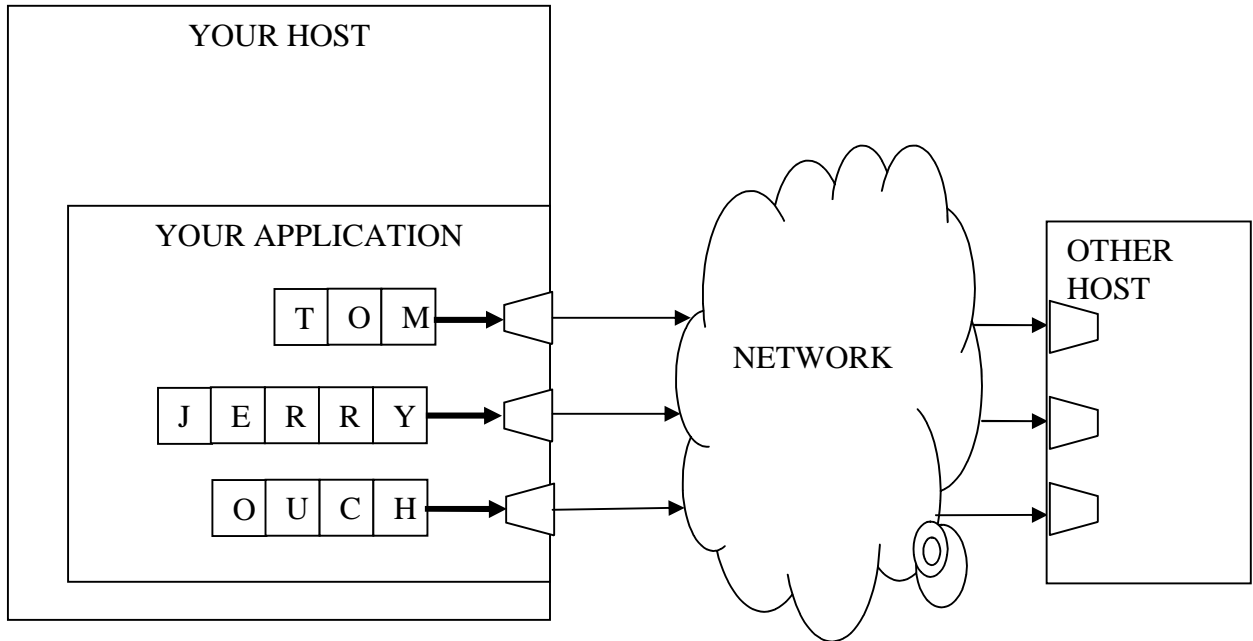


# HW #1 SOLUTION

COMS W4119 – Computer Networks  
Spring 2006

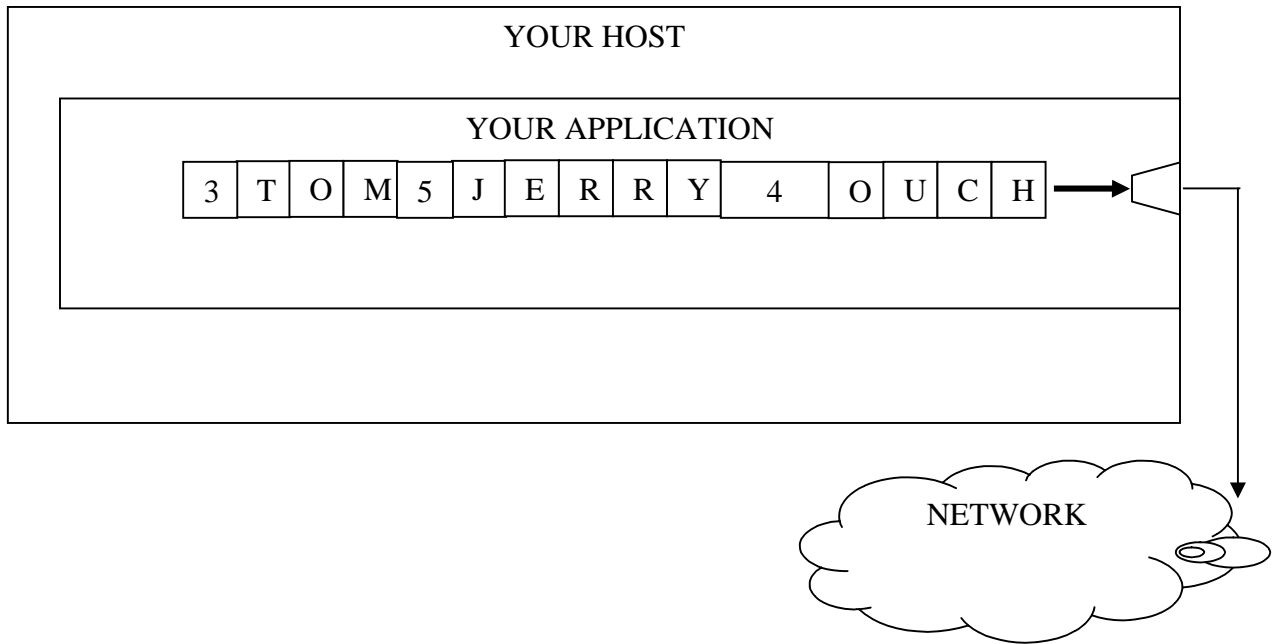
1.

- a) We dedicate three different sockets for the transmission. Each socket binds to a different port, thereby creating three independent transmission channels. All three components of a message are transmitted in parallel, each component on its own dedicated channel. The diagram below applies this solution to the required example:

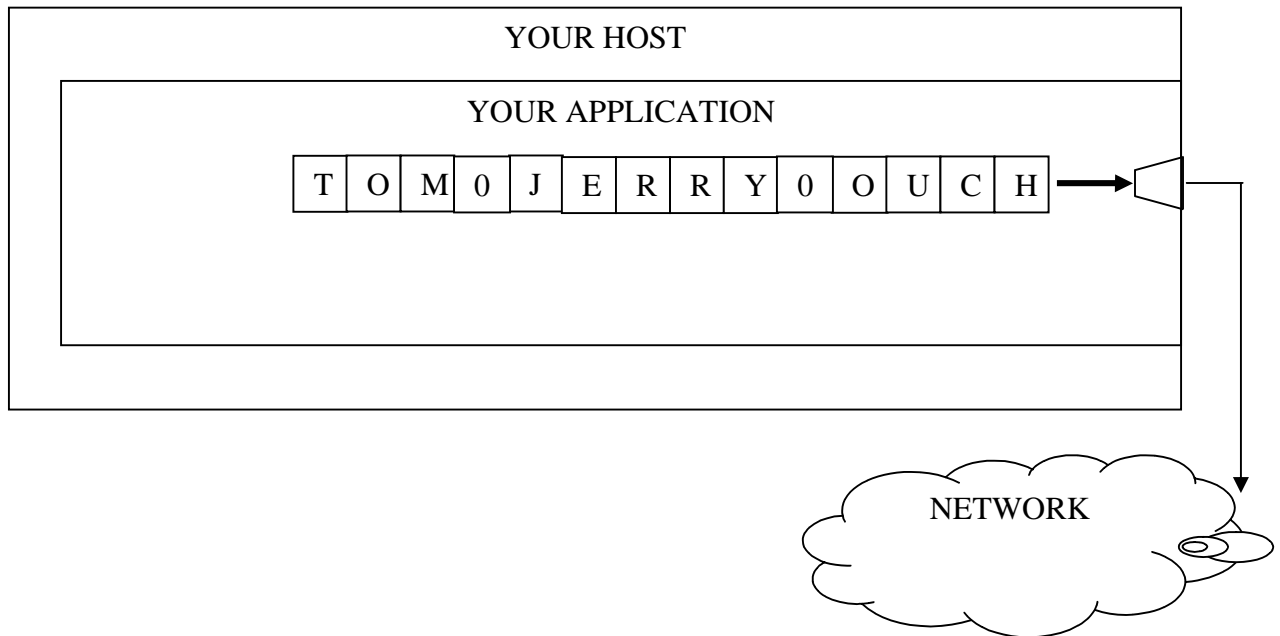


- b) Since only one connection can be used, all three components of a message have to be transmitted in one byte stream. We assign an arbitrary, predetermined order of the components in the stream. Say the sender name first, followed by the receiver name, then the message text. Since the length of every part of the stream can vary (depending on the length of the senders name etc.), we dedicate the first few bytes of every part of the stream to indicated the length of that part. The length of the sender/receiver name is required to be between 1 and 256, therefore 1 byte needs to be dedicated to indicate that length. The message text can be of any length between 1 and 65,536 which are more options that can be expressed by one byte

only. We need to dedicate at least two bytes to indicate that length. The figure below applies this method to the required example:

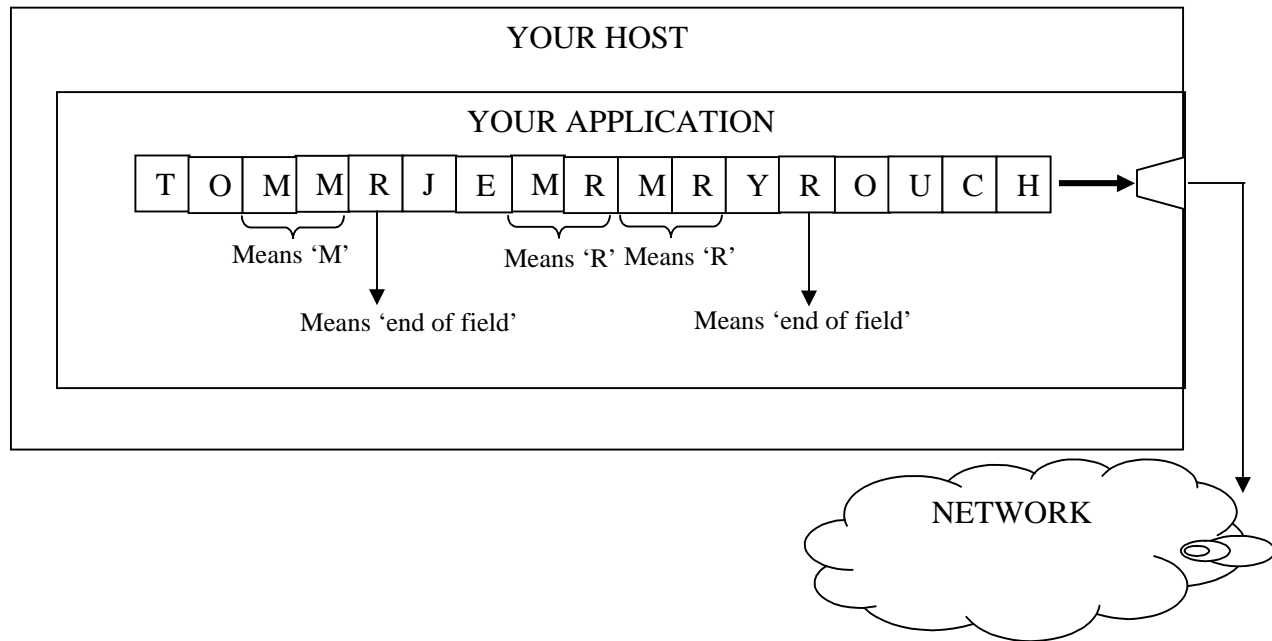


- c) We pick an arbitrary byte value that does not exist in the ASCII table, and assign it with the special task of being the last value of every field in our byte stream. We call such a value a delimiter. For example, the value 0 is a convenient delimiter since no entry in the ASCII table has value 0. This way we no longer need to include the field length in our message; the receiver will keep reading a field until it reads the delimiter (a byte with value 0), which will signal the end of a field. It will then start reading in the next field starting from the byte right after the delimiter and so on.



- d) If the user can send bytes of potentially all possible values, we can no longer reserve a special value to be used as a field delimiter. However, using a simple trick we can still assign a value to be used as delimiter even though it might also be used by the user as part of the message: Say we choose to use 'R' as a delimiter. We also choose another value, say 'M', to serve as a special 'escape' character. We replace every occurrence of the letter R in the message with two characters 'MR'. We also replace every occurrence of the letter M in the message with 'MM'. Now we insert 'R' at the end of every field as a field delimiter. The resulting string is shown in the diagram below.

The recipient of the message will understand every 'MR' character sequence to mean the letter R. It will also interpret every 'MM' sequence as the letter M. Furthermore, whenever the recipient reads a single 'R' in the message, it will treat it as a field delimiter.



2.

Firstly, note that since an identifier is  $k$ -bytes long, and every byte can have one of 36 possible values, the number of all valid identifiers to choose from  $I=36^k$

- a) Let A choose any identifier. Whatever choice A makes, B only has 1 choice out of  $I$  different identifiers in order for A and B to have the same identifier.

$$\text{Therefore } P = 1/I = 1/36^k$$

- b) Given a certain user User1,

$$\begin{aligned}
 P &= 1 - \text{Pr}[\text{no user has the same identifier as User1}] \\
 &= 1 - (\text{Pr}[\text{UserX has the same identifier as User1}]^{N-1}) \\
 &= 1 - (1 - 36^{-(k)})^{N-1}
 \end{aligned}$$

- c) For all users to have different identifiers, the first user can choose any identifier out of  $I$  possibilities, but the second user can only choose from a pool of  $(I-1)$  possibilities. Similarly, the third user has a smaller pool to choose from and the  $N$ th (last) user can only choose from the  $(I-N+1)$  remaining identifiers. Therefore,

$$\begin{aligned}
 P &= (I/I) * ((I-1)/I) * ((I-2)/I) * \dots * ((I-N+1)/I) \\
 P &= I! / ((I-N)! * I^N)
 \end{aligned}$$

And since  $I = 36^k$

$$P = (36^k)! / ((36^k - N)! * (36^{kN}))$$

- d) We want to set  $k$  so that at least  $N = 36^k$  so that there are  $N$  different valid identifiers.

$$\text{Therefore } k = \log_{36} N$$

$$\begin{aligned} \text{e) } I &= 36^1 + 36^2 + \dots + 36^k \\ &= (36^{(k+1)} - 36) / (36-1) \end{aligned}$$

Using the fact that  $x + x^2 + \dots + x^k = (x^{(k+1)} - x) / (x-1)$