

MuleSoft

Exam Questions MCIA-Level-1

MuleSoft Certified Integration Architect - Level 1



NEW QUESTION 1

A company is modernizing its legal systems to accelerate access to applications and data while supporting the adoption of new technologies. The key to achieving this business goal is unlocking the companies' key systems and data including microservices running under Docker and Kubernetes containers using APIs. Considering the current aggressive backlog and project delivery requirements the company wants to take a strategic approach in the first phase of its transformation projects by quickly deploying APIs in mule runtime that are able to scale, connect to on-premises systems and migrate as needed. Which runtime deployment option supports company's goals?

- A. Customer hosted self provisioned runtimes
- B. Cloudhub runtimes
- C. Runtime fabric on self managed Kubernetes
- D. Runtime fabric on VMware metal

Answer: C

NEW QUESTION 2

Additional nodes are being added to an existing customer-hosted Mule runtime cluster to improve performance. Mule applications deployed to this cluster are invoked by API clients through a load balancer. What is also required to carry out this change?

- A. A new load balancer must be provisioned to allow traffic to the new nodes in a round-robin fashion
- B. External monitoring tools or log aggregators must be configured to recognize the new nodes
- C. API implementations using an object store must be adjusted to recognize the new nodes and persist to them
- D. New firewall rules must be configured to accommodate communication between API clients and the new nodes

Answer: B

Explanation:

- * Clustering is a group of servers or mule runtime which acts as a single unit.
 - * Mulesoft Enterprise Edition supports scalable clustering to provide high availability for the Mulesoft application.
 - * In simple terms, virtual servers composed of multiple nodes and they communicate and share information through a distributed shared memory grid.
 - * By default, Mulesoft ensures the High availability of applications if clustering implemented.
 - * Let's consider the scenario one of the nodes in cluster crashed or goes down and under maintenance. In such cases, Mulesoft will ensure that requests are processed by other nodes in the cluster. Mulesoft clustering also ensures that the request is load balanced between all the nodes in a cluster.
 - * Clustering is only supported by on-premise Mule runtime and it is not supported in Cloudhub.
- Correct answer is External monitoring tools or log aggregators must be configured to recognize the new nodes
- * Rest of the options are automatically taken care of when a new node is added in cluster.

NEW QUESTION 3

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 it 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 4

A Mule application is being designed for deployment to a single CloudHub worker. The Mule application will have a flow that connects to a SaaS system to perform some operations each time the flow is invoked.

The SaaS system connector has operations that can be configured to request a short-lived token (fifteen minutes) that can be reused for subsequent connections within the fifteen minute time window. After the token expires, a new token must be requested and stored.

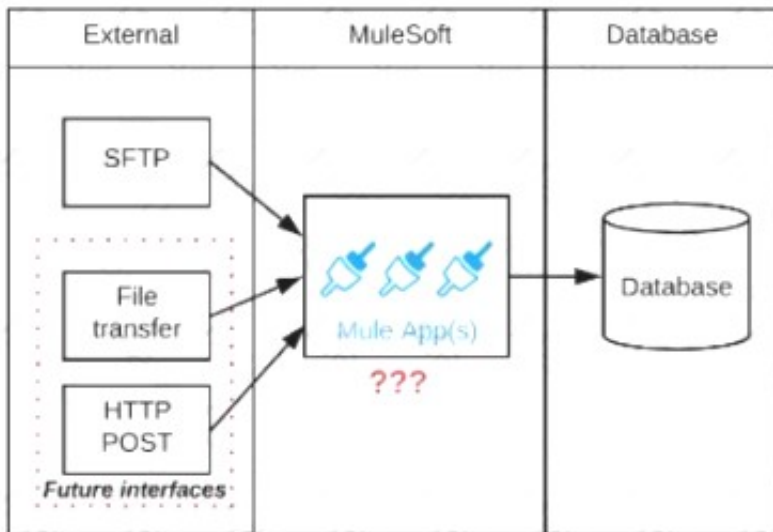
What is the most performant and idiomatic (used for its intended purpose) Anypoint Platform component or service to use to support persisting and reusing tokens in the Mule application to help speed up reconnecting the Mule application to the SaaS application?

- A. Nonpersistent object store
- B. Persistent object store
- C. Variable
- D. Database

Answer: D

NEW QUESTION 5

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.

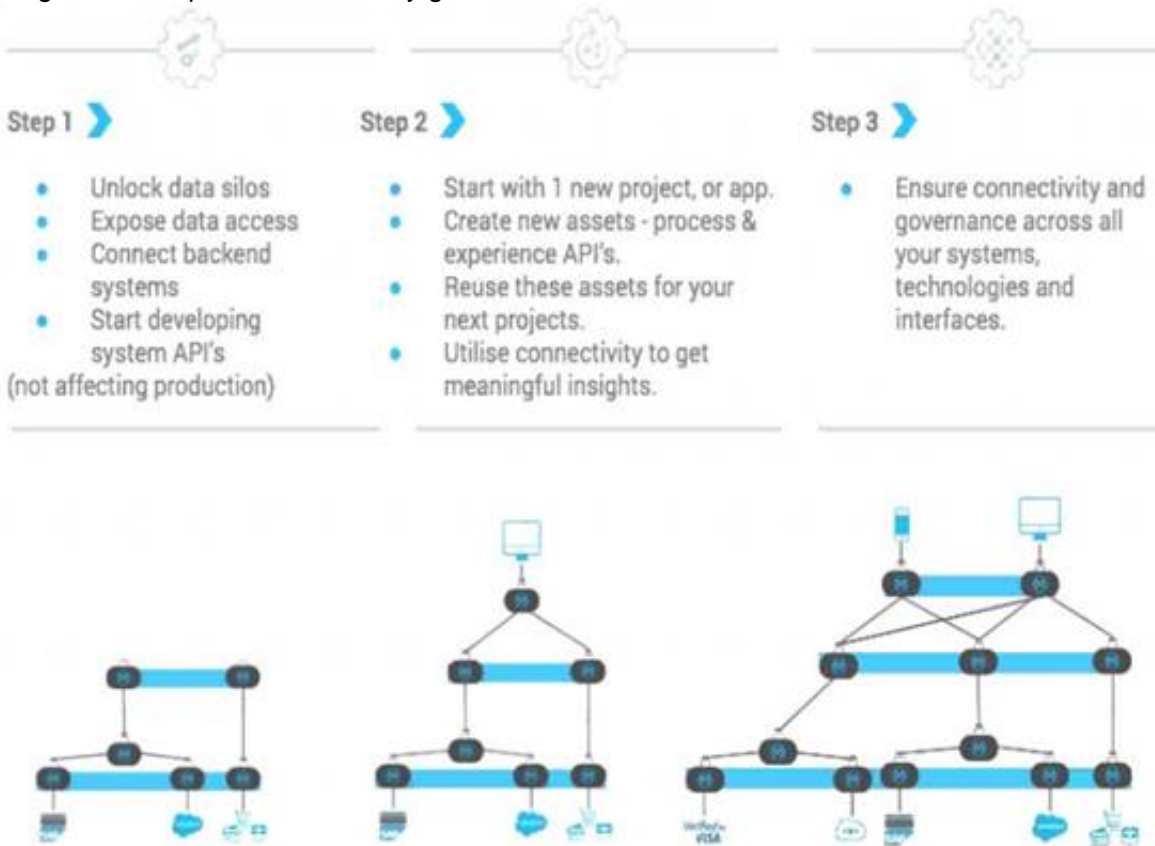
* If we use 1 API to receive all files from different Vendors, any new vendor addition will need changes to that 1 API to accommodate new requirements. So Option A and C are also ruled out.

* 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 6

A mule application is being designed to perform product orchestration. The Mule application needs to join together the responses from an inventory API and a Product Sales History API with the least latency.

To minimize the overall latency. What is the most idiomatic (used for its intended purpose) design to call each API request in the Mule application?

- A. Call each API request in a separate lookup call from Dataweave reduce operator
- B. Call each API request in a separate route of a Scatter-Gather

- C. Call each API request in a separate route of a Parallel For Each scope
- D. Call each API request in a separate Async scope

Answer: B

Explanation:

Scatter-Gather sends a request message to multiple targets concurrently. It collects the responses from all routes, and aggregates them into a single message.

NEW QUESTION 7

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 8

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 routeUse 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 applicationsUse 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 9

An organization's security requirements mandate centralized control at all times over authentication and authorization of external applications when invoking web APIs managed on Anypoint Platform.

What Anypoint Platform feature is most idiomatic (used for its intended purpose), straightforward, and maintainable to use to meet this requirement?

- A. Client management configured in access management
- B. Identity management configured in access management
- C. Enterprise Security module coded in Mule applications
- D. External access configured in API Manager

Answer: B

NEW QUESTION 10

An ABC Farms project team is planning to build a new API that is required to work with data from different domains across the organization.

The organization has a policy that all project teams should leverage existing investments by reusing existing APIs and related resources and documentation that other project teams have already developed and deployed.

To support reuse, where on Anypoint Platform should the project team go to discover and read existing APIs, discover related resources and documentation, and interact with mocked versions of those APIs?

- A. Design Center
- B. API Manager
- C. Runtime Manager
- D. Anypoint Exchange

Answer: D

Explanation:

The mocking service is a feature of Anypoint Platform and runs continuously. You can run the mocking service from the text editor, the visual editor, and from Anypoint Exchange. You can simulate calls to the API in API Designer before publishing the API specification to Exchange or in Exchange after publishing the API specification.

NEW QUESTION 10

A Mule application is deployed to a cluster of two(2) cusomter-hosted Mule runtimes. Currently the node name Alice is the primary node and node named bob is the secondary node. The mule application has a flow that polls a directory on a file system for new files.

The primary node Alice fails for an hour and then restarted.

After the Alice node completely restarts, from what node are the files polled, and what node is now the primary node for the cluster?

- A. Files are polled from Alice node Alice is now the primary node
- B. Files are polled from Bob node Alice is now the primary node
- C. Files are polled from Alice node Bob is now the primary node
- D. Files are polled from Bob node Bob is now the primary node

Answer: D

Explanation:

* Mule High Availability Clustering provides basic failover capability for Mule. * When the primary Mule Runtime becomes unavailable, for example, because of a fatal JVM or hardware failure or it's taken offline for maintenance, a backup Mule Runtime immediately becomes the primary node and resumes processing where the failed instance left off. * After a system administrator recovers a failed Mule Runtime server and puts it back online, that server automatically becomes the backup node. In this case, Alice, once up, will become backup

-----Reference: <https://docs.mulesoft.com/mule-runtime/4.3/hadr-guide> So correct choice is : Files are polled from Bob node Bob is now the primary node

NEW QUESTION 14

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 19

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 23

What aspects of a CI/CD pipeline for Mule applications can be automated using MuleSoft-provided Maven plugins?

- A. Compile, package, unit test, deploy, create associated API instances in API ManagerB Import from API designer, compile, package, unit test, deploy, publish to Anypoint Exchange
- B. Compile, package, unit test, validate unit test coverage, deploy
- C. Compile, package, unit test, deploy, integration test

Answer: C

NEW QUESTION 25

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 29

What is true about automating interactions with Anypoint Platform using tools such as Anypoint Platform REST API's, Anypoint CLI or the Mule Maven plugin?

- A. By default, the Anypoint CLI and Mule Maven plugin are not included in the Mule runtime

- B. Access to Anypoint Platform API's and Anypoint CLI can be controlled separately through the roles and permissions in Anypoint platform, so that specific users can get access to Anypoint CLI while others get access to the platform API's
- C. Anypoint Platform API's can only automate interactions with CloudHub while the Mule maven plugin is required for deployment to customer hosted Mule runtimes
- D. API policies can be applied to the Anypoint platform API's so that only certain LOS's has access to specific functions

Answer: A

Explanation:

Correct answer is By default, the Anypoint CLI and Mule Maven plugin are not included in the Mule runtime Maven is not part of runtime though it is part of studio. You do not need it to deploy in order to deploy your app. Same is the case with CLI.

NEW QUESTION 34

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 can't 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-based repeatable stream

NEW QUESTION 38

An organization is designing the following two Mule applications that must share data via a common persistent object store instance:

- Mule application P will be deployed within their on-premises datacenter.
- Mule application C will run on CloudHub in an Anypoint VPC.

The object store implementation used by CloudHub is the Anypoint Object Store v2 (OSv2).

What type of object store(s) should be used, and what design gives both Mule applications access to the same object store instance?

- A. Application P uses the Object Store connector to access a persistent object store. Application C accesses this persistent object store via the Object Store REST API through an IPsec tunnel
- B. Application C and P both use the Object Store connector to access the Anypoint Object Store v2
- C. Application C uses the Object Store connector to access a persistent object. Application P accesses the persistent object store via the Object Store REST API
- D. Application C and P both use the Object Store connector to access a persistent object store

Answer: C

Explanation:

Correct answer is Application A accesses the persistent object store via the Object Store REST API. Application B uses the Object Store connector to access a persistent object. * Object Store v2 lets CloudHub applications store data and states across batch processes, Mule components and applications, from within an application or by using the Object Store REST API. * On-premise Mule applications cannot use Object Store v2. * You can select Object Store v2 as the implementation for Mule 3 and Mule 4 in CloudHub by checking the Object Store V2 checkbox in Runtime Manager at deployment time. * CloudHub Mule applications can use Object Store connector to write to the object store. * The only way on-premises Mule applications can access Object Store v2 is via the Object Store REST API. * 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.

NEW QUESTION 39

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

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

Answer: D

NEW QUESTION 41

A new Mule application under development must implement extensive data transformation logic. Some of the data transformation functionality is already available as external transformation services that are mature and widely used across the organization; the rest is highly specific to the new Mule application.

The organization follows a rigorous testing approach, where every service and application must be extensively acceptance tested before it is allowed to go into production.

What is the best way to implement the data transformation logic for this new Mule application while minimizing the overall testing effort?

- A. Implement and expose all transformation logic as mlaoservices using DataWeave, so it can be reused by any application component that needs it, including the new Mule application
- B. Implement transformation logic in the new Mule application using DataWeave, replicating the transformation logic of existing transformation services
- C. Extend the existing transformation services with new transformation logic and invoke them from the new Mule application
- D. Implement transformation logic in the new Mule application using DataWeave, invoking existing transformation services when possible

Answer: D

Explanation:

Correct answer is Implement transformation logic in the new Mule application using DataWeave, invoking existing transformation services when possible. * The key here minimal testing effort, "Extend existing transformation logic" is not a feasible option because additional functionality is highly specific to the new Mule application so it should not be a part of commonly used functionality. So this option is ruled out. * "Implement transformation logic in the new Mule application using DataWeave, replicating the transformation logic of existing transformation services" Replicating the transformation logic of existing transformation services will cause duplicity of code. So this option is ruled out. * "Implement and expose all transformation logic as microservices using DataWeave, so it can be reused by any application component that needs it, including the new Mule application" as question specifies that the transformation is app specific and wont be used outside

NEW QUESTION 46

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 51

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 53

An insurance company is using a CloudHub runtime plane. As a part of requirement, email alert should be sent to internal operations team every time of policy applied to an API instance is deleted As an

integration architect suggest on how this requirement be met?

- A. Use audit logs in Anypoint platform to detect a policy deletion and configure the Audit logs alert feature to send an email to the operations team
- B. Use Anypoint monitoring to configure an alert that sends an email to the operations team every time a policy is deleted in API manager
- C. Create a custom connector to be triggered every time of policy is deleted in API manager
- D. Implement a new application that uses the Audit log REST API to detect the policy deletion and send an email to operations team the SMTP connector

Answer: D

NEW QUESTION 56

A company is implementing a new Mule application that supports a set of critical functions driven by a rest API enabled, claims payment rules engine hosted on oracle ERP. As designed the mule application requires many data transformation operations as it performs its batch processing logic.

The company wants to leverage and reuse as many of its existing java-based capabilities (classes, objects, data model etc.) as possible

What approach should be considered when implementing required data mappings and transformations between Mule application and Oracle ERP in the new Mule application?

- A. Create a new metadata RAML classes in Mule from the appropriate Java objects and then perform transformations via Dataweave
- B. From the mule application, transform via theXSLT model
- C. Transform by calling any suitable Java class from Dataweave
- D. Invoke any of the appropriate Java methods directly, create metadata RAML classes and then perform required transformations via Dataweave

Answer: C

NEW QUESTION 61

An organization is designing an integration Mule application to process orders by submitting them to a back-end system for offline processing. Each order will be received by the Mule application through an

HTTPS POST and must be acknowledged immediately. Once acknowledged, the order will be submitted to a back-end system. Orders that cannot be successfully submitted due to rejections from the back-end system will need to be processed manually (outside the back-end system).

The Mule application will be deployed to a customer-hosted runtime and is able to use an existing ActiveMQ broker if needed. The ActiveMQ broker is located inside the organization's firewall. The back-end system has a track record of unreliability due to both minor network connectivity issues and longer outages.

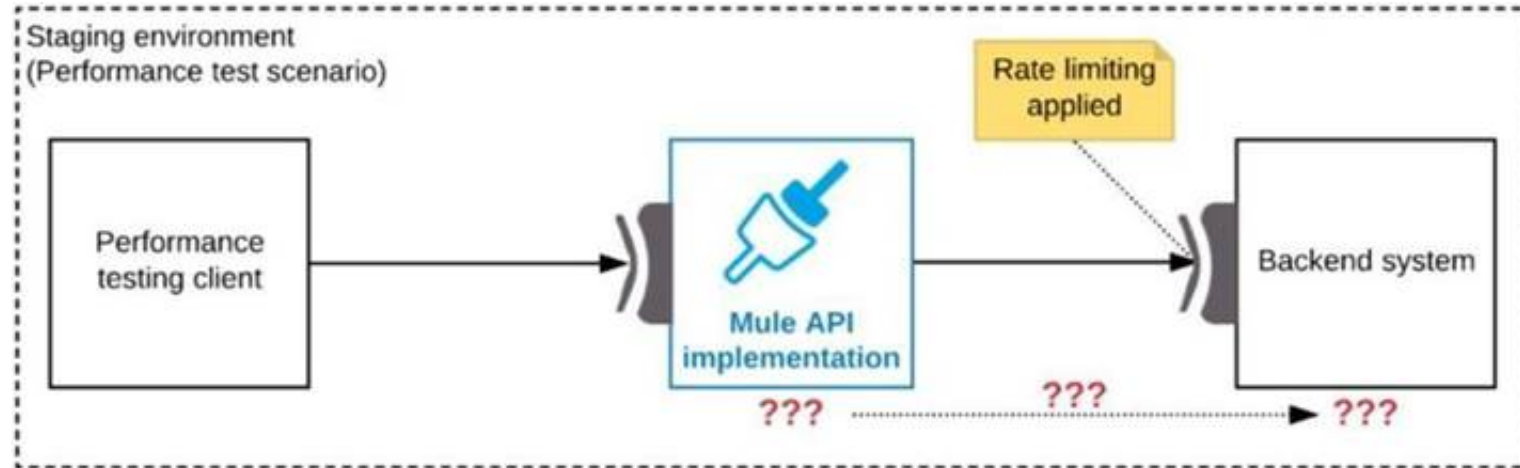
What idiomatic (used for their intended purposes) combination of Mule application components and ActiveMQ queues are required to ensure automatic submission of orders to the back-end system while supporting but minimizing manual order processing?

- A. An Until Successful scope to call the back-end system One or more ActiveMQ long-retry queues One or more ActiveMQ dead-letter queues for manual processing
- B. One or more On Error scopes to assist calling the back-end system An Until Successful scope containing VM components for long retries A persistent dead-letter VM queue configured in CloudHub
- C. One or more On Error scopes to assist calling the back-end system One or more ActiveMQ long-retry queues A persistent dead-letter object store configured in the CloudHub Object Store service
- D. A Batch Job scope to call the back-end system An Until Successful scope containing Object Store components for long retries A dead-letter object store configured in the Mule application

Answer: A

NEW QUESTION 63

Refer to the exhibit.



One of the backend systems invoked by an API implementation enforces rate limits on the number of requests a particular client can make. Both the backend system and the API implementation are deployed to several non-production environments in addition to production. Rate limiting of the backend system applies to all non-production environments. The production environment, however, does NOT have any rate limiting. What is the most effective approach to conduct performance tests of the API implementation in a staging (non-production) environment?

- A. Create a mocking service that replicates the backend system's production performance characteristics. Then configure the API implementation to use the mocking service and conduct the performance tests
- B. Use MUnit to simulate standard responses from the backend system then conduct performance tests to identify other bottlenecks in the system
- C. Include logic within the API implementation that bypasses invocations of the backend system in a performance test situation
- D. Instead invoking local stubs that replicate typical backend system responses then conduct performance tests using this API Implementation
- E. Conduct scaled-down performance tests in the staging environment against the rate limited backend system then upscale performance results to full production scale

Answer: A

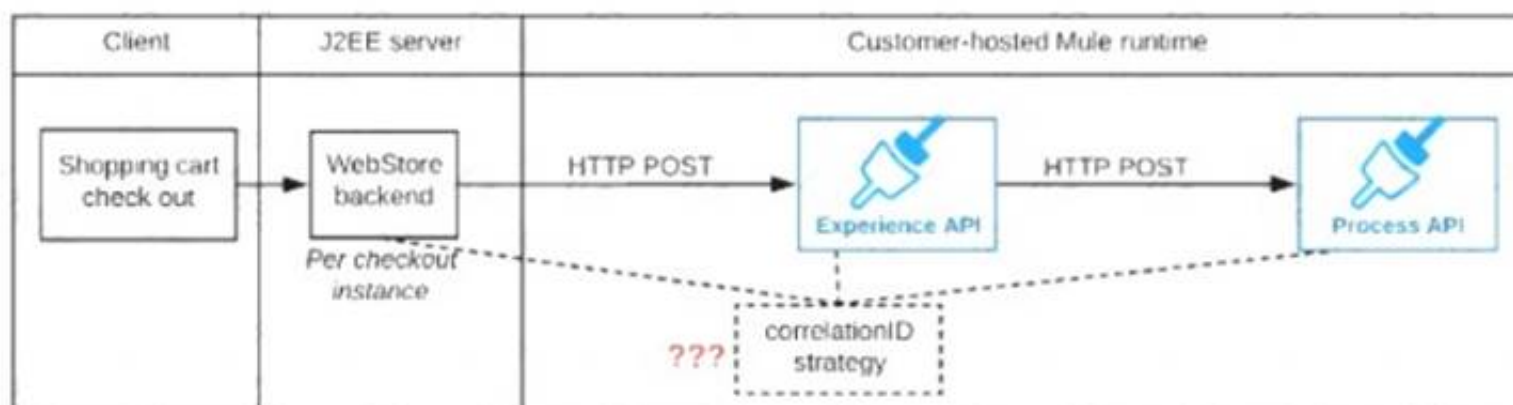
Explanation:

Correct answer is Create a mocking service that replicates the backend system's production performance characteristics. Then configure the API implementation to use the mocking service and conduct the performance tests

- * MUnit is for only Unit and integration testing for APIs and Mule apps. Not for performance Testing, even if it has the ability to Mock the backend.
- * Bypassing the backend invocation defeats the whole purpose of performance testing. Hence it is not a valid answer.
- * Scaled down performance tests cant be relied upon as performance of API's is not linear against load.

NEW QUESTION 68

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.

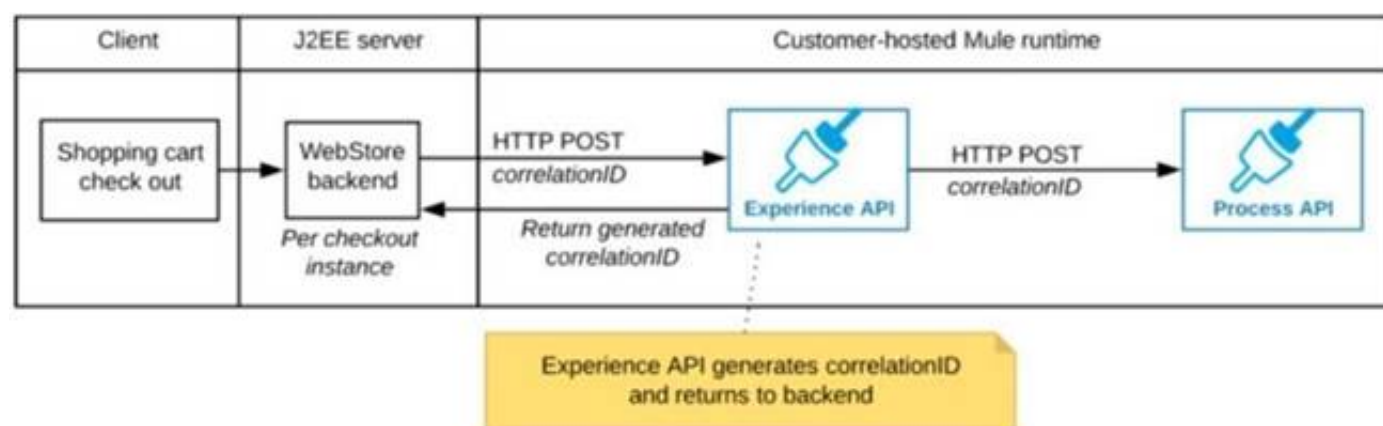
End-to-end correlation of all HTTP requests and responses belonging to each individual checkout Instance is required. This is to be done through a common correlation ID, so that all log entries written by the web store backend, Experience API implementation, and Process API implementation include the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

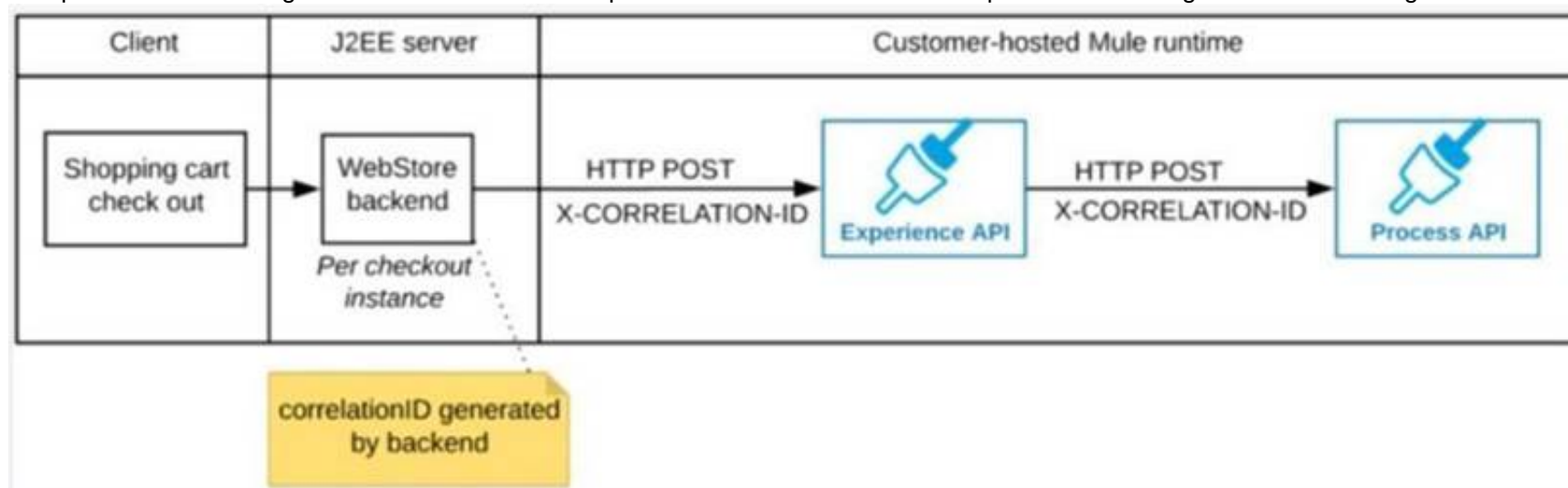
A)

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

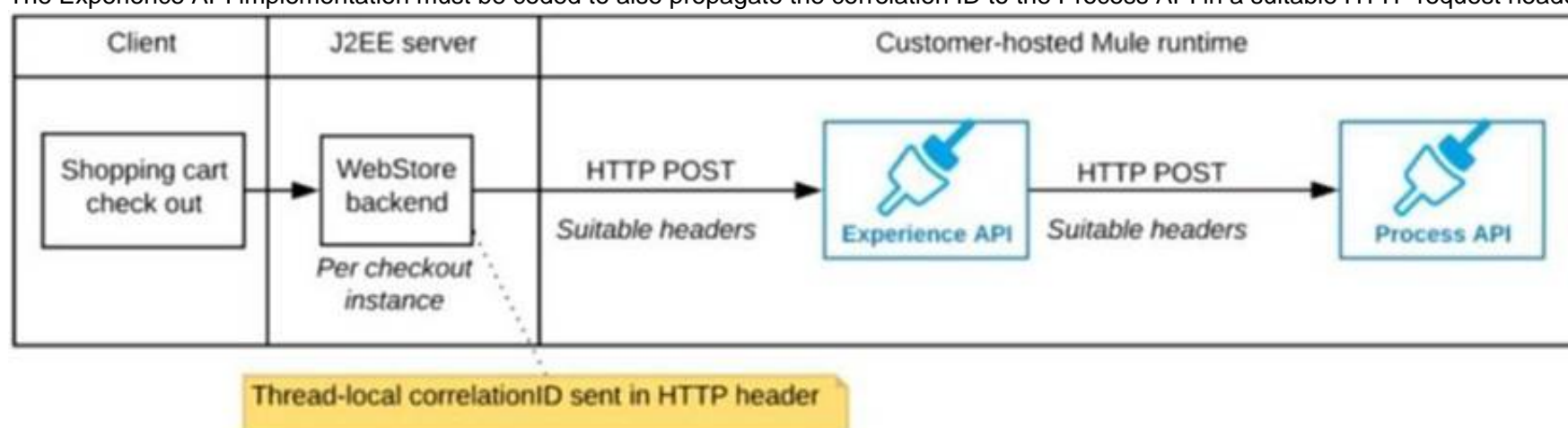
No special code or configuration is included in the web store backend, Experience API, and Process API implementations to generate and manage the correlation ID



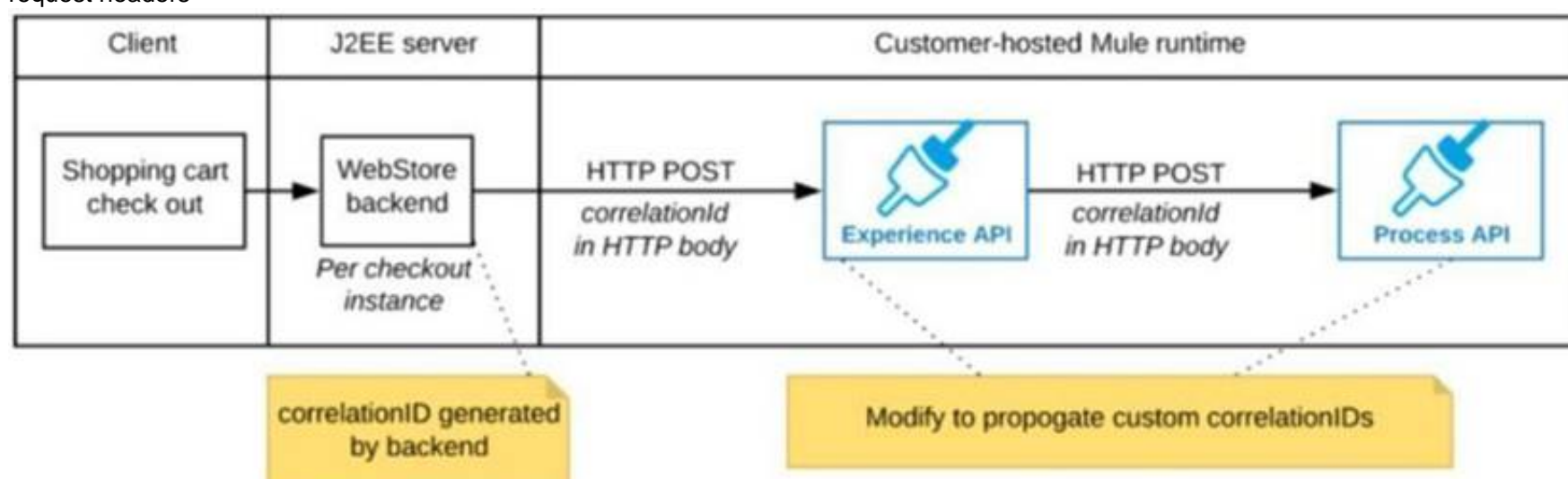
B)
 The web store backend generates a new correlation ID value at the start of checkout and sets it on the X-CORRELATION-Id HTTP request header In each API invocation belonging to that checkout
 No special code or configuration is included in the Experience API and Process API implementations to generate 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

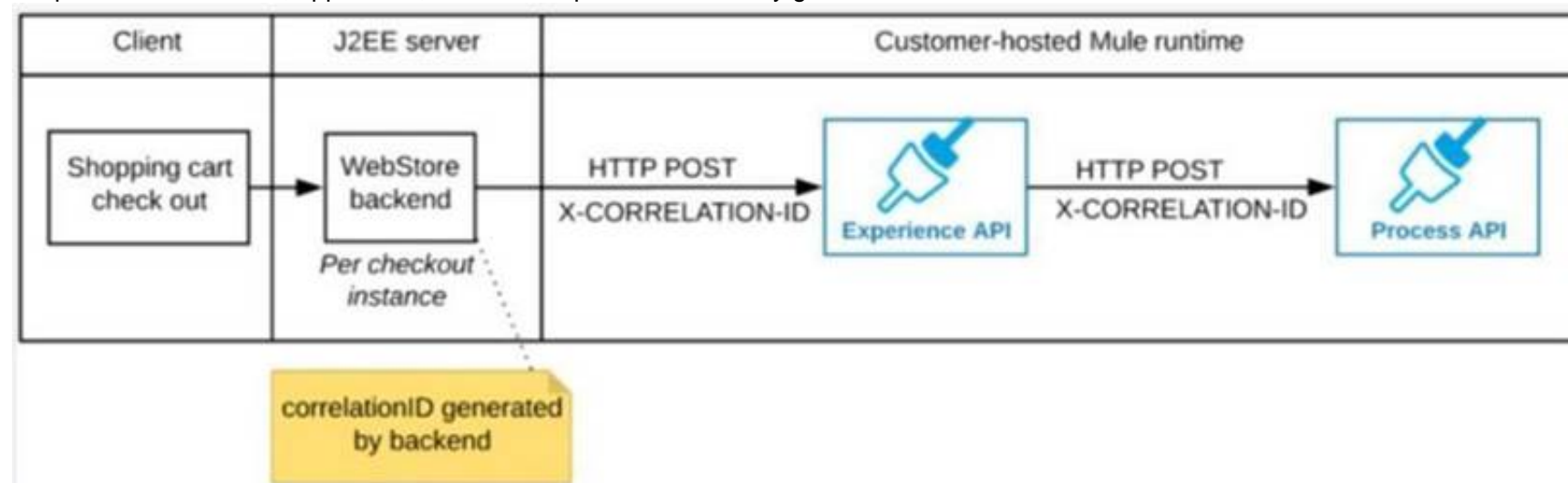
Answer: B

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-

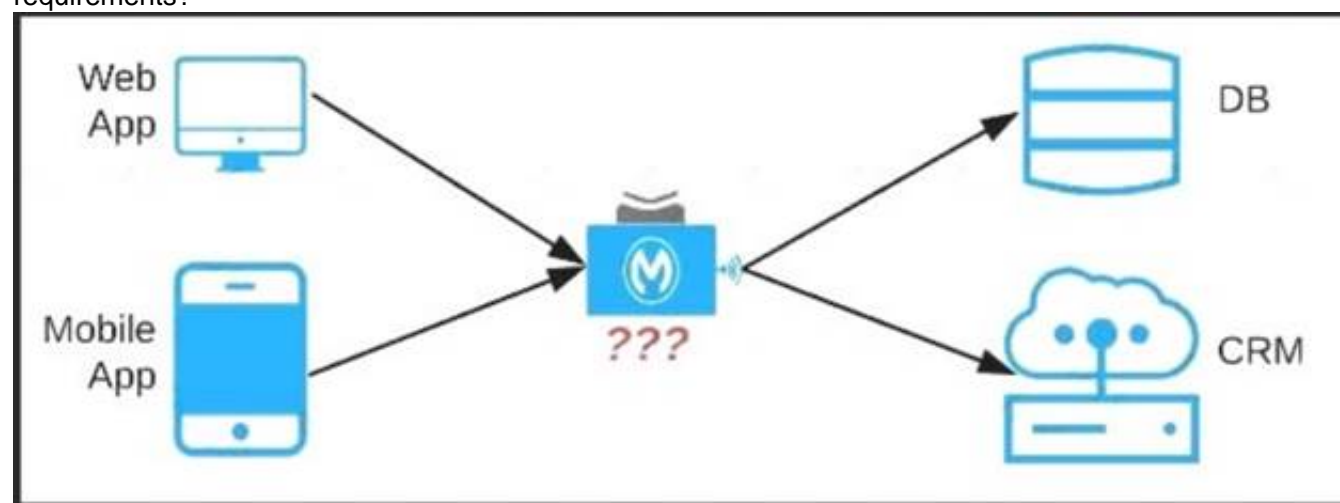
Correlation-Id" header is present, HTTP connector uses this as the Correlation Id. If "X-Correlation-Id" header is NOT present, a Correlation Id is randomly generated. For Incoming HTTP Requests: In order to set a custom Correlation Id, the client invoking the HTTP request must set "X-Correlation-Id" 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 72

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:

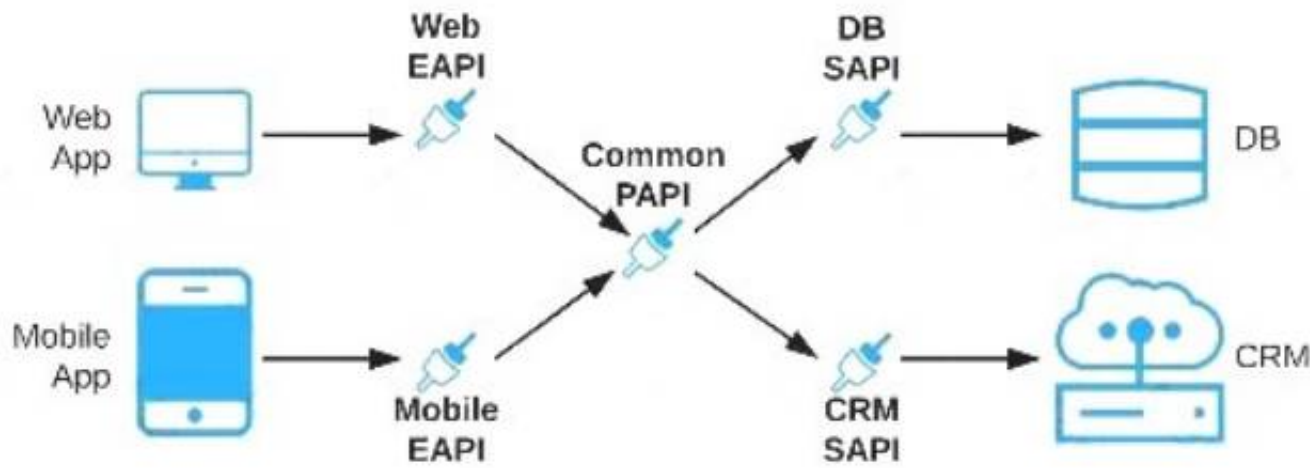
Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements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



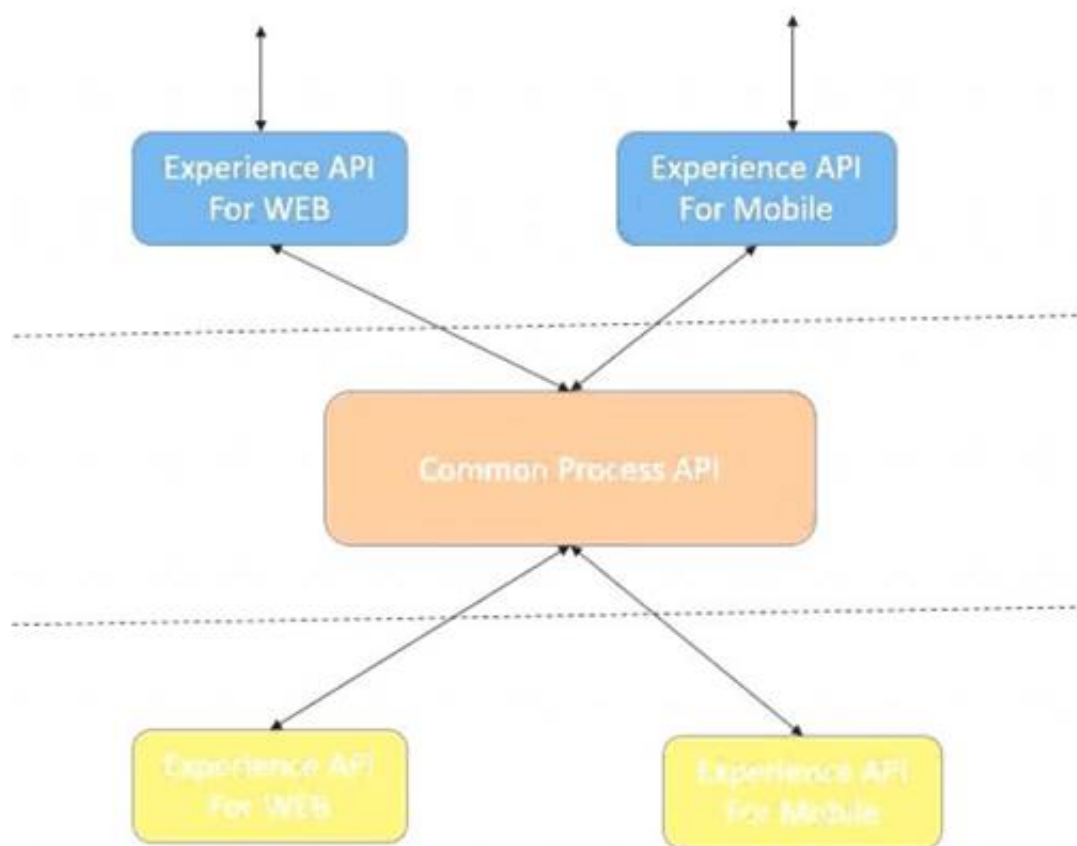
Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements 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 76

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

<bt: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 Requirement 1	All operations inside the transaction must belong to same Connector.	Operations inside the transaction may belong to different Connectors.
Connector Requirement 2	Connectors may not be XA enabled.	Connectors must be XA enabled.
Connector Requirement 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 case of system failure	No	Using Bitronix

NEW QUESTION 79

A project team is working on an API implementation using the RAML definition as a starting point. The team has updated the definition to include new operations and has published a new version to exchange. Meanwhile another team is working on a mule application consuming the same API implementation. During the development what has to be performed by the mule application team to take advantage of the newly added operations?

- A. Scaffold the client application with the new definition
- B. Scaffold API implementation application with the new definition
- C. Update the REST connector from exchange in the client application
- D. Update the API connector in the API implementation and publish to exchange

Answer: C

NEW QUESTION 80

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 85

A Mule application name Pub uses a persistence object store. The Pub Mule application is deployed to Cloudhub and it configured to use Object Store v2. Another Mule application name sub is being developed to retrieve values from the Pub Mule application persistence object Store and will also be deployed to cloudhub.

What is the most direct way for the Sub Mule application to retrieve values from the Pub Mule application persistence object store with the least latency?

- A. Use an object store connector configured to access the Pub Mule application persistence object store
- B. Use a VM connector configured to directly access the persistence queue of the Pub Mule application persistence object store.
- C. Use an Anypoint MQ connector configured to directly access the Pub Mule application persistence object store
- D. Use the Object store v2 REST API configured to access the Pub Mule application persistence object store.

Answer: D

Explanation:

* The Object Store V2 API enables API access to Anypoint Platform Object Store v2.

* 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. To share data between two Mule4 apps, use a queue in Anypoint MQ.

* The Object Store v2 APIs enable you to use REST to perform the following:

- Retrieve a list of object stores and keys associated with an application.
- Store and retrieve key-value pairs in an object store.
- Delete key-value pairs from an object store.
- Retrieve Object Store usage statistics for your organization.
- Object Store provides these APIs: Object Store API

Object Store Stats API

NEW QUESTION 89

A Mule application contains a Batch Job scope with several Batch Step scopes. The Batch Job scope is configured with a batch block size of 25.

A payload with 4,000 records is received by the Batch Job scope.

When there are no errors, how does the Batch Job scope process records within and between the Batch Step scopes?

- A. The Batch Job scope processes multiple record blocks in parallel, and a block of 25 records can jump ahead to the next Batch Step scope over an earlier block of records. Each Batch Step scope is invoked with one record in the payload of the received Mule event. For each Batch Step scope, all 25 records within a block are processed in parallel. All the records in a block must be completed before the block of 25 records is available to the next Batch Step scope.
- B. The Batch Job scope processes each record block sequentially, one at a time. Each Batch Step scope is invoked with one record in the payload of the received Mule event. For each Batch Step scope, all 25 records within a block are processed sequentially, one at a time. All 4000 records must be completed before the blocks of records are available to the next Batch Step scope.
- C. The Batch Job scope processes multiple record blocks in parallel, and a block of 25 records can jump ahead to the next Batch Step scope over an earlier block of records. Each Batch Step scope is invoked with one record in the payload of the received Mule event. For each Batch Step scope, all 25 records within a block are processed sequentially, one record at a time. All the records in a block must be completed before the block of 25 records is available to the next Batch Step scope.
- D. The Batch Job scope processes multiple record blocks in parallel. Each Batch Step scope is invoked with a batch of 25 records in the payload of the received Mule event. For each Batch Step scope, all 4000 records are processed in parallel. Individual records can jump ahead to the next Batch Step scope before the rest of the records finish processing in the current Batch Step scope.

Answer: A

NEW QUESTION 91

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>

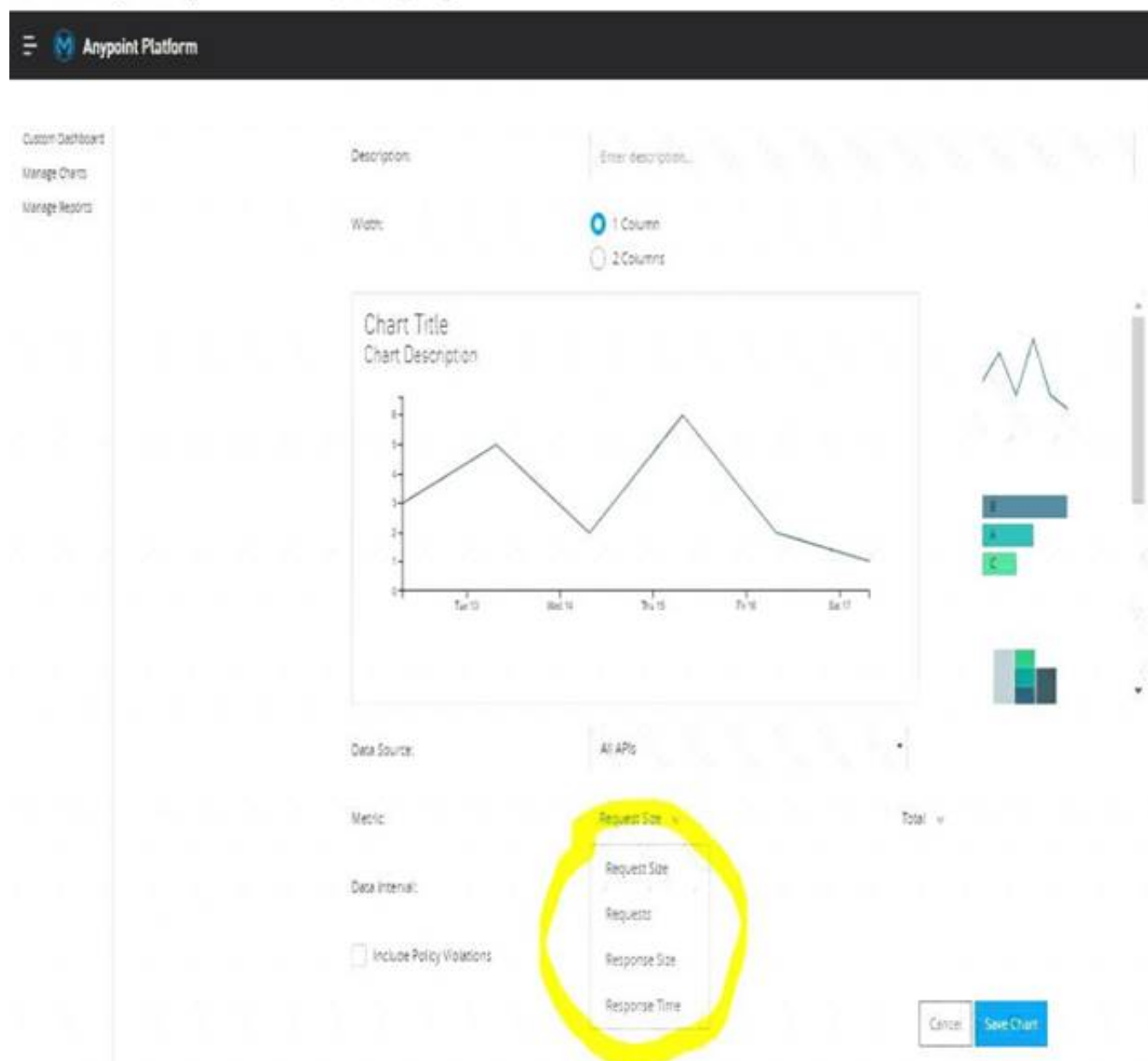
API Analytics provides a summary in chart form of requests, top apps, and latency for a particular duration. The custom dashboard in Anypoint Analytics contains a set of charts for a single API or for all APIs. Each

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 93

When using Anypoint Platform across various lines of business with their own Anypoint Platform business groups, what configuration of Anypoint Platform is always performed at the organization level as opposed to at the business group level?

- A. Environment setup
- B. Identity management setup
- C. Role and permission setup
- D. Dedicated Load Balancer setup

Answer: B

Explanation:

* Roles are business group specific. Configure identity management in the Anypoint Platform master organization. As the Anypoint Platform organization administrator, you can configure identity management in Anypoint Platform to set up users for single sign-on (SSO). * Roles and permissions can be set up at business group and organization level also. But Identity Management setup is only done at Organization level * Business groups are self-contained resource groups that contain Anypoint Platform resources such as applications and APIs. Business groups provide a way to separate and control access to Anypoint Platform resources because users have access only to the business group resources.

NEW QUESTION 94

An API implementation is being designed that must invoke an Order API which is known to repeatedly experience downtime. For this reason a fallback API is to be called when the Order API is unavailable. What approach to designing invocation of the fallback API provides the best resilience?

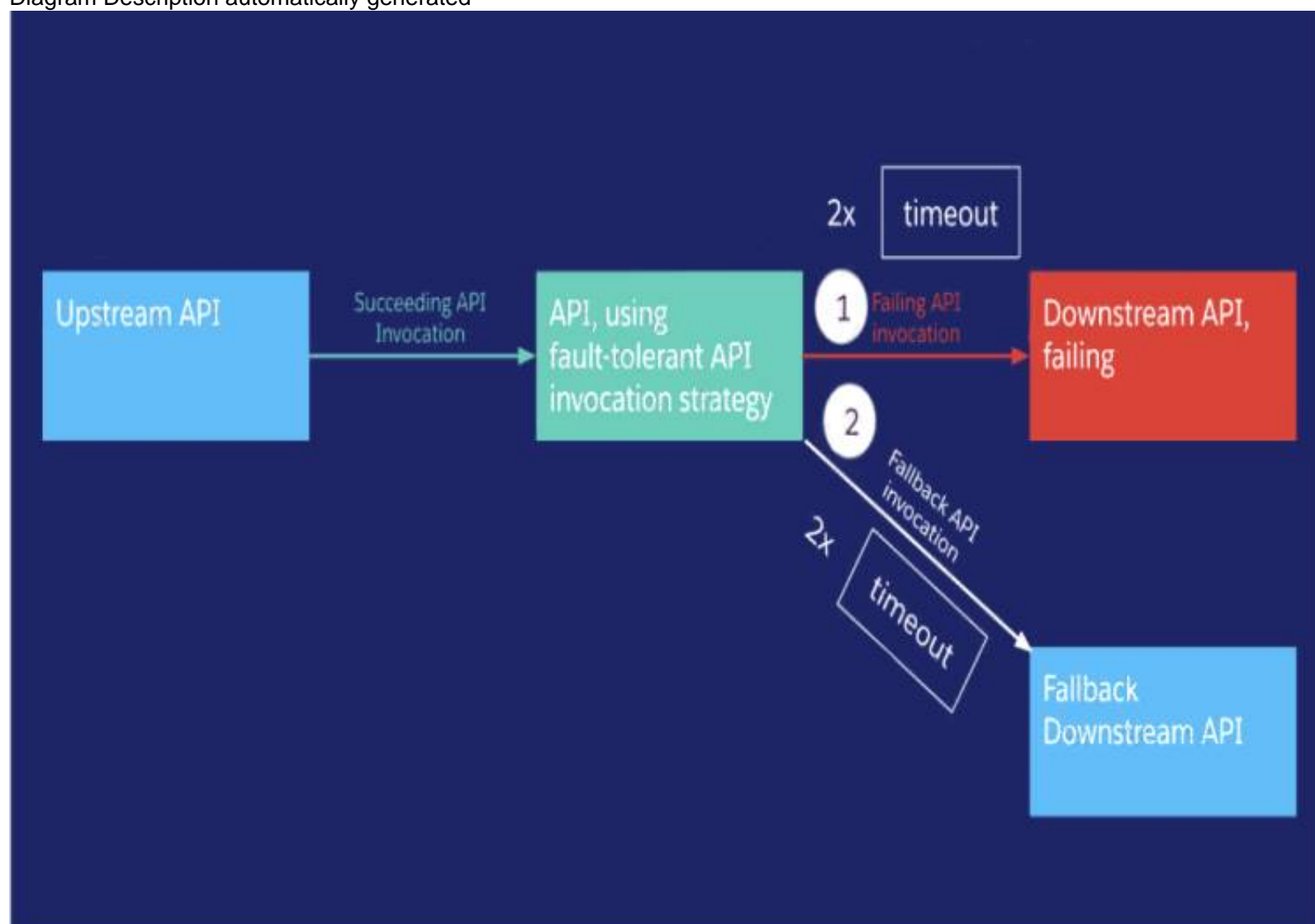
- A. Redirect client requests through an HTTP 303 temporary redirect status code to the fallback API whenever the Order API is unavailable
- B. Set an option in the HTTP Requester component that invokes the order API to instead invoke a fallback API whenever an HTTP 4XX or 5XX response status code is received from Order API
- C. Create a separate entry for the order API in API manager and then invoke this API as a fallback API if the primary Order API is unavailable
- D. Search Anypoint Exchange for a suitable existing fallback API and then implement invocations to their fallback API in addition to the Order API

Answer: A

Explanation:

* Resilience testing is a type of software testing that observes how applications act under stress. It's meant to ensure the product's ability to perform in chaotic conditions without a loss of core functions or data; it ensures a quick recovery after unforeseen, uncontrollable events.
 * In case an API invocation fails — even after a certain number of retries — it might be adequate to invoke a different API as a fallback. A fallback API, by definition, will never be ideal for the purpose of the API client, otherwise it would be the primary API.
 * Here are some examples for fallback APIs:
 - An old, deprecated version of the same API.
 - An alternative endpoint of the same API and version (e.g. API in another CloudHub region).
 - An API doing more than required, and therefore not as performant as the primary API.
 - An API doing less than required and therefore forcing the API Client to offer a degraded service, which is still better than no service at all.
 * API clients implemented as Mule applications offer the 'Until Successful Scope and Exception' strategies at their disposal, which together allow configuring fallback actions such as a fallback API invocation.
 * All HTTP response status codes within the 3xx category are considered redirection messages. These codes indicate to the user agent (i.e. your web browser)

that an additional action is required in order to complete the request and access the desired resource
 Diagram Description automatically generated



Hence correct answer is Redirect client requests through an HTTP 303 temporary redirect status code to the fallback API whenever the Order API is unavailable

NEW QUESTION 96

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 101

In one of the critical payment related mule application, transaction is being used . As an enhancement to implementation , scatter gather route is introduced which is also the part of transaction group. Scatter gather route has 4 routes.
 What will be the behavior of the Mule application in case of error occurs in 4th route of the scatter-gather router and transaction needs to be rolled back?

- A. Only errored route will be rolled back
- B. All routes will be rolled back
- C. Scatter Gather router cannot be part of transaction

Answer: B

Explanation:

•Scatter Gather: When running within a transaction, Scatter Gather does not execute in parallel. This means that the second route is executed after the first one is

processed, the third after the second one, etc. In case of error, all routes will be rolled back

NEW QUESTION 106

An organization has deployed runtime fabric on an eight node cluster with performance profile. An API uses and non persistent object store for maintaining some of its state data. What will be the impact to the state data if server crashes?

- A. State data is preserved
- B. State data is rolled back to a previously saved version
- C. State data is lost
- D. State data is preserved as long as more than one more is unaffected by the crash

Answer: D

NEW QUESTION 107

A retailer is designing a data exchange interface to be used by its suppliers. The interface must support secure communication over the public internet. The interface must also work with a wide variety of programming languages and IT systems used by suppliers.

What are suitable interface technologies for this data exchange that are secure, cross-platform, and internet friendly, assuming that Anypoint Connectors exist for these interface technologies?

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

Answer: C

Explanation:

As per definition of API by Mulesoft , it is Application Programming Interface using HTTP-based protocols. Non-HTTP-based programmatic interfaces are not APIs.

* HTTP-based programmatic interfaces are APIs even if they don't use REST or JSON. Hence implementation based on Java RMI, CORBA/IIOP, raw TCP/IP interfaces are not API's as they are not using HTTP.

* One more thing to note is FTP was not built to be secure. It is generally considered to be an insecure protocol because it relies on clear-text usernames and passwords for authentication and does not use encryption.

* Data sent via FTP is vulnerable to sniffing, spoofing, and brute force attacks, among other basic attack methods.

Considering the above points only correct option is

- XML over ActiveMQ
- XML over SFTP
- XML/REST over HTTPS

NEW QUESTION 110

An organization has chosen Mulesoft for their integration and API platform.

According to the Mulesoft catalyst framework, what would an integration architect do to create achievement goals as part of their business outcomes?

- A. Measure the impact of the centre for enablement
- B. build and publish foundational assets
- C. agree upon KPI's and help develop and overall success plan
- D. evangelize API's

Answer: C

NEW QUESTION 111

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:

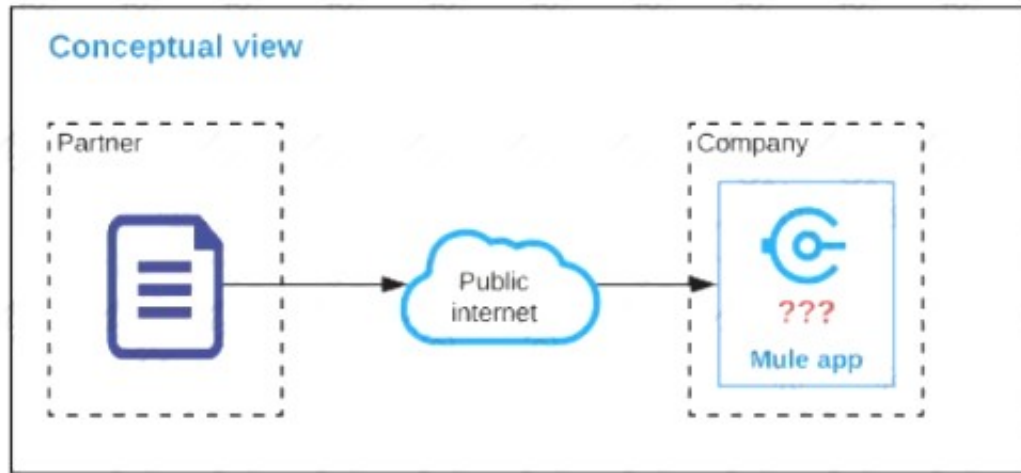
Object Store Connector is a Mule component that allows for simple key-value storage. Although it can serve a wide variety of use cases, it is mainly design for: - Storing synchronization information, such as watermarks. - Storing temporal information such as access tokens. - Storing user information. 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 115

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 116

Mule applications need to be deployed to CloudHub so they can access on-premises database systems. These systems store sensitive and hence tightly protected data, so are not accessible over the internet.

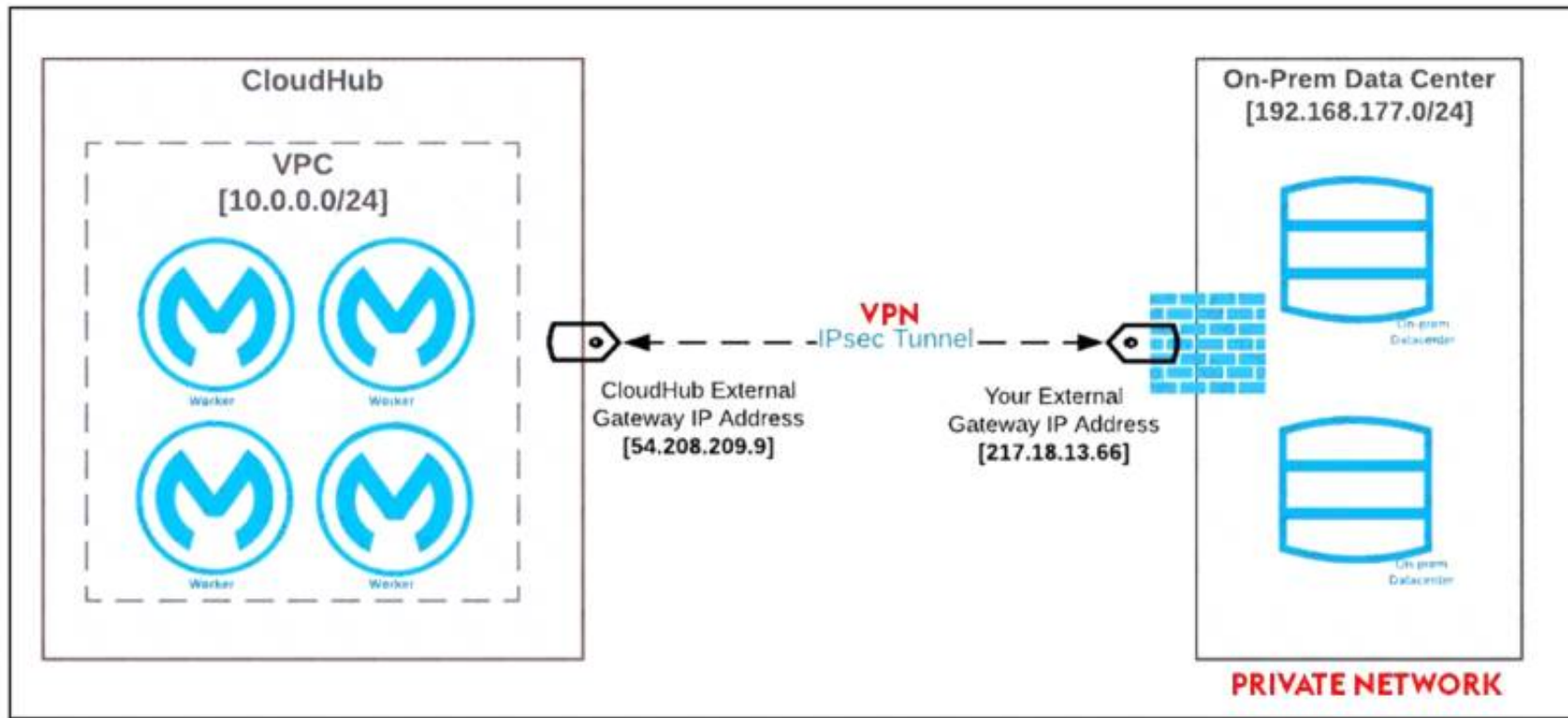
What network architecture supports this requirement?

- A. An Anypoint VPC connected to the on-premises network using an IPsec tunnel or AWS DirectConnect, plus matching firewall rules in the VPC and on-premises network
- B. Static IP addresses for the Mule applications deployed to the CloudHub Shared Worker Cloud, plus matching firewall rules and IPwhitelisting in the on-premises network
- C. An Anypoint VPC with one Dedicated Load Balancer fronting each on-premises database system, plus matching IP whitelisting in the load balancer and firewall rules in the VPC and on-premises network
- D. Relocation of the database systems to a DMZ in the on-premises network, with Mule applications deployed to the CloudHub Shared Worker Cloud connecting only to the DMZ

Answer: A

Explanation:

- * "Relocation of the database systems to a DMZ in the on-premises network, with Mule applications deployed to the CloudHub Shared Worker Cloud connecting only to the DMZ" is not a feasible option
 - * "Static IP addresses for the Mule applications deployed to the CloudHub Shared Worker Cloud, plus matching firewall rules and IP whitelisting in the on-premises network" - It is risk for sensitive data. - Even if you whitelist the database IP on your app, your app wont be able to connect to the database so this is also not a feasible option
 - * "An Anypoint VPC with one Dedicated Load Balancer fronting each on-premises database system, plus matching IP whitelisting in the load balancer and firewall rules in the VPC and on-premises network" Adding one VPC with a DLB for each backend system also makes no sense, is way too much work. Why would you add a LB for one system.
 - * Correct Answer "An Anypoint VPC connected to the on-premises network using an IPsec tunnel or AWS DirectConnect, plus matching firewall rules in the VPC and on-premises network"
- IPsec Tunnel You can use an IPsec tunnel with network-to-network configuration to connect your on-premises data centers to your Anypoint VPC. An IPsec VPN tunnel is generally the recommended solution for VPC to on-premises connectivity, as it provides a standardized, secure way to connect. This method also integrates well with existing IT infrastructure such as routers and appliances.
- Reference: <https://docs.mulesoft.com/runtime-manager/vpc-connectivity-methods-concept>
- Diagram Description automatically generated



NEW QUESTION 118

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 flowOn Error Continue error handler in the child flow
- B. For Each scope in the parent flowOn Error Continue error handler in the child flow
- C. Parallel For Each scope in the parent flowOn Error Propagate error handler in the child flow
- D. Until Successful router in the parent flowOn 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 120

What API policy would LEAST likely be applied to a Process API?

- A. Custom circuit breaker
- B. Client ID enforcement
- C. Rate limiting
- D. JSON threat protection

Answer: D

Explanation:

Key to this question lies in the fact that Process API are not meant to be accessed directly by clients. Lets analyze options one by one. Client ID enforcement : This is applied at process API level generally to ensure that identity of API clients is always known and available for API-based analytics Rate Limiting : This policy is applied on Process Level API to secure API's against degradation of service that can happen in case load received is more than it can handle Custom circuit breaker : This is also quite useful feature on process level API's as it saves the API client the wasted time and effort of invoking a failing API. JSON threat protection : This policy is not required at Process API and rather implemented as Experience API's. This policy is used to safeguard application from malicious attacks by injecting malicious code in JSON object. As ideally Process API's are never called from external world , this policy is never used on Process API's Hence correct answer is

JSON threat protection MuleSoft Documentation Reference : <https://docs.mulesoft.com/api-manager/2.x/policy-mule3-json-threat>

NEW QUESTION 122

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), fo 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 rewrittenDevOps 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 rewrittenDevOps MUST STILL manage port conflicts
- D. NO, the Mule applications do NO need to be rewrittenDevOps 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 127

An organization is successfully using API led connectivity, however, as the application network grows, all the manually performed tasks to publish share and discover, register, apply policies to, and deploy an API are becoming repetitive pictures driving the organization to automate this process using efficient CI/CD pipeline. Considering Anypoint platforms capabilities how should the organization approach automating is API lifecycle?

- A. Use runtime manager rest apis for API management and mavenforAPI deployment
- B. Use Maven with a custom configuration required for the API lifecycle
- C. Use Anypoint CLI or Anypoint Platform REST apis with scripting language such as groovy
- D. Use Exchange rest api's for API management and MavenforAPI deployment

Answer: D

NEW QUESTION 129

An organization is sizing an Anypoint VPC to extend their internal network to Cloudhub.

For this sizing calculation, the organization assumes 150 Mule applications will be deployed among three(3) production environments and will use Cloudhub's default zero-downtime feature. Each Mule application is expected to be configured with two(2) Cloudhub workers. This is expected to result in several Mule application deployments per hour.

- A. 10.0.0.0/21(2048 IPs)
- B. 10.0.0.0/22(1024IPs)
- C. 10.0.0.0/23(512 IPs)
- D. 10.0.0.0/24(256 IPs)

Answer: A

Explanation:

- * When you create an Anypoint VPC, the range of IP addresses for the network must be specified in the form of a Classless Inter-Domain Routing (CIDR) block, using CIDR notation.
- * This address space is reserved for Mule workers, so it cannot overlap with any address space used in your data center if you want to peer it with your VPC.
- * To calculate the proper sizing for your Anypoint VPC, you first need to understand that the number of dedicated IP addresses is not the same as the number of workers you have deployed.
- * For each worker deployed to CloudHub, the following IP assignation takes place: For better fault tolerance, the VPC subnet may be divided into up to four Availability Zones.
- * A few IP addresses are reserved for infrastructure. At least two IP addresses per worker to perform at zero-downtime.
- * Hence in this scenario 2048 IP's are required to support the requirement.

NEW QUESTION 134

One of the backend systems involved by the API implementation enforces rate limits on the number of request a particle client can make.

Both the back-end system and API implementation are deployed to several non-production environments including the staging environment and to a particular production environment. Rate limiting of the back-end system applies to all non-production environments.

The production environment however does not have any rate limiting.

What is the cost-effective approach to conduct performance test of the API implementation in the non-production staging environment?

- A. Including logic within the API implementation that bypasses in locations of the back-end system in the staging environment and invoke a Mocking service that replicates typical back-end system responses Then conduct performance test using this API implementation
- B. Use MUnit to simulate standard responses from the back-end system. Then conduct performance test to identify other bottlenecks in the system
- C. Create a Mocking service that replicates the back-end system's production performance characteristics Then configure the API implementation to use the mocking service and conduct the performance test
- D. Conduct scaled-down performance tests in the staging environment against rate-limiting back-end system
- E. Then upscale performance results to full production scale

Answer: C

NEW QUESTION 139

An external web UI application currently accepts occasional HTTP requests from client web browsers to change (insert, update, or delete) inventory pricing information in an inventory system's database. Each inventory pricing change must be transformed and then synchronized with multiple customer experience systems in near real-time (in under 10 seconds). New customer experience systems are expected to be added in the future.

The database is used heavily and limits the number of SELECT queries that can be made to the database to 10 requests per hour per user.

What is the most scalable, idiomatic (used for its intended purpose), decoupled, reusable, and maintainable integration mechanism available to synchronize each inventory pricing change with the various customer experience systems in near real-time?

- A. Write a Mule application with a Database On Table Row event source configured for the inventory pricing database, with the watermark attribute set to an appropriate database column In the same flow, use a Scatter-Gather to call each customer experience system's REST API with transformed inventory-pricing records
- B. Add a trigger to the inventory-pricing database table so that for each change to the inventory pricing database, a stored procedure is called that makes a REST call to a Mule application Write the Mule application to publish each Mule event as a message to an Anypoint MQ exchange Write other Mule applications to subscribe to the Anypoint MQ exchange, transform each received message, and then update the Mule application's corresponding customer experience system(s)
- C. Replace the external web UI application with a Mule application to accept HTTP requests from client web browsers In the same Mule application, use a Batch Job scope to test if the database request will succeed, aggregate pricing changes within a short time window, and then update both the inventory pricing database and each customer experience system using a Parallel For Each scope
- D. Write a Mule application with a Database On Table Row event source configured for the inventory pricing database, with the ID attribute set to an appropriate database column In the same flow, use a Batch Job scope to publish transformed Inventory-pricing records to an Anypoint MQ queue Write other Mule applications

to subscribe to the Anypoint MQ queue, transform each received message, and then update the Mule application's corresponding customer experience system(s)

Answer: B

NEW QUESTION 143

A marketing organization is designing a Mule application to process campaign data. The Mule application will periodically check for a file in a SFTP location and process the records in the file. The size of the file can vary from 10MB to 5GB. Due to the limited availability of vCores, the Mule application is deployed to a single CloudHub worker configured with vCore size 0.2.

The application must transform and send different formats of this file to three different downstream SFTP locations.

What is the most idiomatic (used for its intended purpose) and performant way to configure the SFTP operations or event sources to process the large files to support these deployment requirements?

- A. Use an in-memory repeatable stream
- B. Use a file-stored non-repeatable stream
- C. Use an in-memory non-repeatable stream
- D. Use a file-stored repeatable stream

Answer: A

NEW QUESTION 145

An API has been updated in Anypoint Exchange by its API producer from version 3.1.1 to 3.2.0 following accepted semantic versioning practices and the changes have been communicated via the API's public portal. The API endpoint does NOT change in the new version. How should the developer of an API client respond to this change?

- A. The update should be identified as a project risk and full regression testing of the functionality that uses this API should be run.
- B. The API producer should be contacted to understand the change to existing functionality.
- C. The API producer should be requested to run the old version in parallel with the new one.
- D. The API client code ONLY needs to be changed if it needs to take advantage of new features.

Answer: D

Explanation:

* Semantic Versioning is a 3-component number in the format of X.Y.Z, where : X stands for a major version.

Y stands for a minor version:

Z stands for a patch.

So, SemVer is of the form Major.Minor.Patch Coming to our question , minor version of the API has been changed which is backward compatible. Hence there is no change required on API client end. If they want to make use of new featured that have been added as a part of minor version change they may need to change code at their end. Hence correct answer is The API client code ONLY needs to be changed if it needs to take advantage of new features.

Diagram Description automatically generated



NEW QUESTION 150

What aspects of a CI/CD pipeline for Mule applications can be automated using MuleSoft-provided Maven plugins?

- A. Compile, package, unit test, validate unit test coverage, deploy
- B. Compile, package, unit test, deploy, integration test (Incorrect)
- C. Compile, package, unit test, deploy, create associated API instances in API Manager
- D. Import from API designer, compile, package, unit test, deploy, publish to Anypoint Exchange

Answer: A

Explanation:

Correct answer is "Compile, package, unit test, validate unit test coverage, deploy"

Anypoint Platform supports continuous integration and continuous delivery using industry standard tools Mule Maven Plugin The Mule Maven plugin can automate building, packaging and deployment of Mule applications from source projects Using the Mule Maven plugin, you can automate your Mule application deployment to CloudHub, to Anypoint Runtime Fabric, or on-premises, using any of the following deployment strategies • CloudHub deployment • Runtime Fabric deployment • Runtime Manager REST API deployment • Runtime Manager agent deployment MUnit Maven Plugin The MUnit Maven plugin can automate test execution, and ties in with the Mule Maven plugin. It provides a full suite of integration and unit test capabilities, and is fully integrated with Maven and Surefire for integration with your continuous deployment environment. Since MUnit 2.x, the coverage report goal is integrated with the maven reporting section. Coverage Reports are generated during Maven's site lifecycle, during the coverage-report goal. One of the features of MUnit Coverage is to fail the build if a certain coverage level is not reached. MUnit is not used for integration testing Also publishing to Anypoint Exchange or to create associated API instances in API Manager is not a part of CI/CD pipeline which can ne achieved using mulesoft provided maven plugin

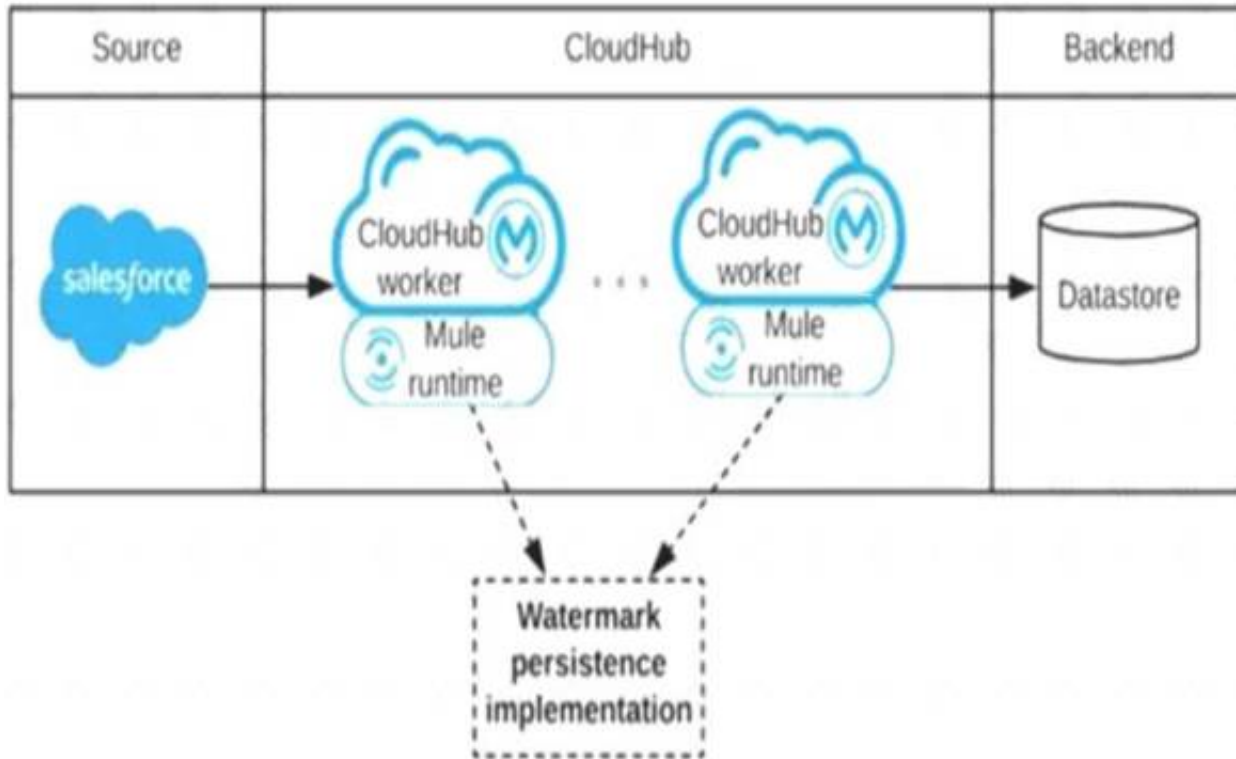
Explanation

Architecture mentioned in the question can be diagrammatically put as below. Persistent Object Store is the correct answer .

* Mule Object Stores: 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. So we cant use in memory store in our scenario as we want to share watermark within all cloudhub workers
- 2) Persistent store – Mule persists data when an object store is explicitly configured to be persistent. Hence this watermark will be available even any of the worker goes down

Diagram Description automatically generated



NEW QUESTION 152

A team would like to create a project skeleton that developers can use as a starting point when creating API Implementations with Anypoint Studio. This skeleton should help drive consistent use of best practices within the team.

What type of Anypoint Exchange artifact(s) should be added to Anypoint Exchange to publish the project skeleton?

- A. A custom asset with the default API implementation
- B. A RAML archetype and reusable trait definitions to be reused across API implementations
- C. An example of an API implementation following best practices
- D. a Mule application template with the key components and minimal integration logic

Answer: D

Explanation:

* Sharing Mule applications as templates is a great way to share your work with other people who are in your organization in Anypoint Platform. When they need to build a similar application they can create the mule application using the template project from Anypoint studio.

* Anypoint Templates are designed to make it easier and faster to go from a blank canvas to a production application. They're bit for bit Mule applications requiring only Anypoint Studio to build and design, and are deployable both on-premises and in the cloud.

* Anypoint Templates are based on five common data Integration patterns and can be customized and extended to fit your integration needs. So even if your use case involves different endpoints or connectors than those included in the template, they still offer a great starting point.

Some of the best practices while creating the template project: - Define the common error handler as part of template project, either using pom dependency or mule config file - Define common logger/audit framework as part of the template project - Define the env specific properties and secure properties file as per the requirement - Define global.xml for global configuration - Define the config file for connector configuration like Http,Salesforce,File,FTP etc - Create separate folders to create DWL,Properties,SSL certificates etc - Add the dependency and configure the pom.xml as per the business need - Configure the mule-artifact.json as per the business need

NEW QUESTION 156

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 158

What Is a recommended practice when designing an integration Mule 4 application that reads a large XML payload as a stream?

- A. The payload should be dealt with as a repeatable XML stream, which must only be traversed (iterated-over) once and CANNOT be accessed randomly from DataWeave expressions and scripts
- B. The payload should be dealt with as an XML stream, without converting it to a single Java object (POJO)
- C. The payload size should NOT exceed the maximum available heap memory of the Mule runtime on which the Mule application executes
- D. The payload must be cached using a Cache scope If It Is to be sent to multiple backend systems

Answer: C

Explanation:

If the size of the stream exceeds the maximum, a `STREAM_MAXIMUM_SIZE_EXCEEDED` error is raised.

NEW QUESTION 162

A company wants its users to log in to Anypoint Platform using the company's own internal user credentials. To achieve this, the company needs to integrate an external identity provider (IdP) with the company's Anypoint Platform master organization, but SAML 2.0 CANNOT be used. Besides SAML 2.0, what single-sign-on standard can the company use to integrate the IdP with their Anypoint Platform master organization?

- A. SAML 1.0
- B. OAuth 2.0
- C. Basic Authentication
- D. OpenID Connect

Answer: D

Explanation:

As the Anypoint Platform organization administrator, you can configure identity management in Anypoint Platform to set up users for single sign-on (SSO). Configure identity management using one of the following single sign-on standards:

- 1) OpenID Connect: End user identity verification by an authorization server including SSO
- 2) SAML 2.0: Web-based authorization including cross-domain SSO

NEW QUESTION 165

What is not true about Mule Domain Project?

- A. This allows Mule applications to share resources
- B. Expose multiple services within the Mule domain on the same port
- C. Only available Anypoint Runtime Fabric
- D. Send events (messages) to other Mule applications using VM queues

Answer: C

Explanation:

* Mule Domain Project is ONLY available for customer-hosted Mule runtimes, but not for Anypoint Runtime Fabric

* Mule domain project is available for Hybrid and Private Cloud (PCE). Rest all provide application isolation and can't support domain project.

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:

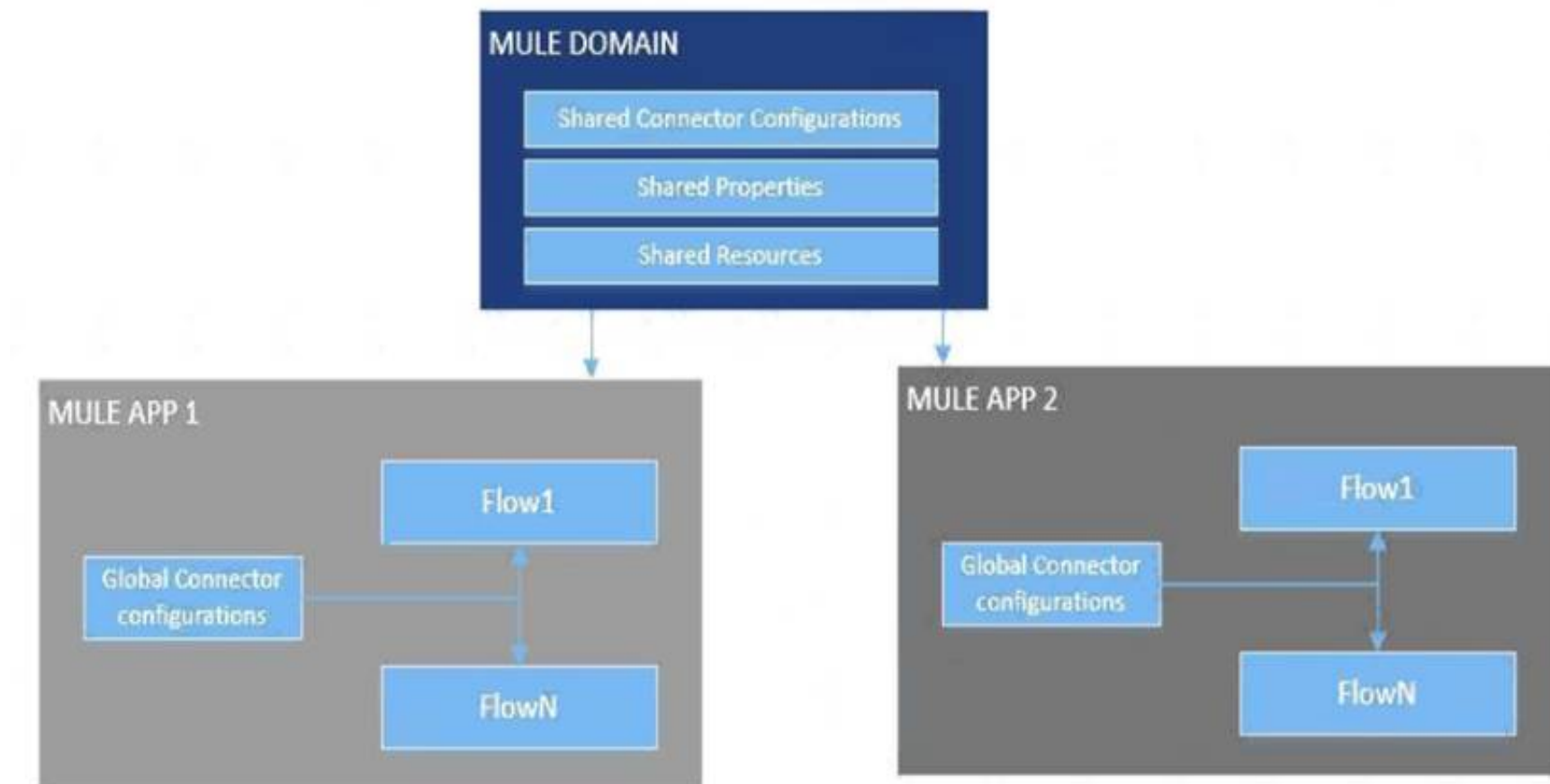
- Expose multiple services within the domain through the same port.
- Share the connection to persistent storage.
- Share services between apps through a well-defined interface.
- Ensure consistency between apps upon any changes because the configuration is only set in one place.

* 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



NEW QUESTION 169

An insurance provider is implementing Anypoint platform to manage its application infrastructure and is using the customer hosted runtime for its business due to certain financial requirements it must meet. It has built a number of synchronous API's and is currently hosting these on a mule runtime on one server. These applications make use of a number of components including heavy use of object stores and VM queues. Business has grown rapidly in the last year and the insurance provider is starting to receive reports of reliability issues from its applications.

The DevOps team indicates that the API's are currently handling too many requests and this is over loading the server. The team has also mentioned that there is a significant downtime when the server is down for maintenance.

As an integration architect, which option would you suggest to mitigate these issues?

- A. Add a load balancer and add additional servers in a server group configuration
- B. Add a load balancer and add additional servers in a cluster configuration
- C. Increase physical specifications of server CPU memory and network
- D. Change applications by use an event-driven model

Answer: B

NEW QUESTION 170

An organization is designing a mule application to support an all or nothing transaction between several 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

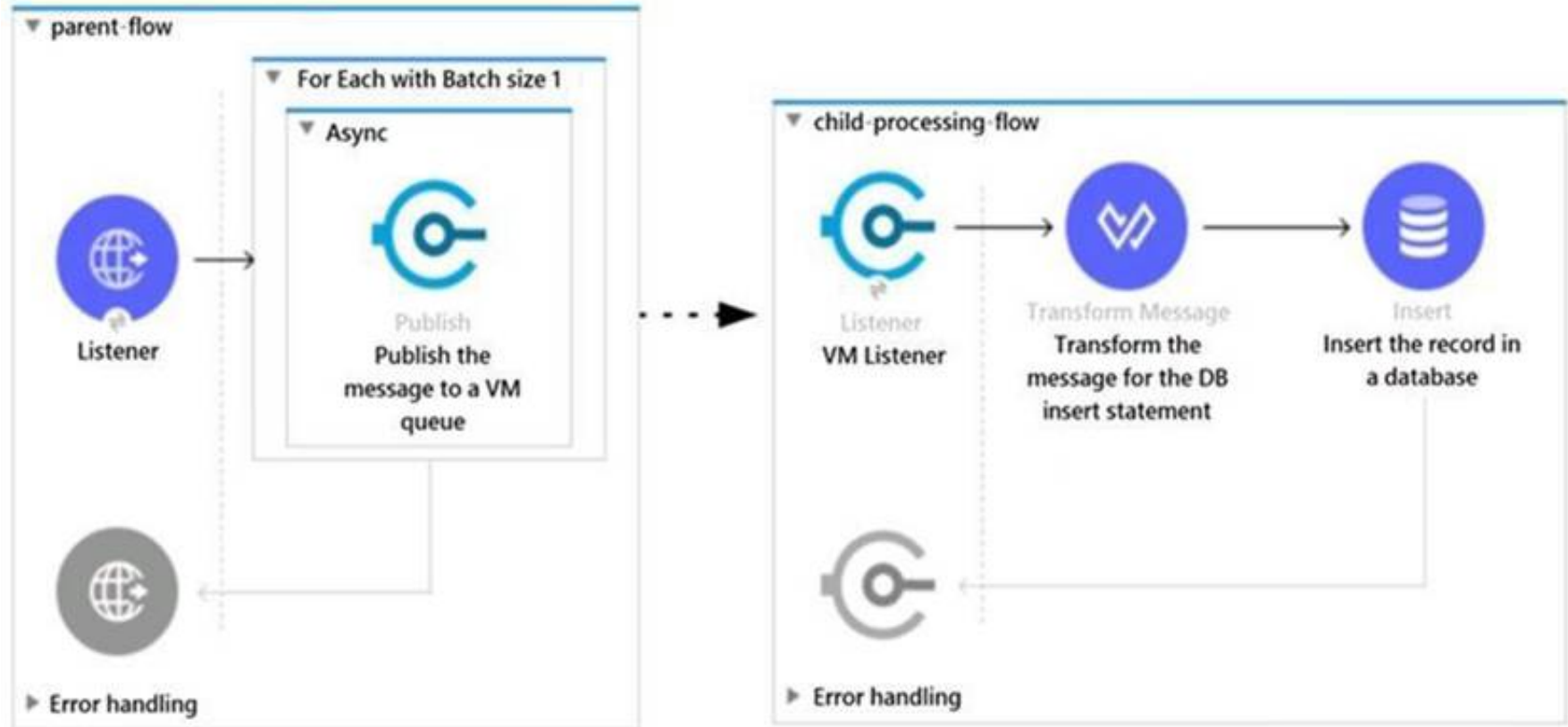
Explanation:

Correct answer is VM. VM supports Transactional Type. When an exception occurs, 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 <small>https://docs.mulesoft.com/runtime-manager/whitelists</small>	Yes
Configure Custom Domain	No	Yes
Custom Certificate	No	Yes
Rate Limit	Lower Rate Limit and applied According to Region	Higher Rate Limit Threshold
VPC	Anypoint VPC optional	Can't Use DLB without Anypoint VPC

NEW QUESTION 175
Refer to the exhibit.



A Mule 4 application has a parent flow that breaks up a JSON array payload into 200 separate items, then sends each item one at a time inside an Async scope to a VM queue.
A second flow to process orders has a VM Listener on the same VM queue. The rest of this flow processes each received item by writing the item to a database.
This Mule application is deployed to four CloudHub workers with persistent queues enabled.
What message processing guarantees are provided by the VM queue and the CloudHub workers, and how are VM messages routed among the CloudHub workers for each invocation of the parent flow under normal operating conditions where all the CloudHub workers remain online?

- A. EACH item VM message is processed AT MOST ONCE by ONE CloudHub worker, with workers chosen in a deterministic round-robin fashion Each of the four CloudHub workers can be expected to process 1/4 of the Item VM messages (about 50 items)
- B. EACH item VM message is processed AT LEAST ONCE by ONE ARBITRARY CloudHub worker Each of the four CloudHub workers can be expected to process some item VM messages

- C. ALL Item VM messages are processed AT LEAST ONCE by the SAME CloudHub worker where the parent flow was invoked This one CloudHub worker processes ALL 200 item VM messages
- D. ALL item VM messages are processed AT MOST ONCE by ONE ARBITRARY CloudHub worker This one CloudHub worker processes ALL 200 item VM messages

Answer: B

Explanation:

Correct answer is EACH item VM message is processed AT LEAST ONCE by ONE ARBITRARY CloudHub worker. Each of the four CloudHub workers can be expected to process some item VM messages In Cloudhub, each persistent VM queue is listened on by every CloudHub worker - But each message is read and processed at least once by only one CloudHub worker and the duplicate processing is possible - If the CloudHub worker fails , the message can be read by another worker to prevent loss of messages and this can lead to duplicate processing - By default , every CloudHub worker's VM Listener receives different messages from VM Queue Referenece: <https://dzone.com/articles/deploying-mulesoft-application-on-1-worker-vs-mult>

NEW QUESTION 176

A global, high-volume shopping Mule application is being built and will be deployed to CloudHub. To improve performance, the Mule application uses a Cache scope that maintains cache state in a CloudHub object store. Web clients will access the Mule application over HTTP from all around the world, with peak volume coinciding with business hours in the web client's geographic location. To achieve optimal performance, what Anypoint Platform region should be chosen for the CloudHub object store?

- A. Choose the same region as to where the Mule application is deployed
- B. Choose the US-West region, the only supported region for CloudHub object stores
- C. Choose the geographically closest available region for each web client
- D. Choose a region that is the traffic-weighted geographic center of all web clients

Answer: A

Explanation:

CloudHub object store should be in same region where the Mule application is deployed. This will give optimal performance.

Before learning about Cache scope and object store in Mule 4 we understand what is in general Caching is and other related things.

WHAT DOES "CACHING" MEAN?

Caching is the process of storing frequently used data in memory, file system or database which saves processing time and load if it would have to be accessed from original source location every time.

In computing, a cache is a high-speed data storage layer which stores a subset of data, so that future requests for that data are served up faster than is possible by accessing the data's primary storage location. Caching allows you to efficiently reuse previously retrieved or computed data.

How does Caching work?

The data in a cache is generally stored in fast access hardware such as RAM (Random-access memory) and may also be used in correlation with a software component. A cache's primary purpose is to increase data retrieval performance by reducing the need to access the underlying slower storage layer.

Caching in MULE 4

In Mule 4 caching can be achieved in mule using cache scope and/or object-store. Cache scope internally uses Object Store to store the data.

What is Object Store

Object Store lets applications store data and states across batch processes, Mule components, and applications, from within an application. If used on cloud hub, the object store is shared between applications deployed on Cluster.

Cache Scope is used in below-mentioned cases:

Need to store the whole response from the outbound processor

Data returned from the outbound processor does not change very frequently

As Cache scope internally handle the cache hit and cache miss scenarios it is more readable Object Store is used in below-mentioned cases:

Need to store custom/intermediary data To store watermarks

Sharing the data/stage across applications, schedulers, batch.

If CloudHub object store is in same region where the Mule application is deployed it will aid in fast access of data and give optimal performance.

NEW QUESTION 179

An API implementation is being developed to expose data from a production database via HTTP requests. The API implementation executes a database SELECT statement that is dynamically created based upon data received from each incoming HTTP request. The developers are planning to use various types of testing to make sure the Mule application works as expected, can handle specific workloads, and behaves correctly from an API consumer perspective. What type of testing would typically mock the results from each SELECT statement rather than actually execute it in the production database?

- A. Unit testing (white box)
- B. Integration testing
- C. Functional testing (black box)
- D. Performance testing

Answer: A

Explanation:

In Unit testing instead of using actual backends, stubs are used for the backend services. This ensures that developers are not blocked and have no dependency on other systems.

In Unit testing instead of using actual backends, stubs are used for the backend services. This ensures that developers are not blocked and have no dependency on other systems.

Below are the typical characteristics of unit testing.

-- Unit tests do not require deployment into any special environment, such as a staging environment

-- Unit tests can be run from within an embedded Mule runtime

-- Unit tests can/should be implemented using MUnit

-- For read-only interactions to any dependencies (such as other APIs): allowed to invoke production endpoints

-- For write interactions: developers must implement mocks using MUnit

-- Require knowledge of the implementation details of the API implementation under test

NEW QUESTION 182

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

MCIA-Level-1 Practice Exam Features:

- * MCIA-Level-1 Questions and Answers Updated Frequently
- * MCIA-Level-1 Practice Questions Verified by Expert Senior Certified Staff
- * MCIA-Level-1 Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * MCIA-Level-1 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The MCIA-Level-1 Practice Test Here](#)