



江苏师范大学  
JIANGSU NORMAL UNIVERSITY

电气工程及自动化学院  
SCHOOL OF ELECTRICAL ENGINEERING & AUTOMATION

# 计算机网络技术

授课教师：李灿

联系方式：57862787

[lic@jsnu.edu.cn](mailto:lic@jsnu.edu.cn)

课程网站：[sslic.cn/cnet](http://sslic.cn/cnet)

教研室：12#-407A（轨道交通系）



# 第3章 计算机网络体系结构



# 本章内容

---

- 网络体系结构
- 网络参考模型
- 五层网络参考模型



# 3.1 网络体系结构

---

- 发展历程
- 分层原理
- 基本概念



## 3.1.1 发展历程

- 网络体系结构提出的背景——计算机网络的复杂性、异质性
  - 不同的通信介质——有线、无线等
  - 不同种类的设备——主机、路由器、交换机、复用设备等
  - 不同的操作系统——UNIX、Windows等
  - 不同的软/硬件、接口和通信约定（协议）
  - 不同的应用环境——固定、移动等...
  - 不同种类业务——分时、交互、实时等
  - 宝贵的投资和积累——有形、无形等
  - 用户业务的延续性——不允许出现大的跌宕起伏



- 对于复杂的网络系统，用什么方法能合理地组织网络的结构，以达到：
  - 结构清晰
  - 简化设计与实现
  - 便于更新与维护
  - 较强的独立性和适应性
  - 解决：分而治之！ ——分层处理



# 层次化方法在其他领域的应用

---

- 程序设计
  - 把一个大的程序分解为若干个层次的小模块来实现，如操作系统。
- 邮政系统
  - 邮递员、邮政分局、邮政总局、邮政运输
- 银行系统
- 物流系统
- .....



## 3.1.2 分层原理

- 计算机网络中也采用了**分层**方法：

把复杂问题划分为若干较小的、单一的局部问题，在不同层上予以解决

- 网络的层次结构方法要解决的问题：

- 网络应该具有**哪些层次**？**每一层的功能**是什么？（**分层与功能**）
- **各层之间的关系**是怎样的？它们**如何进行交互**？（**服务与接口**）
- 通信双方的数据传输要**遵循哪些规则**？（**协议**）





## □ 计算机网络体系结构（概念）：

计算机网络中的分**层**结构与每层的功能、对等层进程间的通信**协议**和相邻层间**接口与服务**的集合

## □ 最早的网络体系结构源于**IBM**的**SNA**（system network architecture）

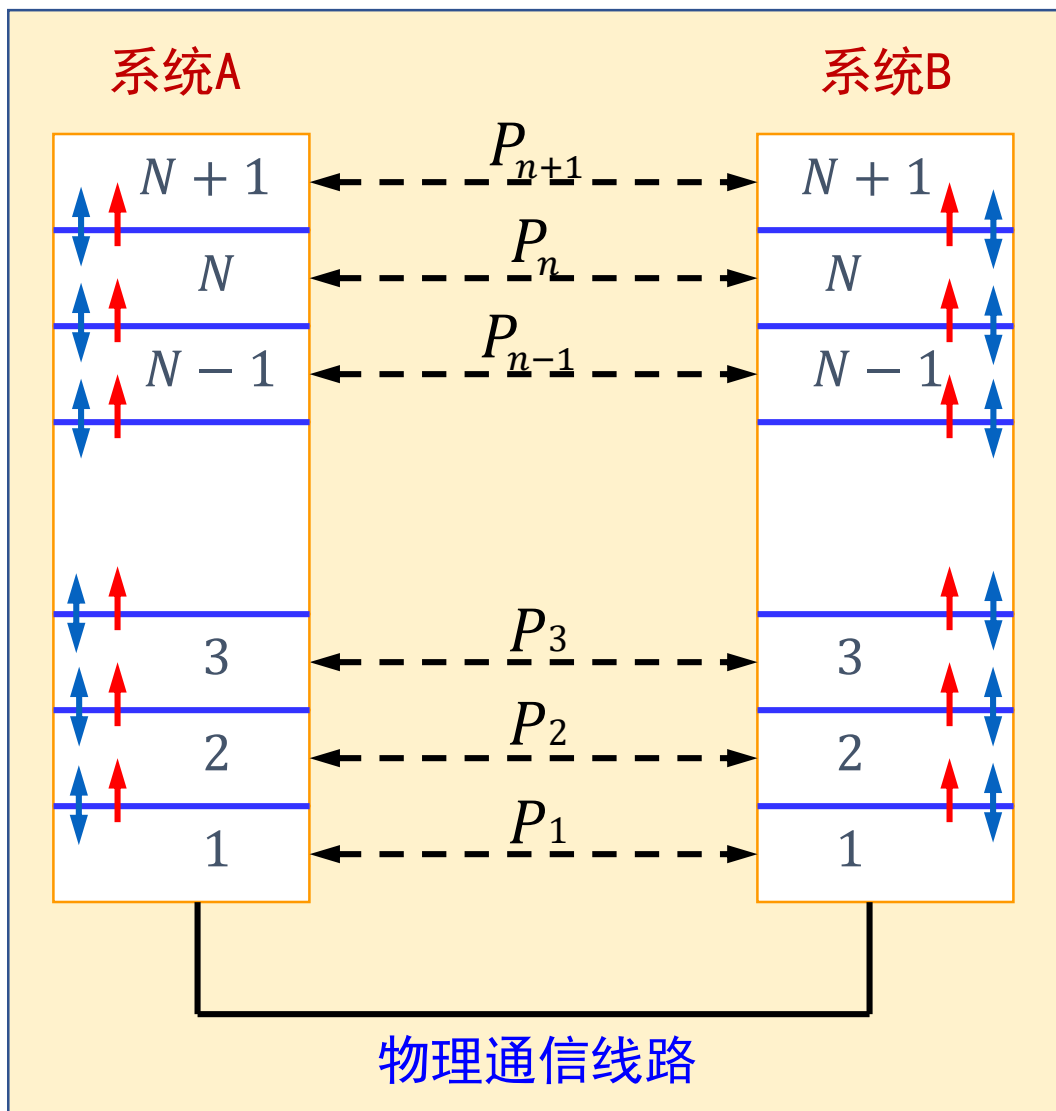
## □ 由国际标准化组织**ISO**制定的网络体系结构国际标准是**OSI/RM**

- **失败：P65**

## □ 实际中应用最广泛的是**TCP/IP**体系结构

- **事实上的标准**

## □ 网络分层体系结构



- 网络中的任何一个系统均按照层次结构来组织
- 同一网络中，任意两个端系统的层次必须相同
- 每层使用其下层的**服务**，并向其上层**提供服务**
- 通信只在**对等层**间进行，非对等层间不能“通信”
- 实际的物理通信只在最底层完成
- $P_n$ ：第 **$n$** 层**协议**，即第 **$n$** 层对等实体间通信时必须遵循的规则或约定



# 层次结构方法的优点

## □ 各层之间是独立的——耦合程度低

- 上层只需了解下层通过层间接口提供什么服务——黑箱方法

## □ 灵活性好

- 只要服务和接口不变，每层的实现方法可任意改变

## □ 易于实现和维护

- 把复杂的系统分解成若干个涉及范围小、功能简单的子单元：
  - 使系统的结构清晰，实现、调试和维护变得简单和容易
  - 使设计人员能专心设计和开发所关心的功能模块

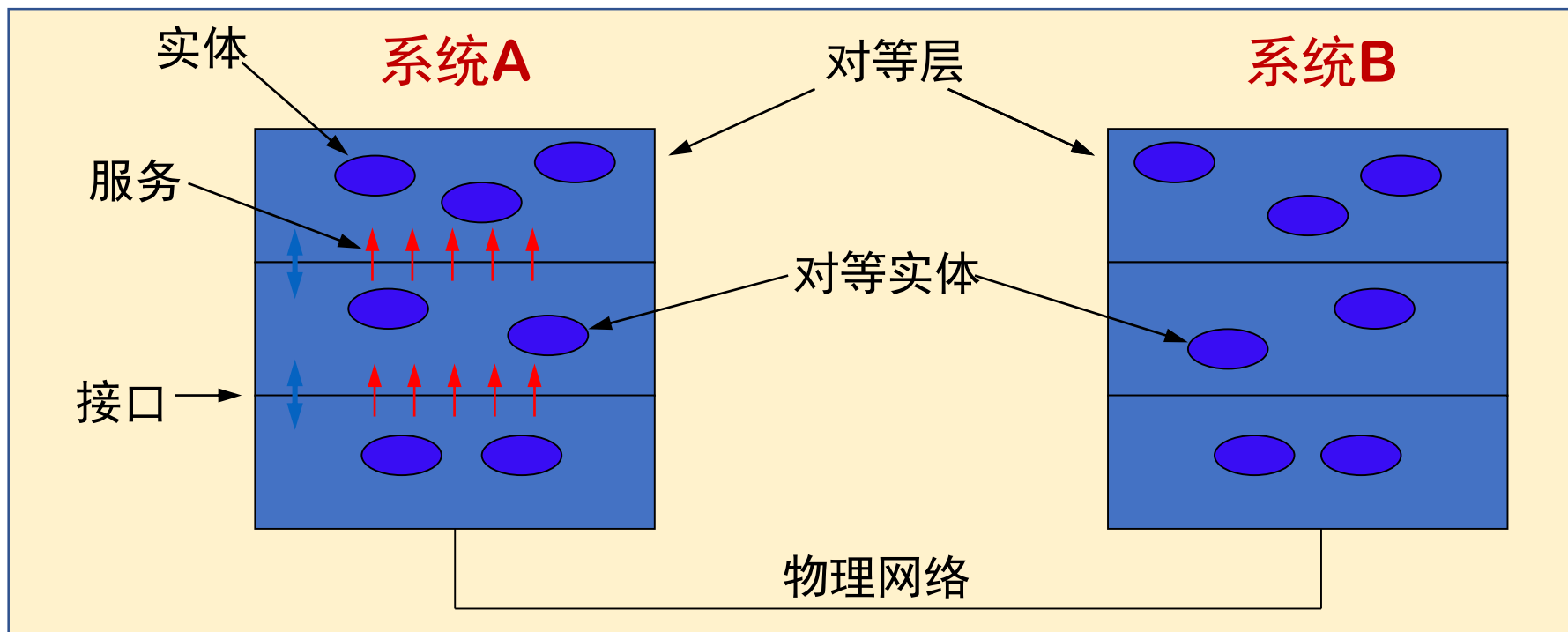
## □ 结构可分割

- 各层都可以采用最合适的技术来实现

## □ 促进标准化

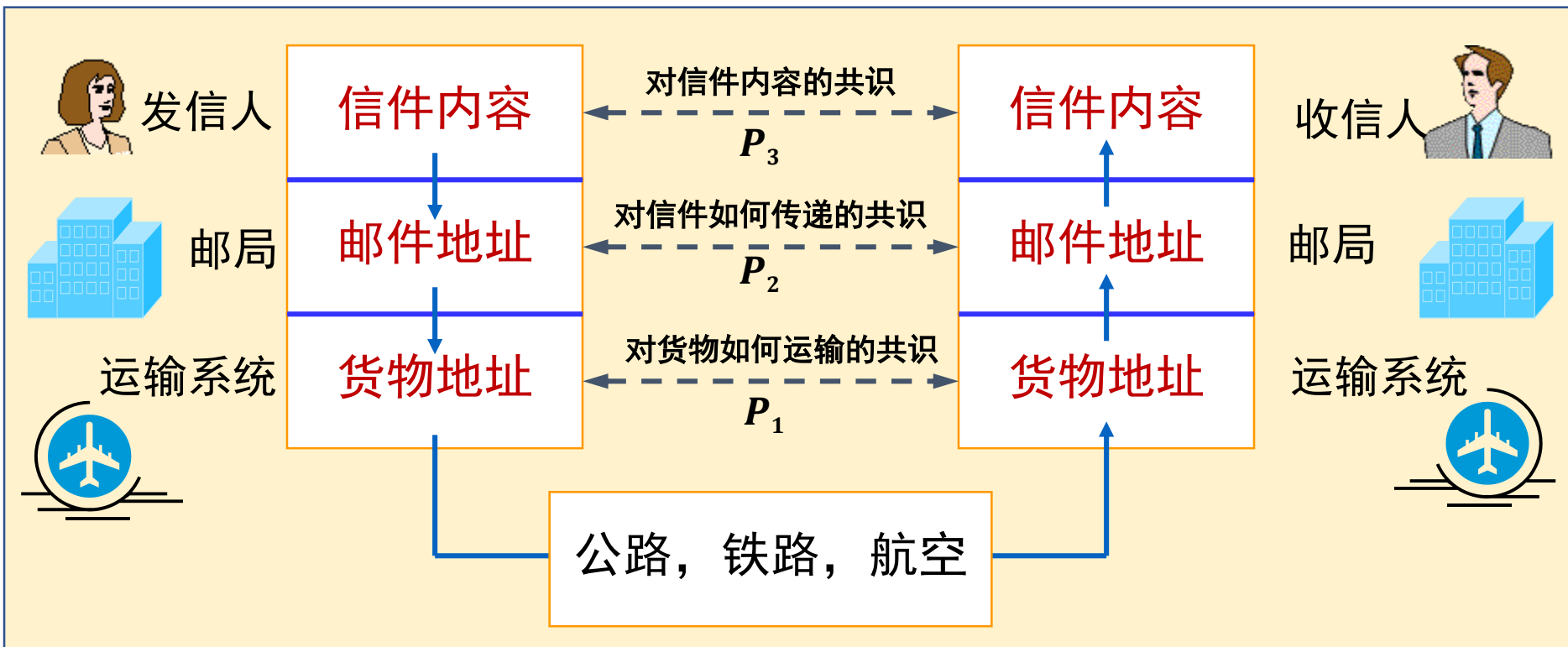
- 功能及服务精确说明，促进标准化

### 3.1.3 基本概念



- **实体**：任何可以发送或接收信息的硬件/软件进程
- **对等层**：两个不同系统的同级层次
- **对等实体**：不同主机中位于同一层次的实体
- **接口**：相邻层间交互的界面，定义相邻层间的操作、下层对上层的**服务**
- **服务**：某一层及其以下各层的一种能力，通过接口提供其相邻上层

# 对等通信例：两个人收发信件



## 想一想：

- 收信人与发信人之间、邮局之间，是在直接通信吗？
- 邮局、运输系统各向谁提供什么样的服务？
- 邮局、收发信人各使用谁提供的什么服务？



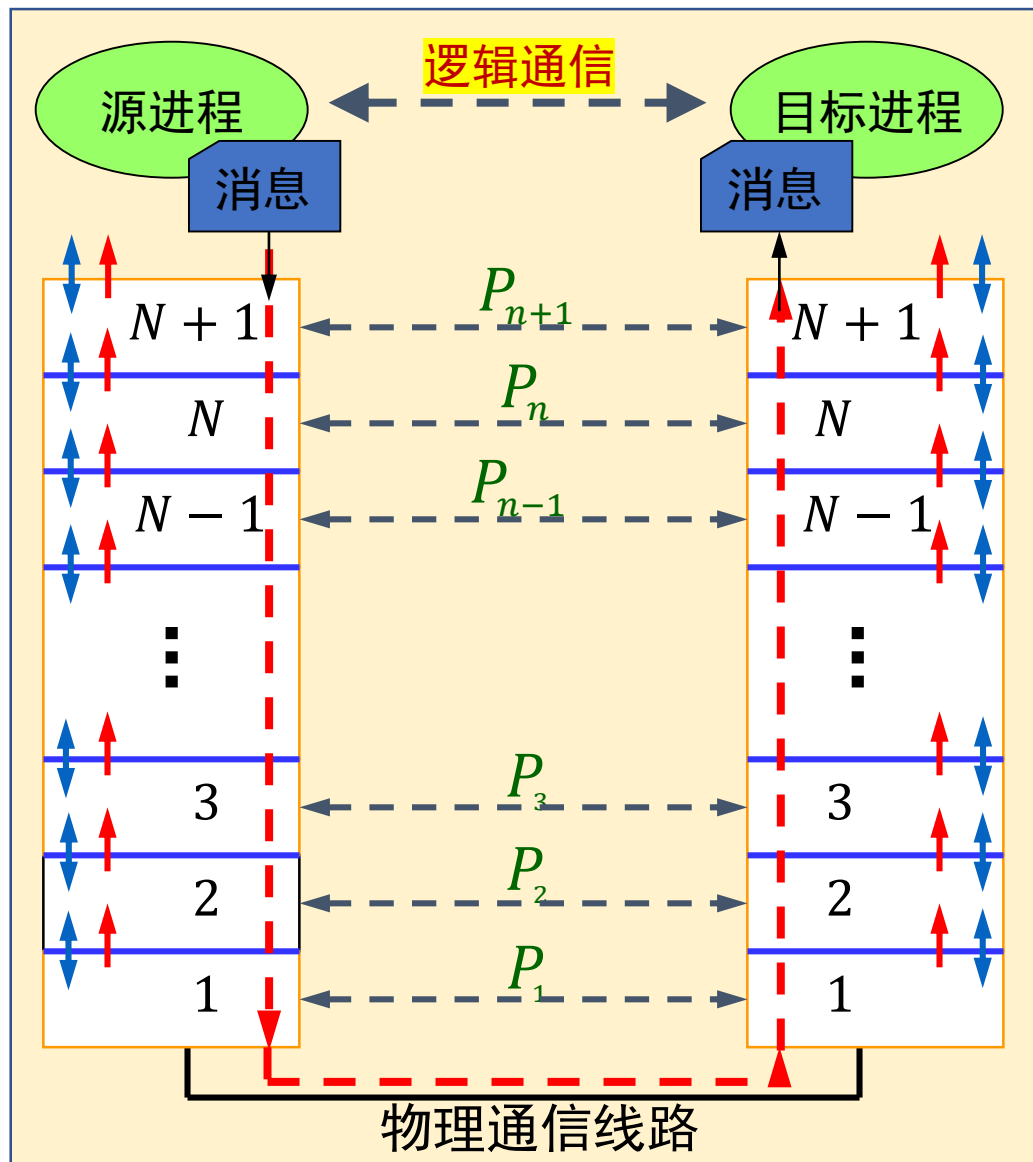
# 1. 对等层通信的实质

---

- 对等层实体之间实现的是**虚拟的逻辑通信**
- 下层向上层提供服务
- 上层依赖下层提供的服务来与其他主机上的对等层通信
- 实际通信在最底层完成

# 源进程传送消息到目标进程的过程

- 消息送到源系统的最高层
- 自上而下逐层封装
- 经物理线路传输到目标系统
- 目标系统收到信息
- 自下而上逐层处理并拆封
- 最高层将消息提交给目标进程





## 2. 通信协议

### ■ 通信协议

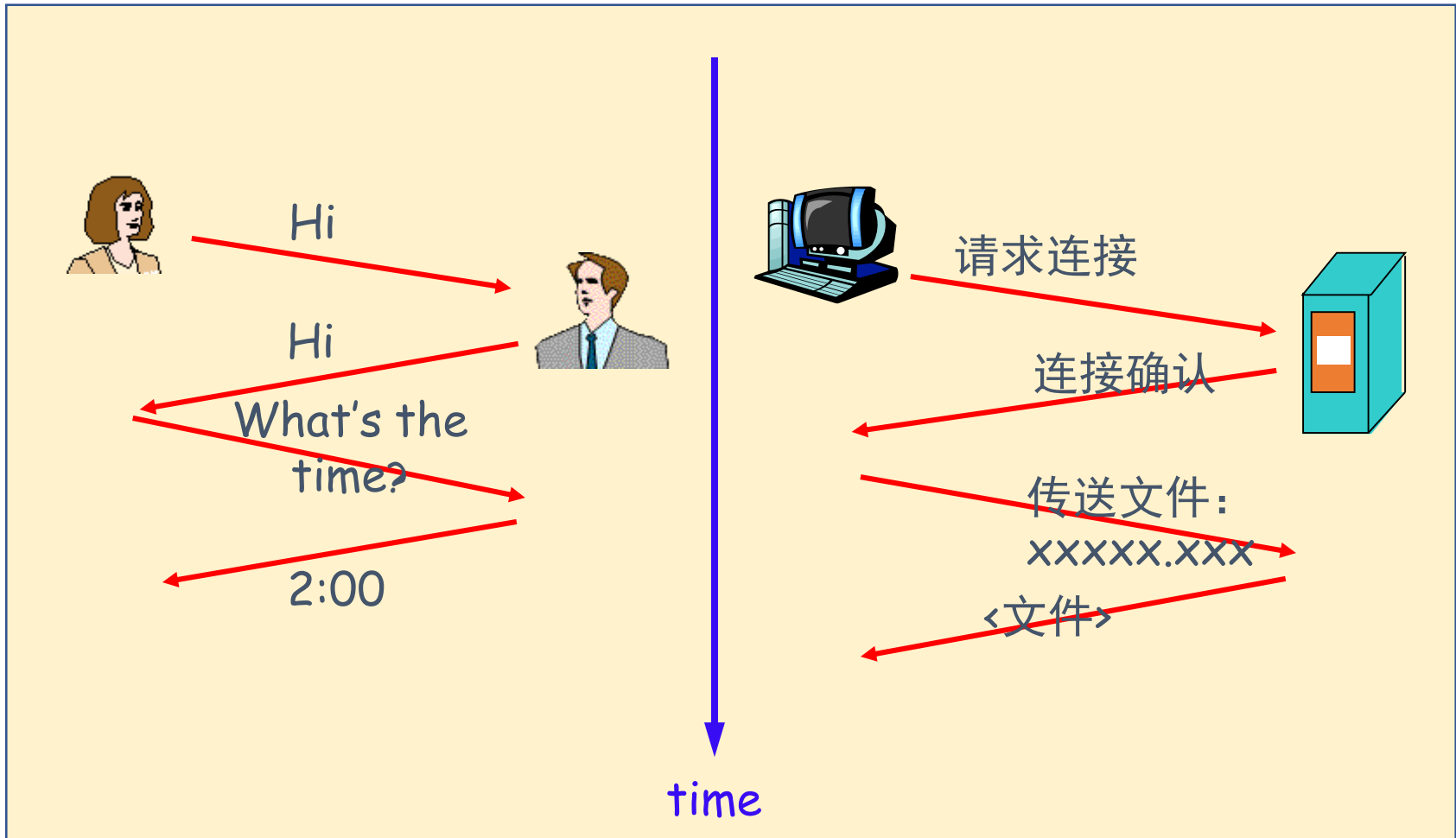
- 计算机之间
- 网络中所有的通信活动都是由协议控制
- 为网络执行数据交换而建立的规则、标准或约定

### ■ 协议内容

- 网络实体间发送和接收报文的格式、顺序以及发送和接收消息时应采取的行动（语法、同步和语义）



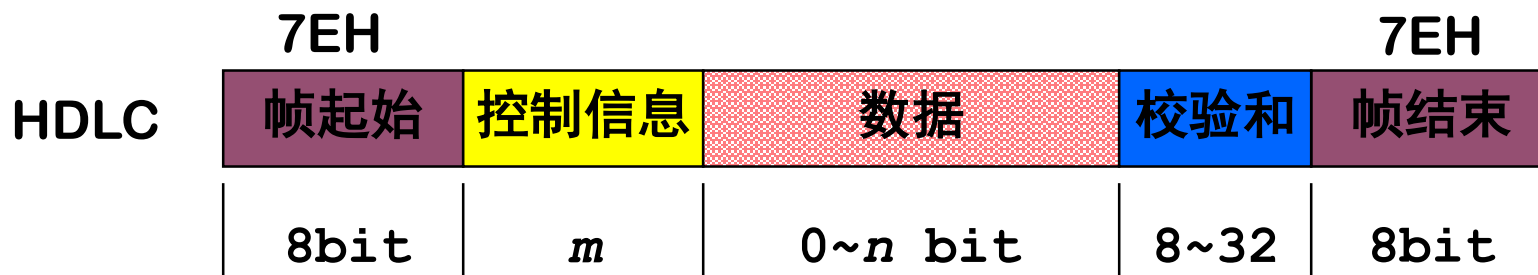
## 人相互交流的协议和通信协议之间的对比



# 通信协议的三要素：语法、语义、同步（时序）

## □ 语法

- 协议元素与数据的组合格式，以便通信双方正确识别  
如报文格式、帧格式



# 通信协议的三要素：语法、语义、同步（时序）

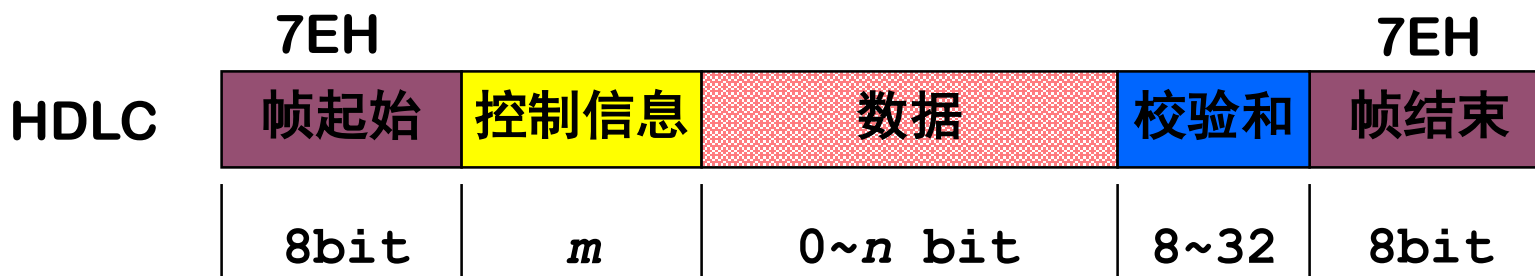
## □ 语义

高级数据链路控制（High-Level Data Link Control或简称HDLC）是一个在同步网上传输数据、面向比特的数据链路协议

• 对协议中各协议元素的含义的解释，例如：

➢ 在HDLC协议中，标志Flag(7EH)表示帧的开始和结束

➢ 在BSC协议中，SOH(01H)表示报文的开始，STX(02H)表示报文正文的开始，ETX(03H)表示报文正文的结束



二进制同步通信规程

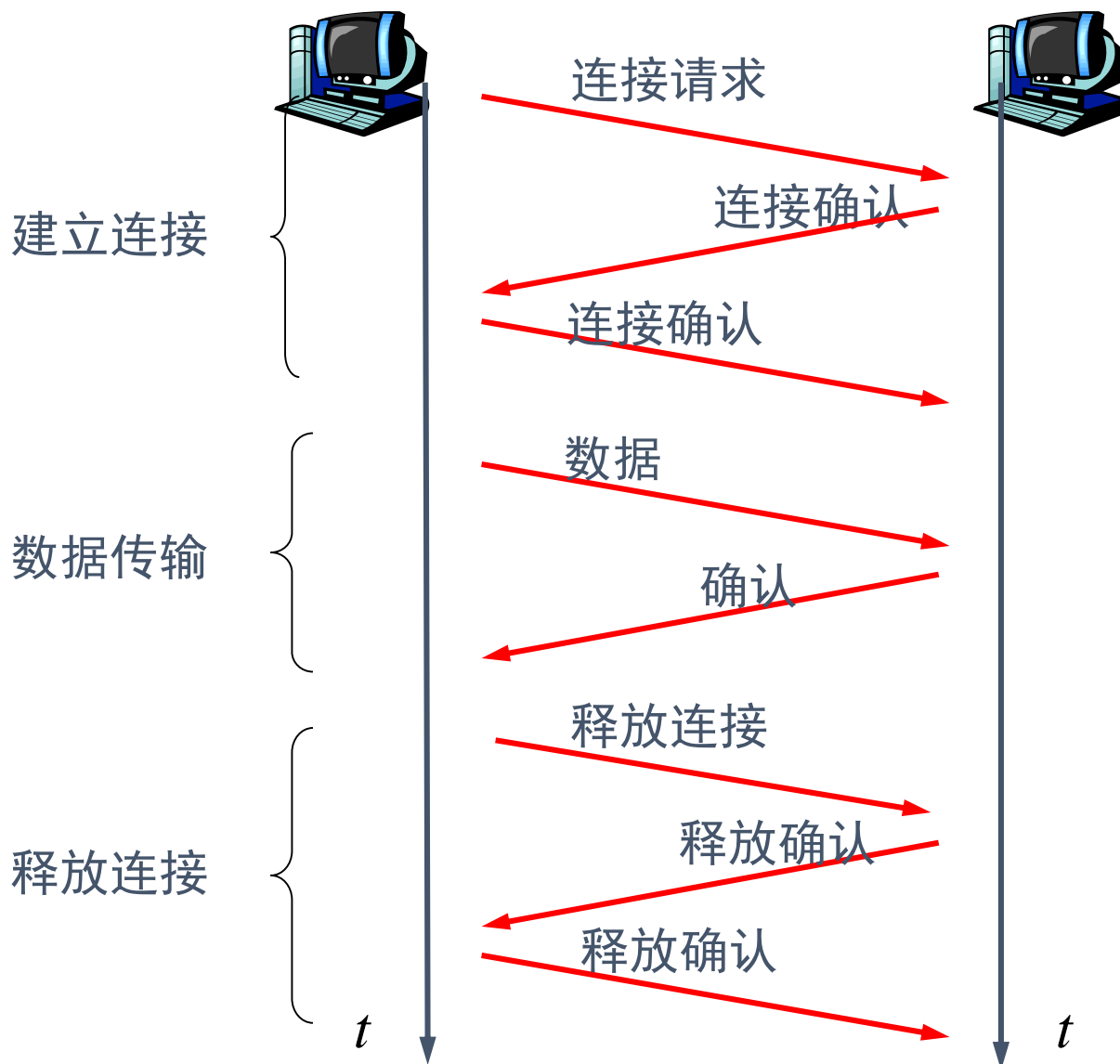
# 通信协议的三要素：语法、语义、同步（时序）

## 同步（时序）

- 通信过程中，通信双方操作的**执行顺序和规则**

## 协议栈：

协议也形成了层次结构，称为协议栈  
**P68**





- 网络体系结构中：
  - 每层可能会有若干个协议
  - 一个协议只属于一个层次
  
- 协议可以由软件或硬件来实现：
  - 网络通信协议软件、网络驱动程序
  - 网络硬件
  
- 常用协议组：
  - **TCP/IP** (Windows、UNIX、Linux、...)
  - **NetBEUI** (Windows)
  - **IPX/SPX** (NetWare、Windows)



## 3. 协议数据单元（PDU） P70

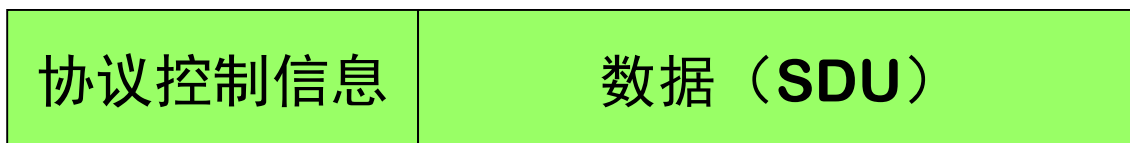
- 网络体系结构中，对等层之间交换的信息报文统称为**协议数据单元**（Protocol Data Unit, PDU）
- 传输层及以下各层的**PDU**另外还有各自**特定的名称**：
  - 传输层——**段**（Segment）
  - 网络层——**分组/包**（Packet）
  - 数据链路层——**帧**（Frame）
  - 物理层——**比特**（Bit）

### 3. 协议数据单元（PDU） P70

□ 传输层及以下各层的PDU另外还有各自特定的名称：

- 传输层——段（Segment）
- 网络层——分组/包（Packet）
- 数据链路层——帧（Frame）
- 物理层——比特（Bit）

□ PDU由协议控制信息（协议头）和数据（SDU）组成：

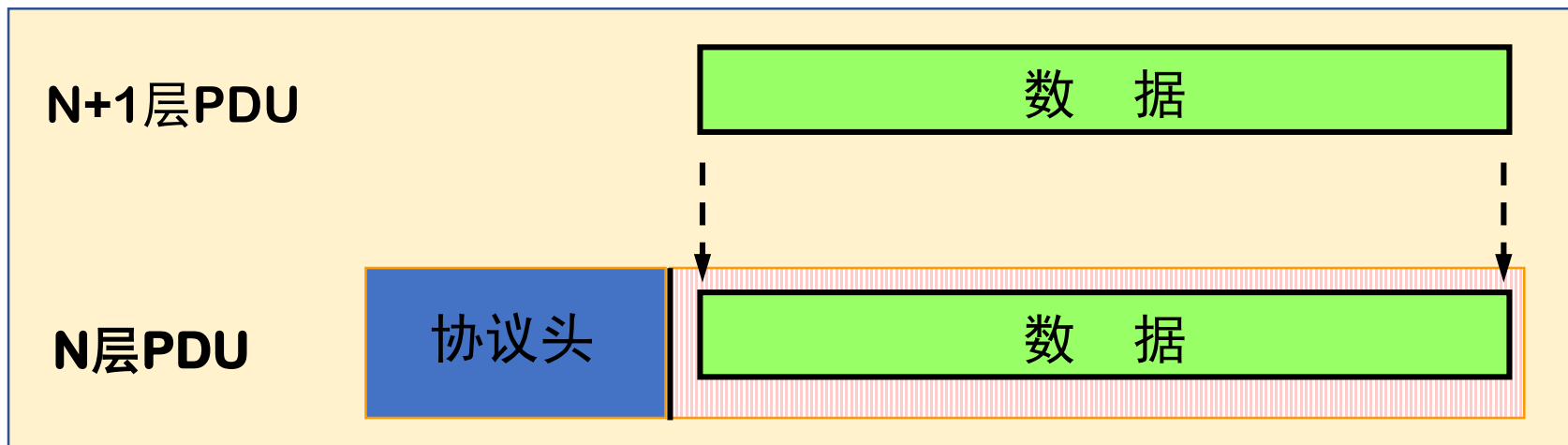


□ 协议头部中含有完成数据传输所需的控制信息：

- 地址、序号、长度、分段标志、差错控制信息……

□ 下层把上层的PDU作为本层的数据加以封装，然后加入本层的协议头部（和尾部）形成本层的PDU

- 封装：在数据前面加上特定的协议头部



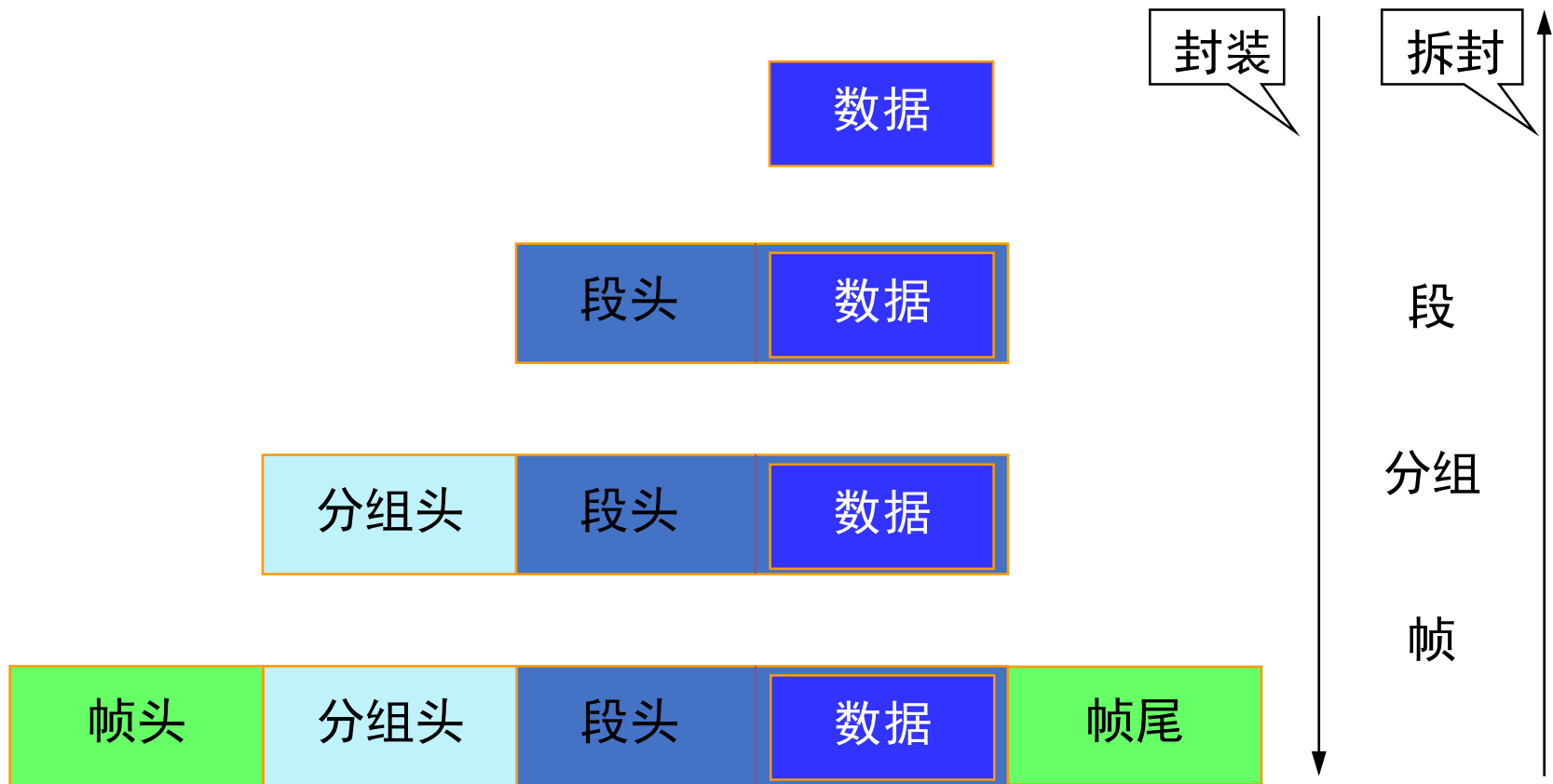
□ 因此，数据在源站自上而下递交的过程实际上就是不断封装的过程。到达目的地后自下而上递交的过程就是不断拆封的过程。——类比：发送信件

- 数据在传输时，其外面实际上要被包封多层“信封”



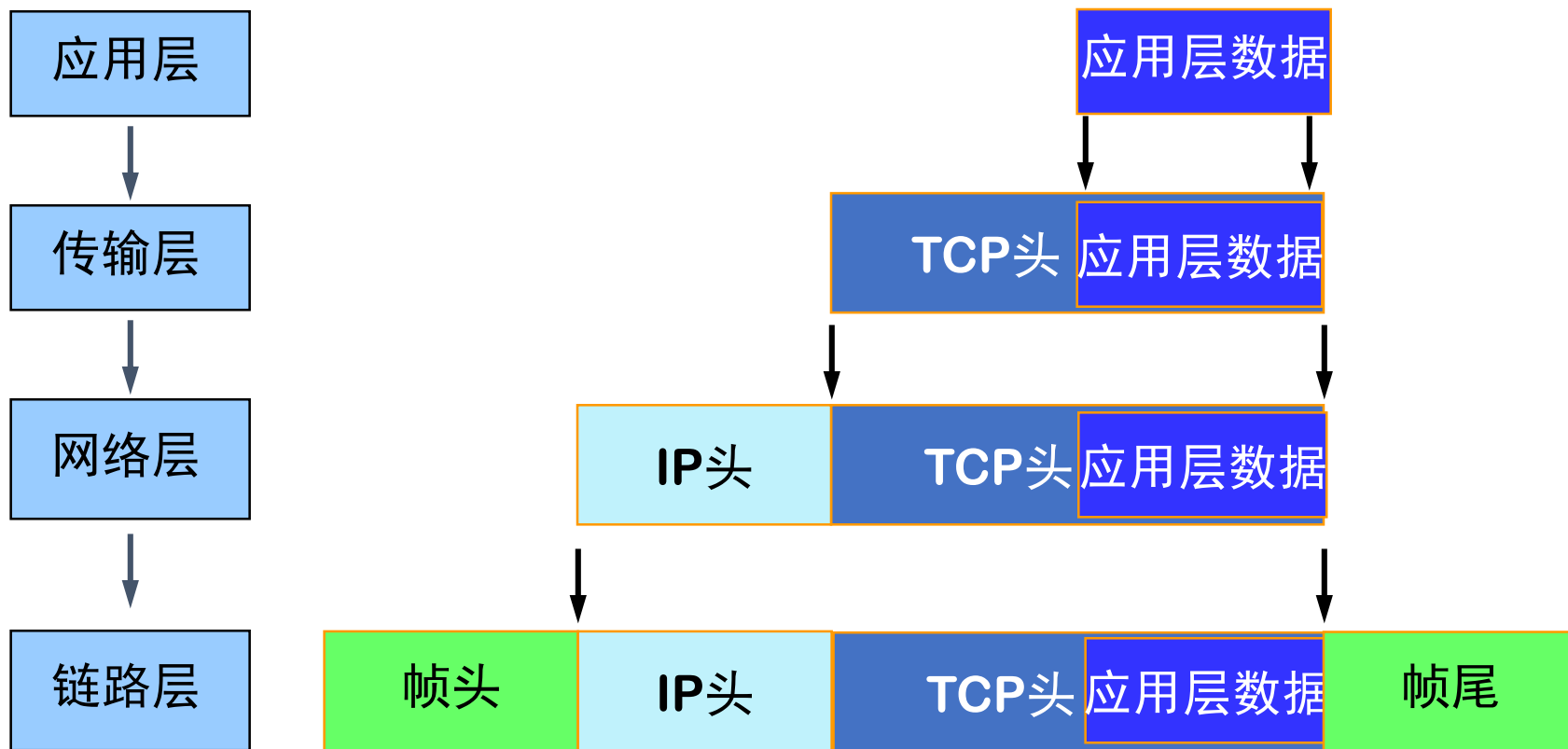


# 数据多层封装





## ■ 例：TCP/IP协议的封装过程





- 在目的站，某一层只能识别由源站对等层封装的“信封”，而对于被封装在“信封”内部的“数据”仅仅是拆封后将其提交给上层，本层不作任何处理。
  - 每一层只处理本层的协议头部！



## 4. 服务的特点

---

- 协议是**水平的**，服务是**垂直的**
- 只有能被高层**看得见**的功能才叫服务
- 相邻两层的实体通过**服务访问点**进行交互
- 服务访问点只是**逻辑接口**，与硬件接口不同



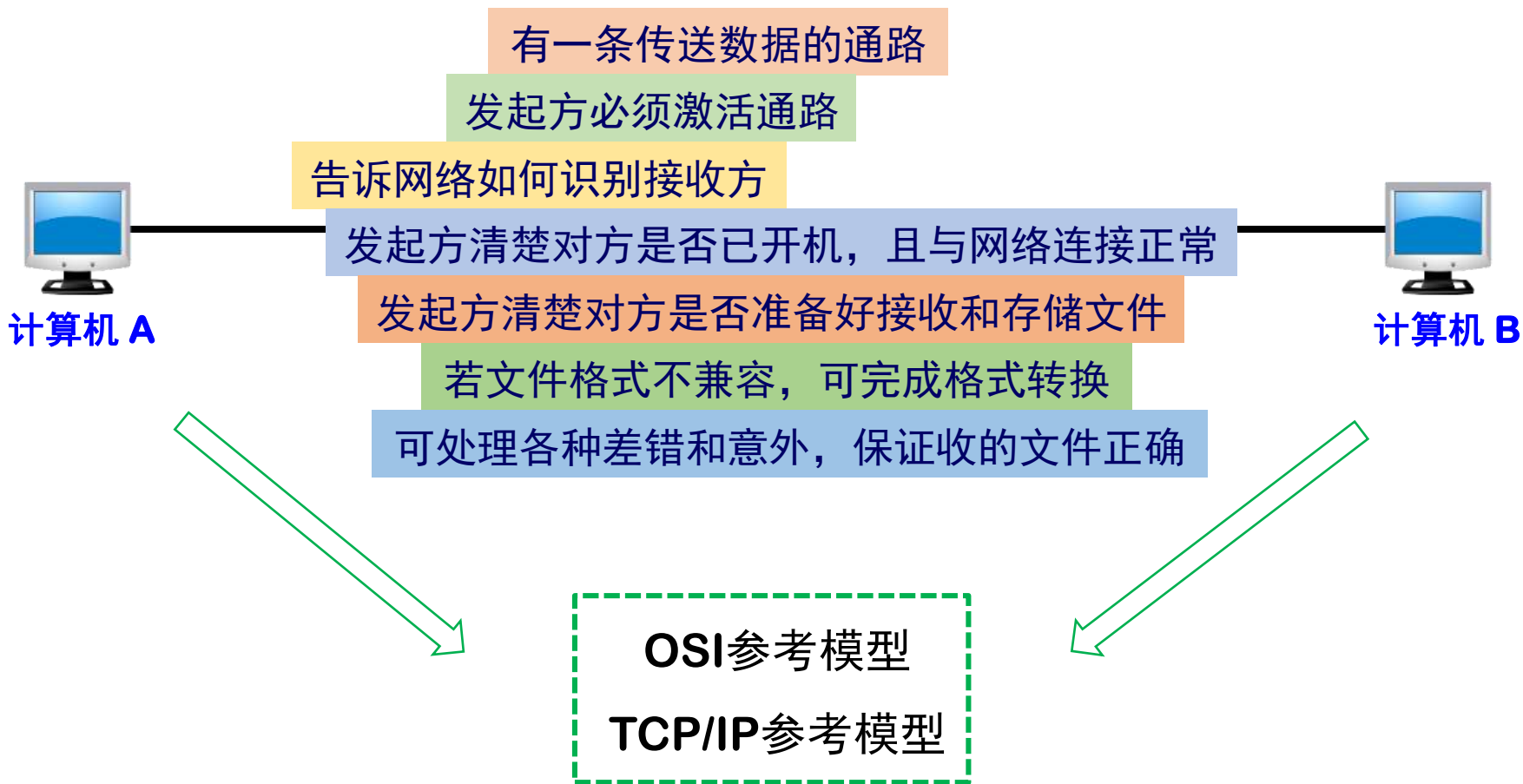
## \*服务原语 P72

- **服务原语 (Service Primitive)** 专供用户和其他实体访问服务。
- 服务原语只是对服务进行概念性的功能描述，至于如何实现并不作明确规定。

原语类型	含义
请求 ( <b>Request</b> )	( <b>N+1</b> ) 实体请求 ( <b>N</b> ) 实体提供服务
指示 ( <b>Indication</b> )	( <b>N</b> ) 实体通知 ( <b>N+1</b> ) 实体发生了某一事件
响应 ( <b>Response</b> )	( <b>N+1</b> ) 实体对 ( <b>N</b> ) 实体指示的响应
确认 ( <b>Confirm</b> )	( <b>N</b> ) 实体向 ( <b>N+1</b> ) 实体确认，( <b>N+1</b> ) 实体请求的服务已完成

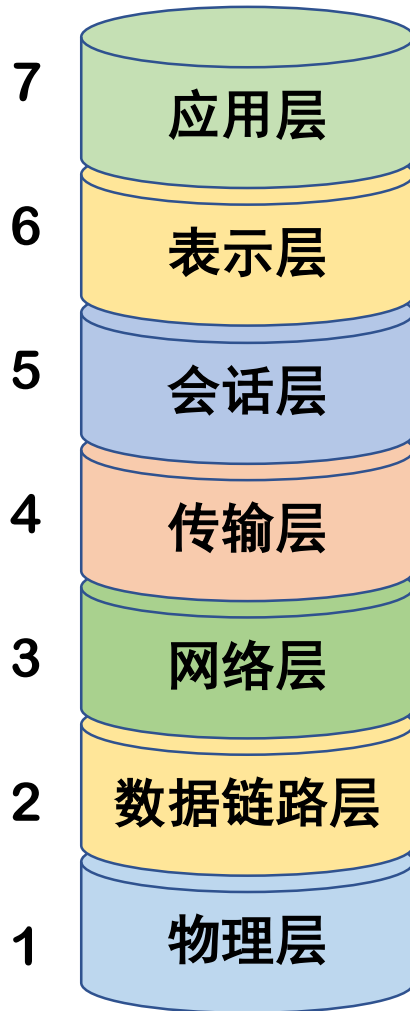
## 3.2 网络参考模型

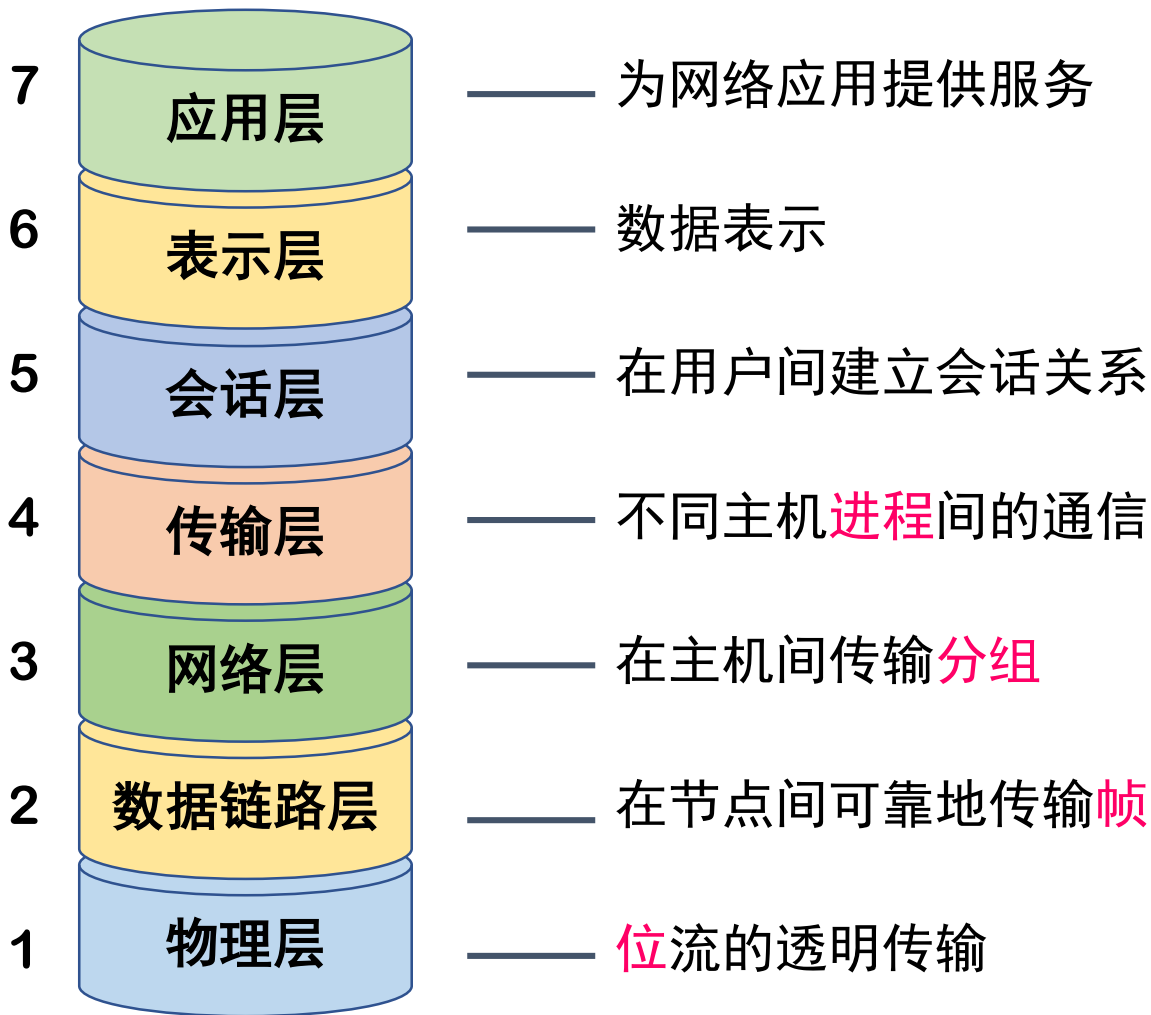
建立分层结构的网络参考模型，解决计算机间传送文件时的问题





## 3.2.1 OSI/RM

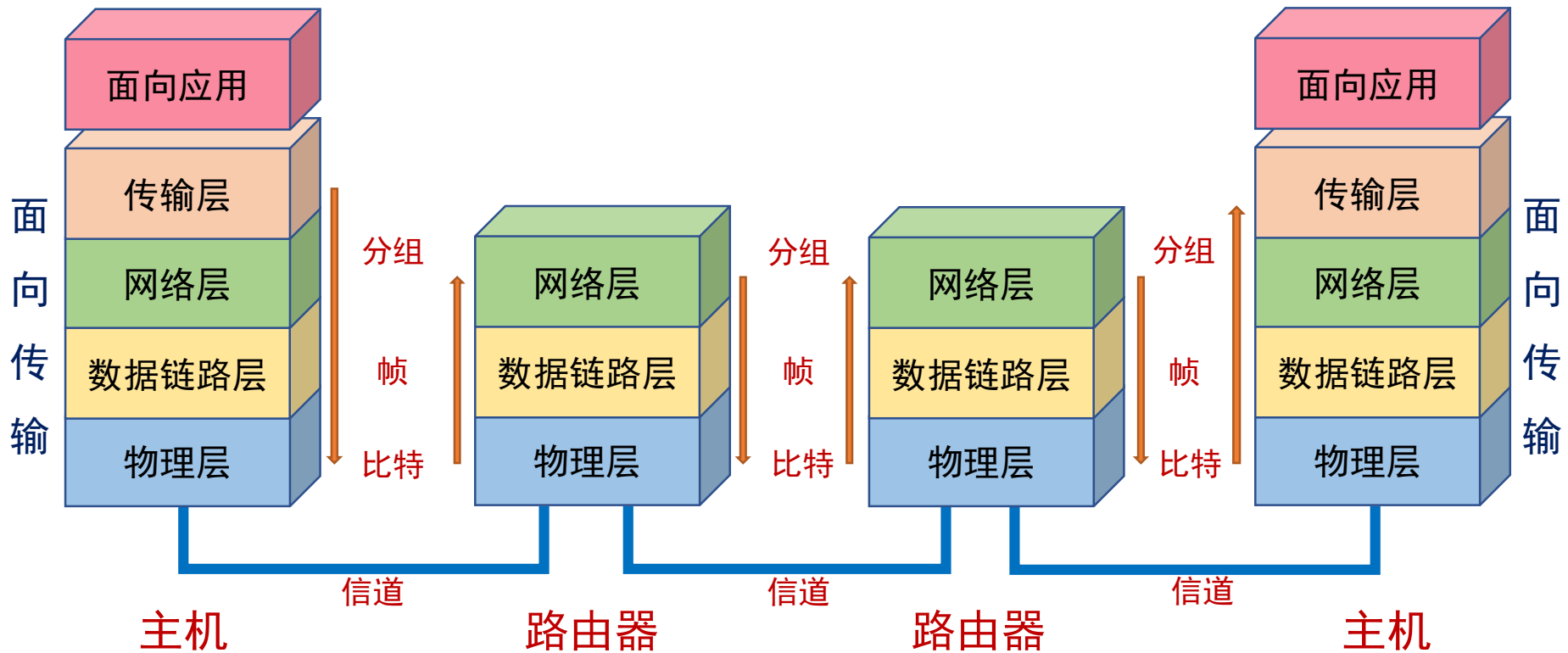




**OSI**参考模型只是指明了每一层应该实现哪些功能，并未定义每层用到的服务和协议。**ISO**为每层定义的标准是独立的，不属于**OSI**本身。



# 3.2.1 OSI/RM





## 3.2.1 OSI/RM: 物理层 (第1层) P75

- **功能**: 在物理媒体(介质)上正确地、透明地传送比特流, 即物理连接和差错检测, 不是物理介质本身。
- **PDU**: 位 (Bit)
- **目的**: 尽可能屏蔽掉不同传输媒体和通信手段的差异
- **服务**: 串行传输方式和并行传输方式
- **协议 (标准)**: 规定了物理接口的各种特性
  - **机械**: 物理连接器的尺寸、形状、规格
  - **电气**: 信号电平, 信号的脉冲宽度和频率, 数据传送速率, 最大传输距离等
  - **功能**: 接口引(线)脚的功能和作用
  - **规程**: 信号时序, 应答关系, 操作过程
  - **例**: RS-232、RS-449、V.24、V.35、G.703/G.704



## □ RS-232-C

- 外形为**25针或9针的D形**连接器
- 通信速率 (b/s) :  
100、300、600、1200、2400、4800、9600、19.2K、33.6K、56K
- 信号电平：
  - 逻辑“1”：-3V~-15V
  - 逻辑“0”：+3V~+15V
- 引脚定义.....



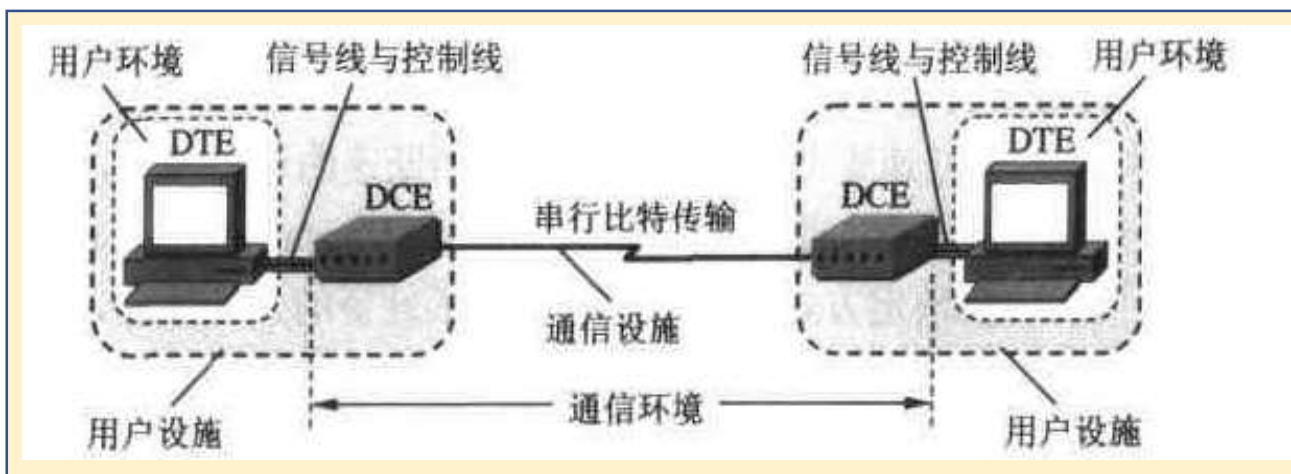
# DTE和DCE

## ■ DTE（Data Terminal Equipment）数据终端设备

- 具有一定的数据处理能力以及收发能力的数据输入/输出设备、终端设备或计算机等终端装置

## ■ DCE（Data Communication Equipment）数据通信设备

- 自动呼叫应答设备、交换机以及其他一些中间装置的集合,其作用就是在DTE和传输线路之间提供信号变换和编码的功能，并且负责建立、保持和释放数据链路连接



**DTE和DCE的接口标准是典型的物理层协议**



## 3.2.1 OSI/RM：数据链路层（第2层）

- **功能：**加强物理层传输原始比特的功能，即在两个相邻节点间可靠地传输数据，使之对网络层呈现为一条无错的链路。
- **PDU：帧（Frame）**
- **具体功能：**
  - **链路管理：**建立与拆除数据链路连接；
  - **成帧：**帧封装，按顺序传送，处理返回的确认帧，帧同步；
  - **流量控制：**抑制发送方的传输速率，使接收方来得及接收；
  - **差错控制：**错误检测/恢复，可靠的传输，**CRC，ARQ**；
  - **物理寻址：****MAC地址/物理地址**。



# 数据链路层的服务

## ■ 无确认无连接服务

源主机的数据链路层可以在**任何时候**发送**任意长度**的信息

## ■ 有确认无连接服务

源主机的数据链路层要发送数据时，就直接发送数据帧，要有确认帧

## ■ 面向连接的服务

数据传送分为数据链路**建立**、**数据帧传送**和数据链路的**拆除**

## ■ 根据确认的发送条件，将确认机制分为三种方式：

- **正向确认**（收到的帧若正确就返回正确，广泛使用）
- **双向确认**（若正确就返回正确；若错误就返回错误）
- **负向确认**（收到的帧若错误就返回错误）



# 数据链路层的协议

## ■ 面向字符的通信规程

- 数据以**字符**为单位传输，用控制字符控制通信

## ■ 面向比特的通信规程

- 数据以**位**为单位传输，用帧中的控制字段控制通信





## 3.2.1 OSI/RM：网络层（第3层）

- 寻址并选择合适的路由，把数据报从源端传送到目的端，在需要时对上层的数据进行分段和重组。
- PDU：数据报（Datagram，IP-分组Packet）
- 只存在于端主机和路由器中，局域网交换机中没有；

### 1. 功能：

- 路由选择（网络中的节点根据通信网络的情况，按照一定的策略，为数据报选择一条可用的传输路由，将其发往目的主机）
- 拥塞控制（一个子网中的分组间避免相互妨碍）
- 网络互连（负责解决不同种类的网络相互连接等问题）

### 2. IP协议[RFC 791]





# 网络层服务（回顾第2章分组交换技术）

## ■ 数据报（Datagram）服务

- 网络层从运输层接受报文（发送时）并拆分为报文分组，把每一个分组作为一个独立的信息单位传送
- **特征：**不需建立连接、每个数据报都附有网络地址、要求路由选择、数据报不能保证按序到达目的地、对故障的适应性强、易于平衡网络流量

## ■ 虚电路（Virtual Circuit）服务

- 在源主机要与目的主机通信之前，应先建立一条网络连接
- **特征：**要求先建立连接、全网地址、路由选择、按序到达、可靠性较高、适用于交互式作用
- 分类：永久虚电路、呼叫虚电路



# 数据报与虚电路（回顾第2章分组交换技术）

项目	虚电路	数据报
目的主机地址	建立连接时需要	每个分组都需要
初始化设置	需要	不需要
分组顺序	由通信子网负责保证	通信子网不负责
差错控制	由通信子网负责，对主机透明	由主机负责
流量控制	通信子网提供	网络层不提供
连接的建立和释放	需要	不需要



# 路由选择

- 如何在多条通信路径中找一条**最佳路径**？

依据：**速度**、**距离(步跳数)**、**价格**、**拥塞程度**

- **定义**：

**路由选择**又称**路径控制**，是指网络中的节点**根据通信网络的情况**，**按照一定的策略**，为数据报**选择**一条可用的**传输路由**，将其发往目的主机

- **路由器——路由表建立与维护**

**静态**：人工设置，只适用于小型网络

**动态**：运行过程中根据网络情况自动地动态维护

- **路由算法——建立与维护路由表的方法**

距离向量算法：**RIP**、**CGP**等

链路状态算法：**OSPF**等



## 3.2.1 OSI/RM：传输层（第4层）

- **概述**：对网络层的连接进行管理，在源端与目的端之间提供可靠的、透明的数据传输，使上层服务用户不必关心通信子网的实现细节。
- **传输层的特点**：
  - 以上各层：面向应用，本层及以下各层：面向传输；
  - 与网络层的部分服务有重叠交叉，功能取舍取决于网络层功能的强弱；
  - 只存在于端主机中，路由器中没有；
  - 实现源主机到目的主机“端到端”的连接

在这一点上与网络层的区别是什么？

网络层：为主机之间提供逻辑传输

传输层：为应用进程之间提供逻辑传输



## ■ 传输层功能

### • 传输连接管理：

传输层是一个真正的端到端的层，所有的处理都是按照从源端到目标端来进行的。传输层一般使用端口与上层进行通信，端口作为通信进程的唯一标识，参与通信的双方进程，在通信开始前必须知道对方的端口号

### • 流量控制：

解决收发方处理能力不匹配的问题，防止数据传输过载

## ■ 传输层服务

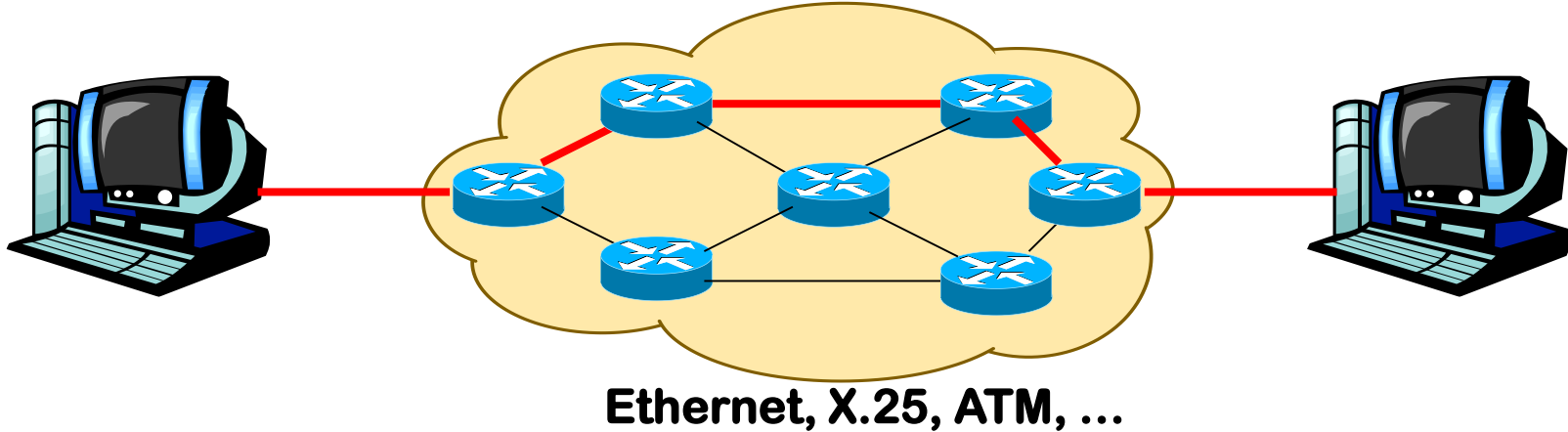
• 提供“面向连接”的可靠服务和“无连接”不可靠服务两种服务

## ■ 传输层协议

• 在因特网的传输层中的协议有面向连接的TCP和无连接的UDP  
具体见第5章

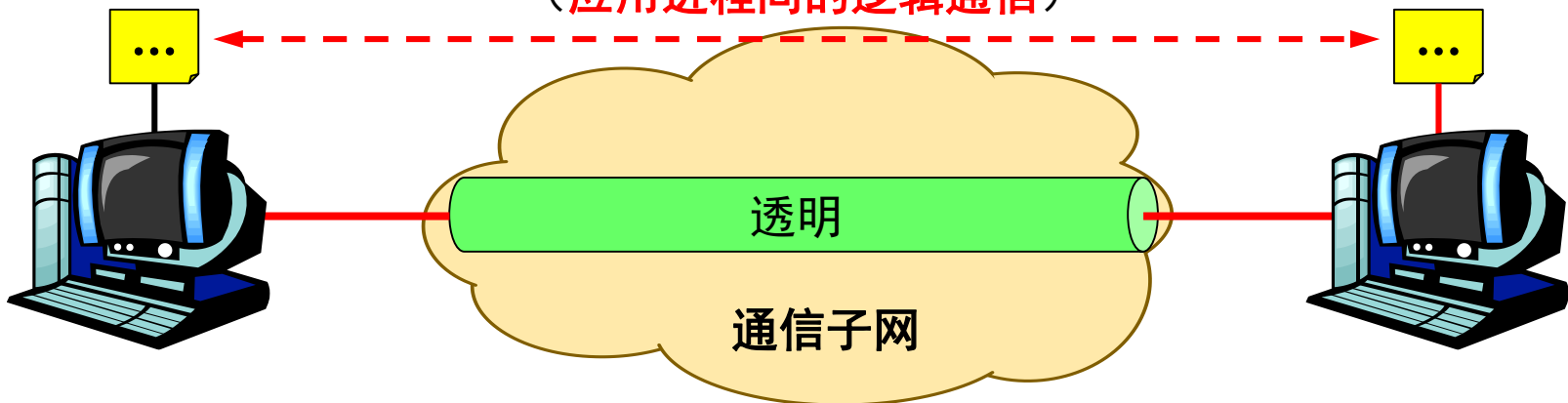
## 网络层的任务

沿两端点间的最佳路由传输数据  
(主机间的逻辑通信)



## 传输层的任务

两端点间可靠的透明数据传输  
(应用进程间的逻辑通信)





## 3.2.1 OSI/RM：会话层（第5层）

- 在传输层服务的基础上增加**控制会话(Session)**的机制，**建立、组织和协调**应用进程之间的交互过程
- **功能**：为用户建立、引导和释放会话连接
- **服务**
  - 服务类型：双向同时(双工)、双向交替(半双工)和单向(单工)
  - 会话层服务是同步的
- **协议**
  - 在**OSI/RM**中没有给出具体协议



## 3.2.1 OSI/RM：表示层（第6层）

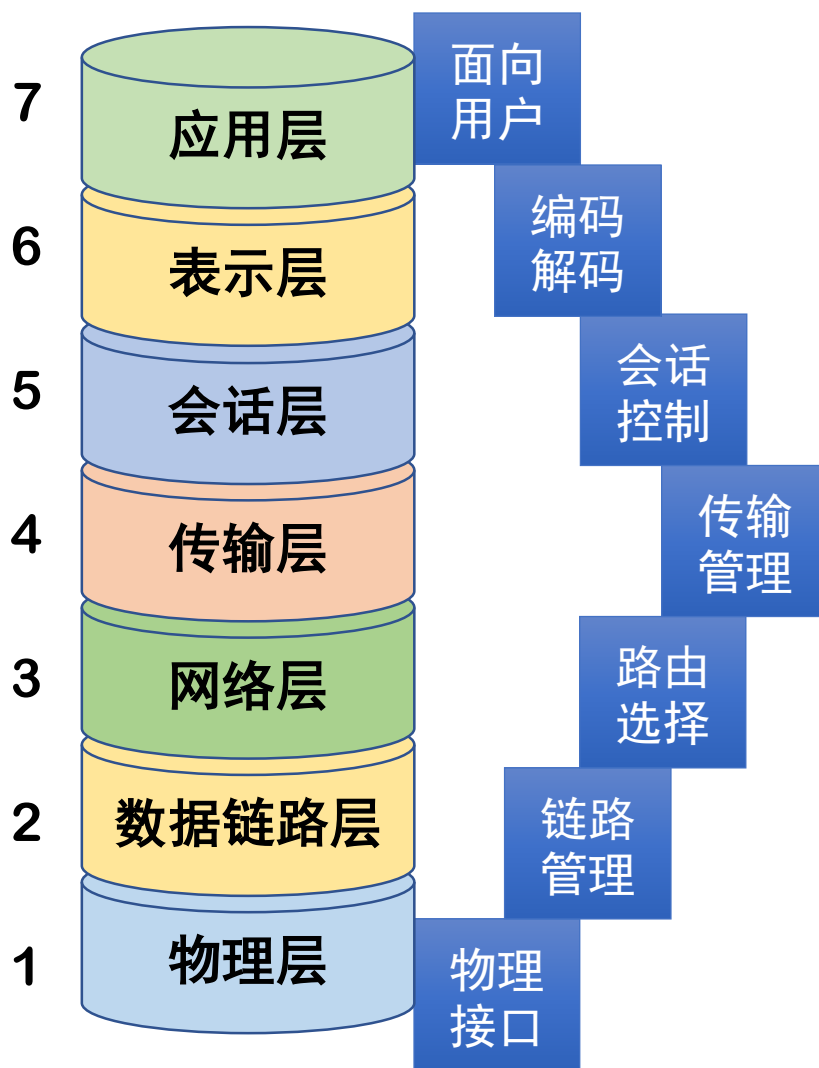
- 定义用户或应用程序之间**交换数据的格式**，**提供数据表示之间的转换服务**，**保证传输的信息到达目的端后意义不变**
- **功能**
  - 对源站内部的数据结构编码，**形成**适合于传输的**比特流**(符合“传输语法”)，到了目的站**再**进行**解码**，**转换成**目的站用户**所要求的格式**(符合目的站“局部语法”)，**保持传输数据的意义不变**
- **服务**
  - 不同类型计算机中内部格式的转换、密码转换和文本压缩的转换
- **协议**：**ASN.1和X.25 PAD协议等**





## 3.2.1 OSI/RM：应用层（第7层）

- 为End-user的应用进程提供标准的网络服务和应用接口
- 功能：
  - 提供各种不同的应用协议以满足应用进程的需求
  - 为用户进程提供应用接口
  - 识别并证实目的通信方的可用性
  - 使协同工作的应用进程之间进行同步
  - 为通信过程申请资源
- 应用层协议的例子：具体见第4章
  - **OSI**：VTP、MHS、FTAM、DS...
  - **TCP/IP**：Telnet、SMTP、FTP、DNS、HTTP...





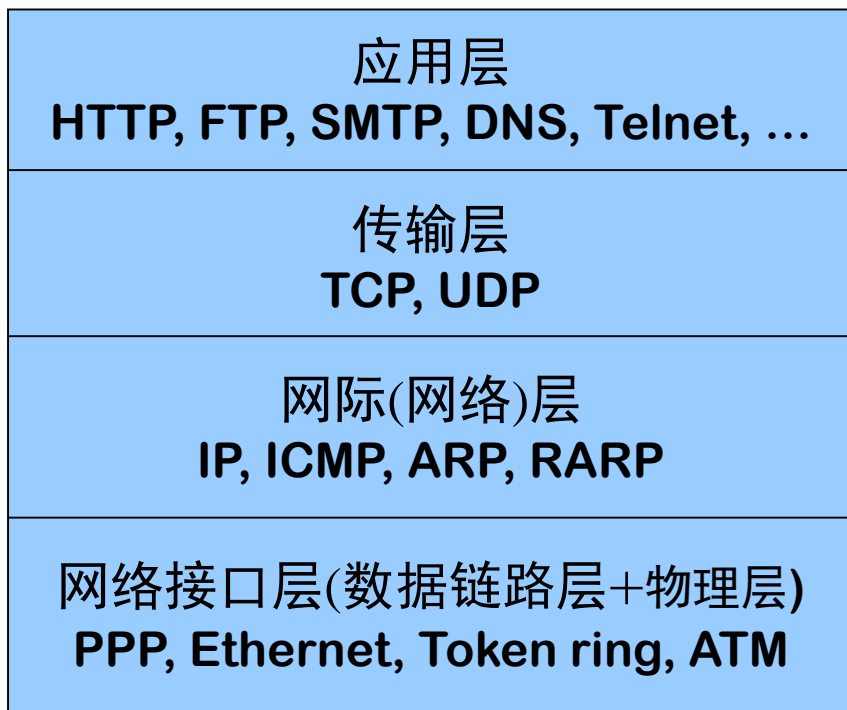
## 3.2.2 TCP/IP参考模型

- **TCP/IP**不是一个单独的协议，而是由数十个具有层次结构的协议组成的一个**协议集**。
  - **TCP**和**IP**是该协议集中的两个最重要的**核心协议**
- **TCP/IP是Internet上实际使用的标准通信协议集**
- **TCP/IP**标准以“请求注释”（**RFC**）文档发布：
  - **TCP** [RFC 768], **UDP** [RFC793]
  - **IP** [RFC 791]
  - **DNS** [RFC 1034, 1035], **FTP** [RFC 959, 1635]

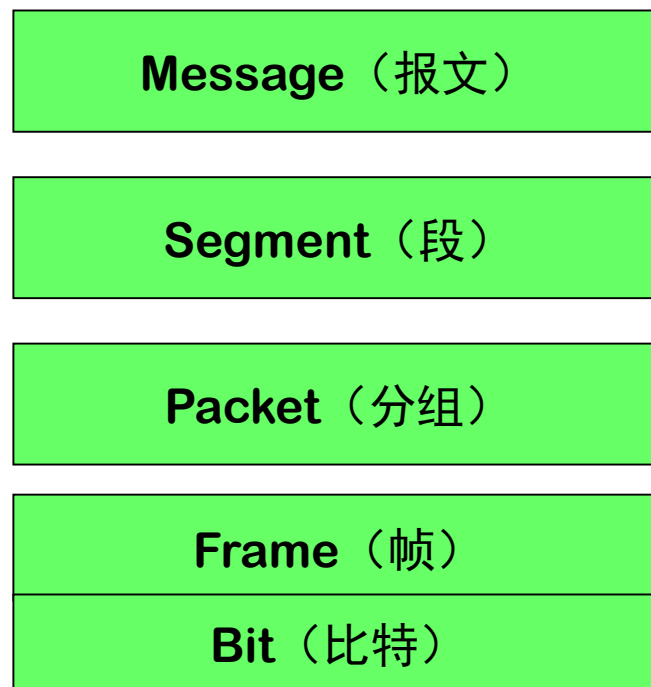


# TCP/IP的体系结构层次

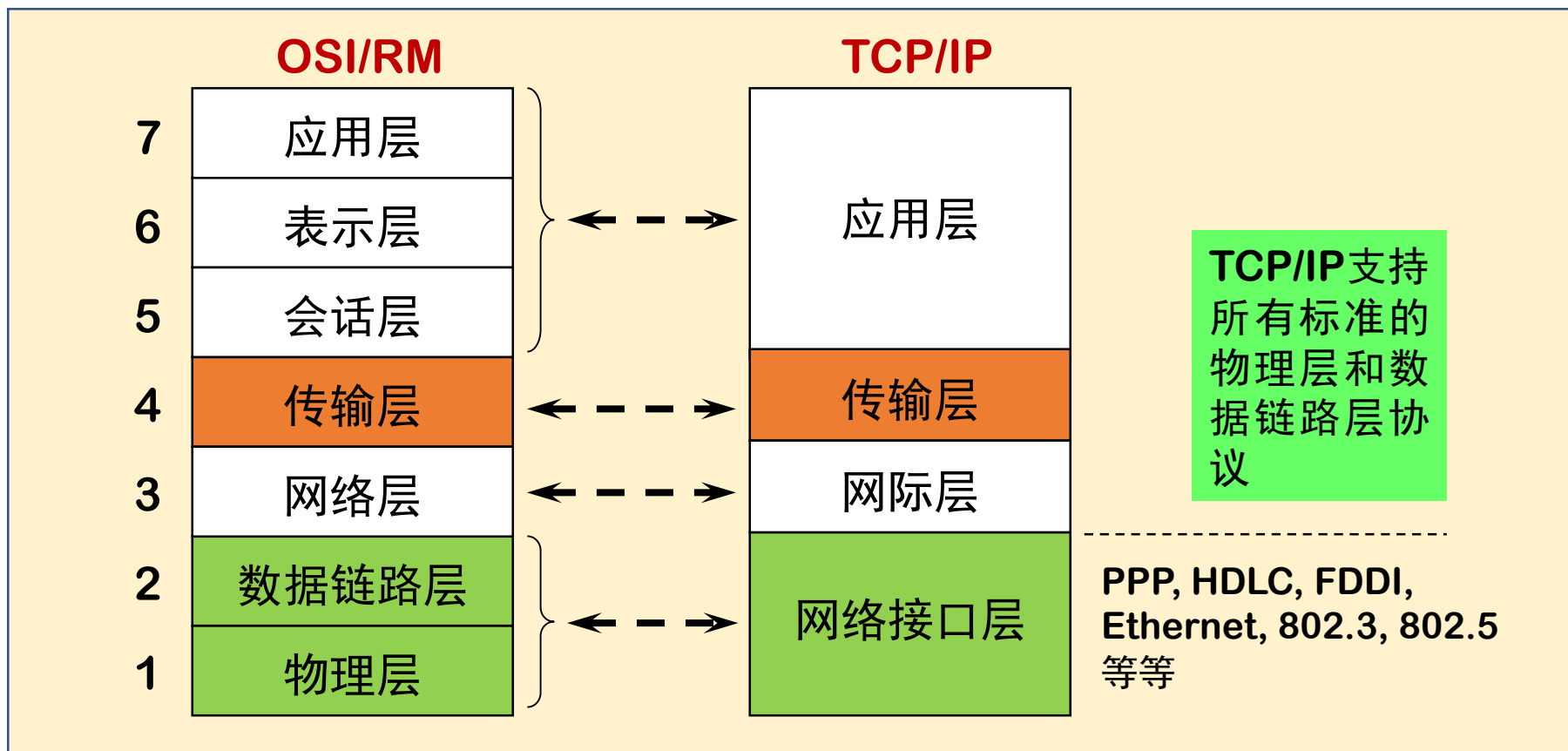
## TCP/IP协议



## PDU



# TCP/IP与OSI/RM的对应关系



- **OSI/RM和TCP/IP相结合的5层结构——原理体系结构：**
  - **应用层、传输层、网络层、数据链路层和物理层**



## 3.2.2 TCP/IP：网络接口层（第1层）

### ■ 功能

- 负责将网络层的IP数据报通过物理网络发送，或从物理网络接收数据帧，抽取出IP数据报上交给网际层。网络接口层使得上层的操作和底层的物理网络无关。

### ■ 协议

- 没有定义任何实际协议，仅定义了网络接口
- 任何已有的数据链路层协议和物理层协议都可以用来支持TCP/IP

### ■ 优点：适应性强、灵活

### ■ 缺点：不能利用已存在的某些有用的功能

- TCP/IP总是认为其下层是不可靠的（尽管可能已经足够可靠）



## 3.2.2 TCP/IP：网际/网络层（第2层）

### ■ 功能

- 把数据报通过最佳路径送到目的端（提供的是一种无连接、不可靠，但尽力而为的数据报传输服务）
- 寻址（IP地址）、路由选择、封包/拆包

### ■ PDU（数据单位）：数据报，也就是分组

### ■ 协议

- 网际层的核心协议——IP，提供了无连接的数据报传输服务（不保证送达，不保证顺序，不保证无错）
  - 传输前不需建立连接
  - 提高了传输效率
- ### ■ 网际层是网络转发节点（如路由器）上的最高层
- 网络节点设备不需要传输层和应用层



## ■ 网际层的其他重要协议

### • **ICMP** (Internet Control Message Protocol) 传递控制消息

- 可达性测试
- 传送路由状态信息
- 超时通知
- 不可达通知
- 封装在IP中进行传输

### • **ARP** (Address Resolution Protocol)

- 为已知的IP地址确定相应的MAC地址

### • **RARP** (Reverse Address Resolution Protocol)

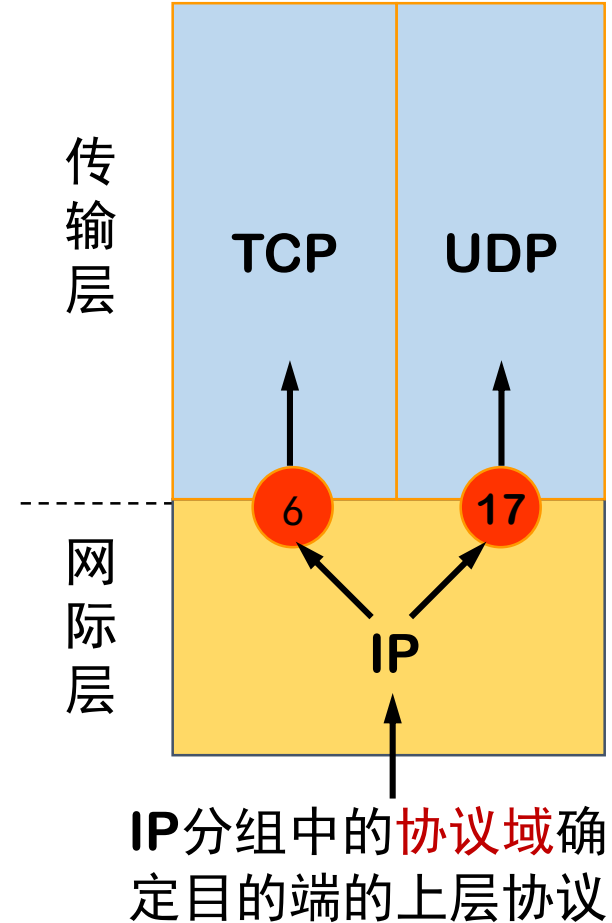
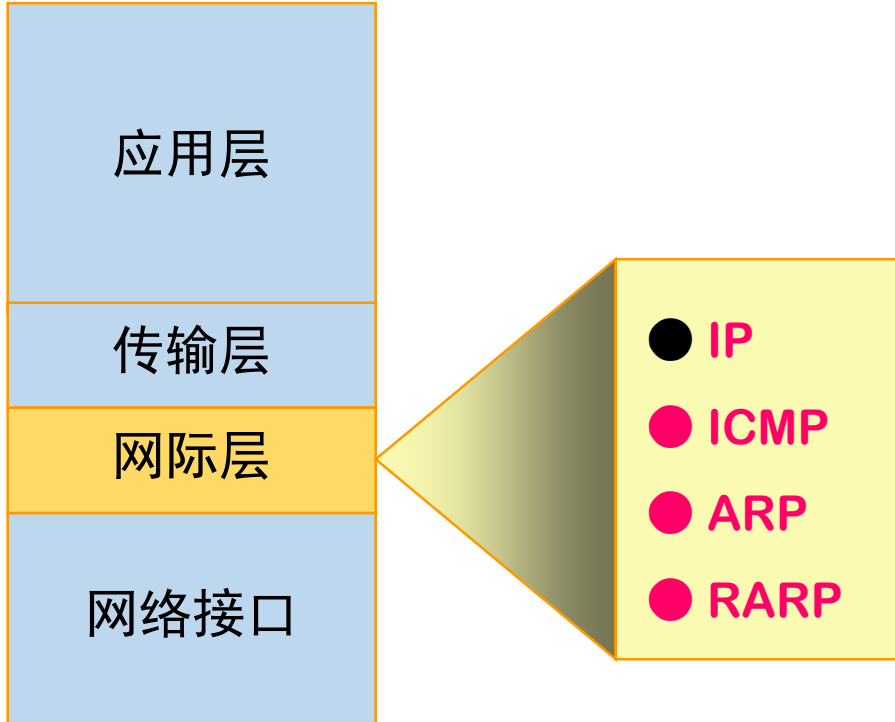
- 为已知的MAC地址确定相应的IP地址

### • **IGMP** (Internet Group Management Protocol)

- 多播组管理



# 网际层的四个主要协议



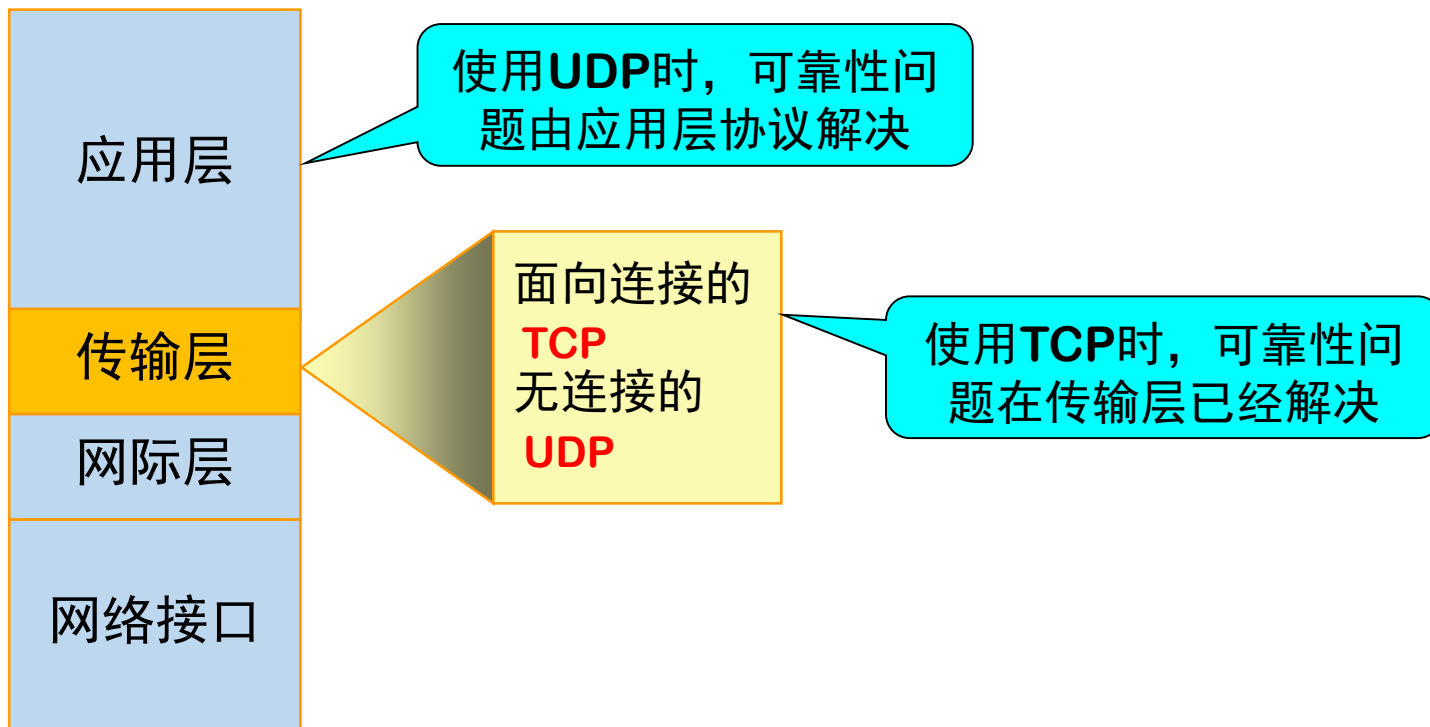


## 3.2.2 TCP/IP：传输层（第3层）

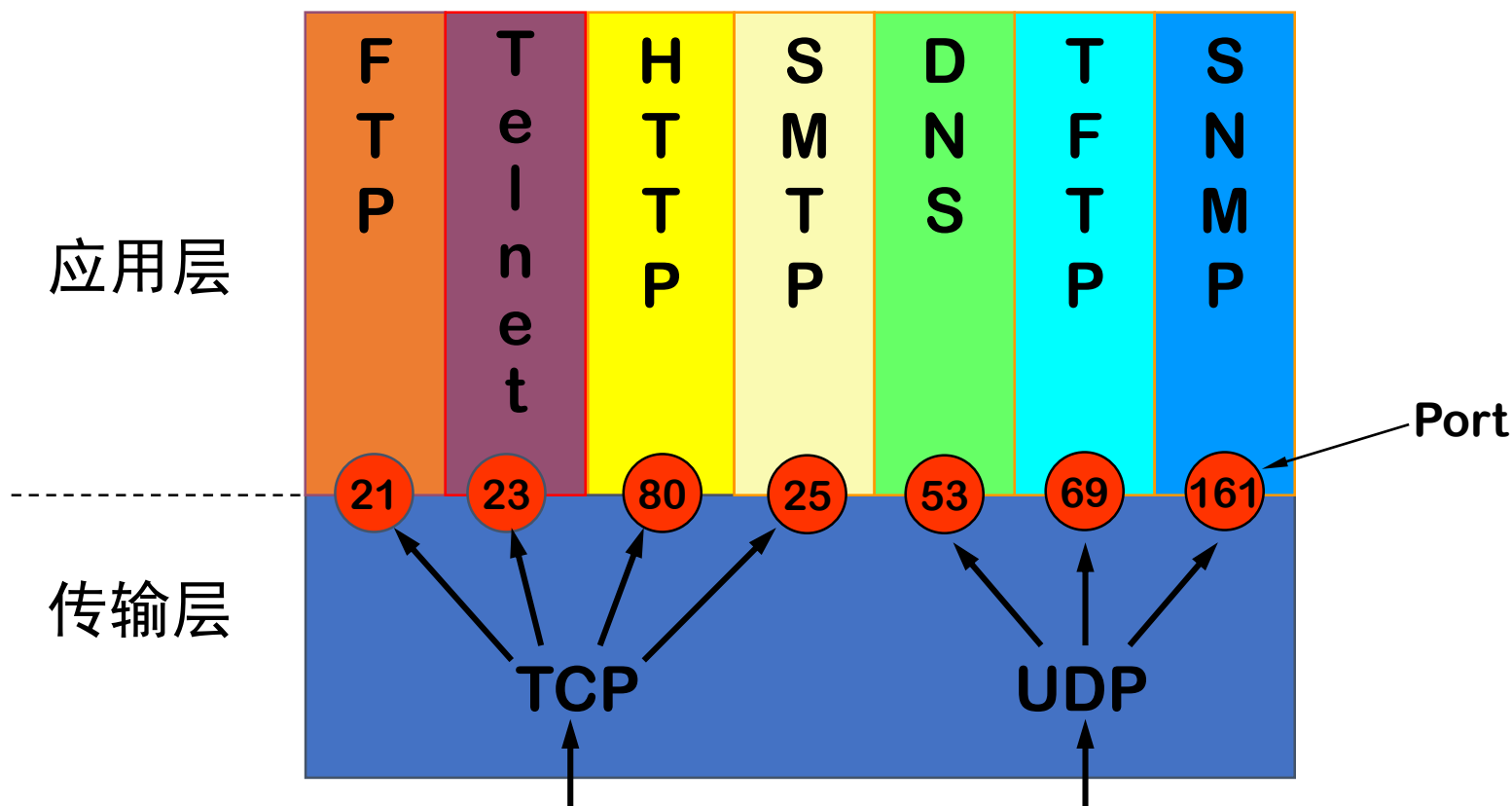
- **功能**：为应用进程提供可靠的端到端的**传输服务**
- **协议**：传输层包括**TCP**和**UDP**两种传输协议
  - **TCP**：**面向连接的**传输协议
    - 在数据传输之前**建立连接**
    - 必要时**重新传输**没有收到或错误的段，因此它是“**可靠**”的
  - **UDP**：**无连接的**传输协议
    - 在数据传输之前**不建立连接**
    - 对发送的段**不进行校验和确认**，因此它是“**不可靠**”的

## 3.2.2 TCP/IP：传输层（第3层）

- **功能**：为应用进程提供可靠的端到端的**传输服务**
- **协议**：传输层包括**TCP**和**UDP**两种传输协议



- **TCP**和**UDP**都用**端口(port)**号来识别应用层实体，以便准确地把信息提交给上层对应的协议（进程）



主机上海二区 - [redacted] 88 - 远程桌面连接

XAMPP Control Panel v3.2.4 [ Compiled: Jun 5th 2019 ]

### XAMPP Control Panel v3.2.4

Service	Module	PID(s)	Port(s)	Actions
<input checked="" type="checkbox"/>	Apache	1664	80, 443	Stop Admin Config Logs
<input checked="" type="checkbox"/>	MySQL	1768	3306	Stop Admin Config Logs
<input checked="" type="checkbox"/>	FileZilla	1020	21, 14147	Stop Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input checked="" type="checkbox"/>	Tomcat			Start Admin Config Logs

Config

Netstat

Shell

Explorer

Services

Help

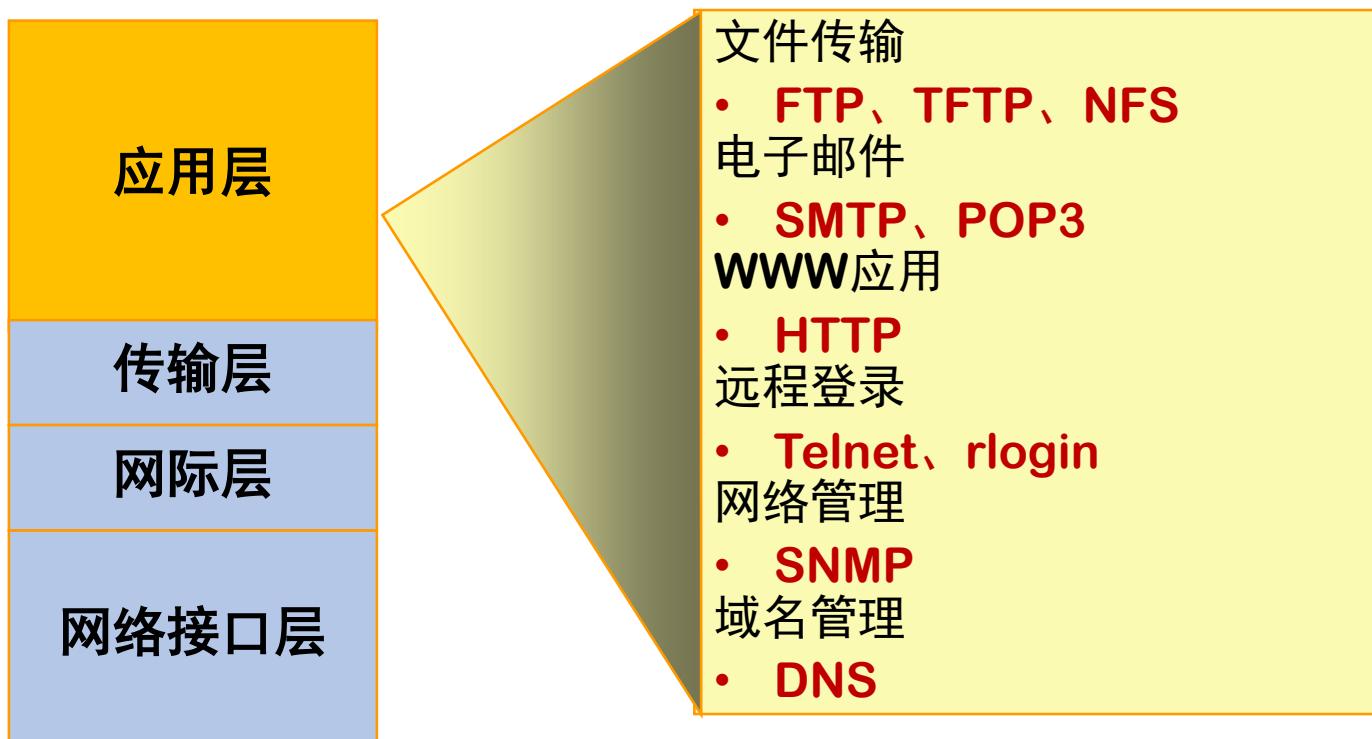
Quit

```

15:59:42 [main]      Initializing Control Panel
15:59:42 [main]      Windows Version: 64-bit
15:59:42 [main]      XAMPP Version: 7.3.11
15:59:42 [main]      Control Panel Version: 3.2.4 [ Compiled: Jun 5th 2019 ]
15:59:42 [main]      Running with Administrator rights - good!
15:59:42 [main]      XAMPP Installation Directory: "c:\xampp\"
15:59:42 [main]      Checking for prerequisites
15:59:47 [main]      All prerequisites found
15:59:47 [main]      Initializing Modules
15:59:47 [Apache]    XAMPP Apache Service is already running on port 80
15:59:47 [Apache]    XAMPP Apache Service is already running on port 443
15:59:47 [mysql]     XAMPP MySQL Service is already running on port 3306
15:59:47 [filezilla] XAMPP FileZilla Service is already running on port 21
15:59:47 [filezilla] XAMPP FileZilla Service is already running on port 14147
15:59:47 [main]      Starting Check-Timer
15:59:47 [main]      Control Panel Ready
    
```

## 3.2.2 TCP/IP：应用层（第4层）

- **功能：** 对应OSI的高三层，提供面向用户的网络服务
- **协议：** 应用层协议为文件传输、电子邮件、远程登录、网络管理、Web浏览等应用提供了支持。具体见第四章。





# OSI与TCP/IP参考模型比较

## ■ 相同点

- 以**协议栈**为基础，层次功能大体一致
- **传输层之上**的各层都是传输服务的**用户**，并且都是**面向应用的用户**

## ■ 不同点

- **OSI模型在前，协议在后**，使得协议相对更加容易被替换为新的协议；**TCP/IP模型在后，协议在前**，模型不适合其他协议栈。
- **OSI模型的网络层**同时支持**无连接**和**面向连接**的通信，但是**传输层上只支持面向连接**的通信；**TCP/IP模型的网际层**上只有一种模式(即**无连接通信**)，但是在**传输层上同时支持两种**通信模式。



## 3.3 五层参考模型

### ■ OSI与TCP/IP参考模型网络的评价

#### • OSI参考模型与协议：

- 缺乏市场与商业动力，结构复杂，实现周期长，运行效率低，这是它没有能够达到预想的目标的重要原因

#### • TCP/IP参考模型：

- 在服务、接口与协议的概念区别上不是很清楚
- TCP/IP参考模型不通用
- TCP/IP参考模型的网络接口层并不是常规意义上的层次概念
- TCP/IP参考模型没有区分物理层和数据链路层





## 3.3 五层参考模型

- 五层网络参考模型：吸收OSI参考和TCP/IP参考模型的优点

OSI/RM	TCP/IP	五层体系结构
高层 (5~7)	应用层	应用层
传输层 (4)	传输层	传输层
网络层 (3)	网际层	网络层
数据链路层 (2)	网络接口层	数据链路层
物理层 (1)		物理层



# 本章小结

- 网络体系结构：分层+协议+接口与服务
- 网络体系结构的层次模型：
  - 为何要分层？优点是什么？
- 网络体系结构的一些基本概念
  - 实体、协议、PDU、封装、服务以及它们之间的关系
- 对等层通信的实质：
  - 信息是如何在端到端之间传输的？
- 协议的三要素：语法、语义、时序
- 网络体系结构主要层次的功能：
  - 应用层、传输层、网络层、数据链路层和物理层



# 作业

---

## ■ 第88页：

1, 9, 11

补充题：计算机网络体系结构中对等层通信的实质是什么？