WHITE PAPER – **NOVEMBER 2016**

PLANNING HIGHLY AVAILABLE, MISSION CRITICAL SQL SERVER DEPLOYMENTS WITH VMWARE vSPHERE

Allan Hirt
Microsoft Cloud and Datacenter Management
MVP and Data Platform MVP
Managing Partner, SQLHA, LLC

vmware® | SQLHA

Table of Contents

1. Introduction

We live in a data-driven world where businesses and end users consume and generate information in increasingly numerous ways. Customers and businesses today increasingly expect around the clock access to this data which is stored somewhere, and one of the most popular options is Microsoft SQL Server.

Ensuring that SQL Server is highly available is one core tenet of mission critical, along with other concepts like performance and scalability. Deploying SQL Server properly will ensure that it achieves the availability, scalability, and reliability that required to be mission critical. Striking the balance between utilization, availability, performance, and manageability is not an easy task, but achievable if planned correctly. Virtualizing SQL Server under VMware vSphere® can help you achieve that goal even though the standard in many companies is to still use physical servers. Why virtualize SQL Server deployments?

While physical hardware is a tried-and-true method of deploying SQL Server implementations, there can be complications that are part of the deployment process, such as the wait time between ordering and receiving the hardware, and then the actual IT work that needed to get done including tasks like racking and stacking in the data center. It could be weeks, or even months between the time a system is ordered to when it is ready for end users. The business as a result is often less agile because of this process. In today's modern world, being less agile can be the difference between success and failure.

Virtualiztion is one of the best methods of achieving that agility. While virtualization has been around for quite some time, there are many data professionals who are unsure if virtualization is the right way to deploy a database server due to years of fear, uncertainty, and doubt based on misconceptions from a time when things were different years ago such as :

- It was less than 10 years ago that the computing world was still largely 32-bit, limiting scalability and density
- Virtual machines (VMs) at that time were limited in the amount of memory and number of processors, which led to bad experiences with virtualizing database workloads
- Virtualization was still maturing as a deployment method, so the tools and processes were not as mature as they are today

Today, those challenges are gone. 64-bit is the standard for server hardware and operating systems. As of SQL Server 2016, Microsoft no longer ships a 32-bit server-based version of SQL Server. Physical servers today have large computing capacity with many cores per physical processor and support for large amounts of memory. Only select applications can push modern hardware to its limits, so finding ways to better utilize that computing power while at the same time increasing agility, performance, and availability but reducing cost is the hallmark of virtualization. As of vSphere 6, the following list shows some of the selected maximums:

- 128 virtual CPUs per VM
- Just under 4TB of memory (4080 GB) per VM
- 480 logical CPUs per hypervisor host

- Up to 12TB per hypervisor host
- 64 hosts per VMware cluster
- 8,000 VMs per VMware cluster

For the complete list of maximums, refer to the document for the version of vSphere you have deployed entitled "Configuration Maximums".

In context of availability, virtualizing also provides other options that could potentially enhance the availability options already supported by both Windows Server and SQL Server. Whether looking to architect a simple availability solution in a single data center, or true business continuity with adding a disaster recovery aspect to the architecture to be able to come online in another site, the availability must meet your needs based on actual requirements.

Whether you are a SQL Server DBA, VMware administrator, architect, or IT decision maker, this paper will help you understand how to architect highly available, mission critical SQL Server solutions utilizing vSphere. Performance and agility aspects as they relate to availability will be presented, but this is not a dedicated performance paper. There is much more in terms of nuances at the Windows Server and SQL Server layers that you will need to know as well; this paper is not meant to be exhaustive, but cover the most important aspects and provide the specific practices as they relate to vSphere.

2. Supportability and Lifecycle Management

Achieving availability is more than choosing and enabling one or more features at the hypervisor or in the guest VM. When deploying mission critical systems that require the reliability and availability to meet business demand, it is imperative that they are supported end-to-end from all vendors. VMware makes this process easier by publishing clear guidelines via their Compatibility Guides (http://www.vmware.com/guides.html). The Compatibility Guide allows you to search for a specific combination of VMware products, releases, and more to determine if your proposed solution is proper. As an example, if you were looking for Cisco blades certified to work with 6.0.1, it would look like Figure 1 for the query.

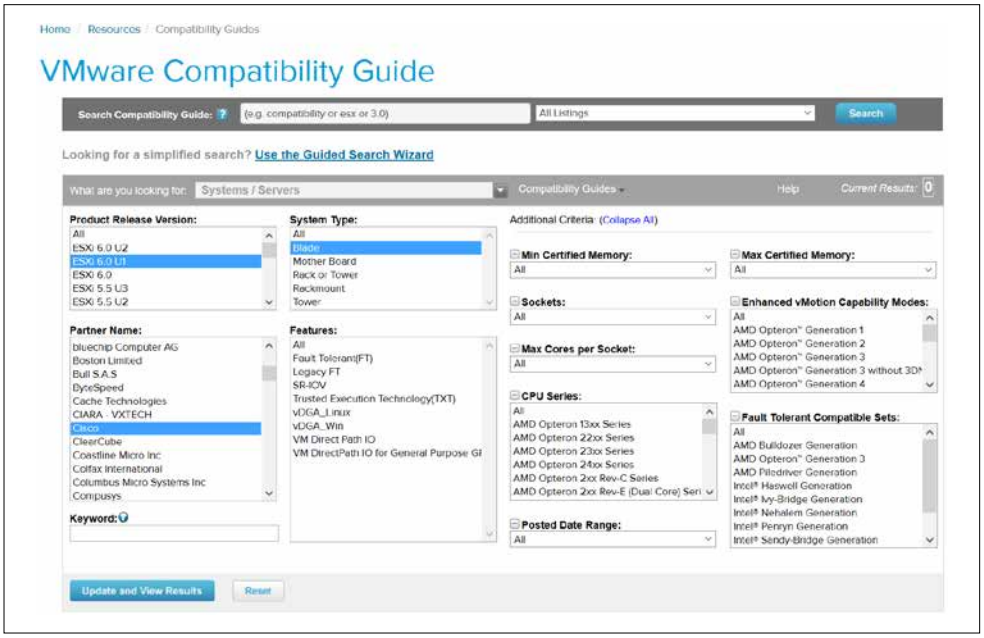

Figure 1: Querying the VMware Compatibility Guide

The results would look like Figure 2.

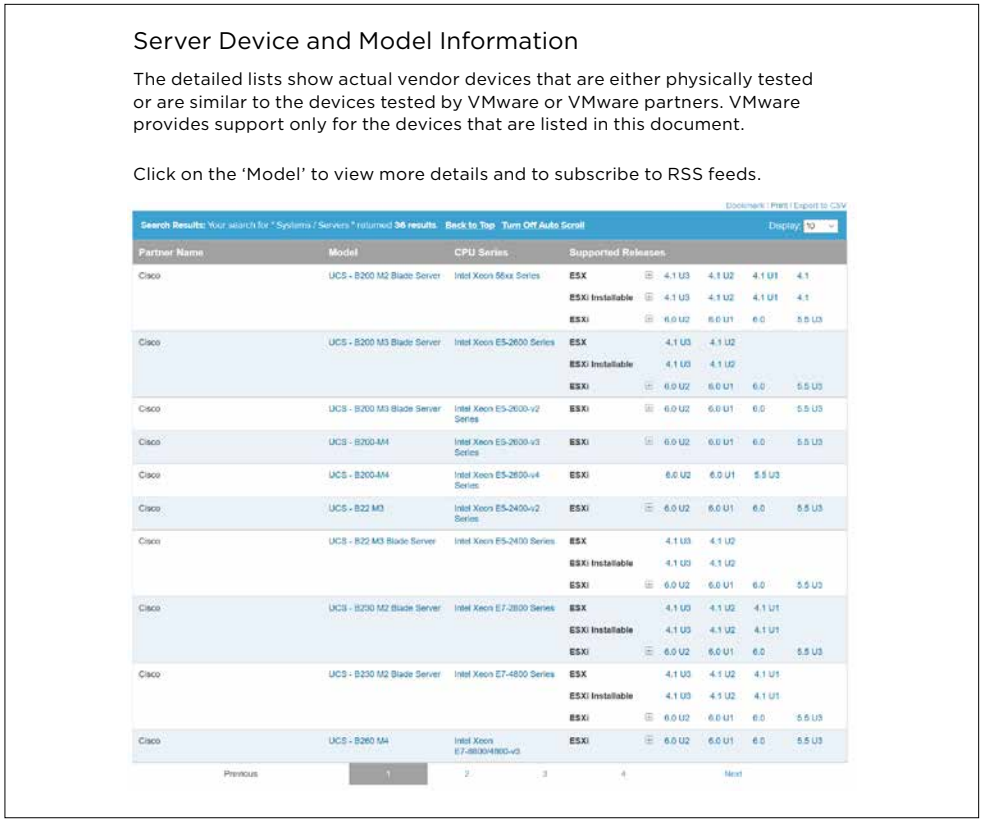

Figure 2: Results of the Compatibility Guide query

The supported versions of Windows Server are also covered by the Compatibility Guide. Under the Compatibility Guide link above, use Guest OS for the value in the What are you looking for dropdown. Note that if you are going to use the OS Family Name category to search, Windows Server 2008 R2 would be found under Windows Server 2008 and similarly, Windows Server 2012 R2 would be associated with Windows Server 2012. To see all versions of Windows Server supported for a specific version of ESXi, choose your version of ESXi in Product Release Version, select Windows for OS Family, and finally an OS Type of Server. Figure 3 shows the result of searching for all ESXi releases that support Windows Server 2016.

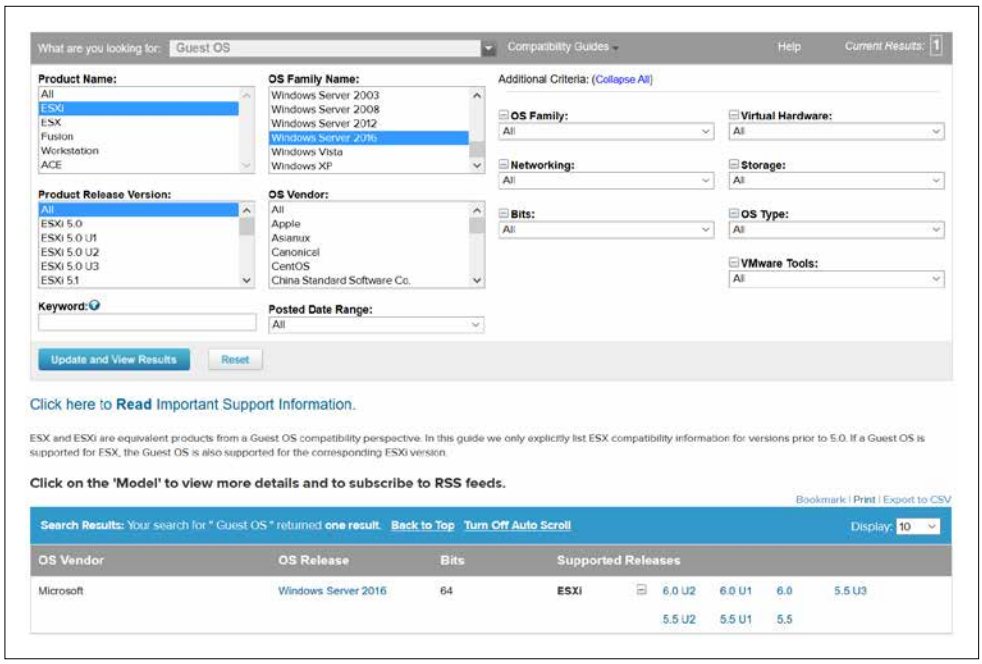

Figure 3: Querying to find which versions of vSphere support Windows Server 2016

SQL Server must be supported by the target Windows Server OS. Table 1 shows the major version of SQL Server and the versions of Windows Server that support them starting with SQL Server 2008 and Windows Server 2008.

	WINDOWS SERVER VERSION				
	2008	**2008 R2**	**2012**	**2012 R2**	**2016**
SQL Server 2008	Yes	Yes	Yes	Yes	No
SQL Server 2008 R2	Yes	Yes	Yes	Yes	No
SQL Server 2012	Yes	Yes	Yes	Yes	Yes
SQL Server 2014	Yes	Yes	Yes	Yes	Yes
SQL Server 2016	No	No	Yes	Yes	Yes

Table 1: SQL Server to Windows Server support matrix

To determine the minimum version of SQL Server required by Windows Server 2012 R2 or later, consult the Microsoft Knowledge Base article "Using SQL Server in Windows 8 and later versions of Windows operating system" (Microsoft KB 2681562). For example, if the desired combination is SQL Server 2012 and Windows Server 2012 R2, SQL Server 2012 Service Pack 1 or later must be applied as that is the minimum version of SQL Server that is supported by Microsoft in that combination.

Platform choices are ones that you live with for a number of years, and the goal is that every component should be supported end-to-end for its entire lifecycle. Microsoft publishes lifecycle guidelines for all of its products including SQL Server and Windows Server at http://support.microsoft.com/lifecycle. As of the writing of this paper, Windows Server versions older than Windows Server 2012 are out of mainstream support, meaning the minimum Windows Server version you should consider deploying under VMware is Windows Server 2012. However, Windows Server 2012 is scheduled to go out of mainstream support in less than 18 months from the publication of this paper, so the reality is you should be looking at Windows Server 2012 R2 with an eye towards Windows Server 2016. Windows Server 2012 R2 also has quite a few enhancements for clustered implementations of SQL Server, and would be the minimum recommended version if not deploying Windows Server 2016.

Based on Figure 3, that means you may also need a later version of VMware ESXi™ as well if your current version is older than the minimum one supported by Windows Server 2012 R2 or Windows Server 2016. This is why keeping your infrastructure mission critical is a coordinated effort: all pieces of the puzzle have to come together.

Understand that third party application vendors may dictate what version of SQL Server (including patch level) and/or Windows Server that can be deployed. Keep that in mind as you look at what Microsoft will be supporting as you do not want to find yourself out of support with Microsoft and possibly VMware, but in support with the application vendor. If possible, all of your versions will be currently supported by each vendor.

You should always strive to deploy platforms that are in Mainstream Support from Microsoft and have plans to migrate to newer versions as they age into Extended Support. This is a critical aspect of availability as well as security, because if the product is no longer being updated, you may encounter functional or compatibility issues, and possibly violate internal rules around security since the platform may no longer receive security updates beyond the extended support timeframe. The biggest long term challenge is to find and retain administrators who have the knowledge and desire to manage older platforms beyond their expiration date.

Last, but certainly not least, you should follow the guidelines in the Microsoft Support Knowledge Base Article "Support policy for Microsoft SQL Server products that are running in a hardware virtualization environment" (Microsoft KB 956893).

The cost of migrating and upgrading to newer versions of Windows Server, SQL Server, and all of VMware's products every few years will often be offset by the enhancements that could benefit your environment. You should have a planned obsolecense for versions and a scheduled refresh of hardware.

3. Planning for SQL Server Availability Features Under vSphere

This section will provide an overview and understanding of the availability features that would be deployed in a guest VM and how to approach deploying them from a VMware perspective.

3.1 SQL Server Protection Levels

There are three levels of protection that the built-in SQL Server availability features can provide: instance, database, and data.

An installation of SQL Server is known as an instance. Therefore, instance-level protection covers the entire installation of SQL Server. Up to 25 instances are supported on a single WSFC a single clustered configuration, and 50 for standalone – but that does not mean you should deploy the maximum number of instances on a WSFC or a single server. Deploy as many as needed for the business taking into account manageability, performance, and availability.An instance of SQL Server is the equivalent of a database in Oracle.

There are two types of instances: default and named. A default instance on a standalone server has the same name as the underlying Windows Server. For example, if a server's name is ALLAN, the name ALLAN is used by applications or end users to access the SQL Server instance. A standalone or clustered configuration can only have one default instance. A clustered instance, also known as an failover cluster instance, works slightly differently, and that will be discussed in Section 2.1.2.2.

A named instance adds a unique identifier when accessing it, otherwise if you had more than one instance, you would not be able to get to the data. This unique name is recognized by SQL Server only. It is not registered anywhere else in your infrastructure. Both clustered and standalone solutions can have all named instances, or one default instance and the rest named if there is more than one installation on that standalone server or cluster.

If the underlying server is named ALLAN and there are two named instances installed – HIRT and BOSTON – an application would access them via name using ALLAN\HIRT and ALLAN\BOSTON. When there are multiple instances installed, each gets its own set of binaries, but there are also shared files which are common to all instances. As with a default instance, FCIs have a slightly different behavior with named instances that will be discussed in Section 3.2.2.1.

Database-level protection ensures that specially configured copy of a database can be brought online in another SQL Server instance on a separate server (physical, virtual, or cloud-based). Bringing one or more databases online is just part of the process; there are objects in the instance that will be required to also be configured on that other instance for the application to work properly after switching. The process for this is not part of database-level protection, and DBAs must handle this manually. A database in SQL Server is the equivalent of a schema in Oracle.

Data-level protection ensures that bussines critical data also exists in a database on another instance of SQL Server and is accessible. Similar to database-level protection, there are other objects in SQL Server that may be required to use the data, and features that fall in this category generally do not make good candidates for making an exact replica of everything in the primary database since it was not designed for that type of protection.

The three levels of protection map to the availability features provided by SQL Server. Table 2 shows the level to feature mapping.

PROTECTION LEVEL	SQL SERVER AVAILABILITY FEATURE
Instance	Always On Failover Cluster Instance
Database	Always On Availability Group (SQL Server 2012 and later) Database Mirroring (deprecated in SQL Server 2012) Log Shipping
Data	Replication (SQL Server's built-in feature, not hardware or non-SQL Server software-based)

Table 2: Protection mapping to SQL Server availability features

This availability section will focus primarily on Always On failover cluster instances (FCIs) and availability groups (AGs) as they are the most widely used availability features in SQL Server. Because both of these features require that Windows Server is clustered underneath them, there are some additional considerations that you must take into account when deploying a clustered configuration of Windows Server and SQL Server underneath You cannot deploy an availability group or clustered instance of SQL Server without clustering Windows Server first as a Windows Server Failover Cluster (WSFC).

Inside the VM, clustering is a stack. This can be seen in Figure 4.

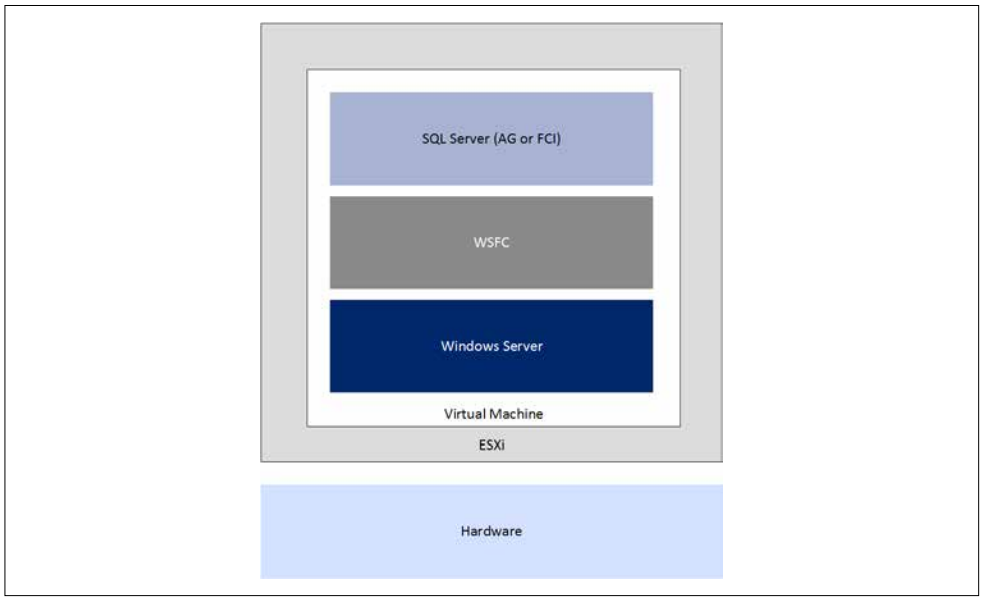

Figure 4: Clustering stack for SQL Server as it relates to vSphere

The other availability features – database mirroring, log shipping, and replication – will only be discussed briefly. Database mirroring (DBM) was deprecated in SQL Server 2012. Underneath the covers, how it works is similar to (but not exactly the same as) an availability group because some of the underpinning is similar, such as the way that the log records are sent to the mirror. Log shipping is based on SQL Server's native backup and restore, so there is not much special that needs to be done from a virtualization standpoint. Many of the generic best practices that will be discussed in this paper could apply, but log shipping itself does not complicate a deployment. SQL Server's replication feature has a few different variants (snapshot, merge, and transactional), but is not designed to provide availability like the database-level features. These three SQL Server availability features will not be discussed further in this paper.

Always On is Microsoft's formal marketing umbrella which encompasses both the availability groups feature as well as clustered instances of SQL Server; it is not the AG feature. Prior to late 2015, the AlwaysOn marketing moniker did not include a space. Since a clustered installation of SQL Server can either be an FCI or an AG, using something like SQL Clustering would be incorrect, and using AAG, AOAG, or another variant for AG is also not correct. With the SQL Server availability features, using the right names helps keep everyone on the same page. If you see incorrect names for these features, refer back to this paper to know how it maps properly to how Microsoft refers to them.

3.2 Planning Clustered Implementations of SQL Server under vSphere

A base WSFC only provides availability. It does not provide scale out or load balancing capabilities. Workloads running on top of a WSFC may add these as part of what they provide, but the general rule of thumb for deploying SQL Server is that you must properly size and scale up an AG or FCI. A server or VM participating in a WSFC is known as a node. It is a general best practice to ensure that each node participating in a clustered SQL Server has the same configuration.

Always refer to VMware Knowledge Base article "Microsoft Clustering on VMware vSphere: Guidelines for supported configurations" (1037959) for the latest information on supported configurations of Microsoft clustered configurations under vSphere.

3.2.1 Deploying a Windows Server Failover Cluster

This section will help you plan a base WSFC configuration for an AG or FCI under vSphere.

3.2.1.1.Windows Server Edition, Number of Nodes, and Supportability

Windows Server 2008 R2 and earlier requires Enterprise or Datacenter Editions to be able to create a WSFC. Windows Server 2012 and later can use either Standard or Datacenter Editions.

Windows Server 2012 and later can support up to 64 servers, known as nodes, in one WSFC. ESXi supports up to five VMs in a single FCI's configuration. For AGs using only standalone instances of SQL Server, vSphere supports the maximum number of servers/VMs that can participate in an AG (five for SQL Server 2012, nine for SQL Server 2014 and later).

Current documentation (including the VMware KB 1037959 "Microsoft Clustering on VMware vSphere: Guidelines for supported configurations") treats FCIs and AGs as mutually exclusive and could potentially be interpreted in different ways for the total number of nodes. For a configuration with shared storage, meaning an FCI, it is five nodes. That does not limit the total nodes of the WSFC itself, but the fact that a single FCI can be configured across a total of five VMs that would be the nodes of the WSFC. Those five nodes can also run other FCIs. Therefore, the total number of nodes in the WSFC can be more than five, keeping in mind that each FCI is constrained to five of those nodes. This limit may change in the future.

Since an AG can support up to the maximum number of replicas, the total number of VMs in a WSFC configuration should not be a concern. Even if combining one or more FCIs are part of an AG, . However, if there is any confusion, you should contact VMware support to ensure that your proposed solution is one that will be supported. Since Microsoft supports up to 64 nodes in a WSFC and SQL Server's number of nodes varies per feature, your limitation will come in based on if you are deploying an FCI and if you have shared storage.To ensure that a WSFC configuration will be supported by Microsoft, a process called validation was introduced in Windows Server 2008. It validates the proposed configuration of the WSFC before creating it. If the validation tests show that the configuration is fine, the VMs can be clustered as a WSFC. The nodes could be all physical, all virtual, or a mix of physical and virtualized nodes as long as that validation process is run and satisfies the supportability laid out by Microsoft in the Microsoft Support Knowledge Base article "The Microsoft support policy for Windows Server 2012 or Windows Server 2012 R2 failover clusters" (Microsoft KB 2775067). A mixed WSFC of physical nodes with VMs in vSphere is supported with vSphere 5.5 or later.

3.2.1.2. Active Directory Domain Services and Domain Name Services

Except for a single scenario using availability groups with SQL Server 2016 and Windows Server 2016, WSFCs require both Active Directory Directory Services (AD DS) and Domain Name Services (DNS). DNS does not have to be Microsoft's implementation. Both AD DS and DNS should be redundant to ensure that your infrastructure, especially WSFCs, can access them if required. In AD DS, corresponding objects are created for any resource that is a name, and subsequently, that name and its associated IP address are stored in DNS. .

3.2.1.3. Networking

Prior to Windows Server 2008, the traditional way to configure a WSFC was to have two separate networks, each on its own subnet:

- One for external traffic, commonly known as the public network
- One for intra-cluster traffic only, commonly known as the private or heartbeat network

This configuration also assumes that underneath those two networks, there are separate network cards. Each network card is plugged into a different physical switch, and so on. This configuration ensures that a WSFC's networking is not a single point of failure and the WSFC nodes can still communicate even if one of the networks is down. Translated to the virtualized world, it would look similar to Figure 5.

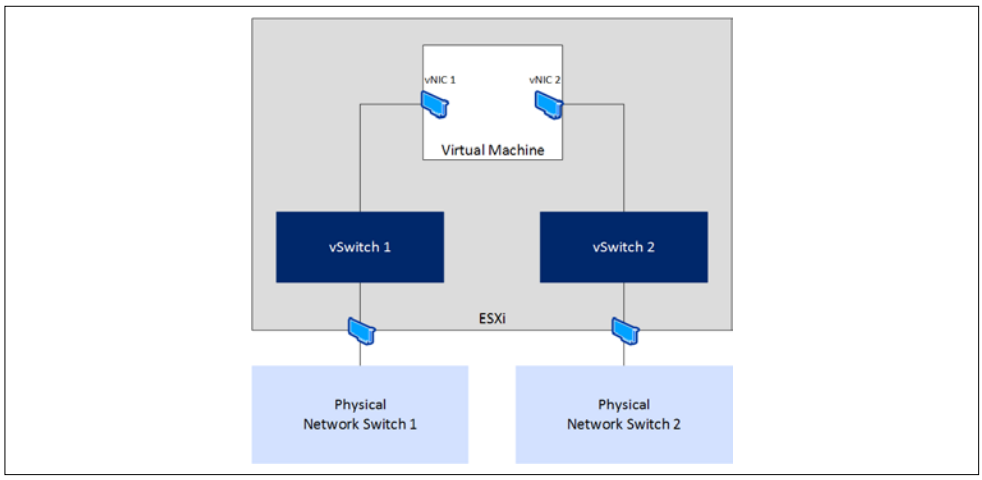

Figure 5: Example logical network topology for a WSFC node

With the introduction of validation in Windows Server 2008, Microsoft still looks for redundant networking, but will allow a configuration that does not have it. In the software defined world, mapping back to physical networks is not the same as it would be if you had physical NICs. This can be even more abstracted when you have a blade-based system where the network connections are aggregated in the chassis. The reality is that a vNIC generally cannot fail in the same way a physical network card can.

If you configure a single vNIC on VMs that will be part of a WSFC, validation will generate a warning if it sees a single network for the WSFC. If you knowingly created this configuration, this would be expected. At the top of the report, you will see that the Network category has a warning similar to the one in Figure 6.

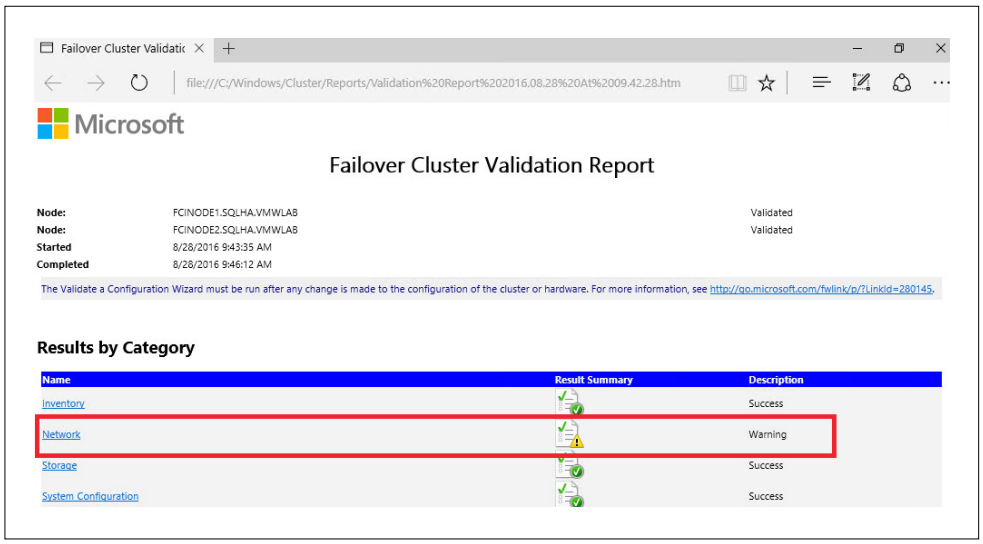

Figure 6: Network validation category reflecting a warning

Clicking on Network, it will bring you down to the section which has a list of the tests that were run. In this case, you can see in Figure 7 that Validate Network Configuration is the test that generated the warning.

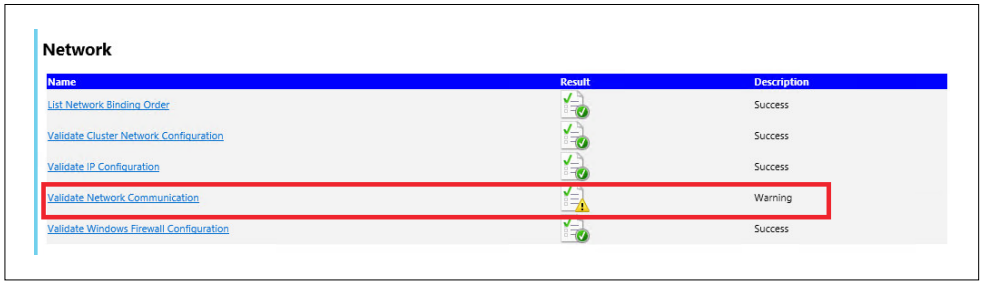

Figure 7: Validate Network Communication with a warning

Drilling down further, in Figure 8, you can see the warning messages generated due to the single vNIC that is configured. The two highlighted messages are identical, and are reporting that it only sees one NIC, and it could be a single point of failure.

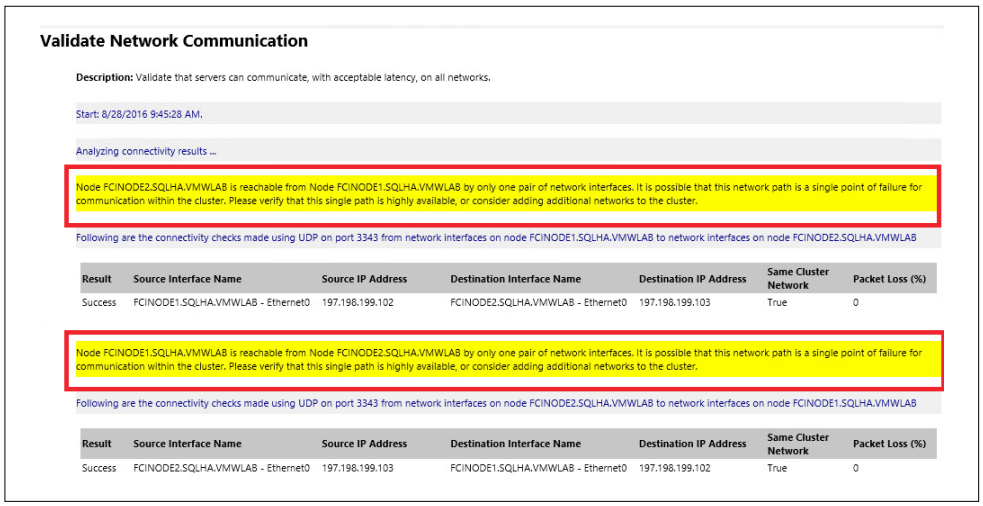

Figure 8: Single vNIC warnings

This configuration may be perfectly acceptable if the networking is fully redundant underneath that single vNIC. If that vNIC is connected to a vSwitch (or vSphere Distributed Switch) which is mapped to a single physical NIC, that means a network blip could possibly take down your WSFC, and subsequently, an AG or FCI configured on it. If this is your configuration, do not cluster Windows Server and create an AG or FCI; you are not highly available nor redundant at the network layer.

In a similar fashion, a single physical NIC inside a vSphere host that has multiple ports connected to multiple physical switches is also a single point of failure. The ports are physically considered separate, but the card itself is the single point of failure. Keep in mind that if you are using blades, the blades share an enclosure and backplane, which would include all networking.

One way to create redundancy at the vSphere level is to team the NICs in ESXi (follow VMware KB 1004088). Teaming NICs may also provide higher bandwidth since they can usually be configured in one of three ways: failover, load balanced, or a combination of both (which is generally recommended) depending on the functionality provided by the driver or software. Microsoft also supports teaming of NICs inside guests, and if you do, it is recommended to use the built-in teaming provided by Windows Server in Windows Server 2012 or later.

3.2.1.4. Quorum

To help ensure that the WSFC is up and running, it employs a mechanism known as quorum. If you lose quorum, the WSFC and anything running on top of it such as an AG or FCI will also be brought offline. Quorum makes use of another resource called a witness, which can be a disk, a file share, or in Windows Server 2016, a cloud witness in Microsoft Azure. For a file share, the share must be a Server Message Block (SMB) 3.0 share that should be highly available. Discussing quorum in-depth is out of scope for this paper. For more information, see the links at the end of the document. The one method that would require more special attention would be a disk-based witness. How to approach shared disk storage will be covered in Section 3.2.2.2.

3.2.1.5. Cluster In a Box, Cluster Across Boxes, Number of Hosts, and Node Placement

vSphere has two ways it supports creating a WSFC which are VMware concepts: cluster in a box (CIB) and cluster across boxes (CAB). CIB is when all WSFC nodes are on the same physical host. CAB is when the WSFC nodes are placed on different hypervisor hosts. These terms can apply to both AG and FCI configurations under vSphere. CIB is generally not recommended for a production environment as the ESXi host is a single point of failure, but would be more than acceptable for test configurations.

In either configuration, you must configure an affinity or anti-affinity rule for the nodes to keep them together (CIB) or separate (CAB). For CAB, configuring the proper anti-affinity rules ensures that the virtualized WSFC nodes cannot run under the same hypervisor host. This will impact the number of hosts that will be required, and dictate how vSphere Distributed Resource Scheduler™ (DRS), vSphere HA, or vMotion® can ultimately be used. For CIB, configuring affinity ensures that the nodes will be kept on the same host.

For example, if you have two WSFC nodes comprising a FCI configuration that should be CAB, you would need at least three vSphere hosts so in the event of one of the hosts failing that was running one of the WSFC nodes, that VM can be started on the third host. Depending on how vSphere is used here, the WSFC mechanisms would most likely take over first and after the node from the failed host is restarted elsewhere, it would no longer be the one owning the resources in the WSFC.

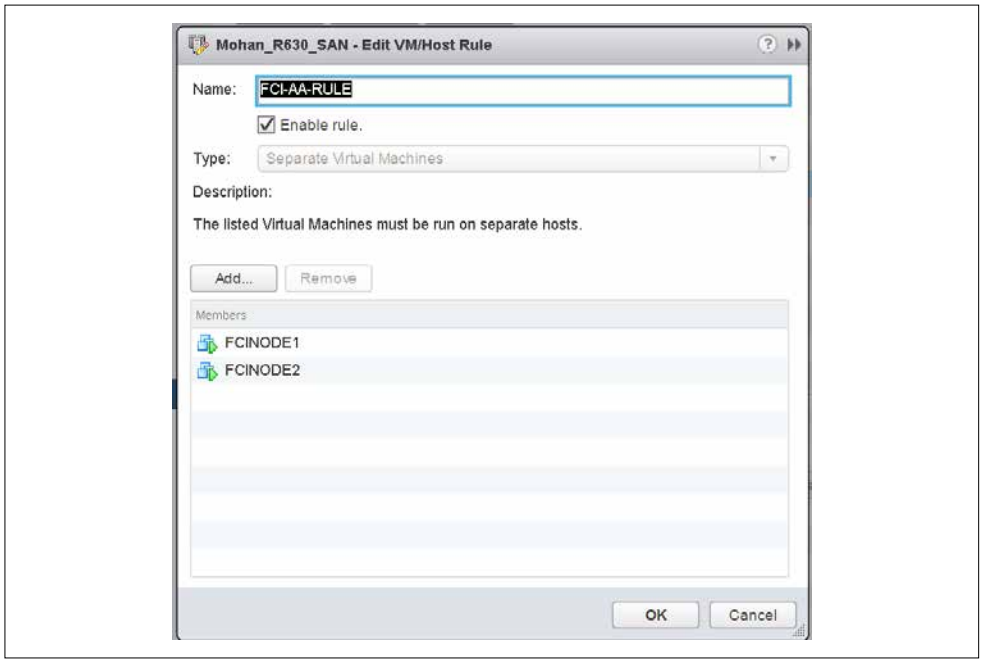

Figure 9: DRS Anti-affinity rule for two WSFC nodes in an FCI configuration

To ensure that vSphere HA and vMotion to respect the affinity/anti-affinity rules, you must also set the apprpriate vSphere HA Rule Settings as shown in Figure 8.

Figure 10: vSphere HA settings for anti-affinity for WSFC nodes set for a CAB configuration

One thing to note is that VMware does not support virtualized WSFCs on hosts that allow overcommitment of resources as documented in the "Setup for Failover Clustering and Microsoft Cluster Service" documentation.

3.2.2 Always On Failover Cluster Instances

FCIs are historically the most common form of a clustered SQL Server configuration whether physical or virtual. A WSFC with a one clustered instance of SQL Server is known as a single instance failover cluster. If you have more than one instance, that FCI configuration is known as a multiple instance failover cluster. Using active/passive and active/active are not correct since you can have more than two nodes and/or two instances in a WSFC For more information on the proper terms for FCIs, see the links later in the paper. SQL Server Standard Edition supports a two node FCI, while Enterprise Edition supports up to FCIs 25 in a single WSFC up to OS maximum for the number of nodes.

FCIs remain popular even under VMware because in a failover of the instance (planned or unplanned), it restarts automatically on another node and there is no need to worry if everything needed (such as logins, SQL Server Agent jobs, etc.) is there. It also provides availability and minimal downtime when the underlying Windows installation needs to be patched since you can just fail the FCI over to another node. The VMware options protect the VM itself , but would not protect against downtime due to patching Windows or a failed Windows installation.

3.2.2.1. AD DS, DNS, and FCIs

An FCI also gets its own unique name that gets an object in AD DS and DNS. This also means that at least one IP address must be assigned to a FCI. If a WSFC is multi-subnet, more than one IP address can be assigned. Each IP address and name are different from the names and IP addresses of the WSFC and the underlying nodes. For example, if the nodes are named PHILLY and SOUTHJERSEY, and the WSFC is named BFBRIDGE, the name for the FCI could be LIBERTYBELL. LIBERTYBELL would then get an object called the virtual computer object (VCO) that is a child to the CNO.

If the instance is a default instance, it would be accessed via LIBERTYBELL, not the other names. A named instance would have the additional name after a slash as noted above, such as LIBERTYBELL\MUSEUM. Installing another FCI in the same WSFC would requite an additional unique name and IP address(es).

3.2.2.2. Storage for FCIs

For the main data and log storage used by FCIs, it requires some sort of storage that is accessible by all of the nodes. Storage is a single point of failure in an FCI configuration (as it is with RAC). From an in-guest perspective, this can be done via the following methods shown in Table 3.

	DRIVE LETTER	DRIVE LETTER + MOUNT POINT	SMB 3.0	CLUSTER SHARED VOLUME (CSV)	LOCAL TEMPDB
SQL Server 2008	Yes	Yes	No	No	No
SQL Server 2008 R2	Yes	Yes	No	No	No
SQL Server 2012	Yes	Yes	Yes	No	Yes
SQL Server 2014	Yes	Yes	Yes	Yes	Yes
SQL Server 2016	Yes	Yes	Yes	Yes	Yes

Table 3: SQL Server FCI shared disk options per version

Drive letters, drive letter with mount points underneath, and CSV all require that the storage is presented to all of the underlying hosts and then to the VMs participating in that WSFC configuration. For more information on how SQL Server FCIs can use CSVs, see this SQLHA blog post.

Presenting that storage is supported under vSphere in the following ways:

- Fibre channel
- Fibre channel over Ethernet (FCoE) – vSphere 5.5 and later supports native; for more information on FCoE see note 4 of Table 1 in the VMware Knowledge Base article "Microsoft Clustering on VMware vSphere: Guidelines for supported configurations"
- iSCSI – native (vSpehre 5.5 or later) or in-guest. If using iSCSI inside the guest, separate vNICs should be used connected to a different network than the one used by the nodes for SQL Server or WSFC traffic.
- Physical raw device mapping (RDM, also known as a physical RDM or pRDM)
- VMFS (CIB only with vSphere 6.0 or later)

For an FCI configuration, the method used the most is to use RDMs, followed by in-guest options such as iSCSI and SMB 3.0. FCIs are virtually the only place that RDMs should be used for VMs under vSphere. One of the limiting factors of RDMs is that as of the writing of this paper, an ESXi host can only support up to 60 RDMs per VM (up to 4 Virtual SCSI controllers, 15 disks per controller), and a total of 256 RDMs per host, so plan your deployments carefully. Note that currently, VMware Virtual SAN™ is not supported for FCI deployments since it cannot present storage as an RDM; it relies on VMs using VMDKs. Similarly, vSphere Virtual Volumes™ (VVOLs) are also currently not supported for any WSFC-based configuration which includes FCIs.

To configure RDMs, the disks must be attached to a SCSI controller and set to physical. An example is shown in Figure 11. With vSphere 5.5 Update 3 and later, use the VMware Paravirtual SCSI (PVSCSI) adapter for the disks that will be used by the RDMs, and for better performance, consider using more than one PVSCSI controller to split up the I/O load for better concurrency.

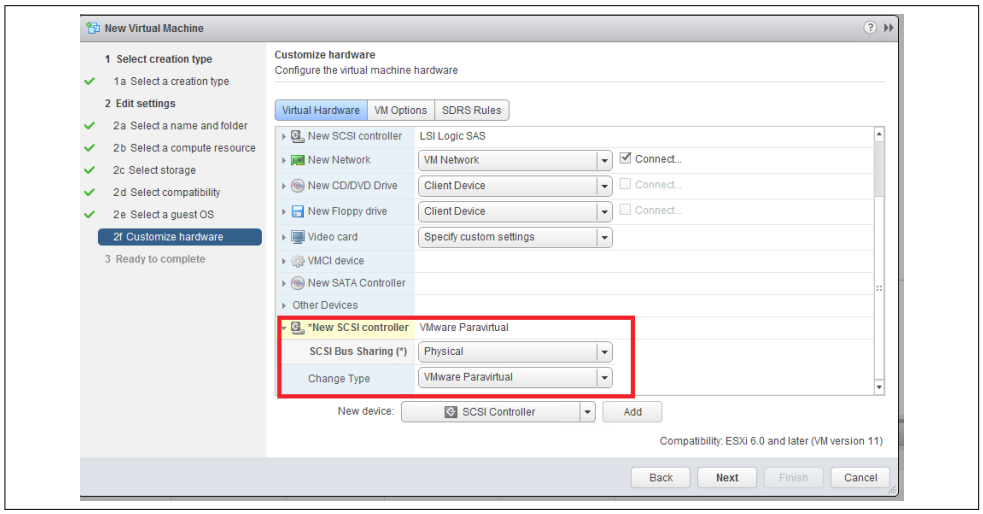

Figure 11: Adding a PVSCSI controller with physical bus sharing

To add a RDM disk to the VMs that will be participating in an FCI, follow the instructions below.

1. Edit the settings of one of the nodes (it does not matter which).
2. Under New Device, select RDM Disk as shown in Figure 12.

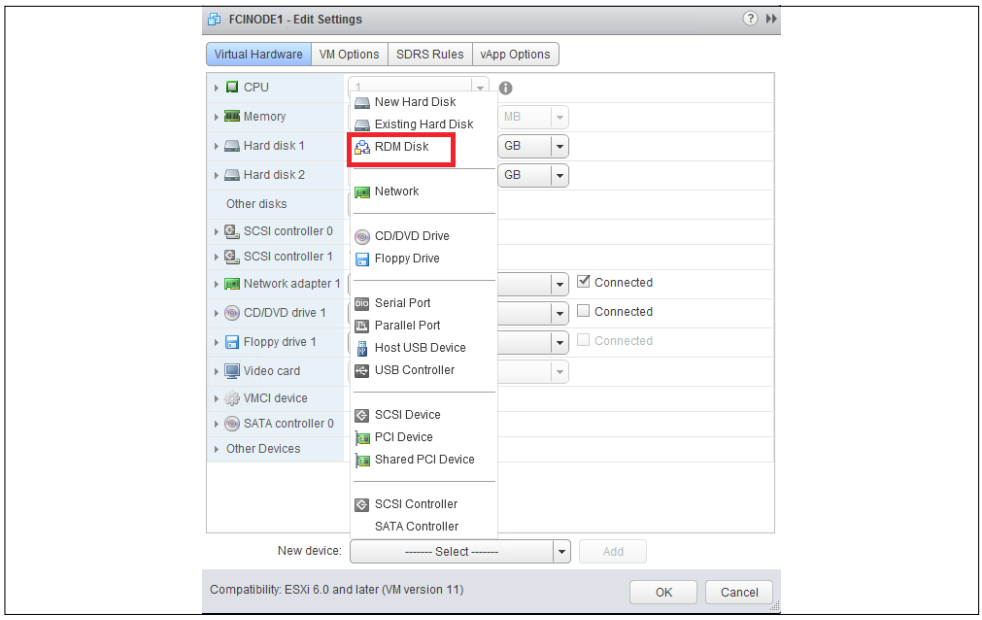

Figure 12: Selecting the RDM Disk option

3. Click Add as shown in Figure 13.

Figure 13: RDM Disk selected

4. A list of the LUNs available will now be displayed. Select the disk to add and click OK. An example is shown in Figure 14.

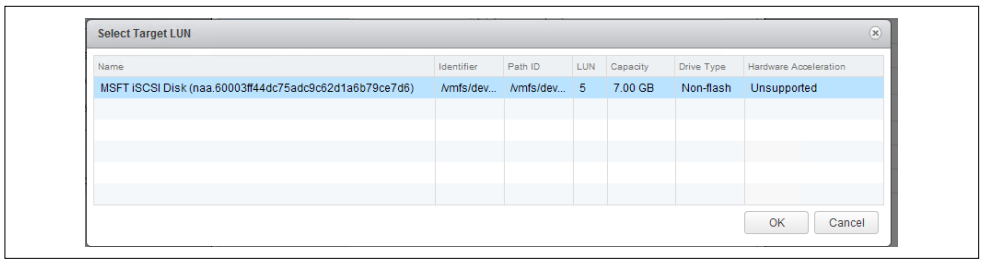

Figure 14: Selecting a LUN

5. Expand New Hard disk. Ensure that Compatibility Mode is set to Physical, the Virtual Device Node is set to the correct PVSCSI controller, and that Disk Mode is set to Independent – Persistent since this is a physical disk. An example is shown in Figure 15.

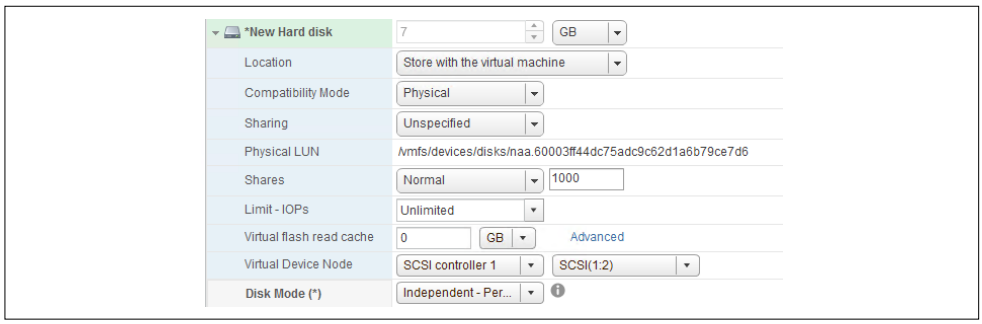

Figure 15: Configuring the RDM

6. Click OK to add the disk. If you look at the corresponding datastore for the VM, you will see a VMDK with the size of the RDM that was added. This is a "friendly" visual representation in vCenter®. What it looks like underneath the covers is a bit different.

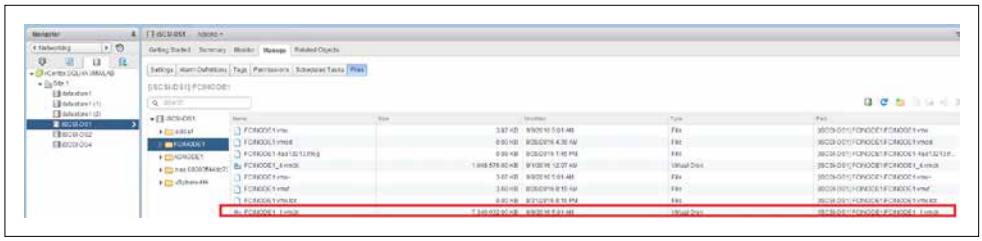

Figure 16: RDM as seen in vCenter

Looking underneath the covers, the actual VMDK is a small file (520 bytes). The RDM is represented by a mapping file with a name of friendlyname-rdmp.vmdk format with the size of the underlying LUN. This can be seen in Figure 17.

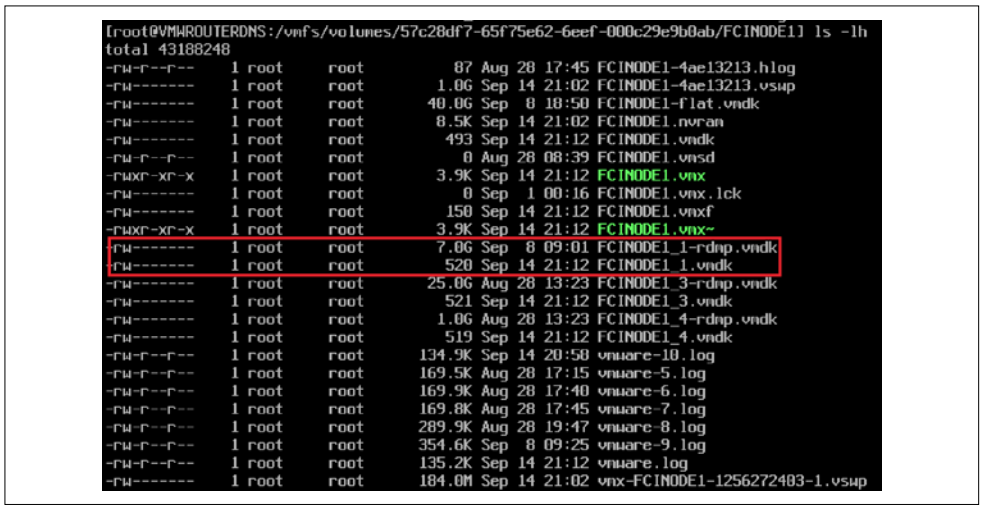

Figure 17: Files associated with the RDM as seen in the command line

7. On another VM that will be participating in this configuration, instead of adding the RDM Disk, select Existing Hard Disk as shown Earlier in Figure 12.
8. Click Add.
9. Navigate and select the file that was created as shown in Figure 18. Click OK.

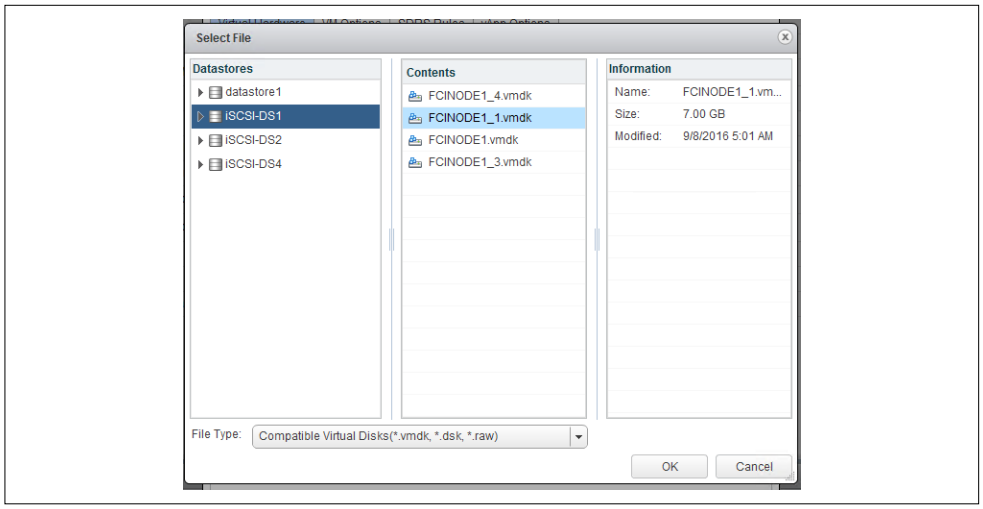

Figure 18: Selecting the file created for the RDM

10. Similar to Step 5, expand New Hard disk. Ensure that the Virtual Device Node is set to the correct PVSCSI controller and that Disk Mode is set to Independent – Persistent since this is a physical disk. An example is shown in Figure 19.

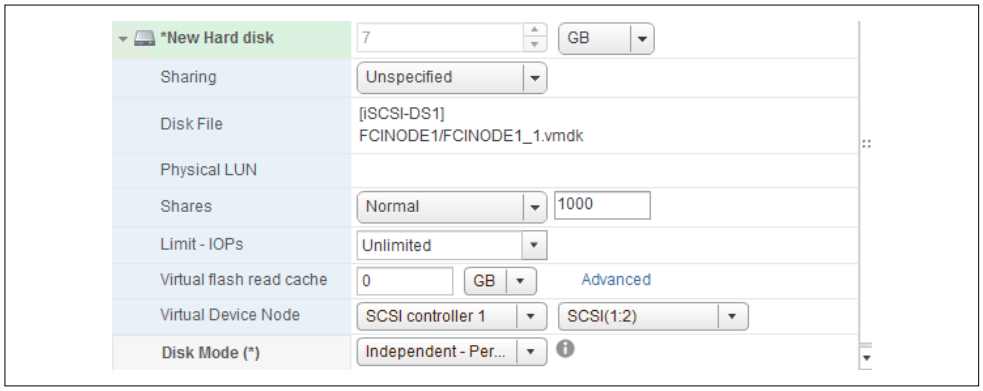

Figure 19: Configuring the RDM on another node

11. Click OK to add the RDM to this node.
12. Repeat Steps 7 – 12 for any other VM which needs this storage.
13. When complete, the disk will be available in all of the nodes as shown in Figure 20 and can be configured for use in Windows Server.

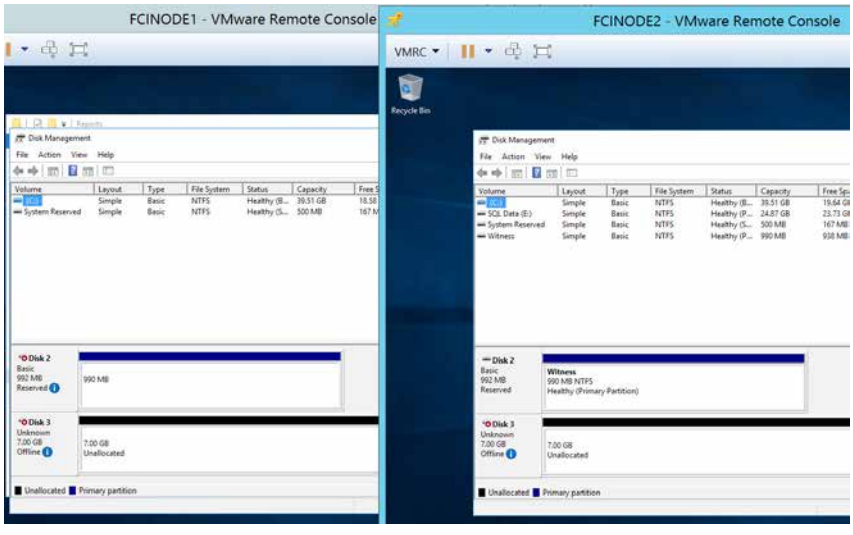

Figure 20: RDM successfully added to all nodes

Remember when formatting disks for use with SQL Server, use a 64KB allocation unit size and for a file system, use NTFS, or with SQL Server 2014 or later, you can also use ReFS.

Note that if you are using RDMs and the WSFC node VMs are for some reason on the same host, you will see errors in validation similar to the one in Figure 21.

Figure 21: Disk errors during validation

Underneath the covers, vSphere 5.5 and later supports a Path Selection Policy (PSP) of Round Robin (PSP_RR) with shared disks as long as the underlying storage supports it. PSP is set at the host level, and hosts with a different PSP setting running clustered Windows Server VMs is a supported configuration.

If using SMB 3.0 file shares for SQL Server data, log, and backups, VMware only supports it with Windows Server 2012 or later. This is not a Microsoft limitation. This requirement would be true if the data, log, or backups were stored using FCIs or standalone instances of SQL Server.

There are three places where "local" disks can be used with a FCI configuration: for the OS disk, TempDB, which is SQL Server's temporary workspace, and for the CIB scenario. The OS disks must be created as Thick provision eager zeroed (eagerzeroedthick) as shown in Figure 22.

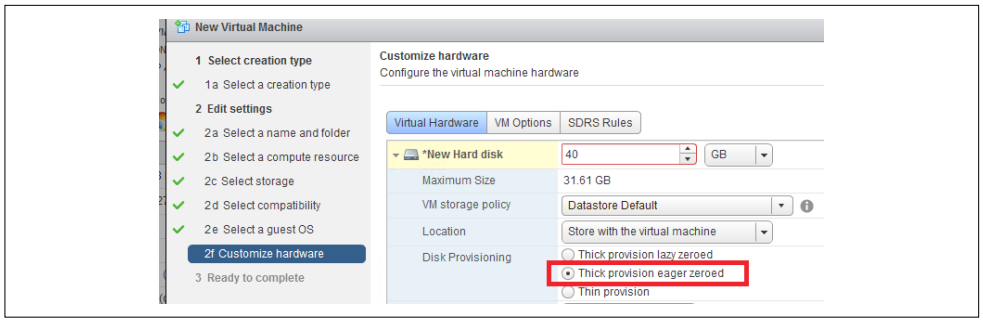

Figure 22: Configuring the OS disk for a WSFC node VM

The OS disks for each nodes should not be placed on the same datastores and where possible, those datastores should not be on the same storage unit. When combining with the data/log/backup RDMs, those disks even though they are technically a single point of failure, should be separate from the OS disks to have some level of redundancy at the storage layer.

If the desire is to create a VMDK to use for TempDB, it should only be done after evaluating that this is required and that you will have enough IOPS to sustain the workload. It also complicates the configuration a bit because it can affect vSphere HA and/or vMotion and have local disks as part of the SQL Server deployment; this will be addressed later in the paper.

If using a drive letter, drive letter + mount point, or CSV, make sure to set HKLM\System\CCS\Services\Disk\TimeOutValue to 60 for each of the WSFC nodes in Windows Server. This is a VMware requirement, not one from Microsoft.

3.2.3 Always On Availability Groups

Availability groups are comprised of replicas – one primary and up to eight secondary in SQL Server 2014 and later, or one primary and four secondary replicas with SQL Server 2012. AGs can be combined with FCIs for even more protection inside a guest environment. Unlike FCIs which are configured at the time of installing SQL Server, an AG can be configured post-SQL Server installation.

AGs have enhancements over other database-level feature such as:

- A Listener which, like the unique name for an FCI, provides abstraction for end users and applications to access an instance without having to worry about which node it is running.
- One or more secondary replicas which can not only be used for availability, but also for read only activity as well as backups and DBCCs (these additional uses affect licensing)
- Providing both high availability and disaster recovery in the same feature

An AG keeps the secondary replicas in sync by sending the transactions across a log stream. The synchronization can occur synchronously or asynchronously, which means that the network between the VMs participating in an AG has to be robust and reliable. In some cases, this may require an additional dedicated network that could be mapped to an additional vNIC (and possibly a dedicated physical network/VLAN). This would be determined through testing.

In SQL Server 2012 and 2014, AGs are an Enterprise Edition feature. In SQL Server 2016 and later, Microsoft allows you to create an AG using Standard Edition, but you are limited to a total of two replicas.

3.2.3.1. AD DS, DNS, and AGs

SQL Server 2012 and 2014 require the use of AD DS with all AG configurations. If a Listener is configured, it requires a VCO along with an entry in DNS. With the combination of Windows Server 2016 and SQL Server 2016, you can create a supported AG configuration that does not require AD DS. This deployment method for AGs uses certificates, similar to what was done for DBM. DNS is still required.

3.2.3.2. Storage and Networking for AGs

Each replica has a copy of the database(s) participating in a given AG. This means that if you have a database that is 2TB, a two-replica configuration will require 4TB of space. An FCI only has a single copy of the data, but is a single point of failure.

Despite the increase in storage usage due to multiple copies of the databases participating in an AG, one of the biggest benefits of using an AG under vSphere is that it does not have any special shared storage requirements unless you are combining FCIs with AGs. This means that with vSphere, unless there is reason that RDMs must be used for the SQL Server data and transaction log files (such as combining FCIs and AGs, and using RDMs for the FCI portion), you should use standard VMDKs and VMFS datastores instead of RDMs. This simplifies the planning and configuration, and makes AGs a good fit for use under vSphere. As noted earlier, VVOLs are not supported with VMs that will contain an AG since it is not supported for a WSFC configuration. .

You must plan how the VMDKs would be laid out not only for performance of SQL Server databases and transaction log files, but for availability as well. Placing all of the files on the same datastore on the same storage unit would be a single point of failure. An example two-node AG configuration using VMDKs would look like Figure 23 if architected properly. There may be more than one data file which may or may not be on the same datastore for a particular VM (for example, two data files for VM 2), but that would be acceptable compared to the alternative.

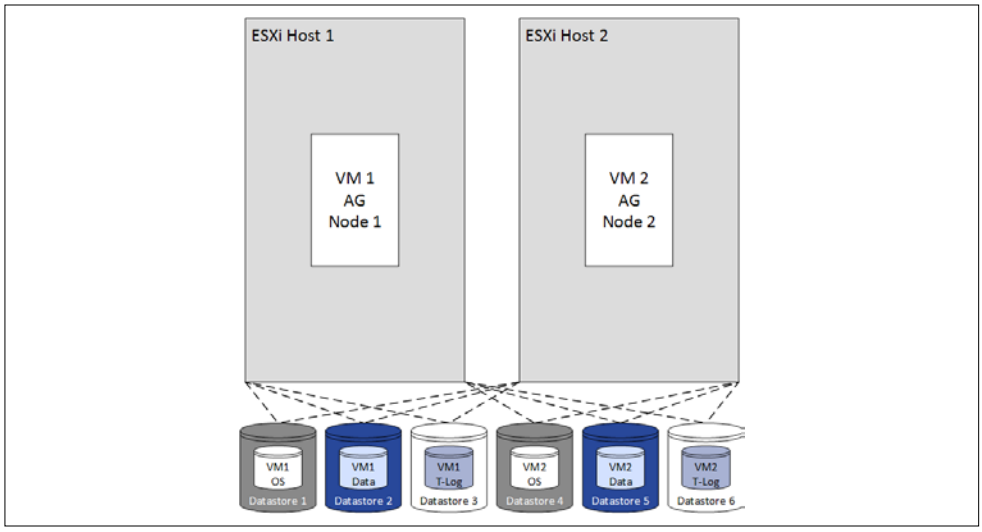

Figure 23: Example AG storage configuration

Imagine that the six datastores are spread over at least two different storage units with logical separation. For example, the two different OS datastores are on different underlying storage, and that the respective data and transaction log ones are as well. The goal of availability groups is to not have the underlying storage be a single point of failure, which is the case with FCIs. Some may configure an AG to just use the readable replica functionality, and that would be a perfectly acceptable use of having the data for all replicas on the same storage unit. This may or may not limit how you would use an underlying option such as VSAN for storage, so you should carefully consider how you plan on using AGs before deploying the VMs for them.

Storage performance is crucial for an AG. Not only do you need to worry about the primary replica which is the read/write copy, but if using synchronous data movement, the secondary replica also needs to have as good, if not better, IOPS to ensure that it does not lag behind. If a secondary replica will be used for other purposes such as read only traffic, on that replica, additional IOPS will be needed for the database participating in the AG as well as TempDB. If these are VMDKs, they should be configured as eagerzeroedthick.

4. vSphere Availability Options and SQL Server Interoperability

This next section will cover the availability features available in vSphere and specifically how each one works in conjunction with VMs that have SQL Server running inside.

4.1 Distributed Resource Scheduler

vSphere's Distributed Resource Scheduler (DRS) is a feature that will reallocate resources based on configured policies. One of DRS' main benefits is that it can provide load balancing for VMs running in a vSphere cluster DRS can move a VM to another ESXi host based on resources set by the virtualization administrator. DRS has a variant (Storage DRS, or SDRS) which would move VMDKs based on IOPS or space utilization of the datastore in a datastore cluster. For Tier 1 mission critical SQL Server workloads, DRS may or may not provide a benefit, so its affect on the VMs should be assessed and tested. For many VMs running SQL Server inside the guest, DRS should be completely transparent. If DRS configured policies seem to cause an noticeable impact on SQL Server's performance due to an unwanted migration of a VM to another ESXi host, the DBAs and virtualization administrators should come up with a solution which may include not enabling DRS for select VMs, or at a minimum, adjusting the management policies and/or the parameters associated with DRS to be less aggressive. For mission critical SQL Server workloads, predictable and reliable performance should be the end goal.

DRS will also enforce the affinity or anti-affinity rules configured for CIB or CAB WSFC VMs. Even if someone initiates an action such as a manual vMotion which can potentially violate the rule, DRS will enforce it within five minutes.

4.2 vSphere High Availability

vSphere HA is a feature where you cluster the servers that will be the ESXi hosts. With vSphere HA, a VM will be restarted on another ESXi host if there is a failure of the host on which it was originally running. There is no graceful shutdown of the guest OS in the event of a possible ESXi host failure. Therefore, if only HA is employed, when the VM restarts on another host, you will need to make sure everything is working properly since the SQL Server instance and/or the databases may be in a state that needs attention from the DBAs.

Using vSphere HA, applications and end users would notice that SQL Server was unavailable while the VM was brought online on another host. One difference from an FCI, however, is that an application can be made cluster aware if coded properly; there is no equivalent API for vSphere HA. At that point, it would be recommended that before any connections were allowed to use SQL Server, the DBAs ensures that the instance and its databases are fine. vSphere HA is a stop and a start as if a physical host was used, so SQL Server will go through its standard recovery mechanisms.

Once configured, vSphere HA is always enabled for every VMs running in the vSphere cluster. Combining vSphere HA with one of the SQL Server availability features is a natural fit and compliments them since vSphere HA can restart the VM on another host but an instance or database can be brought online before that. The VM coming back online would make the entire solution "whole" again. vSphere HA, like the SQL Server availability features, does not supplant the need to have good, known, and tested backups.

4.3 vSphere Fault Tolerance

vSphere Fault Tolerance (FT) creates an identical clone of a VM and keeps it up-to-date continuously. This means that if the ESXi host of the original VM somehow encounters a problem, the FT copy can replace it only if what is running inside the VM is not broken, such as corrupted installation of SQL Server or Windows Server. It is a 1:1 ratio from the Primary to the Secondary VM, meaning with FT you can only have a single copy of a source VM.

If a failover occurs to the Secondary VM and becomes the new Primary VM, a new Secondary VM will now be deployed for its replacement on another host. The Primary and Secondary FT VMs cannot be on the same host, so a minimum of three hypervisor hosts will be needed in the vSphere cluster for full protection. Only one copy of a FT VM is accessible at any given time.

As of the writing of this paper with vSphere 6, FT limits you to at most 4 vCPUs for any VM that would be participating. This would restrict the size of SQL Server implementation that would use vSphere FT. If this changes in the future, larger implementations would be possible.

FT is not currently supported with clustered configurations of SQL Server.

4.4 vSphere vMotion

vSphere vMotion gives VMs the ability to move from host to host, network to network, storage to storage, and datacenter to datacenter without stopping and starting a VM. This means that if SQL Server is running in the guest, there is no downtime – applications and end users can still use it during the migration with no downtime and minimal impact to performance. For mission critical workloads that need to be rehosted, this is one of the biggest benefits of vMotion. vMotion in this manner also can enable seamless hardware migrations for vSphere architectures with no downtime.

From a SQL Server perspective, vMotion is great for standalone SQL Server implementations that are using no other form of protection and if a VM needs to be moved where no downtime inside a guest is possible. vMotion has always been supported for non-clustered (meaning AG or FCI) configurations with networks as low as 1Gb. Multiple physical NICs in the ESXi host could be used for latency versus throughput. How to properly architect vMotion in a general context is outside the scope of this paper. However, if using vMotion with very large VMs with SQL Server inside, be careful of Stun During Page Send (SDPS). SDPS will slow down vCPUs to ensure that memory remains fast since vMotion is largely about moving memory. If you have a lot of memory assigned to the VM with quite a few dirty pages, it could affect performance of SQL Server inside the guest since the job of SDPS is to ensure that vMotion will be successful. So with SDPS, you are trading off performance to be able to successfully vMotion the VM. SDPS can be disabled (see VMware KB 2007595). Test before disabling.

AGs and FCIs are fully supported with vMotion as of vSphere 6, but in those cases, the SQL Server-based feature would be the primary form of failover. vMotion would be a good choice for secondary availability or for other purposes, like migrating hosts. Clustered configurations of SQL Server under vSphere must meet the following conditions for vMotion:

- CAB configuration only
- FCI configurations must use RDMs
- AGs with only standalone instances only have no stringent storage configuration requirement for use with vMotion
- The network used by vMotion where an FCI or AG is involved must be at least 10Gb, not 1Gb. As noted earlier, 1Gb is supported for VMs with purely standalone instances of SQL Server (including not being part of an AG). With 25 and 50 Gb support in vSphere 6.0.1, those options should be explored if necessary.
- The version of Windows Server must be 2008 R2 with Service Pack 1 or later
- Inside the guest, the WSFC value for SameSubnetThreshold must be set to 10 or more. This setting tells the WSFC that it can tolerate up to 10 missed heartbeats before an automatic failover of a Role will occur.
- If combining local VMDK disks for TempDB and pRDMs for an FCI, vMotion requires a destination datastore to place the local disks at the target host.
- The virtual hardware version for the VM must be version 11 or higher

When an affinity/anti-affinity rule is employed with DRS/SDRS, it can be violated by a manual vMotion. Because the DRS/SDRS rules are run periodically, this problem should correct itself and the VMs that should be together or apart depending on the rule will be placed where they should be across the vSphere hosts.

From a SQL Server perspective, one of the biggest benefits of vMotion is that everything that is in memory – such as execution plans and recently used data in cache – remains there. Therefore, performance for SQL Serve instances and databases will remain the same instead of hitting the proverbial reset button in a traditional failover. There may be a short impact during the host switching, but applciations should not really notice.

4.5 Site Recovery Manager

VMware Site Recovery Manager™ is an orchestration solution that enables disaster recovery and avoidance, as well as site migration. Disaster recovery allows you to come online elsewhere if your primary data center was experiencing a major downtime event. Site Recovery Manager requires the use of either block-level replication for storage or vSphere Replication™. Either can be used, or it is possible to mix them as well to synchronize VMs between sites.

Choosing which replication type to use (built-in with vSphere replication or block level via the underlying storage), or whether to combine them, is a broader discussion. Similar to the common discussion in the SQL Server world with AGs since they are transactional and under a DBA's purveyance, it boils down to control: do you want to control the replication, or do you want it to be transparent to the platform sitting above? In the case of AGs, it directly correlates to RPO since LSNs are the unit of measurement. With Site Recovery Manager, the goal is to have an exact, up-to-date copy of the VM at Site A over in Site B. Work with the respective teams in your environment to determine the best method to use.

Site Recovery Manager can be used with AGs and FCIs as well – it is not limited to standalone implementations of SQL Server. Figure 24 shows a node of an AG that is being replicated using vSphere Replication to another site named vcva02.vmlab.local.

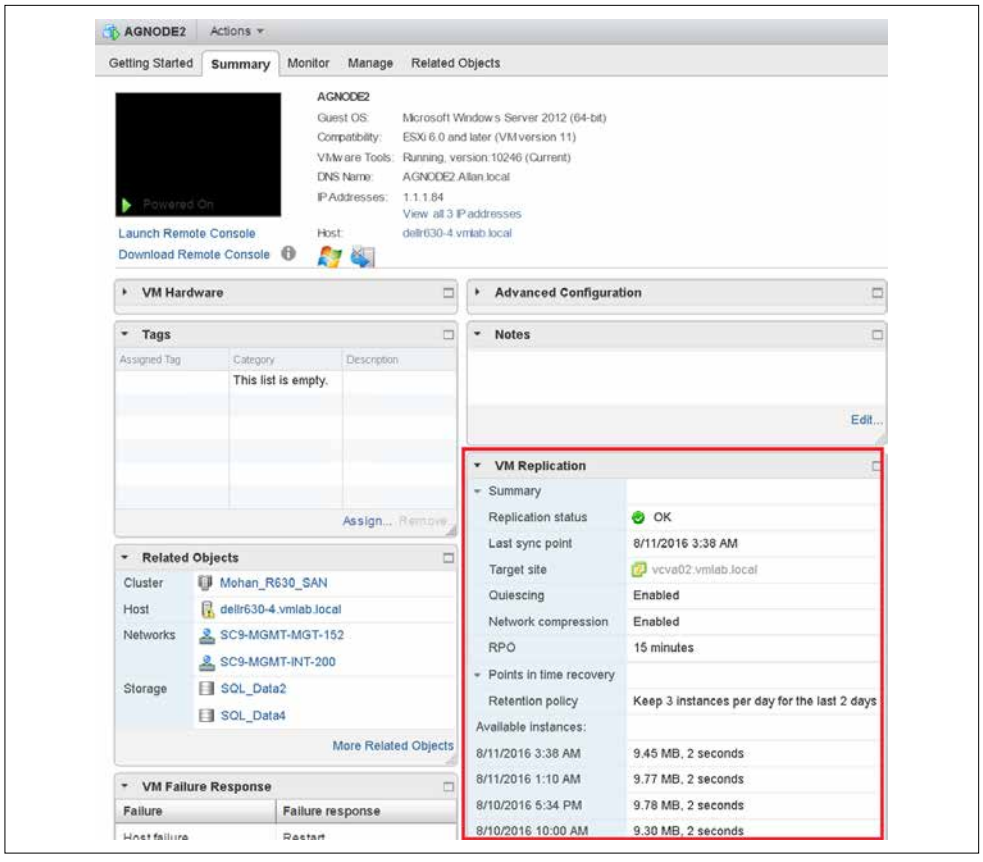

Figure 24: VM with replication enabled

At the remote site, the target VM, known as a placeholder, is not online and looks similar to what is shown in Figure 25.

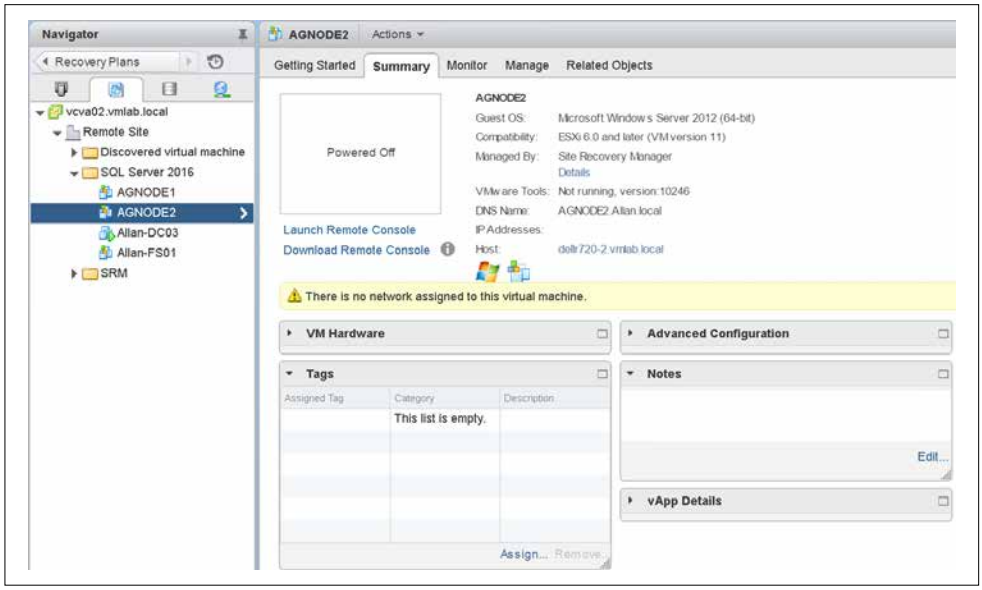

Figure 25: Replicated VM at the remote site

One of the biggest challenges with disaster recovery is that it is very hard to test since it would require actually switching to the disaster recovery site and taking down production. Currently, no Windows Server or SQL Server feature allows you to keep the primary site going while you test your disaster recovery plans. Site Recovery Manager allows you to keep the primary site running and servicing applications and end users while you test and bring the disaster recovery site online – even if it is in the middle of the work day. This allows you to simulate your failures with no impact to the business. Figure 26 shows the recovery plan named AGNodes-RP after initiating the test process.

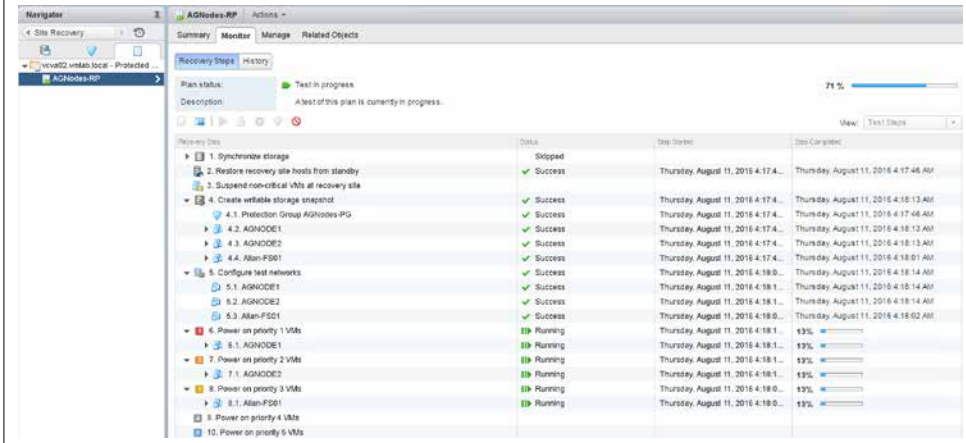

Figure 26: AGNodes-RP recovery plan after starting a test

After the process is complete, as shown in Figure 27, both vcva01 (the primary site) and vcva02 (the disaster recovery site) both have live VMs. Note that for vcva02, the Recovery Site version, the AG is showing that it is running with no issues.

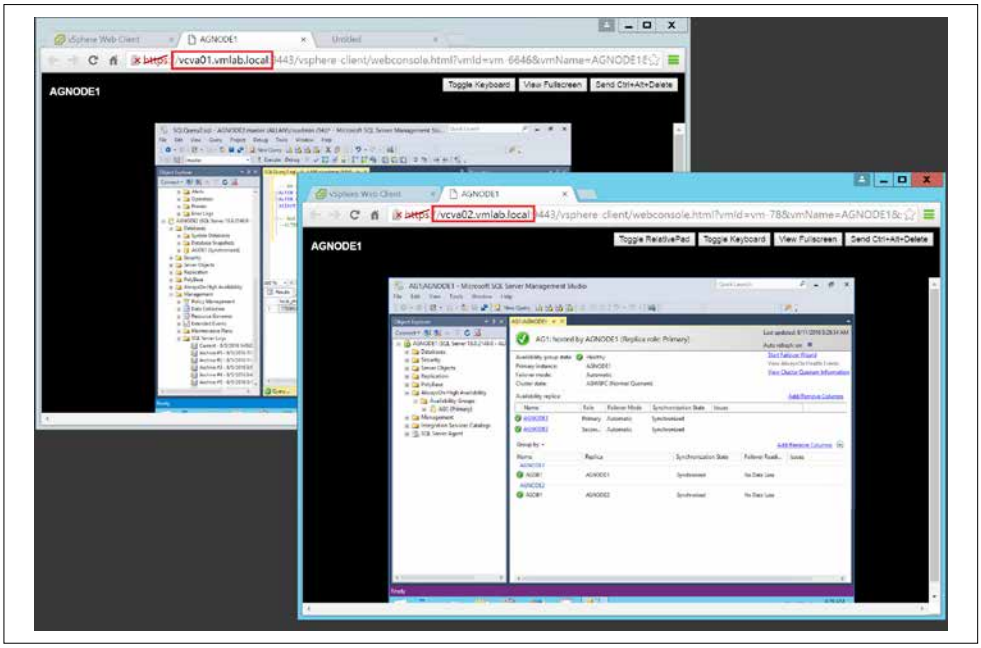

Figure 27: AGNODE1 running in both vcva01 and vcva02

The reason this is a huge benefit and why testing disaster revovery is very difficult is that when things like AD DS are involved, only one server with the same name can be active at the same time. The method deployed by Site Recovery Manager effectively simulates a wall which allows the two sites to coexist without affecting the production site. Outside of using Site Recovery Manager, there is no way to test your disaster recovery while leaving your primary site up and running today that is not extremely difficult (such as severing the network between the sites).

When it comes to business continuity both for testing as well as in the event a real switch of sites occurs, this means that everything – not just the VMs with SQL Server instances and databases – must exist at the location designated for disaster recovery. This would include all application servers, web servers, AD DS, etc. For AD DS, depending on how your administrators deploy AD DS, it could be a replicated VM of a virtualized AD DS server, a live VM that gets replicated changes from the primary AD DS server(s), or a physical server which would function the same as the live VM. If it is a replicated VM is configured as part of a plan in SRM, you may want to include it in the workflow and ensure that it is fully up and running before bringing anything else online.

One of the biggest benefits of Site Recovery Manager is that because a workflow is involved, other tasks can happen such as potentially changing IP addresses and running scripts against the VMs. For more complex configurations like an AG or FCI, changing networks may not be what you should (or can) do, but Site Recovery Manager is a flexible tool in the arsenal for disaster recovery that provides more than an on/off switch for a VM. How you use Site Recovery Manager will depend on your infrastructure, but you may not have to do things like extend your existing production network to another site. No matter how it is configured, Site Recovery Manager and the plans associated with it should be planned and tested accordingly.

5. Determining an Availablity Architecture for SQL Server and vSphere

Architecting the availaibility strategy for your SQL Server deployments is based on requirements. The following are some common questions which can assist in the process:

- What are you trying to protect?
- What are your limitations – skills of the team, resources, versions/editions of Windows Server, SQL Server, etc.?
- What does the application using SQL Server support?
- What are your documented recovery point and recovery time objectives?

There is no one-size-fits-all solution when it comes for ensuring that a VM or set of VMs is highly available. The key difference in the vSphere features versus what is provided by SQL Server is that the vSphere features are designed to protect the VM, whereas the SQL Server features (or ones that hook into the proper SQL Server APIs, such as third party backup utilities) protect the instance or data and provide point-in-time recovery of the data with the use of things like AGs or transaction log backups.

Control plays into this decision – do you want the DBAs to manage the availability story for SQL Server, or the non-SQL Server administrators? The answer is that everyone needs to work together to craft a solution that meets the requirements, and that may mean just using SQL Server features, vSphere features, or a hybrid of both. Since the primary responsibility for an instance of SQL Sevrer lies with the DBAs in most organizations, they should be heavily involved in the decision of what features should be used.

For mission critical, Tier 1 applications, it is common to lead with a SQL Server feature such as an AG where you can configure synchronous replicas that would account for the RPO aspect of availability and would be controlled by the DBA. vSphere features such as HA and vMotion could also be employed and used when needed – or even combined with each other and used in conjunction with an AG or an FCI.

As a counterpoint, for less critical, Tier 3 VMs hosting SQL Server instances and databases which may have had no availability story prior to virtualization, it may be excessive to deploy an AG or FCI, but using something like FT or even just vSphere HA not only gives that VM protection where there was none, but it should exceed expectations where none existed.

The vSphere solution that is deployed should be the result of discussions with all groups involved. At the end of the day, it is usually the DBAs who will be ultimately be responsible for the maintenance, performance, and availability of SQL Server instances and databases running in the VMs. Clear lines of delineation for administration should be discussed, and DBAs – or a subset of them – should be given read only rights in vCenter.

If you have more than a handful of SQL Server deployments that are or will be virtualized, you should consider having dedicated vSphere clusters or ESXi hosts for them. This may reduce your licensing costs instead of having to account for each VM individually, or license an AG or FCI solution as if each VM was like it was on physical hardware. Reducing cost is often one of the drivers for virtualizing SQL Server, so as part of an overall mission critical strategy, licensing is a key component.

6. Summary

Mission critical deployments of SQL Server are commonplace now under vSphere. One of the most important tasks is to ensure that the VMs and the workloads running in them have the availability that is required by the business. Whether you use the native SQL Server and Windows Server features, the vSphere features, or a combination of what Microsoft provides with what VMware provides, you can create resilient deployments. The SQL Server availability features and the ones in vSphere can be used exclusively, or compliment each other; things do not need to be all one or the other. Using both, you can create a best-of-breed solution that will not only provide the reliability that you require, but using vSphere, an agile, robust deployment architecture that will benefit both DBAs and system administrators alike.

7. Acknowledgement and Credits

Allan would like to send a special thanks to both Pure Storage and Tintri for providing the storage used during the writing of this paper to enable the Site Recovery Manager scenario, and Mohan Potheri at VMware for use of his lab for portions of the testing, Deji Akomolafe's technical expertise also was invaluable during the writing of this paper.

Reviewers: Deji Akomolafe, Niran Evan Chen, and Don Sullivan from VMware and Max Myrick from SQLHA.

8. Related Documents and Links

General Information

- Links to the official VMware documentation for configuring a WSFC with vSphere per version can be found in VMware KB 1004617
- The whitepaper "Performance Best Practices for VMware vSpehre 6.0", which can be found here.
- The whitepaper "Architecting Microsoft SQL Server on VMware vSphere" which is found at this link.
- The documentation entitled "vSphere Resource Management", which can be found here.
- A good VMware KB to read to help optimize network performance for a VM is 2008925.
- Although these should be irrelevant for PVSCSI, you may need to follow the advice found in VMware KBs: 1267, 1268, and 2053145, which talk about disk-related configuration topics like queue depth and VMKernel Admittance.

Selected SQL Server Availability Links

- Information about the book Mission Critical SQL Server
- Allan's Always On Availability Groups FAQ
- Video – Did You Vote Today? A DBA's Guide to Cluster Quorum
- What's New in Windows Server vNext: Cloud Witness
- SQL Server Availability Group Configurations with NO Domain Requirement

SQL Server Availability Terminology

- Once More With Feeling: Stop Using Active/Passive and Active/Active
- AlwaysOn is the New Active/Passive and Active/Active
- Dear Microsoft: I Love You, But You're Driving Me Batty

SQL Server and Windows Server Licensing

- SQL Server 2016 Licensing Datasheet
- Windows Server 2016 Licensing Datasheet

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com