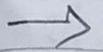


Assignment : 3



Q.1)

What is the role of J2EE in distributed computing?



A distributed system is a collection of individual systems that coordinate and communicate with each other by sharing information over the network. It works on client/server or peer-to-peer model.

A distributed application uses the layered approach to software development using middleware.

Middleware provides a common set of services for applications and has to work in heterogeneous environment.

Here the use of J2EE helps the developers to develop platform independent tools.

J2EE Development Tools:

Using following J2EE development tools for analysis, design, and development leads to the proper configuration of distributed system.

1. Analysis and Design tool: Use of UML diagrams help in visualizing the system. The tools like Rational Rose is used.

2. Development Tool: The development IDE tools speed the development. It results in faster performance of the system.

The most commercial IDEs available on J2EE projects are:

WebGain studio

Borland's JBuilder and

IBM's visual Age.

WebGain studio is a complete J2EE development environment.

On open source server like JBoss or Enhydra, the Ant build tool is used. Ant integration is available for visual age, JBuilder, and the NetBeans open source IDE.

3) Build Tool: once the project is developed it has to be built for execution and deployment. As the system works in different environments the deployments tools are required. J2EE have its own build tools to support this phase of project development. This process involves compiling the components to their relative classes, creating, deployment descriptors, and packaging the components into JAR, WAR, or EAR files. This all process is very complicated and the use of built in tools support in this.

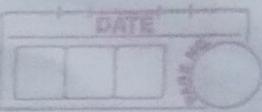
4) Source-code control Tool: J2EE maintains a shared repository for the code base in various phases of project development. Different developers in a team have their own codes timely updated. This should be very well coordinated with other developers. The repository helps to assemble different versions of the codes centrally.

5) Testing Tool(s): Before any project is released a vigorous testing is done. The various utilities perform various types of testing on project components. There are different types of tests performed on the code like:

- i) unit / functional testing.
- ii) system testing.
- iii) Integration testing
- iv) unit / functional testing.
- v) user acceptance testing.

6) Problem Tracking Tools: This tool takes care of integrating modules to construct the overall system.

7) Testing and Deployment in J2EE: J2EE applications are difficult to test and deploy. The difficulty levels occur because of distributed nature of the



system.

Q.2) Explain use of XML in distributed computing?
→ XML is the extensible Markup Language, which allows multiple languages to come together and make the information base. XML provides the basis for a wide variety languages. Examples include mathematical markup Language (MathML), Electronic Business XML (ebXML), and Voice Markup Language (VXML). XML consists of both markup and content. Markup referred to the tags that describes the content in the document. The flexible representation of data allows to easily send and receive data, and transform data from one format to another. Some specialized uses of XML are the Java speech Markup Language and the synchronized multimedia Integration language.

Each XML language has its own grammar and that specific set of rules governing the content and structure of documents written in that language. An XML-based work enables high levels of component reuse and interoperability in the distributed system.

Q.3) What is service Oriented Architecture? Explain key characteristic of soA.

→ Service-oriented architecture (SoA) is a method of software development that uses software components called services to create business application. Each service provides a business capability & services can also communicate with each other across platforms & languages. Developers use SoA to reuse services in different systems or combine several independent service to perform complex tasks.

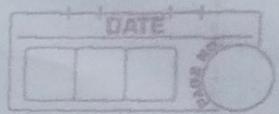
* Key characteristics of SOA:

- i) SOA services use XML and web services Description Language (WSDL) as the standard to describe the services. These are platform independent standards and so they help to distribute the application over Web.
- ii) SOA services communicate with messages formally defined via XML also called XSD. Communication among clients and service providers or services is always in the heterogeneous environments.
- iii) SOA services are maintained in the enterprise by a registry that acts as a directory listing. Applications can look up the services in the registry and invoke the service. Universal Description, Definition, and Integration (UDDI) is the standard used for service registry.
- iv) Each SOA service has a quality of service (QoS) associated with it. Its important elements are security requirements, such as authentication and authorization, reliable messaging, and policies regarding who can invoke services.
- v) With SOA, the hardware, operating systems and softwares can be used as a service.

Q.4) What is stateless & stateful services explain with examples.

→ Stateless:

A stateless architecture or application is a type of Internet protocol where the state of the previous transaction is neither stored nor referenced in subsequent transactions. Each request sent between the sender & receiver can be interpreted & does not need either requests for its execution.



This is a protocol where a client & server requests & response are made in a current state. In addition, the status of the current session is not retained or carried over the next transaction. Stateless applications manage short term requests using print servers & a Content Delivery Network (CDN). An excellent example of stateless protocol work is in the sending of an SMS.

Ex: HTTP, DNS.

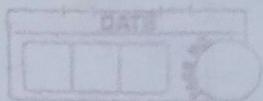
stateful :

Stateful architecture or application describes a structure that allows users to store, record & return to already established information & processes over the internet. It entails transactions that are performed using past transactions as reference point. In stateful applications, the current transactions can be affected by the previous ones.

A stateful application maintains the state of every session irrespective of the importance. Stateful architecture is used as a foundation for several existing technologies today. The file transfer protocol & the telnet were good examples of stateful architecture. The key advantages of the stateful concept

- Examples of the stateless protocol are UDP (User Datagram protocol), HTTP.

- Examples of the stateful protocol are telnet, FTP (File transfer protocol).



Q.5) Explain any one web technology in details used for implementing web services.

→ The web services use XML, HTTP. All the standard web services work using the following components.

i) SOAP (Simple Object Access Protocol): It is an XML-based communication protocol to exchange information between computers.

Different applications can communicate using this protocol.

It frames format for sending messages.

It is platform as well as language independent protocol.

ii) UDDI (Universal Description, Discovery and Integration):

UDDI presents specification for a distributed registry of web services.

It is platform independent, open framework.

UDDI uses SOAP, CORBA, and Java RMI protocol for its communication..

UDDI uses WSDL to describe interfaces to web services.

It is an open industry initiative that enables business to discover each other and define how they interact using the Internet.

iii) WSDL (Web Services Description Language):

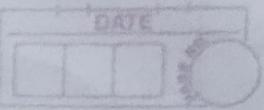
WSDL is joint development of Microsoft and IBM.

It is an XML based protocol to exchange information in decentralized and distributed environments.

It is the standard format for describing a web service.

It describes how to access a web service and what operations it performs.

Using this language one can understand how to interface with XML-based services.



It is an integral part of UDDI, an XML-based worldwide business registry.
WSDL is the language that UDDI uses.

Q.6) What are RPCs?

→ Remote procedure call (RPC) is a software communication protocol that one program can use to request a service from a program located in another computer on a network without having to understand the network details. RPC is used to call other processes on the remote systems like a local system. A procedure call is also sometimes known as a function call or a subroutine call.

RPC uses the client-server model. The requesting program is a client, and the service-providing program is the server. Like a local procedure call, an RPC is a synchronous operation requiring the requesting program to be suspended until the results of the remote procedure are returned.

However, the use of lightweight processes or threads that share the same address space enables multiple RPCs to be performed concurrently.

The interface definition language (IDL) – the specification language used to describe a software component's application programming interface (API) – is commonly used in Remote Procedure Call software. In this case, IDL provides a bridge between the machines at either end of the link that may be using different operating systems (oses) and computer languages.

Q.7) What are the features of SOAP?

-
- SOAP is a communication protocol and it is used for communication between applications.
 - SOAP is a format for sending messages.
 - SOAP communicates through Internet.
 - SOAP communicates through Internet.

Independence :- SOAP allows for any programming model.

- SOAP is platform independent and language independent. That is SOAP can be used in any languages.
- SOAP is based on XML.

Extensibility :- Security and WS-routing are among the extensions under development. SOAP is simple and extensible.

- Neutrality :- SOAP can be used over any transport protocol such as HTTP, SMTP, TCP, or JMS. SOAP allows you to get around firewalls.
- SOAP as a lightweight protocol :- It permits applications to pass messages and data back and forth between disparate systems in a distributed environment enabling remote method invocation. It means SOAP protocol possesses only two fundamental properties. They are,

- 1) send and receive HTTP transport protocol packets
- 2) process XML messages. This can be contrasted with the heavyweight protocols such as ORPC protocols.

- SOAP is a W3C recommendation.