# 【第04章-TCP粘包拆包问题的解决之道】【03-利用LineBasedFrameDecoder解决TCP粘包问题】

## 利用LineBasedFrameDecoder 解决TCP 粘包问题

为了解决TCP 粘包拆包导致的半包读写问题， Netty 默认提供了多种编解器用于处理半包， 只要能熟练掌握这此类库的使用， TCP 粘包问题从此会变得非常容易，你甚至不需要关心它们， 这也是其他NIO框架和J DK 原生NIO API 所无法匹敌的。

## LineBasedFrameDecoder和StringDecoder的原理分析

LineBasedFrameDecoder 的工作原理是它依次遍历ByteBuf中的可读字节， 判断看是否有“\n”或者“\r\n”如果有，就以此位置为结束位置，从可读索引到结束位置区间的字节就组成了一行。它是以换行符为结束标志的解码器，支持携带结束符或者不携带结束符两种解码方式，同时支持配置单行的最大长度。如果连续读取到最大长度后仍然没有发现换行符， 就会抛出异常，同时忽略掉之前读到的异常码流。

StringDecoder 的功能非常简单，就是将接收到的对象转换成字符串，然后继续调用后面的LineBasedFrameDecoder+ StringDecoder组合就是按行切换的文本解码器，它被设计用来支持TCP 的粘包和拆包。

可能读者会提出新的疑问： 如果发迭的消息不是以换行符结束的，那该怎么办昵？或者没有回车换行符， 靠消息头中的长度字段来分包怎么办？是不是需要自己写半包解码器？答案是否定的， Netty提供了多种支持TCP 粘包拆包的解码器，用来满足用户的不同诉求。