

Trade-off Factor – Cisco UCS Fabric Failover or OS based NIC Teaming using Dual Fabric

The Cisco UCS system has lot of different components which work together. All these components' working together seamlessly is what makes UCS look inherent. However a failure can happen at any point and this needs to be dealt with.

In a simple scenario of UCS system with a server with CNA card, following may happen:

- a) FI failure:** Results in fabric failure for all connected UCS chassis
- b) FEX failure:** Results in fabric failure for one UCS chassis
- c) FI-FEX link failure:** Results in fabric failure for some of the servers within a UCS chassis (depending on number of servers and uplinks)
- d) One CNA port failure:** Results in fabric failure for one server

In any of the above cases downtime can be eliminated by using redundant hardware and proper config.

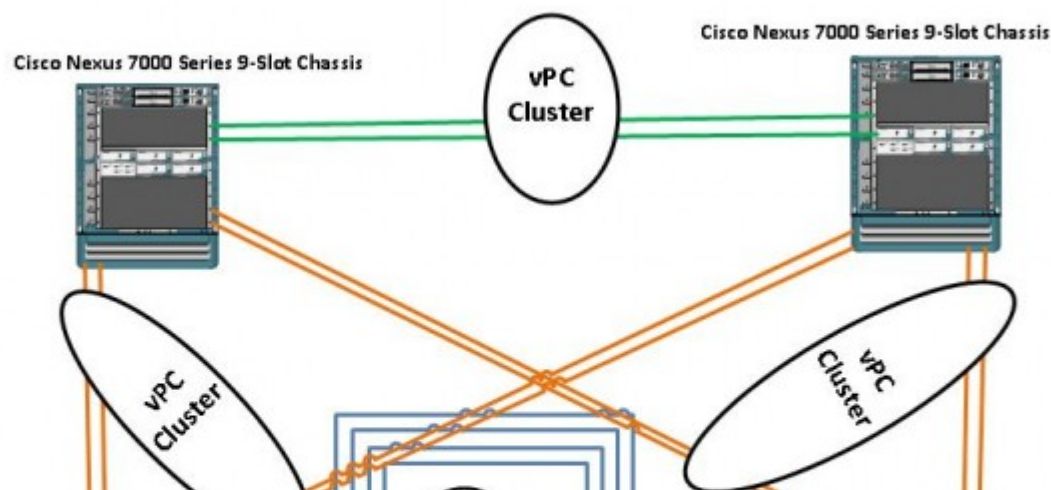
We have two options to deal with any of the above mentioned failure scenario. Those are:

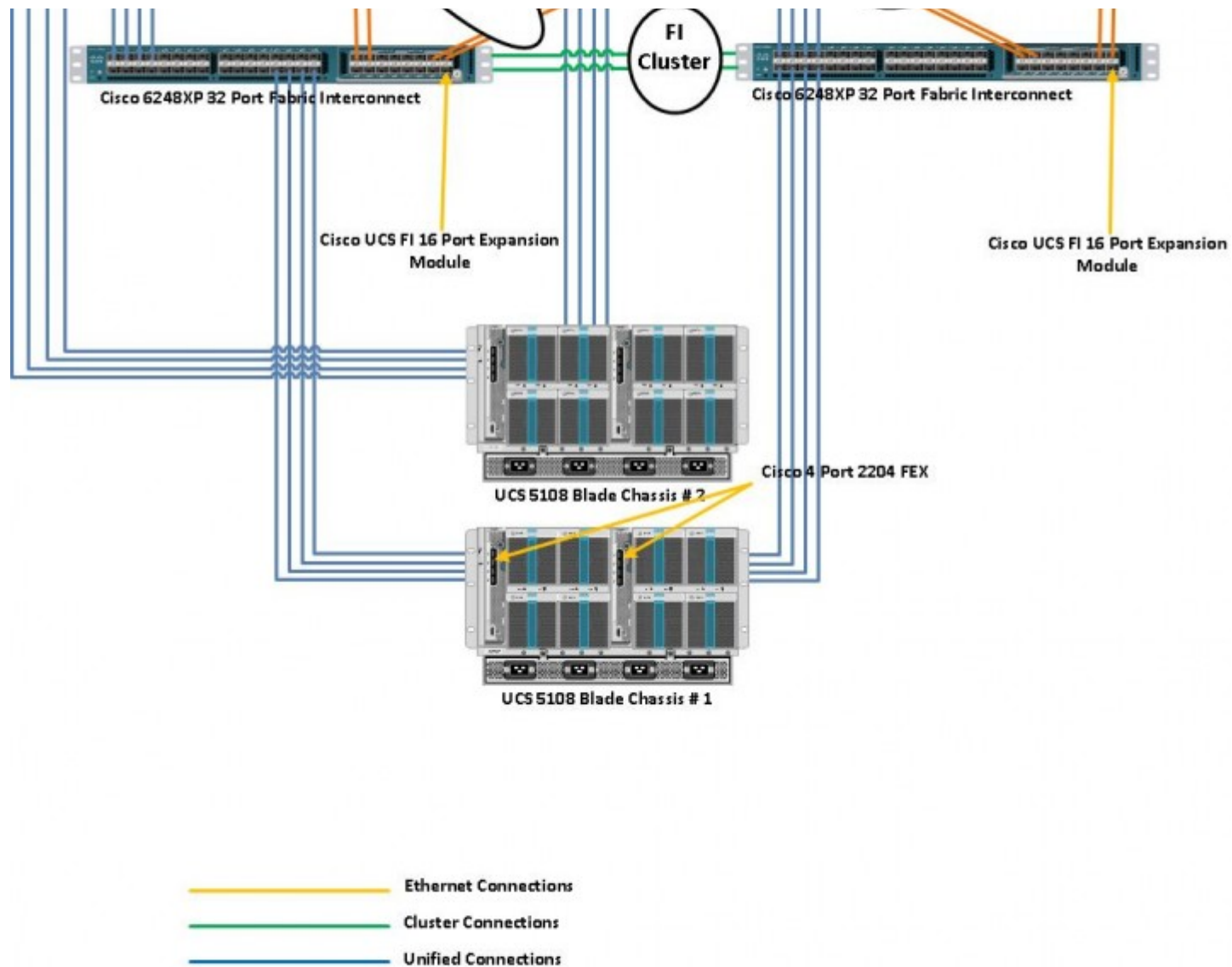
- a) Cisco UCS Fabric Failover**
- b) OS Based NIC Teaming**

But hey wait!!! What about the performance when it happens and even when it does not happen. Did you ever thought about it before we design it for resiliency?

Well, in this post I will tackle both of these two factors. So first let me take you to the Resiliency factor and then I would take you to Performance factors.

So, first of all let me show you a typical Cisco UCS deployment diagram.





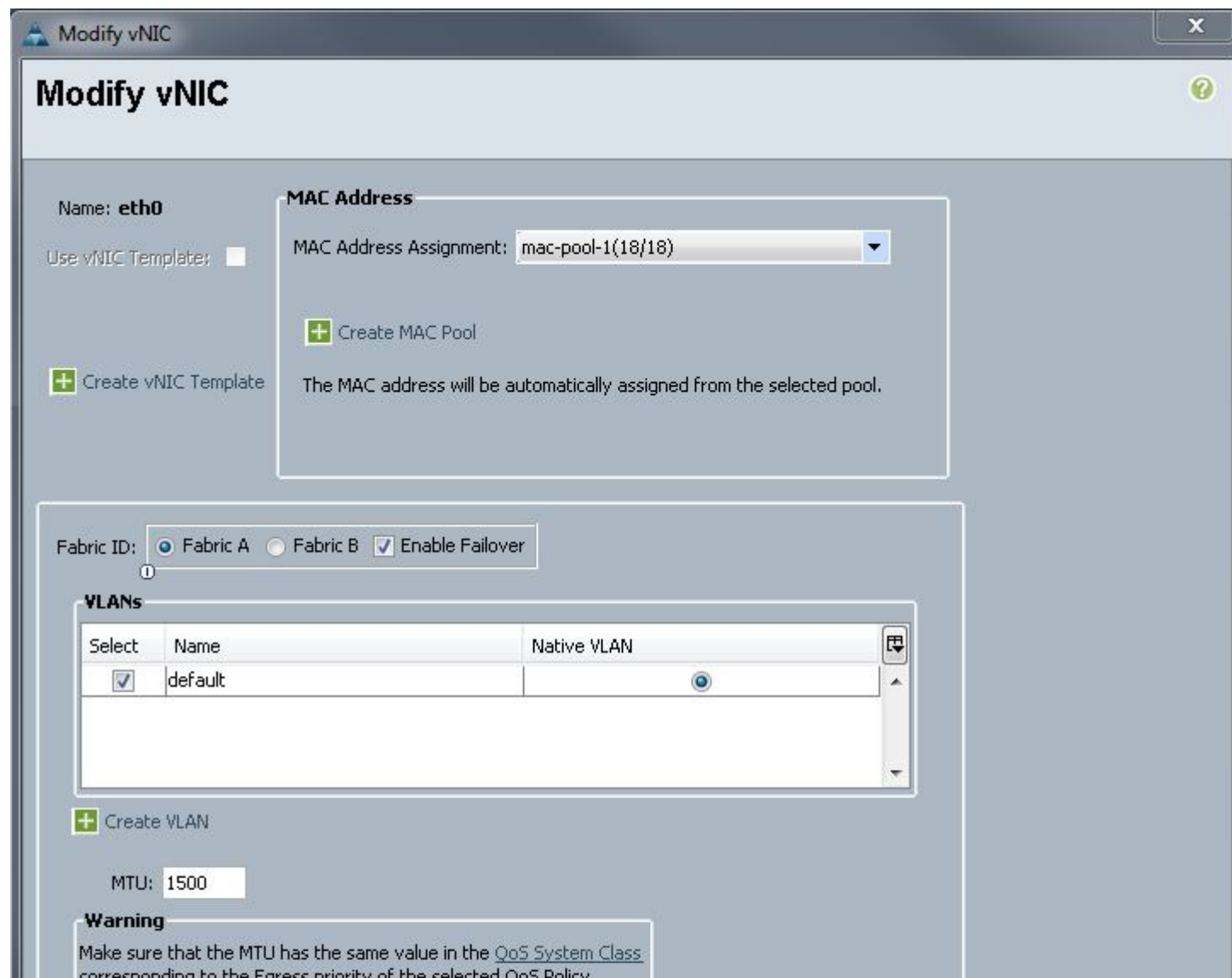
In each UCS chassis, there are 2 x 2204XP I/O Module
 From each FEX, there are 4 x 10GE uplinks to the Fabric Interconnect
 In each blade there is a single Palo network adapter
 This adapter provides a 10GE to each FEX in the chassis, a total of 2 x 10GE
 From each Fabric Interconnect there are 4 x 10GE to each Nexus 7K switch, 4 connections per switch and 8 x 10GE in total

In this deployment we have two UCS Chassis and 2 UCS Fabric Interconnect in Clustered mode. As a upstream core switch we are using Cisco Nexus 7000 switch in pair. So, you can see, in all layer we have maintained resiliency. Now let me take you to the each option of resiliency from UCSM perspective.

Fabric Failover is a unique capability found only in Cisco UCS that allows a server adapter to have a highly available connection to two redundant network switches without any NIC teaming drivers or any NIC failover configuration required in the OS.

With Fabric Failover, the intelligent network provides the server adapter with a virtual cable that can be quickly and automatically moved from one upstream switch to another. Shown in the diagram below.

So, your server NIC will be pinned to one single Fabric which you have to select at the time of creating Service Profile and you need to select "**Enable Failover**" to have the second Fabric in **Standby** for resiliency. It should look like below.



Modify vNIC

Name: **eth0**

Use vNIC Template: ☐

MAC Address

MAC Address Assignment: **mac-pool-1(18/18)**

[+ Create MAC Pool](#)

The MAC address will be automatically assigned from the selected pool.

Fabric ID: ☒ Fabric A ☐ Fabric B ☒ Enable Failover

VLANs

Select	Name	Native VLAN
<input checked="" type="checkbox"/>	default	<input checked="" type="radio"/>

[+ Create VLAN](#)

MTU: **1500**

Warning

Make sure that the MTU has the same value in the [QoS System Class](#) corresponding to the Egress priority of the selected QoS Policy.

Pin Group: <not set> + Create LAN Pin Group

Operational Parameters

Adapter Performance Profile

Adapter Policy: Windows + Create Ethernet Adapter Policy

Dynamic vNIC Connection Policy: <not set> + Create Dynamic vNIC Connection Policy

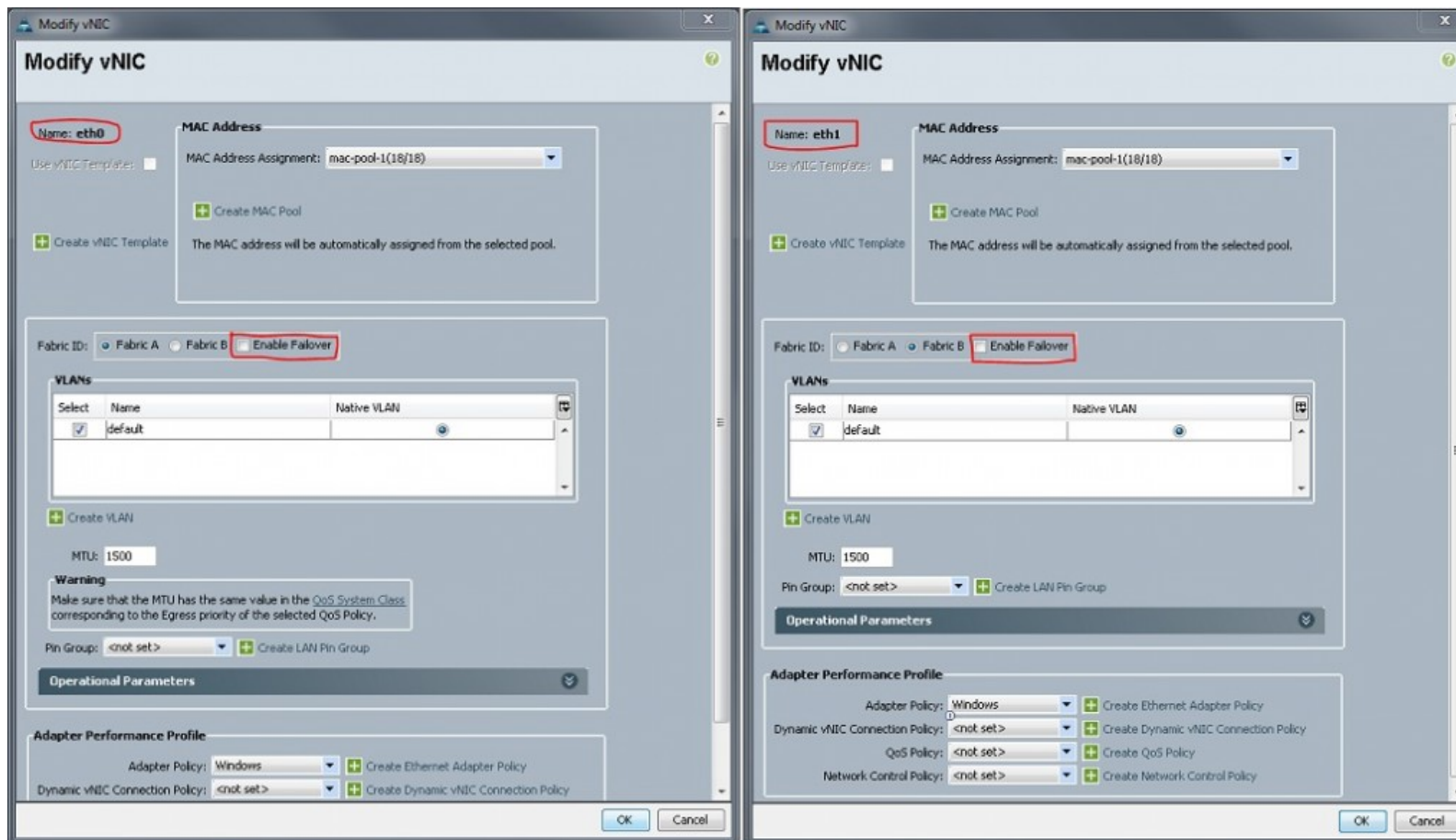
QoS Policy: qos-1 + Create QoS Policy

Network Control Policy: default + Create Network Control Policy

OK Cancel

In this scenario if any of the above mentioned scenario happened UCSM will seamlessly failover the NIC and it will re-pinned the NIC to the other Fabric. You may see one or two Ping RTO if you monitor it.

Now let us look at the other factor which is using OS based NIC teaming and pinning two Physical NIC to two different Fabric (Fabric A and Fabric B) which will make Active / Active connection from OS level to both of the Fabric. However in this case we would not use Fabric Failover, that means the option will remain unchecked. Now if you put both the NIC option side by side, it should look like as below.



Once you do this you need to install the NIC Teaming protocol inside the OS and then do the active/active teaming.

In this scenario you would get active/active connection to both the Fabric and OS NIC team driver will take care of your failover.

Now there are Performance consideration on using both of the mentioned scenarios. Now let me tell you what are those in each design.

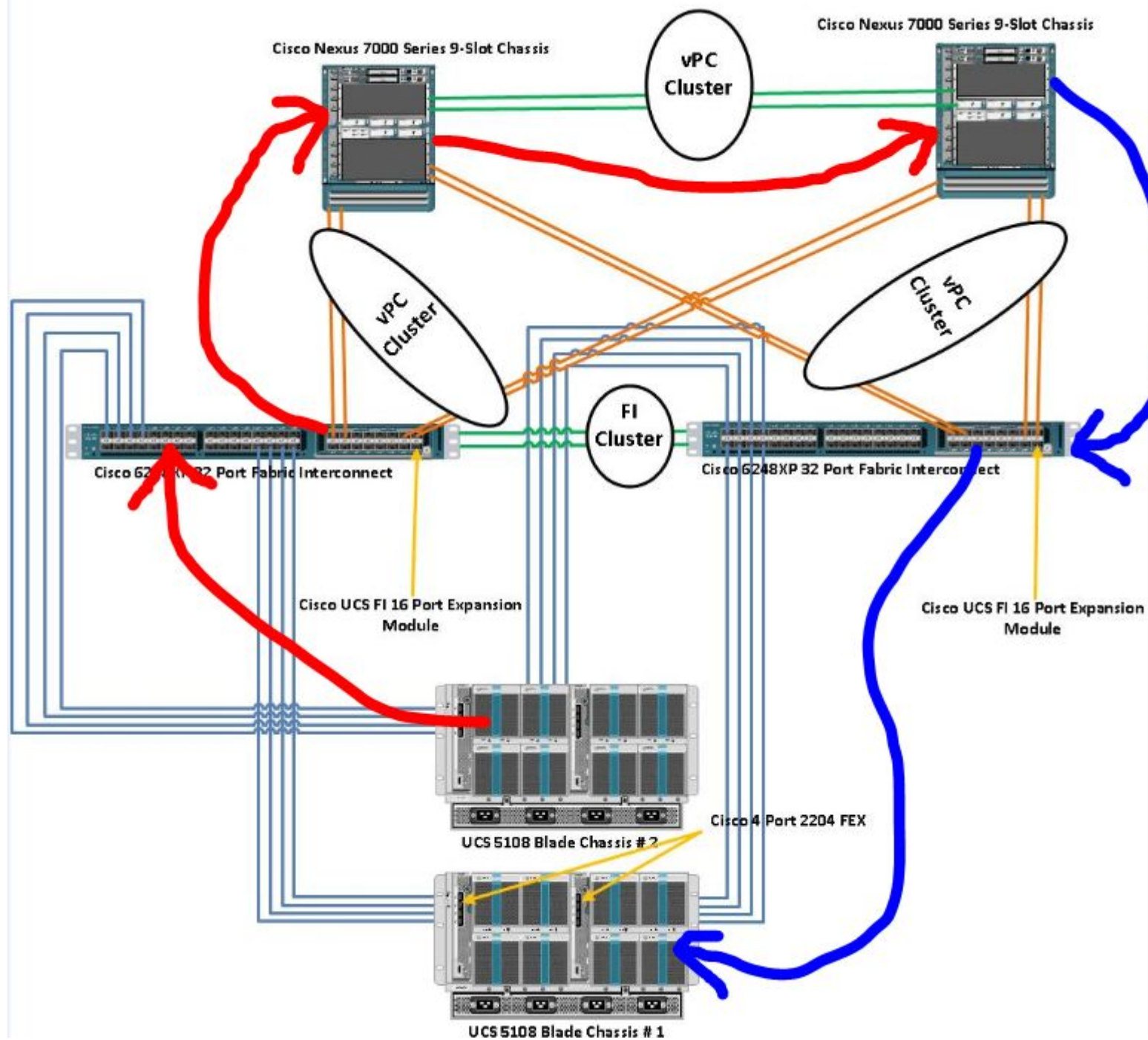
In OS based NIC teaming using both the Fabric, it makes a A/A connection, however this is true in case of sending traffic out. In case of receiving the traffic it would use effectively only one side of the Fabric. So, not much benefit.

Also OS based NIC teaming requires much manual step while preparing the system.

Now the bummer, if there are two servers in two different Chassis and they wanted to talk to each other or send bulk data in between, then even though they could be on same subnet, but they will send the traffic out to core and then it will channel back to the Server. Now the question is why it would do that at the first place.

Well, to answer this question, we need to know that Fabric Interconnect is not a proper switch and there is no learning happened in between them over the L1 and L2 ports. It is just the Cluster Keep-Alive ports.

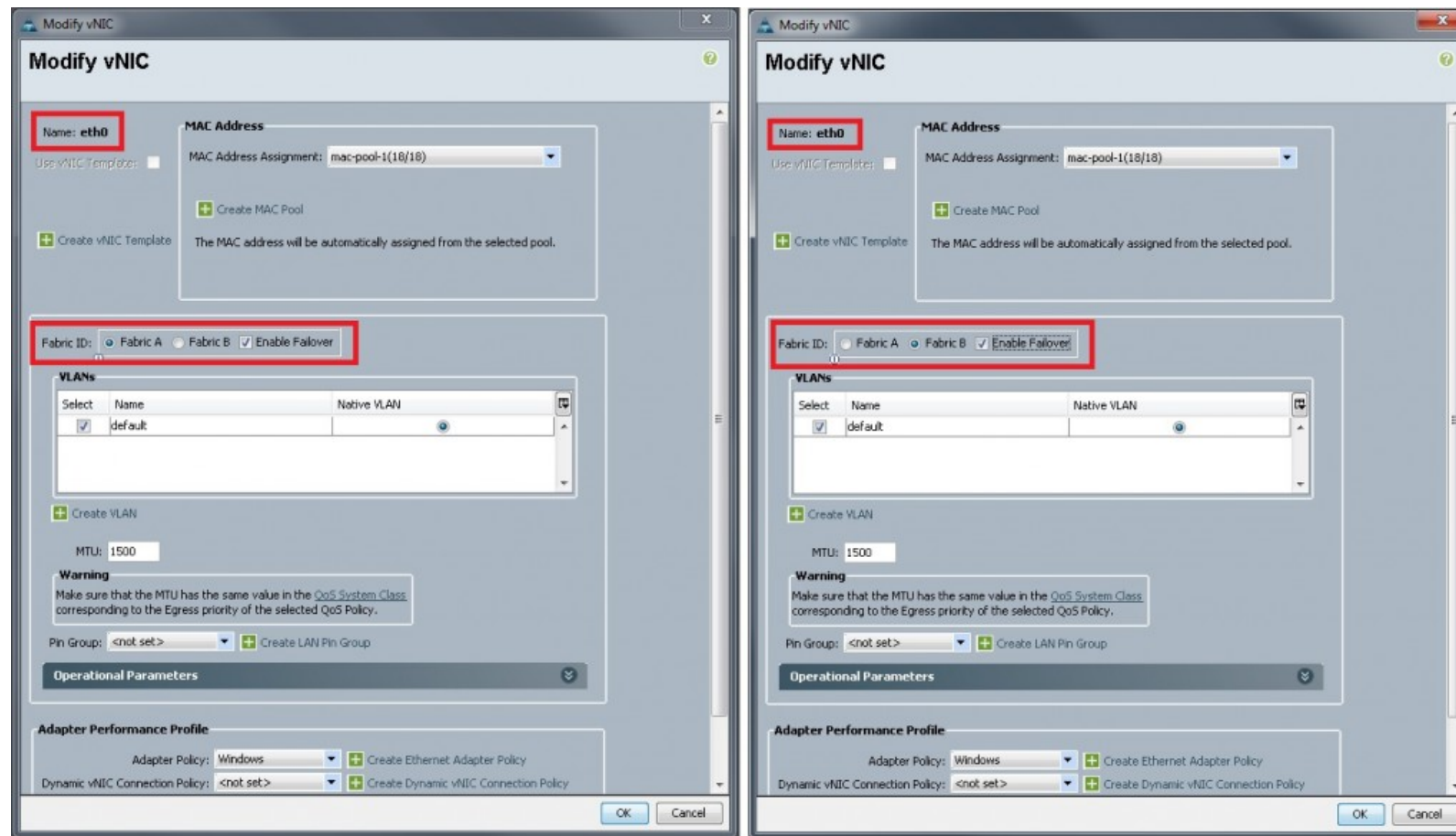
So, if Server A has two NIC pinned to two different Fabric (A and B) and Server B also has the same, while Server A would send traffic through both the Fabric, Server B would listen to only one side of it. That means Server A will send traffic through Fabric A and B but if Server B listens to only Fabric B then the entire traffic has to reach to the Core and then come back. If I want to visualize, it would be like below:



Now you can rectify this by creating Static Pinned group inside UCSM. In that way you can pinned the traffic to keep it inside the same Fabric, which will not go to core. However, that will defeat the resiliency part of it.

So, you can carefully choose now when to use OS based NIC teaming.

On the other hand in UCS Fabric Failover also, it may happen if you want to distribute the load among Fabric Interconnect. Let me show you what it may look like in this case.



In the above case also routing is going to happen, does not matter whether it is OS based or Fabric based failover configuration. It also end up having Performance Issues. So, in my opinion while you have to distribute the load, you also need to group the servers which need to talk to each other and place them onto same Fabric.

As per Cisco TAC they also observed that OS based failover over is slightly faster than the Fabric based failover.

So, in a nutshell, either the scenario you choose, you end up doing lot of pre-planning before you end up putting it all together.

"Cisco are you listening to this"