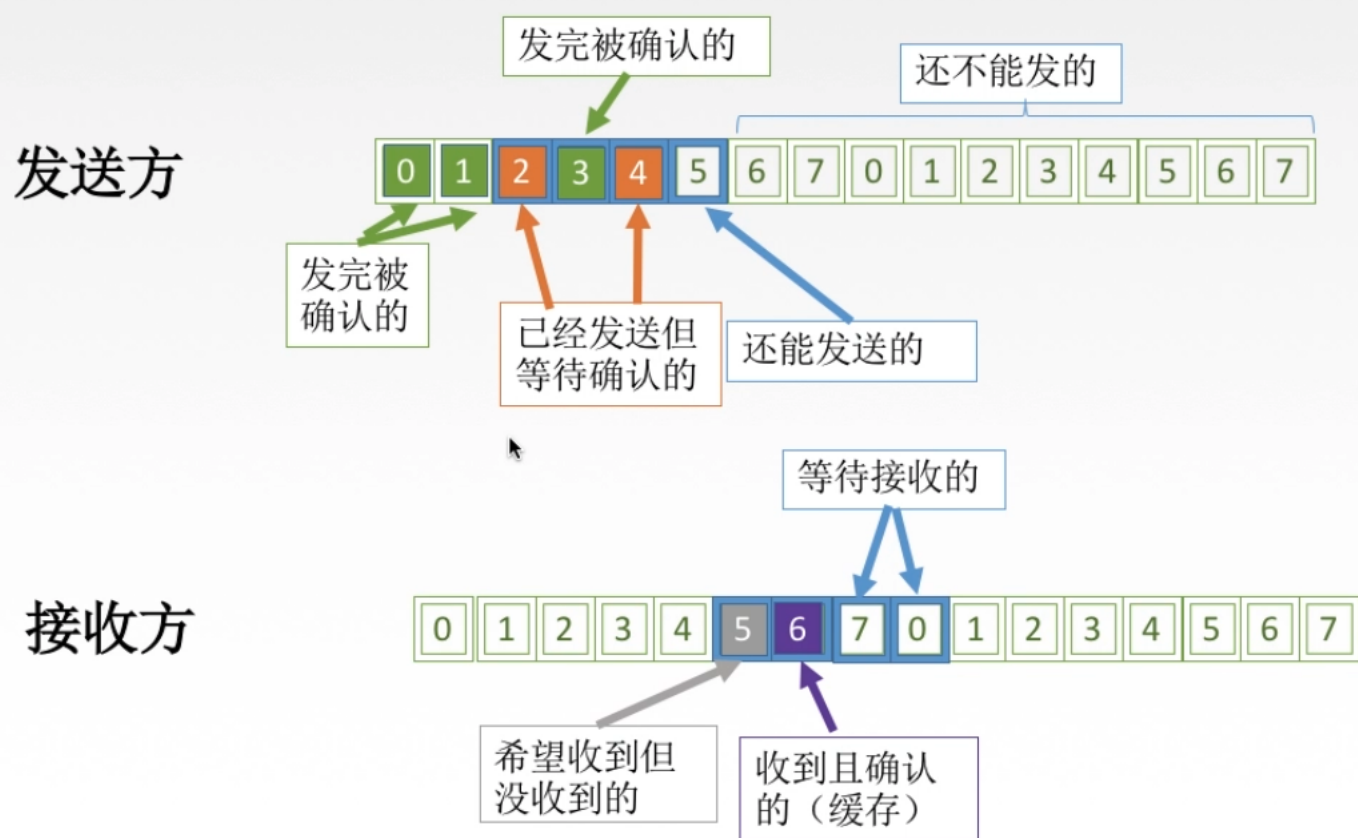


能否只重传出错的帧？

一、选择重传协议 (SR) 中的滑动窗口



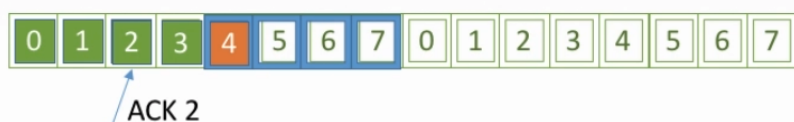
二、SR 发送方要响应的几件事

1. 上层的调用

从上层收到数据后，SR发送方检查下一个可用于该帧的序号，如果序号位于发送窗口内，则发送数据帧；否则就像GBN一样，要么将数据缓存，要么返回给上层之后再传输。

2. 收到了一个ACK

如果收到ACK，加入该帧序号在窗口内，则SR发送方将那个被确认的帧标记为已接收。如果该帧序号是窗口的下界（最左边第一个窗口对应的序号），则窗口向前移动到具有最小序号的未确认帧处。如果窗口移动了并且有序号在窗口内的未发送帧，则发送这些帧。



3. 超时事件

每个帧都有自己的定时器，一个超时事件发生后只重传一个帧。

三、SR 接收方.

确认一个正确接收的帧。(无论是位按序)。

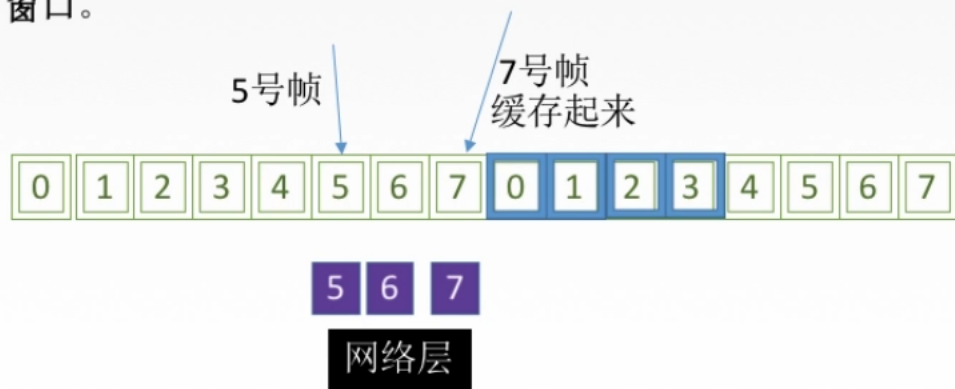
失序则缓存。

正序则交付于上层并向前移动窗口。

返回给发送方该帧的确认帧 (收谁确认谁).

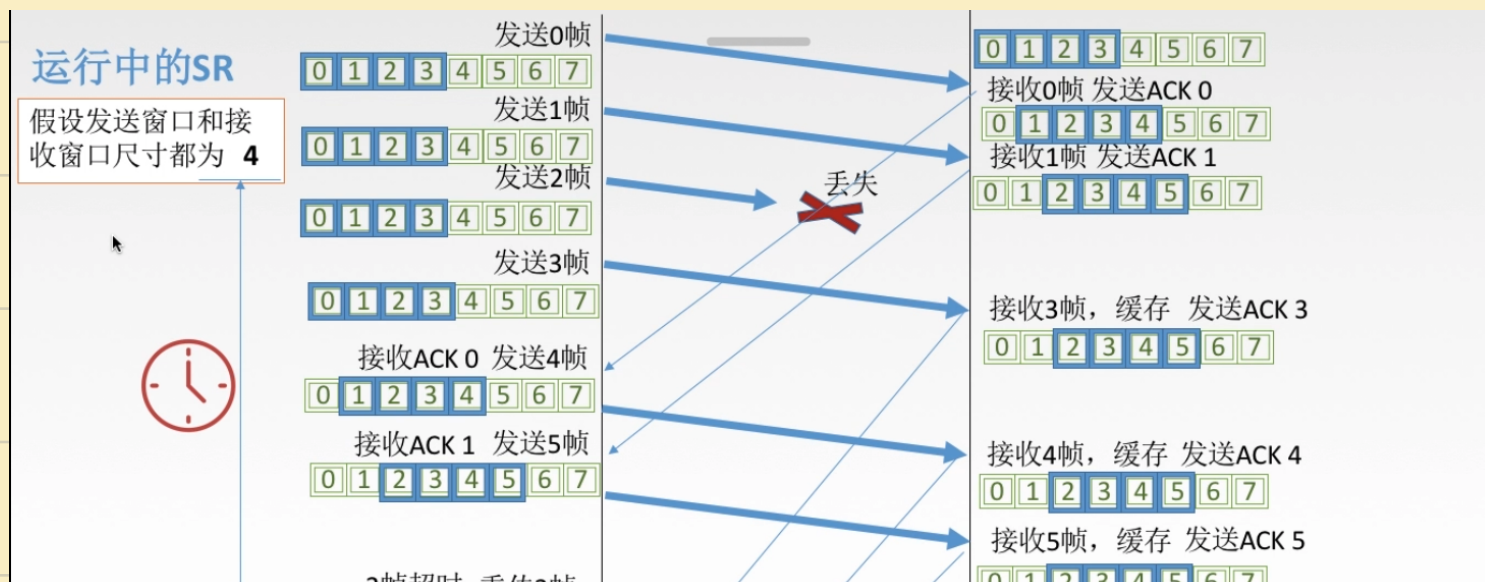
来者不拒（窗口内的帧）

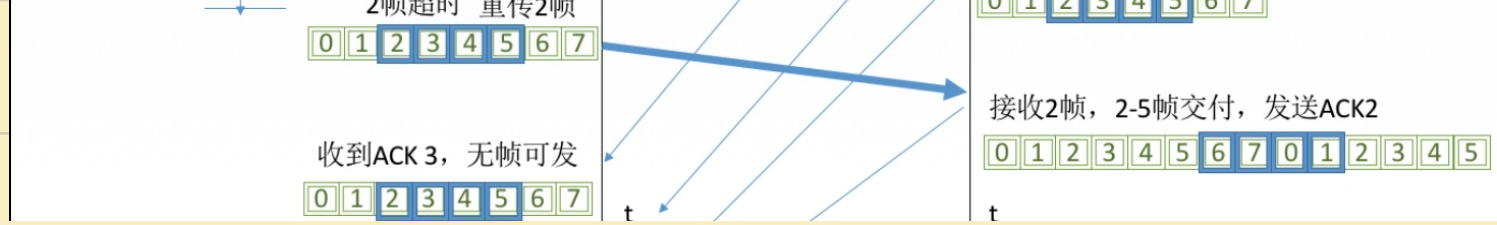
SR接收方将**确认一个正确接收的帧而不管其是否按序**。失序的帧将被**缓存**，并返回给发送方一个该帧的确认帧【**收谁确认谁**】，直到所有帧（即序号更小的帧）皆被收到为止，这时才可以将一批帧按序交付给上层，然后**向前移动滑动窗口**。



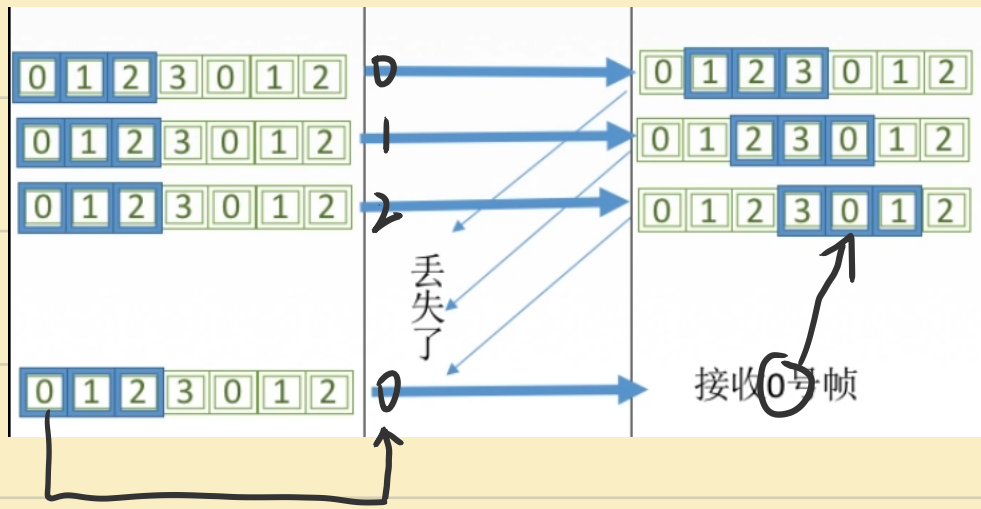
如果收到了窗口序号外（小于窗口下界）的帧，就返回一个ACK。其他情况，就忽略该帧。

四. 运行中的 SR.





五. 滑动窗口的长度.



窗口过大引起帧错位.

发送窗口最好等于接收窗口。(大了会溢出, 小了没意义)

$$W_{Tmax} = W_{Rmax} = 2^{(n-1)}$$

六. 重点总结.

1. 对数据帧逐一确认, 收一个确认一个
2. 只重传出错帧
3. 接收方有缓存

$$4. W_{Tmax} = W_{Rmax} = 2^{n-1}$$