

# MuleSoft

## Exam Questions MCIA-Level-1

MuleSoft Certified Integration Architect - Level 1



### NEW QUESTION 1

An organization has implemented the cluster with two customer hosted Mule runtimes is hosting an application.

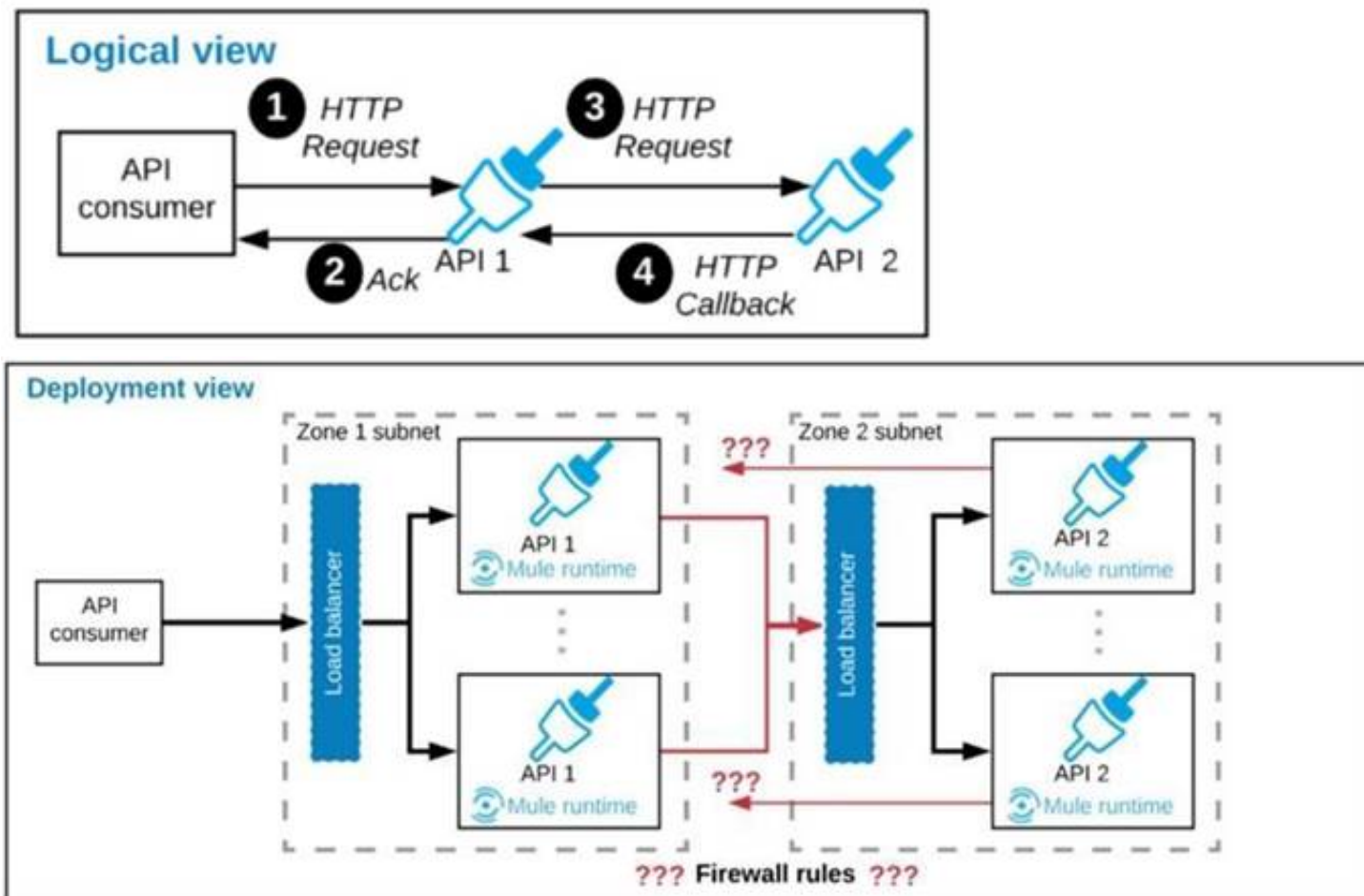
This application has a flow with a JMS listener configured to consume messages from a queue destination. As an integration architect can you advise which JMS listener configuration must be used to receive messages in all the nodes of the cluster?

- A. Use the parameter primaryNodeOnly= "false" on the JMS listener
- B. Use the parameter primaryNodeOnly= "false" on the JMS listener with a shared subscription
- C. Use the parameter primaryNodeOnly= "true" on the JMS listener with a non•shared subscription
- D. Use the parameter primaryNodeOnly= "true" on the JMS listener

**Answer: A**

### NEW QUESTION 2

Refer to the exhibit.



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

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

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

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

**Answer: B**

#### Explanation:

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

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

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

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

### NEW QUESTION 3

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

What is the most performant way of implementing this requirement?

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

**Answer:** A

#### NEW QUESTION 4

A company is planning to migrate its deployment environment from on-premises cluster to a Runtime Fabric (RTF) cluster. It also has a requirement to enable Mule applications deployed to a Mule runtime instance to store and share data across application replicas and restarts. How can these requirements be met?

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

**Answer:** D

#### NEW QUESTION 5

A Mule application is running on a customer-hosted Mule runtime in an organization's network. The Mule application acts as a producer of asynchronous Mule events. Each Mule event must be broadcast to all interested external consumers outside the Mule application. The Mule events should be published in a way that is guaranteed in normal situations and also minimizes duplicate delivery in less frequent failure scenarios.

The organizational firewall is configured to only allow outbound traffic on ports 80 and 443. Some external event consumers are within the organizational network, while others are located outside the firewall.

What Anypoint Platform service is most idiomatic (used for its intended purpose) for publishing these Mule events to all external consumers while addressing the desired reliability goals?

- A. CloudHub VM queues
- B. Anypoint MQ
- C. Anypoint Exchange
- D. CloudHub Shared Load Balancer

**Answer:** B

#### Explanation:

Set the Anypoint MQ connector operation to publish or consume messages, or to accept (ACK) or not accept (NACK) a message.

#### NEW QUESTION 6

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.

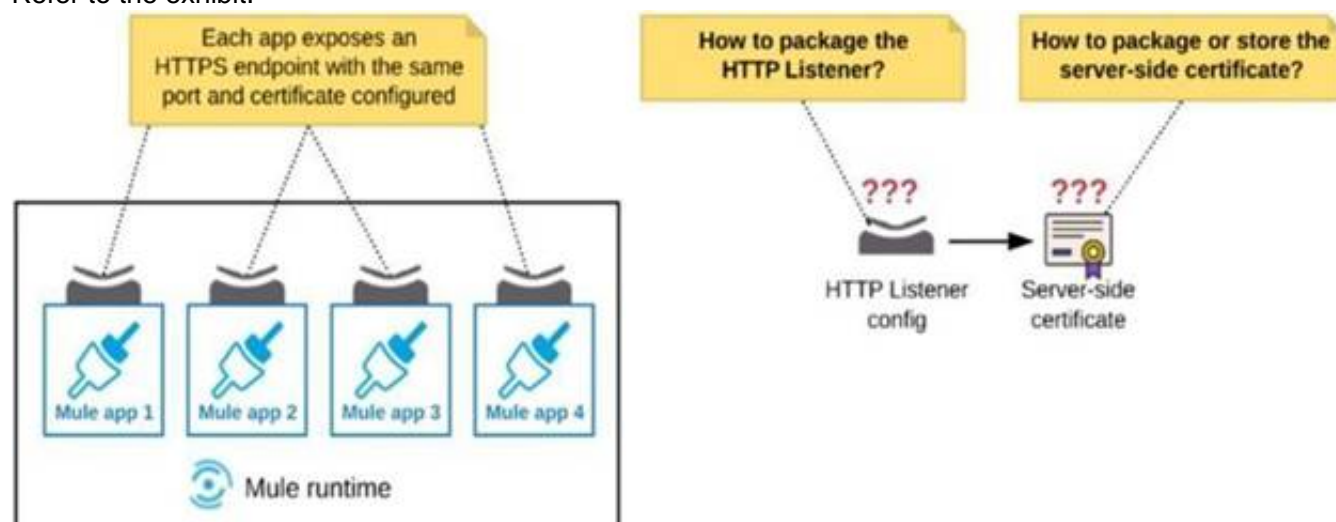
What is the most idiomatic (used for its intended purpose) Anypoint connector, validator, or scope that can be configured in the Mule application to filter duplicate sales orders on the basis of the unique ID field contained in each sales order record?

- 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 7

Refer to the exhibit.



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

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

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

**Answer: B**

**Explanation:**

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

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

What is Mule Domain Project?

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

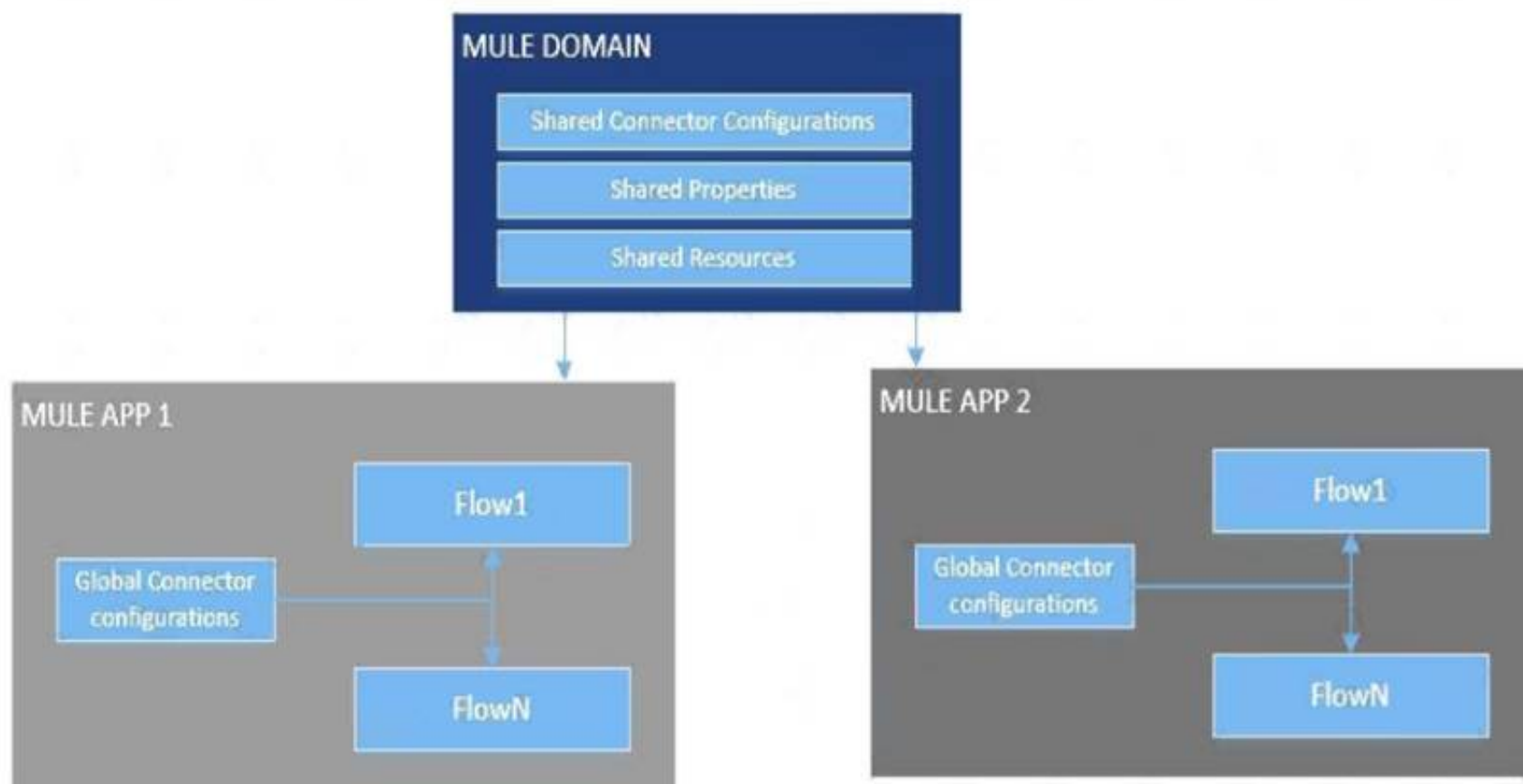
- Expose multiple services within the domain through the same port.
- Share the connection to persistent storage.
- Share services between apps through a well-defined interface.
- Ensure consistency between apps upon any changes because the configuration is only set in one place.

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

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

You can start using domains in only three steps:

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



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

**NEW QUESTION 8**

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 Am/point Exchange
- B. Compile, package, unit test, validate unit test coverage, deploy
- C. Compile, package, unit test, deploy, integration test

**Answer: C**

**NEW QUESTION 9**

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 10**

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

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



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

**Answer:** B

**Explanation:**

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

**NEW QUESTION 10**

An organization is designing Mule application which connects to a legacy backend. It has been reported that backend services are not highly available and experience downtime quite often. As an integration architect which of the below approach you would propose to achieve high reliability goals?

- A. Alerts can be configured in Mule runtime so that backend team can be communicated when services are down
- B. Until Successful scope can be implemented while calling backend API's
- C. On Error Continue scope to be used to call in case of error again
- D. Create a batch job with all requests being sent to backend using that job as per the availability of backend API's

**Answer:** B

**Explanation:**

Correct answer is Untill Successful scope can be implemented while calling backend API's The Until Successful scope repeatedly triggers the scope's components (including flow references) until they all succeed or until a maximum number of retries is exceeded The scope provides option to control the max number of retries and the interval between retries The scope can execute any sequence of processors that may fail for whatever reason and may succeed upon retry

**NEW QUESTION 12**

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

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

**Answer:** D

**NEW QUESTION 13**

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

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

**Answer:** AB

**NEW QUESTION 18**

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 Mute 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 Mute 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 22**

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 24

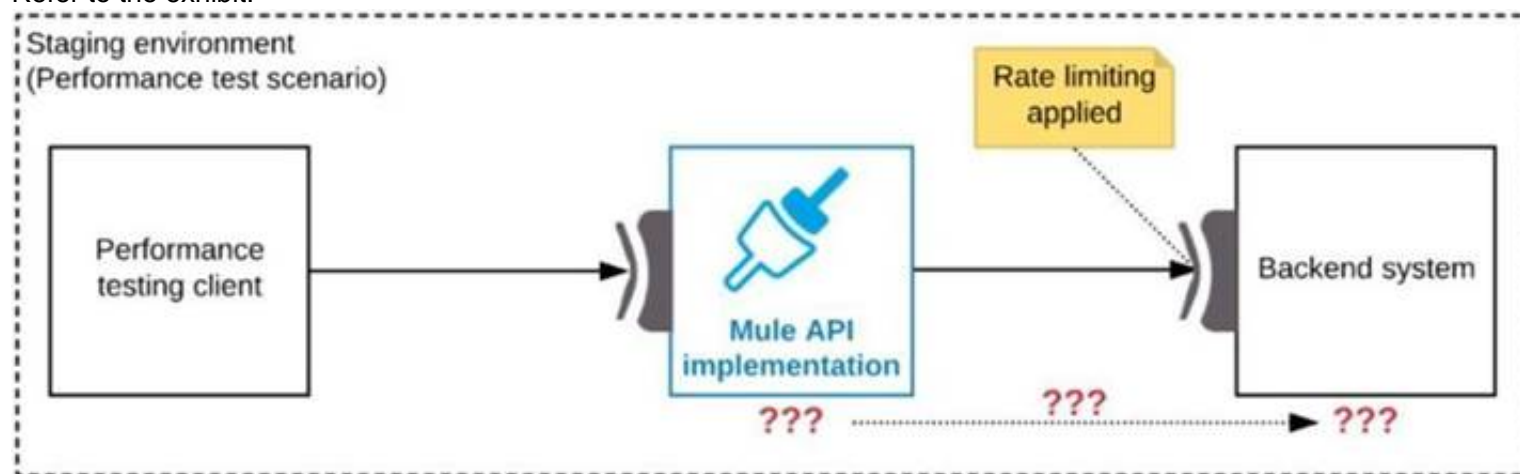
A company is implementing a new Mule application that supports a set of critical functions driven by a rest API enabled, claims payment rules engine hosted on oracle ERP. As designed the mule application requires many data transformation operations as it performs its batch processing logic. The company 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 29

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.

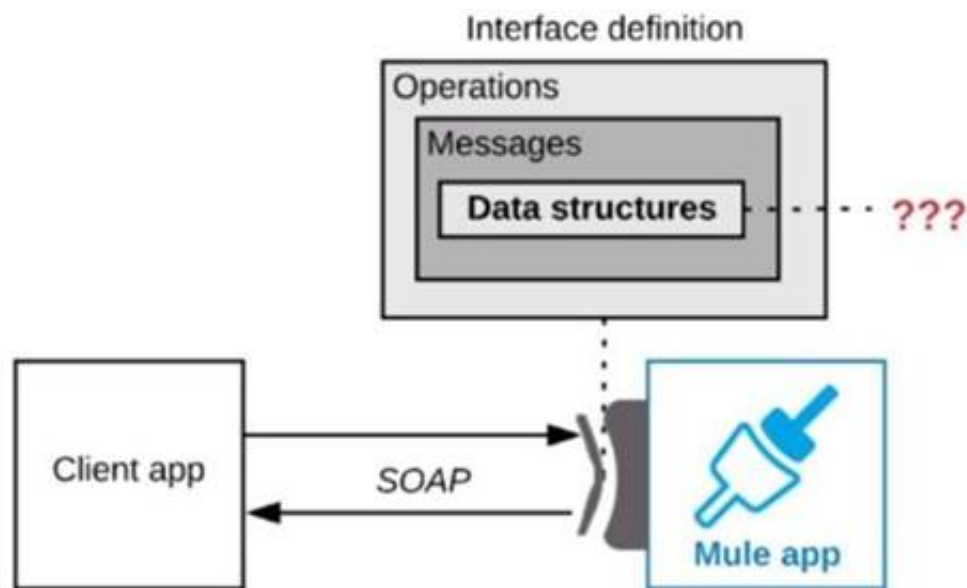
\* Scaled down performance tests cant be relied upon as performance of API's is not linear against load.

#### NEW QUESTION 34

Refer to the exhibit.

A Mule application is being designed to expose a SOAP web service to its clients.

What language is typically used inside the web service's interface definition to define the data structures that the web service is expected to exchange with its clients?



- A. WSDL
- B. XSD
- C. JSON Schema
- D. RAML

**Answer: B**

**Explanation:**

Correct Answer XSD In this approach to developing a web service, you begin with an XML schema (XSD file) that defines XML data structures to be used as parameters and return types in the web service operations.

----- Reference:  
[https://www.w3schools.com/xml/schema\\_intro.asp](https://www.w3schools.com/xml/schema_intro.asp)

**NEW QUESTION 36**

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

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

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

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

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

**Answer: B**

**NEW QUESTION 39**

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 44**

An organization has several APIs that accept JSON data over HTTP POST. The APIs are all publicly available and are associated with several mobile applications and web applications. The organization does NOT want to use any authentication or compliance policies for these APIs, but at the same time, is worried that some bad actor could send payloads that could somehow compromise the applications or servers running the API implementations. What out-of-the-box Anypoint Platform policy can address exposure to this threat?

- A. Apply a Header injection and removal policy that detects the malicious data before it is used
- B. Apply an IP blacklist policy to all APIs; the blacklist will Include all bad actors
- C. Shut out bad actors by using HTTPS mutual authentication for all API invocations
- D. Apply a JSON threat protection policy to all APIs to detect potential threat vectors

**Answer: D**

**Explanation:**

We need to note few things about the scenario which will help us in reaching the correct solution.

Point 1 : The APIs are all publicly available and are associated with several mobile applications and web applications. This means Apply an IP blacklist policy is not viable option. as blacklisting IPs is limited to partial web traffic. It can't be useful for traffic from mobile application

Point 2 : The organization does NOT want to use any authentication or compliance policies for these APIs. This means we can not apply HTTPS mutual authentication scheme.

Header injection or removal will not help the purpose.

By its nature, JSON is vulnerable to JavaScript injection. When you parse the JSON object, the malicious code inflicts its damages. An inordinate increase in the size and depth of the JSON payload can indicate injection. Applying the JSON threat protection policy can limit the size of your JSON payload and thwart recursive



additions to the JSON hierarchy.

Hence correct answer is Apply a JSON threat protection policy to all APIs to detect potential threat vectors

#### NEW QUESTION 48

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.

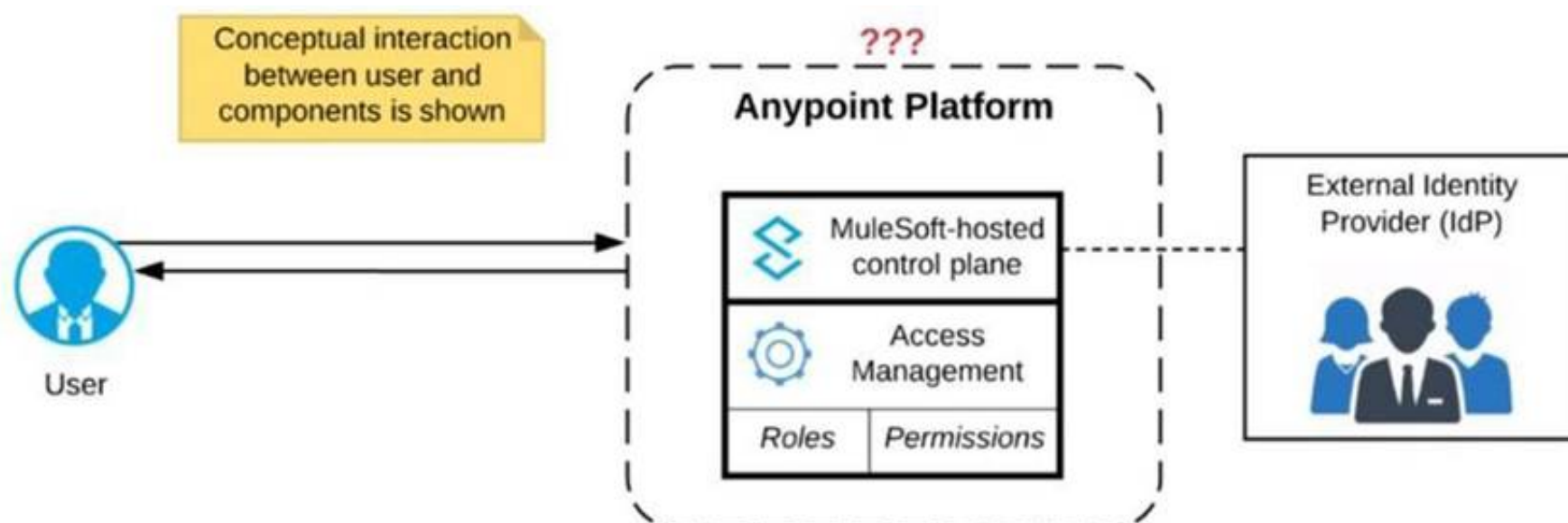
When there are no errors, how does the Batch Job scope process records within and between the Batch Step scopes?

- 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 53

Refer to the exhibit.



Anypoint Platform supports role-based access control (RBAC) to features of the platform. An organization has configured an external Identity Provider for identity management with Anypoint Platform.

What aspects of RBAC must ALWAYS be controlled from the Anypoint Platform control plane and CANNOT be controlled via the external Identity Provider?

- A. Controlling the business group within Anypoint Platform to which the user belongs
- B. Assigning Anypoint Platform permissions to a role
- C. Assigning Anypoint Platform role(s) to a user
- D. Removing a user's access to Anypoint Platform when they no longer work for the organization

**Answer: B**

#### Explanation:

\* By default, Anypoint Platform performs its own user management

– For user management, one external IdP can be integrated with the Anypoint Platform organization (note: not at business group level)

– Permissions and access control are still enforced inside Anypoint Platform and CANNOT be controlled via the external Identity Provider \* As the Anypoint Platform organization administrator, you can configure identity management in Anypoint Platform to set up users for single sign-on (SSO). \* You can map users in a federated organization's group to a role which also gives the flexibility of controlling the business group within Anypoint Platform to which the user belongs to. Also user can be removed from external identity management system when they no longer work for the organization. So they won't be able to authenticate using SSO to login to Anypoint Platform. \* Using external identity we can not change permissions of a particular role in Mulesoft Anypoint platform.

\* So Correct answer is Assigning Anypoint Platform permissions to a role

#### NEW QUESTION 54

An integration Mule application consumes and processes a list of rows from a CSV file. Each row must be read from the CSV file, validated, and the row data sent to a JMS queue, in the exact order as in the CSV file.

If any processing step for a row fails, then a log entry must be written for that row, but processing of other rows must not be affected.

What combination of Mule components is most idiomatic (used according to their intended purpose) when implementing the above requirements?

- A. Scatter-Gather component On Error Continue scope
- B. VM connector first Successful scope On Error Propagate scope
- C. For Each scope On Error Continue scope
- D. Async scope On Error Propagate scope

**Answer: C**

#### Explanation:



- \* On Error Propagate halts execution and sends error to the client. In this scenario it's mentioned that "processing of other rows must not be affected" so Option B and C are ruled out.
- \* Scatter gather is used to club multiple responses together before processing. In this scenario, we need sequential processing. So option A is out of choice.
- \* Correct answer is For Each scope & On Error Continue scope Below requirement can be fulfilled in the below way
- 1) Using For Each scope , which will send each row from csv file sequentially. each row needs to be sent sequentially as requirement is to send the message in exactly the same way as it is mentioned in the csv file
- 2) Also other part of requirement is if any processing step for a row fails then it should log an error but should not affect other record processing . This can be achieved using On error Continue scope on these set of activities. so that error will not halt the processing. Also logger needs to be added in error handling section so that it can be logged.
- \* Attaching diagram for reference. Here it's try scope, but similar would be the case with For Each loop. Diagram Description automatically generated



#### NEW QUESTION 57

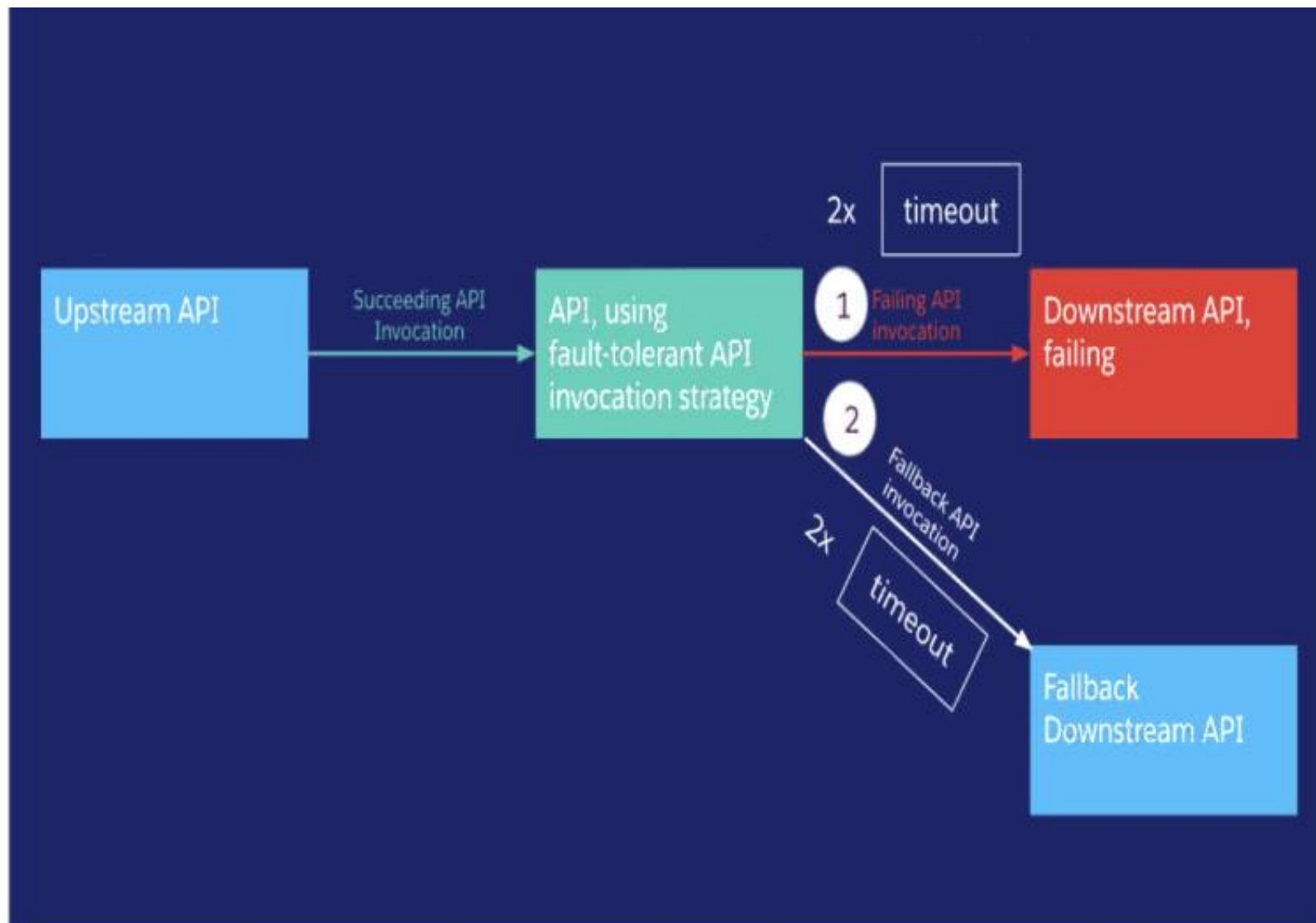
An API implementation is being designed that must invoke an Order API which is known to repeatedly experience downtime. For this reason a fallback API is to be called when the Order API is unavailable. What approach to designing invocation of the fallback API provides the best resilience?

- Redirect client requests through an HTTP 303 temporary redirect status code to the fallback API whenever the Order API is unavailable
- 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
- 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
- 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 62

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 S 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 64

An auto mobile company want to share inventory updates with dealers D1 and D2 asynchronously and concurrently via queues Q1 and Q2. Dealer D1 must consume the message from the queue Q1 and dealer D2 to must consume a message from the queue Q2. Dealer D1 has implemented a retry mechanism to reprocess the transaction in case of any errors while processing the inventers updates. Dealer D2 has not implemented any retry mechanism. How should the dealers acknowledge the message to avoid message loss and minimize impact on the current implementation?

- A. Dealer D1 must use auto acknowledgement and dealer D2 can use manual acknowledgement and acknowledge the message after successful processing
- B. Dealer D1 can use auto acknowledgement and dealer D2 can use IMMEDIATE acknowledgement and acknowledge the message of successful processing
- C. Dealer D1 and dealer D2 must use AUTO acknowledgement and acknowledge the message after successful processing
- D. Dealer D1 can use AUTO acknowledgement and dealer D2 must use manual acknowledgement and acknowledge the message after successful processing

**Answer: D**

#### NEW QUESTION 67

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 69

An organization is evaluating using the CloudHub shared Load Balancer (SLB) vs creating a CloudHub dedicated load balancer (DLB). They are evaluating how this choice affects the various types of certificates used by CloudHub deployed Mule applications, including MuleSoft-provided, customer-provided, or Mule application-provided certificates.

What type of restrictions exist on the types of certificates that can be exposed by the CloudHub Shared Load Balancer (SLB) to external web clients over the public internet?

- A. Only MuleSoft-provided certificates are exposed.
- B. Only customer-provided wildcard certificates are exposed.
- C. Only customer-provided self-signed certificates are exposed.
- D. Only underlying Mule application certificates are exposed (pass-through)

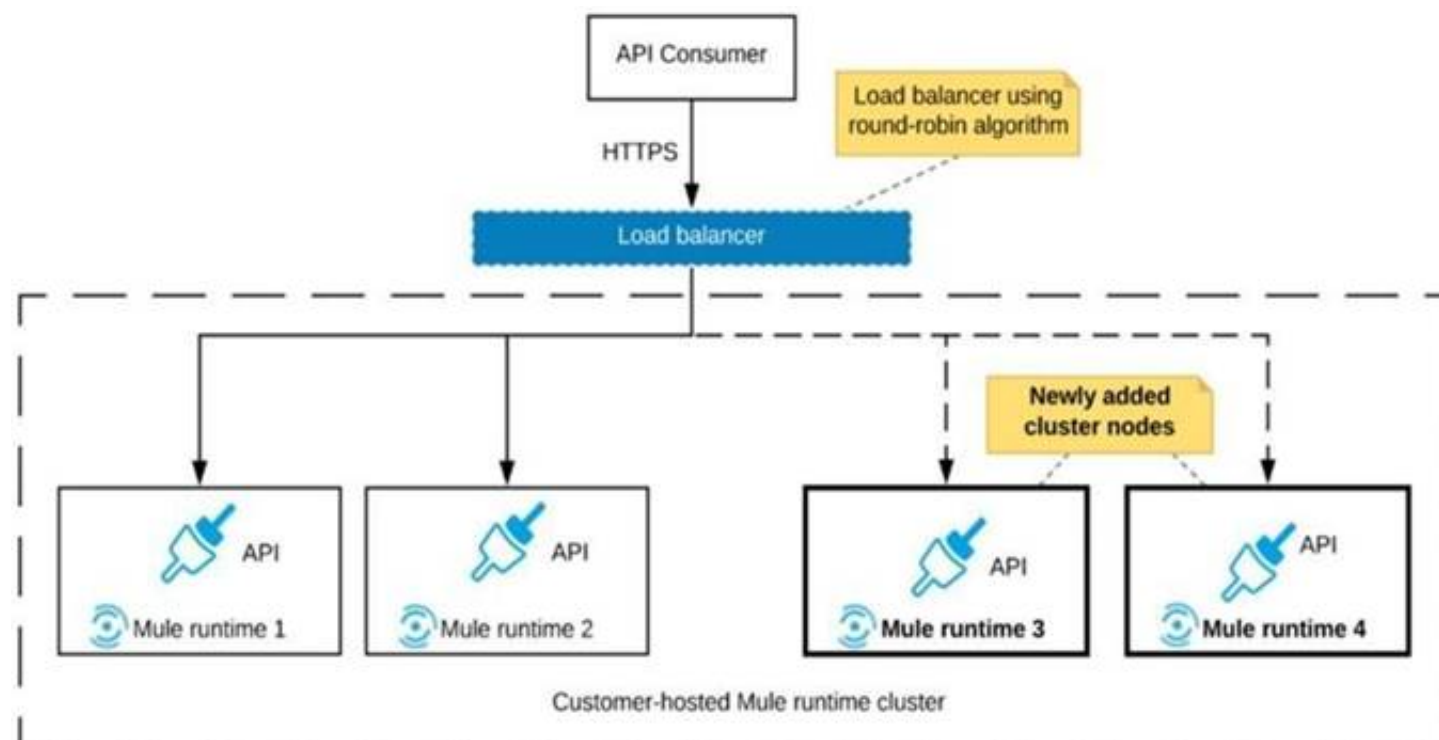
**Answer: A**

#### Explanation:

<https://docs.mulesoft.com/runtime-manager/dedicated-load-balancer-tutorial>

#### NEW QUESTION 72

Refer to the exhibit.



An organization uses a 2-node Mule runtime cluster to host one stateless API implementation. The API is accessed over HTTPS through a load balancer that uses round-robin for load distribution.

Two additional nodes have been added to the cluster and the load balancer has been configured to recognize the new nodes with no other change to the load balancer.

What average performance change is guaranteed to happen, assuming all cluster nodes are fully operational?

- A. 50% reduction in the response time of the API
- B. 100% increase in the throughput of the API
- C. 50% reduction in the JVM heap memory consumed by each node
- D. 50% reduction in the number of requests being received by each node

**Answer: D**

#### NEW QUESTION 77



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

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

**Answer: D**

**Explanation:**

Correct answer is Anypoint Runtime Manager

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

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

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

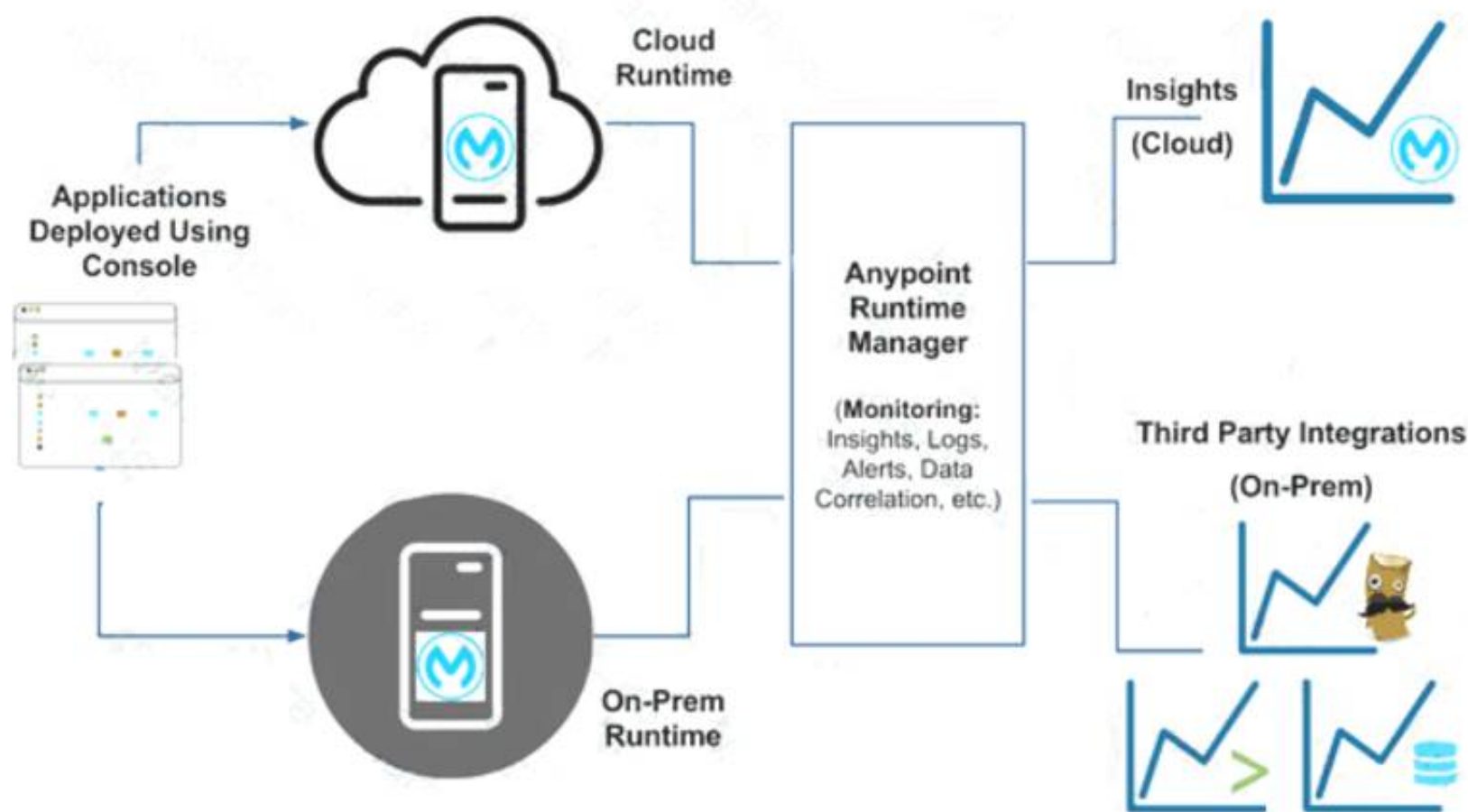
--> Send Mule event notifications, including flow executions and exceptions, to Splunk or ELK.

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

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

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

Diagram Description automatically generated

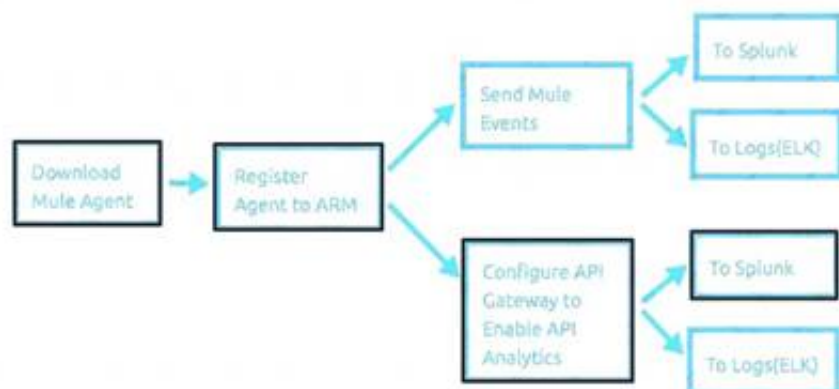


Additional Info:

It can be achieved in 3 steps:

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

Diagram Description automatically generated



**NEW QUESTION 79**

As a part of business requirement , old CRM system needs to be integrated using Mule application. CRM system is capable of exchanging data only via SOAP/HTTP protocol. As an integration architect who follows API led approach , what is the the below step you will perform so that you can share document with CRM team?

- A. Create RAML specification using Design Center
- B. Create SOAP API specification using Design Center
- C. Create WSDL specification using text editor
- D. Create WSDL specification using Design Center



**Answer:** C

**Explanation:**

Correct answer is Create WSDL specification using text editor SOAP services are specified using WSDL. A client program connecting to a web service can read the WSDL to determine what functions are available on the server. We can not create WSDL specification in Design Center. We need to use external text editor to create WSDL.

**NEW QUESTION 84**

An organization uses a four(4) node customer hosted Mule runtime cluster to host one(1) stateless api implementation. The API is accessed over HTTPS through a load balancer that uses round-robin for load distribution. Each node in the cluster has been sized to be able to accept four(4) times the current number of requests.

Two(2) nodes in the cluster experience a power outage and are no longer available. The load balancer directs the outage and blocks the two unavailable the nodes from receiving further HTTP requests.

What performance-related consequence is guaranteed to happen to average, assuming the remaining cluster nodes are fully operational?

- A. 100% increase in the average response time of the API
- B. 50% reduction in the throughput of the API
- C. 100% increase in the number of requests received by each remaining node
- D. 50% increase in the JVM heap memory consumed by each remaining node

**Answer:** C

**Explanation:**

\* "100% increase in the throughput of the API" might look correct, as the number of requests processed per second might increase, but is it guaranteed to increase by 100%? Using 4 nodes will definitely increase throughput of system. But it is cant be precisely said if there would be 100% increase in throughput as it depends on many other factors. Also it is nowhere mentioned in the description that all nodes have same CPU/memory assigned. The question is about the guaranteed behavior \* Increasing number of nodes will have no impact on response time as we are scaling application horizontally and not vertically. Similarly there is no change in JVM heap memory usage. \* So Correct answer is 50% reduction in the number of requests being received by each node This is because of the two reasons. 1) API is mentioned as stateless 2) Load Balancer is used

**NEW QUESTION 86**

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 91**

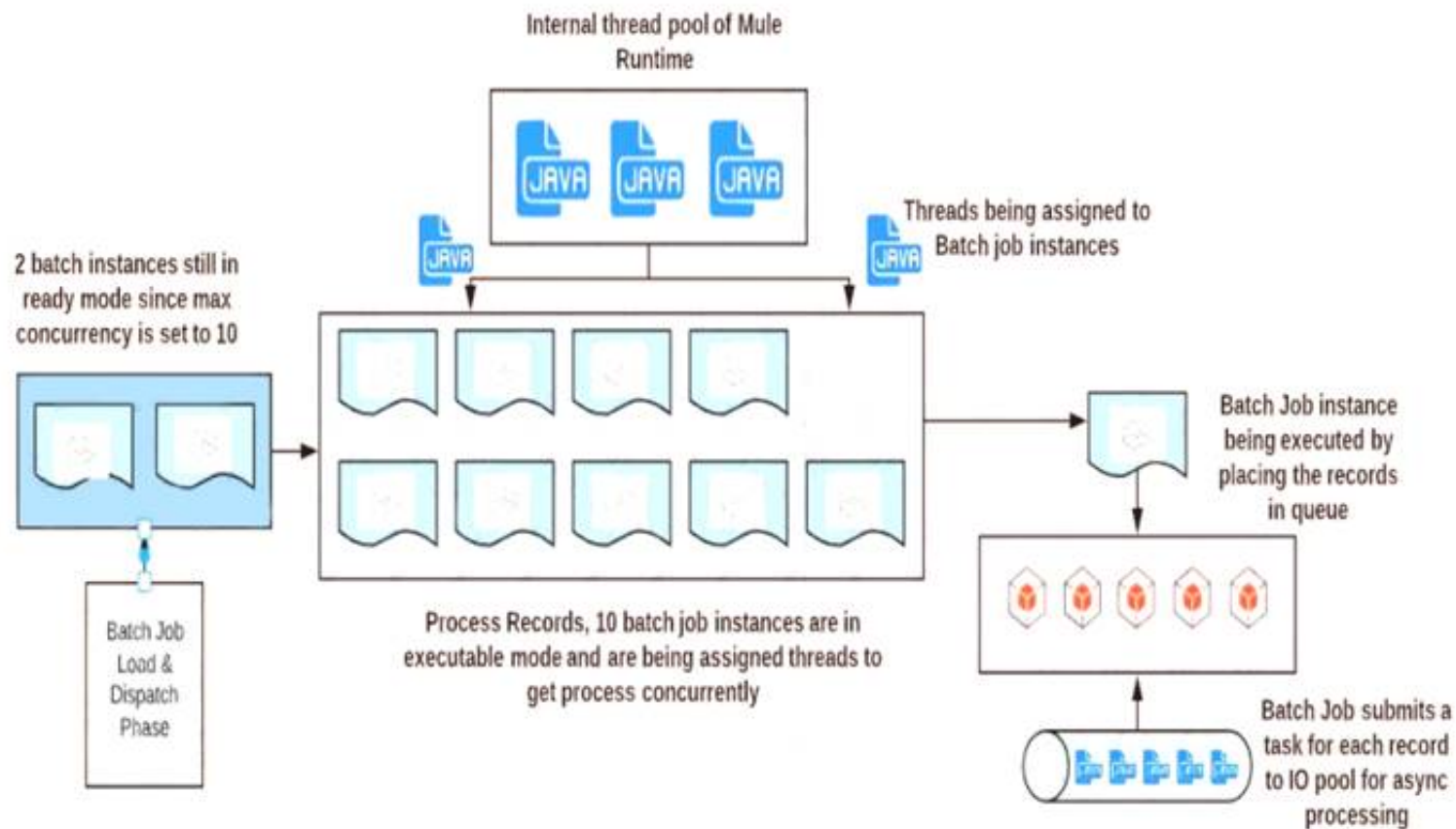
A Mule application contains a Batch Job with two Batch Steps (Batch\_Step\_1 and Batch\_Step\_2). A payload with 1000 records is received by the Batch Job. How many threads are used by the Batch Job to process records, and how does each Batch Step process records within the Batch Job?

- A. Each Batch Job uses SEVERAL THREADS for the Batch Steps Each Batch Step instance receives ONE record at a time as the payload, and RECORDS are processed IN PARALLEL within and between the two Batch Steps
- B. Each Batch Job uses a SINGLE THREAD for all Batch steps Each Batch step instance receives ONE record at a time as the payload, and RECORDS are processed IN ORDER, first through Batch\_Step\_1 and then through Batch\_Step\_2
- C. Each Batch Job uses a SINGLE THREAD to process a configured block size of record Each Batch Step instance receives A BLOCK OF records as the payload, and BLOCKS of records are processed IN ORDER
- D. Each Batch Job uses SEVERAL THREADS for the Batch Steps Each Batch Step instance receives ONE record at a time as the payload, and BATCH STEP INSTANCES execute IN PARALLEL to process records and Batch Steps in ANY order as fast as possible

**Answer:** A

**Explanation:**

\* Each Batch Job uses SEVERAL THREADS for the Batch Steps  
\* Each Batch Step instance receives ONE record at a time as the payload. It's not received in a block, as it does not wait for multiple records to be completed before moving to next batch step. (So Option D is out of choice)  
\* RECORDS are processed IN PARALLEL within and between the two Batch Steps.  
\* RECORDS are not processed in order. Let's say if second record completes batch\_step\_1 before record 1, then it moves to batch\_step\_2 before record 1. (So option C and D are out of choice)  
\* A batch job is the scope element in an application in which Mule processes a message payload as a batch of records. The term batch job is inclusive of all three phases of processing: Load and Dispatch, Process, and On Complete.  
\* A batch job instance is an occurrence in a Mule application whenever a Mule flow executes a batch job. Mule creates the batch job instance in the Load and Dispatch phase. Every batch job instance is identified internally using a unique String known as batch job instance id.



#### NEW QUESTION 95

A manufacturing company is planning to deploy Mule applications to its own Azure Kubernetes Service infrastructure. The organization wants to make the Mule applications more available and robust by deploying each Mule application to an isolated Mule runtime in a Docker container while managing all the Mule applications from the MuleSoft-hosted control plane. What is the most idiomatic (used for its intended purpose) choice of runtime plane to meet these organizational requirements?

- A. Anypoint Platform Private Cloud Edition
- B. Anypoint Runtime Fabric
- C. CloudHub
- D. Anypoint Service Mesh

**Answer: B**

#### NEW QUESTION 99

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 its intended purpose) router or scope to used in the parent flow, and what type of error handler should be used in the child flow?

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

**Answer: B**

#### Explanation:

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

#### NEW QUESTION 102

A mule application must periodically process a large dataset which varies from 6 GB to 8 GB from a back-end database and write transform data to an FTPS server using a properly configured batch job scope. The performance requirements of an application are approved to run in the cloud hub 0.2 vCore with 8 GB storage capacity and currency requirements are met. How can the high rate of records be effectively managed in this application?

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

**Answer: A**

#### NEW QUESTION 105

To implement predictive maintenance on its machinery equipment, ACME Tractors has installed thousands of IoT sensors that will send data for each machinery asset as sequences of JMS messages, in near real-time, to a JMS queue named SENSOR\_DATA on a JMS server. The Mule application contains a JMS Listener

operation configured to receive incoming messages from the JMS servers SENSOR\_DATA JMS queue. The Mule application persists each received JMS message, then sends a transformed version of the corresponding Mule event to the machinery equipment back-end systems. The Mule application will be deployed to a multi-node, customer-hosted Mule runtime cluster. Under normal conditions, each JMS message should be processed exactly once.

How should the JMS Listener be configured to maximize performance and concurrent message processing of the JMS queue?

- A. Set numberOfConsumers = 1 Set primaryNodeOnly = false
- B. Set numberOfConsumers = 1 Set primaryNodeOnly = true
- C. Set numberOfConsumers to a value greater than one Set primaryNodeOnly = true
- D. Set numberOfConsumers to a value greater than one Set primaryNodeOnly = false

**Answer: D**

#### NEW QUESTION 108

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

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

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

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

**Answer: C**

#### NEW QUESTION 112

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 responsesThen 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 characteristicsThen 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 syste
- E. Then upscale performance results to full production scale

**Answer: C**

#### NEW QUESTION 116

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 columnIn the same now, 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 applicationWrite 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 receivedmessage, 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 browsersIn 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 columnIn the same flow, use a Batch Job scope to publish transformed Inventory-pricing records to an Anypoint MQ queueWrite 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 120

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 125**

As a part of project , existing java implementation is being migrated to Mulesoft. Business is very tight on the budget and wish to complete the project in most economical way possible.

Canonical object model using java is already a part of existing implementation. Same object model is required by mule application for a business use case. What is the best way to achieve this?

- A. Make use of Java module
- B. Create similar model for Mule applications
- C. Create a custom application to read Java code and make it available for Mule application
- D. Use Anypoint exchange

**Answer:** A

**Explanation:**

Mule 4 is built to:

- Minimize the need for custom code.
  - Avoid the need for you to know or understand Java.
- However, some advanced uses cases require integration with custom Java code, such as:
- Reuse of a library, such as a tax calculation library.
  - Reuse of a canonical object model that is standard in the organization.
  - Execution of custom logic using Java.

Mule ref doc : <https://docs.mulesoft.com/java-module/1.2/>

**NEW QUESTION 126**

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.

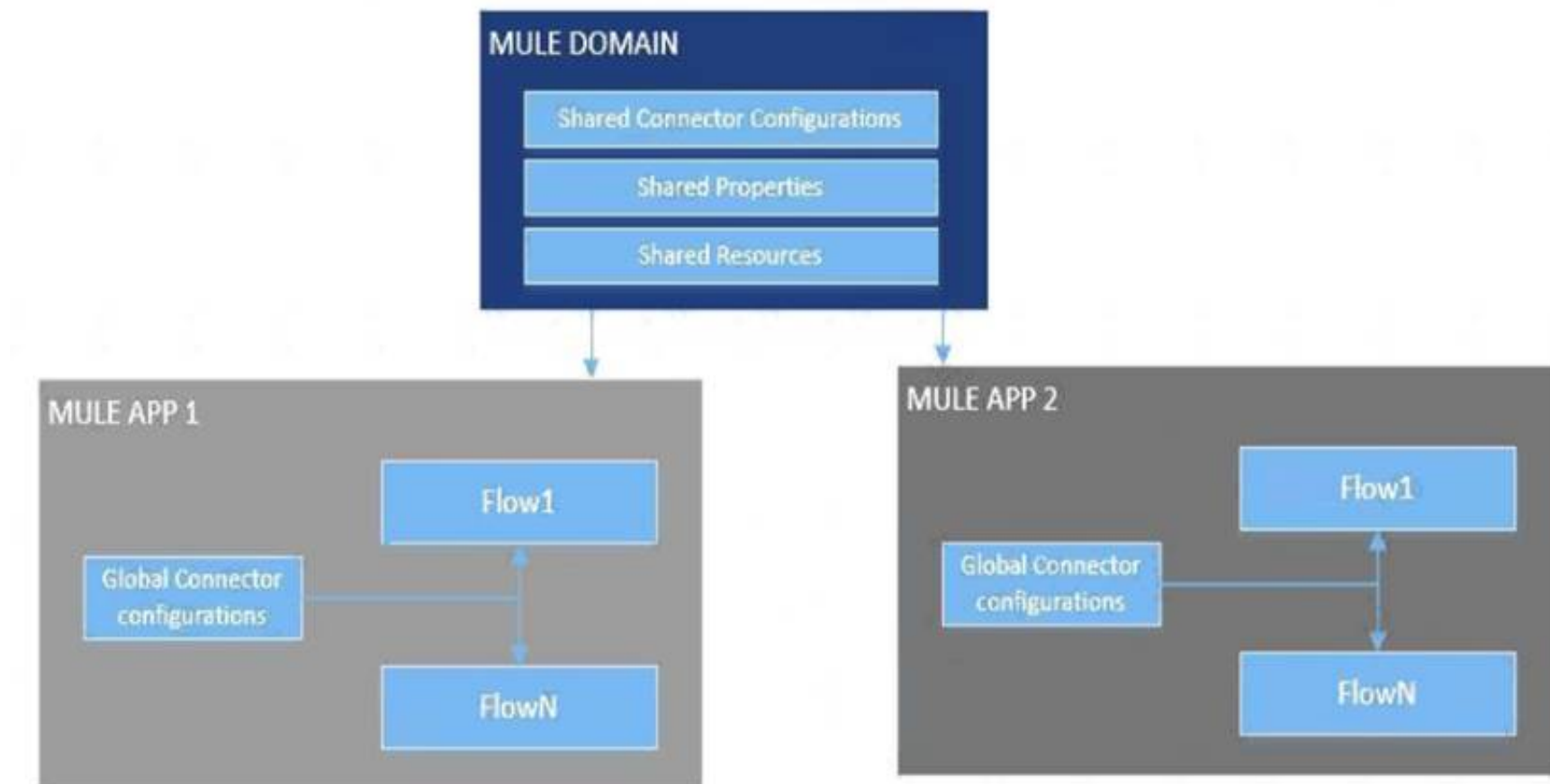
\* Use domains Project to share the same host and port among multiple projects. You can declare the http connector within a domain project and associate the domain project with other projects. Doing this also allows to control thread settings, keystore configurations, time outs for all the requests made 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 the 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 129

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

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

**Answer: C**

#### NEW QUESTION 132

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

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

**Answer: C**

#### Explanation:

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

#### NEW QUESTION 135

.....

## 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](#)**