TECHNICAL WHITE PAPER

Microsoft Hyper-V Stretched Cluster with ActiveCluster

Get an overview of configuration and best practices for using Pure Storage® FlashArray™ ActiveCluster™ with a stretched Hyper-V failover cluster.

Contents

Executive Summary

Business-critical applications and the virtual machines hosting them often need the highest possible resiliency to ensure that business operations do not stop in the case of a disaster—either localized or site-wide. To ensure this, you need to spread the data in use by those applications across two arrays, often in more than one geographic location. Importantly, the data must be available in both sites, at the same time. To achieve this, some arrays offer synchronous replication that 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® introduced active-active replication for FlashArray™ in the Purity™ 5.0.0 release. This paper provides an overview of configuration and best practices for using the Pure FlashArray ActiveCluster™ feature with a stretched Hyper-V failover cluster.

Audience

This paper is intended for storage, Hyper-V, network, or other administrators who plan to implement Pure Storage FlashArray ActiveCluster with a stretched Hyper-V failover cluster. For specific questions or requests for assistance, please reach out to Pure Storage support at any time at support@purestorage.com.

Note: *Before setting up ActiveCluster and a stretched Hyper-V failover cluster, it's important to read the ActiveCluster-specific documentation from Pure as well as Microsoft's own documentation. This document assumes that you've read the following three documents:*

— Failover Clustering in Windows Server

— ActiveCluster Requirements and Best Practices

— ActiveCluster Solution Overview

Solution Overview

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

- **Microsoft Windows Failover Clustering:** A solution that increases the scalability and availability of clustered roles, such as virtual machines, file servers, and SQL Server failover cluster instances
- **Pure Storage FlashArray ActiveCluster:** A simple, built-in, active-active synchronous storage replication solution for FlashArray block storage

Microsoft Windows Failover Clustering

Microsoft Windows Failover Clustering provides cluster-based monitoring of clustered roles running on Windows hosts. In the event of a failure of the storage, network, or host, the remaining Windows hosts coordinate to restart the affected clustered roles on surviving hosts. Through a variety of network and storage-based heart-beating, the Windows hosts can detect and respond to various failure or isolation events to provide the fastest automated recovery of clustered roles.

Virtual machines (VMs) that are configured as highly available show up as roles in Failover Cluster Manager and are protected from server issues because the VM will be failed over to another node in the cluster. See "Configuring VM High Availability" for details on how to configure new and existing VMs.

ActiveCluster

Purity ActiveCluster is a fully symmetric active-active bidirectional replication solution that provides synchronous replication for RPO zero and automatic transparent failover for RTO zero. ActiveCluster spans multiple sites, which enables you to use clustered arrays and hosts to deploy flexible active-active data center configurations.

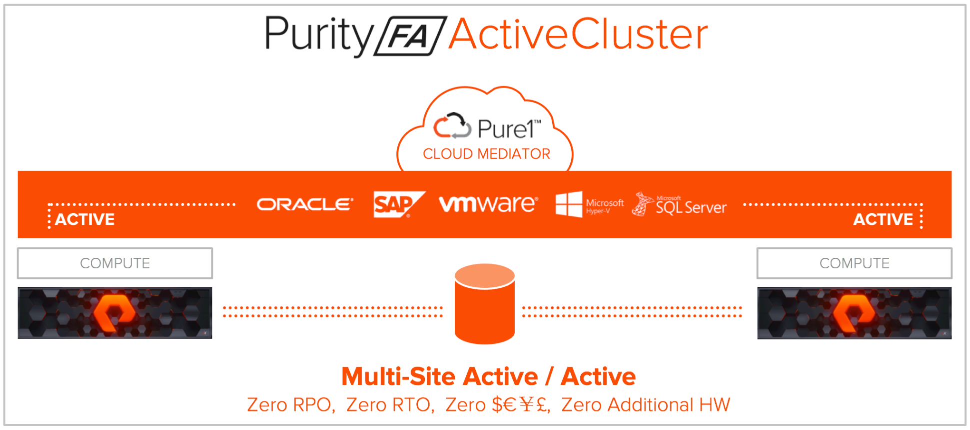


Figure 1: Purity FA ActiveCluster is active-active with a Pure1 Cloud Mediator..

- **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:** Provides automatic non-disruptive failover between synchronously replicating arrays and sites with automatic resynchronization and recovery.

- **Async replication integration:** Uses async for baseline copies and resynchronizing. Convert async relationships to sync without resending data. Async data to a third site for disaster recovery (DR).
- **No bolt-ons, no licenses:** There's no additional hardware or 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, plus provision storage, connect hosts, create snapshots, and create clones.
- **Integrated Pure1® Cloud Mediator:** The automatically configured passive mediator allows transparent failover and prevents split-brain, without the need to deploy and manage another component.

Components

Purity ActiveCluster comprises the following three core components:

- **Pure1 Cloud Mediator:** A required solution component used to determine which array will continue data services should an outage occur in the environment. As an alternative, you can install a local mediator as a virtual machine.
- **Active-active clustered FlashArrays:** Utilize synchronous replication to maintain a copy of data on each array that presents as a single 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.

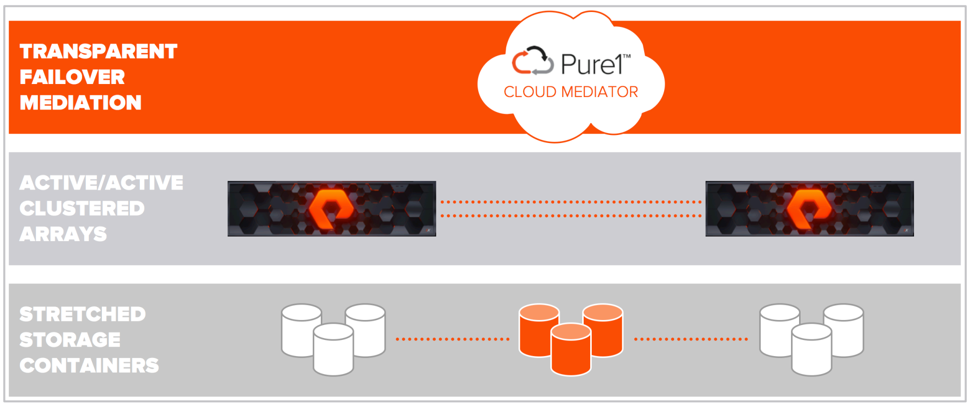


Figure 2: ActiveCluster Components.

Administration

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

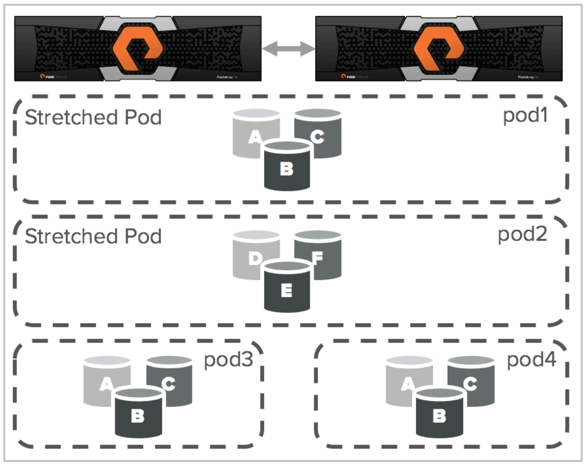


Figure 3: Stretched and unstretched pods.

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 Figure 3, the volumes in pod3 and pod 4 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 doesn't require 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 case where two arrays might serve I/O to the same volume without keeping the data synchronized between the two arrays.

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

The Pure1 Cloud Mediator ensures that only one array can stay active for each pod when there is a loss of communication between the arrays.

If 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 identify which array can stay active for each sync-replicated pod. The first array to reach the mediator can keep its synchronously replicated pods online. The second array 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 resume of I/O.

The Pure1 Cloud Mediator

A failover mediator must be in a third site that exists in a separate failure domain than either site where the arrays are located. Each array site must have independent network connectivity to the mediator so that a single network outage does not prevent both arrays from accessing the mediator. A mediator should also provide a very lightweight and easy-to-administer component of the solution. Pure1 Cloud Mediator provides this automatically by utilizing an integrated cloud-based mediator. Pure1 Cloud Mediator has several advantages over a typical non-Pure, heavy-handed voter or witness component:

- **SaaS operational benefits:** As with any SaaS solution, it removes the complexity of operational maintenance: There's nothing to install onsite, no hardware or software to maintain, nothing to configure and support for HA, no security patch updates, and so on.
- **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:** Automatic and default configuration eliminate the risk of incorrect configuration of the mediator.
- **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. The 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. If they lose connection to the mediator they will continue to synchronously replicate and serve data if the replication link is active.

On-Premises Failover Mediator

You can also provide failover mediation for ActiveCluster using an on-premises mediator distributed as an Open Virtualization Appliance (OVA) file and deployed as a VM. An OVA is a single-file version of the Open Virtualization Format (OVF), that can be converted into a Virtual Hard Disk (VHDx) for Hyper-V use. Failover behaviors are the same as those for Pure1 Cloud Mediator. The on-premises mediator simply replaces the role of Pure1 Cloud Mediator during failover events. You must deploy an on-premises mediator on an ESXi or Hyper-V host that is not running on either FlashArray in the active-active cluster that it is monitoring. (Note: Get instructions for converting the OVA into a VHD file to run on a Hyper-V host.)

The on-premises mediator has the following basic requirements:

- You can only deploy the mediator as a VM on virtualized hardware. It is not installable as a stand-alone application.
- The hosts on which the mediator is deployed provide high availability for the mediator. For example, using Microsoft Hyper-V failover clustering, or VMware HA.
- Storage for the mediator must not allow the mediator's configuration to be rolled back to previous versions. This applies to situations such as storage snapshot restores, or cases where it might be stored on mirrored storage.

- You must configure the arrays to use the on-premises mediator rather than Pure1 Cloud Mediator.
- You must deploy the mediator 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 both arrays. One is called uniform access, the other is non-uniform access.

Uniform Access

You can use a uniform storage access model in environments where there is host-to-array connectivity of either fiber channel 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 and the remote array. The solution supports connecting arrays with up to 5ms of round-trip time (RTT) latency between the arrays.

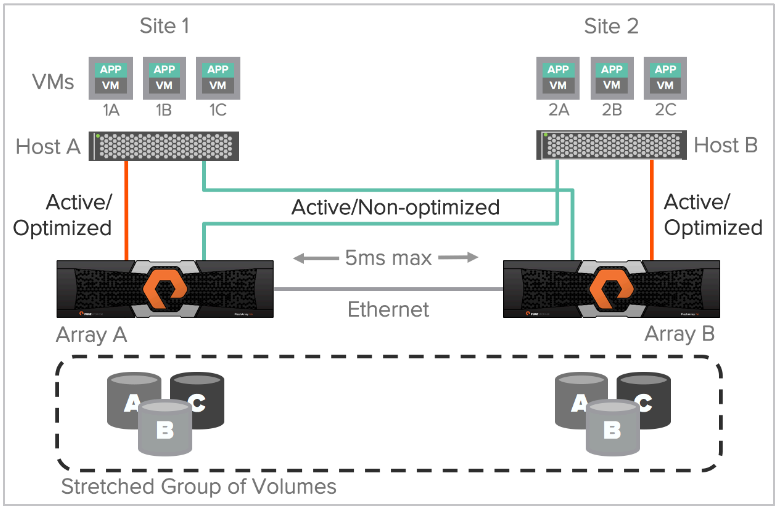


Figure 4: Uniform Access RTT latency maximum increased to 11ms with Purity 5.1.2.

Figure 4 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, the paths will have 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 environments, prevent the hosts from using paths that access the remote array unless necessary. For example, in Figure 4, if VM 2A were to perform a write to volume A over the host side connection to array A, that write would incur twice the latency of the inter-site link, once for each traverse of the network. The write would experience 5ms of latency for the trip from host B to array A plus 5ms of latency while array A synchronously sends the write back to array B.

There are no such management headaches with Purity ActiveCluster. ActiveCluster makes use of Asymmetric Logical Unit Access (ALUA) to expose paths to local hosts as active/optimized paths. It exposes paths to remote hosts as active/non-optimized. However, there are two advantages in the ActiveCluster implementation.

- 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.
- The optimized path is defined on a per host-to-volume connection basis using a preferred-array option. This ensures that the host will have a local optimized path to that volume regardless of the VM or application on which it's running. Ensure that hosts in the uniform configuration have the preferred path set on the FlashArrays in both sites.

With ActiveCluster you may configure truly active-active data centers. You don't have to care on what site or host a VM runs. The VM will always have the same performance regardless of site. While VM 1A is running on host A and accessing volume A, it will use only the local optimized paths as shown in the next image.

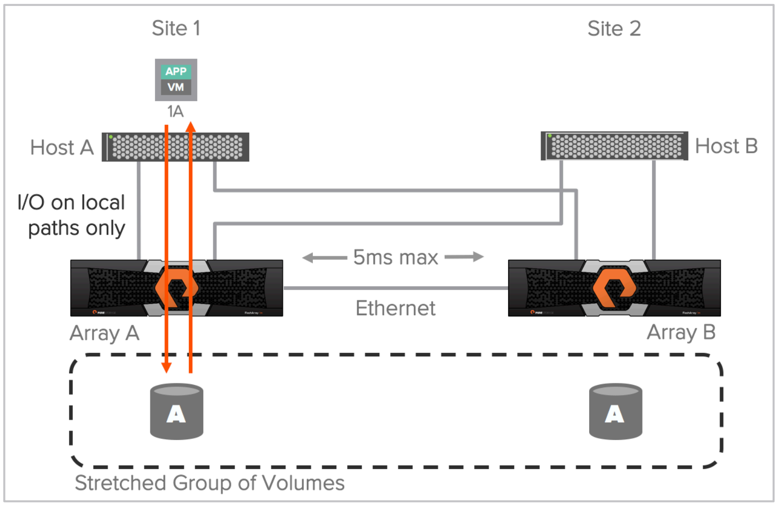


Figure 5: Site 1 Server reads are on the optimized path within the site on Array A.

If the VM or application is switched to a host in the other site with the data left in place, it will use only local paths in the other site. There is no need to adjust path priorities or migrate 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 Fiber Channel or Ethernet (for iSCSI) locally only 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 5ms of RTT latency between the arrays. Hosts will distribute I/O across all paths to the storage only, because only the local Active/Optimized paths are available (Figure 6).

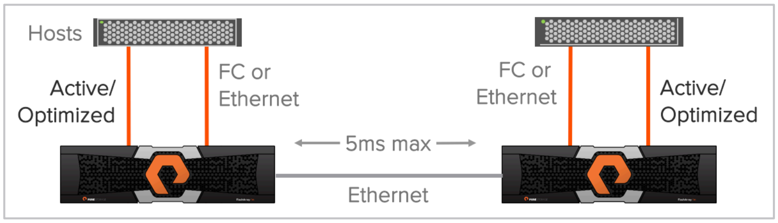


Figure 6: In a non-uniform configuration, the hosts only see the local storage.

Configuring ActiveCluster

A major benefit of using Purity ActiveCluster for your stretched storage solution is the simple set up. Before moving forward, ensure that you've followed the environment configuration requirements.

Step 1. Create a Synchronous Connection

The first step in enabling ActiveCluster is creating a synchronous connection with another FlashArray. It does not matter which FlashArray you use to create the connection—either one is fine.

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

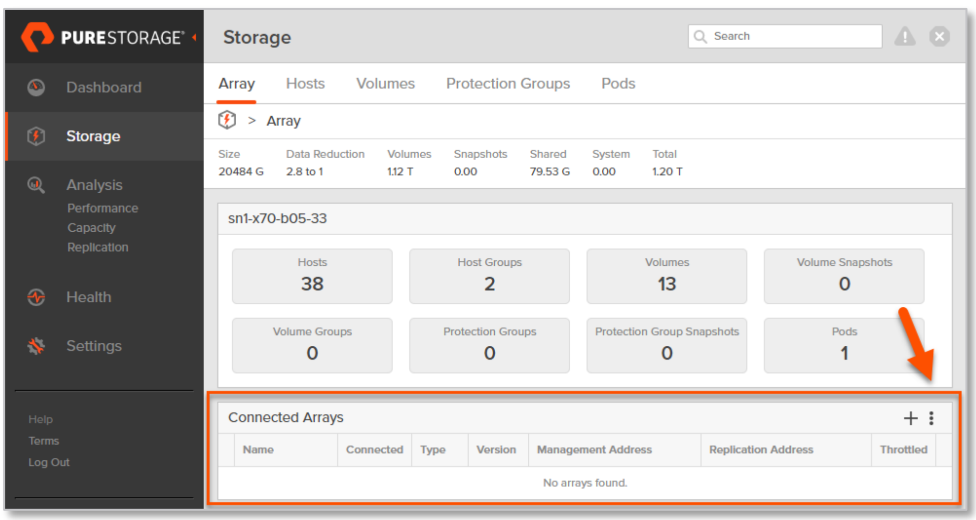


Figure 7: Connect the target array in the FlashArray web interface.

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

- **Management address** is the **virtual IP address** or Fully Qualified Domain Name **(FQDN)** of the remote FlashArray.
- For **Connection type,** choose **Sync Replication** for ActiveCluster.
- **Connection Key** is an **API token** that you can retrieve from the remote FlashArray.

To obtain the connection key, log in to the remote FlashArray web Interface and click on the **Storage** section. Then click on the **vertical ellipsis** and choose **Get Connection Key**.

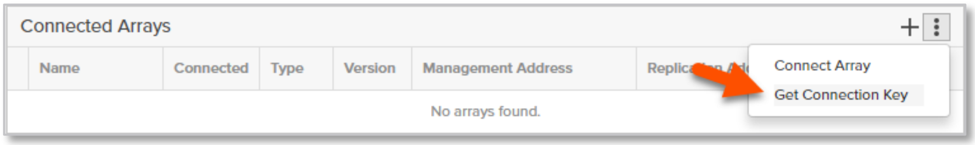


Figure 8: On the target array, get the Connection Key.

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

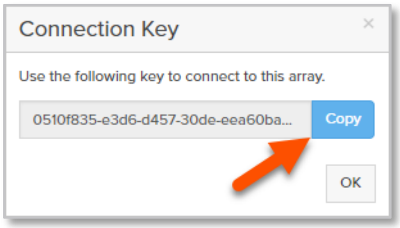


Figure 9: On the target array, copy the Connection Key.

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

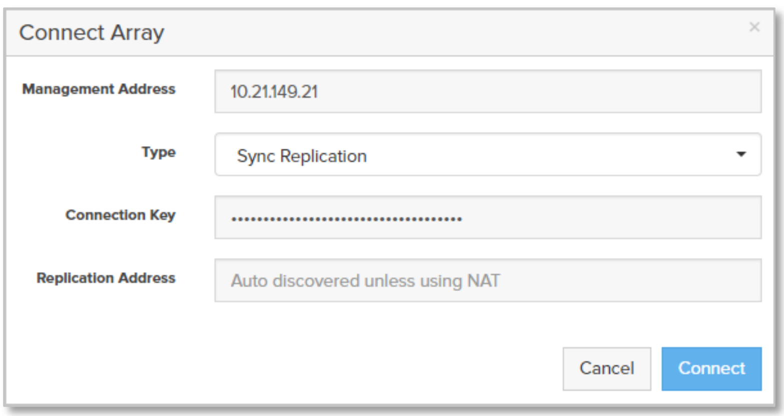


Figure 10: On the source array, add the target array's Connection Key.

Click on Connect when you have filled out information (Figure 10). There is generally no need to enter anything for the replication address. If everything is valid, the connection will appear in the **Connected Arrays** panel (Figure 11).

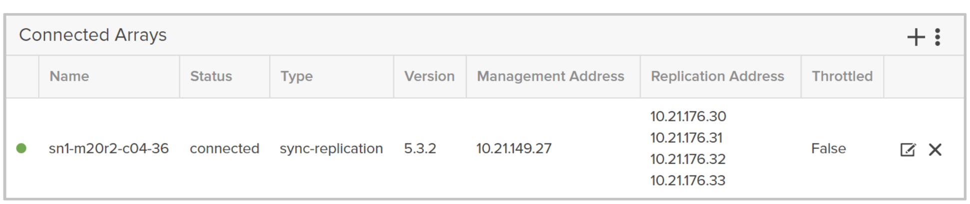


Figure 11: Both FlashArrays will now show that a connection is made.

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

Step 2. Creating a Pod

The next step to enable ActiveCluster is to create a consistency group, or pod. A pod is both a consistency group and a namespace—in effect creating a grouping for related objects in ActiveCluster replication. You can create one or more pods. Pods are stretched, un-stretched, and failed over together.

The basic idea for one or more pods is simply to put related volumes in the same pod. For instance, put them in the same pod If they host related applications that should remain in the same data center or have consistency with one another. Otherwise, put them in the same pod for simplicity, or different ones if they have different requirements.

To create a pod, log in to the FlashArray GUI and click on **Storage**, then **Pods**, then the **plus sign**.

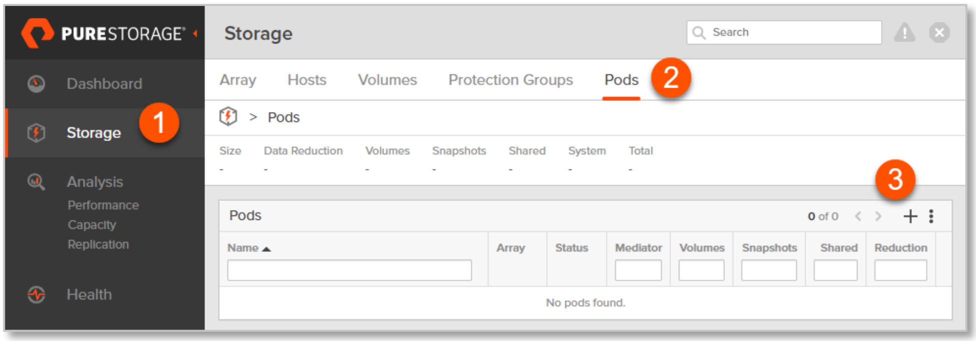


Figure 12: Add a Pod to the FlashArray.

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

Figure 13: Create a Pod.

The pod will then appear in the **Pods** panel (Figure 14).

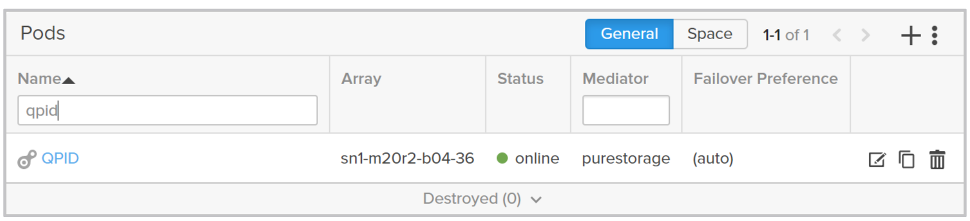


Figure 14: The newly created Pod.

To further configure the pod, click on the pod name (Figure 14).

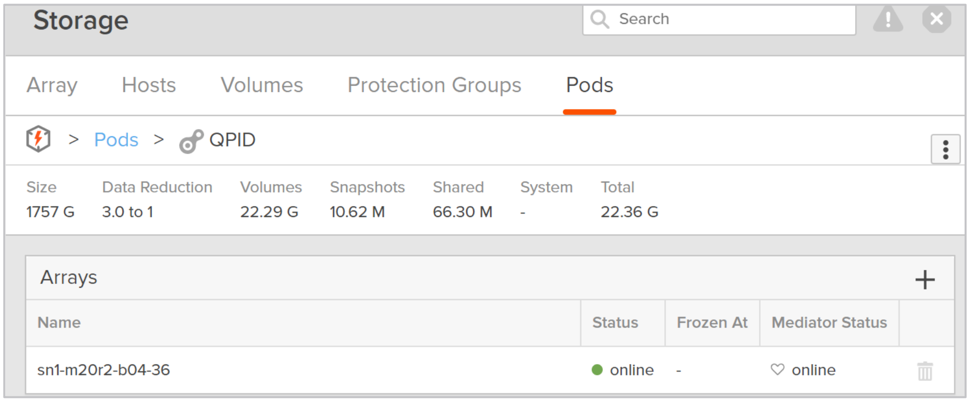


Figure 15: Status of selected Pod, shows a single array.

The default configuration for ActiveCluster is to use the Cloud Mediator. There's no configuration required other than to ensure that 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 best practices document. for more details.)

You can see the mediator status in the overview panel under the Mediator Status heading. Under Details, the mediator is listed as "purestorage" if the Cloud mediator is in use. For configuring a on-premises mediator (in the case of a dark site configuration), refer to ActiveCluster documentation.

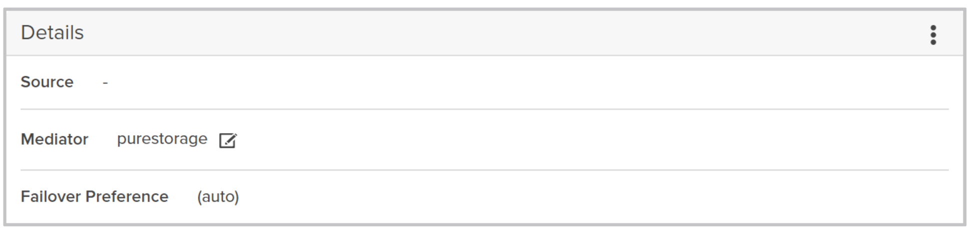


Figure 16: Mediator Status shows 'purestorage', or the Pure1 Cloud Mediator.

Pod configuration is complete.

Step 3. Add Volumes to a Pod

The next step is to add any pre-existing volumes on which you would like to enable ActiveCluster to the pod. Once you've enabled a pod for replication, you can't move pre-existing volumes into it. You can only create new volumes in the pod.

To add a volume to a pod, go to the **Storage** screen in the FlashArray web interface. Click on **Volumes,** then click on the **name** of the volume you would like to add to the pod. You can search by name to find the volume quickly.

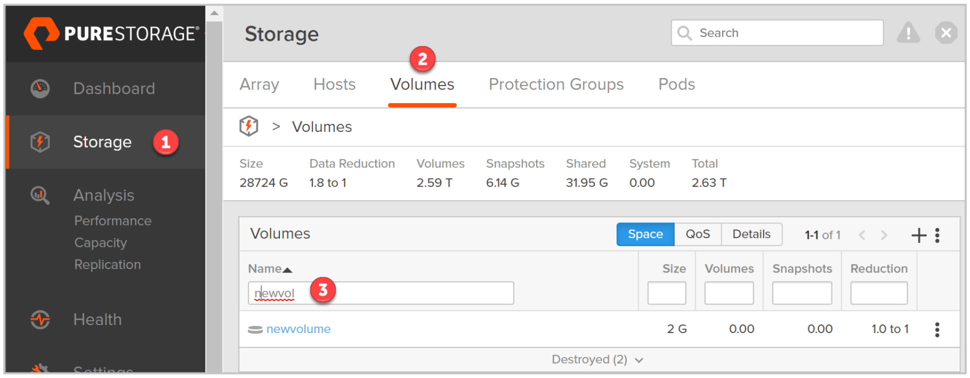


Figure 17: Select a Volume.

When the volume screen loads, click on the **vertical ellipsis** in the upper right-hand corner and choose **Move** (Figure 18).

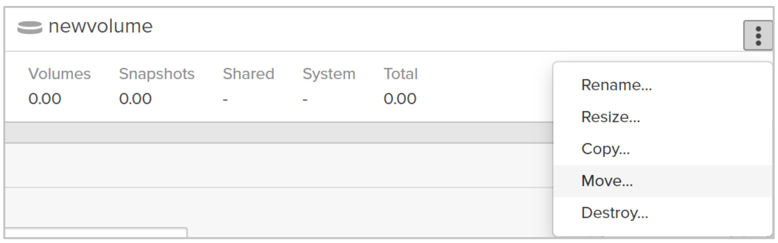


Figure 18: Move a Volume to a Pod.

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

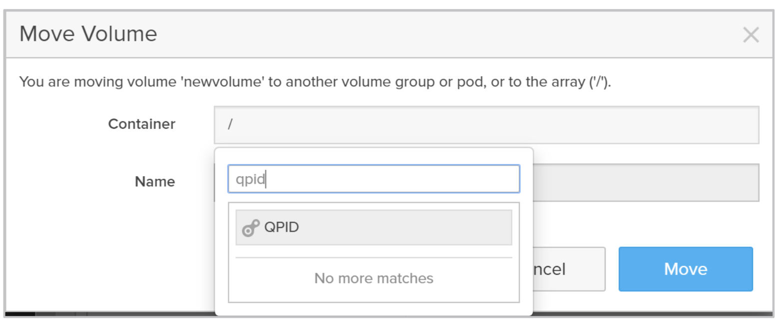


Figure 19: Select the target Pod.

Note: As of Purity 5.0.0, the following limitations can impact moving a volume into a pod:

You can't add a volume that is already in a volume group into a pod. You must first remove it from the volume group.

A volume that is already in a stretched pod must have the pod unstretched before it can be moved.

If the target pod is stretched, you must first unstretch the pod before you can add an existing volume into it. Some of these restrictions may relax in future Purity versions.

— Purity 5.3.1 Release Notes – RTT maximum latency increased from 5->11ms

— Purity 5.3.2 Release Notes

— Purity 5.3.3 Release Notes

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

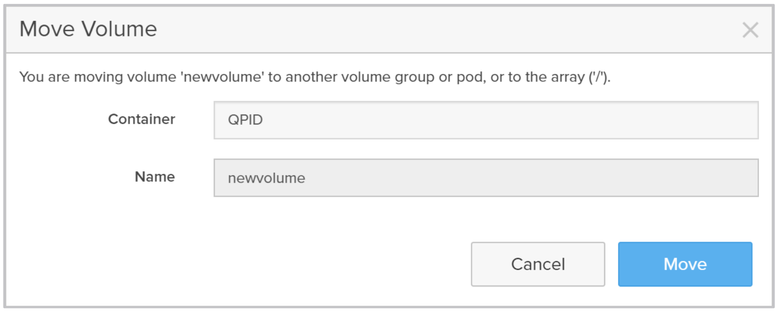


Figure 20: Move an existing volume into an existing Pod.

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

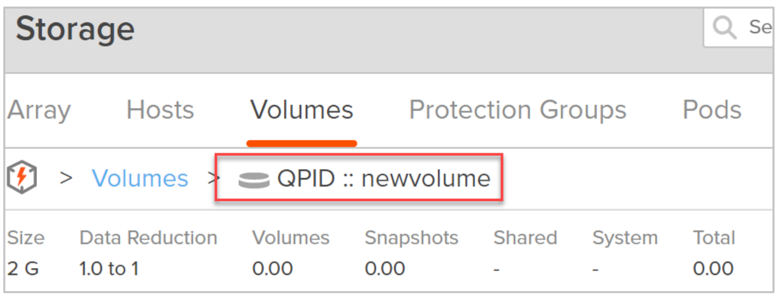


Figure 21: A Volume has been moved inside of a Pod and has a new name, Pod::Volume.

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

Creating a New Volume in a Pod

You can also create new volumes directly in a pod. Click on **Storage**, then **Pods**, then choose your pod. Under the **Volumes** panel, click the **plus sign** to create a new volume.

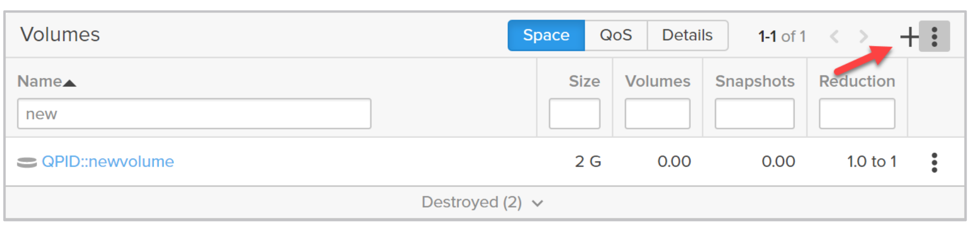


Figure 22: Create a new Volume in an existing Pod.

In the creation window, enter a valid name and a size, and then click **Create**. You can do this whether or not the pod is actively replicating.

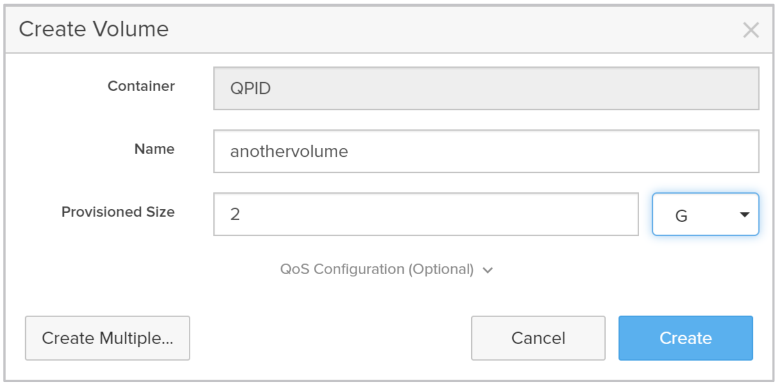


Figure 23: Select the existing Pod, Volume name, and size.

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.

Note: Once a pod has been "stretched", you cannot add pre-existing volumes to it until it is "un-stretched." In other words, once you've stretched a pod, you can only create new volumes within it. If you want to add existing volumes to the pod, follow the instructions in the section Step 3. Add Volumes to a Pod, then stretch the pod.

To stretch a pod, add a second array to it by clicking on the **plus sign** in the **Arrays** panel.

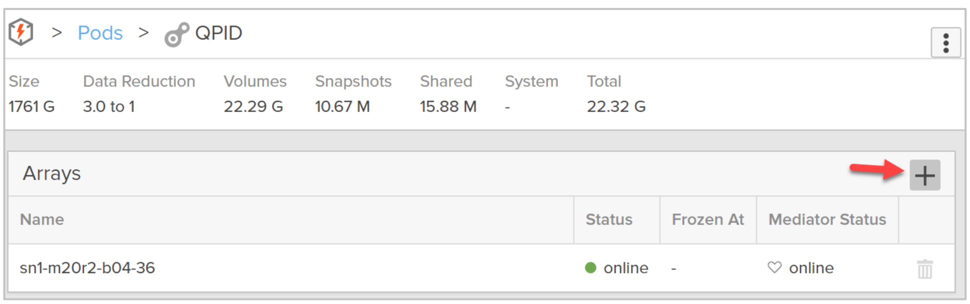


Figure 24: Stretch an existing Pod.

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

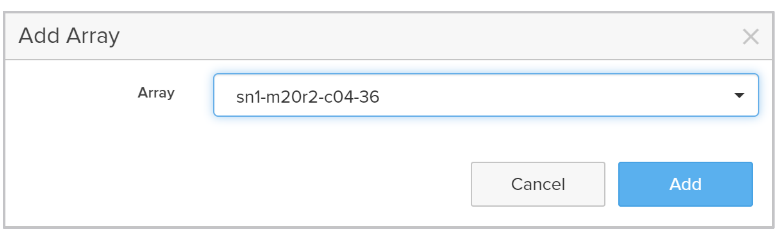


Figure 25: Select the target FlashArray.

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

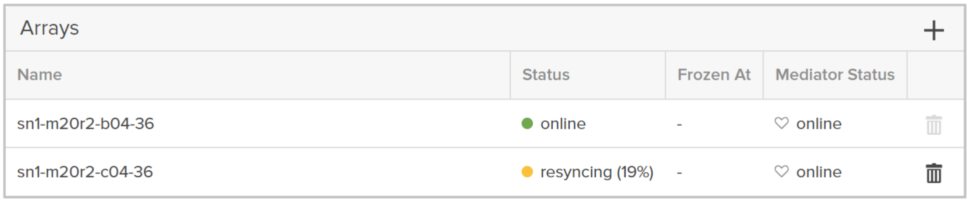


Figure 26: See the status of all FlashArrays configured for the Pod.

Active-active storage isn't available until the synchronization completes, which is shown when the resyncing status ends and both arrays are online.

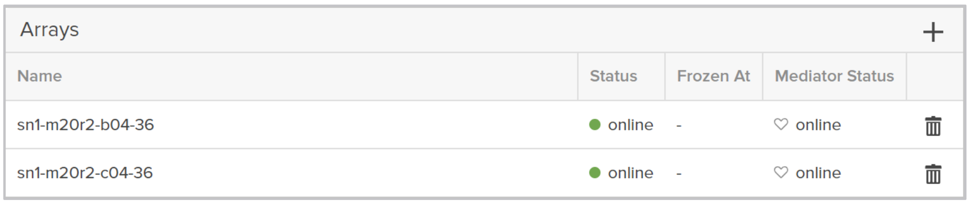


Figure 27: After the initial sync, both FlashArrays should be green and online.

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

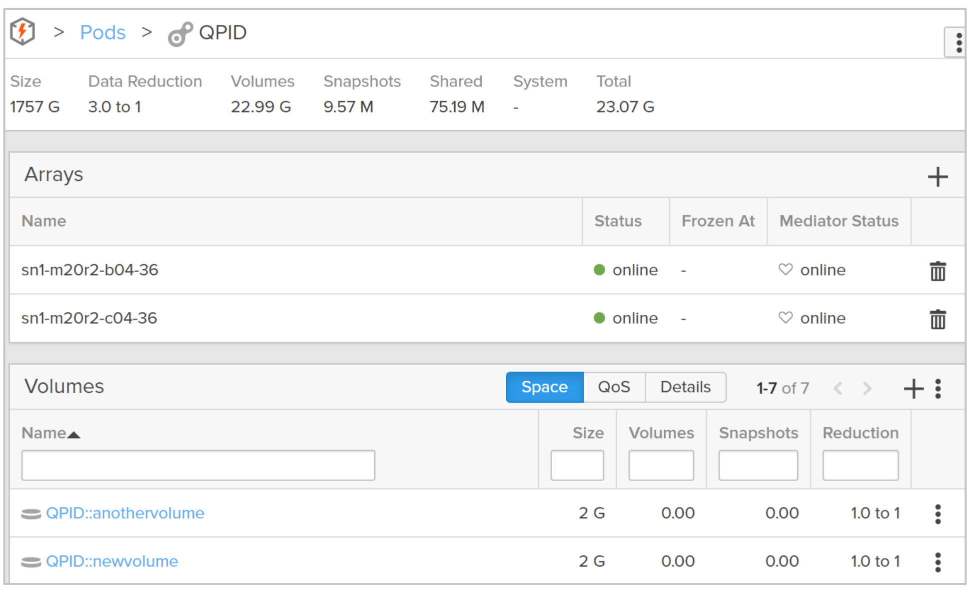


Figure 28: The remote FlashArray now shows the stretched Pod.

Un-stretching a Pod

Once you've enabled ActiveCluster, you may want to terminate the replication, also called "un-stretching." You might want to change the pod volume membership or might have temporarily enabled replication to migrate the volumes from one array to another. To un-stretch a pod, remove the array for which you no longer need the volumes. For example, consider the pod in Figure 29.

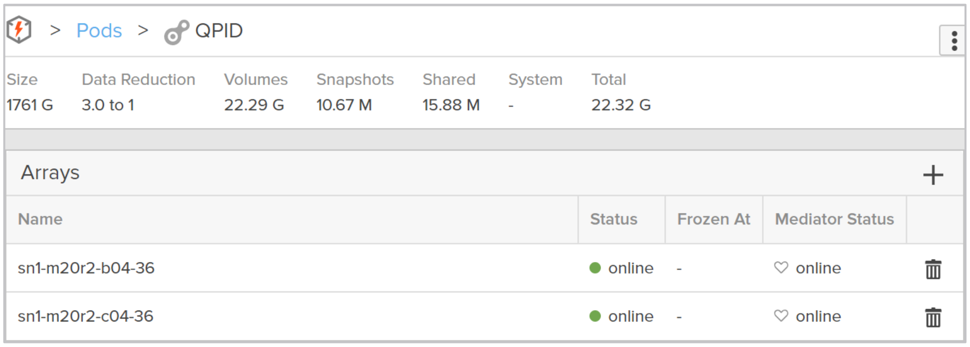


Figure 29: Unstretch a Pod by selecting the trash can icon next to the array it should be removed from.

The pod has two arrays: sn1-m20r2-b04-36 and sn1-m20r2-c04-36. Because the pod is online, both arrays offer up the volumes within it. To keep the volumes only on sn1-m20r2-b04-36, remove the other FlashArray (sn1-m20r2-c04-36). Before removing a FlashArray from a pod, ensure:

- Any hosts using those volumes have access to the FlashArray that you intend to keep. If hosts have access to only the pod you plan to remove, they will lose access to their volumes once you remove it from the pod.
- You are removing the correct FlashArray. Double check the array before removing it.

Note: *Ensure you do not remove the wrong FlashArray from a pod, or hosts may lose access to their storage.*

To remove a FlashArray, choose the appropriate pod. From the Arrays panel, in Figure 29, click on the trash icon next to the array you would like to remove.

When you've verified that it's the correct array, click on Remove to confirm the removal.

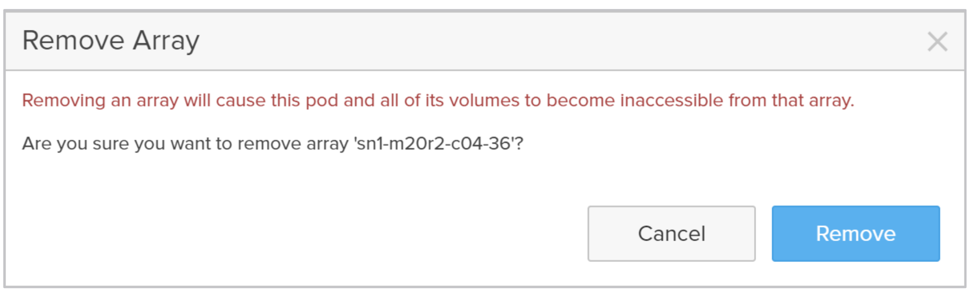


Figure 30: Remove an Array from a stretched Pod to unstretch it.

On the FlashArray you've removed, you will see that pod listed in the **Destroyed Pods** panel under the **Pods** tab.

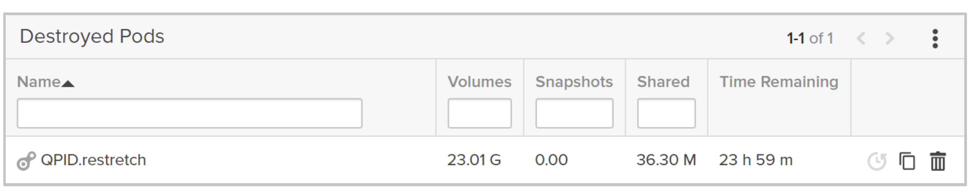


Figure 31: Destroyed Pods are listed under the Pods tab.

If you performed an un-stretch in error, go back to the remaining FlashArray web interface in the pod and add the other FlashArray to move the pod from **Destroyed Pods** status back to active. You can instantly re-stretch the pod for 24 hours.

After that, the removed FlashArray permanently removes its references to the pod. You can force permanent removal early by selecting the destroyed pod and selecting the trash icon next to it.

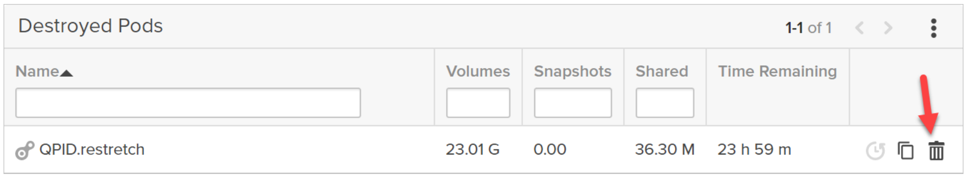


Figure 32: Delete the destroyed Pod.

Click **Eradicate** to confirm the removal. Be warned: *Eradicate* will instantly and permanently destroy the object being eradicated (pod, host, host group, snapshot, volume, etc.).

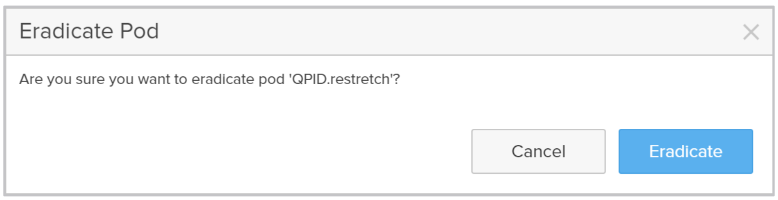


Figure 33: Deleting a Destroyed object, permanently Eradicates it.

Configuring Hyper-V Hosts

Configuring the Hyper-V hosts is no different than configuring non-ActiveCluster FlashArrays, so all best practices described in the Microsoft quick-step guide still apply. There are still a few things worth mentioning here.

Host Connectivity

The FlashArray supports connections through iSCSI and Fibre Channel. Both are supported with ActiveCluster. Prior to being able to provision storage, you must create a “host” object on the FlashArray for each Hyper-V server. You can then further organize hosts into host groups. Host groups collect host objects so that you can provision to them all at once. Host groups are recommended for clustered hosts, like Hyper-V, so that you can provision storage uniformly to all hosts.

Hosts

A FlashArray host object is a logical container that includes host ports, connected volumes, protection groups, preferred arrays, CHAP credentials and personality. Access is granted to the volumes once they’re connected to the FlashArray host object. This allows those specified initiators (and therefore that host) to access the specified volume or volumes. Create a host object on the FlashArray by going to the **Storage** section and then the **Hosts** tab.

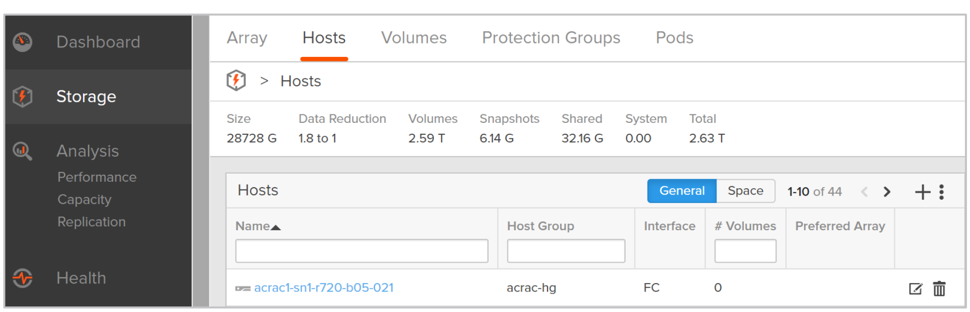


Figure 34: Create a new Host, on the Hosts tab.

Click on the **plus sign** in the **Hosts** panel to create a new host. Assign the host a name that makes sense and click **Create**.

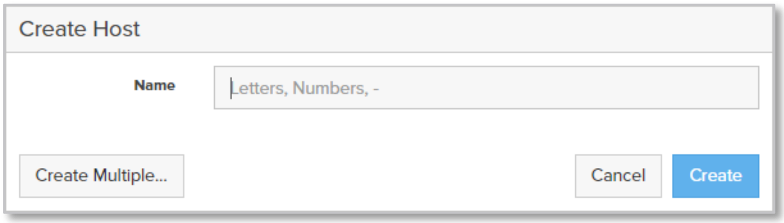


Figure 35: Name the new Host.

Click on the newly created host. In the Host Ports panel, click the **vertical ellipsis** and choose either **Configure WWNs** (for Fibre Channel) or **Configure IQNs** (for iSCSI).

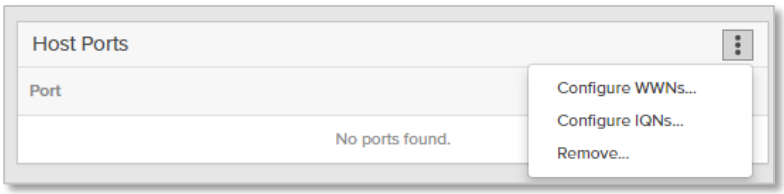


Figure 36: Add Host Ports to the Host.

For WWNs, if the initiator is presented on the fabric to the FlashArray (zoning is complete), click the correct WWN in the left pane to add it to the host or click the **plus sign** to type it manually. iSCSI IQNs must be typed in manually. On the Microsoft iSCSI Initiator configuration tab, you can highlight the Initiator Name, copy it, and then paste it into the Configure IQN section when adding a FlashArray host port to a host.

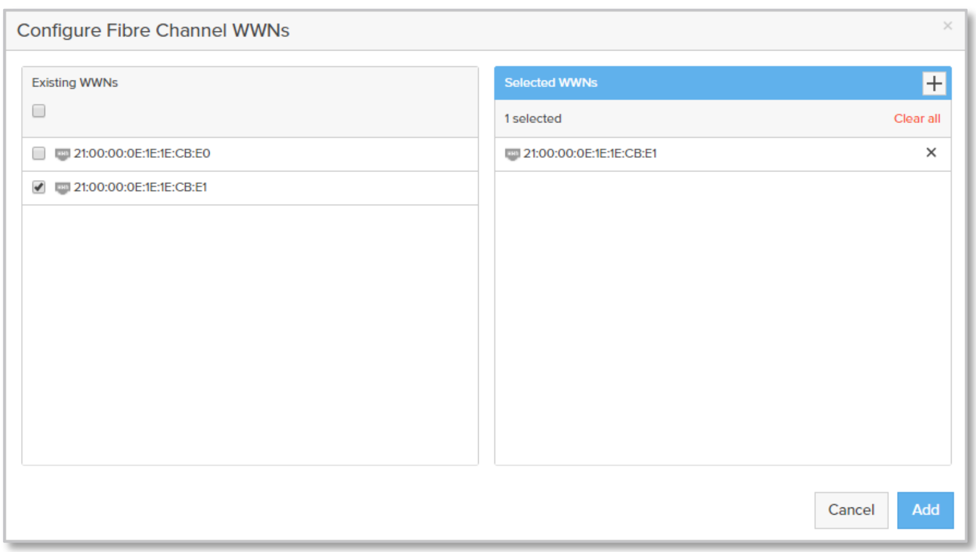


Figure 37: Select the Fibre Channel WWNs.

When all the initiators are added/selected, click **Add**.

Note: All Fibre Channel hosts should have at least two initiators for redundancy. Hyper-V iSCSI usually has only one initiator (IQN) but should have two or more physical NICs in the host that can talk to the FlashArray iSCSI targets.

Verify connectivity by navigating to the **Health** section and then the **Connections** tab. Find the newly created host and look at the **Paths** column. If it lists anything besides **Redundant**, investigate the reported status.

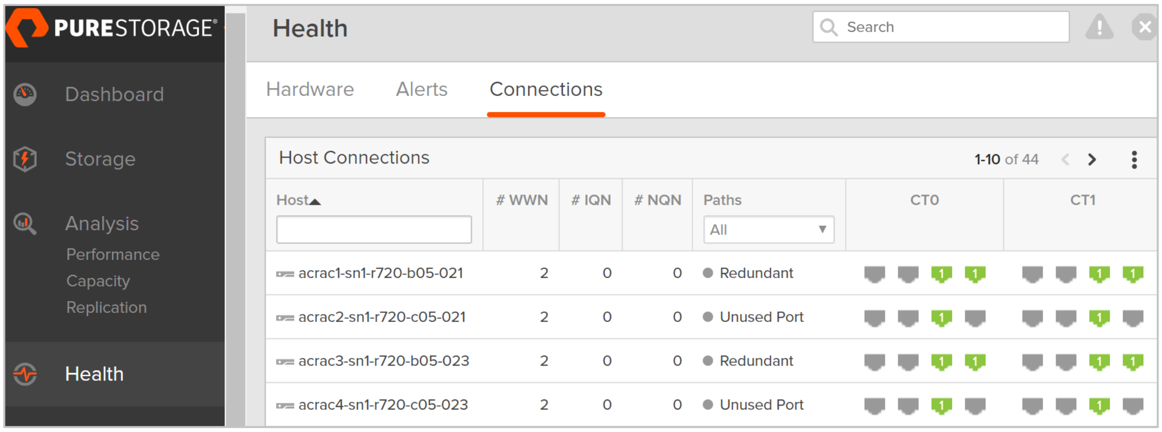


Figure 38: Find the Host under connections to check the connection status.

Host Groups

To simplify provisioning storage to a cluster of hosts, put all the FlashArray host objects into a host group. To create a Host Group, click on the **Storage** section followed by the **Hosts** tab. In the **Host Group** panel, click the plus sign.

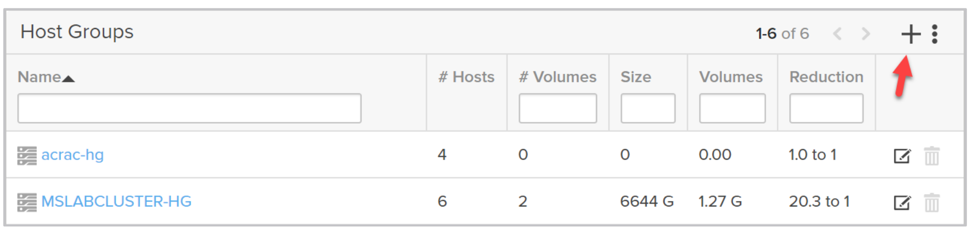


Figure 39: Add a new Host Group.

Enter a name for the host group and click **Create**.

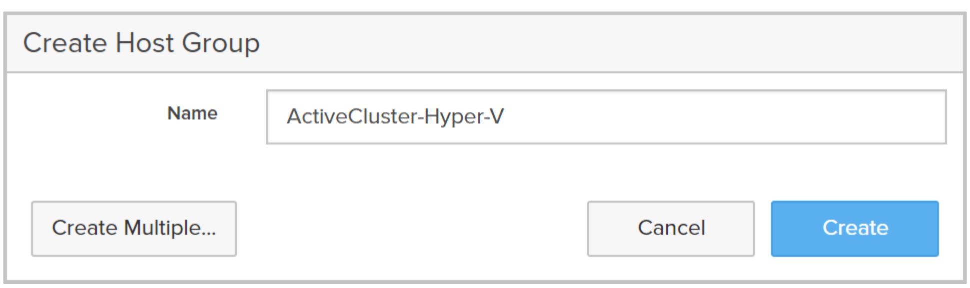


Figure 40: Name the new Host Group.

Click on the host group in the **Host Groups** panel, then click on the **vertical ellipsis** in the **Member Hosts** panel and choose **Add**.

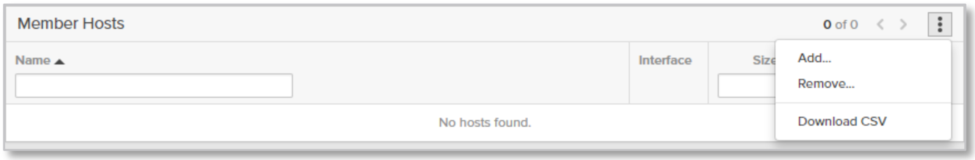


Figure 41: Add a Host, to a Host Group.

Select one or more hosts in the following screen to add to the host group.

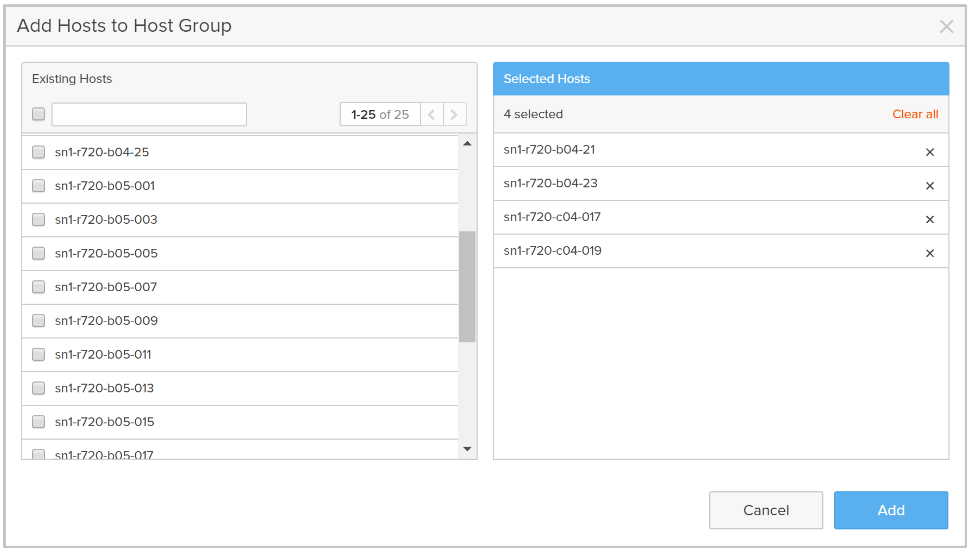


Figure 42: Select the Host or Hosts to add to the Host Group.

If the environment is configured for uniform access, you should configure all hosts in the Hyper-V cluster on both FlashArrays and add them to their host group. If the configuration is non-uniform, you only need to add the hosts that have direct access to the given FlashArray to that FlashArray and its corresponding host group.

Multipathing

Standard Windows multipathing recommendations apply which are described in more detail in the Microsoft Platform guide. At a high level, recommendations include the following:

- Use the MPIO Round Robin path selection policy for FlashArray storage with the I/O Operations Limit set to 1.
- Use multiple HBAs per host for Fibre Channel or multiple NICs per host for iSCSI.
- Connect each host to both controllers.
- In the storage or network fabric, use redundant switches.

Uniform Configuration

In a uniform configuration, all hosts have access to both FlashArrays and can therefore see paths for an ActiveCluster-enabled volume to each FlashArray. To start, create a new volume on the FlashArray. Here's the process using the FlashArray web interface. This environment has a four-node Hyper-V cluster with each host zoned to both FlashArrays.

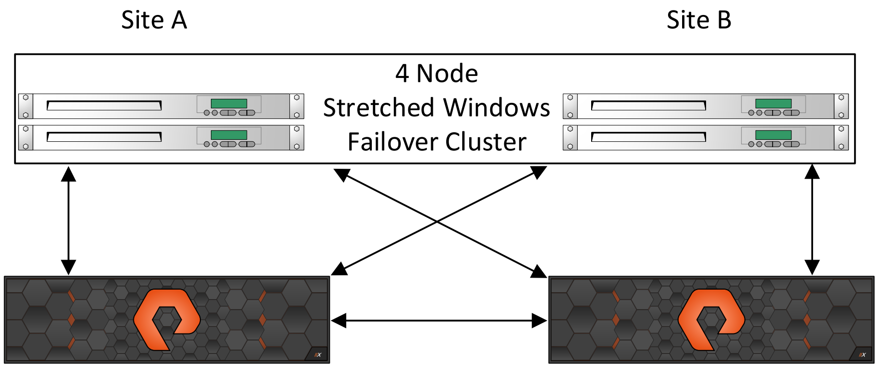
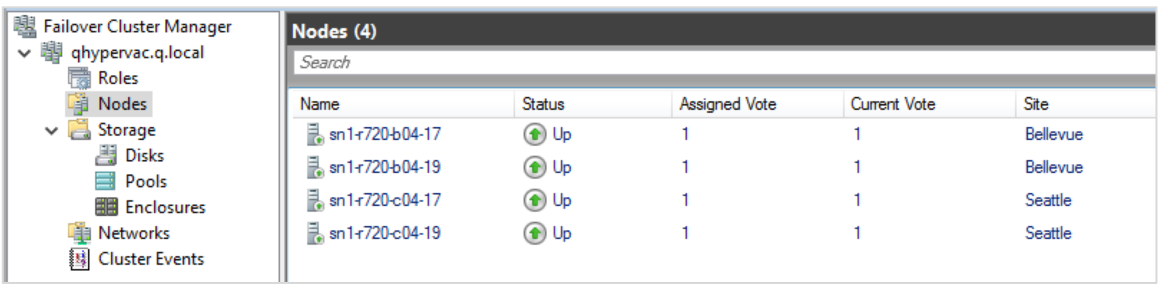


Figure 43: In the Uniform Configuration, all hosts see both FlashArrays.

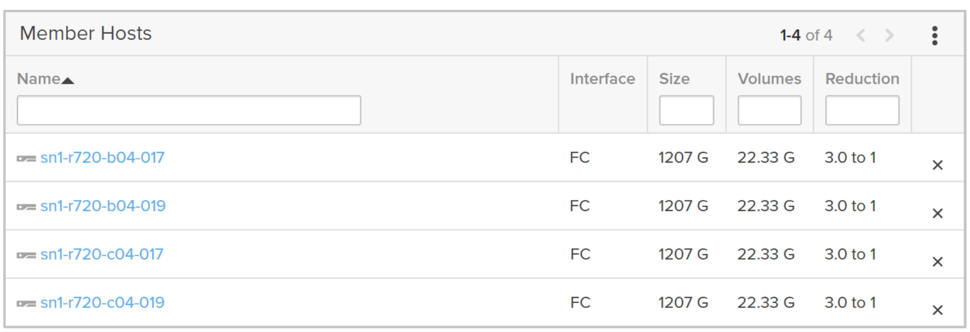
The Hyper-V cluster is shown in Figure 44.



Name	Status	Assigned Vote	Current Vote	Site
sn1-r720-b04-17	Up	1	1	Bellevue
sn1-r720-b04-19	Up	1	1	Bellevue
sn1-r720-c04-17	Up	1	1	Seattle
sn1-r720-c04-19	Up	1	1	Seattle

Figure 44: The Nodes of the Failover Cluster are split between two sites.

The corresponding host group on the first FlashArray:



Name	Interface	Size	Volumes	Reduction
sn1-r720-b04-017	FC	1207 G	22.33 G	3.0 to 1
sn1-r720-b04-019	FC	1207 G	22.33 G	3.0 to 1
sn1-r720-c04-017	FC	1207 G	22.33 G	3.0 to 1
sn1-r720-c04-019	FC	1207 G	22.33 G	3.0 to 1

Figure 45: In a Uniform Configuration, the nodes in both sites, are members of the Host Group.

The first step is to create a volume on my FlashArray in site A and add it to pod “QPID”.

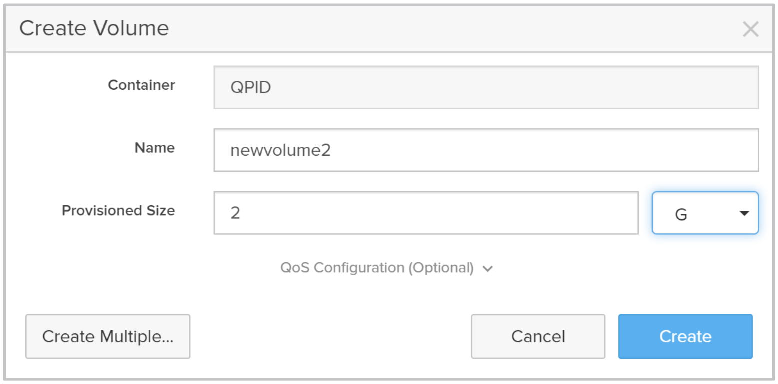


Figure 46: Create a Volume and add it to a Pod in one step.

The pod is not stretched to the FlashArray in site B, so only the site A FlashArray is listed under Arrays.

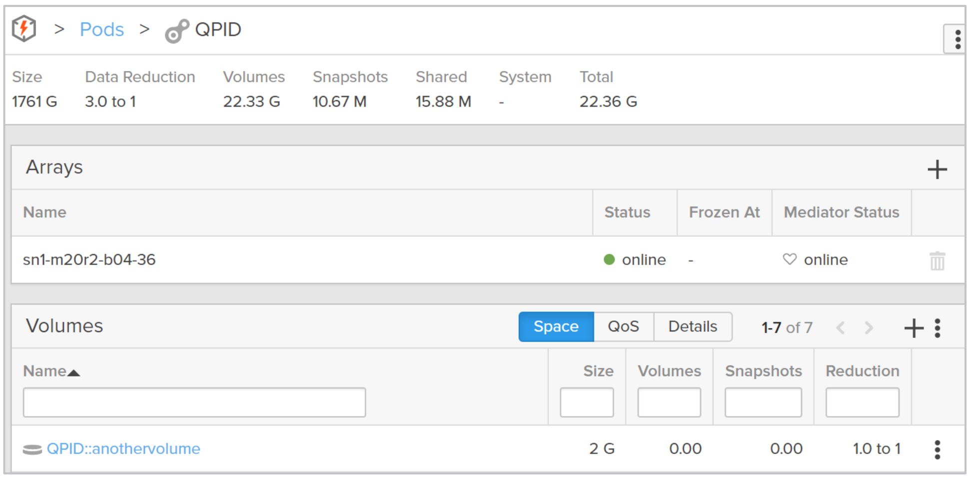


Figure 47: The Pod is not stretched.

The next step is the add it to the host group on the FlashArray in site A. Select the Host Group and then select Connect under Connected Volumes.

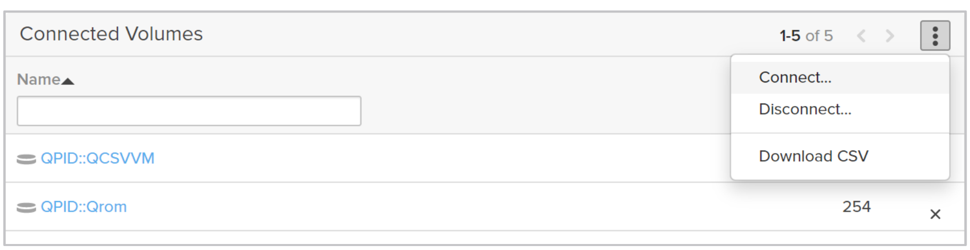


Figure 48: Connect a Volume to a Host Group.

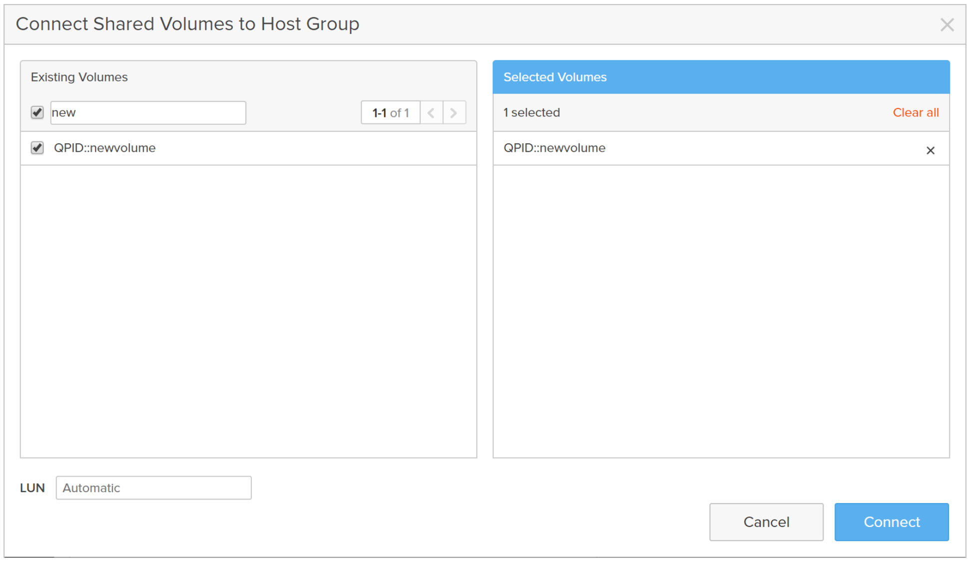


Figure 49: Select the Host Group, then add a Connect Volume.

The volume is now in the host group "QPID" and is in an un-stretched pod on the FlashArray in site A.

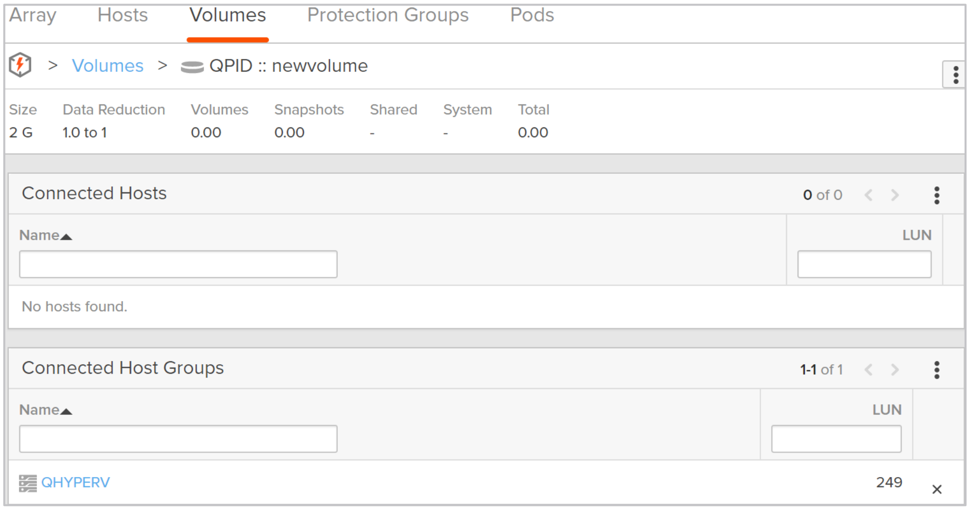


Figure 50: The Volume 'newvolume' is connected to the QHYPERV Host Group.

The next step is to *Rescan Disks* on one node of the Microsoft Failover Cluster. (A disk rescan is not necessary in Purity 5.0.7+.)

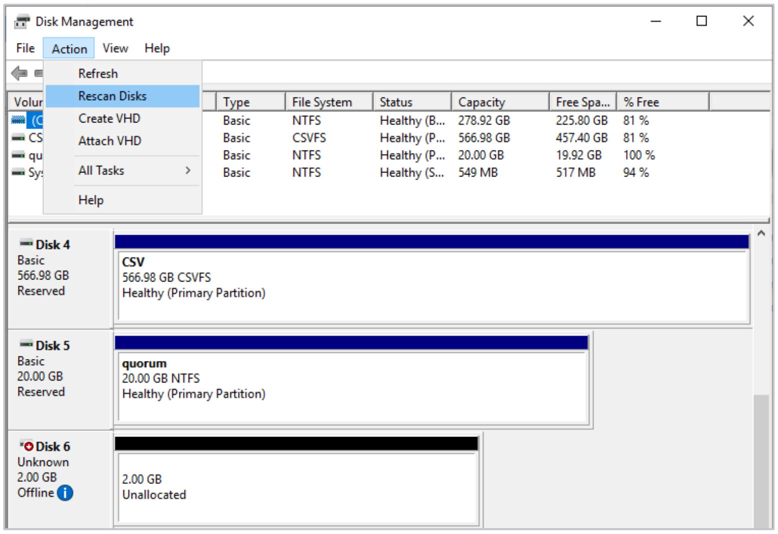


Figure 51: Rescan Disks in Disk Management.

Once the rescan is complete, online and format the disk and add it to the cluster. For many Microsoft database workloads, the recommended allocation unit size (AUS), should be 64KB.

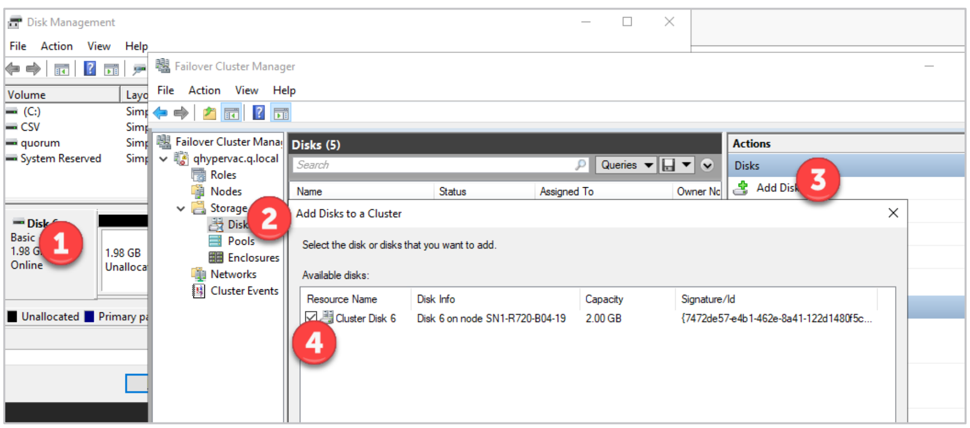


Figure 52: Add a formatted disk to the cluster in Failover Cluster Manager.

Add the online disk to the cluster in Failover Cluster Manager by selecting Disks, then Add Disk, and select the new disk. Next, stretch the pod to the FlashArray in site B.

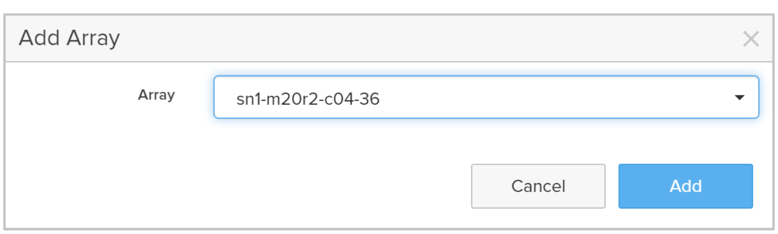


Figure 53: Select the Pod, then add the second Array.

As soon as the FlashArray is added, the pod will start synchronizing. When it is complete, the pod will go fully online and the volume will be available on both FlashArrays.

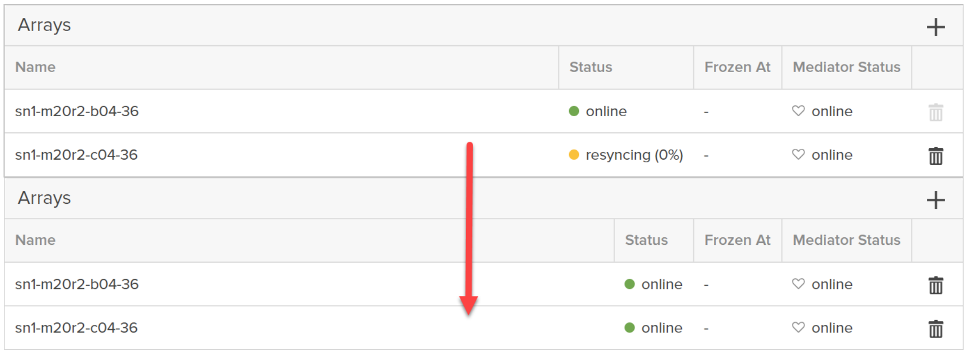


Figure 54: sn1-m20r2-c04-36 has been added and is resyncing until online.

To have the hosts see it on the second FlashArray, you must also add the volume to the proper host or host group on the second FlashArray.

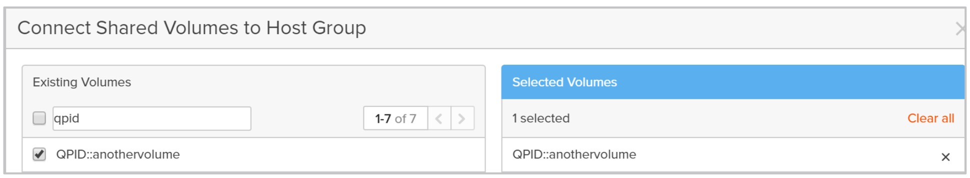


Figure 55: Add the replicated volumes in the Pod to the Host Group on the second FlashArray.

To see the additional paths, rescan disks in Disk Management. (In [Purity 5.0.7+] a disk rescan is not necessary.)

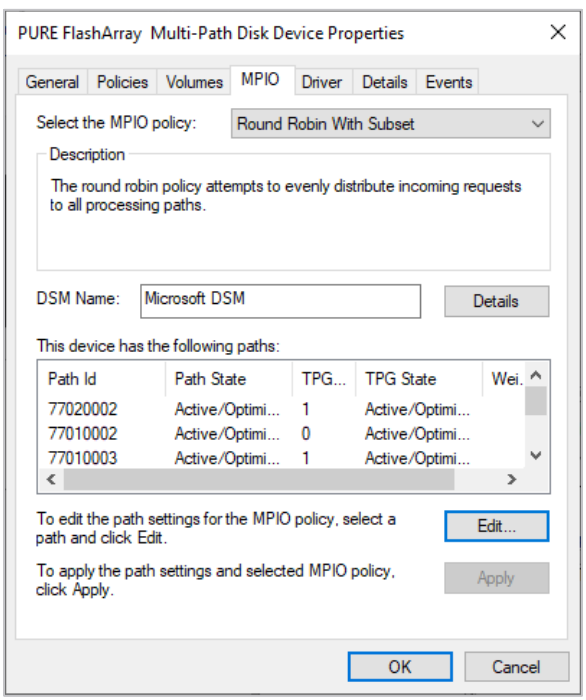


Figure 56: Additional Paths are now available to the Volume on the second FlashArray.

Once the rescan completes, the new paths to the volume via the second FlashArray will appear.

Note: Windows supports up to 32 paths per volume. If you provision more than 32 paths, the additional paths will be dropped, possibly causing uneven access to arrays.

Note in Figure 56 that all paths are active—even those from the secondary FlashArray. This is generally not ideal. In situations where the sites are far apart and both FlashArrays offer an equal number of paths for each device, two performance-impacting things occur:

- Half the writes sent from a host in site A will be sent to the FlashArray in site B. Since writes must be acknowledged in both sites, the data traverses the WAN twice. First the host issues a write across the WAN to the far FlashArray, and then the far FlashArray forwards it back across the WAN to the other 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 only traverses the WAN once.
- Half the reads sent from a host in site A will be sent to the FlashArray in site B. Either side can service reads. And for reads, there is no need for one FlashArray to talk to the other. So, a read doesn't ever have to 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 a host-object setting called Preferred Arrays that intelligently tells Windows which FlashArray should optimally service I/O if a Windows host can see paths to both FlashArrays for a given device.

For a given host, log in to the FlashArray web interface for the local array to that host. Click on the **Storage** section, then the **Hosts** tab, then choose the host you want to configure. In the Details panel, click on the **Add Preferred Arrays** option.

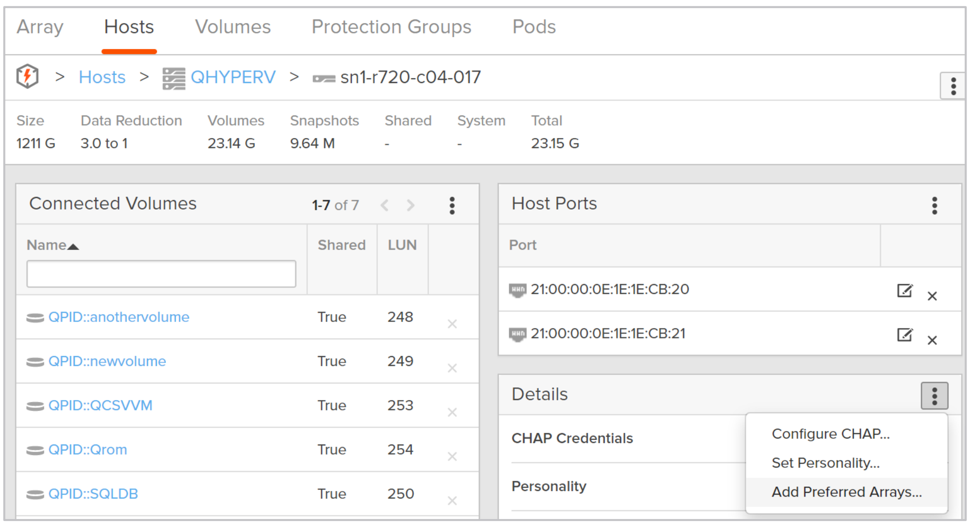


Figure 57: Set the Preferred Arrays in a Uniform Configuration so local reads are preferred.

Choose the FlashArray that you would like to be local for that Hyper-V host and click **Add**.

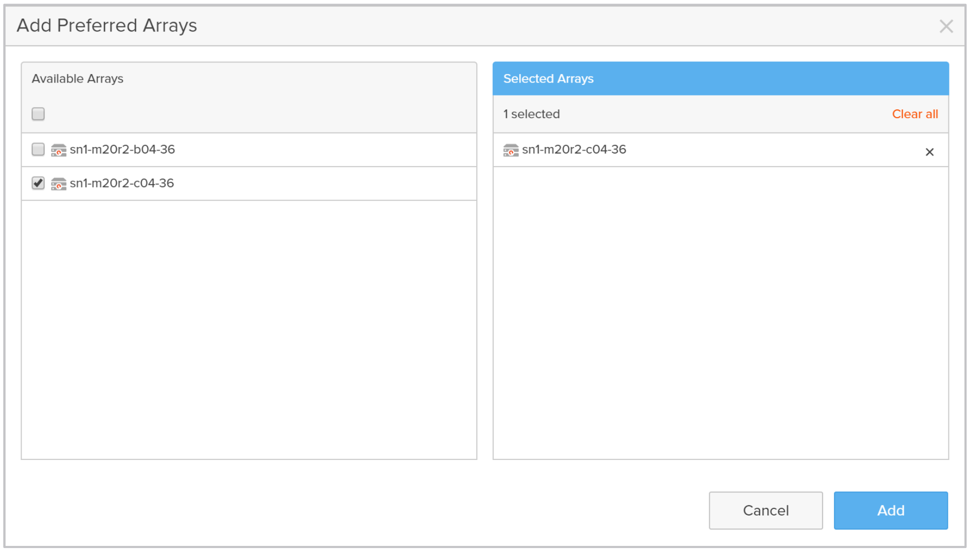


Figure 58: Select the FlashArray that is local to the Host.

You will notice that half of the paths will be marked as Active and the other will be marked as Active/Unoptimized.

```
PS C:\Users\administrator.Q> mpclaim -s -d 2

MPIO Disk2: 04 Paths, Round Robin with Subset, Implicit Only
    Controlling DSM: Microsoft DSM
    SN: 624A93702EBD55E4363D498400063AAD
    Supported Load Balance Policies: FOO RRWS LQD WP LB

    Path ID          State              SCSI Address      Weight
    ---------------------------------------------------------------
    0000000077020003 Active/Optimized   002|000|003|250   0
      TPG_State : Active/Optimized  , TPG_Id: 3, : 50

    0000000077020002 Active/Optimized   002|000|002|250   0
      TPG_State : Active/Optimized  , TPG_Id: 2, : 34

    000000007702000b Active/Unoptimized   002|000|011|253   0
      TPG_State : Active/Unoptimized  , TPG_Id: 1, : 2

    000000007701000a Active/Unoptimized   002|000|010|253   0
      TPG_State : Active/Unoptimized  , TPG_Id: 0, : 18

    0000000077010003 Active/Optimized   001|000|003|250   0
      TPG_State : Active/Optimized  , TPG_Id: 3, : 49

    0000000077010002 Active/Optimized   001|000|002|250   0
      TPG_State : Active/Optimized  , TPG_Id: 2, : 33

    000000007701000b Active/Unoptimized   001|000|011|253   0
      TPG_State : Active/Unoptimized  , TPG_Id: 0, : 1

    000000007702000a Active/Unoptimized   001|000|010|253   0
      TPG_State : Active/Unoptimized  , TPG_Id: 1, : 17
```

Figure 59: Half of the paths are local and Optimized, and half the paths are remote and Unoptimized.

Non-Uniform Configuration

In a non-uniform configuration, hosts have storage access only to the FlashArray local to them. Therefore, in the case of a SAN or storage failure, the hosts local to that array will lose all connectivity to the storage. The example environment uses a four-node Hyper-V cluster: two zoned to FlashArray A and two zoned to FlashArray B.

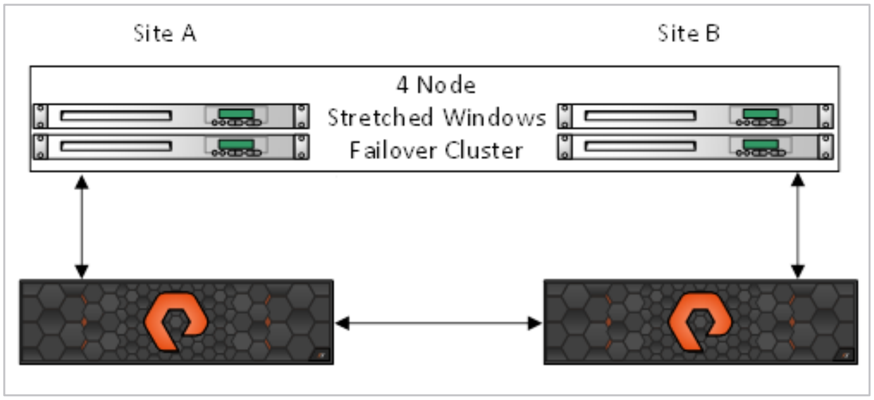


Figure 60: Non-Uniform means that the servers can only see the local FlashArray.

In a non-uniform environment, only the hosts local to a FlashArray have storage connectivity to it. On FlashArray A, the host group only includes two hosts (Figure 60), while the other two are on FlashArray B.

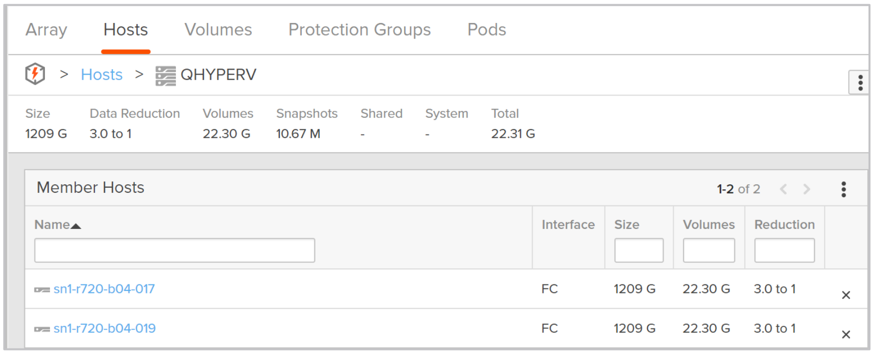


Figure 61: Non-Uniform Host Groups only contain the local Hosts.

The first step is to create a volume on the FlashArray in site A and add it to pod “QPID.”

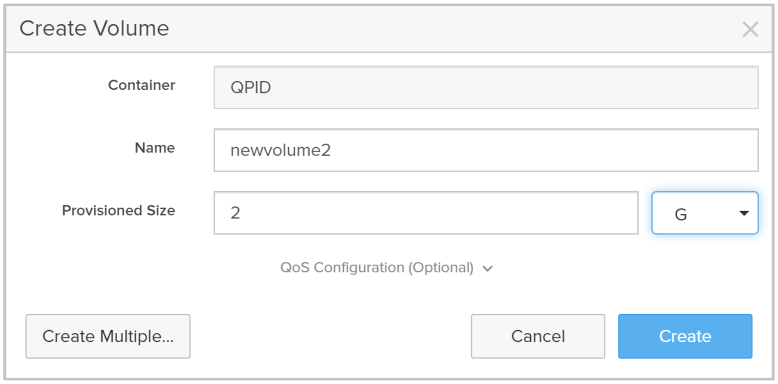


Figure 62: Create a Volume and add it to the Pod in one step.

The pod is not yet stretched to the FlashArray in site B.

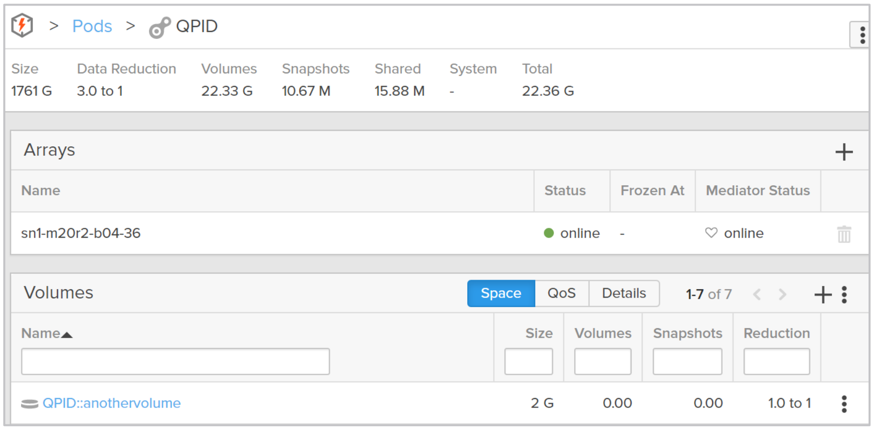


Figure 63: The Volumes in the unstretched Pod are only on one Array.

The next step is to add it to the host group on the FlashArray in site A.

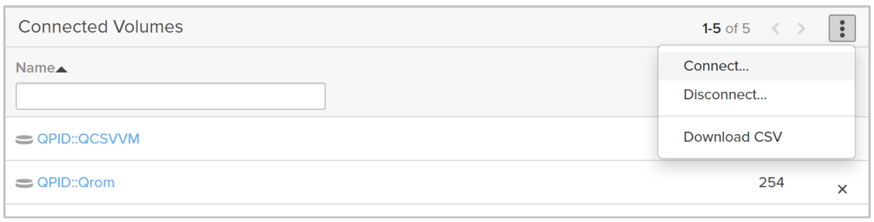


Figure 64: Select the Host Group and then Connect the new Volume.

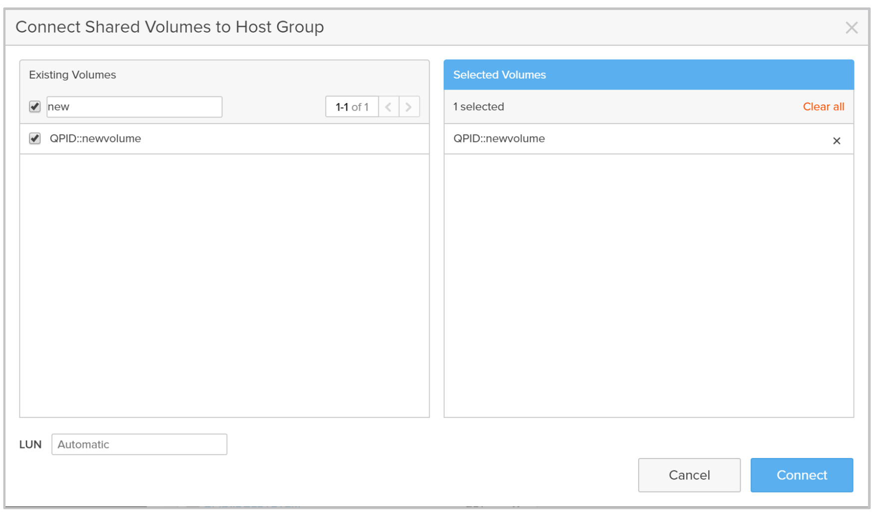


Figure 65: Select the new Volume and then Connect.

As shown in Figure 66, the volume is now in host group "QPID" within an un-stretched pod on the FlashArray in site A.

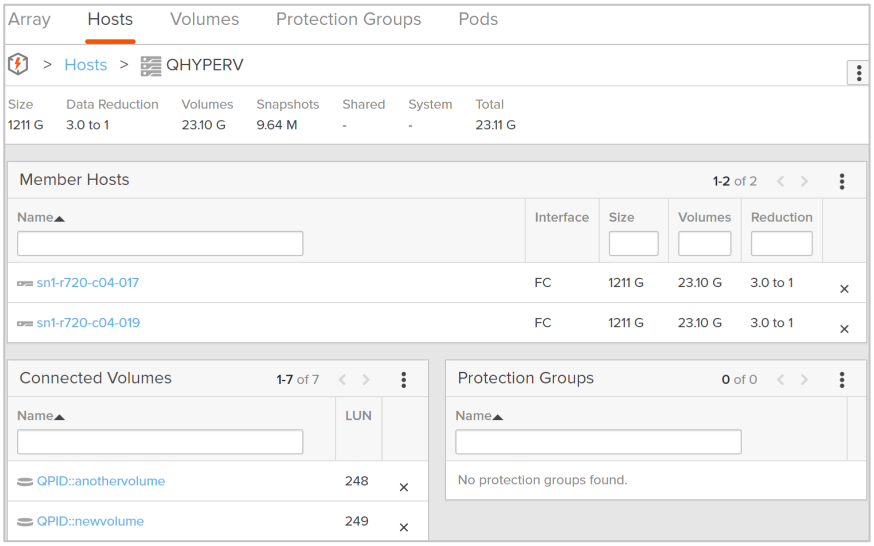


Figure 66: The Host Group contains the Volume 'newvolume' in the unstretched Pod.

The next step is to rescan disks in disk management on one node of the cluster.

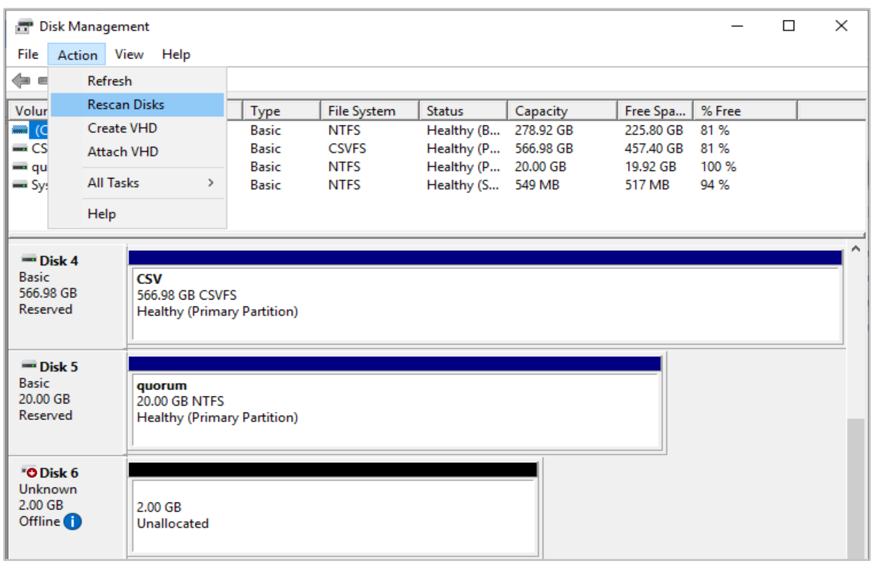


Figure 67: Rescan Disks in Disk Management.

Once the rescan is complete, bring online and format the disk and add it to the cluster.

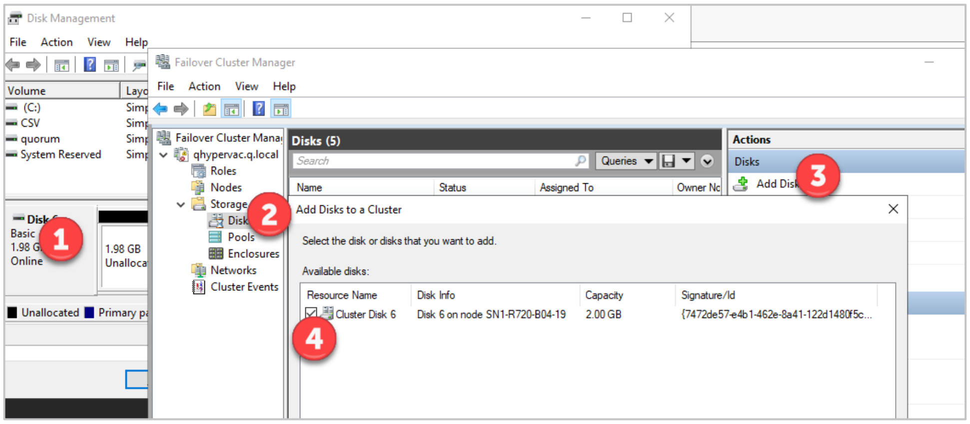


Figure 68: Add a formatted disk to the cluster in Failover Cluster Manager.

The next step is to stretch the pod to the FlashArray in site B.

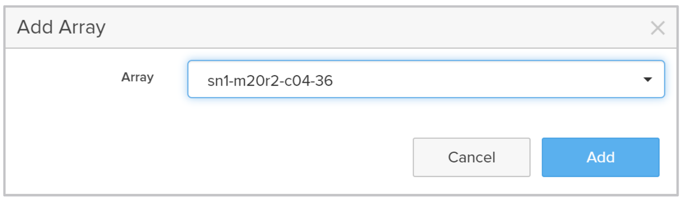


Figure 69: Select the Pod, then add the second Array.

The pod will start synchronizing as soon as you've added the FlashArray. When it is complete, the pod will go fully online and the volume will be available on both FlashArrays.

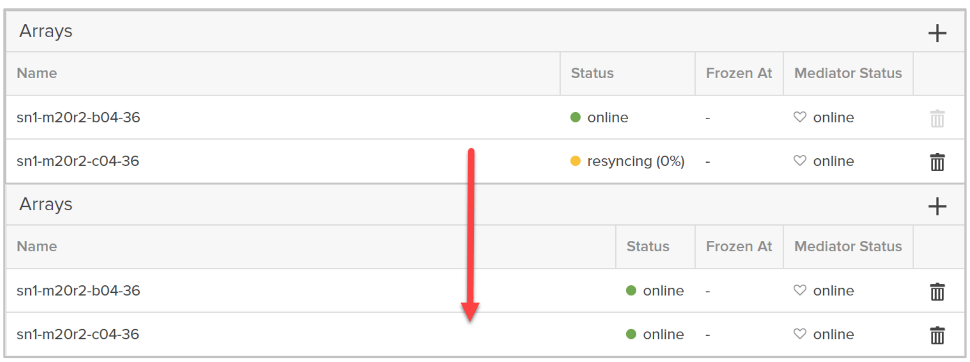


Figure 70: sn1-m20r2-c04-36 has been added and is resyncing until online.

For the hosts in site B to see the volume on the second FlashArray, you must add it to the proper host or host group on the second FlashArray.

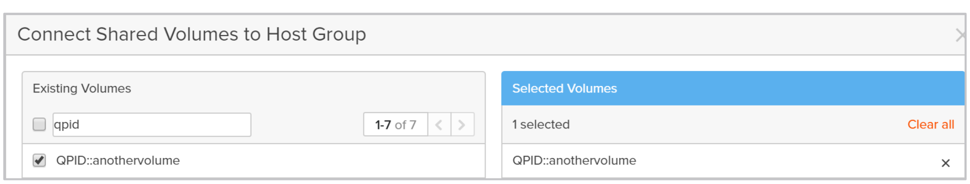


Figure 71: Add the replicated volumes in the Pod to the Host Group on the second FlashArray.

Configuring VM High Availability

High-availability Wizard

VMs that aren't configured as highly available will not automatically failover. You can configure this after VM creation in Failover Cluster Manager, System Center Virtual Machine Manager (SCVMM), or Windows PowerShell. In Failover Cluster Manager, select Roles, and then Configure Role (Figure 72).

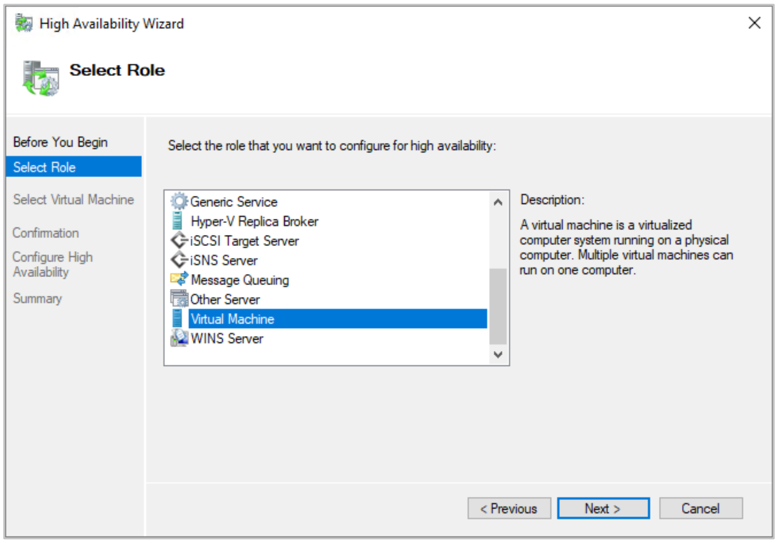


Figure 72: Configure a VM as Highly Available, in Failover Cluster Manager.

In PowerShell use the Add-ClusterVirtualMachineRole cmdlet:

```
Add-ClusterVirtualMachineRole -VMName <YourVM>
```

Create a Highly Available VM

You cannot use Hyper-V Manager to create a new highly available VM. Instead use SCVMM, Failover Cluster Manager, or Windows PowerShell. In **Failover Cluster Manager,** select Roles, then Virtual Machines, then New Virtual Machine.

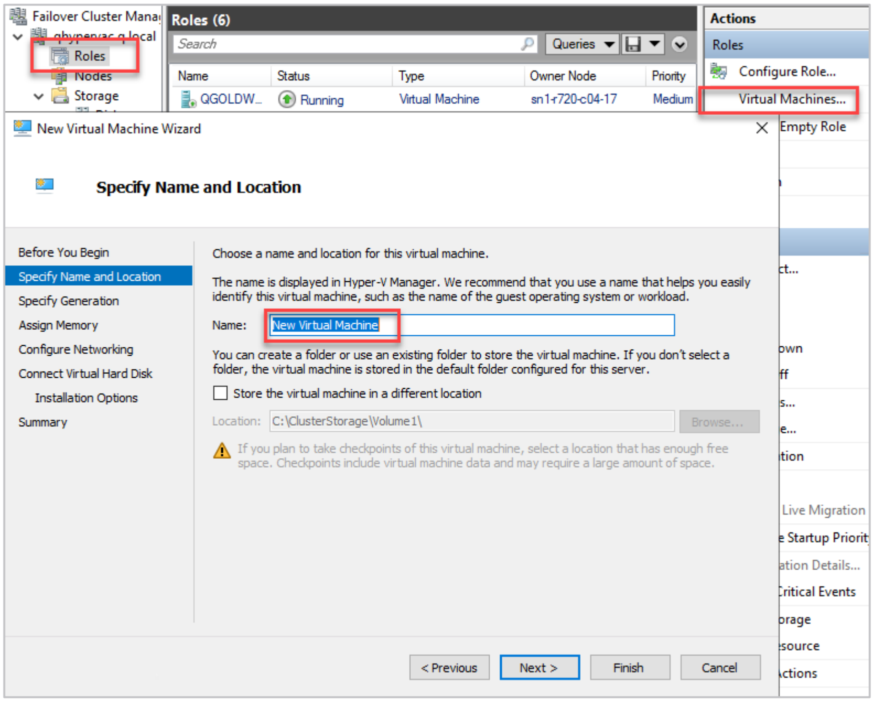


Figure 73: Create a new highly available VM in Failover Cluster Manager.

In PowerShell, pipe the New-VM cmdlet to the Add-ClusterVirtualMachineRole cmdlet.

```
New-VM testps -MemoryStartupBytes 512MB | Add-ClusterVirtualMachineRole
```

VM Preferred Owner

When the automatic balancer is not enabled, VM distribution can become unbalanced in certain failure scenarios. Configuring the preferred owners for a VM can help with that balance. Note that you can fail over a VM to a node that is not preferred, if no preferred nodes are available. Select **Properties** on a Virtual Machine, then select the preferred owners.

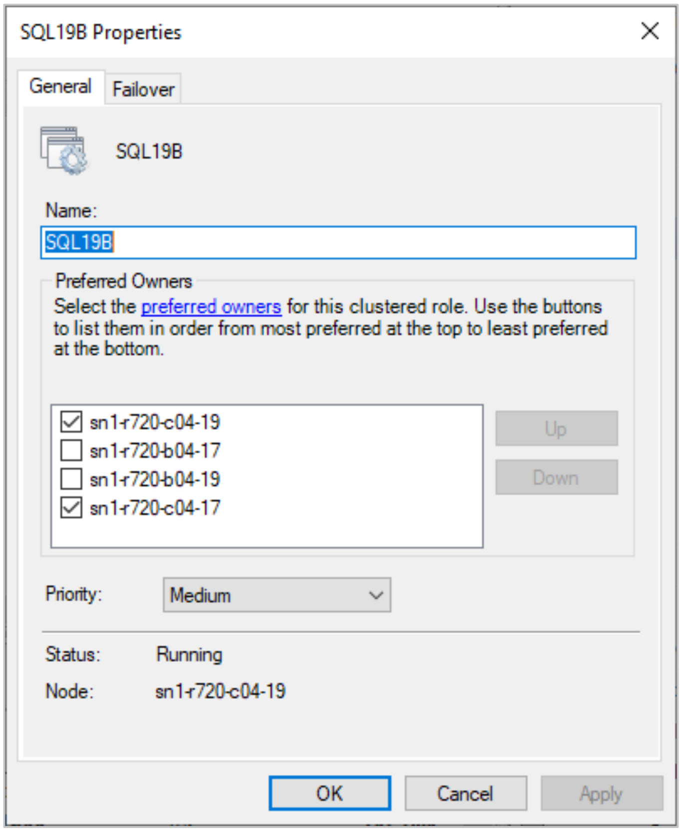


Figure 74: Select Preferred Owners. In this case only the nodes in one site are selected.

VM Possible Owner

In many cases it's impossible to stretch the network subnet to both sites. When using applications with their own high availability, configure the VM's possible owners to keep VMs that should be in a site configured only for the nodes in that site. This will prevent the VM from failing over to a node in another site if none of the nodes in the second site is included in the possible owners.

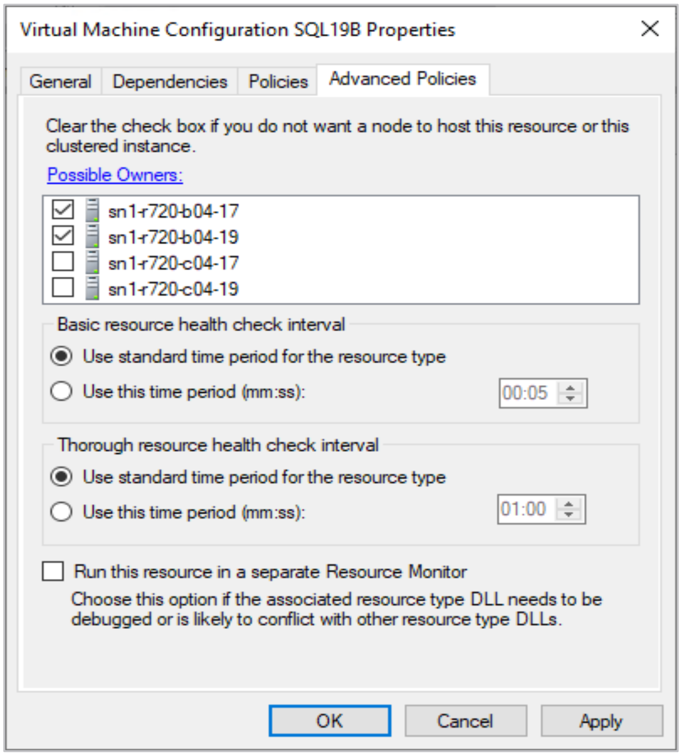


Figure 75: Possible Owners will prevent a VM from being failed over to unselected nodes.

Networking

Prerequisites

Windows Failover Clusters require access to services including Active Directory, DNS, and NTP. Regardless of the implemented networking architecture, nodes, VMs, and applications such as Microsoft Exchange and Microsoft SQL Server require access to these services regardless of the data center in which they live.

Note: Some application configurations that provide replication at the application layer are typically not good candidates for storage-layer replication. If you have an Exchange Server Database Availability Group or a SQL Server Availability Group with two or more copies of your databases, placing the copies in pods so that ActiveCluster will replicate them to another site is not optimal as the round-trip latency will impact write latencies. Most administrators stretch the application and follow the guidelines to ensure configuration of synchronous or asynchronous replication at the application layer. In corner-case networking situations, it may make sense to have one copy of a database availability group in a pod for asynchronous replication to another FlashArray because the networking parameters won't allow the application-replication mechanism to function (although this is rare).

Stretched Layer 2 Network

A stretched VLAN is the easiest configuration from a VM or application perspective. The network is the same in both data centers and no special configuration is required for the VM or failover clusters. With this configuration a VM can move between sites without having to change the network configuration. Stretching a VLAN requires supported networking equipment in both sites and configuration expertise. A caveat: A catastrophic failure can take down the network in both sites without configuration and precautions.

- Separate networks: In the case of separate networks in each data center, normal VM failover behavior, such as a live migration, will succeed. However, the VM or application will be inaccessible until its networking is changed to the current network.
- Manual configuration: The manual approach could be as simple as configuring multiple NICs in the VM and using automation to enable and disable the appropriate NICs depending on in which site the VM is currently running. In the case of Cluster Network Name resources, Microsoft supports the OR tag with the inclusion of multiple IP address resources.

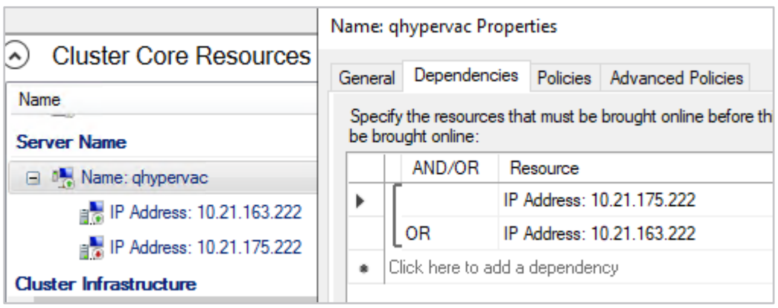


Figure 76: IP Address dependencies using the OR tag.

You can employ this same strategy on other Microsoft cluster roles such as a SQL Server Failover Cluster Instance. In Figure 76, the SQL Server Instance's network name also has an OR dependency with multiple IP addresses.

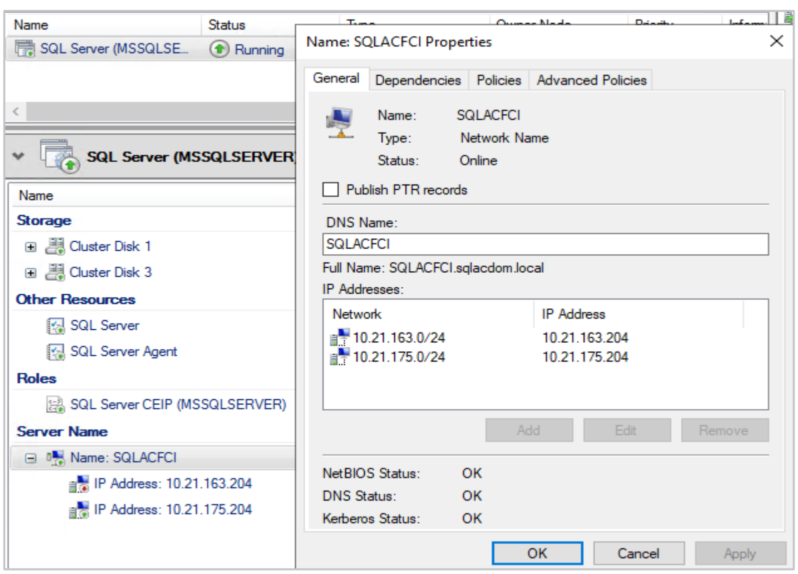


Figure 77: SQL Server FCI Role IP Addresses.

Keep in mind that a SQL Failover Cluster Instance running on VMs requires that you not only add cluster disks to the SQL Server Resource, but also add them as an "AND" dependency on the SQL Server resource itself. Otherwise, the databases will not be allowed on that new disk.

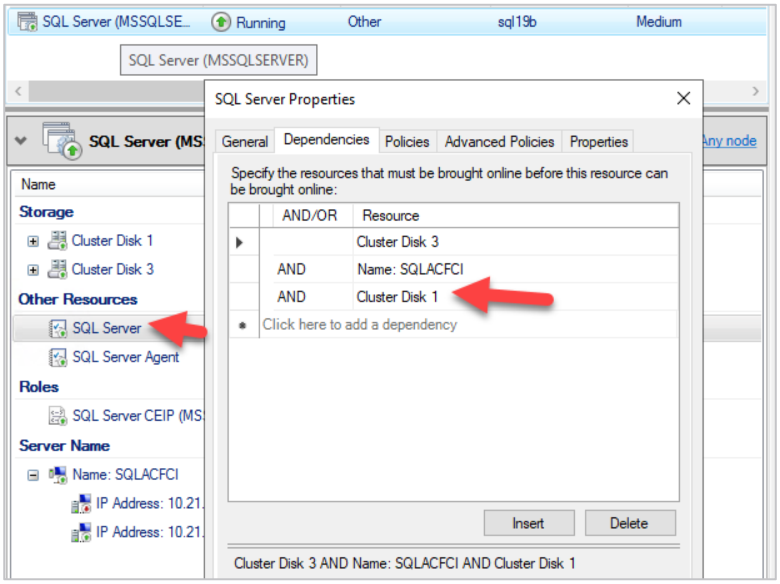


Figure 78: SQL Server Resource also requires Disk Dependencies.

Applications such as SQL Server and Exchange perform best with application-layer availability. The VMs themselves should prefer Hyper-V nodes in one site. The Exchange Database Availability Group (DAG) will activate a passive copy of the database in the surviving site during a failure, while the SQL Server instance is a clustered role that will be moved to a surviving Hyper-V node. The VMs do not need to failover in most cases. Otherwise they require additional work to ensure that they can have their networking configuration updated in the case of failover. To prevent VMs from failover across site boundaries see the VM Possible Owner section.

Software Defined Networking (SDN) Configuration

Windows Server 2012 R2 requires System Center VM Manager (SCVMM) to enable and configure SDN. Microsoft included SDN in the Windows Server 2016 operating system, making SCVMM optional for SDN. SDN requires running infrastructure VMs. The Hyper-V hosts that run those VMs require the data center edition of Windows Server 2016 or newer. Hyper-V Network Virtualization (HNV) abstracts VMs from the physical network and is compatible with VLANs. This SDN can virtualize the network inside the VMs in the case where it isn't feasible to stretch the network. Live Migration will work without issue and the network inside the VM does not require editing. However, it is a bit complex to implement and until recently required SCVMM.

FlashArray Storage Access

Table 1 describes different failure scenarios for ActiveCluster environments and how storage availability is affected.

Solution Component Failure

Array A	Array B	Replication Link	Mediator	Access to Storage
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

Table 1: Storage availability under failure scenarios

* 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 occurred, access to the mediator is no longer required and access to storage remains available on one array. Table 2 explains detailed failure scenarios for ActiveCluster and the surrounding network.

Host and Storage Failures

Failure Scenario	Failure Behavior
Single or multiple host failure	Applications can automatically fail over 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 (Failure of SAN interconnect between sites) **Single or multiple host failure**	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 so not have a SAN interconnect between sites, so the 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: • The site without the SAN outage will experience some storage path failures for paths to the remote array and continue IO on paths to the local array. • The site with the SAN outage will experience total loss of access to volumes and applications must fail over to the other site as mentioned above. Non-uniform connected hosts: • The site without the SAN outage will maintain access to local volumes. • The site with the SAN outage will experience total loss of access to volumes and applications must fail over to the other site as mentioned above.

Table 2: Detailed failure scenarios for ActiveCluster and the surrounding network.

Conclusion

The combination of active-active replication on the FlashArray and Microsoft Failover Clustering of Hyper-V hosts provides a site-resilient ActiveCluster. The simplicity and flexibility of FlashArray itself plus the ActiveCluster feature allows administrators to easily configure and offer a highly available storage solution.

Additional Resources

- Microsoft Platform Guide
- VMware Platform Guide
- FlashStack Converged Infrastructure Solutions
- Best Practices Guide for Hyper-V 2016
- Guide for Microsoft Applications with Hyper-V on FlashStack Mini CI
- ActiveCluster Solution Overview
- ActiveCluster Requirements and Best Practices
- SQL Server with ActiveCluster
- Implementing vSphere Metro Storage Cluster with ActiveCluster

Pure Storage, Inc.
650 Castro Street, #400
Mountain View, CA 94041

PS1055-01 02/2020