
Sessional Assignment
Course: - Distributed Computing

Deadline: - Mentioned on SIC

Marks: - 20

Program: - MS (CS)

Dated: 15 May 2020

Student Name: Zeeshan Sajid Student ID#: 14289

Class and Section: Ms Cs 4th Semester

Question: Assume you have a Client Server Environment in which the client request the server to multiply three given number i.e 67, 90, 34, and return the result. Discuss the steps of the system in each of the following scenarios.

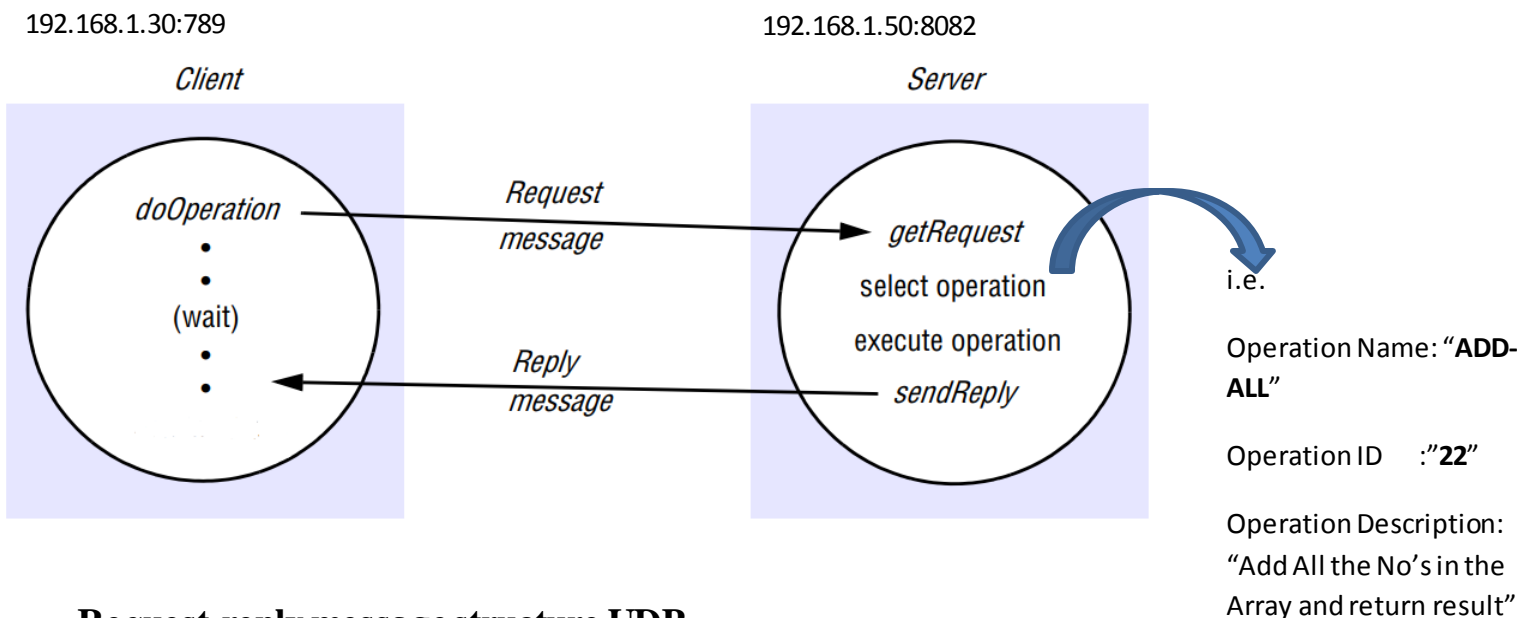
QA) How the Request-Reply Protocols functions will be used with UDP (refer to figure 5.3 in book), how will be the message identifiers used, what will be its failure model, how time outs will be used, how will the system handle duplicate messages and how will the system react if reply is lost. ?

Answer A)

RRP using UDP

sends a request message to the remote server and returns the reply. The arguments specify the remote server, the operation to be invoked and the arguments of that operation. public byte[] getRequest (); acquires a client request via the server port

the below figure shows the working of RRP using UDP



Request-reply message structure UDP

Message identifier

messageType (0)

requestId (1)

remoteReference (IP Address of server)

operationId (Operation ID ie "22")

```
public byte[] doOperation (0,1,192.168.1.50:8082, 22, [67,90,34])
```

```
public byte[] getRequest (8082);
```

```
public void sendReply ([191], 192.168.1.30,789);
```

Failure Model

For **Failure** `doOperation` uses a timeout when it is waiting to get the server's reply message. The action taken when a timeout occurs depends upon the delivery guarantees being offered.

Timeouts:

DoOperation sends the request message repeatedly until either it gets a reply or it is reasonably sure that the delay is due to lack of response from the server rather than to lost messages.

Eventually, when doOperation returns, it will indicate to the client by an exception that no result was received

Duplicate Messages

can be recognized successive messages (from the same client) with the same request identifier and to filter out duplicates. If the server has not yet sent the reply, it need take no special action – it will transmit the reply when it has finished executing the operation.

Lost Reply Message

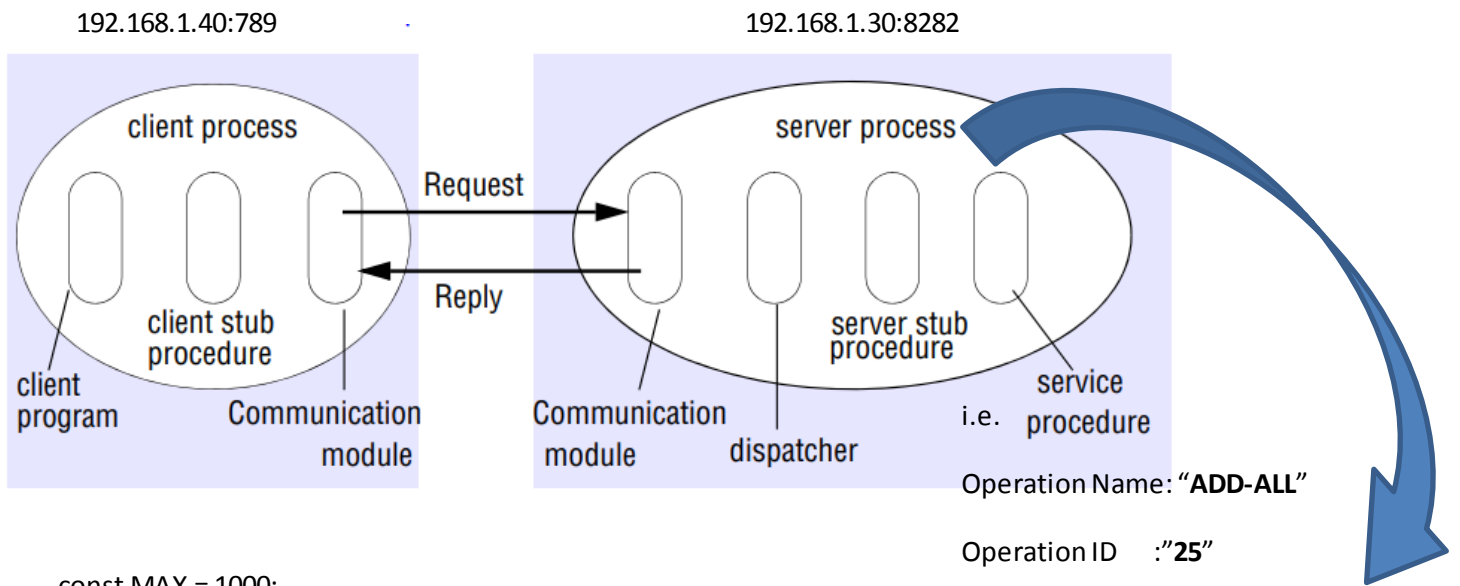
In case of Lost Reply Message when the server get duplicate request An idempotent operation is an operation that can be performed repeatedly with the same effect as if it had been performed exactly once. Or Also we can use History can be used which is retransmission of replies without re-execution of operation.

**QB) How can the above system implemented using Remote Procedure Calls (RPC)?
(Hint: Read Section 5.3.2 in the book).**

Answer B)

System implementing using RPC

In RPC the Client will convert three numbers and multiply operation into stub procedure. The stub procedure behaves like a local procedure to the client but instead of execution of the call it marshals both the procedure identifier and arguments into a request message which it sends to the server through its communication module. On the contrary when the reply message arrives it un marshals or decrypts the result. the server process contains a dispatcher with one server stub procedure and one service procedure of each single procedure in the service interface both together. The dispatcher selects one of the server stub procedure by looking at the procedure identifier in the request message. Then the server stub procedure un marshals the argument in the request message calls the corresponding service procedure and marshal return values for the reply message. The service procedure implement the procedure in the interface service.



```

const MAX = 1000;
typedef int FileIdentifier;
typedef int FilePointer;
typedef int Length;
struct Data {
int length;
char buffer[MAX];
};
struct writeargs {
FileIdentifier f;
FilePointer position;
Data data;
};
struct readargs {
FileIdentifier f;
FilePointer position;
Length length;
};
program FILEREADWRITE {
version VERSION {
void WRITE(writeargs)=1; 1
Data READ(readargs)=2; 2
}=2;
}= 9999;

```

Operation Description: "Add All the No's in the Array and return result"

QC) How can the above system implemented using Remote Method Invocation (RMI)?

(Hint: Read Section 5.4.2 in the book).

(6)

Answer C)

Remote Method Invocation:

In remote method invocation we have different objects and modules. The numbers and operations will be converted into objects. The two cooperating communication modules carry out request reply protocol which transmits request and reply messages between client and server. The communication module only uses the first three items. Message type, Request ID and the remote reference of the object invoked. The operation ID and all marshalling and unmarshalling are the concern of RMI software. The communication modules are together responsible for providing a specified invocation semantics. The communication module in the server selects the dispatcher for the class of the object to be invoked, passing on its local reference, which it gets from the remote reference module in return for the remote object identifier in the request message.

THE END