MCPA-LEVEL-1-MAINTENANCE Q&As

MuleSoft Certified Platform Architect - Level 1 MAINTENANCE

Pass Mulesoft MCPA-LEVEL-1-MAINTENANCE Exam with 100% Guarantee

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

https://www.pass4itsure.com/mcpa-level-1-maintenance.html

100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

An API client calls one method from an existing API implementation. The API implementation is later updated. What change to the API implementation would require the API client\\'s invocation logic to also be updated?

A. When the data type of the response is changed for the method called by the API client

B. When a new method is added to the resource used by the API client

C. When a new required field is added to the method called by the API client

D. When a child method is added to the method called by the API client

Correct Answer: C

When a new required field is added to the method called by the API client **********

>> Generally, the logic on API clients need to be updated when the API contract breaks. >> When a new method or a child method is added to an API , the API client does not break as it can still continue to use its existing method. So these

two options are out. >> We are left for two more where "datatype of the response if changed" and "a new required field is added".

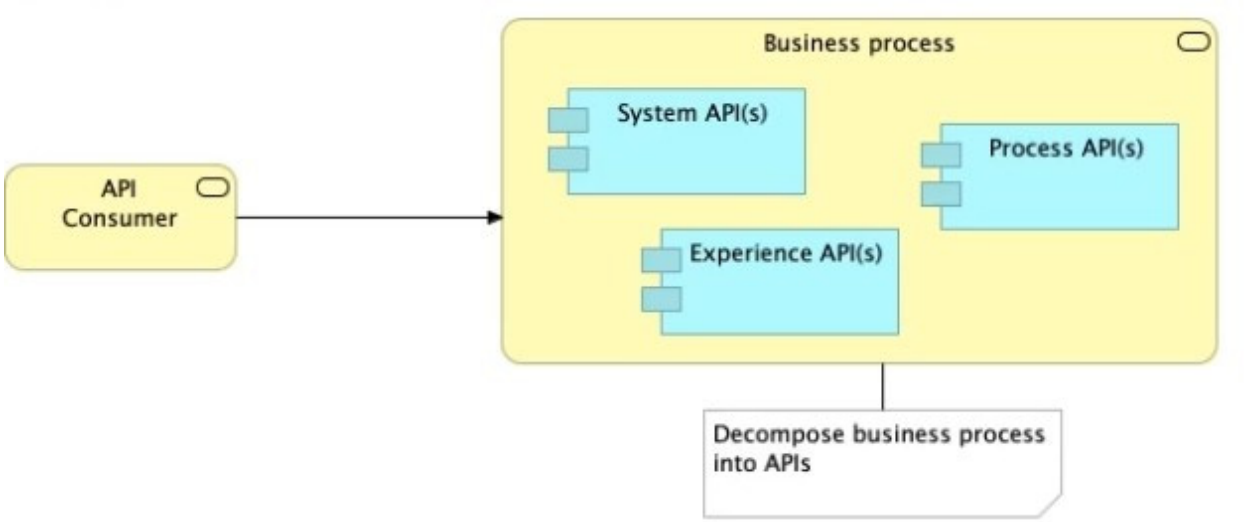
>> Changing the datatype of the response does break the API contract. However, the question is insisting on the "invocation" logic and not about the response handling logic. The API client can still invoke the API successfully and receive the

response but the response will have a different datatype for some field. >> Adding a new required field will break the API\\'s invocation contract. When adding a new required field, the API contract breaks the RAML or API spec agreement that

the API client/API consumer and API provider has between them. So this requires the API client invocation logic to also be updated.

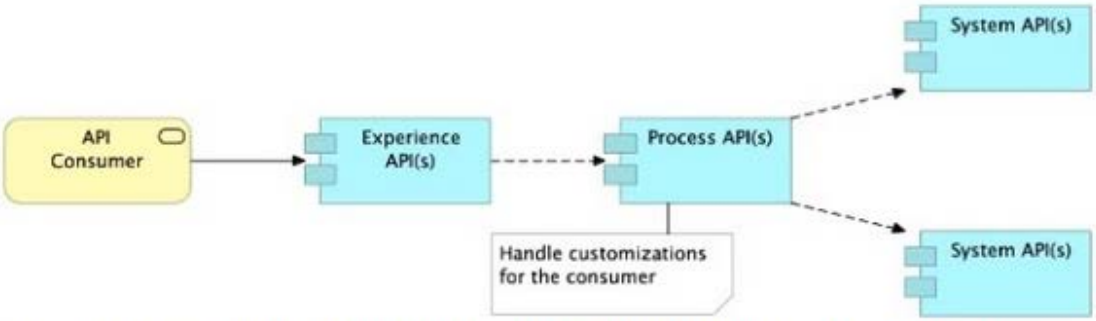
QUESTION 2

Refer to the exhibit.

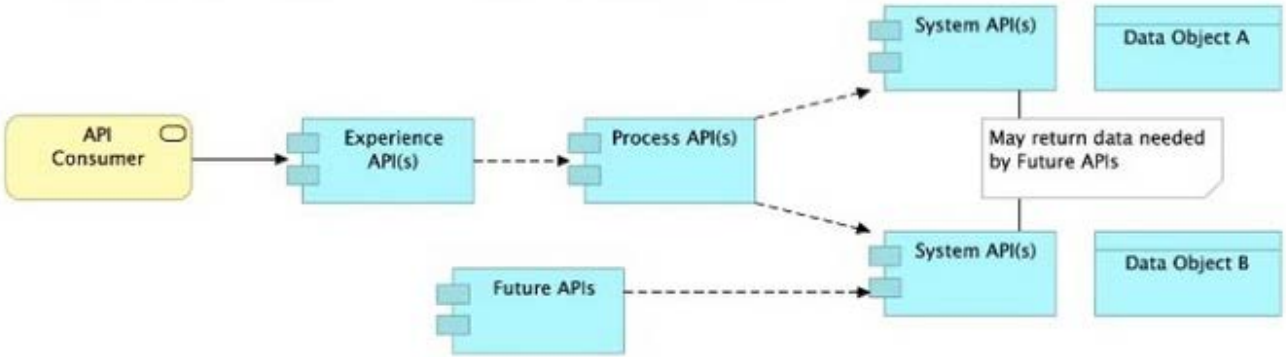


What is the best way to decompose one end-to-end business process into a collaboration of Experience, Process, and System APIs?

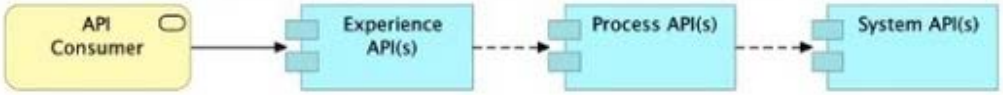
A. Handle customizations for the end-user application at the Process API level rather than the Experience API level



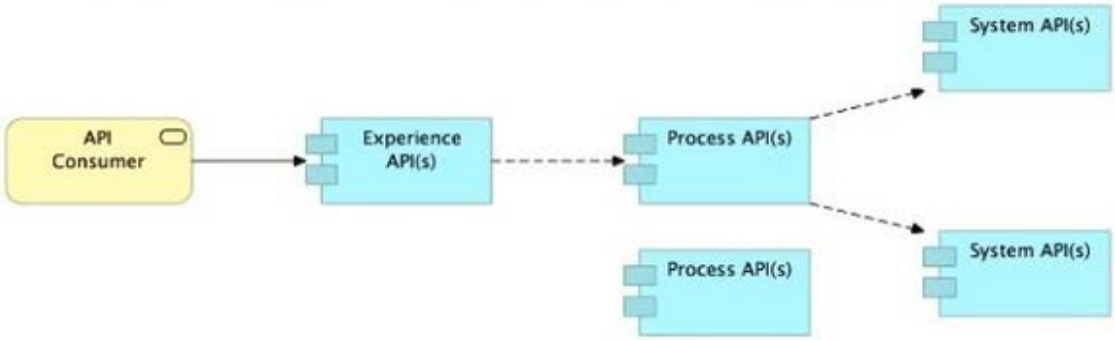
B. Allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs



C. Always use a tiered approach by creating exactly one API for each of the 3 layers (Experience, Process and System APIs)



D. Use a Process API to orchestrate calls to multiple System APIs, but NOT to other Process APIs



A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

Allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs.

****

>> All customizations for the end-user application should be handled in "Experience API" only. Not in Process API

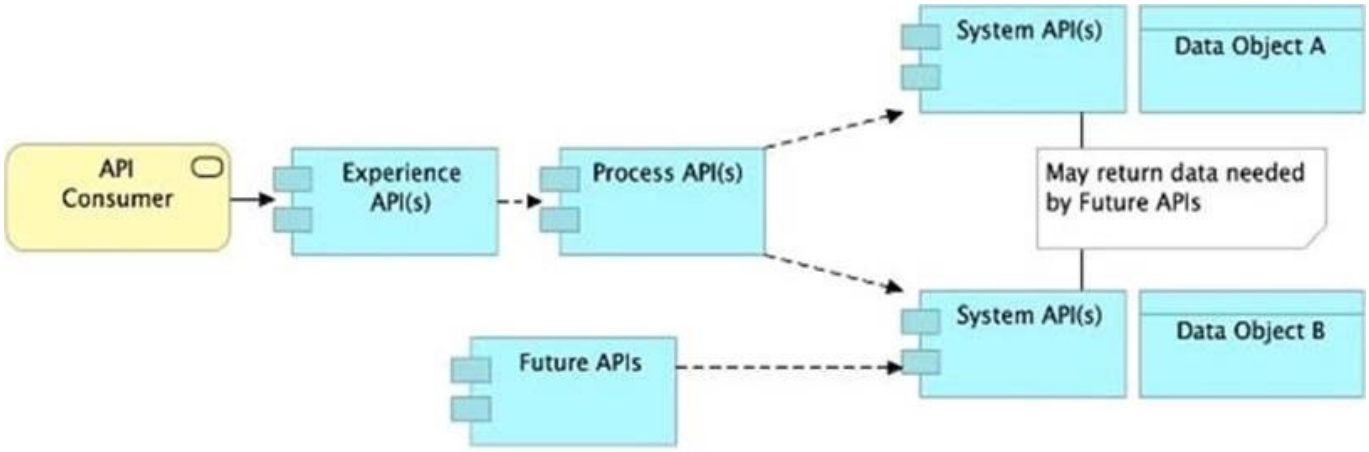
>> We should use tiered approach but NOT always by creating exactly one API for each of the 3 layers. Experience APIs might be one but Process APIs and System APIs are often more than one. System APIs for sure will be more than one

all the time as they are the smallest modular APIs built in front of end systems. >> Process APIs can call System APIs as well as other Process APIs. There is no such anti-design pattern in API-Led connectivity saying Process APIs should

not call other Process APIs.

So, the right answer in the given set of options that makes sense as per API-Led connectivity principles is to allow System APIs to return data that is NOT currently required by the identified Process or Experience APIs. This way, some future

Process APIs can make use of that data from System APIs and we need NOT touch the System layer APIs again and again.



QUESTION 3

What is a key requirement when using an external Identity Provider for Client Management in Anypoint Platform?

A. Single sign-on is required to sign in to Anypoint Platform

B. The application network must include System APIs that interact with the Identity Provider

C. To invoke OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider

D. APIs managed by Anypoint Platform must be protected by SAML 2.0 policies

Correct Answer: C

https://www.folkstalk.com/2019/11/mulesoft-integration-and-platform.html To invoke OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider

****

>> It is NOT necessary that single sign-on is required to sign in to Anypoint Platform because we are using an external Identity Provider for Client Management >> It is NOT necessary that all APIs managed by Anypoint Platform must be

protected by SAML 2.0 policies because we are using an external Identity Provider for Client Management

>> Not TRUE that the application network must include System APIs that interact with the Identity Provider because we are using an external Identity Provider for Client Management Only TRUE statement in the given options is - "To invoke

OAuth 2.0-protected APIs managed by Anypoint Platform, API clients must submit access tokens issued by that same Identity Provider"

References:

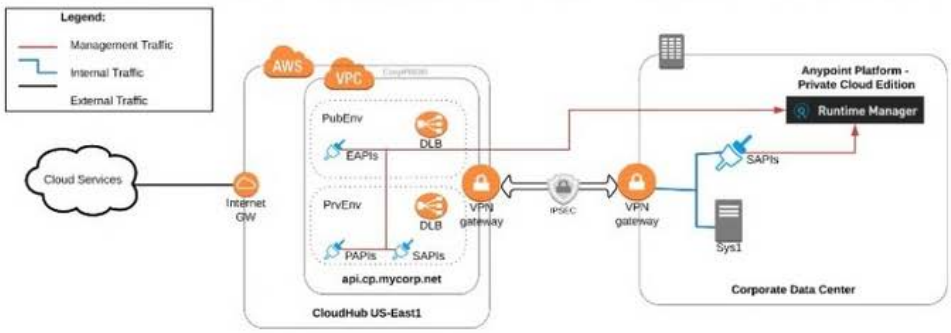
https://docs.mulesoft.com/api-manager/2.x/external-oauth-2.0-token-validation-policy https://blogs.mulesoft.com/dev/api-dev/api-security-ways-to-authenticate-and-authorize/

QUESTION 4

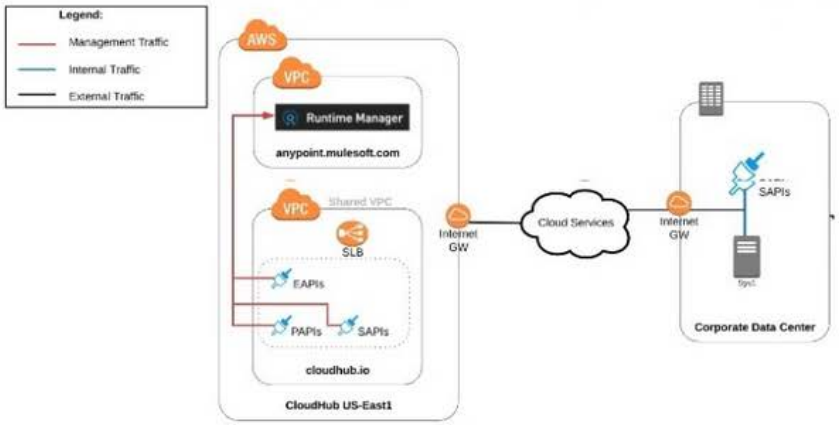
An organization uses various cloud-based SaaS systems and multiple on-premises systems. The on-premises systems are an important part of the organization\\'s application network and can only be accessed from within the organization\\'s intranet.

What is the best way to configure and use Anypoint Platform to support integrations with both the cloud-based SaaS systems and on-premises systems?

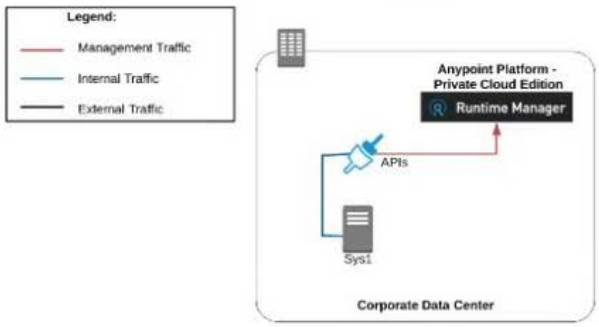
A. Use CloudHub-deployed Mule runtimes in an Anypoint VPC managed by Anypoint Platform Private Cloud Edition control plane



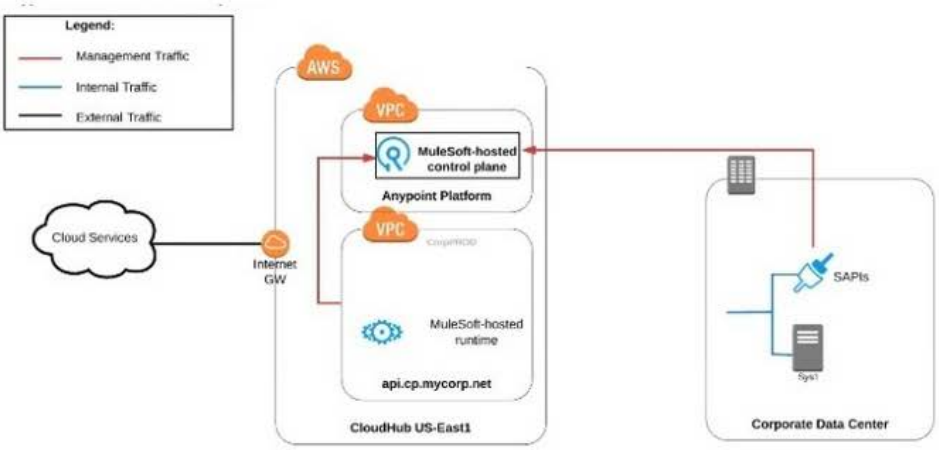
B. Use CloudHub-deployed Mule runtimes in the shared worker cloud managed by the MuleSoft-hosted Anypoint Platform control plane



C. Use an on-premises installation of Mule runtimes that are completely isolated with NO external network access, managed by the Anypoint Platform Private Cloud Edition control plane



D. Use a combination of Cloud Hub-deployed and manually provisioned on-premises Mule runtimes managed by the MuleSoft-hosted Anypoint Platform control plane



A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

Use a combination of CloudHub-deployed and manually provisioned on- premises Mule runtimes managed by the MuleSoft-hosted Platform control plane.

****

Key details to be taken from the given scenario:

>> Organization uses BOTH cloud-based and on-premises systems >> On-premises systems can only be accessed from within the organization\\'s intranet Let us evaluate the given choices based on above key details:

>> CloudHub-deployed Mule runtimes can ONLY be controlled using MuleSoft-hosted control plane. We CANNOT use Private Cloud Edition\\'s control plane to control CloudHub Mule Runtimes. So, option suggesting this is INVALID >> Using

CloudHub-deployed Mule runtimes in the shared worker cloud managed by the MuleSoft-hosted Anypoint Platform is completely IRRELEVANT to given scenario and silly choice. So, option suggesting this is INVALID

>> Using an on-premises installation of Mule runtimes that are completely isolated with NO external network access, managed by the Anypoint Platform Private Cloud Edition control plane would work for On-premises integrations. However,

with NO external access, integrations cannot be done to SaaS-based apps. Moreover CloudHub-hosted apps are best-fit for integrating with SaaS-based applications. So, option suggesting this is BEST WAY.

The best way to configure and use Anypoint Platform to support these mixed/hybrid integrations is to use a combination of CloudHub-deployed and manually provisioned on- premises Mule runtimes managed by the MuleSoft-hosted Platform

control plane.

QUESTION 5

An API implementation is updated. When must the RAML definition of the API also be updated?

A. When the API implementation changes the structure of the request or response messages

B. When the API implementation changes from interacting with a legacy backend system deployed on-premises to a modern, cloud-based (SaaS) system

C. When the API implementation is migrated from an older to a newer version of the Mule runtime

D. When the API implementation is optimized to improve its average response time

Correct Answer: A

When the API implementation changes the structure of the request or response messages

****

>> RAML definition usually needs to be touched only when there are changes in the request/response schemas or in any traits on API.

>> It need not be modified for any internal changes in API implementation like performance tuning, backend system migrations etc..