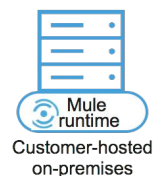
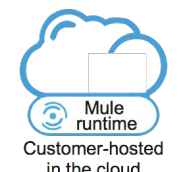
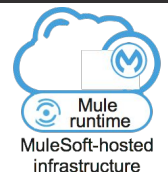
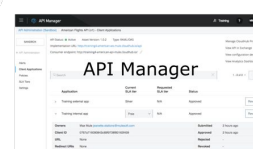
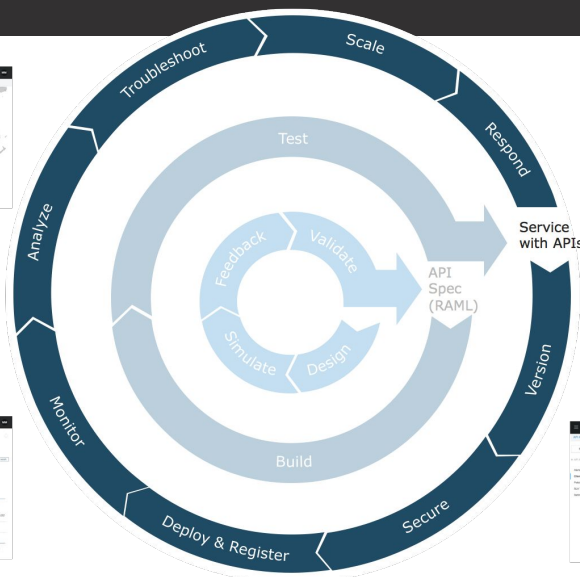
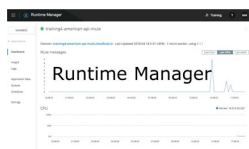
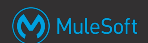




Part 2: Operationalizing Integration Solutions

Goal



At the end of this part, you should be able to

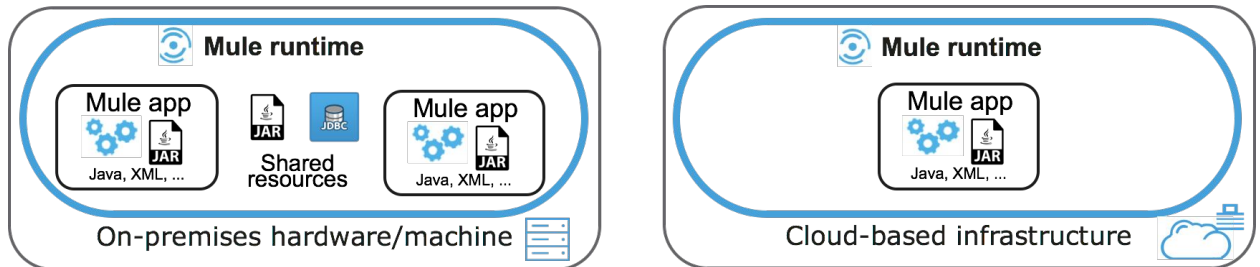


- Decide and develop appropriate and optimal deployment strategies
- Decide the best ways to preserve and manage Mule application state
- Design effective and sufficient logging and monitoring strategies
- Create an efficient and automated software development lifecycle (SDLC)



Module 7: Deciding and Developing a Deployment Strategy





At the end of this module, you should be able to

- Identify the service and deployment models supported by Anypoint Platform
- Decide deployment options for various scenarios
- Design containerized deployments for Mule runtimes

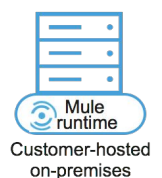
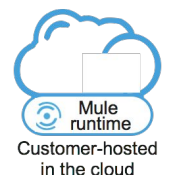
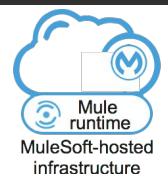
Distinguishing between various Anypoint Platform service models



Identifying the various Mule runtime hosting service models provided by Anypoint Platform



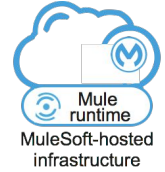
- The **MuleSoft-hosted** runtime plane uses cloud-based infrastructure (iPaaS)
 - **Provisioned** and managed by MuleSoft
 - The service is called **CloudHub**
- The **Customer-hosted** runtime plane is entirely provisioned and managed by the **customer** in
 - Non-MuleSoft-hosted cloud environments
 - Examples: AWS, Azure, Google Cloud, PCF
 - On-premises infrastructure, on bare-metal, virtual machines, or containers like docker



Features of the MuleSoft-hosted runtime plane



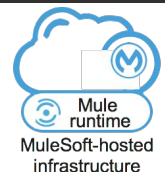
- Deployment to the **MuleSoft-hosted** runtime plane uses cloud-based infrastructure (iPaaS)
 - A new virtual machine is automatically provisioned for the new Mule runtime
 - This is called a **CloudHub worker**
 - Each Mule application is deployed to a separate Mule runtime which runs in a completely isolated CloudHub worker
- A Mule application can be scaled vertically or horizontally
 - Horizontal scaling: The Mule application can be automatically deployed to multiple CloudHub workers
 - Vertical scaling: A Mule application can have its CloudHub worker resized to a larger or smaller vCore size



Other components of a MuleSoft-hosted runtime plane



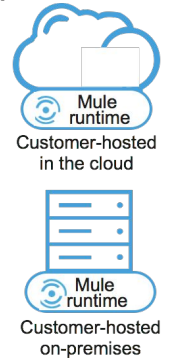
- MuleSoft provides a **load balancing** service
 - Distributes inbound HTTP requests among multiple CloudHub workers
 - Provides automatic redirection to new CloudHub worker(s) after a Mule application is resized or restarted
 - Resulting in **zero downtime** for the Mule application
 - A shared VPC restricts access to CloudHub workers to specific ports
 - 80/443 and 8081/8082
- MuleSoft implements a distributed **object store** service
 - Object stores and VM queues are implemented using this object store service
 - These can be configured so data survives restarting the Mule application



Ways to deploy to multiple Mule runtimes in customer-hosted infrastructure



- With just the Mule runtime software, the customer is responsible for coordinating how a Mule application is deployed to multiple Mule runtimes, and manage any load balancing or other scaling
- **Runtime Manager Fabric** is additional software provided by MuleSoft to scale out customer-hosted Mule runtimes
 - Mule applications deploy to isolated Mule runtime which run in a docker container
 - The containers can be deployed to any cloud service such as Google Cloud, MicroSoft Azure, and AWS
 - Provides additional services and infrastructure to provide CloudHub like services that run on-premises
 - Like zero-downtime redeployment and load balancing



Comparing Anypoint Platform control plane options

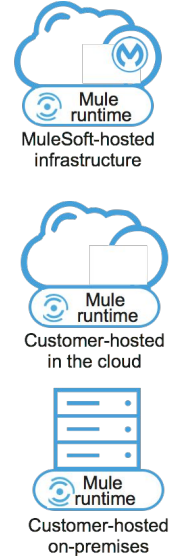


Plane type	MuleSoft-hosted Anypoint Platform	Anypoint Platform - Private Cloud Edition
Managed by	<ul style="list-style-type: none">• MuleSoft	<ul style="list-style-type: none">• Customer
Features	<ul style="list-style-type: none">• Full Anypoint Platform MuleSoft-hosted control plane• Can deploy Mule apps to CH workers• View logs in Runtime Manager for Mule apps deployed to CloudHub workers	<ul style="list-style-type: none">• Most Anypoint Platform features• Can not deploy Mule apps to CH workers• Logs are not available in Runtime Manager
SLAs	<ul style="list-style-type: none">• MuleSoft automatically patches and updates	<ul style="list-style-type: none">• Customer must install patches and updates

Identifying the various Mule runtime hosting service models provided by Anypoint Platform



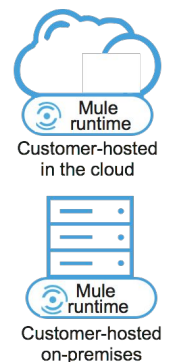
- In each service model, the **same Mule runtime binary executable** is used in a particular runtime plane (infrastructure)



Customer responsibility when using Mule runtimes installed in a customer-hosted runtime plane



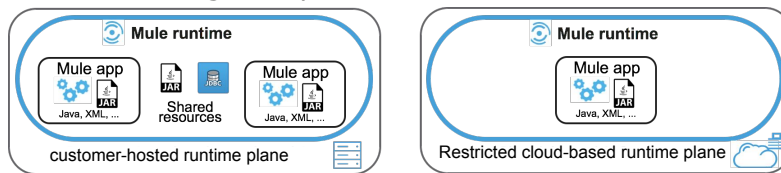
- For customer-hosted, the customer is responsible for provisioning and managing all
 - Hardware, virtual machines, cloud environments, and operating systems
 - Networks, proxies, load balancing, and high availability services
 - Java versions, security enha10



The service models for customer-hosted runtime planes



- The **Customer-hosted** runtime plane (either on bare metal or in virtual machines) can be configured to
 - Allow deployment of multiple Mule applications and domain to the same Mule runtime
 - Host multiple Mule runtimes
- Like CloudHub, other **cloud-based** infrastructure (like PCF) can also be locked down
 - May not allow deployment of multiple Mule applications or Mule domains
 - May not allow hosting multiple Mule runtimes in the same virtual machine



All contents © MuleSoft Inc.

14

15

MuleSoft-hosted vs. Customer-hosted runtime plane



MuleSoft-hosted



Each Mule runtime is automatically installed in a separate CloudHub worker

Can host one Mule application in a dedicated CloudHub worker

No contention for host resources

Automatic upgrades/patches

Globally available

MuleSoft provided security and SLAs

Managed by the MuleSoft-hosted Anypoint Platform control plane

Customer-hosted



Mule runtimes can be deployed to various types (on-premises or cloud) of customer-hosted infrastructure

Some customer-hosted infrastructure can run multiple Mule applications in one Mule runtime

Can have contention for host resources

Customer must schedule and install updates

Customer must provision across regions and multiple premises

Customer defines and controls security and SLAs

Managed by the MuleSoft-hosted or customer-hosted Anypoint Platform control plane

All contents © MuleSoft Inc.

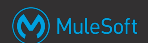
16

Reflection questions: Runtime planes



- What components make up the MuleSoft-hosted runtime planes?
- What components make up various customer-hosted runtime planes?
- What use cases would prohibit using a MuleSoft-hosted runtime plan
- What are the tradeoffs of these runtime plane options?
- What use cases would benefit from a hybrid collection of runtime planes?
- What are the tradeoffs in using a hybrid model?
- How could a hybrid model help an enterprise move to the Cloud?

Reflection questions: Runtime planes



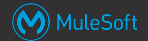
- For what use cases is the MuleSoft-hosted runtime plane the best option?
- For what use cases is a customer-hosted control plane the best option?

- What is the difference between a control plane and a runtime plane?
- What are the components of the MuleSoft-hosted control plane?
- What use cases would prohibit using a MuleSoft-hosted control plane?
- What use cases would use a combination of control planes, and what would be the tradeoffs and benefits?
- For what use cases is the MuleSoft-hosted control plane the best option?
- For what use cases is a customer-hosted control plane the best option?

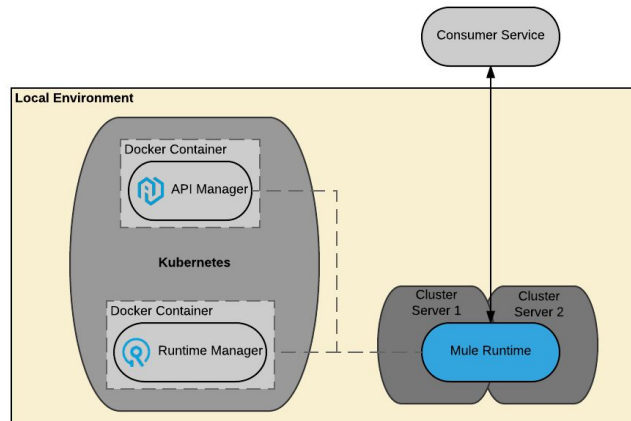
Distinguishing between various Anypoint Platform deployment models



Deployment models for Anypoint Platform - Private cloud edition

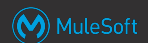


- For customers with **strict regulatory or compliance requirements**

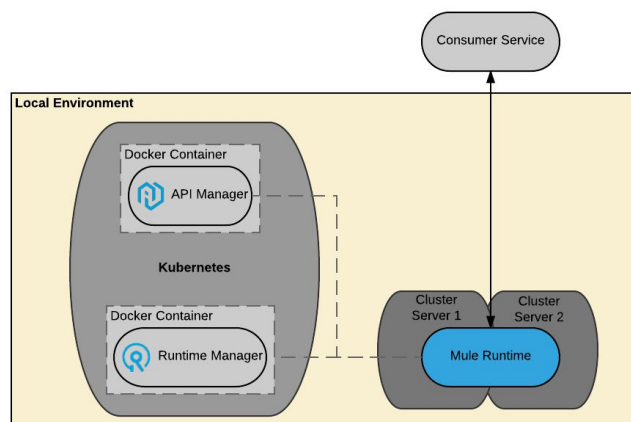


21

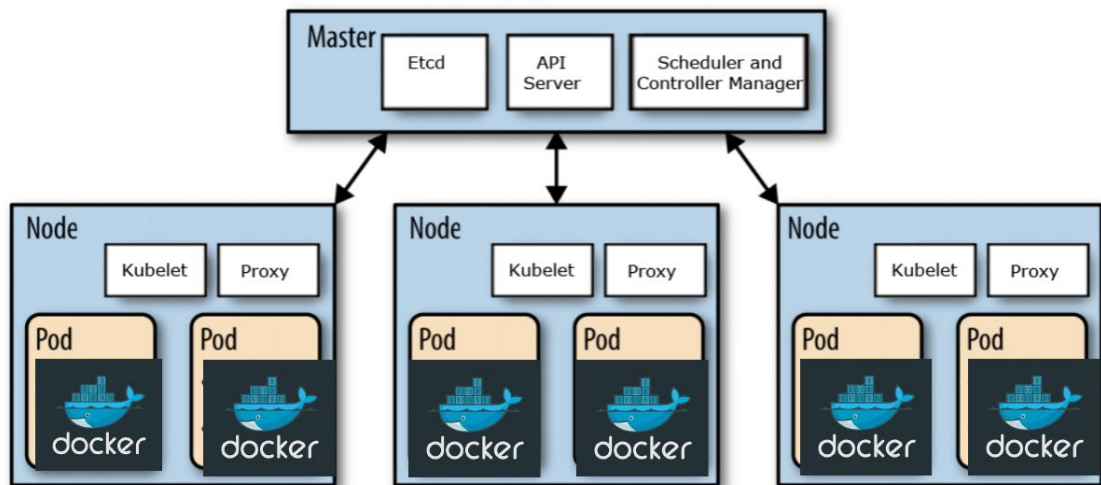
Deployment models for Anypoint Platform - Pivotal cloud foundry (PCF)



- Same as Anypoint Platform - Private Cloud Edition, except the runtime plane is deployed on PCF instance
- Like **CloudHub**, one Mule application per Mule runtime

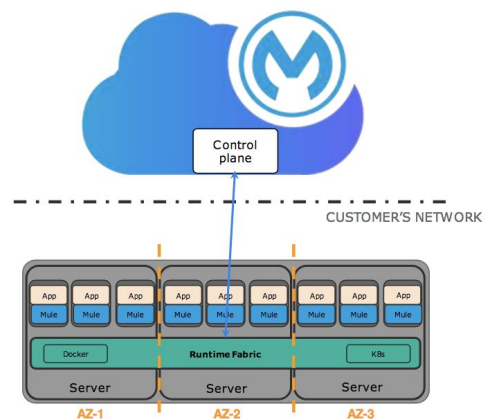


22

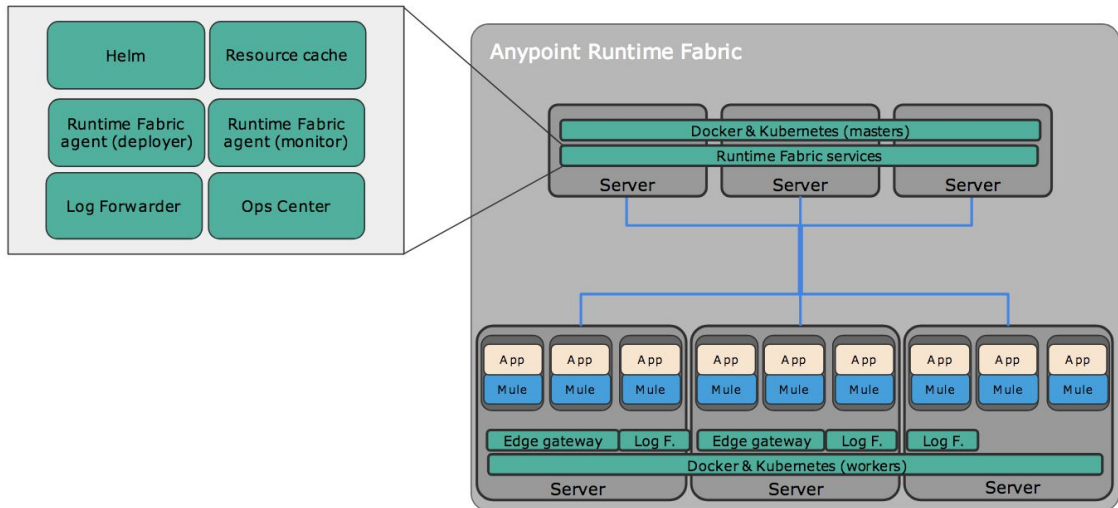


Containerized deployment on Anypoint Platform- Anypoint Runtime Fabric

- Anypoint Runtime Fabric orchestrates and automates the deployment of Mule runtimes into containers in any cloud or on premises
- Deploy consistently across any cloud (Azure & AWS) or data center
- Run multiple Mule runtime versions in the same Runtime Fabric
- Isolate apps, scale horizontally, redeploy w/ zero downtime
- Connect to the control plane hosted by MuleSoft
- No need to dockerize Mule apps



Anypoint Runtime Fabric Components



All contents © MuleSoft Inc.

28

Comparing Anypoint Platform runtime plane options



Plane type	MuleSoft hosted in CloudHub	Runtime Fabric hosted	Customer-hosted
Managed by	<ul style="list-style-type: none"> MuleSoft 	<ul style="list-style-type: none"> Customer 	<ul style="list-style-type: none"> Customer
Description	<ul style="list-style-type: none"> iPaaS solution managed by MuleSoft 	<ul style="list-style-type: none"> Many CH-like runtime services provisioned into customer-hosted infrastructure 	<ul style="list-style-type: none"> Individual Mule runtimes are provisioned into customer-hosted infrastructure No CH-like runtime services provided
Features	<ul style="list-style-type: none"> Inbound HTTP load balancing VM queue messages are load balanced for deployments to multiple CH workers 	<ul style="list-style-type: none"> Inbound HTTP load balancing VM queue messages are load balanced for deployments to a cluster of Mule runtimes 	<ul style="list-style-type: none"> No inbound HTTP load balancing provided VM queue messages are load balanced for deployments to a cluster of Mule runtimes
Mule app SLAs	<ul style="list-style-type: none"> Rescale Mule runtimes (CH workers) with zero downtime 	<ul style="list-style-type: none"> Rescale Mule runtimes with zero downtime 	<ul style="list-style-type: none"> None

All contents © MuleSoft Inc.

29

Comparing access to other Anypoint Platform runtime services based on the runtime plane type



Plane type	MuleSoft hosted in CloudHub	Customer-hosted (including Runtime Fabric hosted)
Object Store types	<ul style="list-style-type: none"> Connect to the Anypoint Object Store service with the Object Store connector In-memory OS 	<ul style="list-style-type: none"> Connect to the Anypoint Object Store service via REST API In-memory OS File-backed OS Hazelcast-backed OS
Anypoint MQ	<ul style="list-style-type: none"> Connect to AnypointMQ service with the AnypointMQ connector 	<ul style="list-style-type: none"> Connect to AnypointMQ service with the AnypointMQ connector (public internet access required)
Safely-hidden properties	<ul style="list-style-type: none"> Properties can be hidden from users of Runtime Manager 	<ul style="list-style-type: none"> Not available
Access security and restrictions	<ul style="list-style-type: none"> Mule apps can always access all CH services Security is automatically managed within MuleSoft managed infrastructure/networks 	<ul style="list-style-type: none"> Mule apps must have public internet access to these online services

All contents © MuleSoft Inc.

30

Deployment models for Anypoint Platform



Deployment type	CloudHub	Standalone Mule runtime	Anypoint Runtime Fabric	Anypoint Platform - Private Cloud Edition
Cloud-based				
Managed by	MuleSoft	MuleSoft	MuleSoft	MuleSoft
On-premises / private IaaS				
Managed by		Customer	Appliance: MuleSoft Infrastructure: Customer	Customer

* Includes load balancing, scaling, and zero downtime features

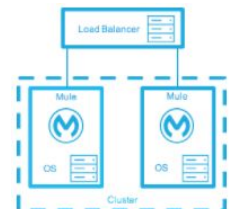
All contents © MuleSoft Inc.

31

- What control plane is used by Runtime Fabric?
- How do Runtime Fabric's runtime plane features and behaviors compare with the MuleSoft-hosted runtime plane
- How do Runtime Fabric's runtime plane features and behaviors compare with other customer-hosted runtime plane options?
- For which use cases is Runtime Fabric the best choice of runtime plane?

Clustered fault-tolerant deployment

- Only available for **customer-hosted runtime** planes
- A **cluster** is a set of Mule runtimes that acts as a unit
 - Servers in a cluster communicate and share information through a Hazelcast-based **distributed shared memory grid**, data is replicated across memory in different physical machines
- All worker nodes work in **active-active** model and always one (auto-elected) primary node
- Node are aware of each other
- If one node fails, outstanding tasks transfer automatically to surviving nodes
- Nodes can be added and subtracted from a cluster to adjust load capacity



Understanding Mule domains deployed to customer-hosted runtime planes



- A **Mule domain** project can be used to share global configuration elements between Mule applications, which can
 - Ensure **consistency** between Mule applications upon any changes, as the **configuration** is only set in one place
 - Expose multiple services within the Mule domain on the same port
 - Share the **connection** to persistent storage
 - Send events (messages) to other Mule applications using VM queues
- This allows Mule applications to **share resources** but NOT behaviours
- **ONLY** available for customer-hosted Mule runtimes, but not for Anypoint Runtime Fabric

How Mule domains are created



- Create a Mule Domain Project with a particular **domain name**
- Add global element configurations to the domain project
- Associate Mule applications with the domain name
 - Each Mule application automatically shares all the Mule domain's global elements

- How do customer-hosted Mule runtimes compare with an Anypoint Runtime Fabric runtime plane?
- Which scenarios are better suited for Anypoint Runtime Fabric, and which are better suited for customer-hosted Mule runtimes?
- Which control planes can control customer-hosted Mule runtimes vs. Anypoint Runtime Fabric?

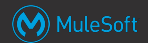
Deciding deployment option for various scenarios



- **Regulatory** requirements of on-premises processing
 - Including metadata about API invocations and messages
- **Time-to-market**
- **IT operations effort**
- Accessing **on-premises data sources**
- **Flexibility of deployment across cloud providers**
- **Isolation** between Mule apps
- **Control over Mule runtime tuning**
- **Scalability** of runtime plane
 - horizontal and vertical; static and dynamic
- Roll-out of **new releases**
- **Redeployment with zero downtime**

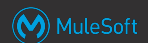
- High availability (HA)
- Automated failover
- Out of box features required
 - Object store
 - Shared resource support
 - Persistent queues
 - Scheduling
 - Logging
 - Dashboards
 - Insights
 - Alerts
 - Auto scaling
- Anypoint Edge Security
- Tokenization

Deciding deployment option for various scenarios



- Technical notes for deciding deployment options are included in student files

Factors for deciding the best runtime plane to use based on organizational business goals



Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted	Pivotal Cloud Foundry
Complex regulatory compliance required	Good	Maybe better	Maybe better	Maybe better
Time to market	Best	Better	Better	Maybe OK
Minimize IT operations effort	Best	Better	Good	OK

Deciding the best runtime plane based on performance and reliability goals



Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Scalability	Auto scaling	Manual scaling	Manual scaling	
HA (multiple workers)	Yes	Yes	unless managed	Yes
Automated failover	Yes	Yes	unless managed	Yes
Redeployment with zero downtime	Supported	Supported	More work	Supported
Clustering of Mule runtimes	No	Yes*	Yes	Yes*

* With an external Hazelcast cluster service

42

Deciding the best runtime plane based on deployment flexibility goals



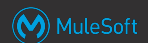
Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Runtime tuning	Not supported	Not supported	Complete control	Template control
Upgrade to new Mule runtime version	Easy	Easy	More work	Moderate work
Access to on-prem data	VPC or over internet	Easier	Easier	Easier
Flexibility to deploy across providers	NA	Easy except GCP not supported	more work	NA

Deciding the best runtime plane based on state management goals



Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Object Store service within the same Runtime plane	Supported	No	No	No
Persistent queues	Supported	Only in a cluster	Supported	Only in a cluster
Access to OSv2	Via an Object Store connector or the OSv2 REST API	Via the OSv2 REST API	Via the OSv2 REST API	Via the OSv2 REST API

Deciding the best runtime plane based on reusability goals



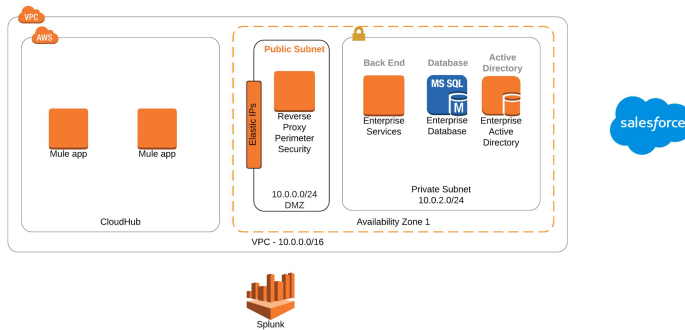
Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Shared resources between apps	No	No	Yes (Mule domains)	No
Isolation between apps	Best (on VMs)	Better	From No to Best depending on configuration	Best (on VMs)

Exercise 7-1: Identify the best deployment model for a specific scenario



- Analyze deployment options for a specific scenario
- Design an optimal service model and deployment model for a specific scenario based on functional and non-functional requirements

ANYAIRLINE CURRENT PHYSICAL ARCHITECTURE



All contents © MuleSoft Inc.

46

Exercise scenario: Organizational requirements



- Data integration should have **minimal network latency**
- **Existing monitoring** capabilities of **Splunk** should be utilized
- **Existing HTTP-based services** should be utilized for cloud based **Salesforce** and **Splunk** systems
- The organization has **strict data residency requirements** and the **organization infrastructure** is **PCI compliant**
- Organization's **DevOps** has **limited expertise** in **containerization**

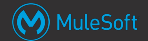
All contents © MuleSoft Inc.

47

- The organization has decided to use MuleSoft Anypoint Platform as the integration platform for data integration between various systems
- The Mule runtimes can use any available runtime plane that best meets the scenario's requirements
- High availability and scaling goals need to be supported
 - Mule applications should be able to be scaled "horizontally" across multiple Mule runtimes, so the services are highly available
 - Should be possible to scale the Mule runtime's host infrastructure "vertically" without service interruption

- The deployment model is influenced by below functional and non-functional requirements of the organization
 - HA and Scalability
 - Network latency
 - Monitoring
 - Networking/load balancing for endpoints
 - Security of data in rest/transit
 - Limited containerization capability of DevOps

Exercise step: Analyze deployment options for the scenario



Runtime plane

CloudHub

Anypoint
Runtime Fabric

Customer-hosted or
Private Cloud Edition

Pivotal Cloud
Foundry

HA and Scalability

Network latency

Monitoring

Networking

Data security

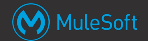
DevOps
containerization
capability

Exercise step: Follow this decision tree to arrive at the best answer



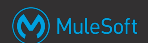
- AWS autoscaling can be achieved using deployment on EC2 instances under ELB control
 - Can CloudHub, PCF, or Runtime Fabric be used to create these types of deployments?
 - If not, what runtime plane choices remain?
- What are the network latency requirements for this scenario?
- What is the fastest way to connect an on-prem network to a Mule runtimes in EC2 virtual images?

Exercise step: Deciding deployment options based on the type of control plane



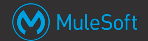
Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Scheduling				
CH enhanced logging				
Dashboard				
Insights				
Alerts				

Exercise step solution: Deciding deployment options based on the type of control plane



Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
Scheduling	Yes	No	Yes	Yes
CH enhanced logging	Yes	No	No	No
Dashboard	Yes	No	Yes	No
Insights	Yes	Basic support	Yes	No
Alerts	Yes	No	Yes	No

Exercise step: Decide deployment option for various possible scenario requirements



Runtime plane	CloudHub	Anypoint Runtime Fabric	On-prem runtime	Private Cloud Edition	Pivotal Cloud Foundry
Auto scaling					
Anypoint Monitoring					
Anypoint Visualizer					
Anypoint Edge Security					
Anypoint Tokenization					
Load balancing					
DLB support					
Anypoint MQ					
File Persistence					

All contents © MuleSoft Inc.

54

Exercise step solution: Decide deployment option for various possible scenario requirements

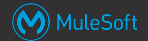


Runtime plane	CloudHub	Anypoint Runtime Fabric	On-prem runtime	Private Cloud Edition	Pivotal Cloud Foundry
Auto scaling	Yes	Manual scaling	unless managed	unless managed	unless managed
Anypoint Monitoring	Yes	No	No	No	No
Anypoint Visualizer	Yes	No	No	No	No
Anypoint Edge Security	No	Yes but minimal support	Yes	No	No
Anypoint Tokenization	No	No	Yes	No	No
Load balancing	Yes	Yes	No	No	No
DLB support	Yes	No	unless managed	unless managed	unless managed

All contents © MuleSoft Inc.

55

Exercise step solution: Deciding deployment option for various scenarios



Runtime plane	CloudHub	Anypoint Runtime Fabric	On-prem runtime	Private Cloud Edition	Pivotal Cloud Foundry
Anypoint MQ	Yes	No	No	No	No
File Persistence	Yes	Less reliable	Yes	Yes	Yes

Exercise step solution: Analyze deployment options for the scenario



Runtime plane	CloudHub	Anypoint Runtime Fabric	Customer-hosted or Private Cloud Edition	Pivotal Cloud Foundry
HA and Scalability	Preferred	Preferred	Preferred	Preferred
Network latency	No (unless customer-hosted Mule runtimes are also on AWS EC2)	Preferred	Preferred	Preferred
Monitoring	No	Preferred	Preferred	Preferred
Networking	No	Preferred	Preferred	Preferred
Data security	No	Preferred	Preferred	Preferred
DevOps containerization capability	Preferred	Preferred	Preferred	No

Exercise step: Deciding



- Using AWS auto-scaling requires deployment on EC2 instances under ELB control, which rules-out PCF, CH and RTF
- Hence only customer-manager "on-prem" Mule runtimes on EC2 remain
- Network latency demands fast IPsec or similar to on-prem network

Exercise step: Select runtime and control planes to meet requirements



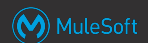
- Which runtime and control planes can support AWS auto-scaling in EC2 instances under ELB control?
- What is the best way to minimize network latency between on-premises systems and the Mule runtimes running in EC2 under ELB control?

Exercise solution: Decision points



- Which runtime and control planes can support AWS auto-scaling in EC2 instances under ELB control?
 - PCF, CloudHub, and Runtime Fabric cannot support this requirement
 - The best choice is customer-managed Mule runtimes deployed into EC2 instances under ELB control
- What is the best way to minimize network latency between on-premises systems and the Mule runtimes running in EC2 under ELB control?
 - Fast IPsec or similar must be used to connect the EC2 instances with the on-premises network

Exercise step



- Decide the best two deployment options for the organization and summarize how each deployment model best serves organization requirement
- After some discussion, apply your analysis to identify the best deployment model for the scenario which meets the requirements

- Customer-hosted Mule runtimes seem to meet all of the organization's requirements
 - Customer-hosted deployment minimizes network latency for data integrations
 - Customer-hosted runtimes in RTF or on-prem solution can use existing monitoring capability of Splunk
 - The data reside in customer environment so strict data residency requirement for organization is met
 - Customer-hosted runtimes in RTF or on-prem solution do not require DevOps with expert capability in K8s and Docker
 - No more outbound SSL endpoints as runtimes are deployed in existing infrastructure

- Verify and discuss the differences between Anypoint Runtime Fabric vs. customer-hosted Mule runtimes, and decide if Anypoint Runtime Fabric is also a viable option

Summary



Summary



- Every Mulesoft service and deployment model provides different features
- Understanding organization ecosystem is key in deciding the deployment model
- Realization of key features for organization is important while deciding deployment model
- MuleSoft adds new features in products often and informs customer about its offerings