



# 计算机网络

俞嘉地

计算机科学与工程系  
上海交通大学



## ❖ **Course Website:**

<http://www.cs.sjtu.edu.cn/~jdyu/teaching/CN/index.html>

## ❖ **About Myself:**

Jiadi Yu, Assistant Prof., Ph.D

Office Room: 3-129, SEIEE

E-Mail: [jdyu@cs.sjtu.edu.cn](mailto:jdyu@cs.sjtu.edu.cn)



# 本课程教材

## Computer Networks (Fourth Edition)

Andrew S. Tanenbaum

Vrije Universiteit

Amsterdam, The Netherlands

(影印版) ISBN 7-302-07815-7/TP.5697

清华大学出版社2004-01



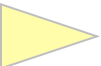
# 本课程参考书目

- ❖ **计算机网络（第四版）**  
A. S. Tenenbaum著潘爱民等译 清华大学出版社 2004-08
- ❖ **数据通信与计算机网络**  
高传善等编 高等教育出版社 2001-07
- ❖ **数据与计算机通信（第六版）**  
William Stallings著王海等译 电子工业出版社 2001-05
- ❖ **计算机网络与因特网**  
Douglas E. Comer著徐良贤等译 机械工业出版社 2000-08
- ❖ **TCP/IP网络原理与技术**  
周明天、汪文勇编 清华大学出版社 1993-12
- ❖ **计算机网络习题与解析**  
鲁士文编 清华大学出版社 2001-06



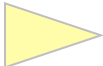


# 第1章 概述

1. 计算机网络组成和分类 
2. 计算机网络参考模型 
3. 数字通信的基本概念 



# 计算机网络组成和分类

- ❖ 主机系统和网络系统 
- ❖ 计算机网络与分布式系统 
- ❖ Internet 
- ❖ 计算机网络的分类 



# 主机系统和网络系统

❖ 主机和终端组成的主机系统



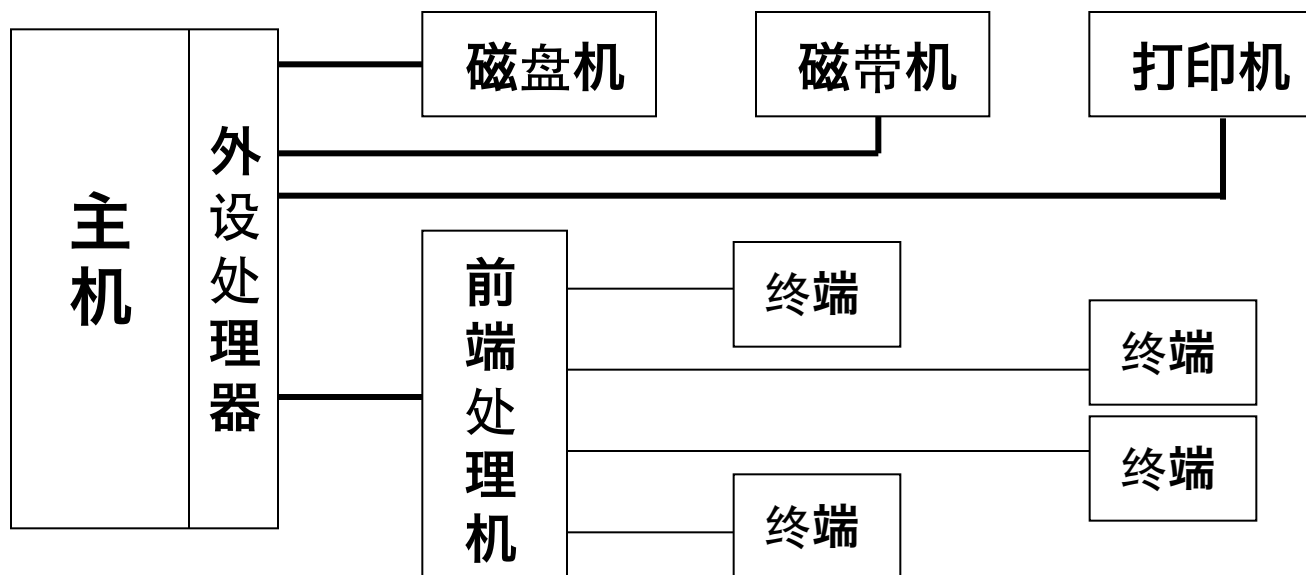
❖ Client/Server结构的计算机

网络系统





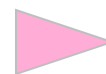
# 主机和终端





# 主机系统和网络系统

❖ 主机和终端组成的主机系统



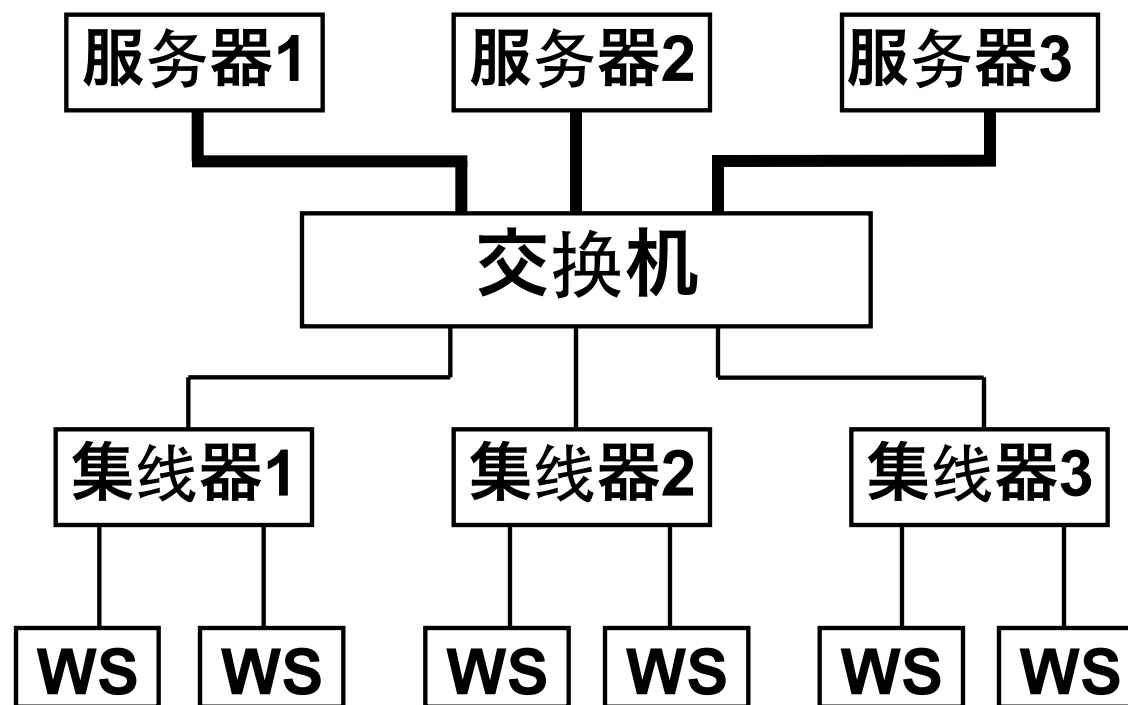
❖ Client/Server结构的计算机

网络系统





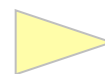
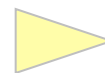
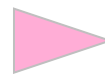
# Client/Server结构 (C/S)





# 计算机网络组成和分类

- ❖ 主机系统和网络系统
- ❖ 计算机网络与分布式系统
- ❖ Internet
- ❖ 计算机网络的分类





# 计算机网络与分布式系统

❖ 计算机网络 

❖ 分布式系统 





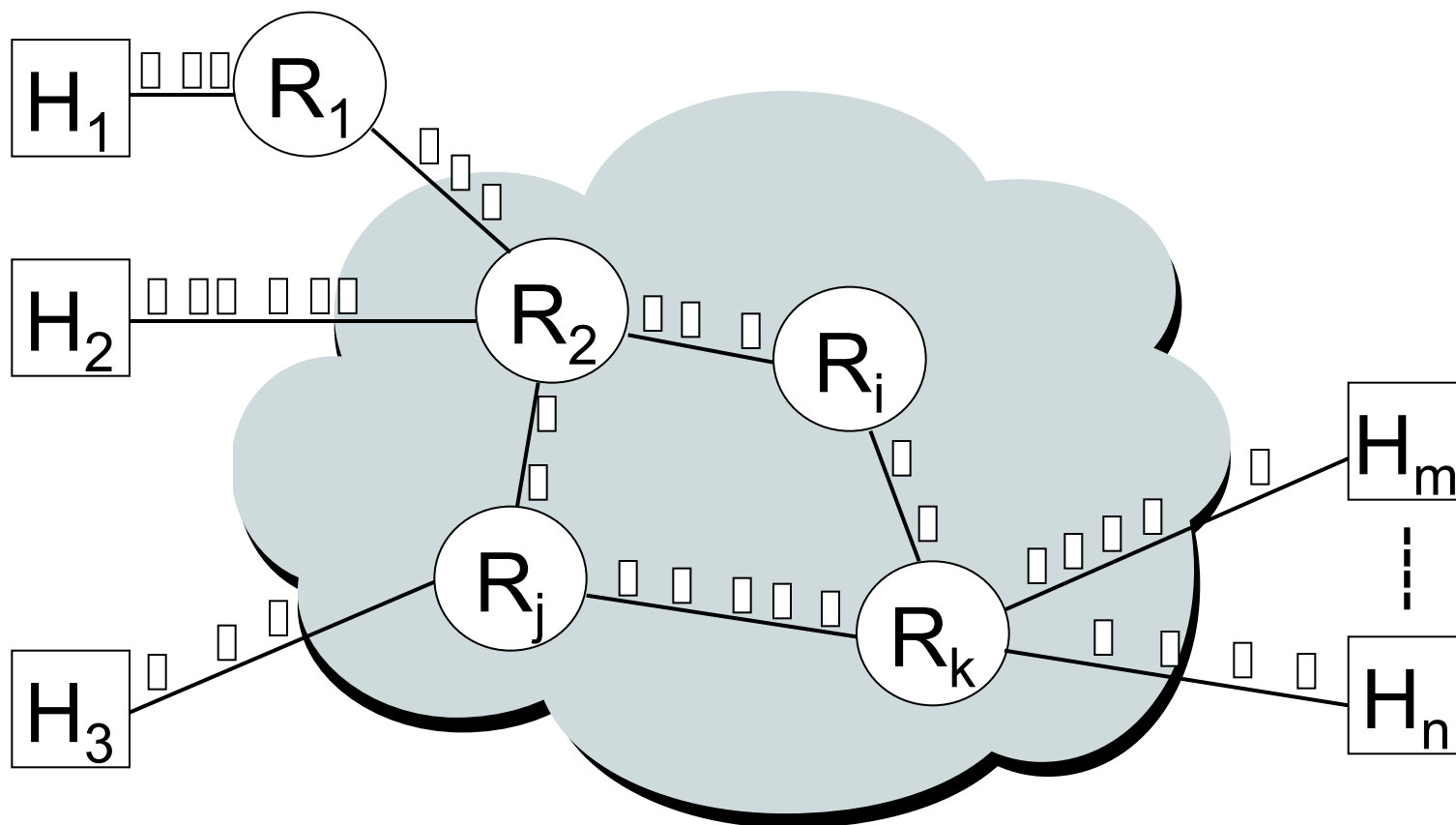
# 计算机网络

**计算机网络是由不同通信媒体连接的、物理上独立的多台计算机组成的、将需传输的数据分成不同长度的分组进行传输和处理的系统**

- **多台自主计算机的互联系统**
- **用户必须指定网络中哪一台计算机来完成什么样的操作**



# 计算机网络图示





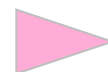
# 两个概念：**互联和自主**

- ❖ **互联**：如两台计算机能互相交换信息，  
则称这两台计算机**互联**
- ❖ **自主计算机**：不从属于其它任何计算机  
的计算机称为**自主计算机**



# 计算机网络与分布式系统

❖ 计算机网络



❖ 分布式系统





# 分布式系统

**分布式系统通常也基于自主计算机的互联，但由一个操作系统统一管理，对用户来说，具有高度的整体性和透明性**

- **虚拟计算机，可能是多台计算机的有机组合**
- **用户向虚拟系统递交一个任务后，由操作系统来安排任务的完成**



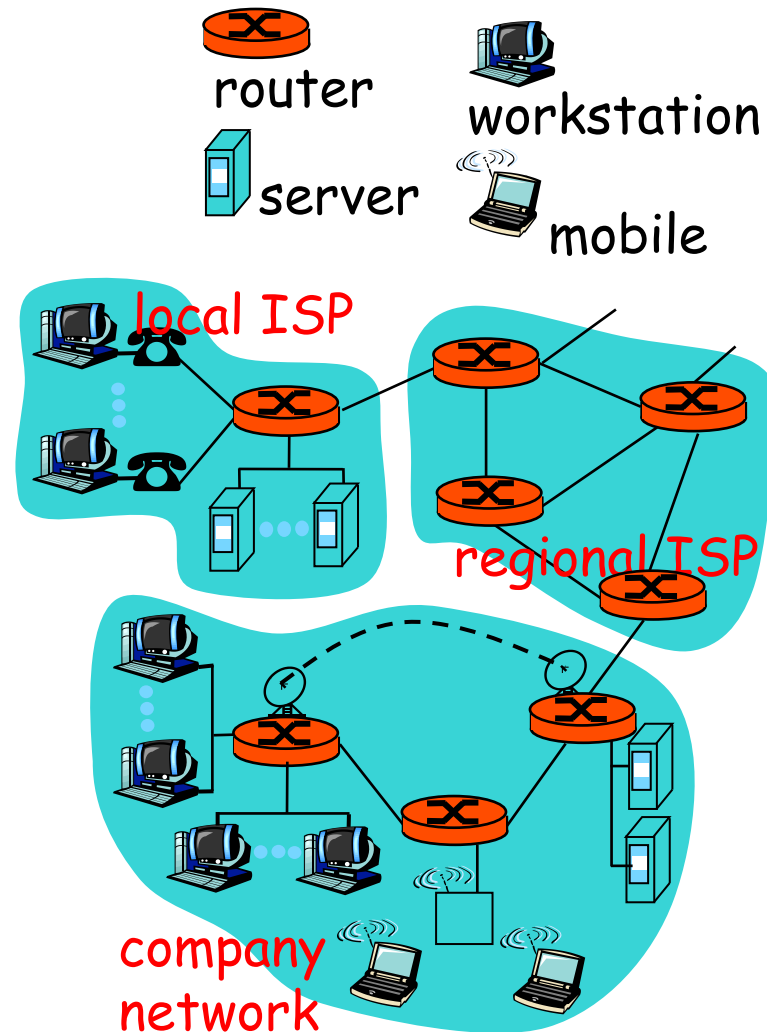
# 因特网

- ❖ 因特网定义
- ❖ 因特网协议
- ❖ 因特网体系结构



# What's the Internet?

- ❖ millions of connected computing devices:  
**hosts = end systems**
- ❖ running **network apps**
- ❖ **communication links**
  - fiber, copper, radio, satellite
  - transmission rate = **bandwidth**
- ❖ **routers**: forward packets (chunks of data)

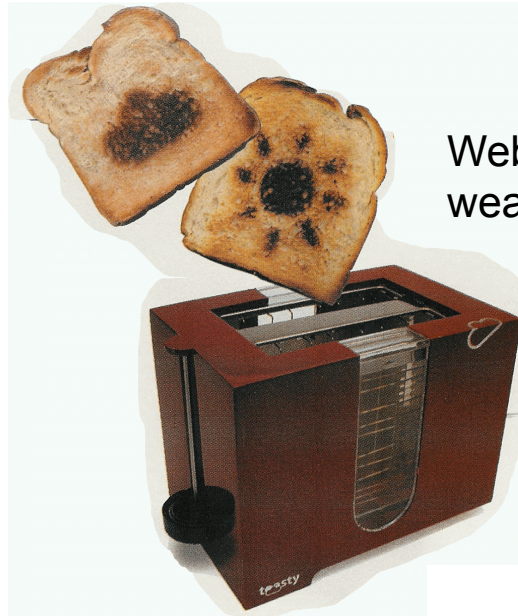




# “Cool” internet appliances



IP picture frame  
<http://www.ceiva.com/>



Web-enabled toaster +  
weather forecaster



World's smallest web server  
<http://www-ccs.cs.umass.edu/~shri/iPic.html>



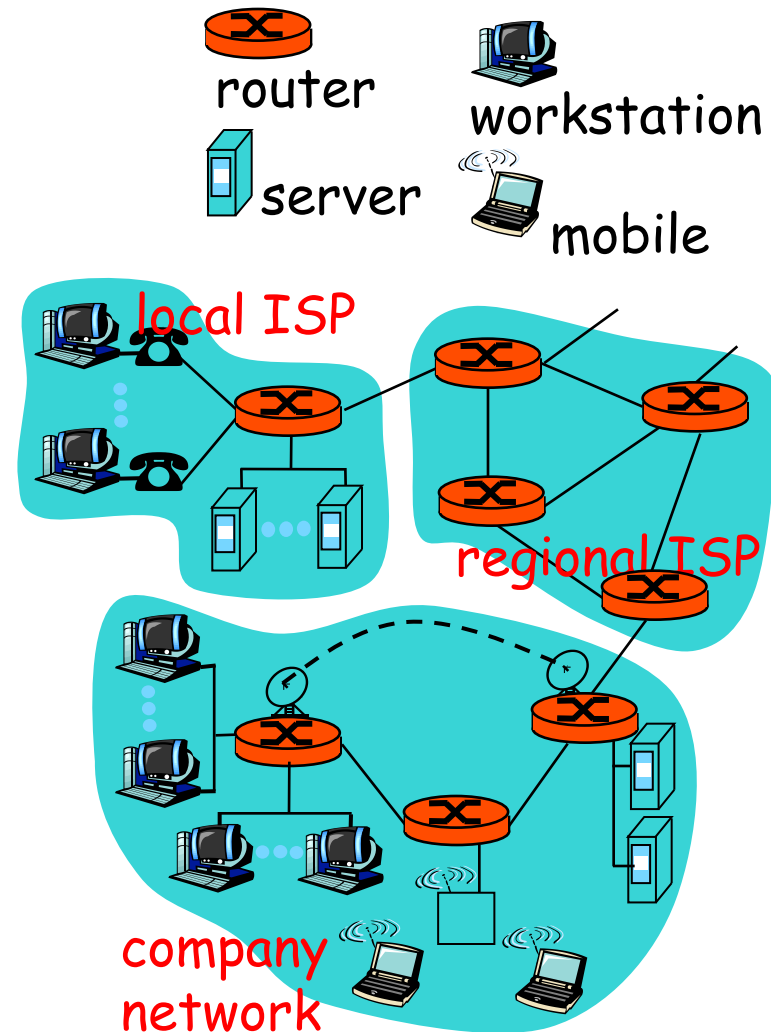
Internet phones





# What's the Internet?

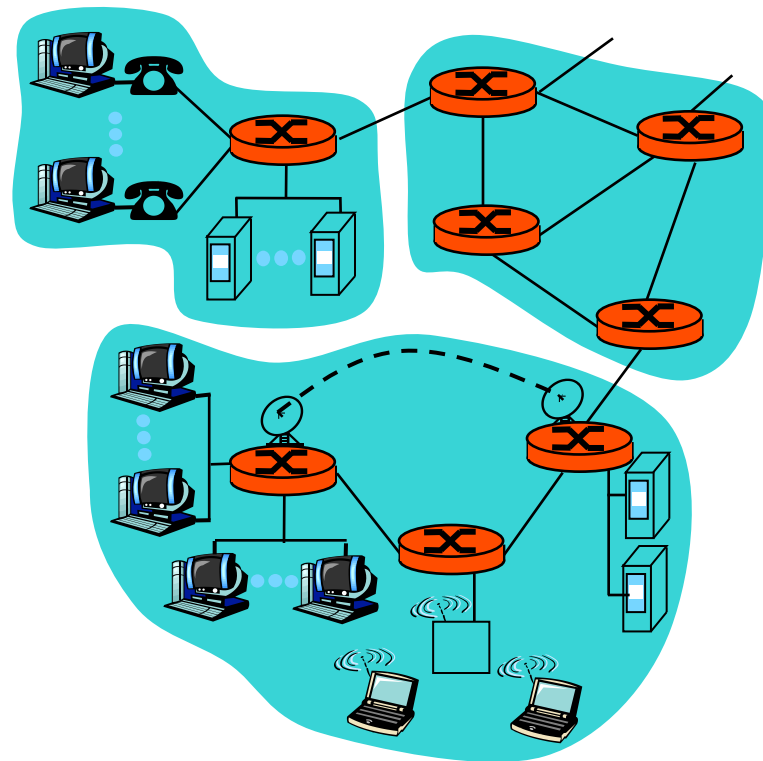
- ❖ **protocols** control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, FTP, PPP
- ❖ **Internet: "network of networks"**
  - loosely hierarchical
  - public Internet versus private intranet
- ❖ **Internet standards**
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force





# What's the Internet: a service view

- ❖ **communication infrastructure** enables distributed applications:
  - Web, email, games, e-commerce, file sharing
- ❖ **communication services** provided to apps:
  - Connectionless unreliable
  - connection-oriented reliable





# What's a protocol?

## human protocols:

- ❖ “what's the time?”
- ❖ “I have a question”
- ❖ introductions

... specific msgs sent

... specific actions  
taken when msgs  
received, or other  
events

## network protocols:

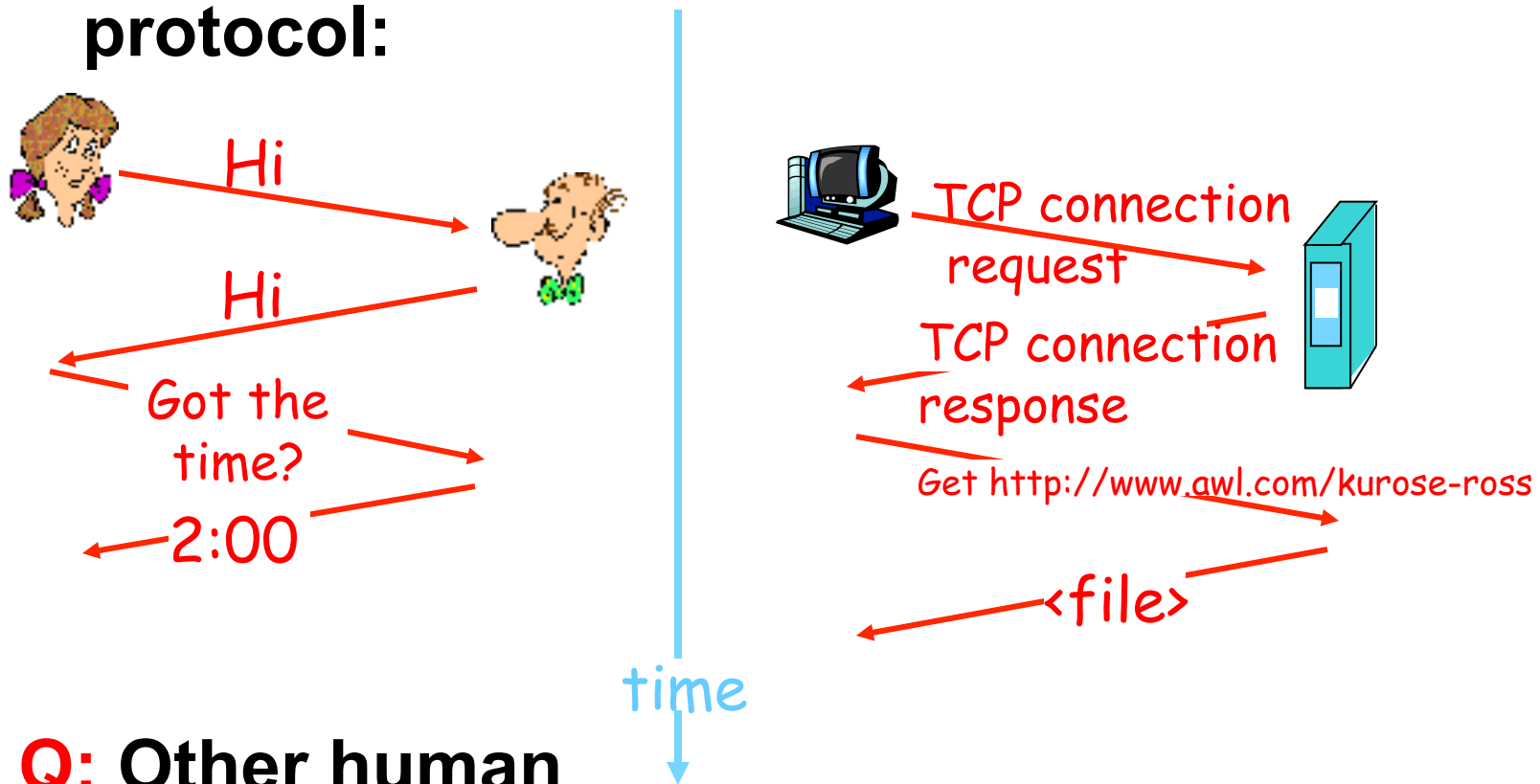
- ❖ machines rather than humans
- ❖ all communication activity in Internet governed by protocols

*protocols define format,  
order of msgs sent and  
received among network  
entities, and actions  
taken on msg  
transmission, receipt*



# What's a protocol?

a human protocol and a computer network protocol:



**Q: Other human protocols?**

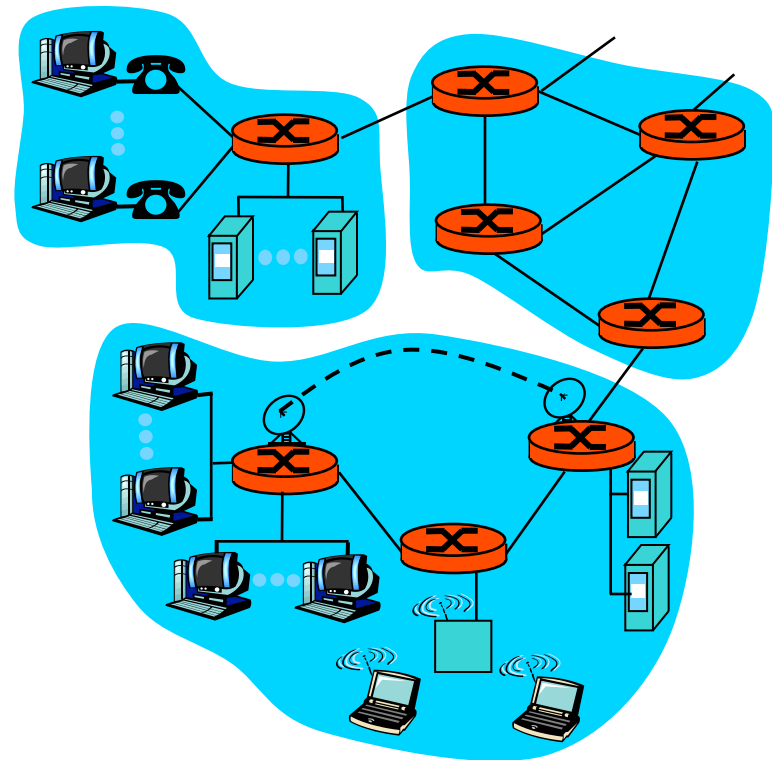


# Network Structure



# A closer look at network structure:

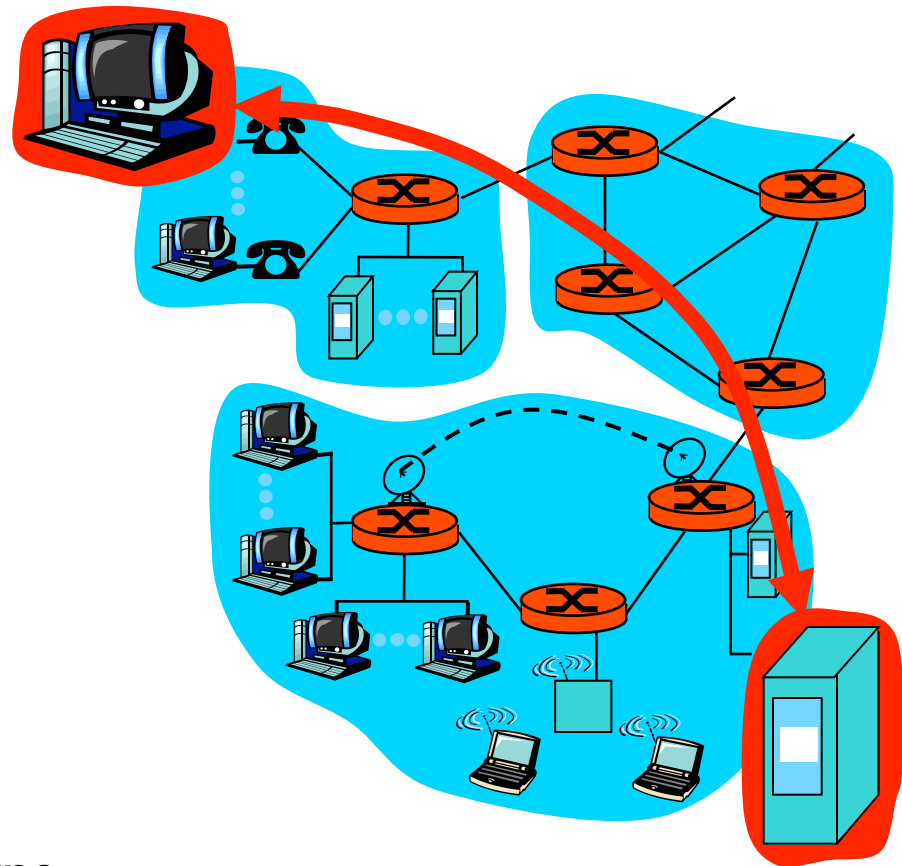
- ❖ **network edge:**  
applications and hosts
- ❖ **network core:**
  - routers
  - network of networks
- ❖ **access networks, physical media:**  
communication links





# The network edge:

- ❖ **end systems (hosts):**
  - run application programs
  - e.g. Web, email
  - at “edge of network”
- ❖ **client/server model**
  - client host requests, receives service from always-on server
  - e.g. Web browser/server; email client/server
- ❖ **peer-peer model:**
  - minimal (or no) use of dedicated servers
  - e.g. Gnutella, KaZaA, Skype





# Network edge: connection-oriented service

**Goal:** data transfer between end systems

- ❖ ***handshaking:*** setup (prepare for) data transfer ahead of time
  - Hello, hello back human protocol
  - ***set up "state"*** in two communicating hosts
- ❖ **TCP - Transmission Control Protocol**
  - Internet's connection-oriented service

**TCP service** [RFC 793]

- ❖ ***reliable, in-order*** byte-stream data transfer
  - **loss:**  
acknowledgements and retransmissions
- ❖ ***flow control:***
  - sender won't overwhelm receiver
- ❖ ***congestion control:***
  - senders "slow down sending rate" when network congested





# Network edge: connectionless service

**Goal:** data transfer  
between end systems

- same as before!
- ❖ **UDP** - User Datagram Protocol [RFC 768]:
  - connectionless
  - unreliable data transfer
  - no flow control
  - no congestion control

## **App's using TCP:**

- ❖ HTTP (Web), FTP (file transfer), Telnet (remote login), SMTP (email)

## **App's using UDP:**

- ❖ streaming media, teleconferencing, DNS, Internet telephony

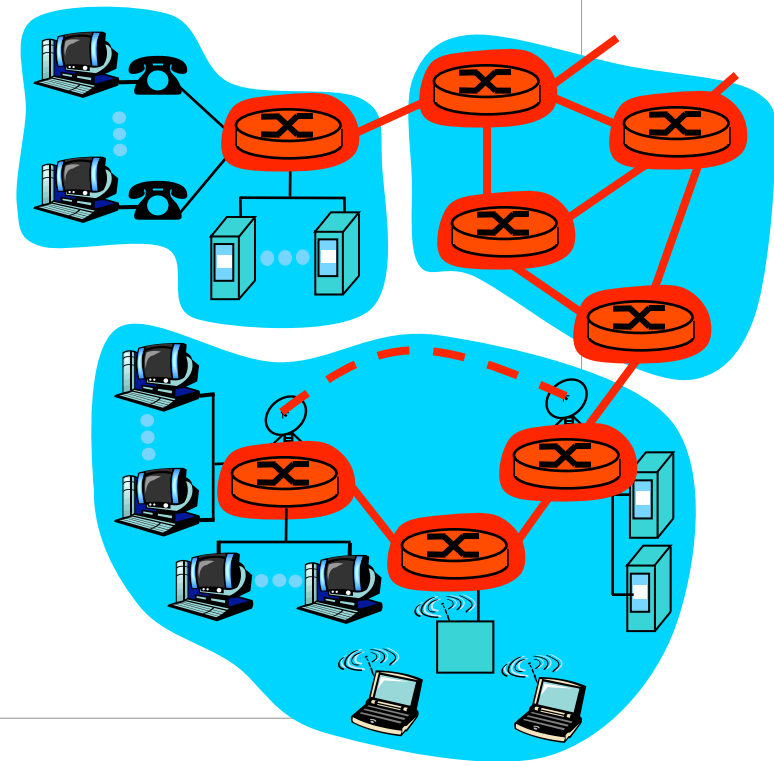


# The Network Core



# The Network Core

- ❖ mesh of interconnected routers
- ❖ the fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete “chunks”

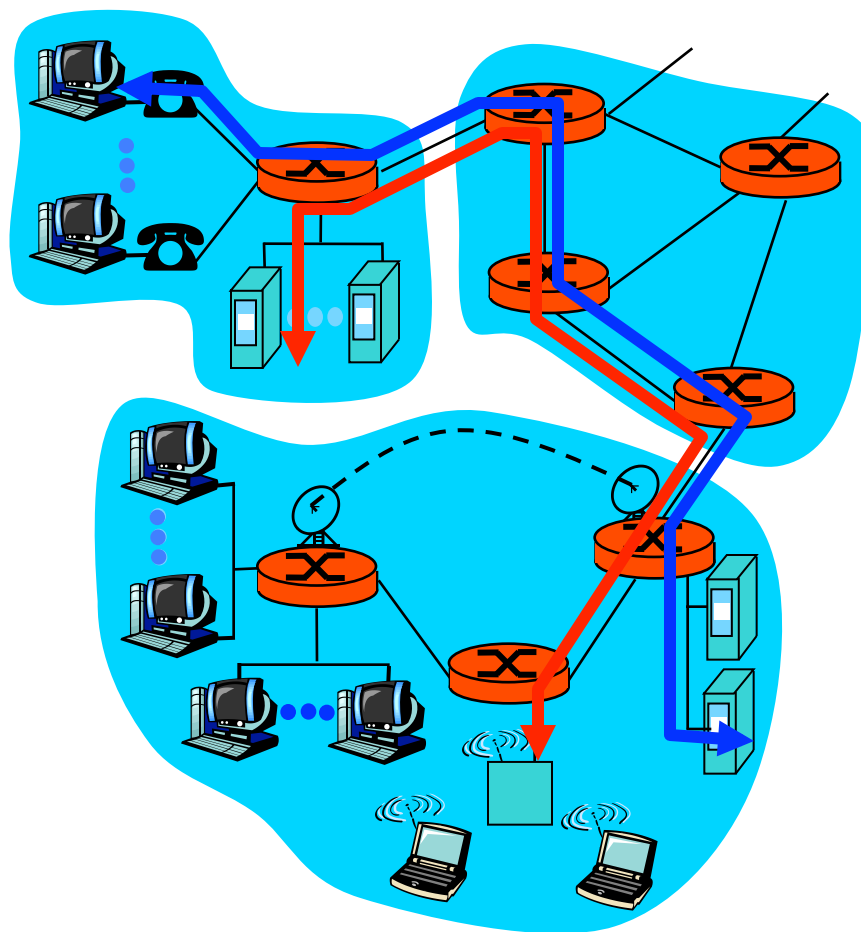




# Network Core: Circuit Switching

## End-end resources reserved for "call"

- ❖ link bandwidth, switch capacity
- ❖ dedicated resources: no sharing
- ❖ circuit-like (guaranteed) performance
- ❖ call setup required





# Network Core: Circuit Switching

**network resources  
(e.g., bandwidth)**

**divided into  
“pieces”**

- ❖ **pieces allocated to calls**
- ❖ **resource piece *idle* if not used by owning call (*no sharing*)**

❖ **dividing link bandwidth into “pieces”**

- **frequency division**
- **time division**

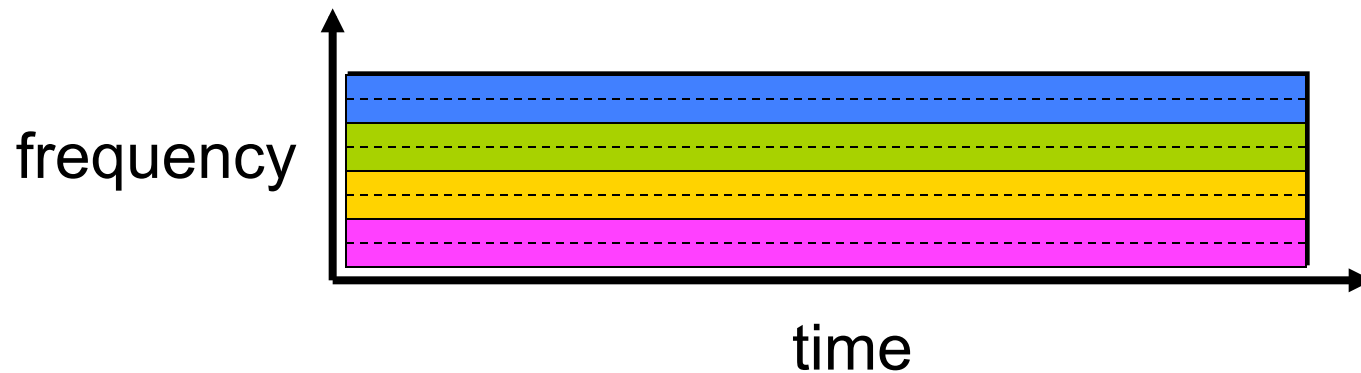


# Circuit Switching: FDM and TDM

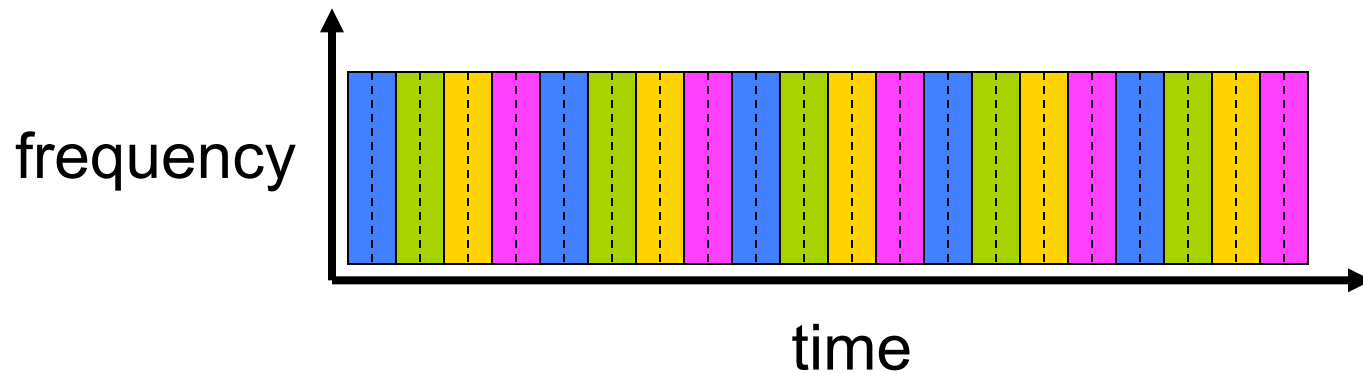
Example:

4 users

FDM



TDM





# Numerical example

- ❖ **How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?**
  - **All links are 1.536 Mbps**
  - **Each link uses TDM with 24 slots/sec**
  - **500 msec to establish end-to-end circuit**

**Let's work it out!**



# Another numerical example

- ❖ **How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?**
  - **All links are 1.536 Mbps**
  - **Each link uses FDM with 24 channels/frequencies**
  - **500 msec to establish end-to-end circuit**

**Let's work it out!**





# Network Core: Packet Switching

**each end-end data stream  
divided into *packets***

- ❖ user A, B packets *share* network resources
- ❖ each packet uses full link bandwidth
- ❖ resources used as *needed*

Bandwidth division into  
“pieces”

Dedicated allocation

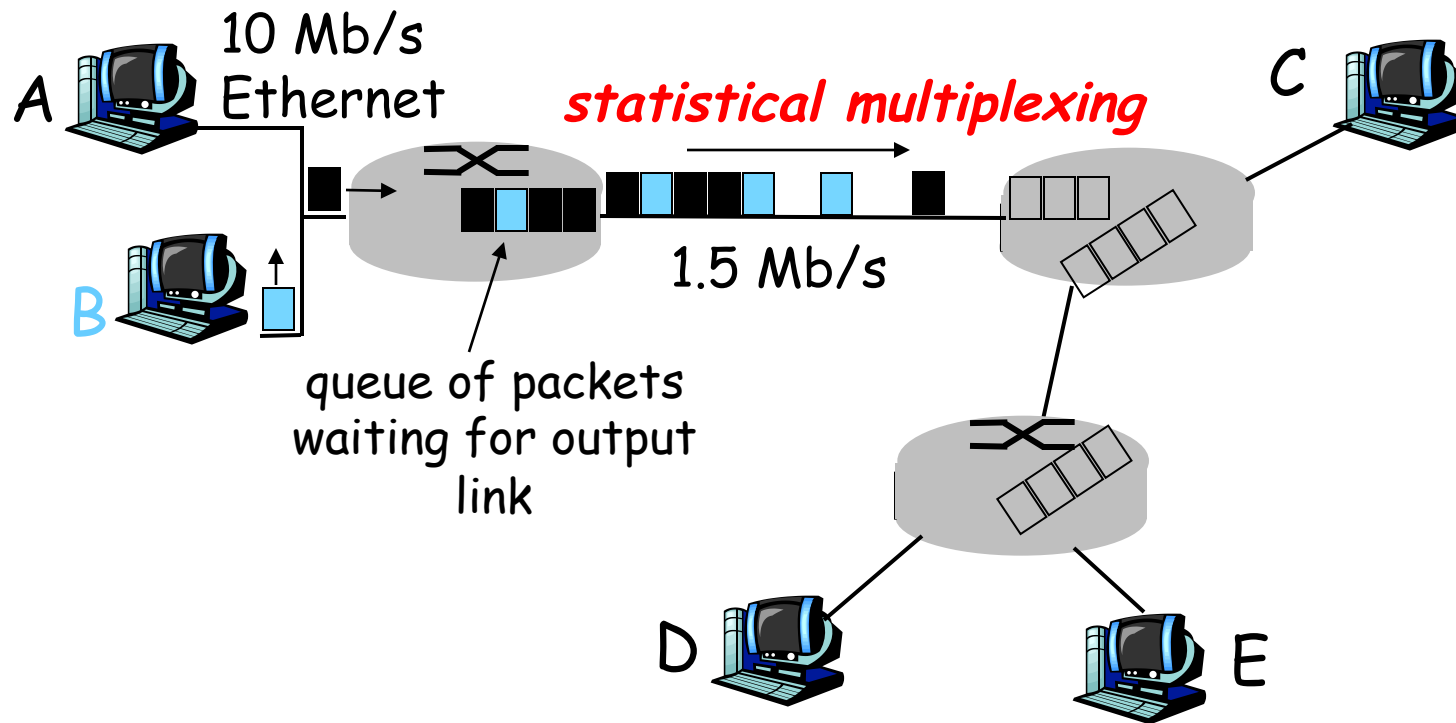
Resource reservation

**resource contention:**

- ❖ aggregate resource demand can exceed amount available
- ❖ congestion: packets queue, wait for link use
- ❖ store and forward: packets move one hop at a time
  - Node receives complete packet before forwarding



# Packet Switching: Statistical Multiplexing



Sequence of A & B packets does not have fixed pattern, shared on demand ➡ *statistical multiplexing*.

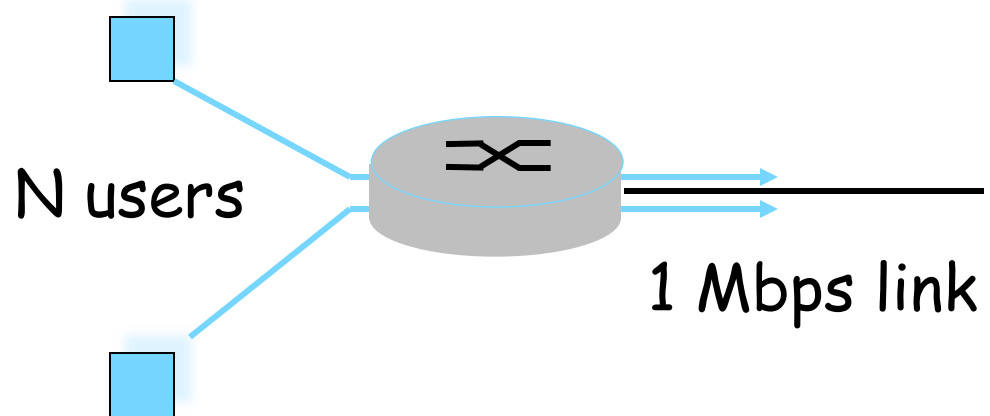
TDM: each host gets same slot in revolving TDM frame.



# Packet switching versus circuit switching

**Packet switching allows more users to use network!**

- ❖ **1 Mb/s link**
- ❖ **each user:**
  - 100 kb/s when "active"
  - active 10% of time
- ❖ **circuit-switching:**
  - 10 users
- ❖ **packet switching:**
  - with 35 users, probability  $> 10^{-4}$  active less than .0004



Q: how did we get value 0.0004?



# Packet switching versus circuit switching

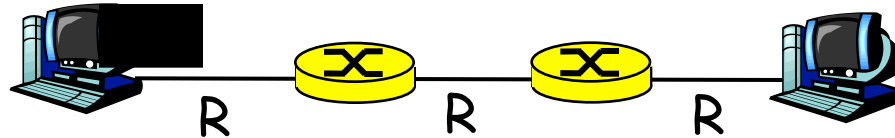
**Is packet switching a “slam dunk winner?”**

- ❖ **Great for bursty data**
  - resource sharing
  - simpler, no call setup
- ❖ **Excessive congestion:** packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- ❖ **Q: How to provide circuit-like behavior?**
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem

**Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?**



# Packet-switching: store-and-forward



- ❖ Takes  $L/R$  seconds to transmit (push out) packet of  $L$  bits on to link or  $R$  bps
- ❖ Entire packet must arrive at router before it can be transmitted on next link: **store and forward**
- ❖  $\text{delay} = 3L/R$  (assuming zero propagation delay)

## Example:

- ❖  $L = 7.5$  Mbits
- ❖  $R = 1.5$  Mbps
- ❖  $\text{delay} = 15$  sec

} more on delay shortly ...

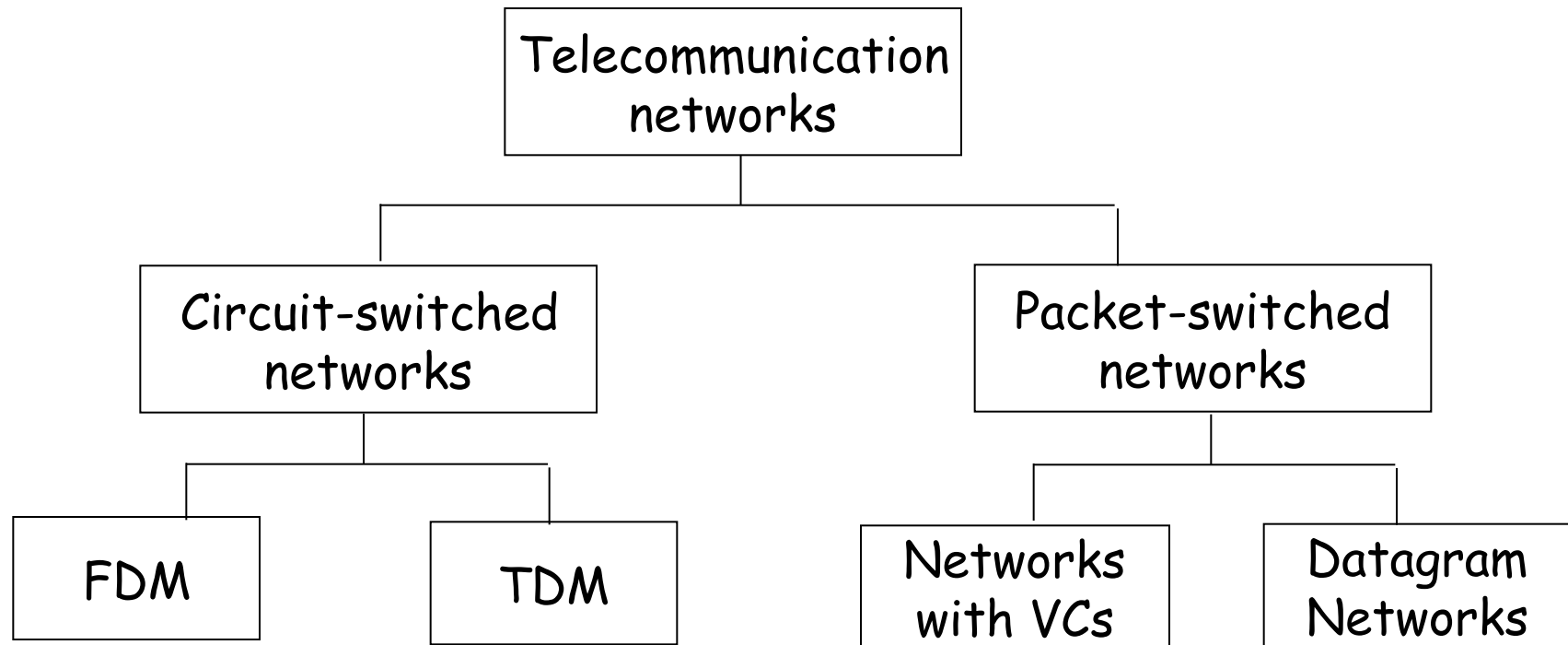


# Packet-switched networks: forwarding

- ❖ **Goal:** move packets through routers from source to destination
  - we'll study several path selection (i.e. routing) algorithms
- ❖ **datagram network:**
  - *destination address* in packet determines next hop
  - routes may change during session
  - analogy: driving, asking directions
- ❖ **virtual circuit network:**
  - each packet carries tag (virtual circuit ID), tag determines next hop
  - fixed path determined at *call setup time*, remains fixed thru call
  - ***routers maintain per-call state***



# Network Taxonomy

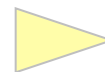
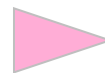
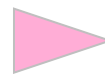


- Datagram network is not either connection-oriented or connectionless.
- Internet provides both connection-oriented (TCP) and connectionless services (UDP) to apps.



# 计算机网络组成和分类

- ❖ 主机系统和网络系统
- ❖ 计算机网络与分布式系统
- ❖ Internet
- ❖ 计算机网络的分类







# 网络分类

- ❖ 按技术分
  - 广播式网络
  - 点到点网络
- ❖ 按规模分
  - 局域网
  - 城域网
  - 广域网
  - 互联网
- ❖ 按传输介质分
  - 有线网
  - 无线网



# 网络分类 (续)

- ❖ 按拓扑结构分
  - 总线
  - 环形
  - 网状
  - 星形
  
- ❖ 按用户类型分
  - 专用网
  - 公共网



# 局域网 LAN (Local Area Network)

- ❖ 地域：覆盖范围较小
- ❖ 传输技术：
  - 总线型 IEEE 802.3(以太网)CSMA/CD 10M
  - 总线型 IEEE 802.4(令牌总线)10M
  - 环 型 IEEE 802.5(IBM令牌环)4M 16M



# 城域网 MAN (Metropolitan Area Network)

## 大型的LAN

- 私有网络 —— 一个连锁超市，有十几个或几十个分布在市内
- 公用网络 —— 上海科技网



# 广域网 WAN (Wide Area Network)

## 跨越地域较大的网络

- 主机 (host) 端点系统 (end system)
- 通信子网 (communication subnet)  
简称子网
- 资源子网 (resource subnet)

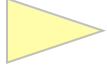


# 互联网通常由 路由器联接的LAN组成

- ❖ 通信子网通常为点到点子网  
(point-to-point)
- ❖ 其传输机理为存储-转发  
(store-and-forward)
- ❖ 其传输方式为分组交换  
(packet-switched)



# 第1章 概述

1. 计算机网络组成和分类 
2. 计算机网络参考模型 
3. 数字通信的基本概念 



# 计算机网络参考模型

❖ ISO/OSI参考模型



❖ TCP/IP参考模型





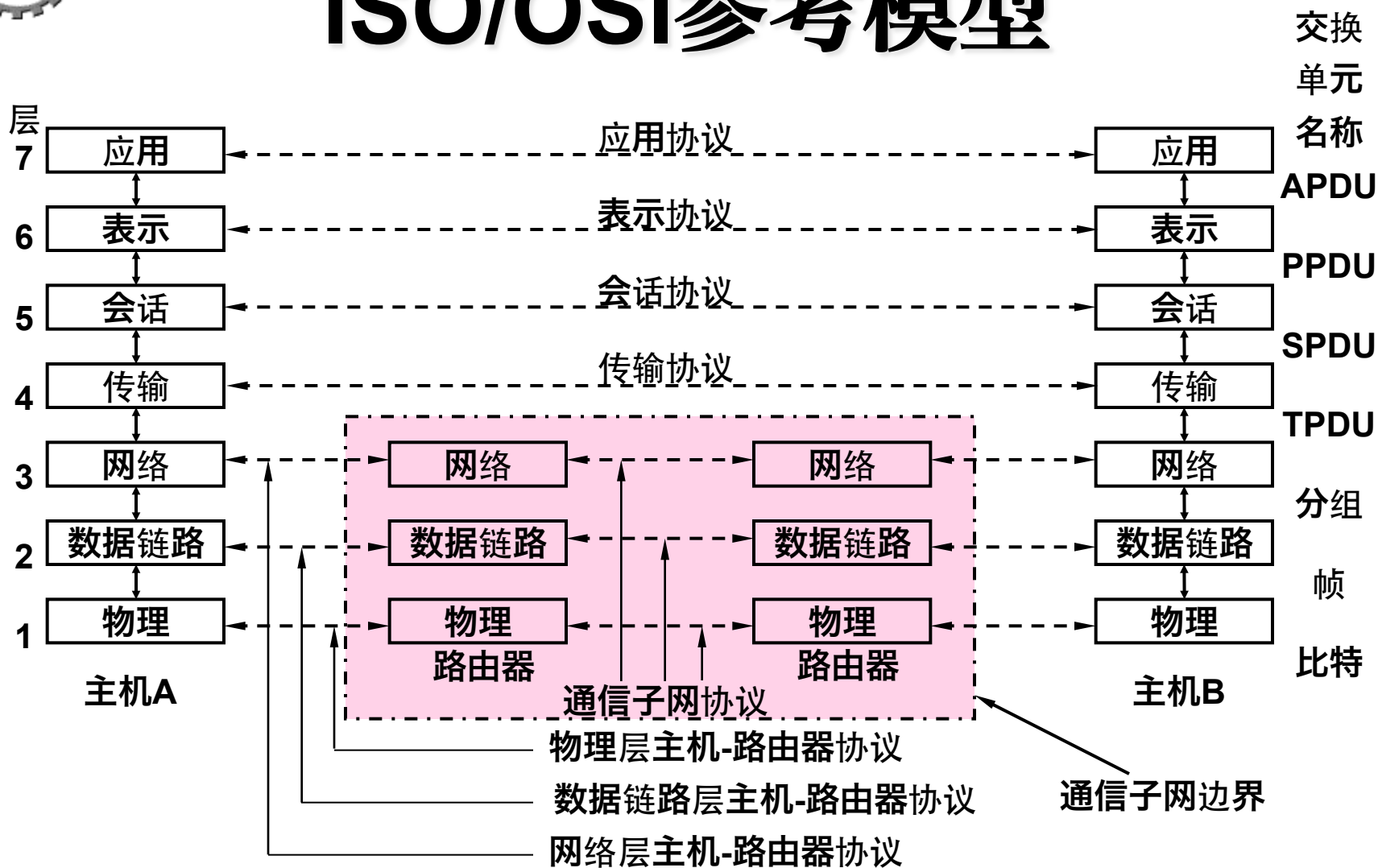


# 计算机网络的体系结构的形成

- ❖ 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。



# ISO/OSI参考模型



Tnbm P39 Fig. 1-20 OSI参考模型



# 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。



# 开放系统互连参考模型

## OSI/RM

- ❖ 只要遵循 OSI 标准，一个系统就可以和位于世界上任何地方的、也遵循这同一标准的其他任何系统进行通信。
- ❖ 在市场化方面 OSI 却失败了。
  - OSI 的专家们在完成 OSI 标准时没有商业驱动力；
  - OSI 的协议实现起来过分复杂，且运行效率很低；
  - OSI 标准的制定周期太长，因而使得按 OSI 标准生产的设备无法及时进入市场；
  - OSI 的层次划分并也不太合理，有些功能在多个层次中重复出现。



# 两种国际标准

- ❖ 法律上的国际标准 OSI 并没有得到市场的认可。
- ❖ 是非国际标准 TCP/IP 现在获得了最广泛的应用。
  - TCP/IP 常被称为事实上的国际标准。



# TCP/IP五层协议的体系结构

- ❖ **TCP/IP 是四层的体系结构：应用层、传输层、互联网层和主机至网络层。**
- ❖ **最下面的主机至网络层并没有具体内容。**
- ❖ **因此往往采取折中的办法，即综合 OSI 和 TCP/IP 的优点，采用一种只有五层协议的体系结构。**



# 五层协议的体系结构

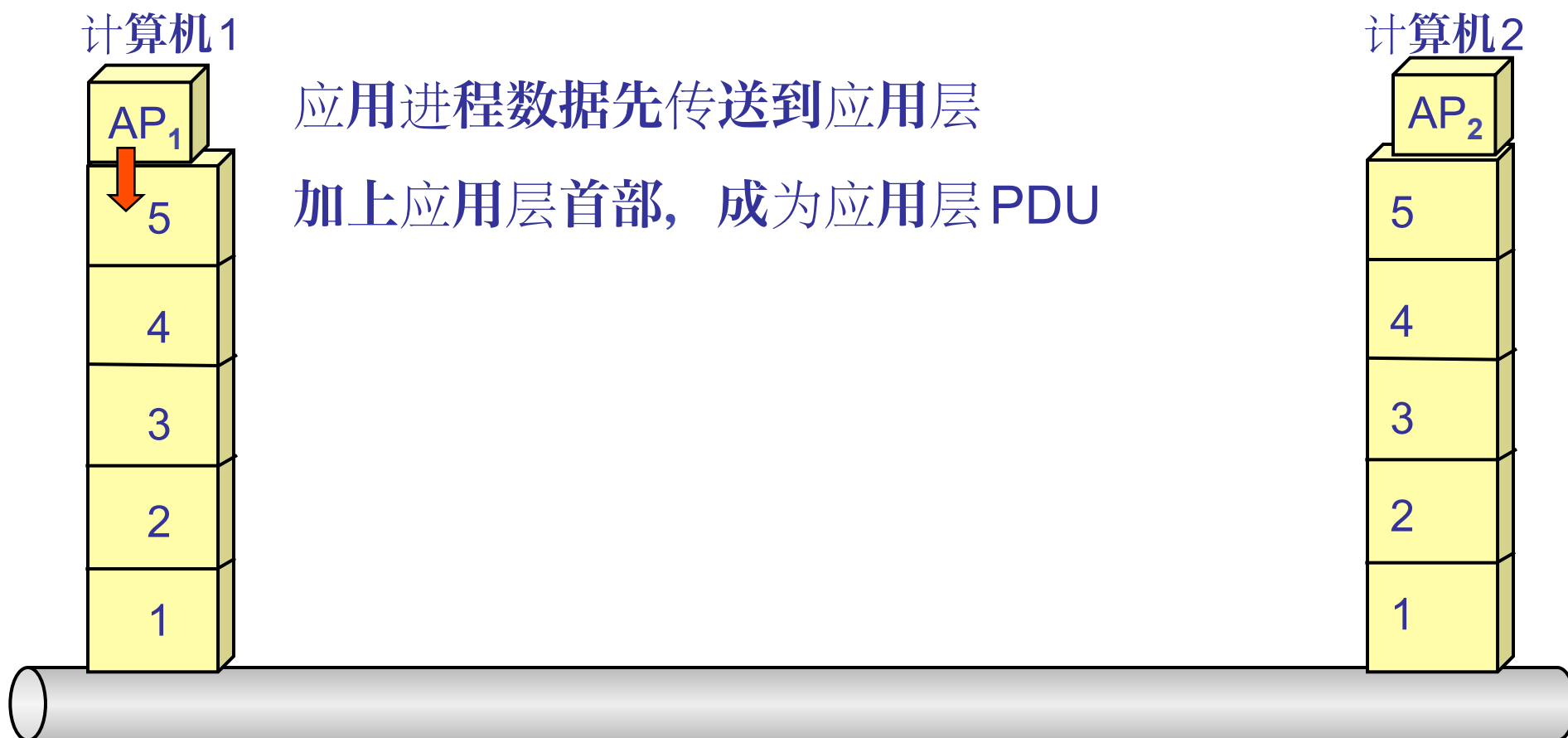


- ❖ 应用层(application layer)
- ❖ 传输层(transport layer)
- ❖ 网络层(network layer)
- ❖ 数据链路层(data link layer)
- ❖ 物理层(physical layer)



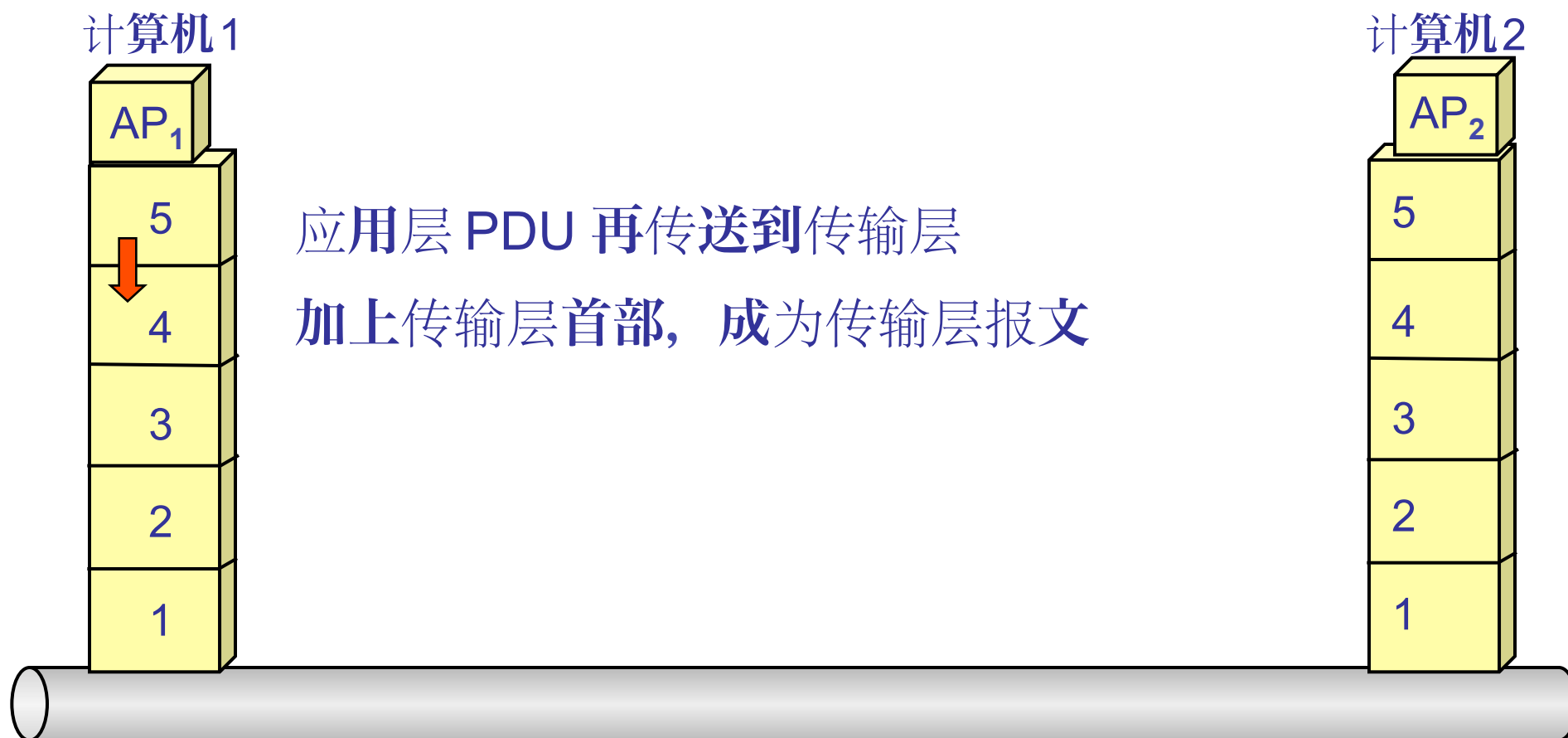


# 计算机1向计算机2发送数据



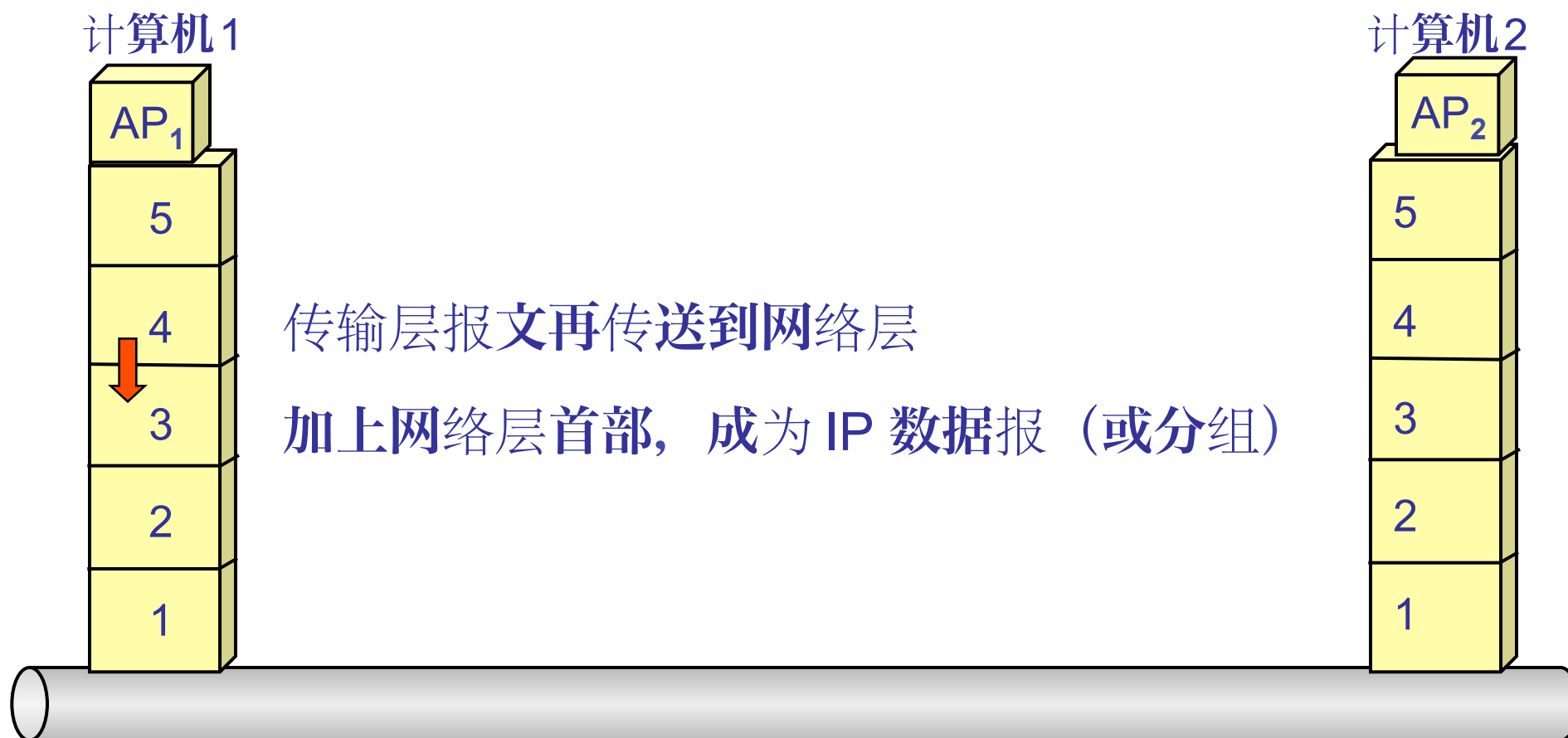


# 计算机1向计算机2发送数据



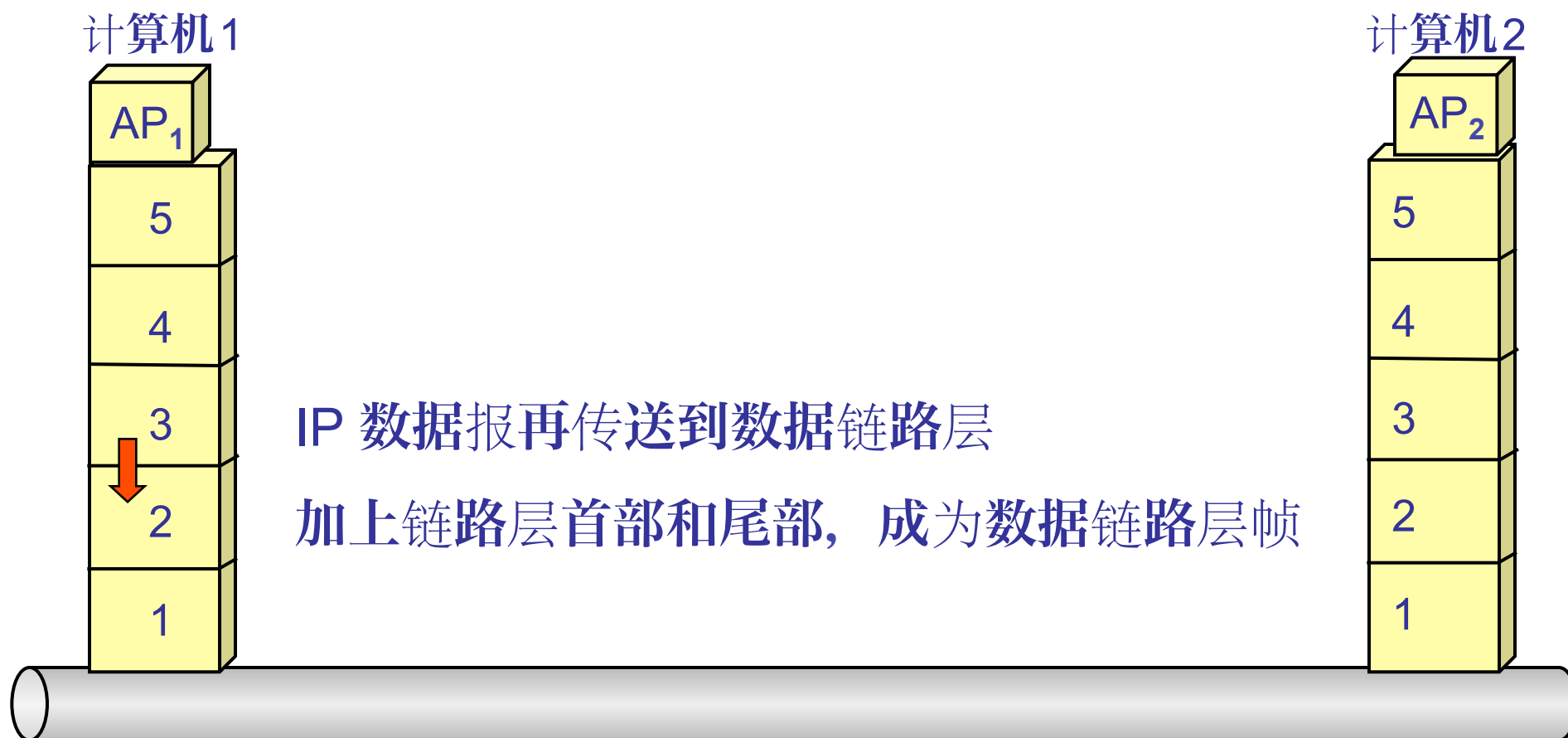


# 计算机1向计算机2发送数据



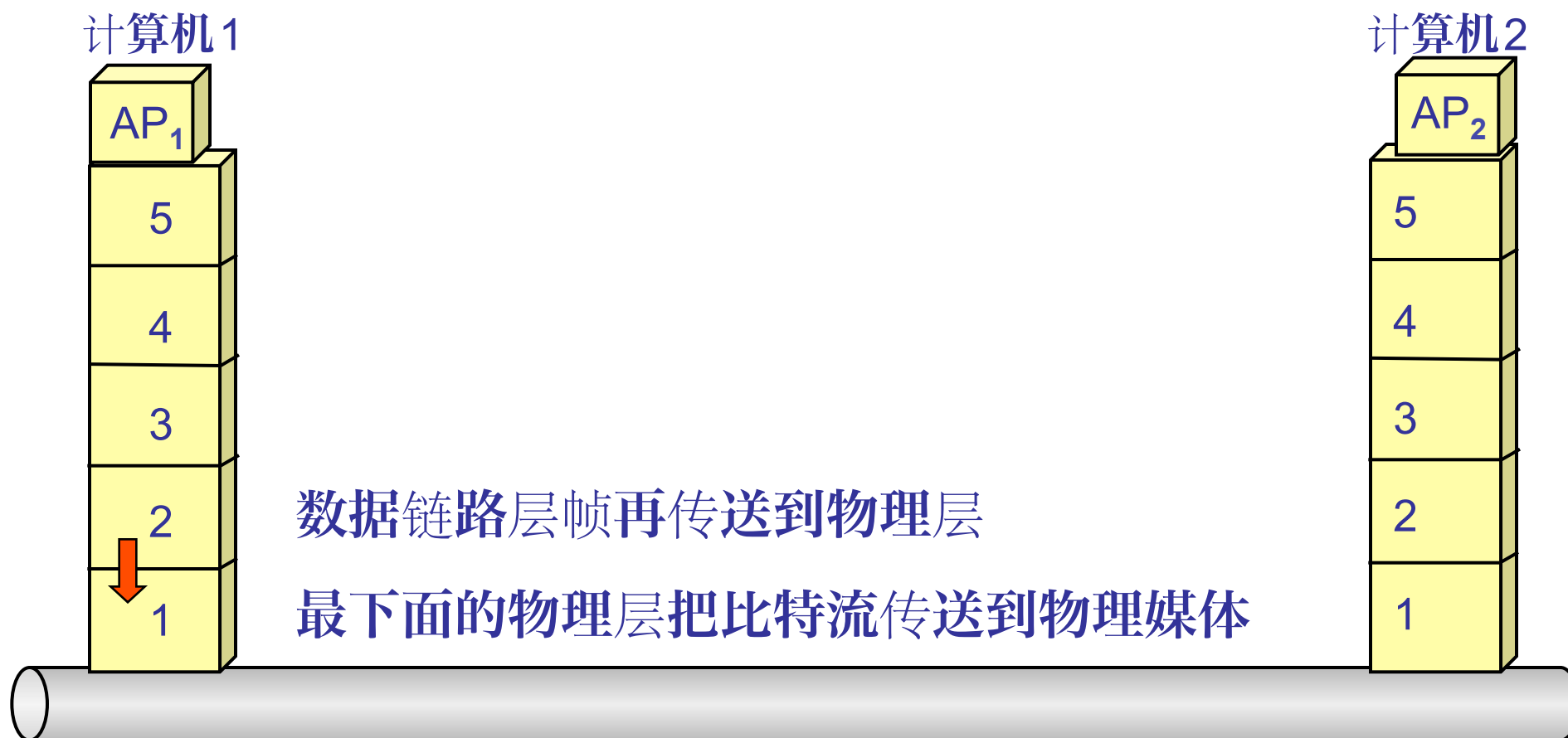


# 计算机1向计算机2发送数据



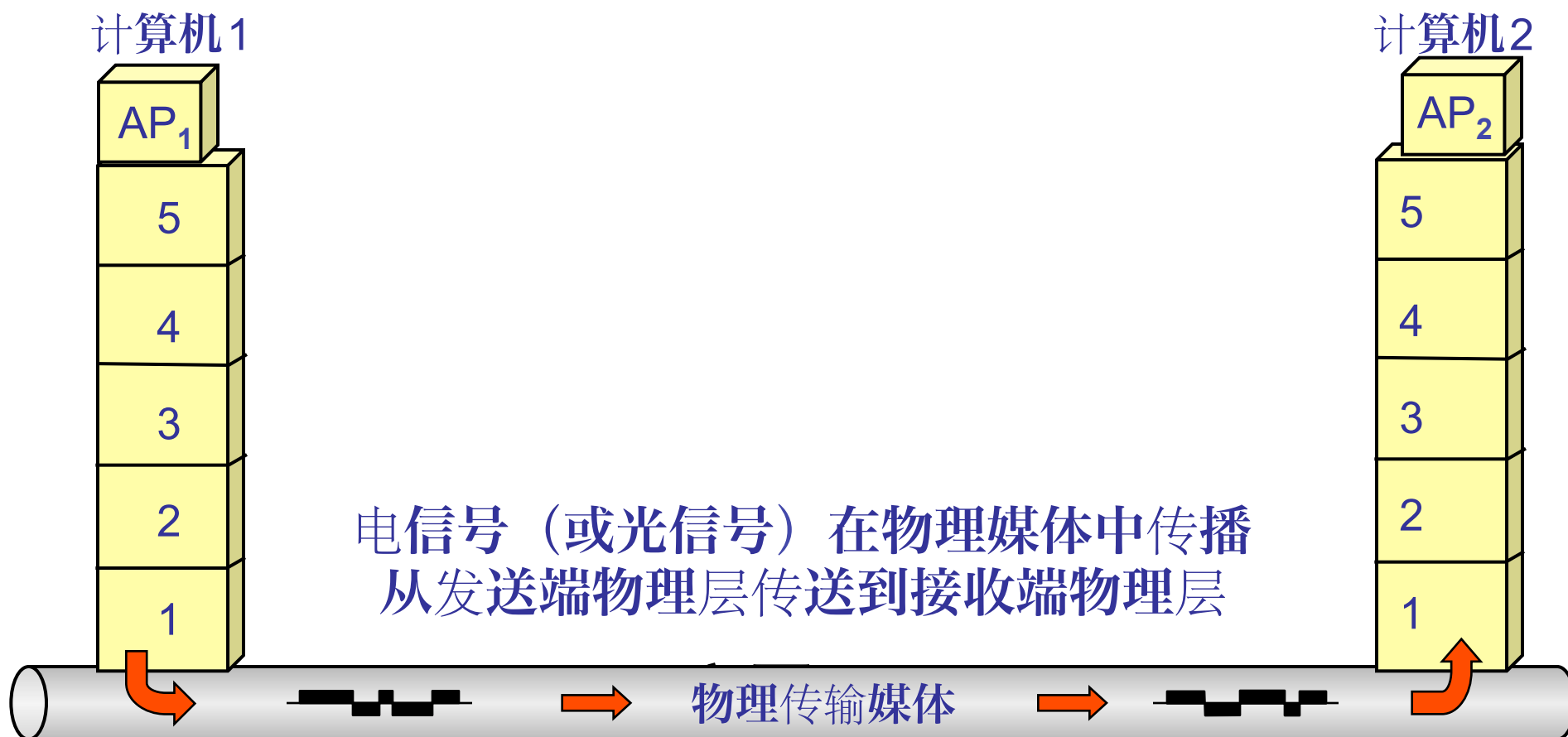


# 计算机1向计算机2发送数据



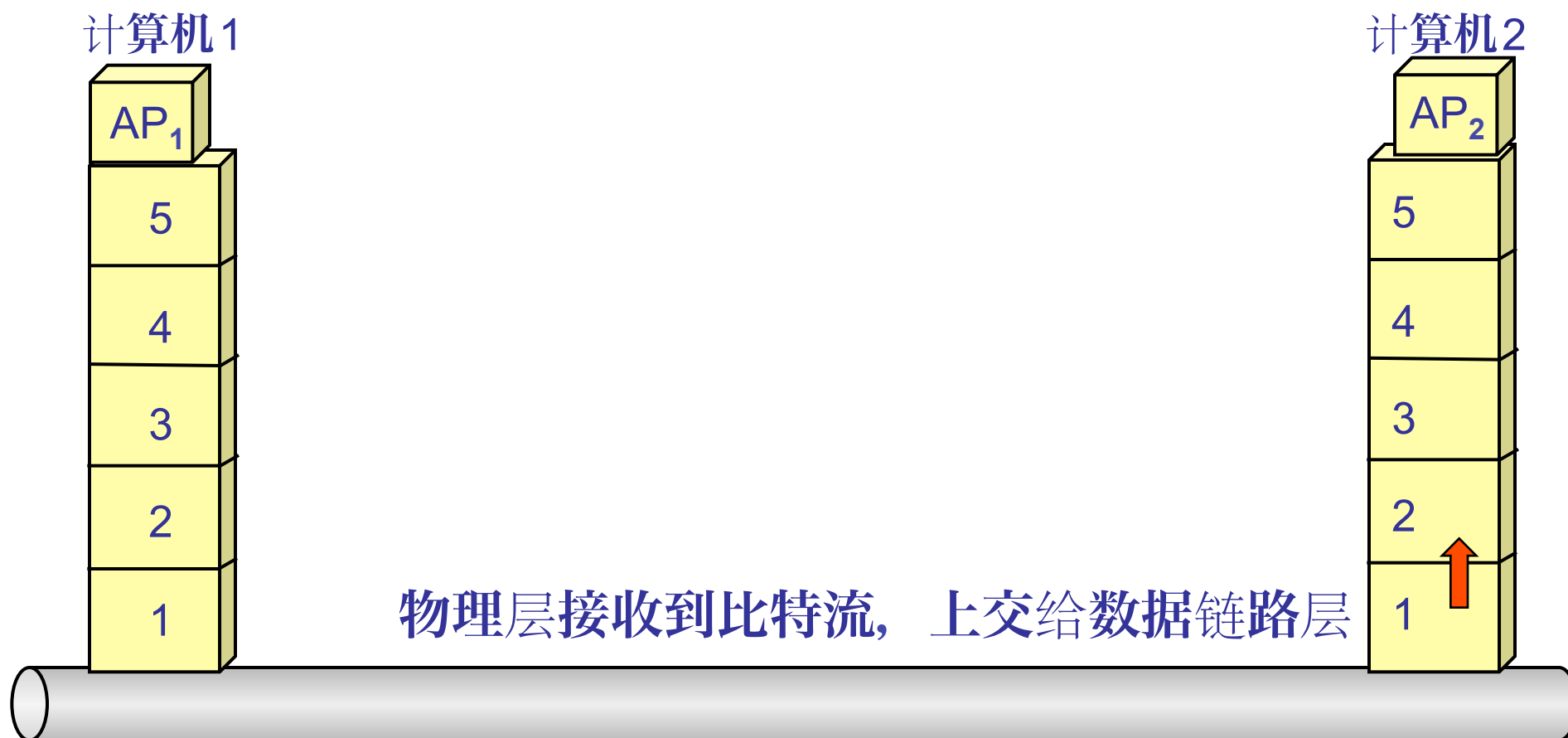


# 计算机1向计算机2发送数据



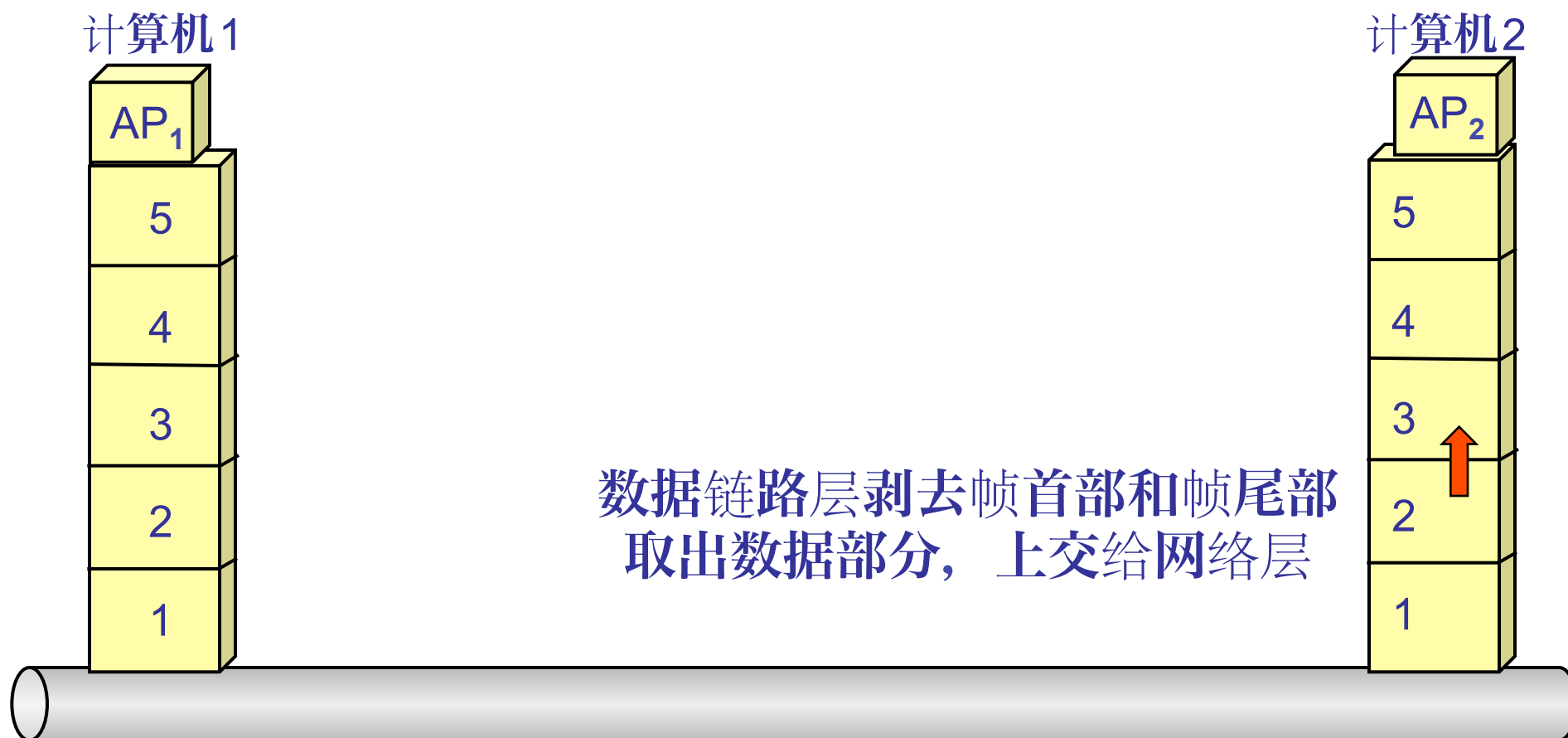


# 计算机1向计算机2发送数据





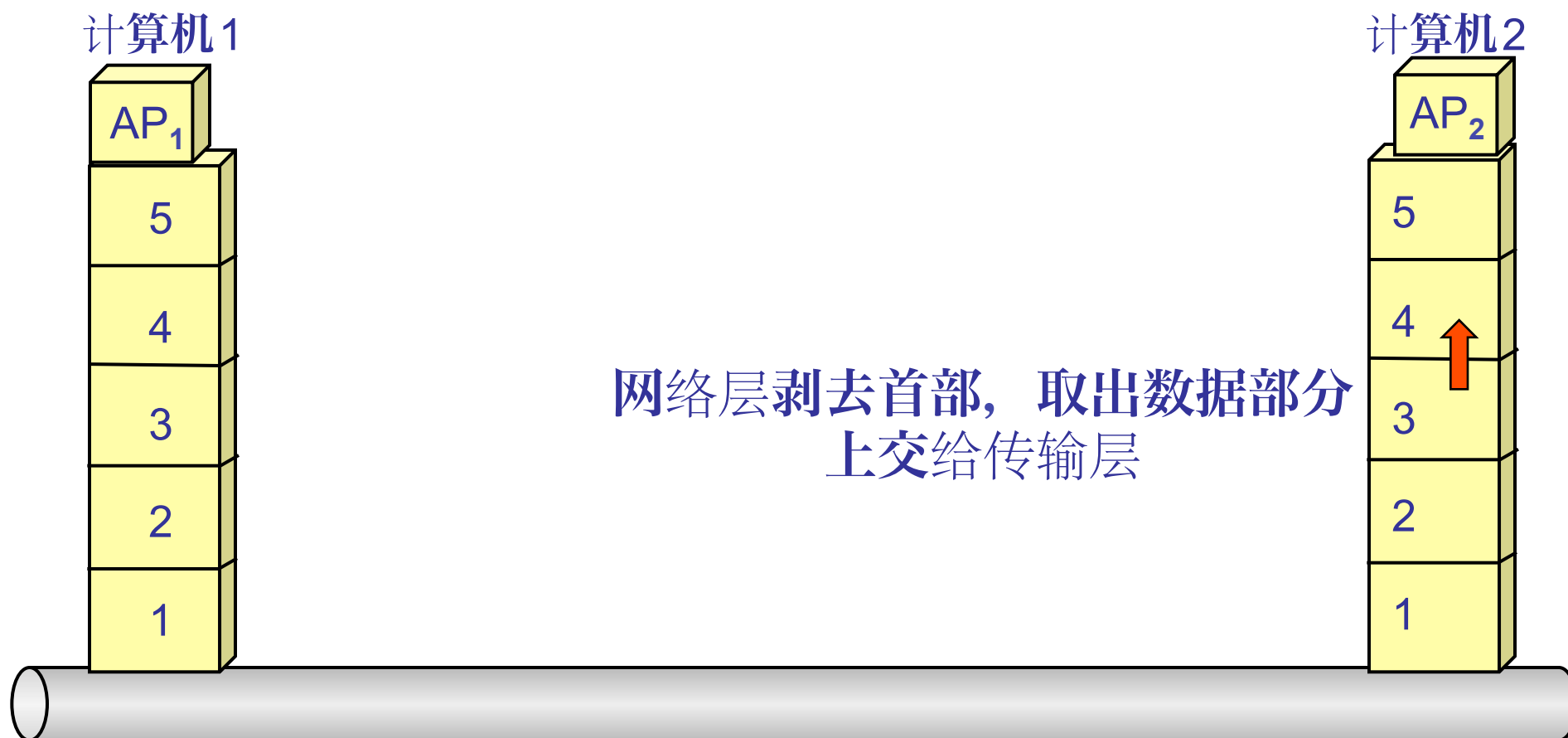
# 计算机1向计算机2发送数据





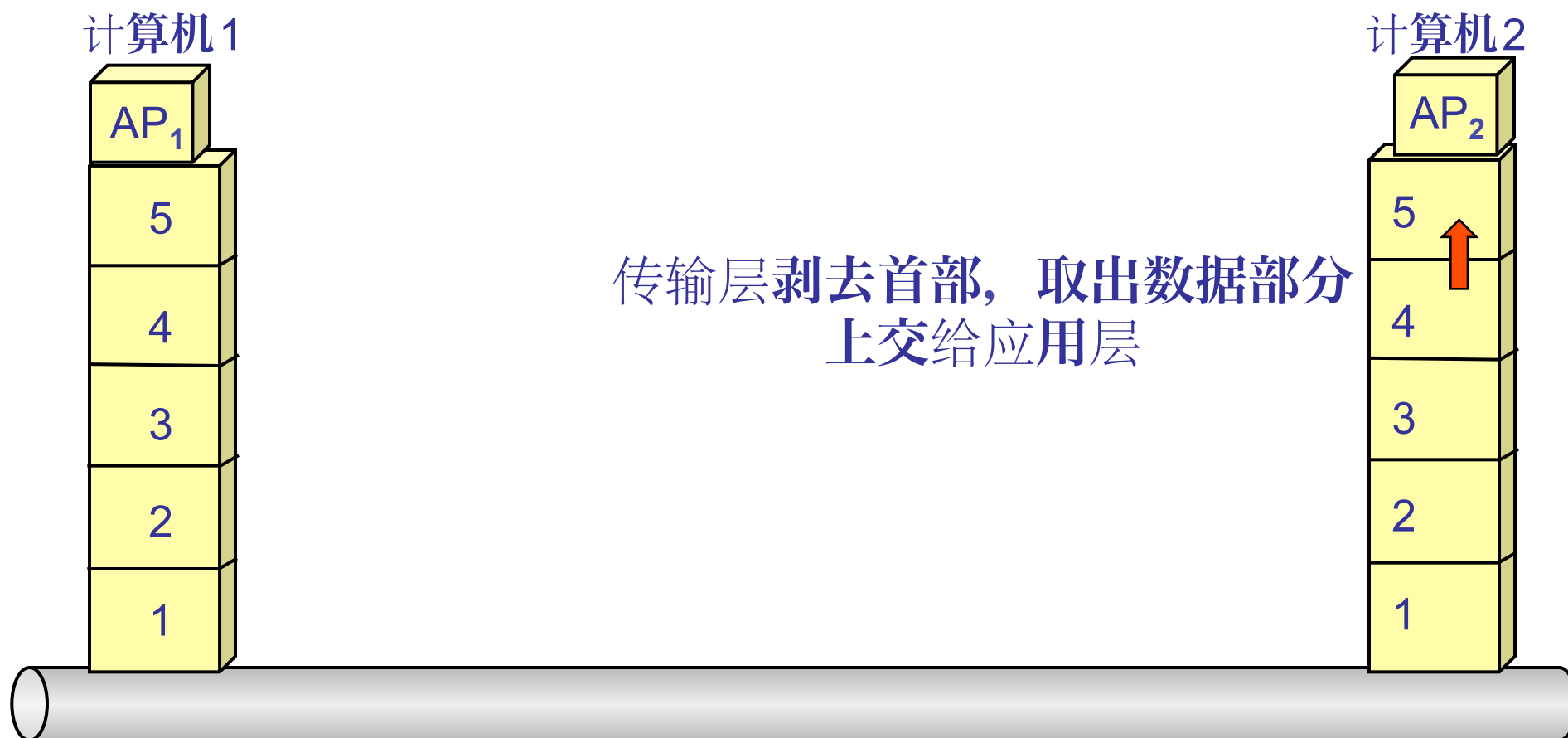


# 计算机1向计算机2发送数据



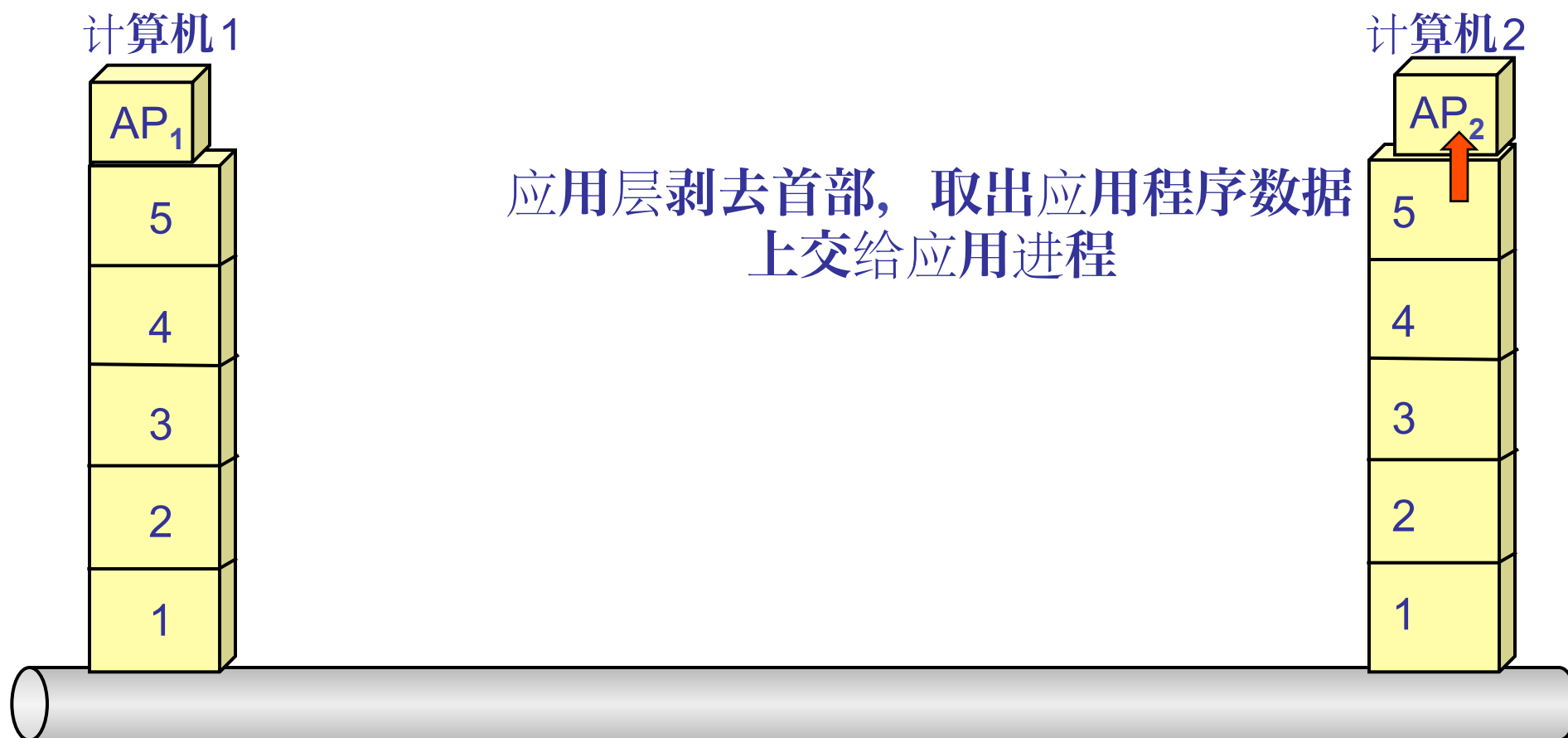


# 计算机1向计算机2发送数据



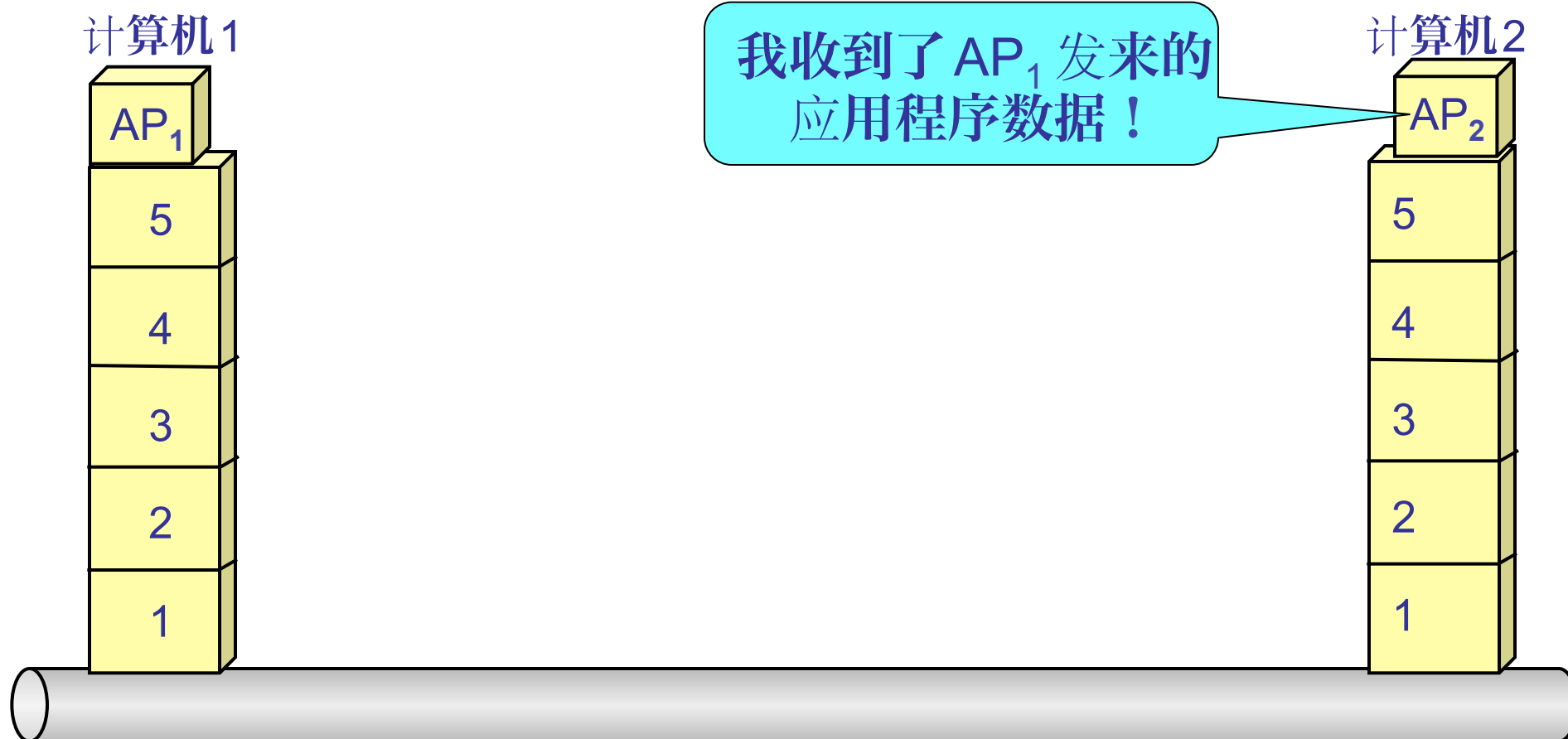


# 计算机1向计算机2发送数据



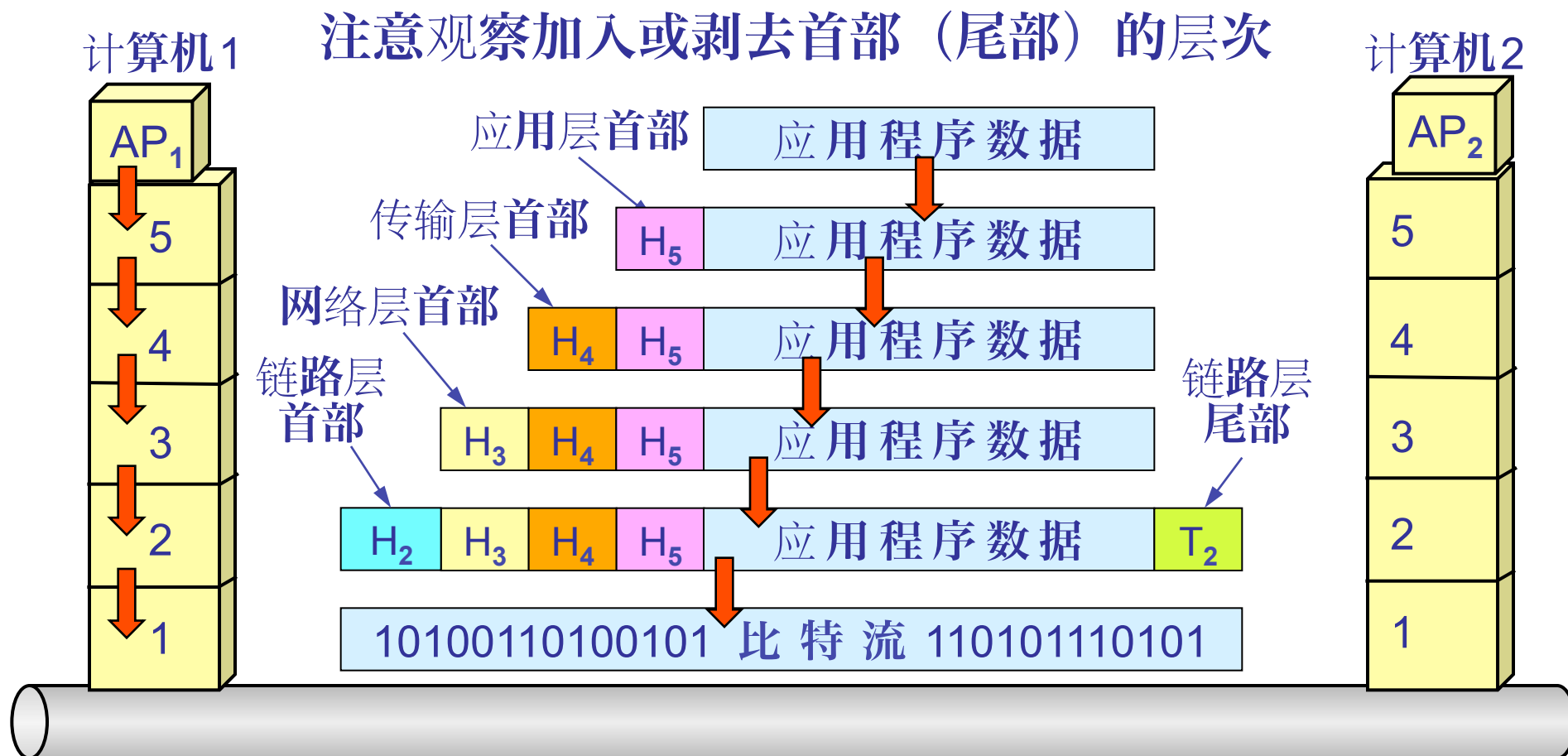


# 计算机1向计算机2发送数据



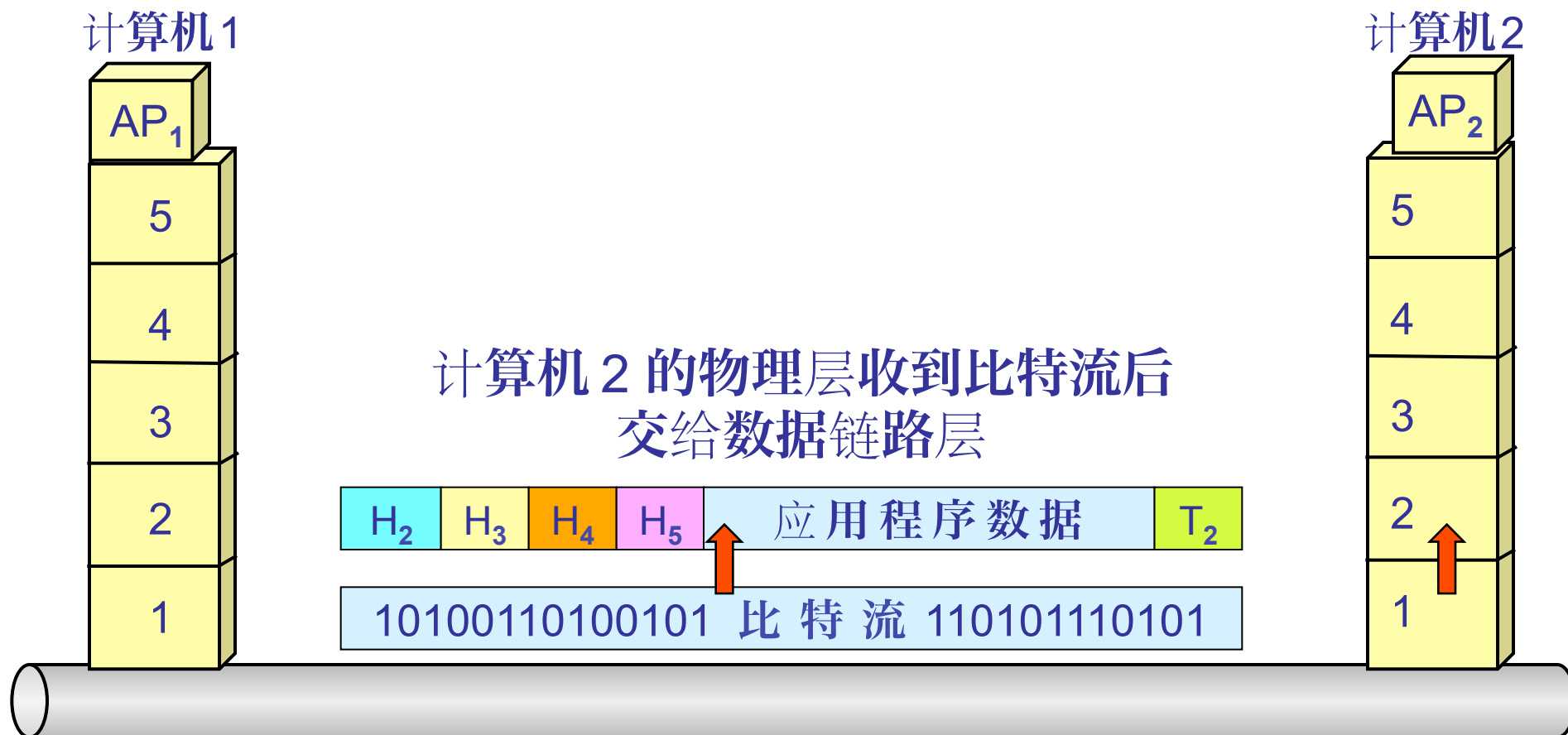


# 计算机1向计算机2发送数据



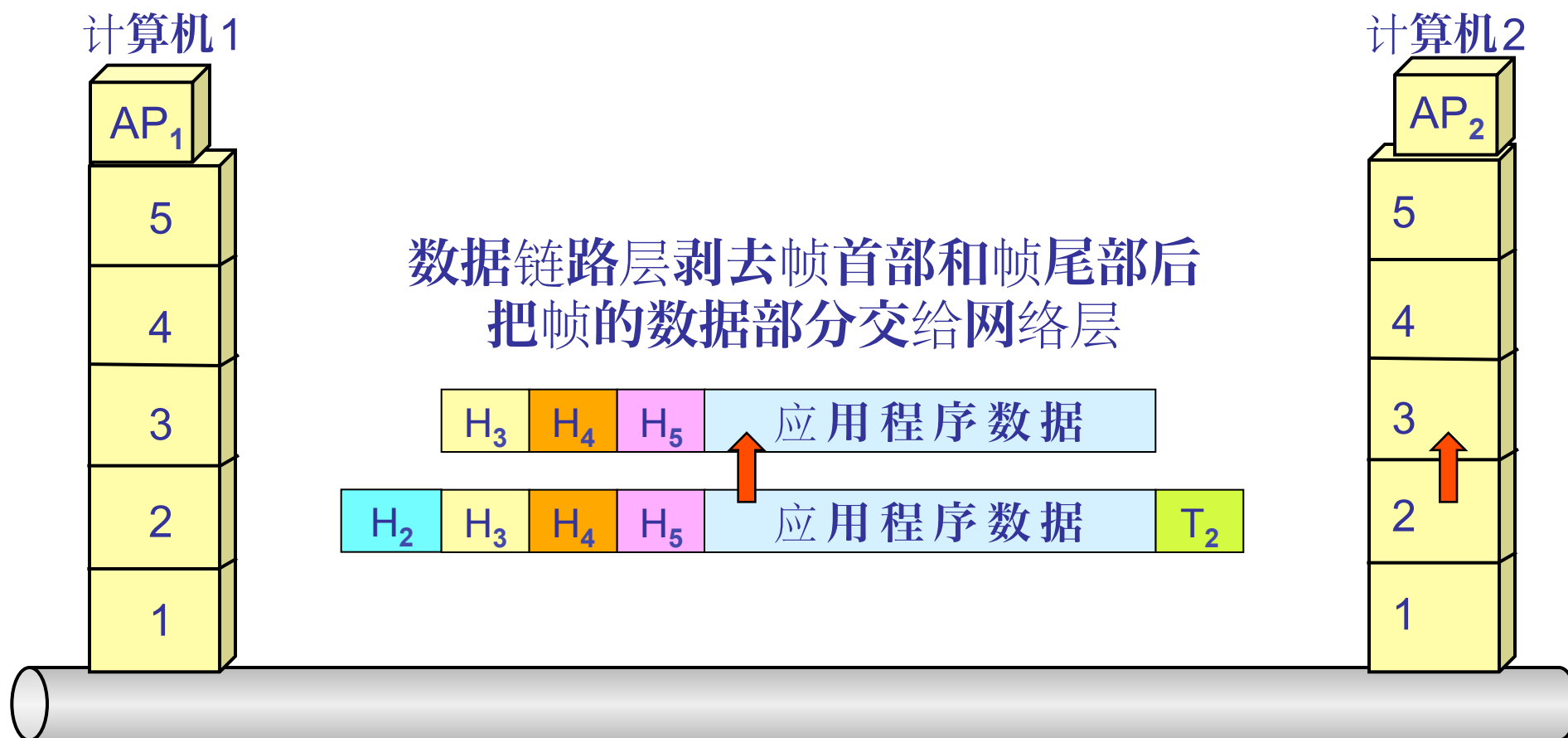


# 计算机1向计算机2发送数据



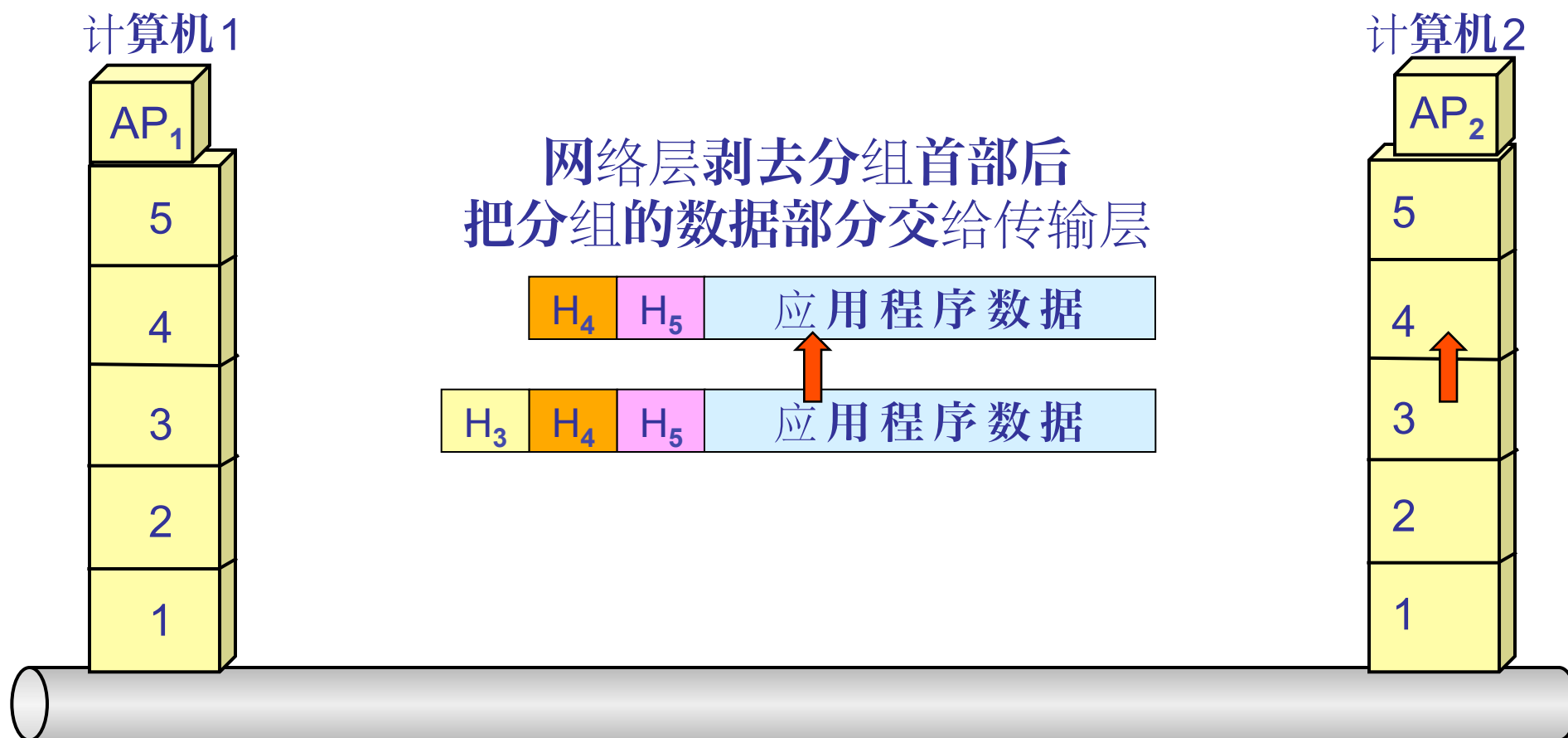


# 计算机1向计算机2发送数据





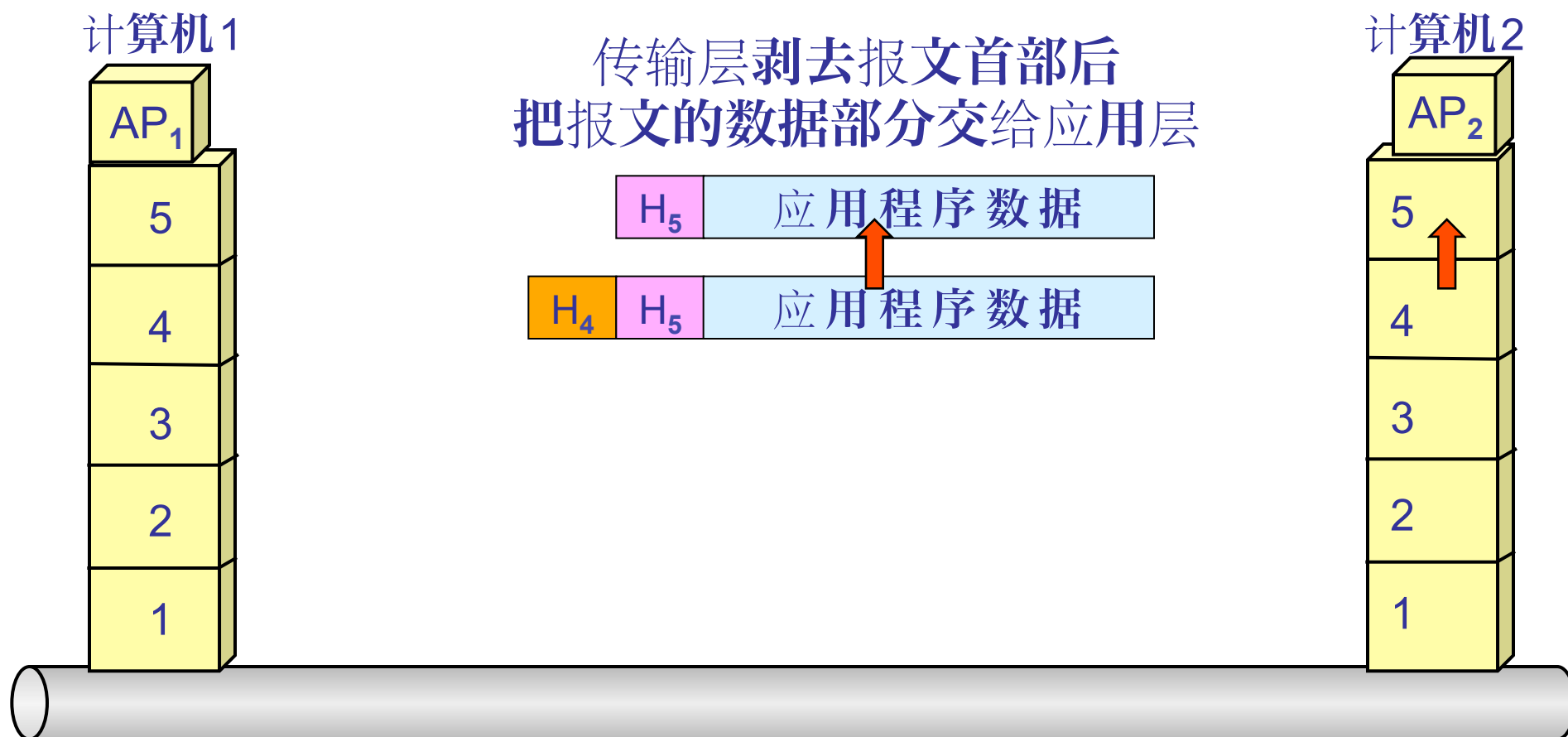
# 计算机1向计算机2发送数据





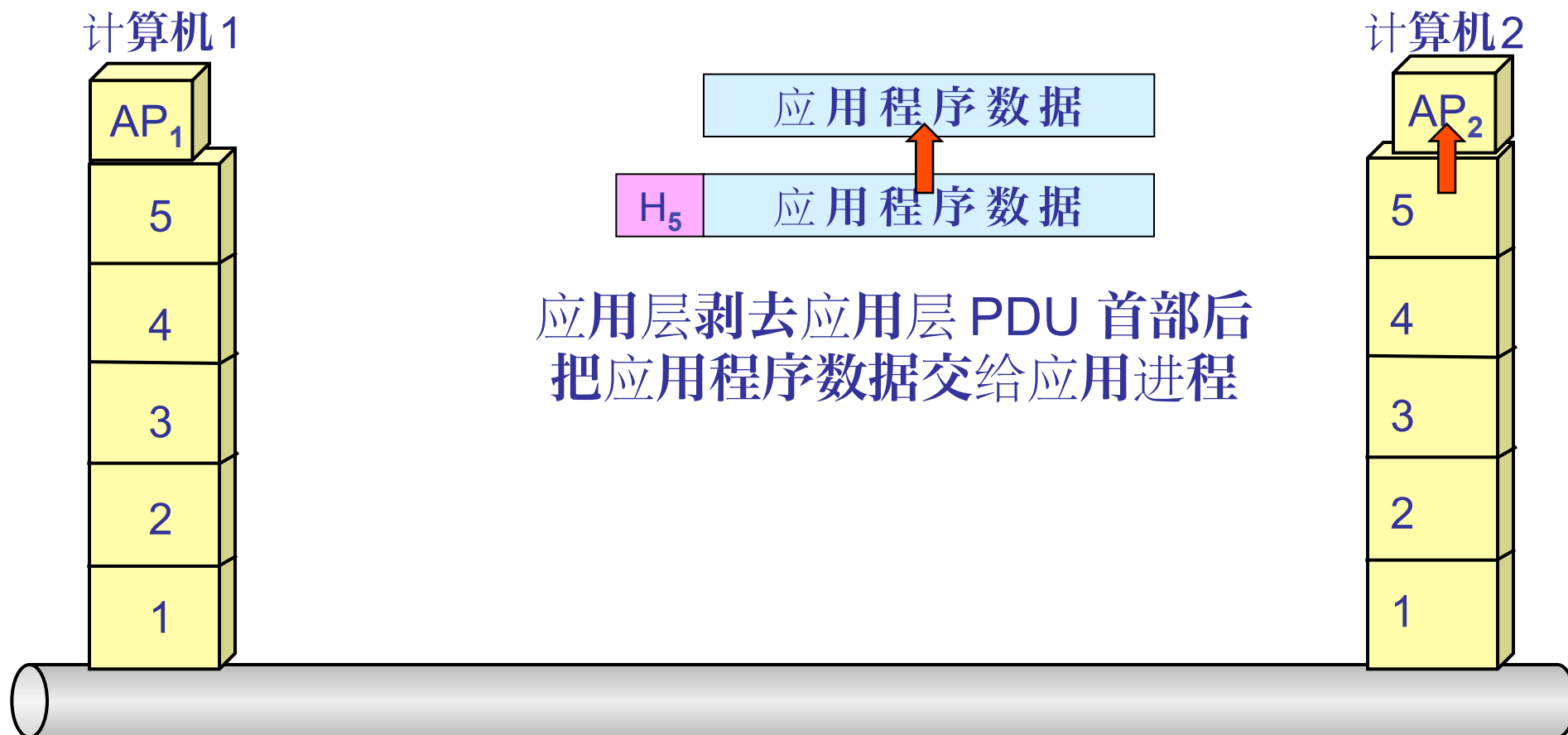


# 计算机1向计算机2发送数据



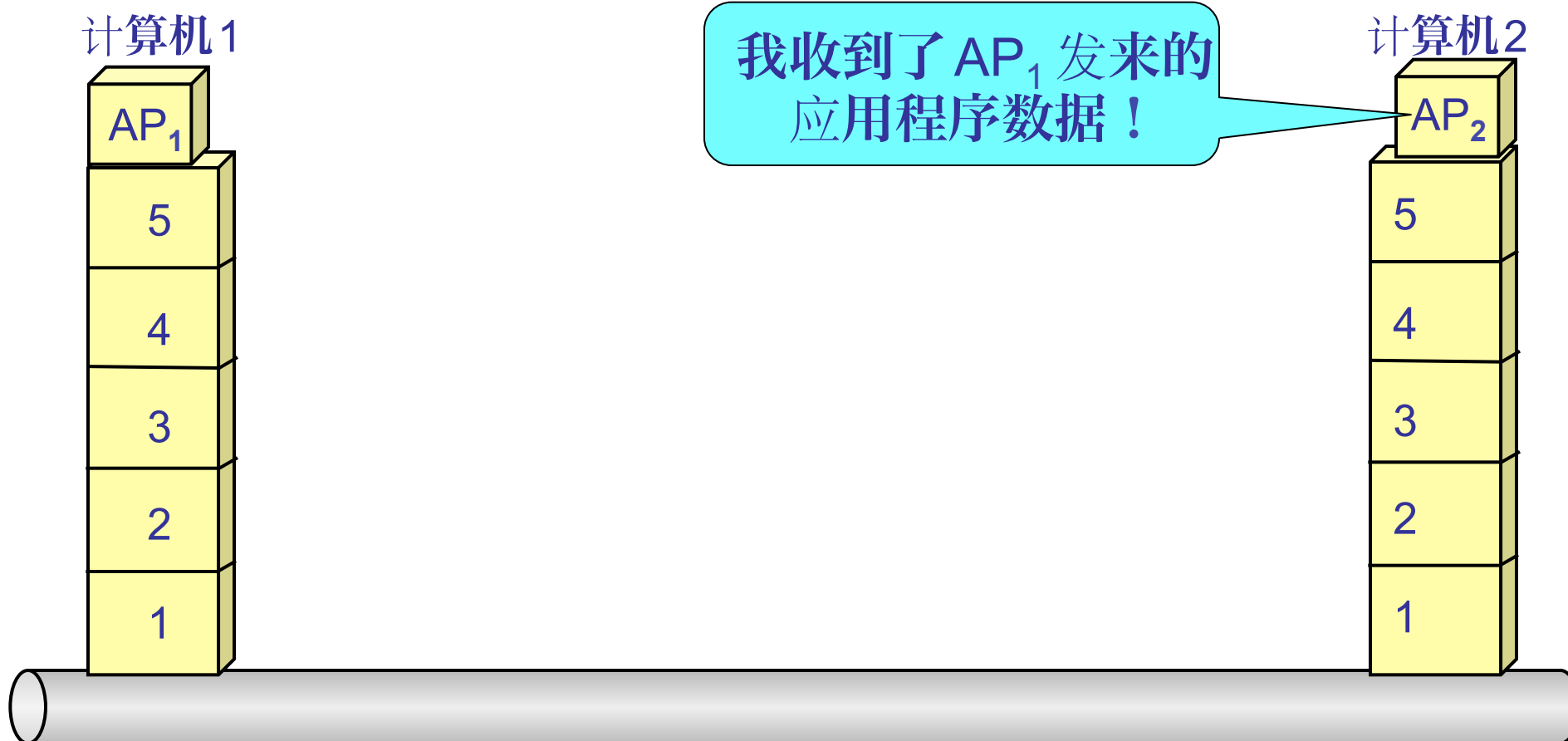


# 计算机1向计算机2发送数据





# 计算机1向计算机2发送数据





# 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**进行通信的规则的组合。
- ❖ 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。

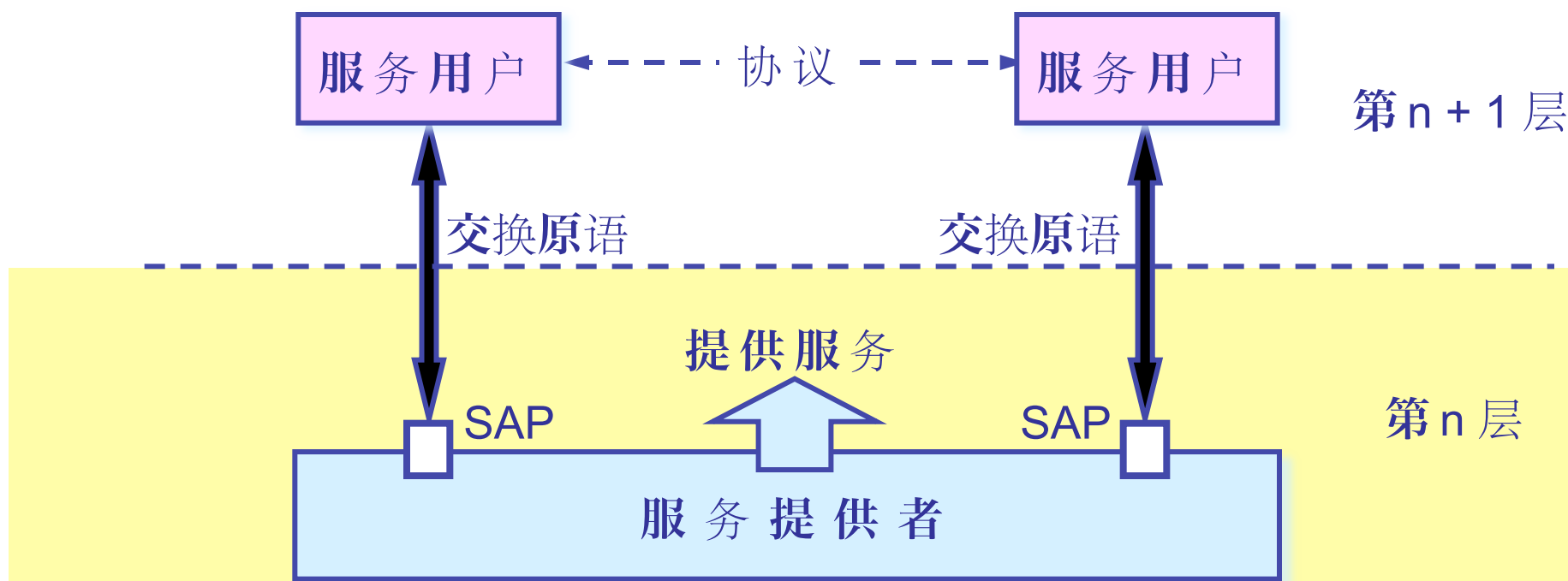


# 实体、协议、服务和 服务访问点（续）

- ❖ 本层的服务用户只能看见服务而无法看见下面的协议。
- ❖ 下面的协议对上面的服务用户是**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)



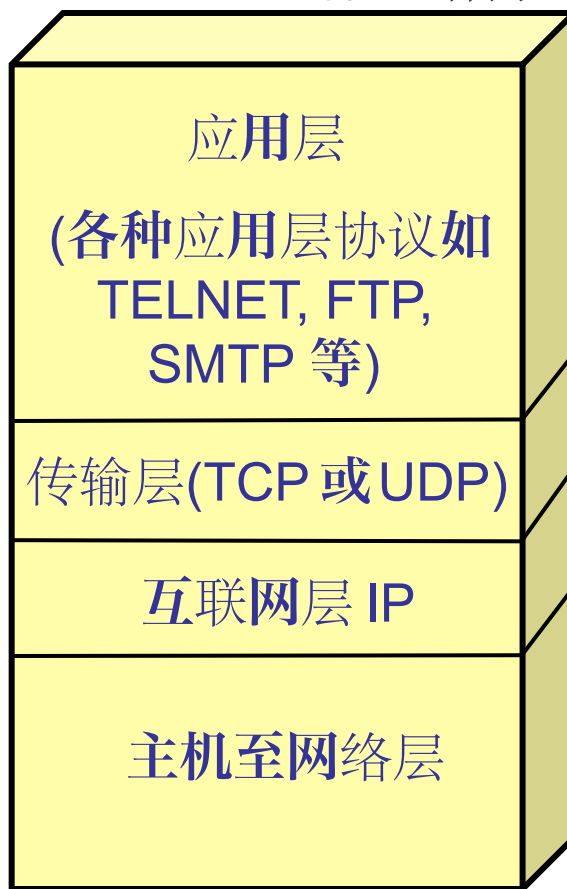


# OSI 与 TCP/IP 体系结构的比较

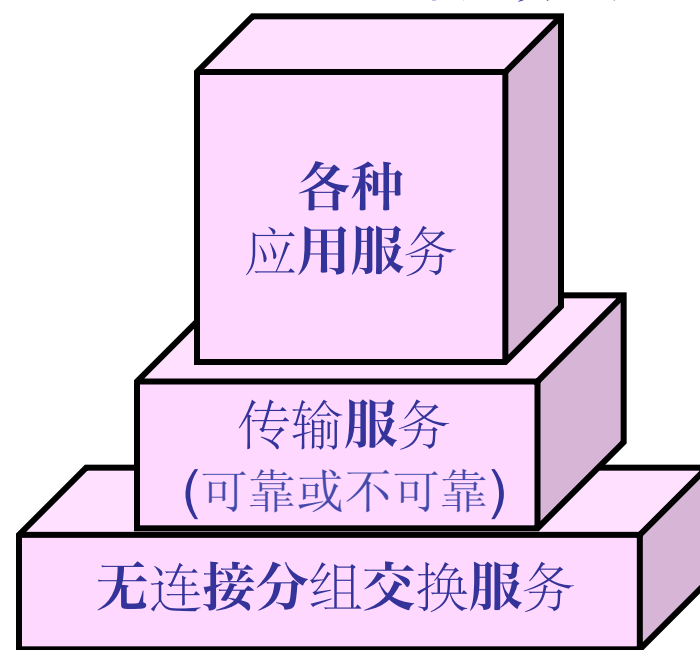
OSI 的体系结构



TCP/IP 的体系结构



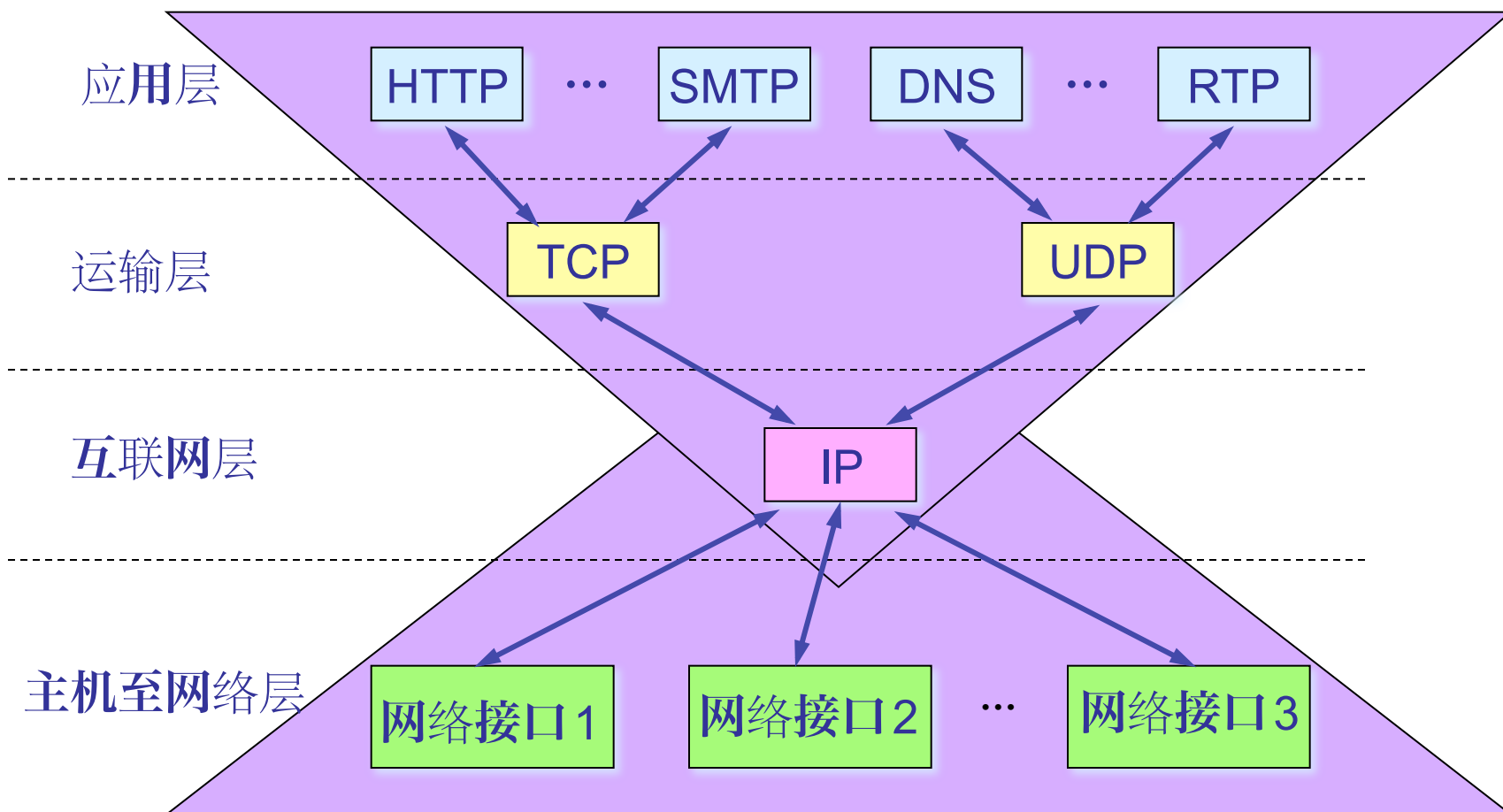
TCP/IP 的三个服务层次





# IP over Everything

## IP可应用到各式各样的网络上







# 计算机网络的发展过程

## 分组交换的产生

### 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，而不是为了打电话。
- ❖ 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。

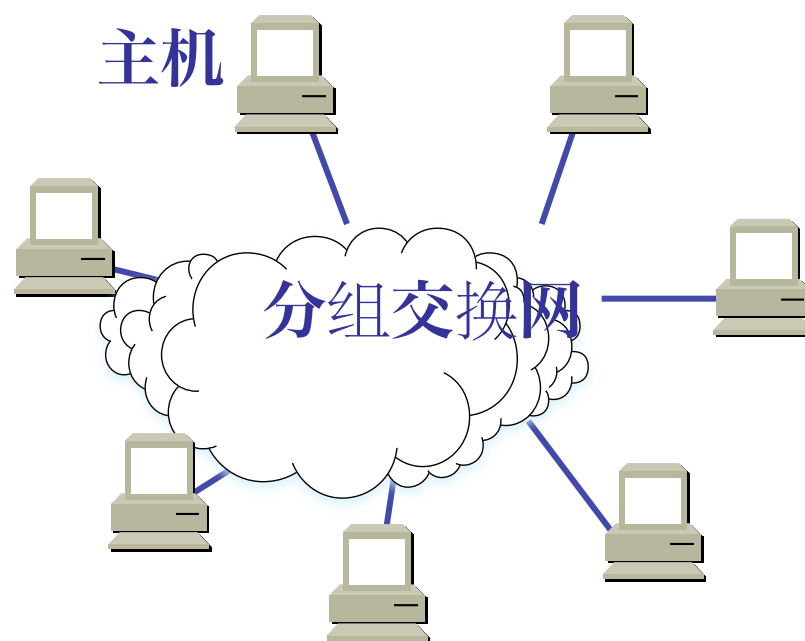
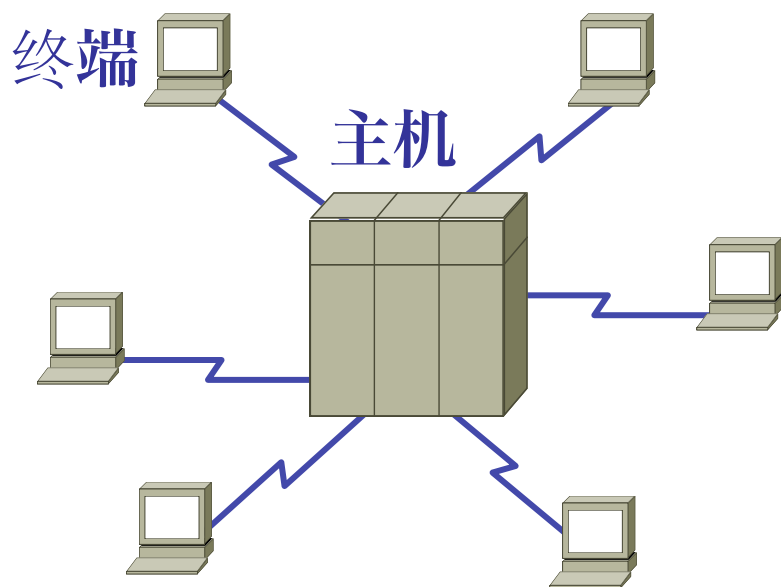


# 从主机为中心到以网络为中心

以主机为中心



以分组交换网为中心





# 因特网时代

- ❖ 因特网的基础结构大体上经历了三个阶段的演进。
- ❖ 但这三个阶段在时间划分上并非截然分开而是**有部分重叠**的，这是因为网络的演进是**逐渐的**而不是**突然的**。



# 因特网发展的第一阶段

- ❖ 第一个分组交换网 ARPANET 最初只是一个单个的分组交换网。
- ❖ ARPA 研究多种网络互连的技术。
- ❖ 1983 年 TCP/IP 协议成为**标准协议**。
- ❖ 同年，ARPANET 分解成两个网络：
  - **ARPANET——进行实验研究用的科研网**
  - **MILNET——军用计算机网络**
- ❖ 1983~1984 年，形成了**因特网** Internet。
- ❖ 1990 年 ARPANET 正式宣布关闭。



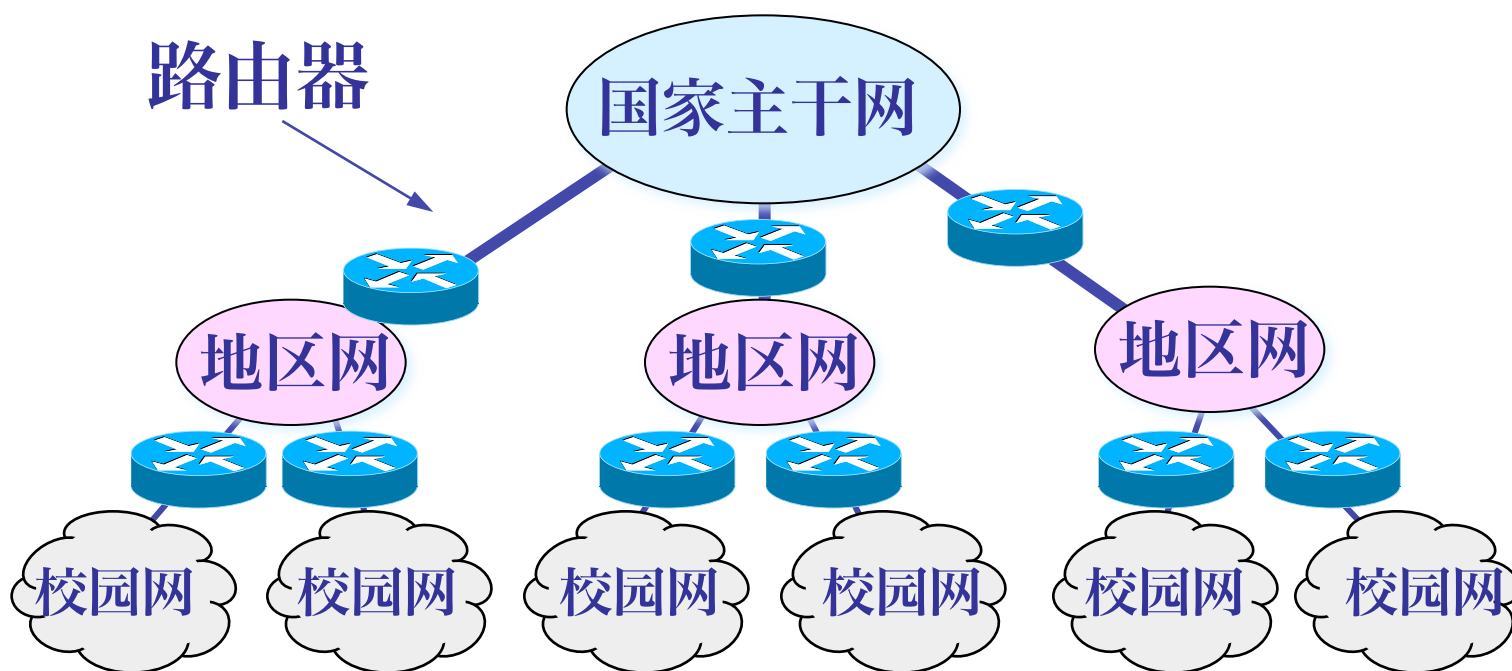
# 因特网发展的第二阶段

- ❖ 1986 年，NSF 建立了国家科学基金网。  
NSFNET。它是一个三级计算机网络：
  - 主干网
  - 地区网
  - 校园网
- ❖ 1991 年，美国政府决定将因特网的主干网转交给私人公司来经营，并开始对接入因特网的单位收费。
- ❖ 1993 年因特网主干网的速率提高到 45 Mb/s (T3 速率) 。



# 三级结构的因特网

- ❖ 各网络之间需要使用**路由器**来连接。



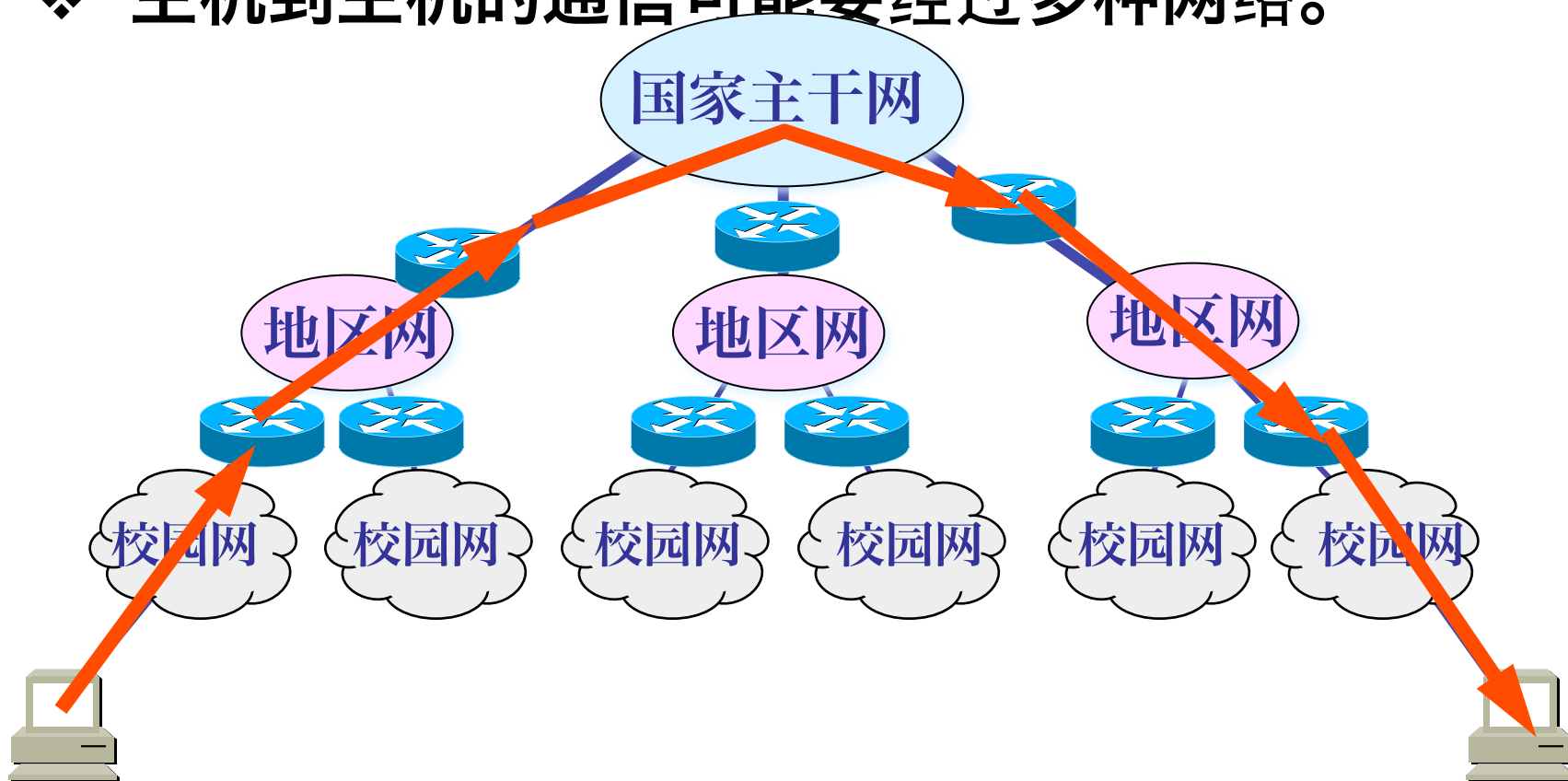
- ❖ 有时在结构图中可不画出路由器。





# 三级结构的因特网

- ❖ 主机到主机的通信可能要经过多种网络。





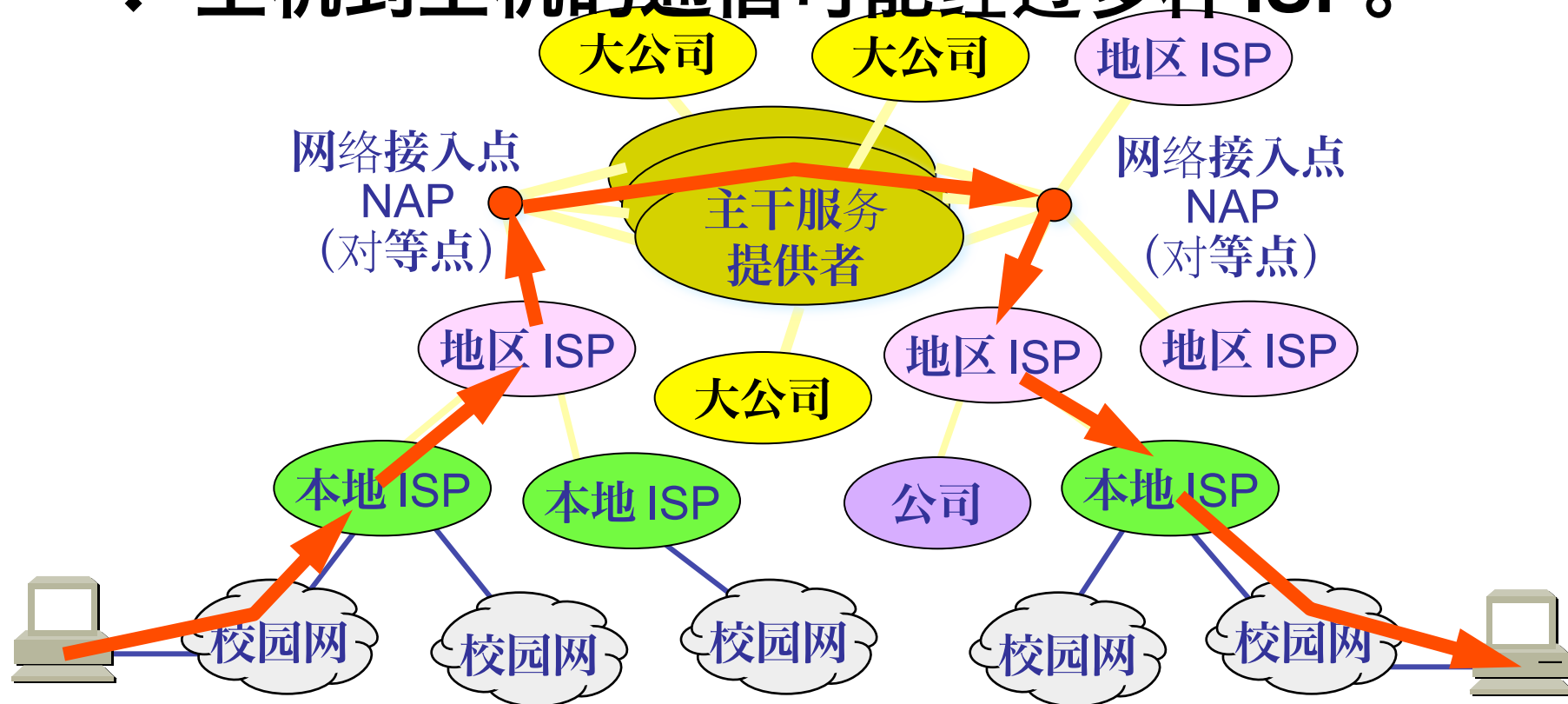
# 因特网发展的第三阶段

- ❖ 从1993年开始，由美国政府资助的 NSFNET 逐渐被若干个商用的 ISP 网络所代替。
- ❖ 1994 年开始创建了 4 个网络接入点 NAP (Network Access Point)，分别由 4 个电信公司经营。
- ❖ NAP 就是用来交换因特网上流量的结点。在 NAP 中安装有性能很好的交换设施。到本世纪初，美国的 NAP 的数量已达到十几个。
- ❖ 从 1994 年到现在，因特网逐渐演变成多级结构网络。



# 多级结构的因特网

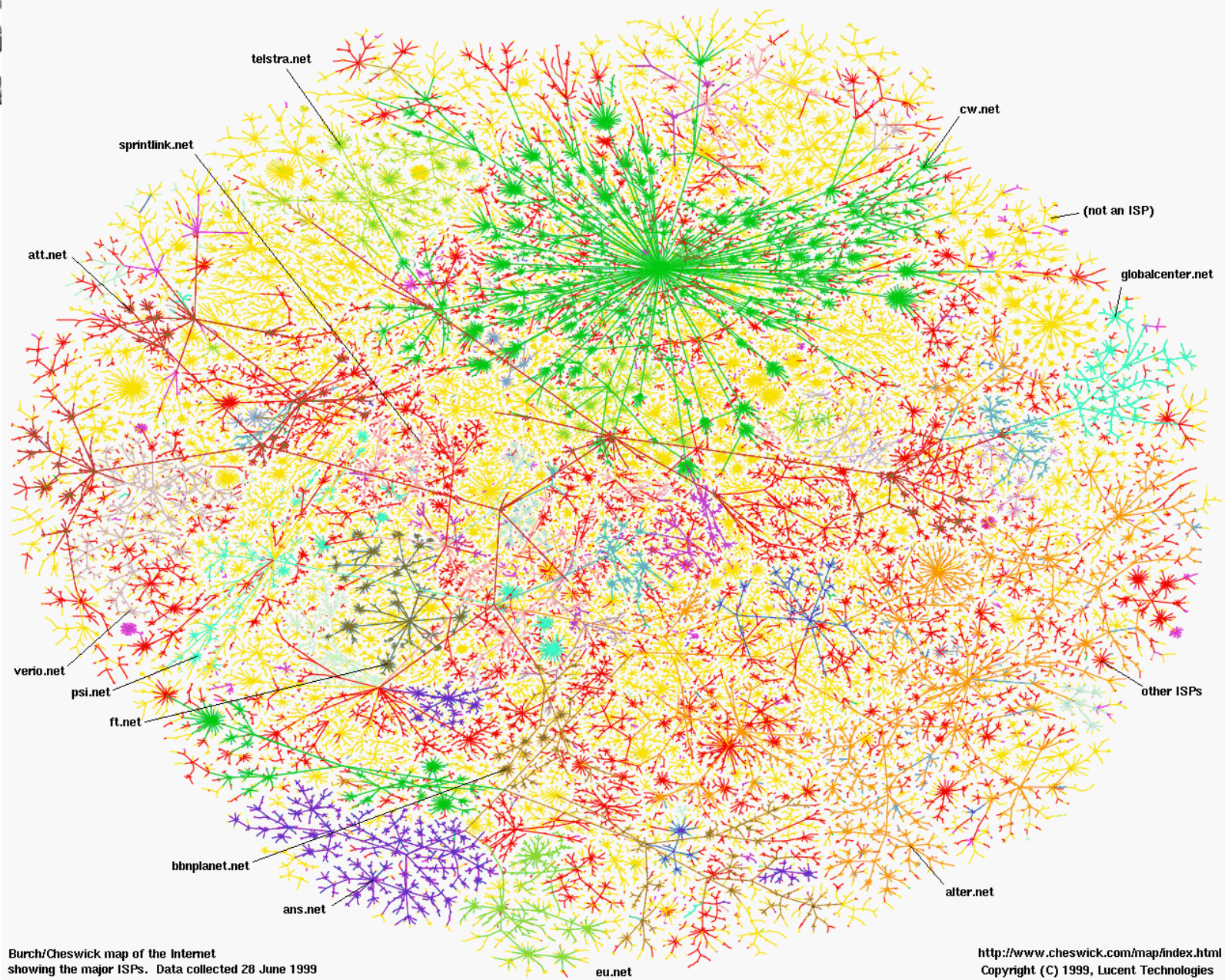
❖ 主机到主机的通信可能经过多种ISP。





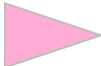
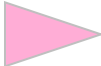
# 今日的多级结构的因特网

- ❖ 大致上可将因特网分为以下五个接入级
  - 网络接入点 **NAP**
  - 国家主干网（主干**ISP**）
  - 地区**ISP**
  - 本地**ISP**
  - 校园网、企业网或 **PC** 机上网用户





# 第1章 概述

1. 计算机网络组成和分类 
2. 计算机网络参考模型 
3. 数字通信的基本概念 





# 数据通信基本概念

|       |         |
|-------|---------|
| 信号和信道 | 位同步和帧同步 |
| 数据的编码 | 串行和并行   |
| 采样和量化 | 复用技术    |
| 数据的传输 | 交换技术    |



# 波特率与比特率

- ❖ **波特率：信号变化次数（每秒采样的次数）**
- ❖ **比特率：数据传输速率（每秒传输的比特数）**

如果信号分为2级：0 1，则比特率 = 波特率

即一次采样得到一个比特

如果信号分为8级：0 1 2 3 4 5 6 7，则1波特相当于3 比特

即一次信号变化（一次采样）可表示3bit

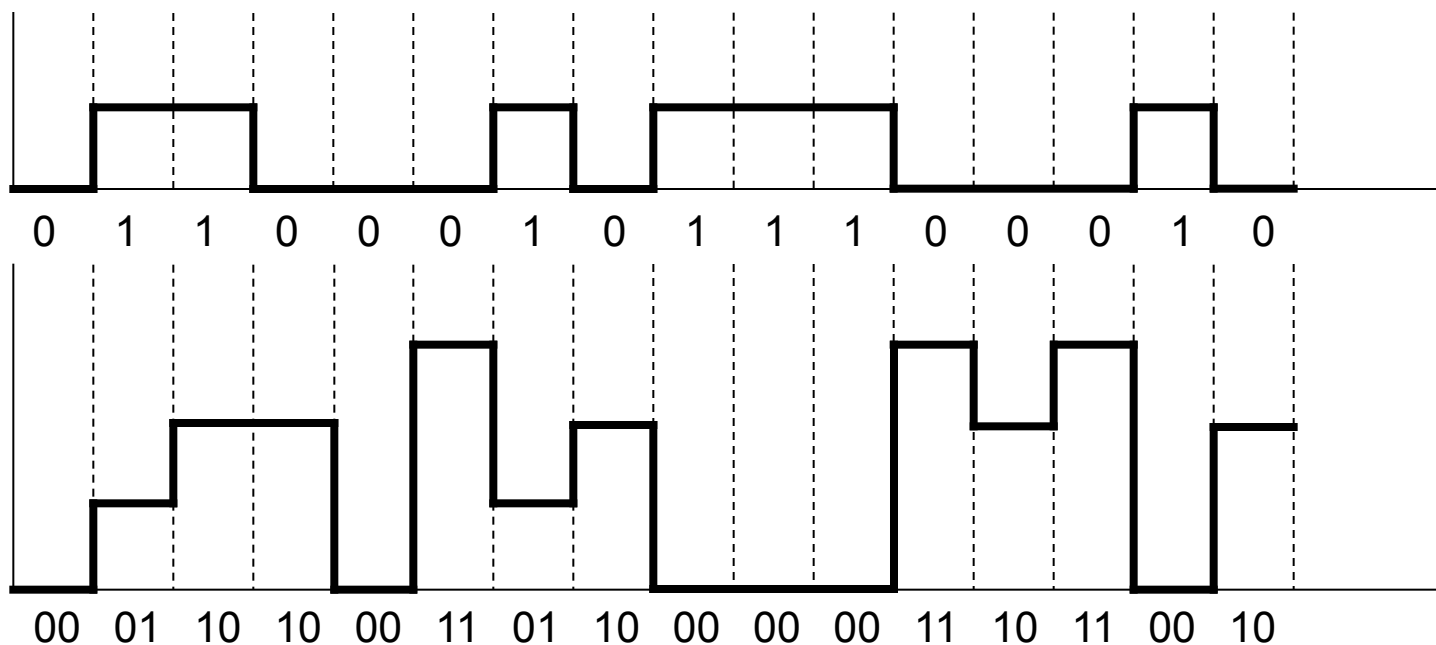
如信号分为V级，则比特率 =  $\log_2 V$  波特率





# 波特率和比特率

❖ 两者数值上的差别在于每次采样的量化值





# 数据与信号

- ❖ 数据涉及的是事物的表现形式
- ❖ 数据有模拟数据和数字数据两种形式
  - 模拟数据是指在某个时间段产生的连续的值, 例如声音和视频、温度和压力等都是时间的连续函数
  - 数字数据是指产生的离散的值, 例如文本信息和整数
- ❖ 信号是数据的表示形式, 或称数据的电磁或电子编码, 它使数据能以适当的形式在介质上传输
  - 信号有模拟信号和数字信号两种基本形式
  - 模拟信号可用来携带数字数据, 数字信号也可用于表示模拟数据



# 数据传输

## 数据传输以信号为载体

- ❖ **模拟传输**：是指模拟数据的传输，不关心所传输信号的内容，而只关心尽量减少信号的衰减和噪声，长距离传输时，采用信号放大器放大被衰减的信号，但同时也放大了信号中的噪声
- ❖ **数字传输**：是指数字数据的传输，关心信号的内容，可以数字信号传输，也可以模拟信号传输，长距离传输时，采用转发器，可消除噪声的累积
- ❖ **长距离传输时，通常采用的是数字传输**



# 信道参数和名称解释

- ❖ 数据的传输速率、带宽
- ❖ 信号的传播速率
- ❖ 载波频率
- ❖ 采样频率
- ❖ 量化
- ❖ 噪声、信噪比



# Nyquist定理

在无噪声信道中，当带宽为 $H$  Hz，信号电平为 $V$ 级，则：

$$\text{数据传输速率} = 2H \log_2 V \text{ b/s}$$

（ $V$ ：信号电平的级数，在二进制中，仅为0、1两级）

即：以每秒高于 $2H$ 次的速率对线路采样是无意义的，因为高频分量已被滤波器滤掉无法再恢复



## 要强调以下两点

- ❖ 实际的信道所能传输的最高码元速率，要明显地低于奈氏准则给出上限数值。
- ❖ 波特(Baud)和比特(bit)是两个不同的概念。
  - **波特**是码元传输的速率单位（每秒传多少个码元）。码元传输速率也称为**调制速率、波形速率或符号速率**。
  - **比特**是信息量的单位。



# Shannon定理

在噪声信道中，当带宽为 $H$  Hz，信噪比为 $S/N$ ，  
则：

$$\text{最大数据传输速率(b/s)} = H \log_2(1+S/N)$$

很多情况下信噪比用分贝（dB）表示

$$\text{信噪比 (dB)} = 10 \log_{10} S/N$$

如：信噪比为30dB，则 $S/N=1000$



# 香农公式表明

- ❖ 信道的带宽或信道中的信噪比越大，则信息的极限传输速率就越高。
- ❖ 只要信息传输速率低于信道的极限信息传输速率，就一定可以找到某种办法来实现无差错的传输。
- ❖ 若信道带宽  $W$  或信噪比  $S/N$  没有上限（当然实际信道不可能有这样的），则信道的极限信息传输速率  $C$  也就没有上限。
- ❖ 实际信道上能够达到的信息传输速率要比香农的极限传输速率低不少。





## 举例：噪声信道中的传输速率

在噪声信道（话音信道）中，当带宽为3500Hz，信噪比为30dB（较为典型的电话信道），则：

$$\begin{aligned}\text{最大数据传输速率(b/s)} &= H \log_2(1+S/N) \\ &= 3500 \log_2(1+1000) \\ &\approx 35000 \text{ (b/s)}\end{aligned}$$

最大数据传输速率为35k bps，这是在噪声信道中的传输速率极限，实际上是不可能达到的



# 信道工作方式

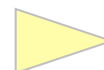
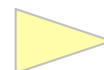
- ❖ **单工通信**：单向传输，如广播、电视
- ❖ **半双工**：双方都可以发送或接收，但不能同时，即当一方发送时，另一方接收
- ❖ **全双工**：双方同时可以发送和接收信息

**全双工则需要两条信道**



# 数据传输

- ❖ 数字数据的模拟信道传输
- ❖ 数字数据的数字信道传输
- ❖ 模拟数据的数字信道传输





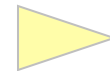
# 数字数据的模拟信号传输

将数字数据调制成模拟信号进行传输，  
通常有三种基本的调制方式：

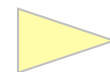
1) 调幅ASK(Amplitude Shift Keying)



2) 调频FSK(Frequency Shift Keying)

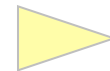


3) 调相PSK(Phase Shift Keying)



目前广泛应用的调制技术是正交调相：

QPSK(Quadrature Phase Shift Keying)

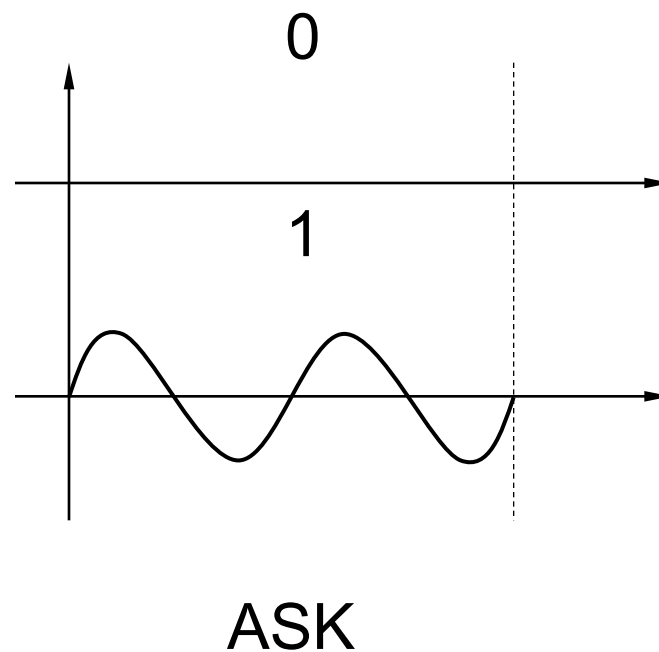




# 调幅ASK (Amplitude Shift Keying)

用载波的两个不同  
的振幅来分别表示  
两个二进制值

如用无信号表示0  
有信号表示1

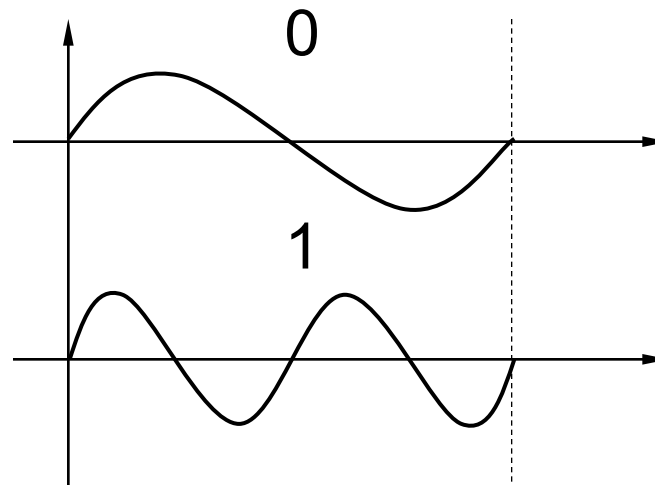




# 调频FSK (Frequency Shift Keying)

用载波频率为 $1.5f$ 附近的  
的两个不同的频率来分  
别表示两个二进制值

如用信号频率为 $f$ 表示0  
信号频率为 $2f$ 表示1



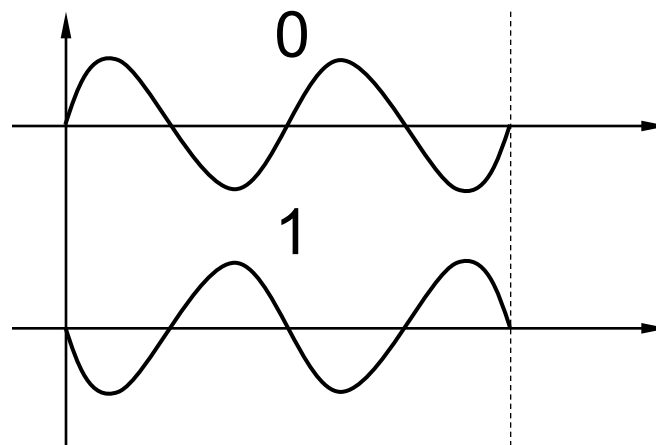
FSK



# 调相PSK (Phase Shift Keying)

用载波的相位移动来分  
别表示两个二进制值

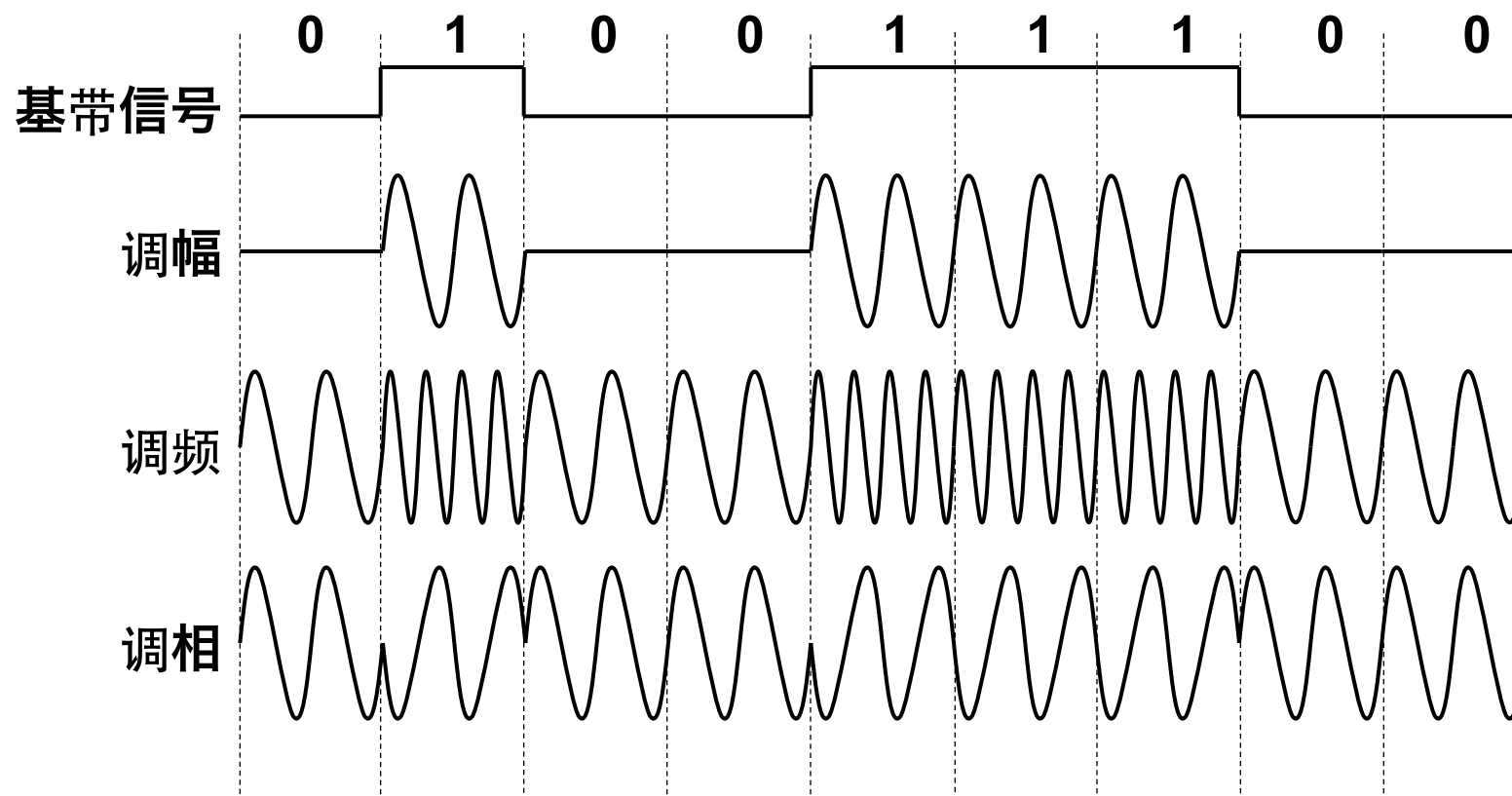
如用信号相位角为0表示0  
相位角为 $\pi$ 表示1



PSK



# 数字数据的调制举例

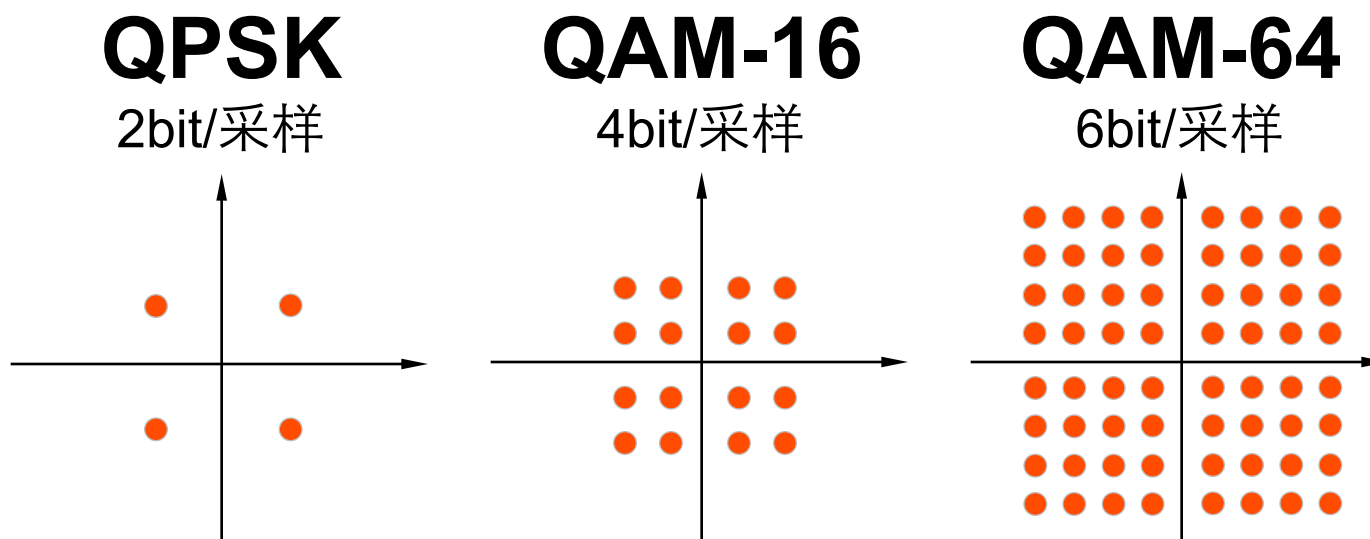






# 正交调相QPSK

## (Quadrature Phase Shift Keying)

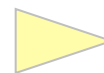
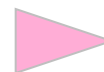


Tnbm P128 Fig. 2-25 (a) (b) (c)



# 数据传输

- ❖ 数字数据的模拟信道传输
- ❖ 数字数据的数字信道传输
- ❖ 模拟数据的数字信道传输



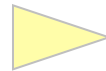
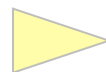
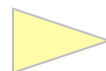


# 数字数据的数字信号传输

最简单的方法是用两个不同的电压信号值来表示两个二进制的数字数据值0和1

常用的数字信号编码有：

- ❖ 不归零编码
- ❖ 曼切斯特编码
- ❖ 差分曼切斯特编码
- ❖ 4B/5B编码

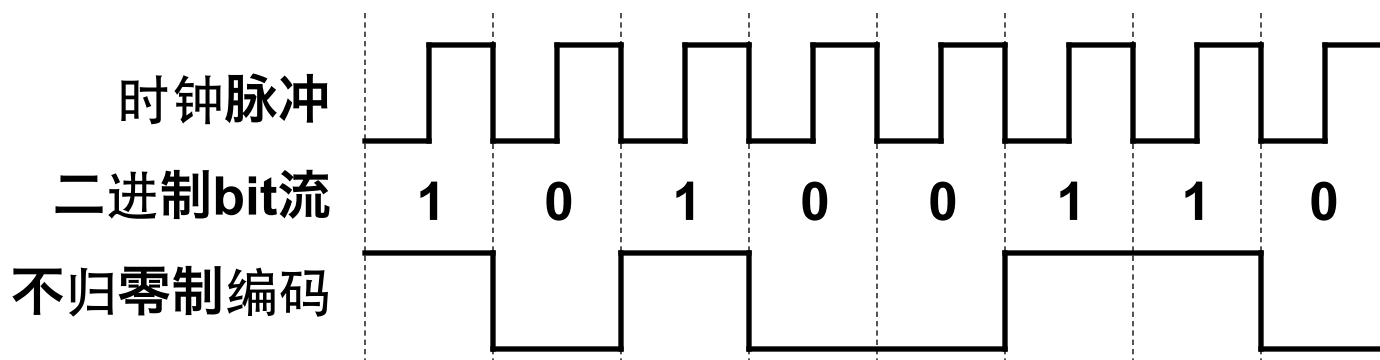




# 不归零编码 NRZ (nonreturn-to-zero)

正电平表示1，零电平表示0，并且在表示完一个码元后，电平无需回到零

缺点是存在发送方和接收方的同步问题



其实，用不归零制编码时，一个时钟周期可表示两个bit

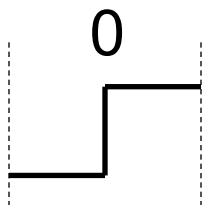
所以，不归零制编码是效率最高的编码

但它不能携带时钟信号，且无法表示没有数据传输

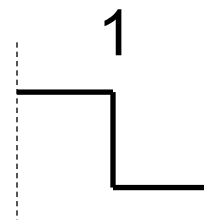


# 曼切斯特编码 (Manchester encoding)

bit中间有信号  
低-高跳变为0



bit中间有信号  
高-低跳变为1



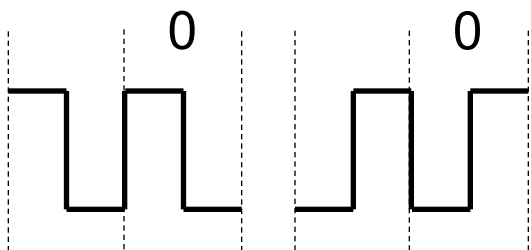
采用曼切斯特编码，一个时钟周期只可表示一个bit  
，并且必须通过两次采样才能得到一个bit  
但它能携带时钟信号，且可表示没有数据传输



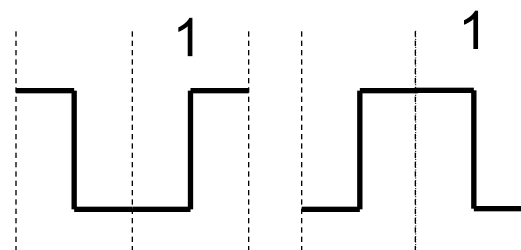
# 差分曼彻斯特编码

(differential Manchester encoding)

bit中间有信号跳变,  
bit与bit之间也有信号跳变,  
表示下一个bit为0



bit中间有信号跳变,  
bit与bit之间无信号跳变,  
表示下一个bit为1



特性与曼切斯特编码相同, 但抗干扰性能强于曼切斯特编码



# 4B/5B编码

- ❖ 不归零制编码的一种变种：逢一变化的不归零制NRZ1 (nonreturn-to-zero change on one)
- ❖ 数据流中每4个bit成一个组合，并对应为5个bit的编码
- ❖ 5B编码中至少有两个1，即保证在传输中信号码元至少发生两次跳变，这是在接收端提取时钟信号所必须的



# 16进制数的4B/5B编码对照表

| 16进制数 | 4bit码 | 5bit码 | 16进制数 | 4bit码 | 5bit码 |
|-------|-------|-------|-------|-------|-------|
| 0     | 0000  | 11110 | 8     | 1000  | 10010 |
| 1     | 0001  | 01001 | 9     | 1001  | 10011 |
| 2     | 0010  | 10100 | A     | 1010  | 10110 |
| 3     | 0011  | 10101 | B     | 1011  | 10111 |
| 4     | 0100  | 01010 | C     | 1100  | 11010 |
| 5     | 0101  | 01011 | D     | 1101  | 11011 |
| 6     | 0110  | 01110 | E     | 1110  | 11100 |
| 7     | 0111  | 01111 | F     | 1111  | 11101 |

发送端发送的每4个bit被用5个bit表示，接收端采样5次便可得到4个bit，由于每个5B编码中至少有两个1，所以接收端能提取时钟信号





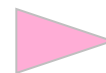
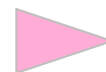
# 四种编码方式的比较

- ❖ 不归零制编码的编码密度最高，接收端一次采样可得到一个bit，即波特率等于比特率，但不能携带时钟
- ❖ 曼切斯特编码的编码密度最低，接收端二次采样才可得到一个bit，即波特率是比特率的两倍，但每个bit中都有信号跳变，即携带了时钟
- ❖ 差分曼切斯特编码与曼切斯特编码基本相同
- ❖ 4B/5B编码的编码密度略低于不归零制编码，但高于曼切斯特编码，即波特率是比特率的1.25倍，然而在接收端能提取时钟



# 数据传输

- ❖ 数字数据的模拟信道传输
- ❖ 数字数据的数字信道传输
- ❖ 模拟数据的数字信道传输





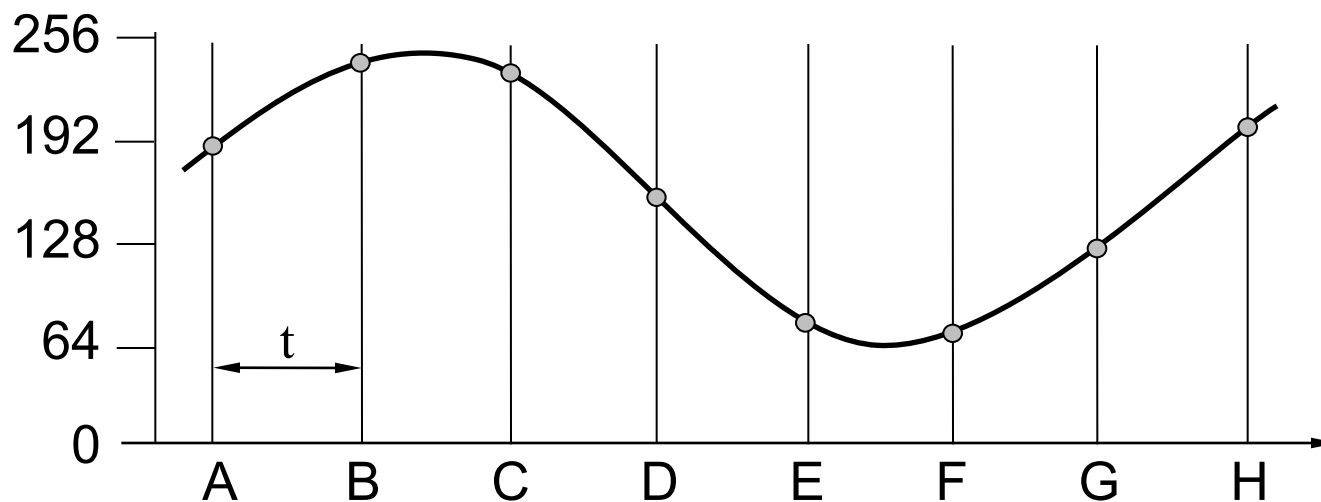
# 模拟数据在数字信道上传输

采用脉冲编码调制PCM技术  
(Pulse Code Modulation)  
PCM以采样定理为基础

**采样定理：**如果在规定的时间内，以有效信号 $f(t)$ 最高频率的二倍或二倍以上的速率对该信号进行采样，则这些采样值中包含了全部原始信号信息



# 采样、量化和编码



| A        | B        | C        | D        | E        | F        | G        | H        |
|----------|----------|----------|----------|----------|----------|----------|----------|
| 188      | 244      | 240      | 144      | 80       | 72       | 122      | 200      |
| 10111100 | 11110100 | 11110000 | 10010000 | 01010000 | 01001000 | 01111100 | 11001000 |



# 话音信道

**话音信道允许的最高频率通常为3500Hz**  
**如果以8000Hz的采样频率对话音信号进行采样的话，则在采样值中包含了话音信号的完整特征，由此而还原出的话音是完全可理解的和可识别的**



# 话音信道的数据传输速率

对于每一个采样值还需要用一个（一位或多位的）二进制代码来表示，二进制代码的位数代表了采样值的量化精度，在主干上，对每一路话音信号通常采用8位二进制代码来表示一个采样值，那么，对话音信号进行PCM编码后所得到的数据传输速率为：

$$8 \text{ bit} \times 8000 \text{ 次采样/秒} = 64\text{k b/s}$$



# 基带传输

❖ **信号源产生的原始电信号称为基带信号**

**即：将数字数据0、1直接用两种不同的电压表示，然后送到线路上去传输**

**用于数字传输：在局域网中通常采用数字传输**

**同轴电缆（50  $\Omega$ ）：185 M（细缆）、500 M（粗缆）**

**双绞线（8芯）：100 M 常用的是UTP cat 3/cat 5**



# 宽带传输

- ❖ 将基带信号进行调制后形成模拟信号，或者说，将基带信号所携带的数据调制在模拟信号上，然后采用频分复用技术实现宽带传输

有线电视网：带宽可达750 MHz，由于以模拟信号传输，所以传输距离可达100 km

宽带系统可分为多个信道，所以模拟和数字数据可混合使用，但通常需解决数据双向传输的问题

在混合光纤电缆 HFC (Hybrid Fiber Coax) 中，频段 54 ~ 550 MHz是电视信号，550-750 MHz是数字数据





# 异步通信与同步通信

在数字数据的数字信号传输中，一个最基本的要求是发送端和接收端之间以某种方式保持位同步，只有保证了位同步才可能保证帧同步，所以接收端必须对它所接收的数据流中每一位进行正确的采样，才能确保数据接收的正确性，为此，通信双方必须遵循同一个通信规程，使用相同的位同步方式进行数据传输，根据通信规程所定义的位同步方式，可分为异步通信和同步通信两大类



# 异步通信

- ❖ 异步通信是指发送方和接收方的采样时钟不是同一个，故名，是以字符为单位的数据传输，常用的是ASCII字符集
- ❖ 数据块以字符为单位，每个字符都要附加1位起始位和1位停止位作标志，以标记字符的开始和结束
- ❖ 此外，还要附加1位奇偶校验位

异步通信必须指定的四个参数：

- 波特率
- 奇偶校验
- 字符长度
- 停止位长度



# 同步通信

- ❖ 同步通信是指发送方和接收方的采样时钟是同一个，故名
- ❖ 通常发送方在发送数据的编码中包含时钟，而接收方则从数据流中提取时钟用以采样，所以说双方所用的时钟是同一个
- ❖ 根据同步通信规程，同步通信又分为面向字符的同步通信和面向bit流的同步通信



# 面向字符的同步通信

在面向字符的同步通信中，字符集可用ASCII或EBCDIC，数据块由字符组成，数据块前加一个或两个同步字符SYN用于数据块的同步

每个字符需起始位和停止位

典型的面向字符的同步通信规程是BISYNC  
与异步通信相类似，必须指定相关参数：

- 波特率
- 字符长度
- 奇偶校验



# 面向bit流的同步通信

在面向bit流的同步通信中，每个数据块的头部和尾部用一个或多个特殊的比特序列(如01111110)来标记数据块的开始和结束，数据块将作为bit流来处理，而不是作为字符流来处理

典型的面向位流的同步通信规程是高级数据链路控制（HDLC）规程和同步数据链路控制（SDLC）规程，收发双方必须约定的参数有：

- 传输速率
- CRC生成多项式 等



# 面向bit流的透明传输

在面向bit流的同步通信中，为了避免在数据块的数据中出现标记数据块开始和结束的特殊位模式(如0111110)，通常采用位插入法，即发送端总是检测所发送的数据流，每当出现连续的五个1后便自动插入一个0，接收端在接收数据流时，如果检测到连续五个1的序列，就检查其后的一个bit，若该bit是0，则删除，若该bit为1，则表示数据块的结束，转入结束处理



# 串行通信和并行通信

❖ 串行通信：

数据按位为单位，以时间为序

❖ 并行通信：

数据按字符为单位，以时间为序



# 典型的串行接口RS-232C

**RS-232-C是用于计算机或终端与Modem间的物理层协议，所谓物理层协议是定义接口的机械、电气、功能和过程特性**

➤ **计算机或终端设备：**

**DTE (data terminal equipment)**

➤ **调制解调器：**

**DCE (data circuit-terminating equipment)**





# RS-232C标准

**机械特性：25针D型插座及相关的长、宽、高**  
**25 PIN 9 PIN**

**DTE端为Male**

**DCE端为Female**

**电气特性：逻辑0：+12 V**

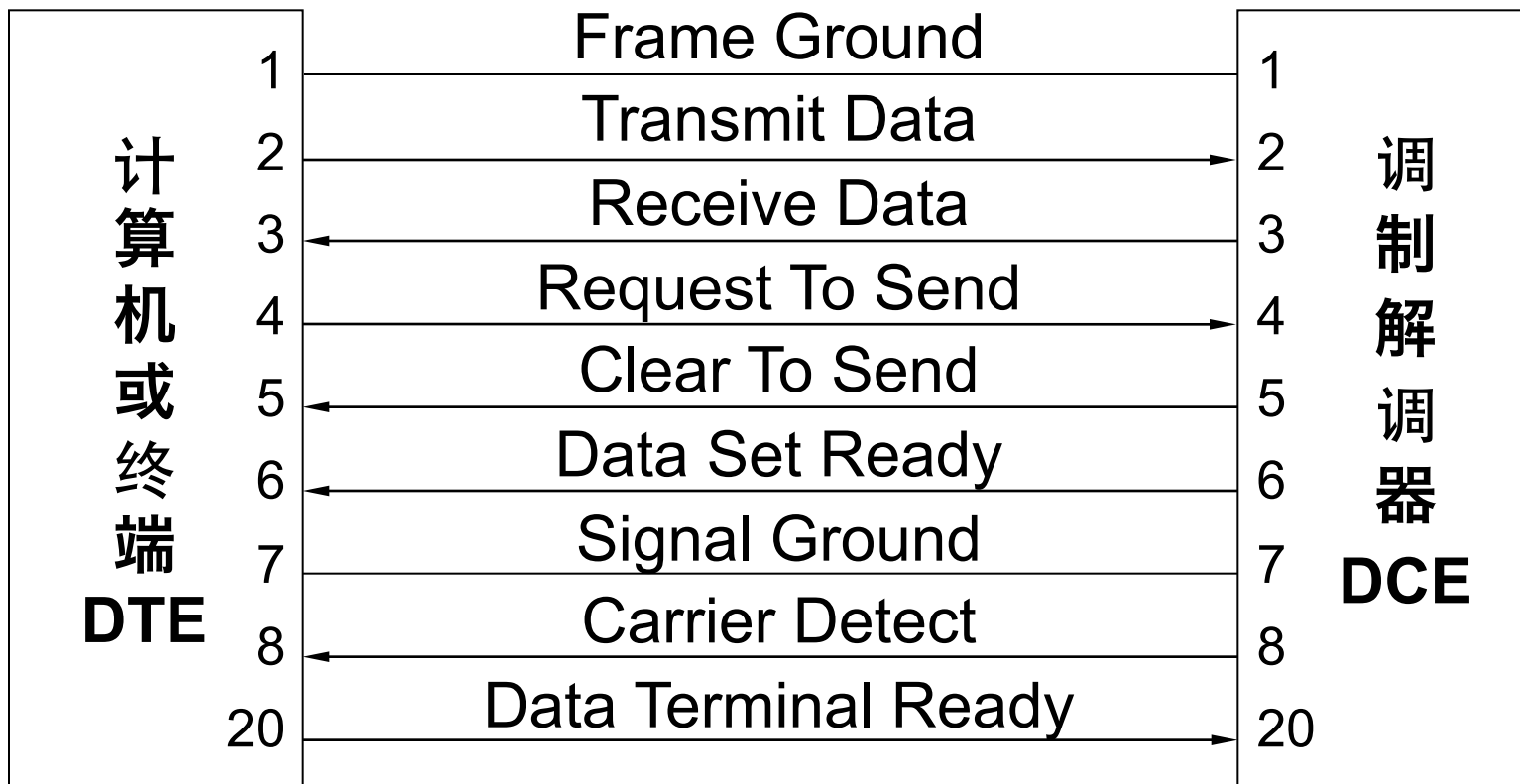
**逻辑1：-12 V**

**最长传输距离15 m**

**最大传输速率 <20k b/s**

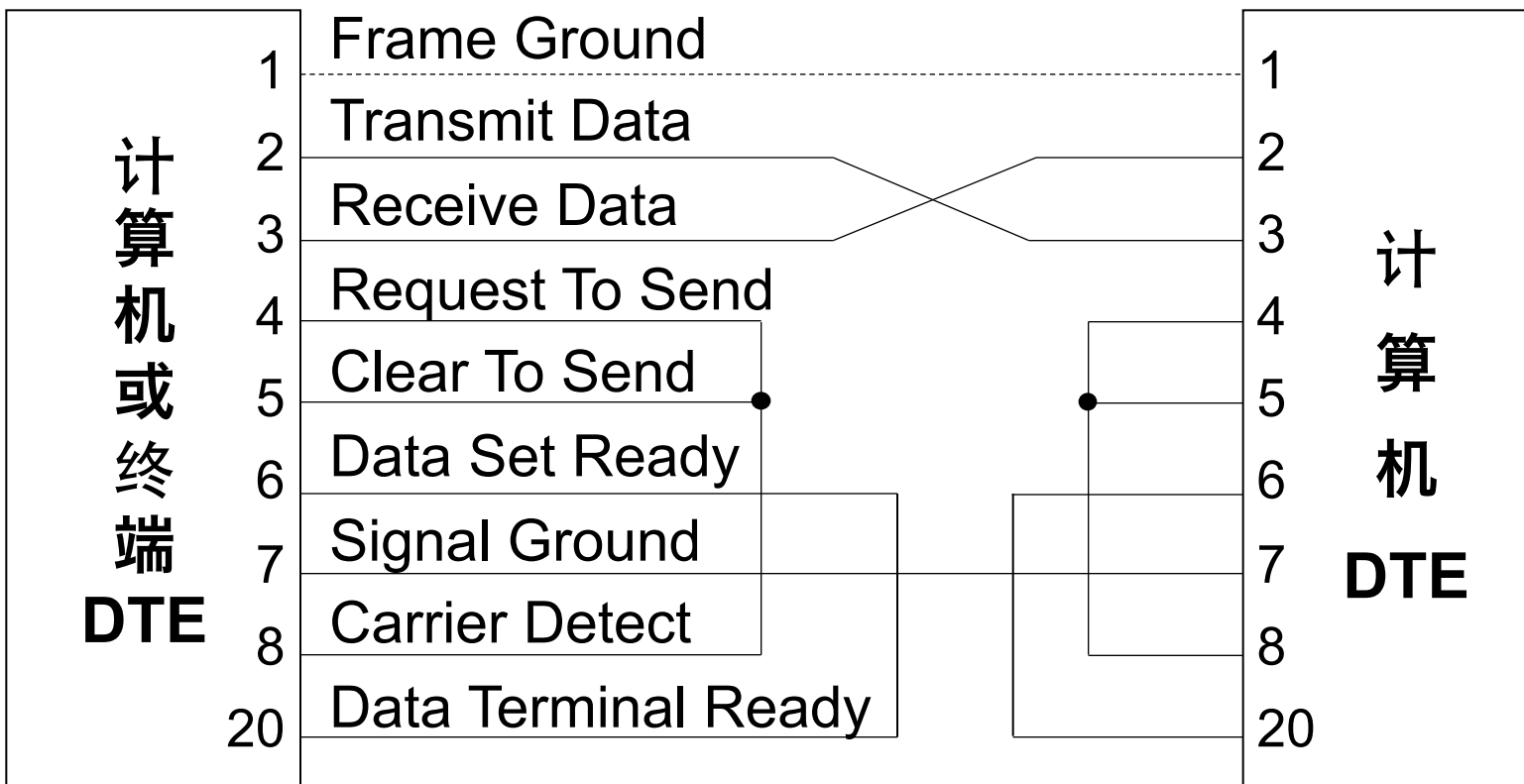


# 计算机或终端通过 RS-232-C 接口与调制解调器连接





# 计算机或终端通过 RS-232-C 接口与计算机连接





# 常用的串行接口

❖ 常用的串行接口有25pin和9pin两种

| 25pin | 9pin | 功能                  | 简称  |
|-------|------|---------------------|-----|
| 1     |      | Frame Ground        | FG  |
| 2     | 3    | Transmit Data       | TxD |
| 3     | 2    | Receive Data        | RxD |
| 4     | 7    | Request To Send     | RTS |
| 5     | 8    | Clear To Send       | CTS |
| 6     | 6    | Data Set Ready      | DSR |
| 7     | 5    | Signal Ground       | GND |
| 8     | 1    | Carrier Detect      | CD  |
| 20    | 4    | Data Terminal Ready | DTR |



# 并行接口

## 打印机是常用的并行接口

| Pin | 信号      | Pin   | 信号          |
|-----|---------|-------|-------------|
| 1   | -STROBE | 10    | -ACK        |
| 2   | DATA0   | 11    | BUSY        |
| 3   | DATA1   | 12    | Paper Empty |
| 4   | DATA2   | 13    | Select      |
| 5   | DATA3   | 14    | -Auto Feed  |
| 6   | DATA4   | 15    | -ERROR      |
| 7   | DATA5   | 16    | -INIT       |
| 8   | DATA6   | 17    | -SELECT IN  |
| 9   | DATA7   | 18-25 | GROUND      |



# 多路复用

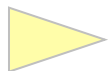
无论是广域网还是局域网，都存在这样一个事实，即传输介质的带宽大于传输单一信号所需的带宽，为了有效地利用传输系统，通常采用多路复用（Multiplexing）技术以同时携带多路信号来高效率地使用传输介质，多路复用主要有两种：

❖ 频分多路复用FDM



(Frequency Division Multiplexing)

❖ 时分多路复用TDM



(Time division Multiplexing)

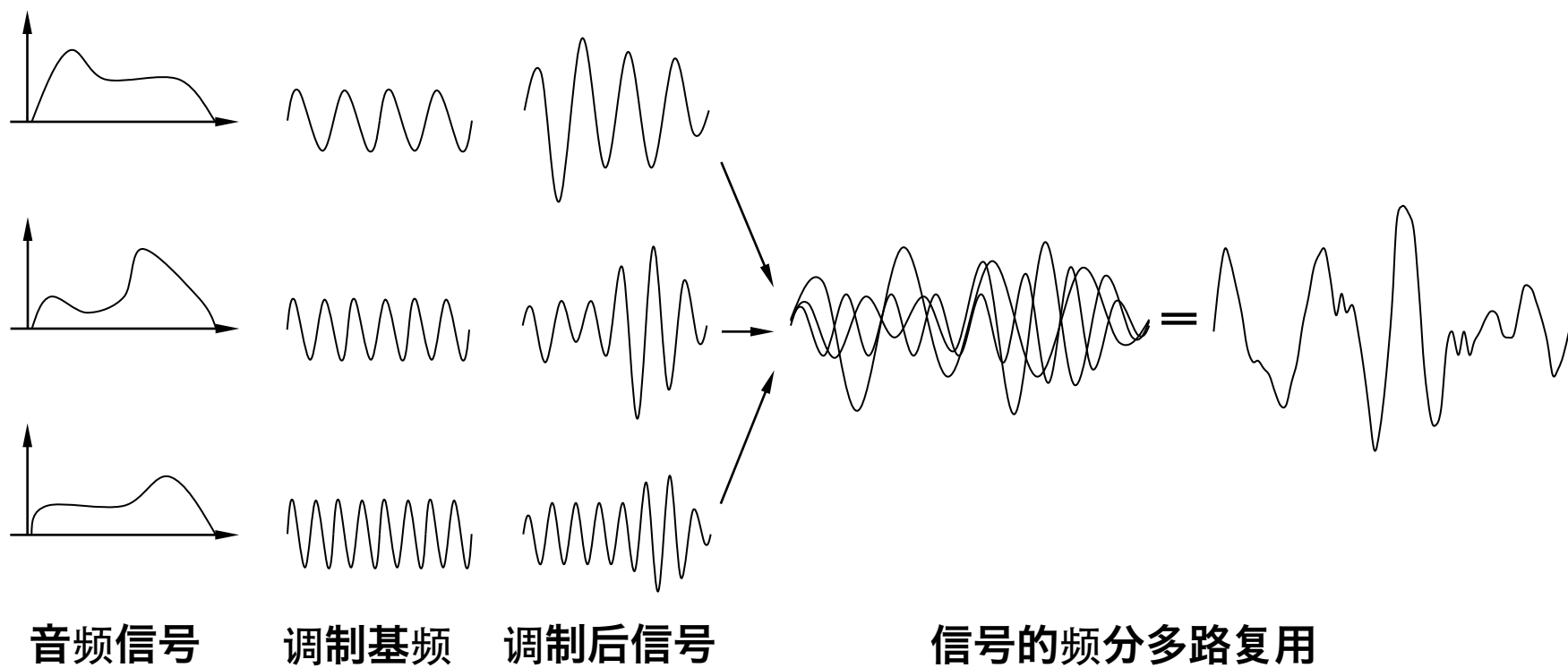


# 频分多路复用FDM

**FDM是基于这样的前提：传输介质的可用带宽必须超过各路给定信号所需带宽的总和，如果将这几路信号中的每路信号都以不同的载波频率进行调制，而且各路载波频率之间留有一定的间隔以使各路信号带宽不相互重叠，那么这些信号就可同时在介质上传输**



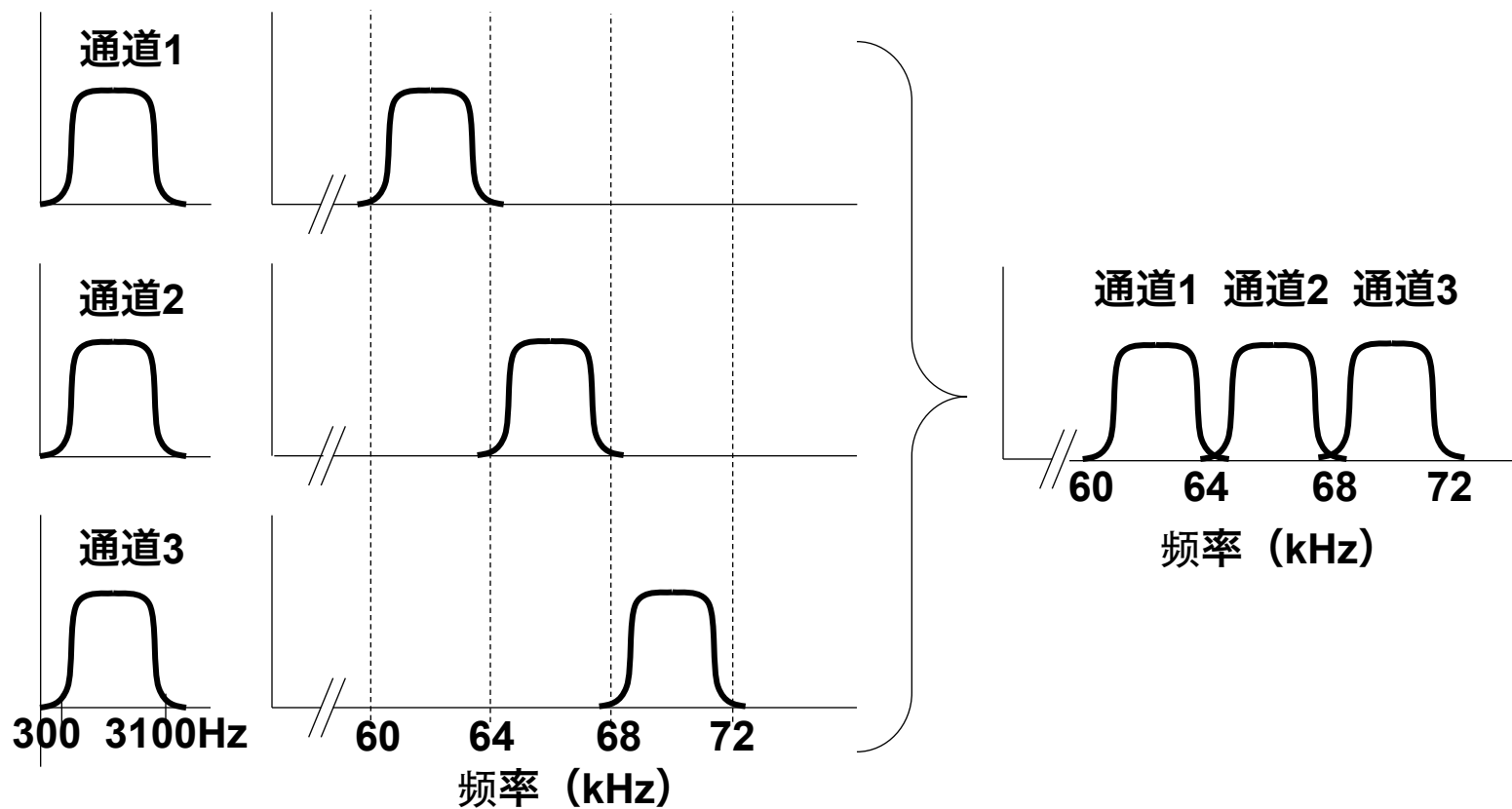
# FDM图示







# FDM图示 (续)



Tnbm P138 Fig. 2-31 (a)、(b)、(c)

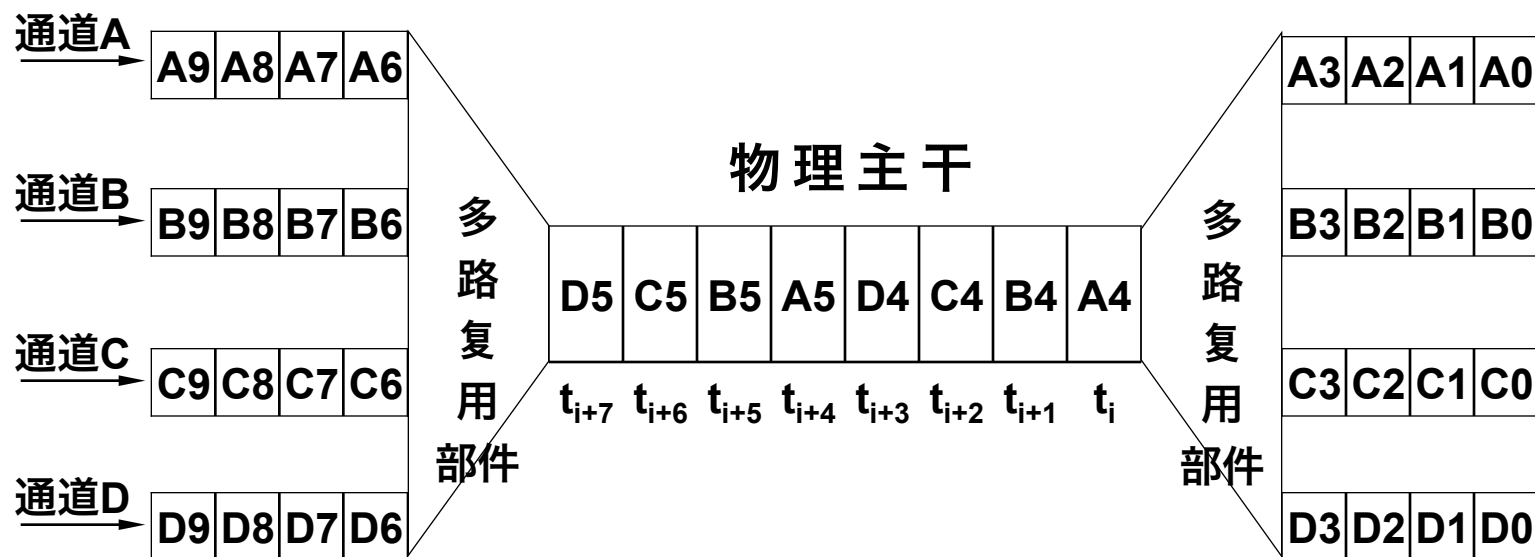


# 时分多路复用TDM

**每个信号按时间先后轮流交替地使用单一信道，那么，多个数字信号在宏观上可认为是同时进行传输，对单一信道的交替使用可以按位、字节或块等单位来进行**



# TDM图示





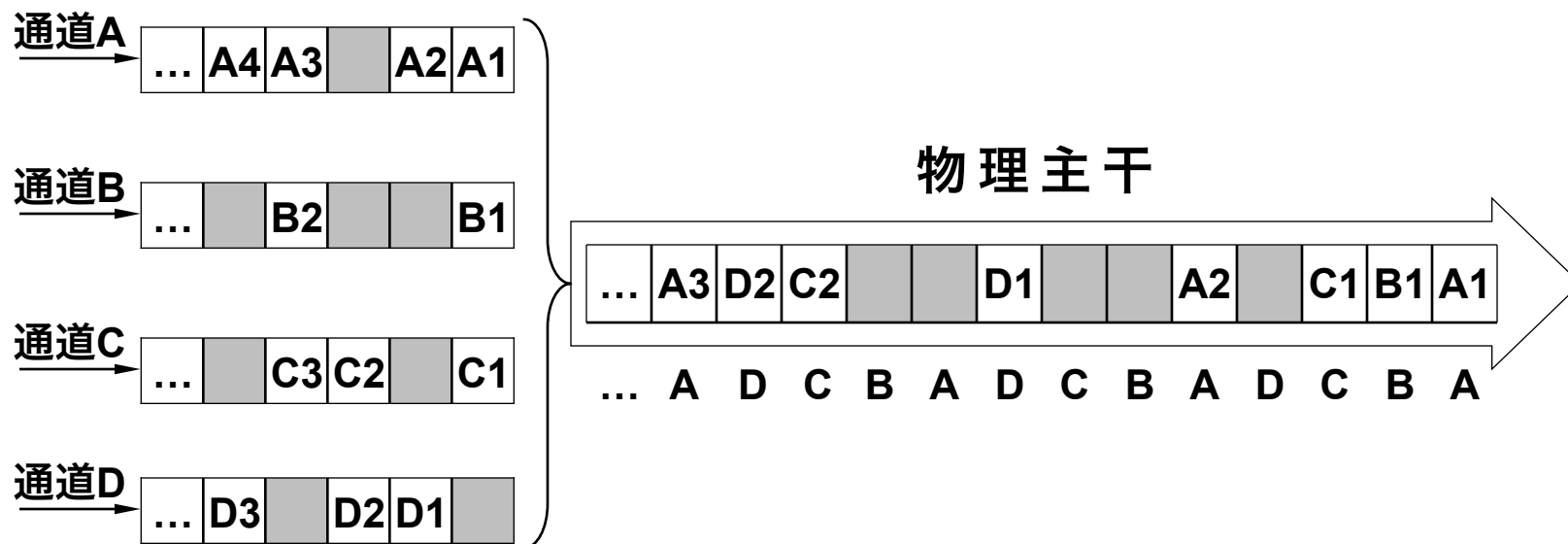
# TDM的同步和异步

## 同步TDM

- ❖ 时间片与输入装置一一对应，即同步
- ❖ 如某个时间片对应的输入装置无数据发送，则该时间片空闲（浪费）
- ❖ 传输介质的传输速率不能低于各个输入信号的数据速率之和



# 同步TDM图示





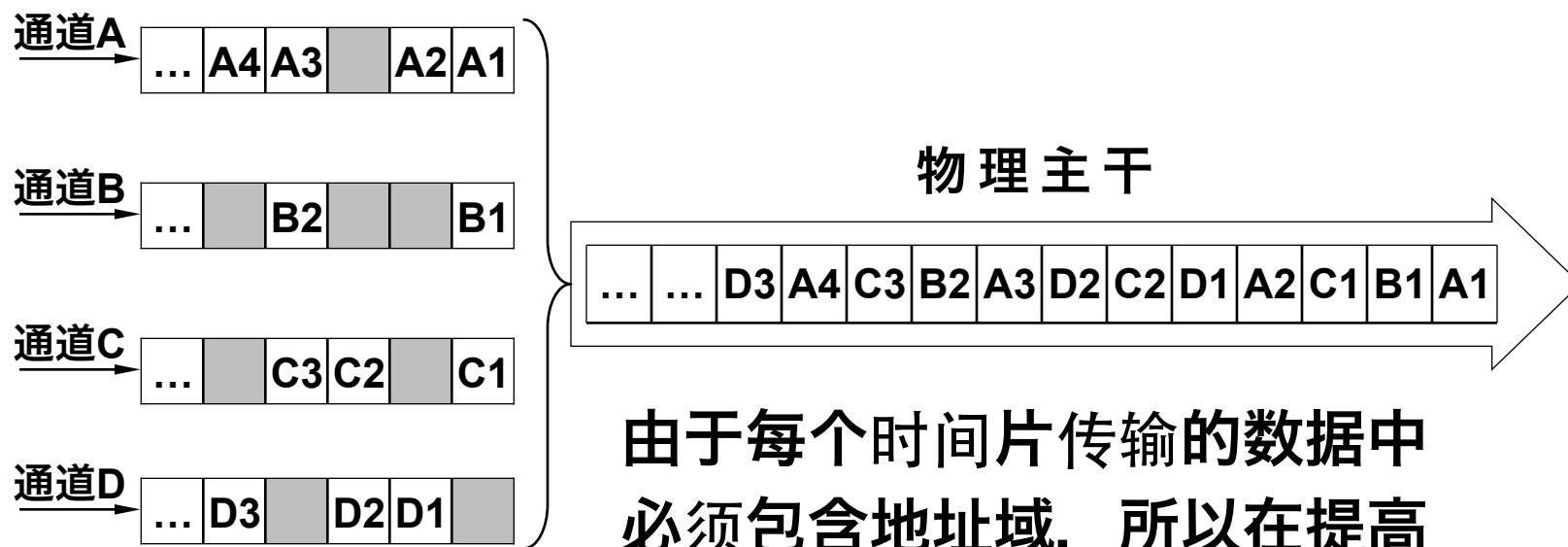
# TDM的同步和异步

## 异步（统计）TDM

- ❖ 时间片是按需动态分配的
- ❖ 时间片与输入装置之间没有对应关系，任何一个时间片都可以被用于传输任何一路输入信号
- ❖ 在传输的数据单元中必须包含地址信息，以便寻址目的节点
- ❖ 传输介质的传输速率只要不低于各个输入信号的平均数据速率即可
- ❖ 异步TDM又称为统计TDM（STDM）



# 异步TDM图示



由于每个时间片传输的数据中必须包含地址域，所以在提高传输介质利用率的同时，也降低了有效数据的传输率



# 传输介质与信道

- ❖ 传输介质与信道是不同范畴的概念
- ❖ 传输介质是指传输信号的物理载体
- ❖ 信道是具有一定带宽的数据通道，强调的是介质的逻辑特性
- ❖ 一根传输介质可能同时提供多个信道
- ❖ 一个信道也可能由多根传输介质级联而成



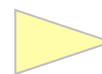


# 交换技术

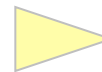
❖ 电路交换（电话）



❖ 报文交换（电报）



❖ 分组交换





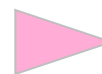
# 电路交换（电话）

- ❖ 在数据传输前，必须建立一条端到端的通路，称为连接，该连接可能穿越多个交换局，此时每个交换局都必须为之提供连接
- ❖ 一旦建立连接，整个通路将被独占，除信号传播的延时之外，数据传输无额外延时，数据中无需包含目的地址
- ❖ 线路的利用率较低
- ❖ 因为连接建立时可能发生冲突，尤其是需要穿越多个交换局的连接，所以建立连接时间可能较长



# 交换技术

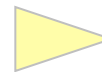
❖ 电路交换（电话）



❖ 报文交换（电报）



❖ 分组交换





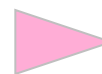
# 报文交换（电报）

- ❖ 无论数据传输过程要跨越多少个交换局（通常是路由器），只要下一站不忙，该数据即送至下一站
- ❖ 数据的传输**毋**需建立连接，数据的传输是一站一站往下送，所以数据中必须包含目的地址，并采用存储-转发（store-forward）机制
- ❖ 线路的利用率较高
- ❖ 由于采用store-forward机制，所以在数据传输过程中，除了信号传播的延时之外，还有存储和转发的延时，而且可能延时较大，且不可估计
- ❖ 每个中间站点都必须有足够大的缓存，但由于报文大小不定，所以缓存通常设置在硬盘中

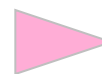


# 交换技术

❖ 电路交换（电话）



❖ 报文交换（电报）



❖ 分组交换



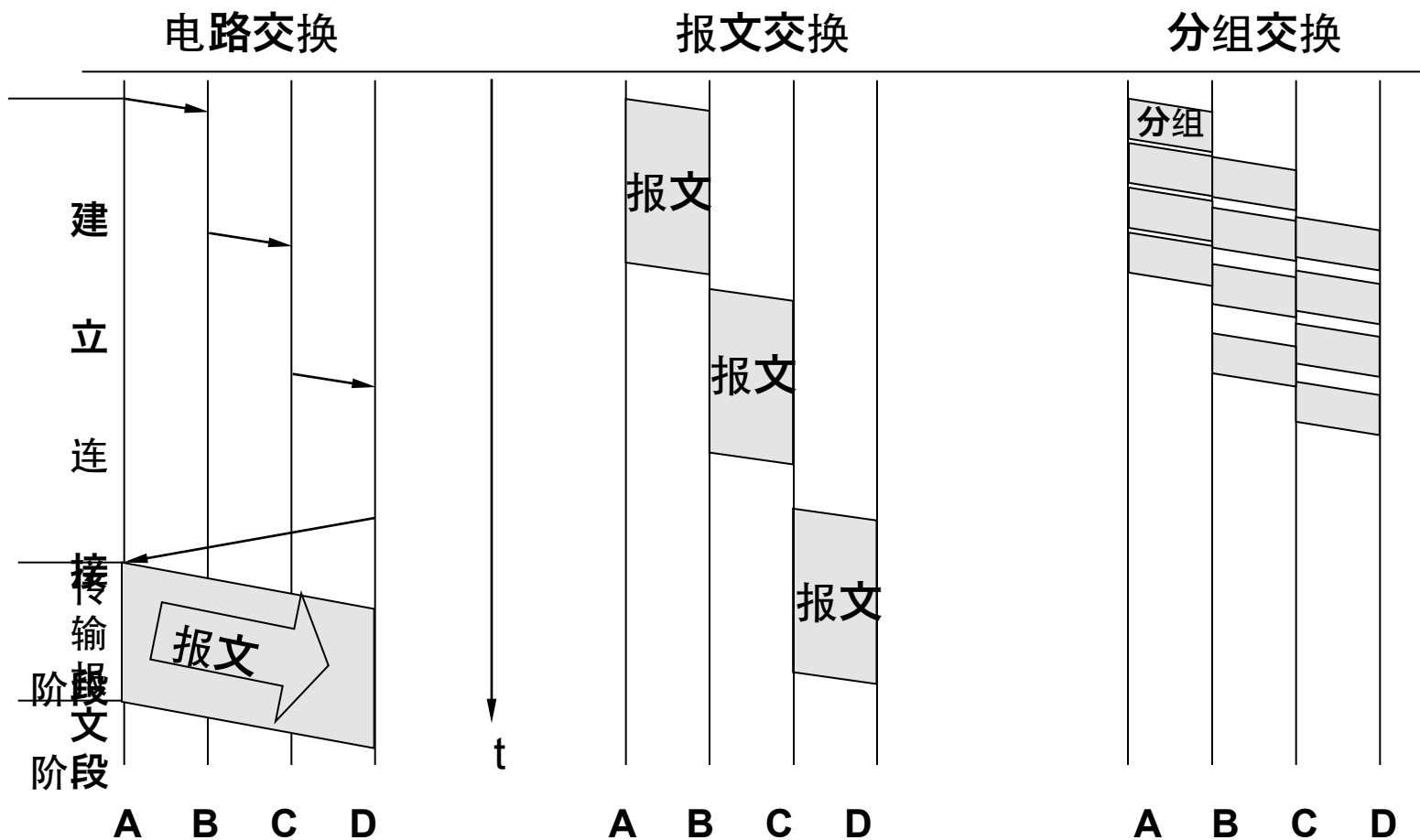


# 分组交换

- ❖ 与报文交换相似，分组交换只是将报文分为若干个定长的分组，每个分组称为一个子报文
- ❖ 每个分组中必须包含目的地址，并采用存储-转发机制
- ❖ 线路的利用率较高
- ❖ 由于采用store-forward机制，所以在数据传输过程中，除了信号传播的延时之外，还有存储和转发的延时，而且可能延时较大，且不可估计
- ❖ 每个中间站点必须有缓存，但由于报文大小固定，所以缓存通常在内存中设置
- ❖ 接收分组和发送分组的顺序可能不一致，并且可能还需要重组



# 交换技术图示



Tnbm P149 Fig. 2-39 各种交换过程的事件时序



# 电路交换和分组交换

计算机网络讨论的是电路交换和分组交换

## ❖ 电路交换

- 在数据传输前，必须建立端到端的连接
- 一旦某个节点故障，必须重新建立连接
- 连接建立后，数据的传输没有额外的延时
- 数据中不必包含地址域，仅需较短的虚电路号
- 数据按序传输，但信道的使用率较低
- 适合长时间传输大批量的数据，如流数据





# 电路交换和分组交换 (续)

## ❖ 分组交换

- 在数据传输前, 不必建立端到端的连接
- 只要下一个节点空闲, 即可传输
- 信道的使用率较高
- 数据的传输采用存储转发, 延时不可估计
- 数据中必须包含地址域
- 接收到的分组不一定按序, 可能还需重组
- 适合传输文本型数据



# 虚电路交换

- ❖ 将电路交换的概念引入到分组传输
- ❖ 虚电路连接的建立

传输方发起连接请求，中间节点根据路径信息建立交换表，在交换表内，节点为连接分配一个虚电路号，并与输出端口号相关联，表示用户信息从该端口输入，立即从相关联的输出端口输出到下一节点

- ❖ 虚电路连接的传输

分组中没有目的地址，只有虚电路号，接收分组时只检查其头部，一旦得到其虚电路号，则立即查交换表，转发至适当的端口（非store-forward）

- ❖ 虚电路连接的拆除



# 信元交换

- ❖ 将分组分成固定长的单元 — 信元,  
并用虚电路方式交换
- ❖ ATM网用这种交换方式



# 数据通信基本概念

|       |         |
|-------|---------|
| 信号和信道 | 位同步和帧同步 |
| 数据的编码 | 串行和并行   |
| 采样和量化 | 复用技术    |
| 数据的传输 | 交换技术    |



# Thanks

**`jdyu@cs.sjtu.edu.cn`**