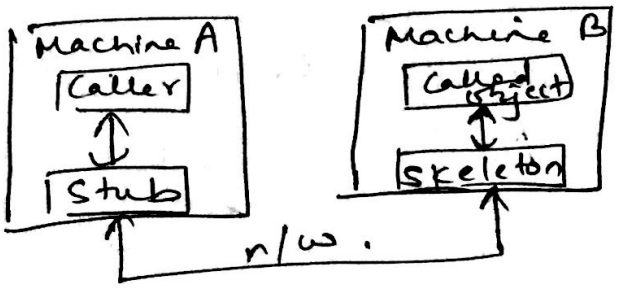
Communication b/w Distributed Objects

The communication in distributed object is done by various middleware language like RMI, CORBA (common object request broker). Invoking a method on a remote object is known as RMI or remote invocation.

→ Distributed object communication realizes communication b/w distributed objects.



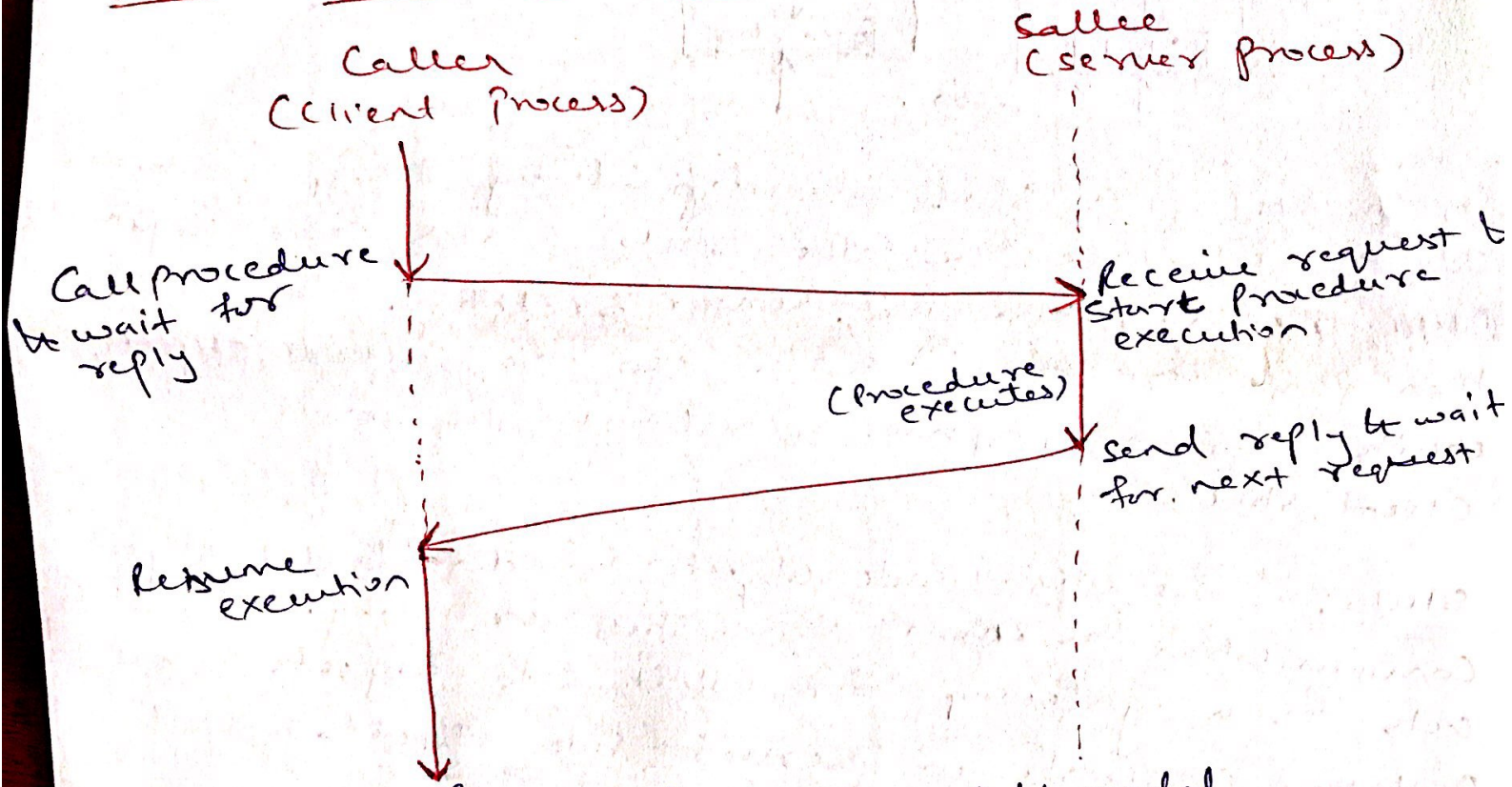
→ In RMI, a stub is defined by the programmer as an interface then the stub passes calls arguments over the n/w to the server skeleton. The skeleton then passes received data to the called object, waits for a response & returns the result to the client stub.

Stub:- The client side object in distributed object communication is known as a stub or proxy. The stub acts as a gateway for client side objects & all outgoing requests to server-side objects that routed through it.

Skeleton:- The server-side object participating in distributed object communication is known as skeleton. A skeleton acts as the gateway for server side objects & all incoming client requests are routed through it. The skeleton wraps server objec functionality & exposes it to the clients

RPC (Remote Procedure Call):- is a powerful technique for constructing distributed, client-server based applications. It is based on extending the conventional local procedure calling, so that the called procedure need not exist in the same address space as the calling procedure. the two processes may be on the same system, or they may be on different system with a n/w connecting them.

When making a Remote Procedure Call:-



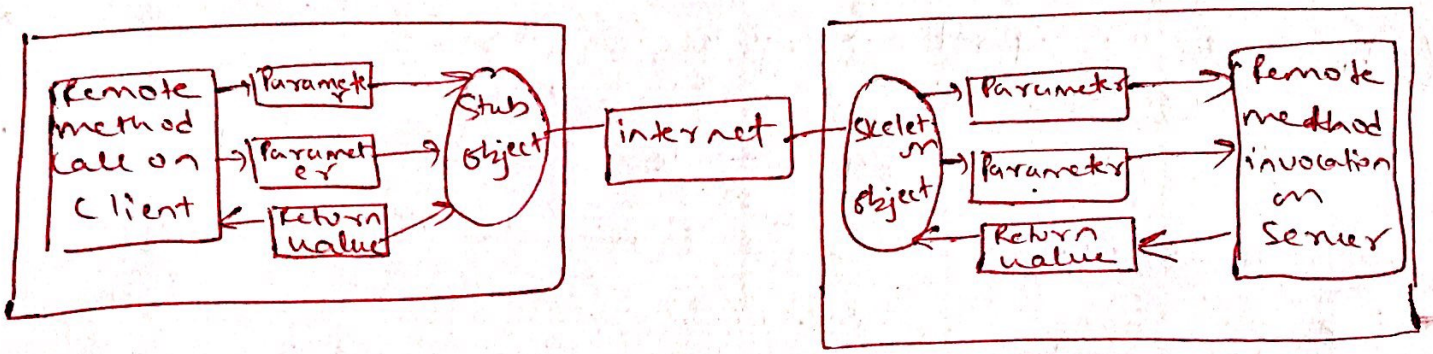
Remote procedure call model

Working of RPC

1. A client invokes a client stub procedure, passing Parameter in usual way.
2. The client stub marshalls (Pack) the Parameter into a msg.
3. The client stub passes the msg to the transport layer, which sends it to the remote server machine

Remote method invocation in Java is an API which allows an object to invoke a method on an object that exists in another address space, which could be on same machine or on a remote machine.

working of RMI



Stub Object :- The stub object on the client machine builds an info block & sends this info to the server. The block consists of

- An identifier of the remote object to be used
- Method name which is to be invoked.
- Parameters to the remote JVM.

Skeleton Object :- The skeleton object passes the request from the stub object to the remote object. It performs the following task.

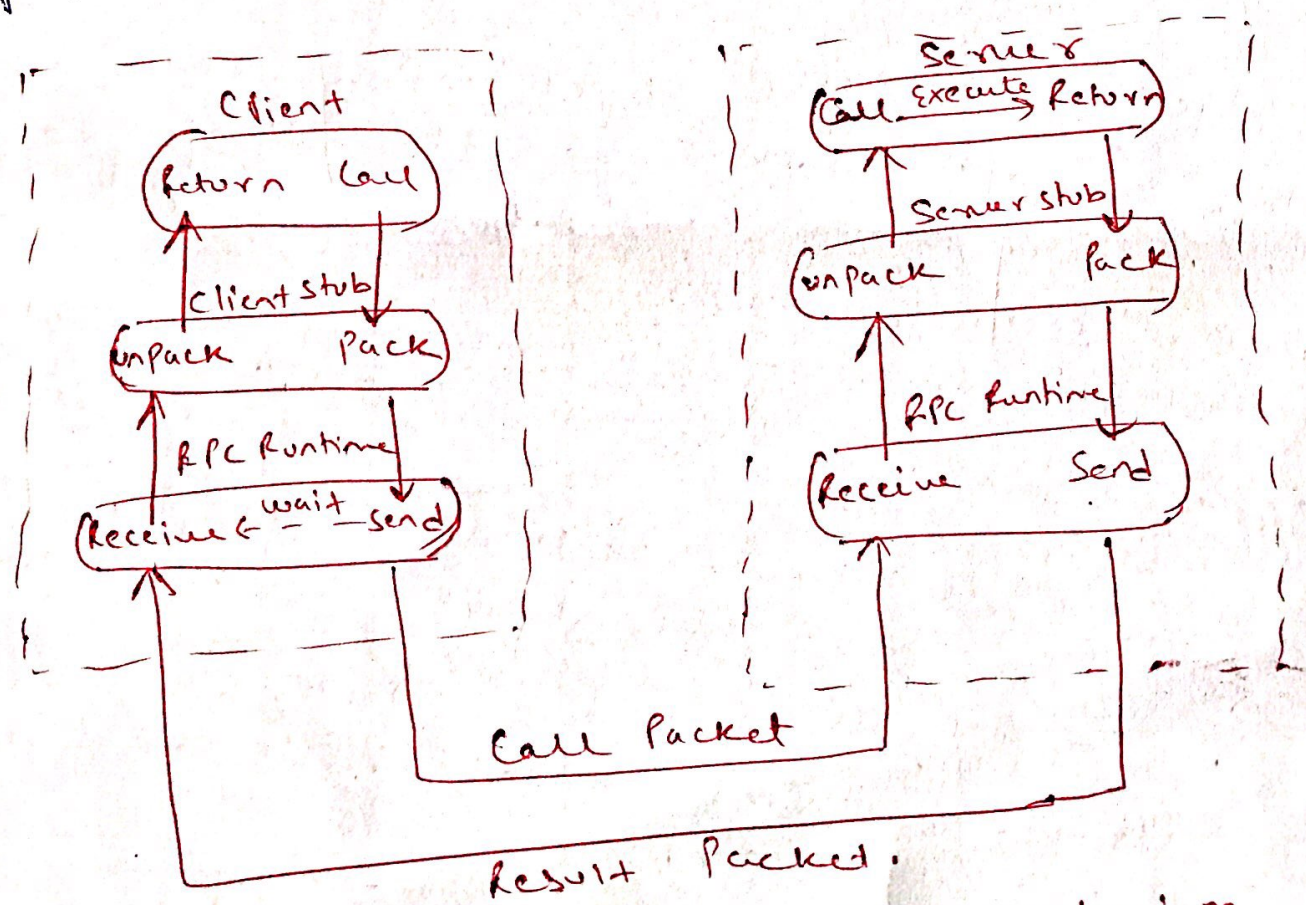
- it calls the desired method on the real object present on the server.
- it forwards the parameters received from stub object to the method.

4. On the server, the transport layer passes the message to a server stub, which demarshalls the parameters & calls the desired server routine using the regular procedure call mechanism.

5. When the server procedure completes, it returns to the server stub, which marshalls the return values into a message.

(6). The transport layer sends the result msg back to the client transport layer, which hands the msg back to the client stub.

(7) The client stub demarshalls the return parameters & execution returns to the caller.



Implementation of RPC mechanism