

## Instant Messaging

### Problem Statement:

You are to build a peer-to-peer messaging interface between applications running on different computers. The applications should support a simple conversation between two parties. Further, an application protocol should be established that defines the rules for conversation.

### Constraints:

- Two conversing partners, designated as Primary and Secondary.
- Either partner may initiate the conversation.
- Each partner must run on a different computer with a unique IP address.
- (HINT: You can test both on the same machine using unique ports.)

### Initial Conditions:

Each side is waiting and ready to receive a request for communication.

### Operation:

Like a telephone conversation. One side initiates the call, and either side can make requests (questions) or respond to requests (answers). Either side can initiate termination.

### Data Displays:

Show the flow of 4 types of messages on each side of the conversation:

- Primary request followed by Secondary response
- Secondary request followed by Primary response

### Strategy

You want to establish a protocol that sets the rules for conversation:

Initiation of the conversation by either party (primary).

Acceptance by the other party (secondary).

Acknowledgment of the acceptance by primary.

Decide which party "talks" next.

Decide on the rules for continued conversation. How do you know when partners is done.

Timeout requirements if one party is waiting for other party's response.

Error recovery in the event of a timeout or unsynchronized conversation.

Ending the conversation gracefully.

Terminating the communication channels.

Reset and Wait for the next request for establishment.