Nexus 7000 F2/F2e: Understand and Mitigate MAC Table Full Issues

Contents

Introduction
Prerequisites
Requirements
Components Used
Background Information
Mitigation Steps
Option 1. Prune Vlans
Option 2. L3 Separation
Option 3. Alternate Design Architecture like Fabricpath
Option 4. Use High Capacity Line Cards like M2/F3 Card

Introduction

This document describes F2/F2e MAC table full condition and methods to mitigate it.

F2 module with 16k MAC limit per Switch on Chip (SoC) reports random MAC table to be of full error messages at 60% utilization. Why is the line card not capable in utilizing the entire 16k MAC table space that is available?

```
%L2MCAST-SLOT2-2-L2MCAST_MAC_FULL_LC: Failed to insert entry in MAC table for FE 1 swidx 271
(0x10f) with err (mac table full). To avoid possible multicast traffic loss, disable OMF. Use
the configuration CLI: "no ip igmp snooping optimise-multicast-flood"
```

Prerequisites

Requirements

Cisco recommends that you have knowledge of the Nexus 7000 architecture.

Components Used

The information in this document is based on these software and hardware versions:

- Nexus 7000 with Release 6.2.10 and later.
- F2e series line card.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Background Information

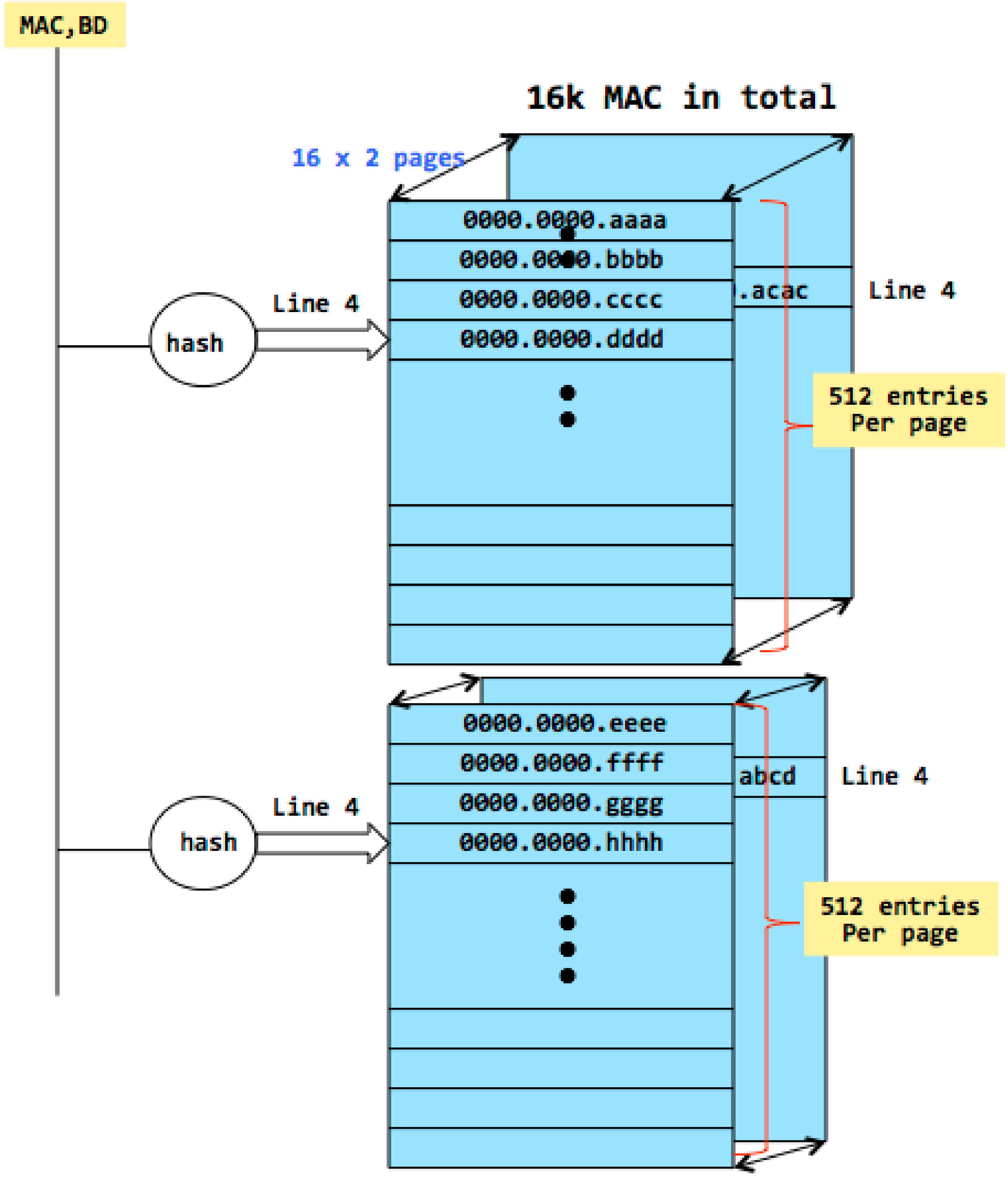
The F2 module has 16k MAC table space per SoC of forwarding engine.

There are 12 such SoC on each module and each services 4 ports each.

```
module-1# show hardware internal forwarding f2 l2 table utilization instance all
L2 Forwarding Resources
-------------------------
 L2 entries: Module inst   total    used   mcast   ucast   lines   lines_full
------------------------------------------------------------------------------
             1         0   16384    9647     265    9382     512            0
             1         1   16384    7430       1    7429     512            0
             1         2   16384    9654     264    9390     512            0
             1         3   16384    7430       7    7423     512            0
             1         4   16384    7564       8    7556     512            0
             1         5   16384    7432       1    7431     512            0
             1         6   16384    7418       0    7418     512            0
             1         7   16384     558       0     558     512            0
             1         8   16384     558       0     558     512            0
             1         9   16384     558       0     558     512            0
             1        10   16384     558       0     558     512            0
             1        11   16384    7416       0    7416     512            0
```

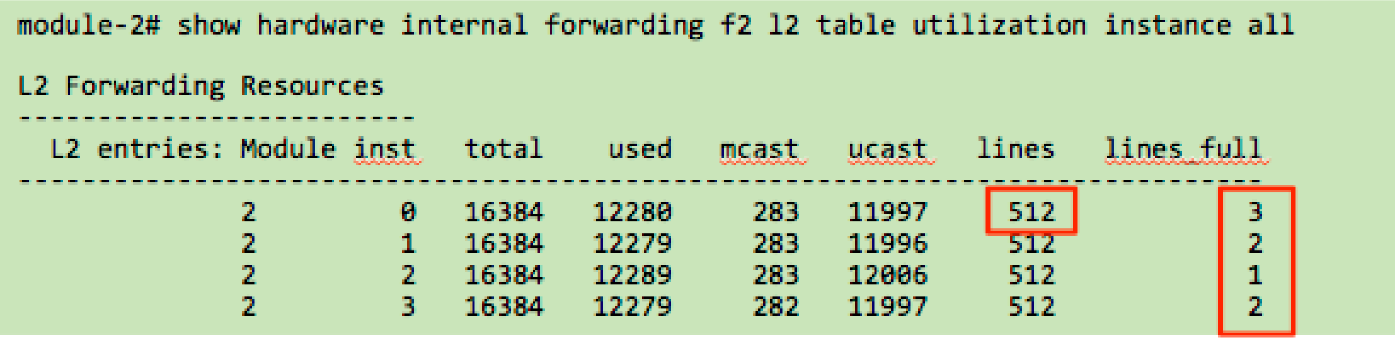
The output here highlights the usage of the hardware MAC address table per SoC.

In order to understand why you get the MAC table full message, you need to understand how the MAC table is divided. This diagram helps you with a visual clarity.



- MAC table which is 16k for F2 line card is distributed into pages. Each page can hold 512 entries. So, you have a total of 32 pages. You can use a 2 way hash in order to place a new MAC into one of the pages.
- Now, lets take a scenario where line 4 is used on every page. What this means is that that 32 unique MACs have ended up with a hashing output that puts it on the same line of every page.
- If a 33rd MAC with the same hash output is generated then you cannot install it and can likely see the error message that is shown earlier.
- The lines full column tracks the number of lines that have reached this state.

The output here also shows the lines per page and also if the lines full condition has reached.



```
module-2# show hardware internal forwarding f2 l2 table utilization instance all

L2 Forwarding Resources
-------------------------
  L2 entries: Module inst   total    used   mcast   ucast   lines    lines full
-------------------------------------------------------------------------------
              2         0   16384   12280     283   11997     512             3
              2         1   16384   12279     283   11996     512             2
              2         2   16384   12289     283   12006     512             1
              2         3   16384   12279     282   11997     512             2
```

Only MAC address that end up hashing a particular way encounters this condition, while you will not see any issues for other MAC address.

Typically, multicast MAC address can see this more often as they are not as randomized as the unicast MACs. The Line cards are usually tested with industry standard RFC testing in order to validate the utilization efficiency. However, there is always a likelihood of certain MAC combinations in specific customer environments to not be very well optimized which leads to this error.

Mitigation Steps

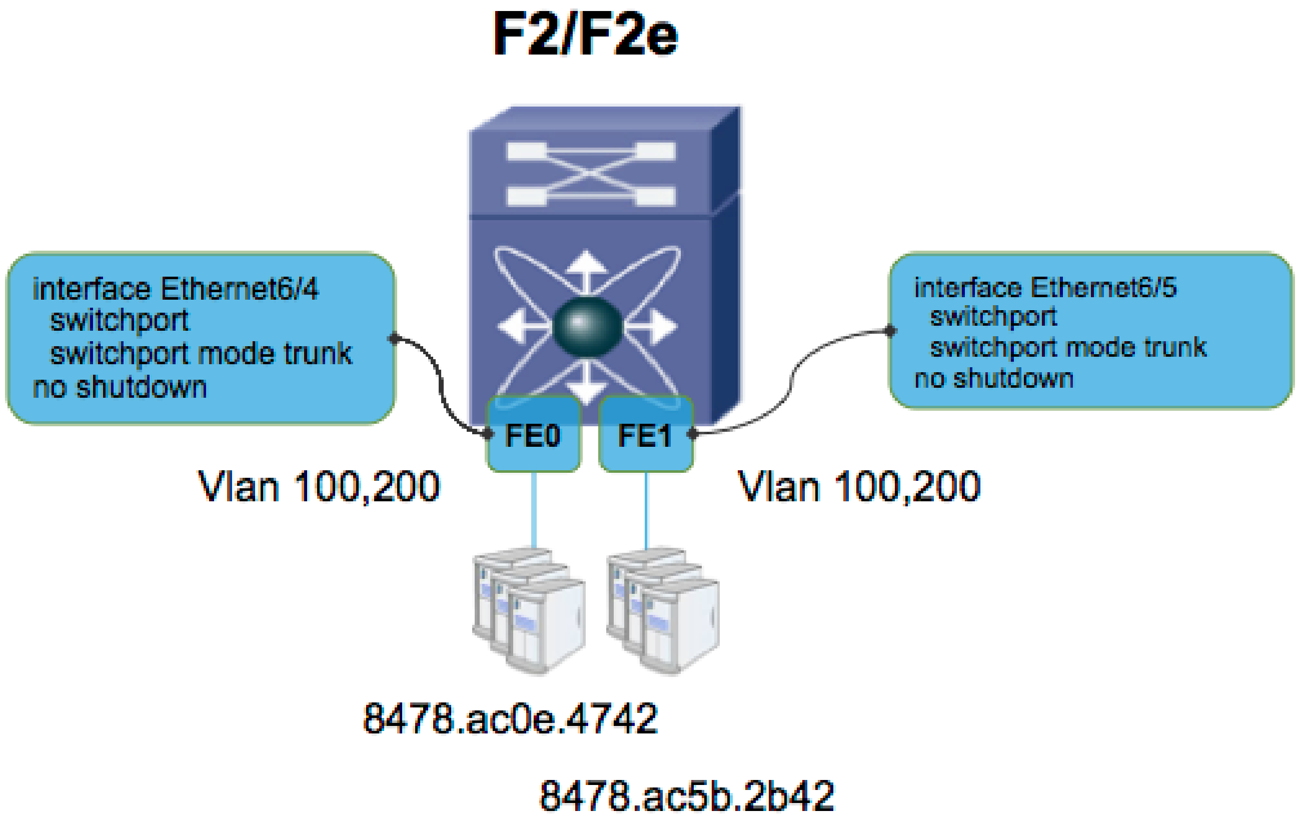
These steps can help to reduce the MAC table usage.

- Prune vlans
- L3 separation
- Other design option (fabricpath)
- M2 or F3 module for future growth

Option 1. Prune Vlans

> **Note**: There are no SVI for vlan 100 and 200. This is an important assumption and it will become clear when you read option 2.

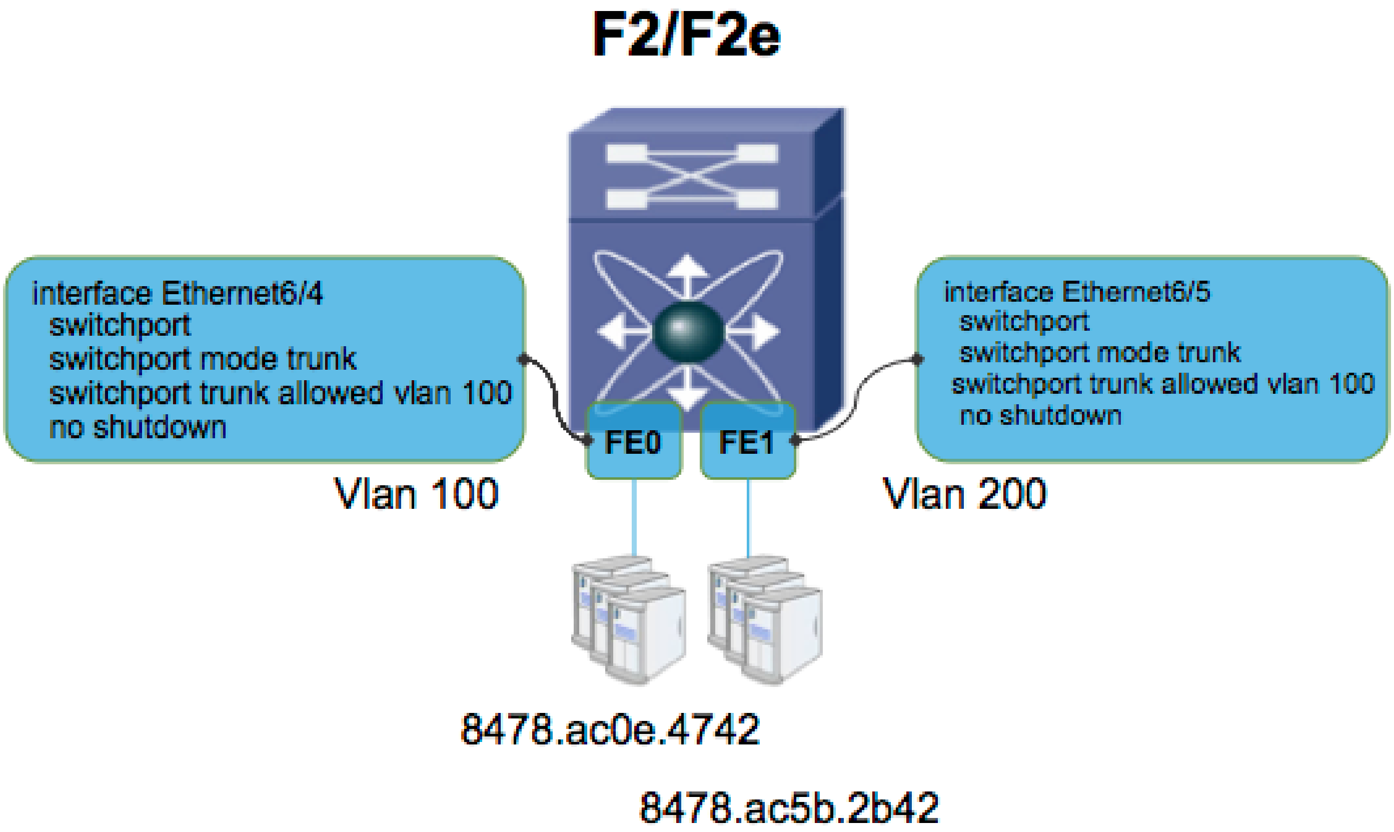
In this simplified setup, there are two host on different SoCs.



```
N7KA-VDC-1(config-vlan)# sh mac address-table  <snip>
 Note: MAC table entries displayed are getting read from software.
 Use the 'hardware-age' keyword to get information related to 'Age'

 Legend:
       * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
       age - seconds since last seen,+ - primary entry using vPC Peer-Link,
       (T) - True, (F) - False ,  ~~~ - use 'hardware-age' keyword to retrieve age info
  VLAN     MAC Address      Type      age     Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
* 100      8478.ac0e.4742    dynamic     ~~~      F    F  Eth6/4
* 200      8478.ac5b.2b42    dynamic     ~~~      F    F  Eth6/5

N7KA-VDC-1# sh vlan internal bd-info vlan-to-bd 100
VDC Id  Vlan Id  BD Id
------  -------  -------
1        100      38

N7KA-VDC-1# sh vlan internal bd-info vlan-to-bd 200
VDC Id  Vlan Id  BD Id
------  -------  -------
1        200      39

N7KA-VDC-1(config-if-range)# sh hard mac address-table 6  <SNIP>
FE | Valid| PI|  BD  |      MAC       |  Index| Stat| SW   | Modi| Age| Tmr|
  |      |   |      |                |       | ic |      | fied|Byte| Sel|
---+------+---+------+---------------+-------+-----+-----+-----+----+----+-
0    1     1   38    8478.ac0e.4742  0x00053   0   0x081   1    138   1
0    1     0   39    8478.ac5b.2b42  0x00054   0   0x091   1    138   1
 1    1     0   38    8478.ac0e.4742  0x00053   0   0x091   1    138   1
 1    1     1   39    8478.ac5b.2b42  0x00054   0   0x081   1    138   1
```

Each FE (forwarding engine = SoC) shows 2 MAC address in use.

Now, you prune the vlans and the config is as shown in this image.



```
interface Ethernet6/4
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 100
  no shutdown
```

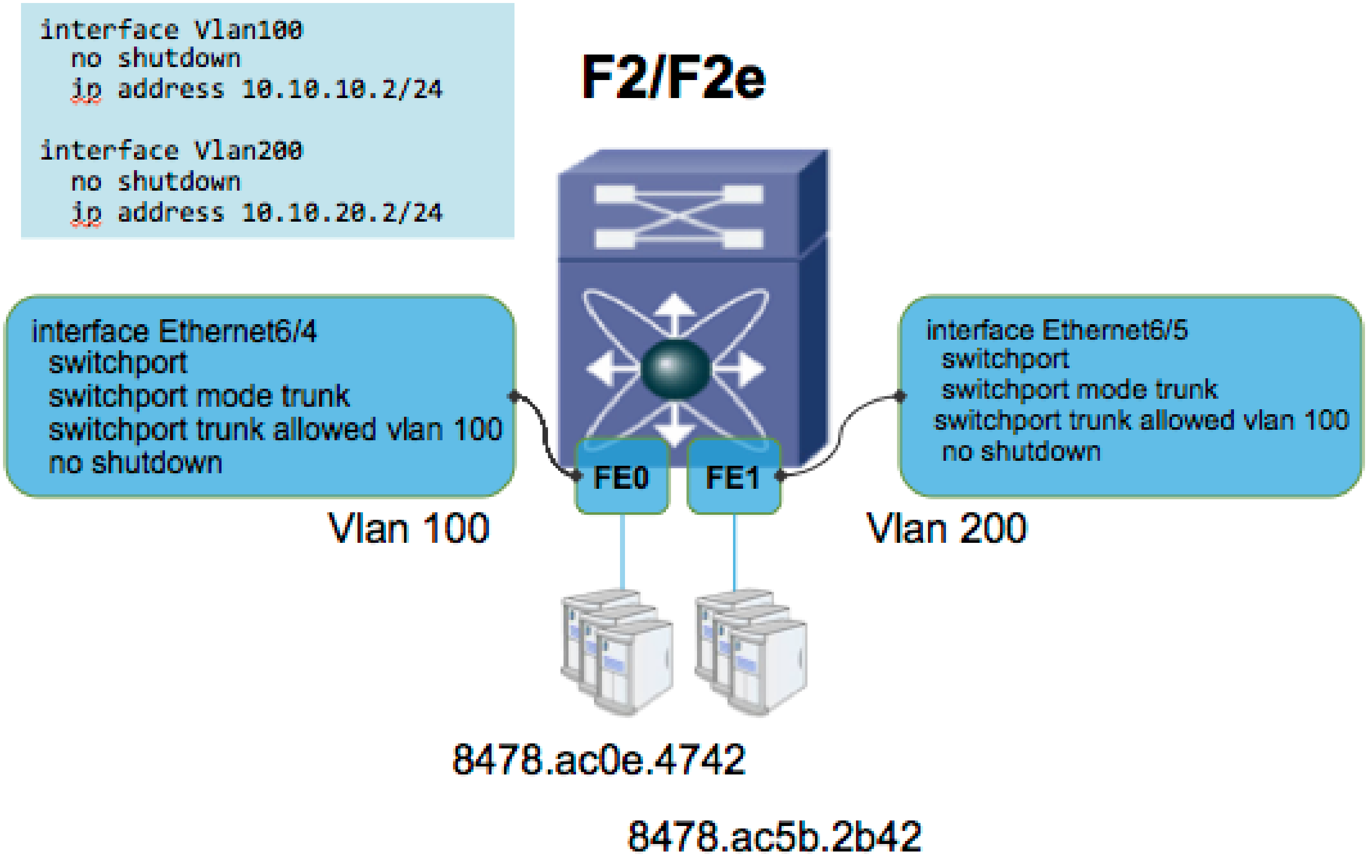
```
interface Ethernet6/5
  switchport
  switchport mode trunk
 switchport trunk allowed vlan 100
  no shutdown
```

After you prune the vlans, you have one less entry per FE (SoC). Pruning the vlan prevented a sync between the FE for MAC address.

```
N7KA-VDC-1(config-if-range)# sh hard mac address-table 6  <SNIP>
FE | Valid| PI|  BD  |      MAC      |  Index| Stat| SW  | Modi| Age| Tmr|
  |      |   |      |               |       | ic |     | fied|Byte| Sel|
---+------+---+------+---------------+-------+-----+-----+-----+----+----+-
0    1     1   38    8478.ac0e.4742  0x00053   0   0x081   1    138   1
 1    1     1   39    8478.ac5b.2b42  0x00054   0   0x081   1    138   1
```

Option 2. L3 Separation

Here, you have the vlans pruned but assume you have the Switch Virtual Interface (SVI) configured on this VDC for both vlan 100 and 200.



```
interface Vlan100
  no shutdown
  ip address 10.10.10.2/24

interface Vlan200
  no shutdown
  ip address 10.10.20.2/24
```

```
interface Ethernet6/4
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 100
  no shutdown
```

```
interface Ethernet6/5
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 100
  no shutdown
```

The MAC table will look like this, where MAC address is synced between FEs even though the vlan is pruned. This is because Switch Virtual Interface (SVI) is enabled which requires the FE to know about MAC addresses from other vlans as well.

```
N7KA-VDC-1(config-if-range)# sh hard mac address-table 6  <SNIP>
FE | Valid| PI|  BD  |      MAC      |  Index| Stat| SW   | Modi| Age| Tmr|
  |      |   |      |               |       | ic  |      | fied|Byte| Sel|
---+------+---+------+---------------+-------+-----+-----+-----+----+----+-
0    1     1   38    8478.ac0e.4742  0x00053   0   0x081   1    138   1
0    1     0   39    8478.ac5b.2b42  0x00054   0   0x091   1    138   1
 1    1     0   38    8478.ac0e.4742  0x00053   0   0x091   1    138   1
 1    1     1   39    8478.ac5b.2b42  0x00054   0   0x081   1    138   1
```

If you remove vlan 200 SVI, then the MAC table does not see a sync for vlan 200 mac on FE0.

```
N7KA-VDC-1(config-if-range)# sh hard mac address-table 6  <SNIP>
FE | Valid| PI|  BD  |      MAC      |  Index| Stat| SW   | Modi| Age| Tmr|
  |      |   |      |               |       | ic  |      | fied|Byte| Sel|
---+------+---+------+---------------+-------+-----+-----+-----+----+----+-
0    1     1   38    8478.ac0e.4742  0x00053   0   0x081   1    138   1
 1    1     0   38    8478.ac0e.4742  0x00053   0   0x091   1    138   1
 1    1     1   39    8478.ac5b.2b42  0x00054   0   0x081   1    138   1
```

The conclusion of the step is not to delete the SVIs but to analyze if moving the SVIs to a different VDC by creating a separate layer 3 VDC is an option. This is not a easy design step and would require detailed planning.

Option 3. Alternate Design Architecture like Fabricpath

These are more complex alternatives which is beyond the scope of this document to detail out but it can provide efficiencies in the MAC usage.

Option 4. Use High Capacity Line Cards like M2/F3 Card

The M2 and F3 line card has much higher MAC table capacity.

M2 Data sheet ==> MAC table (128k per SoC)

F3 Data sheet ==> MAC Table (64k per SoC)