Date 09/02/22
Page 18

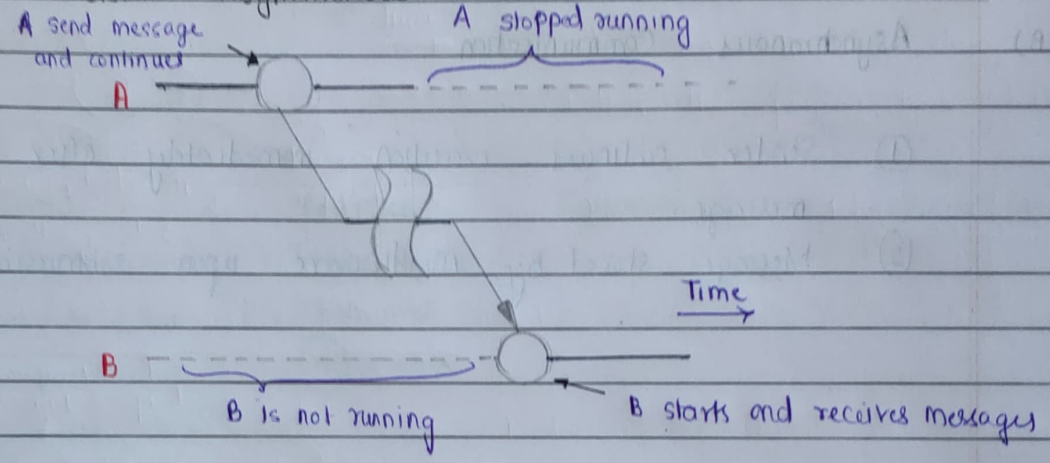
TUTORIAL - 5

1119CS012

1.) Answer the following questions:

a.) Give the explaination of each Figure. 1-6
b.) Mention the application scenario where each one is applicable

(I) Persistent Asynchronous

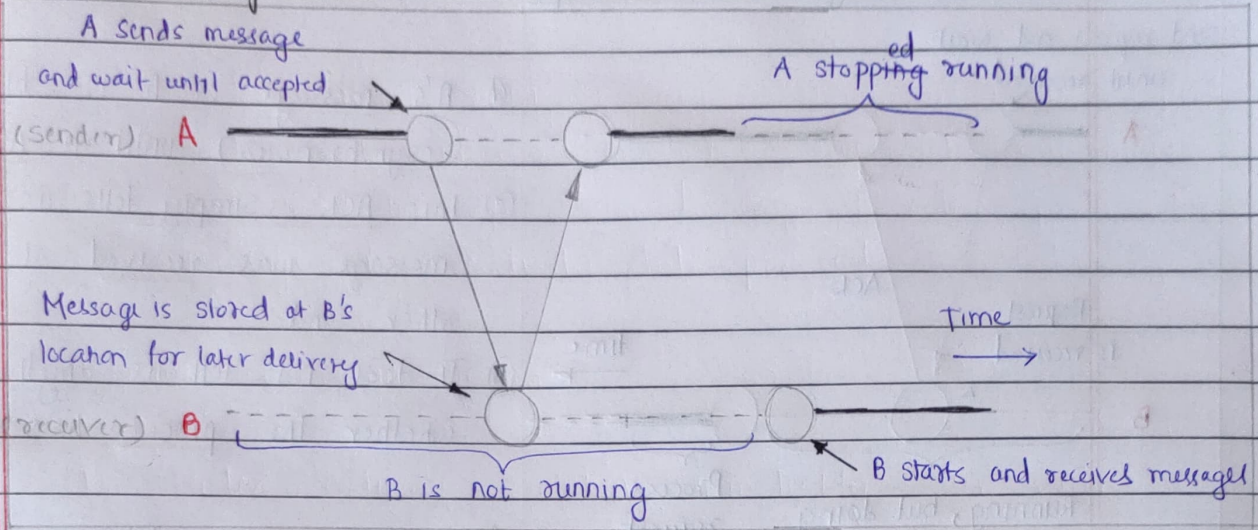


A → sender , B → receiver Dark line → Execution

① A sends a message and keeps executing without blocking.
② The message sent from A will take arbitrary amount of time to reach B.
③ A may or may not be running by the time message reaches B.
④ Disks or multiple memory queues could be used for storage at B's side.
⑤ There is a guarantee that message will eventually reach B.
⑥ Finally, B receives the request and processes it.

Application Scenerio - Emails can take arbitrary amount of time to reach, and when receiver reads the email, the sender might not even be running.

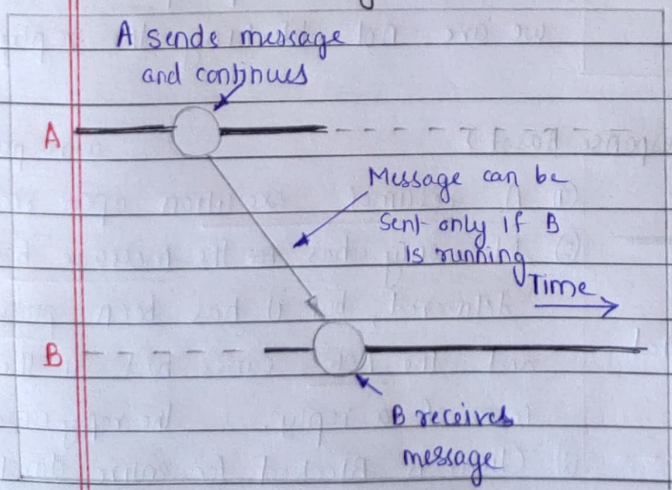
(II) Persistent Asynchronous



① A sends a message and is then blocked (depicted by dotted line)
② The acknowledgement is for receipt (not delivery/response).
③ Because this model is persistent, the message may stay in B's queue (or in any router along the way) for an arbitrary amount of time.

Application Scenario - Messaging and Chat Applications
[Many messaging systems are persistent, and can tell us the delivery status.

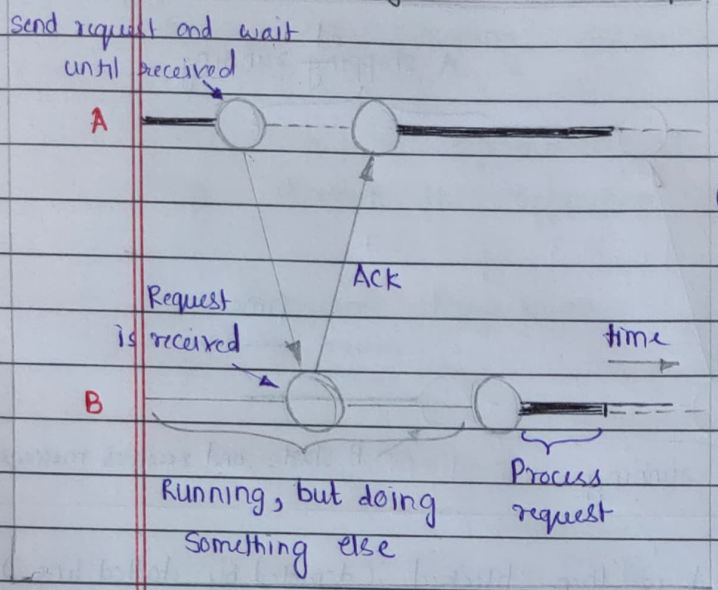
(III) Transient Asynchronous



① A sends the message and continues execution (non-blocking)
② B has to be running, because if it is not running, the message will be discarded.
③ Even if any router along the way is down, the message will be discarded.

Application Scenario - ① UDP communication
② MPI-bsend() function

(IV) Transient Synchronous (Receipt Based)



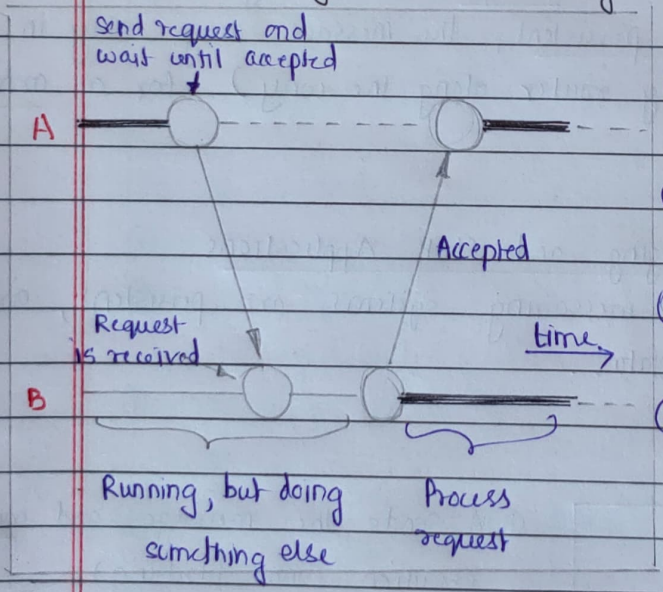
① A's message to B is blocking (synchronous) until an ACK is received

② This ACK simply tells us the message was received at the other end.

③ It does not tell us anything about whether the process has started.

Application Scenario - RPC's, RMI's

(V) Transient Synchronous (Delivery Based)



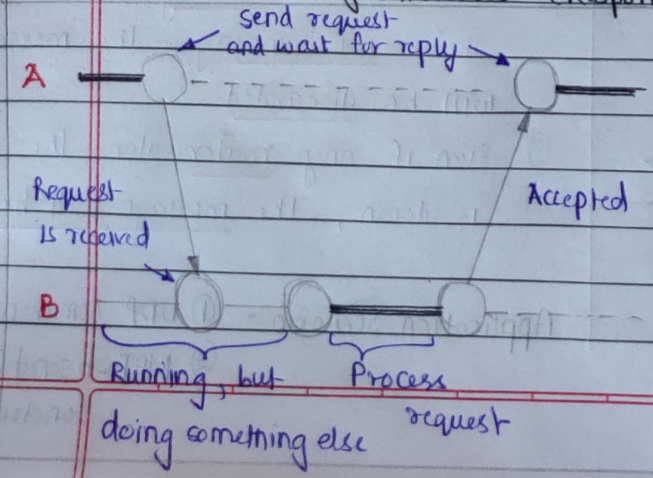
① This is an extension of receipt based transient synchronous communication.

② A will resume when B takes the delivery of the message.

③ The ACK comes a little bit later than the previous method.

④ This is essentially asynchronous RPC because from perspective of RPC we are not blocking the reply.

(VI) Transient Synchronous (Response Based)



① A resumes execution upon receiving a response.

② Not only has the message been delivered, but it has been processed and the ACK comes back in the form of a reply. the reply comes back.

③ Client is Blocked for entire duration

Application Scenario - Traditional RPC