

DUMPS ARENA

MuleSoft Certified Platform Architect - Level 1 MAINTENANCE

Mulesoft MCPA-Level-1-Maintenance

Version Demo

Total Demo Questions: 10

Total Premium Questions: 80

Buy Premium PDF

<https://dumpsarena.co>

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

QUESTION NO: 1

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

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

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

ANSWER: B**QUESTION NO: 2**

Mule applications that implement a number of REST APIs are deployed to their own subnet that is inaccessible from outside the organization.

External business-partners need to access these APIs, which are only allowed to be invoked from a separate subnet dedicated to partners - called Partner-subnet. This subnet is accessible from the public internet, which allows these external partners to reach it.

Anypoint Platform and Mule runtimes are already deployed in Partner-subnet. These Mule runtimes can already access the APIs.

What is the most resource-efficient solution to comply with these requirements, while having the least impact on other applications that are currently using the APIs?

- A. Implement (or generate) an API proxy Mule application for each of the APIs, then deploy the API proxies to the Mule runtimes
- B. Redeploy the API implementations to the same servers running the Mule runtimes
- C. Add an additional endpoint to each API for partner-enablement consumption
- D. Duplicate the APIs as Mule applications, then deploy them to the Mule runtimes

ANSWER: A**QUESTION NO: 3**

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

ANSWER: C

Explanation:

Explanation

(Correct Answer): The API consumer creates an API client, which sends API invocations to an API such that they are processed by an API implementation

Terminology:

>> API Client - It is a piece of code or program that is written to invoke an API

>> API Consumer - An owner/entity who owns the API Client. API Consumers write API clients.

>> API - The provider of the API functionality. Typically an API Instance on API Manager where they are managed and operated.

>> API Implementation - The actual piece of code written by API provider where the functionality of the API is implemented. Typically, these are Mule Applications running on Runtime Manager.

QUESTION NO: 4

An Order API must be designed that contains significant amounts of integration logic and involves the invocation of the Product API.

The power relationship between Order API and Product API is one of "Customer/Supplier", because the Product API is used heavily throughout the organization and is developed by a dedicated development team located in the office of the CTO.

What strategy should be used to deal with the API data model of the Product API within the Order API?

- A.** Convince the development team of the Product API to adopt the API data model of the Order API such that the integration logic of the Order API can work with one consistent internal data model
- B.** Work with the API data types of the Product API directly when implementing the integration logic of the Order API such that the Order API uses the same (unchanged) data types as the Product API
- C.** Implement an anti-corruption layer in the Order API that transforms the Product API data model into internal data types of the Order API
- D.** Start an organization-wide data modeling initiative that will result in an Enterprise Data Model that will then be used in both the Product API and the Order API

ANSWER: C

Explanation:

Explanation

(Correct Answer): Convince the development team of the product API to adopt the API data model of the Order API such that integration logic of the Order API can work with one consistent internal data model

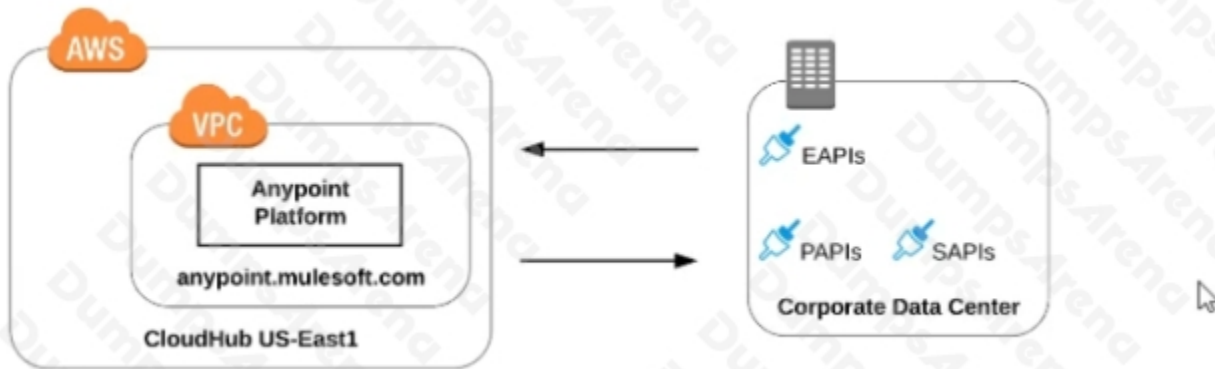
Key details to note from the given scenario:

>> Power relationship between Order API and Product API is customer/supplier

So, as per below rules of "Power Relationships", the caller (in this case Order API) would request for features to the called (Product API team) and the Product API team would need to accomodate those requests.

QUESTION NO: 5

Refer to the exhibit.



what is true when using customer-hosted Mule runtimes with the MuleSoft-hosted Anypoint Platform control plane (hybrid deployment)?

- A. Anypoint Runtime Manager initiates a network connection to a Mule runtime in order to deploy Mule applications
- B. The MuleSoft-hosted Shared Load Balancer can be used to load balance API invocations to the Mule runtimes
- C. API implementations can run successfully in customer-hosted Mule runtimes, even when they are unable to communicate with the control plane
- D. Anypoint Runtime Manager automatically ensures HA in the control plane by creating a new Mule runtime instance in case of a node failure

ANSWER: C

Explanation:

Explanation

(Correct Answer): API implementations can run successfully in customer-hosted Mule runtimes, even when they are unable to communicate with the control plane.

>> We CANNOT use Shared Load balancer to load balance APIs on customer hosted runtimes

◦ Load balancing

Load balancing is not provided for hybrid deployments. You can manage load balancing with the tools connected to your on-premises resources.

>> For Hybrid deployment models, the on-premises are first connected to Runtime Manager using Runtime Manager agent. So, the connection is initiated first from On-premises to Runtime Manager. Then all control can be done from Runtime Manager.

>> Anypoint Runtime Manager CANNOT ensure automatic HA. Clusters/Server Groups etc should be configured before hand.

Only TRUE statement in the given choices is, API implementations can run successfully in customer-hosted Mule runtimes, even when they are unable to communicate with the control plane. There are several references below to justify this statement.

References:

<https://docs.mulesoft.com/runtime-manager/deployment-strategies#hybrid-deployments>

<https://help.mulesoft.com/s/article/On-Premise-Runtimes-Disconnected-From-US-Control-Plane-June-18th-2018>

<https://help.mulesoft.com/s/article/Runtime-Manager-cannot-manage-On-Prem-Applications-and-Servers-from-US-Control-Plane-June-25th-2019>

<https://help.mulesoft.com/s/article/On-premise-Runtimes-Appear-Disconnected-in-Runtime-Manager-May-29th-2018>

On-Premise Runtimes Disconnected From US Control Plane - June 18th 2018

Jun 19, 2018 · RCA

Content

Impacted Platforms	Impacted Duration
--------------------	-------------------

Anypoint Runtime Manager / On-Prem Runtimes	During this time frame, on-prem runtimes appeared disconnected from the US Anypoint Control Plane: June 18, 2018 10:35 AM PST to June 18, 2018 11:12 AM PST
---	--

Incident Description

On-premises applications weren't able to connect to Anypoint Runtime Manager during the length of the incident, which made on-premises runtimes to throw errors in their logs because they received network disconnect messages from the control plane. Other than generating the log as mentioned above entries, on-premises runtimes and applications were not impacted.

=====
Runtime Manager cannot manage On-Prem Applications and Servers from US Control Plane - June 25th 2019

🕒 Jul 3, 2019 · RCA

Content

Incident Summary

Between 2:51 p.m. PT June 25th and 12:41 a.m. PT June 26th, customers were not able to manage their On-Prem applications and servers. The availability of running applications and runtimes were not impacted.

Impacted Platforms Impact Duration

US-Prod 9 hours and 50 minutes

=====
On-premise Runtimes Appear Disconnected in Runtime Manager - May 29th 2018

🕒 Jun 2, 2018 · RCA

Content

Impacted Platforms Impacted Duration

Anypoint Runtime Manager / On-Prem Runtimes	During this time frame, on-prem runtimes appeared disconnected from the US Anypoint Control Plane: Tuesday, May 29, 2018, 3:35 AM PDT to 4:27 AM PDT
---	---

Incident Description

During the incident time frame, managed Runtimes running on-premises disconnected from the US Anypoint Platform Control Plane and may have encountered recurrent re-connection errors. Customers were unable to manage applications running on those runtimes or register new ones during this time. Runtimes and Applications continued to operate without impact.

QUESTION NO: 6

A company has created a successful enterprise data model (EDM). The company is committed to building an application network by adopting modern APIs as a core enabler of the company's IT operating model. At what API tiers (experience, process, system) should the company require reusing the EDM when designing modern API data models?

- A. At the experience and process tiers
- B. At the experience and system tiers
- C. At the process and system tiers
- D. At the experience, process, and system tiers

ANSWER: C

Explanation:

Explanation

(Correct Answer): At the process and system tiers

>> Experience Layer APIs are modeled and designed exclusively for the end user's experience. So, the data models of experience layer vary based on the nature and type of such API consumer. For example, Mobile consumers will need light-weight data models to transfer with ease on the wire, where as web-based consumers will need detailed data models to render most of the info on web pages, so on. So, enterprise data models fit for the purpose of canonical models but not of good use for experience APIs.

>> That is why, EDMs should be used extensively in process and system tiers but NOT in experience tier.

QUESTION NO: 7

Which of the below, when used together, makes the IT Operational Model effective?

- A. Create reusable assets, Do marketing on the created assets across organization, Arrange time to time LOB reviews to ensure assets are being consumed or not
- B. Create reusable assets, Make them discoverable so that LOB teams can self-serve and browse the APIs, Get active feedback and usage metrics
- C. Create reusable assets, make them discoverable so that LOB teams can self-serve and browse the APIs

ANSWER: C

Explanation:

Explanation

(Correct Answer): Create reusable assets, Make them discoverable so that LOB teams can self-serve and browse the APIs, Get active feedback and usage metrics.



QUESTION NO: 8

A company has started to create an application network and is now planning to implement a Center for Enablement (C4E) organizational model. What key factor would lead the company to decide upon a federated rather than a centralized C4E?

- A. When there are a large number of existing common assets shared by development teams
- B. When various teams responsible for creating APIs are new to integration and hence need extensive training
- C. When development is already organized into several independent initiatives or groups
- D. When the majority of the applications in the application network are cloud based

ANSWER: C

Explanation:

Explanation

(Correct Answer): When development is already organized into several independent initiatives or groups

>> It would require lot of process effort in an organization to have a single C4E team coordinating with multiple already organized development teams which are into several independent initiatives. A single C4E works well with different teams having at least a common initiative. So, in this scenario, federated C4E works well instead of centralized C4E.

QUESTION NO: 9

Version 3.0.1 of a REST API implementation represents time values in PST time using ISO 8601 hh:mm:ss format. The API implementation needs to be changed to instead represent time values in CEST time using ISO 8601 hh:mm:ss format. When following the semver.org semantic versioning specification, what version should be assigned to the updated API implementation?

- A. 3.0.2
- B. 4.0.0
- C. 3.1.0
- D. 3.0.1

ANSWER: B

Explanation:

Explanation

(Correct Answer): 4.0.0

As per semver.org semantic versioning specification:

Given a version number MAJOR.MINOR.PATCH, increment the:

- MAJOR version when you make incompatible API changes.
- MINOR version when you add functionality in a backwards compatible manner.
- PATCH version when you make backwards compatible bug fixes.

As per the scenario given in the question, the API implementation is completely changing its behavior. Although the format of the time is still being maintained as hh:mm:ss and there is no change in schema w.r.t format, the API will start functioning different after this change as the times are going to come completely different.

Example: Before the change, say, time is going as 09:00:00 representing the PST. Now on, after the change, the same time will go as 18:00:00 as Central European Summer Time is 9 hours ahead of Pacific Time.

>> This may lead to some uncertain behavior on API clients depending on how they are handling the times in the API response. All the API clients need to be informed that the API functionality is going to change and will return in CEST format. So, this considered as a MAJOR change and the version of API for this new change would be 4.0.0

QUESTION NO: 10

An organization has created an API-led architecture that uses various API layers to integrate mobile clients with a backend system. The backend system consists of a number of specialized components and can be accessed via a REST API. The process and experience APIs share the same bounded-context model that is different from the backend data model. What additional canonical models, bounded-context models, or anti-corruption layers are best added to this architecture to help process data consumed from the backend system?

- A. Create a bounded-context model for every layer and overlap them when the boundary contexts overlap, letting API developers know about the differences between upstream and downstream data models
- B. Create a canonical model that combines the backend and API-led models to simplify and unify data models, and minimize data transformations.
- C. Create a bounded-context model for the system layer to closely match the backend data model, and add an anti-corruption layer to let the different bounded contexts cooperate across the system and process layers

D. Create an anti-corruption layer for every API to perform transformation for every data model to match each other, and let data simply travel between APIs to avoid the complexity and overhead of building canonical models

ANSWER: C

Explanation:

Explanation

(Correct Answer): Create a bounded-context model for the system layer to closely match the backend data model, and add an anti-corruption layer to let the different bounded contexts cooperate across the system and process layers

>> Canonical models are not an option here as the organization has already put in efforts and created bounded-context models for Experience and Process APIs.

>> Anti-corruption layers for ALL APIs is unnecessary and invalid because it is mentioned that experience and process APIs share same bounded-context model. It is just the System layer APIs that need to choose their approach now.

>> So, having an anti-corruption layer just between the process and system layers will work well. Also to speed up the approach, system APIs can mimic the backend system data model.