Best Practices

Dell EMC VPLEX: SAN Connectivity

Implementation planning and best practices

Abstract

This document describes SAN connectivity for both host to Dell EMC™ VPLEX™ front-end and VPLEX to storage array back end. This document covers active/active and active/passive and ALUA array connectivity with illustrations detailing PowerStore™, Dell EMC Unity™ XT, and XtremIO™. Included also is information for metro node connectivity.

January 2021

Revisions

Date	Description
May 2020	Version 4: Updated ALUA connectivity requirements
January 2021	Added metro node content

Acknowledgments

Author: VPLEX CSE Team

VPLEXCSETeam@emc.com

Table of contents

Executive summary

There are four general areas of connectivity for VPLEX:

1. Host to VPLEX connectivity.
2. VPLEX to array connectivity
3. WAN connectivity for either Fibre Channel or Ethernet protocols.
4. LAN connectivity.

This document is designed to address frontend SAN connectivity for host access to Dell EMC™ VPLEX™ Virtual Volumes and VPLEX backend SAN connectivity for access to storage arrays. WAN and LAN connectivity will be presented in VPLEX Networking Best Practices document.

Host connectivity to VPLEX, referred to as frontend connectivity, have specific requirements which differ from the backend or VPLEX to array connectivity. These requirements are based on architectural design limitations and are checked during health checks and ndu pre-checks. Connectivity violations will be flagged and prevent the system from being upgraded. Bypassing these requirements is allowed for test environments or Proof of Concept but it is not supported for production environments. System limits are available in the Release Notes for each GeoSynchrony Code level.

Array connectivity varies based on array type and performance or archival requirements. Path limit requirements are based on “active” path count thereby total path count would be double for active/passive or ALUA arrays. ALUA arrays are considered active/passive from VPLEX point of view as the non-preferred paths are not used except when the primary controller no longer has control of the LUN such as during an array ndu. Active/active array’s paths are all considered active regardless of host multipath software.

Content has been added to cover metro node. Metro node is an external hardware/software add-on feature based on the Dell PowerEdge™ R640 platform for both PowerStore™ and Dell EMC Unity™ XT arrays for which it provides active/active synchronous replication as well as standard local use cases. Additionally, it also provides a solution locally with the Local mirror feature to protect data from a potential array failure. Both of these use cases provide solutions for true continuous availability with zero downtime.

Factors that impact decisions on which product should be used for protection revolve around RT0 (Recover Time Objective) = How long can you can afford your applications to be down, RPO (Recover Point Objective) = How much data you can afford to lose when you roll back to a previous known good point in time, and DTO (Decision Time Objective) = The time it take to make the decision to cut over and roll back the data which adds to the RTO. Metro node provides an active/active storage solution that allows data access for a multitude of host types and applications giving them zero data unavailability (DU) and zero data loss (DL) which translates into zero DTO for any type of failure up to and including loss of an entire datacenter. Furthermore, metro node provides transparent data access even during an array outage through locally mirrored arrays or through a Metro configuration where the read/write I/O is serviced via the remote site. This solution is superior to array based replication for this reason. This solution achieves zero RTO, zero RPO and zero DTO.

The feature is deployed as a greenfield solution and is intended to be purchased at the same time as part of the order for either PowerStore or Unity XT array. All brownfield deployments (those requesting to upgrade an existing environment for active/active synchronous replication) will be supported by VPLEX VS6 hardware/software and the best practices for that solution will be covered in the VPLEX documentation.

The metro node feature must be installed as either a Local or a Metro solution initially. It is currently not supported to upgrade a Local to a Metro at a later date.

This document will cover best practices for installation, configuration and operation of the metro node feature. There may be similarities with the VPLEX product but there are differences within the product code bases which may result in changes to processes, procedures or even some commands which make this the authoritative document for metro node.

Due to feature differences between PowerStore and Unity XT, there will be connectivity and configuration details specific to the array platform. Please familiarize yourself with the specific differences outlined within this document.

Most synchronous replication solutions require like to like between the two sites meaning same array models and microcode. This applies at the metro node layer but not at the array level. The only recommendation is that the arrays at both sites have similar performance levels. It is fully supported to have a Unity XT array at one site of the metro and a PowerStore array at the other and this mixed configuration is also supported starting with Ansible 1.1.

Metro node supports both Local RAID-1 and Metro Distributed RAID-1 device structures. PowerStore can take advantage of the Local RAID-1 by mirroring across appliances in a clustered PowerStore configuration protecting the data and allowing host access even during an outage of one appliance. Local RAID-1 resynchronizations will always be a full resynchronization whereas a Distributed RAID-1 will utilize the Logging volume mapping files for incremental rebuilds.

1 Frontend connectivity

1.1 Frontend/host initiator port connectivity

- Dual fabric designs are considered a best practice
- The front-end I/O modules on each director should have a minimum of two physical connections one to each fabric (required)
- Each host should have at least one path to an A director and one path to a B director on each fabric for a total of four logical paths (required for NDU).
- Multipathing or path failover software is required at the host for access across the dual fabrics
- Each host should have fabric zoning that provides redundant access to each LUN from a minimum of an A and B director from each fabric.
- Four paths are required for NDU
- Dual and Quad engine VPLEX Clusters require spanning engines for host connectivity on each fabric
- Observe Director CPU utilization and schedule NDU for times when average directory CPU utilization is below 50%
- GUI Performance Dashboard in GeoSynchrony 5.1 or newer
- Skipping the NDU pre-checks would be required for host connectivity with less than four paths and is not considered a best practice
  - This configuration would be supported for a test environment or POC

Note: For cluster upgrades when going from a single engine to a dual engine cluster or from a dual to a quad engine cluster you must rebalance the host connectivity across the newly added engines. Adding additional engines and then not connecting host paths to them is of no benefit. The NDU pre-check will flag host connectivity that does not span engines in a multi-engine VPLEX cluster as a configuration issue. When scaling up a single engine cluster to a dual, the ndu pre-check may have passed initially but will fail after the addition of the new engine which is why the host paths must be rebalanced across both engines. Dual to Quad upgrade will not flag an issue provided there were no issues prior to the upgrade. You may choose to rebalance the workload across the new engines or add additional hosts to the pair of new engines.

Complete physical connections to the VPLEX before commissioning/setup.

Use the same FE/BE ports on each director to avoid confusion, that is, B0-FC00 and A0-FC00. Please refer to hardware diagrams for port layout.

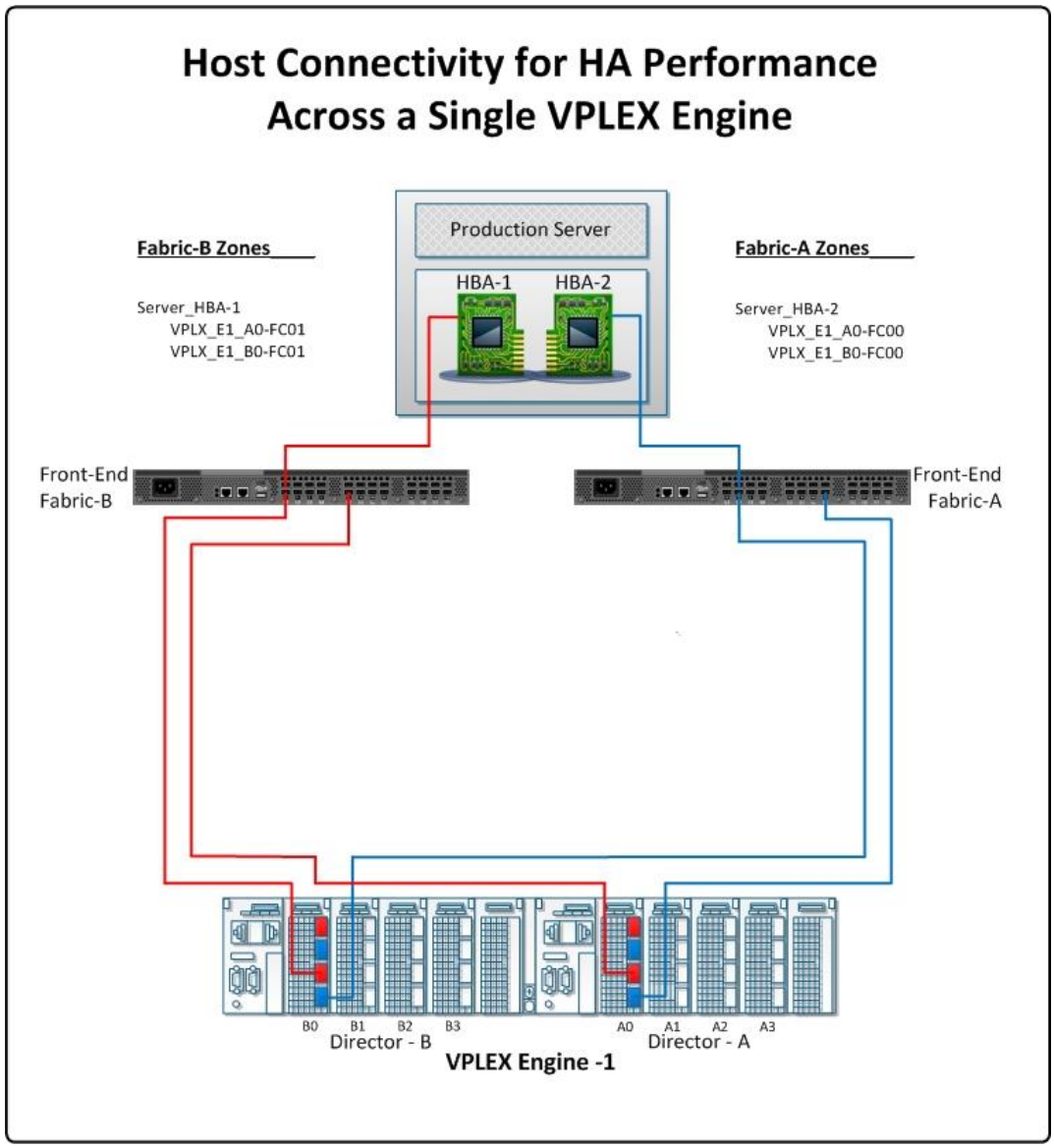

Figure 1 Host connectivity for Single Engine Cluster meeting NDU pre-check requirements

This illustration shows dual HBAs connected to two Fabrics with each connecting to two VPLEX directors on the same engine in the single engine cluster. This is the minimum configuration that would meet NDU requirements.

Please refer to the Release Notes for the total FE port IT Nexus limit.

Note: Each Initiator / Target connection is called an IT Nexus.
Please refer to the VPLEX Hardware Reference guide available on SolVe for port layouts of other VPLEX hardware families.

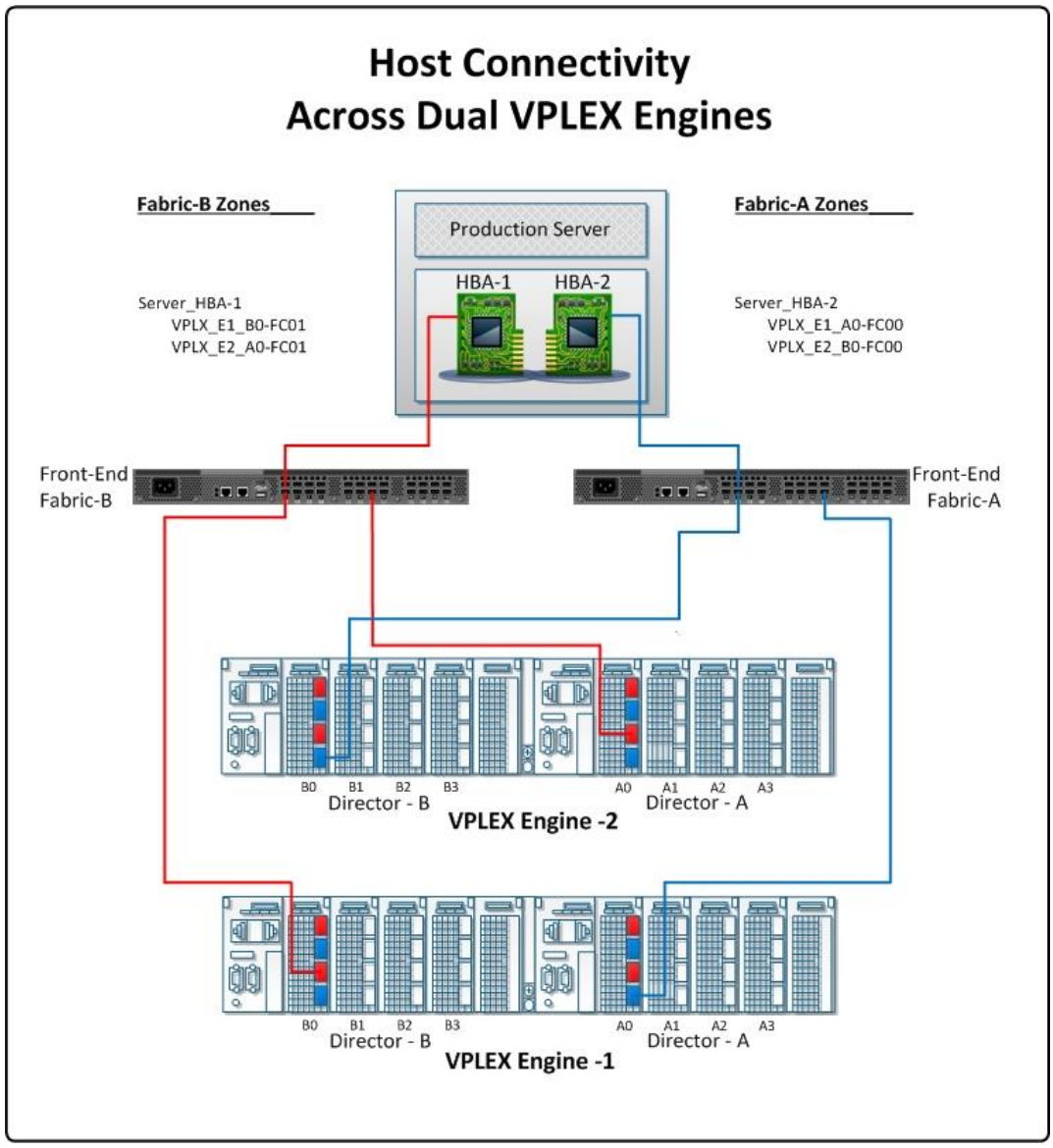

Figure 2 Host connectivity for HA requirements for NDU pre-checks dual or quad engine

The previous illustration shows host connectivity with dual HBAs connected to four VPLEX directors. This configuration offers increased levels of HA as required by the NDU pre-checks. This configuration applies to both the Dual and Quad VPLEX Clusters. This configuration still only counts as four IT Nexus against the limit as identified in the Release Notes for that version of GeoSynchrony.

Note: Please refer to the VPLEX Hardware Reference guide available on SolVe for port layouts of other VPLEX hardware families.

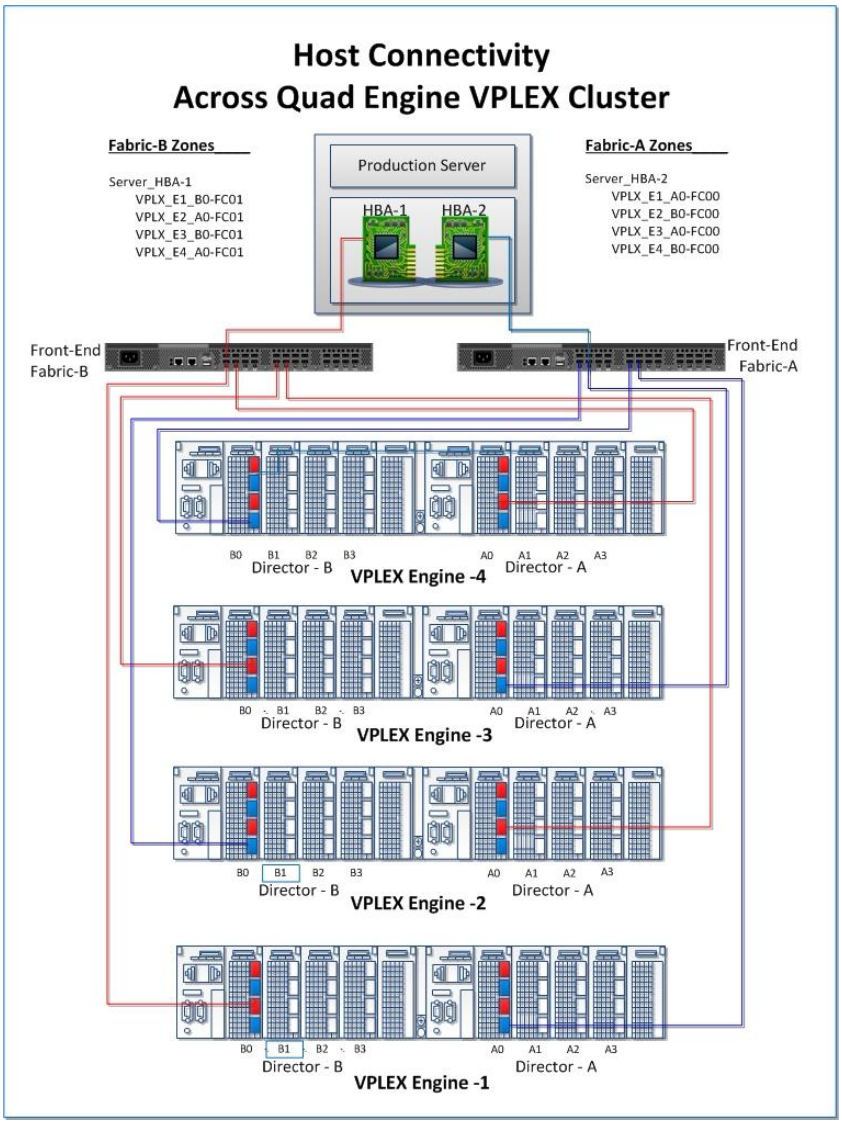

Figure 3 Host connectivity for HA quad engine

The previous illustration shows host connectivity with dual HBAs connected to four VPLEX engines (eight directors). This configuration counts as eight IT Nexuses against the total limit as defined in the Release Notes for that version of GeoSynchrony. Hosts using active/passive path failover such as VMware NMP software should connect a path to all available directors and manual load balance by selecting a different director for the active path with different hosts.

Note: Most host connectivity for hosts running load balancing software should follow the recommendations for a dual engine cluster. The hosts should be configured across two engines and subsequent hosts should alternate between pairs of engines effectively load balancing the total combined I/O across all engines. This will reduce the resource utilization of IT Nexus system limitation.

1.2 Host cross-cluster connect

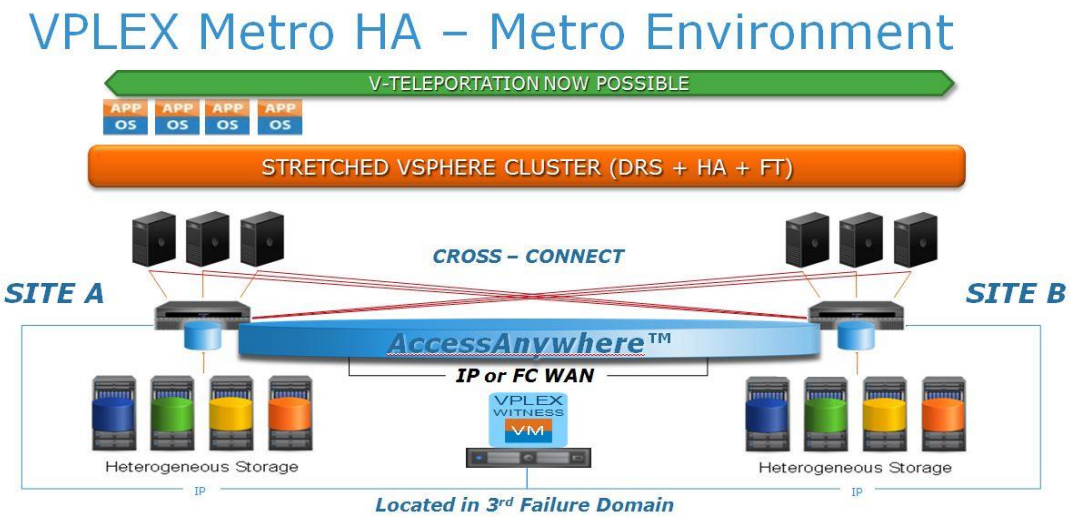

Figure 4 Host Cluster connected across sites to both VPLEX Clusters

- PowerPath VE provides an auto standby feature created specifically for this environment
- Host cross-cluster connect applies to specific host OS and multipathing configurations as listed in the VPLEX ESSM only.
- Host initiators are zoned to both VPLEX clusters in a Metro.
- Host multipathing software can be configured for active path/passive path with active path going to the local VPLEX cluster. When feasible, configure the multipathing driver to prefer all local cluster paths over remote cluster paths.
- Separate HBA ports should be used for the remote cluster connection to avoid merging of the local and remote fabrics
- Connectivity at both sites follow same rules as single host connectivity
- Supported stretch clusters can be configured using host cross-cluster connect (Please refer to VPLEX ESSM)
- Host cross-cluster connect is limited to a VPLEX cluster separation of no more than 1ms latency
- Host cross-cluster connect requires the use of VPLEX Witness
- VPLEX Witness works with Consistency Groups only
- Host cross-cluster connect must be configured using VPLEX Distributed Devices only
- Host cross-cluster connect is supported in a VPLEX Metro synchronous environment only
- At least one backend storage array is required at each site with redundant connection to the VPLEX cluster at that site. Arrays may be cross connected for metadata and logging volumes only
- All Consistency Groups used in a host cross-cluster connect configuration are required to have the auto-resume attribute set to true

The unique solution provided by Host cross-cluster connect requires hosts have access to both datacenters. The latency requirements for host cross-cluster connect can be achieved using an extended fabric or fabrics that span both datacenters. The use of backbone fabrics may introduce additional latency preventing a viable use of host cross-cluster connect. The rtt must be within 1ms.

If using PowerPath VE, the only thing that the customer has to do is enable the autostandby feature:

```
#powermt set autostandby=on trigger=prox host=xxx
```

PowerPath will take care of setting to autostandby those paths associated with the remote/non-preferred VPLEX cluster. PP groups the paths by VPLEX cluster and the one with the lowest minimum path latency is designated as the local/preferred cluster.

1.3 VBLOCK and VPLEX front end connectivity rules

Note: All rules in BOLD cannot be broken, however Rules in *Italics* can be adjusted depending on customer requirement, but if these are general requirements simply use the suggested rule.

1. Physical FE connectivity

   a. Each VPLEX Director has 4 front end ports. 0, 1, 2 and 3. In all cases even ports connect to fabric A and odd ports to fabric B.

   i. For single VBLOCKS connecting to single VPLEX

   - **Only ports 0 and 1 will be used on each director. 2 and 3 are reserved.**
   - **Connect even VPLEX front end ports to fabric A and odd to fabric B.**

   ii. For two VBLOCKS connecting to a single VPLEX

   - **Ports 0 and 1 will be used for VBLOCK A**
   - **Ports 2 and 3 used for VBLOCK B**

2. Connect even VPLEX front end ports to fabric A and odd to fabric B. ESX Cluster Balancing across VPLEX Frontend

All ESX clusters are evenly distributed across the VPLEX front end in the following patterns:

Single Engine		
Engine #	Engine 1	
Director	A	B
Cluster #	1,2,3,4,5,6,7,8	1,2,3,4,5,6,7,8

Dual Engine				
Engine #	Engine 1		Engine 2	
Director	A	B	A	B
Cluster #	1,3,5,7	2,4,6,8	2,4,6,8	1,3,5,7

Quad Engine								
Engine #	Engine 1		Engine 2		Engine 3		Engine 4	
Director	A	B	A	B	A	B	A	B
Cluster#	1,5	2,6	3,7	4,8	4,8	3,7	2,6	1,3

3. Host / ESX Cluster rules

   a. Each ESX cluster must connect to a VPLEX A and a B director.
   b. For dual and quad configs, A and B directors must be picked from different engines (see table above for recommendations)
   c. Minimum directors that an ESX cluster connects to is 2 VPLEX directors.
   d. Maximum directors that an ESX cluster connects to is 2 VPLEX directors.
   e. Any given ESX cluster connecting to a given VPLEX cluster must use the same VPLEX frontend ports for all UCS blades regardless of host / UCS blade count.
   f. Each ESX host should see four paths to the same datastore

i. 2 across fabric A

- A VPLEX A Director port 0 (or 2 if second VBLOCK)
- A VPLEX B Director port 0 (or 2 if second VBLOCK)

ii. 2 across fabric B

- The same VPLEX A Director port 1 (or 3 if second VBLOCK)
- The same VPLEX B Director port 1 (or 3 if second VBLOCK)

4. Pathing policy

a. Non cross connected configurations recommend to use adaptive pathing policy in all cases. Round robin should be avoided especially for dual and quad systems.

b. For cross connected configurations, fixed pathing should be used and preferred paths set per Datastore to the local VPLEX path only taking care to alternate and balance over the whole VPLEX front end (i.e. so that all datastores are not all sending IO to a single VPLEX director).

2 ESXi path loss handling

2.1 Path loss handling semantics (PDL and APD)

vSphere can recognize two different types of total path failures to an ESXi 5.0 u1 and newer server. These are known as "All Paths Down" (APD) and "Persistent Device Loss" (PDL). Either of these conditions can be declared by the VMware ESXi™ server depending on the failure condition.

2.1.1 Persistent device loss (PDL)

A storage device is considered to be in the permanent device loss (PDL) state when it becomes permanently unavailable to your ESXi host. Typically, the PDL condition occurs when a device is unintentionally removed, its unique ID changes, when the device experiences an unrecoverable hardware error, or in the case of a vSphere Metro Storage Cluster WAN partition. When the storage determines that the device is permanently unavailable, it sends SCSI sense codes to the ESXi host. The sense codes allow your host to recognize that the device has failed and register the state of the device as PDL. The sense codes must be received on all paths to the device for the device to be considered permanently lost. If virtual machine files do not all reside on the same datastore and a PDL condition exists on one of the datastores, the virtual machine will not be killed. VMware recommends placing all files for a given virtual machine on a single datastore, ensuring that PDL conditions can be mitigated by vSphere HA.

When a datastore enters a Permanent Device Loss (PDL) state, High Availability (HA) can power off virtual machines and restart them later. A virtual machine is powered off only when issuing I/O to the datastore. Otherwise, it remains active. A virtual machine that is running memory-intensive workloads without issuing I/O to the datastore might remain active in such situations. VMware offers advanced options to regulate the power off and restart operations for virtual machines. The following settings apply only to a PDL condition and not to an APD condition.

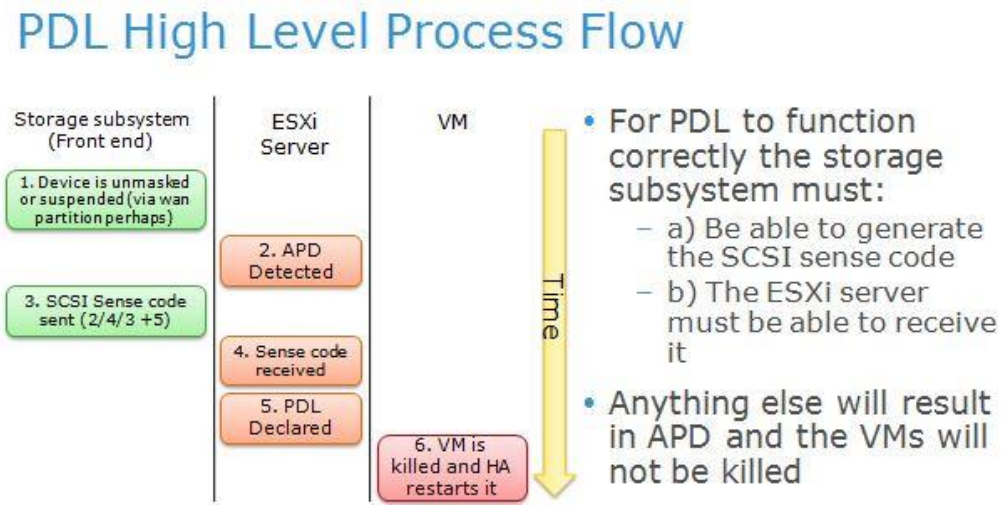

Figure 5 Persistent device loss process flow

Advanced settings have been introduced in VMware vSphere 5.0 Update 1 and 5.5 to enable vSphere HA to respond to a PDL condition. The following settings are for the hosts and VMs in the stretched cluster consuming the virtual storage.

Note: PDL response works in conjunction with DRS rules. If the rule is set to “must”, VMware HA will not violate the rule. If the rule is set to “should”, VMware HA will violate it. The DRS rule should be set to “should” to provide availability.

2.1.2 vSphere storage for vSphere 5.0 U1

2.1.2.1 disk.terminateVMonPDLDefault set to true:

For each host in the cluster, create and edit /etc/vmware/settings with Disk.terminateVMOnPDLDefault=TRUE, then reboot each host.

2.1.2.2 das.maskCleanShutdownEnabled set to true:

HA Advanced Option. If the option is unset in 5.0 U1, a value of false is assumed, whereas in ESXi 5.1 and later, a value of true is assumed. When a virtual machine powers off and its home datastore is not accessible, HA cannot determine whether the virtual machine should be restarted. So, it must make a decision. If this option is set to false, the responding FDM master will assume the virtual machine should not be restarted, while if this option is set to true, the responding FDM will assume the virtual machine should be restarted.

2.1.3 vSphere storage for vSphere 5.5

2.1.3.1 disk.terminateVMOnPDLDefault set to default:

Advanced Virtual Machine Option. Default value is FALSE. When TRUE, this parameter powers off the virtual machine if any device that backs up the virtual machine's datastore enters the PDL state. HA will not restart this virtual machine. When set to DEFAULT, VMkernel.Boot.terminateVMOnPDL is used.

2.1.3.2 VMkernel.Boot.terminateVMOnPDL set to true:

Advanced Vmkernel Option. Default value is FALSE. When set to TRUE, this parameter powers off all virtual machines on the system when storage that they are using enters the PDL state. Setting can be overridden for each virtual machine by disk.terminateVMOnPDLDefault parameter. Can be set only to TRUE or FALSE. With vSphere web client:

1. Browse to the host in the vSphere Web Client navigator.
2. Click the Manage tab and click Settings.
3. Under System, click Advanced System Settings.
4. In Advanced Settings, select the appropriate item.
5. Click the Edit button to edit the value.
6. Click OK
7. Reboot the host

2.1.3.3 das.maskCleanShutdownEnabled set to default:

HA Advanced Option. This option is set to TRUE by default. It allows HA to restart virtual machines that were powered off while the PDL condition was in progress. When this option is set to true, HA restarts all virtual machines, including those that were intentionally powered off by a user.

2.1.3.4 disk.AutoremoveOnPDL set to 0:

Advanced Vmkernel option. Default is 1. In the case of a vMSC environment the PDL’s are likely temporary because one site has become orphaned from the other, in which case a failover has occurred. If the devices in a PDL state are removed permanently when the failure or configuration error of the vMSC environment is fixed they will not automatically be visible to the hosts again. This will require a manual rescan in order to

bring the devices back into service. The whole reason for having a vMSC environment is that it handles these types of things automatically. So you don't want to have to do manual rescans all the time. For this reason the PDL AutoRemove functionality should be disabled on all hosts that are part of a vMSC configuration. Please note that this is recommended for Uniform or Non-Uniform vMSC configurations. Any vMSC configuration that could cause a PDL should have the setting changed. To disable this feature:

1. Connect to the ESXi host using the console or SSH. For more information, see Using Tech Support Mode in ESXi 4.1 and ESXi 5.x (KB article 1017910).
2. Run this command to disable AutoRemove: esxcli system settings advanced set -o "/Disk/AutoremoveOnPDL" -i 0

Or with vSphere web client:

1. Browse to the host in the vSphere Web Client navigator.
2. Click the Manage tab and click Settings.
3. Under System, click Advanced System Settings.
4. In Advanced Settings, select the appropriate item.
5. Click the Edit button to edit the value.
6. Click OK

2.1.3.5 Permanent device loss

- Remove device from VPLEX, remove or offline a LUN from backend.
- WAN partition, disable wan ports from switch or log in vplex and disable wan ports using vplexcli

2.1.3.6 All paths down

- Remove volume from storage view.
- Remove FC ports from ESXi host, can cause other errors.
- Disable FC ports on switch.

2.1.4 APD handling

A storage device is considered to be in the all paths down (APD) state when it becomes unavailable to your ESXi host for an unspecified period of time. The reasons for an APD state can be, for example, a failed switch.

In contrast with the permanent device loss (PDL) state, the host treats the APD state as transient and expects the device to be available again.

The host indefinitely continues to retry issued commands in an attempt to reestablish connectivity with the device. If the host's commands fail the retries for a prolonged period of time, the host and its virtual machines might be at risk of having performance problems and potentially becoming unresponsive.

With vSphere 5.1, a default APD handling feature was introduced. When a device enters the APD state, the system immediately turns on a timer and allows your host to continue retrying non-virtual machine commands for a limited time period.

2.2 vSphere storage for vSphere 5.1

2.2.1 Disable storage APD handling

The storage all paths down (APD) handling on your ESXi host is enabled by default. When it is enabled, the host continues to retry I/O commands to a storage device in the APD state for a limited time period. When the time period expires, the host stops its retry attempts and terminates any I/O. You can disable the APD handling feature on your host.

If you disable the APD handling, the host will indefinitely continue to retry issued commands in an attempt to reconnect to the APD device. Continuing to retry is the same behavior as in ESXi version 5.0. This behavior might cause virtual machines on the host to exceed their internal I/O timeout and become unresponsive or fail. The host might become disconnected from vCenter Server.

2.2.1.1 Procedure

1. Browse to the host in the vSphere Web Client navigator.
2. Click the **Manage** tab and click **Settings**.
3. Under System, click Advanced System Settings.
4. Under Advanced System Settings, select the **Misc.APDHandlingEnable** parameter and click the Edit icon.
5. Change the value to 0.

If you disabled the APD handling, you can re-enable it when a device enters the APD state. The internal APD handling feature turns on immediately and the timer starts with the current timeout value for each device in APD.

Note: The host cannot detect PDL conditions and continues to treat the device connectivity problems as APD when a storage device permanently fails in a way that does not return appropriate SCSI sense codes.

2.2.2 Change timeout limits for storage APD

The timeout parameter controls how many seconds the ESXi host will retry non-virtual machine I/O commands to a storage device in an all paths down (APD) state. If needed, you can change the default timeout value.

The timer starts immediately after the device enters the APD state. When the timeout expires, the host marks the APD device as unreachable and fails any pending or new non-virtual machine I/O. Virtual machine I/O will continue to be retried.

The default timeout parameter on your host is 140 seconds. You can increase the value of the timeout if, for example, storage devices connected to your ESXi host take longer than 140 seconds to recover from a connection loss.

2.2.2.1 Procedure

1. Browse to the host in the vSphere Web Client navigator.
2. Click the **Manage** tab, and click **Settings**.
3. Under System, click Advanced System Settings.
4. Under Advanced System Settings, select the **Misc.APDTimeout** parameter and click the Edit icon.
5. Change the default value.

2.2.3 PDL/APD references

http://www.emc.com/collateral/software/white-papers/h11065-vplex-with-vmware-ft-ha.pdf for ESXi 5.0 U1 test scenarios

http://www.vmware.com/files/pdf/techpaper/vSPHR-CS-MTRO-STOR-CLSTR-USLET-102-HI-RES.pdf for ESXi 5.0 U1 vmware vsphere metro storage cluster case study

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2007545

http://www.emc.com/collateral/software/white-papers/h11065-vplex-with-vmware-ft-ha.pdf

http://www.vmware.com/files/pdf/techpaper/vSPHR-CS-MTRO-STOR-CLSTR-USLET-102-HI-RES.pdf

http://pubs.vmware.com/vsphere-50/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-50-storage-guide.pdf

http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-55-storage-guide.pdf

https://support.emc.com/search/?text=implementation%20and%20planning%20best%20practices%20for%20emc%20vplex

http://www.boche.net/blog/index.php/2014/07/14/yet-another-blog-post-about-vsphere-ha-and-pdl

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2033250

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2059622

3 VPLEX backend array connectivity

3.1 Back-end/storage array connectivity

The best practice for array connectivity is to use A/B fabrics for redundancy however VPLEX is also capable of backend direct connect. This practice is immediately recognizable as being extremely limited. The following best practices for fabric connect should be followed with regards to the direct connect where applicable.

Direct connect is intended for Proof of Concept, test/dev and specific sites that have only 1 array. This allows for backend connectivity while reducing the overall cost of switch ports. Sites with multiple arrays or large implementations should utilize SAN connectivity as that provides the optimal solution overall.

Note: Direct connect applies only to backend connectivity. Frontend direct connect is not supported.

3.1.1 Active/active arrays

- Each director in a VPLEX cluster must have a minimum of two I/O paths to every local back-end storage array and to every storage volume presented to that cluster (required). This is referred to as an ITL or Initiator/Target/LUN
- Each director will have redundant physical connections to the back-end storage across dual fabrics (required). Each director is required to have redundant paths to every back-end storage array across both fabrics
- Each storage array should have redundant controllers connected to dual fabrics, with each VPLEX Director having a minimum of two ports connected to the back-end storage arrays through the dual fabrics (required)

VPLEX recommends a **maximum** of 4 active paths per director to a given LUN (Optimal). This is considered optimal because each director will load balance across the four active paths to the storage volume.

Note: Exceeding the maximum of 4 active paths per director per LUN is not supported.

High quantities of storage volumes or entire arrays provisioned to VPLEX should be divided up into appropriately sized groups (i.e. masking views or storage groups) and presented from the array to VPLEX via groups of four array ports per VPLEX director so as not to exceed the four active paths per VPLEX director limitation. As an example, following the rule of four active paths per storage volume per director (also referred to as ITLs), a four engine VPLEX cluster could have each director connected to four array ports dedicated to that director. In other words, a quad engine VPLEX cluster would have the ability to connect to 32 ports on a single array for access to a single device presented through all 32 ports and still meet the connectivity rules of 4 ITLs per director. This can be accomplished using only two ports per backend I/O module leaving the other two ports for access to another set of volumes over the same or different array ports.

Appropriateness would be judged based on things like the planned total IO workload for the group of LUNs and limitations of the physical storage array. For example, storage arrays often have limits around the number of LUNs per storage port, storage group, or masking view they can have.

Maximum performance, environment wide, is achieved by load balancing across maximum number of ports on an array while staying within the IT limits. Performance is not based on a single host but the overall impact of all resources being utilized. Proper balancing of all available resources provides the best overall performance.

Load balancing via Host Multipath between VPLEX directors and then from the four paths on each director balances the load equally between the array ports.

1. Zone VPLEX director A ports to one group of four array ports.
2. Zone VPLEX director B ports to a different group of four array ports.
3. Repeat for additional VPLEX engines.
4. Create a separate port group within the array for each of these logical path groups.
5. Spread each group of four ports across array engines for redundancy.
6. Mask devices to allow access to the appropriate VPLEX initiators for both port groups.

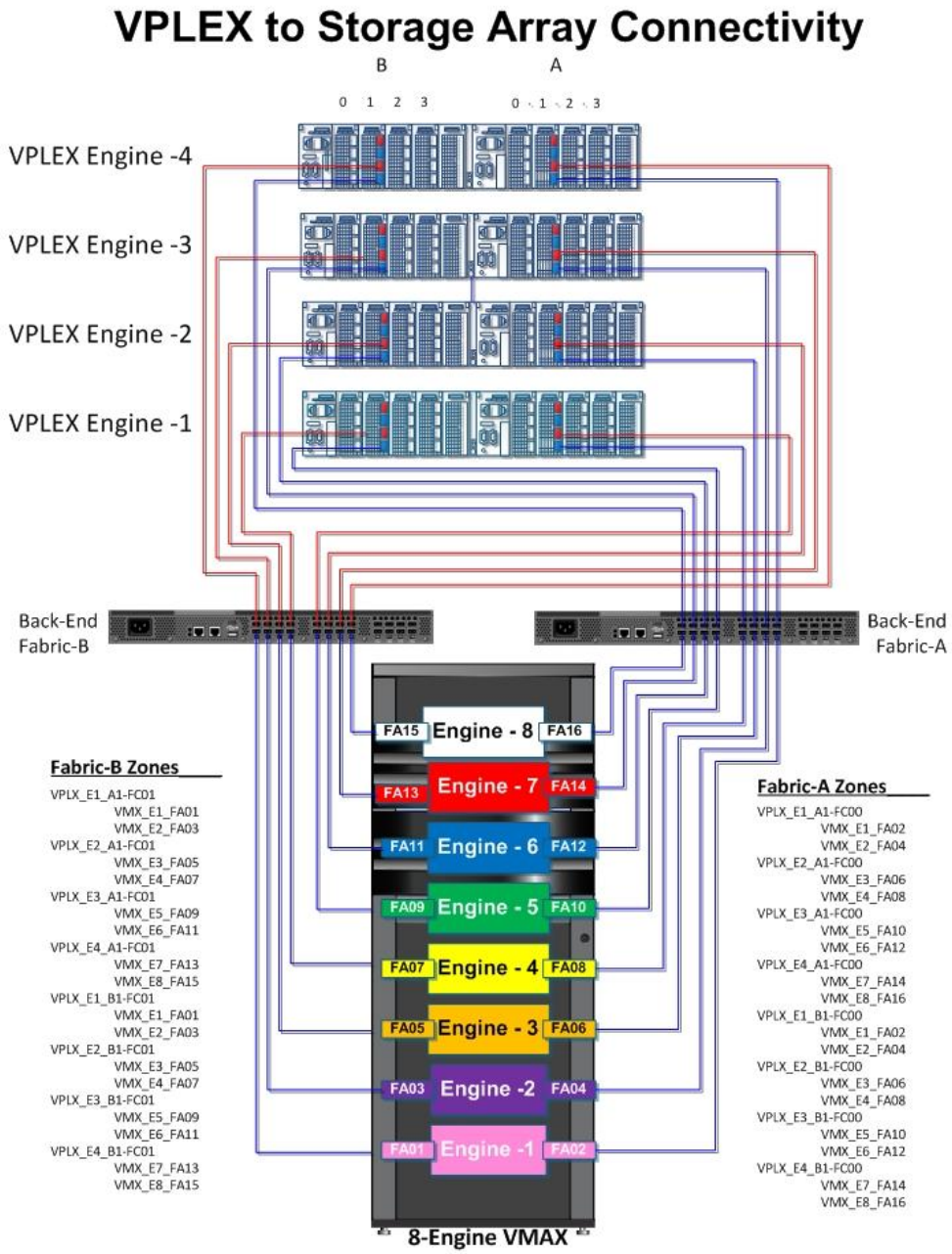

Figure 6 Active/Active Array Connectivity

This illustration shows the physical connectivity to a VMAX array. Similar considerations should apply to other active/active arrays. Follow the array best practices for all arrays including third party arrays.

The devices should be provisioned in such a way as to create “digestible” chunks and provisioned for access through specific FA ports and VPLEX ports. The devices within this device grouping should restrict access to four specific FA ports for each VPLEX Director ITL group.

The VPLEX initiators (backend ports) on a single director should spread across engines to increase HA and redundancy. The array should be configured into initiator groups such that each VPLEX director acts as a single host per four paths.

This could mean four physical paths or four logical paths per VPLEX director depending on port availability and whether or not VPLEX is attached to dual fabrics or multiple fabrics in excess of two.

For the example above following basic limits on the VMAX:

Initiator Groups	(HBAs); max of 32 WWN's per IG; max of 8192 IG's on a VMax; set port flags on the IG; an individual WWN can only belong to 1 IG. **Cascaded** Initiator Groups have other IG's (rather than WWN's) as members.
Port Groups	(FA ports): max of 512 PG's; ACLX flag must be enabled on the port; ports may belong to more than 1 PG
Storage Groups	(LUNs/SymDevs); max of 4096 SymDevs per SG; a SymDev may belong to more than 1 SG; max of 8192 SG's on a VMax
Masking View	= Initiator Group + Port Group + Storage Group

We have divided the backend ports of the VPLEX into two groups allowing us to create four masking views on the VMAX. Ports FC00 and FC01 for both directors are zoned to two FA’s each on the array. The WWN’s of these ports are the members of the first Initiator Group and will be part of Masking View 1. The Initiator Group created with this group of WWN’s will become the member of a second Initiator Group which will in turn become a member of a second Masking View. This is called Cascading Initiator Groups. This was repeated for ports FC02 and FC03 placing them in Masking Views 3 and 4. This is only one example of attaching to the VMAX and other possibilities are allowed as long as the rules are followed.

VPLEX virtual volumes should be added to Storage Views containing initiators from a director A **and** initiators from a director B. This translates to a single host with two initiators connected to dual fabrics and having four paths into two VPLEX directors. VPLEX would access the backend array’s storage volumes via eight FA’s on the array through two VPLEX directors (an A director and a B director). The VPLEX A director and B director each see four different FA’s across at least two VMAX engines if available.

This is an optimal configuration that spreads a single host’s I/O over the maximum number of array ports. Additional hosts will attach to different pairs of VPLEX directors in a dual-engine or quad-engine VPLEX cluster. This will help spread the overall environment I/O workload over more switches, VPLEX and array resources.

This would allow for the greatest possible balancing of all resources resulting in the best possible environment performance.

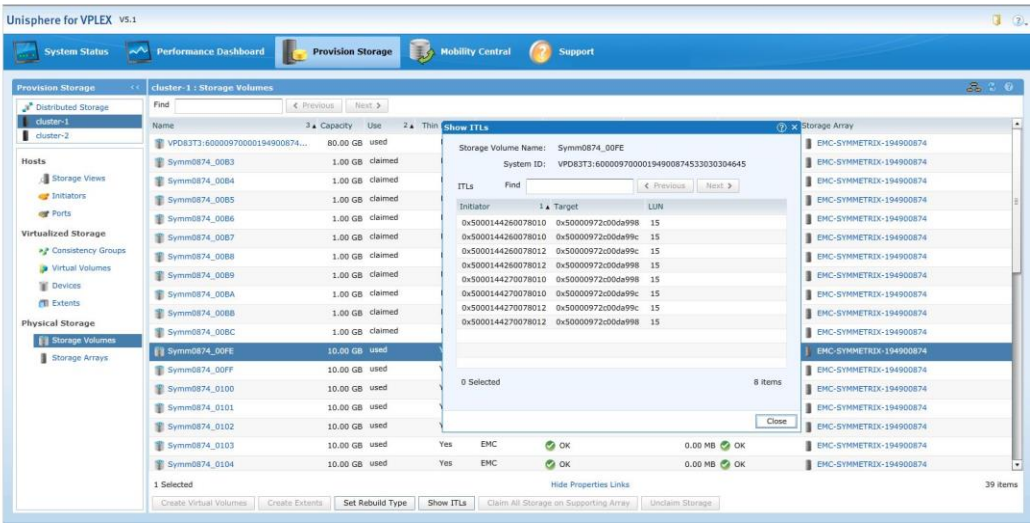

Figure 7 Show ITLs per Storage Volume

This illustration shows the ITLs per Storage Volume. In this example the VPLEX Cluster is a single engine and is connected to an active/active array with four paths per Storage Volume per Director giving us a total of eight logical paths. The Show ITLs panel displays the ports on the VPLEX Director from which the paths originate and which FA they are connected to.

The proper output in the Show ITLs panel for an active/passive or ALUA supported array would have double the count as it would also contain the logical paths for the passive or non-preferred SP on the array.

3.1.2 Active/passive and ALUA arrays

Some arrays have architecture and implementation requirements that necessitate special consideration. When using an active-passive or ALUA supported array, each director needs to have logical (zoning and masking) and physical connectivity to both the active and passive or non-preferred controllers. That way you will not lose access to storage volumes if an active controller should fail. Additionally, arrays may have limitations on the size of initiator or storage groups. It may be necessary to have multiple groups to accommodate provisioning storage to the VPLEX. Adhere to logical and physical connectivity guidelines discussed earlier.

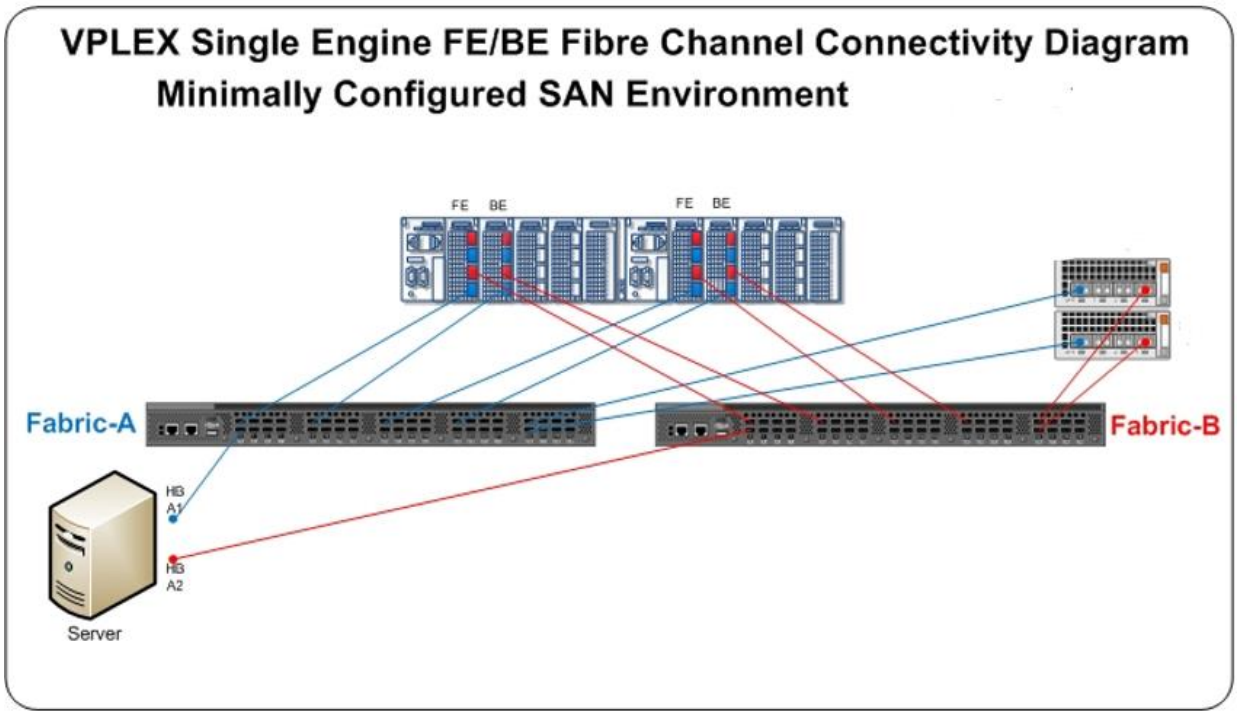

Figure 8 VS2 connectivity to Active/Passive and ALUA Arrays

Points to note would be that for each Active/Passive or ALUA array, each SP has connection to each fabric through which each SP has connections to all VPLEX directors. The above example shows Fabric-A with SPa0 & SPb0 (even ports) and Fabric-B with SPa3 & SPb3(odd ports) for dual fabric redundancy.

ALUA support allows for connectivity similar to Active/Passive arrays. VPLEX will recognize the non-preferred path and refrain from using it under normal conditions. Additionally, some LUNs on the array will be owned by one SP while other LUNs will be owned by the other SP. This means that even though we are calling some paths active and other paths passive, that applies to specific LUNs. The passive paths for some LUNs are the active paths for the LUNs owned by the other SP. A director with proper maximum path connectivity will show eight ITLs per director but will only report four active paths.

When provisioning storage to VPLEX, ensure that mode 4 (ALUA) or mode 1 set during VPLEX initiator registration prior to device presentation. Don't try to change it after devices are already presented.

Note: Proper connectivity for active/passive and ALUA arrays now require 8 (eight) paths per Director per LUN for Active/Passive and ALUA. The previously supported minimal path configuration is no longer supported.

The next set of diagrams depicts both a four "active" path per Director per LUN "Logical" configuration and a four "active" path per Director per LUN "Physical" configuration. Both are supported configurations.

The commands used in the VPlexcli to determine the port WWNs and the ITLs used in the following diagram are:

- VPlexcli:/ll **/hardware/ports
- VPlexcli:/clusters/cluster-<cluster number>/storage-elements/storage-volumes/<storage volume name>>ll --full

As an example:

```
VPlexcli:/> ll **/hardware/ports

/engines/engine-1-1/directors/director-1-1-A/hardware/ports:
Name     Address                 Role        Port Status
-------  ----------------------  ----------  -----------
A0-FC00  0x5000144260abce00      front-end   up
A0-FC01  0x5000144260abce01      front-end   up
A0-FC02  0x5000144260abce02      front-end   up
A0-FC03  0x5000144260abce03      front-end   up
A1-FC00  0x5000144260abce10      back-end    up
A1-FC01  0x5000144260abce11      back-end    up
A1-FC02  0x5000144260abce12      back-end    up
A1-FC03  0x5000144260abce13      back-end    up
A2-FC00  0x5000144260abce20      wan-com     up
A2-FC01  0x5000144260abce21      wan-com     up
A2-FC02  0x0000000000000000      wan-com     down
A2-FC03  0x0000000000000000      wan-com     down
A3-FC00  0x5000144260abce30      local-com   up
A3-FC01  0x5000144260abce31      local-com   up
A3-FC02  0x0000000000000000      -           down
A3-FC03  0x0000000000000000      -           down


/engines/engine-1-1/directors/director-1-1-B/hardware/ports:
Name     Address                 Role        Port Status
-------  ----------------------  ----------  -----------
B0-FC00  0x5000144270abce00      front-end   up
B0-FC01  0x5000144270abce01      front-end   up
B0-FC02  0x5000144270abce02      front-end   up
B0-FC03  0x5000144270abce03      front-end   up
B1-FC00  0x5000144270abce10      back-end    up
B1-FC01  0x5000144270abce11      back-end    up
B1-FC02  0x5000144270abce12      back-end    up
B1-FC03  0x5000144270abce13      back-end    up
B2-FC00  0x5000144270abce20      wan-com     up
B2-FC01  0x5000144270abce21      wan-com     up
B2-FC02  0x0000000000000000      wan-com     down
B2-FC03  0x0000000000000000      wan-com     down
B3-FC00  0x5000144270abce30      local-com   up
B3-FC01  0x5000144270abce31      local-com   up
B3-FC02  0x0000000000000000      -           down
B3-FC03  0x0000000000000000      -           down
```

Figure 9 Backend port WWN identification

Running the long listing on the hardware/ports allows you to determine which WWN is associated with which VPLEX backend port.

```
VPlexcli:/clusters/cluster-1/storage-elements/storage-volumes/vnx-local_0358> ll --full
Name                      Value
------------------------  ------------------------------------------------------------
application-consistent    false
block-count               78643200
block-size                4K
capacity                  300G
description               -
free-chunks               []
health-indications        []
health-state              ok
io-status                 alive
itls                      0x5000144260abce10/0x5006016046abcdef/9,
                          0x5000144260abce11/0x5006016146abcdef/9,
                          0x5000144260abce12/0x5006016446abcdef/9,
                          0x5000144260abce13/0x5006016546abcdef/9,
                          0x5000144260abce10/0x5006016a46abcdef/9,
                          0x5000144260abce11/0x5006016b46abcdef/9,
                          0x5000144260abce12/0x5006016e46abcdef/9,
                          0x5000144260abce13/0x5006016f46abcdef/9,
                          0x5000144270abce10/0x5006016246abcdef/9,
                          0x5000144270abce11/0x5006016346abcdef/9,
                          0x5000144270abce12/0x5006016646abcdef/9,
                          0x5000144270abce13/0x5006016746abcdef/9,
                          0x5000144270abce10/0x5006016846abcdef/9,
                          0x5000144270abce11/0x5006016946abcdef/9,
                          0x5000144270abce12/0x5006016c46abcdef/9,
                          0x5000144270abce13/0x5006016d46abcdef/9
largest-free-chunk        0B
locality                  -
operational-status        ok
storage-array-name        EMC-CLARiiON-CKM001121110358
storage-volumetype        normal
system-id                 VPD83T3:6006016011e02a00108a177fabcde211
thin-rebuild              false
total-free-space          0B
use                       used
used-by                   [extent_vnx-local_0358_1]
vendor-specific-name      DGC
```


Figure 10 ITL nexus association

From the storage-volumes context you can select a sample volume and cd to that context. Running the ll --full command will show the ITLs.

In this example we have sixteen entries for this volume. This is a single engine VPLEX cluster connected to an Active/Passive or ALUA array. Even though this gives us eight paths per director for this volume only four paths go to the array SP that owns the volume. In either mode 1 or mode 4 (ALUA), the paths going to the other SP will not be used for I/O for that given set of LUNs. Only in the case of a trespass will they become active for those LUNs.

Note: All paths, whether active or not, will perform device discovery during an array rediscover. Over allocating the number of paths beyond the supported limits will have detrimental effects on performance and/or backend LUN provisioning.

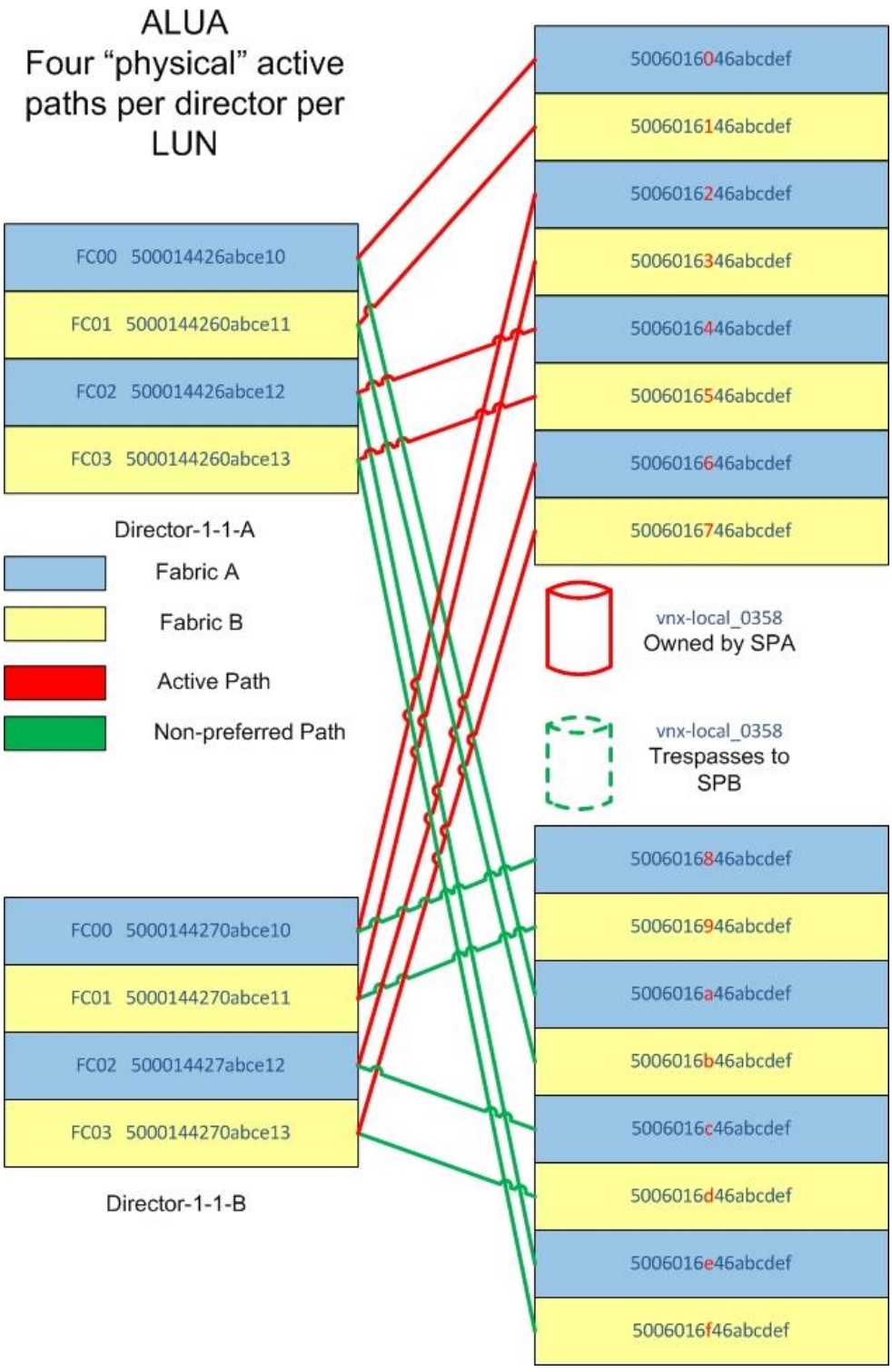

Figure 11 Physical path configuration

This drawing was developed from the output from the two commands shown above.

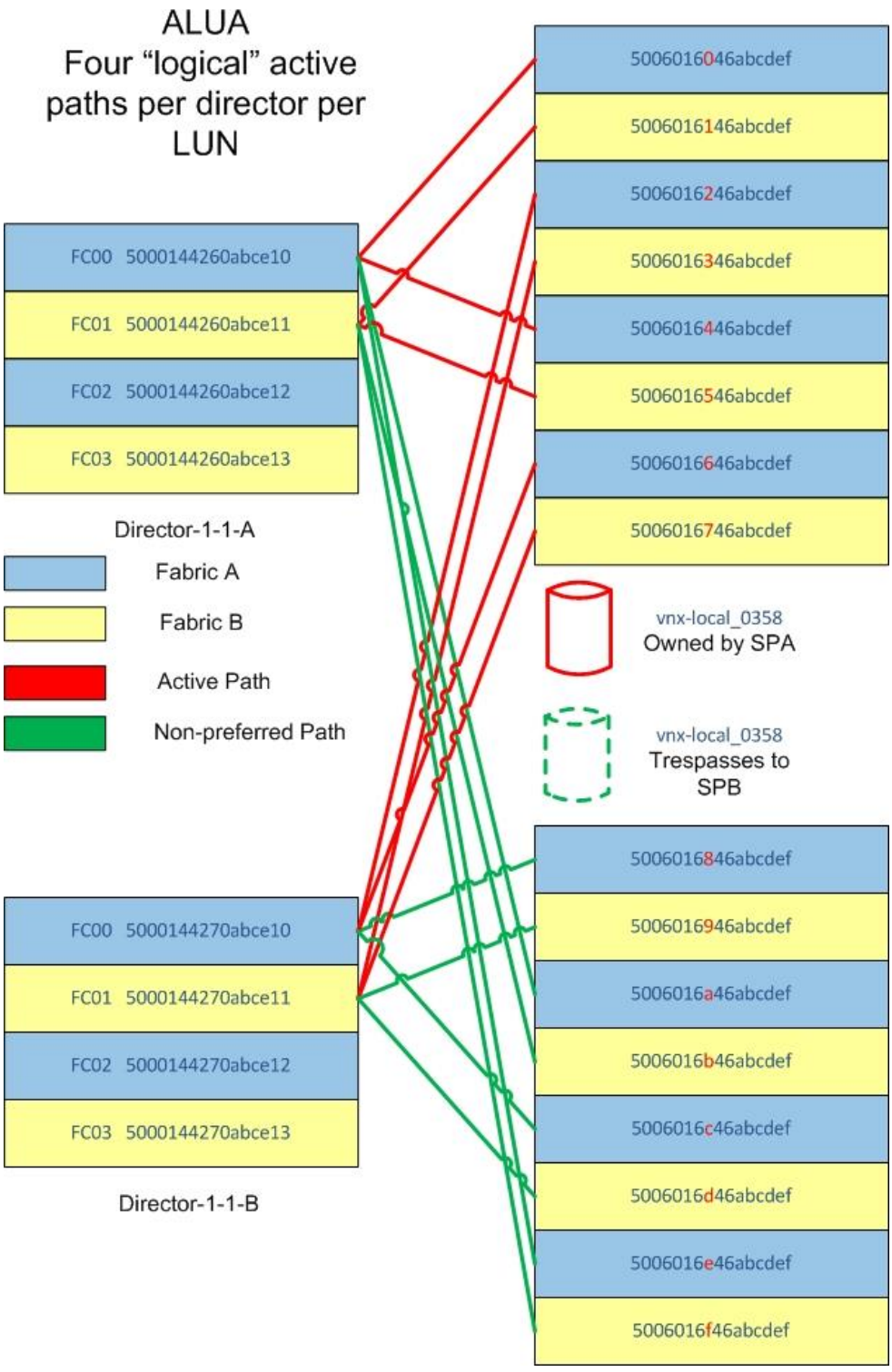

Figure 12 Logical path configuration

A slight modification from the previous drawing helps illustrate the same concept but using only two VPLEX backend ports per director. This gives us the exact same number of ITLs and meets the maximum supported limit as spreading across all four ports.

3.1.3 Additional array considerations

Arrays, such as the Symmetrix®, that do in-band management may require a direct path from some hosts to the array. Such a direct path should be solely for the purposes of in-band management. Storage volumes provisioned to the VPLEX should never simultaneously be masked directly from the array to the host; otherwise there is a high probability of data corruption. It may be best to dedicate hosts for in-band management and keep them outside of the VPLEX environment.

Storage volumes provided by arrays must have a capacity that is a multiple of 4 KB. Any volumes which are not a multiple of 4KB will not show up in the list of available volumes to be claimed. For the use case of presenting storage volumes to VPLEX that contain data and are not a multiple of 4K then those devices will have to be migrated to a volume that is a multiple of 4K first then that device presented to VPLEX. The alternative would be to use a host-based copy utility to move the data to a new and unused VPLEX device.

Remember to reference the EMC Simple Support Matrix, Release Notes, and online documentation for specific array configuration requirements. Remember to follow array best practices for configuring devices to VPLEX.

4 XtremIO connectivity

4.1 XtremIO overview

The Dell EMC XtremIO™ storage array is an all-flash system, based on a scale-out architecture. The system uses building blocks, called X-Bricks, which can be clustered together.

The system operation is controlled via a stand-alone dedicated Linux-based server, called the XtremIO Management Server (XMS). Each XtremIO cluster requires its own XMS host, which can be either a physical or a virtual server. The array continues operating if it is disconnected from the XMS but cannot be configured or monitored.

The XtremIO array architecture is specifically designed to deliver the full performance potential of flash, while linearly scaling all resources such as CPU, RAM, SSDs, and host ports in a balanced manner. This allows the array to achieve any desired performance level, while maintaining consistency of performance that is critical to predictable application behavior.

The XtremIO Storage Array provides an extremely high level of performance that is consistent over time, system conditions and access patterns. It is designed for high granularity true random I/O. The cluster's performance level is not affected by its capacity utilization level, number of volumes, or aging effects.

Due to its content-aware storage architecture, XtremIO provides:

- Even distribution of data blocks, inherently leading to maximum performance and minimal flash wear
- Even distribution of metadata
- No data or metadata hotspots
- Easy setup and no tuning
- Advanced storage functionality, including Inline Data Reduction (deduplication and data compression), thin provisioning, advanced data protection (XDP), snapshots, and more

Additional information about XtremIO can be found in the EMC VPLEX WITH XTREMIO 2.4 White Paper, located at EMC.com.

4.2 VPLEX connectivity to XtremIO

XtremIO is a true active/active array. Each Fibre Channel and iSCSI port on all Storage Controllers has access to all storage provisioned on all X-Bricks within the cluster. This is made possible due to the Infiniband connectivity between Storage Controllers. When deciding how to properly connect VPLEX backend ports to the XtremIO cluster simply follow VPLEX active/active array connectivity best practices. From a high-level perspective, you want to spread the paths from each VPLEX Director across XtremIO Storage Controllers balancing the workload evenly across the overall cluster. The following diagrams illustrate multiple combinations and should clearly show the concept. This concept should be easily adapted for additional configuration possibilities that are not shown.

Within the XtremIO provisioning, a single initiator group can access all storage within the cluster. It is not necessary to try and configure a single VPLEX cluster into two separate initiator groups. The XtremIO provisioning uses a much simpler approach than most other arrays. Most arrays use a masking container called a masking view, storage view or storage group as a few examples, and these containers typically contain the objects such as LUNs or devices, initiators and the ports on the array that the administrator wishes to use on the array for access to those devices from those initiators. XtremIO simply allows you to

create initiator groups which are a simple grouping of initiators from a single host or clustered host which you want to have access to any given volume. When provisioning volumes you simply connect the volume to the initiator group. You do not select which port you want to allow access to that volume from the initiators in that initiator group. Any port can access any volume.

The following illustrations demonstrate best practice recommendations but are not limited to these specific configurations only. We are providing several combinations from which you should be able to extrapolate a design from for your customer's configuration needs.

4.2.1 Single engine/single X-Brick

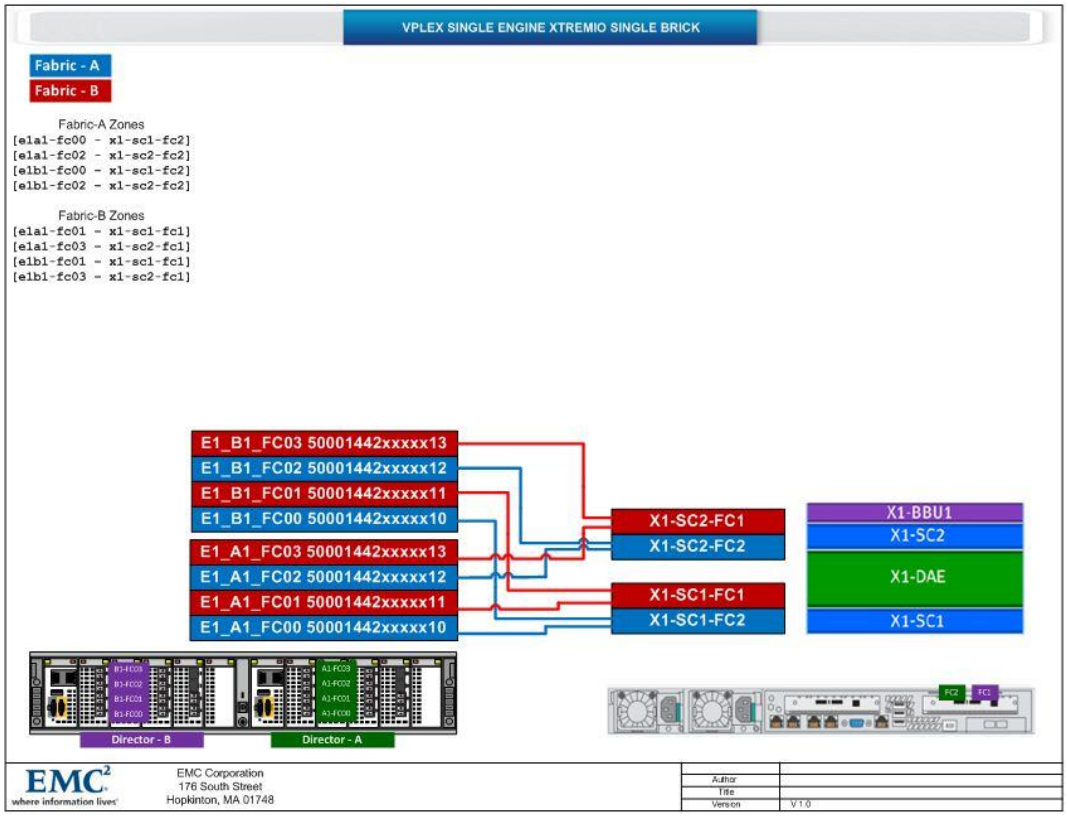

Figure 13 VPLEX single engine – XtremIO single X-Brick

This illustration is a VPLEX single engine cluster connected to an XtremIO single X-Brick™ cluster. This configuration meets the VPLEX backend connectivity rule recommending **four** paths per Director per Storage Volume. Each VPLEX Director and each XtremIO Storage Controller are connected to both fabrics as depicted by the red and blue colorings. Both VPLEX Directors are cross connected to both XtremIO Storage Controllers on both fabrics. This connectivity allows for the highest level of availability.

This illustration is mapping the VPLEX backend ports to the XtremIO ports on a 2:1 basis. Based on total available bandwidth, you may consider mapping the ports on a 1:1 basis thereby leaving two ports available on each VPLEX Director for future scale of the XtremIO cluster. This would reduce the port count on the fabrics at time of initial deployment and still give the same maximum throughput.

4.2.2 Single engine/dual X-Brick

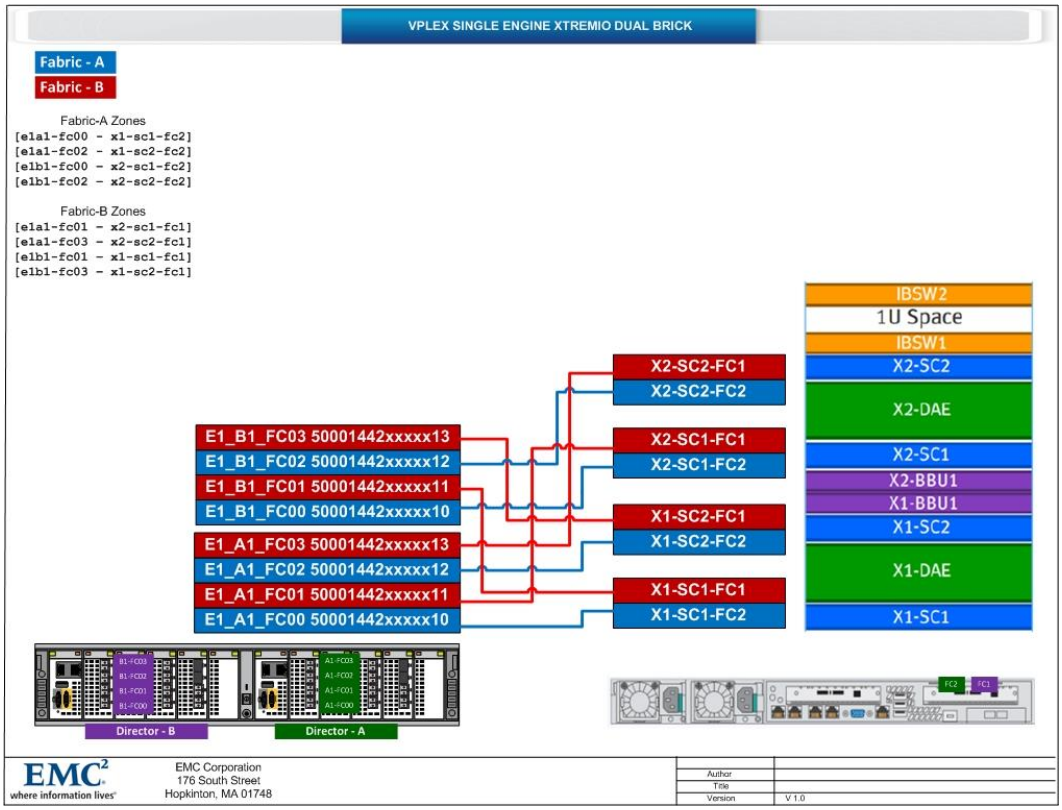

Figure 14 VPLEX single engine – XtremIO dual X-Brick

This example follows the four paths per VPLEX Director per Storage Volume rule as recommended for performance. The ports are mapped 1:1 between the two clusters allowing for maximum throughput using all available ports on both clusters.

4.2.3 Single engine/quad X-Brick

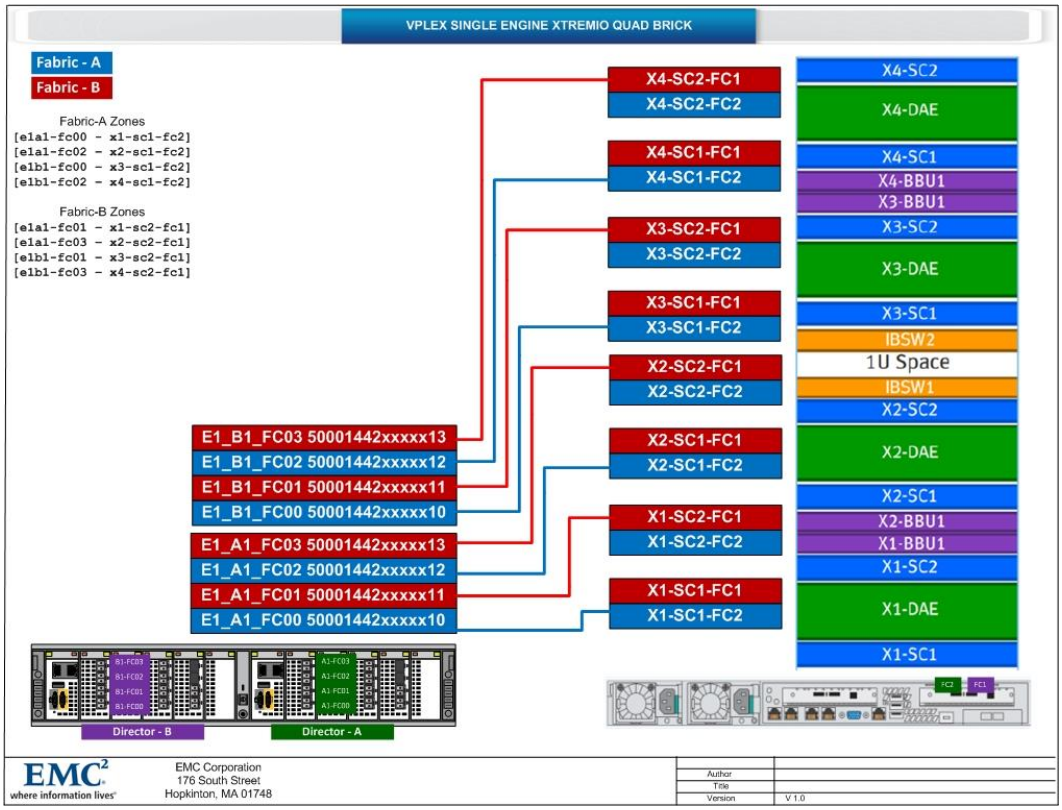

Figure 15 VPLEX single engine – XtremIO quad X-Brick

This illustration demonstrates one possibility of connecting a single engine VPLEX cluster to a quad X-Brick XtremIO cluster. This configuration meets the VPLEX best practices of four paths per VPLEX Director per Storage Volume. This configuration also demonstrates a common performance best practice of spreading across all available resources evenly.

4.2.4 Dual engine/single X-Brick

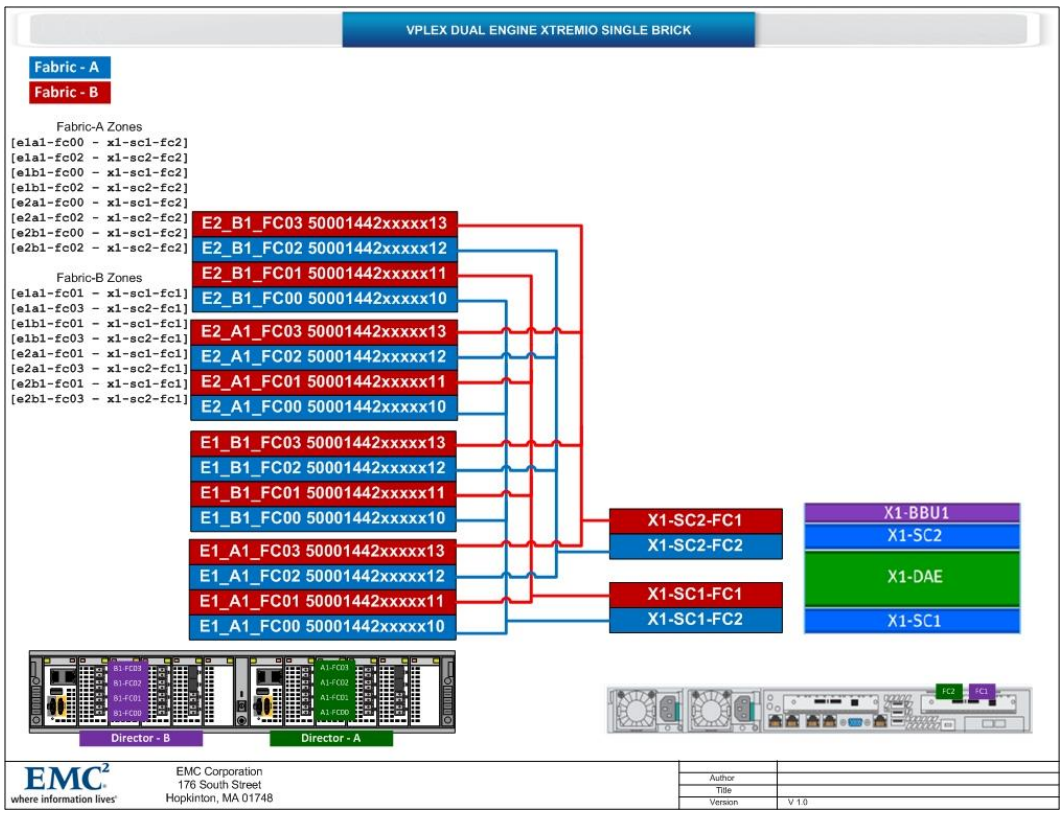

Figure 16 VPLEX dual engine - XtremIO single X-Brick

This illustration demonstrates connectivity of a dual engine VPLEX cluster to a single X-Brick XtremIO cluster.

Due to the limitation of each Storage Controller for the X-Brick having only two Fibre Channel ports we have to double up on the paths from VPLEX in order to achieve the four paths per Director per Storage Volume. You could drop to the minimum VPLEX ndu requirement of two paths per Director per Storage Volume thereby only using two Fibre Channel ports on each VPLEX Director and create a 1:1 port mapping between clusters. The total available bandwidth would remain the same. Performance testing has not been performed comparing these two possible configurations so we can't say if you would experience any difference or not in overall performance.

The advantage to going to the minimum configuration would be in that you would be saving eight ports altogether on the fabrics. Future scale of the XtremIO would also be easier.

4.2.5 Dual engine/dual X-Brick

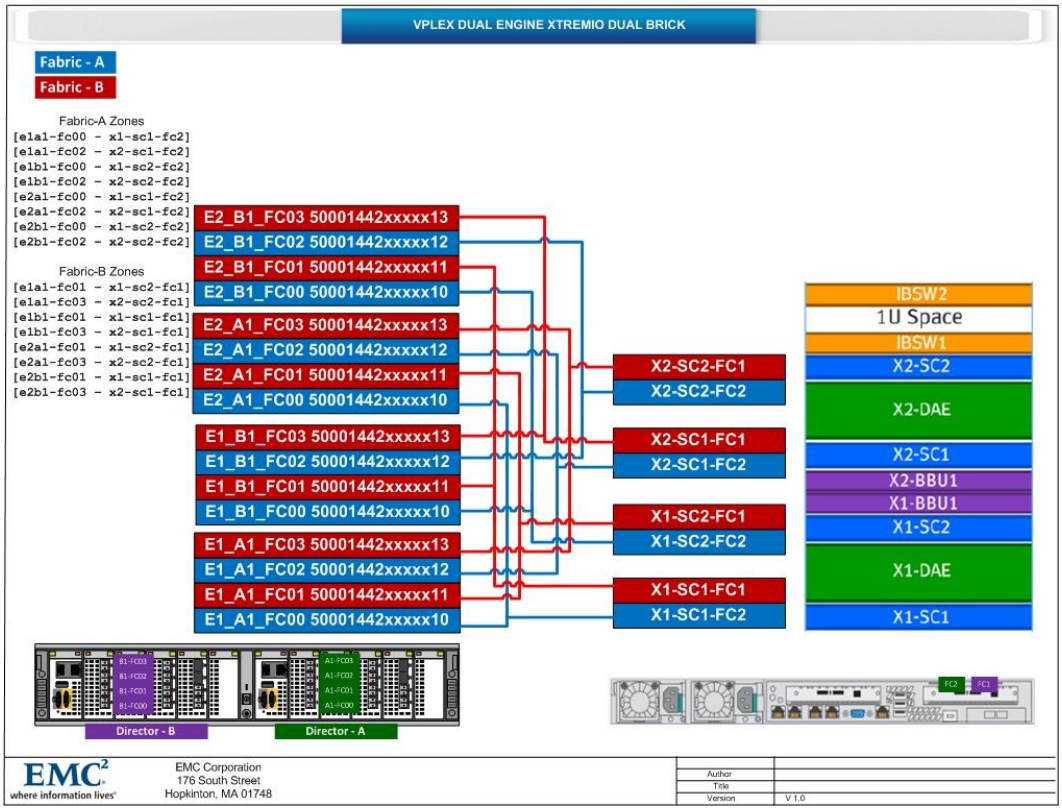

Figure 17 VPLEX dual engine - XtremIO dual X-Brick

This illustration demonstrates connectivity between a dual engine VPLEX cluster and a dual X-Brick XtremIO cluster.

This illustration is mapping the VPLEX backend ports to the XtremIO ports on a 2:1 basis. Based on total available bandwidth, you may consider mapping the ports on a 1:1 basis thereby leaving two ports available on each VPLEX Director for future scale of the XtremIO cluster. This would reduce the port count on the fabrics at time of initial deployment and still give the same maximum throughput.

4.2.6 Dual engine/quad X-Brick

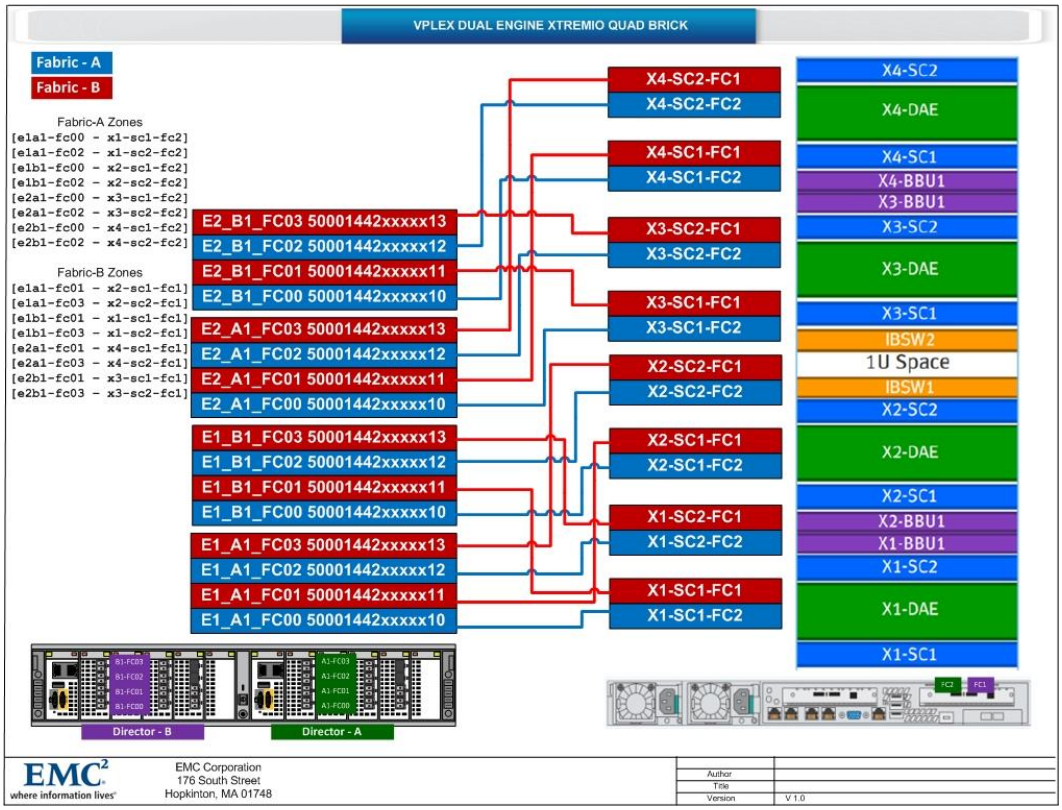

Figure 18 VPLEX dual engine - XtremIO quad X-Brick

This illustration demonstrates a VPLEX dual engine cluster connectivity to an XtremIO quad X-Brick cluster.

This example follows the four paths per VPLEX Director per Storage Volume rule as recommended for performance. The ports are mapped 1:1 between the two clusters allowing for maximum throughput using all available ports on both clusters.

4.2.7 Quad engine/single X-Brick

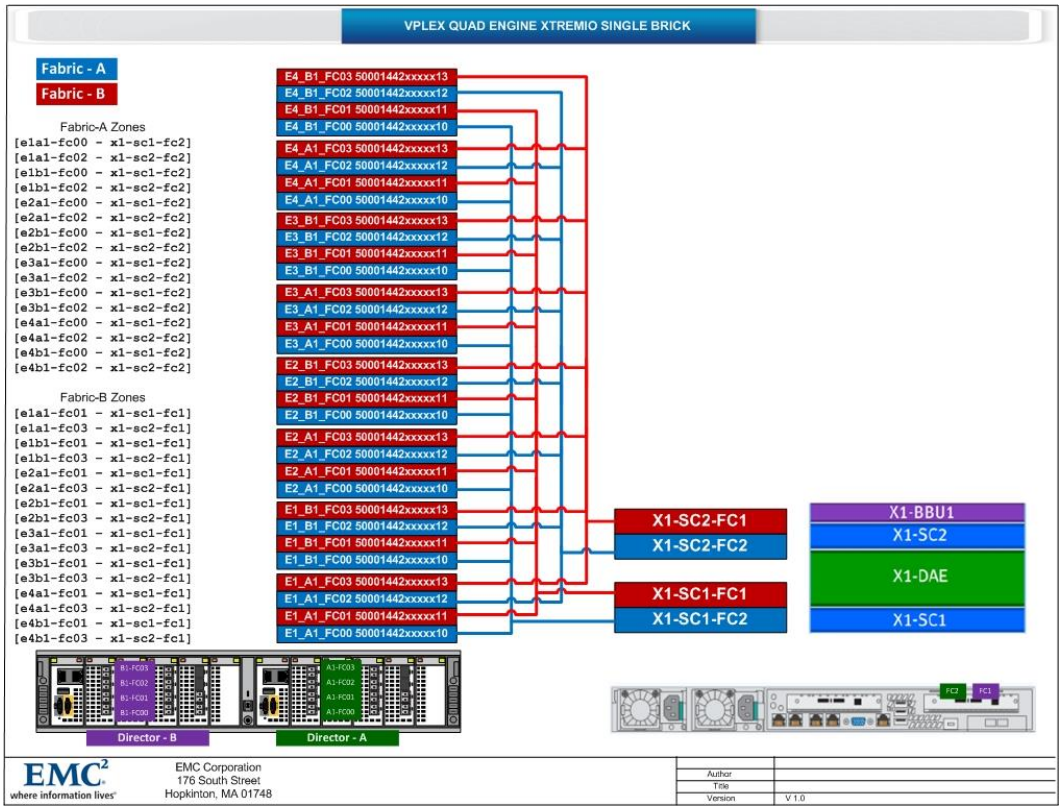

Figure 19 VPLEX quad engine - XtremIO single X-Brick

This illustration demonstrates connectivity of a VPLEX quad engine cluster to an XtremIO single X-Brick cluster.

This configuration meets the VPLEX backend connectivity rule recommending **four** paths per Director per Storage Volume. The ports are mapped 8:1 and may not be desirable from a cost point of view. Reducing to a minimum VPLEX ndu requirement of two paths per Director per Storage Volume would reduce the cost by reducing the port count required on the fabrics by a total of 16 ports overall. This would also leave these ports available for future scale on the XtremIO cluster. The total available bandwidth is dictated by the throughput of the available four ports on the XtremIO cluster in this configuration.

4.2.8 Quad engine/dual X-Brick

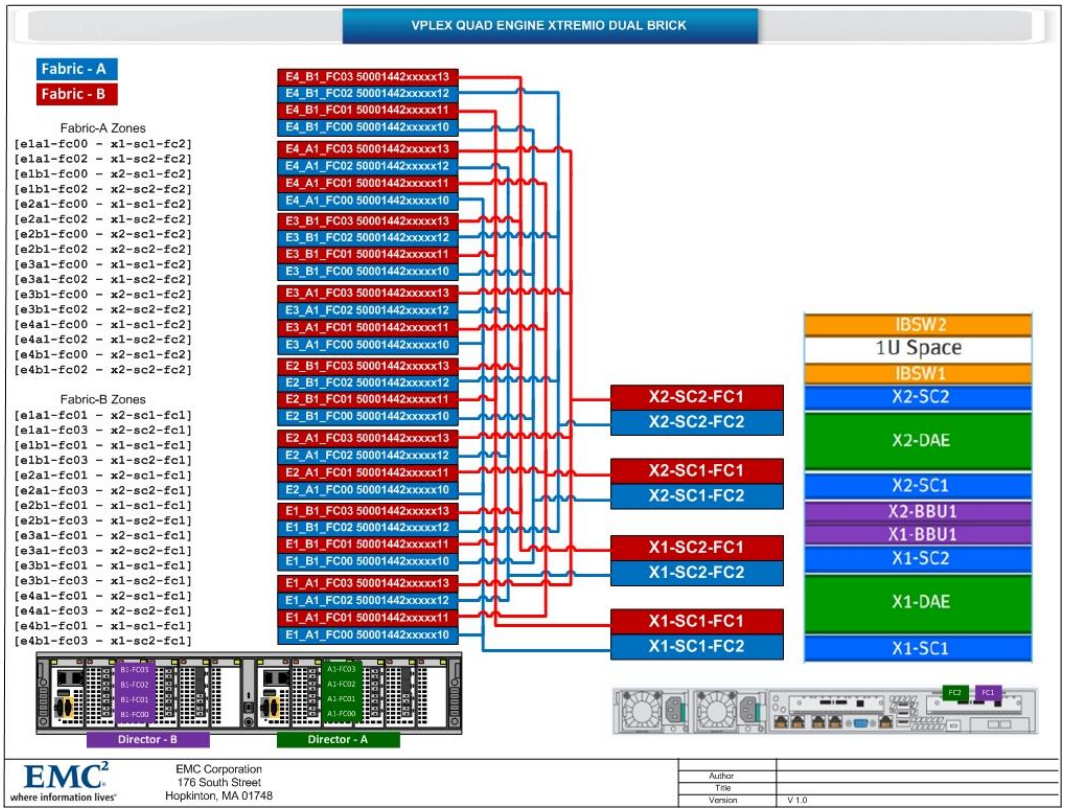

Figure 20 VPLEX quad engine - XtremIO dual X-Brick

This illustration demonstrates the connectivity of a quad engine VPLEX cluster to a dual X-Brick XtremIO cluster.

This configuration meets the VPLEX backend connectivity rule recommending **four** paths per Director per Storage Volume. The ports are mapped 4:1 and may not be desirable from a cost point of view. Reducing to a minimum VPLEX ndu requirement of two paths per Director per Storage Volume would reduce the cost by reducing the port count required on the fabrics by a total of 16 ports overall. This would also leave these ports available for future scale on the XtremIO cluster.

4.2.9 Quad engine/quad X-Brick

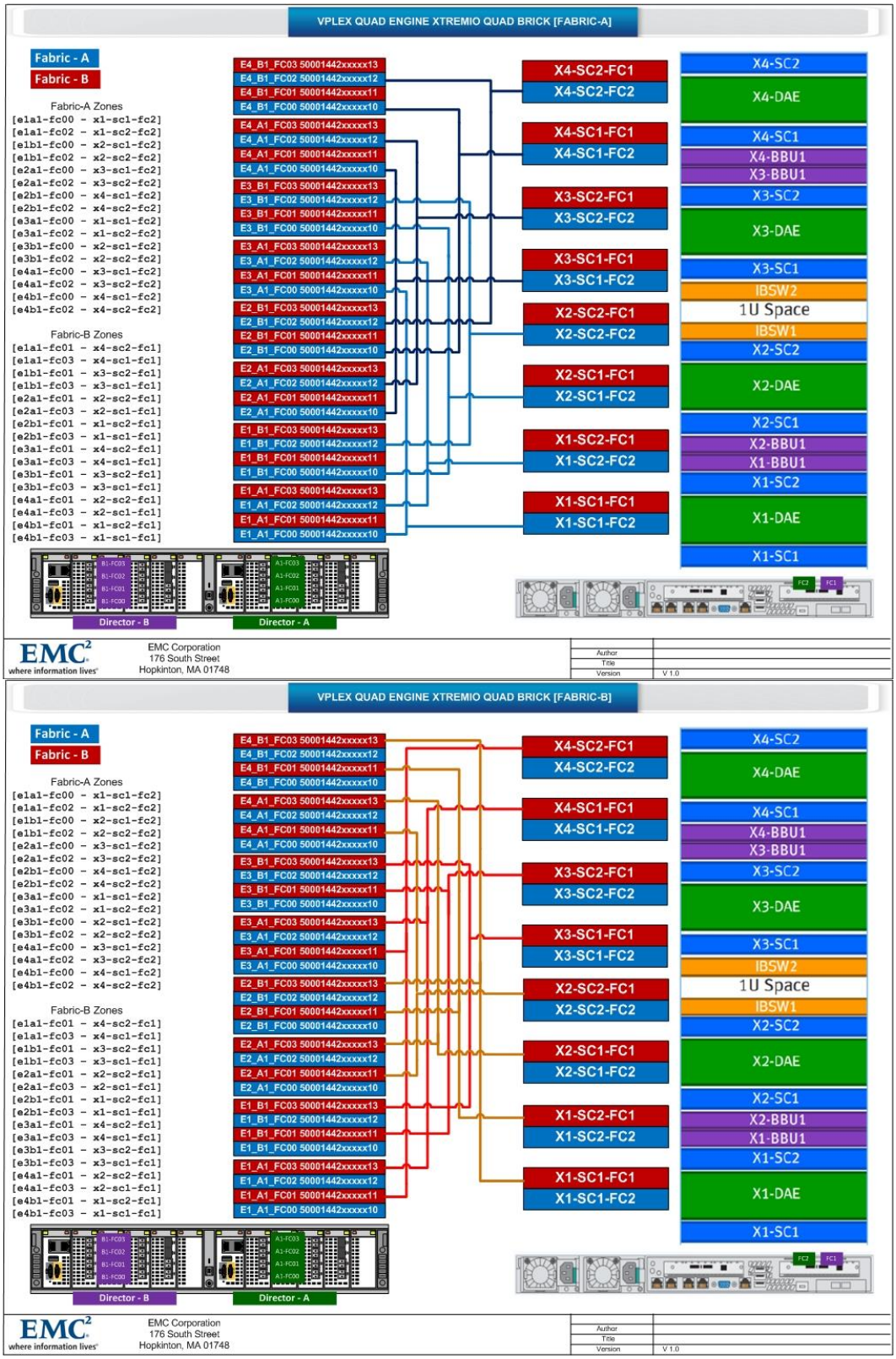

Figure 21 VPLEX quad engine - XtremIO quad X-Brick (Fabric A top Fabric B bottom)

The previous two illustrations combined demonstrate the connectivity between a quad engine VPLEX cluster and a quad X-Brick XtremIO cluster. The illustrations are broken up into two parts showing Fabric connectivity on the top and Fabric B connectivity on the bottom.

This configuration meets the VPLEX best practices for four paths per Director per Storage Volume. The ports are mapped on a 2:1 basis which allows dropping to the minimal configuration requirements on VPLEX as required by the ndu pre-check. This also allows adding future capacity on the XtremIO as well as reducing the port count on the fabrics.

4.3 XtremIO 6 X-Brick connectivity

The purpose of this section is to define the best practice to connect a quad engine VPLEX to a 6 X-Brick XtremIO array. This paper defines two distinctly different approaches with full explanation behind each design consideration.

Note: XtremIO X2 supports only up to four X-Bricks.

4.3.1 Solution 1

4.3.1.1 Expansion from VPLEX quad and XtremIO quad brick

The specific issue with coming up with a connectivity design for a quad engine VPLEX connected to a 6 X-brick XtremIO array is basic math. VPLEX has 32 backend ports and a 6 X-Brick XtremIO array has 24 ports.

The previous section documentation illustrates a quad engine VPLEX connected to a 4 X-Brick XtremIO using a 2:1 zoning configuration. Each port on the XtremIO is connected to two VPLEX backend ports. This pairing is spanning the VPLEX engine groups 1&2 and 3&4 for every XtremIO port. Host connectivity best practices dictate that you connect a host to 4 VPLEX directors across two engines. One group of hosts would be connected to VPLEX engines 1&2 and the next host would be connected to engines 3&4.

Additionally, all XtremIO storage controllers are connected via InfiniBand and all LUNs span all X-Bricks. There is no LUN affinity to any given X-Brick therefore there is no storage controller port that has a performance gain based on location.

Ports queue depth is commonly found to be an issue and throttling is necessary at the host to avoid overrunning the queue. XtremIO handles high I/O workloads very well therefore the queue depth throttle is recommended to be left at max of 256 as the more you hammer the port, the higher the performance. Based on this, if you were to take and upgrade a 4 X-Brick XtremIO currently connected to a Quad engine VPLEX to a 6 X-Brick configuration then there really isn't a necessity to rezone the VPLEX backend ports for connectivity to the additional 4 storage controllers as this would possibly add unnecessary risk.

It is not critical to use all ports if there are hosts outside of VPLEX, so 8 ports could remain free. If all hosts go through VPLEX, the extra ports could be utilized in the following fashion:

- Backup servers
- RecoverPoint
- Native Replication

To sum it up, first recommendation is to use the connectivity illustrated in the best practices for a 4 X-Brick configuration. This recommendation is also very well suited for array expansion when the customer decides they need additional capacity going from a quad X-Brick configuration to a 6 X-Brick configuration.

Please see below two illustrations showing zoning and cabling on both fabrics for this configuration.

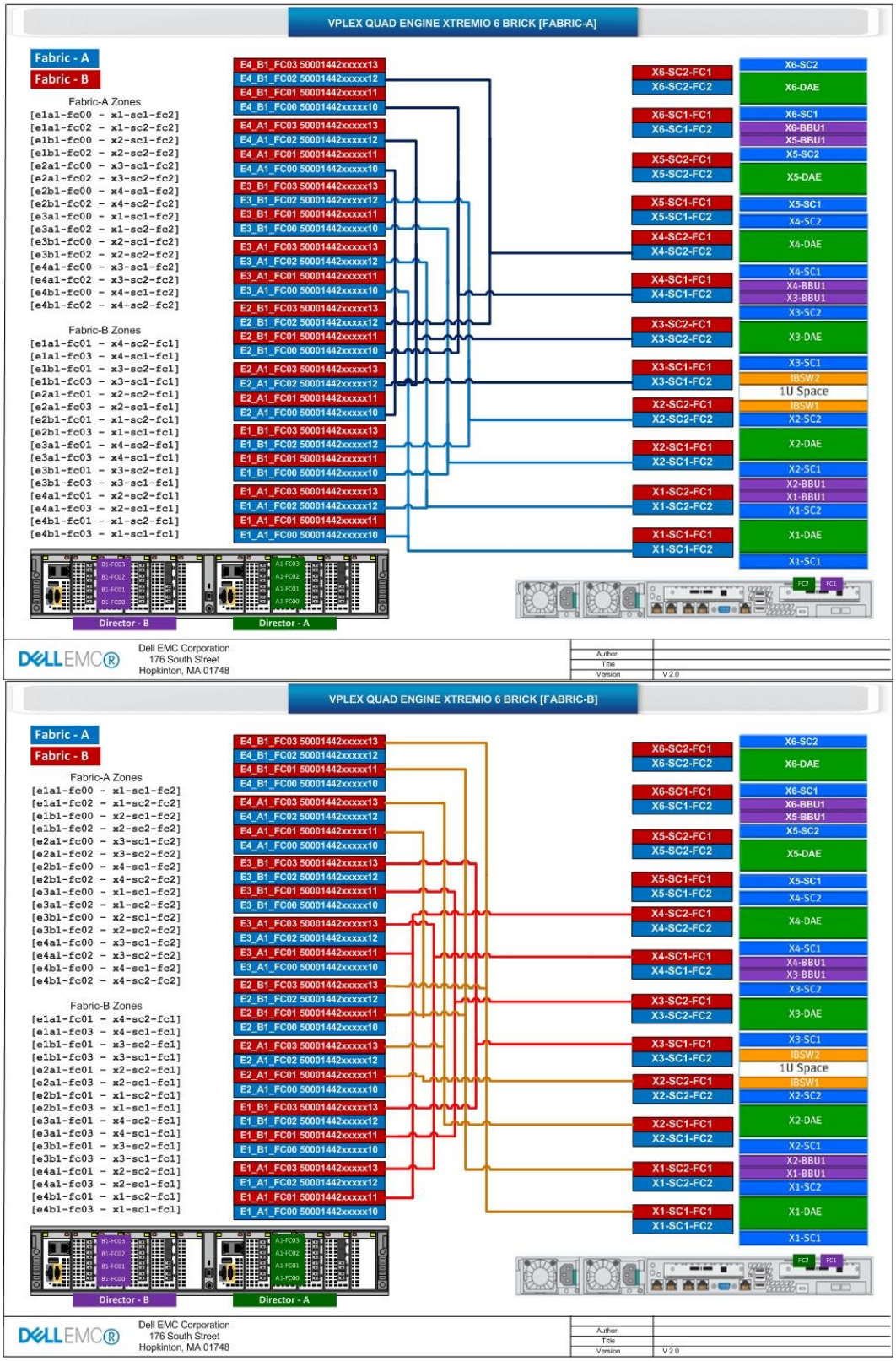

Figure 22 VPLEX quad engine - XtremIO 6 X-Brick (Fabric A top Fabric B bottom)

4.3.2 Solution 2

4.3.2.1 Net new install of a VPLEX quad engine and XtremIO 6 brick array

This solution is designed to utilize all ports on the XtremIO 6 X-brick configuration. Again, it is mathematically impossible to perfectly balance the port connectivity from a Quad Engine VPLEX to a 6 X-Brick XtremIO. The VPLEX has 32 ports and the XtremIO 6 X-brick configuration has 24 ports. Also, the term “perfectly balanced” does not necessarily apply to the actual cabling but rather the I/O throughput which is strictly driven by the hosts attached to the VPLEX.

Following VPLEX host connectivity best practices, each host should be connected to four different directors spanning two engines. One host group will be connected to VPLEX engines 1&2 and another host group will be connected to engines 3&4.

This solution is based on the host connectivity design in that the engines 1&2 are connected on a 1:1 port basis while engines 3&4 are connected on a 2:1 port basis. This means that engines 1&2 will have double the throughput capability as engines 3&4 so this allows the host connectivity to have an advantage with high demanding hosts to be connected to engines 1&2. Hosts with lesser demand can be attached to engines 3&4.

Please be aware that this design does not leave any available ports for use with anything outside of the control of VPLEX. It is considered a best practice to avoid sharing port resources with solutions outside of VPLEX as that will create a workload imbalance that is outside of the control of VPLEX.

Note: All other VPLEX design requirements remain the same such as: “All LUNs must be seen on all directors”.

Please see below illustration for zoning and cabling design:

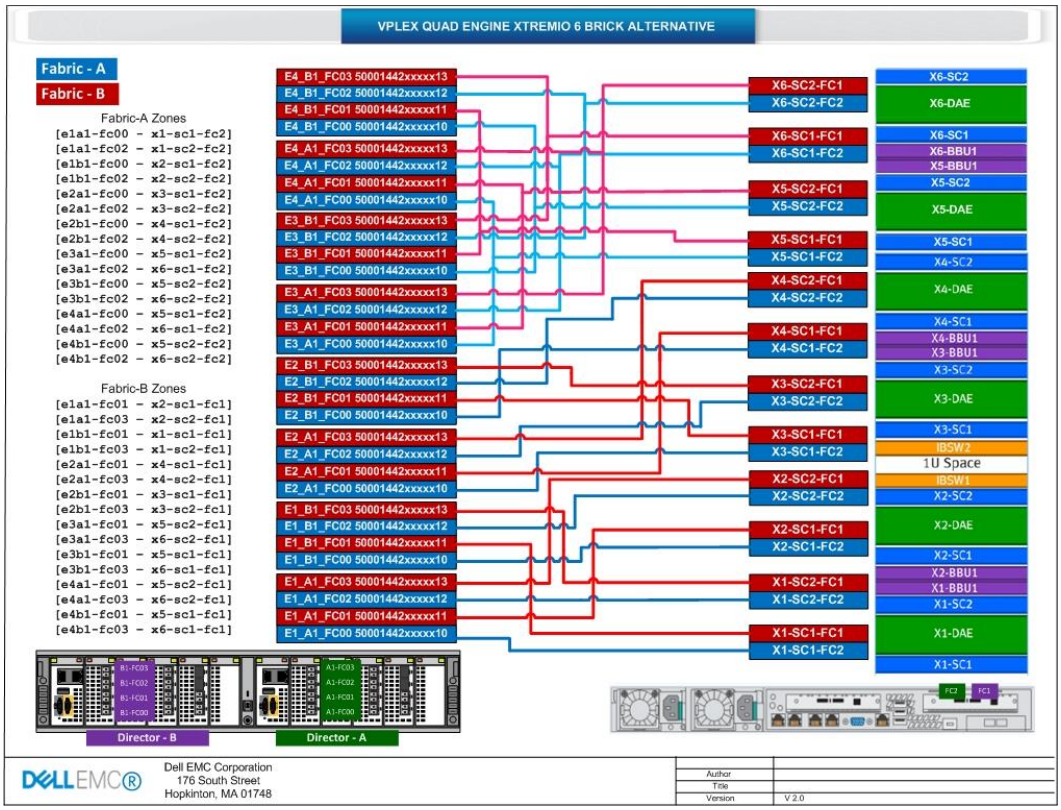

Figure 23 VPLEX quad engine - XtremIO 6 X-Brick

5 PowerStore and Dell EMC Unity XT

5.1 Metro node feature

Metro node is a new feature for the PowerStore or Dell EMC Unity XT bundle. The bundle is a soft bundle (co-sell) that customers can order when they are ordering either of these arrays. The underlying technology is based on the existing VPLEX technology but has significant differences. This bundle will provide ease of ordering capabilities (configuring of the solution in OSC via Guided Journey) for an active/active solution for metro synchronous replication for PowerStore and Dell EMC Unity XT customers. Today, metro node supports up to two PowerStore T model or one PowerStore X model appliances with 32/16/8 Gb Fibre Channel or 16/8/4 Gb Fibre Channel I/O modules in a cluster as what is referred to as a backend storage array from the metro node perspective.

Note: Metro node only supports block storage.

These arrays are considered active/active due to the high-speed data transfer from the non-preferred controller to the preferred controller. Additionally, prior arrays such as VNX had host setting to identify the array to the host as ALUA mode for this purpose. PowerStore and Dell EMC Unity XT are ALUA by default which means that when placed behind metro node it is responding to metro node by SCSI response that tells metro node that it is ALUA. You do not have to specify ALUA mode when registering host initiators. Metro node, in turn, utilizes the backend pathing the same as identified in the active/passive and ALUA section of this document.

Both Dell EMC Unity XT and PowerStore are built on a 2U platform where each node within the engine or appliance may be configured with one or two four-port Fibre Channel I/O modules for host connectivity. The following illustrations will depict both platforms.

Note: It is now a requirement for all-flash arrays is to use the four active path / four passive path configuration as the previously supported minimal configuration has been too problematic as it was only intended for test environments or small workloads.

5.2 PowerStore

The PowerStore hardware platform is created with a focus on performance, reliability, and modular flexibility. Designed from the ground up to leverage next-generation innovations such as NVMe and Intel® Optane™ Storage Class Memory (SCM) to deliver higher IOPS at lower latencies for real-world workloads. This offers all the performance and expansion headroom that is needed to ensure long-term value through multiple solution life cycles. PowerStore helps simplify and consolidate diverse infrastructures with high-performance multi-protocol network I/O options and single-architecture support for block, file, and VMware® vSphere® Virtual Volumes™ (vVols), transforming operations for both traditional and modern workloads.

Figure 24 PowerStore Appliance

Each PowerStore appliance consists of two nodes. Each node includes dual-socket CPUs and an Intel chipset which is used to offload the inline data compression. Each hardware configuration in the PowerStore platform is available in either a PowerStore “T” model or PowerStore “X” model configuration. The base hardware, processors, memory, media support, and services such as deduplication and compression are identical between the two models.

PowerStore has the unique capability of clustering appliances together to help balance the workload across the appliances. This is done by the software automatically selecting which appliance is most optimal to provision a volume from. This will help spread the I/O out evenly. You do have the option of manually selecting which appliance to provision it on. Also, PowerStore can move volumes between appliances. A target appliance is selected to migrate to but the target volume is inaccessible until the data transfer has completed. This presents a problem to VPLEX as the volume may retain its original identity but the paths accessing it have changed. As you are aware, anything that changes outside of VPLEX control is not immediately recognized by VPLEX therefore you are in a potential DU situation, at least temporarily.

The storage behind a PowerStore Appliance is specific to that appliance only. It is not accessible from another appliance until migrated. VPLEX will not pick up the migrated volume automatically therefore you must perform an array rediscover after the migration. This approach would require a host outage as the device would be inaccessible during this phase.

Note: A better approach would be to provision a new device from the target appliance and use the VPLEX Data Mobility feature to move the data to that new volume. This approach would be completely non-disruptive. You would then follow the teardown procedure to remove the source device on the original appliance to free up the capacity.

5.3 Dell EMC Unity XT

The purpose-built Dell EMC Unity system is offered in multiple physical hardware models in both Hybrid configurations and All Flash configurations. Metro node is a feature offering for the All Flash models. For All Flash systems, the platform starts with the Dell EMC Unity 300F and scales up to the Dell EMC Unity XT 880F. The models share similarities in form factor and connectivity, but scale in differently in processing and memory capabilities.

For more information about the Dell EMC Unity XT hardware models, see the Dell EMC Unity hardware guide:

https://www.dellemc.com/en-us/collaterals/unauth/technical-guides-support-information/products/storage/docu69319.pdf

5.4 Metro node connectivity

Within each metro node cluster are two sets of private networks. One set is a redundant internal management IP network between cluster nodes color coded in green and purple. These Ethernet cables are direct connected to the corresponding ports between nodes. The second set is a redundant internal Ethernet based Twin-AX network which provides director to director communications solely within the metro node cluster (intra-cluster). This internal redundant Twin-AX network is commonly referred to as LOCAL COM. A Local COM network simply connects corresponding Local COM ports together with Twin-AX cables direct connected. **At no time should any of these internal networks be connected to customer networks.**

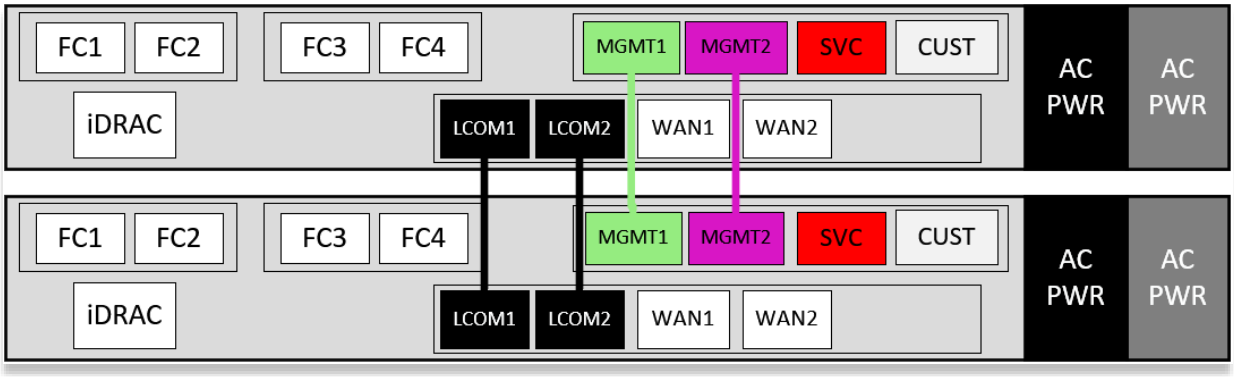

Figure 25 **LCOM** , **MGMT1** & **MGMT2** ports

Each node has an iDRAC port as well as service ports and customer network ports. The iDRAC port will be disabled from the factory. The SVC (service) ports will be used for connecting to the service laptop on IP address 128.221.252.2 using the service login. Follow the installation and configuration guide when performing initial install as it will direct you to which port to connect to during the procedure. After completing the installation, day to day operation may be performed from either node using the CUST (customer) ports.

The CUST ports will be connected to the customer network using customer supplied addresses. This port is used for day-to-day operations as well as file transfer and VPN secure connections between all nodes across a Metro.

5.5 Backend connectivity

Both PowerStore and Unity XT arrays are viewed by metro node as ALUA arrays based on SCSI response data and therefore are required to follow the four active, four passive path connectivity rules. This rule states that both nodes of the metro node must each have four active and four passive paths to all volumes provisioned from the array. The port layout on the metro node provide two backend ports per node so in order to abide by this rule, each port must be zoned to two ports on each array controller or DPE. At the array, the volume is owned by one controller and the ports for this controller are viewed by metro node as the preferred path or “active” path. Anytime the volume trespasses to the other controller the non-preferred paths or “passive” paths will become the “active” paths. This feature may be used to load balance the activity at the controller level allowing for balanced throughput across all eight paths.

All backend paths are required to be connected to the switches prior to running the Ansible script for installation and configuration. The script will fail if it doesn’t see link status up. Connecting to the switch but having the port persistent disabled will cause the script to fail also.

Zoning will be performed after the initial script is run successfully and you will be required to zone, mask and provision four metadata devices before continuing.

PowerStore is unique in that it allows multiple appliances to be configured as a single array. Each appliance manages its own storage and that storage is not accessible via any other appliance. Metro node sees each of these appliances as a separate array and backend pathing must be configured as such.

- Dual fabric designs are considered a best practice
- Back-end port to array direct connect is not supported based on the ITL (Initiator/Target/LUN) path count requirements
- Metro node FC I/O modules are 32Gb capable but are initially only supported in 7.0 at 16Gb
  - Manually set the switch interface to 16Gb for each metro node port connectivity
  - 32Gb connectivity is scheduled for support with 7.1
- The back-end I/O modules on each metro node director should have a minimum of two physical connections one to each fabric (required)
- Each metro node director must have eight logical paths to the storage device or LUN with four going to the active controller and four to the passive controller (required)
- Each metro node back-end port must have at least two paths to each controller at the array (required for NDU)

5.6 Dell EMC Unity XT and single-appliance PowerStore connectivity

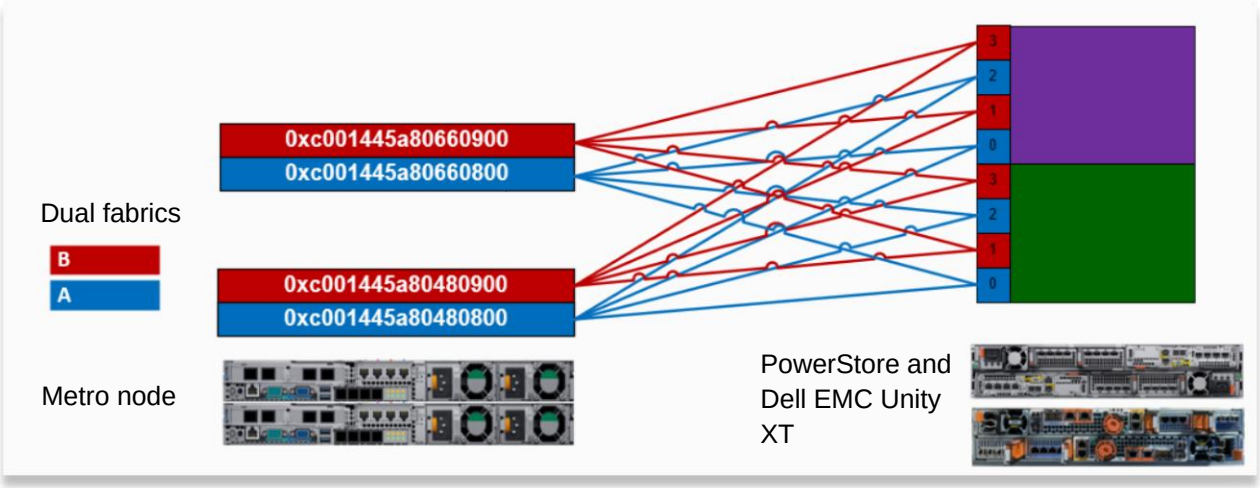

Figure 26 Dell EMC Unity XT and PowerStore (single appliance) dual fabric connectivity

This is based on dual fabric best practices as well as VPLEX best practices for active/passive and ALUA array connectivity. This illustration does not show the fabric switches but just the endpoint connectivity. Overall general best practice is to have dual fabrics and each component must be equally connected to both fabrics. This allows each port in the metro node to connect to both controllers at the array. This illustration shows the four active paths per node per volume rule of connectivity with an emphasis on “active”. Some volumes may be owned by one controller and other volumes owned by the other controller therefore some paths are active and some passive for one set of volumes but are the opposite for volumes owned by the

other controller. This results in all paths having active I/O. The best practice for volume ownership is to have the I/O workload load balanced as opposed to simply splitting the total volume count between the controllers.

Note: Backend array direct connect is not supported.

5.7 PowerStore dual-appliance connectivity

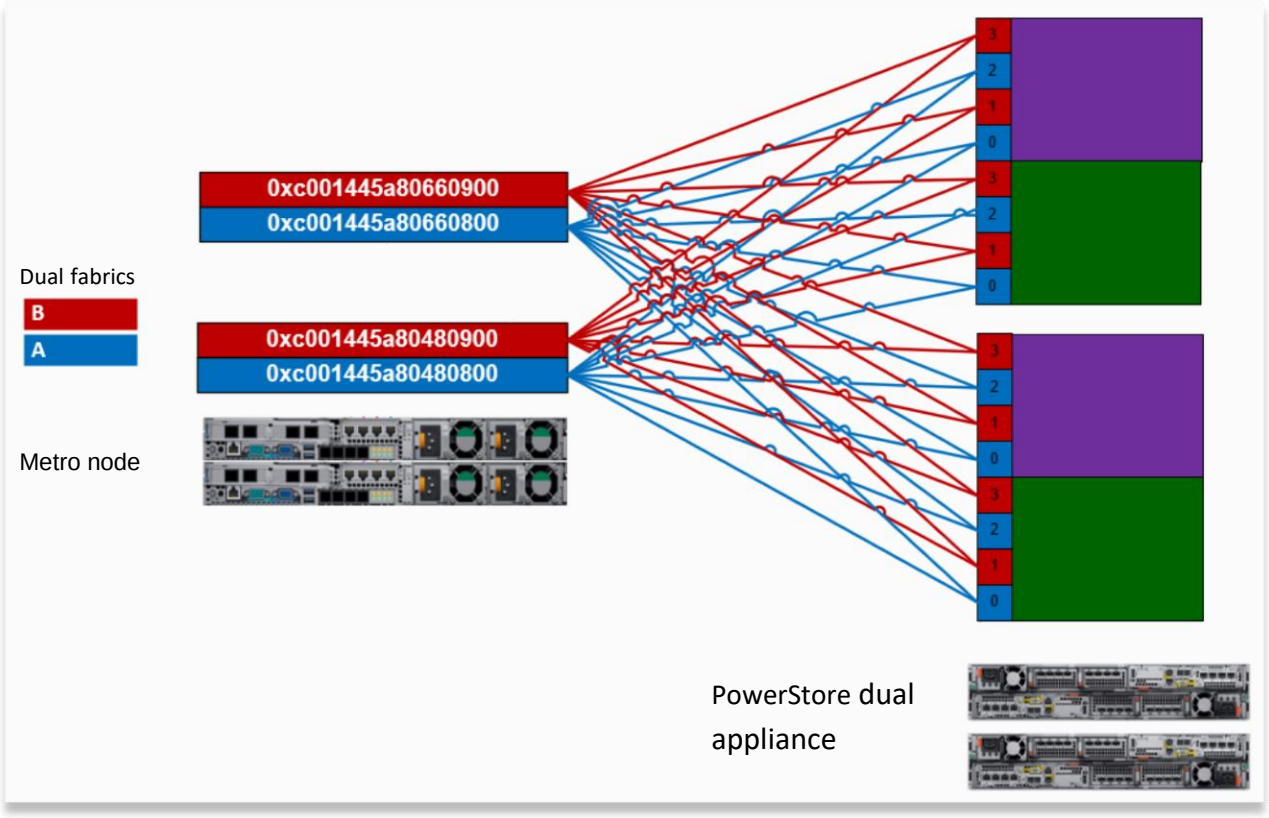

Figure 27 PowerStore dual appliance connectivity

PowerStore has the unique ability to configure multiple appliances together as a single array however volume access is limited to the appliance it is associated with for I/O data access. The advantage to clustering appliances offers the ease of use of volume creation where you may allow the array intelligence to determine best placement of volumes with the consideration around workload balancing. Additionally, it also allows for volume migration between appliances when not under metro node. Best practices for volume migration for volumes under metro node control is to use the Data Mobility feature within metro node otherwise a rolling reboot of metro node will be required to clear up the source volume. Please follow this design concept for all additional appliances that may be added to the PowerStore cluster.

Note: PowerStore Appliances must be presented to all VPLEX directors equally as per standard VPLEX requirements.

5.8 Frontend/host initiator port connectivity

Metro node is seen by the host as the array regardless of what physical array sits behind the metro node. Metro node is a true active/active array even if the backend array is ALUA. For zoning purposes, metro node is the endpoint and metro node frontend ports will present WWPN targets to zone to for the host connectivity.

- Dual fabric designs are considered a best practice
- The front-end I/O modules on each metro node director should have a minimum of two physical connections one to each fabric (required)

- Each host HBA port should have at least one path to an A director and one path to a B director on each fabric for a total of four logical paths (required for NDU)
- Multipathing or path failover software is required at the host for access across the dual fabrics
- Each host should have fabric zoning that provides redundant access to each LUN from a minimum of an A and B director from each fabric
- Four paths from host to metro node are required for NDU
- Using the skip option for the NDU start command would be required for host connectivity with less than four paths and is not considered a best practice

Please refer to hardware diagram for port layout.

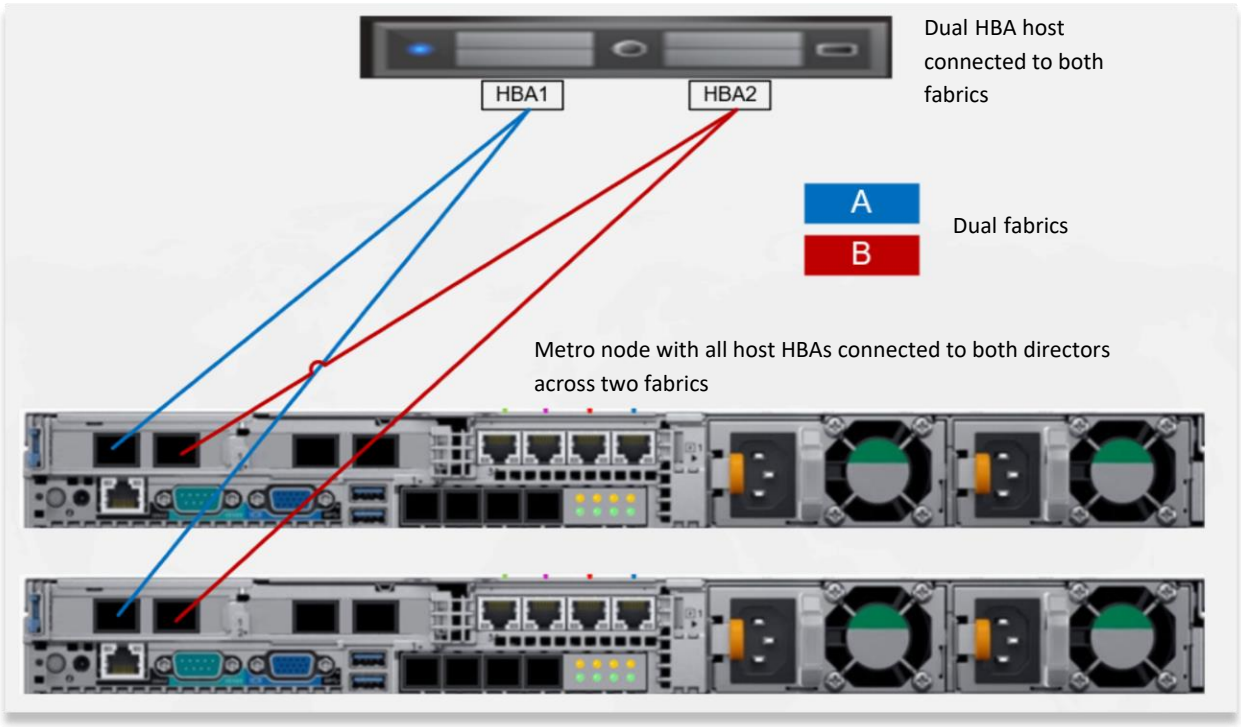

Figure 28 Host connectivity to metro node

A Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

Storage and data protection technical documents and videos provide expertise to ensure customer success with Dell EMC storage and data protection products.