

Middleware Enterprise and Integration Technologies

Q.No.1. Solve any five

- a. Explain CORBA IDL , Write a program for CORBA IDL and server class to calculate interest: Input parameters- Principle amount, rate and no of yrs.\Out Put parameters- interest value 10

Ans: Explanation of CORBA IDL

The CORBA specification define interface and allow interface to define in terms of operation signature, As the IDLs are Declarations, they have no control constructs.

Some characteristics of IDLS.

- Multiple language support.
- Platform independent.
- Implementation independence.
- Increase reused.

2. PROGRAM for CORBA IDL

```
import org.omg.CORBA.*;
import org.omg.PortableServer.POA;

class ShapeListServant extends ShapeListPOA {
private POA theRootpoa;
private Shape theList[];
private int version;
private static int n=0;

public ShapeListServant(POA rootpoa){
theRootpoa = rootpoa;
// initialize the other instance variables
}

public Shape newShape(GraphicalObject g) throws ShapeListPackage.FullException {1
version++; Shape s = null;
ShapeServant shapeRef = new ShapeServant( g, version);
try {
```

Middleware Enterprise and Integration Technologies

```
org.omg.CORBA.Object ref = theRoopoa.servant_to_reference(shapeRef); 2
```

```
s = ShapeHelper.narrow(ref);
```

```
} catch (Exception e) {}
```

```
if(n >=100) throw new ShapeListPackage.FullException();
```

```
theList[n++] = s;
```

```
return s;
```

```
}
```

```
public Shape[] allShapes(){ ... }
```

```
public int getVersion() { ... }
```

```
}
```

Q.No.1. b. Compare RMI, CORBA and DCOM.

10

Ans:

Openness Characteristics	CORBA	RMI	DCOM
Interface	Specified using OMG IDL	Specified using java interface	Specified using MIDL
Operations interface	Default	Default	Default
Signal interface	Realized using CORBA event service.	Realized using JMS	DCOM event service
Stream interface	Not supported	Not supported	Not supported
Interface inheritance	Available	Available	Available
Implementations inheritances	Programming language dependant	Available	Programming language dependent
Object clusters	Managed using POA	Managed using activation group	Object clusters called context are organized

Middleware Enterprise and Integration Technologies

		object	based on properties
Capsules	CORBA servers	Server	Processes
channels	Based on GIOP	Based on JRMP	DCE RPC

Q.No

2. a. Explain the structure of WSDL, document and its constituents elements with a diagram.

10

Ans:

1. Definition:

WSDL (web services descriptions language) describes in XML format how to access a service. WSDL is used to communicate the service interface to developers who use it for service.

WSDL supports run time behavior of complex service:

1. Service discovery:
2. Service invocation:

WSDL it has following components:

- 1) Type element:
- 2) Message element:
- 3) PortType element:
- 4) Binding element:
- 5) Service element;
- 6) Port:

Q.No. 2. b. Elaborately explain the .NET framework remoting architecture with a labeled diagram.

10

Ans:

1. Introduction to .NET:

To face the challenges faced by the internet revolution, Microsoft tried to expand Microsoft foundation calls(MFC) to include networking classes. Using these classes, a programmer can develop a distributed applications.

The .NET technologies it has following objectives:

- To provide an object-oriented programming environment transparent to the location of the server code.
- To provide an IDE that minimizes software deployment and versioning conflicts.
- To provide an efficient code integration environment for off-the-shelf components.
- To facilities application transparency.
- To ensure applications interoperability with other standards.

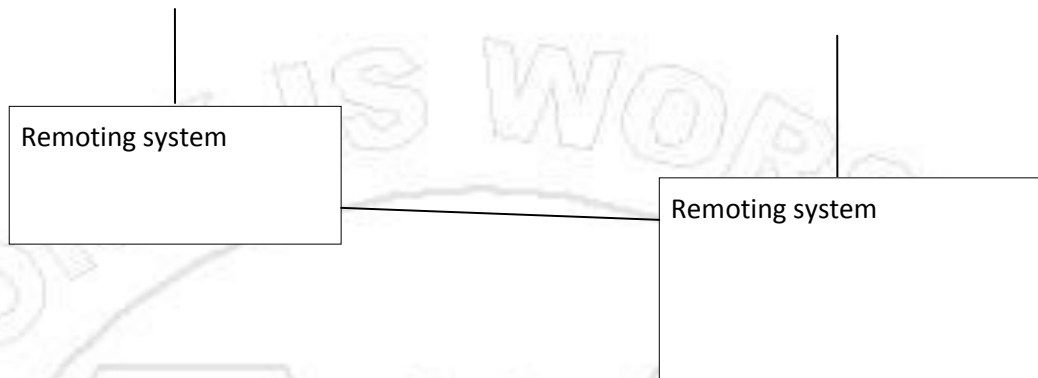
2.Remoting Architecture:

The .NET remoting is a generic system for communication between different applications. The applications can be located on the same computer, different computer on the same network or separate networks.

Diagram:



Middleware Enterprise and Integration Technologies



The steps in .NET remoting process are listed as follows.

1..NET Remoting makes remote object available to the client through activation URL.

The client creates a new instance of the server class and uses it like a local object.

2.The Remoting system creates a proxy object that represents the class and returns to the client object a reference to the proxy.

3. When the client calls a method, the Remoting infrastructure handles the call checks the type information, and sends the call over the channel to the server process.

4. A listening channel picks up the request and forward it to the server Remoting system, which locates and calls the required object.

Q.No. 3. a. Illustrate and describe the different components of EJB framework. 10

Ans:

1. Description of EJB framework

EJB is a server side component. To get the services hosted by the EJB ,the client has to communicate with the EJB component .

2. Different steps in application using EJB framework:

- 1) Creation of home interface:
- 2) Creation of remote interface:
- 3) Creations of primary key class :
- 4) Creation of an enterprise bean class:

Middleware Enterprise and Integration Technologies

5) Creation of a deployment descriptor:

Q.No. 3. *b. Compare POA and BOA in CORBA. 10*

Ans:

1. Portable Object Adapter (POA) - The CORBA object responsible for splitting the server side remote invocation handler into the remote Object and its Servant. The object is exposed for the remote invocations, while the servant contains the methods that are actually handling the requests. The Servant for each object can be chosen either statically (once) or dynamically (for each remote invocation), in both cases allowing the call forwarding to another server

2. Creating and Using the POA

The steps for creating and using a POA will vary according to the specific application being developed. The following steps generally occur during the POA life cycle:

- Get the root POA
- Define the POA's policies
- Create the POA
- Activate the POA Manager
- Activate the servants, which may include activating the Tie
- Create the object reference from the POA

3. Basic Object Adaptors:

The first version of the CORBA specification contained a description of the Basic Object Adaptor, The BOA was intended to be a multi-purpose OA that could support various styles of servants.

The BOA supports four different activation models, described below.

- 1) Unshared server:
- 2) Shared server:
- 3) Persistent server:
- 4) Server-per-operation:

Q.No. 4. *a. Describe SOAP messages, How are they processed. 10*

Ans:

Middleware Enterprise and Integration Technologies

1. SOAP (simple object access protocol) is an XML specification for transmitting data(protocol) to and from a web service. SOAP file can also be generated automatically with a tool which includes details from WSDL definitions. It is a widely used message format because of the following reasons:

1. It is relatively simple, built on top of existing technologies such as HTTP.
2. It provides an extensible, compatible framework that allows solutions to be incrementally applied as needed.

2.SOAP format:

The skeleton of SOAP message is as follows.

```
<env: Envelop smlns:env="http:// www.w3.org/2003/05/soap-envelope">
<env:Header>
....
</env:Header>
<env:Body>
.....
</env:Body>
</env: Envelop>
```

There are three major elements that appear in the structure of a SOAP message.

1. <Envelop>
2. <Header>
3. SOAP body.

Q.No. 4. **Differentiate java beans and Enterprises java beans.** 10

Ans: a) Java Beans:

Java beans are a reusable component that can be used in java programming, for example:

Middleware Enterprise and Integration Technologies

Package util it contain all utility packages for programming

b) Enterprise java beans:

The enterprise java beans specification defines a standard architecture for implementing the business logic for multi-tier applications as reusable components.

Logical architecture:

- The client
- The EJB server.
- The database.

Q.No. 5. a. *Mention and explain the different features of distributed computing environment.* 10

Ans:

1. Distributed computing:

In a heterogeneous distributed environment, a common set of distributed services should be available to the applications. The Open Software Foundations’ distributed computing environment (OSF’s DCE) is a rich software technology that enables the development of distributed applications across heterogeneous systems

Applications				
Time Service	CDS	Service	DFS	Diskless support
Remote procedure call				

Middleware Enterprise and Integration Technologies

Thread service
Local operating system and Transport service

DCE Architecture:

2. The different service of DCE model:

- Directory service:
- The global directory service (GDS):
- The global directory agent(GDA):
- A Directory service programming interface:

Q.No. 5. *b. What is a web service? Explain the different types of web services. 10*

Ans:

1. Definition of web services:

A web service is any piece of software that makes itself available over the internet and uses open standard (XML,SOAP) based web applications that interact with each other for the purpose of exchanging data.

Web services are loosely coupled (platform independent), contracted components that

2. Different types of web services:

- Service Transition plan(STP):
- Service –Oriented Maturity:
- Choreography of Services:
- Granularity of Services:

Middleware Enterprise and Integration Technologies

Q.No. 6. a. Create a XML schema monitors.xsd for the given XML document. Each monitor has a unique modelNo and up to 3 features. The value of the attribute type can only be "LED"

, "LCD" or "CRT".

10

Ans:

An XML Schema:

- Define elements that can appear in a document
- Define attributes that can appear in a document
- Define which element are child element
- Define the order of child elements
- Define the number of child element
- Define whether an element is empty or can include text
- Define data types for elements and attributes
- Define default and fixed values for elements and attributes.

Complex type:

```
<xs:complexType>
<xs:sequence>
<xs:element name="mno" type="xs:string"/>
<xs:element name="por" type="xs:string"/>
.....
.....
</xs:sequence>
</xs:complexType>
```

Q.No. 6. b Explain with neat labeled diagram the working of connection oriented protocol using socket.

10

Ans:

1. Definition:

Socket: An interface between an application process and transport layer

Middleware Enterprise and Integration Technologies

_ The application process can send/receive messages to/from another application process (local or remote) via a socket

_ In Unix jargon, a socket is a file descriptor . an integer associated with an open file

_ Types of Sockets: Internet Sockets, unix sockets,

X.25 sockets etc

_ Internet sockets characterized by IP Address (4 bytes) and port number (2 bytes)

Types of socket:

- a. SOCK-STREAM:
- b. SOCK_DGRAM:
- c. SOCK_RAW:

Following steps:

1. socket:
2. bind
3. listen
4. accept
5. connect
6. write
7. process
8. read

Q.No. 7.

Write short note on: (any two):

20

- .NET Architecture.
- CORBA services.
- DCOM apartments.
- Differentiate Remote Procedure Method and Message Oriented Messaging.

Ans:

1. Introduction to .NET:

To face the challenges faced by the internet revolution, Microsoft tried to expand Microsoft foundation calls(MFC) to include networking classes. Using these classes, a programmer can develop a distributed applications.

The .NET technologies it has following objectives:

1. To provide an object-oriented programming environment transparent to the location of the server code. 2. To provide an IDE that minimizes software deployment and versioning conflicts.

Middleware Enterprise and Integration Technologies

3. To provide an efficient code integration environment for off-the-shelf components.
4. To facilities application transparency.
5. To ensure applications interoperability with other standards.

2. CORBA services.

CORBA defines stand interface and allows users to implement them in different ways, However there are generic services that are required in all application domains.

COS services are as follows:

1. Information management services:
2. Infrastructure services:
3. Task management services:

CORBA services:

- Naming service:
- Trading service:
- Event Service
- Security service:
- Lifecycle service:
- Licensing service:
- Relationship service:
- Externalization service:
- Persistent object service:
- Time service:
- Concurrency control service:
- Object transition service:
- Object query service:
- Object collections services:
- Object property service:

4. DCOM apartments.

DCOM also called "COM" on the wire support remoting objects by using object remote procedure call protocol. This ORPC layer is built on top of DCE's RPC and interact with COM's run-time services.

A DCOM server object can support multiple interfaces each representing different behaviors of services.

The DCOM architecture consists of three layers, and it extends the COM in the following ways:

1. Binary interoperability
 2. Packaging transparency
 3. Free-threading model
 4. Security
 5. Reference counting and ping
 6. Administration.
5. Differentiate Remote Procedure Method and Message Oriented Messaging.

Middleware Enterprise and Integration Technologies

Feature	RPC	Advantages	MOM
Client/server model	Unlimited clients and servers with multiple brokers running anywhere in the network	RPC it supports heterogeneous brokers	Multiple clients with multiple servers restricted to limited number of queues on pre-determined servers in the network.
Client/server relationship	Synchronous with asynchronous possible,	RPC has better performance when compared to MOM	Asynchronous clients and servers may operate at different times and speeds, response to client is not guaranteed.
Platform support	Unlimited; every vendor in the world supports RPC	RPC has wide support	Limited platform support for MOMs
Communication	Listen, connect, accept it is analogous to telephone communication.	Not applicable	Post a message then wait .It is analogous to postal communication.
Administration and monitoring	Many different commercially available monitors.	RPC has better control	None
Scalability	Easily done by starting additional servers based on load	RPC	Very difficult to scale because of the inability to distribute queues dynamically.
Load-balancing	Can be accomplished with performance monitors and application management systems.	RPC has monitor to administer load balancing	Require a monitor to manage more than a single queue. Since queue can be used to implement FIFO or priority based policy.
Transactions support	Transactions supported with RPC	RPC	Limited to some products

Middleware Enterprise and Integration Technologies

Message filtering	Not available	MOM	Available
Performance	Very fast	RPC as less number of message copies occur.	Very slow as a intermediate hop is required to store and retrieve message in the queue
Asynchronous processing	Yes ; easily implemented with callback.	None	Yes queue and triggers are required.