

S90.09^{Q&As}

SOA Design & Architecture Lab

Pass SOA S90.09 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.leads4pass.com/s90-09.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by SOA Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



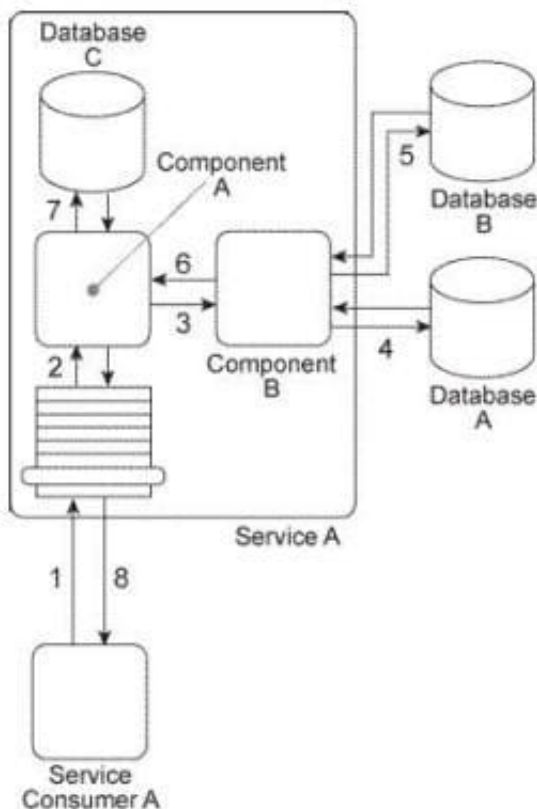
QUESTION 1

Service Consumer A sends Service A a message containing a business document (1). The business document is received by Component A, which keeps the business document in memory and forwards a copy to Component B (3). Component B first writes portions of the business document to Database A (4).

Component B writes the entire business document to Database B and then uses some of the data values from the business document as query parameters to retrieve new data from Database B (5).

Next, Component B returns the new data back to Component A (6), which merges it together with the original business document it has been keeping in memory and then writes the combined data to Database C (7). The Service A service capability invoked by Service Consumer A requires a synchronous request-response data exchange. Therefore, based on the outcome of the last database update, Service A returns a message with a success or failure code back to Service Consumer A (8).

Databases A and B are shared and Database C is dedicated to the Service A service architecture.



There are several problems with this architecture: First, the response time of Database A is often poor, resulting in Component B taking too much time to provide a response to Component A. This results in Component A consuming too many runtime resources while it holds the business document in memory and it also causes unreasonable delays in responding to Service Consumer A. Additionally, Database B is being replaced with a different database product that supports a proprietary file format. This will disable the current interaction between Component B and the new Database B. What steps can be taken to solve these problems?

A. The State Repository pattern is applied so that Component A can defer the business document data to a state database while it waits for a response from Component B. The Service Data Replication pattern is applied so that Component B can interact with a database that is replicated from the shared Database A. This will improve performance and reliability that will affect both Component A and Service Consumer A. Finally, the Legacy Wrapper pattern is applied

so that Database B is wrapped in a standardized contract. This will establish a new wrapper utility service that will allow Database B to be replaced with a different database product without affecting Service A . Furthermore, the Data Format

Transformation pattern can be applied within the new wrapper utility service to enable it to convert to

and from the new proprietary file format.

B. The State Repository pattern is applied so that Component A can defer the business document data to a state database while it waits for a response from Component B. The Asynchronous Queuing pattern can be applied so that a messaging queue is established between Service Consumer A and Service A, thereby guaranteeing delivery and avoiding Service Consumer A from being tied up too long waiting for Service A to respond. Finally, the Data Format Transformation pattern can be applied to enable Component B to convert to and from the new proprietary file format introduced by the database product that is replacing Database B.

C. The Legacy Wrapper pattern is applied so that Database B is wrapped in a standardized contract. This will establish a new wrapper utility service that will allow Database B to be replaced with a different database product without affecting Service A . The Data Format Transformation pattern can be applied within the new wrapper utility service to enable it to convert to and from the new proprietary file format. The Service Data Replication pattern is applied so that Component B can interact with a database that is replicated from the shared Database B, regardless of what database product is used to replace Database B. The Service Abstraction principle can be further applied to hide the implementation details, including the changes mentioned in this solution, from Service Consumer A.

D. None of the above.

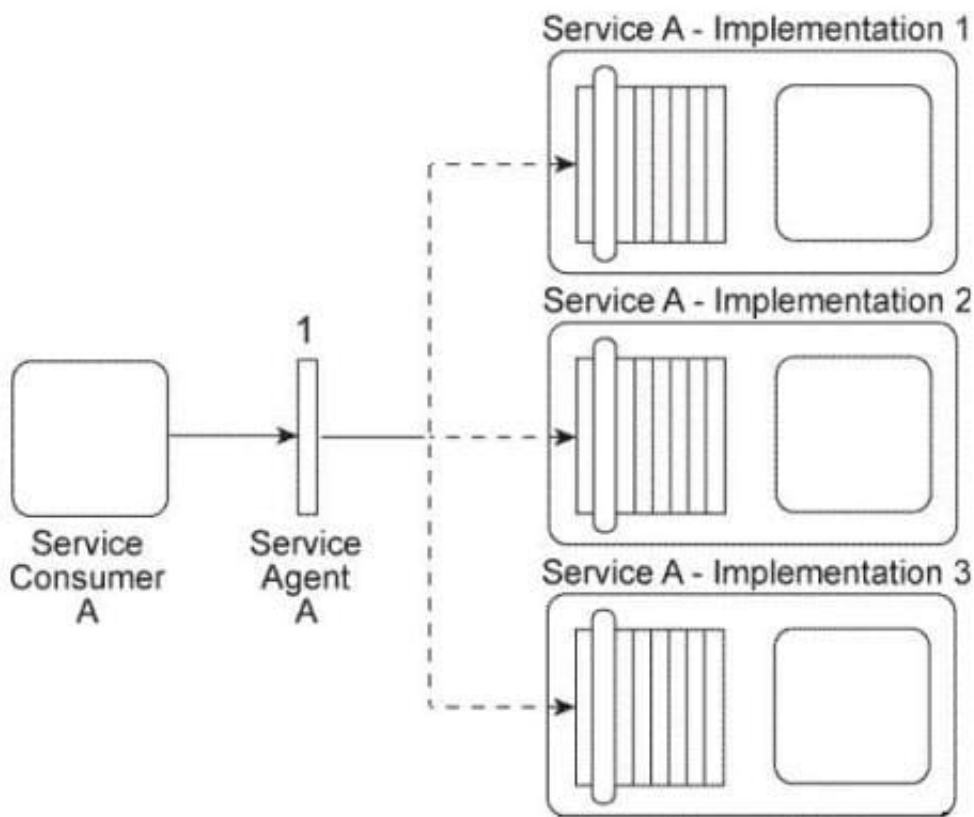
Correct Answer: A

QUESTION 2

Service Consumer A sends a message to Service A. There are currently three duplicate implementations of Service A (Implementation 1, Implementation 2, Implementation 3).

The message sent by Service Consumer A is intercepted by Service Agent A (1), which determines at runtime which implementation of Service A to forward the message to.

All three implementations of Service A reside on the same physical server.



You are told that after Service A was deployed, each of its three implementations was claimed by a different IT department, which means each implementation of Service A has a different owner. You are informed that a new service capability will soon need to be added to Service A. This service capability will introduce new business logic specific to Service A as well as logic required to access a shared database. What steps can be taken to ensure that the service owners will each add the service capability in a consistent manner to their respective implementations of Service A?

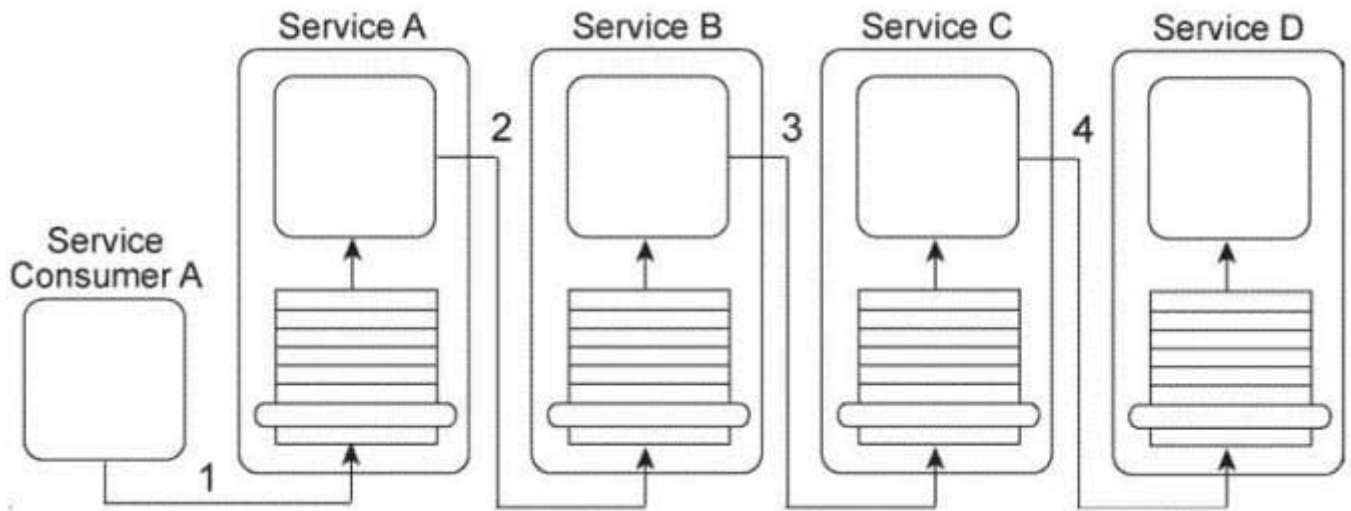
- A. The Contract Centralization pattern can be applied so that when the new service capability is added, the Service A service contract will become the primary contact point for Service A. This will avoid Service Consumer A or any other potential service consumer from being designed to access the shared database directly. The Service Abstraction principle can be applied to further hide the implementation details so that Service Consumer A and other service consumers are unaware of the fact that the shared database is being accessed.
- B. The Legacy Wrapper pattern can be applied to establish a new wrapper utility service that will provide standardized data access service capabilities for the shared database. This will avoid Service A from having to access the shared database directly and will further support the application of the Service Loose Coupling principle between Service A and the new utility service. By abstracting the data access logic into the wrapper service, there is no need to add the new service capability to each implementation of Service A.
- C. The Standardized Service Contract principle is applied to ensure that the new service capability is consistently added to the service contract of each implementation and that it extends the existing Service A service contract in a manner that is compliant with current design standards. The Service Loose Coupling principle is applied to ensure that the new service capability remains decoupled from the underlying logic and implementation so that Service Consumer A does not become indirectly coupled to any new logic or to the shared database.
- D. None of the above.

Correct Answer: C

QUESTION 3

Service Consumer A sends a message to Service A (1), which then forwards the message to Service B (2). Service B forwards the message to Service C (3), which finally forwards the message to Service D (4).

Services A, B, and C each contain logic that reads the content of the message and, based on this content, determines which service to forward the message to. As a result, what is shown in the Figure is one of several possible runtime scenarios.



You are told that the current service composition architecture is having performance problems because of two specific reasons. First, too many services need to be explicitly invoked in order for the message to arrive at its destination. Secondly, because each of the intermediary services is required to read the entire message contents in order to determine where to forward the message to, it is taking too long for the overall task to complete. What steps can be taken to solve these problems without sacrificing any of the functionality that currently exists?

- A. The Intermediate Routing pattern can be applied together with the Service Agent pattern in order to establish a set of service agents capable of intercepting and forwarding the message based on predefined routing logic. To avoid the need for service agents to read the entire message contents, the Messaging Metadata pattern can be applied so that content relevant to the routing logic is placed in the header of a message. This way, only the message header content needs to be read by the service agents.
- B. The Intermediate Routing pattern can be applied together with the Service Agent pattern in order to establish a set of service agents capable of intercepting and forwarding the message based on predefined routing logic. To avoid the need for service agents to read the entire message contents, the Rules Centralization pattern can be applied so that content relevant to the routing logic is isolated into a separate Rules service. This way, service agents are only required to access the Rules service in order to determine where to forward messages to. The Standardized Service Contract principle will need to be applied to ensure that the new Rules service and the new service agents provide service contracts that are compliant to existing design standards.
- C. The Intermediate Routing pattern can be applied together with the Service Agent pattern in order to establish a set of service agents capable of intercepting and forwarding the message based on predefined routing logic. The Service Discoverability principle can be applied to improve the communications quality of message contents, which will reduce the time required by service agents to read the message contents at runtime.
- D. None of the above.

Correct Answer: A

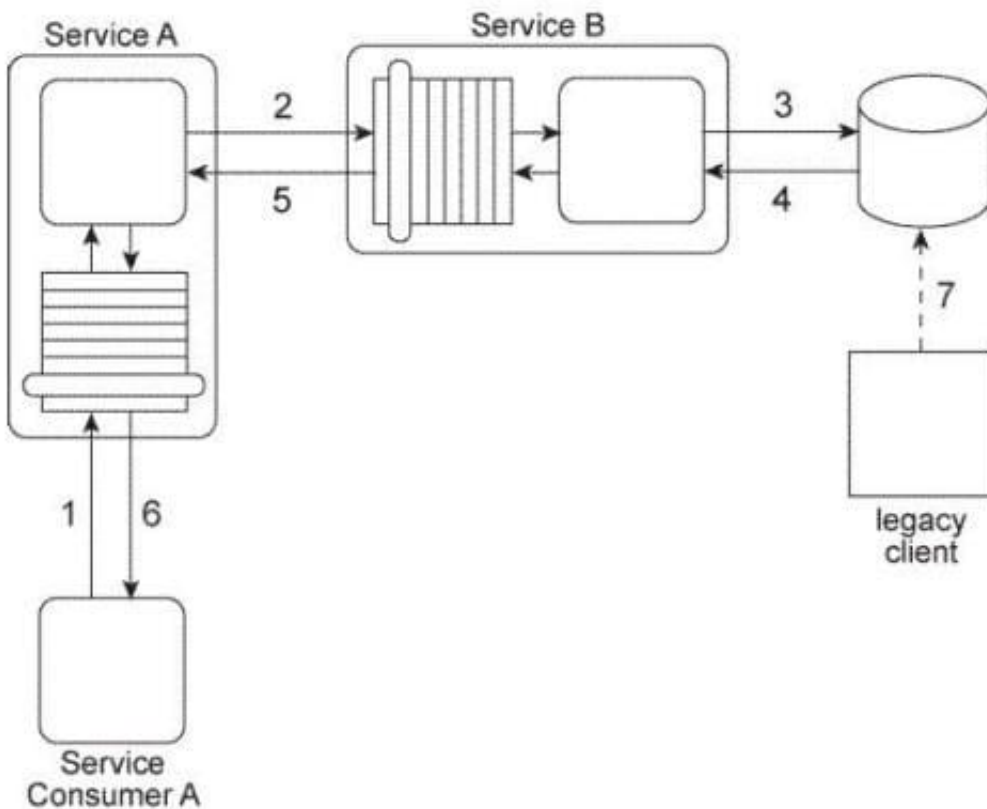
QUESTION 4

Service A is an entity service that provides a Get capability that returns a data value that is frequently changed.

Service Consumer A invokes Service A in order to request this data value (1). For Service A to carry out this request, it must invoke Service B (2), a utility service that interacts (3,4) with the database in which the data value is stored. Regardless of whether the data value changed, Service B returns the latest value to Service A (5), and Service A returns the latest value to Service Consumer A (6).

The data value is changed when the legacy client program updates the database (7) When this change happens is not predictable. Note also that Service A and Service B are not always available at the same time.

Any time the data value changes. Service Consumer A needs to receive it as soon as possible. Therefore, Service Consumer A initiates the message exchange shown in the Figure several times a day. When it receives the same data value as before, the response from Service A is ignored. When Service A provides an updated data value, Service Consumer A can process it to carry out its task.



The current service composition architecture is using up too many resources due to the repeated invocation of Service A by Service Consumer A and the resulting message exchanges that occur with each invocation. What steps can be taken to solve this problem?

A. The Event-Driven Messaging pattern can be applied by establishing a subscriber- publisher relationship between Service A and Service B . This way, every time the data value is updated, an event is triggered and Service B, acting as the publisher, can notify Service A, which acts as the subscriber. The Asynchronous Queuing pattern can be applied between Service A and Service B so that the event notification message sent out by Service B will be received by Service A, even when Service A is unavailable.

B. The Event-Driven Messaging pattern can be applied by establishing a subscriber- publisher relationship between

Service Consumer A and Service A . This way, every time the data value is updated, an event is triggered and Service A, acting as the publisher, can notify Service Consumer A, which acts as the subscriber. The Asynchronous Queuing pattern can be applied between Service Consumer A and Service A so that the event notification message sent out by Service A will be

received by Service Consumer A, even when Service Consumer A is unavailable.

C. The Asynchronous Queuing pattern can be applied so that messaging queues are established between Service A and Service B and between Service Consumer A and Service A . This way, messages are never lost due to the unavailability of Service A or Service B.

D. None of the above.

Correct Answer: D

QUESTION 5

You are an architect with a project team building services for Service Inventory A . You are told that no SLAs for Service B and Service C are available. You cannot determine how available these services will be, but it has been confirmed that both of these services support atomic transactions and the issuance of positive and negative acknowledgements. However, you also find out that the services in Service Inventory B use different data models than the services in Service Inventory A. Furthermore, recent testing results have shown that the performance of Service D is steady and reliable. However, Service D uses a different transport protocol than the services in Service Inventory A. The response time of Service A is not a primary concern, but Service Consumer A does need to be able to issue request messages to Service A 24 hours a day without disruption. What steps can be taken to fulfill these requirements?

A. The Event-Driven Messaging pattern is applied so that a subscriber-publisher relationship is established between Service Consumer A and Service A . This gives Service A the flexibility to provide its response to Service Consumer A whenever it is able to collect the three data values without having to require that Service Consumer A remain stateful. The Asynchronous Queuing pattern is applied so that a central messaging queue is positioned between Service A and Service B and between Service A and Service C . The Data Model Transformation and Protocol Bridging patterns are applied to enable communication between Service A and Service B and between Service A and Service C . The Service Autonomy principle is further applied to Service A in order to improve its overall runtime behavioral predictability.

B. The Reliable Messaging pattern is applied so that a system of acknowledgements is established between Service Consumer A and Service A . This gives Service A the flexibility to provide Service Consumer A with acknowledgements that indicate that the processing steps that are occurring between Service A and Service B, Service C, and Service D are progressing. The Asynchronous Queuing pattern is applied so that a central messaging queue is positioned between Service A and Service B and between Service A and Service C and between Service A and Service D . The Redundant Implementation pattern is applied so that a copy of Service D is brought in-Upon reviewing these requirements it becomes D with a standardized service contract that is in compliance with the design standards used in Service Inventory A.

C. The Asynchronous Queuing pattern is applied so that a central messaging queue is positioned between Service A and Service B and between Service A and Service C and between Service A and Service D and so that a separate messaging queue is positioned between Service A and Service Consumer A. The Data Model Transformation pattern is applied to enable communication between Service A and Service B and between Service A and Service C . The Protocol Bridging pattern is applied to enable communication between Service A and Service D .

D. None of the above.

Correct Answer: C

[Latest S90.09 Dumps](#)

[S90.09 PDF Dumps](#)

[S90.09 Practice Test](#)