1. **Low-level communication facilities** **of computer networks do not offer distribution** **transparency. Justify this statement.**

Low level communication facilities of computer networks do not provide distribution transparency because location transparency means that the user does not need to know where the data to be operated is placed, that is, to which site or sites the data is allocated to the store is transparent to the user. But in low-level communication facilities of computer networks, user must specify the IP address and port number of the communicating party when communicating. Therefore, there is no location transparency.

Middleware system provides a higher level of abstraction, making it easier to communicate. Therefore, middleware communication helps to achieve the transparency of distribution. A special type of distributed transparency is location transparency, which aims to hide the location of objects, in this case user may not need to the specify information such as the IP address and port number.

**Two types of distribution transparency:**

Access transparency: Hide differences in data representation and how an object is

accessed.

Location transparency: Hide where an object is located.

**WebSocket version:**

WebSocket do not provide distribution transparency because it is not a type of middleware and if we focus on the definition of location transparency, which means that the user does not need to know where the data to be operated is placed, that is, to which site or sites the data is allocated to the store is transparent to the user. But in the WebSocket, user must specify the IP address and port number of the communicating party when communicating. Therefore, there is no location transparency.

Middleware system provides a higher level of abstraction, making it easier to communicate. Therefore, middleware communication helps to achieve the transparency of distribution. A special type of distributed transparency is location transparency, which aims to hide the location of objects, in this case user may not need to the specify information such as the IP address and port number.

1. **Explain how middleware can help to achieve distribution transparency.**

In computer networks, communication is based on low-level message-passing primitives,

but it is not “transparency”. What is transparency? In the simplest example, when a Chinese speaks with an American, if they talk directly, they cannot know what the other is talking about, but they only need to bring an interpreter, and the problem can be solved easily. the translation is equivalent to an additional middle layer. The middle layer is used to solve the problem of mismatch between the two sides, so that the communication between the two sides becomes transparent.

If there is only low-level communication without middleware, we can say they has no transparency, because these Low-level communication facilities of computer networks cannot “understand” each other as they have no IP addresses and port numbers for communication.

Here middleware systems offer a higher level of abstraction that make it easier to communicate. Consequently, middleware communication helps achieve distribution transparency.

1. **Describe the difference between connection-oriented and connectionless communication.**

In the connection-oriented communication, the sender and receiver need to establish a connection firstly then they can exchange data, and they also need to close the connection when they finish, such as telephone and TCP.

But in the connectionless communication, we don’ need to setup connection in advance

The sender can just send some messages when they are ready, such as dropping a letter in a mailbox and UDP.

1. **Describe the difference between persistent and transient communication.**

Persistent communication - message submitted for transmission is stored by communication system as long as it takes to deliver it to the receiver [1], and there is no guarantee that the message will arrive, or even that the message will be read, which is entirely up to the receiver

Transient communication - message stored by communication system only as long as the sender and receiver executing[2] .

1. **Describe the difference between asynchronous and synchronous communication.**

Asynchronous communication is that sender continues immediately after it has submitted its message for transmission, client don’t need to wait the return message from server but it has certain some risks. Whereas synchronous communication is that sender is blocked until its request is known to be accepted, client must wait for server, it may be slower than asynchronous but it is safe.

1. **What is the purpose of a** **Remote Procedure Call (RPC)?**

The purpose of Remote Procedure Call (RPC) is calling a procedure that is implemented on a remote machine whose parameters and results are transported but no message passing is visible to the programmer.

1. **Describe how a Remote Procedure Call (RPC) works?**

Client calls directed to a client stub which provides the local procedure call interface to the remote function. And a server stub transforms calls coming in over the network into local calls.

The steps:

1. Client call to procedure

2. Stub builds message

3. Message is sent across the network

4. Server OS hands message to server stub

5. Stub unpacks message

6. Stub makes local call to procedure.

1. **Using a Remote Procedure Call (RPC) presents some challenges. Describe two such challenges**

Two challenges:

1.Handling different data representations such as little endian and big endian, which makes our program so difficult to interact with someone else's program.

2.Handling pointers or references which refer to local memory and fail to call a remote procedure.

1. **What is the purpose of a** **Remote Method Invocation (RMI)?**

The purpose of Remote Method Invocation is same as RPC: to call a method from the remote machine ,but MPI is for a object-oriented language, such as Java RMI.

1. **Describe two reasons why one might use a Message Oriented Middleware (MOM) for communication rather than a Remote Procedure Call (RPC) or Remote Method Invocation (RMI).**

There are two reasons:

1. Message Oriented Middleware achieves persistent but RPC and RMI not. RPC and RMI require client and server to be running at time of communication, whereas Message Oriented Middleware offers intermediate storage for messages, without requiring sender or receiver to be active during transmission [1].
2. Message Oriented Middleware achieves asynchronous but RPC and RMI not. RPC and RMI block the client until its request has been processed, whereas Sender is given only the guarantees that its message will eventually be inserted in the recipient’s queue, No guarantees are given about when, or even if the message will actually be read[1]. The asynchronous method could save much time.
3. **What is the purpose of multicast communication in a distributed system?**

The purpose of multicast communication is sending data to multiple receivers.

1. **With the aid of an example, describe how application-level tree-based multicasting works.**

How it works:

Nodes organize as overlay network with tree topology; unique (overlay) path between every pair of nodes. Each node forwards a message m to each of its neighbors, except to the one from which it received.

Example:

A message multicast by A will traverse the links (B, Rb), (Ra, Rb), (E, Re), (Rc, Rd), and (D, Rd) twice like the diagram below: 