WHITE PAPER

PURE STORAGE ACTIVECLUSTER

WITH MICROSOFT SQL SERVER

PURESTORAGE®

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Business critical applications often need the highest possible resiliency to ensure that operations do not stop in the case of a localized or site-wide disaster. To ensure this, the data in use by those applications needs to be distributed across two storage arrays, which may be located in different geographic locations. It is important that the data is available in both sites at the same time. To achieve this, some arrays offer synchronous replication, which provides the ability to write to the same block storage volume simultaneously while maintaining write-order. This is traditionally called Active-Active replication.

Pure Storage FlashArray™ introduced Active-Active replication with the Purity 5.0.0 release. This paper provides configuration steps and best practices for using the Pure Storage® FlashArray ActiveCluster feature with Windows Server Failover Cluster and SQL Server Failover Cluster Installation as a solution. For specific best practices for individual features, please refer to Pure Storage or Microsoft SQL Server and Windows documentation accordingly.

AUDIENCE

The intended audience for this paper is storage, SQL Server database, and system administrators who plan to implement Pure Storage FlashArray ActiveCluster with Windows Server Failover Cluster and SQL Server Failover Cluster Installation features.

For specific questions or requests for assistance, please reach out to Pure Storage support at any time.

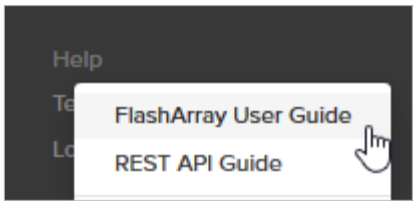
IMPORTANT NOTES

Review the following Microsoft Windows Server Failover Cluster and SQL Server Failover Cluster Installation documentation before implementing these technologies:

1. https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/install/before-installing-failover-clustering?view=sql-server-2017
2. https://docs.microsoft.com/en-us/windows-server/failover-clustering/create-failover-cluster
3. https://docs.microsoft.com/en-us/windows-server/failover-clustering/create-failover-cluster#create-a-failover-cluster-by-using-windows-powershell
4. https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/install/create-a-new-sql-server-failover-cluster-setup?view=sql-server-2017
5. https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/install/sql-server-failover-cluster-installation?view=sql-server-2017

Before setting up ActiveCluster with Windows Server Failover Cluster and SQL Server Failover Cluster Installation, it is important to review the FlashArray User Guide, as well as ActiveCluster, SQL Server, and Windows Best Practices documentation.

6. FlashArray User Guide



7. https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/Microsoft_SQL_Server/001_Microsoft_SQL_Server_Quick_Reference
8. https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/Failover_Clustering
9. https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/aaa_Quick_Setup_Steps
10. https://support.purestorage.com/FlashArray/PurityFA/Protect/Replication/ActiveCluster_Requirements_and_Best_Practices
11. https://support.purestorage.com/FlashArray/PurityFA/Protect/Replication/ActiveCluster_Solution_Overview

> **NOTE:** Ensure that the above-stated best practices have been followed. This document assumes that the aforementioned documentation has been read prior to implementing these best practices.

SOLUTION OVERVIEW & INTRODUCTION

The most robust, resilient, and automated solution for critical data protection and availability combines three technologies:

- **Microsoft SQL Server Database Platform** – Microsoft's relational database product offering with the SQL Server Failover Cluster Installation feature.
- **Pure Storage FlashArray ActiveCluster** – a simple, built-in, active-active synchronous storage replication solution for FlashArray block storage.
- **Microsoft Windows Server Failover Cluster** – a feature in Windows that provides high availability for Windows Server applications for disaster recovery use cases.

SQL Server Instance High Availability

A SQL Server database instance integrates with the Windows Server Failover Cluster (WSFC) service. This is achieved using shared storage, network, and cluster resources that span participating nodes in the WSFC.

SQL Server Failover Cluster

The WSFC feature is a shared-nothing model. A SQL Server failover instance is installed on a primary active node and the remaining nodes are considered passive. The remaining passive nodes are added to the SQL Server failover cluster. If the primary active node where the SQL Server failover instance is running goes offline, a node failover occurs. One of the passive nodes in the WSFC becomes the primary active node, where the SQL Server failover instance will start.

EXAMPLE SQL SERVER FAILOVER CLUSTER: SINGLE SITE

In this example, there are three nodes: one node is actively running a SQL Server failover instance while the remaining nodes are passive until a failover occurs to another node determined by the Windows Server Failover Cluster service and resources. This use case provides the SQL Server application high availability within a single site.

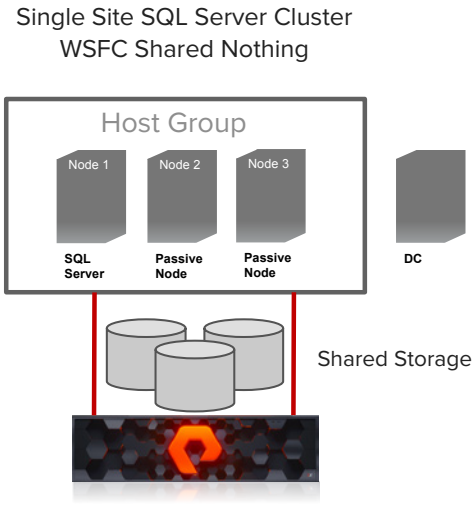


FIGURE 1. Example SQL Server cluster: single site, shared nothing active/passive nodes

The combination of shared storage and WSFC provides an extremely high level of resiliency to the infrastructure. When the automatic failover and recovery features offered by WSFC are enabled on top of these two topologies, Recovery Time Objective (RTO) and Recovery Point Objective (RPO) are reduced even further, as WSFC can quickly and intelligently respond to host, storage, or site failures to bring hosts back online.

ActiveCluster

Pure Storage Purity ActiveCluster is a fully symmetric, active/active bidirectional replication solution that provides synchronous replication. ActiveCluster spans multiple sites, enabling clustered arrays and clustered hosts to be used to deploy flexible active/active datacenter configurations.

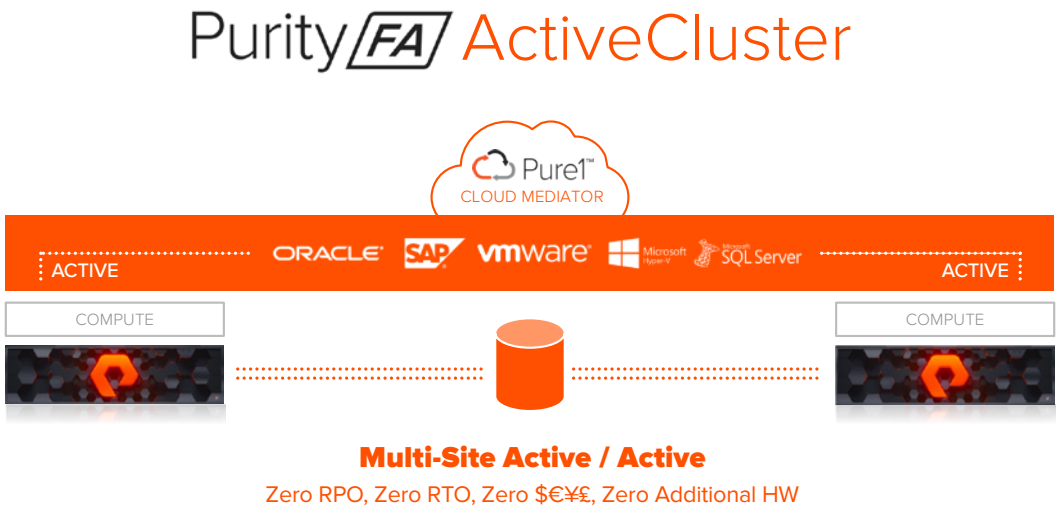


FIGURE 2. ActiveCluster synchronous replication

Synchronous Replication – Writes are synchronized between arrays and protected in non-volatile RAM (NVRAM) on both arrays before being acknowledged to the host.

Symmetric Active/Active – Read and write to the same volumes at either side of the mirror, with optional host-to-array site awareness.

Transparent Failover – Automatic non-disruptive failover between synchronously replicating arrays and sites with automatic resynchronization and recovery.

Asynchronous Replication Integration – Uses async for baseline copies and resynchronizing. Convert async relationships to sync without resending data. Async data to a 3rd site for DR.

No Bolt-ons & No Licenses – No additional hardware required, no costly software licenses required – just upgrade the Purity Operating Environment and go active/active!

Simple Management – Perform data management operations from either side of the mirror, such as provisioning storage, connecting hosts, creating snapshots, and creating clones.

Integrated Pure1® Cloud Mediator – Automatically configured passive mediator that allows transparent failover and prevents split-brain, without the need to deploy and manage another component.

COMPONENTS

Purity ActiveCluster is composed of three core components: Pure1 Mediator, active/active clustered array pairs, and stretched storage containers.

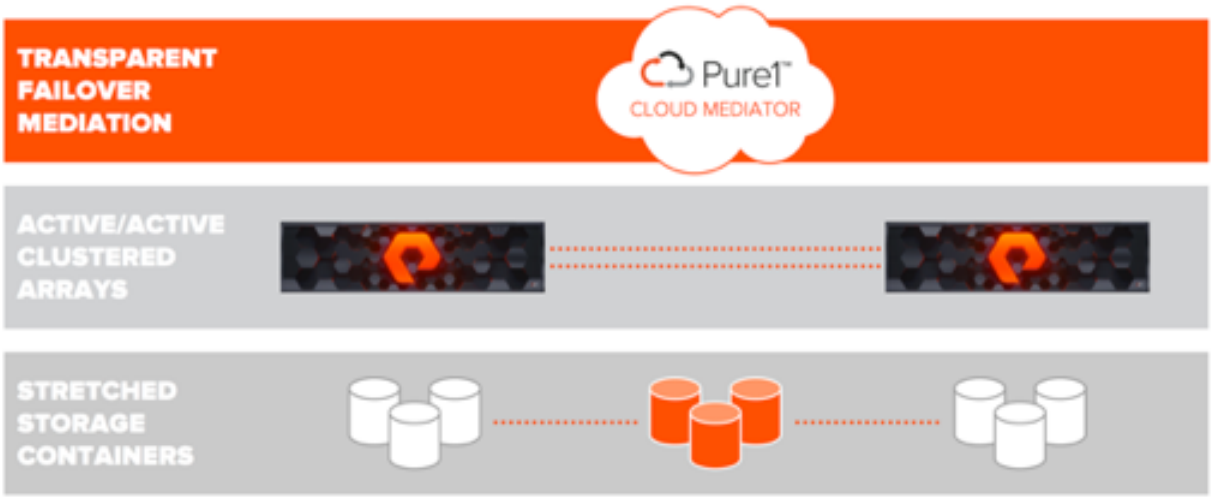


FIGURE 3. ActiveCluster components

Pure1 Cloud Mediator – Required component of the solution that determines which array will continue data services should an outage occur in the environment.

Active/Active Clustered FlashArrays – Utilize synchronous replication to maintain a copy of data on each array and present those as one consistent copy to hosts that are attached to either, or both, arrays.

Stretched Storage Containers – Management containers that collect storage objects such as volumes into groups that are stretched between two arrays.

ADMINISTRATION

ActiveCluster introduces a new management object: **Pods**. A pod is a stretched storage container that defines a set of objects that are synchronously replicated, along with the arrays they are replicated between. An array can support multiple pods. Pods can exist on just one array or on two arrays simultaneously with synchronous replication. Pods that are synchronously replicated between two arrays are said to be stretched between arrays.

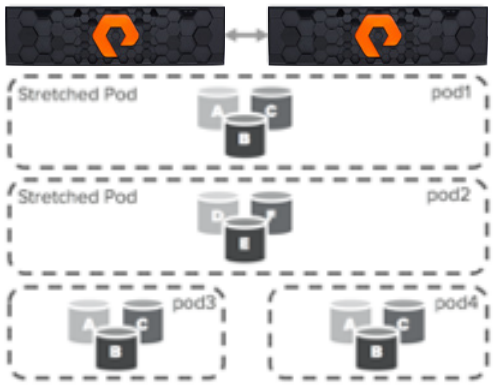


FIGURE 4. ActiveCluster management pods

Pods can contain volumes, protection groups (for snapshot scheduling and asynchronous replication), and other configuration information, such as which volumes are connected to which hosts. The pod acts as a consistency group, ensuring that multiple volumes within the same pod remain write-order consistent.

Pods also provide volume namespaces – that is, different volumes may have the same volume name if they are in different pods. In the image above, the volumes in **pod3** and **pod4** are different volumes than those in **pod1**, a stretched active/active pod. This allows migration of workloads between arrays, or consolidation of workloads from multiple arrays to one, without volume name conflicts.

MEDIATOR

Transparent failover between arrays in ActiveCluster is automatic and requires no intervention from the storage administrator. Failover occurs within standard host I/O timeouts similar to the way failover occurs between two controllers in one array during non-disruptive hardware or software upgrades.

ActiveCluster is designed to provide maximum availability across symmetric active/active storage arrays while preventing a split-brain condition from occurring. Split brain is the undesirable situation in which two arrays serve I/O to the same volume, without keeping the data in sync between the two arrays.

Any active/active synchronous replication solution designed to provide continuous availability across two different sites requires a component referred to as a witness, or voter, to mediate failovers while preventing split brain. ActiveCluster includes a simple-to-use, lightweight, and automatic way for applications to transparently failover, or simply move, between sites in the event of a failure, without user intervention: Pure1 Cloud Mediator.

Pure1 Cloud Mediator is responsible for ensuring that only one array is allowed to stay active for each pod when there is a loss of communication between arrays.

In the event that the arrays can no longer communicate with each other over the replication interconnect, both arrays will pause I/O and reach out to the mediator to determine which array can stay active for each sync replicated pod.

The first array to reach the mediator is allowed to keep its synchronously replicated pods online. The second array to reach the mediator must stop servicing I/O to its synchronously replicated volumes, in order to prevent split brain. The entire operation occurs within standard host I/O timeouts to ensure that applications experience no more than a pause and then resume I/O.

The Pure1 Cloud Mediator

A failover mediator must be located in a third site that is in a separate failure domain from either of the sites hosting the arrays. Each array site must have independent network connectivity to the mediator, such that a single network outage does not prevent both arrays from accessing the mediator. The mediator itself should be a lightweight and easy-to-administer component of the solution. The Pure Storage solution provides this automatically by utilizing an integrated cloud-based mediator. Pure1 Cloud Mediator provides two main functions:

- Prevents a split brain condition from occurring in which both arrays are independently allowing access to data without synchronization between arrays.
- Determines which array will continue to service IO to synchronously replicated volumes in the event of an array failure, replication link outage, or site outage.

Pure1 Cloud Mediator has the following advantages over a typical, non-Pure, heavy-handed voter or witness component:

- **SaaS Operational Benefits** – As with any SaaS solution, the operational maintenance complexity is removed: there is nothing to install onsite, no hardware or software to maintain, nothing to configure and support for High Availability (HA), no security patch updates, etc.
- **Automatically a Third Site** – Pure1 Cloud Mediator is inherently in a separate failure domain from either of the two arrays.
- **Automatic Configuration** – Arrays configured for synchronous replication will automatically connect to and use Pure1 Cloud Mediator.
- **No Misconfiguration** – With automatic and default configuration there is no risk that the mediator could be incorrectly configured.
- **No Human Intervention** – A significant number of issues in non-Pure active/active synchronous replication solutions, particularly those related to accidental split brain, are related to human error. Pure's automated non-human mediator eliminates operator error from the equation.
- **Passive Mediation** – Continuous access to the mediator is not required for normal operations. The arrays will maintain a heartbeat with the mediator. Even if the arrays lose connection to the mediator, they will continue to synchronously replicate and serve data as long as the replication link is active.

On-Premises Failover Mediator

Failover mediation for ActiveCluster can also be provided using an on-premises mediator distributed as an Open Virtual Machine Format (OVF) file and deployed as a Virtual Machine (VM). Failover behaviors are exactly the same as described above. The on-premises mediator simply replaces the role of Pure1 Cloud Mediator during failover events.

The on-premises mediator has the following basic requirements:

- The on-premises mediator can only be deployed as a VM on virtualized hardware. It is not installable as a stand-alone application.
- High Availability for the mediator must be provided by the hosts on which the mediator is deployed, for example, by using VMware HA, or Microsoft Hyper-V HA Clustering.
- Storage for the mediator must not allow the configuration of the mediator to be rolled back to previous versions. This applies to situations such as storage snapshot restores, or cases where the mediator might be stored on mirrored storage.
- The arrays must be configured to use the on-premises mediator rather than Pure1 Cloud Mediator.
- The mediator must be deployed in a third site, in a separate failure domain, that will not be affected by any failures in either of the sites where the arrays are installed.
- Both array sites must have independent network connections to the mediator such that a failure of one network connection does not prevent both arrays from accessing the mediator.

Uniform and Non-Uniform Host Access

Hosts can be configured to see just the array that is local to it, or to both arrays. The former option is called non-uniform, the latter is referred to as uniform.

UNIFORM ACCESS

A uniform storage access model can be used in environments where there is host-to-array connectivity of either Fiber Channel (FC) or Ethernet (for iSCSI), and array-to-array ethernet connectivity, between the two sites. When deployed in this way, a host has access to the same volume through both the local array and the remote array. The solution supports connecting arrays with up to 11ms of round trip time (RTT) latency between the arrays.

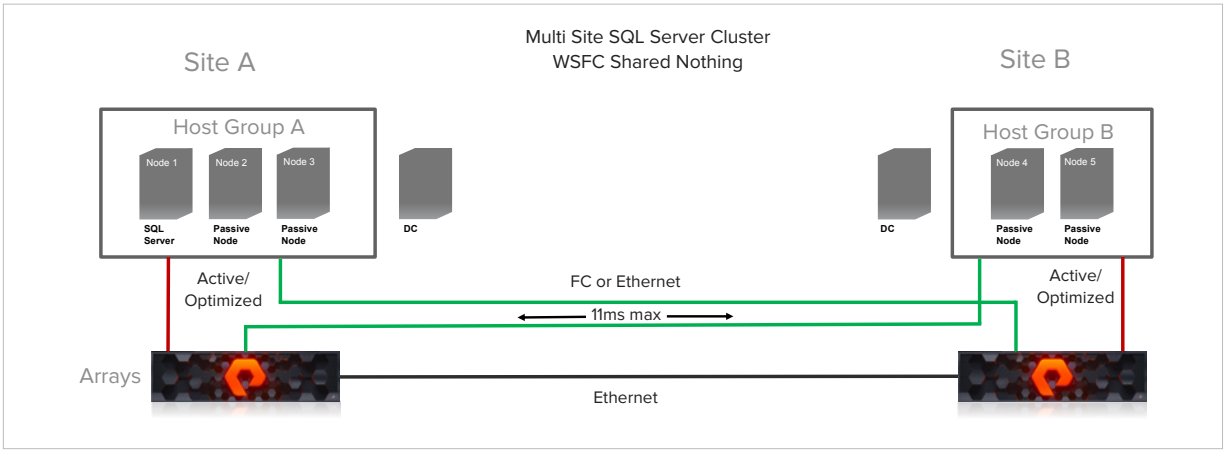


FIGURE 5. Uniform access model

The image above represents the logical paths that exist between the hosts and arrays, and the replication connection between the two arrays in a uniform access model. Because a uniform storage access model allows all hosts, regardless of site location, to access both arrays, there will be paths with different latency characteristics. Paths from hosts to the local array will have lower latency; paths from each local host to the remote array will have higher latency.

For the best performance in an active/active synchronous replication environment, hosts should be prevented from using paths that access the remote array unless it is necessary. For example, in the image below, if the host SQL Server were to perform a write to volume A over the host side connection to array A, that write would incur 2X the latency of the inter-site link (1X for each traverse of the network). The write would experience 11ms of latency for the trip from host B to array A and experience another 11ms of latency while array A synchronously sends the write back to array B.

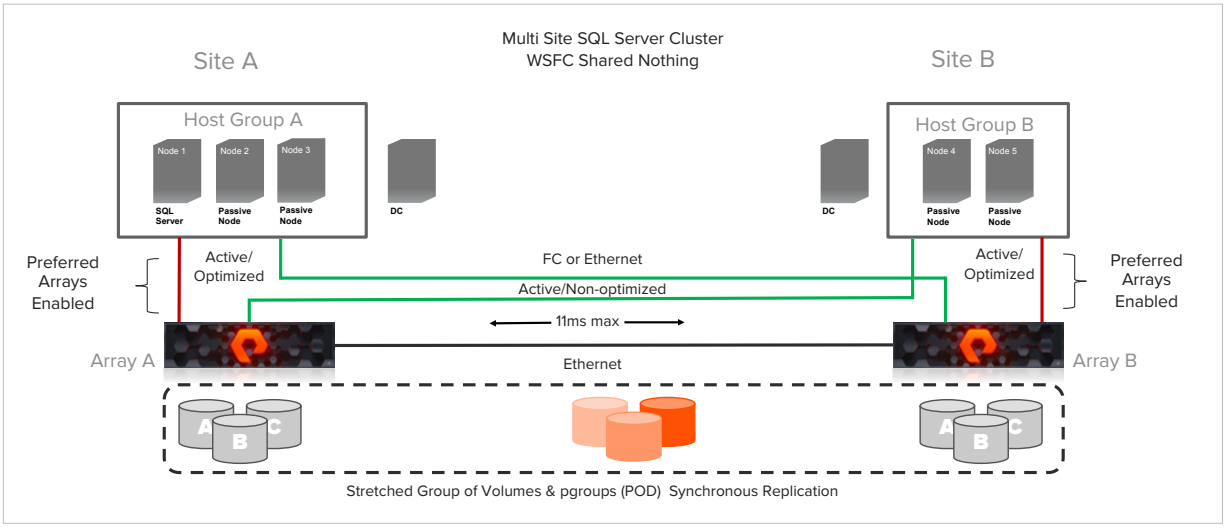


FIGURE 6. Inefficient writes to remote array

With Pure Storage Purity ActiveCluster there are no such management headaches. In the image below, ActiveCluster does make use of Asymmetric Logical Unit Access (ALUA) to expose paths to local hosts as active/optimized paths and expose paths to remote hosts as active/non-optimized. However, there are two advantages in the ActiveCluster implementation:

1. In ActiveCluster, volumes in stretched pods are read/write on both arrays. There is no such thing as a passive volume that cannot service both reads and writes.
2. The optimized path is defined on a per host-to-volume connection basis using a preferred-array option; this ensures that regardless of which host the SQL Server is running on, it will have a local optimized path to that volume.

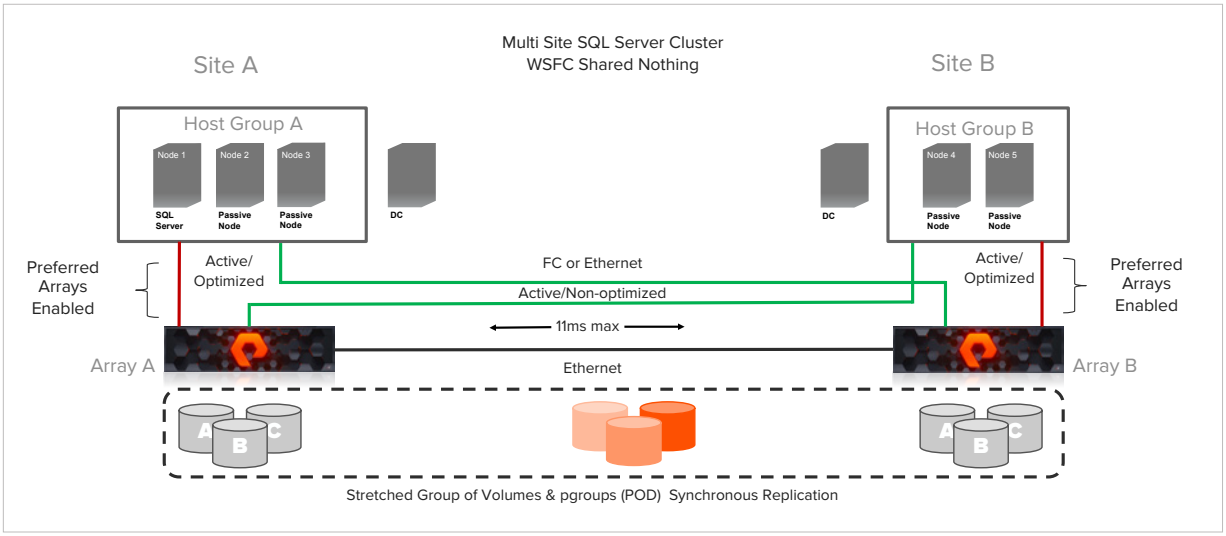


FIGURE 7. Inefficient writes to remote array

ActiveCluster enables truly active/active datacenters and removes the concern around what site or host a SQL Server runs on; the SQL Server host will always have the same performance regardless of site. While a SQL Server host is running on host A and accessing volume A, it will use only the local optimized paths, as shown in the next image.

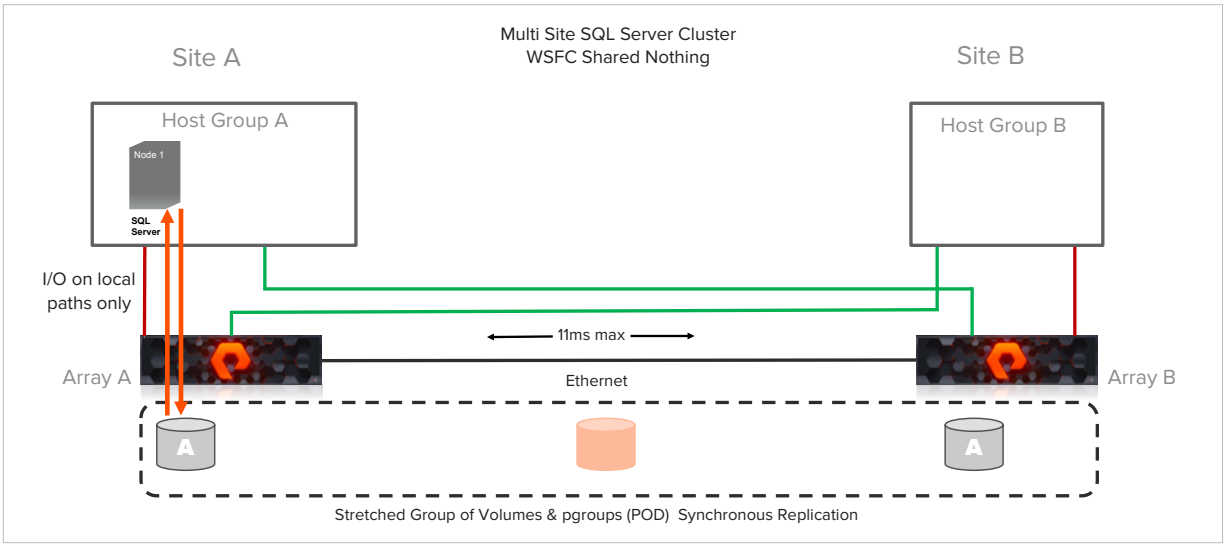


FIGURE 8. Optimized paths

If the SQL Server is switched to a host in the other site, with the data left in place, only local paths will be used in the other site. There is no need to adjust path priorities or migrate the data to a different volume to ensure local optimized access.

Non-Uniform Access

A non-uniform storage access model is used in environments where there is host-to-array connectivity of either FC or ethernet (for iSCSI) only locally within the same site. Ethernet connectivity for the array-to-array replication interconnect must still exist between the two sites. When deployed in this way, each host has access to a volume only through the local array and not the remote array. The solution supports connecting arrays with up to 11ms of round trip time (RTT) latency between the arrays.

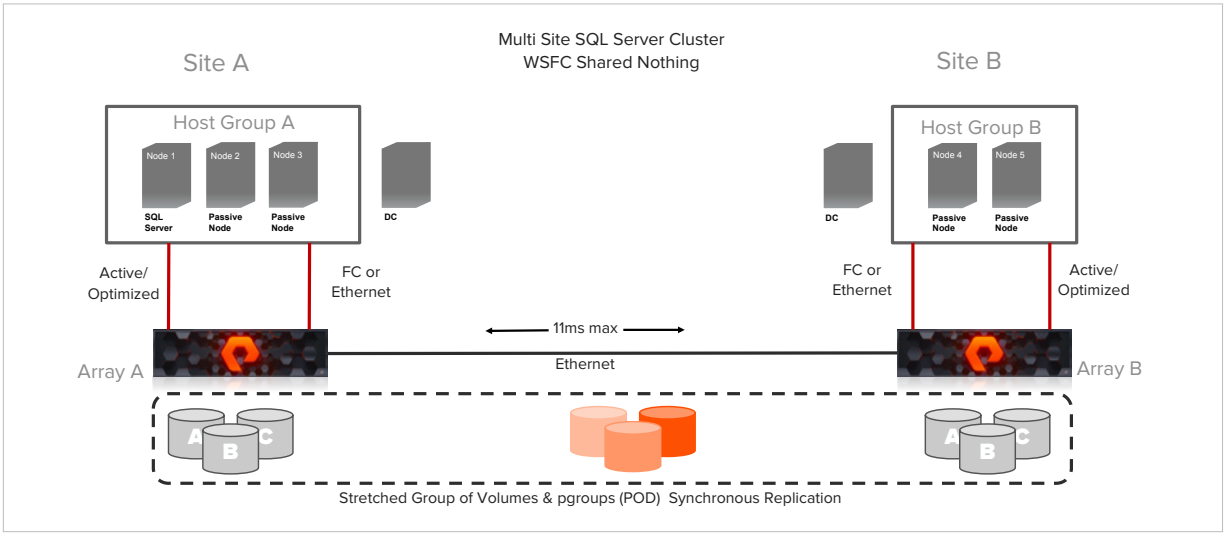


FIGURE 9. Non-uniform access model

Hosts will distribute I/Os across all paths to the storage only, because only the local Active/Optimized paths are available.

CONFIGURING ACTIVECLUSTER

A major benefit of using an ActiveCluster stretched storage solution is simplicity. Before moving forward, ensure environment configuration requirements match the description in this KB article.

ActiveCluster Glossary of Terms

The following terms will be used:

- **Pod** – A pod is a namespace and a consistency group. Synchronous replication can be activated on a pod, which makes all volumes in that pod present on both FlashArrays in the pod.
- **Stretching** – Stretching a pod is the act of adding a second FlashArray to a pod. When stretched to another array, the volume data will begin to synchronize. When synchronization completes, all volumes in the pod will be available on both FlashArrays.
- **Unstretching** – Unstretching a pod is the act of removing a FlashArray from a pod. This can be done from either FlashArray. When removed, the volumes and the pod itself are no longer available on the FlashArray that was removed.
- **Restretching** – When a pod is unstretched, the other array (the array unstretched from) will keep a copy of the pod in the trash can for 24 hours – in order to allow the pod to be quickly re-stretched without having to resend all the data, if restretched within 24 hours.

Creating a Synchronous Connection

The first step to enable ActiveCluster is to create a synchronous connection with another FlashArray. It does not matter which FlashArray is used to create the connection – either one is fine.

Login to the FlashArray web interface and click on the **Storage** section. Click either on the **+** sign or on the **vertical ellipsis** and choose **Connect Array**.

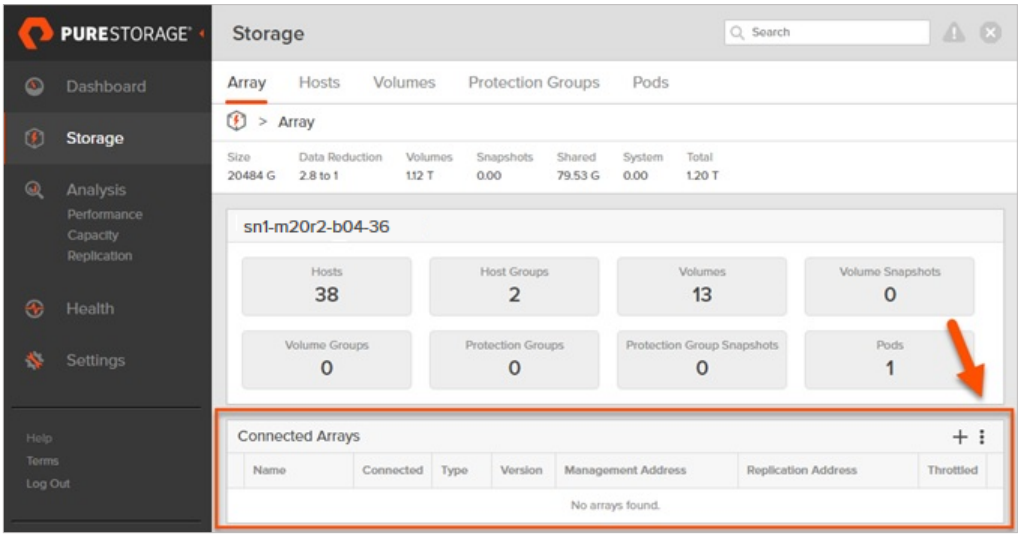


FIGURE 10. FlashArray web interface

The window that pops-up requires three pieces of information:

1. **Management Address** – This is the **virtual IP address** or Fully Qualified Domain Name (FQDN) of the remote FlashArray.
2. **Connection Type** – Choose **Sync Replication** for ActiveCluster.
3. **Connection Key** – This is an **API token** that can be retrieved from the remote FlashArray.

To obtain the connection key, login to the remote FlashArray web interface, click on the **Storage** section, and then click on the **vertical ellipsis** and choose **Get Connection Key**.

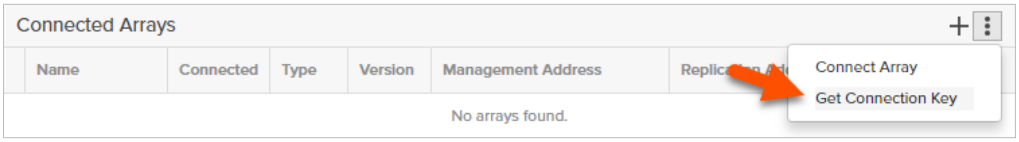


FIGURE 11. Getting a connection key

Copy the key to the clipboard using the **Copy** button.

FIGURE 12. Copying the key

Go back to the original FlashArray web interface and paste in the key.

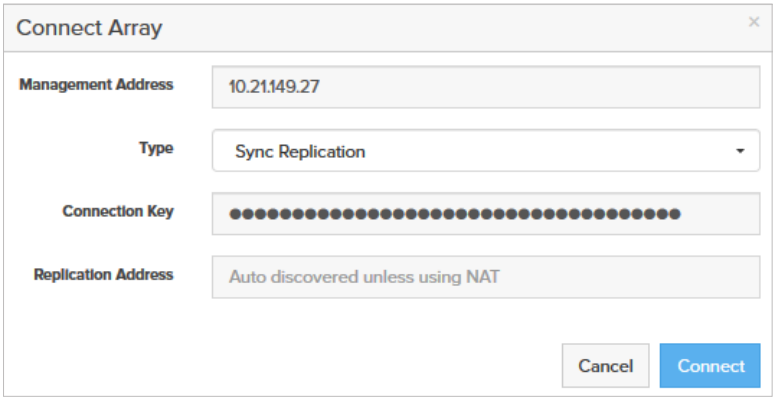


FIGURE 13. Connecting an array with the connection key

The replication address field may be left blank and Purity will automatically discover all the replication port addresses. If the target addresses are via Network Address Translation (NAT), then it is necessary to enter the replication port NAT addresses.

When complete, click **Connect**. If everything is valid, the connection will appear in the **Connected Arrays** pane.

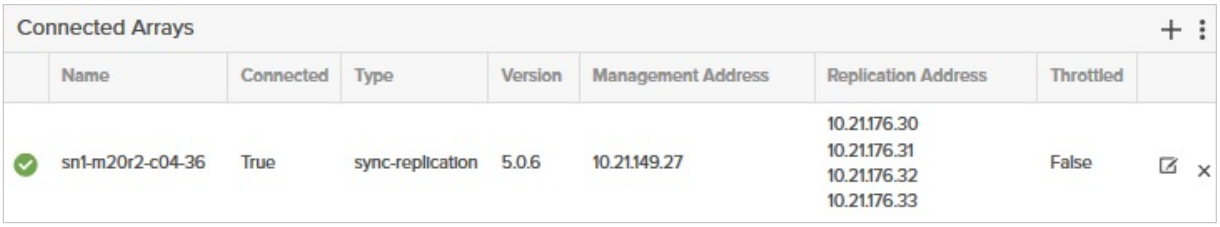


FIGURE 14. Connected Arrays pane

If the connection fails, verify network connectivity and IP information and/or contact Pure Storage Support.

Creating a Pod

The next step to enable ActiveCluster is to create a consistency group. With ActiveCluster, this is called a "pod".

A pod is both a consistency group and a namespace – in effect creating a grouping for related objects involved in ActiveCluster replication. One or more pods can be created. Pods are stretched, unstretched, and failed over together.

Therefore, the basic idea for one or more pods is simply to put related volumes in the same pod. If they host related applications that should remain in the same datacenter together, or have consistency with one another, put them in the same pod. Otherwise, put them in the same pod for simplicity, or different pods if they have different requirements. See the this KB on FlashArray object limits for additional guidance.

To create a pod, login to the FlashArray GUI and click on **Storage**, then **Pods**, then click on the **+** sign.

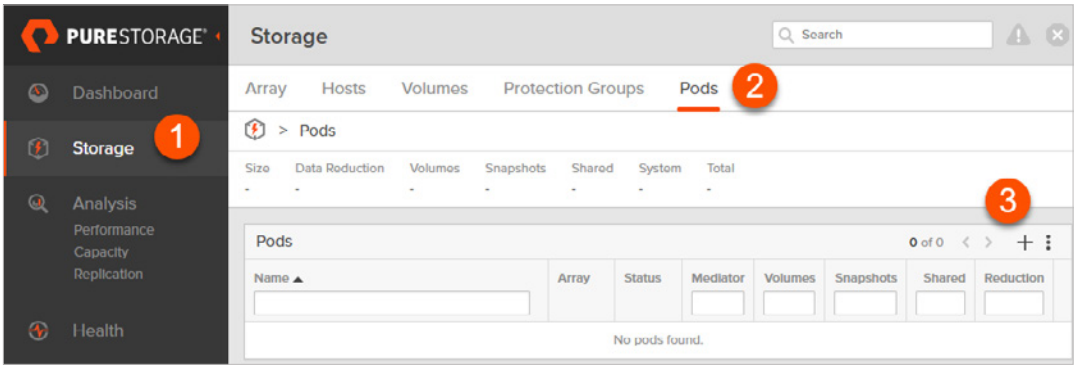


FIGURE 15. FlashArray GUI

Next, enter a name for the pod. The name can include letters, numbers, or dashes (must start with a letter or number). Then click **Create**.

FIGURE 16. Create Pods pane

The pod will then appear in the **Pods** panel.

FIGURE 17. Created pod displays in the Pods pane

To further configure the pod, click on the pod name.

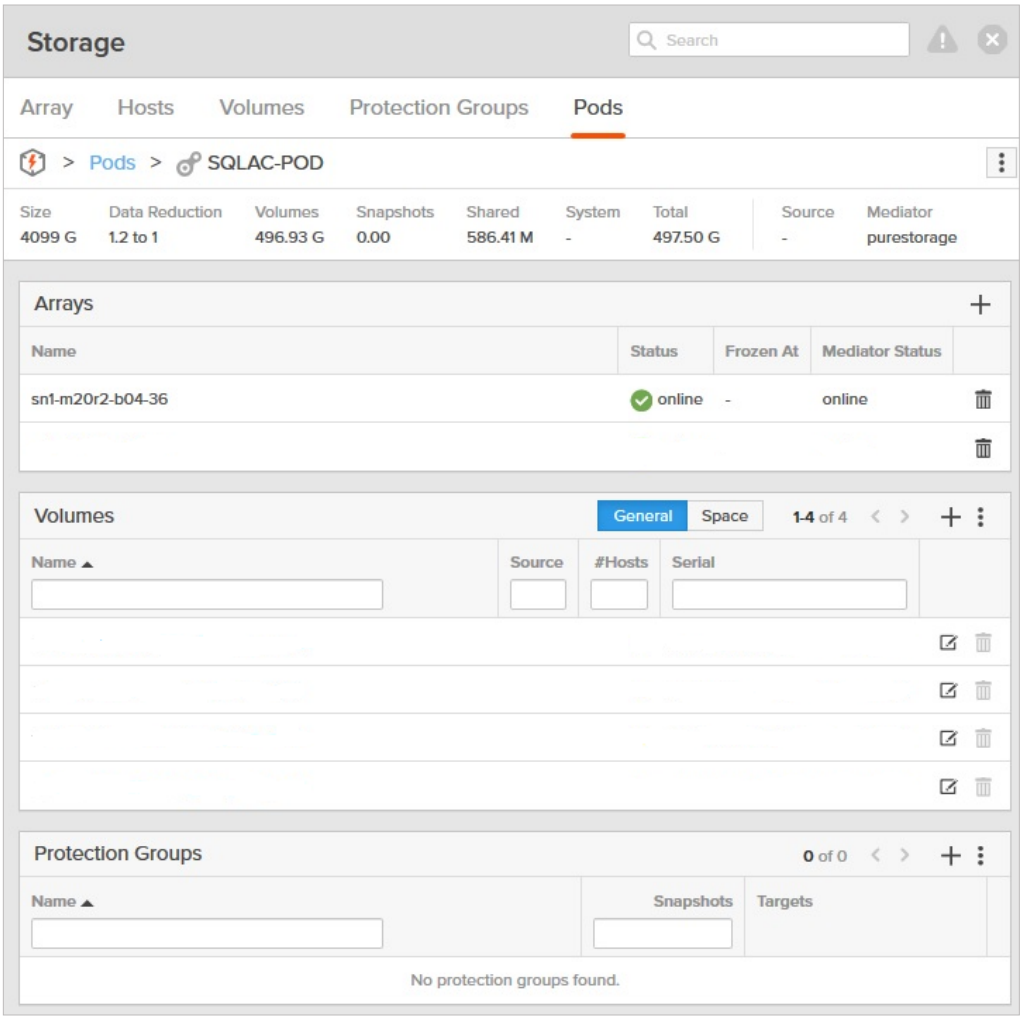


FIGURE 18. Pods panel drill-down

The default configuration for ActiveCluster is to use Pure1 Cloud Mediator – no configuration is required other than to ensure the management network from FlashArray is redundant (uses two NICs per controller) and has IP access to the mediator. Refer to the networking section in the ActiveCluster Best Practices Guide for more details.

The mediator in use can be seen in the overview panel under the **Mediator** heading. If the mediator is listed as "purestorage", Pure1 Cloud Mediator is in use.

For sites that are unable to contact Pure1 Cloud Mediator, the ActiveCluster On-Premises Mediator is available. For deployment instructions, refer to the On-Premises Mediator Deployment Guide.

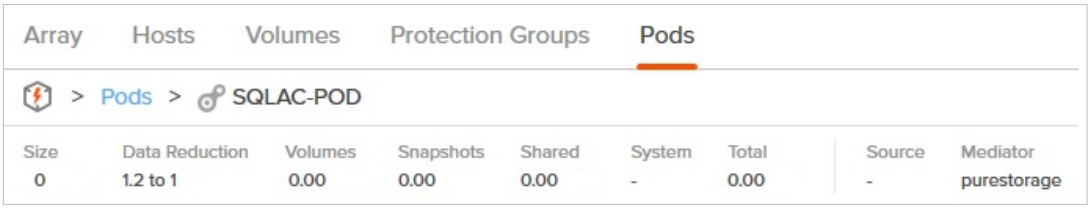


FIGURE 19. Pods configuration, with mediator shown

Pod configuration is complete.

Adding Volumes to a Pod

The next step is to add any pre-existing volumes to the pod. Once a pod has been enabled for replication, pre-existing volumes cannot be moved into the pod; only new volumes can be created in the pod.

To add a volume to a pod, go to the **Storage** screen in the FlashArray web interface, click on **Volumes**, and then click on the **name** of a volume to be added to the pod. To find the volume quickly, search for it by name.

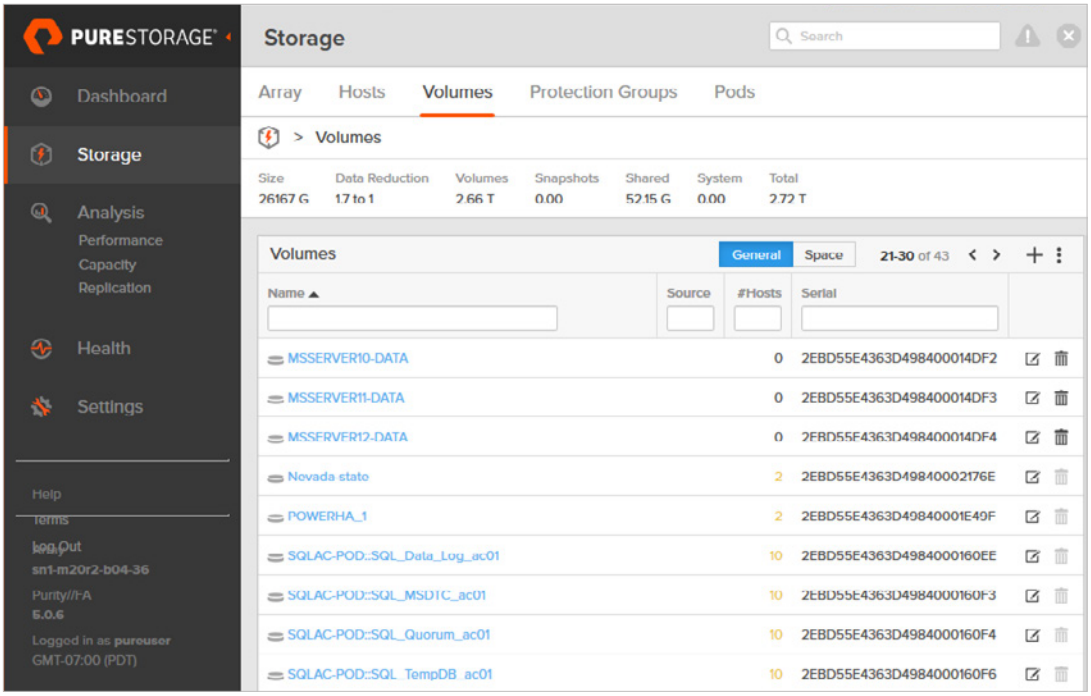


FIGURE 20. Pure Storage GUI, Storage pane

When the volume screen loads, click on the **vertical ellipsis** in the upper right-hand corner and choose **Move**.

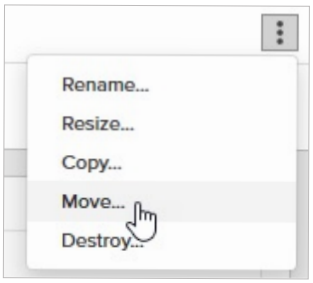


FIGURE 21. Moving a volume

To choose the pod, click on the **Container** box and choose the pod **name**.

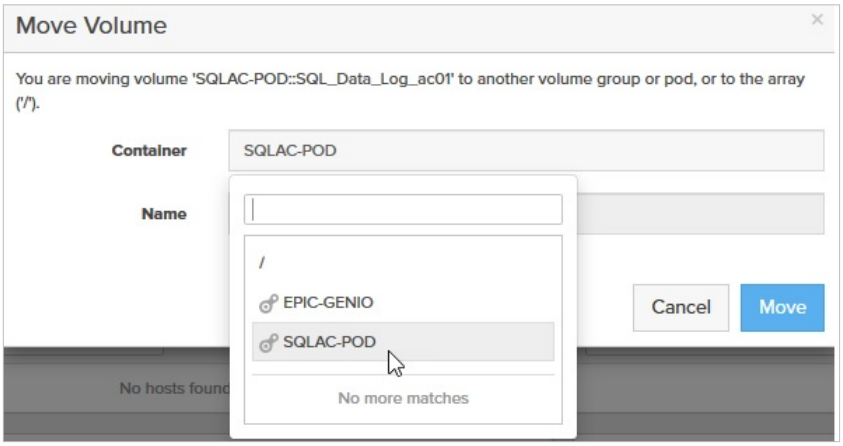


FIGURE 22. Move Volume pane

Note that as of Purity 5.0.0, the following limitations exist with respect to moving a volume into or out of a pod:

- Volumes cannot be moved directly between pods. A volume must first be moved out of a pod, then moved into the other pod.
- A volume in a volume group cannot be added into a pod. It must be removed from the volume group first.
- A volume cannot be moved out of an already stretched pod. The pod must first be unstretched, then the volume can be moved out.
- A volume cannot be moved into an already stretched pod. It must first be unstretched and then an existing volume can be added. At this point the pod can then be re-stretched.

Some of these restrictions may relax in future Purity versions.

Choose the valid target pod and click **Move**.

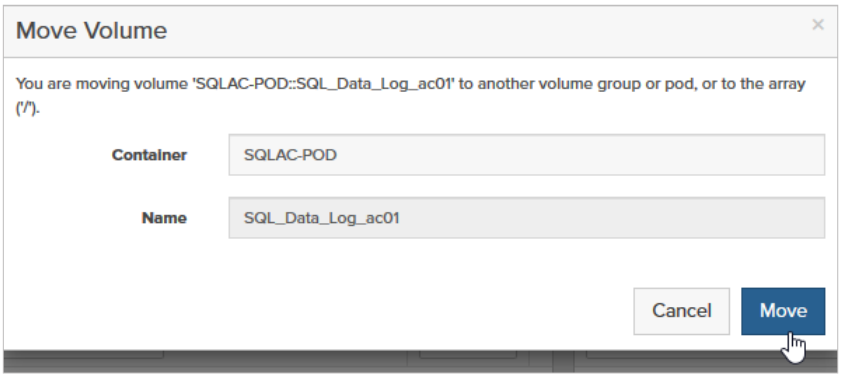


FIGURE 23. Move Volume pane

This will move the volume into the pod and rename the volume to have a prefix consisting of the pod name and two colons. The volume name will then be in the format of **<podname>::<volumename>**.

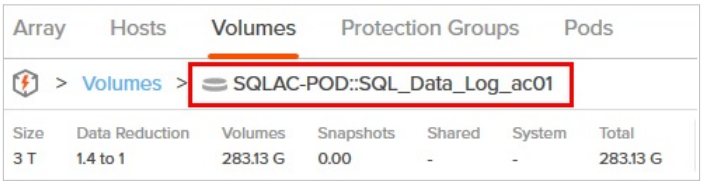


FIGURE 24. Volume moved and renamed

The pod will list the volume under its **Volumes** panel.

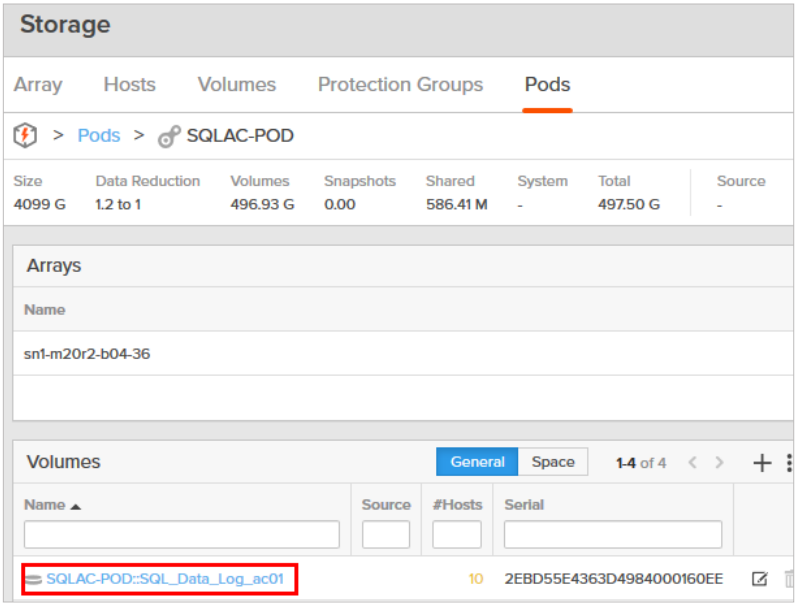


FIGURE 25. Volume displayed

Stretching a Pod

The next step is adding a FlashArray target to the pod. This is called “stretching” the pod, because it automatically makes the pod and its content available on the second array.

> Please note that once a pod has been “stretched”, pre-existing volumes cannot be added to it until it is “un-stretched”. Otherwise, once a pod has been stretched, only new volumes can be created in the pod. Therefore, if it is necessary to add existing volumes to a pod, follow the instructions in the section **Adding Volumes to a Pod**, then stretch the pod.

To stretch a pod, add a second array to the pod. Do this by clicking on the **+** sign in the **Arrays** pane.

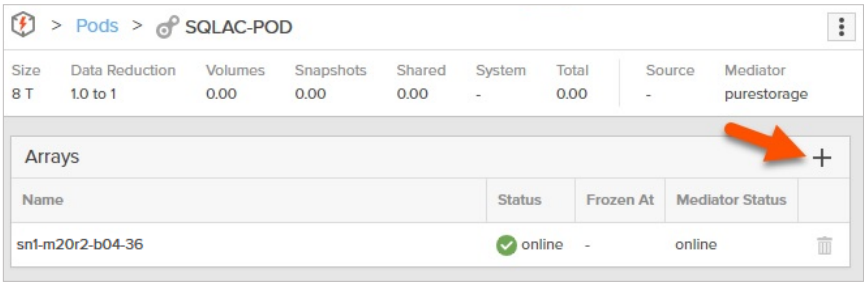


FIGURE 26. Stetching a pod via the Arrays pane

Choose a target FlashArray and click **Add**.

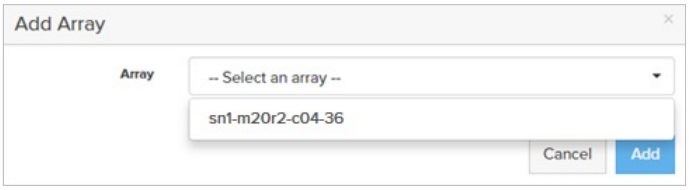


FIGURE 27. Adding a Flasharray target

The arrays will immediately start synchronizing data between the two FlashArrays.

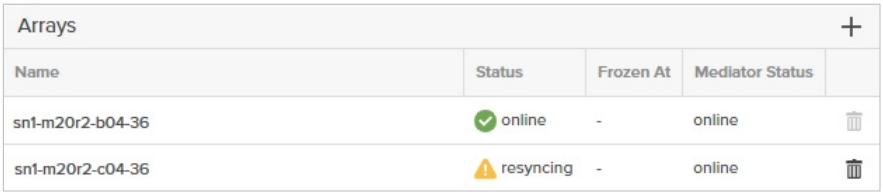


FIGURE 28. Synchronization begins

Active/Active storage is not available until the synchronization completes, which will be shown when the resyncing status ends and both arrays are online.

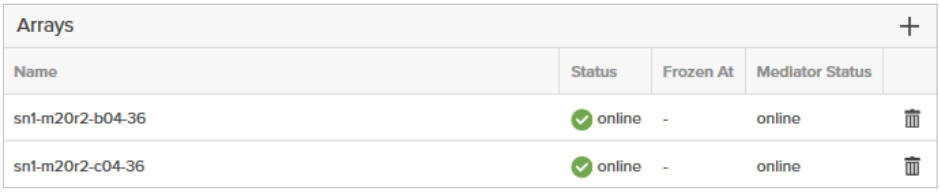


FIGURE 29. Active/Active storage online

On the remote FlashArray, the pod and volumes will now exist in identical fashion and will be available for provisioning to a host or hosts on either FlashArray.

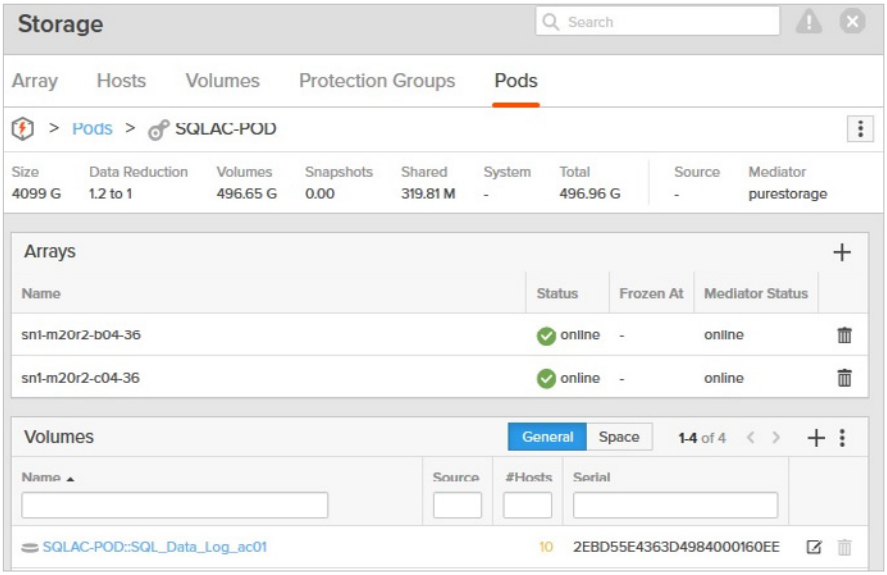


FIGURE 30. Storage pane displaying synchronized arrays

CONFIGURATION INFORMATION

Configuration of the Windows Server hosts is not different with regard to configuration for non-ActiveCluster FlashArrays – the best practices described in the following documents still apply:

https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/Failover_Clustering

https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/aaa_Quick_Setup_Steps

https://docs.microsoft.com/en-us/windows-server/failover-clustering/create-failover-cluster

With that in mind, there are still a few things worth mentioning in this document.

Configure All Windows Hosts Participating in the WSFC

1. Enable the Windows Feature Failover-Clustering. (PowerShell example)

```
Install-WindowsFeature -Name Failover-Clustering –IncludeManagementTools –ComputerName SQL1-A
Install-WindowsFeature -Name Failover-Clustering –IncludeManagementTools –ComputerName SQL2-A
Install-WindowsFeature -Name Failover-Clustering –IncludeManagementTools –ComputerName SQL3-A
Install-WindowsFeature -Name Failover-Clustering –IncludeManagementTools –ComputerName SQL4-B
Install-WindowsFeature -Name Failover-Clustering –IncludeManagementTools –ComputerName SQL5-B
```

2. Install and Configure Multipath-IO for Windows Server per the Pure Storage Microsoft Platform Guide links above. (PowerShell example)

```
#  Add the Windows Feature Multipath-IO
Get-WindowsFeature -Name 'Multipath-IO'
Add-WindowsFeature -Name 'Multipath-IO'

#  Configure Windows Feature Multipath-IO.  Add New MPIO Device
New-MSDSMSupportedHw -VendorId PURE -ProductId FlashArray

#  Retrieve Available MPIO Devices
Get-MSDSMSupportedHw


#  Remove Default MPIO Device
Remove-MSDSMSupportedHw -VendorId 'Vendor*' -ProductId 'Product*'
Get-MSDSMSupportedHw

#  Restart Computer
Restart-Computer
```

```
#  Retrieving Current MPIO Timer Values
 Get-MPIOSetting

#  Update MPIO Timer Values
Set-MPIOSetting -NewPathRecoveryInterval 20 -CustomPathRecovery Enabled
-NewPDORemovePeriod 30 -NewDiskTimeout 60

#  Set Multipath-IO Policy
Get-MSDSMGlobalDefaultLoadBalancePolicy
Set-MSDSMGlobalDefaultLoadBalancePolicy -Policy LQD
Get-MSDSMGlobalDefaultLoadBalancePolicy

#  Restart Computer
Restart-Computer
```

FlashArray Storage Volume Configuration

- Create SQL Server Storage Volumes on Primary Site FlashArray. (example configuration)

```
SQL _ Data _ Log _ ac01   Volume 3TB
SQL _ MSDTC _ ac01        Volume 2GB
ISQL _ TempDB _ ac01      Volume 1TB
```

UNIFORM CONFIGURATION PRIMARY SITE A LOCATION

1. Create site A hosts on FlashArray.
2. Present FlashArray storage volumes to each site A host.
3. Create site A host group.
4. Associate all site A hosts to host group.
5. Connect site A FlashArray storage volumes.
6. In Windows Server ensure storage array volumes are connected at each node located at site A.
7. Online FlashArray storage volumes only at the site A primary node where SQL Server will be active.

> **NOTE:** Online volumes only at the site A primary active node. On Site A, primary active node only initialize, format, and assign derive letter/label to the storage volumes listed in above steps.

8. Stretch site A FlashArray storage to site B FlashArray.

> **NOTE:** Hosts cannot be connected to the volumes on the secondary array until the stretch is complete and the pod is online at both primary and secondary arrays.

9. Verify that pod status is "Online" and baseline stretch is completed.

UNIFORM CONFIGURATION PRIMARY SITE B LOCATION

1. Create site B hosts on FlashArray.
2. Present FlashArray storage volumes to each site B host.
3. Create site B host group.
4. Associate all site B hosts to host group.
5. Connect site B FlashArray storage volumes.
6. In Windows Server, ensure storage array volumes are connected at each node located at site B.

> **NOTE:** Online volumes only at the site A primary active node. On Site A, primary active node only initialize, format, and assign derive letter/label to the storage volumes.

UNIFORM CREATE AND CONFIGURE SITE B HOST GROUP ACCESS FOR SITE A

1. Create a site B host group located on the site A FlashArray.
2. Associate site B hosts to the new host group created above at site A FlashArray.
3. Connect SQL Server cluster FlashArray volumes according to the host group created above at Site A FlashArray.

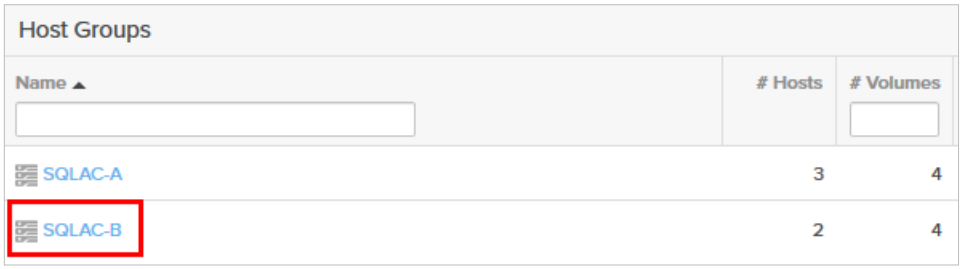


FIGURE 30. Host Groups pane

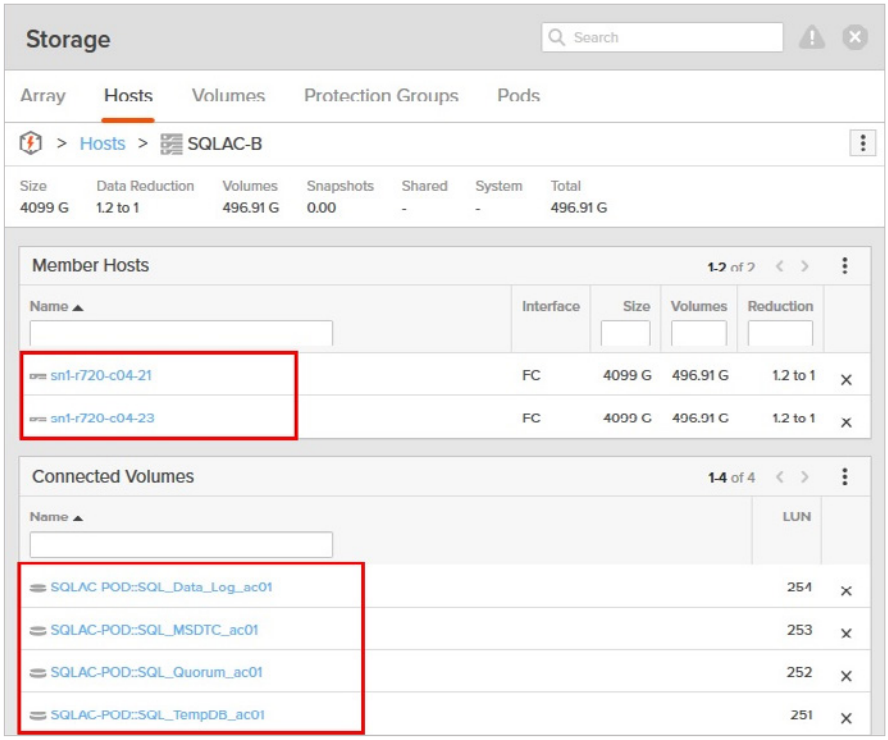


FIGURE 31. Configuring sites in the Storage pane

UNIFORM CREATE AND CONFIGURE SITE A HOST GROUP ACCESS FOR SITE B

1. Create a site A host group located on the site B FlashArray.
2. Associate site A hosts to the new host group created above at the site B FlashArray.
3. Connect SQL Server cluster FlashArray volumes according to the host group created above at Site B FlashArray.

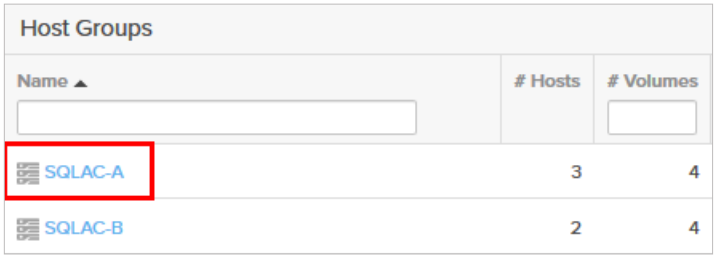


FIGURE 32. Host Groups pane

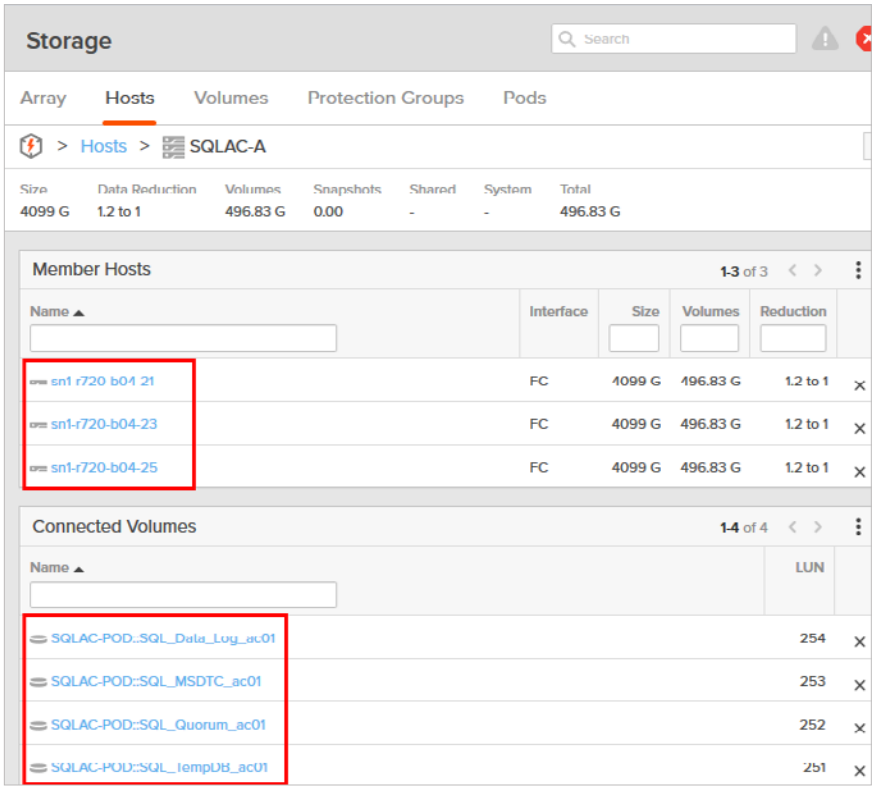


FIGURE 33. Configuring sites in the Storage pane

Preferred Paths

The default behavior is that all paths from a FlashArray to a host will be actively used by SQL Server Primary Active Node – even ones from the secondary FlashArray. When replication occurs over extended distances, this is generally not ideal. In situations where the sites are far apart, two performance-impacting things occur:

- Half of the writes (assuming both FlashArrays offer an equal number of paths for each device) sent from a host in site A will be sent to the FlashArray in site B. Since writes must be acknowledged in both sites, this means the data traverses the WAN twice. First the host issues a write across the WAN to the remote FlashArray, and then the remote FlashArray forwards it back across the WAN to the local FlashArray. This adds unnecessary latency. The optimal path is for the host to send writes to the local FlashArray, which then forwards it to the remote FlashArray. In the optimal situation, the write must only traverse the WAN once.
- Half of the reads (assuming both FlashArrays offer an equal number of paths for each device) sent from a host in site A will be sent to the FlashArray in site B. Reads can be serviced by either side, and for reads there is no need for one FlashArray to talk to the other. So a read need not ever traverse the WAN in normal circumstances. Servicing all reads from the local array to a given host is the best option for performance.

FlashArray offers an option to intelligently tell SQL Server Primary Active Node which FlashArray should optimally service I/O in the event a SQL Server Primary Active Node can see paths to both FlashArrays for a given device. This is a FlashArray host object setting called **Preferred Arrays**.

In a situation where the FlashArrays are in geographically different data centers, it is important to set the preferred array for a host on BOTH FlashArrays.

For each host, login to the FlashArray web interface for the array that is local to that host. Click on the **Storage** section, then the **Hosts** tab, then choose the host to be configured. Then, in the **Details** panel, click on the **Add Preferred Arrays** option.

> **BEST PRACTICE:** For every host that has access to both FlashArrays that host an ActiveCluster volume, set the preferred FlashArray for that host on both FlashArrays. Ensure at FlashArray A that it is preferred for host A. Ensure at FlashArray B that FlashArray A is preferred for host A. Doing this on both FlashArrays allows a host to automatically know which paths are optimized and which are not.

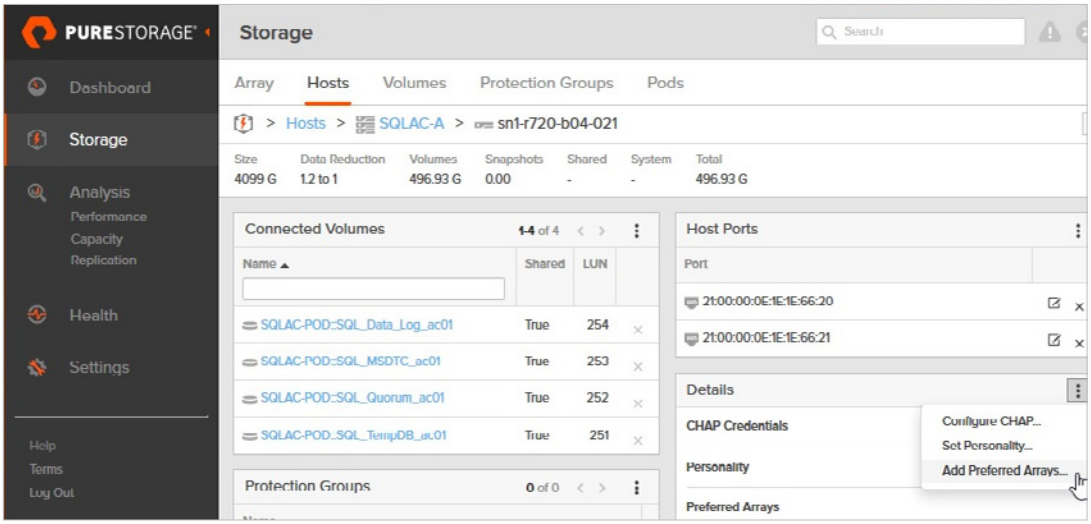


FIGURE 34. Adding Preferred Arrays at site A

Choose that FlashArray as preferred for that SQL Server Primary Active Node and click **Add**.

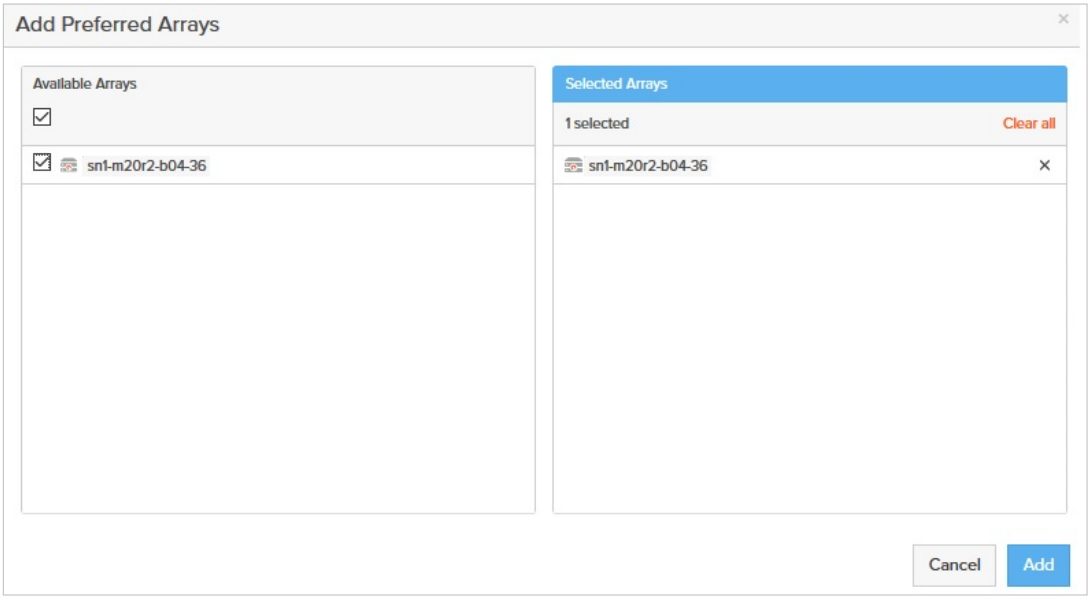


FIGURE 35. Add Preferred Arrays pane for site A

If that same host exists on the remote FlashArray, login to the remote FlashArray web interface. Click on the **Storage** section, then the **Hosts** tab, then choose the host to be configured. Then, in the **Details** panel, click on the **Add Preferred Arrays** option.

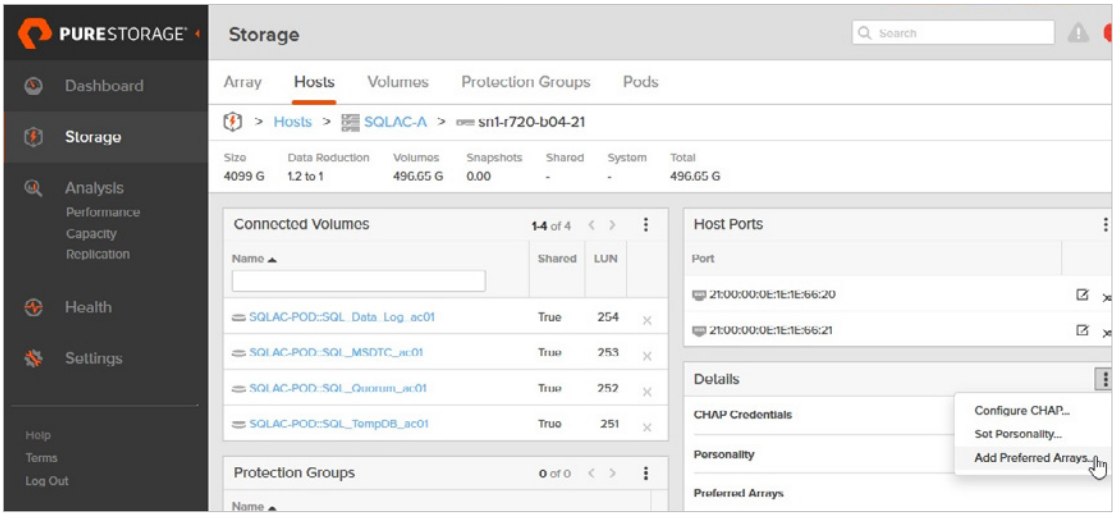


FIGURE 36. Adding Preferred Arrays at site B

Choose the earlier FlashArray as preferred for that SQL Server Primary Active Node and click **Add**.

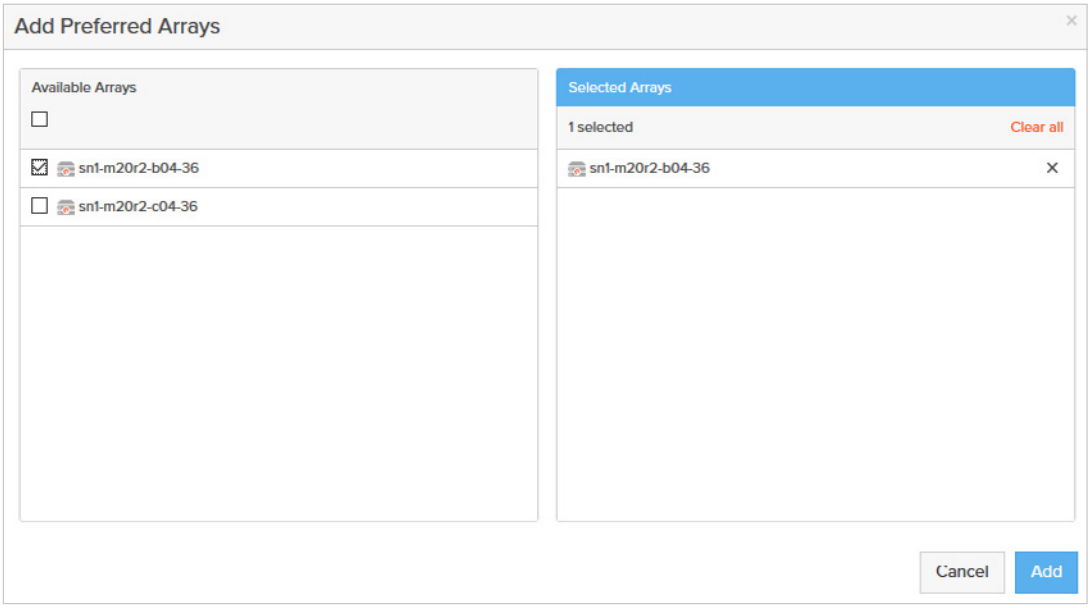


FIGURE 37. Add Preferred Arrays pane for site B

It can then be seen in Windows Server that half of the paths will now be marked as Active and the rest will be marked as Active (I/O). The Active (I/O) paths are the paths which are used for I/O. The other paths will only be used if the paths to the preferred FlashArray are unavailable.

> When a preferred array has been turned off/on or changed, FlashArray issues 6h/2a/6h (Sense code/ASC/ASCQ) which translates to UNIT ATTENTION ASYMMETRIC ACCESS STATE CHANGED to the host to inform it of the path state change proactively.

NON-UNIFORM CONFIGURATION PRIMARY SITE A LOCATION

1. Create site A hosts on FlashArray.
2. Present FlashArray storage volumes to each site A host.
3. Create site A host group.
4. Associate all site A hosts to host group.
5. Connect site A FlashArray storage volumes.
6. In Windows Server ensure storage array volumes are connected at each node located at site A.
7. Online FlashArray storage volumes only at the site A primary node where SQL Server will be active.

> NOTE. Online volumes only at the site A primary active node. On Site A, primary active node only initialize, format, and assign drive letter/label to the storage volumes listed in above steps.

8. Stretch site A FlashArray storage to site B FlashArray.

NOTE: Hosts cannot be connected to the volumes on the secondary array until the stretch is complete and the pod is online at both primary and secondary arrays.

9. Verify that pod status is “Online” and baseline stretch is completed.

NON-UNIFORM CONFIGURATION PRIMARY SITE B LOCATION

1. Create site B hosts.
2. Present FlashArray storage volumes to each site B host.
3. Create site B host group.
4. Associate all site B hosts to host group.
5. Connect site B FlashArray storage volumes.
6. In Windows Server ensure storage array volumes are connected at each node located at site B.

NOTE: Online volumes only at the site A primary active node. On Site A, primary active node only initialize, format, and assign drive letter/label to the storage volumes.

Create WSFC Windows Server Cluster

WINDOWS

1. Create a WSFC via PowerShell or the Create Cluster Wizard in Windows.
2. Run the Failover Cluster Manager from Administrative Tools.
3. Create Cluster Wizard and add the names of the server nodes or IP addresses to participate in the cluster.
4. Run validation tests. (RESOLVE ANY ERRORS).
5. Select Finish.
6. On the Access Point (Administering Cluster), enter the virtual name for the cluster.
7. Enter a Static IP Address (strongly recommended); can use DHCP but not recommended.

MSDTC (MICROSOFT DISTRIBUTED TRANSACTION COORDINATOR)

8. Configure optional SQL Server-related roles based on environment business rules. For example, it is optional to create a MSDTC (Microsoft Distributed Transaction Coordinator) role.
9. Create an MSDTC role.
10. Select the Configure Role option from the Roles context menu in the Failover Cluster Manager.
11. Use the High Availability Wizard.
12. Select the Distributed Transaction Coordinator (DTC) role type.
13. On the Client Access Point page, enter a virtual name and IP Address for MSDTC.
14. Assign name and IP Address.
15. On the Select Storage page of the wizard, select the cluster disk for the MSDTC.
16. To configure a role, select Properties from the role’s context menu.
17. Go to the General tab of the Properties dialog box.
18. Configure a node as the preferred owner of the role.

19. Set the quorum environment specific. (PowerShell example)

```
Set-ClusterQuorum –NodeMajority
```

> **BEST PRACTICE:** A WSFC Windows Server Failover Cluster with an odd number of nodes does not require a quorum disk. For a Pure AC with WSFC and SQL Server use Node Majority without any cluster disk when using an odd number of nodes. When using an even number of nodes, refer to the Microsoft documentation for Windows Server Failover Clustering.

Install SQL Server AlwaysOn Failover Cluster Components

SQL SERVER ACTIVE PRIMARY NODE

1. Run SQL Server Installation.
2. Select the New SQL Server Failover Cluster Installation option from the Installation.
3. Apply necessary updates.
4. Select which features to install. (Note: SSIS is not cluster aware)
5. Optionally select a different drive other than C:\ for the shared directories.
6. Enter the SQL Server Network name since it is a cluster.

SQL SERVER ACTIVE PRIMARY NODE INSTANCE CONFIGURATION

7. Specify the SQL Server Network Name.
8. On the Cluster Resource Group page of the wizard. Enter the name of a new resource group, which will then be created by setup as a name for a new resource group.
9. On the Cluster Disk Selection page, specify that both available disks (Data and TempDB) should be associated with the resource group.
10. On the Cluster Network Configuration page, configure the IP Address of the cluster role. Either choose DHCP, or a static IP Address (strongly recommended).
11. On the Server Configuration page, specify the domain service account that will be used as the security context for each SQL Server service.
12. Specify the startup mode for each of the services.
13. Note: typically when installing a standalone non-clustered SQL Server instance, you will usually set each service to start automatically; this is not the case in a WSFC deployment.

> **NOTE:** When installing a SQL Server failover cluster instance, the cluster-aware services should be configured to start manually; they are managed by the cluster service.

14. Specify the authentication mode to use.
15. Specify the data directories that associate with FlashArray storage for root, data, log, system, and backup locations, etc.
16. Specify the tempdb configuration and location.
17. Complete installation.

SQL SERVER PASSIVE NODE INSTANCE CONFIGURATION

18. Go to the passive node in Windows Server Run SQL Server Installation.
19. Select the Add Node to SQL Server Failover Cluster option from the Installation tab of SQL Server Installation.
20. This starts the Add a Failover Cluster Node Wizard.
21. Confirm Instance Name.
22. Confirm Network details.
23. Re-Enter the required service account passwords.
24. Complete installation.
25. Repeat the above steps at each passive node in the WSFC.

FAILURE SCENARIOS

Non-Uniform Site Storage Failure

In a non-uniform configuration, if the storage becomes unavailable to the Windows Server primary active node where the SQL Server failover cluster instance is running, this will cause a WSFC host failover to the secondary site where an available passive node will become the primary active node to access the necessary storage.

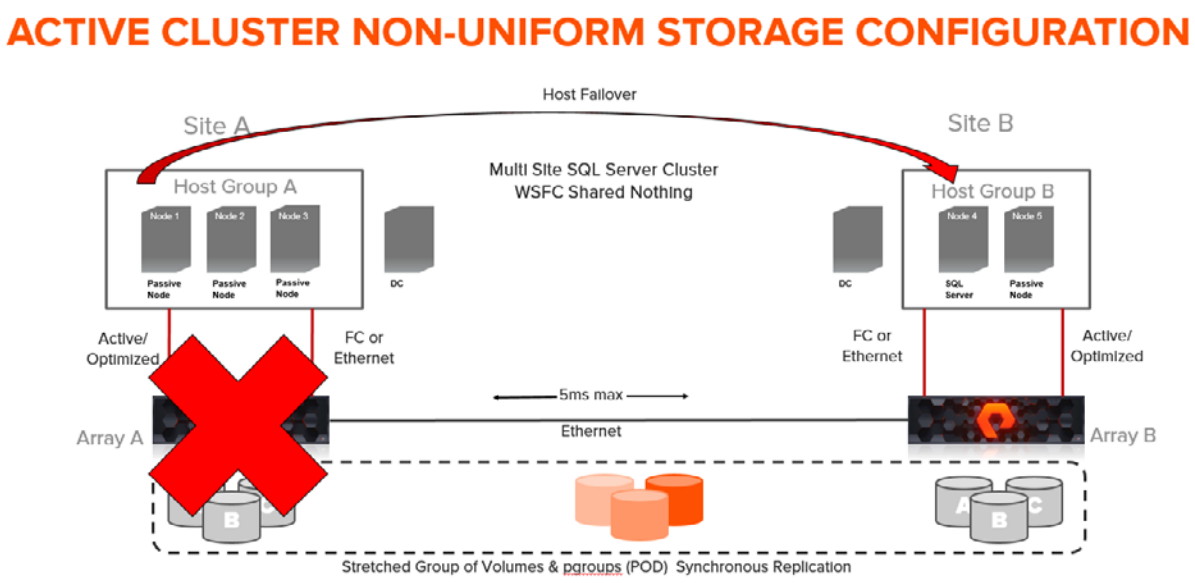


FIGURE 38. WSFC host failover to secondary site in a non-uniform configuration

PATH VISIBILITY NON-UNIFORM CONFIGURATION (example 4 paths Active/Optimized only)

```
PS C:\> (gwmi -Namespace root\wmi -Class mpio_disk_info).driveinfo | % {Write-host "Name: $($_.name) Paths: $($_.numberpaths)"}
Name: MPIO Disk0 Paths: 4
Name: MPIO Disk1 Paths: 4
Name: MPIO Disk2 Paths: 4
Name: MPIO Disk3 Paths: 4
```

```
PS C:\Users\Administrator.SQLACFORREST> mpclaim -s -d

For more information about a particular disk, use 'mpclaim -s -d #' where # is the MPIO disk number.

MPIO Disk     System Disk  LB Policy    DSM Name
-------------------------------------------------------------------------------
MPIO Disk3    Disk 4       LQD          Microsoft DSM
MPIO Disk2    Disk 3       LQD          Microsoft DSM
MPIO Disk1    Disk 2       LQD          Microsoft DSM
MPIO Disk0    Disk 1       LQD          Microsoft DSM
PS C:\Users\Administrator.SQLACFORREST> mpclaim -s -d 2
```

```
PS C:\Users\Administrator.SQLACFORREST> mpclaim -s -d 2

MPIO Disk2: 04 Paths, Least Queue Depth, Implicit Only
    Controlling DSM: Microsoft DSM
    SN: 624A93702EBD55E4363D498400016OF3
    Supported Load Balance Policies: FOO RRWS LQD WP LB

    Path ID          State              SCSI Address      Weight
    -------------------------------------------------------------------
    0000000077020003 Active/Optimized   002|000|003|253   0
      TPG_State : Active/Optimized   , TPG_Id: 0, : 2

    0000000077020002 Active/Optimized   002|000|002|253   0
      TPG_State : Active/Optimized   , TPG_Id: 1, : 18

    0000000077010003 Active/Optimized   001|000|003|253   0
      TPG_State : Active/Optimized   , TPG_Id: 1, : 17

    0000000077010002 Active/Optimized   001|000|002|253   0
      TPG_State : Active/Optimized   , TPG_Id: 0, : 1
```

- Global path information. (PowerShell example)

  ```
  (gwmi -Namespace root\wmi -Class mpio_disk_info).driveinfo | % {Write-host "Name: $($_.name) Paths: $($_.numberpaths)"}
  ```

- Correlate MPIO device to System Disk. (PowerShell example)

  ```
  mpclaim -s -d
  ```

- List individual paths for a specific disk. (PowerShell example)

  ```
  mpclaim -s -d  2
  ```

Uniform Site Storage Failure

In a uniform configuration, if the storage becomes unavailable to the Windows Server primary active node where the SQL Server failover cluster instance is running, this will cause only the I/O paths to change from Active/Optimized to Active/Non-Optimized.

The Windows Server primary active node where the SQL Server failover cluster instance is running will not failover because there are already additional Active/Non-Optimized paths to the secondary site storage that can be utilized.

> **NOTE:** During this I/O path change to Active/Non-Optimized, the processing of I/O performance will degrade momentarily until the path changeover is complete. The time it takes to complete the path changeover is dependent on several factors, such as the active host workload type, network latency, etc. Once the Active/Non-Optimized paths are in use, the processing of I/O will resume to the previous state.

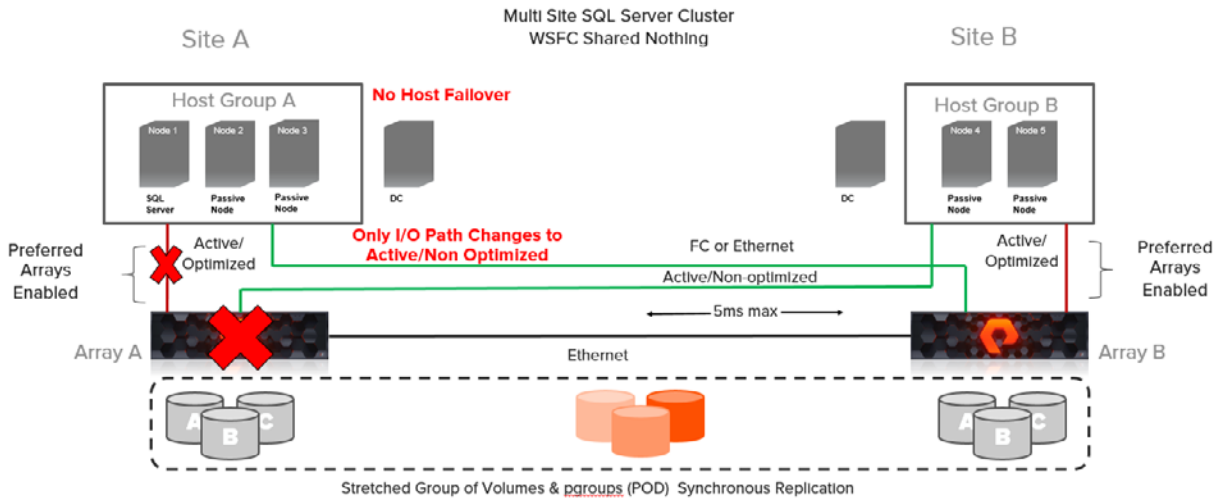


FIGURE 39. WSFC host failover to secondary site in a uniform configuration

PATHS VISIBILITY UNIFORM CONFIGURATION (example 8 paths including additional Active/Unoptimized)

```
PS C:\> (gwmi -Namespace root\wmi -Class mpio_disk_info).driveinfo | % {Write-host "Name: $($_.name) Paths: $($_.numberpaths)"}
Name: MPIO Disk0 Paths: 8
Name: MPIO Disk1 Paths: 8
Name: MPIO Disk2 Paths: 8
Name: MPIO Disk3 Paths: 8
```

```
PS C:\> mpclaim -s -d

For more information about a particular disk, use 'mpclaim -s -d #' where # is the MPIO disk number.

MPIO Disk     System Disk  LB Policy     DSM Name
-------------------------------------------------------------------------------
MPIO Disk3    Disk 4       LQD           Microsoft DSM
MPIO Disk2    Disk 3       LQD           Microsoft DSM
MPIO Disk1    Disk 2       LQD           Microsoft DSM
MPIO Disk0    Disk 1       LQD           Microsoft DSM
```

```
PS C:\> mpclaim -s -d 2

MPIO Disk2: 08 Paths, Least Queue Depth, Implicit Only
    Controlling DSM: Microsoft DSM
    SN: 624A93702EBD55E4363D4984000160F3
    Supported Load Balance Policies: FOO RRWS LQD WP LB

    Path ID          State              SCSI Address      Weight
    -------------------------------------------------------------------------------
    000000007702000b Active/Unoptimized 002|000|011|253   0
      TPG_State : Active/Unoptimized, TPG_Id: 2, : 34

    0000000077020000a Active/Unoptimized 002|000|010|253   0
      TPG_State : Active/Unoptimized, TPG_Id: 3, : 50

    0000000077020003 Active/Optimized   002|000|003|253   0
      TPG_State : Active/Optimized   , TPG_Id: 0, : 2

    0000000077020002 Active/Optimized   002|000|002|253   0
      TPG_State : Active/Optimized   , TPG_Id: 1, : 18

    000000007701000b Active/Unoptimized 001|000|011|253   0
      TPG_State : Active/Unoptimized, TPG_Id: 3, : 49

    000000007701000a Active/Unoptimized 001|000|010|253   0
      TPG_State : Active/Unoptimized, TPG_Id: 2, : 33

    0000000077010003 Active/Optimized   001|000|003|253   0
      TPG_State : Active/Optimized   , TPG_Id: 1, : 17

    0000000077010002 Active/Optimized   001|000|002|253   0
      TPG_State : Active/Optimized   , TPG_Id: 0, : 1
```

- Global path information. (PowerShell example)

```
(gwmi -Namespace root\wmi -Class mpio_disk_info).driveinfo | % {Write-host "Name: $($_.name) Paths: $($_.numberpaths)"}
```

- Correlate MPIO device to System Disk. (PowerShell example)

```
mpclaim -s -d
```

- List individual paths for a specific disk. (PowerShell example)

```
mpclaim -s -d  2
```

PATHS VISIBILITY UNIFORM CONFIGURATION (post path changeover 4 paths only Active/Unoptimized)

- Performance degradation due to Site A primary storage becoming unavailable, as seen on the FlashArray B secondary site. (This is an example; performance degradation metric behavior may vary.)

FIGURE 40. Performance degradation with Site A unavailable

- Windows Server host Site A system event alerts indicating that path changeover operations have occurred (example).

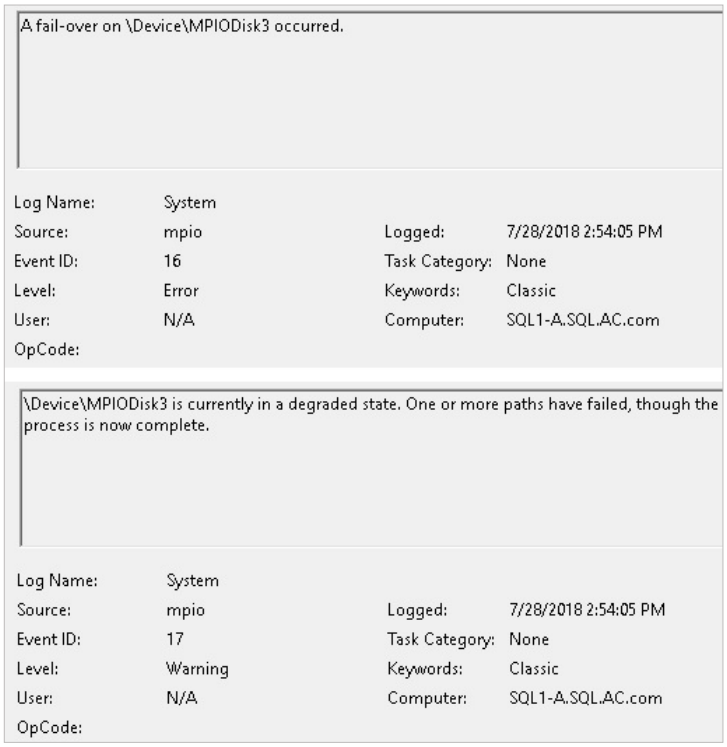
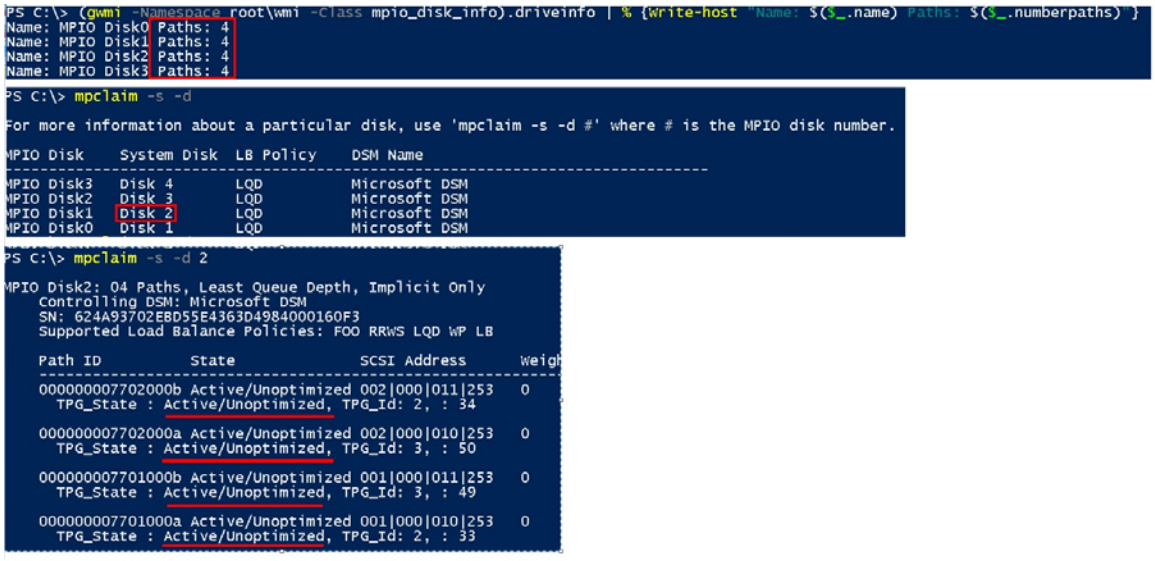


FIGURE 41. Windows Server event alerts

- Windows host path visibility Site A primary during changeover now listing only four paths, which previously was eight paths. Only the paths have changed using the Active/Unoptimized. No host failover has occurred which is still at Site A now using Site B FlashArray storage.



FlashArray Storage Access

The following table describes different failure scenarios for ActiveCluster environments and how storage availability is affected.

SOLUTION COMPONENT FAILURE				ACCESS TO STORAGE
ONE ARRAY	OTHER ARRAY	REPLICATION LINK	MEDIATOR	
UP	UP	UP	UP	AVAILABLE ON BOTH ARRAYS
UP	DOWN	UP	UP	AVAILABLE ON ONE ARRAY
UP	UP	DOWN	UP	AVAILABLE ON ONE ARRAY
UP	UP	UP	DOWN	AVAILABLE ON BOTH ARRAYS
UP	DOWN	DOWN	UP	AVAILABLE ON ONE ARRAY
UP	UP	DOWN	DOWN*	UNAVAILABLE
UP	DOWN	UP	DOWN*	UNAVAILABLE
DOWN	DOWN	N/A	N/A	UNAVAILABLE

* These rows refer to simultaneous failures of other components while the mediator is unavailable. If the mediator becomes unavailable after an array failure or a replication link failure has already been sustained, access to the mediator is no longer required and access to storage remains available on one array.

TABLE 1. Storage availability under failure scenarios

The chart on the next page explains some of the detailed failure scenarios for ActiveCluster and the surrounding network.

Host and Storage Network Failures

FAILURE SCENARIO	FAILURE BEHAVIOR
SINGLE OR MULTIPLE HOST FAILURE	Applications can automatically failover to other hosts in the same site or to other hosts in the other site connected to the other array. This is driven by WSFC assuming clusters are stretched between sites.
STRETCHED SAN FABRIC OUTAGE (FC OR ISCSI) **(FAILURE OF SAN INTERCONNECT BETWEEN SITES)**	Host IO automatically continues on local paths in the local site. Uniform connected hosts: • Experience some storage path failures for paths to the remote array and continue IO on paths to the local array. • In each site will maintain access to local volumes with no more than a pause in IO. Non-uniform connected hosts: • Do not have a SAN interconnect between sites, so this scenario is not applicable.
SAN FABRIC OUTAGE IN ONE SITE	Applications can automatically failover to hosts at the other site connected to the other array. This is driven by host cluster software assuming clusters are stretched between sites. VMware HA, Oracle RAC, SQL Cluster, etc. Uniform connected hosts: • In the site without the SAN outage, experience some storage path failures for paths to the remote array and continue IO on paths to the local array. • In the site with the SAN outage, will experience total loss of access to volumes and applications must failover to the other site as mentioned above. Non-uniform connected hosts: • In the site without the SAN outage, will maintain access to local volumes. • In the site with the SAN outage, will experience total loss of access to volumes and applications must failover to the other site as mentioned above.

TABLE 2. Detailed failure scenarios for ActiveCluster

CONCLUSION

The combination of ActiveCluster active-active replication on FlashArray, WSFC (Windows Server Failover Cluster), and SQL Server Failover Instance Cluster, collectively refer to a solution called SQL Server Active Cluster. The simplicity and flexibility of FlashArray itself, and the ActiveCluster feature, enable administrators to configure and offer a highly-available storage solution with ease.

REFERENCES

ActiveCluster Best Practices

ActiveCluster Solution Overview

ps_wp41p_sql-server-with-activecluster_ltr_01