏冷战时期的产物。
- 60 年代初，美国国防部领导的远景研究规划局 ARPA (Advanced Research Project Agency) 提出要研制一种生存性(survivability)很强的网络。
- 传统的电路交换(circuit switching)的电信网有一个缺点：正在通信的电路中有一个交换机或有一条链路被炸毁，则整个通信电路就要中断。
- 如要改用其他迂回电路，必须重新拨号建立连接。这将要延误一些时间。

新型网络的基本特点

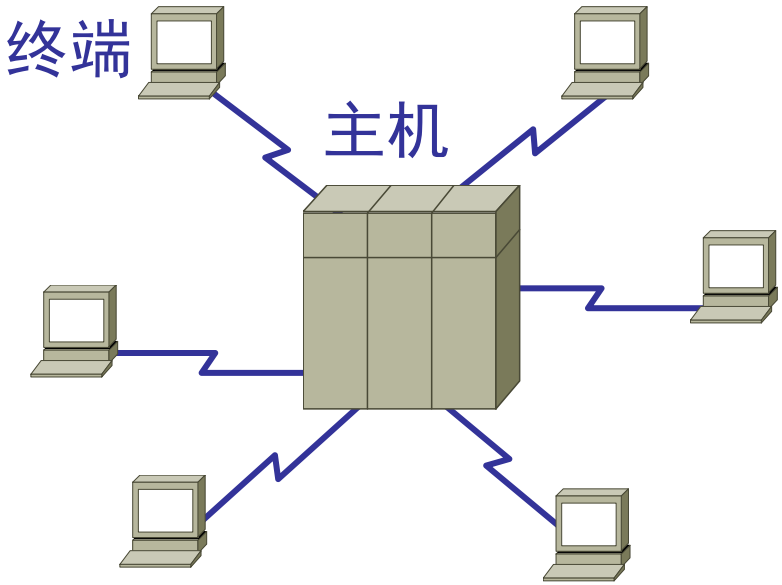
- 网络用于计算机之间的数据传送。
- 网络能够连接不同类型的计算机，不局限于单一类型的计算机。
- 所有的网络结点都同等重要，因而大大提高网络的生存性。
- 计算机在进行通信时，必须有冗余的路由。
- 网络的结构应当尽可能地简单，同时还能够非常可靠地传送数据。

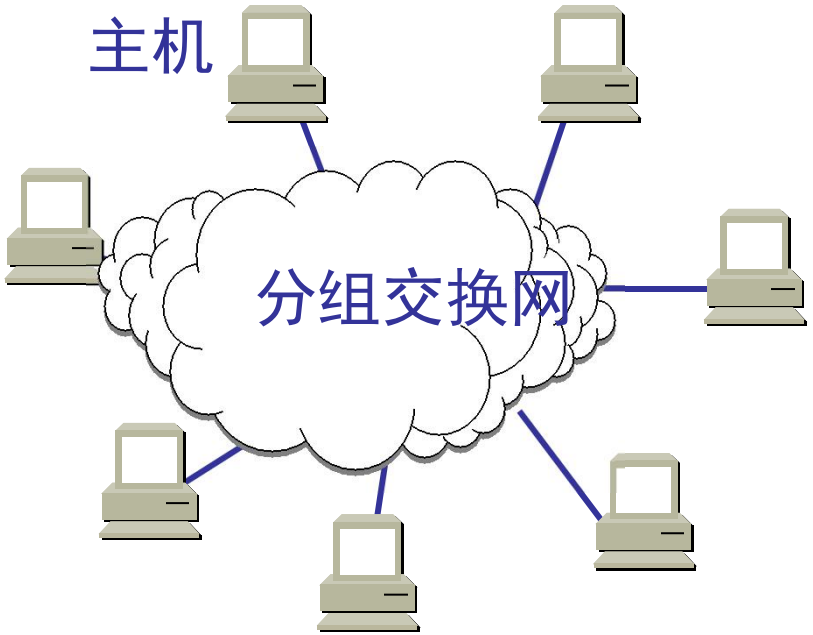
ARPANET的成功使计算机网络的概念发生根本变化

- 早期的面向终端的计算机网络是以单个主机为中心的星形网
  - 各终端通过通信线路共享昂贵的中心主机的硬件和软件资源。
- 分组交换网则是以网络为中心，主机都处在网络的外围。
  - 用户通过分组交换网可共享连接在网络上的许多硬件和各种丰富的软件资源。

从主机为中心到以网络为中心

以主机为中心 → 以分组交换网为中心





计算机网络在我国的发展

(1) 中国公用计算机互联网 CHINANET
(2) 中国教育和科研计算机网 CERNET
(3) 中国科学技术网 CSTNET
(4) 中国联通互联网 UNINET
(5) 中国网通公用互联网 CNCNET
(6) 中国国际经济贸易互联网 CIETNET
(7) 中国移动互联网 CMNET
(8) 中国长城互联网 CGWNET
(9) 中国卫星集团互联网 CSNET

Why Networking?

- The integration of computers and communication techniques:

  (1) communication network is the infrastructure for computer network;

  (2) computer progress facilitates telecommunication

  Computer Network = computer + communication

- Why networking?

  The needs for efficiently collecting, storing, processing, distributing and managing information

  The needs for changing manner of using computers

  The needs for changing manner of using networks

What is a computer network?

- **Tanenbaum's definition**

  Computer network - Interconnected collection of autonomous computers

- **Our definition**

  A system that interconnects multiple autonomous computers in different locations with communication equipment, trunks, and communication software(OS, protocols, etc.), for resource sharing, is so called computer network.

Computer Network vs Distributed System what is the difference?

- Key distinction is that in a distributed system, the existence of multiple autonomous computers is transparent to users. It looks like a virtual machine with uni-processor.
- With a network, users must explicitly log on to one machine, explicitly submit jobs remotely, explicitly move files around and handle network personally.

Important Conclusion

- In effect, a distributed system is a software system build on top of network.
- The software gives it a high degree of cohesiveness and transparency.
- Thus the distinction between a network and a distributed system lies with the software (especially the operating system) rather than with the hardware.

1.1 Use of Computer Networks

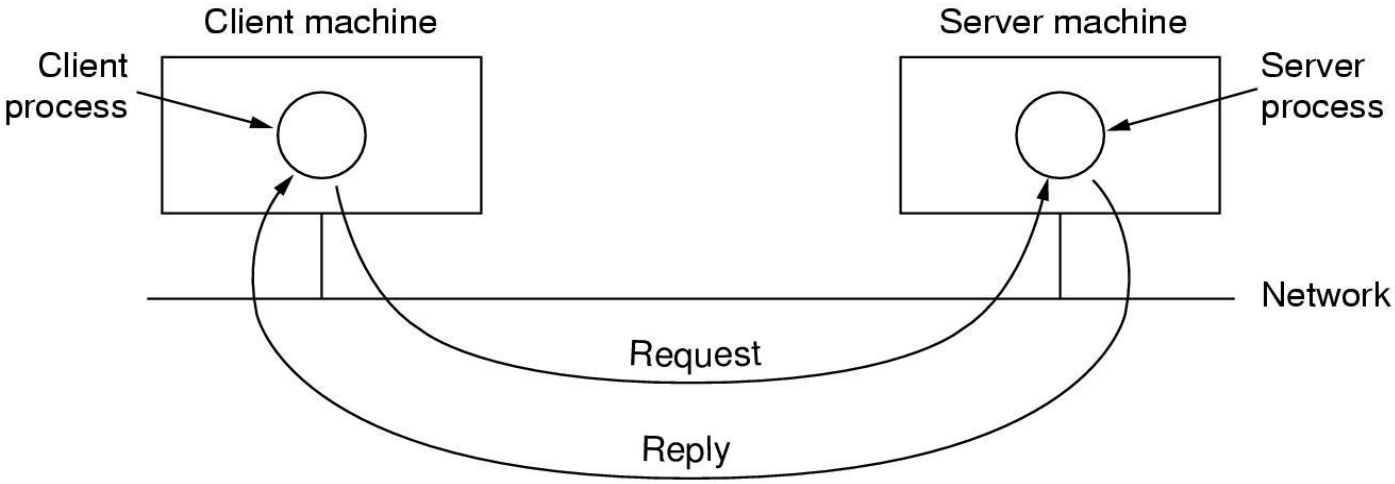
1.1.1 Business applications

Resource sharing, information sharing

High reliability

Saving money – client/server model

Scalability



Business Applications of Networks

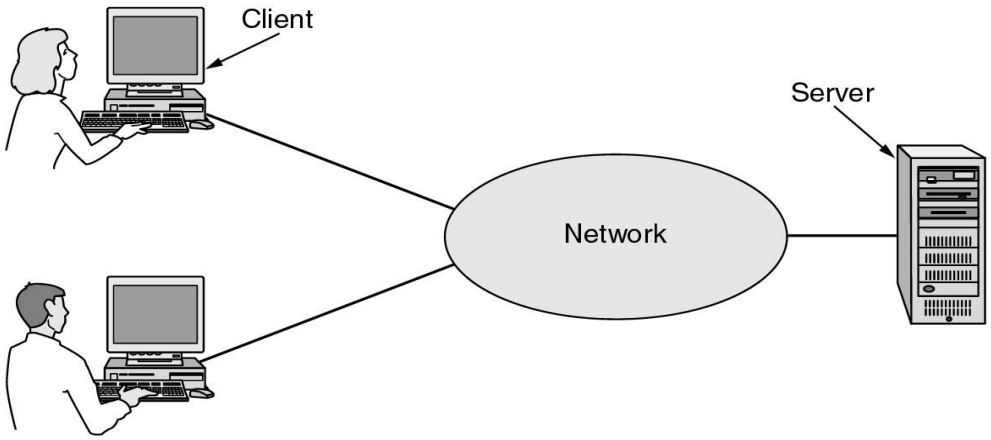


Fig. 1-1 A network with two clients and one server.

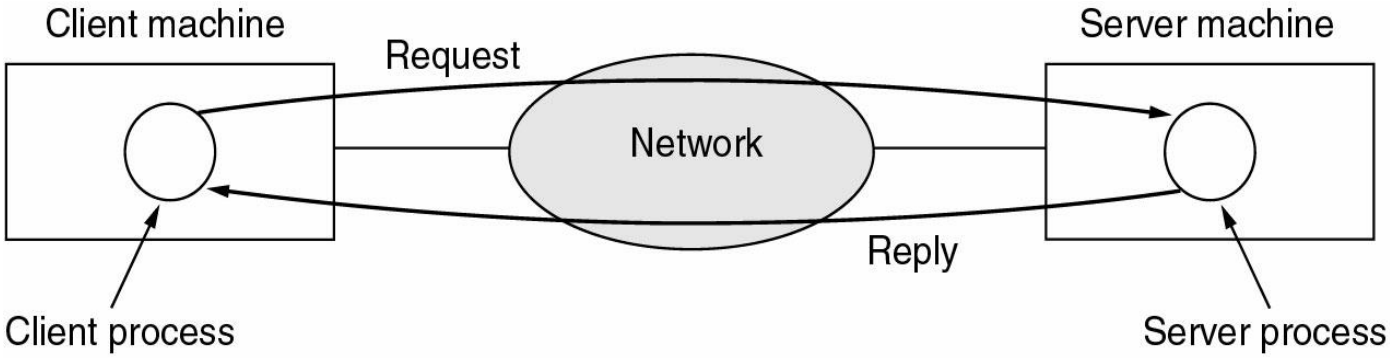
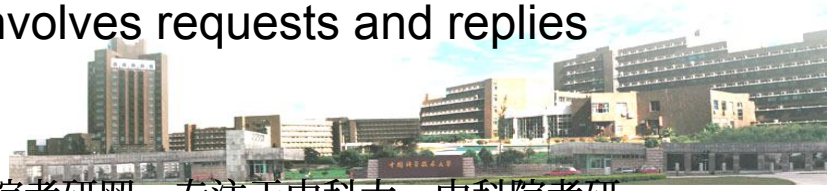


Fig. 1-2 The client-server model involves requests and replies



1.1.2 Home Network Applications

- Access to remote information
- Person-to-person communication
- Interactive entertainment
- Electronic commerce

Tag	Full name	Example
B2C	Business-to-consumer	Ordering books on-line
B2B	Business-to-business	Car manufacturer ordering tires from supplier
G2C	Government-to-consumer	Government distributing tax forms electronically
C2C	Consumer-to-consumer	Auctioning second-hand products on line
P2P	Peer-to-peer	File sharing

Fig. 1-4 Some forms of e-commerce

Home Network Applications (2)

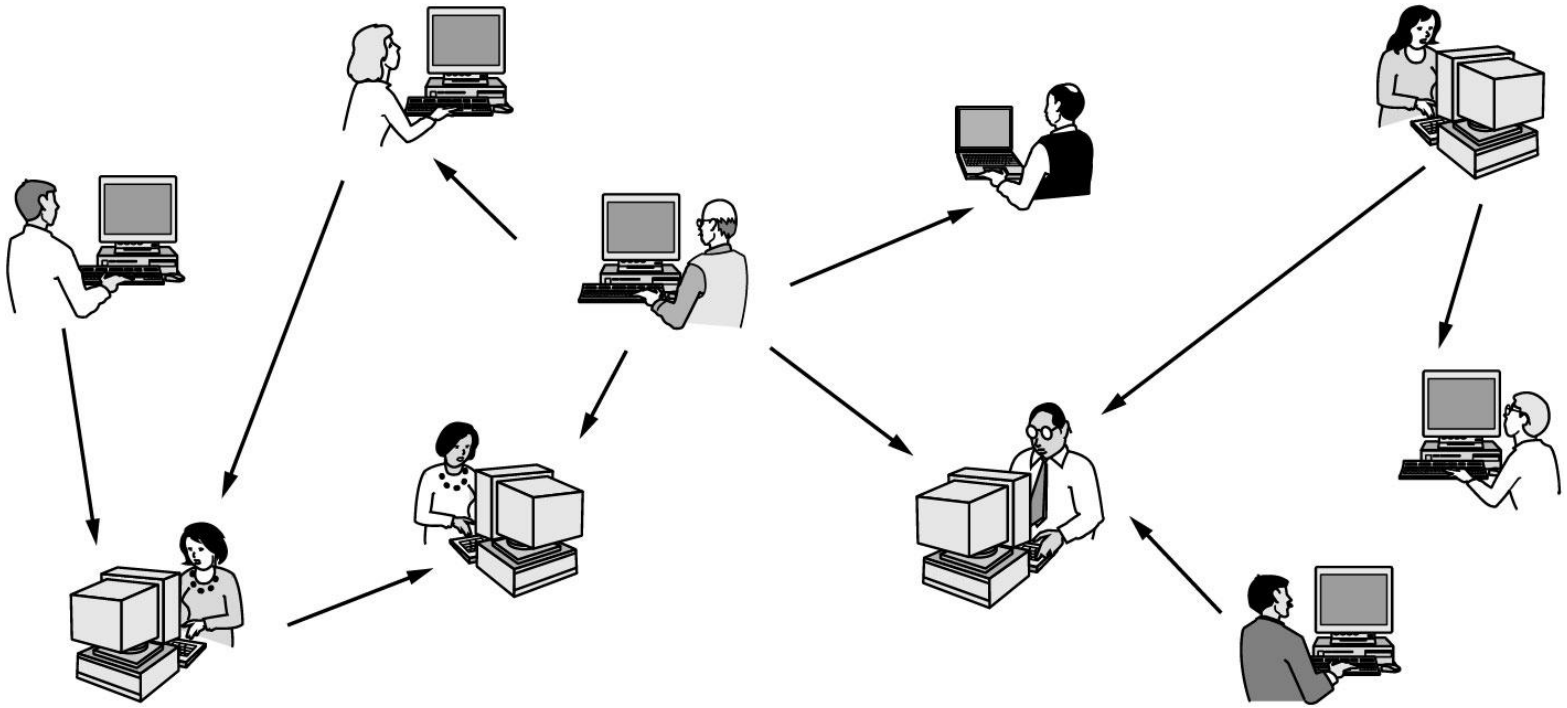


Fig. 1-3 In a peer-to-peer system there are no fixed clients and servers, such as Napster, and PPlive, Skype…

两种通信方式

在网络边缘的端系统中运行的程序之间的通信方式通常可划分为两大类：

- 客户服务器方式（C/S 方式）

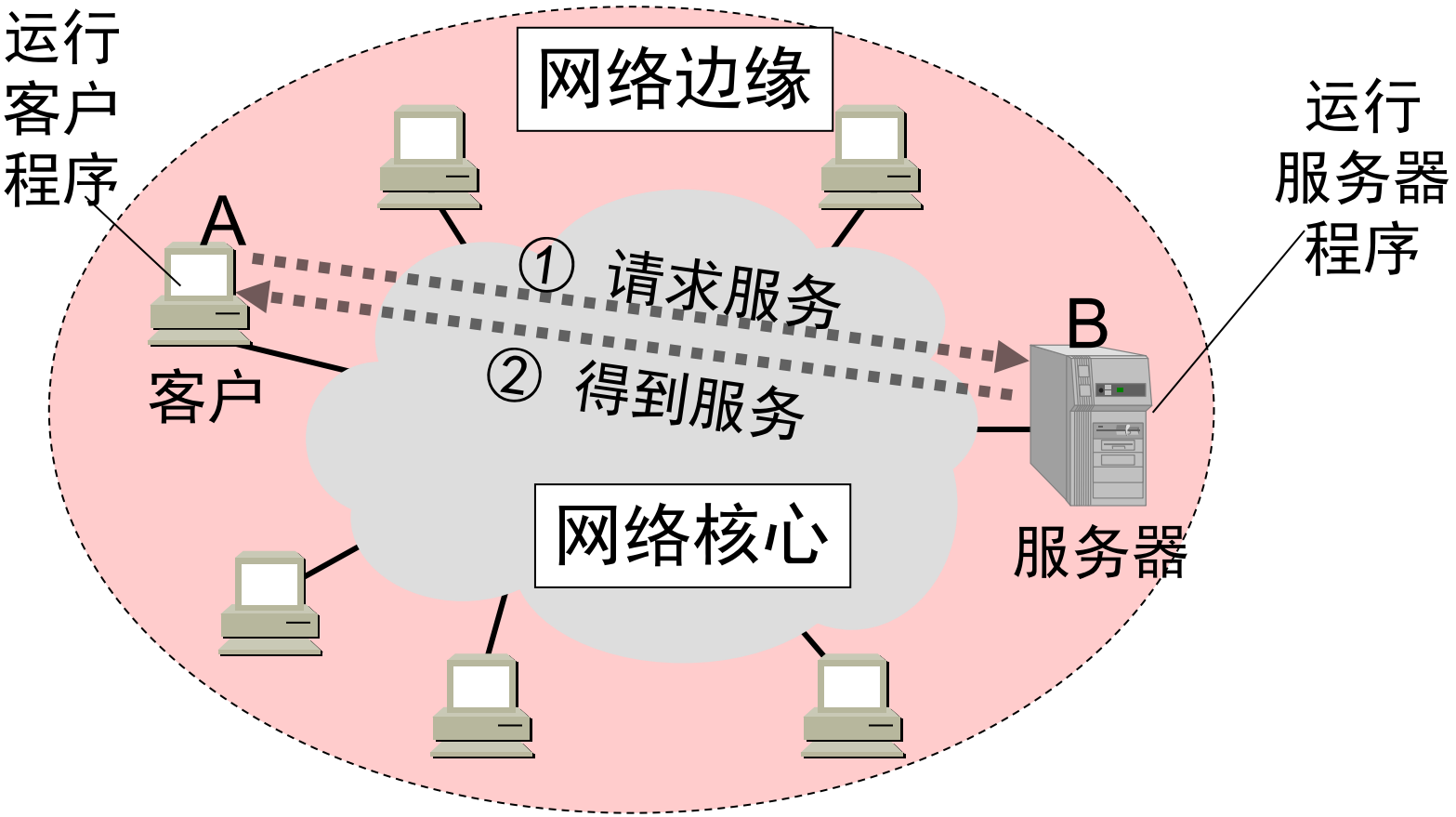
  即Client/Server方式

- 对等方式（P2P 方式）

  即 Peer-to-Peer方式

1. 客户服务器方式

- 客户(client)和服务器(server)都是指通信中所涉及的两个应用进程。
- 客户服务器方式所描述的是进程之间服务和被服务的关系。
- 客户是服务的请求方，服务器是服务的提供方。



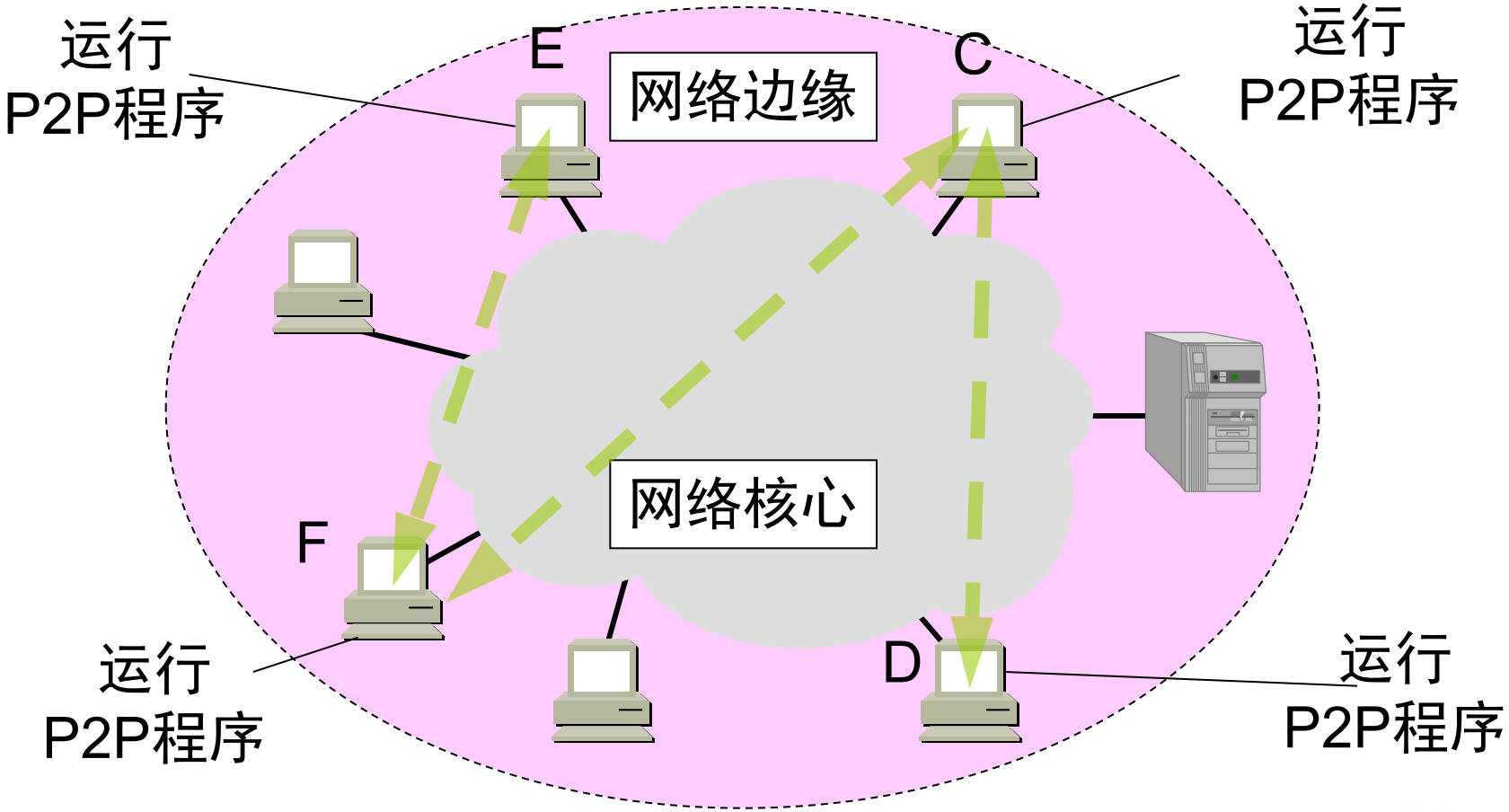
客户 A 向服务器 B 发出请求服务，而服务器 B 向客户 A 提供服务。

客户/服务器软件的特点

- 客户软件的特点
  - 被用户调用后运行，在打算通信时主动向远地服务器发起通信（请求服务）。因此，客户程序必须知道服务器程序的地址。
  - 不需要特殊的硬件和很复杂的操作系统。
- 服务器软件的特点
  - 一种专门用来提供某种服务的程序，可同时处理多个远地或本地客户的请求。
  - 系统启动后即自动调用并一直不断地运行着，被动地等待并接受来自各地的客户的通信请求。因此，服务器程序不需要知道客户程序的地址。
  - 一般需要强大的硬件和高级的操作系统支持。

2. 对等连接方式

- 对等连接(peer-to-peer，简写为 P2P)是指两个主机在通信时并不区分哪一个是服务请求方还是服务提供方。
- 只要两个主机都运行了对等连接软件(P2P 软件)，它们就可以进行平等的、对等连接通信。
- 双方都可以下载对方已经存储在硬盘中的共享文档。
- 对等连接方式从本质上看仍然是使用客户服务器方式，只是对等连接中的每一个主机既是客户又同时是服务器。
- 例如主机C请求D的服务时，C是客户，D是服务器。但如果C又同时向F提供服务，那么C又同时起着服务器的作用。



1.1.3 Mobile users

Wireless LAN

Fixed wireless and mobile wireless

Wireless	Mobile	Applications
No	No	Desktop computers in offices
No	Yes	A notebook computer used in a hotel room
Yes	No	Networks in older, unwired buildings
Yes	Yes	Portable office; PDA for store inventory

1.1.4 Social Issues

social, ethical, political problems,

such as politics, religions, sex, …

Computer Networks Overview (1)

- **Hardware:** Talk just a bit about how you can configure a bunch of computers into a network:
  - Local Area Networks (LAN)
  - Metropolitan Area Networks (MAN)
  - Wide Area Networks (WAN)
  - Internetworks
- **Software:** This is what actually makes computer networks – not the hardware!

Computer Networks Overview(2)

- **Protocols:** *how* two communicating parties exchange information.
- **Services:** *what* a network offers to parties that want to communicate.
- **Interfaces:** *how* a client can make use of network services, i.e. how the services can be accessed.

Reference models: Describe how the OSI and Internet networks are organized.

1.2 Network Hardware

- **Two types of transmission technology: broadcast links and point-to-point links**
- **Broadcast network:** a single communication channel is shared by all computers, that is, sending a packet implies that all others receive it.
- **Multicasting:** transmission to a subnet of users
- **Point-to-point network:** Computers are connected in pairs, that is, sending a packet goes strictly from the sender to the receiver, possibly having to visit intermediate machines (*routing*).
- **Unicasting:** with only one sender and one receiver

Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

Fig. 1-6 Classification of interconnected processors by scale

1.2.1 Local Area Networks (LAN)

- Apart from scale, LANs distinguish themselves from other networks by (generally) using **broadcast** technology, and having simple **topologies**:

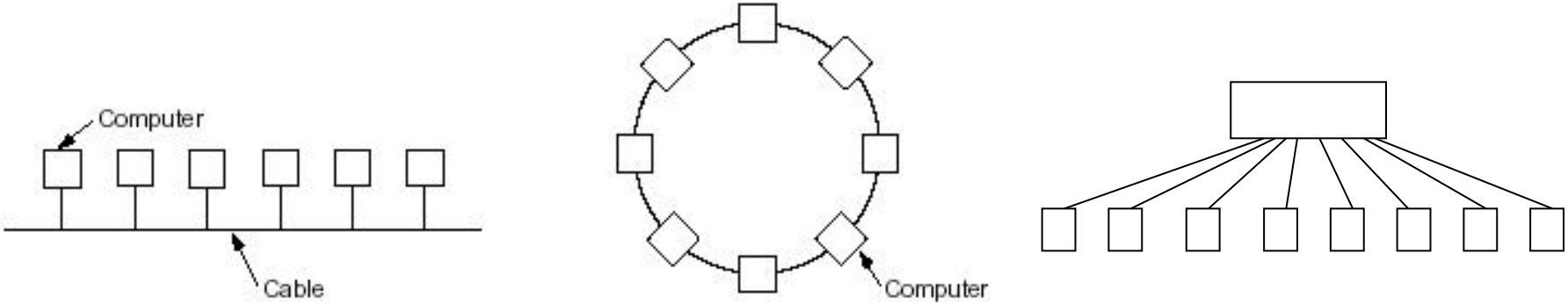


Fig. 1-7 Two broadcast networks. (a) Bus. (b) Ring.

The Star

Typical Topologies of Local Area Networks (LAN)

- **Type (a) (Bus-based):** All computers are connected to the same wire. When one of them starts sending, the signal is propagated to all others. If two of them start sending at the same time, packets collide and rubbish is the result.
- **Type (b) (Token-based):** a token (which is just a small packet) *continuously* circulates along the ring. A sending computer:
  (1) waits until the token passes and removes it
  (2) sends its packet along the ring
  (3) waits until the packet returns
  (4) reinserts the token
- **The star** is the most commonly used today

1.2.2 Metropolitan Area Networks (MAN)

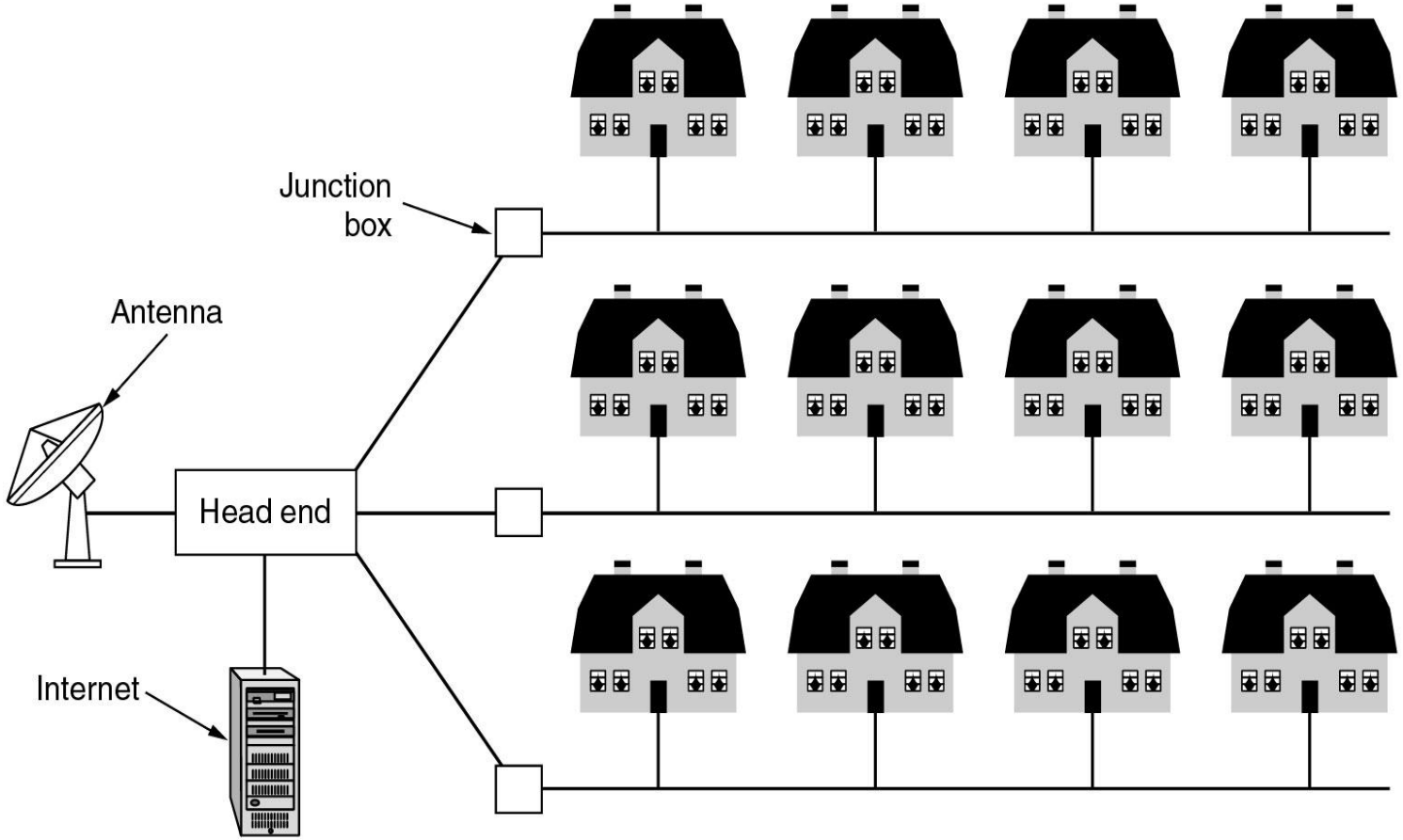


Fig. 1-8. A metropolitan area network based on cable TV.

1.2.3 Wide Area Networks (WAN)

- **Note:** LANs and MANs generally don't have any **switching elements**: the wire does all the work. This makes them extremely efficient, although harder to scale. Here's where WANs come in.

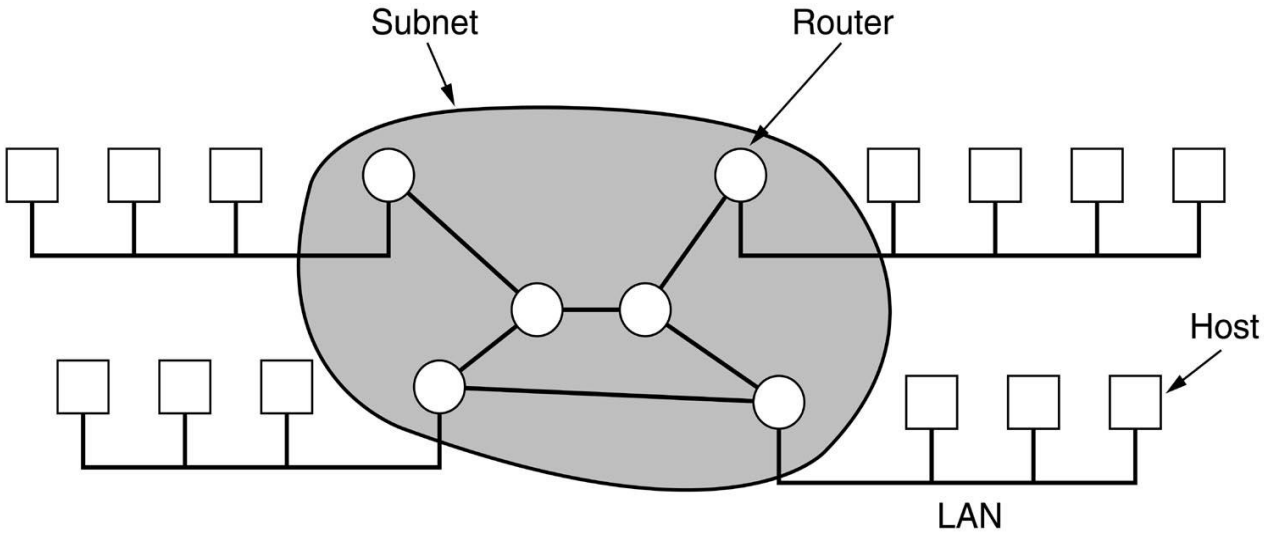
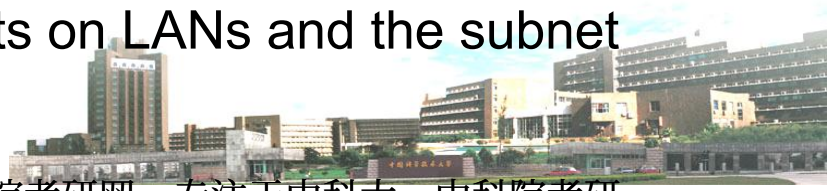


Fig. 1-9. Relation between hosts on LANs and the subnet



Wide Area Networks (2)

- In a WAN, **hosts** are connected to a **subnet**, which in turn consists of **routers** (switching elements) and **trunks**.
- Routers generally adhere to a **store-and-forward** principle: incoming packets are first buffered (stored), the router takes a decision on where the packet has to go, and forwards the packet across the selected output line.

Wide Area Networks (3)

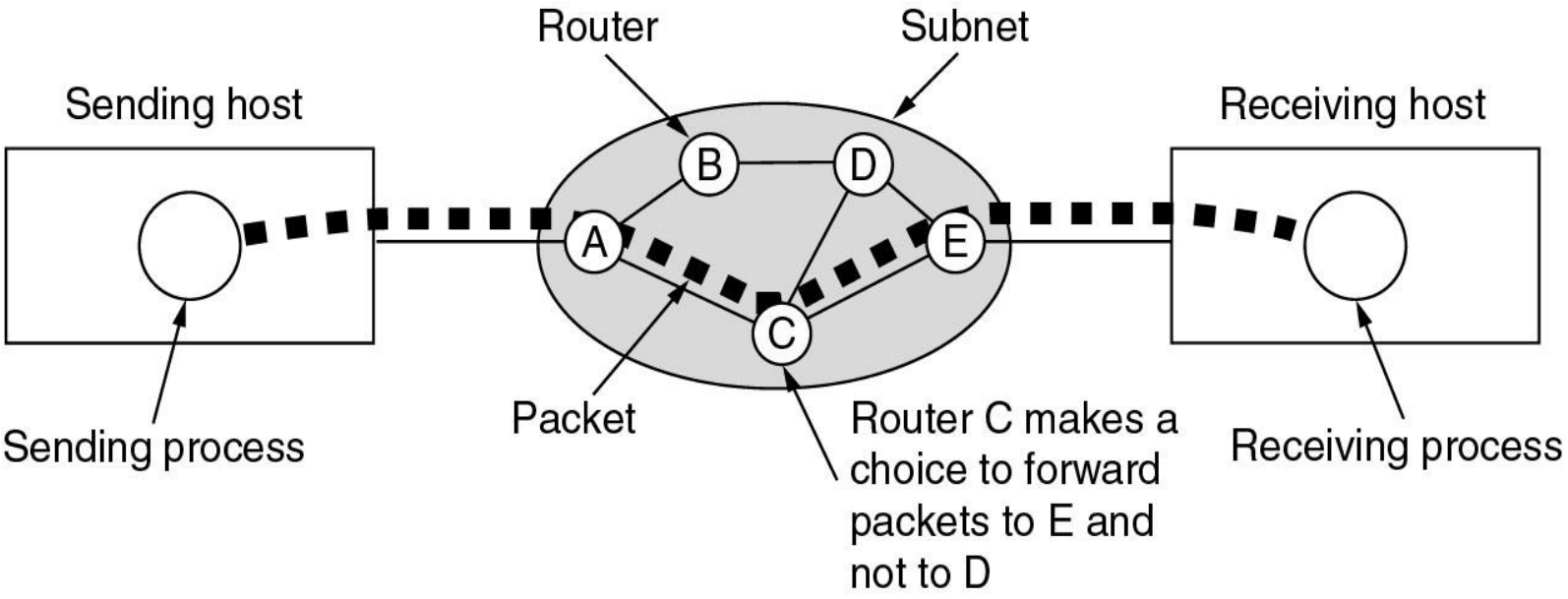


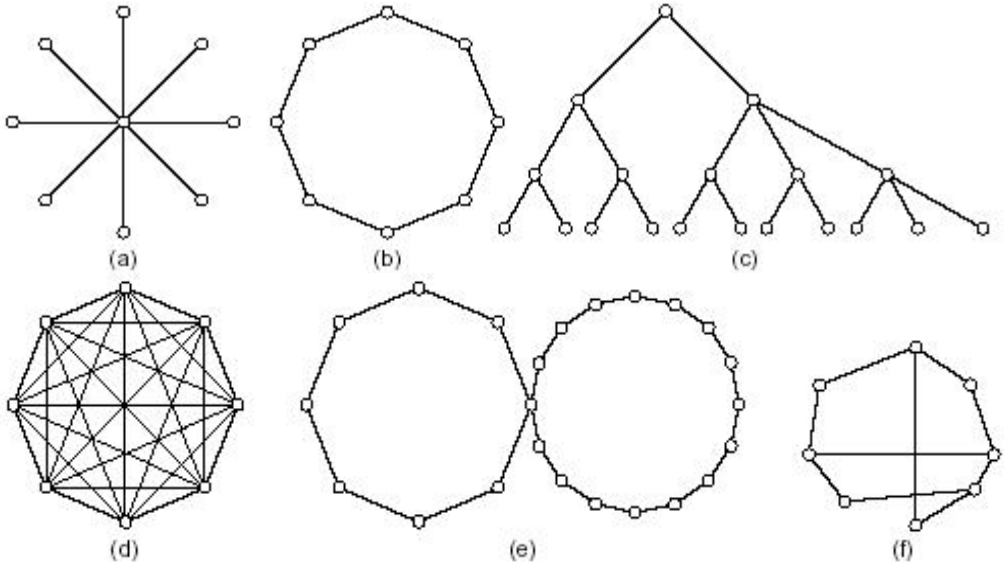
Fig. 1-10. A stream of packets from sender to receiver.

WAN Consists of Two Subnets

- **Resource subnet**
  - Includes computers, terminals, programs, etc.
  - Responsible for information processing and storing
- **Communication subnet**
  - Includes transportation lines, switching elements(routers and switches, etc.)
  - Responsible for information delivery and distributing

WAN Topologies

- **Note:** In contrast to LANs and MANs, the organization of a WAN in terms which hosts are interconnected is important



- **Observation:** Most often you'll see arbitrary topologies; the others are used in application-specific ways (mostly star, ring, and tree)

1.2.4 Wireless Networks (1)

- Wireless networks can be divided into three main categories:
  - System interconnection
  - Wireless LANs
  - Wireless WANs
- System interconnection is all about interconnecting the components of a computer using short-range radio, such as Bluetooth with the master-slave paradigm.

Wireless Networks (2)

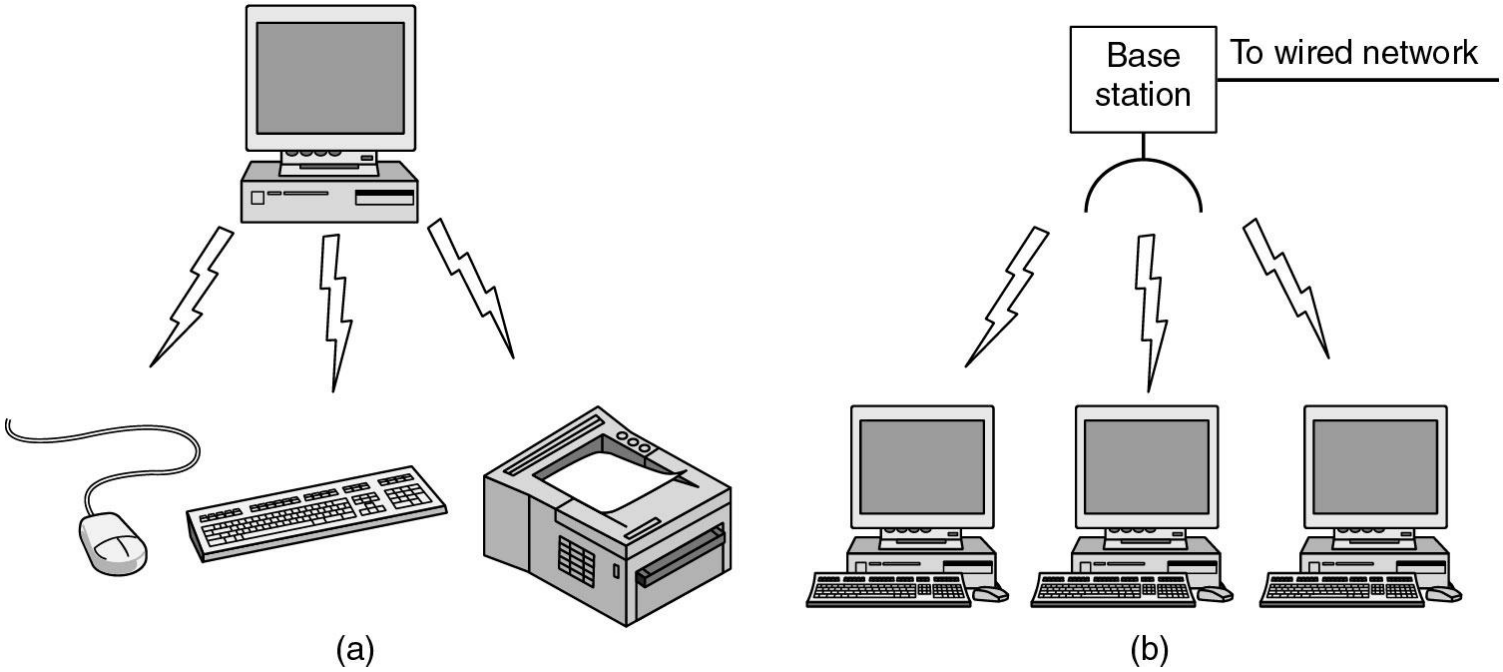


Fig. 1-11. (a) Bluetooth configuration (b) Wireless LAN

Wireless Networks (3)

- **Wireless LANs - IEEE 802.11**
  - Every computer has a radio modem and antenna with which it can communicate with other system
  - If the systems are close enough, they can communicate directly with one another in a peer-to-peer configuration.
  - The standard for wireless LANs is called IEEE 802.11

Wireless Networks (4)

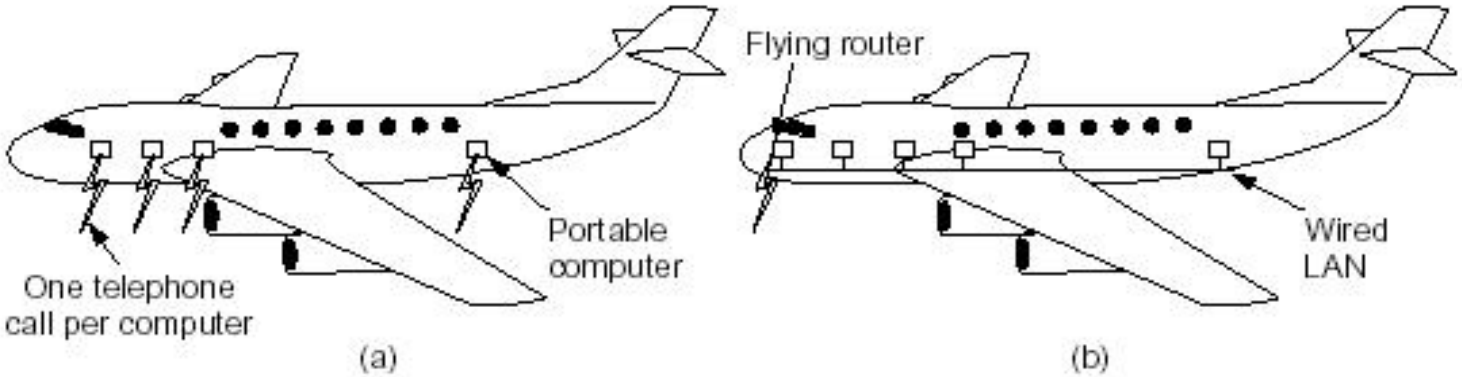
- **Wireless WANs - IEEE 802.16**
  - The radio network used for cellular telephones is an example of a low-bandwidth wireless system.
  - Three generations for cellular telephones: analog for voice only; digital for voice only; digital for both voice and data.
  - High-bandwidth wide area wireless networks focus on high-speed Internet access from homes and businesses, bypassing telephone system (called local multipoint distribution service). The standard is IEEE 802.16.

Wireless Networks (5)

- **Important:** Make distinction between wireless and mobile:

Application	W	M
Office workstations	N	N
Notebooks	N	Y
LAN in the jungle	Y	N
LAN on a ship	N	Y/N
PDA	Y	Y

- **Note:** Distinction is sometimes hard to make, especially when wireless and wired go together



1.2.5 Home Network Categories

- Computers (desktop PC, PDA, shared peripherals)
- Entertainment (TV, DVD, VCR, camera, stereo, MP3)
- Telecomm (telephone, cell phone, intercom, fax)
- Appliances (microwave, fridge, clock, furnace, air conditioner)
- Telemetry (utility meter, burglar alarm, baby cam).

1.2.6 Internetworks (1)

- **Homogeneous**
  - The assumption so far is that a network is **homogeneous**: there is hardly any variation in hardware and software. In practice, large networks can only be constructed by **interconnecting** different kinds of networks, **internet(work)**.
- **Heterogeneous**
  - hardware and software are variable, especially the protocol stacks are different

Internetworks (2)

- **Examples**
  - Connecting a collection of different kinds of LANs (bus-based to token-based) within a department.
  - Connecting LANs to each other through a WAN (think of enterprise networks for multinationals). The WAN acts as a subnet.
  - Connecting WANs to each other (the Internet).

1.3 Network Software

1.3.1. Protocol Hierarchies

- To reduce design complexity, most networks are organized as a series of layers, each layer offers certain services to its upper layer.
- Layer *n* on one machine carries on a conversation with layer *n* on another machine. The rules and conventions used in this conversation are collectively known as the layer *n* protocol.

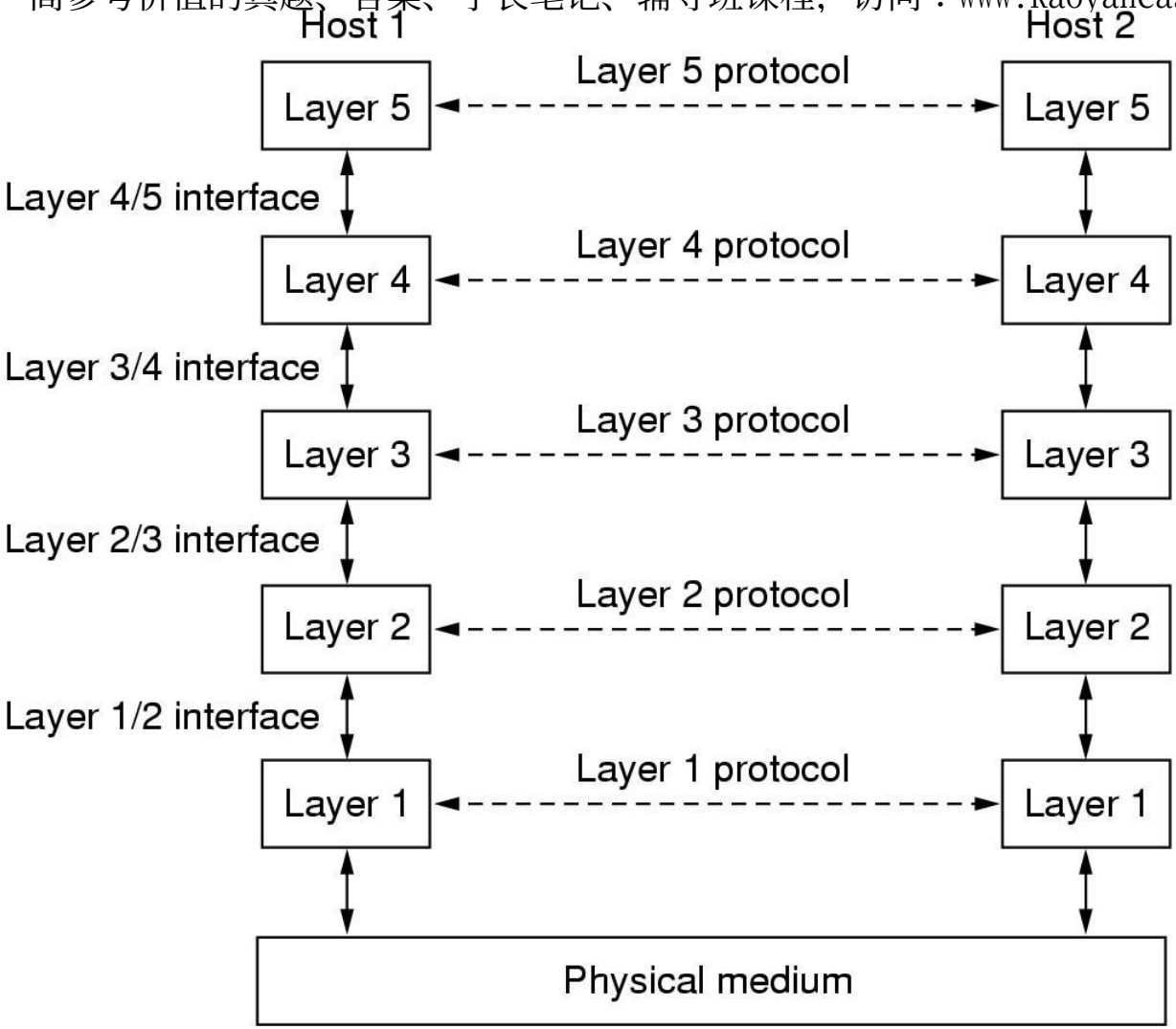
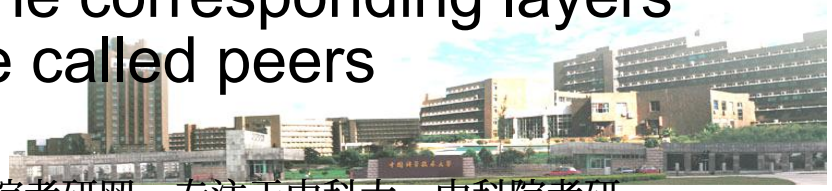


Fig.1.13 Layers, protocols, and interfaces

Some Terminologies (1)

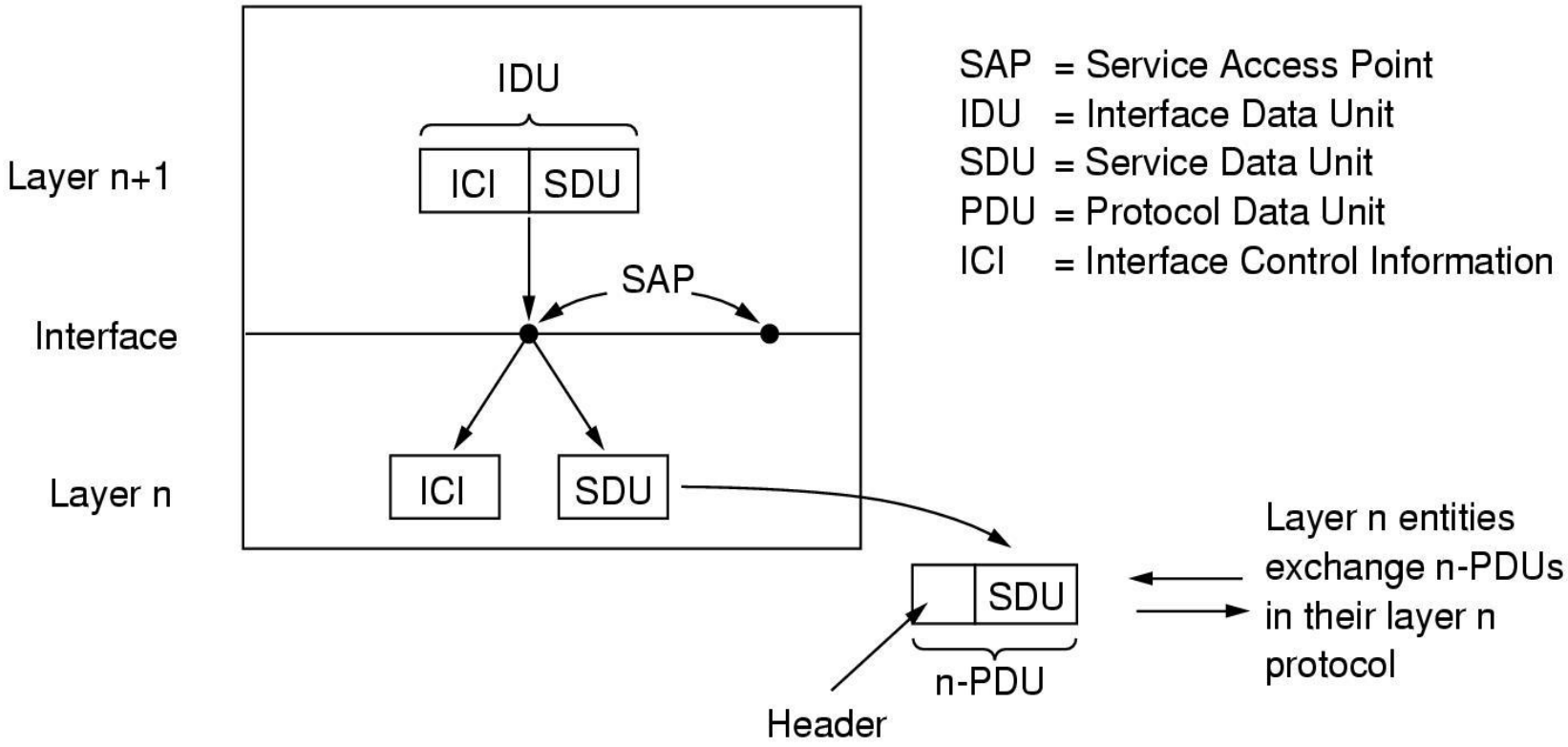
- **Layers**
  - Networks are organized as a stack of layers or levels, each one built upon the one below it
- **Protocol**
  - An agreement between the communicating parties on how communication proceed: how two communicating parties exchange information.
- **Service**
  - What a network offers to parties that want to communicate.
- **Peer**
  - The entities comprising the corresponding layers on different machines are called peers



Some Terminologies (2)

- **Interface**
  - Defines which primitive operations and services the lower layer makes available to the upper one: how the services can be accessed.
- **Network Architecture**
  - A set of layers and protocols, together with interfaces between layers, is called Network Architecture.
- **Protocol stack**
  - A list of protocols used by a certain system, one protocol per layer, is called a protocol stack.
- **Protocol hierarchies**
  - Fundamental to *all* software that makes a computer network run, is the notion of **protocol hierarchies**: structuring the services that a network must offer in terms of **layers**.

Interfaces and Services (1)



Relation between layers at an interface

Interfaces and Services (2)

- A **Service Access Point** is identified by an address, and forms the interface to a set of services.
- The **Service Data Unit** contains the data you want to send.
- The **Interface Control Information** contains info needed to send the SDU, e.g. number of bytes.
- The **Protocol Data Unit** is the data that is sent across the network, containing your SDU as well as protocol-specific data.

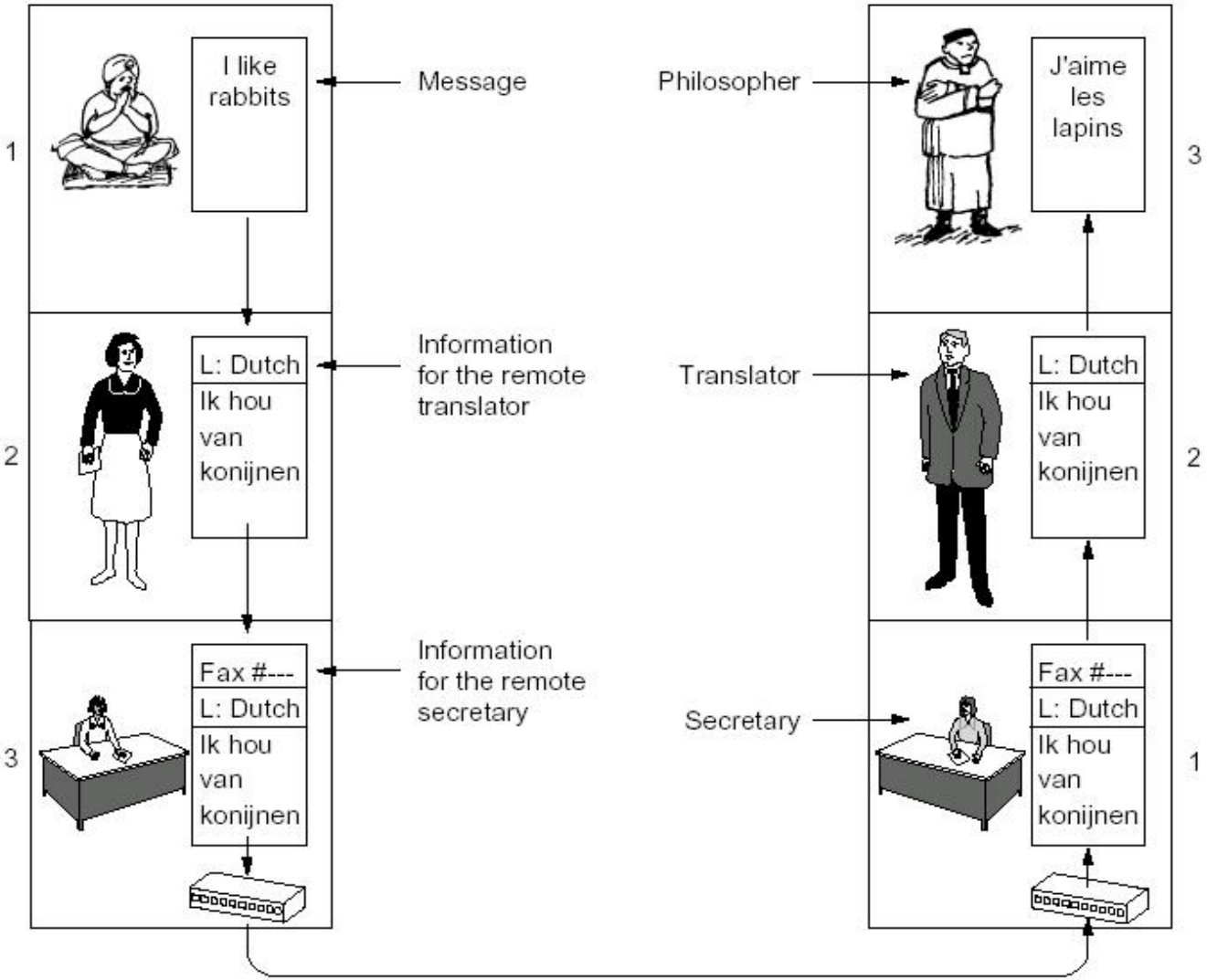


Fig.1.14 The philosopher – translator – secretary architecture

Layering: The Concepts (1)

- Two parties at different sites, but at the same level, always agree on how they will exchange information: specified in a **protocol**. Example: we all agree to speak Chinese (but use English on the sheets).
- In order for one party to send and receive information, it can only make use of the **communication services** offered by the layer directly underneath it.
- Example: The use of interpreters in negotiations between countries.

Layering: The Concepts (2)

- Services offered by a layer are always fully specified in terms of an **interface** that makes those services accessible. Example: phones have buttons that allow you to “dial” a number.
- **Protocol**: A set of rules and specifications for peer-to-peer virtual communication, it is “**horizontal**”.
- **Service**: A set of communication abilities and operations provided by lower layer to its higher layer, it is “**vertical**”.

Layering: An Example

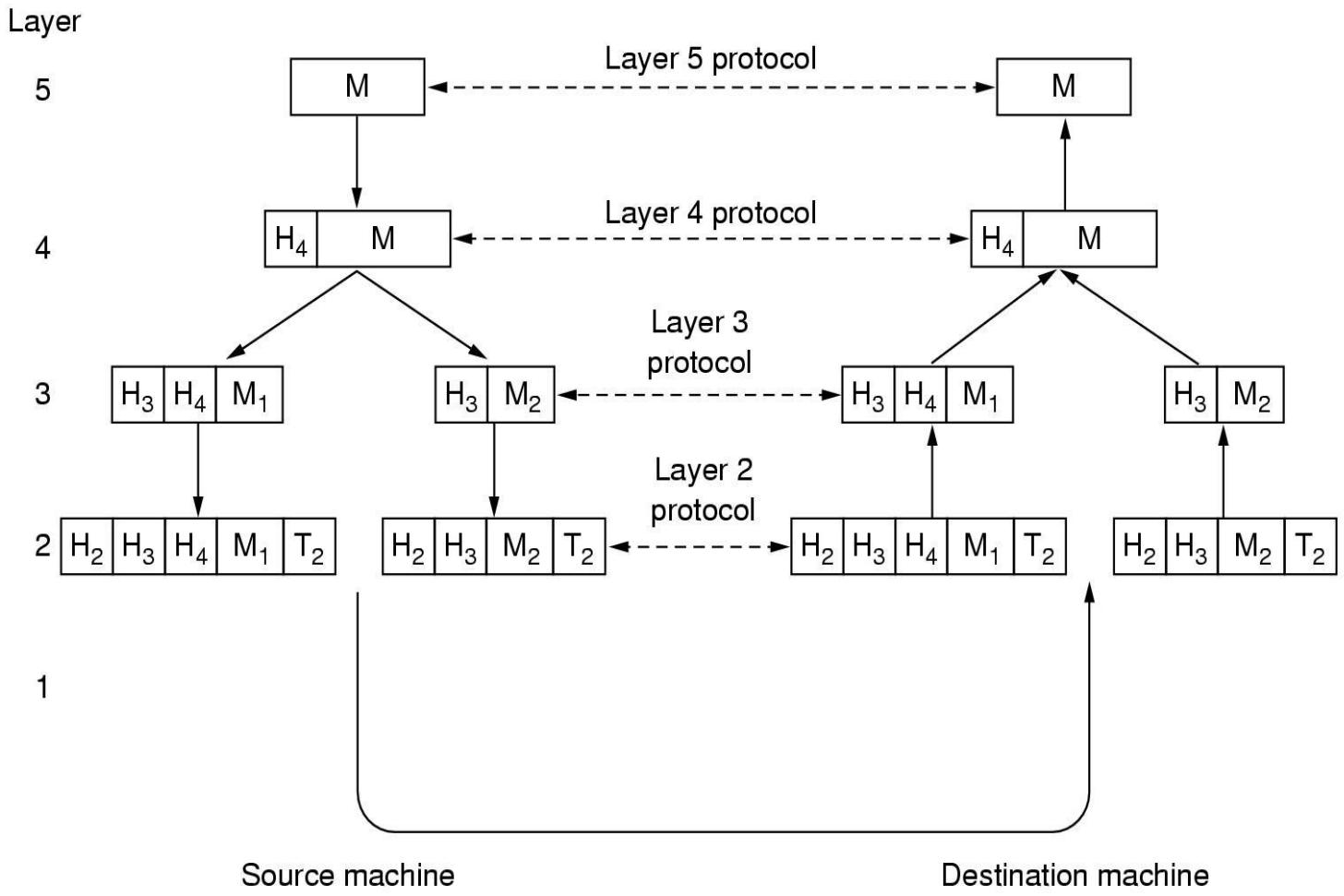


Fig.1.15 Information flow supporting virtual communication in layer 5

Payload and Fragmentation

- **Observation:** In a protocol stack, Layer k puts its entire packet as **data** into a Layer k-1 packet, that is, the k-1 packet payload; the latter may add a header and/or a trailer.
- **Note:** It may even occur that Layer k data has to be split across several Layer k-1 packets, i.e., fragmentation. At the destination, reassembly is needed to recover Layer k data.

计算机网络体系结构的形成

- 相互通信的两个计算机系统必须高度协调工作才行，而这种“协调”是相当复杂的。
- “分层”可将庞大而复杂的问题，转化为若干较小的局部问题，而这些较小的局部问题就比较易于研究和处理。

计算机网络的体系结构

- 计算机网络的体系结构(architecture)是计算机网络的各层及其协议的集合。
- 体系结构就是这个计算机网络及其部件所应完成的功能的精确定义。
- 实现(implementation)是遵循这种体系结构的前提下用何种硬件或软件完成这些功能的问题。
- 体系结构是抽象的，而实现则是具体的，是真正在运行的计算机硬件和软件。

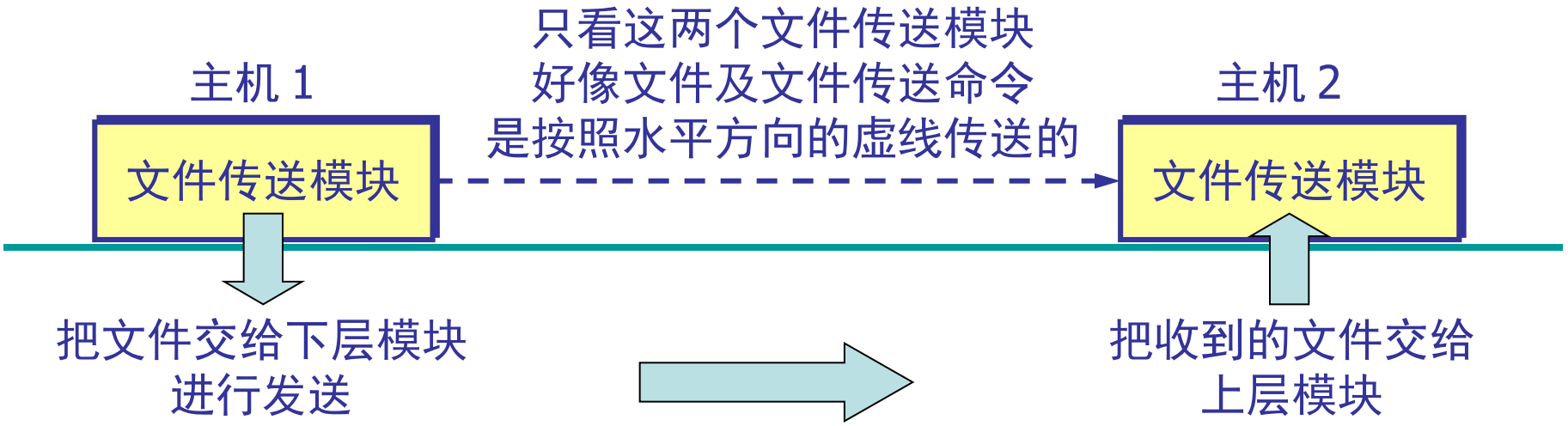
划分层次的必要性

- 计算机网络中的数据交换必须遵守事先约定好的规则。
- 这些规则明确规定了所交换的数据的格式以及有关的同步问题（同步含有时序的意思）。
- 网络协议(network protocol)，简称为协议，是为进行网络中的数据交换而建立的规则、标准或约定。
  - 语法　数据与控制信息的结构或格式 。
  - 语义　需要发出何种控制信息，完成何种动作以及做出何种响应。
  - 同步　事件实现顺序的详细说明。

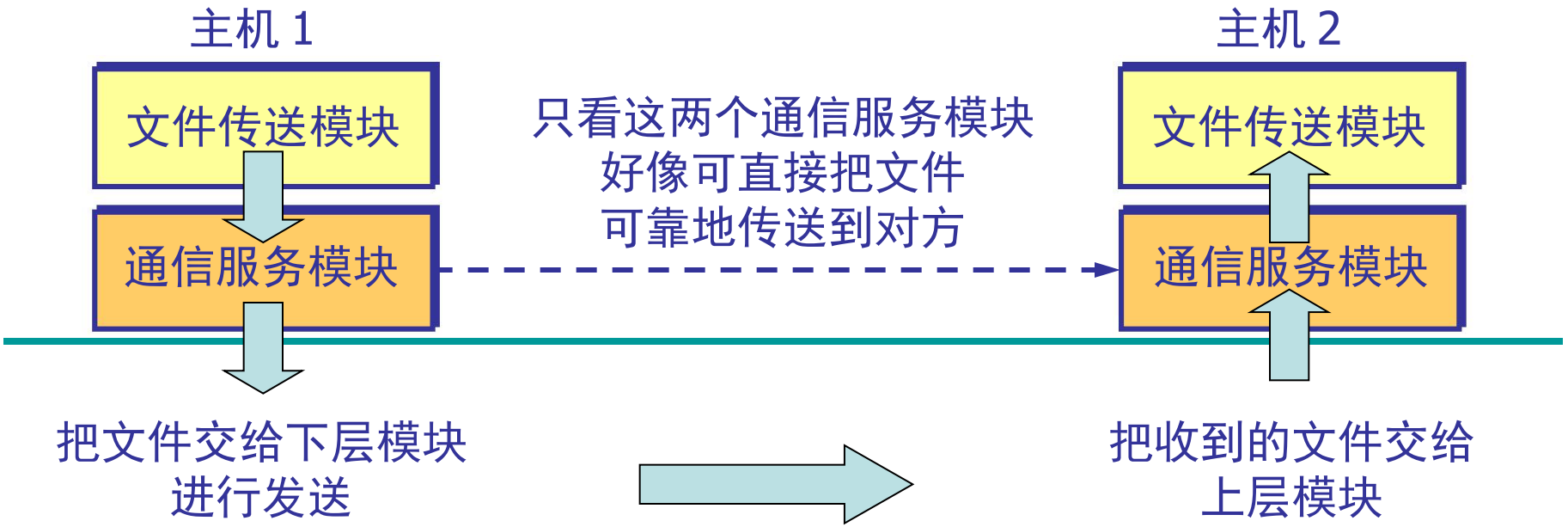
划分层次的概念举例

- 主机 1 向主机 2 通过网络发送文件
- 可以将要做的工作进行如下的划分：
  - 第一类工作与传送文件直接有关。
    - 确信对方已做好接收和存储文件的准备。
    - 双方协调好一致的文件格式。
  - 两个主机将文件传送模块作为最高的一层，剩下的工作由下面的模块负责。

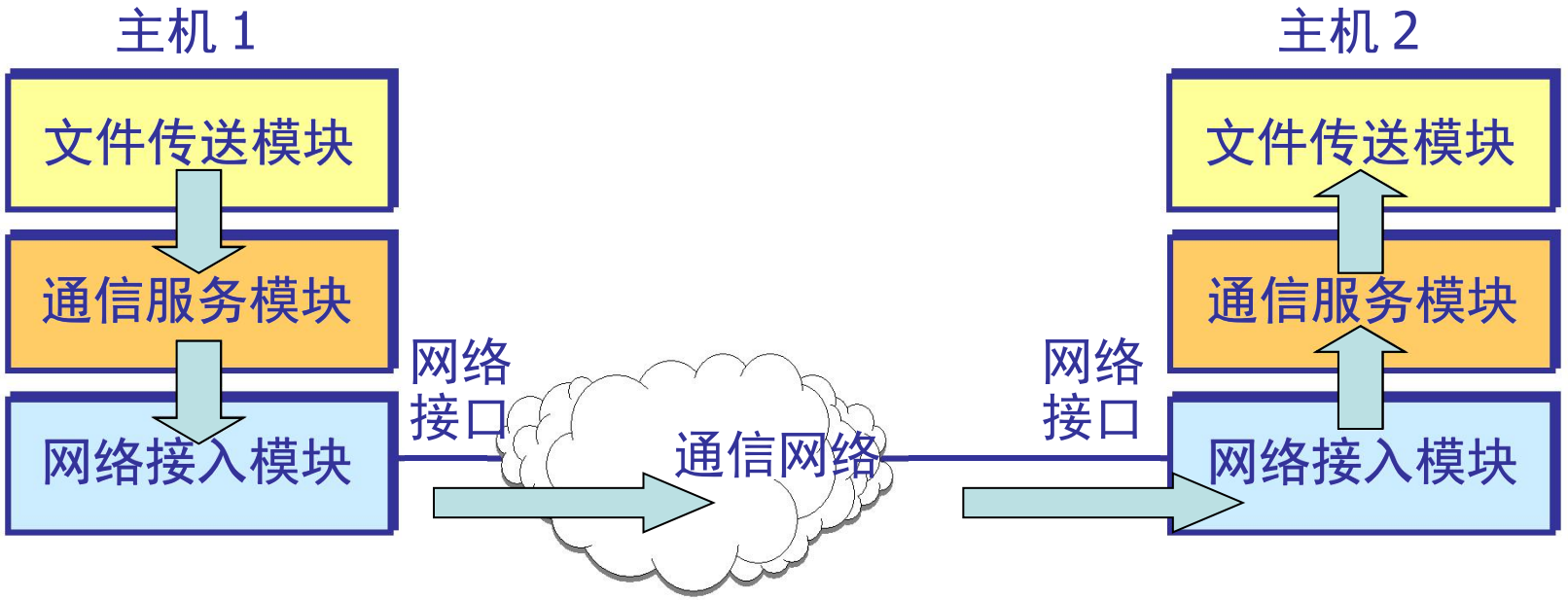
两个主机交换文件



设计一个通信服务模块



再设计一个网络接入模块



网络接入模块负责做与网络接口细节有关的工作
例如，规定传输的帧格式，帧的最大长度等。

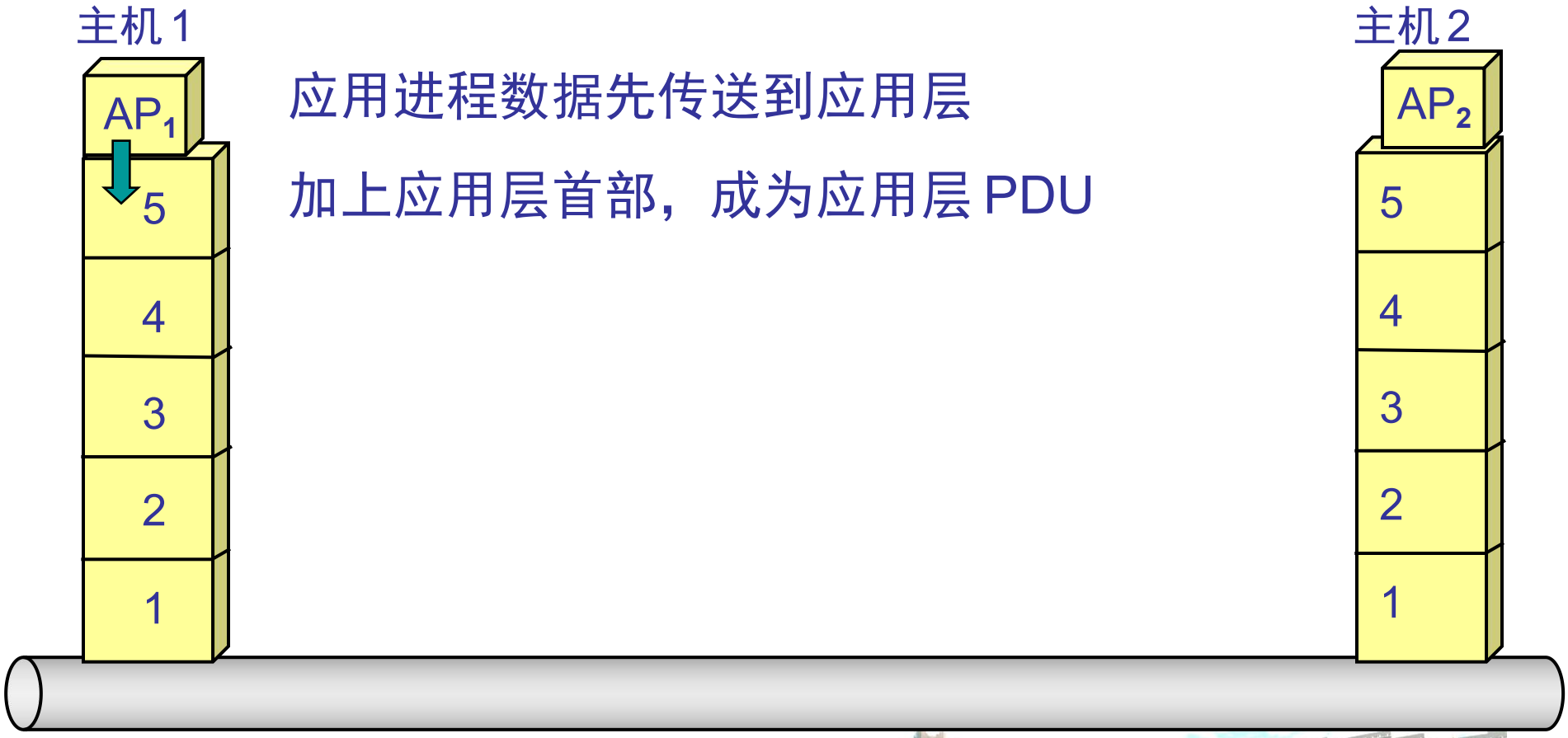
分层的好处

- 各层之间是独立的
- 灵活性好
- 结构上可分割开
- 易于实现和维护
- 能促进标准化工作
- 层数多少要适当
  - 若层数太少，就会使每一层的协议太复杂。
  - 层数太多又会在描述和综合各层功能的系统工程任务时遇到较多的困难。

五层协议的体系结构

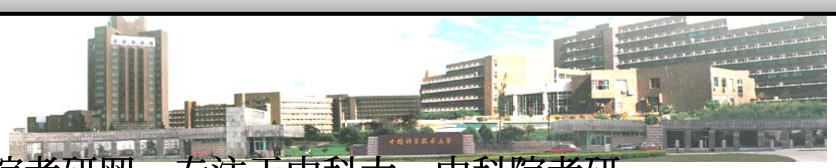
- 应用层(application layer)
- 传输层(transport layer)
- 网络层(network layer)
- 数据链路层(data link layer)
- 物理层(physical layer)

主机1向主机2发送数据

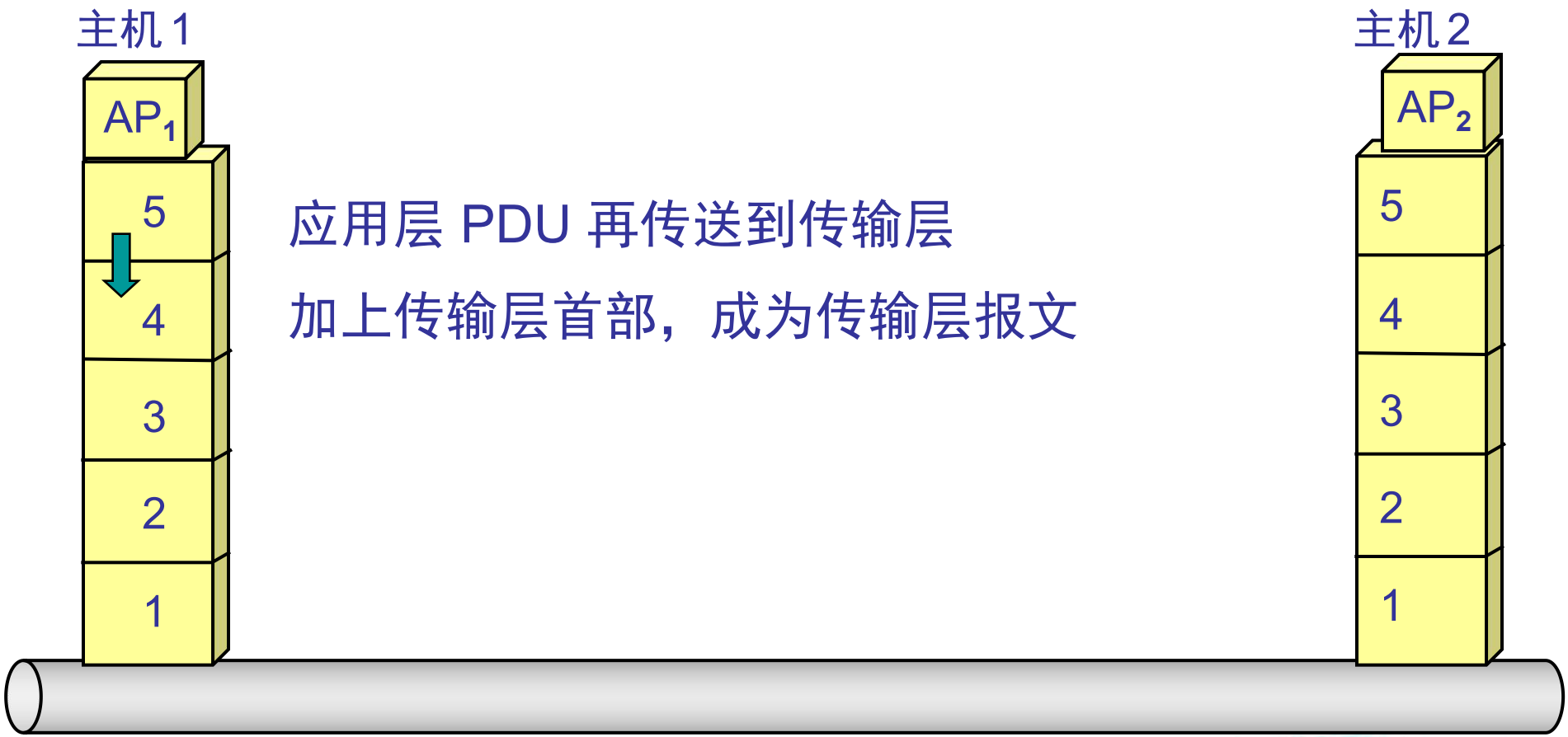


应用进程数据先传送到应用层

加上应用层首部，成为应用层PDU



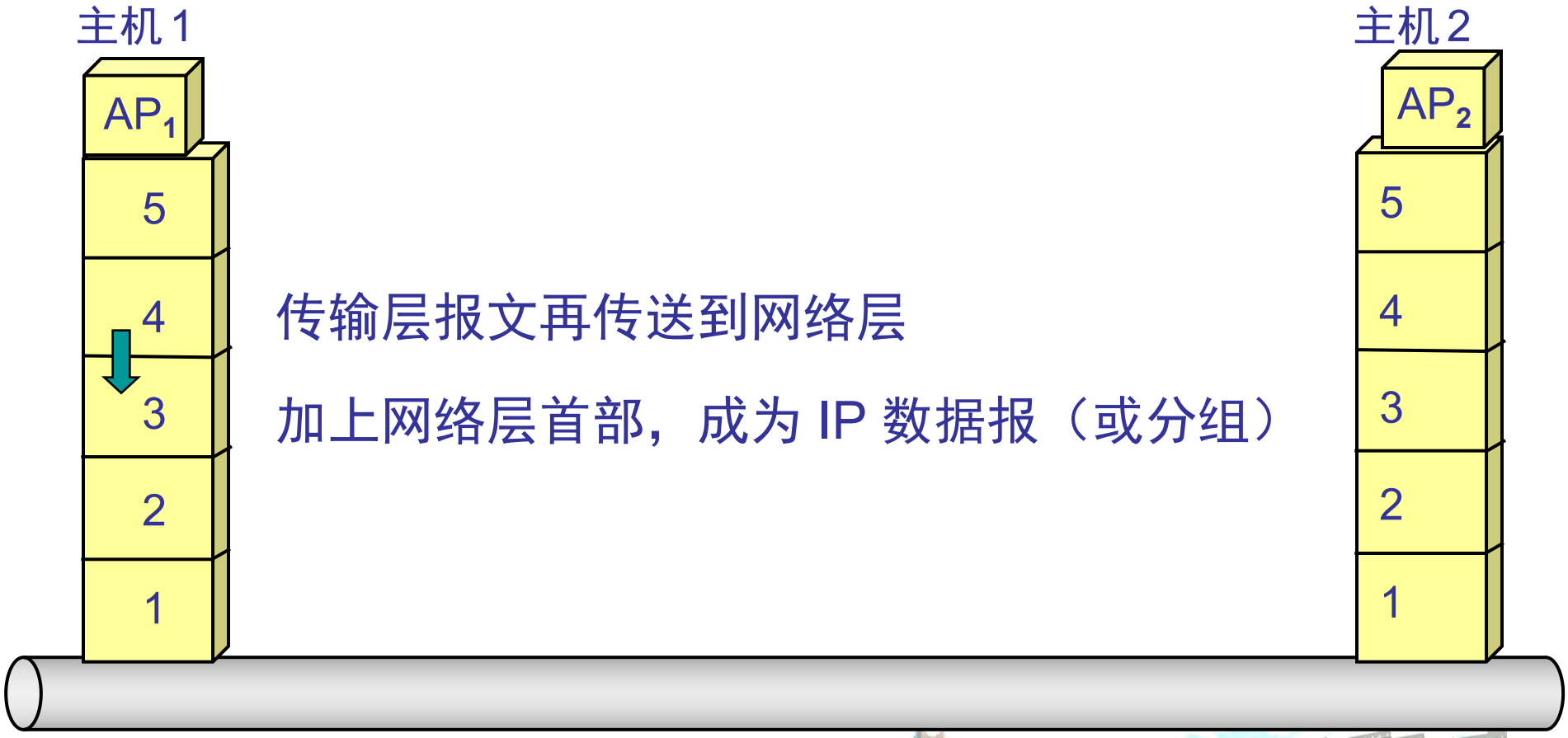
主机1向主机2发送数据

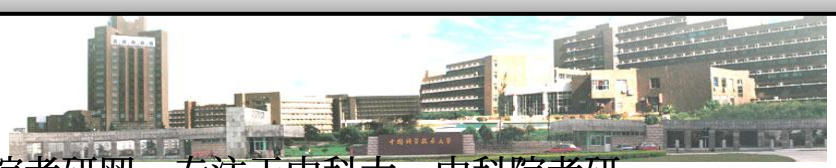


应用层 PDU 再传送到传输层

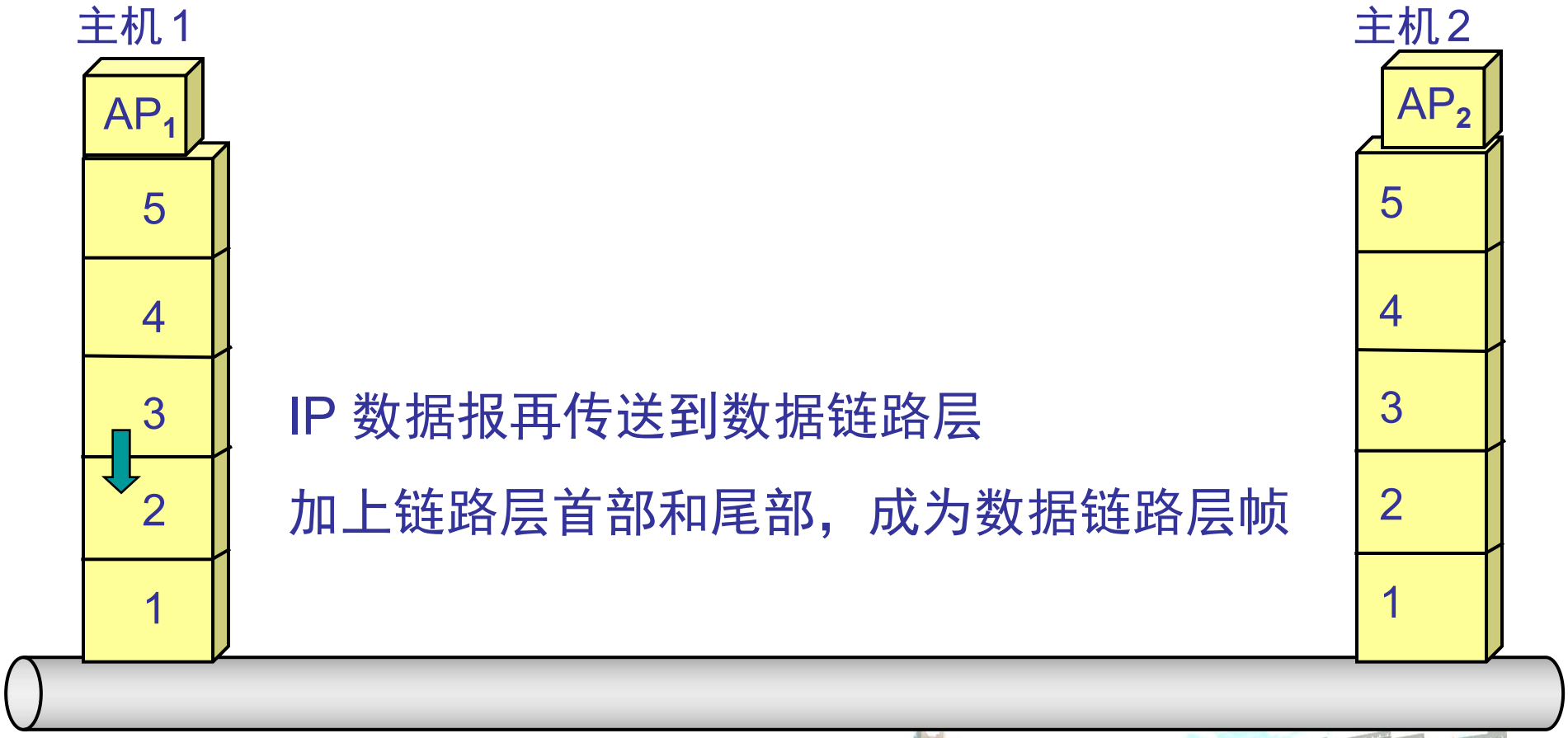
加上传输层首部，成为传输层报文

主机1向主机2发送数据





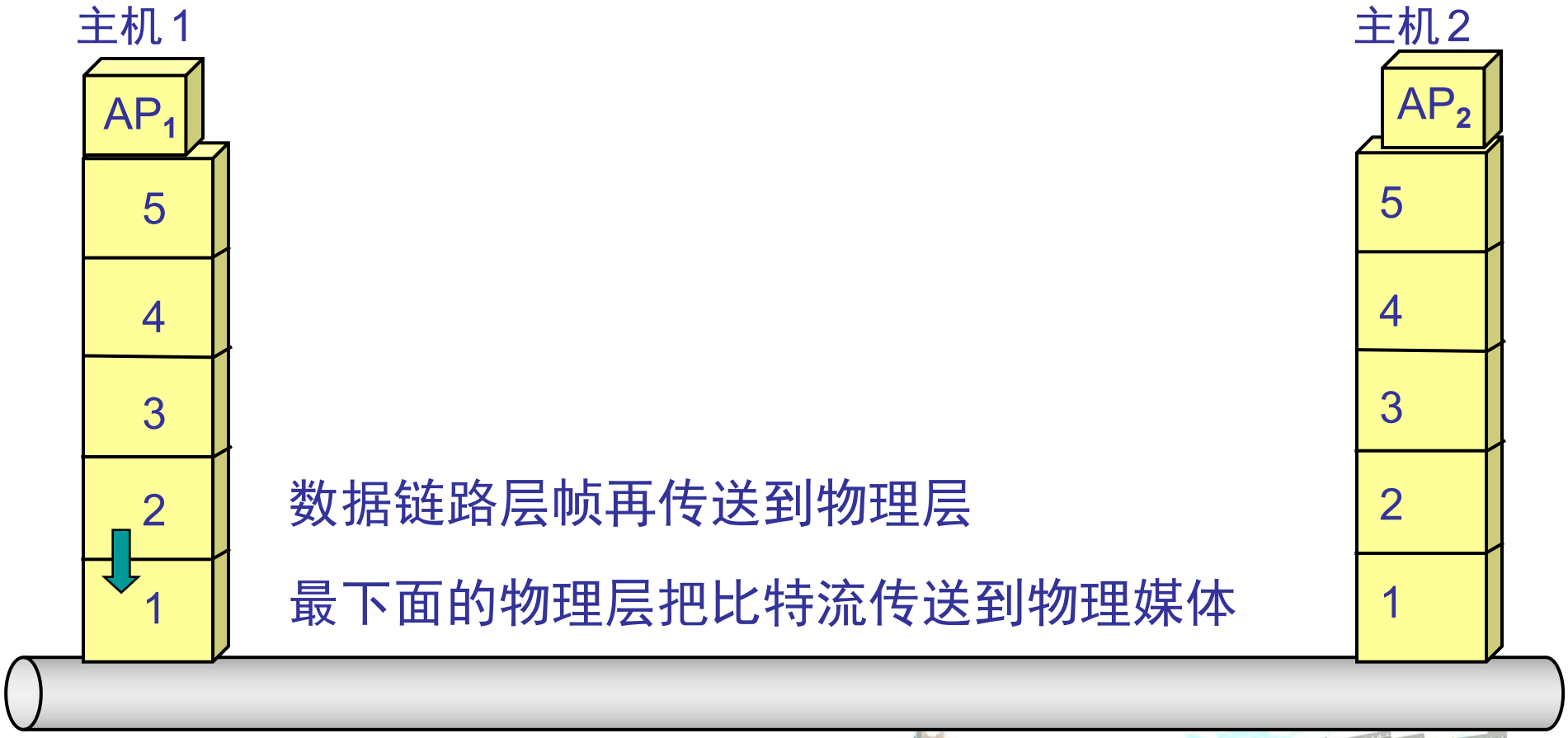
主机1向主机2发送数据

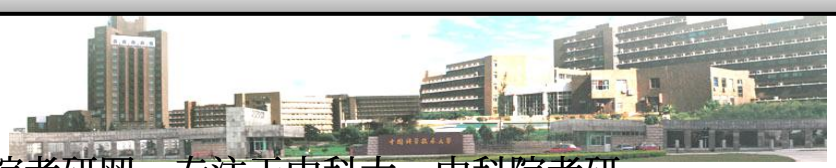


IP 数据报再传送到数据链路层

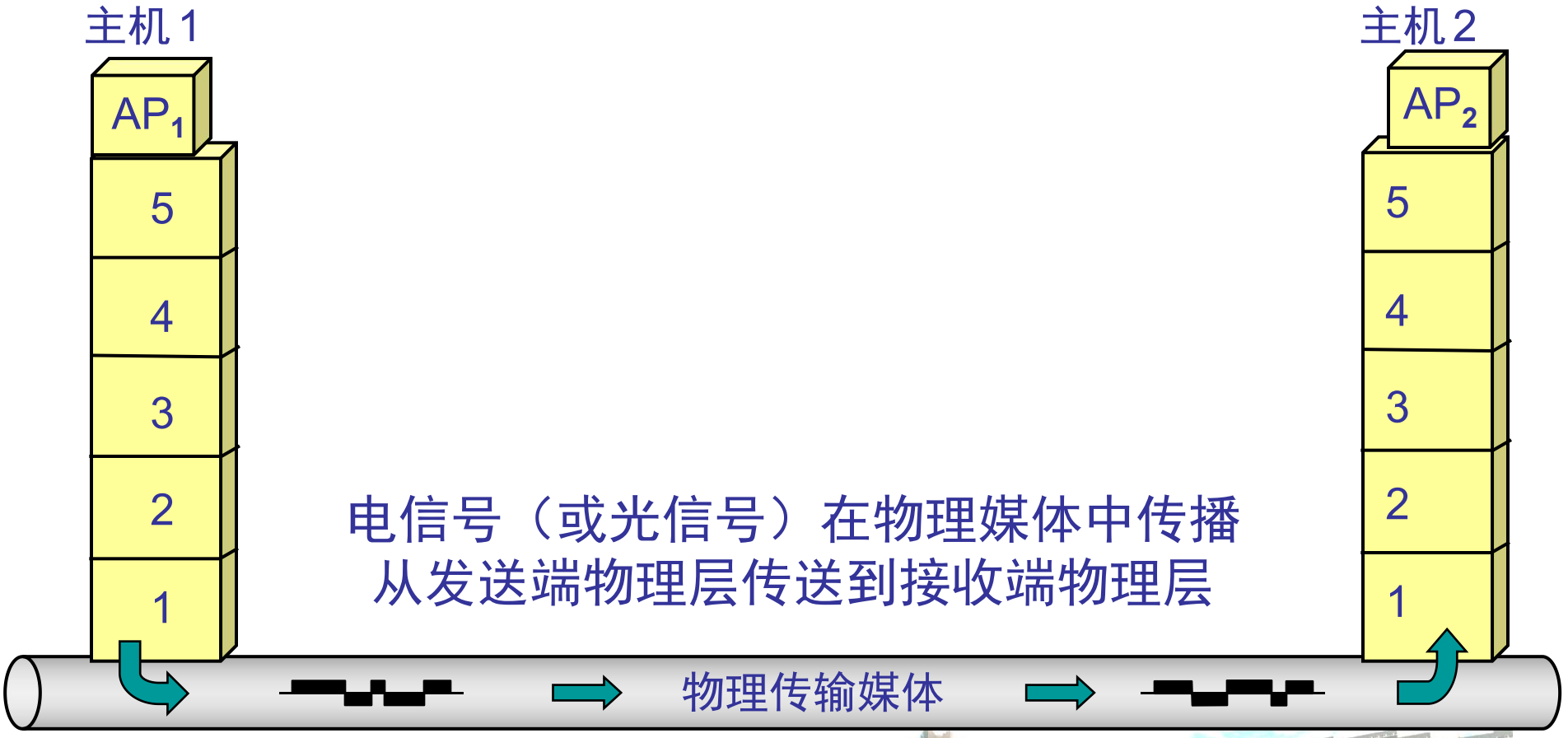
加上链路层首部和尾部，成为数据链路层帧

主机1向主机2发送数据

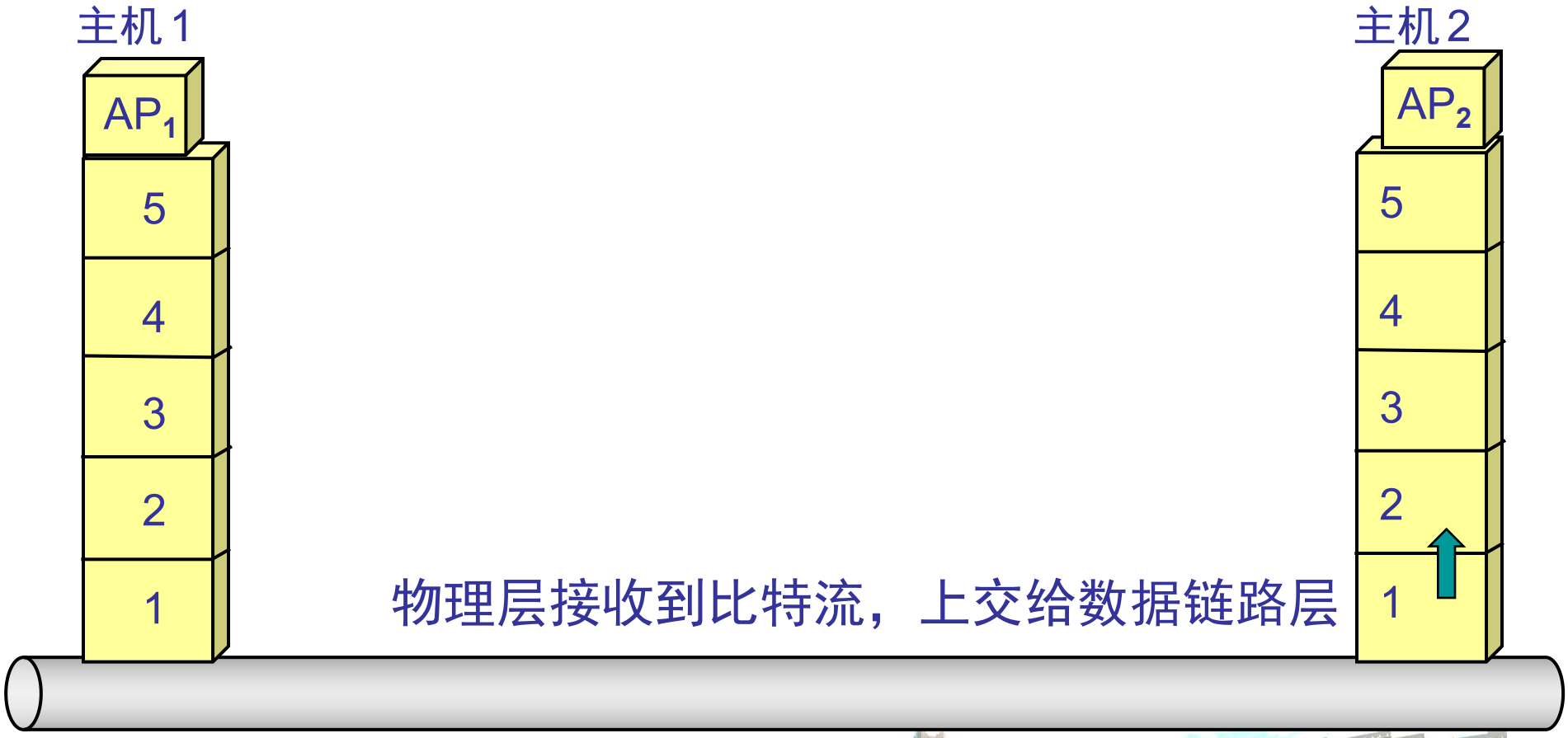


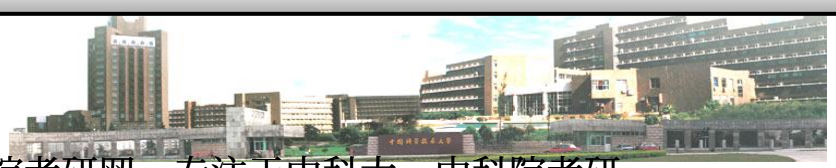


主机1向主机2发送数据

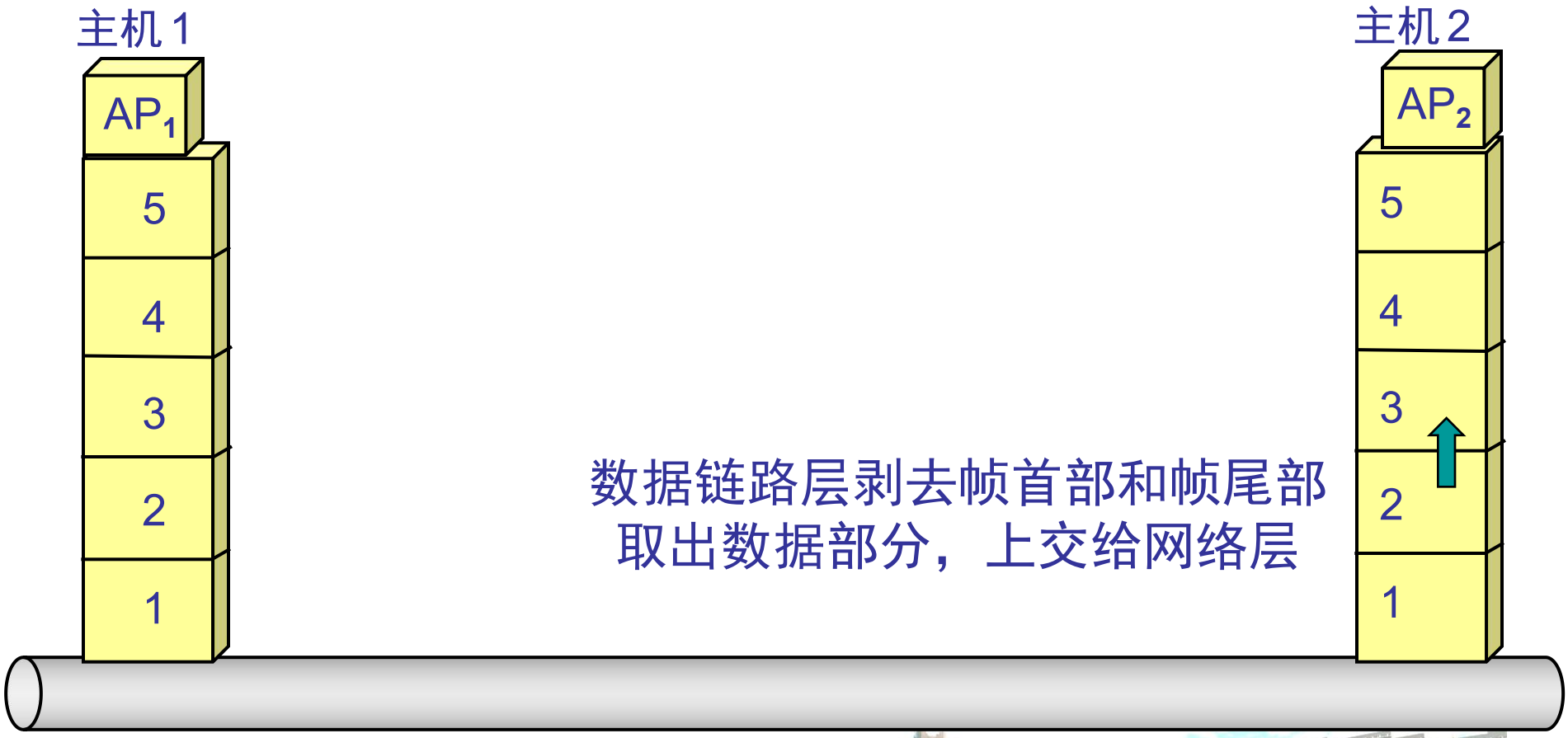


主机1向主机2发送数据

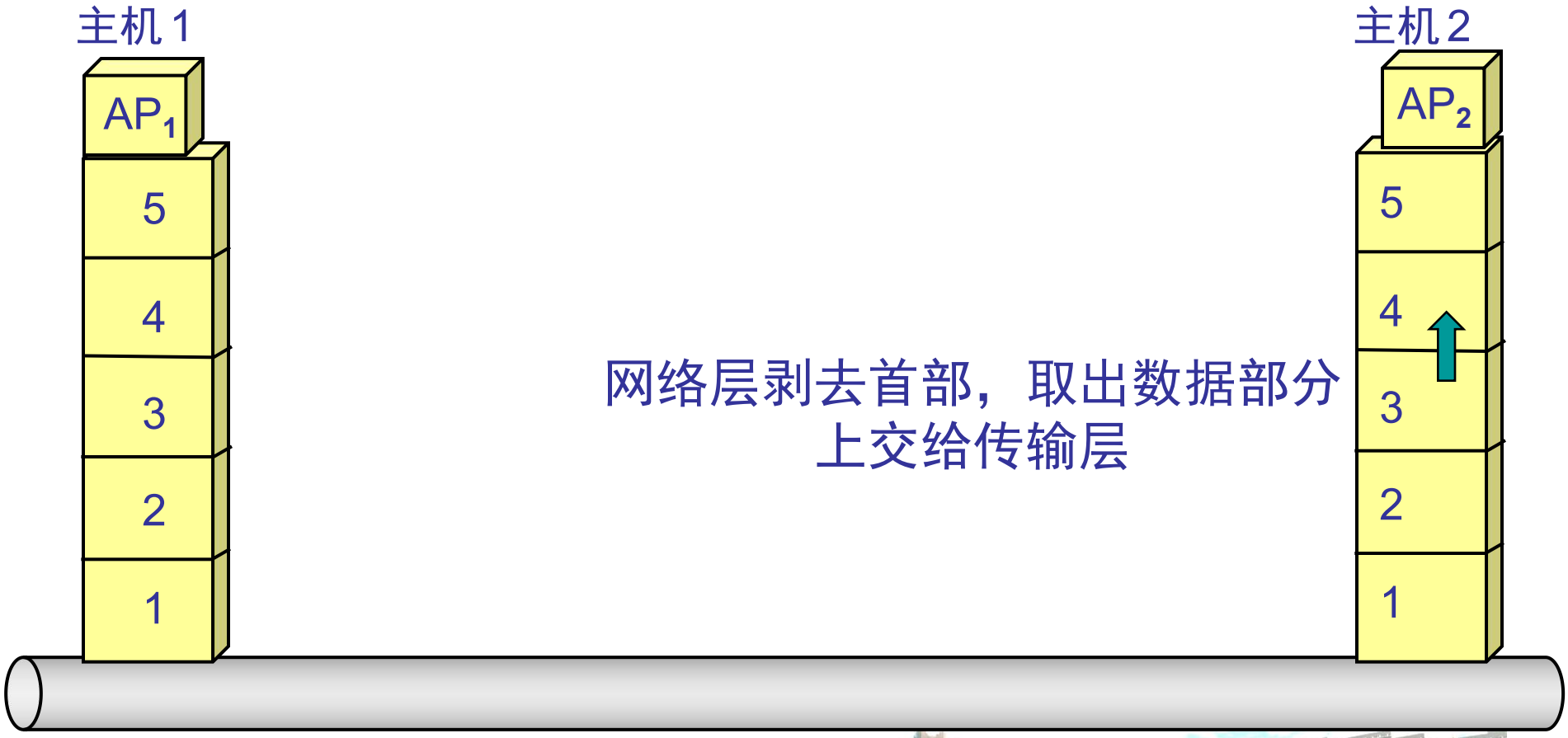


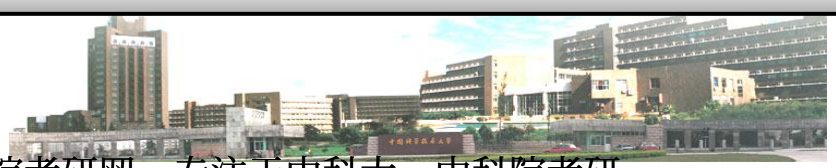


主机1向主机2发送数据

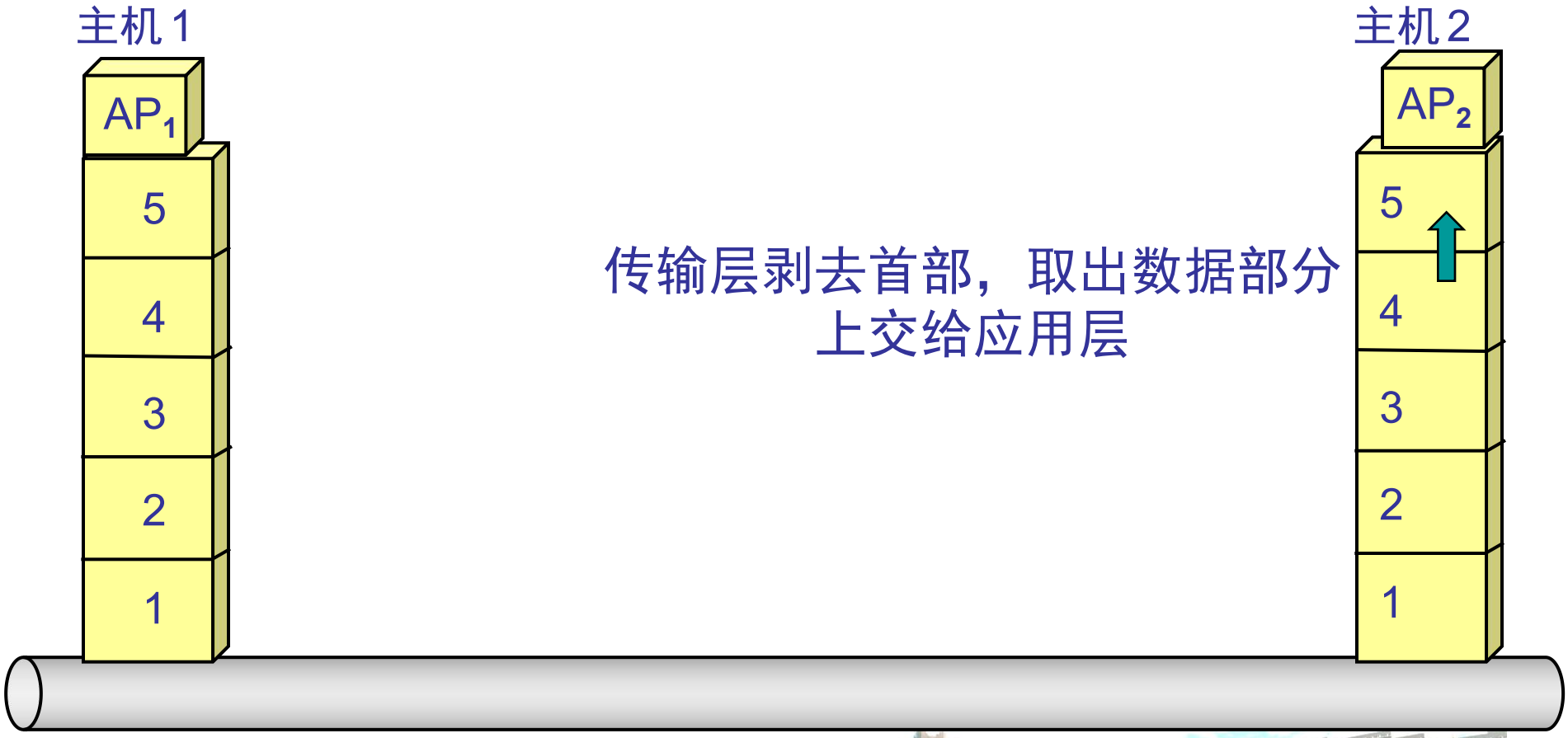


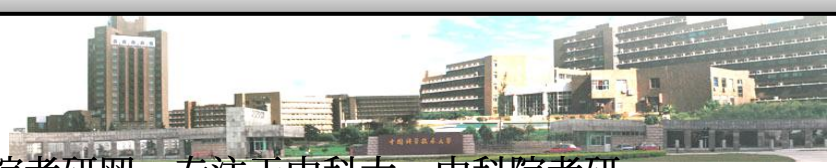
主机1向主机2发送数据



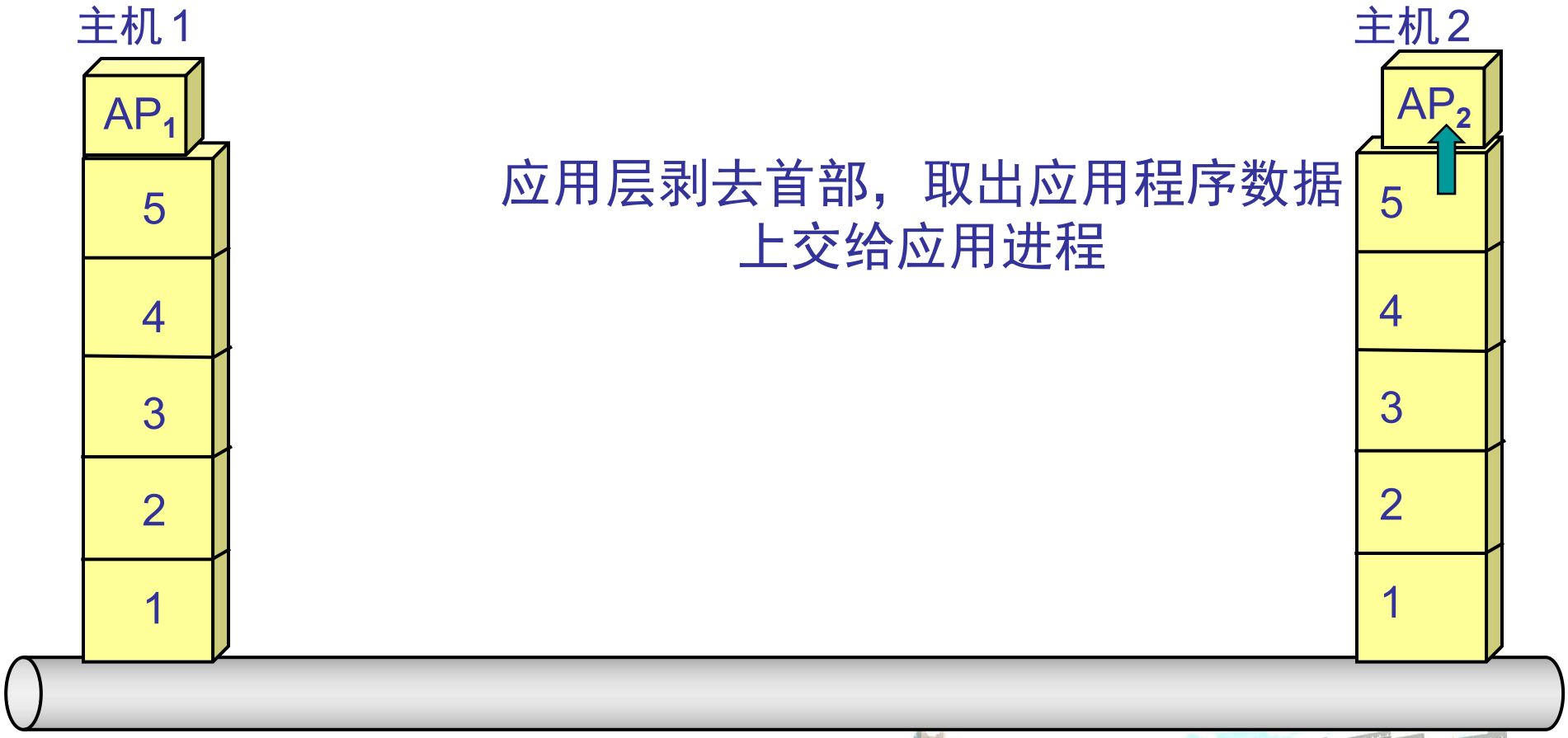


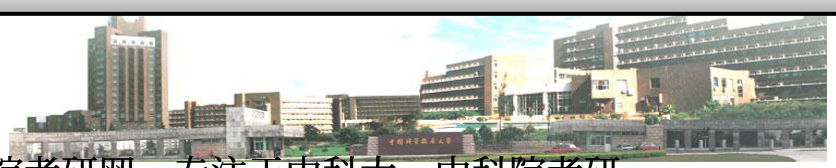
主机1向主机2发送数据



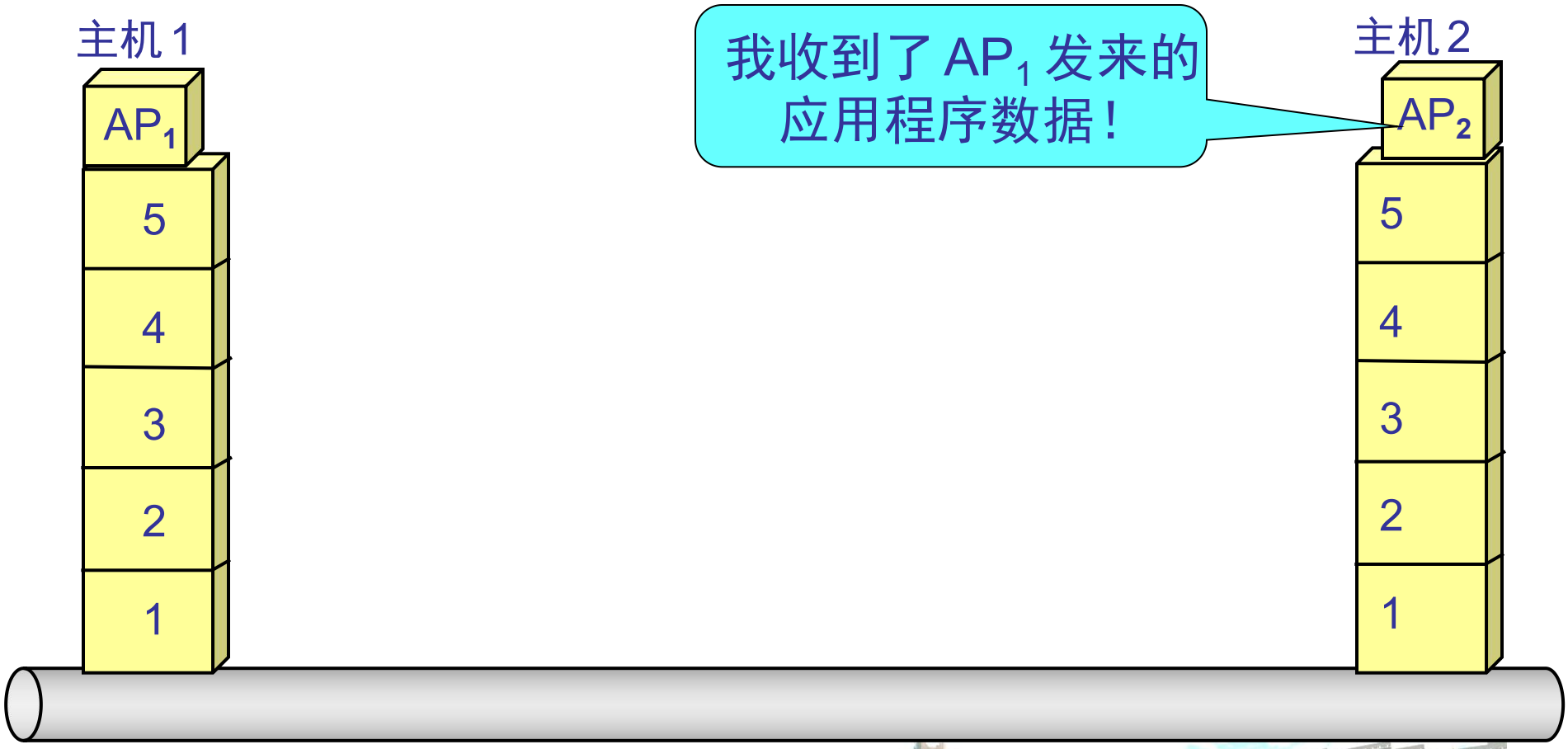


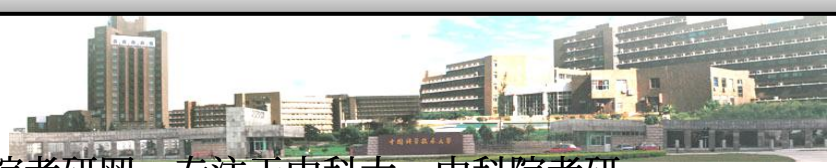
主机1向主机2发送数据



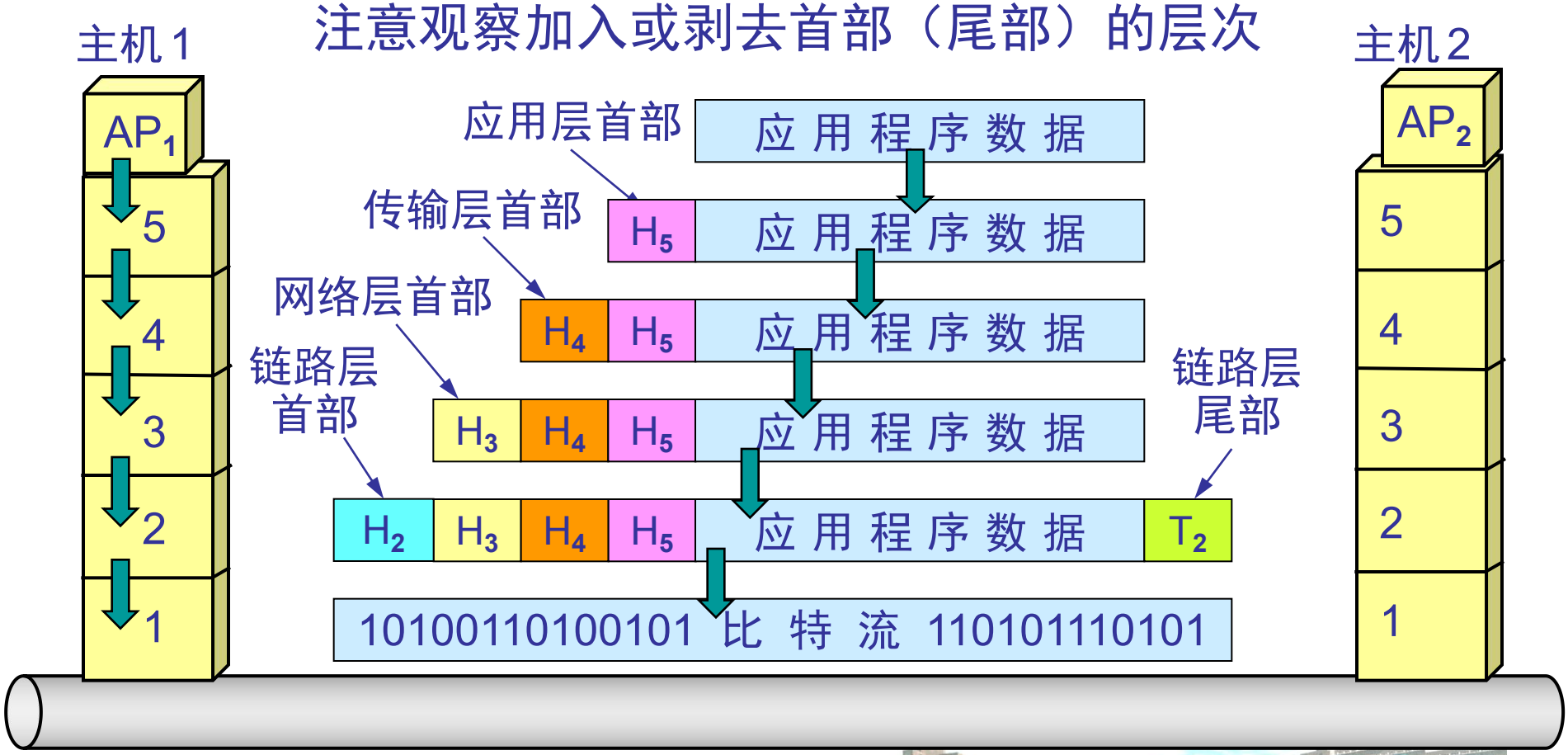


主机1向主机2发送数据

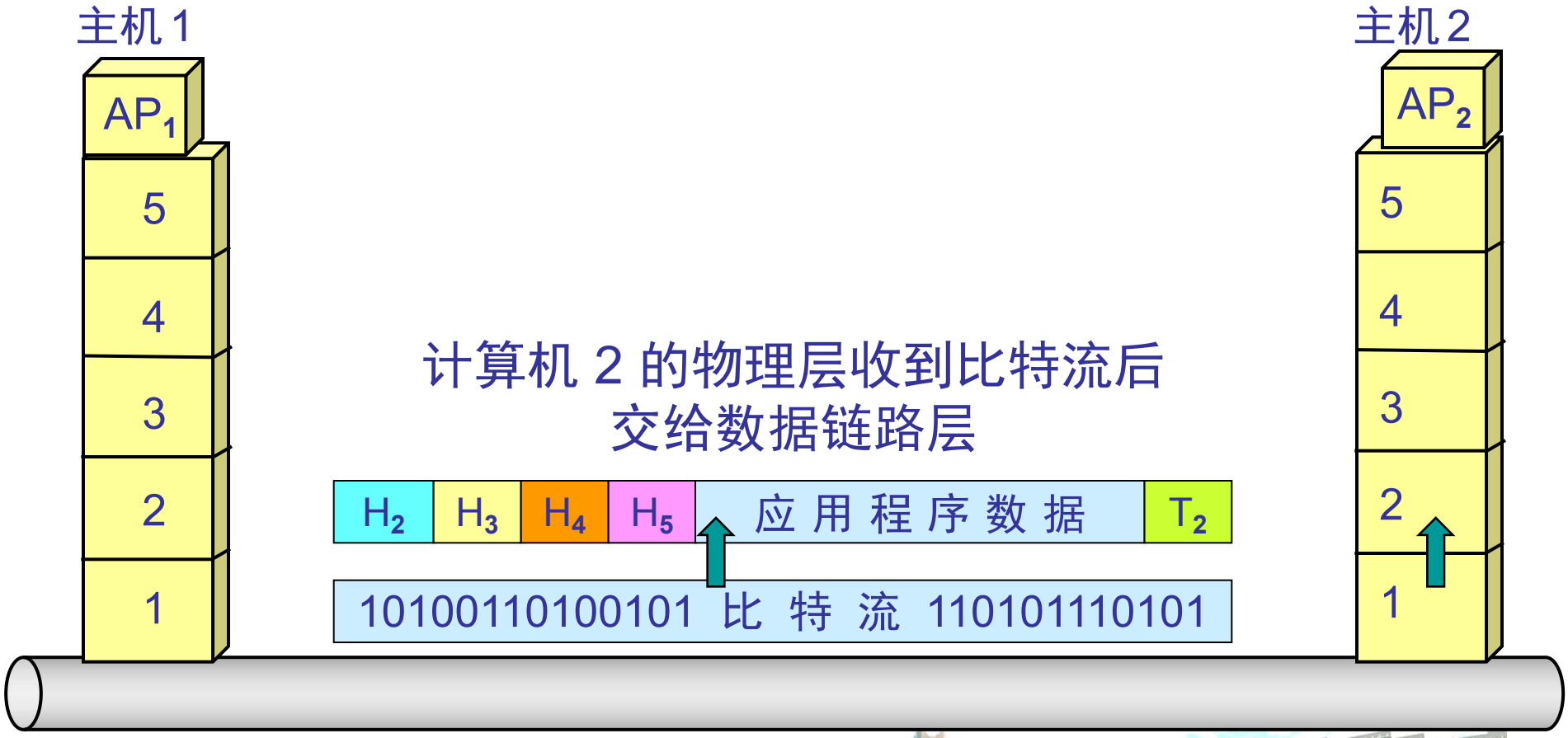


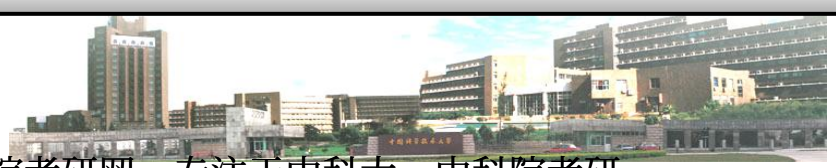


主机1向主机2发送数据

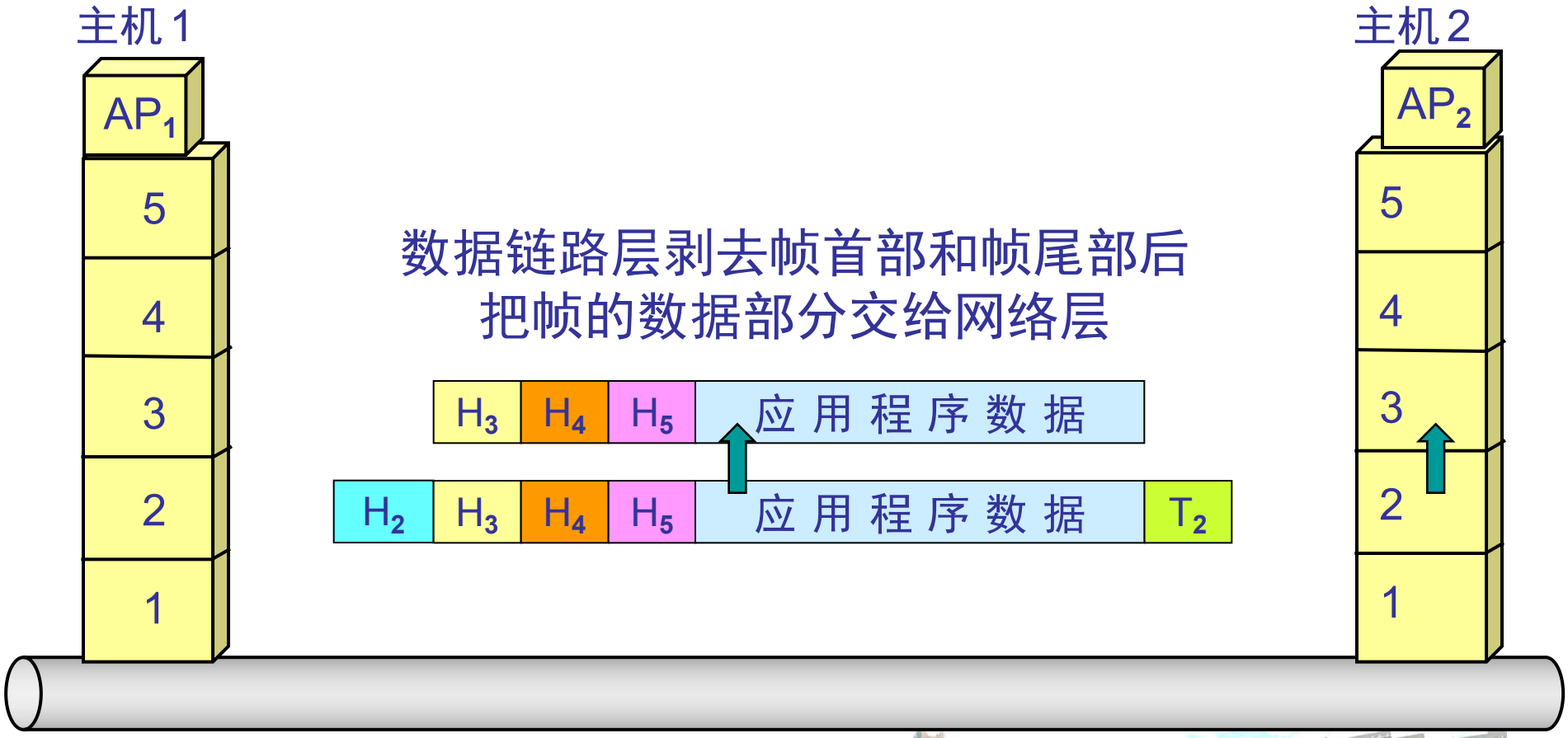


主机1向主机2发送数据

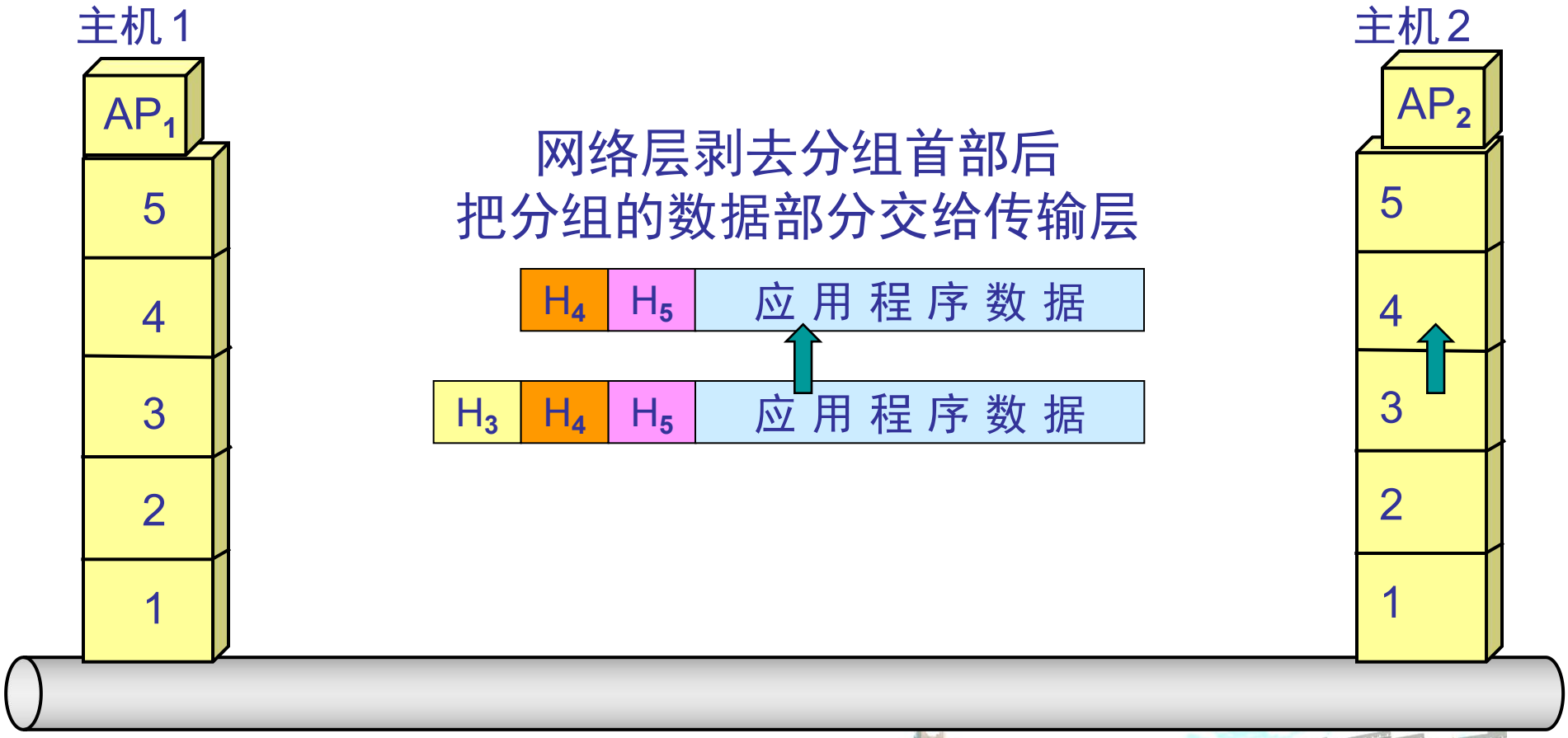


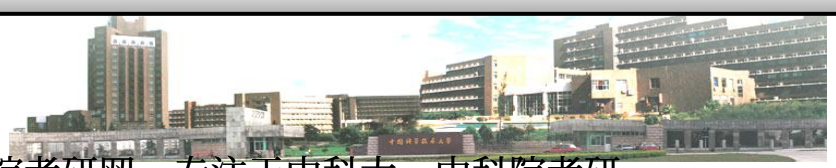


主机1向主机2发送数据

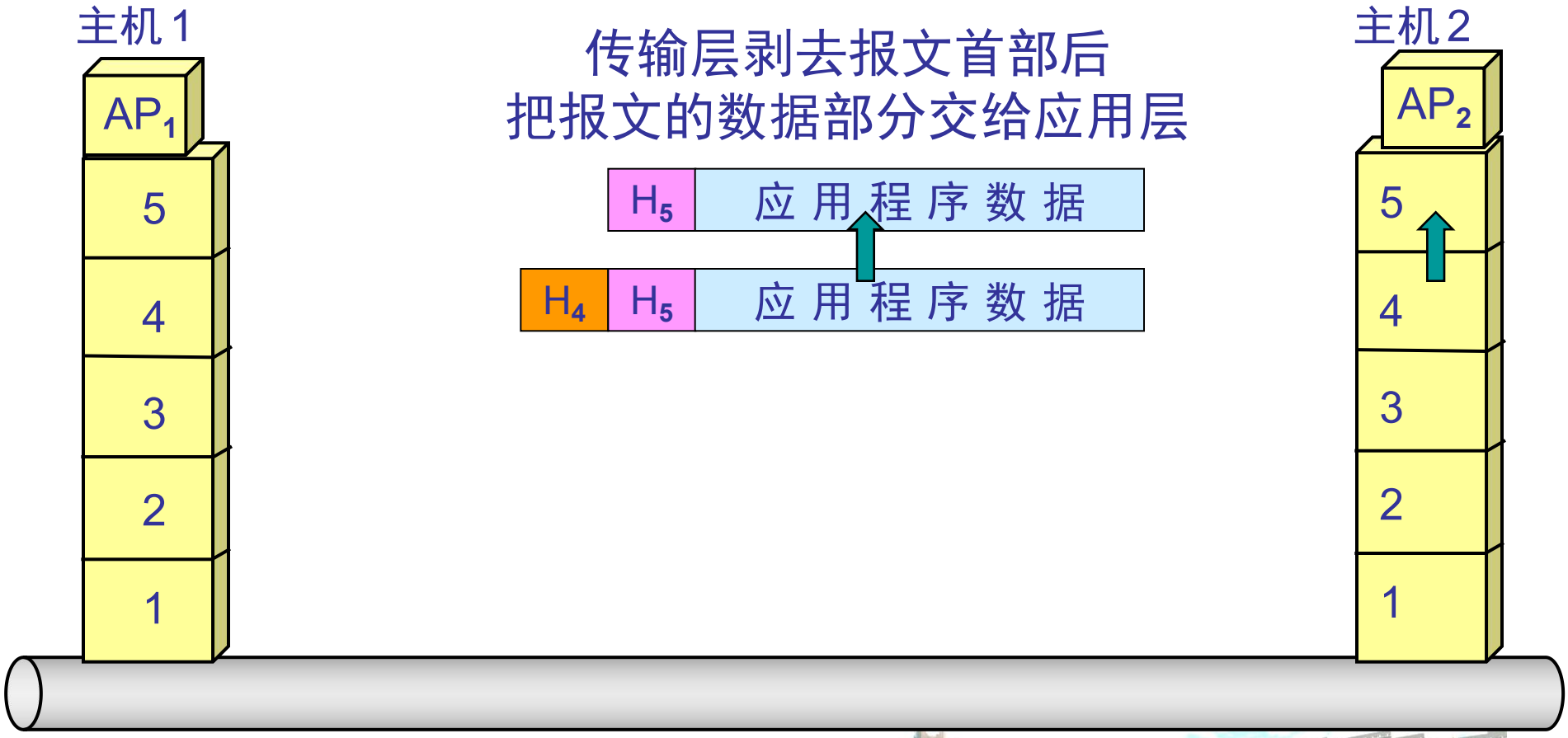


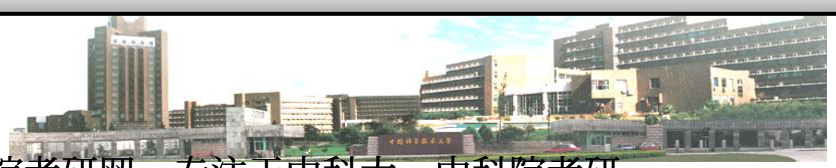
主机1向主机2发送数据



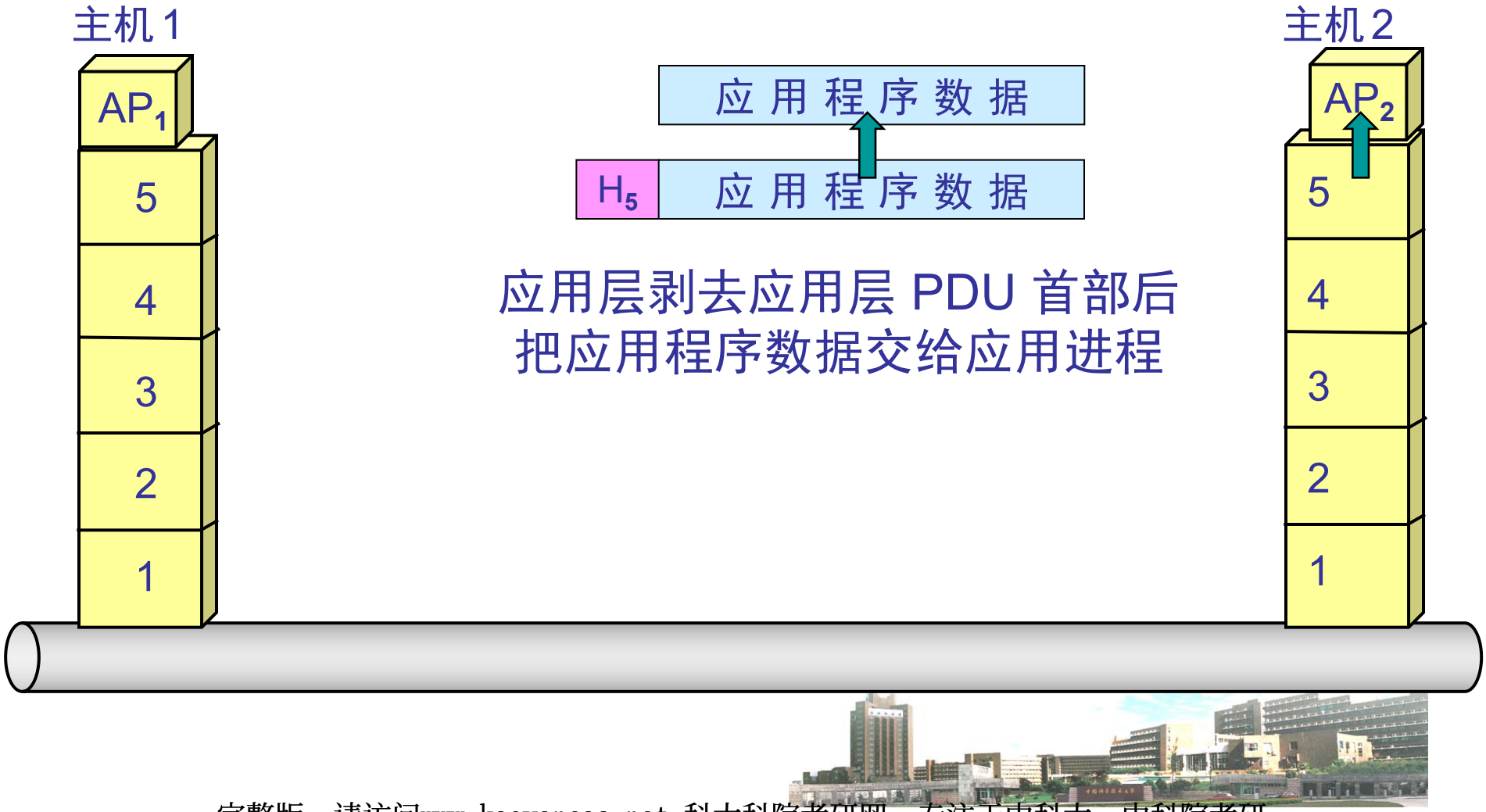


主机1向主机2发送数据



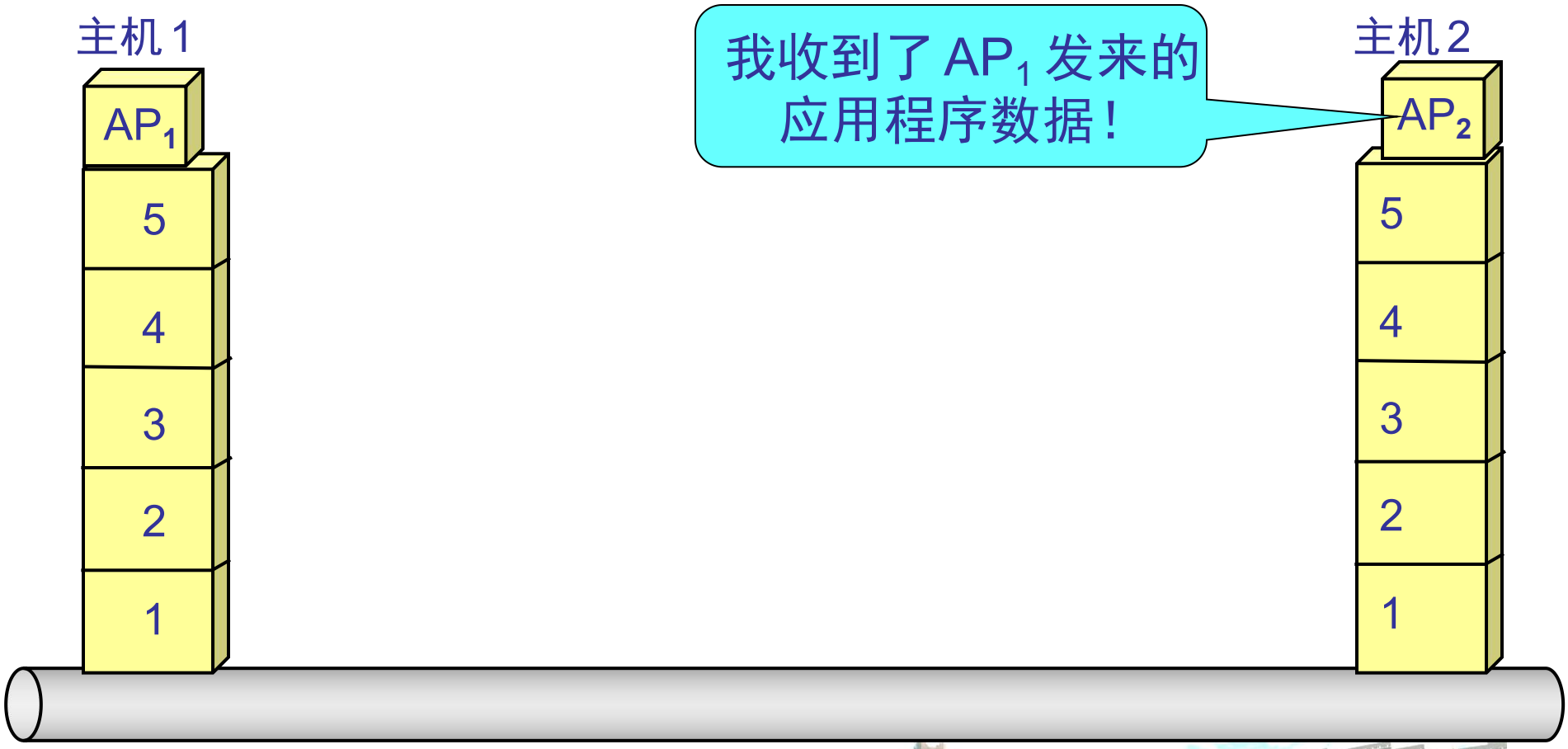


主机1向主机2发送数据



应用层剥去应用层 PDU 首部后
把应用程序数据交给应用进程

主机1向主机2发送数据



1.3.2 Design Issues for the Layers (1)

- **Addressing**
  - Mechanism for identifying senders and receivers: addressing to specify a party
- **Rules for data transfer**
  - Simplex communication, SDX
  - Half-duplex communication, HDX
  - Full-duplex communication, FDX
- **Error control**
  - Physical communication circuits are not perfect, so that error-detecting and error-correcting code are used. In addition, receiver must have some way of telling the sender which messages have been correctly received and which have not.

1.3.2 Design Issues for the Layers (2)

- **Sequencing**
  - Keep the messages order
- **Flow control**
  - Keep a fast sender from swamping a slow receiver with data
- **Disassembling and reassembling**
  - Allow all processes to accept arbitrarily long messages
- **Multiplexing**
  - Use the same connection for multiple, unrelated conversations
- **Routing**
  - Select a route from multiple paths between source and destination

1.3.3 Connection-oriented and Connectionless Services (1)

- **What is service?**
  - A set of communication abilities and operations provided by lower layer to its higher layer.
- **Connection-oriented Services**
  - Three phases: First **establish** a connection, then do a lot **communication**, and finally **release** the connection; all packets are transported on the same connection (route).

Services: Connections or Not (2)

- **Connectionless Services**
  - No connection established before hand.
  - Data is put into some kind of envelope called packet, on which the fully destination address has been written.
  - The envelope + contents (packet) is independently routed in communication subnet (may follow different route), until gets to the destination, and that's it.

Services: Connections or Not (3)

- Each service can provide some quality:
  - *Is data delivered **in the order** it was sent?* With connections, this is generally the case.
  - *Is data transmission **reliable**?* Generally offered with connections, but not always with connectionless services. Reliability requires sending acknowledgements, so that performance may degrade.

Connection-Oriented and Connectionless Services (4)

- **Six different types of service**

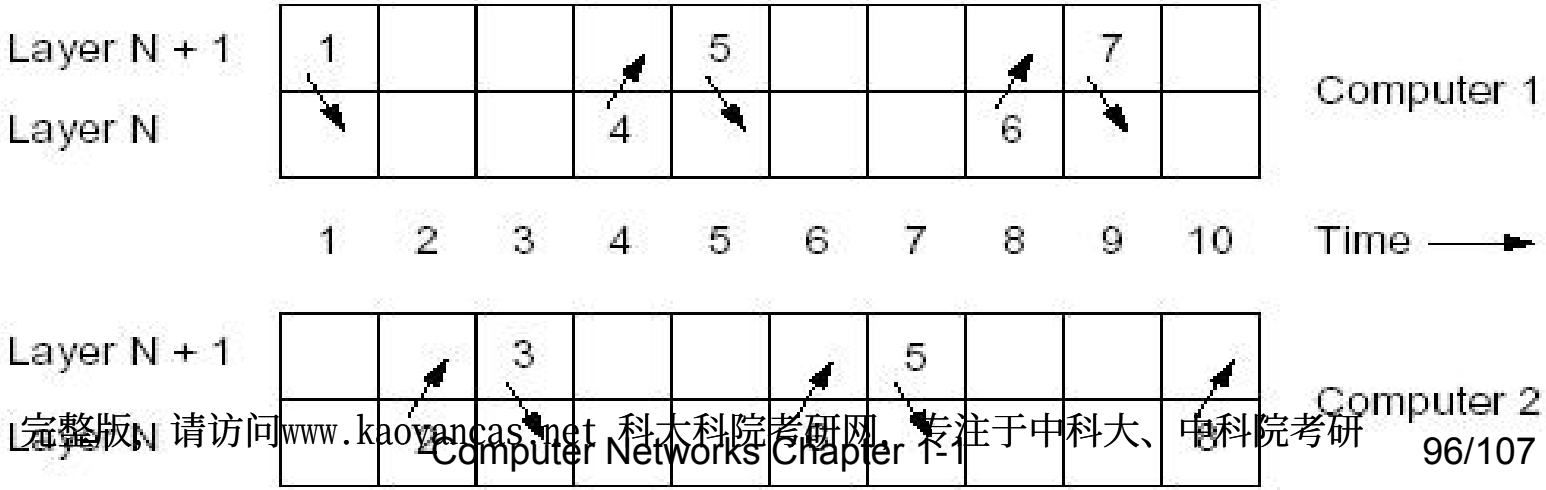
	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Remote login
	Unreliable connection	Digitized voice
Connectionless	Unreliable datagram	Electronic junk mail
	Acknowledged datagram	Registered mail
	Request-reply	Database query

1.3.4 Service Primitives

- Services are generally specified by a set of **primitives** (operations), which tell the service to perform some action or report on an action taken by a peer entity. Primitives are normally system calls.

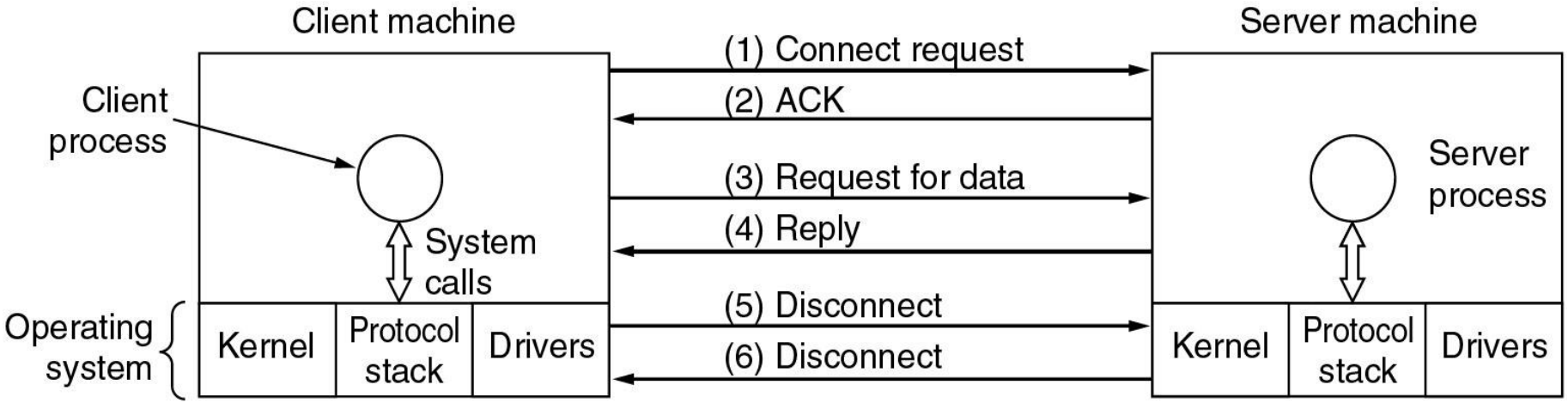
Primitive	Meaning
LISTEN	Block waiting for an incoming connection
CONNECT	Establish a connection with a waiting peer
RECEIVE	Block waiting for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection

1. CONNECT.request: request for establishing a connection (*dial a phone number*).
2. CONNECT.indication: signal the callee (*phone rings*).
3. CONNECT.response: reaction by callee to indication (*pick up the phone*).
4. CONNECT.confirm: tell caller whether call was accepted (*caller hears ringing stop*).
5. DATA.request: request data to be sent (*say something*)
6. DATA.indication: signal arrival of data (*callee hears you*)
7. DISCONNECT.request: request release of connection (*caller hangs up*)
8. DISCONNECT.indication: signal release of connection (*callee hears busy tone*)



Service Primitives (2)

- Packets sent in a simple client-server interaction on a connection-oriented network.



1.3.5 The Relationship of Services to Protocols

- **Service**
  - is a set of primitives (operations) that a layer provides to the layer above it, defining what operations the layer is prepared to perform on behalf of its users, but nothing about how these operations are implemented.
- **Protocol**
  - is a set of rules governing the format and meaning of the packets, or messages that are exchanged by the peer entities within a layer. Entities use protocols to implement their service definitions.

The Relationship of Services to Protocols

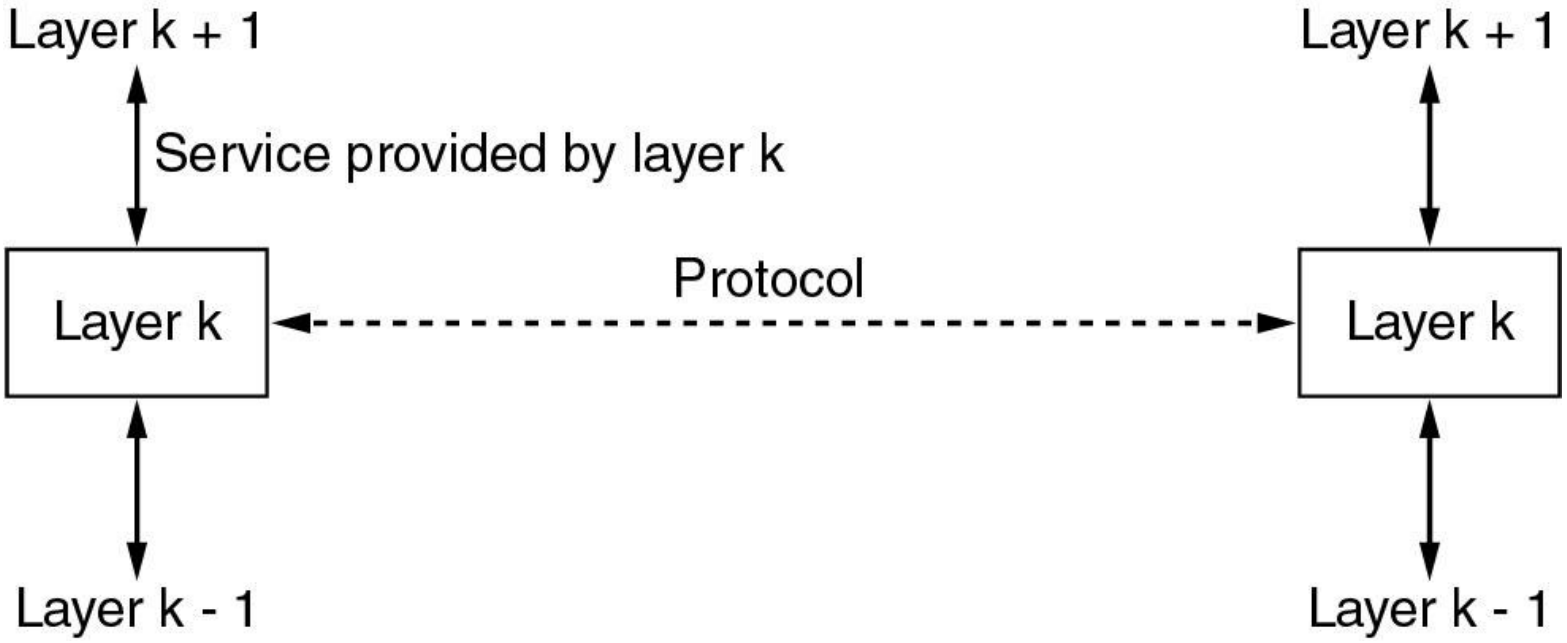


Fig. 1.19 The relationship between a service and a protocol

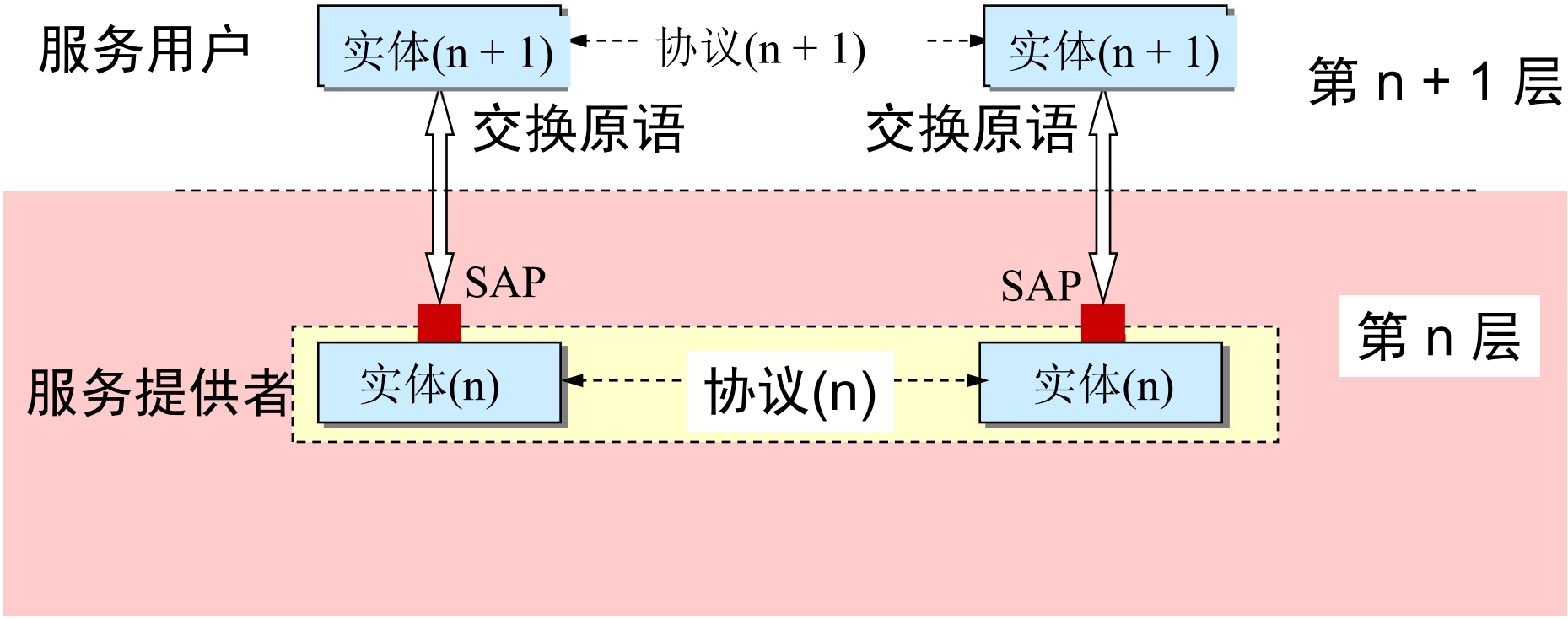
实体、协议、服务和服务访问点

- 实体(entity) 表示任何可发送或接收信息的硬件或软件进程。
- 协议是控制两个对等实体进行通信的规则的集合。
- 在协议的控制下，两个对等实体间的通信使得本层能够向上一层提供服务。
- 要实现本层协议，还需要使用下层所提供的服务。

实体、协议、服务和服务访问点（续）

- 本层的服务用户只能看见服务而无法看见下层的协议。
- 下层的协议对上层的服务用户是透明的。
- 协议是“水平的”，即协议是控制对等实体之间通信的规则。
- 服务是“垂直的”，即服务是由下层向上层通过层间接口提供的。
- 同一系统相邻两层的实体进行交互的地方，称为服务访问点 SAP (Service Access Point)。

实体、协议、服务和服务访问点（续）



协议很复杂

- 协议必须把所有不利的条件事先都估计到，而不能假定一切都是正常的和非常理想的。
- 看一个计算机网络协议是否正确，不能光看在正常情况下是否正确，而且还必须非常仔细地检查这个协议能否应付各种异常情况。

著名的协议举例

- 占据东、西两个山顶的蓝军1和蓝军2与驻扎在山谷的白军作战。其力量对比是：单独的蓝军1或蓝军2打不过白军，但蓝军1和蓝军2协同作战则可战胜白军。
- 现蓝军1拟于次日正午向白军发起攻击，于是发送电文给蓝军2。但通信线路很不好，电文出错或丢失的可能性较大（没有电话可使用）。因此要求收到电文的友军必须送回一个确认电文。确认电文也可能出错或丢失。
- 试问能否设计出一种协议使得蓝军1和蓝军2能够实现协同作战因而一定（即100 %而不是99.999…%）取得胜利？

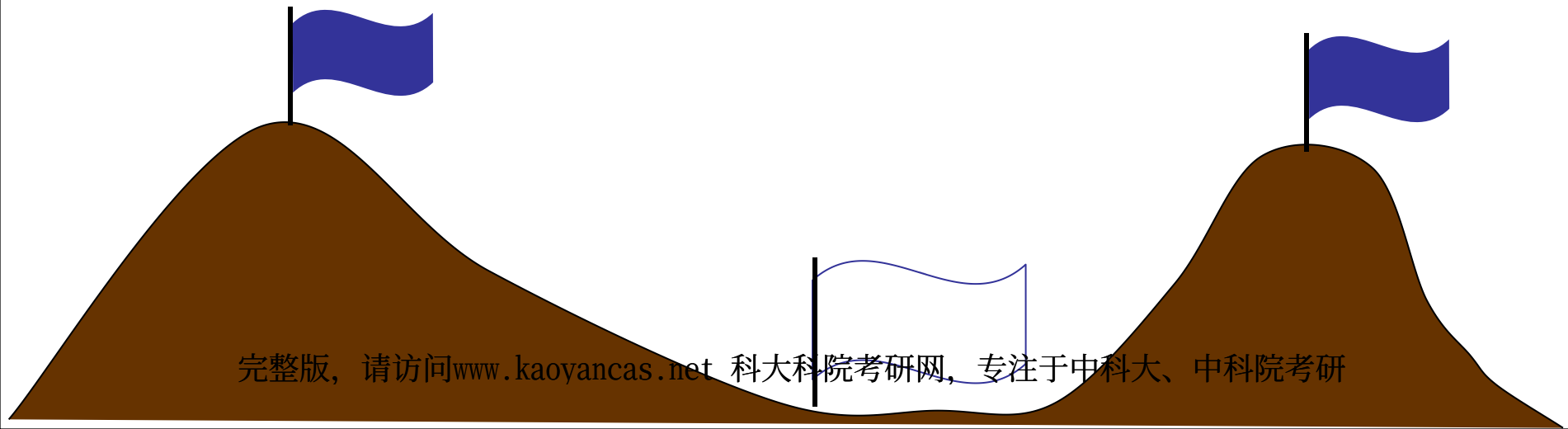
明日正午进攻，如何？

同意

收到“同

这样的协议无法实现！

收到：收到“同意”



结 论

- 这样无限循环下去，两边的蓝军都始终无法确定自己最后发出的电文对方是否已经收到。
- 没有一种协议能够使蓝军 100% 获胜。

End of part 1, Chapter 1