Technical Paper

Leveraging VMware Software to Provide Failover Protection for the Platform for SAS® Business Analytics

April 2011

Table of Contents

About this Document

This document describes recent research, on leveraging features of VMware software to provide failover protection for the Platform for SAS Business Analytics. It is intended to prepare SAS field technologists (system engineers and consultants) for customer discussions about increasing availability in virtualized environments built with VMware software. The intended audience is expected to be knowledgeable about SAS software and the deployment of the Platform for SAS Business Analytics. However, the intended audience is likely to have a more limited knowledge of the VMware technologies. The goal is to provide the reader with a level of comfort when discussing deployment details with the customer’s IT staff. The goal is not to make the reader a VMware expert. It is to allow the reader to form an effective team with the customer’s VMware experts, leveraging the expertise each brings from their different domains.

Introduction

VMware software provides mechanisms that support maintaining availability and minimizing service interruptions for the Platform for SAS Business Analytics users. This is a critical task for IT staff. This paper focuses on how VMware software can be used to increase availability of the platform by providing failover protection to critical SAS® components. While failover, in itself, does not guarantee high availability, it is an important part of a larger, overall, effort to ensure availability. Testing was done to better understand the roles VMware vMotion and VMware HA (two features of VMware software) can play in providing both planned and unplanned failover protection. This paper provides a high-level discussion of the relevant concepts and descriptions of the software components involved. Detailed implementation instructions are not included. Existing documentation from both VMware and SAS provide this information for their respective products. However, specific information about appropriate configuration options, for both products, is included. Descriptions of the testing environment, the testing scenarios examined and the results of those tests are outlined.

What is Failover?

The term failover is used to describe a process in which activity on one server is moved to another server in the event the first server is, or expected to be, no longer available. The goal is to move the processing to a healthy server and ensure the continued availability of important services to users. Ideally, this is done quickly, automatically and without interruption in service to the end users. Failover is most often thought of in situations where a server fails unexpectedly (what can be described as an “unplanned failover”). However, it is possible to make use of failover in planned situations as well (planned failover). An example of planned failover is moving workloads to a second server to perform maintenance on the first. In fact, the majority of system downtime is generally this type of planned downtime for system maintenance. This paper touches on features of VMware software that can be helpful in handling both planned and unplanned failover scenarios.

Goals and Objectives

The focus of the effort described in this paper was kept very narrow: understanding how the VMware vMotion and VMware HA technologies could be used to support planned and unplanned failover for a SAS deployment built using a realistic architecture. During testing, user activity was simulated through several of the SAS client applications and touched most of the back-end SAS servers running in the compute tier. However, no effort was made to heavily tax the systems either through analyzing huge quantities of data or through the simulated activity of a large user population. Nor was the deployment configuration tweaked to maximize performance. Readers are, therefore, discouraged from using the testing environment described here as a model for their own SAS® software deployment.

As a final caveat, while failover can be an important part of the overall strategy, it does not, by itself, guarantee high availability. The testing described here focused on planned and unplanned failures at the virtual machine or physical server level. A comprehensive availability strategy should also take into account the possibility of a failure of individual SAS processes running on the virtual machine. Similarly, while failover can ensure that failed virtual machines are restarted; this document does not include discussion about any cross-server orchestration that might be necessary due to dependencies between SAS servers. The most notable example of a cross-server dependency is centered on the SAS Metadata Server. Virtually all of the SAS servers depend on the availability of the SAS Metadata Server. When it is not running or not available for some reason, the rest of the SAS deployment can become unusable even if the processes continue to execute.

SAS Software	*VMware*
Foundation SAS® (SAS 9.2M3) *SAS® Enterprise BI Server* • *SAS Enterprise BI Clients 4.3* *SAS Enterprise Data Integration Server* *SAS® Enterprise Miner™ version 6.2* *SAS® Enterprise Guide® 4.3*	*VMware vSphere* • *ESX 4.1.0 (hypervisor)* • *HA (failover)* • *vMotion (live migration)* • *Client 4.1.0 (desktop admin client)* *VMware vCenter Server 4.1.0 (management platform)*

Table 1: Software Technology Used

VMware

The VMware products and components used during this project include the following:

VMware ESX – This component, part of the larger, VMware vSphere product suite, provides the hypervisor that enables servers that have been “virtualized” run as virtual machines on physical host servers. The hypervisor mediates the request for hardware resources from the virtual machine to the underlying physical host server. VMware recommends moving to its newer VMware ESXi hypervisor. This provides the same functionality and performance as VMware ESX. However, it was re-architected to eliminate a Linux OS instance used by VMware ESX for certain management activities. Eliminating this reduces the hypervisor footprint significantly. However, the behavior of VMware HA or VMware vMotion should be identical across the two technologies.

VMware High Availability (VMware HA) – This component of the VMware vSphere product suite provides high availability capabilities. When VMware HA is enabled, it monitors the health of both the underlying physical host server and the virtual machines running on those servers. If it detects a failure of either the hardware or the guest operating system (i.e. the VM), it restarts the virtual machine on another server in the cluster. This effort leveraged VMware HA to handle unplanned failover.

VMware vSphere Client – This component provides a desktop client application for configuring, monitoring and controlling the entire vSphere environment. Using the client, administrators can create, modify, and delete virtual machines. They can also manage those virtual machines; starting them, stopping them, and monitoring their performance. The client allows administrators to configure the networking and storage available to the virtual machines.

VMware vMotion – This is another component of the VMware vSphere product and provides live migration capabilities. Live migration is the process of transparently moving a running virtual machine to another server without interrupting on-going processing or the availability of services to the user. This effort leveraged VMware vMotion to handle planned failover.

VMware vCenter – This component forms the management platform sitting on top of the vSphere virtualization platform.

SAS Software

This project focused on a deployment of the Platform for SAS Business Analytics configured to support SAS® Enterprise BI Server, SAS® Enterprise Data Integration Server and SAS® Enterprise Miner™. The platform is composed on a number of SAS servers grouped, logically, into multiple tiers. These servers are logical servers rather than pieces of computing hardware. They represent long-running processes that work together as the users interact with the SAS clients. These logical servers are responsible for accessing the data, performing the analytical processing and rendering the results requested by the users. The servers can be grouped together based on the role that they play in a deployment. The most common approach, and the approach used during this project, is to group the servers into the four tiers as shown in Figure 1.

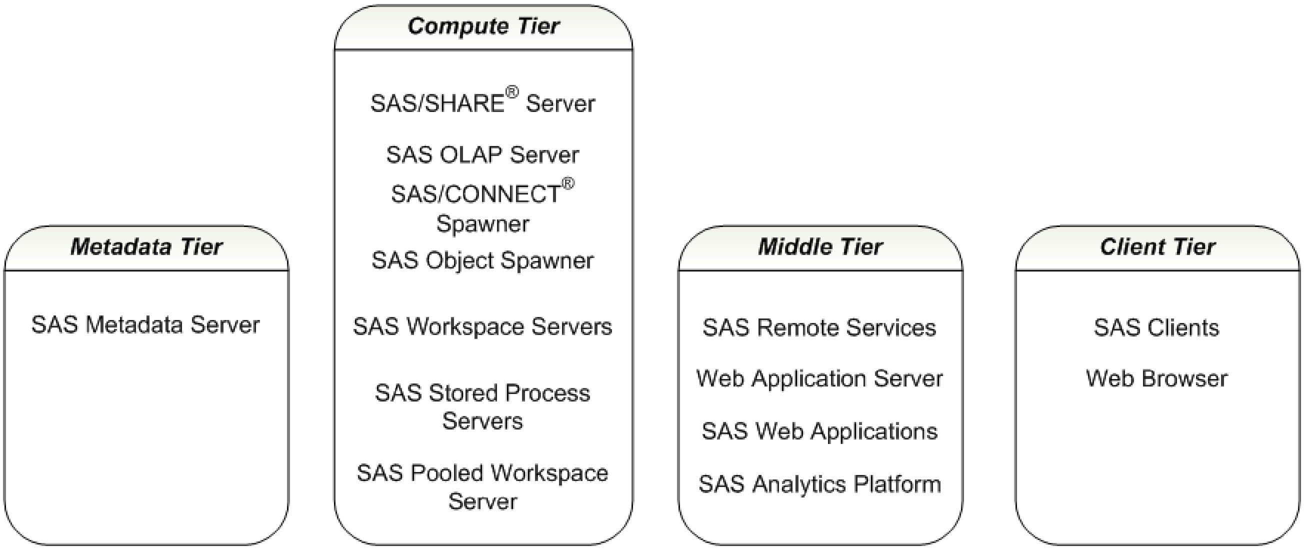


Figure 1: A Typical SAS Architecture

It is important to keep in mind that these SAS servers are logical servers; there is no requirement that each be deployed to their own physical (or virtual) server. Some deployments place all of these servers on a single machine; other deployments make use of a number of server machines.

The Testing Environment

The SAS deployment was designed as a multi-machine deployment, that is, the different tiers were installed onto different virtual machines. Although SAS software can be deployed onto a single server, a multi-machine deployment is more representative of how customers most commonly use SAS in a production environment. To allow testing of how each individual tier behaved, each was deployed onto its own operating system instance (i.e. to its own virtual machine). Figure 2 shows how the various SAS tiers were distributed across the virtual machines.

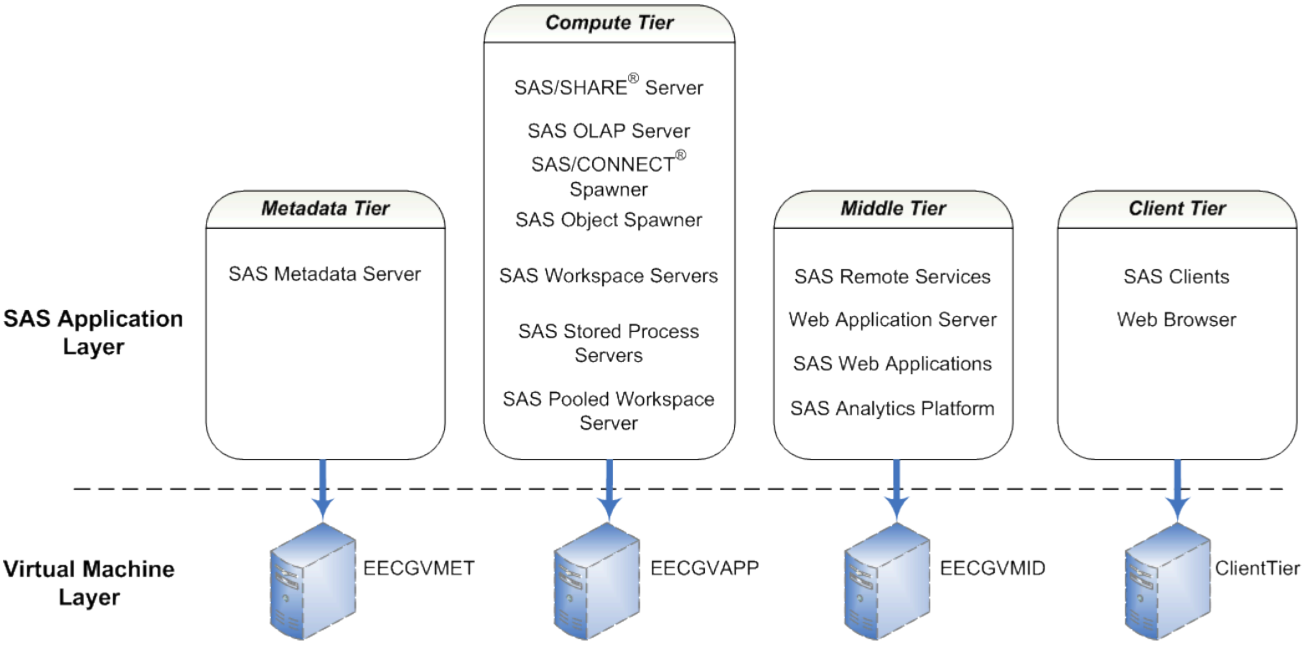


Figure 2: The Distribution of SAS Tiers onto Virtual Machines

Beneath this layer of virtual machines, the test environment consisted of a set of physical servers, which served as the virtual hosts. The physical host environment used in the testing environment was composed of Cisco UCS blade servers, two at first, and later expanded to include two additional (and more powerful) servers for a total of four servers. All of the servers were configured on the same network sub-net and all were attached to the same IBM XIV® SAN system for storage.

Refer to Table 2 for details of the physical environment. Table 3 describes the configuration of the virtual machines.

Physical Servers (Virtual Hosts)	*Server 1 (10.9.0.21)*	*Model: Cisco UCS B200 M1* *Processors: 2 Quad-Core Intel Xeon* *RAM: 24GB*
	Server 2 (10.9.0.22)	*Model: Cisco UCS B200 M1* *Processors: 2 Quad-Core Intel Xeon* *RAM: 24GB*
	Server 3 (10.9.0.25)	*Model: Cisco UCS B440 M1* *Processors: 4 8-Core Intel Xeon processors* *RAM: 132 GB*
	Server 4 (10.9.0.26)	*Model: Cisco UCS B440 M1* *Processors: 4 8-Core Intel Xeon processors* *RAM: 132GB*
Storage	*IBM XIV Storage System*	*1TB of storage, from multiple LUNs, surfaced to VMware as a single data store*
Networking		*All servers were connected to each other via the 10GbE network built into the Cisco UCS environment.* *They were connected to the SAN environment through a 4Gb Fibre Channel network.*

Table 2: The Physical Test Environment

Figure 3 shows all three levels of the testing environment. Throughout the testing, different combinations of virtual machines ran on different physical host servers. The general approach was to isolate the specific SAS tier (and, therefore, the specific virtual machine) being tested onto one physical host server and place the others on one or more of the other servers. For example, in Figure 3 the Metadata tier (running in the EECGVMET virtual machine) is running on Server 3, while all of the other virtual machines are running on different physical host servers. After a failover event makes Server 3 unavailable, the EECGVMET virtual machine is restarted on Server 4, the same physical host on which the client tier virtual machine is running.

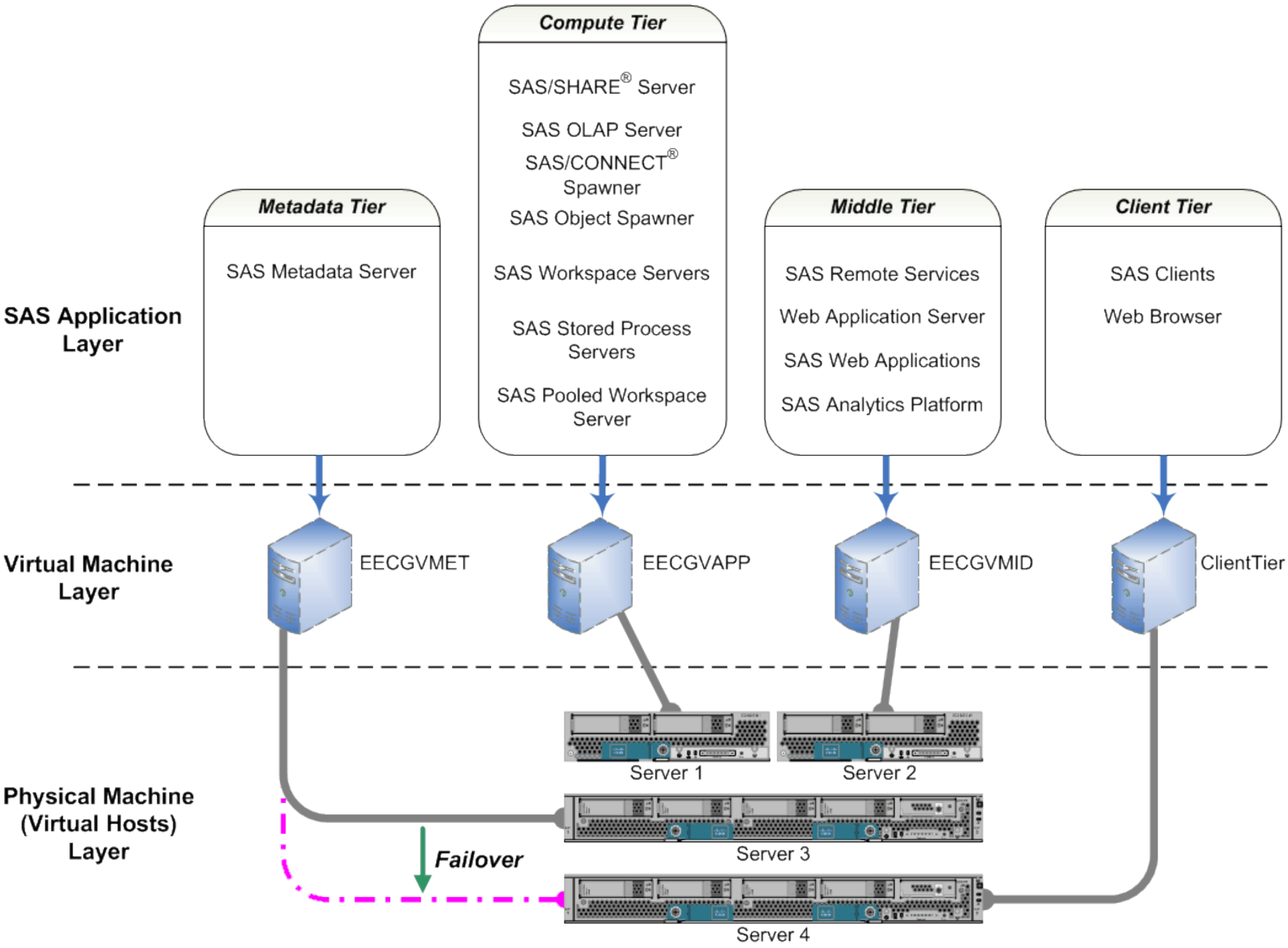


Figure 3: The Test Environment

Virtual Machine Name	*Resources*	*Operating System*	*Purpose*
EECGVMET	*4 vCPU* *8GB RAM*	*Microsoft Windows Server 2008*	*Metadata tier*
EECGVAPP	*4 vCPU* *12GB RAM*		*Compute tier*
EECGVMID	*4 vCPU* *8GB RAM*		*Middle tier*
Client tier	*4 vCPU* *8GB RAM*		*Client tier*

Table 3: Virtual Machine Configurations and Roles

Configuration

Configuring SAS Software

The SAS software installation and configuration processes in a virtual environment are no different from those used with physical servers environment. There are no special steps or considerations needed. From an installation and configuration standpoint, there is no difference between working with virtual and physical servers. Although the server is a virtual server, this does not eliminate the need to understand how it is used and to size it appropriately for the intended deployment. A benefit of virtualization is that it allows administrators to change the hardware resources available to the virtual server easily. However, administrators should not use this as a justification to skip critically important planning steps such as sizing.

Configuring VMware vMotion

This effort leveraged two different VMware technologies to enable failover in different situations. The first, VMware vMotion, was used to handle planned failover. VMware vMotion allows a running virtual machine to be moved from one physical host to another physical host with no loss of availability. In this effort, vMotion was looked at as a technique for handling planned failover. This could be useful when administrators need to move a portion of the SAS deployment of one server to another, perhaps due to the need to perform maintenance on the original server. With its promise of no loss of availability, vMotion could be very helpful to allow system maintenance without impacting the users' ability to do their work.

The vMotion process involves three steps:

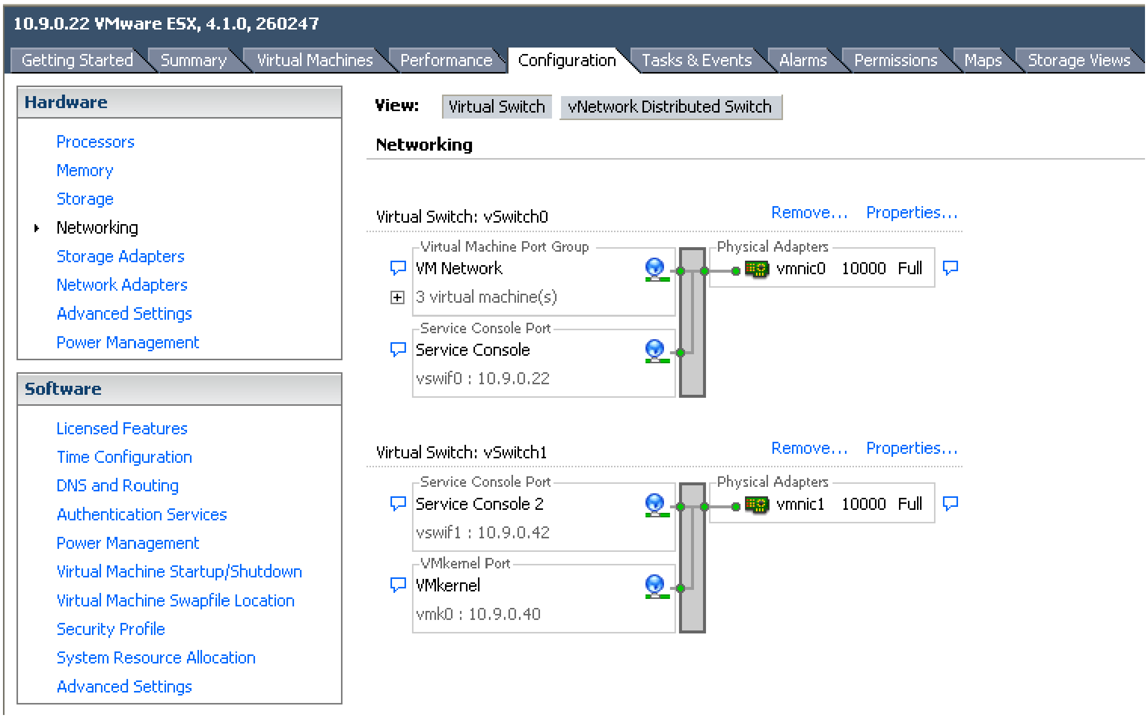
1. The current state of the source virtual machine is captured.
2. This captured state information is copied to a new virtual machine on the target host server.
3. The state of the original is checked again. If all changes have been copied to the target virtual machine, the virtual machine running on the original server is turned off and the virtual machine on the new host becomes the active virtual machine. If the state of the original virtual machine has changed, the process is repeated until the states of the two virtual machines have been synchronized.

If the virtual machines on the original and target host servers cannot be synchronized within a certain amount of time, or if some other error was encountered, VMware cancels the vMotion migration. It is important to understand that the virtual machine is available to users throughout this entire process; there is no loss of availability.

The following checklist outlines some key requirements for configuring an environment to leverage vMotion. Refer to the VMware Administration guide for a detailed description of the requirements and configuration available at: **http://www.vmware.com/pdf/vsphere4/r41/vsp_41_dc_admin_guide.pdf**

- Each host must be licensed for vMotion
- Each host must be stored on shared storage that is available to all of the hosts in the cluster
- Each host must meet the following networking requirements:
    - Have at least two network adapters.
    - Be connected to the other vMotion-enabled hosts via a Gigabit Ethernet connection and use this connection for vMotion traffic.
    - Have access to the same subnets.

If the requirements are met, configuring vMotion is a straightforward process using the vSphere Client. The network configuration for each of the hosts should be modified to add a VMkernal network connection; this is a special connection used by hosts for vMotion traffic. The host network configuration is accessed from the **Configuration** tab for the individual hosts and selecting **Networking** from the **Hardware** menu. Figure 4 shows the network configuration for one of the hosts used in this effort. Notice that two virtual switches have been defined and each is linked to a physical adapter. The first, ***vSwitch0***, is configured to handle virtual machine network traffic (through the ***VM Network*** port group) and management traffic (through ***Service Console*** port). The second switch, ***vSwitch1***, is configured to handle management traffic (through the ***Service Console 2*** port) and the vMotion traffic (through the ***VMkernal*** port group).



Virtual Switch: vSwitch0

Virtual Machine Port Group: VM Network — 3 virtual machine(s)

Service Console Port: Service Console — vswif0 : 10.9.0.22

Physical Adapters: vmnic0 10000 Full

Virtual Switch: vSwitch1

Service Console Port: Service Console 2 — vswif1 : 10.9.0.42

VMkernel Port: VMkernel — vmk0 : 10.9.0.40

Physical Adapters: vmnic1 10000 Full

Figure 4: The Network Configuration for a Host Configured for vMotion

Once the configuration is complete, migrating virtual machines using vMotion can be accomplished by clicking on the icon for the virtual machine to be migrated, within the vSphere Client application, and selecting the **Migrate…** option from the right-mouse button pop-up menu. Migration with vMotion can also be accomplished using drag-and-drop from within the vSphere Client application by dragging the virtual machine icon and dropping it on the icon for the new host. In either case, the **Migrate Virtual Machine** wizard dialog box is displayed. The wizard checks that the new host is compatible with the virtual machine being moved, and if so, prompts the administrator to confirm the migration.

Configuring VMware HA

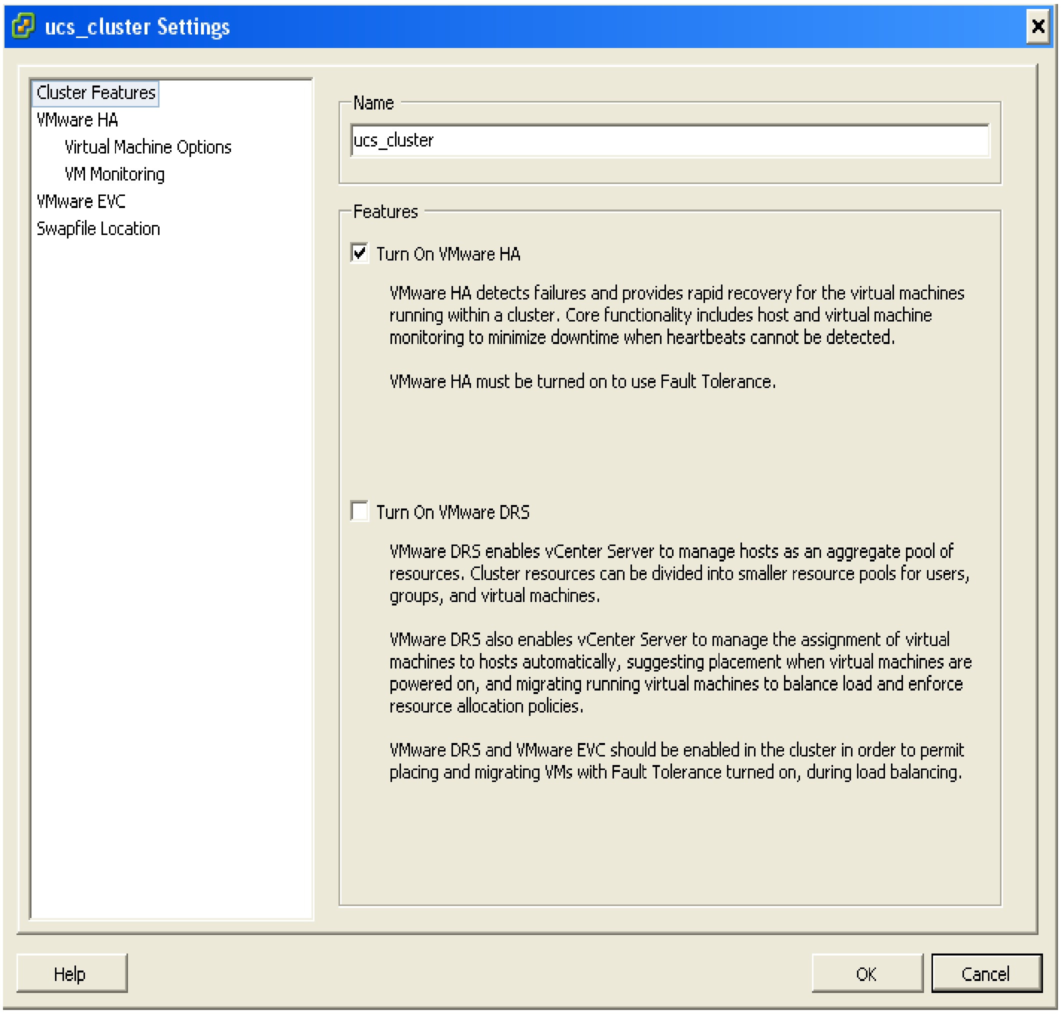
The second VMware technology, VMware HA, was used to handle unplanned failover. This form of failover comes into play when dealing with a true failure of a physical (host) server or the guest OS instance that makes up the virtual machine. To support VMware HA, multiple physical servers are configured into a logical unit called a cluster. The hardware resources of the physical servers within the cluster are managed together. (In addition to supporting VMware HA, clusters can also be used to enable VMware DRS; a feature of vCenter Server that balances server workloads across the physical servers within the cluster.) VMware HA is enabled at the cluster level and can be configured through the vSphere client application. Each physical server running in the cluster periodically emits a “heartbeat” to tell the other hosts that it is up and running. Similarly, each virtual machine emits heartbeats to its physical host server as well. When heartbeat messages from a physical server or virtual machines are not received as expected, failover is triggered and VMware HA restarts the failed virtual machines on a new physical server.

The following checklist, based on VMware documentation, details the key requirements of a VMware HA environment. The VMware documentation provides a full explanation of the VMware HA configuration process and the various configuration options.

- A cluster must contain at least two hosts; each of which must be licensed for VMware HA, have a unique host name and have VMware Tools installed.
- All hosts must have a static IP address that persists across reboots and DNS must be configured to resolve the short host name to that IP address.
- All hosts must have access to the same networks and the same datastores. Virtual machines should be located on shared storage that is available to the other hosts in the cluster rather than be placed on storage local to a single host.

The following information provides a high-level overview of the configuration steps or configuration settings necessary to configure VMware HA. The options described below are organized by the panel on which they appear within the **Edit Settings** dialog box available through **Summary** tab for the cluster in the vSphere client. Refer to the VMware documentation for a fuller discussion of the various configuration options and settings.

The Cluster Features Panel



ucs_cluster Settings

Name

ucs_cluster

Features

☑ Turn On VMware HA

VMware HA detects failures and provides rapid recovery for the virtual machines running within a cluster. Core functionality includes host and virtual machine monitoring to minimize downtime when heartbeats cannot be detected.

VMware HA must be turned on to use Fault Tolerance.

□ Turn On VMware DRS

VMware DRS enables vCenter Server to manage hosts as an aggregate pool of resources. Cluster resources can be divided into smaller resource pools for users, groups, and virtual machines.

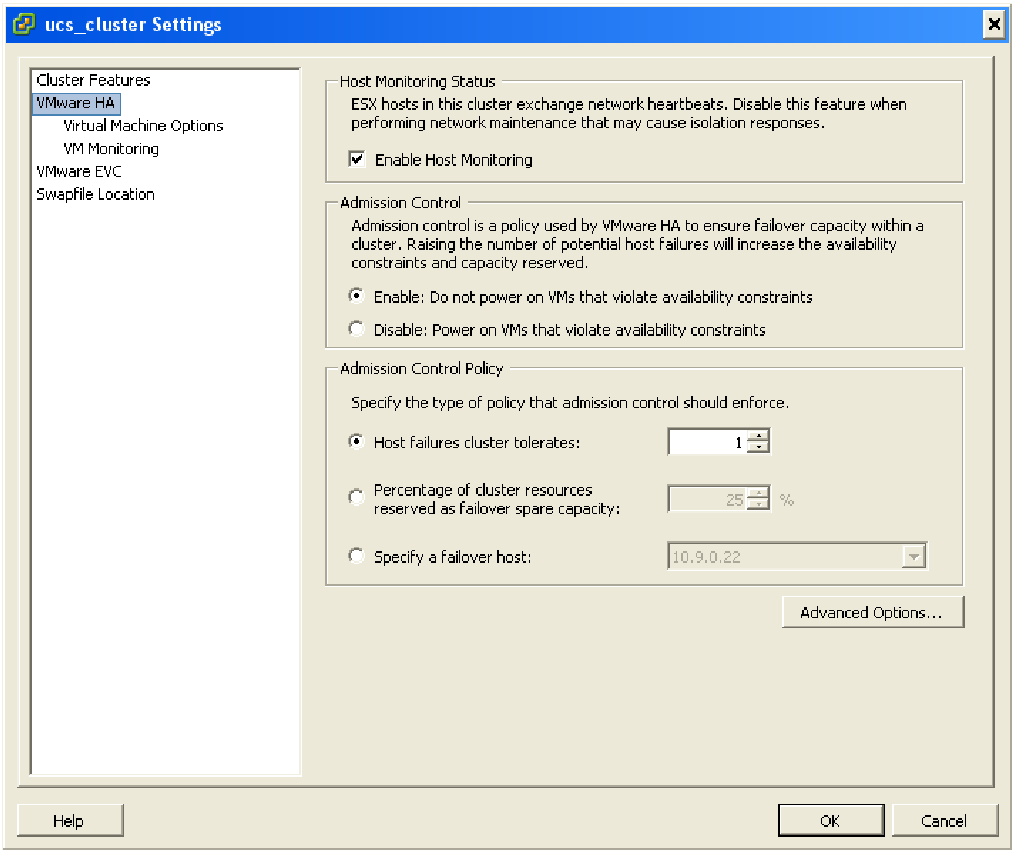
VMware DRS also enables vCenter Server to manage the assignment of virtual machines to hosts automatically, suggesting placement when virtual machines are powered on, and migrating running virtual machines to balance load and enforce resource allocation policies.

VMware DRS and VMware EVC should be enabled in the cluster in order to permit placing and migrating VMs with Fault Tolerance turned on, during load balancing.

Enable VMware HA by checking the **Turn On VMware HA** check box

Figure 5: The Cluster Features Panel

The VMware HA Panel



ucs_cluster Settings

Host Monitoring Status

ESX hosts in this cluster exchange network heartbeats. Disable this feature when performing network maintenance that may cause isolation responses.

☑ Enable Host Monitoring

Admission Control

Admission control is a policy used by VMware HA to ensure failover capacity within a cluster. Raising the number of potential host failures will increase the availability constraints and capacity reserved.

● Enable: Do not power on VMs that violate availability constraints

○ Disable: Power on VMs that violate availability constraints

Admission Control Policy

Specify the type of policy that admission control should enforce.

● Host failures cluster tolerates: 1

○ Percentage of cluster resources reserved as failover spare capacity: 25 %

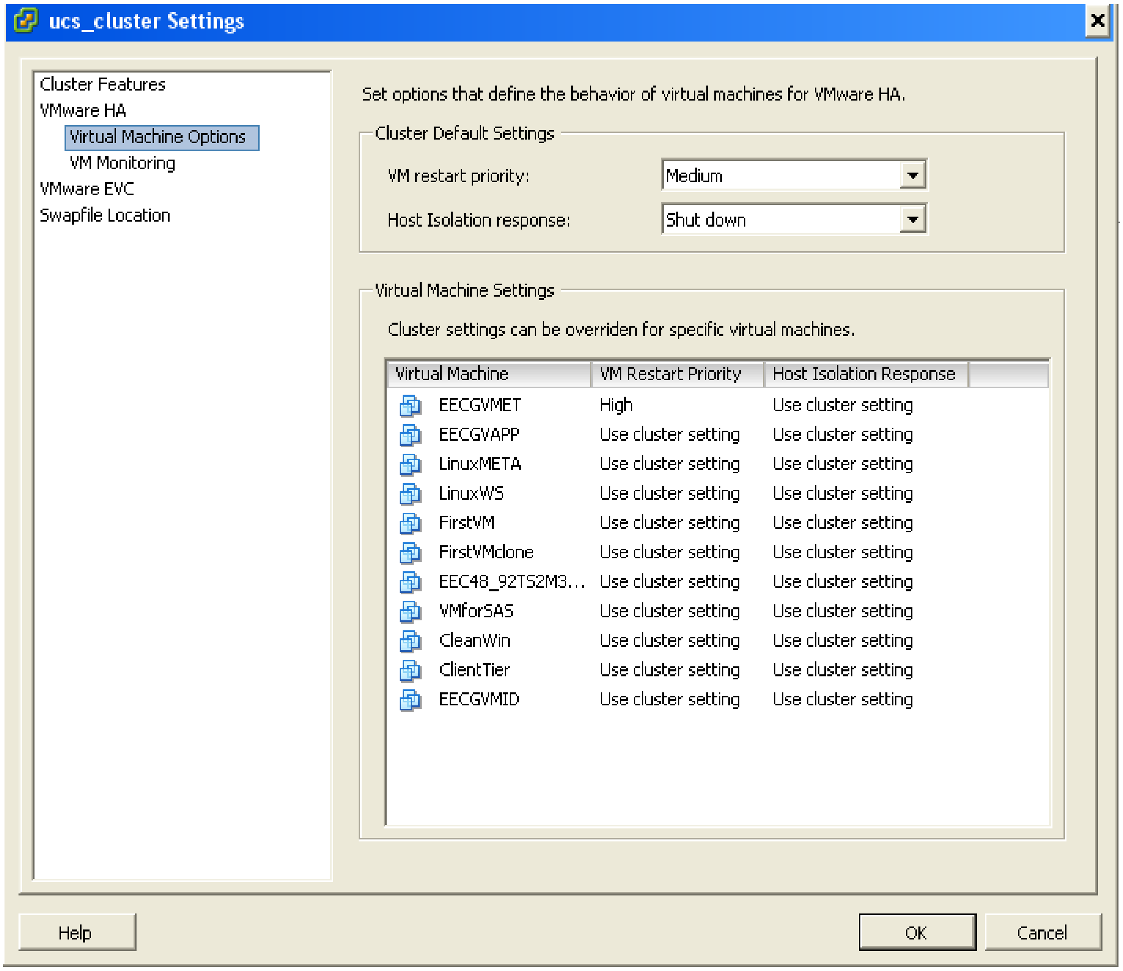
○ Specify a failover host: 10.9.0.22

Figure 6: The VMware HA Panel

Check the **Enable Host Monitoring** check box. When checked, virtual hosts send out “heartbeats” which the other hosts monitor to determine whether a host has failed and failover needs to be initiated. Disabling this feature effectively turns off VMware HA

The options for **Admission Control** and **Admission Control Policy** allow administrators to reserve some of the cluster capacity, in terms of CPU and memory, to handle failovers or, to identify a specific host on which all failed virtual machines should be restarted. Deciding on an appropriate admission control policy depends on many factors and administrators should refer to the VMware documentation for more information.

The Virtual Machine Options Panel



Set options that define the behavior of virtual machines for VMware HA.

Cluster Default Settings

VM restart priority: Medium

Host Isolation response: Shut down

Virtual Machine Settings

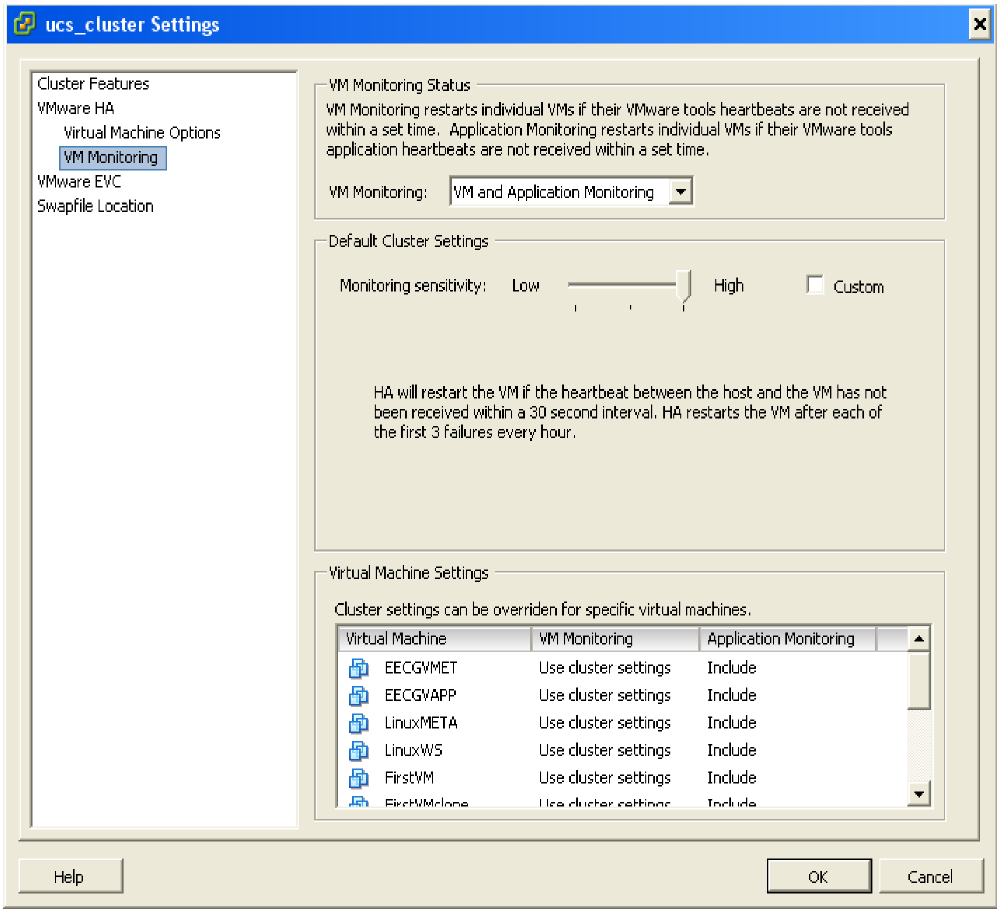
Cluster settings can be overriden for specific virtual machines.

Virtual Machine	VM Restart Priority	Host Isolation Response
EECGVMET	High	Use cluster setting
EECGVAPP	Use cluster setting	Use cluster setting
LinuxMETA	Use cluster setting	Use cluster setting
LinuxWS	Use cluster setting	Use cluster setting
FirstVM	Use cluster setting	Use cluster setting
FirstVMclone	Use cluster setting	Use cluster setting
EEC48_92TS2M3...	Use cluster setting	Use cluster setting
VMforSAS	Use cluster setting	Use cluster setting
CleanWin	Use cluster setting	Use cluster setting
ClientTier	Use cluster setting	Use cluster setting
EECGVMID	Use cluster setting	Use cluster setting

Figure 7: The Virtual Machine Options Panel

The **VM Restart Priority** setting determines the order in which virtual machines are re-started after failover. By default, all virtual machines have the same priority and are all restarted simultaneously. However, it is possible to assign a priority to individual virtual machines. During this project, the use of this feature was investigated to determine whether it could help guarantee that the various SAS tiers started in the right order. Testing results are outlined in the **“Results”** section.

The VM Monitoring Panel



VM Monitoring Status

VM Monitoring restarts individual VMs if their VMware tools heartbeats are not received within a set time. Application Monitoring restarts individual VMs if their VMware tools application heartbeats are not received within a set time.

VM Monitoring: VM and Application Monitoring

Default Cluster Settings

Monitoring sensitivity: Low High □ Custom

HA will restart the VM if the heartbeat between the host and the VM has not been received within a 30 second interval. HA restarts the VM after each of the first 3 failures every hour.

Virtual Machine Settings

Cluster settings can be overriden for specific virtual machines.

Virtual Machine	VM Monitoring	Application Monitoring
EECGVMET	Use cluster settings	Include
EECGVAPP	Use cluster settings	Include
LinuxMETA	Use cluster settings	Include
LinuxWS	Use cluster settings	Include
FirstVM	Use cluster settings	Include
FirstVMclone	Use cluster settings	Include

Figure 8: The VM Monitoring Panel

The **VM Monitoring Status** setting identifies the types of heartbeats VMware HA monitors to determine whether a virtual machine has failed. This is an additional level of protection that initiates failover if the *virtual machine*, rather than the *physical host server*, fails. Furthermore, it is possible for applications to leverage a VMware API so that it can emit heartbeats for VMware to monitor as well. However, this application level capability has not been implemented for any of the SAS components. Therefore, if the virtual machine is hosting only SAS components, the setting of "***VM Monitoring Only***" is adequate. This will allow VMware to detect initiate failover in case of a failure of the virtual machine.

The Default Cluster Settings determine how often VMware checks for heartbeats from the virtual machines and the threshold (that is, the number of missed heartbeats) for a virtual machine to be considered failed. In addition to three predefined sets of settings for monitoring sensitivity (HIGH/MEDIUM/LOW),

VMware allows administrators to create their own customized set of settings. To prevent falling into an endless loop of failing and restarting the same virtual machines on failed servers, VMware HA keeps track of how often each virtual server has failed. If the number of failures during a specific time window exceeds this threshold, a failed virtual machine will not be restarted. Administrators are free to customize these settings as well.

Test Scenarios and Methodology

For each of the SAS server tiers the same testing scenarios were used. The first test scenario subjected each tier to a planned failover. This was accomplished by initiating a vMotion migration using the vSphere Client application. Throughout the testing process, including during the actual vMotion migration, the servers in the tier were being actively used. This was accomplished through a variety of simulated user activities using multiple SAS software applications. For example, during the testing of the SAS compute tier, there was activity by users using SAS® Web Report Studio (requiring the involvement of the Pooled Workspace Servers and or the Stored Process Server) and SAS Enterprise Guide (involving the unpooled workspace server and SAS OLAP Server). During the test for the middle tier, SAS Web Report Studio users were active throughout the testing period.

The second test scenario focused on unplanned failover. To simulate a failure of a host, the host server was rebooted while the various SAS servers were busy. The server reboot was an immediate reboot without any quiescing of running processes. In this scenario, users were actively working through a variety of SAS client interfaces when the simulated host failure was triggered. In a second round of testing of this scenario, the host server was powered down rather than rebooted. This was done to more realistically simulate the failure of the server. This also eliminated the possibility that virtual machines would be given any opportunity to shutdown cleanly. Here, too, users were actively working when the server was powered down.

All test scenarios were repeated a number of times to confirm the results obtained were representative. Although there was realistic user activity during the execution of the tests, the systems were only lightly used. The user activity was performed by testing staff rather than through the use of automated testing tool. After each test scenario, testing staff confirmed that the SAS servers were returned to a healthy

Results

Scenario One: Planned Failover

The results from the planned failover test scenario were positive. All of the SAS servers, across all of the tiers, handled the process of vMotion migration well. All of the SAS servers continued to work during and after the vMotion migration. The end-user SAS client applications maintained their connections to the SAS Metadata Server and the other SAS servers. The vMotion migration of the individual SAS tiers was completed very quickly each time, generally taking less than a minute. No response time metrics were captured on the end-user client side. However, any delays appeared negligible to the testing staff. This is in keeping with the results of an earlier study of the effect of vMotion on a running SAS deployment. In that study, summarized in the paper “VMWare VMotion Migration of the Platform for SAS® Business Analytics”, vMotion was found to impact response times but only slightly (increasing it by a few seconds) and only for the short period while the migration was actually taking place.

Scenario Two: Unplanned Failover

In the unplanned failover scenario, VMware HA worked as expected. In all cases, VMware HA was able to detect the failure of a physical host server and quickly restart the affected virtual machines on healthy servers. However, as expected (and unlike the planned failover scenario), the impact of this restart is not transparent to end users. In most cases the end user's work flow was interrupted and there was a loss of unsaved work. The user experience varied depending on the SAS client used and which server failed. For example, after a failure of the SAS compute tier, SAS® Enterprise Guide® users were informed that they need to reconnect to the server and any temporary data sets were lost. However, the SAS Enterprise Guide client session remained open and the end user was able to rerun the entire process flow, recreating the lost data or resetting any macro variables. On the other hand, end users working with the SAS Web applications, such as SAS Web Report Studio lost their current work and needed to log in again when VMware HA was forced to restart a compute or middle tier server. In the test environment, all of the SAS components were configured as Windows services that were started automatically when the virtual machine was restarted. This allowed these virtual machines to resume providing the necessary SAS functionality after the restart without administrator intervention.

The effects of a failure of the metadata tier server were more extensive. Virtually all of the SAS components are dependent on the SAS Metadata Server. Therefore, a failure of the virtual machine on which it is running generally requires a restart of these dependent components, even if these other components operate on other virtual machines. Even if some components appear to continue operating, the recommended best practice is to always restart the other components after a restart of the SAS Metadata Server. This is a result of the SAS architecture and is unrelated to VMware HA. Organizations need to consider these dependencies when developing their availability strategies and processes.

During the testing of this scenario with the metadata tier, VMware HA behaved as expected. It detected the failure and restarted the metadata tier virtual machine on a new host server. In all but a few tests, the SAS Metadata Server (which was configured as a Windows service) was restarted when the virtual machine was restarted. However, there were cases when the SAS Metadata Server was not able to be restarted. These were a result of the metadata repository being corrupted during the server failure. The only remedy for this was to restore the metadata repository from an earlier backup. It was not possible to identify the specific cause of this corruption or the exact conditions under which it would occur. A limited amount of testing using the same test scenario and methodology was done in a comparable Linux environment. In those tests, there were no cases of the metadata repository becoming corrupted after a failover event. It remains unclear whether the corruption seen in the original environment was a result of the behavior of the Windows operating system, or the combination of the Windows operating system and VMware virtualization technology. In any case, a failure of the SAS Metadata Server is a very significant event, regardless of whether it is in a virtualized or non-virtualized environment. Organizations should be diligent in performing regular backups of their metadata server and their entire environment. This will allow them to recover more quickly from a corrupted metadata repository should one be encountered.

VMware HA behaved as expected, it is intended to ensure that specific virtual machines are up and running. It is not designed to, nor is it positioned as a way, to eliminate the impact of a server failure on the end users. VMware does have technology, called VMware FT that provides truly fault tolerant protection, eliminating the service interruption end users would experience under VMware HA. The suitability of VMware FT for protecting SAS deployments is discussed in the Related Features and Technologies section.

Related Features and Technologies

VM Restart Priority (VMware HA)

The VMware HA technology allows administrators to assign a restart priority to individual virtual machines. When multiple virtual machines fail, they are restarted based on this priority. During this effort, the question of whether this feature could be used to ensure that the various SAS tiers were restarted in the proper order was investigated. Because of the difficulty in dealing with cross server dependencies, this feature would theoretically be most useful in two specific situations. The first is if all of the SAS tiers were running on the same physical host. The second is if all of the virtual machines housing the SAS tiers failed simultaneously. During the testing, the middle tier virtual machine was assigned a lower priority than the SAS Metadata Tier virtual machine in an attempt to ensure that the SAS Metadata Server was started before the web application server was restarted. Unfortunately, the testing was unsuccessful. This was surprising because the test scenario closely mirrors the sample use-case described in the VMware documentation for this technology.

Further discussions with VMware clarified the situation. Currently, the **VM Restart Priority** settings are applied only when there are limited resources available. When there are adequate resources available in the cluster to start all of the failed virtual machines, VMware HA restarts them all in parallel. However, if the cluster is resource-constrained, that is, if the resources available to the cluster are less than that needed for all of the failed virtual machines, the virtual machines are restarted based on this setting. This ensures higher priority virtual machines are restarted and that lower priority virtual machines might not be restarted. VMware has indicated that a future release will take these settings into account in all cases, even if the cluster is not resource constrained. In any case, even if the feature restarted the virtual machines in the proper order that is not a guarantee that the SAS components deployed on those VMs would also be started in the proper order. There would still likely be the need to develop and or deploy some sort of scripting logic to ensure everything started in the proper order. Therefore, these settings are not suitable for controlling the restart of virtual machines in a SAS deployment.

VMware Distributed Resource Scheduler (DRS)

VMware DRS is a component of VMware vSphere that allows administrators to control how resources and virtual machines are allocated across a cluster. VMware DRS monitors the resource utilization across the virtual machines and physical hosts. It uses this information to load-balance the virtual machines across the host servers or to make recommendations to administrators on more efficient use of available resources. It also has power management capabilities that allow it to consolidate virtual machines to fewer host servers and identify host servers that can be powered down. Later, if demand grows, it can power up those servers to handle the increased demand. Organizations that use VMware HA often make use of this technology as well. This allows VMware vCenter to restart virtual machines on the least used physical hosts or to take an organization’s resource policies into account. Although VMware DRS was not included in this effort, its use should not alter the results observed in this project.

VMware Fault Tolerance (FT)

VMware FT is another technology that is intended to prevent the failure of a server from taking virtual machines off-line. Rather than simply restarting failed virtual machines, VMware FT maintains a second virtual machine running at the same time. This shadow copy is kept fully synchronized with the primary virtual machine. If the first virtual machine fails, VMware FT redirects all activity away from the failed server to this shadow server. This is intended to provide continuous, uninterrupted availability of key services. This could be very useful for components such as the SAS Metadata Server or the SAS Remote Services application. The failure of either of these can require the restart of several other components and a significant interruption in the availability of the platform of SAS business analytics. The current version of VMware FT is limited to protecting virtual machines with a single virtual CPU. Since the standard recommendation is to deploy SAS components to servers with multiple processors, VMware FT (with the current limits) is not suitable for use with SAS software.

Conclusion

This paper described recent research into the use of two VMware technologies for providing failover protection to a SAS deployment. VMware vMotion, with its ability to move a running virtual machine to a different server seamlessly and without interruption, can be a powerful tool for handling planned failover. It allows administrators to migrate virtual machines housing any of the SAS server tiers to new servers when the original server needs maintenance or an upgrade. The second technology, VMware HA, provides a good solution for cases of unplanned failover, especially for the compute and middle tiers. By automatically restarting failed virtual machines, it can help administrators minimize downtime. In multi-machine deployments, such as the test environment described in this paper, it can serve as an important part in a more sophisticated high-availability strategy. In single machine deployments, without the need to coordinate the starting of SAS servers across multiple virtual machines, it could, by itself, play a larger role in maximizing availability. The testing described in this document demonstrated that both technologies work well with SAS software.

References and Additional Information

SAS

SAS Institute Inc. 2010 *VMWare VMotion Migration of the Platform for SAS® Business Analytics.* Cary, NC: SAS Institute Inc. Available internally and upon request

VMware

VMware, Inc. 2010 *Introduction to VMware vSphere* Palo Alto, CA: VMware, Inc. available at: **http://www.vmware.com/pdf/vsphere4/r40/vsp_40_intro_vs.pdf**

VMware, Inc. 2010 vSphere Datacenter Administration Guide Palo Alto, CA: VMware, Inc available at: **http://www.vmware.com/pdf/vsphere4/r41/vsp_41_dc_admin_guide.pdf**

VMware, Inc. 2010 vSphere Availability Guide Palo Alto, CA: VMware, Inc. available at: **http://www.vmware.com/pdf/vsphere4/r41/vsp_41_availability.pdf**