PASS QUESTION

QUESTION & ANSWER

HIGHER QUALITY, BETTER SERVICE

Provide One Year Free Update!

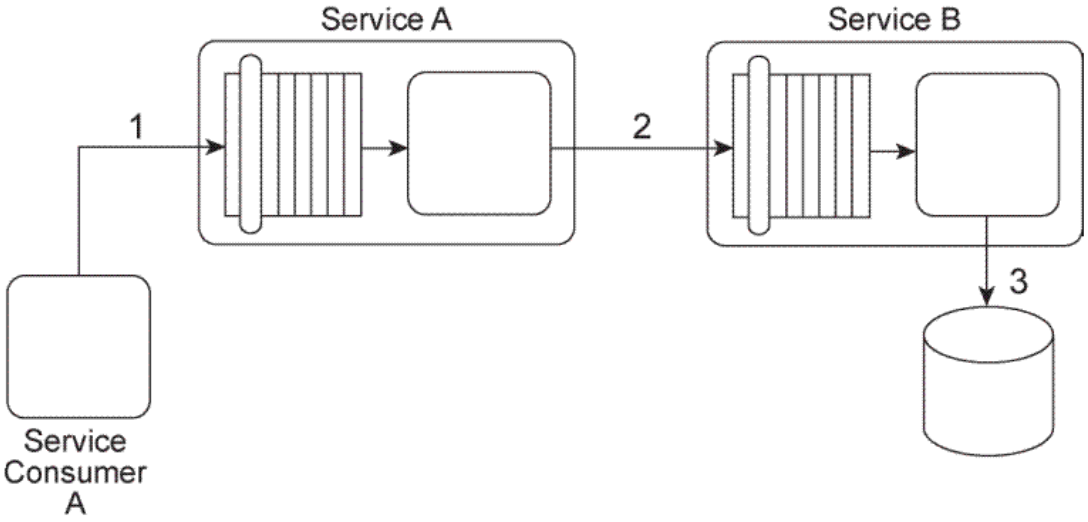
https://www.passquestion.com

Exam : **S90-09A**

Title : SOA Design & Architecture Lab

Version : DEMO

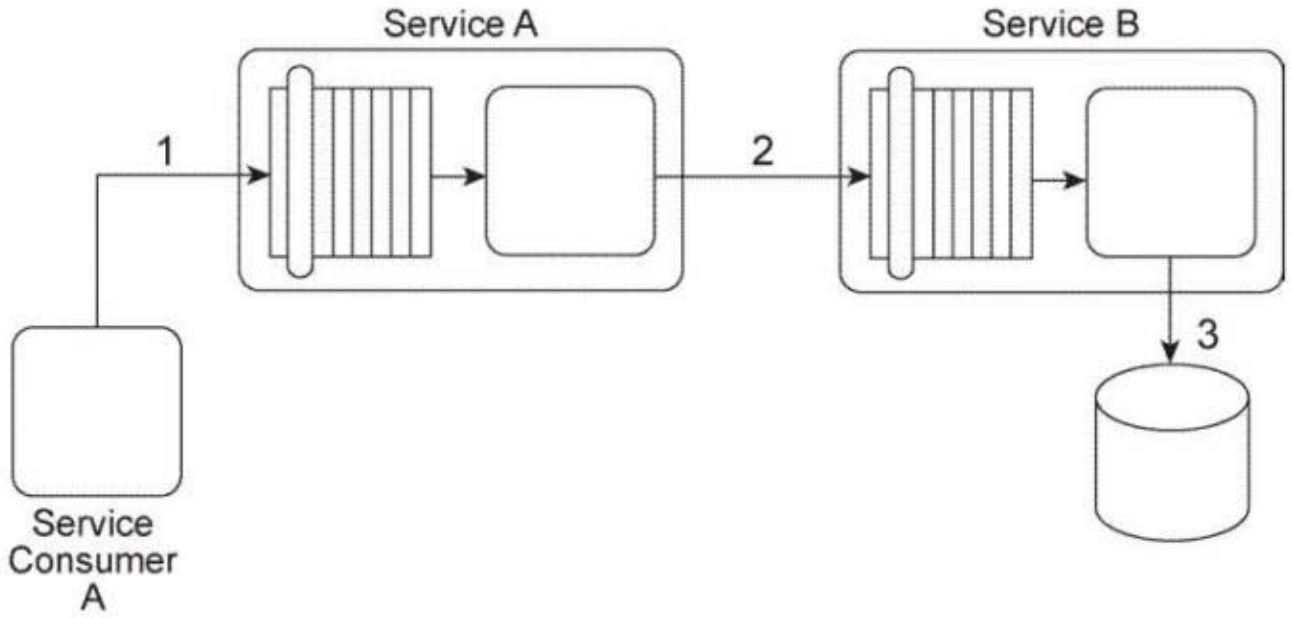
1.Service A is an entity service with a functional context dedicated to invoice-related processing. Service B is a utility service that provides generic data access to a database.
In this service composition architecture, Service Consumer A sends a SOAP message containing an invoice XML document to Service A (1). Service A then sends the invoice XML document to Service B (2), which then writes the invoice document to a database.
The data model used by Service Consumer A to represent the invoice document is based on XML Schema A. The service contract of Service A is designed to accept invoice documents based on XML Schema B. The service contract for Service B is designed to accept invoice documents based on XML Schema A. The database to which Service B needs to write the invoice record only accepts entire business documents in Comma Separated Value (CSV) format.
Due to the incompatibility of the XML schemas used by the services, the sending of the invoice document from Service Consumer A through to Service B cannot be accomplished using the services as they currently exist. Assuming that the Contract Centralization pattern is being applied and that the Logic Centralization is not being applied, what steps can be taken to enable the sending of the invoice document from Service Consumer A to the database without adding logic that will increase the runtime performance requirements of the service composition?



A.Service Consumer A can be redesigned to use XML Schema B so that the SOAP message it sends is compliant with the service contract of Service A. The Data Model Transformation pattern can then be applied to transform the SOAP message sent by Service A so that it conforms to the XML Schema A used by Service B. The Standardized Service Contract principle must then be applied to Service B and Service Consumer A so that the invoice XML document is optimized to avoid unnecessary validation.
B.The service composition can be redesigned so that Service Consumer A sends the invoice document directly to Service B. Because Service Consumer A and Service B use XML Schema A, the need for transformation logic is avoided. This naturally applies the Service Loose Coupling principle because Service Consumer A is not required to send the invoice document in a format that is compliant with the database used by Service B.

C.Service Consumer A can be redesigned to write the invoice document directly to the database. This reduces performance requirements by avoiding the involvement of Service A and Service B. It further supports the application of the Service Abstraction principle by ensuring that Service Consumer A hides the details of the data access logic required to write to the database.
D.None of the above.
Answer:B

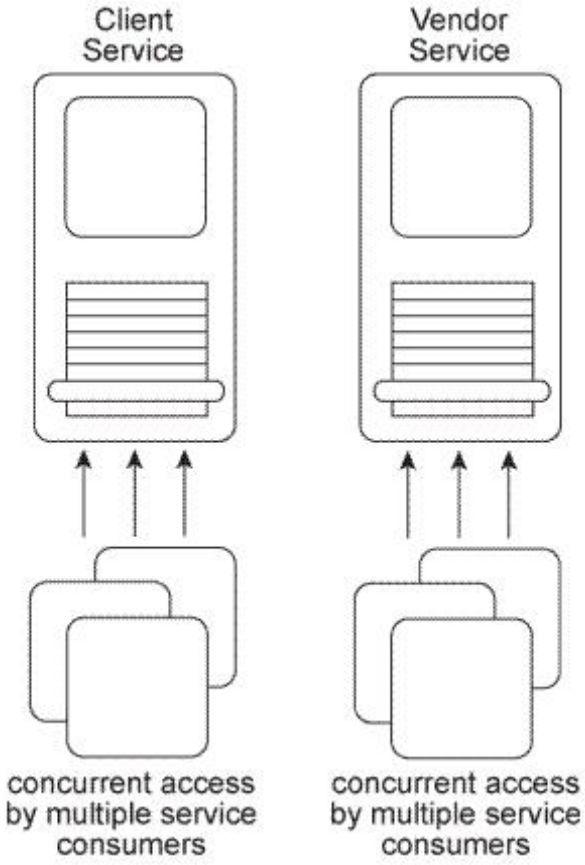
2.Service A is an entity service with a functional context dedicated to invoice-related processing. Service B is a utility service that provides generic data access to a database.
In this service composition architecture, Service Consumer A sends a SOAP message containing an invoice XML document to Service A (1). Service A then sends the invoice XML document to Service B (2), which then writes the invoice document to a database.
The data model used by Service Consumer A to represent the invoice document is based on XML Schema A. The service contract of Service A is designed to accept invoice documents based on XML Schema B. The service contract for Service B is designed to accept invoice documents based on XML Schema A. The database to which Service B needs to write the invoice record only accepts entire business documents in Comma Separated Value (CSV) format.
Due to the incompatibility of XML schemas used by the services, the sending of the invoice document from Service Consumer A through to Service B cannot be accomplished using the services as they currently exist. Assuming that the Contract Centralization and Logic Centralization patterns are being applied, what steps can be taken to enable the sending of the invoice document from Service Consumer A to the database without adding logic that will increase the runtime performance of the service composition?



A.The Data Model Transformation pattern can be applied so that the invoice document sent by Service Consumer A is transformed into an invoice document that is compliant with the XML Schema B used by

Service A. The Data Model Transformation pattern can be applied again to ensure that the invoice document sent by Service A is compliant with XML Schema A used by Service B.
B.The service composition can be redesigned so that Service Consumer A sends the invoice document directly to Service B. Because Service Consumer A and Service B use XML Schema A, the need for transformation logic is avoided. This naturally applies the Service Loose Coupling principle because Service Consumer A is not required to send the invoice document in a format that is compliant with the database used by Service B.
C.The Standardized Service Contract principle can be applied to the service contract of Service A so that it is redesigned to use XML Schema A. This would make it capable of receiving the invoice document from Service Consumer A and sending the invoice document to Service B without the need to further apply the Data Model Transformation pattern.
D.None of the above.
Answer:C

3.The Client and Vendor services are agnostic services that are both currently part of multiple service compositions. As a result, these services are sometimes subjected to concurrent access by multiple service consumers.
The Client service is an entity service that primarily provides data access logic to a client database but also provides some calculation logic associated with determining a client's credit rating. The Vendor service is also an entity service that provides some data access logic but can also generate various dynamic reports.
After reviewing historical statistics about the runtime activity of the two services, it was discovered that the majority of concurrent runtime access is related to the processing of business rules. With the Client service, it is the calculation logic that is frequently required and with the Vendor service it is the dynamic reporting logic that needs to be accessed separately from the actual report generation.
Currently, due to the increasing amount of concurrent access by service consumers, the runtime performance of both the Client and Vendor services has worsened and has therefore reduced their effectiveness as service composition members. What steps can be taken to solve this problem without introducing new services?

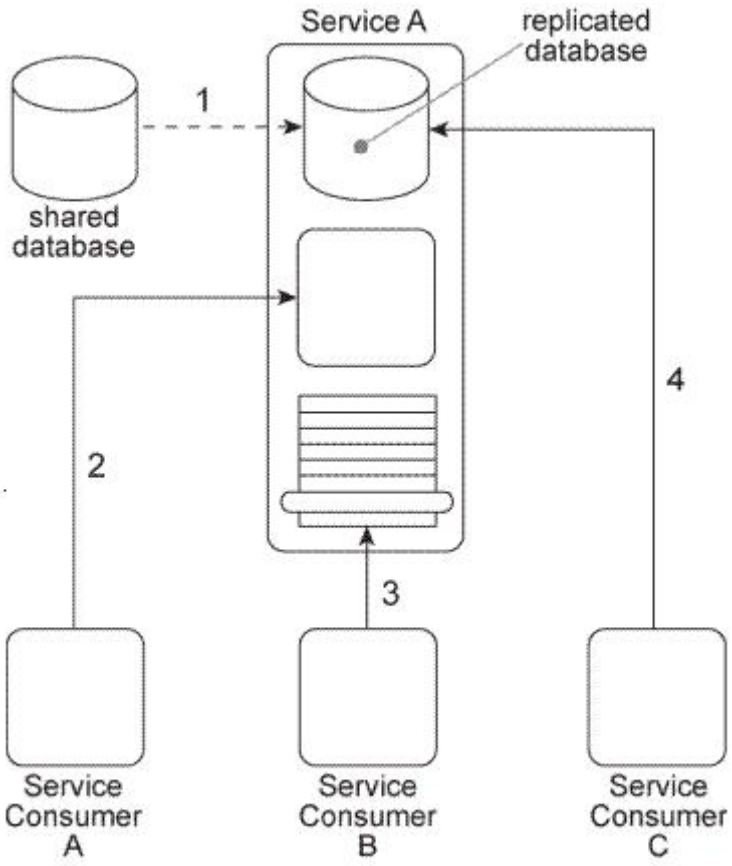


A.The Rules Centralization pattern can be applied by extracting the business rule logic from the Client and Vendor services and placing it into a new Rules service. This will naturally improve the runtime performance of the Client and Vendor services because they will no longer be subjected to the high concurrent access of service consumers that require access to the business rules logic.
B.The Redundant Implementation pattern can be applied to the Client and Vendor services, thereby establishing duplicate implementations that can be accessed when a service reaches its runtime usage threshold. The Intermediate Routing pattern can be further applied to provide load balancing logic that can, at runtime, determine which of the redundant service implementations is the least busy for a given service consumer request.
C.The Rules Centralization pattern can be applied together with the Redundant Implementation pattern to establish a scalable Rules service that is redundantly implemented and therefore capable of supporting high concurrent access from many service consumers. The Service Abstraction principle can be further applied to hide the implementation details of the Rules service.
D.None of the above.
Answer:B

4.The Client and Vendor services are agnostic services that are both currently part of multiple service compositions. As a result, these services are sometimes subjected to concurrent access by multiple service consumers.

The Client service is an entity service that primarily provides data access logic to a client database but also provides some calculation logic associated with determining a client's credit rating. The Vendor service is also an entity service that provides some data access logic but can also generate various dynamic reports.
After reviewing historical statistics about the runtime activity of the two services, it was discovered that the majority of concurrent runtime access is related to the processing of business rules. With the Client service, it is the calculation logic that is frequently required and with the Vendor service it is the dynamic reporting logic that needs to be accessed separately from the actual report generation.
Currently, due to the increasing amount of concurrent access by service consumers, the runtime performance of both the Client and Vendor services has worsened and has therefore reduced their effectiveness as service composition members. Additionally, a review of the logic of both services has revealed that some of the business rules used by the Client and Vendor services are actually the same.
What steps can be taken to improve performance and reduce redundant business rule logic?
A.The Rules Centralization pattern can be applied by extracting the business rule logic from the Client and Vendor services and placing it into a new Rules service, thereby reducing the redundancy of business rules logic. The Redundant Implementation pattern can then be applied to establish a scalable Rules service that is capable of supporting concurrent access from many service consumers.
B.The Redundant Implementation pattern can be applied to the Client and Vendor services, thereby establishing duplicate service implementations that can be accessed when a service reaches its runtime usage threshold. The Intermediate Routing pattern can be further applied to provide load balancing logic that can, at runtime, determine which of the redundant service implementations is the least busy for a given service consumer request.
C.The Rules Centralization pattern can be applied to isolate business rules logic into a central and reusable Rules service. Additionally, the Service Abstraction principle can be applied to hide the implementation details of new the Rules service.
D.None of the above.
Answer:A

5.Service A is a utility service that provides generic data access logic to a database that contains data that is periodically replicated from a shared database (1). Because the Standardized Service Contract principle was applied to the design of Service A, its service contract has been fully standardized.
The service architecture of Service A is being accessed by three service consumers. Service Consumer A accesses a component that is part of the Service A implementation by invoking it directly (2). Service Consumer B invokes Service A by accessing its service contract (3). Service Consumer C directly accesses the replicated database that is part of the Service A implementation (4).
You've been told that the shared database will soon be replaced with a new database product that will have new data models and new replication technology. How can the Service A architecture be changed to avoid negative impacts that may result from the replacement of the database and to establish a service architecture in which negative forms of coupling can be avoided in the future?



A.The Contract Centralization pattern can be applied to force all service consumers to access the Service A architecture via its published service contract. This will prevent negative forms of coupling that could lead to problems when the database is replaced. The Service Abstraction principle can then be applied to hide underlying service implementation details so that future service consumers cannot be designed to access any part of the underlying service implementation.

B.The Contract Centralization pattern can be applied to force Service Consumer C to access the Service A architecture via its published service contract. This will prevent Service Consumer A from being negatively impacted when the database is replaced in the future.

C.The Standardized Service Contract principle can be applied to force Service Consumer B to comply to the standardized service contract of Service A. As a result, the coupling between Service Consumer B and Service A is reduced. The Logic Centralization pattern can then be applied to position the logic provided by Service A as a primary access point for the database. As a result, the component within the Service A architecture abstracts the proprietary details of the database, thereby shielding Service Consumer A (and any future service consumers) from changes made to the database.

D.None of the above.

Answer:A