FC-NVMe – Deployment Considerations and Best Practices

This document provides deployment considerations and best practices for FC-NVMe with Dell EMC Connectrix B-Series (Brocade) and Connectrix MDS Series (Cisco).

November 2019
**Revisions**

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2019</td>
<td>Initial release</td>
</tr>
</tbody>
</table>

**Acknowledgements**

This paper was produced by the following members of the Dell EMC storage engineering team:

Alan Rajapa  
Erik Smith  
Jai Bansal

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying and distribution of any software described in this publication requires an applicable software license.

©Published November 2019: Dell Inc. or its subsidiaries. All Rights Reserved. Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners.

Dell believes the information in this document is accurate as of its publication date. The information is subject to change without notice.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>REVISIONS</td>
<td>2</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>2</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>4</td>
</tr>
<tr>
<td>WHEN TO USE FC-NVME?</td>
<td>5</td>
</tr>
<tr>
<td>FC-NVME DEPLOYMENT OPTIONS</td>
<td>6</td>
</tr>
<tr>
<td>Options 1 and 2 – Low latency not required</td>
<td>6</td>
</tr>
<tr>
<td>Option 3 – Connectivity needs require a multi-switch fabric – Low latency is required</td>
<td>7</td>
</tr>
<tr>
<td>Option 4 – Connectivity needs met by a single switch fabric – Lowest latency is required</td>
<td>7</td>
</tr>
<tr>
<td>FC-NVME BEST PRACTICES</td>
<td>7</td>
</tr>
<tr>
<td>FC-NVME CASE STUDY</td>
<td>7</td>
</tr>
<tr>
<td>Overview</td>
<td>8</td>
</tr>
<tr>
<td>Brocade</td>
<td>8</td>
</tr>
<tr>
<td>Pre-Requisites</td>
<td>8</td>
</tr>
<tr>
<td>Starting Topology</td>
<td>8</td>
</tr>
<tr>
<td>Target Topology</td>
<td>9</td>
</tr>
<tr>
<td>Detail Configuration steps</td>
<td>9</td>
</tr>
<tr>
<td>Creating a logical Switch</td>
<td>10</td>
</tr>
<tr>
<td>Check Point</td>
<td>11</td>
</tr>
<tr>
<td>Cisco</td>
<td>12</td>
</tr>
<tr>
<td>Pre-Requisites</td>
<td>12</td>
</tr>
<tr>
<td>Starting Topology</td>
<td>13</td>
</tr>
<tr>
<td>Target Topology</td>
<td>13</td>
</tr>
<tr>
<td>Detail Configuration steps</td>
<td>13</td>
</tr>
<tr>
<td>Creating new VSANs &amp; assign ports</td>
<td>13</td>
</tr>
<tr>
<td>Check Point</td>
<td>14</td>
</tr>
<tr>
<td>Dedicate ISL links</td>
<td>15</td>
</tr>
</tbody>
</table>
Summary

By now, many of you will have heard about the NVMe protocol. In fact, some of you might even have NVMe-based drives in your laptop or are perhaps using NVMe drives on the servers in your datacenter and/or using storage arrays with NVMe drives. If this is the case, you’re probably doing so because you’ve heard of the potential boost in performance that can be realized when using NVMe-based drives, especially when these drives use so called next generation storage media (e.g., Intel Optane based running). These devices promise latencies in the 10usec range, which is approximately 1/100th the latency of an SAS or SATA based SSD. Although the storage media itself is largely responsible for this latency improvement, the NVMe protocol plays a part because it allows IO Parallelism and because the protocol is relatively lightweight when compared to SCSI.

Dell EMC has been using NVMe-based drives inside PowerMAX for well over 2 years. We can confirm that we see a significant improvement in CPU utilization and latency when NVMe drives are used. In other words, given the same footprint, NVMe allows us to deliver more throughput to our end users.

**FC-NVMe** is different from NVMe. It was designed to allow users to realize the same CPU and latency improvements when they access external NVMe drives (e.g., such as those located in a storage array). However, to realize these benefits, the latency across the fabric Transport (FC, TCP, RoCE, IB) must remain as low as possible (i.e., 10usec Round Trip Time) in order not to mask the latency benefits of the drive itself.

Keeping the fabric latency low does not just result in a theoretical improvement. When uncongested FC-NVMe is compared to uncongested traditional FC (labeled FC-SQ in the charts below), FC-NVMe decreases latency by 10% and allows for almost ten times the IOPs.

However, before you decide to transition to NVMe, please keep in mind that the improvements observed in the transport layer (shown in the charts below) do not mean that you will also observe the same improvement from the application layer (i.e., your applications will NOT be up to 10 times faster). This is because, although the NVMe protocol over a FC transport is much more efficient than traditional FC, transport latency makes up for only a fraction of the total latency of the end-to-end data path. End devices that you use in the real world, along with their internal system latencies, will have a huge impact on the actual benefit that you will be able to observe at the application layer. In fact, the biggest benefit we have observed at the application layer was a 25% reduction in CPU utilization and a slight reduction in latency (~10%).
When to use FC-NVMe?

Given the modest benefits that your existing applications are likely to see with FC-NVMe, perhaps you are starting to wonder if moving to a new protocol is worth the effort?
In the long-term, as newer applications are deployed, especially ones that make use of in-memory databases, the answer is almost certainly going to be "yes". In the short-term, not only deciding if you should deploy FC-NVMe, but also deciding how it should be deployed, will depend heavily on the needs of the environment. You might want to ask yourself the following questions:

- Do my applications require lower latency access to storage?
- Can a single switch meet my connectivity needs?

Once you can answer these questions, the following table can be used to determine where NVMe over fabrics fits into today's infrastructure.

**FC-NVMe Deployment Options**

<table>
<thead>
<tr>
<th>Connectivity needs exceed single switch?</th>
<th>Low latency required?</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>Y</td>
<td>Option 1 Good candidate for running FC-NVMe &amp; Traditional FC on multi-switch SAN</td>
</tr>
<tr>
<td>N</td>
<td>N</td>
<td>Option 2 Good candidate for running FC-NVMe &amp; Traditional FC on single switch SAN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Option 3 Requires non-oversubscribed multi-switch SAN or isolated subset (VSAN/VF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Option 4 Best option: Dedicated Single Switch for FC-NVMe Devices</td>
</tr>
</tbody>
</table>

**Options 1 and 2 – Low latency not required.**

If you are thinking about trying FC-NVMe and you do not need to take full advantage of its ability to provide lower latency access to storage, you can simply go ahead and use a traditional SAN design. This includes running concurrent FC-NVMe and FCP traffic in the same SAN. In this type of configuration, the FC-NVMe traffic will be treated exactly like the traditional FC traffic is and will be susceptible to the effects of congestion spreading if you have slow drain devices or a bandwidth mismatch somewhere in the SAN (especially in the multi-switch SAN).

For more information, refer to the FC-NVMe case study below.
Option 3 – Connectivity needs require a multi-switch fabric – Low latency is required

Although this type of configuration does not provide the lowest possible transport latency under all operating conditions (for the lowest refer to Option 4), if you plan correctly, you can still use a multi-switch SAN for NVMe traffic and get lower latency access to storage.

The trick here is to isolate the FC-NVMe traffic from other types of traffic. You do this either by logically isolating the traffic (putting it onto a dedicated VSAN/Virtual Fabric) or physically isolating it by attaching the FC-NVMe interfaces to its own dedicated multi-switch SAN.

For more information, refer to the Fibre Channel SAN Topologies Techbook which can be found in the Dell EMC Topology Resource Center: [https://elabnavigator.emc.com/eln/topologyResource](https://elabnavigator.emc.com/eln/topologyResource)

Option 4 – Connectivity needs met by a single switch fabric – Lowest latency is required

To ensure that you can achieve the lowest latency possible, we recommend isolating the NVMe traffic and keeping it local (within the same switch). Keeping the NVMe traffic local reduces fabric latency to a minimum (each switch can add 5usec of Round Trip latency) and eliminates many of the conditions that can lead to congestion spreading.

For more information, refer to the Fibre Channel SAN Topologies Techbook which can be found in the Dell EMC Topology Resource Center: [https://elabnavigator.emc.com/eln/topologyResource](https://elabnavigator.emc.com/eln/topologyResource)

**FC-NVMe Best Practices**

The four deployment options provided above are based on the following FC-NVMe Best Practices. They’re intended to help you get the most out of the FC-NVMe protocol.

1. To ensure maximum performance / minimum latency, consider the following:
   - You should avoid ISLs because each switch will add approximately 5us of Round-Trip-Time latency.
   - If ISLs cannot be avoided, then use non-over-subscribed ISLs.
   - FCR, IVR and FCIP/distance extension are not recommended to avoid additional latency.
   - To reduce the possibility of congestion spreading and maximize ease of management, connect hosts and storage port pairs to the same switch where possible.
   - To learn more about “Congestion Spreading”, refer to the Whitepaper Connectrix - Congestion Spreading and how to avoid it [https://www.dell EMC.com/resources/en-us/asset/white-papers/products/networking/h17762-connectrix-congestion-spreading-and-how-to-avoid-it-wp.pdf](https://www.dell EMC.com/resources/en-us/asset/white-papers/products/networking/h17762-connectrix-congestion-spreading-and-how-to-avoid-it-wp.pdf)
   - If you must use ISLs, ensure you have checked on DLS settings for Brocade.
2. For example, avoid mixing different speeds over end-to-end paths.
   - For example, avoid using 16G host ports to access 32G storage ports.
   - Maintain standard speeds for all NVMe ports in the fabric.
   - If you must mix speeds, try to isolate the host-to-storage traffic to a single switch. This will help avoid Congestion Spreading.
3. Use the latest supported firmware version.
   - Where possible, ensure that the same version of firmware is used throughout the fabric. In homogeneous switch vendor environments, all switch firmware versions inside each fabric should be equivalent, except during the firmware upgrade process.
4. NVMe end-devices use a different FC-4 type from traditional FCP devices. To properly view the FC-4 type (e.g., via “nsshow” or “show fcns database”) you must run either:
   - Brocade FOS: 8.1.0 (or higher)
   - Cisco NX-OS: 8.1(1) (or higher)

5. It is recommended that you use Smart zoning (Cisco) or Peer-zoning (Broadcom/Brocade) in order to maintain zoning best practices as well as for ease of zone configuration & management.


7. To ensure maximum data availability, Dell EMC requires redundant physical and logical paths to prevent any single point of failure.

8. Consider using Dell EMC CloudIQ, a free Cloud application by Dell EMC, to monitor FC-NVMe and traditional FC devices in your SAN. CloudIQ also proactively monitors the SAN for Congestion and Slow Drain devices.

9. You should also enable vendor-specific monitoring features that alert on Slow drain and Congestion.

For other general best practices for Fibre Channel networks, please refer to the Fibre Channel SAN Topologies Techbook which can be found in the Dell EMC Topology Resource Center: https://elabnavigator.emc.com/eln/topologyResource.

**FC-NVMe Best Case Study**

**Overview**

As stated in Option 3 above, the purpose of this section is to provide detailed steps on virtually separating SCSI & NVMe traffic from each other so that a slow drain event does not impact our NVMe traffic. The slow drain could be caused by bandwidth mismatch and large block reads. To learn more about Congestion Spreading, refer to the Connectrix Whitepaper, Congestion Spreading and how to avoid it.

In this section you will find detailed steps on how to move from our starting topology to a virtually isolated fabric for both Connectrix B-Series and MDS series.

**Brocade**

**Pre-Requisites**

The following prerequisites should be met before proceeding:
1. FOS is at 8.1.0 or higher
2. Host Bus Adapter(HBA) is running the latest driver for the model according to Dell EMC E-Lab navigator support matrix at https://www.dellemc.com/en-us/products/interoperability/elab.htm#tab0=2

**Starting Topology**

![Figure 1](https://elabnavigator.emc.com/eln/topologyResource)
Figure 1 displays our starting topology. It is a traditional Fibre Channel SAN, where we have a Dell EMC server with a Fibre Channel Host Bus Adapter (HBA) port that is attached to a two-switch Connectrix 32G Fibre Channel SAN. The HBA port is running 16Gb/s. The Connectrix SAN has two 32Gb/s aggregate ISL links between them. There is a Dell EMC PowerMax with a Fibre Adapter (FA) front end port running at 16Gb/s.

Target Topology

![Target Topology Diagram]

Figure 2 displays our target topology. We are rolling in a new FC-NVMe capable server as well as adding new Fibre Channel NVMe Adapters (FN) on an existing Dell EMC PowerMax. In addition, you can see that we have increased our ISL count by providing dedicated ISL ports for FC-NVMe.

Our goal here is to create 2 logical switches to separate traditional SCSI traffic away from our high performing low latency NVMe traffic. In addition, we are also dedicating specific ISL links. By doing so, if there is a slow drain due to bandwidth mismatch and large block reads in our traditional SCSI fabric, our NVMe fabric is unaffected even though they are physically on the same switch. To do so, we will use a Connectrix B-Series feature called Virtual Fabrics (VF).

Detail Configuration steps

**WARNING:** The following steps will be disruptive! The user should plan for an outage window.

1. Backup the existing configuration in case it needs to be rolled back.
   
   *Note: Please review the file, as it is a text file, to ensure all the contents that you expect is there, specifically around zoning.*

   ```
   #switch_1> configupload
   Protocol (scp, ftp, sftp, local) [ftp]: ftp
   Server Name or IP Address [host]: 1.1.1.1
   User Name [user]: emc
   Path/Filename [<home dir>/config.txt]: before_virtual_fabrics.txt
   Section (all|chassis|switch [all]): all
   Password: 
   configUpload complete: All selected config parameters are uploaded
   ```

2. By default, Connectrix B-Series Virtual Fabric (VF) feature is disabled. Enabling VF will require an outage window because the switch will do a full cold reboot. There is no license needed for this feature. **THIS IS DISRUPTIVE – EXERCISE EXTREME CAUTION!**
#switch_1> fosconfig --enable vf
WARNING: This is a disruptive operation that requires a reboot to take effect.
All EX ports will be disabled upon reboot.
Would you like to continue [Y/N]: y
VF has been enabled. Your system is being rebooted.
Rebooting! Tue Sep 3 22:11:28 EDT 2019

3. After enabling VF, you will find that all your ports are listed in the default VF that has an FID of 128. In addition, your previous zoning will still be there, and your prompt will be different, indicating which FID you are currently in.

   Note: We will not use the default VF as there are specific rules around using it.

   #switch_1:FID128>

4. Repeat this section for Connectrix Switch 2.

Creating a logical Switch

1. For ease of use and management ability, we want to create two new logical switches, each of which will have an FID associated with them. One is for our traditional SCSI traffic and the other is for NVMe.

   Note: A logical switch is a construct for the end users (a.k.a just a “fake” reference). In real world practice, end users will typically refer to a logical switch by their FID number (i.e FID 100 on Connectrix Switch 1). FIDs must be the same ID number to merge between physically different switches. Please refer to the Brocade Fabric O.S. Administrator Guide for more details.

2. Create a SCSI VF ID

   #switch_1:FID128> lscfg --create 100
   A Logical switch with FID 100 will be created with default configuration.
   Would you like to continue [y/n]?: y
   Logical Switch with fid=100 has been successfully created.
   Logical Switch has been created with default configurations.
   Please configure the Logical Switch with appropriate switch and protocol settings before activating the Logical Switch.

3. Move all the traditional SCSI host, storage and ISL port(s) into the newly created VF.

   Note: When moving a port to a new VF, the port will be in a disabled state.

   #switch_1:FID128> lscfg --config 100 -port 0
   This operation requires that the affected ports be disabled.
   Would you like to continue [y/n]?: y
   Making this configuration change. Please wait...
   Configuration change successful.
   Please enable your ports/switch when you are ready to continue.

4. Once you have all your ports in the new SCSI VF, change your context to the SCSI VF and enable the ports.

   #switch_1:FID128> setcontext 100
   #switch_100:FID100:root> portenable 0

5. Repeat step 3-4 for additional SCSI ports.
6. Create a NVMe VF ID
   `#switch_1:FID128> lscfg --create 50
   A Logical switch with FID 50 will be created with default configuration.
   Would you like to continue [y/n]?: y
   About to create switch with fid=50. Please wait...
   Logical Switch with FID (50) has been successfully created.
   Logical Switch has been created with default configurations. Please configure
   the Logical Switch with appropriate switch and protocol settings before
   activating the Logical Switch.

7. Move all the new NVMe host, storage and ISL port(s) into the newly created VF.
   Note: When moving a port to a new VF, the port will be in a disabled state.
   `#switch_1:FID128> lscfg --config 50 -port 25
   This operation requires that the affected ports be disabled.
   Would you like to continue [y/n]?: y
   Making this configuration change. Please wait...
   Configuration change successful.
   Please enable your ports/switch when you are ready to continue.

8. Once you have all yours ports into the new NVMe VF, change your context to the NVMe VF and enable the ports.
   `#switch_1:FID128> setcontext 50
   #switch_50:FID50:root> portenable 25

9. Repeat step 7-8 for additional NVMe ports.

10. For your NVMe VF, make sure you disable the use of XISLs.
    `#switch_1:FID128> setcontext 50
    #switch_50:FID50:root> configure
    Not all options will be available on an enabled switch.
    To disable the switch, use the "switchDisable" command.
    Configure...
    Fabric parameters (yes, y, no, n): [no] y
    WWN Based persistent PID (yes, y, no, n): [no]
    Allow XISL Use (yes, y, no, n): [yes] n
    WARNING!! Disabling this parameter will cause removal of LISLs to
    other logical switches.
    Do you want to continue? (yes, y, no, n): [no] y
    Location ID: (0..4) [0] <use the key stroke "ctrl+d" to accept all other defaults>

11. Repeat this section for Connectrix switch 2.

Check Point

1. At this point, Connectrix switch 1 and 2 should have 2 additional VFs (FID 100,50) along with the default VF (128).

2. On the Connectrix switch 2, make sure that the switch domain id is different on FID 100 & 50 from that on the Connectrix switch 1. By default, the switch will sequentially assign Domain IDs, so you will get a "Domain Overlap" error when trying to bring up the ISLs.
#switch_100:FID100:root> switchshow

switchName: switch_100
switchType: 162.0
switchState: Online
switchMode: Native
switchRole: Principal
switchDomain: 1
switchId: fffc01
switchWwn: 10:00:50:eb:1a:f8:28:85
zoning: OFF
switchBeacon: OFF
FC Router: OFF
HIF Mode: OFF
Allow XISL Use: ON
LS Attributes: [FID: 100, Base Switch: No, Default Switch: No, Ficon Switch: No, Address Mode 0]

<table>
<thead>
<tr>
<th>Index</th>
<th>Port</th>
<th>Address</th>
<th>Media</th>
<th>Speed</th>
<th>State</th>
<th>Proto</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>35</td>
<td>010000</td>
<td>id</td>
<td>N16</td>
<td>Online</td>
<td>FC E-Port</td>
</tr>
<tr>
<td></td>
<td></td>
<td>segmented,10:00:c4:f5:7c:46:d1:8d (domain overlap)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#switch_100:FID100> switchdisable
#switch_100:FID100> configure

Configure...

Fabric parameters (yes, y, no, n): [no] y

Domain: (1..239) [1] 2

WWN Based persistent PID (yes, y, no, n): [no] <use the key stroke “ctrl+d” to accept all other defaults>

WARNING: The domain ID will be changed. The port level zoning may be affected

#switch_100:FID100> switchenable

3. Ensure that the fabric has successfully merged.

#switch_100:FID100:root> fabricshow

<table>
<thead>
<tr>
<th>Switch ID</th>
<th>Worldwide Name</th>
<th>Enet IP Addr</th>
<th>FC IP Addr</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: fffc01</td>
<td>10:00:c4:f5:7c:46:d1:8d</td>
<td>1.1.1.2</td>
<td>0.0.0.0</td>
<td>switch_100</td>
</tr>
<tr>
<td>2: fffc02</td>
<td>10:00:50:eb:1a:f8:28:85</td>
<td>1.1.1.2</td>
<td>0.0.0.0</td>
<td>switch_100</td>
</tr>
</tbody>
</table>

The Fabric has 2 switches

4. Now you can create a zoning configuration and zone the pwwns of your end devices.

Cisco

Pre-Requisites

The following prerequisites should be met before proceeding:
1. NX-OS is at 8.1(1) or higher
2. Host Bus Adapter(HBA) is running the latest driver for the model according to Dell EMC E-Lab navigator support matrix at 
   https://www.dellemc.com/en-us/products/interoperability/elab.htm#tab0=2
Starting Topology

Figure 3 displays our starting topology. It is a traditional Fibre Channel SAN, where we have a Dell EMC server with a Fibre Channel Host Bus Adapter (HBA) port that is attached to a two-switch Connectrix 32G Fibre Channel SAN. The HBA port is running 16Gb/s. The Connectrix SAN has two 32Gb/s aggregate ISL links between them. There is a Dell EMC PowerMax with a Fibre Adapter (FA) front end port running at 16Gb/s.

Target Topology

Figure 4 displays our target topology. We are rolling in a new FC-NVMe capable server as well as adding new Fibre Channel NVMe Adapters (FN) on an existing Dell EMC PowerMax. In addition, you can see that we have increased our ISL count by providing dedicated ISL ports for FC-NVMe.

Our goal here is to create 2 additional VSANs to isolate traditional SCSI traffic away from our high performing low latency NVMe traffic. In addition, we are also dedicating a specific ISL link. This way, if there is a slow drain due to bandwidth mismatch and large block reads in our traditional SCSI fabric, our NVMe fabric is unaffected even though they are both physically on the same switch.

Detail Configuration steps

Creating new VSANs & assign ports

WARNING: The following steps will be disruptive! The user should plan for an outage window.

1. Backup the existing configuration in the event we need to roll back.
   
   Note: Since it is a text file, please review the file to make sure that all the contents that you expect is there, specifically around zoning.
2. Create a SCSI and NVMe VSAN.

```plaintext
switch_1# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch_1(config)# vsan database
switch_1(config-vsan-db)# vsan 100 name scsi
switch_1(config-vsan-db)# vsan 50 name fc_nvme
```

3. Move all traditional SCSI host, storage and ISL ports to their new VSAN

```plaintext
switch_1# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch_1(config)# vsan database
switch_1(config-vsan-db)# vsan 100 interface fc 1/1
```

4. Repeat step 4 for additional SCSI ports.

5. Move all the NVMe host, storage and ISL ports to their new VSAN.

```plaintext
switch_1# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch_1(config)# vsan database
switch_1(config-vsan-db)# vsan 50 interface fc 1/8
```

6. Repeat step 6 for additional NVMe ports.

7. Repeat this section for Connectrix switch 2.

**Check Point**

1. At this point, Connectrix switch 1 and 2 should have 2 additional VSANs (100,50) along with the default VSAN 1.

2. The fabric should be fully merged and all 3 VSANs should be up and allowed (by default) in the “Trunk vsans (admin allowed and active)” for all ISLs interfaces.

```plaintext
switch_1# show topo
FC Topology for VSAN 1 :
--------------------------------------------------------------------------------
Interface  Peer Domain Peer Interface     Peer IP Address(Switch Name)
--------------------------------------------------------------------------------
fc1/7 0xc3(195)            fc1/7 1.1.1.2(switch_2)
fc1/8 0xc3(195)            fc1/8 1.1.1.2(switch_2)
fc1/9 0xc3(195)            fc1/9 1.1.1.2(switch_2)
fc1/10 0xc3(195)           fc1/10 1.1.1.2(switch_2)

FC Topology for VSAN 50 :
```
---

<table>
<thead>
<tr>
<th>Interface</th>
<th>Peer Domain</th>
<th>Peer Interface</th>
<th>Peer IP Address (Switch Name)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fc1/7</td>
<td>fcl/7</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/8</td>
<td>fcl/8</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/9</td>
<td>fcl/9</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/10</td>
<td>fcl/10</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
</tbody>
</table>

**FC Topology for VSAN 100:**

---

<table>
<thead>
<tr>
<th>Interface</th>
<th>Peer Domain</th>
<th>Peer Interface</th>
<th>Peer IP Address (Switch Name)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fc1/7</td>
<td>fcl/7</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/8</td>
<td>fcl/8</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/9</td>
<td>fcl/9</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
<tr>
<td>fc1/10</td>
<td>fcl/10</td>
<td>0xc3(195)</td>
<td>1.1.1.2(switch_2)</td>
</tr>
</tbody>
</table>

**Dedicate ISL links**

1. Dedicate specific ISL links to the traditional SCSI fabric by removing the NVMe dedicated VSAN.

   switch_1# config t
   Enter configuration commands, one per line. End with CNTL/Z.

   switch_1(config)# int fc 1/7-8

---

Dell Customer Communication - Confidential
2. Dedicate specific ISL links to the NVMe fabric by removing the SCSI dedicated VSAN.
   
   switch_1(config-if)# no switchport trunk allowed vsan 50

3. Repeat this section for Connectrix Switch 2.

4. At this point the fabric should still be fully merged, and you can now zone the devices within their respected dedicated VSANs.