

UNIT III – SERVICE ORIENTED ARCHITECTURE**1. What is Service Oriented Architecture?**

Service oriented architecture is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity.

2. Define Contemporary SOA.

Contemporary SOA represents an architecture that promotes service orientation through the use of web services.

Contemporary SOA represents an open, agile, extensible, federated, composable architecture comprised of autonomous, QoS-capable, vendor diverse, interoperable, discoverable, and potentially reusable services, implemented as Web services.

3. List out some characteristics of Contemporary SOA.

Some of the characteristics of contemporary SOA are:-

- i. Contemporary SOA is at the core of the service oriented platform.
- ii. Contemporary SOA increases quality of service.
- iii. Contemporary SOA is fundamentally autonomous.
- iv. Contemporary SOA is based on open standards.
- v. Contemporary SOA supports vendor diversity.
- vi. Contemporary SOA fosters intrinsic interoperability.
- vii. Contemporary SOA promotes discovery.
- viii. Contemporary SOA promotes federation.
- ix. Contemporary SOA promotes architectural composability.
- x. Contemporary SOA fosters inherent reusability.

4. What are the benefits of SOA?

The benefits of SOA are:

- i. Improved integration and intrinsic interoperability
- ii. Inherent reuse
- iii. Streamlined architectures and solutions
- iv. Leveraging the legacy investment
- v. Establishing standardized XML data representation
- vi. Focused investment on communications infrastructure
- vii. “Best-of-breed” alternatives
- viii. Organizational agility

5. What are the common pitfalls of adopting SOA?

The common pitfalls of adopting SOA are:

- i. Building service oriented architectures like traditional distributed architectures
- ii. Not standardizing SOA
- iii. Not creating a transition plan
- iv. Not starting with an XML foundation architecture
- v. Not understanding SOA performance requirements

- vi. Not understanding web services security
- vii. Not keeping in touch with product platforms and standards development

6. What are the requirements is needed to fill QoS gaps between contemporary and Primitive SOA?

Contemporary SOA is striving to fill the QoS gaps of the primitive SOA model with the following requirements:

- i. Security (contents and access)
- ii. Reliability (message guaranteed delivery)
- iii. Appropriate performance
- iv. Protecting business integrity
- v. Executing exception logic in case of failure

7. What is Autonomous Principle?

Autonomous Principle represents the ability of a service to carry out its logic independently of outside influences.

8. List out the different levels of Autonomy.

Different levels of Autonomy are:

- i. Runtime autonomy
- ii. Design time autonomy

9. What is Runtime Autonomy?

Runtime Autonomy represents the amount of control a service has over its execution environment at runtime.

10. What is Design time autonomy?

Design time autonomy represents the amount of governance control a service owner has over the service design.

11. Expand UDDI.

UDDI stands for Universal Description Discovery and Integration.

12. What are the design characteristics required to facilitate interoperability in contemporary SOA?

The design characteristics required to facilitate interoperability are:

- i. Standardization
- ii. Scalability
- iii. Behavioral predictability
- iv. Reliability

13. How is loose coupling concept achieved in SOA?

The loose coupling concept is achieved by implementing standardized service abstraction layers when service orientation principles are applied to both business modeling and technical design.

14. What is referred as Organizational Agility?

Organizational Agility refers to efficiency with which an organization can respond to change.

15. What is Architecture?

Architecture refers a systematic arrangement of computerized automation technological solutions.

16. What is application architecture?

Application architecture is a template for all others which specifically explained the technology, boundaries, rules, limitations, and design characteristics that apply to all solutions based on this template.

17. What is enterprise architecture?

Enterprise architecture is a creation of master specification when numerous, disparate and integrate application architectures exist within an organization.

18. What is Single-tier client-server architecture?

Single-tier client-server architecture is an environment in which bulky mainframe back-ends server served the thin clients.

19. List out the primary characteristics of the two tier client server architecture?

The primary characteristics of the two tier client server architectures is given below which is compared to SOA

- i. Application logic
- ii. Application processing
- iii. Technology
- iv. Security
- v. Administration

20. What is multi-tier client-server architectures?

Multi-tier architecture (often referred to as n-tier architecture) is a client-server architecture in which the presentation, the application processing, and the data management are logically separate processes.

21. List out the types of communications of mainframe systems?

The different types of communications of mainframe systems are:

- i. Synchronous communication
- ii. Asynchronous communication

22. Define synchronous communication.

Synchronous communication allows the client and server to wait for each other to transfer the message. That is, the client will not continue until the server has received the message.

23. Define asynchronous communication.

Asynchronous communication allows the server to continuously receive messages from the client without waiting for the server to respond.

24. List out the types of service autonomy?

The different types of service autonomy are:

- i. Service-level autonomy
- ii. Pure autonomy

- Service-level autonomy Service boundaries are distinct from each other, but the service may share underlying resources. For example, a wrapper service that encapsulates a legacy environment that also is used independently from the service has service-level autonomy. It governs the legacy system but also shares resources with other legacy clients.

- Pure autonomy The underlying logic is under complete control and ownership of the service. This is typically the case when the underlying logic is built from the ground up in support of the service.

25. What are the key benefits of service reuse?

The key benefits of service reuse are:

- i. Accommodate future requirements with less development effort
- ii. Reduce the need for creating wrapper services
- iii. Reduction of cost by not just avoiding duplication of code
- iv. Reducing risks by reusing well-tested code and runtime environments

26. State Separation of concerns.

“Separation of concerns” is an established software engineering theory based on the idea of breaking down a large problem into a series of individual concerns.

27. What are the parts of automation logic?

The four identified parts of automation logic related to different sized units of logic as follows:

- i. messages = units of communication
- ii. operations = units of work
- iii. services = units of processing logic (collections of units of work)
- iv. processes = units of automation logic (coordinated aggregation units of work)

28. What are the issues that are raised in the client-server and the distributed Internet architecture?

The issues that are raised in the client-server and the distributed Internet architecture comparisons are discussed in a comparison between multi-tier client-server and SOA.

- i. Application logic
- ii. Application processing
- iii. Technology
- iv. Security
- v. Administration

29. What is the use of RPC?

Client-server remote procedure call (RPC) connection is used for remote communication between components residing on client workstations and servers.

30. Write down the advantage of RPC?

Advantages of RPC are:

- i. Better load balancing:
More evenly distributed processing (e.g., application logic distributed between several servers)
- ii. More scalable:
Only servers experiencing high demand need be upgraded
- iii. Multiple concurrent requests are processed

31. Write down the disadvantages of RPC?

Disadvantages of RPC are:

In heavily loaded network

- i. More distributed processing necessitates more data exchanges
- ii. Difficult to program and test due to increased complexity

32. What is the difference between services and components?

Services are logical grouping of components to achieve business functionality. Components are implementation approaches to make a service.

33. Define supplementary definition of SOA?

Supplementary definition that can be applied to both primitive and contemporary SOA. SOA is a form of technology architecture that adheres to the principles of service-orientation. When realized through the Web services technology platform, SOA establishes the potential to support and promote these principles throughout the business process and automation domains of an enterprise.

34. Write down the layers of abstraction identified for SOA.

The three layers of abstraction identified for SOA are:

- i. the application service layer

- ii. the business service layer
- iii. the orchestration service layer

35. List some of the characteristics of Application Service layer.

- i. Expose functionality within a specific processing context
- ii. Draw upon available resources within a given platform
- iii. Solution – agnostic
- iv. Generic and reusable
- v. Achieve point-to-point integration with other application services
- vi. Inconsistent in terms of the interface granularity they expose
- vii. Mixture of custom-developed and third-party purchased services

Part B (16 Marks)

- 1. Explain briefly about Characteristics of SOA
- 2. Explain briefly about principles of SOA
- 3. Explain briefly about Client / Server architecture
- 4. Explain briefly about Distributed Internet architecture
- 5. Explain in detail about the 3 layers of abstraction