System Platform in a Virtualized Environment Implementation Guide

Publication date: 10/12/2018

Contact Information

AVEVA Group plc
High Cross
Madingley Road
Cambridge
CB3 OHB. UK

https://sw.aveva.com/

For information on how to contact sales, customer training, and technical support, see https://sw.aveva.com/contact.

Contents

CHAPTER 1

Welcome

This guide describes the implementation of System Platform in a virtualized environment, using Microsoft Hyper-V technology, failover clustering, and other strategies to create High Availability, Disaster Recovery, and High Availability with Disaster Recovery capabilities. You can view this document online or you can print it, in part or whole, by using the print feature in Adobe Acrobat Reader.

In This Chapter

Documentation Conventions

This documentation uses the following conventions:

Convention	Used for
Initial Capitals	Paths and file names.
Bold	Menus, commands, dialog box names, and dialog box options.
`Monospace`	Code samples and display text.

Technical Support

Before you contact Technical Support, refer to the relevant section(s) in this documentation for a possible solution to the problem. If you need to contact technical support for help, have the following information ready:

- The type and version of the operating system you are using.
- Details of how to recreate the problem.
- The exact wording of the error messages you saw.
- Any relevant output listing from the Log Viewer or any other diagnostic applications.
- Details of what you did to try to solve the problem(s) and your results.
- If known, the Technical Support case number assigned to your problem, if this is an ongoing problem.

CHAPTER 2

Getting Started with Virtualization

Virtualization technologies are becoming high priority for IT administrators and managers, software and systems engineers, plant managers, software developers, and system integrators.

Mission-critical operations in both small- and large-scale organizations demand availability—defined as the ability of the user community to access the system—along with dependable recovery from natural or man-made disasters. Virtualization technologies provide a platform for High Availability and Disaster Recovery solutions.

In This Chapter

Using this Guide

Note: This guide has been updated to include support for Hyper-V 3.0 and vSphere 6.0.

The purpose of this guide is to help you to implement System Platform in a virtualized environment, including:

- Implementing some of the new features in Microsoft Windows Server 2008 R2 and higher
- Implementing High Availability, Disaster Recovery, or High Availability with Disaster Recovery utilizing Windows Server virtualization technologies such as Hyper-V
- Implementing High Availability and Disaster Recovery using VMware technology

This chapter introduces and defines virtualization concepts in general, as well as in a System Platform context. This chapter also defines a basic workflow and planning framework for your virtualization implementation.

Subsequent chapters describe features of Windows Server and Hyper-V, and provide an outline of how to use them. Among the topics discussed are how to configure High Availability, Disaster Recovery, High Availability with Disaster Recover, how create virtual images, and how to implement a virtualized backup strategy.

Subsequent chapters also provide test and performance metrics for a wide variety of system configurations, including Recovery Time Objective (RTO) and Recovery Point Objective (RPO).

Understanding Virtualization

Virtualization is the creation of an abstracted or simulated—virtual, rather than actual—version of something, such as an operating system, server, network resource, or storage device. Virtualization technology abstracts the hardware from the software, extending the life cycle of a software platform.

In virtualization, a single piece of hardware, such as a server, hosts and coordinates multiple guest operating systems. No guest operating system is aware that it is sharing resources and running on a layer of virtualization software rather than directly on the host hardware. Each guest operating system appears as a complete, hardware-based OS to the applications running on it.

Definitions

This implementation guide assumes that you and your organization have done the necessary research and analysis and have made the decision to implement System Platform in a virtualized environment. Such an environment can take advantage of advanced virtualization features including better utilization of hardware resources, High Availability and Disaster Recovery. In that context, we'll define the terms as follows:

- Virtualization can be defined as **creating a virtual, rather than real, version of System Platform or one of its components, including servers, nodes, databases, storage devices, and network resources**.
- High Availability (HA) can be defined as a **primarily automated System Platform design and associated services implementation which ensures that a pre-defined level of operational performance will be met during a specified, limited time frame**.
- Disaster Recovery (DR) can be defined as **the organizational, hardware and software preparations for System Platform recovery or continuation of critical System Platform infrastructure after a natural or human-induced disaster**.

While these definitions are general and allow for a variety of HA and DR designs, this implementation guide focuses on virtualization, an indispensable element in creating the redundancy necessary for HA and DR solutions.

The virtualized environment described in this guide is based on Microsoft Hyper-V technology incorporated in the Windows Server 2008 R2 and higher operating systems, and on VMware technology.

Types of Virtualization

There are eight types of virtualization:

Hardware	A software execution environment separated from underlying hardware resources. Includes hardware-assisted virtualization, full and partial virtualization and paravirtualization.
Memory	An application operates as though it has sole access to memory resources, which have been virtualized and aggregated into one memory pool. Includes virtual memory and memory virtualization.
Storage	Complete abstraction of logical storage from physical storage
Software	Multiple virtualized environments hosted within a single operating system instance. Related is a virtual machine (VM) which is a software implementation of a computer, possibly hardware-assisted, which behaves like a real computer.

Mobile	Uses virtualization technology in mobile phones and other types of wireless devices.
Data	Presentation of data as an abstract layer, independent of underlying databases, structures, and storage. Related is database virtualization, which is the decoupling of the database layer within the application stack.
Desktop	Remote display, hosting, or management of a graphical computer environment—a desktop.
Network	Implementation of a virtualized network address space within or across network subnets.

Virtualization Using a Hypervisor

Virtualization technology implements a type of hardware virtualization using a hypervisor, permitting a number of guest operating systems (virtual machines) to run concurrently on a host computer. The hypervisor is so named because it exists above the usual supervisory portion of the operating system.

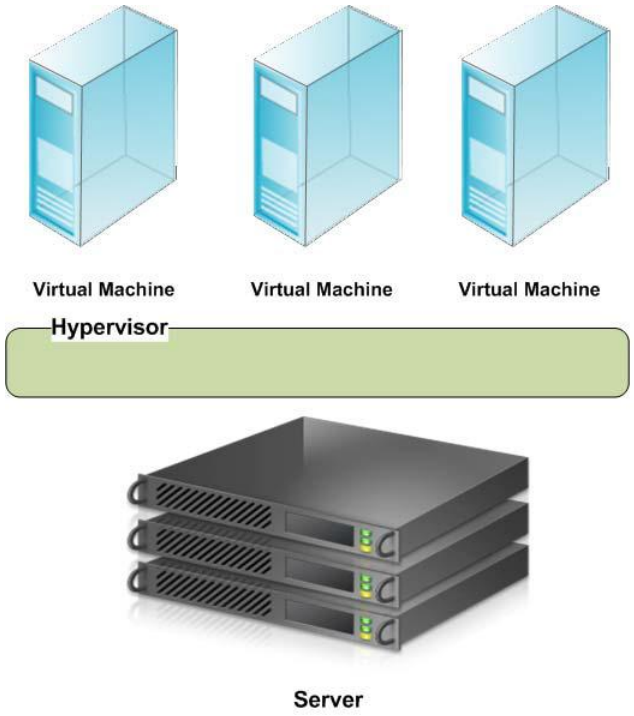
Hypervisor Classifications

There are two classifications of hypervisor:

- **Type 1**: Also known as a bare metal hypervisor, runs directly on the host hardware to control the hardware and to monitor the guest operating systems. Guest operating systems run as a second level above the hypervisor. ESXi for VMware vSphere, and Hyper-V for Windows Server are examples of type 1 hypervisors.
- **Type 2**: Also known as a hosted hypervisor, runs within a conventional operating system environment as a second software level. Guest operating systems run as a third level above the hypervisor. VMWorkstation is an examples of a type 2 hypervisor.

Hypervisor Architecture

Hyper-V and VMware implement Type 1 hypervisor virtualization, in which the hypervisor primarily is responsible for managing the physical CPU and memory resources among the virtual machines. This basic architecture is illustrated in the following diagram.



Virtualizing System Platform

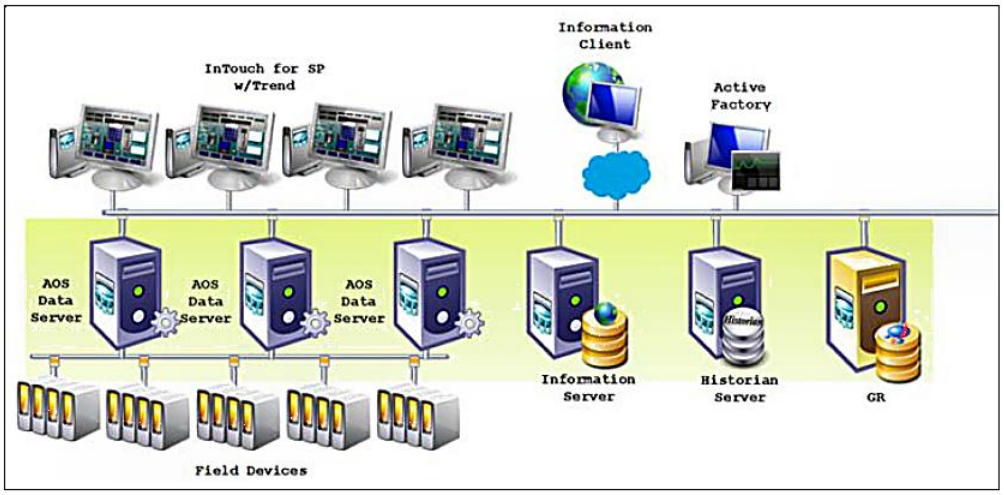
Abstraction Versus Isolation

With the release of InTouch 10.0, supporting the VMWare ESX platform, we became one of the first companies to support virtual machine operation of industrial software. VMware ESX is referred to as a "bare metal" virtualization system. The virtualization is run in an **abstraction layer**, rather than in a standard operating system.

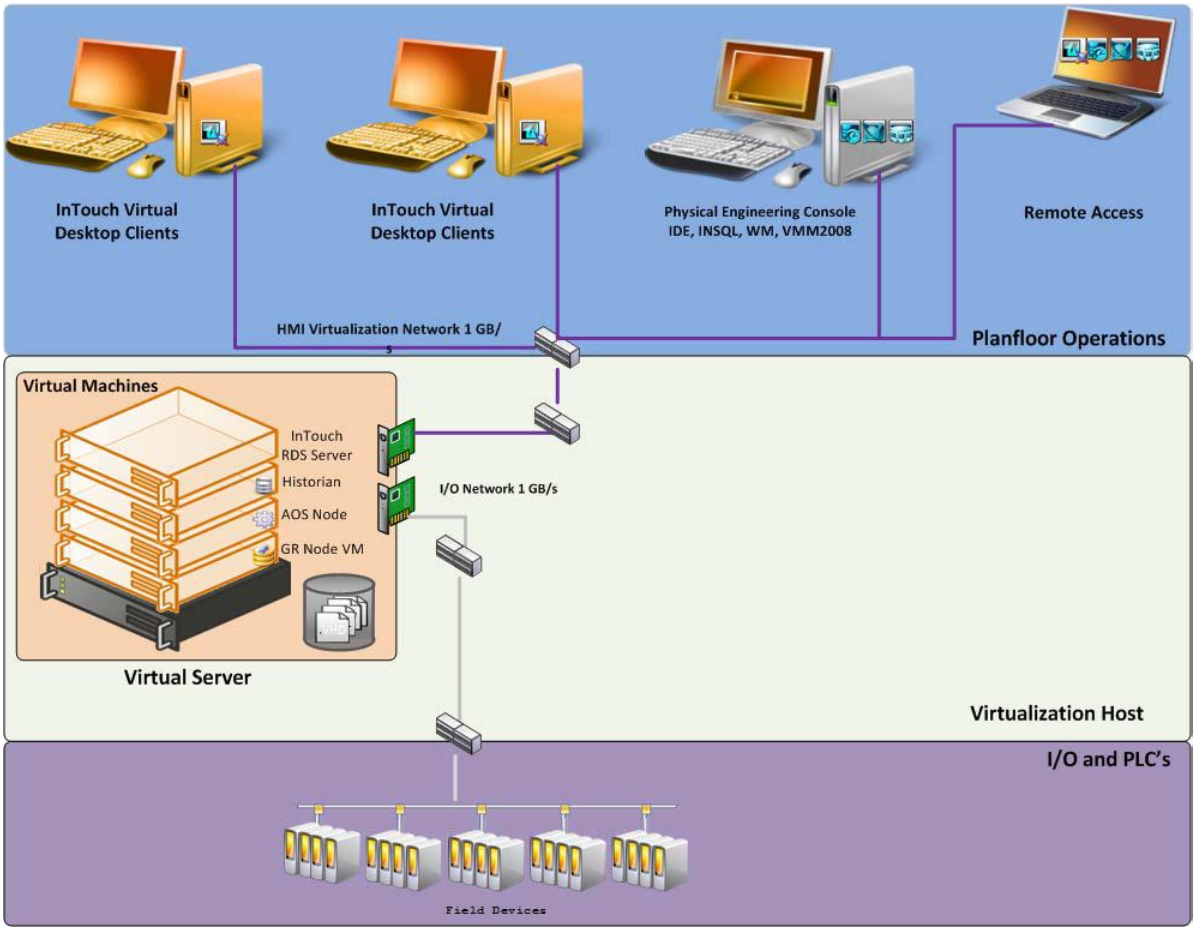
Microsoft takes a different approach to virtualization. Microsoft Hyper-V is a hypervisor-based virtualization system. The hypervisor is essentially an **isolation layer** between the hardware and partitions which contain guest systems. This requires at least one parent partition, which runs Windows Server 2008 or higher.

Note: An abstraction layer is a layer with drivers that make it possible for the virtual machine (VM) to communicate with hardware (VMware). In this scenario the drivers need to be present for proper communication with the hardware. With an isolation layer, the VM uses the operating system, its functionality, and its installed drivers. This scenario does not require special drivers. As a comparison, the abstraction layer in VMware is 32MB and in Hyper-V it is 256kb.

The following diagram shows a common System Platform topology, non-virtualized:



The following diagram shows the same environment virtualized:



Levels of Availability

When planning a virtualization implementation—for High Availability, Disaster Recovery, Fault Tolerance, and Redundancy—it is helpful to consider levels or degrees of redundancy and availability, described in the following table.

Level	Description	Comments
Level 0 Redundancy	No redundancy built into the architecture for safeguarding critical architectural components	Expected failover: None
Level 1 Cold Stand-by Redundancy	Redundancy at the Application Object level Safeguards single points of failure at the OI Server level or AOS redundancy.	Expected failover: 10 to 60 seconds Availability 99%: Annual uptime impact is approximately four days down per year
Level 2 High Availability (HA)	• With provision to synchronize in real-time • Uses virtualization techniques • Can be 1-n levels of hot standby • Can be geographically diverse (DR) • Uses standard OS and nonproprietary hardware	Expected failover: Uncontrolled 30 seconds to 2 minutes, DR 2 - 7 minutes Availability 99.9%: Annual uptime impact is approximately 8 hrs down per year
Level 3 Hot Redundancy:	Redundancy at the application level typically provided by Schneider Electric controllers. For example, hot backup of Schneider Electric software such as Alarm System.	Expected failover: Next cycle or single digit seconds Availability 99.99%: Annual uptime impact is approximately 52 minutes down per year.

Level	Description	Comments
Level 4 Lock-step Fault Tolerance (FT)	Provides lock-step failover	Expected failover: Next cycle or without loss of data. Availability 99.999%: Annual uptime impact is considered as "continuous availability" with downtime less than 5 minutes per year. A 99.999% availability is considered the "gold standard." For System Platform, this would be a Marathon-type solution, which also can be a virtualized system.

A typical system without virtualization, using a High Availability implementation, might attain Level 1 availability with a good server. With a good infrastructure, you can achieve Level 3 availability by using virtualized High Availability.

A typical system could reach Level 4 availability by using virtualization with more than two possible hosts, RAID storage options, dual power supplies, teamed NICs, and by implementing application monitoring. This allows the application to restart on another host if a crash occurs.

Performance of failover can vary and is dependent on the quality and implementation of the HA architecture.

About RTO and RPO

The Recovery Time Objective (RTO) is the duration of time within which a business process must be restored to its service level after a disaster or other disruption in order to avoid a break in business continuity.

A Recovery Point Objective (RPO), is defined by business continuity planning. It is the maximum tolerable period in which data might be lost from an IT Service due to a major incident.

For System Platform in a normal, non-virtualized, implementation, depending on the system size, RTO could be hours or days on a complete loss of the system. The RPO would be 45 seconds or more for Application Server redundancy, or more—in terms of hours— for non-redundant components such as Terminal Servers for InTouch HMI or Information Server.

For System Platform in a virtualized High Availability implementation that uses double-host configuration, the measured recovery time is as follows:

- RTO is less than 2 minutes for the complete system. Controlled RTO is seconds, with un-controlled RTO less than 2 minutes.
- RPO is within 2 minutes.

High Availability

About HA

High Availability refers to the availability of resources in a computer system following the failure or shutdown of one or more components of that system.

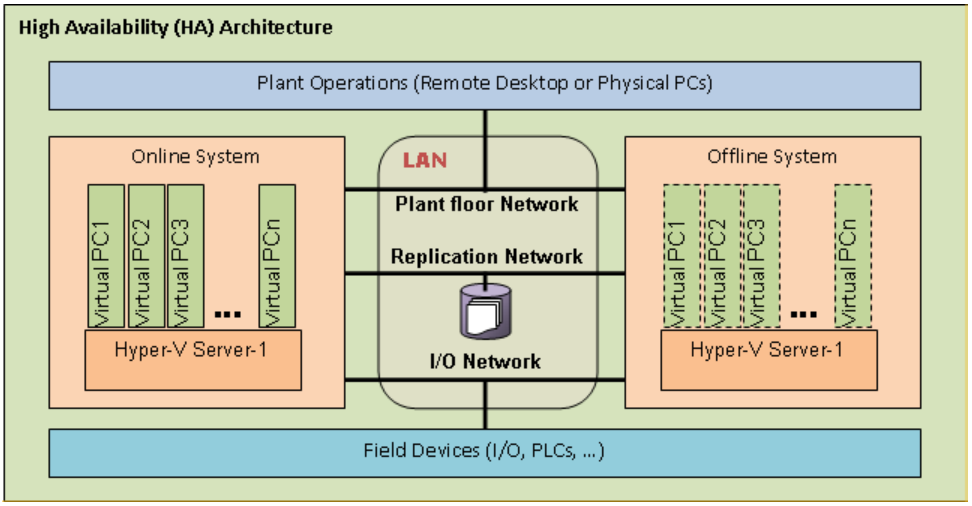
At one end of the spectrum, traditional HA has been achieved through custom-designed and redundant hardware. This solution produces High Availability, but has proven to be very expensive.

At the other end of the spectrum are software solutions designed to function with off-the-shelf hardware. This type of solution typically results in significant cost reduction, and has proven to survive single points of failure in the system.

High Availability Scenarios

The basic HA architecture implementation described in this guide consists of an online system including a Hyper-V or VMware Server and a number of virtual PCs, linked by a LAN to an offline duplicate system. The LAN accommodates a number of networks including a plant floor network linked to plant operations, an I/O network linked to field devices, and a replication network linked to storage.

The following example shows Hyper-V implementation.



This basic architecture permits a number of common scenarios.

IT maintains a virtual server	• A system engineer fails over all virtual nodes hosting System Platform software to back up the virtualization server over the LAN. • For a distributed system, the system engineer fails over all virtual nodes to back up the virtualization server over a WAN. • IT performs the required maintenance, requiring a restart of the primary virtualization server.

Virtualization server hardware fails	• The primary virtualization server hardware fails with a backup virtualization server on the same LAN. • For a distributed system, the virtualization server hardware fails with a backup virtualization server over WAN. **Note:** This scenario is a hardware failure, not software. A program that crashes or hangs is a failure of software within a given OS.
A network fails on a virtual server	• Any of the primary virtualization server network components fail with a backup virtualization server on the same LAN, triggering a backup of virtual nodes to the backup virtualization server. • Any of the primary virtualization server network components fail with a backup virtualization server connected via WAN, triggering a backup of virtual nodes to the backup virtualization server over WAN.

For these scenarios, the following expectations apply:

- For the maintenance scenario, all virtual images are up and running from the last state of execution prior to failover.
- For the hardware and network failure scenarios, the virtual images restart following failover.
- For LAN operations, you should see operational disruptions for approximately 2-15 seconds (LAN operations assumes recommended speeds and bandwidth. For more information refer to *"Networks"*).
- For WAN operations, you should see operational disruptions for approximately 2 minutes (WAN operations assumes recommended speeds and bandwidth. For more information refer to *"Networks"*).

Note: The disruption spans described here are general and approximate. For specific metrics under a variety of scenarios, see the relevant Recovery Time Objective (RTO) and Recovery Point Objective (RPO) sections in chapters 2, 3, and 4.

Disaster Recovery

About DR

Disaster Recovery planning typically involves policies, processes, and planning at the enterprise level, which is well outside the scope of this implementation guide.

DR, at its most basic, is all about data protection. The most common strategies for data protection include the following:

- Backups made to tape and sent off-site at regular intervals, typically daily.
- For the hardware and network failure scenarios, the virtual images restart following failover
- For the hardware and network failure scenarios, the virtual images restart following failover
- Backups made to disk on-site, automatically copied to an off-site disk, or made directly to an off-site disk.

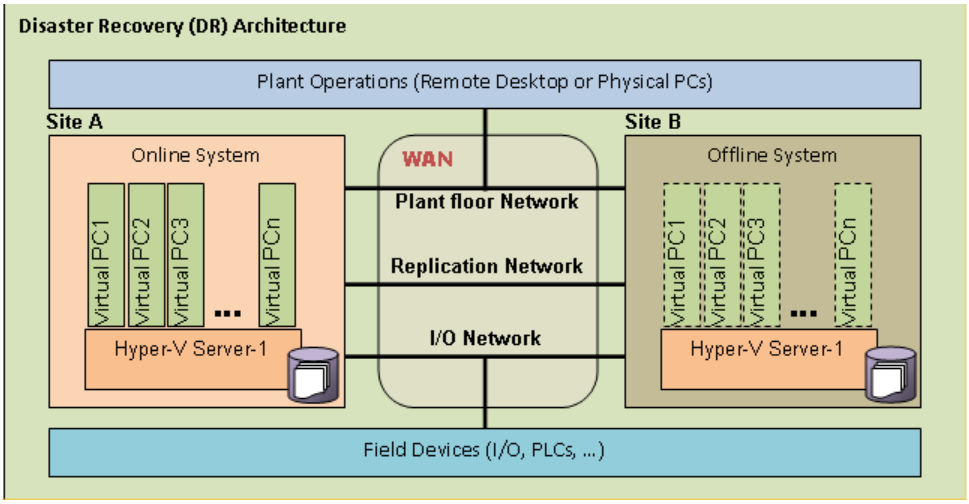
- Replication of data to an off-site location, making use of storage area network (SAN) technology. This strategy eliminates the need to restore the data. Only the systems need to be restored or synced.
- High availability systems which replicate both data and system off-site. This strategy enables continuous access to systems and data.

The System Platform virtualized environment implements the fourth strategy—building DR on an HA implementation.

Disaster Recovery Scenarios

The basic DR architecture implementation described in this guide builds on the HA architecture by moving storage to each Hyper-V or VMware server, and moving the offline system to an off-site location.

The following example shows Hyper-V implementation.



The DR scenarios duplicate those described in *"High Availability Scenarios"*, with the variation that all fail-overs and backups occur over WAN.

High Availability with Disaster Recovery

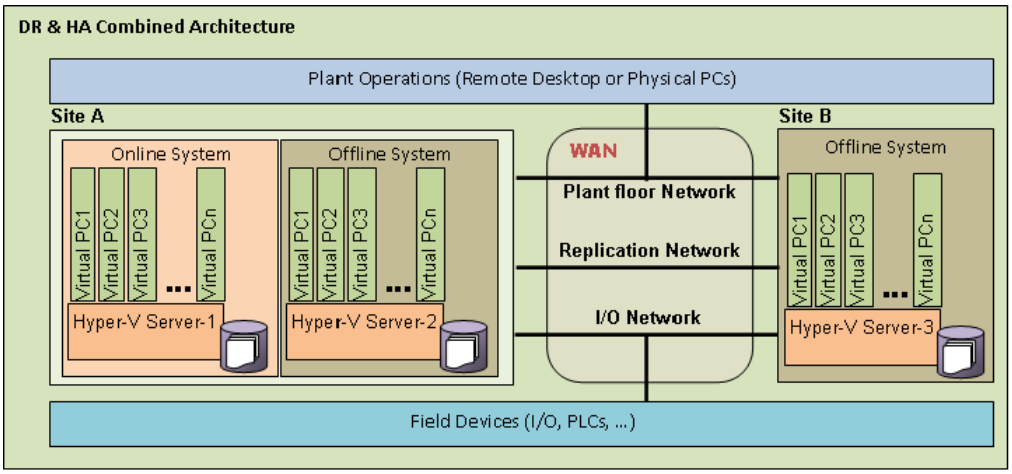
About HADR

The goal of a High Availability and Disaster Recovery (HADR) solution is to provide a means to shift data processing and retrieval to a standby system in the event of a primary system failure.

Typically, HA and DR are considered as individual architectures. HA and DR combined treat these concepts as a continuum. If your system is geographically distributed, for example, HA combined with DR can make it both highly available and quickly able to recover from a disaster.

HADR Scenarios

The basic HADR architecture implementation described in this guide builds on both the HA and DR architectures adding an offline system plus storage at "Site A". This creates a complete basic HA implementation at "Site A" plus a DR implementation at "Site B" when combined with distributed storage.



The scenarios and basic performance metrics described in *"High Availability Scenarios"* also apply to HADR.

Planning the Virtualized System

Planning an System Platform virtualization implementation is a three-step process—based upon an understanding of the available technology:

1. Assess your existing System Platform installation
2. Assess virtualization requirements
3. Extend your assessment to define HA, DR, or HADR

For more information about configuring HA, DR, and HADR, see the following chapters in this guide.

Chapter 2, "Implementing High Availability Using Hyper-V."

Chapter 3, "Implementing High Availability Using vSphere."

Chapter 4, "Implementing Disaster Recovery Using Hyper-V."

Chapter 5, "Implementing Disaster Recovery Using vSphere."

Chapter 6, "Implementing High Availability and Disaster Recovery Using Virtualization."

Planning Information for a Hyper-V Implementation

About Hyper-V

Hyper-V was initially released with Service Pack 1 (SP1) for Windows Server 2008 R2. It has been updated, with improved capabilities and specifications, in subsequent Windows Server releases. In addition, Hyper-V is also available as a standalone product. The following summarizes key Hyper-V features:

Dynamic Memory	Dynamic Memory enables better utilization of Hyper-V host memory resources by balancing how memory is distributed between running virtual machines. Memory can be dynamically reallocated between different virtual machines in response to the changing workloads of these machines.
Live Migration	Data-centers with multiple Hyper-V physical hosts can move running virtual machines to the best physical computer for performance, scaling, or optimal consolidation without affecting users.
Hardware Support for Hyper-V Virtual Machines	Depending on the operating system, Hyper-V supports up to 320 logical processors in the host processor pool, allowing greater VM density per host, and more flexibility in assigning CPU resources to VMs, and enabling migration across a broader range of server host hardware.
Cluster Shared Volumes	Hyper-V uses Cluster Shared Volumes (CSV) storage to simplify and enhance shared storage usage. CSV enables multiple Windows Servers to access SAN storage using a single consistent namespace for all volumes on all hosts.
Cluster Node Connectivity Fault Tolerance	CSV architecture improves cluster node connectivity fault tolerance that directly affects VMs running on the cluster. The CSV architecture implements a mechanism, known as dynamic I/O redirection, where I/O can be rerouted within the failover cluster based on connection availability.
Enhanced Cluster Validation Tool	A Best Practices Analyzer (BPA) is included for all major server roles, including Failover Clustering. This analyzer examines the best practices configuration settings for a cluster and cluster nodes.
Management of Virtual Datacenters	The number of VMs tends to proliferate much faster than physical computers because machines typically do not require a hardware acquisition. This makes efficient management of virtual data centers more imperative than ever.
Virtual Networking Performance	Hyper-V leverages networking technologies to improve overall VM networking performance.
Performance & Power Consumption	Enhancements have been added that reduce virtual machine power consumption.

Networking Support	Jumbo Frames, previously available in non-virtual environments, has been extended to work with VMs. The Virtual Machine Queue (VMQ) feature allows physical computer network interface cards (NICs) to use direct memory access (DMA) to place the contents of packets directly into VM memory, increasing I/O performance.
Dynamic VM storage	Hyper-V supports hot plug-in and hot removal of storage. This allows the addition and removal of both VHD files and pass-through disks to existing SCSI controllers for VMs.
Broad OS Support	Broad support for simultaneously running different types of operating systems, including 32-bit and 64-bit systems across different server platforms, such as Windows, Linux, and others.
Network Load Balancing	Hyper-V includes virtual switch capabilities. This means virtual machines can be easily configured to run with Windows Network Load Balancing (NLB) Service to balance load across virtual machines on different servers.
Hardware Sharing Architecture	With virtual service provider/virtual service client (VSP/VSC) architecture, Hyper-V provides improved access and utilization of core resources, such as disk, networking, and video.
Virtual Machine Snapshot	Hyper-V provides the ability to take snapshots of a running virtual machine so you can easily revert to a previous state, and improve the overall backup and recoverability solution.
Extensibility	Standards-based Windows Management Instrumentation (WMI) interfaces and APIs in Hyper-V enable independent software vendors and developers to quickly build custom tools, utilities, and enhancements for the virtualization platform.

VM and Hyper-V Limits in Windows Server

Refer to Microsoft documentation for maximum values that apply to Hyper-V. These values vary significantly between releases and versions of Windows Server. For example, for Windows Server 2008 R2 Standard, the limits are 4 virtual processors and 64 GB memory per virtual machine. For Windows Server 2012 and Windows Server 2012 R2, these limits are significantly increased to 64 virtual processors and 1 TB memory per virtual machine.

For more information on Hyper-V and its maximums, refer to the Microsoft TechNet resources on Hyper-V, available via the following link:

https://technet.microsoft.com/en-us/windowsserver/dd448604.aspx
https://technet.microsoft.com/en-us/windowsserver/dd448604.aspx

Planning Information for a VMware Implementation

About vCenter Server and vSphere

VMware vCenter Server is a simple and efficient way to manage multiple VMware vSpheres. It provides unified management of all the hosts and VMs in your datacenter from a single console monitoring the performance of clusters, hosts, and VMs.One administrator can manage 100 or more workloads.

VMware vCenter Servers allow you to provide VMs and hosts using standardized templates. Use of templates helps to ensure compliance with vSphere host configurations and host and VM patch levels with automated remediation. With proactive management, VMware vCenter Server allows you to dynamically provide new services, allocate resources, and automate high availability.

VMware vCenter Server enables management of a large scale enterprise, more than 1,000 hosts and up to 10,000 VMs, from a single console.

Extensibility

VMware vCenter Server's open plug-in architecture supports a broad range of additional capabilities that can directly integrate with vCenter Server, allowing you to easily extend the platform for more advanced management capability in areas such as:

- Capacity management
- Compliance management
- Business continuity
- Storage monitoring
- Integration of physical and virtual management tools

VMware vSphere 6.0 Editions

VMware vSphere 6 includes scalability improvements and security enhancements over previous releases. It is available in three editions: Standard, Enterprise, and Enterprise Plus. One instance of VMware vCenter Server, sold separately, is required for all VMware vSphere deployments. ESXi is the type 1 (bare metal) hypervisor included in the vSphere suite of products.

For information about each edition’s features and capabilities, refer to the VMware website:

https://www.vmware.com/products/vSphere/compare
https://www.vmware.com/products/vSphere/compare

VM and Virtual Server Limits in VMware

The following tables show maximum values for VMs and for a server running vSphere 6.0. By understanding the limits of the hardware, software, and virtual machines, you can better plan your System Platform virtualized environment.

vSphere Virtual Machine Maximums (ESXi 6.0)

Component	Maximum	Notes
Virtual CPUs per VM	128	32 previously

Component	Maximum	Notes
RAM per VM	4 TB	1 TB previously
IDE controllers per VM	1	Supports two channels (primary and secondary) each with a master and slave device.
SCSI adapters per VM	4	Any combination of supported SCSI virtual storage controllers. Four Paravirtual SCSI adapters may be used only if the virtual machine boots from a device attached to an IDE controller, or from the network.
Virtual SCSI targets per virtual SCSI adapter	15	Any combination of disk, CD-ROM, or VMDirectPath SCSI target
Virtual hard disk capacity	62 TB	2TB previously
Size of physical disks attached to a VM	Varies	Maximum size is determined by the guest operating system.
Checkpoints (Snapshots)	32	The actual number depends on the available storage and may be lower. Each snapshot is stored as a file that consumes physical storage.
Virtual network adapters	10	Any combination of supported virtual NICs.
Virtual floppy controllers	1	
Virtual floppy devices	2	BIOS is configured for 1 floppy device.
USB controllers	1	Supports USB 1.x, 2.x,and 3.x devices
USB devices connected to a virtual machine	20	
Parallel ports	3	
Serial (COM) ports	32	4 previously

vSphere ESXi 6.0 Host Maximums

Component	Maximum	Notes
Logical CPUs per host	480	160 previously
Virtual machines per host	1024	512 previously
Virtual CPUs per host	4096	2048 previously
Memory	6 TB	12 TB is supported on specific OEM certified platforms. Refer to the VMware Compatibility Guide for additional information.
Virtual disks per host	2048	
Physical network adapters	32	tg3 1 Gb Ethernet ports (Broadcom) • 16 with NetQueue enabled • 32 with NetQueue disabled NetQueue is enabled by default in vSphere 6.0
Maximum active ports per host	1016	
Virtual network switch ports per host	4096	vSphere Standard and Distributed Switch

VMware Requirements

VMware ESXi 6.0 Installation Requirements

The minimum requirements to install the vSphere Hypervisor (ESXi 6.0) are listed in the following table. For complete installation requirements and additional information, refer to vSphere Installation and Setup instructions:

https://pubs.vmware.com/vSphere-60/topic/com.vmware.ICbase/PDF/vSphere-esxi-vcenter-server-60-installation-setup-guide.pdf

For additional information refer to the VMWare Compatibility Guide:

http://www.vmware.com/resources/compatibility http://www.vmware.com/resources/compatibility

Component	Requirement
64-bit Processor	ESXi 6.0 installs and run only on servers with 64-bit x86 CPUs. ESXi 6.0 requires a host machine with at least two cores. ESXi 6.0 requires the NX/XD bit to be enabled for the CPU in the BIOS.
RAM	4GB RAM minimum; 8 GB RAM minimum to fully leverage ESXI features and run machines in typical production environments.
Network Adapters	One or more Gigabit or faster Ethernet controllers.
Installation and Storage	SCSI disk or a local, non-network, RAID LUN with unpartitioned space for the virtual machines. For Serial ATA (SATA), a disk connected through supported SAS controllers or supported on-board SATA controllers. SATA disks will be considered remote, not local. These disks will not be used as a scratch partition by default because they are seen as remote. **Note:** You cannot connect a SATA CD-ROM device to a virtual machine on an ESXi 6.0 host. IDE emulation mode must be used to enable a SATA CD-ROM device.
Support for 64-bit Virtual Machines	Support for hardware virtualization (Intel VT-x or AMD RVI) must be enabled.
Storage Systems	Refer to the VMWare Compatibility Guide: http://www.vmware.com/resources/compatibility

VMware Disaster Recovery Requirements

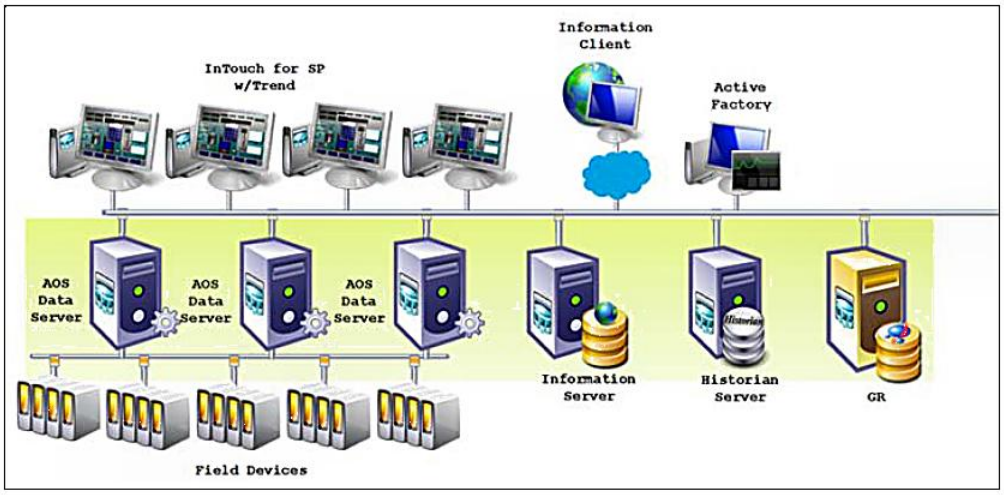
VMware Disaster Recovery (DR) implementations require installation of VMware vCenter Site Recovery Manager, Standard or Enterprise edition.

Scalability limits of the vCenter Site Recovery Manager editions are:

- Standard Edition: 75 virtual machines
- Enterprise Edition: Unlimited, subject to the product's technical scalability limits.

Assessing Your System Platform Installation

In most cases, a System Platform installation already exists. You will need to create an assessment of the current architecture. You can start with a basic topology diagram, similar to the following:



Once you have diagramed your topology, you can build a detailed inventory of the system hardware and software.

Microsoft Planning Tools

Microsoft tools to assist with virtualization assessment and planning:

- Microsoft Assessment and Planning Toolkit (MAP)

  The MAP toolkit is useful for a variety of migration projects, including virtualization. The component package for this automated tool is available for download from Microsoft at the following address:

  http://www.microsoft.com/en-us/download/details.aspx?id=7826

- Infrastructure Planning and Design Guides for Virtualization (IPD)

  The IPD Guides from Microsoft provide a series of guides specifically geared to assist with virtualization planning. They are available for download from Microsoft at the following address:

  http://technet.microsoft.com/en-us/solutionaccelerators/ee395429

VMware Planning Tools

VMware tools to assist with virtualization assessment and planning:

- VMware Capacity Planner

  The VMware Capacity Planner is a business and IT tool for datacenter and desktop capacity planning.

  http://www.vmware.com/products/capacity-planner/overview.html

- VMware SAN System Design and Deployment Guide

  This guide describes how to design and deploy virtual infrastructures using VMware technology.

  http://www.vmware.com/files/pdf/techpaper/SAN_Design_and_Deployment_Guide.pdf

- VMware Infrastructure 3 Planning

  This guide is specific to planning virtualization using Hewlett-Packard computer equipment. It offers considerable insight into planning, architecture, and deployment.

  http://www.vmware.com/support/pubs/vi_pubs.html

Sizing Recommendations for Virtualization

This section provides sizing guidelines and recommended minimums for System Platform installations.

For a only implementation, you can use these minimums and guidelines to size the virtualization server or servers that will host your System Platform configuration.

Cores and Memory

Spare Resources

The host server should always have spare resources of 25% above what the guest machines require.

For example, if a configuration with five nodes requires 20GB of RAM and 10 CPUs, the host system should have 25GB of RAM and 13 CPUs. If this is not feasible, choose the alternative closest to the 25% figure, but round up so the host server has 32GB of RAM and 16 cores.

Hyper-Threading

Hyper-Threading Technology can be used to extend the amount of cores, but it does impact performance. An 8-core CPU will perform better than a 4-core CPU that is Hyper-Threading.

Storage

It is always important to plan for proper Storage. A best practice is to dedicate a local drive or virtual drive on a Logical Unit Number (LUN) to each of the VMs being hosted. We recommend SATA or higher interfaces.

Recommended Storage Topology

To gain maximum performance, the host OS also should have a dedicated storage drive. A basic storage topology would include:

- Host storage
- VM storage for each VM
- A general disk

  This disk should be large enough to hold snapshots, backups, and other content. It should not be used by the host or by a VM.

Recommended Storage Speed

Boot times and VM performance are impacted both by storage bandwidth and storage speed. Faster is always better. Drives rated at 7200 rpm perform better than those rated at 5400 rpm. Solid-state drives (SSDs) perform better than 7200-rpm drives.

Keep in mind that multiple VMs attempting to boot from one hard drive will be slow, and your performance will experience a significant degrade. Attempting to save on storage could well become more costly in the end.

Networks

Networking is as important as any other component for the overall performance of the system.

Recommended Networking for Virtualization

If virtualization is your only requirement, your network topology could include the following elements:

- Plant network
- Storage network
- Virtualization network.

A best practice is to establish, on every node, an internal-only Static Virtual Network. In the event that the host and the guest VMs become disconnected from the outside world, you will still be able to communicate through an RDP session independent of external network connectivity.

Recommended Networking for HA

If HA is your requirement, then we recommend using fast, dedicated drives for local use. In the case of a Storage Area Network (SAN), we recommend using iSCSI 1GB/s as a minimum configuration.

A higher-performance configuration would be an FO connection to the storage at 10GB/s. For HA, we recommend a dedicated network for virtualization at 1GB/s. This will ensure fast transfers under different migration scenarios.

Recommended Minimums for System Platform

Following are approximate numbers of nodes to define small, medium, and large systems.

- Small: 1–3 nodes
- Medium: 4–8 nodes
- Large: More than 8 nodes

The following table provides recommended minimums for System Platform configurations.

	Cores	RAM	Storage
Small Systems			
GR Node	2	2	100
Historian	2	2	250
Application Server	2	2	100
RDS Servers	2	2	100
Information Servers	2	2	100
Historian Clients	2	2	100
Medium and Large Systems			
GR Node	4	4	250
Historian	4	4	500
Application Server	2–4	4	100

	Cores	RAM	Storage
RDS Servers	4–8	4–8	100
Information Server	4	4	100
Historian Clients	2	4	100

After installation of the server, you will start from scratch, or you can use the existing installation. A free tool on Microsoft TechNet called Disk2vhd supports extracting a physical machine to a VHD file. The Disk2vhd tool is available for download from Microsoft at the following address:

http://technet.microsoft.com/en-us/sysinternals/ee656415

Another tool you can use to migrate physical machines into to a virtual environment is Virtual Machine Manager. This tool is available for purchase from Microsoft. For more information, see the following Microsoft address:

https://www.microsoft.com/en-us/download/details.aspx?id=10712

A VMware tool for disk conversion is the vCenter Converter Standalone for P2V Conversion, available from VMware as a free download at the following address:

https://www.vmware.com/tryvmware/?p=converter&rct=j&q=vmware%20converter&source=web&cd=6&sqi=2&ved=0CEoQFjAF&url=http://www.vmware.com/go/getconverter&ei=4XIPT7ePB7CPigLR0OzSDQ&usg=AFQjCNH3Et0HlSZPzkw2VZxLVZoNZ_yY5g

Defining High Availability

To define a High Availability implementation, you need to plan for the following requirements:

- Server specification doubles

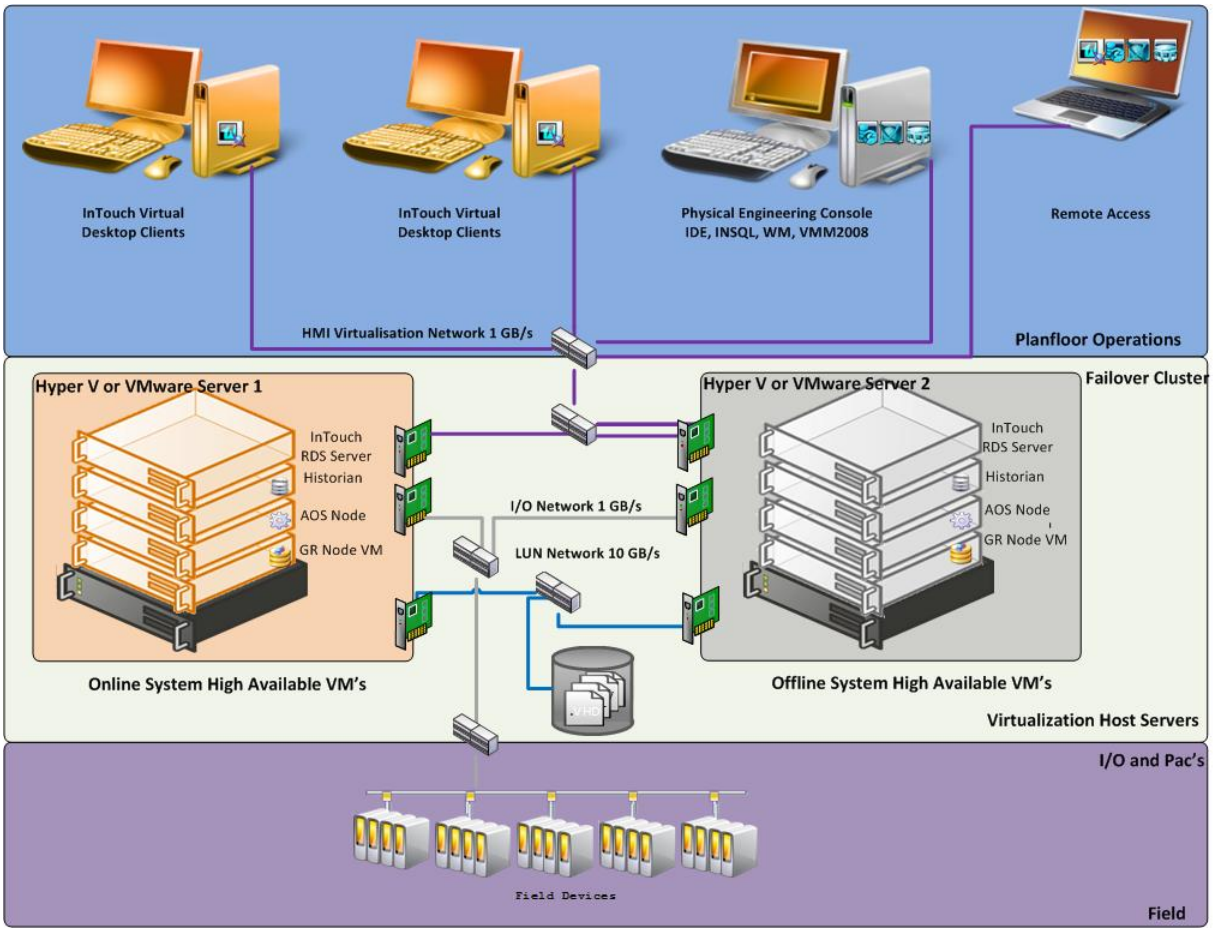
  Double the baseline configuration is required for shadow nodes in the Failover Cluster.

- Minimum OS requirements increase

  Hyper-V failover is supported only on Windows Server 2008 R2 and higher Enterprise operating system editions.

  Also, Hyper-V live migration, remote applications, and other features are available only if the host machines are Windows Server 2008 R2 and higher editions.

The following shows a System Platform HA implementation:



To implement HA, we strongly recommend the use of a SAN configured with the sizing guidelines and recommendations outlined in the preceding section.

Defining Disaster Recovery

To define a Disaster Recovery implementation, you need to plan for the following requirements:

- Adding a second server set with the same specifications as the first

  The second server set moves to the off-site location and connects over LAN or (more likely) WAN. Hyper-V Replica provides asynchronous replication of Hyper-V VMs on a second server.

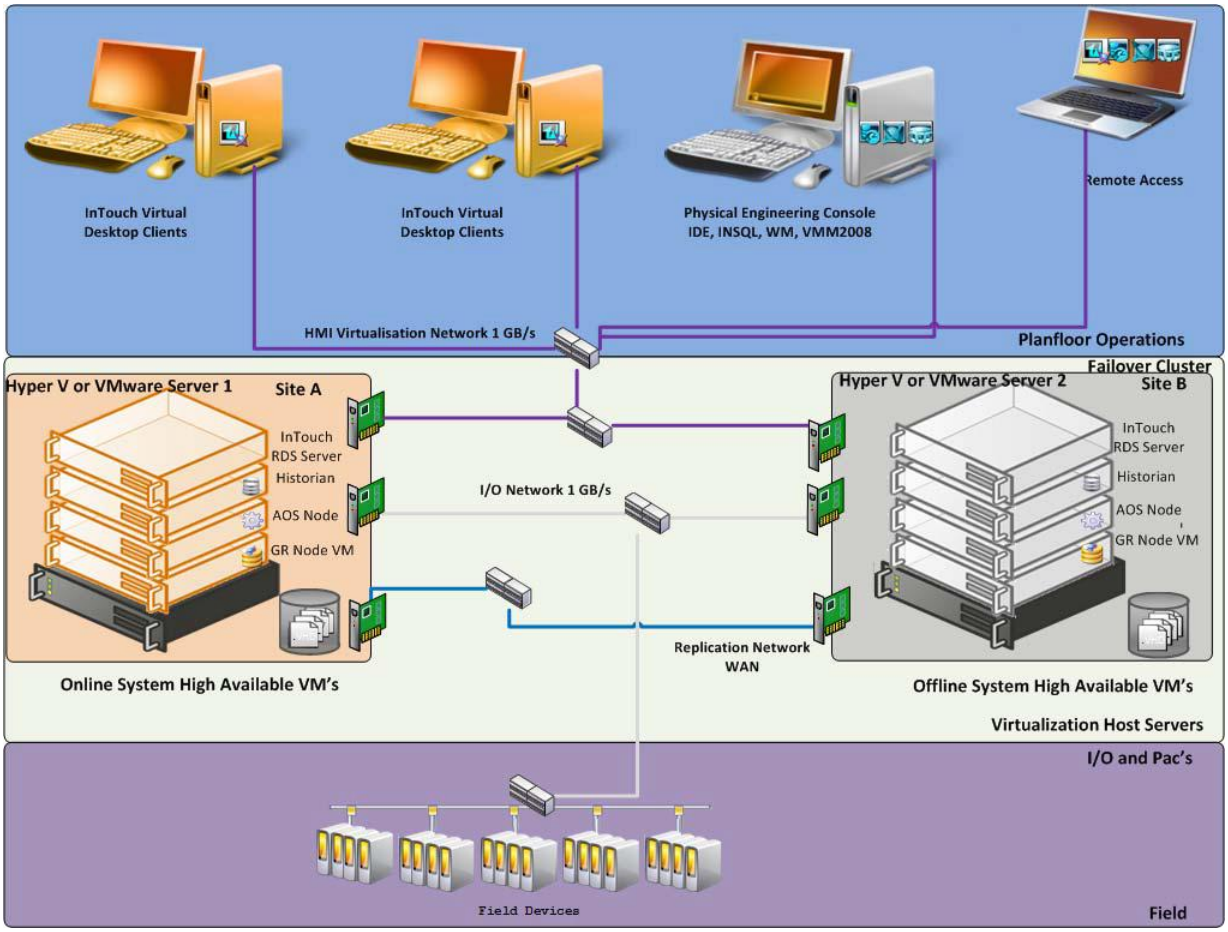
- Configuring minimum bandwidth

  The minimum network bandwidth is 100MB/sec. Recovery times improve with higher network speeds.

- Installing and configuring third-party software with Hyper-V virtualization

  Third party software from SIOS (SteelEye) mirrors the drives from site A to site B. The replication can be done on a SAN system or as shown in the illustration, with regular local hard drives.

Important: Mirrored partitions must have identical drive letters and sizes.



Defining High Availability and Disaster Recovery Combined

An important advantage from implementing HA and DR in the same scenario is that a local HA set can quickly resume functionality upon failure. In the event that site A is offline, the system can resume at site B without intervention from site A.

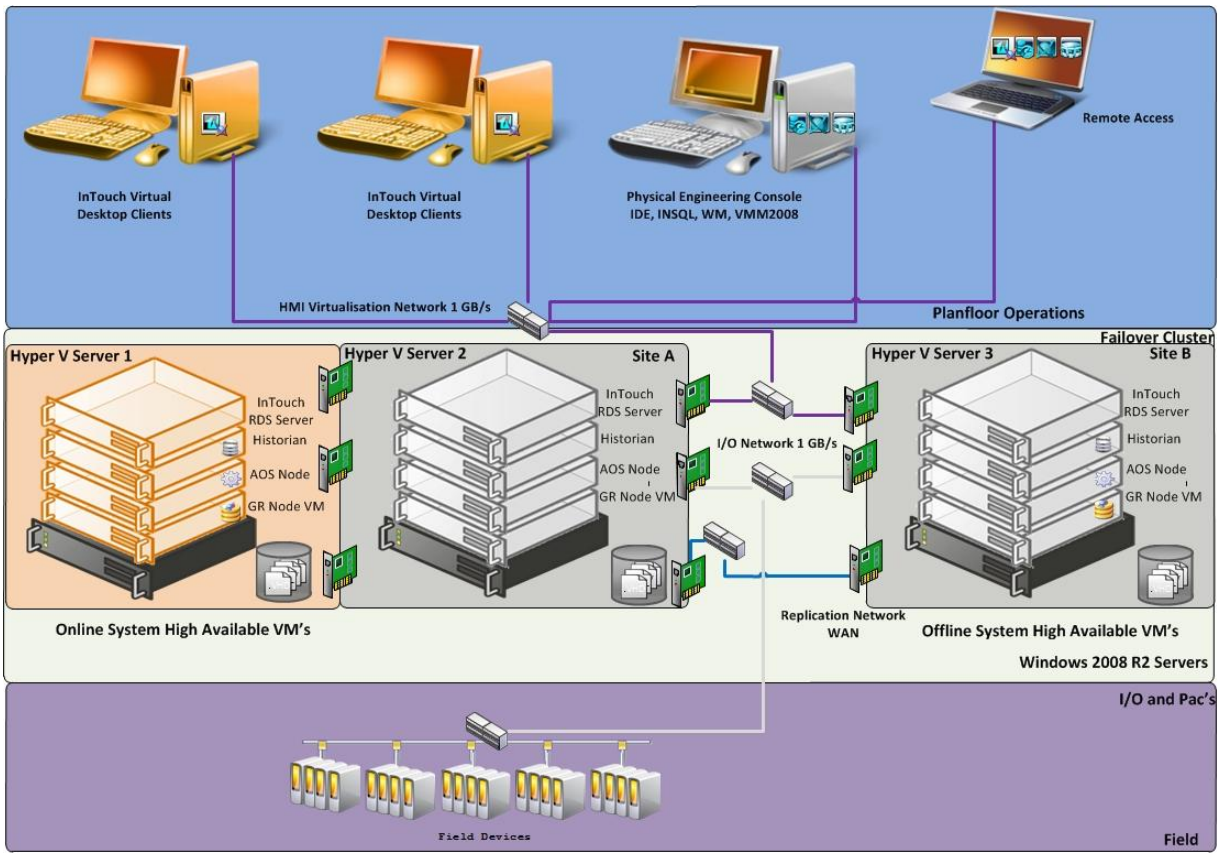
To define a HADR implementation, you need to plan for the following requirements:

- Sizing

  You'll need to triple the size of the estimated baseline server.

- SANs

  Two SANs are required—one local and one remote—to host the storage. In HADR implementation, the local configuration uses the failover cluster configuration and the set of VMs are replicated to a remote site.



Recommendations and Best Practices

The following recommendations and best practices apply to all for High Availability (HA), Disaster Recovery (DR), and HADR implementations, with guidelines specific to System Platform products.

- Ensure that auto log on is set up for all virtual machines running the System Platform products. This is to ensure that these virtual machines start automatically after the failover.
- Ensure the time on all the host servers, the virtual machines, and all other nodes which are part of the High Availability Environment are continuously synchronized. Otherwise, the virtual machines running on the host experience time drifts and results in discarding of data.You can add the time synchronization utility in the Start Up programs so that this utility starts automatically whenever the machine reboots.
- On the host servers disable all the network cards which are not utilized by the System Platform Environment. This is to avoid any confusion during the network selections while setting up the cluster.
- Ensure the Virtual Networks have the same name across all the nodes which are participating in the Cluster. Otherwise, migration/failover of virtual machines will fail.

System Platform Product-specific Recommendations and Observations

- During the preparation for Live and Quick migrations it is observed that the network freezes intermittently and then at the time of actual migration connectivity to the VM is lost. As a result, the System Platform node under migration experiences intermittent data loss during the preparation for Live and Quick migrations, and then has a data gap for the duration of actual migration.

The Historian

- In case of Live and Quick migration of the Historian, you may notice that the Historian logs values with quality detail 448 and there may be values logged twice with same timestamps. This is because the suspended Historian VM starts on the other cluster node with the system time it was suspended at before the migration. As a result, some of the data points it is receiving with the current time seem to be in the future to the Historian. This results in the Historian modifying the timestamps to its system time and updating the QD to 448. This happens until the system time of the Historian node catches up with the real current time using the TimeSync utility, after which the problem goes away. So, it is recommended to stop the historian before the migration and restart it after the VM is migrated and its system time is synced up with the current time.
- Live and Quick migration of the Historian should not be done when the block change over is in progress on the Historian node.
- If a failover happens (for example, due to a network disconnect on the source Host Virtualization Server) while the Historian status is still "Starting", the Historian node fails over to the target Host Virtualization Server. In the target host, the Historian fails to start. To recover from this state, kill the Historian services that failed to start and then start the Historian by launching the SMC.

InTouch HMI

- Ensure that InTouch Window Viewer is added to the Start Up programs so that the view is started automatically when the virtual machine reboots.

Application Server

- If a failover happens (for example, due to a network disconnect on the source Host Virtualization Server) while the Galaxy Migration is in progress, the GR node fails over to the target Host Virtualization Server. In the target host, on opening the IDE for the galaxy, the templates do not appear in the Template toolbox and in Graphic toolbox. To recover from this state, delete the Galaxy and create a new Galaxy. Initiate the migration process once again.
- If a failover happens (for example, due to an abrupt power-off on the source Host Virtualization Server) while a platform deploy is in progress, the Platform node fails over to the target Host Virtualization Server. In the target host, some objects will be in deployed state and the rest will be in undeployed state. To recover from this state, redeploy the whole Platform once again.
- If a failover happens (for example, due to an abrupt power-off on the source Host Virtualization Server) while a platform undeploy is in progress, the Platform node fails over to the target Host Virtualization Server. In the target host, some objects will be in undeployed state and the rest will be in deployed state. To recover from this state, undeploy the whole Platform once again.

Operations Integration Server

In case of Live and Quick migration of an I/O Server node (for example, DASSIDirect), InTouch I/O Tags acquiring data from that I/O server need to be reinitialized after the I/O server node is migrated. To automatically acquire the data for these tags from the I/O server after migration, it is recommended to have an InTouch script which monitors the quality status of any of those tags and triggers reinitialize I/O once the quality goes to bad. Execute this script every 3 to 5 seconds until the tag quality becomes good.

Additional Guidelines for DR and HADR Implementations (only)

The following guidelines apply to DR and HADR implementation only, and are in addition to all of the guidelines and recommendations listed under *"Recommendations and Best Practices"* and *System Platform Product-specific Recommendations and Observations* on page 37.

- As per the topology described earlier for the Disaster Recovery environment, only one network is used for all communications. If multiple networks are being used, then make sure only the primary network which is used for the communication between the Nodes is enabled for the Failover Cluster Communication. Disable the remaining cluster networks in Failover Cluster Manager.

Best Practices for SIOSIQ Mirroring

- While creating the SIOS IQ mirroring job, ensure the drive letters of the source and target drives to be mirrored are the same.
- We suggest that you have zero latency in the network when SIOS IQ mirroring, failover/migration of virtual machines between host servers take place. In the case of networks with latency, refer to the SIOS documentation on network requirements.
- While designing the network architecture, particularly with regard to bandwidth between the hosts in the Disaster Recovery environment, make sure to select the bandwidth based on the rate of data change captured from Disk Write Bytes/Sec on the host server for all the mirrored volumes. To verify that you have sufficient network bandwidth to successfully replicate your volume, use the Windows Performance Monitoring and Alerts tool to collect Write Bytes/sec on the replicated volumes to calculate the rate of data change. Collect this counter every 10 seconds and use your own data analysis program to estimate your rate of data change. For more details, refer to SIOS documentation on network requirements.

SIOS IQ can handle the following approximate average rates of change:

Network Bandwidth	**Rate of Change**
1.5 Mbps(T1)	182,000 Bytes/sec (1.45 Mbps)
10 Mbps	1,175,000 Bytes/sec (9.4 Mbps)
45 Mbps (T3)	5,250,000 Bytes/sec (41.75 Mbps)
100 Mbps	12,000,000 Bytes/sec (96 Mbps)
1000 Mbps (Gigabit)	65,000,000 Bytes/sec (520 Mbps)

The following table lists the impact on CPU utilization and bandwidth with various compression levels.

- Medium Configuration Load: Approx. 50000 IO Points with Approx. 20000 attributes being historized

- Network: Bandwidth controller with bandwidth: 45Mbps and No Latency

These readings are when the mirroring is continuously happening between the source and destination storage SANs when all the VM are running on the source host server. The data captured shows that the % CPU utilization of the SIOS mirroring process increases with increasing compression levels. Based on these findings we recommend Compression Level 2 in the Medium scale virtualization environment.

	Impact on CPU of Source Host Server		**Impact on Bandwidth**
	% Processor Time (ExtMirrSvc) - SIOS Mirroring process	**% Processor Time (CPU) - Overall CPU**	**Total Bytes / Sec**
Compression 0	Min: 0 Max:4.679 Avg: 0.157	Min: 0 Max:28.333 Avg: 1.882	Min: 0 Max: 11,042,788 Avg: 2,686,598
Compression 1	Min: 0 Max: 4.680 Avg: 0.254	Min: 0 Max: 31.900 Avg: 1.895	Min: 0 Max: 10,157,373 Avg: 1,871,426
Compression 2	Min: 0 Max:6.239 Avg: 0.402	Min: 0 Max:37.861 Avg: 2.622	Min: 791.970 Max: 10,327,221 Avg: 1,199,242
Compression 9	Min: 0 Max:13.525 Avg: 0.308	Min: 0 Max:42.094 Avg: 3.244	Min: 0 Max: 7,066,439 Avg: 649,822

Additional Guidelines for HADR Implementations (only)

The following guidelines apply to HADR implementation only, and are in addition to all of the guidelines and recommendations listed under *"Recommendations and Best Practices"*, including:

- *System Platform Product-specific Recommendations and Observations* on page 37
- *"Best Practices for SIOSIQ Mirroring"*
- *"Additional Guidelines for DR and HADR Implementations (only)"*

Though this is a three-node failover topology, to achieve the required failover order, a fourth node is required for setting up the Node Majority in the failover cluster. The three nodes are used for virtual machine services and the fourth node is used for Quorum witness. The fourth node is not meant for failover of virtual machines running on the cluster. This fourth node should not be marked as the preferred owner while setting up the preferred owners for the virtual machines running on the cluster.

The following scenario is a description of the failover order. Node 1 and Node 2 are in High Available site and Node 3 is in Disaster site. The failover sequence is Node 1 > Node 2 > Node 3.

- When all VMs are running on Node 1:
  - All three nodes are up. Now Node 1 goes down. The VMs running on Node 1 move to Node 2.
  - Node 1 and Node 3 are up and Node 2 is down. Now Node 1 goes down. The VMs running on Node 1 move to Node 3.
- When all VMs are running on Node 2:
  - Node 2 and Node 3 are up and Node 1 is down. Now Node 2 goes down. The VMs running on Node 2 move to Node 3.
  - All three nodes are up. Now Node 2 goes down. The VMs running on Node 2 move to Node 3.

CHAPTER 3

Implementing High Availability Using Hyper-V

This section introduces virtualization high-availability solutions that improve the availability of System Platform Products. A high-availability solution masks the effects of a hardware or software failure, and maintains the availability of applications so that the perceived downtime for users is minimized.

The set-up and configuration procedures, expected Recovery Time Objective (RTO) observations, Recovery Point Objective (RPO) observations, and data trend snapshots are presented first for small-scale virtualization environment, and are then repeated for medium-scale virtualization environment.

In This Chapter

Small Scale Virtualization Environments

This section contains the following topics:

- *Set Up Small Scale Virtualization Environment*
- *Configuration of System Platform Products in a Typical Small Scale Virtualization*
- *Expected Recovery Time Objective and Recovery Point Objective*
- *Medium Scale Virtualization Environments*

Set Up Small Scale Virtualization Environment

The following procedures help you to set up and implement a small scale virtualization environment.

Note: In the event that the private network becomes disabled, you may need to add a script to enable a failover. For more information, see "Add Script to Force Failover of the Virtual Machine if the Domain/ Private Network is disabled"

Plan for Small Scale Virtualization Environment

The following table lists the minimum and recommended hardware and software requirements for the machines used for a small scale virtualization environment:

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

Hyper-V Hosts

Processor:	Two - 2.66 GHz Intel Xeon with - 8 Cores

Operating System	Windows Server 2008 R2 or higher Enterprise with Hyper-V Enabled
Memory	12GB
Storage	Local Volume with Capacity of 500 GB

Note: For the Hyper-V Host to function optimally, the server should have the same processor, RAM, storage and service pack level. Preferably the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the above Specified Hyper-V Host, three virtual machines can be created with the following Configuration.

Virtual Machine 1: DAS SI, Historian, and Application Server (GR) node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian, ArchestrA, DAS SI

Virtual Machine 2: Application Server Runtime node 1

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	2 GB
Storage	40 GB
System Platform Products Installed	Application Server Runtime only, and InTouch

Virtual Machine 3: Information Server node, InTouch, and Historian Client

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 or higher Standard
Memory	4 GB
Storage	40 GB
System Platform Products Installed	Information Server, InTouch, Historian Client

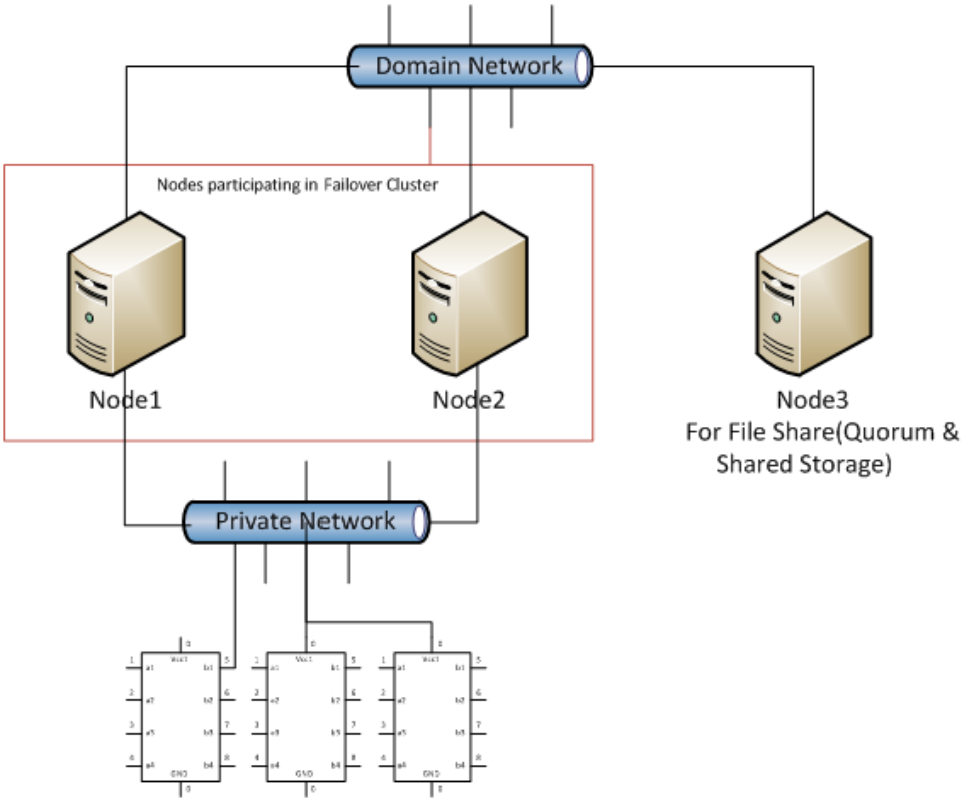
Note: There should be a minimum of two Hyper-V hosts to configure the failover cluster.

Network Requirements

For this high availability architecture, you can use two physical network cards that need to be installed on a host computer and configured, to separate the domain network and the process network.

Configure Failover Cluster

The following is the recommended topology of the failover cluster for a small scale virtualization high availability environment.



This setup requires a minimum of two host servers and one storage server shared across two hosts. Another independent node is used for configuring the quorum. For more information on configuring the quorum, refer to "Configure Cluster Quorum Settings".

The workflow for installing and configuring a failover cluster with two nodes is outlined in the following section. This workflow is applicable to setting up a small scale virtualization high availability environment.

Install Failover Cluster

To install the failover cluster feature, you need to run Windows Server 2008 R2 or higher Enterprise Edition on your server. Refer to Microsoft's server documentation for information about installing the failover cluster feature and step-by-step instructions.

Microsoft TechNet Library: Using Hyper-V and Failover Clustering
https://technet.microsoft.com/en-us/library/cc732181%28v=ws.10%29.aspx

Validate Failover Cluster Configuration

You must validate your configuration before you create a cluster. Validation helps you confirm the configuration of your servers, network, and storage meets the specific requirements for failover clusters. Refer to the Microsoft TechNet Library: Using Hyper-V and Failover Clustering for additional information

Create a Cluster

To create a cluster, you need to run the Create Cluster wizard. Refer to Microsoft Windows Server TechNet for information about creating a cluster and step-by-step instructions.

Windows Server 2008:
https://technet.microsoft.com/en-us/library/cc731844%28v=ws.10%29.aspx

Windows Server 2012, 2012 R2:
https://technet.microsoft.com/en-us/library/dn505754.aspx

Disable the Plant Network for Cluster Communication

After creating the Failover cluster using two or more Network Cards enabled, Make sure only Primary Network card which is used for the Communication between the Hyper-V nodes is enabled for the Failover Communication. You must disable the remaining custer networks.

Configure Cluster Quorum Settings

After both nodes have been added to the cluster, and the cluster networking components have been configured, you must configure the failover cluster quorum. Refer to Microsoft Windows Server TechNet for information about configuring the cluster quorum.
https://technet.microsoft.com/en-us/library/cc731739.aspx

The file share to be used for the node and File Share Majority quorum must be created and secured before configuring the failover cluster quorum. If the file share has not been created or correctly secured, the following procedure to configure a cluster quorum will fail. The file share can be hosted on any computer running a Windows operating system.

To configure the cluster quorum, you need to perform the following procedures:

- Create and secure a file share for the node and file share majority quorum
- Use the failover cluster management tool to configure a node and file share majority quorum

After you configure the cluster quorum, you must validate the cluster. For more information, refer to http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx.

Configure Storage

For a smaller virtualization environment, storage is one of the central barriers to implementing a good virtualization strategy. But with Hyper-V, VM storage is kept on a Windows file system. Users can put VMs on any file system that a Hyper-V server can access. As a result, HA can be built into the virtualization platform and storage for the virtual machines. This configuration can accommodate a host failure by making storage accessible to all Hyper-V hosts so that any host can run VMs from the same path on the shared folder. The back-end part of this storage can be a local or storage area network, iSCSI or whatever is available to fit the implementation. For this architecture, the Shared Folder is used.

The following table lists the minimum storage recommendations to configure storage for each VM:

System	Processor
Historian and Application Server (GR node) Virtual Machine	80 GB
Application Engine (Runtime node) Virtual Machine	40 GB
InTouch and Information Server Virtual Machine	40 GB

The recommended total storage capacity for a high availability virtual environment should be minimum 1TB.

Configure Hyper-V

With Microsoft Hyper-V, you can create a virtual environment that improves server utilization. It enhances patching, provisioning, management, support tools, processes, and skills. Microsoft Hyper-V provides live migration, cluster shared volume support, expanded processor, and memory support for host systems. Refer to Microsoft Technet library for Hyper-V installation prerequisites and other considerations.

The pre-requisites to set up Hyper-V include:

- x64-based processor
- Hardware-assisted virtualization
- Hardware Data Execution Prevention (DEP)

Configure Virtual Machines

After installing Hyper-V, you need to create a virtual machine. For more information, refer to *https://technet.microsoft.com/en-us/library/cc772480.aspx*
https://technet.microsoft.com/en-us/library/cc772480.aspx

Add Script to Force Failover of the Virtual Machine if the Domain/ Private Network is disabled

Whenever public network is disconnected on the node where the virtual machines are running, Failover Cluster Manager forces failover of all the Virtual Machine Services and applications to the other host node in the cluster. If the private network which is not participating in the cluster communication fails, Failover Cluster Manager does not failover any Cluster Service or Application.

To overcome this, we need to add a script which detects the private network failure as a dependency to the Virtual Machine. This results in failover of the Virtual Machine when the script fails.

Follow the process mentioned in the following URL to add the script:
http://gallery.technet.microsoft.com/ScriptCenter/5f7b4df3-af02-47bf-b275-154e5edf17e6/
http://gallery.technet.microsoft.com/ScriptCenter/5f7b4df3-af02-47bf-b275-154e5edf17e6/

The recommended setting for maximum failures in a period is 15, and the period should be set to 1 hour.

Configuration of System Platform Products in a Typical Small Scale Virtualization

To record the expected Recovery Time Objective (RTO) and Recovery Point Objective (RPO) trends and various observations in a small scale virtualization environment, tests are performed with System Platform Product configuration shown below.

The virtualization host server used for small scale configuration consists of three virtual machines listed below.

Node 1: GR, Historian and DAS SI Direct - Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 2 (AppEngine): Bootstrap, IDE and InTouch (Managed App) - Windows 2008 R2 Standard edition (64bit) OS

Node 3: Information Server, Bootstrap and IDE, InTouch Terminal Service and Historian Client - Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

Virtual Node	IO tags (Approx.)	Historized tags (Approx.)
GR	10000	2500
AppEngine	10000	5000

Expected Recovery Time Objective and Recovery Point Objective

This section provides the indicative Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for the load of IO and Attributes historized shown above and with the configuration of Host Virtualization Servers and Hyper-V virtual machines explained in the Setup instructions of Medium Scale Virtualization. In addition to these factors, the exact RTO and RPO depend on factors like storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

RTO and RPO Observations—HA Small Configuration

Important: The following sample data are provided only as guidelines for establishing testing specific to your needs.

Scenarios and observations in this section:

Scenario	Observation
Scenario 1: IT provides maintenance on Virtualization Server	"Live Migration"
	"Quick Migration of all nodes simultaneously"
	"Shut down"
Scenario 2: Virtualization Server hardware fails	"Scenario 2: Virtualization Server hardware fails"
cenario 3: Network fails on Virtualization Server	"Failover due to network disconnect (private)"

Scenario	Observation
Scenario 4: Virtualization Server becomes unresponsive	"Scenario 4: Virtualization Server becomes unresponsive"

The following tables display RTO and RPO observations with approximately 20000 IO points with approximately 7500 attributes being historized:

Scenario 1: IT provides maintenance on Virtualization Server

Live Migration

Primary Node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	Application Server	2 sec	Application Server tag (Script)	8 sec
			Application Server IO Tag (DASSiDirect)	13 sec
	Historian Client	2 sec	Historian Local tag	0 sec
			InTouch Tag $Second	4 sec
			Application Server IO Tag (DASSiDirect)	20 sec
			Application Server tag (Script)	0 sec
	OI Server	5 sec	N/A	N/A
Information Server Node	InTouch	5 sec		5 sec
	Information Server	5 sec	N/A	N/A
	Historian Client	5 sec	N/A	N/A
AppEngine	AppEngine	1 sec	Application IO tag (DASSiDirect)	3 sec

			Application Server tag (Script)	6 sec

Quick Migration

Primary node	**Products**	**RTO**	**RPO**	
			Tags	**Data Loss Duration**
GR	Application Server	134 sec	Application ServerTag (Script)	183 sec
			Application Server IO Tag (DASSiDirect)	184 sec
	Historian Client	145 sec	Historian Local tag	148 sec
			InTouch tag $Second	152 sec
			Application Server IO Tag (DASSiDirect)	165 sec
			IAS tag (Script)	0 sec
	OI Server	146 sec	N/A	N/A
Information Server Node	InTouch HMI	79 sec		89 sec
	Information Server	79 sec	N/A	N/A
	Historian Client	79 sec	N/A	N/A
AppEngine	AppEngine	59 sec	Application Server IO tag (DASSiDirect)	105 sec
			Application Server Tag (Script)	104 sec

Quick Migration of all nodes simultaneously

Primary node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	IAS	188 sec	Application Server tag (Script)	222 sec
			IAS IO tag (DASSiDirect)	227 sec
	Historian Client	220 sec	Historian Local tag	221 sec
			InTouch tag $Second	228 sec
			Application Server IO tag (DASSiDirect)	238 sec
			Application Server tag (Script)	135 sec
	OI Server	221 sec	N/A	
Information Server Node	InTouch HMI	183 sec		228 sec
	Information Server	183 sec	N/A	N/A
	Historian Client	183 sec	N/A	N/A
AppEngine	AppEngine	100 sec	Application Server IO tag (DASSiDirect)	238 sec
			Application Server tag (Script)	135 sec

Shut down

Primary node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	Application Server	160 sec	Application Server tag (Script)	3 min 36 sec

			Application Server IO tag (DASSiDirect)	3 min 43 sec
	Historian Client	211 sec	Historian Local tag	3 min 25 sec
			InTouch tag $Second	3 min 32 sec
			Application Server IO tag (DASSiDirect)	3 min 50 sec
			Application Server tag (Script)	2 min 46 sec
	OI Server	212 sec	N/A	N/A
Information Server Node	InTouch HMI	202 sec		212 sec
	Information Server	202 sec	N/A	N/A
	Historian Client	202 sec	N/A	N/A
AppEngine	AppEngine	114 sec	Application Server IO tag (DASSiDirect)	3 min 50 sec
			Application Server tag (Script)	2 min 46 sec

Scenario 2: Virtualization Server hardware fails

The failover occurs due to hardware failure, and it is simulated with power-off on the host server.

Primary node	**Products**	**RTO**	**RPO**	
			Tags	**Data Loss Duration**
GR	Application Server	497 sec	Application Server Tag (Script)	9min

			Application Server IO tag (DASSiDirect)	9 min
	Historian Client	532 sec	Historian local tag	9 min 23 sec
			InTouch tag $Second	10 min + time taken to start viewer
			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
			Application Server IO tag (DASSiDirect)	8 min 23 sec
			Application Server tag (Script)	7 min 1 sec
	OI Server	269 sec	N/A	N/A
Information Server Node	InTouch HMI	601 sec + time taken by the user to start the InTouchView		611 sec
			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
	Information Server	601 sec + time taken by the user to start the Information Server	N/A	N/A
	Historian Client	601 sec+ time taken by the user to start the Hist Client	N/A	N/A
AppEngine	AppEngine	366 sec	Application Server IO Tag (DASSiDirect)	8 min 23 sec

			Application Server tag (Script)	7 min 1 sec

Scenario 3: Network fails on Virtualization Server

The failover occurs due to network disconnect (public). In this case, the VMs restart, after moving to the other host server.

Primary node	Products	RTO	RPO	
			Tags	**Data Loss Duration**
GR	Application Server	535 sec	Application Server Tag (Script)	9 min 8 sec
			IAS IO Tag (DASSiDirect)	8 min 53 sec
	Historian Client	544 sec	Historian Local Tag	9 min 35 sec
			InTouch Tag $Second	9 min 16 sec
			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
			Application Server IO Tag (DASSiDirect)	8 min 57 sec
			Application Server Tag (Script)	7 min 52 sec
	OI Server	457sec	N/A	N/A
Information Server	InTouch HMI	415 sec + time taken by the user to start the InTouchView	N/A	556 sec + Time taken to run viewer)

			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
	Information Server	415 sec + time taken by the user to start the Information Server	N/A	N/A
	Historian Client	415 sec + time taken by the user to start the Hist Client	N/A	N/A
AppEngine	AppEngine	463 sec	N/A	8 min 57 sec
			N/A	7 min 52 sec

Failover due to network disconnect (private)

In this case, the private network disconnects on GR, VM will be moved to the other host server.

Primary node	**Products**	**RTO**	**RPO**	
			Tags	**Data Loss Duration**
GR	Application Server	118 sec	Application Server Tag (Script)	132 sec
			Application Server IO Tag (DASSiDirect)	140 sec
	Historian Client	128 sec	Historian Local Tag	132 sec
			InTouch Tag $Second	147 sec
			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	

			Application Server IO Tag (DASSiDirect)	145 sec
			IAS Tag (Script)	0 (Sfed)
	OI Server	134 sec		
Information Server Node	InTouch HMI	N/A	N/A	
	Information Server	N/A	N/A	
	Historian Client	N/A	N/A	N/A
AppEngine	AppEngine	N/A	Application Server IO Tag (DASSiDirect)	
			Application Server Tag (Script)	

Scenario 4: Virtualization Server becomes unresponsive

There is no failover of VMs to the other host server when the CPU utilization on the host server is 100%.

Primary node	**Products**	**RTO (sec)**	**RPO**	
			Tags	**Data Loss Duration**
GR	Application Server	N/A	N/A	
	Historian Client	N/A	N/A	
Information Server Node	InTouch HMI	N/A	N/A	
	Information Server	N/A	N/A	

	Historian Client	N/A	N/A	N/A
AppEngine	AppEngine	N/A	N/A	
	InTouch HMI	N/A	N/A	
Information Server Node	InTouch HMI	N/A	N/A	

Medium Scale Virtualization Environments

This section contains the following topics:

- *Set Up Medium Scale Virtualization Environment*
- *Configuration of System Platform Products in a Typical Medium Scale Virtualization*
- *Expected Recovery Time Objective and Recovery Point Objective*

Set Up Medium Scale Virtualization Environment

The following procedures help you to set up and implement the medium scale virtualization high availability environment.

Note: In the event that the private network becomes disabled, you may need to add a script to enable a failover. For more information, see "Add Script to Force Failover of the Virtual Machine if the Domain/ Private Network is disabled"

Plan for Medium Scale Virtualization Environment

The minimum recommended hardware and software requirements for the Host and Virtual machines used for medium virtualization environment are provided in the table below:

Hyper-V Host

Processor	Two 2.79 GHz Intel Xeon with 24 Cores
Operating System	Windows Server 2008 R2 or higher Enterprise with Hyper-V enabled
Memory	48 GB
Storage	SAN with 1TB storage disk

Note: For the Hyper-V Host to function optimally, the server should have the same processor, RAM, storage and service pack level. Preferably the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the Hyper-V host specified above, seven virtual machines can be created in the environment with the configuration given below.

Virtual Machine 1: Historian node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	200 GB
System Platform Products Installed	Historian

Virtual Machine 2: Application Server node, DAS SI

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	100 GB
System Platform Products Installed	ArchestrA-Runtime, DAS SI

Virtual Machine 3: InTouch TS node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch with TS enabled

Virtual Machine 4: Application Server Runtime node 1

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard

Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only and InTouch

Virtual Machine 5: Application Server Runtime node 2

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only

Virtual Machine 6: Information Server node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Information Server

Virtual Machine 7: Historian Client node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows 7 or higher Enterprise
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian Client

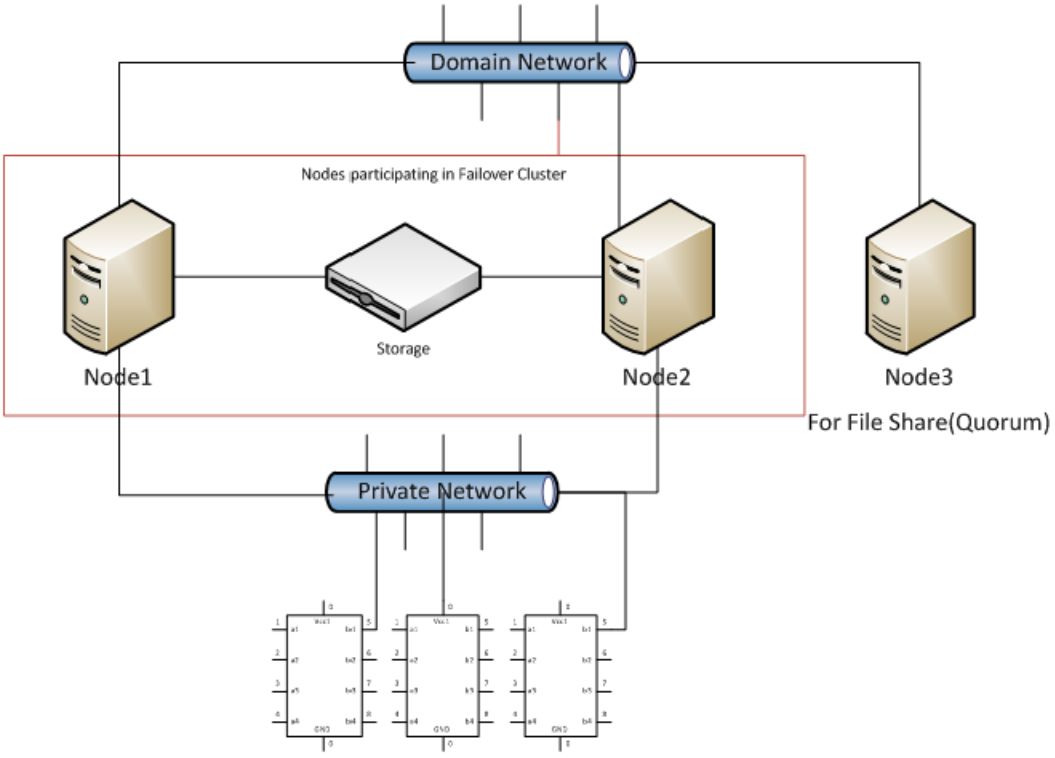
Note: There should be a minimum of two Hyper-V hosts to configure the failover cluster.

Network Requirements

For this high availability architecture, you can use two physical network cards that need to be installed on a host computer and configured to separate the domain network and the process network.

Configure Failover Cluster

The following is the recommended topology of the failover cluster for a medium scale virtualization high availability environment.



This setup requires a minimum of two host servers and one storage server shared across two hosts. Another independent node is used for configuring the quorum. For more information on configuring the quorum, refer to "Configure Cluster Quorum Settings".

The workflow for installing and configuring a failover cluster with two nodes is outlined in the following section. This workflow is applicable to setting up a medium scale virtualization high availability environment.

Install Failover Cluster

To install the failover cluster feature, you need to run Windows Server 2008 R2 or higher Enterprise Edition on your server. Refer to Microsoft's server documentation for information about installing the failover cluster feature and step-by-step instructions.

Microsoft TechNet Library: Using Hyper-V and Failover Clustering

https://technet.microsoft.com/en-us/library/cc732181%28v=ws.10%29.aspx
https://technet.microsoft.com/en-us/library/cc732181%28v=ws.10%29.aspx

Validate Failover Cluster Configuration

You must validate your configuration before you create a cluster. Validation helps you confirm the configuration of your servers, network, and storage meets the specific requirements for failover clusters.

Create a Cluster

To create a cluster, you need to run the Create Cluster wizard. Refer to Microsoft Windows Server TechNet for information about creating a cluster and step-by-step instructions.

Windows Server 2008:

https://technet.microsoft.com/en-us/library/cc731844%28v=ws.10%29.aspx
https://technet.microsoft.com/en-us/library/cc731844%28v=ws.10%29.aspx

Windows Server 2012, 2012 R2:

https://technet.microsoft.com/en-us/library/dn505754.aspx
https://technet.microsoft.com/en-us/library/dn505754.aspx

Disable the Plant Network for Cluster Communication

After creating the Failover cluster using two or more Network Cards enabled, Make sure only Primary Network card which is used for the Communication between the Hyper-V nodes is enabled for the Failover Communication Disable the remaining Cluster Networks

Configure Cluster Quorum Settings

Quorum is the number of elements that need to be online to enable continuous running of a cluster. In most instances, the elements are nodes. In some cases, the elements also consist of disk or file share witnesses. Each of these elements determines whether the cluster should continue to run. Refer to Microsoft Windows Server TechNet for information about configuring the cluster quorum.

https://technet.microsoft.com/en-us/library/cc731739.aspx
https://technet.microsoft.com/en-us/library/dn505754.aspx

All elements, except the file share witnesses, have a copy of the cluster configuration. The cluster service ensures that the copies are always synchronized. The cluster should stop running if there are multiple failures or if there is a communication error between the cluster nodes.

After both nodes have been added to the cluster, and the cluster networking components have been configured, you must configure the failover cluster quorum.

The file share to be used for the node and File Share Majority quorum must be created and secured before configuring the failover cluster quorum. If the file share has not been created or correctly secured, the following procedure to configure a cluster quorum will fail. The file share can be hosted on any computer running a Windows operating system.

To configure the cluster quorum, you need to perform the following procedures:

- Create and secure a file share for the node and file share majority quorum
- Use the failover cluster management tool to configure a node and file share majority quorum

After you configure the cluster quorum, you must validate the cluster. For more information, refer to *http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx*
http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx.

Configure Storage

For any virtualization environment, storage is one of the central barriers to implementing a good virtualization strategy. But with Hyper-V, VM storage is kept on a Windows file system. Users can put VMs on any file system that a Hyper-V server can access. As a result, you can build HA into the virtualization platform and storage for the virtual machines. This configuration can accommodate a host failure by making storage accessible to all Hyper-V hosts so that any host can run VMs from the same path on the shared folder. The back-end part of this storage can be a local storage area network, iSCSI or whatever is available to fit the implementation.

The following table lists the minimum storage recommendations for each VM:

System	Processor
Historian Virtual Machine	200 GB
Application Server (GR node) Virtual Machine	100 GB
Application Engine 1(Runtime node) Virtual Machine	80 GB
Application Engine 2 (Runtime node) Virtual Machine	80 GB
InTouch Virtual Machine	80 GB
Information Server Virtual Machine	80 GB
Historian Client	80 GB

The recommended total storage capacity for a high availability virtual environment should be minimum 1TB.

Configure Hyper-V

With Microsoft Hyper-V, you can create a virtual environment that improves server utilization. It enhances patching, provisioning, management, support tools, processes, and skills. Microsoft Hyper-V provides live migration, cluster shared volume support, expanded processor, and memory support for host systems. Refer to Microsoft Technet library for Hyper-V installation prerequisites and other considerations.

The pre-requisites to set up Hyper-V include:

- x64-based processor
- Hardware-assisted virtualization
- Hardware Data Execution Prevention (DEP)

Configure Virtual Machines

After installing Hyper-V, you need to create a virtual machine. For more information, refer to *https://technet.microsoft.com/en-us/library/cc772480.aspx*
https://technet.microsoft.com/en-us/library/cc772480.aspx

Add Script to Force Failover of the Virtual Machine if the Domain/ Private Network is disabled

Whenever public network is disconnected on the node where the virtual machines are running, Failover Cluster Manager forces failover of all the Virtual Machine Services and applications to the other host node in the cluster. If the private network which is not participating in the cluster communication fails, Failover Cluster Manager does not failover any Cluster Service or Application.

To overcome this, we need to add a script which detects the private network failure as a dependency to the Virtual Machine. This results in failover of the Virtual Machine when the script fails.

Follow the process mentioned in the following URL to add the script:
http://gallery.technet.microsoft.com/ScriptCenter/5f7b4df3-af02-47bf-b275-154e5edf17e6/
http://gallery.technet.microsoft.com/ScriptCenter/5f7b4df3-af02-47bf-b275-154e5edf17e6/

Configuration of System Platform Products in a Typical Medium Scale Virtualization

To record the expected Recovery Time Objective (RPO) and Recovery Point Objective (RPO), trends and various observations in a medium scale virtualization environment, tests are performed with System Platform Product configuration shown below.

The virtualization host server used for medium scale configuration consists of seven virtual machines listed below.

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

Node 1 (GR): GR, InTouch and DAS SI Direct – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 2 (AppEngine1): Bootstrap, IDE and InTouch (Managed App) – Windows 2008 R2 Standard edition (64bit) OS

Node 3 (AppEngine2): Bootstrap, IDE – Windows 2008 R2 Standard edition (64bit) OS

Node 4: Historian – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 5: Information Server, Bootstrap and IDE – Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007

Node 6: InTouch Terminal Service – Windows 2008 R2 Standard edition (64bit) OS enabled with Terminal Service

Node 7: Historian Client and InTouch – Windows 7 Professional Edition (64bit) OS with SQL Server 2008 SP1 32 bit

Virtual Node	IO tags (Approx.)	Historized tags (Approx.)
AppEngine1	25000	10000
AppEngine2	25000	10000

Expected Recovery Time Objective and Recovery Point Objective

This section provides the indicative Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for the load of IO and Attributes historized shown above and with the configuration of Host Virtualization Servers and Hyper-V virtual machines explained in the Setup instructions of Medium Scale Virtualization. In addition to these factors, the exact RTO and RPO depend on factors like storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

RTO and RPO Observations—HA Medium Configuration

Important: The following sample data are provided only as guidelines for establishing testing specific to your needs.

Scenarios and observations in this section:

Scenario	Observation
Scenario 1: IT provides maintenance on Virtualization Server	"Scenario 1: IT provides maintenance on Virtualization Server"
	"Quick Migration"
	"Quick Migration of all nodes simultaneously"
Scenario 2: Virtualization Server hardware fails	"Scenario 2: Virtualization Server hardware fails"
Scenario 3: Network fails on Virtualization Server	"Scenario 3: Network fails on Virtualization Server"
Scenario 4: Virtualization Server becomes unresponsive	"Scenario 4: Virtualization Server becomes unresponsive"

The following tables display RTO and RPO observations with approximately 50000 IO points with approximately 20000 attributes being historized:

Scenario 1: IT provides maintenance on Virtualization Server

Live Migration

Products	RTO	RPO	
		Tags	Data Loss Duration
InTouch HMI	13 sec	Data Loss for $Second tag (Imported to Historian)	13 sec

GR	10 sec	Application Server Tag (Script)	12 sec
		Application Server IO Tag (DASSiDirect)	59 sec
AppEngine1	15 sec	Application Tag (Script)	22 sec
		Application Server IO Tag (DASSiDirect)	57 sec
AppEngine2	7 sec	Application Server Tag (Script)	11 sec
		Application Server IO Tag (DASSiDirect)	57 sec
Historian Client	9 sec	SysTimeSec (Historian)	0 sec
		$Second (InTouch)	2 sec
		Application Server Tag (Script)	0 (Data is SFed)
		Application Server IO Tag (DASSiDirect)	0 (Data is SFed)
OI Server SIDirect	14 sec	N/A	N/A
Historian Client	0 sec	N/A	N/A
Information Server	5 sec	N/A	N/A

Quick Migration

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	31 sec	Data Loss for $Second tag (Imported to Historian)	27 sec
GR	50 sec	IAS Tag (Script)	50 sec
		Application Server IO Tag (DASSiDirect)	1 Min 51 Sec
AppEngine1	35 sec	Application Server Tag (Script)	35 sec

		Application Server IO Tag (DASSiDirect)	54 sec
AppEngine2	41 sec	Application Server Tag (Script)	44 sec
		Application Server IO Tag (DASSiDirect)	1 Min 14 Sec
Historian Client	84 sec	SysTimeSec (Historian)	1 Min 25 Sec
		$Second (InTouch)	1 Min 51 Sec
		Application Server Tag (Script)	0 (data is SFed)
		Application Server IO Tag (DASSiDirect)	0 (data is SFed)
OI Server SIDirect	50 sec	N/A	N/A
Historian Client	1 Min 32 Sec	N/A	N/A
Information Server	33 sec	N/A	N/A

Quick Migration of all nodes simultaneously

The following table displays the data for Quick Migration of all nodes.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	28 Sec	Data Loss for $Second tag (Imported to Historian)	1 Min 40 Sec
GR	04 Sec	Application Server Tag (Script)	1 Min 36 Sec
		Application Server IO Tag (DASSiDirect)	4 Min 14 Sec
AppEngine1	67 Sec	Application Server Tag (Script)	1 Min 20 Sec
		Application Server IO Tag (DASSiDirect)	4 Min 11 Sec
AppEngine2	54 Sec	Application Server Tag (Script)	52 Sec

		Application Server IO Tag (DASSiDirect)	4 Min 28 Sec
Historian Client	73 Sec	SysTimeSec (Historian)	1 Min 14 Sec
		$Second (InTouch)	1 Min 40 Sec
		Application ServerTag (Script)	1 Min 36 Sec
		Application Server IO Tag (DASSiDirect)	4 Min 14 Sec
OI Server SIDirect	107 Sec	N/A	
Historian Client	38 Sec	N/A	
Information Server	36 Sec	N/A	

Scenario 2: Virtualization Server hardware fails

The Virtualization Server hardware failure results in failover that is simulated with power-off on the host server. In this case, the VMs restart, after moving to the other host server.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	335 Sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	6 Min 47 Sec.
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
GR	313 Sec	Application Server Tag (Script)	5 Min 44 Sec
		Application Server IO Tag (DASSiDirect)	7 Min 28 Sec
AppEngine1	365 Sec	Application Server Tag (Script)	6 Min 35 Sec
		Application Server IO Tag (DASSiDirect)	7 Min 29 Sec

AppEngine2	372 Sec	Application Server Tag (Script)	6 Min 41 Sec
		Application Server IO Tag (DASSiDirect)	7 Min 20 Sec
Historian Client	381 Sec	SysTimeSec (Historian)	6 Min 33 Sec
		$Second (InTouch)	6 Min 47 Sec
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
		Application Server Tag (Script)	5 Min 45 Sec
		Application Server IO Tag (DASSiDirect)	7 Min 30 Sec
DAS SIDirect	265 Sec	N/A	N/A
Historian Client	214 Sec + time taken by the user to start the Historian Client	N/A	N/A
Information Server	255 Sec + time taken by the user to start the Information Server	N/A	N/A

Scenario 3: Network fails on Virtualization Server

Failover due to Network Disconnect (Public)

In this case, after the VMs move to the other host server, the VMs restart.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	150 sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	4 Min 14 Sec

		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
GR	197 sec	Application Server Tag (Script)	3 Min 41 Sec
		Application Server IO Tag (DASSiDirect)	3 Min 50 Sec

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
AppEngine1	188 sec	Application Server Tag (Script)	3 Min 31 Sec
		Application Server IO Tag (DASSiDirect)	4 Min 2 Sec
AppEngine2	200 sec	Application Server Tag (Script)	3 Min 41 Sec
		Application Server IO Tag (DASSiDirect)	4 Min 08 Sec
Historian Client	236 sec	SysTimeSec (Historian)	3 Min 55 Sec
		$Second (InTouch)	4 Min 14 Sec
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
		Application Server Tag (Script)	3 Min 41 Sec
		Application Server IO Tag (DASSiDirect)	3 Min 50 Sec
OI Server SIDirect	174 sec	N/A	N/A
Historian Client	163 sec + time taken by the user to start the Historian Client	N/A	N/A

Information Server	66 sec + time taken by the user to start the Information Server	N/A	N/A

Failover due to network disconnect (plant)

In this case, only the GR Node moves to other host server and restarts. Only GR has data acquisition through Plant network and disconnected Plant network results in failover of GR alone.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	N/A	Data Loss for $Second tag (Imported to Historian)	N/A
GR	97 Sec	IAS Tag (Script)	1 Min 43 Sec
		Application Server IO Tag (DASSiDirect)	1 Min 46 Sec
AppEngine1	N/A	Application Server Tag (Script)	N/A
		Application Server IO Tag (DASSiDirect)	1 Min 50 Sec
AppEngine2	N/A	Application Server Tag (Script)	N/A
		Application Server IO Tag (DASSiDirect)	1 Min 58 Sec
Historian Client	N/A	SysTimeSec (Historian)	N/A
		$Second (InTouch)	N/A
		Application Server Tag (Script)	1 Min 43 Sec
		Application Server IO Tag (DASSiDirect)	1 Min 46 Sec
OI Server SIDirect	111 Sec	N/A	N/A
Historian Client	N/A	N/A	N/A
Information Server	N/A	N/A	N/A

Scenario 4: Virtualization Server becomes unresponsive

There is no failover of VMs to the other host server when the CPU utilization on the host server is 100%.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch HMI	N/A	N/A	N/A
GR	N/A	N/A	N/A
	N/A	N/A	N/A
AppEngine1	N/A	N/A	N/A
	N/A	N/A	N/A
AppEngine2	N/A	N/A	N/A
Historian Client	N/A	N/A	N/A
	N/A	N/A	N/A
	N/A	N/A	N/A
	N/A	N/A	N/A
OI Server SIDirect	N/A	N/A	N/A
Historian Client	N/A	N/A	N/A
Information Server	N/A	N/A	N/A

CHAPTER 4

Implementing High Availability Using vSphere

The following procedures are designed to help you set up and implement High Availability using VMware vSphere. These procedures assume that you have VMware ESXi™ 5.0 or above, vCenter Server™, and vSphere Client already installed.

For basic procedures to install these and other VMware products, see product support and user documentation at http://www.vmware.com/.

The High Availability vSphere implementation assumes that you are implementing a a medium-scale system.

This section contains the following topics:

- *Plan the Virtualization Environment*
- *Configuration of System Platform Products in a Typical Virtualization Environment*
- *Set up the Virtualization Environment*
- *Expected Recovery Time Objective and Recovery Point Objective*

In This Chapter

Plan the Virtualization Environment

The minimum recommended hardware and software requirements for the Host and Virtual machines used for virtualization environment are provided in the following table:

ESXi Host

Processor	Two 2.79 GHz Intel Xeon with 8 cores (Hyper-threaded)
Operating System	ESXi 5.0 or higher
Memory	48 GB
Storage	SAN with 1TB storage disk

Note: For the ESXi Host to function optimally, the server should have the same processor, RAM, storage and service pack level. Preferably the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the ESXi host specified above, seven virtual machines can be created in the environment with the configuration given below.

Virtual Machine 1: Historian node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	4 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	200 GB
System Platform Products Installed	Historian

Virtual Machine 2: Application Server node, DAS SI

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	4 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	100 GB
System Platform Products Installed	ArchestrA-Runtime, DAS SI

Virtual Machine 3: InTouch TS node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch with TS enabled

Virtual Machine 4: Application Server Runtime node 1

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only and InTouch

Virtual Machine 5: Application Server Runtime node 2

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only

Virtual Machine 6: Information Server node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	1 vCPU
Operating System	Windows Server 2008 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Information Server

Virtual Machine 7: Historian Client node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	1 vCPUs

Operating System	Windows 7 or higher Enterprise
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian Client

Note: There should be a minimum of two vSphere hosts to configure the failover cluster.

Network Requirements

For this high availability architecture, you can use two physical network cards that need to be installed on a host computer and configured to separate the domain network and the process network.

Configuration of System Platform Products in a Typical Virtualization Environment

To record the expected Recovery Time Objective (RPO) and Recovery Point Objective (RPO), trends and various observations in a virtualization environment, tests are performed with System Platform Product configuration shown below.

The virtualization host server used for configuration consists of seven virtual machines listed below.

Node 1 (GR): GR, InTouch and DAS SI Direct – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 2 (AppEngine1): Bootstrap, IDE and InTouch (Managed App) – Windows 2008 R2 Standard edition (64bit) OS

Node 3 (AppEngine2): Bootstrap, IDE – Windows 2008 R2 Standard edition (64bit) OS

Node 4: Historian – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 5: Information Server, Bootstrap and IDE – Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007

Node 6: InTouch Terminal Service – Windows 2008 R2 Standard edition (64bit) OS enabled with Terminal Service

Node 7: Historian Client and InTouch – Windows 7 Professional Edition (64bit) OS with SQL Server 2008 SP1 32 bit

Virtual Node	IO tags (Approx.)	Historized tags(Approx.)
AppEngine1	25000	10000
AppEngine2	25000	10000

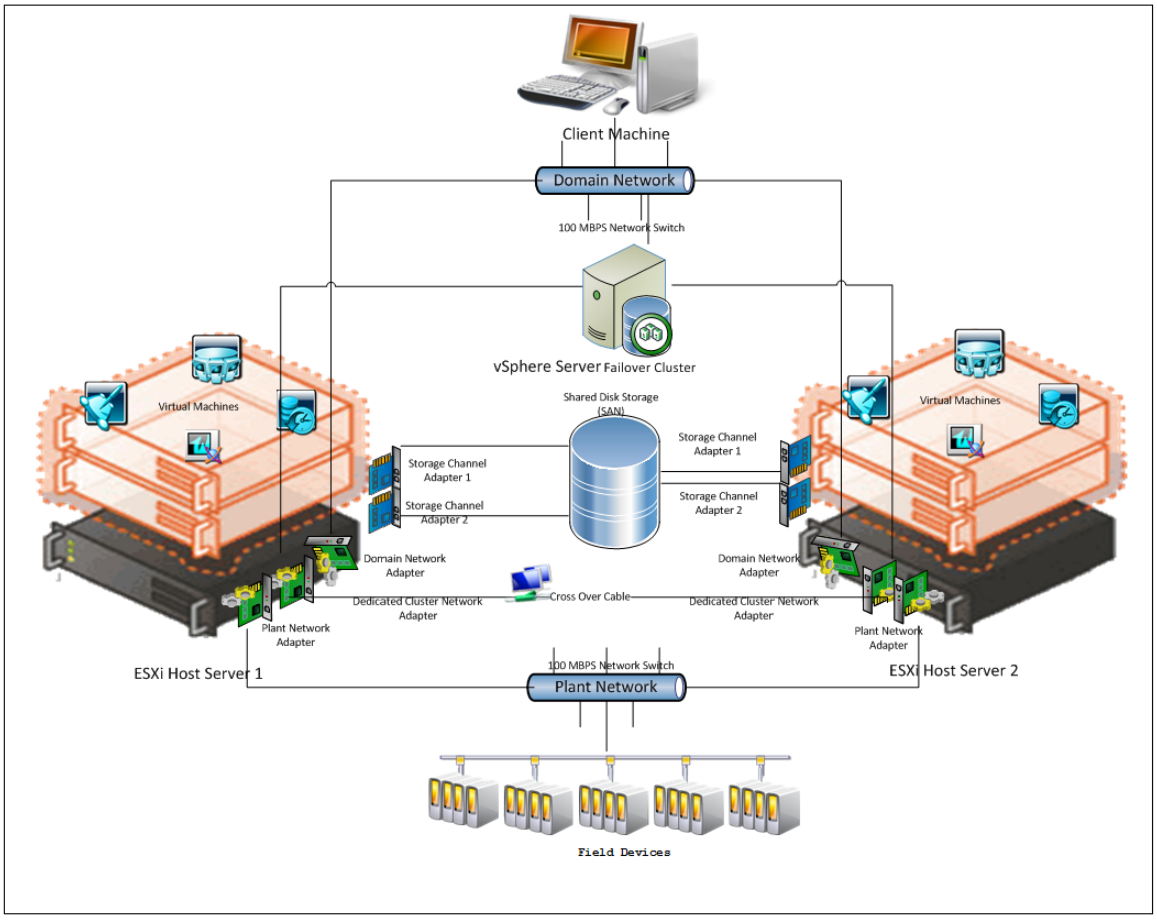
Set up the Virtualization Environment

The following procedures help you to set up and implement the high availability virtualization environment using vSphere technology.

Note: In the event that the private network becomes disabled, you may need to add a script to enable a failover.

Create a Datacenter

The vSphere Datacenter virtualizes an infrastructure that includes servers, storage, networks. It provides for end-to-end connectivity between client machines and field devices. The following is the recommended topology of the Datacenter, with a vSphere Failover Cluster, for a High Availability environment.



The following workflow outlines how to configure a Datacenter as a virtualized High Availability environment with a failover cluster consisting of two nodes. This setup requires a minimum of two host servers and one storage server shared across two hosts.

Create the Datacenter

Use the vSphere Client to create the DataCenter. Refer to your vSphere documentation for additional information.

Add hosts to the Datacenter

Refer to your vSphere documentation or the VMware knowledge base for additional information and add an ESXi host.

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896
http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896

Repeat this procedure to add another ESXi host.

Create a Failover Cluster

A cluster in vSphere is a group of hosts. Resources of a host added to a cluster, also known as a failover cluster, become part of the cluster's resources, and are managed by the cluster. In a vSphere High Availability environment, virtual machines automatically restart on a different physical server in a cluster if a host fails.

Refer to your vSphere documentation or the VMware knowledge base for information about adding the failover cluster.

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896
http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896

To create a failover cluster:

1. Select the new cluster option.
2. Select vSphere HA.
3. Use the New Cluster Wizard to complete configuration.

Configure Storage

VMware Virtual Machine File System (VMFS) datastores serve as repositories for virtual machines. You can set up VMFS data stores on any SCSI-based storage devices that the host discovers, including Fibre Channel, iSCSI, and local storage devices.

Use the following workflow to create a datastore. Your new datastore is added to all hosts if you use the vCenter Server system to manage your hosts.

Important: Install and configure any adapter that your storage requires before creating datastores. After you create a datastore, rescan the adapters to discover the new storage device.

Create a datastore

1. Log on to vSphere Client and select a host.
2. Configure storage for the host.
3. Configure the file system version.
4. Check the parameters you have selected and create the datastore.

Configure Networks

After you create a datacenter, use the following the workflow to configure one or more networks on the ESXi host.

Configure networks on the ESXi host

1. Log on to vSphere Client and select a host

2. Add networking through the Add Network Wizard.
3. Click **Add Networking.** The **Add Network Wizard** appears.
4. Use the Wizard to complete network configuration.

Create a Virtual Machine in vSphere Client

You can populate your virtualization environment by creating virtual machines, which are the key components in a virtual infrastructure.

When you create a virtual machine, you associate it with a particular datacenter, host, cluster or resource pool, and a datastore. The virtual machine consumes resources dynamically as the workload increases, or it returns resources dynamically as the workload decreases.

Every virtual machine has virtual devices that provide the same function as the physical hardware. A virtual machine derives the following attributes from the host with which it is associated:

- A CPU and memory space
- Access to storage
- Network connectivity

User vSphere Client to create a Virtual Machine and configure its properties.

Enable vMotion for Migration

VMware vMotion enables migration of a running virtual machine from one server to another, including the VM's associated storage, network identity, and network connections. Access to the VM's storage switches to the new physical host. Access to the VM continues with its same virtualized network identity.

Following are typical migration scenarios:

- Removing VMs from underperforming or problematic servers
- Performing hardware maintenance and upgrades
- Optimizing VMs within resource pools

Use vSphere Client to enable vMotion for migration.

Expected Recovery Time Objective and Recovery Point Objective

This section provides the expected Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for a load of 50,000 I/O and approximately 20,000 historized Attributes by virtualization servers and vSphere VMs set up for High Availability as described in this chapter. The exact RTO and RPO depend on factors such as storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

Scenarios and observations in this section:

Scenario	Observation
Scenario 1: IT provides maintenance on virtualization server using VMware	"An example of a graceful host server shutdown would be when IT provides maintenance on a virtualization server using VMware"
Scenario 2: Virtualization server hardware fails while using VMware	"Scenario 2: Virtualization server hardware fails while using VMware"
Scenario 3: Network fails on Virtualization server that uses VMware	(A): "Scenario 3: Network fails on Virtualization server that uses VMware"
Scenario 4: Migration of vSphere High Availability Medium Configuration	"Scenario 4: Migration of VMs Using VMware vMotion"

Scenario 1: Graceful shutdown of the host server

An example of a graceful host server shutdown would be when IT provides maintenance on a virtualization server using VMware

Products	RTO (sec)	RPO	
		Tags	Data Loss Duration
AppEngine1	89	P28762.I15	96
AppEngine2	89	P30443.I1	96
Application Server	77	Integer_001.PV.I1	102

Observations:

1. Shut down the slave host machines for the VMs to move to the master host node.
2. The above readings were taken with the WIS node machine is on the master node.
3. The VMs are rebooted while they migrate from the slave host machine to the master machine.

Scenario 2: Virtualization server hardware fails while using VMware

Products	RTO (sec)	RPO	
		Tags	Data Loss Duration
AppEngine1	135	P28762.I15	96
AppEngine2	134	P30443.I1	96
Application Server	125	Integer_001.PV.I1	102

Observations

1. Remove the power cable of the slave host machine so that the VMs can move to the master host node.
2. The above readings were taken when the WIS node machine is on the master node and the remaining VMs are on the slave node.
3. You need not have a VM in the master node to migrate VMs while the slave power cables are removed, as in the case of the Slave Shutdown scenario.
4. The VMs are rebooted while they migrate from the slave host machine to the master machine.

Scenario 3: Network fails on Virtualization server that uses VMware

Products	RTO (sec)	RPO	
		Tags	Data Loss Duration
AppEngine1	185	P28762.I15	120 sec
AppEngine2	190	P30443.I1	120 sec
Application Server	210	Integer_001.PV.I1	150 sec
DAServer	190	N/A	190
Historian	255	SysTimeSec	200 sec
InTouch HMI	190	$Second	210 sec

Observations

1. Remove the domain Network cable of the slave host machines so that the VMs can move to the master host node.

2. You need not have a virtual machine in the master node to migrate VMs, while the slave Domain Network cable is removed as in the case of the Slave Shutdown scenario.
3. The above readings were taken when the WIS node machine was on the master node.
4. The VMs get rebooted while they migrate from the slave host machine to the master machine.

Scenario 4: Migration of VMs Using VMware vMotion

The following table displays the data loss duration, when the VMs are migrated individually from one host to another using vMotion.

Products	**RTO (sec)**	**RPO**	
		Tags	**Data Loss Duration**
Application Server	1	Integer_001.PV	0.02 sec
AppEngine1	0	P28762.I15	0 sec
AppEngine2	1	P30443.I1	0.01 sec
InTouch HMI	0	$Second	0 sec
Historian	0	SysTimeSec	0 sec
Information Server	0	N/A	N/A
DAServer	0	N/A	N/A
Historian Client	0	N/A	N/A

Observations

1. Migrate the VMs individually from one host to another host.
2. The VMs will migrate from one host to another host without being rebooted.

CHAPTER 5

Implementing Disaster Recovery Using Hyper-V

This section introduces several Disaster Recovery (DR) virtualization solutions that improve the availability of System Platform Products. For more information refer to Chapter 1 Getting Started with High Availability and Disaster Recovery.

The set-up and configuration procedures, expected Recovery Time Objective (RTO) observations, Recovery Point Objective (RPO) observations, and data trend snapshots are presented first for small-scale virtualization environment, and are then repeated for medium-scale virtualization environment.

In This Chapter

Small Scale Virtualization Environments

This chapter contains the following topics:

- *Set Up Small Scale Virtualization Environment*
- *Configuration of System Platform Products in a Typical Small Scale Virtualization*
- *Expected Recovery Time Objective and Recovery Point Objective*
- *Medium Scale Virtualization Environments*

Set Up Small Scale Virtualization Environment

The following procedures help you to set up small scale virtualization disaster recovery environment.

Plan for Disaster Recovery

The minimum and recommended hardware and software requirements for the Host and Virtual machines used for small scale virtualization disaster recovery environment.

Hyper-V Hosts	
Processor	Two 2.66 GHz Intel Xeon with 8 Cores
Operating System	Windows Server 2008 R2 or higher Enterprise with Hyper-V enabled
Memory	12 GB
Storage	Local Volume with Capacity 500 GB

Note: For the Hyper-V Host to function optimally, the server should have the same processor, RAM, storage and service pack level. Preferably the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the above specified Hyper-V host, three virtual machines can be created with below configuration.

Virtual Machine 1: DAS SI, Historian, and Application Server (GR) Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian, ArchestrA, DAS SI

Virtual Machine 2: Application Server Runtime Node 1

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	2 GB
Storage	40 GB
System Platform Products Installed	Application Server Runtime only and InTouch

Virtual Machine 3: Information Server Node, InTouch, Historian Client

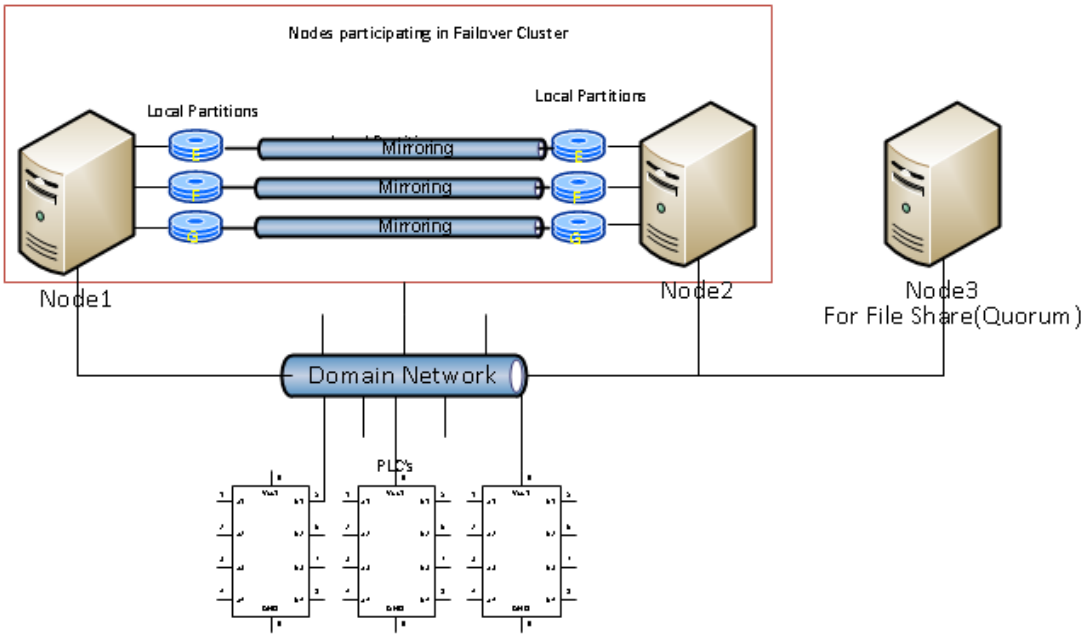
Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	40 GB
System Platform Products Installed	Information Server, InTouch, Historian Client

Network Requirements

For this architecture, you can use one physical network card that needs to be installed on a host computer for the domain network and the process network.

Configure Failover Cluster

The recommended topology of the failover cluster for disaster recovery process for small scale virtualization environment is given below:



This setup requires a minimum of two host servers with sufficient local disk space on each server to create logical drives for the virtual machines. Each logical drive is replicated to the two hosts for disaster recovery. Another independent node is used for configuring the quorum. For more information on configuring the quorum, refer to "Configure Cluster Quorum Settings".

The workflow for setting up the small virtualization disaster recovery environment is outlined in the following section.

Install Failover Cluster

To install the failover cluster feature, you need to run Windows Server 2008 R2 or higher Enterprise Edition on your server. Refer to Microsoft's server documentation for information about installing the failover cluster feature and step-by-step instructions.

Microsoft TechNet Library: Using Hyper-V and Failover Clustering
https://technet.microsoft.com/en-us/library/cc732181%28v=ws.10%29.aspx

Validate Cluster Configuration

Before creating a cluster, you must validate your configuration. Validation helps you to confirm the configuration of your servers, network, and to storage meets the specific requirements for failover clusters. Refer to the Microsoft TechNet Library: Using Hyper-V and Failover Clustering for additional information.

Create a Cluster

To create a cluster, run the Create Cluster wizard. Refer to Microsoft Windows Server TechNet for information about creating a cluster and step-by-step instructions.

Windows Server 2008:
https://technet.microsoft.com/en-us/library/cc731844%28v=ws.10%29.aspx

Windows Server 2012, 2012 R2:
https://technet.microsoft.com/en-us/library/dn505754.aspx

Configure Cluster Quorum Settings

Quorum is the number of elements that need to be online to enable continuous running of a cluster. In most instances, the elements are nodes. In some cases, the elements also consist of disk or file share witnesses. Each of these elements determines whether the cluster should continue to run.

All elements, except the file share witnesses, have a copy of the cluster configuration. The cluster service ensures that the copies are always synchronized. The cluster should stop running if there are multiple failures or if there is a communication error between the cluster nodes.

After both nodes have been added to the cluster, and the cluster networking components have been configured, you must configure the failover cluster quorum.

The file share to be used for the node and File Share Majority quorum must be created and secured before configuring the failover cluster quorum. If the file share has not been created or correctly secured, the following procedure to configure a cluster quorum will fail. The file share can be hosted on any computer running a Windows operating system.

To configure the cluster quorum, you need to perform the following procedures:

- Create and secure a file share for the node and file share majority quorum
- Use the failover cluster management tool to configure a node and file share majority quorum

Refer to Microsoft Windows Server TechNet for information about configuring the cluster quorum.
https://technet.microsoft.com/en-us/library/cc731739.aspx

Validate the cluster quorum after you have configured it. For more information, refer to http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx.

Configure Storage

For a smaller virtualization environment, storage is one of the central considerations in implementing a good virtualization strategy. But with Hyper-V, VM storage is kept on a Windows file system. You can put VMs on any file system that a Hyper-V server can access. As a result, HA can be built into the virtualization platform and storage for the virtual machines. This configuration can accommodate a host failure by making storage accessible to all Hyper-V hosts so that any host can run VMs from the same path on the shared folder. The back-end part of this storage can be a local, storage area network, iSCSI, or whatever is available to fit the implementation.

For this architecture, local partitions are used.

The following table lists the minimum storage recommendations to configure storage for each VM:

System	Storage Capacity
Historian and Application Server (GR node) Virtual Machine	80 GB

System	Storage Capacity
Application Engine (Runtime node) Virtual Machine	40 GB
InTouch and Information Server Virtual Machine	40 GB

The total storage capacity should be minimum recommended 1 TB.

Configure Hyper-V

Microsoft® Hyper-V™ helps in creating a virtual environment that improves server utilization. It enhances patching, provisioning, management, support tools, processes, and skills. Microsoft Hyper-V provides live migration, cluster shared volume support, expanded processor, and memory support for host systems.

Hyper-V is available in x64-based versions of Windows Server 2008 R2 operating system and higher, specifically the x64-based versions of Windows Server 2008 R2 and higher Standard, Windows Server 2008 R2 and higher Enterprise, and Windows Server 2008 and higher Datacenter.

The following are the pre-requisites to set up Hyper-V:

- x64-based processor
- Hardware-assisted virtualization
- Hardware Data Execution Prevention (DEP)

Configure SIOS (SteelEye) DataKeeper and Hyper-V Replica

SIOS (SteelEye) DataKeeper and Hyper-V Replica are replication software for real-time Windows data. Both can be used to replicate all data types, including the following:

- Open files
- SQL and Exchange Server databases
- Hyper-V .vhd files

Hyper-V Replica is a built-in replication mechanism that was introduced in the Windows Server 2012 Hyper-V Role.

The ability of both SteelEye DataKeeper and Hyper-V Replica to replicate live Hyper-V virtual machines ensures that a duplicate copy is available in case the primary storage array fails. This helps in disaster recovery (DR) without impacting production.

SteelEye DataKeeper Cluster Edition is a host-based replication solution, which extends Microsoft Windows Server Failover Clustering (WSFC) and Microsoft Cluster Server (MSCS) features such as cross-subnet failover and tunable heartbeat parameters. These features make it possible to deploy geographically distributed clusters.

You can replicate a virtual machine across LAN, WAN, or any Windows server through SIOS Microsoft Management Console (MMC) interface. You can run the DataKeeper MMC snap-in from any server. The DataKeeper MMC snap-in interface is similar to the existing Microsoft Management tools.

Note: For information on installing the SteelEye DataKeeper, refer to http://www.sios.com. Ensure that the local security policies, firewall, and port settings are configured as per the details provided in the SteelEye DataKeeper documents. For information on using Hyper-V Replica, refer to the Hyper-V Replica Overview at https://technet.microsoft.com/en-us/library/jj134172.aspx

The following sections outline the workflow for using SteelEye DataKeeper.

Configure Virtual Machines

Create a SteelEye mirroring job and then create a virtual machine in the disk.

After creating the virtual machine, you need to add the dependency between the virtual machine and the datakeeper volume in the cluster. This dependency triggers the switching of the source and target Servers of the SteelEye DataKeeper Volume resource when failover of the virtual machines occurs in the Failover Cluster Manager.

You can create multiple virtual machines.

Configuration of System Platform Products in a Typical Small Scale Virtualization

To record the expected Recovery Time Objective (RTO) and Recovery Point Objective (RPO), trends and various observations in a small scale virtualization environment, tests are performed with System Platform Product configuration shown below.

The virtualization host server used for small scale configuration consists of three virtual machines listed below.

Node 1: GR, Historian and DAS SI Direct – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit

Node 2 (AppEngine): Bootstrap, IDE and InTouch (Managed App) – Windows 2008 R2 Standard edition (64bit) OS

Node 3: Information Server, Bootstrap and IDE, InTouch Terminal Service and Historian Client – Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007

Virtual Node	IO tags (Approx.)	Historized tags (Approx.)
GR	10000	2500
AppEngine	10000	5000

Expected Recovery Time Objective and Recovery Point Objective

This section provides the indicative Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for the load of IO and Attributes historized shown above and with the configuration of Host Virtualization Servers and Hyper-V virtual machines explained in the Setup instructions of Medium Scale Virtualization. For more information refer to *"Set Up Medium Scale Virtualization Environment"*. In addition to these factors, the exact RTO and RPO depend on factors like storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

RTO and RPO Observations - DR Small Configuration

Important: The following sample data are provided only as guidelines for establishing testing specific to your needs.

Scenarios and observations in this section:

Scenario	Observation
Scenario 1: IT provides maintenance on Virtualization Server	"Live Migration"
	"Quick Migration"
	"Quick Migration of All Nodes Simultaneously"
Scenario 2: Virtualization Server hardware fails	"Scenario 2: Virtualization Server hardware fails"
Scenario 3: Network fails on Virtualization Server	"Scenario 3: Network fails on Virtualization server"
Scenario 4: Virtualization Server becomes unresponsive	"Scenario 4: Virtualization Server becomes unresponsive"

The following tables display RTO and RPO Observations with approximately 20000 IO points with approximately 7500 attributes being historized:

Scenario 1: IT provides maintenance on Virtualization Server

Live Migration

Primary Node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	IAS	14 sec	IAS tag (Script)	20 sec
			IAS IO tag (DASSiDirect)	26 sec
	Historian Client	19 sec	Historian Local tag	22 sec
			InTouch Tag $Second	27 sec
			IAS IO Tag (DASSiDirect)	32 sec

			IAS tag (Script)	0 (data is SFed)
	DAServer	21 sec	N/A	N/A
WIS	InTouch HMI	12 sec	$Second	12 sec
	Information Server	12 sec	N/A	N/A
	Historian Client	12 sec	N/A	N/A
AppEngine	AppEngine	12 sec	IAS IO tag (DASSiDirect)	26 sec
			IAS tag Script)	13 sec
	InTouch HMI	12 sec	$Second	12 sec

Quick Migration

Node Name	**Products**	**RTO**	**RPO**	
			Tags	**Data Loss Duration**
GR	IAS	147 sec	IAS tag (Script)	160 sec
			IAS IO Tag (DASSiDirect)	167 sec
	Historian Client	156 sec	Historian Local tag	164 sec
			InTouch tag $Second	171 sec
			IAS IO Tag (DASSiDirect)	170 sec
			IAS tag (Script)	0 (data is SFed)
	DAServer	156 sec	N/A	N/A
WIS	InTouch HMI	91 sec	$Second	91 sec
	Information Server	91 sec	N/A	N/A
	Historian Client	91 sec	N/A	N/A
AppEngine	AppEngine	59 sec	IAS IO tag (DASSiDirect)	80 sec

			IAS Tag (Script)	73 sec
	InTouch HMI	68 sec	$Second	68 sec

Quick Migration of All Nodes Simultaneously

Primary node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	IAS	221 sec	IAS tag (Script)	229 sec
			IAS IO tag (DASSiDirect)	234 sec
	Historian Client	225 sec	Historian Local tag	226 sec
			InTouch tag $Second	238 sec
			IAS IO tag (DASSiDirect)	242 sec
			IAS tag (Script)	160 sec
	DAServer	225 sec	N/A	
WIS	InTouch HMI	225 sec	$Second	255 sec
	Information Server	225 sec	N/AS	
	Historian Client	225 sec	N/A	
AppEngine	AppEngine	150 sec	IAS IO tag (DASSiDirect)	242 sec
			IAS tag (Script)	160 sec
	InTouch HMI	149 sec	$Second	149 sec

Scenario 2: Virtualization Server hardware fails

The Virtualization Server hardware failure results in failover that is simulated with power-off on the host server. In this case, the VMs restart, after moving to the other host server.

Primary node	Products	RTO	RPO	
			Tags	**Data Loss Duration**
GR	IAS	270 sec	IAS tag (Script)	5 Min 22 sec
			IAS IO tag (DASSiDirect)	5 Min 12 sec
	Historian Client	362 sec	Historian Local tag	6 Min 40 sec
			InTouch tag $Second	6 Min 58 sec **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of he Historian node, which historizes this tag.
			IAS IO tag (DASSiDirect)	5 Min 16 sec
			IAS tag (Script)	4 Min 55 sec
	DAServer	196 sec	N/A	N/A

Primary Node	Products	RTO	RPO	
			Tags	**Data Loss Duration**

WIS	InTouch HMI	240 sec + time taken by the user to start the InTouchView	$Second	6 Min 58 sec **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.
	Information Server	240 sec + time taken by the user to start the Information Server	N/A	N/A
	Historian Client	240 sec + time taken by the user to start the Historian Client	N/A	N/A
AppEngine	AppEngine	267 sec	IAS IO tag (DASSiDirect)	5 Min 16 sec
			IAS tag (Script)	4 Min 55 sec
	InTouch HMI	267 sec + time taken by the user to start the ITView	$Second	267 sec + time taken by the user to start the ITView
			Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	

Scenario 3: Network fails on Virtualization server

The failure of network on the Virtualization Server results in failover due to network disconnect (Public). Bandwidth used is 45Mbps and there is no latency. In this case, the VMs restart, after moving to the other host server.

Primary Node	Products	RTO	RPO	
			Tags	Data Loss Duration
GR	IAS	251 sec	IAS tag (Script)	4 Min 42 sec
			IAS IO tag (DASSiDirect)	4 Min 47 sec
	Historian Client	290 sec	Historian local tag	5 Min 11 sec
			InTouch tag $Second	5 Min 10 sec **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.
			IAS IO tag (DASSiDirect)	4 Min 42 sec
			IAS tag (Script)	3 Min 58 sec
	DAServer	191 sec	N/A	N/A

Primary Node	Products	RTO	RPO	
			Tags	Data Loss Duration
WIS	InTouch HMI	215 sec + time taken by the user to start the InTouchView	$Second	5 Min 10 sec
			Note: RPO is dependent on time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node which historizes this tag	

	Information Server	215 sec + time taken by the user to start the Information Server	N/A	N/A
	Historian Client	215 sec + time taken by the user to start the Historian Client	N/A	N/A
AppEngine	AppEngine	209 sec	IAS IO Tag (DASSiDirect)	4 Min 42 sec
			IAS tag (Script)	3 Min 58 sec
	InTouch HMI	195 sec + time taken by the user to start the ITView	$Second	195 sec
			Note: RPO is dependent on time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node which historizes this tag.	

Scenario 4: Virtualization Server becomes unresponsive

There is no failover of VMs to the other host server when the CPU utilization on the host server is 100%.

Primary Node	**Products**	**RTO**	**RPO**	
			Tags	**Data Loss Duration**
GR	IAS	N/A	N/A	N/A
			N/A	N/A
	Historian Client	N/A	N/A	N/A
			N/A	N/A
			N/A	N/A
			N/A	N/A
	DAServer	N/A	N/A	N/A

WIS	InTouch HMI	N/A	N/A	N/A
	Information Server	N/A	N/A	N/A
	Historian Client	N/A	N/A	N/A
AppEngine	AppEngine	N/A	N/A	N/A
			N/A	N/A
	InTouch HMI	N/A	N/A	N/A

Medium Scale Virtualization Environments

This section contains the following topics:

- *Set Up Medium Scale Virtualization Environment*
- *Configure System Platform Products in a Typical Medium Scale Virtualization*
- *Expected Recovery Time Objective and Recovery Point Objective*

Set Up Medium Scale Virtualization Environment

The following procedures help you to set up small scale virtualization disaster recovery environment.

Plan for Disaster Recovery

The minimum and recommended hardware and software requirements for the Host and Virtual machines used for medium scale virtualization disaster recovery environment.

Hyper-V Hosts

Processor	Two 2.79 GHz Intel Xeon with 24 Cores
Operating System	Windows Server 2008 R2 or higher Enterprise with Hyper-V enabled
Memory	48 GB
Storage	SAN with 1TB storage disk

Note: For the Hyper-V Host to function optimally, the server should have the same processor, RAM, storage and service pack level. Preferably the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the Hyper-V host specified above, seven virtual machines can be created in the environment with the configuration given below.

Virtual Machine 1: Historian Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	200 GB
System Platform Products Installed	Historian

Virtual Machine 2: Application Server Node, DAS SI

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	100 GB
System Platform Products Installed	ArchestrA-Runtime, DAS SI

Virtual Machine 3: InTouch TS Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch with TS enabled

Virtual Machine 4: Application Server Runtime Node 1

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard

Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only and InTouch

Virtual Machine 5: Application Server Runtime Node 2

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only

Virtual Machine 6: Information Server Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Information Server

Virtual Machine 7: Historian Client Node

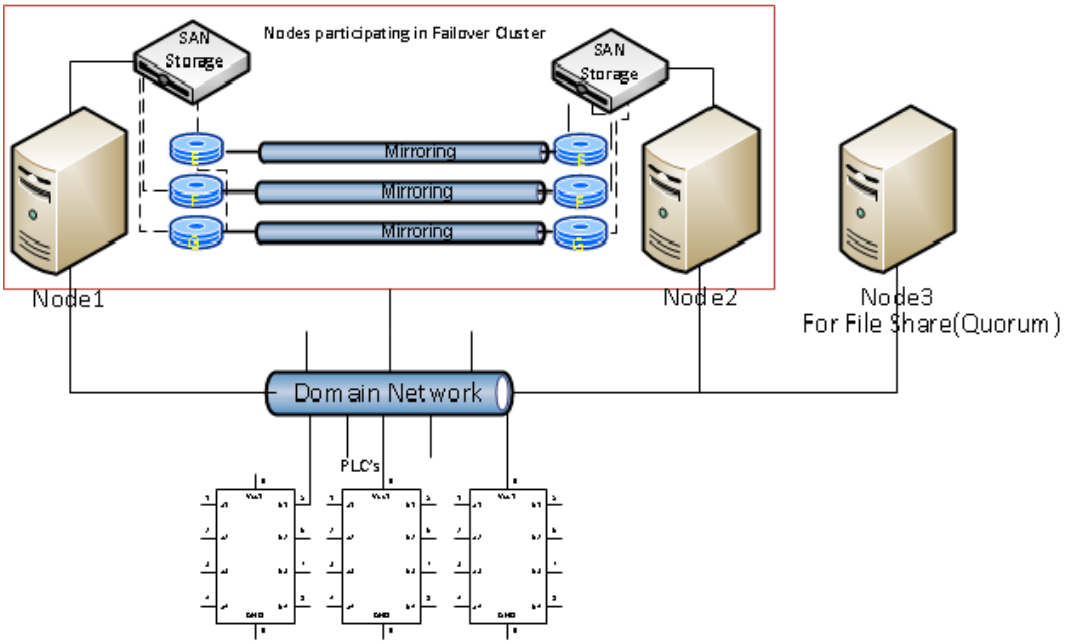
Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows 7 or higher Enterprise
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian Client

Network Requirements

For this architecture, you can use one physical network card that needs to be installed on a host computer for the domain network and the process network.

Configure Failover Cluster

The recommended topology of the failover cluster for disaster recovery process for medium scale virtualization environment is given below:



This setup requires a minimum of two host servers and two storage servers connected to each host independently. Another independent node is used for configuring the quorum. For more information on configuring the quorum, refer to "Configure Cluster Quorum Settings".

The workflow for installing and configuring a failover cluster with two nodes is outlined in the following section. This workflow is applicable to setting up a medium configuration.

Install Failover Cluster

To install the failover cluster feature, you need to run Windows Server 2008 R2 or higher Enterprise Edition on your server. Refer to Microsoft's server documentation for information about installing the failover cluster feature and step-by-step instructions.

Microsoft TechNet Library: Using Hyper-V and Failover Clustering
https://technet.microsoft.com/en-us/library/cc732181%28v=ws.10%29.aspx

Validate Cluster Configuration

You must validate your configuration before you create a cluster. Validation helps you to confirm the configuration of your servers, network, and to storage meets the specific requirements for failover clusters. Refer to the Microsoft TechNet Library: Using Hyper-V and Failover Clustering for additional information.

Create a Cluster

To create a cluster, run the Create Cluster wizard. Refer to Microsoft Windows Server TechNet for information about creating a cluster and step-by-step instructions.

Windows Server 2008:
https://technet.microsoft.com/en-us/library/cc731844%28v=ws.10%29.aspx

Windows Server 2012, 2012 R2:
https://technet.microsoft.com/en-us/library/dn505754.aspx

Configure Cluster Quorum Settings

Quorum is the number of elements that need to be online to enable continuous running of a cluster. In most instances, the elements are nodes. In some cases, the elements also consist of disk or file share witnesses. Each of these elements determines whether the cluster should continue to run.

All elements, except the file share witnesses, have a copy of the cluster configuration. The cluster service ensures that the copies are always synchronized. The cluster should stop running if there are multiple failures or if there is a communication error between the cluster nodes.

After both nodes have been added to the cluster, and the cluster networking components have been configured, you must configure the failover cluster quorum.

You must create and secure the file share that you want to use for the node and the file share majority quorum before configuring the failover cluster quorum. If the file share has not been created or correctly secured, the following procedure to configure a cluster quorum will fail. The file share can be hosted on any computer running a Windows operating system.

To configure the cluster quorum, you need to perform the following procedures:

- Create and secure a file share for the node and file share majority quorum
- Use the failover cluster management tool to configure a node and file share majority quorum

Create and secure a file share for the node and file share majority quorum

Create a new folder on the system that will host the share directory and allow sharing.

Then, use the failover cluster management tool to configure the node and file share majority quorum.

After you configure the cluster quorum, validate the cluster. For more information, refer to http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx.

Configure Storage

For any virtualization environment, storage is one of the central barriers to implementing a good virtualization strategy. But with Hyper-V, VM storage is kept on a Windows file system. Users can put VMs on any file system that a Hyper-V server can access. As a result, you can build HA into the virtualization platform and storage for the virtual machines. This configuration can accommodate a host failure by making storage accessible to all Hyper-V hosts so that any host can run VMs from the same path on the shared folder. The back-end part of this storage can be a local, storage area network, iSCSI or whatever is available to fit the implementation.

The following table lists the minimum storage recommendations for each VM:

System	Storage Capacity
Historian Virtual Machine	200 GB

System	Storage Capacity
Application Server (GR node) Virtual Machine	100 GB
Application Engine 1(Runtime node) Virtual Machine	80 GB
Application Engine 2 (Runtime node) Virtual Machine	80 GB
InTouch Virtual Machine	80 GB
Information Server Virtual Machine	80 GB
Historian Client	80 GB

The total storage capacity should be minimum recommended 1 TB.

Configure Hyper-V

With Microsoft® Hyper-V™, you can create a virtual environment that improves server utilization. It enhances patching, provisioning, management, support tools, processes, and skills. Microsoft Hyper-V provides live migration, cluster shared volume support, expanded processor, and memory support for host systems. Refer to Microsoft Technet library for Hyper-V installation prerequisites and other considerations.

The following are the pre-requisites to set up Hyper-V:

- x64-based processor
- Hardware-assisted virtualization
- Hardware Data Execution Prevention (DEP)

Configuring SIOS (SteelEye) DataKeeper and Hyper-V Replica

SIOS (SteelEye) DataKeeper and Hyper-V Replica are replication software for real-time Windows data. Both can be used to replicate all data types, including the following:

- Open files
- SQL and Exchange Server databases
- Hyper-V .vhd files

Hyper-V Replica is a built-in replication mechanism that was introduced in the Windows Server 2012 Hyper-V Role.

The ability of both SteelEye DataKeeper and Hyper-V Replica to replicate live Hyper-V virtual machines ensures that a duplicate copy is available in case the primary storage array fails. This helps in disaster recovery (DR) without impacting production.

SteelEye DataKeeper Cluster Edition is a host-based replication solution, which extends Microsoft Windows Server Failover Clustering (WSFC) and Microsoft Cluster Server (MSCS) features such as cross-subnet failover and tunable heartbeat parameters. These features make it possible to deploy geographically distributed clusters.

You can replicate a virtual machine across LAN, WAN, or any Windows server through SIOS Microsoft Management Console (MMC) interface. You can run the DataKeeper MMC snap-in from any server. The DataKeeper MMC snap-in interface is similar to the existing Microsoft Management tools.

Note: For information on installing the SteelEye DataKeeper, refer to http://www.steeleye.com. Ensure that the local security policies, firewall, and port settings are configured as per the details provided in the SteelEye DataKeeper documents. For information on using Hyper-V Replica, refer to the Hyper-V Replica Overview at https://technet.microsoft.com/en-us/library/jj134172.aspx

The following sections outline the workflow for using SteelEye DataKeeper.

You can replicate a virtual machine across LAN, WAN, or any Windows server through SIOS Microsoft Management Console (MMC) interface. You can run the DataKeeper MMC snap-in from any server. The DataKeeper MMC snap-in interface is similar to the existing Microsoft Management tools.

Note: For information on installing the SteelEye DataKeeper, refer to SteelEye DataKeeper for Windows Server 2003/2008 Planning and Install Guide and SteelEye DataKeeper for Windows Server 2003/2008 Administration Guide. Ensure that the local security policies, firewall, and port settings are configured as per the details in these documents.

The following procedures help you set up a virtual machine in the Disaster Recovery environment.

Configure Virtual Machines

Create a SteelEye mirroring job and then create a virtual machine in the disk.

After creating the virtual machine, you need to add the dependency between the virtual machine and the datakeeper volume in the cluster. This dependency triggers the switching of the source and target Servers of the SteelEye DataKeeper Volume resource when failover of the virtual machines occurs in the Failover Cluster Manager.

You can create multiple virtual machines.

Configure a Virtual Machine

After creating a DataKeeper mirroring job, you need to create a virtual Adding the Dependency between the Virtual Machine and the Disk in the Cluster

After creating the virtual machine, you need to add the dependency between the virtual machine and the datakeeper volume in the cluster. This dependency triggers the switching of the source and target Servers of the SteelEye DataKeeper Volume resource when failover of the virtual machines occurs in the Failover Cluster Manager.

Configure System Platform Products in a Typical Medium Scale Virtualization

The expected Recovery Time Objective (RTO) and Recovery Point Objective (RPO), trends and various observations in a medium scale virtualization environment are recorded by performing tests with System Platform Product configuration.

The virtualization host server used for medium scale configuration consists of seven virtual machines listed below.

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

- Node 1 (GR): GR, InTouch and DAS SI Direct – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit
- Node 2 (AppEngine1): Bootstrap, IDE and InTouch (Managed App) – Windows 2008 R2 Standard edition (64bit) OS

- Node 3 (AppEngine2): Bootstrap, IDE – Windows 2008 R2 Standard edition (64bit) OS
- Node 4: Historian – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit
- Node 5: Information Server, Bootstrap and IDE – Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007
- Node 6: InTouch Terminal Service – Windows 2008 R2 Standard edition (64bit) OS enabled with Terminal Service
- Node 7: Historian Client and InTouch – Windows 7 Professional Edition (64bit) OS with SQL Server 2008 SP1 32 bit

The following table displays the approximate data of virtual nodes, IO tags and historized tags in a medium scale virtualization environment:

Virtual Node	IO tags (Approx.)	Historized tags (Approx.)
AppEngine1	25000	10000
AppEngine2	25000	10000

Expected Recovery Time Objective and Recovery Point Objective

This section provides the indicative Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for the load of IO and Attributes historized shown above and with the configuration of Host Virtualization Servers and Hyper-V virtual machines explained in the Setup instructions of Medium Scale Virtualization. For more information refer to *"Set Up Medium Scale Virtualization Environment"*. In addition to these factors, the exact RTO and RPO depend on factors like storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

RTO and RPO Observations - DR Medium Configuration

Important: The following sample data are provided only as guidelines for establishing testing specific to your needs.

Scenario	Observation
Scenario 1: IT provides maintenance on Virtualization Server	"Live Migration"
	"Quick Migration of all nodes simultaneously"
	"Shut down of host server"
Scenario 2: Virtualization Server hardware fails	"Scenario 2: Virtualization Server hardware fails"
Scenario 3: Network fails on Virtualization Server	"Scenario 3: Network fails on Virtualization Server"

Scenario	Observation
Scenario 4: Virtualization Server becomes unresponsive	"Scenario 4: Virtualization Server becomes unresponsive"

The following tables display RTO and RPO Observations with approximately 50000 IO points with approximately 20000 attributes being historized:

Scenario 1: IT provides maintenance on Virtualization Server

Live Migration

Product	RTO	RPO	
		Tags	Data Loss Duration
InTouch HMI	9 sec	Data Loss for $Second tag (Imported to Historian)	1min 52 sec
GR	8 sec	IAS tag (Script)	13 sec
		IAS IO tag (DASSiDirect)	1 min 35 sec
AppEngine1	7 sec	IAS tag (Script)	15 sec
		IAS IO Tag (DASSiDirect)	1 min 13 sec
AppEngine2	13 sec	IAS tag (Script)	15 sec
		IAS IO tag (DASSiDirect)	1 min 14 sec
Historian Client	27 sec	SysTimeSec (Historian)	17 sec
		$Second (InTouch)	26 sec
		IAS tag (Script)	0 (data is SFed)
		IAS IO tag (DASSiDirect)	0 (data is SFed)
DAServer SIDirect	13 sec	N/A	N/A

Historian Client	12 sec	N/A	N/A
Information Server	9 sec	N/A	N/A
InTouch HMI	1 min 18 sec	Data Loss for $Second tag (Imported to Historian)	1min 23 sec
GR	1 min 55 sec	IAS tag (Script)	2 min 43 sec
		IAS IO tag (DASSiDirect)	2 min 55 sec
AppEngine1	3 min 25 sec	IAS Tag (Script)	3 min 40 sec
		IAS IO Tag (DASSiDirect)	3min 49 sec
AppEngine2	2 min 20 sec	IAS Tag (Script)	2 min 48 sec
		IAS IO tag (DASSiDirect)	2 min 54 sec
Historian Client	6 min 27 sec	SysTimeSec (Historian)	5 min 57 sec
		$Second (InTouch)	6 min 19 sec
		IAS tag (Script)	0 (data is SFed)
		IAS IO tag (DASSiDirect)	0 (data is SFed)
DAServer SIDirect	2min 1 sec	N/A	N/A
InTouch HMI	1 min 18 sec	Data Loss for $Second tag (Imported to Historian)	1min 23 sec

Product	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
GR	1 min 55 sec	IAS tag (Script)	2 min 43 sec

		IAS IO tag (DASSiDirect)	2 min 55 sec
AppEngine1	3 min 25 sec	IAS Tag (Script)	3 min 40 sec
		IAS IO Tag (DASSiDirect)	3min 49 sec
AppEngine2	2 min 20 sec	IAS Tag (Script)	2 min 48 sec
		IAS IO tag (DASSiDirect)	2 min 54 sec
Historian Client	6 min 27 sec	SysTimeSec (Historian)	5 min 57 sec
		$Second (InTouch)	6 min 19 sec
		IAS tag (Script)	0 (data is SFed)
		IAS IO tag (DASSiDirect)	0 (data is SFed)
DAServer SIDirect	2min 1 sec	N/A	N/A

Quick Migration of all nodes simultaneously

Quick Migration of all nodes occurs simultaneously to migrate all nodes.

Product	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch	3 min 29 sec	Data Loss for $Second tag (Imported to Historian)	12 min 8 sec
GR	6 min 11 sec	IAS tag (Script)	6 Min 35 sec
		IAS IO tag (DASSiDirect)	7 Min 26 sec
AppEngine1	8 min 12 sec	IAS tag (Script)	8 Min 6 sec
		IAS IO Tag (DASSiDirect)	8 Min 28 sec

AppEngine2	6min 6 sec	IAS tag (Script)	6 min 58 sec
		IAS IO tag (DASSiDirect)	7 min 34 sec
Historian	11 min 59 sec	SysTimeSec (Historian)	12 min 2 sec
		$Second (InTouch)	12 min 8 sec
		IAS tag (Script)	6 min 35 sec
		IAS IO tag (DASSiDirect)	7 min 26 sec
DAS SIDirect	6 min 48 sec	N/A	N/A
Historian Client	9 min 4 sec	N/A	N/A
Information Server	4 min 59 sec	N/A	N/A

Shut down of host server

Product	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch	12 min 32 sec	Data Loss for $Second tag (Imported to Historian)	14 min
GR	11 min 41 sec	IAS tag (Script)	12 Min 58 sec
		IAS IO tag (DASSiDirect)	13 Min 11 sec
AppEngine1	11 min 38 sec	IAS tag (Script)	12 Min 6 sec
		IAS IO Tag (DASSiDirect)	13 Min 49 sec
AppEngine2	11 min 57 sec	IAS tag (Script)	12 Min 58 sec
		IAS IO tag (DASSiDirect)	13 Min 54 sec
Historian	12 Min 55 sec	SysTimeSec (Historian)	13 Min

		$Second (InTouch)	14 Min
		IAS tag (Script)	12 Min 58 sec
		IAS IO tag (DASSiDirect)	13 Min 11 sec
DAS SIDirect	6 Min 48 sec	N/A	N/A
Historian Client	9 Min 4 sec	N/A	N/A
Information Server	4 Min 59 sec	N/A	N/A

Scenario 2: Virtualization Server hardware fails

The failover occurs due to hardware failure, and it is simulated with power-off on the host server.

Product	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch	11 Min 43 sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	12 Min 27 **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.
GR	10 Min 51 sec	IAS tag (Script)	11 Min 16
		IAS IO tag (DASSiDirect)	11 Min 02
AppEngine1	10 min 29 sec	IAS tag (Script)	10 Min 40
		IAS IO Tag (DASSiDirect)	11 Min 16
AppEngine2	10 min 59 sec	IAS tag (Script)	9 Min 26
		IAS IO tag (DASSiDirect)	11 Min 08

Product	RTO	RPO	
		Tags	Data Loss Duration
Historian	14 Min 49 sec	SysTimeSec (Historian)	12 Min 21
		$Second (InTouch)	12 Min 27 **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node which historizes this tag.
		IAS tag (Script)	11 Min 16
		IAS IO tag (DASSiDirect)	11 Min 02
DAS SIDirect	11 Min 20 sec	N/A	N/A
Historian Client	7 Min 16 sec + time taken by the user to start the Historian Client	N/A	N/A
Information Server	9 Min 39 sec + time taken by the user to start the Information Server	N/A	N/A
Historian Client	7 Min 16 sec + time taken by the user to start the Historian Client	N/A	N/A
Information Server	9 Min 39 sec + time taken by the user to start the Information Server	N/A	N/A

Scenario 3: Network fails on Virtualization Server

There is a failover due to network disconnect (Public). In this case, the VMs restart, after moving to the other host server.

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
InTouch	8 min 55 sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	14 min
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
GR	11 min 32 sec	IAS Tag (Script)	12 min 01
		IAS IO Tag (DASSiDirect)	12 min
AppEngine1	10 min 52 sec	IAS Tag (Script)	11 min 26
		IAS IO Tag (DASSiDirect)	11 min 58
AppEngine2	10 min 28 sec	IAS Tag (Script)	10 min 19
		IAS IO Tag (DASSiDirect)	12 min 04

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
Historian	13 min 20 sec	SysTimeSec (Historian)	13 min 52

		$Second (InTouch)	14 min **Note:** RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.
		IAS Tag (Script)	12 min 01
		IAS IO Tag (DASSiDirect)	12 min
DAS SIDirect	9 min 9 sec	N/A	N/A
Historian Client	8 min + time taken by the user to start the Historian Client	N/A	N/A
Information Server	8 min 25 sec + time taken by the user to start the Information Server	N/A	N/A

Scenario 4: Virtualization Server becomes unresponsive

There is no failover of VMs to the other host server when the CPU utilization on the host server is 100%.

Primary Node	Products	RTO (sec)	RPO
InTouch	N/A	N/A	N/A
GR	N/A	N/A	N/A
	N/A	N/A	N/A
AppEngine1	N/A	N/A	N/A
	N/A	N/A	N/A
AppEngine2	N/A	N/A	N/A
	N/A	N/A	N/A
Historian	N/A	N/A	N/A
	N/A	N/A	N/A

	N/A	N/A	N/A
	N/A	N/A	N/A
DAS SIDirect	N/A	N/A	N/A
Historian Client	N/A	N/A	N/A
Information Server	N/A	N/A	N/A

CHAPTER 6

Implementing Disaster Recovery Using vSphere

The workflows described below are designed to help you set up and implement Disaster Recovery using VMware vSphere. These workflows assume that VMware ESXi™ 5.0 or above, vCenter Server™, and vSphere Client already installed.

For basic procedures to install these and other VMware products, see product support and user documentation at http://www.vmware.com/.

The Disaster Recovery vSphere implementation assumes that you are implementing a medium-scale system.

This section contains the following topics:

- *Plan the Virtualization Environment*
- *Configure System Platform Products in a Typical Virtualization Environment*
- *Set Up the Virtualization Environment*
- *Recover Virtual Machines to a Disaster Recovery Site*

In This Chapter

Plan the Virtualization Environment

The recommended hardware and software requirements for the Host and Virtual machines used for the virtualization Disaster Recovery environment are as follows:

ESXi Hosts	
Processor	Two 2.79 GHz Intel Xeon with 8 Cores (Hyper-threaded)
Operating System	ESXi 5.0 or above
Memory	48 GB
Storage	SAN with 1TB storage disk

Note: For the ESXi Host to function optimally, the server should have the same processor, RAM, storage, and service pack level. To avoid hardware discrepancies, the servers should preferably be purchased in pairs. Though differences are supported, it will impact the performance during failovers.

Virtual Machines

Using the specified ESXi host configuration, seven virtual machines can be created in the environment with the following configuration.

Virtual Machine 1: Historian Node

Processor	Host Compatible Processor with 2-4 Cores
irtual CPUs	4 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	200 GB
System Platform Products Installed	Historian

Virtual Machine 2: Application Server Node, DAS SI

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	4 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	100 GB
System Platform Products Installed	ArchestrA-Runtime, DAS SI

Virtual Machine 3: InTouch TS Node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch with TS enabled

Virtual Machine 4: Application Server Runtime Node 1

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only and InTouch

Virtual Machine 5: Application Server Runtime Node 2

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	2 vCPUs
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only

Virtual Machine 6: Information Server Node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	1 vCPU
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Information Server

Virtual Machine 7: Historian Client Node

Processor	Host Compatible Processor with 2-4 Cores
Virtual CPUs	1 vCPUs

Operating System	Windows 7 or higher Enterprise
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Historian Client

Network Requirements

For this architecture, you can use one physical network card that needs to be installed on a host computer for the domain network and the process network.

Configure System Platform Products in a Typical Virtualization Environment

The expected Recovery Time Objective (RTO) and Recovery Point Objective (RPO), trends, and various observations in a virtualization environment are recorded by performing tests with System Platform Product configuration.

The virtualization host server consists of the following seven virtual machines:

Important: The following information is provided as an example of this kind of configuration, and is not intended to be used as instructions to set up and configure your environment.

- Node 1 (GR): GR, InTouch and DAS SI Direct – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit
- Node 2 (AppEngine1): Bootstrap, IDE and InTouch (Managed App) – Windows 2008 R2 Standard edition (64bit) OS
- Node 3 (AppEngine2): Bootstrap, IDE – Windows 2008 R2 Standard edition (64bit) OS
- Node 4: Historian – Windows 2008 R2 Standard edition (64bit) OS with SQL Server 2008 SP1 32 bit
- Node 5: Information Server, Bootstrap and IDE – Windows Server 2008 SP2 (32bit) with SQL Server 2008 SP1 and Office 2007
- Node 6: InTouch Terminal Service – Windows 2008 R2 Standard edition (64bit) OS enabled with Terminal Service
- Node 7: Historian Client and InTouch – Windows 7 Professional Edition (64bit) OS with SQL Server 2008 SP1 32 bit

The following table displays the approximate data of virtual nodes, IO tags and historized tags in the virtualization environment:

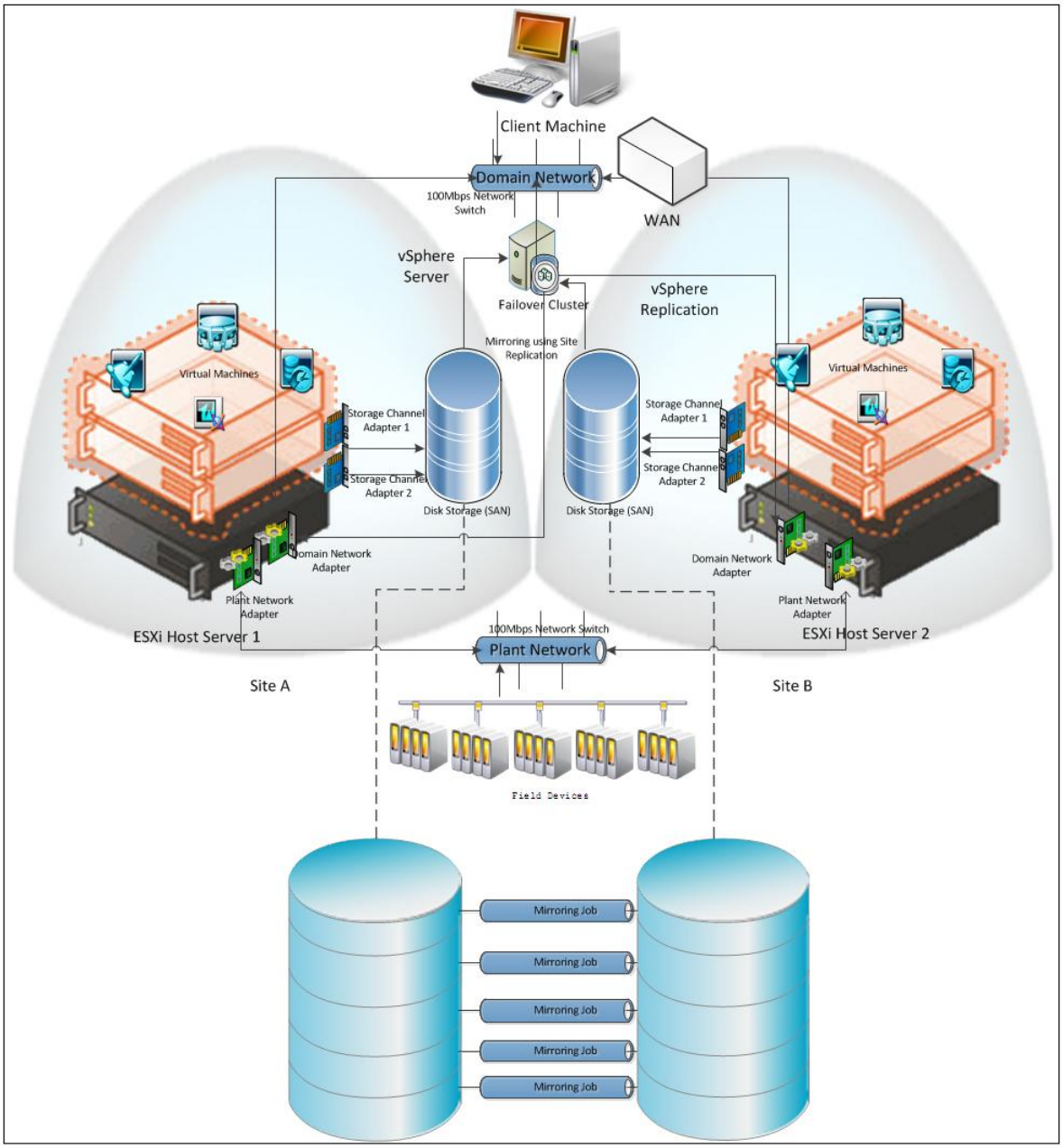
Virtual Node	IO tags (Approx.)	Historized tags (Approx.)
AppEngine1	25000	10000
AppEngine2	25000	10000

Set Up the Virtualization Environment

Use the following workflows to set up the virtualization environment for Disaster Recovery using vSphere technology.

Create a Datacenter

The vSphere Datacenter virtualizes an infrastructure that includes servers, storage, networks, and provides for end-to-end connectivity from client machines to field devices and back. The recommended topology of the Datacenter for a Disaster Recovery is:



The following workflow requires a minimum of two host servers and two storage servers connected to each host independently. This workflow will help you configure a virtualized Disaster Recovery environment consisting of a Datacenter with a Failover Cluster that has two nodes and two Storage Area Networks (SANs).

Create the Datacenter

Use the vSphere Client to create the DataCenter. Refer to your vSphere documentation for additional information.

Add hosts to the Datacenter

Refer to your vSphere documentation or the VMware knowledge base for additional information and add an ESXi host.

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896

Repeat this procedure to add another ESXi host.

Create a Failover Cluster

A cluster in vSphere is a group of hosts. Resources of a host added to a cluster, also known as a failover cluster, become part of the cluster's resources, and are managed by the cluster.

Refer to your vSphere documentation or the VMware knowledge base for information about adding the failover cluster.

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032896

To create a failover cluster:

1. Select the new cluster option.
2. Select vSphere HA.
3. Use the New Cluster Wizard to complete configuration.

Configure Storage

VMware Virtual Machine File System (VMFS) datastores serve as repositories for the virtual machines. You can set up VMFS datastores on any SCSI-based storage devices that the host discovers, including Fibre Channel, iSCSI, and local storage devices.

Use the following workflow to create a datastore. Your new datastore is added to all hosts if you use the vCenter Server system to manage your hosts.

Important: Install and configure any adapters that your storage requires before creating datastores. After you create a datastore, rescan the adapters to discover the new storage device.

Create a datastore

1. Log on to vSphere Client and select a host.
2. Configure storage for the host.
3. Configure the file system version.
4. Check the parameters you have selected and create the datastore.

Configure Networks

After you create a datacenter, use the following the workflow to configure one or more networks on the ESXi host.

Configure networks on the ESXi host

1. Log on to vSphere Client and select a host
2. Add networking through the Add Network Wizard.
3. Click **Add Networking.** The **Add Network Wizard** appears.
4. Use the Wizard to complete network configuration.

After you create a datacenter, add a host and configure storage. You can configure multiple networks on the ESXi host networks.

Create a Virtual Machine in the vSphere Client

You can populate your virtualization environment by creating virtual machines, which are the key components in a virtual infrastructure.

When you create a virtual machine, you associate it to a datastore and datacenter, host, cluster or resource pool. The virtual machine consumes resources dynamically as the workload increases, or it returns resources dynamically as the workload decreases.

Every virtual machine has virtual devices that provide the same function as physical hardware. A virtual machine gets CPU and memory, access to storage, and network connectivity from the host with which it is associated.

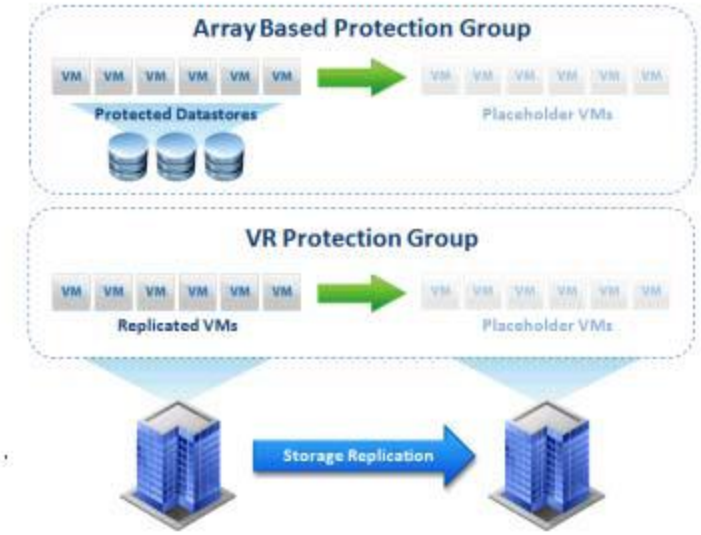
Set up Replication

Replicating live virtual machines ensures that a duplicate copy is available in case the primary storage array fails. This helps in Disaster Recovery without impacting production. Set up vSphere replication through the vSphere client.

Configure Protection Groups

Protection groups identify the virtual components that are considered to be most important for maintaining business continuity. Protection groups can define groups of associated VMs that should be recovered together, such as Infrastructure (Windows Active Directory or DNS), Mission Critical, or Business Critical.

Storage array-based protection groups include protected datastores. VR protection groups include replicated VMs. Recovery plans, detailed later in this chapter, are encapsulations of one or more protection groups stored at the recovery site to define the Disaster Recovery failover process.



Create a Recovery Plan

After creating the protection group, you must create the recovery plan for Disaster Recovery. Use vSphere Client to create the recovery plan.

Recover Virtual Machines to a Disaster Recovery Site

To recover the virtual machines in case of a disaster at a site, you must you must set up a disaster recovery site through vSphere Client.

CHAPTER 7

Implementing High Availability and Disaster Recovery Using Virtualization

This section introduces several High Availability and Disaster Recovery (HADR) virtualization solutions that improve the availability of System Platform Products. A HADR solution offsets the effects of a hardware or software failure across multiple sites during a disaster. It makes sure all applications are available in order to minimize the downtime during times of crisis.

Important: The information and procedures in this chapter are specific to Hyper-V. You can implement a VMware HADR virtualization solution by following the procedures and settings in *Chapter 3, "Implementing High Availability Using vSphere,"*and in *Chapter 5, "Implementing Disaster Recovery Using vSphere."*

In This Chapter

Working with a Medium Scale Virtualization Environment

This section contains the following topics:

- *Set Up the Virtualization Environment*
- *Expected Recovery Time Objective and Recovery Point Objective*

Set Up the Virtualization Environment

The following procedures help you to set up and implement the high availability and disaster recovery for the medium scale virtualization environment.

Plan the Virtualization Environment

The minimum recommended hardware and software requirements for the Host and Virtual machines used for this virtualization environment are provided in the table below:

Hyper-V Hosts

Processor	Two 2.79 GHz Intel Xeon Processor with 24 Cores
Operating System	Windows Server 2008 R2 or higher Enterprise with Hyper-V enabled
Memory	48 GB
Storage	SAN with 1 TB storage disk

Note: For the Hyper-V Host to function optimally, the server should have the same processor, RAM, storage, and OS version (including service packs). Preferably, the servers should be purchased in pairs to avoid hardware discrepancies. Though the differences are supported, it impacts the performance during failovers.

Virtual Machines

Using the Hyper-V host specified above, seven virtual machines can be created in the environment with the configuration given below.

Virtual Machine 1: Historian Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	200 GB
System Platform Products Installed	Historian

Virtual Machine 2: Application Server Node and OI SI

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	8 GB
Storage	100 GB
System Platform Products Installed	ArchestrA-Runtime and DAS SI

Virtual Machine 3: InTouch TS Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch with TS enabled

Virtual Machine 4: Application Server Runtime Node 1

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	InTouch and Application Server Runtime only

Virtual Machine 5: Application Server Runtime Node 2

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 R2 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Application Server Runtime only

Virtual Machine 6: Information Server Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows Server 2008 or higher Standard
Memory	4 GB
Storage	80 GB
System Platform Products Installed	Information Server

Virtual Machine 7: Historian Client Node

Processor	Host Compatible Processor with 2-4 Cores
Operating System	Windows 7 or higher Enterprise
Memory	4 GB
Storage	80 GB

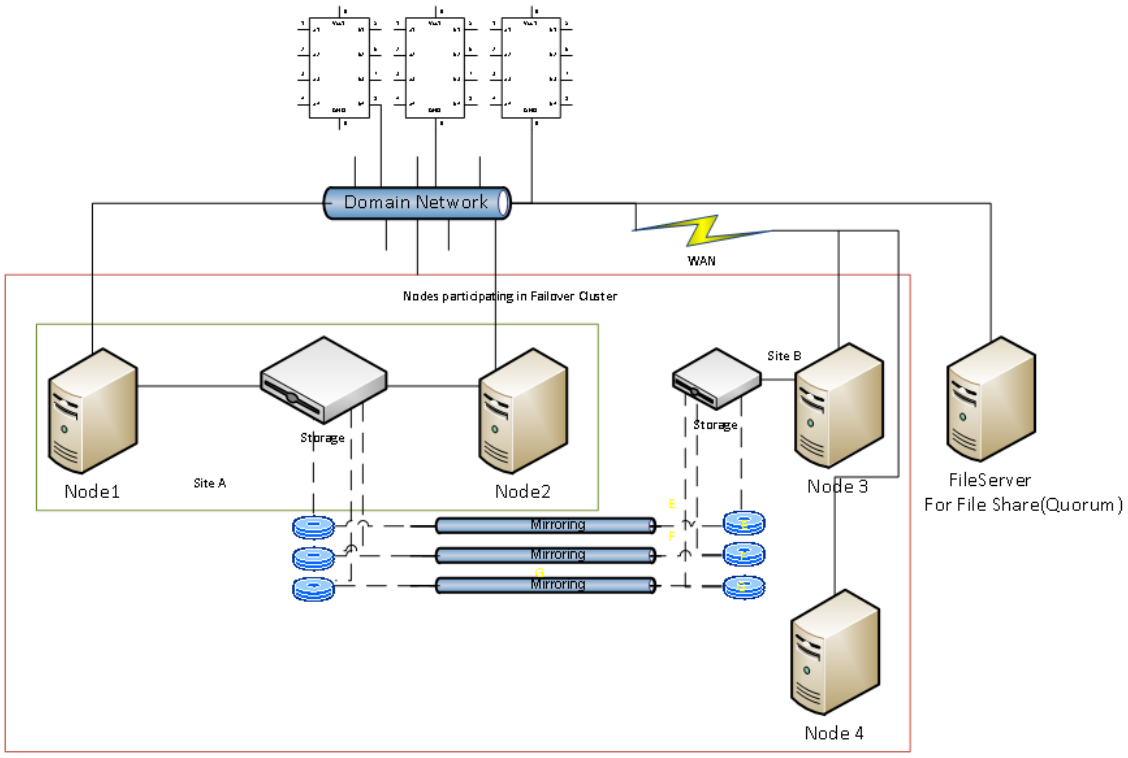
System Platform Products Installed	Historian Client

Network Requirements

For this architecture, you can use one physical network card that needs to be installed on a host computer for both the domain and the process networks.

Configure a Failover Cluster

The following diagram shows the recommended topology of the failover cluster for high availability and disaster recovery for the virtualization environment:



The following workflow will guide you on how to setup high availability and disaster recovery for medium scale virtualization environment.

This setup requires a minimum of three host servers and two storage servers with sufficient disk space to host the virtual machines on each disk. One storage server is shared across two servers on one site and another storage server is connected to the third host. Each disk created on the storage server is replicated in all the sites for disaster recovery. Node 4 is used for Node Majority in the failover cluster. Another independent node is used for configuring the quorum. For more information on configuring the quorum, refer to "Configure Cluster Quorum Settings".

Install Failover Cluster

To install the failover cluster feature, you need to run Windows Server 2008 R2 or higher Enterprise Edition on your server.

To install failover cluster on a server

1. Go to Initial Configuration Tasks > Customize This Server > Add features.
2. Select Failover Clustering.

Validate Cluster Configuration

Before creating a cluster, you must validate your configuration. Validation helps you to confirm that the configuration of your servers, network, and storage meet the specific requirements for failover clusters. Use the Failover Cluster Manager to validate the configuration.

Create a Cluster

To create a cluster, run the Create Cluster wizard from the Server Manager.

To create a cluster

1. From the Failover Cluster Manager, open the Create Cluster Wizard.
2. Enter the server name to be added. Skip the validation test.
3. Enter the cluster name and close the wizard.

Configure Cluster Quorum Settings

Quorum is the number of elements that need to be online to enable continuous running of a cluster. In most instances, the elements are nodes. In some cases, the elements also consist of disk or file share witnesses. Each of these elements determines whether the cluster should continue to run or not.

All elements, except the file share witnesses, have a copy of the cluster configuration. The cluster service ensures that the copies are always synchronized. The cluster should stop running if there are multiple failures or if there is a communication error between the cluster nodes.

After both nodes have been added to the cluster, and the cluster networking components have been configured, you must configure the failover cluster quorum.

You must create and secure the file share that you want to use for the node and the file share majority quorum before configuring the failover cluster quorum. If the file share has not been created or correctly secured, the following procedure to configure a cluster quorum will fail. The file share can be hosted on any computer running a Windows operating system.

To configure the cluster quorum, you need to perform the following procedures:

- Create and secure a file share for the node and file share majority quorum
- Use the failover cluster management tool to configure a node and file share majority quorum

After you configure the cluster quorum, validate the cluster. For more information, refer to http://technet.microsoft.com/en-us/library/bb676379(EXCHG.80).aspx.

Configure Storage

For any virtualization environment, storage is one of the central barriers to implementing a good virtualization strategy. However in Hyper-V, VM storage is kept on a Windows file system. You can put VMs on any file system that a Hyper-V server can access. As a result, you can build HA into the virtualization platform and storage for the virtual machines. This configuration can accommodate a host failure by making storage accessible to all Hyper-V hosts so that any host can run VMs from the same path on the shared folder. The back-end of this storage can be a local, storage area network, iSCSI or whatever is available to fit the implementation.

The following table lists the minimum storage recommendations for each VM in medium scale virtualization environment:

System	Storage Capacity
Historian Virtual Machine	200 GB
Application Server 1 (GR Node) Virtual Machine	100 GB
Application Engine 2 (Runtime Node) Virtual Machine	80 GB
InTouch Virtual Machine	80 GB
Information Server Virtual Machine	80 GB
Historian Client	80 GB

To build up High Availability and Disaster Recovery system, you must have a minimum of two SAN storage servers, each installed at different sites with the above storage recommendations.

The total storage capacity should be minimum recommended 1 TB.

Configure Hyper-V

Microsoft Hyper-V helps in creating virtual environment that improves server utilization. It enhances patching, provisioning, management, support tools, processes, and skills. Microsoft Hyper-V provides live migration, cluster shared volume support, expanded processor, and memory support for host systems.

Hyper-V is available in x64-based versions of Windows Server 2008 R2 and higher operating system.

The following are the pre-requisites to set up Hyper-V:

- x64-based processor
- Hardware-assisted virtualization
- Hardware Data Execution Prevention (DEP)

Configure SIOS (SteelEye) DataKeeper and Hyper-V Replica

SteelEye DataKeeper and Hyper-V Replica are replication software for real-time Windows data. Both can be used to replicate all data types, including the following:

- Open files

- SQL and Exchange Server databases
- Hyper-V .vhd files

The ability of both SteelEye DataKeeper and Hyper-V Replica to replicate logical disk partitions hosting the .vhd files for the Hyper-V virtual machines ensures that a mirrored disk image is available on the stand-by cluster host in case the primary cluster host fails. This helps provide disaster recovery (DR) solutions.

SteelEye DataKeeper Cluster Edition is a host-based replication solution, which extends Microsoft Windows Server Failover Clustering (WSFC) and Microsoft Cluster Server (MSCS) features such as cross-subnet failover and tunable heartbeat parameters. These features make it possible to deploy geographically distributed clusters.

You can replicate .vhd files across LAN, WAN, or any Windows server through SIOS Microsoft Management Console (MMC) interface. You can run the DataKeeper MMC snap-in from any server. The DataKeeper MMC snap-in interface is similar to the existing Microsoft Management tools.

Note: For information on installing the SteelEye DataKeeper, refer to http://www.steeleye.com. Ensure that the local security policies, firewall, and port settings are configured as per the details provided in the SteelEye DataKeeper documents. For information on using Hyper-V Replica, refer to the Hyper-V Replica Overview at https://technet.microsoft.com/en-us/library/jj134172.aspx

The following workflow outlines how to set up a virtual machine in the Disaster Recovery environment, using SteelEye DataKeeper.

1. Create a SteelEye DataKeeper Mirroring Job.
2. Enter the relevant job name and description
3. Choose a source and a target.
4. Select the destination server, IP address and volume.
5. After you have completed setting up SteelEye DataKeeper Mirroring jobs and created the datakeeper, you can view the disk management topologies.

Configure Virtual Machines

After creating a steel eye mirroring job, you need to create a virtual machine in the disk.

Use the New Virtual Machine Wizard to create and configure a new virtual machine.

Add the Dependency between the Virtual Machine and the DataKeeper volume in the Cluster

After creating the virtual machine, you need to add the dependency between the virtual machine and the datakeeper volume in the cluster. This dependency triggers the switching of the source and target servers of the SteelEye DataKeeper Volume resource when failover of the virtual machines occurs in the Failover Cluster Manager.

Expected Recovery Time Objective and Recovery Point Objective

This section provides the indicative Recovery Time Objectives (RTO) and Recovery Point Objectives (RPO) for the load of IO and Attributes historized as shown in Configuring System Platform Products in a Typical Medium Scale Virtualization in Chapter 3 and with the configuration of Host Virtualization Servers and Hyper-V virtual machines explained in the setting up Medium Scale Virtualization Environment. For more information refer to, *"Set Up the Virtualization Environment"*. In addition to these factors, the exact RTO and RPO depend on factors like storage I/O performance, CPU utilization, memory usage, and network usage at the time of failover/migration activity.

RTO and RPO Observations - HADR Medium Configuration

Important: The following sample data are provided only as guidelines for establishing testing specific to your needs.

Scenarios and observations in this section:

Scenario	Observation
HA-Scenario: Virtualization Server hardware fails	"HA-Scenario: Virtualization Server hardware fails"
DR-Scenario: Network fails on Virtualization Server	"DR-Scenario: Network fails on Virtualization Server"

The following tables display RTO and RPO Observations with approximately 50000 IO points with approximately 20000 attributes being historized:

HA-Scenario: Virtualization Server hardware fails

The failover occurs due to hardware failure, and it is simulated with power-off on the host server.

Products	RTO	RPO	
		Tags	Data Loss Duration
InTouch	5 min 35 sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	6 min 47 sec
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
GR	5 min 13 sec	IAS Tag (Script)	5 min 44 sec
		IAS IO Tag (DASSiDirect)	7 min 28 sec
AppEngine1	6 min 05 sec	IAS Tag (Script)	6 min 35 sec
		IAS IO Tag (DASSiDirect)	7 min 29 sec

AppEngine2	6 Min 12 sec	IAS Tag (Script)	6 Min 41 sec
		IAS IO Tag (DASSiDirect)	7 Min 20 sec

Products	**RTO**	**RPO**	
		Tags	**Data Loss Duration**
Historian	6 min 21 sec	SysTimeSec (Historian)	6 Min 33 sec
		$Second (InTouch)	6 Min 47 sec
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
		IAS Tag (Script)	5 Min 45 sec
		IAS IO Tag (DASSiDirect)	7 Min 30 sec
DAS SIDirect	4 Min 25 sec	N/A	N/A
Historian Client	3 Min 34 sec + time taken by the user to start the Historian Client	N/A	N/A
Information Server	4 Min 15 sec + time taken by the user to start the Information Server	N/A	N/A

DR-Scenario: Network fails on Virtualization Server

There is a failover due to network disconnect (Public). In this case, the VMs restart after moving to the other host server.

Products	RTO	RPO	
		Tags	Data Loss Duration
InTouch	11 min 4 sec + time taken by the user to start the InTouchView	Data Loss for $Second tag (Imported to Historian)	15 min 32 sec
		Note: RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.	
GR	12 min 20 sec	IAS Tag (Script)	13 min 11 sec
		IAS IO Tag (DASSiDirect)	13 min 01 sec
AppEngine1	11 min 35 sec	IAS Tag (Script)	12 min 26 sec
		IAS IO Tag (DASSiDirect)	13 min 05 sec
AppEngine2	11 min 48 sec	IAS Tag (Script)	11 min 24 sec
		IAS IO Tag (DASSiDirect)	13 min 19 sec
Historian	20 min 0 sec	SysTimeSec (Historian)	15 min 16 sec
		$Second (InTouch)	15 min 32 sec RPO is dependent on the time taken by the user to start the InTouchView on the InTouch node and the RTO of the Historian node, which historizes this tag.
		IAS Tag (Script)	13 min 11 sec
		IAS IO Tag (DASSiDirect)	13 min 01 sec
DAS SIDirect	12 min 25 sec	N/A	N/A

Historian Client	5 min 32 sec + time taken by the user to start the Historian Client	N/A	N/A
Information Server	5 min 38 sec + time taken by the user to start the Information Server	N/A	N/A

CHAPTER 8

Working with Windows Server

This chapter provides an overview of Windows Server features as they relate to following functions:

- *Communication Between System Platform Nodes with VLAN*
- *RMC Communication Between Redundant Application Server Nodes with VLAN*
- *Access a System Platform Node with a Remote Desktop*
- *Access System Platform Applications as Remote Applications*
- *Display the System Platform Nodes on a Multi-Monitor with a Remote Desktop*
- *Network Load Balancing*
- *Hardware Licenses in a Virtualized Environment*

In This Chapter

About Microsoft Hyper-V

Hyper-V is available both as a standaone product and as part of Windows 2008 R2 Server and higher. Choosing which one to use will generally be an issue of licensing costs. While the standalone Hyper-V is available as a free download, you will have to license the virtual machines that will run on top of the hypervisor. If you have Windows Server Datacenter, you can run unlimited numbers of VMs without having to pay for additional licenses.

A virtualized environment can run multiple virtual machines (VMs) on a single server, thereby reducing the number of physical servers required on the network. Hyper-V provides a virtualized computing environment on Windows Server. Hyper-V is a hardware-assisted virtualization platform that uses partitions to host VMs. One of the benefits that Hyper-V provides is isolation, which ensures that the child VMs execute in their individual partitions and exist on the host as separate machines. This allows multiple operating systems and conflicting applications to run on the same server.

Hyper-V provides support for using Virtual LANs (VLANs) on both parent and child partitions. By configuring VLAN, VMs can communicate over the specified VLAN using Virtual Network switch.

Microsoft introduced RemoteApp with the release of Windows 2008 Terminal Services. In the past, Windows 2008 TS Microsoft Terminal Services solutions only supported the publication of a full desktop using the RDP protocol. In Windows 2008, it was possible to start an application seamlessly from a Terminal Server making it appear as it were running locally on the client machine.

RemoteApp is an application that runs on from a Terminal Server running seamlessly to the client.

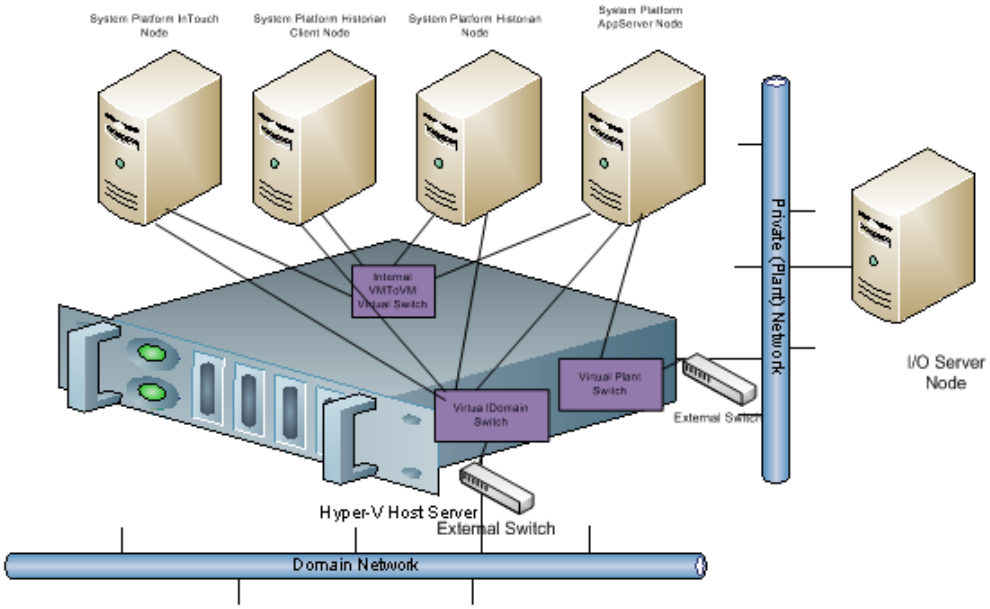
Communication Between System Platform Nodes with VLAN

Virtual LANs perform traffic separation within a shared network environment. All released versions of Hyper-V support virtual local area networks (VLANs). Since the VLAN configuration is software-based, you can move a computer and still maintain the network configurations. For each virtual network adapter you connect to a virtual machine, you can configure a VLAN ID for the virtual machine.

You need the following network adapters to configure VLANs:

- A physical network adapter that supports VLANs
- A physical network adapter that supports network packets with VLAN IDs that are already applied

On the management operating system, you need to configure the virtual network to allow network traffic on the physical port. This enables you to use the VLAN IDs internally with the VMs. You can then configure the VM to specify the virtual LAN that the VM will use for all network communications.



Configure Virtual Network Switches on the Hyper-V Host Server and Add Virtual Network Adapters on the VM Nodes

You can create virtual networks on a server running Hyper-V to define various networking topologies for VMs and the virtualization server. Following are the three types of virtual networks:

- Private network: Provides communication between VMs
- Internal network: Provides communication between the virtualization server and VMs
- External network: Provides communication between a VM and a physical network by associating to a physical network adapter on the virtualization server

On a Hyper-V host server, you can create the following virtual network adapter switches.

- External Network adapter switch to communicate with the external domain network.
- External Network adapter switch to communicate with the external plant network.

- Internal Network adapter switch to communicate between VM nodes created on Hyper-V host server.

For more information, refer to http://technet.microsoft.com/en-us/library/cc732470(WS.10).aspx

Create a Virtual Network Switch for Communication Between a VM Node and an External Domain or a Plant Network

A virtual network switch or a virtual switch is a virtual version of a physical network switch. A virtual network provides access to local or external network resources for one or more VMs. You need to create a virtual network switch to communicate with the external domain or plant network.

Note: A virtual network works like a physical network except that the switch is software based. After an external virtual network is configured, all networking traffic is routed through the virtual switch.

Use the Hyper-V Manager on a Hyper-V host to create a virtual network switch for communication between a VM node and an external domain network or a plant network. Then, use the Virtual Network Manager to add a new external virtual network.

To create a virtual network switch for communication between a VM node and an external domain network or a plant network

1. Open the Hyper-V Manager on a Hyper-V host.
2. Go to the Virtual Network Manager.
3. Add a new external virtual network.
4. Enter the new virtual network details, including the network name, and the external domain or plant network to which you are connecting,

This creates the external network switch which will be used to communicate between the VM nodes and the domain or plant network.

Create a Virtual Network Switch for Communication Between Internal VM Nodes

To communicate with the other VMs hosted on the Hyper-V host server, you need to create an internal virtual network switch.

Use the Hyper-V Manager add a new internal virtual network. The internal virtual network switch is used to communicate between the VM nodes on the host server.

To create a virtual network switch for communication between internal VM nodes

1. Open the Hyper-V Manager on a Hyper-V host.
2. Go to the Virtual Network Manager.
3. Add a new internal virtual network.
4. Enter the new virtual network name. Be sure to specify that this is an internal network.

This creates the internal network switch which will be used to communicate between the VM nodes and the host server.

Add an Internal Virtual Network Adapter to a VM Node for Communication Between VM Nodes

You can configure one or more virtual network adapters for a VM by creating or modifying the hardware profile of a VM.

If you connect a virtual network adapter configured for a VM to an internal network, you can connect to the VMs deployed on the same host and communicate over that internal network.

Use the Hyper-V Manager add and enable a new internal virtual network adapter for communication between VM nodes. All traffic for the management operating system that goes through the network adapter is tagged with the VLAN ID you provide.

To add an internal virtual network adapter to a VM node for communication between VM nodes

1. Open the Hyper-V Manager on a Hyper-V host.
2. Shut down the VM node to which you want to add the network adapter.
3. Select the hardware settings for the VM node. Add a network adapter and enter virtual LAN ID.

Note: All traffic for the management operating system that goes through the network adapter is tagged with the VLAN ID you enter.

Add a Virtual Network Adapter to a VM Node for Communication Between a VM Node and a Plant Network

If you connect a virtual network adapter configured for a VM to a physical network adapter on the host on which the VM is deployed, the VM can access the network to which the physical host computer is connected and can function on the host's local area network (LAN) in the same way that physical computers connected to the LAN can function.

Use the Hyper-V Manager add a virtual network adapter for communication between a VM node and a plant network.

To add a virtual network adapter to a VM node for communication between a VM node and a plant network

1. Open the Hyper-V Manager on a Hyper-V host.
2. Shut down the VM node to which you want to add the network adapter.
3. Select the hardware settings for the VM node and add a network adapter.

Configure Network Adapters on the System Platform Virtual Machine (VM) Nodes

By default, one network adapter is added to the VM node when you create the VM nodes on a Hyper-V host server.

Based on the requirements, you can add multiple internal or external network adapters.

For the VM System Platform node to communicate with the external domain or external plant network, it needs to have external network adapter added.

For the VM System Platform node to communicate internally to the other VM System Platform nodes hosted by the Hyper -V server, it needs to have internal network adapter added.

You can create the following VM nodes on the virtualization server for which the VLAN communication needs to be set up:

- InTouch VM node
- Historian VM node
- Application Server VM node
- Historian Client VM node
- Information Server VM node

VM nodes on Hyper-V host server have the following network adapters:

- An external network adapter to communicate with the external domain network
- An external network adapter to communicate with the external plant network. This is available if the VM node is acquiring the data from the IOServer connected to the external plant network
- An internal network adapter to communicate internally between the VM nodes configured on Hyper-V host server
- An internal network adapter to communicate between the Application Server nodes to use for Redundancy Message Channel (RMC) communication. Only the Application Server VM nodes configured for Redundant Application Engines have this network adapter.

Each System Platform node can have various combinations of the following network adapters, depending on your configuration:

Note: It is assumed that the host virtualization server is configured with one external virtual network switch to communicate with the domain network, one external virtual network switch to communicate with the plant network, and one internal virtual network switch for the internal VM to VM communication.

Product node	Network adapters
InTouch	• An external network adapter to communicate with the external domain network • An external network adapter to communicate with the external plant network (This is to acquire the data from the IOServer which is connected to the plant network.) • An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server (For example, to a Historian VM node)

Product node	Network adapters
Historian	• An external network adapter to communicate with the external domain network • An external network adapter to communicate with the external plant network (This is to acquire the data from the IOServer which is connected to the plant network.) • An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server (for example, an InTouch VM node).
Historian Client	• An external network adapter to communicate with the external domain network. • An external network adapter to communicate with the external plant network. This is to acquire the data from the IO Server which is connected to the plant network. • An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server (for example, a Historian VM node).
Information Server	• An external network adapter to communicate with the external domain network. • An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server (for example, to a Historian Client VM node).

Product node	Network adapters
Application Server	• An external network adapter to communicate with the external domain network. • An external network adapter to communicate with the external plant network. This is to acquire the data from the IO Server which is connected to the plant network. • An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server (for example, a Historian VM node).

You will need to create VM nodes with the specified OS installed on all the nodes. Configure one physical machine in the workgroup with an IO Server installed and connected to a plant or private network. Add one internal and one external virtual network adapter to the VM node. Use the same VLAN ID that you used for Configure the required VM node. Repeat for each VN node you are configuring. The general workflow is as follows:

To configure virtual network adapters on VM node

1. Add an internal virtual network adapter to the required node, for example, an InTouch node.

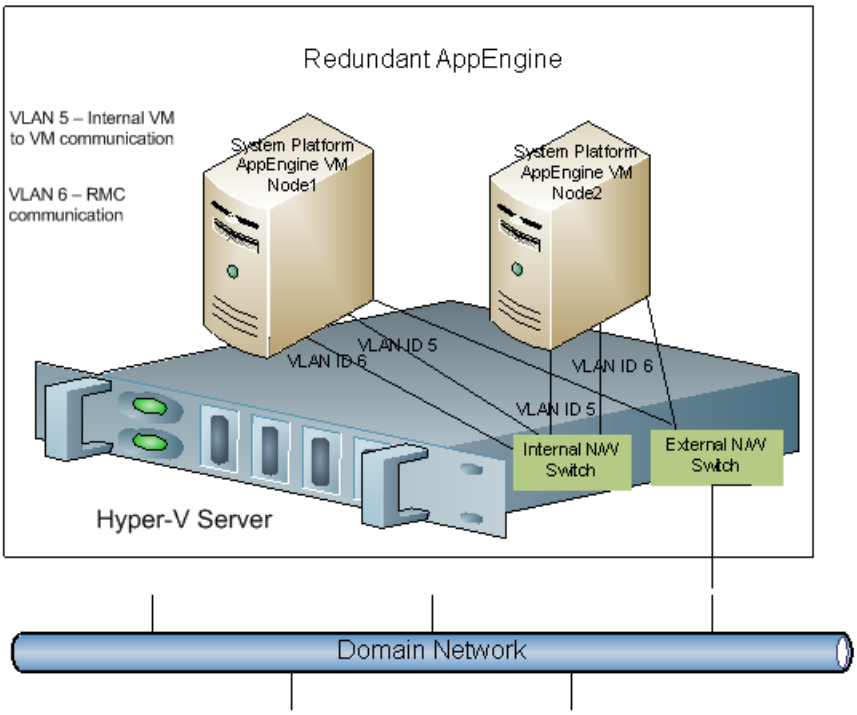
Note: You must provide the same VLAN ID that you provided for the first VM node you configured.

2. Add an external virtual network adapter to the required node, for example an InTouch node.
3. Connect to the required VM node.
4. Configure the required VM node. Select IPv4.
5. Enter the IP address for the network adapter.
   - For the internal network added for communication between VM nodes, enter the required IP address.
   - For external network adapter added for communication between a VM node and an external plant network communication, enter the required static IP address.

Note: Configure the other VM nodes following the same steps.

RMC Communication Between Redundant Application Server Nodes with VLAN

For successful communication between a redundant pair of Application Engines, each Application Engine must be assigned to a separate WinPlatform and a valid redundancy message channel (RMC) must be configured for each WinPlatform. You can configure an RMC using a virtual LAN.



Configure RMC for Redundant AppEngine over a VLAN

For a successful communication between a redundant pair of Application Engines, each Application Engine should configure a valid redundancy message channel (RMC). You can configure the RMC using Virtual LAN (VLAN). For configuring the RMC, Application Server VM System Platform node requires the internal network adapters for communication:

- An internal network adapter to communicate between the other VM nodes configured on a Hyper-V host server, for example, a Historian VM node
- An internal network adapter to communicate with the other Application Server VM nodes configured as Redundancy Application Engine to use as a RMC

To configure RMC for a Redundant AppEngine node, you will need to add an internal virtual network adapter to a Application Server node. Use the same VLAN ID for both Application Server nodes. This allows the Application Server VM nodes to internally communicate with each other over the specified LAN ID as RMC channel. The general workflow is as follows:

Note: While installing the AVEVA products, select the **Create Local Account** check box and provide the same user name and password to use as network account user.

To configure RMC for a Redundant AppEngine node

1. Add an internal virtual network adapter to a Application Server node.

Note: In the Settings window, enter the same VLAN ID that you entered while configuring the InTouch and Historian Client nodes. This enables the VM nodes to communicate internally over the specified LAN ID.

2. Add an internal virtual network adapter to a Application Server node to use as RMC communication.

Note: In the Settings window, enter the same VLAN ID you entered on both the Application Server nodes for virtual network adapter. This enables the Application Server VM node to communicate internally over the specified LAN ID as an RMC channel to communicate to another Application Server VM node.

3. Add an external virtual network adapter to a Application Server node.
4. Connect to the required Application Server VM node.
5. Configure the internal/external network adapter for the node. Be sure to select IPv4, and enter the IP address. For the internal network adapter added to use as RMC, enter the required static IP address in the IP address box and subnet mask in the Subnet mask box.

   For example:

   10.0.0.1

   255.0.0.0

6. Follow the same steps to configure another Application Server node for Redundant Application Server.

Note: Note: While installing the AVEVA products, select the Create Local Account check box and provide the same user name and password to use as network account user.

Access a System Platform Node with a Remote Desktop

You can use Hyper-V to access a system platform node through a remote desktop. You can specify the required remote users, who will be able to access the VM running the system platform.

To access a system platform node with a remote desktop, log on to the system platform node as a member of the local administrators group. Then, modify the remote settings of the system platform node to specify the remote desktop versions to which you want to allow access, and select the user to whom you want to provide access.

To access a system platform node with a remote desktop

1. Log on to the system platform node as a member of the local administrators group.
2. Modify the remote settings of the system platform node. Specify the remote desktop option you want to use.
3. Add users to allow them to access the system.

Access System Platform Applications as Remote Applications

Remote Desktop Services (RDS) Remote Applications enables you to deploy RemoteApp programs to users. With RemoteApp, the remote session connects with a specific application rather than with the entire desktop. You can access the RemoteApp programs remotely through Remote Desktop Service. A RemoteApp program appears as if it is running on your local computer. Instead of being present on the desktop of the remote terminal server, the RemoteApp program is integrated with the client's desktop, running in its own resizable window with its own entry in the task bar.

Prerequisites for accessing Remote Applications

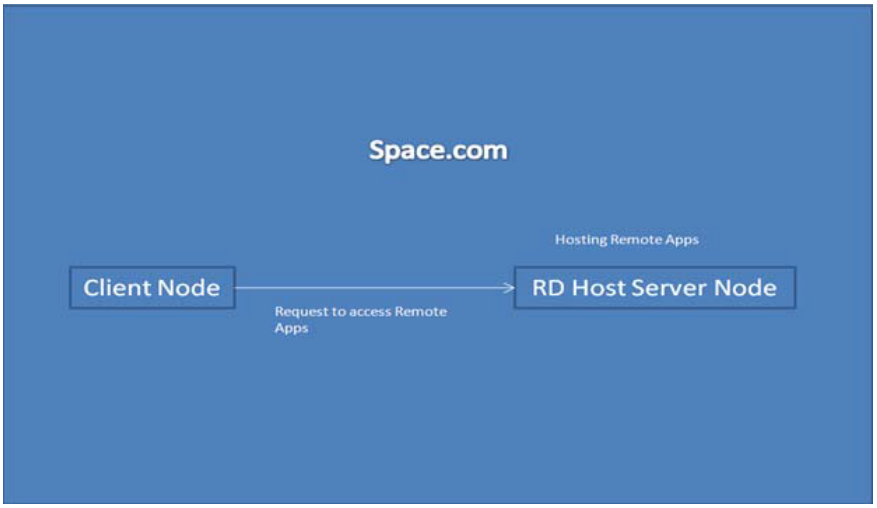
- A virtual machine node or physical node with Windows Server R2 which has Remote Desktop Session Host server installed
- Remote Applications, part of the Windows Server 2008 (or higher) Terminal Services role that are available on Windows Server 2008 and higher Standard and Enterprise Editions

- VM nodes (Remote Desktop Session Host server) running IOM Products, such as InTouch and Historian Client need to be on Windows Server 2008 R2 or higher where Remote Desktop Services are available
- Client node with a browser (any operating system)

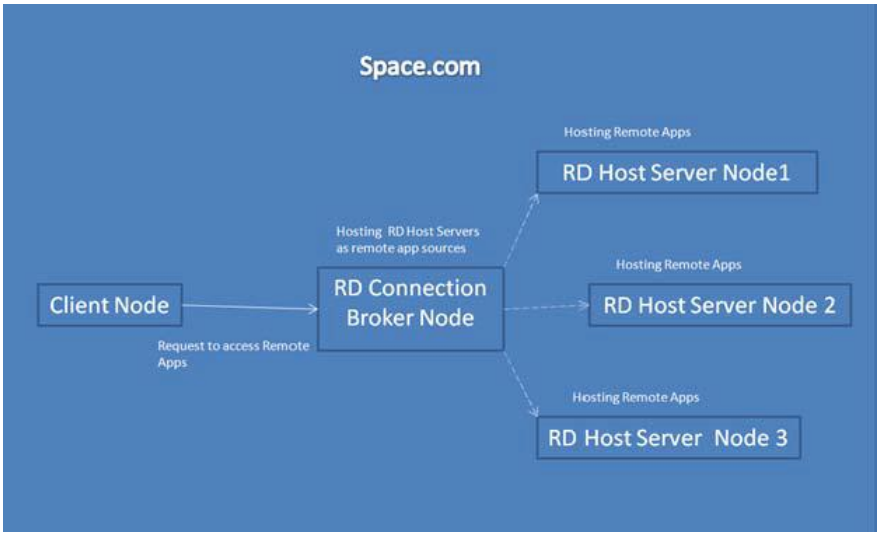
Note: To access RemoteApp programs through Remote Desktop-Web Access, the client computer must be running RDC 6.1. RDC 6.1 is included with Windows Server 2008 and higher operating systems, Windows Vista SP1 or later, and Windows XP SP 3. Use **About** dialog box to verify which version of RDC your system has.

- The client node and the Remote Desktop Session Host server should be able to communicate.

The following figure illustrates how RemoteApps configured at Remote Desktop Host Server node can be accessed:



The following figure illustrates how RemoteApps configured at multiple Remote Desktop Host Server nodes through Remote Desktop Connection Broker server can be accessed:



You need to perform the following procedures to deploy remote application programs through a remote desktop Web access:

- Install and configure the Remote Desktop Web access role service at an Remote Desktop Session Host server node installed with Window 2008 R2 or higher.
- Configure remote applications at a server node.
- Access the remote applications from a client node.

Install and Configure the Remote Desktop Web Access Role Service at a Remote Desktop Session Host Server Node

Remote Desktop Web Access service and Remote Desktop Host service (Remote Application) allow you to deploy a single Web site to run programs, access the full remote desktop, or connect remotely to the desktop of any computer in the internal network where you have the required permissions.

Log on to the Remote Desktop Session Host server node with local administrator privileges to install and configure the Remote Desktop web access role service at a remote Desktop Session Host server node.Use Network Level Authentication to provide a secure authentication method.

To install and configure the Remote Desktop web access role service at an Remote Desktop Session Host server node

1. Log on to the Remote Desktop Session Host server node with local administrator privileges.
2. Open the Server Manager and add roles and the required role services.
3. For Remote Desktop Services, select Remote Desktop Session Host and Remote Desktop Web Access. Add Required Role Services.
4. Specify the authentication method for the remote desktop session host.

Note: Click the Require Network Level Authentication option for a secure authentication method.

5. Add the required user group you want to allow access to the Remote Desktop Session Host server.
6. Install the Remote Desktop Web Access role service.

You will be prompted to restart your computer once the installation is complete.

Configure Remote Applications at Remote Desktop Session Host Server Node

After the Remote Desktop Web Access role is installed and configured, you can configure the remote applications at Remote Desktop Session Host server node.

Use the Server Manager window to select the programs you want to add to the RemoteApps list. The general workflow is as follows:

1. Open the Server Manager window.
2. Add the required remote programs.

Allow Application Access to Specific Users

After the remote applications are configured, you can define users or user groups who can access the applications at the client node, if required.

Select which users or user groups you want to provide access to the application. The general workflow is as follows:

1. Configure remote applications.

2. Select the required remote application.
3. Add users.

The added users or user groups can now access the application at the client node.

Access the Remote Applications from a Client Node

At the client node, you can access the configured remote applications in the following ways:

- Access a program on a Web site using Remote Desktop Web Access.

  At the client node, open **Internet Explorer** and connect to the Remote Desktop Web Access Web site using the following URL: https://technet.microsoft.com/en-us/library/cc731508.aspx.

  Log on with a domain account of the Remote Desktop Session Host server's administrators group.

Note: Any application launched from Remote Desktop Connection Broker appears as it were running on your local computer.

- Access a program on a Web site using Remote Desktop Web Access with Remote Desktop Connection Broker

  You can also access the configured remote applications from a client through another Remote Desktop Connection Broker Server node.

  Remote Desktop Connection Broker (RD Connection Broker), earlier known as Terminal Services Session Broker (TS Session Broker), provides access to remote applications and desktop connections. Accessing the remote applications and a desktop connection you can get a single, personalized, and aggregated view of RemoteApp programs, session-based desktops, and virtual desktops. Remote Desktop Connection Broker also supports load balancing and reconnection to existing sessions on virtual desktops, Remote Desktop sessions, and RemoteApp programs and aggregates RemoteApp sources from multiple Remote Desktop Session host (RD Session Host) servers that host different RemoteApp programs.

  Remote Desktop Connection Broker extends the TS Session Broker capabilities included in Windows Server 2008 and higher by creating a unified administrative experience for traditional session-based remote desktops and VM-based remote desktops. A VM-based remote desktop can be either a personal virtual desktop or part of a virtual desktop pool.

  In case of a personal virtual desktop, there is a one-to-one mapping of VMs. You are assigned a personal virtual desktop that can be personalized and customized. These changes are available to you each time you log on to your personal virtual desktop. For a virtual desktop pool, a single image is replicated across many VMs.Virtual desktop pool is to provide users with a virtual desktop that is dynamically assigned from a pool of identically configured virtual machines. As you connect to the shared virtual desktop pool, you are dynamically assigned a virtual desktop. You may not be assigned the same virtual desktop when you connect the next time. This means that any personalization and customization made by you are not saved. If you use a virtual desktop pool and want to save any customization, you can use roaming profiles and folder redirection.

Note: The improvements to the Remote Desktop Connection Broker role service are particularly useful while implementing a Virtual Desktop Infrastructure (VDI) or deploying session-based desktops or RemoteApp programs. These improvements further enhance the Remote Desktop Services.

  Add the Remote Desktop Connection Broker role service on a computer running Windows Server 2008 R2 or higher, and then use Remote Desktop Connection Manager to identify the RemoteApp programs and virtual desktops that are available through RemoteApp and Desktop Connection.

  You will need to prepare another node where Remote Desktop role service is installed and Remote Desktop Connection Broker service is enabled. For more information, refer to *"Install and Configure the Remote Desktop Web Access Role Service at a Remote Desktop Session Host Server Node"*.

To add Remote Desktop Session Host server in RemoteApp sources of Remote Desktop connection broker server

1. Open the Server Manager window.
2. Add the RemoteApp Source.
3. Add the Remote Desktop Session Host server name.
4. Add the Remote Desktop Connection Broker Server name in the TS Web Access Computers security group.

Note: Enable Network Discovery on the NLB Cluster nodes and RD Connection Broker node so that nodes can able to see each other and other network computers and devices and allows people on other network computers to see your computer.

5. Add the client node name in TS Web Access Computers security group on the Remote Desktop Connection Broker Server name.

To access RemoteApps configured at a Remote Desktop Session Host server from a client node

1. Connect to the Remote Desktop Web Access Web site.

   At the client node, open Internet Explorer and connect to https://technet.microsoft.com/en-us/library/cc731508.aspx.
2. Open the Enterprise Remote Access window.
3. Log on with a domain account of the local administrators group in all the nodes (Remote Desktop Connection Broker Server and Remote Desktop Session Host server).
4. Connect to the required Remote Desktop Connection Broker Server.

Note: Any application launched from the RD Connection Server Broker appears as it were running on your local computer. You can connect to the client machine through the VPN and access the RemoteApps.

The following table lists the applications which can be accessed as RemoteApp of the different System Platform nodes.

In Touch	Historian	Historian Client	Application Server	Common Utilities
Alarm DB Logger Manager	ITTagImporter	Trend	ArchestrA IDE	ArchestrA License Manager
Alarm DB Purge – Archive	Import InTouch Historical Data	Query	Object Viewer	Change Network Account
Alarm DB Restore	aahDBdump			Historian Configurator
Alarm Hot Backup Manager	ITHistImporter			License Utility

In Touch	Historian	Historian Client	Application Server	Common Utilities
Alarm Printer	aahHistorianCfg			SMC
Alarm Suite History Migration				
InTouch				
Window Maker				
Window Viewer				

Display the System Platform Nodes on a Multi-Monitor with a Remote Desktop

Prerequisites for the client node where the remote desktop is invoked

- Graphics card that supports multi-monitor and associated drivers
- Client Machine with an operating system (OS) that has RDP 7.0 or higher

After the client machine is prepared, you can display the system platform on a multi-monitor with a remote desktop.

To display the system platform nodes on a multi-monitor with a remote desktop

1. Ensure that the client machine is able to detect plugged-in secondary monitors.
2. Use the Control Panel Modify to configure the display settings. Use the "Extend these displays" option from the multiple displays list.

Verify the Display of System Platform Nodes on a Multi-Monitor with a Remote Desktop

Prerequisites for VMs running on the host Virtualization Server:

- VM nodes with OS that has RDP 7.0 or higher
- VM nodes running products such as InTouch

Note: The host virtualization server runs on Windows 2008 R2 or higher.

To verify system platform nodes display on a multi-monitor with a remote desktop, access any VM node installed with an IOM product from the client machine.

1. Open the Remote Desktop Connection window. Go to Run, and then enter "mstsc /admin". The Remote Desktop Connection window appears.

Note: Enter mstsc /console if you are using Windows XP.

2. Click Display, and select the Use all my monitors for the remote session check box and then click **Connect**. The VM node opens.

Note: If the client machine does not have RDP 7.0, this option will not be available to you.

3. Launch the IOM product and test the application. Drag and drop to move the application between the different monitors.

Use the Multi-Monitors as a Single Display

The multiple monitors configured on the client node, from where the remote desktop session is invoked, are used as independent displays when the remote session is used to connect to System Platform products installed on the VM nodes (with the exception of InTouch). In case of InTouch, the multi-monitors can be used either as independent displays or as a single display.

To use the multi-monitors as a single display, on an InTouch VM node, go to the path where win.ini exists and open win.ini. For example, the path is C:\User\<User_Name>\AppData\Local\Wonderware, where <User_Name> is the user login with which the remote session from the client connects to this VM node.

Enter the following parameters under the InTouch section and save it.

- MultiScreen – Enter "1" to enable the multi-monitor mode. Enter "0" to disable the multi-monitor mode.
- MultiScreenWidth – Enter the width of a single screen in pixels.
- MultiScreenHeight – Enter the height of a single screen in pixels. For example, if you want to show your InTouch application with a screen resolution of 2560 x 1024 on two horizontal monitors, enter the following:
  - "[InTouch]
  - MultiScreen=1
  - MultiScreenWidth=1280
  - MultiScreenHeight=1024"

Refer to the TechNote on multi-monitors for InTouch at https://gcsresource.invensys.com/support/kbcd/html/1/T001115.htm

Network Load Balancing

Network Load Balancing (NLB) distributes traffic across several servers by using the TCP/IP networking protocol. You can use NLB with a terminal server farm to scale the performance of a single terminal server by distributing sessions across multiple servers.

About the Network Load Balancing Feature

The NLB feature in Windows Server 2008 R2 and higher enhances the availability and scalability of Internet server applications such as those used on Web, FTP, firewall, proxy, virtual private network (VPN), and other mission-critical servers. A single computer running Windows Server provides a limited level of server reliability and scalable performance. However, by combining the resources of two or more computers running one of the products in Windows Server into a single virtual cluster, an NLB can deliver the reliability and performance that Web servers and other mission-critical servers need.

About Remote Desktop Connection Broker

Remote Desktop Connection Broker keeps track of user sessions in a load-balanced Remote Desktop Session Host server farm. The Remote Desktop Connection Broker database stores session information, (including the name of the Remote Desktop Session Host server where each session resides), the session state for each session, the session ID for each session; and the user name associated with each session. Remote Desktop Connection Broker uses this information to redirect a user who has an existing session to the Remote Desktop Session Host server where the user's session resides.

Remote Desktop Connection Broker is also used to provide users with access to RemoteApp and Desktop Connection. RemoteApp and Desktop Connection provide a customized view of RemoteApp programs and virtual desktops. Remote Desktop Connection Broker supports load balancing and reconnection to existing sessions on virtual desktops accessed by using RemoteApp and Desktop Connection. To configure the Remote Desktop Connection Broker server to support RemoteApp and Desktop Connection, use the Remote Desktop Connection Manager tool. For more information, see the Remote Desktop Connection Manager Help in Windows Server 2008 R2 and higher.

Remote Desktop Connection Broker that is used in an NLB setup is included in Windows Server® 2008 R2 and higher Standard, Windows Server 2008 R2 and higher Enterprise and Windows 2008 R2 and higher Datacenter.

The NLB feature is included in Windows Server 2008 R2 and higher. You do not require a license to use this feature.

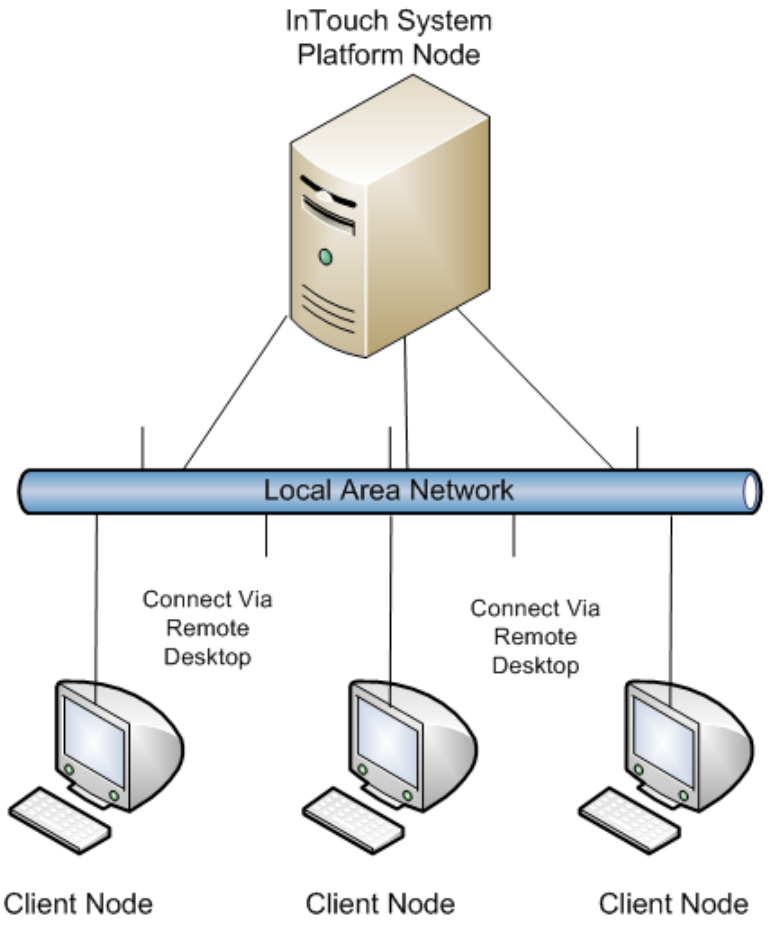
You need a Microsoft TS license for managing the remote desktop terminal server sessions.

About Managed InTouch Application with Network Load Balancing

The features provided by Remote Desktop are made available through the Remote Desktop Protocol (RDP). RDP is a presentation protocol that allows a Windows-based terminal (WBT), or other Windows-based clients, to communicate with a Windows-based Terminal Server. RDP is designed to provide remote display and input capabilities over network connections for Windows-based applications running on your Windows XP Professional desktop.

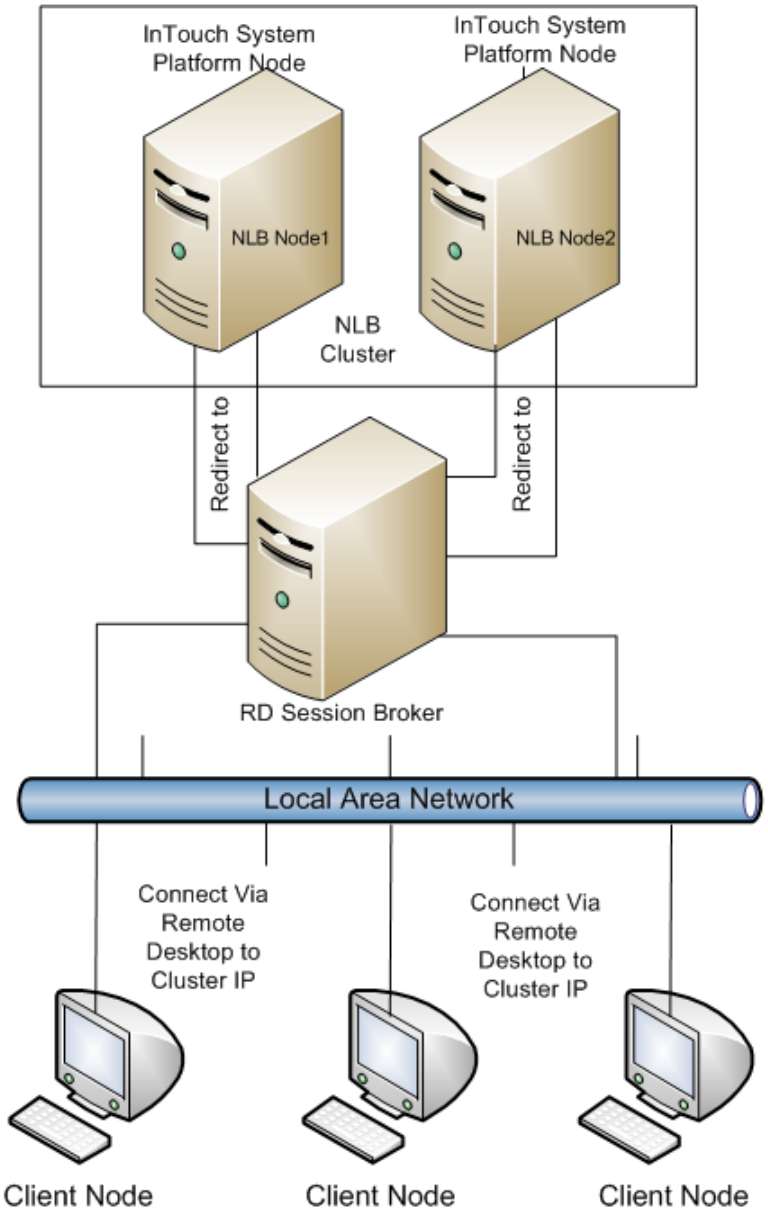
In this topology, clients can access the InTouch System Platform node via Remote Desktop. Whenever a new connection is requested to the InTouch System Platform Node, a new session is created. So all the traffic goes to the system platform node and degrades the performance of the InTouch node.

The following figure displays a topology without Network Load Balancing (NLB):



Network Load Balancing distributes IP traffic to multiple copies (or instances) of a TCP/IP service, such as a Web server, each running on a host within the cluster. Network Load Balancing transparently partitions the client requests among the hosts and enables the client to access the cluster using one or more "virtual" IP addresses. The cluster appears to be a single server that answers these client requests.

The following figure displays a topology with Networking Load Balancing:



Note: The Remote Desktop Connection Broker shown, as a separate node in the above topology, can be configured on one of the NLB cluster nodes itself.

You can leverage the load balancing for InTouch-managed applications.

To configure an NLB for managed InTouch application

1. Configure one VM or Physical machine with Application Server
2. On both the NLB cluster nodes, install InTouch TS with terminal server license.
3. Configure an NLB cluster as explained below.

4. On the Application Server node, develop managed InTouch application and deploy it on each of the NLB Cluster node.

Configuring an NLB for InTouch System Platform nodes, allows you to combine application servers to provide a level of scaling and availability that is not possible with an individual server.

NLB distributes incoming client requests to InTouch System Platform nodes among the servers in the cluster to more evenly balance the workload of each InTouch System Platform server and prevent overload on any InTouch System Platform server. To client computers, the NLB cluster appears as a single server that is highly scalable and fault tolerant.

Leveraging Network Load Balancing

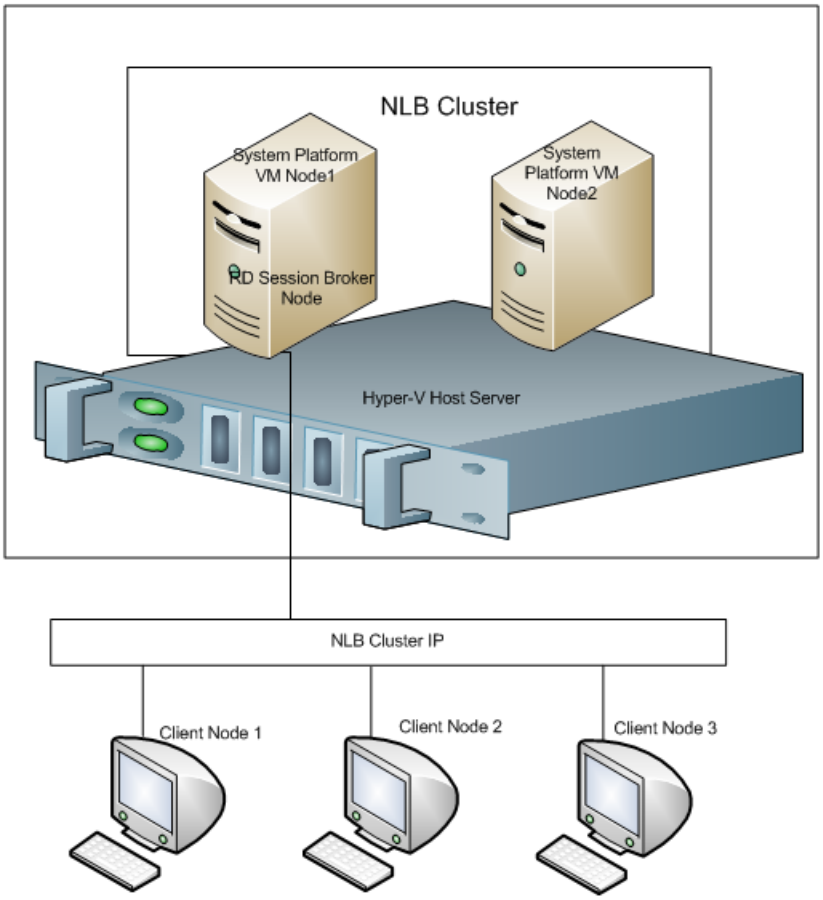
To setup an NLB:

1. Prepare two VM nodes that are remote desktop-enabled and have Windows Server 2008 R2 or higher.
2. Assign static IPs to both nodes.

Note: NLB disables Dynamic Host Configuration Protocol (DHCP) on each interface it configures, so the IP addresses must be static.

Example Topology 1: Configuring Remote Desktop

You can configure an NLB cluster configuring the Remote Desktop Connection Broker on one of the NLB cluster nodes.



To configure NLB with Topology 1

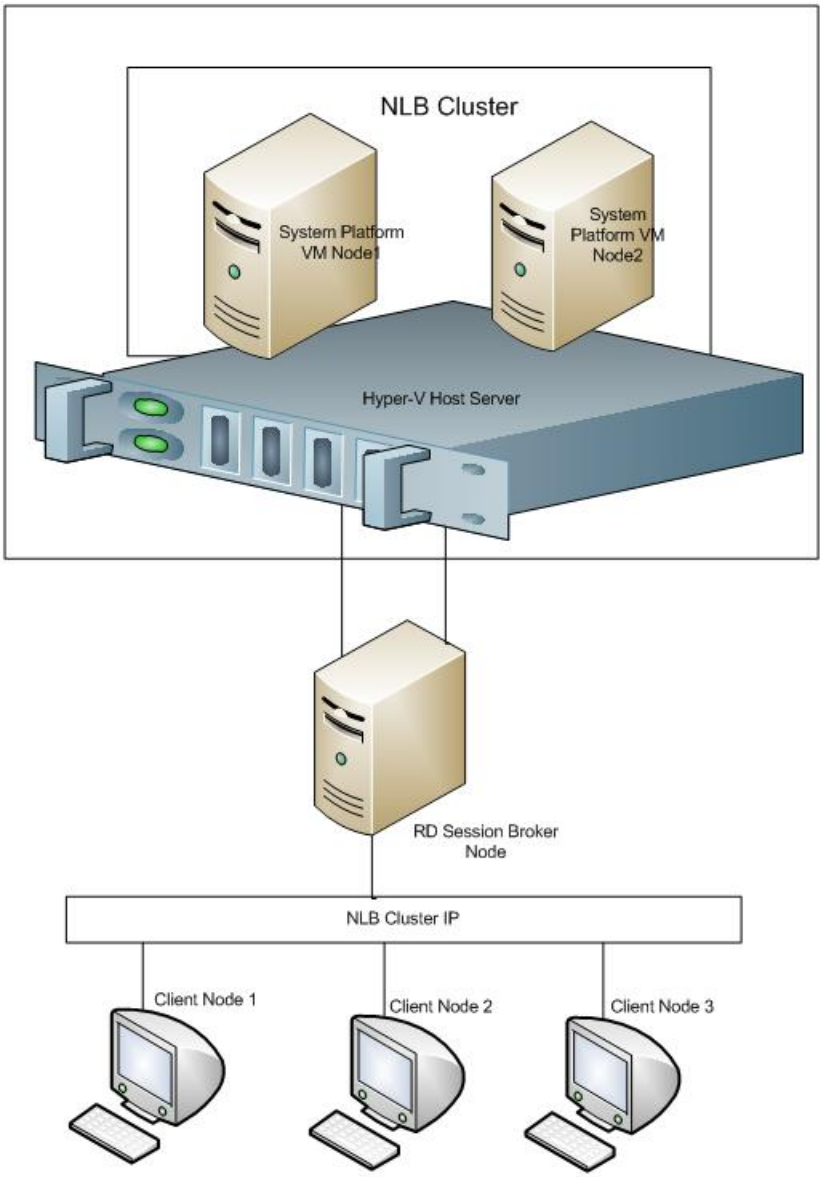
1. On each of the cluster nodes install Remote Desktop Services. For more information, refer to *"Install Remote Desktop Services"*.

Note: On the **Select Role Services** screen, select Remote Desktop Session Host and Remote Desktop Connection Broker on one of the Cluster Nodes to configure it as NLB Cluster node as well as RD connection broker node. On the other NLB Cluster node, select only Remote Desktop Session Host.

2. On each of the cluster nodes, install Network Load Balancing. For more information, refer to *"Install Network Load Balancing"*.
3. On the NLB cluster node which is configured as RD connection broker as well, add a Remote Desktop Session Host Server. For more information, refer to *"Add a Remote Desktop Session Host Server"*.
4. On each of the cluster nodes, create a Network Load Balancing Cluster. For more information, refer to *"Create a Network Load Balancing Cluster"*.
5. On each of the cluster nodes, configure Remote Desktop Connection Broker Settings. For more information, refer to *"Configure Remote Desktop Connection Broker Settings"*.

Example Topology 2: Configuring Remote Desktop Connection Broker on a Separate Node

Instead of configuring the Remote Desktop Connection Broker on one of the NLB cluster nodes, you can also configure the Remote Desktop Connection Broker on a separate node.



To configure NLB with Topology 2

On the NLB Cluster nodes, do the following:

1. Install Remote Desktop Services. For more information refer to *"Install Remote Desktop Services"*.

Note: In **Select Role Services** screen, select **Remote Desktop Session Host** on the NLB Cluster nodes.

2. Install Network Load Balancing. For more information, refer to *"Install Remote Desktop Services"*.
3. Create a Network Load Balancing Cluster. For more information, refer to *"Create a Network Load Balancing Cluster"*.
4. Configure remote desktop connection broker settings.For more information, refer to *"Configure Remote Desktop Connection Broker Settings"*.

On the Remote Desktop Connection Broker Node do the following:

1. Install Remote Desktop Services. For more information, refer to *"Install Remote Desktop Services"*.

Note: On the **Select Role Services** screen, select only Remote Desktop Connection Broker on the Remote Desktop Connection Broker Node.

2. Add a Remote Desktop Session Host Server. For more information, refer to *"Add a Remote Desktop Session Host Server"*.

Install Remote Desktop Services

Remote Desktop Services, earlier called Terminal Services, provides technologies that enable access to session-based desktops, VM-based desktops, or applications in the datacenter from both within a corporate network and the Internet. Remote Desktop Services enables a rich-fidelity desktop or application experience, and helps to securely connect remote users from managed or unmanaged devices.

Use the Server Manager to install Remote Desktop Services. Specify the option, "Do not require network level authentication."

There are two types of Windows Client Access Licenses from which to choose: device-based or user-based, also known as Windows Device CALs or Windows User CALs. This means you can choose to acquire a Windows CAL for every device (used by any user) accessing your servers, or you can choose to acquire a Windows CAL for every named user accessing your servers (from any device).

When you complete Remote Desktop Services installation, restart the node.

To install Remote Desktop Services

1. Open the Server Manager window.
2. Add the required role services. Select Remote Desktop Session Host and Remote Desktop Connection Broker.
3. Specify Do not require Network Level Authentication.
4. Select the applicable licensing option.

Note: There are two types of Windows Client Access Licenses from which to choose: device-based or user-based, also known as Windows Device CALs or Windows User CALs. This means you can choose to acquire a Windows CAL for every device (used by any user) accessing your servers, or you can choose to acquire a Windows CAL for every named user accessing your servers (from any device).

5. Confirm the details you entered, and install the services.

Install Network Load Balancing

You will need to install a Network Load Balancer (NLB) on the network adapter that you want to use for the Remote Desktop Protocol (RDP) connection.

Use the Server Manager to install the NLB. Select Network Load Balancing from the list of options.

To install NLB

1. Open the Server Manager window.
2. Add the Network Load Balancing feature and install it.

Add a Remote Desktop Session Host Server

A Remote Desktop Session host (RD Session Host) server hosts Windows-based programs or the full Windows desktop for Remote Desktop services client. You can connect to an Remote Desktop Session Host server to run programs, save files, and use network resources on this server. You can access an Remote Desktop Session Host server by using Remote Desktop Connection or RemoteApp.

You can add a Remote Desktop Session Host server to the connection broker computers' local group.

Use the Configuration option in the Server Manager to add an RD Session Host server. Select the required group to add to the Remote Desktop Session Host server.

To add an RD Session Host server

1. Open the Server Manager window.
2. Select the Session Broker Computers group to add to the Remote Desktop Session Host server.
3. Add the computer account for the Remote Desktop Session Host server.

Create a Network Load Balancing Cluster

To configure an NLB cluster, you need to configure the following parameters:

- Host parameters that are specific to each host in an NLB cluster.
- Cluster parameters that apply to an NLB cluster as a whole.
- Port rules

Note: You can also use the default port rules to create an NLB cluster.

Use the Network Load Balancing Manger to connect the required host to a new cluster.

- If you select the Unicast option, NLB instructs the driver that belongs to the cluster adapter to override the adapter's unique, built-in network address and change its MAC address to the cluster's MAC address. Nodes in the cluster can communicate with addresses outside the cluster subnet. However, no communication occurs between the nodes in the cluster subnet.
- If you select the Multicast option, both network adapter and cluster MAC addresses are enabled. Nodes within the cluster are able to communicate with each other within the cluster subnet, and also with addresses outside the subnet.

Add additional hosts as needed for load balancing. Then, add users to the Remote Desktop Users group to access the Network Load Balancing Cluster.

Note: Users can be local users and need not be domain users/administrators. If the users are local users they should be added on both the NLB cluster nodes with same user name and password.

To create an NLB cluster

1. Open the Network Load Balancing Manager window.
2. Enter the name of the host for the new cluster.
3. Select an interface for the new cluster, and create the cluster.

Note: The Priority value is the unique ID for each host. The host with the lowest numerical priority among the current members of the cluster handles the entire cluster's network traffic that is not covered by a port rule. You can override these priorities or provide load balancing for specific ranges of ports by specifying the rules on the Port rules tab of the Network Load Balancing Properties window.

- Add a cluster IPv4 static address, and enter the subnet mask.
- Enter the internet name of the new cluster.
- Select either the unicast or multicast option.
  - If you click the Unicast option, NLB instructs the driver that belongs to the cluster adapter to override the adapter's unique, built-in network address and change its MAC address to the cluster's MAC address. Nodes in the cluster can communicate with addresses outside the cluster subnet. However, no communication occurs between the nodes in the cluster subnet.
  - If you click the Multicast option, both network adapter and cluster MAC addresses are enabled. Nodes within the cluster are able to communicate with each other within the cluster subnet, and also with addresses outside the subnet.

4. Add another host to the cluster. Enter the name of node 2 and connect to it.
5. Enter a priority value for the host.

To add users to the Remote Desktop Users group to access Network Load Balancing Cluster

1. Specify which remote desktop versions you want to allow access to.
2. Select which user you want for which you want to allow access.

Note: The users can be local users and need not be domain users/administrators. If the users are local users they should be added on both the NLB cluster nodes with same user name and password.

Configure Remote Desktop Connection Broker Settings

Remote Desktop Connection Broker, earlier called Terminal Services Session Broker (TS Session Broker), is a role service that enables you to do the following:

- Reconnect to existing sessions in a load-balanced Remote Desktop Session Host server farm. You cannot connect a different Remote Desktop Session Host server with a disconnected session and start a new session
- Evenly distribute the session load among Remote Desktop Session Host servers in a load-balanced Remote Desktop Session Host server farm.
- Access virtual desktops hosted on Remote Desktop Virtualization host servers and RemoteApp programs hosted on Remote Desktop Session Host servers through RemoteApp and Desktop Connection.

To configure Remote Desktop connection broker settings, select the Farm member option and enter the name of the node where RD Connection Broker is installed. Then, enter the name of the farm that you want to join in the Remote Desktop Session Broker, select the option to participate in Connection Broker Load Balancing and assign weight for the server. You do this for both nodes.

Note: By assigning a relative weight value, you can distribute the load between more powerful and less powerful servers in the farm.

To add users to the Remote Desktop Users group to access Network Load Balancing Cluster

1. From Control Panel > System and Security > System Remote, select System Properties.
2. Under Remote Desktop, click the relevant option to specify the remote desktop versions you want to allow access to.
3. Select users to provide access to the system.

Note: The users can be local users and need not be domain users/administrators. If the users are local users they should be added on both the NLB cluster nodes with same user name and password.

Disconnect from and Connect to a Remote Desktop Session

If you disconnect from a session (whether intentionally or because of a network failure), the applications you were running will continue to run. When you reconnect, the Remote Desktop Connection Broker is queried to determine whether you had an existing session, and if so, on which Remote Desktop Session Host server. If there is an existing session, Remote Desktop Connection Broker redirects the client to the Remote Desktop Session Host server where the session exists.

With Remote Desktop Connection Broker Load Balancing, if you do not have an existing session and you connect to an Remote Desktop Session Host server in the load-balanced Remote Desktop Session Host server farm, you will be redirected to the Remote Desktop Session Host server with the fewest sessions. If you have an existing session and you reconnect, you will be redirected to the Remote Desktop Session Host server where your existing session resides. To distribute the session load between more powerful and less powerful servers in the farm, you can assign a relative server weight value to a server.

View Connected Sessions

You can use Remote Desktop Services Manager to view sessions connected to each node of the NLB cluster, and view information and monitor users and processes on Remote Desktop Session host (RD Session Host) servers. Open the Remote Desktop Services Manager window from any node of the NLB to view sessions connected to each node of the cluster.

To view sessions connected to each node of the cluster

1. On any node of NLB, open the Remote Desktop Services. From Administrative tools, open the Remote Desktop Services Manager.
2. Create a new group and enter the group name.

Note: The group name need not be the same as the cluster name.

3. Add the required computers to the group.

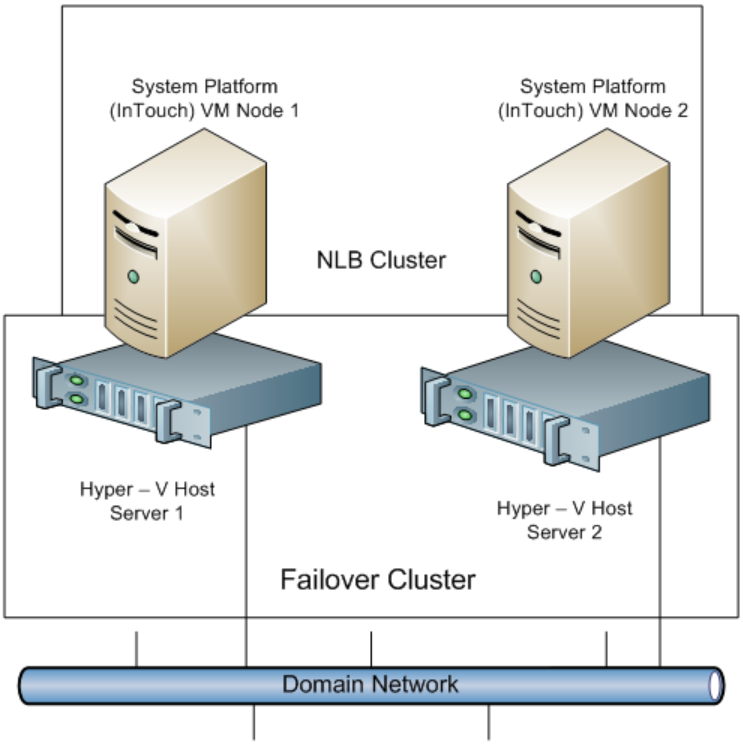
You can now select the newly-created group name and view the sessions connected to each node of the cluster.

Configure Network Load Balancing Cluster on Microsoft Failover Cluster

Windows Server® 2008 R2 and higher provide two clustering technologies: failover clusters and NLB. Failover clusters primarily provide high availability; NLB provides scalability and, at the same time, helps increase availability of Web-based services.

By using a failover cluster, you can ensure that there is nearly constant access to important server-based resources. A failover cluster is a set of independent computers that work together to increase the availability of services and applications. The clustered servers (called nodes) are connected by physical cables and by software. If one of the nodes fails, another node begins to provide service through a process known as failover.

NLB that is configured in a failover cluster offers high performance in environments in which each request from a client is stateless, and there is no in-memory application state to maintain



To configure NLB cluster on Microsoft failover cluster

1. Set up Microsoft Failover Cluster out of two Hyper-V host servers.
2. Configure two VM nodes one on each Hyper-V host server.
3. Configure the NLB cluster out of two VM nodes hosted by each Hyper-V host server following the procedures in Leveraging NLB by Configuring Remote Desktop Session Broker on a NLB Cluster Node explained in topology 1. For more information, refer to *"Example Topology 1: Configuring Remote Desktop"*.

Understanding the Behavior of NLB Cluster in Microsoft Failover Cluster

1. During a live migration of one of the NLB cluster nodes, there are no disruptions in the active sessions connected to the cluster node. The Reconnect window will not appear on the NLB cluster node as there is no disruption of the active session. After the live migration is complete all sessions connected to the NLB cluster node are retained.
2. During a quick migration, when one of the Hyper-V host servers (Microsoft Failover Cluster Node) is shut down or switched off and the failover is completed, all active sessions on the NLB cluster node hosted by the Microsoft failover cluster node are automatically connected and all sessions on the NLB cluster node are retained.

Observations while using NLB for Managed InTouch System Platform node:

- The NLB feature is qualified for InTouch managed application. InTouch TSE license is required on each of the NLB cluster nodes.

- Local InTouch Tag Alarms are local to the session. Local InTouch Tag alarms updated in a session remain local to that session only.
- ArchestrA Alarms are common across all sessions. ArchestrA Alarms updated in one of the sessions get reflected across all the sessions.
- For IO tags poking in one session, the data reflects across all the sessions. However, while poking local InTouch tags, data does not get updated across all sessions since it is local to the session.
- When you lose the NLB cluster node with the active sessions, all the active sessions on the NLB cluster node closes. To retain all the active sessions, configure the NLB Cluster in a Microsoft Failover Cluster in a Hyper-V environment. The NLB cluster nodes are VM nodes hosted by Hyper-V host servers and Hyper-V host. For more information, refer to *"Configure Network Load Balancing Cluster on Microsoft Failover Cluster"*.

Hardware Licenses in a Virtualized Environment

Windows Server 2008 R2 does not support hardware licenses in the Hyper-V virtualized environment. You may want to verify support under later server editions.

Hardware licensing using AnywhereUSB are supported. For more information, visit http://www.digi.com/products/usb-and-serial-connectivity/usb-over-ip-hubs/anywhereusb

CHAPTER 9

Planning Storage in a Virtualized Environment

In virtual environments, storage options require special consideration. This chapter introduces available and recommended options and provides information to guide decisions.

As a rule of thumb, instead of simply directing the traffic of four VMs to one hard disk, use one spindle (drive arm) per VM. This guideline will help you to avoid the most common resource problem.

Shared storage is one of the least-understood components because, outside of traditional large data centers, it is rarely used. If a common file storage environment is needed, the solution is often to install extra hard drives in a server, and sharing a drive or a folder.

The major advantage of a shared storage solution is high availability. While most hypervisors can utilize local storage on a server to run virtual machines, high availability and portability functions are lost if data is stored on a local machine instead of on a shared storage solution.

In This Chapter

Choosing Connectivity

When establishing shared storage to take advantage of high availability, the first major choice involves connectivity to the device. While shared storage is more accurately referred to as network storage, the "network" component does not necessarily refer to ethernet.

The first type of network shared storage used for sared storage is Fibre Channel. It uses a proprietary protocol (as opposed to TCP/IP) transported over a fiber-optic cable. Like traditional Ethernet, Fibre Channel requires a specialized adapter in the host, along with switches to aggregate connectivity.

The second major technology choice is the more familiar Ethernet. Using Ethernet involves standard network interface cards located in the server, standard Ethernet switches, and copper cabling for up to 1GB/s speeds.

Fibre Channel

For many years, there were three primary reasons for choosing Fibre Channel.

- Fibre Channel was typically always faster than any competing Ethernet protocol. The most common top end speed of Fibre Channel today is 8 GB/s, with 16 GB/s becoming more common.
- Because Fibre Channel utilizes a proprietary protocol that does not contain the overhead of TCP/IP, the communications latency is extremely low, typically < 1 ms.
- Until recently, almost all quality storage arrays implemented Fibre Channel as their interface of choice.

Ethernet

While Fibre Channel still enjoys some advantages over Ethernet, modern implementations of Ethernet protocols are eliminating these advantages.

- The speed advantage enjoyed by Fibre Channel over Ethernet has effectively been nullified. Ethernet at 10GB/s has been in common use since 2010, and 40GB and 100GB speeds will soon be available.
- Specialized Ethernet switches can help you obtain ultra-low latency connections within your Ethernet fabric.
- You can usually connect to all but the most high-end network storage devices with Ethernet protocols.
- Specialized Ethernet cards offer TCP Offload Engine (TOE) technology, which offloads all processing of the TCP/IP stack to the netwok controller. This becomes a major advantage when using gigabit and 10 gigabit Ethernet speeds, in which processing the network stack becomes a significant task.

Today, it is not typical to see a new installation utilize Fibre Channel, and is generally not recommended.

Instead, most new small and mid-sized environments are choosing Ethernet protocols. Of special note is the need to separate the storage and computer networks, a concept discussed in further detail later.

Choosing Protocols

If you have chosen Ethernet connectivity, you must next select the protocol, either NFS or iSCSI.

Another popular term used when discussing NFS vs. ISCSI is File vs. Block. Understanding this difference will help you understand the difference in the protocols. The following table lists the File and Block storage options.

Block Storage Options (SAN)	File-Based Storage Options (NAS)
iSCSI	NFS
Fibre Channel	CIFS
AoE (ATA over Ethernet)	

Both protocols are capable of acceptable performance in small- to medium-sized environments. The primary differences are reflected in setup and scalability.

Advantages: Protocol Setup and Scalability

Although there are a few more steps in setting up an iSCSI datastore, this is usually a one-time activity that only takes a few extra minutes more than configuring an NFS datastore.

Another major difference between protocols regards their scalability. While you can connect multiple network cables to an NFS controller, only one of them can be used at a time between a computer host and the storage device. This is an inherent limitation in the protocol. The primary purpose of using multiple connections to an NFS array is for redundancy, not throughput. iSCSI, on the other hand, can use as many network connections as you configure between the computer host and the network device. It is also a protocol-specific ability that allows for this functionality. The real question is determining if this difference actually matters. A single host rarely saturates a typical 1 GB/s link between the host and storage unit. Only in extreme cases (for example, when the user is running a specialized test) does saturation of the 1 GB/s link occur.

Pros and Cons: NFS vs SAN Protocols

One final consideration is the optimizations present in the VMFS file system utilized by VMWare when implementing block storage. While NFS was designed from the ground up as a multi-user protocol, it was never designed with the intent of handling really large virtual machine disk files. VMFS, on the other hand, has always been and will continue to be designed to handle extremely large files with simultaneous multi-host access.

NFS Protocol

Pros:

- Usually less expensive
- Simpler configuration
- Simpler management

Cons:

- Higher CPU overhead

SAN Protocol

Pros:

- Higher performance
- Ability to offload protocol overhead to hardware components
- Allows hypervisor to use specialized file system

Cons:

- Usually more expensive
- More complex configuration

Initializing the NFS Protocol

For NFS, the following tasks are involved in setting up a datastore:

1. Create an NFS export on your storage device.
2. Use vSphere to create a special network port over which you will connect to your NFS datastore.
3. Connect vSphere to the NFS export and create a new datastore. Once connected, the user can immediately begin to store virtual machines on the newly created datastore.

Initializing the iSCISI Protocol

For iSCSI, there are a few additional steps.

1. First, create an LUN on your storage device.

2. From vSphere create a special network connection for ISCSI data.
3. Next, add your storage device as an ISCSI target.
4. After adding the device, perform a rescan of ISCSI targets. At this point, you should see the LUN created.
5. Select the LUN and create a new datastore.
6. Once the datastore is created, format the datastore with VMFS.

Choosing Features

Once connectivity methods and protocols have been determined, select a system with components and features that match your needs.

Controllers

All disk systems have at least one disk controller through which read/write requests are passed. The following sections detail the types, attributes, and features of controllers to help you to select the configuration that best suits your needs.

Controller Attributes

The primary difference between storage arrays, and the type of storage used in standard desktops and servers, is the quality and quantity of controllers. In a storage array, the controllers are designed for high throughput, as when many (10+) computer hosts access data simultaneously. For this reason, these specialized controllers usually contain more RAM and a faster chipset than what is found in a typical Redundant Array of Independent Disks (RAID) controller in a standalone server. Arrays installed in a production environments must have a redundant configuration.

There are two reasons for this.

- When a controller fails, a backup controller takes over its duties without any system interruption. Depending on the architecture, this may cause a decrease in performance:
  - In an "Active-Active" array, both controllers are processing the workload. If one controller is lost, performance is cut in half.
  - In an "Active-Passive" array, only one controller is active at a time. In the event of a failure, no performance difference should be experienced. Depending on your performance requirements, this may not be much of a distinguishing factor.
- If using multiple controllers, ensuring that they are hot-swappable enables their replacement at the time of failure with no interruption in availability.

Tip: Though not necessary, the ability to upgrade firmware by failing back and forth between controllers can be a useful addition.

Controller NVRAM and Cache

Another feature to consider carefully is the presence of NVRAM, or a battery-backed cache, on the controllers. These features store data as it is being written, so that if power is cut while the array is writing data, the write can be completed when power is restored.

Depending on numerous different factors, this downtime can be as long as a week or two before potential corruption becomes an issue. This is a feature that quality arrays will implement as a matter of standard configuration. If this is offered as an option rather than being automatically included, it is a red flag that this array may not be suitable for a demanding environment. While downtime is bad, corrupted data is unacceptable in a manufacturing environment.

Network Accessibility

When selecting a controller, consider the number of network ports offered. At a minimum, there should be two network ports for data and one network port for a maintenance interface. Having a separate port for maintenance allows routing to your production network for configuration and maintenance while leaving the actual storage data on separate ports on a separate network.

Expansion

To prepare for future expansion, consider whether the unit has additional "shelves" or "enclosures." Typically, these units take the form of a 2U device with nothing but hard drives and an interconnect. These enclosures are connected via an external SAS cable to the controllers in the first enclosure. On each of these shelves, there are usually IN and OUT connections, allowing more shelves to be added in a daisy chain fashion. If implemented properly, these new enclosures can be added without any disruption to the running array. It is not atypical for a modest array to support as many as 48 to 96 hard drives on a single set of controllers.

Online Maintenance

Just as modern DCS systems are engineered to run for years without downtime, a quality array should be similarly designed.

Downtime can typically have two sources:

- Component malfunction. This is mitigated via component-level redundancy and careful design of parts to maximize Mean Time Between Failure (MTBF).
- Upgrades and modifications. A quality array will allow for the creation, maintenance, and resizing of disk volumes with no downtime on the system or the volume. As mentioned earlier, a quality array will also allow for firmware updates with no downtime.

Software Features

The software features available in the modern storage array are extensive. Some of the major features include compression, deduplication, snapshots, and replication.

Compression works in a similar manner as creating ZIP files on your computer and unzipping the file when you need to access the files. Compression and decompression are performed by the array in the background as data are stored and retrieved.

Deduplication involves a system analyzing each block of data being stored and determining if an exact copy already exists. If a copy exists, then the system simply stores a pointer to the existing block instead of storing the data a second time. Typical deduplication ratios in a relatively homogenous environment (i.e. lots of Windows installs) are approximately 5x to 10x. With deduplication, what previously took 5 TB to store now only takes 1 TB.

A snapshot is a method of backup that takes place on the array itself rather than by software installed on the machine. While snapshots are efficient, some users may find that they are slightly more difficult to work with as opposed to a typical virtual machine backup software package.

Some arrays provide the ability to perform near real-time replication. Though expensive, this is a excellent method for ensuring business continuity in the event of a disaster taking out the primary array. A major drawback is that a corrupted file may be replicated, creating two corrupted files or databases. Some replication methods combat this by allowing a rollback to a previous state.

Performance

The final and most important item to consider when purchasing an array is its performance.

Disk performance takes two major forms; Input/Output Operations Per Second (IOPS) and throughput. IOPS is measured in total read / write operations per second. Throughput is typically measured in MB/second.

While both are important measures, the primary limiting factor in most environments is IOPS. An I/O operation occurs whenever data is written to, or read from, the disk.

There are three major factors under the user's control that influence IOPS.

The first is disk speed. The faster the speed of the underlying disk, the more IO operations a particular disk can support. Although seek time and rotational latency are factors, we will focus on disk speed.

Second, the total number of drives - commonly referred to as spindles - in a volume (aggregated set of disks with a particular capacity) can influence IOPS. The more spindles in a volume, the more IOPS it can support. Using multiple slower disks can sometimes provide better performance than fewer fast disks.

Finally, the RAID configuration of the volume can have a substantial effect on the IOPS performance. The easiest way to see the effect of each is to calculate the average IOPS for a particular disk arrangement while adjusting different parameters to see the effect.

RAID Impact on System Performance

When writing to a RAID array, the system must not only split the write across multiple disks, but also calculate parity bits (in all but mirrored (RAID 1), striped (RAID 0), or mirrored + striped configurations). The more parity bits required, the more severe the hit on write performance.

This can have a dramatic effect on performance; on a typical Application Object Server on a System Platform environment, disk access consists almost entirely of write operations. Though historians also typically have a high percentage of write operations, keep in mind the number of clients that may be running trends at the same time.

In order to measure read and write operations, use Perfmon, a tool included with all Microsoft operating systems. As a best practice, run these metrics at one minute intervals for 24 hours. This frequency should account for daily activities and for backups.

In a real-world case study across multiple Application Object servers, an average of 130 write operations per second - accounting for nearly 100% of disk activity - was observed. These writes operations, in turn, were almost exclusively the result of the application engines writing checkpoint files and historical store/forward data to protect against engine failures. As a side note, the system originally distributed its entire load across three machines instead of five. When the system only had three machines, the checkpointing was slowed to once every five seconds because the machines could not keep up.

This was initially speculated to represent insufficient RAM and CPU, but a closer look at disk statistics revealed that the bottleneck was the disk subsystem. In response, a pair of 2.5" 10K RPM drives in a RAID 1 configuration was installed in each machine. According to an online calculator, this configuration was capable of supporting 140 IOPS. A quick check of the math yields (130 IOPS * 5 new machines)/(3 old machines) = 217 IOPS/old machine.

In summary, performance - rather than capacity - is the primary concern when planning an array for your environment. If a system cannot meet I/O demand, capacity is not a concern. For this reason, you will typically see that the highest quality arrays provide capacity at or under a terabyte. This is because manufacturers realize that a user will typically run out of IOPS before GBs.

SSD Performance

When shopping for arrays, you will find those using solid state disks (SSDs). Though these devices were initially created and marketed to fight vibration and shock issues in industrial PCs, this has become a secondary concern. The primary reason for inclusion of SSDs in newer arrays is their superior performance vs. traditional, mechanical hard drives.

A glance at available online hard drive benchmarks like those available at http://www.harddrivebenchmark.net/ shows that even low-end, consumer grade SSDs far outperform expensive, high-RPM, mechanical drives.

Just as critical, however, is a disk's lifetime. Users are accustomed to hard drives lasting at least five years, and it is not unusual for a drive to last even longer. However, SSDs have a shorter useful life, and typically fail suddenly and catastrophically.

Many of the more traditional vendors utilize SSDs as a tier of storage. In this scenario, the array watches the blocks of data that are most active in terms of read/write activity. The most active blocks are transferred by the array from the slowest mechanical drives up to the faster mechanical drives, and finally to a layer of SSDs providing the best performance. The purpose of this approach is economic, which allows using much more expensive devices without making the overall unit unaffordable.

A second approach involves using SSDs as a conduit, or sort of cache, to slower disks. In this configuration, all write operations are performed on the SSDs first. Once the resources are are available, this data is transferred to slower, cheaper disks in the array.

A third and riskier configuration involves packing the entire array with consumer grade SSDs, using sophisticated software to perform inline compression, deduplication, and other advanced techniques to reduce the number of writes required.

Networking

Network storage refers to a physically isolated network that manages all storage traffic communication. The storage network should be a dedicated system of cables and switches, for many of the same reasons needed to isolate the PLC network from general client-server traffic. When designing your network, plan for redundant switches. Losing your storage backend will typically allow machines to run for about 20 or 30 seconds in a frozen state until they fail.

A properly configured storage network involves a single controller that connects to multiple switches with a minimum of four network connections. This ensures that the system can continue operation in the event of a controller and switch failing simultaneously.

Cost Factors

A good general rule when budgeting for a virtualization project is that your main server(s) should account for 50% of the total hardware budget, and storage for the other 50% of the total hardware budget (including 4 switches, 2 for virtual machine traffic, and 2 for storage traffic).

Note that some vendors include only base functionality in a starting price, and allow you to select features, like those discussed earlier, in an a la carte fashion. It can sometimes double or triple the starting price of your unit.

Finally, pay close attention to warranty costs. While higher-end units will typically include three to five years of base warranty, maintenance costs after warranty expiration can become extremely expensive. This can be a driver in the refresh cycle for a typical IT organization, since while old hardware may be functioning well, the costs of maintaining the warranty for old hardware can sometimes make it more affordable to purchase new hardware. Work closely with your budget managers on this detail.

Conclusions

As regards the cost of a virtualization environment, note the following:

- A three physical host system can easily support 24-30 servers.
- An average physical server should cost approximately $5K if properly specified.
- The acquisition cost for these servers, ignoring the additional cost of networking, would be approximately $120K.

Contrasted with a $50K acquisition cost for a virtualized three-host system with storage, a virtualized system with high quality storage is much less expensive. The economic advantages are significant once you pass 8-10 servers.

Acknowledgements

The preceding information is provided with express permission, and with content created by, Avid Solutions, and Andy Robinson with Avid Solutions.

The original content was also authored by A. Robinson and R. Kambach as part of a white paper for the Developer Network resource.

CHAPTER 10

Implementing Backup Strategies in a Virtualized Environment

A virtual server backup is a copy of data stored on a virtual server to prevent data loss. There are two fundamental types of backups:

- Guest-level backup
- Host-level backup

Backup and Restore Strategies

There are a number of backup and restore strategies in both virtualized and non-virtualized environments. For the guest level, the virtual machines (VMs) are backed up as if they were physical servers. Although this strategy is among the simplest, it also has several drawbacks. You need to install backup software in each virtual machine (VM) to be copied in Guest Operating Systems, and maintain separate backup jobs (or even multiple backup jobs) per VM. This approach requires additional resources to execute the backups, and can affect the performance of the virtual machines. This backup type is not suitable for restoration in the event of a disaster or granular restores within applications, such as databases or email.

Another backup strategy is to use a host-level backup. In this approach, back up the entire VM at one time. However, it can be as granular as your backup and restore application allows it to be.

We recommend using the host-level backup. It creates a complete disaster recovery image of the virtual server, which can be restored directly into the source virtual infrastructure.

Checkpointing Method

In this method you can take point-in-time checkpoints (snapshots) of the entire VM. We recommend this method as it ensures data consistency and allows for a fast and complete restore of the entire VM. One of the few disadvantages in this method is that you need to restore the entire checkpoint even if a file is lost or corrupt.

In a Microsoft virtualized environment, you can take and restore checkpoints using either System Center Virtual Machine Manager (VMM) or Microsoft® Hyper-V Manager. The following sections describe how to implement backup strategies using SCVMM.

In This Chapter

Taking Checkpoints Using SCVMM

By creating a checkpoint, you can save all contents of a virtual machine hard disk. You can reset your machine to a previous configuration if required, without having to uninstall programs or reinstall operating systems. This also helps you test applications across various configurations.

You can checkpoint one or more VMs both in the online and offline modes. However, you can checkpoint a VM only when it is deployed on a host.

Important: Typically, there are dependencies among nodes. Taking a checkpoint of a VM and restoring it later could negatively impact those dependencies. For more information, refer to *"Checkpoints of System Platform Products - Observations and Recommendations"*.

Take a Checkpoint of an Offline VM

It is recommended that you shut down the virtual machine before creating a checkpoint. You can also create a checkpoint of the virtual machine offline. This stops the machine temporarily while the checkpoint is created. Turning off the virtual machine prevents loss of data while the conversion takes place. The general workflow for offline VMs is as follows:

To take a checkpoint of an offline VM

1. Open the System Center Virtual Machine Manager (SCVMM).
2. Select the VM that you want to checkpoint.
3. Shut down the selected VM you selected.
4. Make a new checkpoint.
5. Verify the checkpoint.

Take a Checkpoint of an Online VM

You can create checkpoints of a virtual machine while it is running. However, creating a checkpoint in online mode requires special application support.

Important: To avoid losing any data, do not make any configuration changes to the machine while creating a checkpoint. For more information, refer to *"Checkpoints of System Platform Products - Observations and Recommendations"*.

If you create a checkpoint after making configuration changes when the VM is online there may be issues when you restore the VM to that checkpoint.

For example, if you create a checkpoint for an online IOM Historian Product VM state and then try to restore it, the history block that is created shows a discrepancy in the start and end time and the following errors are displayed.

Warning: aahIndexSvc Attempted to create history block ending in the future

Error: aahIndexSvc ERROR: Invalid file format

To avoid such errors, stop the Historian VM before creating a checkpoint in the online mode. The general workflow for online VMs is as follows:

To take a checkpoint of an online VM

1. Open the System Center Virtual Machine Manager (SCVMM).
2. Select the VM that you want to checkpoint.
3. Make a new checkpoint.
4. Verify the checkpoint.

Restore Checkpoints

You can revert a virtual machine to a previous state by restoring it to the required checkpoint. When you restore a virtual machine to a checkpoint, VMM stops the virtual machine and the files on the virtual machine are restored to their previous state.

Important: If the virtual machine has been in use since the checkpoint was created, take a backup of the data files before you restore the virtual machine to avoid loss of any data.

Restore Checkpoints from a Virtual System Platform Backup

You can restore a VM to its previous state by using checkpoints. You can restore checkpoints of VMs both in the online and offline modes.

Restore a Checkpoint of an Offline VM

When you restore a VM to a checkpoint taken of an offline VM, there should not be any loss of data. When checkpoints are taken from a VM that is offline, the machine temporarily stops, minimizing data loss during the conversion process. The general workflow for an offline VM is as follows:

To restore a checkpoint of an offline VM

1. Open the System Center Virtual Machine Manager (SCVMM).
2. Select the offline VM for which you want to restore a checkpoint.
3. Restore the checkpoint.

Restore a Checkpoint of an Online VM

You can restore a VM to a checkpoint that was taken when the machine was online. Restoring a VM to a checkpoint taken while online may lead to loss of data. However, if no changes to the configuration were made while creating the checkpoint, there should not be any data loss. The general workflow for an online VM is as follows:

To restore a checkpoint of an online VM

1. Open the System Center Virtual Machine Manager (SCVMM).
2. Select the VM for which you want to restore a checkpoint.
3. Restore the checkpoint.

Take and Restore Checkpoints of Products with No Dependencies

You can create and restore checkpoints of IOM products that do not have dependencies. When you restore the VM to a checkpoint, data is restored up to the point at which you took the checkpoint. Data related to all changes made after the checkpoint was taken is not captured and will not be restored.

For example, on an Application Server node, two User Defined Objects (UDOs) are created at different points in time and checkpoints taken at each point. If you restore your VM to the first checkpoint, it will be restored to the state where only the first UDO was created. The second UDO created will not be backed up or restored in your system. The general workflow is as follows:

To take and restore checkpoints of products with no dependencies

1. Open the System Center Virtual Machine Manager (SCVMM).
2. Select the VM for which you want to create and restore a checkpoints.
3. Connect to the virtual machine.
4. In **Application Server** under **Platform**, **Engine**, and **Area**, create UDO1.
5. Use Virtual Machine Manager to select the VM.
6. Make a new checkpoint.

7. Connect to the virtual machine, if not already connected.
8. In **Application Server** under **Platform**, **Engine**, and **Area**, create UDO2.
9. Restore the VM.

Checkpoints of System Platform Products - Observations and Recommendations

The following are some of the observations and recommendations to take and restore checkpoints of System Platform Products.

- Take checkpoints of System Platform Products only when there are no configuration changes. For example, some of the scenarios where the checkpoints should not be taken are as follows:

System Platform Product	Configuration Changes
Application Server	deploy, migrate, import, export, check-in, check-out
Historian	import, export, create history block

- You must be aware of the consequences and make decisions when taking and restoring checkpoints of System Platform Products that have dependencies. If the configuration of a System Platform node has a dependency on the configuration of another System Platform node, it is recommended to take and restore checkpoints on such dependent nodes together. For more information, refer to "Recommendations".

Take and Restore Checkpoints (Snapshots) in the Offline Mode

It is recommended that you take checkpoints of System Platform Products when the VMs hosting them are in the offline mode. Turn off the System Platform Product VM before taking a checkpoint.

Restoring checkpoints of VMs in the offline mode result in smooth functioning of the System Platform Products after the restoration. After restoring a checkpoint, start the VM, and then start the System Platform Product hosted in the VM.

Take and Restore Checkpoints (Snapshots) in the Online Mode

While the VM is in the online mode, the System Platform Product hosted on the VM functions in one of the following ways:

- **Scenario 1**: If the System Platform Product is not running on an online VM, it functions smoothly after the restoration of checkpoints.
- **Scenario 2**: If a checkpoint is taken while the System Platform Product is running on an online VM and there are no configuration changes in progress, the System Platform Product performs normally. However, when checkpoints are restored, there would be issues with the System Platform Product running on that VM. Some of the issues are explained in the following table.

Recommendations

Observation	Recommendations
Historian	

Observation	Recommendations
	• Do not take checkpoints while a history block change is in progress. Restoring such a checkpoint leads to unpredictable behavior of the product. • In case of communication issues between the Historian and dependent System Platform Products, restart the VMs. • If a checkpoint is taken before configuring Application Server to historize attributes, re-deploy the platform after the Historian is restored.
Issue 1: When you restore a checkpoint of the Historian node taken while the Historian was running and the block change was in progress, there is a conflict in the start and end time in the history block. The following errors and warnings are logged: Warning: aahIndexSvc Attempted to create history block ending in the future. Error: aahIndexSvc ERROR: Invalid file format.	As a recovery step of Issue 1, shut down and disable the Historian, and then start and enable it.
Issue 2: While creating a checkpoint there may be an action in progress resulting from an event. The incomplete action is not saved when you restore such a checkpoint.	
Application Server	
If checkpoints are restored on either GR node or remote IDE node, the configurations might go out of synchronization.	Perform galaxy object component synchronization (GOCS) after opening the IDE on the remote node.
Data Acquisition Server (DAS)	

Observation	Recommendations
If checkpoints are restored on DAS, there may be connectivity and configuration mismatch issues for the dependent System Platform Products.	Deactivate, and then activate the DAS with appropriate configuration file. If it does not resolve the connectivity issues, restart the dependent System Platform Product VMs.
InTouch	
If the AlarmDBLogger is configured on the local SQL Server, restoring checkpoints results in expected data loss.	If the alarm data is critical, configure the AlarmDBLogger on a remote SQL Server.
Information Server (WIS)	
If checkpoints are restored on WIS, there may be connectivity issues for the dependent System Platform Products.	Log off and re-launch the WIS browser.
Historian Client	
If checkpoints are restored on Historian Client, there may be connectivity issues to access the Historian.	Log off the server connection and log on to the Historian Client Applications.

APPENDIX A

Glossary

Application Engine (AppEngine)
A scan-based engine that hosts and executes the run-time logic contained within Automation Objects.

application object
An Automation Object that represents some element of your production environment. This can include things like:

An automation process component. For example, a thermocouple, pump, motor, valve, reactor, or tank

An associated application component. For example, function block, PID loop, sequential function chart, ladder logic program, batch phase, or SPC data sheet

Application Server
It is the supervisory control platform. Application Server uses our existing products, such as InTouch for visualization, Historian for data storage, and the device integration product line like a Operations Integration Server (OI Server) for device communications.

An Application Server can be distributed across multiple computers as part of a single Galaxy namespace.

ArchestrA
The distributed architecture for supervisory control and manufacturing information systems. It is an open and extensible technology based on a distributed, object-based design.

child partition
Child partitions are made by the hypervisor in response to a request from the parent partition. There are a couple of key differences between a child partition and a parent/root partition. Child partitions are unable to create new partitions. Child partitions do not have direct access to devices (any attempt to interact with hardware directly is routed to the parent partition). Child partitions do not have direct access to memory. When a child partition tries to access memory the hypervisor / virtualization stack re-maps the request to different memory locations.

clone
A VM clone is an exact copy of a VM at a specific moment in time. The most common use of a VM clone is for mass deployment of standardized VMs, called VM templates. VM clones also come in handy for test and development; because they allow use of a real workload without affecting the production environment. A VM clone is not appropriate for backup, disaster recovery, or other data protection methods.

clustered file system
A clustered file system organizes files, stored data, and access for multiple servers in a cluster. Clustered file systems are most useful when clusters work together and require shared access, which individual file systems do not provide. A Windows or Linux clustered file system can also identify and isolate defective nodes in a cluster. A Windows clustered file system will isolate the node logically, while a Linux clustered file system will use a utility to power down the node.

compact
To reduce the size of a dynamically expanding virtual hard disk by removing unused space from the .vhd file. See also dynamically expanding virtual hard disk

differencing disk
A virtual hard disk that is associated with another virtual hard disk in a parent-child relationship. The differencing disk is the child and the associated virtual hard disk is the parent.

differencing virtual hard disk (diffdisk)
A virtual hard disk that stores the changes or "differences" to an associated parent virtual hard disk for the purpose of keeping the parent intact. The differencing disk is a separate .vhd file (that may be stored in a separate location) that is associated with the .vhd file of the parent disk. These disks are often referred to as "children" or "child" disks to distinguish them from the "parent" disk. There can be only one parent disk in a chain of differencing disks. There can be one or more child disks in a differencing disk chain of disks that are "related" to each other. Changes continue to accumulate in the differencing disk until it is merged to the parent disk. See also virtual hard disk. A common use for differencing disks is to manage storage space on a virtualization server. For example, you can create a base parent disk- such as a Windows 2008 R2 Standard base image - and use it as the foundation for all other guest virtual machines and disks that will be based on Windows Server 2008 R2.

dynamically expanding virtual hard disk (dynamic VHD, DVHD)
A virtual hard disk that grows in size each time it is modified. This type of virtual hard disk starts as a 3 KB .vhd file and can grow as large as the maximum size specified when the file was created. The only way to reduce the file size is to zero out the deleted data and then compact the virtual hard disk. See also virtual hard disk, VHD.

external virtual network
A virtual network that is configured to use a physical network adapter. These networks are used to connect virtual machines to external networks. See also internal virtual network, private virtual network.

failover
In server clusters, failover is the process of taking resource groups offline on one node and bringing them online on another node.

Fibre Channel
A high-speed network technology (commonly running at 2-, 4-, 8- and 16-gigabit speeds) primarily used for storage networking.

fragmentation
The scattering of parts of the same disk file over different areas of the disk.

guest operating system
This is the operating system/runtime environment that is present inside a partition. Historically with Virtual Server / Virtual PC, in a host operating system and a guest operating system where the host ran on the physical hardware and the guest ran on the host. In Hyper-V, all operating systems on the physical computer are running on top of the hypervisor so the correct equivalent terms are parent guest operating system and child guest operating system. Many find these terms confusing and instead use physical operating system and guest operating system to refer to parent and child guest operating systems, respectively.

guests and hosts
A guest virtual machine and host server are the two main building blocks of virtualization. The guest virtual machine is a file that contains a virtualized operating system and application, and the host server is the hardware on which it runs. The other important component is the hypervisor—the software that creates the guest virtual machine and lets it interact with the host server. The hypervisor also makes the host server run multiple guest virtual machines.

historical storage system (Historian)
The time series data storage system that compresses and stores high volumes of time series data for later retrieval. The standard Historian is the Historian.

Internet Small Computer Storage Interface (iSCSI)
Takes standard SCSI disk commands and, instead of executing them over a local SCSI connection, encapsulates them in TCP/IP packets for transmission over Ethernet. These low-level commands do not directly interact with files, but rather with arbitrary blocks of data on disk. The system reading and writing the data implements a file system on top of the ISCSI share, or LUN (logical unit number), to be able to read and write data for files. In the case of vSphere, this file system is called VMFS. Hyper-V also implements a file system on top of ISCSI LUNs. This allows the Hypervisor to implement a file system (like VMFS) that is optimized for the I/O needs of the hypervisor.

hypervisor
The hypervisor is to Hyper-V what the kernel is to Windows. The hypervisor is the lowest level component that is responsible for interaction with core hardware. It is responsible for creating, managing, and destroying partitions. It directly controls access to processor resource and enforces an externally-delivered policy on memory and device access. The hypervisor is just over 100k in size and the entire Hyper-V role is around 100mb in size. A full installation of Windows Server 2008 with Hyper-V will be multiple gigabytes in size. After you have installed the Hyper-V role, the hypervisor is loaded as a boot critical device.

live migration
Virtual machine live migration is the process of moving a VM from one host server to another without shutting down the application. The benefits of virtual machine live migration are some of the biggest selling points for virtualization, affecting business continuity, disaster recovery, and server consolidation. Virtual machine live migration is a feature in all of the major virtualization platforms, including VMware vSphere, Microsoft Hyper-V R2, and Citrix Systems XenServer.

logical processor
This is a single execution pipeline on the physical processor.Earlier, if someone told you that they had a two-processor system, you would know exactly what they had. Today, if someone told you they had a two-processor system, you do not know how many cores each processor has, or if hyperthreading is present. A two-processor computer with hyperthreading would actually have four execution pipelines, or four logical processors. A two-processor computer with quad-core processors would, in turn, have eight logical processors.

management operating system
The operating system that was originally installed on the physical machine when the Hyper-V role was enabled. After installing the Hyper-V role, this operating system is moved into the parent partition. The management operating system automatically launches when you reboot the physical machine. The management operating system actually runs in a special kind of virtual machine that can create and manage the virtual machines that are used to run workloads and/or different operating systems. These virtual machines are sometimes also called child partitions. The management operating system provides management access to the virtual machines and an execution environment for the Hyper-V services. The management operating system also provides the virtual machines with access to the hardware resources it owns.

memory overcommit
A hypervisor can let a guest VM use more memory space than that available in the host server. This feature is called memory overcommit. Memory overcommit is possible because most VMs use only a little bit of their allocated physical memory. That frees up memory for the few VMs that need more. Hypervisors with memory overcommit features can identify unused memory and reallocate it to more memory-intensive VMs as needed.

Network-Attached Storage (NAS)
Network-attached storage (NAS), in contrast to SAN, uses file-based protocols such as NFS or SMB / CIFS where it is clear that the storage is remote, and computers request a portion of an abstract file rather than a disk block.

Network File System (NFS)
A file system originally created by Sun Microsystems as a way to allow multiple clients to access files on a central network storage device. When the Hypervisor accesses data on an NFS share, it accesses the files directly because the protocol itself provides the file system.

Network Load Balancing (NLB)
A Windows network component that uses a distributed algorithm to load-balance IP traffic across a number of hosts, helping to enhance the scalability and availability of mission-critical, IP-based services.

network virtualization
Network virtualization lets you combine multiple networks into one, divide one network into many and even create software-only networks between VMs. The basis of network virtualization is virtual network software, to which there are two approaches: internal and external. Internal network virtualization uses virtual network software to emulate network connectivity among VMs inside a host server. External network virtualization virtual network software to consolidate multiple physical networks or create several virtual networks out of one physical network.

NTFS
An advanced file system that provides performance, security, reliability, and advanced features that are not found in any version of the file allocation table (FAT).

parent partition
The parent partition can call hypervisor and request for new partitions to be created. There can only be one parent partition. In the first release of Hyper-V, the parent and root partitions are one and the same.
partition

A partition is the basic entity that is managed by the hypervisor. It is an abstract container that consists of isolated processor and memory resources with policies on device access. A partition is a lighter weight concept than a virtual machine and could be used outside the context of virtual machines to provide a highly isolated execution environment.

physical computer
The computer, or more specifically, the hardware that is running the Hyper-V role.

physical processor
It is the squarish chip that you put in your computer to make it run. This is sometimes also referred to as a "package" or a "socket".

private virtual network
A virtual network without a virtual network adapter in the management operating system. It allows communication only between virtual machines on the same physical server.

processor topology
This is the concept by which your logical processors correlate to your physical processors. For example, a two processor, quad-core system and a four-processor dual-core system both have eight logical processors but they have different processor topologies.

P2V
A physical-to-virtual server migration, also known as a P2V server migration, is the process of converting a physical workload into a VM. To perform a physical-to-virtual server migration, copy bits from the physical disk to the VM, inject drivers, then modify other bits to support the drivers. Some operating systems and virtual server migration tools let you perform a P2V server migration while the host is running, but others require a shutdown.

release key combination
The key combination (CTRL+ALT+LEFT ARROW by default) that must be pressed to move keyboard and mouse focus from a guest operating system back to the physical computer.

root partition
This is the first partition on the computer. This is the partition that is responsible for starting the hypervisor. It is also the only partition that has direct access to memory and devices.

saved state
A manner of storing a virtual machine so that it can be quickly resumed (similar to a hibernated laptop). When you place a running virtual machine in a saved state, Virtual Server and Hyper-V stop the virtual machine, write the data that exists in memory to temporary files, and stop the consumption of system resources. Restoring a virtual machine from a saved state returns it to the same condition it was in when its state was saved.

small computer system interface (SCSI)
A standard high-speed parallel interface used for connecting microcomputers to peripheral devices, such as hard disks and printers, and to other computers and local area networks (LANs).

snapshot
A VM snapshot backup is the most common way to protect a virtual machine. A VM snapshot is a copy of the state of a VM (and any virtual disks assigned to it) as it exists in server memory at a specific moment. The snapshot is usually saved to the SAN, where it can be recovered in case of a failure. Regular VM snapshot backups can significantly reduce recovery point objectives.

storage area network (SAN)
A set of interconnected devices, such as disks and tapes, and servers that are connected to a common communication and data transfer infrastructure, such as Fibre Channel.

storage array
A disk storage system which contains multiple disk drives. It is differentiated from a disk enclosure in that an array has cache memory and advanced functionality, like RAID and virtualization.

storage virtualization
Storage virtualization separates the operating system from physical disks used for storage, making the storage location independent. The benefits of storage virtualization include more efficient storage use and better management. Dynamic provisioning is similar to storage virtualization, but it still requires more traditional storage management.

system center virtual machine manager (SCVMM)
A centralized management console that helps you manage and administer a virtual environment.

vfd or virtual floppy disk
The file format for a virtual floppy disk. See also virtual floppy disk.

vhd or virtual hard disk
The file format for a virtual hard disk, the storage medium for a virtual machine. It can reside on any storage topology that the management operating system can access, including external devices, storage area networks, and network-attached storage.

virtual hardware
The computing resources that the host server assigns to a guest VM make up the virtual hardware platform. The hypervisor controls the virtual hardware platform and allows the VM to run on any host server, regardless of the physical hardware. The virtual hardware platform includes memory, processor cores, optical drives, network adapters, I/O ports, a disk controller and virtual hard disks. Virtualization lets a user adjust the levels of these resources on each VM as needed.

virtual machine
A virtual machine (VM) is a file that includes an application and an underlying operating system combines with a physical host server and a hypervisor to make server virtualization possible. A virtual machine is a super-set of a child partition. A virtual machine is a child partition combined with virtualization stack components that provide functionality, such as access to emulated devices, and features like being able to save state a virtual machine. As a virtual machine is essentially a specialized partition, the terms "partition" and "virtual machine" is often used interchangeably. But, while a virtual machine will always have a partition associated with it, a partition may not always be a virtual machine.

virtual machine bus
A communications line used in Hyper-V by virtual machines and certain types of virtual devices. The virtual devices that use virtual machine bus have been optimized for use in virtual machines.

virtual machine configuration
The configuration of the resources assigned to a virtual machine. Examples include devices such as disks and network adapters, as well as memory and processors.

Virtual machine connection
A Hyper-V management tool that allows a running virtual machine to be managed through an interactive session.

virtual machine management service
The SCVMM service that provides management access to virtual machines.

virtual machine monitoring
Virtual machine monitoring actually means virtual machine performance monitoring. Virtual machine performance monitoring tools keep tabs on the state of VMs in an environment. Though it is possible to monitor the VM performance from within, but it's recommended to monitor it from outside the VM.

virtual machine snapshot
A virtual machine snapshot is a point in time image of a virtual machine that includes its disk, memory and device state at the time that the snapshot was taken. At any time can be used to return a virtual machine to a specific moment in time, at any time. Virtual machine snapshots can be taken irrespective of the state or type of child guest operating system being used.

virtual network
A virtual version of a physical network switch. A virtual network can be configured to provide access to local or external network resources for one or more virtual machines.

virtual network manager
The Hyper-V component used to create and manage virtual networks.

virtualization server
A physical computer with the Hyper-V role installed. This server contains the management operating system and it provides the environment for creating and running virtual machines. Sometimes referred to as a server running Hyper-V.

virtualization stack
The virtualization stack is everything else that makes up Hyper-V. This is the user interface, management services, virtual machine processes, emulated devices.

virtual processor
A virtual processor is a single logical processor that is exposed to a partition by the hypervisor. Virtual processors can be mapped to any of the available logical processors in the physical computer and are scheduled by the hypervisor to allow you to have more virtual processors than you have logical processors.

virtual switch
A virtual switch is the key to network virtualization. It connects physical switches to VMs through physical network interface cards and ports. A virtual switch is similar to a virtual bridge, which many virtualization platforms use, but it is more advanced. Virtual LANs, EtherChannel and additional virtual networking tools are only available in a virtual switch. Some virtual switches even offer their own security features.

virtualization WMI provider
The WMI provider for virtualization that can be used with the hypervisor API to enable developers and scripters to build custom tools, utilities, and enhancements for the virtualization platform.

VMDK
The Virtual Machine Disk (VMDK) file format is used to identify VMware virtual machines. (In virtualization, the hypervisor creates a VM file that consists of an operating system instance, an application and other associated components.) Other platforms that support the VMDK file format include Sun Microsystems xVM, Oracle VirtualBox, and QEMU. It competes with Microsoft's Virtual Hard Disk format, which is used in Virtual Server and Hyper-V.