fineDoc Number: EDCS-<XXXXX>
Last Revision Date: December 9, 2010
Created by: Louis Watta
Template Ver. Number: EDCS-XXXX Rev X

CAE

UCS MAC-SYNC Feature in 1.4.1

“CISCO CONFIDENTIAL”

TABLE OF CONTENTS

1 Introduction

The purpose of this document is to provide a step by step sample configuration of the UCS MAC-SYNC feature functionality.

2 Test Requirements

- Software
  - UCS 1.4.1 image

- Hardware Requirements
  - UCS Test system with two 6100s
  - Upstream switch (Catalyst or Nexus)
  - A Service Profile with 2 NICs defined with Fabric Failover
  - ESX installed on above Service Profile with several VMs available
  - Ability to fail connections between the 6100s and the upstream switches

2.1 Description of the Problem

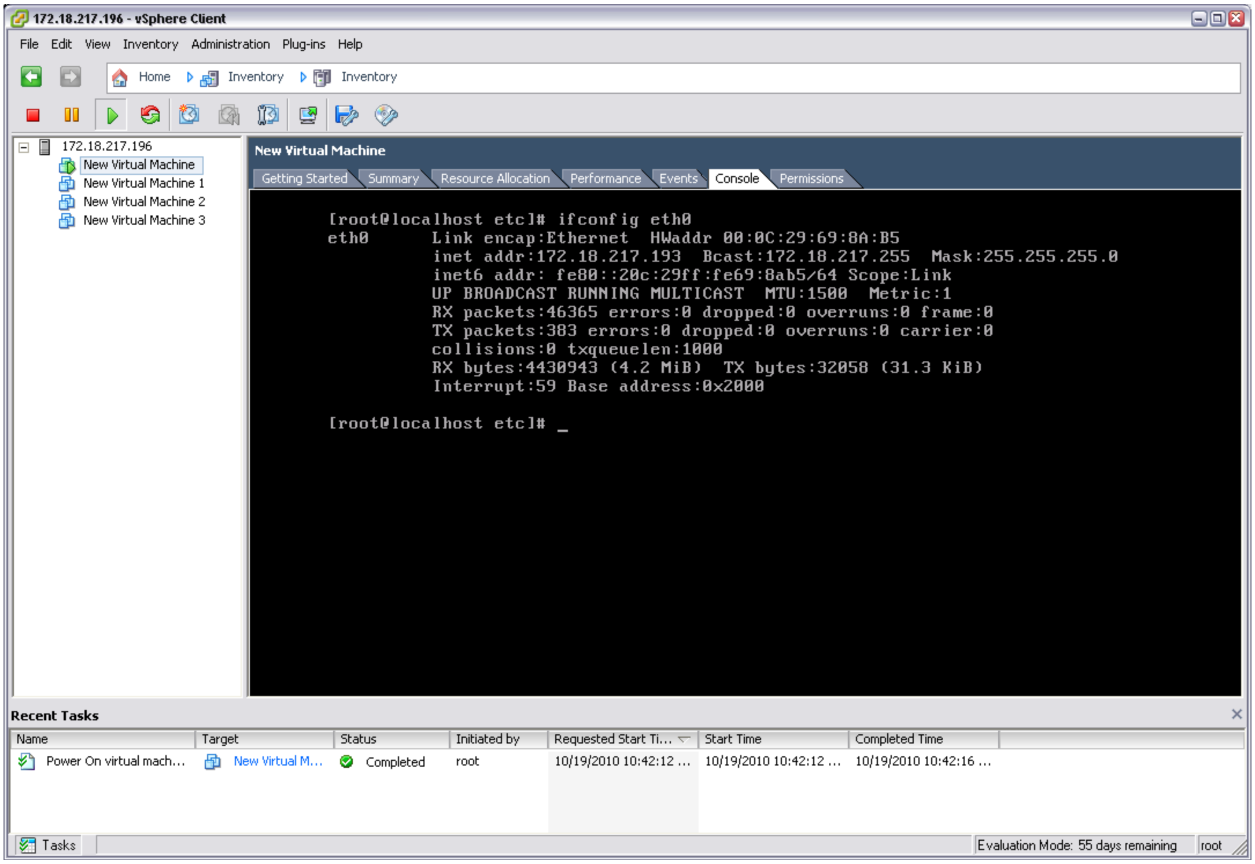
Currently UCS provides a hardware failover mechanism for NICs called hardware failover. This mechanism provides NIC redundancy for blades using the Qlogic and Emulex generation 1 cards and for the Cisco VIC card. The cards provide the ability to reroute traffic from one failed path to the other active path. This requires that VLAN connectivity be the same for each Fabric Interconnect (FI) but it can be a valuable tool in replacing software based NIC teaming at the OS level.

The problem today is that when there is an upstream network outage, the hardware does indeed failover to the alternate path, but assumes that enough traffic is passing from a VM that it will cause the MAC tables to get updated letting the switches know there is now a new path to the NIC. Currently the Fabric Interconnects to do not maintain a MAC table sync for Virtual Machines between themselves nor do they send a gratuitous ARP to upstream switches to notify them of a change in the topology.

To fix this problem MAC-SYNC function has been introduced in the 1.4.1 code of UCSM. Now each Fabric Interconnect notifies the other of their MAC tables. Only one FI will have the active MAC but by keeping a list of active MACs on both FIs it allows for fast failover. If an upstream failure is detected the MAC is made active on the other FI without needing to actually learn from the VM itself. The FI also now sends a GARP to the upstream switches to let them know about the change.

We can demonstrate the problem using current 1.3.1 code of UCSM, a single blade, ESX, and a VM.
We will power on a VM on a blade and then check MAC address tables on FIs and upstream switch

Here you can see the VM has a MAC ending in 8A:B5 and IP address 172.18.217.193



```
[root@localhost etc]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:69:8A:B5
          inet addr:172.18.217.193  Bcast:172.18.217.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe69:8ab5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46365 errors:0 dropped:0 overruns:0 frame:0
          TX packets:383 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4430943 (4.2 MiB)  TX bytes:32058 (31.3 KiB)
          Interrupt:59 Base address:0x2000

[root@localhost etc]# _
```

Name	Target	Status	Initiated by	Requested Start Ti...	Start Time	Completed Time
Power On virtual mach...	New Virtual M...	Completed	root	10/19/2010 10:42:12 ...	10/19/2010 10:42:12 ...	10/19/2010 10:42:16 ...

Show MAC address table on the FIs shows the following

```
CAE-UCS-RTP-A(nxos)# show mac-address-table vlan 1
VLAN      MAC Address      Type    Age       Port
---------+-----------------+-------+---------+------------------------------
1         000c.2969.8ab5   dynamic 80        veth757
1         0021.d8f5.0556   static  0         veth757
1         0050.5647.c776   dynamic 40        veth757

CAE-UCS-RTP-B(nxos)# show mac-address-table vlan 1
VLAN      MAC Address      Type    Age       Port
---------+-----------------+-------+---------+------------------------------
1         0021.d8f5.0557   static  0         veth755
```

You can see the MAC is active on FI A.

My upstream switch shows the following

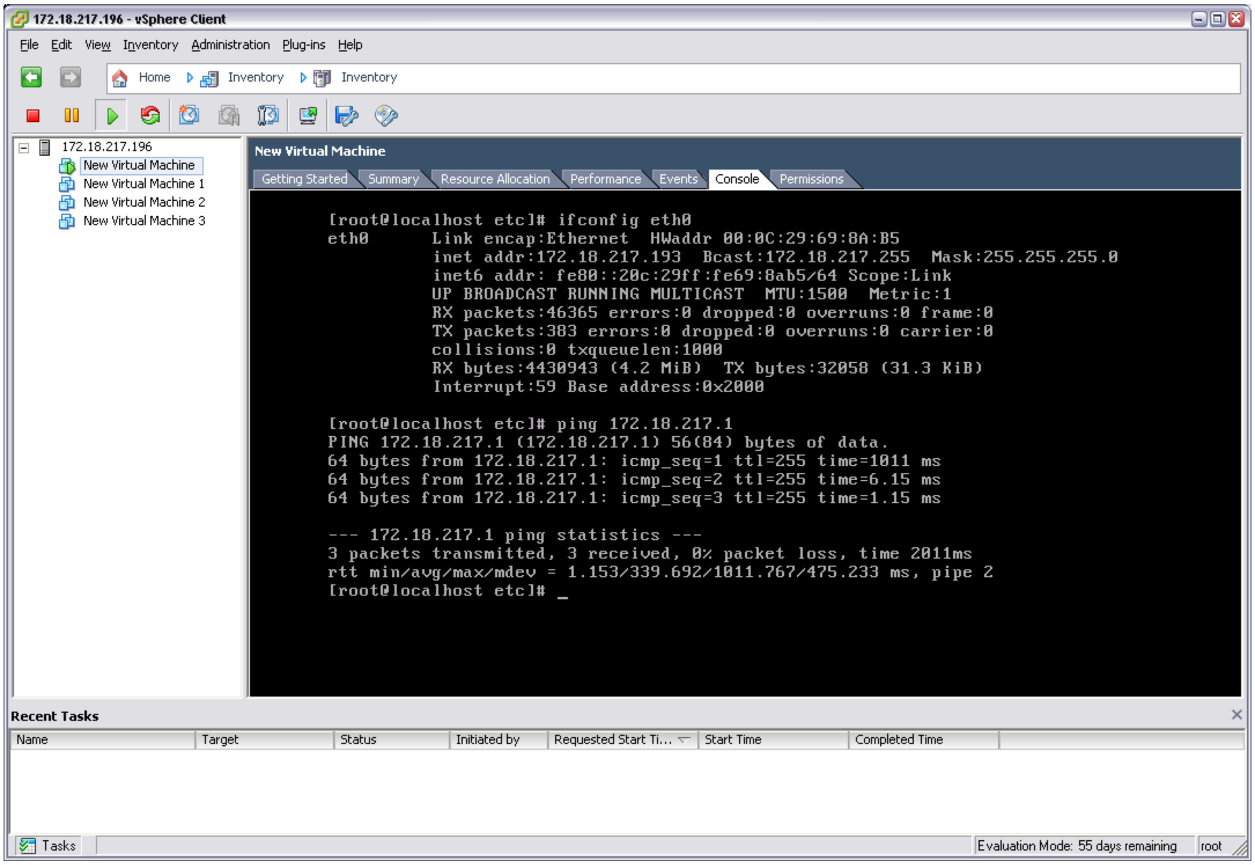
```
RTP-UCS-N5K-1# show mac address-table vlan 1 | include 8ab5
* 1         000c.2969.8ab5    dynamic   180         F     F   Eth1/21
```

Now fail the uplink on FI-A.
If I check the MAC tables on the FIs they look like

```
CAE-UCS-RTP-B(nxos)# show mac-address-table vlan 1
```

```
VLAN      MAC Address       Type    Age       Port
---------+-----------------+-------+---------+------------------------------
1         0021.d8f5.0556    static  0         veth758
1         0021.d8f5.0557    static  0         veth755
Total MAC Addresses: 2

CAE-UCS-RTP-A(nxos)# show mac-address-table vlan 1
Total MAC Addresses: 0

On the Upstream N5K
RTP-UCS-N5K-1# show mac address-table vlan 1 | include 8ab5
RTP-UCS-N5K-1#
```

You can see the MAC has not failed over. If we start traffic from the VM itself it will force an ARP request and populate the tables.



```
[root@localhost etc]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:69:8A:B5
          inet addr:172.18.217.193  Bcast:172.18.217.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe69:8ab5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46365 errors:0 dropped:0 overruns:0 frame:0
          TX packets:383 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4430943 (4.2 MiB)  TX bytes:32058 (31.3 KiB)
          Interrupt:59 Base address:0x2000

[root@localhost etc]# ping 172.18.217.1
PING 172.18.217.1 (172.18.217.1) 56(84) bytes of data.
64 bytes from 172.18.217.1: icmp_seq=1 ttl=255 time=1011 ms
64 bytes from 172.18.217.1: icmp_seq=2 ttl=255 time=6.15 ms
64 bytes from 172.18.217.1: icmp_seq=3 ttl=255 time=1.15 ms

--- 172.18.217.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2011ms
rtt min/avg/max/mdev = 1.153/339.692/1011.767/475.233 ms, pipe 2
[root@localhost etc]# _
```

Above you can see the first ping took a long time to succeed. If I check the MAC tables again we will see

```
CAE-UCS-RTP-B(nxos)# show mac-address-table vlan 1
VLAN      MAC Address       Type    Age       Port
---------+-----------------+-------+---------+------------------------------
1         000c.2969.8ab5    dynamic 0         veth758
1         0021.d8f5.0556    static  0         veth758
1         0021.d8f5.0557    static  0         veth755
1         0050.5647.c776    dynamic 0         veth758
Total MAC Addresses: 4
```

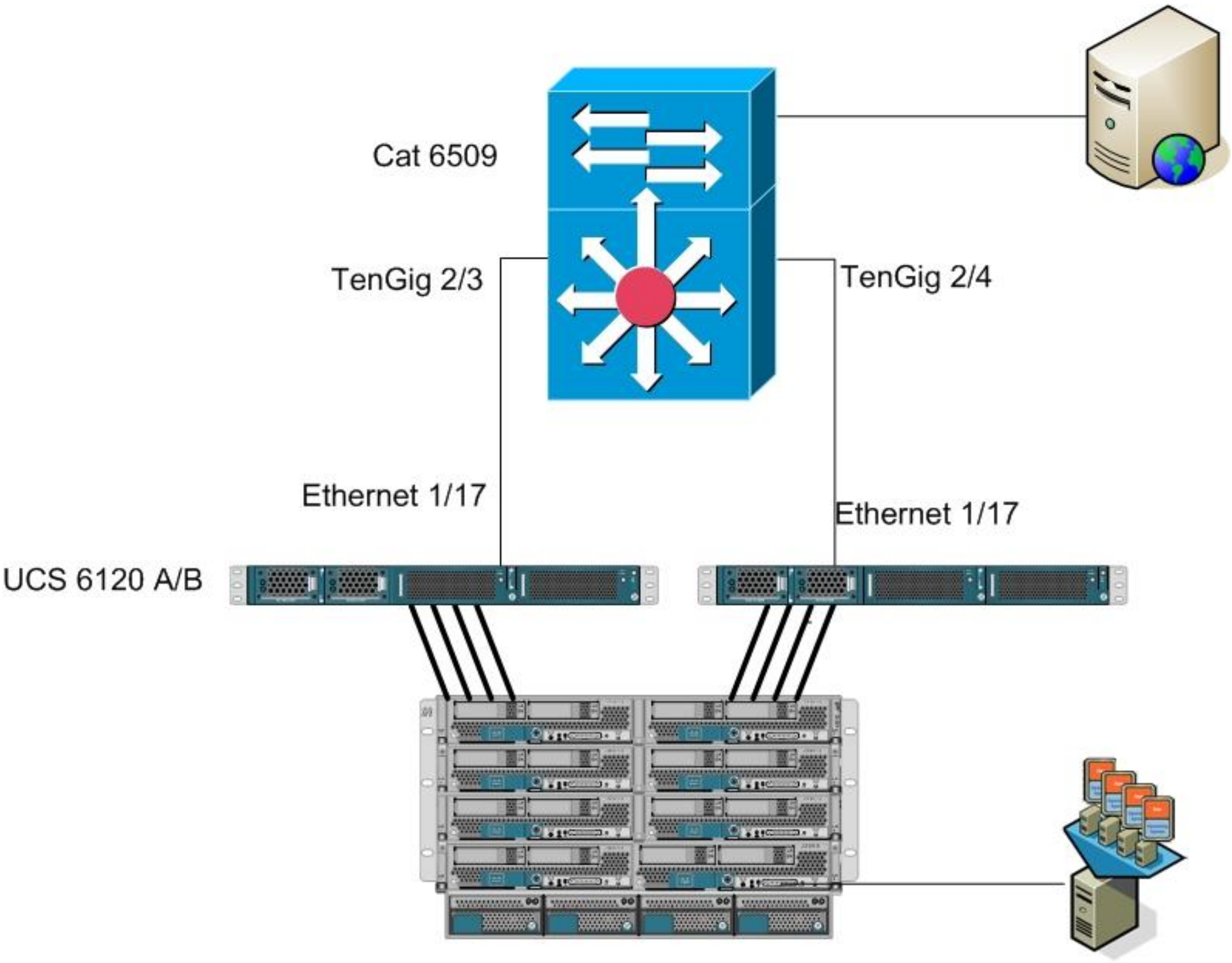
```
CAE-UCS-RTP-A(nxos)# show mac-address-table vlan 1
Total MAC Addresses: 0

RTP-UCS-N5K-1# show mac address-table vlan 1 | include 8ab5
* 1        000c.2969.8ab5    dynamic   10         F    F  Eth1/17
```

Now you can see that the MAC has been learned on the proper devices and traffic can flow again.

MAC-Sync prevents traffic loss from happening in the above scenario.

3 Test Cases

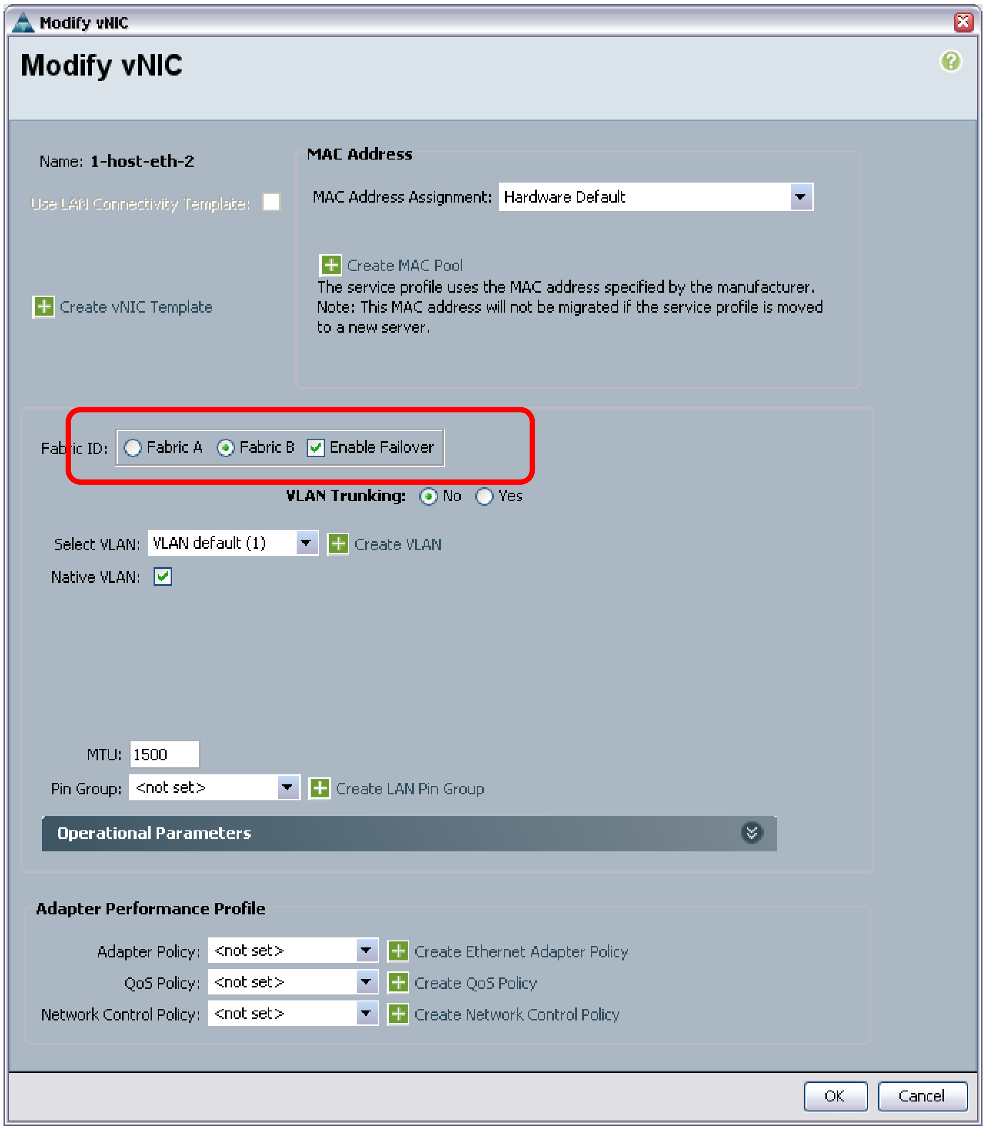
3.1 MAC-SYNC topology for testing



3.1.1 MAC-Sync Prerequisites

Configuration steps:

A service profile needs to be created and hardware NIC failover should be chosen. For the NICs in the SP make sure to create 2 vNICs. One for fabric A and the other for Fabric B.



Name: 1-host-eth-2

Use LAN Connectivity Template: ☐

MAC Address

MAC Address Assignment: Hardware Default

The service profile uses the MAC address specified by the manufacturer. Note: This MAC address will not be migrated if the service profile is moved to a new server.

Fabric ID: ○ Fabric A ● Fabric B ☑ Enable Failover

VLAN Trunking: ● No ○ Yes

Select VLAN: VLAN default (1)

Native VLAN: ☑

MTU: 1500

Pin Group: <not set>

Adapter Performance Profile

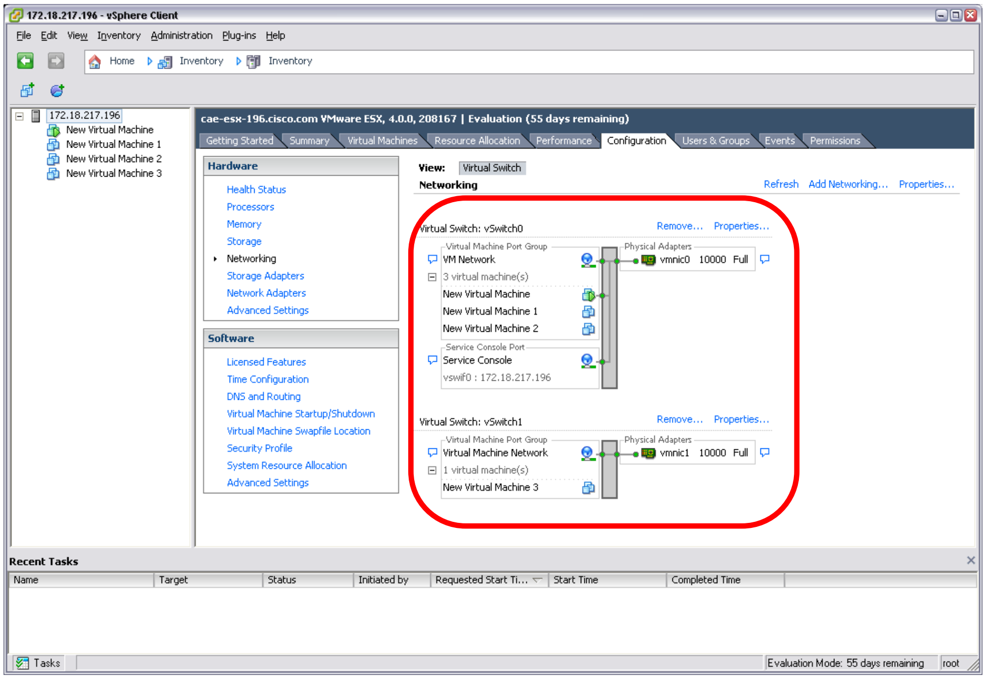
Adapter Policy: <not set>

QoS Policy: <not set>

Network Control Policy: <not set>

Install ESX on the blade and create several VMs. This test will use Linux based VMs. On the ESX host multiple vSwitches will be used but only add one link per vSwitch. This will force traffic for one vSwitch up fabric A and traffic for the other vSwitch up fabric B. We want to control where the VM traffic goes to observe the MAC-SYNC behavior.

Here you can see I have two vSwitches with only one link per vSwitch



.

MAC-SYNC option is on by default. There is nothing to do in the UCSM GUI or the CLI to enable. As long as 1.4.1 is installed MAC-SYNC should be working.

Verify Steps:

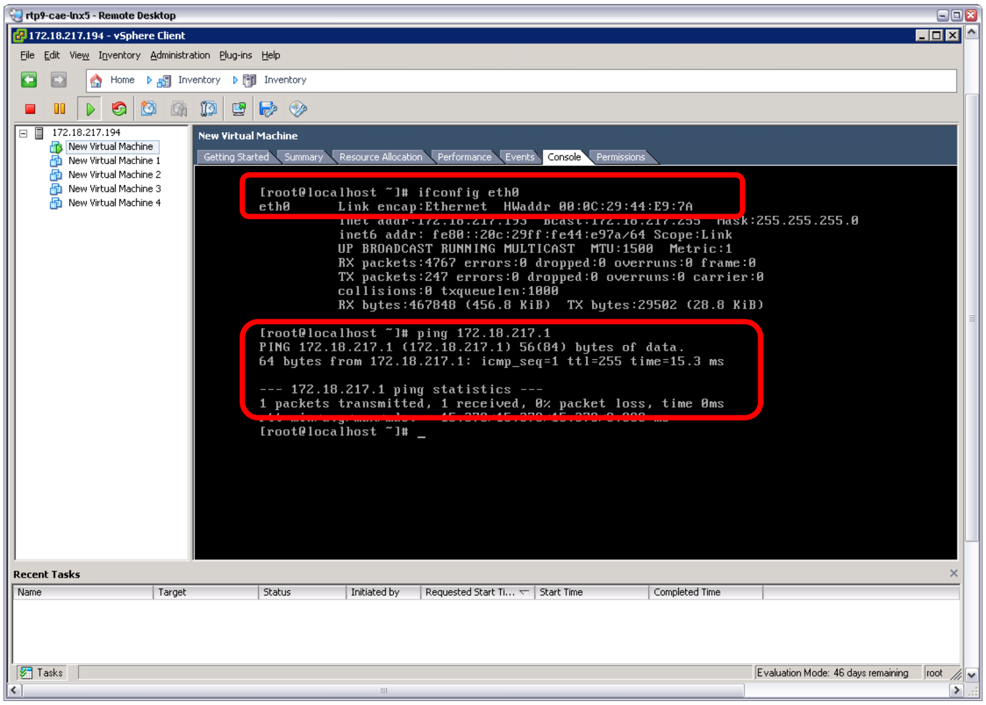
If you want to verify that MAC-Sync is active you can run the following commands on the FI. Check and make sure that CFS state is enabled, hand-shake is done and peer state is up.

```
rtp-6100-2-A(nxos)# show platform fwm info replmac
Current CFS state: enabled(yes) hand-shake(done) peer state(up)
Mac replication DB:
  Mac: 000c.2944.e97a Vlan 1 Flags local     Age: 0 Vnic id: SAM Fixed - vnic uuid: 000000
000000031b
  Mac: 0050.564e.a17f Vlan 1 Flags  remote    Age: 0 Vnic id: SAM Fixed - vnic uuid: 00000
0000000031d
```

3.1.2 Verify MAC-Sync works

Configuration steps:

From a VM on the ESX host ping out to the gateway and ping from outside the gateway to the VM to verify that networking is working.



```
[root@localhost ~]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0C:29:44:E9:7A
          inet addr:172.18.217.193  Bcast:172.18.217.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe44:e97a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4767 errors:0 dropped:0 overruns:0 frame:0
          TX packets:247 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:467848 (456.8 KiB)  TX bytes:29502 (28.8 KiB)

[root@localhost ~]# ping 172.18.217.1
PING 172.18.217.1 (172.18.217.1) 56(84) bytes of data.
64 bytes from 172.18.217.1: icmp_seq=1 ttl=255 time=15.3 ms

--- 172.18.217.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 15.378/15.378/15.378/0.000 ms
[root@localhost ~]# _
```

Here you can see the VM can ping out and the MAC address ends in E9:7A

Verification steps:

Check each Fabric Interconnect MAC address table.

```
rtp-6100-2-A(nxos)# show mac address-table vlan 1
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1          000c.2944.e97a    dynamic   0          F    F   veth795
* 1          0026.510b.78d7    static    0          F    F   veth795


rtp-6100-2-B(nxos)# show mac address-table vlan 1
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1          0026.510b.78d6    static    0          F    F   veth797
* 1          0050.564e.a17f    dynamic   20         F    F   veth797
```

Here you can see the MAC of the VM is active on FI-A. Let’s check the upstream switch as well.

```
cae-cat6k-1#show mac-address-table vlan 1 | include e97a
*    1  000c.2944.e97a   dynamic  Yes          0   Te2/3
```

You can see it is active on Te2/3

Fail the upstream connection on FI-A. Once you have failed the connection check the MAC tables on the FIs

```
rtp-6100-2-A(nxos)# show mac address-table vlan 1
rtp-6100-2-A(nxos)#
```

No MAC addresses are now present on FI-A

```
rtp-6100-2-B(nxos)# show mac address-table vlan 1
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1         000c.2944.e97a    dynamic   0          F    F  veth796
* 1         0026.510b.78d6    static    0          F    F  veth797
* 1         0026.510b.78d7    static    0          F    F  veth796
* 1         0050.564e.a17f    dynamic   0          F    F  veth797
```

Here you can see that FI-B immediately has the MAC address for the VM with address ending with .e97a

The upstream CAT shows

```
cae-cat6k-1#show mac-address-table vlan 1 | include e97a
*    1  000c.2944.e97a   dynamic  Yes          0   Te2/4
```

You can see that the CAT correctly was notified that the MAC was moved.

Now you can fail back the link and watch the MAC tables

```
rtp-6100-2-A(nxos)# show mac address-table vlan 1
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1         000c.2944.e97a    dynamic   0          F    F  veth795
* 1         0026.510b.78d7    static    0          F    F  veth795
```

```
rtp-6100-2-B(nxos)# show mac address-table vlan 1
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link
   VLAN     MAC Address      Type      age     Secure NTFY    Ports
---------+-----------------+--------+---------+------+----+------------------
* 1         0026.510b.78d6    static    0          F    F  veth797
* 1        0050.564e.a17f    dynamic   50         F    F  veth797
```

You can see that the MAC moved back to FI-A and on the CAT output below it was correctly notified that the MAC was moved back to FI-A

```
cae-cat6k-1#show mac-address-table vlan 1 | include e97a
*    1  000c.2944.e97a   dynamic  Yes          0   Te2/3
```