



石家庄铁道大学  
SHIJIAZHUANG TIEDAO UNIVERSITY

# 计算机网络

## 第 2 讲 网络概述(II)



# 上讲内容回顾

1. 计算机网络基本概念
2. 在网络边缘的端系统中运行的程序间的两类通信方式：客户服务器方式（C/S 方式）和对等方式（P2P 方式）
3. 电路交换技术和分组交换技术（分组交换技术的特点及优点）
4. 计算机网络的基本分类



# 本讲内容

## 1.6 计算机网络的性能

### 1.6.1 计算机网络的性能指标

### 1.6.2 计算机网络的非性能特征

## 1.7 计算机网络的体系结构

### 1.7.1 计算机网络体系结构的形成

### 1.7.2 协议与划分层次

### 1.7.3 具有五层协议的体系结构

### 1.7.4 实体、协议、服务和服务访问点

### 1.7.5 TCP/IP 的体系结构



# 1.6 计算机网络的性能

## 1.6.1 计算机网络的性能指标

### 1. 速率

- **比特** (bit) 是计算机中数据量的单位，也是信息论中使用的信息量的单位。
- Bit 来源于 binary digit，意思是一个“**二进制数字**”，因此一个比特就是二进制数字中的一个 1 或 0。
- **速率**即**数据率** (data rate) 或**比特率** (bit rate) 是计算机网络中最重要的一個性能指标。速率的单位是 b/s，或 kb/s，Mb/s，Gb/s 等。
- 速率往往是指**额定速率**或**标称速率**。



## 2. 带宽

- “**带宽**” (bandwidth) 本来是指信号具有的频带宽度，单位是赫（或千赫、兆赫、吉赫等）。
- 现在“带宽”是数字信道所能传送的“**最高数据率**”的同义语，单位是“比特每秒”，或 b/s (bit/s)。



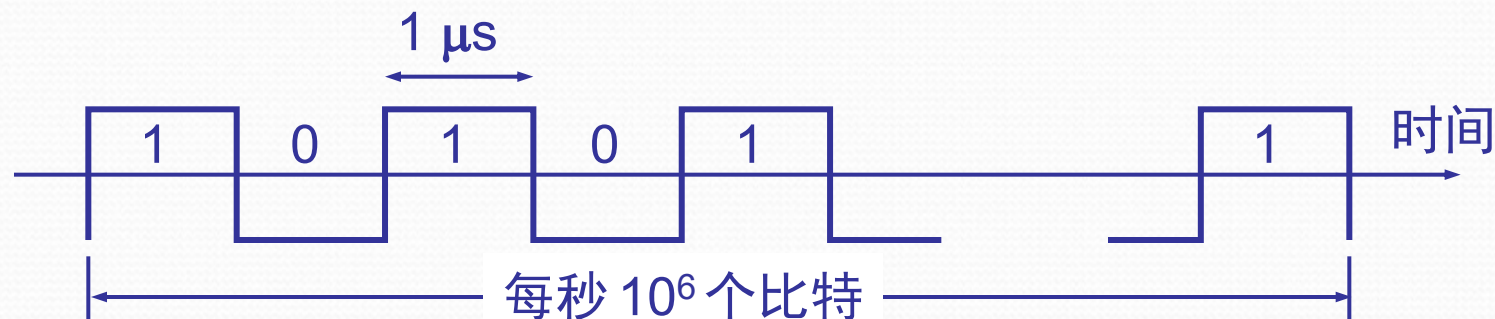
# 常用的带宽单位

- 更常用的带宽单位是
  - 千比每秒，即  $\text{kb/s}$  ( $10^3 \text{ b/s}$ )
  - 兆比每秒，即  $\text{Mb/s}$  ( $10^6 \text{ b/s}$ )
  - 吉比每秒，即  $\text{Gb/s}$  ( $10^9 \text{ b/s}$ )
  - 太比每秒，即  $\text{Tb/s}$  ( $10^{12} \text{ b/s}$ )
- 请注意：在计算机界， $\text{K} = 2^{10} = 1024$   
 $\text{M} = 2^{20}$ ,  $\text{G} = 2^{30}$ ,  $\text{T} = 2^{40}$ 。

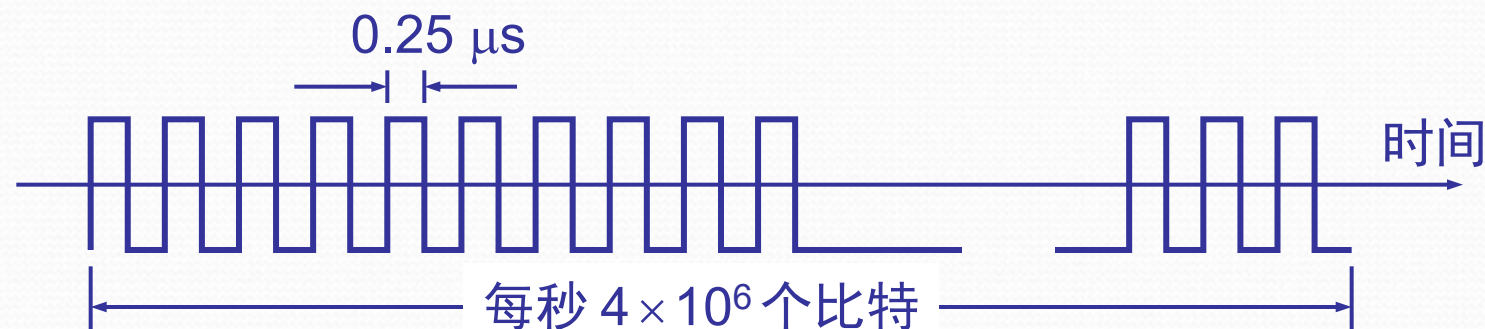
# 数字信号流随时间的变化

- 在时间轴上信号的宽度随带宽的增大而变窄。

带宽为  
1 Mb/s



带宽为  
4 Mb/s





### 3. 吞吐量

- **吞吐量** (throughput) 表示在单位时间内通过某个网络（或设备、信道、接口）的数据量。
- 吞吐量更经常地用于对现实世界中的网络的一种测量，以便知道实际上到底有多少数据量能够通过网络。
- 吞吐量受网络的带宽或网络的额定速率的限制，一般小于带宽。
- 注意：1、设备的吞吐量是处理数据的能力  
2、有的吞吐量单位是字节或数据报数量





## 4. 时延(delay 或 latency)

- **传输时延**（发送时延） 发送数据时，数据块从结点进入到传输媒体（介质）所需要的时间。
- 也就是从发送数据帧的第一个比特算起，到该帧的最后一个比特发送完毕所需的时间。

$$\text{发送时延} = \frac{\text{数据块长度（比特）}}{\text{信道带宽（比特/秒）}}$$



# 时延(delay 或 latency)

- **传播时延** 电磁波在信道中需要传播一定的距离而花费的时间。
- **信号传输速率**（即发送速率）和信号在信道上的**传播速率**是完全不同的概念。

$$\text{传播时延} = \frac{\text{信道长度 (米)}}{\text{信号在信道上的传播速率 (米/秒)}}$$



# 时延(delay 或 latency)

- **处理时延** 交换结点为存储转发而进行一些必要的处理所花费的时间。
- **排队时延** 结点缓存队列中分组**排队**所经历的时延。
- 排队时延的长短往往取决于网络中**当时的通信量**。





# 时延(delay 或 latency)

- 数据经历的总时延就是发送时延、传播时延、处理时延和排队时延之和：

总时延 = 发送时延 + 传播时延 + 处理时延 + 处理时延

# 四种时延所产生的地方

从结点 A 向结点 B 发送数据

在结点 A 中产生  
处理时延和排队时延

在发送器产生传输时延  
(即发送时延)

在链路上产生  
传播时延





# 容易产生的错误概念

- 对于高速网络链路，我们提高的仅仅是数据的**发送速率**而不是比特在链路上的**传播速率**。
- 提高链路带宽减小了数据的发送时延。



## 5. 时延带宽积

时延带宽积

(传播) 时延

带宽



$$\text{时延带宽积} = \text{传播时延} \times \text{带宽}$$

- 链路的时延带宽积又称为以**比特**为单位的链路长度。



## 6. 利用率

- **信道利用率**指出某信道有百分之几的时间是被利用的（有数据通过）。完全空闲的信道的利用率是零。
- **网络利用率**则是全网络的信道利用率的加权平均值。
- 信道利用率并非越高越好。



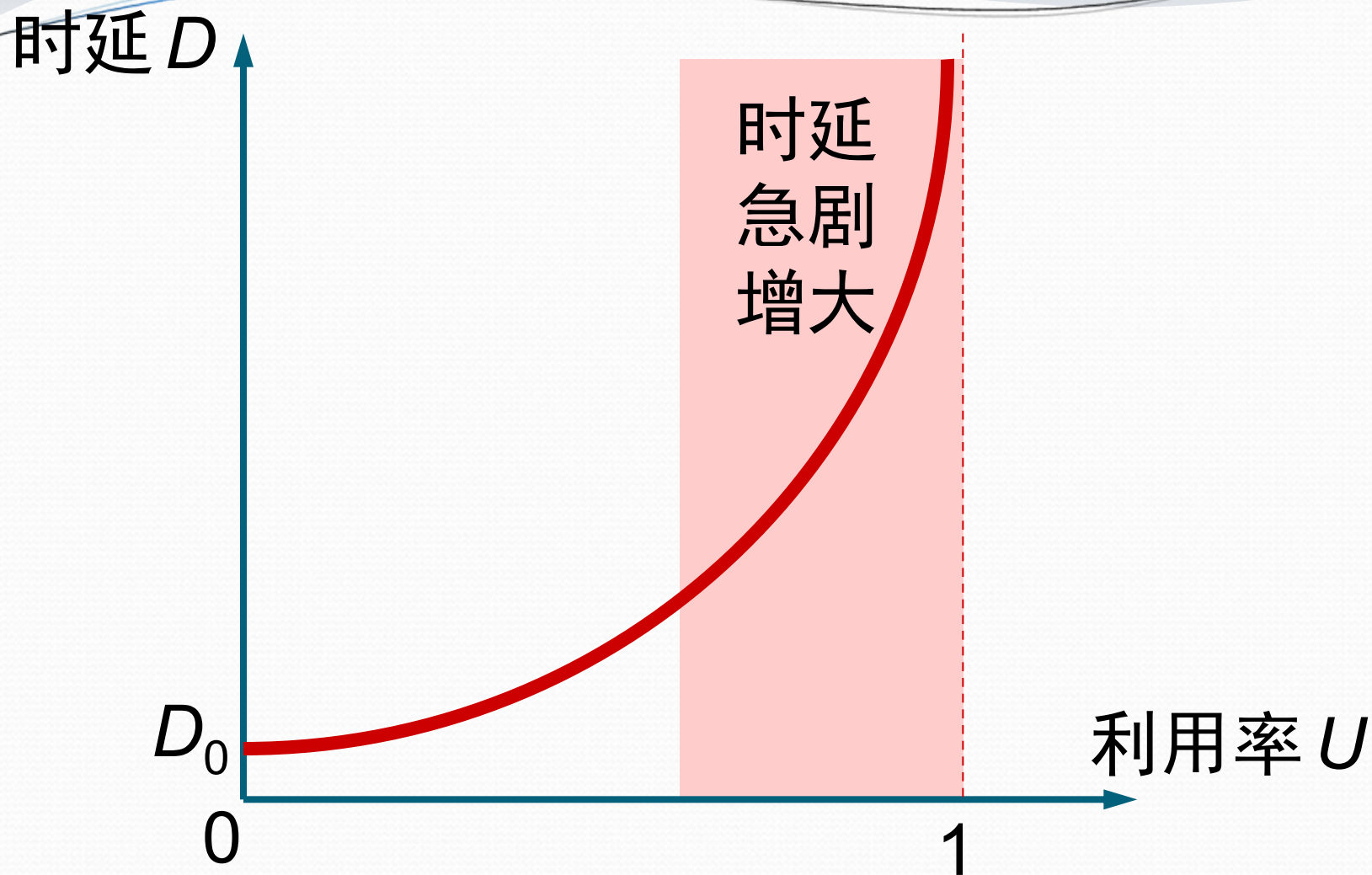
# 时延与网络利用率的关系

- 根据排队论的理论，当某信道的利用率增大时，该信道引起的时延也就迅速增加。
- 若令  $D_0$  表示网络空闲时的时延， $D$  表示网络当前的时延，则在适当的假定条件下，可以用下面的简单公式表示  $D$  和  $D_0$  之间的关系：

$$D = \frac{D_0}{1 - U}$$

$U$  是网络的利用率，数值在 0 到 1 之间。







## 7. 往返时间

- **往返时间** RTT (Round-Trip Time) 表示从发送方发送数据开始，到接收到接收方的确认信息经历的时间。
- 很多协议是发送数据后，等若干时间，如果没有接收到接收方的确认信息就会重新发送一遍。



## 1.6.2 计算机网络的非性能特征

- 费用
- 质量
- 标准化
- 可靠性
- 可扩展性和可升级性
- 易于管理和维护



# 1.7 计算机网络的体系结构

## 1.7.1 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。
- 从网络技术一开始就是按层次开发的，只不过各个技术的层次划分不同而已。

# 关于开放系统互连参考模型OSI/RM

- 1983年ISO公布了OSI/RM。开放系统互连参考模型，即七层协议。
- 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。





## 1.7.2 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。

# 网络协议的组成要素

- **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。  
(有的资料上称为**时序**)
- 可以形象地解释这3要素为: **如何做? 做什么? 什么时候做?**



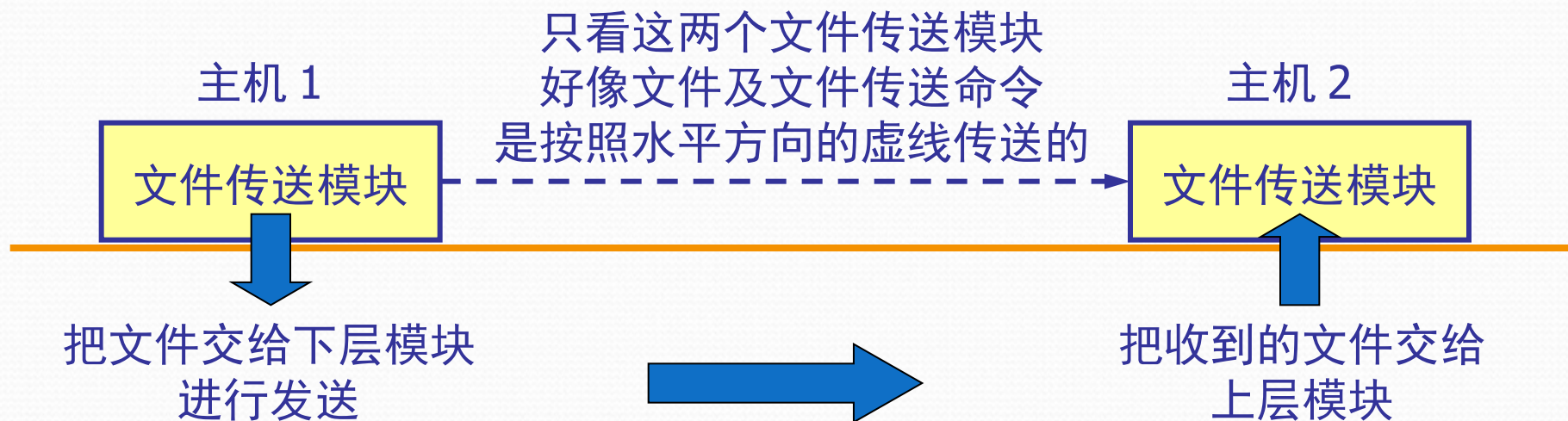


# 划分层次的概念举例

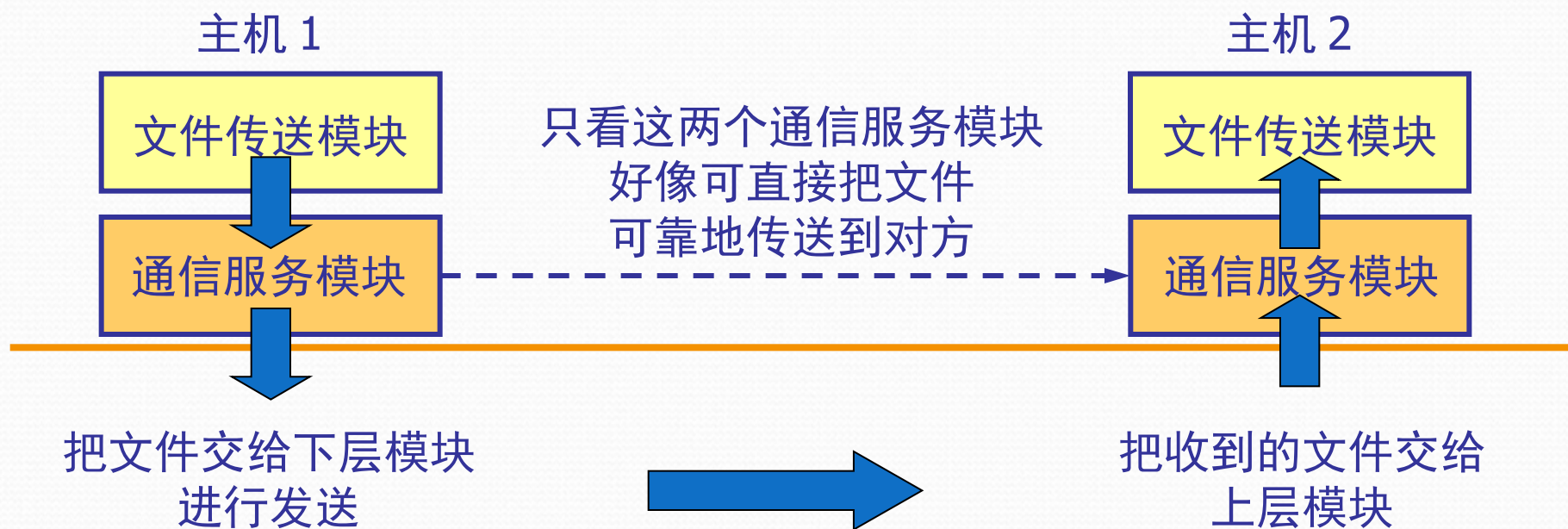
- 主机1向主机2通过网络发送文件。
- 可以将要做的工作进行如下的划分。
- 第一类工作与传送文件直接有关。
  - 确信对方已做好接收和存储文件的准备。
  - 双方协调好一致的文件格式。
- 两个主机将文件传送模块作为最高的一层。  
剩下的工作由下面的模块负责。



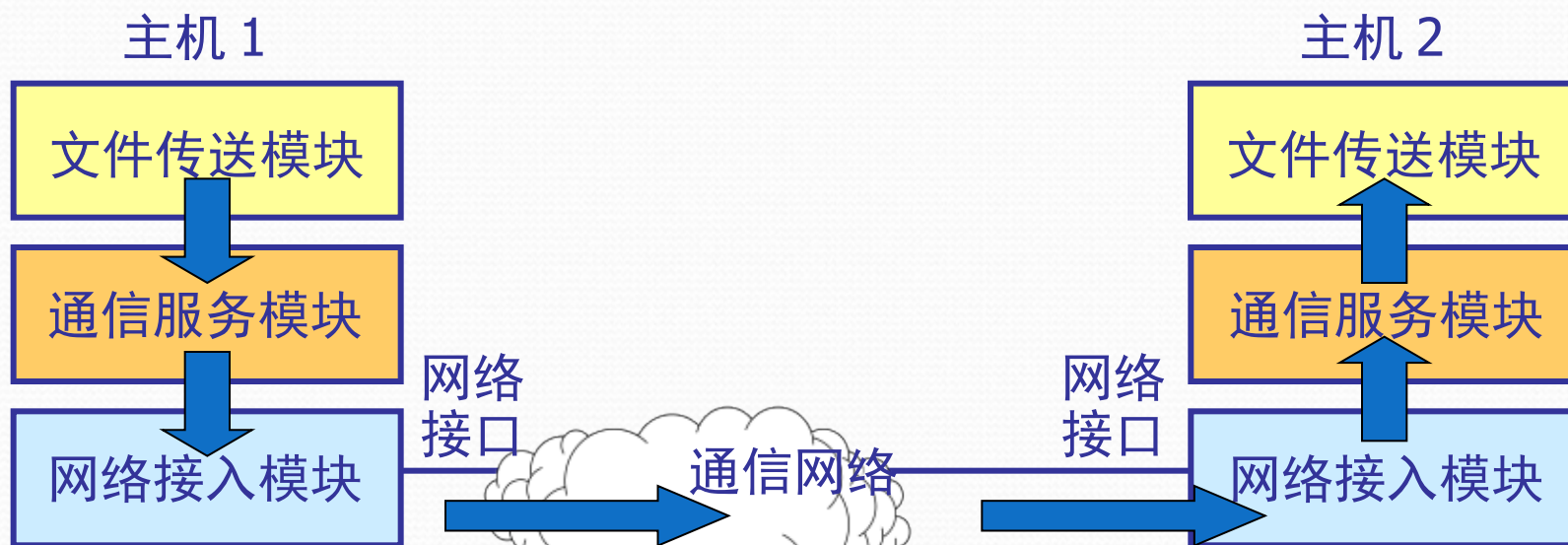
# 两个主机交换文件



# 再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。

1. ~~如何~~ 什么是边缘智能，核心简单，如何理解  
2. 什么是数据交换，信息交换，数据交换，互操作性？



# 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。

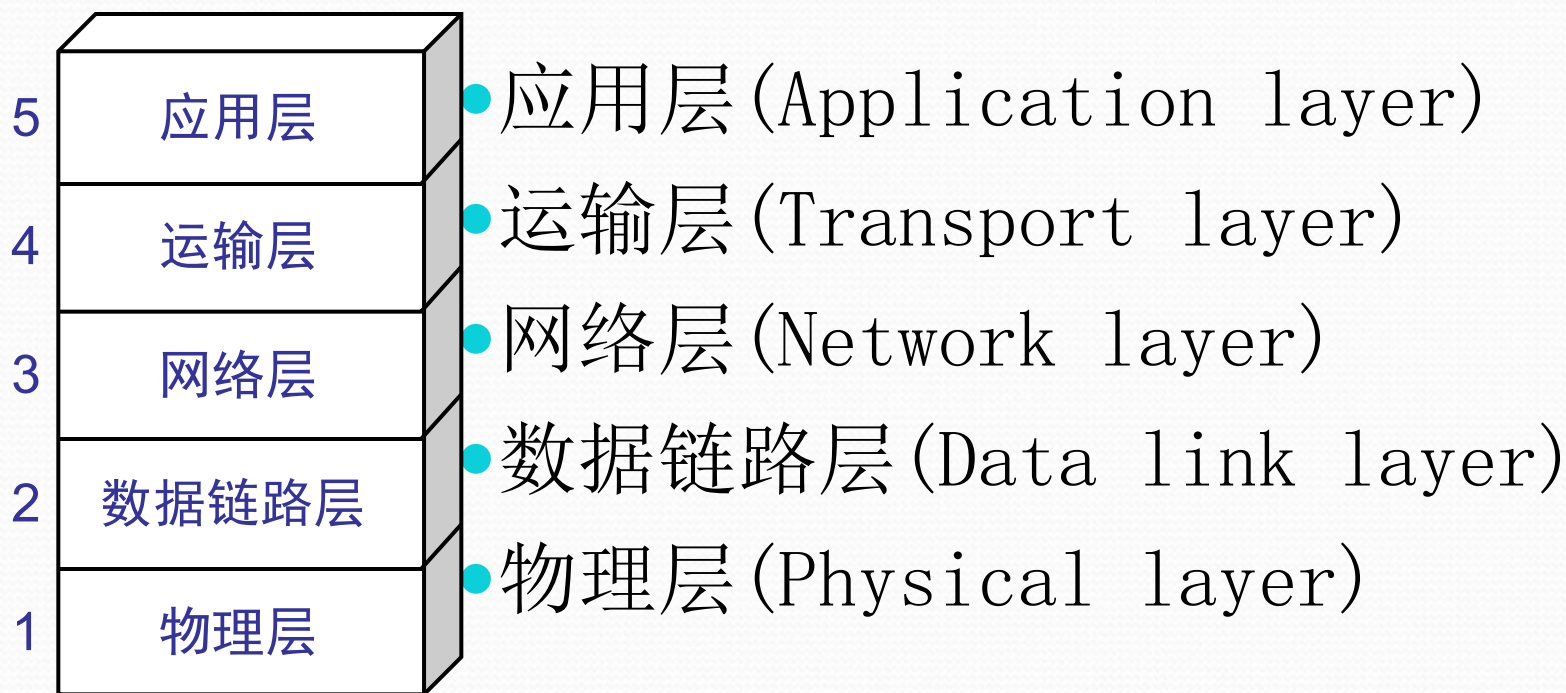




## 1.7.3 具有五层协议的体系结构

- TCP/IP 是四层的体系结构：应用层、运输层、网际层和网络接口层。
- 但最下面的网络接口层并没有具体内容。
- 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。（见P27）

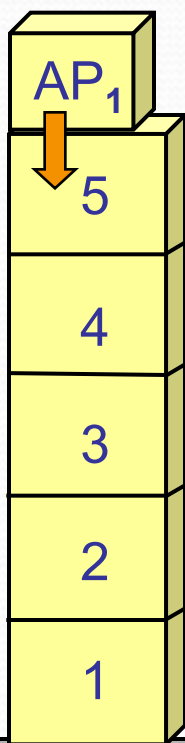
# 五层协议的体系结构





# 主机 1 向主机 2 发送数据

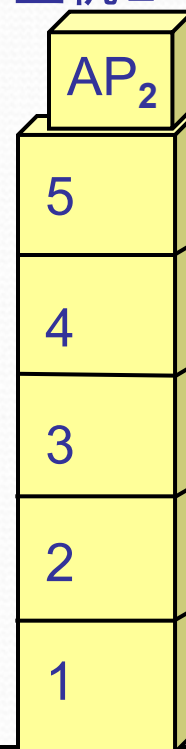
主机 1



应用进程数据先传送到应用层

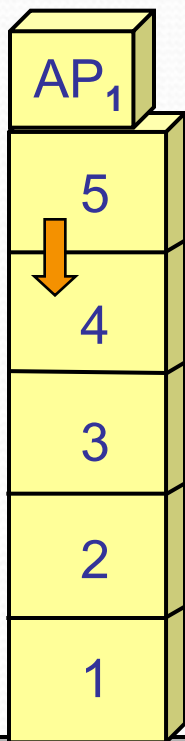
加上应用层首部，成为应用层 PDU

主机 2



# 主机 1 向主机 2 发送数据

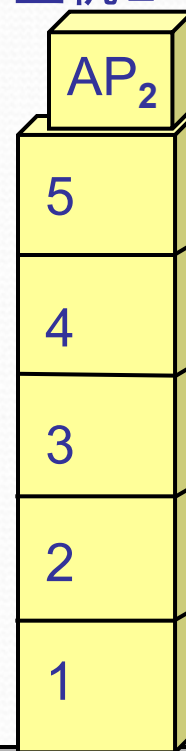
主机 1



应用层 PDU 再传送到运输层

加上运输层首部，成为运输层报文

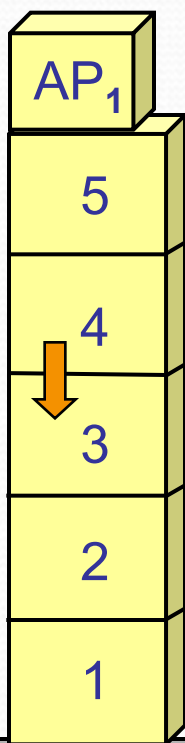
主机 2





# 主机 1 向主机 2 发送数据

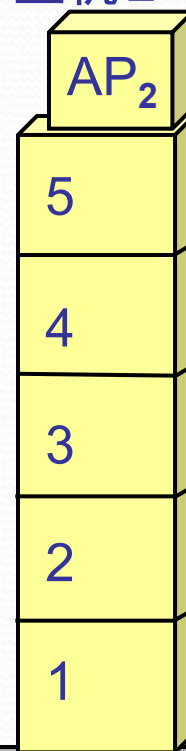
主机 1



运输层报文再传送到网络层

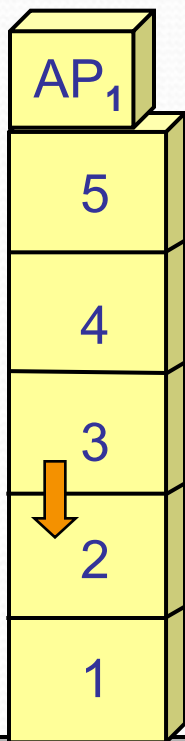
加上网络层首部，成为 IP 数据报（或分组）

主机 2



# 主机 1 向主机 2 发送数据

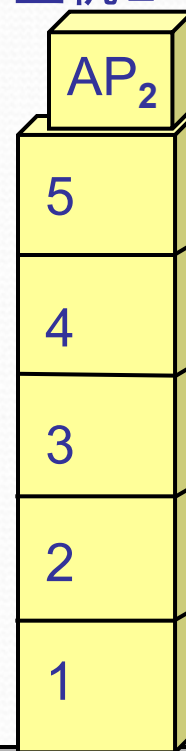
主机 1



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 2

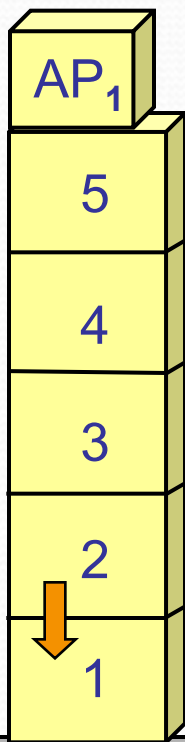






# 主机 1 向主机 2 发送数据

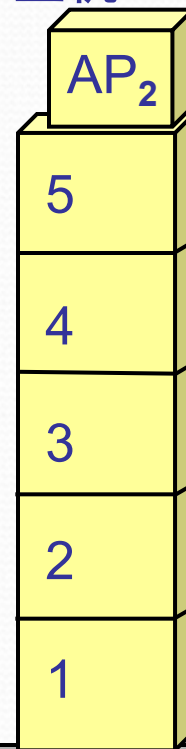
主机 1



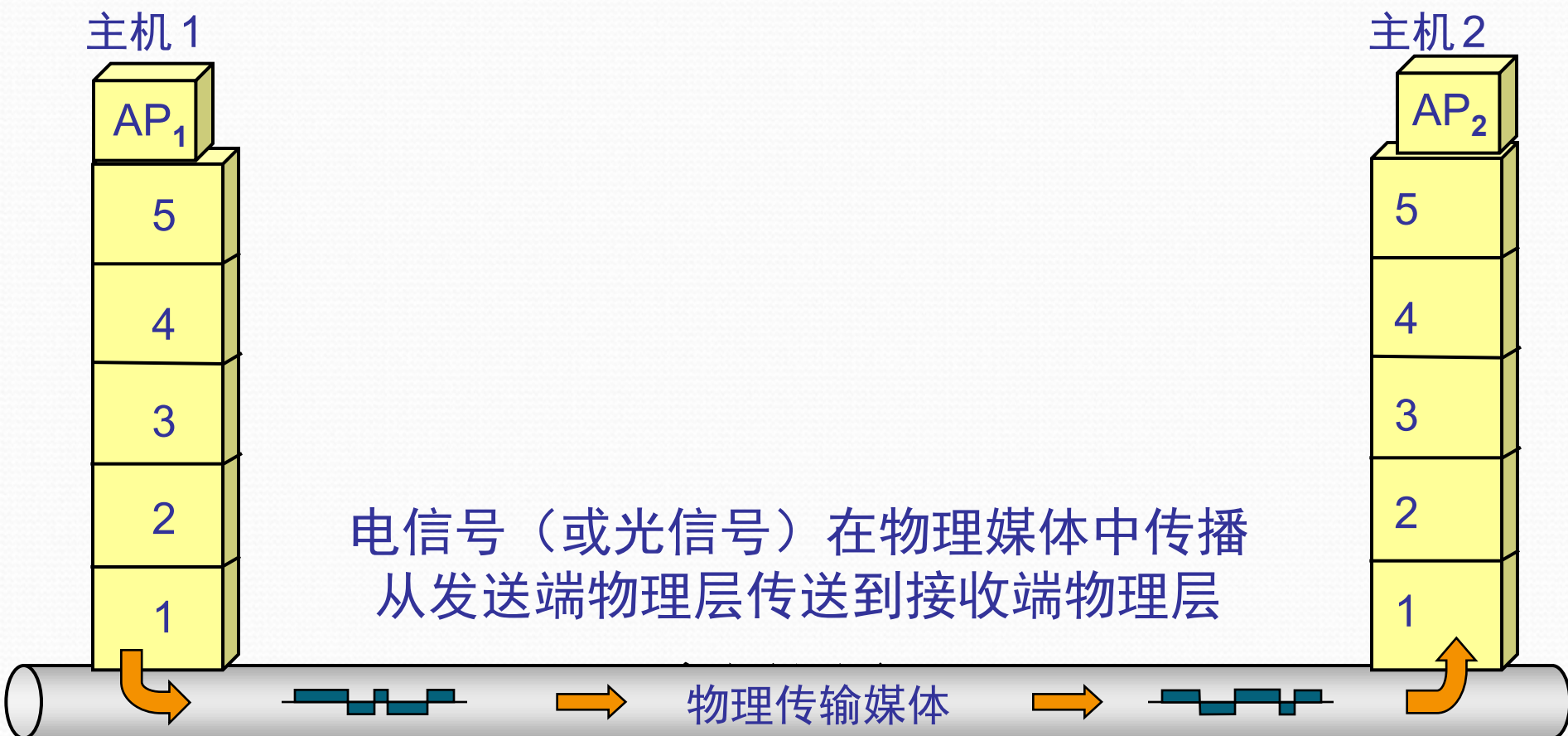
数据链路层帧再传送到物理层

最下面的物理层把比特流传送到物理媒体

主机 2



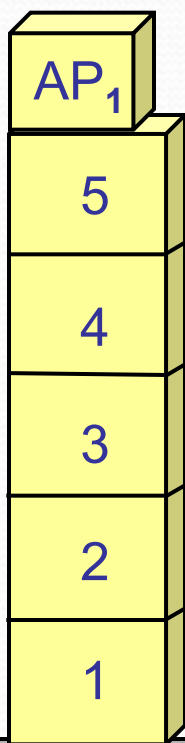
# 主机 1 向主机 2 发送数据



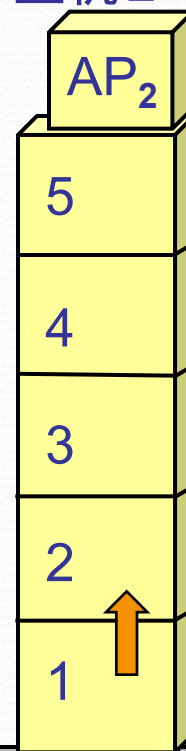


# 主机 1 向主机 2 发送数据

主机 1



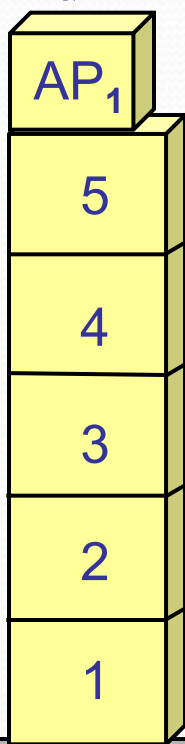
主机 2



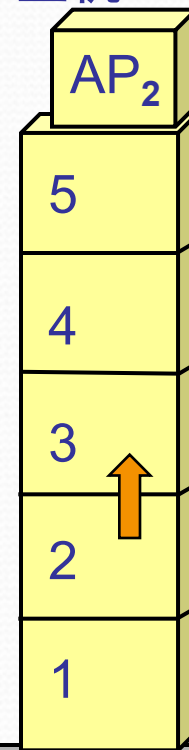
物理层接收到比特流，上交给数据链路层

# 主机 1 向主机 2 发送数据

主机 1



主机 2

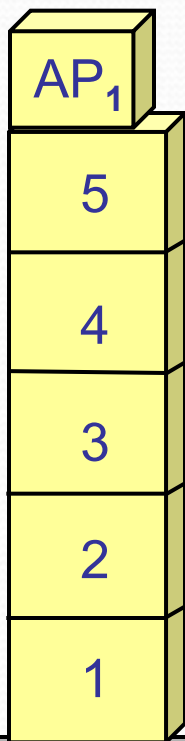


数据链路层剥去帧首部和帧尾部  
取出数据部分，上交给网络层

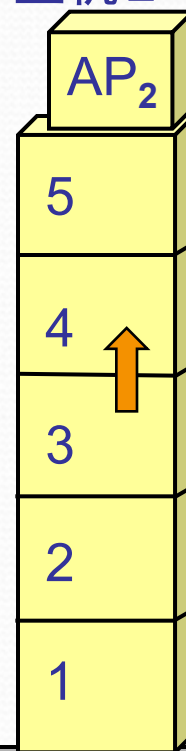


# 主机 1 向主机 2 发送数据

主机 1



主机 2

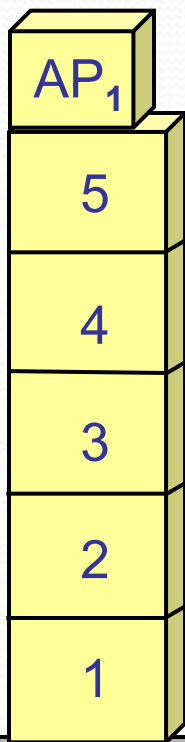


网络层剥去首部，取出数据部分  
上交给运输层

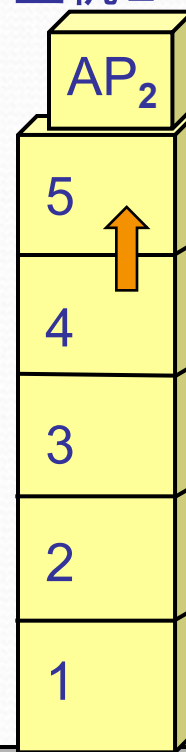


# 主机 1 向主机 2 发送数据

主机 1



主机 2

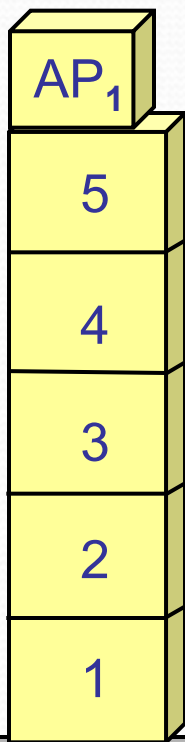


运输层剥去首部，取出数据部分  
上交给应用层



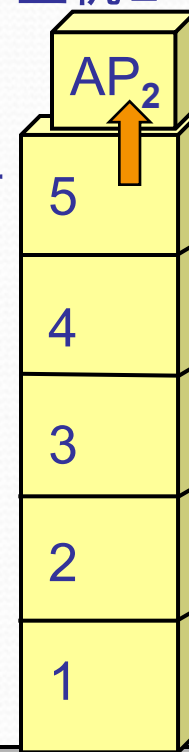
# 主机 1 向主机 2 发送数据

主机 1



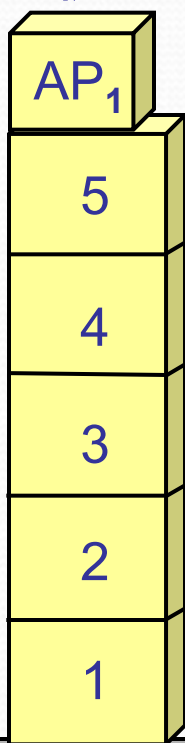
应用层剥去首部，取出应用程序数据  
上交给应用进程

主机 2



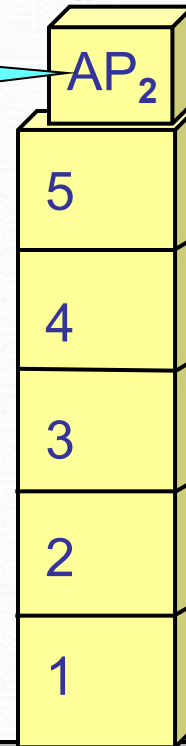
# 主机 1 向主机 2 发送数据

主机 1

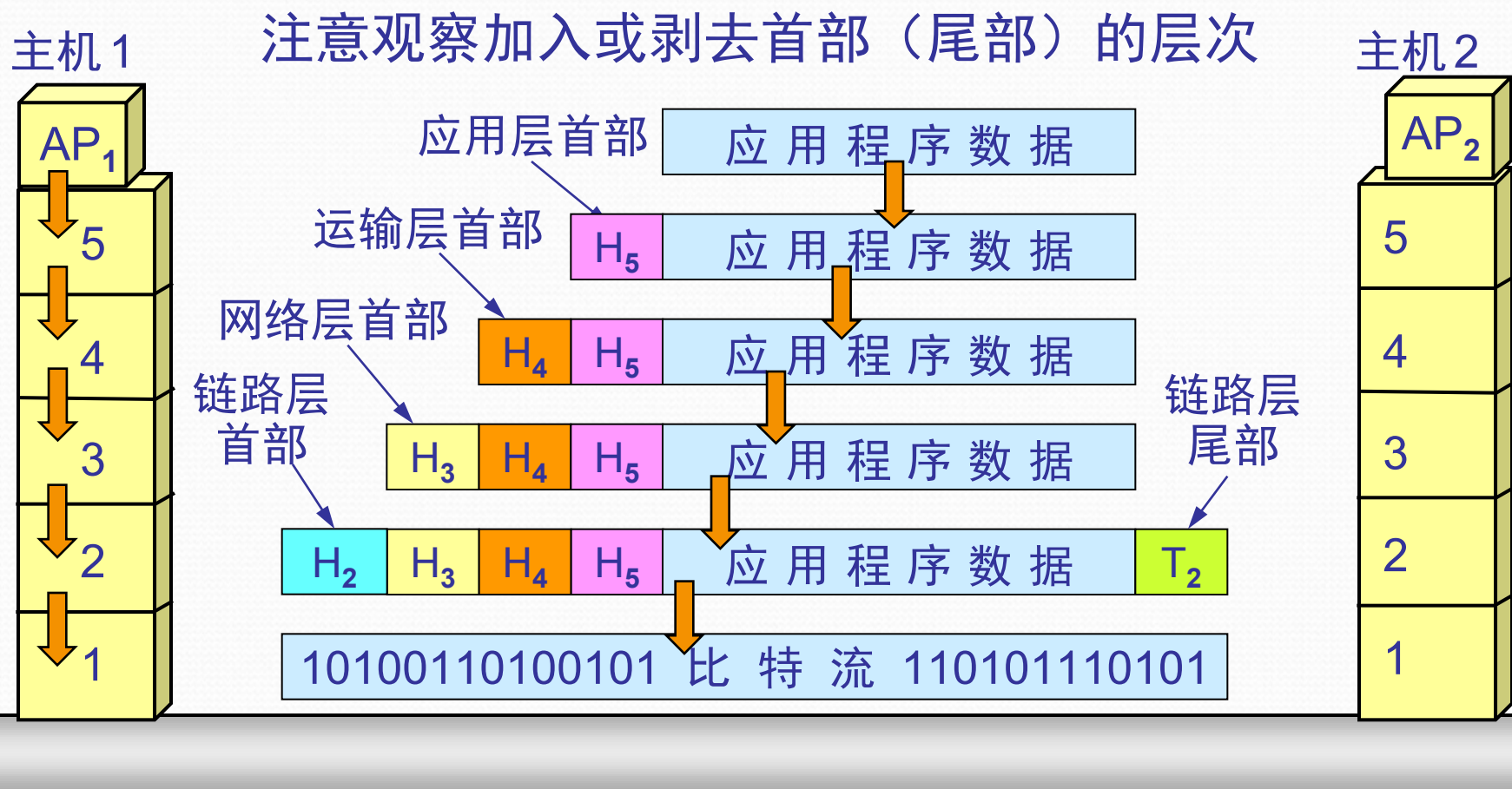


我收到了  $AP_1$  发来的  
应用程序数据！

主机 2



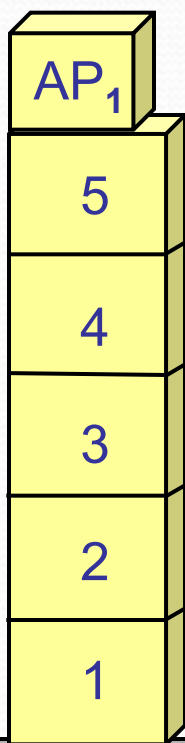
# 主机 1 向主机 2 发送数据



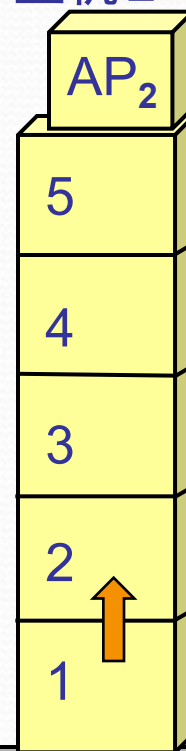


# 主机 1 向主机 2 发送数据

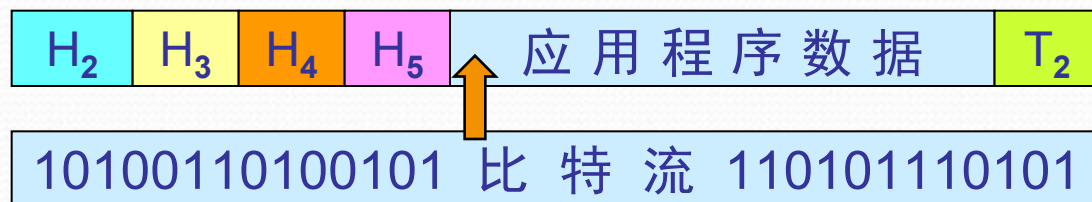
主机 1



主机 2

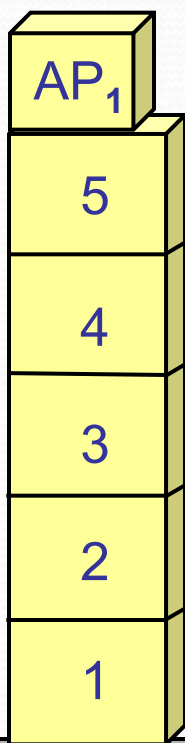


计算机 2 的物理层收到比特流后  
交给数据链路层

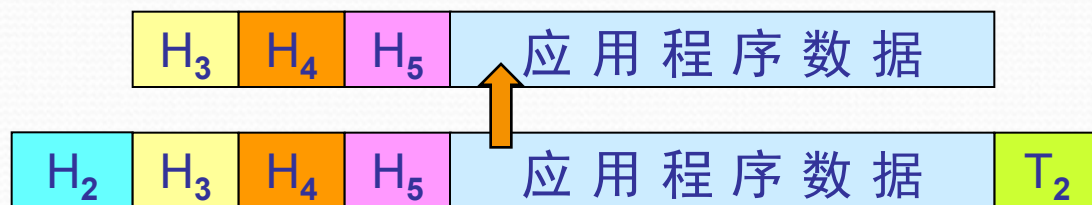


# 主机 1 向主机 2 发送数据

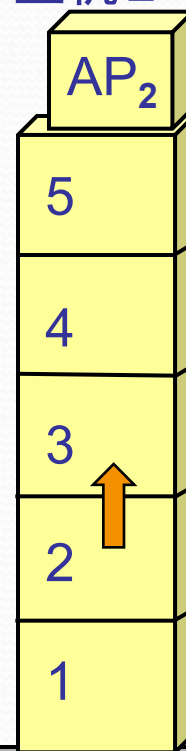
主机 1



数据链路层剥去帧首部和帧尾部后  
把帧的数据部分交给网络层

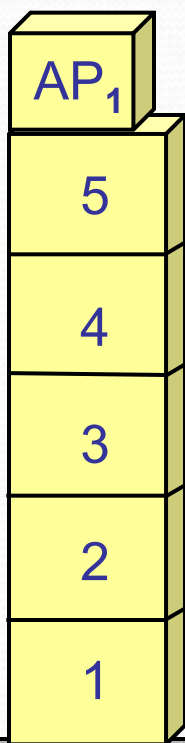


主机 2

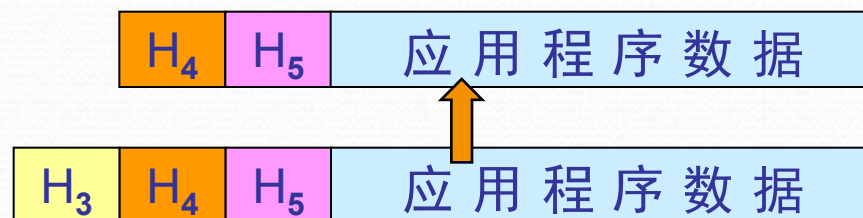


# 主机 1 向主机 2 发送数据

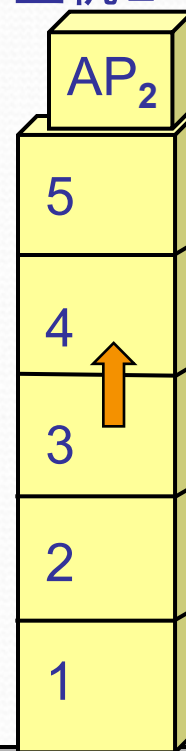
主机 1



网络层剥去分组首部后  
把分组的数据部分交给运输层



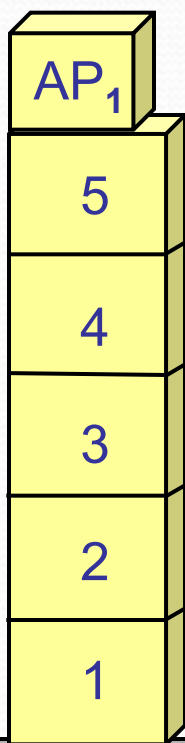
主机 2



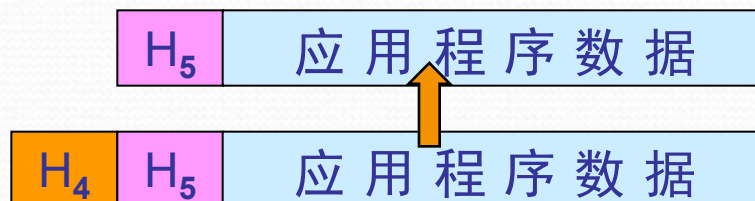


# 主机 1 向主机 2 发送数据

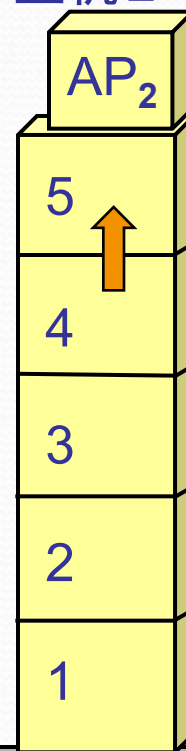
主机 1



运输层剥去报文首部后  
把报文的数据部分交给应用层

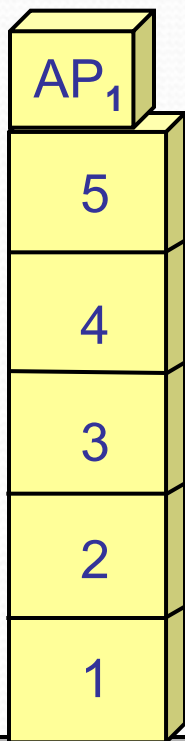


主机 2



# 主机 1 向主机 2 发送数据

主机 1



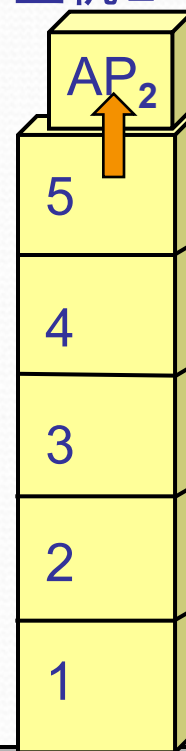
应用程序数据

$H_5$

应用程序数据

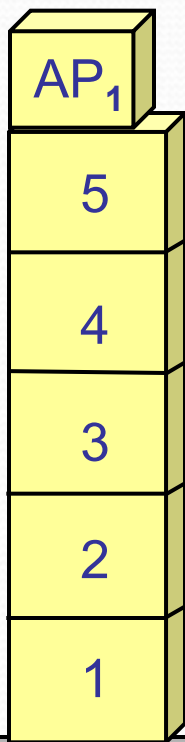
应用层剥去应用层 PDU 首部后  
把应用程序数据交给应用进程

主机 2



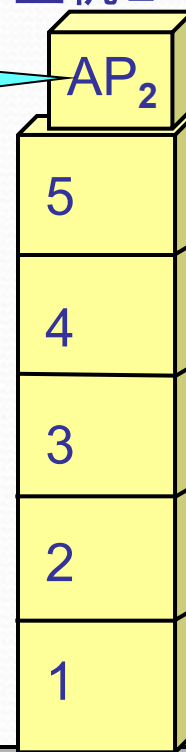
# 主机 1 向主机 2 发送数据

主机 1



我收到了  $AP_1$  发来的  
应用程序数据！

主机 2







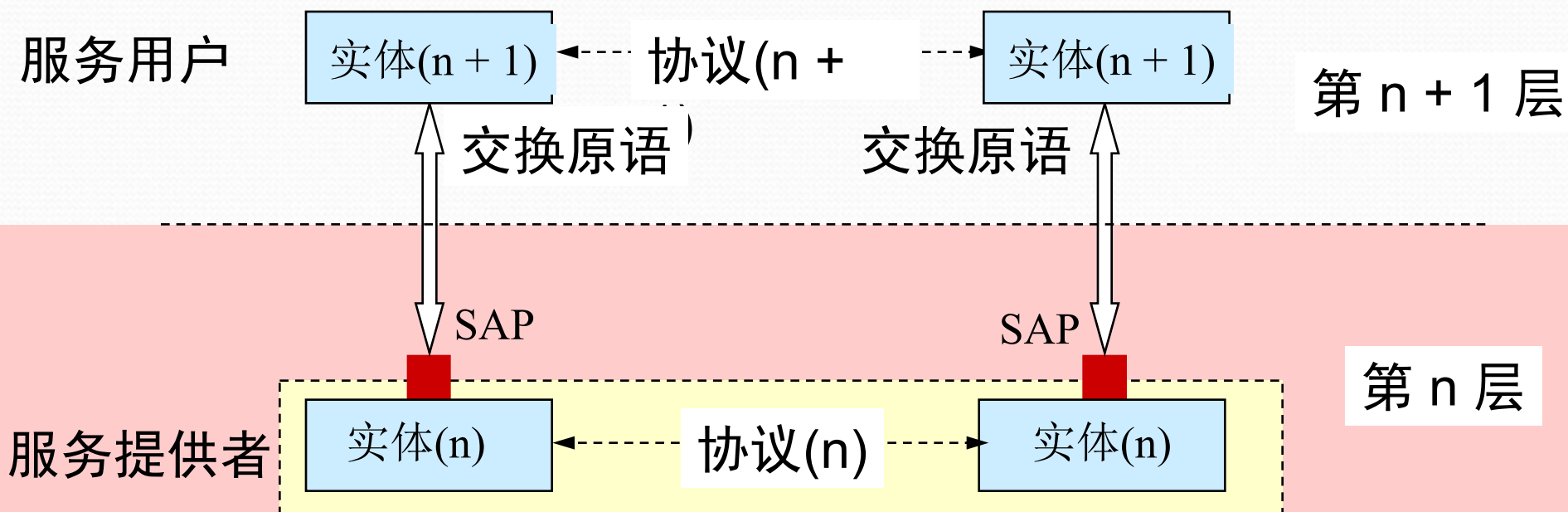
## 1.7.4 实体、协议、服务 和服务访问点

- **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）





## 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 【例1-1】

- 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。于是用计算机发送电文给蓝军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。但此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？

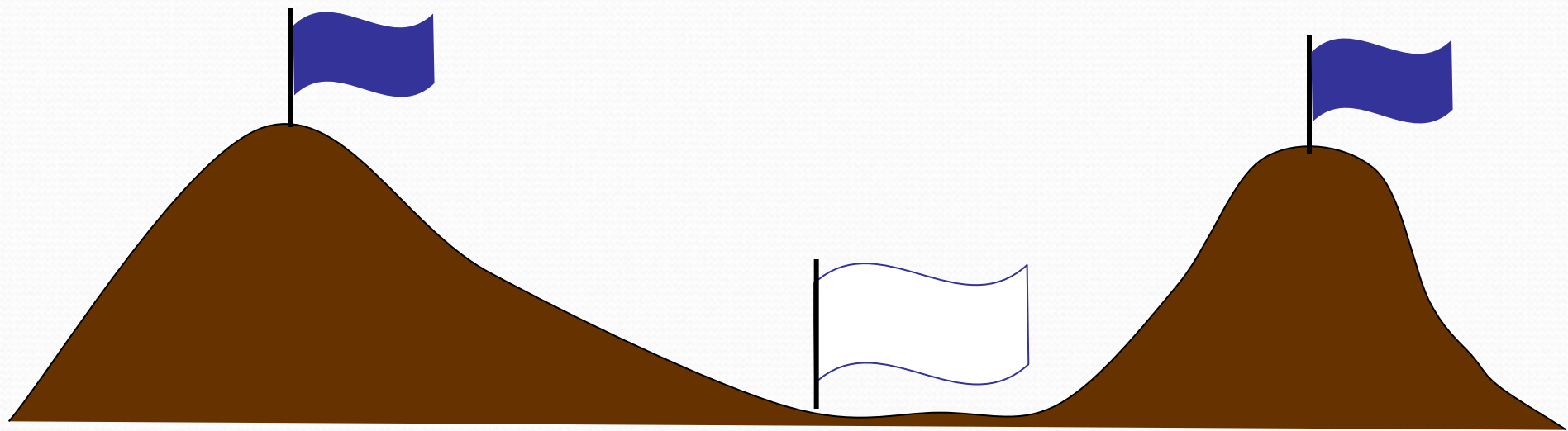
明日正午进攻，如何？

同意

收到“同意”

这样的协议无法实现！

收到：收到“同意”



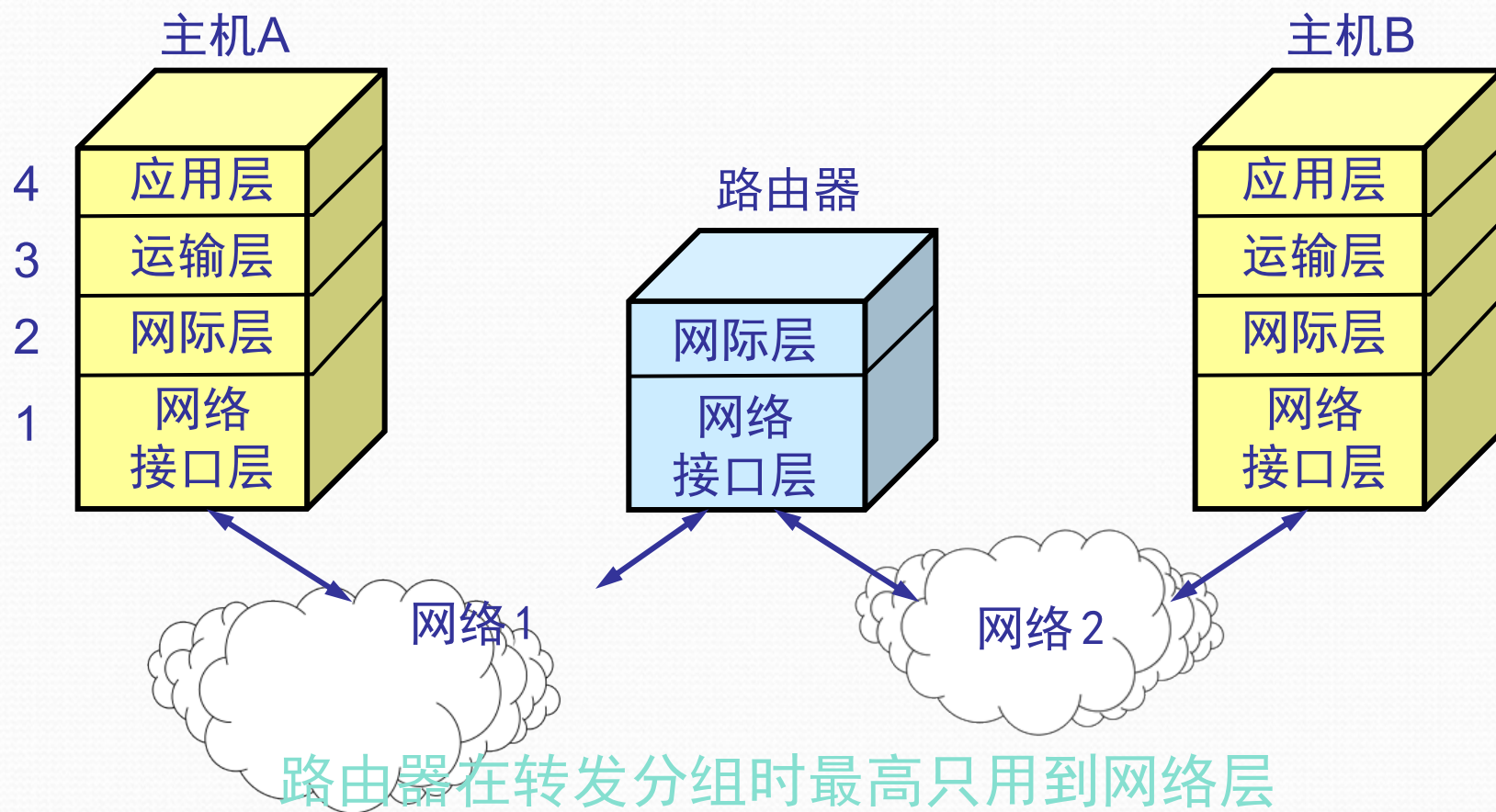




## 结论

- 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。
- 没有一种协议能够蓝军能 100% 获胜。

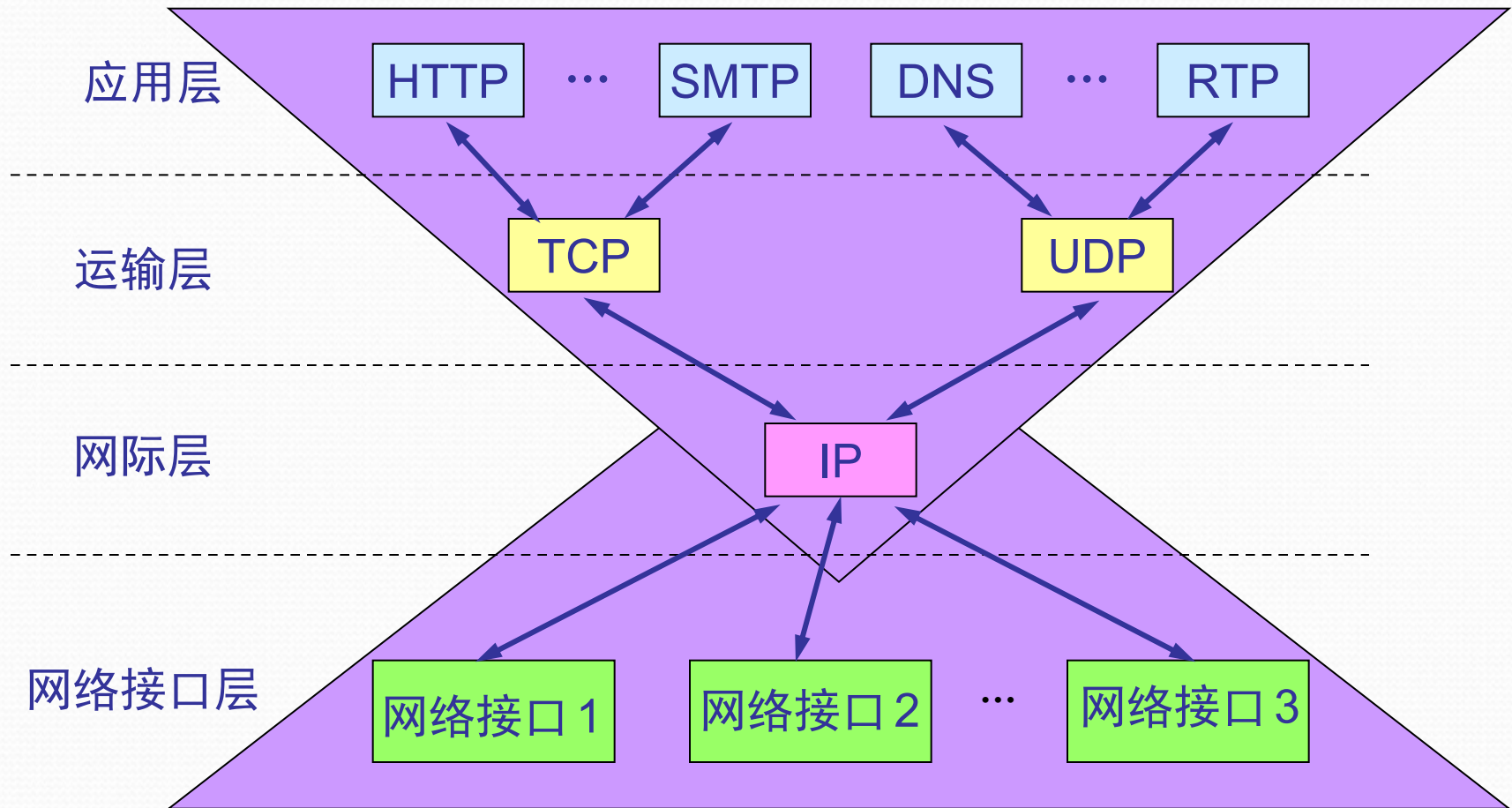
## 1.7.5 TCP/IP的体系结构



路由器在转发分组时最高只用到网络层  
而没有使用运输层和应用层。

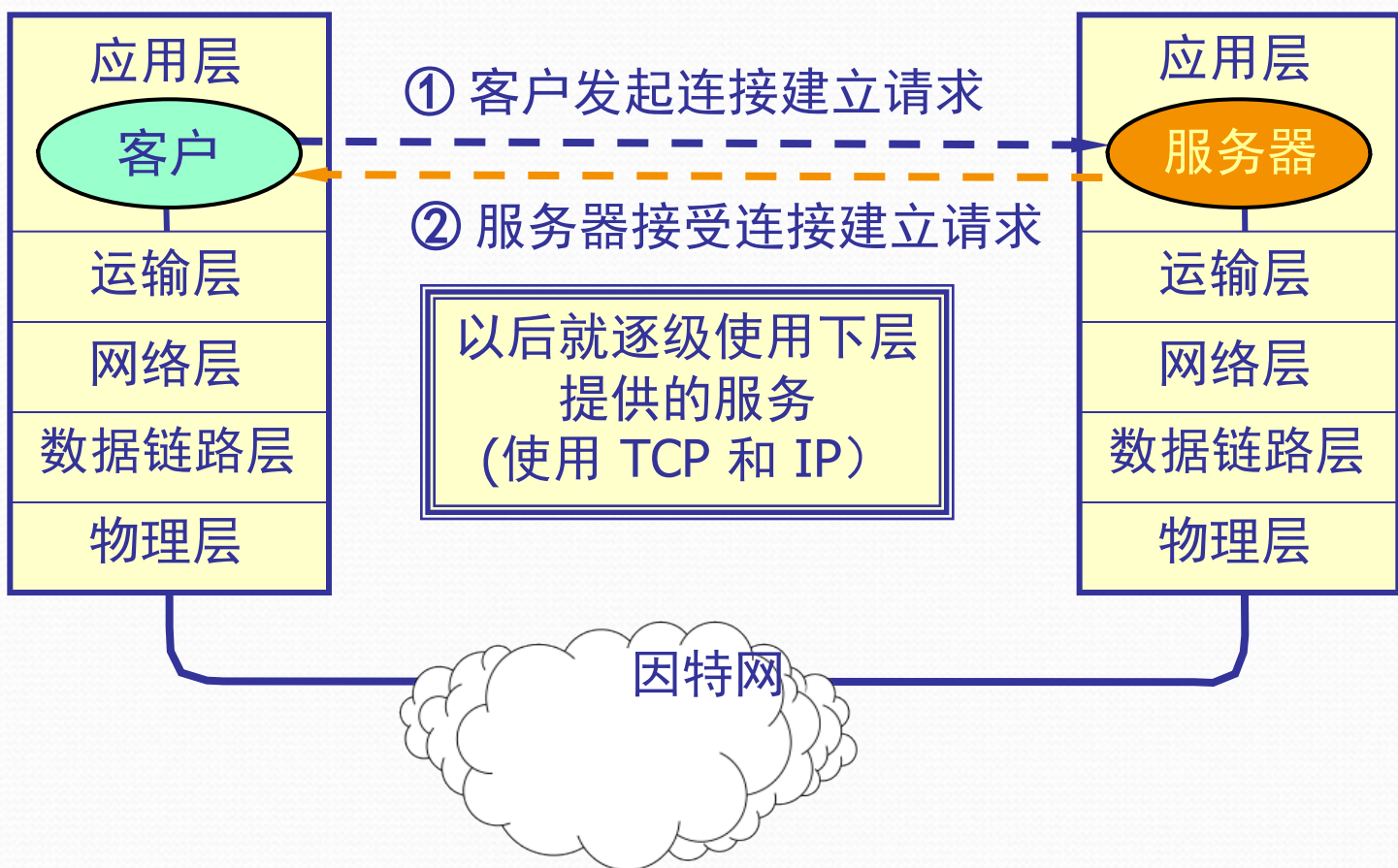
# IP over Everything

## IP可应用到各式各样的网络上

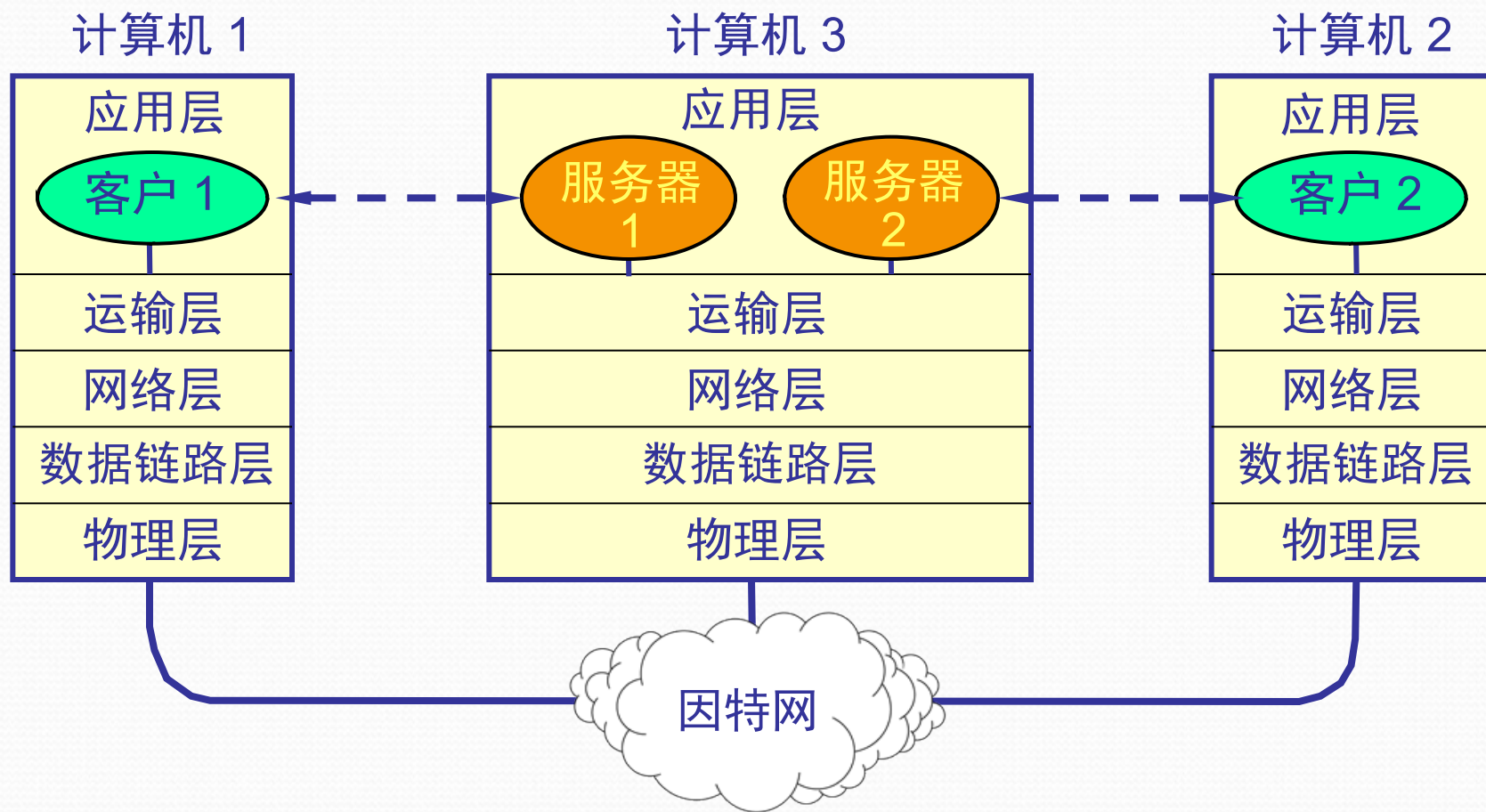




## 【例1-2】客户进程和服务进程使用 TCP/IP 协议进行通信



# 功能较强的计算机可同时运行多个服务器进程





# 本讲总结

## 1. 计算机网络性能

- ① 速率
- ② 带宽
- ③ 时延（重点）
- ④ 利用率

## 2. 计算机网络体系结构

OSI七层体系结构与TCP/IP的四（五）层





# 作业

- 思考题：  
P34： 1-20、 1-24、 1-26、
- 作业题：  
P35： 1-15、 1-17、 1-19