

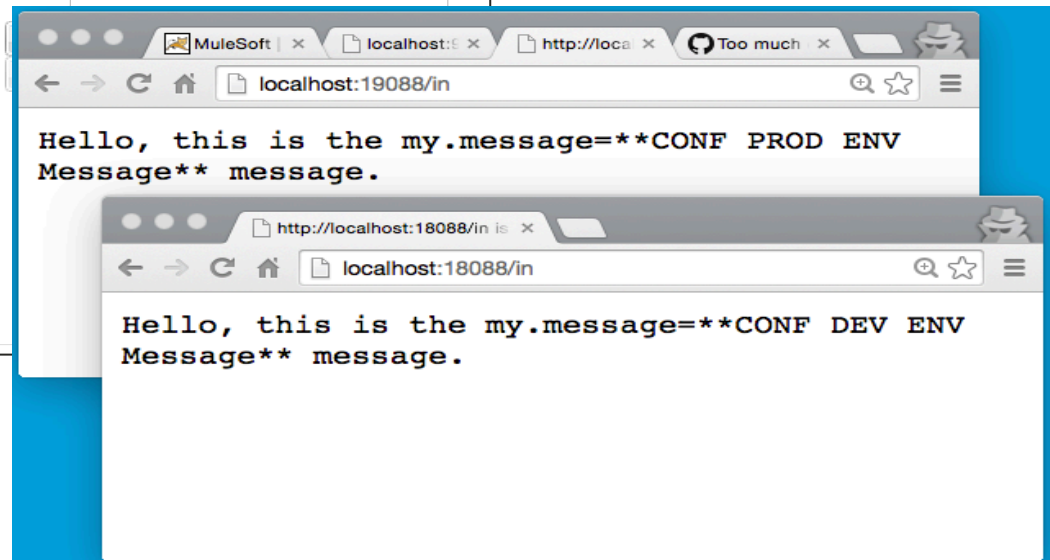


# Module 5: Managing Server Clusters



# Goal

The screenshot shows the MuleSoft Mule ESB Enterprise console. The left sidebar contains a navigation menu with 'Dashboard' selected. Below it, a 'New Group' button and a list of server groups: 'All (2)', 'Barren (2)', 'Development (2)', 'Production (1)', 'Staging (0)', 'Test (0)', and 'Unregistered (0)'. The main content area is titled 'Create Cluster'. It includes an information icon and text: 'Enter or modify the cluster name and assign nodes to the cluster. Selecting servers running agent. Make sure to... [more]'. There are two input fields: 'Cluster Name' with the value 'HABarren', and 'Available Nodes' which is currently empty. Below the 'Available Nodes' field are 'Select All' and 'Deselect All' buttons. To the right, under 'Cluster Nodes', there are two checkboxes: 'Apollo' and 'Max', both of which are unchecked. At the bottom, there is a checkbox labeled 'Multicast is disabled in my network.' which is checked, and 'Add' and 'Cancel' buttons.



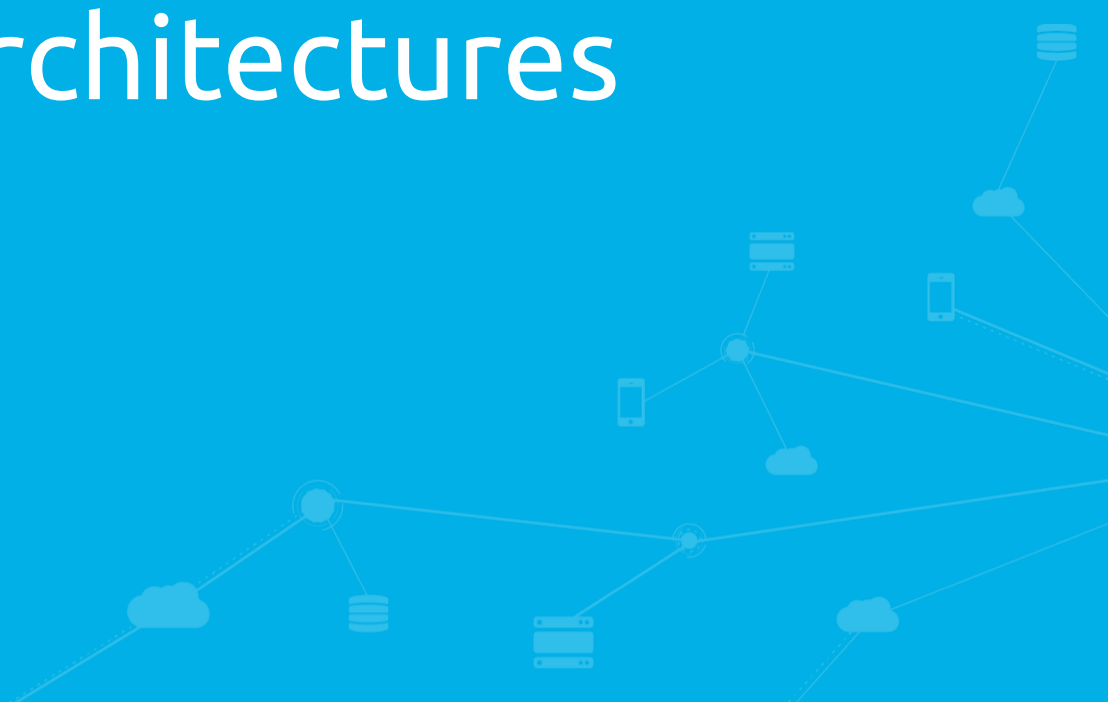
# Objectives

- Create a new Mule server Cluster
- Add Mule servers to a Cluster
- Deploy Applications to a Cluster
- Test cluster failover behavior

# Topics

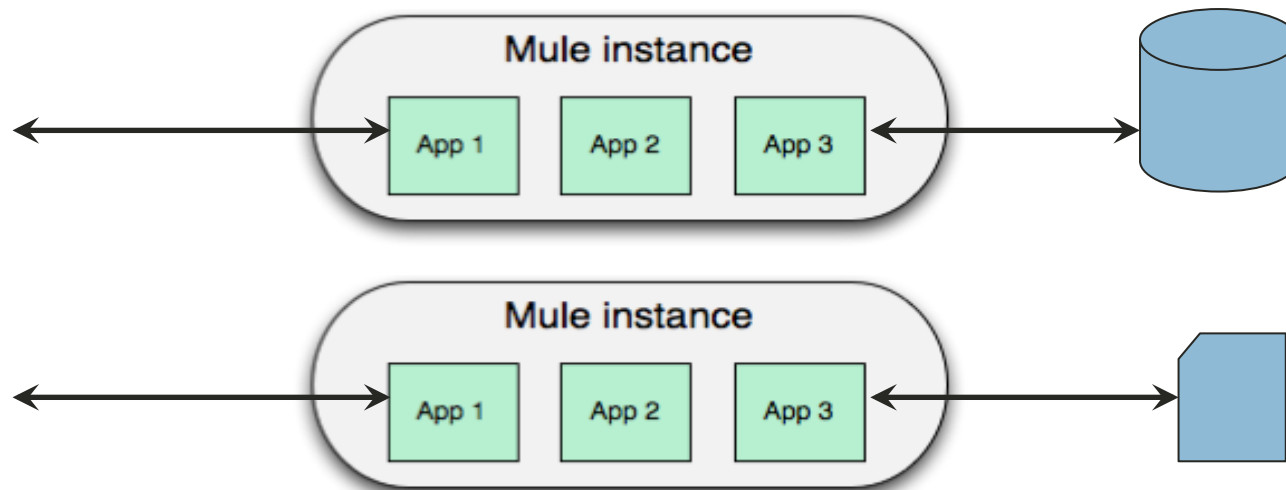
- Clustered server solutions and architectures:
  - What problem clusters solve
  - What they are and what they do
  - When to use clusters and when not to
- Building a Cluster:
  - From MMC
  - With Multicast enabled or disabled
- Deploying applications to clusters

# Clustered Servers Solutions and Architectures



# The need for Mule server clusters

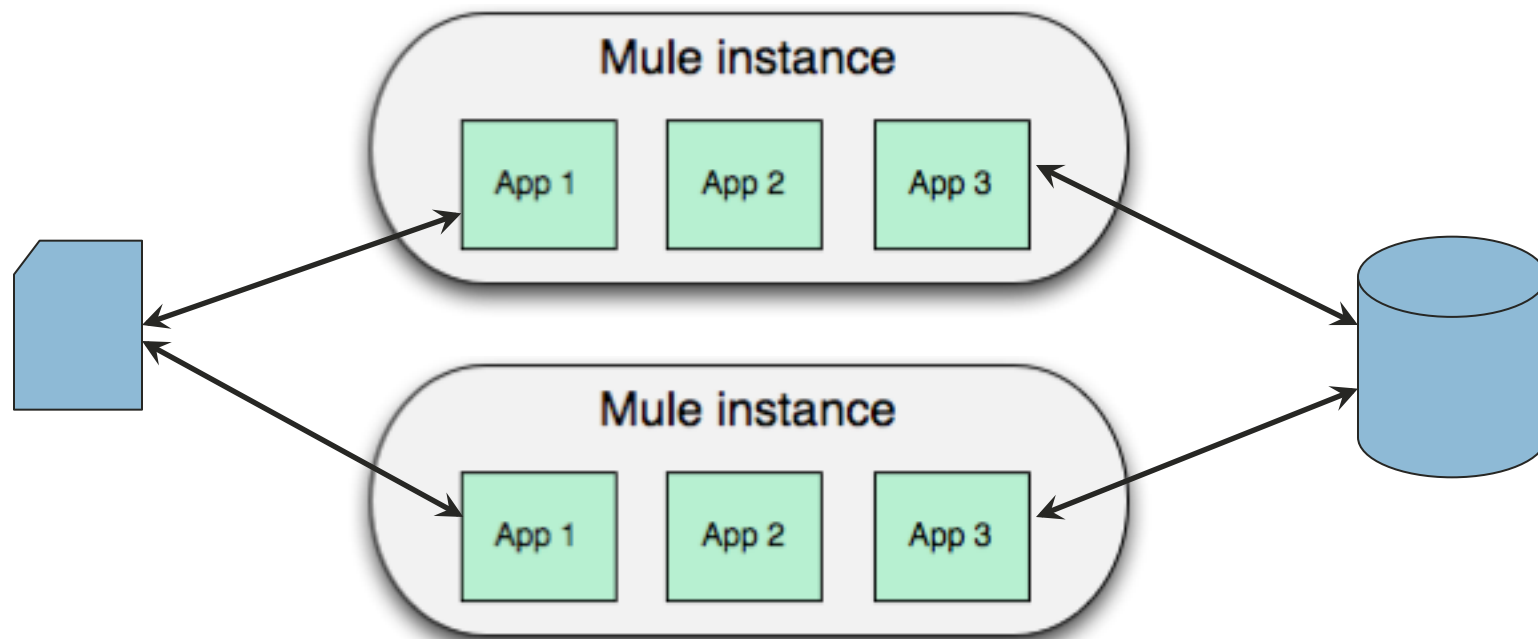
- So far we have two Mule instances:



- They work independently:
  - They don't communicate with each other or share resources
  - What kind of situations could independent Mule instances get into if they are running the same application?

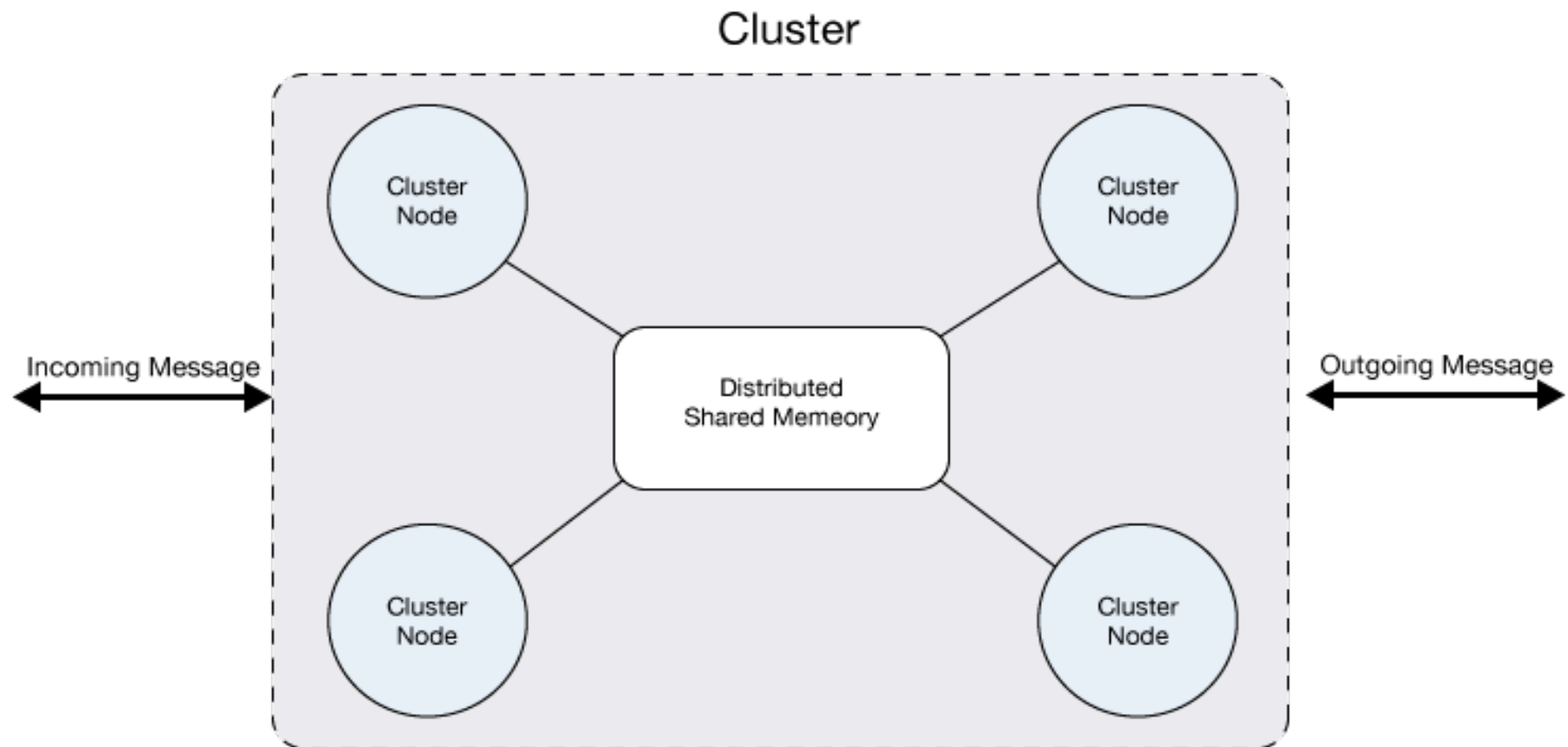
# Common problem: resource contention

- May poll the same resource!



# Solution: Clustering

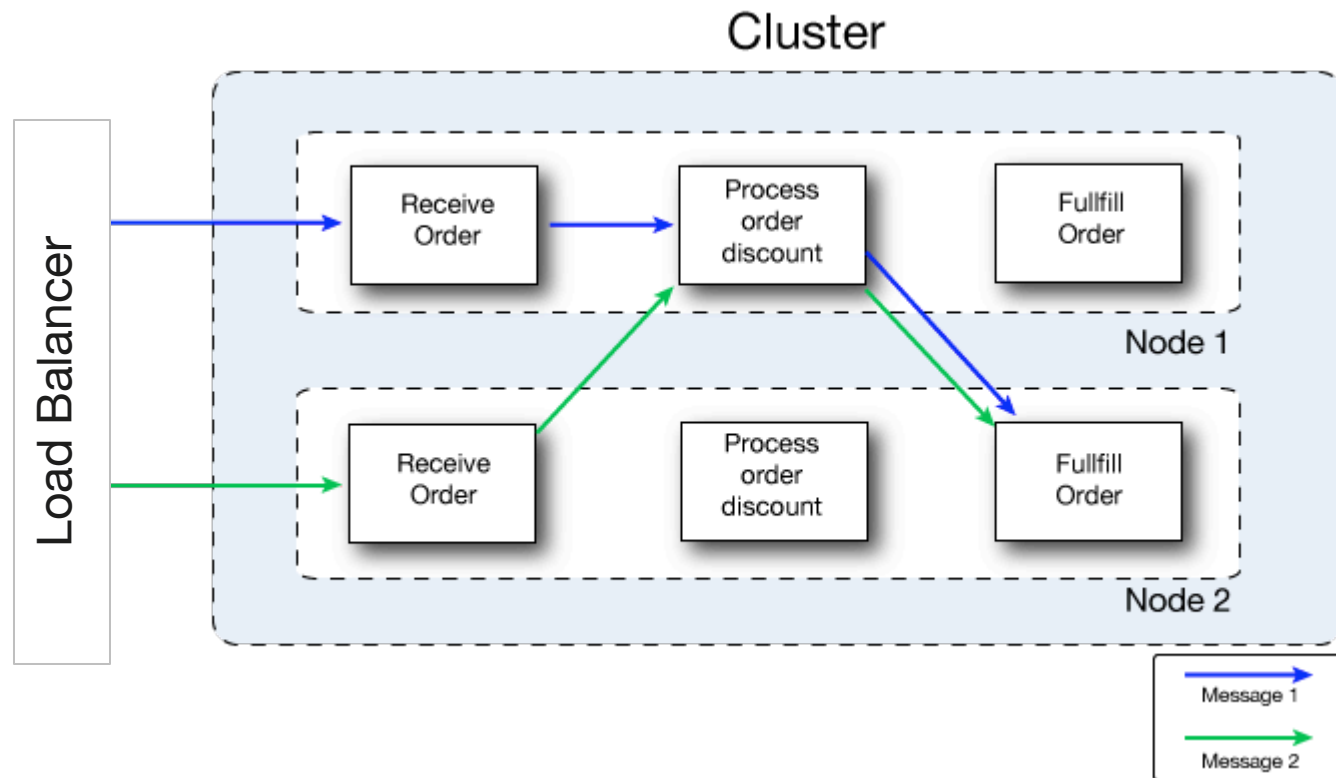
- A cluster is a set of 2 or more Mule Servers that act as a single server:





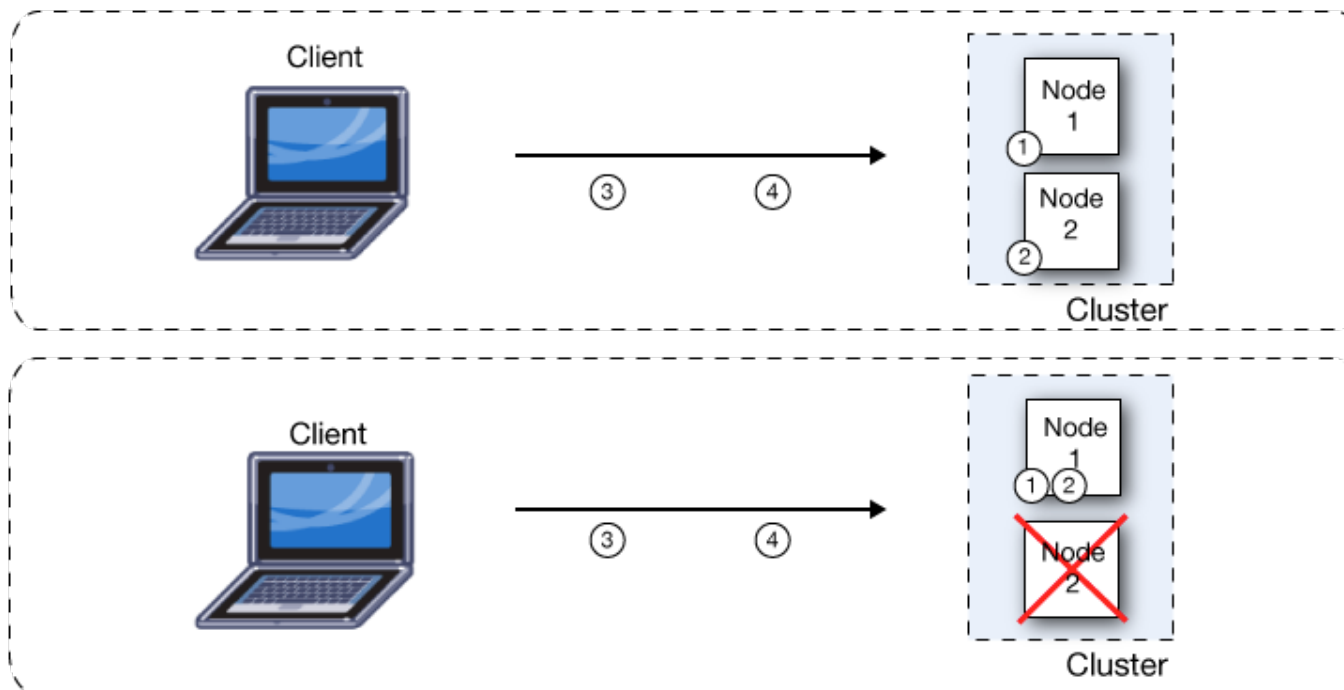
# Benefits of Clustering

- Load Balancing:
  - Sometimes you need extra hardware/software (e.g. for webservices)



## Benefits of Clustering (2)

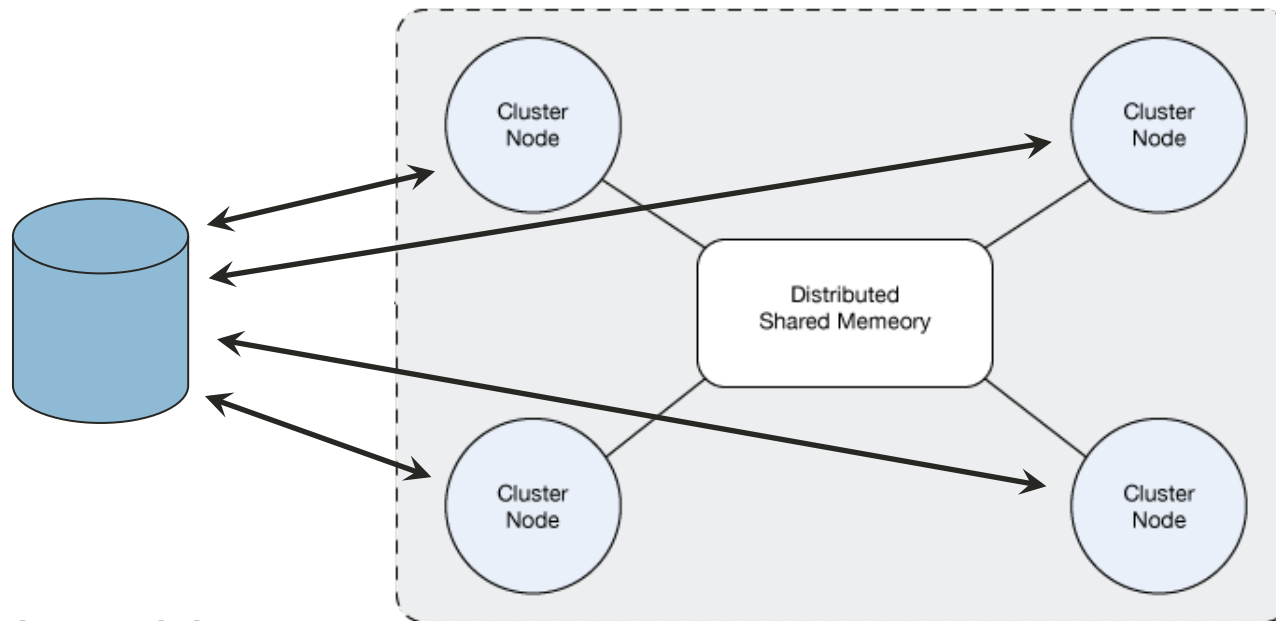
- High Availability:



- If one node fails, the other node will take the failed node's workload as well as its own

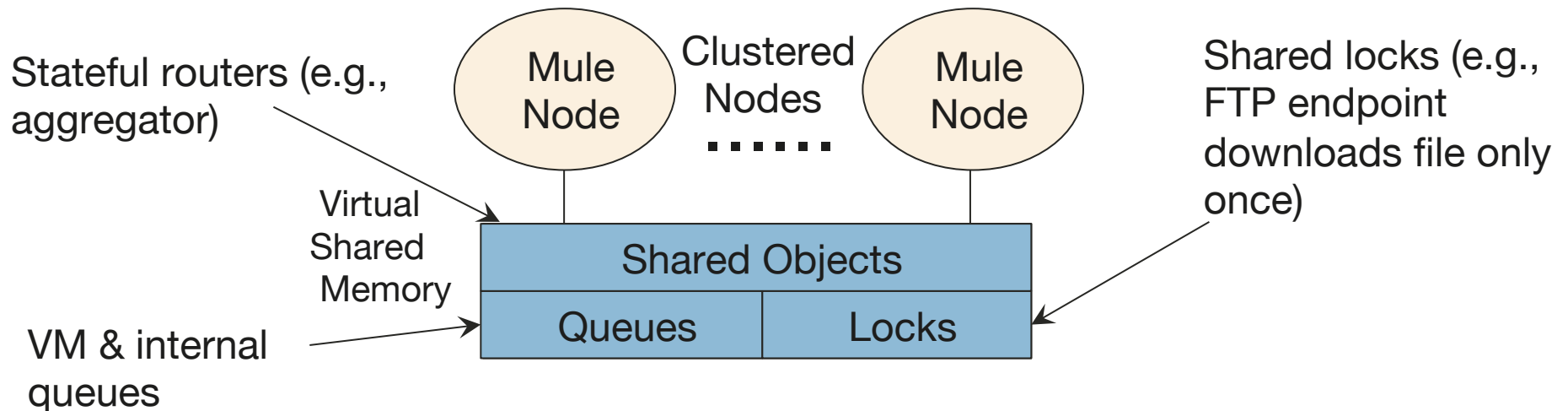
## Benefits of Clustering (3)

- Polling shared resources:



- Sharing object stores
- Components that store state:
  - Cache, Idempotent filters (to remove duplicates)

# Mule Active/Active Clusters



- All nodes share a common objects repository:
  - Actually, several repositories; one per SEDA Queue
  - All nodes have the same priority when reading/writing
- Shared memory is also distributed:
  - 271 partitions (buckets) for objects
  - Each object is replicated to 1 other node (default)
    - Can be changed up to 6
  - Automatically rebalanced

# When would you NOT use clustering?

Don't use clustering if:

- Your load-balancing is done by 3<sup>rd</sup> party components
  - Or you don't need load balancing and failover at all
- You need to scale up over 8 nodes (Mule supports more, just not officially...)
  - Some customers are successful with many more nodes
- Your nodes are geographically distributed
  - Or there is high latency in the network
- You handle large stream payloads (E.g. videos, large documents)
- All your services are synchronous (E.g. SOAP services)
  - And you don't need failover

# Building a Cluster



# Managing clusters

- Clusters are managed as if they were one single server
  - Doesn't matter if there are 2, 3 or 4 nodes, you will only see one entity:

The screenshot displays the MuleSoft management console interface. At the top, the 'All servers' section shows a single entity named 'MyCluster' with version 3.5.1 and location 127.0.1.1. Below this, the 'Nodes Control' section lists two nodes: 'Mule-3.5.1@192.168.2.33:7777' and 'Mule-3.5.1@192.168.2.33:7778'. An orange arrow points from the 'MyCluster' entry to the 'Nodes Control' section. The 'Cluster Metrics' section on the right shows detailed metrics for both nodes, including JVM Memory Usage, OS CPU Usage, OS Memory Usage, and OS Swap Usage, each with a corresponding line graph.

**All servers**

A server is a single instance of Mule ESB, either standalone or embedded in an application

Name	Version	Location
MyCluster	3.5.1	127.0.1.1, 127.0.1.1

**Nodes Control**

Restart Stop Remove from Cluster

Name	Agent URL	Location
Mule-3.5.1@192.168.2.33:7777	https://192.168.2.33:7777/mmc-support	127.0.1.1
Mule-3.5.1@192.168.2.33:7778	https://192.168.2.33:7778/mmc-support	127.0.1.1

**Cluster Metrics**

Name	Last Data	Trend Changes
Node: Mule-3.5.1@192.168.2.33:7777 (4 Items)		
JVM Memory Usage	129 MB	
OS CPU Usage	1 %	
OS Memory Usage	6 GB	
OS Swap Usage	951 MB	
Node: Mule-3.5.1@192.168.2.33:7778 (4 Items)		
JVM Memory Usage	105 MB	
OS CPU Usage	1 %	
OS Memory Usage	6 GB	
OS Swap Usage	951 MB	

# Walkthrough 5-1: Create a cluster

- Create a new Mule Server Cluster
- Add Mule servers to a Cluster

**All servers**

**A server is a single instance of Mule ESB, either standalone or embedded in an application**

Name	Version	Location
MyCluster	3.5.1	127.0.1.1, 127.0.1.1

**Nodes Control**

Restart Stop Remove from Cluster

Name	Agent URL	Location
Mule-3.5.1@192.168.2.33:7777	https://192.168.2.33:7777/mmc-support	127.0.1.1
Mule-3.5.1@192.168.2.33:7778	https://192.168.2.33:7778/mmc-support	127.0.1.1

**Cluster Metrics**

Name	Last Data	Trend Changes
<b>Node: Mule-3.5.1@192.168.2.33:7777 (4 Items)</b>		
JVM Memory Usage	129 MB	
OS CPU Usage	1 %	
OS Memory Usage	6 GB	
OS Swap Usage	951 MB	
<b>Node: Mule-3.5.1@192.168.2.33:7778 (4 Items)</b>		
JVM Memory Usage	105 MB	
OS CPU Usage	1 %	
OS Memory Usage	6 GB	
OS Swap Usage	951 MB	



# Things to keep in mind

- Cluster-wide operations are done sequentially:
  - Deployments
  - Undeployments
  - The more nodes you have, the longer it takes
  - Updating application; first undeploy old, then deploy new
- The clustering protocol is chatty and operations are synchronous:
  - Make sure that nodes are on a fast LAN
- The Applications tab is populated from MMC's repository:
  - Any application not deployed through MMC will NOT show
  - If you need to see all apps, click on the server name -> applications
  - Applications with the same name will be clustered anyway

# The gory details

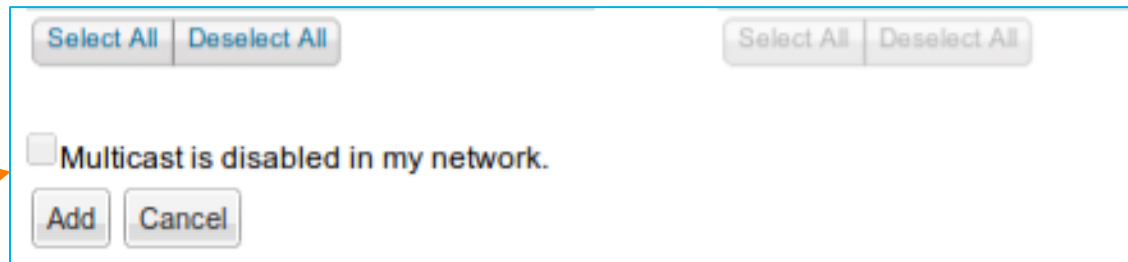
- What happens to Mules when you cluster them?
  - MMC creates `$MULE_HOME/.mule/mule-cluster.properties`
  - You can create this file manually, so that you can cluster Mules without even using MMC
  - Make sure your license supports clustering

```
#Mule cluster properties
#Fri Jul 19 16:17:52 PST 2014
mule.clusterSize=2
mule.clusterSchema=partitioned-sync2backup
mule.clusterId=5e775a47-f6b9-43da-8e85-31167a86b7b6
mule.clusterPartitioningMode=OPTIMIZE_PERFORMANCE
mule.clusterNodeId=2
```

# Multicast (mDNS)

- On startup, nodes find each other using Multicast (mDNS), but after that everything is Unicast
- If you want to disable Multicast:

- Either tick here when creating a cluster



- Or add this to `$MULE_HOME/conf/wrapper.conf`

```
wrapper.java.additional.15=-Dmule.cluster.multicastenabled=false  
wrapper.java.additional.16=-Dmule.cluster.nodes=a.b.c.d, e.f.g.h
```

- mule.cluster.nodes are TWO nodes that act as LOCATORS
  - Do NOT put all the IPs for all the nodes here.

## Walkthrough 5-2: Deploy applications to clusters

- Deploy Applications to a Cluster
- Test cluster fail over behavior

# Summary

- Mule Servers can be joined into Clusters
  - Implemented using Hazelcast distributed shared-memory grid
- Applications can be deployed to Clusters
  - Message processing is automatically Load Balanced for Asynchronous flows

