



# VMware vSphere Metro Storage Cluster (vMSC)

VMware Storage

## Table of contents

VMware vSphere Metro Storage Cluster (vMSC)	3
Purpose and Overview	3
Target Audience	3
Introduction	4
Technical Requirements and Constraints	4
Uniform Versus Non-uniform vMSC Configurations	5
Infrastructure Architecture	7
Infrastructure	7
vSphere Configuration	9
vSphere HA	9
Admission Control	9
vSphere HA heartbeats	11
Permanent Device Loss and All Paths Down Scenarios	12
Restart ordering and priority	14
ProActive HA	19
vSphere DRS	21
Site Affinity	21
Advanced Settings	25
Correcting Affinity Rule Violation	26
vSphere Storage DRS	27
Migrations	28
Failure Scenarios	29
Single-Host Failure in Frimley Data Center	29
Single-Host Isolation in Frimley Data Center	31
Storage Partition	31
Data Center Partition	33
Disk Shelf Failure in Frimley Data Center	35
Full Storage Failure in Frimley Data Center	36
Permanent Device Loss	38
Full Compute Failure in Frimley Data Center	39
Loss of Frimley Data Center	40
Summary and Acknowledgement	42

## VMware vSphere Metro Storage Cluster (vMSC)

### Purpose and Overview

VMware vSphere® Metro Storage Cluster (vMSC) is a specific storage configuration that is commonly referred to as stretched storage clusters or metro storage clusters. These configurations are usually implemented in environments where disaster and downtime avoidance is a key requirement. This recommended practice document was developed to provide additional insight and information for the operation of a vMSC infrastructure in conjunction with VMware vSphere. This paper explains how vSphere handles specific failure scenarios, and it discusses various design considerations and operational procedures. For detailed information about storage implementations, refer to the documentation provided by the appropriate VMware storage partner.

Note that initially, vMSC storage configurations had to go through a mandatory certification program. As of vSphere 6.0 this is no longer needed. vMSC configurations are now fully partner supported and can be found on the vmware.com website under PVSP (Partner Verified and Supported Products). Before purchasing, designing or implementing please consult the PVSP listing to ensure the partner has filed for PVSP and has tested with the correct vSphere versions.

( <https://www.vmware.com/resources/compatibility/vcl/partnersupport.php> )The vMSC listings typically also provide a link to the specifics of the implementation by the partner. As an example, the PVSP Listing for EMC VPLEX provides the following link: <https://kb.vmware.com/kb/2007545> . This link provides all tested scenarios and supported components with EMC VPLEX.

What does this mean for support? Let's take a look at two scenarios:

You have implemented Dell/EMC VPLEX, which is listed under the PVSP program, and you phone up support that you have an issue in your environment. What will happen when you phone up support?

- If the issue appears to not be related to the storage component then VMware Global Support Services will process the support request and provide the help required to solve the problem.
- If the issue appears to be related to the storage component then VMware Global Support Services will ask you (the customer) to reproduce the issue without the PVSP component
  - If the problem can be reproduced without the PVSP component then VMware GSS will process the request and provide the help required to solve the problem
  - If the problem cannot be reproduced without the PVSP component then VMware GSS will ask you (the customer) to contact the partner for support

You have implemented a stretched storage solution which is not listed under the PVSP program and experience a problem, what will happen when you phone up support?

- The used solution is not on the VMware HCL or under the PVSP program, as such VMware GSS can refuse to process the request and will ask you to contact the partner for support.

Hopefully, this makes it clear that although the HCL is no longer available, a listing of the vMSC storage solution under the PVSP program is critical for the supportability of your environment.

### Target Audience

This document is intended for individuals with a technical background who design, deploy, or manage a vSphere Metro Storage Cluster infrastructure. This includes but is not limited to technical consultants, infrastructure architects, IT managers, implementation engineers, partner engineers, sales engineers, and customer staff. This solution brief is not intended to replace or override existing certified designs for vSphere Metro Storage Cluster solutions; it instead is meant to supplement the knowledge and provide additional information.

The authors of this document assume that the reader is familiar with vSphere, VMware vCenter Server™, VMware vSphere High Availability (vSphere HA), VMware vSphere Distributed Resource Scheduler™ (vSphere DRS), VMware vSphere Storage DRS™, and replication and storage clustering technology and terminology.

## Introduction

A VMware vSphere Metro Storage Cluster configuration is a specific storage configuration that combines replication with array-based clustering. These solutions are typically deployed in environments where the distance between data centers is limited, often metropolitan or campus environments.

vMSC infrastructures are implemented with a goal of reaping the same benefits that vSphere HA clusters provide to a local site, in a geographically dispersed model with two data centers in different locations. A vMSC infrastructure is essentially a stretched cluster. The architecture is built on the premise of extending what is defined as “local” in terms of network, storage, and compute to enable these subsystems to span geographies, presenting a single and common base infrastructure set of resources to the vSphere cluster at both sites. It in essence stretches storage, network, and compute between sites.

The primary benefit of a stretched cluster model is that it enables fully active and workload-balanced data centers to be used to their full potential and it allows for extremely fast recovery in the event of a host or even full site failure. The capability of a stretched cluster to provide this active balancing of resources should always be the primary design and implementation goal. Although often associated with disaster recovery, vMSC infrastructures are not recommended as primary solutions for pure disaster recovery.

This document does not explain the difference between disaster recovery and a downtime- or disaster-avoidance solution. For more details on this distinction, refer to *Stretched Clusters and VMware vCenter Site Recovery Manager: Understanding the Options and Goals*, located here:

<https://www.vmware.com/techpapers/2012/stretched-clusters-and-vmware-vcenterm-site-recov-10262.html>

Stretched cluster solutions offer the following benefits:

- Workload mobility
- Cross-site automated load balancing
- Enhanced downtime avoidance
- Disaster avoidance
- Fast recovery

## Technical Requirements and Constraints

Due to the technical constraints of an online migration of VMs, the following specific requirements must be met prior to consideration of a stretched cluster implementation:

- Storage connectivity using Fibre Channel, iSCSI, NFS, and FCoE is supported.
- The maximum supported network latency between sites for the vSphere ESXi™ management networks is 10ms round-trip time (RTT).
- vSphere vMotion, and vSphere Storage vMotion, supports a maximum of 150ms latency as of vSphere 6.0, but this is not intended for stretched clustering usage. (Requires Enterprise Plus license)
- The maximum supported latency for synchronous storage replication links is 10ms RTT. Refer to documentation from the storage vendor because the maximum tolerated latency is lower in most cases. The most commonly supported maximum RTT for storage systems is 5ms.
- The vSphere vMotion network has a 250 Mbps of dedicated bandwidth per concurrent vMotion session requirement.
- Only legacy FT is supported, SMP FT is not supported on vMSC.
  - Note that when a DRS VM/Host rule is created for a VM both the primary as well as the secondary FT VM will respect the rule!
- Storage IO Control is not supported on a vMSC enabled datastore
  - Note that SDRS IO Metric enables Storage IO Control, as such this feature needs to be deactivated

The question we typically get is if there is a minimum license edition of vSphere required to create a vSphere Metro Storage Cluster. The answer to that question is no. You can create a stretched cluster with any edition, however, if you have a requirement for automated workload balancing from either a CPU or storage perspective then the minimum required license level is vSphere Enterprise Plus, as this license includes vSphere DRS and Storage DRS.

The storage requirements are slightly more complex. A vSphere Metro Storage Cluster requires what is in effect a single storage

subsystem that spans both sites. In this design, a given datastore must be accessible—that is, be able to be read and be written to—simultaneously from both sites. Further, when problems occur, the vSphere hosts must be able to continue to access datastores from either location transparently and with no impact on ongoing storage operations.

This precludes traditional synchronous replication solutions because they create a primary-secondary relationship between the active (primary) LUN where data is being accessed and the secondary LUN that is receiving replication. To access the secondary LUN, replication is stopped, or reversed, and the LUN is made visible to hosts. This “promoted” secondary LUN has a completely different LUN ID and is essentially a newly available copy of a former primary LUN. This type of solution works for traditional disaster recovery-type configurations because it is expected that VMs must be started up on the secondary site. The vMSC configuration requires simultaneous, uninterrupted access to enable live migration of running VMs between sites.

The storage subsystem for a vMSC must be able to be read from and write to both locations simultaneously. All disk writes are committed synchronously at both locations to ensure that data is always consistent regardless of the location from which it is being read. This storage architecture requires significant bandwidth and very low latency between the sites in the cluster. Increased distances or latencies cause delays in writing to disk and a dramatic decline in performance. They also preclude successful vMotion migration between cluster nodes that reside in different locations.

### Uniform Versus Non-uniform vMSC Configurations

vMSC solutions are classified into two distinct types. These categories are based on a fundamental difference in how hosts access storage. It is important to understand the different types of stretched storage solutions because this influences design considerations. The two types are:

- Uniform host access configuration - vSphere hosts from both sites are all connected to a storage node in the storage cluster across all sites. Paths presented to vSphere hosts are stretched across a distance.
- Non-uniform host access configuration - vSphere hosts at each site are connected only to the storage node(s) at the same site. Paths presented to vSphere hosts from storage nodes are limited to the local site.

The following in-depth descriptions of both types clearly define them from architectural and implementation perspectives.

With uniform host access configuration, hosts in data center A and data center B have access to the storage systems in both data centers. In effect, the storage area network is stretched between the sites, and all hosts can access all LUNs. NetApp MetroCluster is an example of uniform storage. In this configuration, read/write access to a LUN takes place on one of the two arrays, and a synchronous mirror is maintained in a hidden, read-only state on the second array. For example, if a LUN containing a datastore is read/write on the array in data center A, all vSphere hosts access that datastore via the array in data center A. For vSphere hosts in data center A, this is local access. vSphere hosts in data center B that are running VMs hosted on this datastore send read/write traffic across the network between data centers. In case of an outage or an operator-controlled shift of control of the LUN to data center B, all vSphere hosts continue to detect the identical LUN being presented, but it is now being accessed via the array in data center B.

The ideal situation is one in which VMs access a datastore that is controlled (read/write) by the array in the same data center. This minimizes traffic between data centers to avoid the performance impact of reads' traversing the interconnect.

The notion of “site affinity” for a VM is dictated by the read/write copy of the datastore. “Site affinity” is also sometimes referred to as “site bias” or “LUN locality.” This means that when a VM has site affinity with data center A, its read/write copy of the datastore is located in data center A. This is explained in more detail in the “vSphere DRS” subsection of this paper.

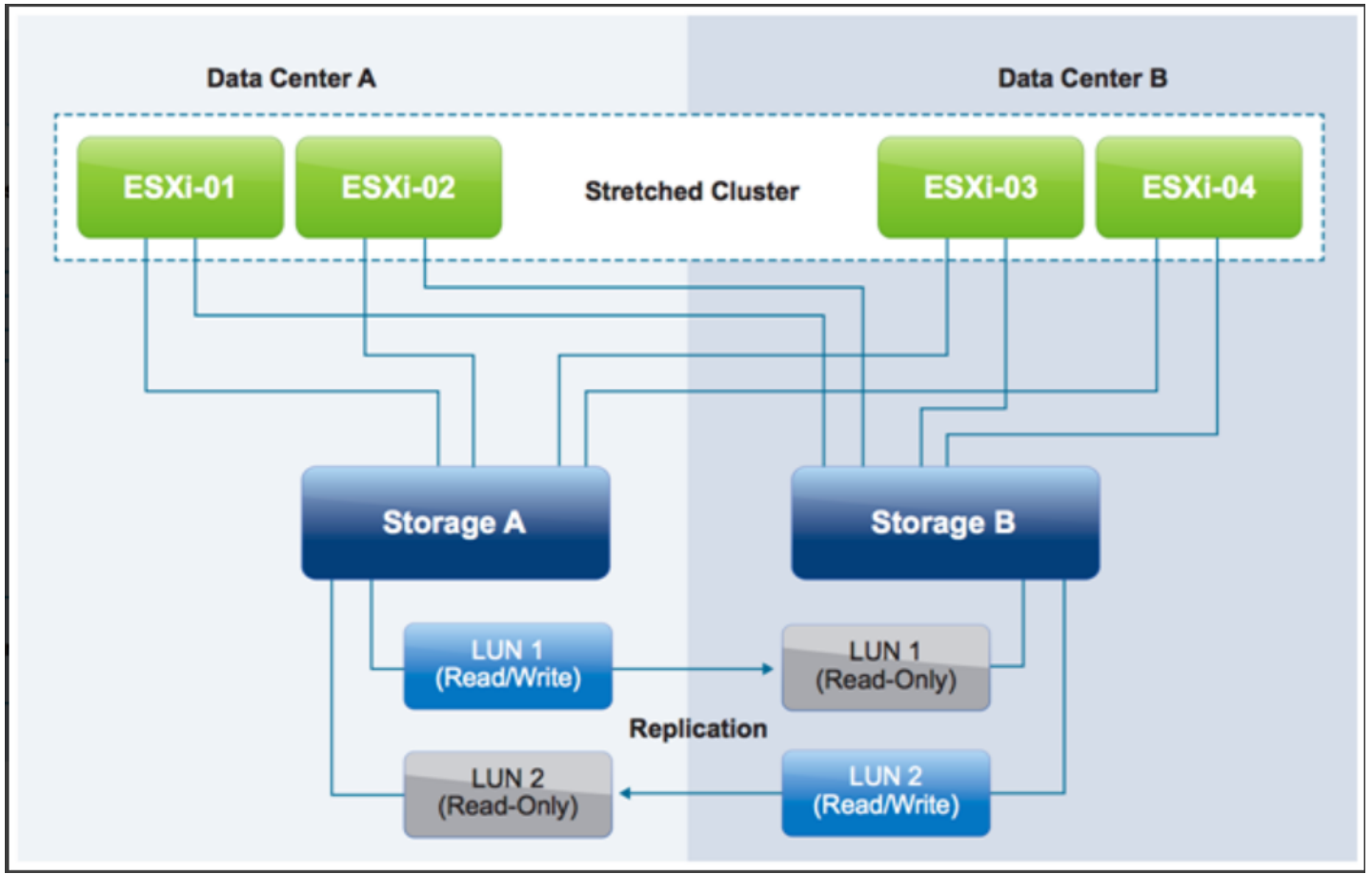


Figure 1 - Uniform Configuration

With **non-uniform** host access configuration, hosts in data center A have access only to the array within the local data center. The array, as well as its peer array in the opposite data center, is responsible for providing access to datastores in one data center. EMC® VPLEX® is an example of a storage system that can be deployed as a non-uniform storage cluster, although it can also be configured in a uniform manner. VPLEX provides the concept of a “virtual LUN,” which enables vSphere hosts in each data center to read and write to the same datastore or LUN. VPLEX technology maintains the cache state on each array so vSphere hosts in either data center detect the LUN as local. EMC calls this solution “write anywhere.” Even when two VMs reside on the same datastore but are located in different data centers, they write locally without any performance impact on either VM. A key point with this configuration is that each LUN or datastore has “site affinity,” also sometimes referred to as “site bias” or “LUN locality.” In other words, if anything happens to the link between the sites, the storage system on the preferred site for a given datastore will be the only one remaining with read/write access to it. This prevents any data corruption in case of a failure scenario. If VMs by any chance are not running within the location that has read/write access to the datastore they will end up being halted and restarted by vSphere HA. We will discuss this in-depth in the upcoming sections.

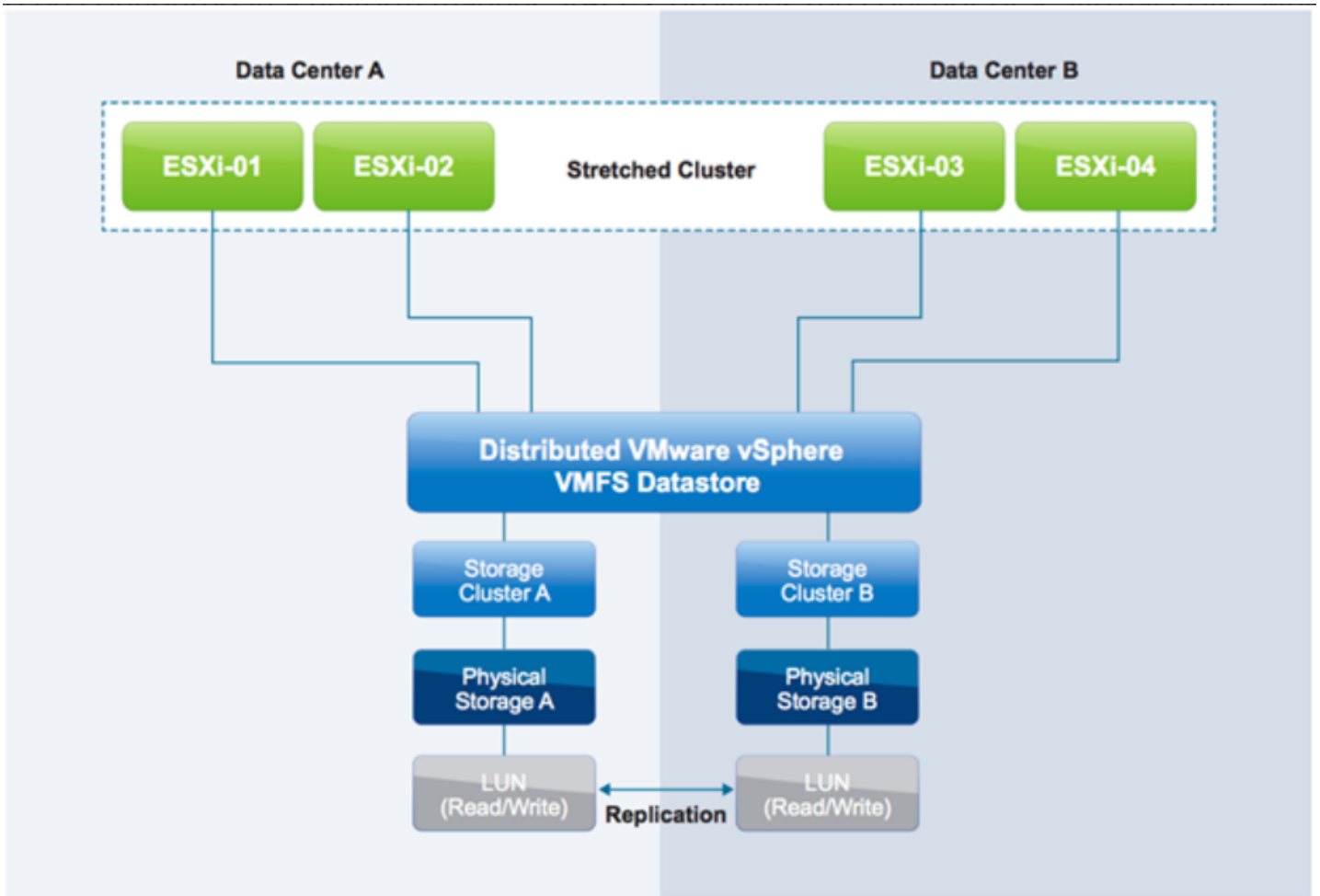


Figure 2 - Non-Uniform Configuration

Our examples use uniform storage because these configurations are currently the most commonly deployed. Many of the design considerations, however, also apply to non-uniform configurations. We point out exceptions when this is not the case.

### Infrastructure Architecture

In this section, we describe the basic architecture referenced in this document. We also discuss the basic configuration and performance of the various vSphere features. For an in-depth explanation of each feature, refer to the vSphere 6.5 Availability Guide and the vSphere 6.5 Resource Management Guide. We make specific recommendations based on VMware best practices and provide operational guidance where applicable. It is explained in our failure scenarios how these best practices prevent or limit downtime.

### Infrastructure

The described infrastructure consists of a single vSphere 6.5 cluster with four ESXi 6.5 hosts. These hosts are managed by a VMware vCenter Server Appliance™ 6.5 instance. The first site is called Frimley; the second site is called Bluefin. The network between Frimley data center and Bluefin data center is a stretched layer 2 network. There is a minimal distance between the sites, as is typical in campus cluster scenarios.

Each site has two vSphere hosts, and the vCenter Server instance is configured with vSphere VM-Host affinity to the hosts in Bluefin data center. In a stretched cluster environment, only a single vCenter Server instance is used. This is different from a traditional Site Recovery Manager configuration in which a dual vCenter Server configuration is required. The configuration of VM-Host affinity rules is discussed in more detail in the “vSphere DRS” subsection of this document.

Eight LUNs are depicted in Figure 3. Four of these are accessed through the virtual IP address active on the iSCSI storage system in the Frimley data center; four are accessed through the virtual IP address active on the iSCSI storage system in the Bluefin data center.

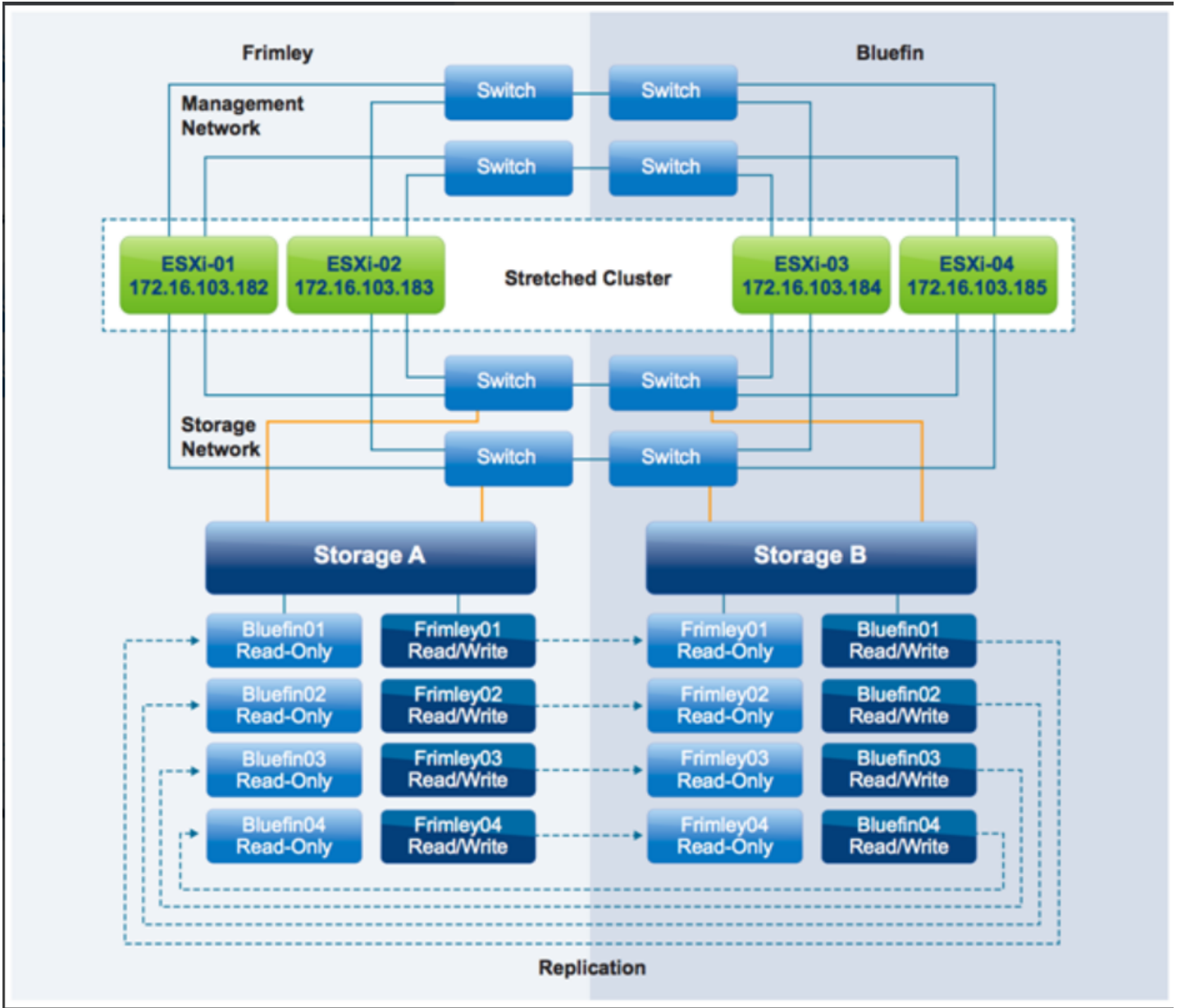


Figure 3 - Test Environment


Table 1: Infrastructure details



The vSphere cluster is connected to a stretched storage system in a fabric configuration with a uniform device access model. This means that every host in the cluster is connected to both storage heads. Each of the heads is connected to two switches, which are connected to two similar switches in the secondary location. For any given LUN, one of the two storage heads present the LUN as read/write via iSCSI. The other storage head maintains the replicated, read-only copy that is effectively hidden from the vSphere hosts.

### vSphere Configuration

Our focus in this document is on vSphere HA, vSphere DRS, and vSphere Storage DRS in relation to stretched cluster environments. Design and operational considerations regarding vSphere are commonly overlooked and underestimated. Much emphasis has traditionally been placed on the storage layer, but little attention has been applied to how workloads are provisioned and managed.

One of the key drivers for using a stretched cluster is workload balance and disaster avoidance. How do we ensure that our environment is properly balanced without impacting availability or severely increasing the operational expenditure? How do we build the requirements into our provisioning process and validate periodically that we still meet them? Ignoring these requirements makes the environment confusing to administrate and less predictable during the various failure scenarios for which it should be of help.

Each of these three vSphere features has very specific configuration requirements and can enhance environment resiliency and workload availability. Architectural recommendations based on our findings during the testing of the various failure scenarios are given throughout this section.

### vSphere HA

Our environment has four hosts and a uniform stretched storage solution. A full site failure is one scenario that must be taken into account in a resilient architecture, alongside host failures, network failures and different types of storage failures. vSphere HA is crucial in any environment to provide a certain level of availability, but even more so in a vMSC configuration. VMware recommends enabling vSphere HA and recommend thoroughly reading the following guidelines for an optimal vSphere HA configuration for vMSC based infrastructures.

### Admission Control

VMware recommends enabling **vSphere HA Admission Control**. Workload availability is the primary driver for most stretched cluster environments, so providing sufficient capacity for a full site failure is recommended. Such hosts are equally divided across both sites. To ensure that all workloads can be restarted by vSphere HA on just one site, configuring the admission control policy to 50 percent for both memory and CPU is recommended.

VMware recommends using a **percentage-based** policy because it offers the most flexibility and reduces operational overhead. Even when new hosts are introduced to the environment, there is no need to change the percentage and no risk of a skewed consolidation ratio due to the possible use of VM-level reservations. For more details about admission control policies and the associated algorithms, refer to the vSphere 6.5 Availability Guide.

Additionally, as of vSphere 6.5, it is also possible to specify how much **Performance Degradation** you are willing to tolerate for your workloads. By default, this setting is configured to 100%. You can change this to your liking, and should be based on your SLA with the business. As this setting is new, we will briefly explain how it works by looking at an example.

An environment has 75GB of memory available in a three-node cluster. One host failure to tolerate is specified and 60GB of memory is actively used by VMs. In the UI it is also specified that 0% resource reduction is tolerated.

vSphere HA will now take a single host failure in to account for this cluster. This results in 75GB - 25GB (1 host worth of memory) = 50GB of memory available to run workloads. There is 60GB of memory used. This implies that with 0% resource reduction to tolerate, 60GB of memory is required. However, after a failure there is only 50GB available, and as such vSphere issues a warning. Note that this does not stop the provisioning of new VMs or the power-on of VMs. That is what Admission Control is for.

Figure 4 shows a vSphere HA cluster configured with Admission Control enabled. Note that as of vSphere 6.5 specific settings around admission control need to be set after creating the cluster.

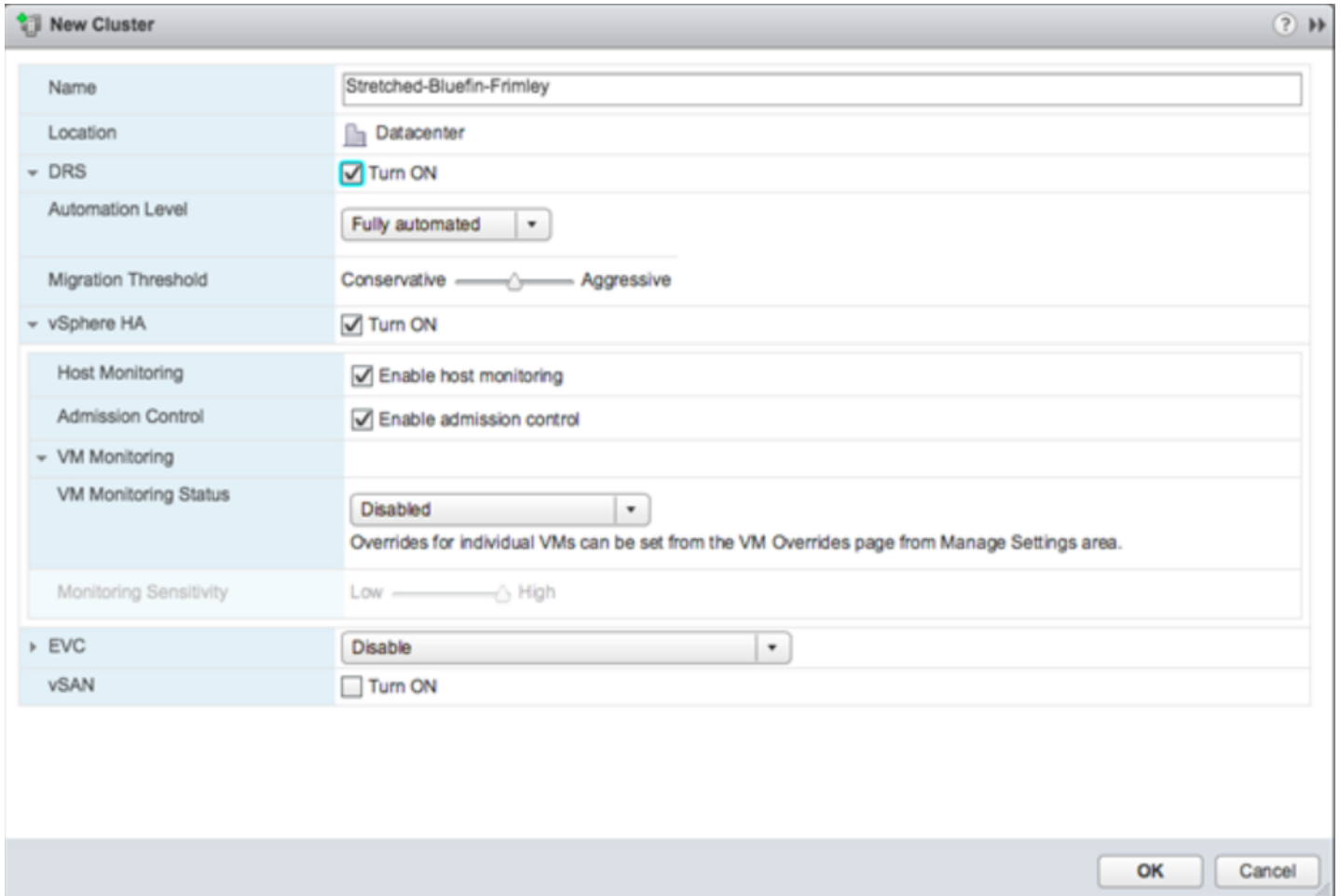


Figure 4 - vSphere HA Configuration

The first recommendation is to ensure that admission control is configured to reserve 50% of both CPU and Memory resources for HA. This is to ensure, that in the case of a full site failure, all VMs can be restarted. Starting vSphere 6.5 the UI for Admission Control has slightly changed. In the interface, you now specify **the number of hosts failures to tolerate** which is then, as shown in Figure 5, converted to a percentage. VMware recommends using the **Cluster Resource Percentage admission control policy** as it is the most flexible policy. For more details, refer to the vSphere Availability Guide.

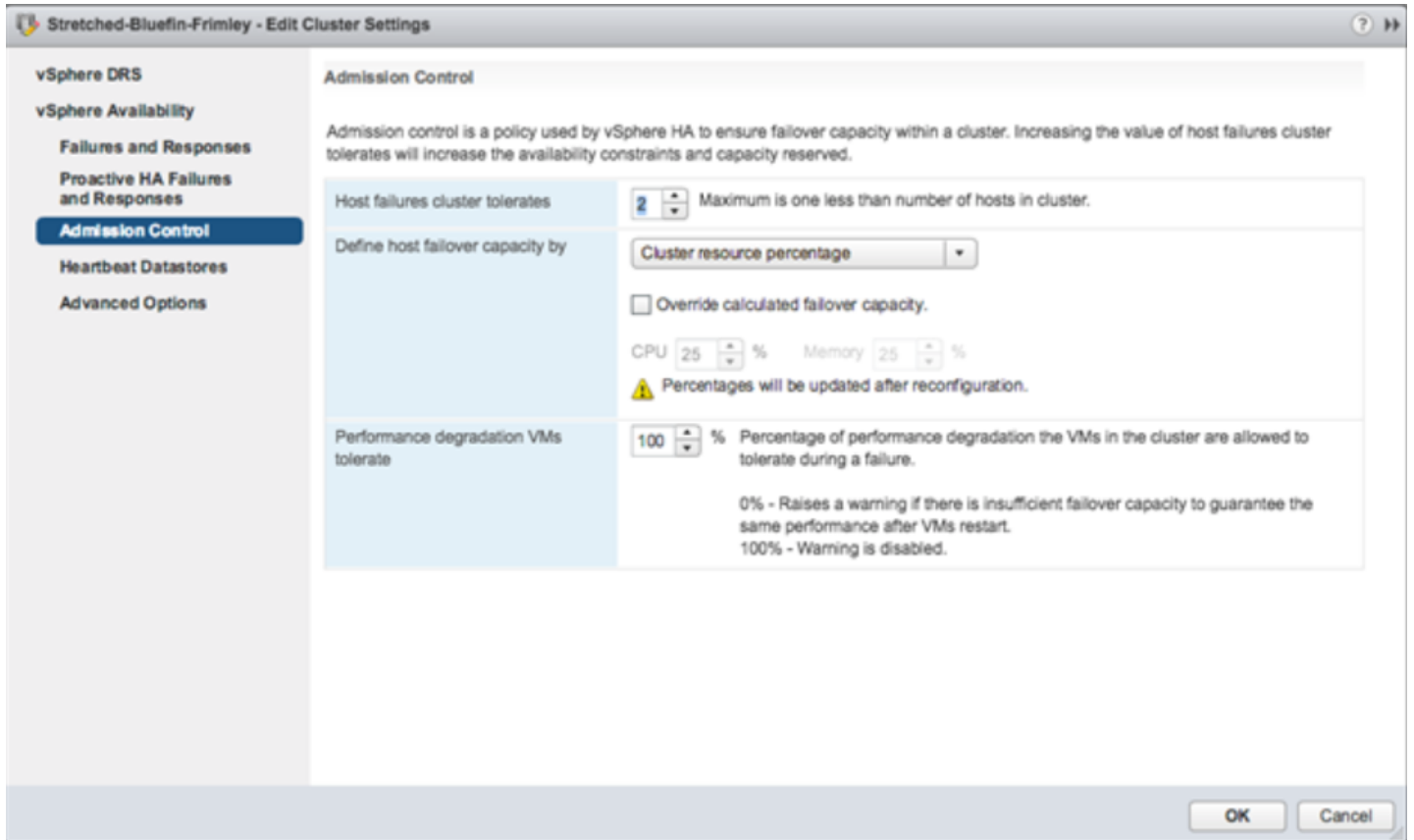


Figure 5 – vSphere HA Admission Control

## vSphere HA heartbeats

vSphere HA uses heartbeat mechanisms to validate the state of a host. There are two such mechanisms: network heartbeating and datastore heartbeating. Network heartbeating is the primary mechanism for vSphere HA to validate the availability of the hosts. Datastore heartbeating is the secondary mechanism used by vSphere HA; it determines the exact state of the host after network heartbeating has failed.

If a host is not receiving any heartbeats, it uses a fail-safe mechanism to detect if it is merely isolated from its primary node or completely isolated from the network. It does this by pinging the default gateway. In addition to this mechanism, one or more isolation addresses can be specified manually to enhance reliability of isolation validation. VMware recommends specifying a minimum of two additional isolation addresses, with each address being local to a particular site.

In our scenario, one of these addresses physically resides in the Frimley data center; the other physically resides in the Bluefin data center. This enables vSphere HA validation for complete network isolation, even in case of a connection failure between sites. Figure 5 shows an example of how to configure multiple isolation addresses. The vSphere HA advanced setting used is `das.isolationaddress`. More details on how to configure this can be found in [VMware Knowledge Base article 1002117](#).

The minimum number of heartbeat datastores is two and the maximum is five. For vSphere HA datastore heartbeating to function correctly in any type of failure scenario, VMware recommends increasing the number of **heartbeat datastores** from two to four in a stretched cluster environment. This provides full redundancy for both data center locations. Defining four specific datastores as preferred heartbeat datastores is also recommended, selecting two from one site and two from the other. This enables vSphere HA to heartbeat to a datastore even in the case of a connection failure between sites. Subsequently, it enables vSphere HA to determine the state of a host in any scenario. Adding an advanced setting called `das.heartbeatDsPerHost` can increase the number of heartbeat datastores. This is shown in Figure 6.

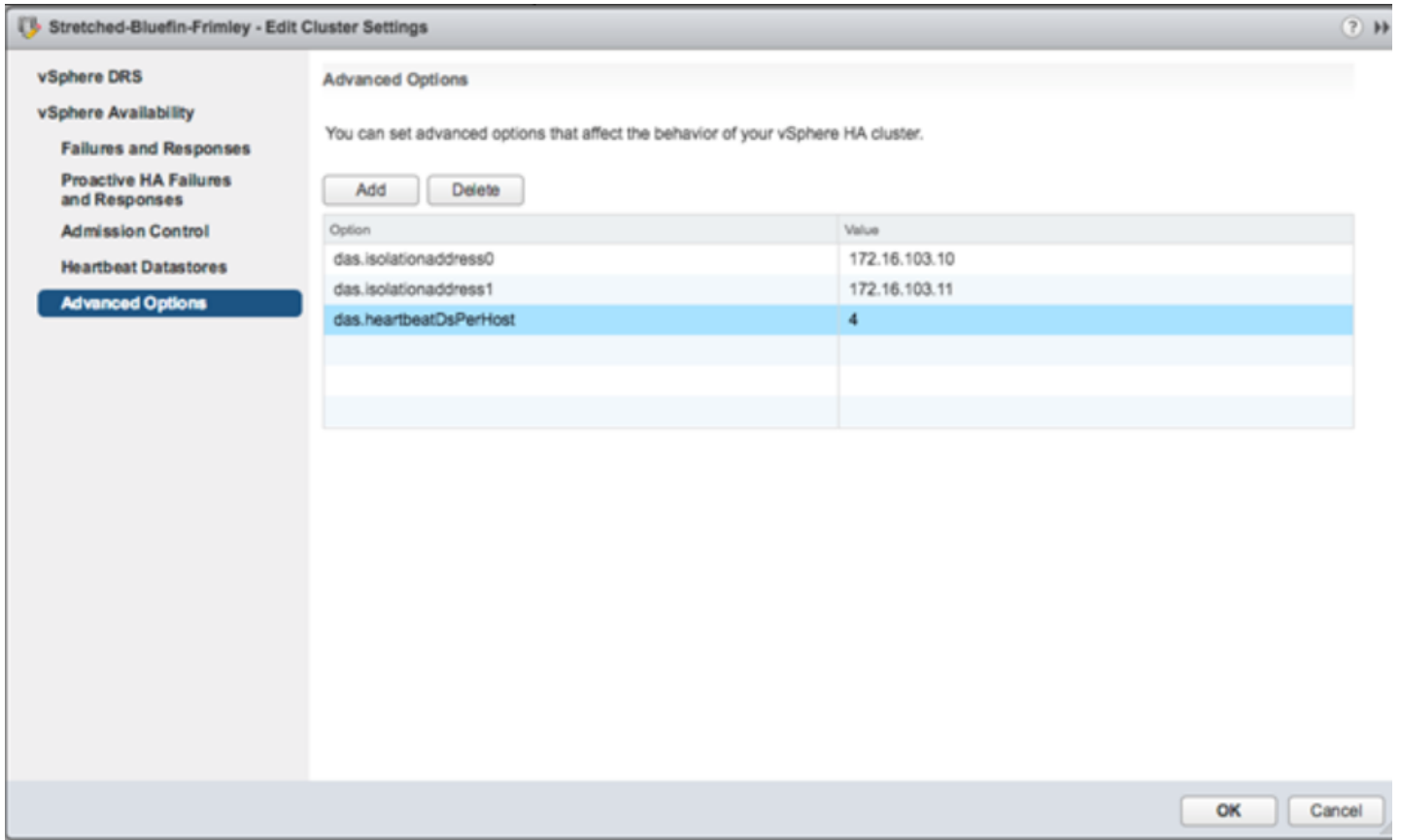


Figure 6 - vSphere HA Advanced Options

To designate specific datastores as heartbeat devices, VMware recommends using **Use datastores from the specified list and complement automatically if needed**. This enables vSphere HA to select any other datastore if the four designated datastores that have been manually selected become unavailable. VMware recommends selecting two datastores in each location to ensure that datastores are available at each site in the case of a site partition.

### Heartbeat Datastores

vSphere HA uses datastores to monitor hosts and virtual machines when management network has failed. vCenter Server selects two datastores for each host using the policy and datastore preferences specified below.

Heartbeat datastore selection policy:

- Automatically select datastores accessible from the host
- Use datastores only from the specified list
- Use datastores from the specified list and complement automatically if needed

Figure 7 - Datastore heartbeating

### Permanent Device Loss and All Paths Down Scenarios

As of vSphere 6.0, enhancements have been introduced to enable an automated failover of VMs residing on a datastore that has either an all paths down (APD) or a permanent device loss (PDL) condition. PDL is applicable only to block storage devices.

A PDL condition, as is discussed in one of our failure scenarios, is a condition that is communicated by the array controller to the vSphere host via a SCSI sense code. This condition indicates that a device (LUN) has become unavailable and is likely permanently unavailable. An example scenario in which this condition is communicated by the array is when a LUN is set offline. This condition is used in non-uniform models during a failure scenario to ensure that the vSphere host takes appropriate action when access to a LUN is revoked. When a full storage failure occurs, it is impossible to generate the PDL condition because there is no communication possible between the array and the vSphere host. This state is identified by the vSphere host as an APD condition.

Another example of an APD condition is where the storage network has failed completely. In this scenario, the vSphere host also does not detect what has happened with the storage and declares an APD.

To enable vSphere HA to respond to both an APD and a PDL condition, vSphere HA must be configured in a specific way. VMware recommends enabling **VM Component Protection (VMCP)**. Note that in the current UI this feature is not labeled as “VM Component Protection” or “VMCP”. Within the UI you simply specify the response to a **Datastore with PDL** and a **Datastore with APD** under **Failures and Responses** as shown in Figure 8.

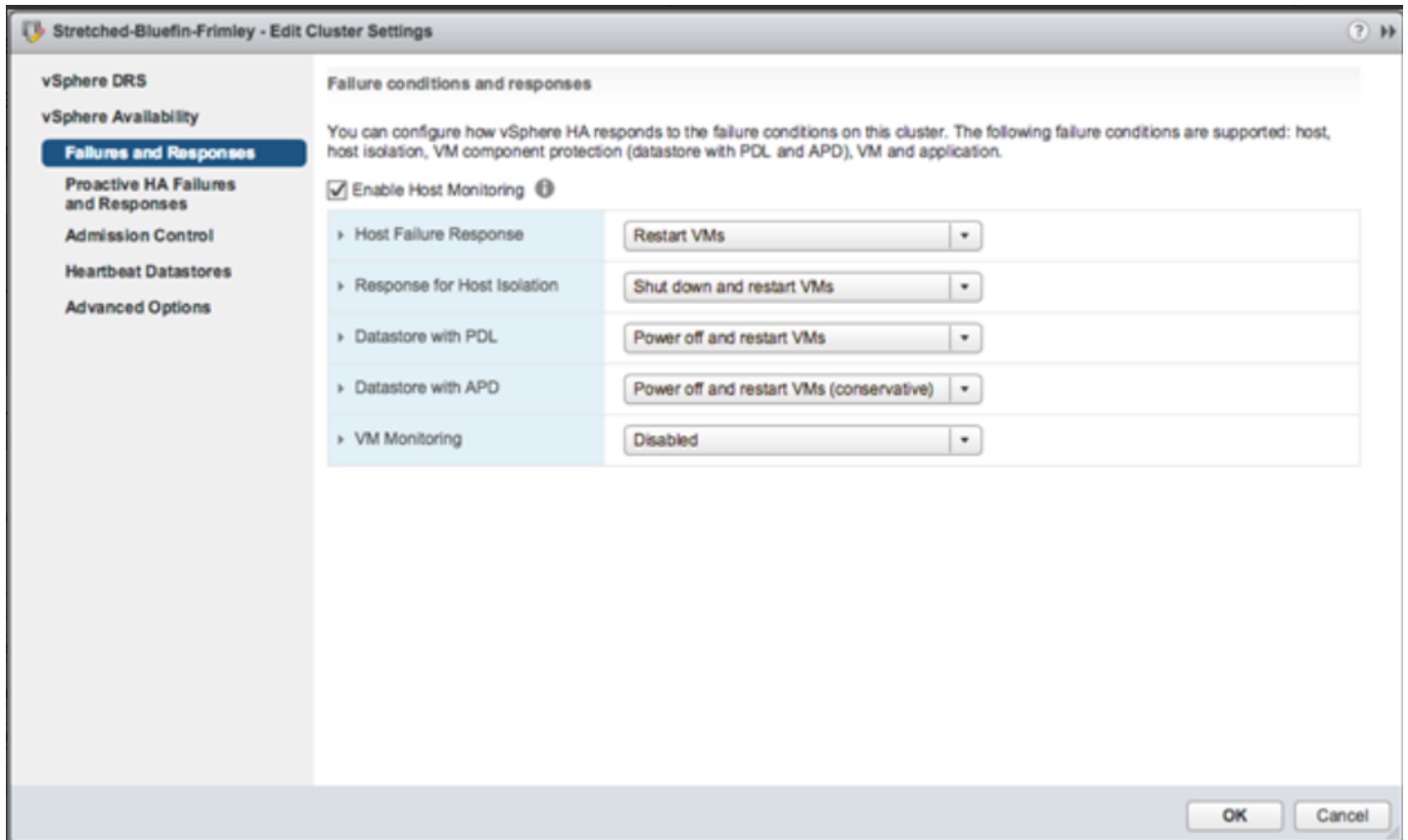


Figure 8 – VM Component Protection

The configuration screen can be found as follows:

- Log in to VMware vSphere Web Client.
- Click **Hosts and Clusters**.
- Click the cluster object.
- Click the **Manage** tab.
- Click **vSphere HA** and then **Edit**.
- Select **Failures and Responses**.
- Select individual functionality, as displayed in figure 8.

The configuration for PDL is basic. In the **Failures and Responses** section, the response following the detection of a PDL condition can be configured. VMware recommends setting this to **Power off and restart VMs**. When this condition is detected, a VM is restarted instantly on a healthy host within the vSphere HA cluster.

For an APD scenario, the configuration must occur in the same section, as is shown in Figure 8. Besides defining the response to an APD condition, it is also possible to alter the timing and to configure the behavior when the failure is restored before the APD timeout has passed as shown in Figure 9.

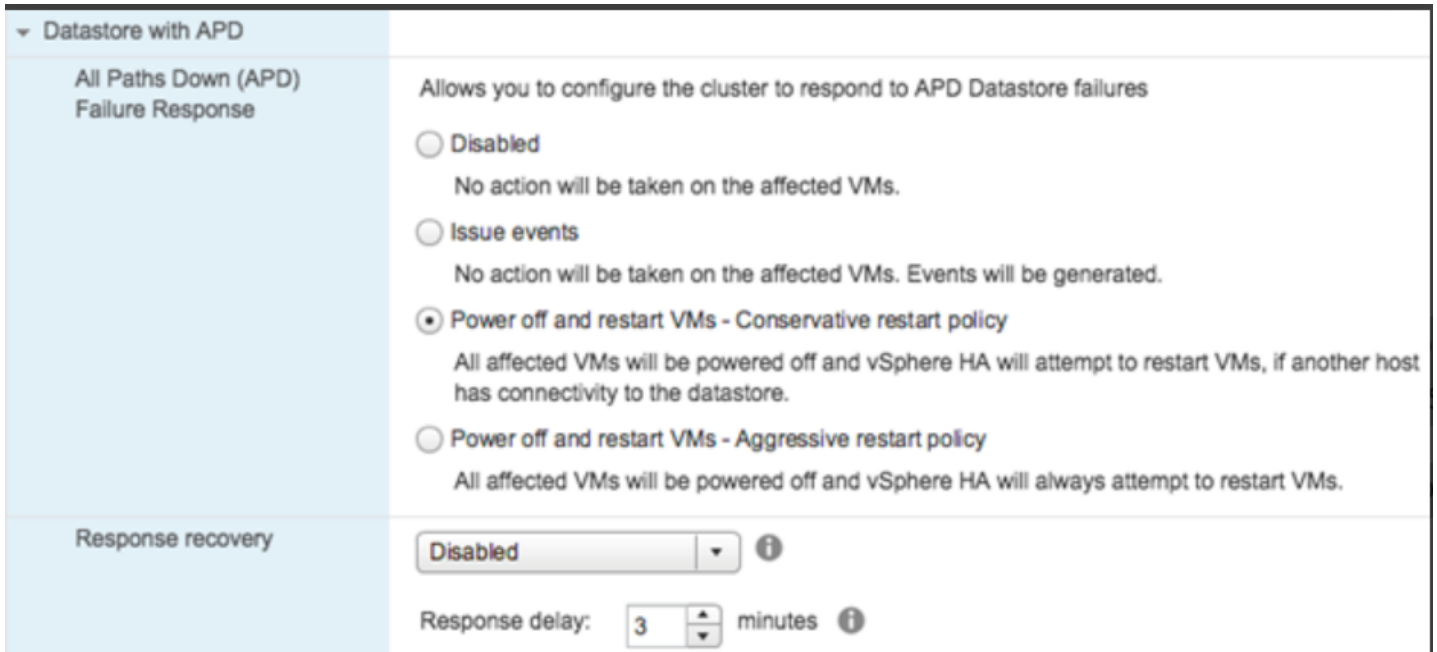


Figure 9 - VMCP Detailed Configuration

When an APD condition is detected, a timer is started. After 140 seconds, the APD condition is officially declared and the device is marked as APD timeout. When 140 seconds have passed, vSphere HA starts counting. The default vSphere HA timeout is 3 minutes. When the 3 minutes have passed, vSphere HA restarts the impacted VMs, but VMCP can be configured to respond differently if preferred. VMware recommends configuring it to **Power off and restart VMs - Conservative restart policy**.

*Conservative* refers to the likelihood that vSphere HA will be able to restart VMs. When set to conservative, vSphere HA restarts only the VM that is impacted by the APD if it detects that a host in the cluster can access the datastore on which the VM resides. In the case of *aggressive*, vSphere HA attempts to restart the VM even if it doesn't detect the state of the other hosts. This can lead to a situation in which a VM is not restarted because there is no host that has access to the datastore on which the VM is located.

If the APD is lifted and access to the storage is restored before the timeout has passed, vSphere HA does not unnecessarily reset the VM unless explicitly configured to do so. If a response is chosen even when the environment has recovered from the APD condition, **Response recovery** can be configured to **Reset VMs**. VMware recommends leaving this setting **deactivated**. Note that if **Reset VMs** is selected, any configured Restart Priority and / or dependency will not be taken in to account during the reset phase. Restart priority and / or dependency is only applied when VMs are restarted. (A restart is substantially different then a reset!)

Starting with vSphere 5.5, an advanced setting called **Disk.AutoremoveOnPDL** was introduced and is implemented by default. This functionality enables vSphere to remove devices marked as PDL and helps prevent reaching, for example, the 512-device limit for a vSphere host. However, if the PDL scenario is resolved and the device returns, the vSphere host's storage system must be rescanned before the device will appear. VMware recommends disabling Disk.AutoremoveOnPDL for vSphere 5.5 hosts by setting the host advanced settings value to 0. For **vSphere 6.0 hosts**, this advanced setting is no longer required to be changed from the default configuration to properly recover the devices marked as PDL, it should be **set to 1**. Please ensure to change the setting from 0 to 1 when upgrading from vSphere 5.5 to vSphere 6.0.

### Restart ordering and priority

Starting with vSphere 6.5 it is now possible to provide additional granularity and control in terms of restart sequence and dependencies for VMs in the **VM Overrides** section. Pre-vSphere 6.5 it was already possible to set a restart priority, but as each host in a cluster could power-on 32 VMs at a time and there was no option to specify a delay or a dependency of any kind, it would usually lead to all VMs being powered-on simultaneously. Per vSphere 6.5 it is now possible to specify to which priority group each VM belongs ( **VM Restart Priority**: Lowest, low, medium, high, highest).

VMware recommends configuring **VM Restart Priority** for important infrastructure components like DNS or Active Directory, and for applications which are formed out of multiple tiers. For example, a VM may have an application, web, and database tier where, in most cases, the database tier would need to be started first. First, select all the VMs for which you want to change the restart priority as shown in Figure 10. By default, all VMs have a restart priority of Medium.

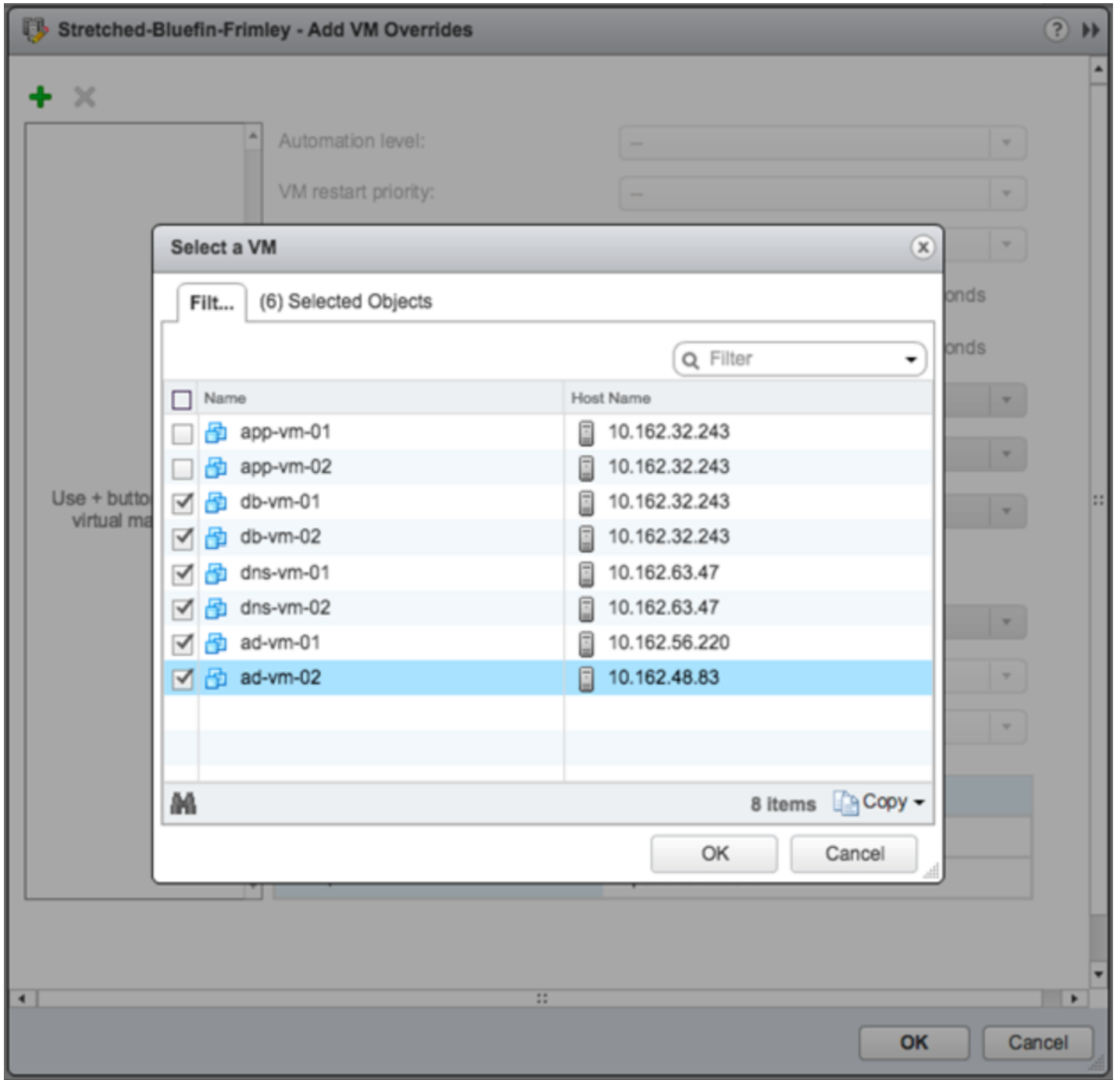


Figure 10 - VM Restart Priority

Next, select the desired restart priority. As the selected VMs in our case are important for the application layer we give them the restart priority of **Highest** as shown in Figure 11.



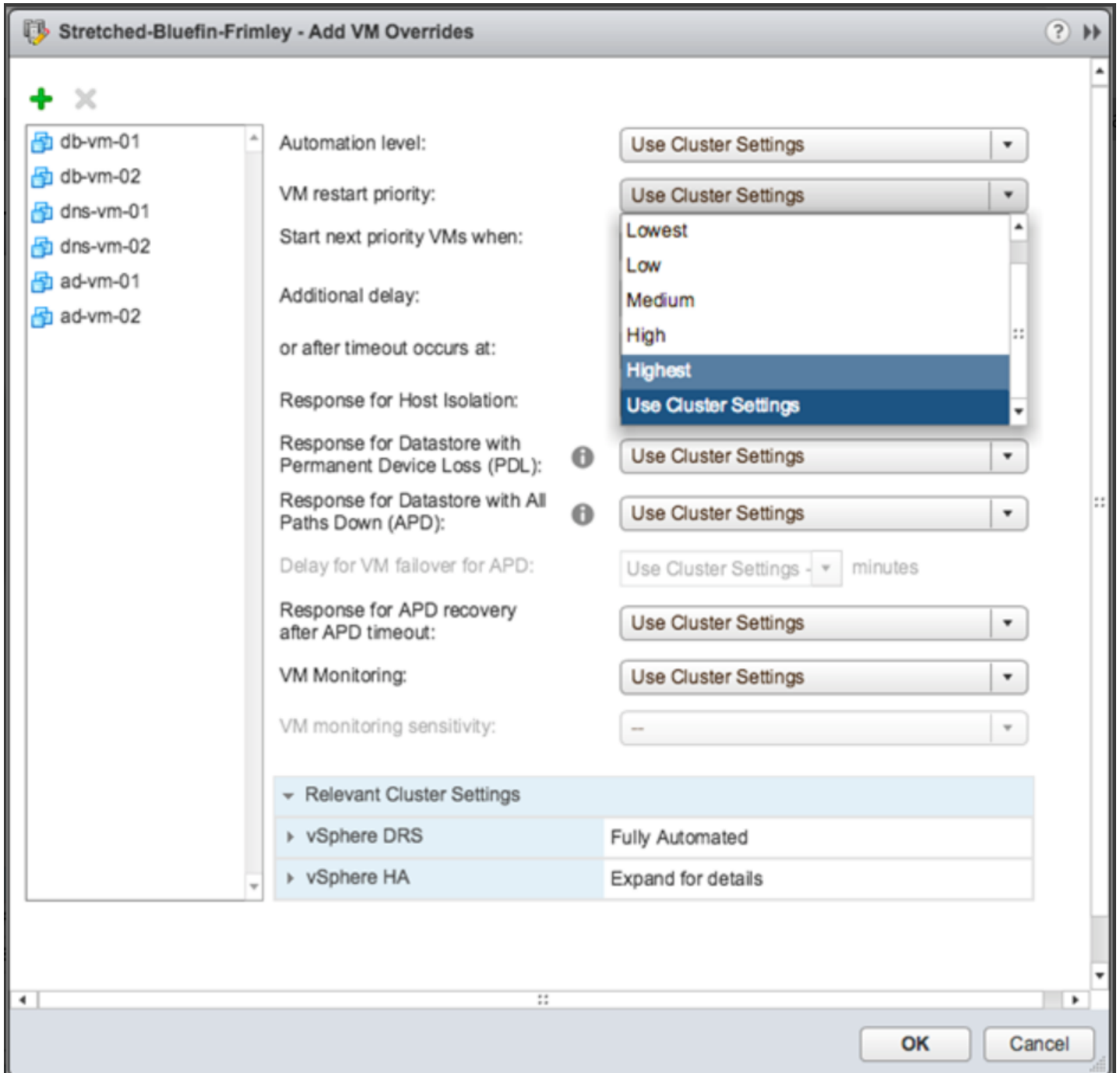


Figure 11 – Changing VM Restart Priority

On top of that, it is possible to specify when to start powering-on the VMs which belong to the next group. This is specified by the “ **Start next priority VMs when** ” dropdown. You can set this to “guest heartbeats detected” for instance, which means vSphere HA will wait for VMware Tools to report a “liveness” heartbeat, as shown in Figure 12. If, however, a heartbeat is not received then by default after 600 seconds the next group is started. This can be increased, or decreased, by specifying the “ **or after timeout occurs at** ” value. VMware recommends to leave this set to the default of 600.



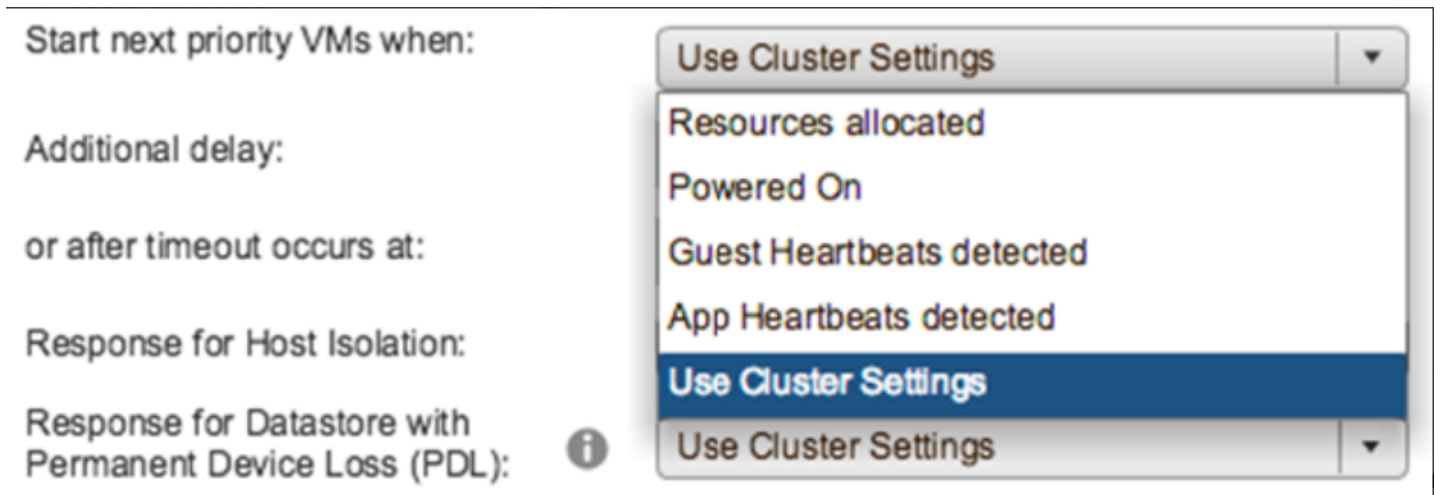


Figure 12 - Additional Delay

Starting vSphere 6.5 there is a second option to specify restart order, but in this case, it is a restart dependency between VMs. This is leveraging the vSphere HA and DRS cluster rules. A dependency can simply be created by creating two VM groups (Figure 13) and then defining a VM to VM rule in which the dependency is specified as shown in Figure 14. Here the next group is only started when the configured **VM Dependency Restart Condition** is met. If the cluster wide condition is set to the default **Resources Allocated**, then the second group of VMs is powered-on a split second after the first group as this is purely a restart scheduling exercise. **Powered-on** or even **Guest Heartbeats** detected are more appropriate in most cases. Note that the specified rules are considered mandatory rules and these rules will not be violated as a result. In other words, if the power-on of the first group is unsuccessful and the specified condition is “powered on” then the second group will never be powered on.

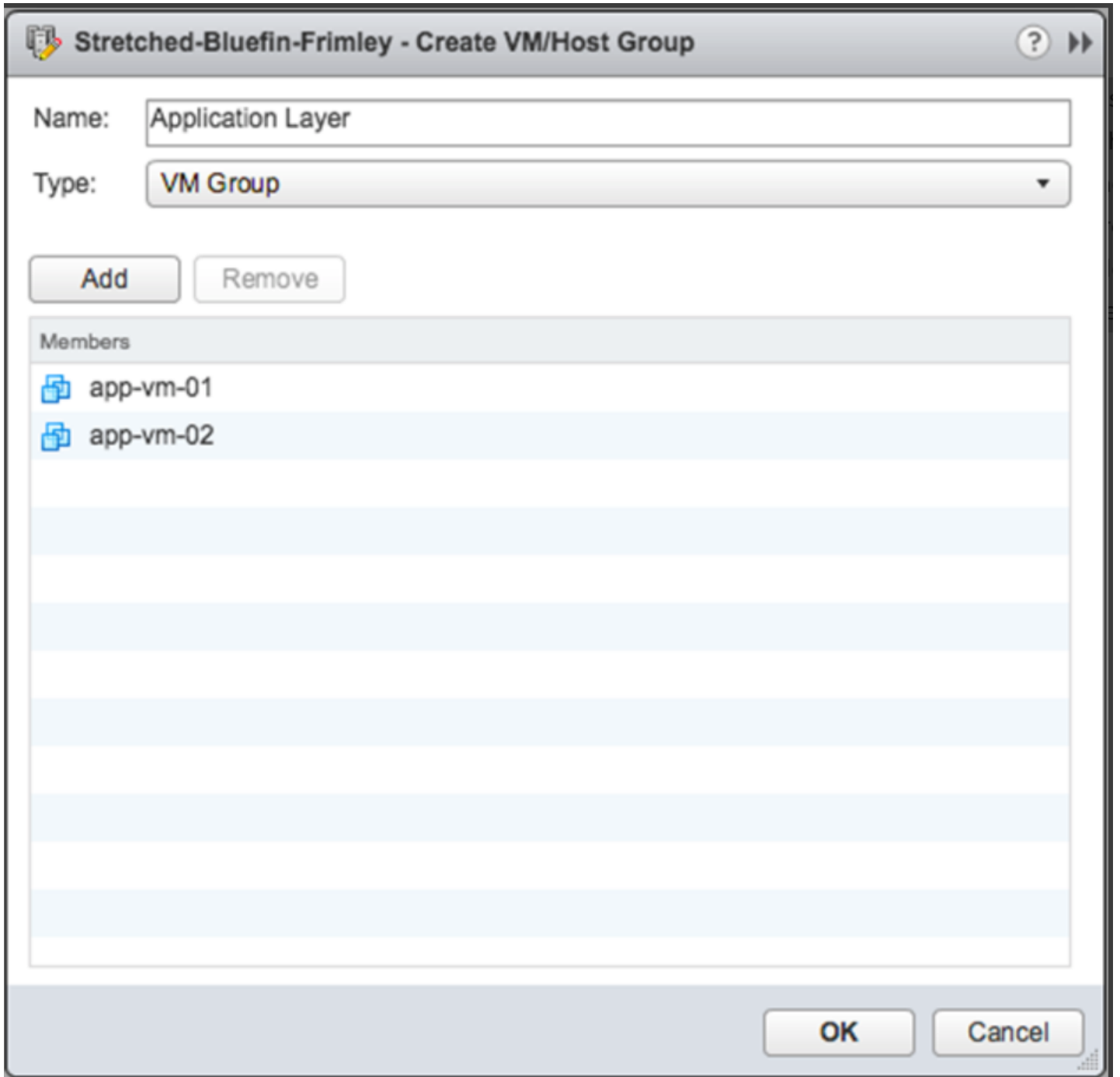


Figure 13 - Restart Dependency

**Stretched-Bluefin-Frimley - Create VM/Host Rule**

Name:

Enable rule.

Type:

Description:

Virtual machines in the VM group Database Layer will be restarted first. Virtual machines in the VM group Application Layer will be restarted afterwards, when the cluster dependency restart condition has been met.

First restart VMs in VM group:

Then restart VMs in VM group:

OK Cancel

Figure 14 - Restart Dependency

### ProActive HA

ProActive HA was introduced in vSphere 6.5 and is found in the UI under Availability but technically it is a function of vSphere DRS. ProActive HA enables you to configure actions for events that may lead to VM downtime. Downtime avoidance is the key reason typically for deploying a vMSC configuration, therefore VMware recommends **enabling ProActive HA**.

In order to enable ProActive HA, a health provider (vSphere Web Client Plugin) needs to be installed first. At the time of writing, only the following server vendors provide a health provider: HPE, Dell, and Cisco. After the installation of the health provider, Proactive HA can be enabled. VMware recommends to set the Proactive HA automation level to **Automated** so that action is immediately taken when a potential issue arises.

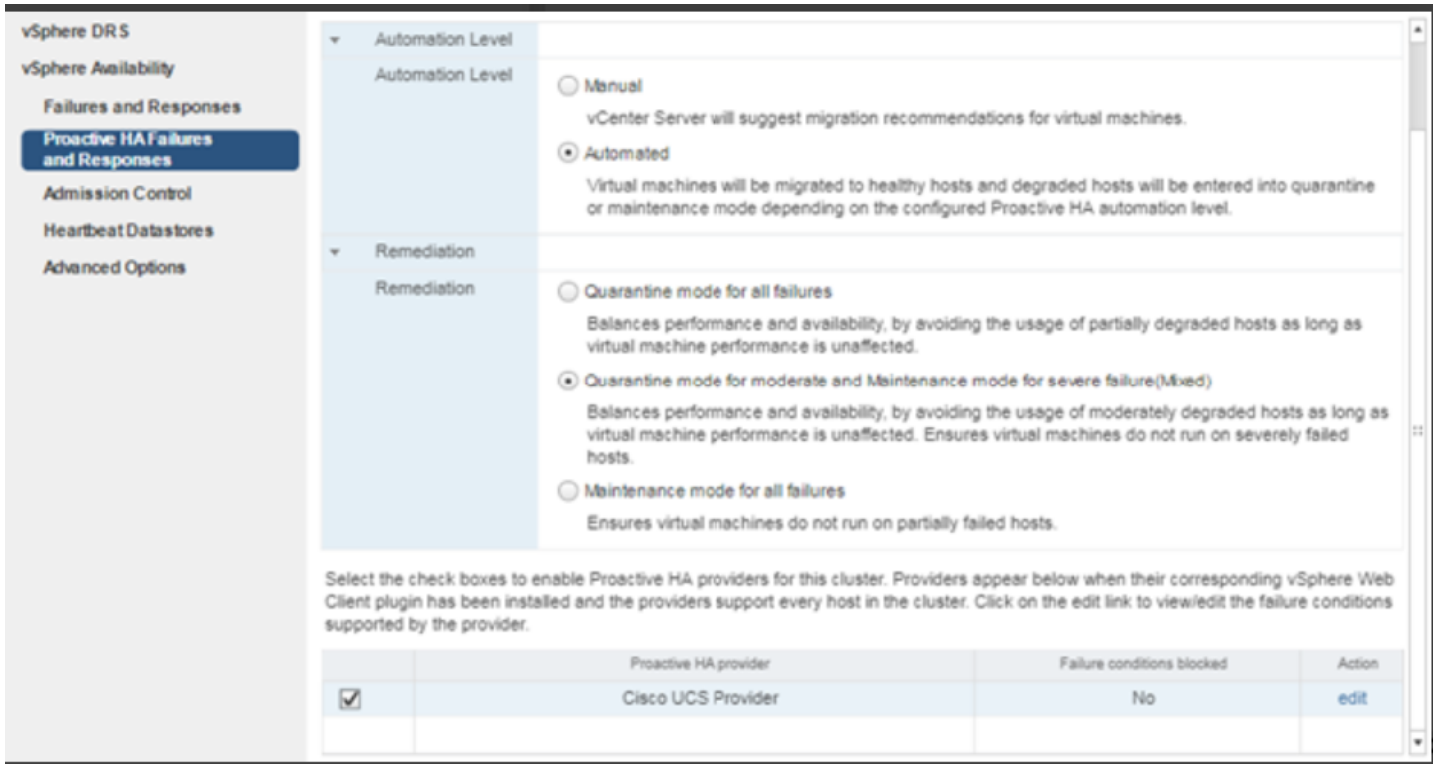


Figure 15 – ProActive HA Configuration

Next, the **Remediation** needs to be configured (also shown in figure 15). VMware recommends configuring this with **Quarantine mode for moderate and Maintenance mode for severe failure(Mixed)**. The other options are **Maintenance Mode** and **Quarantine Mode**. Before selecting either, it is useful to understand the difference.

- Quarantine Mode – Only places the host in quarantine mode. vSphere DRS will try to migrate VMs from this host to others. However, if there is some level of over-commitment or vSphere HA/DRS Cluster rules are defined VMs may not be migrated.
- Maintenance Mode – All VMs will be migrated
- Quarantine for Moderate / Maintenance for severe – Depending on the type of failure, the host will either be placed into Quarantine Mode or Maintenance Mode. This is best described as conservative (should) and aggressive (must) host evacuation.

ProActive HA can respond to different types of failures, depending on the version of the health provider plugin and the different types of vendor components that are being monitored. For example, Dell OpenManage (OMIVV) 4.0.x only supports the monitoring of Power Supplies, Fans, and Storage (SD Cards). The ProActive HA framework however also additionally supports Memory and Network monitoring. However, responses to Memory and Network failures can only be configured when the health provider supports this.

When an issue arises, the severity of the issue determines the ProActive HA response. These severities are color-coded (Table 2), where Yellow and Red are the states in which ProActive HA takes action. Some plugins will allow you to specify the severity of an issue manually, like Dell OpenManage. VMware recommends however to leave severity as pre-defined by the server vendor.


Table 2:Status ProActive HA

For more details on configuring the health provider and health provider specific settings we would like to refer to the server vendor documentation.

### vSphere DRS

vSphere DRS is used in many environments to distribute the load within a cluster. It offers many other features that can be very helpful in stretched cluster environments. VMware recommends enabling vSphere DRS to facilitate load balancing across hosts in the cluster. The vSphere DRS load-balancing calculation is based on CPU and memory use. Care should be taken with regard to both storage and networking resources as well as to traffic flow. To avoid storage and network traffic overhead in a stretched cluster environment, VMware recommends implementing vSphere DRS affinity rules to enable a logical separation of VMs. This subsequently helps improve availability. For VMs that are responsible for infrastructure services, such as Microsoft Active Directory and DNS, it assists in ensuring the separation of these services across sites.

### Site Affinity

**vSphere DRS affinity rules** also help prevent unnecessary downtime, and storage and network traffic flow overhead, by enforcing preferred site affinity. VMware recommends aligning vSphere VM-to-host affinity rules with the storage configuration—that is, setting VM-to-host affinity rules with a preference that a VM runs on a host at the same site as the array that is configured as the primary read/write node for a given datastore. For example, in our test configuration, VMs stored on the Frimley01 datastore are set with VM-to-host affinity with a preference for hosts in the Frimley data center. This ensures that in the case of a network connection failure between sites, VMs do not lose connection with the storage system that is primary for their datastore. VM-to-host affinity rules aim to ensure that VMs stay local to the storage primary for that datastore. This coincidentally also results in all read I/O's staying local.

*NOTE: Different storage vendors use different terminology to describe the relationship of a LUN to a particular array or controller. For the purposes of this document, we use the generic term " storage site affinity, " which refers to the preferred location for access to a given LUN.*

VMware recommends implementing " **should rules** " because these are violated by vSphere HA in the case of a full site failure. The availability of services should always prevail. In the case of " **must rules** ," vSphere HA does not violate the rule set, and this can potentially lead to service outages. In the scenario where a full data center fails, "must rules" do not allow vSphere HA to restart the VMs, because they do not have the required affinity to start on the hosts in the other data center. This necessitates the recommendation to implement "should rules." vSphere DRS communicates these rules to vSphere HA, and these are stored in a "compatibility list" governing allowed start-up. If a single host fails, VM-to-host "should rules" are respected by default. Pre-vSphere 6.5 VMware recommended configuring the vSphere HA rule settings to **respect VM-to-host affinity rules** where possible, as by default vSphere HA used to ignore these should rules during a restart event.

With a full site failure, vSphere HA can restart the VMs on hosts that violate the rules. Availability takes preference in this scenario. In Figure 16 below it can be seen what this looked like with vSphere 6.0. In vSphere 6.5 this option disappeared completely, by default vSphere HA respects the rules going forward.



Figure 16 - vSphere HA Rule Settings

Under certain circumstances, such as massive host saturation coupled with aggressive recommendation settings, vSphere DRS can also violate "should rules." Although this is very rare, we recommend monitoring for violation of these rules because a violation might impact availability and workload performance. If there is a desire to change the default behavior the following advanced settings can be configured:

das.respectVmVmAntiAffinityRules - set to "true" by default, set to "false" if you want to deactivate vSphere HA respecting VM-VM affinity and anti-affinity rules

das.respectVmHostSoftAffinityRules - set to "true" by default, set to "false" if you want to deactivate vSphere HA respecting VM-Host affinity rules

VMware recommends manually defining “sites” by creating a group of hosts that belong to a site and then adding VMs to these sites based on the affinity of the datastore on which they are provisioned. In our scenario, only a limited number of VMs were provisioned. VMware recommends automating the process of defining site affinity by using tools such as VMware vRealize Orchestrator™ or VMware vSphere PowerCLI™. If automating the process is not an option, use of a generic naming convention is recommended to simplify the creation of these groups. VMware recommends that these groups be validated on a regular basis to ensure that all VMs belong to the group with the correct site affinity.

Figures 17 through 20 depict the configuration used for this white paper. In the first screenshot, all VMs that should remain local to the Bluefin data center are added to the Bluefin VM group.

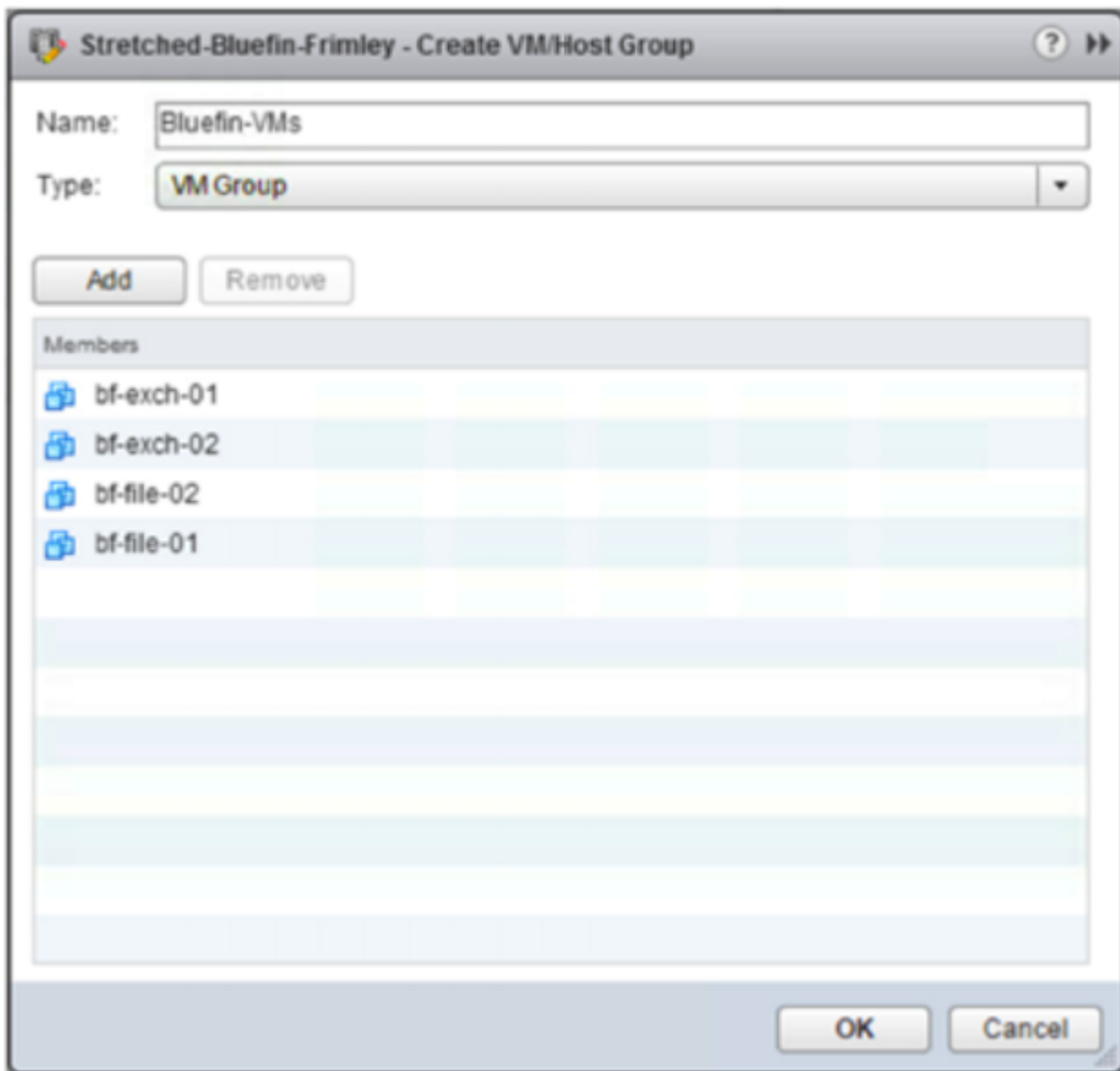


Figure 17 - VM Group

Next, a Bluefin host group is created that contains all hosts residing in this location.

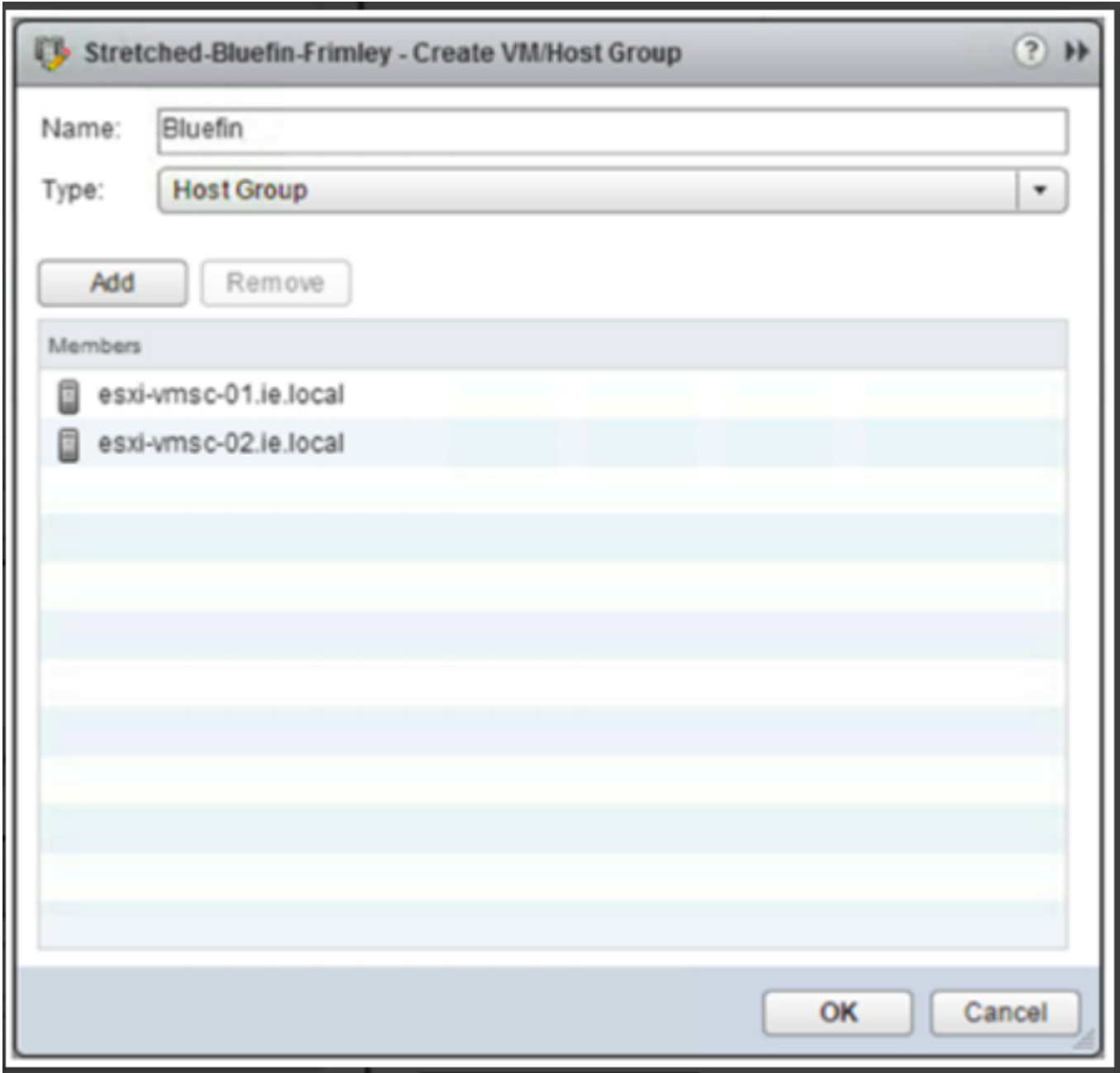


Figure 18 – Host Group

Next, a new rule is created that is defined as a “should run on rule.” It links the host group and the VM group for the Bluefin location.

**Stretched-Bluefin-Frimley - Create VM/Host Rule**

Name:

Enable rule.

Type:

Description:  
Virtual machines that are members of the Cluster VM Group Bluefin-VMs should run on host group Bluefin.

VM Group:

Host Group:

Figure 19 - Rule Definition

This should be done for both locations, which should result in two rules.



VM/Host Rules

Add... Edit... Delete


Name	Type	Enabled	Conflicts	Defined By
 Bluefin-Affinity(VMstoHosts)	Run VMs on Hosts	Yes	0	User
 Frimley-Affinity(VMstoHosts)	Run VMs on Hosts	Yes	0	User

Figure 20 - VM-to-host rules

### Advanced Settings

Starting vSphere 6.5 the UI for vSphere DRS has also changed. In the past many configurations required advanced settings to be entered in the DRS configuration screen. Now that these settings are easier to find, the chances are that you will also want to try these. Note that we do not describe all advanced configuration options here, or any functionality which has no specific impact on vMSC configurations. As an example, the option **VM Distribution** allows you to distribute the number of VMs more evenly across hosts. However, this is based on the number of VMs per host, and not on resources. In order to ensure distribution, it can (and will) ignore any configured non-mandatory (should) “VM-to-Host” rules. In other words, it could force VMs to be placed in a location where you do not expect them to be placed. Before using any of these new options, ensure that after configuring you re-test the different failures scenarios and validate the outcome with the expected outcome.

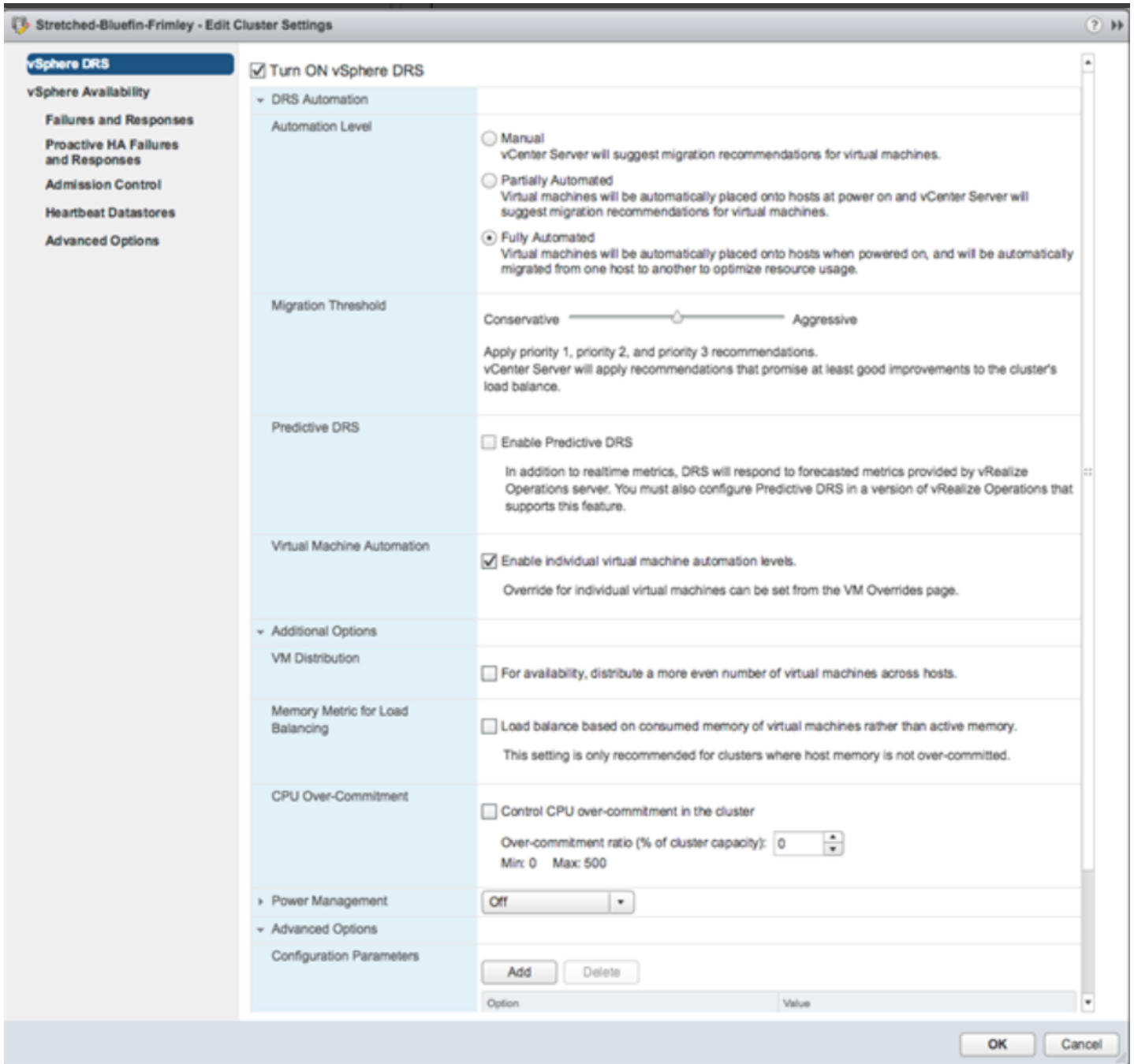


Figure 21 – Advanced DRS options

### Correcting Affinity Rule Violation

vSphere DRS assigns a high priority to correct affinity rule violations. During the invocation, the primary goal of vSphere DRS is to correct any violations and generate recommendations to migrate VMs to the hosts listed in the host group. These migrations have a higher priority than load-balancing moves and are started before them.

vSphere DRS is invoked every **5 minutes** by default, but it is also triggered if the cluster detects changes. For instance, when a host reconnects to the cluster, vSphere DRS is invoked and generates recommendations to correct the violation. Our testing has shown that vSphere DRS generates recommendations to correct affinity rules violations within 30 seconds after a host reconnects to the cluster. vSphere DRS is limited by the overall capacity of the vSphere vMotion network, so it might take multiple invocations before all affinity rule violations are corrected.

### vSphere Storage DRS

vSphere Storage DRS enables aggregation of datastores to a single unit of consumption from an administrative perspective, and it balances VM disks when defined thresholds are exceeded. It ensures that sufficient disk resources are available to a workload. VMware recommends enabling vSphere Storage DRS in Manual Mode (Figure 22) with I/O Metric deactivated (Figure 23). The use of I/O Metric or VMware vSphere Storage I/O Control is **not supported in a vMSC** configuration, as is described in [VMware Knowledge Base article 2042596](#).

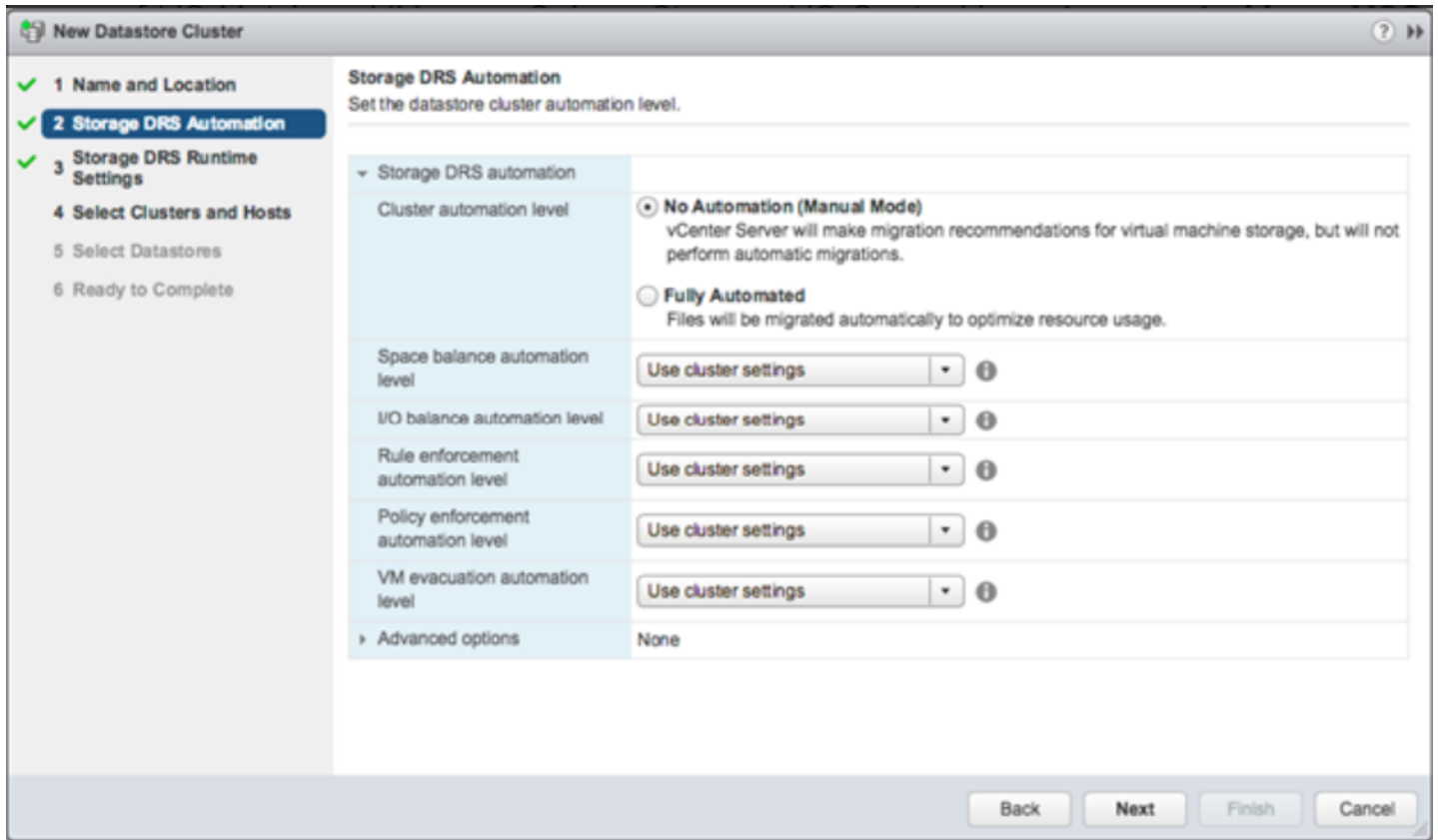


Figure 22 - Storage DRS Configuration

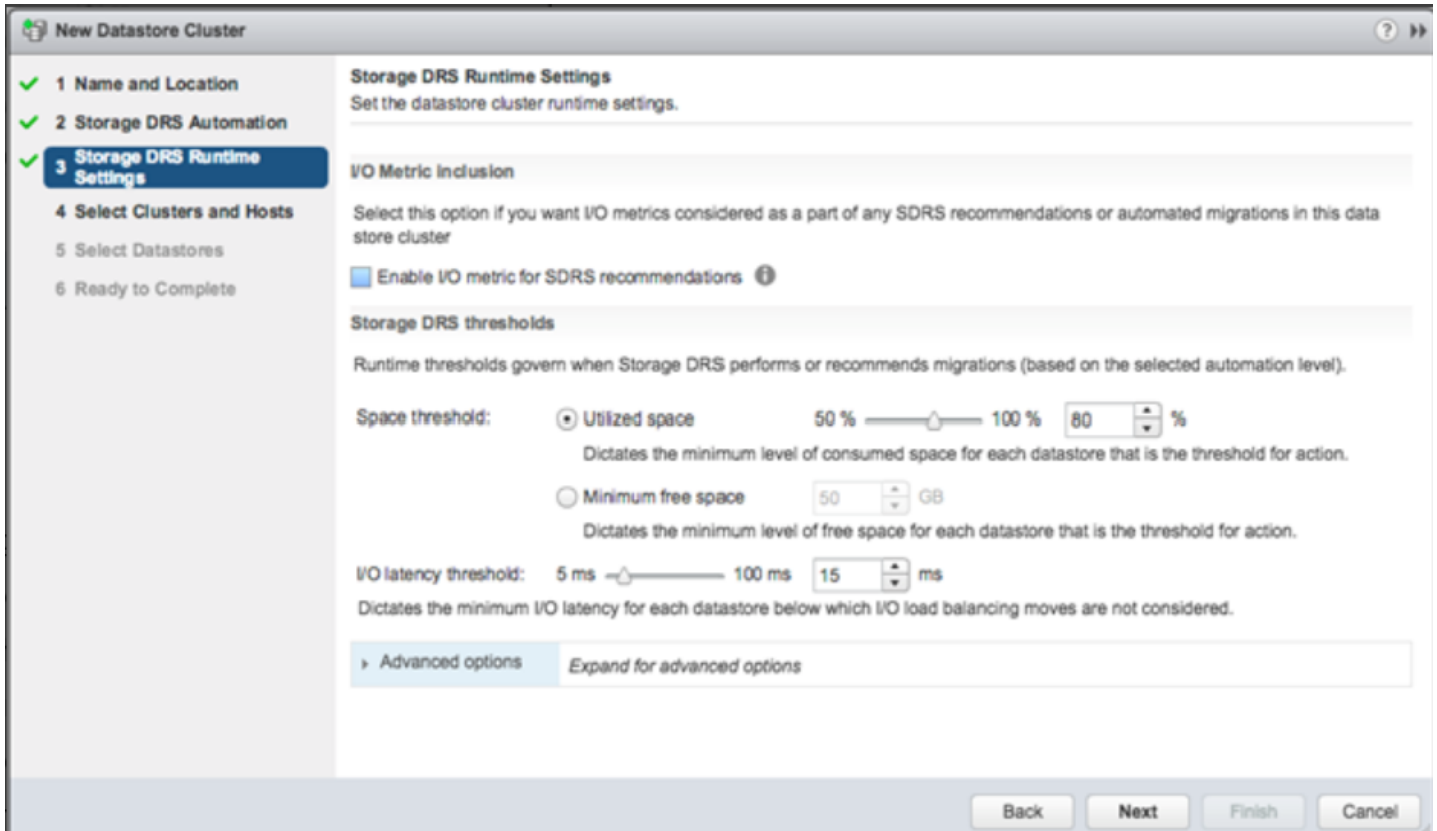


Figure 23 – Deactivate IO Metric inclusion

## Migrations

vSphere Storage DRS uses vSphere Storage vMotion to migrate VM disks between datastores within a datastore cluster. Because the underlying stretched storage systems use synchronous replication, migration or series of migrations have an impact on replication traffic and might cause the VMs to become temporarily unavailable due to contention for network resources during the movement of disks. Migration to random datastores can also potentially lead to additional I/O latency in uniform host access configurations if VMs are not migrated along with their virtual disks. For example, if a VM residing on a host at site A has its disk migrated to a datastore at site B, it continues operating but with potentially degraded performance. The VM's disk reads now are subject to the increased latency associated with reading from the array at site B. Reads are subject to inter-site latency rather than being satisfied by a local target.

To control if and when migrations occur, VMware recommends configuring vSphere Storage DRS in manual mode. This enables human validation per recommendation as well as recommendations to be applied during off-peak hours while gaining the operational benefit and efficiency of the initial placement functionality.

VMware recommends creating datastore clusters based on the storage configuration with respect to storage site affinity. Datastores with a site affinity for site A should not be mixed in datastore clusters with datastores with a site affinity for site B. This enables operational consistency and eases the creation and ongoing management of vSphere DRS VM-to-host affinity rules. Ensure that all vSphere DRS VM-to-host affinity rules are updated accordingly when VMs are migrated via vSphere Storage vMotion between datastore clusters and when crossing defined storage site affinity boundaries. To simplify the provisioning process, VMware recommends aligning naming conventions for datastore clusters and VM-to-host affinity rules. Note that vSphere Storage DRS automatically keeps all VMDKs of the same VM on the same datastore. It does this leveraging Storage DRS affinity rules, VMware recommends to not change this default configuration as it could impact availability during various failure scenarios.

The naming convention used in our testing gives both datastores and datastore clusters a site-specific name to provide ease of alignment of vSphere DRS host affinity with VM deployment in the correlate site.

## Failure Scenarios

There are many failures that can be introduced in clustered systems. But in a properly architected environment, vSphere HA, vSphere DRS, and the storage subsystem do not detect many of these. We do not address the zero-impact failures, such as the failure of a single network cable, because they are explained in depth in the documentation provided by the storage vendor of the various solutions. We discuss the following “common” failure scenarios:

- Single-host failure in Frimley data center
- Single-host isolation in Frimley data center
- Storage partition
- Data center partition
- Disk shelf failure in Frimley data center
- Full storage failure in Frimley data center
- Full compute failure in Frimley data center
- Full compute failure in Frimley data center and full storage failure in Bluefin data center
- Loss of complete Frimley data center

We also examine scenarios in which specific settings are incorrectly configured. These settings determine the availability and recoverability of VMs in a failure scenario. It is important to understand the impact of misconfigurations such as the following:

- Incorrectly configured VM-to-host affinity rules
- Incorrectly configured heartbeat datastores
- Incorrectly configured isolation address
- Incorrectly configured PDL handling
- vCenter Server split-brain scenario

### Single-Host Failure in Frimley Data Center

In this scenario, we describe the complete failure of a host in Frimley data center. This scenario is depicted in Figure 24.

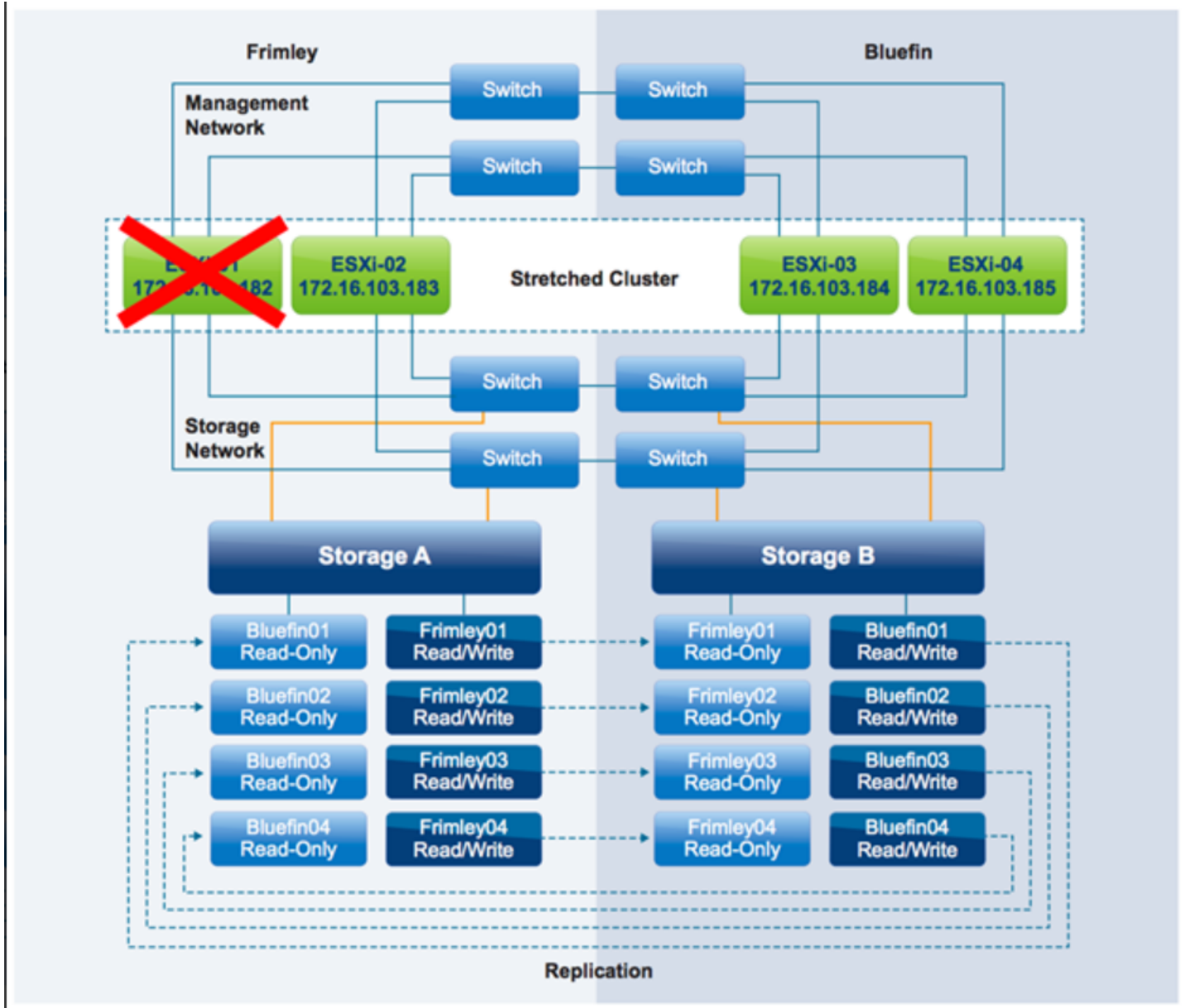


Figure 24 – Single host failure scenario

**Result**

vSphere HA successfully restarted all VMs in accordance with VM-to-host affinity rules.

**Explanation**

If a host fails, the cluster’s vSphere HA primary node detects the failure because it no longer is receiving network heartbeats from the host. Then the primary starts monitoring for datastore heartbeats. Because the host has failed completely, it cannot generate datastore heartbeats; these too are detected as missing by the vSphere HA primary node. During this time, a third availability check—pinging the management addresses of the failed hosts—is conducted. If all of these checks return as unsuccessful, the primary declares the missing host as dead and attempts to restart all the protected VMs that had been running on the host before the primary lost contact with the host.

The vSphere VM-to-host affinity rules defined on a cluster level are “should rules.” vSphere HA VM-to-host affinity rules will be respected so all VMs are restarted within the correct site.

However, if the host elements of the VM-to-host group are temporarily without resources, or if they are unavailable for restarts for any other reason, vSphere HA can disregard the rules and restart the remaining VMs on any of the remaining hosts in the cluster, regardless of location and rules. If this occurs, vSphere DRS attempts to correct any violated affinity rules at the first invocation

and automatically migrates VMs in accordance with their affinity rules to bring VM placement in alignment. VMware recommends manually invoking vSphere DRS after the cause for the failure has been identified and resolved. This ensures that all VMs are placed on hosts in the correct location to avoid possible performance degradation due to misplacement.

### Single-Host Isolation in Frimley Data Center

In this scenario, we describe the response to isolation of a single host in Frimley data center from the rest of the network.

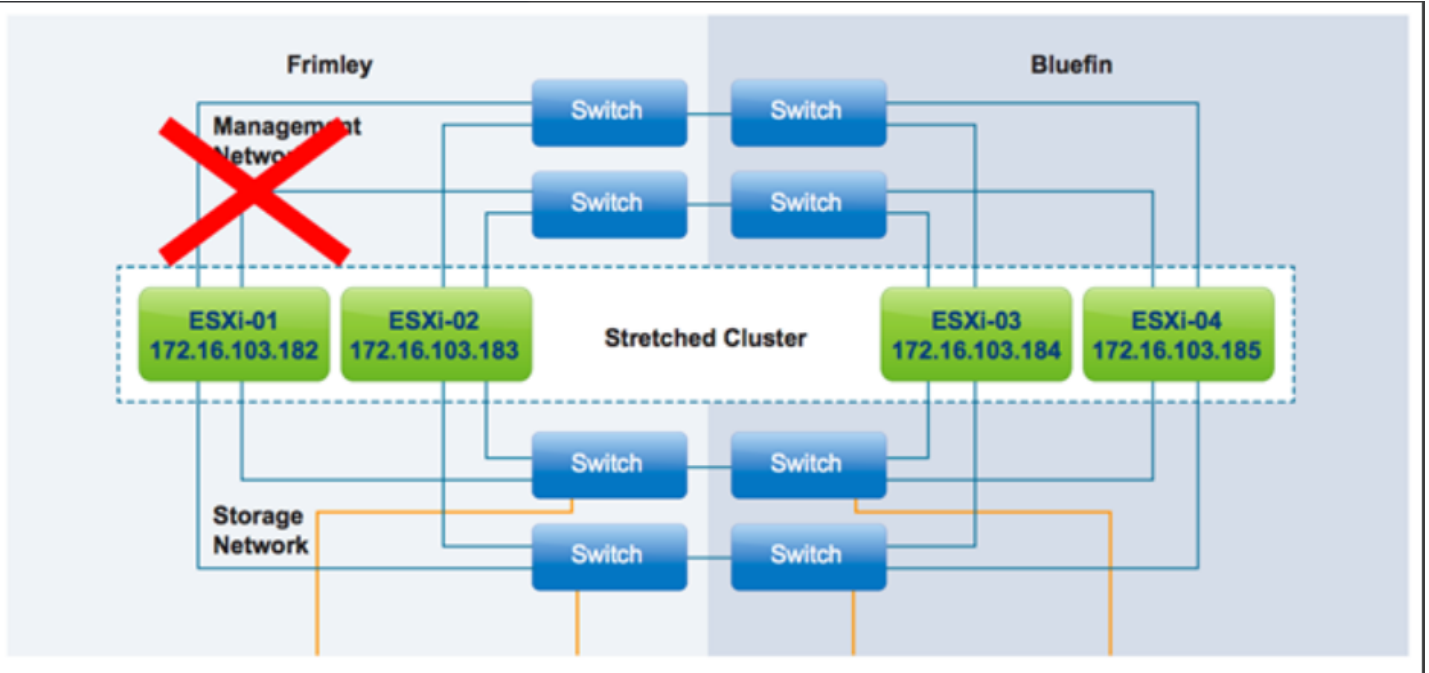


Figure 25 – Single host isolation scenario

#### Result

VMs remain running because isolation response is configured to **leave powered on**.

#### Explanation

When a host is isolated, the vSphere HA primary node detects the isolation because it no longer is receiving network heartbeats from the host. Then the primary starts monitoring for datastore heartbeats. Because the host is isolated, it generates datastore heartbeats for the secondary vSphere HA detection mechanism. Detection of valid host heartbeats enables the vSphere HA primary node to determine that the host is running but is isolated from the network. Depending on the isolation response configured, the impacted host can power off or shut down VMs or can leave them powered on. The isolation response is triggered 30 seconds after the host has detected that it is isolated.

VMware recommends aligning the isolation response to business requirements and physical constraints. From a best practices perspective, **leave powered on** is the recommended isolation response setting for the majority of environments. Isolated hosts are rare in a properly architected environment, given the built-in redundancy of most modern designs. In environments that use network-based storage protocols, such as iSCSI and NFS, and where networks are converged, the recommended isolation response is **power off**. In these environments, it is more likely that a network outage that causes a host to become isolated also affects the host’s ability to communicate to the datastores.

If an isolation response different from the recommended **leave powered on** is selected and **power off** or **shut down** response is triggered, the vSphere HA primary restarts VMs on the remaining nodes in the cluster. The vSphere VM-to-host affinity rules defined on a cluster level are “should rules.” However, because vSphere HA respects VM-to-host affinity rules by all VMs are restarted within the correct site under “normal” circumstances.

#### Storage Partition

In this scenario, a failure has occurred on the storage network between data centers, as is depicted in Figure 26.



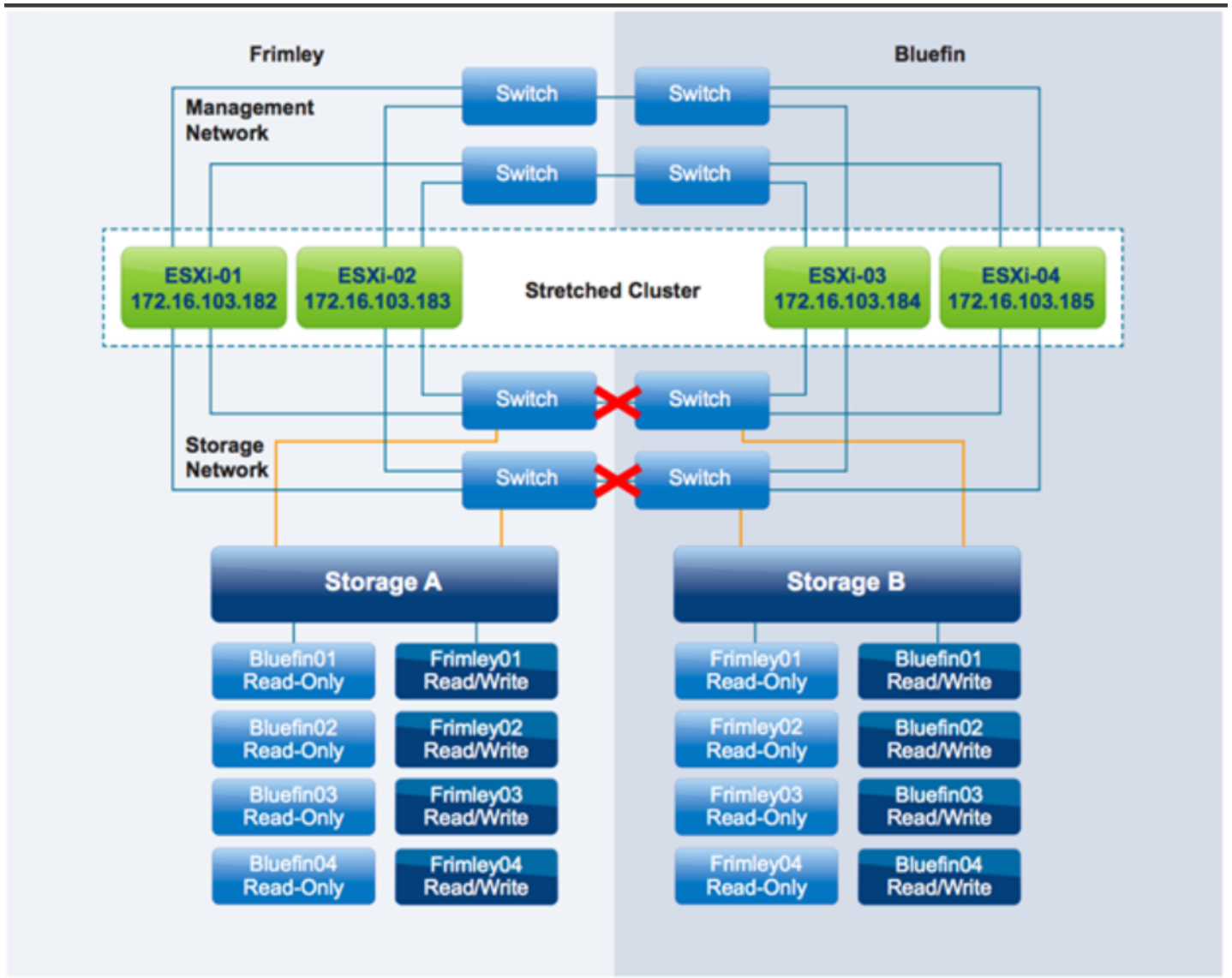


Figure 26 - Storage Partition

**Result**

VMs remain running with no impact.

**Explanation**

Storage site affinity is defined for each LUN, and vSphere DRS rules align with this affinity. Therefore, because storage remains available within the site, no VM is impacted.

If for any reason the affinity rule for a VM has been violated and the VM is running on a host in Frimley data center while its disk resides on a datastore that has an affinity with Bluefin data center, it cannot successfully issue I/O following an inter-site storage partition. This is because the datastore is in an APD condition. In this scenario, the VM can be restarted because vSphere HA is configured to respond to APD conditions. The response occurs after the 3-minute grace period has passed. This 3-minute period starts after the APD timeout of 140 seconds has passed and the APD condition has been declared.

Note that it is possible to define what should happen when the APD is lifted before the 3-minute grace period has passed. One of the options is to reset the VM. If resetting the VM is configured then when the APD is lifted the VM will be reset. Note that any restart ordering will not take effect. Restart Order and dependency only apply to VMs which are restarted.

To avoid unnecessary downtime in an APD scenario, VMware recommends monitoring compliance of vSphere DRS rules. Although vSphere DRS is invoked every 5 minutes, this does not guarantee resolution of all affinity rule violations. Therefore, to prevent unnecessary downtime, rigid monitoring is recommended that enables quick identification of anomalies such as a VM's computes



residing in one site while its storage resides in the other site.

### Data Center Partition

In this scenario, the Frimley data center is isolated from the Bluefin data center, as is depicted in Figure 27.

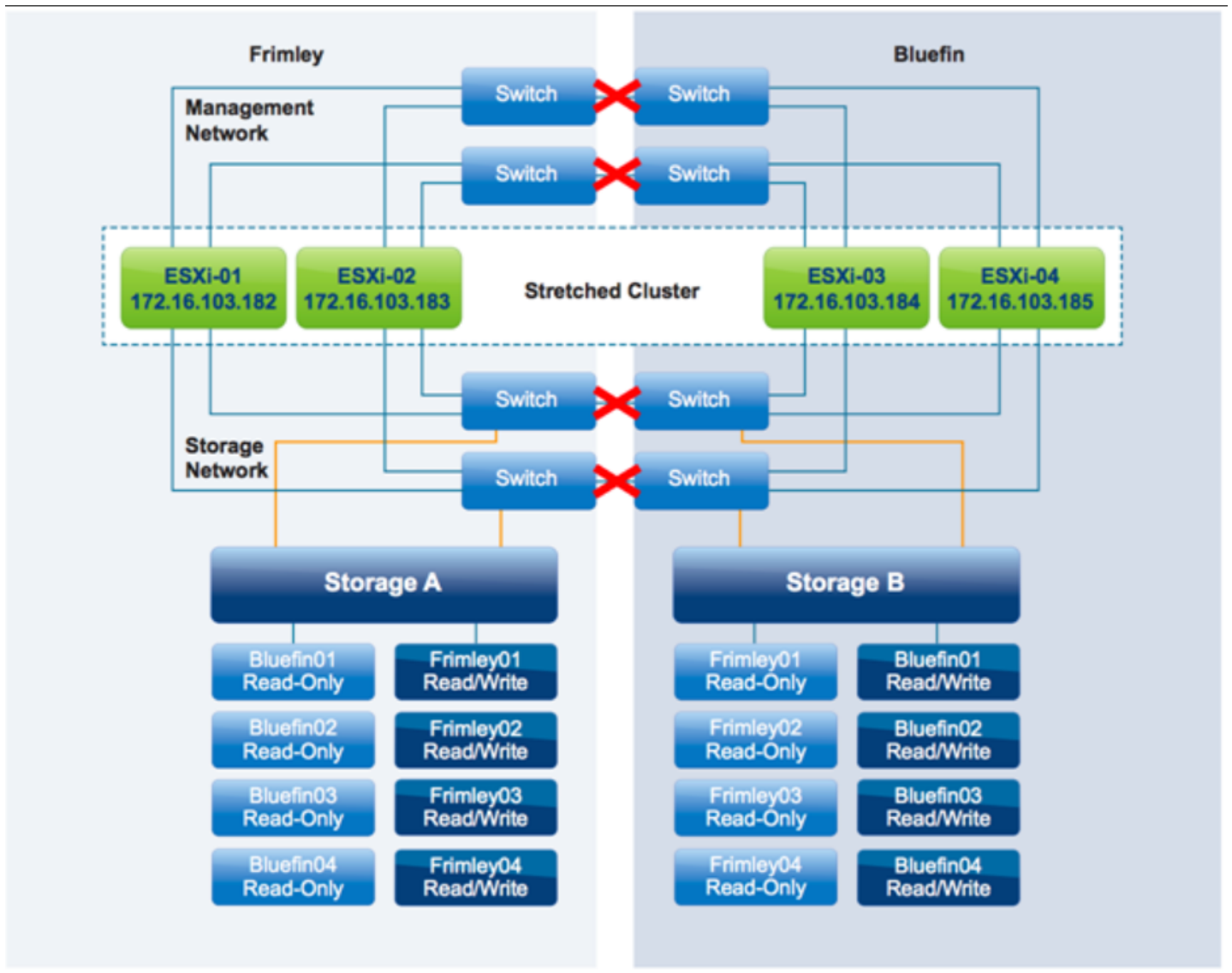


Figure 27 – Datacenter Partition

### Result

VMs remain running with no impact.

### Explanation

In this scenario, the two data centers are fully isolated from each other. This scenario is similar to both the storage partition and the host isolation scenario. VMs are not impacted by this failure because vSphere DRS rules were correctly implemented and no rules were violated.

vSphere HA follows this logical process to determine which VMs require restarting during a cluster partition:

The vSphere HA primary node running in Frimley data center detects that all hosts in Bluefin data center are unreachable. It first detects that no network heartbeats are being received. It then determines whether any storage heartbeats are being generated. This check does not detect storage heartbeats because the storage connection between sites also has failed, and the heartbeat datastores are updated only "locally." Because the VMs with affinity to the remaining hosts are still running, no action is needed for them. Next, vSphere HA determines whether a restart can be attempted. However, the read/write version of the datastores

located in Bluefin data center are not accessible by the hosts in Frimley data center. Therefore, no attempt is made to start the missing VMs.

Similarly, the vSphere hosts in Bluefin data center detect that there is no primary available, and they initiate a primary election process. After the primary has been elected, it tries to determine which VMs had been running before the failure and it attempts to restart them. Because all VMs with affinity to Bluefin data center are still running there, there is no need for a restart. Only the VMs with affinity to Frimley data center are unavailable, and vSphere HA cannot restart them because the datastores on which they are stored have affinity with Frimley data center and are unavailable in Bluefin data center.

If VM-to-host affinity rules have been violated—that is, VMs have been running at a location where their storage is not defined as read/write by default—the behavior changes. The following sequence describes what would happen in that case:

1. The VM with affinity to Frimley data center but residing in Bluefin data center is unable to reach its datastore. This results in the VM's being unable to write to or read from disk.
2. In Frimley data center, this VM is restarted by vSphere HA because the hosts in Frimley data center do not detect the instance's running in Bluefin data center.
3. Because the datastore is available only to Frimley data center, one of the hosts in Frimley data center acquires a lock on the VMDK and is able to power on this VM.
4. This can result in a scenario in which the same VM is powered on and running in both data centers.

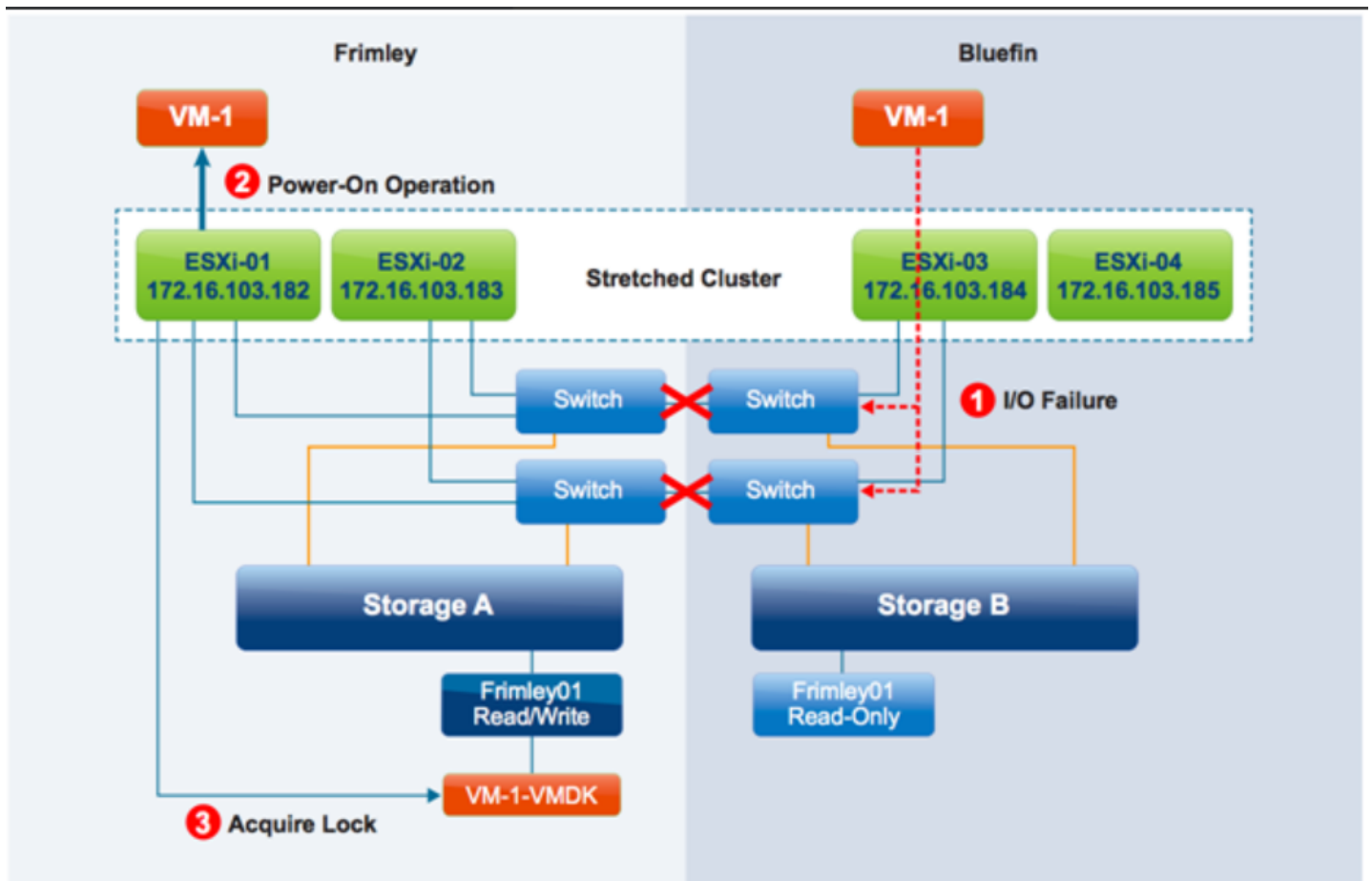


Figure 28 – Ghost VM

5. If the APD response is configured to **Power off and restart VMs (aggressive)**, the VM is powered off after the APD timeout and the grace period have passed. This behavior is new starting vSphere 6.0.

If the APD response is not correctly configured, two VMs will be running, for the following possible reasons:

The network heartbeat from the host that is running this VM is missing because there is no connection to that site.

The datastore heartbeat is missing because there is no connection to that site.

A ping to the management address of the host that is running the VM fails because there is no connection to that site.

The primary located in Frimley data center detects that the VM had been powered on before the failure. Because it is unable to communicate with the VM’s host in Bluefin data center after the failure, it attempts to restart the VM because it cannot detect the actual state.

If the connection between sites is restored, a classic “VM split-brain scenario” will exist. For a short period of time, two copies of the VM will be active on the network, with both having the same MAC address. Only one copy, however, will have access to the VM files, and vSphere HA will detect this. As soon as this is detected, all processes belonging to the VM copy that has no access to the VM files will be halted, as is depicted in Figure 29.

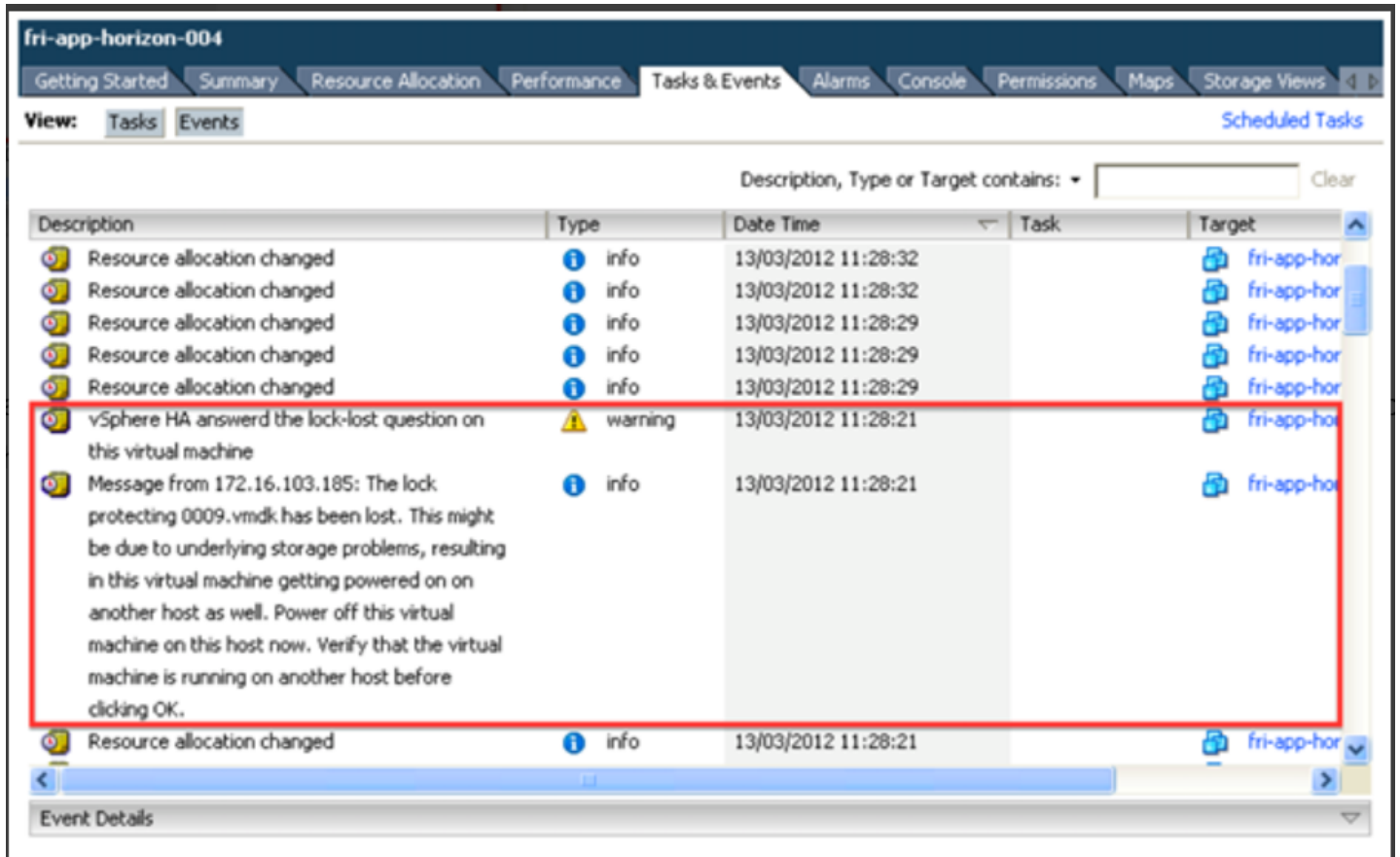


Figure 29 – Lock lost

In this example, the downtime equates to a VM’s having to be restarted. Proper maintenance of site affinity can prevent this. To avoid unnecessary downtime, VMware recommends close monitoring to ensure that vSphere HA and DRS cluster rules align with datastore site affinity.

### Disk Shelf Failure in Frimley Data Center

In this scenario, one of the disk shelves in Frimley data center has failed. Both Frimley01 and Frimley02 on storage A are impacted.

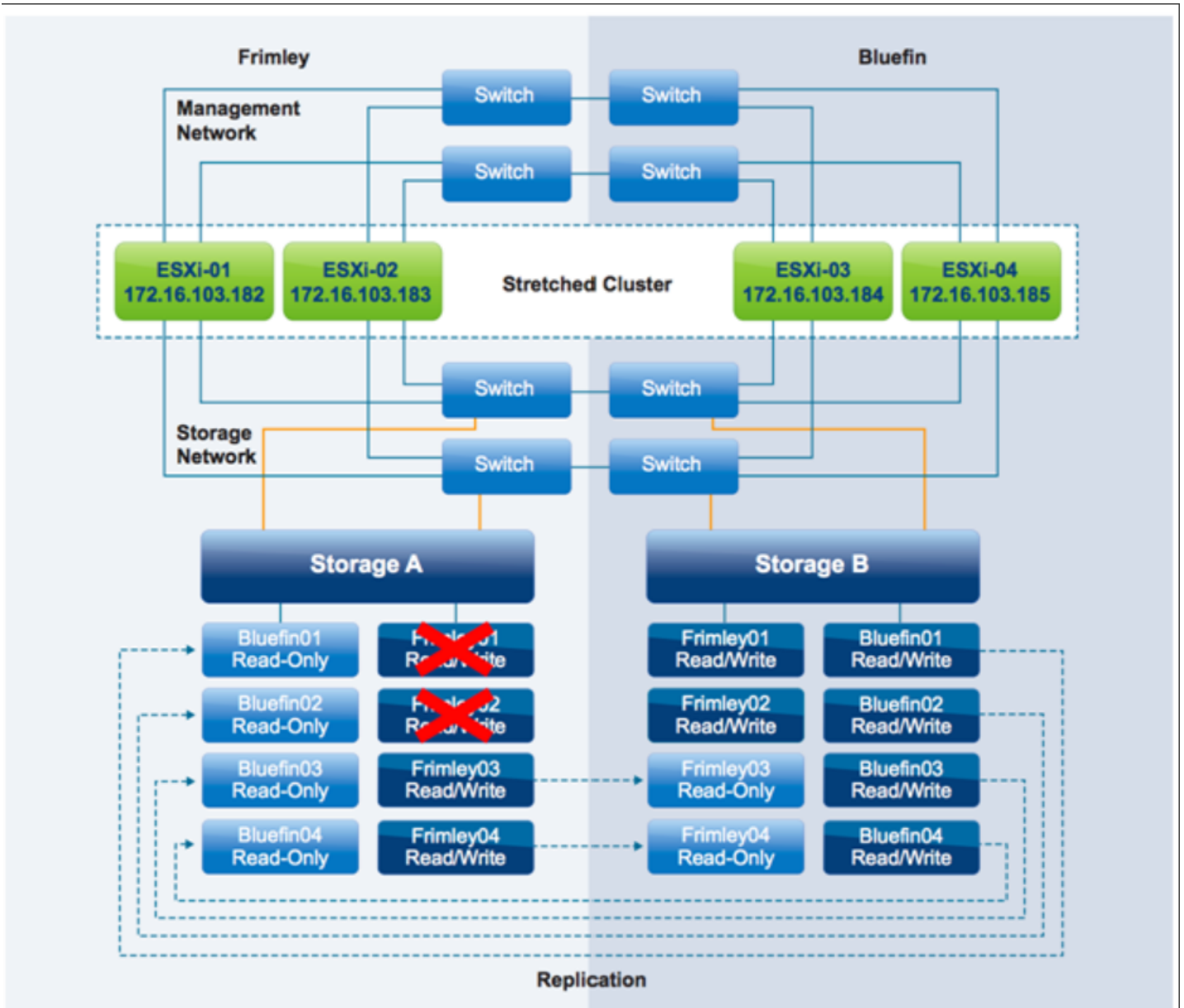


Figure 30 - Disk Shelf Failure

**Result**

VMs remain running with no impact.

**Explanation**

In this scenario, only a disk shelf in Frimley data center has failed. The storage processor has detected the failure and has instantly switched from the primary disk shelf in Frimley data center to the mirror copy in Bluefin data center. There is no noticeable impact to any of the VMs except for a typical short spike in I/O response time. The storage solution fully detects and handles this scenario. There is no need for a rescan of the datastores or the HBAs because the switchover is seamless and the LUNs are identical from the vSphere host perspective.

**Full Storage Failure in Frimley Data Center**

In this scenario, a full storage system failure has occurred in Frimley data center.

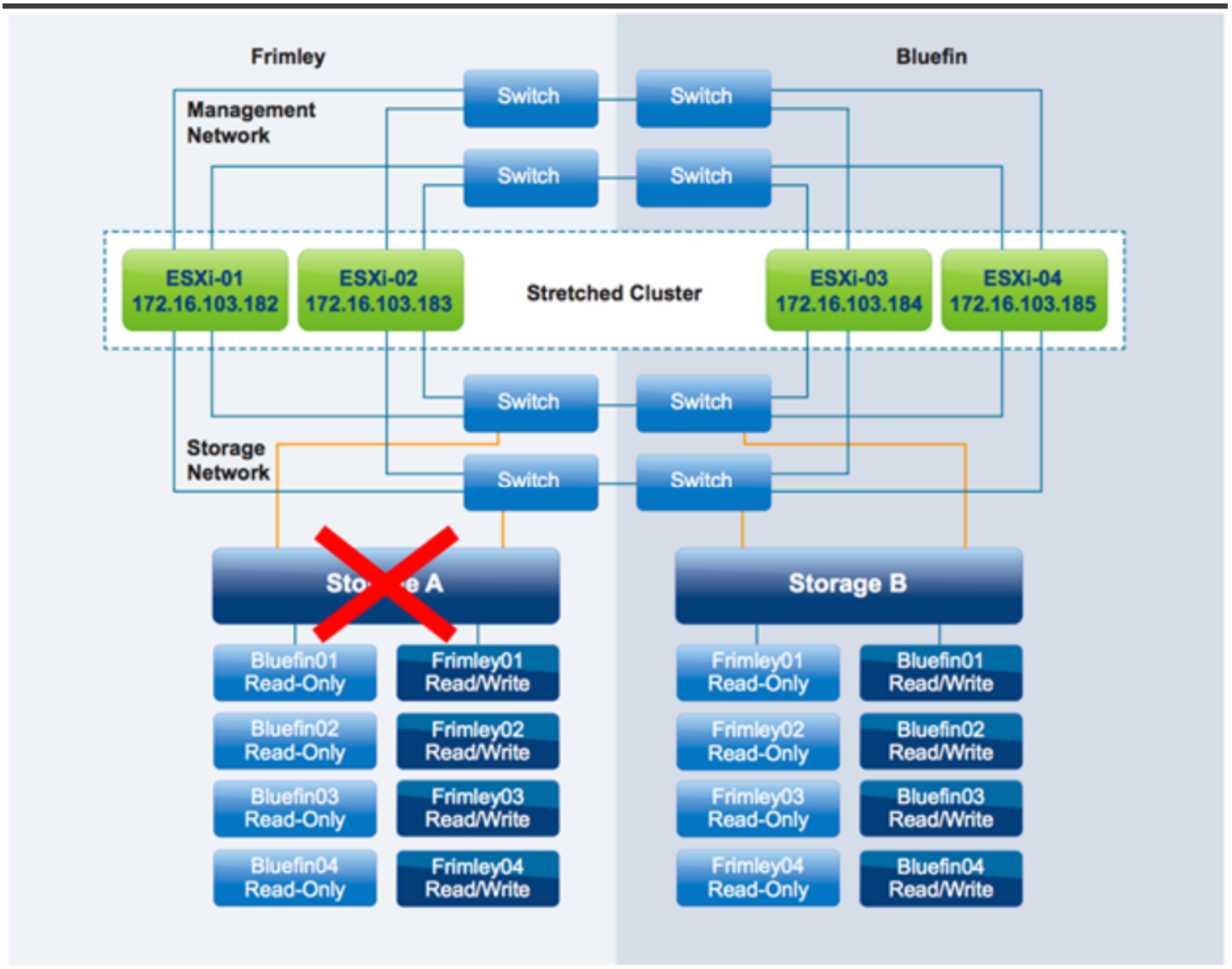


Figure 31 - Full storage failure

**Result**

VMs remain running with no impact.

**Explanation**

When the full storage system fails in Frimley data center, a **take over** command must be initiated manually. As described previously, we used a NetApp MetroCluster configuration to describe this behavior. This take over command is particular to NetApp environments; depending on the implemented storage system, the required procedure can differ. After the command has been initiated, the mirrored, read-only copy of each of the failed datastores is set to read/write and is instantly accessible. We have described this process at an extremely high level. For more details, refer to the storage vendor’s documentation.

From the VM perspective, this failover is seamless: The storage controllers handle this, and no action is required from either the vSphere or storage administrator. All I/O now pass across the intra-site connection to the other data center because VMs remain running in Frimley data center while their datastores are accessible only in Bluefin data center.

vSphere HA does not detect this type of failure. Although the datastore heartbeat might be lost briefly, vSphere HA does not take action because the vSphere HA primary agent checks for the datastore heartbeat only when the network heartbeat is not received for 3 seconds. Because the network heartbeat remains available throughout the storage failure, vSphere HA is not required to initiate any restarts.



### Permanent Device Loss

In the scenario shown in Figure 30, a permanent device loss (PDL) condition occurs because datastore Frimley01 has been taken offline for ESXi-01 and ESXi-02. PDL scenarios are uncommon in uniform configurations and are more likely to occur in a non-uniform vMSC configuration. However, a PDL scenario can, for instance, occur when the configuration of a storage group changes as in the case of this described scenario.

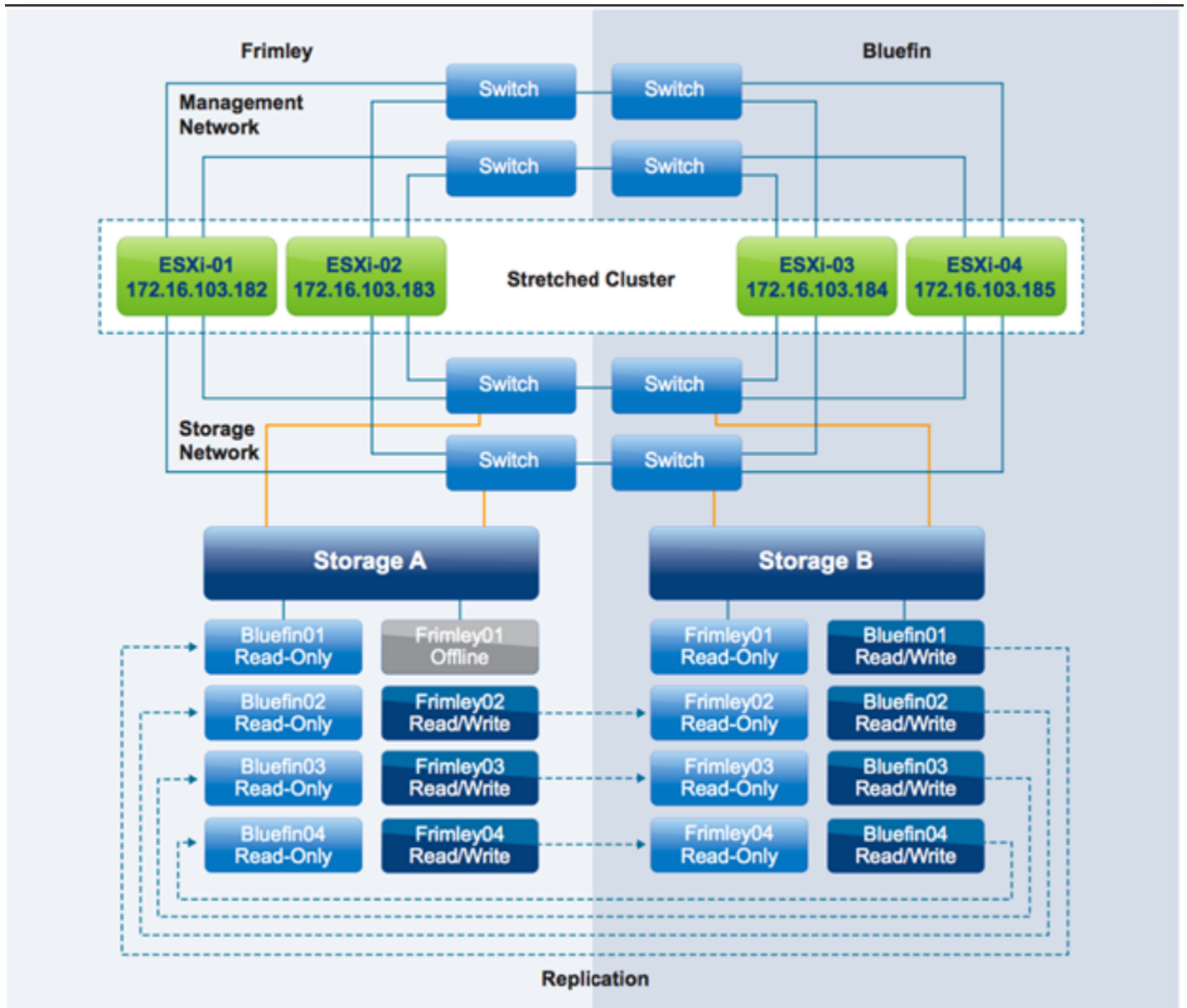


Figure 32 - Permanent Device Loss Scenario

### Explanation

When the PDL condition occurs, VMs running on datastore Frimley01 on hosts ESXi-01 and ESXi-02 are halted instantly. They then are restarted by vSphere HA on hosts within the cluster that have access to the datastore, ESXi-03, and ESXi-04 in this scenario. The PDL and halting of the VM world group leader can be witnessed by following the entries in the vmkernel.log file located in /var/log/ on the vSphere hosts. The following is an outtake of the vmkernel.log file where a PDL is recognized and appropriate action is taken.

```
2017-03-14T13:39:25.085Z cpu7:4499)WARNING: VSCSI: 4055: handle 8198(vscsi4:0):opened by wid 4499 (vmm0:fri-iscsi-02) has Permanent Device Loss. HALTING world group leader 4491
```

VMware recommends configuring **Response for Datastore with Permanent Device Loss (PDL)** to **Power off and restart**

**VMs** . This setting ensures that appropriate action is taken when a PDL condition exists. The correct configuration is shown in Figure 33.

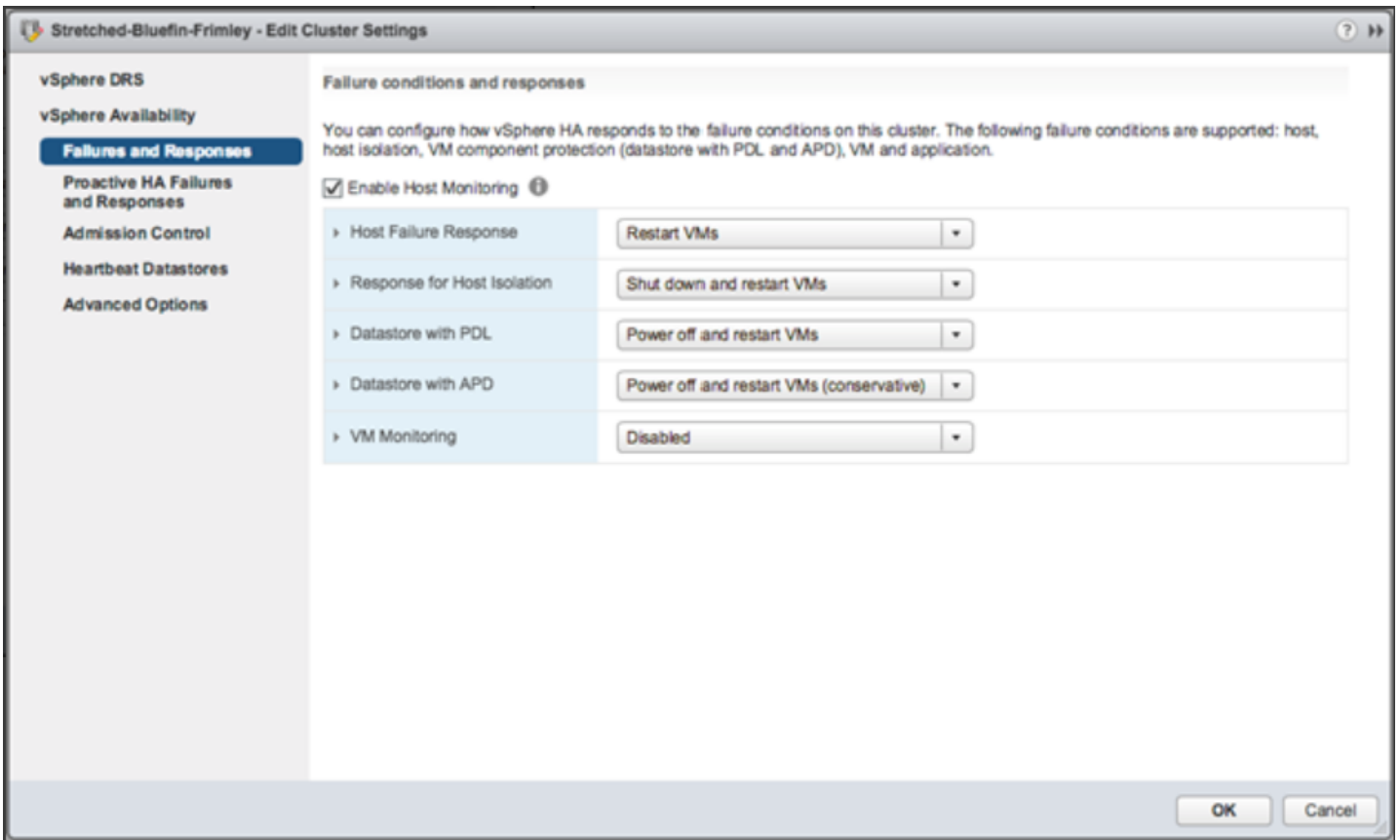


Figure 33 - VCMF - PDL Configuration

### Full Compute Failure in Frimley Data Center

In this scenario, a full compute failure has occurred in Frimley data center.

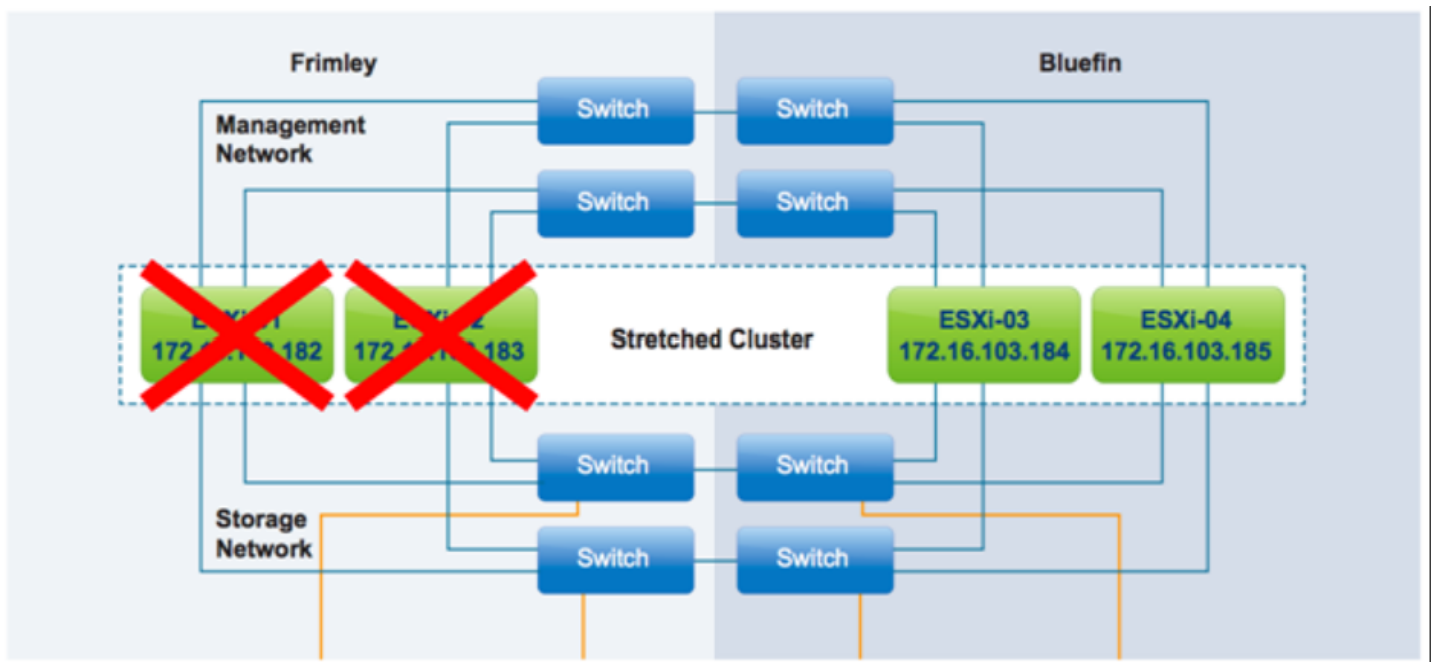


Figure 34 - Full compute failure

### Result

All VMs are successfully restarted in Bluefin data center.

### Explanation

The vSphere HA primary was located in Frimley data center at the time of the full compute failure at that location. After the hosts in Bluefin data center detected that no network heartbeats had been received, an election process was started. Within approximately 20 seconds, a new vSphere HA primary was elected from the remaining hosts. Then the new primary determined which hosts had failed and which VMs had been impacted by this failure. Because all hosts at the other site had failed and all VMs residing on them had been impacted, vSphere HA initiated the restart of all of these VMs. vSphere HA first schedules the restart, which only can only succeed when sufficient unreserved resources are available. In order to ensure this we had vSphere HA admission control enabled and set to reserve 50% (2 host failures) of CPU and memory capacity.

vSphere HA can initiate 32 concurrent restarts on a single host, providing a low restart latency for most environments. As described in the vSphere HA section of this paper, there is the ability to sequence the order of restart for VMs leveraging the **VM Overrides** feature (there are 5 options: lowest, low, medium, high, highest). This policy must be set on a per-VM basis. These policies were determined to have been adhered to; highest-priority VMs started first, followed by high-, medium-, low- and lowest-priority VMs.

As part of the test, the hosts at the Frimley data center were again powered on. As soon as vSphere DRS detected that these hosts were available, a vSphere DRS run was invoked. Because the initial vSphere DRS run corrects only the vSphere DRS affinity rule violations, resource imbalance was not corrected until the next full invocation of vSphere DRS. vSphere DRS is invoked by default every 5 minutes or when VMs are powered off or on through the use of vSphere Web Client.

### Loss of Frimley Data Center

In this scenario, a full failure of Frimley data center is simulated.



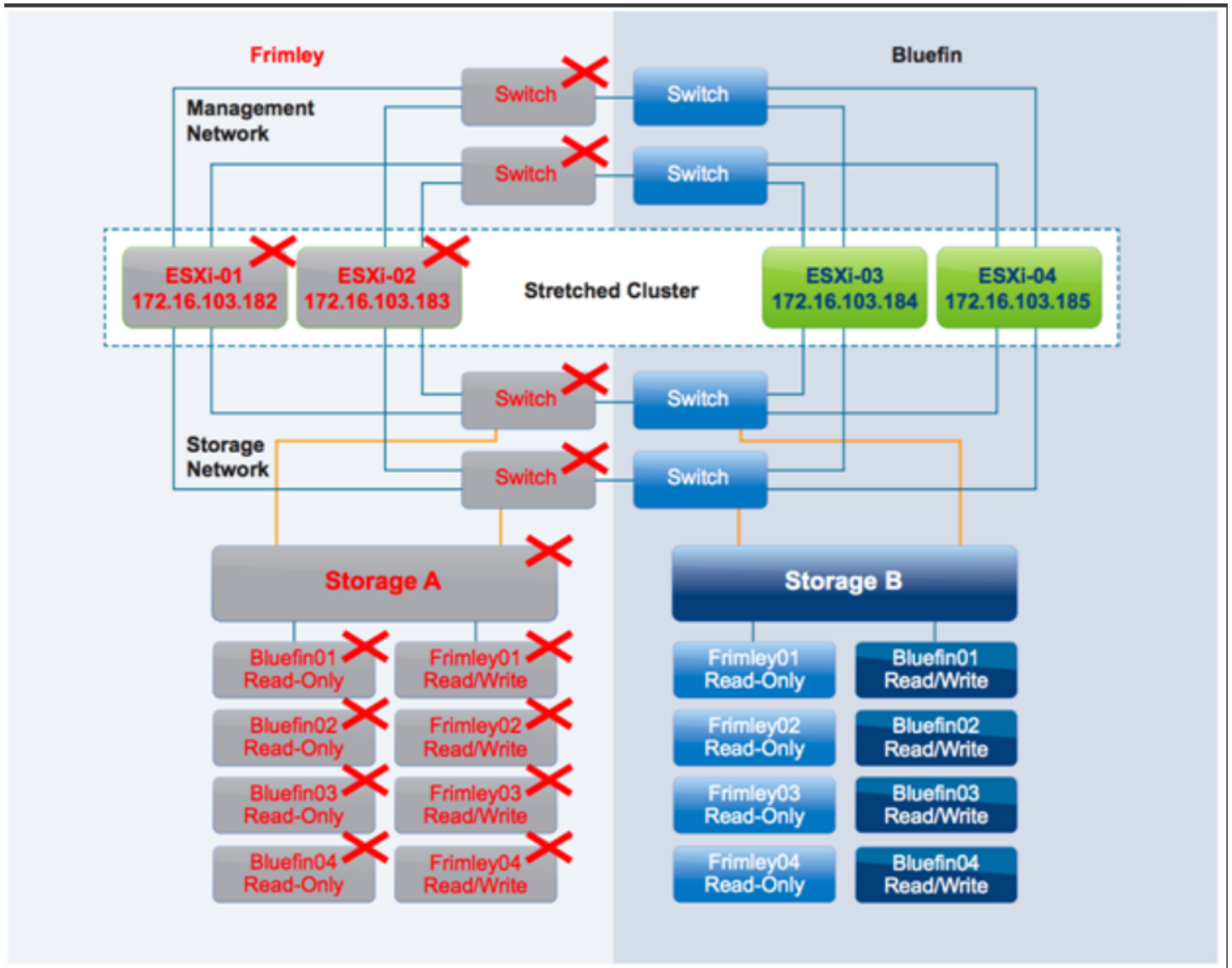


Figure 35 - Full datacenter failure

**Result**

All VMs were successfully restarted in Bluefin data center.

**Explanation**

In this scenario, the hosts in Bluefin data center lost contact with the vSphere HA primary and elected a new vSphere HA primary. Because the storage system had failed, a **take over** command had to be initiated on the surviving site, again due to the NetApp-specific process. After the **take over** command had been initiated, the new vSphere HA primary accessed the per-datastore files that vSphere HA uses to record the set of protected VMs. The vSphere HA primary then attempted to restart the VMs that were not running on the surviving hosts in Bluefin data center. In our scenario, all VMs were restarted within 2 minutes after failure and were fully accessible and functional again.

*NOTE: By default, vSphere HA stops attempting to start a VM after 30 minutes. If the storage team does not issue a takeover command within that time frame, the vSphere administrator must manually start up VMs after the storage becomes available.*

## Summary and Acknowledgement

When properly operated and architected, stretched clusters are an excellent solution to increase resiliency and offer inter-site workload mobility. There has always been, however, confusion regarding failure scenarios and the various types of responses from both the vSphere layer and the storage layer. In this white paper, we have tried to explain how vSphere HA and vSphere DRS respond to certain failures in a stretched cluster environment and to offer recommendations for configuration of a vSphere cluster in this type of environment. This paper highlights the importance of site affinity, the role played by vSphere HA and DRS Cluster rules and groups, how vSphere HA interacts with those rules and groups, and how users must ensure that the logic enforced by those rules and groups is maintained over time to provide the reliability and predictability of the cluster.

