

MESSAGE PACKING & UNPACKING

Arthur Pai

WHAT'S THE PROBLEM IN THIS CHATROOM DEMO?

- An event/structure to replace sending string
- Thread termination
- TcpClient close
- Lock issue
- CPU cost by “while” / blocking API
- Memory footprint

MESSAGE PACKING

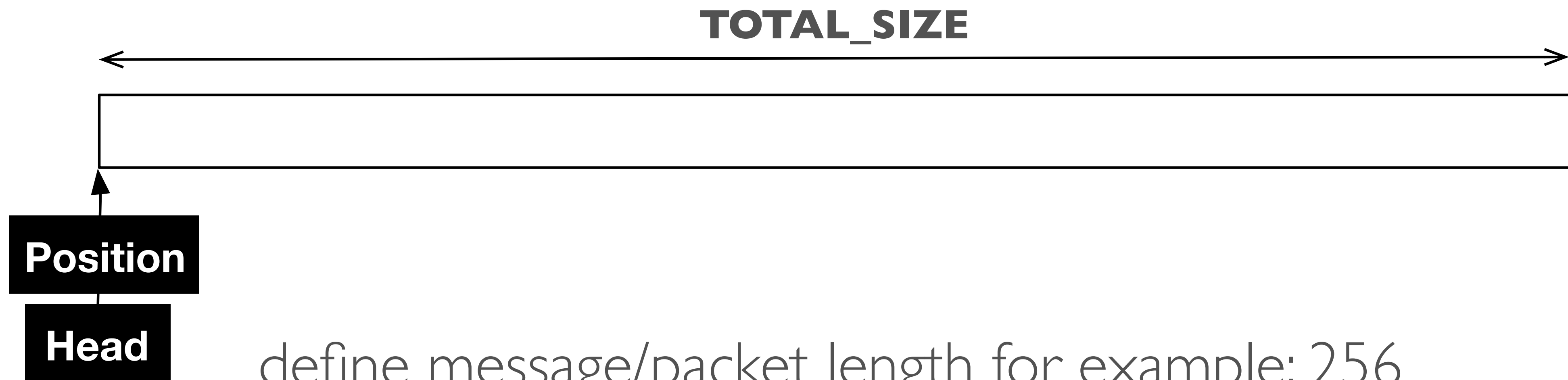
Message ID	Data 0	Data 1	Data 2	Data 3
------------	--------	--------	--------	--------

Message ID

use **constant integer** as ID

for defining the structure of transferring data

MESSAGE PACKING

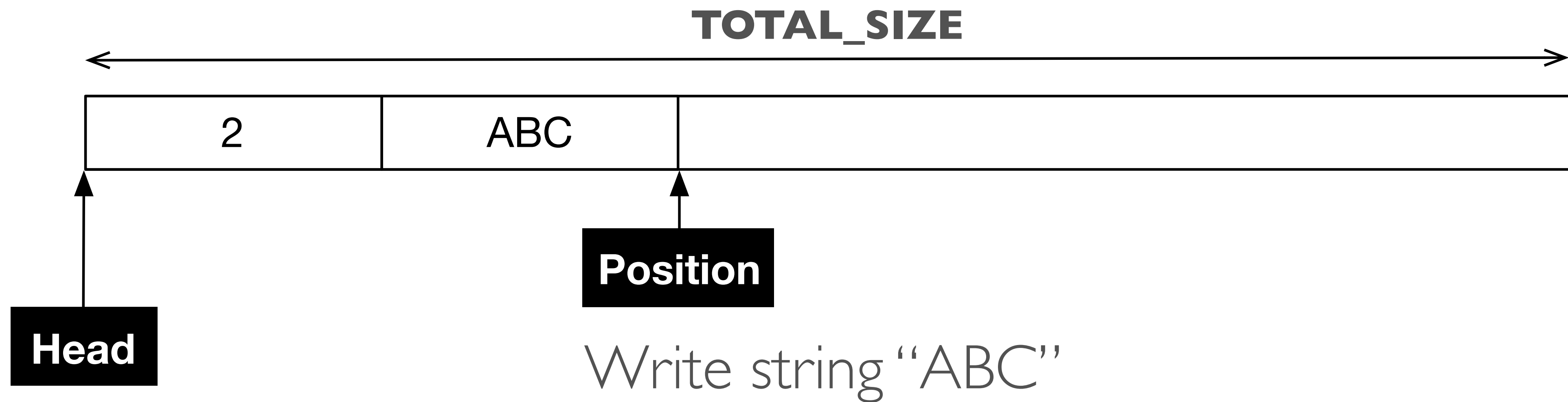


define message/packet length for example: 256
byte[] bytes = new byte[TOTAL_SIZE];

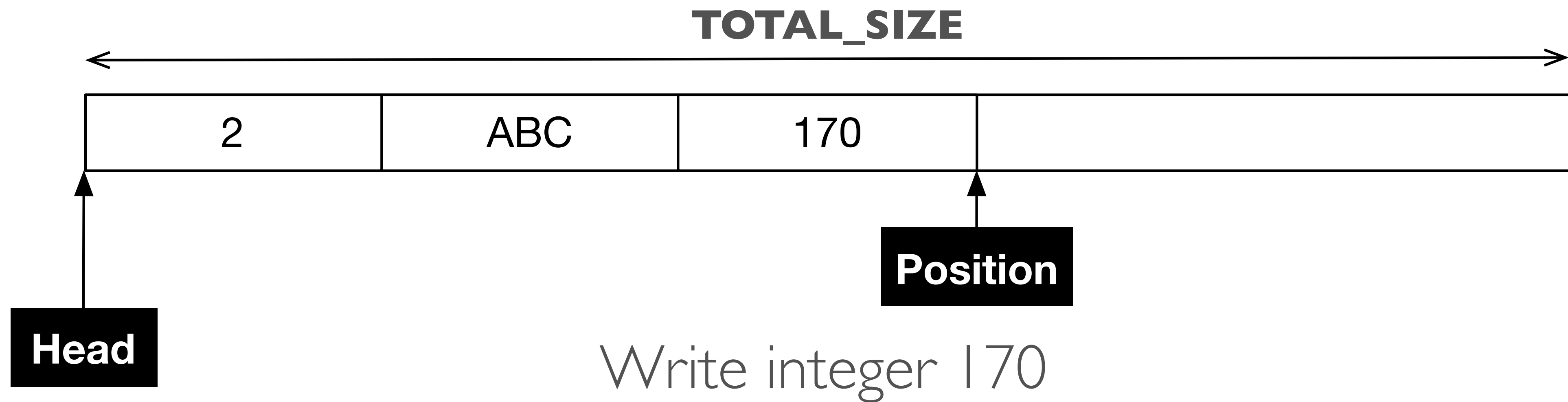
MESSAGE PACKING



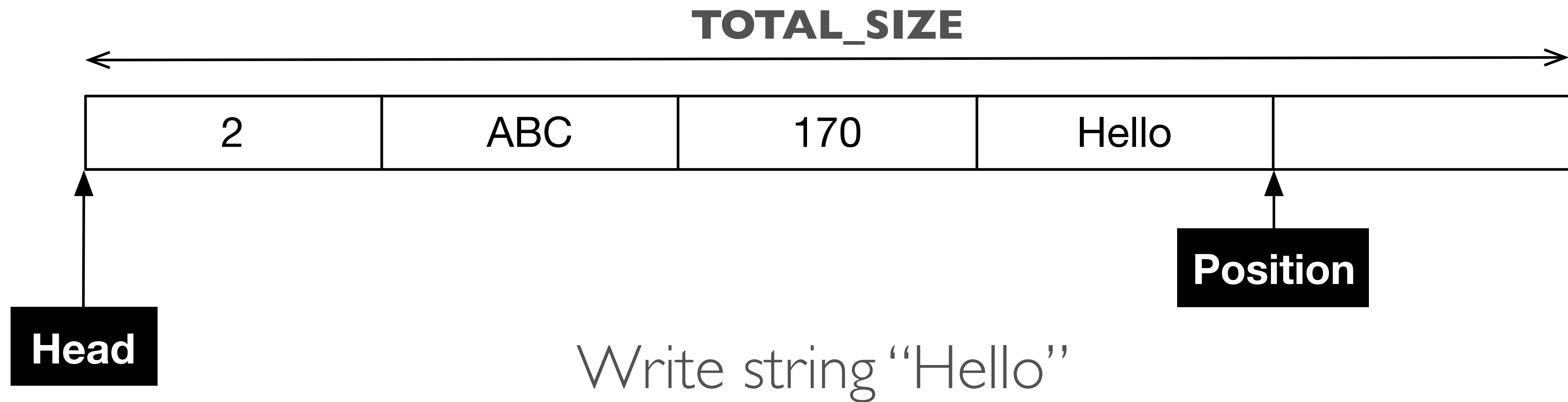
MESSAGE PACKING



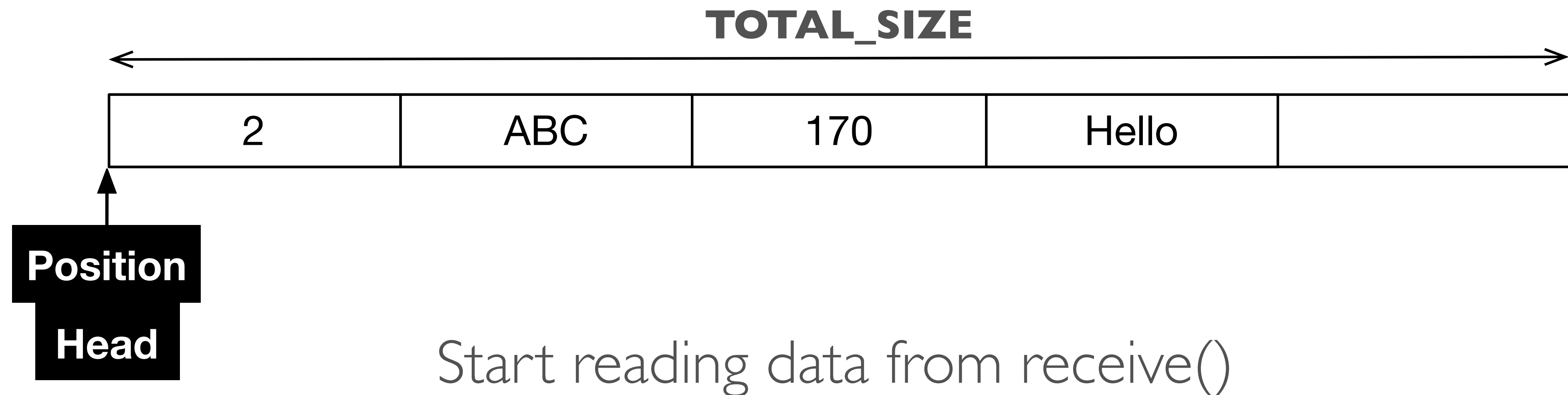
MESSAGE PACKING



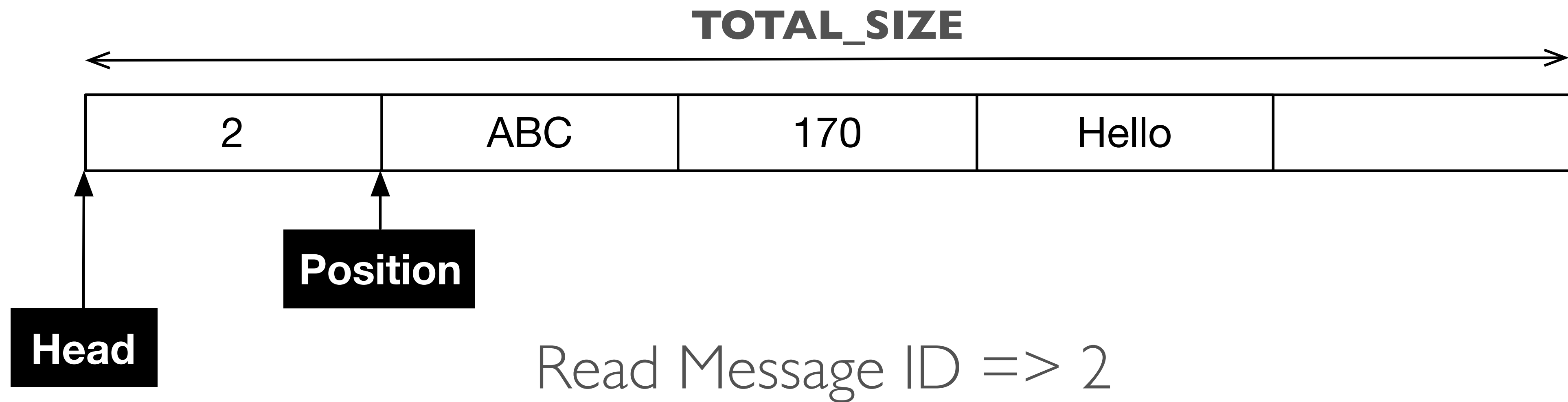
MESSAGE PACKING



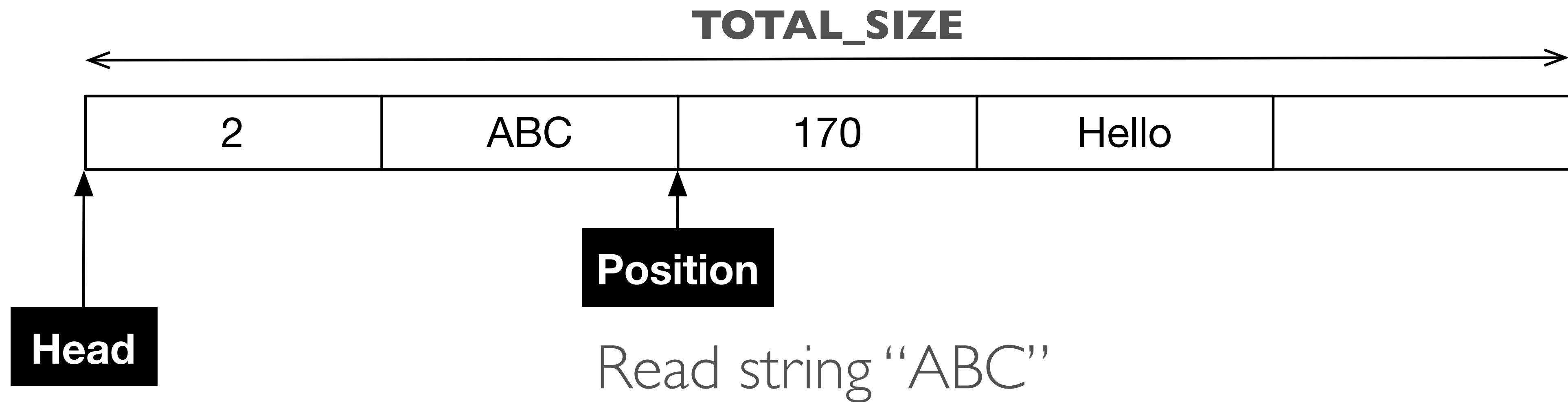
MESSAGE UNPACKING



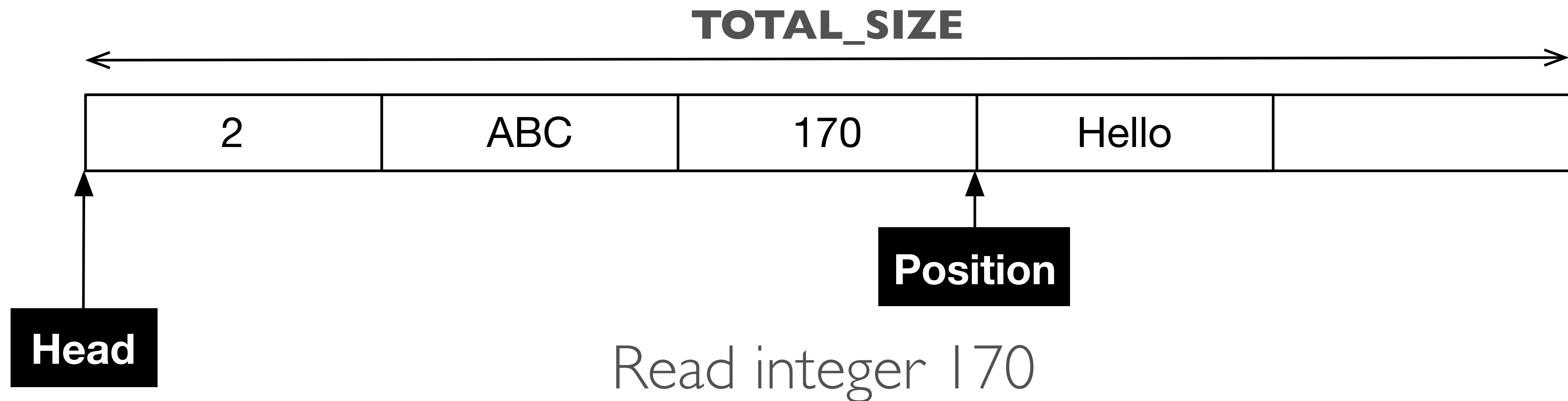
MESSAGE UNPACKING



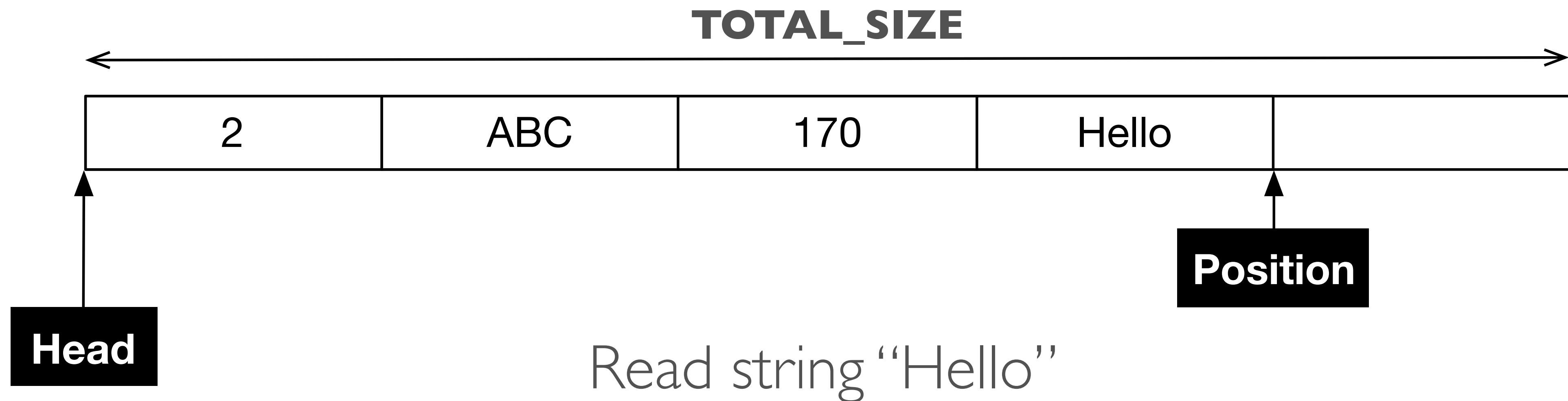
MESSAGE UNPACKING



MESSAGE UNPACKING



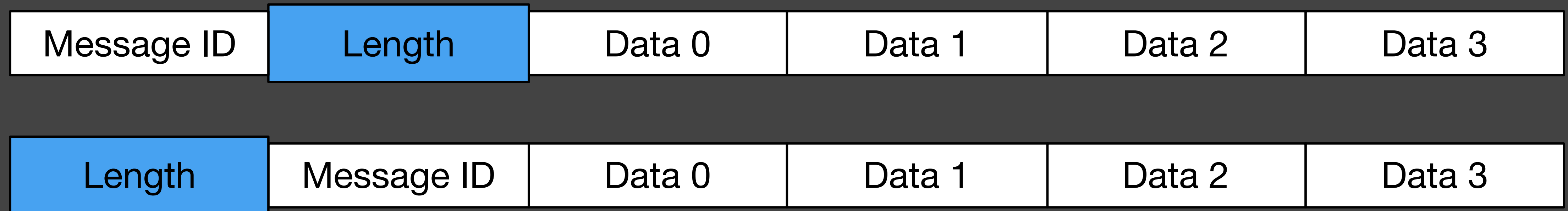
MESSAGE UNPACKING



EXERCISE 1

Client pack message then send to server for unpacking

Length?



We usually need “Length”
to indicate how long the message is

EXERCISE 2

Add Length to package

MORE ABOUT MESSAGE PACKING / UNPACKING

Message 1	Message 2	Message 3	Message 4
-----------	-----------	-----------	-----------

There's different meaning in “**Message**” and “**Packet**”

We might also need to know how many messages in a packet

Switch Protocol to
Struct/Buffer format
or
JSON format

Switch Protocol to Struct/Buffer format

