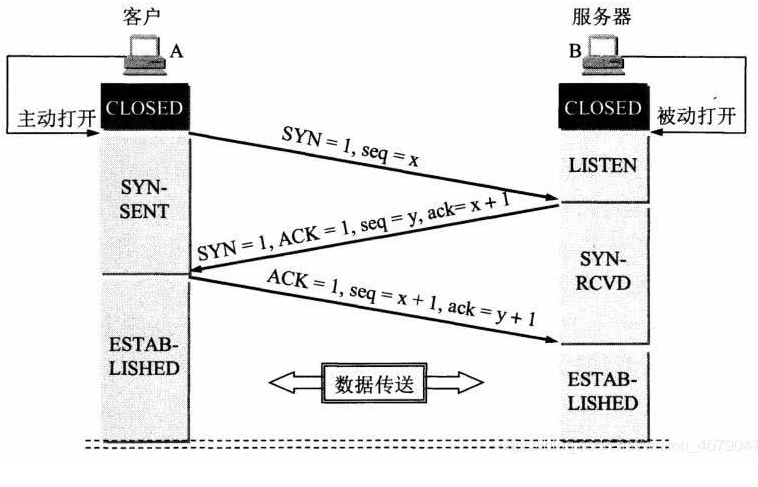
# 三次握手

TCP是面向连接的协议。TCP建立连接的过程叫做握手，握手需要在客户和服务器之间交换三个TCP报文段。

## 三次握手示意图



## 三次挥手过程

如图，主机A是客户端，B是服务器，起初两端的TCP进程都处于CLOSE（关闭）状态。主机下面的方框代表TCP进程所处的状态。

首先是A向服务器B发出请求连接报文段，发送的内容为 SYN = 1，seq = x。TCP客户端进入SYN-SENT（同步已发送）状态。SYN是TCP首部的同步位，在建立连接时用来同步序号。SYN = 1表明这个报文是连接请求或者连接接受报文。seq是指报文段序号，因为TCP是面向字节流的（但是传送的是报文段，也就是包含多个字节），在TCP连接中传送的字节流中的每一个字节都按序号编号，这个seq就是这个报文段所发送数据的第一个字节的序号

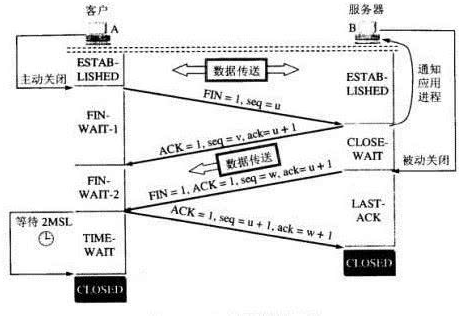
B收到请求报文段后，则向A发送确认报文，内容是SYN = 1，ACK = 1，seq = y，ack = x + 1。B进入SYN-RCVD(同步已发送)状态。这里ACK是表示确认（acknowledgement），TCP中规定在建立连接后所有传送的报文段都要把ACK置为1，当ACK = 0时，报文无效。seq = y 是确认报文第一个字段用y表示。ack是确认号，是期望收到对方下一个报文段的第一个数据字节的序号，这里ack = x +1 表明B已经收到了x号以前的报文，如果A再发报文过来，我希望报文第一个字节序号为x+1,但是A请求连接时发送seq = x，这里就说明了A向B发送的请求报文只有一个字节，是的，这就是TCP中的规定，SYN = 1的报文不能携带数据，但是要消耗掉一个序号

客户端收到B的确认后，还要向B发出确认，确认报文为ACK = 1，ack =y+1，seq = x + 1。A进入ESTABLISHED(已建立连接)状态，B收到A的确认后也进入ESTABLISHED(已建立连接)状态

# 四次挥手

数据传输结束后，客户端和服务器都可以释放连接。

## 四次挥手示意图



## 四次挥手过程

目前A和B都处于ESTABLISHED(已建立连接)状态。这个过程简单来说就是A说我要关闭了，B回答说我知道了，B说我要关闭了，A回答说我知道了，然后就都关闭了。

A先发出连接释放报文段，主动关闭TCP连接，报文内容为FIN = 1，seq = u，A进入FIN-WAIT-1(终止等待1)状态。FIN(finis)表示终止，FIN = 1 时，表明此报文段的发送方的数据已经发送完毕，并要求释放连接。FIN报文段不携带数据，但是消耗一个序号。

B收到连接释放报文段后发出确认，报文内容为ACK = 1，seq = v，ack = u+1，B进入CLOSE-WAIT(关闭等待)状态，A收到B的确认后进入FIN-WAIT-2(终止等待2)转态。

若B没有要向A发送的数据，B就发送连接释放报文，报文内容FIN = 1，ACK = 1，seq = w，ack = u+1。B进入LAST-ACK(最后确认)状态。这里B还要重复发送已经发送过的确认号ack = 1。

A收到B的连接释放报文后对此进行确认，确认报文为ACK = 1 ，seq = u+1，ack = w+1，A进入TIME-WAIT(时间等待)状态。要注意的是，现在TCP连接还没有没释放，必须经过时间等待计时器设置的时间2\*MSL后，A才进入CLOSE状态。MSL（maximum segment lifetime）是最长报文段寿命。