



MuleSoft

Exam Questions MCIA-Level-1

MuleSoft Certified Integration Architect - Level 1

About ExamBible

Your Partner of IT Exam

Found in 1998

ExamBible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, ExamBible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

All examinations will be up to date.

* 24/7 Quality Support

We will provide service round the clock.

* 100% Pass Rate

Our guarantee that you will pass the exam.

* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1

An Integration Mule application is being designed to synchronize customer data between two systems. One system is an IBM Mainframe and the other system is a Salesforce Marketing Cloud (CRM) instance. Both systems have been deployed in their typical configurations, and are to be invoked using the native protocols provided by Salesforce and IBM.

What interface technologies are the most straightforward and appropriate to use in this Mule application to interact with these systems, assuming that Anypoint Connectors exist that implement these interface technologies?

- A. IBM: DB access CRM: gRPC
- B. IBM: REST CRM: REST
- C. IBM: Active MQ CRM: REST
- D. IBM: CICS CRM: SOAP

Answer: D

Explanation:

Correct answer is IBM: CICS CRM: SOAP

* Within Anypoint Exchange, MuleSoft offers the IBM CICS connector. Anypoint Connector for IBM CICS Transaction Gateway (IBM CTG Connector) provides integration with back-end CICS apps using the CICS Transaction Gateway.

* Anypoint Connector for Salesforce Marketing Cloud (Marketing Cloud Connector) enables you to connect to the Marketing Cloud API web services (now known as the Marketing Cloud API), which is also known as the Salesforce Marketing Cloud. This connector exposes convenient operations via SOAP for exploiting the capabilities of Salesforce Marketing Cloud.

NEW QUESTION 2

A system API EmployeeSAPI is used to fetch employee's data from an underlying SQL database.

The architect must design a caching strategy to query the database only when there is an update to the employees stable or else return a cached response in order to minimize the number of redundant transactions being handled by the database.

What must the architect do to achieve the caching objective?

- A. Use an On Table Row on employees table and call invalidate cache Use an object store caching strategy and expiration interval to empty
- B. Use a Scheduler with a fixed frequency every hour triggering an invalidate cache flow Use an object store caching strategy and expiration interval to empty
- C. Use a Scheduler with a fixed frequency every hour triggering an invalidate cache flow Use an object store caching strategy and set expiration interval to 1-hour
- D. Use an on table rule on employees table call invalidate cache and said new employees data to cache Use an object store caching strategy and set expiration interval to 1-hour

Answer: A

NEW QUESTION 3

An organization has implemented a continuous integration (CI) lifecycle that promotes Mule applications through code, build, and test stages. To standardize the organization's CI journey, a new dependency control approach is being designed to store artifacts that include information such as dependencies, versioning, and build promotions.

To implement these process improvements, the organization will now require developers to maintain all dependencies related to Mule application code in a shared location.

What is the most idiomatic (used for its intended purpose) type of system the organization should use in a shared location to standardize all dependencies related to Mule application code?

- A. A MuleSoft-managed repository at repository.mulesoft.org
- B. A binary artifact repository
- C. API Community Manager
- D. The Anypoint Object Store service at cloudhub.io

Answer: C

NEW QUESTION 4

An organization has various integrations implemented as Mule applications. Some of these Mule applications are deployed to custom hosted Mule runtimes (on-premises) while others execute in the MuleSoft-hosted runtime plane (CloudHub). To perform the Integra functionality, these Mule applications connect to various backend systems, with multiple applications typically needing to access the backend systems.

How can the organization most effectively avoid creating duplicates in each Mule application of the credentials required to access the backend systems?

- A. Create a Mule domain project that maintains the credentials as Mule domain-shared resources Deploy the Mule applications to the Mule domain, so the credentials are available to the Mule applications
- B. Store the credentials in properties files in a shared folder within the organization's data center Have the Mule applications load properties files from this shared location at startup
- C. Segregate the credentials for each backend system into environment-specific properties files Package these properties files in each Mule application, from where they are loaded at startup
- D. Configure or create a credentials service that returns the credentials for each backend system, and that is accessible from customer-hosted and MuleSoft-hosted Mule runtimes Have the Mule applications load the properties at startup by invoking that credentials service

Answer: D

Explanation:

* "Create a Mule domain project that maintains the credentials as Mule domain-shared resources" is wrong as domain project is not supported in Cloudhub * We should Avoid Creating duplicates in each Mule application but below two options cause duplication of credentials - Store the credentials in properties files in a shared folder within the organization's data center. Have the Mule applications load properties files from this shared location at startup - Segregate the credentials for each backend system into environment-specific properties files. Package these properties files in each Mule application, from where they are loaded at startup So these are also wrong choices * Credentials service is the best approach in this scenario. Mule domain projects are not supported on CloudHub. Also its is not recommended to have multiple copies of configuration values as this makes difficult to maintain Use the Mule Credentials Vault to encrypt data in a .properties file. (In the context of this document, we refer to the .properties file simply as the properties file.) The properties file in Mule stores data as key-value pairs which may contain information such as usernames, first and last names, and credit card numbers. A Mule application may access this data as it processes messages, for

example, to acquire login credentials for an external Web service. However, though this sensitive, private data must be stored in a properties file for Mule to access, it must also be protected against unauthorized – and potentially malicious – use by anyone with access to the Mule application

NEW QUESTION 5

In Anypoint Platform, a company wants to configure multiple identity providers (Idps) for various lines of business (LOBs). Multiple business groups and environments have been defined for these LOBs. What Anypoint Platform feature can use multiple Idps to access the company's business groups and environment?

- A. User management
- B. Roles and permissions
- C. Dedicated load balancers
- D. Client Management

Answer: D

Explanation:

Correct answer is Client Management

* Anypoint Platform acts as a client provider by default, but you can also configure external client providers to authorize client applications.

* As an API owner, you can apply an OAuth 2.0 policy to authorize client applications that try to access your API. You need an OAuth 2.0 provider to use an OAuth 2.0 policy.

* You can configure more than one client provider and associate the client providers with different environments. If you configure multiple client providers after you have already created environments, you can associate the new client providers with the environment.

* You should review the existing client configuration before reassigning client providers to avoid any downtime with existing assets or APIs.

* When you delete a client provider from your master organization, the client provider is no longer available in environments that used it.

* Also, assets or APIs that used the client provider can no longer authorize users who want to access them.

-----MuleSoft

Reference: <https://docs.mulesoft.com/access-management/managing-api-clients>

<https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html>

NEW QUESTION 6

A set of integration Mule applications, some of which expose APIs, are being created to enable a new business process. Various stakeholders may be impacted by this. These stakeholders are a combination of semi-technical users (who understand basic integration terminology and concepts such as JSON and XML) and technically skilled potential consumers of the Mule applications and APIs.

What is an effective way for the project team responsible for the Mule applications and APIs being built to communicate with these stakeholders using Anypoint Platform and its supplied toolset?

- A. Use Anypoint Design Center to implement the Mule applications and APIs and give the various stakeholders access to these Design Center projects, so they can collaborate and provide feedback
- B. Create Anypoint Exchange entries with pages elaborating the integration design, including API notebooks (where applicable) to help the stakeholders understand and interact with the Mule applications and APIs at various levels of technical depth
- C. Use Anypoint Exchange to register the various Mule applications and APIs and share the RAML definitions with the stakeholders, so they can be discovered
- D. Capture documentation about the Mule applications and APIs inline within the Mule integration flows and use Anypoint Studio's Export Documentation feature to provide an HTML version of this documentation to the stakeholders

Answer: B

Explanation:

As the stakeholders are semitechnical users, preferred option is Create Anypoint Exchange entries with pages elaborating the integration design, including API notebooks (where applicable) to help the stakeholders understand and interact with the Mule applications and APIs at various levels of technical depth

NEW QUESTION 7

What requires configuration of both a key store and a trust store for an HTTP Listener?

- A. Support for TLS mutual (two-way) authentication with HTTP clients
- B. Encryption of requests to both subdomains and API resource endpoints `https://aDi.customer.com/` and `https://customer.com/api`
- C. Encryption of both HTTP request and HTTP response bodies for all HTTP clients
- D. Encryption of both HTTP request header and HTTP request body for all HTTP clients

Answer: A

Explanation:

1- way SSL : The server presents its certificate to the client and the client adds it to its list of trusted certificates. And so, the client can talk to the server.

2- way SSL: The same principle but both ways. i.e. both the client and the server have to establish trust between themselves using a trusted certificate. In this way of a digital handshake, the server needs to present a certificate to authenticate itself to the client and the client has to present its certificate to the server.

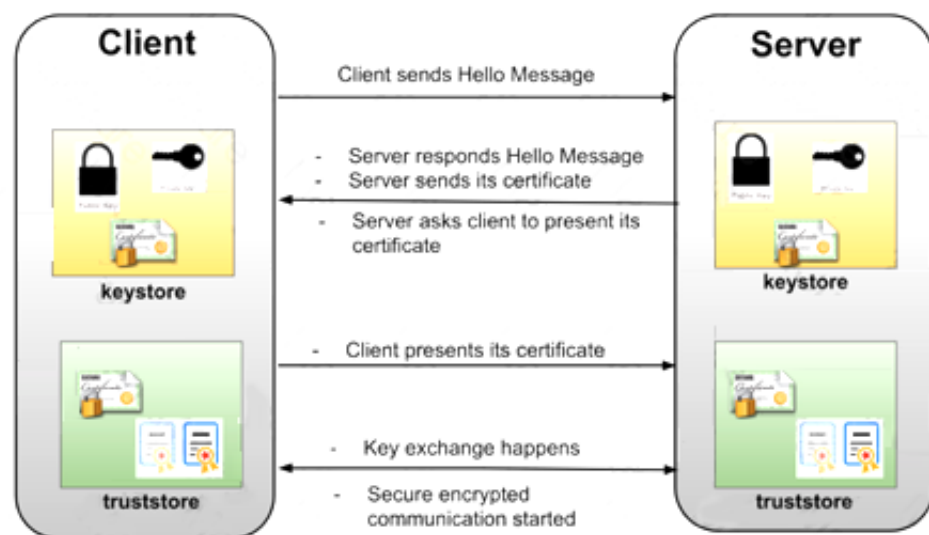
* TLS is a cryptographic protocol that provides communications security for your Mule app.

* TLS offers many different ways of exchanging keys for authentication, encrypting data, and guaranteeing message integrity. Keystores and Truststores. Truststore and keystore contents differ depending on whether they are used for clients or servers:

For servers: the truststore contains certificates of the trusted clients, the keystore contains the private and public key of the server. For clients: the truststore contains certificates of the trusted servers, the keystore contains the private and public key of the client.

Adding both a keystore and a truststore to the configuration implements two-way TLS authentication, also known as mutual authentication.

* In this case, correct answer is Support for TLS mutual (two-way) authentication with HTTP clients.



NEW QUESTION 8

A REST API is being designed to implement a Mule application.

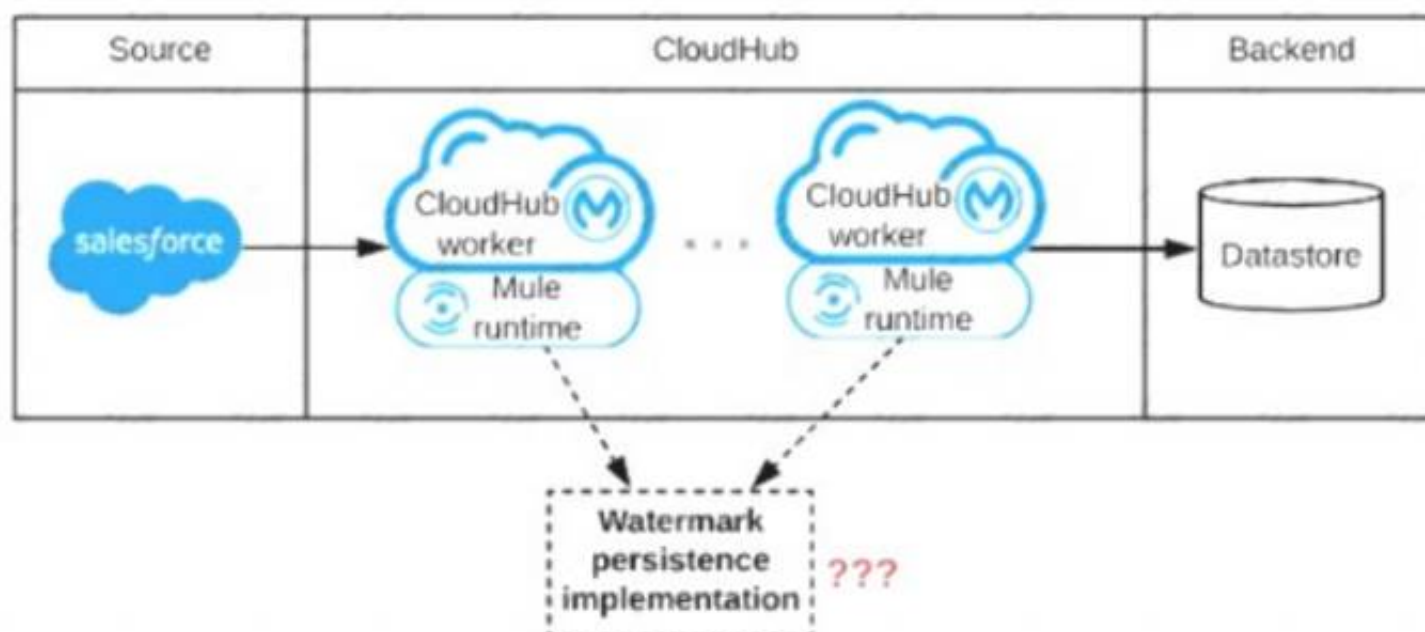
What standard interface definition language can be used to define REST APIs?

- A. Web Service Definition Language(WSDL)
- B. OpenAPI Specification (OAS)
- C. YAML
- D. AsyncAPI Specification

Answer: B

NEW QUESTION 9

Refer to the exhibit.



A Mule application is being designed to be deployed to several CloudHub workers. The Mule application's integration logic is to replicate changed Accounts from Satesforce to a backend system every 5 minutes.

A watermark will be used to only retrieve those Satesforce Accounts that have been modified since the last time the integration logic ran.

What is the most appropriate way to implement persistence for the watermark in order to support the required data replication integration logic?

- A. Persistent Anypoint MQ Queue
- B. Persistent Object Store
- C. Persistent Cache Scope
- D. Persistent VM Queue

Answer: B

Explanation:

* An object store is a facility for storing objects in or across Mule applications. Mule uses object stores to persist data for eventual retrieval.

* Mule provides two types of object stores:

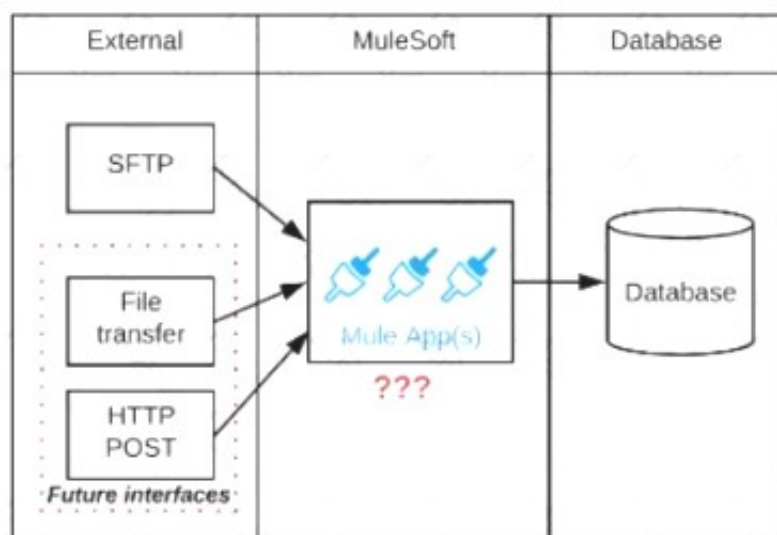
- 1) In-memory store – stores objects in local Mule runtime memory. Objects are lost on shutdown of the Mule runtime.
- 2) Persistent store – Mule persists data when an object store is explicitly configured to be persistent.

In a standalone Mule runtime, Mule creates a default persistent store in the file system. If you do not specify an object store, the default persistent object store is used.

MuleSoft Reference: <https://docs.mulesoft.com/mule-runtime/3.9/mule-object-stores>

NEW QUESTION 10

Refer to the exhibit.



A business process involves the receipt of a file from an external vendor over SFTP. The file needs to be parsed and its content processed, validated, and ultimately persisted to a database. The delivery mechanism is expected to change in the future as more vendors send similar files using other mechanisms such as file transfer or HTTP POST.

What is the most effective way to design for these requirements in order to minimize the impact of future change?

- A. Use a MuleSoft Scatter-Gather and a MuleSoft Batch Job to handle the different files coming from different sources
- B. Create a Process API to receive the file and process it using a MuleSoft Batch Job while delegating the data save process to a System API
- C. Create an API that receives the file and invokes a Process API with the data contained In the file, then have the Process API process the data using a MuleSoft Batch Job and other System APIs as needed
- D. Use a composite data source so files can be retrieved from various sources and delivered to a MuleSoft Batch Job for processing

Answer: C

Explanation:

* Scatter-Gather is used for parallel processing, to improve performance. In this scenario, input files are coming from different vendors so mostly at different times. Goal here is to minimize the impact of future change. So scatter Gather is not the correct choice.

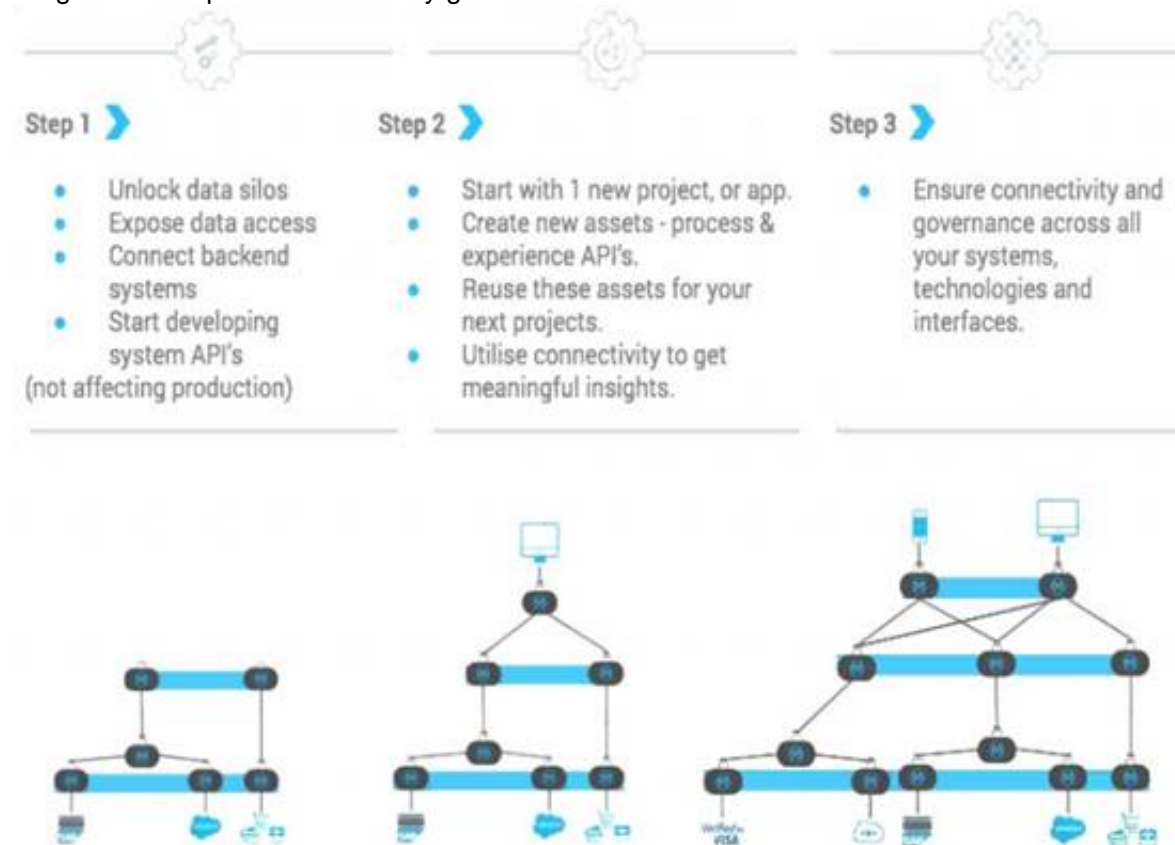
* If we use 1 API to receive all files from different Vendors, any new vendor addition will need changes to that 1 API to accommodate new requirements. So Option A and C are also ruled out.

* Correct answer is Create an API that receives the file and invokes a Process API with the data contained in the file, then have the Process API process the data using a MuleSoft Batch Job and other System APIs as needed. Answer to this question lies in the API led connectivity approach.

* API-led connectivity is a methodical way to connect data to applications through a series of reusable and purposeful modern APIs that are each developed to play a specific role – unlock data from systems, compose data into processes, or deliver an experience. System API : System API tier, which provides consistent, managed, and secure access to backend systems. Process APIs : Process APIs take core assets and combines them with some business logic to create a higher level of value. Experience APIs : These are designed specifically for consumption by a specific end-user app or device.

So in case of any future plans , organization can only add experience API on addition of new Vendors, which reuse the already existing process API. It will keep impact minimal.

Diagram Description automatically generated



NEW QUESTION 10

In a Mule Application, a flow contains two (2) JMS consume operations that are used to connect to a JMS broker and consume messages from two(2) JMS destination. The Mule application then joins the two JMS messages together.

The JMS broker does not implement high availability (HA) and periodically experiences scheduled outages of upto 10 mins for routine maintenance.

What is the most idiomatic (used for its intended purpose) way to build the mule flow so it can best recover from the expected outages?

- A. Configure a reconnection strategy for the JMS connector
- B. Enclose the two(2) JMS operation in an Until Successful scope
- C. Consider a transaction for the JMS connector
- D. Enclose the two(2) JMS operations in a Try scope with an Error Continue error handler

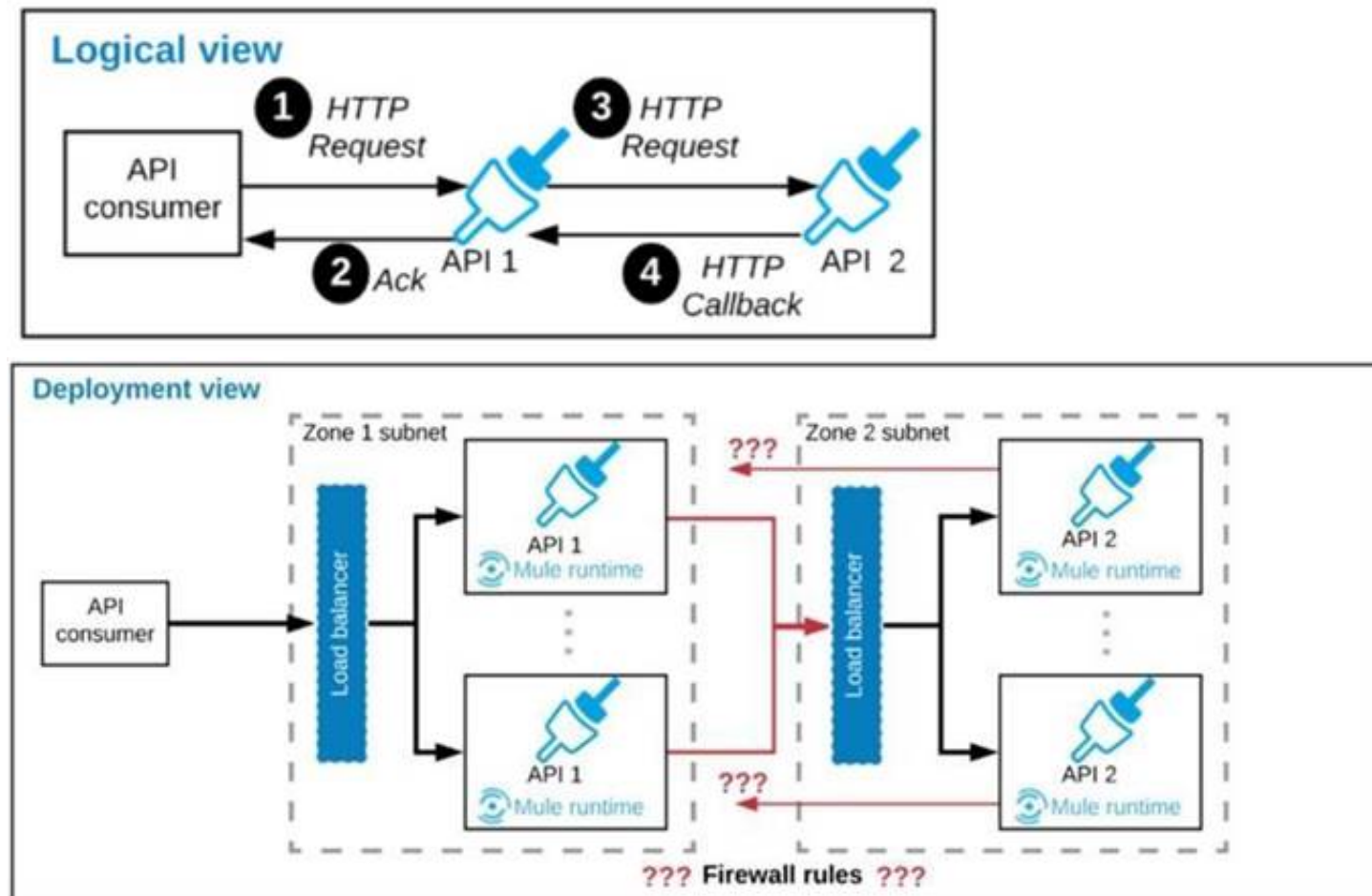
Answer: A

Explanation:

When an operation in a Mule application fails to connect to an external server, the default behavior is for the operation to fail immediately and return a connectivity error. You can modify this default behavior by configuring a reconnection strategy for the operation. You can configure a reconnection strategy for an operation either by modifying the operation properties or by modifying the configuration of the global element for the operation. The following are the available reconnection strategies and their behaviors: None Is the default behavior, which immediately returns a connectivity error if the attempt to connect is unsuccessful Standard (reconnect) Sets the number of reconnection attempts and the interval at which to execute them before returning a connectivity error Forever (reconnect-forever) Attempts to reconnect continually at a given interval

NEW QUESTION 15

Refer to the exhibit.



A business process involves two APIs that interact with each other asynchronously over HTTP. Each API is implemented as a Mule application. API 1 receives the initial HTTP request and invokes API 2 (in a fire and forget fashion) while API 2, upon completion of the processing, calls back into API 1 to notify about completion of the asynchronous process.

Each API is deployed to multiple redundant Mule runtimes and a separate load balancer, and is deployed to a separate network zone.

In the network architecture, how must the firewall rules be configured to enable the above Interaction between API 1 and API 2?

- A. To authorize the certificate to be used both APIs
- B. To enable communication from each API's Mule Runtimes and Network zone to the load balancer of the other API
- C. To open direct two-way communication between the Mule Runtimes of both API's
- D. To allow communication between load balancers used by each API

Answer: B

Explanation:

* If your API implementation involves putting a load balancer in front of your APIkit application, configure the load balancer to redirect URLs that reference the baseUrl of the application directly. If the load balancer does not redirect URLs, any calls that reach the load balancer looking for the application do not reach their destination.

* When you receive incoming traffic through the load balancer, the responses will go out the same way. However, traffic that is originating from your instance will not pass through the load balancer. Instead, it is sent directly from the public IP address of your instance out to the Internet. The ELB is not involved in that scenario.

* The question says "each API is deployed to multiple redundant Mule runtimes", that seems to be a hint for self hosted Mule runtime cluster. Set Inbound allowed for the LB, outbound allowed for runtime to request out.

* Hence correct way is to enable communication from each API's Mule Runtimes and Network zone to the load balancer of the other API. Because communication is asynchronous one

NEW QUESTION 19

An insurance company is implementing a MuleSoft API to get inventory details from the two vendors. Due to network issues, the invocations to vendor applications are getting timed-out intermittently. But the transactions are successful upon reprocessing

What is the most performant way of implementing this requirement?

- A. Implement a scatter-gather scope to invoke the two vendor applications on two different route Use the Until-Successful scope to implement the retry mechanism for timeout errors on each route
- B. Implement a Choice scope to invoke the two vendor applications on two different route Use the try-catch scope to implement the retry mechanism for timeout errors on each route
- C. Implement a For-Each scope to invoke the two vendor applications Use until successful scope to implement the retry mechanism for the timeout errors
- D. Implement Round-Robin scope to invoke the two vendor applications on two different routes Use the Try-Catch scope to implement retry mechanism for timeout errors on each route

Answer: A

NEW QUESTION 23

A company is planning to migrate its deployment environment from on-premises cluster to a Runtime Fabric (RTF) cluster. It also has a requirement to enable Mule

applications deployed to a Mule runtime instance to store and share data across application replicas and restarts.
 How can these requirements be met?

- A. Anypoint object store V2 to share data between replicas in the RTF cluster
- B. Install the object store pod on one of the cluster nodes
- C. Configure Persistence Gateway in any of the servers using Mule Object Store
- D. Configure Persistent Gateway at the RTF

Answer: D

NEW QUESTION 25

Organization wants to achieve high availability goal for Mule applications in customer hosted runtime plane. Due to the complexity involved, data cannot be shared among of different instances of same Mule application. What option best suits to this requirement considering high availability is very much critical to the organization?

- A. The cluster can be configured
- B. Use third party product to implement load balancer
- C. High availability can be achieved only in CloudHub
- D. Use persistent object store

Answer: B

Explanation:

High availability is about up-time of your application

A) High availability can be achieved only in CloudHub isn't correct statement. It can be achieved in customer hosted runtime planes as well

B) An object store is a facility for storing objects in or across Mule applications. Mule runtime engine (Mule) uses object stores to persist data for eventual retrieval. It can be used for disaster recovery but not for High Availability. Using object store can't guarantee that all instances won't go down at once. So not an appropriate choice.

NEW QUESTION 27

What is a key difference between synchronous and asynchronous logging from Mule applications?

- A. Synchronous logging writes log messages in a single logging thread but does not block the Mule event being processed by the next event processor
- B. Asynchronous logging can improve Mule event processing throughput while also reducing the processing time for each Mule event
- C. Asynchronous logging produces more reliable audit trails with more accurate timestamps
- D. Synchronous logging within an ongoing transaction writes log messages in the same thread that processes the current Mule event

Answer: B

Explanation:

Types of logging:

A) Synchronous: The execution of thread that is processing messages is interrupted to wait for the log message to be fully handled before it can continue.

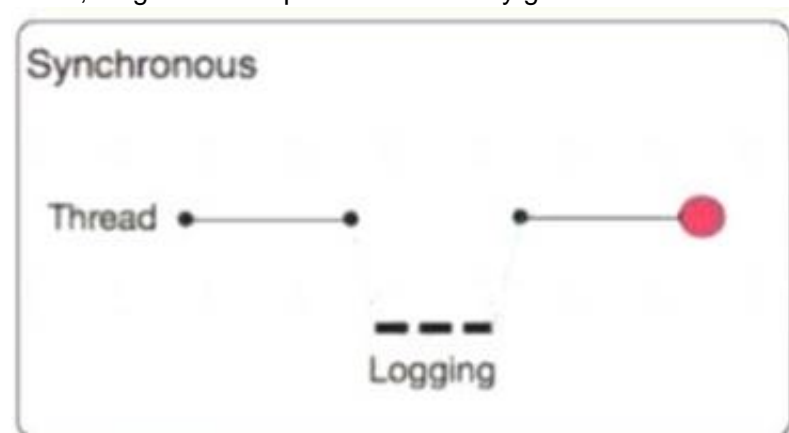
The execution of the thread that is processing your message is interrupted to wait for the log message to be fully output before it can continue

Performance degrades because of synchronous logging

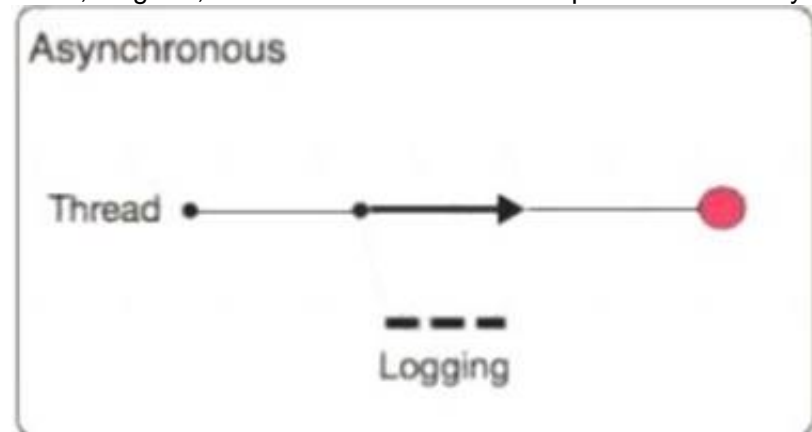
Used when the log is used as an audit trail or when logging ERROR/CRITICAL messages

If the logger fails to write to disk, the exception would raise on the same thread that's currently processing the Mule event. If logging is critical for you, then you can rollback the transaction.

Chart, diagram Description automatically generated



Chart, diagram, box and whisker chart Description automatically generated



B) Asynchronous:

The logging operation occurs in a separate thread, so the actual processing of your message won't be delayed to wait for the logging to complete

Substantial improvement in throughput and latency of message processing Mule runtime engine (Mule) 4 uses Log4j 2 asynchronous logging by default The disadvantage of asynchronous logging is error handling.

If the logger fails to write to disk, the thread doing the processing won't be aware of any issues writing to the disk, so you won't be able to rollback anything.

Because the actual writing of the log gets deferred, there's a chance that log messages might never make it to disk and get lost, if Mule were to crash before the

buffers are flushed.

So Correct answer is: Asynchronous logging can improve Mule event processing throughput while also reducing the processing time for each Mule event

NEW QUESTION 29

An organization is designing a Mule application to periodically poll an SFTP location for new files containing sales order records and then process those sales orders. Each sales order must be processed exactly once.

To support this requirement, the Mule application must identify and filter duplicate sales orders on the basis of a unique ID contained in each sales order record and then only send the new sales orders to the downstream system.

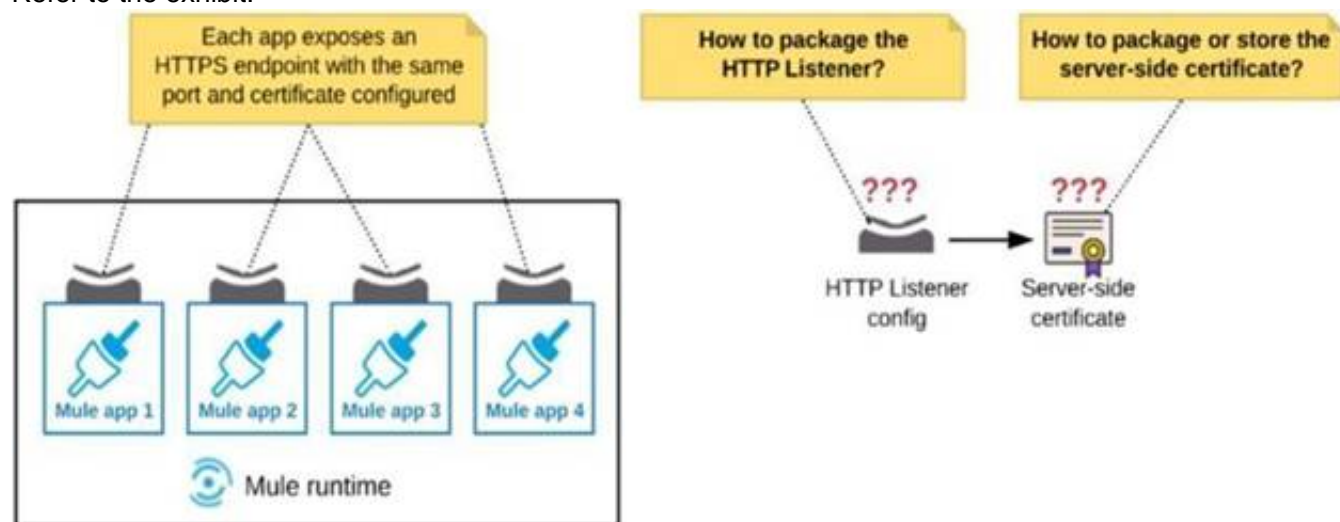
What is the most idiomatic (used for its intended purpose) Anypoint connector, validator, or scope that can be configured in the Mule application to filter duplicate sales orders on the basis of the unique ID field contained in each sales order record?

- A. Configure a Cache scope to filter and store each record from the received file by the order ID
- B. Configure a Database connector to filter and store each record by the order ID
- C. Configure an Idempotent Message Validator component to filter each record by the order ID
- D. Configure a watermark In an On New or Updated File event source to filter unique records by the order ID

Answer: C

NEW QUESTION 32

Refer to the exhibit.



An organization deploys multiple Mule applications to the same customer -hosted Mule runtime. Many of these Mule applications must expose an HTTPS endpoint on the same port using a server-side certificate that rotates often.

What is the most effective way to package the HTTP Listener and package or store the server-side certificate when deploying these Mule applications, so the disruption caused by certificate rotation is minimized?

- A. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint Package the server-side certificate in ALL Mule APPLICATIONS that need to expose an HTTPS endpoint
- B. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint
- C. Store the server-side certificate in a shared filesystem location in the Mule runtime's classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION
- D. Package an HTTPS Listener configuration In all Mule APPLICATIONS that need to expose an HTTPS endpoint Package the server-side certificate in a NEW Mule DOMAIN project
- E. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing It from all Mule applications that need to expose an HTTPS endpoint
- F. Package the server-side certificate in the SAME Mule DOMAIN project Go to Set

Answer: B

Explanation:

In this scenario, both A & C will work, but A is better as it does not require repackaging to the domain project at all.

Correct answer is Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint. Store the server-side certificate in a shared filesystem location in the Mule runtime's classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION.

What is Mule Domain Project?

* A Mule Domain Project is implemented to configure the resources that are shared among different projects. These resources can be used by all the projects associated with this domain. Mule applications can be associated with only one domain, but a domain can be associated with multiple projects. Shared resources allow multiple development teams to work in parallel using the same set of reusable connectors. Defining these connectors as shared resources at the domain level allows the team to:

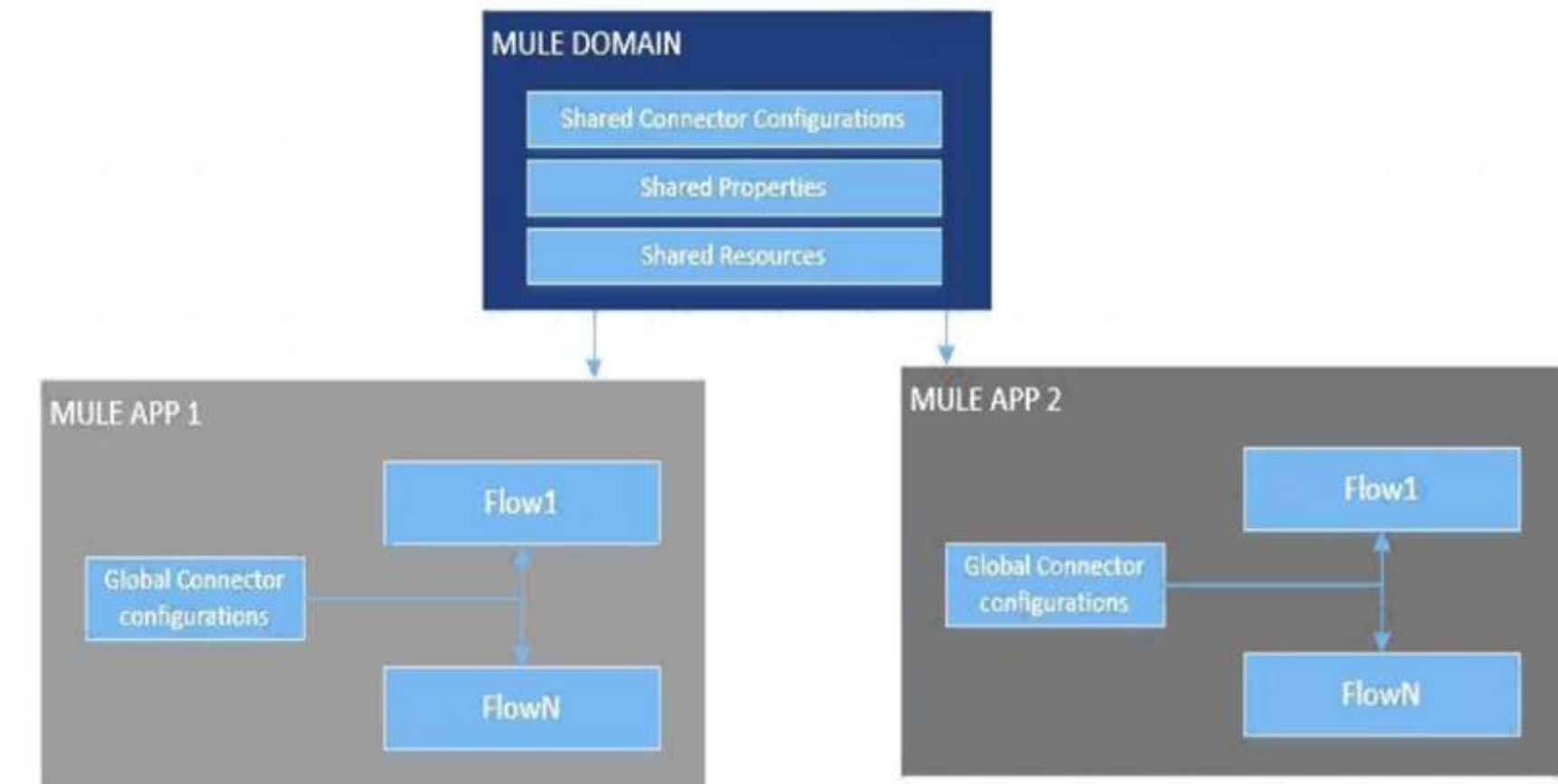
- Expose multiple services within the domain through the same port.
- Share the connection to persistent storage.
- Share services between apps through a well-defined interface.
- Ensure consistency between apps upon any changes because the configuration is only set in one place.

* Use domains Project to share the same host and port among multiple projects. You can declare the http connector within a domain project and associate the domain project with other projects. Doing this also allows to control thread settings, keystore configurations, time outs for all the requests made within multiple applications. You may think that one can also achieve this by duplicating the http connector configuration across all the applications. But, doing this may pose a nightmare if you have to make a change and redeploy all the applications.

* If you use connector configuration in the domain and let all the applications use the new domain instead of a default domain, you will maintain only one copy of the http connector configuration. Any changes will require only the domain to be redeployed instead of all the applications.

You can start using domains in only three steps:

- 1) Create a Mule Domain project
- 2) Create the global connector configurations which needs to be shared across the applications inside the Mule Domain project
- 3) Modify the value of domain in mule-deploy.properties file of the applications Graphical user interface Description automatically generated



Use a certificate defined in already deployed Mule domain Configure the certificate in the domain so that the API proxy HTTPS Listener references it, and then deploy the secure API proxy to the target Runtime Fabric, or on-premises target. (CloudHub is not supported with this approach because it does not support Mule domains.)

NEW QUESTION 37

An organization uses a set of customer-hosted Mule runtimes that are managed using the Mulesoft-hosted control plane. What is a condition that can be alerted on from Anypoint Runtime Manager without any custom components or custom coding?

- A. When a Mule runtime on a given customer-hosted server is experiencing high memory consumption during certain periods
- B. When an SSL certificate used by one of the deployed Mule applications is about to expire
- C. When the Mule runtime license installed on a Mule runtime is about to expire
- D. When a Mule runtime's customer-hosted server is about to run out of disk space

Answer: A

Explanation:

Correct answer is When a Mule runtime on a given customer-hosted server is experiencing high memory consumption during certain periods Using Anypoint Monitoring, you can configure two different types of alerts: Basic alerts for servers and Mule apps Limit per organization: Up to 50 basic alerts for users who do not have a Titanium subscription to Anypoint Platform You can set up basic alerts to trigger email notifications when a metric you are measuring passes a specified threshold. You can create basic alerts for the following metrics for servers or Mule apps: For on-premises servers and CloudHub apps: * CPU utilization * Memory utilization * Thread count Advanced alerts for graphs in custom dashboards in Anypoint Monitoring. You must have a Titanium subscription to use this feature. Limit per organization: Up to 20 advanced alerts

NEW QUESTION 38

A Mule application is being designed to do the following:

Step 1: Read a SalesOrder message from a JMS queue, where each SalesOrder consists of a header and a list of SalesOrderLineItems.

Step 2: Insert the SalesOrder header and each SalesOrderLineItem into different tables in an RDBMS.

Step 3: Insert the SalesOrder header and the sum of the prices of all its SalesOrderLineItems into a table in a different RDBMS.

No SalesOrder message can be lost and the consistency of all SalesOrder-related information in both RDBMSs must be ensured at all times.

What design choice (including choice of transactions) and order of steps addresses these requirements?

- A. 1) Read the JMS message (NOT in an XA transaction)2) Perform BOTH DB inserts in ONE DB transaction3) Acknowledge the JMS message
- B. 1) Read the JMS message (NOT in an XA transaction)2) Perform EACH DB insert in a SEPARATE DB transaction3) Acknowledge the JMS message
- C. 1) Read the JMS message in an XA transaction2) In the SAME XA transaction, perform BOTH DB inserts but do NOT acknowledge the JMS message
- D. 1) Read and acknowledge the JMS message (NOT in an XA transaction)2) In a NEW XA transaction, perform BOTH DB inserts

Answer: A

Explanation:

Option A says "Perform EACH DB insert in a SEPARATE DB transaction". In this case if first DB insert is successful and second one fails then first insert won't be rolled back causing inconsistency. This option is ruled out.

Option D says Perform BOTH DB inserts in ONE DB transaction.

Rule of thumb is when one or more DB connections are required we must use XA transaction as local transactions support only one resource. So this option is also ruled out.

Option B acknowledges the before DB processing, so message is removed from the queue. In case of system failure at later point, message can't be retrieved.

Option C is Valid: Though it says "do not ack JMS message", message will be auto acknowledged at the end of transaction. Here is how we can ensure all components are part of XA transaction: <https://docs.mulesoft.com/jms-connector/1.7/jms-transactions>

Additional Information about transactions:

XA Transactions - You can use an XA transaction to group together a series of operations from multiple transactional resources, such as JMS, VM or JDBC resources, into a single, very reliable, global transaction.

The XA (eXtended Architecture) standard is an X/Open group standard which specifies the interface between a global transaction manager and local transactional resource managers.

The XA protocol defines a 2-phase commit protocol which can be used to more reliably coordinate and sequence a series of "all or nothing" operations across multiple servers, even servers of different types

Use JMS ack if

- Acknowledgment should occur eventually, perhaps asynchronously
- The performance of the message receipt is paramount
- The message processing is idempotent
- For the choreography portion of the SAGA pattern Use JMS transactions
- For all other times in the integration you want to perform an atomic unit of work
- When the unit of work comprises more than the receipt of a single message
- To simply and unify the programming model (begin/commit/rollback)

NEW QUESTION 40

An organization is using Mulesoft cloudhub and develops API's in the latest version. As a part of requirements for one of the API's, third party API needs to be called. The security team has made it clear that calling any external API needs to have include listing

As an integration architect please suggest the best way to accomplish the design plan to support these requirements?

- A. Implement includelist IP on the cloudhub VPC firewall to allow the traffic
- B. Implement the validation of includelisted IP operation
- C. Implement the Any point filter processor to implement the include list IP
- D. Implement a proxy for the third party API and enforce the IPinclude list policy and call this proxy from the flow of the API

Answer: D

NEW QUESTION 44

As a part of project requirement, client will send a stream of data to mule application. Payload size can vary between 10mb to 5GB. Mule application is required to transform the data and send across multiple sftp servers. Due to the cost cuttings in the organization, mule application can only be allocated one worker with size of 0.2 vCore.

As an integration architect , which streaming strategy you would suggest to handle this scenario?

- A. In-memory non repeatable stream
- B. File based non-repeatable stream
- C. In-memory repeatable stream
- D. File based repeatable storage

Answer: D

Explanation:

As the question says that data needs to be sent across multiple sftp serves , we cannot use non-repeatable streams. The non-repeatable strategy disables repeatable streams, which enables you to read an input stream only once.

You cant use in memory storage because with 0.2 vcore you will get only 1 GB of heap memory. Hence application will error out for file more than 1 GB.

Hence the correct option is file base repeatable stream

NEW QUESTION 48

A mule application uses an HTTP request operation to involve an external API. The external API follows the HTTP specification for proper status code usage.

What is possible cause when a 3xx status code is returned to the HTTP Request operation from the external API?

- A. The request was not accepted by the external API
- B. The request was Redirected to a different URL by the external API
- C. The request was NOT RECEIVED by the external API
- D. The request was ACCEPTED by the external API

Answer: B

Explanation:

3xx HTTP status codes indicate a redirection that the user agent (a web browser or a crawler) needs to take further action when trying to access a particular resource.

NEW QUESTION 51

What is maximum vCores can be allocated to application deployed to CloudHub?

- A. 1 vCores
- B. 2 vCores
- C. 4 vCores
- D. 16 vCores

Answer: D

NEW QUESTION 55

As an enterprise architect, what are the two reasons for which you would use a canonical data model in the new integration project using Mulesoft Anypoint platform (choose two answers)

- A. To have consistent data structure aligned in processes
- B. To isolate areas within a bounded context
- C. To incorporate industry standard data formats
- D. There are multiple canonical definitions of each data type
- E. Because the model isolates the back and systems and support mule applications from change

Answer: AB

NEW QUESTION 60

An organization is designing multiple new applications to run on CloudHub in a single Anypoint VPC and that must share data using a common persistent Anypoint object store V2 (OSv2).

Which design gives these mule applications access to the same object store instance?

- A. AVM connector configured to directly access the persistence queue of the persistent object store
- B. An Anypoint MQ connector configured to directly access the persistent object store
- C. Object store V2 can be shared across cloudhub applications with the configured osv2 connector
- D. The object store V2 rest API configured to access the persistent object store

Answer: D

NEW QUESTION 62

An automation engineer needs to write scripts to automate the steps of the API lifecycle, including steps to create, publish, deploy and manage APIs and their implementations in Anypoint Platform.

What Anypoint Platform feature can be used to automate the execution of all these actions in scripts in the easiest way without needing to directly invoke the Anypoint Platform REST APIs?

- A. Automated Policies in API Manager
- B. Runtime Manager agent
- C. The Mule Maven Plugin
- D. Anypoint CLI

Answer: D

Explanation:

Anypoint Platform provides a scripting and command-line tool for both Anypoint Platform and Anypoint Platform Private Cloud Edition (Anypoint Platform PCE). The command-line interface (CLI) supports both the interactive shell and standard CLI modes and works with: Anypoint Exchange Access management Anypoint Runtime Manager

NEW QUESTION 64

An organization has an HTTPS-enabled Mule application named Orders API that receives requests from another Mule application named Process Orders. The communication between these two Mule applications must be secured by TLS mutual authentication (two-way TLS).

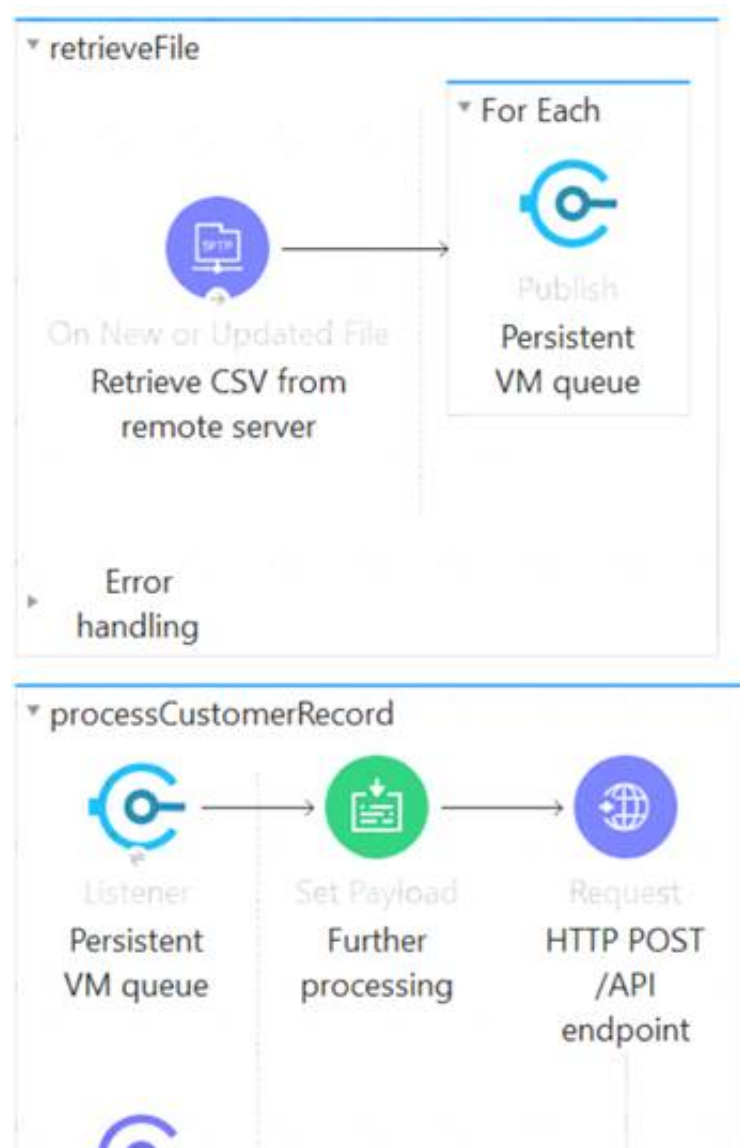
At a minimum, what must be stored in each truststore and keystore of these two Mule applications to properly support two-way TLS between the two Mule applications while properly protecting each Mule application's keys?

- A. Orders API truststore: The Orders API public keyProcess Orders keystore: The Process Orders private key and public key
- B. Orders API truststore: The Orders API private key and public key Process Orders keystore: The Process Orders private key public key
- C. Orders API truststore: The Process Orders public keyOrders API keystore: The Orders API private key and public key Process Orders truststore: The Orders API public keyProcess Orders keystore: The Process Orders private key and public key
- D. Orders API truststore: The Process Orders public key Orders API keystore: The Orders API private key Process Orders truststore: The Orders API public key Process Orders keystore: The Process Orders private key

Answer: C

NEW QUESTION 65

Refer to the exhibit.



This Mule application is deployed to multiple Cloudhub workers with persistent queue enabled. The retrievefile flow event source reads a CSV file from a remote SFTP server and then publishes each record in the CSV file to a VM queue. The processCustomerRecords flow's VM Listener receives messages from the same VM queue and then processes each message separately.

How are messages routed to the cloudhub workers as messages are received by the VM Listener?

- A. Each message is routed to ONE of the Cloudhub workers in a DETERMINISTIC round robin fashion thereby EXACTLY BALANCING messages among the cloudhub workers
- B. Each messages routes to ONE of the available Clouhub workers in a NON- DETERMINISTIC non round-robin fashion thereby APPROXIMATELY BALANCING messages among the cloudhub workers
- C. Each message is routed to the SAME Cloudhub worker that retrieved the file, thereby BINDING ALLmessages to ONLY that ONE Cloudhub worker
- D. Each message is duplicated to ALL of the Cloudhub workers, thereby SHARING EACH message with ALL the Cloudhub workers.

Answer: B

NEW QUESTION 69

An API has been unit tested and is ready for integration testing. The API is governed by a Client ID Enforcement policy in all environments. What must the testing team do before they can start integration testing the API in the Staging environment?

- A. They must access the API portal and create an API notebook using the Client ID and Client Secret supplied by the API portal in the Staging environment
- B. They must request access to the API instance in the Staging environment and obtain a Client ID and Client Secret to be used for testing the API
- C. They must be assigned as an API version owner of the API in the Staging environment
- D. They must request access to the Staging environment and obtain the Client ID and Client Secret for that environment to be used for testing the API

Answer: B

Explanation:

- * It's mentioned that the API is governed by a Client ID Enforcement policy in all environments.
- * Client ID Enforcement policy allows only authorized applications to access the deployed API implementation.
- * Each authorized application is configured with credentials: client_id and client_secret.
- * At runtime, authorized applications provide the credentials with each request to the API implementation. MuleSoft Reference: <https://docs.mulesoft.com/api-manager/2.x/policy-mule3-client-id-based-policies>

NEW QUESTION 73

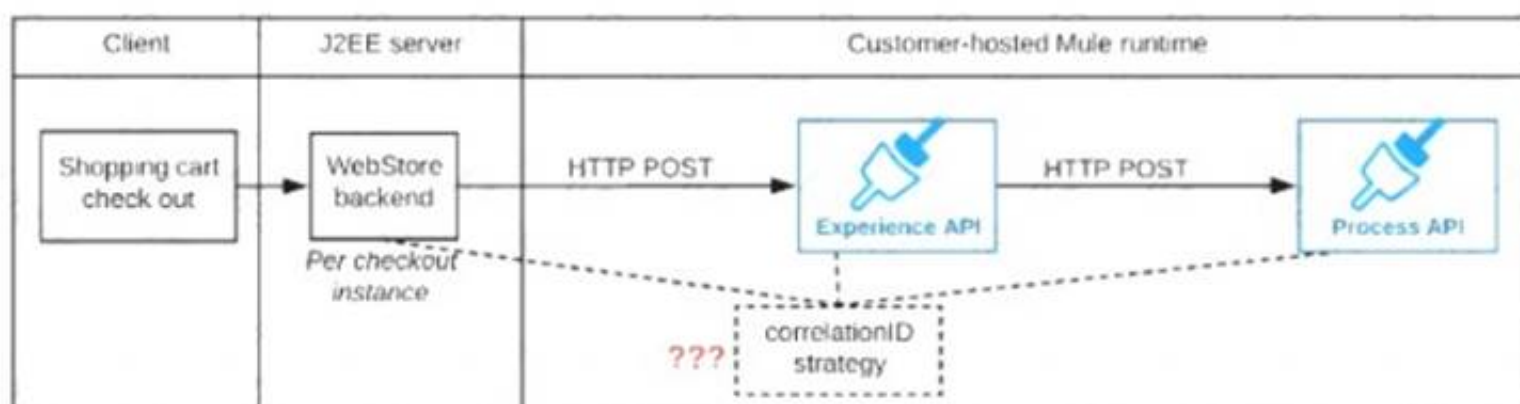
As a part of project requirement, Java Invoke static connector in a mule 4 application needs to invoke a static method in a dependency jar file. What are two ways to add the dependency to be visible by the connectors class loader?
 (Choose two answers)

- A. In the Java Invoke static connector configuration, configure a path and name of the dependency jar file
- B. Add the dependency jar file to the java classpath by setting the JVM parameters
- C. Use Maven command to include the dependency jar file when packaging the application
- D. Configure the dependency as a shared library in the project POM
- E. Update mule-artefact.json to export the Java package

Answer: BD

NEW QUESTION 75

Refer to the exhibit.



A shopping cart checkout process consists of a web store backend sending a sequence of API invocations to an Experience API, which in turn invokes a Process API. All API invocations are over HTTPS POST. The Java web store backend executes in a Java EE application server, while all API implementations are Mule applications executing in a customer -hosted Mule runtime.

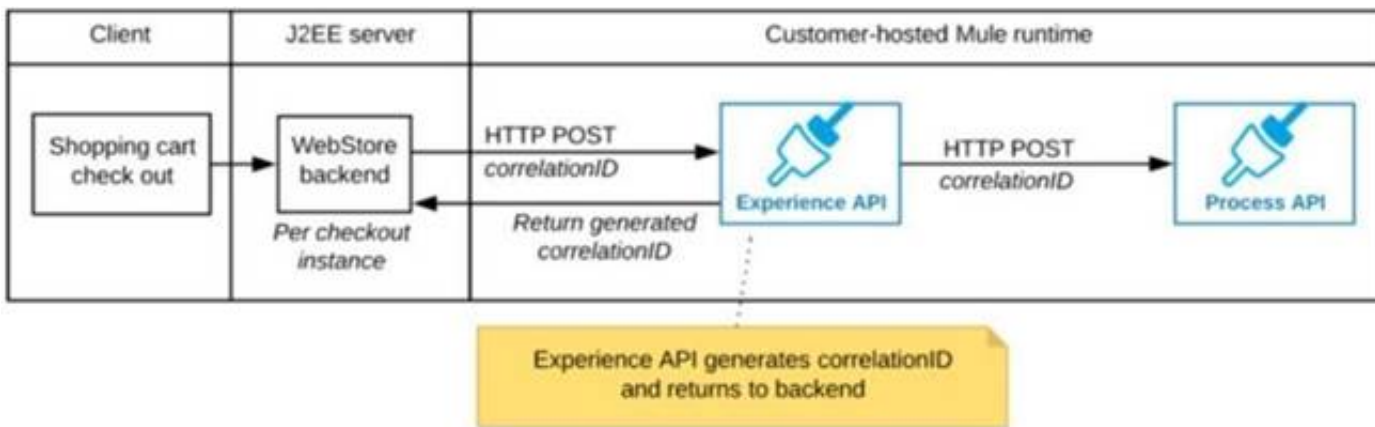
End-to-end correlation of all HTTP requests and responses belonging to each individual checkout Instance is required. This is to be done through a common correlation ID, so that all log entries written by the web store backend, Experience API implementation, and Process API implementation include the same correlation ID for all requests and responses belonging to the same checkout instance.

What is the most efficient way (using the least amount of custom coding or configuration) for the web store backend and the implementations of the Experience API and Process API to participate in end-to-end correlation of the API invocations for each checkout instance?

A)

The web store backend, being a Java EE application, automatically makes use of the thread-local correlation ID generated by the Java EE application server and automatically transmits that to the Experience API using HTTP-standard headers

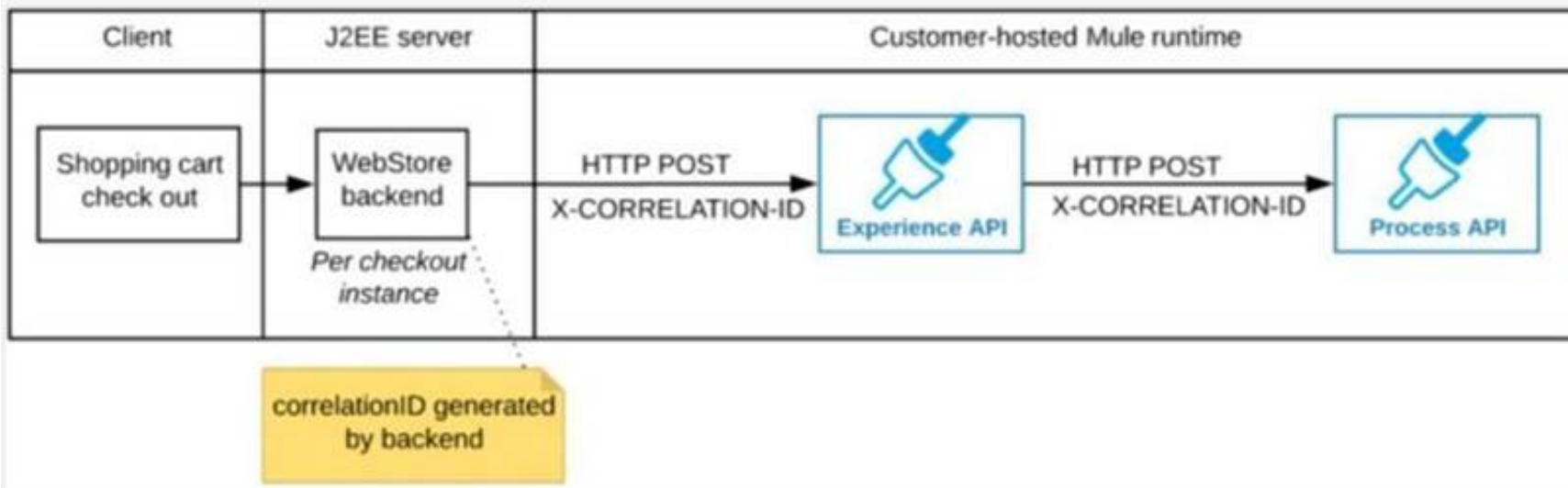
No special code or configuration is included in the web store backend, Experience API, and Process API implementations to generate and manage the correlation ID



B)

The web store backend generates a new correlation ID value at the start of checkout and sets it on the X-CORRELATION-Id HTTP request header In each API invocation belonging to that checkout

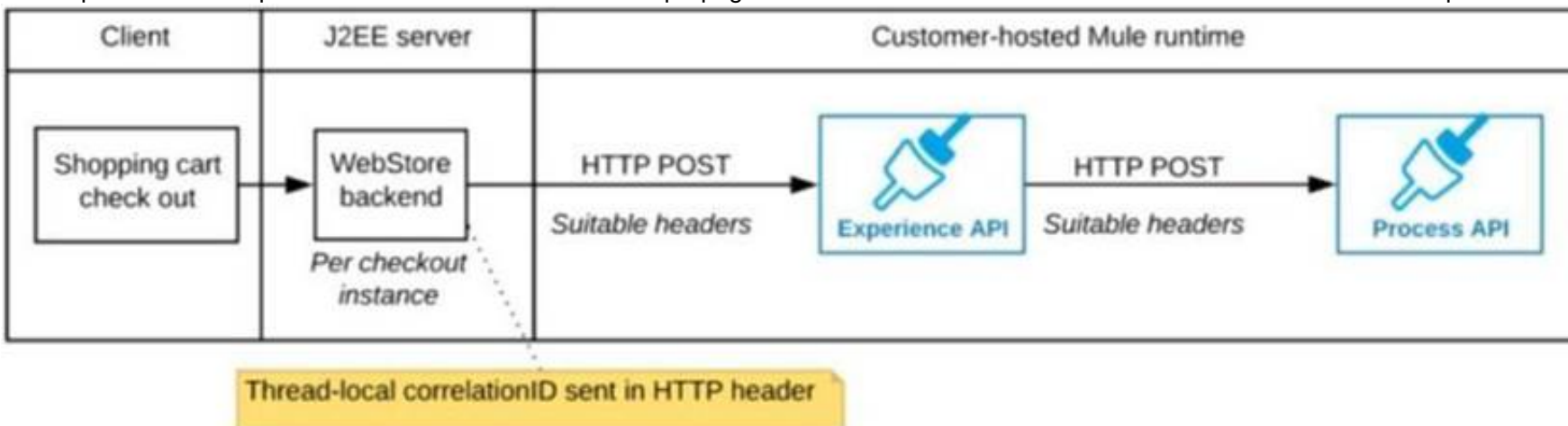
No special code or configuration is included in the Experience API and Process API implementations to generate and manage the correlation ID



C)

The Experience API implementation generates a correlation ID for each incoming HTTP request and passes it to the web store backend in the HTTP response, which includes it in all subsequent API invocations to the Experience API.

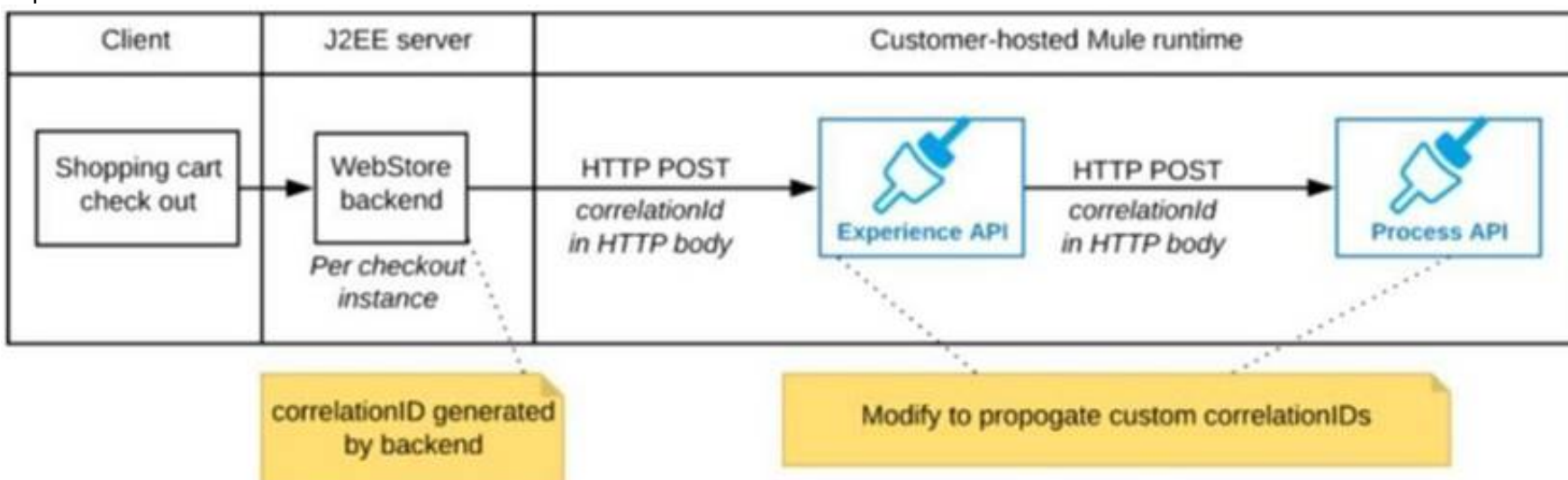
The Experience API implementation must be coded to also propagate the correlation ID to the Process API in a suitable HTTP request header



D)

The web store backend sends a correlation ID value in the HTTP request body In the way required by the Experience API

The Experience API and Process API implementations must be coded to receive the custom correlation ID In the HTTP requests and propagate It in suitable HTTP request headers



A. Option A

B. Option B

C. Option C

D. Option D

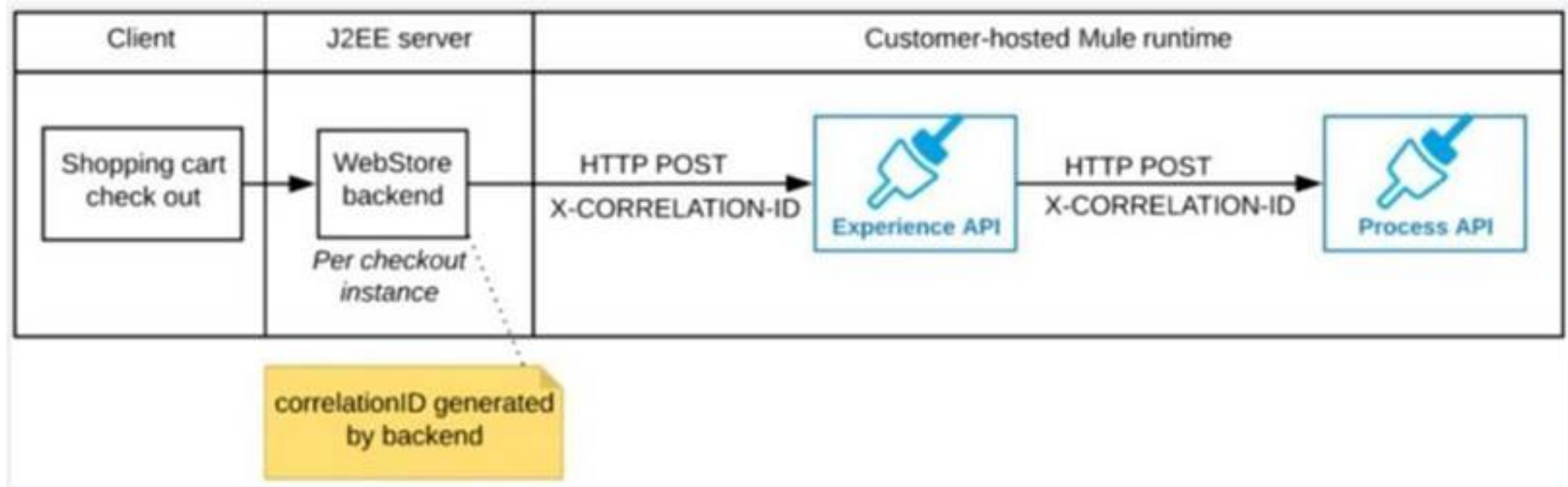
Answer: B

Explanation:

: By design, Correlation Ids cannot be changed within a flow in Mule 4 applications and can be set only at source. This ID is part of the Event Context and is generated as soon as the message is received by the application. When a HTTP Request is received, the request is inspected for "X-Correlation-Id" header. If "X-

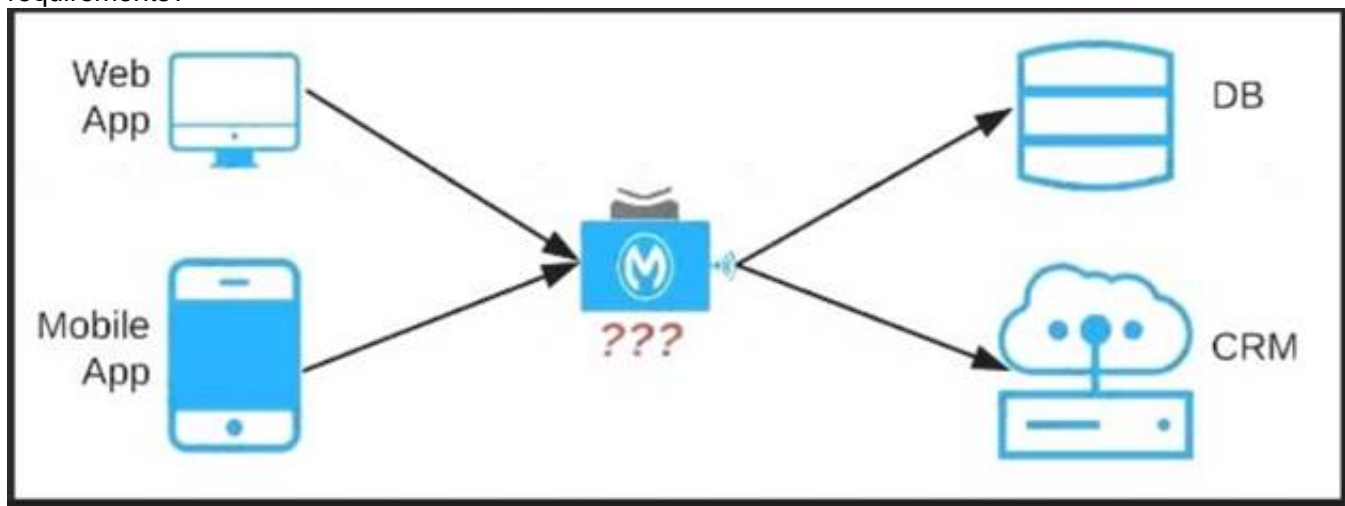
Correlation-Id" header is present, HTTP connector uses this as the Correlation Id. If "X-Correlation-Id" header is NOT present, a Correlation Id is randomly generated. For Incoming HTTP Requests: In order to set a custom Correlation Id, the client invoking the HTTP request must set "X-Correlation-Id" header. This will ensure that the Mule Flow uses this Correlation Id. For Outgoing HTTP Requests: You can also propagate the existing Correlation Id to downstream APIs. By default, all outgoing HTTP Requests send "X-Correlation-Id" header. However, you can choose to set a different value to "X-Correlation-Id" header or set "Send Correlation Id" to NEVER.

Mulesoft Reference:
<https://help.mulesoft.com/s/article/How-to-Set-Custom-Correlation-Id-for-Flows-with-HTTP-Endpoint-in-Mule>
 Graphical user interface, application, Word Description automatically generated



NEW QUESTION 79

An organization needs to enable access to their customer data from both a mobile app and a web application, which each need access to common fields as well as certain unique fields. The data is available partially in a database and partially in a 3rd-party CRM system. What APIs should be created to best fit these design requirements?



- A. A Process API that contains the data required by both the web and mobile apps, allowing these applications to invoke it directly and access the data they need thereby providing the flexibility to add more fields in the future without needing API changes.
- B. One set of APIs (Experience API, Process API, and System API) for the web app, and another set for the mobile app.
- C. Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system
- D. A common Experience API used by both the web and mobile apps, but separate Process APIs for the web and mobile apps that interact with the database and the CRM System.

Answer: C

Explanation:

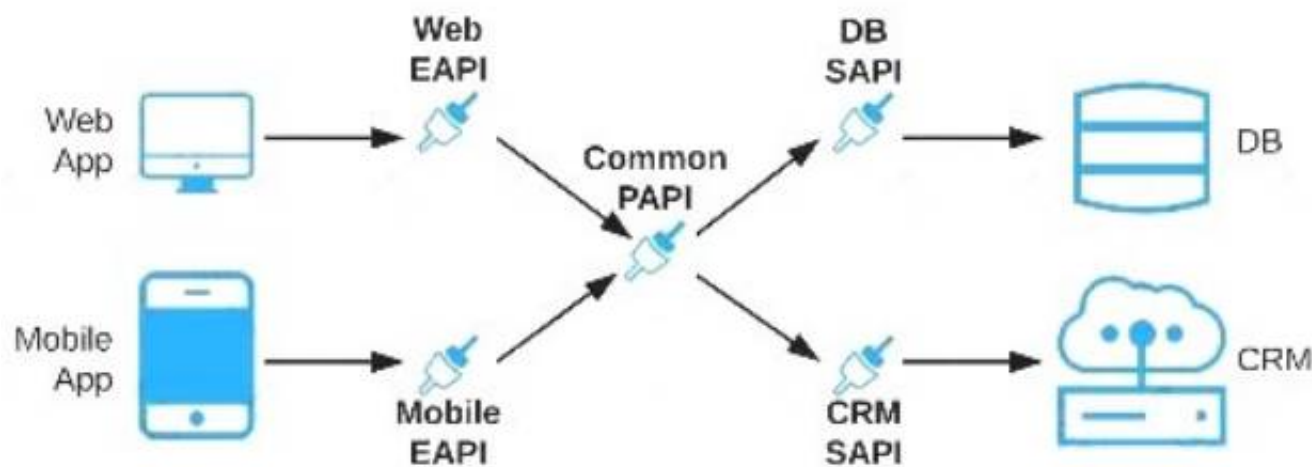
Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements which cannot be fulfilled by single experience layer API

Option : Common Process API Analysis : This solution will not work because creating a common process API will impose limitations in terms of flexibility to customize API;s as per the requirements of different applications. It is not a recommended approach.

Option : Separate set of API's for both the applications Analysis : This goes against the principle of Anypoint API-led connectivity approach which promotes creating reusable assets. This solution may work but this is not efficient solution and creates duplicity of code.

Hence the correct answer is: Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system

A screenshot of a computer Description automatically generated with low confidence



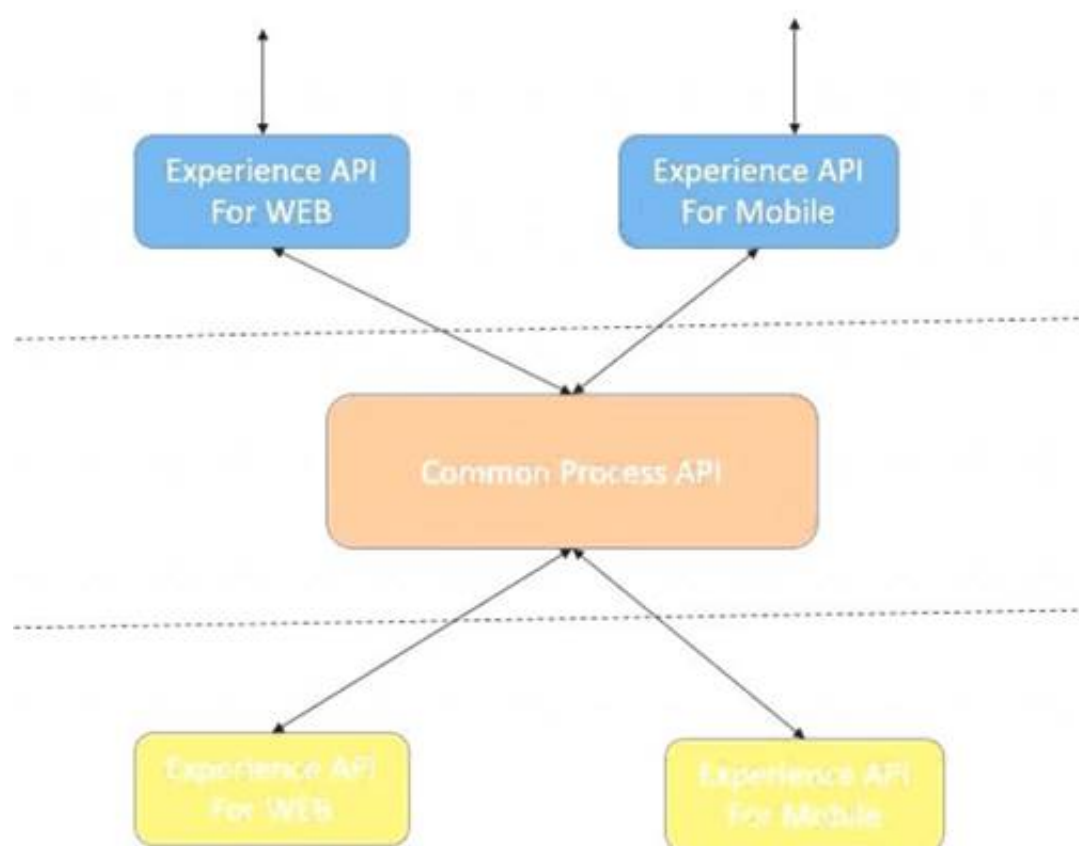
Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements which cannot be fulfilled by single experience layer API

Option : Common Process API Analysis : This solution will not work because creating a common process API will impose limitations in terms of flexibility to customize API;s as per the requirements of different applications. It is not a recommended approach.

Option : Separate set of API's for both the applications Analysis : This goes against the principle of Anypoint API-led connectivity approach which promotes creating reusable assets. This solution may work but this is not efficient solution and creates duplicity of code.

Hence the correct answer is: Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system

Diagram Description automatically generated



NEW QUESTION 81

In Anypoint Platform, a company wants to configure multiple identity providers (IdPs) for multiple lines of business (LOBs). Multiple business groups, teams, and environments have been defined for these LOBs.

What Anypoint Platform feature can use multiple IdPs across the company's business groups, teams, and environments?

- A. MuleSoft-hosted (CloudHub) dedicated load balancers
- B. Client (application) management
- C. Virtual private clouds
- D. Permissions

Answer: A

Explanation:

To use a dedicated load balancer in your environment, you must first create an Anypoint VPC. Because you can associate multiple environments with the same Anypoint VPC, you can use the same dedicated load balancer for your different environments.

NEW QUESTION 84

An XA transaction is being configured that involves a JMS connector listening for Incoming JMS messages. What is the meaning of the timeout attribute of the XA transaction, and what happens after the timeout expires?

- A. The time that is allowed to pass between committing the transaction and the completion of the Mule flow After the timeout, flow processing triggers an error
- B. The time that is allowed to pass between receiving JMS messages on the same JMS connection After the timeout, a new JMS connection is established
- C. The time that is allowed to pass without the transaction being ended explicitly After the timeout, the transaction is forcefully rolled-back
- D. The time that is allowed to pass for state JMS consumer threads to be destroyed After the timeout, a new JMS consumer thread is created

Answer: C

Explanation:

* Setting a transaction timeout for the Bitronix transaction manager Set the transaction timeout either

- In wrapper.conf
 - In CloudHub in the Properties tab of the Mule application deployment The default is 60 secs. It is defined as mule.bitronix.transactiontimeout = 120
- * This property defines the timeout for each transaction created for this manager.
If the transaction has not terminated before the timeout expires it will be automatically rolled back.

Additional Info around Transaction Management:

Bitronix is available as the XA transaction manager for Mule applications

To use Bitronix, declare it as a global configuration element in the Mule application

<bti:transaction-manager />

Each Mule runtime can have only one instance of a Bitronix transaction manager, which is shared by all Mule applications

For customer-hosted deployments, define the XA transaction manager in a Mule domain

- Then share this global element among all Mule applications in the Mule runtime Graphical user interface, table Description automatically generated with medium confidence

Transaction Management		
Characteristics	Local Transactions	Extended Architecture (XA) Transactions
Connector Requisite 1	All operations inside the transaction must belong to same Connector.	Operations inside the transaction may belong to different Connectors.
Connector Requisite 2	Connectors may not be XA enabled	Connectors must be XA enabled
Connector Requisite 3	Connectors should use single config reference	Connectors may use multiple config references
Available resources	JMS, VM, JDBC	JMS, VM, JDBC
Uses Two Phase Commit (2PC)	No	Yes
DB Operations	Perform database operation to only one database resource	Perform database operation to one or more transactional resource.
Supports Nested Transactions	Does not support nested transactions.	Supports nested transactions.
Bitronix is available	No	Yes
A.C.I.D Properties	No	Yes
Performance	Better than XA	Latency Increases
Thread Pooling	BLOCKING_IO	BLOCKING_IO
Recovery is cause of system failure	No	Using Bitronix

NEW QUESTION 86

When designing an upstream API and its implementation, the development team has been advised to not set timeouts when invoking downstream API. Because the downstream API has no SLA that can be relied upon. This is the only downstream API dependency of that upstream API. Assume the downstream API runs uninterrupted without crashing. What is the impact of this advice?

- A. The invocation of the downstream API will run to completion without timing out.
- B. An SLA for the upstream API CANNOT be provided.
- C. A default timeout of 500 ms will automatically be applied by the Mule runtime in which the upstream API implementation executes.
- D. A load-dependent timeout of less than 1000 ms will be applied by the Mule runtime in which the downstream API implementation executes.

Answer: B

Explanation:

An SLA for the upstream API CANNOT be provided.

NEW QUESTION 91

An organization is creating a Mule application that will be deployed to CloudHub. The Mule application has a property named dbPassword that stores a database user's password.

The organization's security standards indicate that the dbPassword property must be hidden from every

Anypoint Platform user after the value is set in the Runtime Manager Properties tab.

What configuration in the Mule application helps hide the dbPassword property value in Runtime Manager?

- A. Use secure::dbPassword as the property placeholder name and store the cleartext (unencrypted) value in a secure properties placeholder file
- B. Use secure::dbPassword as the property placeholder name and store the property encrypted value in a secure properties placeholder file
- C. Add the dbPassword property to the secureProperties section of the pom.xml file
- D. Add the dbPassword property to the secureProperties section of the mule-artifact.json file

Answer: B

NEW QUESTION 94

A company is planning to extend its Mule APIs to the Europe region. Currently all new applications are deployed to Cloudhub in the US region following this

naming convention

{API name}-{environment}. for example, Orders-SAPI-dev, Orders-SAPI-prod etc.

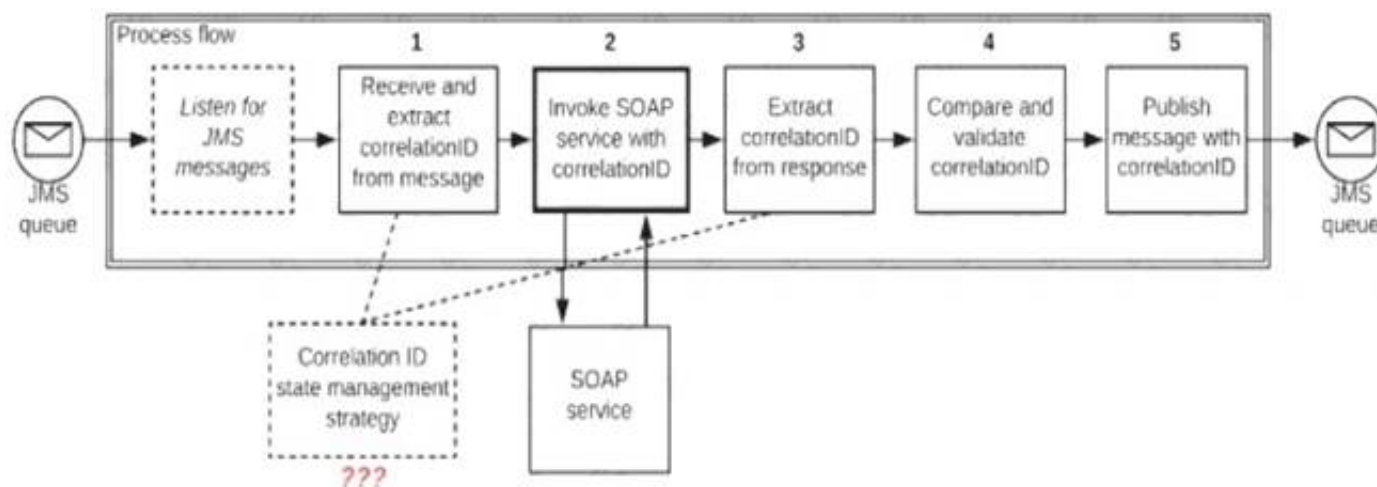
Considering there is no network restriction to block communications between API's, what strategy should be implemented in order to apply the same new API's running in the EU region of CloudHub as well to minimize latency between API's and target users and systems in Europe?

- A. Set region property to Europe (eu-de) in API manager for all the mule application No need to change the naming convention
- B. Set region property to Europe (eu-de) in API manager for all the mule application Change the naming convention to {API name}-{environment}-{region} and communicate this change to the consuming applications and users
- C. Set region property to Europe (eu-de) in runtime manager for all the mule application No need to change the naming convention
- D. Set region property to Europe (eu-de) in runtime manager for all the mule application Change the naming convention to {API name}-{environment}-{region} and communicate this change to the consuming applications and users

Answer: D

NEW QUESTION 96

Refer to the exhibit.



A Mule application is deployed to a multi-node Mule runtime cluster. The Mule application uses the competing consumer pattern among its cluster replicas to receive JMS messages from a JMS queue. To process each received JMS message, the following steps are performed in a flow:

Step 1: The JMS Correlation ID header is read from the received JMS message.

Step 2: The Mule application invokes an idempotent SOAP webservice over HTTPS, passing the JMS Correlation ID as one parameter in the SOAP request.

Step 3: The response from the SOAP webservice also returns the same JMS Correlation ID.

Step 4: The JMS Correlation ID received from the SOAP webservice is validated to be identical to the JMS Correlation ID received in Step 1.

Step 5: The Mule application creates a response JMS message, setting the JMS Correlation ID message header to the validated JMS Correlation ID and publishes that message to a response JMS queue.

Where should the Mule application store the JMS Correlation ID values received in Step 1 and Step 3 so that the validation in Step 4 can be performed, while also making the overall Mule application highly available, fault-tolerant, performant, and maintainable?

- A. Both Correlation ID values should be stored in a persistent object store
- B. Both Correlation ID values should be stored In a non-persistent object store
- C. The Correlation ID value in Step 1 should be stored in a persistent object storeThe Correlation ID value in step 3 should be stored as a Mule event variable/attribute
- D. Both Correlation ID values should be stored as Mule event variable/attribute

Answer: C

Explanation:

* If we store Correlation id value in step 1 as Mule event variables/attributes, the values will be cleared after server restart and we want system to be fault tolerant.

* The Correlation ID value in Step 1 should be stored in a persistent object store.

* We don't need to store Correlation ID value in Step 3 to persistent object store. We can store it but as we also need to make application performant. We can avoid this step of accessing persistent object store.

* Accessing persistent object stores slow down the performance as persistent object stores are by default stored in shared file systems.

* As the SOAP service is idempotent in nature. In case of any failures , using this Correlation ID saved in first step we can make call to SOAP service and validate the Correlation ID.

Top of Form

Additional Information:

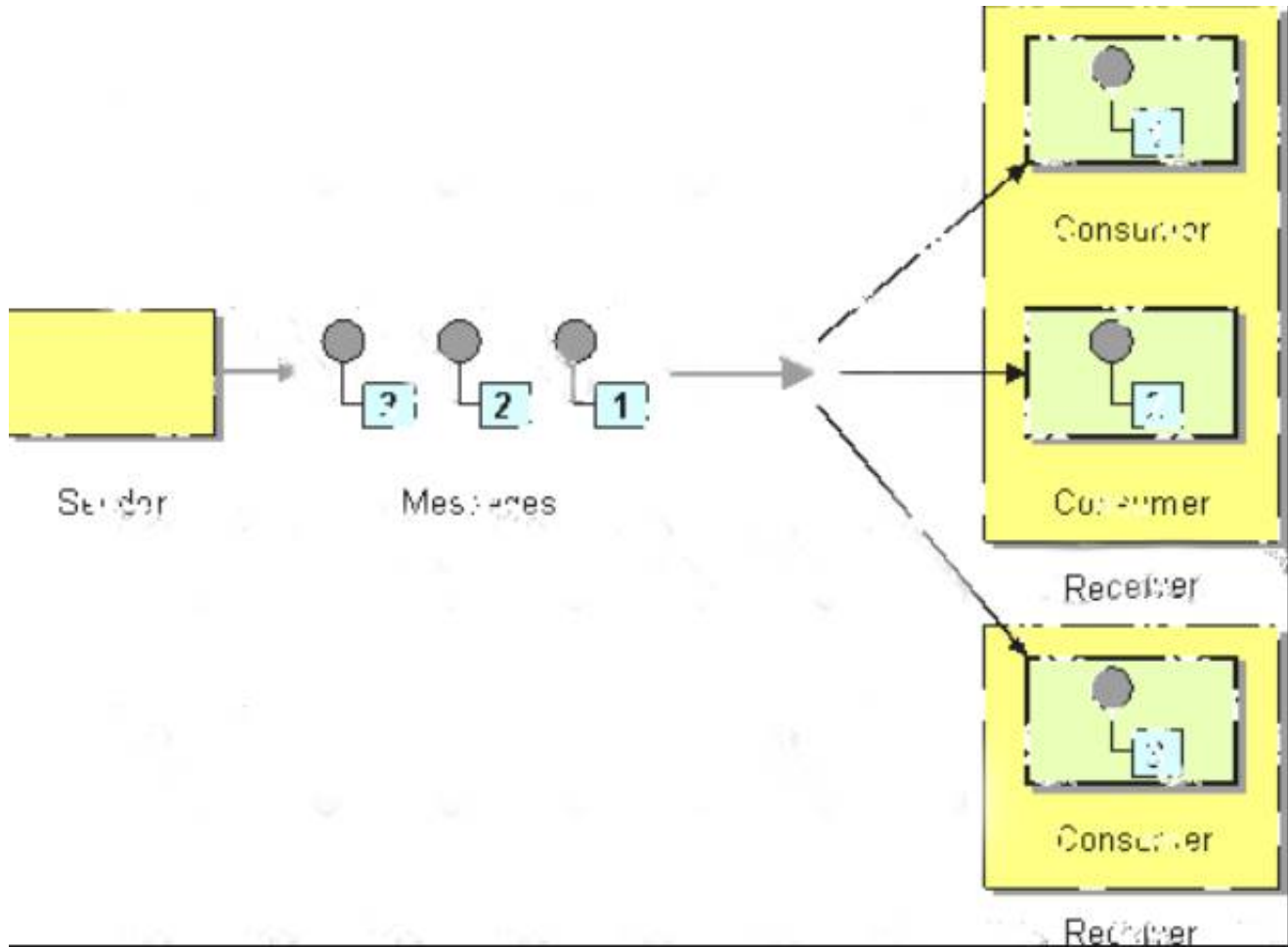
* Competing Consumers

are multiple consumers that are all created to receive messages from a single

Point-to-Point Channel. When the channel delivers a message, any of the consumers could potentially receive it. The messaging system's implementation

determines which consumer actually receives the message, but in effect the consumers compete with each other to be the receiver. Once a consumer receives a message, it can delegate to the rest of its application to help process the message.

Diagram Description automatically generated



* In case you are unaware about term idempotent re is more info:
Idempotent operations means their result will always same no matter how many times these operations are invoked.
Table Description automatically generated

IDEMPOTENCE		
WHEN PERFORMING AN OPERATION AGAIN GIVES THE SAME RESULT		
HTTP METHOD	IDEMPOTENCE	SAFETY
GET	YES	YES
HEAD	YES	YES
PUT	YES	NO
DELETE	YES	NO
POST	NO	NO
PATCH	NO	NO

Bottom of Form

NEW QUESTION 98

An organization has defined a common object model in Java to mediate the communication between different Mule applications in a consistent way. A Mule application is being built to use this common object model to process responses from a SOAP API and a REST API and then write the processed results to an order management system. The developers want Anypoint Studio to utilize these common objects to assist in creating mappings for various transformation steps in the Mule application. What is the most idiomatic (used for its intended purpose) and performant way to utilize these common objects to map between the inbound and outbound systems in the Mule application?

- A. Use JAXB (XML) and Jackson (JSON) data bindings
- B. Use the WSS module
- C. Use the Java module
- D. Use the Transform Message component

Answer: A

NEW QUESTION 102

What metrics about API invocations are available for visualization in custom charts using Anypoint Analytics?

- A. Request size, request HTTP verbs, response time

- B. Request size, number of requests, JDBC Select operation result set size
- C. Request size, number of requests, response size, response time
- D. Request size, number of requests, JDBC Select operation response time

Answer: C

Explanation:

Correct answer is Request size, number of requests, response size, response time Analytics API Analytics can provide insight into how your APIs are being used and how they are performing. From API Manager, you can access the Analytics dashboard, create a custom dashboard, create and manage charts, and create reports. From API Manager, you can get following types of analytics: - API viewing analytics - API events analytics - Charted metrics in API Manager

It can be accessed using: <http://anypoint.mulesoft.com/analytics>

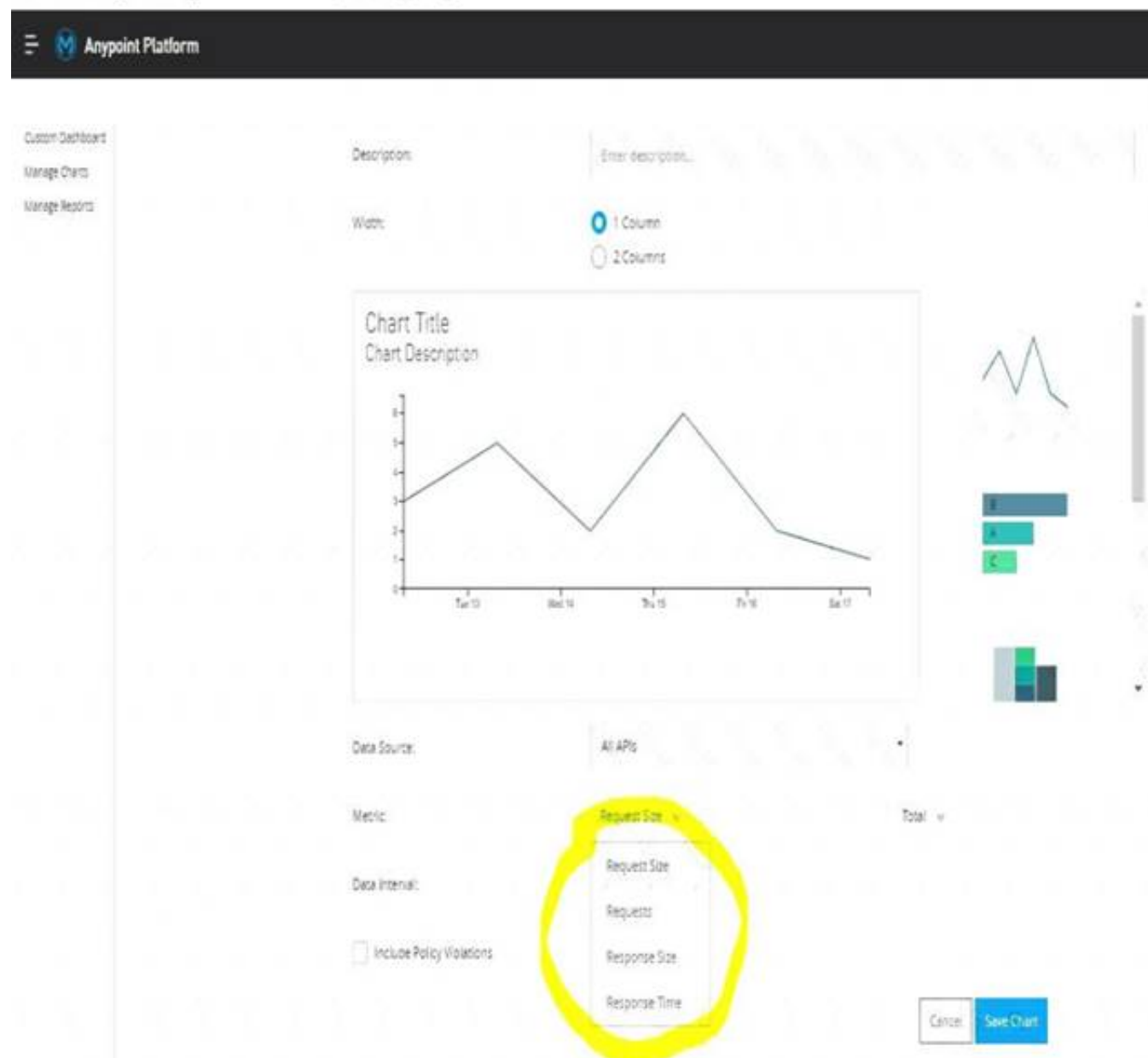
API Analytics provides a summary in chart form of requests, top apps, and latency for a particular duration. The custom dashboard in Anypoint Analytics contains a set of charts for a single API or for all APIs Each

chart displays various API characteristics

- Requests size: Line chart representing size of requests in KBs
- Requests : Line chart representing number of requests over a period
- Response size : Line chart representing size of response in KBs
- Response time :Line chart representing response time in ms

* To check this, You can go to API Manager > Analytics > Custom Dashboard > Edit Dashboard > Create Chart > Metric

Graphical user interface, chart Description automatically generated



NEW QUESTION 107

An Order microservice and a Fulfillment microservice are being designed to communicate with their clients through message-based integration (and NOT through API invocations).

The Order microservice publishes an Order message (a kind of command message) containing the details of an order to be fulfilled. The intention is that Order messages are only consumed by one Mule application, the Fulfillment microservice.

The Fulfillment microservice consumes Order messages, fulfills the order described therein, and then publishes an OrderFulfilled message (a kind of event message). Each OrderFulfilled message can be consumed by any interested Mule application, and the Order microservice is one such Mule application.

What is the most appropriate choice of message broker(s) and message destination(s) in this scenario?

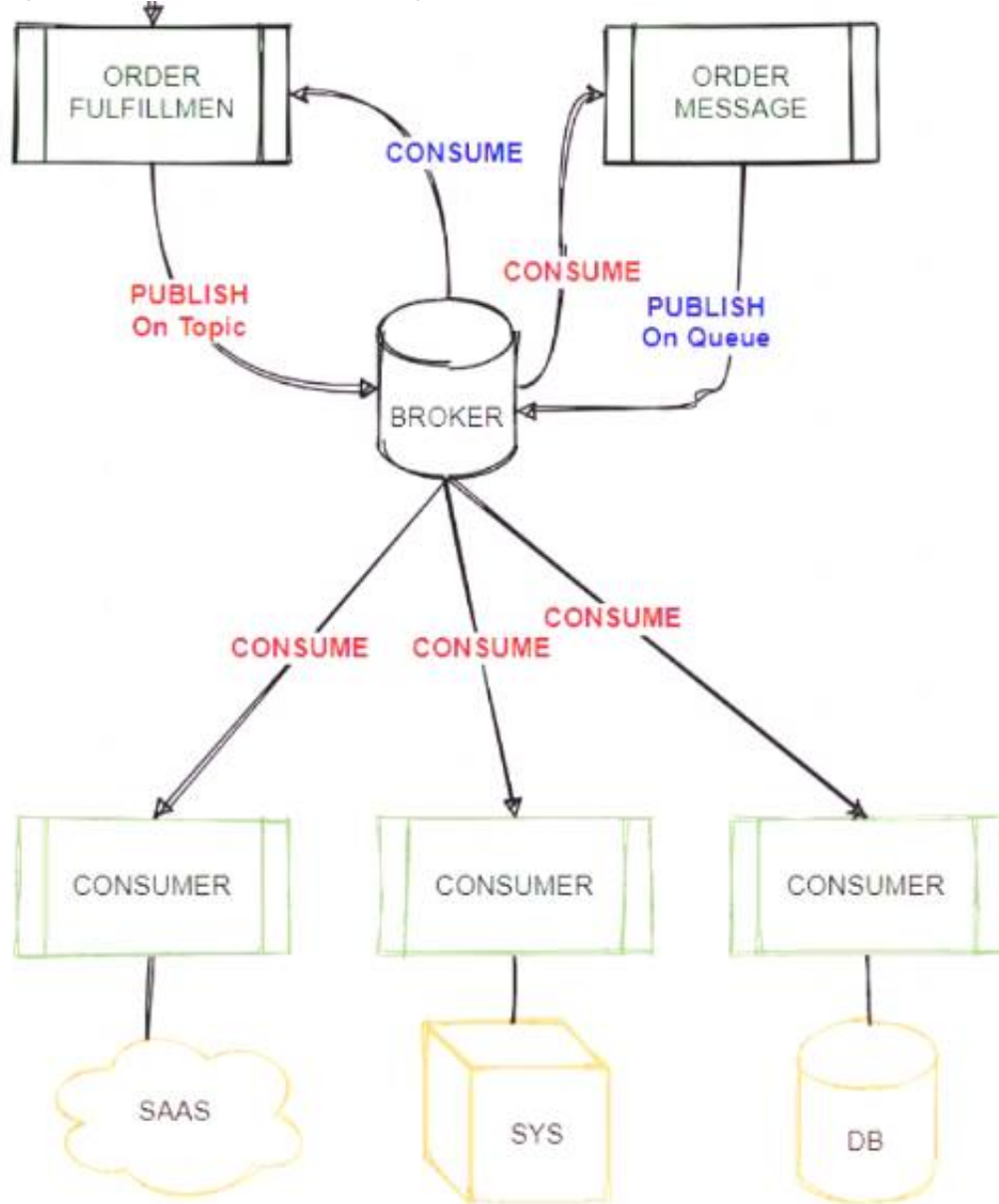
- A. Order messages are sent to an Anypoint MQ exchange OrderFulfilled messages are sent to an Anypoint MQ queue Both microservices interact with Anypoint MQ as the message broker, which must therefore scale to support the load of both microservices
- B. Order messages are sent to a JMS queue
- C. OrderFulfilled messages are sent to a JMS topic Both microservices interact with the same JMS provider (message broker) instance, which must therefore scale to support the load of both microservices
- D. Order messages are sent directly to the Fulfillment microservice
- E. OrderFulfilled messages are sent directly to the Order microservice The Order microservice interacts with one AMQP-compatible message broker and the Fulfillment microservice interacts with a different AMQP-compatible message broker, so that both message brokers can be chosen and scaled to best support the load of each microservice
- F. Order messages are sent to a JMS queue
- G. OrderFulfilled messages are sent to a JMS topic The Order microservice interacts with one JMS provider (message broker) and the Fulfillment microservice interacts with a different JMS provider, so that both message brokers can be chosen and scaled to best support the load of each microservice

Answer: B

Explanation:

* If you need to scale a JMS provider/ message broker, - add nodes to scale it horizontally or - add memory to scale it vertically * Cons of adding another JMS provider/ message broker: - adds cost. - adds complexity to use two JMS brokers - adds Operational overhead if we use two brokers, say, ActiveMQ and IBM MQ * So Two options that mention to use two brokers are not best choice. * It's mentioned that "The Fulfillment microservice consumes Order messages, fulfills the order described therein, and then publishes an OrderFulfilled message. Each OrderFulfilled message can be consumed by any interested Mule application." - When you publish a message on a topic, it goes to all the subscribers who are interested - so zero to many subscribers will receive a copy of the message. - When you send a message on a queue, it will be received by exactly one consumer. * As we need multiple consumers to consume the message below option is not valid choice: "Order messages are sent to an Anypoint MQ exchange. OrderFulfilled messages are sent to an Anypoint MQ queue. Both microservices interact with Anypoint MQ as the message broker, which must therefore scale to support the load of both microservices" * Order messages are only consumed by one Mule application, the Fulfillment microservice, so we will publish it on queue and OrderFulfilled message can be consumed by any interested Mule application so it need to be published on Topic using same broker. * Correct Answer Best choice in this scenario is: "Order messages are sent to a JMS queue. OrderFulfilled messages are sent to a JMS topic. Both microservices interact with the same JMS provider (message broker) instance, which must therefore scale to support the load of both microservices" Tried to depict scenario in diagram:

Diagram Description automatically generated



NEW QUESTION 108

An airline is architecting an API connectivity project to integrate its flight data into an online aggregation website. The interface must allow for secure communication high-performance and asynchronous message exchange.

What are suitable interface technologies for this integration assuming that Mulesoft fully supports these technologies and that Anypoint connectors exist for these interfaces?

- A. AsyncAPI over HTTPS AMQP with RabbitMQ JSON/REST over HTTPS
- B. XML over ActiveMQ XML over SFTP XML/REST over HTTPS
- C. CSV over FTP YAML over TLS JSON over HTTPS
- D. SOAP over HTTPS HOP over TLS gRPC over HTTPS

Answer: A

NEW QUESTION 111

Mule application A receives a request Anypoint MQ message REQU with a payload containing a variable-length list of request objects. Application A uses the For Each scope to split the list into individual objects and sends each object as a message to an Anypoint MQ queue.

Service S listens on that queue, processes each message independently of all other messages, and sends a response message to a response queue.

Application A listens on that response queue and must in turn create and publish a response Anypoint MQ message RESP with a payload containing the list of responses sent by service S in the same order as the request objects originally sent in REQU.

Assume successful response messages are returned by service S for all request messages.

What is required so that application A can ensure that the length and order of the list of objects in RESP and REQU match, while at the same time maximizing message throughput?

- A. Use a Scatter-Gather within the For Each scope to ensure response message order Configure the Scatter-Gather with a persistent object store
- B. Perform all communication involving service S synchronously from within the For Each scope, so objects in RESP are in the exact same order as request objects in REQU
- C. Use an Async scope within the For Each scope and collect response messages in a second For Each scope in the order In which they arrive, then send RESP using this list of responses
- D. Keep track of the list length and all object indices in REQU, both in the For Each scope and in all communication involving service Use persistent storage when

creating RESP

Answer: D

Explanation:

: Using Anypoint MQ, you can create two types of queues: Standard queue These queues don't guarantee a specific message order. Standard queues are the best fit for applications in which messages must be delivered quickly. FIFO (first in, first out) queue These queues ensure that your messages arrive in order. FIFO queues are the best fit for applications requiring strict message ordering and exactly-once delivery, but in which message delivery speed is of less importance Use of FIFO queue is no where in the option and also it decreased throughput. Similarly persistent object store is not the preferred solution approach when you maximizing message throughput. This rules out one of the options. Scatter Gather does not support ObjectStore. This rules out one of the options. Standard Anypoint MQ queues don't guarantee a specific message order hence using another for each block to collect response wont work as requirement here is to ensure the order. Hence considering all the above factors the feasible approach is Perform all communication involving service S synchronously from within the For Each scope, so objects in RESP are in the exact same order as request objects in REQU

NEW QUESTION 112

A Mule application is synchronizing customer data between two different database systems.

What is the main benefit of using eXtended Architecture (XA) transactions over local transactions to synchronize these two different database systems?

- A. An XA transaction synchronizes the database systems with the least amount of Mule configuration or coding
- B. An XA transaction handles the largest number of requests in the shortest time
- C. An XA transaction automatically rolls back operations against both database systems if any operation falls
- D. An XA transaction writes to both database systems as fast as possible

Answer: B

NEW QUESTION 114

An organization has decided on a cloud migration strategy to minimize the organization's own IT resources. Currently the organization has all of its new applications running on its own premises and uses an on-premises load balancer that exposes all APIs under the base URL (<https://api.rutujar.com>).

As part of migration strategy, the organization is planning to migrate all of its new applications and load balancer CloudHub.

What is the most straightforward and cost-effective approach to Mule application deployment and load balancing that preserves the public URL's?

- A. Deploy the Mule application to CloudhubCreate a CNAME record for base URL(<https://api.rutujar.com>) in the Cloudhub shared load balancer that points to the A record of the on-premises load balancerApply mapping rules in SLB to map URLto their corresponding Mule applications
- B. Deploy the Mule application to CloudhubUpdate a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the Cloudhub dedicated load balancerApply mapping rules in DLB to map URLto their corresponding Mule applications
- C. Deploy the Mule application to CloudhubUpdate a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the CloudHub shared load balancerApply mapping rules in SLB to map URLto their corresponding Mule applications
- D. For each migrated Mule application, deploy an API proxy application to Cloudhub with all traffic to themule applications routed through a Cloud Hub Dedicated load balancer (DLB)Update a CNAME record for base URL (<https://api.rutujar.com>) in the organization's DNS server to point to the A record of the CloudHub dedicated load balancerApply mapping rules in DLB to map each API proxy application who is responding new application

Answer: B

NEW QUESTION 119

An organization will deploy Mule applications to Cloudhub, Business requirements mandate that all application logs be stored ONLY in an external splunk consolidated logging service and NOT in Cloudhub.

In order to most easily store Mule application logs ONLY in Splunk, how must Mule application logging be configured in Runtime Manager, and where should the log4j2 splunk appender be defined?

- A. Keep the default logging configuration in RuntimeManagerDefine the splunk appender in ONE global log4j.xml file that is uploaded once to Runtime Manager to support at Mule application deployments.
- B. Disable Cloudhub logging in Runtime ManagerDefine the splunk appender in EACH Mule application's log4j2.xml file
- C. Disable Cloudhub logging in Runtime ManagerDefine the splunk appender in ONE global log4j.xml file that is uploaded once to Runtime Manger tosupport at Mule application deployments.
- D. Keep the default logging configuration in Runtime ManagerDefine the Splunk appender in EACH Mule application log4j2.xml file

Answer: B

Explanation:

By default, CloudHub replaces a Mule application's log4j2.xml file with a CloudHub log4j2.xml file. In CloudHub, you can disable the CloudHub provided Mule application log4j2 file. This allows integrating Mule application logs with custom or third-party log management systems

NEW QUESTION 121

A stock broking company makes use of CloudHub VPC to deploy Mule applications. Mule application needs to connect to a database application in the customers on-premises corporate data center and also to a Kafka cluster running in AWS VPC.

How is access enabled for the API to connect to the database application and Kafka cluster securely?

- A. Set up a transit gateway to the customers on-premises corporate datacenter to AWS VPC
- B. Setup AnyPoint VPN to the customer's on-premise corporate data center and VPC peering with AWS VPC
- C. Setup VPC peering with AWS VPC and the customers devices corporate data center
- D. Setup VPC peering with the customers onto my service corporate data center and Anypoint VPN to AWS VPC

Answer: B

NEW QUESTION 124

An organization's governance process requires project teams to get formal approval from all key stakeholders for all new Integration design specifications. An integration Mule application Is being designed that interacts with various backend systems. The Mule application will be created using Anypoint Design Center or Anypoint Studio and will then be deployed to a customer-hosted runtime.

What key elements should be included in the integration design specification when requesting approval for this Mule application?

- A. SLAs and non-functional requirements to access the backend systems
- B. Snapshots of the Mule application's flows, including their error handling
- C. A list of current and future consumers of the Mule application and their contact details
- D. The credentials to access the backend systems and contact details for the administrator of each system

Answer: A

Explanation:

SLAs and non-functional requirements to access the backend systems. Only this option actually speaks to design parameters and reqs. * Below two are technical implementations and not the part of design: - Snapsho of the Mule application's flows, including their error handling - The credentials to access the backend systems and contact details for the administrator of each system * List of consumers is not relevant to the design

NEW QUESTION 127

An organization uses Mule runtimes which are managed by Anypoint Platform - Private Cloud Edition. What MuleSoft component is responsible for feeding analytics data to non-MuleSoft analytics platforms?

- A. Anypoint Exchange
- B. The Mule runtimes
- C. Anypoint API Manager
- D. Anypoint Runtime Manager

Answer: D

Explanation:

Correct answer is Anypoint Runtime Manager

MuleSoft Anypoint Runtime Manager (ARM) provides connectivity to Mule Runtime engines deployed across your organization to provide centralized management, monitoring and analytics reporting. However, most enterprise customers find it necessary for these on-premises runtimes to integrate with their existing non MuleSoft analytics / monitoring systems such as Splunk and ELK to support a single pane of glass view across the infrastructure.

* You can configure the Runtime Manager agent to export data to external analytics tools.

Using either the Runtime Manager cloud console or Anypoint Platform Private Cloud Edition, you can:

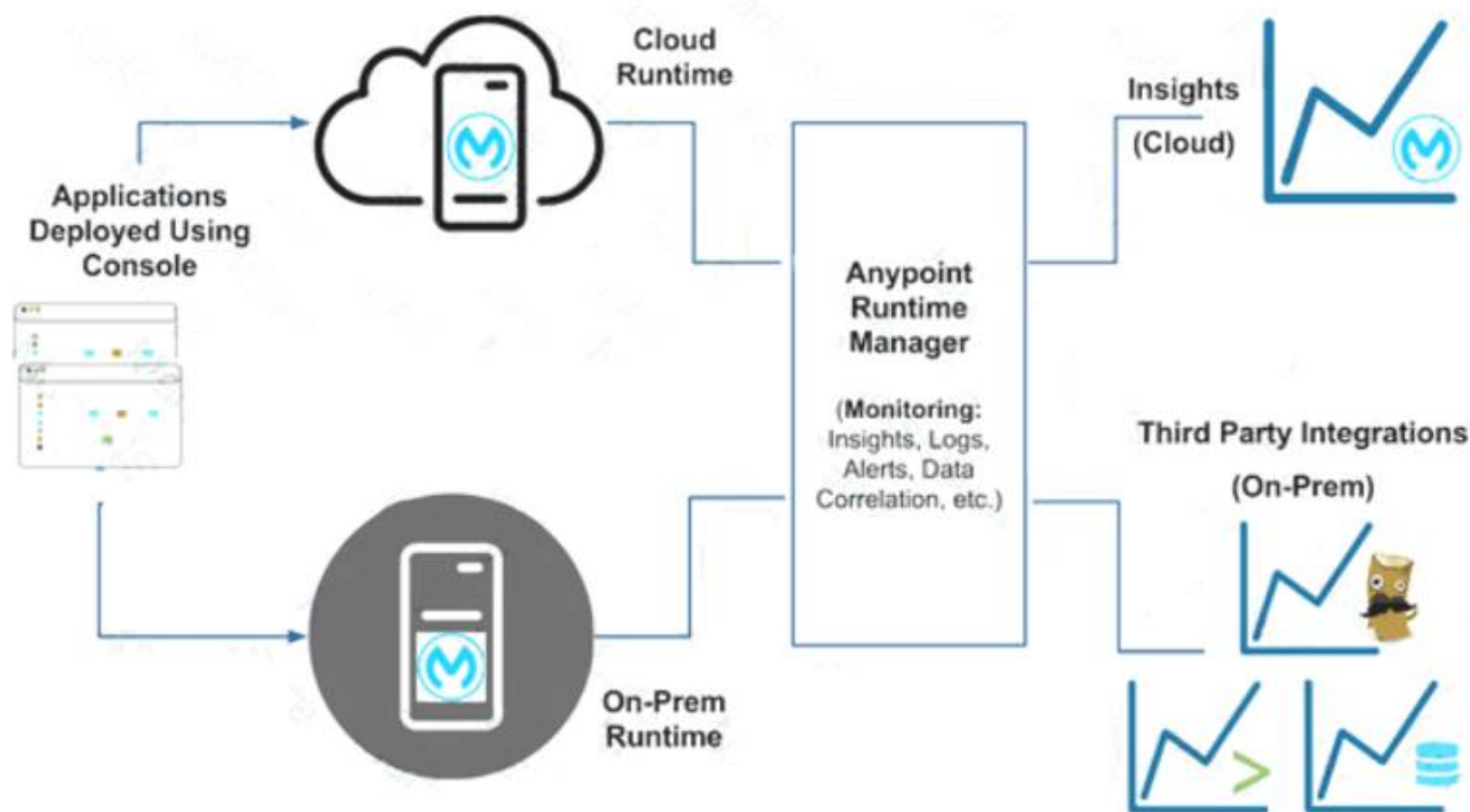
--> Send Mule event notifications, including flow executions and exceptions, to Splunk or ELK.

--> Send API Analytics to Splunk or ELK. Sending data to third-party tools is not supported for applications deployed on CloudHub.

You can use the CloudHub custom log appender to integrate with your logging system. Reference: <https://docs.mulesoft.com/runtime-manager/>

<https://docs.mulesoft.com/release-notes/runtime-manager-agent/runtime-manager-agent-release-notes>

Diagram Description automatically generated



Additional Info:

It can be achieved in 3 steps:

- 1) register an agent to a runtime manager,
- 2) configure a gateway to enable API analytics to be sent to non MuleSoft analytics platform (Splunk for ex.) – as highlighted in the following diagram and
- 3) setup dashboards.

Diagram Description automatically generated



NEW QUESTION 129

An organization is implementing a Quote of the Day API that caches today's quote. What scenario can use the CloudHub Object Store connector to persist the cache's state?

- A. When there is one deployment of the API implementation to CloudHub and another one to customer hosted mule runtime that must share the cache state.
- B. When there are two CloudHub deployments of the API implementation by two Anypoint Platform business groups to the same CloudHub region that must share the cache state.
- C. When there is one CloudHub deployment of the API implementation to three workers that must share the cache state.
- D. When there are three CloudHub deployments of the API implementation to three separate CloudHub regions that must share the cache state.

Answer: C

Explanation:

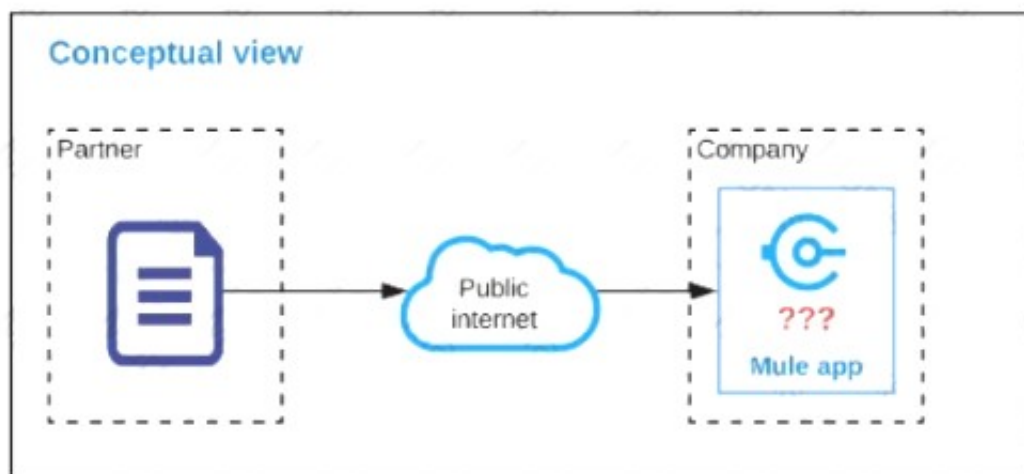
Object Store Connector is a Mule component that allows for simple key-value storage. Although it can serve a wide variety of use cases, it is mainly design for: - Storing synchronization information, such as watermarks. - Storing temporal information such as access tokens. - Storing user information. Additionally, Mule Runtime uses Object Stores to support some of its own components, for example: - The Cache module uses an Object Store to maintain all of the cached data. - The OAuth module (and every OAuth enabled connector) uses Object Stores to store the access and refresh tokens. Object Store data is in the same region as the worker where the app is initially deployed. For example, if you deploy to the Singapore region, the object store persists in the Singapore region. MuleSoft Reference : <https://docs.mulesoft.com/object-store-connector/1.1/> Data can be shared between different instances of the Mule application. This is not recommended for Inter Mule app communication. Coming to the question, object store cannot be used to share cached data if it is deployed as separate Mule applications or deployed under separate Business Groups. Hence correct answer is When there is one CloudHub deployment of the API implementation to three workers that must share the cache state.

NEW QUESTION 133

Refer to the exhibit.

An organization is designing a Mule application to receive data from one external business partner. The two companies currently have no shared IT infrastructure and do not want to establish one. Instead, all communication should be over the public internet (with no VPN).

What Anypoint Connector can be used in the organization's Mule application to securely receive data from this external business partner?



- A. File connector
- B. VM connector
- C. SFTP connector
- D. Object Store connector

Answer: C

Explanation:

- * Object Store and VM Store is used for sharing data inter or intra mule applications in same setup. Can't be used with external Business Partner
- * Also File connector will not be useful as the two companies currently have no shared IT infrastructure. It's specific for local use.
- * Correct answer is SFTP connector. The SFTP Connector implements a secure file transport channel so that your Mule application can exchange files with external resources. SFTP uses the SSH security protocol to transfer messages. You can implement the SFTP endpoint as an inbound endpoint with a one-way exchange pattern, or as an outbound endpoint configured for either a one-way or request-response exchange pattern.

NEW QUESTION 135

What condition requires using a CloudHub Dedicated Load Balancer?

- A. When cross-region load balancing is required between separate deployments of the same Mule application
- B. When custom DNS names are required for API implementations deployed to customer-hosted Mule runtimes
- C. When API invocations across multiple CloudHub workers must be load balanced

D. When server-side load-balanced TLS mutual authentication is required between API implementations and API clients

Answer: D

Explanation:

Correct answer is When server-side load-balanced TLS mutual authentication is required between API implementations and API clients CloudHub dedicated load balancers (DLBs) are an optional component of Anypoint Platform that enable you to route external HTTP and HTTPS traffic to multiple Mule applications deployed to CloudHub workers in a Virtual Private Cloud (VPC). Dedicated load balancers enable you to:

- * Handle load balancing among the different CloudHub workers that run your application.
- * Define SSL configurations to provide custom certificates and optionally enforce two-way SSL client authentication.
- * Configure proxy rules that map your applications to custom domains. This enables you to host your applications under a single domain

NEW QUESTION 136

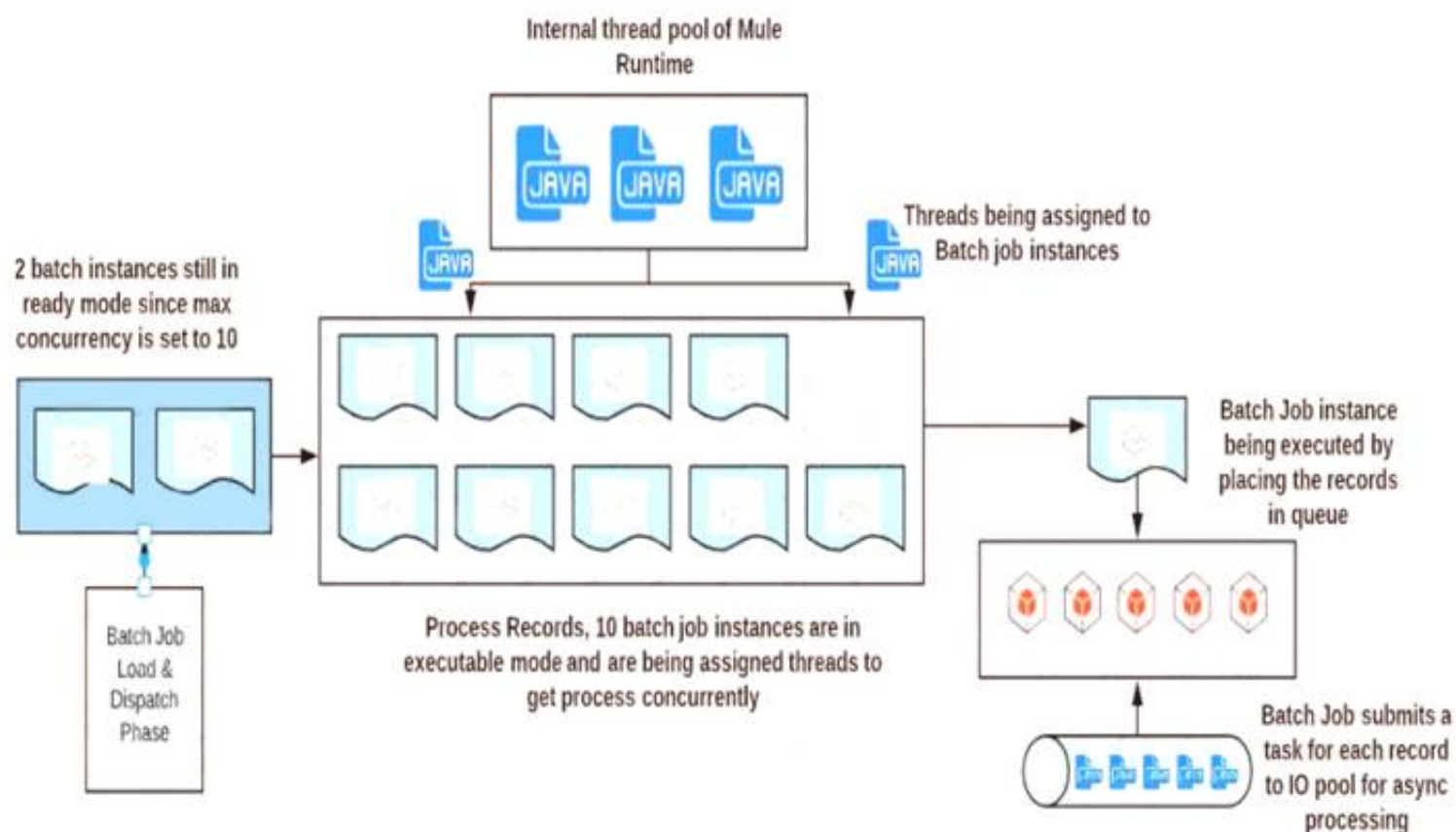
A Mule application contains a Batch Job with two Batch Steps (Batch_Step_1 and Batch_Step_2). A payload with 1000 records is received by the Batch Job. How many threads are used by the Batch Job to process records, and how does each Batch Step process records within the Batch Job?

- A. Each Batch Job uses SEVERAL THREADS for the Batch Steps Each Batch Step instance receives ONE record at a time as the payload, and RECORDS are processed IN PARALLEL within and between the two Batch Steps
- B. Each Batch Job uses a SINGLE THREAD for all Batch steps Each Batch step instance receives ONE record at a time as the payload, and RECORDS are processed IN ORDER, first through Batch_Step_1 and then through Batch_Step_2
- C. Each Batch Job uses a SINGLE THREAD to process a configured block size of record Each Batch Step instance receives A BLOCK OF records as the payload, and BLOCKS of records are processed IN ORDER
- D. Each Batch Job uses SEVERAL THREADS for the Batch Steps Each Batch Step instance receives ONE record at a time as the payload, and BATCH STEP INSTANCES execute IN PARALLEL to process records and Batch Steps in ANY order as fast as possible

Answer: A

Explanation:

- * Each Batch Job uses SEVERAL THREADS for the Batch Steps
- * Each Batch Step instance receives ONE record at a time as the payload. It's not received in a block, as it does not wait for multiple records to be completed before moving to next batch step. (So Option D is out of choice)
- * RECORDS are processed IN PARALLEL within and between the two Batch Steps.
- * RECORDS are not processed in order. Let's say if second record completes batch_step_1 before record 1, then it moves to batch_step_2 before record 1. (So option C and D are out of choice)
- * A batch job is the scope element in an application in which Mule processes a message payload as a batch of records. The term batch job is inclusive of all three phases of processing: Load and Dispatch, Process, and On Complete.
- * A batch job instance is an occurrence in a Mule application whenever a Mule flow executes a batch job. Mule creates the batch job instance in the Load and Dispatch phase. Every batch job instance is identified internally using a unique String known as batch job instance id.



NEW QUESTION 140

A project uses Jenkins to implement CI/CD process. It was observed that each Mule package contains some of the Jenkins files and folders for configurations of CI/CD jobs.

As these files and folders are not part of the actual package, expectation is that these should not be part of deployed archive. Which file can be used to exclude these files and folders from the deployed archive?

- A. muleignore
- B. _unTrackMule
- C. muleInclude
- D. _muleExclude

Answer: D

NEW QUESTION 145

An external REST client periodically sends an array of records in a single POST request to a Mule application API endpoint.

The Mule application must validate each record of the request against a JSON schema before sending it to a downstream system in the same order that it was received in the array
Record processing will take place inside a router or scope that calls a child flow. The child flow has its own error handling defined. Any validation or communication failures should not prevent further processing of the remaining records.
To best address these requirements what is the most idiomatic(used for it intended purpose) router or scope to used in the parent flow, and what type of error handler should be used in the child flow?

- A. First Successful router in the parent flow On Error Continue error handler in the child flow
- B. For Each scope in the parent flow On Error Continue error handler in the child flow
- C. Parallel For Each scope in the parent flow On Error Propagate error handler in the child flow
- D. Until Successful router in the parent flow On Error Propagate error handler in the child flow

Answer: B

Explanation:

Correct answer is For Each scope in the parent flow On Error Continue error handler in the child flow. You can extract below set of requirements from the question
a) Records should be sent to downstream system in the same order that it was received in the array
b) Any validation or communication failures should not prevent further processing of the remaining records
First requirement can be met using For Each scope in the parent flow and second requirement can be met using On Error Continue scope in child flow so that error will be suppressed.

NEW QUESTION 148

A mule application must periodically process a large dataset which varies from 6 GB to 8 GB from a back-end database and write transform data to an FTPS server using a properly configured batch job scope.

The performance requirements of an application are approved to run in the cloud hub 0.2 vCore with 8 GB storage capacity and currency requirements are met. How can the high rate of records be effectively managed in this application?

- A. Use streaming with a file storage repeatable strategy for reading records from the database and batch aggregator with streaming to write to FTPS
- B. Use streaming with an in-memory repeatable store strategy for reading records from the database and batch aggregator with streaming to write to FTPS
- C. Use streaming with a file store repeatable strategy for reading records from the database and batch aggregator with an optimal size
- D. Use streaming with a file store repeatable strategy reading records from the database and batch aggregator without any required configuration

Answer: A

NEW QUESTION 149

The AnyAirline organization's passenger reservations center is designing an integration solution that combines invocations of three different System APIs (bookFlight, bookHotel, and bookCar) in a business transaction. Each System API makes calls to a single database.

The entire business transaction must be rolled back when at least one of the APIs fails.

What is the most idiomatic (used for its intended purpose) way to integrate these APIs in near real-time that provides the best balance of consistency, performance, and reliability?

- A. Implement eXtended Architecture (XA) transactions between the API implementations Coordinate between the API implementations using a Saga pattern Implement caching in each API implementation to improve performance
- B. Implement local transactions within each API implementation Configure each API implementation to also participate in the same eXtended Architecture (XA) transaction Implement caching in each API implementation to improve performance
- C. Implement local transactions in each API implementation Coordinate between the API implementations using a Saga pattern Apply various compensating actions depending on where a failure occurs
- D. Implement an eXtended Architecture (XA) transaction manager in a Mule application using a Saga pattern Connect each API implementation with the Mule application using XA transactions Apply various compensating actions depending on where a failure occurs

Answer: C

NEW QUESTION 152

An organization is migrating all its Mule applications to Runtime Fabric (RTF). None of the Mule applications use Mule domain projects.

Currently, all the Mule applications have been manually deployed to a server group among several customer hosted Mule runtimes.

Port conflicts between these Mule application deployments are currently managed by the DevOps team who carefully manage Mule application properties files.

When the Mule applications are migrated from the current customer-hosted server group to Runtime Fabric (RTF), for the Mule applications need to be rewritten and what DevOps port configuration responsibilities change or stay the same?

- A. Yes, the Mule applications Must be rewritten DevOps No Longer needs to manage port conflicts between the Mule applications
- B. Yes, the Mule applications Must be rewritten DevOps Must Still Manage port conflicts.
- C. NO, The Mule applications do NOT need to be rewritten DevOps MUST STILL manage port conflicts
- D. NO, the Mule applications do NO need to be rewritten DevOps NO LONGER needs to manage port conflicts between the Mule applications.

Answer: C

Explanation:

* Anypoint Runtime Fabric is a container service that automates the deployment and orchestration of your Mule applications and gateways.

* Runtime Fabric runs on customer-managed infrastructure on AWS, Azure, virtual machines (VMs) or bare-metal servers.

* As none of the Mule applications use Mule domain projects. applications are not required to be rewritten. Also when applications are deployed on RTF, by default ingress is allowed only on 8081.

* Hence port conflicts are not required to be managed by DevOps team

NEW QUESTION 154

49 of A popular retailer is designing a public API for its numerous business partners. Each business partner will invoke the API at the URL 58.

<https://api.acme.com/partners/v1>. The API implementation is estimated to require deployment to 5 CloudHub workers.

The retailer has obtained a public X.509 certificate for the name api.acme.com, signed by a reputable CA, to be used as the server certificate.

Where and how should the X.509 certificate and Mule applications be used to configure load balancing among the 5 CloudHub workers, and what DNS entries should be configured in order for the retailer to support its numerous business partners?

- A. Add the X.509 certificate to the Mule application's deployable archive, then configure a CloudHub Dedicated Load Balancer (DLB) for each of the Mule application's CloudHub workers
Create a CNAME for api.acme.com pointing to the DLB's A record
- B. Add the X.509 certificate to the CloudHub Shared Load Balancer (SLB), not to the Mule application
Create a CNAME for api.acme.com pointing to the SLB's A record
- C. Add the X.509 certificate to a CloudHub Dedicated Load Balancer (DLB), not to the Mule application
Create a CNAME for api.acme.com pointing to the DLB's A record
- D. Add the x.509 certificate to the Mule application's deployable archive, then configure the CloudHub Shared Load Balancer (SLB) for each of the Mule application's CloudHub workers
Create a CNAME for api.acme.com pointing to the SLB's A record

Answer: C

Explanation:

- * An X.509 certificate is a vital safeguard against malicious network impersonators. Without x.509 server authentication, man-in-the-middle attacks can be initiated by malicious access points, compromised routers, etc.
- * X.509 is most used for SSL/TLS connections to ensure that the client (e.g., a web browser) is not fooled by a malicious impersonator pretending to be a known, trustworthy website.
- * Coming to the question, we can not use SLB here as SLB does not allow to define vanity domain names. * Hence we need to use DLB and add certificate in there

Hence correct answer is Add the X 509 certificate to the cloudhub Dedicated Load Balancer (DLB), not the Mule application. Create the CNAME for api.acme.com pointing to the DLB's record

NEW QUESTION 156

An organization is building a test suite for their applications using m-unit. The integration architect has recommended using test recorder in studio to record the processing flows and then configure unit tests based on the capture events

What are the two considerations that must be kept in mind while using test recorder (Choose two answers)

- A. Tests for flows cannot be created with Mule errors raised inside the flow or already existing in the incoming event
- B. Recorder supports smoking a message before or inside a ForEach processor
- C. The recorder support loops where the structure of the data been tested changes inside the iteration
- D. A recorded flow execution ends successfully but the result does not reach its destination because the application is killed
- E. Mocking values resulting from parallel processes are possible and will not affect the execution of the processes that follow in the test

Answer: AD

NEW QUESTION 157

The implementation of a Process API must change. What is a valid approach that minimizes the impact of this change on API clients?

- A. Implement required changes to the Process API implementation so that whenever possible, the Process API's RAML definition remains unchanged
- B. Update the RAML definition of the current Process API and notify API client developers by sending them links to the updated RAML definition
- C. Postpone changes until API consumers acknowledge they are ready to migrate to a new Process API or API version
- D. Implement the Process API changes in a new API implementation, and have the old API implementation return an HTTP status code 301 - Moved Permanently to inform API clients they should be calling the new API implementation

Answer: A

Explanation:

- * Option B shouldn't be used unless extremely needed, if RAML is changed, client needs to accommodate changes. Question is about minimizing impact on Client. So this is not a valid choice.
- * Option C isn't valid as Business can't stop for consumers acknowledgment.
- * Option D again needs Client to accommodate changes and isn't viable option.
- * Best choice is A where RAML definition isn't changed and underlined functionality is changed without any dependency on client and without impacting client.

NEW QUESTION 162

A travel company wants to publish a well-defined booking service API to be shared with its business partners. These business partners have agreed to ONLY consume SOAP services and they want to get the service contracts in an easily consumable way before they start any development. The travel company will publish the initial design documents to Anypoint Exchange, then share those documents with the business partners. When using an API-led approach, what is the first design document the travel company should deliver to its business partners?

- A. Create a WSDL specification using any XML editor
- B. Create a RAML API specification using any text editor
- C. Create an OAS API specification in Design Center
- D. Create a SOAP API specification in Design Center

Answer: A

Explanation:

SOAP API specifications are provided as WSDL. Design center doesn't provide the functionality to create WSDL file. Hence WSDL needs to be created using XML editor

NEW QUESTION 167

An insurance company has an existing API which is currently used by customers. API is deployed to customer hosted Mule runtime cluster. The load balancer that is used to access any APIs on the mule cluster is only configured to point to applications hosted on the server at port 443.

Mule application team of a company attempted to deploy a second API using port 443 but the application will not start and checking logs shows an error indicating the address is already in use.

Which steps must the organization take to resolve this error and allow customers to access both the API's?

- A. Change the base path of the HTTP listener configuration in the second API to a different one from the first API
- B. Set HTTP listener configuration in both API's to allow for connections from multiple ports

- C. Move the HTTP listener configurations from the API's and package them in a mule domain project using port 443
- D. Set the HTTP listener of the second API to use different port than the one used in the first API

Answer: C

NEW QUESTION 169

An organization has decided on a cloudhub migration strategy that aims to minimize the organizations own IT resources. Currently, the organizational has all of its Mule applications running on its own premises and uses an premises load balancer that exposes all APIs under the base URL <https://api.acme.com>
As part of the migration strategy, the organization plans to migrate all of its Mule applications and load balancer to cloudhub
What is the most straight-forward and cost effective approach to the Mule applications deployment and load balancing that preserves the public URLs?

- A. Deploy the Mule applications to CloudhubUpdate the CNAME record for an api.acme.com in the organizations DNS server pointing to the A record of a cloudhub dedicated load balancer(DLB)Apply mapping rules in the DLB to map URLs to their corresponding Mule applications
- B. For each migrated Mule application, deploy an API proxy Mule application to Cloudhub with all applications under the control of a dedicated load balancer(CLB)Update the CNAME record for api.acme.com in the organization DNS server pointing to the A record of a cloudhub dedicated load balancer(DLB)Apply mapping rules in the DLB to map each API proxy application to its corresponding Mule applications
- C. Deploy the Mule applications to CloudhubCreate CNAME record for api.acme.com in the Cloudhub Shared load balancer (SLB) pointing to the A record of the on-premise load balancerApply mapping rules in the SLB to map URLs to their corresponding Mule applications
- D. Deploy the Mule applications to CloudhubUpdate the CNAME record for api.acme.com in the organization DNS server pointing to the A record of the cloudhub shared load balancer(SLB)Apply mapping rules in the SLB to map URLs to their corresponding Mule applications.

Answer: A

Explanation:

<https://help.mulesoft.com/s/feed/0D52T000055pzgsSAA>.

NEW QUESTION 172

Mule application muleA deployed in cloudhub uses Object Store v2 to share data across instances. As a part of new requirement , application muleB which is deployed in same region wants to access this Object Store.
Which of the following option you would suggest which will have minimum latency in this scenario?

- A. Object Store REST API
- B. Object Store connector
- C. Both of the above option will have same latency
- D. Object Store of one mule application cannot be accessed by other mule application.

Answer: A

Explanation:

V2 Rest API is recommended for on premise applications to access Object Store. It also comes with overhead of encryption and security of using rest api. With Object Store v2, the API call is localized to the same data center as the Runtime Manager app.
But in this case requirement is to access the OS of other mule application and not the same mule application. You can configure a Mule app to use the Object Store REST API to store and retrieve values from an object store in another Mule app.
However, Object Store v2 is not designed for app-to-app communication.

NEW QUESTION 173

A company is building an application network and has deployed four Mule APIs: one experience API, one process API, and two system APIs. The logs from all the APIs are aggregated in an external log aggregation tool. The company wants to trace messages that are exchanged between multiple API implementations. What is the most idiomatic (based on its intended use) identifier that should be used to implement Mule event tracing across the multiple API implementations?

- A. Mule event ID
- B. Mule correlation ID
- C. Client's IP address
- D. DataWeave UUID

Answer: B

Explanation:

Correct answer is Mule correlation ID By design, Correlation Ids cannot be changed within a flow in Mule 4 applications and can be set only at source. This ID is part of the Event Context and is generated as soon as the message is received by the application. When a HTTP Request is received, the request is inspected for "X-Correlation-Id" header. If "X-Correlation-Id" header is present, HTTP connector uses this as the Correlation Id. If "X-Correlation-Id" header is NOT present, a Correlation Id is randomly generated. For Incoming HTTP Requests: In order to set a custom Correlation Id, the client invoking the HTTP request must set "X-Correlation-Id" header. This will ensure that the Mule Flow uses this Correlation Id. For Outgoing HTTP Requests: You can also propagate the existing Correlation Id to downstream APIs. By default, all outgoing HTTP Requests send "X-Correlation-Id" header. However, you can choose to set a different value to "X-Correlation-Id" header or set "Send Correlation Id" to NEVER.

NEW QUESTION 176

A customer wants to use the mapped diagnostic context (MDC) and logging variables to enrich its logging and improve tracking by providing more context in the logs.

The customer also wants to improve the throughput and lower the latency of message processing.

As an Mulesoft integration architect can you advise, what should the customer implement to meet these requirements?

- A. Use synchronous logging and use pattern layout with [%MDC] in the log4j2.xml configuration file and then configure the logging variables
- B. Use async logger at the level greater than INFO and use pattern layout with [%MDC] in the log4j2.xml configuration file and then configure the logging variables
- C. Use async logger at the level equal to DEBUG or TRACE and use pattern layout with [%MDC] in the log4j2.xml configuration file and then configure the logging variables
- D. Use synchronous logging at the INFO DEBUG or Trace level and use pattern layout with [%MDC] in the log4j2.xml configuration file and then configure the logging variables

Answer: B

NEW QUESTION 179

When the mule application using VM is deployed to a customer-hosted cluster or multiple cloudhub workers, how are messages consumed by the Mule engine?

- A. in non-deterministic way
- B. by starting an XA transaction for each new message
- C. in a deterministic way
- D. the primary only in order to avoid duplicate processing

Answer: C

NEW QUESTION 184

An organization is designing a mule application to support an all or nothing transaction between serval database operations and some other connectors so that they all roll back if there is a problem with any of the connectors

Besides the database connector , what other connector can be used in the transaction.

- A. VM
- B. Anypoint MQ
- C. SFTP
- D. ObjectStore

Answer: A

Explanation:

Correct answer is VM VM support Transactional Type. When an exception occur, The transaction rolls back to its original state for reprocessing. This feature is not supported by other connectors.

Here is additional information about Transaction management: Table Description automatically generated

	Shared Load Balancer	Dedicated Load Balancer
VPC	Shared VPC (Mulesoft)	VPC (Customer)
Default Load Balancer	Cloudhub provides Default Shared Load Balancer available in All Environment	Need to Purchase
Organization Use	Multiple Organization	Specific to Organization
Certificate	Mulesoft Certificate	Organization Certificate
TLS Support	Yes	Yes,
URL Mapping	Fixed URL Mapping	Customer URL Mapping
Timeout	30 Sec Session Timeout	Custom Timeout
Ports	Public Port (80 : 8081, 443 : 8082)	Private Port (80 : 8091, 443 : 8092)
Fashion	Round Robin	Round Robin
Supports HTTPS Protocol	Yes	Yes.
Worker Assignment	No	Yes
IP Blacklisting/ Whitelisting	No	Yes
	https://docs.mulesoft.com/runtime-manager/ib-whitelists	
Configure Custom Domain	No	Yes
Custom Certificate	No	Yes
Rate Limit	Lower Rate Limit and applied According to Region	Higher Rate Limit Thresboid
VPC	Anypoint VPC optional	Can't Use DLB without Anypoint VPC

NEW QUESTION 188

What is required before an API implemented using the components of Anypoint Platform can be managed and governed (by applying API policies) on Anypoint Platform?

- A. The API must be published to Anypoint Exchange and a corresponding API instance ID must be obtained from API Manager to be used in the API implementation

- B. The API implementation source code must be committed to a source control management system (such as GitHub)
- C. A RAML definition of the API must be created in API designer so it can then be published to Anypoint Exchange
- D. The API must be shared with the potential developers through an API portal so API consumers can interact with the API

Answer: A

Explanation:

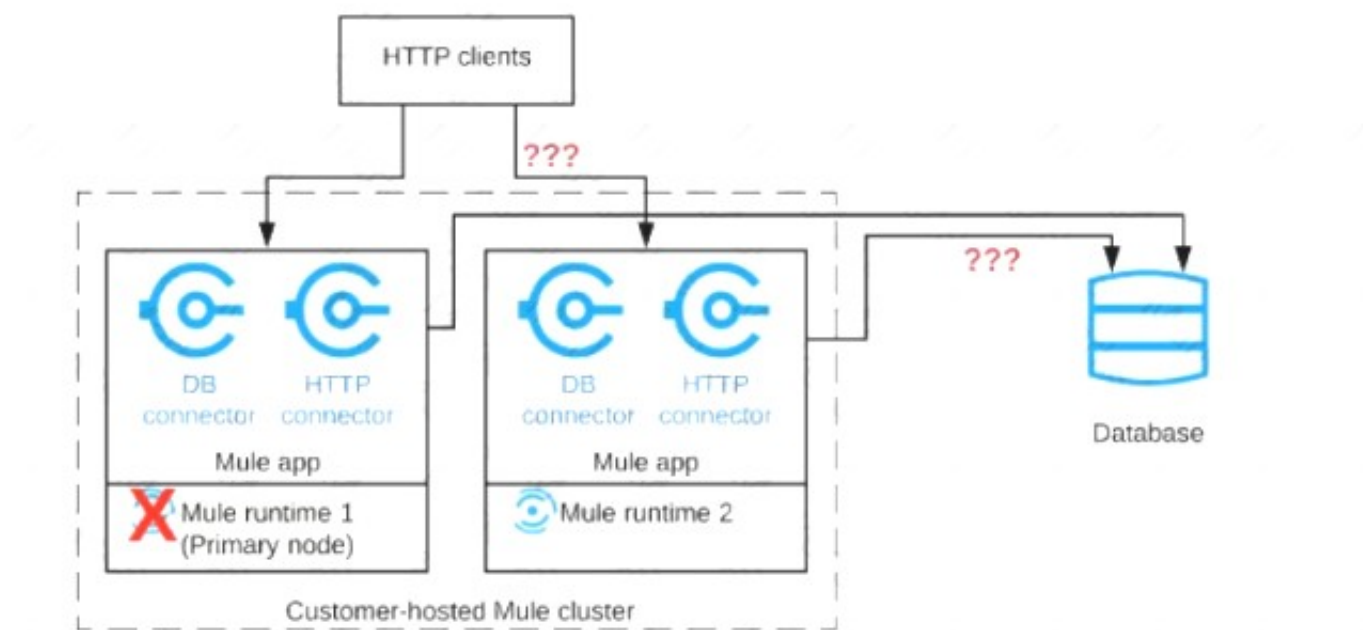
Context of the question is about managing and governing mule applications deployed on Anypoint platform.

Anypoint API Manager (API Manager) is a component of Anypoint Platform that enables you to manage, govern, and secure APIs. It leverages the runtime capabilities of API Gateway and Anypoint Service Mesh, both of which enforce policies, collect and track analytics data, manage proxies, provide encryption and authentication, and manage applications.

Mule Ref Doc : <https://docs.mulesoft.com/api-manager/2.x/getting-started-proxy>

NEW QUESTION 193

Refer to the exhibit.



A Mule application is deployed to a cluster of two customer-hosted Mule runtimes. The Mule application has a flow that polls a database and another flow with an HTTP Listener. HTTP clients send HTTP requests directly to individual cluster nodes.

What happens to database polling and HTTP request handling in the time after the primary (master) node of the cluster has failed, but before that node is restarted?

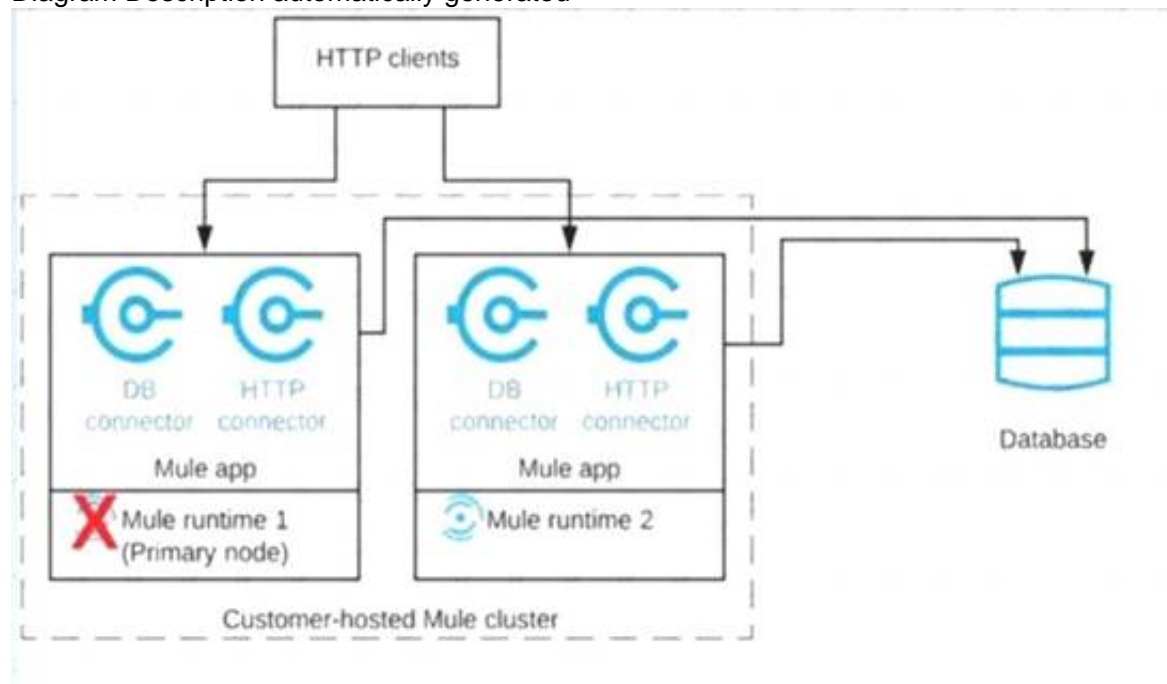
- A. Database polling continues Only HTTP requests sent to the remaining node continue to be accepted
- B. Database polling stops All HTTP requests continue to be accepted
- C. Database polling continues All HTTP requests continue to be accepted, but requests to the failed node Incur increased latency
- D. Database polling stops All HTTP requests are rejected

Answer: A

Explanation:

: Architecture described in the question could be described as follows. When node 1 is down, DB polling will still continue via node 2. Also requests which are coming directly to node 2 will also be accepted and processed in BAU fashion. Only thing that wont work is when requests are sent to Node 1 HTTP connector. The flaw with this architecture is HTTP clients are sending HTTP requests directly to individual cluster nodes. By default, clustering Mule runtime engines ensures high system availability. If a Mule runtime engine node becomes unavailable due to failure or planned downtime, another node in the cluster can assume the workload and continue to process existing events and messages

Diagram Description automatically generated



NEW QUESTION 198

What limits if a particular Anypoint Platform user can discover an asset in Anypoint Exchange?

- A. Design Center and RAML were both used to create the asset
- B. The existence of a public Anypoint Exchange portal to which the asset has been published

- C. The type of the asset in Anypoint Exchange
- D. The business groups to which the user belongs

Answer: D

Explanation:

* "The existence of a public Anypoint Exchange portal to which the asset has been published" - question does not mention anything about the public portal. Beside the public portal is open to the internet, to anyone. * If you cannot find an asset in the current business group scopes, search in other scopes. In the left navigation bar click All assets (assets provided by MuleSoft and your own master organization), Provided by MuleSoft, or a business group scope. User belonging to one Business Group can see assets related to his group only Reference: <https://docs.mulesoft.com/exchange/to-find-info> <https://docs.mulesoft.com/exchange/asset-details> Correct answer is The business groups to which the user belongs

NEW QUESTION 202

How are the API implementation , API client, and API consumer combined to invoke and process an API ?

- A. The API consumer creates an API implementation , which receives API invocations from an API such that they are processed for an API client
- B. The API consumer creates an API client which sends API invocations to an API such that they are processed by an API implementation
- C. An API client creates an API consumer, which receives API invocation from an API such that they are processed for an API implementation
- D. The API client creates an API consumer which sends API invocations to an API such that they are processed by API implementation

Answer: C

Explanation:

The API consumer creates an API client which sends API invocations to an API such that they are processed by an API implementation
This is based on below definitions API client • An application component • that accesses a service • by invoking an API of that service - by definition of the term API over HTTP API consumer • A business role, which is often assigned to an individual • that develops API clients, i.e., performs the activities necessary for enabling an API client to invoke APIs API implementation • An application component • that implements th functionality

NEW QUESTION 205

A new upstream API Is being designed to offer an SLA of 500 ms median and 800 ms maximum (99th percentile) response time. The corresponding API implementation needs to sequentially invoke 3 downstream APIs of very similar complexity. The first of these downstream APIs offers the following SLA for its response time: median: 100 ms, 80th percentile: 500 ms, 95th percentile: 1000 ms. If possible, how can a timeout be set in the upstream API for the invocation of the first downstream API to meet the new upstream API's desired SLA?

- A. Set a timeout of 100 ms; that leaves 400 ms for the other two downstream APIs to complete
- B. Do not set a timeout; the Invocation of this API Is mandatory and so we must wait until it responds
- C. Set a timeout of 50 ms; this times out more invocations of that API but gives additional room for retries
- D. No timeout is possible to meet the upstream API's desired SLA; a different SLA must be negotiated with the first downstream API or invoke an alternative API

Answer: D

Explanation:

Before we answer this question , we need to understand what median (50th percentile) and 80th percentile means. If the 50th percentile (median) of a response time is 500ms that means that 50% of my transactions are either as fast or faster than 500ms.

If the 90th percentile of the same transaction is at 1000ms it means that 90% are as fast or faster and only 10% are slower. Now as per upstream SLA , 99th percentile is 800 ms which means 99% of the incoming requests should have response time less than or equal to 800 ms. But as per one of the backend API , their 95th percentile is 1000 ms which means that backend API will take 1000 ms or less than that for 95% of. requests. As there are three API invocation from upstream API , we can not conclude a timeout that can be set to meet the desired SLA as backend SLA's do not support it.

Let see why other answers are not correct.

1) Do not set a timeout --> This can potentially violate SLA's of upstream API

2) Set a timeout of 100 ms; ---> This will not work as backend API has 100 ms as median meaning only 50% requests will be answered in this time and we will get timeout for 50% of the requests. Important thing to note here is, All APIs need to be executed sequentially, so if you get timeout in first API, there is no use of going to second and third API. As a service provider you wouldn't want to keep 50% of your consumers dissatisfied. So not the best option to go with.

*To quote an example: Let's assume you have built an API to update customer contact details.

- First API is fetching customer number based on login credentials

- Second API is fetching Info in 1 table and returning unique key

- Third API, using unique key provided in second API as primary key, updating remaining details

* Now consider, if API times out in first API and can't fetch customer number, in this case, it's useless to call API 2 and 3 and that is why question mentions specifically that all APIs need to be executed sequentially.

3) Set a timeout of 50 ms --> Again not possible due to the same reason as above Hence correct answer is No timeout is possible to meet the upstream API's desired SLA; a different SLA must be negotiated with the first downstream API or invoke an alternative API

NEW QUESTION 206

A leading e-commerce giant will use Mulesoft API's on runtime fabric (RTF) to process customer orders. Some customer's sensitive information such as credit card information is also there as a part of a API payload.

What approach minimizes the risk of matching sensitive data to the original and can convert back to the original value whenever and wherever required?

- A. Apply masking to hide the sensitive information and then use API
- B. manager to detokenize the masking format to return the original value
- C. create a tokenization format and apply a tokenization policy to the API Gateway
- D. Used both masking and tokenization
- E. Apply a field level encryption policy in the API Gateway

Answer: A

NEW QUESTION 211

.....

Relate Links

100% Pass Your MCIA-Level-1 Exam with Exam Bible Prep Materials

<https://www.exambible.com/MCIA-Level-1-exam/>

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - <https://www.exambible.com/>