



White Paper

Implementing High Availability for Virtual Servers

Abstract

The need to protect virtual machine environments against both planned and unplanned outages is increasing as the number of business critical applications deployed within VMs grows. This paper shows how the use of high availability clustering software can ensure application uptime and presents various configurations in which virtual machines can be used in HA clusters, with an emphasis on describing solutions that are enabled by virtual machine technologies. The advantages and disadvantages of the

various configurations are described, and some guidance is provided for system administrators who need to determine the best configurations for their own uses.

SteelEye, SteelEye Technology, and LifeKeeper are registered trademarks of SteelEye Technology, Inc. Other brand and product names used herein are for identification purposes only and may be trademarks of their respective companies.

This document is for informational purposes and is believed to be correct at the time of publication. However, SteelEye Technology does not guarantee the accuracy of the information and reserves the right to change the document at any time. SteelEye Technology makes no warranties, expressed or implied, in this document.

Copyright © 2006
SteelEye Technology, Inc.
Palo Alto, CA U.S.A.
All Rights Reserved

Table of Contents

Introduction.....	4
Virtual Machine Technology and Environments	5
High Availability Software Tools.....	6
Application Monitoring.....	6
Local Recovery	6
Machine Failover.....	6
High Availability Configurations with Virtual Machines.....	7
Multi-system configurations	7
Physical-to-virtual	7
Virtual-to-physical	8
Virtual-to-virtual.....	9
Single-system configurations	10
Mixed OS configurations.....	10
Technical I/O Considerations.....	11
Network Interfaces	11
Storage Interfaces.....	12
Alternate Storage Configurations.....	12
Conclusion	13
About SteelEye Technology® and LifeKeeper®	14

Introduction

Virtual machine technologies including VMware ESX Server, Microsoft Virtual Server, Xen, and IBM POWER Virtualization are being deployed increasingly for the hardware consolidation and configuration flexibility benefits that they offer. However, there is an inherent and not well acknowledged danger in virtualization. Virtualization **will** decrease the availability of your applications and services unless it is properly combined with a high availability clustering solution.

This decrease in application and service availability when using virtualization results from two factors. First, virtualization increases the potential for failure because of the increased complexity and additional potential sources of failure in the virtualization environment itself, including the virtual machine host layer and the dual or split device driver model.

Second, virtualization increases the scope of failures. In a virtualized environment, more applications and services are at risk of failure due to the failure of a single physical machine. This is the old “all your eggs in one basket” problem.

Despite the very real danger of increased application downtime, there is actually a growing misconception that virtualization technology can increase the availability of your applications, i.e. that virtualization provides high availability. This myth is based in large part on the virtual machine migration capabilities offered with some virtualization technologies. These features allow a virtual machine to be moved from one physical machine to another, either by stopping it completely on one machine and starting it from scratch on another, or by using a suspend/resume technique. But these virtual machine migration technologies have several significant shortcomings which make them unsuitable for insuring the availability of your applications.

- Virtual machine migration is designed primarily for planned maintenance activities, not failure scenarios. In most products, a suspend/resume type of migration must be initiated by an administrator, and doesn't work at all if the virtual machine has already failed. And a start-from-scratch migration is slow, as the entire OS must be booted and all of the applications started in the new location.
- Virtual machine migration may simply move the problem, if the cause of the application or system failure is inherent in the configuration of the virtual machine itself.
- Virtual machine migration operates only at the level of the entire virtual machine and its operating system. A complete HA solution requires the monitoring and management of individual applications and the services on which they depend. Without that level of protection, failures of individual applications and services go undetected and unresolved, and failures at the virtual machine or OS level require moving all applications together, without the flexibility of distributing them across your remaining systems.
- Virtual machine migration requires the use of virtualization on all of the machines in your environment, and requires that those machines be identical or nearly identical in their hardware configuration. This puts extreme limits on the flexibility of your configuration choices for achieving high availability.
- Virtual machine migration generally requires the use of shared storage (e.g. SAN). This means that they cannot easily support the migration of virtual machines across geographically separated systems, such as for disaster recovery purposes.

The solution to all of these shortcomings is the implementation of a high availability clustering solution alongside the virtualization technology.

Virtual machine technology allows multiple instances of an operating system to run simultaneously within a single physical computing platform. This technology is advancing rapidly both in function and in use,

and when combined with high availability clustering software such as SteelEye Technology's LifeKeeper, offers tremendous flexibility to define and build cluster configurations and meet requirements that were previously beyond reach with purely physical machine environments. In this paper, we will explore the ways in which products such as SteelEye's LifeKeeper can be deployed in a virtual machine environment to combine the power of both technologies into an extremely flexible range of high availability clustering solutions.

The first section below offers an introduction to virtual machine technology. Following this introduction is a detailed look at a range of potential high availability configurations using virtual machines. The next section covers some special considerations regarding I/O in a virtual machine environment.

Virtual Machine Technology and Environments

A virtual machine environment requires some form of *virtual machine host* layer, which interfaces with the *physical machine* hardware and creates one or more *virtual machines* in which an operating system can run. Each of these virtual machines is provided with an environment that appears to the operating system as an actual physical machine, but the resources of that environment, such as processors, memory and I/O devices, are virtual resources provided by the virtual machine host layer. Figure 1 below shows the relationship of the components in a virtual machine environment.

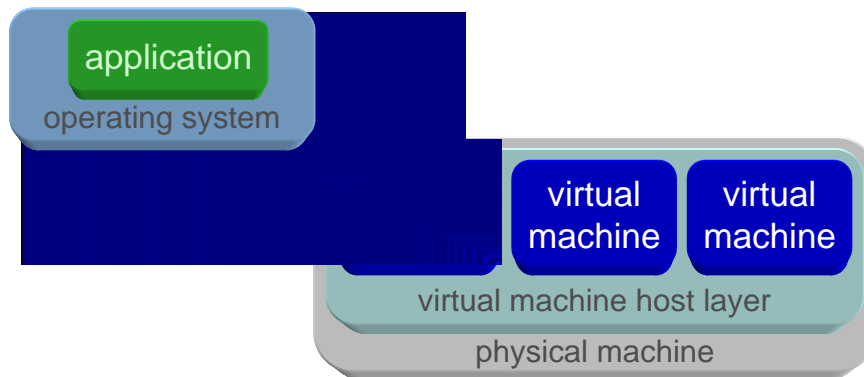


Figure 1: Relationship between virtual machine components

The virtual machine host layer may be implemented as a combination of hardware, firmware and software, or totally in software. The available virtualization technologies offer varying degrees of flexibility in the way that they map physical machine resources into the environment of each virtual machine. These mappings may include any or all of the following approaches:

- Dedicating or assigning a distinct physical resource to a specific virtual machine.
- Subdividing a physical resource and providing some percentage of its capacity to each of a set of virtual machines.
- Dynamically reassigning a dedicated physical resource from one virtual machine to another.
- Dynamically modifying the allocated percentages of a physical resource, and reassigning them to a set of virtual machines.

The dynamic modification of resource assignments to virtual machines, as in the last two items above, generally also requires some degree of support within the operating system running in the virtual machine.

Because the virtual machines run independently from one another, it is possible to stop, start, and reboot them separately, and even to run different operating systems within the virtual machines on a single physical machine.

High Availability Software Tools

High availability software consists of several tools for application monitoring, local recovery (optional), and machine failover that are usually provided in an integrated package.

Good supporting software for your high availability solution should be completely hardware-agnostic, so that it functions inside a virtual machine just as if it were running inside an OS directly on a physical system. There are several advantages of this behavior. First, you gain a tremendous ease of use advantage from the fact that administering the software inside a virtual machine is the same as administering it outside a virtual machine. Second, deployment planning is simplified, since you only need to consider failover speed, redundancy, and cost. You might save a little money up front with a software solution that functions differently inside and outside of virtual machines, but you'll lose money due to higher costs of training and application administration.

Application Monitoring

The application monitoring tool detects unplanned events such as application, network, or hardware failures. Once the failure is detected, the tool typically fires an event to initiate either a local recovery or a failover. This tool is not needed for planned events such as system upgrades and applying software service packs.

Local Recovery

The local recovery tool attempts to bring the application back into service on the same machine, usually in response to an unplanned application, network, or hardware failure. If this attempt fails to bring the application back into service, the machine failover tool is typically called. Local recovery has the advantage of being a quicker method of getting applications back into service by removing the need to manage cross-system dependencies.

Machine Failover

The machine failover tool stops the application and brings it into service on a second machine. It can be called directly in the case of planned events such as maintenance, or by the application monitor or local recovery tools to handle unplanned failures. This action is what is typically thought of as high availability clustering.

High Availability Configurations with Virtual Machines

High availability clustering involves the grouping of two or more systems into a cluster for the primary purpose of monitoring for and responding to failure events, related to either some component of the application environment or the entire system. When a failure of some type is detected, the high availability clustering software seeks to return the application environment to a working status as quickly as possible, thereby increasing the availability of that application. As discussed above, failure recovery operations generally involve either attempts to restore an application and its environment to a working status on the system on which the application was originally running (*a local recovery*), or moving an application and its environment to another system in the cluster. The latter operation is typically called a *failover*.

For the purposes of our discussion of high availability configurations with virtual machines, we will focus on the ways that virtual machines may be involved in failover operations in order to reduce the risk of application failure. Local recovery of an application within the same virtual machine is identical to a physical machine recovery, so that aspect will not be discussed here.

The section that follows presents several configurations which can be built using a combination of virtualization and high availability clustering technologies. We will cover multi-system, single-system and mixed OS configurations and show how the availability of applications can be increased using these joint solutions.

Multi-system configurations

Physical-to-virtual

The physical-to-virtual failover configuration involves a physical, non-virtualized system serving as the primary server for an application with failovers occurring to a backup virtual machine within some other physical system. This configuration is shown in Figure 2 below.

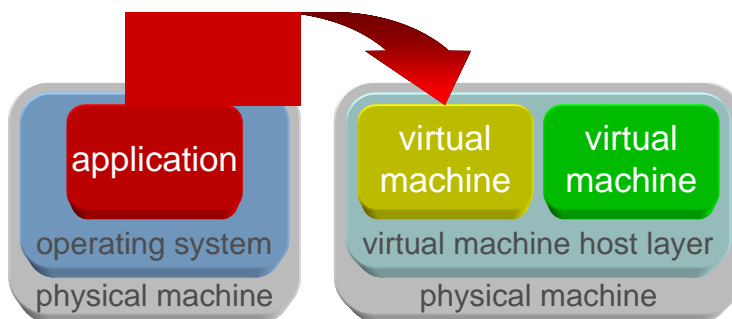


Figure 2: Physical-to-virtual

This type of configuration is most useful to allow a single physical system, running multiple virtual machines within it, to serve as the backup for multiple physical, non-virtualized system. This is an adaptation of the many-to-one cluster configuration, where N active systems are physical, non-virtualized servers and the single backup system is a physical machine running N virtual machines, each serving as the backup system for one of the active physical servers. An example of this type of many-to-one configuration, with N=2, is shown in Figure 3 below.

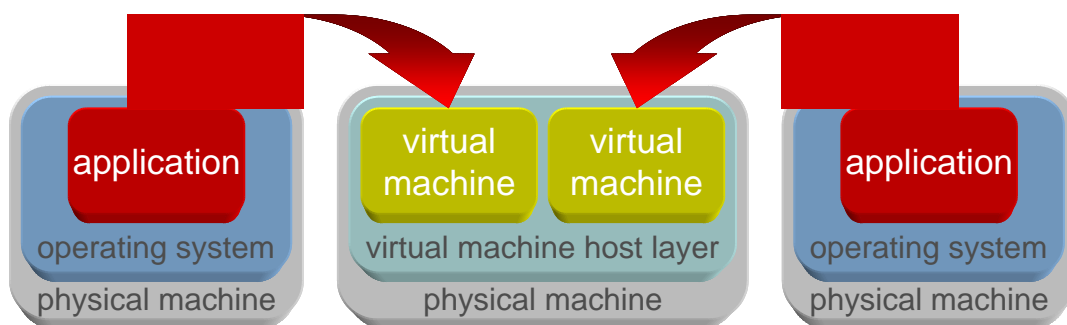


Figure 3: Many-to-1 configuration

This technique allows many-to-one cluster support for an application that could not normally be configured as such due to a limitation of only a single application instance running within an operating system environment. An example of such an application might be Microsoft Exchange. Typically, only one copy of Exchange can be run on a server, thus requiring a dedicated backup server for every primary Exchange server within a cluster configuration. By using virtual machines as failover targets, however, a single physical machine can now act as backup for any number of active Exchange systems, up to the limit of the power of the system hosting the virtual machines.

This use of virtualization technology in combination with high availability clustering allows for the building of HA clusters with fewer systems than would typically be required and subsequently allows for the greater use of hardware resources.

Virtual-to-physical

This configuration is essentially the same as the previous one, but with the primary and backup system roles reversed such that application failover is from a virtual machine to a separate physical, non-virtualized system. An example use of this type of configuration is a 1-to-many cluster, where there is 1 active system, implemented as a physical server running multiple virtual machines, and a number of backup systems, each a physical, non-virtualized system acting as the backup for one of the virtual machines on the single active system. This type of configuration, depicted in Figure 4 below, might be used if you wanted to use a single, powerful, virtualized machine as the normal primary node for all of your application instances, while using lower-cost physical machines as the backup systems for those instances. You might do this to centralize the management of your applications to a single platform while reusing existing hardware for the few occasions that the application will be running on the backup server.

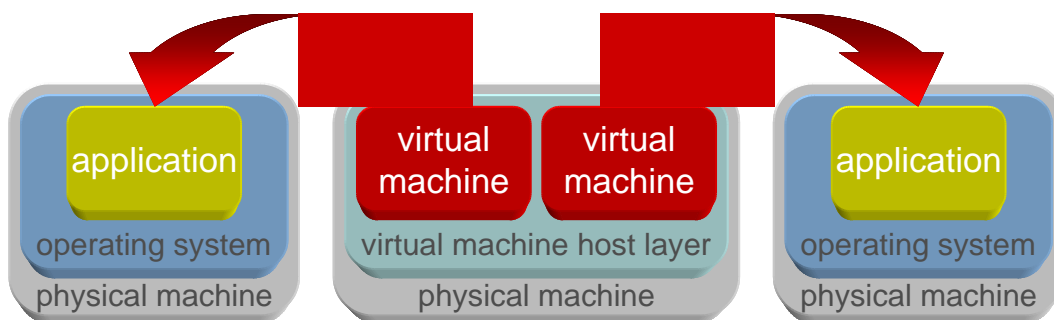


Figure 4: 1-to-Many configuration

Virtual-to-virtual

In the configuration shown in Figure 5 below failovers occur between virtual machines running on different physical systems.

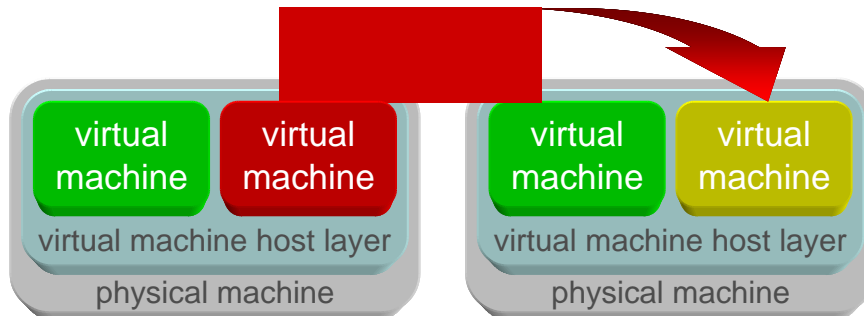


Figure 5: Virtual-to-virtual

This configuration represents the most typical and straightforward means of providing high availability protection for virtual machine environments. In its simplest form, each virtual machine on a primary server can be paired with a corresponding backup virtual machine on a backup server for failover purposes, thereby providing complete protection for the entire primary server. Many other variations are possible, of course, including active/active configurations in which some applications are normally active in a virtual machine on the first system, while others are normally active in a virtual machine on the second system, and with both physical systems acting as backup for the other using corresponding backup virtual machines.

For certain applications that do not allow multiple instances of the application to run within the same operating system environment, this type of configuration may be the only means available for deploying an active/active cluster supporting multiple instances of the application. Because virtual machine technology allows multiple operating system instances to be run within the same system, it becomes possible for multiple instances of such an application to also run within the same system, each in its own isolated environment. This allows configuring two physical systems to run different instances of the same application with each also serving as backup for the other. An example of this type of configuration is shown in Figure 6 below.

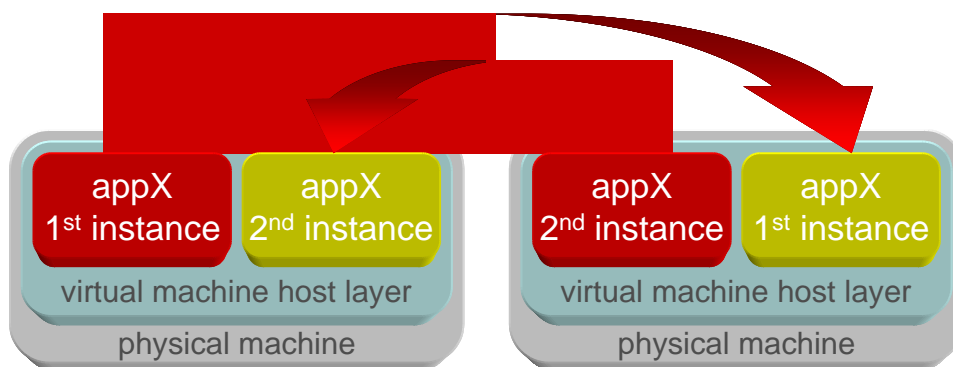


Figure 6: Multiple Instance Active/Active

There are two types of virtual-to-virtual failover – application failover, and virtual machine failover. If your supporting software is hardware-agnostic, application failover between virtual machines is the same as failover between physical machines, which is discussed elsewhere. Virtual machine failover – failover of the entire virtual machine, with all supported applications – is roughly equivalent to failover of a generic application.

Virtual machine failover can be simpler to install and administer than application because the software is protecting a generic operating system, rather than a specific application for which it may need extra configuration information. However, because the software configuration is not tailored to the specific application being protected, the monitoring and failover support will be correspondingly less sophisticated, possibly leading to undetected failures, and local recovery will be impossible.

Virtual machine failover configurations may be the best option for protecting simple applications, especially those for which no off-the-shelf high availability software is available, for situations where uncomplicated system administration is important. For more robust support, it's best to go with application failover, with software that's tailored to the specific application.

Single-system configurations

Single-system configurations, allowing failover from one virtual machine to another virtual machine within the same physical machine, as depicted in Figure 7 below, are only possible with virtual machine technology.

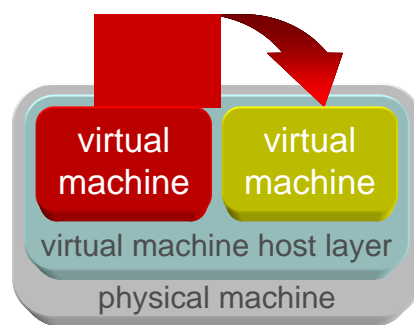


Figure 7: Single-system configuration

Single-system configurations allow assets to be moved from one virtual machine to another within the same physical system. Failover speed can be higher because communication is confined to a single physical machine. Single-system configurations can be very useful for planned events – application maintenance, evaluations, and demonstrations. However, the physical machine itself is a single point of failure for the application environment, so these are not robust configurations for handling unplanned failures, unless you're trying to provide high availability support for an application that's very prone to software failures.

There are two types of single-system failover – application failover and virtual machine failover. If your supporting software is hardware-agnostic, application failover between virtual machines is the same as failover between physical machines, which is discussed elsewhere. Virtual machine failover – failover of the entire virtual machine, with all supported applications – is roughly equivalent to failover of a generic application, as discussed above. Since single-system configurations are suitable mainly for planned events, the lack of sophisticated monitoring support is unlikely to be important, so virtual machine failover is likely to be the best solution.

Mixed OS configurations

One of the most powerful advantages of using virtual machine technology in a high availability clustering environment is the ability to support both Linux and Windows operating system environments within the same set of clustered servers. A business running a mixed environment of both Linux and Windows would traditionally have to allocate separate clusters of machines to their Linux and Windows applications. But because virtual machine technology creates separate virtual machine environments for multiple operating system instances within a single physical system, it is now possible to run Linux and Windows simultaneously on a single piece of hardware. This opens the door for implementing cluster configurations such as the ones below.

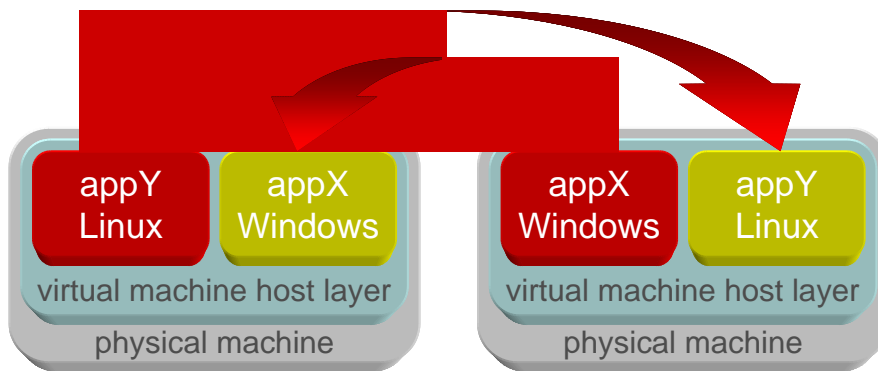


Figure 8: Active/Active with Mixed OS

Figure 8 above depicts an active/active cluster configuration consisting of two physical systems, each running both Linux and Windows virtual machines. This allows one of the physical systems to act as the primary system for a Linux application, while the other physical machine acts as the primary for a Windows application. The two physical systems can each serve as the backup for the other due to the virtual machine division within them. Instead of the minimum of four physical systems that would normally be required to provide HA support for the two applications in a non-virtualized environment, virtual machine technology can allow the same level of support using only two physical systems.

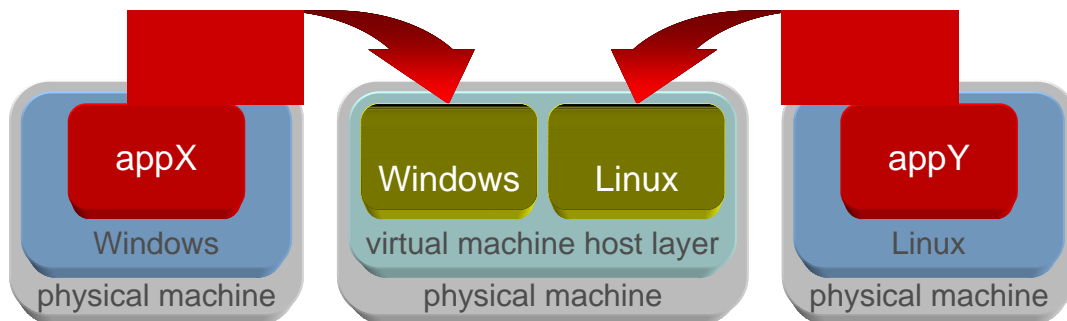


Figure 9: Single Failover target for mixed OS environment

Similarly, Figure 9 above shows a many-to-one cluster configuration consisting of two physical, non-virtualized primary systems, one running Linux and one running Windows, and a single physical machine running two virtual machines serving as the backup for both. This allows the building of an “availability appliance” which is able to act as the failover target for all active systems within the IT environment, whether they are running Windows or Linux.

Technical I/O Considerations

A primary technical consideration in the implementation of high availability clustering in a virtual machine environment is with I/O – specifically network connectivity and access to storage. It is important to understand how each virtual machine is configured for network and storage access and the implications that configuration has on management of those network and storage resources.

Network Interfaces

Most virtual machine technologies employ a technique in which the virtual machine host layer manages and interfaces with the physical network interface cards in the server and provides each of the virtual machines with its own virtual network interface that maps onto the physical interface. Thus, the virtual

machines effectively share the physical network interface card, but in a way that each virtual machine appears to have its own dedicated network interface. If there are multiple physical network cards (or ports) available in the machine, each of those cards or ports can be shared, so that each virtual machine can have multiple network interfaces.

Some virtual machine technologies allow a physical network interface card or port to be assigned to a specific virtual machine. In this configuration, the virtual machine uses the network interface directly and exclusively, and multiple physical network interfaces are required in order for multiple virtual machines in the same physical server.

HA management systems typically interact with network interfaces to create and manage virtual IP addresses that can be moved from one system to another in order to provide seamless client access to the applications being protected. This interaction is exactly the same within virtual machines and physical machines and works equally well with both the shared and dedicated network interface mechanisms described above.

Storage Interfaces

Similar to network interfaces, most virtual machine technologies provide a means by which the virtual machine host layer manages and interfaces with direct-attached storage devices on behalf of the virtual machines. The virtual machine host layer may allow control over which storage elements can be accessed by the virtual machines at the level of individual disk devices or volumes, or it may allow entire storage controllers (e.g. fibre channel host bus adapters (HBAs) or SCSI controllers) to be shared with and accessed by the virtual machines. And as with network interfaces, some virtual machine technologies also allow a storage controller to be dedicated or assigned to a specific virtual machine for its exclusive use.

Shared, direct-attached storage, in the form of fibre channel SANs or parallel SCSI, is an important component of most high availability clustering configurations. By placing application data on, for example, a fibre channel disk array that is accessible from all of the machines in a cluster, that data is made available to the application regardless of which cluster node the application is running on. This enables the movement of the application from one node to the other by high availability clustering software such as LifeKeeper, in response to failure conditions or administrative requirements.

When using shared storage in this manner, the use of adequate mechanisms for correctly identifying shared disks or volumes in the cluster and for managing and enforcing access by the correct cluster nodes is extremely important. LifeKeeper takes on the responsibility for both of these functions in the high availability clusters that it manages. On Linux, LifeKeeper uses primitives within the SCSI command set for acquiring unique IDs for shared storage devices and for establishing and releasing SCSI reservations on those devices as a means of access control. On Windows, standard APIs within a filter are used for disk identification and IO fencing.

Alternate Storage Configurations

In addition to shared storage in the form of fibre channel SANs or parallel SCSI, LifeKeeper supports two other means by which application data can be made available to multiple nodes in an HA cluster. Both of these mechanisms are fully supported in a virtual machine environment.

The first of these options, Network Attached Storage (NAS) configurations, is similar to SANs, except that the storage device is attached to the communications network rather than a separate fibre channel storage-only network, and the device is accessed by the compute nodes with a network file protocol such as NFS. Because this access uses standard network interfaces, it behaves exactly the same in a virtual machine environment as it would in a purely physical machine.

The second option for application data sharing is data replication. By maintaining an exact block-for-block replica of a local disk device on a remote system, an accurate copy of the application data can be made available for application failovers. LifeKeeper supports multiple mechanisms for performing data replication, SteelEye Data Replication (SDR) is available on both Linux and Windows to perform volume-based synchronous and asynchronous replication. The native Linux replication mechanism DRBD is also supported for configurations where it may be the preferred replication engine. Both of these facilities

perform replication over standard network interfaces, and therefore work well in a virtual machine environment.

Conclusion

The use of virtual servers to host business critical applications is expanding rapidly as organizations experience both significant cost savings and optimized resource utilization through virtualization supported server consolidation. The use of virtualization technologies, however, can decrease the availability of applications and services. The need for integrating an automated high availability clustering solution to monitor and protect the virtual servers becomes evident when the increased risk of failure is understood. Through this integration, a variety of configurations can be built using a combination of physical and virtual systems, each providing the benefits of virtualization without sacrificing application availability.

SteelEye LifeKeeper's support for virtual machine environments brings together the power and flexibility of virtual machine technology with the most proven and fully-featured high availability clustering technology available for Linux and Windows.

About SteelEye Technology® and LifeKeeper®

SteelEye Technology is a leading provider of application and data availability management solutions for business continuity and disaster recovery on Linux and Windows. The SteelEye LifeKeeper family of software products enables enterprises of all sizes to ensure continuous availability of business-critical applications and data on industry standard AMD, Intel and Power servers and storage systems whether those applications are running direct on physical systems or within virtual machines.

The SteelEye family of data replication, high availability clustering and disaster recovery solutions enable enterprises of all sizes to ensure continuous availability of business-critical applications, data and supporting IT infrastructure. SteelEye LifeKeeper offers enterprise-grade reliability while simplifying implementation with certified and easy-to-deploy solutions for a wide range of applications and databases running on the most common Linux distributions and on Windows 2000/2003.

SteelEye LifeKeeper provides fault-resilience for the most demanding enterprise deployments at a low cost by supporting configurations built on commodity servers and storage and by removing the need to make application environment changes. LifeKeeper delivers advanced monitoring of servers, networks, operating systems, applications and data along with automated healing capabilities based on policies set by system administrators to ensure that applications and data are always available. LifeKeeper is ideal in environments that run mission critical database and ERP operations and is today protecting Oracle, DB2, MySQL, SQL Server, SAP NetWeaver, Exchange, Apache and Sendmail in major corporations worldwide. As customers look to deploy enterprise applications in concert with virtualization technology, they can be assured of the high availability benefits of LifeKeeper across the spectrum of AMD, Intel and IBM Power platforms.

Supporting the widest range of server and storage configurations, from simple LAN clusters, to clusters across a Fibre Channel SAN, to shared-nothing clusters built around data replication, LifeKeeper lets you deploy the solution that best meets recovery time objectives today while allowing for expansion in the future. With support of both Linux and Windows and the ability to manage both environments from a single console, LifeKeeper can protect the entire IT infrastructure regardless of platform.

With over 6,000 licenses installed in companies of all sizes world-wide, LifeKeeper has established itself as the gold-standard for High Availability Clustering, Data Replication and Disaster Recovery. SteelEye software products are available worldwide and may be purchased directly from SteelEye or through the SteelEye international network of business partners.

Learn more and download a free evaluation version at www.steeeye.com.

About Open Minds High Availability Solutions

Open Minds provide software solutions, support, consultancy and training for High Availability, and Disaster Recovery solutions for IT systems. A strong team of technical consultants offer experience gained from many years of successfully installing and supporting LifeKeeper solutions throughout the world in order to implement effective and easily maintained High Availability Solutions.

Open Minds is the registered SteelEye Competence and Support Centre for the United Kingdom and Ireland. We have longstanding partnerships with VAR's, Software Houses and Systems Integrators to provide a full technical backup including pre-sales, installation services and support required to fully implement a high-availability solution.

For more information, please visit www.openminds.co.uk



This document was created with Win2PDF available at <http://www.win2pdf.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.