# 网络编程

## 协议与网络分层

### 协议

协议：是规定发送与接收的规则。

**典型协议**：TCP/UDP,HTTP,FTP,IP,ARP（IP找mac地址）

### 网络开发模式

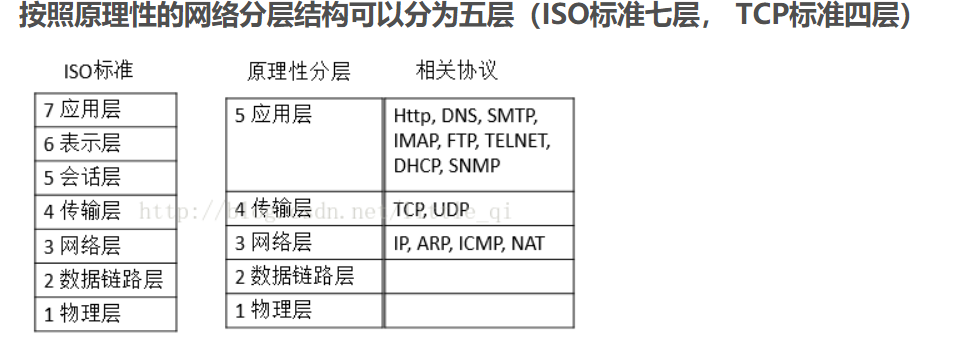
C/S

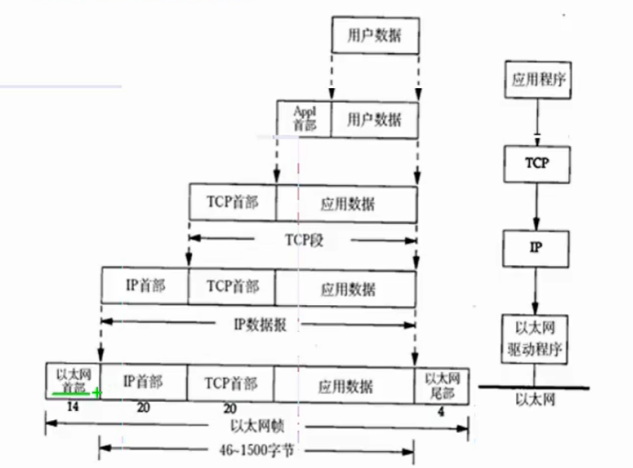
B/S (只能支持HTTP)

### 网络分层

**OSI七层**（最底层为物理层，顶层为应用层）；

开发的时候关注应用层，会逐级往下包装协议（ 应用层(http)--->数据链路层(以太网协议) ）





## TCP与UDP协议

### 2.1 TCP与UDP区别

TCP：

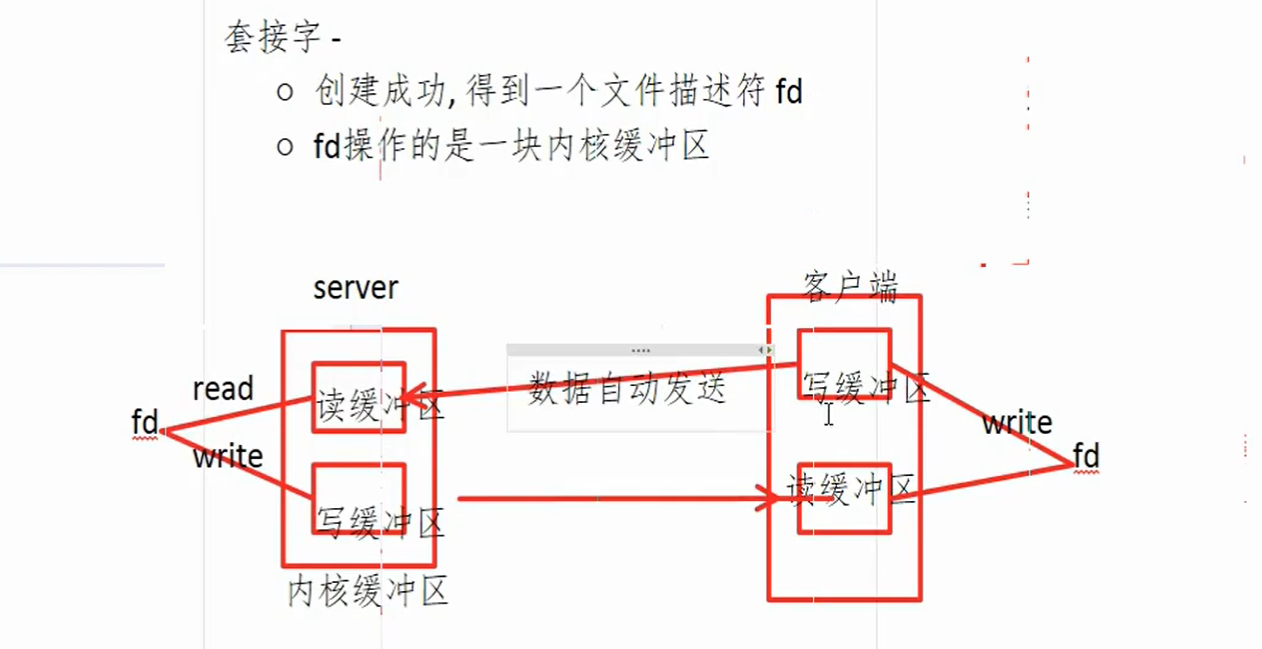
1. 面向连接的安全的流式传输协议；
2. 连接时，会进行三次握手。
3. 发送数据，会进行确认（丢失重传）。
4. 速度慢，消耗性能大。

UDP:

1. 面向无连接的不安全的报式传输；
2. 数据传输不安全，没有确认机制。
3. 报式传输（传报文，丢失会全部丢失）。
4. 速度快，消耗性能小。

## Socket

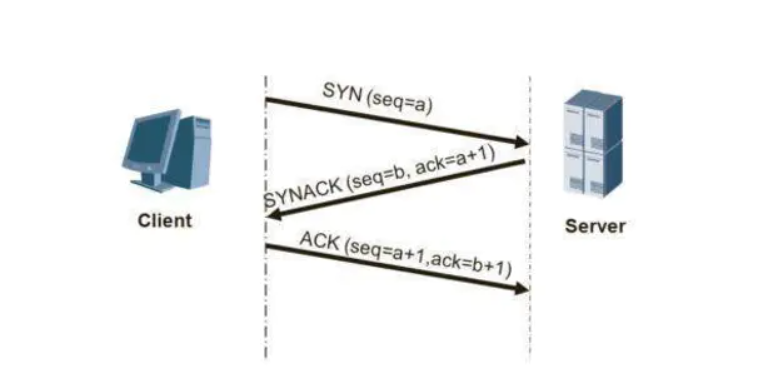
数据会先写到自己的写缓冲区。然后自动发送到对方的读缓冲区。



## TCP的发送

三次握手，4次分手，滑动窗口（一个缓冲区）

**三次握手：**

  
第一次握手：客户端发送syn包(syn=x)到服务器，并进入SYN\_SEND状态，等待服务器确认；  
第二次握手：服务器收到syn包，必须确认客户的SYN（ack=x+1），同时自己也发送一个SYN包（syn=y），即SYN+ACK包，此时服务器进入SYN\_RECV状态；  
第三次握手：客户端收到服务器的SYN＋ACK包，向服务器发送确认包ACK(ack=y+1)，此包发送完毕，客户端和服务器进入ESTABLISHED状态，完成三次握手。  
握手过程中传送的包里不包含数据，三次握手完毕后，客户端与服务器才正式开始传送数据。理想状态下，TCP连接一旦建立，在通信双方中的任何一方主动关闭连接之前，TCP 连接都将被一直保持下去。  
**四次挥手**  
与建立连接的“三次握手”类似，断开一个TCP连接则需要“四次握手”。  
第一次挥手：主动关闭方发送一个FIN，用来关闭主动方到被动关闭方的数据传送，也就是主动关闭方告诉被动关闭方：我已经不 会再给你发数据了(当然，在fin包之前发送出去的数据，如果没有收到对应的ack确认报文，主动关闭方依然会重发这些数据)，但是，此时主动关闭方还可 以接受数据。  
第二次挥手：被动关闭方收到FIN包后，发送一个ACK给对方，确认序号为收到序号+1（与SYN相同，一个FIN占用一个序号）。  
第三次挥手：被动关闭方发送一个FIN，用来关闭被动关闭方到主动关闭方的数据传送，也就是告诉主动关闭方，我的数据也发送完了，不会再给你发数据了。  
第四次挥手：主动关闭方收到FIN后，发送一个ACK给被动关闭方，确认序号为收到序号+1，至此，完成四次挥手。

