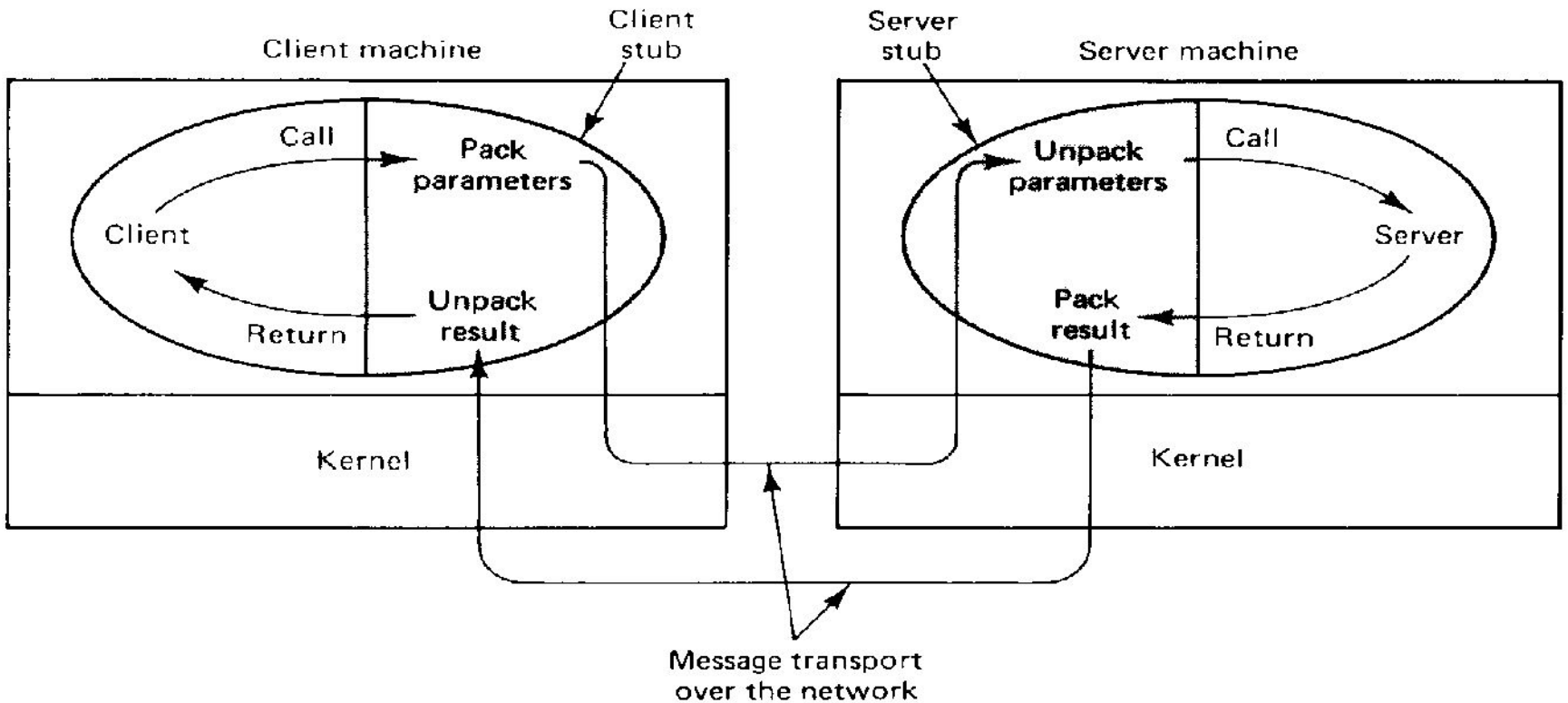
Communications in Distributed Systems

Remote Procedure Call

Remote procedure call:- Information can be transported from the caller to the callee in the parameters and can come back in the procedure result.

Calling and Called procedures run on different machines and they execute in diffrent address spaces.

RPC is the widely used approach for Distributed Operating System.

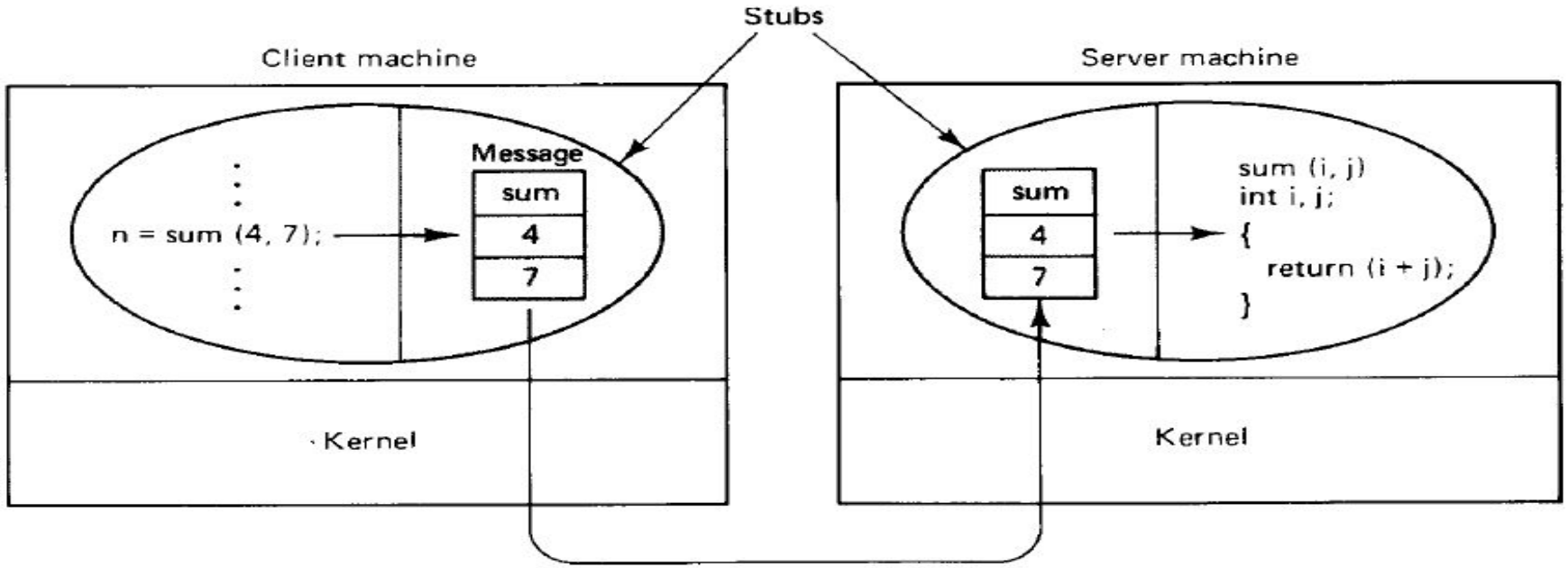

Client machine
Client stub
Server stub
Server machine
Call
Pack parameters
Client
Unpack result
Return
Unpack parameters
Call
Server
Pack result
Return
Kernel
Kernel
Message transport over the network

1. The client procedure calls the client stub in the normal way.
2. The client stub builds a message and traps to the kernel.
3. The kernel sends the message to the remote kernel.
4. The remote kernel gives the message to the server stub.
5. The server stub unpacks the parameters and calls the server.
6. The server does the work and returns the result to the stub.
7. The server stub packs it in a message and traps to the kernel.
8. The remote kernel sends the message to the client's kernel.
9. The client's kernel gives the message to the client stub.
10. The stub unpacks the result and returns to the client.

Parameter Passing

Parameter Marshaling:- Packing the parameters in the message.

The Client Stub takes the parameters and put them in a message. It also puts the number or name of the procedure to be called in the message. The server machine might support different calls.



- **Problem occurs when the system at client and server end is different.**
- Each machine has its own representation for numbers ,characters and other data items.
- IBM Mainframe machines :-EBCDIC character code.
- IBM personal Computer :- ASCII character code.

- Similar problem occurs with representation of integers and floating numbers.

- Client End:-The compiler reads the server specification and generate a client stub that packs its parameters into the officially approved message format.

- Server End:- The compiler can also produce a server stub that unpacks them and calls the server procedure.

- The system is trasparent with respect to the differences in the internal representations of the data items.

- **Dynamic Binding.**

- The client locates server in distributed syetem using Dynamic Binding.
- **Registering the Server to Binder:-**
- The server send a message to a program called a binder, to make its existance known.
- The server specifies its name, version number ,a unique identifier (32 bit long), and a handle used to locate it.
- The handle is system dependent (Ehternet Address, IP Address and X.500 Address, a sparse process identifier).
- It can deregister with the binder when it is no longer prepared to offer service.

- **How client locates server?**

- The client stub send message to the binder asking it to import version of the server interface.

- The binder checks to see if one or more servers have already exported an interface with the version and name . If no server is found the read call fails.

- Otherwise , the binder gives its handle and unique identifier to the client stub. The client stub uses the handle as the address to send the request message

Thank You!