VMware® Infrastructure 3

Advanced Technical Design Guide

~and~

Advanced Operations Guide

Two books in one!

Ron Oglesby
Scott Herold
Mike Laverick

Chapter 10 - Recovery and Business Continuity

Not long after VMware and virtualization started to take off for server consolidation projects, people started to truly realize the portability and recoverability aspects of virtualization. In fact, VMware started to win awards for "Disaster Recovery Product of the Year" without actually being a disaster recovery product.

Recently, VMware has significantly modified its marketing message and has moved away from "Server Consolidation" and now focuses very heavily on "Disaster Recovery." While VI3 does provide a strong platform for recoverability, it is important to note that it is no magic bullet to simply toss a virtual infrastructure into an environment. It would be foolish to think that by simply installing a few ESX servers with VirtualCenter an organization has a recovery plan without additional effort.

In this chapter, we are going to dig beyond "Disaster Recovery" and address a larger issue that virtualization with VMware actually enables: "Business Continuity." By the end of this chapter it will be quite simple to see that Disaster Recovery is a very small part of the equation when providing highly available and recoverable systems.

What is Business Continuity?

Business Continuity, as we will address it in this book, comes down to two key points. The first aspect of business continuity is an organization's capability to meet or exceed application and system availability service levels. This is done through the use of several technologies that will be discussed in this chapter, including various methods of High Availability and Fault Tolerance.

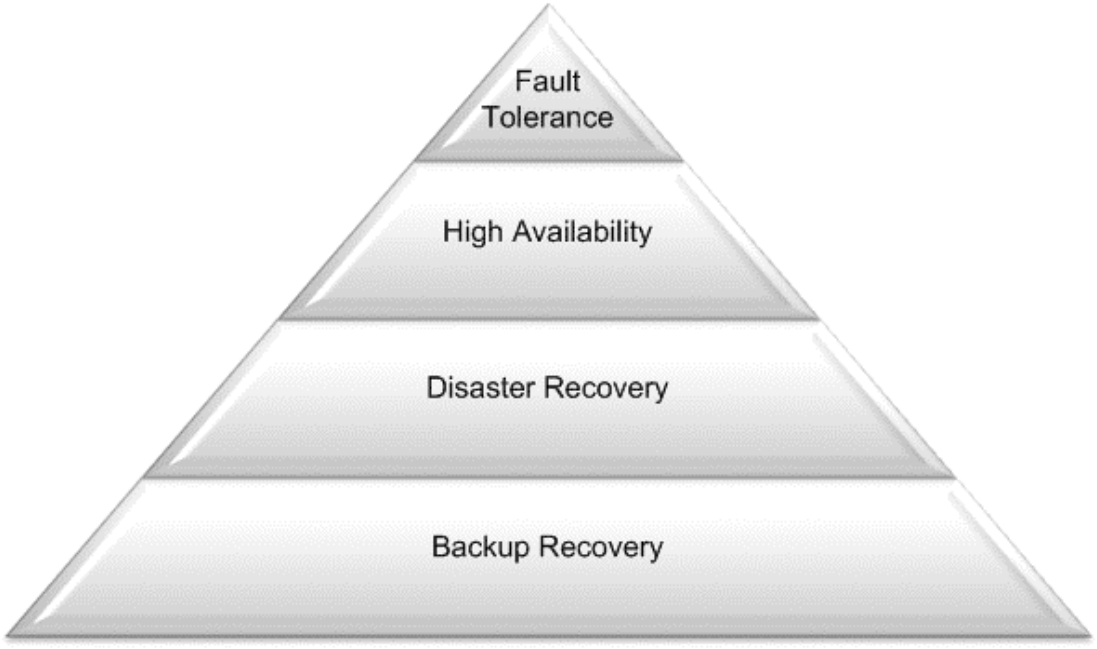
The second aspect to business continuity is an organization's ability to recover from failures, both minor and catastrophic, with minimum business impact. In order to meet this set of requirements a solid backup recovery and disaster recovery plan is required. This also shares some territory in the High Availability

space by the use of virtual machine replication, which has become quite popular over the last few months.

In short, this comes down to a simple premise of keeping your systems running and getting them back as fast as possible if something really bad were to occur.

In order to properly define business continuity, I would like to refer to the business continuity pyramid as defined by Vizioncore, Inc, one of the primary vendors in the backup and disaster recovery space for VMware.

Figure 10- 1: Business Continuity Pyramid



This pyramid dictates the four primary areas of business continuity on which this chapter focuses. The lower layers of the pyramid, which are the widest, dictate that more virtual machines in your environment will make use of the particular technology than at the higher layers, which get increasingly narrower.

For the purpose of this chapter we are going to begin by discussing the lowest layer of the pyramid, as this encompasses the largest number of systems, and work our way towards the top.

Backup Recovery

If we need to explain exactly what backup recovery is to the audience of this book, you probably have the wrong book in your hands. This process has been around just about as long as computers have as a way to recover data in the event of corruption or loss. Over time, this technology has matured to allow for entire system recovery, which we'll discuss further when we talk about disaster recovery later in this chapter.

As with just about anything virtualization, backup recovery can still be done using the same tried and true mechanisms that have been in place for ages, thanks to the portability and flexibility provided by the virtualization platform. Backup recovery can actually be further optimized beyond the capabilities provided by typical backup software.

Backup and Recovery Strategies

When dealing with designing a backup recovery strategy for your virtual infrastructure it is important to consider just exactly what it is you wish to backup. There are several components of the entire virtual infrastructure that require attention when creating your backup recovery plan. Each of these items does need to be addressed and managed individually.

ESX Hosts

The ESX host itself, or more accurately, the Service Console was a very critical piece of an ESX 2.X infrastructure. VMware has done several things in VI3 to deemphasize the importance of backing up various aspects of the SC, and we will mention those in context.

At this point we are assuming you are backing up the host server for recovery to like hardware. We are not discussing backing up any of the data that exists inside the virtual machines at this point. That will be discussed a little later in this chapter. There are three primary components of the ESX host that require consideration when developing your backup recovery plan for your host systems themselves: SC Configurations, VMware ESX Configurations, and virtual machine configurations. Since this isn't a database host or a file server (or sure shouldn't be) the backup job should be relatively small and easy to maintain.

Following this line of thought, you need to determine if you are going to backup your ESX servers at all. If you have a completely scripted installation of ESX, that covers all of your configurations, additional software packages for monitoring, security management, etc, and if you have decided to recover your virtual machines by storing their VMDK's on centralized storage, then you may have no need to backup your ESX servers at all.

When creating a backup plan for the SC, we want to temporarily forget about the configured virtual machines, as those will probably be recovered differently than an ESX host. Virtual machine backup and recovery has its own special set of considerations that we will look at in a minute.

Critical Components of the Service Console

As previously mentioned, there are three primary components that require backup and recovery consideration within the SC.

Service Console Configurations

VMware has taken significant steps and still continues to move forward on turning the ESX host into an appliance whose files are very static in nature. This does not mean there is no need to backup specific files that make up advanced or customized SC configurations. This is primarily focused in the area of security configurations, as this seems to be the most unique component across every organization. While every implementation will be different, the following files are those that we often find are modified to better enhance security or user access/control.

- /etc/profile
- /etc/ssh/sshd_config
- /etc/pam.d/ (Entire Directory)
- /etc/ntp/ (Entire Directory)
- /etc/ntp.conf
- /etc/passwd
- /etc/group

- /etc/sudoers
- /etc/shadow
- /etc/syslog.conf

VMware ESX Configurations

All configurations of ESX and the VMkernel are stored in the /etc/vmware directory. Like the Service Console configuration files, it can be time consuming to reconfigure ESX to its exact pre-failover state. By backing up the entire contents of the /etc/vmware directory you can easily return ESX to its original state after a base installation using the install media. VMware has enhanced the /etc/vmware directory in ESX3 by consolidating many log files and directories into a smaller number of objects.

Virtual Machine Configurations

The biggest of these modifications is that virtual machine configurations are no longer independently managed by the ESX host where they reside. All files that make up the configuration of the virtual machine are now located in the centralized storage infrastructure. This, of course, assumes there is a centralized storage infrastructure and that each host is not using local VMFS volumes to run virtual machines.

Due to this change in configuration file management; there is almost no point to backing up individual virtual machine configuration files from the Service Console. There are several methods to backup entire virtual machines, each of which backup the required configuration files. These methods will be discussed when we start talking about virtual machine backups.

At this point the plan is to backup and restore only configuration files. That means during the recovery of a server you will perform a base install of ESX using the original name and IP information, then restore that server's configuration to its previous state. With that in mind, the next step is to install the backup agent onto the console and configure it for your backups.

Which Backup Software Should You Use?

Since the SC is really a RedHat Enterprise Linux based system, you will need a backup agent that works for a Linux OS. Every major backup vendor that is worth trusting in an environment has Linux agents available, and these can be installed within the SC for use. If you manage to find a vendor that does not provide a Linux agent, or you do not want to spend the additional money for agents to place on ESX, it is possible to use a script to copy the required files and directories to another location.

Backup Schedule

Once all configurations are made and everything is running properly, changes are rarely made to the above mentioned files. Since the SC uses text configuration files for all settings, the amount of data being backed up is extremely small. For this reason, we recommend a nightly backup schedule on the files and directories that have changed. While nightly backups may not be necessary, the required capacity and amount of time to perform the job should have no impact on the performance of other backup jobs in the environment.

VirtualCenter

Backing up VirtualCenter is a relatively simple process. The only component of the VirtualCenter infrastructure that is important and difficult to duplicate without a backup is the database. Every configuration option, permission, and performance metric is stored within the database. VirtualCenter can be easily installed on a compatible operating system and returned to service by simply pointing at the existing database server.

Because of the importance of the data stored within the VirtualCenter database, it is critical that this be backed up using either a database compatible backup agent, or if VirtualCenter is running within a virtual machine, through an image level backup utility on a regular basis.

Virtual Machines

The most important asset to your virtual infrastructure is by far the virtual machines that are being run. Like just about any other system in the environment,

your virtual machines probably need to be backed up. Traditionally this has been performed by an organization through the use of an enterprise backup agent from any number of vendors. Through the use of virtualization, we have the opportunity to enhance this process and better enable data recovery through several technologies.

Remember, in this section we are discussing the normal backup and recovery strategies for a guest virtual machine. The idea here is that the strategies are for "normal" backup and restorations and not strategies for disaster recovery. We should note that you may wind up using the same recovery procedures for DR that you use on a daily basis, but we will not make that our primary focus for this section.

There are two basic approaches to perform virtual machine backups. First, you can treat the virtual machines as you do any physical server in your environment and leverage processes inside the operating system to perform backups. The alternative to this method takes advantage of the portability of the virtual machine and captures an image of the virtual machine VMDK and configuration files. Of course, some of the best backup recovery designs use the best that each of these technologies has to offer.

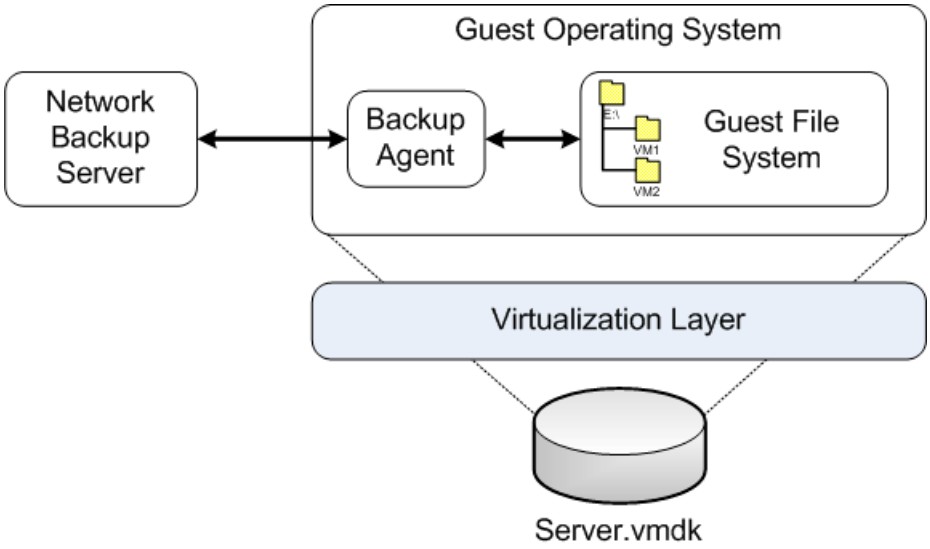
File Level Backups

File-level backups are traditionally used to recover individual files in the event of accidental deletion, corruption, or any other number of reasons your users can find to destroy data. Recovery is typically a straight forward process of choosing the proper file or directory and choosing the proper date of the object you wish to restore. There are some applications now available that allow better application integration with Exchange, SQL, Active Directory, etc… that enable individual object recovery such as database tables or user mailboxes directly from a backup of the respective databases.

Agents

In this scenario, a backup agent is installed within the guest operating system; backup jobs are then created and run on a regular basis, and recovery is handled all by communicating with a centralized backup server.

Figure 10- 2: File Agent Backup



The primary advantage of this design is that it can simple integrate into your existing backup system and provide for full, partial, or file level restores to the virtual machines. The primary disadvantage is that it does not tap into the abilities of a virtual machine to be completely restored in its entirety. Instead, to do a complete restore, you would have to load an operating system onto a new virtual machine, install the agent, perform a complete restore of all files, and finally hope and pray that the applications on the server will run when the process is completed.

Advantages of Performing File Level Backups

- Allows for simple restore of individual files or directories
- Fits easily into most existing backup architectures
- Requires no VMware knowledge
- Day-to-day procedures for backup recovery do not change
- Has application awareness for transactional systems
- Only way to capture data within a Physical RDM

Disadvantages of Performing File Level Backups

- Does not take advantage of virtual machine portability
- Difficult to perform recovery of operating system components

You'll notice that the final advantage that we listed discusses capturing data from within a physical mode RDM. As we learned in Chapter 5, the use of physical mode RDM's provides the best performance for high I/O activities like Exchange or SQL. Since a physical mode RDM does not leverage the VMFS file system to store data or act as a pointer, there is no way for the virtualization layer to receive a SCSI lock for a volume. Because of this, the only way you will be able to retrieve data from a physical mode RDM is through the use of an in-guest agent that has exclusive rights to the NTFS volume. It is possible to convert your physical mode RDM to a virtual mode RDM and access it through a typical virtualization backup utility, but think long and hard before doing this, as there are potential performance implications in this situation.

Image Level Backups

Image level backups leverage processes to get "underneath" the guest operating system and capture the entire state of a server. In the physical world this was done by the use of "bare metal" backup agents. This functionality typically adds significant cost to the overall backup solution, but provides a very easy mechanism to capture and recover an entire operating system, something that is extremely difficult and sometimes impossible when using a file-level agent.

Virtualization provides us with a very simple mechanism to get underneath the guest operating system from the virtualization level. In this scenario the entire virtual machine is treated like a set of files within the ESX Service Console. Here, the VMDK, log, and virtual machine configuration files are backed up and stored remotely.

The obvious benefit of this configuration is that with the restoration of a few files you can successfully restore your virtual machine to its previously backed up state. While this may seem like an optimal solution because of the recoverability of the system, there are some drawbacks that need to be considered.

The sizes of the VMDK files tend to be very large. In some cases, we have seen people create VMDK files several TB in size (which isn't a good idea in general). When creating backups of VMDK files you obviously need to put them somewhere. The old adage "storage is cheap" does not apply when you are trying to cram several TB of space in an enterprise. Also because of the size, they

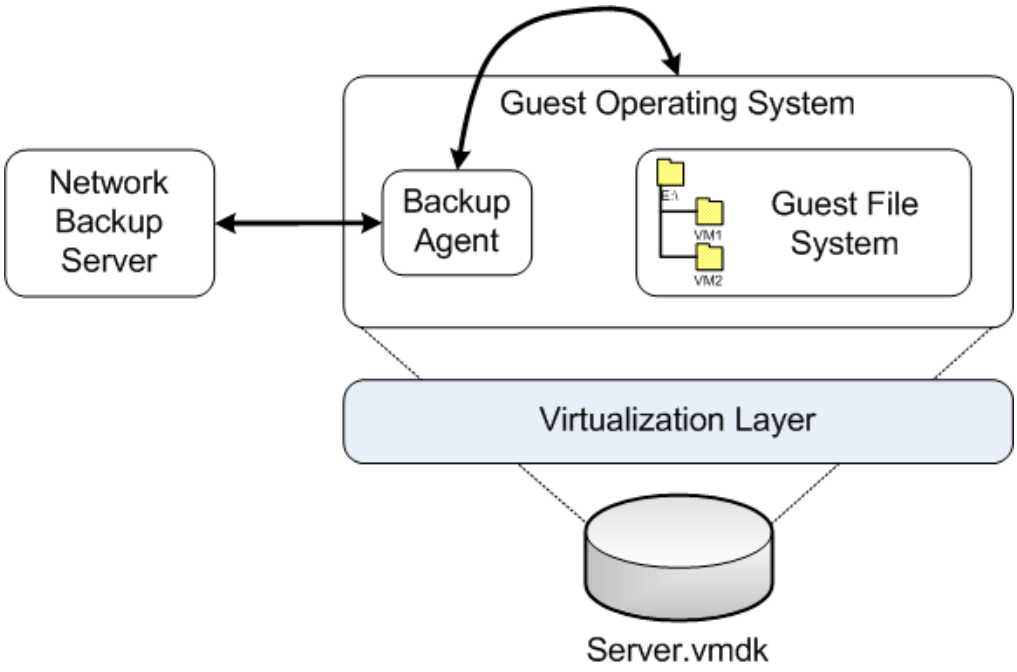
tend to take a long time to copy over the network or even the SAN. We will discuss ways to optimize image size and backup speed in our "Optimizing Backup and Disaster Recovery" section of this chapter.

When looking to perform image level backups of virtual machines there are several approaches that can be taken. Each provides a very unique way to perform the required work and has just as unique advantages and disadvantages associated with it.

Guest Agents

As previously mentioned, many backup agents provide "bare metal" backup and recovery solutions that allow for the capture of an entire system image. You will typically need to pay a premium for these solutions, often times in excess of $1000 USD per server. In order to recover using an image backup from your bare metal agent you will often require a boot CD to properly enter a preboot environment that has enough functionality to connect to the network and access the backup server to recover the image. This does require a bit of manual intervention and VMware knowledge.

Figure 10- 3: Image Agent Backup



Advantages of using Guest Agents for Image Backup

- Uses same software as physical environment
- Uses same backup infrastructure as physical environment
- Daily backup procedures don't change
- Typically allows file recovery from image backup

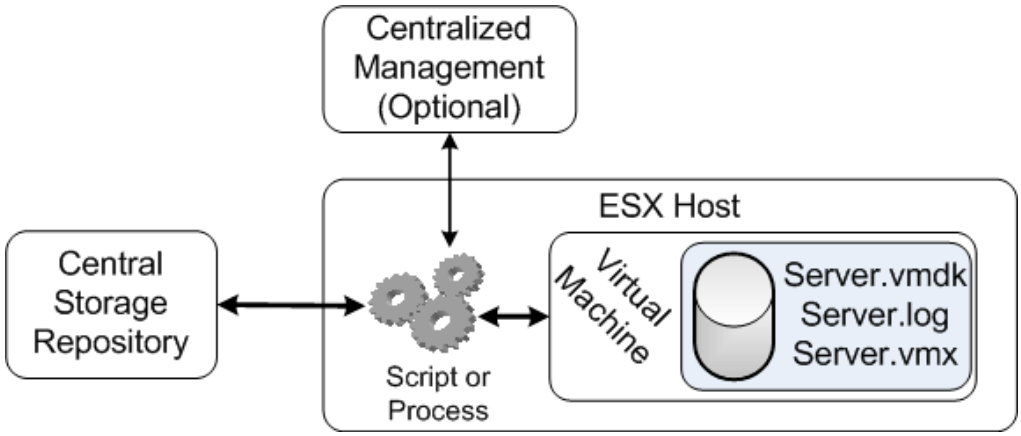
Disadvantages of using Guest Agents for Image Backup

- Costly solution
- Manual restore process

Applications

After VMware ESX really started to hit the mainstream, the recoverability aspect of virtual machines became extremely popular. Several individuals and vendors have been able to capitalize on VMware's capabilities and have written applications specifically suited for virtual machine backup. These applications either run in or communicate with the virtualization layer to capture entire virtual machines, which often consist of at least one VMDK file, VMX configuration files, and log files. The best part of these solutions is that they can run against running virtual machines without the operating system, and most importantly the end users, even knowing that it is occurring.

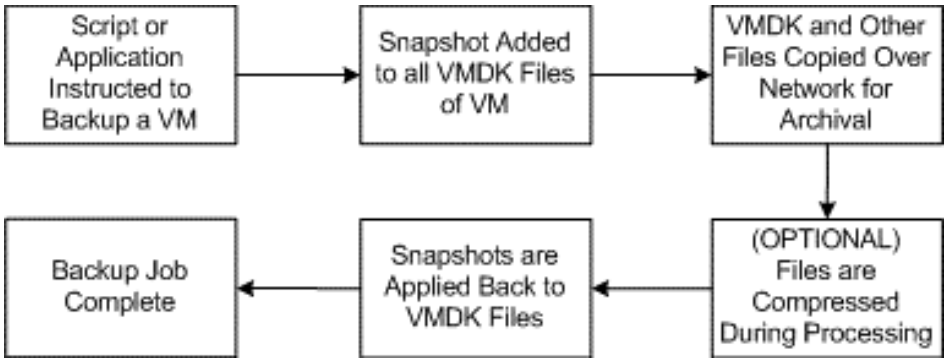
Figure 10- 4: Virtualization Backup Application



As mentioned earlier, there are several very reliable options when looking for applications to assist in the image level backup and recovery of your virtual ma-

chines. All available scripts or applications leverage the same set of steps to capture an image of a running virtual machine.

Figure 10- 5: Backup Process



The oldest method to backup virtual machines, which is still available and actively maintained today, is a Perl script called VMBK.pl, written by Massimiliano Daneri. While scripting normally gets a bad rap, Massimiliano's script should almost be considered more of a full-fledged application. The best part is that this script is completely free. It is by far the optimal solution for smaller implementations of VMware. More information can be found on the script's home page: http://www.vmts.net/vmbk3.htm.

For larger implementations of VMware something a little more advanced is required to effectively manage the backup infrastructure. The clear leader in virtualization backups is Vizioncore's esxRanger. In addition to being an easy to use centralized Windows application, esxRanger provides advanced functionality such as enhanced compression, differential backups, integration with VCB (which will be discussed in a few minutes), VMotion/DRS support, simple recovery, and has basic capabilities for file level recovery from an image backup. While this solution is not free, it is still reasonably priced for the functionality it provides. Further details can be found at Vizioncore's website: http://www.vizioncore.com.

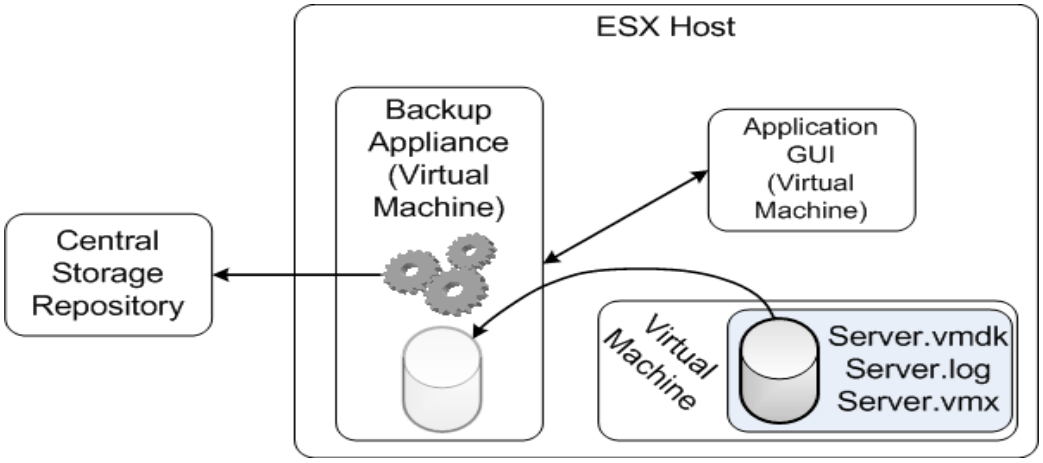
Appliances

A creative way to backup entire virtual machine's images, which has become increasingly popular with the release of VI3, is the use of a backup appliance. The "appliance" itself is a lightweight virtual machine (typically Linux due to

distribution rights) that mounts VMDK files of the virtual machines being backed up. Once mounted inside the appliance, processes are run to copy data from the appliance operating system to remote storage for archival. The key to using appliances is in the fact that a backup appliance does not have the same speed limitations that the ESX Service Console has. This means that appliances have slightly higher speed access, primarily to disk resources, than stand alone backup applications.

In order to provide end users with the capability to configure and manage a backup architecture, there is typically a GUI component involved. This can either be in the form of a web interface or a small "GUI Helper" virtual machine.

Figure 10- 6: Virtual Appliance Backup



Additional benefit can be found from the backup appliance simply by the fact that it is a virtual machine. The appliance itself can participate in VMotion and DRS activity without impacting running backup jobs. This provides a layer of high availability in the event of host performance issues.

While it may seem like there are a lot of advantages of using appliances, there are also some disadvantages. Appliances run in and leverage the same resources as the actual virtual machines being hosted in the virtual infrastructure. If the appliance is performing compression of an image or doing other CPU intensive

activity this will potentially impact the same virtual machines they are trying to backup. ESX does have several mechanisms to limit resource utilization, and appliances can be set up to run in resource pools, but the more you limit the appliance, the slower the backup jobs will run. The key is finding the proper schedule and balance of resource utilization to minimize the impact on the running virtual machines. This does make it slightly more difficult to get up and running.

The primary tool used when considering the backup appliance approach is esXpress by PHD Consulting. It is full of features such as differential backups, a fault tolerant architecture, archive encryption, and even includes virtual machine replication (which we will discuss in the High Availability section of this chapter) whereas this is typically a separate product from vendors such as Vizioncore or DoubleTake. More information can be found at their website: http://www.esxpress.com.

VCB

As previously mentioned in Chapter 2, VMware released add-on functionality for SAN-Based backups for VI3. This add-on is referred to as VMware Consolidated Backup, or simply VCB. We break VCB away separately in this section, as it can ultimately be run as either a file-level agent replacement or as an image capture tool.

A common misconception of many people is that VCB itself is a backup tool. This is absolutely not the case. VCB, as it stands, is a command line driven framework that allows other backup tools to access virtual machine data over the SAN. VCB does not provide data archival and retention, tape management, or advanced scheduling like any number of true backup utilities. Nor does it provide an efficient means to restore virtual machines.

This section is going to focus on the current capabilities of what VCB provides, what some of the important limitations are, and in what situations it can be leveraged best.

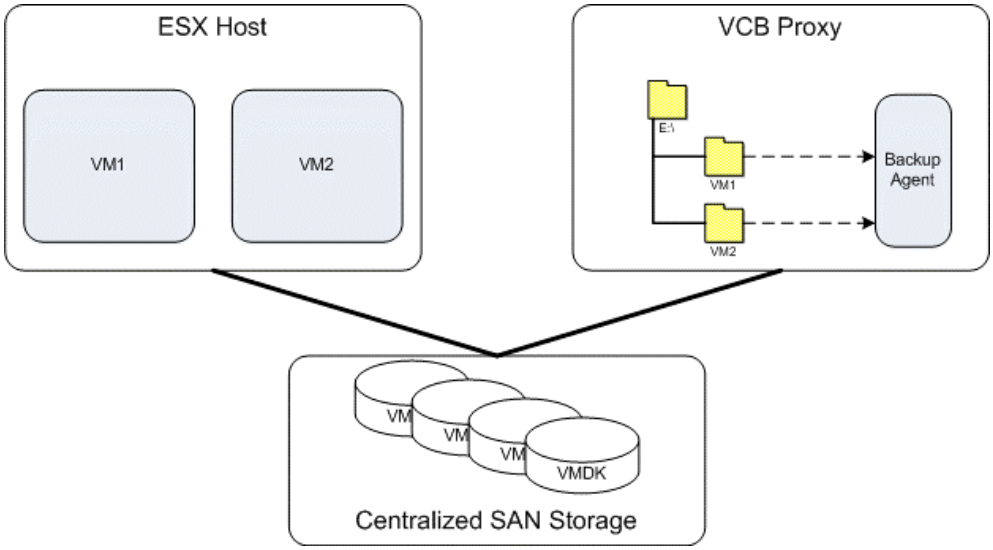
VCB for File Level Backup

The primary focus by VMware for the release of VCB was in eliminating (consolidating being the more backup friendly term) the amount of agents purchased and deployed to virtual machines within the virtual infrastructure. In a typical backup agent implementation, each operating system (whether physical or virtual) has a file-level agent installed. In many organizations, there is a SCt associated with each individual agent that is deployed.

VCB can help in this scenario by leveraging the centralized storage architecture of a virtual infrastructure. Instead of installing a backup agent in each individual virtual machine, VCB allows you to mount the VMDK files in a Read-Only mode to a directory on a Windows proxy server. The proxy server itself has the proper file-level backup agent and backs up each virtual machine's directory. Since the proxy server is accessing the individual files of each virtual machine, backup agents are not required in each operating system instance of the virtual infrastructure.

This solution is only compatible with Windows operating systems at this point. The Windows VCB Proxy does not have the capability to read and understand Linux partitions.

Figure 10- 7: VCB Backup



Behind the scenes, there are several things going on when VCB is executed and the VMDK files are mounted on the VCB Proxy Server.

1. VCB is executed from the command line on the VCB Proxy Server either as a scheduled task or as a pre-script to a File-Level Agent.
2. (Optional) A "Pre Freeze" script is executed on the target virtual machine. This script must be located on the virtual machine. A user can configure a custom script, but it must be called by the "C:\Windows\pre-freeze-script.bat" file.
3. (Optional) A VMware data consistency driver freezes the I/O of the virtual machine and ensures no file system writes are occurring.
4. A snapshot file is added to each VMDK file configured by the virtual machine. It is important to note that if a virtual machine has an incompatible disk, such as a physical RDM or a non-persistent disk, the process will fail and the user will not be able to leverage VCB for this system. There is no way to perform this activity on an individual disk of a VM. It is an "all or nothing" deal.
5. Once the snapshot file is applied to the virtual machine, the consistency driver allows I/O of the virtual machine to resume.
6. (Optional) A "Post Thaw" script is executed on the target virtual machine. This script must be located on the virtual machine. A user can configure a custom script, but it must be called by the "C:\Windows\post-thaw-script.bat" file.
7. The VMDK file(s) of the virtual machine are mounted as directories on the VCB proxy server.
8. The File-Level backup agent backs up the files contained within the VMDK file(s) over the SAN using the standard backup infrastructure.
9. Upon completion of the File-Level backup, the VCB CLI command to unmount the virtual machine is executed either manually or as a post-script of the backup agent.
10. The virtual machine's snapshot file is applied back into the main VMDK file and the system returns to normal operation.

Advantages of Using VCB as a Part of your File-Level Backup Strategy

- Minimize the number of backup agents deployed
- No host or guest CPU utilization to backup guest data
- No network impact since all virtual machines are backed up over the SAN
- Depending on the backup tool being used, only modified files are backed up nightly
- Allows an organization to use a single toolset for backing up the entire environment

Disadvantages of Using VCB as Part of Your File-Level Backup Strategy

- Requires Fiber SAN (iSCSI and NFS implementations are not supported)
- All virtual machines are backed up from a single host, making restoration with some backup applications more difficult
- Since VCB mounts images Read-Only, archive bit on files cannot be modified- may impact some backup applications
- Windows must have access to the shared VMFS LUNs, increasing risk of accidental corruption

VCB for Image Backup

The second way that VCB can be leveraged is to provide a VMDK image of the entire virtual machine. This type of backup is similar to a "Bare Metal" export of the entire virtual machine. This method does not mount the VMDK file or look at the contents inside of it, so is compatible with any operating system that is supported on the ESX platform.

There are several export options when performing a VCB Image backup in regards to what the final VMDK archive contains after the process completes. A user can specify to create sparse or flat VMDK files on the target. If the Sparse file is exported it should be noted that additional steps of converting the VMDK file with vmkfstools will be required before the archive can be properly recovered on an ESX host. Sparse files will tend to run more quickly, as 0-filled data is not copied and written into the archive. Flat files will match the file size of the original source VMDK, as an exact binary copy of the file (including the zeroes) is written on the destination.

There are some minor differences in what occurs behind the scenes when VCB executes an Image Backup vs. a File-Level Backup.

1. VCB is executed from the command line on the VCB Proxy Server either as a scheduled task or as a pre-script to a File-Level Agent.
2. (Optional) A "Pre Freeze" script is executed on the target virtual machine. This script must be located on the virtual machine. A user can configure a custom script, but it must be called by the "C:\Windows\pre-freeze-script.bat" file on a Windows VM or the "/usr/sbin/pre-freeze-script" file on a Linux VM.
3. (Optional) A VMware data consistency driver freezes the I/O of the virtual machine and ensures no file system writes are occurring.
4. A snapshot file is added to each VMDK file configured by the virtual machine. It is important to note that if a virtual machine has an incompatible disk such as a physical RDM or a non-persistent disk, the process will fail and the user will not be able to leverage VCB for this system. There is no way to perform this activity on an individual disk of a VM. It is an "all or nothing" deal.
5. Once the snapshot file is applied to the virtual machine, the consistency driver allows I/O of the virtual machine to resume.
6. (Optional) A "Post Thaw" script is executed on the target virtual machine. This script must be located on the virtual machine. A user can configure a custom script, but it must be called by the "C:\Windows\post-thaw-script.bat" file on a Windows VM, or the "/usr/sbin/post-thaw-script" file on a Linux VM.
7. The VMDK file(s) of the virtual machine are exported to the specified directory on the VCB Proxy Server. No VMDK files are mounted.
8. (Optional) The File-Level backup agent backs up the exported VMDK files using the backup infrastructure.
9. Upon completion of the agent backup of the archive files, the VCB CLI command to unmount the virtual machine is executed either manually or as a post-script of the backup agent.
10. The virtual machine's snapshot file is applied back into the main VMDK file and the system returns to normal operation.

Advantages of Using VCB as a Part of your Image Backup Strategy

- Simplified recovery of entire virtual machine during OS corruption
- No host or guest CPU utilization to backup guest data
- No network impact since all virtual machines are backed up over the SAN
- "Bare Metal" backups without the added cost of upgraded agents

Disadvantages of Using VCB as Part of Your Image Backup Strategy

- Hefty storage requirements since each VM is backed up in its entirety nightly
- No File-Level restore capability from image backups
- Limited number of simultaneous data streams due to I/O requirements
- Windows must have access to the shared VMFS LUNs, increasing risk of accidental corruption

Requirements for Running VCB

There are several key requirements necessary to successfully implement VCB into your virtual infrastructure. Most people will find that the worst part of using VCB is actually in the initial configuration and SAN Zoning.

Physical VCB Server

To start, a standalone Windows 2003 server is required. Previous versions of Windows will not be able to run the VCB framework. Unfortunately, you cannot install the VCB Framework on a system that has VirtualCenter installed. There are common components in both VirtualCenter and the VCB Framework that are not compatible with one another, and certain functionality within VirtualCenter will actually break. This Windows 2003 Server must have a single HBA connected to the SAN that is storing the virtual machine VMDK data. There is no multipath support, and attempting to configure the VMware LUNs down more than a single path will cause issues with running VCB backup jobs.

Temporary Storage Space

It is also a good idea to have ample temporary storage space if you intend to perform image level backups on the VCB Server. These often take up large amounts of temporary space, as an Image Level VCB Backup must actually export the VMDK file and store it locally. The size of this temporary space must be large enough to store at least 2-3 virtual machines' data at any given point in time. Also remember, VCB does not allow a user to selectively choose which VMDK files of a virtual machine to export. If a virtual machine has a VMDK attached, it is getting exported.

SAN Zoning

Once the VCB Proxy server is built and configured the SAN team will need to step in and zone your precious VMware LUNs to the WWN its HBA. The LUN ID's as seen by the Windows VCB Proxy Server must match the LUN IDs as seen by the ESX Hosts in order for VCB to function. Every storage array has a different way to configure this, but it should be a very simple process to complete (even though you will probably hear about how much of a pain it was). When all is said and done, several new hard disk devices should appear in the "Disk Drives" section of the Device Manager.

THIS FOLLOWING PARAGRAPH IS ONE OF THE BIGGEST WARNINGS THAT WE CAN PLACE IN THIS BOOK

Do NOT choose to initialize any disk if Disk Manager is accessed on the VCB Proxy System. Although VMware claims that Microsoft writes its disk signature to a different offset than VMware does, you would be extremely foolish to put that much faith in Microsoft and your own ability to not instantly click "Yes" on any message that pops up on your screen. Although VCB opens VMDK files in a Read-Only mode, it still has full access to the zoned LUNs and an uninformed user can very easily reformat these partitions with NTFS, effectively destroying every virtual machine with a VMDK file on the LUN.

VCB Framework Installation

The VCB Framework installation package can be downloaded and installed from VMware's website for licensed customers. The installation of the VCB

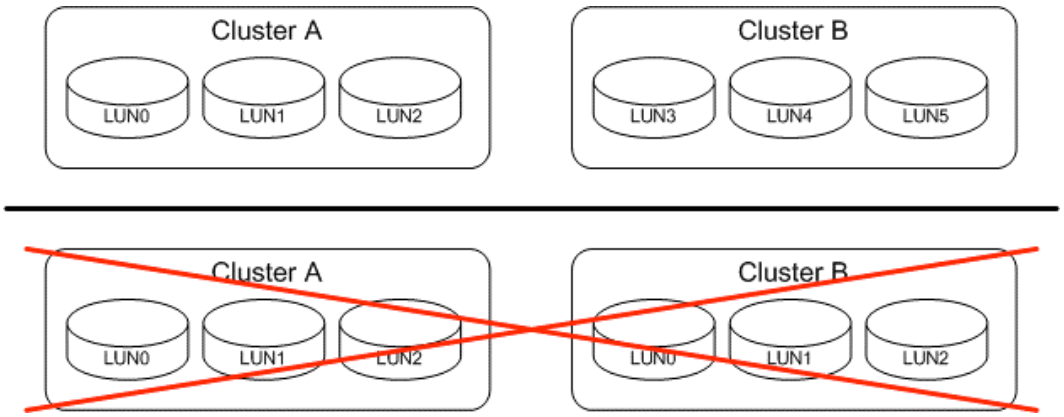
Framework is very straight-forward. The only way to test whether it is working is to properly configure your SAN Zoning and run a test VCB command.

Designing your VCB Infrastructure

In order to have a successful implementation of VMware Consolidated Backup, it is important to understand the limitations and best practices behind integrating VCB with your environment. The challenge with setting up and running VCB is that it is extremely sensitive to the back end storage array being used for the virtual infrastructure. In addition, your mileage will vary based on whether File-Level or Image-based VCB backups will be occurring. The larger, and often times more expensive, storage arrays will be able to execute more concurrent backups than basic, lower cost, storage arrays. The recommendations we make in this section will be middle of the road, but it is entirely possible that you will notice differences based on your particular environment.

The general rule to building out a VCB infrastructure is to have one VCB server per ESX Cluster configured within VirtualCenter. The reason behind this is that a VirtualCenter Cluster serves as a shared storage boundary. ESX Hosts within a cluster are, assuming the environment is properly configured per the guidelines of Chapter 5, guaranteed to see the same LUNs, and most importantly, with the same LUN ID across all hosts. It is possible to have 2 ESX Clusters within VirtualCenter leveraging the same VCB server, but configuring LUN IDs requires extra special care. Having a LUN0 in each Cluster with each being unique will cause some significant challenges when VCB attempts to access VMDK data. VCB may not find the object for which it looks.

Figure 10- 8: LUN ID Assignment is Critical



The type of backup that is being performed by VCB plays a critical role in your sizing strategy for the number of VCB Proxy Servers that are required to backup your infrastructure. When performing backups using the File-Level method, there is a limit of 60 concurrent VMDK files that may be mounted at any given time on the VCB Proxy Server. The important thing to remember here, and any time anyone mentions a limit, is that a maximum number almost never references a realistic value. The amount of data that is being backed up will ultimately drive how many File-Level backups can be run at any given time. Backup agents that have file-level differential technology will allow for more and faster concurrent backups once an entire set of full backups is created. On average, it will be safe to backup about 10-15 virtual machines without VCB starting to experience connection issues to VirtualCenter and have jobs fail with little to no information as to why. Again, higher end storage infrastructures will be able to better support more virtual machines.

When performing image-level exports through VCB there is a strong recommendation to limit the number of concurrent backup jobs to 2-4. The amount of disk I/O generated by doing block level dumps of entire VMDK files not only impacts the VCB Proxy, but could also have an adverse affect on your storage infrastructure. Since VCB has no bandwidth control, it is entirely possible to flood the VCB Proxy with enough I/O to cripple it and cause jobs to fail.

Testing in each particular environment needs to be performed to properly determine whether all virtual machines can be backed up within a specified window when leveraging VCB. Backup speeds are typically dictated by the back-end storage array, but extensive testing shows VCB has the capability to move data at a rate of about 1.5GB/minute on an average sized array. By testing the speed in your particular environment it should be relatively simple to estimate the number of VCB Proxy Servers required to backup the infrastructure. As the virtual infrastructure grows, the VCB environment should grow and scale appropriately alongside it. Often times, people overlook the VCB environment until they run out of capacity and backup jobs begin failing.

Limitations of VCB in a Virtual Infrastructure

While VCB may seem like the ultimate solution, there are several things that you need to be aware of before implementing it wide scale in your virtual infrastructure. Since VCB can move data quickly you need to consider the fact that just because it isn't using any local resources of an ESX host doesn't mean there is

no impact. Remember, this data is flowing over the exact same storage infrastructure that is actually servicing your virtual machines. This doesn't impact the host itself, but it sure could impact the storage infrastructure to the point that every host appears sluggish simply because the SAN is too busy streaming massive amounts of data.

If your environment breaks the few hundred virtual machine barrier, you will find that performing 2-4 concurrent backups on a single VCB proxy is going to be a nightmare to schedule and manage. Even if you add several VCB proxy servers you need to properly split the workload to backup everything you need. This is quite a manual undertaking.

A final thing we have found is that the VMware consistency driver being used by the VMware Tools Service does not always effectively manage transactional systems such as database or email systems. If a snapshot is added to a virtual machine with one of these workloads, take a look at the Event Log of the system. There is a good chance that for every snapshot that is added, there is a database consistency check that goes along with it. While these systems have internal mechanisms to protect the integrity of their data in this circumstance, it does add unnecessary workload to the virtual machine to verify and correct potential issues.

To combat this issue, we recommend writing a small script that can take advantage of alternate technologies, such as Microsoft's Volume Shadow Copy Service (VSS), to better protect the consistency and integrity of transactional applications. While this functionality is only supported in Windows 2003 or higher, it can eliminate the consistency checks that must occur when a snapshot is added to a virtual machine.

As you can see there are quite a few options surrounding simply finding the best way to backup your virtual infrastructure. The good thing is that much of the knowledge you have just gained is also applicable to providing disaster recovery for your virtual infrastructure.

Disaster Recovery

Now that we have spent a very large portion of time discussing the various backup recovery methods and technologies let's dive into the similar topic of disaster recovery. Disaster Recovery refers to your capability to get your system(s) back online and running in the event of a catastrophic failure.

There are only a few things that are for sure about DR. First, it takes a lot of planning and preparation to do right, and second, it's never cheap when done correctly. DR with ESX server is no different. We have all heard the sales pitch as to how easy DR is with VMware, but is it really easy or cheaper? Maybe it's easier; however, I am not sure about cheaper.

To recover a VMware environment, we need to analyze three primary components. The first is the ESX Servers themselves. Without them, we won't get a single VM up and running. The second is the supporting infrastructure such as storage and networking components. The third and most important component is your virtual machines themselves. Notice we didn't mention VirtualCenter. In reality, this is not required to get your DR environment up and running. We will look at a few different ways to handle each of these and let you see which fits best into your environment.

ESX Server Recovery

It would be nice during a disaster recovery scenario to have a bunch of preconfigured DR hosts setup and ready to go. If this is the case in your company, then your virtual infrastructure disaster recovery is one step easier. Have the server built and ready for the situation, and you are ready to go. However in most business this is not the case. Basically, if you don't have a bunch of standby hardware, you are dealing with one of two solutions:

- A contract facility that will provide you with "like" hardware.
- The repurposing of existing company owned hardware.

The first option is used by a majority of companies out there. The basic idea is that they contract with someone like SunGard, so that in the event of a disaster,

SunGard guarantees them X amount of rack space and bandwidth and Y amount of different types of hardware.

The issue here is that you will most likely not get the SAME EXACT hardware. Or you may get the same model, but will it have the same HBA's in the same slots, the same types of NICs plugged into the same speed network ports, etc? If not, your unattended install for ESX that does all the cool NIC and HBA configurations just went out the Window. So when planning for the redeployment of ESX or when trying to ensure that you get the correct contract level that will ensure you get hardware as close as possible to your existing configuration, you will need to adjust your DR builds for ESX to be extremely flexible and allow for manual configuration of the server.

The second option of repurposing existing hardware is much more interesting. In repurposing, you are taking existing hardware that you own and rebuilding it as ESX servers. This gives you a little more flexibility as you will have a better idea of the hardware that will be used during DR. Some organizations will attempt to repurpose other "less critical" production servers for the critical production boxes. This is a poor plan, as you are really robbing Peter to pay Paul. You will eventually have to bring back those production servers that were "less critical".

If your organization already has multiple datacenters running their own virtual infrastructures, your DR plan may come down to prioritizing your virtual machines and turning off test, development, and non-critical production virtual machines to ensure the necessary host capacity is available to bring up the critical virtual machines. Through the use of technologies such as virtual machine and SAN replication, that we will discuss in the High Availability and Fault Tolerance sections of this chapter, this type of scenario can actually be simplified quite a bit.

Infrastructure Recovery

Just having ESX hosts available does not make for a solid disaster recovery plan. There is some consideration that needs to go into portions of the infrastructure that support the virtual environment. Since this book is not a disaster recovery book, we will only lightly touch on these topics to provide awareness

that disaster recovery of virtualization does require thought outside just the virtual infrastructure.

Storage

When you need to get your virtual machines up and running at a disaster site the first thing that should cross your mind is "Where should I put them?" The virtual machines obviously need to be stored on a platform that VMware supports (SAN, iSCSI, NFS, etc) and you need to make sure you have enough storage to bring up all of the systems you require. If you are using SAN replication technologies, which are very costly, you do not need to worry as much as someone who is backing up images of virtual machines and needs to recover them to alternate storage types at a destination datacenter. The keys when taking your storage environment into consideration when planning your virtual infrastructure disaster recovery plan are to make sure you have enough storage and that it is compatible with ESX3. If you have the right recovery tools, it won't matter if you are using local SCSI, iSCSI, or a SAN from an alternate vendor.

Network

Just as important as having a place to store your disaster virtual machines is the capability to communicate with them in the event of a disaster. In most disaster recovery plans there are several configuration items that will change and will need to be addressed to get your virtual machines up and running on the network again.

There is a very good chance that the VLAN's at the disaster site do not match the VLAN's at the primary site. This means that in order to communicate on your network you will probably need to change the IP Addresses of your virtual machines. Any time you change an IP Address of a system you need to know if there are also DNS entry changes that are required. If you change the IP Address of a web server and its communicating on the network, it doesn't do any good until you let people know that website.example.com now points to a different IP Address. Having a solid runbook of each virtual machine is a critical component in knowing everything that needs to be modified or anticipated during any change made to the system, especially in a disaster scenario.

Virtual Machine Recovery

Now that there is a plan around host and infrastructure recovery we can finally look at what it takes to get our virtual machines up and running in a disaster. If your backup recovery process is leveraging the correct data capture methods, your disaster recovery process will actually be relatively simple.

Leveraging Backup Recovery as Disaster Recovery

Since there are many tools available to assist in backing up virtual machines using various methods, there is no reason that your daily backup procedures can't be used to provide disaster recovery for your virtual infrastructure also. There are a few things to be careful of if you are planning on using your backup procedures for this purpose. The way that the data is captured and archived will play a key role in the level of recoverability you have in your environment.

Inefficiencies of File Agents for DR

Using the file level recovery method for you disaster recovery plan will only truly help you if you are only concerned about retrieving application data. It will not help you if you need to recover entire operating systems. We will see towards the end of this chapter how we can properly leverage both file level and image level backups for disaster recovery, but here we just want to point out that if you want your operating systems back, don't use file agents exclusively. It is just as much of a pain in the butt to restore a virtual machine using a file level backup as it is a physical.

Image Backups for DR

If you plan things properly, and we certainly hope that we point out in this book that properly means performing image level backups of your virtual machines, then your disaster recovery process will be almost as easy as VMware says it will be.

First off, let's determine what components of a virtual machine are required in a disaster scenario. At the absolute bare minimum, your VMDK files will be required. The VMDK data is the only component that is absolutely required to get your data back. Unless you are using your own scripting mechanisms, and

not doing the best job at it, you most likely will have the VMX configuration file of the virtual machine also, as well as the NVRAM file, which stores the BIOS settings for that particular guest.

If you are using a manual process as opposed to one of the available automated tools, you will need a way to get your backup files onto the destination ESX host (assuming they aren't already there). One of the best free tools available to perform large file copies to ESX is FastSCP from Veeam Software: http://www.veeam.com. If you do not have the VMX file you will need to leverage the Virtual Infrastructure Client to build a new virtual machine on a specific host. When it prompts you for disk configuration you will need to point to the directory within VMFS that you copied your VMDK data to. You will need to add a new hard drive to each virtual machine for each VMDK file you have that belongs to the system.

If you were smart enough to grab the VMX file with the VMDK file of the virtual machine, the recovery process is simplified even further. Unfortunately, the Virtual Infrastructure Client does not have any functionality to import an existing virtual machine onto a host. So, you will need to break out your trusty command line skills. You will need to log in and escalate your privileges to run a command as the root user. With the following command your virtual machine will be registered, visible and ready to start on the ESX host where you registered it.

```
vmware-cmd –s register /path/to/servername/servername.vmx
```

If you are using one of the many automated backup recovery tools available such as esxRanger or esXpress you will not need to worry about any manual intervention using either the Service Console CLI or the Virtual Infrastructure Client. These tools will take care of copying your data back to the host, registering the virtual machine, and even giving you the option to power on the virtual machine when the process is completed.

Optimizing Backup and Disaster Recovery

There are several methods that backup which storage vendors are employing that significantly enhance the efficiency of backup and disaster recovery in a

virtual infrastructure. When you take a step back and look at what it takes to most effectively implement a solid set of backup and disaster recovery processes, you will see that there are three primary points that must be addressed.

- The size of the backup archives
- The amount of time it takes to create an archive
- The overall impact that creating an archive has on your virtual infrastructure

There are several advanced technologies available to us through backup software or storage hardware that assist in the three key areas identified above. When at all possible, these advanced technologies should be reviewed to see if they can be fit into your backup and disaster recovery plan.

Minimize the Size of Archives

VMware image files have a tendency to get quite large. When capturing images for backup or disaster recovery you would be foolish not to enable or leverage compression options for the archive files. There are some cases where compressing archives just won't cut it. When a virtual machine is scheduled to run nightly and has 20GB of data that is backed up, there is a lot of money going into storing these archive files. Over the period of a single week 140GB of storage space would be used just for this single virtual machine.

Thankfully, there are several options available that help alleviate these ridiculous storage requirements for virtual machine archives. You will notice that many of the technologies have been in place for years in typical file level backup agents. There is no reason these same technologies cannot be applied specifically to virtual machine archives since, well, they are just files.

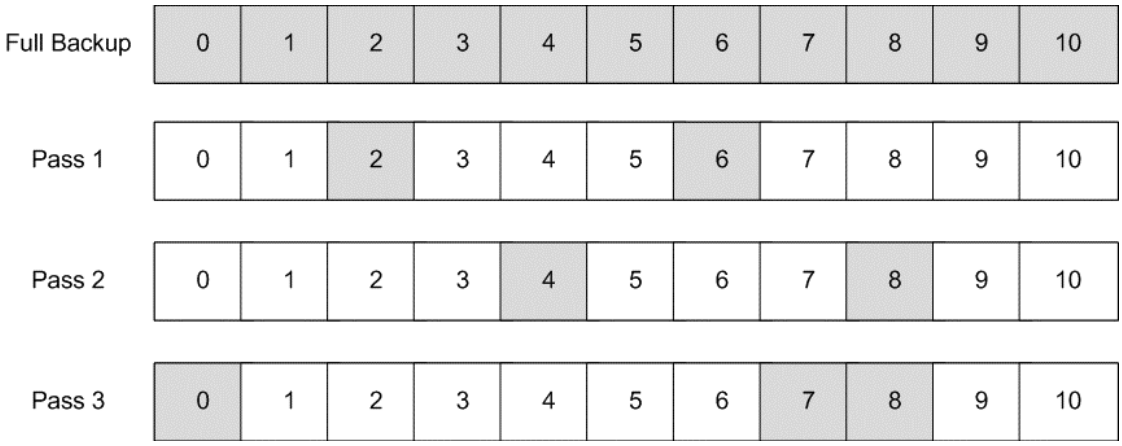
Incremental Backups

In a typical file level backup architecture incremental backups only archive files whose data has changed since the last backup job was run. Since a VMDK file is constantly changing every second, it does make sense to say "Only backup my VMDK file if it has changed since the last time I ran a backup job." In fact, the only time this is possible is if the virtual machine is powered off. In order to

combat this virtualization, backup vendors had to break down a VMDK file into smaller pieces.

By scanning and comparing each individual block of data that makes up a VMDK file, backup vendors have a way to backup only the portions of the large files that change between backup jobs. When performing an incremental backup of a virtual machine, only the data blocks that change between every backup job are archived.

Figure 10- 9: Incremental Backup



As you can see in the diagram, a full backup is required at least once; otherwise there is nothing to compare against for an incremental backup. On the first pass after the full backup there are two data blocks that change and are backed up for that particular pass. The same is true for the second pass where two more unique data blocks change. Pass 3 is unique in the fact that three data blocks changed since the previous pass

Based on the given example we can do a quick comparison of how much space is theoretically saved by using incremental backups vs. a full backup on every pass.

Nightly Full Backups

Pass	Blocks
Full	10
1	10
2	10
3	10
Total	40

Incremental Backups

Pass	Blocks
Full	10
1	2
2	2
3	3
Total	17

Incremental backups have always had one fatal flaw. In order to recover you need to have the original full backup job plus every incremental file between the full backup and the point in time to which you would like to recover. In the event that one of the incremental archives is deleted or corrupt, anything after that particular incremental file is suspect.

Going back to our diagram, if the archive file from Pass 2 was accidentally deleted and we needed to recover up to Pass 3 we have an issue. Blocks 0, 2, 6, 7, and 8 are actually safe because they are protected by recovering Pass 1 and Pass 3. The problem is that we have no idea that block 4 ever changed, so will still contain data from the full backup. While this does not seem like a major issue, remember that this is a VERY simplified example, and VMDK files often consist of thousands of blocks of which hundreds can change on a nightly basis. Corruption of a single archive file can turn your recovery result into a giant block of Swiss cheese.

Advantages of Incremental Backups

- Only captures data that changes between passes
- Maximum space savings for archival of virtual machines through software

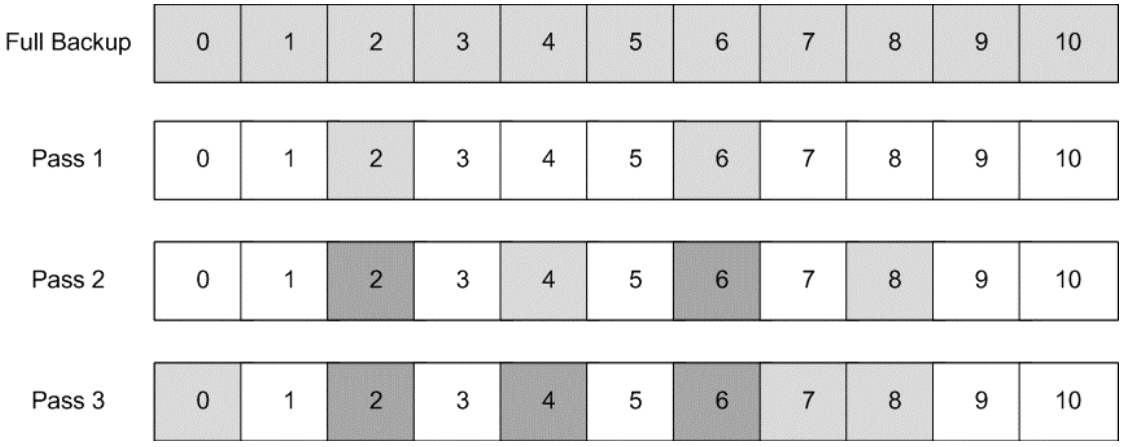
Disadvantages of Incremental Backups

- Requires every incremental file to properly reassemble a virtual machine
- Increased probability of archive corruption

Differential Backups

Differential backups actually put a slight spin on the method of using incremental backups to archive your virtual machines. The difference between a differential and incremental backup is based on the fact that instead of comparing data that changes night to night, differential backups always perform a comparison to the original full backup. This actually makes each block modification cumulative for every pass.

Figure 10- 10: Differential Backup



This concept is a little confusing so let's refer to the diagram. The full backup still needs to occur and will still consist of 10 data blocks. For the first pass, there is actually nothing different between an incremental and a differential backup; two blocks have changed, so two blocks are archived. The difference between differential and incremental backups comes into play for Pass 2. Remember, differential backups always compare what has changed back to the original full backup, not the previous pass. A differential backup will not only

capture blocks 4 and 8, which changed between Pass 1 and Pass 2, but it will also re-capture blocks 2 and 6 since they have also been modified since the full backup was taken. Pass 3 will use the same logic and will back up all six blocks that have changed since the full backup was taken.

Let's take a quick look at our table comparing block utilization when using differential backups vs. a full backup.

Nightly Full Backups

Pass	Blocks
Full	10
1	10
2	10
3	10
Total	40

Differential Backups

Pass	Blocks
Full	10
1	2
2	4
3	6
Total	22

While it is still a space savings, it is not as good as the disk savings achieved by Incremental backups. We see that over time differential archive files will continue to grow in size. Theoretically, they can grow as large as a full backup job, although this is highly unlikely. Just don't defrag your hard drive and continue to perform a differential backup… that gets pretty ugly.

The best part about using the differential backup methodology comes into play when it's time to actually recover a virtual machine. Unlike Incremental archives where you need every single archive file, with differentials you need a maximum of two files. You will always need the original full archive. If you need to recover to a point in time, you only need the differential file that was

created for that particular pass. This is possible because that differential file has every block that has changed since the full backup was taken. It does not matter if the prior pass failed, was deleted, or is corrupt.

Advantages of Differential Backups

- Simplified recovery: only need a maximum of 2 archive files to recover a virtual machine
- Low risk of corruption
- Still provides a space savings over performing a full backup with every pass

Disadvantages of Differential Backups

- Each differential archive is increasingly larger than the previous
- Over time differential files can become as large as full backups

Both incremental and differential backups have significant risks the more times they are run in between new, full backup jobs. Incremental files run the risk of corrupting significant amounts of data, and differential files grow in space very quickly. Both of these issues can easily be solved by running regular, full backups. This allows you to start with a clean slate on a regular basis. The frequency that a full backup job should run depends on the backup method you are using and how much data ultimately changes between backup passes. For information on optimizing the speed of your full backup jobs, take a look at the "Minimize Time Required to Capture Images" section later in this chapter.

Deduplication

The process of deduplication is typically used by storage vendors to remove commonalities in data across multiple files. These commonalities are stored in a high speed index file so they can be quickly scanned and compared. This technology is always run at either the block or byte level of the storage device.

The easiest way to explain deduplication is to look at a typical virtual infrastructure. Most organizations will deploy their virtual machines from a standardized template. If an organization deploys thirty virtual machines from the same template there is a very large amount of data that is identical across each of those instances. In fact, the only differences are those caused by guest customization. If a backup job leveraging deduplication backs up every virtual machine

in the infrastructure it will perform a backup of the first virtual machine and build an index file of each block. As future virtual machines are backed up, any block that matches an entry that exists in the index will simply have a pointer generated that states "Block X of data = Index entry Y". This is much more efficient that backing up the entire data block.

By the time the process is complete for all 30 virtual machines it is not uncommon to notice that there is significant storage space savings involved. While the example we gave is a very simplistic example, it goes to show how much benefit can be achieved using this technology. There are some challenges that do pop up when using this scenario in the real world. The index that is generated is absolutely critical. If this were to become corrupt or deleted it will be impossible to recover any of your virtual machines, since the archives are fragmented with real data and pointers to index records. Many deduplication vendors have failsafe measures in place, but you'd better make sure you fully understand what they are and how to recover in the event that something goes wrong.

When it comes time to recover your virtual machine there is no simple process of saying "Here is my archive file, let me restore it." Once deduplication works its magic and properly indexes the data, there is no way to recover without having the entire backup infrastructure in place. This makes it slightly more complicated (and costly) since every ounce of stored data will need to be stored and replicated across multiple datacenters to have true DR capabilities of your virtual machine images.

There is a lot of movement in this area, and nearly every hardware vendor providing storage solutions is putting deduplication technology into their data protection offerings.

Advantages of Deduplication

- Most efficient use of storage space available
- Very high speed backups for virtual machines
- Backup workload is offloaded if managed by the storage hardware

Disadvantages of Deduplication

- Index vulnerability
- Entire backup infrastructure must be available to recover

Minimize Time Required to Capture Images

The second key aspect to optimizing backup and disaster recovery in a virtual environment is capturing your images as fast as possible. You do not want your backup jobs taking so much time that they extend beyond backup windows, as this could technically impact portions of your virtual infrastructure to the point that it will be noticed in your virtual machines.

The Smaller the Archive, the Faster the Backup

The easiest way to increase the backup speed of your virtual machines is to backup less data. Fortunately, we discussed several methods of doing that when we mentioned Incremental and Differential backups and deduplication technology. Keeping the archive file that is moving across the network to a minimal size will ultimately allow you to backup more virtual machines in a given backup window.

Don't Backup Useless Data

This one is definitely easier said than done. When you are dealing with modern operating systems they do some things to enhance performance of the operating system that don't necessarily play nice with virtualization. When data is deleted from an operating system the data itself is not deleted, just the pointers that tell the operating system where that data is. If you are using image level backups and delete 4GB of data just before performing a backup, you will still capture all of the data that was just removed. The question then becomes, "How do we get rid of this data?" Well right now there is an answer, but it is not optimal due to the amount of I/O required to make it happen.

There are several tools available, many of them free, which will allow you to write 0's over data that is stale to the operating systems. The most popular of these is the SDelete utility from Sysinternals. This utility is Windows based. If you want similar functionality for Linux guests you will need to take a look at some creative scripting to perform similar functionality. Zero filling your VMDK files should be considered if you have a system that reports it is using a small amount of data, but the backup archive size is consistently larger than reported.

Minimize the Performance Impact to the Virtual Infrastructure

The final aspect to optimizing your backup and recovery process is minimizing the impact that performing backups or capturing images has on your virtual infrastructure. The only way to really do this is to offload this work onto a component of the infrastructure that is removed from the ESX hosts themselves.

VCB

We've spent a significant amount of time discussing what VCB is and how it can be used. If you are looking to perform file level backups of your virtual machines, this is easily the best available option. If you intend to use VCB for image level backups you are best off leveraging a product like Vizioncore's esxRanger that can further enhance VCB by enhancing the compression of the backup archives.

SAN Snapshots

A second option to offload your image archives of your virtual machines is to keep everything entirely on the SAN. Many SAN vendors have snapshot functionality built in that can take a point-in-time snapshot of a running virtual machine and allow you to move it straight to tape or a secondary SAN. This solution is nice as it requires no additional hardware such as a proxy server. These SAN based solutions often come at an additional cost to enable the required functionality which may put the overall solution out of reach for small to medium-sized businesses.

High Availability

We finally have the opportunity to stop talking about backup and disaster recovery for a while and get to talk about the next layer of the pyramid, High Availability. Having a highly available environment gives you the capability to recover from a catastrophic failure extremely quickly. If your infrastructure is considered "Highly Available" there will still be a minimal amount of downtime

while the virtual machines are recovered. This downtime should not exceed more than 10 minutes in duration for the most critical systems.

In highly available infrastructures, you should not need to recover any data from tape or other media. The virtual machines should have the capability to simply be powered on and continue running where they left stopped. There are ultimately two scenarios that you should prepare for when designing your high availability infrastructure.

Local HA

Providing highly available services still has some benefit, even if there is only a single site involved. No matter how hard they try, hardware vendors will never be able to manufacture hardware that has a 0% failure rating. In the event that one of your ESX hosts goes down due to a hardware or even software issue, you need to have the capacity to run your virtual machines on alternate hardware.

VMware HA

As we learned in Chapter 4, where we discussed your VirtualCenter design, VMware provides their High Availability service to counter such issues in your environment. VMware HA enables a heartbeat between all ESX hosts configured within a cluster. In the event that one of these hosts stops contributing to the heartbeat traffic, VMware HA assumes the host has failed. Within 15 seconds of a host failure, VMware HA starts analyzing the other hosts in your VirtualCenter cluster and begins powering on the failed virtual machines on the hosts that DRS deems best suited to handle the workload.

VMware HA is an automatic process but does require some configuration to work properly. The options chosen for your clusters around maximum number of host failures allowed, restart priority of your virtual machines, and isolation response are fully described in Chapter 4. You will want to follow the guidelines of that chapter to best design your VMware HA specific options for your organizations availability requirements.

The single downside to VMware HA is that it requires a centralized storage infrastructure to properly function. This is the only way every host in the cluster has access to all required virtual machines. If there is a larger issue in the environment such as a wide-scale storage or network failure you will need something more than simply having the capability to power the virtual machines up on an alternate local host.

Remote HA

More important than being able to recover from a local host failure is the capability to recover from a more drastic scenario such as losing your storage infrastructure. In this event, you had better hope that you have a remote high availability plan in place. This can be done using two different methods, each of which serve a different level of SLA and come at very different cost points.

Virtual Machine Replication

Virtual machine level replication is one of the hottest technologies in business continuity right now. Virtual machine replication takes the idea of incremental image backups to the next level. This functionality uses the same process of tracking blocks of data that are modified between backup passes. Instead of storing archive files off for reassembly at a later point in time, replication takes these incremental files and applies them straight into a VMDK file being stored on an alternate host attached to an alternate storage infrastructure.

The primary advantage to performing virtual machine replication is in the fact that individual virtual machines can be replicated without the need to take every other virtual machine that with which it shares a LUN. This solution is often extremely affordable when compared to providing SAN level replication. Another major advantage is that virtual machine-level replication is storage independent. That means you can replicate your virtual machines from a Fiber connected SAN at your primary location to lower cost storage such as iSCSI or NFS at the destination site. You do not need duplicate storage architecture at your disaster site.

While it may seem like the ultimate solution, it is important to note that this solution is for one-off virtual machines that aren't necessarily "SLA 1" systems.

Due to the incremental scanning processes it is not possible, or recommended, to replicate a virtual machine at smaller interval than every 10-15 minutes.

Many people ask "If I have replication, why do I need backup recovery as well?" Fortunately, this one is easy to answer. Performing image level backups still maintains a point in time to which you can recover your system. If you pick up a virus or you lose data, you can simply specify the archive you would like to recover and you are back in business. If you are leveraging a block-level replication technology that is only looking at the VMDK file and you get a virus or lose data, those same changes are going to be replicated to your destination host on the next incremental replication pass.

There are several vendors that are writing software applications to perform this level of functionality; most notably Vizioncore and DoubleTake. The low cost of these solutions make them the optimal method for small to medium sized businesses. That is not to say that large enterprises with thousands of virtual machines cannot also benefit. Replicating a field office Exchange server, using low cost storage, back to the corporate headquarters is just one example of how a low cost solution has quite a few benefits over SAN level replication. The greatest advantage of these vendors is that there is no manual intervention required in the event of a failure other than powering on the target virtual machines. All of the configuration and registration is already handled as a part of the toolset.

Advantages of Virtual Machine Replication

- Low cost point to enter high availability
- Can selectively choose which virtual machines to replicate
- Automatically manages configuration of target virtual machine

Disadvantages of Virtual Machine Replication

- Low replication pass frequency

SAN Replication Technologies

There are instances where an application or set of virtual machines require more frequent updates than every 10-15 minutes. When this situation comes up, the only available option at this point is SAN level replication. SAN replication has been in use for years and is a proven technology to provide some of the highest service levels.

SAN replication works at the storage processor level of the storage array. For this reason, SAN based replication technologies do not have any awareness of what exists inside the storage volume being presented to your ESX hosts. The significant advantage to this is that no host resources are leveraged to replicate your data volumes. If you set up replication, every byte of data on the replicated LUN will be sent to the disaster storage array. This is not an optimal solution if you have large volumes containing multiple virtual machines. What will end up happening is you will have a need to replicate one or two virtual machines, but have to copy the entire LUN, which contains every virtual machine. If you plan on using a SAN replication technology we highly recommend that you identify the systems that truly require the highest levels of high availability and you create RDM's for their operating system and data volumes.

There are two strategies for SAN based replication that are used; Synchronous and Asynchronous. Most implantations use Asynchronous replication. This means that data is committed to the source disk and is buffered and written at the destination second. Synchronous replication forces the local data write to wait until acknowledgment has been received that the destination successfully committed the data write. Asynchronous replication will yield the best performance, since it will immediately write data to your local copy, whereas Synchronous replication will give you the absolute highest level of data protection at a performance penalty. This penalty often comes down to how much bandwidth you have between your primary and secondary storage arrays. Unless you are using dark fiber or other extremely high-speed communication between sites it is highly recommended that you stick with Asynchronous replication.

One final note on leveraging SAN based replication is that there is no awareness of the virtual infrastructure at the destination site. You will either need to script a process to scan for and register virtual machines on the replicated volumes, or you will need to manually configure each virtual machine on the destination side. Just like with virtual machine level replication, the target virtual machines will be stored in a powered off, or "cold" state until they are required.

While the cost of SAN based replication is significantly higher than that of virtual machine based replication, the cost is starting to come down thanks to some of the lower cost iSCSI storage alternatives. If you do need this level of recovery, be warned now that it still will cost you.

Advantages of SAN Replication

- Higher replication pass frequency
- No ESX host resources used

Disadvantages of SAN Replication

- High cost
- Increased complexity
- Limited control over which virtual machines are replicated

Fault Tolerance

We've already discussed backup and disaster recovery of your virtual achiness, as well as making them highly available through several mechanisms. Now we want to discuss what it takes to ensure your applications stay running even when portions of the virtual infrastructure fail. This provides the highest level of availability for your systems, and the use of the technologies described in this section are the only way to truly read "five nine" availability (99.999% uptime).

Fault Tolerant Technologies

At this point in time, there is no way to achieve true fault tolerance by simply using the capabilities of your virtual infrastructure. We define fault tolerance as the capability to lose a portion of your infrastructure and still have no end-user impact or downtime as a result. The only way to make this possible is to leverage not only the benefits of the virtual platform, but also the advanced solutions provided by the physical infrastructure, operating systems, and applications running inside your virtual environment. With the move to a virtual infrastructure, you, as an engineer or architect, need to understand how these work and when they should be implemented.

Network Load Balancing

Network load balancing involved creating a publicly available virtual IP address and placing a pool of servers, typically web servers behind this. The pool of servers behind the virtual IP address not only share connection load, but also provide fault tolerance to one another. If any server in a network load balanced

pool were to fail, that server would simply be excluded from the algorithm to assign connections to a particular system.

Virtual machines can benefit from this functionality just the same as physical systems. The mechanisms to configure this can be done through software, such as Microsoft's Network Load Balancing (NLB), or through hardware such as F5's BIG-IP. For the type of workload that a network load balancing infrastructure protects, it is often a better idea to scale out to multiple servers than to cram more connections or user sessions into a single virtual instance.

Clustering Services

It will probably be about a week after the first virtual machines start to go into production that the VMware administrators will be asked about clustering. More than likely, it will be even before then. There are several technologies available, and many applications have their own form of clustering. Most of them have commonalities in the way they must be configured within a virtual infrastructure, but for the purpose of this book we are going to focus on what it takes to provide clustering services for Microsoft Clustering Services (MSCS).

Cluster in a Box

A cluster in a box is a configuration in which two virtual machines running on the same ESX host are configured in a cluster. The main purpose of using clustering services is to protect against hardware failure.

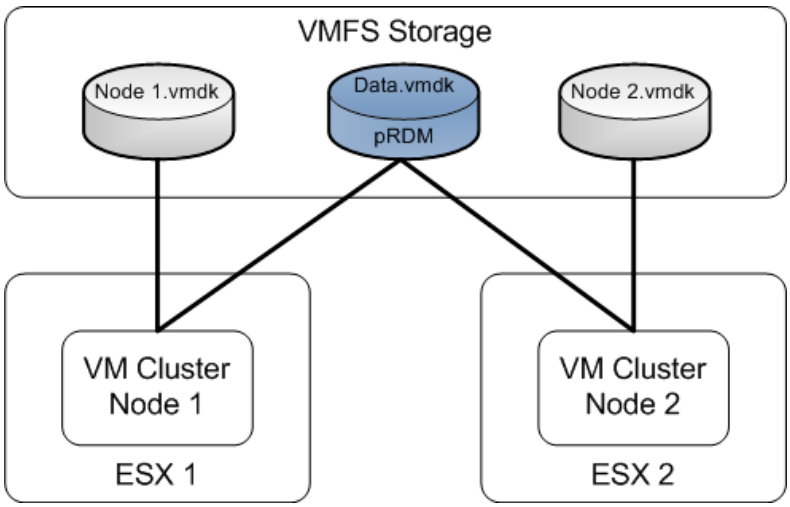
Virtual to Virtual Clusters

Using a Virtual to Virtual cluster is just as it sounds: two virtual machines participating as cluster nodes to serve up a common resource. These virtual machines can be located on the same ESX server or spread across multiple ESX servers to protect against a host or hardware failure.

Properly configuring clustering requires a few components; you will need some type of shared storage, a front-end and heartbeat network, and you will need to ensure that the common data remains intact. Remember a cluster only protects

against hardware failure (in this case virtual hardware) or possibly a service failure. It will not keep data from becoming corrupt.

Figure 10- 11: Virtual to Virtual Cluster Disk Configurations

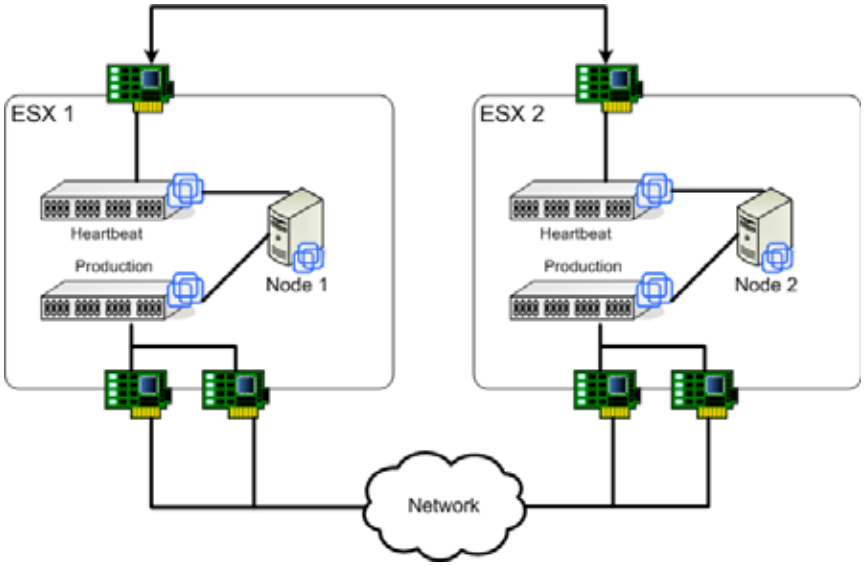


In a Virtual to Virtual cluster, the most common configuration is to store each cluster node on a separate ESX Server. This provides protection against a hardware or host failure. In addition, it would be a good idea to keep the operating system VMDK files on separate LUN's from each other. This simply removes a single point of failure for the cluster. Finally, you will still need at least one LUN to share across both hosts and act as a datastore for any number of cluster aware applications. This storage must be a physical mode RDM. A physical RDM turns over SCSI-3 reservation control to the Windows guest operating system and away from the VMkernel.

Once you have the storage properly configured you need to ensure you have adequate network connectivity. Using a product like MSCS, you are often required to have two network connections for each node, one for the production network and another for a heartbeat network.

In this configuration you should put the heartbeat network on its own virtual switch. On a physical network, these machines are generally connected via a cross over cable or isolated on their own segment. In the VMware world, we must design our hosts to accommodate the extra NIC's or VLAN's required for the heartbeat network.

Figure 10- 12: Virutal to Virutal Cluster Network Configurations



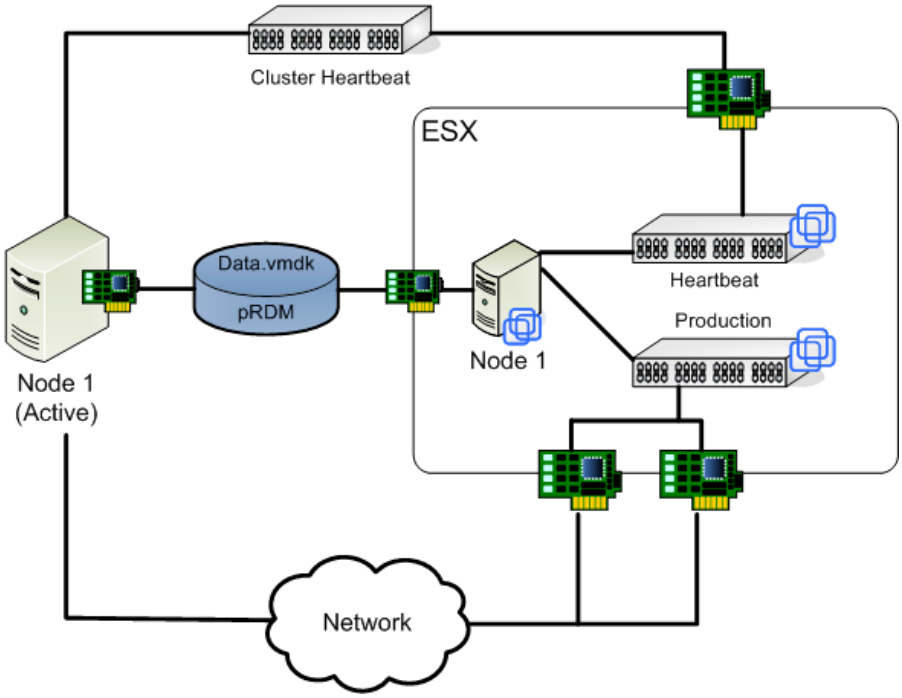
As you can see by the image, the additional heartbeat network reduces your number of available physical adapters by 1. If you think back to Chapter 6, another possibility is to utilize 802.1Q VLAN tagging for your virtual switches. This will allow you to create several port groups on the same virtual switch, eliminating the need to add additional physical NICs to your ESX hosts.

Physical to Virtual Clusters

Another way virtual machines are being used is to consolidate the number of passive cluster nodes on the network. Let's assume you have plans to implement 10 active-passive clusters in your environment. Assuming that you went with a simple configuration like a pair of dual processor systems for each cluster, you would wind up with 20 processors, which basically sit around all the time waiting for a failure that rarely occurs.

Some companies have decided that using virtual machines as the passive cluster nodes in this scenario makes perfect sense. Assuming you have 10 passive nodes, that will use very little processor and memory unless there is a failure, it is possible to consolidate these nodes onto your ESX servers and use physical RDMs in ESX to connect the virtual machines to the shared storage that the physical machines will be using.

Figure 10- 13: Physical to Virtual Cluster



Using this configuration, the virtual machine passive nodes have the ability to access the production network, the shared data and quorum, and maintain a heartbeat with the active node. In this situation, it is possible to consolidate as many as 8 to 10 passive nodes onto a single dual or quad processor piece of hardware. In the event of a failure of a single node or the need to induce a failure for maintenance on the active node, the ESX Guest will pick up the load and maintain its active status until failed back.

The drawback here is that you do not retain 100% failover capacity. In other words, if you happen to have a situation where multiple active nodes fail and your ESX host is underpowered, your users may experience a decline in performance.

Business Continuity Use Cases

Rarely will an organization use only one method for providing business continuity to its virtual infrastructure. We also realize that every organization is completely different in regards to SLA requirements and business processes.

Instead of providing use cases in terms of Small/Medium/Enterprise organizations, we want to take a slightly different approach and provide examples for different application workloads and SLA requirements based on what we have found to be the most common configurations during virtual infrastructure implementation. By leveraging the examples below you should be able to successfully build a solid business continuity plan that leverages the best that virtualization has to offer.

	BR	DR	HA	FT
Active Directory Controller				
Standard Application Server	I	X		
File Server	F/I	X		
Print Server	I	X		
Web Server	I		*	X
Development Server	I			
Small Database Server	F/I	X	X	
Enterprise Database Server	F/I	X		X
Messaging System	F/I	X		X

F= File, I=Image, *=See Description

Active Directory Domain Controller

Ok, so the first one is a trick. It is also one of the most commonly asked questions about providing business continuity in a virtual infrastructure. When dealing with an individual Active Directory server running as a virtual machine there is little to no need to go out of your way to provide business continuity for it. Unless you only have one Active Directory server in your environment, in which case you should close this book and proceed to beat yourself with it until you pass out, it will always be easier to deploy a new server and follow the proper steps to create a new domain controller and allow the existing AD infrastructure to replicate the proper data to it. If you are serious about Active Directory recovery, you should look into software products such as Quest's "Recovery Manager for Active Directory" which provides enhanced capabilities to recover individual Active Directory objects.

Standard Application Server

Most organizations run a lot of applications. While there is no definite mold that these easily fit into, we have seen that often times a majority of these application servers do not have a high enough SLA to be considered for high availability or fault tolerance. What happens more often than not is these applications are simply backed up using an image-level technology. Based on the fact that an image technology is used as opposed to a file-level technology, disaster recovery is inherent to the process. If something were to happen to one of these servers, having data that is up to 24 hours old is not a major issue. A nightly backup using an image backup technology works quite well for this type of system. Make sure you use differential backups, as this can save significant amounts of space if there are not a lot of changes occurring on these servers.

File Server

File servers are unique in the fact that they can benefit from having both an image backup and file-level backup performed. In this scenario, you should consider using an image technology to backup the operating system VMDK drive. Since the operating system drive is very rarely ever likely to change, make sure you leverage a differential technology with your image backups. There is a very good chance that there is a second VMDK file hosting the data and a

slightly smaller chance yet that it is an RDM. The best way to protect this system and provide the highest level of recoverability of the files being served is to use a file-level backup technology to capture the date contained in the data VMDK from within the operating system. This allows you to get the base operating system back up and running in the event of a disaster, but also gives you the best flexibility when recovering files when your users delete them on you.

Print Server

Let's face it; print servers are more often than not the bastard stepchildren of an organization. In many cases, you just find any system with extra processing capacity and throw some print queues on it. This makes them the perfect virtualization candidate. If using standalone virtual machines whose sole purpose is to manage print jobs, we don't care what files exist inside the VMDK files. All we care about is getting the print queues back up and running so your manager can print his multi-colored spreadsheet. A simple image-level backup is more than enough to protect this system. Unless you are an insurance company that I used to work for, you should not be adding enough print queues to the system to justify backing up these servers more than once a week. It's almost not even worth the effort to configure differential backups for a system such as this.

Web Server

This is where creating business continuity for an organization really starts to get fun. When you need to take off your Windows administrator hat and go argue with the network guys about what the best solution for protecting your web application is, it's a great day. In theory, web servers should remain quite static. A simple image-level backup on a regular basis is more than enough. Assuming you are running a web application that has a higher SLA requirement than most you will need to put some thought into how to keep the system available. The best solution is to build a load balanced farm so you have not only load balancing, but also fault tolerance built into the solution. There is some high availability consideration that should go into this design as well. What happens if you lose an entire datacenter and the only hosts serving your application no longer exist? For this reason alone it is worth using virtual machine replication to get at least one of your, *hopefully* identical, web servers from one datacenter and off to a disaster site. In the event that the worst happens, simply power on the des-

tination virtual machine, update your DNS entries, and possibly whip out a few quick clones of the system to help handle the load on the disaster side.

Development Server

In every organization there is some manager who stands up in the middle of a room when discussing business continuity and says "It's a development server, who cares if we don't get it back?" Being in the software business, I can easily tell you that developers care. Their sole purpose for being at your company is to develop software. If the systems that they are depending on to perform their daily functions are disabled for too long, they will get quite upset and you will be wasting a lot of money while they watch the clock tick. I'm not telling you to fire up the SAN replication by any means, but you most definitely should not simply ignore the fact that quite a few people rely on development servers day in and day out. Since developers are notorious for doing crazy things seemingly at random, a nightly image backup of the systems they most frequently use will help protect their long hours of coding and debugging.

Small Database Server

When we start talking about small database servers we are starting to hit our limit as far as what we are truly seeing virtualized in the real world. As with most systems, an image-level backup should be taken on a fairly regular basis. This will help in the event that the database becomes corrupted or someone accidentally deletes and entire table by running a bad query. It should be noted, however, that image-level backups do not treat transactional systems well. These systems are often in a crash-consistent state, and while that shouldn't cause a major issue, it is better to be safe than sorry. For that reason, it is highly recommended that a proper file-level agent be used inside the guest operating system to backup the database the proper way. This will provide a failback in case issues arise from being in a crash-consistent state. Depending on what type of data is being served by these database systems, it may not be a bad idea to perform virtual machine-level replication to a remote site. Many times, the loss of the 10-15 minutes of data will not be the end of the world in a system such as this, and it will allow you to recover the system with minimal effort in the event of a failure.

Enterprise Database Server

If you decide to place an enterprise database server within your virtual infrastructure you better have all of your bases covered. As with a small database server, you will want to leverage an image-level backup of the operating system and use a typical file agent to properly backup the database. There is a good chance that the database files will reside on a physical RDM, so this is the only way you will be able to get at the data anyway. Instead of looking at virtual machine-level replication for these systems you should just focus on SAN replication to get your data to an alternate datacenter. If this were a true enterprise-level database system, you would have a high enough communication link to actually be able to provide clustering services between sites for the database. Personally, I have yet to see anyone brave enough to put a system that critical inside of a virtual machine.

Messaging System

Email is one of the most important applications to any organization. It's quite sad how much we depend on being able to instantly communicate with one another. If it weren't for email, I'd actually have to talk to people on a regular basis. Because of the critical nature of messaging systems they should actually be handled quite similarly to enterprise database servers. Since email is a transactional system as well, don't trust your recovery to only an image level backup; use the proper file-level agents to get your messages to safety. In fact, laws in many countries force you to backup and store your mail messages in a very specific manner. Don't interfere with doing this the right way. Again, many organizations will have mechanisms in place to have remote clustering available to keep email systems up and running. Just don't forget that you'll need more than one MX record for your domain on your DNS server. Assign a significantly lower priority to the MX record that correlates with your disaster site.

Use VMware HA

You will notice that there was no mention of using VMware HA in any example on the previous pages. This is because VMware HA should be carefully planned and configured on EVERY cluster in your virtual infrastructure when it is available. For as basic as the idea of powering on virtual machines during a host failure sounds, not having to manually perform the task and balance your

own workloads while doing it will allow you to sleep comfortably at night. Make sure you review Chapter 4 to ensure you are designing VMware HA to meet your needs on a day to day basis, and don't focus on it for DR.